



注意：この日本語版文書は参考資料としてご利用ください。  
最新情報は必ずオリジナルの英語版をご参照願います。

# MICROCHIP PIC32MZ グラフィック (DA) ファミリ

## XLP テクノロジ採用 32 ビット グラフィック アプリケーション向け MCU (最大 2 MB ライブアップデート フラッシュ / 640 KB SRAM/ 32 MB DDR2 SDRAM 内蔵)

### 動作条件

- 2.2 ~ 3.6 V, -40 ~ +85 °C, DC ~ 200 MHz
- 2.2 ~ 3.6 V, -40 ~ +105 °C (発売予定)

### コア : 200 MHz / 330 DMIPS MIPS32<sup>®</sup> microAptiv<sup>™</sup>

- 32 KB の I キャッシュと 32 KB の D キャッシュ
- 組み込み OS の実行を最適化する MMU (メモリ管理ユニット)
- コードサイズを最大 35% 削減する microMIPS<sup>™</sup> モード
- DSP 機能を強化したコア
  - 4 つの 64 ビット アキュムレータ
  - シングルサイクルの MAC、飽和、小数演算
- コード効率の高い (C およびアセンブリ) アーキテクチャ

### クロックの管理

- プログラマブル PLL とオシレータ クロック源
- DDR2 専用 PLL
- フェイルセーフ クロック監視機能
- 独立したウォッチドッグ タイマとデッドマンタイマ
- 高速な復帰と起動

### 電源管理

- 各種の電源管理オプション (VBAT、ディープスリープ、スリープ、アイドル) による消費電力の大幅削減
- ディープスリープ電流 : < 1 μA (typ.)
- 統合した POR と BOR
- VDDIO に対するプログラマブル高/低電圧検出 (HLVD) と VDDR1V8 に対する高電圧検出 (HVD)

### メモリ インターフェイス

- DDR2 SDRAM インターフェイス (最高 DDR2-400)
- SD/SDIO/eMMC バス インターフェイス (最大 50 MHz)
- シリアルクワッド インターフェイス (最大 80 MHz)
- 外部バス インターフェイス (最大 50 MHz)

### グラフィック機能

- 3 レイヤ グラフィック コントローラ (最大 24 ビットカラーをサポート)
- 高性能 2D グラフィック処理ユニット (GPU)

### オーディオ インターフェイス

- オーディオデータ通信 : I<sup>2</sup>S、LJ、RJ
- オーディオ制御インターフェイス : SPI、I<sup>2</sup>C
- オーディオ マスタクロック : USB に同期した小数分周クロック周波数

### 高速通信インターフェイス (専用 DMA を使用)

- USB 2.0 互換ハイスピード On-The-Go (OTG) コントローラ
- MII および RMII インターフェイスを備えた 10/100 Mbps Ethernet MAC

### セキュリティ機能

- 乱数生成器 (RNG) を備えた暗号エンジンによるデータの暗号化 / 復号と認証 (AES、3DES、SHA、MD5、HMAC)
- 先進のメモリ保護機能
  - 周辺モジュールとメモリ領域のアクセス制御

### ダイレクトメモリアクセス (DMA)

- 各チャンネルで自動的にデータサイズを検出
- プログラマブル巡回冗長検査 (CRC)

### パッケージ

タイプ	LFBGA		LQFP
ピン数	169	288	176
I/O ピン数 (最大)	120	120	120
コンタクト / ピンピッチ	0.8 mm	0.8 mm	0.4 mm
寸法	11x11 mm	15x15 mm	20x20 mm

### 先進のアナログ機能

- 12 ビット ADC モジュール
  - 最大 6 個の ADC 回路 (5 つが専用、1 つが共有) による 18 Msps の変換レート
  - 最大 45 個のアナログ入力
  - スリープおよびアイドル中に動作可能
  - 複数のトリガ源
  - 6 個のデジタル コンパレータと 6 個のデジタルフィルタ
- 2 個のコンパレータ (参照電圧レベルを 32 通りに設定可能)
- 温度センサ (精度 : ±2 °C)
- 充電時間計測ユニット (CTMU)

### 通信インターフェイス

- 専用 DMA チャンネルを備えた 2 個の CAN モジュール
  - CAN 2.0B Active (DeviceNet<sup>™</sup> アドレッシングをサポート)
- 6 個の UART モジュール (25 Mbps)
  - LIN 1.2 および IrDA<sup>®</sup> プロトコルをサポート
- 4 個の 4 線式 SPI モジュール (最大 50 MHz)
- 追加の SPI モジュールとして構成可能な SQI モジュール (最大 80 MHz)
- SMBus をサポートする 5 個の I<sup>2</sup>C モジュール (最大 1 Mbaud)
- パラレル マスタポート (PMP)
- ピン割り当てを変更できるペリフェラル ピン セレクト (PPS)

### タイマ / 出力コンペア / 入力キャプチャ

- 9 個の 16 ビットタイマ / カウンタと最大 4 個の 32 ビットタイマ / カウンタ
- 9 個の出力コンペア (OC) モジュール
- 9 個の入力キャプチャ (IC) モジュール
- リアルタイム クロック / カレンダー (RTCC) モジュール

### 入出力

- 5 V 対応ピン (最大 32 mA のソース / シンク電流)
- オープンドレイン、プルアップ、プルダウンを選択可能
- 選択可能なスルーレート制御
- 全 I/O ピンでの外部割り込み
- PPS による機能割り当て変更

### 信頼性認定とクラス B サポート

- AEC-Q100 REVG (グレード 2: -40 ~ 105 °C) (認定取得予定)
- クラス B 安全ライブラリ (IEC 60730)
- バックアップ内部オシレータ

### デバッグ開発サポート

- インサーキットおよびイン アプリケーション プログラミング
- 4 線式 MIPS<sup>®</sup> 拡張 JTAG インターフェイス
- 制限なしのソフトウェア ブレークポイントと 12 個の複合ブレークポイント
- IEEE 標準 1149.2 互換 (JTAG) バウンダリ スキャン
- 動作に影響しないハードウェアベースの命令トレース

### 統合ソフトウェア ライブラリ / ツール

- C/C++ コンパイラ (ネイティブの DSP / 浮動小数点サポート)
- MPLAB<sup>®</sup> Harmony 統合ソフトウェア フレームワーク
- TCP/IP、USB、グラフィック、mTouch<sup>™</sup> ミドルウェア
- MFi、Android<sup>™</sup>、Bluetooth<sup>®</sup> オーディオ フレームワーク
- RTOS カーネル : Express Logic ThreadX、FreeRTOS<sup>™</sup>、OPENRTOS<sup>®</sup>、Micrium<sup>®</sup> μC/OS<sup>™</sup>、SEGGER embOS<sup>®</sup>

# PIC32MZ グラフィック (DA) ファミリ

表 1: PIC32MZ DA 全デバイスに共通の機能

160	割り当て変更可能周辺モジュール						45	2	Y	Y	5	Y	Y	Y	Y	Y	Y	Y	120	Y	Y
	ブートフラッシュメモリ (KB)	割り当て変更可能ピン	タイマ <sup>(1)</sup> /キヤプチャ/コンペア	UART	SPI/I <sup>2</sup> S	CAN 2.0B															
160	47	9/9/9	6	6	2	5	45	2	Y	Y	5	Y	Y	Y	Y	Y	Y	Y	120	Y	Y

Note 1: 9個あるタイマの内の8個が割り当て変更可能です。  
 2: 5個ある外部割り込みの内の4個が割り当て変更可能です。

表 2: 169 ピン LFBGA PIC32MZ DA の機能

デバイス	プログラムメモリ (KB)	データメモリ (KB)	DDR2 コントローラ インターフェイス (内部/外部)	DDR2 SDRAM 容量 (MB)	暗号/乱数生成	DMA チャンネル (設定可能/専用)
PIC32MZ1025DAA169	1024	256	No	—	N	8/24
PIC32MZ1025DAB169					Y	8/26
PIC32MZ1064DAA169		640			N	8/24
PIC32MZ1064DAB169					Y	8/26
PIC32MZ2025DAA169	2048	256			N	8/24
PIC32MZ2025DAB169					Y	8/26
PIC32MZ2064DAA169		640			N	8/24
PIC32MZ2064DAB169					Y	8/26
PIC32MZ1025DAG169	1024	256	Yes (INT)	32	N	8/24
PIC32MZ1025DAH169					Y	8/26
PIC32MZ1064DAG169		640			N	8/24
PIC32MZ1064DAH169					Y	8/26
PIC32MZ2025DAG169	2048	256			N	8/24
PIC32MZ2025DAH169					Y	8/26
PIC32MZ2064DAG169		640			N	8/24
PIC32MZ2064DAH169					Y	8/26

表 3: 176 ピン LQFP PIC32MZ DA の機能

デバイス	プログラムメモリ (KB)	データメモリ (KB)	DDR2 コントローラ インターフェイス (内部/外部)	DDR2 SDRAM 容量 (MB)	暗号/乱数生成	DMA チャンネル (設定可能/専用)
PIC32MZ1025DAA176	1024	256	No	—	N	8/24
PIC32MZ1025DAB176					Y	8/26
PIC32MZ1064DAA176		640			N	8/24
PIC32MZ1064DAB176					Y	8/26
PIC32MZ2025DAA176	2048	256			N	8/24
PIC32MZ2025DAB176					Y	8/26
PIC32MZ2064DAA176		640			N	8/24
PIC32MZ2064DAB176					Y	8/26
PIC32MZ1025DAG176	1024	256	Yes (INT)	32	N	8/24
PIC32MZ1025DAH176					Y	8/26
PIC32MZ1064DAG176		640			N	8/24
PIC32MZ1064DAH176					Y	8/26
PIC32MZ2025DAG176	2048	256			N	8/24
PIC32MZ2025DAH176					Y	8/26
PIC32MZ2064DAG176		640			N	8/24
PIC32MZ2064DAH176					Y	8/26

表 4: 288 ピン LFBGA PIC32MZ DA の機能

デバイス	プログラムメモリ (KB)	データメモリ (KB)	DDR2 コントローラ インターフェイス (内部/外部)	暗号/乱数生成	DMA チャンネル (設定可能/専用)
PIC32MZ1025DAA288	1024	256	Yes (EXT)	N	8/24
PIC32MZ1025DAB288				Y	8/26
PIC32MZ1064DAA288		640		N	8/24
PIC32MZ1064DAB288				Y	8/26
PIC32MZ2025DAA288	2048	256		N	8/24
PIC32MZ2025DAB288				Y	8/26
PIC32MZ2064DAA288		640		N	8/24
PIC32MZ2064DAB288				Y	8/26

# PIC32MZ グラフィック (DA) ファミリ

## デバイスピン一覧

表 5: 169 ピンデバイスのピン名

169ピンLFBGA (底面)		ピン識別マーク	
PIC32MZ1025DAA169			
PIC32MZ1025DAB169			
PIC32MZ1064DAA169			
PIC32MZ1064DAB169			
PIC32MZ2025DAA169			
PIC32MZ2025DAB169			
PIC32MZ2064DAA169			
PIC32MZ2064DAB169			
PIC32MZ1025DAG169			
PIC32MZ1025DAH169			
PIC32MZ1064DAG169			
PIC32MZ1064DAH169			
PIC32MZ2025DAG169			
PIC32MZ2025DAH169			
PIC32MZ2064DAG169			
PIC32MZ2064DAH169			

ボール/ ピン番号	ピンのフルネーム	ボール/ ピン番号	ピンのフルネーム
A1	未接続	C5	EBIA2/AN23/C2INC/RPG9/PMA2/RG9
A2	Vbus	C6	TDO/AN31/RPF12/RF12
A3	RPF2/SDA3/RF2	C7	EBID7/AN15/PMD7/RE7
A4	EBID1/AN39/PMD1/RE1	C8	AVss
A5	AN21/RG15	C9	VDDCORE
A6	TDI/AN17/SCK5/RF13	C10	VREF+/CVREF+/AN28/RA10
A7	EBIWE/AN34/RPC3/PMWR/RC3	C11	CVREFOUT/AN5/RPB10/RB10
A8	EBID12/AN10/RPC2/PMD12/RC2	C12	PGED1/AN0/RPB0/CTED2/RB0
A9	EBID10/AN4/RPB8/PMD10/RB8	C13	SOSCI/RPC13 <sup>(6)</sup> /RC13 <sup>(6)</sup>
A10	AN8/RPB3/RB3	D1	TRD3/SDDATA3/SQID3/RA7
A11	EBIA5/AN7/PMA5/RA5	D2	TMS/SDCD/RA0
A12	AN2/C1INB/RB4	D3	USBID
A13	AN1/C2INB/RPB2/RB2	D4	AN20/RH4
B1	D-	D5	AN13/C1INC/RPG7/SDA4/RG7
B2	VUSB3v3	D6	AN26/RPE9/RE9
B3	EBID4/AN18/PMD4/RE4	D7	PGEC2/RPB6/RB6
B4	VDDCORE	D8	AVss
B5	AN30/C2IND/RPG8/SCL4/RG8	D9	AVDD
B6	VDDIO	D10	VBAT
B7	EBID5/AN12/RPC1/PMD5/RC1	D11	AN45/RPB5/RB5
B8	EBIOE/AN19/RPC4/PMRD/RC4	D12	PGED2/C1INA/AN46/RPB7/RB7
B9	PGEC1/AN9/RPB1/CTED1/RB1	D13	SOSCO/RPC14 <sup>(6)</sup> /T1CK/RC14 <sup>(6)</sup>
B10	AN3/C2INA/RPB15/OCFB/RB15	E1	TRD2/SDDATA2/SQID2/RG14
B11	VREF-/CVREF-/AN27/RA9	E2	TRD0/SDDATA0/SQID0/RG13
B12	EBIA7/AN47/HLVDIN/RPB9/PMA7/RB9	E3	TRD1/SDDATA1/SQID1/RG12
B13	AN6/RB12	E4	TRCLK/SDCK/SQICLK/RA6
C1	D+	E5	AN14/C1IND/SCK2/RG6
C2	Vss	E6	AN25/RPE8/RE8
C3	INT0/RH14	E7	AN49/RB11
C4	EBID0/PMD0/RE0	E8	GD20/EBIA22/RJ3

- Note
- 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使えます。利用可能な周辺モジュールは表 1 と表 2 を参照してください。制約については 12.4 「ペリフェラルピンセレクト (PPS)」を参照してください。
  - 2: 全ての I/O ポートピン (RAX ~ RKX) は、状態変化通知ピン (CNAX ~ CNKX) として使えます。詳細は 12.0 「I/Oポート」を参照してください。
  - 3: 網掛けしたピンは 5V 対応です。
  - 4: DDR 非実装デバイスの場合、このピンは 20kΩ 抵抗を介して Vss に接続する必要があります。
  - 5: DDR 非実装デバイスの場合、このピンは未接続 (NC) です。
  - 6: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

表 5: 169 ピンデバイスのピン名 ( 続き )

**169ピンLFBGA (底面)**

PIC32MZ1025DAA169  
 PIC32MZ1025DAB169  
 PIC32MZ1064DAA169  
 PIC32MZ1064DAB169  
 PIC32MZ2025DAA169  
 PIC32MZ2025DAB169  
 PIC32MZ2064DAA169  
 PIC32MZ2064DAB169  
 PIC32MZ1025DAG169  
 PIC32MZ1025DAH169  
 PIC32MZ1064DAG169  
 PIC32MZ1064DAH169  
 PIC32MZ2025DAG169  
 PIC32MZ2025DAH169  
 PIC32MZ2064DAG169  
 PIC32MZ2064DAH169

ボール / ピン番号	ピンのフルネーム	ボール / ピン番号	ピンのフルネーム
E9	AN22/RPD14/RD14	H2	SCK4/RD10
E10	AN29/SCK3/RB14	H3	RTCC/RPD0/RD0
E11	TCK/AN24/RA1	H4	Vss1v8
E12	OSC1/CLKI/RC12	H5	VDDR1V8 <sup>(4)</sup>
E13	OSC2/CLKO/RC15	H6	VDDR1V8 <sup>(4)</sup>
F1	SDCMD/SQICS0/RPD4/RD4	H7	Vss
F2	SQICS1/RPD5/RD5	H8	Vss
F3	EBIA6/RPE5/PMA6/RE5	H9	VDDIO
F4	DDRVREF <sup>(5)</sup>	H10	GD13/EBIA18/RK4
F5	Vss	H11	EBIA3/AN11/PMA3/RK2
F6	EBID6/AN16/PMD6/RE6	H12	SDWP/EBIRP/RH2
F7	AN48/CTPLS/RB13	H13	EBIA0/PMA0/RJ15
F8	GD18/EBIBS1/RJ10	J1	GD7/EBIA12/RPD12/PMA12/RD12
F9	GD9/EBIBS0/RJ12	J2	GD22/EBIA13/PMA13/RD13
F10	EBIRDY3/AN32/RJ2	J3	RPF8/SCL3/RF8
F11	AN33/SCK6/RD15	J4	Vss1v8
F12	HSYNC/EBICS1/RJ5	J5	VDDR1V8 <sup>(4)</sup>
F13	VSYSN/EBICS0/RJ4	J6	VDDR1V8 <sup>(4)</sup>
G1	SCK1/RD1	J7	Vss
G2	GD10/EBIA14/RPD2/PMA14/PMCS1/RD2	J8	Vss
G3	GD11/EBIA15/RPD3/PMA15/PMCS2/RD3	J9	VDDIO
G4	Vss1v8	J10	GD14/EBIA19/RK5
G5	Vss	J11	EBIA1/AN38/PMA1/RK1
G6	Vss	J12	EBIA4/AN36/PMA4/RH7
G7	Vss	J13	AN35/RH3
G8	Vss	K1	MCLR
G9	VDDIO	K2	GD16/EBID8/RPF5/SCL5/PMD8/RF5
G10	GD8/EBID11/PMD11/RJ14	K3	GD5/EBIA10/RPF1/PMA10/RF1
G11	GCLK/EBICS2/RJ6	K4	Vss1v8
G12	GD0/EBID13/PMD13/RJ13	K5	VDDR1V8 <sup>(4)</sup>
G13	GEN/EBICS3/RJ7	K6	VDDR1V8 <sup>(4)</sup>
H1	GD2/EBID15/RPD9/PMD15/RD9	K7	Vss

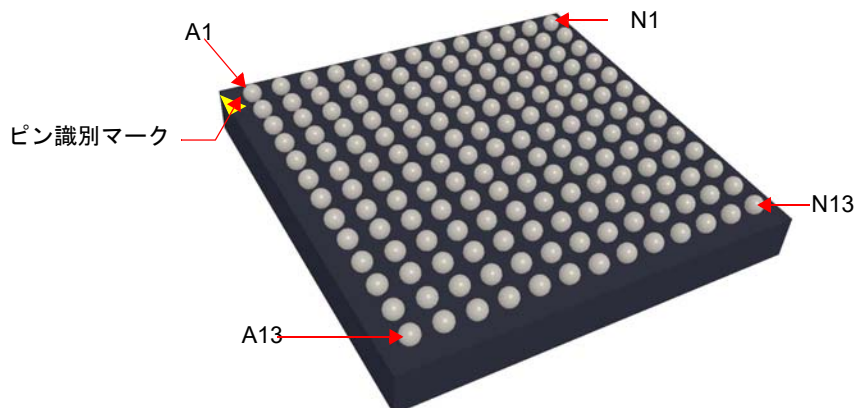
- Note 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使えます。利用可能な周辺モジュールは表 1 と表 2 を参照してください。制約については 12.4 「ペリフェラル ピンセレクト (PPS)」を参照してください。
- 2: 全ての I/O ポートピン (RAX ~ RKx) は、状態変化通知ピン (CNAX ~ CNKx) として使えます。詳細は 12.0 「I/O ポート」を参照してください。
- 3: 網掛けしたピンは 5V 対応です。
- 4: DDR 非実装デバイスの場合、このピンは 20kΩ 抵抗を介して Vss に接続する必要があります。
- 5: DDR 非実装デバイスの場合、このピンは未接続 (NC) です。
- 6: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

表 5: 169 ピンデバイスのピン名 ( 続き )

## 169ピンLFBGA (底面)

PIC32MZ1025DAA169  
 PIC32MZ1025DAB169  
 PIC32MZ1064DAA169  
 PIC32MZ1064DAB169  
 PIC32MZ2025DAA169  
 PIC32MZ2025DAB169  
 PIC32MZ2064DAA169  
 PIC32MZ2064DAB169  
 PIC32MZ1025DAG169  
 PIC32MZ1025DAH169  
 PIC32MZ1064DAG169  
 PIC32MZ1064DAH169  
 PIC32MZ2025DAG169  
 PIC32MZ2025DAH169  
 PIC32MZ2064DAG169  
 PIC32MZ2064DAH169



ボール/ ピン番号	ピンのフルネーム	ボール/ ピン番号	ピンのフルネーム
K8	VSS	M5	ERXDV/ECRSRV/RH13
K9	VDDIO	M6	ECOL/RH10
K10	EMDIO/RJ1	M7	ETXD3/RH1
K11	ETXEN/RPD6/RD6	M8	ETXD2/RH0
K12	GD23/EBIA16/RK0	M9	ETXD1/RJ9
K13	EBIRDY2/AN37/RH11	M10	ETXCLK/RPD7/RD7
L1	GD6/EBIA11/RPF0/PMA11/RF0	M11	RPA14/SCL1/RA14
L2	GD21/EBIA23/RH15	M12	GD19/EBIA21/RK7
L3	GD17/EBID9/RPF4/SDA5/PMD9/RF4	M13	GD15/EBIA20/RK6
L4	VSS1V8	N1	VDDCORE
L5	VSS1V8	N2	GD3/EBIA8/RPG0/PMA8/RG0
L6	VDDIO	N3	EBID2/PMD2/RE2
L7	VDDIO	N4	ERXD2/RH6
L8	VDDCORE	N5	ECRS/RH12
L9	VDDIO	N6	ERXD3/RH9
L10	ETXERR/RJ0	N7	ERXD0/RH8
L11	GD1/EBID14/PMD14/RA4	N8	ERXCLK/EREFCLK/RJ11
L12	SCL2/RA2	N9	ETXD0/RJ8
L13	GD12/EBIA17/RK3	N10	EMDC/RPD11/RD11
M1	ERXERR/RPF3/RF3	N11	RPA15/SDA1/RA15
M2	GD4/EBIA9/RPG1/PMA9/RG1	N12	EBIRDY1/SDA2/RA3
M3	EBID3/RPE3/PMD3/RE3	N13	未接続
M4	ERXD1/RH5		

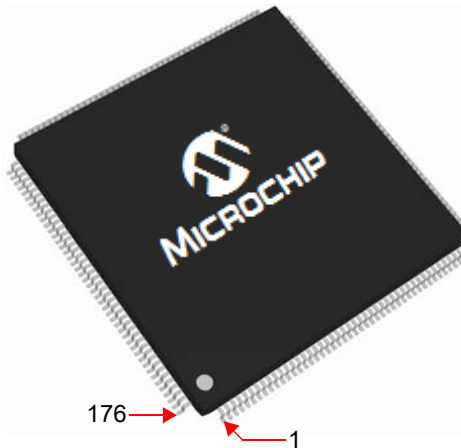
- Note
- 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使用します。利用可能な周辺モジュールは表 1 と表 2 を参照してください。制約については 12.4 「ペリフェラル ピンセレクト (PPS)」 を参照してください。
  - 2: 全ての I/O ポートピン (RAX ~ RKX) は、状態変化通知ピン (CNAX ~ CNKX) として使えます。詳細は 12.0 「I/O ポート」 を参照してください。
  - 3: 網掛けしたピンは 5 V 対応です。
  - 4: DDR 非実装デバイスの場合、このピンは 20k Ω 抵抗を介して VSS に接続する必要があります。
  - 5: DDR 非実装デバイスの場合、このピンは未接続 (NC) です。
  - 6: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

表 6: 176 ピンデバイスのピン名

## 176ピンLQFP (上面)

PIC32MZ1025DAA176  
 PIC32MZ1025DAB176  
 PIC32MZ1064DAA176  
 PIC32MZ1064DAB176  
 PIC32MZ2025DAA176  
 PIC32MZ2025DAB176  
 PIC32MZ2064DAA176  
 PIC32MZ2064DAB176  
 PIC32MZ1025DAG176  
 PIC32MZ1025DAH176  
 PIC32MZ1064DAG176  
 PIC32MZ1064DAH176  
 PIC32MZ2025DAG176  
 PIC32MZ2025DAH176  
 PIC32MZ2064DAG176  
 PIC32MZ2064DAH176



ピン番号	ピンのフルネーム	ピン番号	ピンのフルネーム
1	VREF-/CVREF-/AN27/RA9	37	Vss
2	VREF+/CVREF+/AN28/RA10	38	VDDIO
3	AVDD	39	VDDCORE
4	AVDD	40	EBID0/PMD0/RE0
5	AVss	41	RPF2/SDA3/RF2
6	AVss	42	INT0/RH14
7	AN3/C2INA/RPB15/OCFB/RB15	43	EBID4/AN18/PMD4/RE4
8	AN8/RPB3/RB3	44	No Connect
9	AN48/CTPLS/RB13	45	Vbus
10	EBID10/AN4/RPB8/PMD10/RB8	46	VUSB3v3
11	PGEC1/AN9/RPB1/CTED1/RB1	47	VUSB3v3
12	AN49/RB11	48	Vss
13	PGEC2/RPB6/RB6	49	Vss
14	EBID12/AN10/RPC2/PMD12/RC2	50	D-
15	EBIWE/AN34/RPC3/PMWR/RC3	51	D+
16	EBIOE/AN19/RPC4/PMRD/RC4	52	USBID
17	EBID5/AN12/RPC1/PMD5/RC1	53	TMS/SDCD/RA0
18	VDDCORE	54	TRCLK/SDCK/SQICLK/RA6
19	VDDIO	55	TRD3/SDDATA3/SQID3/RA7
20	未接続	56	TRD1/SDDATA1/SQID1/RG12
21	Vss	57	VDDR1V8 <sup>(5)</sup>
22	Vss	58	VDDR1V8 <sup>(5)</sup>
23	EBID6/AN16/PMD6/RE6	59	VDDR1V8 <sup>(5)</sup>
24	EBID7/AN15/PMD7/RE7	60	VDDR1V8 <sup>(5)</sup>
25	AN25/RPE8/RE8	61	VDDR1V8 <sup>(5)</sup>
26	AN26/RPE9/RE9	62	VDDR1V8 <sup>(5)</sup>
27	TDO/AN31/RPF12/RF12	63	VDDR1V8 <sup>(5)</sup>
28	TDI/AN17/SCK5/RF13	64	TRD0/SDDATA0/SQID0/RG13
29	Vss	65	TRD2/SDDATA2/SQID2/RG14
30	AN14/C1IND/SCK2/RG6	66	DDRVREF <sup>(6)</sup>
31	AN13/C1INC/RPG7/SDA4/RG7	67	VDDR1V8 <sup>(5)</sup>
32	AN30/C2IND/RPG8/SCL4/RG8	68	VDDR1V8 <sup>(5)</sup>
33	EBIA2/AN23/C2INC/RPG9/PMA2/RG9	69	EBIA6/RPE5/PMA6/RE5
34	AN21/RG15	70	SDCMD/SQICS0/RPD4/RD4
35	AN20/RH4	71	SQICS1/RPD5/RD5
36	EBID1/AN39/PMD1/RE1	72	VDDR1V8 <sup>(5)</sup>

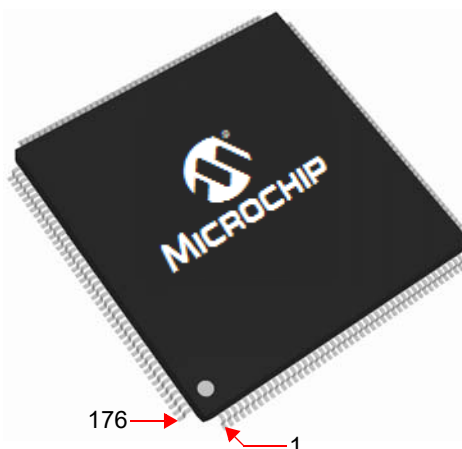
- Note
- 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使用します。利用可能な周辺モジュールは表 1 と表 3 を参照してください。制約については 12.4 「ペリフェラル ピンセレクト (PPS)」 を参照してください。
  - 2: 全ての I/O ポートピン (RAX ~ RKx) は、状態変化通知ピン (CNAX ~ CNKx) として使えます。詳細は 12.0 「I/Oポート」 を参照してください。
  - 3: 網掛けしたピンは 5V 対応です。
  - 4: デバイス底面の金属パッドは内部で Vss1V8 に接続しており、1.8V グランドに外部で接続する必要があります。
  - 5: DDR 非実装デバイスの場合、このピンは 20kΩ 抵抗を介して Vss に接続する必要があります。
  - 6: DDR 非実装デバイスの場合、このピンは未接続 (NC) です。
  - 7: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

表 6: 176 ピンデバイスのピン名 ( 続き )

## 176 ピンLQFP ( 上面 )

PIC32MZ1025DAA176  
 PIC32MZ1025DAB176  
 PIC32MZ1064DAA176  
 PIC32MZ1064DAB176  
 PIC32MZ2025DAA176  
 PIC32MZ2025DAB176  
 PIC32MZ2064DAA176  
 PIC32MZ2064DAB176  
 PIC32MZ1025DAG176  
 PIC32MZ1025DAH176  
 PIC32MZ1064DAG176  
 PIC32MZ1064DAH176  
 PIC32MZ2025DAG176  
 PIC32MZ2025DAH176  
 PIC32MZ2064DAG176  
 PIC32MZ2064DAH176



ピン番号	ピンのフルネーム	ピン番号	ピンのフルネーム
73	SCK1/RD1	109	ETXD3/RH1
74	GD10/EBIA14/RPD2/PMA14/PMCS1/RD2	110	ETXD2/RH0
75	GD11/EBIA15/RPD3/PMA15/PMCS2/RD3	111	ERXCLK/EREFCLK/RJ11
76	GD2/EBID15/RPD9/PMD15/RD9	112	ETXD1/RJ9
77	SCK4/RD10	113	ETXD0/RJ8
78	VDDR1V8 <sup>5)</sup>	114	EMDIO/RJ1
79	RTCC/RPD0/RD0	115	Vss
80	GD7/EBIA12/RPD12/PMA12/RD12	116	VDDCORE
81	GD22/EBIA13/PMA13/RD13	117	VDDIO
82	RPF8/SCL3/RF8	118	ETXERR/RJ0
83	Vss	119	EMDC/RPD11/RD11
84	VDDCORE	120	ETXCLK/RPD7/RD7
85	MCLR	121	ETXEN/RPD6/RD6
86	VDDIO	122	Vss
87	Vss	123	Vss
88	未接続	124	VDDIO
89	GD16/EBID8/RPF5/SCL5/PMD8/RF5	125	RPA15/SDA1/RA15
90	GD5/EBIA10/RPF1/PMA10/RF1	126	RPA14/SCL1/RA14
91	GD6/EBIA11/RPF0/PMA11/RF0	127	GD1/EBID14/PMD14/RA4
92	GD21/EBIA23/RH15	128	EBIRDY1/SDA2/RA3
93	ERXERR/RPF3/RF3	129	SCL2/RA2
94	Vss	130	GD19/EBIA21/RK7
95	GD4/EBIA9/RPG1/PMA9/RG1	131	GD15/EBIA20/RK6
96	GD3/EBIA8/RPG0/PMA8/RG0	132	GD14/EBIA19/RK5
97	GD17/EBID9/RPF4/SDA5/PMD9/RF4	133	GD13/EBIA18/RK4
98	EBID3/RPE3/PMD3/RE3	134	GD12/EBIA17/RK3
99	EBID2/PMD2/RE2	135	EBIA3/AN11/PMA3/RK2
100	ERXD1/RH5	136	EBIA1/AN38/PMA1/RK1
101	ERXD2/RH6	137	GD23/EBIA16/RK0
102	VDDIO	138	EBIRDY2/AN37/RH11
103	Vss	139	EBIA4/AN36/PMA4/RH7
104	ERXDV/ECRSRV/RH13	140	AN35/RH3
105	ECRS/RH12	141	SDWP/EBIRP/RH2
106	ECOL/RH10	142	EBIA0/PMA0/RJ15
107	ERXD3/RH9	143	GD8/EBID11/PMD11/RJ14
108	ERXD0/RH8	144	GD0/EBID13/PMD13/RJ13

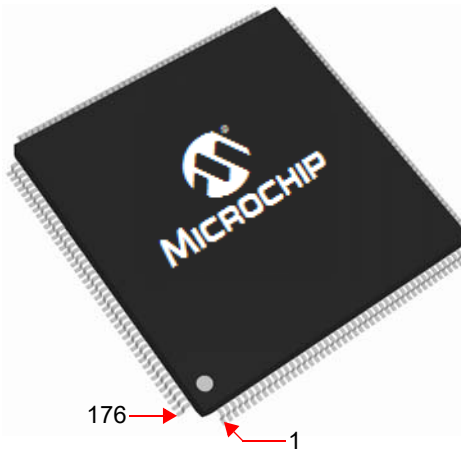
- Note
- 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使えます。利用可能な周辺モジュールは表 1 と表 3 を参照してください。制約については 12.4 「ペリフェラル ピンセレクト (PPS)」 を参照してください。
  - 2: 全ての I/O ポートピン (RAX ~ RKX) は、状態変化通知ピン (CNAX ~ CNKX) として使えます。詳細は 12.0 「I/O ポート」 を参照してください。
  - 3: 網掛けしたピンは 5V 対応です。
  - 4: デバイス底面の金属パッドは内部で Vss1v8 に接続しており、1.8V グランドに外部で接続する必要があります。
  - 5: DDR 非実装デバイスの場合、このピンは 20kΩ 抵抗を介して Vss に接続する必要があります。
  - 6: DDR 非実装デバイスの場合、このピンは未接続 (NC) です。
  - 7: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

表 6: 176 ピンデバイスのピン名 (続き)

## 176ピンLQFP (上面)

PIC32MZ1025DAA176  
 PIC32MZ1025DAB176  
 PIC32MZ1064DAA176  
 PIC32MZ1064DAB176  
 PIC32MZ2025DAA176  
 PIC32MZ2025DAB176  
 PIC32MZ2064DAA176  
 PIC32MZ2064DAB176  
 PIC32MZ1025DAG176  
 PIC32MZ1025DAH176  
 PIC32MZ1064DAG176  
 PIC32MZ1064DAH176  
 PIC32MZ2025DAG176  
 PIC32MZ2025DAH176  
 PIC32MZ2064DAG176  
 PIC32MZ2064DAH176



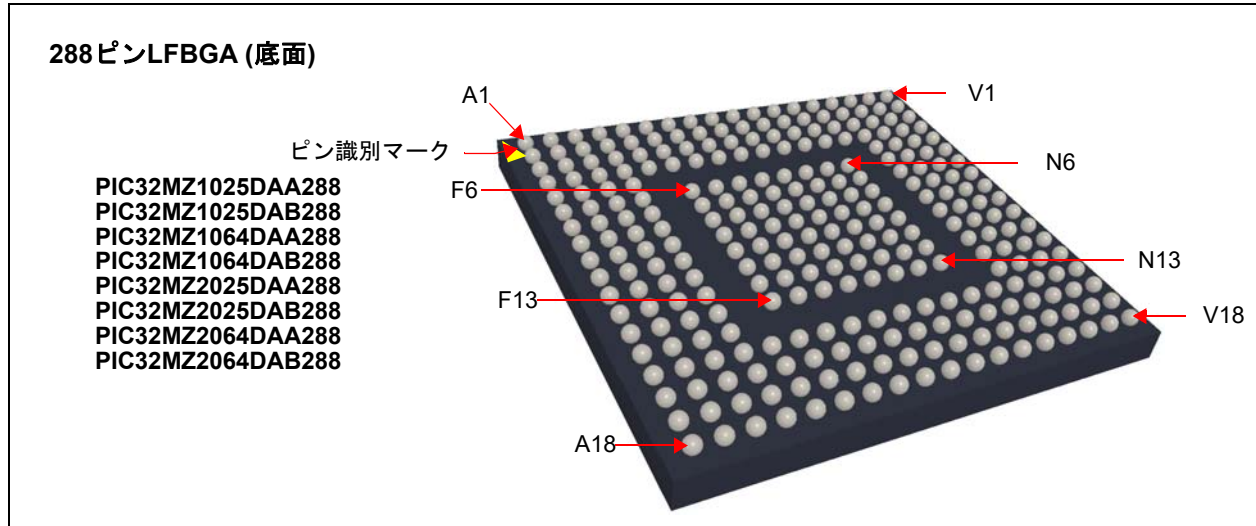
ピン番号	ピンのフルネーム	ピン番号	ピンのフルネーム
145	GD9/EBIBS0/RJ12	161	SOSCO/RPC14 <sup>(7)</sup> /T1CK/RC14 <sup>(7)</sup>
146	GD18/EBIBS1/RJ10	162	SOSCI/RPC13 <sup>(7)</sup> /RC13 <sup>(7)</sup>
147	GEN/EBICS3/RJ7	163	OSC2/CLKO/RC15
148	GCLK/EBICS2/RJ6	164	OSC1/CLKI/RC12
149	HSYNC/EBICS1/RJ5	165	VDDIO
150	VSYNC/EBICS0/RJ4	166	VBAT
151	GD20/EBIA22/RJ3	167	AN45/RPB5/RB5
152	EBIRDY3/AN32/RJ2	168	AN5/RPB10/RB10
153	Vss	169	PGED1/AN0/RPB0/CTED2/RB0
154	Vss	170	PGED2/C1INA/AN46/RPB7/RB7
155	VDDIO	171	AN6/RB12
156	VDDIO	172	AN1/C2INB/RPB2/RB2
157	AN33/SCK6/RD15	173	EBIA7/AN47/HLVDIN/RPB9/PMA7/RB9
158	AN22/RPD14/RD14	174	EBIA5/AN7/PMA5/RA5
159	AN29/SCK3/RB14	175	AN2/C1INB/RB4
160	TCK/AN24/RA1	176	未接続

- Note
- 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使えます。利用可能な周辺モジュールは表 1 と表 3 を参照してください。制約については 12.4 「ペリフェラル ピンセレクト (PPS)」 を参照してください。
  - 2: 全ての I/O ポートピン (RAX ~ RKx) は、状態変化通知ピン (CNAX ~ CNKx) として使えます。詳細は 12.0 「I/Oポート」 を参照してください。
  - 3: 網掛けしたピンは 5V 対応です。
  - 4: デバイス底面の金属パッドは内部で VSS1V8 に接続しており、1.8V グランドに外部で接続する必要があります。
  - 5: DDR 非実装デバイスの場合、このピンは 20kΩ 抵抗を介して Vss に接続する必要があります。
  - 6: DDR 非実装デバイスの場合、このピンは未接続 (NC) です。
  - 7: これらのピンは、入力機能のみに制限されます。



# PIC32MZ グラフィック (DA) ファミリ

表 7: 288 ピンデバイスのピン名

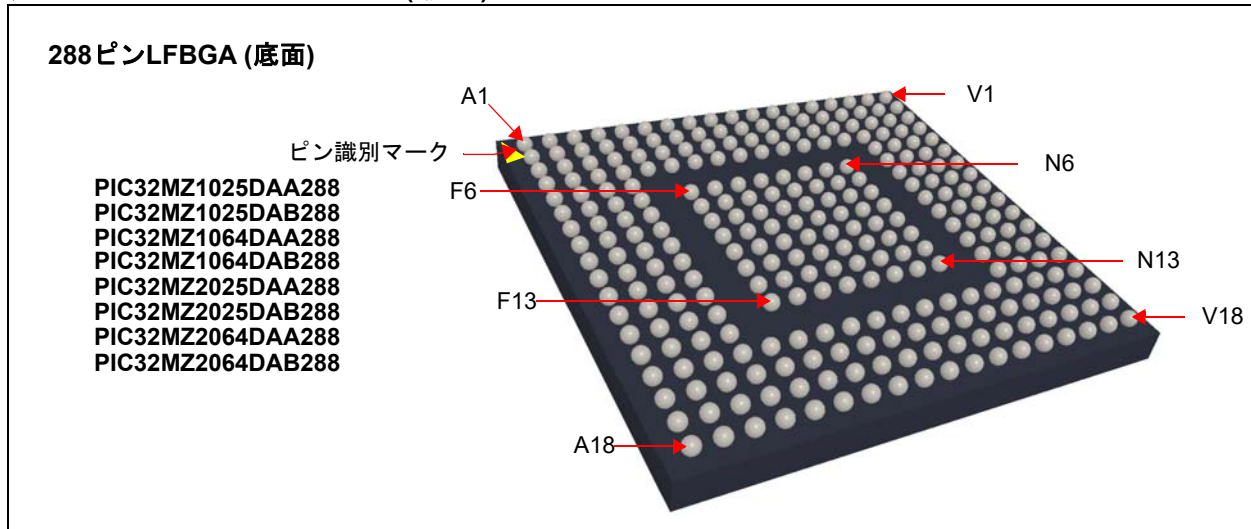


ボール/ ピン番号	ピンのフルネーム	ボール/ ピン番号	ピンのフルネーム
A1	未接続	B17	AN2/C1INB/RB4
A2	DDRUDQS	B18	EBIA5/AN7/PMA5/RA5
A3	DDRDM1	C1	DDRQ8
A4	D-	C2	DDRQ15
A5	Vss	C3	DDRQ9
A6	INT0/RH14	C4	VUSB3V3
A7	RPF2/SDA3/RF2	C5	VBUS
A8	AN21/RG15	C6	USBID
A9	AN14/C1IND/SCK2/RG6	C7	Vss
A10	TDI/AN17/SCK5/RF13	C8	未接続
A11	TDO/AN31/RPF12/RF12	C9	AN30/C2IND/RPG8/SCL4/RG8
A12	EBID5/AN12/RPC1/PMD5/RC1	C10	AN25/RPE8/RE8
A13	EBIOE/AN19/RPC4/PMRD/RC4	C11	EBID6/AN16/PMD6/RE6
A14	PGEC1/AN9/RPB1/CTED1/RB1	C12	未接続
A15	EBID10/AN4/RPB8/PMD10/RB8	C13	EBID12/AN10/RPC2/PMD12/RC2
A16	AN8/RPB3/RB3	C14	AN49/RB11
A17	VREF-/CVREF-/AN27/RA9	C15	VREF+/CVREF+/AN28/RA10
A18	未接続	C16	VDDIO
B1	未接続	C17	AN1/C2INB/RPB2/RB2
B2	DDRUDQS	C18	AN6/RB12
B3	DDRQ14	D1	DDRQ13
B4	D+	D2	DDRQ10
B5	Vss	D3	Vss1V8
B6	EBID4/AN18/PMD4/RE4	D4	TMS/SDC $\bar{D}$ /RA0
B7	EBID0/PMD0/RE0	D5	VUSB3V3
B8	AN20/RH4	D6	未接続
B9	EBIA2/AN23/C2INC/RPG9/PMA2/RG9	D7	VDDCORE
B10	AN26/RPE9/RE9	D8	EBID1/AN39/PMD1/RE1
B11	EBID7/AN15/PMD7/RE7	D9	AN13/C1INC/RPG7/SDA4/RG7
B12	未接続	D10	Vss
B13	EBIWE/AN34/RPC3/PMWR/RC3	D11	Vss
B14	PGEC2/RPB6/RB6	D12	Vss
B15	AN48/CTPLS/RB13	D13	Vss
B16	AN3/C2INA/RPB15/OCFB/RB15	D14	VDDCORE

- Note
- 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使用します。利用可能な周辺モジュールは表 1 と表 4 を参照してください。制約については 12.4 「ペリフェラル ピンセレクト (PPS)」を参照してください。
  - 2: 全ての I/O ポートピン (RAX ~ RKX) は、状態変化通知ピン (CNAX ~ CNKX) として使用します。詳細は 12.0 「I/O ポート」を参照してください。
  - 3: 網掛けしたピンは 5 V 対応です。
  - 4: DDR がシステム内で未接続の場合、このピンは 20k  $\Omega$  抵抗を介して Vss に接続する必要があります。
  - 5: システム内で DDR が未接続の場合、このピンは未接続 (NC) です。
  - 6: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

表 7: 288 ピンデバイスのピン名 ( 続き )

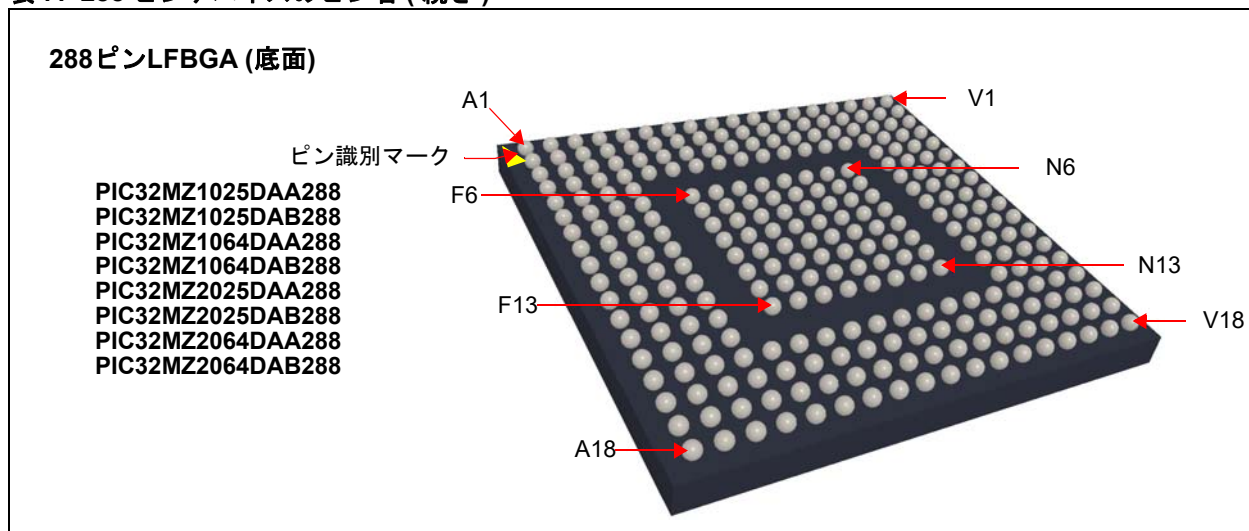


ボール / ピン番号	ピンのフルネーム	ボール / ピン番号	ピンのフルネーム
D15	VDDIO	G8	VSS1V8
D16	VDDIO	G9	VSS1V8
D17	PGED2/C1INA/AN46/RPB7/RB7	G10	VSS
D18	PGED1/AN0/RPB0/CTED2/RB0	G11	VDDIO
E1	DDRLDQS	G12	AVSS
E2	DDRLDQS	G13	AVDD
E3	DDRQ12	G15	VDDIO
E4	TRCLK/SDCK/SQICLK/RA6	G16	未接続
E15	VDDIO	G17	OSC1/CLKI/RC12
E16	EBIA7/AN47/HLVDIN/RPB9/PMA7/RB9	G18	OSC2/CLKO/RC15
E17	AN45/RPB5/RB5	H1	DDRQ2
E18	CVREFOUT/AN5/RPB10/RB10	H2	DDRQ5
F1	DDRQ0	H3	DDRQ6
F2	DDRQ7	H4	TRD0/SDDATA0/SQID0/RG13
F3	DDRQ11	H6	VDDR1V8 <sup>(4)</sup>
F4	TRD3/SDDATA3/SQID3/RA7	H7	VDDR1V8 <sup>(4)</sup>
F6	VSS1V8	H8	VDDR1V8 <sup>(4)</sup>
F7	VSS1V8	H9	VSS1V8
F8	VSS1V8	H10	VSS
F9	VSS	H11	VDDIO
F10	VSS	H12	VDDIO
F11	VDDIO	H13	VDDIO
F12	AVSS	H15	VDDIO
F13	AVDD	H16	TCK/AN24/RA1
F15	VDDIO	H17	SOSCI/RPC13 <sup>(6)</sup> /RC13 <sup>(6)</sup>
F16	VBAT	H18	SOSCO/RPC14 <sup>(6)</sup> /T1CK/RC14 <sup>(6)</sup>
F17	未接続	J1	DDRVREF <sup>(5)</sup>
F18	未接続	J2	未接続
G1	DDRQ3	J3	DDRQ1
G2	DDRQ4	J4	TRD2/SDDATA2/SQID2/RG14
G3	DDRDM0	J6	VDDR1V8 <sup>(4)</sup>
G4	TRD1/SDDATA1/SQID1/RG12	J7	VDDR1V8 <sup>(4)</sup>
G6	VSS1V8	J8	VDDR1V8 <sup>(4)</sup>
G7	VSS1V8	J9	VSS1V8

- Note
- 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使用します。利用可能な周辺モジュールは表 1 と表 4 を参照してください。制約については 12.4 「ペリフェラル ピンセレクト (PPS)」を参照してください。
  - 2: 全ての I/O ポートピン (RAX ~ RKX) は、状態変化通知ピン (CNAX ~ CNKX) として使えます。詳細は 12.0 「I/O ポート」を参照してください。
  - 3: 網掛けしたピンは 5V 対応です。
  - 4: DDR がシステム内で未接続の場合、このピンは 20k Ω 抵抗を介して Vss に接続する必要があります。
  - 5: システム内で DDR が未接続の場合、このピンは未接続 (NC) です。
  - 6: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

表 7: 288 ピンデバイスのピン名 ( 続き )

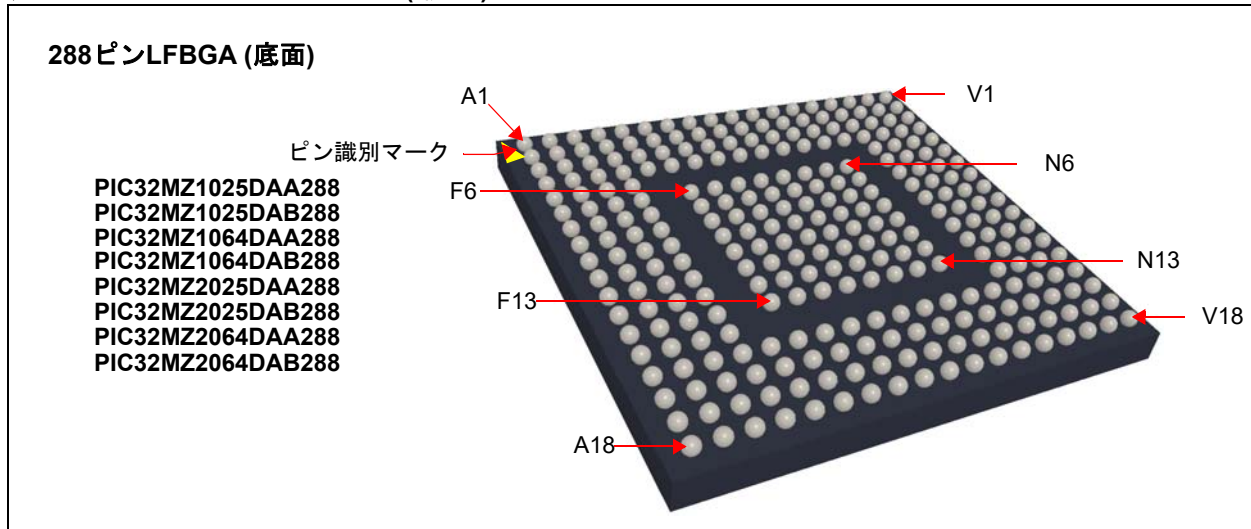


ボール / ピン番号	ピンのフルネーム	ボール / ピン番号	ピンのフルネーム
J10	VDDIO	L12	VDDIO
J11	Vss	L13	Vss
J12	Vss	L15	Vss
J13	Vss	L16	GEN/EBICS3/RJ7
J15	VDDIO	L17	GCLK/EBICS2/RJ6
J16	AN33/SCK6/RD15	L18	HSYNC/EBICS1/RJ5
J17	AN29/SCK3/RB14	M1	DDRRAS
J18	AN22/RPD14/RD14	M2	DDRBA0
K1	DDRCK	M3	DDRBA1
K2	DDRCK	M4	SCK1/RD1
K3	EBIA6/RPE5/PMA6/RE5	M6	Vss1V8
K4	SDCMD/SQICS0/RPD4/RD4	M7	Vss1V8
K6	VDDR1V8 <sup>(4)</sup>	M8	Vss1V8
K7	VDDR1V8 <sup>(4)</sup>	M9	Vss1V8
K8	VDDR1V8 <sup>(4)</sup>	M10	Vss
K9	Vss1V8	M11	Vss
K10	VDDIO	M12	VDDIO
K11	Vss	M13	VDDIO
K12	Vss	M15	VDDIO
K13	Vss	M16	GD0/EBID13/PMD13/RJ13
K15	Vss	M17	GD9/EBIBS0/RJ12
K16	EBIRDY3/AN32/RJ2	M18	GD18/EBIBS1/RJ10
K17	GD20/EBIA22/RJ3	N1	DDRODT
K18	VSYNC/EBICS0/RJ4	N2	DDRCS0
L1	DDRWE	N3	DDRA2
L2	DDRCKE	N4	GD22/EBIA13/PMA13/RD13
L3	DDRA1	N6	Vss1V8
L4	SQICS1/RPD5/RD5	N7	Vss1V8
L6	VDDR1V8 <sup>(4)</sup>	N8	Vss1V8
L7	VDDR1V8 <sup>(4)</sup>	N9	Vss1V8
L8	VDDR1V8 <sup>(4)</sup>	N10	Vss
L9	Vss1V8	N11	Vss
L10	Vss	N12	VDDIO
L11	VDDIO	N13	VDDIO

- Note 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使用します。利用可能な周辺モジュールは表 1 と表 4 を参照してください。制約については 12.4 「ペリフェラル ピンセレクト (PPS)」 を参照してください。
- 2: 全ての I/O ポートピン (RAX ~ RKX) は、状態変化通知ピン (CNAX ~ CNKX) として使用します。詳細は 12.0 「I/O ポート」 を参照してください。
- 3: 網掛けしたピンは 5 V 対応です。
- 4: DDR がシステム内で未接続の場合、このピンは 20k Ω 抵抗を介して Vss に接続する必要があります。
- 5: システム内で DDR が未接続の場合、このピンは未接続 (NC) です。
- 6: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

表 7: 288 ピンデバイスのピン名 ( 続き )

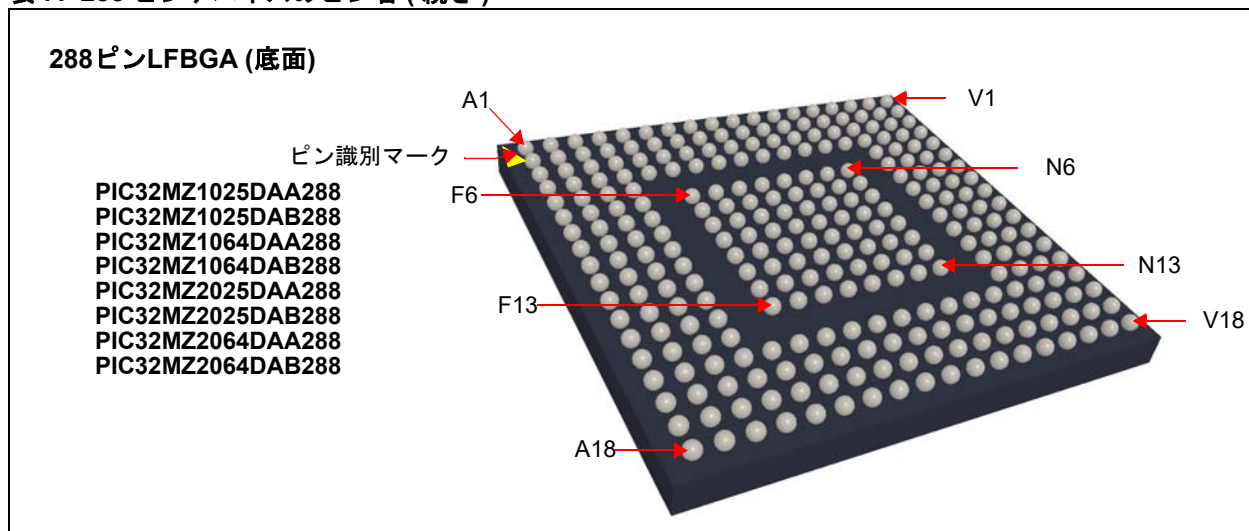


ボール / ピン番号	ピンのフルネーム	ボール / ピン番号	ピンのフルネーム
N15	EBIA4/AN36/PMA4/RH7	T5	未接続
N16	SDWP/EBIRP/RH2	T6	GD11/EBIA15/RPD3/PMA15/PMCS2/RD3
N17	EBIA0/PMA0/RJ15	T7	GD16/EBID8/RPF5/SCL5/PMD8/RF5
N18	GD8/EBID11/PMD11/RJ14	T8	GD4/EBIA9/RPG1/PMA9/RG1
P1	DDRA10	T9	EBID3/RPE3/PMD3/RE3
P2	DDRCAS	T10	ERXD2/RH6
P3	DDRA4	T11	ECOL/RH10
P4	RPF8/SCL3/RF8	T12	ETXD3/RH1
P15	GD13/EBIA18/RK4	T13	ETXD1/RJ9
P16	GD23/EBIA16/RK0	T14	未接続
P17	EBIRDY2/AN37/RH11	T15	ETXCLK/RPD7/RD7
P18	AN35/RH3	T16	RPA14/SCL1/RA14
R1	DDRA0	T17	GD19/EBIA21/RK7
R2	DDRA3	T18	GD15/EBIA20/RK6
R3	DDRA9	U1	DDRA6
R4	VSS1V8	U2	DDRA8
R5	MCLR	U3	DDRA13
R6	GD10/EBIA14/RPD2/PMA14/PMCS1/RD2	U4	DDRBA2
R7	Vss	U5	GD7/EBIA12/RPD12/PMA12/RD12
R8	Vss	U6	GD2/EBID15/RPD9/PMD15/RD9
R9	VDDIO	U7	GD5/EBIA10/RPF1/PMA10/RF1
R10	VDDIO	U8	ERXERR/RPF3/RF3
R11	VDDCORE	U9	GD17/EBID9/RPF4/SDA5/PMD9/RF4
R12	VDDIO	U10	ERXD1/RH5
R13	VDDIO	U11	ECRS/RH12
R14	VDDIO	U12	ERXD0/RH8
R15	GD14/EBIA19/RK5	U13	ERXCLK/EREFCLK/RJ11
R16	GD12/EBIA17/RK3	U14	EMDIO/RJ1
R17	EBIA3/AN11/PMA3/RK2	U15	EMDC/RPD11/RD11
R18	EBIA1/AN38/PMA1/RK1	U16	RPA15/SDA1/RA15
T1	DDRA5	U17	EBIRDY1/SDA2/RA3
T2	DDRA7	U18	SCL2/RA2
T3	DDRA12	V1	未接続
T4	DDRA14	V2	DDRA11

- Note 1: RPn ピンは、割り当て変更可能な周辺モジュール向けに使用します。利用可能な周辺モジュールは表 1 と表 4 を参照してください。制約については 12.4 「ペリフェラル ピンセレクト (PPS)」を参照してください。
- 2: 全ての I/O ポートピン (RAX ~ RKx) は、状態変化通知ピン (CNAX ~ CNKx) として使えます。詳細は 12.0 「I/O ポート」を参照してください。
- 3: 網掛けしたピンは 5V 対応です。
- 4: DDR がシステム内で未接続の場合、このピンは 20k Ω 抵抗を介して Vss に接続する必要があります。
- 5: システム内で DDR が未接続の場合、このピンは未接続 (NC) です。
- 6: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

表 7: 288 ピンデバイスのピン名 ( 続き )



ボール/ ピン番号	ピンのフルネーム	ボール/ ピン番号	ピンのフルネーム
V3	DDRA15	V11	ERXDV/ECRSDV/RH13
V4	VDDCORE	V12	ERXD3/RH9
V5	RTCC/RPD0/RD0	V13	ETXD2/RH0
V6	SCK4/RD10	V14	ETXD0/RJ8
V7	GD6/EBIA11/RPF0/PMA11/RF0	V15	ETXERR/RJ0
V8	GD21/EBIA23/RH15	V16	ETXEN/RPD6/RD6
V9	GD3/EBIA8/RPG0/PMA8/RG0	V17	GD1/EBID14/PMD14/RA4
V10	EBID2/PMD2/RE2	V18	未接続

- Note
- 1: RPN ピンは、割り当て変更可能な周辺モジュール向けに使用します。利用可能な周辺モジュールは表 1 と表 4 を参照してください。制約については 12.4 「ペリフェラルピンセレクト (PPS)」を参照してください。
  - 2: 全ての I/O ポートピン (RAX ~ RKX) は、状態変化通知ピン (CNAX ~ CNKX) として使用します。詳細は 12.0 「I/Oポート」を参照してください。
  - 3: 網掛けしたピンは 5V 対応です。
  - 4: DDR がシステム内で未接続の場合、このピンは 20k Ω 抵抗を介して Vss に接続する必要があります。
  - 5: システム内で DDR が未接続の場合、このピンは未接続 (NC) です。
  - 6: これらのピンは、入力機能のみに制限されます。

# PIC32MZ グラフィック (DA) ファミリ

---

## 目次

1.0	デバイス概要	17
2.0	32 ビット マイクロコントローラを使うためのガイドライン	39
3.0	CPU	45
4.0	メモリ構成	55
5.0	フラッシュ プログラムメモリ	105
6.0	リセット	115
7.0	CPU 例外と割り込みコントローラ	123
8.0	オシレータの設定	157
9.0	プリフェッチ モジュール	173
10.0	ダイレクト メモリアクセス (DMA) コントローラ	177
11.0	OTG (On-The-Go) 対応のハイスピード USB	201
12.0	I/O ポート	251
13.0	Timer1	279
14.0	Timer2/3、Timer4/5、Timer6/7、Timer8/9	283
15.0	入力キャプチャ	289
16.0	出力コンペア	293
17.0	デッドマウント タイマ (DMT)	299
18.0	ウォッチドッグ タイマ (WDT)	307
19.0	ディープスリープ ウォッチドッグ タイマ (DSWDT)	311
20.0	リアルタイム クロック / カレンダ (RTCC)	313
21.0	SPI (Serial Peripheral Interface) と I <sup>2</sup> S (Inter-IC Sound)	323
22.0	SQI (Serial Quad Interface)	333
23.0	I <sup>2</sup> C (Inter-Integrated Circuit)	361
24.0	UART (Universal Asynchronous Receiver Transmitter)	369
25.0	パラレル マスタポート (PMP)	377
26.0	外部バス インターフェイス (EBI)	391
27.0	暗号エンジン	399
28.0	乱数生成器 (RNG)	421
29.0	高速 12 ビット 逐次比較型 レジスタ (SAR) A/D コンバータ (ADC)	427
30.0	CAN (コントローラ エリア ネットワーク)	483
31.0	Ethernet コントローラ	521
32.0	コンパレータ	565
33.0	コンパレータ参照電圧 (CVREF)	569
34.0	HIGH/LOW 電圧検出 (HLVD)	573
35.0	充電時間計測ユニット (CTMU)	577
36.0	グラフィック LCD (GLCD) コントローラ	583
37.0	2D グラフィック処理ユニット (GPU)	603
38.0	DDR2 SDRAM コントローラ	605
39.0	セキュア デジタルホスト コントローラ (SDHC)	641
40.0	省電力機能	669
41.0	その他の特殊機能	683
42.0	命令セット	711
43.0	開発サポート	713
44.0	電気的特性	717
45.0	AC/DC 特性グラフ	777
46.0	パッケージ情報	785
索引		799

# PIC32MZ グラフィック (DA) ファミリ

## 大切なお客様へ

弊社は、大切なお客様が Microchip 社製品を最適にお使いになれるよう、文書の作成に最善の努力を尽くしています。このため、弊社はおお客様のニーズにより的確に応える事ができるよう、文書の改善を続けており、更新版をリリースする際に内容の見直しと充実を図って参ります。

本書に関してご質問またはご意見がございましたら、マーケティング コミュニケーション部宛てにメールでご連絡ください。メールの宛先は [docerrors@microchip.com](mailto:docerrors@microchip.com) です。お客様からのご意見とご感想をお待ちしております。

## 最新のデータシート

本書の最新版を入手するには、以下のウェブサイト登録手続きを行ってください。

<http://www.microchip.com>

データシートのバージョンは、各ページの欄外下隅に記載されている文書番号で確認できます。文書番号の末尾のアルファベットがリビジョンを表します (例: DS30000000A\_JP のリビジョンは「A」です)。

## エラッタ

現行のデバイスに対して、データシートとの動作上の微妙な相違点と推奨回避策を説明したエラッタシートを発行する場合があります。弊社はデバイスや文書に関する問題を認識した時点でエラッタを発行します。エラッタには該当するシリコンと文書のリビジョンを明記しています。

お使いのデバイス向けにエラッタシートが発行されているかどうかは以下で確認できます。

- Microchip 社のウェブサイト: <http://www.microchip.com>
- Microchip 社営業所 (本書の最後のページに記載)

お問い合わせの際は、お使いのデバイス、シリコンとデータシートのリビジョン (文書番号を含む) をお知らせください。

## 顧客変更通知システム

弊社ウェブサイト ([www.microchip.com](http://www.microchip.com)) で登録になったお客様には、弊社の全製品に関する最新情報をお届けします。

# PIC32MZ グラフィック (DA) ファミリ

---

## 参考資料

本書は『PIC32 ファミリ リファレンス マニュアル』の各セクションに基づきます。それらの文書は、特定のモジュールの動作またはデバイス機能に関する全般的リファレンスとしてお使いください。

Note: 以下の文書 (英語版) は、Microchip 社ウェブサイト (<http://www.microchip.com>) の [DESIGN SUPPORT] から入手できます。一部の文書の日本語版はマイクロチップ・テクノロジー・ジャパン株式会社ウェブサイト (<http://www.microchip.co.jp/>) の [日本語資料] から入手できます。

- Section 1. "Introduction" (DS60001127)
- Section 7. "Resets" (DS60001118)
- Section 8. "Interrupt Controller" (DS60001108)
- Section 9. "Watchdog, Deadman, and Power-up Timers" (DS60001114)
- Section 10. "Power-Saving Features" (DS60001130)
- Section 12. "I/O Ports" (DS60001120)
- Section 13. "Parallel Master Port (PMP)" (DS60001128)
- Section 14. "Timers" (DS60001105)
- Section 15. "Input Capture" (DS60001122)
- Section 16. "Output Compare" (DS60001111)
- Section 19. "Comparator" (DS60001110)
- Section 20. "Comparator Voltage Reference (CVREF)" (DS60001109)
- Section 21. "Universal Asynchronous Receiver Transmitter (UART)" (DS60001107)
- Section 22. "12-bit High-Speed Successive Approximation Register (SAR) Analog-to-Digital Converter (ADC)" (DS60001344)
- Section 23. "Serial Peripheral Interface (SPI)" (DS60001106)
- Section 24. "Inter-Integrated Circuit (I2C)" (DS60001116)
- Section 29. "Real-Time Clock and Calendar (RTCC)" (DS60001125)
- Section 31. "Direct Memory Access (DMA) Controller" (DS60001117)
- Section 32. "Configuration" (DS60001124)
- Section 33. "Programming and Diagnostics" (DS60001129)
- Section 34. "Controller Area Network (CAN)" (DS60001154)
- Section 35. "Ethernet Controller" (DS60001155)
- Section 37. "Charge Time Measurement Unit (CTMU)" (DS60001167)
- Section 38. "High/Low Voltage Detect (HLVD)" (DS60001408)
- Section 41. "Prefetch Module for Devices with L1 CPU Cache" (DS60001183)
- Section 42. "Oscillators with Enhanced PLL" (DS60001250)
- Section 46. "Serial Quad Interface (SQI)" (DS60001244)
- Section 47. "External Bus Interface (EBI)" (DS60001245)
- Section 48. "Memory Organization and Permissions" (DS60001214)
- Section 49. "Crypto Engine (CE) and Random Number Generator (RNG)" (DS60001246)
- Section 50. "CPU for Devices with MIPS32® microAptiv™ and M-Class Cores" (DS60001192)
- Section 51. "High-Speed USB with On-The-Go (OTG)" (DS60001326)
- Section 52. "Flash Program Memory with Support for Live Update" (DS60001193)
- Section 54. Graphics LCD (GLCD) Controller (DS60001379)
- Section 55. "DDR SDRAM Controller" (DS60001321)
- Section 57. "Secure Digital Host Controller (SDHC)" (DS60001334)







# PIC32MZ グラフィック (DA) ファミリ

表 1-2: オシレータの I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>オシレータ</b>						
CLKI	E12	164	G17	I	ST/CMOS	外部クロック源入力 常に OSC1 ピン機能と共有されます。
CLKO	E13	163	G18	O		オシレータ水晶振動子出力 水晶振動子オシレータモードでは、このピンに水晶振動子またはレゾネータを接続します。 オプションにより、RC モードと EC モードでは CLKO として機能します。 常に OSC2 ピン機能と共有されます。
OSC1	E12	164	G17	I	ST/CMOS	オシレータ水晶振動子入力 RC モードに設定されている場合は ST バッファ、その他の場合は CMOS です。
OSC2	E13	163	G18	O		オシレータ水晶振動子出力 水晶振動子オシレータモードでは、このピンに水晶振動子またはレゾネータを接続します。 オプションにより、RC モードと EC モードでは CLKO として機能します。
SOSCI	C13	162	H17	I	ST/CMOS	32.768 kHz 低消費電力オシレータ水晶振動子入力 その他の場合は CMOS
SOSCO	D13	161	H18	O	ST/CMOS	32.768 kHz 低消費電力オシレータ水晶振動子出力
REFCLKI1	PPS	PPS	PPS	I	—	参照クロック ジェネレータ入力 1 ~ 4
REFCLKI3	PPS	PPS	PPS	I	—	
REFCLKI4	PPS	PPS	PPS	I	—	
REFCLKO1	PPS	PPS	PPS	O	—	
REFCLKO3	PPS	PPS	PPS	O	—	参照クロック ジェネレータ出力 1 ~ 4
REFCLKO4	PPS	PPS	PPS	O	—	

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

表 1-3: IC1 ~ IC9 の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>入力キャプチャ</b>						
IC1	PPS	PPS	PPS	I	ST	入力キャプチャ入力 1 ~ 9
IC2	PPS	PPS	PPS	I	ST	
IC3	PPS	PPS	PPS	I	ST	
IC4	PPS	PPS	PPS	I	ST	
IC5	PPS	PPS	PPS	I	ST	
IC6	PPS	PPS	PPS	I	ST	
IC7	PPS	PPS	PPS	I	ST	
IC8	PPS	PPS	PPS	I	ST	
IC9	PPS	PPS	PPS	I	ST	

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-4: OC1 ~ OC9 の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>出力コンペア</b>						
OC1	PPS	PPS	PPS	O	—	出力コンペア出力 1 ~ 9
OC2	PPS	PPS	PPS	O	—	
OC3	PPS	PPS	PPS	O	—	
OC4	PPS	PPS	PPS	O	—	
OC5	PPS	PPS	PPS	O	—	
OC6	PPS	PPS	PPS	O	—	
OC7	PPS	PPS	PPS	O	—	
OC8	PPS	PPS	PPS	O	—	
OC9	PPS	PPS	PPS	O	—	
OCFA	PPS	PPS	PPS	I	ST	出力コンペアフォルト A 入力
OCFB	PPS	PPS	PPS	I	ST	出力コンペアフォルト B 入力

凡例 : CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

表 1-5: 外部割り込みの I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>外部割り込み</b>						
INT0	C3	42	A6	I	ST	外部割り込み 0
INT1	PPS	PPS	PPS	I	ST	外部割り込み 1
INT2	PPS	PPS	PPS	I	ST	外部割り込み 2
INT3	PPS	PPS	PPS	I	ST	外部割り込み 3
INT4	PPS	PPS	PPS	I	ST	外部割り込み 4

凡例 : CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-6: PORTA ~ PORTK の I/O ピン機能

ピン名	ピン番号			ピン タイプ	バッファ タイプ	概要	
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA				
<b>PORTA</b>							
RA0	D2	53	D4	I/O	ST	PORTA は双方向 I/O ポートです。	
RA1	E11	160	H16	I/O	ST		
RA2	L12	129	U18	I/O	ST		
RA3	N12	128	U17	I/O	ST		
RA4	L11	127	V17	I/O	ST		
RA5	A11	174	B18	I/O	ST		
RA6	E4	54	E4	I/O	ST		
RA7	D1	55	F4	I/O	ST		
RA9	B11	1	A17	I/O	ST		
RA10	C10	2	C15	I/O	ST		
RA14	M11	126	T16	I/O	ST		
RA15	N11	125	U16	I/O	ST		
<b>PORTB</b>							
RB0	C12	169	D18	I/O	ST		PORTB は双方向 I/O ポートです。
RB1	B9	11	A14	I/O	ST		
RB2	A13	172	C17	I/O	ST		
RB3	A10	8	A16	I/O	ST		
RB4	A12	175	B17	I/O	ST		
RB5	D11	167	E17	I/O	ST		
RB6	D7	13	B14	I/O	ST		
RB7	D12	170	D17	I/O	ST		
RB8	A9	10	A15	I/O	ST		
RB9	B12	173	E16	I/O	ST		
RB10	C11	168	E18	I/O	ST		
RB11	E7	12	C14	I/O	ST		
RB12	B13	171	C18	I/O	ST		
RB13	F7	9	B15	I/O	ST		
RB14	E10	175	J17	I/O	ST		
RB15	B10	7	B16	I/O	ST		
<b>PORTC</b>							
RC1	B7	17	A12	I/O	ST	PORTC は双方向 I/O ポートです。	
RC2	A8	14	C13	I/O	ST		
RC3	A7	15	B13	I/O	ST		
RC4	B8	16	A13	I/O	ST		
RC12	E12	164	G17	I/O	ST		
RC13	C13	162	H17	I	ST		
RC14	D13	161	H18	I	ST		
RC15	E13	163	G18	I/O	ST		

**凡例:** CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-6: PORTA ~ PORTK の I/O ピン機能 ( 続き )

ピン名	ピン番号			ピン タイプ	バッファ タイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>PORTD</b>						
RD0	H3	79	V5	I/O	ST	PORTD は双方向 I/O ポートです。
RD1	G1	73	M4	I/O	ST	
RD2	G2	74	R6	I/O	ST	
RD3	G3	75	T6	I/O	ST	
RD4	F1	70	K4	I/O	ST	
RD5	F2	71	L4	I/O	ST	
RD6	K11	121	V16	I/O	ST	
RD7	M10	120	T15	I/O	ST	
RD9	H1	76	U6	I/O	ST	
RD10	H2	77	V6	I/O	ST	
RD11	N10	119	U15	I/O	ST	
RD12	J1	80	U5	I/O	ST	
RD13	J2	81	N4	I/O	ST	
RD14	E9	158	J18	I/O	ST	
RD15	F11	157	J16	I/O	ST	
<b>PORTE</b>						
RE0	C4	40	B7	I/O	ST	PORTE は双方向 I/O ポートです。
RE1	A4	36	D8	I/O	ST	
RE2	N3	99	V10	I/O	ST	
RE3	M3	98	T9	I/O	ST	
RE4	B3	43	B6	I/O	ST	
RE5	F3	17	K3	I/O	ST	
RE6	F6	23	C11	I/O	ST	
RE7	C7	24	B11	I/O	ST	
RE8	E6	25	C10	I/O	ST	
RE9	D6	26	B10	I/O	ST	
<b>PORTF</b>						
RF0	L1	91	V7	I/O	ST	PORTF は双方向 I/O ポートです。
RF1	K3	90	U7	I/O	ST	
RF2	A3	41	A7	I/O	ST	
RF3	M1	93	U8	I/O	ST	
RF4	L3	44	U9	I/O	ST	
RF5	K2	89	T7	I/O	ST	
RF8	J3	82	P4	I/O	ST	
RF12	C6	27	A11	I/O	ST	
RF13	A6	28	A10	I/O	ST	

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト



# PIC32MZ グラフィック (DA) ファミリ

表 1-6: PORTA ~ PORTK の I/O ピン機能 ( 続き )

ピン名	ピン番号			ピン タイプ	バッファ タイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>PORTK</b>						
RK0	K12	137	P16	I/O	ST	PORTK は双方向 I/O ポートです。
RK1	J11	136	R18	I/O	ST	
RK2	H11	135	R17	I/O	ST	
RK3	L13	134	R16	I/O	ST	
RK4	H10	133	P15	I/O	ST	
RK5	J10	132	R15	I/O	ST	
RK6	M13	131	T18	I/O	ST	
RK7	M12	130	T17	I/O	ST	

凡例 : CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

表 1-7: Timer1 ~ Timer9 および RTCC の I/O ピン機能

ピン名	ピン番号			ピン タイプ	バッファ タイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>Timer1 ~ Timer9</b>						
T1CK	D13	161	H18	I	ST	Timer1 外部クロック入力
T2CK	PPS	PPS	PPS	I	ST	Timer2 外部クロック入力
T3CK	PPS	PPS	PPS	I	ST	Timer3 外部クロック入力
T4CK	PPS	PPS	PPS	I	ST	Timer4 外部クロック入力
T5CK	PPS	PPS	PPS	I	ST	Timer5 外部クロック入力
T6CK	PPS	PPS	PPS	I	ST	Timer6 外部クロック入力
T7CK	PPS	PPS	PPS	I	ST	Timer7 外部クロック入力
T8CK	PPS	PPS	PPS	I	ST	Timer8 外部クロック入力
T9CK	PPS	PPS	PPS	I	ST	Timer9 外部クロック入力
<b>リアルタイムクロック / カレンダー</b>						
RTCC <sup>(1)</sup>	H3	79	V5	O	—	リアルタイムクロックアラーム / 秒パルス出力

凡例 : CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

Note 1: VBAT 動作中は、RTCC ピン機能は使えません。



# PIC32MZ グラフィック (DA) ファミリ

表 1-8: UART1 ~ UART6 の I/O ピン機能

ピン名	ピン番号			ピン名 タイプ	バッファ タイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>UART1</b>						
U1RX	PPS	PPS	PPS	I	ST	UART1 受信
U1TX	PPS	PPS	PPS	O	—	UART1 送信
U1CTS	PPS	PPS	PPS	I	ST	UART1 の CTS (Clear to Send)
U1RTS	PPS	PPS	PPS	O	—	UART1 の RTS (Ready to Send)
<b>UART2</b>						
U2RX	PPS	PPS	PPS	I	ST	UART2 受信
U2TX	PPS	PPS	PPS	O	—	UART2 送信
U2CTS	PPS	PPS	PPS	I	ST	UART2 の CTS (Clear to Send)
U2RTS	PPS	PPS	PPS	O	—	UART2 の RTS (Ready to Send)
<b>UART3</b>						
U3RX	PPS	PPS	PPS	I	ST	UART3 受信
U3TX	PPS	PPS	PPS	O	—	UART3 送信
U3CTS	PPS	PPS	PPS	I	ST	UART3 の CTS (Clear to Send)
U3RTS	PPS	PPS	PPS	O	—	UART3 の RTS (Ready to Send)
<b>UART4</b>						
U4RX	PPS	PPS	PPS	I	ST	UART4 受信
U4TX	PPS	PPS	PPS	O	—	UART4 送信
U4CTS	PPS	PPS	PPS	I	ST	UART4 の CTS (Clear to Send)
U4RTS	PPS	PPS	PPS	O	—	UART4 の RTS (Ready to Send)
<b>UART5</b>						
U5RX	PPS	PPS	PPS	I	ST	UART5 受信
U5TX	PPS	PPS	PPS	O	—	UART5 送信
U5CTS	PPS	PPS	PPS	I	ST	UART5 の CTS (Clear to Send)
U5RTS	PPS	PPS	PPS	O	—	UART5 の RTS (Ready to Send)
<b>UART6</b>						
U6RX	PPS	PPS	PPS	I	ST	UART6 受信
U6TX	PPS	PPS	PPS	O	—	UART6 送信
U6CTS	PPS	PPS	PPS	I	ST	UART6 の CTS (Clear to Send)
U6RTS	PPS	PPS	PPS	O	—	UART6 の RTS (Ready to Send)

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-9: SPI1 ~ SPI6 の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>シリアルペリフェラルインターフェイス 1</b>						
SCK1	G1	73	M4	I/O	ST	SPI1 同期シリアルクロック入出力
SDI1	PPS	PPS	PPS	I	ST	SPI1 データ入力
SDO1	PPS	PPS	PPS	O	—	SPI1 データ出力
SS1	PPS	PPS	PPS	I/O	ST	SPI1 スレーブ同期またはフレームパルス I/O
<b>シリアルペリフェラルインターフェイス 2</b>						
SCK2	E5	30	A9	I/O	ST	SPI2 同期シリアルクロック入出力
SDI2	PPS	PPS	PPS	I	ST	SPI2 データ入力
SDO2	PPS	PPS	PPS	O	—	SPI2 データ出力
SS2	PPS	PPS	PPS	I/O	ST	SPI2 スレーブ同期またはフレームパルス I/O
<b>シリアルペリフェラルインターフェイス 3</b>						
SCK3	E10	159	J17	I/O	ST	SPI3 同期シリアルクロック入出力
SDI3	PPS	PPS	PPS	I	ST	SPI3 データ入力
SDO3	PPS	PPS	PPS	O	—	SPI3 データ出力
SS3	PPS	PPS	PPS	I/O	ST	SPI3 スレーブ同期またはフレームパルス I/O
<b>シリアルペリフェラルインターフェイス 4</b>						
SCK4	H2	77	V6	I/O	ST	SPI4 同期シリアルクロック入出力
SDI4	PPS	PPS	PPS	I	ST	SPI4 データ入力
SDO4	PPS	PPS	PPS	O	—	SPI4 データ出力
SS4	PPS	PPS	PPS	I/O	ST	SPI4 スレーブ同期またはフレームパルス I/O
<b>シリアルペリフェラルインターフェイス 5</b>						
SCK5	A6	28	A10	I/O	ST	SPI5 同期シリアルクロック入出力
SDI5	PPS	PPS	PPS	I	ST	SPI5 データ入力
SDO5	PPS	PPS	PPS	O	—	SPI5 データ出力
SS5	PPS	PPS	PPS	I/O	ST	SPI5 スレーブ同期またはフレームパルス I/O
<b>シリアルペリフェラルインターフェイス 6</b>						
SCK6	F11	157	J16	I/O	ST	SPI6 同期シリアルクロック入出力
SDI6	PPS	PPS	PPS	I	ST	SPI6 データ入力
SDO6	PPS	PPS	PPS	O	—	SPI6 データ出力
SS6	PPS	PPS	PPS	I/O	ST	SPI6 スレーブ同期またはフレームパルス I/O

凡例 : CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-10: I2C1 ~ I2C5 の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>I<sup>2</sup>C1</b>						
SCL1	M11	126	T16	I/O	ST	I2C1 同期シリアルクロック入出力
SDA1	N11	125	U16	I/O	ST	I2C1 同期シリアルデータ入出力
<b>I<sup>2</sup>C2</b>						
SCL2	L12	129	U18	I/O	ST	I2C2 同期シリアルクロック入出力
SDA2	N12	128	U17	I/O	ST	I2C2 同期シリアルデータ入出力
<b>I<sup>2</sup>C3</b>						
SCL3	J3	82	P4	I/O	ST	I2C3 同期シリアルクロック入出力
SDA3	A3	42	A7	I/O	ST	I2C3 同期シリアルデータ入出力
<b>I<sup>2</sup>C4</b>						
SCL4	B5	32	C9	I/O	ST	I2C4 同期シリアルクロック入出力
SDA4	D5	31	D9	I/O	ST	I2C4 同期シリアルデータ入出力
<b>I<sup>2</sup>C5</b>						
SCL5	K2	89	T7	I/O	ST	I2C5 同期シリアルクロック入出力
SDA5	L3	97	U9	I/O	ST	I2C5 同期シリアルデータ入出力

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

表 1-11: コンパレータ 1、コンパレータ 2、CVREF の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>コンパレータ参照電圧</b>						
CVREF+	C10	2	C15	I	アナログ	コンパレータ参照電圧 (HIGH) 入力
CVREF-	B11	1	A17	I	アナログ	コンパレータ参照電圧 (LOW) 入力
CVREFOUT	C11	168	E18	O	アナログ	コンパレータ参照電圧出力
<b>コンパレータ 1</b>						
C1INA	D12	170	D17	I	アナログ	コンパレータ 1 非反転入力
C1INB	A12	176	B17	I	アナログ	
C1INC	D5	31	D9	I	アナログ	
C1IND	E5	30	A9	I	アナログ	
C1OUT	PPS	PPS	PPS	O	—	コンパレータ 1 出力
<b>コンパレータ 2</b>						
C2INA	B10	7	B16	I	アナログ	コンパレータ 2 非反転入力
C2INB	A13	172	C17	I	アナログ	
C2INC	C5	33	B9	I	アナログ	
C2IND	B5	32	C9	I	アナログ	
C2OUT	PPS	PPS	PPS	O	—	コンパレータ 2 出力

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-12: PMP の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>パラレル マスタポート</b>						
PMA0	H13	142	N17	I/O	TTL/ST	パラレル マスタポート アドレス ビット 0 入力 (バッファ付きスレーブモード) / 出力 (マスタモード)
PMA1	J11	136	R18	I/O	TTL/ST	パラレル マスタポート アドレス ビット 1 入力 (バッファ付きスレーブモード) / 出力 (マスタモード)
PMA2	C5	33	B9	O	—	パラレル マスタポート アドレス (非多重化マスタモード)
PMA3	H11	135	R17	O	—	
PMA4	J12	139	N15	O	—	
PMA5	A11	174	B18	O	—	
PMA6	F3	69	K3	O	—	
PMA7	B12	173	E16	O	—	
PMA8	N2	96	V9	O	—	
PMA9	M2	95	T8	O	—	
PMA10	K3	90	U7	O	—	
PMA11	L1	91	V7	O	—	
PMA12	J1	80	U5	O	—	
PMA13	J2	81	N4	O	—	
PMA14	G2	74	R6	O	—	
PMA15	G3	75	T6	O	—	
PMCS1	G2	74	R6	O	—	
PMCS2	G3	75	T6	O	—	パラレル マスタポート チップセレクト 2 ストローブ
PMD0	C4	40	B7	I/O	TTL/ST	パラレル マスタポート データ (非多重化マスタモード) またはアドレス / データ (多重化マスタモード)
PMD1	A4	36	D8	I/O	TTL/ST	
PMD2	N3	99	V10	I/O	TTL/ST	
PMD3	M3	98	T9	I/O	TTL/ST	
PMD4	B3	43	B6	I/O	TTL/ST	
PMD5	B7	17	A12	I/O	TTL/ST	
PMD6	F6	23	C11	I/O	TTL/ST	
PMD7	C7	24	B11	I/O	TTL/ST	
PMD8	K2	89	T7	I/O	TTL/ST	
PMD9	L3	97	U9	I/O	TTL/ST	
PMD10	A9	10	A15	I/O	TTL/ST	
PMD11	G10	143	N18	I/O	TTL/ST	
PMD12	A8	14	C13	I/O	TTL/ST	
PMD13	G12	144	M16	I/O	TTL/ST	
PMD14	L11	127	V17	I/O	TTL/ST	
PMD15	H1	76	U6	I/O	TTL/ST	
PMALL	H13	142	N17	O	—	パラレル マスタポート アドレスラッチ イネーブル下位バイト (多重化マスタモード)
PMALH	J11	136	R18	O	—	パラレル マスタポート アドレスラッチ イネーブル上位バイト (多重化マスタモード)
PMRD	B8	16	A13	O	—	パラレル マスタポート 読み出し ストローブ
PMWR	A7	15	B13	O	—	パラレル マスタポート 書き込み ストローブ

**凡例:** CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-13: EBI の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>外部バス インターフェイス</b>						
EBIA0	H13	142	N17	O	—	外部バス インターフェイス アドレスバス
EBIA1	J11	136	R18	O	—	
EBIA2	C5	33	B9	O	—	
EBIA3	H11	135	R17	O	—	
EBIA4	J12	139	N15	O	—	
EBIA5	A11	174	B18	O	—	
EBIA6	F3	69	K3	O	—	
EBIA7	B12	173	E16	O	—	
EBIA8	N2	96	V9	O	—	
EBIA9	M2	95	T8	O	—	
EBIA10	K3	90	U7	O	—	
EBIA11	L1	91	V7	O	—	
EBIA12	J1	80	U5	O	—	
EBIA13	J2	81	N4	O	—	
EBIA14	G2	74	R6	O	—	
EBIA15	G3	75	T6	O	—	
EBIA16	K12	137	P16	O	—	
EBIA17	L13	134	R16	O	—	
EBIA18	H10	133	P15	O	—	
EBIA19	J10	132	R15	O	—	
EBIA20	M13	131	T18	O	—	
EBIA21	M12	130	T17	O	—	
EBIA22	E8	151	K17	O	—	
EBIA23	L2	92	V8	O	—	
EBID0	C4	40	B7	I/O	ST	外部バス インターフェイス データ I/O バス
EBID1	A4	40	D8	I/O	ST	
EBID2	N3	36	V10	I/O	ST	
EBID3	M3	99	T9	I/O	ST	
EBID4	B3	98	B6	I/O	ST	
EBID5	B7	43	A12	I/O	ST	
EBID6	F6	17	C11	I/O	ST	
EBID7	C7	23	B11	I/O	ST	
EBID8	K2	24	T7	I/O	ST	
EBID9	L3	89	U9	I/O	ST	
EBID10	A9	97	A15	I/O	ST	
EBID11	G10	10	N18	I/O	ST	
EBID12	A8	143	C13	I/O	ST	
EBID13	G12	14	M16	I/O	ST	
EBID14	L11	144	V17	I/O	ST	
EBID15	H1	127	U6	I/O	ST	

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-13: EBI の I/O ピン機能 ( 続き )

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
EBIBS0	J11	145	M17	O	—	外部バス インターフェイス バイト選択
EBIBS1	J12	146	M18	O	—	
EBICS0	G10	150	K18	O	—	外部バス インターフェイス チップセレクト
EBICS1	H12	149	L18	O	—	
EBICS2	H11	148	L17	O	—	
EBICS3	H10	147	L16	O	—	
EBIOE	E12	16	A13	O	—	外部バス インターフェイス出力イネーブル
EBIRDY1	M10	128	U17	I	ST	外部バス インターフェイス レディー入力
EBIRDY2	C5	138	P17	I	ST	
EBIRDY3	C4	152	K16	I	ST	
EBIRP	F1	141	N16	O	—	外部バス インターフェイス フラッシュ リセットピン
EBIWE	D11	15	B13	O	—	外部バス インターフェイス書き込みイネーブル

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

表 1-14: USB の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>USB</b>						
VBUS	A2	45	C5	I	アナログ	USB バスパワーモニタ
VUSB3V3	B2	46、47	C4, D5	P	—	USB 内部トランシーバ電源 USB モジュールを使わない場合、このピンは Vss に接続する必要があります。
D+	C1	51	B4	I/O	アナログ	USB D+
D-	B1	50	A4	I/O	アナログ	USB D-
USBID	D3	52	C6	I	ST	USB OTG ID 検出

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

表 1-15: CAN1、CAN2 の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>コントローラ エリア ネットワーク</b>						
C1TX	PPS	PPS	PPS	O	—	CAN1 バス送信ピン
C1RX	PPS	PPS	PPS	I	ST	CAN1 バス受信ピン
C2TX	PPS	PPS	PPS	O	—	CAN2 バス送信ピン
C2RX	PPS	PPS	PPS	I	ST	CAN2 バス受信ピン

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-16: Ethernet MII の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>Ethernet</b>						
ERXD0	N7	108	U12	I	ST	Ethernet 受信データ 0
ERXD1	M4	100	U10	I	ST	Ethernet 受信データ 1
ERXD2	N4	101	T10	I	ST	Ethernet 受信データ 2
ERXD3	N6	107	V12	I	ST	Ethernet 受信データ 3
ERXERR	M1	93	U8	I	ST	Ethernet 受信エラー入力
ERXDV	M5	104	V11	I	ST	Ethernet 受信データ有効
ERXCLK	N8	111	U13	I	ST	Ethernet 受信クロック
ETXD0	N9	113	V14	O	—	Ethernet 送信データ 0
ETXD1	M9	112	T13	O	—	Ethernet 送信データ 1
ETXD2	M8	110	V13	O	—	Ethernet 送信データ 2
ETXD3	M7	109	T12	O	—	Ethernet 送信データ 3
ETXERR	L10	118	V15	O	—	Ethernet 送信エラー
ETXEN	K11	121	V16	O	—	Ethernet 送信イネーブル
ETXCLK	M10	120	T15	I	ST	Ethernet 送信クロック
ECOL	M6	106	T11	I	ST	Ethernet コリジョン検出
ECRS	N5	105	U11	I	ST	Ethernet キャリア検出
EMDC	N10	119	U15	O	—	Ethernet マネジメント データ クロック
EMDIO	K10	114	U14	I/O	—	Ethernet マネジメント データ

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

表 1-17: Ethernet RMII の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>Ethernet MII インターフェイス</b>						
ERXD0	N7	108	U12	I	ST	Ethernet 受信データ 0
ERXD1	M4	100	U10	I	ST	Ethernet 受信データ 1
ERXERR	M1	93	U8	I	ST	Ethernet 受信エラー入力
ETXD0	N9	113	V14	O	—	Ethernet 送信データ 0
ETXD1	M9	112	T13	O	—	Ethernet 送信データ 1
ETXEN	K11	121	V16	O	—	Ethernet 送信イネーブル
EMDC	N10	119	U15	O	—	Ethernet マネジメント データ クロック
EMDIO	K10	114	U14	I/O	—	Ethernet マネジメント データ
EREFCLK	N8	111	U13	I	ST	Ethernet 参照クロック
ECRSDV	M5	104	V11	I	ST	Ethernet キャリア検出データ有効

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-18: SQI1 の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>SQI (Serial Quad Interface)</b>						
SQICLK	E4	54	E4	O	—	SQI クロック
SQICS0	F1	70	K4	O	—	SQI チップセレクト 0
SQICS1	F2	71	L4	O	—	SQI チップセレクト 1
SQID0	E2	64	H4	I/O	ST	SQI データ 0
SQID1	E3	56	G4	I/O	ST	SQI データ 1
SQID2	E1	65	J4	I/O	ST	SQI データ 2
SQID3	D1	55	F4	I/O	ST	SQI データ 3

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

表 1-19: SDHC の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>SDHC</b>						
SDCK	E4	54	E4	O	—	SD シリアルクロック
SDCMD	F1	70	K4	O	—	SD コマンド / 応答
SDDATA0	E2	64	H4	I/O	ST	SD シリアルデータ 0
SDDATA1	E3	56	G4	I/O	ST	SD シリアルデータ 1
SDDATA2	E1	65	J4	I/O	ST	SD シリアルデータ 2
SDDATA3	D1	55	F4	I/O	ST	SD シリアルデータ 3/ カード検出
SDCD	D2	53	D4	I	ST	SD 機械式カード検出
SDWP	H12	141	N16	I	ST	SD 書き込み保護

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

表 1-20: CTMU の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>充電時間計測ユニット (CTMU)</b>						
CTED1	B9	11	A14	I	ST	CTMU 外部エッジ入力 1
CTED2	C12	169	D18	I	ST	CTMU 外部エッジ入力 2
CTPLS	F7	9	B15	O	—	CTMU 出力パルス

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト



# PIC32MZ グラフィック (DA) ファミリ

表 1-21: グラフィック LCD (GLCD) コントローラの I/O ピン機能

ピン名	ピン番号			ピン タイプ	バッファ タイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>GLCD コントローラ</b>						
GCLK	G11	148	L17	O	—	グラフィック ディスプレイ画素クロック
HSYNC	F12	149	L18	O	—	グラフィック ディスプレイ水平同期パルス
VSYNC	F13	150	K18	O	—	グラフィック ディスプレイ垂直同期パルス
GEN	G13	147	L16	O	—	グラフィック ディスプレイ イネーブル出力
GD0	G12	144	M16	O	—	グラフィック コントローラ データ出力
GD1	L11	127	V17	O	—	
GD2	H1	76	U6	O	—	
GD3	N2	96	V9	O	—	
GD4	M2	95	T8	O	—	
GD5	K3	90	U7	O	—	
GD6	L1	91	V7	O	—	
GD7	J1	80	U5	O	—	
GD8	G10	143	N18	O	—	
GD9	F9	145	M17	O	—	
GD10	G2	74	R6	O	—	
GD11	G3	75	T6	O	—	
GD12	L13	134	R16	O	—	
GD13	H10	133	P15	O	—	
GD14	J10	132	R15	O	—	
GD15	M13	131	T18	O	—	
GD16	K2	89	T7	O	—	
GD17	L3	97	U9	O	—	
GD18	F8	146	M18	O	—	
GD19	M12	130	T17	O	—	
GD20	E8	151	K17	O	—	
GD21	L2	92	V8	O	—	
GD22	J2	81	N4	O	—	
GD23	K12	137	P16	O	—	

凡例: CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

# PIC32MZ グラフィック (DA) ファミリ

表 1-22: DDR2 SDRAM コントローラの I/O ピン機能

ピン名	ピン番号			ピン タイプ	バッファ タイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>DDR2 SDRAM コントローラ</b>						
DDRCK	パッケージ 内部の DDR	パッケージ 内部の DDR	K2	O	SSTL	差動クロック
DDRCK			K1	O	SSTL	
DDRCKE			L2	O	SSTL	クロック イネーブル
DDRC $\overline{S}0$			N2	O	SSTL	チップセレクト 0
DDRRAS			M1	O	SSTL	行アドレス ストローブ
DDRCAS			N2	O	SSTL	列アドレス ストローブ
DDRWE			L1	O	SSTL	書き込みイネーブル ストローブ
DDRLDM			G3	O	SSTL	下位データバイトマスク
DDRUDM			A3	O	SSTL	上位データバイトマスク
DDRODT			N1	O	SSTL	オンダイ ターミネーション
DDRLDQS			E1	I/O	SSTL	下位データバイト修飾子ストローブ (差動)
DDRLDQS			E2	I/O	SSTL	
DDRUDQS			B2	I/O	SSTL	上位データバイト修飾子ストローブ (差動)
DDRUDQS			A2	I/O	SSTL	
DDRBA0			M2	O	SSTL	バンクアドレス選択 0
DDRBA1			M3	O	SSTL	バンクアドレス選択 1
DDRBA2			U4	O	SSTL	バンクアドレス選択 2
DDRA0			R1	O	SSTL	DDR2 アドレスバス
DDRA1			L3	O	SSTL	
DDRA2			N3	O	SSTL	
DDRA3			R2	O	SSTL	
DDRA4			P3	O	SSTL	
DDRA5			T1	O	SSTL	
DDRA6			U1	O	SSTL	
DDRA7			T2	O	SSTL	
DDRA8			U2	O	SSTL	
DDRA9			R3	O	SSTL	
DDRA10			P1	O	SSTL	
DDRA11			V2	O	SSTL	
DDRA12			T3	O	SSTL	
DDRA13			U3	O	SSTL	
DDRA14			T4	O	SSTL	
DDRA15	V3	O	SSTL			

**凡例:** CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト  
 SSTL = Stub Series Terminated Logic

# PIC32MZ グラフィック (DA) ファミリ

表 1-22: DDR2 SDRAM コントローラの I/O ピン機能 ( 続き )

ピン名	ピン番号			ピン タイプ	バッファ タイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
DDRDQ0	パッケージ 内部の DDR	パッケージ 内部の DDR	F1	I/O	SSTL	DDR2 データバス
DDRDQ1			J3	I/O	SSTL	
DDRDQ2			H1	I/O	SSTL	
DDRDQ3			G1	I/O	SSTL	
DDRDQ4			G2	I/O	SSTL	
DDRDQ5			H2	I/O	SSTL	
DDRDQ6			H3	I/O	SSTL	
DDRDQ7			F2	I/O	SSTL	
DDRDQ8			C1	I/O	SSTL	DDR2 データバス
DDRDQ9			C3	I/O	SSTL	
DDRDQ10			D2	I/O	SSTL	
DDRDQ11			F3	I/O	SSTL	
DDRDQ12			E3	I/O	SSTL	
DDRDQ13			D1	I/O	SSTL	
DDRDQ14			B3	I/O	SSTL	
DDRDQ15	C2	I/O	SSTL			

**凡例 :** CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト  
 SSTL = Stub Series Terminated Logic

# PIC32MZ グラフィック (DA) ファミリ

表 1-23: 電源、グランド、参照電圧の I/O ピン機能

ピン名	ピン番号			ピンタイプ	バッファタイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>電源とグランド</b>						
AVDD	D9	3, 4	F13, G13	P	P	アナログ モジュール向け正極性電源 このピンは常に接続する必要があります。
AVSS	C8, D8	5, 6	F12, G12	P	P	アナログ モジュール向けグランド基準 このピンは常に接続する必要があります。
VDDIO	B6, G9, H9, J9, K9, L6, L7, L9	19, 38, 86, 102, 117, 124, 155, 156, 165	C16, D15, D16, E15, F11, F15, G11, G15, H11, H12, H13, H15, J10, J15, K10, L11, L12, M12, M13, M15, N12, N13, R9, R10, R12, R13, R14	P	—	周辺ロジックおよび I/O ピン向け正極性電源 このピンは常に接続する必要があります。
VDDCORE	B4, C9, L8, N1	18, 39, 84, 116	D7, D14, R11, V4	P	—	周辺ロジック向け 1.8 V 正極性電源 このピンは常に接続する必要があります。
VSS	C2, F5, G5, G6, G7, G8, H7, H8, J7, J8, K7, K8	21, 22, 29, 37, 48, 49, 83, 87, 94, 103, 115, 122, 123, 153, 154	A5, B5, C7, D10, D11, D12, D13, F9, F10, G10, H10, J11, J12, J13, K11, K12, K13, K15, L10, L13, L15, M10, M11, N10, N11, R7, R8	P	—	ロジック、I/O ピン、USB 向けグランド基準 このピンは常に接続する必要があります。
HLVDIN	B12	173	E16	P	—	低電圧検出ピン
VBAT	D10	166	F16	P	—	バッテリー バックアップ セクション向け正極性電源 VBAT モードを使わない場合 (バッテリーに接続しない場合)、このピンは VDDIO に接続する事を推奨します。
VDDR1V8	H5, H6, J5, J6, K5, K6 (Note 2)	57, 58, 59, 60, 61, 62, 63, 67, 68, 72, 78 (Note 2)	H6, H7, H8, J6, J7, J8, K6, K7, K8, L6, L7, L8 (Note 2)	P	—	DDR2 SDRAM メモリ向け正極性電源

凡例 : CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

- Note 1:** デバイス底面の金属パッドは内部で VSS1V8 に接続しており、外部で 1.8 V グランドに接続する必要があります。  
**Note 2:** DDR 非実装デバイスの場合、このピンは 20k Ω 抵抗を介して VSS に接続する必要があります。  
**Note 3:** DDR 非実装デバイスの場合、このピンは未接続 (NC) です。



# PIC32MZ グラフィック (DA) ファミリ

表 1-24: JTAG、トレース、プログラミング/デバッグの I/O ピン機能

ピン名	ピン番号			ピン タイプ	バッファ タイプ	概要
	169 ピン LFBGA	176 ピン LQFP	288 ピン LFBGA			
<b>JTAG</b>						
TCK	E11	160	H16	I	ST	JTAG テストクロック入力ピン
TDI	A6	28	A10	I	ST	JTAG テストデータ入力ピン
TDO	C6	27	A11	O	—	JTAG テストデータ出力ピン
TMS	D2	53	D4	I	ST	JTAG テストモード選択ピン
<b>トレース</b>						
TRCLK	E4	54	E4	O	—	トレースクロック
TRD0	E2	64	H4	O	—	トレース データビット 0-3
TRD1	E3	56	G4	O	—	
TRD2	E1	65	J4	O	—	
TRD3	D1	55	F4	O	—	
<b>プログラミング/デバッグ</b>						
PGED1	C12	169	D18	I/O	ST	プログラミング/デバッグ通信チャンネル 1 向けデータ I/O ピン
PGEC1	B9	11	A14	I	ST	プログラミング/デバッグ通信チャンネル 1 向けクロック 入力ピン
PGED2	D12	170	D17	I/O	ST	プログラミング/デバッグ通信チャンネル 2 向けデータ I/O ピン
PGEC2	D7	13	B14	I	ST	プログラミング/デバッグ通信チャンネル 2 向けクロック 入力ピン
MCLR	K1	85	R5	I/P	ST	マスタクリア (リセット) 入力 このピンはアクティブ LOW でデバイスをリセットします。

凡例 : CMOS = CMOS 互換入出力      アナログ = アナログ入力      P = 電源  
 ST = CMOS レベルでのシュミットトリガ入力      O = 出力      I = 入力  
 TTL = TTL (Transistor-transistor Logic) 入力バッファ      PPS = ペリフェラル ピンセレクト

## 2.0 32 ビット マイクロコントローラ を使うためのガイドライン

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補う文書は、Microchip 社 PIC32 ウェブサイト ([www.microchip.com/pic32](http://www.microchip.com/pic32)) の [Documentation > Reference Manual] セクションから入手できます。

### 2.1 基本的な接続要件

32 ビット マイクロコントローラ PIC32MZ DA ファミリを使うには、デバイスピンの最小接続要件に関する注意が必要です。以下のピンは必ず接続する必要があります。

- 全ての VDDIO、VDD コア、VSS ピン (2.2「[デカップリング コンデンサ](#)」参照)
- 全ての AVDD/AVSS ピン (ADC モジュールを使わない場合でも接続が必要) (2.2「[デカップリング コンデンサ](#)」参照)
- VBAT ピン (2.2「[デカップリング コンデンサ](#)」参照)
- 全ての VDDR1V8/VSS1V8 ピン (2.2「[デカップリング コンデンサ](#)」参照)
- MCLR ピン (2.3「[マスタクリア \(MCLR\) ピン](#)」参照)
- PGECx/PGEDx ピン (インサーキット シリアル プログラミング (ICSP™) とデバッグ用に使用) (2.4「[ICSP ピン](#)」参照)
- OSC1/OSC2 ピン (外部オシレータソースを使う場合) (2.7「[外部オシレータピン](#)」)

以下のピンの接続も必要な場合があります。

VREF+/VREF- ピン (ADC モジュール用に外部参照電圧を実装する場合)

**Note:** AVDD/AVSS ピンは、ADC および ADC 参照電圧源を使うかどうかに関係なく、常に接続する必要があります。

### 2.2 デカップリング コンデンサ

VDD/VSS、AVDD/AVSS 等、全ての電源ピンにはデカップリング コンデンサが必要です。図 2-1 を参照してください。

デカップリング コンデンサに関しては以下を考慮する必要があります。

#### • コンデンサの値とタイプ

2 個のコンデンサ (0.1  $\mu$ F (10 ~ 20 V) と 0.01  $\mu$ F) を並列に接続する事を推奨します。0.1  $\mu$ F コンデンサには、共振周波数が 20 MHz 以上の低 ESR (低等価直列抵抗) 型を使う必要があります。2 つのコンデンサは互いに近付けて電源およびグランドピンの可能な限り近くに配置します。さらに、セラミック コンデンサの使用を推奨します。

#### • プリント基板上の配置

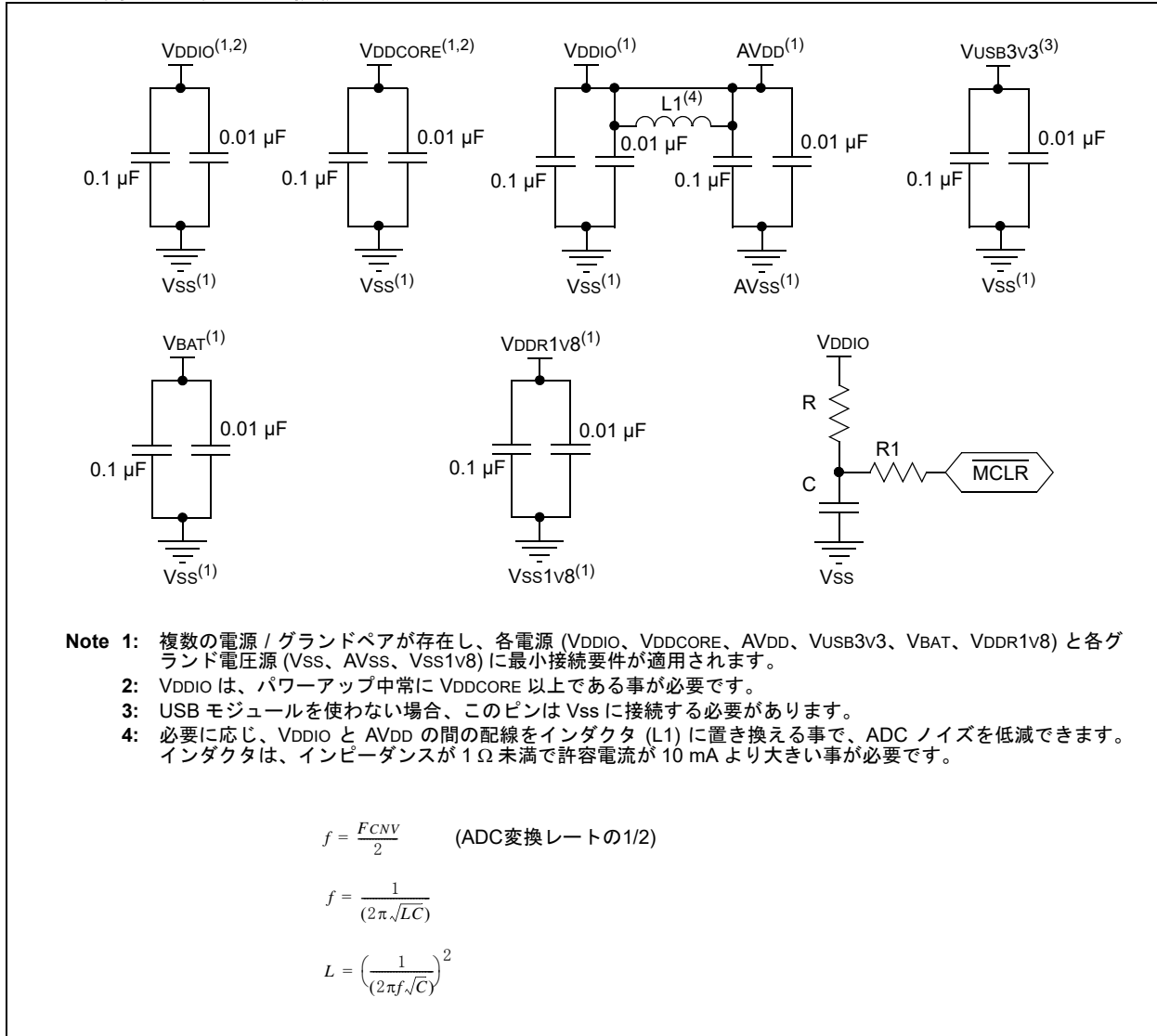
デカップリング コンデンサはできるだけピンの近くに配置する必要があります。コンデンサとデバイスを基板の同一面上に配置する事を推奨します。スペースに制約がある場合、ビアを使ってコンデンサを基板の別の層に配置する事もできますが、ピンからコンデンサまでのトレース長を 0.25 インチ (6 mm) 以下にする必要があります。

#### • 性能の最適化

電源回路からの基板レイアウトを設計する際は、まずデカップリング コンデンサまでの電源ラインの往復パターンを配置してからデバイスピンへのパターンを配置します。これにより、デカップリング コンデンサを電源システムの初段として確実に配置します。コンデンサと電源ピンの間のパターン長を最短にしてプリント基板の配線インダクタンスを低減する事も重要です。

# PIC32MZ グラフィック (DA) ファミリ

図 2-1: 推奨する最小限の接続



## 2.2.1 バルクコンデンサ

電源の安定性を向上させるため、VDDIO と VDDCORE に対してバルクコンデンサの使用を推奨します。標準的な容量は 4.7 ~ 47 μF です。このコンデンサは、できるだけデバイスの近くに配置する必要があります。



# PIC32MZ グラフィック (DA) ファミリ

## 2.3 マスタクリア (MCLR) ピン

MCLR ピンは、以下の 2 つのデバイス機能を提供します。

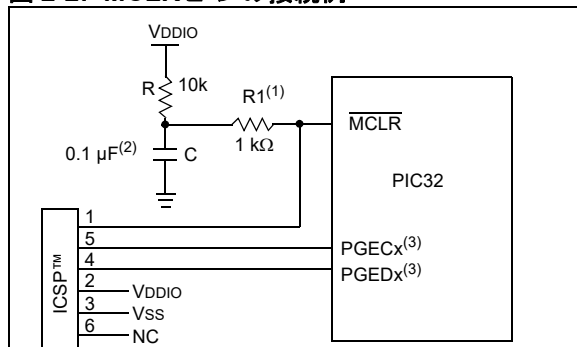
- デバイスリセット
- デバイスのプログラミング / デバッグ

MCLR ピンを LOW にする事でデバイスをリセットします。図 2-2 に、代表的な MCLR 回路を示します。デバイスのプログラミング / デバッグでは、このピンに付加される抵抗と容量に注意する必要があります。デバイス プログラマ / デバッガは MCLR ピンを駆動します。これによって電圧レベル ( $V_{IH}$ 、 $V_{IL}$ ) と高速な信号遷移に悪影響が及ぶ事を防ぐ必要があります。このため、アプリケーションとプリント基板の要件に基づいて特定の R と C の値を調整する必要があります。

例えば図 2-2 の場合、プログラミング / デバッグ動作時にコンデンサ C を MCLR ピンから切り離す事を推奨します。

図 2-2 内の素子は、MCLR ピンから 1/4 インチ (6 mm) 以内に配置します。

図 2-2: MCLR ピンの接続例



**Note 1:**  $470\Omega \leq R1 \leq 1\Omega$  は、静電放電 (ESD) または電氣的オーバーストレス (EOS) による MCLR ピンのブレイクダウン時に、外付けコンデンサ C から MCLR ピンに流れる電流を制限します。デバッグ / プログラミング ツールと干渉する事なく MCLR ピンの  $V_{IH}/V_{IL}$  仕様値が満たされる必要があります。

- 2: コンデンサの容量を適切に選択する事で、短いグリッチによる意図しないリセットを防ぐ事ができる他、POR 時のデバイスリセット期間を延ばす事ができます。
- 3: アクティブなデバッグ / プログラミング (PGECx / PGEDx) ピンには、プルアップ抵抗もバイパスコンデンサも使えません。

## 2.4 ICSP ピン

PGECx および PGEDx ピンは、インサーキット シリアル プログラミング (ICSP™) およびデバッグ用に使います。ICSP コネクタとデバイスの ICSP ピン間のトレースは、できるだけ短くする事を推奨します。ICSP コネクタで ESD が予測される場合、数十  $\Omega$  の抵抗 (100  $\Omega$  以下) を直列に接続する事を推奨します。

PGECx および PGEDx ピンにプルアップ抵抗、直列ダイオード、コンデンサを接続する事は推奨しません。それらは、プログラマ / デバッガとデバイスの間の通

信を障害します。アプリケーションがこれらのディスクリット部品を必要とする場合、それらはプログラミング / デバッグ時に回路から取り外す必要があります。また、容量性負荷の制限、ピン入力電圧 HIGH ( $V_{IH}$ ) および LOW ( $V_{IL}$ ) 要件の詳細は各デバイスのフラッシュプログラミング仕様書内の AC/DC 特性とタイミング要件を参照してください。

デバイスに設定する「通信チャンネル選択」(PGECx / PGEDx ピン) は、ICSP 用に使う MPLAB® ICD 3 または MPLAB REAL ICE™ への物理的接続に適合している事が必要です。

ICD 3 と REAL ICE の接続要件の詳細は、Microchip 社ウェブサイト ([www.microchip.com](http://www.microchip.com)) からダウンロード可能な以下の文書を参照してください。

- 『Using MPLAB® ICD 3』 (ポスター) (DS50001765)
- 『MPLAB® ICD 3 Design Advisory』 (DS50001764)
- 『MPLAB® REAL ICE™ インサーキット エミュレータ ユーザガイド』 (DS50001616)
- 『Using MPLAB® REAL ICE™ Emulator』 (ポスター) (DS50001749)

## 2.5 JTAG ピン

TMS、TDO、TDI、TCK ピンは JTAG (Joint Test Action Group) 規格に従うテストおよびデバッグ用に使います。JTAG コネクタとデバイスの JTAG ピン間のトレースは、できるだけ短くする事を推奨します。JTAG コネクタで ESD が予測される場合、数十  $\Omega$  の抵抗 (100  $\Omega$  以下) を直列に接続する事を推奨します。

TMS、TDO、TDI ピンにプルアップ抵抗、直列ダイオード、コンデンサを接続する事は推奨しません。それらはデバイスとプログラマ / デバッガ間の通信を障害します。アプリケーションがこれらのディスクリット部品を必要とする場合、それらはプログラミング / デバッグ時に回路から取り外す必要があります。また、容量性負荷の制限、ピン入力電圧 HIGH ( $V_{IH}$ ) および LOW ( $V_{IL}$ ) 要件の詳細は各デバイスのフラッシュプログラミング仕様書内の AC/DC 特性とタイミング要件を参照してください。

## 2.6 トレースピン

トレースピンをハードウェア トレース対応のプログラマに接続する事で、データを圧縮したリアルタイム命令トレースが可能です。トレース用に使う場合、TRD3、TRD2、TRD1、TRD0、TRCLK ピンはトレース専用にする必要があります。トレース ハードウェアは、トレースピンとトレースコネクタの間に 22  $\Omega$  の直列抵抗を必要とします。

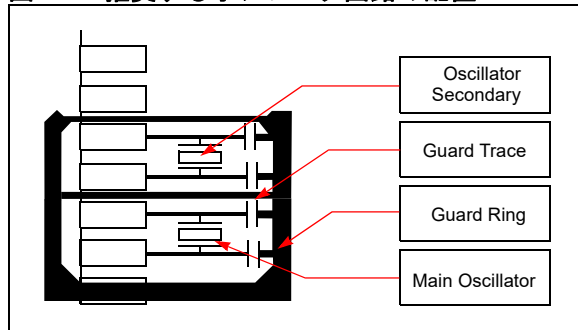
# PIC32MZ グラフィック (DA) ファミリ

## 2.7 外部オシレータピン

多くの MCU は、少なくとも 2 つのオシレータ オプション (高速のプライマリオシレータと低速のセカンダリオシレータ) を備えています (8.0「オシレータの設定」参照)。

オシレータ回路と MCU は基板の同一面上に配置する必要があります。また、オシレータ回路は各オシレータピンの近く (1/2 インチ (12 mm) 以内の距離) に配置します。負荷コンデンサは、デバイスと同一面上で、オシレータの隣に配置する必要があります。オシレータ回路は、グラウンドに接続した銅ベタで囲む事により、周囲の回路から隔離します。この銅ベタは、MCU のグラウンドに直接接続する必要があります。銅ベタ内に信号トレースまたは電源トレースを配置しない事が重要です。また、両面基板を使う場合、水晶振動子の裏側にはトレースを配置しない事が重要です。図 2-3 に、推奨レイアウトを示します。

図 2-3: 推奨するオシレータ回路の配置



## 2.8 未使用 I/O ピン

未使用の I/O ピンは、入力としてフローティング状態にしない事が重要です。それらのピンは出力として設定し、論理 LOW 状態に駆動できます。

あるいは、未使用ピンを入力として設定し、1 k $\Omega$  ~ 10 k $\Omega$  の抵抗を介して V<sub>ss</sub> に接続する事で、それらのピンを入力用に予約しておく事ができます。

## 2.9 高速周辺モジュールのための設計

PIC32MZ DA ファミリのデバイスは、組み込み環境向けとしては標準よりも大幅に高周波数で動作する周辺モジュールを内蔵しています。表 2-1 に、外部ピンで高速信号を生成する周辺モジュールの一覧を示します。

表 2-1: 外部ピンで高速信号を生成する周辺モジュール

周辺モジュール	高速信号ピン	信号ピンでの最大速度
DDR2 SDRAM コントローラ	DDRCLK DDRCLK DDRUDQS DDRUDQS DDRLDQS DDRLDQS	200 MHz
	DDRAx DDRDX	400 MHz
EBI	EBIAx EBIDx	50 MHz
HS USB	D+, D-	480 MHz
SDHC	SDCK, DATAx	50 MHz
SQI	SQICLK SQIDx	80 MHz

信号が高速であるため、これらの周辺モジュールを使った回路を設計する際は、以下を目標として、各種の影響因子とプリント基板上の部品配置に注意する必要があります。

- 電磁干渉の影響を最小化する
- 各信号が接続先に同時に到達するように配慮する
- クロストークを最小化する
- シグナル インテグリティを確保する
- システムノイズを低減する
- グラウンドバウンスと電源サグを最小化する

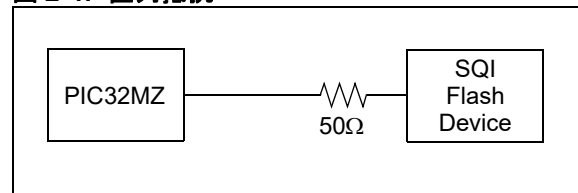
### 2.9.1 システム設計

#### 2.9.1.1 インピーダンスの適合

高速バス (特に SQI バス) 上に配置する部品 (周辺デバイス) のインピーダンスが接続先の PIC32MZ DA のピンのインピーダンスと適合しない場合、反射が生じて信号の品質が低下する可能性があります。

インピーダンスが適合する製品を選定できない場合、負荷に直列抵抗を接続する事でインピーダンスを適合させます (図 2-4 参照)。

図 2-4: 直列抵抗



# PIC32MZ グラフィック (DA) ファミリ

## 2.9.1.2 プリント基板の推奨レイアウト

上記の設計目標に向けて推奨するプリント基板のレイアウト方法は以下の通りです。

### • 部品配置

- バイパス コンデンサは、デバイスと同じプリント基板面で、デバイスの電源およびグランドピンにできるだけ近付けて配置します。
- 同じバスに接続する複数のデバイスは、セットアップ時間が長いデバイスほどPIC32MZ DAの近くに配置します。

### • 電源とグランド

- 多層プリント基板を使う事で電源プレーンとグランドプレーンを分離できます。
- 各グランドピンは別々にグランドプレーンに接続します。
- バイパス コンデンサのビアは、パッドのできるだけ近くに配置します (パッドの内側に配置する事を推奨)。
- 電源プレーンとグランドプレーンを使わない場合、電源トレースとグランドトレースを可能な限り太くします。
- 低 ESR の表面実装型バイパス コンデンサを使います。

### • クロックとオシレータ

- 水晶振動子は、PIC32MZ DA の OSC/SOSC ピンにできるだけ近付けて配置します。
- 高速信号の配線はクロックまたはオシレータから遠ざけます。
- クロックライン (SQICLK) では、ビアの使用と分岐は避けます。
- クロックラインには終端抵抗を配置します。

### • トレース

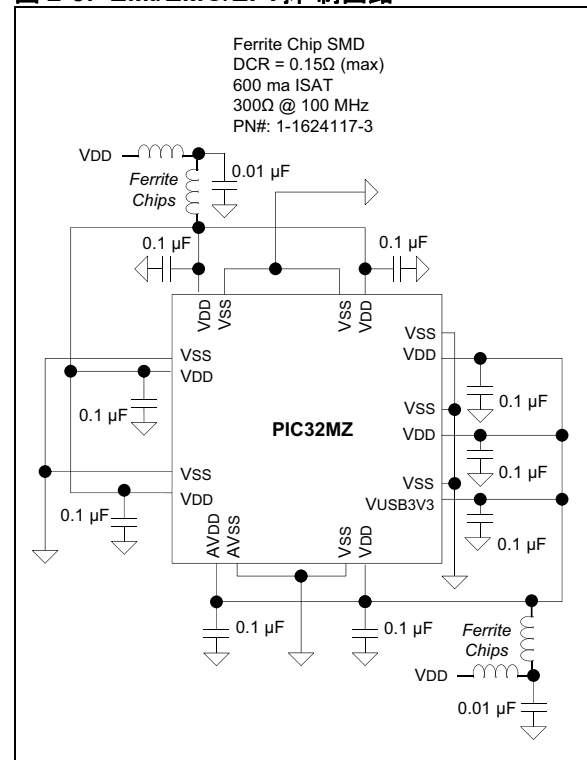
- 重要な信号は最短のトレースで配線します。
- DDR2 インターフェイスのレイアウトは、メーカーが推奨するガイドラインに従います。
- パラレルバス (EBIAx, EBIDx, SQIDx) のトレースは同じ長さにします。
- トレース同士が平行に走る区間を短くします (カップリングを低減するため)。
- クロックトレースは可能な限り直線にします。
- トレースは直角または鋭角に曲げない必要があります (円弧で曲げる事を推奨)。
- 層の異なるトレース同士は直角に交差させます (クロストークを最小化するため)。
- トレース同士はできるだけ遠ざけます (トレース幅の3倍以上の距離を保つ事を推奨)。
- 電源トレースはできるだけ太く短くします。
- 高速信号のトレースはグランドプレーンの近くに配置します。

## 2.9.1.3 EMI/EMC/EFT (IEC 61000-4-4、IEC 61000-4-2) を抑制するための配慮

システム全体のノイズを低減する (クリーンな電源を提供する) ために、LDO レギュレータの使用を推奨します。しかし、PIC32MZ DA 向けのローカル電源として降圧型または昇圧型スイッチング電源を使う場合、周囲の電氣的ノイズが多い場合、IEC 61000-4-4 および IEC 61000-4-2 に対応する試験条件が要求される場合、電源ピンに対して T 型フィルタ (L-C-L フィルタ) の使用を検討する必要があります (図 2-5 参照)。安定した電源を使った上に T 型フィルタを追加すると、EMI 発生源と EMI イベントによる影響を大幅に低減できます。

**Note:** 図 2-5 の回路は一部の電源 / グランドペアしか示していません。ペアの数はパッケージによって異なります。システム内の T 型フィルタの数は、フェライトチップの電流制限値と電源 / グランドペアの数によって決まります。例えば、288 ピン LFBGA パッケージで電流制限値が 600 mA の T 型フィルタを使う場合、T 型フィルタは 3 個必要です。

図 2-5: EMI/EMC/EFT 抑制回路

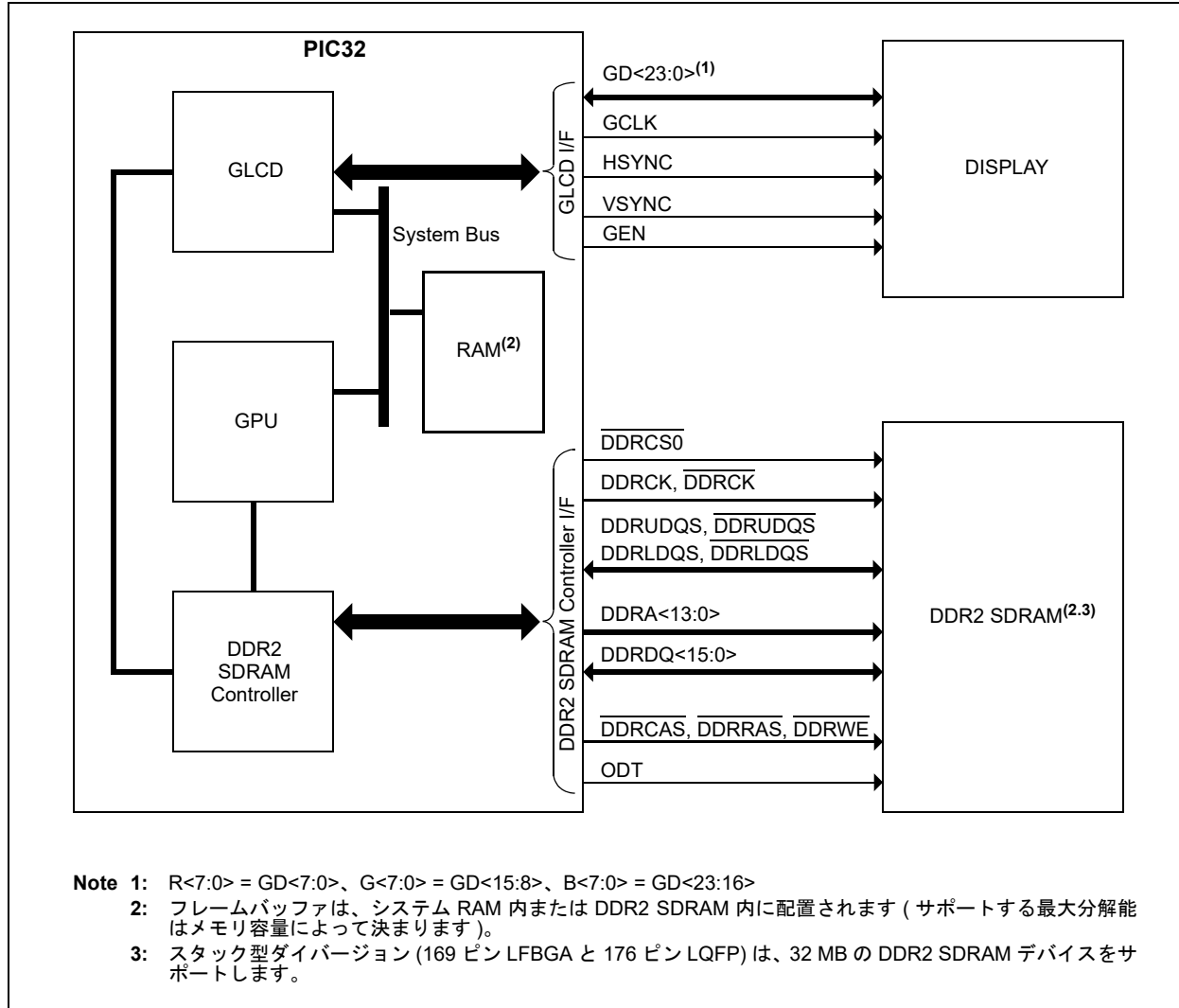


# PIC32MZ グラフィック (DA) ファミリ

## 2.10 代表的な接続例

図 2-6 に、代表的な接続例を示します。

図 2-6: グラフィック アプリケーション



# PIC32MZ グラフィック (DA) ファミリ

## 3.0 CPU

Note 1: 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。このセクションの内容を補う情報は『PIC32 Family Reference Manual, Section 50. CPU for Devices with MIPS32<sup>®</sup> microAptiv<sup>™</sup> and M-Class Cores』(DS60001192)に記載しています。この文書は、Microchip 社 PIC32 ウェブページ([www.microchip.com/pic32](http://www.microchip.com/pic32))の [Documentation > Reference Manual] セクションから入手できます。

2: MIPS32<sup>®</sup> microAptiv<sup>™</sup> マイクロプロセッサ コアの資料は以下で入手できます。  
<http://www.imgtec.com>

MIPS32 microAptiv マイクロプロセッサ コアは、PIC32MZ DAファミリデバイス プロセッサの中核です。CPU は命令のフェッチ、各命令のデコード、ソースオペランドのフェッチ、各命令の実行の後に、命令実行結果を適切なデスティネーションに書き込みます。

### 3.1 特長

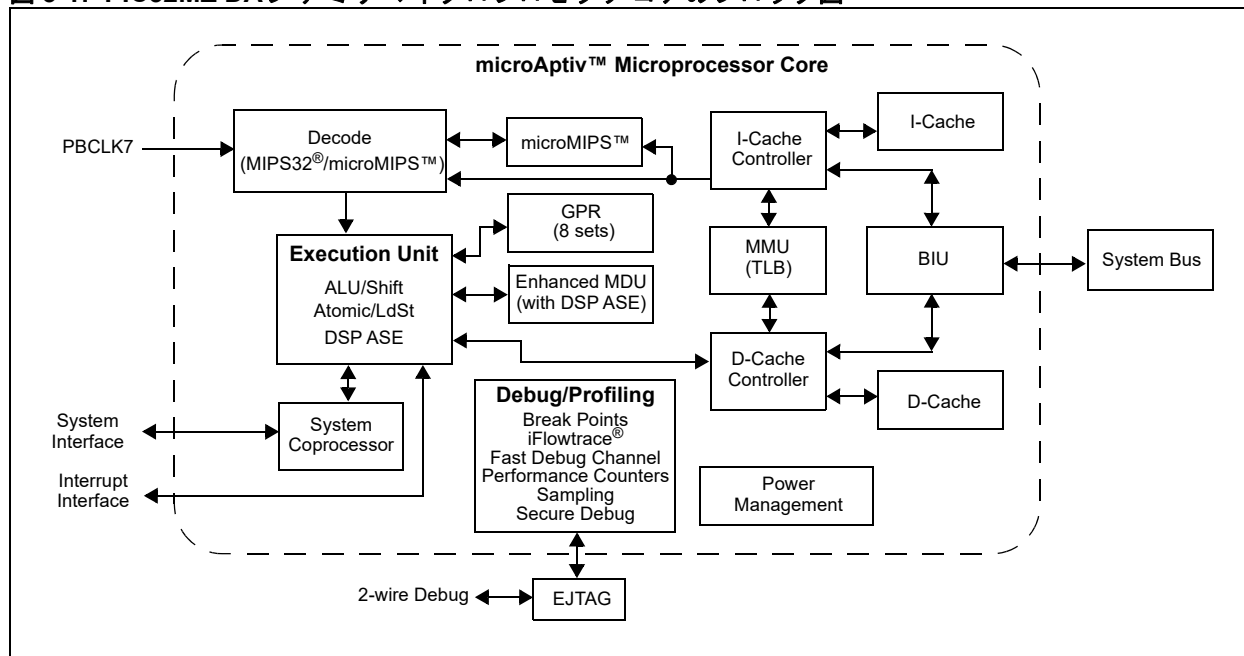
PIC32MZ DA ファミリ プロセッサコアの主な特長は以下の通りです。

- 5 段パイプライン
- 32 ビットのアドレスおよびデータパス
- MIPS32 拡張アーキテクチャ (リリース 2)
  - 積和および積差演算命令
  - デスティネーションに汎用レジスタを指定する乗算命令
  - 0/1 検出命令
  - WAIT 命令
  - 条件付き移動命令 (MOVN、MOVZ)
  - ベクタ割り込み
  - プログラムグルナ例外ベクタベース
  - アトミックな割り込み有効化 / 無効化
  - 割り込みハンドラのレイテンシを最小化する GPR シャドーレジスタ
  - ビットフィールド操作命令
  - 仮想メモリのサポート
- microMIPS 互換命令セット
  - MIPS32 の性能を維持しながらコードサイズ密度が向上
  - 全ての MIPS32 命令をサポート (Branch Likely 命令を除く)
  - 15 個の追加 32 ビット命令と、よく使われる MIPS32 命令に対応する 39 個の 16 ビット命令
  - 命令内で暗黙的なスタックポインタ
  - MIPS32 アセンブリおよび ABI 互換
- TLB (Translation Lookaside Buffer) メカニズムを備えた MMU
  - 32 デュアルエントリのフルアソシエティブ ジョイント TLB
  - 4 エントリのフルアソシエティブ命令 TLB
  - 4 エントリのフルアソシエティブデータ TLB
  - 4 KB ページ
- 分離した L1 データキャッシュと命令キャッシュ
  - 32 KB の 4 way 命令キャッシュ (I- キャッシュ)
  - 32 KB の 4 way データキャッシュ (D- キャッシュ)
- 自立した乗除算ユニット (MDU)
  - 最大命令レート: クロックあたり 1 回の 32x32 乗算
  - Early-in 繰り返し除算
  - 12 ~ 38 クロックのレイテンシ (被除数 (rs) の符号拡張に依存)
- 電源制御
  - 最小周波数: 0 MHz
  - 低消費電力モード (WAIT 命令で開始)
  - ローカルゲートクロックを広範囲に使用
- EJTAG デバッグと命令トレース
  - シングルステップ実行のサポート
  - 仮想命令とデータアドレス / 値ブレイクポイント
  - ハードウェア ブレイクポイントはアドレスマッチトリガとアドレスレンジトリガの両方をサポート
  - 8 命令 / 4 データの複合ブレイクポイント
- iFlowtrace<sup>®</sup> version 2.0 をサポート
  - リアルタイム命令プログラムカウンタ
  - 特殊イベントトレース機能
  - 2 個の性能カウンタ (ユーザ選択可能な 34 種類のカウンタ可能イベント)
  - プロセッサがデバッグモードに移行した時に無効化
- 4 個のウォッチレジスタ
  - 命令、データ読み出し、データ書き込みオプション
  - アドレスマッチ マスキング オプション
- DSP ASE 拡張
  - ネイティブの小数データ型演算
  - SIMD (Register Single Instruction Multiple Data) 演算 (加算、減算、乗算、シフト)
  - GPR を使ったシフト
  - ビット操作
  - コンペア -PICK
  - DSP 制御アクセス
  - インデックス付きロード
  - 分岐
  - 複素数オペランドの乗算
  - 可変ビット挿入 / 抽出
  - 仮想リングバッファ
  - 算術飽和 / オーバーフロー処理
  - ゼロサイクル オーバーヘッドの飽和 / 丸め演算

# PIC32MZ グラフィック (DA) ファミリ

PIC32MZ DA ファミリ プロセッサコアのブロック図  
を 図 3-1 に示します。

図 3-1: PIC32MZ DAファミリ マイクロプロセッサコアのブロック図



# PIC32MZ グラフィック (DA) ファミリ

## 3.2 アーキテクチャの概要

PIC32MZ DA ファミリデバイスの MIPS32 microAptiv マイクロプロセッサ コアは並列に相互動作する複数のロジックブロックを備え、高効率 / 高性能な演算エンジンを提供します。コアは以下のブロックを内蔵します。

- 実行ユニット
- 汎用レジスタ (GPR)
- 乗除算ユニット (MDU)
- システム制御コプロセッサ (CP0)
- メモリ管理ユニット (MMU)
- 命令 / データ キャッシュ コントローラ
- 電源管理
- 命令キャッシュとデータキャッシュ
- microMIPS サポート
- 拡張 JTAG (EJTAG) コントローラ

### 3.2.1 実行ユニット

プロセッサコアの実行ユニットは、シングルサイクル ALU 演算 (論理、シフト、加算、減算) と自立的乗除算ユニットによるロード / ストア アーキテクチャを実装します。本コアは、整数演算とアドレス計算に使う 32 個の 32 ビット汎用レジスタ (GPR) を備えます。割り込み / 例外処理時のコンテキスト切り換えオーバーヘッドを最小化するため、7 個のレジスタファイルシャドウセット (32 個のレジスタを含む) が追加されています。このレジスタファイルは 2 個の読み出しポートと 1 個の書き込みポートを備え、パイプライン内の動作遅延を最小化するために完全にバイパスされます。

実行ユニットは以下を含みます。

- 32 ビット加算器 - データアドレスの計算に使用
- アドレスユニット - 次の命令アドレスの計算に使用
- ロジック - 分岐の条件判定と分岐先アドレスの計算に使用
- ロード用アライナ
- トラップ条件コンパレータ
- バイパス マルチプレクサ - データを生成する命令の直後にその結果を使う命令が続く場合にストールを回避するために使用

- 先頭 0/1 検出ユニット - CLZ および CLO 命令の実装に使用
- 算術論理ユニット (ALU) - 算術演算とビットワイズ論理演算の実行に使用
- シフトとストア用アライナ
- DSP ALU および論理ブロック - DSP 命令 (算術 / シフト / 比較等) の実行に使用

### 3.2.2 乗除算ユニット (MDU)

プロセッサコアは乗除算ユニット (MDU) を内蔵しています。MDU は乗除算および DPS ASE 乗算命令用に分離したパイプラインを備えます。このパイプラインは整数ユニット (IU) パイプラインと並列に動作し、IU パイプラインがストールしてもこのパイプラインはストールしません。これにより、MDU 演算はシステムストールまたは他の整数ユニット命令 (またはその両方) によって部分的にマスクできます。

高性能 MDU は 32x32 Booth リコーディング乗算器、4 ペア (HI/LO) の結果 / 積算レジスタ、除算ステートマシン、必要とされるマルチプレクサ、制御ロジックで構成されます。32x32 の前の「32」は *rs* オペランドを表し、後の「32」は *rt* オペランドを表します。

MDU は、1 クロックサイクルあたり 1 回の乗算または積和演算の実行をサポートします。

除算は、単純な 1 ビット / クロックの繰り返しアルゴリズムを使って実装されます。Early-in 検出により、被除数 (*rs*) オペランドの符号拡張をチェックします。*rs* が 8 ビット幅である場合、23 回の繰り返しスキップされます。*rs* が 16 ビット幅の場合は 15 回の繰り返しスキップされ、24 ビット幅の場合は 7 回の繰り返しスキップされます。除算がまだ実行中である時に次の MDU 命令の発行が試みられた場合、その除算演算が完了するまで IU パイプラインはストールします。

表 3-1 に、プロセッサコアの乗除算命令の繰り返しレート (次の演算が可能になるまでの最大命令レート (サイクル数)) とレイテンシ (結果が利用可能になるまでのサイクル数) を示します。この表は、レイテンシと繰り返しレートの近似値をパイプライン クロックサイクル数で示します。

表 3-1: MIPS32 microAptiv マイクロプロセッサ コアの高性能整数乗除算ユニットのレイテンシと繰り返しレート

オペコード	オペランドのサイズ (mul <i>rt</i> ) (div <i>rs</i> )	レイテンシ	繰り返しレート
MULT/MULTU, MADD/MADDU, MSUB/MSUBU (HI/LO デスティネーション)	16 ビット	5	1
	32 ビット	5	1
MUL (GPR デスティネーション)	16 ビット	5	1
	32 ビット	5	1
DIV/DIVU	8 ビット	12/14	12/14
	16 ビット	20/22	20/22
	24 ビット	28/30	28/30
	32 ビット	36/38	36/38

# PIC32MZ グラフィック (DA) ファミリ

MIPS アーキテクチャは、演算結果を 4 個ある HI/LO レジスタペアの内の 1 つに格納する乗除算演算を定義しています。これらの値は、Move-From-HI (MFHI) 命令と Move-From-LO (MFLO) 命令を使って汎用レジスタファイルへ転送できます。

HI/LO をデスティネーションとする演算に加え、MIPS32 アーキテクチャは、最下位結果を HI/LO レジスタペアではなくプライマリ レジスタファイルに格納する乗算命令 (MUL) も定義しています。LO レジスタを使う場合、明示的な Move-From-LO(MFLO) 命令が必要ですが、これを不要とし、複数のデスティネーションレジスタをサポートする事により、乗算を多用する演算のスループットが向上します。

積和演算には MADD 命令を使い、積差演算には MSUB 命令を使います。MADD 命令は、2 つのオペランドを乗算した後に、積を HI および LO レジスタの内容に加算します。同様に、MSUB 命令は、2 つのオペランドを乗算した後に、積を HI および LO レジスタの内容から減算します。MADD および MSUB 演算は、DSP アルゴリズムで一般的に使われます。

MDU は、HI/LO レジスタを使う各種のシフト命令と、DSP ASE で定義されている乗算命令も実装しています。MDU は、このために必要な全てのデータ型をサポートします。また、MDU は ASE によって定義されている追加の 3 個の HI/LO レジスタを備えています。

表 3-3: コプロセッサ 0 (CP0) レジスタ

レジスタ番号	レジスタ名称	機能
0	Index	TLB 配列に対するインデックス (microAptiv MPU 専用)
1	Random	TLB 配列に対するランダムに生成されたインデックス (microAptiv MPU 専用)
2	EntryLo0	偶数番号の仮想ページに対する TLB エントリの下位部 (microAptiv MPU 専用)
3	EntryLo1	奇数番号の仮想ページに対する TLB エントリの下位部 (microAptiv MPU 専用)
4	Context/ UserLocal	メモリ内のページテーブル エントリを指すポインタ (microAptiv MPU 専用) 特権ソフトウェアを使って書き込む事ができ、RDHWR 命令を使って読み出す事ができるユーザ情報
5	PageMask/ PageGrain	PageMask は TLB エントリ内の可変ページサイズを制御し、PageGrain は TLB 内の 1 KB ページのサポートを有効します (microAptiv MPU 専用)。
6	Wired	固定された (wired の) TLB エントリの数を制御します (microAptiv MPU 専用)。
7	HWREna	非特権モードでの RDHWR 命令による一部ハードウェア レジスタへのアクセスを有効にします。
8	BadVAddr	直近に発生したアドレス関連例外のアドレスを格納します。
9	Count	プロセッサ サイクル数
10	EntryHi	TLB エントリの上位部 (microAptiv MPU 専用)
11	Compare	コアタイム割り込みの制御

表 3-2 に、DSP 乗算とドット積演算のレイテンシと繰り返しレートを示します。この表は、レイテンシと繰り返しレートの近似値をパイプライン クロックサイクル数で示します。

表 3-2: DSP 関連のレイテンシと繰り返しレート

オペコード	レイテンシ	繰り返しレート
乗算およびドット積 (積算後の飽和なし)	5	1
乗算およびドット積 (積算後の飽和あり)	5	1
積算なしの乗算	5	1

### 3.2.3 システム制御 コプロセッサ (CP0)

MIPS アーキテクチャでは、CP0 は仮想アドレスから物理アドレスへの変換とキャッシュ プロトコル、例外制御システム、プロセッサの診断機能、動作モード (カーネル/ユーザ/デバッグ)、割り込みの有効/無効に関与します。また、CP0 レジスタにアクセスする事により、コンフィグレーション情報(キャッシュサイズと設定連想度、microMIPS 等のオプションの有無等)も得られます (表 3-3 参照)。



# PIC32MZ グラフィック (DA) ファミリ

表 3-3: コプロセッサ 0 (CP0) レジスタ (続き)

レジスタ番号	レジスタ名称	機能
12	Status	プロセッサのステータスと制御
	IntCtl	ベクタ間隔の割り込み制御
	SRSCtl	シャドーレジスタ セットの制御
	SRSMap	シャドーレジスタ割り当ての制御
	View_IPL	Status レジスタのビットを出し入れする事なく優先度を読み / 書きできるようにします。
	SRSMAP2	2 個の 4 ビットフィールドを格納します (これらは、ベクタ番号からシャドーセット番号への割り当てを定義し、割り込み等をサービスする時に使われます)。
13	Cause	直近の例外の原因を示します。
	NestedExc	現在の例外が発生する前のエラー / 例外レベル ステータスビットの値を格納します。
	View_RIPL	Cause レジスタ内の RIPL ビットへの読み出しアクセスを有効にします。
14	EPC	直近の例外時のプログラム カウンタ
	NestedEPC	現在の例外が発生する前の例外プログラム カウンタの値を格納します。
15	PRID	プロセッサの ID とリビジョン
	Ebase	例外ベクタの例外ベースアドレス
	CDMMBase	共通デバイスメモリ マップのベースアドレス
16	Config	コンフィグレーション レジスタ
	Config1	コンフィグレーション レジスタ 1
	Config2	コンフィグレーション レジスタ 2
	Config3	コンフィグレーション レジスタ 3
	Config4	コンフィグレーション レジスタ 4
	Config5	コンフィグレーション レジスタ 5
	Config7	コンフィグレーション レジスタ 7
17	LLAddr	ロードリンク アドレス (microAptiv MPU 専用)
18	WatchLo	ウォッチポイントの下位アドレス (microAptiv MPU 専用)
19	WatchHi	ウォッチポイントの上位アドレス (microAptiv MPU 専用)
20-22	予約済み	PIC32 コア内で予約済み
23	Debug	EJTAG デバッグレジスタ
	TraceControl	EJTAG トレース制御
	TraceControl2	EJTAG トレース制御 2
	UserTraceData1	EJTAG ユーザ トレースデータ 1 レジスタ
	TraceBPC	EJTAG トレース ブレークポイント レジスタ
	Debug2	デバッグ制御 / 例外ステータス 1
24	DEPC	直近のデバッグ例外時のプログラム カウンタ
	UserTraceData2	EJTAG ユーザ トレースデータ 2 レジスタ
25	PerfCtl0	性能カウンタ 0 制御
	PerfCnt0	性能カウンタ 0
	PerfCtl1	性能カウンタ 1 制御
	PerfCnt1	性能カウンタ 1
26	ErrCtl	I キャッシュおよび D キャッシュの way 選択配列とデータ RAM 配列のソフトウェアテストを有効にします (microAptiv MPU 専用)。
27	予約済み	PIC32 コア内で予約済み
28	TagLo/DataLo	キャッシュタグ インターフェイスの下位部 (microAptiv MPU 専用)

# PIC32MZ グラフィック (DA) ファミリ

## 3.3 電源管理

本プロセッサコアは電源管理機能として低消費電力設計、アクティブ電源管理、動作のパワーダウンモード等を提供します。本プロセッサコアは、クロックの速度低減と停止をサポートする静的設計により、アイドル期間中のシステム消費電力を低減します。

### 3.3.1 命令による電源管理の制御

パワーダウンモードはWAIT命令を実行する事で開始します。電源管理の詳細は [4.0「省電力機能」](#) を参照してください。

### 3.3.2 ローカルなクロックゲート処理

プロセッサコアによる消費電力の大部分は、クロックツリーとクロッキングレジスタで発生します。PIC32MZファミリは、この動的な消費電力を低減するために、ローカルにゲート処理されたクロックを広く用います。

## 3.4 L1 命令キャッシュ / データキャッシュ

### 3.4.1 命令キャッシュ (I-キャッシュ)

I-キャッシュは32Kバイトのコア上のメモリブロックです。I-キャッシュは仮想的に索引付けされるため、キャッシュアクセスと並列に仮想-物理アドレス変換が発生します(物理アドレスへの変換のために待機する必要はありません)。タグは23ビットの物理アドレス、有効ビット、ロックビットを保持します。LRU置換ビットは別の配列内に保存されます。

I-キャッシュブロックは命令の行フィルバッファも含み、これを管理します。キャッシュへの書き込みデータが蓄積されると共に、行フィルバッファ内のデータを参照する命令フェッチが、そのデータのバイパスまたは外部インターフェイスからのデータによって処理されます。I-キャッシュ制御ロジックはバイパス機能を制御します。

プロセッサコアはI-キャッシュのロックをサポートします。キャッシュのロックにより、重要なコードまたはデータセグメントを行単位でキャッシュ内へロックできます。これによりシステムプログラマは、システムキャッシュの効率を最大化できます。

キャッシュのロック機能は、全てのI-キャッシュエントリで常に利用可能です。CACHE命令を使う事で、エントリごとにロックまたは非ロックを指定できます。

### 3.4.2 データキャッシュ (D-キャッシュ)

D-キャッシュは、32Kバイトのコア上のメモリブロックです。このキャッシュは仮想的に索引付け / 物理的にタグ付けされ、保護されます。D-キャッシュは仮想的に索引付けされるため、キャッシュアクセスと並列に仮想-物理アドレス変換が発生します。タグは23ビットの物理アドレス、有効ビット、ロックビットを保持します。キャッシュの各セットのdirtyビットとLRU置換アルゴリズムビットは、追加の配列内で保持されます。

プロセッサコアは、I-キャッシュと同様にD-キャッシュのロックもサポートします。重要データセグメントは、行ごとにキャッシュ内へロックされます。ロックされた内容はストアヒット時に更新可能ですが、キャッシュミス時の置換用に選択する事はできません。

D-キャッシュのロック機能は、全てのD-キャッシュエントリで常に利用可能です。CACHE命令を使う事で、エントリごとにロックまたは非ロックを指定できます。

### 3.4.3 属性

プロセッサコアのI-キャッシュおよびD-キャッシュ属性については、[コンフィグレーションレジスタの説明\(レジスタ3-1~レジスタ3-4\)](#)を参照してください。

## 3.5 EJTAG デバッグサポート

本プロセッサコアは、アプリケーションコードとカーネルコードのソフトウェアデバッグ向けに拡張JTAG (EJTAG) インターフェイスを提供します。本プロセッサコアは、標準のユーザモードとカーネルモードの動作に加えて、デバッグモードを提供します。デバッグモードは、デバッグ例外(ハードウェアブレイクポイント、シングルステップ実行等により生成)後に始まり、デバッグ例外からの戻り(DERET)命令が実行されるまで続きます。デバッグモード中は、プロセッサはデバッグ例外ハンドラルーチンを実行します。

EJTAG インターフェイスは、TAP (Test Access Port: テストデータをコアとの間で双方向に転送するためのシリアル通信ポート)を介して動作します。EJTAG仕様では、標準のJTAG命令に加えて、どのレジスタをどのように使うか指定する特殊な命令が定義されています。

## 3.6 MIPS® DSP ASE 拡張

MIPS DSP ASE (Application-Specific Extension) リビジョン2はMIPS32®アーキテクチャを拡張します。この拡張は、新しい整数命令とステート(新しいHI/LOアキュムレータレジスタペアとDSP制御レジスタを含む)を含みます。これらは、各種のDSPマルチメディアやオーディオ/ビデオアプリケーション向けのDSP関連アルゴリズムにとって極めて重要です。この拡張は、ネイティブの小数データ型演算とレジスタSIMD (Single Instruction Multiple Data) 演算(加算、減算、乗算、シフト等)をサポートします。加えて、DSPアルゴリズムを効率的に計算するために重要となる以下の機能も含まれます。

- 複素数オペランドの乗算
- 可変ビット挿入 / 抽出
- 仮想リングバッファの実装と使用
- 算術飽和 / オーバーフロー処理サポート
- ゼロサイクルオーバーヘッドの飽和 / 丸め演算

# PIC32MZ グラフィック (DA) ファミリ

## 3.7 microAptiv コアのコンフィグレーション

レジスタ 3-1 ~ レジスタ 3-4 に、PIC32MZ DA ファミリデバイス内の microAptiv コアの既定値コンフィグレーションを示します。

レジスタ 3-1: CONFIG: コンフィグレーション レジスタ (CP0 レジスタ 16、Select 0)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	r-1	U-0	U-0	U-0	U-0	U-0	U-0	R-0
	—	—	—	—	—	—	—	ISP
23:16	R-0	R-0	R-1	R-0	U-0	R-1	R-0	R-0
	DSP	UDI	SB	MDU	—	MM<1:0>		BM
15:8	R-0	R-0	R-0	R-0	R-0	R-1	R-0	R-0
	BE	AT<1:0>		AR<2:0>			MT<2:1>	
7:0	R-1	U-0	U-0	U-0	U-0	R/W-0	R/W-1	R/W-0
	MT<0>	—	—	—	—	K0<2:0>		

<b>凡例:</b>	r = 予約済みビット
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

- bit 31 **予約済み:** このビットは、CONFIG1 レジスタが存在する事を示すために、ハードウェアで「1」に結線されています。
- bit 30-25 **未実装:** 「0」として読み出し
- bit 24 **ISP:** 命令スクラッチパッド RAM ビット  
0 = 命令スクラッチパッド RAM は実装していない
- bit 23 **DSP:** データ クラッチパッド RAM ビット  
0 = データ スクラッチパッド RAM は実装していない
- bit 22 **UDI:** ユーザ定義ビット  
0 = CorExtend ユーザ定義命令は実装していない
- bit 21 **SB:** SimpleBE ビット  
1 = 内部バス インターフェイスでは SimpleBE (Simple バイトイネーブル) だけを許容する
- bit 20 **MDU:** 乗除算ユニットビット  
0 = 高速 / 高性能型 MDU
- bit 19 **未実装:** 「0」として読み出し
- bit 18-17 **MM<1:0>:** マージモード ビット  
10 = マージを許可する
- bit 16 **BM:** バーストモード ビット  
0 = バースト順はシーケンシャル
- bit 15 **BE:** エンディアン モードビット  
0 = リトルエンディアン
- bit 14-13 **AT<1:0>:** アーキテクチャ タイプビット  
00 = MIPS32
- bit 12-10 **AR<2:0>:** アーキテクチャ リビジョン レベルビット  
001 = MIPS32 リリース 2
- bit 9-7 **MT<2:0>:** MMU タイプビット  
001 = microAptiv MPU マイクロプロセッサ コアは TLB ベースの MMU を使う
- bit 6-3 **未実装:** 「0」として読み出し
- bit 2-0 **K0<2:0>:** Kseg0 コヒーレンシ アルゴリズム ビット  
010 = キャッシュなし

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 3-2: CONFIG1: コンフィグレーション レジスタ 1 (CP0 レジスタ 16、Select 1)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	r-1	R-0	R-1	R-1	R-1	R-1	R-1	R-0
	MMU Size<5:0>							IS<2>
23:16	R-1	R-1	R-0	R-1	R-1	R-0	R-1	R-1
	IS<1:0>		IL<2:0>			IA<2:0>		
15:8	R-0	R-1	R-1	R-0	R-1	R-1	R-0	R-1
	DS<2:0>			DL<2:0>			DA<2:1>	
7:0	R-1	U-0	U-0	R-1	R-0	R-0	R-1	R-0
	DA<0>	—	—	PC	WR	CA	EP	FP

**凡例:**

r = 予約済みビット  
R = 読み出し可能ビット  
W = 書き込み可能ビット  
U = 未実装、「0」として読み出し  
-n = POR 時の値  
1 = ビットはセット  
0 = ビットはクリア  
x = ビットは未知

- bit 31 **予約済み:** このビットは、CONFIG2 レジスタが存在する事を示すために、ハードウェアで「1」に結線されています。
- bit 30-25 **MMU Size<5:0>:** TLB エントリ数 - 1  
011111 = 32 個の TLB エントリ
- bit 24-22 **IS<2:0>:** 命令キャッシュセット ビット  
011 = way あたり 512 個の命令キャッシュセット
- bit 21-19 **IL<2:0>:** 命令キャッシュ行ビット  
011 = 命令キャッシュ行のサイズは 16 バイト
- bit 18-16 **IA<2:0>:** 命令キャッシュ連想度ビット  
011 = 命令キャッシュの連想度は 4 way
- bit 15-13 **DS<2:0>:** データ キャッシュセット ビット  
011 = way あたり 512 個のデータ キャッシュセット
- bit 12-10 **DL<2:0>:** データ キャッシュ行ビット  
011 = データ キャッシュ行のサイズは 16 バイト
- bit 9-7 **DA<2:0>:** データキャッシュ連想度ビット  
011 = データキャッシュの連想度は 4 way
- bit 6-5 **未実装:** 「0」として読み出し
- bit 4 **PC:** 性能カウンタビット  
1 = プロセッサコアは性能カウンタを実装している
- bit 3 **WR:** ウォッチレジスタ存在ビット  
1 = 4 個のウォッチレジスタが存在する
- bit 2 **CA:** コード圧縮実装ビット  
0 = MIPS16e<sup>®</sup> は実装していない
- bit 1 **EP:** EJTAG 実装ビット  
1 = コアは EJTAG を実装している
- bit 0 **FP:** 浮動小数点ユニット実装ビット  
0 = 浮動小数点ユニットは実装していない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 3-3: CONFIG3: コンフィグレーション レジスタ 3 (CP0 レジスタ 16、Select 3)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	r-1 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	R-0 IPLW<1:0>	R-1	R-0	R-0	R-0	R-1	R/W-y ISAONEXC <sup>(1)</sup>
15:8	R-y ISA<1:0> <sup>(1)</sup>	R-y	R-1 ULRI	R-1 RXI	R-1 DSP2P	R-1 DSPP	U-0	R-1 ITL
7:0	U-0 —	R-1 VEIC	R-1 VINT	R-0 SP	R-1 CDMM	U-0	U-0	R-1 TL

<b>凡例:</b>	r = 予約済みビット	y = POR 時にコンフィグレーション ビットからの値に設定
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31 **予約済み:** このビットは、CONFIG4 レジスタが存在する事を示すために、ハードウェアで「1」に結線されています。

bit 30-23 **未実装:** 「0」として読み出し

bit 22-21 **IPLW<1:0>:** Status レジスタの IPL ビットと Cause レジスタの RIPL ビットのビット幅  
01 = IPL および RIPL ビットは 8 ビット幅

bit 20-18 **MMAR<2:0>:** microMIPS アーキテクチャ リビジョン レベルビット  
000 = リリース 1

bit 17 **MCU:** MIPS MCU ASE 実装ビット  
1 = MCU™ ASE を実装している

bit 16 **ISAONEXC:** 例外時 ISA ビット<sup>(1)</sup>  
1 = 例外ベクタへの移行に microMIPS を使う  
0 = 例外ベクタへの移行に MIPS32 ISA を使う

bit 15-14 **ISA<1:0>:** 使用命令セットビット<sup>(1)</sup>  
11 = MIPS32 と microMIPS の両方を実装し、リセット時は microMIPS を使う  
10 = MIPS32 と microMIPS の両方を実装し、リセット時は MIPS32 ISA を使う

bit 13 **ULRI:** UserLocal レジスタ実装ビット  
1 = UserLocal コプロセッサ 0 レジスタを実装している

bit 12 **RXI:** PageGrain 内 RIE/XIE 実装ビット  
1 = RIE および XIE ビットを実装している

bit 11 **DSP2P:** MIPS DSP ASE リビジョン 2 実装ビット  
1 = DSP リビジョン 2 を実装している

bit 10 **DSPP:** MIPS DSP ASE 実装ビット  
1 = DSP を実装している

bit 9 **未実装:** 「0」として読み出し

bit 8 **ITL:** iFlowtrace ハードウェア実装ビット  
1 = コアに iFlowtrace を実装している

bit 7 **未実装:** 「0」として読み出し

bit 6 **VEIC:** 外部ベクタ割り込みコントローラ実装ビット  
1 = 外部割り込みコントローラのサポートを実装している

bit 5 **VINT:** ベクタ割り込み実装ビット  
1 = ベクタ割り込みを実装している

bit 4 **SP:** スモールページ ビット  
0 = ページサイズは 4 KB

bit 3 **CDMM:** 共通デバイスメモリ マップ実装ビット  
1 = CDMM を実装している

bit 2-1 **未実装:** 「0」として読み出し

bit 0 **TL:** トレースロジック実装ビット  
0 = トレースロジックは実装していない (これは古いタイプのトレースロジックであり、iFlowtrace (ITL ビット) によって置き換えられます)

**Note 1:** これらのビットは BOOTISA コンフィグレーション ビット (DEVCFG0<6>) の値に基づいて設定されます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 3-4: CONCI5: コンフィグレーション レジスタ 5 (CP0 レジスタ 16、Select 5)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-1
	—	—	—	—	—	—	—	NF

**凡例:**

r = 予約済み

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-1 **未実装:** 「0」として読み出し

bit 0 **NF:** ネスト異常検出機能実装ビット

1 = ネスト異常検出機能を実装している

レジスタ 3-5: CONFIG7: コンフィグレーション レジスタ 7 (CP0 レジスタ 16、Select 7)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-1	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	W1	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31 **W1:** IE 待機無視ビット

1 = プロセッサは割り込みによる WAIT 命令の解除を許容する

bit 30-0 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## 4.0 メモリ構成

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。このセクションの内容を補う情報は『PIC32 Family Reference Manual, Section 48. Memory Organization and Permissions』(DS60001214) に記載しています。この文書は、Microchip 社ウェブサイト PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

PIC32MZ DA マイクロコントローラは、連続した 4 GB の仮想メモリアドレス空間を提供します。プログラムメモリ、データメモリ、SFR、コンフィグレーションレジスタを含む全てのメモリ領域は、このアドレス空間内の一意のアドレスに配置されます。オプション設定により、プログラムメモリとデータメモリはユーザーメモリとカーネルメモリに分割できます。加えて、PIC32MZ DA ではデータメモリからコードを実行する事ができます。

主な特長は以下の通りです。

- 32 ビットのネイティブデータ幅
- ユーザモード (KUSEG) モードとカーネルモード (KSEG0/KSEG1/KSEG2/KSEG3) のアドレス空間
- コード保護用のブートフラッシュメモリ
- コードの暴走を阻止する信頼性の高いバス例外処理
- キャッシュ適用アドレス領域 (KSEG0/KSEG2) とキャッシュ非適用アドレス領域 (KSEG1/KSEG3)
- あらかじめ決められたメモリ領域への読み書きパーミッションアクセス

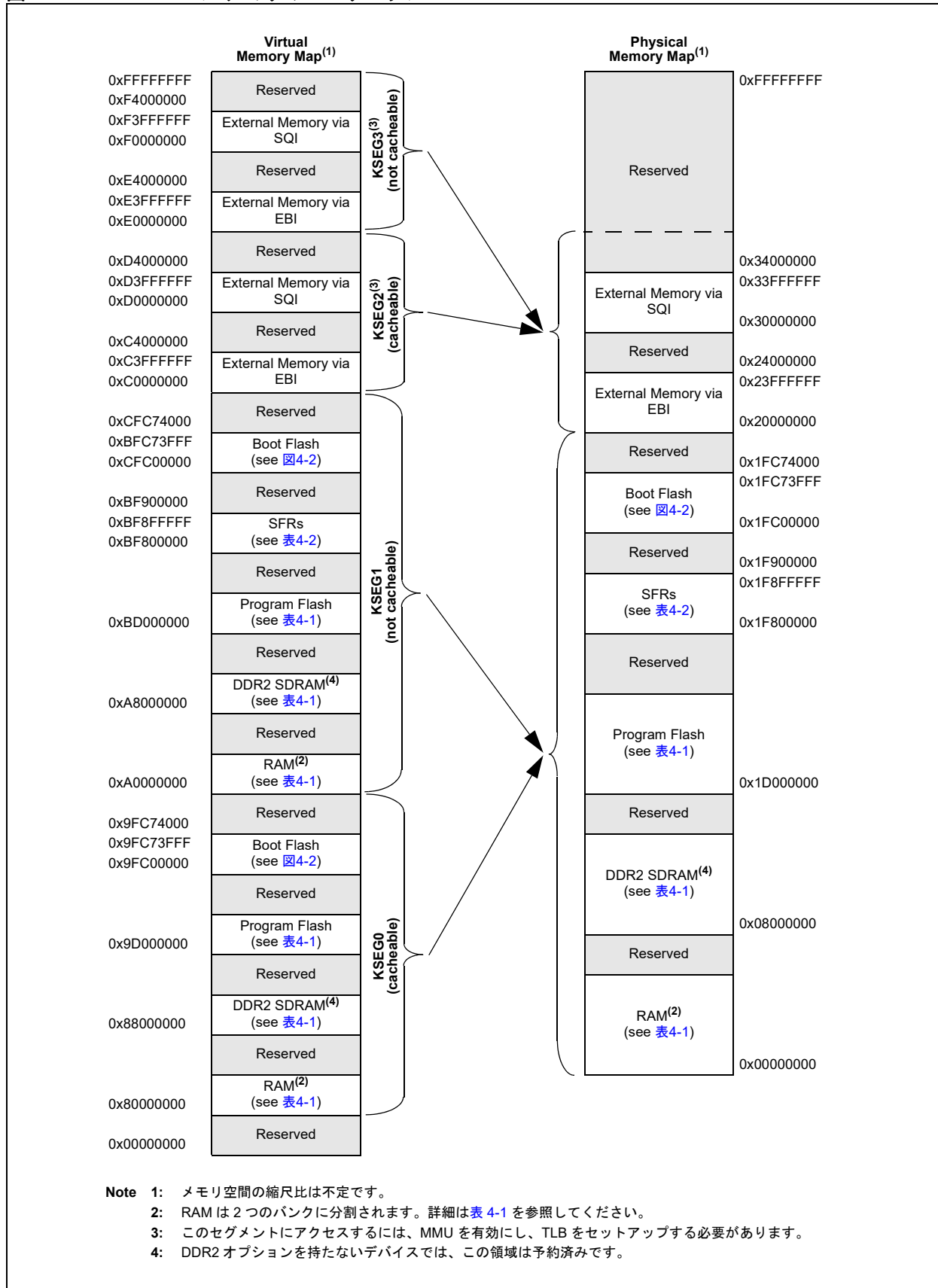
## 4.1 メモリ配置

PIC32MZ DA では、仮想アドレスと物理アドレスを使います。プログラムメモリ、データメモリ、周辺モジュール等を含む全てのハードウェアリソースは、それぞれ特定の物理アドレスに配置されます。CPU は、命令のフェッチと実行および周辺モジュールへのアクセスのために仮想アドレスだけを使います。CPU とは別にメモリにアクセスするバスマスタ周辺モジュール (DMA、フラッシュコントローラ等) は、物理アドレスを使います。

PIC32MZ DA のメインメモリマップを [図 4-1](#) に示し、ブートフラッシュとブートエイリアスのメモリマップ情報を [図 4-2](#) に示します。プログラムフラッシュ、RAM、DDR2 SDRAM のメモリマップ情報を [表 4-1](#) に示します。特殊機能レジスタ (SFR) のメモリマップ情報を [表 4-2](#) に示します。

# PIC32MZ グラフィック (DA) ファミリ

図 4-1: PIC32MZ DAファミリのメモリマップ





# PIC32MZ グラフィック (DA) ファミリ

表 4-1: アドレス マッピング テーブル

メモリ	サイズ	領域終了アドレス (KSEG1)	領域終了アドレス (KSEG0)	領域終了アドレス (物理)
プログラム フラッシュ	2 MB	0xBD1FFFFFF	0x9D1FFFFFF	0x1D1FFFFFF
	1 MB	0xBD0FFFFFF	0x9D0FFFFFF	0x1D0FFFFFF
DDR2 SDRAM	EXT <sup>(1)</sup>	0xAFFFFFFF	0x8FFFFFFF	0x0FFFFFFF
	32 MB <sup>(5)</sup>	0xA9FFFFFF	0x89FFFFFF	0x09FFFFFF
	— <sup>(2)</sup>	予約済み	予約済み	予約済み
RAM	640 KB <sup>(3)</sup>	0xA009FFFF	0x8009FFFF	0x0009FFFF
	256 KB <sup>(4)</sup>	0xA003FFFF	0x8003FFFF	0x0003FFFF

**Note 1:** 外付け DDR2 SDRAM の容量 (最大 128 MB) は EXTDDRSIZE<3:0> ビット (DEVCFG3<19:16>) で設定します。領域終了アドレスは、この容量に応じてスケーリングする必要があります。

**2:** DDR2 オプションを持たないデバイス

**3:** 640 KB の RAM (SRAM バンク 1 = 256 KB と SRAM バンク 2 = 384 KB) を実装したデバイス

**4:** 256 KB の RAM (SRAM バンク 1 = 128 KB と SRAM バンク 2 = 128 KB) を実装したデバイス

**5:** DDR2 SDRAM を内蔵したデバイスの DDR2 SDRAM 機能については [4.2 「DDR2 SDRAM」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

図 4-2: ブートおよびエイリアスメモリのマップ

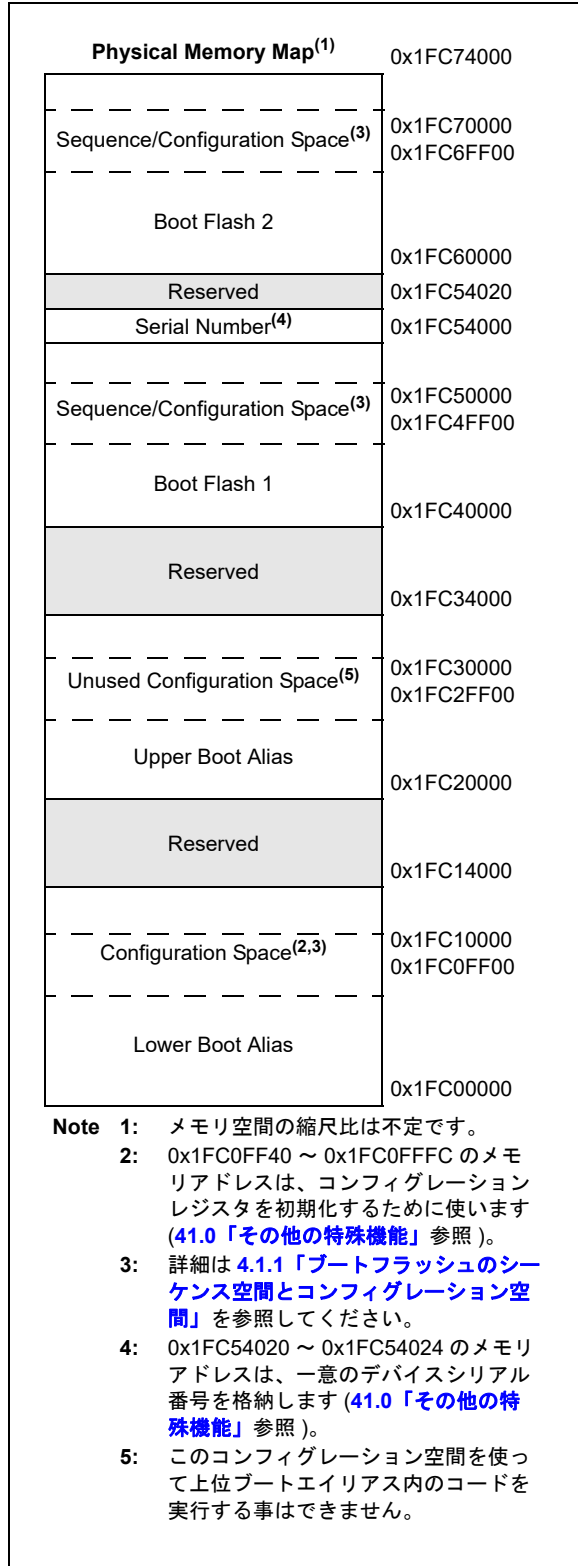


表 4-2: SFR のメモリマップ

周辺モジュール	仮想アドレス		
	ベースアドレス	オフセット	
システムバス <sup>(1)</sup>	0xBF8F0000	0x0000	
SDHC	0xBF8E0000	0xC000	
GPU		0xB000	
GLCD		0xA000	
DDRPHY		0x9100	
DDRC		0x8000	
RNG		0x6000	
暗号エンジン		0x5000	
USB		0x3000	
SQI1		0x2000	
EBI		0x1000	
プリフェッチ		0x0000	
DSCTRL		0xBF8C0000	0x0200
RTCC			0x0000
USBCR		0xBF880000	0x4000
Ethernet	0x2000		
CAN1、CAN2		0x0000	
PORTA-PORTK	0xBF860000	0x0000	
CTMU	0xBF840000	0xC200	
コンパレータ 1/2		0xC000	
ADC		0xB000	
OC1-OC9		0x4000	
IC1-IC9		0x2000	
Timer1-Timer9		0x0000	
PMP	0xBF820000	0xE000	
UART1-UART6		0x2000	
SPI1-SPI6		0x1000	
I2C1-I2C5		0x0000	
DMA	0xBF810000	0x1000	
割り込みコントローラ		0x0000	
HLVD	0xBF800000	0x1800	
PPS		0x1400	
オシレータ		0x1200	
CVREF		0x0E00	
デッドマンタイマ		0x0A00	
ウォッチドッグタイマ		0x0800	
フラッシュコントローラ		0x0600	
コンフィグレーション		0x0000	

**Note** 1: 4.4「システムバスの調停」に重要な法務情報を記載しています。

# PIC32MZ グラフィック (DA) ファミリ

## 4.1.1 ブートフラッシュのシーケンス空間とコンフィグレーション空間

シーケンス空間は、エイリアスされた領域によってエイリアスされるブートフラッシュ空間を識別するために使います。BF1SEQ3 ワードの TSEQ<15:0> ビットの値が BF2SEQ3 ワードの TSEQ<15:0> ビットの値以上である場合、ブートフラッシュ 1 は下位のブートエイリアス領域によってエイリアスされ、ブートフラッシュ 2 は上位ブートエイリアス領域によってエイリアスされます。BF2SEQ3 ワードの TSEQ<15:0> ビットの値が BF1SEQ3 ワードの TSEQ<15:0> ビットの値より大きい場合、これと逆になります。BFxSEQ3 ワードのメモリ位置は表 4-3 と表 4-4 参照してください。

CSEQ<15:0> ビットには TSEQ<15:0> ビットの補数を格納する必要があります。そうではない場合、TSEQ<15:0> ビットの値は無効と見なされ、代替シーケンスが使われます。詳細は 4.1.2「シーケンスおよびコンフィグレーション空間の代替ワード」を参照してください。

ブートフラッシュ メモリがエイリアスされると、下位ブートエイリアス領域内のコンフィグレーション空間がコンフィグレーションワード (DEVSIGN0、DEVCP0、DEVCFGx、および関連する代替コンフィグレーションレジスタ) 向けのベースとして使われます。従って、下位ブートエイリアス領域のメモリによってエイリアスされるブートフラッシュ領域は、適切なメモリ位置にコンフィグレーション値を格納する必要があります。

**Note:** データをシーケンス空間とコンフィグレーション空間に書き込む場合、ワード書き込み動作 (NVMOP<3:0> = 0001) を使わない必要があります。

## 4.1.2 シーケンスおよびコンフィグレーション空間の代替ワード

コンフィグレーション空間とシーケンス空間内の全てのワード (プライマリワード) には、対応する代替ワードが存在します (代替ワードの名前には、対応するプライマリワードの名前の前に文字「A」が付加されず)。デバイスの起動中に、プライマリワードが読み出されて修正不可能な ECC エラーが見つかった場合、BCFGERR (RCON<27>) フラグがセットされて代替ワードが使われます。プライマリワードと代替ワードの両方で修正不可能な ECC エラーが見つかった場合、BCFGFAIL (RCON<26>) フラグがセットされて既定値コンフィグレーションが使われます。

表 4-3: ブートフラッシュ 1 のシーケンスおよびコンフィグレーションワードのまとめ

仮アドレス (BFC4_#)	レジスタ名	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
FF3C	ABF1DEVCFG4	31:0	Note: ビットの説明は表 41-2 参照																xxxx
FF40	ABF1DEVCFG3	31:0																	xxxx
FF44	ABF1DEVCFG2	31:0																	xxxx
FF48	ABF1DEVCFG1	31:0																	xxxx
FF4C	ABF1DEVCFG0	31:0																	xxxx
FF50	ABF1DEVCP3	31:0																	xxxx
FF54	ABF1DEVCP2	31:0																	xxxx
FF58	ABF1DEVCP1	31:0																	xxxx
FF5C	ABF1DEVCP0	31:0																	xxxx
FF60	ABF1DEVSIGN3	31:0																	xxxx
FF64	ABF1DEVSIGN2	31:0																	xxxx
FF68	ABF1DEVSIGN1	31:0																	xxxx
FF6C	ABF1DEVSIGN0	31:0																	xxxx
FF70	ABF1SEQ3	31:16 15:0																	CSEQ<15:0> TSEQ<15:0>
FFF4	ABF1SEQ2	31:16 15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FF78	ABF1SEQ1	31:16 15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FF7C	ABF1SEQ0	31:16 15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFBC	BF1DEVCFG4	31:0	Note: ビットの説明は表 41-1 参照																xxxx
FFC0	BF1DEVCFG3	31:0																	xxxx
FFC4	BF1DEVCFG2	31:0																	xxxx
FFC8	BF1DEVCFG1	31:0																	xxxx
FFCC	BF1DEVCFG0	31:0																	xxxx
FFD0	BF1DEVCP3	31:0																	xxxx
FFD4	BF1DEVCP2	31:0																	xxxx
FFD8	BF1DEVCP1	31:0																	xxxx
FFDC	BF1DEVCP0	31:0																	xxxx
FFE0	BF1DEVSIGN3	31:0																	xxxx
FFE4	BF1DEVSIGN2	31:0																	xxxx
FFE8	BF1DEVSIGN1	31:0																	xxxx
FFEC	BF1DEVSIGN0	31:0																	xxxx
FFF0	BF1SEQ3	31:16 15:0																	CSEQ<15:0> TSEQ<15:0>
FFF4	BF1SEQ2	31:16 15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFF8	BF1SEQ1	31:16 15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFFC	BF1SEQ0	31:16 15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

表 4-4: ブートフラッシュ 2 のシーケンスおよびコンフィグレーションワードのまとめ

仮アドレ ス (BFC_#)	レジスタ名	ビット レンジ	Bit														全リセッ ト		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
FF3C	ABF2DEVCFG4	31:0	Note: ビットの説明は表 41-2 参照														XXXX		
FF40	ABF2DEVCFG3	31:0															XXXX		
FF44	ABF2DEVCFG2	31:0															XXXX		
FF48	ABF2DEVCFG1	31:0															XXXX		
FF4C	ABF2DEVCFG0	31:0															XXXX		
FF50	ABF2DEVCP3	31:0															XXXX		
FF54	ABF2DEVCP2	31:0															XXXX		
FF58	ABF2DEVCP1	31:0															XXXX		
FF5C	ABF2DEVCP0	31:0															XXXX		
FF60	ABF2DEVSIGN3	31:0															XXXX		
FF64	ABF2DEVSIGN2	31:0															XXXX		
FF68	ABF2DEVSIGN1	31:0															XXXX		
FF6C	ABF2DEVSIGN0	31:0															XXXX		
FF70	ABF2SEQ3	31:16															CSEQ<15:0>		
		15:0	TSEQ<15:0>														XXXX		
FFF4	ABF2SEQ2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
FF78	ABF2SEQ1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
FF7C	ABF2SEQ0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
FFBC	BF2DEVCFG4	31:0	Note: ビットの説明は表 41-1 参照														XXXX		
FFC0	BF2DEVCFG3	31:0															XXXX		
FFC4	BF2DEVCFG2	31:0															XXXX		
FFC8	BF2DEVCFG1	31:0															XXXX		
FFCC	BF2DEVCFG0	31:0															XXXX		
FFD0	BF2DEVCP3	31:0															XXXX		
FFD4	BF2DEVCP2	31:0															XXXX		
FFD8	BF2DEVCP1	31:0															XXXX		
FFDC	BF2DEVCP0	31:0															XXXX		
FFE0	BF2DEVSIGN3	31:0															XXXX		
FFE4	BF2DEVSIGN2	31:0															XXXX		
FFE8	BF2DEVSIGN1	31:0															XXXX		
FFEC	BF2DEVSIGN0	31:0															XXXX		
FFF0	BF2SEQ3	31:16															CSEQ<15:0>		
		15:0	TSEQ<15:0>														XXXX		
FFF4	BF2SEQ2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
FFF8	BF2SEQ1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
FFFC	BF2SEQ0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	XXXX

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-1: BFxSEQ3/ABFxSEQ3: ブートフラッシュ「x」シーケンスワード 0 レジスタ (x = 1, 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	CSEQ<15:8>							
23:16	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	CSEQ<7:0>							
15:8	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	TSEQ<15:8>							
7:0	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	TSEQ<7:0>							

**凡例:**

P = プログラム可能ビット

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **CSEQ<15:0>**: ブートフラッシュ Complement シーケンス番号ビット

bit 15-0 **TSEQ<15:0>**: ブートフラッシュ True シーケンス番号ビット

**Note:** BFxSEQ0 ~ BFxSEQ2 レジスタと ABFxSEQ0 ~ ABFxSEQ2 レジスタは、BFxSEQ3/ABFxSEQ3 レジスタに対するクワッドワードプログラミング動作に使用します。これらのレジスタは有効な情報を格納しません。

# PIC32MZ グラフィック (DA) ファミリ

## 4.2 DDR2 SDRAM

DDR2 SDRAM メモリをスタックしたデバイスは 32 MB の DDR2 SDRAM をサポートします。これらのデバイス内のメモリは 4,194,304 x 4 バンク x 16 ビットとして構成されます。DDR2 SDRAM のアドレスレンジは図 4-1 ~ 表 4-1 を参照してください。

### 4.2.1 特長

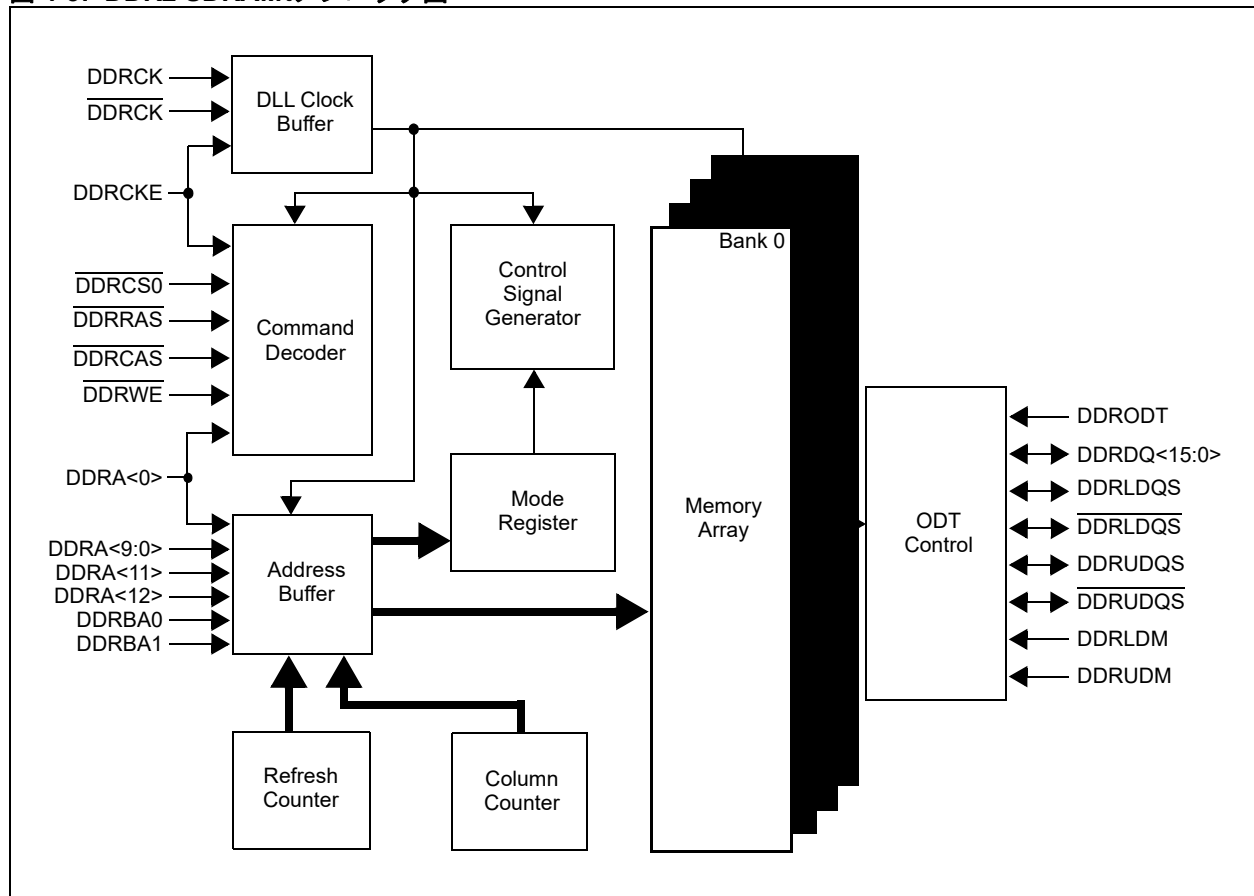
DDR2 SDRAM の特長は以下の通りです。

- ダブル データレート アーキテクチャ:  
クロックサイクルあたり 2 個のデータ転送
- CAS レイテンシ: 3、4
- バースト長: 8
- 双方向の差動データストロブ (DDRUDQS/DDRLDQS、DDRUDQS/DDRLDQS) はデータと一緒に送受信
- エッジアラインの読み出しデータ、センターアラインの書き込みデータ
- DLL はデータ (DDR DQx) と QDS (DDRxDQS、DDRxDQS) の遷移をクロックに対してアラインメントする
- 差動クロック入力 (DDRCK、/DDRCK)
- データ書き込み用のデータマスク (DDRUDM、DDRLDM)

- コマンドは DDRCK の各立ち上がりエッジで入力され、データとデータマスクは DDRxDQS の両方のエッジで参照される
- ポステッド CAS による設定可能アディティブ レイテンシをサポートする事で コマンドバスとデータバスの効率が向上
- 読み出しレイテンシ = アディティブ レイテンシ + CAS レイテンシ (RL = AL + CL)
- OCD (Off-Chip-Driver) インピーダンス調整と ODT (On-Die-Termination) による信号品質の向上
- 読み書きバーストのオート プリチャージ動作
- オート リフレッシュ モードとセルフ リフレッシュ モード
- プリチャージド パワーダウンとアクティブ パワーダウン
- 書き込みレイテンシ = 読み出しレイテンシ - 1 (WL = RL - 1)

図 4-3 に、DDR2 SDRAM のブロック図を示します。

図 4-3: DDR2 SDRAM のブロック図



# PIC32MZ グラフィック (DA) ファミリ

## 4.3 タイミングパラメータ

表 4-5 は、32 MB DDR2 SDRAM を内蔵したデバイスにのみ適用します。外付けの DDR2 SDRAM メモリをサポートするデバイスの場合、タイミングパラメータはベンダーのデータシートを参照してください。

表 4-5: タイミングパラメータ

パラメータ	概要	値	単位
tRFC	オートリフレッシュサイクル時間	130	ns
tWR	書き込みリカバリ時間	25	ns
tRP	プリチャージ-アクティベートコマンド間の遅延時間	20	ns
tRCD	アクティベート-読み書きコマンド間の遅延時間	20	ns
tRRD	2つのRAS(行アドレスストローブ)コマンド間の遅延時間	7.5	ns
tWTR	読み出し-書き込みコマンド間の遅延時間	15	ns
tRTP	読み出し-プリチャージコマンド間の遅延時間	20	ns
tDLLK	DLL ロック遅延時間	200	クロックサイクル
tRAS	アクティベート-プリチャージコマンド間の最小遅延時間	40	ns
tRC	行サイクル時間	110	ns
tFAW	4バンクアクティベートウィンドウ	35	ns
tMRD	モードレジスタ設定コマンドサイクル遅延	4	クロックサイクル
tXP	省電力モード終了遅延	6	クロックサイクル
tCKE	省電力モード最小遅延	6	クロックサイクル
RL	CAS レイテンシ	4	クロックサイクル
tRFI	周期的リフレッシュの平均間隔	7.8	μs
WL	書き込みレイテンシ	3	クロックサイクル
BL	バースト長さ(サイクル数)	8	クロックサイクル

## 4.4 システムバスの調停

**Note:** システムバスの相互接続には、Sonics, Inc. の SonicsSX<sup>®</sup> Interconnect の 1 つまたは複数のインスタンスを使います。本書は Sonics, Inc. の機密情報を構成する内容 (© 2003-2015 Sonics, Inc.) を含みます。そのような全ての内容と商標の使用については、Sonics, Inc. から許可を受けています。

PIC32MZ DA ファミリのブロック図 (図 1-1) が示す通り、システム内には複数のイニシエータ モジュール (I1 ~ I14) が存在し、それらは各種のターゲット モジュール (T1 ~ T23) にアクセスできます。表 4-6 に、各イニシエータからアクセス可能なターゲットの一覧を示します。システムバスは、複数のイニシエータから互いに異なるターゲットへの同時アクセスをサポートしません。複数のイニシエータが同時に同じターゲットにアクセスを試みた場合、システムバスはバス調停を実行します。



表 4-6: イニシエータからターゲットへのアクセスの一覧

ターゲット 番号	イニシエータ ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14
	名称	CPU	DMA 読み出し	DMA 書き込み	USB	Ethernet 読み出し	Ethernet 書き込み	CAN1	CAN2	SQ11	フラッシュ コントローラ	暗号 エンジン	GLCD	GPU	SDHC
1	フラッシュメモリ: プログラムフラッシュ ブートフラッシュ プリフェッチモジュール	X	X		X	X		X	X			X			
2	RAMバンク1メモリ	X	X	X	X	X	X	X	X	X	X	X	X	X	X
3	RAMバンク2メモリ	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5	周辺モジュールセット1: システム制御 フラッシュ制御 DMT CVREF PPS入力 PPS出力 割り込み DMA WDT	X													
6	周辺モジュールセット2: SPI1-SPI6 I2C1-I2C5 UART1-UART6 PMP	X	X	X											
7	周辺モジュールセット3: Timer1-Timer9 IC1-IC9 OC1-OC9 ADC コンパレータ1 コンパレータ2 CTMU	X	X	X											
8	周辺モジュールセット4: PORTA-PORTK	X	X	X											
9	周辺モジュールセット5: CAN1 CAN2 Ethernetコントローラ	X	X	X											
10	USB	X	X	X											
11	SQ11を介する外部メモリとSQ11モジュール	X													
12	暗号エンジン	X													
13	RNGモジュール	X													
14	グラフィックLCDコントローラ	X													
15	DDR2を介する外部メモリとDDR2ターゲット0	X													
16	DDR2を介する外部メモリとDDR2ターゲット1/2		X	X	X	X	X	X	X	X	X	X			X
17	DDR2を介する外部メモリとDDR2ターゲット3/4												X <sup>(1)</sup>	X <sup>(1)</sup>	

Note 1: DDR2 SDRAM をフレームバッファ向けに使うため、GLCD と GPU は DDR2 SDRAM コントローラに直接接続されます。調停制御は、DDR2 SDRAM コントローラの調停エンジンを介して行われます。詳細は『PIC32 Family Reference Manual, DDR2 SDRAM Controller』(DS60001321) を参照してください。

表 4-6: イニシエータからターゲットへのアクセスの一覧 ( 続き )

ターゲット 番号	イニシエータ ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14
	名称	CPU	DMA 読み出し	DMA 書き込み	USB	Ethernet 読み出し	Ethernet 書き込み	CAN1	CAN2	SQI1	フラッシュ コントローラ	暗号 エンジン	GLCD	GPU	SDHC
18	2D グラフィック処理ユニット	X													
19	セキュア デジタルホストコントローラ	X													
20	DDR2 PHY 制御レジスタ インターフェイス	X													
21	DDR2 制御レジスタ インターフェイス	X													
22	周辺モジュールセット 6: RTCC DSCTRL	X													
23	EBI を介する外部メモリと EBI モジュール	X	X	X	X	X	X	X	X	X		X			X

**Note 1:** DDR2 SDRAM をフレームバッファ向けに使うため、GLCD と GPU は DDR2 SDRAM コントローラに直接接続されます。調停制御は、DDR2 SDRAM コントローラの調停エンジンを介して行われます。詳細は『PIC32 Family Reference Manual、DDR2 SDRAM Controller』(DS60001321) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

システムバス調停は LRS (Least Recently Serviced) 方式を実装します (設定変更はできません)。

各イニシエータに対する調停方式を表4-7に示します。

表 4-7: イニシエータ ID に対する調停方式

ID	名称	調停方式
1	CPU	LRS
2	DMA 読み出し	LRS
3	DMA 書き込み	LRS
4	USB	LRS
5	Ethernet 読み出し	LRS
6	Ethernet 書き込み	LRS
7	CAN1	LRS
8	CAN2	LRS
9	SQI1	LRS
10	フラッシュ コントローラ	LRS
11	暗号エンジン	LRS
12	GLCD <sup>(1)</sup>	LRS
13	GPU <sup>(1)</sup>	LRS
14	SDHC	LRS

**Note 1:** DDR2 SDRAM をフレームバッファ向けに使うため、GLCD と GPU は DDR2 SDRAM コントローラに直接接続されます。調停制御は、DDR2 SDRAM コントローラの調停エンジンを介して行われます。

## 4.5 パーミッションアクセスとシステムバス レジスタ

PIC32MZ DA ファミリ マイクロコントローラのシステムバスは、システムバス上のトランザクション イニシエータにアクセス制御能力を提供します。

システムバスはメモリ空間全体を 17 個の領域に分割し、パーミッショングループを介してイニシエータによる各ターゲットへのアクセスを許可します。各イニシエータには 4 通りのパーミッショングループ (0 ~ 3) を割り当てる事ができます。各パーミッショングループは互いに独立しており、1 つの領域に対して排他的にアクセスする事も、他のグループとアクセス権を共有する事もできます。

CFGPG レジスタ (41.0「その他の特殊機能」内のレジスタ 41-12 参照) により、ブート ファームウェアはシステムバス上で要求を生成可能な各イニシエータにパーミッショングループを割り当てる事ができます。

利用可能なターゲットとそれらの領域および保護を割り当てるための制御レジスタを表 4-8 に示します。

アクセス パーミッショングループと領域の設定および制御にはレジスタ 4-2 ~ レジスタ 4-13 を使います。

これらのレジスタを変更するには、ハードウェアでレジスタのロックを解除する必要があります。レジスタのロックは PGLOCK コンフィグレーション ビット (CFGCON<11>) で制御します。PGLOCK ビットをセットすると制御レジスタへの書き込みが禁止され、クリアすると書き込みが可能になります。

PGLOCK ビットをセットまたはクリアするには、ロック解除シーケンスを実行する必要があります。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

表 4-8: システムバス ターゲットと対応する保護レジスタ

ターゲット 保護番号	ターゲット (Note 5 参照)	SBTxREGy レジスタ (Note 7 参照)							SBTxRDy レジスタ		SBTxWRy レジスタ	
		名称	領域ベース アドレス (BASE<21:0>) (Note 2 参照)	開始物理 アドレス	領域サイズ (SIZE<4:0>) (Note 3 参照)	領域サイズ	優先度 ビット (PRI)	優先度	名称	読み出し パーミッション (GROUP3、 GROUP2、 GROUP1、 GROUP0)	名称	書き込み パーミッション (GROUP3、 GROUP2、 GROUP1、 GROUP0)
0	システムバス	SBT0REG0	R	0x1F8F0000	R	64 KB	—	0	SBT0RD0	0,1,1,1	SBT0WR0	0,1,1,1
		SBT0REG1	R	0x1F8F8000	R	32 KB	—	3	SBT0RD1	0,0,0,1	SBT0WR1	0,0,0,1
1	フラッシュメモリ (6): プログラム フラッシュ ブートフラッシュ プリフェッチ	SBT1REG0	R	0x1D000000	R <sup>(4)</sup>	R <sup>(4)</sup>	—	0	SBT1RD0	0,0,0,0	SBT1WR0	0,0,0,0
		SBT1REG2	R	0x1F8E0000	R	4 KB	1	2	SBT1RD2	R/W <sup>(1)</sup>	SBT1WR2	R/W <sup>(1)</sup>
		SBT1REG3	R/W	R/W	R/W	R/W	1	2	SBT1RD3	0,0,0,0	SBT1WR3	0,0,0,0
		SBT1REG4	R/W	R/W	R/W	R/W	1	2	SBT1RD4	0,0,0,0	SBT1WR4	0,0,0,0
		SBT1REG5	R/W	R/W	R/W	R/W	1	2	SBT1RD5	0,0,0,0	SBT1WR5	0,0,0,0
		SBT1REG6	R/W	R/W	R/W	R/W	1	2	SBT1RD6	0,0,0,0	SBT1WR6	0,0,0,0
		SBT1REG7	R/W	R/W	R/W	R/W	0	1	SBT1RD7	0,0,0,0	SBT1WR7	0,0,0,0
		SBT1REG8	R/W	R/W	R/W	R/W	0	1	SBT1RD8	0,0,0,0	SBT1WR8	0,0,0,0
2	RAM バンク 1 メモリ	SBT2REG0	R	0	R <sup>(4)</sup>	R <sup>(4)</sup>	—	0	SBT2RD0	R/W <sup>(1)</sup>	SBT2WR0	R/W <sup>(1)</sup>
		SBT2REG1	R/W	R/W	R/W	R/W	—	3	SBT2RD1	R/W <sup>(1)</sup>	SBT2WR1	R/W <sup>(1)</sup>
		SBT2REG2	R/W	R/W	R/W	R/W	0	1	SBT2RD2	R/W <sup>(1)</sup>	SBT2WR2	R/W <sup>(1)</sup>
3	RAM バンク 2 メモリ	SBT3REG0	R <sup>(4)</sup>	R <sup>(4)</sup>	R <sup>(4)</sup>	R <sup>(4)</sup>	—	0	SBT3RD0	R/W <sup>(1)</sup>	SBT3WR0	R/W <sup>(1)</sup>
		SBT3REG1	R/W	R/W	R/W	R/W	—	3	SBT3RD1	R/W <sup>(1)</sup>	SBT3WR1	R/W <sup>(1)</sup>
		SBT3REG2	R/W	R/W	R/W	R/W	0	1	SBT3RD2	R/W <sup>(1)</sup>	SBT3WR2	R/W <sup>(1)</sup>
4	DDR2 を介する外部メモリと DDR2 ターゲット 0	SBT4REG0	R	0x08000000	R	R(4)	—	0	SBT4RD0	R/W <sup>(1)</sup>	SBT4WR0	R/W <sup>(1)</sup>
		SBT4REG1	R/W	R/W	R/W	R/W	—	3	SBT4RD1	R/W <sup>(1)</sup>	SBT4WR1	R/W <sup>(1)</sup>
		SBT4REG2	R/W	R/W	R/W	R/W	1	2	SBT4RD2	R/W <sup>(1)</sup>	SBT4WR2	R/W <sup>(1)</sup>
		SBT4REG3	R/W	R/W	R/W	R/W	1	2	SBT4RD3	R/W <sup>(1)</sup>	SBT4WR3	R/W <sup>(1)</sup>
		SBT4REG4	R/W	R/W	R/W	R/W	1	2	SBT4RD4	R/W <sup>(1)</sup>	SBT4WR4	R/W <sup>(1)</sup>
5	DDR2 を介する外部メモリと DDR2 ターゲット 1/2	SBT5REG0	R	0x08000000	R	R(4)	—	0	SBT5RD0	R/W <sup>(1)</sup>	SBT5WR0	R/W <sup>(1)</sup>
		SBT5REG1	R/W	R/W	R/W	R/W	—	3	SBT5RD1	R/W <sup>(1)</sup>	SBT5WR1	R/W <sup>(1)</sup>
		SBT5REG2	R/W	R/W	R/W	R/W	1	2	SBT5RD2	R/W <sup>(1)</sup>	SBT5WR2	R/W <sup>(1)</sup>
		SBT5REG3	R/W	R/W	R/W	R/W	1	2	SBT5RD3	R/W <sup>(1)</sup>	SBT5WR3	R/W <sup>(1)</sup>
		SBT5REG4	R/W	R/W	R/W	R/W	1	2	SBT5RD4	R/W <sup>(1)</sup>	SBT5WR4	R/W <sup>(1)</sup>

- 凡例: R = 読み出し専用 R/W = 読み書き可能 レジスタ名内の「x」= 0-13 レジスタ名内の「y」= 0-8
- Note 1: これらのビットのリセット値は、それぞれ「0」、「1」、「1」、「1」です。
- 2: BASE<21:0> ビットは、対応する物理アドレスを右へ 10 ビットシフトした値に設定する必要があります。読み出し専用ビットの場合、この値はリセット時にハードウェアによって設定されます。
- 3: SIZE<4:0> ビットは、以下の式に基づいて、対応する領域サイズに設定する必要があります。  
領域サイズ =  $2^{(SIZE-1)} \times 1024$  (byte)  
読み出し専用ビットの場合、この値はリセット時にハードウェアによって設定されます。
- 4: 本デバイスのメモリサイズと開始アドレスはデバイス メモリマップ (図 4-1) を参照してください。
- 5: 各ターゲットのメモリサイズと開始アドレスは表 4-2 を参照してください。
- 6: このターゲットの場合、SBTxREG1 SFR は予約済みであるため、表には記載していません。
- 7: SBTxREGy、SBTxRDy、SBTxWRy レジスタの「x」はターゲット保護番号を表します (実際のターゲット番号ではありません)。例えば SQI の場合、「x」= 13 ですが、実際のターゲット番号は 11 です。

表 4-8: システムバス ターゲットと対応する保護レジスタ (続き)

ターゲット 保護番号	ターゲット (Note 5 参照)	SBTxREGy レジスタ (Note 7 参照)							SBTxRDy レジスタ		SBTxWRy レジスタ	
		名称	領域ベース アドレス (BASE<21:0> (Note 2 参照))	開始物理 アドレス	領域サイズ (SIZE<4:0> (Note 3 参照))	領域サイズ	優先度 ビット (PRI)	優先度	名称	読み出し パーミッション (GROUP3、 GROUP2、 GROUP1、 GROUP0)	名称	書き込み パーミッション (GROUP3、 GROUP2、 GROUP1、 GROUP0)
6	EEBI を介する外部メモリと EBI モジュール <sup>(6)</sup>	SBT6REG0	R	0x20000000	R	64 MB	—	0	SBT6RD0	R/W <sup>(1)</sup>	SBT6WR0	R/W <sup>(1)</sup>
		SBT6REG2	R	0x1F8EC000	R	4 KB	0	1	SBT6RD2	R/W <sup>(1)</sup>	SBT6WR2	R/W <sup>(1)</sup>
7	システム コントローラ	SBT7REG0	R	0x1F800000	R	—	—	0	SBT7RD0	R/W <sup>(1)</sup>	SBT7WR0	R/W <sup>(1)</sup>
	フラッシュ コントローラ	SBT7REG1	R/W	R/W	R/W	R/W	—	3	SBT7RD1	R/W <sup>(1)</sup>	SBT7WR1	R/W <sup>(1)</sup>
	DMT/WDT CVREF PPS 入力 PPS 出力 割り込み DMA	SBT7REG2	R/W	R/W	R/W	R/W	0	1	SBT7RD2	R/W <sup>(1)</sup>	SBT7WR2	R/W <sup>(1)</sup>
8	SPI1-SPI6	SBT8REG0	R	0x1F820000	R	64 KB	—	0	SBT8RD0	R/W <sup>(1)</sup>	SBT8WR0	R/W <sup>(1)</sup>
	I2C1-I2C5 UART1-UART6 PMP	SBT8REG1	R/W	R/W	R/W	R/W	—	3	SBT8RD1	R/W <sup>(1)</sup>	SBT8WR1	R/W <sup>(1)</sup>
9	Timer1-Timer9	SBT9REG0	R	0x1F840000	R	64 KB	—	0	SBT9RD0	R/W <sup>(1)</sup>	SBT9WR0	R/W <sup>(1)</sup>
	IC1-IC9 OC1-OC9 ADC コンパレータ 1 コンパレータ 2	SBT9REG1	R/W	R/W	R/W	R/W	—	3	SBT9RD1	R/W <sup>(1)</sup>	SBT9WR1	R/W <sup>(1)</sup>
10	PORTA-PORTK	SBT10REG0	R	0x1F860000	R	64 KB	—	0	SBT10RD0	R/W <sup>(1)</sup>	SBT10WR0	R/W <sup>(1)</sup>
		SBT10REG1	R/W	R/W	R/W	R/W	—	3	SBT10RD1	R/W <sup>(1)</sup>	SBT10WR1	R/W <sup>(1)</sup>
11	CAN1	SBT11REG0	R	0x1F880000	R	64 KB	—	0	SBT11RD0	R/W <sup>(1)</sup>	SBT11WR0	R/W <sup>(1)</sup>
	CAN2 Ethernet	SBT11REG1	R/W	R/W	R/W	R/W	—	3	SBT11RD1	R/W <sup>(1)</sup>	SBT11WR1	R/W <sup>(1)</sup>
12	GLCD	SBT12REG0	R	0x1F8EA000	R	4 KB	—	0	SBT12RD0	R/W <sup>(1)</sup>	SBT12WR0	R/W <sup>(1)</sup>
	GPU		R	0x1F8EB000	R	4 KB	—	0		R/W <sup>(1)</sup>		R/W <sup>(1)</sup>
	DDR2PHY		R	0x1F8E9000	R	4 KB	—	0		R/W <sup>(1)</sup>		R/W <sup>(1)</sup>
	DDR2SFR		R	0x1F8E8000	R	4 KB	—	0		R/W <sup>(1)</sup>		R/W <sup>(1)</sup>
13	SQ11 を介する外部メモリと SQ11	SBT13REG0	R	0x30000000	R	64 MB	—	0	SBT13RD0	R/W <sup>(1)</sup>	SBT13WR0	R/W <sup>(1)</sup>
		SBT13REG1	R	0x1F8E2000	R	4 KB	—	3	SBT13RD1	R/W <sup>(1)</sup>	SBT13WR1	R/W <sup>(1)</sup>

凡例: R = 読み出し専用 R/W = 読み書き可能 レジスタ名内の「x」= 0-13 レジスタ名内の「y」= 0-8

- Note
- これらのビットのリセット値は、それぞれ「0」、「1」、「1」、「1」です。
  - BASE<21:0> ビットは、対応する物理アドレスを右へ 10 ビットシフトした値に設定する必要があります。読み出し専用ビットの場合、この値はリセット時にハードウェアによって設定されます。
  - SIZE<4:0> ビットは、以下の式に基づいて、対応する領域サイズに設定する必要があります。  
領域サイズ =  $2^{(\text{SIZE}-1)} \times 1024$  (byte)  
読み出し専用ビットの場合、この値はリセット時にハードウェアによって設定されます。
  - 本デバイスのメモリサイズと開始アドレスはデバイス メモリマップ (図 4-1) を参照してください。
  - 各ターゲットのメモリサイズと開始アドレスは表 4-2 を参照してください。
  - このターゲットの場合、SBTxREG1 SFR は予約済みであるため、表には記載していません。
  - SBTxREGy、SBTxRDy、SBTxWRy レジスタの「x」はターゲット保護番号を表します (実際のターゲット番号ではありません)。例えば SQ1 の場合、「x」= 13 ですが、実際のターゲット番号は 11 です。

表 4-8: システムバス ターゲットと対応する保護レジスタ (続き)

ターゲット 保護番号	ターゲット (Note 5 参照)	SBTxREGy レジスタ (Note 7 参照)							SBTxRDy レジスタ		SBTxWRy レジスタ	
		名称	領域ベース アドレス (BASE<21:0>) (Note 2 参照)	開始物理 アドレス	領域サイズ (SIZE<4:0>) (Note 3 参照)	領域サイズ	優先度 ビット (PRI)	優先度	名称	読み出し パーミッション (GROUP3、 GROUP2、 GROUP1、 GROUP0)	名称	書き込み パーミッション (GROUP3、 GROUP2、 GROUP1、 GROUP0)
14	DSCTRL RTCC	SBT14REG0	R	0x1F8C0000	R	4 KB	—	0	SBT14RD0	R/W <sup>(1)</sup>	SBT14WR0	R/W <sup>(1)</sup>
		SBT14REG1	R/W	R/W	R/W	R/W	—	3	SBT14RD1	R/W <sup>(1)</sup>	SBT14WR1	R/W <sup>(1)</sup>
15	USB	SBT15REG0	R	0x1F8E0000	R	4 KB	—	0	SBT15RD0	R/W <sup>(1)</sup>	SBT15WR0	R/W <sup>(1)</sup>
	暗号エンジン		R	0x1F8E5000	R	4 KB	—	0		R/W <sup>(1)</sup>		R/W <sup>(1)</sup>
	RNG		R	0x1F8E6000	R	4 KB	—	0		R/W <sup>(1)</sup>		R/W <sup>(1)</sup>
	SDHC		R	0x1F8EC000	R	4 KB	—	0		R/W <sup>(1)</sup>		R/W <sup>(1)</sup>
16	DDR2 を介する外部メモリと DDR2 ターゲット 3/4	SBT16REG0	R	0x08000000	R	R(4)	—	0	SBT16RD0	R/W <sup>(1)</sup>	SBT16WR0	R/W <sup>(1)</sup>
		SBT16REG1	R/W	R/W	R/W	R/W	—	3	SBT16RD1	R/W <sup>(1)</sup>	SBT16WR1	R/W <sup>(1)</sup>
		SBT16REG2	R/W	R/W	R/W	R/W	1	2	SBT16RD2	R/W <sup>(1)</sup>	SBT16WR2	R/W <sup>(1)</sup>
		SBT16REG3	R/W	R/W	R/W	R/W	1	2	SBT16RD3	R/W <sup>(1)</sup>	SBT16WR3	R/W <sup>(1)</sup>
		SBT16REG4	R/W	R/W	R/W	R/W	1	2	SBT16RD4	R/W <sup>(1)</sup>	SBT16WR4	R/W <sup>(1)</sup>

凡例: R = 読み出し専用 R/W = 読み書き可能 レジスタ名内の「x」= 0-13 レジスタ名内の「y」= 0-8

- Note
- これらのビットのリセット値は、それぞれ「0」、「1」、「1」、「1」です。
  - BASE<21:0> ビットは、対応する物理アドレスを右へ 10 ビットシフトした値に設定する必要があります。読み出し専用ビットの場合、この値はリセット時にハードウェアによって設定されます。
  - SIZE<4:0> ビットは、以下の式に基づいて、対応する領域サイズに設定する必要があります。  
領域サイズ =  $2^{(\text{SIZE} - 1)} \times 1024$  (byte)  
読み出し専用ビットの場合、この値はリセット時にハードウェアによって設定されます。
  - 本デバイスのメモリサイズと開始アドレスはデバイス メモリマップ (図 4-1) を参照してください。
  - 各ターゲットのメモリサイズと開始アドレスは表 4-2 を参照してください。
  - このターゲットの場合、SBTxREG1 SFR は予約済みであるため、表には記載していません。
  - SBTxREGy、SBTxRDy、SBTxWRy レジスタの「x」はターゲット保護番号を表します (実際のターゲット番号ではありません)。例えば SQI の場合、「x」= 13 ですが、実際のターゲット番号は 11 です。

表 4-9: システムバス違反フラグのレジスタマップ

仮アドレス (BFxx_#)	レジスタ名	ビットレンジ	Bit																リセット 値
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
8F_0510	SBFLAG0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	T0PGV0	T3PGV	T6PGV	T2PGV	T5PGV	T4PGV	T1PGV	0000
90_0510	SBFLAG1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	T0PGV1	T12PGV	T11PGV	T10PGV	T9PGV	T8PGV	T7PGV	0000
91_0510	SBFLAG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	T0PGV2	T15PGV	T14PGV	T13PGV	0000
92_0510	SBFLAG3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	T0PGV3	T16PGV	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

表 4-10: システムバス ターゲット保護グループ 0 (TOPGV0 ~ TOPGV3) のレジスタマップ

アドレス (BF8F_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
8020	SBT0ELOG1	31:16	MULTI	—	—	—	CODE<3:0>				—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>				REGION<3:0>				—	CMD<2:0>				0000			
8024	SBT0ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>			0000
8028	SBT0ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
8030	SBT0ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8038	SBT0ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8040	SBT0REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
8050	SBT0RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8058	SBT0WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8060	SBT0REG1	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
8070	SBT0RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8078	SBT0WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。



表 4-11: システムバス ターゲット保護グループ 1 のレジスタマップ

アドレス (BF8F_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
8420	SBT1ELOG1	31:16	MULTI	—	—	—	CODE<3:0>			—	—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>			REGION<3:0>			CMD<2:0>									0000	
8424	SBT1ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>		0000
8428	SBT1ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
8430	SBT1ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8438	SBT1ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8440	SBT1REG0	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>			PRI	—	SIZE<4:0>			—	—	—	—	—	—	—	—	xxxxx
8450	SBT1RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8458	SBT1WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8480	SBT1REG2	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>			PRI	—	SIZE<4:0>			—	—	—	—	—	—	—	—	xxxxx
8490	SBT1RD2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8498	SBT1WR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
84A0	SBT1REG3	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>			PRI	—	SIZE<4:0>			—	—	—	—	—	—	—	—	xxxxx
84B0	SBT1RD3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
84B8	SBT1WR3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
84C0	SBT1REG4	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>			PRI	—	SIZE<4:0>			—	—	—	—	—	—	—	—	xxxxx
84D0	SBT1RD4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
84D8	SBT1WR4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-11: システムバス ターゲット保護グループ1 のレジスタマップ ( 続き )

仮アドレ ス (BF0E_#)	レジスタ名	ビットレンジ	Bit														リセット値		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
84E0	SBT1REG5	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>					PRI	—	SIZE<4:0>					—	—	—	xxxxx	
84F0	SBT1RD5	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
84F8	SBT1WR5	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8500	SBT1REG6	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>					PRI	—	SIZE<4:0>					—	—	—	xxxxx	
8510	SBT1RD6	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8518	SBT1WR6	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8520	SBT1REG7	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>					PRI	—	SIZE<4:0>					—	—	—	xxxxx	
8530	SBT1RD7	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8538	SBT1WR7	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8540	SBT1REG8	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>					PRI	—	SIZE<4:0>					—	—	—	xxxxx	
8550	SBT1RD8	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8558	SBT1WR8	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx

凡例: x = リセット時に未知の値、— = 未実装 ( 「0」 として読み出し )、リセット値は 16 進表記  
 Note: 「xxxxx」 として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-12: システムバス ターゲット保護グループ 2 のレジスタマップ

アドレス (BF8F_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
8820	SBT2ELOG1	31:16	MULTI	—	—	—	—	CODE<3:0>				—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>				REGION<3:0>				—	CMD<2:0>				0000			
8824	SBT2ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>				0000
8828	SBT2ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
8830	SBT2ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000	
8838	SBT2ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000	
8840	SBT2REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
8850	SBT2RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8858	SBT2WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8860	SBT2REG1	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
8870	SBT2RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8878	SBT2WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8880	SBT2REG2	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
8890	SBT2RD2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8898	SBT2WR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-13: システムバス ターゲット保護グループ 3 のレジスタマップ

アドレス (BF8F_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
8C20	SBT3ELOG1	31:16	MULTI	—	—	—	CODE<3:0>				—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>							REGION<3:0>				—	CMD<2:0>				0000
8C24	SBT3ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>			
8C28	SBT3ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
8C30	SBT3ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8C38	SBT3ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8C40	SBT3REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	—	xxxx
8C50	SBT3RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8C58	SBT3WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8C60	SBT3REG1	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	—	xxxx
8C70	SBT3RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8C78	SBT3WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8C80	SBT3REG2	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	—	xxxx
8C90	SBT3RD2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8C98	SBT3WR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-14: システムバス ターゲット保護グループ 4 のレジスタマップ

アドレス (BF8F_#)	レジスタ名	ビットレンジ	Bit														リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
9020	SBT4ELOG1	31:16	MULTI	—	—	CODE<3:0>				—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>				REGION<3:0>				—	CMD<2:0>				0000		
9024	SBT4ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>			
9028	SBT4ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
9030	SBT4ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
9038	SBT4ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
9040	SBT4REG0	31:16	BASE<21:6>														xxxxx	
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxxx	
9050	SBT4RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
9058	SBT4WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
9060	SBT4REG1	31:16	BASE<21:6>														xxxxx	
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxxx	
9070	SBT4RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
9078	SBT4WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
9080	SBT4REG2	31:16	BASE<21:6>														xxxxx	
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxxx	
9090	SBT4RD2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
9098	SBT4WR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
90A0	SBT4REG3	31:16	BASE<21:6>														xxxxx	
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxxx	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-14: システムバス ターゲット保護グループ 4 のレジスタマップ ( 続き )

仮アドレス (BF8E_#)	レジスタ名	ビットレンジ	Bit														リセット 値		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
90B0	SBT4RD3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
90B8	SBT4WR3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
90C0	SBT4REG4	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>					PRI	—	SIZE<4:0>					—	—	—	xxxx	
90D0	SBT4RD4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
90D8	SBT4WR4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

**凡例:** x = リセット時に未知の値、— = 未実装 ( 「0」 として読み出し )、リセット値は 16 進表記  
**Note:** 「xxxx」 として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-15: システムバス ターゲット保護グループ 5 のレジスタマップ

アドレス (BF8F_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
9420	SBT5ELOG1	31:16	MULTI	—	—	—	—	CODE<3:0>				—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>				REGION<3:0>				—	CMD<2:0>				0000			
9424	SBT5ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>				0000
9428	SBT5ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
9430	SBT5ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000	
9438	SBT5ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000	
9440	SBT5REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
9450	SBT5RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx	
9458	SBT5WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx	
9460	SBT5REG1	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
9470	SBT5RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx	
9478	SBT5WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx	
9480	SBT5REG2	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
9490	SBT5RD2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx	
9498	SBT5WR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-15: システムバス ターゲット保護グループ 5 のレジスタマップ ( 続き )

仮アドレ ス (BF8E_#)	レジスタ名	ビットレンジ	Bit														リセット値	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
94A0	SBT5REG3	31:16	BASE<21:6>														xxxx	
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	xxxx
94B0	SBT5RD3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
94B8	SBT5WR3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
94C0	SBT5REG4	31:16	BASE<21:6>														xxxx	
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	xxxx
94D0	SBT5RD4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
94D8	SBT5WR4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

凡例: x = リセット時に未知の値、— = 未実装 ( 「0」 として読み出し )、リセット値は 16 進表記

Note: 「xxxx」 として示されたリセット値の実際の値は表 4-8 参照してください。



表 4-16: システムバス ターゲット保護グループ 6 のレジスタマップ

アドレス (BF8F_#)	レジスタ名	ビットレンジ	Bit														リセット値		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
9820	SBT6ELOG1	31:16	MULTI	—	—	—	CODE<3:0>				—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>				REGION<3:0>				—	CMD<2:0>				0000			
9824	SBT6ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>			0000
9828	SBT6ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
9830	SBT6ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
9838	SBT6ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
9840	SBT6REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
9850	SBT6RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
9858	SBT6WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
9860	SBT6REG1	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
9870	SBT6RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
9878	SBT6WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-17: システムバス ターゲット保護グループ7 のレジスタマップ

アドレス (BF90_#)	レジスタ名	ビットレンジ	Bit														リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
8420	SBT7ELOG1	31:16	MULTI	—	—	—	CODE<3:0>			—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>						REGION<3:0>				—	CMD<2:0>			0000	
8424	SBT7ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>		
8428	SBT7ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
8430	SBT7ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8438	SBT7ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8440	SBT7REG0	31:16	BASE<21:6>														xxxx	
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	xxxx
8450	SBT7RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
8458	SBT7WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
8460	SBT7REG1	31:16	BASE<21:6>														xxxx	
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	xxxx
8470	SBT7RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
8478	SBT7WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
8480	SBT7REG2	31:16	BASE<21:6>														xxxx	
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	xxxx
8490	SBT7RD2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
8498	SBT7WR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-18: システムバス ターゲット保護グループ 8 のレジスタマップ

アドレス (BF90_#)	レジスタ名	ビットレンジ	Bit														リセット値		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
8820	SBT8ELOG1	31:16	MULTI	—	—	—	CODE<3:0>				—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>				REGION<3:0>				—	CMD<2:0>				0000			
8824	SBT8ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>			0000
8828	SBT8ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
8830	SBT8ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8838	SBT8ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8840	SBT8REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
8850	SBT8RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8858	SBT8WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8860	SBT8REG1	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
8870	SBT8RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8878	SBT8WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-19: システムバス ターゲット保護グループ 9 のレジスタマップ

アドレス (BF90_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
8C20	SBT9ELOG1	31:16	MULTI	—	—	—	CODE<3:0>			—	—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>						REGION<3:0>				—	CMD<2:0>			0000		
8C24	SBT9ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>			0000
8C28	SBT9ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
8C30	SBT9ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8C38	SBT9ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8C40	SBT9REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	—	xxxx
8C50	SBT9RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8C58	SBT9WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8C60	SBT9REG1	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	—	xxxx
8C70	SBT9RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
8C78	SBT9WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-20: システムバス ターゲット保護グループ 10 のレジスタマップ

アドレス (BF90_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
9020	SBT10ELOG1	31:16	MULTI	—	—	—	CODE<3:0>				—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>				REGION<3:0>				—	CMD<2:0>				0000			
9024	SBT10ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>				0000
9028	SBT10ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
9030	SBT10ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000	
9038	SBT10ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000	
9040	SBT10REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
9050	SBT10RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
9058	SBT10WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
9060	SBT10REG1	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
9070	SBT10RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
9078	SBT10WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-21: システムバス ターゲット保護グループ 11 のレジスタマップ

アドレス (BF90_#)	レジスタ名	ビットレンジ	Bit														リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
9420	SBT11ELOG1	31:16	MULTI	—	—	—	CODE<3:0>			—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>						REGION<3:0>				—	CMD<2:0>			0000	
9424	SBT11ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>		
9428	SBT11ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
9430	SBT11ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
9438	SBT11ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
9440	SBT11REG0	31:16	BASE<21:6>														xxxx	
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	xxxx
9450	SBT11RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
9458	SBT11WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
9460	SBT11REG1	31:16	BASE<21:6>														xxxx	
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	xxxx
9470	SBT11RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
9478	SBT11WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-22: システムバス ターゲット保護グループ 12 のレジスタマップ

アドレス (BF90_#)	レジスタ名	ビットレンジ	Bit														全 リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
9820	SBT12ELOG1	31:16	MULTI	—	—	—	CODE<3:0>				—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>				REGION<3:0>				—	CMD<2:0>				0000			
9824	SBT12ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>			0000
9828	SBT12ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
9830	SBT12ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
9838	SBT12ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
9840	SBT12REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	xxxx		
9850	SBT12RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
9858	SBT12WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-23: システムバス ターゲット保護グループ 13 のレジスタマップ

アドレス (BF91_#)	レジスタ名	ビットレンジ	Bit														リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
8420	SBT13ELOG1	31:16	MULTI	—	—	—	CODE<3:0>			—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>						REGION<3:0>				—	CMD<2:0>			0000	
8424	SBT13ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>		
8428	SBT13ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
8430	SBT13ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8438	SBT13ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8440	SBT13REG0	31:16	BASE<21:6>														xxxx	
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	xxxx
8450	SBT13RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
8458	SBT13WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
8460	SBT13REG1	31:16	BASE<21:6>														xxxx	
		15:0	BASE<5:0>						PRI	—	SIZE<4:0>				—	—	—	xxxx
8470	SBT13RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
8478	SBT13WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。



表 4-24: システムバス ターゲット保護グループ 14 のレジスタマップ

アドレス (BF91_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
8820	SBT14ELOG1	31:16	MULTI	—	—	—	CODE<3:0>			—	—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>			REGION<3:0>			CMD<2:0>			0000							
8824	SBT14ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>		0000	
8828	SBT14ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
8830	SBT14ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8838	SBT14ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8840	SBT14REG0	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>			PRI	—	SIZE<4:0>			—	—	—	—	—	—	—	—	xxxxx
8850	SBT14RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8858	SBT14WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8860	SBT14REG1	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>			PRI	—	SIZE<4:0>			—	—	—	—	—	—	—	—	xxxxx
8870	SBT14RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
8878	SBT14WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-25: システムバス ターゲット保護グループ 15 のレジスタマップ

アドレス (BF91_#)	レジスタ名	ビットレンジ	Bit														リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
8C20	SBT15ELOG1	31:16	MULTI	—	—	—	CODE<3:0>			—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>				REGION<3:0>				—	CMD<2:0>				0000		
8C24	SBT15ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>			0000
8C28	SBT15ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
8C30	SBT15ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8C38	SBT15ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
8C40	SBT15REG0	31:16	BASE<21:6>														xxxxx	
		15:0	BASE<5:0>					PRI	—	SIZE<4:0>				—	—	—	—	xxxxx
8C50	SBT15RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
8C58	SBT15WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-26: システムバス ターゲット保護グループ 16 のレジスタマップ

アドレス (BF92_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
C420	SBT16ELOG1	31:16	MULTI	—	—	—	CODE<3:0>			—	—	—	—	—	—	—	—	—	0000
		15:0	INITID<7:0>			REGION<3:0>			CMD<2:0>			—	—	—	—	—	—	—	0000
C424	SBT16ELOG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>	0000
C428	SBT16ECON	31:16	—	—	—	—	—	—	ERRP	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
C430	SBT16ECLRS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
C438	SBT16ECLRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR	0000
C440	SBT16REG0	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	—	—	—
C450	SBT16RD0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
C458	SBT16WR0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
C460	SBT16REG1	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	—	—	—
C470	SBT16RD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
C478	SBT16WR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
C480	SBT16REG2	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	—	—	—
C490	SBT16RD2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
C498	SBT16WR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
C4A0	SBT16REG3	31:16	BASE<21:6>														xxxx		
		15:0	BASE<5:0>				PRI	—	SIZE<4:0>				—	—	—	—	—	—	—
C4B0	SBT16RD3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx
C4B8	SBT16WR3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note: 「xxxx」として示されたリセット値の実際の値は表 4-8 参照してください。

表 4-26: システムバス ターゲット保護グループ 16 のレジスタマップ ( 続き )

仮読アドレス (BF92_#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
C4C0	SBT16REG4	31:16	BASE<21:6>														xxxxx		
		15:0	BASE<5:0>					PRI	—	SIZE<4:0>					—	—	—	xxxxx	
C4D0	SBT16RD4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx
C4D8	SBT16WR4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0	xxxxx

凡例: x = リセット時に未知の値、— = 未実装 ( 「0」 として読み出し )、リセット値は 16 進表記  
 Note: 「xxxxx」 として示されたリセット値の実際の値は表 4-8 参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-2: SBFlag0: システムバス ステータスフラグ レジスタ 0

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	R/W-0 T0PGV0 <sup>(1)</sup>	R/W-0 T3PGV	R/W-0 T6PGV	R/W-0 T2PGV	R/W-0 T5PGV	R/W-0 T4PGV	R/W-0 T1PGV

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア

- bit 31-7 **未実装:** 「0」として読み出し
- bit 6 **T0PGV0:** ターゲット 0 (システムバス 0) パーミッション グループ違反ステータスビット<sup>(1)</sup>  
1 = ターゲット 0 (システムバス 0) はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 0 (システムバス 0) はパーミッション グループ (PG) 違反を報告していない
- bit 5 **T3PGV:** ターゲット 3 (システムバンク 2) パーミッション グループ違反ステータスビット  
1 = ターゲット 3 はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 3 は PG 違反を報告していない
- bit 4 **T6PGV:** ターゲット 6 (EBI) パーミッション グループ違反ステータスビット  
1 = ターゲット 6 はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 6 は PG 違反を報告していない
- bit 3 **T2PGV:** ターゲット 2 (システムバンク 1) パーミッション グループ違反ステータスビット  
1 = ターゲット 2 はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 2 は PG 違反を報告していない
- bit 2 **T5GV:** ターゲット 5 (DDR2 ターゲット 1/2) パーミッション グループ違反ステータスビット  
1 = ターゲット 5 はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 5 は PG 違反を報告していない
- bit 1 **T4PGV:** ターゲット 4 (DDR2 ターゲット 0) パーミッション グループ違反ステータスビット  
1 = ターゲット 4 はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 4 は PG 違反を報告していない
- bit 0 **T1PGV:** ターゲット 1 (フラッシュメモリ) パーミッション グループ違反ステータスビット  
1 = ターゲット 1 はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 1 は PG 違反を報告していない

**Note 1:** システムバス 0 は内部サブシステム エレメントを表し、一般システムバス違反として扱う必要があります。

**Note:** 全てのエラーは、違反報告元 (SBTxELOG1、SBTxELOG2、SBTxECLRS、SBTxECLRM レジスタのいずれか) でクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-3: SBFlag1: システムバス ステータスフラグ レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	T0PGV1 <sup>(1)</sup>	T12PGV <sup>(2)</sup>	T11PGV	T10PGV	T9PGV	T8PGV	T7PGV

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット                      U = 未実装、「0」として読み出し  
 -n = POR 時の値                                  1 = ビットはセット                                  0 = ビットはクリア

bit 31-7 **未実装:** 「0」として読み出し

bit 6 **T0PGV1:** ターゲット 1 (システムバス 1) パーミッション グループ違反ステータスビット<sup>(1)</sup>

1 = ターゲット 0 (システムバス 1) はパーミッション グループ (PG) 違反を報告している  
 0 = ターゲット 0 (システムバス 1) はパーミッション グループ (PG) 違反を報告していない

bit 5 **T12PGV:** ターゲット 12 (GLCD、GPU、DDR2PHY、DDR2SFR) パーミッション グループ違反ステータスビット<sup>(2)</sup>

1 = ターゲット グループ 12 はパーミッション グループ (PG) 違反を報告している  
 0 = ターゲット グループ 12 は PG 違反を報告していない

bit 4 **T11PGV:** ターゲット 11 (PB5) パーミッション グループ違反ステータスビット

1 = ターゲット 11 はパーミッション グループ (PG) 違反を報告している  
 0 = ターゲット 11 は PG 違反を報告していない

bit 3 **T10PGV:** ターゲット 10 (PB4) パーミッション グループ違反ステータスビット

1 = ターゲット 10 はパーミッション グループ (PG) 違反を報告している  
 0 = ターゲット 10 は PG 違反を報告していない

bit 2 **T9PGV:** ターゲット 9 (PB3) パーミッション グループ違反ステータスビット

1 = ターゲット 9 はパーミッション グループ (PG) 違反を報告している  
 0 = ターゲット 9 は PG 違反を報告していない

bit 1 **T8PGV:** ターゲット 8 (PB2) パーミッション グループ違反ステータスビット

1 = ターゲット 8 はパーミッション グループ (PG) 違反を報告している  
 0 = ターゲット 8 は PG 違反を報告していない

bit 0 **T7PGV:** ターゲット 7 (PB1) パーミッション グループ違反ステータスビット

1 = ターゲット 7 はパーミッション グループ (PG) 違反を報告している  
 0 = ターゲット 7 は PG 違反を報告していない

**Note 1:** システムバス 1 は内部サブシステム エLEMENT を表し、一般システムバス違反として扱う必要があります。  
**Note 2:** このビットはターゲット 14 (GLCD)、18 (GPU)、20 (DDR2PHY)、21 (DDR2SFR) の違反を報告します。

**Note:** 全てのエラーは、違反報告元 (SBTxELOG1、SBTxELOG2、SBTxECLRS、SBTxECLRM レジスタのいずれか) でクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-4: SBFlag2: システムバス ステータスフラグ レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 T0PGV2 <sup>(1)</sup>	R/W-0 T15PGV <sup>(2)</sup>	R/W-0 T14PGV	R/W-0 T13PGV

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
-n = POR 時の値                              1 = ビットはセット              0 = ビットはクリア

bit 31-4 **未実装:** 「0」として読み出し

bit 3 **T0PGV2:** ターゲット 0 (システムバス 2) パーミッション グループ違反ステータスビット<sup>(1)</sup>

1 = ターゲット 0 (システムバス 2) はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 0 (システムバス 2) はパーミッション グループ (PG) 違反を報告していない

bit 2 **T15PGV:** ターゲット グループ 15 (USB、暗号エンジン、RNG、SDHC) パーミッション グループ違反ステータスビット<sup>(2)</sup>

1 = ターゲット グループ 15 はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット グループ 15 は PG 違反を報告していない

bit 1 **T14PGV:** ターゲット 14 (PB6) パーミッション グループ違反ステータスビット

1 = ターゲット 14 はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 14 は PG 違反を報告していない

bit 0 **T13PGV:** ターゲット 13 (SQI) パーミッション グループ違反ステータスビット

1 = ターゲット 13 はパーミッション グループ (PG) 違反を報告している  
0 = ターゲット 13 は PG 違反を報告していない

**Note 1:** システムバス 2 は内部サブシステム エLEMENT を表し、一般システムバス違反として扱う必要があります。

**2:** このビットはターゲット 10 (USB)、12 (暗号エンジン)、13 (RNG)、19 (SDHC) の違反を報告します。

**Note:** 全てのエラーは、違反報告元 (SBTxELOG1、SBTxELOG2、SBTxECLRS、SBTxECLRM レジスタのいずれか) でクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-5: SBFlag3: システムバス ステータスフラグ レジスタ 3

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	T0PGV3 <sup>(1)</sup>	T16PGV

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア

bit 31-2 **未実装:** 「0」として読み出し

bit 1 **T0PGV3:** ターゲット 0 (システムバス 3) パーミッショングループ違反ステータスビット<sup>(1)</sup>

1 = ターゲット 0 (システムバス 3) はパーミッショングループ (PG) 違反を報告している  
 0 = ターゲット 0 (システムバス 3) はパーミッショングループ (PG) 違反を報告していない

bit 0 **T16PGV:** ターゲット 16 (DDR2 ターゲット 3/4) パーミッショングループ違反ステータスビット

1 = ターゲット 16 はパーミッショングループ (PG) 違反を報告している  
 0 = ターゲット 16 は PG 違反を報告していない

**Note 1:** システムバス 3 は内部サブシステム エLEMENT を表し、一般システムバス違反として扱う必要があります。

**Note:** 全てのエラーは、違反報告元 (SBTxELOG1、SBTxELOG2、SBTxECLRS、SBTxECLRM レジスタのいずれか) でクリアされます。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-6: SBTxELOG1: システムバス ターゲット「x」エラーログ レジスタ 1 (x = 0 ~ 13)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0, C	U-0	U-0	U-0	R/W-0, C	R/W-0, C	R/W-0, C	R/W-0, C
	MULTI	—	—	—	CODE<3:0>			
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	INITID<7:0>							
7:0	R-0	R-0	R-0	R-0	U-0	R-0	R-0	R-0
	REGION<3:0>				—	CMD<2:0>		

**凡例:**

C = クリア可能ビット  
R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値                1 = ビットはセット      0 = ビットはクリア

bit 31    **MULTI:** 複数パーミッション違反ステータスビット  
このビットは「1」を書き込む事によってクリアします。  
1 = 複数エラーを検出した  
0 = 複数エラーは検出していない

bit 30-28 **未実装:** 「0」として読み出し

bit 27-24 **CODE<3:0>:** エラーコード ビット  
このビットは検出されたエラーのタイプを示します。これらのビットは「1」を書き込む事によってクリアします。  
1111 = 予約済み  
1101 = 予約済み  
.  
.  
0011 = パーミッション違反  
0010 = 予約済み  
0001 = 予約済み  
0000 = エラーなし

bit 23-16 **未実装:** 「0」として読み出し

**Note:** 利用可能なターゲットとそれらの説明は表 4-8 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-6: SBTxELOG1: システムバス ターゲット 「x」 エラーログ レジスタ 1 (x = 0 ~ 13) (続き)

bit 15-8 **INITID<7:0>**: アクセス要求イニシエータ ID ビット

11111111 = 予約済み  
.  
.  
00001111 = 予約済み  
00001110 = SDHC  
00001101 = GPU  
00001100 = GLCD  
00001011 = 暗号エンジン  
00001010 = フラッシュ コントローラ  
00001001 = SQ1  
00001000 = CAN2  
00000111 = CAN1  
00000110 = Ethernet 書き込み  
00000101 = Ethernet 読み出し  
00000100 = USB  
00000011 = DMA 書き込み  
00000010 = DMA 読み出し  
00000001 = CPU  
00000000 = 予約済み

bit 7-4 **REGION<3:0>**: ターゲット領域番号ビット

1111 ~ 0000 = パーミッション グループ違反を報告しているターゲットの領域

bit 3 **未実装**: 「0」 として読み出し

bit 2-0 **CMD<2:0>**: 要求トランザクション命令ビット

111 = 予約済み  
110 = 予約済み  
101 = 書き込み (非 POST 書き込み)  
100 = 予約済み  
011 = 読み出し (読み出し - 変更 - 書き込みトランザクションによって発生するロックされた読み出し)  
010 = 読み出し  
001 = 書き込み  
000 = アイドル

**Note:** 利用可能なターゲットとそれらの説明は表 4-8 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-7: SBTxELOG2: システムバス ターゲット「x」エラーログ レジスタ 2 (x = 0 ~ 13)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	R-0	R-0
	—	—	—	—	—	—	GROUP<1:0>	

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア

- bit 31-3 **未実装**: 「0」として読み出し  
 bit 1-0 **GROUP<1:0>**: 要求パーミッショングループビット  
     11 = グループ 3  
     10 = グループ 2  
     01 = グループ 1  
     00 = グループ 0

**Note:** 利用可能なターゲットとそれらの説明は表 4-8 を参照してください。

レジスタ 4-8: SBTxECON: システムバス ターゲット「x」エラー制御レジスタ (x = 0 ~ 13)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
	—	—	—	—	—	—	—	ERRP
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア

- bit 31-25 **未実装**: 「0」として読み出し  
 bit 24 **ERRP**: エラー制御ビット  
     1 = 保護グループ違反エラーを報告する  
     0 = 保護グループ違反エラーを報告しない  
 bit 23-0 **未実装**: 「0」として読み出し

**Note:** 利用可能なターゲットとそれらの説明は表 4-8 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-9: SBTxECLRS: システムバス ターゲット 「x」 単一エラークリア レジスタ (x = 0 ~ 13)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0
	—	—	—	—	—	—	—	CLEAR

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア

bit 31-1 **未実装:** 「0」として読み出し

bit 0 **CLEAR:** 読み出し時単一エラークリア ビット

このレジスタを読み出す事で、SBTxELOG1 と SBTxELOG2 で報告された単一のエラーをクリアします。

**Note:** 利用可能なターゲットとそれらの説明は表 4-8 を参照してください。

レジスタ 4-10: SBTxECLRM: システムバス ターゲット 「x」 複数エラークリア レジスタ (x = 0 ~ 13)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0
	—	—	—	—	—	—	—	CLEAR

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア

bit 31-1 **未実装:** 「0」として読み出し

bit 0 **CLEAR:** 読み出し時複数エラークリア ビット

このレジスタを読み出す事で、SBTxELOG1 と SBTxELOG2 で報告された複数のエラーをクリアします。

**Note:** 利用可能なターゲットとそれらの説明は表 4-8 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-11: SBTxREGy: システムバス ターゲット「x」領域「y」レジスタ (x = 0 ~ 13、y = 0 ~ 8)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W0	R/W-0	R/W0	R/W-0	R/W0	R/W-0	R/W0	R/W-0
BASE<21:14>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BASE<13:6>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	U-0
BASE<5:0>							PRI	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
SIZE<4:0>						—	—	—

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア

bit 31-10 **BASE<21:0>**: 領域ベースアドレス ビット

bit 9 **PRI**: 領域優先度ビット

1 = レベル 2  
 0 = レベル 1

bit 8 **未実装**: 「0」として読み出し

bit 7-3 **SIZE<4:0>**: 領域サイズビット

領域のパーミッションは、SIZE が非ゼロである場合にのみ有効です。

11111 = 領域サイズ =  $2^{(SIZE-1)} \times 1024$  (byte)

•  
 •  
 •

00001 = 領域サイズ =  $2^{(SIZE-1)} \times 1024$  (byte)

00000 = 領域は存在しない

bit 2-0 **未実装**: 「0」として読み出し

**Note 1:** 利用可能なターゲットとそれらの説明は表 4-8 を参照してください。

**2:** 一部のターゲット領域では、このレジスタ内の特定のビットはプリセット値を格納した読み出し専用ビットです。詳細は表 4-8 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-12: SBTxRDy: システムバス ターゲット「x」領域「y」読み出しパーミッション レジスタ (x = 0 ~ 13、y = 0 ~ 8)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-1	R/W-1	R/W-1
	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

## 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット            0 = ビットはクリア

- bit 31-4 **未実装:** 「0」として読み出し
- bit 3 **Group3:** グループ 3 読み出しパーミッション ビット  
 1 = グループ 3 に読み出しアクセス権を与える  
 0 = グループ 3 に読み出しアクセス権を与えない
- bit 2 **Group2:** グループ 2 読み出しパーミッション ビット  
 1 = グループ 2 に読み出しアクセス権を与える  
 0 = グループ 2 に読み出しアクセス権を与えない
- bit 1 **Group1:** グループ 1 読み出しパーミッション ビット  
 1 = グループ 1 に読み出しアクセス権を与える  
 0 = グループ 1 に読み出しアクセス権を与えない
- bit 0 **Group0:** グループ 0 読み出しパーミッション ビット  
 1 = グループ 0 に読み出しアクセス権を与える  
 0 = グループ 0 に読み出しアクセス権を与えない

- Note 1:** 利用可能なターゲットとそれらの説明は表 4-8 を参照してください。
- 2:** 一部のターゲット領域では、このレジスタ内の特定のビットはプリセット値を格納した読み出し専用ビットです。詳細は表 4-8 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 4-13: SBTxWRy: システムバス ターゲット 「x」 領域 「y」 書き込みパーミッション レジスタ (x = 0 ~ 13、y = 0 ~ 8)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-1	R/W-1	R/W-1
	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

## 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア

- bit 31-4 **未実装:** 「0」として読み出し
- bit 3 **Group3:** グループ 3 書き込みパーミッション ビット  
1 = グループ 3 に書き込みアクセス権を与える  
0 = グループ 3 に書き込みアクセス権を与えない
- bit 2 **Group2:** グループ 2 書き込みパーミッション ビット  
1 = グループ 2 に書き込みアクセス権を与える  
0 = グループ 2 に書き込みアクセス権を与えない
- bit 1 **Group1:** グループ 1 書き込みパーミッション ビット  
1 = グループ 1 に書き込みアクセス権を与える  
0 = グループ 1 に書き込みアクセス権を与えない
- bit 0 **Group0:** グループ 0 書き込みパーミッション ビット  
1 = グループ 0 に書き込みアクセス権を与える  
0 = グループ 0 に書き込みアクセス権を与えない

- Note 1:** 利用可能なターゲットとそれらの説明は表 4-8 を参照してください。
- 2:** 一部のターゲット領域では、このレジスタ内の特定のビットはプリセット値を格納した読み出し専用ビットです。詳細は表 4-8 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:



# PIC32MZ グラフィック (DA) ファミリ

## 5.0 フラッシュ プログラムメモリ

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 52. Flash Program Memory with Support for Live Update』(DS60001193) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

PIC32MZ DA は、ユーザコードの実行用にフラッシュプログラムメモリを内蔵しています。このメモリの特長は以下の通りです。

- 2つのフラッシュバンクによりライブアップデートをサポート
- デュアルブートのサポート
- プログラムフラッシュとブートフラッシュの書き込み保護
- ECC サポート

このメモリは以下の3通りの方法でプログラミングできます。

- 実行時自己プログラミング (RTSP)
- EJTAG プログラミング
- インサーキット シリアル プログラミング (ICSP™)

RTSP は、フラッシュメモリまたは RAM からソフトウェアで実行できます。RTSP については、『PIC32 Family Reference Manual, Section 52. Flash Program Memory with Support for Live Update』(DS60001193) を参照してください。

EJTAG プログラミングには、デバイスの EJTAG ポートと EJTAG 対応プログラマを使います。

ICSP プログラミングでは、シリアルデータ接続を使って RTSP よりも大幅に高速なプログラミングが可能です。

EJTAG と ICSP によるプログラミング法については、『PIC32 Flash Programming Specification』(DS60001145) で説明しています。この文書は Microchip 社のウェブサイト ([www.microchip.com](http://www.microchip.com)) からダウンロードできます。

**Note:** PIC32MZ DA のフラッシュ ページサイズは 16 KB (4K IW)、行サイズは 2 KB (512 IW) です。

## 5.1 フラッシュ制御レジスタ

表 5-1: フラッシュコントローラのレジスタマップ

仮想アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0600	NVMCON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	WR	WREN	WRERR	LVDERR	—	—	—	—	PFSWAP	BFSWAP	—	—	NVMOP<3:0>				0000
0610	NVMKEY	31:16	NVMKEY<31:0>																0000
		15:0																	0000
0620	NVMADDR <sup>(1)</sup>	31:16	NVMADDR<31:0>																0000
		15:0																	0000
0630	NVMDATA0	31:16	NVMDATA0<31:0>																0000
		15:0																	0000
0640	NVMDATA1	31:16	NVMDATA1<31:0>																0000
		15:0																	0000
0650	NVMDATA2	31:16	NVMDATA2<31:0>																0000
		15:0																	0000
0660	NVMDATA3	31:16	NVMDATA3<31:0>																0000
		15:0																	0000
0670	NVMSRC ADDR	31:16	NVMSRCADDR<31:0>																0000
		15:0																	0000
0680	NVMPWP <sup>(1)</sup>	31:16	PWPLOCK	—	—	—	—	—	—	—	PWP<23:16>							8000	
		15:0	PWP<15:0>																0000
0690	NVMBWP <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	LBWPLOCK	—	—	LBWP4	LBWP3	LBWP2	LBWP1	LBWP0	UBWPLOCK	—	—	UBWP4	UBWP3	UBWP2	UBWP1	UBWP0	9FDF
06A0	NVMCON2 <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	00xx
		15:0	—	—	—	—	—	—	—	—	SWAPLOCK<1:0>		—	—	—	—	—	—	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: このレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 5-1: NVMCON: プログラミング制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-0, HC WR <sup>(1)</sup>	R/W-0 WREN <sup>(1)</sup>	R-0, HS, HC WRERR <sup>(1)</sup>	R-0, HS, HC LVDERR <sup>(1)</sup>	U-0 —	U-0 —	U-0 —	U-0 —
7:0	R/W-0 PFSWAP <sup>(3)</sup>	R/W-0 BFSWAP <sup>(3,4)</sup>	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0
					NVMOP<3:0>			

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **WR:** 書き込み制御ビット<sup>(1)</sup>

このビットはソフトウェアでクリアできません。このビットをセットするには、WREN = 1 の状態でロック解除シーケンスを実行する必要があります。

1 = フラッシュ動作を開始する

0 = フラッシュ動作は完了した、または、フラッシュ動作中ではない

bit 14 **WREN:** 書き込みイネーブルビット<sup>(1)</sup>

1 = WR ビットと SWAP ビットへの書き込みを許可し、NVMOP<3:0> ビットへの書き込みを禁止する

0 = WR ビットと SWAP ビットへの書き込みを禁止し、NVMOP<3:0> ビットへの書き込みを許可する

bit 13 **WRERR:** 書き込みエラービット<sup>(1)</sup>

このビットをクリアするには、NVMOP<3:0> ビットを「0000」に設定してからフラッシュ動作を開始する必要があります。

1 = プログラミングまたは消去シーケンスは正常に完了しなかった

0 = プログラミングまたは消去シーケンスは正常に完了した

bit 12 **LVDERR:** 低電圧検出エラービット<sup>(1)</sup>

このビットをクリアするには、NVMOP<3:0> ビットを「0000」に設定してからフラッシュ動作を開始する必要があります。

1 = 低電圧を検出した (WRERR がセットされた場合、データは破損している可能性があります)

0 = 電圧レベルは適正 (プログラミングが可能)

bit 11-8 **未実装:** 「0」として読み出し

bit 7 **PFSWAP:** プログラム フラッシュバンク スワップ制御ビット<sup>(3)</sup>

1 = PFM バンク 2 を下位マップ領域にマッピングし、PFM バンク 1 を上位マップ領域にマッピングする

0 = PFM バンク 1 を下位マップ領域にマッピングし、PFM バンク 2 を上位マップ領域にマッピングする

**Note 1:** これらのビットは、パワーオン リセット (POR) によってのみリセットされ、その他のリセット要因には影響されません。

**2:** 動的フラッシュ ECC コンフィグレーション ビット (FECCCON<1:0> (DVCFG0<9:8>)) が「00」(ECC は常時有効) の場合、ワードプログラミングは機能しません (「動作なし」(NOP) と同じ扱い)。これ以外の FECCCON<1:0> ビット設定では、ワードはプログラミングされます。しかし、動的フラッシュ ECC が有効 (FECCCON<1:0> = 01) の場合、そのワード向けに ECC ビットは書き込まれず、DED エラーが発生します。ECC およびフラッシュプログラミングの詳細は『PIC32 Family Reference Manual, Section 52. Flash Program Memory with Support for Live Update』(DS60001193) を参照してください。

**3:** このビットの変更は、WREN ビット = 0 の時に NVMKEY ロック解除シーケンスを実行し、SWAPLOCK<1:0> ビット (NVMCON2<7:6>) が「0」にクリアされている場合にのみ可能です。

**4:** BFSWAP の値は、各ブートパネルでユーザが設定したシーケンス番号によって決まります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 5-1: NVMCON: プログラミング制御レジスタ (続き)

- bit 6 **BFSWAP**: ブートフラッシュバンク スワップ制御ビット<sup>(3,4)</sup>  
1 = ブートフラッシュバンク2を下位マップ領域にマッピングし、ブートフラッシュバンク1を上位マップ領域にマッピングする  
0 = ブートフラッシュバンク1を下位マップ領域にマッピングし、ブートフラッシュバンク2を上位マップ領域にマッピングする
- bit 5-4 **未実装**: 「0」として読み出し
- bit 3-0 **NVMOP<3:0>**: NVM 動作ビット  
これらのビットは WREN = 0 の時にのみ書き込み可能です。  
1111 = 予約済み  
.  
.  
1000 = 予約済み  
0111 = PFM消去動作: 全てのPFM領域を消去する(全てのページは非保護(PWP<23:0> = 0x000000)である必要がある)  
0110 = 上位PFM消去動作: 上位マップ領域のPFMだけを消去する(その領域内の全てのページは非保護である必要がある)  
0101 = 下位PFM消去動作: 下位マップ領域のPFMだけを消去する(その領域内の全てのページは非保護である必要がある)  
0100 = ページ消去動作: NVMADDR で選択したページを消去する(書き込み保護されていない場合)  
0011 = 行プログラミング動作: NVMADDR で選択した行をプログラミングする(書き込み保護されていない場合)  
0010 = クワッドワード(128ビット)プログラミング動作: NVMADDRで選択した128ビットのフラッシュワードをプログラミングする(書き込み保護されていない場合)  
0001 = ワードプログラミング動作: NVMADDRで選択したワードをプログラミングする(書き込み保護されていない場合)<sup>(2)</sup>  
0000 = 動作なし

**Note 1:** これらのビットは、パワーオン リセット (POR) によってのみリセットされ、その他のリセット要因には影響されません。

- 動的フラッシュ ECC コンフィグレーション ビット (FECCCON<1:0> (DVCFG0<9:8>)) が「00」(ECC は常時有効) の場合、ワードプログラミングは機能しません(「動作なし」(NOP)と同じ扱い)。これ以外の FECCCON<1:0> ビット設定では、ワードはプログラミングされます。しかし、動的フラッシュ ECC が有効 (FECCCON<1:0> = 01) の場合、そのワード向けに ECC ビットは書き込まれず、DED エラーが発生します。ECC およびフラッシュ プログラミングの詳細は『PIC32 Family Reference Manual, Section 52. Flash Program Memory with Support for Live Update』(DS60001193) を参照してください。
- このビットの変更は、WREN ビット = 0 の時に NVMKEY ロック解除シーケンスを実行し、SWAPLOCK<1:0> ビット (NVMCON2<7:6>) が「0」にクリアされている場合にのみ可能です。
- BFSWAP の値は、各ブートパネルでユーザが設定したシーケンス番号によって決まります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 5-2: NVMKEY: プログラミング ロック解除レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
NVMKEY<31:24>								
23:16	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
NVMKEY<23:16>								
15:8	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
NVMKEY<15:8>								
7:0	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
NVMKEY<7:0>								

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **NVMKEY<31:0>**: ロック解除レジスタビット  
 これらのビットは書き込み専用であり、読み出しは常に「0」を返します。

**Note:** このレジスタは、PFM への意図しない書き込みを防ぐためのロック解除シーケンス用に使います。

レジスタ 5-3: NVMADDR: フラッシュ アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMADDR<31:24> <sup>(1)</sup>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMADDR<23:16> <sup>(1)</sup>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMADDR<15:8> <sup>(1)</sup>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMADDR<7:0> <sup>(1)</sup>								

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **NVMADDR<31:0>**: フラッシュ アドレスビット<sup>(1)</sup>

NVMOP<3:0> で 選択した動作	フラッシュ アドレスビット (NVMADDR<31:0>)
ページ消去	アドレスは消去するページを指定する (NVMADDR<13:0> ビットは無視される)
行書き込み	アドレスはプログラミングする行を指定する (NVMADDR<11:0> ビットは無視される)
ワード書き込み	アドレスはプログラミングするワードを指定する (NVMADDR<1:0> ビットは無視される)
クワッドワード 書き込み	アドレスはプログラミングするクワッドワード (128 ビット) を指定する (NVMADDR<3:0> ビットは無視される)

**Note:** このレジスタ内のビットは、パワーオン リセット (POR) によってのみリセットされ、その他のリセット要因には影響されません。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 5-4: NVMDATAx: フラッシュデータ レジスタ (x = 0 ~ 3)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMDATA<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMDATA<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMDATA<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMDATA<7:0>								

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **NVMDATA<31:0>**: フラッシュデータ ビット

ワード書き込み:

NVMDATA0 の内容を、NVMAADDR で定義されたターゲット フラッシュ アドレスに書き込みます。

クワッドワード書き込み:

NVMDATA3:NVMDATA2:NVMDATA1:NVMDATA0 の内容を、NVMAADDR で定義されたターゲット フラッシュ アドレスに書き込みます。NVMDATA0 は最下位命令ワードを格納します。

**Note:** このレジスタ内のビットは、パワーオン リセット (POR) によってのみリセットされ、その他のリセット要因には影響されません。

## レジスタ 5-5: NVMSRCADDR: ソースデータ アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMSRCADDR<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMSRCADDR<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMSRCADDR<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMSRCADDR<7:0>								

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **NVMSRCADDR<31:0>**: ソースデータ アドレスビット

NVMOP<3:0> ビット (NVMCON<3:0>) を行書き込み向けに設定した場合、このビットはフラッシュに書き込むソースデータのシステム物理アドレスを指定します。

**Note:** このレジスタ内のビットはパワーオン リセット (POR) によってのみリセットされ、その他のリセット要因には影響されません。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 5-6: NVMPWP: PFM 書き込み保護レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-1	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	PWPULOCK	—	—	—	—	—	—	—
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PWP<23:16>							
15:8	R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
	PWP<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	PWP<7:0>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31 **PWPULOCK:** PFM ページ書き込み保護ロック解除ビット

1 = レジスタはロックされていない (変更可能)

0 = レジスタはロックされている (変更不可)

このビットはクリアのみ可能であり、全ての種類のリセットによってのみセットされます。

bit 30-24 **未実装:** 「0」として読み出し

bit 23-0 **PWP<23:0>:** PFM 書き込み保護 (ページ) アドレスビット

アドレスが 0x1Dxxxxxx 以下の物理メモリ領域を書き込み保護します。「xxxxxx」は PWP<23:0> により指定します。PWP<23:0> の値が「0」の場合、PFM の全域で書き込み保護は無効になります。このレジスタで指定したアドレスを含むページの全体と、そのページよりも下位アドレスのページが全て保護されます。

**Note:** このレジスタ内のビットに書き込むには、NVMKEY ロック解除シーケンスを実行する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 5-7: NVMBWP: フラッシュブート (ページ) 書き込み保護レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-1	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	LBWPULOCK	—	—	LBWP4 <sup>(1)</sup>	LBWP3 <sup>(1)</sup>	LBWP2 <sup>(1)</sup>	LBWP1 <sup>(1)</sup>	LBWP0 <sup>(1)</sup>
7:0	R/W-1	r-1	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	UBWPULOCK	—	—	UBWP4 <sup>(1)</sup>	UBWP3 <sup>(1)</sup>	UBWP2 <sup>(1)</sup>	UBWP1 <sup>(1)</sup>	UBWP0 <sup>(1)</sup>

<b>凡例:</b>	r = 予約済み
R = 読み出し可能ビット	W = 書き込み可能ビット
-n = POR 時の値	U = 未実装、「0」として読み出し
	1 = ビットはセット
	0 = ビットはクリア
	x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **LBWPULOCK:** 下位 BFM エイリアス書き込み保護ロック解除ビット

1 = LBWPx ビットはロックされていない (変更可能)

0 = LBWPx ビットはロックされている (変更不可)

このビットはクリアのみ可能であり、全ての種類のリセットによってのみセットされます。

bit 14-13 **未実装:** 「0」として読み出し

bit 12 **LBWP4:** 下位 BFM エイリアスページ 4 書き込み保護ビット<sup>(1)</sup>

1 = 物理アドレス 0x01FC10000 ~ 0x1FC13FFF の書き込み保護を有効にする

0 = 物理アドレス 0x01FC10000 ~ 0x1FC13FFF の書き込み保護を無効にする

bit 11 **LBWP3:** 下位 BFM エイリアスページ 3 書き込み保護ビット<sup>(1)</sup>

1 = 物理アドレス 0x01FC0C000 ~ 0x1FC0FFFF の書き込み保護を有効にする

0 = 物理アドレス 0x01FC0C000 ~ 0x1FC0FFFF の書き込み保護を無効にする

bit 10 **LBWP2:** 下位 BFM エイリアスページ 2 書き込み保護ビット<sup>(1)</sup>

1 = 物理アドレス 0x01FC08000 ~ 0x1FC0BFFF の書き込み保護を有効にする

0 = 物理アドレス 0x01FC08000 ~ 0x1FC0BFFF の書き込み保護を無効にする

bit 9 **LBWP1:** 下位 BFM エイリアスページ 1 書き込み保護ビット<sup>(1)</sup>

1 = 物理アドレス 0x01FC04000 ~ 0x1FC07FFF の書き込み保護を有効にする

0 = 物理アドレス 0x01FC04000 ~ 0x1FC07FFF の書き込み保護を無効にする

bit 8 **LBWP0:** 下位 BFM エイリアスページ 0 書き込み保護ビット<sup>(1)</sup>

1 = 物理アドレス 0x01FC00000 ~ 0x1FC03FFF の書き込み保護を有効にする

0 = 物理アドレス 0x01FC00000 ~ 0x1FC03FFF の書き込み保護を無効にする

bit 7 **UBWPULOCK:** 上位 BFM エイリアス書き込み保護ロック解除ビット

1 = UBWPx ビットはロックされていない (変更可能)

0 = UBWPx ビットはロックされている (変更不可)

このビットはクリアのみ可能であり、全ての種類のリセットによってのみセットされます。

bit 6 **予約済み:** このビットは開発ツール用に予約済みです。

bit 5 **未実装:** 「0」として読み出し

**Note 1:** これらのビットは、NVMKEY ロック解除シーケンスを実行し、対応するロックビット (LBWPULOCK または UBWPULOCK) がセットされている時のみ利用できます。

**Note:** このレジスタ内のビットに書き込むには、NVMKEY ロック解除シーケンスを実行する必要があります。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 5-7: NVMBWP: フラッシュブート (ページ) 書き込み保護レジスタ

- bit 4     **UBWP4:** 上位 BFM エイリアスページ 4 書き込み保護ビット<sup>(1)</sup>  
1 = 物理アドレス 0x01FC30000 ~ 0x1FC33FFF の書き込み保護を有効にする  
0 = 物理アドレス 0x01FC30000 ~ 0x1FC33FFF の書き込み保護を無効にする
- bit 3     **UBWP3:** 上位 BFM エイリアスページ 3 書き込み保護ビット<sup>(1)</sup>  
1 = 物理アドレス 0x01FC2C000 ~ 0x1FC2FFFF の書き込み保護を有効にする  
0 = 物理アドレス 0x01FC2C000 ~ 0x1FC2FFFF の書き込み保護を無効にする
- bit 2     **UBWP2:** 上位 BFM エイリアスページ 2 書き込み保護ビット<sup>(1)</sup>  
1 = 物理アドレス 0x01FC28000 ~ 0x1FC2BFFF の書き込み保護を有効にする  
0 = 物理アドレス 0x01FC28000 ~ 0x1FC2BFFF の書き込み保護を無効にする
- bit 1     **UBWP1:** 上位 BFM エイリアスページ 1 書き込み保護ビット<sup>(1)</sup>  
1 = 物理アドレス 0x01FC24000 ~ 0x1FC27FFF の書き込み保護を有効にする  
0 = 物理アドレス 0x01FC24000 ~ 0x1FC27FFF の書き込み保護を無効にする
- bit 0     **UBWP0:** 上位 BFM エイリアスページ 0 書き込み保護ビット<sup>(1)</sup>  
1 = 物理アドレス 0x01FC20000 ~ 0x1FC23FFF の書き込み保護を有効にする  
0 = 物理アドレス 0x01FC20000 ~ 0x1FC23FFF の書き込み保護を無効にする

**Note 1:** これらのビットは、NVMKEY ロック解除シーケンスを実行し、対応するロックビット (LBWPULOCK または UBWPULOCK) がセットされている時のみ利用できます。

**Note:** このレジスタ内のビットに書き込むには、NVMKEY ロック解除シーケンスを実行する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 5-8: NVMCON2: プログラミング制御レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
	SWAPLOCK<1:0> <sup>(1)</sup>		—	—	—	—	—	—

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-8 **未実装**: 「0」として読み出し

bit 7-6 **SWAPLOCK<1:0>**: PFM ページ書き込み保護ロック解除ビット<sup>(1)</sup>

11 = NVMCON レジスタの PFSWAP および BFSWP ビットと SWAPLOCK<1:0> は書き込み不可

10 = NVMCON レジスタの PFSWAP および BFSWP ビットは書き込み不可、SWAPLOCK<1:0> は書き込み可

01 = NVMCON レジスタの PFSWAP および BFSWP ビットは書き込み不可、SWAPLOCK<1:0> は書き込み可

00 = NVMCON レジスタの PFSWAP および BFSWP ビットと SWAPLOCK<1:0> は書き込み可

bit 5-0 **未実装**: 「0」として読み出し

**Note 1:** これらのビットは、NVMKEY ロック解除シーケンスを実行し、SWAPLOCK<1:0> ビット ≠ 11 の場合にのみ変更可能です。SWAPLOCK<1:0> ビット == 11 の場合、リセットだけがこれらのビットをクリアできます。

# PIC32MZ グラフィック (DA) ファミリ

## 6.0 リセット

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 07. Reset』(DS60001118) を参照してください。この文書は、Microchip 社ウェブサイトの PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

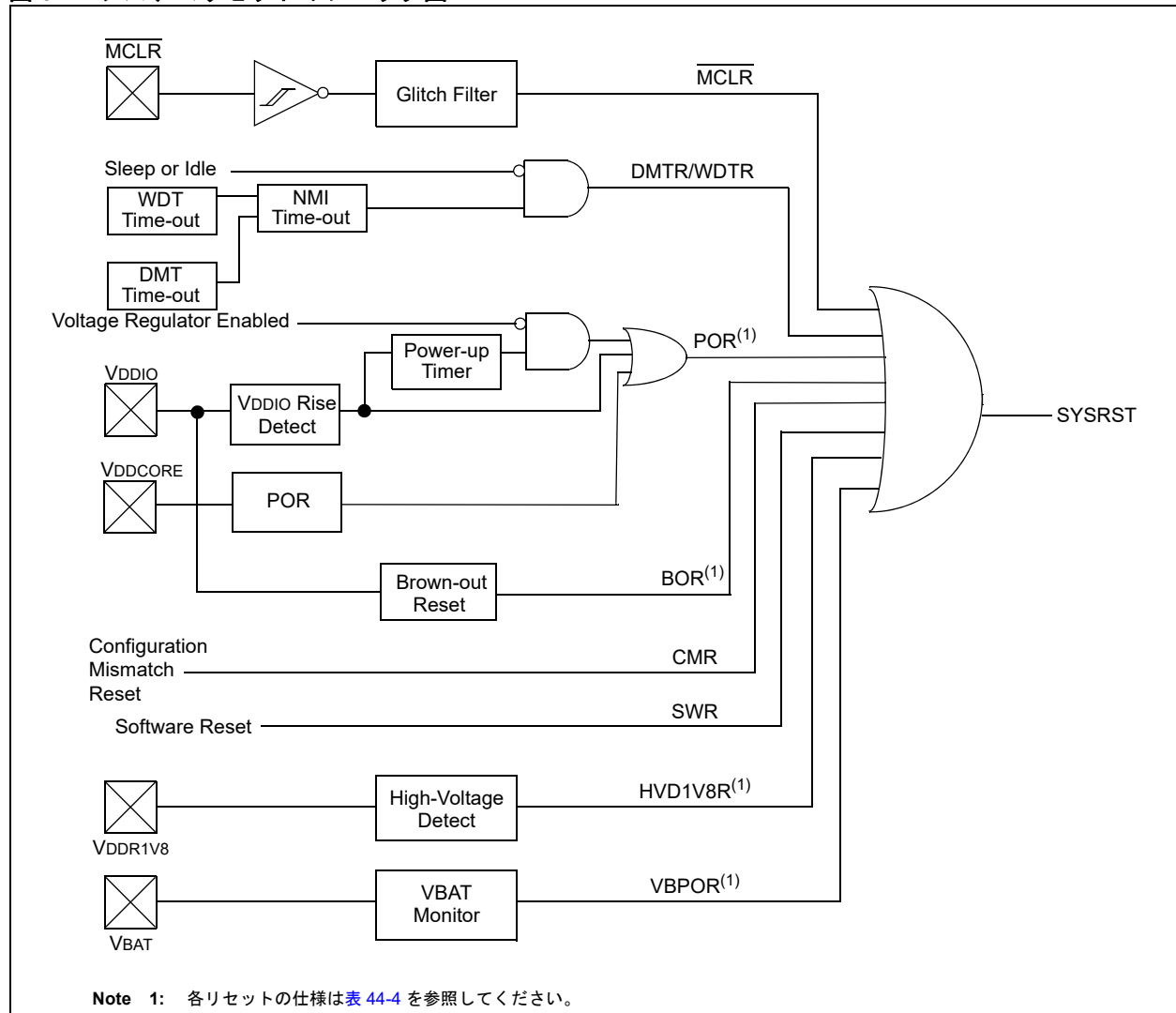
リセット モジュールは全てのリセット要因を集約し、デバイスのマスタリセット信号 (SYSRST) を制御します。以下はデバイスのリセット要因です。

- パワーオン リセット (POR)
- ブラウンアウト リセット (BOR)
- VBAT パワーオン リセット (VBPOR)
- VDDR1V8 での高電圧検出リセット (HVD1V8R)
- マスタクリア リセット ( $\overline{\text{MCLR}}$ )
- ソフトウェア リセット (SWR)
- ウォッチドッグ タイマ リセット (WDTR)
- コンフィグレーション不一致リセット (CMR)
- デッドマンタイマ リセット (DMTR)

全てのタイプのデバイスリセットは、発生時に RCON レジスタ内の対応するステータスビットをセットする事で、発生したリセットのタイプを示します ( [レジスタ 6-1](#) 参照 )。

リセット モジュールの概略ブロック図を [図 6-1](#) に示します。

図 6-1: システムリセットのブロック図



## 6.1 リセット制御レジスタ

表 6-1: リセット関連のレジスタマップ

仮想アドレス (BF80_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0			
1240	RCON	31:16	—	—	HVD1V8R	—	BCFGERR	BCFGFAIL	—	—	—	—	—	—	—	—	VBPOR	VBAT	C802		
		15:0	—	—	—	—	—	DPSLP	CMR	—	EXTR	SWR	DMTO	WDTO	SLEEP	IDLE	BOR	POR	0003		
1250	RSWRST	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SWRST	0000	
1260	RNMICON	31:16	—	—	—	—	—	—	—	DMTO	WDTO	SWNMI	—	—	—	—	GNMI	HLVD	CF	WDTS	0000
		15:0	NMI<15:0>																0000		
1270	PWRCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VREGS	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: 全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 6-1: RCON: リセット制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	R/W-0, HS	U-0	RW-0, HC	RW-0, HC	U-0	U-0
	—	—	HVD1V8R	—	BCFGERR	BCFGFAIL	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	R/W-1, HS	R/W-1, HS
	—	—	—	—	—	—	VBPOR	VBAT
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0, HS	R/W-0, HS	U-0
	—	—	—	—	—	DPSLP <sup>(1)</sup>	CMR	—
7:0	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-1, HS	R/W-1, HS
	EXTR	SWR	DMTO	WDTO	SLEEP	IDLE	BOR <sup>(1)</sup>	POR <sup>(1)</sup>

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

- bit 31-30 **未実装:** 「0」として読み出し
- bit 29 **HVD1V8R:** VDDR1V8 (DDR2) 高電圧検出フラグビット  
1 = VDDR1V8 で高電圧条件が発生した  
0 = VDDR1V8 で高電圧条件は発生していない
- bit **未実装:** 「0」として読み出し
- bit 27 **BCFGERR:** プライマリ コンフィグレーション レジスタ エラーフラグ ビット  
1 = プライマリ コンフィグレーション レジスタの読み出し中にエラーが発生した  
0 = プライマリ コンフィグレーション レジスタの読み出し中にエラーは発生していない
- bit 26 **BCFGFAIL:** プライマリ / 代替コンフィグレーション レジスタ エラーフラグ ビット  
1 = プライマリおよび代替コンフィグレーション レジスタの読み出し中にエラーが発生した  
0 = プライマリおよび代替コンフィグレーション レジスタの読み出し中にエラーは発生していない
- bit 25-18 **未実装:** 「0」として読み出し
- bit 17 **VBPOR:** VBPOR モード フラグビット  
1 = VBAT 領域の POR が発生した  
0 = VBAT 領域の POR は発生していない
- bit 16 **VBAT:** VBAT モード フラグビット  
1 = VBAT からの POR による終了が発生した (真の POR は、有効な VBAT ピン電圧によって確立する必要があります)  
0 = VBAT からの POR による終了は発生していない
- bit 15-11 **未実装:** 「0」として読み出し
- bit 10 **DPSLP:** ディープスリープ モード フラグビット<sup>(1)</sup>  
1 = ディープスリープ モードが発生した  
0 = ディープスリープ モードは発生していない
- bit 9 **CMR:** コンフィグレーション不一致リセット フラグビット  
1 = コンフィグレーション不一致リセットが発生した  
0 = コンフィグレーション不一致リセットは発生していない
- bit 8 **未実装:** 「0」として読み出し
- bit 7 **EXTR:** 外部リセット (MCLR) ピン フラグビット  
1 = マスタクリア (ピン) リセットが発生した  
0 = マスタクリア (ピン) リセットは発生していない
- bit 6 **SWR:** ソフトウェア リセット フラグビット  
1 = ソフトウェア リセットが実行された  
0 = ソフトウェア リセットは実行されなかった
- bit 5 **DMTO:** デッドマンタイマ タイムアウト フラグ ビット  
1 = DMT タイムアウトが発生した  
0 = DMT タイムアウトは発生していない

**Note 1:** 次の発生に備えるため、このビットはユーザ ソフトウェアでクリアする必要があります。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 6-1: RCON: リセット制御レジスタ

- bit 4    **WDTO:** ウォッチドッグ タイマ タイムアウト フラグビット  
1 = WDT タイムアウトが発生した  
0 = WDT タイムアウトは発生していない
- bit 3    **SLEEP:** スリープからの復帰フラグビット  
1 = デバイスはスリープモードから復帰した  
0 = デバイスはスリープモードから復帰したのではない
- bit 2    **IDLE:** アイドルからの復帰フラグビット  
1 = デバイスはアイドルモードから復帰した  
0 = デバイスはアイドルモードから復帰したのではない
- bit 1    **BOR:** ブラウンアウト リセット フラグビット<sup>(1)</sup>  
1 = ブラウンアウト リセットが発生した  
0 = ブラウンアウト リセットは発生していない
- bit 0    **POR:** パワーオン リセット フラグビット<sup>(1)</sup>  
1 = パワーオン リセットが発生した  
0 = パワーオン リセットは発生していない

**Note 1:** 次の発生に備えるため、このビットはユーザ ソフトウェアでクリアする必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 6-2: RSWRST: ソフトウェア リセットレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	W-0, HC
	—	—	—	—	—	—	—	SWRST <sup>(1,2)</sup>

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-1 **未実装:** 「0」として読み出し

bit 0 **SWRST:** ソフトウェア リセット トリガビット (1,2)

1 = ソフトウェア リセットイベントを有効にする

0 = 何もしない

**Note 1:** SWRST ビットに書き込む前に、システムのロック解除シーケンスを実行する必要があります。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張PLLを備えたオシレータ』(DS60001250)を参照してください。

**2:** このビットをセットした場合、RSWRST レジスタの読み出しによってリセットが発生します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 6-3: RNMICON: ノンマスカブル割り込み (NMI) 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	DMTO	WDTO
23:16	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	SWNMI	—	—	—	GNMI	HLVD	CF	WDTs
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	NMIcnt<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	NMIcnt<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-26 **未実装:** 「0」として読み出し

bit 25 **DMTO:** デッドマンタイム タイムアウト フラグ ビット

1 = DMT タイムアウトが発生し、NMI が生成された

0 = DMT タイムアウトは発生していない

このビットをセットすると DMT NMI イベントが発生し、NMIcnt がカウントを始めます。

bit 24 **WDTO:** ウォッチドッグ タイム タイムアウト フラグビット

1 = WDT タイムアウトが発生し、NMI が生成された

0 = WDT タイムアウトは発生していない

このビットをセットすると WDT NMI イベントが発生し、NMIcnt がカウントを始めます。

bit 23 **SWNMI:** ソフトウェア NMI トリガ

1 = NMI を生成する

0 = NMI は生成しない

bit 22-20 **未実装:** 「0」として読み出し

bit 19 **GNMI:** ジェネラル NMI ビット

1 = ジェネラル NMI イベントが検出されたか、ユーザによる NMI イベントが発生した

0 = ジェネラル NMI イベントは検出されていない

ユーザは GNMI を「1」にセットする事で NMI イベントを生成できます。

このビットは、NMIKEY<7:0>(INTCON<31:24>) ビットに 0x4E を書き込む事でセットされます。

bit 18 **HLVD:** 高 / 低電圧検出ビット

1 = HLVD は低電圧条件を検出して NMI を生成した

0 = HLVD は低電圧条件を検出していない

bit 17 **CF:** クロック障害検出ビット

1 = FSCM はクロック障害を検出して NMI を生成した

0 = FSCM はクロック障害を検出していない

このビットをセットすると CF(クロック障害) NMI イベントは発生しますが、クロックは BFRC に切り換わりません

bit 16 **WDTs:** スリープ中ウォッチドッグ タイム タイムアウト フラグビット

1 = スリープ中に WDT タイムアウトが発生し、スリープから復帰した

0 = スリープ中に WDT タイムアウトは発生していない

このビットをセットすると WDT NMI が生成されます。

**Note 1:** このカウンタが「0」に達する前にウォッチドッグ タイム NMI イベントが (スリープ中ではない時に) クリアされた場合、またはデッドマンタイム NMI イベントがクリアされた場合、デバイスリセットは発生しません。この NMI リセットカウンタは、これら 2 種類の NMI イベントにのみ適用されます。

**Note:** SWRST ビットに書き込む前に、システムのロック解除シーケンスを実行する必要があります。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 6-3: RNMICON: ノンマスカブル割り込み (NMI) 制御レジスタ

bit 15-0 **NMICNT<15:0>**: NMI リセットカウンタ値ビット

これらのビットは NMI リセットカウンタが使うリロード値を指定します。

1111111111111111-00000000000000001 = デバイス リセットが発生するまでの SYSCLK クロックサイクル数<sup>(1)</sup>

0000000000000000 = デバイスリセットは NMI のアサートから遅延なく発生する

**Note 1:** このカウンタが「0」に達する前にウォッチドッグ タイマ NMI イベントが (スリープ中ではない時に) クリアされた場合、またはデッドマンタイマ NMI イベントがクリアされた場合、デバイスリセットは発生しません。この NMI リセットカウンタは、これら 2 種類の NMI イベントにのみ適用されます。

**Note:** SWRST ビットに書き込む前に、システムのロック解除シーケンスを実行する必要があります。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 6-4: PWRCON: 電源制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
	—	—	—	—	—	—	—	VREGS

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-1 **未実装:** 「0」として読み出し

bit 0 **VREGS:** 電圧レギュレータ スタンバイ イネーブルビット

1 = スリープ中も電圧レギュレータは動作する

0 = スリープ中は電圧レギュレータはスタンバイモードに移行する

# PIC32MZ グラフィック (DA) ファミリ

## 7.0 CPU 例外と割り込みコントローラ

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『Family Reference Manual, Section 8. Interrupt Controller』(DS60001108) と『PIC32 Family Reference Manual, Section 50. CPU for Devices with MIPS32® microAptiv™ and M-Class Cores』(DS60001192) を参照してください。これらの文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

PIC32MZ DA は、周辺モジュールからの割り込みイベントにตอบสนองして割り込み要求を生成します。割り込みコントローラは CPU の外部にあり、CPU に対して割り込みイベントを提示する前に、それらに優先順位を付けます。

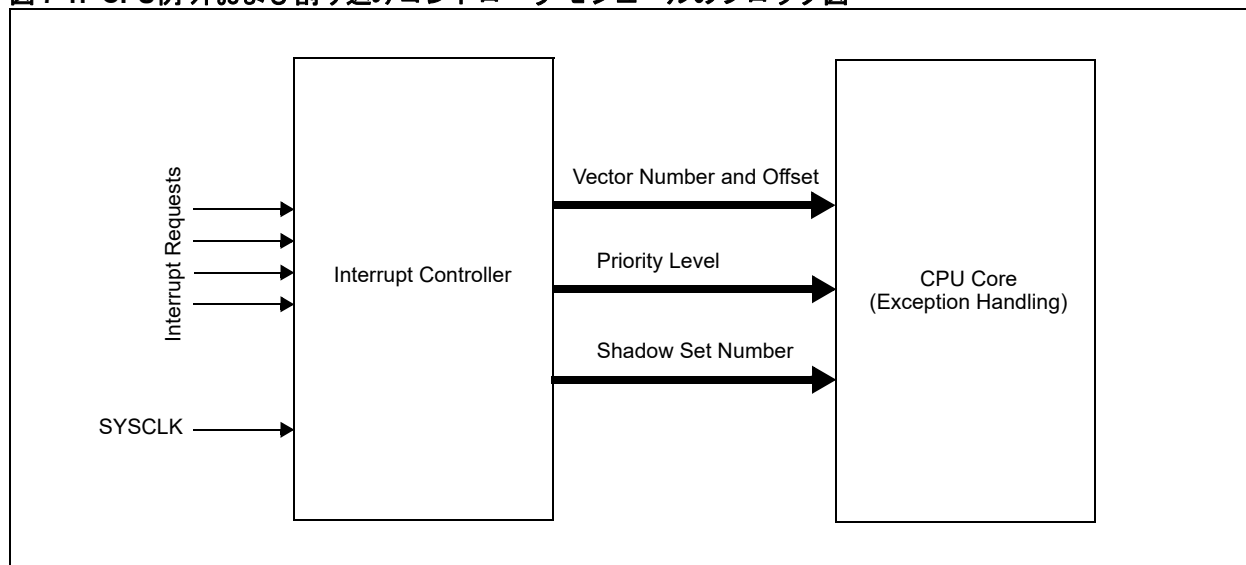
CPU は、例外処理 (7.1 「CPU 例外」参照) の一部として割り込みイベントを処理します。

割り込みコントローラ モジュールの特長は以下の通りです。

- 一意の設定可能オフセットを持つ最大 210 個の割り込み要因と割り込みベクタにより、リダイレクトは不要
- シングルベクタおよびマルチベクタ動作モード
- 5 個の外部割り込み (エッジ極性を選択可能)
- 割り込み近接タイマ
- 各ベクタに対して 7 通りの優先度を選択可能
- 各優先度に 4 通りの副優先度を選択可能
- 任意の優先度に対して使える 7 個のシャドーレジスタセットにより、ソフトウェア コンテキストスイッチを不要にし、割り込みレイテンシを低減
- ソフトウェアで任意の割り込みを生成可能

図 7-1 に割り込みコントローラと CPU 例外のブロック図を示します。

図 7-1: CPU 例外および割り込みコントローラ モジュールのブロック図



## 7.1 CPU 例外

CPU コプロセッサ 0 は、例外を識別して管理するためのロジックを備えています。例外は各種の要因 (Boundary Scan ケース、外部イベント、プログラムエラー等) によって生成できます。表 7-1 に、例外のタイプを優先度の高い物から順番に示します。

表 7-1: MIPS32<sup>®</sup> microActiv<sup>™</sup> マイクロプロセッサ コアの例外タイプ

例外タイプ (優先度順)	概要	分岐先	ステータス ビット	デバッグ ビット	EXCCODE	XC32 関数名
最優先						
リセット	MCLR のアサートまたは パワーオン リセット (POR)	0xBFC0_0000	BEV、ERL	—	—	_on_reset
ソフトリセット	ソフトウェア リセットのアサート	0xBFC0_0000	BEV、SR、 ERL	—	—	_on_reset
DSS	EJTAG デバッグ - シングルステップ	0xBFC0_0480	—	DSS	—	—
DINT	EJTAG デバッグ 割り込み 外部 EJ_DINT 入力のアサートまたは ECR レジス タの EtagBrk ビットのセットにより発生	0xBFC0_0480	—	DINT	—	—
NMI	NMI 信号のアサート	0xBFC0_0000	BEV、 NMI、ERL	—	—	_nmi_handler
マシンチェック	既存エントリと競合する TLB 書き込み	EBASE+0x180	MCHECK、 EXL	—	0x18	_general_exception_handler
割り込み	マスクされていないハードウェアまたはソフト ウェア割り込み信号のアサート	表 7-2 参照	IPL<2:0>	—	0x00	表 7-2 参照
Deferred Watch	Deferred Watch (KIDM=>!KIDM) の遷移によりマスク解除)	EBASE+0x180	WP、EXL	—	0x17	_general_exception_handler
DIB	EJTAG デバッグ ハードウェア命令ブレーク一致	0xBFC0_0480	—	DIB	—	—
WATCH	いずれかのウォッチレジスタ内のアドレスに対す る参照 (フェッチ)	EBASE+0x180	EXL	—	0x17	_general_exception_handler
AdEL	フェッチアドレスのアラインメント エラー 保護されたアドレスを参照するフェッチ	EBASE+0x180	EXL	—	0x04	_general_exception_handler
TLBL	V = 0 のページに対するフェッチの TLB ミスまた は TLB ヒット	Status.EXL = 0 の場合、 EBASE	—	—	0x02	—
		Status.EXL == 1 の場合、 EBASE+0x180	—	—	0x02	_general_exception_handler
TLBL 実行抑止	命令フェッチが、XI ビットがセットされた有効 TLB エントリにマッチした。	EBASE+0x180	EXL	—	0x14	_general_exception_handler
IBE	命令フェッチ バスエラー	EBASE+0x180	EXL	—	0x06	_general_exception_handler

表 7-1: MIPS32<sup>®</sup> microActiv™ マイクロプロセッサ コアの例外タイプ (続き)

例外タイプ (優先度順)	概要	分岐先	ステータス ビット	デバッグ ビット	EXCCODE	XC32 関数名
命令有効性例外	要求されたリソースへのアクセスが許可されなかったために命令が完了できなかった (コプロセッサ使用不可例外)。または命令が不正であった (予約済み命令例外)。同じ命令で両方の例外が発生した場合、コプロセッサ使用不可例外の方が優先されます。	EBASE+0x180	EXL	—	0x0A または 0x0B	_general_exception_handler
実行例外	命令ベースの例外 (整数オーバーフロー、トラップ、システム呼び出し、ブレークポイント、浮動小数点、DSP ASE ステート無効のいずれかの例外) が発生した。	EBASE+0x180	EXL	—	0x08 ~ 0x0C	_general_exception_handler
Tr	トラップの実行 (トラップ条件が真の場合)	EBASE+0x180	EXL	—	0x0D	_general_exception_handler
DDBL/DDBS	EJTAG データアドレス ブレーク (アドレスのみ) またはストア時の EJTAG データ値ブレーク (アドレス + 値)	0xBFC0_0480	—	DDBL また は DDBS	—	—
WATCH	いずれかのウォッチレジスタ内のアドレスに対する参照 (データ)	EBASE+0x180	EXL	—	0x17	_general_exception_handler
AdEL	ロードアドレスのアラインメント エラー ユーザモードのロードがカーネルアドレスを参照した。	EBASE+0x180	EXL	—	0x04	_general_exception_handler
AdES	ストアアドレスのアラインメント エラー ユーザモードのストアがカーネルアドレスを参照した。	EBASE+0x180	EXL	—	0x05	_general_exception_handler
TLBL	V = 0 のページに対するロードの TLB ミスまたは TLB ヒット	EBASE+0x180	EXL	—	0x02	_general_exception_handler
TLBS	V = 0 のページに対するストアの TLB ミスまたは TLB ヒット	EBASE+0x180	EXL	—	0x03	_general_exception_handler
DBE	ロードまたはストア バスエラー	EBASE+0x180	EXL	—	0x07	_general_exception_handler
DDBL	EJTAG データ ハードウェア ブレークポイントがロードデータの比較で一致した。	0xBFC0_0480	—	DDBL	—	—
CBrk	EJTAG 複合ブレークポイント	0xBFC0_0480	—	DIBIMPR、 DDBLIMPR および/または DDBSIMPR	—	—
最低優先度						

## 7.2 割り込み

PIC32MZ DA ファミリーは、ベクタ間隔に変オフセットを使います。これにより、アプリケーション要件に合わせて割り込みベクタの間隔を設定できます。対応する OFFx レジスタにより、各ベクタに一意的割り込みベクタオフセットを設定できます。

可変オフセット機能の詳細は『PIC32 ファミリー リファレンス マニュアル、セクション 08. 割り込み』(DS60001108) 内の「可変オフセット」を参照してください。

表 7-2 に割り込み IRQ、ベクタ、ビット位値を示します。

表 7-2: 割り込み IRQ、ベクタ、ビット位値

割り込み要因 <sup>(1)</sup>	XC32 ベクタ名	IRQ #	ベクタ #	割り込みビット位置				永続的 割り込み
				フラグ	イネーブル	優先度	副優先度	
最高自然順序優先度								
コアタイマ割り込み	_CORE_TIMER_VECTOR	0	OFF000<17:1>	IFS0<0>	IEC0<0>	IPC0<4:2>	IPC0<1:0>	No
コア ソフトウェア割り込み 0	_CORE_SOFTWARE_0_VECTOR	1	OFF001<17:1>	IFS0<1>	IEC0<1>	IPC0<12:10>	IPC0<9:8>	No
コア ソフトウェア割り込み 1	_CORE_SOFTWARE_1_VECTOR	2	OFF002<17:1>	IFS0<2>	IEC0<2>	IPC0<20:18>	IPC0<17:16>	No
外部割り込み 0	_EXTERNAL_0_VECTOR	3	OFF003<17:1>	IFS0<3>	IEC0<3>	IPC0<28:26>	IPC0<25:24>	No
Timer1	_TIMER_1_VECTOR	4	OFF004<17:1>	IFS0<4>	IEC0<4>	IPC1<4:2>	IPC1<1:0>	No
入力キャプチャ 1 エラー	_INPUT_CAPTURE_1_ERROR_VECTOR	5	OFF005<17:1>	IFS0<5>	IEC0<5>	IPC1<12:10>	IPC1<9:8>	Yes
入力キャプチャ 1	_INPUT_CAPTURE_1_VECTOR	6	OFF006<17:1>	IFS0<6>	IEC0<6>	IPC1<20:18>	IPC1<17:16>	Yes
出力コンペア 1	_OUTPUT_COMPARE_1_VECTOR	7	OFF007<17:1>	IFS0<7>	IEC0<7>	IPC1<28:26>	IPC1<25:24>	No
外部割り込み 1	_EXTERNAL_1_VECTOR	8	OFF008<17:1>	IFS0<8>	IEC0<8>	IPC2<4:2>	IPC2<1:0>	No
Timer2	_TIMER_2_VECTOR	9	OFF009<17:1>	IFS0<9>	IEC0<9>	IPC2<12:10>	IPC2<9:8>	No
入力キャプチャ 2 エラー	_INPUT_CAPTURE_2_ERROR_VECTOR	10	OFF010<17:1>	IFS0<10>	IEC0<10>	IPC2<20:18>	IPC2<17:16>	Yes
入力キャプチャ 2	_INPUT_CAPTURE_2_VECTOR	11	OFF011<17:1>	IFS0<11>	IEC0<11>	IPC2<28:26>	IPC2<25:24>	Yes
出力コンペア 2	_OUTPUT_COMPARE_2_VECTOR	12	OFF012<17:1>	IFS0<12>	IEC0<12>	IPC3<4:2>	IPC3<1:0>	No
外部割り込み 2	_EXTERNAL_2_VECTOR	13	OFF013<17:1>	IFS0<13>	IEC0<13>	IPC3<12:10>	IPC3<9:8>	No
Timer3	_TIMER_3_VECTOR	14	OFF014<17:1>	IFS0<14>	IEC0<14>	IPC3<20:18>	IPC3<17:16>	No
入力キャプチャ 3 エラー	_INPUT_CAPTURE_3_ERROR_VECTOR	15	OFF015<17:1>	IFS0<15>	IEC0<15>	IPC3<28:26>	IPC3<25:24>	Yes
入力キャプチャ 3	_INPUT_CAPTURE_3_VECTOR	16	OFF016<17:1>	IFS0<16>	IEC0<16>	IPC4<4:2>	IPC4<1:0>	Yes
出力コンペア 3	_OUTPUT_COMPARE_3_VECTOR	17	OFF017<17:1>	IFS0<17>	IEC0<17>	IPC4<12:10>	IPC4<9:8>	No
外部割り込み 3	_EXTERNAL_3_VECTOR	18	OFF018<17:1>	IFS0<18>	IEC0<18>	IPC4<20:18>	IPC4<17:16>	No
Timer4	_TIMER_4_VECTOR	19	OFF019<17:1>	IFS0<19>	IEC0<19>	IPC4<28:26>	IPC4<25:24>	No
入力キャプチャ 4 エラー	_INPUT_CAPTURE_4_ERROR_VECTOR	20	OFF020<17:1>	IFS0<20>	IEC0<20>	IPC5<4:2>	IPC5<1:0>	Yes
入力キャプチャ 4	_INPUT_CAPTURE_4_VECTOR	21	OFF021<17:1>	IFS0<21>	IEC0<21>	IPC5<12:10>	IPC5<9:8>	Yes
出力コンペア 4	_OUTPUT_COMPARE_4_VECTOR	22	OFF022<17:1>	IFS0<22>	IEC0<22>	IPC5<20:18>	IPC5<17:16>	No

**Note 1:** 全てのデバイスで全ての割り込み要因が使える訳ではありません。デバイスで利用可能な周辺モジュールについてはファミリー機能一覧表 (表 1 ~ 表 2) を参照してください。

**2:** リセット時の GLDC 割り込み (HSYNC と VSYNC) は永続的です。しかし、IRQCON ビット (GLCDINT<31>) を使って割り込みのタイプを非永続的に変更できます。

表 7-2: 割り込み IRQ、ベクタ、ビット位値 (続き)

割り込み要因 <sup>(1)</sup>	XC32 ベクタ名	IRQ #	ベクタ #	割り込みビット位置				永続的 割り込み
				フラグ	イネーブル	優先度	副優先度	
外部割り込み 4	_EXTERNAL_4_VECTOR	23	OFF023<17:1>	IFS0<23>	IEC0<23>	IPC5<28:26>	IPC5<25:24>	No
Timer5	_TIMER_5_VECTOR	24	OFF024<17:1>	IFS0<24>	IEC0<24>	IPC6<4:2>	IPC6<1:0>	No
入力キャプチャ 5 エラー	_INPUT_CAPTURE_5_ERROR_VECTOR	25	OFF025<17:1>	IFS0<25>	IEC0<25>	IPC6<12:10>	IPC6<9:8>	Yes
入力キャプチャ 5	_INPUT_CAPTURE_5_VECTOR	26	OFF026<17:1>	IFS0<26>	IEC0<26>	IPC6<20:18>	IPC6<17:16>	Yes
出力コンペア 5	_OUTPUT_COMPARE_5_VECTOR	27	OFF027<17:1>	IFS0<27>	IEC0<27>	IPC6<28:26>	IPC6<25:24>	No
Timer6	_TIMER_6_VECTOR	28	OFF028<17:1>	IFS0<28>	IEC0<28>	IPC7<4:2>	IPC7<1:0>	No
入力キャプチャ 6 エラー	_INPUT_CAPTURE_6_ERROR_VECTOR	29	OFF029<17:1>	IFS0<29>	IEC0<29>	IPC7<12:10>	IPC7<9:8>	Yes
入力キャプチャ 6	_INPUT_CAPTURE_6_VECTOR	30	OFF030<17:1>	IFS0<30>	IEC0<30>	IPC7<20:18>	IPC7<17:16>	Yes
出力コンペア 6	_OUTPUT_COMPARE_6_VECTOR	31	OFF031<17:1>	IFS0<31>	IEC0<31>	IPC7<28:26>	IPC7<25:24>	No
Timer7	_TIMER_7_VECTOR	32	OFF032<17:1>	IFS1<0>	IEC1<0>	IPC8<4:2>	IPC8<1:0>	No
入力キャプチャ 7 エラー	_INPUT_CAPTURE_7_ERROR_VECTOR	33	OFF033<17:1>	IFS1<1>	IEC1<1>	IPC8<12:10>	IPC8<9:8>	Yes
入力キャプチャ 7	_INPUT_CAPTURE_7_VECTOR	34	OFF034<17:1>	IFS1<2>	IEC1<2>	IPC8<20:18>	IPC8<17:16>	Yes
出力コンペア 7	_OUTPUT_COMPARE_7_VECTOR	35	OFF035<17:1>	IFS1<3>	IEC1<3>	IPC8<28:26>	IPC8<25:24>	No
Timer8	_TIMER_8_VECTOR	36	OFF036<17:1>	IFS1<4>	IEC1<4>	IPC9<4:2>	IPC9<1:0>	No
入力キャプチャ 8 エラー	_INPUT_CAPTURE_8_ERROR_VECTOR	37	OFF037<17:1>	IFS1<5>	IEC1<5>	IPC9<12:10>	IPC9<9:8>	Yes
入力キャプチャ 8	_INPUT_CAPTURE_8_VECTOR	38	OFF038<17:1>	IFS1<6>	IEC1<6>	IPC9<20:18>	IPC9<17:16>	Yes
出力コンペア 8	_OUTPUT_COMPARE_8_VECTOR	39	OFF039<17:1>	IFS1<7>	IEC1<7>	IPC9<28:26>	IPC9<25:24>	No
Timer9	_TIMER_9_VECTOR	40	OFF040<17:1>	IFS1<8>	IEC1<8>	IPC10<4:2>	IPC10<1:0>	No
入力キャプチャ 9 エラー	_INPUT_CAPTURE_9_ERROR_VECTOR	41	OFF041<17:1>	IFS1<9>	IEC1<9>	IPC10<12:10>	IPC10<9:8>	Yes
入力キャプチャ 9	_INPUT_CAPTURE_9_VECTOR	42	OFF042<17:1>	IFS1<10>	IEC1<10>	IPC10<20:18>	IPC10<17:16>	Yes
出力コンペア 9	_OUTPUT_COMPARE_9_VECTOR	43	OFF043<17:1>	IFS1<11>	IEC1<11>	IPC10<28:26>	IPC10<25:24>	No
ADC グローバル割り込み	_ADC_VECTOR	44	OFF044<17:1>	IFS1<12>	IEC1<12>	IPC11<4:2>	IPC11<1:0>	Yes
ADC FIFO 割り込み	_ADC_FIFO_VECTOR	45	OFF045<17:1>	IFS1<13>	IEC1<13>	IPC11<12:10>	IPC11<9:8>	Yes
ADC デジタル コンパレータ 1	_ADC_DC1_VECTOR	46	OFF046<17:1>	IFS1<14>	IEC1<14>	IPC11<20:18>	IPC11<17:16>	Yes
ADC デジタル コンパレータ 2	_ADC_DC2_VECTOR	47	OFF047<17:1>	IFS1<15>	IEC1<15>	IPC11<28:26>	IPC11<25:24>	Yes
ADC デジタル コンパレータ 3	_ADC_DC3_VECTOR	48	OFF048<17:1>	IFS1<16>	IEC1<16>	IPC12<4:2>	IPC12<1:0>	Yes
ADC デジタル コンパレータ 4	_ADC_DC4_VECTOR	49	OFF049<17:1>	IFS1<17>	IEC1<17>	IPC12<12:10>	IPC12<9:8>	Yes
ADC デジタル コンパレータ 5	_ADC_DC5_VECTOR	50	OFF050<17:1>	IFS1<18>	IEC1<18>	IPC12<20:18>	IPC12<17:16>	Yes
ADC デジタル コンパレータ 6	_ADC_DC6_VECTOR	51	OFF051<17:1>	IFS1<19>	IEC1<19>	IPC12<28:26>	IPC12<25:24>	Yes

**Note 1:** 全てのデバイスで全ての割り込み要因が使える訳ではありません。デバイスで利用可能な周辺モジュールについてはファミリ機能一覧表 (表 1 ~ 表 2) を参照してください。

**2:** リセット時の GLDC 割り込み (HSYNC と VSYNC) は永続的です。しかし、IRQCON ビット (GLCDINT<31>) を使って割り込みのタイプを非永続的に変更できます。

表 7-2: 割り込み IRQ、ベクタ、ビット位値 (続き)

割り込み要因 <sup>(1)</sup>	XC32 ベクタ名	IRQ #	ベクタ #	割り込みビット位置				永続的 割り込み
				フラグ	イネーブル	優先度	副優先度	
ADC デジタルフィルタ 1	_ADC_DF1_VECTOR	52	OFF052<17:1>	IFS1<20>	IEC1<20>	IPC13<4:2>	IPC13<1:0>	Yes
ADC デジタルフィルタ 2	_ADC_DF2_VECTOR	53	OFF053<17:1>	IFS1<21>	IEC1<21>	IPC13<12:10>	IPC13<9:8>	Yes
ADC デジタルフィルタ 3	_ADC_DF3_VECTOR	54	OFF054<17:1>	IFS1<22>	IEC1<22>	IPC13<20:18>	IPC13<17:16>	Yes
ADC デジタルフィルタ 4	_ADC_DF4_VECTOR	55	OFF055<17:1>	IFS1<23>	IEC1<23>	IPC13<28:26>	IPC13<25:24>	Yes
ADC デジタルフィルタ 5	_ADC_DF5_VECTOR	56	OFF056<17:1>	IFS1<24>	IEC1<24>	IPC14<4:2>	IPC14<1:0>	Yes
ADC デジタルフィルタ 6	_ADC_DF6_VECTOR	57	OFF057<17:1>	IFS1<25>	IEC1<25>	IPC14<12:10>	IPC14<9:8>	Yes
ADC 異常	_ADC_FAULT_VECTOR	58	OFF058<17:1>	IFS1<26>	IEC1<26>	IPC14<20:18>	IPC14<17:16>	Yes
ADC データ 0	_ADC_DATA0_VECTOR	59	OFF059<17:1>	IFS1<27>	IEC1<27>	IPC14<28:26>	IPC14<25:24>	Yes
ADC データ 1	_ADC_DATA1_VECTOR	60	OFF060<17:1>	IFS1<28>	IEC1<28>	IPC15<4:2>	IPC15<1:0>	Yes
ADC データ 2	_ADC_DATA2_VECTOR	61	OFF061<17:1>	IFS1<29>	IEC1<29>	IPC15<12:10>	IPC15<9:8>	Yes
ADC データ 3	_ADC_DATA3_VECTOR	62	OFF062<17:1>	IFS1<30>	IEC1<30>	IPC15<20:18>	IPC15<17:16>	Yes
ADC データ 4	_ADC_DATA4_VECTOR	63	OFF063<17:1>	IFS1<31>	IEC1<31>	IPC15<28:26>	IPC15<25:24>	Yes
ADC データ 5	_ADC_DATA5_VECTOR	64	OFF064<17:1>	IFS2<0>	IEC2<0>	IPC16<4:2>	IPC16<1:0>	Yes
ADC データ 6	_ADC_DATA6_VECTOR	65	OFF065<17:1>	IFS2<1>	IEC2<1>	IPC16<12:10>	IPC16<9:8>	Yes
ADC データ 7	_ADC_DATA7_VECTOR	66	OFF066<17:1>	IFS2<2>	IEC2<2>	IPC16<20:18>	IPC16<17:16>	Yes
ADC データ 8	_ADC_DATA8_VECTOR	67	OFF067<17:1>	IFS2<3>	IEC2<3>	IPC16<28:26>	IPC16<25:24>	Yes
ADC データ 9	_ADC_DATA9_VECTOR	68	OFF068<17:1>	IFS2<4>	IEC2<4>	IPC17<4:2>	IPC17<1:0>	Yes
ADC データ 10	_ADC_DATA10_VECTOR	69	OFF069<17:1>	IFS2<5>	IEC2<5>	IPC17<12:10>	IPC17<9:8>	Yes
ADC データ 11	_ADC_DATA11_VECTOR	70	OFF070<17:1>	IFS2<6>	IEC2<6>	IPC17<20:18>	IPC17<17:16>	Yes
ADC データ 12	_ADC_DATA12_VECTOR	71	OFF071<17:1>	IFS2<7>	IEC2<7>	IPC17<28:26>	IPC17<25:24>	Yes
ADC データ 13	_ADC_DATA13_VECTOR	72	OFF072<17:1>	IFS2<8>	IEC2<8>	IPC18<4:2>	IPC18<1:0>	Yes
ADC データ 14	_ADC_DATA14_VECTOR	73	OFF073<17:1>	IFS2<9>	IEC2<9>	IPC18<12:10>	IPC18<9:8>	Yes
ADC データ 15	_ADC_DATA15_VECTOR	74	OFF074<17:1>	IFS2<10>	IEC2<10>	IPC18<20:18>	IPC18<17:16>	Yes
ADC データ 16	_ADC_DATA16_VECTOR	75	OFF075<17:1>	IFS2<11>	IEC2<11>	IPC18<28:26>	IPC18<25:24>	Yes
ADC データ 17	_ADC_DATA17_VECTOR	76	OFF076<17:1>	IFS2<12>	IEC2<12>	IPC19<4:2>	IPC19<1:0>	Yes
ADC データ 18	_ADC_DATA18_VECTOR	77	OFF077<17:1>	IFS2<13>	IEC2<13>	IPC19<12:10>	IPC19<9:8>	Yes
ADC データ 19	_ADC_DATA19_VECTOR	78	OFF078<17:1>	IFS2<14>	IEC2<14>	IPC19<20:18>	IPC19<17:16>	Yes
ADC データ 20	_ADC_DATA20_VECTOR	79	OFF079<17:1>	IFS2<15>	IEC2<15>	IPC19<28:26>	IPC19<25:24>	Yes
ADC データ 21	_ADC_DATA21_VECTOR	80	OFF080<17:1>	IFS2<16>	IEC2<16>	IPC20<4:2>	IPC20<1:0>	Yes

**Note 1:** 全てのデバイスで全ての割り込み要因が使える訳ではありません。デバイスで利用可能な周辺モジュールについてはファミリー機能一覧表 (表 1 ~ 表 2) を参照してください。

**2:** リセット時の GLDC 割り込み (HSYNC と VSYNC) は永続的です。しかし、IRQCON ビット (GLCDINT<31>) を使って割り込みのタイプを非永続的に変更できます。



表 7-2: 割り込み IRQ、ベクタ、ビット位値 (続き)

割り込み要因 <sup>(1)</sup>	XC32 ベクタ名	IRQ #	ベクタ #	割り込みビット位置				永続的 割り込み
				フラグ	イネーブル	優先度	副優先度	
ADC データ 22	_ADC_DATA22_VECTOR	81	OFF081<17:1>	IFS2<17>	IEC2<17>	IPC20<12:10>	IPC20<9:8>	Yes
ADC データ 23	_ADC_DATA23_VECTOR	82	OFF082<17:1>	IFS2<18>	IEC2<18>	IPC20<20:18>	IPC20<17:16>	Yes
ADC データ 24	_ADC_DATA24_VECTOR	83	OFF083<17:1>	IFS2<19>	IEC2<19>	IPC20<28:26>	IPC20<25:24>	Yes
ADC データ 25	_ADC_DATA25_VECTOR	84	OFF084<17:1>	IFS2<20>	IEC2<20>	IPC21<4:2>	IPC21<1:0>	Yes
ADC データ 26	_ADC_DATA26_VECTOR	85	OFF085<17:1>	IFS2<21>	IEC2<21>	IPC21<12:10>	IPC21<9:8>	Yes
ADC データ 27	_ADC_DATA27_VECTOR	86	OFF086<17:1>	IFS2<22>	IEC2<22>	IPC21<20:18>	IPC21<17:16>	Yes
ADC データ 28	_ADC_DATA28_VECTOR	87	OFF087<17:1>	IFS2<23>	IEC2<23>	IPC21<28:26>	IPC21<25:24>	Yes
ADC データ 29	_ADC_DATA29_VECTOR	88	OFF088<17:1>	IFS2<24>	IEC2<24>	IPC22<4:2>	IPC22<1:0>	Yes
ADC データ 30	_ADC_DATA30_VECTOR	89	OFF089<17:1>	IFS2<25>	IEC2<25>	IPC22<12:10>	IPC22<9:8>	Yes
ADC データ 31	_ADC_DATA31_VECTOR	90	OFF090<17:1>	IFS2<26>	IEC2<26>	IPC22<20:18>	IPC22<17:16>	Yes
ADC データ 32	_ADC_DATA32_VECTOR	91	OFF091<17:1>	IFS2<27>	IEC2<27>	IPC22<28:26>	IPC22<25:24>	Yes
ADC データ 33	_ADC_DATA33_VECTOR	92	OFF092<17:1>	IFS2<28>	IEC2<28>	IPC23<4:2>	IPC23<1:0>	Yes
ADC データ 34	_ADC_DATA34_VECTOR	93	OFF093<17:1>	IFS2<29>	IEC2<29>	IPC23<12:10>	IPC23<9:8>	Yes
ADC データ 35	_ADC_DATA35_VECTOR	94	OFF094<17:1>	IFS2<30>	IEC2<30>	IPC23<20:18>	IPC23<17:16>	Yes
ADC データ 36	_ADC_DATA36_VECTOR	95	OFF095<17:1>	IFS2<31>	IEC2<31>	IPC23<28:26>	IPC23<25:24>	Yes
ADC データ 37	_ADC_DATA37_VECTOR	96	OFF096<17:1>	IFS3<0>	IEC3<0>	IPC24<4:2>	IPC24<1:0>	Yes
ADC データ 38	_ADC_DATA38_VECTOR	97	OFF097<17:1>	IFS3<1>	IEC3<1>	IPC24<12:10>	IPC24<9:8>	Yes
ADC データ 39	_ADC_DATA39_VECTOR	98	OFF098<17:1>	IFS3<2>	IEC3<2>	IPC24<20:18>	IPC24<17:16>	Yes
ADC データ 40	_ADC_DATA40_VECTOR	99	OFF099<17:1>	IFS3<3>	IEC3<3>	IPC24<28:26>	IPC24<25:24>	Yes
ADC データ 41	_ADC_DATA41_VECTOR	100	OFF100<17:1>	IFS3<4>	IEC3<4>	IPC25<4:2>	IPC25<1:0>	Yes
ADC データ 42	_ADC_DATA42_VECTOR	101	OFF101<17:1>	IFS3<5>	IEC3<5>	IPC25<12:10>	IPC25<9:8>	Yes
ADC データ 43	_ADC_DATA43_VECTOR	102	OFF102<17:1>	IFS3<6>	IEC3<6>	IPC25<20:18>	IPC25<17:16>	Yes
USB サスペンド/レジュームイベント	_USB1_SR_VECTOR	103	OFF103<17:1>	IFS3<7>	IEC3<7>	IPC25<28:26>	IPC25<25:24>	No
コア性能カウンタ割り込み	_CORE_PERF_COUNT_VECTOR	104	OFF104<17:1>	IFS3<8>	IEC3<8>	IPC26<4:2>	IPC26<1:0>	No
コア高速デバッグチャンネル割り込み	_CORE_FAST_DEBUG_CHAN_VECTOR	105	OFF105<17:1>	IFS3<9>	IEC3<9>	IPC26<12:10>	IPC26<9:8>	Yes
システムバス保護違反	_SYSTEM_BUS_PROTECTION_VECTOR	106	OFF106<17:1>	IFS3<10>	IEC3<10>	IPC26<20:18>	IPC26<17:16>	Yes
暗号エンジンイベント	_CRYPTO_VECTOR	107	OFF107<17:1>	IFS3<11>	IEC3<11>	IPC26<28:26>	IPC26<25:24>	Yes
予約済み	—	108	—	—	—	—	—	—
SPI1 異常	_SPI1_FAULT_VECTOR	109	OFF109<17:1>	IFS3<13>	IEC3<13>	IPC27<12:10>	IPC27<9:8>	Yes

**Note 1:** 全てのデバイスで全ての割り込み要因が使える訳ではありません。デバイスで利用可能な周辺モジュールについてはファミリー機能一覧表 (表 1 ~ 表 2) を参照してください。

**2:** リセット時の GLDC 割り込み (HSYNC と VSYNC) は永続的です。しかし、IRQCON ビット (GLCDINT<31>) を使って割り込みのタイプを非永続的に変更できます。

表 7-2: 割り込み IRQ、ベクタ、ビット位値 (続き)

割り込み要因 <sup>(1)</sup>	XC32 ベクタ名	IRQ #	ベクタ #	割り込みビット位置				永続的 割り込み
				フラグ	イネーブル	優先度	副優先度	
SPI1 受信完了	_SPI1_RX_VECTOR	110	OFF110<17:1>	IFS3<14>	IEC3<14>	IPC27<20:18>	IPC27<17:16>	Yes
SPI1 転送完了	_SPI1_TX_VECTOR	111	OFF111<17:1>	IFS3<15>	IEC3<15>	IPC27<28:26>	IPC27<25:24>	Yes
UART1 異常	_UART1_FAULT_VECTOR	112	OFF112<17:1>	IFS3<16>	IEC3<16>	IPC28<4:2>	IPC28<1:0>	Yes
UART1 受信完了	_UART1_RX_VECTOR	113	OFF113<17:1>	IFS3<17>	IEC3<17>	IPC28<12:10>	IPC28<9:8>	Yes
UART1 転送完了	_UART1_TX_VECTOR	114	OFF114<17:1>	IFS3<18>	IEC3<18>	IPC28<20:18>	IPC28<17:16>	Yes
I2C1 バス衝突イベント	_I2C1_BUS_VECTOR	115	OFF115<17:1>	IFS3<19>	IEC3<19>	IPC28<28:26>	IPC28<25:24>	Yes
I2C1 スレーブイベント	_I2C1_SLAVE_VECTOR	116	OFF116<17:1>	IFS3<20>	IEC3<20>	IPC29<4:2>	IPC29<1:0>	Yes
I2C1 マスタイベント	_I2C1_MASTER_VECTOR	117	OFF117<17:1>	IFS3<21>	IEC3<21>	IPC29<12:10>	IPC29<9:8>	Yes
PORTA 入力状態変化割り込み	_CHANGE_NOTICE_A_VECTOR	118	OFF118<17:1>	IFS3<22>	IEC3<22>	IPC29<20:18>	IPC29<17:16>	Yes
PORTB 入力状態変化割り込み	_CHANGE_NOTICE_B_VECTOR	119	OFF119<17:1>	IFS3<23>	IEC3<23>	IPC29<28:26>	IPC29<25:24>	Yes
PORTC 入力状態変化割り込み	_CHANGE_NOTICE_C_VECTOR	120	OFF120<17:1>	IFS3<24>	IEC3<24>	IPC30<4:2>	IPC30<1:0>	Yes
PORTD 入力状態変化割り込み	_CHANGE_NOTICE_D_VECTOR	121	OFF121<17:1>	IFS3<25>	IEC3<25>	IPC30<12:10>	IPC30<9:8>	Yes
PORTE 入力状態変化割り込み	_CHANGE_NOTICE_E_VECTOR	122	OFF122<17:1>	IFS3<26>	IEC3<26>	IPC30<20:18>	IPC30<17:16>	Yes
PORTF 入力状態変化割り込み	_CHANGE_NOTICE_F_VECTOR	123	OFF123<17:1>	IFS3<27>	IEC3<27>	IPC30<28:26>	IPC30<25:24>	Yes
PORTG 入力状態変化割り込み	_CHANGE_NOTICE_G_VECTOR	124	OFF124<17:1>	IFS3<28>	IEC3<28>	IPC31<4:2>	IPC31<1:0>	Yes
PORTH 入力状態変化割り込み	_CHANGE_NOTICE_H_VECTOR	125	OFF125<17:1>	IFS3<29>	IEC3<29>	IPC31<12:10>	IPC31<9:8>	Yes
PORTJ 入力状態変化割り込み	_CHANGE_NOTICE_J_VECTOR	126	OFF126<17:1>	IFS3<30>	IEC3<30>	IPC31<20:18>	IPC31<17:16>	Yes
PORTK 入力状態変化割り込み	_CHANGE_NOTICE_K_VECTOR	127	OFF127<17:1>	IFS3<31>	IEC3<31>	IPC31<28:26>	IPC31<25:24>	Yes
パラレルマスタ ポート	_PMP_VECTOR	128	OFF128<17:1>	IFS4<0>	IEC4<0>	IPC32<4:2>	IPC32<1:0>	Yes
パラレルマスタ ポートエラー	_PMP_ERROR_VECTOR	129	OFF129<17:1>	IFS4<1>	IEC4<1>	IPC32<12:10>	IPC32<9:8>	Yes
コンパレータ 1 割り込み	_COMPARATOR_1_VECTOR	130	OFF130<17:1>	IFS4<2>	IEC4<2>	IPC32<20:18>	IPC32<17:16>	No
コンパレータ 2 割り込み	_COMPARATOR_2_VECTOR	131	OFF131<17:1>	IFS4<3>	IEC4<3>	IPC32<28:26>	IPC32<25:24>	No
USB 一般イベント	_USB1_VECTOR	132	OFF132<17:1>	IFS4<4>	IEC4<4>	IPC33<4:2>	IPC33<1:0>	Yes
USB DMA イベント	_USB1_DMA_VECTOR	133	OFF133<17:1>	IFS4<5>	IEC4<5>	IPC33<12:10>	IPC33<9:8>	Yes
DMA チャンネル 0	_DMA0_VECTOR	134	OFF134<17:1>	IFS4<6>	IEC4<6>	IPC33<20:18>	IPC33<17:16>	No
DMA チャンネル 1	_DMA1_VECTOR	135	OFF135<17:1>	IFS4<7>	IEC4<7>	IPC33<28:26>	IPC33<25:24>	No
DMA チャンネル 2	_DMA2_VECTOR	136	OFF136<17:1>	IFS4<8>	IEC4<8>	IPC34<4:2>	IPC34<1:0>	No
DMA チャンネル 3	_DMA3_VECTOR	137	OFF137<17:1>	IFS4<9>	IEC4<9>	IPC34<12:10>	IPC34<9:8>	No
DMA チャンネル 4	_DMA4_VECTOR	138	OFF138<17:1>	IFS4<10>	IEC4<10>	IPC34<20:18>	IPC34<17:16>	No

**Note 1:** 全てのデバイスで全ての割り込み要因が使える訳ではありません。デバイスで利用可能な周辺モジュールについてはファミリー機能一覧表 (表 1 ~ 表 2) を参照してください。

**2:** リセット時の GLDC 割り込み (HSYNC と VSYNC) は永続的です。しかし、IRQCON ビット (GLCDINT<31>) を使って割り込みのタイプを非永続的に変更できます。

表 7-2: 割り込み IRQ、ベクタ、ビット位値 (続き)

割り込み要因 <sup>(1)</sup>	XC32 ベクタ名	IRQ #	ベクタ #	割り込みビット位置				永続的 割り込み
				フラグ	イネーブル	優先度	副優先度	
DMA チャンネル 5	_DMA5_VECTOR	139	OFF139<17:1>	IFS4<11>	IEC4<11>	IPC34<28:26>	IPC34<25:24>	No
DMA チャンネル 6	_DMA6_VECTOR	140	OFF140<17:1>	IFS4<12>	IEC4<12>	IPC35<4:2>	IPC35<1:0>	No
DMA チャンネル 7	_DMA7_VECTOR	141	OFF141<17:1>	IFS4<13>	IEC4<13>	IPC35<12:10>	IPC35<9:8>	No
SPI2 異常	_SPI2_FAULT_VECTOR	142	OFF142<17:1>	IFS4<14>	IEC4<14>	IPC35<20:18>	IPC35<17:16>	Yes
SPI2 受信完了	_SPI2_RX_VECTOR	143	OFF143<17:1>	IFS4<15>	IEC4<15>	IPC35<28:26>	IPC35<25:24>	Yes
SPI2 転送完了	_SPI2_TX_VECTOR	144	OFF144<17:1>	IFS4<16>	IEC4<16>	IPC36<4:2>	IPC36<1:0>	Yes
UART2 異常	_UART2_FAULT_VECTOR	145	OFF145<17:1>	IFS4<17>	IEC4<17>	IPC36<12:10>	IPC36<9:8>	Yes
UART2 受信完了	_UART2_RX_VECTOR	146	OFF146<17:1>	IFS4<18>	IEC4<18>	IPC36<20:18>	IPC36<17:16>	Yes
UART2 転送完了	_UART2_TX_VECTOR	147	OFF147<17:1>	IFS4<19>	IEC4<19>	IPC36<28:26>	IPC36<25:24>	Yes
I2C2 バス衝突イベント	_I2C2_BUS_VECTOR	148	OFF148<17:1>	IFS4<20>	IEC4<20>	IPC37<4:2>	IPC37<1:0>	Yes
I2C2 スレーブイベント	_I2C2_SLAVE_VECTOR	149	OFF149<17:1>	IFS4<21>	IEC4<21>	IPC37<12:10>	IPC37<9:8>	Yes
I2C2 マスタイベント	_I2C2_MASTER_VECTOR	150	OFF150<17:1>	IFS4<22>	IEC4<22>	IPC37<20:18>	IPC37<17:16>	Yes
CAN1	_CAN1_VECTOR	151	OFF151<17:1>	IFS4<23>	IEC4<23>	IPC37<28:26>	IPC37<25:24>	Yes
CAN2	_CAN2_VECTOR	152	OFF152<17:1>	IFS4<24>	IEC4<24>	IPC38<4:2>	IPC38<1:0>	Yes
Ethernet 割り込み	_ETHERNET_VECTOR	153	OFF153<17:1>	IFS4<25>	IEC4<25>	IPC38<12:10>	IPC38<9:8>	Yes
SPI3 異常	_SPI3_FAULT_VECTOR	154	OFF154<17:1>	IFS4<26>	IEC4<26>	IPC38<20:18>	IPC38<17:16>	Yes
SPI3 受信完了	_SPI3_RX_VECTOR	155	OFF155<17:1>	IFS4<27>	IEC4<27>	IPC38<28:26>	IPC38<25:24>	Yes
SPI3 転送完了	_SPI3_TX_VECTOR	156	OFF156<17:1>	IFS4<28>	IEC4<28>	IPC39<4:2>	IPC39<1:0>	Yes
UART3 異常	_UART3_FAULT_VECTOR	157	OFF157<17:1>	IFS4<29>	IEC4<29>	IPC39<12:10>	IPC39<9:8>	Yes
UART3 受信完了	_UART3_RX_VECTOR	158	OFF158<17:1>	IFS4<30>	IEC4<30>	IPC39<20:18>	IPC39<17:16>	Yes
UART3 転送完了	_UART3_TX_VECTOR	159	OFF159<17:1>	IFS4<31>	IEC4<31>	IPC39<28:26>	IPC39<25:24>	Yes
I2C3 バス衝突イベント	_I2C3_BUS_VECTOR	160	OFF160<17:1>	IFS5<0>	IEC5<0>	IPC40<4:2>	IPC40<1:0>	Yes
I2C3 スレーブイベント	_I2C3_SLAVE_VECTOR	161	OFF161<17:1>	IFS5<1>	IEC5<1>	IPC40<12:10>	IPC40<9:8>	Yes
I2C3 マスタイベント	_I2C3_MASTER_VECTOR	162	OFF162<17:1>	IFS5<2>	IEC5<2>	IPC40<20:18>	IPC40<17:16>	Yes
SPI4 異常	_SPI4_FAULT_VECTOR	163	OFF163<17:1>	IFS5<3>	IEC5<3>	IPC40<28:26>	IPC40<25:24>	Yes
SPI4 受信完了	_SPI4_RX_VECTOR	164	OFF164<17:1>	IFS5<4>	IEC5<4>	IPC41<4:2>	IPC41<1:0>	Yes
SPI4 転送完了	_SPI4_TX_VECTOR	165	OFF165<17:1>	IFS5<5>	IEC5<5>	IPC41<12:10>	IPC41<9:8>	Yes
リアルタイムクロック	_RTCC_VECTOR	166	OFF166<17:1>	IFS5<6>	IEC5<6>	IPC41<20:18>	IPC41<17:16>	No
フラッシュ制御イベント	_FLASH_CONTROL_VECTOR	167	OFF167<17:1>	IFS5<7>	IEC5<7>	IPC41<28:26>	IPC41<25:24>	No

**Note 1:** 全てのデバイスで全ての割り込み要因が使える訳ではありません。デバイスで利用可能な周辺モジュールについてはファミリ機能一覧表 (表 1 ~ 表 2) を参照してください。

**2:** リセット時の GLDC 割り込み (HSYNC と VSYNC) は永続的です。しかし、IRQCON ビット (GLCDINT<31>) を使って割り込みのタイプを非永続的に変更できます。

表 7-2: 割り込み IRQ、ベクタ、ビット位値 (続き)

割り込み要因 <sup>(1)</sup>	XC32 ベクタ名	IRQ #	ベクタ #	割り込みビット位置				永続的 割り込み
				フラグ	イネーブル	優先度	副優先度	
プリフェッチ モジュール SEC イベント	_PREFETCH_VECTOR	168	OFF168<17:1>	IFS5<8>	IEC5<8>	IPC42<4:2>	IPC42<1:0>	Yes
SQI1 イベント	_SQI1_VECTOR	169	OFF169<17:1>	IFS5<9>	IEC5<9>	IPC42<12:10>	IPC42<9:8>	Yes
UART4 異常	_UART4_FAULT_VECTOR	170	OFF170<17:1>	IFS5<10>	IEC5<10>	IPC42<20:18>	IPC42<17:16>	Yes
UART4 受信完了	_UART4_RX_VECTOR	171	OFF171<17:1>	IFS5<11>	IEC5<11>	IPC42<28:26>	IPC42<25:24>	Yes
UART4 転送完了	_UART4_TX_VECTOR	172	OFF172<17:1>	IFS5<12>	IEC5<12>	IPC43<4:2>	IPC43<1:0>	Yes
I2C4 バス衝突イベント	_I2C4_BUS_VECTOR	173	OFF173<17:1>	IFS5<13>	IEC5<13>	IPC43<12:10>	IPC43<9:8>	Yes
I2C4 スレーブイベント	_I2C4_SLAVE_VECTOR	174	OFF174<17:1>	IFS5<14>	IEC5<14>	IPC43<20:18>	IPC43<17:16>	Yes
I2C4 マスタイベント	_I2C4_MASTER_VECTOR	175	OFF175<17:1>	IFS5<15>	IEC5<15>	IPC43<28:26>	IPC43<25:24>	Yes
SPI5 異常	_SPI5_FAULT_VECTOR	176	OFF176<17:1>	IFS5<16>	IEC5<16>	IPC44<4:2>	IPC44<1:0>	Yes
SPI5 受信完了	_SPI5_RX_VECTOR	177	OFF177<17:1>	IFS5<17>	IEC5<17>	IPC44<12:10>	IPC44<9:8>	Yes
SPI5 転送完了	_SPI5_TX_VECTOR	178	OFF178<17:1>	IFS5<18>	IEC5<18>	IPC44<20:18>	IPC44<17:16>	Yes
UART5 異常	_UART5_FAULT_VECTOR	179	OFF179<17:1>	IFS5<19>	IEC5<19>	IPC44<28:26>	IPC44<25:24>	Yes
UART5 受信完了	_UART5_RX_VECTOR	180	OFF180<17:1>	IFS5<20>	IEC5<20>	IPC45<4:2>	IPC45<1:0>	Yes
UART5 転送完了	_UART5_TX_VECTOR	181	OFF181<17:1>	IFS5<21>	IEC5<21>	IPC45<12:10>	IPC45<9:8>	Yes
I2C5 バス衝突イベント	_I2C5_BUS_VECTOR	182	OFF182<17:1>	IFS5<22>	IEC5<22>	IPC45<20:18>	IPC45<17:16>	Yes
I2C5 スレーブイベント	_I2C5_SLAVE_VECTOR	183	OFF183<17:1>	IFS5<23>	IEC5<23>	IPC45<28:26>	IPC45<25:24>	Yes
I2C5 マスタイベント	_I2C5_MASTER_VECTOR	184	OFF184<17:1>	IFS5<24>	IEC5<24>	IPC46<4:2>	IPC46<1:0>	Yes
SPI6 異常	_SPI6_FAULT_VECTOR	185	OFF185<17:1>	IFS5<25>	IEC5<25>	IPC46<12:10>	IPC46<9:8>	Yes
SPI6 受信完了	_SPI6_RX_VECTOR	186	OFF186<17:1>	IFS5<26>	IEC5<26>	IPC46<20:18>	IPC46<17:16>	Yes
SPI6 転送完了	_SPI6_TX_VECTOR	187	OFF187<17:1>	IFS5<27>	IEC5<27>	IPC46<28:26>	IPC46<25:24>	Yes
UART6 異常	_UART6_FAULT_VECTOR	188	OFF188<17:1>	IFS5<28>	IEC5<28>	IPC47<4:2>	IPC47<1:0>	Yes
UART6 受信完了	_UART6_RX_VECTOR	189	OFF189<17:1>	IFS5<29>	IEC5<29>	IPC47<12:10>	IPC47<9:8>	Yes
UART6 転送完了	_UART6_TX_VECTOR	190	OFF190<17:1>	IFS5<30>	IEC5<30>	IPC47<20:18>	IPC47<17:16>	Yes
SDHC 割り込み	_SDHC_VECTOR	191	OFF191<17:1>	IFS5<31>	IEC5<31>	IPC47<28:26>	IPC47<25:24>	Yes
GLCD 割り込み	_GLCD_VECTOR	192	OFF192<17:1>	IFS6<0>	IEC6<0>	IPC48<4:2>	IPC48<1:0>	Yes/No <sup>(2)</sup>
GPU 割り込み	_GPU_VECTOR	193	OFF193<17:1>	IFS6<1>	IEC6<1>	IPC48<12:10>	IPC48<9:8>	Yes
予約済み	—	—	—	—	—	—	—	—
CTMU 割り込み	_CTMU_VECTOR	195	OFF195<17:1>	IFS6<3>	IEC6<3>	IPC48<28:26>	IPC48<25:24>	Yes
ADC スキャン終了	_ADC_EOS_VECTOR	196	OFF196<17:1>	IFS6<4>	IEC6<4>	IPC49<4:2>	IPC49<1:0>	Yes

**Note 1:** 全てのデバイスで全ての割り込み要因が使える訳ではありません。デバイスで利用可能な周辺モジュールについてはファミリー機能一覧表 (表 1 ~ 表 2) を参照してください。

**2:** リセット時の GLDC 割り込み (HSYNC と VSYNC) は永続的です。しかし、IRQCON ビット (GLCDINT<31>) を使って割り込みのタイプを非永続的に変更できます。

表 7-2: 割り込み IRQ、ベクタ、ビット位値 (続き)

割り込み要因 <sup>(1)</sup>	XC32 ベクタ名	IRQ #	ベクタ #	割り込みビット位置				永続的 割り込み
				フラグ	イネーブル	優先度	副優先度	
ADC アナログ回路準備完了	_ADC_ARDY_VECTOR	197	OFF197<17:1>	IFS6<5>	IEC6<5>	IPC49<12:10>	IPC49<9:8>	Yes
ADC 更新準備完了	_ADC_URDY_VECTOR	198	OFF198<17:1>	IFS6<6>	IE6<6>	IPC49<20:18>	IPC49<17:16>	Yes
ADC0 早期割り込み	_ADC0_EARLY_VECTOR	199	OFF199<17:1>	IFS6<7>	IEC6<7>	IPC49<28:26>	IPC49<25:24>	Yes
ADC1 早期割り込み	_ADC1_EARLY_VECTOR	200	OFF200<17:1>	IFS6<8>	IEC6<8>	IPC50<4:2>	IPC50<1:0>	Yes
ADC2 早期割り込み	_ADC2_EARLY_VECTOR	201	OFF201<17:1>	IFS6<9>	IEC6<9>	IPC50<12:10>	IPC50<9:8>	Yes
ADC3 早期割り込み	_ADC3_EARLY_VECTOR	202	OFF202<17:1>	IFS6<10>	IEC6<10>	IPC50<20:18>	IPC50<17:16>	Yes
ADC4 早期割り込み	_ADC4_EARLY_VECTOR	203	OFF203<17:1>	IFS6<11>	IEC6<11>	IPC50<28:26>	IPC50<25:24>	Yes
予約済み	—	—	—	—	—	—	—	—
ADC グループ早期割り込み要求	_ADC_EARLY_VECTOR	205	OFF205<17:1>	IFS6<13>	IEC6<13>	IPC51<12:10>	IPC51<9:8>	Yes
ADC7 早期割り込み	_ADC7_EARLY_VECTOR	206	OFF206<17:1>	IFS6<14>	IEC6<14>	IPC51<20:18>	IPC51<17:16>	Yes
ADC0 Warm 割り込み	_ADC0_WARM_VECTOR	207	OFF207<17:1>	IFS6<15>	IEC6<15>	IPC51<28:26>	IPC51<25:24>	Yes
ADC1 Warm 割り込み	_ADC1_WARM_VECTOR	208	OFF208<17:1>	IFS6<16>	IEC6<16>	IPC52<4:2>	IPC52<1:0>	Yes
ADC2 Warm 割り込み	_ADC2_WARM_VECTOR	209	OFF209<17:1>	IFS6<17>	IEC6<17>	IPC52<12:10>	IPC52<9:8>	Yes
ADC3 Warm 割り込み	_ADC3_WARM_VECTOR	210	OFF210<17:1>	IFS6<18>	IEC6<18>	IPC52<20:18>	IPC52<17:16>	Yes
ADC4 Warm 割り込み	_ADC4_WARM_VECTOR	211	OFF211<17:1>	IFS6<19>	IEC6<19>	IPC52<28:26>	IPC52<25:24>	Yes
予約済み	—	—	—	—	—	—	—	—
予約済み	—	—	—	—	—	—	—	—
ADC7 Warm 割り込み	_ADC7_WARM_VECTOR	214	OFF214<17:1>	IFS6<22>	IEC6<22>	IPC53<20:18>	IPC53<17:16>	Yes
MPLL 異常割り込み	_MPLL_FAULT_VECTOR	215	OFF215<17:1>	IFS6<23>	IEC6<23>	IPC53<28:26>	IPC53<25:24>	Yes

最低自然順序優先度

**Note 1:** 全てのデバイスで全ての割り込み要因が使える訳ではありません。デバイスで利用可能な周辺モジュールについてはファミリー機能一覧表 (表 1 ~ 表 2) を参照してください。

**2:** リセット時の GLDC 割り込み (HSYNC と VSYNC) は永続的です。しかし、IRQCON ビット (GLCDINT<31>) を使って割り込みのタイプを非永続的に変更できます。

### 7.3 割り込み制御レジスタ

表 7-3: 割り込み関連のレジスタマップ

仮想アドレス (BF01_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0000	INTCON	31:16	NMIKEY<7:0>																0000
		15:0	—	—	—	MVEC	—	TPC<2:0>			—	—	—	—	—	—	—	—	—
0010	PRISS	31:16	PRI7SS<3:0>				PRI6SS<3:0>				PRI5SS<3:0>				PRI4SS<3:0>				0000
		15:0	PRI3SS<3:0>				PRI2SS<3:0>				PRI1SS<3:0>				—	—	—	—	SS0
0020	INTSTAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	SRIPL<2:0>				SIRQ<7:0>				0000			
0030	IPTMR	31:16	IPTMR<31:0>																0000
		15:0																	0000
0040	IFS0	31:16	OC6IF	IC6IF	IC6EIF	T6IF	OC5IF	IC5IF	IC5EIF	T5IF	INT4IF	OC4IF	IC4IF	IC4EIF	T4IF	INT3IF	OC3IF	IC3IF	0000
		15:0	IC3EIF	T3IF	INT2IF	OC2IF	IC2IF	IC2EIF	T2IF	INT1IF	OC1IF	IC1IF	IC1EIF	T1IF	INT0IF	CS1IF	CS0IF	CTIF	0000
0050	IFS1	31:16	ADCD4IF	ADCD3IF	ADCD2IF	ADCD1IF	ADCD0IF	ADCFLTIF	ADDCF6IF	ADDCF5IF	ADDCF4IF	ADDCF3IF	ADDCF2IF	ADDCF1IF	ADDC6IF	ADDC5IF	ADDC4IF	ADDC3IF	0000
		15:0	ADDC2IF	ADDC1IF	ADCFIFOIF	ADCIF	OC9IF	IC9IF	IC9EIF	T9IF	OC8IF	IC8IF	IC8EIF	T8IF	OC7IF	IC7IF	IC7EIF	T7IF	0000
0060	IFS2	31:16	ADCD36IF	ADCD35IF	ADCD34IF	ADCD33IF	ADCD32IF	ADCD31IF	ADCD30IF	ADCD29IF	ADCD28IF	ADCD27IF	ADCD26IF	ADCD25IF	ADCD24IF	ADCD23IF	ADCD22IF	ADCD21IF	0000
		15:0	ADCD20IF	ADCD19IF	ADCD18IF	ADCD17IF	ADCD16IF	ADCD15IF	ADCD14IF	ADCD13IF	ADCD12IF	ADCD11IF	ADCD10IF	ADCD9IF	ADCD8IF	ADCD7IF	ADCD6IF	ADCD5IF	0000
0070	IFS3	31:16	CNKIF	CNJIF	CNHIF	CNGIF	CNFIF	CNEIF	CNDIF	CNCIF	CNBIF	CNAIF	I2C1MIF	I2C1SIF	I2C1BIF	U1TXIF	U1RXIF	U1EIF	0000
		15:0	SPI1TXIF	SPI1RXIF	SPI1EIF	—	CRPTIF <sup>(2)</sup>	SBIF	CFDCIF	CPCIF	USBSRIF	ADCD43IF	ADCD42IF	ADCD41IF	ADCD40IF	ADCD39IF	ADCD38IF	ADCD37IF	0000
0080	IFS4	31:16	U3TXIF	U3RXIF	U3EIF	SPI3TXIF	SPI3RXIF	SPI3EIF	ETHIF	CAN2IF	CAN1IF	I2C2MIF	I2C2SIF	I2C2BIF	U2TXIF	U2RXIF	U2EIF	SPI2TXIF	0000
		15:0	SPI2RXIF	SPI2EIF	DMA7IF	DMA6IF	DMA5IF	DMA4IF	DMA3IF	DMA2IF	DMA1IF	DMA0IF	USBDMAIF	USBIF	CMP2IF	CMP1IF	PMPEIF	PMPIF	0000
0090	IFS5	31:16	SDHCIF	U6TXIF	U6RXIF	U6EIF	SPI6TX	SPI6RXIF	SPI6IF	I2C5MIF	I2C5SIF	I2C5BIF	U5TXIF	U5RXIF	U5EIF	SPI5TXIF	SPI5RXIF	SPI5EIF	0000
		15:0	I2C4MIF	I2C4SIF	I2C4BIF	U4TXIF	U4RXIF	U4EIF	SQ11IF	PREIF	FCEIF	RTCCIF	SPI4TXIF	SPI4RXIF	SPI4EIF	I2C3MIF	I2C3SIF	I2C3BIF	0000
00A0	IFS6	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ADC0WIF	ADC7EIF	ADCGRPIF	—	ADC4EIF	ADC3EIF	ADC2EIF	ADC1EIF	ADC0EIF	ADCURDYIF	ADCARDYIF	ADCEOSIF	CTMUIF	—	—	—	—
00C0	IEC0	31:16	OC6IE	IC6IE	IC6EIE	T6IE	OC5IE	IC5IE	IC5EIE	T5IE	INT4IE	OC4IE	IC4IE	IC4EIE	T4IE	INT3IE	OC3IE	IC3IE	0000
		15:0	IC3EIE	T3IE	INT2IE	OC2IE	IC2IE	IC2EIE	T2IE	INT1IE	OC1IE	IC1IE	IC1EIE	T1IE	INT0IE	CS1IE	CS0IE	CTIE	0000
00D0	IEC1	31:16	ADCD4IE	ADCD3IE	ADCD2IE	ADCD1IE	ADCD0IE	ADCFLTIE	ADDCF6IE	ADDCF5IE	ADDCF4IE	ADDCF3IE	ADDCF2IE	ADDCF1IE	ADDC6IE	ADDC5IE	ADDC4IE	ADDC3IE	0000
		15:0	ADDC2IE	ADDC1IE	ADCFIFOIE	ADCIE	OC9IE	IC9IE	IC9EIE	T9IE	OC8IE	IC8IE	IC8EIE	T8IE	OC7IE	IC7IE	IC7EIE	T7IE	0000
00E0	IEC2	31:16	ADCD36IE	ADCD35IE	ADCD34IE	ADCD33IE	ADCD32IE	ADCD31IE	ADCD30IE	ADCD29IE	ADCD28IE	ADCD27IE	ADCD26IE	ADCD25IE	ADCD24IE	ADCD23IE	ADCD22IE	ADCD21IE	0000
		15:0	ADCD20IE	ADCD19IE	ADCD18IE	ADCD17IE	ADCD16IE	ADCD15IE	ADCD14IE	ADCD13IE	ADCD12IE	ADCD11IE	ADCD10IE	ADCD9IE	ADCD8IE	ADCD7IE	ADCD6IE	ADCD5IE	0000
00F0	IEC3	31:16	CNKIE	CNJIE	CNHIE	CNGIE	CNFIE	CNEIE	CNDIE	CNCIE	CNBIE	CNAIE	I2C1MIE	I2C1SIE	I2C1BIE	U1TXIE	U1RXIE	U1EIE	0000
		15:0	SPI1TXIE	SPI1RXIE	SPI1EIE	—	CRPTIE <sup>(2)</sup>	SBIE	CFDCIE	CPCIE	USBSRIE	ADCD43IE	ADCD42IE	ADCD41IE	ADCD40IE	ADCD39IE	ADCD38IE	ADCD37IE	0000
0100	IEC4	31:16	U3TXIE	U3RXIE	U3EIE	SPI3RXIE	SPI3EIE	ETHIE	CAN2IE	CAN1IE	I2C2MIE	I2C2SIE	I2C2BIE	U2TXIE	U2RXIE	U2EIE	SPI2TXIE	0000	
		15:0	SPI2RXIE	SPI2EIE	DMA7IE	DMA6IE	DMA5IE	DMA4IE	DMA3IE	DMA2IE	DMA1IE	DMA0IE	USBDMAIE	USBIE	CMP2IE	CMP1IE	PMPEIE	PMPIE	0000
0110	IEC5	31:16	SDHCIE	U6TXIE	U6RXIE	U6EIE	SPI6TXIE	SPI6RXIE	SPI6IE	I2C5MIE	I2C5SIE	I2C5BIE	U5TXIE	U5RXIE	U5EIE	SPI5TXIE	SPI5RXIE	SPI5EIE	0000
		15:0	I2C4MIE	I2C4SIE	I2C4BIE	U4TXIE	U4RXIE	U4EIE	SQ11IE	PREIE	FCEIE	RTCCIE	SPI4TXIE	SPI4RXIE	SPI4EIE	I2C3MIE	I2C3SIE	I2C3BIE	0000
0120	IEC6	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ADC0WIE	ADC7EIE	ADCGRPIE	—	ADC4EIE	ADC3EIE	ADC2EIE	ADC1EIE	ADC0EIE	ADCURDYIE	ADCARDYIE	ADCEOSIE	CTMUIE	—	—	—	—

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0		
0140	IPC0	31:16	—	—	—	—	INT0IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	INT0IS<1:0>	0000
		15:0	—	—	—	—	CS0IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	CS0IS<1:0>	0000
0150	IPC1	31:16	—	—	—	—	OC1IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	OC1IS<1:0>	0000
		15:0	—	—	—	—	IC1EIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	IC1EIS<1:0>	0000
0160	IPC2	31:16	—	—	—	—	IC2IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	IC2EIS<1:0>	0000
		15:0	—	—	—	—	T2IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	INT1IP<2:0>	0000
0170	IPC3	31:16	—	—	—	—	IC3EIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	T3IP<2:0>	0000
		15:0	—	—	—	—	INT2IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	INT2IS<1:0>	0000
0180	IPC4	31:16	—	—	—	—	T4IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	INT3IP<2:0>	0000
		15:0	—	—	—	—	OC3IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	OC3IS<1:0>	0000
0190	IPC5	31:16	—	—	—	—	INT4IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	INT4IS<1:0>	0000
		15:0	—	—	—	—	IC4IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	IC4EIS<1:0>	0000
01A0	IPC6	31:16	—	—	—	—	OC5IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	OC5IS<1:0>	0000
		15:0	—	—	—	—	IC5EIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	IC5EIS<1:0>	0000
01B0	IPC7	31:16	—	—	—	—	OC6IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	OC6IS<1:0>	0000
		15:0	—	—	—	—	IC6EIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	IC6EIS<1:0>	0000
01C0	IPC8	31:16	—	—	—	—	OC7IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	OC7IS<1:0>	0000
		15:0	—	—	—	—	IC7EIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	IC7EIS<1:0>	0000
01D0	IPC9	31:16	—	—	—	—	OC8IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	OC8IS<1:0>	0000
		15:0	—	—	—	—	IC8EIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	IC8EIS<1:0>	0000
01E0	IPC10	31:16	—	—	—	—	OC9IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	OC9IS<1:0>	0000
		15:0	—	—	—	—	IC9EIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	IC9EIS<1:0>	0000
01F0	IPC11	31:16	—	—	—	—	ADCDC2IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCDC2IS<1:0>	0000
		15:0	—	—	—	—	ADCFIFOIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCFIFOIS<1:0>	0000
0200	IPC12	31:16	—	—	—	—	ADCDC6IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCDC6IS<1:0>	0000
		15:0	—	—	—	—	ADCDC4IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCDC4IS<1:0>	0000
0210	IPC13	31:16	—	—	—	—	ADDCF4IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADDCF4IS<1:0>	0000
		15:0	—	—	—	—	ADDCF2IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADDCF2IS<1:0>	0000
0220	IPC14	31:16	—	—	—	—	ADCD0IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCD0IS<1:0>	0000
		15:0	—	—	—	—	ADCDF6IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCDF6IS<1:0>	0000
0230	IPC15	31:16	—	—	—	—	ADCD4IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCD4IS<1:0>	0000
		15:0	—	—	—	—	ADCD2IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCD2IS<1:0>	0000
0240	IPC16	31:16	—	—	—	—	ADCD8IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCD8IS<1:0>	0000
		15:0	—	—	—	—	ADCD6IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCD6IS<1:0>	0000
0250	IPC17	31:16	—	—	—	—	ADCD12IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCD12IS<1:0>	0000
		15:0	—	—	—	—	ADCD10IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCD10IS<1:0>	0000
0260	IPC18	31:16	—	—	—	—	ADCD16IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCD16IS<1:0>	0000
		15:0	—	—	—	—	ADCD14IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	ADCD14IS<1:0>	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。  
 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	
0270	IPC19	31:16	—	—	—	ADCD20IP<2:0>	ADCD20IS<1:0>	—	—	—	—	ADCD19IP<2:0>	ADCD19IS<1:0>	0000			
		15:0	—	—	—	ADCD18IP<2:0>	ADCD18IS<1:0>	—	—	—	—	ADCD17IP<2:0>	ADCD17IS<1:0>	0000			
0280	IPC20	31:16	—	—	—	ADCD24IP<2:0>	ADCD24IS<1:0>	—	—	—	—	ADCD23IP<2:0>	ADCD23IS<1:0>	0000			
		15:0	—	—	—	ADCD22IP<2:0>	ADCD22IS<1:0>	—	—	—	—	ADCD21IP<2:0>	ADCD21IS<1:0>	0000			
0290	IPC21	31:16	—	—	—	ADCD28IP<2:0>	ADCD28IS<1:0>	—	—	—	—	ADCD27IP<2:0>	ADCD27IS<1:0>	0000			
		15:0	—	—	—	ADCD26IP<2:0>	ADCD26IS<1:0>	—	—	—	—	ADCD25IP<2:0>	ADCD25IS<1:0>	0000			
02A0	IPC22	31:16	—	—	—	ADCD32IP<2:0>	ADCD32IS<1:0>	—	—	—	—	ADCD31IP<2:0>	ADCD31IS<1:0>	0000			
		15:0	—	—	—	ADCD30IP<2:0>	ADCD30IS<1:0>	—	—	—	—	ADCD29IP<2:0>	ADCD29IS<1:0>	0000			
02B0	IPC23	31:16	—	—	—	ADCD36IP<2:0>	ADCD36IS<1:0>	—	—	—	—	ADCD35IP<2:0>	ADCD35IS<1:0>	0000			
		15:0	—	—	—	ADCD34IP<2:0>	ADCD34IS<1:0>	—	—	—	—	ADCD33IP<2:0>	ADCD33IS<1:0>	0000			
02C0	IPC24	31:16	—	—	—	ADCD40IP<2:0>	ADCD40IS<1:0>	—	—	—	—	ADCD39IP<2:0>	ADCD39IS<1:0>	0000			
		15:0	—	—	—	ADCD38IP<2:0>	ADCD38IS<1:0>	—	—	—	—	ADCD37IP<2:0>	ADCD37IS<1:0>	0000			
02D0	IPC25	31:16	—	—	—	USBSRIP<2:0>	USBSRIS<1:0>	—	—	—	—	ADCD43IP<2:0>	ADCD43IS<1:0>	0000			
		15:0	—	—	—	ADCD42IP<2:0>	ADCD42IS<1:0>	—	—	—	—	ADCD41IP<2:0>	ADCD41IS<1:0>	0000			
02E0	IPC26	31:16	—	—	—	CRPTIP<2:0> <sup>(2)</sup>	CRPTIS<1:0> <sup>(2)</sup>	—	—	—	—	SBIP<2:0>	SBIS<1:0>	0000			
		15:0	—	—	—	CFDCIP<2:0>	CFDCIS<1:0>	—	—	—	—	GPCIP<2:0>	CPCIS<1:0>	0000			
02F0	IPC27	31:16	—	—	—	SPI1TXIP<2:0>	SPI1TXIS<1:0>	—	—	—	—	SPI1RXIP<2:0>	SPI1RXIS<1:0>	0000			
		15:0	—	—	—	SPI1EIP<2:0>	SPI1EIS<1:0>	—	—	—	—	—	—	—	0000		
0300	IPC28	31:16	—	—	—	I2C1BIP<2:0>	I2C1BIS<1:0>	—	—	—	—	U1TXIP<2:0>	U1TXIS<1:0>	0000			
		15:0	—	—	—	U1RXIP<2:0>	U1RXIS<1:0>	—	—	—	—	U1EIP<2:0>	U1EIS<1:0>	0000			
0310	IPC29	31:16	—	—	—	CNBIP<2:0>	CNBIS<1:0>	—	—	—	—	CNAIP<2:0>	CNAIS<1:0>	0000			
		15:0	—	—	—	I2C1MIP<2:0>	I2C1MIS<1:0>	—	—	—	—	I2C1SIP<2:0>	I2C1SIS<1:0>	0000			
0320	IPC30	31:16	—	—	—	CNFIP<2:0>	CNFIS<1:0>	—	—	—	—	CNEIP<2:0>	CNEIS<1:0>	0000			
		15:0	—	—	—	CNDIP<2:0>	CNDIS<1:0>	—	—	—	—	CNCIP<2:0>	CNCIS<1:0>	0000			
0330	IPC31	31:16	—	—	—	CNKIP<2:0>	CNKIS<1:0>	—	—	—	—	CNJIP<2:0>	CNJIS<1:0>	0000			
		15:0	—	—	—	CNHIP<2:0>	CNHIS<1:0>	—	—	—	—	CNGIP<2:0>	CNGIS<1:0>	0000			
0340	IPC32	31:16	—	—	—	CMP2IP<2:0>	CMP2IS<1:0>	—	—	—	—	CMP1IP<2:0>	CMP1IS<1:0>	0000			
		15:0	—	—	—	PMPEIP<2:0>	PMPEIS<1:0>	—	—	—	—	PMPIP<2:0>	PMPIS<1:0>	0000			
0350	IPC33	31:16	—	—	—	DMA1IP<2:0>	DMA1IS<1:0>	—	—	—	—	DMA0IP<2:0>	DMA0IS<1:0>	0000			
		15:0	—	—	—	USBDMAIP<2:0>	USBDMASIS<1:0>	—	—	—	—	USBIP<2:0>	USBIS<1:0>	0000			
0360	IPC34	31:16	—	—	—	DMA5IP<2:0>	DMA5IS<1:0>	—	—	—	—	DMA4IP<2:0>	DMA4IS<1:0>	0000			
		15:0	—	—	—	DMA3IP<2:0>	DMA3IS<1:0>	—	—	—	—	DMA2IP<2:0>	DMA2IS<1:0>	0000			
0370	IPC35	31:16	—	—	—	SPI2RXIP<2:0>	SPI2RXIS<1:0>	—	—	—	—	SPI2EIP<2:0>	SPI2EIS<1:0>	0000			
		15:0	—	—	—	DMA7IP<2:0>	DMA7IS<1:0>	—	—	—	—	DMA6IP<2:0>	DMA6IS<1:0>	0000			
0380	IPC36	31:16	—	—	—	U2TXIP<2:0>	U2TXIS<1:0>	—	—	—	—	U2RXIP<2:0>	U2RXIS<1:0>	0000			
		15:0	—	—	—	U2EIP<2:0>	U2EIS<1:0>	—	—	—	—	SPI2TXIP<2:0>	SPI2TXIS<1:0>	0000			
0390	IPC37	31:16	—	—	—	CAN1IP<2:0>	CAN1IS<1:0>	—	—	—	—	I2C2MIP<2:0>	I2C2MIS<1:0>	0000			
		15:0	—	—	—	I2C2SIP<2:0>	I2C2SIS<1:0>	—	—	—	—	I2C2BIP<2:0>	I2C2BIS<1:0>	0000			

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

- Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。
- 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。



表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット			
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0	
03A0	IPC38	31:16	—	—	—	SPI3RXIP<2:0>			SPI3RXIS<1:0>			—	—	—	SPI3EIP<2:0>			SPI3EIS<1:0>		0000
		15:0	—	—	—	ETHIP<2:0>			ETHIS<1:0>			—	—	—	CAN2IP<2:0>			CAN2IS<1:0>		0000
03B0	IPC39	31:16	—	—	—	U3TXIP<2:0>			U3TXIS<1:0>			—	—	—	U3RXIP<2:0>			U3RXIS<1:0>		0000
		15:0	—	—	—	U3EIP<2:0>			U3EIS<1:0>			—	—	—	SPI3TXIP<2:0>			SPI3TXIS<1:0>		0000
03C0	IPC40	31:16	—	—	—	SPI4EIP<2:0>			SPI4EIS<1:0>			—	—	—	I2C3MIP<2:0>			I2C3MIS<1:0>		0000
		15:0	—	—	—	I2C3SIP<2:0>			I2C3SIS<1:0>			—	—	—	I2C3BIP<2:0>			I2C3BIS<1:0>		0000
03D0	IPC41	31:16	—	—	—	FCEIP<2:0>			FCEIS<1:0>			—	—	—	RTCCIP<2:0>			RTCCIS<1:0>		0000
		15:0	—	—	—	SPI4TXIP<2:0>			SPI4TXIS<1:0>			—	—	—	SPI4RXIP<2:0>			SPI4RXIS<1:0>		0000
03E0	IPC42	31:16	—	—	—	U4RXIP<2:0>			U4RXIS<1:0>			—	—	—	U4EIP<2:0>			U4EIS<1:0>		0000
		15:0	—	—	—	SQI1IP<2:0>			SQI1IS<1:0>			—	—	—	PREIP<2:0>			PREIS<1:0>		0000
03F0	IPC43	31:16	—	—	—	I2C4MIP<2:0>			I2C4MIS<1:0>			—	—	—	I2C4SIP<2:0>			I2C4SIS<1:0>		0000
		15:0	—	—	—	I2C4BIP<2:0>			I2C4BIS<1:0>			—	—	—	U4TXIP<2:0>			U4TXIS<1:0>		0000
0400	IPC44	31:16	—	—	—	U5EIP<2:0>			U5EIS<1:0>			—	—	—	SPI5TXIP<2:0>			SPI5TXIS<1:0>		0000
		15:0	—	—	—	SPI5RXIP<2:0>			SPI5RXIS<1:0>			—	—	—	SPI5EIP<2:0>			SPI5EIS<1:0>		0000
0410	IPC45	31:16	—	—	—	I2C5SIP<2:0>			I2C5SIS<1:0>			—	—	—	I2C5BIP<2:0>			I2C5BIS<1:0>		0000
		15:0	—	—	—	U5TXIP<2:0>			U5TXIS<1:0>			—	—	—	U5RXIP<2:0>			U5RXIS<1:0>		0000
0420	IPC46	31:16	—	—	—	SPI6TXIP<2:0>			SPI6TXIS<1:0>			—	—	—	SPI6RXIP<2:0>			SPI6RXIS<1:0>		0000
		15:0	—	—	—	SPI6EIP<2:0>			SPI6EIS<1:0>			—	—	—	I2C5MIP<2:0>			I2C5MIS<1:0>		0000
0430	IPC47	31:16	—	—	—	SDHCIP<2:0>			SDHCIS<1:0>			—	—	—	U6TXIP<2:0>			U6TXIS<1:0>		0000
		15:0	—	—	—	U6RXIP<2:0>			U6RXIS<1:0>			—	—	—	U6EIP<2:0>			U6EIS<1:0>		0000
0440	IPC48	31:16	—	—	—	CTMU1IP<2:0>			CTMU1IS<1:0>			—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	GPUIP<2:0>			GPU1IS<1:0>			—	—	—	GLCDIP<2:0>			GLCDIS<1:0>		0000
0450	IPC49	31:16	—	—	—	ADC0EIP<2:0>			ADC0EIS<1:0>			—	—	—	ADCURDYIP<2:0>			ADCURDYIS<1:0>		0000
		15:0	—	—	—	ADCARDYIP<2:0>			ADCARDYIS<1:0>			—	—	—	ADCEOSIP<2:0>			ADCEOSIS<1:0>		0000
0460	IPC50	31:16	—	—	—	ADC4EIP<2:0>			ADC4EIS<1:0>			—	—	—	ADC3EIP<2:0>			ADC3EIS<1:0>		0000
		15:0	—	—	—	ADC2EIP<2:0>			ADC2EIS<1:0>			—	—	—	ADC1EIP<2:0>			ADC1EIS<1:0>		0000
0470	IPC51	31:16	—	—	—	ADC0WIP<2:0>			ADC0WIS<1:0>			—	—	—	ADC7EIP<2:0>			ADC7EIS<1:0>		0000
		15:0	—	—	—	ADCGRPIP<2:0>			ADCGRPIS<1:0>			—	—	—	—	—	—	—	—	0000
0480	IPC52	31:16	—	—	—	ADC4WIP<2:0>			ADC4WIS<1:0>			—	—	—	ADC3WIP<2:0>			ADC3WIS<1:0>		0000
		15:0	—	—	—	ADC2WIP<2:0>			ADC2WIS<1:0>			—	—	—	ADC1WIP<2:0>			ADC1WIS<1:0>		0000
0490	IPC53	31:16	—	—	—	MPLLFLTIP<2:0>			MPLLFLTIS<1:0>			—	—	—	ADC7WIP<2:0>			ADC7WIS<1:0>		0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
0540	OFF000	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000	
		15:0	VOFF<15:1>														—	0000		
0544	OFF001	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000	
		15:0	VOFF<15:1>														—	0000		
0548	OFF002	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000	
		15:0	VOFF<15:1>														—	0000		

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。  
 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
054C	OFF003	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0550	OFF004	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0554	OFF005	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0558	OFF006	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
055C	OFF007	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0560	OFF008	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0564	OFF009	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0568	OFF010	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
056C	OFF011	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0570	OFF012	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0574	OFF013	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0578	OFF014	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
057C	OFF015	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0580	OFF016	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0584	OFF017	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0588	OFF018	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
058C	OFF019	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0590	OFF020	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0594	OFF021	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
**Note** 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。  
 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
0598	OFF022	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
059C	OFF023	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05A0	OFF024	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05A4	OFF025	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05A8	OFF026	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05AC	OFF027	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05B0	OFF028	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05B4	OFF029	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05B8	OFF030	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05BC	OFF031	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05C0	OFF032	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05C4	OFF033	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05C8	OFF034	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05CC	OFF035	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05D0	OFF036	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05D4	OFF037	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05D8	OFF038	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05DC	OFF039	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
05E0	OFF040	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

- Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。
- 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
05E4	OFF041	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
05E8	OFF042	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
05EC	OFF043	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
05F0	OFF044	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
05F4	OFF045	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
05F8	OFF046	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
05FC	OFF047	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0600	OFF048	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0604	OFF049	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0608	OFF059	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
060C	OFF051	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0610	OFF052	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0614	OFF053	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0618	OFF054	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
061C	OFF055	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0620	OFF056	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0624	OFF057	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0628	OFF058	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
062C	OFF059	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。  
 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
0630	OFF060	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0634	OFF061	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0638	OFF062	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
063C	OFF063	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0640	OFF064	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0644	OFF065	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0648	OFF066	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
064C	OFF067	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0650	OFF068	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0654	OFF069	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0658	OFF070	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
065C	OFF071	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0660	OFF072	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0664	OFF073	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0668	OFF074	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
066C	OFF075	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0670	OFF076	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0674	OFF077	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0678	OFF078	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

- Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。
- 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 (1)	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
067C	OFF079	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0680	OFF080	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0684	OFF081	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0688	OFF082	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
068C	OFF083	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0690	OFF084	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0694	OFF085	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0698	OFF086	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
069C	OFF087	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06A0	OFF088	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06A4	OFF089	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06A8	OFF090	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06AC	OFF091	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06B0	OFF092	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06B4	OFF093	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06B8	OFF094	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06BC	OFF095	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06C0	OFF096	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
06C4	OFF097	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。  
 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
06C8	OFF098	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06CC	OFF099	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06D0	OFF100	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06D4	OFF101	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06D8	OFF102	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06DC	OFF103	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06E0	OFF104	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06E4	OFF105	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06E8	OFF106	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06EC	OFF107	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06F4	OFF109	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06F8	OFF110	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
06FC	OFF111	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0700	OFF112	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0704	OFF113	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0708	OFF114	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
070C	OFF115	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0710	OFF116	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0714	OFF117	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

- Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。
- 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF81.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
0718	OFF118	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
071C	OFF119	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0720	OFF120	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0724	OFF121	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0728	OFF122	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
072C	OFF123	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0730	OFF124	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0734	OFF125	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0738	OFF126	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
073C	OFF127	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0740	OFF128	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0744	OFF129	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0748	OFF130	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
074C	OFF131	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0750	OFF132	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0754	OFF133	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0758	OFF134	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
075C	OFF135	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0760	OFF136	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。  
 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。



表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
0764	OFF137	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0768	OFF138	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
076C	OFF139	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0770	OFF140	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0774	OFF141	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0778	OFF142	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
077C	OFF143	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0780	OFF144	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0784	OFF145	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0788	OFF146	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
078C	OFF147	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0790	OFF148	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0794	OFF149	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
0798	OFF150	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
079C	OFF151	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
07A0	OFF152	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
07A4	OFF153	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
07A8	OFF154	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000
07AC	OFF155	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

- Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。
- 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF81.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
07B0	OFF156	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07B4	OFF157	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07B8	OFF158	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07BC	OFF159	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07C0	OFF160	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07C4	OFF161	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07C8	OFF162	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07CC	OFF163	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07D0	OFF164	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07D4	OFF165	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07D8	OFF166	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07DC	OFF167	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07E0	OFF168	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07E4	OFF169	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07E8	OFF170	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07EC	OFF171	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07F0	OFF172	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07F4	OFF173	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
07F8	OFF174	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。  
 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
07FC	OFF175	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0800	OFF176	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0804	OFF177	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0808	OFF178	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
080C	OFF179	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0810	OFF180	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0814	OFF181	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0818	OFF182	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
081C	OFF183	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0820	OFF184	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0824	OFF185	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0828	OFF186	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
082C	OFF187	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0830	OFF188	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0834	OFF189	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0838	OFF190	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
083C	OFF191	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0840	OFF192	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0844	OFF193	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

- Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。
- 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

表 7-3: 割り込み関連のレジスタマップ (続き)

仮想アドレス (BF01.#)	レジスタ名 (1)	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
084C	OFF195	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0850	OFF196	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0854	OFF197	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0858	OFF198	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
085C	OFF199	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0860	OFF200	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0864	OFF201	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0868	OFF202	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
086C	OFF203	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0874	OFF205	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0878	OFF206	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
087C	OFF207	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0880	OFF208	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0884	OFF209	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0888	OFF210	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
08A4	OFF211	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
0898	OFF214	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	
089C	OFF215	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOFF<17:16>	0000
		15:0	VOFF<15:1>														—	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の OFFx レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

Note 2: このビットは暗号モジュールを備えたデバイスでのみ使えます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 7-1: INTCON: 割り込み制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NMIKEY<7:0>								
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	MVEC	—	TPC<2:0>		
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-24 **NMIKEY<7:0>**: ノンマスカブル割り込みキービット  
 正しいキー (0x4E) を書き込むと、ソフトウェア NMI が生成されます。ステータスは GNMI ビット (RNMICON<19>) で示されます。

bit 23-13 **未実装**: 「0」として読み出し

bit 12 **MVEC**: マルチベクタ コンフィグレーション ビット

1 = 割り込みコントローラをマルチベクタ モードに設定する  
 0 = 割り込みコントローラをシングルベクタ モードに設定する

bit 11 **未実装**: 「0」として読み出し

bit 10-8 **TPC<2:0>**: 割り込み近接タイマ制御ビット

111 = グループ優先度 7 以下の割り込みが割り込み近接タイマをトリガする  
 110 = グループ優先度 6 以下の割り込みが割り込み近接タイマをトリガする  
 101 = グループ優先度 5 以下の割り込みが割り込み近接タイマをトリガする  
 100 = グループ優先度 4 以下の割り込みが割り込み近接タイマをトリガする  
 011 = グループ優先度 3 以下の割り込みが割り込み近接タイマをトリガする  
 010 = グループ優先度 2 以下の割り込みが割り込み近接タイマをトリガする  
 001 = グループ優先度 1 の割り込みが割り込み近接タイマをトリガする  
 000 = 割り込み近接タイマを無効にする

bit 7-5 **未実装**: 「0」として読み出し

bit 4 **INT4EP**: 外部割り込み 4 エッジ極性制御ビット

1 = 立ち上がりエッジ  
 0 = 立ち下がりエッジ

bit 3 **INT3EP**: 外部割り込み 3 エッジ極性制御ビット

1 = 立ち上がりエッジ  
 0 = 立ち下がりエッジ

bit 2 **INT2EP**: 外部割り込み 2 エッジ極性制御ビット

1 = 立ち上がりエッジ  
 0 = 立ち下がりエッジ

bit 1 **INT1EP**: 外部割り込み 1 エッジ極性制御ビット

1 = 立ち上がりエッジ  
 0 = 立ち下がりエッジ

bit 0 **INT0EP**: 外部割り込み 0 エッジ極性制御ビット

1 = 立ち上がりエッジ  
 0 = 立ち下がりエッジ

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 7-2: PRISS: 優先度シャドールート選択レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PRI7SS<3:0> <sup>(1)</sup>				PRI6SS<3:0> <sup>(1)</sup>			
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PRI5SS<3:0> <sup>(1)</sup>				PRI4SS<3:0> <sup>(1)</sup>			
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PRI3SS<3:0>				PRI2SS<3:0> <sup>(1)</sup>			
7:0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0
	PRI1SS<3:0> <sup>(1)</sup>				—	—	—	SS0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-28 **PRI7SS<3:0>**: 優先度 7 割り込みシャドールートセット ビット <sup>(1)</sup>

1xxx = 予約済み (既定値により、優先度 7 の割り込みはシャドールートセット 0 を使います)  
 0111 = 優先度 7 の割り込みはシャドールートセット 7 を使う  
 0110 = 優先度 7 の割り込みはシャドールートセット 6 を使う  
 .  
 .  
 0001 = 優先度 7 の割り込みはシャドールートセット 1 を使う  
 0000 = 優先度 7 の割り込みはシャドールートセット 0 を使う

bit 27-24 **PRI6SS<3:0>**: 優先度 6 割り込みシャドールートセット ビット <sup>(1)</sup>

1xxx = 予約済み (既定値により、優先度 6 の割り込みはシャドールートセット 0 を使います)  
 0111 = 優先度 6 の割り込みはシャドールートセット 7 を使う  
 0110 = 優先度 6 の割り込みはシャドールートセット 6 を使う  
 .  
 .  
 0001 = 優先度 6 の割り込みはシャドールートセット 1 を使う  
 0000 = 優先度 6 の割り込みはシャドールートセット 0 を使う

bit 23-20 **PRI5SS<3:0>**: 優先度 5 割り込みシャドールートセット ビット <sup>(1)</sup>

1xxx = 予約済み (既定値により、優先度 5 の割り込みはシャドールートセット 0 を使います)  
 0111 = 優先度 5 の割り込みはシャドールートセット 7 を使う  
 0110 = 優先度 5 の割り込みはシャドールートセット 6 を使う  
 .  
 .  
 0001 = 優先度 5 の割り込みはシャドールートセット 1 を使う  
 0000 = 優先度 5 の割り込みはシャドールートセット 0 を使う

bit 19-16 **PRI4SS<3:0>**: 優先度 4 割り込みシャドールートセット ビット <sup>(1)</sup>

1xxx = 予約済み (既定値により、優先度 4 の割り込みはシャドールートセット 0 を使います)  
 0111 = 優先度 4 の割り込みはシャドールートセット 7 を使う  
 0110 = 優先度 4 の割り込みはシャドールートセット 6 を使う  
 .  
 .  
 0001 = 優先度 4 の割り込みはシャドールートセット 1 を使う  
 0000 = 優先度 4 の割り込みはシャドールートセット 0 を使う

**Note 1:** MVEC ビット (INTCON<12>) = 0 の場合、これらのビットは無視されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 7-2: PRISS: 優先度シャドー選択レジスタ (続き)

- bit 15-12 **PRI3SS<3:0>**: 優先度 3 割り込みシャドーセット ビット<sup>(1)</sup>
- 1xxx = 予約済み (既定値により、優先度 3 の割り込みはシャドーセット 0 を使います)
  - 0111 = 優先度 3 の割り込みはシャドーセット 7 を使う
  - 0110 = 優先度 3 の割り込みはシャドーセット 6 を使う
  - ⋮
  - 0001 = 優先度 3 の割り込みはシャドーセット 1 を使う
  - 0000 = 優先度 3 の割り込みはシャドーセット 0 を使う
- bit 11-8 **PRI2SS<3:0>**: 優先度 2 割り込みシャドーセット ビット<sup>(1)</sup>
- 1xxx = 予約済み (既定値により、優先度 2 の割り込みはシャドーセット 0 を使います)
  - 0111 = 優先度 2 の割り込みはシャドーセット 7 を使う
  - 0110 = 優先度 2 の割り込みはシャドーセット 6 を使う
  - ⋮
  - 0001 = 優先度 2 の割り込みはシャドーセット 1 を使う
  - 0000 = 優先度 2 の割り込みはシャドーセット 0 を使う
- bit 7-4 **PRI1SS<3:0>**: 優先度 1 割り込みシャドーセット ビット<sup>(1)</sup>
- 1xxx = 予約済み (既定値により、優先度 1 の割り込みはシャドーセット 0 を使います)
  - 0111 = 優先度 1 の割り込みはシャドーセット 7 を使う
  - 0110 = 優先度 1 の割り込みはシャドーセット 6 を使う
  - ⋮
  - 0001 = 優先度 1 の割り込みはシャドーセット 1 を使う
  - 0000 = 優先度 1 の割り込みはシャドーセット 0 を使う
- bit 3-1 **未実装**: 「0」として読み出し
- bit 0 **SS0**: シングルベクタ シャドーレジスタ セットビット
- 1 = シングルベクタを提示するためにシャドーセットを使う
  - 0 = シングルベクタを提示するためにシャドーセットを使わない

**Note 1:** MVEC ビット (INTCON<12>) = 0 の場合、これらのビットは無視されます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 7-3: INTSTAT: 割り込みステータス

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
	—	—	—	—	—	SRIPL<2:0> <sup>(1)</sup>		
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	SIRQ<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-11 **未実装:** 「0」として読み出し

bit 10-8 **SRIPL<2:0>:** シングルベクタ モード向け要求優先度ビット<sup>(1)</sup>

111-000 = 直近に CPU に提示された割り込みの優先度

bit 7-6 **未実装:** 「0」として読み出し

bit 7-0 **SIRQ<7:0>:** 直近割り込み要求処理ステータスビット

11111111-00000000 = 直近に CPU が処理した割り込み要求の番号

**Note 1:** この値は、割り込みコントローラをシングルベクタ モード向けに設定した場合にのみ使われます。

レジスタ 7-4: IPTMR: 割り込み近接タイマレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IPTMR<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IPTMR<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IPTMR<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IPTMR<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **IPTMR<31:0>:** 割り込み近接タイマリロード ビット

割り込み近接タイマは、割り込みイベントによってトリガされた時に、この値をリロード値として使います。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 7-5: IFSx: 割り込みフラグ ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IFS31	IFS30	IFS29	IFS28	IFS27	IFS26	IFS25	IFS24
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IFS23	IFS22	IFS21	IFS20	IFS19	IFS18	IFS17	IFS16
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IFS15	IFS14	IFS13	IFS12	IFS11	IFS10	IFS9	IFS8
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IFS7	IFS6	IFS5	IFS4	IFS3	IFS2	IFS1	IFS0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **IFS31-IFS0:** 割り込みフラグ ステータスビット

1 = 割り込み要求が発生した  
 0 = 割り込み要求は発生していない

**Note:** ここには IFSx レジスタの一般的な定義しか記載していません。ビットのエンコードについては表 7-2 を参照してください。

レジスタ 7-6: IECx: 割り込みイネーブル制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IEC31	IEC30	IEC29	IEC28	IEC27	IEC26	IEC25	IEC24
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IEC23	IEC22	IEC21	IEC20	IEC19	IEC18	IEC17	IEC16
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IEC15	IEC14	IEC13	IEC12	IEC11	IEC10	IEC9	IEC8
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	IEC7	IEC6	IEC5	IEC4	IEC3	IEC2	IEC1	IEC0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **IEC31-IEC0:** 割り込みイネーブルビット

1 = 割り込みを有効にする  
 0 = 割り込みを無効にする

**Note:** ここには IECx レジスタの一般的な定義しか記載していません。ビットのエンコードについては表 7-2 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 7-7: IPCx: 割り込み優先度制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	IP3<2:0>			IS3<1:0>	
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	IP2<2:0>			IS2<1:0>	
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	IP1<2:0>			IS1<1:0>	
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	IP0<2:0>			IS0<1:0>	

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28-26 **IP3<2:0>**: 割り込み優先度ビット

111 = 割り込み優先度は 7  
 .  
 .

010 = 割り込み優先度は 2  
 001 = 割り込み優先度は 1  
 000 = 割り込みを無効にする

bit 25-24 **IS3<1:0>**: 割り込み副優先度ビット

11 = 割り込み副優先度は 3  
 10 = 割り込み副優先度は 2  
 01 = 割り込み副優先度は 1  
 00 = 割り込み副優先度は 0

bit 23-21 **未実装**: 「0」として読み出し

bit 20-18 **IP2<2:0>**: 割り込み優先度ビット

111 = 割り込み優先度は 7  
 .  
 .

010 = 割り込み優先度は 2  
 001 = 割り込み優先度は 1  
 000 = 割り込みを無効にする

bit 17-16 **IS2<1:0>**: 割り込み副優先度ビット

11 = 割り込み副優先度は 3  
 10 = 割り込み副優先度は 2  
 01 = 割り込み副優先度は 1  
 00 = 割り込み副優先度は 0

bit 15-13 **未実装**: 「0」として読み出し

**Note:** ここには IPCx レジスタの一般的な定義しか記載していません。ビットのエンコードについては表 7-2 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 7-7: IPCx: 割り込み優先度制御レジスタ (続き)

- bit 12-10 **IP1<2:0>**: 割り込み優先度ビット
- 111 = 割り込み優先度は 7
  - ⋮
  - 010 = 割り込み優先度は 2
  - 001 = 割り込み優先度は 1
  - 000 = 割り込みを無効にする
- bit 9-8 **IS1<1:0>**: 割り込み副優先度ビット
- 11 = 割り込み副優先度は 3
  - 10 = 割り込み副優先度は 2
  - 01 = 割り込み副優先度は 1
  - 00 = 割り込み副優先度は 0
- bit 7-5 **未実装**: 「0」として読み出し
- bit 4-2 **IP0<2:0>**: 割り込み優先度ビット
- 111 = 割り込み優先度は 7
  - ⋮
  - 010 = 割り込み優先度は 2
  - 001 = 割り込み優先度は 1
  - 000 = 割り込みを無効にする
- bit 1-0 **IS0<1:0>**: 割り込み副優先度ビット
- 11 = 割り込み副優先度は 3
  - 10 = 割り込み副優先度は 2
  - 01 = 割り込み副優先度は 1
  - 00 = 割り込み副優先度は 0

**Note:** ここには IPCx レジスタの一般的な定義しか記載していません。ビットのエンコードについては[表 7-2](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 7-8: OFFx: 割り込みベクタアドレス オフセット レジスタ (x = 0 ~ 190)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	VOFF<17:16>	
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	VOFF<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
	VOFF<7:1>							—

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 17-1 **VOFF<17:1>**: 割り込みベクタ「x」アドレス オフセットビット

bit 0 **未実装**: 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## 8.0 オシレータの設定

**Note:** 本書は PIC32MZ DA の機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 42. Oscillators with Enhanced PLL』(DS60001250) を参照してください。この文書は、Microchip 社ウェブサイト([www.microchip.com/PIC32](http://www.microchip.com/PIC32))から入手できます。

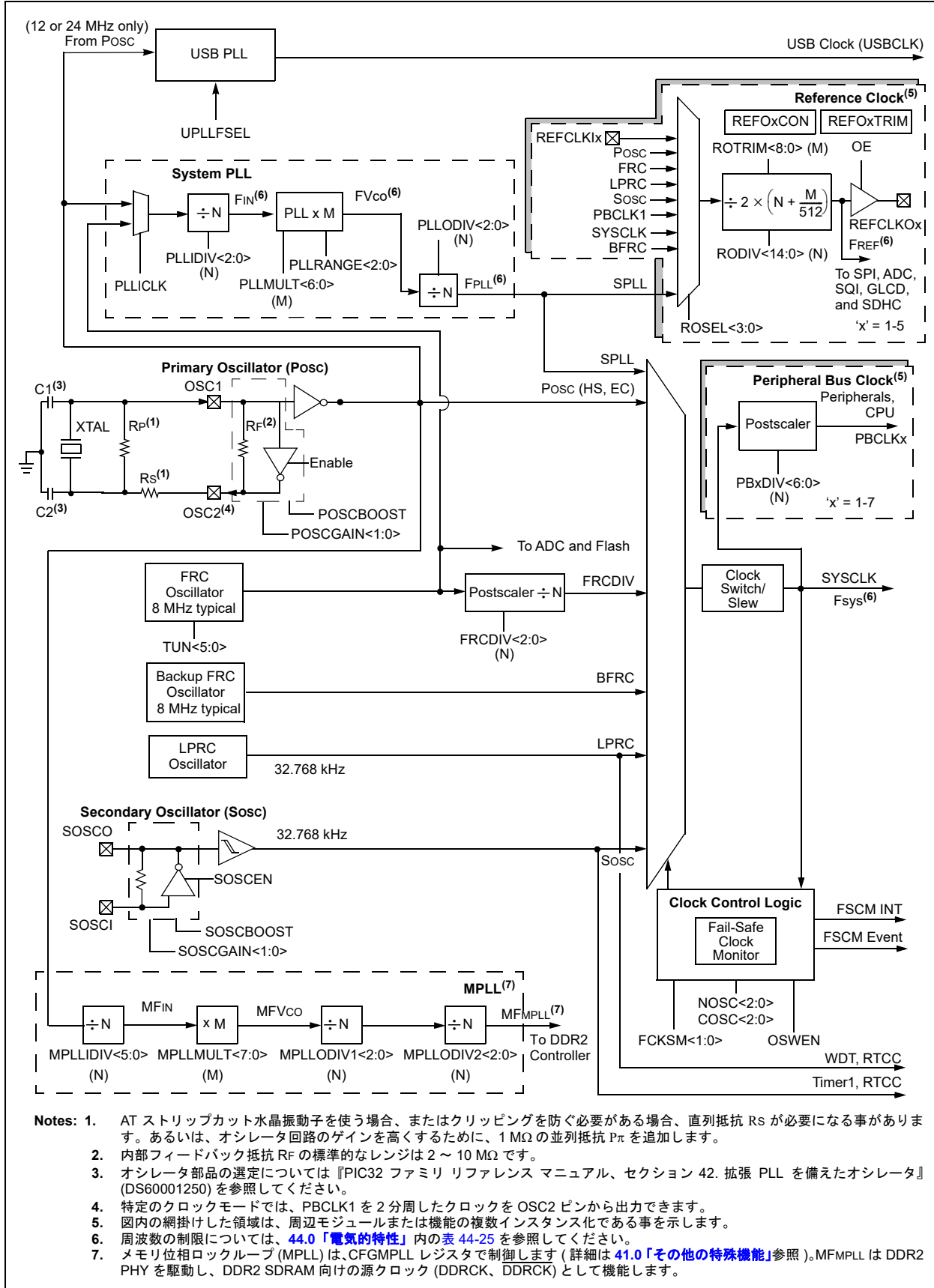
PIC32MZ DA のオシレータ システムは以下のモジュールと機能を備えます。

- クロック源として選択可能な 5 つの外部および内部オシレータ オプション
- 入力分周比 / 逡倍比 / 出力分周比を選択可能な内蔵 PLL により、内部および外部オシレータソースの動作周波数を逡倍可能
- オシレータソースを分周する内蔵ポストスケアラ (分周比を選択可能)
- ソフトウェアによるクロック源の切り換えが可能
- クロック障害を検出し、専用のバックアップ FRC (BFRC) を使ってアプリケーションを安全に回復またはシャットダウンするフェイルセーフ クロック監視 (FSCM) 機能
- DDR2 および USB モジュール向けの専用内蔵 PLL
- 柔軟な参照クロック出力
- 周辺モジュール向けに周波数の異なる複数のクロックを生成する事で性能の柔軟性が向上

図 8-1 に、オシレータ システムのブロック図を示します。周辺モジュールに対するクロックの割り振りを表 8-1 に示します。

# PIC32MZ グラフィック (DA) ファミリ

図 8-1: PIC32MZ DAファミリのオシレータ ブロック図



# PIC32MZ グラフィック (DA) ファミリ

表 8-1: システムおよび周辺モジュールへのクロックの割り振り

周辺モジュール	クロック源																		
	FRC	LPRC	SOSC	SYCLK	USBCLK	MPLL	PBCLK1 <sup>(1)</sup>	PBCLK2	PBCLK3	PBCLK4	PBCLK5	PBCLK6	PBCLK7	REFCLK01	REFCLK02	REFCLK03	REFCLK04	REFCLK5	
CPU													X						
WDT		X		X			X <sup>(3)</sup>												
DMT				X			X <sup>(3)</sup>						X						
GLCD				X <sup>(3)</sup>															X <sup>(6)</sup>
GPU				X															
DDR2C				X <sup>(3)</sup>		X													
SDHC											X <sup>(3)</sup>						X		
フラッシュ	X <sup>(2)</sup>			X <sup>(2)</sup>							X <sup>(2)</sup>								
ADC	X			X					X <sup>(3)</sup>							X			
コンパレータ									X <sup>(3)</sup>										
CTMU									X <sup>(3)</sup>										
暗号エンジン											X <sup>(3)</sup>								
RNG											X <sup>(3)</sup>								
USB					X						X <sup>(3)</sup>								
USBCR <sup>(7)</sup>											X <sup>(3)</sup>								
CAN											X <sup>(3)</sup>								
Ethernet											X <sup>(3)</sup>								
PMP								X <sup>(3)</sup>											
I <sup>2</sup> C								X <sup>(3)</sup>											
UART								X <sup>(3)</sup>											
RTCC		X	X									X <sup>(3)</sup>							
EBI				X															
SQI											X <sup>(3)</sup>				X				
SPI								X						X					
タイマ		X	X <sup>(4)</sup>						X										
出力コンペア									X										
入力キャプチャ									X										
ポート										X <sup>(3)</sup>									
DMA				X															
割り込み				X															
プリフェッチ				X															
OSC2 ピン							X <sup>(5)</sup>												
DSCTRL <sup>(8)</sup>				X							X								
HLVD							X <sup>(3)</sup>												

- Note**
- 1: PBCLK1 はシステム モジュールによって使われ、OFF にはできません。
  - 2: SYCLK/PBCLK5 はフラッシュ コントローラとの間の双方向のデータのフェッチ用に使い、FRC クロックはプログラミング用に使います。
  - 3: 特殊機能レジスタ (SFR) へのアクセス専用です。
  - 4: Timer1 のみ
  - 5: 特定のクロックモードでは、PBCLK1 を 2 分周したクロックを OSC2 ピンから出力できます。
  - 6: REFCLK05 (SPLL クロックから分周) は画素クロック用に使います。
  - 7: USBCR は USB 用のクロック / リセット制御ブロックです。
  - 8: DSCTRL は、ディープスリープ制御ブロックです。

# PIC32MZ グラフィック (DA) ファミリ

---

## 8.1 フェイルセーフ クロック監視 (FSCM) 機能

PIC32MZ DA のオシレータ システムはフェイルセーフ クロック監視 (FSCM) 機能を備えています。FSCM は SYSCLK の動作を監視します。FSCM は、SYSCLK の障害を検出した時にクロック源を SYSCLK から BFRC オシレータに切り換えて、NMI をトリガします。BFRC は未調整の 8 MHz オシレータであり、FSCM イベント中に SYSCLK を駆動します。NMI の実行時に、ソフトウェアはメインオシレータの再起動を試みるか、システムをシャットダウンできます。

スリープ中は、SYSCLK と FSCM の両方が停止します (従って、FSCM によるクロック障害の検出は発生しません)。



## 8.2 オシレータ制御レジスタ

TABLE 8-2:オシレータ設定関連のレジスタマップ

レジスタアドレス (BF80_#)	レジスタ名	ビットレンジ	Bit														リセット値 <sup>(1)</sup>			
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0	
1200	OSCCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0020	
		15:0	—	COSC<2:0>				—	NOSC<2:0>			—	CLKLOCK	—	—	SLPEN	CF	—	SOSCEN	OSWEN
1210	OSCTUN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	TUN<5:0>				00xx	
1220	SPLLCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	01xx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLLMULT<6:0>		0x0x
1280	REFO1CON	31:16	—	RODIV<14:0>														0000		
		15:0	ON	—	SIDL	OE	RSLP	—	DIVSWEN	ACTIVE	—	—	—	—	—	—	—	—	—	ROSEL<3:0>
1290	REFO1TRIM	31:16	ROTRIM<8:0>														0000			
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
12A0	REFO2CON	31:16	—	RODIV<14:0>														0000		
		15:0	ON	—	SIDL	—	RSLP	—	DIVSWEN	ACTIVE	—	—	—	—	—	—	—	—	—	ROSEL<3:0>
12B0	REFO2TRIM	31:16	ROTRIM<8:0>														0000			
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
12C0	REFO3CON	31:16	—	RODIV<14:0>														0000		
		15:0	ON	—	SIDL	OE	RSLP	—	DIVSWEN	ACTIVE	—	—	—	—	—	—	—	—	—	ROSEL<3:0>
12D0	REFO3TRIM	31:16	ROTRIM<8:0>														0000			
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
12E0	REFO4CON	31:16	—	RODIV<14:0>														0000		
		15:0	ON	—	SIDL	OE	RSLP	—	DIVSWEN	ACTIVE	—	—	—	—	—	—	—	—	—	ROSEL<3:0>
12F0	REFO4TRIM	31:16	ROTRIM<8:0>														0000			
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
1300	REFO5CON	31:16	—	RODIV<14:0>														0000		
		15:0	ON	—	SIDL	—	RSLP	—	DIVSWEN	ACTIVE	—	—	—	—	—	—	—	—	—	ROSEL<3:0>
1310	REFO5TRIM	31:16	ROTRIM<8:0>														0000			
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
1340	PB1DIV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	PBDIVRDY	—	—	—	—	—	—	—	—	—	—	—	PBDIV<6:0>
1350	PB2DIV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	—	PBDIVRDY	—	—	—	—	—	—	—	—	—	—	—	PBDIV<6:0>
1360	PB3DIV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	—	PBDIVRDY	—	—	—	—	—	—	—	—	—	—	—	PBDIV<6:0>
1370	PB4DIV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	—	PBDIVRDY	—	—	—	—	—	—	—	—	—	—	—	PBDIV<6:0>

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: リセット値は DEVCFGx コンフィグレーション ビットの設定とリセットのタイプによって決まります。

TABLE 8-2: オシレータ設定関連のレジスタマップ (続き)

レジスタアドレス (BF80_#)	レジスタ名	ビットレンジ	Bit														全リセット <sup>(1)</sup>		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
1380	PB5DIV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	PBDIVRDY	—	—	—	—	PBDIV<6:0>						8801	
1390	PB6DIV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ON	—	—	—	PBDIVRDY	—	—	—	—	PBDIV<6:0>						8803	
13A0	PB7DIV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ON	—	—	—	PBDIVRDY	—	—	—	—	PBDIV<6:0>						8800	
13C0	SLEWCON	31:16	—	—	—	—	—	—	—	—	—	—	—	SYSDIV<3:0>				0000	
		15:0	—	—	—	—	—	SLWDIV<2:0>			—	—	—	—	—	UPEN	DNEN	BUSY	0000
13D0	CLKSTAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	SPLLRDY	—	LPRCRDY	SOSCRDY	—	POSCRDY	—	FRCRDY	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: リセット値は DEVCFGx コンフィグレーション ビットの設定とリセットのタイプによって決まります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 8-1: OSCCON: オシレータ制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	FRCDIV<2:0>		
23:16	R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
	DRMEN	—	SLP2SPD	—	—	—	—	—
15:8	U-0	R-0	R-0	R-0	U-0	R/W-y	R/W-y	R/W-y
	—	COSC<2:0>			—	NOSC<2:0>		
7:0	R/W-0	U-0	U-0	R/W-0	R/W-0, HS	U-0	R/W-y	R/W-y
	CLKLOCK	—	—	SLPEN	CF	—	SOSCEN	OSWEN <sup>(1)</sup>

**凡例:** y = POR 時にコンフィグレーションビットからの値に設定 HS = ハードウェアでセット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-27 **未実装:** 「0」として読み出し

bit 26-24 **FRCDIV<2:0>:** 内部高速 RC (FRC) オシレータ クロック分周比ビット

- 111 = FRC を 256 分周する
- 110 = FRC を 64 分周する
- 101 = FRC を 32 分周する
- 100 = FRC を 16 分周する
- 011 = FRC を 8 分周する
- 010 = FRC を 4 分周する
- 001 = FRC を 2 分周する
- 000 = FRC を 1 分周する (既定値)

bit 23 **DRMEN:** ドリームモード イネーブルビット

- 1 = ドリームモードを有効にする
- 0 = ドリームモードを無効にする

bit 22 **未実装:** 「0」として読み出し

bit 21 **SLP2SPD:** スリープからの 2 段階起動制御ビット

- 1 = 選択したクロックの動作準備が完了するまで FRC を SYSCLK として使う
- 0 = 選択したクロックを直接使う

bit 20-15 **未実装:** 「0」として読み出し

bit 14-12 **COSC<2:0>:** 切り換え前オシレータ選択ビット

- 111 = システム PLL (SPLL)
- 110 = バックアップ高速 RC (BFRC) オシレータ
- 101 = 内部低消費電力 RC (LPRC) オシレータ
- 100 = セカンダリ オシレータ (Sosc)
- 011 = 予約済み
- 010 = プライマリ オシレータ (Posc) (HS または EC)
- 001 = システム PLL (SPLL)
- 000 = FRCDIV<2:0> ビット (FRCDIV) に基づいて分周した内部高速 RC (FRC) オシレータ

bit 11 **未実装:** 「0」として読み出し

**Note 1:** このビットのリセット値は、IESO ビット (DEVCFG1<7>) の設定によって決まります。IESO = 1 の場合、リセット値は「1」です。IESO = 0 の場合、リセット値は「0」です。

**Note:** このレジスタに書き込むには、ロック解除シーケンスが必要です。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 8-1: OSCCON: オシレータ制御レジスタ

bit 10-8 **NOOSC<2:0>**: 切り換え後オシレータ選択ビット

111 = システム PLL (SPLL)

110 = 予約済み

101 = 内部低消費電力 RC (LPRC) オシレータ

100 = セカンダリ オシレータ (Sosc)

011 = 予約済み

010 = プライマリ オシレータ (Posc) (HS または EC)

001 = システム PLL (SPLL)

000 = FRCDIV<2:0> ビット (FRCDIV) に基づいて分周した内部高速 RC (FRC) オシレータ

これらのビットは、リセット時に FNOSC<2:0> コンフィグレーション ビット (DEVCFG1<2:0>) の値に設定されます。

bit 7 **CLKLOCK**: クロック選択ロック イネーブルビット

1 = クロックと PLL の選択をロックする

0 = クロックと PLL の選択をロックしない (変更が可能)

bit 6-5 **未実装**: 「0」として読み出し

bit 4 **SLPEN**: スリープモード イネーブルビット

1 = WAIT 命令の実行によってデバイスはスリープに移行する

0 = WAIT 命令の実行によってデバイスはアイドルに移行する

bit 3 **CF**: クロック障害検出ビット

1 = FSCM がクロック障害を検出した

0 = クロック障害は検出されていない

bit 2 **未実装**: 「0」として読み出し

bit 1 **SOSCEN**: セカンダリ オシレータ (Sosc) イネーブルビット

1 = セカンダリ オシレータを有効にする

0 = セカンダリ オシレータを無効にする

bit 0 **OSWEN**: オシレータ切り換えイネーブルビット (1)

1 = NOOSC<2:0> ビットで指定されたオシレータへの切り換えを開始する

0 = オシレータ切り換えは完了した

**Note 1:** このビットのリセット値は、IESO ビット (DEVCFG1<7>) の設定によって決まります。IESO = 1 の場合、リセット値は「1」です。IESO = 0 の場合、リセット値は「0」です。

**Note:** このレジスタに書き込むには、ロック解除シーケンスが必要です。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 8-2: OSCTUN: FRC 調整レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	TUN<5:0> <sup>(1)</sup>					

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-6 **未実装:** 「0」として読み出し

bit 5-0 **TUN<5:0>:** FRC オシレータ調整ビット <sup>(1)</sup>

100000 = 中心周波数 -4%

100001 =

•

•

•

111111 =

000000 = 中心周波数 (オシレータは公称周波数 (8 MHz) で動作する)

000001 =

•

•

•

011110 =

011111 = 中心周波数 +4%

**Note 1:** OSCTUN レジスタにより、FRC 周波数に対する温度の影響を幅広いレンジで補償できます。上記の周波数調整量は近似値であり、正確な特性値ではありません。

**Note:** このレジスタに書き込むには、ロック解除シーケンスが必要です。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 8-3: SPLLCON: システム PLL 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-y	R/W-y	R/W-y
	—	—	—	—	—	PLLODIV<2:0>		
23:16	U-0	R/W-y	R/W-y	R/W-y	R/W-y	R/W-y	R/W-y	R/W-y
	—	PLLMULT<6:0>						
15:8	U-0	U-0	U-0	U-0	U-0	R/W-y	R/W-y	R/W-y
	—	PLLIDIV<2:0>						
7:0	R/W-y	U-0	U-0	U-0	U-0	R/W-y	R/W-y	R/W-y
	PLLICK	—	—	—	—	PLLRANGE<2:0>		

**凡例:**

y = POR 時にコンフィグレーション ビットからの値に設定

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-27 **未実装:** 「0」として読み出し

bit 26-24 **PLLODIV<2:0>:** システム PLL 出カクロック分周比ビット

111 = 予約済み

110 = 予約済み

101 = PLL を 32 分周する

100 = PLL を 16 分周する

011 = PLL を 8 分周する

010 = PLL を 4 分周する

001 = PLL を 2 分周する

000 = 予約済み

これらのビットの既定値設定は、DEVCFG2 レジスタの FPLLODIV<2:0> コンフィグレーション ビットで指定します。41.0「その他の特殊機能」内のレジスタ 41-5 を参照してください。

bit 23 **未実装:** 「0」として読み出し

bit 22-16 **PLLMULT<6:0>:** システム PLL 通倍比ビット

11111111 = 128 通倍する

11111110 = 127 通倍する

11111101 = 126 通倍する

11111100 = 125 通倍する

•

•

•

00000000 = 1 通倍する

これらのビットの既定値設定は、DEVCFG2 レジスタの FPLLMULT<6:0> コンフィグレーション ビットで指定します。41.0「その他の特殊機能」内のレジスタ 41-5 を参照してください。

bit 15-11 **未実装:** 「0」として読み出し

**Note 1:** このレジスタに書き込むには、ロック解除シーケンスが必要です。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

**2:** SPLLCON をクロック源として選択 (COSC<2:0> = 001) している場合、このレジスタへの書き込みは禁止されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 8-3: SPLLCON: システム PLL 制御レジスタ

bit 10-8 **PLLIDIV<2:0>**: システム PLL 入力クロック分周比ビット

111 = 8 分周する  
110 = 7 分周する  
101 = 6 分周する  
100 = 5 分周する  
011 = 4 分周する  
010 = 3 分周する  
001 = 2 分周する  
000 = 1 分周する

これらのビットの既定値設定は、DEVCFG2 レジスタの FPLLIDIV<2:0> コンフィグレーション ビットで指定します。41.0「その他の特殊機能」内のレジスタ 41-5 を参照してください。PLLICK を「1」(FRC をシステム PLL への入力として選択する)に設定した場合、PLL は PLLIDIV<2:0> の設定を無視し、分周比は 1 分周に設定されます。

bit 7 **PLLICK**: システム PLL 入力クロック源ビット

1 = FRC をシステム PLL への入力として選択する  
0 = Posc をシステム PLL への入力として選択する

POR 時の既定値は、DEVCFG2 レジスタの FPLLICK コンフィグレーション ビットで指定します。41.0「その他の特殊機能」内のレジスタ 41-5 を参照してください。

bit 6-3 **未実装**: 「0」として読み出し

bit 2-0 **PLLRRANGE<2:0>**: システム PLL 周波数レンジ選択ビット

111 = 予約済み  
110 = 予約済み  
101 = 34 ~ 64 MHz  
100 = 21 ~ 42 MHz  
011 = 13 ~ 26 MHz  
010 = 8 ~ 16 MHz  
001 = 5 ~ 10 MHz  
000 = バイパス

これらのビットの既定値設定は、DEVCFG2 レジスタの FPLLRRNG<2:0> コンフィグレーション ビットで指定します。41.0「その他の特殊機能」内のレジスタ 41-5 を参照してください。

**Note 1:** このレジスタに書き込むには、ロック解除シーケンスが必要です。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

**2:** SPLLCON をクロック源として選択 (COSC<2:0> = 001) している場合、このレジスタへの書き込みは禁止されます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 8-4: REFOxCON: 参照オシレータ制御レジスタ (x = 1 ~ 4)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RODIV<14:8>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RODIV<7:0>							
15:8	R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0, HC	R-0, HS, HC
	ON <sup>(1)</sup>	—	SIDL	OE	RSLP <sup>(2)</sup>	—	DIVSWEN	ACTIVE <sup>(1)</sup>
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	ROSEL<3:0> <sup>(3)</sup>							

<b>凡例:</b>	HC=ハードウェアでクリア	HS = ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

- bit 31 **未実装:** 「0」として読み出し
- bit 30-16 **RODIV<14:0>:** 参照クロック分周比ビット  
参照クロックの分周比を選択します (詳細は [図 8-1](#) 参照)。「0」は分周なし (1 分周) を選択します。
- bit 15 **ON:** 出カインーブルビット<sup>(1)</sup>  
1 = 参照オシレータ モジュールを有効にする  
0 = 参照オシレータ モジュールを無効にする
- bit 14 **未実装:** 「0」として読み出し
- bit 13 **SIDL:** アイドル中周辺モジュール停止ビット  
1 = デバイスがアイドルモードに移行するとモジュールは動作を停止する  
0 = デバイスがアイドル中でもモジュールは動作を継続する
- bit 12 **OE:** 参照クロック出カインーブルビット  
1 = REFCLK0x ピンで参照クロックを出力する  
0 = REFCLK0x ピンで参照クロックを出力しない
- bit 11 **RSLP:** スリープ中参照オシレータ モジュール動作ビット<sup>(2)</sup>  
1 = スリープ中も参照オシレータ モジュール出力は動作を継続する  
0 = スリープ中に参照オシレータ出力は停止する
- bit 10 **未実装:** 「0」として読み出し
- bit 9 **DIVSWEN:** 分周比切り換えイネーブルビット  
1 = 分周比切り換えを実行中  
0 = 分周比切り換えは完了した
- bit 8 **ACTIVE:** 参照クロック要求ステータスビット<sup>(1)</sup>  
1 = 参照クロック要求はアクティブ  
0 = 参照クロック要求は非アクティブ
- bit 7-4 **未実装:** 「0」として読み出し
- bit 3-0 **ROSEL<3:0>:** 参照クロック源選択ビット<sup>(3)</sup>  
1111 = 予約済み  
.  
.  
.  
1001 = BFRC  
1000 = REFCLKIx  
0111 = システム PLL 出力  
0110 = 予約済み  
0101 = SOSC  
0100 = LPRC  
0011 = FRC  
0010 = Posc  
0001 = PBCLK1  
0000 = SYSCLK

- Note 1:** ON ビットと ACTIVE ビットの値が一致していない時にこのレジスタに書き込まないでください。  
**2:** ROSEL<3:0> ビットが「0000」または「0001」の場合、このビットは無視されます。  
**3:** ACTIVE ビットが「1」の時に ROSEL<3:0> ビットに書き込まないでください。書き込んだ場合の結果は未確定です。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 8-5: REFOxTRIM: 参照オシレータ調整レジスタ (x = 1 ~ 4)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ROTRIM<8:1>							
23:16	R/W-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
	ROTRIM<0>	—	—	—	—	—	—	—
15:8	U-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-23 **ROTRIM<8:0>**: 参照オシレータ調整ビット

- 111111111 = RODIV 値に分周比 511/512 を加算する
- 111111110 = RODIV 値に分周比 511/512 を加算する
- 
- 
- 100000000 = RODIV 値に分周比 256/512 を加算する
- 
- 
- 000000010 = RODIV 値に分周比 2/512 を加算する
- 000000001 = RODIV 値に分周比 1/512 を加算する
- 000000000 = RODIV 値に分周比を加算しない

bit 22-0 **未実装**: 「0」として読み出し

- Note 1:** ON ビット (REFOxCON<15>) が「1」の時にこのビットに書き込んだ場合、その効果は DIVSWEN ビットが「1」にセットされるまで反映されません。
- 2:** ON ビット (REFOxCON<15>) と ACTIVE ビット (REFOxCON<8>) の値が一致していない時にこのレジスタに書き込まないでください。
- 3:** RODIV<14:0> (REFOxCON<30:16>) = 0 の場合、このレジスタで指定した値は効力を持ちません。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 8-6: PBxDIV: 周辺モジュールバス「x」クロック分周比制御レジスタ (x = 1 ~ 7)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-1 ON <sup>(1)</sup>	U-0 —	U-0 —	U-0 —	R-1 PBDIVRDY	U-0 —	U-0 —	U-0 —
7:0	U-0 —	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PBDIV<6:0>								

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ON:** 周辺モジュールバス「x」出力クロック イネーブルビット<sup>(1)</sup>  
 1 = 出力クロックを有効にする  
 0 = 出力クロックを無効にする

bit 14-12 **未実装:** 「0」として読み出し

bit 11 **PBDIVRDY:** 周辺モジュールバス「x」クロック分周レディビット  
 1 = クロック分周ロジックは分周比を切り換え中ではない (PBxDIV<6:0> ビットは書き込み可能)  
 0 = クロック分周ロジックは分周比を切り換え中 (PBxDIV<6:0> ビットは書き込み不可)

bit 10-7 **未実装:** 「0」として読み出し

bit 6-0 **PBDIV<6:0>:** 周辺モジュールバス「x」クロック分周比制御ビット  
 11111111 = SYSCLK を 128 分周して PBCLKx を生成する  
 11111110 = SYSCLK を 127 分周して PBCLKx を生成する  
 .  
 .  
 .  
 00000111 = SYSCLK を 4 分周して PBCLKx を生成する  
 00000110 = SYSCLK を 3 分周して PBCLKx を生成する  
 00000011 = SYSCLK を 2 分周して PBCLKx を生成する (x < 7 の場合の既定値)  
 00000000 = SYSCLK を 1 分周して PBCLKx を生成する (x ≥ 7 の場合の既定値)

**Note 1:** 周辺モジュールバス 1 のクロックは OFF にできません。従って、PB1DIV レジスタの ON ビットに「0」を書き込む事はできません。

**Note:** このレジスタに書き込むには、ロック解除シーケンスが必要です。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 8-7: SLEWCON: オシレータ スルーイング制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	SYSDIV<3:0> <sup>(1)</sup>			
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	SLWDIV<2:0>		
7:0	U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R-0, HS, HC
	—	—	—	—	—	UPEN	DNEN	BUSY

<b>凡例:</b>	HC=ハードウェアでクリア	HS =ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31-20 **未実装:** 「0」として読み出し

bit 19-16 **SYSDIV<3:0>:** システムクロック分周比制御ビット<sup>(1)</sup>

1111 = SYSCLK を 16 分周する  
 1110 = SYSCLK を 15 分周する  
 .  
 .  
 .  
 0010 = SYSCLK を 3 分周する  
 0001 = SYSCLK を 2 分周する  
 0000 = SYSCLK を分周しない

bit 15-11 **未実装:** 「0」として読み出し

bit 10-8 **SLWDIV<2:0>:** スルーイング分周比ステップ制御ビット

これらのビットは、周波数変更時のスルーイングにおける最大分周比ステップ数を制御します。  
 111 = 128/64/32/16/8/4/2 分周した後に分周なしにする  
 110 = 64/32/16/8/4/2 分周した後に分周なしにする  
 101 = 32/16/8/4/2 分周した後に分周なしにする  
 100 = 16/8/4/2 分周した後に分周なしにする  
 011 = 8/4/2 分周した後に分周なしにする  
 010 = 4/2 分周した後に分周なしにする  
 001 = 2 分周した後に分周なしにする  
 000 = スルーイング中に分周しない  
 周波数を下げる場合、分周比ステップは逆順に適用します (例: 2/4/8.. 分周)。

bit 7-3 **未実装:** 「0」として読み出し

bit 2 **UPEN:** 上向きスルーイング イネーブルビット

1 = 上向きの周波数切り換えに対してスルーイングを有効にする  
 0 = 上向きの周波数切り換えに対してスルーイングを無効にする

bit 1 **DNEN:** 下向きスルーイング イネーブルビット

1 = 下向きの周波数切り換えに対してスルーイングを有効にする  
 0 = 下向きの周波数切り換えに対してスルーイングを無効にする

bit 0 **BUSY:** クロック切り換えスルーイング アクティブ ステータスビット

1 = クロック周波数は新しい周波数へ向かってスルーイング中  
 0 = クロック切り換えは最終周波数に達した

**Note 1:** UPEN ビットと DNEN ビットの両方が「0」の場合、SYSDIV<3:0> ビットの設定は無視され、SYSCLK は分周されません。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 8-8: CLKSTAT: オシレータ クロック ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R-0	U-0	R-0	R-0	U-0	R-0	U-0	R-0
	SPLLRDY	—	LPRCRDY	SOSCRDY	—	POSCRDY	—	FRCRDY

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-8 **未実装:** 「0」として読み出し

bit 7 **SPLLRDY:** システム PLL (SPLL) 準備完了ステータスビット

1 = SPLL は準備完了済み  
 0 = SPLL の準備は完了していない

bit 6 **未実装:** 「0」として読み出し

bit 5 **LPRCRDY:** 低消費電力 RC (LPRC) オシレータ準備完了ステータスビット

1 = LPRC は安定して準備完了済み  
 0 = LPRC は無効または動作していない

bit 4 **SOSCRDY:** セカンダリ オシレータ (Sosc) 準備完了ステータスビット

1 = Sosc は安定して準備完了済み  
 0 = Sosc は無効または動作していない

bit 3 **未実装:** 「0」として読み出し

bit 2 **POSCRDY:** プライマリ オシレータ (Posc) 準備完了ステータスビット

1 = Posc は安定して準備完了済み  
 0 = Posc は無効または動作していない

bit 1 **未実装:** 「0」として読み出し

bit 0 **FRCRDY:** 高速 RC (FRC) オシレータ準備完了ステータスビット

1 = FRC は安定して準備完了済み  
 0 = FRC は無効または動作していない

# PIC32MZ グラフィック (DA) ファミリ

## 9.0 プリフェッチ モジュール

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。このセクションの内容を補足する情報は『PIC32 Family Reference Manual, Section 41. Prefetch Module for Devices with L1 CPU Cache』(DS60001183) を参照してください。この文書は Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

プリフェッチ モジュールは、PIC32MZ DA ファミリのデバイスが内蔵する性能拡張モジュールです。デバイスが高速なクロックで動作している場合、プログラムフラッシュメモリ (PFM) へのアクセス時間要件を満たすには PFM 読み出しトランザクションに待機状態を挿入する必要があります。CPU から高速にアクセスできる一時的な保持領域に、プリフェッチした命令を格納する事により、CPU は待機状態を意識する事なくアクセスできます。CPU へのデータパスは 32 ビット幅ですが、PFM へのデータパスは 128 ビット幅です。この広いデータパスにより、4 倍の周波数で動作する 32 ビットパスと同じ帯域幅が得られます。

プリフェッチ モジュールは、PFM の一部を一時的な保持領域 (ラインと呼ぶ) に格納します。各ラインはタグとデータフィールドを格納します。通常、これらのラインがメモリの現在の内容のコピーを保持する事により、CPU はフラッシュ待機状態を必要とせず命令またはデータを利用できます。

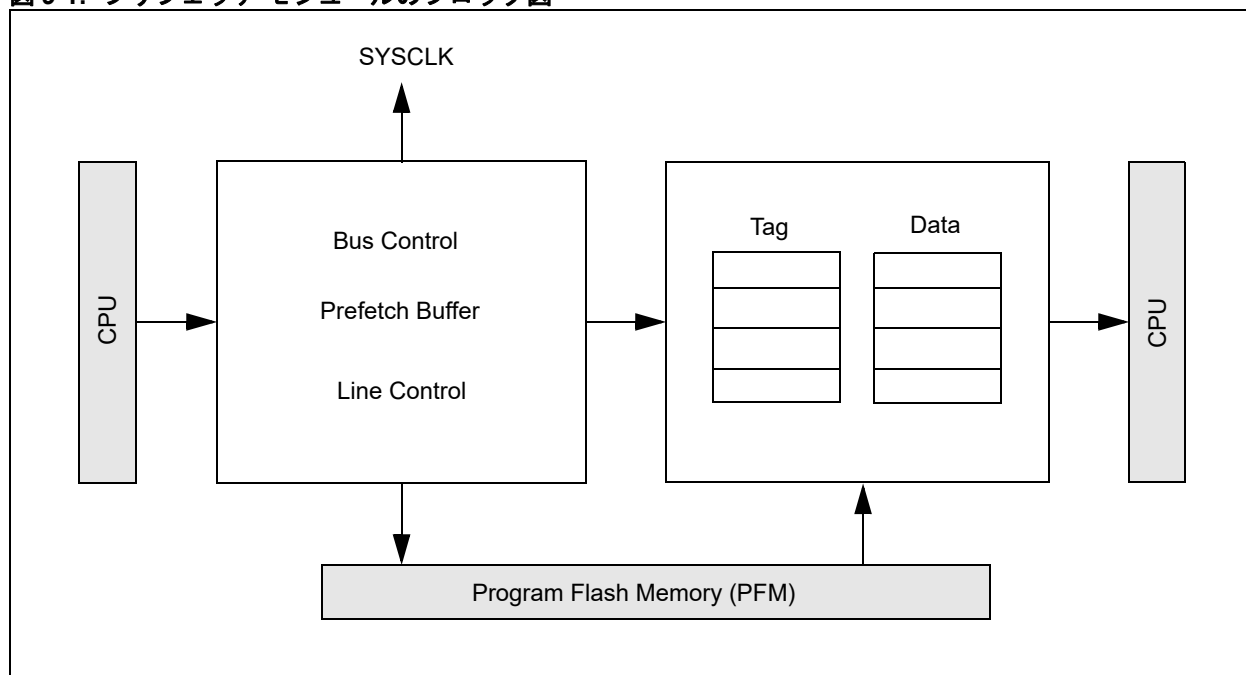
### 9.1 特長

プリフェッチ モジュールの主な特長は下記の通りです。

- 4x16 バイトの完全連想方式ライン
- CPU 命令用の 1 ライン
- CPU データ用の 1 ライン
- 周辺モジュールデータ用の 2 ライン
- 16 バイトの平行メモリフェッチ
- 構成可能な予測プリフェッチ
- 誤りの検出と訂正

図 9-1 に、プリフェッチ モジュールの概略ブロック図を示します。

図 9-1: プリフェッチ モジュールのブロック図



## 9.2 プリフェッチ制御レジスタ

表 9-1: プリフェッチ関連のレジスタマップ

仮想アドレス (BF8E_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
0000	PRECON	31:16	—	—	—	—	—	PFMSECEN	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	PREFEN<1:0>	—	—	PFMWS<2:0>	—	0007
0010	PRESTAT	31:16	—	—	—	—	PFMDDED	PFMSEC	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PFMSECCNT<7:0>	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 9-1: PRECON: プリフェッチ モジュール制御レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0
	—	—	—	—	—	PFMSECEN	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1
	—	—	PREFEN<1:0>		—	PFMWS<2:0> <sup>(1)</sup>		

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-27    **未実装:** 「0」を書き込み、読み出しは無視
- bit 26      **PFMSECEN:** フラッシュ SEC 割り込みイネーブルビット  
             1 = PFMSEC ビット (PRESTAT<26>) がセットされた時に割り込みを生成する  
             0 = PFMSEC ビットがセットされた時に割り込みを生成しない
- bit 25-6    **未実装:** 「0」を書き込み、読み出しは無視
- bit 5-4     **PREFEN<1:0>:** 予測プリフェッチ イネーブルビット  
             11 = 全てのアドレスに対して予測プリフェッチを有効にする  
             10 = CPU 命令と CPU データに対して予測プリフェッチを有効にする  
             01 = CPU 命令に対してのみ予測プリフェッチを有効にする  
             00 = 予測プリフェッチを無効にする
- bit 3        **未実装:** 「0」を書き込み、読み出しは無視
- bit 2-0     **PFMWS<2:0>:** PFM アクセス時間 (SYSCLK 待機状態数) ビット <sup>(1)</sup>  
             111 = 7 待機状態  
             .  
             .  
             .  
             010 = 2 待機状態  
             001 = 1 待機状態  
             000 = 待機状態なし

**Note 1:** 待機状態と SYSCLK の関係は [44.0「電気的特性」](#) 内の表 44-16 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 9-2: PRESTAT: プリフェッチ モジュール ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	R/W-0, HS	R/W-0, HS	U-0	U-0
	—	—	—	—	PFMDED	PFMSEC	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PFMSECCNT<7:0>							

<b>凡例:</b>	HS = ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-28 **未実装:** 「0」を書き込み、読み出しは無視

bit 27 **PFMDED:** フラッシュ ダブルビット エラー検出 (DED) ステータスビット  
 このビットはハードウェアによってセットされ、ソフトウェアでのみクリア (「0」に設定) できます。  
 1 = DED エラーが発生した  
 0 = DED エラーは発生していない

bit 26 **PFMSEC:** フラッシュ シングルビット エラー訂正 (SEC) ステータスビット  
 1 = PFMSECCNT<7:0> が「0」の時に SEC エラーが発生した  
 0 = SEC エラーは発生していない

bit 25-8 **未実装:** 「0」を書き込み、読み出しは無視

bit 7-0 **PFMSECCNT<7:0>:** フラッシュ SEC カウントビット  
 11111111 - 00000000 = SEC カウント



# PIC32MZ グラフィック (DA) ファミリ

## 10.0 ダイレクトメモリアクセス (DMA) コントローラ

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 31. Direct Memory Access (DMA) Controller』(DS60001117) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

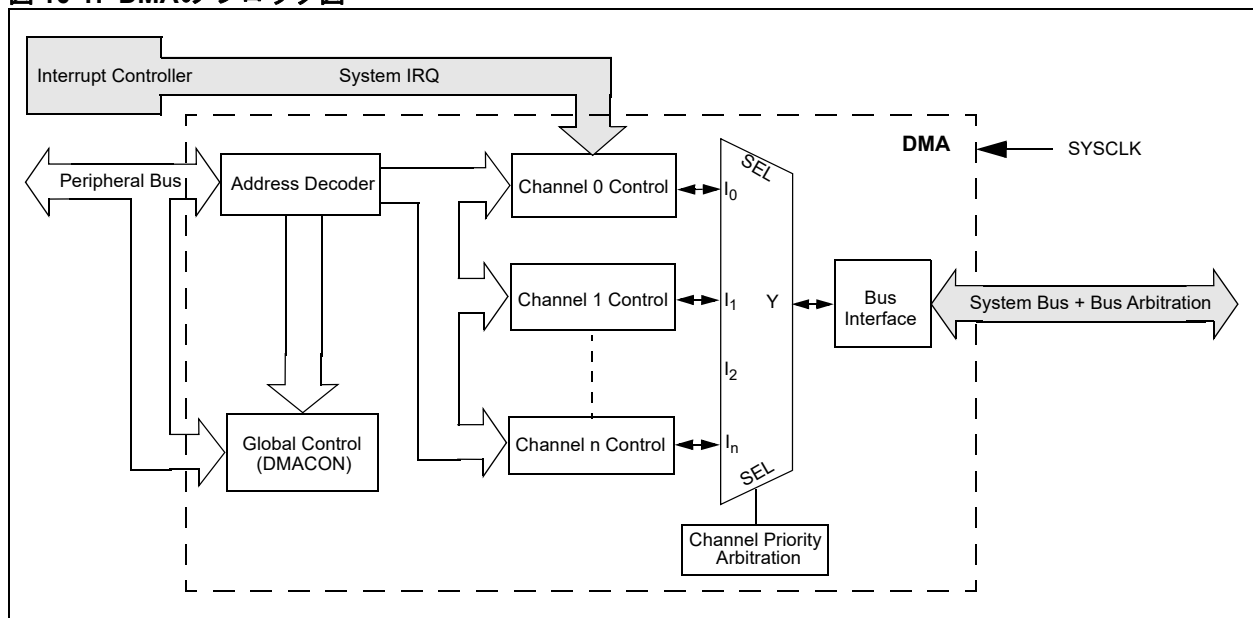
ダイレクトメモリアクセス (DMA) コントローラは、CPU を介さずに各種デバイス間のデータ転送を可能にするバスマスタ モジュールです。DMA 転送のソースとデスティネーションには、メモリにマッピングされたデバイス内部の任意のモジュール (SPI、UART、PMP 等の周辺モジュールとメモリ自体) が使えます。

DMA モジュールの主な特長は以下の通りです。

- 8 つのチャンネル (各チャンネルは以下を備える)
  - 自動的にインクリメントするソースおよびデスティネーション アドレスレジスタ
  - ソースおよびデスティネーション ポインタ
  - メモリ - メモリ間およびメモリ - 周辺モジュール間の転送
- 自動ワードサイズ検出機能
  - バイトレベルの最小転送サイズ
  - ワード境界制約のないソースおよびデスティネーション

- 固定優先度のチャンネル調停
- 柔軟な DMA チャンネル動作モード
  - 手動 (ソフトウェア) または自動 (割り込み) DMA 要求
  - ワンショットまたは自動再送ブロック転送モード
  - チャンネルを連係させるチャンネルチェーンモード
- 柔軟な DMA 要求
  - DMA 要求源として任意の周辺モジュール割り込み要因を選択可能
  - チャンネルごとの DMA 要求源として任意の (適切な) 観測可能割り込みを選択可能
  - DMA 転送中止要求源として任意の周辺モジュール割り込み要因を選択可能
  - 最大 2 バイトのパターン (データ) 照合による転送終了
- 各種の DMA チャンネル ステータス割り込み
  - DMA チャンネル ブロック転送完了
  - ソースのエンプティまたは 1/2 エンプティ
  - デスティネーションのフルまたは 1/2 フル
  - 外部イベントによる DMA 転送中止
  - 無効な DMA アドレス生成
- DMA デバッグサポート機能
  - DMA チャンネルが直近にアクセスしたエラーアドレス
  - 直近にデータを転送した DMA チャンネル
- CRC 生成モジュール
  - 任意のチャンネルに割り当て可能
  - 柔軟に構成可能

図 10-1: DMA のブロック図



## 10.1 DMA 制御レジスタ

表 10-1: DMA グローバル レジスタマップ

仮想アドレス (BF81_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
1000	DMACON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	SUSPEND	DMABUSY	—	—	—	—	—	—	—	—	—	—	—	0000
1010	DMASTAT	31:16	RDWR	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMACH<2:0>
1020	DMAADDR	31:16	DMAADDR<31:0>																0000
		15:0																	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

表 10-2: DMA CRC のレジスタマップ

仮想アドレス (BF81_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
1030	DCRCCON	31:16	—	—	BYTO<1:0>			WBO	—	—	BITO	—	—	—	—	—	—	—	0000
		15:0	—	—	—	PLEN<4:0>						CRCEN	CRCAPP	CRCTYP	—	—	CRCCH<2:0>		
1040	DCRCDATA	31:16	DCRCDATA<31:0>																0000
		15:0																	0000
1050	DCRCXOR	31:16	DCRCXOR<31:0>																0000
		15:0																	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

表 10-3: DMA チャンネル 0 ~ 7 のレジスタマップ

仮想アドレス (BF81_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0		
1060	DCH0CON	31:16	CHPIGN<7:0>										—	—	—	—	—	—	—	7700
		15:0	CHBUSY	—	CHPIGNEN	—	CHPATLEN	—	—	CHCHNS	CHEN	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	0000		
1070	DCH0ECON	31:16	—	—	—	—	—	—	—	—	CHAIRQ<7:0>							00FF		
		15:0	CHSIRQ<7:0>							CFORCE	CABORT	PATEN	SIRQEN	AIRQEN	—	—	—	FF00		
1080	DCH0INT	31:16	—	—	—	—	—	—	—	—	CHSDIE	CHSHIE	CHDDIE	CHDHIE	CHBCIE	CHCCIE	CHTAIE	CHERIE	0000	
		15:0	—	—	—	—	—	—	—	—	CHSDIF	CHSHIF	CHDDIF	CHDHIF	CHBCIF	CHCCIF	CHTAIF	CHERIF	0000	
1090	DCH0SSA	31:16	CHSSA<31:0>																xxxx	
		15:0	CHSSA<31:0>																xxxx	
10A0	DCH0DSA	31:16	CHDSA<31:0>																xxxx	
		15:0	CHDSA<31:0>																xxxx	
10B0	DCH0SSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CHSSIZ<15:0>																xxxx	
10C0	DCH0DSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CHDSIZ<15:0>																xxxx	
10D0	DCH0SPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CHSPTR<15:0>																0000	
10E0	DCH0PTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CHDPTR<15:0>																0000	
10F0	DCH0CSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CHCSIZ<15:0>																xxxx	
1100	DCH0CPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CHCPTR<15:0>																0000	
1110	DCH0DAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CHPDAT<15:0>																xxxx	
1120	DCH1CON	31:16	CHPIGN<7:0>										—	—	—	—	—	—	—	7700
		15:0	CHBUSY	—	CHPIGNEN	—	CHPATLEN	—	—	CHCHNS	CHEN	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	0000		
1130	DCH1ECON	31:16	—	—	—	—	—	—	—	—	CHAIRQ<7:0>							00FF		
		15:0	CHSIRQ<7:0>							CFORCE	CABORT	PATEN	SIRQEN	AIRQEN	—	—	—	FF00		
1140	DCH1INT	31:16	—	—	—	—	—	—	—	—	CHSDIE	CHSHIE	CHDDIE	CHDHIE	CHBCIE	CHCCIE	CHTAIE	CHERIE	0000	
		15:0	—	—	—	—	—	—	—	—	CHSDIF	CHSHIF	CHDDIF	CHDHIF	CHBCIF	CHCCIF	CHTAIF	CHERIF	0000	
1150	DCH1SSA	31:16	CHSSA<31:0>																xxxx	
		15:0	CHSSA<31:0>																xxxx	
1160	DCH1DSA	31:16	CHDSA<31:0>																xxxx	
		15:0	CHDSA<31:0>																xxxx	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 10-3: DMA チャンネル 0 ~ 7 のレジスタマップ (続き)

仮想アドレス (BF81_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0			
1170	DCH1SSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHSSIZ<15:0>																xxxx		
1180	DCH1DSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHDSIZ<15:0>																xxxx		
1190	DCH1SPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHSPTR<15:0>																0000		
11A0	DCH1DPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHDPTR<15:0>																0000		
11B0	DCH1CSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHCSIZ<15:0>																xxxx		
11C0	DCH1CPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHCPTR<15:0>																0000		
11D0	DCH1DAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHPDAT<15:0>																xxxx		
11E0	DCH2CON	31:16	CHPIGN<7:0>										—	—	—	—	—	—	—	—	7700
		15:0	CHBUSY	—	CHPIGNEN	—	CHPATLEN	—	—	—	CHCHNS	CHEN	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	0000		
11F0	DCH2ECON	31:16	CHAIRQ<7:0>										—	—	—	—	—	—	—	—	00FF
		15:0	CHSIRQ<7:0>										CFORCE	CABORT	PATEN	SIRQEN	AIRQEN	—	—	—	FF00
1200	DCH2INT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
1210	DCH2SSA	31:16	CHSSA<31:0>																xxxx		
		15:0	CHSSA<31:0>																xxxx		
1220	DCH2DSA	31:16	CHDSA<31:0>																xxxx		
		15:0	CHDSA<31:0>																xxxx		
1230	DCH2SSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHSSIZ<15:0>																xxxx		
1240	DCH2DSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHDSIZ<15:0>																xxxx		
1250	DCH2SPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHSPTR<15:0>																0000		
1260	DCH2DPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHDPTR<15:0>																0000		
1270	DCH2CSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	CHCSIZ<15:0>																xxxx		

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 10-3: DMA チャンネル 0 ~ 7 のレジスタマップ (続き)

仮想アドレス (BF81_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット 値
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
1280	DCH2CPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHCPTR<15:0>																0000
1290	DCH2DAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHPDAT<15:0>																xxxx
12A0	DCH3CON	31:16	CHPIGN<7:0>																7700
		15:0	CHBUSY	—	CHPIGNEN	—	CHPATLEN	—	—	CHCHNS	CHEN	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	—	0000
12B0	DCH3ECON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	00FF
		15:0	CHSIRQ<7:0>																FF00
12C0	DCH3INT	31:16	—	—	—	—	—	—	—	—	CHSDIE	CHSHIE	CHDDIE	CHDHIE	CHBCIE	CHCCIE	CHTAIE	CHERIE	0000
		15:0	—	—	—	—	—	—	—	—	CHSDIF	CHSHIF	CHDDIF	CHDHIF	CHBCIF	CHCCIF	CHTAIF	CHERIF	0000
12D0	DCH3SSA	31:16	CHSSA<31:0>																xxxx
		15:0	CHSSA<31:0>																xxxx
12E0	DCH3DSA	31:16	CHDSA<31:0>																xxxx
		15:0	CHDSA<31:0>																xxxx
12F0	DCH3SSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHSSIZ<15:0>																xxxx
1300	DCH3DSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHDSIZ<15:0>																xxxx
1310	DCH3SPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHSPTR<15:0>																0000
1320	DCH3DPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHDPTR<15:0>																0000
1330	DCH3CSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHCSIZ<15:0>																xxxx
1340	DCH3CPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHCPTR<15:0>																0000
1350	DCH3DAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHPDAT<15:0>																xxxx
1360	DCH4CON	31:16	CHPIGN<7:0>																7700
		15:0	CHBUSY	—	CHPIGNEN	—	CHPATLEN	—	—	CHCHNS	CHEN	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	—	0000
1370	DCH4ECON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	00FF
		15:0	CHSIRQ<7:0>																FF00
1380	DCH4INT	31:16	—	—	—	—	—	—	—	—	CHSDIE	CHSHIE	CHDDIE	CHDHIE	CHBCIE	CHCCIE	CHTAIE	CHERIE	0000
		15:0	—	—	—	—	—	—	—	—	CHSDIF	CHSHIF	CHDDIF	CHDHIF	CHBCIF	CHCCIF	CHTAIF	CHERIF	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 10-3: DMA チャンネル 0 ~ 7 のレジスタマップ (続き)

仮想アドレス (BF81_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット 値
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
1390	DCH4SSA	31:16	CHSSA<31:0>																xxxx
		15:0																	xxxx
13A0	DCH4DSA	31:16	CHDSA<31:0>																xxxx
		15:0																	xxxx
13B0	DCH4SSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHSSIZ<15:0>																xxxx
13C0	DCH4DSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHDSIZ<15:0>																xxxx
13D0	DCH4SPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHSPTR<15:0>																0000
13E0	DCH4DPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHDPTR<15:0>																0000
13F0	DCH4CSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHCSIZ<15:0>																xxxx
1400	DCH4CPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHCPTR<15:0>																0000
1410	DCH4DAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHPDAT<15:0>																xxxx
1420	DCH5CON	31:16	CHPIGN<7:0>							—	—	—	—	—	—	—	—	7700	
		15:0	CHBUSY	—	CHPIGNEN	—	CHPATLEN	—	—	CHCHNS	CHEN	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	—	0000
1430	DCH5ECON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	00FF	
		15:0	CHSIRQ<7:0>							CFORCE	CABORT	PATEN	SIRQEN	AIRQEN	—	—	—	FF00	
1440	DCH5INT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	CHSDIE	CHSHIE	CHDDIE	CHDHIE	CHBCIE	CHCCIE	CHTAIE	CHERIE	0000
1450	DCH5SSA	31:16	CHSSA<31:0>																xxxx
		15:0																	xxxx
1460	DCH5DSA	31:16	CHDSA<31:0>																xxxx
		15:0																	xxxx
1470	DCH5SSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHSSIZ<15:0>																xxxx
1480	DCH5DSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHDSIZ<15:0>																xxxx
1490	DCH5SPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHSPTR<15:0>																0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 10-3: DMA チャンネル 0 ~ 7 のレジスタマップ (続き)

仮想アドレス (BF81_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット 全			
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0				
14A0	DCH5DPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHDPTR<15:0>																0000			
14B0	DCH5CSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHCSIZ<15:0>																XXXX			
14C0	DCH5CPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHCPTR<15:0>																0000			
14D0	DCH5DAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHPDAT<15:0>																XXXX			
14E0	DCH6CON	31:16	CHPIGN<7:0>																7700			
		15:0	CHBUSY	—	CHPIGNEN	—	CHPATLEN	—	—	CHCHNS	CHEN	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	—	—	0000		
14F0	DCH6ECON	31:16	—	—	—	—	—	—	—	—	CHAIRQ<7:0>							—	—	—	—	00FF
		15:0	CHSIRQ<7:0>								CFORCE	CABORT	PATEN	SIRQEN	AIRQEN	—	—	—	—	FF00		
1500	DCH6INT	31:16	—	—	—	—	—	—	—	—	CHSDIE	CHSHIE	CHDDIE	CHDHIE	CHBCIE	CHCCIE	CHTAIE	CHERIE	0000			
		15:0	—	—	—	—	—	—	—	—	CHSDIF	CHSHIF	CHDDIF	CHDHIF	CHBCIF	CHCCIF	CHTAIF	CHERIF	0000			
1510	DCH6SSA	31:16	CHSSA<31:0>																XXXX			
		15:0	CHSSA<31:0>																XXXX			
1520	DCH6DSA	31:16	CHDSA<31:0>																XXXX			
		15:0	CHDSA<31:0>																XXXX			
1530	DCH6SSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHSSIZ<15:0>																XXXX			
1540	DCH6DSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHDSIZ<15:0>																XXXX			
1550	DCH6SPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHSPTR<15:0>																0000			
1560	DCH6DPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHDPTR<15:0>																0000			
1570	DCH6CSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHCSIZ<15:0>																XXXX			
1580	DCH6CPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHCPTR<15:0>																0000			
1590	DCH6DAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000			
		15:0	CHPDAT<15:0>																XXXX			
15A0	DCH7CON	31:16	CHPIGN<7:0>																7700			
		15:0	CHBUSY	—	CHPIGNEN	—	CHPATLEN	—	—	CHCHNS	CHEN	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	—	—	0000		

例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 10-3: DMA チャンネル 0 ~ 7 のレジスタマップ (続き)

仮想アドレス (BF81_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット値
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
15B0	DCH7ECON	31:16	—	—	—	—	—	—	—	—	CHAIRQ<7:0>							00FF	
		15:0	CHSIRQ<7:0>							CFORCE	CABORT	PATEN	SIRQEN	AIRQEN	—	—	—	FF00	
15C0	DCH7INT	31:16	—	—	—	—	—	—	—	—	CHSDIE	CHSHIE	CHDDIE	CHDHIE	CHBCIE	CHCCIE	CHTAIE	CHERIE	0000
		15:0	—	—	—	—	—	—	—	—	CHSDIF	CHSHIF	CHDDIF	CHDHIF	CHBCIF	CHCCIF	CHTAIF	CHERIF	0000
15D0	DCH7SSA	31:16	CHSSA<31:0>																xxxx
		15:0	CHSSA<31:0>																xxxx
15E0	DCH7DSA	31:16	CHDSA<31:0>																xxxx
		15:0	CHDSA<31:0>																xxxx
15F0	DCH7SSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHSSIZ<15:0>																xxxx
1600	DCH7DSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHDSIZ<15:0>																xxxx
1610	DCH7SPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHSPTR<15:0>																0000
1620	DCH7DPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHDPTR<15:0>																0000
1630	DCH7CSIZ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHCSIZ<15:0>																xxxx
1640	DCH7CPTR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHCPTR<15:0>																0000
1650	DCH7DAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CHPDAT<15:0>																xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-1: DMACON: DMA コントローラ制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	U-0	U-0	R/W-0	R/W-0	U-0	U-0	U-0
	ON	—	—	SUSPEND	DMABUSY	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ON:** DMA ON ビット  
 1 = DMA モジュールを有効にする  
 0 = DMA モジュールを無効にする

bit 14-13 **未実装:** 「0」として読み出し

bit 12 **SUSPEND:** DMA サスペンドビット  
 1 = DMA 転送をサスペンドして CPU からデータバスへの連続したアクセスを可能にする  
 0 = DMA は通常動作する

bit 11 **DMABUSY:** DMA モジュール ビジービット  
 1 = DMA モジュールはアクティブでありデータを転送中  
 0 = DMA モジュールは無効でありデータを転送中ではない

bit 10-0 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-2: DMASTAT: DMA ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	RDWR	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
	—	—	—	—	—	DMACH<2:0>		

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31    **RDWR:** 読み / 書きステータスビット

1 = エラーが検出された時、直近の DMA バスアクセスは読み出しであった  
 0 = エラーが検出された時、直近の DMA バスアクセスは書き込みであった

bit 30-3    **未実装:** 「0」として読み出し

bit 2-0    **DMACH<2:0>:** DMA チャンネルビット

エラーが検出された時、これらのビットは直近にアクティブであったDMAチャンネルの番号を格納します。

レジスタ 10-3: DMAADDR: DMA アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DMAADDR<31:24>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DMAADDR<23:16>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DMAADDR<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DMAADDR<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-0    **DMAADDR<31:0>:** DMA モジュール アドレスビット

エラーが検出された時、これらのビットは直近の DMA アクセスのアドレスを格納します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-4: DCRCCON: DMA CRC 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
	—	—	BYTO<1:0>		WBO <sup>(1)</sup>	—	—	BITO
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	PLEN<4:0> <sup>(1)</sup>				
7:0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	CRCEN	CRCAPP <sup>(1)</sup>	CRCTYP	—	—	CRCCH<2:0>		

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-30 **未実装:** 「0」として読み出し

bit 29-28 **BYTO<1:0>:** CRC バイト順選択ビット

- 11 = ハーフワード境界を中心にバイト順をスワップする (ハーフワード順はソースのまま、各ハーフワード内のバイト順を反転する)
- 10 = ワード境界を中心にハーフワードをスワップする (ハーフワード順を反転し、各ハーフワード内のバイト順はソースのまま)
- 01 = ワード境界を中心にバイト順をスワップする (ソースのバイト順を完全に反転する)
- 00 = スワップしない (ソースのバイト順のまま)

bit 27 **WBO:** CRC 書き込みバイト順選択ビット<sup>(1)</sup>

- 1 = BYTO<1:0> の定義に従ってバイト順を変更したソースデータをデスティネーションに書き込む
- 0 = バイト順を変更せずにソースデータをデスティネーションに書き込む

bit 26-25 **未実装:** 「0」として読み出し

bit 24 **BITO:** CRC ビット順選択ビット

CRCTYP (DCRCCON<15>) = 1 (CRC モジュールは IP ヘッダモード) の場合:

- 1 = IP ヘッダ チェックサムは最下位ビット (LSb) から順番に (つまり逆順に) 計算する
- 0 = IP ヘッダ チェックサムは最上位ビット (MSb) から順番に (つまり正順に) 計算する

CRCTYP (DCRCCON<15>) = 0 (CRC モジュールは LFSR モード) の場合:

- 1 = LFSR CRC は最下位ビット (LSb) から順番に (つまり逆順に) 計算する
- 0 = LFSR CRC は最上位ビット (MSb) から順番に (つまり正順に) 計算する

bit 23-13 **未実装:** 「0」として読み出し

bit 12-8 **PLEN<4:0>:** 多項式長ビット<sup>(1)</sup>

CRCTYP (DCRCCON<15>) = 1 (CRC モジュールは IP ヘッダモード) の場合:

これらのビットは使いません。

CRCTYP (DCRCCON<15>) = 0 (CRC モジュールは LFSR モード) の場合:

これらのビットは [多項式長 -1] を表します。

bit 7 **CRCEN:** CRC イネーブルビット

- 1 = CRC モジュールを有効にする (チャンネル転送は CRC モジュールを経由する)
- 0 = CRC モジュールを無効にする (チャンネル転送は CRC モジュールを経由しない)

**Note 1:** WBO = 1 の場合、バイト順を変更しない転送はサポートされず、CRCAPP ビットはセットできません。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 10-4: DCRCCON: DMA CRC 制御レジスタ ( 続き )

- bit 6 **CRCAPP**: CRC アペンドモード ビット (1)  
1 = DMA はデータをソースから CRC に転送するが、デスティネーションには転送しない  
ブロック転送が完了すると、DMA は CRC 計算値を CHxDSA が指定するアドレスに書き込みます。  
0 = DMA はデータをソースから CRC を経由してデスティネーションに転送する  
デスティネーションに書き込むバイト順は WBO の設定に従います。
- bit 5 **CRCTYP**: CRC タイプ選択ビット  
1 = CRC モジュールは IP ヘッダ チェックサムを計算する  
0 = CRC モジュールは LFSR CRC を計算する
- bit 4-3 **未実装**: 「0」として読み出し
- bit 2-0 **CRCCH<2:0>**: CRC チャンネル選択ビット  
111 = CRC をチャンネル 7 に割り当てる  
110 = CRC をチャンネル 6 に割り当てる  
101 = CRC をチャンネル 5 に割り当てる  
100 = CRC をチャンネル 4 に割り当てる  
011 = CRC をチャンネル 3 に割り当てる  
010 = CRC をチャンネル 2 に割り当てる  
001 = CRC をチャンネル 1 に割り当てる  
000 = CRC をチャンネル 0 に割り当てる

**Note 1:** WBO = 1 の場合、バイト順を変更しない転送はサポートされず、CRCAPP ビットはセットできません。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-5: DCRCDATA: DMA CRC データレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCDATA<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCDATA<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCDATA<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCDATA<7:0>								

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

bit 31-0 **DCRCDATA<31:0>**: CRC データレジスタ ビット

このレジスタへの書き込みによって CRC ジェネレータをシードします。このレジスタからの読み出しは CRC の現在の値を返します。PLEN を超えるビットの読み出し値は「0」です。

CRCTYP (DCRCCON<15>) = 1 (CRC モジュールは IP ヘッダモード) の場合:

IP ヘッダ チェックサム情報は下位 16 ビットに格納され、上位 16 ビットは常に「0」です。このレジスタに書き込まれたデータは変換され、読み出しは 1 の補数表現値 (つまり現在の IP ヘッダ チェックサム値) を返します。

CRCTYP (DCRCCON<15>) = 0 (CRC モジュールは LFSR モード) の場合:

PLEN を超えるビットの読み出し値は「0」です。

レジスタ 10-6: DCRCXOR: DMA CRCXOR イネーブル レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCXOR<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCXOR<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCXOR<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DCRCXOR<7:0>								

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

bit 31-0 **DCRCXOR<31:0>**: CRC XOR レジスタビット

CRCTYP (DCRCCON<15>) = 1 (CRC モジュールは IP ヘッダモード) の場合:

このレジスタは使いません。

CRCTYP (DCRCCON<15>) = 0 (CRC モジュールは LFSR モード) の場合:

1 = シフトレジスタへの XOR 入力を有効にする

0 = シフトレジスタへの XOR 入力を無効にする (前段からレジスタにデータを直接シフトインする)

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-7: DCHxCON: DMA チャンネル x 制御レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHPIGN<7:0>							
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	U-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0
	CHBUSY	—	CHIPGNEN	—	CHPATLEN	—	—	CHCHNS <sup>(1)</sup>
7:0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R-0	R/W-0	R/W-0
	CHEN <sup>(2)</sup>	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-24 **CHPIGN<7:0>**: チャンネル レジスタ データビット  
 パターンマッチ転送終了モードの場合:  
 CHPIGNEN ビットをセットした場合、パターンマッチ判定中にこのビットフィールドで定義されている値に一致したバイトは無視されます。つまりパターンマッチ ロジックが有効であり、かつ CHPIGN ビットがセットされている場合、CHPIGN に一致するバイトが読み出されると、パターンマッチ ロジックはこれを「ドントケア」として扱います。
- bit 23-16 **未実装**: 「0」として読み出し
- bit 15 **CHBUSY**: チャンネル ビジービット  
 1 = チャンネルはアクティブまたは有効  
 0 = チャンネルは非アクティブまたは無効
- bit 14 **未実装**: 「0」として読み出し
- bit 13 **CHIPGNEN**: パターンマッチ バイト無視イネーブルビット  
 1 = パターンマッチが有効な場合、CHPIGN<7:0> ビットに一致したバイトを「ドントケア」として扱う  
 0 = この機能を無効にする
- bit 12 **未実装**: 「0」として読み出し
- bit 11 **CHPATLEN**: パターン長ビット  
 1 = 2 バイト長  
 0 = 1 バイト長
- bit 10-9 **未実装**: 「0」として読み出し
- bit 8 **CHCHNS**: チェーン チャンネル選択ビット<sup>(1)</sup>  
 1 = 優先度が低い方のチャンネルに向かってチェーンする (CH2 の転送が完了すると CH1 が有効になる)  
 0 = 優先度が高い方のチャンネルに向かってチェーンする (CH0 の転送が完了すると CH1 が有効になる)
- bit 7 **CHEN**: チャンネル イネーブルビット<sup>(2)</sup>  
 1 = チャンネルを有効にする  
 0 = チャンネルを無効にする
- bit 6 **CHAED**: 無効中チャンネル イベント検出ビット  
 1 = チャンネルが無効中であってもチャンネル転送開始 / 中止イベントを検出して保留する  
 0 = チャンネルが無効中はチャンネル転送開始 / 中止イベントを無視する
- bit 5 **CHCHN**: チャンネル チェーン イネーブルビット  
 1 = チャンネル チェーンを有効にする  
 0 = チャンネル チェーンを無効にする

**Note 1:** チェーン選択ビットは、チャンネル チェーンを有効(CHCHN = 1)にしている場合にのみ効力を持ちます。  
**2:** このビットをクリアしてチャンネルを無効にした場合、実行中のトランザクションが完了してチャンネルが実際に無効になるまでに数クロックサイクルを要する場合があります。従ってユーザアプリケーションは、CHBUSY ビット (一部デバイスのみ実装) をポーリングして、チャンネルが実際に無効になった事を検出する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 10-7: DCHxCON: DMA チャンネル x 制御レジスタ ( 続き )

- bit 4     **CHAEN:** チャンネル自動有効化ビット  
1 = ブロック転送完了後にチャンネルを自動的に無効にせず有効なままにする  
0 = ブロック転送完了時にチャンネルを無効にする
- bit 3     **未実装:** 「0」として読み出し
- bit 2     **CHEDET:** チャンネル イベント検出ビット  
1 = イベントを検出した  
0 = イベントは検出していない
- bit 1-0   **CHPRI<1:0>:** チャンネル優先度ビット  
11 = チャンネルの優先度は 3 ( 最優先 )  
10 = チャンネルの優先度は 2  
01 = チャンネルの優先度は 1  
00 = チャンネルの優先度は 0

- Note 1:** チェーン選択ビットは、チャンネル チェーンを有効(CHCHN = 1)にしている場合にのみ効力を持ちます。
- 2:** このビットをクリアしてチャンネルを無効にした場合、実行中のトランザクションが完了してチャンネルが実際に無効になるまでに数クロックサイクルを要する場合があります。従ってユーザ アプリケーションは、CHBUSY ビット ( 一部デバイスのみ実装 ) をポーリングして、チャンネルが実際に無効になった事を検出する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-8: DCHxECON: DMA チャンネル x イベント制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	CHAIRQ<7:0> <sup>(1)</sup>							
15:8	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	CHSIRQ<7:0> <sup>(1)</sup>							
7:0	S-0	S-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
	CFORCE	CABORT	PATEN	SIRQEN	AIRQEN	—	—	—

<b>凡例:</b>	S = セット可能ビット
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-24 **未実装:** 「0」として読み出し

bit 23-16 **CHAIRQ<7:0>:** チャンネル転送中止 IRQ ビット<sup>(1)</sup>

11111111 = 割り込み 255 が発生すると実行中の転送を中止して CHAIF フラグをセットする  
 .  
 .  
 .

00000001 = 割り込み 1 が発生すると実行中の転送を中止して CHAIF フラグをセットする  
 00000000 = 割り込み 0 が発生すると実行中の転送を中止して CHAIF フラグをセットする

bit 15-8 **CHSIRQ<7:0>:** チャンネル転送開始 IRQ ビット<sup>(1)</sup>

11111111 = 割り込み 255 が発生すると DMA 転送を開始する  
 .  
 .  
 .

00000001 = 割り込み 1 が発生すると DMA 転送を開始する  
 00000000 = 割り込み 0 が発生すると DMA 転送を開始する

bit 7 **CFORCE:** DMA 転送開始ビット

1 = このビットに「1」を書き込む事で DMA 転送を強制的に開始する  
 0 = このビットは常に「0」として読み出される

bit 6 **CABORT:** DMA 転送中止ビット

1 = このビットに「1」を書き込む事で DMA 転送を強制的に中止する  
 0 = このビットは常に「0」として読み出される

bit 5 **PATEN:** チャンネル パターンマッチ転送中止イネーブルビット

1 = パターンマッチが発生すると転送を中止して CHEN ビットをクリアする  
 0 = パターンマッチを無効にする

bit 4 **SIRQEN:** チャンネル転送開始 IRQ イネーブルビット

1 = CHSIRQ ビットで指定されている番号の割り込みが発生するとチャンネルセル転送を開始する  
 0 = CHSIRQ ビットで指定されている番号の割り込みが発生しても無視する (転送を開始しない)

bit 3 **AIRQEN:** チャンネル中止 IRQ イネーブルビット

1 = CHAIRQ ビットで指定されている番号の割り込みが発生するとチャンネル転送を中止する  
 0 = CHAIRQ ビットで指定されている番号の割り込みが発生しても無視する (転送を中止しない)

bit 2-0 **未実装:** 「0」として読み出し

**Note 1:** 利用可能な割り込み IRQ ソースの一覧は表 7-2:「割り込み IRQ、ベクタ、ビット位値」に記載しています。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-9: DCHxINT: DMA チャンネル x 割り込み制御レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHSDIE	CHSHIE	CHDDIE	CHDHIE	CHBCIE	CHCCIE	CHTAIE	CHERIE
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHSDIF	CHSHIF	CHDDIF	CHDHIF	CHBCIF	CHCCIF	CHTAIF	CHERIF

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-24 **未実装:** 「0」として読み出し
- bit 23 **CHSDIE:** チャンネルソース完了割り込みイネーブルビット  
 1 = この割り込みを有効にする  
 0 = この割り込みを無効にする
- bit 22 **CHSHIE:** チャンネルソース 1/2 エンプティ割り込みイネーブルビット  
 1 = この割り込みを有効にする  
 0 = この割り込みを無効にする
- bit 21 **CHDDIE:** チャンネル デスティネーション完了割り込みイネーブルビット  
 1 = この割り込みを有効にする  
 0 = この割り込みを無効にする
- bit 20 **CHDHIE:** チャンネル デスティネーション 1/2 フル割り込みイネーブルビット  
 1 = この割り込みを有効にする  
 0 = この割り込みを無効にする
- bit 19 **CHBCIE:** チャンネル ブロック転送完了割り込みイネーブルビット  
 1 = この割り込みを有効にする  
 0 = この割り込みを無効にする
- bit 18 **CHCCIE:** チャンネルセル転送完了割り込みイネーブルビット  
 1 = この割り込みを有効にする  
 0 = この割り込みを無効にする
- bit 17 **CHTAIE:** チャンネル転送中止割り込みイネーブルビット  
 1 = この割り込みを有効にする  
 0 = この割り込みを無効にする
- bit 16 **CHERIE:** チャンネル アドレスエラー割り込みイネーブルビット  
 1 = この割り込みを有効にする  
 0 = この割り込みを無効にする
- bit 15-8 **未実装:** 「0」として読み出し
- bit 7 **CHSDIF:** チャンネルソース完了割り込みフラグビット  
 1 = チャンネル ソースポインタがソースの終端に達した (CHSPTR = CHSSIZ)  
 0 = この割り込みは保留されていない
- bit 6 **CHSHIF:** チャンネルソース 1/2 エンプティ割り込みフラグビット  
 1 = チャンネル ソースポインタがソースの 1/2 エンプティ位置に達した (CHSPTR = CHSSIZ/2)  
 0 = この割り込みは保留されていない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 10-9: DCHxINT: DMA チャンネル x 割り込み制御レジスタ (続き)

- bit 5     **CHDDIF:** チャンネル デスティネーション完了割り込みフラグビット  
1 = チャンネル デスティネーション ポインタがデスティネーションの終端に達した (CHDPTR = CHDSIZ)  
0 = この割り込みは保留されていない
- bit 4     **CHDHIF:** チャンネル デスティネーション 1/2 フル割り込みフラグビット  
1 = チャンネル デスティネーションポインタがデスティネーションの1/2フル位置に達した(CHDPTR=CHDSIZ/2)  
0 = この割り込みは保留されていない
- bit 3     **CHBCIF:** チャンネル ブロック転送完了割り込みフラグビット  
1 = ブロック転送が完了した (CHSSIZ または CHDSIZ バイト (どちらか大きい方) を転送した)、  
      またはパターンマッチ イベントが発生した  
0 = この割り込みは保留されていない
- bit 2     **CHCCIF:** チャンネル セル転送完了割り込みフラグビット  
1 = セル転送が完了した (CHCSIZ バイトを転送した)  
0 = この割り込みは保留されていない
- bit 1     **CHTAIF:** チャンネル転送中止割り込みフラグビット  
1 = CHAIRQ ビットで指定されている番号の割り込みを検出したため DMA 転送を中止した  
0 = この割り込みは保留されていない
- bit 0     **CHERIF:** チャンネル アドレスエラー割り込みフラグビット  
1 = チャンネル アドレスエラーを検出した (ソースまたはデスティネーションアドレスのどちらかが無効)  
0 = この割り込みは保留されていない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-10: DCHxSSA: DMA チャンネル x ソース開始アドレスレジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHSSA<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHSSA<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHSSA<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHSSA<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-0    **CHSSA<31:0>**: チャンネルソース開始アドレスビット  
 チャンネルソースの開始アドレスです。  
**Note:** これはソースの物理アドレスである事が必要です。

レジスタ 10-11: DCHxDSA: DMA チャンネル x デスティネーション開始アドレスレジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHDSA<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHDSA<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHDSA<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHDSA<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-0    **CHDSA<31:0>**: チャンネル デスティネーション開始アドレスビット  
 チャンネル デスティネーションの開始アドレスです。  
**Note:** これはデスティネーションの物理アドレスである事が必要です。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-12: DCHxSSIZ: DMA チャンネル x ソース サイズレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHSSIZ<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHSSIZ<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **CHSSIZ<15:0>**: チャンネル ソースサイズ ビット  
 1111111111111111 = ソースサイズは 65,535 バイト  
 .  
 .  
 0000000000000010 = ソースサイズは 2 バイト  
 0000000000000001 = ソースサイズは 1 バイト  
 0000000000000000 = ソースサイズは 65,536 バイト

レジスタ 10-13: DCHxDSIZ: DMA チャンネル x デスティネーション サイズレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHDSIZ<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHDSIZ<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **CHDSIZ<15:0>**: チャンネル デスティネーション サイズビット  
 1111111111111111 = デスティネーション サイズは 65,535 バイト  
 .  
 .  
 0000000000000010 = デスティネーション サイズは 2 バイト  
 0000000000000001 = デスティネーション サイズは 1 バイト  
 0000000000000000 = デスティネーション サイズは 65,536 バイト

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 10-14: DCHxSPTR: DMA チャンネル x ソースポインタ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	CHSPTR<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	CHSPTR<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **CHSPTR<15:0>**: チャンネル ソースポインタ ビット

1111111111111111 = ポインタはソースのバイト 65,535 を指している  
 .  
 .  
 .  
 0000000000000001 = ポインタはソースのバイト 1 を指している  
 0000000000000000 = ポインタはソースのバイト 0 を指している

**Note:** パターンマッチ検出モードでは、このレジスタはパターンマッチ検出時にリセットされます。

## レジスタ 10-15: DCHxDPTR: DMA チャンネル x デスティネーション ポインタレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	CHDPTR<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	CHDPTR<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **CHDPTR<15:0>**: チャンネル デスティネーション ポインタビット

1111111111111111 = ポインタはデスティネーションのバイト 65,535 を指している  
 .  
 .  
 .  
 0000000000000001 = ポインタはデスティネーションのバイト 1 を指している  
 0000000000000000 = ポインタはデスティネーションのバイト 0 を指している

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-16: DCHxCSIZ: DMA チャンネル x セルサイズ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHCSIZ<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHCSIZ<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15-0 **CHCSIZ<15:0>:** チャンネル セルサイズビット

1111111111111111 = イベントあたり 65,535 バイトを転送する  
 .  
 .  
 .  
 0000000000000010 = イベントあたり 2 バイトを転送する  
 0000000000000001 = イベントあたり 1 バイトを転送する  
 0000000000000000 = イベントあたり 65,536 バイトを転送する

レジスタ 10-17: DCHxCPTR: DMA チャンネル x セルポインタ レジスタ

ビットレ レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	CHCPTR<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	CHCPTR<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15-0 **CHCPTR<15:0>:** チャンネル セル転送進捗ポインタビット

1111111111111111 = 直前のイベントから現在までに 65,535 バイトを転送した  
 .  
 .  
 .  
 0000000000000001 = 直前のイベントから現在までに 1 バイトを転送した  
 0000000000000000 = 直前のイベントから現在までに 0 バイトを転送した

**Note:** パターンマッチ検出モードでは、このレジスタはパターンマッチ検出時にリセットされます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 10-18: DCHxDAT: DMA チャンネル x パターンデータ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHPDAT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHPDAT<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **CHPDAT<15:0>**: チャンネルデータ レジスタビット

パターンマッチ転送終了モードの場合:

パターンマッチ発生時に転送を終了させるには、パターンマッチの判定に使うデータをこのレジスタに書き込んでおく必要があります。

その他の全てのモードの場合:

このレジスタは使いません。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:



# PIC32MZ グラフィック (DA) ファミリ

## 11.0 OTG (On-The-Go) 対応のハイ スピード USB

**Note:** 本書はPIC32MZ DAの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 51. Hi-Speed USB with On-The-Go (OTG)』(DS60001326)を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/PIC32](http://www.microchip.com/PIC32)) からダウンロードできます。

本 USB モジュールは、USB 2.0 組み込みホスト / デバイス / OTG を最小限の外付け部品で実装できるアナログおよびデジタル構成要素を内蔵しています。

このモジュールは全ての動作モードでハイスピード、フルスピード、ロースピードをサポートします。このモジュールは、ホストモードにおいて組み込みホストとしての使用を目的とするため、UHCI または OHCI コントローラを実装していません。

本 USB モジュールは RAM コントローラ、パケット エンコード / デコード ブロック、UTM 同期ブロック、エンドポイント制御ブロック、専用 USB DMA コントローラ、プルアップ / プルダウン抵抗、レジスタ インターフェイスで構成されます。PIC32 の USB OTG モジュールのブロック図を [図 11-1](#) に示します。

本 USB モジュールは以下の特長を備えます。

- ホストとデバイスでハイスピード、フルスピード、ロースピードをサポート
- 1 つまたは複数のハイスピード、フルスピード、ロースピード デバイスで USB OTG をサポート
- 信号ラインのプルアップ / プルダウン抵抗を内蔵
- VBUS 監視用にアナログ コンパレータを内蔵
- USB トランシーバを内蔵
- ハードウェアで実行するトランザクション ハンドシェイク
- 内蔵 8 チャンネル DMA によるシステム RAM とフラッシュメモリへのアクセス
- エンドポイント 0 に加えて 7 個の送信エンドポイントと 7 個の受信エンドポイント
- セッション リクエスト プロトコル (SRP) とホスト ネゴシエーション プロトコル (HNP) をサポート
- サスペンドおよびレゾーム信号をサポート
- 動的な FIFO サイズ
- FIFO用の内蔵RAM (FIFO用にシステムRAMは不要)
- LPM (Link Power Management) をサポート

**Note 1:** 他のサードパーティ仕様や技術と同様に、USB の実装と使用には使用許諾が必要とされる場合があります。詳細は USB implementers forum (USB-IF) 等を参照してください。使用許諾義務に関する調査と履行は全てユーザの責任です。

**2:** USB モジュールを使う場合、プライマリオシレータ (Posc) の周波数は 12 MHz または 24 MHz に制限されます。

図 11-1: PIC32MZ DA ファミリ USB インターフェイスのブロック図

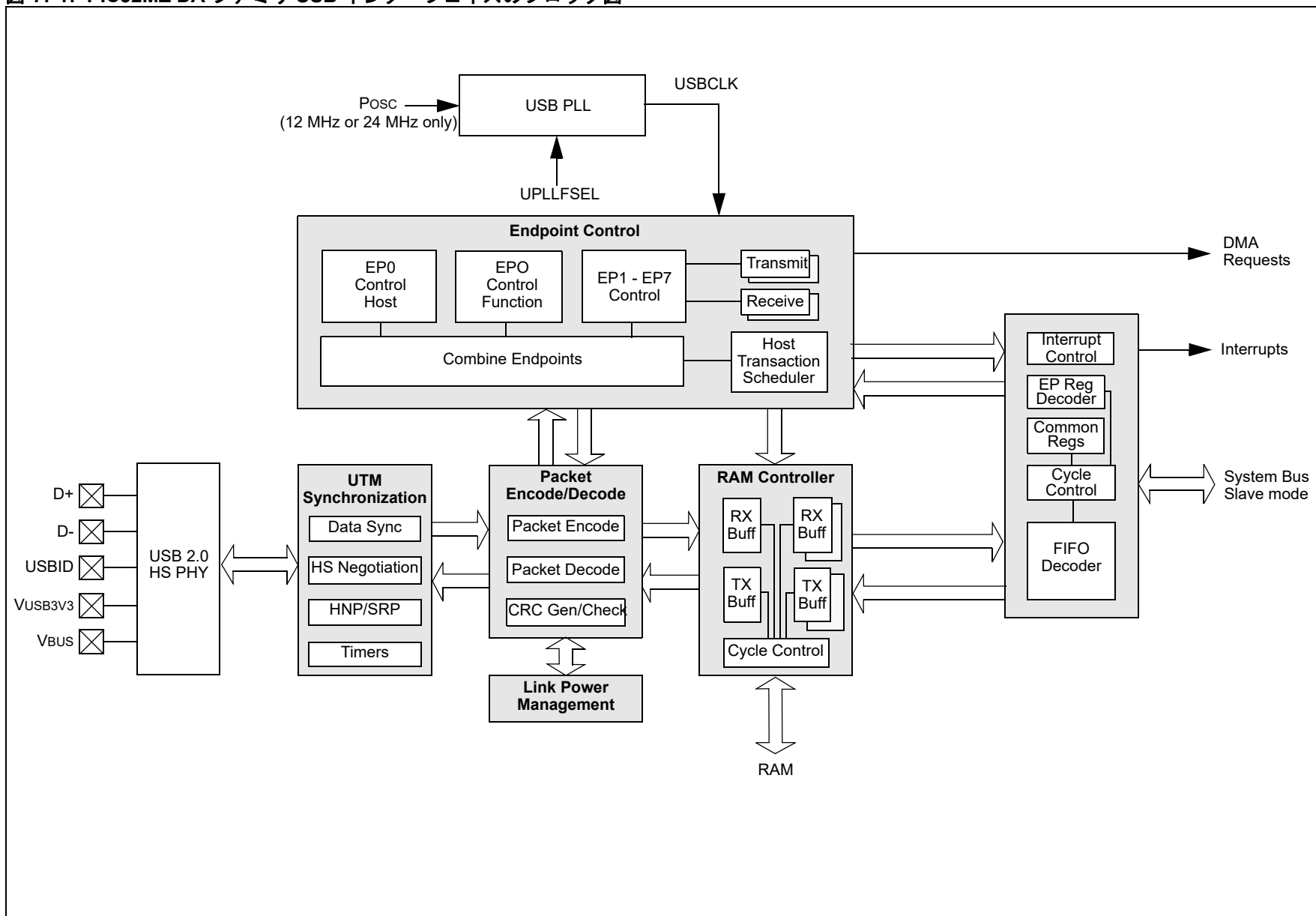




表 11-1: USB のレジスタマップ 1 (続き)

レジスタ名 (BFRE_#)	ビットレンジ	Bit																リセット
		31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
3028 USB FIFO2	31:16	DATA<31:16>																0000
	15:0	DATA<15:0>																0000
302C USB FIFO3	31:16	DATA<31:16>																0000
	15:0	DATA<15:0>																0000
3030 USB FIFO4	31:16	DATA<31:16>																0000
	15:0	DATA<15:0>																0000
3034 USB FIFO5	31:16	DATA<31:16>																0000
	15:0	DATA<15:0>																0000
3038 USB FIFO6	31:16	DATA<31:16>																0000
	15:0	DATA<15:0>																0000
303C USB FIFO7	31:16	DATA<31:16>																0000
	15:0	DATA<15:0>																0000
3060 USBOTG	31:16	—	—	—	RXDPB	RXFIFOSZ<3:0>			—	—	—	TXDPB	TXFIFOSZ<3:0>				0000	
	15:0	—	—	—	—	—	—	TXEDMA	RXEDMA	BDEV	FSDEV	LSDEV	VBUS<1:0>		HOSTMODE	HOSTREQ	SESSION	0088
3064 USB FIFOA	31:16	—	—	—	—	RXFIFOAD<12:0>											0000	
	15:0	—	—	—	—	TXFIFOAD<12:0>											0000	
306C USB HWVER	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
	15:0	RC	VERMAJOR<4:0>				VERMINOR<9:0>											0800
3078 USB INFO	31:16	VPLEN<7:0>						WTCON<3:0>			WTID<3:0>						3c5c	
	15:0	DMACHANS<3:0>			RAMBITS<3:0>			RXENDPTS<3:0>			TXENDPTS<3:0>						8c77	
307C USB EOFRST	31:16	—	—	—	—	—	—	NRSTX	NRST	LSEOF<7:0>							0072	
	15:0	FSEOF<7:0>								HSEOF<7:0>							7780	
3080 USB E0TXA	31:16	—	TXHUBPRT<6:0>						MULTTRAN	TXHUBADD<6:0>						0000		
	15:0	—	—	—	—	—	—	—	—	TXFADDR<6:0>						0000		
3084 USB E0RXA	31:16	—	RXHUBPRT<6:0>						MULTTRAN	RXHUBADD<6:0>						0000		
	15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
3088 USB E1TXA	31:16	—	TXHUBPRT<6:0>						MULTTRAN	TXHUBADD<6:0>						0000		
	15:0	—	—	—	—	—	—	—	—	TXFADDR<6:0>						0000		
308C USB E1RXA	31:16	—	RXHUBPRT<6:0>						MULTTRAN	RXHUBADD<6:0>						0000		
	15:0	—	—	—	—	—	—	—	—	RXFADDR<6:0>						0000		
3090 USB E2TXA	31:16	—	TXHUBPRT<6:0>						MULTTRAN	TXHUBADD<6:0>						0000		
	15:0	—	—	—	—	—	—	—	—	TXFADDR<6:0>						0000		
3094 USB E2RXA	31:16	—	RXHUBPRT<6:0>						MULTTRAN	RXHUBADD<6:0>						0000		
	15:0	—	—	—	—	—	—	—	—	RXFADDR<6:0>						0000		
3098 USB E3TXA	31:16	—	TXHUBPRT<6:0>						MULTTRAN	TXHUBADD<6:0>						0000		
	15:0	—	—	—	—	—	—	—	—	TXFADDR<6:0>						0000		

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: デバイスマード  
 2: ホストモード  
 3: エンドポイント 0 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 0)  
 4: エンドポイント 1 ~ 7 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 1 ~ 7)

表 11-1: USB のレジスタマップ 1 (続き)

アドレス (BF8#)	レジスタ名	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
309C	USB E3RXA	31:16	—	RXHUBPRT<6:0>							MULTTRAN	RXHUBADD<6:0>							0000
		15:0	—	—	—	—	—	—	—	—	—	RXFADDR<6:0>							0000
30A0	USB BE4TXA	31:16	—	TXHUBPRT<6:0>							MULTTRAN	TXHUBADD<6:0>							0000
		15:0	—	—	—	—	—	—	—	—	—	TXFADDR<6:0>							0000
30A4	USB E4RXA	31:16	—	RXHUBPRT<6:0>							MULTTRAN	RXHUBADD<6:0>							0000
		15:0	—	—	—	—	—	—	—	—	—	RXFADDR<6:0>							0000
30A8	USB E5TXA	31:16	—	TXHUBPRT<6:0>							MULTTRAN	TXHUBADD<6:0>							0000
		15:0	—	—	—	—	—	—	—	—	—	TXFADDR<6:0>							0000
30AC	USB E5RXA	31:16	—	RXHUBPRT<6:0>							MULTTRAN	RXHUBADD<6:0>							0000
		15:0	—	—	—	—	—	—	—	—	—	RXFADDR<6:0>							0000
30B0	USB E6TXA	31:16	—	TXHUBPRT<6:0>							MULTTRAN	TXHUBADD<6:0>							0000
		15:0	—	—	—	—	—	—	—	—	—	TXFADDR<6:0>							0000
30B4	USB E6RXA	31:16	—	RXHUBPRT<6:0>							MULTTRAN	RXHUBADD<6:0>							0000
		15:0	—	—	—	—	—	—	—	—	—	RXFADDR<6:0>							0000
30B8	USB E7TXA	31:16	—	TXHUBPRT<6:0>							MULTTRAN	TXHUBADD<6:0>							0000
		15:0	—	—	—	—	—	—	—	—	—	TXFADDR<6:0>							0000
30BC	USB E7RXA	31:16	—	RXHUBPRT<6:0>							MULTTRAN	RXHUBADD<6:0>							0000
		15:0	—	—	—	—	—	—	—	—	—	RXFADDR<6:0>							0000
3100	USB E0CSR0	31:16	USBIE0CSR0 内の同じビットによりインデックス付けされる														0000		
		15:0															0000		
3108	USB E0CSR2	31:16	USBIE0CSR2 内の同じビットによりインデックス付けされる														0000		
		15:0															0000		
310C	USB E0CSR3	31:16	USBIE0CSR3 内の同じビットによりインデックス付けされる														0000		
		15:0															0000		
3110	USB E1CSR0	31:16	USBIE1CSR0 内の同じビットによりインデックス付けされる														0000		
		15:0															0000		
3114	USB E1CSR1	31:16	USBIE1CSR1 内の同じビットによりインデックス付けされる														0000		
		15:0															0000		
3118	USB E1CSR2	31:16	USBIE1CSR2 内の同じビットによりインデックス付けされる														0000		
		15:0															0000		
311C	USB E1CSR3	31:16	USBIE1CSR3 内の同じビットによりインデックス付けされる														0000		
		15:0															0000		
3120	USB E2CSR0	31:16	USBIE2CSR0 内の同じビットによりインデックス付けされる														0000		
		15:0															0000		
3124	USB E2CSR1	31:16	USBIE2CSR1 内の同じビットによりインデックス付けされる														0000		
		15:0															0000		

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note  
 1: デバイスモード  
 2: ホストモード  
 3: エンドポイント 0 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 0)  
 4: エンドポイント 1 ~ 7 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 1 ~ 7)

表 11-1: USB のレジスタマップ 1 (続き)

アドレス (BF8E_#)	レジスタ名	ビットレンジ	Bit														リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	
3128	USB E2CSR2	31:16	USBIE2CSR2 内の同じビットによりインデックス付けされる														0000
		15:0															0000
312C	USB E2CSR3	31:16	USBIE2CSR3 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3130	USB E3CSR0	31:16	USBIE3CSR0 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3134	USB E3CSR1	31:16	USBIE3CSR1 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3138	USB E3CSR2	31:16	USBIE3CSR2 内の同じビットによりインデックス付けされる														0000
		15:0															0000
313C	USB E3CSR3	31:16	USBIE3CSR3 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3140	USB E4CSR0	31:16	USBIE4CSR0 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3144	USB E4CSR1	31:16	USBIE4CSR1 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3148	USB E4CSR2	31:16	USBIE4CSR2 内の同じビットによりインデックス付けされる														0000
		15:0															0000
314C	USB E4CSR3	31:16	USBIE4CSR3 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3150	USB E5CSR0	31:16	USBIE5CSR0 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3154	USB E5CSR1	31:16	USBIE5CSR1 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3158	USB E5CSR2	31:16	USBIE5CSR2 内の同じビットによりインデックス付けされる														0000
		15:0															0000
315C	USB E5CSR3	31:16	USBIE5CSR3 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3160	USB E6CSR0	31:16	USBIE6CSR0 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3164	USB E6CSR1	31:16	USBIE6CSR1 内の同じビットによりインデックス付けされる														0000
		15:0															0000
3168	USB E6CSR2	31:16	USBIE6CSR2 内の同じビットによりインデックス付けされる														0000
		15:0															0000
316C	USB E6CSR3	31:16	USBIE6CSR3 内の同じビットによりインデックス付けされる														0000
		15:0															0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note  
 1: デバイスマード  
 2: ホストモード  
 3: エンドポイント 0 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 0)  
 4: エンドポイント 1 ~ 7 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 1 ~ 7)

表 11-1: USB のレジスタマップ 1 (続き)

アドレス (BF8E#)	レジスタ名	ビットレンジ	Bit															リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
3170	USB E7CSR0	31:16	USBIE7CSR0 内の同じビットによりインデックス付けされる															0000	
		15:0																0000	
3174	USB E7CSR1	31:16	USBIE7CSR1 内の同じビットによりインデックス付けされる															0000	
		15:0																0000	
3178	USB E7CSR2	31:16	USBIE7CSR2 内の同じビットによりインデックス付けされる															0000	
		15:0																0000	
317C	USB E7CSR3	31:16	USBIE7CSR3 内の同じビットによりインデックス付けされる															0000	
		15:0																0000	
3200	USB DMAINT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	DMA8IF	DMA7IF	DMA6IF	DMA5IF	DMA4IF	DMA3IF	DMA2IF	DMA1IF	0000	
3204	USB DMA1C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	DMABRSTM<1:0>	DMAERR	DMAEP<3:0>			DMAIE	DMAMODE	DMADIR	DMAEN	0000	
3208	USB DMA1A	31:16	DMAADDR<31:16>															0000	
		15:0	DMAADDR<15:0>															0000	
320C	USB DMA1N	31:16	DMACOUNT<31:16>															0000	
		15:0	DMACOUNT<15:0>															0000	
3214	USB DMA2C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	DMABRSTM<1:0>	DMAERR	DMAEP<3:0>			DMAIE	DMAMODE	DMADIR	DMAEN	0000	
3218	USB DMA2A	31:16	DMAADDR<31:16>															0000	
		15:0	DMAADDR<15:0>															0000	
321C	USB DMA2N	31:16	DMACOUNT<31:16>															0000	
		15:0	DMACOUNT<15:0>															0000	
3224	USB DMA3C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	DMABRSTM<1:0>	DMAERR	DMAEP<3:0>			DMAIE	DMAMODE	DMADIR	DMAEN	0000	
3228	USB DMA3A	31:16	DMAADDR<31:16>															0000	
		15:0	DMAADDR<15:0>															0000	
322C	USB DMA3N	31:16	DMACOUNT<31:16>															0000	
		15:0	DMACOUNT<15:0>															0000	
3234	USB DMA4C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	DMABRSTM<1:0>	DMAERR	DMAEP<3:0>			DMAIE	DMAMODE	DMADIR	DMAEN	0000	
3238	USB DMA4A	31:16	DMAADDR<31:16>															0000	
		15:0	DMAADDR<15:0>															0000	
323C	USB DMA4N	31:16	DMACOUNT<31:16>															0000	
		15:0	DMACOUNT<15:0>															0000	
3244	USB DMA5C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	DMABRSTM<1:0>	DMAERR	DMAEP<3:0>			DMAIE	DMAMODE	DMADIR	DMAEN	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note  
 1: デバイスモード  
 2: ホストモード  
 3: エンドポイント 0 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 0)  
 4: エンドポイント 1 ~ 7 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 1 ~ 7)

表 11-1: USB のレジスタマップ 1 (続き)

レジスタ名 (BFRE#)	レジスタ名	ビットレンジ	Bit														リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
3248	USB DMA5A	31:16	DMAADDR<31:16>														0000	
		15:0	DMAADDR<15:0>														0000	
324C	USB DMA5N	31:16	DMACOUNT<31:16>														0000	
		15:0	DMACOUNT<15:0>														0000	
3254	USB DMA6C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	DMABRSTM<1:0>	DMAERR	DMAEP<3:0>			DMAIE	DMAMODE	DMADIR	DMAEN	0000
3258	USB DMA6A	31:16	DMAADDR<31:16>														0000	
		15:0	DMAADDR<15:0>														0000	
325C	USB DMA6N	31:16	DMACOUNT<31:16>														0000	
		15:0	DMACOUNT<15:0>														0000	
3264	USB DMA7C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	DMABRSTM<1:0>	DMAERR	DMAEP<3:0>			DMAIE	DMAMODE	DMADIR	DMAEN	0000
3268	USB DMA7A	31:16	DMAADDR<31:16>														0000	
		15:0	DMAADDR<15:0>														0000	
326C	USB DMA7N	31:16	DMACOUNT<31:16>														0000	
		15:0	DMACOUNT<15:0>														0000	
3274	USB DMA8C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	DMABRSTM<1:0>	DMAERR	DMAEP<3:0>			DMAIE	DMAMODE	DMADIR	DMAEN	0000
3278	USB DMA8A	31:16	DMAADDR<31:16>														0000	
		15:0	DMAADDR<15:0>														0000	
327C	USB DMA8N	31:16	DMACOUNT<31:16>														0000	
		15:0	DMACOUNT<15:0>														0000	
3304	USB E1RPC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RQPCTCNT<15:0>														0000	
3308	USB E2RPC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RQPCTCNT<15:0>														0000	
330C	USB E3RPC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RQPCTCNT<15:0>														0000	
3310	USB E4RPC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RQPCTCNT<15:0>														0000	
3314	USB E5RPC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RQPCTCNT<15:0>														0000	
3318	USB E6RPC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RQPCTCNT<15:0>														0000	
331C	USB E7RPC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RQPCTCNT<15:0>														0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: デバイスマード  
 2: ホストモード  
 3: エンドポイント 0 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 0)  
 4: エンドポイント 1 ~ 7 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 1 ~ 7)



表 11-1: USB のレジスタマップ 1 (続き)

レジスタアドレ (BF88_#)	レジスタ名	ビットレンジ	Bit															全リセット			
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0		
3340	USB DPBFD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
3344	USB TMCON1	31:16	THHSRTN<15:0>															05B6			
		15:0	TUCH<15:0>															4074			
3348	USB TMCON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	THSBT<3:0>															0000			
3360	USB LPMR1	31:16	—	—	LPM ERRIE	LPM RESIE	LPMACKIE	LPMNYIE	LPMSTIE	LPMTOIE	—	—	—	—	LPMNAK <sup>(1)</sup>	LPMEN<1:0>		LPMRES	LPMXMT	0000	
		15:0	ENDPOINT<3:0>			—	—	—	—	RMTWAK	HIRD<3:0>			LNKSTATE<3:0>			0000				
3364	USB LMPR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	LPMFADDR<6:0>					—	—	LPMERR <sup>(1)</sup>	—	—	LPMRES	LPMNC	LPMACK	LPMNY	LPMST	0000			
																				0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note  
 1: デバイスモード  
 2: ホストモード  
 3: エンドポイント 0 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 0)  
 4: エンドポイント 1 ~ 7 の定義 (ENDPOINT<3:0> (USBCSR<19:16>) = 1 ~ 7)

表 11-2: USB のレジスタマップ 2

レジスタアドレ (BF88_#)	レジスタ名	ビットレンジ	Bit															全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0	
4000	USB CRCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 11-1: USBCSR0: USB 制御ステータスレジスタ 0

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	R-0, HS EP7TXIF	R-0, HS EP6TXIF	R-0, HS EP5TXIF	R-0, HS EP4TXIF	R-0, HS EP3TXIF	R-0, HS EP2TXIF	R-0, HS EP1TXIF	R-0, HS EP0IF
	R/W-0 ISOUPD	R/W-0 SOFTCONN	R/W-1 HSEN	R-0, HS HSMODE	R-0 RESET	R/W-0 RESUME	R-0, HC SUSPMODE	R/W-0 SUSPEN
15:8	—	—	—	—	—	—	—	—
	U-0 —	R/W-0 —	R/W-0 —	R/W-0 —	R/W-0 —	R/W-0 —	R/W-0 —	R/W-0 —
7:0	FUNC<6:0>							
	—	—	—	—	—	—	—	—

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

bit 31-24 **未実装:** 「0」として読み出し

bit 23-17 **EP7TXIF:EP1TXIF:** エンドポイント「n」TX 割り込みフラグビット

- 1 = エンドポイントには対応が必要な送信割り込みが存在する
- 0 = 割り込みイベントは発生していない

bit 16 **EP0IF:** エンドポイント 0 割り込みビット

- 1 = エンドポイント 0 には対応が必要な割り込みが存在する
- 0 = 割り込みイベントは発生していない

バイトで読み出すと、全ての EPxTX および EP0 ビットはクリアされます。これらのビットが誤ってクリアされないようにするため、これらのビットはこのレジスタ内の他のビットと一緒に読み出さなければなりません。

bit 15 **ISOUPD:** ISO 更新ビット (デバイスモード専用: ホストモードではこのビットは未実装)

- 1 = USB モジュールは、パケットを送信する前に、TXPKTRDY ビットがセットされた時点から SOF トークンを待機する
- 0 = 動作を変更しない

このビットは、**デバイスモード**でアイソクロナス転送を実行するエンドポイントにのみ影響します。**ホストモード**では、このビットは未実装です。

bit 14 **SOFTCONN:** ソフト接続 / 切断機能選択ビット

- 1 = USB D+/D- ラインは有効 (アクティブ)
- 0 = USB D+/D- ラインは無効 (3 ステート)

このビットは**デバイスモード**でのみ使えます。

bit 13 **HSEN:** ハイスピード イネーブルビット

- 1 = ハブによってデバイスがリセットされた時に USB モジュールはハイスピード モードのネゴシエーションを行う
- 0 = USB モジュールはフルスピード モードでのみ動作する

bit 12 **HSMODE:** ハイスピード モード ステータスビット

- 1 = USB リセット時にハイスピード モードのネゴシエーションに成功した
- 0 = USB モジュールはハイスピード モード中ではない

**デバイスモード**では、このビットは USB リセットの完了時に有効になります。**ホストモード**では、このビットは RESET ビットのクリア時に有効になります。

bit 11 **RESET:** モジュール リセット ステータスビット

- 1 = バス上にリセット信号が存在する
- 0 = 通常のモジュール動作

**デバイスモード**では、このビットは読み出し専用です。**ホストモード**では、このビットは読み書き可能です。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-1: USBCSR0: USB 制御ステータスレジスタ 0 ( 続き )

bit 10 **RESUME:** サスペンドからのレジューム制御ビット

1 = デバイスがサスペンドモードの時にレジューム信号を生成する  
0 = レジューム信号の生成を停止する

**デバイスモード**では、ソフトウェアは 10 ms ( 最大 15 ms) 後にこのビットをクリアしてレジューム信号の生成を終了する必要があります。**ホストモード**では、ソフトウェアは 20 ms 後にこのビットをクリアする必要があります。

bit 9 **SUSPMODE:** サスペンドモードステータスビット

1 = USB モジュールはサスペンドモード中  
0 = USB モジュールは通常動作中

**デバイスモード**では、このビットは読み出し専用です。**ホストモード**では、このビットはソフトウェアによってセットでき、ハードウェアによってクリアされます。

bit 8 **SUSPEN:** サスペンドモードイネーブルビット

1 = サスペンドモードを有効にする  
0 = サスペンドモードを有効にしない

bit 7 **未実装:** 「0」として読み出し

bit 6-0 **FUNC<6:0>:** デバイス ファンクション アドレスビット

これらのビットは**デバイスモード**でのみ使えます。このフィールドには、SET\_ADDRESS コマンドを介して受信したアドレスが書き込まれます。このアドレスは、後続のトークンパケット内のファンクションアドレスをデコードするために使われます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 11-2: USBCSR1: USB 制御ステータスレジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-0
	EP7TXIE	EP6TXIE	EP5TXIE	EP4TXIE	EP3TXIE	EP2TXIE	EP1TXIE	EP0IE
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS	U-0
	EP7RXIF	EP6RXIF	EP5RXIF	EP4RXIF	EP3RXIF	EP2RXIF	EP1RXIF	—

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-24 **未実装:** 「0」として読み出し
- bit 23-17 **EP7TXIE:EP1TXIE:** エンドポイント「n」送信割り込みイネーブルビット
  - 1 = エンドポイント送信割り込みイベントを有効にする
  - 0 = エンドポイント送信割り込みイベントを有効にしない
- bit 16 **EP0IE:** エンドポイント0 割り込みイネーブルビット
  - 1 = エンドポイント0 送信割り込みイベントを有効にする
  - 0 = エンドポイント0 送信割り込みイベントを有効にしない
- bit 15-8 **未実装:** 「0」として読み出し
- bit 7-1 **EP7RXIF:EP1RXIF:** エンドポイント「n」受信割り込みビット
  - 1 = エンドポイントには対応が必要な受信イベントが存在する
  - 0 = 割り込みイベントは存在しない
- bit 0 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 11-3: USBCSR2: USB 制御ステータスレジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0
	VBUSIE	SESSRQIE	DISCONIE	CONNIE	SOFIE	RESETIE	RESUMEIE	SUSPIE
23:16	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS	R-0, HS
	VBUSIF	SESSRQIF	DISCONIF	CONNIF	SOFIF	RESETIF	RESUMEIF	SUSPIF
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	U-0
	EP7RXIE	EP6RXIE	EP5RXIE	EP4RXIE	EP3RXIE	EP2RXIE	EP1RXIE	—

<b>凡例:</b>	HS = ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

- bit 31 **VBUSIE:** VBUS エラー割り込みイネーブルビット  
1 = VBUS エラー割り込みを有効にする  
0 = VBUS エラー割り込みを無効にする
- bit 30 **SESSRQIE:** セッション要求割り込みイネーブルビット  
1 = セッション要求割り込みを有効にする  
0 = セッション要求割り込みを無効にする
- bit 29 **DISCONIE:** デバイス切断割り込みイネーブルビット  
1 = デバイス切断割り込みを有効にする  
0 = デバイス切断割り込みを無効にする
- bit 28 **CONNIE:** デバイス接続割り込みイネーブルビット  
1 = デバイス接続割り込みを有効にする  
0 = デバイス接続割り込みを無効にする
- bit 27 **SOFIE:** SOF (Start of Frame) 割り込みイネーブルビット  
1 = SOF イベント割り込みを有効にする  
0 = SOF イベント割り込みを無効にする
- bit 26 **RESETIE:** リセット/バブル割り込みイネーブルビット  
1 = リセット ( **デバイスモード** ) またはバブル ( **ホストモード** ) 時の割り込みを有効にする  
0 = リセット/バブル割り込みを無効にする
- bit 25 **RESUMEIE:** レジューム割り込みイネーブルビット  
1 = レジューム信号割り込みを有効にする  
0 = レジューム信号割り込みを無効にする
- bit 24 **SUSPIE:** サスペンド割り込みイネーブルビット  
1 = サスペンド信号割り込みを有効にする  
0 = サスペンド信号割り込みを無効にする
- bit 23 **VBUSIF:** VBUS エラー割り込みビット  
1 = セッション中に VBUS が VBUS 有効しきい値よりも低下した  
0 = 割り込まない
- bit 22 **SESSRQIF:** セッション要求割り込みビット  
1 = セッション要求信号を検出した  
0 = セッション要求は検出していない
- bit 21 **DISCONIF:** デバイス切断割り込みビット  
1 = **ホストモードの場合:** デバイスの切断を検出した  
**デバイスモードの場合:** セッションが終了した  
0 = デバイスの切断は検出していない
- bit 20 **CONNIF:** デバイス接続割り込みビット  
1 = **ホストモードの場合:** デバイスの接続を検出した  
0 = デバイスの接続は検出していない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-3: USBCSR2: USB 制御ステータスレジスタ 2 ( 続き )

- bit 19 **SOFIF:** SOF (Start of Frame) 割り込みビット  
1 = 新しいフレームが始まった  
0 = SOF は検出していない
- bit 18 **RESETIF:** リセット / バブル割り込みビット  
1 = **ホストモードの場合:** バブルを検出した  
**デバイスモードの場合:** バス上でリセット信号を検出した  
0 = リセット / バブルは検出していない
- bit 17 **RESUMEIF:** レジューム割り込みビット  
1 = USB モジュールがサスペンドモード中にバス上でレジューム信号を検出した  
0 = レジューム信号は検出していない
- bit 16 **SUSPIF:** サスペンド割り込みビット  
1 = **デバイスモードの場合:** バス上でサスペンド信号を検出した  
0 = サスペンド信号は検出していない
- bit 15-8 **未実装:** 「0」 として読み出し
- bit 7-1 **EP7RXIE:EP1RXIE:** エンドポイント「n」受信割り込みイネーブルビット  
1 = このエンドポイントに対して受信割り込みを有効にする  
0 = 受信割り込みを有効にしない
- bit 0 **未実装:** 「0」 として読み出し

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 11-4: USBCSR3: USB 制御ステータスレジスタ 3

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0 FORCEHST	R/W-0, HC FIFOACC	R/W-0 FORCEFS	R/W-0 FORCEHS	R/W-0 PACKET	R/W-0 TESTK	R/W-0 TESTJ	R/W-0 NAK
23:16	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 —	ENDPOINT<3:0>		
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	RFRMUM<10:8>		
7:0	RFRMNUM<7:0>							

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

- bit 31 **FORCEHST:** テストモード向けホスト強制選択ビット  
1 = 周辺デバイスへ接続されているかどうかに関係なく USB モジュールをホストモードに設定する  
0 = 通常動作
- bit 30 **FIFOACC:** テストモード向けエンドポイント 0 FIFO 強制転送ビット  
1 = エンドポイント 0 の TX FIFO 内のパケットをエンドポイント 0 の RX FIFO に転送する  
0 = 転送しない
- bit 29 **FORCEFS:** テストモード向けフルスピード モード強制選択ビット  
このビットは FORCEHST = 1 の場合にのみ有効です。  
1 = USB モジュールをフルスピード モードに設定する (FORCEHS = 1 の場合の動作は不確定)  
0 = FORCEHS = 0 の場合、USB モジュールをロースピード モードに設定する
- bit 28 **FORCEHS:** テストモード向けハイスピード モード強制選択ビット  
このビットは FORCEHST = 1 の場合にのみ有効です。  
1 = USB モジュールをハイスピード モードに設定する (FORCEFS = 1 の場合の動作は不確定)  
0 = FORCEFS = 0 の場合、USB モジュールをロースピード モードに設定する
- bit 27 **PACKET:** Test\_Packet テストモード選択ビット  
このビットは、モジュールがハイスピード モード中の場合にのみ有効です。  
1 = USB モジュールは 53 バイトのテストパケットをバスへ繰り返し送信する  
テストモードに移行する前に、テストパケットをエンドポイント 0 の FIFO バッファに書き込んでおく必要があります。  
0 = 通常動作
- bit 26 **TESTK:** Test\_K テストモード選択ビット  
1 = Test\_K テストモードに移行する  
USB モジュールは、バスへ連続する「K」を送信します。  
0 = 通常動作  
このビットは、USB モジュールがハイスピード モード中である場合にのみ有効です。
- bit 25 **TESTJ:** Test\_J テストモード選択ビット  
1 = Test\_J テストモードに移行する  
USB モジュールは、バスへ連続する「J」を送信します。  
0 = 通常動作  
このビットは、USB モジュールがハイスピード モード中である場合にのみ有効です。
- bit 24 **NAK:** Test\_SE0\_NAK テストモード選択ビット  
1 = Test\_SE0\_NAK テストモードに移行する  
USB モジュールはハイスピード モードのままですが、全ての有効 IN トークンに対して NAK を返します。  
0 = 通常動作  
このモードは、モジュールがハイスピード モード中の場合にのみ有効です。
- bit 23-20 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-4: USBCSR3: USB 制御ステータスレジスタ 3 ( 続き )

bit 19-16 **ENDPOINT<3:0>**: エンドポイント レジスタ 選択ビット

1111 = 予約済み

•

•

•

1000 = 予約済み

0111 = エンドポイント 7

•

•

•

0000 = エンドポイント 0

これらのビットは、エンドポイント レジスタ ( アドレス 3010 ~ 301F を介してアクセス ) を選択します。

bit 15-11 **未実装**: 「0」 として読み出し

bit 10-0 **RFRMNUM<10:0>**: 直近受信フレーム番号ビット



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 11-5: USBIE0CSR0: USB インデックス付きエンドポイント制御ステータス レジスタ 0  
(エンドポイント 0)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	R/W-0	R/W-0, HC	R/W-0	R/W-0, HC
	—	—	—	—	—	—	—	FLSHFIFO
23:16	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/C-0, HS	R/W-0, HS	R-0, HS	R-0	R-0
	SVCSETEND	SVCRPR	SENDSTALL	SETUPEND	DATAEND	SENTSTALL	TXPKTRDY	RXPKTRDY
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

<b>凡例:</b>	HC = ハードウェアでクリア HS = ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-28 **未実装:** 「0」として読み出し

bit 27 **DISPING:** PING トークン無効化制御ビット (ホストモード)

- 1 = ハイスピードコントロール転送のデータおよびステータス フェイズ中にUSBモジュールはPINGトークンを発行しない
- 0 = PING トークンを発行する

bit 26 **DTWREN:** データトグル書き込みイネーブルビット (ホストモード)

- 1 = エンドポイント 0 データトグルの現在の状態の書き込みを有効にする  
このビットは自動的にクリアされます。
- 0 = データトグルの書き込みを無効にする

bit 25 **DATATGGL:** データトグル ビット (ホストモード)

- このビットの読み値は、エンドポイント 0 データトグルの現在の状態を示します。
- DTWREN = 1 の場合、このビットに設定値を書き込みます。
- DTWREN = 0 の場合、このビットは読み出し専用です。

bit 24 **FLSHFIFO:** フラッシュ FIFO 制御ビット

- 1 = エンドポイント 0 FIFO から次に送信 / 読み出しするパケットをフラッシュする  
FIFO ポインタはリセットされ、TXPKTRDY/RXPKTRDY ビットはクリアされます。このビットは、動作完了時に自動的にクリアされます。このビットは、TXPKTRDY/RXPKTRDY = 1 の場合にのみ使います。
- 0 = フラッシュしない

bit 23 **SVCSETEND:** SETUPEND クリア制御ビット (デバイスモード)

- 1 = このレジスタ内の SETUPEND ビットをクリアする  
このビットは自動的にクリアされます。
- 0 = クリアしない

**NAKTMOUT:** NAK タイムアウト制御ビット (ホストモード)

- 1 = エンドポイント 0 は NAK 応答の受信が NAKLIM<4:0> ビット (USBICSR<28:24>) で設定された時間より長く続くと停止する
- 0 = エンドポイントは動作を継続できる

bit 22 **SVCRPR:** RXPKTRDY クリア制御ビット (デバイスモード)

- 1 = このレジスタ内の RXPKTRDY ビットをクリアする  
このビットは自動的にクリアされます。
- 0 = クリアしない

**STATPKT:** ステータス ステージ トランザクション制御ビット (ホストモード)

- 1 = TXPKTRDY または REQPKT ビットと同時にセットする事でステータス ステージ トランザクションを実行する
- 0 = ステータス ステージ トランザクションを実行しない

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-5: USBIE0CSR0: USB インデックス付きエンドポイント制御ステータス レジスタ 0 (エンドポイント 0) (続き)

- bit 21 **SENDSTALL:** ストール送信制御ビット (デバイスモード)  
1 = 実行中のトランザクションを終了し、STALL ハンドシェイクを送信する  
このビットは自動的にクリアされます。  
0 = STALL ハンドシェイクを送信しない
- REQPKT:** IN トランザクション要求制御ビット (ホストモード)  
1 = IN トランザクションを要求する  
このビットは RXPKT RDY ビットがセットされた時にクリアされます。  
0 = IN トランザクションを要求しない
- bit 20 **SETUPEND:** コントロール トランザクション早期終了ステータスビット (デバイスモード)  
1 = DATAEND ビットがセットされる前にコントロール トランザクションが終了した  
この時、割り込みが生成され、FIFO がフラッシュされます。  
0 = 通常動作  
このビットは、このレジスタ内の SVCSETEND ビットに「1」を書き込む事によりクリアされます。
- ERROR:** 応答なしエラー ステータスビット (ホストモード)  
1 = トランザクションを 3 回試行したが周辺デバイスから応答が返されなかった (割り込みが生成される)  
0 = このフラグをクリアする  
このビットをクリアするには、ソフトウェアでこのビットに「0」を書き込む必要があります。
- bit 19 **DATAEND:** データ終了制御ビット (デバイスモード)  
以下の場合、ソフトウェアはこのビットをセットします。  
• 最後のデータパケットに対して TXPKTRDY がセットされた  
• 最後のデータパケットを読み出した後に RXPKT RDY がクリアされた  
• 長さ 0 のデータパケットに対して TXPKTRDY がセットされた  
このビットはハードウェアによってクリアされます。
- SETUPPKT:** SETUP トークン送信制御ビット (ホストモード)  
1 = TXPKTRDY ビットと同時にセットする事で、モジュールはトランザクションに対して OUT トークンの  
代わりに SETUP トークンを送信する  
0 = 通常の OUT トークン動作  
このビットをセットするとデータトグルがクリアされます。
- bit 18 **SENTSTALL:** STALL 送信ステータスビット (デバイスモード)  
1 = STALL ハンドシェイクを送信した  
0 = ソフトウェアによるビットのクリア
- RXSTALL:** STALL ハンドシェイク受信ステータスビット (ホストモード)  
1 = STALL ハンドシェイクを受信した  
0 = ソフトウェアによるビットのクリア
- bit 17 **TXPKTRDY:** TX パケットレディー制御ビット  
1 = FIFO にデータパケットが格納されている  
このビットは自動的にクリアされます。  
0 = 送信可能なデータパケットは存在しない
- bit 16 **RXPKT RDY:** RX パケットレディーステータスビット  
1 = データパケットを受信した  
このビットがセットされた時に割り込みが生成されます (割り込みが有効にされている場合)。  
0 = データパケットは受信していない  
このビットは SVCRRPR をセットする事によりクリアされます。
- bit 15-0 **未実装:** 「0」 として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-6: USBIE0CSR2: USB インデックス付きエンドポイント制御ステータス レジスタ 2 (エンドポイント 0)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	NAKLIM<4:0>				
23:16	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
	SPEED<1:0>		—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	—	RXCNT<6:0>						

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28-24 **NAKLIM<4:0>**: エンドポイント 0 NAK 制限ビット

一連の NAK 応答の受信中に、このビットで指定した期間 ( フレーム / マイクロフレーム数 ( ハイスピード転送時 ) ) が過ぎると、エンドポイント 0 はタイムアウトします。

bit 23-22 **SPEED<1:0>**: 動作速度制御ビット

11 = ロースピード

10 = フルスピード

01 = ハイスピード

00 = 予約済み

bit 21-7 **未実装**: 「0」として読み出し

bit 6-0 **RXCNT<6:0>**: 受信バイト数ビット

エンドポイント 0 FIFO で受信したデータバイト数です。戻り値は FIFO の内容の変化に応じて変化し、その値は RXPCKTRDY ビットがセットされている時にのみ有効です。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 11-7: USBIE0CSR3: USB インデックス付きエンドポイント制御ステータス レジスタ 3  
(エンドポイント 0)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-x	R-x	R-0	R-x	R-x	R-x	R-1	R-0
	MPRXEN	MPTXEN	BIGEND	HBRXEN	HBTXEN	DYNFIFOS	SOFTCONE	UTMIDWID
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
-n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31 **MPRXEN:** 自動アマルガメーション オプションビット  
1 = バルクパケットの自動アマルガメーションを実行する  
0 = 自動アマルガメーションは実行しない
- bit 30 **MPTXEN:** 自動分割オプションビット  
1 = バルクパケットの自動分割を実行する  
0 = 自動分割は実行しない
- bit 29 **BIGEND:** バイト順オプションビット  
1 = ビッグ エンディアン  
0 = リトル エンディアン
- bit 28 **HBRXEN:** 高帯域 RX ISO オプションビット  
1 = 高帯域 RX ISO エンドポイントをサポートする  
0 = 高帯域 RX ISO エンドポイントはサポートしない
- bit 27 **HBTXEN:** 高帯域 TX ISO オプションビット  
1 = 高帯域 TX ISO エンドポイントをサポートする  
0 = 高帯域 TX ISO エンドポイントはサポートしない
- bit 26 **DYNFIFOS:** 動的 FIFO サイズ オプションビット  
1 = 動的な FIFO サイズをサポートする  
0 = 動的な FIFO サイズはサポートしない
- bit 25 **SOFTCONE:** ソフト接続 / 切断オプションビット  
1 = ソフト接続 / 切断機能をサポートする  
0 = ソフト接続 / 切断機能はサポートしない
- bit 24 **UTMIDWID:** UTMI+ データ幅オプションビット  
常に「0」(UTMI+ データ幅は常に 8 ビット)
- bit 23-0 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 11-8: USBIENCSR0: USB インデックス付きエンドポイント制御ステータス レジスタ 0  
(エンドポイント 1~7)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	AUTOSET	ISO —	MODEMode 1	DMAREQEN	FRCDATTG	DMAREQMD	— DATAWEN	— DATATGGL
23:16	R/W-0, HS	R/W-0, HC	R/W-0, HS	R/W-0	R/W-0	R/W-0, HS	R/W-0	R/W-0, HC
	INCOMPTX NAKTMOU	CLRDT	SENTSTALL RXSTALL	SENDSTALL SETUPPKT	FLUSH	UNDERRUN ERROR	FIFONE	TXPKTRDY
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MULT<4:0>					TXMAXP<10:8>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXMAXP<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
-n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31 **AUTOSET:** TXPKTRDY 自動セット制御ビット  
1 = 最大パケットサイズ(TXMAXPで指定)のデータがTX FIFOに格納された時にTXPKTRDYを自動的にセットする  
パケットサイズが最大パケットサイズ未満である場合、TXPKTRDYは手動でセットする必要があります。  
0 = パケットサイズに関係なくTXPKTRDYを常に手動でセットする
- bit 30 **ISO:** アイソクロナス TX エンドポイント イネーブルビット (デバイスモード)  
1 = エンドポイントをアイソクロナス転送向けに有効にする  
0 = エンドポイントをアイソクロナス転送向けに有効にしない(バルクまたはインタラプト転送向けに有効にする)  
このビットは、デバイスモード中でのみ効力を有します。ホストモードでは常に「0」を返します。
- bit 29 **MODE:** エンドポイント方向制御ビット  
1 = エンドポイントはTX  
0 = エンドポイントはRX  
このビットは、TX および RX トランザクションの両方に同じエンドポイント FIFO を使う場合にのみ効力を有します。
- bit 28 **DMAREQEN:** エンドポイント DMA 要求イネーブルビット  
1 = このエンドポイント向けに DMA 要求を有効にする  
0 = このエンドポイント向けに DMA 要求を無効にする
- bit 27 **FRCDATTG:** エンドポイント強制データトグル制御ビット  
1 = ACK を受信したかどうかに関係なく、強制的にエンドポイント データトグルを反転させ、データパケットを FIFO からクリアする  
0 = 上記の動作を強制しない
- bit 26 **DMAREQMD:** エンドポイント DMA 要求モード制御ビット  
1 = DMA 要求モード 1  
0 = DMA 要求モード 0  
このビットは、DMAREQEN ビットをクリアした後でクリアする必要があります。
- bit 25 **DATAWEN:** データトグル書き込みイネーブルビット (ホストモード)  
1 = TX エンドポイントのデータトグルの現在の状態 (DATATGGL) の書き込みを有効にする  
0 = DATATGGL ビットの書き込みを無効にする
- bit 24 **DATATGGL:** データトグル制御ビット (ホストモード)  
このビットの読み値は、TX エンドポイント データトグルの現在の状態を示します。DATAWEN = 1 の場合、このビットにはデータトグルの設定値が書き込まれます。DATAWEN = 0 の場合、このビットへの書き込みは無視されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-8: USBIENCSR0: USB インデックス付きエンドポイント制御ステータス レジスタ 0 (エンドポイント 1~7) (続き)

bit 23 **INCOMPTX:** 不完全 TX ステータスビット (デバイスモード)  
1 = 高帯域アイソクロナス転送において、大きな送信パケットが 2~3 個のパケットに分割されたが、全てのパケットを送信するために必要な数の IN トークンを受信しなかった  
0 = 通常動作  
アイソクロナス転送ではない場合、このビットは常に「0」を返します。

### **NAKTMOUT:** NAK タイムアウト ステータスビット (ホストモード)

1 = NAK 応答の受信が NAKLIM の設定よりも長く続いたため TX エンドポイントは停止した  
0 = ソフトウェアによる書き込みでこのビットをクリア

bit 22 **CLRDT:** データトグル クリア制御ビット

1 = エンドポイント データトグルを 0 にリセットする  
0 = データトグルをクリアしない

bit 21 **SENTSTALL:** STALL ハンドシェイク送信ステータスビット (デバイスモード)

1 = STALL ハンドシェイクを送信した  
FIFO はフラッシュされ、TXPKTRDY ビットはクリアされます。  
0 = ソフトウェアによる書き込みでこのビットをクリア

### **RXSTALL:** STALL 受信ビット (ホストモード)

1 = STALL ハンドシェイクを受信した  
処理中の DMA 要求は停止し、FIFO は完全にフラッシュされ、TXPKTRDY ビットはクリアされます。  
0 = ソフトウェアによる書き込みでこのビットをクリア

bit 20 **SENDSTALL:** STALL ハンドシェイク送信制御ビット (デバイスモード)

1 = IN トークンに対して STALL ハンドシェイクを発行する  
0 = ストール条件を終了する

エンドポイントがアイソクロナス転送用に使われている場合、このビットの効果はありません。

### **SETUPPKT:** 定義ビット (ホストモード)

1 = TXPKTRDY ビットと同時にセットする事で、トランザクションに対して OUT トークンではなく SETUP トークンを送信する  
これによりデータトグルはクリアされます。  
0 = トランザクションに対して通常の OUT トークンを送信する

bit 19 **FLUSH:** FIFO フラッシュ制御ビット

1 = エンドポイント TX の FIFO から最新の packets をフラッシュする  
FIFO ポインタはリセットされ、TXPKTRDY ビットはクリアされ、割り込みが生成されます。  
0 = FIFO をフラッシュしない

bit 18 **UNDERRUN:** アンダーラン ステータスビット (デバイスモード)

1 = TXPKTRDY ビットがセットされていない時に IN トークンを受信した  
0 = ソフトウェアによる書き込みでこのビットをクリア

### **ERROR:** ハンドシェイク障害ステータスビット (ホストモード)

1 = パケットの送信を 3 回試みたがハンドシェイク パケットを受信しなかった  
0 = ソフトウェアによる書き込みでこのビットをクリア

bit 17 **FIFONE:** FIFO 非エンプティ ステータスビット

1 = TX FIFO 内に 1 つ以上のパケットが存在する  
0 = TX FIFO はエンプティ

bit 16 **TXPKTRDY:** TX パケットレディー制御ビット

ソフトウェアは、FIFO にデータパケットを書き込んだ後にこのビットをセットします。このビットは、データパケットが送信されると自動的にクリアされます。このビットは、ダブルバッファリング FIFO に 2 つ目のパケットを書き込む前にも自動的にクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-8: USBIENCSR0: USB インデックス付きエンドポイント制御ステータス レジスタ 0 (エンドポイント 1 ~ 7) (続き)

bit 15-11 **MULT<4:0>**: 乗数制御ビット

アイソクロナス / インタラプト エンドポイントの場合、またはバルク エンドポイントでパケットを分割する場合、ペイロードサイズは TXMAXP に MULT+1 を掛け算する事で求められます。

バルク エンドポイントの場合、MULT は指定されたペイロードの「USB」パケットの数 (最大 32) を定義します。それらのパケットには、FIFO 内の 1 つのデータパケットが転送前に分割されて格納されます。このデータパケットは、TXMAXP によって指定されたペイロードの倍数である事が必要です。

ハイスピード モードで動作するアイソクロナス / インタラプト エンドポイントの場合、MULT は 1 つのマイクロフレーム内で実行可能なトランザクションの最大数 (2 または 3) を指定します。

bit 10-0 **TXMAXP<10:0>**: トランザクションあたり最大 TX ペイロード制御ビット

このフィールドは、1 回のトランザクションで送信する最大ペイロード (バイト数) を設定します。この値には、フルスピードおよびハイスピード動作のバルク、インタラプト、アイソクロナス転送に対する USB 仕様の制約が適用されます。

DMA モード 1 で割り込みを正しく生成するには、TXMAXP を偶数バイト数に設定する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 11-9: USBIENCSR1: USB インデックス付きエンドポイント制御ステータス レジスタ 1  
(エンドポイント1~7)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0, HC	R-0	R/W-0
	AUTOCLR	ISO AUTORQ	DMAREQEN	DISNYET PIDERR	DMAREQMD	— DATATWEN	— DATATGGL	INCOMPRX
23:16	R/W-0, HC	R/W-0, HS	R/W-0	R/W-0, HC	R-0, HS	R/W-0, HS	R-0, HS, HC	R/W-0, HS
	CLRDT	SENTSTALL RXSTALL	SENDSTALL REQPKT	FLUSH	DATAERR DERRNAKT	OVERRUN ERROR	FIFOFULL	RXPKTRDY
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MULT<4:0>					RXMAXP<10:8>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXMAXP<7:0>							

<b>凡例:</b>	HC = ハードウェアでクリア	HS = ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31 **AUTOCLR:** RXPKTRDY 自動クリア制御ビット

1 = RX FIFO から RXMAXP バイトの packets をアンロードした時に RXPKTRDY を自動的にクリアする  
アンロードした packets のサイズが最大 packets サイズより小さい場合、RXPKTRDY は手動でクリアする必要  
があります。DMA を使って RX FIFO をアンロードする場合、RXMAXP に関係なく、データは RX FIFO から  
4 バイトのチャンクとして読み出されます。

0 = RXPKTRDY は自動的にクリアしない

高帯域アイソクロナス エンドポイントの場合、このビットをセットしない必要があります。

bit 30 **ISO:** アイソクロナス エンドポイント制御ビット (デバイスモード)

1 = RX エンドポイントをアイソクロナス転送向けに有効にする

0 = RX エンドポイントをバルク/インタラプト転送向けに有効にする

**AUTORQ:** 自動 packets 要求制御ビット (ホストモード)

1 = RXPKTRDY ビットがクリアされた時に REQPKT ビットを自動的にセットする

0 = 自動的に packets を要求しない

このビットは、ショート packets を受信した時に自動的にクリアされます。

bit 29 **DMAREQEN:** DMA 要求イネーブル制御ビット

1 = RX エンドポイント向けの DMA 要求を有効にする

0 = RX エンドポイント向けの DMA 要求を無効にする

bit 28 **DISNYET:** NYET ハンドシェイク ディセーブル制御ビット /PID エラー ステータスビット (デバイスモード)

1 = バルク/インタラプト転送において NYET ハンドシェイクの送信を無効にする

受信に成功した全ての RX packets には ACK が返されます (FIFO がフルになった時も含む)。

0 = 通常動作

バルク/インタラプト転送の場合、このビットはハイスピード モード中のみ効力を有します。ハイスピー  
ドモードでは、このビットを全てのインタラプト エンドポイント向けにセットする必要があります。

**PIDERR:** PID エラー ステータスビット (ホストモード)

1 = ISO トランザクション中に受信 packets で PID エラーが発生した

0 = エラーは発生していない

bit 27 **DMAREQMD:** DMA 要求モード選択ビット

1 = DMA 要求モード 1

0 = DMA 要求モード 0



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-9: USBIENCSR1: USB インデックス付きエンドポイント制御ステータス レジスタ 1 (エンドポイント 1~7) (続き)

- bit 26 **DATATWEN:** データトグル書き込みイネーブル制御ビット (ホストモード)  
1 = DATATGGL は書き込み可能  
0 = DATATGGL は書き込み禁止
- bit 25 **DATATGGL:** データトグル ビット (ホストモード)  
このビットの読み値は、エンドポイント データトグルの現在の状態を示します。  
DATATWEN = 1 の場合、このビットにはデータトグルの設定値が書き込めます。  
DATATWEN = 0 の場合、このビットへの書き込みは無視されます。
- bit 24 **INCOMPRX:** 不完全パケット ステータスビット  
1 = 高帯域アイソクロナス/インタラプト転送中に RX FIFO 内のパケットを完全に受信しなかった (データの一部を受信しなかった)  
0 = ソフトウェアによる書き込みでこのビットをクリア  
アイソクロナス転送ではない場合、このビットは常に「0」を返します。
- bit 23 **CLRDT:** データトグル クリア制御ビット  
1 = エンドポイント データトグルを 0 にリセットする  
0 = エンドポイント データトグルはそのままにする
- bit 22 **SENTSTALL:** STALL ハンドシェイク ステータスビット (デバイスモード)  
1 = STALL ハンドシェイクを送信した  
0 = ソフトウェアによる書き込みでこのビットをクリア
- RXSTALL:** STALL ハンドシェイク受信ステータスビット (ホストモード)  
1 = STALL ハンドシェイクを受信した (割り込みが生成される)  
0 = ソフトウェアによる書き込みでこのビットをクリア
- bit 21 **SENDSTALL:** STALL ハンドシェイク制御ビット (デバイスモード)  
1 = STALL ハンドシェイクを発行する  
0 = ストール条件を終了する
- REQPKT:** IN トランザクション要求制御ビット (ホストモード)  
1 = IN トランザクションを要求する  
0 = 要求しない  
このビットは RXPKTRDY がセットされた時にクリアされます。
- bit 20 **FLUSH:** FIFO フラッシュ制御ビット  
1 = エンドポイント RX FIFO から次に読み出すパケットをフラッシュする  
FIFO ポインタはリセットされ、RXPKTRDY ビットはクリアされます。このビットは、RXPKTRDY がセットされている時にのみ使います。FIFO をダブルバッファリングする場合、FLUSH を 2 回セットしなくても FIFO は完全にクリアされます。  
0 = 通常の FIFO 動作  
このビットは自動的にクリアされます。
- bit 19 **DATAERR:** データパケット エラー ステータスビット (デバイスモード)  
1 = データパケットは CRC またはビットスタッフィング エラーを含んでいる  
0 = エラーを含んでいない  
このビットは RXPKTRDY がクリアされた時にクリアされます。このビットは、エンドポイントが ISO モードで動作している時のみ有効です。パルクモードでは、常に「0」を返します。
- DERRNAKT:** データエラー /NAK タイムアウト ステータスビット (ホストモード)  
1 = データパケットは CRC またはビットスタッフィング エラーを含んでいる  
パルクモードでは、NAK 応答の受信が NAK リミットとして設定された期間よりも長く続くと、RX エンドポイントは停止します。  
0 = データまたは NAK タイムアウト エラーは含んでいない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-9: USBIENCSR1: USB インデックス付きエンドポイント制御ステータス レジスタ 1 (エンドポイント 1~7) (続き)

bit 18 **OVERRUN:** データ オーバーラン ステータスビット (デバイスモード)

1 = OUT パケットを RX FIFO に転送できなかった

0 = ソフトウェアによる書き込みでこのビットをクリア

このビットは、エンドポイントが ISO モードで動作している時にのみ有効です。バルクモードでは、常に「0」を返します。

**ERROR:** 受信データパケットなしステータスビット (ホストモード)

1 = パケットの受信を 3 回試みたが、データパケットを受信しなかった (割り込みが生成される)

0 = ソフトウェアによる書き込みでこのビットをクリア

このビットは、RX エンドポイントがバルクまたはインタラプト モードで動作している時にのみ有効です。ISO モードでは、常に「0」を返します。

bit 17 **FIFOFULL:** FIFO フル ステータスビット

1 = FIFO がフルのため後続のパケットは格納できない

0 = RX FIFO 内には少なくとも 1 つの空き位置がある

bit 16 **RXPKTRDY:** データパケット受信ステータスビット

1 = データパケットを受信した (割り込みが生成される)

0 = このビットは、RX FIFO からパケットをアンロードした時にソフトウェアでクリアする必要があります。

bit 15-11 **MULT<4:0>:** 乗数制御ビット

アイソクロナス/インタラプト エンドポイントの場合、またはバルク エンドポイントでパケットを分割する場合、ペイロードサイズは TXMAXP に MULT+1 を掛け算する事で求められます。

バルク エンドポイントの場合、MULT は指定されたペイロードの「USB」パケットの数 (最大 32) を定義します。それらのパケットには、FIFO 内の 1 つのデータパケットが転送前に分割されて格納されます。このデータパケットは、TXMAXP によって指定されたペイロードの倍数である必要があります。

ハイスピード モードで動作するアイソクロナス/インタラプト エンドポイントの場合、MULT は 1 つのマイクロフレーム内で実行可能なトランザクションの最大数 (2 または 3) を指定します。

bit 10-0 **RXMAXP<10:0>:** トランザクションあたり最大 RX ペイロード制御ビット

このフィールドは、1 回のトランザクションで受信する最大ペイロード (バイト数) を設定します。この値には、フルスピードおよびハイスピード動作のバルク、インタラプト、アイソクロナス転送に対する USB 仕様の制約が適用されます。

DMA モード 1 で割り込みを正しく生成するには、RXMAXP を偶数バイト数に設定する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-10: USBIENCSR2: USB インデックス付きエンドポイント制御ステータス レジスタ 2 (エンドポイント 1~7)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TXINTERV<7:0>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPEED<1:0>			PROTOCOL<1:0>			TEP<3:0>		
15:8	U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
RXCNT<13:8>								
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RXCNT<7:0>								

<b>凡例:</b>	HC=ハードウェアでクリア HS = ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

### bit 31-24 TXINTERV<7:0>: エンドポイント TX ポーリング間隔 / NAK リミットビット (ホストモード)

インタラプトおよびアイソクロナス転送の場合、このフィールドはエンドポイントのポーリング間隔を定義します。バルク エンドポイントの場合、一連の NAK 応答の受信中に、このフィールドで指定した期間 (フレーム / マイクロフレーム数) が過ぎると、エンドポイントはタイムアウトします。

以下の表に、これらのビットの有効値と意味を示します。

転送タイプ	速度	有効値 (m)	意味
インタラプト	ロースピード フルスピード	0x01 ~ 0xFF	ポーリング間隔は「m」フレーム
	ハイスピード	0x01 ~ 0x10	ポーリング間隔は $2^{(m-1)}$ フレーム
アイソクロナス	フルスピード ハイスピード	0x01 ~ 0x10	ポーリング間隔は $2^{(m-1)}$ フレーム / マイクロフレーム
バルク	フルスピード ハイスピード	0x02 ~ 0x10	NAK リミットは $2^{(m-1)}$ フレーム / マイクロフレーム 値を「0」または「1」に設定すると NAK タイムアウト機能は無効になります。

### bit 23-22 SPEED<1:0>: TX エンドポイント動作速度制御ビット (ホストモード)

11 = ロースピード  
10 = フルスピード  
01 = ハイスピード  
00 = 予約済み

### bit 21-20 PROTOCOL<1:0>: TX エンドポイント プロトコル制御ビット

11 = インタラプト  
10 = バルク  
01 = アイソクロナス  
00 = コントロール

### bit 19-16 TEP<3:0>: TX ターゲット エンドポイント番号ビット

この値は、デバイス エニュメレーション中に USB モジュールに返される TX エンドポイント ディスクリプタ内のエンドポイント番号です。

### bit 15-14 未実装: 「0」として読み出し

### bit 13-0 RXCNT<13:0>: 受信バイト数ビット

エンドポイント RX FIFO で受信したデータバイトの数です。戻り値は FIFO の内容の変化に応じて変化し、その値は RXPCKTRDY がセットされている時のみ有効です。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-11: USBIENCSR3: USB インデックス付きエンドポイント制御ステータス レジスタ 3 (エンドポイント 1 ~ 7)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
	RXFIFOSZ<3:0>				TXFIFOSZ<3:0>			
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXINTERV<7:0>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SPEED<1:0>		PROTOCOL<1:0>		TEP<3:0>			

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

### bit 31-28 **RXFIFOSZ<3:0>**: 受信 FIFO サイズビット

1111 = 予約済み  
 1110 = 予約済み  
 1101 = 8192 バイト  
 1100 = 4096 バイト

•  
•  
•

0011 = 8 バイト  
 0010 = 予約済み  
 0001 = 予約済み  
 0000 = 予約済み、またはエンドポイントは未設定

このレジスタは、動的 FIOF サイズが選択されていない場合にのみ、上記の効力を有します。動的 FIFO サイズを使う場合、効力はありません。

### bit 27-24 **TXFIFOSZ<3:0>**: 送信 FIFO サイズビット

1111 = 予約済み  
 1110 = 予約済み  
 1101 = 8192 バイト  
 1100 = 4096 バイト

•  
•  
•

0011 = 8 バイト  
 0010 = 予約済み  
 0001 = 予約済み  
 0000 = 予約済み、またはエンドポイントは未設定

このレジスタは、動的 FIOF サイズが選択されていない場合にのみ、上記の効力を有します。動的 FIFO サイズを使う場合、効力はありません。

### bit 23-16 **未実装**: 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-11: USBIENCSR3: USB インデックス付きエンドポイント制御ステータス レジスタ 3 (エンドポイント 1 ~ 7) (続き)

bit 15-8 **RXINTERV<7:0>**: エンドポイント RX ポーリング間隔 / NAK リミットビット

インタラプトおよびアイソクロナス転送の場合、このフィールドはエンドポイントのポーリング間隔を定義します。バルク エンドポイントの場合、一連の NAK 応答の受信中に、このフィールドで指定した期間 (フレーム / マイクロフレーム数) が過ぎると、エンドポイントはタイムアウトします。

以下の表に、このフィールドの有効値と意味を示します。

転送タイプ	速度	有効値 (m)	意味
インタラプト	ロースピード フルスピード	0x01 ~ 0xFF	ポーリング間隔は「m」フレーム
	ハイスピード	0x01 ~ 0x10	ポーリング間隔は $2^{(m-1)}$ フレーム
アイソクロナス	フルスピード ハイスピード	0x01 ~ 0x10	ポーリング間隔は $2^{(m-1)}$ フレーム / マイクロフレーム
バルク	フルスピード ハイスピード	0x02 ~ 0x10	NAK リミットは $2^{(m-1)}$ フレーム / マイクロフレーム 値を「0」または「1」に設定すると NAK タイムアウト機能は無効になります。

bit 7-6 **SPEED<1:0>**: RX エンドポイント動作速度制御ビット

11 = ロースピード  
10 = フルスピード  
01 = ハイスピード  
00 = 予約済み

bit 5-4 **PROTOCOL<1:0>**: RX エンドポイント プロトコル制御ビット

11 = インタラプト  
10 = バルク  
01 = アイソクロナス  
00 = コントロール

bit 3-0 **TEP<3:0>**: RX ターゲット エンドポイント番号ビット

この値は、デバイス エnumeration中に USB モジュールに返される RX エンドポイント ディスクリプタ内のエンドポイント番号です。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-12: USBFIFOx: USB FIFO データレジスタ「x」(x = 0 ~ 7)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DATA<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DATA<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DATA<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DATA<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

### bit 31-0 DATA<31:0>: USB 送信 / 受信 FIFO データビット

このレジスタに書き込む事で、対応するエンドポイントの TxFIFO にデータを格納します。このレジスタを読み出す事で、対応するエンドポイントの RxFIFO からデータをアンロードします。

転送のビット幅は必要に応じて 8 ビット、16 ビット、32 ビットにできます (連続的にデータにアクセスする限り、異なるビット幅を任意に組み合わせる事ができます)。しかし、1 つのパケットに関連する全ての転送は同じビット幅にする必要があります (データをバイト、ワード、ダブルワードのいずれかで一貫して整列させるため)。奇数個のバイトまたはワードの転送を完了させるため、最後の転送のバイト数は他より少なくても構いません。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-13: USBOTG: USB OTG 制御 / ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	RXDPB	RXFIFOSZ<3:0>			
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TXDPB	TXFIFOSZ<3:0>			
15:8	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	TXEDMA	RXEDMA
7:0	R-1	R-0	R-0	R-0	R-0	R-0	R/W-0, HC	R/W-0
	BDEV	FSDEV	LSDEV	VBUS<1:0>		HOSTMODE	HOSTREQ	SESSION

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                    1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28 **RXDPB**: RX エンドポイント ダブルパケット バッファリング制御ビット

- 1 = ダブルパケット バッファリングをサポートする  
RXFIFOSZ で設定されたサイズを 2 倍にします。
- 0 = ダブルパケット バッファリングはサポートしない

bit 27-24 **RXFIFOSZ<3:0>**: RX エンドポイント FIFO パケットサイズ ビット

最大許容パケットサイズです (バルク / 高帯域パケットの場合、転送前に FIFO 内で分割する前のサイズ)。

1111 = 予約済み

•  
•  
•

1010 = 予約済み

- 1001 = 4096 バイト
- 1000 = 2048 バイト
- 0111 = 1024 バイト
- 0110 = 512 バイト
- 0101 = 256 バイト
- 0100 = 128 バイト
- 0011 = 64 バイト
- 0010 = 32 バイト
- 0001 = 16 バイト
- 0000 = 8 バイト

bit 23-21 **未実装**: 「0」として読み出し

bit 20 **TXDPB**: TX エンドポイント ダブルパケット バッファリング制御ビット

- 1 = ダブルパケット バッファリングをサポートする  
TXFIFOSZ で設定されたサイズを 2 倍にします。
- 0 = ダブルパケット バッファリングはサポートしない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-13: USBOTG: USB OTG 制御 / ステータス レジスタ (続き)

bit 19-16 **TXFIFOSZ<3:0>**: TX エンドポイント FIFO パケットサイズ ビット

最大許容パケットサイズです (バルク / 高帯域パケットの場合、転送前に FIFO 内で分割する前のサイズ)。

1111 = 予約済み

.

.

.

1010 = 予約済み

1001 = 4096 バイト

1000 = 2048 バイト

0111 = 1024 バイト

0110 = 512 バイト

0101 = 256 バイト

0100 = 128 バイト

0011 = 64 バイト

0010 = 32 バイト

0001 = 16 バイト

0000 = 8 バイト

bit 15-10 **未実装**: 「0」として読み出し

bit 9 **TXEDMA**: TX エンドポイント DMA アサート制御ビット

1 = エンドポイントに MAXP-8 バイトを書き込んだ時に、全ての IN エンドポイントに対して DMA\_REQ 信号を生成する (早期モード)

0 = エンドポイントに MAXP バイトを書き込んだ時に、全ての IN エンドポイントに対して DMA\_REQ 信号を生成する (通常モード)

bit 8 **RXEDMA**: RX エンドポイント DMA アサート制御ビット

1 = エンドポイントに MAXP-8 バイトを書き込んだ時に、全ての OUT エンドポイントに対して DMA\_REQ 信号を生成する (早期モード)

0 = エンドポイントに MAXP バイトを書き込んだ時に、全ての OUT エンドポイントに対して DMA\_REQ 信号を生成する (通常モード)

bit 7 **BDEV**: USB デバイスタイプ ビット

1 = USB は「B」デバイスとして動作している

0 = USB は「A」デバイスとして動作している

bit 6 **FSDEV**: フルスピード / ハイスピード デバイス検出ビット (ホストモード)

1 = ポートに接続されたフルスピードまたはハイスピード デバイスを検出した

0 = フルスピード デバイスもハイスピード デバイスも検出していない

bit 5 **LSDEV**: ロースピード デバイス検出ビット (ホストモード)

1 = ポートに接続されたロースピード デバイスを検出した

0 = ロースピード デバイスは検出していない

bit 4-3 **VBUS<1:0>**: VBUS レベル検出ビット

11 = VBUS 有効しきい値より高い

10 = AValid より高く、VBUS 有効しきい値より低い

11 = セッション終了しきい値より高く、AValid より低い

00 = セッション終了しきい値より低い

bit 2 **HOSTMODE**: ホストモード ビット

1 = USB モジュールはホストとして動作している

0 = USB モジュールはホストとして動作していない

bit 1 **HOSTREQ**: ホスト要求制御ビット

「B」デバイスのみ:

1 = USB モジュールは、サスペンドモードに移行した時にホスト ネゴシエーションを開始する  
このビットはホスト ネゴシエーションの完了時にクリアされます。

0 = ホスト ネゴシエーションは実行しない



# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-13: USBOTG: USB OTG 制御 / ステータス レジスタ ( 続き )

bit 0 **SESSION:** アクティブ セッション制御 / ステータスビット

「A」 デバイス:

1 = セッションを開始する

0 = セッションを終了する

「B」 デバイス:

1 = 読み出し: セッションは開始された ( または実行中 )

書き込み: セッション リクエスト プロトコルを開始する

0 = USB モジュールがサスペンドモード中の場合:

このビットをクリアするとソフトウェア切断が発生します。

サスペンドモード中ではない場合: 動作は未確定

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-14: USBFIFOA: USB FIFO アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	RXFIFOAD<12:8>				
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXFIFOAD<7:0>							
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TXFIFOAD<12:8>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXFIFOAD<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28-16 **RXFIFOAD<12:0>**: 受信エンドポイント FIFO アドレスビット

エンドポイント FIFO の開始アドレス (8 バイト単位で以下の通りに指定):

11111111111111 = 0xFFFF8

•  
•  
•

00000000000010 = 0x0010

00000000000001 = 0x0008

00000000000000 = 0x0000

bit 15-13 **未実装**: 「0」として読み出し

bit 12-0 **TXFIFOAD<12:0>**: 送信エンドポイント FIFO アドレスビット

エンドポイント FIFO の開始アドレス (8 バイト単位で以下の通りに指定):

11111111111111 = 0xFFFF8

•  
•  
•

00000000000010 = 0x0010

00000000000001 = 0x0008

00000000000000 = 0x0000

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-15: USBHWVER: USB ハードウェアバージョンレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R-0	R-0	R-0	R-0	R-1	R-0	R-0	R-0
	RC	VERMAJOR<4:0>					VERMINOR<9:8>	
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	VERMINOR<7:0>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15 **RC**: 候補版リリースビット

- 1 = USB モジュールは候補版リリースに基づく
- 0 = USB モジュールは完全リリース版に基づく

bit 14-10 **VERMAJOR<4:0>**: USB モジュール メジャーバージョン番号ビット

この読み出し専用値は、USB モジュールのメジャーバージョン番号を表します。

bit 9-0 **VERMINOR<9:0>**: USB モジュール マイナーバージョン番号ビット

この読み出し専用値は、USB モジュールのマイナーバージョン番号を表します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-16: USBINFO: USB 情報レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0
	VPLEN<7:0>							
23:16	R/W-0	R/W-1	R/W-0	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0
	WTCON<3:0>				WTID<3:0>			
15:8	R-1	R-0	R-0	R-0	R-1	R-1	R-0	R-0
	DMACHANS<3:0>				RAMBITS<3:0>			
7:0	R-0	R-1	R-1	R-1	R-0	R-1	R-1	R-1
	RXENDPTS<3:0>				TXENDPTS<3:0>			

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-24 **VPLEN<7:0>**: VBUS パルス充電時間ビット

VBUS パルス充電の期間を 546.1  $\mu$ s 単位で設定します。既定値設定は 32.77 ms です。

bit 23-20 **WTCON<3:0>**: フィルタ接続 / 切断制御ビット

フィルタを接続 / 切断するための待機時間を 533.3 ns 単位で設定します。既定値設定は 2.667  $\mu$ s です。

bit 19-6 **WTID<3:0>**: ID 遅延有効制御ビット

IDPULLUP がアサートされてから IDDIG を有効と見なすまでの遅延時間を 4.369 ms 単位で設定します。既定値設定は 52.43 ms です。

bit 15-12 **DMACHANS<3:0>**: DMA チャンネルビット

これらの読み出し専用ビットは、USB モジュール内の DMA チャンネルの数を示します。PIC32MZ DA ファミリの場合、この値は 8 です。

bit 11-8 **RAMBITS<3:0>**: RAM アドレスバス幅ビット

これらの読み出し専用ビットは、RAM アドレスバスの幅を示します。PIC32MZ DA ファミリの場合、この値は 12 です。

bit 7-4 **RXENDPTS<3:0>**: RX エンドポイント数ビット

この読み出し専用レジスタは、回路内の RX エンドポイントの数を示します。PIC32MZ DA ファミリの場合、この値は 7 です。

bit 3-0 **TXENDPTS<3:0>**: TX エンドポイント数ビット

この読み出し専用レジスタは、回路内の TX エンドポイントの数を示します。PIC32MZ DA ファミリの場合、この値は 7 です。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-17: USBE0FRST: USB EOF (End-of-Frame)/ ソフトリセット制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	NRSTX	NRST
23:16	R/W-0	R/W-1	R/W-1	R/W-1	R/W-0	R.W-0	R/W-1	R/W-0
	LSEOF<7:0>							
15:8	R/W-0	R/W-1	R/W-1	R/W-1	R/W-0	R.W-1	R/W-1	R/W-1
	FSEOF<7:0>							
7:0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R.W-0	R/W-0	R/W-0
	HSEOF<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-26 **未実装**: 「0」として読み出し

bit 25 **NRSTX**: XCLK 領域リセットビット

1 = XCLK(PHY によって受信データから復元されるクロック) 領域をリセットする  
 0 = 通常動作

bit 24 **NRST**: CLK 領域リセットビット

1 = CLK(周辺モジュールバスから復元されるクロック) 領域をリセットする  
 0 = 通常動作

bit 23-16 **LSEOF<7:0>**: ロースピード EOF ビット

これらのビットは、次のトランザクションの開始を止めるための EOF に先立つロースピード トランザクションの期間を 1.067  $\mu$ s 単位で設定します (既定値は 121.6  $\mu$ s)。

bit 15-8 **FSEOF<7:0>**: フルスピード EOF ビット

これらのビットは、次のトランザクションの開始を止めるための EOF に先立つフルスピード トランザクションの期間を 533.3 ns 単位で設定します (既定値は 63.46  $\mu$ s)。

bit 7-0 **HSEOF<7:0>**: ハイスピード EOF ビット

これらのビットは、次のトランザクションの開始を止めるための EOF に先立つハイスピード トランザクションの期間を 133.3 ns 単位で設定します (既定値は 17.07  $\mu$ s)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-18: USBExTXA: USB エンドポイント「x」送信アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	TXHUBPRT<6:0>						
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MULTTRAN	TXHUBADD<6:0>						
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	TXFADDR<6:0>						

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31    **未実装:** 「0」として読み出し

bit 30-24 **TXHUBPRT<6:0>:** TX ハブポートビット (ホストモード)

ロースピードまたはフルスピード デバイスがハイスピード USB 2.0 ハブを介してこのエンドポイントに接続された時、このフィールドはその USB 2.0 ハブのポート番号を記録します。

bit 23    **MULTTRAN:** TX ハブ複数トランスレータ ビット (ホストモード)

1 = USB 2.0 ハブは複数のトランザクション トランスレータを備えている  
 0 = USB 2.0 ハブはトランザクション トランスレータを1つだけ備えている

bit 22-16 **TXHUBADD<6:0>:** TX ハブ アドレスビット (ホストモード)

ロースピードまたはフルスピード デバイスがハイスピード USB 2.0 ハブを介してこのエンドポイントに接続された時、これらのビットはその USB 2.0 ハブのアドレスを記録します。

bit 15-7    **未実装:** 「0」として読み出し

bit 6-0    **TXFADDR<6:0>:** TX ファンクション アドレスビット (ホストモード)

対応するエンドポイントを介してアクセスするターゲット ファンクションのアドレスを指定します。これは、使用する各 TX エンドポイントに対して定義する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-19: USBExRXA: USB エンドポイント「x」受信アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXHUBPRT<6:0>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MULTTRAN RXHUBADD<6:0>							
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXFADDR<6:0>							

**凡例:** HC = ハードウェアでクリア HS = ハードウェアでセット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31 **未実装:** 「0」として読み出し

bit 30-24 **RXHUBPRT<6:0>:** RX ハブポートビット (ホストモード)

ロースピードまたはフルスピード デバイスがハイスピード USB 2.0 ハブを介してをこのエンドポイントに接続された時、このフィールドはその USB 2.0 ハブのポート番号を記録します。

bit 23 **MULTTRAN:** RX ハブ複数トランスレータ ビット (ホストモード)

1 = USB 2.0 ハブは複数のトランザクション トランスレータを備えている  
0 = USB 2.0 ハブはトランザクション トランスレータを1つだけ備えている

bit 22-16 **RXHUBADD<6:0>:** RX ハブ アドレスビット (ホストモード)

ロースピードまたはフルスピード デバイスがハイスピード USB 2.0 ハブを介してこのエンドポイントに接続された時、これらのビットはその USB 2.0 ハブのアドレスを記録します。

bit 15-7 **未実装:** 「0」として読み出し

bit 6-0 **RXFADDR<6:0>:** RX ファンクション アドレスビット (ホストモード)

対応するエンドポイントを介してアクセスするターゲット ファンクションのアドレスを指定します。これは、使用する各 RX エンドポイントに対して定義する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-20: USBDMAINT: USB DMA 割り込みレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS
	DMA8IF	DMA7IF	DMA6IF	DMA5IF	DMA4IF	DMA3IF	DMA2IF	DMA1IF

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-8 **未実装**: 「0」として読み出し

bit 7-0 **DMAxIF**: DMA チャンネル「x」割り込みビット

1 = DMA チャンネルで割り込みイベントが発生した

0 = 割り込みイベントは発生していない

このレジスタを読み出すと、全てのビットがクリアされます。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-21: USBDMAxC: USB DMA チャンネル「x」制御レジスタ (x = 1 ~ 8)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 DMABRSTM<1:0>	R/W-0	R/W-0 DMAERR
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DMAEP<3:0>				DMAIE	DMAMODE	DMADIR	DMAEN

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-11 **未実装**: 「0」として読み出し

bit 10-9 **DMABRSTM<1:0>**: DMA バーストモード選択ビット

11 = バーストモード 3: INCR16、INCR8、INCR4、未指定長  
 10 = バーストモード 2: INCR8、INCR4、未指定長  
 01 = バーストモード 1: INCR4、未指定長  
 00 = バーストモード 0: 未指定長

bit 8 **DMAERR**: バスエラービット

1 = 入力でバスエラーが見つかった  
 0 = ソフトウェアで書き込む事でエラーをクリアする

bit 7-4 **DMAEP<3:0>**: DMA エンドポイント割り当てビット

これらのビットは、DMA チャンネルが割り当てられたエンドポイントの番号を保持します。有効な値は 0 ~ 7 です。

bit 3 **DMAIE**: DMA 割り込みイネーブルビット

1 = このチャンネル向けに割り込みを有効にする  
 0 = このチャンネル向けに割り込みを無効にする

bit 2 **DMAMODE**: DMA 転送モードビット

1 = DMA モード 1 転送  
 0 = DMA モード 0 転送

bit 1 **DMADIR**: DMA 転送方向ビット

1 = DMA 読み出し (TX エンドポイント)  
 0 = DMA 書き込み (RX エンドポイント)

bit 0 **DMAEN**: DMA イネーブルビット

1 = DMA 転送を有効にして転送を開始する  
 0 = DMA 転送を無効にする

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-22: USBDMAxA: USB DMA チャンネル「x」メモリアドレス レジスタ (x = 1 ~ 8)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DMAADDR<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DMAADDR<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DMAADDR<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R-0
DMAADDR<7:0>								

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

#### bit 31-0 DMAADDR<31:0>: DMA メモリアドレス ビット

このレジスタは、対応する DMA チャンネルの現在のメモリアドレスを示します。初期化中にこのレジスタに書き込まれる初期メモリアドレス値の 4 による剰余は「0」である必要があります。このレジスタの下位 2 ビットは読み出し専用であり、ソフトウェアで設定する事はできません。DMA 転送の進捗 (転送バイト数) に応じてメモリアドレスはインクリメントします。

## レジスタ 11-23: USBDMAxN: USB DMA チャンネル「x」カウントレジスタ (x = 1 ~ 8)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DMACOUNT<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DMACOUNT<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DMACOUNT<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DMACOUNT<7:0>								

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

#### bit 31-0 DMACOUNT<31:0>: DMA 転送カウントビット

このレジスタは、DMA 転送の現在のカウント値を示します。転送の初期カウント値 (転送の総長) はソフトウェアで設定します。転送の進捗 (転送バイト数) に応じてこのカウント値はデクリメントします。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 11-24: USBExRPC: USB エンドポイント「x」要求パケットカウントレジスタ  
(ホストモード専用) (x = 1 ~ 7)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-0 RQPKTCNT<15:8>							
7:0	R/W-0 RQPKTCNT<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
-n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15-0 **RQPKTCNT<15:0>:** 要求パケット カウントビット

ブロック転送で転送するパケット (サイズは MAXP で指定) の数を設定します。このレジスタは、ホストモードで AUTOREQ がセットされている場合にのみ使えます。

レジスタ 11-25: USBDPBFD: USB ダブルパケットバッファ ディセーブル レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	R/W-0 EP7TXD	R/W-0 EP6TXD	R/W-0 EP5TXD	R/W-0 EP4TXD	R/W-0 EP3TXD	R/W-0 EP2TXD	R/W-0 EP1TXD	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	R/W-0 EP7RXD	R/W-0 EP6RXD	R/W-0 EP5RXD	R/W-0 EP4RXD	R/W-0 EP3RXD	R/W-0 EP2RXD	R/W-0 EP1RXD	U-0 —

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
-n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-17 **EP7TXD:EP1TXD:** TX エンドポイント「x」ダブルパケットバッファ ディセーブル ビット

1 = エンドポイント「x」の TX ダブルパケットバッファリングを無効にする  
0 = エンドポイント「x」の TX ダブルパケットバッファリングを有効にする

bit 16 **未実装:** 「0」として読み出し

bit 15-1 **EP7RXD:EP1RXD:** RX エンドポイント「x」ダブルパケットバッファ ディセーブル ビット

1 = エンドポイント「x」の RX ダブルパケットバッファリングを無効にする  
0 = エンドポイント「x」の RX ダブルパケットバッファリングを有効にする

bit 0 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-26: USBTMCON1: USB タイミング制御レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-1
THHSRTN<15:8>								
23:16	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0
THHSRTN<7:0>								
15:8	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TUCH<15:8>								
7:0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-0	R/W-1	R/W-0	R/W-0
TUCH<7:0>								

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **THHSRTN<15:0>**: ハイスピード レジューム信号遅延ビット

ホストとして動作している場合、これらのビットはハイスピード レジューム信号の終了から UTM 通常動作モードの有効化までの遅延時間を設定します。

bit 15-0 **TUCH<15:0>**: Chirp タイムアウト ビット

これらのビットは Chirp タイムアウトを設定します。この値を 4 倍した値は、タイムアウトが発生する前の USB モジュール クロックサイクル数を表します。

**Note:** このレジスタを使う事で、ハイスピード タイムアウトを USB 2.0 仕様が規定している最大値よりも大きな値に設定できません (USB モジュールは仕様に非準拠となります)。

## レジスタ 11-27: USBTMCON2: USB タイミング制御レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—								
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—								
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—								
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
THBST<3:0>								

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-4 **未実装**: 「0」として読み出し

bit 3-0 **THBST<3:0>**: ハイスピード タイムアウト加算値ビット

これらのビットは、最小ハイスピード タイムアウト期間 (736 ビット時間) に加算する値を表します。タイムアウト期間は、64 ハイスピード ビット時間 (133 ns) 刻みで延長できます。

**Note:** このレジスタを使う事で、ハイスピード タイムアウトを USB 2.0 仕様が規定している最大値よりも大きな値に設定できません (USB モジュールは仕様に非準拠となります)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-28: USBLPMR1: USB リンク電力管理制御レジスタ 1:

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	LPMERRIE	LPMRESIE	LPMACKIE	LPMNYIE	LPMSTIE	LPMTIOE
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0, HC	R/W-0, HC
	—	—	—	LPMNAK	LPMEN<1:0>		LPMRES	LPMXMT
15:8	R-0	R-0	R-0	R-0	U-0	U-0	U-0	R-0
	ENDPOINT<3:0>				—	—	—	RMTWAK
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	HIRD<3:0>				LNKSTATE<3:0>			

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-30 **未実装:** 「0」として読み出し

bit 29 **LPMERRIE:** USB エラー割り込みイネーブルビット

1 = LPMERR 割り込みを有効にする

0 = LPMERR 割り込みを無効にする

bit 28 **LPMRESIE:** LPM レジューム割り込みイネーブルビット

1 = LPMRES 割り込みを有効にする

0 = LPMRES 割り込みを無効にする

bit 27 **LPMACKIE:** LPM ACK 割り込みイネーブルビット

1 = LPMACK 割り込みを有効にする

0 = LPMACK 割り込みを無効にする

bit 26 **LPMNYIE:** LPM NYET 割り込みイネーブルビット

1 = LPMNYET 割り込みを有効にする

0 = LPMNYET 割り込みを無効にする

bit 25 **LPMSTIE:** LPM STALL 割り込みイネーブルビット

1 = LPMST 割り込みを有効にする

0 = LPMST 割り込みを無効にする

bit 24 **LPMTIOE:** LPM タイムアウト割り込みイネーブルビット

1 = LPMTIO 割り込みを有効にする

0 = LPMTIO 割り込みを無効にする

bit 23-21 **未実装:** 「0」として読み出し

bit 20 **LPMNAK:** LPM 専用トランザクション設定ビット

1 = 全てのエンドポイントは、LPM 以外の全てのトランザクションに NAK で応答する

0 = 通常のトランザクション動作

このビットを「1」にセットした効果は、USB モジュールが LPM サスペンドされた後に生じます。

bit 19-18 **LPMEN<1:0>:** LPM イネーブルビット (デバイスモード)

11 = LPM 拡張トランザクションをサポートする

10 = LPM 拡張トランザクションはサポートしない

01 = LPM モードをサポートしないが、拡張トランザクションはサポートする

00 = LPM 拡張トランザクションをサポートする

bit 17 **LPMRES:** LPM レジュームビット

1 = レジュームを開始する (リモート復帰)

レジューム信号は 50 μs 間アサートされます。

0 = レジューム動作はしない

このビットは自動的にクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-28: USBLPMR1: USB リンク電力管理制御レジスタ 1: ( 続き )

bit 16 **LPMXMT:** LPM L1 ステートへの遷移ビット

### デバイスモードの場合:

1 = 次の LPM トランザクションの受信時に USB モジュールは L1 ステートへ遷移する LPMEN を '0b11 に設定する必要があります。LPMXMT と LPMEN は同一サイクル内でセットする必要があります。

0 = 現在のステートを維持する

LPMXMT と LPMEN の両方をセットした場合、USB モジュールは以下の通りに応答できます。

- データが保留中ではない ( 全ての TX FIFO がエンプティ ) の場合、USB モジュールは ACK で応答します。このビットは自動的にクリアされ、ソフトウェア割り込みが生成されます。
- データが保留中 ( TX FIFO 内に少なくとも 1 つのデータが存在する ) 場合、USB モジュールは NYET で応答します。この場合、このビットは自動的にクリアされませんが、ソフトウェア割り込みは生成されます。

### ホストモードの場合:

1 = USB モジュールは LPM トランザクションを送信する

このビットは自動的にクリアされます。また、トークンを受信するか 3 回のタイムアウトが発生すると直ちにクリアされます。

0 = 現在のステートを維持する

bit 15-12 **ENDPOINT<3:0>:** LPM トークンパケット エンドポイント ビット

これは、LPM トランザクションのトークンパケットでのエンドポイントです。

bit 11-9 **未実装:** 「0」として読み出し

bit 8 **RMTWAK:** リモート復帰イネーブルビット

このビットは、現在サスペンド中のステートに対して一時的にだけ適用されます。

1 = リモート復帰を有効にする

0 = リモート復帰を無効にする

bit 7-4 **HIRD<3:0>:** ホストによるレジャーム期間ビット

ホストがバス上でレジャーム信号を駆動する最小時間です。このレジスタの値は、下式により実際のレジャーム期間に換算します。

$$\text{レジャーム期間} = 50 \mu\text{s} + \text{HIRD} \times 75 \mu\text{s}$$

レジャーム期間のレンジは 50 ~ 1200  $\mu\text{s}$  です。

bit 3-0 **LNKSTATE<3:0>:** リンク ステート ビット

この値はホストから周辺デバイスへ提供され、周辺デバイスが LPM トランザクションを受信および承認した後にどのステートに移行する必要があるか指定します。このレジスタの有効値は「1」( スリープステート (L1) ) のみです。その他の値は全て予約済みです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-29: USBLPMR2: USB リンク電力管理制御レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	LPMFADDR<6:0>							
7:0	U-0 —	U-0 —	R-0 LPMERRIF	R-0, HS LPMRESIF	R-0, HS LPMNCIF	R-0, HS LPMACKIF	R-0, HS LPMNYIF	R-0, HS LPMSTIF

<b>凡例:</b>	HS = ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-15 **未実装:** 「0」として読み出し

bit 14-8 **LPMFADDR<6:0>:** LPM ペイロード ファンクション アドレスビット  
これらのビットは LPM ペイロード ファンクションのアドレスを格納します。

bit 7-6 **未実装:** 「0」として読み出し

bit 5 **LPMERRIF:** LPM エラー割り込みフラグビット (**デバイスモード**)  
1 = 非サポートの LINKSTATE フィールドを含む LPM トランザクションを受信した  
応答は STALL です。  
0 = エラー条件は発生していない

bit 4 **LPMRESIF:** LPM レジューム割り込みフラグビット  
1 = USB モジュールがレジュームした (理由は問わず)  
0 = レジューム条件は発生していない

bit 3 **LPMNCIF:** LPM NC 割り込みフラグビット

**デバイスモードの場合:**

1 = USB モジュールは LPM トランザクションを受信したが、RX FIFO 内でデータが保留中であるため  
NYET を返した  
0 = NC 割り込み条件は発生していない

**ホストモードの場合:**

1 = LPM トランザクションを送信し、デバイスが ACK を返した  
0 = NC 割り込み条件は発生していない

bit 2 **LPMACKIF:** LPM ACK 割り込みフラグビット

**デバイスモードの場合:**

1 = LPM トランザクションを受信し、USB モジュールは ACK を返した  
0 = NC 割り込み条件は発生していない

**ホストモードの場合:**

1 = LPM トランザクションを送信し、デバイスが ACK を返した  
0 = NC 割り込み条件は発生していない

bit 1 **LPMNYIF:** LPM NYET 割り込みフラグビット

**デバイスモードの場合:**

1 = LPM トランザクションを受信し、USB モジュールは NYET を返した  
0 = NYET 割り込み条件は発生していない

**ホストモードの場合:**

1 = LPM トランザクションを送信し、デバイスが NYET を返した  
0 = NYET 割り込み条件は発生していない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-29: USBLPMR2: USB リンク電力管理制御レジスタ 2 ( 続き )

bit 0      LPMSTIF: LPM STALL 割り込みフラグビット

デバイスモードの場合:

1 = LPM トランザクションを受信し、USB モジュールは STALL を返した  
0 = ストール条件は発生していない

ホストモードの場合:

1 = LPM トランザクションを送信し、デバイスが STALL を返した  
0 = ストール条件は発生していない



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 11-30: USBCRCON: USB クロック / リセット制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R-0, HS, HC	R-0, HS, HC	R/W-1, HS
	—	—	—	—	—	USBIF	USBRF	USBWKUP
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	r-1	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	USB IDOVEN	USB IDVAL
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PHYIDEN	VBUS MONEN	ASVAL MONEN	BSVAL MONEN	SEND MONEN	USBIE	USBRIE	USB WKUPEN

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-27 **未実装**: 「0」として読み出し

bit 26 **USBIF**: USB 一般割り込みフラグビット

1 = USB バス上でイベントが発生した

0 = USB モジュールからの割り込みは発生していない、または割り込みが有効にされていない

bit 25 **USBRF**: USB レジューム フラグビット

1 = サスペンドステートからレジュームした

デバイス復帰動作を開始できます。

0 = サスペンド中にレジューム動作は検出されていない、またはサスペンドモード中ではない

bit 24 **USBWKUP**: USB アクティビティ ステータスビット

1 = このビットがクリアされた後に接続、切断、その他のアクティビティが USB で検出された

0 = USB でアクティビティは検出されていない

**Note:** このビットは、スリープに移行する直前にクリアする必要があります。しかし、実際にスリープに移行する前に、USB 上でアクティビティが何も発生していなかった事をこのビットで確認する必要があります。

bit 23-16 **未実装**: 「0」として読み出し

bit 15 **予約済み**: 「1」として読み出し

bit 14-10 **未実装**: 「0」として読み出し

bit 9 **USBIDOVEN**: USB オーバーライド イネーブルビット

1 = USBIDVAL ビットの使用を有効にする

0 = USBIDVAL ビットの使用を無効にし、代わりに PHY 値を使う

bit 8 **USBIDVAL**: USB ID 値ビット

1 = ID オーバーライド値は 1

0 = ID オーバーライド値は 0

bit 7 **PHYIDEN**: PHY ID 監視イネーブルビット

1 = USB PHY からの ID ビットの監視を有効にする

0 = USB PHY からの ID ビットの監視を無効にする

bit 6 **VBUSMONEN**: OTG の VBUS 監視イネーブルビット

1 = VBUS の監視 (VBUS 有効レンジ = 4.4 ~ 4.75 V) を有効にする

0 = VBUS の監視 (VBUS 有効レンジ) を無効にする

bit 5 **ASVALMONEN**: OTG の A デバイス VBUS 監視イネーブルビット

1 = VBUS の監視 (A デバイス向けセッション有効レンジ = 0.8 ~ 2.0 V) を有効にする

0 = VBUS の監視 (A デバイス向けセッション有効レンジ) を有効にする

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 11-30: USBCRCON: USB クロック / リセット制御レジスタ ( 続き )

- bit 4    **BSVALMONEN:** OTG の B デバイス VBUS 監視イネーブルビット  
1 = VBUS の監視 (B デバイス向けセッション有効レンジ = 0.8 ~ 4.0 V) を有効にする  
0 = VBUS の監視 (B デバイス向けセッション有効レンジ) を無効にする
- bit 3    **SENDMONEN:** OTG のセッション終了 VBUS 監視イネーブルビット  
1 = VBUS の監視 (セッション終了レンジ = 0.2 ~ 0.8 V) を有効にする  
0 = VBUS の監視 (セッション終了レンジ) を無効にする
- bit 2    **USBIE:** USB 一般割り込みイネーブルビット  
1 = USB モジュールからの一般割り込みを有効にする  
0 = USB モジュールからの一般割り込みを無効にする
- bit 1    **USBRIE:** USB レジューム割り込みイネーブルビット  
1 = サスペンドからのリモート レジューム割り込みを有効にする  
0 = リモートデバイス USB レジューム信号に対する割り込みを無効にする
- bit 0    **USBWKUPEN:** USB アクティビティ検出割り込みイネーブルビット  
1 = スリープ中の USB バス上アクティビティ検出割り込みを有効にする  
0 = スリープ中の USB バス上アクティビティ検出割り込みを無効にする

# PIC32MZ グラフィック (DA) ファミリ

## 12.0 I/O ポート

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 12. I/O Ports』(DS60001120) を参照してください。この文書は、Microchip 社ウェブサイトの PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

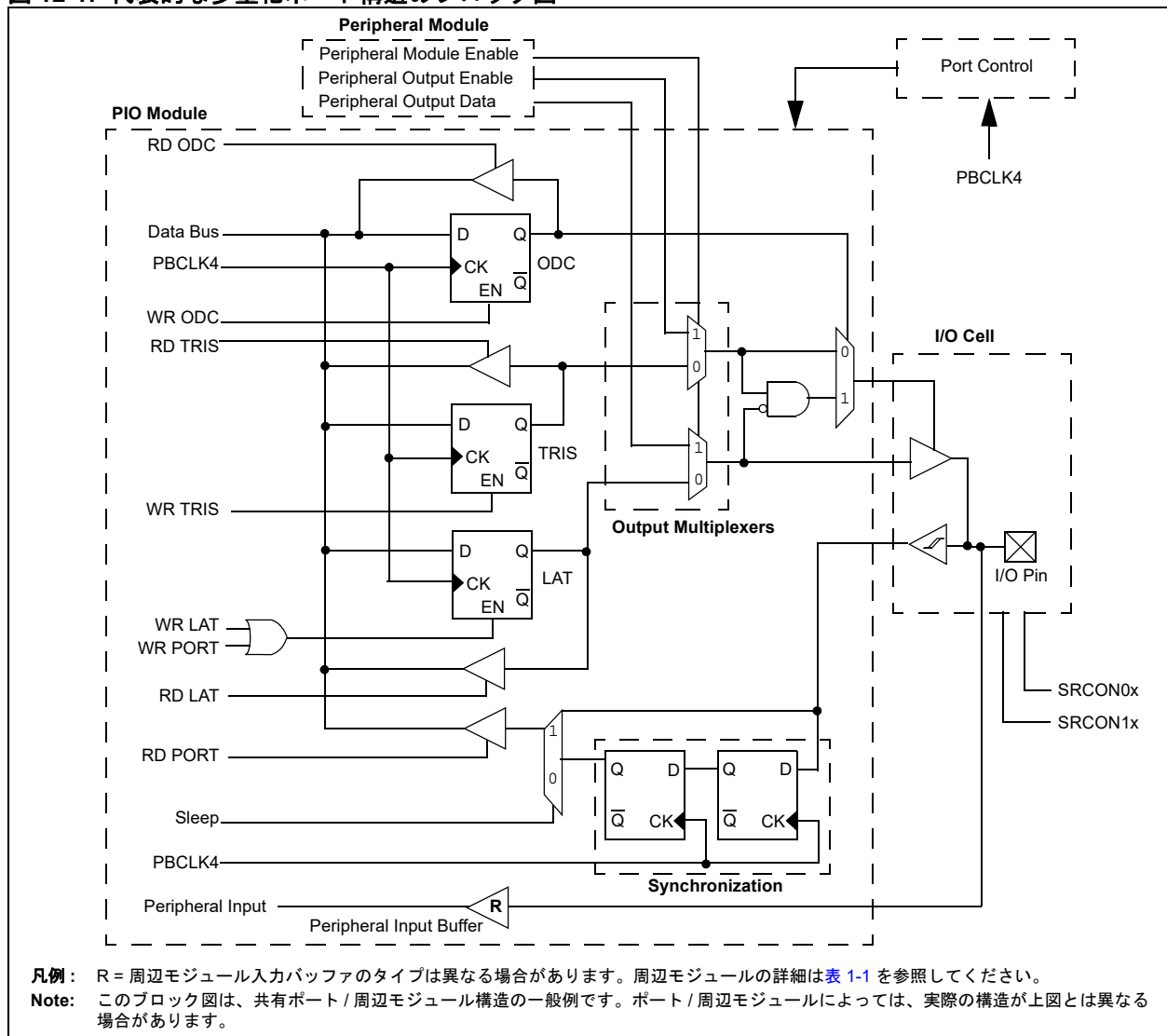
汎用 I/O ピンは最も単純な周辺モジュールです。PIC32MZ DA ファミリのデバイスは、これらのピンを使って他のデバイスを監視および制御します。デバイスの柔軟性と機能性を高めるため、一部のピンには複数の機能が多重化されています。これらのピンの機能は、デバイスが実際に内蔵している周辺機能に応じて

異なります。通常、周辺モジュールが機能している時は、対応するピンを汎用 I/O ピンとして使う事はできません。

I/O ポート モジュールの主な特長は以下の通りです。

- 出力ピンごとのオープンドレイン イネーブル / ディセーブル
  - 入力ピンごとの弱いプルアップ / プルダウン
  - 選択した入力を監視し、ピンの状態変化を検出した時に割り込みを生成
  - スリープおよびアイドル中の動作
  - CLR、SET、INV レジスタを使った高速なビット操作
- 図 12-1 に代表的な多重化 I/O ポートのブロック図を示します。

図 12-1: 代表的な多重化ポート構造のブロック図



# PIC32MZ グラフィック (DA) ファミリ

## 12.1 パラレル I/O (PIO) ポート

全てのポートピンは、デジタル I/O としての動作に直接関連する 10 個のレジスタを備えています。データ方向レジスタ (TRISx) は、そのピンの動作方向 (入力か出力か) を指定します。データ方向ビットが「1」の場合、そのピンは入力です。全てのポートピンはリセット後に入力ピンとして設定されます。LATx レジスタを読み出すと、ラッチが読み出されます。LATx レジスタに書き込むと、ラッチが書き込まれます。PORTx レジスタを読み出すとポートピンが読み出されますが、ポートピンに書き込むとラッチに書き込まれます。

### 12.1.1 オープンドレイン構成

PORTx、LATx、TRISx レジスタによるデータ制御に加えて、一部のポートピンは個別にデジタル出力またはオープンドレイン出力用に設定する事もできます。この設定には、各ポートに割り当てられたオープンドレイン制御レジスタ (ODCx) を使います。このレジスタ内のビットをセットすると、対応するピンはオープンドレイン出力として動作します。

オープンドレイン機能により、外付けプルアップ抵抗を使って任意の 5V 対応ピンで VDDIO (例: 5V) よりも高い出力を生成できます。オープンドレインの最大許容電圧は、V<sub>IH</sub> の最大仕様値と同じです。

利用可能なピンと、それらの機能については、ピン一覧表 (表 5 と表 7) を参照してください。

### 12.1.2 アナログおよびデジタル ポートピンの設定

ANSELx レジスタはアナログポートピンの動作を制御します。ポートピンをアナログ入力として動作させる場合、そのピンに対応する ANSEL ビットと TRIS ビットをセットする必要があります。ポートピンをデジタルモジュール (タイマ、UART 等) との I/O 機能用に使う場合、対応する ANSELx ビットをクリアする必要があります。

ANSELx レジスタの既定値は 0xFFFF です。従って、アナログ機能を共有する全てのピンの既定値はアナログです (デジタルではありません)。

ANSELx ビットをセットした状態で TRIS ビットをクリア (出力として設定) すると、アナログ周辺モジュール (ADC、コンパレータ等) はデジタル出力レベル (V<sub>OH</sub> または V<sub>OL</sub>) を変換します。

アナログ入力チャンネルとして設定された全てのピンは、PORT レジスタを読み出すとクリア状態 (LOW レベル) として読み出されます。

デジタル入力として設定されたピンはアナログ入力を変換しません。デジタル入力として定義されたピン (ANx ピンを含む) にアナログレベルの電圧がかかると、入力バッファがデバイスの仕様を超える電流を消費する可能性があります。

### 12.1.3 I/O ポートの読み書きタイミング

ポートの方向を変更した直後またはポートに書き込んだ直後に同じポートを読み出す場合、間に 1 命令サイクルを挿入する必要があります。これには通常 NOP 命令を使います。

### 12.1.4 入力状態変化通知

I/O ポートの状態変化通知 (CN) 機能を使う事で、PIC32MZ DA は選択した入力ピンの状態変化に応じてプロセッサへの割り込み要求を生成できます。この機能を使うと、クロックが停止するスリープ中でも入力の状態変化を検出できます。全ての I/O ポートピンは、状態変化時に割り込み要求を生成するよう設定できます。

各 I/O ポートの CN 機能には 5 つの制御レジスタが割り当てられています。CNENx/CNNEx レジスタは、各入力ピンの CN 割り込みイネーブル制御ビットを格納しています。これらのビットをセットすると、対応するピンの CN 割り込みが有効になります。EDGEDETECT ビット (CNCONx<11>) がセットされていない場合、CNENx は不一致 CN 割り込み条件を有効にします。EDGEDETECT ビットがセットされている場合、CNNEx は立ち下がりエッジを制御し、CNENx は立ち上がりエッジを制御します。

CNSTATx/CNFx レジスタは、EDGEDETECT ビットの設定に基づいて状態変化のステータスを示します。EDGEDETECT ビットが「0」の場合、CNSTATx レジスタは、直近の PORTx ビットの読み出し後にピンの状態が変化したかどうかを示します。EDGEDETECT ビットが「1」の場合、CNFx レジスタは状態変化が発生したかどうかを示し、CNNEx/CNNEx レジスタは発生した変化のエッジタイプを示します。

各 I/O ピンには弱いプルアップおよび弱いプルダウン抵抗が接続されています。プルアップ抵抗は、ピンに接続した電流ソースまたはシンクとして機能し、押しボタンまたはキーパッド デバイスを接続する場合に外付け抵抗は不要です。プルアップおよびプルダウンは、それぞれ CNPUx および CNPDx レジスタ内の各ピンに対応する制御ビットを使って別々に有効にできます。これらの制御ビットをセットすると、対応するピンのプルアップおよび / またはプルダウンが有効になります。

**Note:** ポートピンをデジタル出力として設定する場合、状態変化通知ピンのプルアップおよびプルダウンは常に無効にする必要があります。

状態変化通知制御 (CNCONx) レジスタは [レジスタ 12-3](#) に記載しています。

## 12.2 CLR、SET、INV レジスタ

各 I/O モジュール レジスタには、高速なビット操作を可能にする CLR (クリア)、SET (セット)、INV (反転) レジスタが割り当てられています。SET、CLR、INV レジスタに値を書き込む事によって特定のビットを「1」にセットすると、対応するベースレジスタ内の対応するビットだけがセット (SET)、クリア (CLR)、反転 (INV) されます。これらのレジスタ内で「0」に設定されているビットに対応するベースレジスタ内のビットは変更されません。

SET、CLR、INV レジスタの読み値は未確定です。SET、CLR、INV レジスタに書き込んだ結果を確認するには、ベースレジスタを読み出す必要があります。

## 12.3 スルーレート レジスタ

各 I/O ピンは、割り当てられたポート上で、各種のスルーレート制御向けに設定できます。これには、各 I/O ポートに対応する SRCON1x および SRCON0x レジスタ内のスルーレート制御ビットを使います。

## 12.4 ペリフェラル ピンセレクト (PPS)

汎用デバイスの大きな課題は、I/O ピンでの機能の競合を最小限に抑えつつ、できるだけ多くの周辺機能を提供する事です。これは特に少ピンデバイスで困難です。1本のピンに複数の周辺モジュールを割り当てる必要のあるアプリケーションでは、コードによって面倒な回避策を講じるか、完全に設計し直すしか方法がない場合があります。

PPS 機能は、周辺モジュールの選択と I/O ピンへの割り当てを柔軟に行えるようにする事で、この課題を克服します。ピン割り当ての自由度が増す事で、デバイスに合わせてアプリケーションの機能を削るのではなく、デバイスをアプリケーションに合わせる事ができます。

PPS 機能は、デジタル I/O ピンの一部で利用できます。これらの I/O ピンには、デジタル周辺モジュール入出力の大部分を割り当てる事ができます。PPS 機能はソフトウェアで実行し、通常デバイスの再プログラミングは不要です。周辺モジュールの割り当てが偶発的または誤って変更されないようにするため、ハードウェアによる安全措置が組み込まれています。

### 12.4.1 使用可能なピン

使用可能なピンの数は、デバイスとピン数に応じて異なります。PPS 機能をサポートするピンの名前には「RPn」が付きます。「RP」は割り当て変更可能周辺モジュール (Remappable Peripheral) を意味し、「n」はポート番号を表します。

### 12.4.2 使用可能な周辺モジュール

PPS は、デジタル専用の周辺モジュールだけをサポートします。これには汎用シリアル通信 (UART、SPI、CAN)、汎用タイマクロック入力、タイマ関連周辺モジュール (入力キャプチャと出力コンペア)、状態変化時割り込み入力、参照クロック (入力と出力) が含まれます。

デジタル専用周辺モジュールの中には、PPS 機能から除外される物もあります。それらの周辺モジュールは特定のポート上で特殊な I/O 回路を必要とするため、異なるピンに接続するのは困難です。このようなモジュールには I<sup>2</sup>C が含まれます。また同様の理由により、アナログ入力を持つ全てのモジュール (アナログ / デジタル コンバータ (ADC) 等) も除外されます。

割り当て変更が可能な周辺モジュールとそうでない周辺モジュールの主な違いとして、前者には既定値の I/O ピンが割り当てられていません。割り当て変更可能な周辺モジュールを使うには、特定の I/O ピンをあらかじめ割り当てておく必要があります。これに対し、割り当て変更ができない周辺モジュールは常に既定値のピンを介して使えます。ただし、その周辺モジュールがアクティブであり、他の周辺モジュールと競合していない事が必要です。

ある I/O ピンで割り当て変更可能周辺モジュールがアクティブである場合、その周辺モジュールは、そのピンに関連付けられているどのデジタル I/O およびデジタル通信周辺モジュールよりも高い優先度を持ちます。この優先度は、割り当てられた周辺モジュールのタイプには関係なく与えられます。割り当て変更可能周辺モジュールは、そのピンに関連付けられているアナログ機能より高い優先度を持つ事はありません。

### 12.4.3 PPS の制御

PPS 機能は2セットの特殊機能レジスタ (SFR) で制御します。1つは周辺モジュールの入力を割り当てるために使い、もう1つは出力を割り当てるために使います。これらは別々に制御されるため、周辺モジュールに入力と出力の両方がある場合、モジュールの入力と出力は別々に任意のピンに割り当てる事ができます。周辺モジュールを PPS ピンに割り当てる方法は入力と出力で異なります。

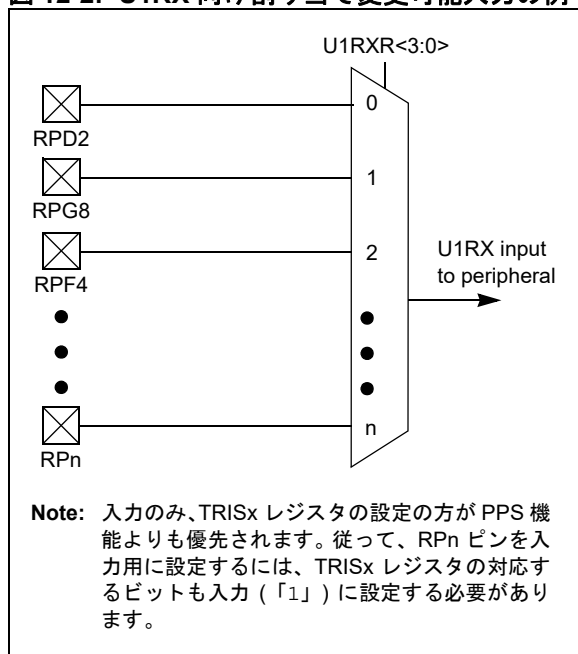
# PIC32MZ グラフィック (DA) ファミリ

## 12.4.4 入力の割り当て

PPS オプションの入力は、周辺モジュール側から割り当てます。すなわち、各周辺モジュールに対応する制御レジスタを使って、割り当て先のピンを指定します。周辺モジュール入力の割り当てには、 $[pin\ name]R$  という名前のレジスタを使います ( $[pin\ name]$  は表 12-1 内のペリフェラルピン名) (レジスタ 12-1 参照)。各レジスタは一連の 4 ビットフィールドを格納します。これらのビットフィールドに適切な値をプログラミングする事により、対応する  $RPn$  ピンをその周辺モジュールに割り当てます。表 12-1 は、どのデバイスのどのビットフィールドに対しても有効な値のレンジを示しています。

例として、図 12-2 に、U1RX 入力向けの割り当て変更可能ピンを示します。

図 12-2: U1RX 向け割り当て変更可能入力の例



# PIC32MZ グラフィック (DA) ファミリ

表 12-1: 入力ピンの選択

ペリフェラルピン	[pin name]R SFR	[pin name]R ビット	RPn ピンを選択する [pin name]R 値
INT3	INT3R	INT3R<3:0>	0000 = RPD2
T2CK	T2CKR	T2CKR<3:0>	0001 = RPG8
T6CK	T6CKR	T6CKR<3:0>	0010 = RPF4
IC3	IC3R	IC3R<3:0>	0011 = 予約済み
IC7	IC7R	IC7R<3:0>	0100 = RPF1
U1RX	U1RXR	U1RXR<3:0>	0101 = RPB9
$\overline{U2CTS}$	U2CTSR	U2CTSR<3:0>	0110 = RPB10
U5RX	U5RXR	U5RXR<3:0>	0111 = RPC14
$\overline{U6CTS}$	U6CTSR	U6CTSR<3:0>	1000 = RPB5
SDI1	SDI1R	SDI1R<3:0>	1001 = 予約済み
SDI3	SDI3R	SDI3R<3:0>	1010 = RPC1
SDI5	SDI5R	SDI5R<3:0>	1011 = RPD14
SS6	SS6R	SS6R<3:0>	1100 = RPG1
REFCLKI1	REFCLKI1R	REFCLKI1R<3:0>	1101 = RPA14
			1110 = RPD6
			1111 = 予約済み
INT4	INT4R	INT4R<3:0>	0000 = RPD3
T5CK	T5CKR	T5CKR<3:0>	0001 = RPG7
T7CK	T7CKR	T7CKR<3:0>	0010 = RPF5
IC4	IC4R	IC4R<3:0>	0011 = RPD11
IC8	IC8R	IC8R<3:0>	0100 = RPF0
U3RX	U3RXR	U3RXR<3:0>	0101 = RPB1
$\overline{U4CTS}$	U4CTSR	U4CTSR<3:0>	0110 = RPE5
SDI2	SDI2R	SDI2R<3:0>	0111 = RPC13
SDI4	SDI4R	SDI4R<3:0>	1000 = RPB3
C1RX	C1RXR	C1RXR<3:0>	1001 = 予約済み
REFCLKI4	REFCLKI4R	REFCLKI4R<3:0>	1010 = RPC4
			1011 = 予約済み
			1100 = RPG0
			1101 = RPA15
			1110 = RPD7
			1111 = 予約済み

# PIC32MZ グラフィック (DA) ファミリ

表 12-1: 入力ピンの選択 ( 続き )

ペリフェラルピン	[pin name]R SFR	[pin name]R ビット	RPn ピンを選択する [pin name]R 値
INT2	INT2R	INT2R<3:0>	0000 = RPD9
T3CK	T3CKR	T3CKR<3:0>	0001 = 予約済み
T8CK	T8CKR	T8CKR<3:0>	0010 = RPB8
IC2	IC2R	IC2R<3:0>	0011 = RPB15
IC5	IC5R	IC5R<3:0>	0100 = RPD4
IC9	IC9R	IC9R<3:0>	0101 = RPB0
$\overline{U1CTS}$	U1CTSR	U1CTSR<3:0>	0110 = RPE3
U2RX	U2RXR	U2RXR<3:0>	0111 = RPB7
$\overline{U5CTS}$	U5CTSR	U5CTSR<3:0>	1000 = 予約済み
$\overline{SS1}$	SS1R	SS1R<3:0>	1001 = RPF12
$\overline{SS3}$	SS3R	SS3R<3:0>	1010 = RPD12
$\overline{SS4}$	SS4R	SS4R<3:0>	1011 = RPF8
$\overline{SS5}$	SS5R	SS5R<3:0>	1100 = RPC3
C2RX	C2RXR	C2RXR<3:0>	1101 = RPE9
INT1	INT1R	INT1R<3:0>	1110 = 予約済み
T4CK	T4CKR	T4CKR<3:0>	1111 = 予約済み
T9CK	T9CKR	T9CKR<3:0>	0000 = 予約済み
IC1	IC1R	IC1R<3:0>	0001 = RPG9
IC6	IC6R	IC6R<3:0>	0010 = 予約済み
$\overline{U3CTS}$	U3CTSR	U3CTSR<3:0>	0011 = RPD0
U4RX	U4RXR	U4RXR<3:0>	0100 = 予約済み
U6RX	U6RXR	U6RXR<3:0>	0101 = RPB6
$\overline{SS2}$	SS2R	SS2R<3:0>	0110 = RPD5
SDI6	SDI6R	SDI6R<3:0>	0111 = RPB2
OCFA	OCFAR	OCFAR<3:0>	1000 = RPF3
REFCLKI3	REFCLKI3R	REFCLKI3R<3:0>	1001 = 予約済み
			1010 = 予約済み
			1011 = RPF2
			1100 = RPC2
			1101 = RPE8
			1110 = 予約済み
			1111 = 予約済み



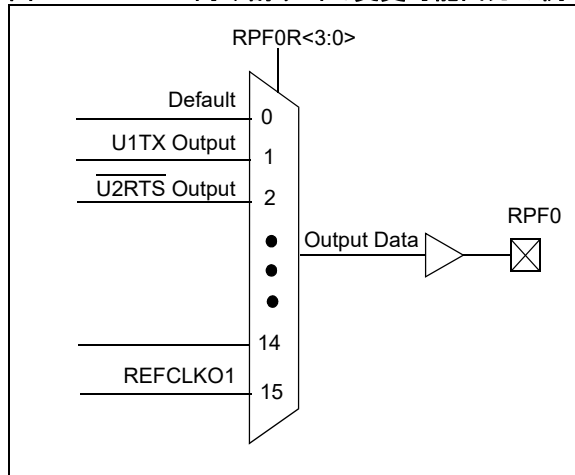
# PIC32MZ グラフィック (DA) ファミリ

## 12.4.5 出力の割り当て

入力とは異なり、PPS オプションの出力はピン側から割り当てます。すなわち、各ピンに対応する制御レジスタを使って、そのピンに割り当てする周辺モジュール出力を指定します。出力の割り当てには RPnR レジスタ (レジスタ 12-2) を使います。[pin name]R レジスタと同様に、各レジスタは一連の 4 ビットフィールドを格納します。ビットフィールドの値に対応する周辺モジュールの出力がそのピンに割り当てられます (表 12-2 と図 12-3 参照)。

出力レジスタのリセット値「0」には NULL 出力が対応します。このため、既定値状態では、割り当て変更可能出力は全ての出力ピンから切り離されたままです。

図 12-3: RPF0 向け割り当て変更可能出力の例



## 12.4.6 コンフィグレーション変更の制御

周辺モジュールの割り当て変更は実行時も可能であるため、偶発的なコンフィグレーションの変更を防ぐための制約が必要です。PIC32MZ DA は、周辺機モジュール割り当ての変更を防ぐために以下の 2 つの機能を備えています。

- 制御レジスタのロックシーケンス
- コンフィグレーション ビット選択のロック

### 12.4.6.1 制御レジスタのロック

通常動作時は、RPnR および [pin name]R レジスタへの書き込みが禁止されます。書き込みを試みると、正常に実行されたかのように見えますが、実際にはレジスタの内容は変更されません。これらのレジスタを変更するには、ハードウェアでレジスタのロックを解除する必要があります。レジスタのロックは IOLOCK コンフィグレーション ビット (CFGCON<13>) で制御します。IOLOCK ビットをセットすると制御レジスタへの書き込みが禁止され、クリアすると書き込みが許可されます。

IOLOCK ビットをセットまたはクリアするには、ロック解除シーケンスを実行する必要があります。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

### 12.4.6.2 コンフィグレーション ビット選択のロック

さらに安全性を高めるため、RPnR および [pin name]R レジスタに対して複数回の書き込みを防ぐようにデバイスを設定できます。IOL1WAY コンフィグレーション ビット (DEVCFG3<29>) は、IOLOCK ビットが一度セットされた後でクリアされる事を防ぎます。IOLOCK ビットがセットされたままの場合、レジスタロック解除シーケンスは実行されず、PPS 制御レジスタに書き込む事はできません。このビットをクリアして周辺モジュールの割り当て変更を再び可能にするには、デバイスリセットを実行する以外に方法はありません。

既定値 (未プログラム) 状態では IOL1WAY はセットされており、ユーザは 1 回の書き込みしかできません。

# PIC32MZ グラフィック (DA) ファミリ

表 12-2: 出力ピンの選択

RPn ポートピン	RPnR SFR	RPnR ビット	周辺モジュールを選択する RPnR 値
RPD2	RPD2R	RPD2R<3:0>	0000 = 未接続
RPG8	RPG8R	RPG8R<3:0>	0001 = U3TX
RPF4	RPF4R	RPF4R<3:0>	0010 = U4RTS
RPF1	RPF1R	RPF1R<3:0>	0011 = 予約済み
RPB9	RPB9R	RPB9R<3:0>	0100 = 予約済み
RPB10	RPB10R	RPB10R<3:0>	0101 = SDO1
RPB5	RPB5R	RPB5R<3:0>	0110 = SDO2
RPC1	RPC1R	RPC1R<3:0>	0111 = SDO3
RPD14	RPD14R	RPD14R<3:0>	1000 = 予約済み
RPG1	RPG1R	RPG1R<3:0>	1001 = SDO5
RPA14	RPA14R	RPA14R<3:0>	1010 = SS6
RPD6	RPD6R	RPD6R<3:0>	1011 = OC3
RPD3	RPD3R	RPD3R<3:0>	1100 = OC6
RPG7	RPG7R	RPG7R<3:0>	1101 = REFCLKO4
RPF5	RPF5R	RPF5R<3:0>	1110 = C2OUT
RPD11	RPD11R	RPD11R<3:0>	1111 = C1TX
RPE5	RPE5R	RPE5R<3:0>	0000 = 未接続
RPB3	RPB3R	RPB3R<3:0>	0001 = U1TX
RPC4	RPC4R	RPC4R<3:0>	0010 = U2RTS
RPG0	RPG0R	RPG0R<3:0>	0011 = U5TX
RPA15	RPA15R	RPA15R<3:0>	0100 = U6RTS
RPD7	RPD7R	RPD7R<3:0>	0101 = SDO1
RPF0	RPF0R	RPF0R<3:0>	0110 = SDO2
RPB1	RPB1R	RPB1R<3:0>	0111 = SDO3
RPE5	RPE5R	RPE5R<3:0>	1000 = SDO4
RPC4	RPC4R	RPC4R<3:0>	1001 = SDO5
RPG0	RPG0R	RPG0R<3:0>	1010 = 予約済み
RPA15	RPA15R	RPA15R<3:0>	1011 = OC4
RPD7	RPD7R	RPD7R<3:0>	1100 = OC7
RPF0	RPF0R	RPF0R<3:0>	1101 = 予約済み
RPB1	RPB1R	RPB1R<3:0>	1110 = 予約済み
RPE5	RPE5R	RPE5R<3:0>	1111 = REFCLKO1

# PIC32MZ グラフィック (DA) ファミリ

表 12-2: 出力ピンの選択 ( 続き )

RPn ポートピン	RPnR SFR	RPnR ビット	周辺モジュールを選択する RPnR 値
RPD9	RPD9R	RPD9R<3:0>	0000 = 未接続
RPB8	RPB8R	RPB8R<3:0>	0001 = U3RTS
RPB15	RPB15R	RPB15R<3:0>	0010 = U4TX
RPD4	RPD4R	RPD4R<3:0>	0011 = 予約済み
RPB0	RPB0R	RPB0R<3:0>	0100 = U6TX
RPE3	RPE3R	RPE3R<3:0>	0101 = SS1
RPB7	RPB7R	RPB7R<3:0>	0110 = 予約済み
RPF12	RPF12R	RPF12R<3:0>	0111 = SS3
RPD12	RPD12R	RPD12R<3:0>	1000 = SS4
RPF8	RPF8R	RPF8R<3:0>	1001 = SS5
RPC3	RPC3R	RPC3R<3:0>	1010 = SDO6
RPE9	RPE9R	RPE9R<3:0>	1011 = OC5
RPG9	RPG9R	RPG9R<3:0>	1100 = OC8
RPD0	RPD0R	RPD0R<3:0>	1101 = 予約済み
RPB6	RPB6R	RPB6R<3:0>	1110 = C1OUT
RPD5	RPD5R	RPD5R<3:0>	1111 = REFCLKO3
RPB2	RPB2R	RPB2R<3:0>	0000 = 未接続
RPF3	RPF3R	RPF3R<3:0>	0001 = U1RTS
RPC2	RPC2R	RPC2R<3:0>	0010 = U2TX
RPE8	RPE8R	RPE8R<3:0>	0011 = U5RTS
RPF2	RPF2R	RPF2R<3:0>	0100 = U6TX
RPD5	RPD5R	RPD5R<3:0>	0101 = 予約済み
RPB2	RPB2R	RPB2R<3:0>	0110 = SS2
RPF3	RPF3R	RPF3R<3:0>	0111 = 予約済み
RPC2	RPC2R	RPC2R<3:0>	1000 = SDO4
RPE8	RPE8R	RPE8R<3:0>	1001 = 予約済み
RPF2	RPF2R	RPF2R<3:0>	1010 = SDO6
RPD5	RPD5R	RPD5R<3:0>	1011 = OC2
RPB2	RPB2R	RPB2R<3:0>	1100 = OC1
RPF2	RPF2R	RPF2R<3:0>	1101 = OC9
RPD5	RPD5R	RPD5R<3:0>	1110 = 予約済み
RPF2	RPF2R	RPF2R<3:0>	1111 = C2TX

## 12.5 I/O ポート制御レジスタ

表 12-3: PORTA のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0000	ANSELA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	ANSA10	ANSA9	—	—	—	—	ANSA5	—	—	—	ANSA1	—
0010	TRISA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TRISA15	TRISA14	—	—	—	TRISA10	TRISA9	—	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	C6FF
0020	PORTA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RA15	RA14	—	—	—	RA10	RA9	—	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx
0030	LATA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	LATA15	LATA14	—	—	—	LATA10	LATA9	—	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx
0040	ODCA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ODCA15	ODCA14	—	—	—	ODCA10	ODCA9	—	ODCA7	ODCA6	ODCA5	ODCA4	ODCA3	ODCA2	ODCA1	ODCA0	0000
0050	CNPUA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPUA15	CNPUA14	—	—	—	CNPUA10	CNPUA9	—	CNPUA7	CNPUA6	CNPUA5	CNPUA4	CNPUA3	CNPUA2	CNPUA1	CNPUA0	0000
0060	CNPDA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPDA15	CNPDA14	—	—	—	CNPDA10	CNPDA9	—	CNPDA7	CNPDA6	CNPDA5	CNPDA4	CNPDA3	CNPDA2	CNPDA1	CNPDA0	0000
0070	CNCONA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	—	0000
0080	CNENA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNIEA15	CNIEA14	—	—	—	CNIEA10	CNIEA9	—	CNIEA7	CNIEA6	CNIEA5	CNIEA4	CNIEA3	CNIEA2	CNIEA1	CNIEA0	0000
0090	CNSTATA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CN STATA15	CN STATA14	—	—	—	CN STATA10	CN STATA9	—	CN STATA7	CN STATA6	CN STATA5	CN STATA4	CN STATA3	CN STATA2	CN STATA1	CN STATA0	0000
00A0	CNNEA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNNEA15	CNNEA14	—	—	—	CNNEA10	CNNEA9	—	CNNEA7	CNNEA6	CNNEA5	CNNEA4	CNNEA3	CNNEA2	CNNEA1	CNNEA0	0000
00B0	CNFA	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNFA15	CNFA14	—	—	—	CNFA10	CNFA9	—	CNFA7	CNFA76	CNFA5	CNFA4	CNFA3	CNFA2	CNFA1	CNFA0	0000
00C0	SRCON0A	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR1A15	SR1A14	—	—	—	SR1A10	SR1A9	—	SR1A7	SR1A6	SR1A5	SR1A4	SR1A3	SR1A2	SR1A1	SR1A0	0000
00D0	SRCON1A	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR0A15	SR0A14	—	—	—	SR0A10	SR0A9	—	SR0A7	SR0A6	SR0A5	SR0A4	SR0A3	SR0A2	SR0A1	SR0A0	0000

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)

Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

表 12-4: PORTB のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0100	ANSELB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ANSB15	ANSB14	ANSB13	ANSB12	ANSB11	ANSB10	ANSB9	ANSB8	ANSB7	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	FFBF
0110	TRISB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	FFFF
0120	PORTB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx
0130	LATB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx
0140	ODCB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ODCB15	ODCB14	ODCB13	ODCB12	ODCB11	ODCB10	ODCB9	ODCB8	ODCB7	ODCB6	ODCB5	ODCB4	ODCB3	ODCB2	ODCB1	ODCB0	0000
0150	CNPUB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPUB15	CNPUB14	CNPUB13	CNPUB12	CNPUB11	CNPUB10	CNPUB9	CNPUB8	CNPUB7	CNPUB6	CNPUB5	CNPUB4	CNPUB3	CNPUB2	CNPUB1	CNPUB0	0000
0160	CNPDB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPDB15	CNPDB14	CNPDB13	CNPDB12	CNPDB11	CNPDB10	CNPDB9	CNPDB8	CNPDB7	CNPDB6	CNPDB5	CNPDB4	CNPDB3	CNPDB2	CNPDB1	CNPDB0	0000
0170	CNCONB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	—	0000
0180	CNENB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNIEB15	CNIEB14	CNIEB13	CNIEB12	CNIEB11	CNIEB10	CNIEB9	CNIEB8	CNIEB7	CNIEB6	CNIEB5	CNIEB4	CNIEB3	CNIEB2	CNIEB1	CNIEB0	0000
0190	CNSTATB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CN STATB15	CN STATB14	CN STATB13	CN STATB12	CN STATB11	CN STATB10	CN STATB9	CN STATB8	CN STATB7	CN STATB6	CN STATB5	CN STATB4	CN STATB3	CN STATB2	CN STATB1	CN STATB0	0000
01A0	CNNEB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNNEB15	CNNEB14	CNNEB13	CNNEB12	CNNEB11	CNNEB10	CNNEB9	CNNEB8	CNNEB7	CNNEB6	CNNEB5	CNNEB4	CNNEB3	CNNEB2	CNNEB1	CNNEB0	0000
01B0	CNFB	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNFB15	CNFB14	CNFB13	CNFB12	CNFB11	CNFB10	CNFB9	CNFB8	CNFB7	CNFB76	CNFB5	CNFB4	CNFB3	CNFB2	CNFB1	CNFB0	0000
01C0	SRCON0B	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR1B15	SR1B14	SR1B13	SR1B12	SR1B11	SR1B10	SR1B9	SR1B8	SR1B7	SR1B6	SR1B5	SR1B4	SR1B3	SR1B2	SR1B1	SR1B0	0000
01D0	SRCON1B	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR0B15	SR0B14	SR0B13	SR0B12	SR0B11	SR0B10	SR0B9	SR0B8	SR0B7	SR0B6	SR0B5	SR0B4	SR0B3	SR0B2	SR0B1	SR0B0	0000

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)

Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 12-5: PORTC のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0200	ANSELC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	ANSC4	ANSC3	ANSC2	ANSC1	—
0210	TRISC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TRISC15	—	—	TRISC12	—	—	—	—	—	—	—	—	TRISC4	TRISC3	TRISC2	TRISC1	—
0220	PORTC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RC15	RC14	RC13	RC12	—	—	—	—	—	—	—	—	RC4	RC3	RC2	RC1	—
0230	LATC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	LATC15	LATC14	LATC13	LATC12	—	—	—	—	—	—	—	—	LATC4	LATC3	LATC2	LATC1	—
0240	ODCC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ODCC15	ODCC14	ODCC13	ODCC12	—	—	—	—	—	—	—	—	ODCC4	ODCC3	ODCC2	ODCC1	—
0250	CNPUC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPUC15	CNPUC14	CNPUC13	CNPUC12	—	—	—	—	—	—	—	—	CNPUC4	CNPUC3	CNPUC2	CNPUC1	—
0260	CNPDC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPDC15	CNPDC14	CNPDC13	CNPDC12	—	—	—	—	—	—	—	—	CNPDC4	CNPDC3	CNPDC2	CNPDC1	—
0270	CNCONC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	—	—
0280	CNENC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNIEC15	CNIEC14	CNIEC13	CNIEC12	—	—	—	—	—	—	—	—	CNIEC4	CNIEC3	CNIEC2	CNIEC1	—
0290	CNSTATC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNSTATC15	CNSTATC14	CNSTATC13	CNSTATC12	—	—	—	—	—	—	—	—	CNSTATC4	CNSTATC3	CNSTATC2	CNSTATC1	—
02A0	CNNEC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNNEC15	CNNEC14	CNNEC13	CNNEC12	—	—	—	—	—	—	—	—	CNNEC4	CNNEC3	CNNEC2	CNNEC1	—
02B0	CNFC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNFC15	CNFC14	CNFC13	CNFC12	—	—	—	—	—	—	—	—	CNFC4	CNFC3	CNFC2	CNFC1	—
02C0	SRCON0C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR1C15	SR1C14	SR1C13	SR1C12	—	—	—	—	—	—	—	—	SR1C4	SR1C3	SR1C2	SR1C1	—
02D0	SRCON1C	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR0C15	SR0C14	SR0C13	SR0C12	—	—	—	—	—	—	—	—	SR0C4	SR0C3	SR0C2	SR0C1	—

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)  
 Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

表 12-6: PORTD のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 (1)	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0300	ANSELD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ANSD15	ANSD14	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0310	TRISD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TRISD15	TRISD14	TRISD13	TRISD12	TRISD11	TRISD10	TRISD9	—	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	FEFF
0320	PORTD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RD15	RD14	RD13	RD12	RD11	RD10	RD9	—	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx
0330	LATD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	LATD15	LATD14	LATD13	LATD12	LATD11	LATD10	LATD9	—	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0	xxxx
0340	ODCD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ODCD15	ODCD14	ODCD13	ODCD12	ODCD11	ODCD10	ODCD9	—	ODCD7	ODCD6	ODCD5	ODCD4	ODCD3	ODCD2	ODCD1	ODCD0	0000
0350	CNPUD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPUD15	CNPUD14	CNPUD13	CNPUD12	CNPUD11	CNPUD10	CNPUD9	—	CNPUD7	CNPUD6	CNPUD5	CNPUD4	CNPUD3	CNPUD2	CNPUD1	CNPUD0	0000
0360	CNPDD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPDD15	CNPDD14	CNPDD13	CNPDD12	CNPDD11	CNPDD10	CNPDD9	—	CNPDD7	CNPDD6	CNPDD5	CNPDD4	CNPDD3	CNPDD2	CNPDD1	CNPDD0	0000
0370	CNCOND	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	—	0000
0380	CNEND	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNIED15	CNIED14	CNIED13	CNIED12	CNIED11	CNIED10	CNIED9	—	CNIED7	CNIED6	CNIED5	CNIED4	CNIED3	CNIED2	CNIED1	CNIED0	0000
0390	CNSTATD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNS TATD15	CN STATD14	CN STATD13	CN STATD12	CN STATD11	CN STATD10	CN STATD9	—	CN STATD7	CN STATD6	CN STATD5	CN STATD4	CN STATD3	CN STATD2	CN STATD1	CN STATD0	0000
03A0	CNNED	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNNED15	CNNED14	CNNED13	CNNED12	CNNED11	CNNED10	CNNED9	—	CNNED7	CNNED6	CNNED5	CNNED4	CNNED3	CNNED2	CNNED1	CNNED0	0000
03B0	CNFD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNFD15	CNFD14	CNFD13	CNFD12	CNFD11	CNFD10	CNFD9	—	CNFD7	CNFD6	CNFD5	CNFD4	CNFD3	CNFD2	CNFD1	CNFD0	0000
03C0	SRCON0D	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR1D15	SR1D14	SR1D13	SR1D12	SR1D11	SR1D10	SR1D9	—	SR1D7	SR1D6	SR1D5	SR1D4	SR1D3	SR1D2	SR1D1	SR1D0	0000
03D0	SRCON1D	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR0D15	SR0D14	SR0D13	SR0D12	SR0D11	SR0D10	SR0D9	—	SR0D7	SR0D6	SR0D5	SR0D4	SR0D3	SR0D2	SR0D1	SR0D0	0000

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)  
 Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 12-7: PORTE のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0400	ANSELE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	ANSE9	ANSE8	ANSE7	ANSE6	—	ANSE4	—	—	—	ANSE1	—
0410	TRISE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	TRISE9	TRISE8	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	03FF
0420	PORTE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	RE9	RE8	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0	xxxx
0430	LATE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	LATE9	LATE8	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0	xxxx
0440	ODCE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	ODCE9	ODCE8	ODCE7	ODCE6	ODCE5	ODCE4	ODCE3	ODCE2	ODCE1	ODCE0	0000
0450	CNPUE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	CNPUE9	CNPUE8	CNPUE7	CNPUE6	CNPUE5	CNPUE4	CNPUE3	CNPUE2	CNPUE1	CNPUE0	0000
0460	CNPDE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	CNPDE9	CNPDE8	CNPDE7	CNPDE6	CNPDE5	CNPDE4	CNPDE3	CNPDE2	CNPDE1	CNPDE0	0000
0470	CNCONE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	—	0000
0480	CNENE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	CNIEE9	CNIEE8	CNIEE7	CNIEE6	CNIEE5	CNIEE4	CNIEE3	CNIEE2	CNIEE1	CNIEE0	0000
0490	CNSTATE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	CN STATE9	CN STATE8	CN STATE7	CN STATE6	CN STATE5	CN STATE4	CN STATE3	CN STATE2	CN STATE1	CN STATE0	0000
04A0	CNNEE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	CNNEE9	CNNEE8	CNNEE7	CNNEE6	CNNEE5	CNNEE4	CNNEE3	CNNEE2	CNNEE1	CNNEE0	0000
04B0	CNFE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	CNFE9	CNFE8	CNFE7	CNFE6	CNFE5	CNFE4	CNFE3	CNFE2	CNFE1	CNFE0	0000
04C0	SRCON0E	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	SR1E9	SR1E8	SR1E7	SR1E6	SR1E5	SR1E4	SR1E3	SR1E2	SR1E1	SR1E0	0000
04D0	SRCON1E	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	SR0E9	SR0E8	SR0E7	SR0E6	SR0E5	SR0E4	SR0E3	SR0E2	SR0E1	SR0E0	0000

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)  
 Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。



表 12-8: PORTF のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit															全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
0500	ANSELF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	ANSF13	ANSF12	—	—	—	—	—	—	—	—	—	—	—	—	—
0510	TRISF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	TRISF13	TRISF12	—	—	—	TRISF8	—	—	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	313F
0520	PORTF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	RF13	RF12	—	—	—	RF8	—	—	RF5	RF4	RF3	RF2	RF1	RF0	xxxx
0530	LATF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	LATF13	LATF12	—	—	—	LATF8	—	—	LATF5	LATF4	LATF3	LATF2	LATF1	LATF0	xxxx
0540	ODCF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	ODCF13	ODCF12	—	—	—	ODCF8	—	—	ODCF5	ODCF4	ODCF3	ODCF2	ODCF1	ODCF0	0000
0550	CNPUF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	CNPUF13	CNPUF12	—	—	—	CNPUF8	—	—	CNPUF5	CNPUF4	CNPUF3	CNPUF2	CNPUF1	CNPUF0	0000
0560	CNPDF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	CNPDF13	CNPDF12	—	—	—	CNPDF8	—	—	CNPDF5	CNPDF4	CNPDF3	CNPDF2	CNPDF1	CNPDF0	0000
0570	CNCONF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	—	0000
0580	CNENF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	CNIEF13	CNIEF12	—	—	—	CNIEF8	—	—	CNIEF5	CNIEF4	CNIEF3	CNIEF2	CNIEF1	CNIEF0	0000
0590	CNSTATF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	CN STATF13	CN STATF12	—	—	—	CN STATF8	—	—	CN STATF5	CN STATF4	CN STATF3	CN STATF2	CN STATF1	CN STATF0	0000
05A0	CNNEF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	CNNEF13	CNNEF12	—	—	—	CNNEF8	—	—	CNNEF5	CNNEF4	CNNEF3	CNNEF2	CNNEF1	CNNEF0	0000
05B0	CNFF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	CNFF13	CNFF12	—	—	—	CNFF8	—	—	CNFF5	CNFF4	CNFF3	CNFF2	CNFF1	CNFF0	0000
05C0	SRCON0F	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	SR1F13	SR1F12	—	—	—	SR1F8	—	—	SR1F5	SR1F4	SR1F3	SR1F2	SR1F1	SR1F0	0000
05D0	SRCON1F	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	SR0F13	SR0F12	—	—	—	SR0F8	—	—	SR0F5	SR0F4	SR0F3	SR0F2	SR0F1	SR0F0	0000

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)

Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

表 12-9: PORTG のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0600	ANSELG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ANSG15	—	—	—	—	—	—	ANSG9	ANSG8	ANSG7	ANSG6	—	—	—	—	—	83C0
0610	TRISG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	TRISG15	TRISG14	TRISG13	TRISG12	—	—	TRISG9	TRISG8	TRISG7	TRISG6	—	—	—	—	TRISG1	TRISG0	F3C3
0620	PORTG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	RG15	RG14	RG13	RG12	—	—	RG9	RG8	RG7	RG6	—	—	—	—	RG1	RG0	xxxxx
0630	LATG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	LATG15	LATG14	LATG13	LATG12	—	—	LATG9	LATG8	LATG7	LATG6	—	—	—	—	LATG1	LATG0	xxxxx
0640	ODCG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ODCG15	ODCG14	ODCG13	ODCG12	—	—	ODCG9	ODCG8	ODCG7	ODCG6	—	—	—	—	ODCG1	ODCG0	0000
0650	CNPUG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CNPUG15	CNPUG14	CNPUG13	CNPUG12	—	—	CNPUG9	CNPUG8	CNPUG7	CNPUG6	—	—	—	—	CNPUG1	CNPUG0	0000
0660	CNPDG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CNPDG15	CNPDG14	CNPDG13	CNPDG12	—	—	CNPDG9	CNPDG8	CNPDG7	CNPDG6	—	—	—	—	CNPDG1	CNPDG0	0000
0670	CNCONG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	0000	
0680	CNENG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CNIEG15	CNIEG14	CNIEG13	CNIEG12	—	—	CNIEG9	CNIEG8	CNIEG7	CNIEG6	—	—	—	—	CNIEG1	CNIEG0	0000
0690	CNSTATG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CN STATG15	CN STATG14	CN STATG13	CN STATG12	—	—	CN STATG9	CN STATG8	CN STATG7	CN STATG6	—	—	—	—	CN STATG1	CN STATG0	0000
06A0	CNNEG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CNNEG15	CNNEG14	CNNEG13	CNNEG12	—	—	CNNEG9	CNNEG8	CNNEG7	CNNEG6	—	—	—	—	CNNEG1	CNNEG0	0000
06B0	CNFG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CNFG15	CNFG14	CNFG13	CNFG12	—	—	CNFG9	CNFG8	CNFG7	CNFG6	—	—	—	—	CNFG1	CNFG0	0000
06C0	SRCON0G	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	SR1G15	SR1G14	SR1G13	SR1G12	—	—	SR1G9	SR1G8	SR1G7	SR1G6	—	—	—	—	SR1G1	SR1G0	0000
06D0	SRCON1G	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	SR0G15	SR0G14	SR0G13	SR0G12	—	—	SR0G9	SR0G8	SR0G7	SR0G6	—	—	—	—	SR0G1	SR0G0	0000

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)  
 Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 12-10: PORTH のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0700	ANSELH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	ANSH11	—	—	—	ANSH7	—	—	ANSH4	ANSH3	—	—	—	0898
0710	TRISH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TRISH15	TRISH14	TRISH13	TRISH12	TRISH11	TRISH10	TRISH9	TRISH8	TRISH7	TRISH6	TRISH5	TRISH4	TRISH3	TRISH2	TRISH1	TRISH0	FFFF
0720	PORTH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RH15	RH14	RH13	RH12	RH11	RH10	RH9	RH8	RH7	RH6	RH5	RH4	RH3	RH2	RH1	RH0	xxxxx
0730	LATH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	LATH15	LATH14	LATH13	LATH12	LATH11	LATH10	LATH9	LATH8	LATH7	LATH6	LATH5	LATH4	LATH3	LATH2	LATH1	LATH0	xxxxx
0740	ODCH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ODCH15	ODCH14	ODCH13	ODCH12	ODCH11	ODCH10	ODCH9	ODCH8	ODCH7	ODCH6	ODCH5	ODCH4	ODCH3	ODCH2	ODCH1	ODCH0	0000
0750	CNPUH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPUH15	CNPUH14	CNPUH13	CNPUH12	CNPUH11	CNPUH10	CNPUH9	CNPUH8	CNPUH7	CNPUH6	CNPUH5	CNPUH4	CNPUH3	CNPUH2	CNPUH1	CNPUH0	0000
0760	CNPDH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPDH15	CNPDH14	CNPDH13	CNPDH12	CNPDH11	CNPDH10	CNPDH9	CNPDH8	CNPDH7	CNPDH6	CNPDH5	CNPDH4	CNPDH3	CNPDH2	CNPDH1	CNPDH0	0000
0770	CNCONH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	—	0000
0780	CNENH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNIEH15	CNIEH14	CNIEH13	CNIEH12	CNIEH11	CNIEH10	CNIEH9	CNIEH8	CNIEH7	CNIEH6	CNIEH5	CNIEH4	CNIEH3	CNIEH2	CNIEH1	CNIEH0	0000
0790	CNSTATH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CN STATH15	CN STATH14	CN STATH13	CN STATH12	CN STATH11	CN STATH10	CN STATH9	CN STATH8	CN STATH7	CN STATH6	CN STATH5	CN STATH4	CN STATH3	CN STATH2	CN STATH1	CN STATH0	0000
07A0	CNNEH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNNEH15	CNNEH14	CNNEH13	CNNEH12	CNNEH11	CNNEH10	CNNEH9	CNNEH8	CNNEH7	CNNEH6	CNNEH5	CNNEH4	CNNEH3	CNNEH2	CNNEH1	CNNEH0	0000
07B0	CNFH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNFH15	CNFH14	CNFH13	CNFH12	CNFH11	CNFH10	CNFH9	CNFH8	CNFH7	CNFH6	CNFH5	CNFH4	CNFH3	CNFH2	CNFH1	CNFH0	0000
07C0	SRCON0H	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR1H15	SR1H14	SR1H13	SR1H12	SR1H11	SR1H10	SR1H9	SR1H8	SR1H7	SR1H6	SR1H5	SR1H4	SR1H3	SR1H2	SR1H1	SR1H0	0000
07D0	SRCON1H	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR0H15	SR0H14	SR0H13	SR0H12	SR0H11	SR0H10	SR0H9	SR0H8	SR0H7	SR0H6	SR0H5	SR0H4	SR0H3	SR0H2	SR0H1	SR0H0	0000

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)

Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 12-11: PORTJ のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 (1)	ビットレンジ	Bit																リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0		
0800	ANSELJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ANSJ2	—	—
0810	TRISJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TRISJ15	TRISJ14	TRISJ13	TRISJ12	TRISJ11	TRISJ10	TRISJ9	TRISJ8	TRISJ7	TRISJ6	TRISJ5	TRISJ4	TRISJ3	TRISJ2	TRISJ1	TRISJ0	—	FFFF
0820	PORTJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RJ15	RJ14	RJ13	RJ12	RJ11	RJ10	RJ9	RJ8	RJ7	RJ6	RJ5	RJ4	RJ3	RJ2	RJ1	RJ0	—	xxxxx
0830	LATJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	LATJ15	LATJ14	LATJ13	LATJ12	LATJ11	LATJ10	LATJ9	LATJ8	LATJ7	LATJ6	LATJ5	LATJ4	LATJ3	LATJ2	LATJ1	LATJ0	—	xxxxx
0840	ODCJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ODCJ15	ODCJ14	ODCJ13	ODCJ12	ODCJ11	ODCJ10	ODCJ9	ODCJ8	ODCJ7	ODCJ6	ODCJ5	ODCJ4	ODCJ3	ODCJ2	ODCJ1	ODCJ0	—	0000
0850	CNPUJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPUJ15	CNPUJ14	CNPUJ13	CNPUJ12	CNPUJ11	CNPUJ10	CNPUJ9	CNPUJ8	CNPUJ7	CNPUJ6	CNPUJ5	CNPUJ4	CNPUJ3	CNPUJ2	CNPUJ1	CNPUJ0	—	0000
0860	CNPDJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNPDJ15	CNPDJ14	CNPDJ13	CNPDJ12	CNPDJ11	CNPDJ10	CNPDJ9	CNPDJ8	CNPDJ7	CNPDJ6	CNPDJ5	CNPDJ4	CNPDJ3	CNPDJ2	CNPDJ1	CNPDJ0	—	0000
0870	CNCONJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	—	—	0000
0880	CNENJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNIEJ15	CNIEJ14	CNIEJ13	CNIEJ12	CNIEJ11	CNIEJ10	CNIEJ9	CNIEJ8	CNIEJ7	CNIEJ6	CNIEJ5	CNIEJ4	CNIEJ3	CNIEJ2	CNIEJ1	CNIEJ0	—	0000
0890	CNSTATJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CN STATJ15	CN STATJ14	CN STATJ13	CN STATJ12	CN STATJ11	CN STATJ10	CN STATJ9	CN STATJ8	CN STATJ7	CN STATJ6	CN STATJ5	CN STATJ4	CN STATJ3	CN STATJ2	CN STATJ1	CN STATJ0	—	0000
08A0	CNNEJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNNEJ15	CNNEJ14	CNNEJ13	CNNEJ12	CNNEJ11	CNNEJ10	CNNEJ9	CNNEJ8	CNNEJ7	CNNEJ6	CNNEJ5	CNNEJ4	CNNEJ3	CNNEJ2	CNNEJ1	CNNEJ0	—	0000
08B0	CNFJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CNFJ15	CNFJ14	CNFJ13	CNFJ12	CNFJ11	CNFJ10	CNFJ9	CNFJ8	CNFJ7	CNFJ6	CNFJ5	CNFJ4	CNFJ3	CNFJ2	CNFJ1	CNFJ0	—	0000
08C0	SRCONJ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR1J15	SR1J14	SR1J13	SR1J12	SR1J11	SR1J10	SR1J9	SR1J8	SR1J7	SR1J6	SR1J5	SR1J4	SR1J3	SR1J2	SR1J1	SR1J0	—	0000
08D0	SRCON1J	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SR0J15	SR0J14	SR0J13	SR0J12	SR0J11	SR0J10	SR0J9	SR0J8	SR0J7	SR0J6	SR0J5	SR0J4	SR0J3	SR0J2	SR0J1	SR0J0	—	0000

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)  
 Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 12-12: PORTK のレジスタマップ

仮想アドレス (BF86_#)	レジスタ名 (1)	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
0900	ANSELK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ANSK2	ANSK1	—
0910	TRISK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	TRISK7	TRISK6	TRISK5	TRISK4	TRISK3	TRISK2	TRISK1	TRISK0	00E9
0920	PORTK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	RK7	RK6	RK5	RK4	RK3	RK2	RK1	RK0	xxxxxx
0930	LATK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	LATK7	LATK6	LATK5	LATK4	LATK3	LATK2	LATK1	LATK0	xxxxxx
0940	ODCK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	ODCK7	ODCK6	ODCK5	ODCK4	ODCK3	ODCK2	ODCK1	ODCK0	0000
0950	CNPUK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	CNPUK7	CNPUK6	CNPUK5	CNPUK4	CNPUK3	CNPUK2	CNPUK1	CNPUK0	0000
0960	CNPDK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	CNPDK7	CNPDK6	CNPDK5	CNPDK4	CNPDK3	CNPDK2	CNPDK1	CNPDK0	0000
0970	CNCONK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	EDGE DETECT	—	—	—	—	—	—	—	—	—	—	—	0000
0980	CNENK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	CNIEK7	CNIEK6	CNIEK5	CNIEK4	CNIEK3	CNIEK2	CNIEK1	CNIEK0	0000
0990	CNSTATK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	CN STATK7	CN STATK6	CN STATK5	CN STATK4	CN STATK3	CN STATK2	CN STATK1	CN STATK0	0000
09A0	CNNEK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	CNNEK7	CNNEK6	CNNEK5	CNNEK4	CNNEK3	CNNEK2	CNNEK1	CNNEK0	0000
09B0	CNFK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	CNFK7	CNFK6	CNFK5	CNFK4	CNFK3	CNFK2	CNFK1	CNFK0	0000
09C0	SRCON0K	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	SR1K7	SR1K6	SR1K5	SR1K4	SR1K3	SR1K2	SR1K1	SR1K0	0000
09D0	SRCON1K	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	SR0K7	SR0K6	SR0K5	SR0K4	SR0K3	SR0K2	SR0K1	SR0K0	0000

凡例: x = リセット時の値は未知、— = 未実装 (「0」として読み出し、リセット値は 16 進数で表記)  
 Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 12-13: ペリフェラル ピンセレクト入力のレジスタマップ

仮アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit																リセット 全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
1404	INT1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	INT1R<3:0>				0000
1408	INT2R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	INT2R<3:0>				0000
140C	INT3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	INT3R<3:0>				0000
1410	INT4R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	INT4R<3:0>				0000
1418	T2CKR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	T2CKR<3:0>				0000
141C	T3CKR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	T3CKR<3:0>				0000
1420	T4CKR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	T4CKR<3:0>				0000
1424	T5CKR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	T5CKR<3:0>				0000
1428	T6CKR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	T6CKR<3:0>				0000
142C	T7CKR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	T7CKR<3:0>				0000
1430	T8CKR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	T8CKR<3:0>				0000
1434	T9CKR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	T9CKR<3:0>				0000
1438	IC1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	IC1R<3:0>				0000
143C	IC2R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	IC2R<3:0>				0000
1440	IC3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	IC3R<3:0>				0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

表 12-13: ペリフェラル ピンセレクト入力のレジスタマップ (続き)

仮アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
1444	IC4R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	IC4R<3:0>				0000
1448	IC5R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	IC5R<3:0>				0000
144C	IC6R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	IC6R<3:0>				0000
1450	IC7R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	IC7R<3:0>				0000
1454	IC8R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	IC8R<3:0>				0000
1458	IC9R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	IC9R<3:0>				0000
1460	OCFAR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	OCFAR<3:0>				0000
1468	U1RXR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	U1RXR<3:0>				0000
146C	U1CTSR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	U1CTSR<3:0>				0000
1470	U2RXR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	U2RXR<3:0>				0000
1474	U2CTSR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	U2CTSR<3:0>				0000
1478	U3RXR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	U3RXR<3:0>				0000
147C	U3CTSR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	U3CTSR<3:0>				0000
1480	U4RXR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	U4RXR<3:0>				0000
1484	U4CTSR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	U4CTSR<3:0>				0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

表 12-13: ペリフェラル ピンセレクト入力のレジスタマップ (続き)

仮アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit																全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0		
1488	U5RXR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	U5RXR<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
148C	U5CTSR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	U5CTSR<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
1490	U6RXR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	U6RXR<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
1494	U6CTSR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	U6CTSR<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
149C	SDI1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDI1R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14A0	SS1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SS1R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14A8	SDI2R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDI2R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14AC	SS2R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SS2R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14B4	SDI3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDI3R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14B8	SS3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SS3R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14C0	SDI4R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDI4R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14C4	SS4R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SS4R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14CC	SDI5R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDI5R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14D0	SS5R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SS5R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000
14D8	SDI6R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDI6R<3:0>	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—		0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記



表 12-13: ペリフェラル ピンセレクト入力のレジスタマップ (続き)

仮アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
14DC	SS6R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	SS6R<3:0>				0000
14E0	C1RXR <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	C1RXR<3:0>				0000
14E4	C2RXR <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	C2RXR<3:0>				0000
14E8	REFCLKI1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	REFCLKI1R<3:0>				0000
14F0	REFCLKI3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	REFCLKI3R<3:0>				0000
14F4	REFCLKI4R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	REFCLKI4R<3:0>				0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

表 12-14: ペリフェラル ピンセレクト出力のレジスタマップ

アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
1538	RPA14R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPA14R<3:0>
153C	RPA15R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPA15R<3:0>
1540	RPB0R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB0R<3:0>
1544	RPB1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB1R<3:0>
1548	RPB2R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB2R<3:0>
154C	RPB3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB3R<3:0>
1554	RPB5R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB5R<3:0>
1558	RPB6R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB6R<3:0>
155C	RPB7R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB7R<3:0>
1560	RPB8R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB8R<3:0>
1564	RPB9R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB9R<3:0>
1568	RPB10R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB10R<3:0>
157C	RPB15R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPB15R<3:0>
1584	RPC1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPC1R<3:0>
1588	RPC2R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPC2R<3:0>
158C	RPC3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPC3R<3:0>
15B4	RPC13R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPC13R<3:0>
15B8	RPC14R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPC14R<3:0>
15C0	RPD0R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD0R<3:0>

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

表 12-14: ペリフェラル ピンセレクト出力のレジスタマップ (続き)

仮アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
15C8	RPD2R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD2R<3:0>			0000
15CC	RPD3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD3R<3:0>			0000
15D0	RPD4R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD4R<3:0>			0000
15D4	RPD5R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD5R<3:0>			0000
15D8	RPD6R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD6R<3:0>			0000
15DC	RPD7R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD7R<3:0>			0000
15E4	RPD9R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD9R<3:0>			0000
15EC	RPD11R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD11R<3:0>			0000
15F0	RPD12R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD12R<3:0>			0000
15F8	RPD14R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPD14R<3:0>			0000
160C	RPE3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPE3R<3:0>			0000
1614	RPE5R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPE5R<3:0>			0000
1620	RPE8R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPE8R<3:0>			0000
1624	RPE9R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPE9R<3:0>			0000
1640	RPF0R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPF0R<3:0>			0000
1644	RPF1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPF1R<3:0>			0000
1648	RPF2R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPF2R<3:0>			0000
164C	RPF3R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPF3R<3:0>			0000
1650	RPF4R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	RPF4R<3:0>			0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

表 12-14: ペリフェラル ピンセレクト出力のレジスタマップ (続き)

仮番アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
1654	RPF5R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPF5R<3:0>				0000
1660	RPF8R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPF8R<3:0>				0000
1670	RPF12R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPF12R<3:0>				0000
1680	RPG0R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPG0R<3:0>				0000
1684	RPG1R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPG1R<3:0>				0000
169C	RPG7R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPG7R<3:0>				0000
16A0	RPG8R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPG8R<3:0>				0000
16A4	RPG9R	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	RPG9R<3:0>				0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 12-1: [pin name]R: ペリフェラル ピンセレクト入力のレジスタマップ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0
[pin name]R<3:0>								

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-4 **未実装:** 「0」として読み出し

bit 3-0 **[pin name]R<3:0>:** ペリフェラル ピンセレクト入力ビット

[pin name] は、周辺モジュールに入力を割り当てるために使うピンを表します。入力ピンの選択値については表 12-1 を参照してください。

IOLOCK コンフィグレーション ビット (CFGCON<13>) が「0」の場合にのみ、レジスタの値を変更できます。

レジスタ 12-2: RPnR: ペリフェラル ピンセレクト出力レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0
RPnR<3:0>								

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-4 **未実装:** 「0」として読み出し

bit 3-0 **RPnR<3:0>:** ペリフェラル ピンセレクト出力ビット

出力ピンの選択値については表 12-2 を参照してください。

**Note:** IOLOCK コンフィグレーション ビット (CFGCON<13>) が「0」の場合にのみ、レジスタの値を変更できます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 12-3: CNCONx: PORTx 向け状態変更通知制御レジスタ (x = A ~ G)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
	ON	—	—	—	EDGE DETECT	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ON:** 状態変化通知 (CN) 制御 ON ビット

1 = CN を有効にする  
 0 = CN を無効にする

bit 14-12 **未実装:** 「0」として読み出し

bit 11 **EDGEDETECT:** エッジ検出タイプ制御ビット

1 = ピンで全てのエッジを検出する (CNF<sub>x</sub> を CN イベント向けに使う)  
 0 = ピンで全てのエッジを検出する (CNSTAT<sub>x</sub> を CN イベント向けに使う)

bit 10-0 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## 13.0 Timer1

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 14. Timers』(DS60001105) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

PIC32MZ DA は同期 / 非同期 16 ビットタイマを 1 つ備えています。このタイマは各種のタイミングアプリケーションおよび外部イベントのカウント用にフリーランニングインターバル タイマとして動作可能です。

このタイマは、低消費電力セカンダリ オシレータ (Sosc) と一緒にリアルタイム クロック アプリケーション向けに使う事もできます。

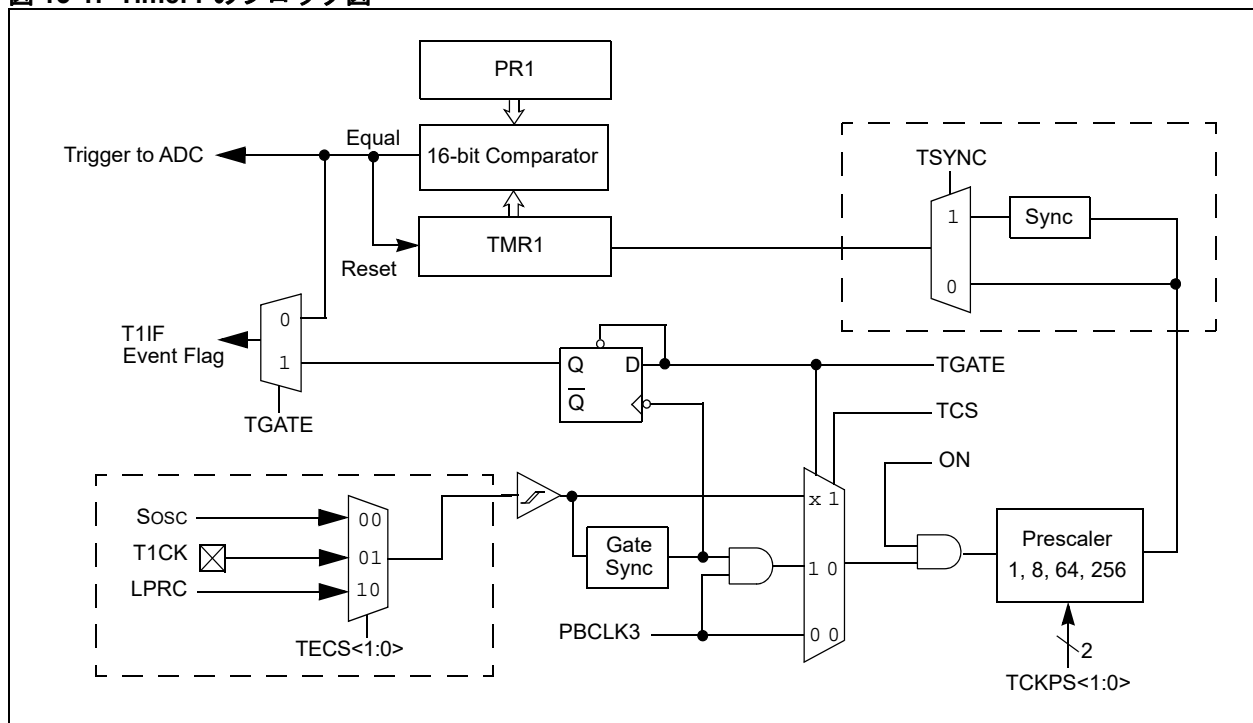
Timer1 は以下のモードをサポートします。

- 同期内部タイマ
- 同期内部ゲート付きタイマ
- 同期外部タイマ
- 非同期外部タイマ

### 13.1 その他の特長

- 選択可能なクロック プリスケーラ
- スリープおよびアイドル中のタイマ動作
- CLR、SET、INV レジスタを使った高速なビット操作
- Sosc を使ってリアルタイム クロックとして機能可能な非同期モード
- ADC イベントトリガ

図 13-1: Timer1 のブロック図



## 13.2 Timer1 制御レジスタ

表 13-1: Timer1 のレジスタマップ

仮想アドレス (BF8_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0000	T1CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	TWDIS	TWIP	—	TECS<1:0>	TGATE	—	TCKPS<1:0>	—	TSYNC	TCS	—	—	—	0000
0010	TMR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	TMR1<15:0>																0000
0020	PR1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	PR1<15:0>																FFFF

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 13-1: T1CON: タイプ A タイマ制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	U-0	R/W-0	R/W-0	R-0	U-0	R/W-0	R/W-0
	ON	—	SIDL	TWDIS	TWIP	—	TECS<1:0>	
7:0	R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
	TGATE	—	TCKPS<1:0>		—	TSYNC	TCS	—

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ON:** タイマ ON ビット  
 1 = タイマを有効にする  
 0 = タイマを無効にする

bit 14 **未実装:** 「0」として読み出し

bit 13 **SIDL:** アイドル中停止ビット  
 1 = デバイスがアイドルモードに移行した時に動作を停止する  
 0 = アイドル中も動作を継続する

bit 12 **TWDIS:** 非同期タイマ書き込みディセーブル ビット  
 1 = 保留中の書き込み動作が完了するまで TMR1 への書き込みを無視する  
 0 = 連続的な書き込みを可能にする (レガシーの非同期タイマ機能)

bit 11 **TWIP:** 非同期タイマ書き込み中ステータスビット  
非同期タイマモードの場合:  
 1 = TMR1 レジスタへの非同期書き込みを実行中  
 0 = TMR1 レジスタへの非同期書き込みは完了した  
同期タイマモードの場合:  
 このビットは常に「0」として読み出されます。

bit 10 **未実装:** 「0」として読み出し

bit **TECS<1:0>:** Timer1 外部クロック選択ビット  
 11 = 予約済み  
 10 = LPRC から外部クロックを供給する  
 01 = T1CK ピンから外部クロックを供給する  
 00 = Sosc から外部クロックを供給する

bit 7 **TGATE:** タイマゲート時間積算イネーブルビット  
TCS = 1 の場合:  
 このビットは無視されます。  
TCS = 0 の場合:  
 1 = ゲート時間積算を有効にする  
 0 = ゲート時間積算を無効にする

bit 6 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 13-1: T1CON: タイプ A タイマ制御レジスタ (続き)

bit 5-4 **TCKPS<1:0>**: タイマ入力クロック プリスケール選択ビット

11 = 1:256 のプリスケール値

10 = 1:64 のプリスケール値

01 = 1:8 のプリスケール値

00 = 1:1 のプリスケール値

bit 3 **未実装**: 「0」 として読み出し

bit 2 **TSYNC**: タイマ外部クロック入力同期選択ビット

TCS = 1 の場合:

1 = 外部クロック入力は同期される

0 = 外部クロック入力は同期されない

TCS = 0 の場合:

このビットは無視されます。

bit 1 **TCS**: タイマクロック源選択ビット

1 = TECS<1:0> ビットで定義された外部クロック

0 = 内部周辺モジュール用クロック

bit 0 **未実装**: 「0」 として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## 14.0 Timer2/3、Timer4/5、Timer6/7、Timer8/9

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 14. Timers』(DS60001105) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

PIC32 MZ GA ファミリのデバイスは 8 個の同期 16 ビットタイマ (既定値) を備えています。これらのタイマは各種のタイミングアプリケーションおよび外部イベントのカウント用にフリーランニングインターバルタイマとして動作可能です。

これらのタイマは以下のモードをサポートします。

- 同期内部 16 ビットタイマ
- 同期内部 16 ビットゲート付きタイマ
- 同期外部 16 ビットタイマ

Timer2 と Timer3、Timer4 と Timer5、Timer6 と Timer7、Timer8 と Timer9 を組み合わせる事で、4 個の 32 ビット同期タイマが使えます。

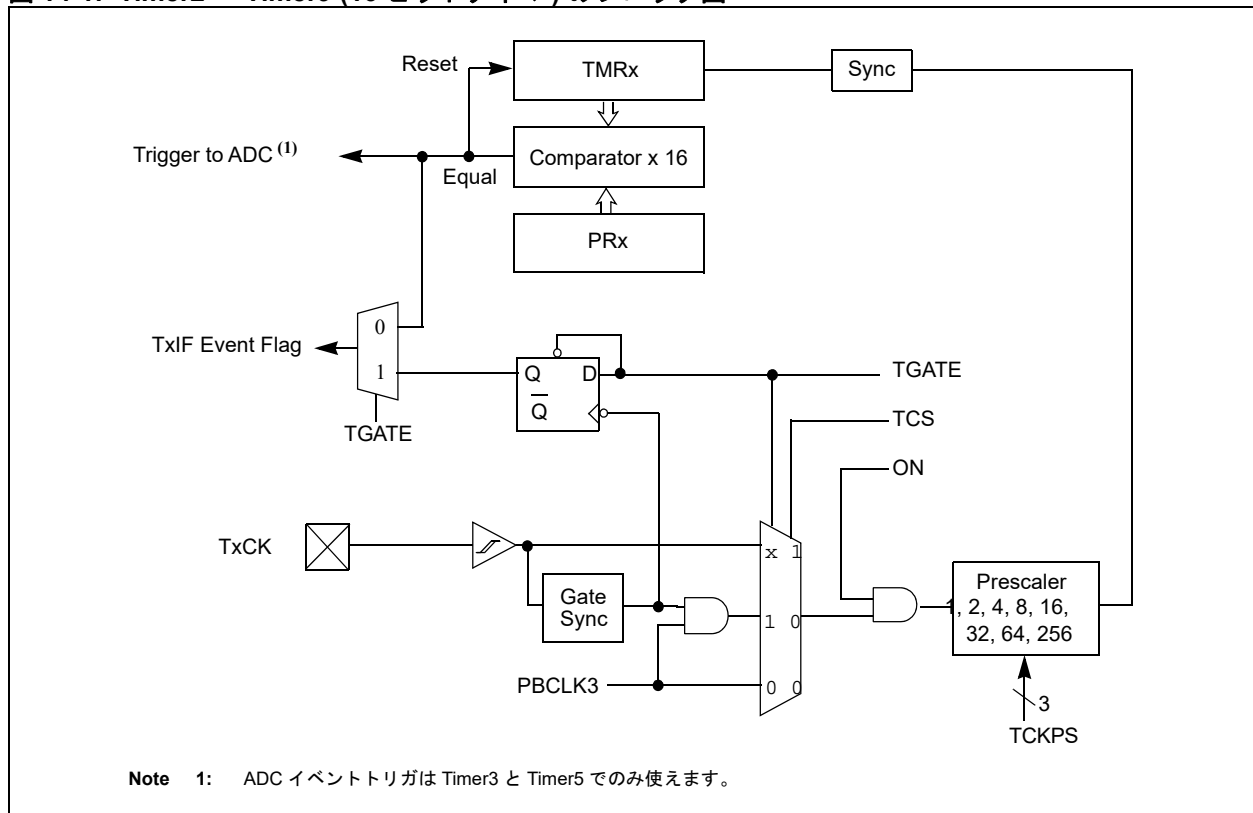
32 ビットタイマは以下の 3 通りのモードで動作できません。

- 同期内部 32 ビットタイマ
- 同期内部 32 ビットゲート付きタイマ
- 同期外部 32 ビットタイマ

### 14.1 その他の特長

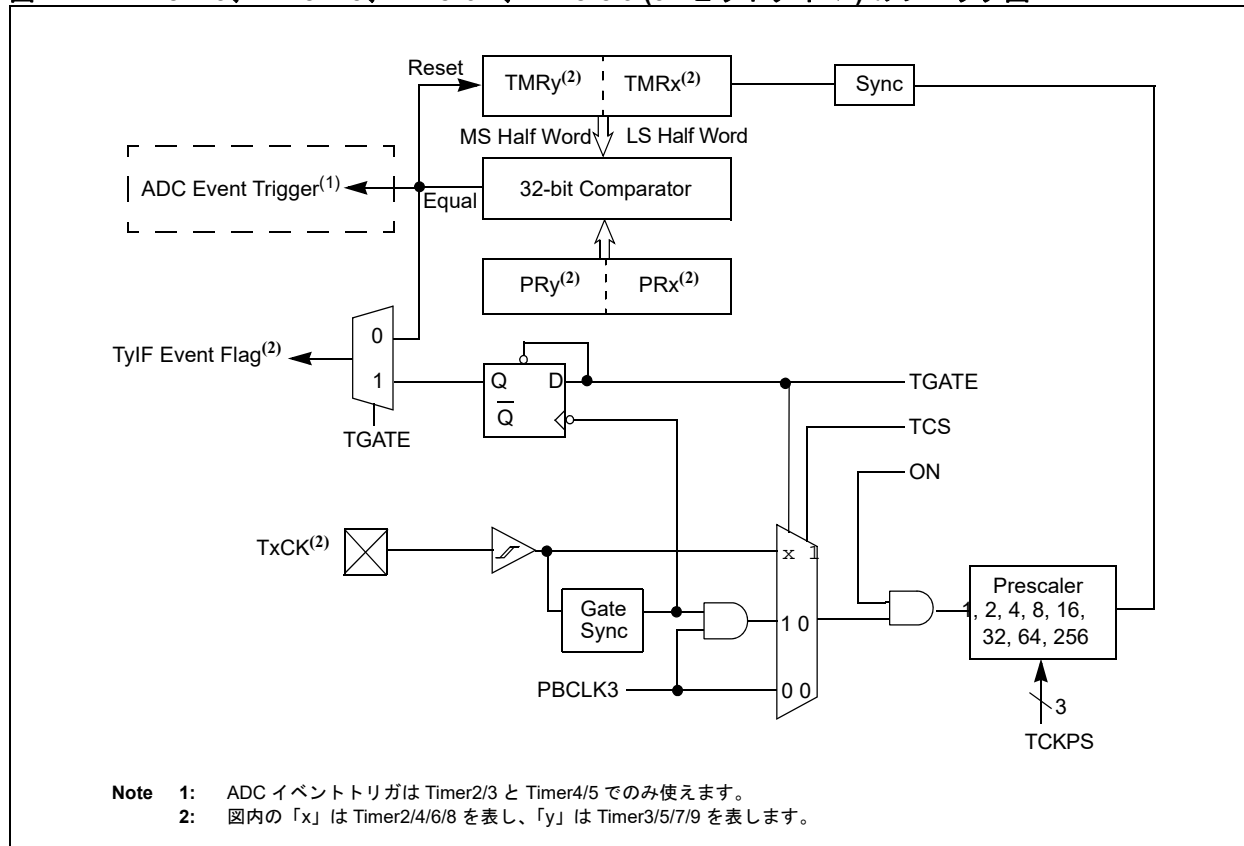
- 選択可能なクロック プリスケーラ
- CPU アイドル中のタイマ動作
- 入力キャプチャおよび出力コンペア モジュール向けのタイムベース (Timer2 ~ Timer7 のみ)
- ADC イベントトリガ (Timer3 と Timer5 のみ)
- CLR、SET、INV レジスタを使った高速なビット操作

図 14-1: Timer2 ~ Timer9 (16 ビットタイマ) のブロック図



# PIC32MZ グラフィック (DA) ファミリ

図 14-2: Timer2/3、Timer4/5、Timer6/7、Timer8/9 (32 ビットタイマ) のブロック図



## 14.2 Timer2 ~ Timer9 制御レジスタ

表 14-1: Timer2 ~ Timer9 のレジスタマップ

仮想アドレス (BF8#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0200	T2CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	TGATE	TCKPS<2:0>			—	T32	—	TCS	—
0210	TMR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TMR2<15:0>																0000
0220	PR2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PR2<15:0>																FFFF
0400	T3CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	TGATE	TCKPS<2:0>			—	—	TCS	—
0410	TMR3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TMR3<15:0>																0000
0420	PR3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PR3<15:0>																FFFF
0600	T4CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	TGATE	TCKPS<2:0>			T32	—	TCS	—
0610	TMR4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TMR4<15:0>																0000
0620	PR4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PR4<15:0>																FFFF
0800	T5CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	TGATE	TCKPS<2:0>			—	—	TCS	—
0810	TMR5	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TMR5<15:0>																0000
0820	PR5	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PR5<15:0>																FFFF
0A00	T6CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	TGATE	TCKPS<2:0>			T32	—	TCS	—
0A10	TMR6	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TMR6<15:0>																0000
0A20	PR6	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PR6<15:0>																FFFF
0C00	T7CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	TGATE	TCKPS<2:0>			—	—	TCS	—

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 14-1: Timer2 ~ Timer9 のレジスタマップ (続き)

仮想アドレス (BF8_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0C10	TMR7	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TMR3<15:0>																0000
0C20	PR7	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PR3<15:0>																FFFF
0E00	T8CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	TGATE	TCKPS<2:0>			T32	—	TCS	—	0000
0E10	TMR8	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TMR4<15:0>																0000
0E20	PR8	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PR4<15:0>																FFFF
1000	T9CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	TGATE	TCKPS<2:0>			—	—	TCS	—	0000
1010	TMR9	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	TMR5<15:0>																0000
1020	PR9	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PR5<15:0>																FFFF

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 14-1: TxCON: タイプ B タイマ制御レジスタ (x = 2 ~ 9)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-0 ON <sup>(1)</sup>	U-0 —	R/W-0 SIDL <sup>(2)</sup>	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	R/W-0 TGATE <sup>(1)</sup>	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
		TCKPS<2:0> <sup>(1)</sup>			T32 <sup>(3)</sup>	—	TCS <sup>(1)</sup>	—

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ON:** タイマ ON ビット<sup>(1)</sup>

1 = モジュールを有効にする

0 = モジュールを無効にする

bit 14 **未実装:** 「0」として読み出し

bit 13 **SIDL:** アイドルモード時停止ビット<sup>(2)</sup>

1 = デバイスがアイドルモードに移行した時に動作を停止する

0 = アイドル中も動作を継続する

bit 12-8 **未実装:** 「0」として読み出し

bit 7 **TGATE:** タイマゲート時間積算イネーブルビット<sup>(1)</sup>

TCS = 1 の場合:

このビットは無視され「0」として読み出されます。

TCS = 0 の場合:

1 = ゲート時間積算を有効にする

0 = ゲート時間積算を無効にする

bit 6-4 **TCKPS<2:0>:** タイマ入カクロック プリスケール選択ビット<sup>(1)</sup>

111 = 1:256 のプリスケール値

110 = 1:64 のプリスケール値

101 = 1:32 のプリスケール値

100 = 1:16 のプリスケール値

011 = 1:8 のプリスケール値

010 = 1:4 のプリスケール値

001 = 1:2 のプリスケール値

000 = 1:1 のプリスケール値

bit 3 **T32:** 32 ビット タイマモード選択ビット<sup>(3)</sup>

1 = 奇数番号タイマと偶数番号タイマを組み合わせて 32 ビットタイマとして使う

0 = 各タイマを 16 ビットタイマとして使う

**Note 1:** 32 ビットモードで動作している場合、このビットは奇数番号タイマ (Timer1、Timer3、Timer5、Timer7、Timer9) に対して効果を持ちません。全てのタイマ機能は偶数番号タイマで設定します。

**2:** 32 ビットモードで動作している場合、アイドル中の 32 ビットタイマ動作を有効にするには、奇数番号タイマのこのビットをクリアする必要があります。

**3:** このビットは偶数番号タイマ (Timer2、Timer4、Timer6、Timer8) でのみ利用できます。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 14-1: TxCON: タイプ B タイマ制御レジスタ (x = 2 ~ 9) ( 続き )

- bit 2      **未実装**: 「0」として読み出し
- bit 1      **TCS**: タイマクロック源選択ビット (1)  
            1 = TxCK ピンからの外部クロック  
            0 = 内部の周辺モジュール用クロック
- bit 0      **未実装**: 「0」として読み出し

- Note 1**: 32 ビットモードで動作している場合、このビットは奇数番号タイマ (Timer1、Timer3、Timer5、Timer7、Timer9) に対して効果を持ちません。全てのタイマ機能は偶数番号タイマで設定します。
- 2**: 32 ビットモードで動作している場合、アイドル中の 32 ビットタイマ動作を有効にするには、奇数番号タイマのこのビットをクリアする必要があります。
- 3**: このビットは偶数番号タイマ (Timer2、Timer4、Timer6、Timer8) でのみ利用できます。



# PIC32MZ グラフィック (DA) ファミリ

## 15.0 入力キャプチャ

**Note:** 本書はPIC32MZグラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 15. Input Capture』(DS60001122)を参照してください。この文書は、Microchip社PIC32ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

入力キャプチャ モジュールは周波数 (周期) およびパルス計測が必要なアプリケーションに役立ちます。

ICxピンでイベントが発生すると、入力キャプチャモジュールは選択されているタイムベースレジスタの16ビットまたは32ビット値をキャプチャします。

キャプチャ イベントは以下により発生します。

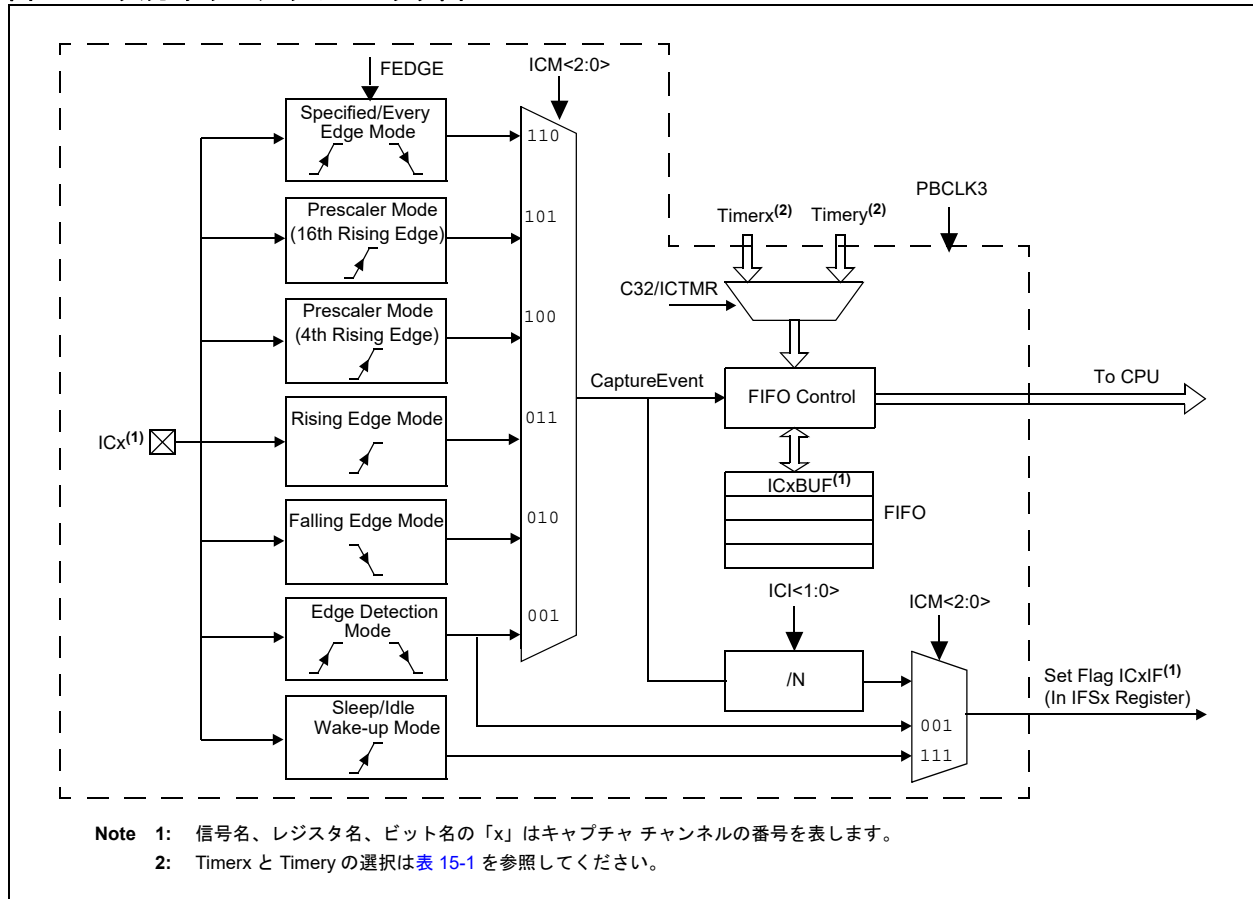
- 指定されたエッジでキャプチャを開始し、全てのエッジ (立ち上がり / 立ち下りの両方) でタイム値をキャプチャ
- プリスケアラ キャプチャ イベントモード:
  - ICx ピン入力の立ち上がりエッジ 4 回ごとにタイム値をキャプチャ
  - ICx ピン入力の立ち上がりエッジ 16 回ごとにタイム値をキャプチャ

各入力キャプチャ チャンネルには、タイムベース用に6個ある16ビットタイマの中から1つを選択するか、2個の16ビットタイマを組み合わせた32ビットタイマを選択できます。選択したタイマには、内部クロックまたは外部クロックが使えます。

その他の特長は以下の通りです。

- キャプチャピンによるCPUスリープ/アイドルからの復帰
- 入力キャプチャ イベントでの割り込み
- キャプチャ値用の4ワードFIFOバッファ (各バッファ位置 (1/2/3/4) が書き込まれた時に割り込みを生成可能)
- 入力キャプチャを使って外部割り込み要因を追加する事が可能

図 15-1: 入力キャプチャのブロック図



# PIC32MZ グラフィック (DA) ファミリ

---

各入力キャプチャ モジュールのタイマ源は、CFGCON レジスタの ICACLK ビットの設定によって決まります。表 15-1 に、使用可能な設定を示します。

表 15-1: タイマ源の設定

入力キャプチャ モジュール	Timerx	Timery
ICACLK (CFGCON<17>) = 0		
IC1	Timer2	Timer3
⋮	⋮	⋮
IC9	Timer 2	Timer 3
ICACLK (CFGCON<17>) = 1		
IC1	Timer4	Timer5
IC2	Timer4	Timer5
IC3	Timer4	Timer5
IC4	Timer2	Timer3
IC5	Timer2	Timer3
IC6	Timer2	Timer3
IC7	Timer6	Timer7
IC8	Timer6	Timer7
IC9	Timer6	Timer7

## 15.1 入力キャプチャ制御レジスタ

TABLE 15-2: 入力キャプチャ 1 ~ 9 のレジスタマップ

仮想アドレス (BF84_#)	レジスタ名	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
2000	IC1CON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>	0000			
2010	IC1BUF	31:16	IC1BUF<31:0>																xxxx
		15:0																	xxxx
2200	IC2CON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>	0000			
2210	IC2BUF	31:16	IC2BUF<31:0>																xxxx
		15:0																	xxxx
2400	IC3CON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>	0000			
2410	IC3BUF	31:16	IC3BUF<31:0>																xxxx
		15:0																	xxxx
2600	IC4CON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>	0000			
2610	IC4BUF	31:16	IC4BUF<31:0>																xxxx
		15:0																	xxxx
2800	IC5CON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>	0000			
2810	IC5BUF	31:16	IC5BUF<31:0>																xxxx
		15:0																	xxxx
2A00	IC6CON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>	0000			
2A10	IC6BUF	31:16	IC6BUF<31:0>																xxxx
		15:0																	xxxx
2C00	IC7CON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>	0000			
2C10	IC7BUF	31:16	IC7BUF<31:0>																xxxx
		15:0																	xxxx
2E00	IC8CON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>	0000			
2E10	IC8BUF	31:16	IC8BUF<31:0>																xxxx
		15:0																	xxxx
3000	IC9CON <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	FEDGE	C32	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>	0000			
3010	IC9BUF	31:16	IC9BUF<31:0>																xxxx
		15:0																	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: このレジスタは対応する CLR、SET、INV レジスタを持ち、それらはレジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 15-1: ICxCON: 入力キャプチャ x 制御レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-0 ON	U-0 —	R/W-0 SIDL	U-0 —	U-0 —	U-0 —	R/W-0 FEDGE	R/W-0 C32
7:0	R/W-0 ICTMR	R/W-0 ICI<1:0>	R/W-0	R-0 ICOV	R-0 ICBNE	R/W-0	R/W-0	R/W-0 ICM<2:0>

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット  
 -n = POR 時のビット値: (0、1、x = ビットは未知)      P = プログラミング可能ビット      r = 予約済みビット

- bit 31-16      **未実装:** 「0」として読み出し
- bit 15      **ON:** 入力キャプチャ モジュール イネーブルビット  
 1 = モジュールを有効にする  
 0 = モジュールを無効にしてリセットする (クロックと割り込み生成を無効にし、SFRの変更を許可する)
- bit 14      **未実装:** 「0」として読み出し
- bit 13      **SIDL:** アイドル中停止制御ビット  
 1 = CPU アイドルモード時に停止する  
 0 = CPU アイドルモード時も動作を続ける
- bit 12-10      **未実装:** 「0」として読み出し
- bit 9      **FEDGE:** キャプチャ開始エッジ選択ビット (モード 6 (ICM<2:0> = 110) の場合にのみ使用)  
 1 = 最初に立ち上がりエッジをキャプチャする  
 0 = 最初に立ち下がりエッジをキャプチャする
- bit 8      **C32:** 32 ビットキャプチャ選択ビット  
 1 = 32 ビットタイマ リソースをキャプチャする  
 0 = 16 ビットタイマ リソースをキャプチャする
- bit 7      **ICTMR:** タイマ選択ビット (C32 (ICxCON<8>) = 「1」の場合、タイマの選択に影響しない)<sup>(1)</sup>  
 0 = Timery をキャプチャのカウンタ源に使う  
 1 = Timerx をキャプチャのカウンタ源に使う
- bit 6-5      **ICI<1:0>:** 割り込み制御ビット  
 11 = キャプチャ イベント 4 回毎に割り込む  
 10 = キャプチャ イベント 3 回毎に割り込む  
 01 = キャプチャ イベント 2 回毎に割り込む  
 00 = キャプチャ イベント毎に割り込む
- bit 4      **ICOV:** 入力キャプチャ オーバーフロー ステータスフラグ ビット (読み出し専用)  
 1 = 入力キャプチャ オーバーフローが発生した  
 0 = 入力キャプチャ オーバーフローは発生していない
- bit 3      **ICBNE:** 入力キャプチャ バッファ非エンプティ ステータスビット (読み出し専用)  
 1 = 入力キャプチャ バッファはエンプティではない (少なくとも 1 つのキャプチャ値を読み出せる)  
 0 = 入力キャプチャ バッファはエンプティである
- bit 2-0      **ICM<2:0>:** 入力キャプチャモード選択ビット  
 111 = 割り込み専用モード - スリープまたはアイドル中にのみサポート  
 110 = 単純キャプチャ イベントモード - 指定極性のエッジでキャプチャを開始し、その後は全エッジでキャプチャする  
 101 = プリスケール キャプチャ イベントモード - 立ち上がりエッジ 6 回毎にキャプチャする  
 100 = プリスケール キャプチャ イベントモード - 立ち上がりエッジ 4 回毎にキャプチャする  
 011 = 単純キャプチャ イベントモード - 立ち上がりエッジでキャプチャする  
 010 = 単純キャプチャ イベントモード - 立ち下がりエッジでキャプチャする  
 001 = エッジ検出モード - 全エッジ (立ち上がり / 立ち下がり両エッジ) でキャプチャする  
 000 = 入力キャプチャ モジュールを無効にする

**Note 1:** Timerx と Timery の選択は表 15-1 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## 16.0 出力コンペア

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 16. Output Compare』(DS60001111) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

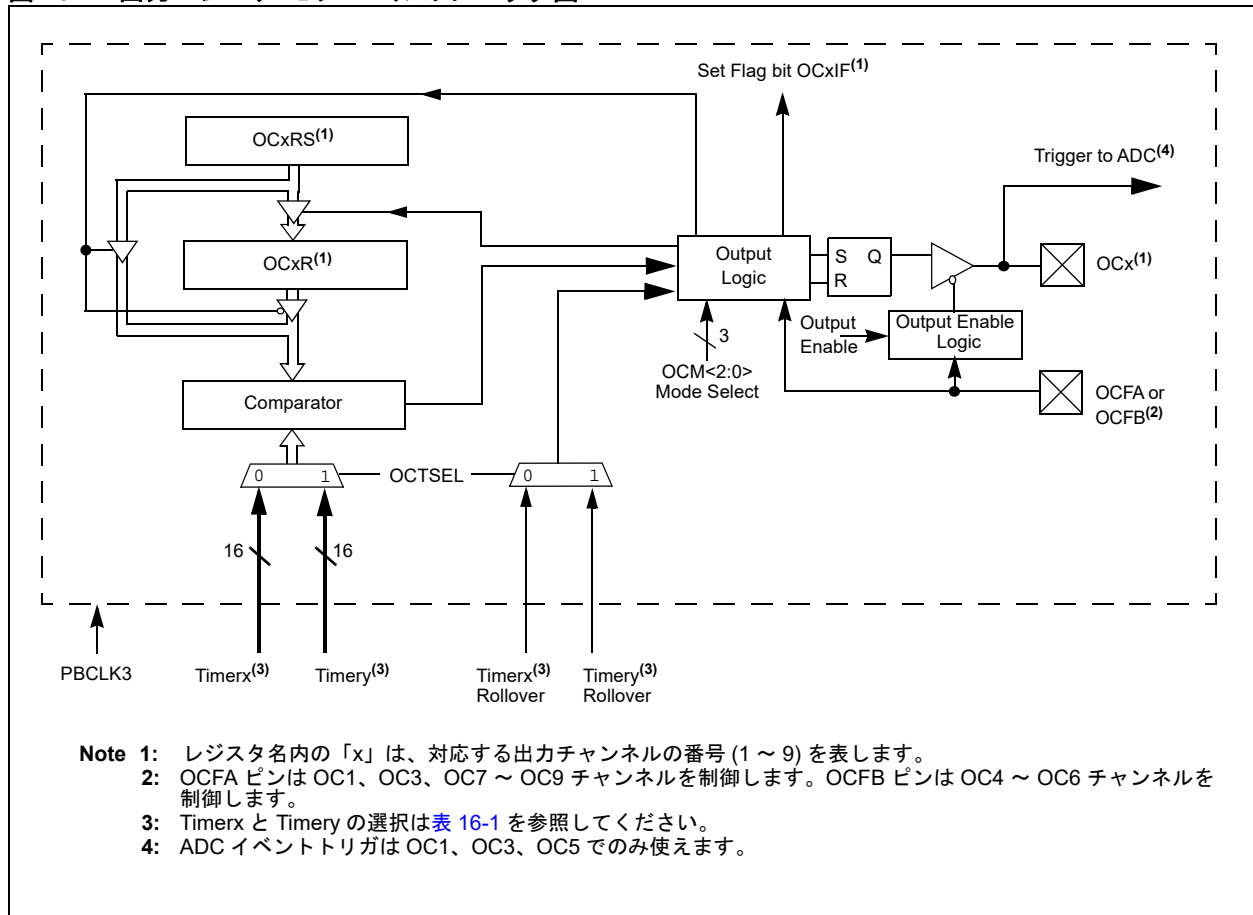
出力コンペア モジュールは、選択されたタイムベース イベントへの応答として単発または連続パルスを生成するために使います。

全ての動作モードにおいて、出力コンペア モジュールは OCxR および / または OCxRS レジスタの値と選択されたタイマの値を比較します。値が一致すると、出力コンペア モジュールは選択された動作モードに基づいてイベントを生成します。

出力コンペア モジュールの主な特長は以下の通りです。

- 複数の出力コンペア モジュールをデバイスに内蔵
- コンペアイベント時のプログラマブルな割り込み生成
- シングルおよびダブルコンペア モード
- 単発または連続出力パルスの生成
- パルス幅変調 (PWM) モード
- ハードウェアによる PWM 異常検出と自動的な出力無効化
- 16ビットまたは32ビットのタイムベースを選択可能
- 2つある 16ビットタイムベースのどちらか、または1つの 32ビットタイムベースを使って動作可能
- ADC イベントトリガ

図 16-1: 出力コンペア モジュールのブロック図



# PIC32MZ グラフィック (DA) ファミリ

---

各出力コンペア モジュールのタイマ源は、CFGCON レジスタの OCACLK ビットの設定によって決まります。表 16-1 に、使用可能な設定を示します。

表 16-1: タイマ源の設定

出力コンペア モジュール	Timerx	Timery
OCACLK (CFGCON<16>) = 0		
OC1	Timer2	Timer3
⋮	⋮	⋮
OC9	Timer 2	Timer 3
OCACLK (CFGCON<16>) = 1		
OC1	Timer4	Timer5
OC2	Timer4	Timer5
OC3	Timer4	Timer5
OC4	Timer2	Timer3
OC5	Timer2	Timer3
OC6	Timer2	Timer3
OC7	Timer6	Timer7
OC8	Timer6	Timer7
OC9	Timer6	Timer7

## 16.1 出力コンペア制御レジスタ

表 16-2: 出力コンペア 1 ~ 9 のレジスタマップ

仮想アドレス (BF0#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
4000	OC1CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>		0000
4010	OC1R	31:16	OC1R<31:0>																xxxx
		15:0																	xxxx
4020	OC1RS	31:16	OC1RS<31:0>																xxxx
		15:0																	xxxx
4200	OC2CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>		0000
4210	OC2R	31:16	OC2R<31:0>																xxxx
		15:0																	xxxx
4220	OC2RS	31:16	OC2RS<31:0>																xxxx
		15:0																	xxxx
4400	OC3CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>		0000
4410	OC3R	31:16	OC3R<31:0>																xxxx
		15:0																	xxxx
4420	OC3RS	31:16	OC3RS<31:0>																xxxx
		15:0																	xxxx
4600	OC4CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>		0000
4610	OC4R	31:16	OC4R<31:0>																xxxx
		15:0																	xxxx
4620	OC4RS	31:16	OC4RS<31:0>																xxxx
		15:0																	xxxx
4800	OC5CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>		0000
4810	OC5R	31:16	OC5R<31:0>																xxxx
		15:0																	xxxx
4820	OC5RS	31:16	OC5RS<31:0>																xxxx
		15:0																	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

表 16-2: 出力コンペア 1 ~ 9 のレジスタマップ (続き)

仮想アドレス (BF84_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
4A00	OC6CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>		0000
4A10	OC6R	31:16	OC6R<31:0>																xxxxx
		15:0																	xxxxx
4A20	OC6RS	31:16	OC6RS<31:0>																xxxxx
		15:0																	xxxxx
4C00	OC7CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>		0000
4C10	OC7R	31:16	OC7R<31:0>																xxxxx
		15:0																	xxxxx
4C20	OC7RS	31:16	OC7RS<31:0>																xxxxx
		15:0																	xxxxx
4E00	OC8CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>		0000
4E10	OC8R	31:16	OC8R<31:0>																xxxxx
		15:0																	xxxxx
4E20	OC8RS	31:16	OC8RS<31:0>																xxxxx
		15:0																	xxxxx
5000	OC9CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	—	—	—	—	—	—	—	—	OC32	OCFLT	OCTSEL	OCM<2:0>		0000
5010	OC9R	31:16	OC9R<31:0>																xxxxx
		15:0																	xxxxx
5020	OC9RS	31:16	OC9RS<31:0>																xxxxx
		15:0																	xxxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 16-1: OCxCON: 出力コンペア「x」制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
	ON	—	SIDL	—	—	—	—	—
7:0	U-0	U-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	OC32	OCFLT <sup>(1)</sup>	OCTSEL <sup>(2)</sup>	OCM<2:0>		

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ON:** 出力コンペア モジュール ON ビット  
 1 = 出力コンペア モジュールを有効にする  
 0 = 出力コンペア モジュールを無効にする

bit 14 **未実装:** 「0」として読み出し

bit 13 **SIDL:** アイドル中停止ビット  
 1 = CPU がアイドルモードに移行した時に動作を停止する  
 0 = アイドル中も動作を継続する

bit 12-6 **未実装:** 「0」として読み出し

bit 5 **OC32:** 32 ビット コンペアモード ビット  
 1 = OCxR<31:0> および / または OCxRS<31:0> を使って 32 ビットタイマ源と比較する  
 0 = OCxR<15:0> と OCxRS<15:0> を使って 16 ビットタイマ源と比較する

bit 4 **OCFLT:** PWM 異常条件ステータスビット<sup>(1)</sup>  
 1 = PWM 異常条件が発生した (ハードウェアでのみクリア)  
 0 = PWM 異常条件は発生していない

bit 3 **OCTSEL:** 出力コンペアタイマ選択ビット<sup>(2)</sup>  
 1 = Timery をこの出力コンペア モジュールのクロック源として使う  
 0 = Timerx をこの出力コンペア モジュールのクロック源として使う

bit 2-0 **OCM<2:0>:** 出力コンペアモード選択ビット  
 111 = OCx を PWM モードにし、フォルトピンを有効にする  
 110 = OCx を PWM モードにし、フォルトピンを無効にする  
 101 = OCx ピンを LOW に初期化し、OCx ピンで連続出力パルスを生成する  
 100 = OCx ピンを LOW に初期化し、OCx ピンで単発出力パルスを生成する  
 011 = コンペアイベントで OCx ピンをトグルする  
 010 = OCx ピンを HIGH に初期化し、コンペアイベントで OCx ピンを LOW にする  
 001 = OCx ピンを LOW に初期化し、コンペアイベントで OCx ピンを HIGH にする  
 000 = 出力コンペア モジュールを無効にする (電流は消費し続けます)

- Note 1:** このビットは OCM<2:0> = 111 の場合にのみ使います。他のモードでは、このビットは「0」として読み出されません。  
**2:** Timerx と Timery の選択は表 16-1 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

## 17.0 デッドマンタイマ (DMT)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 9. Watchdog, Deadman, and Power-up Timers』(DS60001114) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

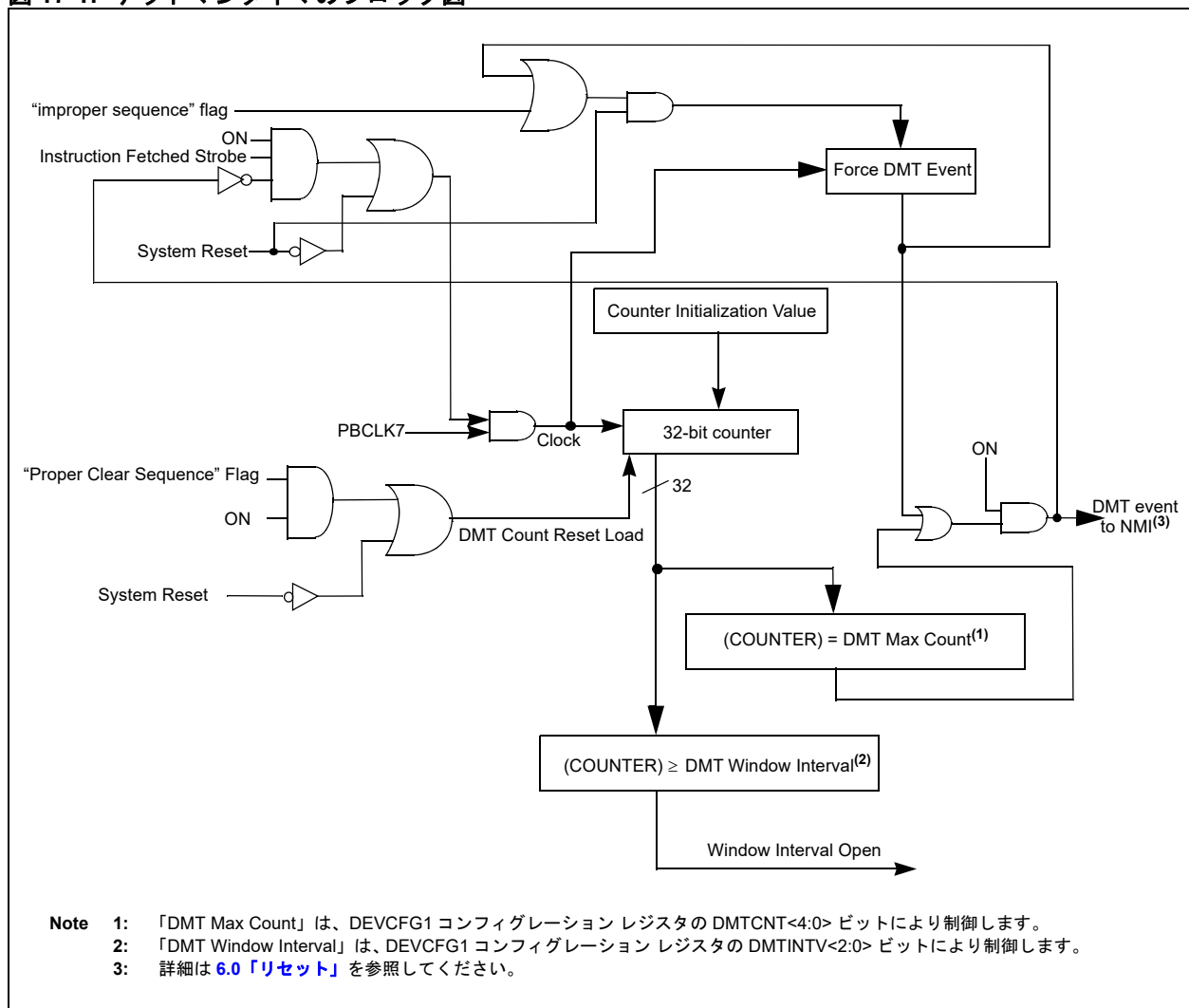
デッドマンタイマ (DMT) の主な機能は、ソフトウェアの誤動作時にプロセッサをリセットする事です。DMT はフリーランの命令フェッチタイマであり、カウントが一致するまで命令フェッチが発生するたびにクロッキングされます。プロセッサのスリープ中は、命令はフェッチされません。

DMT は 1 つの 32 ビットカウンタで構成され、そのタイムアウト カウント値は DEVCFG1 コンフィグレーション レジスタの DMTCNT<4:0> ビットで指定されます。

通常デッドマンタイマは、ソフトウェア機能およびシーケンシングの障害を全て検出する必要があるミッションクリティカルおよびセーフティクリティカルアプリケーションで使います。

図 17-1 に、デッドマンタイマ モジュールのブロック図を示します。

図 17-1: デッドマンタイマのブロック図



## 17.1 デッドマンタイマ制御レジスタ

TABLE 17-1: デッドマンタイマのレジスタマップ

仮アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0A00	DMTCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0A10	DMTPRECLR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	STEP1<7:0>										—	—	—	—	—	—	—
0A20	DMTCLR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	STEP2<7:0>								0000
0A30	DMTSTAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	BAD1	BAD2	DMTEVENT	—	—	—	—	—	WINOPN
0A40	DMTCNT	31:16	COUNTER<31:0>																0000
		15:0	COUNTER<31:0>																0000
0A60	DMTPSCNT	31:16	PSCNT<31:0>																0000
		15:0	PSCNT<31:0>																0000
0A70	DMTPSINTV	31:16	PSINTV<31:0>																0000
		15:0	PSINTV<31:0>																0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 17-1: DMTCON: デッドマンタイム制御レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	ON <sup>(1)</sup>	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装ビット  
 -n = POR 時のビット値 :(「0」、「1」、x = ビットは未知)    P = プログラミング可能ビット    r = 予約済みビット

- bit 31-16    **未実装:** 「0」として読み出し
- bit 15        **ON:** デッドマンタイム モジュール イネーブルビット <sup>(1)</sup>  
               1 = デッドマンタイムを有効にする  
               0 = デッドマンタイムを無効にする
- bit 13-0     **未実装:** 「0」として読み出し

**Note 1:** このビットは、FDMTEN (DEVCFG1<3>) = 0 の場合にのみ効力を有します。

レジスタ 17-2: DMTPRECLR: デッドマンタイム プリスケラ レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	STEP1<7:0>							
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装ビット  
 -n = POR 時のビット値 :(「0」、「1」、x = ビットは未知)    P = プログラミング可能ビット    r = 予約済みビット

- bit 31-16    **未実装:** 「0」として読み出し
- bit 15-8     **STEP1<7:0>:** プリスケラ イネーブルビット  
               01000000 = デッドマン タイマ プリスケラ (STEP1) を有効にする  
               他のビットパターンを書き込むと、BAD1 フラグがセットされます。これらのビットは、DMT リセット イベント発生時にクリアされます。STEP2<7:0> ビットに正しい値が正しい手順で書き込まれた場合にも、STEP1<7:0> はクリアされます。
- bit 7-0       **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 17-3: DMTCLR: デッドマンタイマクリアレジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	STEP2<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装ビット  
 -n = POR 時のビット値 : (「0」、「1」、x = ビットは未知)    P = プログラミング可能ビット    r = 予約済みビット

bit 31-8

**未実装:** 「0」として読み出し

bit 7-0

**STEP2<7:0>:** タイマクリアビット

00001000 = 先に STEP1<7:0> ビットが正しい手順で正しく書き込まれた場合に限り STEP1<7:0>、STEP2<7:0>、デッドマンタイマをクリアする

これらのビットに書き込んだ結果は、DMTCNT ビットを読み出す事で確認できます (カウンタがリセットされている事を確認)。これ以外のビットパターンを書き込んだ場合、BAD2 ビットがセットされ、STEP1<7:0> の値は変更されず、STEP2<7:0> に書き込まれた新しい値がキャプチャされます。これらのビットは、DMT リセットイベント発生時にもクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 17-4: DMTSTAT: デッドマンタイム ステータス レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	R-0, HC BAD1	R-0, HC BAD2	R-0, HC DMTEVENT	R/W-0	R/W-0	R/W-0	R/W-0	R-0 WINOPN

**凡例:**

HC = ハードウェアでクリア

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装ビット

-n = POR 時のビット値 : (「0」、「1」、x = ビットは未知) P = プログラミング可能ビット r = 予約済みビット

bit 31-8 **未実装:** 「0」として読み出し

bit 7 **BAD1:** 不正 STEP1<7:0> 値検出ビット

1 = 不正な STEP1<7:0> 値を検出した

0 = 不正な STEP1<7:0> 値は検出していない

bit 6 **BAD2:** 不正 STEP2<7:0> 値検出ビット

1 = 不正な STEP2<7:0> 値を検出した

0 = 不正な STEP2<7:0> 値は検出していない

bit 5 **DMTEVENT:** デッドマンタイム イベントビット

1 = デッドマンタイム イベントを検出した (カウンタがタイムアウトしたか、カウンタがインクリメントする前に不正な STEP1<7:0> または STEP2<7:0> 値が書き込まれた)

0 = デッドマンタイム イベントは検出していない

bit 4-1 **未実装:** 「0」として読み出し

bit 0 **WINOPN:** デッドマンタイム クリアウィンドウ ビット

1 = デッドマンタイム クリアウィンドウは開いている

0 = デッドマンタイム クリアウィンドウは開いていない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 17-5: DMTCNT: デッドマンタイマ カウントレジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
COUNTER<31:24>								
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
COUNTER<23:16>								
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
COUNTER<15:8>								
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
COUNTER<7:0>								

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装ビット  
 -n = POR 時のビット値 : (「0」、「1」、x = ビットは未知)    P = プログラミング可能ビット    r = 予約済みビット

bit 31-8        **COUNTER<31:0>**: DMT カウンタの現在の値

レジスタ 17-6: DMTPSCNT: ポストステータス DMT カウント設定ステータス レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
PSCNT<31:24>								
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
PSCNT<23:16>								
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
PSCNT<15:8>								
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
PSCNT<7:0>								

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装ビット  
 -n = POR 時のビット値 : (「0」、「1」、x = ビットは未知)    P = プログラミング可能ビット    r = 予約済みビット

bit 31-8        **PSCNT<31:0>**: DMT 命令カウント値設定ステータスビット  
 これは、常に DEVCFG1 コンフィグレーション レジスタの DMTCNT<4:0> ビットの値です。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 17-7: DMTPSINTV: ポストステータス DMT インターバル設定ステータス レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	PSINTV<31:24>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	PSINTV<23:16>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	PSINTV<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	PSINTV<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装ビット

-n = POR 時のビット値 : (「0」、「1」、x = ビットは未知) P = プログラミング可能ビット r = 予約済みビット

bit 31-8 **PSINTV<31:0>**: DMT ウィンドウ インターバル設定ステータスビット

これは、常に DEVCFG1 コンフィグレーション レジスタの DMTINTV<2:0> ビットの値です。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

## 18.0 ウォッチドッグ タイマ (WDT)

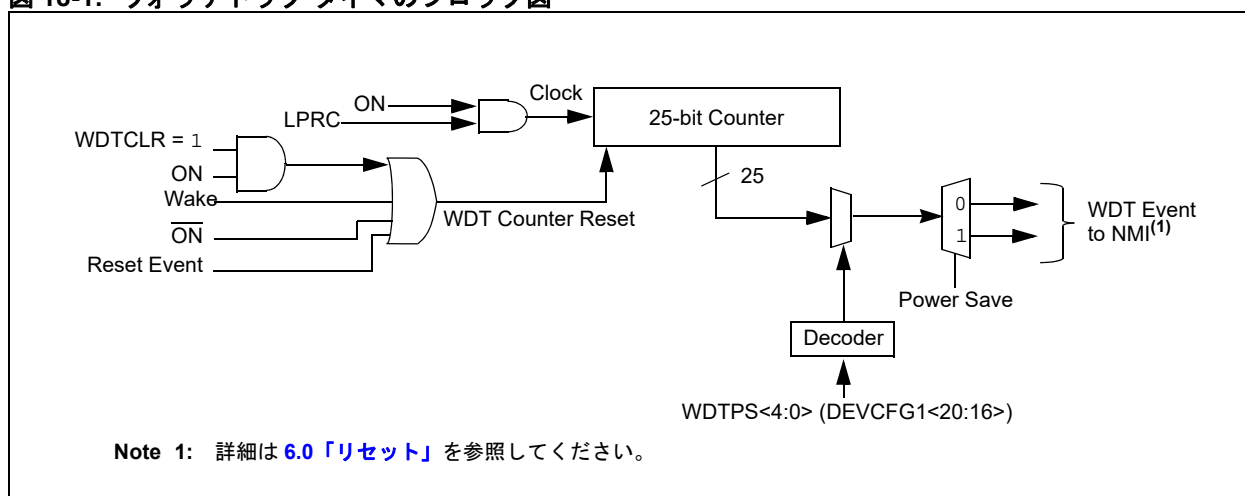
**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 9. Watchdog, Deadman, and Power-up Timers』(DS60001114) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ([www.microchip.com/pic32](http://www.microchip.com/pic32))内の [Documentation > Reference Manual] セクションから入手できます。

ウォッチドッグ タイマ (WDT) を有効にすると、WDT は内部低消費電力オシレータ (LPRC) をクロック源として動作します。WDT がソフトウェアで周期的にクリアされなかった場合にデバイスをリセットする事により、システム ソフトウェアの異常を検出できます。WDT ポストスケラを使う事で、各種の WDT タイムアウト周期を選択できます。WDT を使ってスリープまたはアイドルモードからデバイスを復帰させる事もできます。

WDT モジュールの主な特長は以下の通りです。

- コンフィグレーションまたはソフトウェアによる制御
- タイムアウト期間を設定可能
- スリープまたはアイドルからデバイスを復帰させる事が可能

図 18-1: ウォッチドッグ タイマのブロック図



## 18.1 ウォッチドッグ タイマ制御レジスタ

表 18-1: ウォッチドッグ タイマのレジスタマップ

Virtual Address (BF80_#)	レジスタ名	ビットレンジ	Bit																全リセット
			31、15	30、14	29、13	28、12	27、11	26、10	25、9	24、8	23、7	22、6	21、5	20、4	19、3	18、2	17、1	16、0	
0800	WDTCON <sup>(1)</sup>	31:16	WDTCLRKEY<15:0>																0000
		15:0	ON	—	—	RUNDIV<4:0>				—	—	SLPDIV<4:0>				WDTWINEN	xxxx		

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: このレジスタは対応する CLR、SET、INV レジスタを持ち、それらはレジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 18-1: WDTCON: ウォッチドッグ タイマ制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
WDTCLRKEY<15:8>								
23:16	W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
WDTCLRKEY<7:0>								
15:8	R/W-0	U-0	U-0	R-y	R-y	R-y	R-y	R-y
	ON <sup>(1)</sup>	—	—	RUNDIV<4:0>				
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
	—	—	SLPDIV<4:0>					WDTWINEN

<b>凡例:</b>	y = POR 時にコンフィグレーション ビットからの値に設定
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

- bit 31-16 **WDTCLRKEY<15:0>**: ウォッチドッグ タイマクリア キービット  
ウォッチドッグ タイマをクリアしてタイムアウトを防ぐため、ソフトウェアは 1 回の 16 ビット書き込みによってこれらのビットに値 0x5743 を書き込む必要があります。
- bit 15 **ON**: ウォッチドッグ タイマ イネーブルビット<sup>(1)</sup>  
1 = ウォッチドッグ タイマモジュールを有効にする  
0 = ウォッチドッグ タイマモジュールを無効にする
- bit 14-13 **未実装**: 「0」として読み出し
- bit 12-8 **RUNDIV<4:0>**: 動作モード中ウォッチドッグ タイマ ポストスケアラ値ビット  
これらのビットは、動作モード中に DEVCFG1 レジスタの WDTPS<4:0> コンフィグレーション ビットの値に設定されます。
- bit 7-6 **未実装**: 「0」として読み出し
- bit 5-1 **SLPDIV<4:0>**: スリープ中ウォッチドッグ タイマ ポストスケアラ値ビット  
これらのビットは、スリープ中に DEVCFG4 レジスタの SWDTPS<4:0> コンフィグレーション ビットの値に設定されます。
- bit 0 **WDTWINEN**: ウォッチドッグ タイマ ウィンドウ イネーブルビット  
1 = ウィンドウ式ウォッチドッグ タイマを有効にする  
0 = ウィンドウ式ウォッチドッグ タイマを無効にする

**Note 1:** このビットは、FWDTEN (DEVCFG1<23>) = 0 の場合にのみ効力を有します。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

## 19.0 ディープスリープウォッチドッグタイマ (DSWDT)

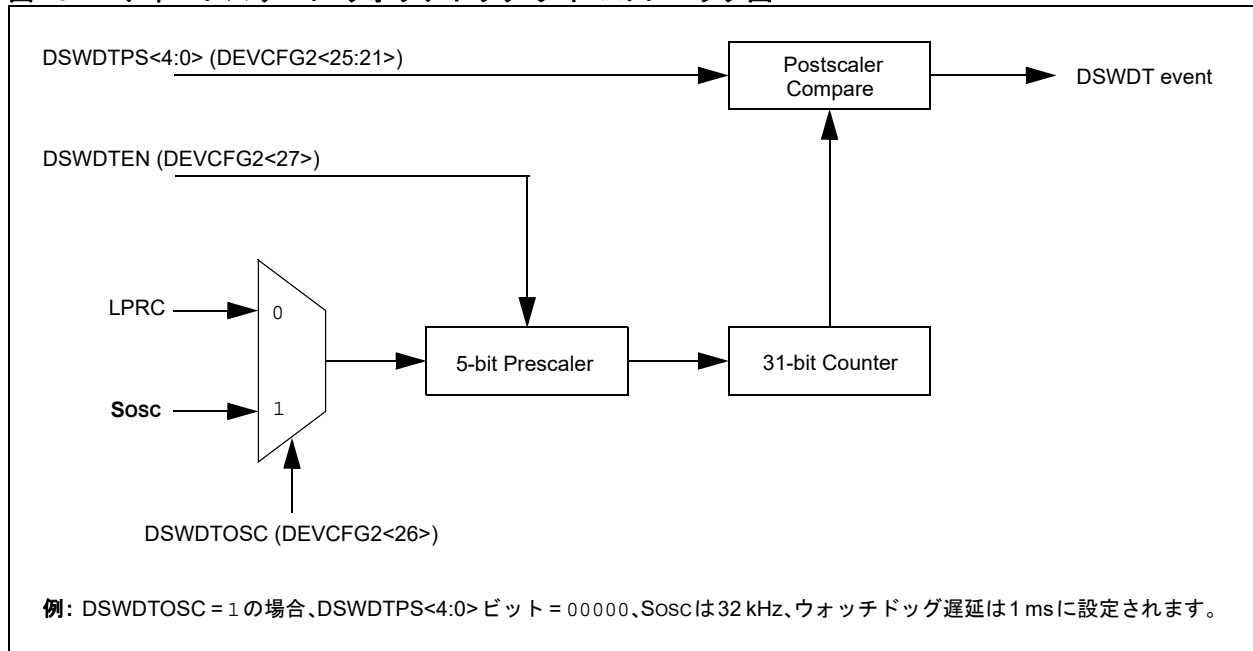
**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 9. Watchdog, Deadman, and Power-up Timers』(DS60001114) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ([www.microchip.com/pic32](http://www.microchip.com/pic32))内の [Documentation > Reference Manual] セクションから入手できます。

ディープスリープウォッチドッグタイマ (DSWDT) は、デバイスのディープスリープモード専用のウォッチドッグタイマです。DSWDT は、バッテリー駆動型アプリケーションと低消費電力動作モードで非常に便利に使用できます。

DSWDT の主な機能は、あらかじめ決められた時間が過ぎた時にデバイスを自動的にディープスリープから復帰させる事です。

DSWDT は、ブート時に DEVCFG2 コンフィグレーションレジスタにより制御されます。このレジスタは POR 後に 1 回だけプログラミングできます。DSWDT は、DEVCFG2 レジスタの DSWDTEN ビットにより有効になり、内部低消費電力 RC (LPRC) クロックまたはセカンダリオシレータ (Sosc) を使って動作します。DSWDT のクロック源は、DEVCFG2 レジスタの DSWDTOSC ビットで選択します。

図 19-1: ディープスリープウォッチドッグタイマのブロック図



# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:



# PIC32MZ グラフィック (DA) ファミリ

## 20.0 リアルタイムクロック/カレンダー (RTCC)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 29. Real-Time Clock and Calendar (RTCC)』(DS60001125) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

RTCC モジュールは、CPU からの介入をほとんど (または全く) 要せずに長期にわたって正確な計時を維持する必要のあるアプリケーション向けに設計されています。最適化された省電力設計により、計時を維持しながらバッテリー寿命を延ばすことができます。

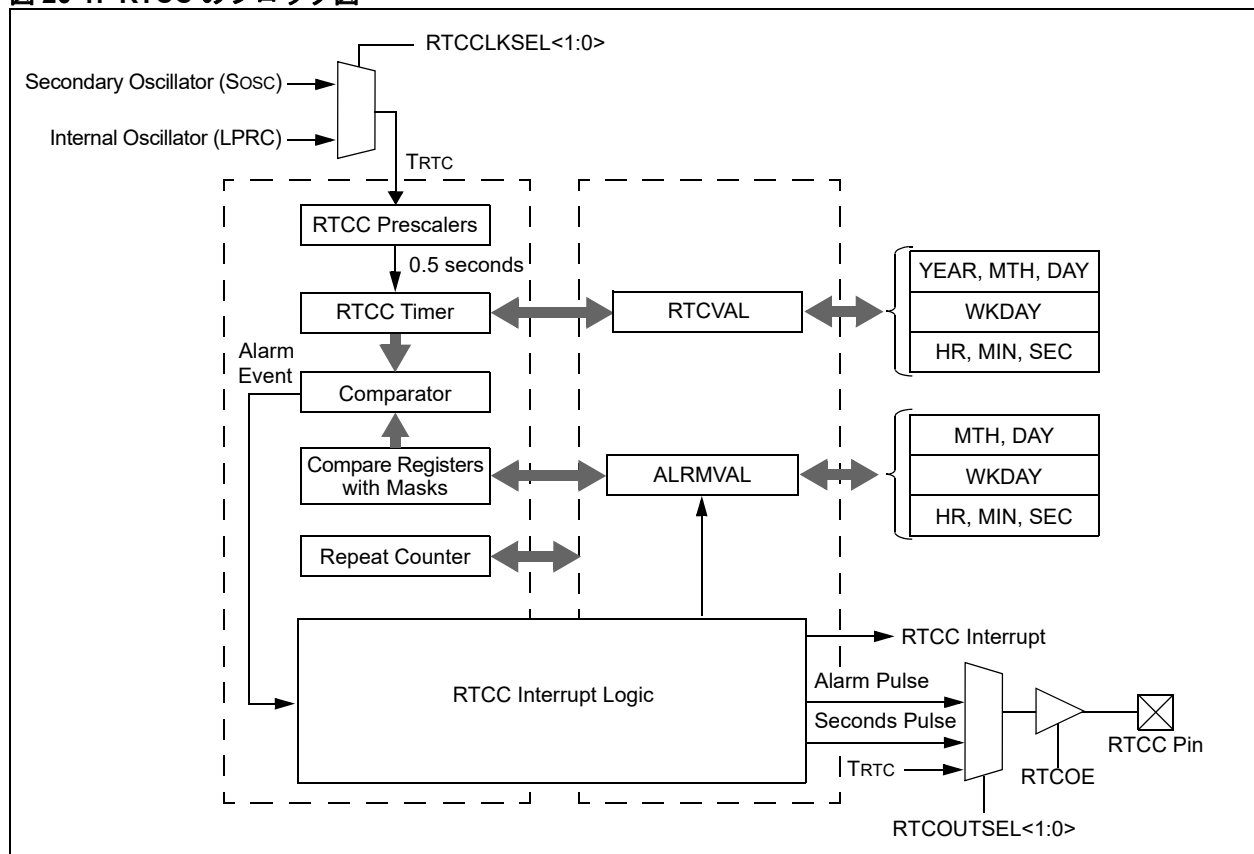
VDDIO ピンの電源が失われても、RTCC モジュールは VBAT モードで動作できます。VBAT ピン (通常はバッテリーに接続) に電力が供給されていれば、RTCC モジュールは動作を継続します。

PMP モジュールの主な特長は以下の通りです。

- 時刻 (時 / 分 / 秒) を出力
- 24 時間フォーマット
- 0.5 秒の分解能
- カレンダーを出力: 曜日、日、月、年
- アラームの設定単位: 1/2 秒、1 秒、10 秒、1 分、10 分、1 時間、1 日、1 週間、1 月、1 年
- カウンタのデクリメントによる一定回数のアラームの繰り返し
- 無限回数のアラームの繰り返し (チャイム)
- 年のレンジ: 2000 ~ 2099
- うるう年補正
- BCD 表記によるファームウェア オーバーヘッドの軽減
- 長期間のバッテリー動作向けに最適化
- 秒未満の同期
- ユーザによるクロック水晶振動子周波数の校正 (周期的な自動調整)
- 校正レンジ:  $\pm 0.66$  秒 / 月の誤差
- 260 ppm まで水晶振動子誤差を校正可能
- 外付け水晶振動子または内部オシレータを使用
- アラームパルス、秒周期クロック、内部クロックを RTCC ピンで出力可能

**Note:** VBAT 動作中は、RTCC ピン機能は使えません。

図 20-1: RTCC のブロック図



## 20.1 RTCC 制御レジスタ

表 20-1: RTCC のレジスタマップ

仮想アドレス (BF8C_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
0000	RTCCON	31:16	—	—	—	—	—	—	CAL<9:0>									0000	
		15:0	ON	—	SIDL	—	—	RTCCLKSEL<1:0>	RTCCOUTSEL<1:0>	RTCCLKON	—	—	RTCWREN	RTCSYNC	HALFSEC	RTCOE	0000		
0010	RTCALRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ALRMEN	CHIME	PIV	ALRMSYNC	AMASK<3:0>				ARPT<7:0>						0000		
0020	RTCTIME	31:16	HR10<3:0>				HR01<3:0>				MIN10<3:0>				MIN01<3:0>				xxxx
		15:0	SEC10<3:0>				SEC01<3:0>				—	—	—	—	—	—	—	—	xx00
0030	RTCDATE	31:16	YEAR10<3:0>				YEAR01<3:0>				MONTH10<3:0>				MONTH01<3:0>				xxxx
		15:0	DAY10<3:0>				DAY01<3:0>				—	—	—	—	WDAY01<3:0>				xx00
0040	ALRMTIME	31:16	HR10<3:0>				HR01<3:0>				MIN10<3:0>				MIN01<3:0>				xxxx
		15:0	SEC10<3:0>				SEC01<3:0>				—	—	—	—	—	—	—	—	xx00
0050	ALRMDATE	31:16	—	—	—	—	—	—	—	MONTH10<3:0>				MONTH01<3:0>				00xx	
		15:0	DAY10<3:0>				DAY01<3:0>				—	—	—	—	WDAY01<3:0>				xx0x

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 20-1: RTCCON: リアルタイム クロック / カレンダー制御レジスタ

ビットレ ンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	CAL<9:8>	
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CAL<7:0>							
15:8	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	ON <sup>(1)</sup>	—	SIDL	—	—	RTCCLKSEL<1:0>		RTC OUTSEL<1> <sup>(2)</sup>
7:0	R/W-0	R-0	U-0	U-0	R/W-0	R-0	R-0	R/W-0
	RTC OUTSEL<0> <sup>(2)</sup>	RTC CLKON	—	—	RTC WREN <sup>(3)</sup>	RTC SYNC	HALFSEC <sup>(4)</sup>	RTCOE

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値                              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-26 **未実装:** 「0」として読み出し

bit 25-16 **CAL<9:0>:** リアルタイム クロック ドリフト校正ビット (符号付き 10 ビット整数値を格納)

0111111111 = 正の最大調整値 (毎分 511 個のリアルタイム クロックパルスを追加する)

⋮

0000000001 = 正の最小調整値 (毎分 1 個のリアルタイム クロックパルスを追加する)

0000000000 = 調整しない

1111111111 = 負の最小調整値 (毎分 1 個のリアルタイム クロックパルスを間引く)

⋮

1000000000 = 負の最大調整値 (毎分 512 個のリアルタイム クロックパルスを間引く)

bit 15 **ON:** RTCC On ビット <sup>(1)</sup>

1 = RTCC モジュールを有効にする

0 = RTCC モジュールを無効にする

bit 14 **未実装:** 「0」として読み出し

bit 13 **SIDL:** アイドル中停止ビット

1 = CPU がアイドルに移行した時に RTCC 動作を無効にする

0 = CPU がアイドルに移行しても RTCC 動作を継続する

bit 12-11 **未実装:** 「0」として読み出し

**Note 1:** ON ビットは RTCWREN = 1 の場合にのみ書き込み可能です。

**2:** 出力を有効にするには RTCOE ビット (RTCCON<0>) を「1」にセットする必要があります。

**3:** RTCWREN ビットをセットするには、書き込みシーケンスを有効にする必要があります。

**4:** このビットは読み出し専用です。このビットは秒ビットフィールド (RTCTIME<14:8>) への書き込み時に「0」にクリアされます。

**Note:** このレジスタはパワーオンリセット (POR) 時のみリセットされます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 20-1: RTCCON: リアルタイム クロック / カレンダー制御レジスタ (続き)

- bit 10-9 **RTCCLKSEL<1:0>**: RTCC クロック選択ビット  
RTCC 内のクロック プリスケアラを正しくリセットするため、これらのビットに新しい値を書き込む時は、秒値レジスタにも書き込む必要があります。  
11 = 予約済み  
10 = 予約済み  
01 = RTCC は外部 32.768 kHz セカンダリ オシレータ (SOSC) を使う  
00 = RTCC は内部 32 kHz オシレータ (LPRC) を使う
- bit 8-7 **RTCOUTSEL<1:0>**: RTCC 出力データ選択ビット (2)  
11 = 予約済み  
10 = RTCC クロックを RTCC ピンで出力する  
01 = 秒周期クロックを RTCC ピンで出力する  
00 = アラーム割り込みがトリガされた時にアラームパルスを RTCC ピンで出力する
- bit 6 **RTCCLKON**: RTCC クロック イネーブル ステータスビット  
1 = RTCC クロックは動作している  
0 = RTCC クロックは動作していない
- bit 5-4 **未実装**: 「0」として読み出し
- bit 3 **RTCWREN**: リアルタイム クロック値レジスタ書き込みイネーブルビット (3)  
1 = ユーザによるリアルタイム クロック値レジスタの書き込みを許可する  
0 = ユーザによるリアルタイム クロック値レジスタの書き込みを禁止する
- bit 2 **RTCSYNC**: リアルタイム クロック値レジスタ読み出しイネーブルビット  
1 = リアルタイム クロック値レジスタ読み出し中の値の変更を許可する  
ロールオーバー リップルにより、無効なデータを読み出す可能性があります。レジスタを 2 回読み出して値が同じであれば、そのデータは有効であると見なせます。  
0 = リアルタイム クロック値レジスタはロールオーバー リップルを気にせずに読み出せる
- bit 1 **HALFSEC**: 秒の前半 / 後半ステータスビット (4)  
1 = 1 秒の後半  
0 = 1 秒の前半
- bit 0 **RTCOE**: RTCC 出力イネーブルビット  
1 = RTCC 出力を有効にする  
0 = RTCC 出力を有効にしない

- Note 1:** ON ビットは RTCWREN = 1 の場合にのみ書き込み可能です。
- 2:** 出力を有効にするには RTCOE ビット (RTCCON<0>) を「1」にセットする必要があります。
- 3:** RTCWREN ビットをセットするには、書き込みシーケンスを有効にする必要があります。
- 4:** このビットは読み出し専用です。このビットは秒ビットフィールド (RTCTIME<14:8>) への書き込み時に「0」にクリアされます。

**Note:** このレジスタはパワーオン リセット (POR) 時にのみリセットされます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 20-2: RTCALRM: リアルタイム クロックアラーム制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
	ALRMEN <sup>(1,2)</sup>	CHIME <sup>(2)</sup>	PIV <sup>(2)</sup>	ALRMSYNC	AMASK<3:0> <sup>(2)</sup>			
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ARPT<7:0> <sup>(2)</sup>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット

U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ALRMEN:** アラーム イネーブルビット (1,2)

- 1 = アラームを有効にする
- 0 = アラームを無効にする

bit 14 **CHIME:** チャイム イネーブルビット (2)

- 1 = チャイムを有効にする (ARPT<7:0> ビットは 0x00 から 0xFF へロールオーバーする)
- 0 = チャイムを無効にする (ARPT<7:0> ビットは 0x00 で停止する)

bit 13 **PIV:** アラームパルス初期値ビット (2)

ALRMEN = 0 の場合、PIV ビットは書き込み可能です。このビットはアラームパルスの初期状態を定義します。  
ALRMEN = 1 の場合、PIV ビットは読み出し専用です。このビットからアラームパルスの状態を読み出せます。

bit 12 **ALRMSYNC:** アラーム同期ビット

- 1 = ARPT<7:0>およびALRMENビットは1/2秒ロールオーバーによって読み出し中に変化する可能性がある。ARPT ビットは、2 回続けて同じ値が読み出せるまで、繰り返し読み出す必要があります。複数のビットが変化する可能性があるため、このような読み出し手順が必要です。
- 0 = ARPT<7:0>およびALRMENビットは、ロールオーバーを気にせずに読み出せる(プリスケアラの1/2秒ロールオーバーまでに 33 リアルタイム クロック以上の時間が残されている)

bit 11-8 **AMASK<3:0>:** アラームマスク コンフィグレーション ビット (2)

- 0000 = 毎 1/2 秒
- 0001 = 毎秒
- 0010 = 毎 10 秒
- 0011 = 毎分
- 0100 = 毎 10 分
- 0101 = 毎時
- 0110 = 毎日
- 0111 = 毎週
- 1000 = 毎月
- 1001 = 毎年 (2 月 29 日に設定した場合は 4 年に一度)
- 1010 = 予約済み
- 1011 = 予約済み
- 11xx = 予約済み

**Note 1:** ARPT<7:0> = 00 かつ CHIME = 0 の場合、ハードウェアはアラームイベントが発生するたびに ALRMEN ビットをクリアします。

**2:** RTCC ON ビット (RTCCON<15>) = 1 かつ ALRMSYNC = 1 の場合、このフィールドに書き込まない事が必要です。

**Note:** このレジスタはパワーオン リセット (POR) 時にのみリセットされます。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 20-2: RTCALRM: リアルタイム クロックアラーム制御レジスタ (続き)

bit 7-0 ARPT<7:0>: アラーム リピートカウンタ値ビット<sup>(2)</sup>

11111111 =アラームは 256 回発生する

.

.

.

00000000 =アラームは一度だけ発生する

このカウンタはアラームイベントのたびにデクリメントします。CHIME = 1 の場合にのみ、カウンタは 0x00 から 0xFF にロールオーバーします。

- Note 1:** ARPT<7:0> = 00 かつ CHIME = 0 の場合、ハードウェアはアラームイベントが発生するたびに ALRMEN ビットをクリアします。
- 2:** RTCC ON ビット (RTCCON<15>) = 1 かつ ALRMSYNC = 1 の場合、このフィールドに書き込まない事が必要です。

**Note:** このレジスタはパワーオン リセット (POR) 時にのみリセットされます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 20-3: RTCTIME: リアルタイムクロック時刻値レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	HR10<3:0>				HR01<3:0>			
23:16	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	MIN10<3:0>				MIN01<3:0>			
15:8	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	SEC10<3:0>				SEC01<3:0>			
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-28 **HR10<3:0>**: 「時」(10の桁) BDC 値ビット (0 ~ 2 を格納)

bit 27-24 **HR01<3:0>**: 「時」(1の桁) BDC 値ビット (0 ~ 9 を格納)

bit 23-20 **MIN10<3:0>**: 「分」(10の桁) BDC 値ビット (0 ~ 5 を格納)

bit 19-16 **MIN01<3:0>**: 「分」(1の桁) BDC 値ビット (0 ~ 9 を格納)

bit 15-12 **SEC10<3:0>**: 「秒」(10の桁) BDC 値ビット (0 ~ 5 を格納)

bit 11-8 **SEC01<3:0>**: 「秒」(1の桁) BDC 値ビット (0 ~ 9 を格納)

bit 7-0 **未実装**: 「0」として読み出し

**Note:** このレジスタは、RTCWREN (RTCCON<3>) = 1 の場合にのみ書き込み可能です。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 20-4: RTCDATE: リアルタイム クロック日付値レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	YEAR10<3:0>				YEAR01<3:0>			
23:16	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	MONTH10<3:0>				MONTH01<3:0>			
15:8	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	DAY10<3:0>				DAY01<3:0>			
7:0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
	—	—	—	—	WDAY01<3:0>			

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-28 **YEAR10<3:0>**: 「年」(10 の桁) BDC 値ビット
- bit 27-24 **YEAR01<3:0>**: 「年」(1 の桁) BDC 値ビット
- bit 23-20 **MONTH10<3:0>**: 「月」(10 の桁) BDC 値ビット (0 または 1 を格納)
- bit 19-16 **MONTH01<3:0>**: 「月」(1 の桁) BDC 値ビット (0 ~ 9 を格納)
- bit 15-12 **DAY10<3:0>**: 「日」(10 の桁) BDC 値ビット (0 ~ 3 を格納)
- bit 11-8 **DAY01<3:0>**: 「日」(1 の桁) BDC 値ビット (0 ~ 9 を格納)
- bit 7-4 **未実装**: 「0」として読み出し
- bit 3-0 **WDAY01<3:0>**: 「曜日」BCD 値ビット (0 ~ 6 を格納)

**Note:** このレジスタは、RTCWREN (RTCCON<3>) = 1 の場合にのみ書き込み可能です。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 20-5: ALRMTIME: アラーム時刻値レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	HR10<3:0>				HR01<3:0>			
23:16	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	MIN10<3:0>				MIN01<3:0>			
15:8	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	SEC10<3:0>				SEC01<3:0>			
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-28 **HR10<3:0>**: 「時」(10の桁) BDC 値ビット (0 ~ 2 を格納)
- bit 27-24 **HR01<3:0>**: 「時」(1の桁) BDC 値ビット (0 ~ 9 を格納)
- bit 23-20 **MIN10<3:0>**: 「分」(10の桁) BDC 値ビット (0 ~ 5 を格納)
- bit 19-16 **MIN01<3:0>**: 「分」(1の桁) BDC 値ビット (0 ~ 9 を格納)
- bit 15-12 **SEC10<3:0>**: 「秒」(10の桁) BDC 値ビット (0 ~ 5 を格納)
- bit 11-8 **SEC01<3:0>**: 「秒」(1の桁) BDC 値ビット (0 ~ 9 を格納)
- bit 7-0 **未実装**: 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 20-6: ALRMDATE: アラーム日付値レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	MONTH10<3:0>				MONTH01<3:0>			
15:8	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	DAY10<1:0>				DAY01<3:0>			
7:0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
	—	—	—	—	WDAY01<3:0>			

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-24 **未実装**: 「0」として読み出し

bit 23-20 **MONTH10<3:0>**: 「月」(10の桁) BDC 値ビット (0 または 1 を格納)

bit 19-16 **MONTH01<3:0>**: 「月」(1の桁) BDC 値ビット (0 ~ 9 を格納)

bit 15-12 **DAY10<3:0>**: 「日」(10の桁) BDC 値ビット (0 ~ 3 を格納)

bit 11-8 **DAY01<3:0>**: 「日」(1の桁) BDC 値ビット (0 ~ 9 を格納)

bit 7-4 **未実装**: 「0」として読み出し

bit 3-0 **WDAY01<3:0>**: 「週」(1の桁) BDC 値ビット (0 ~ 6 を格納)

# PIC32MZ グラフィック (DA) ファミリ

## 21.0 SPI (Serial Peripheral Interface) と I<sup>2</sup>S (Inter-IC Sound)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 23. Serial Peripheral Interface (SPI)』(DS60001106) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

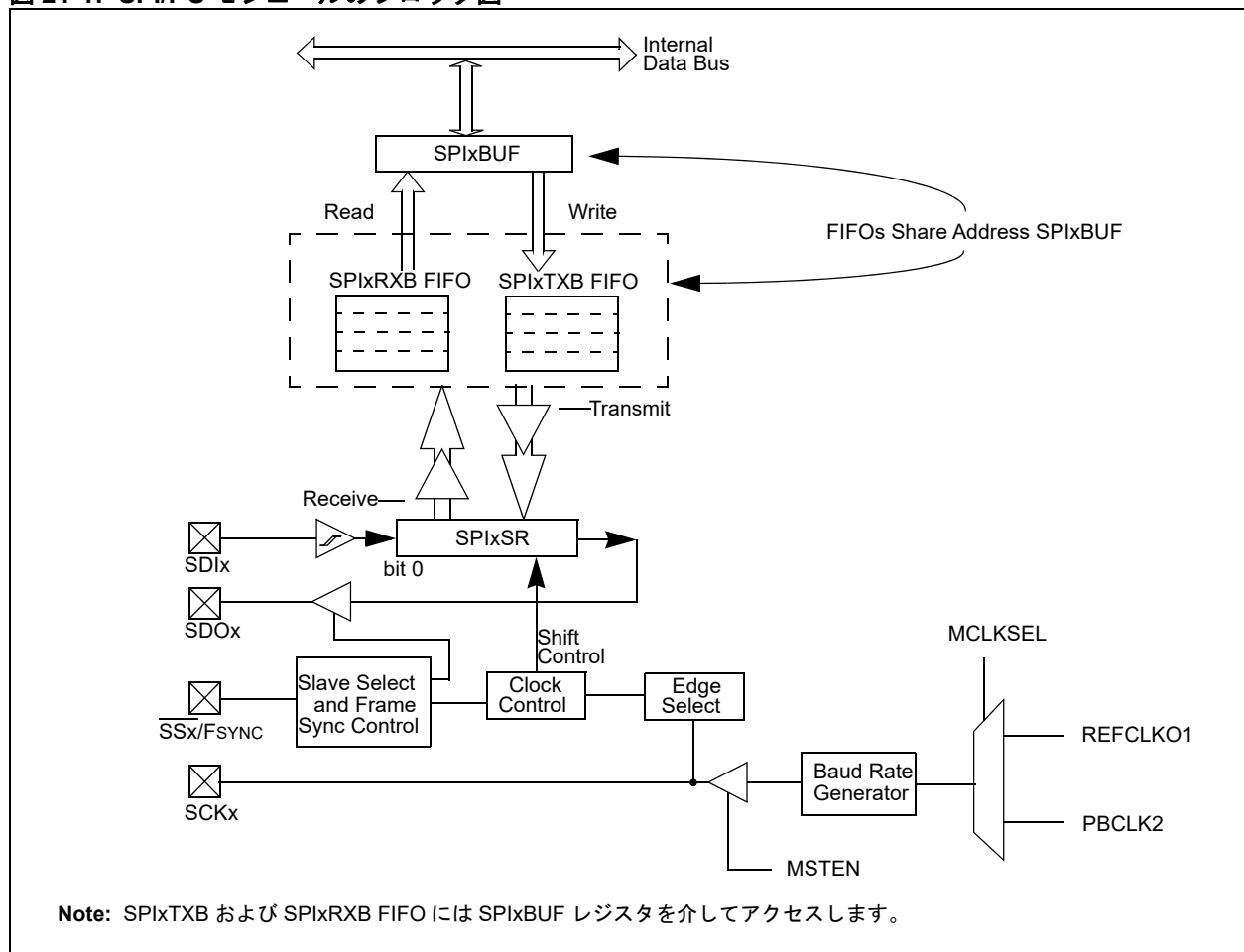
SPI/I<sup>2</sup>S モジュールは外部の周辺デバイス、他のマイクロコントローラ、デジタル オーディオ デバイス等と通信するための同期シリアル インターフェイスです。これらの周辺デバイスには EEPROM、シフトレジスタ、ディスプレイドライバ、アナログ/デジタルコンバータ (ADC) 等が含まれます。

SPI/I<sup>2</sup>S モジュールは、Motorola® SPI および SIOP インターフェイスと互換です。

SPI モジュールの主な特長は以下の通りです。

- マスタモードとスレーブモードをサポート
- 4 種類のクロック フォーマット
- 拡張フレーム化 SPI プロトコルをサポート
- ユーザによるデータ幅(8/16/32ビット)の設定が可能
- 受信と送信で別々の SPI FIFO バッファ
  - FIFO バッファはデータ幅 (32/26/8 ビット) に応じて 4/8/16 段 FIFO として動作
- 8/16/32 ビットデータ転送毎の割り込みイベント(設定可能)
- スリープおよびアイドル中の動作
- 以下のオーディオ コーデックをサポート
  - I<sup>2</sup>S プロトコル
  - 左詰め
  - 右詰め
  - PCM

図 21-1: SPI/I<sup>2</sup>S モジュールのブロック図



## 21.1 SPI 制御レジスタ

表 21-1: SPI1 ~ SPI6 のレジスタマップ

仮想アドレス (BF02_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
1000	SPI1CON	31:16	FRMEN	FRMSYNC	FRMPOL	MSEN	FRMSYPW	FRMCNT<2:0>			MCLKSEL	—	—	—	—	—	SPIFE	ENHBUF	0000
		15:0	ON	—	SIDL	DISSDO	MODE32	MODE16	SMP	CKE	SSEN	CKP	MSTEN	DISSDI	STXISEL<1:0>	SRXISEL<1:0>	0000		
1010	SPI1STAT	31:16	—	—	—	RXBUFELM<4:0>				—	—	—	TXBUFELM<4:0>				0000		
		15:0	—	—	—	FRMERR	SPIBUSY	—	—	SPITUR	SRMT	SPIROV	SPIRBE	—	SPITBE	—	SPITBF	SPIRBF	00A8
1020	SPI1BUF	31:16	DATA<31:0>														0000		
		15:0	DATA<31:0>														0000		
1030	SPI1BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	BRG<12:0>														0000		
1040	SPI1CON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SPI SGNEXT	—	—	FRM ERREN	SPI ROVEN	SPI TUREN	IGNROV	IGNTUR	AUDEN	—	—	—	AUD MONO	—	AUDMOD<1:0>	0C00	
1200	SPI2CON	31:16	FRMEN	FRMSYNC	FRMPOL	MSEN	FRMSYPW	FRMCNT<2:0>			MCLKSEL	—	—	—	—	—	SPIFE	ENHBUF	0000
		15:0	ON	—	SIDL	DISSDO	MODE32	MODE16	SMP	CKE	SSEN	CKP	MSTEN	DISSDI	STXISEL<1:0>	SRXISEL<1:0>	0000		
1210	SPI2STAT	31:16	—	—	—	RXBUFELM<4:0>				—	—	—	TXBUFELM<4:0>				0000		
		15:0	—	—	—	FRMERR	SPIBUSY	—	—	SPITUR	SRMT	SPIROV	SPIRBE	—	SPITBE	—	SPITBF	SPIRBF	0008
1220	SPI2BUF	31:16	DATA<31:0>														0000		
		15:0	DATA<31:0>														0000		
1230	SPI2BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	BRG<8:0>														0000		
1240	SPI2CON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SPI SGNEXT	—	—	FRM ERREN	SPI ROVEN	SPI TUREN	IGNROV	IGNTUR	AUDEN	—	—	—	AUD MONO	—	AUDMOD<1:0>	0000	
1400	SPI3CON	31:16	FRMEN	FRMSYNC	FRMPOL	MSEN	FRMSYPW	FRMCNT<2:0>			MCLKSEL	—	—	—	—	—	SPIFE	ENHBUF	0000
		15:0	ON	—	SIDL	DISSDO	MODE32	MODE16	SMP	CKE	SSEN	CKP	MSTEN	DISSDI	STXISEL<1:0>	SRXISEL<1:0>	0000		
1410	SPI3STAT	31:16	—	—	—	RXBUFELM<4:0>				—	—	—	TXBUFELM<4:0>				0000		
		15:0	—	—	—	FRMERR	SPIBUSY	—	—	SPITUR	SRMT	SPIROV	SPIRBE	—	SPITBE	—	SPITBF	SPIRBF	0008
1420	SPI3BUF	31:16	DATA<31:0>														0000		
		15:0	DATA<31:0>														0000		
1430	SPI3BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	BRG<8:0>														0000		
1440	SPI3CON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SPI SGNEXT	—	—	FRM ERREN	SPI ROVEN	SPI TUREN	IGNROV	IGNTUR	AUDEN	—	—	—	AUD MONO	—	AUDMOD<1:0>	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタ (SPIxBUF を除く) は対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 21-1: SPI1 ~ SPI6 のレジスタマップ (続き)

仮想アドレス (BF82_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
1600	SPI4CON	31:16	FRMEN	FRMSYNC	FRMPOL	MSSSEN	FRMSYPW	FRMCNT<2:0>			MCLKSEL	—	—	—	—	—	SPIFE	ENHBUF	0000
		15:0	ON	—	SIDL	DISSDO	MODE32	MODE16	SMP	CKE	SSEN	CKP	MSTEN	DISSDI	STXISEL<1:0>	SRXISEL<1:0>	0000		
1610	SPI4STAT	31:16	—	—	—	RXBUFELM<4:0>				—	—	—	TXBUFELM<4:0>				0000		
		15:0	—	—	—	FRMERR	SPIBUSY	—	—	SPITUR	SRMT	SPIROV	SPIRBE	—	SPITBE	—	SPITBF	SPIRBF	0008
1620	SPI4BUF	31:16	DATA<31:0>														0000		
		15:0															0000		
1630	SPI4BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
1640	SPI4CON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SPI SGNEXT	—	—	FRM ERREN	SPI ROVEN	SPI TUREN	IGNROV	IGNTUR	AUDEN	—	—	—	AUD MONO	—	AUDMOD<1:0>	0000	
1800	SPI5CON	31:16	FRMEN	FRMSYNC	FRMPOL	MSSSEN	FRMSYPW	FRMCNT<2:0>			MCLKSEL	—	—	—	—	—	SPIFE	ENHBUF	0000
		15:0	ON	—	SIDL	DISSDO	MODE32	MODE16	SMP	CKE	SSEN	CKP	MSTEN	DISSDI	STXISEL<1:0>	SRXISEL<1:0>	0000		
1810	SPI5STAT	31:16	—	—	—	RXBUFELM<4:0>				—	—	—	TXBUFELM<4:0>				0000		
		15:0	—	—	—	FRMERR	SPIBUSY	—	—	SPITUR	SRMT	SPIROV	SPIRBE	—	SPITBE	—	SPITBF	SPIRBF	0008
1820	SPI5BUF	31:16	DATA<31:0>														0000		
		15:0															0000		
1830	SPI5BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
1840	SPI5CON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SPI SGNEXT	—	—	FRM ERREN	SPI ROVEN	SPI TUREN	IGNROV	IGNTUR	AUDEN	—	—	—	AUD MONO	—	AUDMOD<1:0>	0000	
1A00	SPI6CON	31:16	FRMEN	FRMSYNC	FRMPOL	MSSSEN	FRMSYPW	FRMCNT<2:0>			MCLKSEL	—	—	—	—	—	SPIFE	ENHBUF	0000
		15:0	ON	—	SIDL	DISSDO	MODE32	MODE16	SMP	CKE	SSEN	CKP	MSTEN	DISSDI	STXISEL<1:0>	SRXISEL<1:0>	0000		
1A10	SPI6STAT	31:16	—	—	—	RXBUFELM<4:0>				—	—	—	TXBUFELM<4:0>				0000		
		15:0	—	—	—	FRMERR	SPIBUSY	—	—	SPITUR	SRMT	SPIROV	SPIRBE	—	SPITBE	—	SPITBF	SPIRBF	0008
1A20	SPI6BUF	31:16	DATA<31:0>														0000		
		15:0															0000		
1A30	SPI6BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
1A40	SPI6CON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SPI SGNEXT	—	—	FRM ERREN	SPI ROVEN	SPI TUREN	IGNROV	IGNTUR	AUDEN	—	—	—	AUD MONO	—	AUDMOD<1:0>	0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタ (SPIxBUF を除く) は対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 21-1: SPIxCON: SPI 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0 FRMEN	R/W-0 FRMSYNC	R/W-0 FRMPOL	R/W-0 MSSSEN	R/W-0 FRMSYPW	R/W-0	R/W-0	R/W-0 FRMCNT<2:0>
23:16	R/W-0 MCLKSEL <sup>(1)</sup>	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 SPIFE	R/W-0 ENHBUF <sup>(1)</sup>
15:8	R/W-0 ON	U-0 —	R/W-0 SIDL	R/W-0 DISSDO <sup>(4)</sup>	R/W-0 MODE32	R/W-0 MODE16	R/W-0 SMP	R/W-0 CKE <sup>(2)</sup>
7:0	R/W-0 SSEN	R/W-0 CKP <sup>(3)</sup>	R/W-0 MSTEN	R/W-0 DISSDI <sup>(4)</sup>	R/W-0	R/W-0	R/W-0	R/W-0
					STXISEL<1:0>		SRXISEL<1:0>	

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31      **FRMEN:** フレーム化 SPI サポートビット  
 1 = フレーム化 SPI サポートを有効にする (SSx ピンを FSYNC 入力/出力として使う)  
 0 = フレーム化 SPI サポートを無効にする
- bit 30      **FRMSYNC:** フレーム同期パルス SSx ピン方向制御ビット (フレーム化 SPI モード専用)  
 1 = フレーム同期パルス入力 (スレーブモード)  
 0 = フレーム同期パルス出力 (マスターモード)
- bit 29      **FRMPOL:** フレーム同期パルス極性ビット (フレーム化 SPI モード専用)  
 1 = フレームパルスはアクティブ HIGH  
 0 = フレームパルスはアクティブ LOW
- bit 28      **MSSSEN:** マスタモード スレーブ選択イネーブルビット  
 1 = スレーブ選択 SPI サポートを有効にする  
 SS ピンは、マスタモードでの送信中に自動的に駆動されます。極性は FRMPOL ビットで指定します。  
 0 = スレーブ選択 SPI サポートを無効にする
- bit 27      **FRMSYPW:** フレーム同期パルス幅ビット  
 1 = フレーム同期パルスは 1 キャラクタ幅  
 0 = フレーム同期パルスは 1 クロック幅
- bit 26-24      **FRMCNT<2:0>:** フレーム同期パルス カウンタビット  
 このビットは、パルスあたりの送信データキャラクタ数を制御します。このビットは、フレーム化モードでのみ有効です。  
 111 = 予約済み  
 110 = 予約済み  
 101 = 32 データキャラクタごとにフレーム同期パルスを生成する  
 100 = 16 データキャラクタごとにフレーム同期パルスを生成する  
 011 = 8 データキャラクタごとにフレーム同期パルスを生成する  
 010 = 4 データキャラクタごとにフレーム同期パルスを生成する  
 001 = 2 データキャラクタごとにフレーム同期パルスを生成する  
 000 = 1 データキャラクタごとにフレーム同期パルスを生成する
- bit 23      **MCLKSEL:** マスタクロック イネーブルビット <sup>(1)</sup>  
 1 = baud レート ジェネレータは REFCLK01 を使う  
 0 = baud レート ジェネレータは PBCLK2 を使う
- bit 22-18      **未実装:** 「0」として読み出し

- Note 1:** このビットは ON ビット = 0 の場合にのみ書き込み可能です。最大クロック周波数要件は [44.0「電気的特性」](#) を参照してください。
- 2:** フレーム化 SPI モードではこのビットを使いません。フレーム化 SPI モード (FRMEN = 1) の場合、ユーザはこのビットを「0」に設定する必要があります。
- 3:** AUDEN = 1 の場合、CKP ビットの実際の値に関係なく、SPI/I<sup>2</sup>S モジュールは CKP ビットが「1」であるかのように動作します。
- 4:** このビットはレガシー互換性のために提供されます。PIC32MZ DA ファミリデバイスでは、このビットの機能は PPS 機能に取って代わられます (詳細は [12.4「ペリフェラルピンセレクト \(PPS\)」](#) 参照)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 21-1: SPIxCON: SPI 制御レジスタ (続き)

- bit 17 **SPIFE**: フレーム同期パルスエッジ選択ビット (フレーム化 SPI モード専用)  
1 = フレーム同期パルスは先頭ビットクロックと同時に発生する  
0 = フレーム同期パルスは先頭ビットクロックに先行して発生する
- bit 16 **ENHBUF**: 拡張バッファ イネーブルビット<sup>(1)</sup>  
1 = 拡張バッファを有効にする  
0 = 拡張バッファを無効にする
- bit 15 **ON**: SPI/I<sup>2</sup>S モジュール On ビット  
1 = SPI/I<sup>2</sup>S モジュールを有効にする  
0 = SPI/I<sup>2</sup>S モジュールを無効にする
- bit 14 **未実装**: 「0」として読み出し
- bit 13 **SIDL**: アイドル中停止ビット  
1 = CPU がアイドルに移行した時に動作を停止する  
0 = アイドル中も動作を継続する
- bit 12 **DISSDO**: SDOx ピン ディセーブル ビット<sup>(4)</sup>  
1 = SDOx ピンを使わない  
ピンは対応する PORT レジスタにより制御されます。  
0 = モジュールは SDOx ピンを制御する
- bit 11-10 **MODE<32,16>**: 32/16 ビット通信選択ビット  
AUDEN = 1 の場合:
- | MODE32 | MODE16 | 通信  |
|--------|--------|---|
| 1      | 1      | 24 ビットデータ、32 ビット FIFO、32 ビット チャンネル /64 ビット フレーム |
| 1      | 0      | 32 ビットデータ、32 ビット FIFO、32 ビット チャンネル /64 ビット フレーム |
| 0      | 1      | 16 ビットデータ、32 ビット FIFO、16 ビット チャンネル /64 ビット フレーム |
| 0      | 0      | 16 ビットデータ、16 ビット FIFO、16 ビット チャンネル /32 ビット フレーム |
- AUDEN = 0 の場合:
- | MODE32 | MODE16 | 通信     |
|--------|--------|--------|
| 1      | x      | 32 ビット |
| 0      | 1      | 16 ビット |
| 0      | 0      | 8 ビット  |
- bit 9 **SMP**: SPI データ入力サンプル位相ビット  
マスタモード (MSTEN = 1) の場合:  
1 = データ出力時間の最後で入力データをサンプリングする  
0 = データ出力時間の途中で入力データをサンプリングする  
スレーブモード (MSTEN = 0) の場合:  
SPI をスレーブモードで使う場合、SMP ビットの値は無視されます。モジュールは常に SMP = 0 を使います。
- bit 8 **CKE**: SPI クロックエッジ選択ビット<sup>(2)</sup>  
1 = シリアル出力データはクロックがアクティブ状態からアイドル状態へ遷移した時に変化する (CKP ビット参照)  
0 = シリアル出力データはクロックがアイドル状態からアクティブ状態へ遷移した時に変化する (CKP ビット参照)
- bit 7 **SSEN**: スレーブ選択イネーブル (スレーブモード) ビット  
1 = スレーブモードで SSx ピンを使う  
0 = スレーブモードで SSx ピンを使わない (ポート機能がこのピンを制御する)
- bit 6 **CKP**: クロック極性選択ビット<sup>(3)</sup>  
1 = クロックのアイドル状態が HIGH レベル、アクティブ状態が LOW レベル  
0 = クロックのアイドル状態が LOW レベル、アクティブ状態が HIGH レベル

- Note 1:** このビットは ON ビット = 0 の場合にのみ書き込み可能です。最大クロック周波数要件は **44.0「電気的特性」** を参照してください。
- 2:** フレーム化 SPI モードではこのビットを使いません。フレーム化 SPI モード (FRMEN = 1) の場合、ユーザはこのビットを「0」に設定する必要があります。
- 3:** AUDEN = 1 の場合、CKP ビットの実際の値に関係なく、SPI/I<sup>2</sup>S モジュールは CKP ビットが「1」であるかのように動作します。
- 4:** このビットはレガシー互換性のために提供されます。PIC32MZ DA ファミリデバイスでは、このビットの機能は PPS 機能に取って代わられます (詳細は **12.4「ペリフェラルピンセレクト (PPS)」** 参照)。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 21-1: SPIxCON: SPI 制御レジスタ ( 続き )

- bit 5     **MSTEN**: マスタモード イネーブルビット  
          1 = マスタモード  
          0 = スレーブモード
- bit 4     **DISSDI**: SDI ディセーブル ビット (4)  
          1 = SPI モジュールは SDI ピンを使わない ( ポート機能がこのピンを制御する )  
          0 = SPI モジュールが SDI ピンを制御する
- bit 3-2   **STXISEL<1:0>**: SPI 送信バッファ エンプティ割り込みモードビット  
          11 = バッファが非フル (1 つまたは複数の空き位置がある ) 時に割り込みを生成する  
          10 = バッファに 1/2 以上の空きがある時に割り込みを生成する  
          01 = バッファが完全にエンプティの時に割り込みを生成する  
          00 = 最後のデータが SPISR からシフトアウトされて送信が完了した時に割り込みを生成する
- bit 1-0   **SRXISEL<1:0>**: SPI 受信バッファフル割り込みモードビット  
          11 = バッファがフルの時に割り込みを生成する  
          10 = バッファが 1/2 フル以上のデータを格納している時に割り込みを生成する  
          01 = バッファが非エンプティの時に割り込みを生成する  
          00 = 受信バッファ内の最後のワードが読み出された時 ( すなわちバッファがエンプティの時 ) に割り込みを生成する

- Note 1:** このビットは ON ビット = 0 の場合にのみ書き込み可能です。最大クロック周波数要件は [44.0「電氣的特性」](#) を参照してください。
- 2:** フレーム化 SPI モードではこのビットをしません。フレーム化 SPI モード (FRMEN = 1) の場合、ユーザはこのビットを「0」に設定する必要があります。
- 3:** AUDEN = 1 の場合、CKP ビットの実際の値に関係なく、SPI/I<sup>2</sup>S モジュールは CKP ビットが「1」であるかのように動作します。
- 4:** このビットはレガシー互換性のために提供されます。PIC32MZ DA ファミリデバイスでは、このビットの機能は PPS 機能に取って代わられます ( 詳細は [12.4「ペリフェラルピンセレクト \(PPS\)」](#) 参照 )。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 21-2: SPIxCON2: SPI 制御レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SPISGNEXT	—	—	FRMERREN	SPIROVEN	SPITUREN	IGNROV	IGNTUR
7:0	R/W-0	U-0	U-0	U-0	R/W-0	U-0	R/W-0	R/W-0
	AUDEN <sup>(1)</sup>	—	—	—	AUDMONO <sup>(1,2)</sup>	—	AUDMOD<1:0> <sup>(1,2)</sup>	

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15 **SPISGNEXT**: 受信 FIFO 読み出しデータ符号拡張ビット

1 = 受信 FIFO からのデータを符号拡張する

0 = 受信 FIFO からのデータを符号拡張しない

bit 14-13 **未実装**: 「0」として読み出し

bit bit 12 **FRMERREN**: FRMERR 割り込みイベント イネーブルビット

1 = フレームエラーはエラーイベントを生成する

0 = フレームエラーはエラーイベントを生成しない

bit 11 **SPIROVEN**: SPIROV 割り込みイベント イネーブルビット

1 = 受信オーバーフローはエラーイベントを生成する

0 = 受信オーバーフローはエラーイベントを生成しない

bit 10 **SPITUREN**: SPITUR 割り込みイベント イネーブルビット

1 = 送信アンダーランはエラーイベントを生成する

0 = 送信アンダーランはエラーイベントを生成しない

bit 9 **IGNROV**: 受信オーバーフロー無視ビット (オーディオデータ送信用)

1 = ROV を重大なエラーと見なさない (ROV 発生中、FIFO 内のデータは受信データによって上書きされない)

0 = ROV を重大なエラーと見なし、SPI 動作を停止する

bit 8 **IGNTUR**: 送信アンダーラン無視ビット (オーディオデータ送信用)

1 = TUR を重大なエラーと見なさず、SPIxTXB がエンプティではなくなるまでゼロを送信する

0 = TUR を重大なエラーと見なし、SPI 動作を停止する

bit 7 **AUDEN**: オーディオコーデック サポート イネーブルビット<sup>(1)</sup>

1 = オーディオ プロトコルを有効にする

0 = オーディオ プロトコルを無効にする

bit 6-5 **未実装**: 「0」として読み出し

bit 3 **AUDMONO**: オーディオデータ送信フォーマットビット<sup>(1,2)</sup>

1 = オーディオデータはモノラル (各データワードを左右両方のチャンネルで送信する)

0 = オーディオデータはステレオ

bit 2 **未実装**: 「0」として読み出し

bit 1-0 **AUDMOD<1:0>**: オーディオ プロトコル モードビット<sup>(1,2)</sup>

11 = PCM/DSP モード

10 = 右寄せモード

01 = 左寄せモード

00 = I<sup>2</sup>S モード

**Note 1:** このビットは ON ビット = 0 の場合にのみ書き込み可能です。

**2:** このビットは AUDEN = 1 の場合にのみ有効です。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 21-3: SPIxSTAT: SPI ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
	—	—	—	RXBUFELM<4:0>				
23:16	U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
	—	—	—	TXBUFELM<4:0>				
15:8	U-0	U-0	U-0	R/C-0, HS	R-0	U-0	U-0	R-0
	—	—	—	FRMERR	SPIBUSY	—	—	SPITUR
7:0	R-0	R/W-0	R-0	U-0	R-1	U-0	R-0	R-0
	SRMT	SPIROV	SPIRBE	—	SPITBE	—	SPITBF	SPIRBF

<b>凡例:</b>	C = クリア可能ビット	HS = ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装:** 「0」として読み出し

bit 28-24 **RXBUFELM<4:0>:** 受信バッファ エレメント カウントビット (ENHBUF = 1 の場合にのみ有効)

bit 23-21 **未実装:** 「0」として読み出し

bit 20-16 **TXBUFELM<4:0>:** 送信バッファ エレメント カウントビット (ENHBUF = 1 の場合にのみ有効)

bit 15-13 **未実装:** 「0」として読み出し

bit bit 12 **FRMERR:** SPI フレームエラー ステータスビット

1 = フレームエラーを検出した

0 = フレームエラーは検出していない

このビットは FRMEN = 1 の場合にのみ有効です。

bit 11 **SPIBUSY:** SPI 動作ステータスビット

1 = SPI モジュールはトランザクションを処理中 ( ビジー )

0 = SPI モジュールはアイドル中

bit 10-9 **未実装:** 「0」として読み出し

bit 8 **SPITUR:** 送信アンダーラン ビット

1 = 送信バッファでアンダーラン条件が発生した

0 = 送信バッファでアンダーラン条件は発生していない

このビットはフレーム化同期モードでのみ有効です。アンダーラン条件は、モジュールを無効にしてから再度有効にする事によってクリアする必要があります。

bit 7 **SRMT:** シフトレジスタ エンプティビット (ENHBUF = 1 の場合にのみ有効)

1 = SPI モジュール シフトレジスタはエンプティ

0 = SPI モジュール シフトレジスタは非エンプティ

bit 6 **SPIROV:** 受信オーバーフロー フラグビット

1 = 新しいデータを完全に受信したが破棄した (ユーザ ソフトウェアは SPIxBUF レジスタ内の先に受信したデータをまだ読み出していない)

0 = オーバーフローは発生していない

このビットはハードウェアでセットされ、ソフトウェアでのみクリア (= 0) できます。

bit 5 **SPIRBE:** 受信 FIFO エンプティビット (ENHBUF = 1 の場合にのみ有効)

1 = RX FIFO はエンプティ (CRPTR = SWPTR)

0 = RX FIFO は非エンプティ (CRPTR ≠ SWPTR)

bit 4 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 21-3: SPIxSTAT: SPI ステータス レジスタ

bit 3 **SPITBE:** SPI 送信バッファエンプティ ステータスビット

1 = 送信バッファ (SPIxTXB) はエンプティ  
0 = 送信バッファ (SPIxTXB) は非エンプティ

SPI がデータを SPIxTXB から SPIxSR に転送すると、このビットはハードウェアで自動的にセットされます。SPIxBUF にデータを書き込むと (SPIxTXB にデータが転送されると)、このビットはハードウェアで自動的にクリアされます。

bit 2 **未実装:** 「0」として読み出し

bit 1 **SPITBF:** SPI 送信バッファフル ステータスビット

1 = 送信はまだ開始していない (SPITXB はフル)  
0 = 送信バッファは非フル

標準バッファモードの場合:

コアが SPIBUF にデータを書き込むと (データが SPITXB に転送されると)、このビットはハードウェアで自動的にセットされます。SPI モジュールがデータを SPITXB から SPISR に転送すると、このビットはハードウェアで自動的にクリアされます。

拡張バッファモードの場合:

CWPTR + 1 = SRPTR の場合にセットされ、それ以外の場合はクリアされます。

bit 0 **SPIRBF:** SPI 受信バッファフル ステータスビット

1 = 受信バッファ (SPIxRXB) はフル  
0 = 受信バッファ (SPIxRXB) は非フル

標準バッファモードの場合:

SPI モジュールがデータを SPIxSR から SPIxRXB に転送すると、このビットはハードウェアで自動的にセットされます。SPIxBUF を読み出すと (SPIxRXB 内のデータが読み出されると)、このビットはハードウェアで自動的にクリアされます。

拡張バッファモードの場合:

SWPTR + 1 = CRPTR の場合にセットされ、それ以外の場合はクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

## 22.0 SQI (Serial Quad Interface)

**Note:** 本書はPIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 46. Serial Quad Interface (SQI)』(DS60001244) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

SQI モジュールは、シリアルフラッシュメモリを含む各種シリアルデバイスへのアクセスを提供する同期シリアルインターフェイスです。SQI モジュールはシングルレーン (SPI と等価)、デュアルレーン、クワッドレーンモードをサポートします。

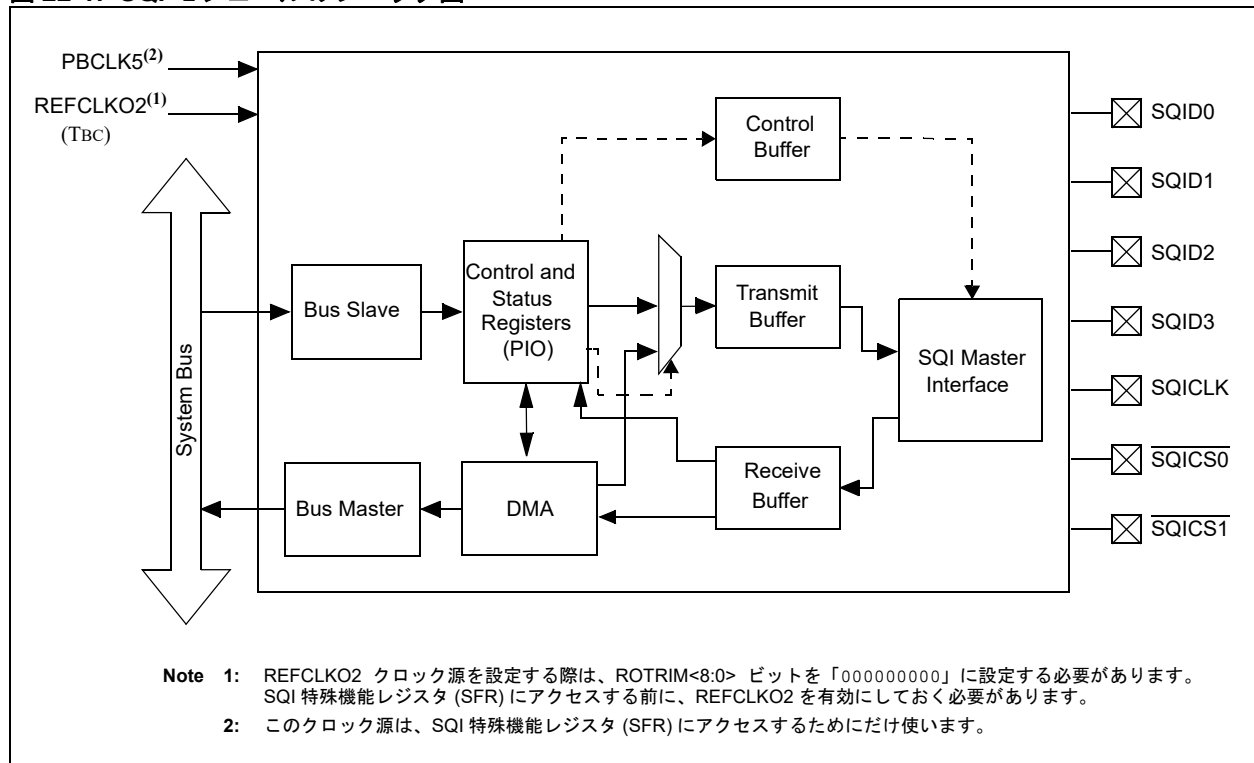
SQI モジュールの主な特長は以下の通りです。

- シングル、デュアル、クワッドレーンモードをサポート
- シングルデータレート (SDR) モードとダブルデータレート (DDR) モードをサポート
- プログラマブルな命令シーケンス

- XIP (eXecute-In-Place) モード
- データ転送:
  - PIO (Programmed I/O) モード
  - バッファディスクリプタDMA
- SPI モード 0 および 3 をサポート
- クロック極性 (CPOL ビット) と クロック位相 (CPHA ビット) を設定可能
- 最大 2 つのチップセレクトをサポート
- 最大 4 バイトのフラッシュアドレスをサポート
- 割り込みしきい値を設定可能
- 32 バイトの送信データバッファ
- 32 バイトの受信データバッファ
- 4 ワードの制御バッファ

**Note:** SQI モジュールを設定すると、外部デバイスは KSEG2 メモリ空間内にマッピングされます (4.0 「メモリ構成」内の図 4-1 と図 4-2 参照)。このメモリにアクセスするには、MMU を有効にし、TLB をセットアップする必要があります。詳細は『PIC32 Family Reference Manual, Section 50. CPU for Devices with MIPS32® microAptiv™ and M-Class Cores』(DS60001192) を参照してください。

図 22-1: SQI モジュールのブロック図



## 22.1 SQI 制御レジスタ

表 22-1: SQI (Serial Quadrature Interface) のレジスタマップ

仮アドレス (BFGE_#)	レジスタ名	ビットレンジ	Bit														全リセット			
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0	
2000	SQI1 XCON1	31:16	—	—	SDRCMD	DDRDATA	DDR DUMMY	DDR MODE	DDR ADDR	DDRCMD	DUMMYBYTES<2:0>			ADDRBYTES<2:0>			READOPCODE<7:6>			0000
		15:0	READOPCODE<5:0>					TYPEDATA<1:0>		TYPEDUMMY<1:0>		TYPEMODE<1:0>		TYPEADDR<1:0>		TYPECMD<1:0>			0000	
2004	SQI1 XCON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	DEVSEL<1:0>		MODEBYTES<1:0>		MODECODE<7:0>							0000		
2008	SQI1CFG	31:16	—	—	—	—	—	CSEN<1:0>		SQIEN	—	DATAEN<1:0>		CON BUFRST	RXBUFST	TXBUFST	RESET	0000		
		15:0	—	—	—	BURSTEN	—	HOLD	WP	—	—	—	LSBF	CPOL	CPHA	MODE<2:0>		0000		
200C	SQI1CON	31:16	—	—	—	—	—	—	—	SCHECK	DDRMODE	DASSERT	DEVSEL<1:0>		LANEMODE<1:0>		CMDINIT<1:0>		0000	
		15:0	TXRXCOUNT<15:0>														0000			
2010	SQI1 CLKCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLKDIV<10:8>		0000	
		15:0	CLKDIV<7:0>										—	—	—	—	—	—	STABLE	EN
2014	SQI1 CMDTHR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	TXCMDTHR<5:0>					RXCMDTHR<5:0>					RXCMDTHR<5:0>					0000		
2018	SQI1 INTTHR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	TXINTTHR<5:0>					RXINTTHR<5:0>					RXINTTHR<5:0>					0000		
201C	SQI1 INTEN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	DMAEIE	PKT COMPIE	BD DONEIE	CON THRIE	CON EMPTYIE	CON FULLIE	RX THRIE	RX FULLIE	RX EMPTYIE	TX THRIE	TX FULLIE	TX EMPTYIE	0000	
2020	SQI1 INTSTAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	DMAEIF	PKT COMPIF	BD DONEIF	CON THRIF	CON EMPTYIF	CON FULLIF	RX THRIF	RX FULLIF	RX EMPTYIF	TX THRIF	TX FULLIF	TX EMPTYIF	0000	
2024	SQI1 TXDATA	31:16	TXDATA<31:16>														0000			
		15:0	TXDATA<15:0>														0000			
2028	SQI1 RXDATA	31:16	RXDATA<31:16>														0000			
		15:0	RXDATA<15:0>														0000			
202C	SQI1 STAT1	31:16	—	—	—	—	—	—	—	—	—	—	—	TXBUFFREE<5:0>				0000		
		15:0	—	—	—	—	—	—	—	—	—	—	RXBUFCNT<5:0>					0000		
2030	SQI1 STAT2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMDSTAT<1:0>		0000	
		15:0	CONAVAIL<3:0>							SDID3	SDID2	SDID1	SDID0	—	—	RXUN	TXOV	00x0		
2034	SQI1 BDCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	START	POLLEN	DMAEN	0000	
2038	SQI1BD CURADD	31:16	BDCURRADDR<31:16>														0000			
		15:0	BDCURRADDR<15:0>														0000			
2040	SQI1BD BASEADD	31:16	BDADDR<31:16>														0000			
		15:0	BDADDR<15:0>														0000			

表 22-1: SQI (Serial Quadrature Interface) のレジスタマップ (続き)

仮アドレス (BF8E#)	レジスタ名	ビットレンジ	Bit															全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
2044	SQI1BD STAT	31:16	—	—	—	—	—	—	—	—	—	—	BDSTATE<3:0>				DMA START	DMAACTV	0000
		15:0	BDCON<15:0>															0000	
2048	SQI1BD POLLCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	POLLCON<15:0>															0000	
204C	SQI1BD TXDSTAT	31:16	TXSTATE<3:0>						—	—	—	TXBUFCNT<5:0>					0000		
		15:0	TXCURBUFLEN<8:0>															0000	
2050	SQI1BD RXDSTAT	31:16	RXSTATE<3:0>						—	—	—	RXBUFCNT<5:0>					0000		
		15:0	RXCURBUFLEN<8:0>															0000	
2054	SQI1THR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	THRES<3:0>															0000	
2058	SQI1INT SIGEN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	DMAEISE	PKT DONEISE	BD DONEISE	CON THRISE	CON EMPTYISE	CON FULLISE	RX THRISE	RX FULLISE	RX EMPTYISE	TX THRISE	TX FULLISE	TX EMPTYISE	0000
205C	SQI1 TAPCON	31:16	DDRCLKINDLY<5:0>						SDRDATAINDLY<3:0>					DDRDATAINDLY<3:0>				0000	
		15:0	SDRCLKINDLY<5:0>						DATAOUTDLY<3:0>					CLKOUTDLY<3:0>				0000	
2060	SQI1 MEMSTAT	31:16	—	—	—	—	—	—	—	—	—	—	—	STATPOS	TYPESTAT<1:0>		STATBYTES<1:0>		0000
		15:0	STATCMD<15:0>															0000	
2064	SQI1 XCON3	31:16	—	—	—	INIT1 SCHECK	INIT1COUNT<1:0>	INIT1TYPE<1:0>		INIT1CMD3<7:0>							0000		
		15:0	INIT1CMD2<7:0>							INIT1CMD1<7:0>							0000		
2068	SQI1 XCON4	31:16	—	—	—	INIT2 SCHECK	INIT2COUNT<1:0>	INIT2TYPE<1:0>		INIT2CMD3<7:0>							0000		
		15:0	INIT2CMD2<7:0>							INIT2CMD1<7:0>							0000		

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 22-1: SQI1XCON1: SQI XIP 制御レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	R/W-0 SDRCMD	R/W-0 DDRDATA	R/W-0 DDRUMMY	R/W-0 DDRMODE	R/W-0 DDRADDR	R/W-0 DDRCMD <sup>(1)</sup>
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DUMMYBYTES<2:0>			ADDRBYTES<2:0>			READOPCODE<7:6>	
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	READOPCODE<5:0>						TYPEDATA<1:0>	
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TYPEDUMMY<1:0>		TYPEMODE<1:0>		TYPEADDR<1:0>		TYPECMD<1:0>	

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-30 **未実装:** 「0」として読み出し

bit 29 **SDRCMD:** SQI コマンド SDR モードビット

1 = SQI コマンドを SDR モードで転送し、SQI データを DDR モードで転送する  
 0 = SQI コマンドも SQI データも DDR モードで転送する

bit 28 **DDRDATA:** SQI データ DDR モードビット

1 = SQI データバイトを DDR モードで転送する  
 0 = SQI データバイトを SDR モードで転送する

bit 27 **DDRUMMY:** SQI ダミー DDR モードビット

1 = SQI ダミーバイトを DDR モードで転送する  
 0 = SQI ダミーバイトを SDR モードで転送する

bit 26 **DDRMODE:** SQI モード DDR モードビット

1 = SQI モードバイトを DDR モードで転送する  
 0 = SQI モードバイトを SDR モードで転送する

bit 25 **DDRADDR:** SQI アドレス DDR モードビット

1 = SQI アドレスバイトを DDR モードで転送する  
 0 = SQI アドレスバイトを SDR モードで転送する

bit 24 **DDRCMD:** SQI コマンド DDR モードビット<sup>(1)</sup>

1 = SQI コマンドバイトを DDR モードで転送する  
 0 = SQI コマンドバイトを SDR モードで転送する

bit 23-21 **DUMMYBYTES<2:0>:** ダミーバイト送信数ビット

111 = アドレスバイトの後で 7 個のダミーバイトを送信する  
 .  
 .  
 011 = アドレスバイトの後で 3 個のダミーバイトを送信する  
 010 = アドレスバイトの後で 2 個のダミーバイトを送信する  
 001 = アドレスバイトの後で 1 個のダミーバイトを送信する  
 000 = アドレスバイトの後でダミーバイトを送信しない

**Note 1:** DDRCMD が「0」の場合、SQI モジュールは SDRCMD ビットの値を無視します。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-1: SQI1XCON1: SQI XIP 制御レジスタ 1 (続き)

bit 20-18 **ADDRBYTES<2:0>**: アドレスバイト数ビット

111 = 予約済み  
.  
.  
.  
101 = 予約済み  
100 = 4 アドレスバイト  
011 = 3 アドレスバイト  
010 = 3 アドレスバイト  
001 = 1 アドレスバイト  
000 = 0 アドレスバイト

bit 17-10 **READOPCODE<7:0>**: 読み出し動作オペコード値ビット

これらのビットは読み出し動作の 8 ビット オペコードを格納します。

bit 9-8 **TYPEDATA<1:0>**: SQI タイプ データ イネーブルビット

ブート コントローラはシングルレーン、デュアルレーン、クワッドレーンのいずれかでデータを受信します。

11 = 予約済み  
10 = クワッドレーン モード データを有効にする  
01 = デュアルレーン モード データを有効にする  
00 = シングルレーン モード データを有効にする

bit 7-6 **TYPEDUMMY<1:0>**: SQI タイプ ダミー イネーブルビット

ブート コントローラはシングルレーン、デュアルレーン、クワッドレーンのいずれかでダミーを送信します。

11 = 予約済み  
10 = クワッドレーン モード ダミーを有効にする  
01 = デュアルレーン モード ダミーを有効にする  
00 = シングルレーン モード ダミーを有効にする

bit 5-4 **TYPEMODE<1:0>**: SQI タイプ モード イネーブルビット

ブート コントローラはシングルレーン、デュアルレーン、クワッドレーンのいずれかでモードを送信します。

11 = 予約済み  
10 = クワッドレーン モードを有効にする  
01 = デュアルレーン モードを有効にする  
00 = シングルレーン モードを有効にする

bit 3-2 **TYPEADDR<1:0>**: SQI タイプ アドレス イネーブルビット

ブート コントローラはシングルレーン、デュアルレーン、クワッドレーンのいずれかでアドレスを送信します。

11 = 予約済み  
10 = クワッドレーン モード アドレスを有効にする  
01 = デュアルレーン モード アドレスを有効にする  
00 = シングルレーン モード アドレスを有効にする

bit 1-0 **TYPECMD<1:0>**: SQI タイプ コマンド イネーブルビット

ブート コントローラはシングルレーン、デュアルレーン、クワッドレーンのいずれかでコマンドを送信します。

11 = 予約済み  
10 = クワッドレーン モード コマンドを有効にする  
01 = デュアルレーン モード コマンドを有効にする  
00 = シングルレーン モード コマンドを有効にする

**Note 1:** DDRCMD が「0」の場合、SQI モジュールは SDRCMD ビットの値を無視します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 22-2: SQI1XCON2: SQI XIP 制御レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	DEVSEL<1:0>		MODEBYTES<1:0>	
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MODECODE<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-12 **未実装:** 「0」として読み出し

bit 11-10 **DEVSEL<1:0>:** デバイス選択ビット

- 11 = 予約済み
- 10 = 予約済み
- 01 = デバイス 1 を選択する
- 00 = デバイス 0 を選択する

bit 9-8 **MODEBYTES<1:0>:** モードバイト サイクルビット

- 11 = 3 サイクル
- 10 = 2 サイクル
- 01 = 1 サイクル
- 00 = 0 サイクル

bit 7-0 **MODECODE<7:0>:** モードコード値ビット

これらのビットは、モードビット向けの 8 ビットコード値を格納します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-3: SQI1CFG: SQI コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	CSEN<1:0>	
23:16	R/W-0	U-0	R/W-0	R/W-0	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
	SQIEN	—	DATAEN<1:0>		CON BUFRST	RX BUFRST	TX BUFRST	RESET
15:8	U-0	r-0	r-0	R/W-0	r-0	R/W-0	R/W-0	U-0
	—	—	—	BURSTEN <sup>(1)</sup>	—	HOLD	WP	—
7:0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	LSBF	CPOL	CPHA	MODE<2:0>		

<b>凡例:</b>	HC=ハードウェアでクリア	r = 予約済み
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31-26 **未実装**: 「0」として読み出し

bit 25-24 **CSEN<1:0>**: チップセレクト出力イネーブルビット

- 11 = チップセレクト 0 とチップセレクト 1 を使う
- 10 = チップセレクト 1 のみ使う
- 01 = チップセレクト 0 のみ使う
- 00 = チップセレクト 0 もチップセレクト 1 も使わない

bit 23 **SQIEN**: SQI イネーブルビット

- 1 = SQI モジュールを有効にする
- 0 = SQI モジュールを無効にする

bit 22 **未実装**: 「0」として読み出し

bit 21-20 **DATAEN<1:0>**: データ出力イネーブルビット

- 11 = 予約済み
- 10 = SQID3 ~ SQID0 データ出力を有効にする
- 01 = SQID1 および SQID0 データ出力を有効にする
- 00 = SQID0 データ出力のみ有効にする

bit 19 **CONBUFRST**: 制御バッファリセット ビット

- 1 = 制御バッファをクリアするためのリセットパルスを生成する
- 0 = リセットパルスを生成しない

bit 18 **RXBUFRST**: 受信バッファリセット ビット

- 1 = 受信バッファをクリアするためのリセットパルスを生成する
- 0 = リセットパルスを生成しない

bit 17 **TXBUFRST**: 送信バッファリセット ビット

- 1 = 送信バッファをクリアするためのリセットパルスを生成する
- 0 = リセットパルスを生成しない

bit 16 **RESET**: ソフトウェア リセット 選択ビット

- このビットは SQI モジュールによって自動的にクリアされます。このリセットパルスは、内部ステートマシンとバッファポインタの全てをリセットします。
- 1 = リセットパルスを生成する
- 0 = リセットパルスを生成しない

bit 15 **未実装**: 「0」として読み出し

bit 14-13 **予約済み**: 常に「0」として書き込み

**Note 1:** このビットは「1」として書き込む必要があります。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 22-3: SQI1CFG: SQI コンフィグレーション レジスタ (続き)

- bit 12 **BURSTEN:** バースト コンフィグレーション ビット<sup>(1)</sup>  
1 = バーストを有効にする  
0 = バーストを無効にする
- bit 11 **予約済み:** 常に「0」として書き込み
- bit 10 **HOLD:** ホールドビット  
このビットは、シングルレーンまたはデュアルレーン モード中に SQID3 ピンを駆動するために使います。この機能は HOLD 入力ピンを備えたデバイスでのみ利用可能です。このビット値の意味は、SQID3 に接続するデバイスによって異なります。
- bit 9 **WP:** 書き込み保護ビット  
このビットは、シングルレーンまたはデュアルレーン モード中に SQID2 ピンを駆動するために使います。この機能は書き込み保護ピンを備えたデバイスでのみ利用可能です。このビット値の意味は、SQID2 ピンに接続するデバイスによって異なります。
- bit 8-6 **未実装:** 「0」として読み出し
- bit 5 **LSBF:** データ形式選択ビット  
1 = LSB から順番に送受信する  
0 = MSB から順番に送受信する
- bit 4 **CPOL:** クロック極性選択ビット  
1 = SQICLK をアクティブ LOW にする (HIGH でアイドル状態)  
0 = SQICLK をアクティブ HIGH にする (LOW でアイドル状態)
- bit 3 **CPHA:** クロック位相選択ビット  
1 = SQICLK のトグルは最初のデータビット期間の開始時に始まる  
0 = SQICLK のトグルは最初のデータビット期間の中央で始まる
- bit 2-0 **MODE<2:0>:** モード選択ビット  
111 = 予約済み  
.  
.  
.  
100 = 予約済み  
011 = XIP モードを選択する  
このモードでは、モジュールは XIP (executing in place) のように動作しますが、タイミングの制御にはレジスタデータを使います。  
010 = DMA モードを選択する  
001 = CPU モードを選択する  
モジュールは PIO モードで CPU によって制御されます。このモードへは、ブートまたは XIP モードの終了時に移行します。  
000 = 予約済み

**Note 1:** このビットは「1」として書き込む必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-4: SQI1CON: SQI 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	r-0	R/W-0
	—	—	—	—	—	—	—	SCHECK <sup>(1)</sup>
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DDRMODE	DASSERT	DEVSEL<1:0>		LANEMODE<1:0>		CMDINIT<1:0>	
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXRXCOUNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXRXCOUNT<7:0>							

<b>凡例:</b>	r = 予約済み
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-26 **未実装:** 「0」として読み出し

bit 25 **予約済み:** 常に「0」として書き込み

bit 24 **SCHECK:** フラッシュステータスチェックビット<sup>(1)</sup>

- 1 = フラッシュのステータスをチェックする
- 0 = フラッシュのステータスをチェックしない

bit 23 **DDRMODE:** DDR モードビット

- 1 = SQI 転送を DDR モードに設定する
- 0 = SQI 転送を SDR モードに設定する

bit 22 **DASSERT:** チップセレクトアサートビット

- 1 = 指定バイト数の送信または受信が終了した後にチップセレクトをネゲートする
- 0 = 指定バイト数の送信または受信が終了した後にチップセレクトをネゲートしない

bit 21-20 **DEVSEL<1:0>:** SQI デバイス選択ビット

- 11 = 予約済み
- 10 = 予約済み
- 01 = デバイス 1 を選択する
- 00 = デバイス 0 を選択する

bit 19-18 **LANEMODE<1:0>:** SQI レーンモード選択ビット

- 11 = 予約済み
- 10 = クワッドレーン モード
- 01 = デュアルレーン モード
- 00 = シングルレーン モード

bit 17-16 **CMDINIT<1:0>:** コマンド開始モード選択ビット

このビットを送信向け (= 01) に設定した場合、コマンドは送信レジスタに対する書き込みまたは TX バッファの内容に基づいて開始されます。

このビットを受信向け (= 10) に設定した場合、コマンドは読み出しレジスタに対する読み出しまたは RX バッファが読み出し可能かどうかに基づいて開始されます。

- 11 = 予約済み
- 10 = 受信
- 01 = 送信
- 00 = アイドル

bit 15-0 **TXRXCOUNT<15:0>:** 送信 / 受信バイト数ビット

これらのビットは、CMDINIT<1:0> の設定に基づいて、送信または受信する総バイト数を指定します。

**Note 1:** このビットを「1」にセットした場合、SQI モジュールは SQI1MEMSTAT レジスタを使ってステータスチェック コマンド処理を制御します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 22-5: SQI1CLKCON: SQI クロック制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	CLKDIV<10:8> <sup>(1)</sup>		
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CLKDIV<7:0> <sup>(1)</sup>							
7:0	U-0	U-0	U-0	U-0	U-0	U-0	R-0	R/W-0
	—	—	—	—	—	—	STABLE	EN

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

bit 31-19 **未実装:** 「0」として読み出し

bit 18-8 **CLKDIV<10:0>:** SQI クロック Tsqi 周波数選択ビット<sup>(1)</sup>

- 10000000000 = ベースクロック Tbc を 2048 分周する
- 01000000000 = ベースクロック Tbc を 1024 分周する
- 00100000000 = ベースクロック Tbc を 512 分周する
- 00010000000 = ベースクロック Tbc を 256 分周する
- 00001000000 = ベースクロック Tbc を 128 分周する
- 00000100000 = ベースクロック Tbc を 64 分周する
- 00000010000 = ベースクロック Tbc を 32 分周する
- 00000001000 = ベースクロック Tbc を 16 分周する
- 00000000100 = ベースクロック Tbc を 8 分周する
- 00000000010 = ベースクロック Tbc を 4 分周する
- 00000000001 = ベースクロック Tbc を 2 分周する
- 00000000000 = ベースクロック Tbc のまま

このビットフィールドを「00000000」に設定すると最大の SQI クロック周波数が得られます。

bit 7-2 **未実装:** 「0」として読み出し

bit 1 **STABLE:** Tsqi クロック安定ビット

EN ビットに「1」を書き込んだ場合、このビットは SQI クロック Tsqi が安定した時点で「1」にセットされます。

- 1 = Tsqi クロックは安定している
- 0 = Tsqi クロックは安定していない

bit 0 **EN:** Tsqi クロック イネーブル選択ビット

クロックの発振が安定した時点で、SQI モジュールは STABLE ビットを「1」にセットします。

- 1 = SQI クロック (Tsqi) を有効にする
- 発振が安定すると、SQI モジュールは STABLE ビットを「1」にセットします

0 = SQI クロック (Tsqi) を無効にする  
 SQI モジュールを低消費電力状態にするには、SQI クロックを停止する必要があります。このクロックを無効にしても SFR にはアクセスできます (SFR は PBCLK5 を使うため)。

**Note 1:** 最大クロック周波数仕様は [44.0「電气的特性」](#) 内の [表 44-41](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-6: SQI1CMDTHR: SQI コマンドしきい値レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	TXCMDTHR<5:0>					
7:0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	RXCMDTHR<5:0> <sup>(1)</sup>					

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-14 **未実装**: 「0」として読み出し

bit 13-8 **TXCMDTHR<5:0>**: 送信コマンドしきい値ビット

送信開始モードでは、SQI モジュールは、送信バッファ内のバイト数がこの送信コマンドしきい値 (TXCMDTHR<5:0>) と等しくなった時に送信動作を実行します。通常、これらのビットは通常のフラッシュ コマンド向けに「1」に設定し、ページ プログラミングを行う場合に大きな値に変更します。16 ビットモードの場合、この値は 2 の倍数である必要があります。

bit 7-6 **未実装**: 「0」として読み出し

bit 5-0 **RXCMDTHR<5:0>**: 受信コマンドしきい値ビット<sup>(1)</sup>

受信開始モードでは、SQI モジュールは、受信コマンドしきい値 (RXCMDTHR<5:0>) に等しい数のバイトを受信バッファ内にフェッチするまで受信動作を実行します。バッファ内にこれらのバイトを格納する空きがない場合、SQI は転送を開始しません。16 ビットモードの場合、この値は 2 の倍数である必要があります。

ソフトウェアが読み出しを実行してバッファカウントが減少すると、ハードウェアはバッファカウントが受信コマンドしきい値 (RXCMDTHR<5:0>) と等しくなるよう受信転送を開始します。後続のワードをバッファにラッチする必要がない場合、ソフトウェアがバッファを読み出す前にコマンド開始モードをアイドルへ変更する必要があります。

ブート /XIP モードの場合、SQI モジュールは受信コマンドしきい値の代わりにシステムバス バーストサイズを使います。

**Note 1:** RXCMDTHR<5:0> は、受信が非アクティブ中 (すなわちアイドルまたは送信モード中) にのみ書き込み可能です。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 22-7: SQI1INTTHR: SQI 割り込みしきい値レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	TXINTTHR<5:0>					
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	RXINTTHR<5:0>					

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-14 **未実装:** 「0」として読み出し

bit 13-8 **TXINTTHR<5:0>:** 送信割り込みしきい値ビット

送信割り込みは、送信バッファ内の空きバイト数が設定されたバイト数よりも大きい場合にセットされます。16 ビットモードの場合、この値は2の倍数である必要があります。

bit 7-6 **未実装:** 「0」として読み出し

bit 5-0 **RXINTTHR<5:0>:** 受信割り込みしきい値ビット

受信割り込みは、受信バッファ内のバイト数が設定されたバイト数以上である場合にセットされます。16 ビットモードの場合、この値は2の倍数である必要があります。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 22-8: SQI1INTEN: SQI 割り込みイネーブル レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 DMAEIE	R/W-0 PKTCOMPIE	R/W-0 BDDONEIE	R/W-0 CONTHRIE
7:0	R/W-0 CONEMPTYIE	R/W-0 CONFULLIE	R/W-0 RXTHRIE	R/W-0 RXFULLIE	R/W-0 RXEMPTYIE	R/W-0 TXTHRIE	R/W-0 TXFULLIE	R/W-0 TXEMPTYIE

<b>凡例:</b>	HS = ハードウェアでセット
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-12 **未実装:** 「0」として読み出し

bit 11 **DMAEIE:** DMA バスエラー割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 10 **PKTCOMPIE:** DMA バッファ ディスクリプタ パケット完了割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 9 **BDDONEIE:** DMA バッファ ディスクリプタ完了割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 8 **CONTHRIE:** 制御バッファしきい値割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 7 **CONEMPTYIE:** 制御バッファ エンプティ割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 6 **CONFULLIE:** 制御バッファフル割り込みイネーブルビット

- このビットは、受信バッファがフルの時の割り込みを有効にします。
- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 5 **RXTHRIE:** 受信バッファしきい値割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 4 **RXFULLIE:** 受信バッファフル割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 3 **RXEMPTYIE:** 受信バッファ エンプティ割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 2 **TXTHRIE:** 送信バッファしきい値割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 1 **TXFULLIE:** 送信バッファフル割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

bit 0 **TXEMPTYIE:** 送信バッファ エンプティ割り込みイネーブルビット

- 1 = 割り込みを有効にする
- 0 = 割り込みを無効にする

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 22-9: SQI1INTSTAT: SQI 割り込みステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS
	—	—	—	—	DMA EIF	PKT COMPIF	BD DONEIF	CON THRIF
7:0	R/W-1, HS	R/W-0, HS	R/W-1, HS	R/W-0, HS	R/W-1, HS	R/W-1, HS	R/W-0, HS	R/W-1, HS
	CON EMPTYIF	CON FULLIF	RXTHRIF <sup>(1)</sup>	RXFULLIF	RX EMPTYIF	TXTHRIF	TXFULLIF	TX EMPTYIF

**凡例:** HS = ハードウェアでセット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-12 **未実装:** 「0」として読み出し

bit 11 **DMAEIF:** DMA バスエラー割り込みフラグビット

1 = DMA バスエラーが発生した  
0 = DMA バスエラーは発生していない

bit 10 **PKTCOMPIF:** DMA バッファ ディスクリプタ プロセッサ パケット完了割り込みフラグビット

1 = DMA BD パケットは完了した  
0 = DMA BD パケットは処理中

bit 9 **BDDONEIF:** DMA バッファ ディスクリプタ完了割り込みフラグビット

1 = DMA BD 処理は完了した  
0 = DMA BD は処理中

bit 8 **CONTHRIF:** 制御バッファしきい値割り込みフラグビット

1 = 制御バッファの空きワード数は THRES の値よりも大きい  
0 = 制御バッファの空きワード数は THRES の値よりも小さい

bit 7 **CONEMPTYIF:** 制御バッファ エンプティ割り込みフラグビット

1 = 制御バッファはエンpty  
0 = 制御バッファは非エンpty

bit 6 **CONFULLIF:** 制御バッファフル割り込みフラグビット

1 = 制御バッファはフル  
0 = 制御バッファは非フル

bit 5 **RXTHRIF:** 受信バッファしきい値割り込みフラグビット<sup>(1)</sup>

1 = 受信バッファの空きワード数は RXINTTHR の値よりも大きい  
0 = 受信バッファの空きワード数は RXINTTHR の値よりも小さい

bit 4 **RXFULLIF:** 受信バッファフル割り込みフラグビット

1 = 受信バッファはフル  
0 = 受信バッファは非フル

bit 3 **RXEMPTYIF:** 受信バッファ エンプティ割り込みフラグビット

1 = 受信バッファはエンpty  
0 = 受信バッファは非エンpty

**Note 1:** ブート /XIP モードの場合、POR 時の受信バッファしきい値の値は 0 です。従って、このビットは POR の直後に「1」にセットされ、システムバス上で読み出し要求を受信するまでセットされたままです。

**Note:** このレジスタ内のビットをクリアするには、対応するビット位値に「1」を書き込みます。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 22-9: SQI1INTSTAT: SQI 割り込みステータス レジスタ ( 続き )

- bit 2     **TXTHRIF:** 送信バッファしきい値割り込みフラグビット  
1 = 送信バッファの空きワード数は TXINTTHR の値よりも大きい  
0 = 送信バッファの空きワード数は TXINTTHR の値よりも小さい
- bit 1     **TXFULLIF:** 送信バッファフル割り込みフラグビット  
1 = 送信バッファはフル  
0 = 送信バッファは非フル
- bit 0     **TXEMPTYIF:** 送信バッファ エンプティ割り込みフラグビット  
1 = 送信バッファはエンプティ  
0 = 送信バッファは非エンプティ

**Note 1:** ブート /XIP モードの場合、POR 時の受信バッファしきい値の値は 0 です。従って、このビットは POR の直後に「1」にセットされ、システムバス上で読み出し要求を受信するまでセットされたままです。

**Note:** このレジスタ内のビットをクリアするには、対応するビット位値に「1」を書き込みます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-10: SQI1TXDATA: SQI 送信データバッファ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXDATA<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXDATA<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXDATA<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXDATA<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

### bit 31-0 TXDATA<31:0>: 送信命令データビット

データは送信前にこのレジスタに転送されます。データ転送が始まる直前に、TXDATA 内のデータがシフトレジスタ (SFDR) に書き込まれます。

データ転送が既に実行中であっても、TXDATA に対して複数回の書き込みが可能です。最大で 8 個の命令までキュー (待ち行列) に入れる事ができます。

## レジスタ 22-11: SQI1RXDATA: SQI 受信データバッファ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXDATA<31:24>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXDATA<23:16>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXDATA<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXDATA<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

### bit 31-0 RXDATA<31:0>: 受信データバッファ ビット

データ転送の最後に、シフトレジスタ内のデータが RXDATA レジスタに書き込まれます。このレジスタはバッファのように機能します。受信バッファの深さは 8 ワードです。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 22-12: SQI1STAT1: SQI ステータス レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
	—	—	TXBUFFFREE<5:0>					
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
	—	—	RXBUFCNT<5:0>					

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

- bit 31-22 **未実装**: 「0」として読み出し
- bit 21-16 **TXBUFFFREE<5:0>**: 送信バッファ空きワード数ビット
- bit 15-6 **未実装**: 「0」として読み出し
- bit 5-0 **RXBUFCNT<5:0>**: バッファ内読み出しデータワード数

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-13: SQI1STAT2: SQI ステータス レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R-0 CMDSTAT<1:0>	R-0
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R-0 CONAVAIL<3:1>	R-0	R-0
7:0	R-0 CONAVAIL<0>	R-0 SQID3	R-0 SQID2	R-0 SQID1	R-0 SQID0	U-0 —	R-0 RXUN	R-0 TXOV

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット

U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-18 **未実装**: 「0」として読み出し

bit 17-16 **CMDSTAT<1:0>**: 現在のコマンドステータスビット

これらのビットは、現在のコマンドステータスを示します。

11 = 予約済み

10 = 受信

01 = 送信

00 = アイドル

bit 15-11 **未実装**: 「0」として読み出し

bit 10-7 **CONAVAIL<3:0>**: 制御バッファ空きワード数ビット

これらのビットは、制御バッファ内の空きワード数を示します。

1000 = 8 ワードの空きがある

0111 = 7 ワードの空きがある

.

.

.

0001 = 1 ワードの空きがある

0000 = 空きはない

bit 6 **SQID3**: SQID3 ステータスビット

1 = SQID3 にデータが存在する

0 = SQID3 にデータは存在しない

bit 5 **SQID2**: SQID2 ステータスビット

1 = SQID2 にデータが存在する

0 = SQID2 にデータは存在しない

bit 4 **SQID1**: SQID1 ステータスビット

1 = SQID1 にデータが存在する

0 = SQID1 にデータは存在しない

bit 3 **SQID0**: SQID0 ステータスビット

1 = SQID0 にデータが存在する

0 = SQID0 にデータは存在しない

bit 2 **未実装**: 「0」として読み出し

bit 1 **RXUN**: 受信バッファ アンダーフロー ステータスビット

1 = 受信バッファでアンダーフローが発生した

0 = 受信バッファでアンダーフローは発生していない

bit 0 **TXOV**: 送信バッファ オーバーフロー ステータスビット

1 = 送信バッファでオーバーフローが発生した

0 = 送信バッファでオーバーフローは発生していない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 22-14: SQI1BDCON: SQI バッファ ディスクリプタ制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	START	POLLEN	DMAEN

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-3 **未実装:** 「0」として読み出し
- bit 2 **START:** バッファ ディスクリプタ プロセッサ (BDP) 起動ビット  
 1 = バッファ ディスクリプタ プロセッサを起動する  
 0 = バッファ ディスクリプタ プロセッサを無効にする
- bit 1 **POLLEN:** バッファ ディスクリプタ ポーリング イネーブルビット  
 1 = BDP ポーリングを有効にする  
 0 = BDP ポーリングを有効にしない
- bit 0 **DMAEN:** DMA イネーブルビット  
 1 = DMA を有効にする  
 0 = DMA を無効にする

レジスタ 22-15: SQI1BDCURADD: 処理中 SQI バッファ ディスクリプタ アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCURRADDR<31:24>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCURRADDR<23:16>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCURRADDR<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCURRADDR<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-0 **BDCURRADDR<31:0>:** 処理中バッファ ディスクリプタ アドレスビット  
 これらのビットは、バッファ ディスクリプタ プロセッサが現在処理中のディスクリプタのアドレスを格納します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-16: SQI1BDBASEADD: SQI バッファ ディスクリプタ ベースアドレス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BDADDR<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BDADDR<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BDADDR<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BDADDR<7:0>								

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **BDADDR<31:0>**: DMA ベースアドレス ビット  
 これらのビットは、ルート バッファ ディスクリプタの物理アドレスを格納します。このレジスタは、DMA がアイドル状態である時に変更する必要があります。

## レジスタ 22-17: SQI1BDSTAT: SQI バッファ ディスクリプタ ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	
—									
23:16	U-0	U-0	R-x	R-x	R-x	R-x	R-x	R-x	
—							BDSTATE<3:0>	DMASTART	DMAACTV
15:8	R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x	
BDCON<15:8>									
7:0	R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x	
BDCON<7:0>									

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-22 **未実装**: 「0」として読み出し

bit 21-18 **BDSTATE<3:0>**: DMA バッファ ディスクリプタ プロセッサ ステータス ビット  
 これらのビットは、バッファ ディスクリプタ プロセッサの現在のステータスを返します。  
 5 = フェッチしたバッファ ディスクリプタは無効  
 4 = ディスクリプタは完了した  
 3 = データ フェイズ  
 2 = バッファ ディスクリプタをロード中  
 1 = ディスクリプタのフェッチ要求を保留中  
 0 = アイドル

bit 17 **DMASTART**: DMA バッファ ディスクリプタ開始ステータスビット  
 1 = DMA は開始した  
 0 = DMA は開始していない

bit 16 **DMAACTV**: DMA バッファ ディスクリプタ アクティブステータスビット  
 1 = バッファ ディスクリプタ プロセッサはアクティブ  
 0 = バッファ ディスクリプタ プロセッサはアイドル状態

bit 15-0 **BDCON<15:0>**: DMA バッファ ディスクリプタ制御ワードビット  
 これらのビットは、現在処理中のバッファ ディスクリプタの制御ワードを格納します。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 22-18: SQI1BDPOLLCON: SQI バッファ ディスクリプタ ポーリング制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	POLLCON<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	POLLCON<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15-0 **POLLCON<15:0>:** バッファ ディスクリプタ プロセッサ ポーリング ステータスビット  
 直前にフェッチされたディスクリプタが無効であった場合、BDP(バッファ ディスクリプタ プロセッサ) はこれらのビットが指定するサイクル数を待機した後に、ディスクリプタ制御ワードを再フェッチします。

レジスタ 22-19: SQI1BDTXDSTAT: SQI バッファ ディスクリプタ DMA 転送ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R-x	R-x	R-x	R-x	U-0
	—	—	—	TXSTATE<3:0>				—
23:16	U-0	U-0	U-0	R-x	R-x	R-x	R-x	R-x
	—	—	—	TXBUFCNT<4:0>				—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
	TXCURBUFLEN<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-29 **未実装:** 「0」として読み出し

bit 28-25 **TXSTATE<3:0>:** 実行中 DMA 送信ステート ステータスビット  
 これらのビットは、実行中の DMA 送信のステートを示します。

bit 24-21 **未実装:** 「0」として読み出し

bit 20-16 **TXBUFCNT<4:0>:** DMA バッファ バイト数ステータスビット  
 これらのビットは、内部バッファの空き情報を提供します。

bit 15-8 **未実装:** 「0」として読み出し

bit 7-0 **TXCURBUFLEN<7:0>:** 実行中 DMA 送信バッファ長ステータスビット  
 これらのビットは、現在実行中の DMA 送信のバッファ長を示します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-20: SQI1BDRXDSTAT: SQI バッファ ディスクリプタ DMA 受信ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R-x	R-x	R-x	R-x	U-0
	—	—	—	RXSTATE<3:0>				—
23:16	U-0	U-0	U-0	R-x	R-x	R-x	R-x	R-x
	—	—	—	RXBUFCNT<4:0>				—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
	RXCURBUFLEN<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28-25 **RXSTATE<3:0>**: 実行中 DMA 受信ステータスビット  
 これらのビットは、実行中の DMA 受信のステータスを示します。

bit 24-21 **未実装**: 「0」として読み出し

bit 20-16 **RXBUFCNT<4:0>**: DMA バッファバイト数ステータスビット  
 これらのビットは、内部バッファの空き情報を提供します。

bit 15-8 **未実装**: 「0」として読み出し

bit 7-0 **RXCURBUFLEN<7:0>**: 実行中 DMA 受信バッファ長ステータスビット  
 これらのビットは、実行中 DMA 受信のバッファ長を示します。

## レジスタ 22-21: SQI1THR: SQI しきい値制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	THRES<3:0>			

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-4 **未実装**: 「0」として読み出し

bit 3-0 **THRES<3:0>**: SQI 制御しきい値ビット  
 SQI 制御しきい値割り込みは、このビットフィールドで指定したサイズの空間が SQI 制御バッファ内で  
 利用可能な場合にアサートされます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-22: SQI1INTSIGEN: SQI 割り込み信号イネーブル レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 DMAEISE	R/W-0 PKT DONEISE	R/W-0 BD DONEISE	R/W-0 CON THRISE
7:0	R/W-0 CON EMPTYISE	R/W-0 CON FULLISE	R/W-0 RX THRISE	R/W-0 RX FULLISE	R/W-0 RX EMPTYISE	R/W-0 TX THRISE	R/W-0 TX FULLISE	R/W-0 TX EMPTYISE

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-12 **未実装**: 「0」として読み出し

bit 11 **DMAEISE**: DMA バスエラー割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 10 **PKTDONEISE**: 受信エラー割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 9 **BDDONEISE**: 送信エラー割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 8 **CONTHRISE**: 制御バッファしきい値割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 7 **CONEMPTYISE**: 制御バッファ エンプティ割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 6 **CONFULLISE**: 制御バッファフル割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 5 **RXTHRISE**: 受信バッファしきい値割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 4 **RXFULLISE**: 受信バッファフル割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 3 **RXEMPTYISE**: 受信バッファ エンプティ割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 2 **TXTHRISE**: 送信バッファしきい値割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 1 **TXFULLISE**: 送信バッファフル割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

bit 0 **TXEMPTYISE**: 送信バッファ エンプティ割り込み信号イネーブルビット

1 = この割り込み信号を有効にする  
0 = この割り込み信号を無効にする

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-23: SQI1TAPCON: SQI TAP 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DDRCLKINDLY<5:0>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SDRDATINDLY<3:0>				DDRDATINDLY<3:0>			
15:8	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SDRCLKINDLY<5:0>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DATAOUTDLY<3:0>				CLKOUTDLY<3:0>			

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-30 **未実装**: 「0」として読み出し

bit 29-24 **DDRCLKINDLY<5:0>**: DDR モード SQI クロック入力遅延ビット

これらのビットは、DDR モードで受信データをサンプリングしている時に、SQI クロック入力に小数分周した遅延を追加するために使います。

111111 = クロック入力に 64 タップを追加する

111110 = クロック入力に 63 タップを追加する

•  
•  
•

000001 = クロック入力に 2 タップを追加する

000000 = クロック入力に 1 タップを追加する

bit 23-20 **SDRDATINDLY<3:0>**: SDR モード SQI データ入力遅延ビット

これらのビットは、SDR モードでデータをフラッシュに書き込んでいる時に、SQI データ出力に小数分周した遅延を追加するために使います。

1111 = データ入力に 16 タップを追加する

1110 = データ入力に 15 タップを追加する

•  
•  
•

0001 = データ入力に 2 タップを追加する

0000 = データ入力に 1 タップを追加する

bit 19-16 **DDRDATINDLY<3:0>**: DDR モード SQI データ出力遅延ビット

これらのビットは、DDR モードでデータをフラッシュに書き込んでいる時に、SQI データ出力に小数分周した遅延を追加するために使います。

1111 = データ入力に 16 タップを追加する

1110 = データ入力に 15 タップを追加する

•  
•  
•

0001 = データ入力に 2 タップを追加する

0000 = データ入力に 1 タップを追加する

bit 15-14 **未実装**: 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 22-23: SQI1TAPCON: SQI TAP 制御レジスタ (続き)

bit 13-8 **SDRCLKINDLY<5:0>**: SDR モード SQI クロック入力遅延ビット

これらのビットは、DDR モードで受信データをサンプリングしている時に、SQI クロック入力に小数分周した遅延を追加するために使います。

111111 = クロック入力に 64 タップを追加する

111110 = クロック入力に 63 タップを追加する

•

•

•

000001 = クロック入力に 2 タップを追加する

000000 = クロック入力に 1 タップを追加する

bit 7-4 **DATAOUTDLY<3:0>**: SQI データ出力遅延ビット

これらのビットは、全ての動作モードでデータをフラッシュに書き込んでいる時に、SQI データ出力に小数分周した遅延を追加するために使います。

1111 = データ出力に 16 タップを追加する

1110 = データ出力に 15 タップを追加する

•

•

•

0001 = データ出力に 2 タップを追加する

0000 = データ出力に 1 タップを追加する

bit 3-0 **CLKOUTDLY<3:0>**: SQI クロック出力遅延ビット

これらのビットは、全ての動作モードでデータをフラッシュに書き込んでいる時に、SQI クロック出力に小数分周した遅延を追加するために使います。

1111 = クロック出力に 16 タップを追加する

1110 = クロック出力に 15 タップを追加する

•

•

•

0001 = クロック出力に 2 タップを追加する

0000 = クロック出力に 1 タップを追加する

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-24: SQI1MEMSTAT: SQI メモリステータス制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	STATPOS	STATTYPE<1:0>	STATBYTES<1:0>		
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	STATCMD<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	STATCMD<7:0>							

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-21 **未実装**: 「0」として読み出し

bit 20 **STATPOS**: フラッシュ内ステータスビット位値ビット

フラッシュ ステータス レジスタ内の BUSY ビットの位置を示します。このビットは、全てのフラッシュタイプ (BUSY ビットが bit 0 のタイプと bit 7 のタイプ) をサポートするために追加されました。

1 = BUSY ビットはステータス レジスタ内の bit 7

0 = BUSY ビットはステータス レジスタ内の bit 0

bit 19-18 **STATTYPE<1:0>**: ステータス コマンド レーンモード ビット

11 = 予約済み

10 = ステータス コマンドおよび読み出しはクワッド レーンモードで実行する

01 = ステータス コマンドおよび読み出しはデュアル レーンモードで実行する

00 = ステータス コマンドおよび読み出しはシングル レーンモードで実行する

bit 17-16 **STATBYTES<1:0>**: ステータスバイト数ビット

11 = 予約済み

10 = ステータス コマンドの長さは 2 バイト

01 = ステータス コマンドの長さは 1 バイト

00 = 予約済み

bit 15-0 **STATCMD<15:0>**: ステータス コマンドビット

これらのビットにはステータス チェックコマンドが書き込まれます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-25: SQI1XCON3: SQI XIP 制御レジスタ 3

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	INIT1SCHECK	INIT1COUNT<1:0>	INIT1TYPE<1:0>		
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	INIT1CMD3<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	INIT1CMD2<7:0>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	INIT1CMD1<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28 **INIT1SCHECK**: フラッシュ初期化 (INIT1) コマンド ステータス チェックビット  
 1 = INIT1 コマンドの実行後にステータスをチェックする  
 0 = ステータスをチェックしない

bit 27-26 **INIT1COUNT<1:0>**: フラッシュ初期化 (INIT1) コマンド数ビット  
 11 = INIT1CMD1、INIT1CMD2、INIT1CMD3 を送信する  
 10 = INIT1CMD1 と INIT1CMD2 を送信し、INIT1CMD3 は保留したまま  
 01 = INIT1CMD1 を送信し、INIT1CMD2 と INIT1CMD3 は保留したまま  
 00 = コマンドを送信しない

bit 25-24 **INIT1TYPE<1:0>**: フラッシュ初期化 (INIT1) コマンド タイプビット  
 11 = 予約済み  
 10 = INIT1 コマンドをクワッド レーンモードで送信する  
 01 = INIT1 コマンドをデュアル レーンモードで送信する  
 00 = INIT1 コマンドをシングル レーンモードで送信する

bit 24-16 **INIT1CMD3<7:0>**: フラッシュ初期化コマンド第 3 ビット  
 フラッシュ初期化のための第 3 のコマンド

bit 15-8 **INIT1CMD2<7:0>**: フラッシュ初期化コマンド第 2 ビット  
 フラッシュ初期化のための第 2 のコマンド

bit 7-0 **INIT1CMD1<7:0>**: フラッシュ初期化コマンド第 1 ビット  
 フラッシュ初期化のための最初のコマンド

**Note:** 一部のフラッシュ デバイスでは、読み書き動作の前に書き込みイネーブル コマンドとセクタ保護解除コマンドが必要です。そのような場合、このレジスタが役立ちます (XIP モード専用)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 22-26: SQI1XCON4: SQI XIP 制御レジスタ 4

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	INIT2SCHECK	INIT2COUNT<1:0>		INIT2TYPE<1:0>	
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	INIT2CMD3<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	INIT2CMD2<7:0>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	INIT2CMD1<7:0>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット

U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28 **INIT2SCHECK**: フラッシュ初期化 (INIT2) コマンド ステータス チェックビット

1 = INIT2 コマンドの実行後にステータスをチェックする

0 = ステータスをチェックしない

bit 27-26 **INIT2COUNT<1:0>**: フラッシュ初期化 (INIT2) コマンド数ビット

11 = INIT2CMD1、INIT2CMD2、INIT2CMD3 を送信する

10 = INIT2CMD1 と INIT2CMD2 を送信し、INIT2CMD3 は保留したまま

01 = INIT2CMD1 を送信し、INIT2CMD2 と INIT2CMD3 は保留したまま

00 = コマンドを送信しない

bit 25-24 **INIT2TYPE<1:0>**: フラッシュ初期化 (INIT2) コマンド タイプビット

11 = 予約済み

10 = INIT2 コマンドをクワッド レーンモードで送信する

01 = INIT2 コマンドをデュアル レーンモードで送信する

00 = INIT2 コマンドをシングル レーンモードで送信する

bit 24-16 **INIT2CMD3<7:0>**: フラッシュ初期化コマンド第 3 ビット

フラッシュ初期化のための第 3 のコマンド

bit 15-8 **INIT2CMD2<7:0>**: フラッシュ初期化コマンド第 2 ビット

フラッシュ初期化のための第 2 のコマンド

bit 7-0 **INIT2CMD1<7:0>**: フラッシュ初期化コマンド第 1 ビット

フラッシュ初期化のための最初のコマンド

**Note:** 一部のフラッシュ デバイスでは、読み書き動作の前に書き込みイネーブル コマンドとセクタ保護解除コマンドが必要です。そのような場合、このレジスタが役立ちます (XIP モード専用)。



# PIC32MZ グラフィック (DA) ファミリ

## 23.0 I<sup>2</sup>C (Inter-Integrated Circuit)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 24. Inter-Integrated Circuit (I<sup>2</sup>C)』(DS60001116) を参照してください。この文書は Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内のセクション [Documentation > Reference Manual] から入手できます。

I<sup>2</sup>C (Inter-Integrated Circuit) モジュールは、I<sup>2</sup>C シリアル通信規格のスレーブモードとマルチマスタモードの両方に完全なハードウェアサポートを提供します。

各 I<sup>2</sup>C モジュールは 2 ピンインターフェイス (クロック用の SCLx ピンとデータ用の SDAx ピン) を備えます。

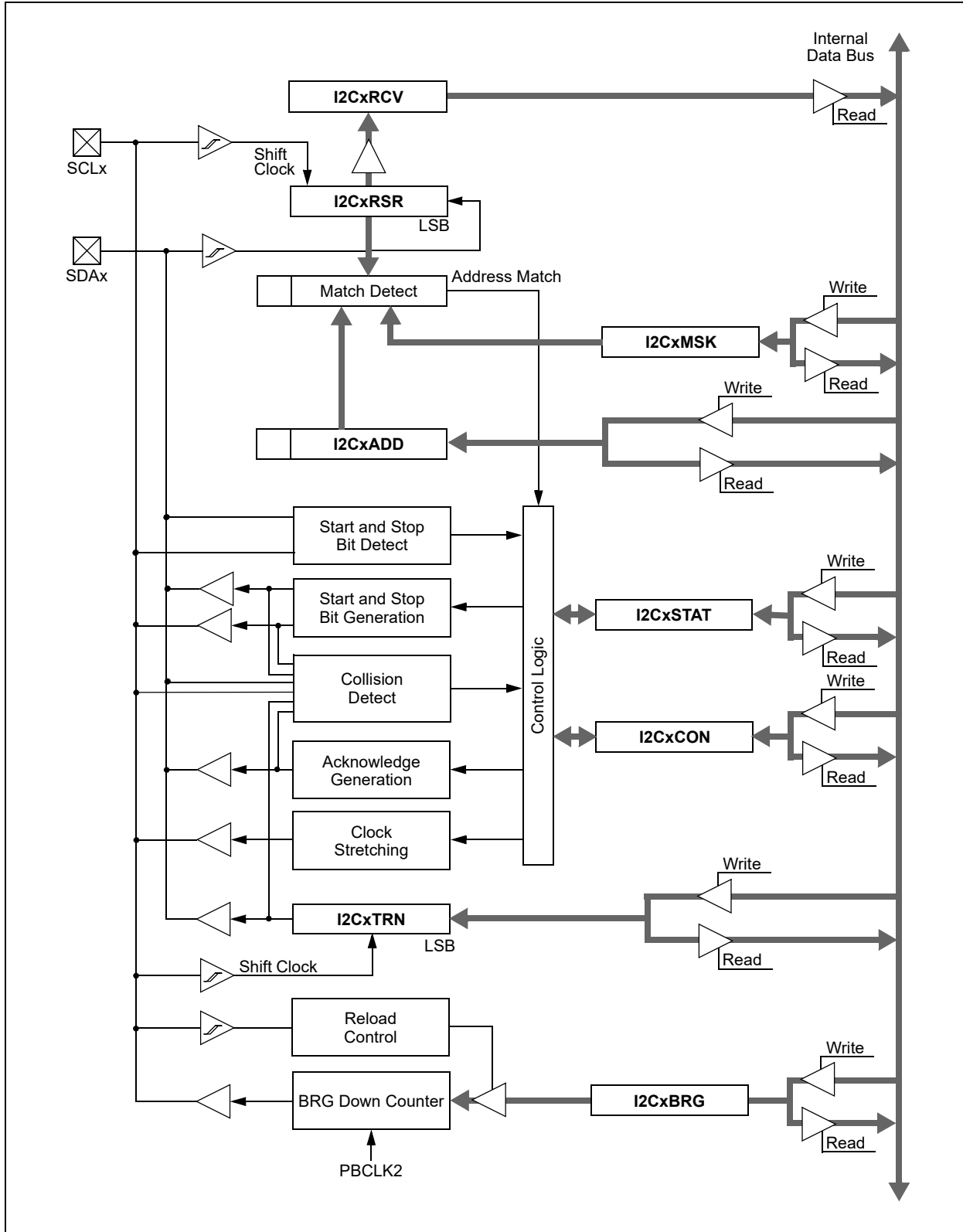
各 I<sup>2</sup>C モジュールは以下の特長を備えます。

- マスタおよびスレーブ動作をサポート
- スレーブモードは 7 ビットおよび 10 ビット アドレス指定をサポート
- マスタモードは 7 ビットおよび 10 ビット アドレス指定をサポート
- I<sup>2</sup>C ポートはマスタとスレーブで双方向に転送可能
- I<sup>2</sup>C ポートのシリアルクロック同期はシリアル転送を一時停止 / 再開するハンドシェイクメカニズムとして使用可能 (SCLREL 転送)
- マルチマスタ動作をサポート (バス衝突を検出/調停)
- アドレスビットのマスキングをサポート
- SMBus をサポート

図 23-1 に、I<sup>2</sup>C モジュールのブロック図を示します。

# PIC32MZ グラフィック (DA) ファミリ

図 23-1: I<sup>2</sup>C のブロック図



23.1 I<sup>2</sup>C 制御レジスタ

表 23-1: I2C1 ~ I2C5 のレジスタマップ

仮想アドレス (BF82_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0000	I2C1CON	31:16	—	—	—	—	—	—	—	—	—	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000
		15:0	ON	—	SIDL	SCLREL	STRICT	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
0010	I2C1STAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ACKSTAT	TRSTAT	ACKTIM	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D/A	P	S	R/W	RBF	TBF	0000
0020	I2C1ADD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ADD<9:0>																0000
0030	I2C1MSK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ADD<9:0>																0000
0040	I2C1BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	I2C1BRG<15:0>																0000
0050	I2C1TRN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	I2C1TXDATA<7:0>																0000
0060	I2C1RCV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	I2C1RXDATA<7:0>																0000
0200	I2C2CON	31:16	—	—	—	—	—	—	—	—	—	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000
		15:0	ON	—	SIDL	SCLREL	STRICT	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
0210	I2C2STAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ACKSTAT	TRSTAT	ACKTIM	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D/A	P	S	R/W	RBF	TBF	0000
0220	I2C2ADD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ADD<9:0>																0000
0230	I2C2MSK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	MSK<9:0>																0000
0240	I2C2BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	I2C2BRG<15:0>																0000
0250	I2C2TRN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	I2C2TXDATA<7:0>																0000
0260	I2C2RCV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	I2C2RXDATA<7:0>																0000
0400	I2C3CON	31:16	—	—	—	—	—	—	—	—	—	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000
		15:0	ON	—	SIDL	SCLREL	STRICT	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
0410	I2C3STAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ACKSTAT	TRSTAT	ACKTIM	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D/A	P	S	R/W	RBF	TBF	0000
0420	I2C3ADD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ADD<9:0>																0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタ (I2CxRCV を除く) は対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

表 23-1: I2C1 ~ I2C5 のレジスタマップ (続き)

仮想アドレス (BF82_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット値
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0430	I2C3MSK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADD<9:0>
0440	I2C3BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	I2C3BRG<15:0>	
0450	I2C3TRN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	I2C3TXDATA<7:0>	
0460	I2C3RCV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	I2C3RXDATA<7:0>	
0600	I2C4CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ON	—	SIDL	SCLREL	STRICT	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
0610	I2C4STAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ACKSTAT	TRSTAT	ACKTIM	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D/A	P	S	R/W	RBF	TBF	0000
0620	I2C4ADD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADD<9:0>	
0630	I2C4MSK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADD<9:0>	
0640	I2C4BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	I2C4BRG<15:0>	
0650	I2C4TRN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	I2C4TXDATA<7:0>	
0660	I2C4RCV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	I2C4RXDATA<7:0>	
0800	I2C5CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ON	—	SIDL	SCLREL	STRICT	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
0810	I2C5STAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ACKSTAT	TRSTAT	ACKTIM	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D/A	P	S	R/W	RBF	TBF	0000
0820	I2C5ADD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADD<9:0>	
0830	I2C5MSK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADD<9:0>	
0840	I2C5BRG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	I2C5BRG<15:0>	
0850	I2C5TRN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	I2C4TXDATA<7:0>	
0860	I2C5RCV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	I2C4RXDATA<7:0>	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の各レジスタ (I2CxRCV を除く) は対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 23-1: I2CxCON: I<sup>2</sup>C 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	R/W-0 PCIE	R/W-0 SCIE	R/W-0 BOEN	R/W-0 SDAHT	R/W-0 SBCDE	R/W-0 AHEN	R/W-0 DHEN
15:8	R/W-0 ON	U-0 —	R/W-0 SIDL	R/W-1, HC SCLREL	R/W-0 STRICT	R/W-0 A10M	R/W-0 DISSLW	R/W-0 SMEN
7:0	R/W-0 GCEN	R/W-0 STREN	R/W-0 ACKDT	R/W-0, HC ACKEN	R/W-0, HC RCEN	R/W-0, HC PEN	R/W-0, HC RSEN	R/W-0, HC SEN

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-23 **未実装:** 「0」として読み出し

bit 22 **PCIE:** STOP 条件割り込みイネーブルビット (I<sup>2</sup>C スレーブモードのみ)

1 = STOP 条件検出時の割り込みを有効にする

0 = STOP 条件検出時の割り込みを無効にする

bit 21 **SCIE:** START 条件割り込みイネーブルビット (I<sup>2</sup>C スレーブモードのみ)

1 = START または RESTART 条件検出時の割り込みを有効にする

0 = START 条件検出時の割り込みを無効にする

bit 20 **BOEN:** バッファ上書きイネーブルビット (I<sup>2</sup>C スレーブモードのみ)

1 = I2COV ビット (I2CxSTAT<6>) の状態を無視する

RBF ビット (I2CxSTAT<2>) = 0 の場合にのみ I2CxRCV を更新し、受信したアドレス / データバイトに対して ACK を生成します。

0 = I2COV ビット (I2CxSTAT<6>) がクリアされている場合にのみ I2CxRCV を更新する

bit 19 **SDAHT:** SDA ホールド時間選択ビット

1 = SCL 立ち下がりエッジ後の SDA の最小ホールド時間を 300 ns にする

0 = SCL 立ち下がりエッジ後の SDA の最小ホールド時間を 100 ns にする

bit 18 **SBCDE:** スレーブモード バスコリジョン検出イネーブルビット (I<sup>2</sup>C スレーブモードのみ)

1 = スレーブ バスコリジョン割り込みを有効にする

0 = スレーブ バスコリジョン割り込みを無効にする

bit 17 **AHEN:** アドレスホールド イネーブルビット (スレーブモード専用)

1 = 一致した受信アドレスバイトの SCL の 8 番目の立ち下がりエッジ後に SCKREL ビットがクリアされ、SCL は LOW を保持する

0 = アドレスホールド機能を無効にする

bit 16 **DHEN:** データホールド イネーブルビット (I<sup>2</sup>C スレーブモードのみ)

1 = 受信データバイトの SCL の 8 番目の立ち下がりエッジ後にスレーブ ハードウェアは SCKREL ビットをクリアし、SCL は LOW を保持する

0 = データホールド機能を無効にする

bit 15 **ON:** I<sup>2</sup>C イネーブルビット

1 = I<sup>2</sup>Cx モジュールを有効にし、SDA および SCL ピンをシリアルポート ピンとして設定する

0 = I<sup>2</sup>Cx モジュールを無効にする (ポート機能が全ての I<sup>2</sup>C ピンを制御する)

bit 14 **未実装:** 「0」として読み出し

bit 13 **SIDL:** アイドル中停止ビット

1 = デバイスがアイドルに移行した時にモジュールの動作を停止する

0 = アイドル中もモジュールの動作を継続する

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 23-1: I2CxCON: I<sup>2</sup>C 制御レジスタ (続き)

- bit 12 **SCLREL:** SCLx 解放制御ビット (I<sup>2</sup>C スレーブとして動作時)  
1 = SCLx クロックを解放する  
0 = SCLx クロックを LOW に保持する (クロック ストレッチング)  
STREN = 1 の場合:  
このビットは R/W です (ソフトウェアは「0」を書き込む事でクロックをストレッチし、「1」を書き込む事でクロックを解放できます)。このビットは、スレーブ送信の開始時およびスレーブ受信の終了時にハードウェアによってクリアされます。  
STREN = 0 の場合:  
このビットは R/S です (ソフトウェアは「1」(クロックを解放)しか書き込めません)。このビットはスレーブ送信の開始時にハードウェアによってクリアされます。
- bit 11 **STRICT:** I<sup>2</sup>C 予約済みアドレス規則遵守イネーブルビット  
1 = 予約済みアドレス規則を有効にする  
デバイスは予約済みアドレス空間内のアドレスにตอบสนองしません。また、それらのアドレスを生成しません。  
0 = I<sup>2</sup>C 予約済みアドレス規則を有効にしない
- bit 10 **A10M:** 10 ビット スレーブアドレス ビット  
1 = I2CxADD は 10 ビット スレーブアドレス  
0 = I2CxADD は 7 ビット スレーブアドレス
- bit 9 **DISSLW:** スルーレート制御ディセーブル ビット  
1 = スルーレート制御を無効にする  
0 = スルーレート制御を有効にする
- bit 8 **SMEN:** SMBus 入力レベルビット  
1 = SMBus 仕様に準拠した I/O ピンしきい値を有効にする  
0 = SMBus 入力しきい値を無効にする
- bit 7 **GCEN:** ジェネラルコール イネーブルビット (I<sup>2</sup>C スレーブとして動作時)  
1 = I2CxRSR にジェネラルコール アドレスを受信した時に割り込みを生成し、モジュールは受信を待機する  
0 = ジェネラルコール アドレスを無効にする
- bit 6 **STREN:** SCLx クロック ストレッチ イネーブルビット (I<sup>2</sup>C スレーブとして動作時)  
このビットは SCLREL ビットと組み合わせて使います。  
1 = ソフトウェアまたは受信クロック ストレッチングを有効にする  
0 = ソフトウェアまたは受信クロック ストレッチングを無効にする
- bit 5 **ACKDT:** 肯定応答 (ACK) データビット (I<sup>2</sup>C マスタとして動作時、マスタ受信中に適用)  
ソフトウェアが肯定応答シーケンスで送信する値を指定します。  
1 = 肯定応答時に NACK を送信する  
0 = 肯定応答時に ACK を送信する
- bit 4 **ACKEN:** 肯定応答シーケンス イネーブルビット (I<sup>2</sup>C マスタとして動作時、マスタ受信中に適用)  
1 = SDAx ピンと SCLx ピンで肯定応答シーケンスを開始し、ACKDT データビットを送信する  
このビットは、マスタ肯定応答シーケンスの終了時にハードウェアによってクリアされます。  
0 = 肯定応答シーケンスは進行中ではない
- bit 3 **RCEN:** 受信イネーブルビット (I<sup>2</sup>C マスタとして動作時)  
1 = I<sup>2</sup>C の受信モードを有効にする  
このビットは、マスタ受信データバイトの 8 番目のビットの終了時にハードウェアによってクリアされます。  
0 = 受信シーケンスは進行中ではない
- bit 2 **PEN:** STOP 条件イネーブルビット (I<sup>2</sup>C マスタとして動作時)  
1 = SDAx ピンと SCLx ピンで STOP 条件を開始する  
このビットは、マスタ STOP シーケンスの終了時にハードウェアによってクリアされます。  
0 = STOP 条件は進行中ではない
- bit 1 **RSEN:** RESTART 条件イネーブルビット (I<sup>2</sup>C マスタとして動作時)  
1 = SDAx ピンと SCLx ピンで RESTART 条件を開始する  
このビットは、マスタ RESTART シーケンスの終了時にハードウェアによってクリアされます。  
0 = RESTART 条件は進行中ではない
- bit 0 **SEN:** START 条件イネーブルビット (I<sup>2</sup>C マスタとして動作時)  
1 = SDAx ピンと SCLx ピンで START 条件を開始する  
このビットは、マスタ START シーケンスの終了時にハードウェアによってクリアされます。  
0 = START 条件は進行中ではない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 23-2: I2CxSTAT: I<sup>2</sup>C ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R-0, HS, HC ACKSTAT	R-0, HS, HC TRSTAT	R/C-0, HS, HC ACKTIM	U-0 —	U-0 —	R/C-0, HS BCL	R-0, HS, HC GCSTAT	R-0, HS, HC ADD10
7:0	R/C-0, HS, SC IWCOL	R/C-0, HS, SC I2COV	R-0, HS, HC D_A	R/C-0, HS, HC P	R/C-0, HS, HC S	R-0, HS, HC R_W	R-0, HS, HC RBF	R-0, HS, HC TBF

**凡例:** HS=ハードウェアでセット HC=ハードウェアでクリア SC=ソフトウェアでクリア  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア C = クリア可能ビット

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ACKSTAT:** 肯定応答 (ACK) ステータスビット (I<sup>2</sup>C マスタとして動作時、マスタ送信動作に適用)

1 = スレーブから NACK を受信した

0 = スレーブから ACK を受信した

このビットは、スレーブ肯定応答の終了時にハードウェアによってセットまたはクリアされます。

bit 14 **TRSTAT:** 送信ステータスビット (I<sup>2</sup>C マスタとして動作時、マスタ送信動作に適用)

1 = マスタ送信 (8 ビット + ACK) が進行中

0 = マスタ送信は進行中ではない

このビットは、マスタ送信開始時にハードウェアによってセットされ、スレーブ肯定応答終了時にハードウェアによってクリアされます。

bit 13 **ACKTIM:** 肯定応答時間ステータスビット (I<sup>2</sup>C スレーブモード専用)

1 = I<sup>2</sup>C バスは肯定応答シーケンスを実行中 (SCL クロックの 8 番目の立ち下がりエッジでセット)

0 = 肯定応答シーケンス中ではない (SCL クロックの 9 番目の立ち上がりエッジでクリア)

bit 12-11 **未実装:** 「0」として読み出し

bit 10 **BCL:** マスタ バスコリジョン検出ビット

1 = マスタ動作中にバスコリジョンを検出した

0 = バスコリジョンは検出していない

このビットは、バスコリジョンの検出時にハードウェアによってセットされます。

bit 9 **GCSTAT:** ジェネラルコール ステータスビット

1 = ジェネラルコール アドレスを受信した

0 = ジェネラルコール アドレスは受信していない

このビットは、アドレスがジェネラルコール アドレスと一致した時にハードウェアによってセットされ、STOP 検出時にハードウェアによってクリアされます。

bit 8 **ADD10:** 10 ビットアドレス ステータスビット

1 = 10 ビットアドレスが一致した

0 = 10 ビットアドレスは一致していない

このビットは、10 ビットアドレスの 2 番目のバイトが一致した時にハードウェアによってセットされ、STOP 検出時にハードウェアによってクリアされます。

bit 7 **IWCOL:** 書き込みコリジョン検出ビット

1 = I<sup>2</sup>C モジュールがビジーのため、I2CxTRN レジスタの書き込みに失敗した

0 = コリジョンは発生していない

このビットは、ビジー中に I2CxTRN の書き込みが発生した時にハードウェアによってセットされます (ソフトウェアでクリア)。

bit 6 **I2COV:** 受信オーバーフロー フラグビット

1 = I2CxRCV レジスタが先に受信したバイトをまだ保持している間に次のバイトを受信した

0 = オーバーフローは発生していない

このビットは、I2CxRSR から I2CxRCV への転送が試行された時にハードウェアによってセットされます (ソフトウェアでクリア)。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 23-2: I2CxSTAT: I<sup>2</sup>C ステータス レジスタ (続き)

- bit 5 **D\_A:** データ / アドレスビット (I<sup>2</sup>C スレーブとして動作時)  
1 = 直近に受信したバイトはデータ  
0 = 直近に受信したバイトはデバイスアドレス  
このビットは、デバイスアドレスが一致した時にハードウェアによってクリアされ、スレーブバイトを受信した時にハードウェアによってセットされます。
- bit 4 **P:** STOP ビット  
1 = STOP ビットを検出した  
0 = STOP ビットは検出していない  
このビットは START、RESTART、STOP の検出時にハードウェアによってセットまたはクリアされます。
- bit 3 **S:** START ビット  
1 = START (または RESTART) ビットを検出した  
0 = START ビットは検出していない  
このビットは START、RESTART、STOP の検出時にハードウェアによってセットまたはクリアされます。
- bit 2 **R\_W:** Read/Write 情報ビット (I<sup>2</sup>C スレーブとして動作時)  
1 = 読み出し - スレーブからマスタへのデータ転送  
0 = 書き込み - マスタからスレーブへのデータ転送  
このビットは、I<sup>2</sup>C デバイスアドレス バイトの受信時にハードウェアによってセットまたはクリアされます。
- bit 1 **RBF:** 受信バッファフル ステータスビット  
1 = 受信は完了した (I2CxRCV はフル)  
0 = 受信は未完了 (I2CxRCV はエンプティ)  
このビットは、I2CxRCV に受信バイトが書き込まれた時にハードウェアによってセットされ、ソフトウェアが I2CxRCV を読み出した時にハードウェアによってクリアされます。
- bit 0 **TBF:** 送信バッファフル ステータスビット  
1 = 送信は未完了 (I2CxTRN はフル)  
0 = 送信は完了した (I2CxTRN はエンプティ)  
このビットは、ソフトウェアが I2CxTRN に書き込んだ時にハードウェアによってセットされ、データ送信の完了時にハードウェアによってクリアされます。



# PIC32MZ グラフィック (DA) ファミリ

## 24.0 UART (Universal Asynchronous Receiver Transmitter)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 21. Universal Asynchronous Receiver Transmitter (UART)』(DS60001107) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

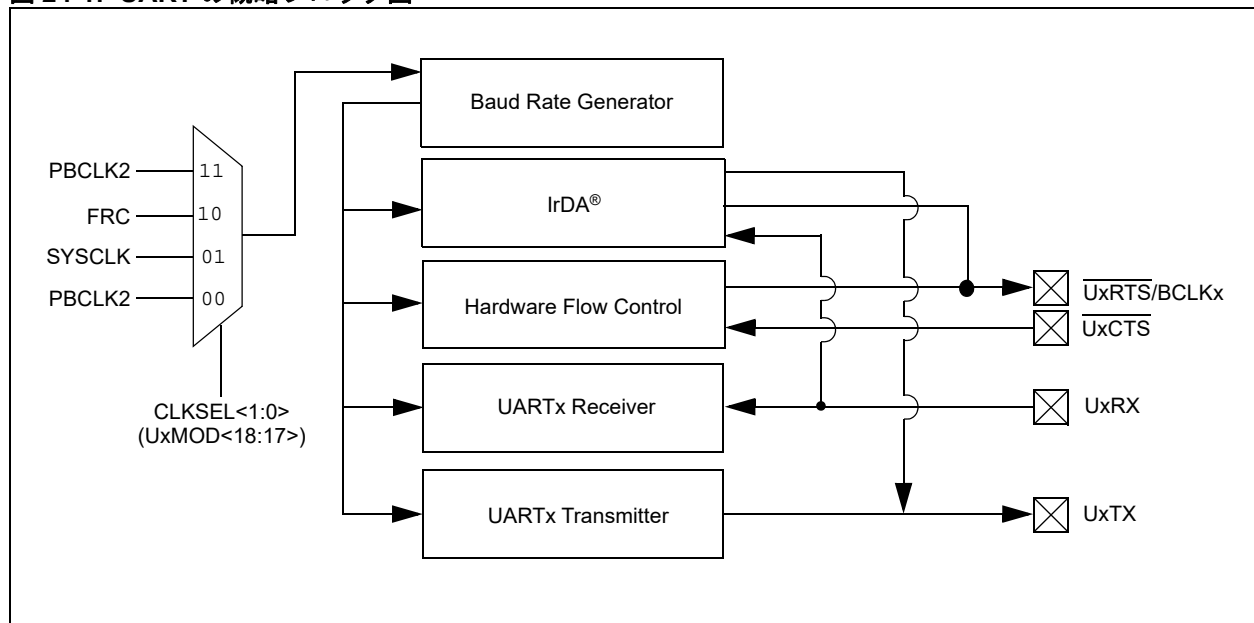
UART モジュールは PIC32MZ DA ファミリデバイスが提供するシリアル I/O モジュールの 1 つです。UART は RS-232、RS-485、LIN、IrDA<sup>®</sup> 等のプロトコルを使って周辺デバイスやパーソナルコンピュータと通信する全二重方式の非同期通信チャンネルです。本モジュールは UxCTS ピンと UxRTS ピンを使ったハードウェアフロー制御オプションもサポートします。また、IrDA エンコーダ / デコーダも備えています。

UART モジュールの主な特長は以下の通りです。

- 全二重の 8 ビットまたは 9 ビットデータ転送
- 偶数パリティ、奇数パリティ、パリティなしを選択可能 (8 ビットデータの場合)
- 1 個また 2 個のストップビット
- ハードウェアによる baud レート自動検出機能
- ハードウェア フロー制御オプション
- 16 ビット プリスケアラ付き baud レート ジェネレータ (BRG) を内蔵
- 100 MHz (PBCLK2) で 76 bps ~ 25 Mbps の baud レートレンジ
- 8 段の先入れ先出し (FIFO) 送信データバッファ
- 8 段の FIFO 受信データバッファ
- パリティ、フレーミング、バッファ オーバーランエラーの検出
- アドレス検出 (第 9 ビット = 1) 時にのみ割り込む事が可能
- 送信割り込みと受信割り込みを別々に生成
- 診断用ループバック モードをサポート
- LIN プロトコルをサポート
- 16x baud クロック出力を備えた IrDA エンコーダ / デコーダにより外部 IrDA エンコーダ / デコーダをサポート
- baud レート自動検出機能をサポート
- スリープ中でもデータを受信可能

図 24-1 に、UART モジュールの概略ブロック図を示します。

図 24-1: UART の概略ブロック図



## 24.1 UART 制御レジスタ

表 24-1: UART1 ~ UART6 のレジスタマップ

仮想アドレス (BF02_#)	レジスタ名	ビットレンジ	Bit														リセット				
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0		
2000	U1MODE <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLKSEL<1:0>	RUNOVF	0000
		15:0	ON	—	SIDL	IREN	RTSMD	—	—	—	—	—	—	—	—	—	—	—	—	PDSEL<1:0>	STSEL
2010	U1STA <sup>(1)</sup>	31:16	MASK<7:0>							ADDR<7:0>							0000				
		15:0	UTXISEL<1:0>		UTXINV	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA	0110		
2020	U1TXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	TX8	Transmit Register							0000		
2030	U1RXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
2040	U1BRG <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	Baud Rate Generator Prescaler														0000				
2200	U2MODE <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	IREN	RTSMD	—	—	—	—	—	—	—	—	—	—	—	—	—	—
2210	U2STA <sup>(1)</sup>	31:16	MASK<7:0>							ADDR<7:0>							0000				
		15:0	UTXISEL<1:0>		UTXINV	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA	0110		
2220	U2TXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	TX8	Transmit Register							0000	
2230	U2RXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
2240	U2BRG <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	Baud Rate Generator Prescaler														0000				
2400	U3MODE <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	SIDL	IREN	RTSMD	—	—	—	—	—	—	—	—	—	—	—	—	—	—
2410	U3STA <sup>(1)</sup>	31:16	MASK<7:0>							ADDR<7:0>							0000				
		15:0	UTXISEL<1:0>		UTXINV	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA	0110		
2420	U3TXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	TX8	Transmit Register							0000	
2430	U3RXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
2440	U3BRG <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	Baud Rate Generator Prescaler														0000				

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: このレジスタは対応する CLR、SET、INV レジスタを持ち、それらはレジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

表 24-1: UART1 ~ UART6 のレジスタマップ (続き)

仮想アドレス (BF82_#)	レジスタ名	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
2600	U4MODE <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	SLPEN	ACTIVE	—	—	—	CLKSEL<1:0>	RUNOVF	0000
		15:0	ON	—	SIDL	IREN	RTSMD	—	—	—	—	UEN<1:0>	WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL<1:0>	STSEL
2610	U4STA <sup>(1)</sup>	31:16	MASK<7:0>							ADDR<7:0>							0000		
		15:0	UTXISEL<1:0>		UTXINV	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA	0110
2620	U4TXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	TX8	Transmit Register							0000
2630	U4RXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	RX8	Receive Register							0000
2640	U4BRG <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	Baud Rate Generator Prescaler														0000		
2800	U5MODE <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	SLPEN	ACTIVE	—	—	—	CLKSEL<1:0>	RUNOVF	0000
		15:0	ON	—	SIDL	IREN	RTSMD	—	—	—	—	—	UEN<1:0>	WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL<1:0>
2810	U5STA <sup>(1)</sup>	31:16	MASK<7:0>							ADDR<7:0>							0000		
		15:0	UTXISEL<1:0>		UTXINV	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA	0110
2820	U5TXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	TX8	Transmit Register							0000
2830	U5RXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	RX8	Receive Register							0000
2840	U5BRG <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	Baud Rate Generator Prescaler														0000		
2A00	U6MODE <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	SLPEN	ACTIVE	—	—	—	CLKSEL<1:0>	RUNOVF	0000
		15:0	ON	—	SIDL	IREN	RTSMD	—	—	—	—	—	UEN<1:0>	WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL<1:0>
2A10	U6STA <sup>(1)</sup>	31:16	MASK<7:0>							ADDR<7:0>							0000		
		15:0	UTXISEL<1:0>		UTXINV	URXEN	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA	0110
2A20	U6TXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	TX8	Transmit Register							0000
2A30	U6RXREG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	RX8	Receive Register							0000
2A40	U6BRG <sup>(1)</sup>	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	Baud Rate Generator Prescaler														0000		

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: このレジスタは対応する CLR、SET、INV レジスタを持ち、それらはレジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 24-1: UxMODE: UARTx モードレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	R/W-0 SLPEN	R-0, HS, HC ACTIVE	U-0 —	U-0 —	U-0 —	R/W-0 CLKSEL<1:0>	R/W-0	R/W-0 RUNOVF
15:8	R/W-0 ON	U-0 —	R/W-0 SIDL	R/W-0 IREN	R/W-0 RTSMD	U-0 —	R/W-0 UEN<1:0> <sup>(1)</sup>	R/W-0
7:0	R/W-0 WAKE	R/W-0 LPBACK	R/W-0 ABAUD	R/W-0 RXINV	R/W-0 BRGH	R/W-0 PDSSEL<1:0>	R/W-0	R/W-0 STSEL

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

bit 31-24 **未実装:** 「0」として読み出し

bit 23 **SLPEN:** スリープ中動作イネーブルビット

- 1 = スリープ中も UARTx BRG クロックは動作する
- 0 = スリープ中に UARTx BRG クロックは停止する

**Note:** SLPEN = 1 は、CLKSEL = FRC の場合にのみ適用されます。全てのクロックと UART は、ディープスリープ中に無効になります。

bit 22 **ACTIVE:** UARTx モジュール動作ステータスビット

- 1 = UARTx モジュールはアクティブ (UxMODE レジスタは変更不可)
- 0 = UARTx モジュールは非アクティブ (UxMODE レジスタは変更可能)

bit 21-19 **未実装:** 「0」として読み出し

bit 18-17 **CLKSEL<1:0>:** UARTx モジュール クロック 選択ビット

- 11 = BRG クロックは PBCLK2
- 10 = BRG クロックは FRC
- 01 = BRG クロックは SYSCLK (スリープ中は停止)
- 00 = BRG クロックは PBCLK2 (スリープ中は停止)

bit 16 **RUNOVF:** オーバーフロー条件中動作モードビット

- 1 = オーバーフロー エラー (OERR) 条件の検出時もシフトレジスタは動作を継続して同期を維持する
- 0 = オーバーフロー エラー (OERR) 条件の検出時にシフトレジスタは新しいデータを受け入れない (レガシーモード)

bit 15 **ON:** UARTx イネーブルビット

- 1 = UARTx モジュールを有効にする
- UARTx ピンは UARTx モジュールによって制御されます (UEN<1:0> および UTXEN 制御ビットの定義に従う)。
- 0 = UARTx モジュールを無効にする
- 全ての UARTx ピンは PORTx、TRISx、LATx レジスタ内の対応するビットによって制御され、UARTx の消費電力は最小になります。

bit 14 **未実装:** 「0」として読み出し

bit 13 **SIDL:** アイドル中停止ビット

- 1 = デバイスがアイドルに移行した時に動作を停止する
- 0 = アイドル中も動作を継続する

**Note 1:** これらのビットはレガシー互換性のために提供されます。PIC32MZ DA ファミリのデバイスでは、このビットの機能は PPS 機能に取って代わられます (詳細は [12.4「ペリフェラル ピンセレクト \(PPS\)」](#)参照)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 24-1: UxMODE: UARTx モードレジスタ (続き)

- bit 12 **IREN:** IrDA<sup>®</sup> エンコーダ / デコーダ イネーブルビット  
1 = IrDA を有効にする  
0 = IrDA を無効にする
- bit 11 **RTSM:**  $\overline{\text{UxRTS}}$  ピンモード選択ビット  
1 =  $\overline{\text{UxRTS}}$  ピンを片方向モードにする  
0 =  $\overline{\text{UxRTS}}$  ピンをフロー制御モードにする
- bit 10 **未実装:** 「0」として読み出し
- bit 9-8 **UEN<1:0>:** UARTx モジュール イネーブルビット (1)  
11 = UxTX, UxRX, UxBCLK ピンを有効にして使い、 $\overline{\text{UxCTS}}$  ピンは PORTx レジスタ内の対応するビットにより制御する  
10 = UxTX, UxRX,  $\overline{\text{UxCTS}}$ ,  $\overline{\text{UxRTS}}$  ピンを有効にして使う  
01 = UxTX, UxRX, UxRTS ピンを有効にして使い、 $\overline{\text{UxCTS}}$  ピンは PORTx レジスタ内の対応するビットにより制御する  
00 = UxTX および UxRX ピンを有効にして使い、 $\overline{\text{UxCTS}}$  および  $\overline{\text{UxRTS}}$  / UxBCLK ピンは PORTx レジスタ内の対応するビットにより制御する
- bit 7 **WAKE:** スリープ中スタートビット検出時復帰イネーブルビット  
1 = 復帰を有効にする  
0 = 復帰を無効にする
- bit 6 **LPBACK:** UARTx ループバック モード選択ビット  
1 = ループバック モードを有効にする  
0 = ループバック モードを無効にする
- bit 5 **ABAUD:** baud レート自動検出イネーブルビット  
1 = 次のキャラクタで baud レート計測を有効にする (同期キャラクタ (0x55) の受信が必要) このビットは完了時にハードウェアによってクリアされます。  
0 = baud レート自動検出を無効にする、または自動検出は完了した
- bit 4 **RXINV:** 受信極性反転ビット  
1 = UxRX のアイドル状態は「0」  
0 = UxRX のアイドル状態は「1」
- bit 3 **BRGH:** 高 baud レート イネーブルビット  
1 = 高速モード (4x baud クロックを有効にする)  
0 = 標準速モード (16x baud クロックを有効にする)
- bit 2-1 **PDSEL<1:0>:** パリティ / データ選択ビット  
11 = 9 ビットデータ、パリティなし  
10 = 8 ビットデータ、奇数パリティ  
01 = 8 ビットデータ、偶数パリティ  
00 = 8 ビットデータ、パリティなし
- bit 0 **STSEL:** ストップビット選択ビット  
1 = 2 個のストップビット  
0 = 1 個のストップビット

**Note 1:** これらのビットはレガシー互換性のために提供されます。PIC32MZ DA ファミリのデバイスでは、このビットの機能は PPS 機能に取って代わられます (詳細は [12.4「ペリフェラルピンセレクト \(PPS\)」](#)参照)。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 24-2: UxSTA: UARTx ステータス / 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MASK<7:0>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADDR<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R-1
	UTXISEL<1:0>		UTXINV	URXEN	UTXBRK	UTXEN	UTXBF	TRMT
7:0	R/W-0	R/W-0	R/W-0	R-1	R-0	R-0	R/W-0	R-0
	URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR	OERR	URXDA

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-25 **MASK<7:0>**: UARTx アドレス一致マスクビット

これらのビットは ADDR<7:0> ビットのマスク用に使います。  
 11111111 = 対応する ADDR<sub>x</sub> ビットを使ってアドレス一致を検出する

**Note:** この設定により、個々のアドレスおよびグループブロードキャストアドレスを UART に割り当てる事ができます。

00000000 = 対応する ADDR<sub>x</sub> ビットはアドレス一致検出用に使わない

bit 23-16 **ADDR<7:0>**: 自動アドレスマスク ビット

ADDEN ビットを「1」にセットした場合、この値によって自動アドレス検出に使うアドレス キャラクタを定義します。

bit 15-14 **UTXISEL<1:0>**: TX 割り込みモード選択ビット

11 = 予約済み (使用禁止)  
 10 = 送信バッファがエンプティであれば割り込みフラグビットをアサートし続ける  
 01 = 全てのキャラクタの送信が完了した後に割り込みフラグビットをアサートし続ける  
 00 = 送信バッファに 1 つ以上の空き位置が存在すれば割り込みフラグビットをアサートし続ける

bit 13 **UTXINV**: 送信極性反転ビット

IrDA モードが無効 (IREN (UxMODE<12>) = 0) の場合:  
 1 = UxTX のアイドル状態は「0」  
 0 = UxTX のアイドル状態は「1」

IrDA モードが有効 (IREN (UxMODE<12>) = 1) の場合:  
 1 = IrDA エンコードされ UxTX のアイドル状態は「1」  
 0 = IrDA エンコードされた UxTX のアイドル状態は「0」

bit 12 **URXEN**: レシーバイネーブルビット

1 = UARTx レシーバを有効にする  
 ON = 1 の場合、UARTx モジュールが UxRX ピンを制御します。  
 0 = UARTx レシーバを無効にする  
 UARTx モジュールは UxRX ピンを無視します。

**Note:** 有効であったレシーバを無効にすると、RX ピンはポート機能向けに解放されますが、受信バッファはリセットされません。レシーバを無効にしても受信ステータスフラグには影響しません。

bit 11 **UTXBRK**: 送信ブレイクビット

1 = 次の送信でブレイクを送信する  
 スタートビットの後に 12 個の「0」ビットとストップビットが続きます。このビットはブレイク送信完了時にハードウェアによってクリアされます。  
 0 = ブレイク送信は無効または完了した

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 24-2: UxSTA: UARTx ステータス / 制御レジスタ ( 続き )

- bit 10 **UTXEN:** 送信イネーブルビット  
1 = UARTx トランスミッタを有効にする  
ON = 1 の場合、UARTx モジュールが UxTX ピンを制御します。  
0 = UARTx トランスミッタを無効にする  
保留中の送信は全て中止され、バッファはリセットされます。  
**Note:** 有効であったトランスミッタを無効にすると、TX ピンはポート機能向けに解放され、送信バッファはリセットされてエンプティになります。保留中の送信は全て中止され、送信バッファ内のデータ キャラクタは失われます。全ての送信ステータスフラグはクリアされ、TRMT ビットがセットされます。
- bit 9 **UTXBF:** 送信バッファフルステータスビット ( 読み出し専用 )  
1 = 送信バッファはフル  
0 = 送信バッファは非フル ( 少なくとも 1 つのキャラクタが書き込める )
- bit 8 **TRMT:** 送信シフトレジスタ エンプティビット ( 読み出し専用 )  
1 = 送信シフトレジスタも送信バッファもエンプティ ( 直前の送信は完了した )  
0 = 送信シフトレジスタは非エンプティ ( 送信中または送信バッファ内に送信待ちのデータが存在する )
- bit 7-6 **URXISEL<1:0>:** 受信割り込みモード選択ビット  
11 = 予約済み  
10 = 受信バッファが 3/4 フル以上であれば割り込みフラグビットをアサートし続ける  
01 = 受信バッファが 1/2 フル以上であれば割り込みフラグビットをアサートし続ける  
00 = 受信バッファがエンプティでなければ(少なくとも1個のデータキャラクタを格納していれば)割り込みフラグビットをアサートし続ける
- bit 5 **ADDEN:** アドレス キャラクタ ( 受信データの bit 8 = 1 ) 検出ビット  
1 = アドレス検出モードを有効にする  
このビットは 9 ビットモードを選択した場合にのみ効果を持ちます。  
0 = アドレス検出モードを無効にする
- bit 4 **RIDL:** レシーバアイドル ビット ( 読み出し専用 )  
1 = レシーバはアイドル中  
0 = レシーバはデータを受信中
- bit 3 **PERR:** パリティエラー ステータスビット ( 読み出し専用 )  
1 = 現在のキャラクタでパリティエラーを検出した  
0 = パリティエラーは検出していない
- bit 2 **FERR:** フレーミング エラー ステータスビット ( 読み出し専用 )  
1 = 現在のキャラクタでフレーミング エラーを検出した  
0 = フレーミング エラーは検出していない
- bit 1 **OERR:** 受信バッファ オーバーラン エラー ステータスビット  
このビットはハードウェアでセットされ、ソフトウェアでのみクリア (=0) できます。セットされた OERR ビットをクリアすると、受信バッファと RSR はエンプティ状態にリセットされます。  
1 = 受信バッファがオーバーフローした  
0 = 受信バッファはオーバーフローしていない
- bit 0 **URXDA:** 受信バッファ内データ ステータスビット ( 読み出し専用 )  
1 = 受信バッファ内にデータが存在する ( 少なくとも 1 つのキャラクタを読み出せる )  
0 = 受信バッファはエンプティ

# PIC32MZ グラフィック (DA) ファミリ

図 24-2 と図 24-3 に、UART モジュールの代表的な送受信タイミングを示します。

図 24-2: UART 受信

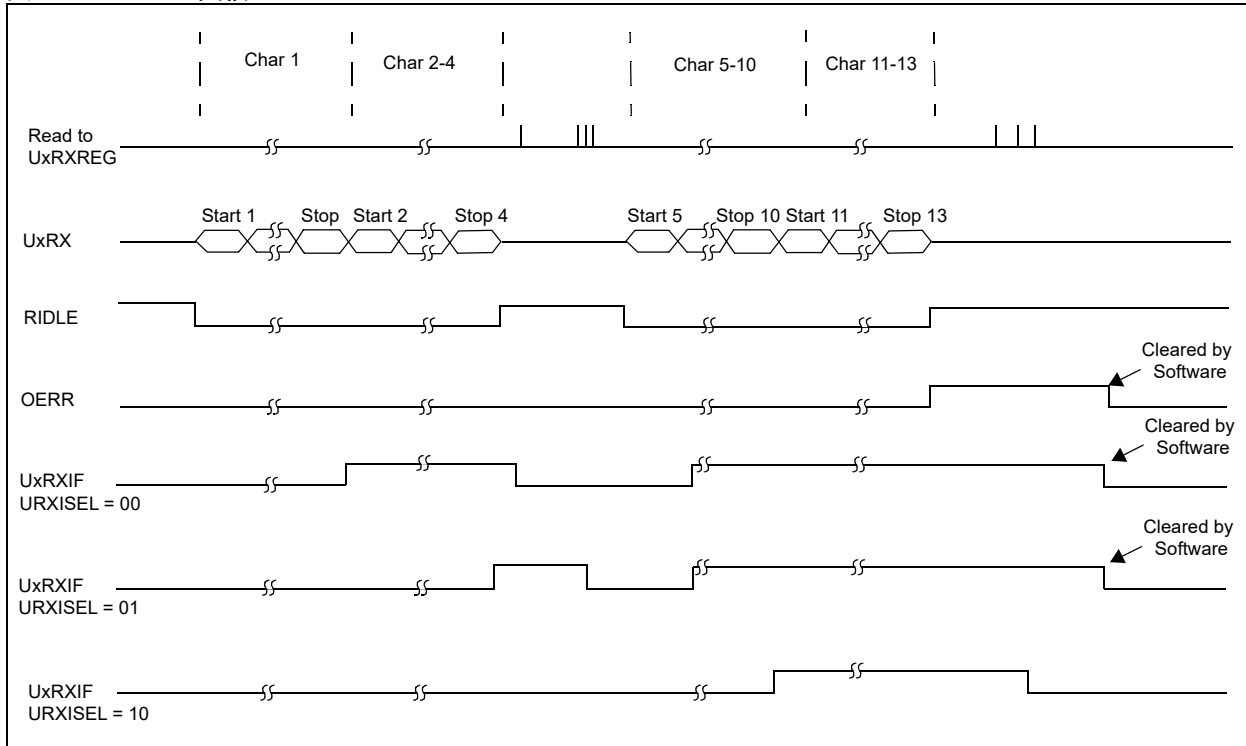
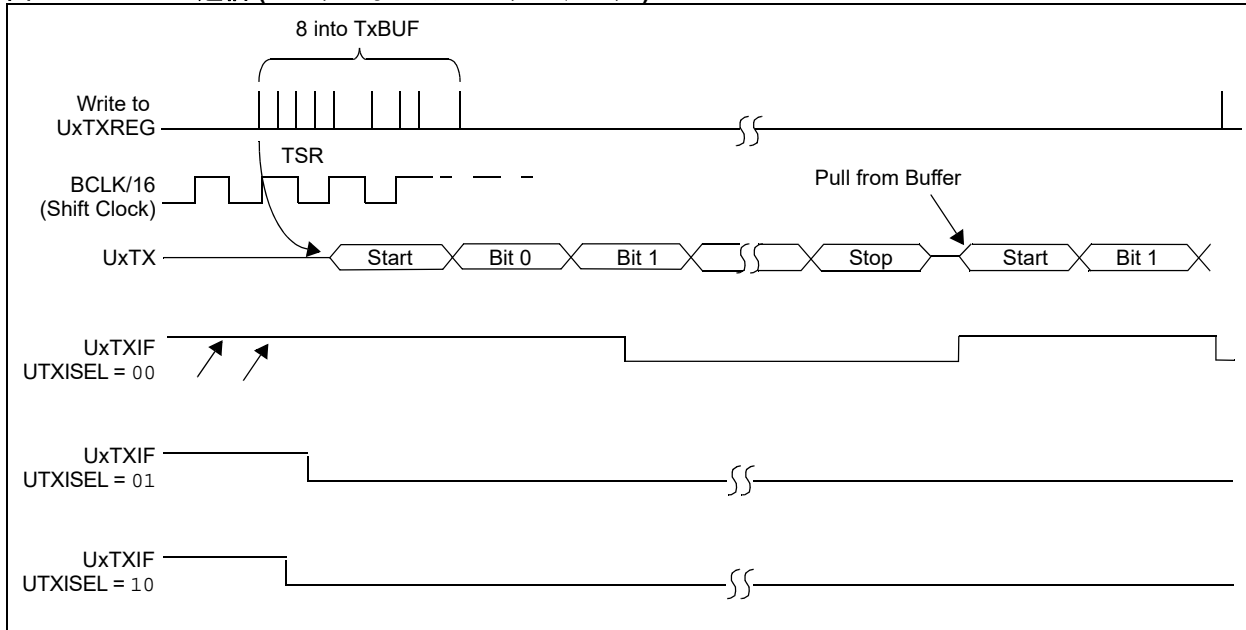


図 24-3: UART 送信 (8 ビットまたは 9 ビットデータ)





# PIC32MZ グラフィック (DA) ファミリ

## 25.0 パラレルマスタポート (PMP)

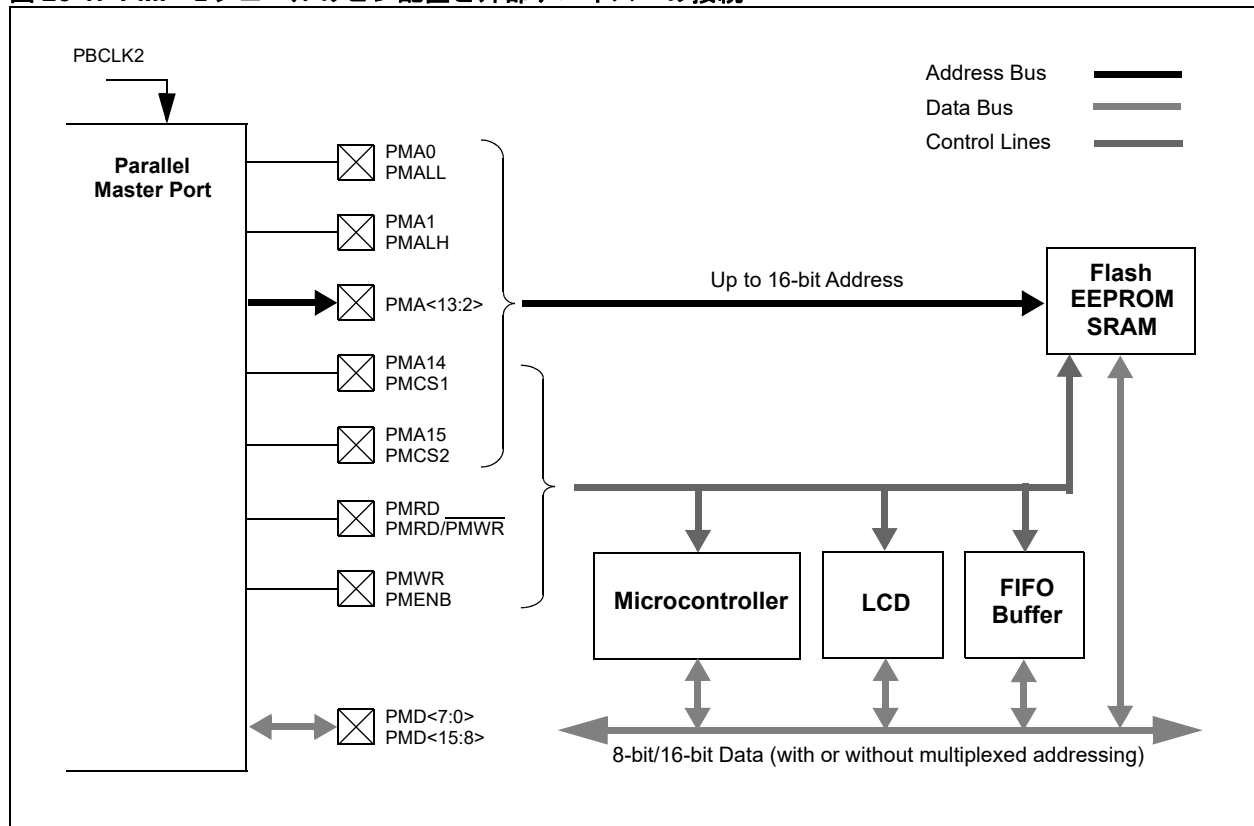
**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 13. Parallel Master Port (PMP)』(DS60001128) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

PMPは各種の平行デバイス(通信用周辺モジュール、LCD、外部メモリデバイス、マイクロコントローラ等)との通信用に設計された平行8/16ビット入出力モジュールです。平行周辺モジュールとのインターフェイスは非常に多岐にわたるため、PMPモジュールは非常に柔軟に設定できます。

PMPモジュールの主な特長は以下の通りです。

- 8ビット/16ビットインターフェイス
- 最大16本の設定可能アドレスライン
- 最大2本のチップセレクトライン
- プログラマブルなストロブオプション
  - 読み出しと書き込みで別々のストロブ、または、
  - イネーブルストロブを使った読み/書きストロブ
- アドレスの自動インクリメント/デクリメント
- プログラマブルなアドレス/データ多重化
- プログラマブルな制御信号極性
- 平行スレーブポートをサポート
  - レガシーのアドレス指定が可能
  - アドレスサポート
  - 4段のバイト幅自動インクリメントバッファ
- プログラマブルな待機ステート
- スリープおよびアイドル中の動作
- マスタモードでは、読み出しレジスタと書き込みレジスタを別々にしたデュアルバッファを構成可能
- CLR、SET、INVレジスタを使う高速なビット操作

図 25-1: PMP モジュールのピン配置と外部デバイスへの接続



## 25.1 制御レジスタ

表 25-1: パラレルマスタ ポートのレジスタマップ

仮想アドレス (BF82_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット 値
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
E000	PMCON	31:16	—	—	—	—	—	—	—	—	RDSTART	—	—	—	—	—	DUALBUF	—	0000
		15:0	ON	—	SIDL	ADRMUX<1:0>	PMPCTL	PTWREN	PTRDEN	CSF<1:0>	ALP	CS2P	CS1P	—	—	—	—	WRSP	RDSP
E010	PMMODE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	BUSY	IRQM<1:0>	INCM<1:0>	MODE16	MODE<1:0>	WAITB<1:0>	WAITM<3:0>	WAITE<1:0>	—	—	—	—	—	—	—	—	0000
E020	PMADDR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CS2	CS1	ADDR<13:0>														0000
E030	PMDOUT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	DATAOUT<15:0>																0000
E040	PMDIN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	DATAIN<15:0>																0000
E050	PMAEN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PTEN<15:0>																0000
E060	PMSTAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	IBF	IBOV	—	—	IB3F	IB2F	IB1F	IB0F	OBE	OBUF	—	—	OB3E	OB2E	OB1E	OB0E	008F
E070	PMWADDR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	WCS2	WCS1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
E080	PMRADDR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RCS2	RCS1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
E090	PMRDIN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RDATAIN<15:0>																0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-1: PMCON: パラレルポート制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	R/W-0, HC RDSTART	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 DUALBUF	U-0 —
15:8	R/W-0 ON <sup>(1)</sup>	U-0 —	R/W-0 SIDL	R/W-0 ADRMUX<1:0>	R/W-0 —	R/W-0 PMPTTL	R/W-0 PTWREN	R/W-0 PTRDEN
7:0	R/W-0 CSF<1:0> <sup>(2)</sup>	R/W-0 —	R/W-0 ALP <sup>(2)</sup>	R/W-0 CS2P <sup>(2)</sup>	R/W-0 CS1P <sup>(2)</sup>	U-0 —	R/W-0 WRSP	R/W-0 RDSP

凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-24 **未実装**: 「0」として読み出し

bit 23 **RDSTART**: PMP バス読み出し開始ビット

このビットは、読み出しサイクルの最後でハードウェアによってクリアされます。

1 = PMP バスで読み出しサイクルを開始する

0 = 何もしない

bit 22-18 **未実装**: 「0」として読み出し

bit 17 **DUALBUF**: パラレルマスタポートデュアル読み書きバッファイネーブルビット

このビットはマスタモードでのみ有効です。

1 = PMP は読み出しと書き込み用に別々のレジスタを使う

読み出し用: PMRADDR と PMRDIN

書き込み用: PMRWADDR と PMDOUT

0 = PMP は読み出しと書き込み用にレガシーのレジスタを使う

読み/書き用: PMADDR と PMRDIN

bit 16 **未実装**: 「0」として読み出し

bit 15 **ON**: パラレルマスタポートイネーブルビット<sup>(1)</sup>

1 = PMP を有効にする

0 = PMP を無効にする (外部と通信しない)

bit 14 **未実装**: 「0」として読み出し

bit 13 **SIDL**: アイドル中停止ビット

1 = デバイスがアイドルに移行した時にモジュールの動作を停止する

0 = アイドル中もモジュールの動作を継続する

bit 12-11 **ADRMUX<1:0>**: アドレス/データ多重化選択ビット

11 = アドレスの 16 ビット全てを PMD<15:0> に多重化する

10 = アドレスの 16 ビット全てを PMD<7:0> に多重化する

01 = アドレスの下位 8 ビットを PMD<7:0> に多重化し、上位 8 ビットを PMA<15:8> に多重化する

00 = アドレスとデータに別々のピンを使う

bit 10 **PMPTTL**: PMP モジュール TTL 入力バッファ選択ビット

1 = PMP モジュールは TTL 入力バッファを使う

0 = PMP モジュールはシュミットトリガ入力バッファを使う

bit 9 **PTWREN**: 書き込みイネーブルストローブポートイネーブルビット

1 = PMWR/PMENB ポートを有効にする

0 = PMWR/PMENB ポートを無効にする

**Note 1**: PBCLK 分周比を 1:1 にしている場合、周辺モジュールの ON 制御ビットをクリアした命令の直後の SYSCLK サイクルでは、ユーザソフトウェアでそのモジュールの SFR を読み書きしない必要があります。

**2**: 対応するピンをアドレスラインとして使う場合、これらのビットは効果を持ちません。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 25-1: PMCON: パラレルポート制御レジスタ (続き)

- bit 8 **PTRDEN**: 読み / 書きストローブポート イネーブルビット  
1 = PMRD/PMWR ポートを有効にする  
0 = PMRD/PMWR ポートを無効にする
- bit 7-6 **CSF<1:0>**: チップセレクト機能ビット (2)  
11 = 予約済み  
10 = PMCS1 と PMCS2 の両方がチップセレクトとして機能する  
01 = PMCS1 はアドレスの bit 14 として機能し、PMCS2 はチップセレクトとして機能する  
00 = PMCS1 と PMCS2 はそれぞれアドレスの bit 14 と bit 15 として機能する
- bit 5 **ALP**: アドレスラッチ極性ビット (2)  
1 = アクティブ HIGH (PMALL、PMALH)  
0 = アクティブ LOW (PMALL、PMALH)
- bit 4 **CS2P**: チップセレクト 0 極性ビット (2)  
1 = アクティブ HIGH (PMCS2)  
0 = アクティブ LOW (PMCS2)
- bit 3 **CS1P**: チップセレクト 0 極性ビット (2)  
1 = アクティブ HIGH (PMCS1)  
0 = アクティブ LOW (PMCS1)
- bit 2 **未実装**: 「0」 として読み出し
- bit 1 **WRSP**: 書き込みストローブ極性ビット  
スレーブモードとマスタモード 2 (MODE<1:0> = 00、01、10) の場合:  
1 = 書き込みストローブはアクティブ HIGH (PMWR)  
0 = 書き込みストローブはアクティブ LOW (PMWR)  
マスタモード 1 (MODE<1:0> = 11) の場合:  
1 = イネーブル ストローブはアクティブ HIGH (PMENB)  
0 = イネーブル ストローブはアクティブ LOW (PMENB)
- bit 0 **RDSP**: 読み出しストローブ極性ビット  
スレーブモードとマスタモード 2 (MODE<1:0> = 00、01?10) の場合:  
1 = 読み出しストローブはアクティブ HIGH (PMRD)  
0 = 読み出しストローブはアクティブ LOW (PMRD)  
マスタモード 1 (MODE<1:0> = 11) の場合:  
1 = 読み / 書きストローブはアクティブ HIGH (PMRD/PMWR)  
0 = 読み / 書きストローブはアクティブ LOW (PMRD/PMWR)

- Note 1:** PBCLK 分周比を 1:1 にしている場合、周辺モジュールの ON 制御ビットをクリアした命令の直後の SYSCLK サイクルでは、ユーザ ソフトウェアでそのモジュールの SFR を読み書きしない必要があります。
- 2:** 対応するピンをアドレスラインとして使う場合、これらのビットは効果を持ちません。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-2: PMMODE: パラレルポート モードレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R-0 BUSY	R/W-0 IRQM<1:0>	R/W-0	R/W-0 INCM<1:0>	R/W-0	R/W-0 MODE16	R/W-0 MODE<1:0>	R/W-0
7:0	R/W-0 WAITB<1:0> <sup>(1)</sup>	R/W-0	R/W-0	R/W-0	R/W-0 WAITM<3:0> <sup>(1)</sup>	R/W-0	R/W-0 WAITE<1:0> <sup>(1)</sup>	R/W-0

凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15 **BUSY**: ビジービット (マスタモード専用)

1 = ポートはビジー

0 = ポートは非ビジー

bit 14-13 **IRQM<1:0>**: 割り込み要求モードビット

11 = 予約済み (使用禁止)

10 = 読み出しバッファ3からの読み出し時または書き込みバッファ3への書き込み時に割り込みを生成する (バッファ付き PSP モード)

または、PMA<1:0> = 11 の場合、読み / 書き動作時に割り込みを生成する (アドレス可能スレーブモードのみ)

01 = 読み / 書きサイクルの終了時に割り込みを生成する

00 = 割り込みを生成しない

bit 12-11 **INCM<1:0>**: インクリメント モードビット

11 = スレーブモードの読み出しおよび書き込みバッファを自動的にインクリメントする (MODE<1:0> = 00 の場合のみ)

10 = 読み / 書きサイクルのたびに ADDR<15:0> を 1 つデクリメントする <sup>(2)</sup>

01 = 読み / 書きサイクルのたびに ADDR<15:0> を 1 つインクリメントする <sup>(2)</sup>

00 = アドレスをインクリメントもデクリメントもしない

bit 10 **MODE16**: 8/16 ビットモード ビット

1 = 16 ビットモード: データレジスタを読み / 書きするたびに 16 ビット転送を 1 回実行する

0 = 8 ビットモード: データレジスタを読み / 書きするたびに 8 ビット転送を 1 回実行する

bit 9-8 **MODE<1:0>**: パラレルポート モード選択ビット

11 = マスタモード 1 (PMCSx, PMRD/PMWR, PMENB, PMA<x:0>, PMD<7:0>, PMD<8:15><sup>(3)</sup>)

10 = マスタモード 2 (PMCSx, PMRD, PMWR, PMA<x:0>, PMD<7:0>, PMD<8:15><sup>(3)</sup>)

01 = 拡張スレーブモード (PMRD, PMWR, PMCS, PMD<7:0>, PMA<1:0>)

00 = レガシー パラレル スレーブポート (PMRD, PMWR, PMCS, PMD<7:0>)

bit 7-6 **WAITB<1:0>**: 読み / 書きスロープ待機ステート データ セットアップ ビット <sup>(1)</sup>

11 = 4 TPB のデータ待機 (4 TPB のアドレス多重化フェイズ)

10 = 3 TPB のデータ待機 (3 TPB のアドレス多重化フェイズ)

01 = 2 TPB のデータ待機 (2 TPB のアドレス多重化フェイズ)

00 = 1 TPB のデータ待機 (1 TPB のアドレス多重化フェイズ) (既定値)

**Note 1:** WAITM<3:0> = 0000 に設定した場合、WAITB および WAITE ビットの設定に関係なく、書き込み動作の場合は WAITB と WAITE の両方に 1 TPBCLK サイクルを適用し、読み出し動作の場合は WAITB に 1 TPBCLK サイクルと WAITE に 0 TPBCLK サイクルを適用します。

**2:** アドレスビット A15/A14 をチップセレクト CS2/CS1 として設定した場合、これらのビットには自動インクリメント / デクリメントは適用されません。

**3:** これらのピンは、MODE16 = 1 (16 ビットモード) の場合にアクティブです。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 25-2: PMMODE: パラレルポート モードレジスタ (続き)

bit 5-2 **WAITM<3:0>**: 読み / 書きストローブ データ待機ステートビット<sup>(1)</sup>

1111 = 16 TPB の待機

•  
•  
•

0001 = 2 TPB の待機

0000 = 1 TPB の待機 (既定値)

bit 1-0 **WAITE<1:0>**: 読み / 書きストローブ後データホールド待機ステートビット<sup>(1)</sup>

11 = 4 TPB の待機

10 = 3 TPB の待機

01 = 2 TPB の待機

00 = 1 TPB の待機 (既定値)

読み出し動作の場合:

11 = 3 TPB の待機

10 = 2 TPB の待機

01 = 1 TPB の待機

00 = 0 TPB の待機 (既定値)

- Note 1:** WAITM<3:0> = 0000 に設定した場合、WAITB および WAITE ビットの設定に関係なく、書き込み動作の場合は WAITB と WAITE の両方に 1 TPBCLK サイクルを適用し、読み出し動作の場合は WAITB に 1 TPBCLK サイクルと WAITE に 0 TPBCLK サイクルを適用します。
- 2:** アドレスビット A15/A14 をチップセレクト CS2/CS1 として設定した場合、これらのビットには自動インクリメント / デクリメントは適用されません。
- 3:** これらのピンは、MODE16 = 1 (16 ビットモード) の場合にアクティブです。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-3: PMADDR: パラレルポート アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CS2 <sup>(1)</sup>	CS1 <sup>(3)</sup>	ADDR<13:8>					
	ADDR15 <sup>(2)</sup>	ADDR14 <sup>(4)</sup>						
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADDR<7:0>							

凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15 **CS2**: チップセレクト 2 ビット<sup>(1)</sup>

1 = チップセレクト 2 はアクティブ

0 = チップセレクト 2 は非アクティブ

bit 15 **ADDR<15>**: ターゲット アドレス bit 15<sup>(2)</sup>

bit 14 **CS1**: チップセレクト 1 ビット<sup>(3)</sup>

1 = チップセレクト 1 はアクティブ

0 = チップセレクト 1 は非アクティブ

bit 14 **ADDR<14>**: ターゲット アドレス bit 14<sup>(4)</sup>

bit 13-0 **ADDR<13:0>**: アドレスビット

**Note 1**: CSF<1:0> ビット (PMCON<7:6>) = 10 または 01 の場合

**2**: CSF<1:0> ビット (PMCON<7:6>) = 00 の場合

**3**: CSF<1:0> ビット (PMCON<7:6>) = 10 の場合

**4**: CSF<1:0> ビット (PMCON<7:6>) = 00 または 01 の場合

**Note**: DUALBUF ビット (PMCON<17>) = 0 の場合、このレジスタ内のビットは、読み出しと書き込み両方のターゲット アドレス指定を制御します。DUALBUF ビット = 1 の場合、このレジスタ内のビットは使いません。この場合、PMRADDR レジスタを読み出し動作に使い、PMWADDR レジスタを書き込み動作に使います。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-4: PMDOUT: パラレルポート出力データレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DATAOUT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DATAOUT<7:0>							

凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **DATAOUT<15:0>**: ポートデータ出力ビット

このレジスタは、拡張パラレルスレーブモードでの読み出し動作と、デュアルバッファマスタモードでの書き込み動作向けに使用します。

デュアルバッファマスタモード (DUALBUF ビット (PMPCON<17>) = 1) の場合、MSB への書き込みによって PMP ポート上のトランザクションをトリガします。MODE16 = 1 の場合、MSB は DATAOUT<15:8> です。MODE16 = 0 の場合、MSB は DATAOUT<7:0> です。

**Note:** マスタモードの場合、読み出しはレジスタに直近に書き込まれた値を返します。スレーブモードの場合、読み出しは中間結果を返します。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-5: PMDIN: パラレルポート入力データレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DATAIN<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DATAIN<7:0>							

凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **DATAIN<15:0>**: ポートデータ入力ビット

このレジスタは、パラレルマスタポートモードと拡張パラレルスレーブモードで使います。

パラレルマスタモードの場合、MSB に対する書き込みによって PMP ポート上の書き込みトランザクションをトリガします。同様に、MSB に対する読み出しによって PMP ポート上の読み出しトランザクションをトリガします。MODE16 = 1 の場合、MSB は DATAIN<15:8> です。MODE16 = 0 の場合、MSB は DATAIN<7:0> です。

**Note:** このレジスタは、デュアルバッファマスタモード (DUALBUF ビット (PMPCON<17>) = 1) では使いません。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-6: PMAEN: パラレルポート ピンイネーブル レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PTEN<15:14> <sup>(1)</sup>		PTEN<13:8>					
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PTEN<7:2>						PTEN<1:0> <sup>(2)</sup>	

凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」を書き込み、読み出しは無視

bit 15-14 **PTEN<15:14>**: PMCSx アドレスポート イネーブルビット

1 = PMA15 と PMA14 は PMA<15:14> または PMCS2/PMCS1 として機能する <sup>(1)</sup>

0 = PMA15 と PMA14 はポート I/O として機能する

bit 13-2 **PTEN<13:2>**: PMP アドレスポート イネーブルビット

1 = PMA<13:2> は PMP アドレスラインとして機能する

0 = PMA<13:2> はポート I/O として機能する

bit 1-0 **PTEN<1:0>**: PMALH/PMALL アドレスポート イネーブルビット

1 = PMA1 と PMA0 は PMA<1:0> または PMALH/PMALL として機能する <sup>(2)</sup>

0 = PMA1 と PMA0 はポート I/O として機能する

**Note 1:** これらのピンを PMA<15:14> または PMCS2/PMCS1 のどちらとして使うかは、CSF<1:0> ビット (PMCON<7:6>) で選択します。

**2:** これらのピンを PMA<1:0> または PMALH/PMALL のどちらとして使うかは、PMCON レジスタの ADRMUX<1:0> ビットで選択されているアドレス / データ多重化モードによって決まります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-7: PMSTAT: パラレルポート ステータス レジスタ (スレーブモード専用)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R-0	R/W-0, HS, SC	U-0	U-0	R-0	R-0	R-0	R-0
	IBF	IBOV	—	—	IB3F	IB2F	IB1F	IB0F
7:0	R-1	R/W-0, HS, SC	U-0	U-0	R-1	R-1	R-1	R-1
	OBE	OBUF	—	—	OB3E	OB2E	OB1E	OB0E

凡例:	HS = ハードウェアでセット	SC = ソフトウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15 **IBF**: 入力バッファフル ステータスビット

- 1 = 書き込み可能な入力バッファレジスタの全てがフル
- 0 = 書き込み可能な入力バッファレジスタの一部または全てがエンプティ

bit 14 **IBOV**: 入力バッファ オーバーフロー ステータスビット

- 1 = フル状態の入力バイトバッファに対する書き込みが発生した (このビットはソフトウェアでクリアする必要があります)
- 0 = オーバーフローは発生していない

bit 13-12 **未実装**: 「0」として読み出し

bit 11-8 **IBxF**: 入力バッファ 「x」ステータスフル ビット

- 1 = 入力バッファに未読データが存在する (このビットはバッファ読み出し時にクリアされます)
- 0 = 入力バッファに未読データは存在しない

bit 7 **OBE**: 出力バッファ エンプティ ステータスビット

- 1 = 読み出し可能な出力バッファレジスタの全てがエンプティ
- 0 = 読み出し可能な出力バッファレジスタの一部または全てがフル

bit 6 **OBUF**: 出力バッファ アンダーフロー ステータスビット

- 1 = エンプティ状態の出力バイトバッファからの読み出しが発生した (このビットはソフトウェアでクリアする必要があります)
- 0 = アンダーフローは発生していない

bit 5-4 **未実装**: 「0」として読み出し

bit 3-0 **OBxE**: 出力バッファ 「x」ステータス エンプティビット

- 1 = 出力バッファはエンプティ (このビットはバッファへのデータ書き込み時にクリアされます)
- 0 = 出力バッファ内に未送信データが存在する

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-8: PMWADDR: パラレルポート書き込みアドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	WCS2 <sup>(1)</sup>	WCS1 <sup>(3)</sup>	WADDR<13:8>					
	WADDR15 <sup>(2)</sup>	WADDR14 <sup>(4)</sup>						
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	WADDR<7:0>							

凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15 **WCS2**: チップセレクト 2 ビット<sup>(1)</sup>

1 = チップセレクト 2 はアクティブ

0 = チップセレクト 2 は非アクティブ

bit 15 **WADDR<15>**: ターゲット アドレス bit 15<sup>(2)</sup>

bit 14 **WCS1**: チップセレクト 1 ビット<sup>(3)</sup>

1 = チップセレクト 1 はアクティブ

0 = チップセレクト 1 は非アクティブ

bit 14 **WADDR<14>**: ターゲット アドレス bit 14<sup>(4)</sup>

bit 13-0 **WADDR<13:0>**: アドレスビット

**Note 1**: CSF<1:0> ビット (PMCON<7:6>) = 10 または 01 の場合

**2**: CSF<1:0> ビット (PMCON<7:6>) = 00 の場合

**3**: CSF<1:0> ビット (PMCON<7:6>) = 10 の場合

**4**: CSF<1:0> ビット (PMCON<7:6>) = 00 または 01 の場合

**Note**: このレジスタは、DUALBUF ビット (PMCON<17>) を「1」にセットした場合にのみ使います。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-9: PMRADDR: パラレルポート読み出しアドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RCS2 <sup>(1)</sup>	RCS1 <sup>(3)</sup>	RADDR<13:8>					
	RADDR15 <sup>(2)</sup>	RADDR14 <sup>(4)</sup>						
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RADDR<7:0>							

凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15 RCS2: チップセレクト 2 ビット<sup>(1)</sup>

1 = チップセレクト 2 はアクティブ

0 = チップセレクト 2 は非アクティブ (RADDR15 機能を選択する)

bit 15 **RADDR<15>**: ターゲット アドレス bit 15<sup>(2)</sup>

bit 14 **RCS1**: チップセレクト 1 ビット<sup>(3)</sup>

1 = チップセレクト 1 はアクティブ

0 = チップセレクト 1 は非アクティブ (RADDR14 機能を選択する)

bit 14 **RADDR<14>**: ターゲット アドレス bit 14<sup>(4)</sup>

bit 13-0 **RADDR<13:0>**: アドレスビット

**Note 1:** CSF<1:0> ビット (PMCON<7:6>) = 10 または 01 の場合

**2:** CSF<1:0> ビット (PMCON<7:6>) = 00 の場合

**3:** CSF<1:0> ビット (PMCON<7:6>) = 10 の場合

**4:** CSF<1:0> ビット (PMCON<7:6>) = 00 または 01 の場合

**Note:** このレジスタは、DUALBUF ビット (PMCON<17>) を「1」にセットした場合にのみ使います。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 25-10: PMRDIN: パラレルポート読み出し入力データレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RDATAIN<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RDATAIN<7:0>							

凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **RDATAIN<15:0>**: ポート読み出し入力データビット

**Note:** このレジスタは、DUALBUF ビット (PMCON<17>) を「1」にセットした場合にのみ、読み出し専用に使います。DUALBUF ビットが「0」の場合、PMRDIN の代わりに PMDIN レジスタ ( [レジスタ 25-5](#) ) を読み出し用に使います。

# PIC32MZ グラフィック (DA) ファミリ

## 26.0 外部バス インターフェイス (EBI)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 47. External Bus Interface (EBI)』を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] から入手できます。

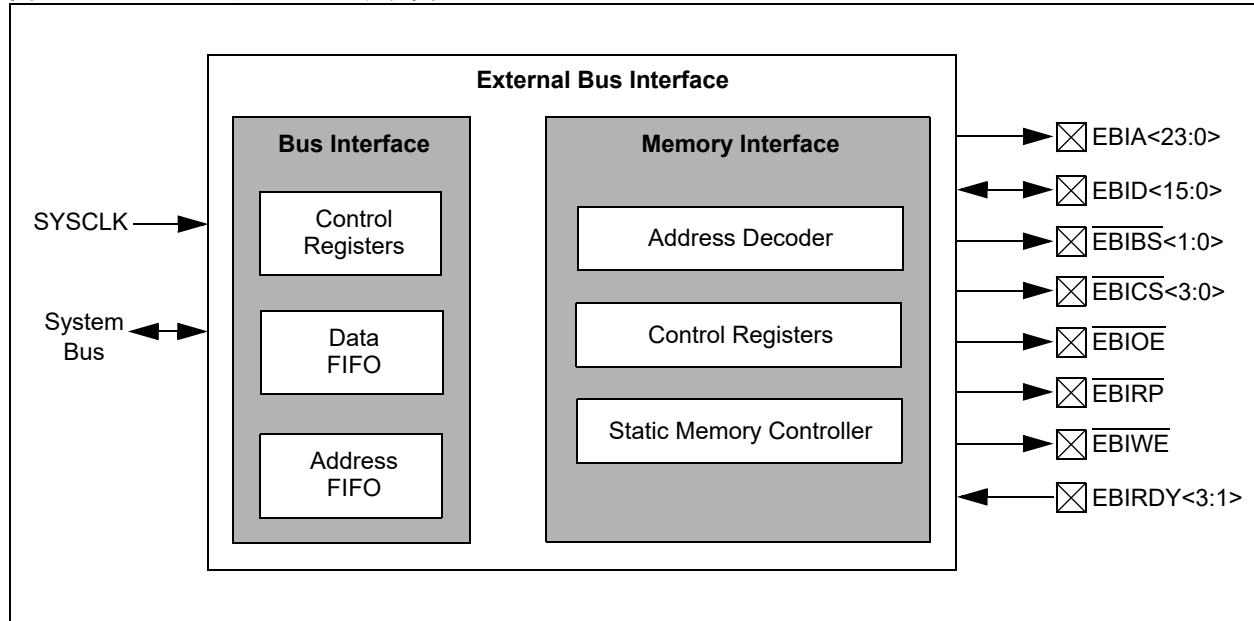
外部バス インターフェイス (EBI) モジュールは、外付けパラレルメモリ デバイスを PIC32MZ DA ファミリデバイスに接続するための高速で便利な方法を提供します。

EBI モジュールを使うと、非同期 SRAM および NOR フラッシュ デバイスを接続できる他、カメラセンサや LCD といったメモリ以外のデバイスも接続できます。

**Note 1:** EBI モジュールを設定する事で外部デバイスはメモリにマッピングされ、KSEG2 メモリ空間からアクセス可能になります (詳細は 4.0 「メモリ構成」内の図 4-1 ~ 図 4-2 を参照)。このメモリにアクセスするには、MMU を有効にし、TLB をセットアップする必要があります。詳細は『PIC32 Family Reference Manual, Section 50. CPU for Devices with MIPS32® microAptiv™ and M-Class Cores』(DS60001192) を参照してください。

**2:** EBI モジュールとグラフィック LCD (GLCD) は I/O の大部分を共有するため、EBI モジュールを使う場合、グラフィック LCD (GLCD) コントローラ機能は使えません。

図 26-1: EBI システムのブロック図



## 26.1 EBI 制御レジスタ

表 26-1: EBI のレジスタマップ

仮アドレ ス (BF8E_#)	レジスタ名	ビットレンジ	Bit															リセット値	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
1014	EBICS0	31:16	CSADDR<15:0>															2000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
1018	EBICS1	31:16	CSADDR<15:0>															1000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
101C	EBICS2	31:16	CSADDR<15:0>															2040	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
1020	EBICS3	31:16	CSADDR<15:0>															1040	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
1054	EBIMSK0	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
1058	EBIMSK1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0020
105C	EBIMSK2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0120
1060	EBIMSK3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0120
1094	EBISMT0	31:16	—	—	—	—	—	—	RDYMODE	PAGESIZE<1:0>	PAGEMODE	TPRC<3:0>			TBTA<2:0>			041C	
		15:0	TWP<5:0>					TWR<1:0>	TAS<1:0>	TRC<5:0>						2D4B			
1098	EBISMT1	31:16	—	—	—	—	—	—	RDYMODE	PAGESIZE<1:0>	PAGEMODE	TPRC<3:0>			TBTA<2:0>			041C	
		15:0	TWP<5:0>					TWR<1:0>	TAS<1:0>	TRC<5:0>						2D4B			
109C	EBISMT2	31:16	—	—	—	—	—	—	RDYMODE	PAGESIZE<1:0>	PAGEMODE	TPRC<3:0>			TBTA<2:0>			041C	
		15:0	TWP<5:0>					TWR<1:0>	TAS<1:0>	TRC<5:0>						2D4B			
10A0	EBIFTRPD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	TRPD<11:0>															00C8	
10A4	EBISMCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SMDWIDTH2<2:0>			SMDWIDTH1<2:0>			SMDWIDTH0<2:0>			—	—	—	—	—	—	—	SMRP 0201

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 26-1: EBICSx: 外部バス インターフェイス チップセレクト レジスタ (x = 0 ~ 3)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CSADDR<15:8>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CSADDR<7:0>							
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **CSADDR<15:0>**: デバイス ベースアドレス ビット

物理メモリ内のアドレスを格納します。このアドレスを使って外部デバイスを選択します。

bit 15-0 **未実装**: 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 26-2: EBIMSKx: 外部バス インターフェイス アドレスマスク レジスタ (x = 0 ~ 3)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	REGSEL<2:0>		
7:0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MEMTYPE<2:0>			MEMSIZE<4:0>				

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-11 **未実装:** 「0」として読み出し

bit 10-8 **REGSEL<2:0>:** チップセレクト「x」タイミング レジスタセット ビット

111 = 予約済み

⋮

011 = 予約済み

010 = EBISMT2 を使う

001 = EBISMT1 を使う

000 = EBISMT0 を使う

bit 7-5 **MEMTYPE<2:0>:** チップセレクト「x」メモリタイプ選択ビット

111 = 予約済み

⋮

011 = 予約済み

010 = NOR フラッシュ

001 = SRAM

000 = 予約済み

bit 4-0 **MEMSIZE<4:0>:** チップセレクト「x」メモリ容量選択ビット

11111 = 予約済み

⋮

01010 = 予約済み

01001 = 16 MB

01000 = 8 MB

00111 = 4 MB

00110 = 2 MB

00101 = 1 MB

00100 = 512 KB

00011 = 256 KB

00010 = 128 KB

00001 = 64 KB (64 KB よりも小さなメモリも 64 KB 領域に配置されます)

00000 = このチップセレクトを使わない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 26-3: EBISMTx: 外部バス インターフェイス スタティック メモリ タイミング レジスタ (x=0~2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	RDYMODE	PAGESIZE<1:0>	
23:16	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0
	PAGEMODE	TPRC<3:0> <sup>(1)</sup>				TBTA<2:0> <sup>(1)</sup>		
15:8	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-1
	TWP<5:0> <sup>(1)</sup>						TWR<1:0> <sup>(1)</sup>	
7:0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0
	TAS<1:0> <sup>(1)</sup>		TRC<5:0> <sup>(1)</sup>					

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-27 **未実装:** 「0」として読み出し

bit 26 **RDYMODE:** データレディ デバイス選択ビット

レジスタセット「x」に割り当てたデバイスがデータレディ デバイスかどうか (EBIRDYx ピンを使うかどうか) を選択します。

1 = EBIRDYx 入力を使う  
 0 = EBIRDYx 入力を使わない

bit 25-24 **PAGESIZE<1:0>:** ページサイズ ビット (ページモード対応デバイス用)

11 = 32 ワードページ  
 10 = 16 ワードページ  
 01 = 8 ワードページ  
 00 = 4 ワードページ

bit 23 **PAGEMODE:** メモリデバイス ページモード サポートビット

1 = デバイスはページモードをサポートする  
 0 = デバイスはページモードをサポートしない

bit 22-19 **TPRC<3:0>:** ページモード読み出しサイクル時間ビット<sup>(1)</sup>

読み出しサイクル時間 = TPRC + 1 (クロックサイクル)

bit 18-16 **TBTA<2:0>:** データバス ターンアラウンド時間ビット<sup>(1)</sup>

スタティック メモリの読み出し→書き込み、書き込み→読み出し、読み出し→読み出し (チップセレクト変更時) の間に挿入するクロックサイクル数 (0 ~ 7) を指定します。

bit 15-10 **TWP<5:0>:** 書き込みパルス幅ビット<sup>(1)</sup>

書き込みパルス幅 = TWP + 1 (クロックサイクル)

bit 9-8 **TWR<1:0>:** 書き込みアドレス/データ保持時間ビット<sup>(1)</sup>

バス上でアドレスまたはデータを保持する期間をクロックサイクル数で指定します。

bit 7-6 **TAS<1:0>:** 書き込みアドレス セットアップ時間ビット<sup>(1)</sup>

アドレス セットアップ期間をクロックサイクル数で指定します。値「0」は SSRAM の場合にのみ有効です。

bit 5-0 **TRC<5:0>:** 読み出しサイクル時間ビット<sup>(1)</sup>

読み出しサイクル時間 = TRC + 1 (クロックサイクル)

**Note 1:** EBI のタイミング図とその他の情報は、『PIC32 ファミリ リファレンス マニュアル、セクション 47. 外部バス インターフェイス (EBI)』(DS60001245) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 26-4: EBIFTRPD: 外部バス インターフェイス フラッシュ タイミング レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	TRPD<11:8>			
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TRPD<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

bit 31-12 **未実装**: 「0」として読み出し

bit 11-0 **TRPD<11:0>**: フラッシュ タイミングビット

これらのビットは、外部フラッシュメモリをリセット状態に保持する期間をクロックサイクル数で定義します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 26-5: EBISMCON: 外部バス インターフェイス スタティック メモリ制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0
	SMDWIDTH2<2:0>			SMDWIDTH1<2:0>			SMDWIDTH0<2:1>	
7:0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-1
	SMDWIDTH0<0>	—	—	—	—	—	—	SMRP

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15-13 **SMDWIDTH2<2:0>:** EBISM2 レジスタ スタティック メモリ幅ビット

- 111 = 予約済み
- 110 = 予約済み
- 101 = 予約済み
- 100 = 8 ビット
- 011 = 予約済み
- 010 = 予約済み
- 001 = 予約済み
- 000 = 16 ビット

bit 12-10 **SMDWIDTH1<2:0>:** EBISM1 レジスタ スタティック メモリ幅ビット

- 111 = 予約済み
- 110 = 予約済み
- 101 = 予約済み
- 100 = 8 ビット
- 011 = 予約済み
- 010 = 予約済み
- 001 = 予約済み
- 000 = 16 ビット

bit 9-7 **SMDWIDTH0<2:0>:** EBISM0 レジスタ スタティック メモリ幅ビット

- 111 = 予約済み
- 110 = 予約済み
- 101 = 予約済み
- 100 = 8 ビット
- 011 = 予約済み
- 010 = 予約済み
- 001 = 予約済み
- 000 = 16 ビット

bit 6-1 **未実装:** 「0」として読み出し

bit 0 **SMRP:** フラッシュ リセット / パワーダウン モード選択ビット

リセット後、コントローラは内部でフラッシュメモリのパワーダウンを実行した後に、このビットを「1」にセットします。

- 1 = フラッシュメモリのパワーダウン モードは完了した
- 0 = フラッシュメモリのパワーダウン モード中

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

## 27.0 暗号エンジン

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 49. Crypto Engine (CE) and Random Number Generator (RNG)』(DS60001246) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

暗号エンジンは、暗号機能が必要なアプリケーションを高速化する事を目的とします。これらの機能をハードウェア モジュールで実行する事でソフトウェア オーバーヘッドが減少し、暗号化、復号、認証等の実行速度が大幅に向上します。

暗号エンジンは、セキュリティ アソシエーション データとパケットポインタ (Scatter/Gather データフェッチが可能) を効率的にプログラミングするために、内部ディスクリプタ方式の DMA を使います。インテリジェントなステートマシンは、プロトコルの選択とパケット境界に基づいて暗号エンジンをスケジューリングします。本ハードウェア エンジンは暗号化と認証を順次または並列に実行できます。

暗号エンジンの主な特長は以下の通りです。

- バルク暗号化およびハッシュエンジン
- 処理負荷を軽減する内蔵 DMA
  - バッファ ディスクリプタ方式
  - バッファ ディスクリプタごとのセキュア アソシエーション
- 一部の機能を並列に実行可能

本暗号エンジンは以下のバルク暗号を処理可能です。

- AES:
  - 128 ビット、192 ビット、256 ビットの鍵サイズ
  - CBC、ECB、CTR、CFB、OFB モード
- DES/TDES:
  - CBC、ECB、CFB、OFB モード

本暗号エンジンでは以下の認証エンジンが使えます。

- SHA-1
- SHA-256
- MD-5
- AES-GCM
- HMAC 演算 (全ての認証エンジン向け)

暗号エンジンが処理可能なデータレートは、以下を含む各種要因によって決まります。

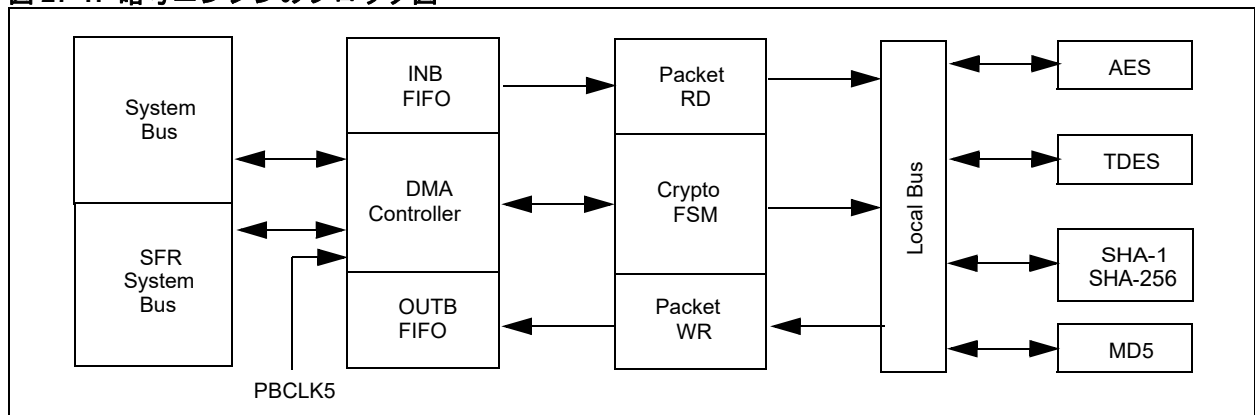
- どのエンジンを使うか
- 複数のエンジンを並列で使うのか直列で使うのか
- ソースおよびデスティネーション メモリに対するシステムの他の部分 (CPU、DMA 等) からの要求
- 暗号エンジンを駆動する PBCLK5 の速度

表 27-1 に、各種エンジンの代表的性能を示します。  
図 27-1 に、暗号エンジンのブロック図を示します。

表 27-1: 暗号エンジンの性能

エンジン / アルゴリズム	性能係数 (Mbps/MHz)	最大 Mbps (PBCLK5 = 100 MHz)
DES	14.4	1440
TDES	6.6	660
AES-128	9.0	900
AES-192	7.9	790
AES-256	7.2	720
MD5	15.6	1560
SHA-1	13.2	1320
SHA-256	9.3	930

図 27-1: 暗号エンジンのブロック図



## 27.1 暗号エンジン制御レジスタ

表 27-2: 暗号エンジンのレジスタマップ

仮アドレ ス (BF8E_#)	レジスタ名	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
5000	CEVER	31:16	REVISION<7:0>							VERSION<7:0>							0000		
		15:0	ID<15:0>														0000		
5004	CECON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	SWAPOEN	SWRST	SWAPEN	—	—	BDPCHST	BDPPLEN	DMAEN
5008	CEBDADDR	31:16	BDPADDR<31:0>														0000		
		15:0															0000		
500C	CEBDPADDR	31:16	BASEADDR<31:0>														0000		
		15:0															0000		
5010	CESTAT	31:16	ERRMODE<2:0>			ERROP<2:0>			ERRPHASE<1:0>		—	—	BDSTATE<3:0>			START	ACTIVE	0000	
		15:0	BDCTRL<15:0>														0000		
5014	CEINTSRC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	AREIF	PKTIF	CBDIF	PENDIF
5018	CEINTEN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	AREIE	PKTIE	CBDIE	PENDIE
501C	CEPOLLCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	BDPPLCON<15:0>														0000		
5020	CEHDLEN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	HDRLEN<7:0>							0000	
5024	CETRLLEN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	TRLRLEN<7:0>							0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 27-1: CEVER: 暗号エンジン リビジョン / バージョン / ID レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	REVISION<7:0>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	VERSION<7:0>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	ID<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	ID<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-24 **REVISION<7:0>**: 暗号エンジン リビジョンビット

bit 23-16 **VERSION<7:0>**: 暗号エンジン バージョンビット

bit 15-0 **ID<15:0>**: 暗号エンジン ID ビット

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 27-2: CECON: 暗号エンジン制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0, HC	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	SWAPOEN	SWRST	SWAPEN	—	—	BDPCHST	BDPPLEN	DMAEN

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-8 **未実装:** 「0」として読み出し

bit 7 **SWAPOEN:** スワップ出力データ イネーブルビット

- 1 = 専用 DMA による書き込み時に出力データをバイトスワップする
- 0 = 専用 DMA による書き込み時に出力データをバイトスワップしない

bit 6 **SWRST:** ソフトウェア リセットビット

- 1 = 暗号エンジンのソフトウェア リセットを開始する
- 0 = 通常動作

bit 5 **SWAPEN:** I/O スワップ イネーブルビット

- 1 = TFDMA 入力と RFDMA 出力をスワップする
- 0 = TFDMA 入力と RFDMA 出力をスワップしない

bit 4-3 **未実装:** 「0」として読み出し

bit 2 **BDPCHST:** バッファ ディスクリプタ プロセッサ (BDP) フェッチ イネーブルビット

- このビットは、全ての DMA ディスクリプタのプログラミングを完了した後で有効にする必要があります。
- 1 = BDP ディスクリプタ フェッチを有効にする
- 0 = BDP ディスクリプタ フェッチを無効にする

bit 1 **BDPPLEN:** バッファ ディスクリプタ プロセッサ ポーリング イネーブルビット

- このビットは、全ての DMA ディスクリプタのプログラミングを完了した後で有効にする必要があります。
- 1 = 有効ビットがセットされるまでディスクリプタをポーリングする
- 0 = ポーリングしない

bit 0 **DMAEN:** DMA イネーブルビット

- 1 = 暗号エンジン DMA を有効にする
- 0 = 暗号エンジン DMA を無効にする

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 27-3: CEBDADDR: 暗号エンジン バッファ ディスクリプタ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
BDPADDR<31:24>								
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
BDPADDR<23:16>								
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
BDPADDR<15:8>								
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
BDPADDR<7:0>								

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **BDPADDR<31:0>**: 処理中バッファ ディスクリプタ アドレス ステータスビット  
 これらのビットは、バッファ ディスクリプタ プロセッサ (BDP) が現在処理中のディスクリプタのアドレスを格納します。

レジスタ 27-4: CEBDPADDR: 暗号エンジン バッファ ディスクリプタ プロセッサ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BASEADDR<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BASEADDR<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BASEADDR<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BASEADDR<7:0>								

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **BASEADDR<31:0>**: DMA ベースアドレス ステータスビット  
 これらのビットは、DMA コントローラのベースアドレスを格納します。リセット後、フェッチはこのアドレスから始まります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 27-5: CESTAT: 暗号エンジン ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	ERRMODE<2:0>			ERROP<2:0>			ERRPHASE<1:0>	
23:16	U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
	—	—	BDSTATE				START	ACTIVE
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCTRL<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	BDCTRL<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-29 **ERRMOD<2:0>**: 内部エラーモード ステータスビット

- 111 = 予約済み
- 110 = 予約済み
- 101 = 予約済み
- 100 = 予約済み
- 011 = CEK 動作
- 010 = KEK 動作
- 001 = プリブート認証
- 000 = 通常動作

bit 28-26 **ERROP<2:0>**: 内部エラー動作ステータスビット

- 111 = 予約済み
- 110 = 予約済み
- 101 = 予約済み
- 100 = 認証
- 011 = 予約済み
- 010 = 復号
- 001 = 暗号化
- 000 = 予約済み

bit 25-24 **ERRPHASE<1:0>**: DMA 内部エラーフェイズ ステータスビット

- 11 = デスティネーション データ
- 10 = ソースデータ
- 01 = セキュリティ アソシエーション (SA) アクセス
- 00 = バッファ ディスクリプタ (BD) アクセス

bit 23-22 **未実装**: 「0」として読み出し

bit 21-18 **BDSTATE<3:0>**: バッファ ディスクリプタ プロセッサ ステート ステータスビット

これらのビットは、BDP の現在のステートを示す値を格納します。

- 1111 = 予約済み
- .
- .
- .
- 0111 = 予約済み
- 0110 = SA フェッチ
- 0101 = フェッチ BDP は無効
- 0100 = ディスクリプタは完了した
- 0011 = データフェイズ
- 0010 = BDP をロード中
- 0001 = ディスクリプタ フェッチ要求が保留中
- 0000 = BDP はアイドル状態

bit 17 **START**: DMA 開始ステータスビット

- 1 = DMA は開始された
- 0 = DMA は開始されていない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 27-5: CESTAT: 暗号エンジン ステータス レジスタ (続き)

bit 16 **ACTIVE:** バッファ ディスクリプタ プロセッサ ステータスビット  
1 = BDP はアクティブ  
0 = BDP はアイドル

bit 15-0 **BDCTRL<15:0>:** ディスクリプタ制御ワード ステータスビット  
これらのビットは、処理中ディスクリプタの制御ワードを格納します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 27-6: CEINTSRC: 暗号エンジン割り込み要因レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
	—	—	—	—	AREIF	PKTIF	CBDIF	PENDIF

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-4 **未実装:** 「0」として読み出し

bit 3 **AREIF:** アクセス応答エラー割り込みビット  
 1 = 暗号エンジンの領域外のメモリへのアクセスが試行された  
 0 = エラーは発生していない

bit 2 **PKTIF:** DMA パケット完了割り込みステータスビット  
 1 = DMA パケットが完了した  
 0 = DMA パケットは完了していない

bit 1 **CBDIF:** BD 送信ステータスビット  
 1 = 直近の BD 送信は処理された  
 0 = 直近の BD 送信はまだ処理されていない

bit 0 **PENDIF:** 暗号エンジン割り込み保留ステータスビット  
 1 = 未処理の暗号エンジン割り込みが存在する  
 このビットの値は、暗号エンジンの全ての割り込みステータスビットの論理和です。  
 0 = 未処理の暗号エンジン割り込みは存在しない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 27-7: CEINTEN: 暗号エンジン割り込みイネーブル レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	AREIE	PKTIE	BDPIE	PENDIE <sup>(1)</sup>

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-4 **未実装:** 「0」として読み出し

bit 3 **AREIE:** アクセス応答エラー割り込みイネーブルビット

- 1 = アクセス応答エラー割り込みを有効にする
- 0 = アクセス応答エラー割り込みを無効にする

bit 2 **PKTIE:** DMA パケット完了割り込みイネーブルビット

- 1 = DMA パケット完了割り込みを有効にする
- 0 = DMA パケット完了割り込みを無効にする

bit 1 **BDPIE:** DMA バッファ ディスクリプタ プロセッサ (BDP) 割り込みイネーブルビット

- 1 = BDP 割り込みを有効にする
- 0 = BDP 割り込みを無効にする

bit 0 **PENDIE:** マスタ割り込みイネーブルビット<sup>(1)</sup>

- 1 = 暗号エンジン割り込みを有効にする
- 0 = 暗号エンジン割り込みを無効にする

**Note 1:** PENDIE ビットはグローバル イネーブルビットであり、必要な個別の割り込みと一緒に有効にする必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 27-8: CEPOLLCON: 暗号エンジン ポーリング制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BDPPLCON<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BDPPLCON<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **BDPPLCON<15:0>:** バッファ ディスクリプタ プロセッサ (BDP) ポーリング制御ビット

直前にフェッチされたディスクリプタが無効になった場合、DMA 送信 BDP は、これらのビットが指定するサイクル数を待機した後に、ディスクリプタ制御ワードを再フェッチします。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 27-9: CEHDLEN: 暗号エンジン ヘッダ長レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	HDRLEN<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-8 **未実装:** 「0」として読み出し

bit 7-0 **HDRLEN<7:0>:** DMA ヘッダ長ビット

各パケットでは、このビットで指定した長さをスキップしてからデータの書き込みを始めます。

レジスタ 27-10: CETRLLEN: 暗号エンジン トレーラ長レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TRLRLEN<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-8 **未実装:** 「0」として読み出し

bit 7-0 **TRLRLEN<7:0>:** DMA トレーラ長ビット

各パケットでは、処理中パケットの最後でこのビットが指定する長さをスキップしてから次のパケットの書き込みを始めます。

# PIC32MZ グラフィック (DA) ファミリ

## 27.2 暗号エンジンバッファ ディスクリプタ

ホスト ソフトウェアはバッファ ディスクリプタのリンクリストを生成し、それらはハードウェアによって更新されます。表 27-3 に、暗号エンジンバッファ ディスクリプタの一覧を示し、図 27-2 ~ 図 27-10 に各バッファ ディスクリプタのフォーマットを示します。

表 27-3: 暗号エンジンのバッファ ディスクリプタ

名称 (Note 1 参照)	Bit 31:23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
BD_CTRL	31:24	DESC_EN	—	CRY_MODE<2:0>			—	—
	23:16	—	SA_FETCH_EN	—	—	LAST_BD	LIFM	PKT_INT_EN
	15:8	BD_BUFLEN<15:8>						
	7:0	BD_BUFLEN<7:0>						
BD_SA_ADDR	31:24	BD_SAADDR<31:24>						
	23:16	BD_SAADDR<23:16>						
	15:8	BD_SAADDR<15:8>						
	7:0	BD_SAADDR<7:0>						
BD_SCRADDR	31:24	BD_SRCADDR<31:24>						
	23:16	BD_SRCADDR<23:16>						
	15:8	BD_SRCADDR<15:8>						
	7:0	BD_SRCADDR<7:0>						
BD_DSTADDR	31:24	BD_DSTADDR<31:24>						
	23:16	BD_DSTADDR<23:16>						
	15:8	BD_DSTADDR<15:8>						
	7:0	BD_DSTADDR<7:0>						
BD_NXTPTR	31:24	BD_NXTADDR<31:24>						
	23:16	BD_NXTADDR<23:16>						
	15:8	BD_NXTADDR<15:8>						
	7:0	BD_NXTADDR<7:0>						
BD_UPDPTR	31:24	BD_UPDADDR<31:24>						
	23:16	BD_UPDADDR<23:16>						
	15:8	BD_UPDADDR<15:8>						
	7:0	BD_UPDADDR<7:0>						
BD_MSG_LEN	31:24	MSG_LENGTH<31:24>						
	23:16	MSG_LENGTH<23:16>						
	15:8	MSG_LENGTH<15:8>						
	7:0	MSG_LENGTH<7:0>						
BD_ENC_OFF	31:24	ENCR_OFFSET<31:24>						
	23:16	ENCR_OFFSET<23:16>						
	15:8	ENCR_OFFSET<15:8>						
	7:0	ENCR_OFFSET<7:0>						

Note 1: 各バッファ ディスクリプタは、メモリ内の 64 ビット境界に配置する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

図 27-2: BD\_CTRL のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/ 13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/ 10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	DESC_EN	—	CRY_MODE<2:0>			—	—	—
23-16	—	SA_FETCH_EN	—	—	LAST_BD	LIFM	PKT_INT_EN	CBD_INT_EN
15-8	BD_BUFLen<15:8>							
7-0	BD_BUFLen<7:0>							

- bit 31 **DESC\_EN:** ディスクリプタ イネーブル  
 1 = ハードウェアがディスクリプタを所有する  
 BD の処理後に、ハードウェアがこのビットを「0」にリセットします。  
 0 = ソフトウェアがディスクリプタを所有する
- bit 30 **未実装:** 常に「0」として書き込み
- bit 29-27 **CRY\_MODE<2:0>:** 暗号モード  
 111 = 予約済み  
 110 = 予約済み  
 101 = 予約済み  
 100 = 予約済み  
 011 = CEK 動作  
 010 = KEK 動作  
 001 = プリブート認証  
 000 = 通常動作
- bit 22 **SA\_FETCH\_EN:** 外部メモリからのセキュリティ アソシエーション (SA) のフェッチ  
 1 = SA ポインタから SA をフェッチする  
 このビットは、新しいパケットごとに「1」にセットする必要があります。  
 0 = 現在フェッチ中の SA または内部 SS を使う
- bit 21-20 **未実装:** 常に「0」として書き込み
- bit 19 **LAST\_BD:** 最終バッファ ディスクリプタ  
 1 = チェーン内に最後のバッファ ディスクリプタだけが存在する  
 0 = チェーン内に複数のバッファ ディスクリプタが存在する  
 最後の BD の後に、CEBDADDR は CEBDPADDR 内のベースアドレスへ移動します。
- bit 18 **LIFM:** フレーム内最終  
 受信パケット (ハードウェア → ホスト) の場合、このフィールドはハードウェアによって書き込まれ、パケットが複数のバッファ ディスクリプタにまたがるかどうかを示します。送信パケット (ホスト → ハードウェア) の場合、このフィールドはこの BD がフレーム内の最後の BD かどうかを示します。
- bit 17 **PKT\_INT\_EN:** パケット割り込みイネーブル  
 パケットの最後のバッファ ディスクリプタの処理が完了した後に割り込みを生成します。
- bit 16 **CBD\_INT\_EN:** CBD 割り込みイネーブル  
 現在のバッファ ディスクリプタの処理が完了した後に割り込みを生成します。
- bit 15-0 **BD\_BUFLen<15:0>:** バッファ ディスクリプタ長  
 このフィールドはバッファの長さを格納します。この値は、レシーバが実際に書き込んだ長さに更新されます。

# PIC32MZ グラフィック (DA) ファミリ

図 27-3: BD\_SADDR のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	BD_SAADDR<31:24>							
23-16	BD_SAADDR<23:16>							
15-8	BD_SAADDR<15:8>							
7-0	BD_SAADDR<7:0>							

bit 31-0 **BD\_SAADDR<31:0>**: セキュリティ アソシエーション IP セッション アドレス  
セッションの SA ポインタは鍵と IV 値を持ちます。

図 27-4: BD\_SADDR のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	BD_SAADDR<31:24>							
23-16	BD_SAADDR<23:16>							
15-8	BD_SAADDR<15:8>							
7-0	BD_SAADDR<7:0>							

bit 31-0 **BD\_SAADDR<31:0>**: セキュリティ アソシエーション IP セッション アドレス  
セッションの SA ポインタは鍵と IV 値を持ちます。

図 27-5: BD\_SRCADDR のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	BD_SCRADDR<31:24>							
23-16	BD_SCRADDR<23:16>							
15-8	BD_SCRADDR<15:8>							
7-0	BD_SCRADDR<7:0>							

bit 31-0 **BD\_SCRADDR**: バッファ ソースアドレス  
暗号化または認証向けに PE-CRDMA を介して引き渡す必要のあるバッファのソースアドレスこの  
アドレスは 32 ビット境界である事が必要です。

# PIC32MZ グラフィック (DA) ファミリ

図 27-6: BD\_DSTADDR のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	BD_DSTADDR<31:24>							
23-16	BD_DSTADDR<23:16>							
15-8	BD_DSTADDR<15:8>							
7-0	BD_DSTADDR<7:0>							

bit 31-0 **BD\_DSTADDR**: バッファ デスティネーション アドレス  
暗号化または認証向けに PE-CRDMA を介して引き渡す必要のあるバッファのデスティネーション アドレス。このアドレスは 32 ビット境界である必要があります。

図 27-7: BD\_NXTADDR のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	BD_NXTADDR<31:24>							
23-16	BD_NXTADDR<23:16>							
15-8	BD_NXTADDR<15:8>							
7-0	BD_NXTADDR<7:0>							

bit 31-0 **BD\_NXTADDR**: 次の BD ポインタアドレス  
次のバッファ ディスクリプタのアドレスを格納します。  
次のバッファは、直前のバッファの次のセグメントまたは新しいパケットです。

図 27-8: BD\_UPDPTR のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	BD_UPDADDR<31:24>							
23-16	BD_UPDADDR<23:16>							
15-8	BD_UPDADDR<15:8>							
7-0	BD_UPDADDR<7:0>							

bit 31-0 **BD\_UPDADDR**: 更新アドレス  
更新アドレスは CRDMA 結果が書き込まれるアドレスです。更新結果は必要に応じて ICV 値または鍵出力値のどちらかです。

# PIC32MZ グラフィック (DA) ファミリ

図 27-9: BD\_MSG\_LEN のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	MSG_LENGTH<31:24>							
23-16	MSG_LENGTH<23:16>							
15-8	MSG_LENGTH<15:8>							
7-0	MSG_LENGTH<7:0>							

bit 31-0 **MSG\_LENGTH**: 総メッセージ長

ハッシュおよび HMAC アルゴリズム場合、これは総メッセージ長 (バイト数) です。GCM アルゴリズム (LEN-C) の場合、これは暗号バイトの総数です。

図 27-10: BD\_ENC\_OFF のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	ENCR_OFFSET<31:24>							
23-16	ENCR_OFFSET<23:16>							
15-8	ENCR_OFFSET<15:8>							
7-0	ENCR_OFFSET<7:0>							

bit 31-0 **ENCR\_OFFSET**: 暗号オフセット

マルチタスク テスト (暗号化および認証向け) の場合、これは暗号オフセットです。GMC アルゴリズム (LEN-A) の場合、これは AAD バイトの数です。

# PIC32MZ グラフィック (DA) ファミリ

## 27.3 セキュリティ アソシエーションの構造

表 27-11 に、セキュリティ アソシエーションの構造を示します。

暗号エンジンは、セキュリティ アソシエーションを使って、バッファ ディスクリプタ プロセッサを処理するための設定を定義します。セキュリティ アソシエーションは以下を定義します。

- 使用するアルゴリズム
- 暗号エンジンを認証と暗号化 / 復号の両方に並列で使うかどうか
- 鍵のサイズ
- 認証鍵
- 暗号化 / 復号鍵
- 認証初期化ベクタ (IV)
- 暗号化初期化ベクタ (IV)

図 27-11: 暗号エンジン セキュリティ アソシエーションの構造

名称	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	
SA_CTRL	31:24	—	—	VERIFY	—	NO_RX	OR_EN	ICVONLY	
	23:16	LNC	LOADIV	FB	FLAGS	—	—	ALGO<6>	
	15:8	ALGO<5:0>						ENCTYPE	KEYSIZE<1>
	7:0	KEYSIZE<0>	MULTITASK<2:0>			CRYPTOALGO<3:0>			
SA_AUTHKEY1	31:24	AUTHKEY<31:24>							
	23:16	AUTHKEY<23:16>							
	15:8	AUTHKEY<15:8>							
	7:0	AUTHKEY<7:0>							
SA_AUTHKEY2	31:24	AUTHKEY<31:24>							
	23:16	AUTHKEY<23:16>							
	15:8	AUTHKEY<15:8>							
	7:0	AUTHKEY<7:0>							
SA_AUTHKEY3	31:24	AUTHKEY<31:24>							
	23:16	AUTHKEY<23:16>							
	15:8	AUTHKEY<15:8>							
	7:0	AUTHKEY<7:0>							
SA_AUTHKEY4	31:24	AUTHKEY<31:24>							
	23:16	AUTHKEY<23:16>							
	15:8	AUTHKEY<15:8>							
	7:0	AUTHKEY<7:0>							
SA_AUTHKEY5	31:24	AUTHKEY<31:24>							
	23:16	AUTHKEY<23:16>							
	15:8	AUTHKEY<15:8>							
	7:0	AUTHKEY<7:0>							
SA_AUTHKEY6	31:24	AUTHKEY<31:24>							
	23:16	AUTHKEY<23:16>							
	15:8	AUTHKEY<15:8>							
	7:0	AUTHKEY<7:0>							
SA_AUTHKEY7	31:24	AUTHKEY<31:24>							
	23:16	AUTHKEY<23:16>							
	15:8	AUTHKEY<15:8>							
	7:0	AUTHKEY<7:0>							
SA_AUTHKEY8	31:24	AUTHKEY<31:24>							
	23:16	AUTHKEY<23:16>							
	15:8	AUTHKEY<15:8>							
	7:0	AUTHKEY<7:0>							
SA_ENCKEY1	31:24	ENCKEY<31:24>							
	23:16	ENCKEY<23:16>							
	15:8	ENCKEY<15:8>							
	7:0	ENCKEY<7:0>							
SA_ENCKEY2	31:24	ENCKEY<31:24>							

# PIC32MZ グラフィック (DA) ファミリ

図 27-11: 暗号エンジン セキュリティ アソシエーションの構造 (続き)

名称	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
	23:16							ENCKEY<23:16>
	15:8							ENCKEY<15:8>
	7:0							ENCKEY<7:0>
SA_ENCKEY3	31:24							ENCKEY<31:24>
	23:16							ENCKEY<23:16>
	15:8							ENCKEY<15:8>
	7:0							ENCKEY<7:0>
	31:24							ENCKEY<31:24>
	23:16							ENCKEY<23:16>
SA_ENCKEY4	15:8							ENCKEY<15:8>
	7:0							ENCKEY<7:0>
	31:24							ENCKEY<31:24>
SA_ENCKEY5	23:16							ENCKEY<23:16>
	15:8							ENCKEY<15:8>
	7:0							ENCKEY<7:0>
SA_ENCKEY6	31:24							ENCKEY<31:24>
	23:16							ENCKEY<23:16>
	15:8							ENCKEY<15:8>
	7:0							ENCKEY<7:0>
	31:24							ENCKEY<31:24>
	23:16							ENCKEY<23:16>
SA_ENCKEY7	15:8							ENCKEY<15:8>
	7:0							ENCKEY<7:0>
	31:24							ENCKEY<31:24>
SA_ENCKEY8	23:16							ENCKEY<23:16>
	15:8							ENCKEY<15:8>
	7:0							ENCKEY<7:0>
SA_AUTHIV1	31:24							AUTHIV<31:24>
	23:16							AUTHIV<23:16>
	15:8							AUTHIV<15:8>
	7:0							AUTHIV<7:0>
	31:24							AUTHIV<31:24>
	23:16							AUTHIV<23:16>
SA_AUTHIV2	15:8							AUTHIV<15:8>
	7:0							AUTHIV<7:0>
	31:24							AUTHIV<31:24>
SA_AUTHIV3	23:16							AUTHIV<23:16>
	15:8							AUTHIV<15:8>
	7:0							AUTHIV<7:0>
SA_AUTHIV4	31:24							AUTHIV<31:24>
	23:16							AUTHIV<23:16>
	15:8							AUTHIV<15:8>
	7:0							AUTHIV<7:0>
	31:24							AUTHIV<31:24>
	23:16							AUTHIV<23:16>
SA_AUTHIV5	15:8							AUTHIV<15:8>
	7:0							AUTHIV<7:0>
	31:24							AUTHIV<31:24>
SA_AUTHIV6	23:16							AUTHIV<23:16>
	15:8							AUTHIV<15:8>
	7:0							AUTHIV<7:0>
SA_AUTHIV7	31:24							AUTHIV<31:24>
	23:16							AUTHIV<23:16>
	15:8							AUTHIV<15:8>
	7:0							AUTHIV<7:0>
	31:24							AUTHIV<31:24>
	23:16							AUTHIV<23:16>
SA_AUTHIV8	15:8							AUTHIV<15:8>
	7:0							AUTHIV<7:0>



# PIC32MZ グラフィック (DA) ファミリ

図 27-11: 暗号エンジンセキュリティ アソシエーションの構造 (続き)

名称		Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
SA_ENCIV1	31:24	ENCIV<31:24>							
	23:16	ENCIV<23:16>							
	15:8	ENCIV<15:8>							
	7:0	ENCIV<7:0>							
SA_ENCIV2	31:24	ENCIV<31:24>							
	23:16	ENCIV<23:16>							
	15:8	ENCIV<15:8>							
	7:0	ENCIV<7:0>							
SA_ENCIV3	31:24	ENCIV<31:24>							
	23:16	ENCIV<23:16>							
	15:8	ENCIV<15:8>							
	7:0	ENCIV<7:0>							
SA_ENCIV4	31:24	ENCIV<31:24>							
	23:16	ENCIV<23:16>							
	15:8	ENCIV<15:8>							
	7:0	ENCIV<7:0>							

# PIC32MZ グラフィック (DA) ファミリ

表 27-12 に、セキュリティ アソシエーション (SA) 制御ワードの構造を示します。

暗号エンジンはフローに応じて異なる構造をフェッチします。これにより、ハードウェアは処理に必要な最小限のワードだけを SA からフェッチします。この構造は、ハードウェアが最適にデータをフェッチできるよう準備されています。

図 27-12: SA\_CTRL のフォーマット

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31-24	—	—	VERIFY	—	NO_RX	OR_EN	ICVONLY	IRFLAG
23-16	LNC	LOADIV	FB	FLAGS	—	—	—	ALGO<6>
15-8	ALGO<5:0>						ENC	KEY SIZE<1>
7-0	KEY SIZE<0>	MULTITASK<2:0>			CRYPTOALGO<3:0>			

bit 31-30 **予約済み**：使用不可

bit 29 **VERIFY**: NIST 手順検証設定  
1 = NIST 手順を使う  
0 = NIST 手順を使わない

bit 28 **予約済み**：使用不可

bit 27 **NO\_RX**: 受信 DMA 制御設定  
1 = 認証計算用に ICV だけを計算する  
0 = 通常処理

bit 26 **OR\_EN**: レジスタビット論理和イネーブル設定  
1 = レジスタビットと CSR レジスタ内の値の論理和を取る  
0 = 通常処理

bit 25 **ICVONLY**: 未完了チェック値限定フラグ  
このビットは SHA-1 アルゴリズムにのみ影響します。AES アルゴリズムには影響しません。  
1 = HMAC 結果の 3 ワードだけが利用可能  
0 = HMAC 結果の全てが利用可能

bit 24 **IRFLAG**: ハッシュの即時結果設定  
このビットは、ハッシュの即時結果が必要な場合にセットします。  
1 = ハッシュの即時結果を保存する  
0 = 即時結果を保存しない

bit 23 **LNC**: 新鍵ロード設定  
1 = 暗号化と認証向けに新しい鍵のセットをロードする  
0 = 新しい鍵をロードしない

bit 22 **LOADIV**: IV ロード設定  
1 = このセキュリティ アソシエーションから IV をロードする  
0 = 次の IV を使う

bit 21 **FB**: 先頭ブロック設定  
このビットは、このブロックが IV 値を与えるデータの最初のブロックであることを示します。  
1 = これはデータの先頭ブロックである  
0 = これはデータの先頭ブロックではない

bit 20 **FLAGS**: 受信 / 送信フロー設定  
1 = セキュリティ アソシエーションは送信フローに関連する  
0 = セキュリティ アソシエーションは受信フローに関連する

bit 19-17 **予約済み**：使用不可

# PIC32MZ グラフィック (DA) ファミリ

図 27-12: SA\_CTRL のフォーマット ( 続き )

bit 16-10	<b>ALGO&lt;6:0&gt;</b> : 使用するアルゴリズムのタイプ 1xxxxxxx = HMAC 1 x1xxxxxx = SHA-256 xx1xxxxx = SHA1 xxx1xxxx = MD5 xxxx1xxx = AES xxxxx1xx = TDES xxxxxxx1 = DES
bit 9	<b>ENC</b> : 暗号処理タイプ設定 1 = 暗号化 0 = 復号
bit 8-7	<b>KEYSIZE&lt;1:0&gt;</b> : SA_AUTHKEYx または SA_ENCKEYx 内の鍵のサイズ 11 = 予約済み ( 使用禁止 ) 10 = 256 ビット 01 = 192 ビット 00 = 128 ビット (1)
bit 6-4	<b>MULTITASK&lt;2:0&gt;</b> : 暗号エンジンでの並列動作の組み合わせ方法 111 = 並列パス ( 受信データの復号と認証を並列に実行する ) 101 = パイプパス ( 受信データを暗号化した後に、暗号化済みのデータに対して認証を実行する ) 011 = 予約済み 010 = 予約済み 001 = 予約済み 000 = パスなし ( 暗号化、認証、復号のいずれかを実行する )
bit 3-0	<b>CRYPTOALGO&lt;3:0&gt;</b> : 暗号アルゴリズムの動作モード 1111 = 予約済み 1110 = AES_GCM (AES 処理向け) 1101 = RCTR (AES 処理向け) 1100 = RCBC_MAC (AES 処理向け) 1011 = ROFB (AES 処理向け) 1010 = RCFB (AES 処理向け) 1001 = RCBC (AES 処理向け) 1000 = REBC (AES 処理向け) 0111 = TOFB (トリプル DES 処理向け) 0110 = TCFB (トリプル DES 処理向け) 0101 = TCBC (トリプル DES 処理向け) 0100 = TECB (トリプル DES 処理向け) 0011 = OFB (DES 処理向け) 0010 = CFB (DES 処理向け) 0001 = CBC (DES 処理向け) 0000 = ECB (DES 処理向け)

**Note 1:** この設定は、セキュリティ アソシエーション内の SA\_AUTHKEYx または SA\_ENCKEYx のサイズを変更せず、使用する SA\_AUTHKEYx および SA\_ENCKEYx のビット数だけを変更します。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

## 28.0 乱数生成器 (RNG)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 49. Crypto Engine (CE) and Random Number Generator (RNG)』(DS60001246) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

乱数生成器 (RNG) コアは、熱雑音を用いる真性乱数生成器 (TRNG) と、暗号的に安全な擬似乱数生成器 (PRNG) を実装しています。

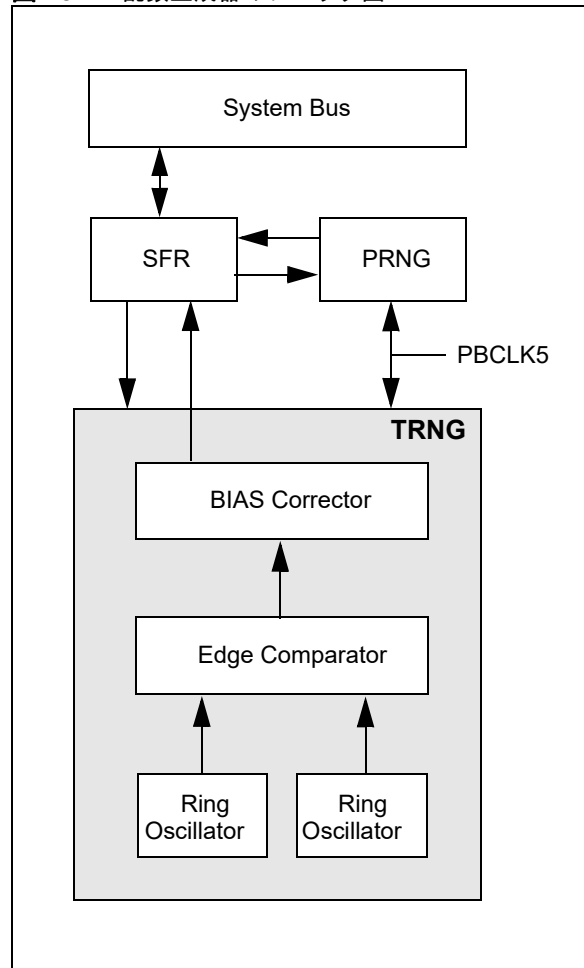
TRNG は、複数のリングオシレータと、集積回路で発生する熱雑音を使って、真性乱数を生成します。この乱数は PRNG の初期化用に使えます。

PRNG は最大長 64 ビットの柔軟な LSFR です。

乱数生成器の主な特長は以下の通りです。

- TRNG:
  - 最大 25 Mbps のランダムビット
  - マルチ リングオシレータ構造
  - バイアス補正器を内蔵
- PRNG:
  - LSFR 方式
  - 最大 64 ビットの多項式長
  - 多項式はプログラマブル
  - TRNG が生成した乱数をシード値として使用可能

図 28-1: 乱数生成器のブロック図



## 28.1 RNG 制御レジスタ

表 28-1: 乱数生成器 (RNG) のレジスタマップ

依拠アドレス (BF8E_#)	レジスタ名	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
6000	RNGVER	31:16	ID<15:0>																XXXXX
		15:0	VERSION<7:0>								REVISION<7:0>								XXXXX
6004	RNGCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	LOAD	TRNGMODE	CONT	PRNGEN	TRNGEN	PLEN<7:0>								0064
6008	RNGPOLY1	31:16	POLY<31:0>																FFFF
		15:0																	0000
600C	RNGPOLY2	31:16	POLY<31:0>																FFFF
		15:0																	0000
6010	RNGNUMGEN1	31:16	RNG<31:0>																FFFF
		15:0																	FFFF
6014	RNGNUMGEN2	31:16	RNG<31:0>																FFFF
		15:0																	FFFF
6018	RNGSEED1	31:16	SEED<31:0>																0000
		15:0																	0000
601C	RNGSEED2	31:16	SEED<31:0>																0000
		15:0																	0000
6020	RNGCNT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	RCNT<6:0>						0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 28-1: RNGVER: 乱数生成器バージョンレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	ID<15:8>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	ID<7:0>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	VERSION<7:0>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	REVISION<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-6 **ID<15:0>**: ブロック ID ビット

bit 15-8 **VERSION<7:0>**: ブロック バージョンビット

bit 7-0 **REVISION<7:0>**: ブロック リビジョンビット

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 28-2: RNGCON: 乱数生成器制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	LOAD	TRNGMODE <sup>(1)</sup>	CONT	PRNGEN	TRNGEN
7:0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0
	PLEN<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-13 **未実装:** 「0」として読み出し

bit bit 12 **LOAD:** デバイス選択ビット

このビットは、TRNG からの値 (すなわち乱数) をシード値として PRNG に書き込むために使います。

bit 11 **TRNGMODE:** TRNG モードビット<sup>(1)</sup>

1 = 拡張 TRNG モードを選択する  
 0 = 通常の TRNG モードを選択する

bit 10 **CONT:** 周期的 PRNG 乱数生成イネーブルビット

1 = 周期的に PRNG 乱数を生成する  
 0 = 直前の乱数が読み出された時に次の PRNG 乱数を生成する

bit 9 **PRNGEN:** PRNG 動作イネーブルビット

1 = PRNG 動作を有効にする  
 0 = PRNG 動作を無効にする

bit 8 **TRNGEN:** TRNG 動作イネーブルビット

1 = TRNG 動作を有効にする  
 0 = TRNG 動作を無効にする

bit 7-0 **PLEN<7:0>:** PRNG 多項式長ビット

これらのビットは、PRNG で使う多項式の長さを格納します。

**Note 1:** このビットは、TRNGEN ビットを「1」にセットした場合にのみ効力を有します。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 28-3: RNGPOLYx: 乱数生成器多項式レジスタ「x」 (x = 1 または 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	POLY<31:24>							
23:16	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	POLY<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	POLY<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	POLY<7:0>							

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-0    **POLY<31:0>**: PRNG LFSR 多項式 MSb/LSb ビット (RNGPOLY1 = LSb、RNGPOLY2 = MSb)

## レジスタ 28-4: RNGNUMGENx: 乱数生成器レジスタ「x」 (x = 1 または 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	RNG<31:24>							
23:16	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	RNG<23:16>							
15:8	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	RNG<15:8>							
7:0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	RNG<7:0>							

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-0    **RNG<31:0>**: 現在の PRNG MSb/LSb 値ビット (RNGNUMGEN1 = LSb、RNGNUMGEN2 = MSb)

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 28-5: RNGSEEDx: TRNG シード値レジスタ「x」(x = 1 または 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
SEED<31:24>								
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
SEED<23:16>								
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
SEED<15:8>								
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
SEED<7:0>								

**凡例:**

R = 読み出し可能ビット                  W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値                          1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-0 **SEED<31:0>**: TRNG MSb/LSb 値ビット (RNGSEED1 = LSb、RNGSEED2 = MSb)

レジスタ 28-6: RNGCNT: TRNG カウントレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—								
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—								
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—								
7:0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RCNT<6:0>								

**凡例:**

R = 読み出し可能ビット                  W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値                          1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-7 **未実装**: 「0」として読み出し

bit 6-0 **RCNT<6:0>**: TRNG が生成した有効な乱数のビット数

# PIC32MZ グラフィック (DA) ファミリ

## 29.0 高速 12 ビット逐次比較型レジスタ (SAR) A/D コンバータ (ADC)

**Note:** 本書は PIC32MZ DA の機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 22. 12-bit High-Speed Successive Approximation Register (SAR) Analog-to-Digital Converter (ADC)』(DS60001344) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/PIC32](http://www.microchip.com/PIC32)) から入手できます。

高速 12 ビット逐次比較型レジスタ (SAR) A/D コンバータ (ADC) の主な特長は以下の通りです。

- 12 ビット分解能
- 専用のサンプル / ホールド (S&H) 回路を備えた 6 個の ADC モジュール
- 2 個の専用 ADC モジュールを結合して変換レートを倍増できるターボモード (2 つの ADC 間でクロック源の同期が必要)
- 最大 45 個のアナログ入力に加えて内部 CTMU、VBAT、内部参照電圧、内部温度センサを入力源として選択可能
- シングルエンド入力と差動入力が選択可能
- スリープ中も動作可能
- タッチセンシング アプリケーションをサポート
- 6 個のデジタル コンパレータ
- 6 個のデジタルフィルタ (以下のモードをサポート):
  - オーバー サンプリング モード
  - 平均化モード
- 16 ワード FIFO によるスループットの向上 (ADC0 ~ ADC4)
- 変換データの処理を高速化する早期割り込み生成
- モータ制御、電力変換、汎用アプリケーション向けの設計
- スリープおよびアイドル中の動作

図 29-1 に、ADC モジュールの概略ブロック図を示します。

12 ビット HS SAR ADC は最大 5 個の専用 ADC モジュール (ADC0 ~ ADC4) と、共有 ADC モジュール (ADC7) を内蔵します。各専用 ADC モジュールは 1 つのアナログ入力 (またはその代替入力) を使い、時間的制約のある入力や過渡入力的高速 / 高精度サンプリング向けに使えます。共有 ADC モジュールは入力にマルチプレクサを備え、各種の入力をより柔軟に選択できます。共有 ADC のサンプリングは専用 ADC よりも低速ですが、入力スキャンロジックによる自動スキャン機能が使えます。

各 ADC モジュールのアナログ入力は S&H コンデンサに接続しています。各 ADC モジュールのクロック、サンプリング時間、出力データ分解能は別々に設定できます。ADC モジュールは、レジスタの設定に基づいて入力アナログ信号の変換を実行します。変換が完了すると、変換結果は各アナログ入力の結果バッファに保存されます。デジタルフィルタとデジタル コンパレータがその入力からのデータを処理するよう設定されている場合、データはそれらに渡されます。各 ADC に対する入力の割り当てを図 29-2 に示します。

### 式 29-1: ADC のスループット レート

$$FTP = \frac{T_{AD}}{(T_{SAMP} + T_{CONV})}$$

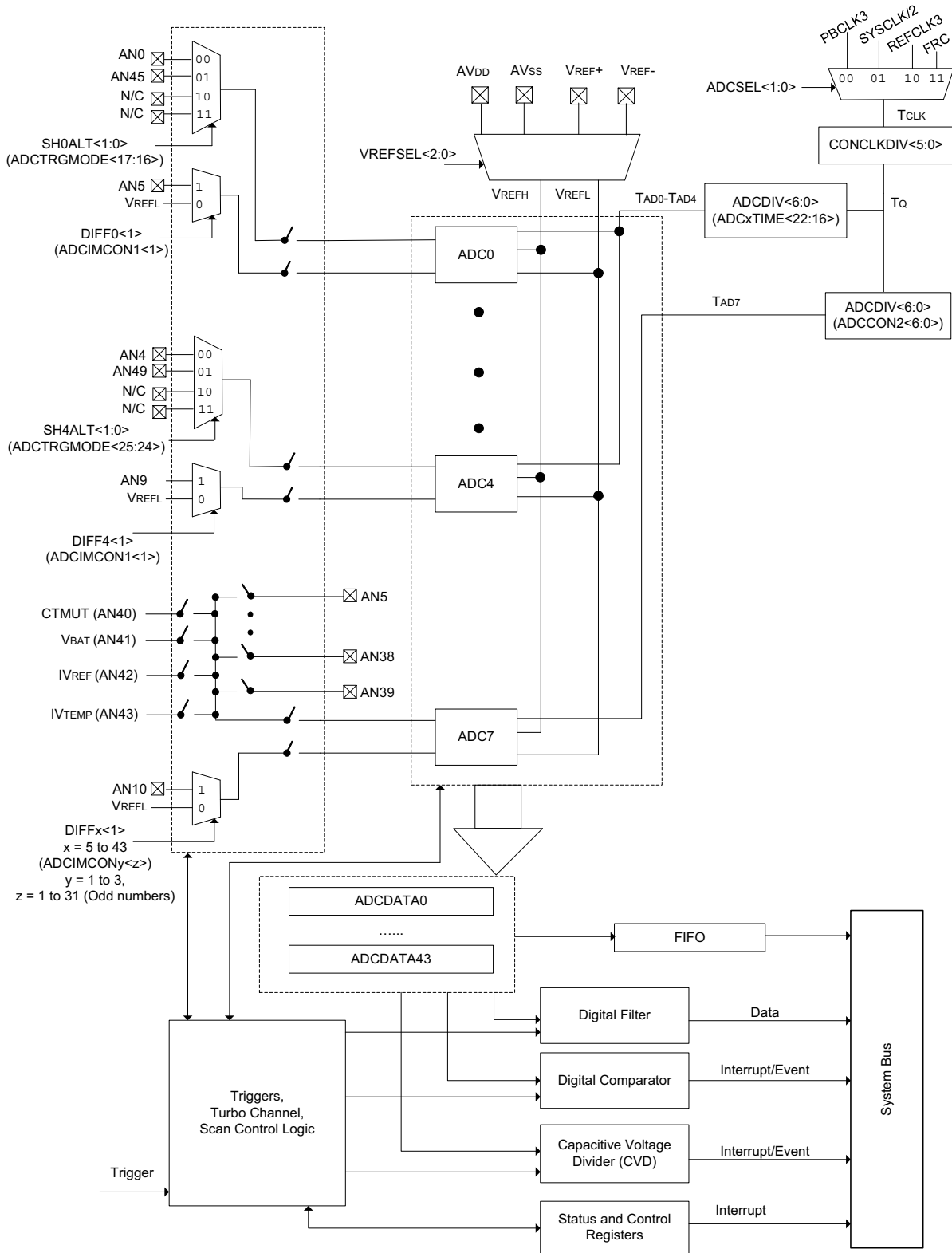
$T_{AD}$  = 各 ADC モジュールの周波数

**Note 1:** ADC モジュールを有効にする前に、ユーザ アプリケーションはコンフィグレーション メモリ内の ADC 校正データ (DEVADC0 ~ DEVADC4、DEVADC7、レジスタ 41-8 参照) を ADC コンフィグレーション レジスタ (ADC0CFG ~ ADC4CFG、ADC7CFG) にコピーする必要があります。

**2:** VDDIO が 2.5 V より高い場合、AICMPEN ビット (ADCCON1<12>) と IOANCPEN ビット (CFGCON<7>) を「0」に設定します。VDDIO が 2.5 V より低い場合、両方のビットを「1」に設定します。

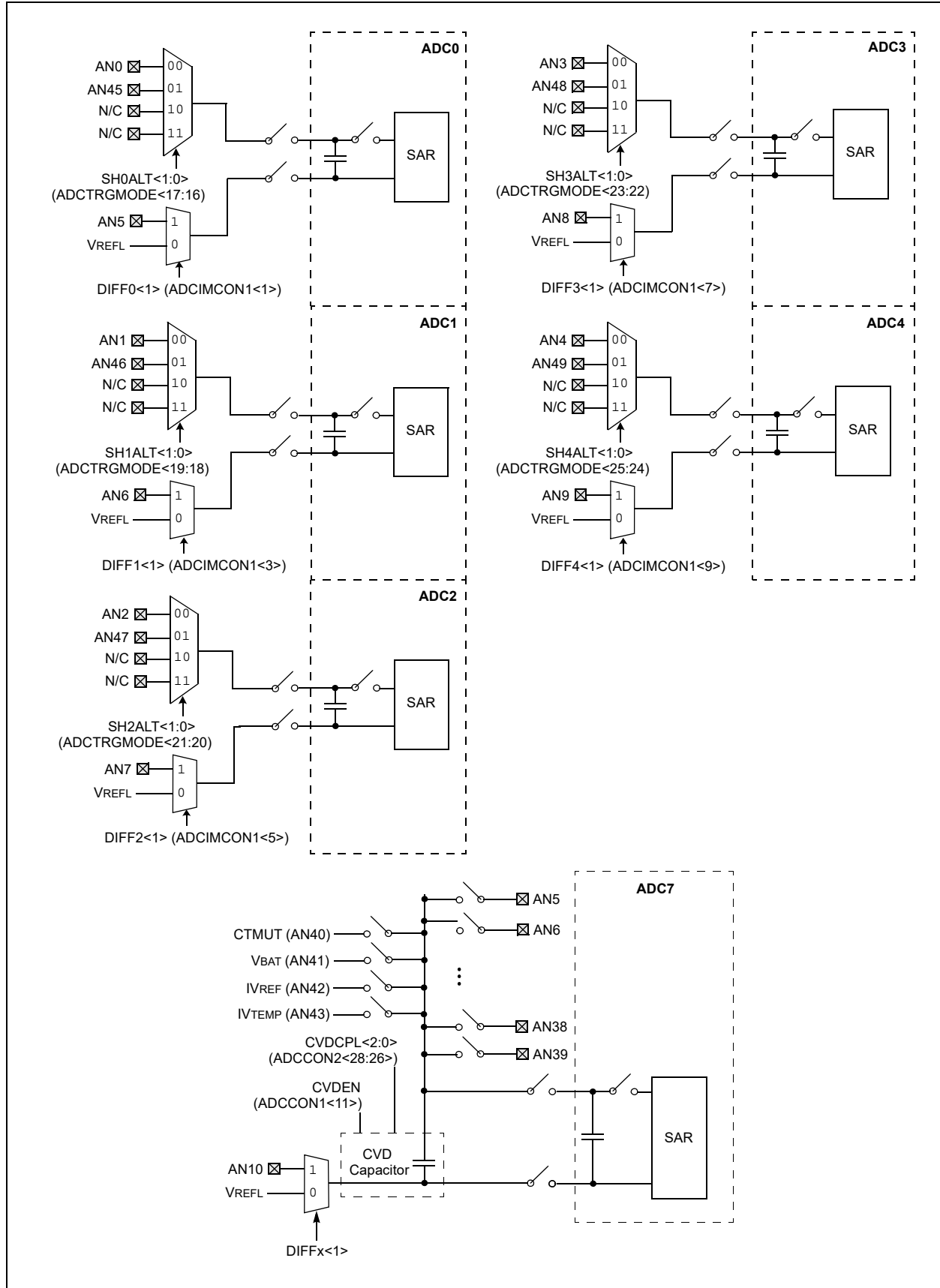
# PIC32MZ グラフィック (DA) ファミリ

図 29-1: ADC のブロック図



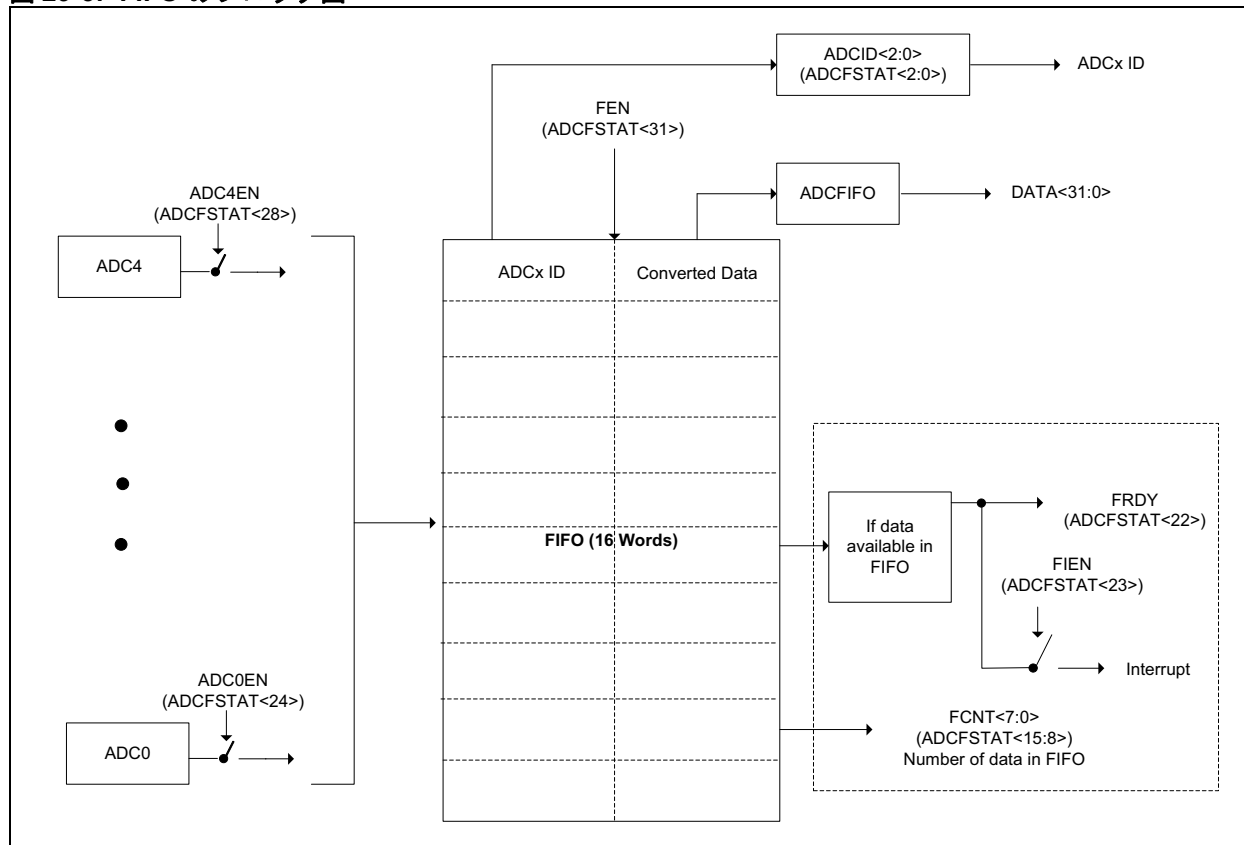
# PIC32MZ グラフィック (DA) ファミリ

図 29-2: S&H 回路のブロック図



# PIC32MZ グラフィック (DA) ファミリ

図 29-3: FIFO のブロック図



## 29.1 ADC 制御レジスタ

表 29-1: ADC のレジスタマップ

レジスタアドレス (BF84#)	レジスタ名	ビット番号	Bit														書きセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
B000	ADCCON1	31:16	TRBEN	TRBERR	TRBMST<2:0>			TRBSLV<2:0>			FRACT	SELRES<1:0>		STRGSRC<4:0>				0060	
		15:0	ON	—	SIDL	AICMPEN	CVDEN	FSSCLKEN	FSPBCLKEN	—	—	IRQVS<2:0>		STRGLVL	—	—	—	0000	
B004	ADCCON2	31:16	BGVRDY	REFFLT	EOSRDY	CVD CPL<2:0>			SAMC<9:0>								0000		
		15:0	BGVRIEN	REFFLTIEN	EOSIEN	ADCEIOVR	—	ADCEIS<2:0>			—	ADCDIV<6:0>					0000		
B008	ADCCON3	31:16	ADCSEL<1:0>		CONCLKDIV<5:0>					DIGEN7	—	—	DIGEN4	DIGEN3	DIGEN2	DIGEN1	DIGEN0	0000	
		15:0	VREFSEL<2:0>			TRGSUSP	UPDIEN	UPDRDY	SAMP	RQCNVRT	GLSWTRG	GSWTRG	ADINSEL<5:0>					0000	
B00C	ADCTRGMODE	31:16	—	—	—	—	SH4ALT<1:0>			SH3ALT<1:0>			SH2ALT<1:0>		SH0ALT<1:0>			0000	
		15:0	—	—	—	STRGEN4	STRGEN3	STRGEN2	STRGEN1	STRGEN0	—	—	—	SSAMPEN4	SSAMPEN3	SSAMPEN2	SSAMPEN1	SSAMPEN0	0000
B010	ADCIMCON1	31:16	DIFF15	SIGN15	DIFF14	SIGN14	DIFF13	SIGN13	DIFF12	SIGN12	DIFF11	SIGN11	DIFF10	SIGN10	DIFF9	SIGN9	DIFF8	SIGN8	0000
		15:0	DIFF7	SIGN7	DIFF6	SIGN6	DIFF5	SIGN5	DIFF4	SIGN4	DIFF3	SIGN3	DIFF2	SIGN2	DIFF1	SIGN1	DIFF0	SIGN0	0000
B014	ADCIMCON2	31:16	DIFF31	SIGN31	DIFF30	SIGN30	DIFF29	SIGN29	DIFF28	SIGN28	DIFF27	SIGN27	DIFF26	SIGN26	DIFF25	SIGN25	DIFF24	SIGN24	0000
		15:0	DIFF23	SIGN23	DIFF22	SIGN22	DIFF21	SIGN21	DIFF20	SIGN20	DIFF19	SIGN19	DIFF18	SIGN18	DIFF17	SIGN17	DIFF16	SIGN16	0000
B018	ADCIMCON3	31:16	—	—	—	—	—	—	—	—	DIFF43	SIGN43	DIFF42	SIGN42	DIFF41	SIGN41	DIFF40	SIGN40	0000
		15:0	DIFF39	SIGN39	DIFF38	SIGN38	DIFF37	SIGN37	DIFF36	SIGN36	DIFF35	SIGN35	DIFF34	SIGN34	DIFF33	SIGN33	DIFF32	SIGN32	0000
B020	ADCGIRQEN1	31:16	AGIEN31	AGIEN30	AGIEN29	AGIEN28	AGIEN27	AGIEN26	AGIEN25	AGIEN24	AGIEN23	AGIEN22	AGIEN21	AGIEN20	AGIEN19	AGIEN18	AGIEN17	AGIEN16	0000
		15:0	AGIEN15	AGIEN14	AGIEN13	AGIEN12	AGIEN11	AGIEN10	AGIEN9	AGIEN8	AGIEN7	AGIEN6	AGIEN5	AGIEN4	AGIEN3	AGIEN2	AGIEN1	AGIEN0	0000
B024	ADCGIRQEN2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	AGIEN43	AGIEN42	AGIEN41	AGIEN40	AGIEN39	AGIEN38	AGIEN37	AGIEN36	AGIEN35	AGIEN34	AGIEN33	AGIEN32	0000
B028	ADCCSS1	31:16	CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24	CSS23	CSS22	CSS21	CSS20	CSS19	CSS18	CSS17	CSS16	0000
		15:0	CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000
B02C	ADCCSS2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	CSS43	CSS42	CSS41	CSS40	CSS39	CSS38	CSS37	CSS36	CSS35	CSS34	CSS33	CSS32	0000
B030	ADCDSTAT1	31:16	ARDY31	ARDY30	ARDY29	ARDY28	ARDY27	ARDY26	ARDY25	ARDY24	ARDY23	ARDY22	ARDY21	ARDY20	ARDY19	ARDY18	ARDY17	ARDY16	0000
		15:0	ARDY15	ARDY14	ARDY13	ARDY12	ARDY11	ARDY10	ARDY9	ARDY8	ARDY7	ARDY6	ARDY5	ARDY4	ARDY3	ARDY2	ARDY1	ARDY0	0000
B034	ADCDSTAT2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	ARDY43	ARDY42	ARDY41	ARDY40	ARDY39	ARDY38	ARDY37	ARDY36	ARDY35	ARDY34	ARDY33	ARDY32	0000
B038	ADCCMPEN1	31:16	CMPE31	CMPE30	CMPE29	CMPE28	CMPE27	CMPE26	CMPE25	CMPE24	CMPE23	CMPE22	CMPE21	CMPE20	CMPE19	CMPE18	CMPE17	CMPE16	0000
		15:0	CMPE15	CMPE14	CMPE13	CMPE12	CMPE11	CMPE10	CMPE9	CMPE8	CMPE7	CMPE6	CMPE5	CMPE4	CMPE3	CMPE2	CMPE1	CMPE0	0000
B03C	ADCCMP1	31:16	DCMPHI<15:0>															0000	
		15:0	DCMPLO<15:0>															0000	
B040	ADCCMPEN2	31:16	CMPE31	CMPE30	CMPE29	CMPE28	CMPE27	CMPE26	CMPE25	CMPE24	CMPE23	CMPE22	CMPE21	CMPE20	CMPE19	CMPE18	CMPE17	CMPE16	0000
		15:0	CMPE15	CMPE14	CMPE13	CMPE12	CMPE11	CMPE10	CMPE9	CMPE8	CMPE7	CMPE6	CMPE5	CMPE4	CMPE3	CMPE2	CMPE1	CMPE0	0000
B044	ADCCMP2	31:16	DCMPHI<15:0>															0000	
		15:0	DCMPLO<15:0>															0000	
B048	ADCCMPEN3	31:16	CMPE31	CMPE30	CMPE29	CMPE28	CMPE27	CMPE26	CMPE25	CMPE24	CMPE23	CMPE22	CMPE21	CMPE20	CMPE19	CMPE18	CMPE17	CMPE16	0000
		15:0	CMPE15	CMPE14	CMPE13	CMPE12	CMPE11	CMPE10	CMPE9	CMPE8	CMPE7	CMPE6	CMPE5	CMPE4	CMPE3	CMPE2	CMPE1	CMPE0	0000

Note 1: ADC を有効にする前に、ユーザアプリケーションは ADC 校正値を初期化する (工場 で DEVADCx フラッシュ レジスタに書き込まれた校正値に対応する ADCxCFG レジスタにコピーする) 必要があります。

表 29-1: ADC のレジスタマップ (続き)

レジスタアドレ ス (BF04_#)	レジスタ名	ビット 範囲	Bit														リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
B04C	ADCCMP3	31:16	DCMPHI<15:0>														0000		
		15:0	DCMPLO<15:0>														0000		
B050	ADCCMPEN4	31:16	CMPE31	CMPE30	CMPE29	CMPE28	CMPE27	CMPE26	CMPE25	CMPE24	CMPE23	CMPE22	CMPE21	CMPE20	CMPE19	CMPE18	CMPE17	CMPE16	0000
		15:0	CMPE15	CMPE14	CMPE13	CMPE12	CMPE11	CMPE10	CMPE9	CMPE8	CMPE7	CMPE6	CMPE5	CMPE4	CMPE3	CMPE2	CMPE1	CMPE0	0000
B054	ADCCMP4	31:16	DCMPHI<15:0>														0000		
		15:0	DCMPLO<15:0>														0000		
B058	ADCCMPEN5	31:16	CMPE31	CMPE30	CMPE29	CMPE28	CMPE27	CMPE26	CMPE25	CMPE24	CMPE23	CMPE22	CMPE21	CMPE20	CMPE19	CMPE18	CMPE17	CMPE16	0000
		15:0	CMPE15	CMPE14	CMPE13	CMPE12	CMPE11	CMPE10	CMPE9	CMPE8	CMPE7	CMPE6	CMPE5	CMPE4	CMPE3	CMPE2	CMPE1	CMPE0	0000
B05C	ADCCMP5	31:16	DCMPHI<15:0>														0000		
		15:0	DCMPLO<15:0>														0000		
B060	ADCCMPEN6	31:16	CMPE31	CMPE30	CMPE29	CMPE28	CMPE27	CMPE26	CMPE25	CMPE24	CMPE23	CMPE22	CMPE21	CMPE20	CMPE19	CMPE18	CMPE17	CMPE16	0000
		15:0	CMPE15	CMPE14	CMPE13	CMPE12	CMPE11	CMPE10	CMPE9	CMPE8	CMPE7	CMPE6	CMPE5	CMPE4	CMPE3	CMPE2	CMPE1	CMPE0	0000
B064	ADCCMP6	31:16	DCMPHI<15:0>														0000		
		15:0	DCMPLO<15:0>														0000		
B068	ADCFLTR1	31:16	AFEN	DATA16EN	DFMODE	OVRSAM<2:0>			AFGIEN	AFRDY	—	—	—	CHNLID<4:0>				0000	
		15:0	FLTRDATA<15:0>														0000		
B06C	ADCFLTR2	31:16	AFEN	DATA16EN	DFMODE	OVRSAM<2:0>			AFGIEN	AFRDY	—	—	—	CHNLID<4:0>				0000	
		15:0	FLTRDATA<15:0>														0000		
B070	ADCFLTR3	31:16	AFEN	DATA16EN	DFMODE	OVRSAM<2:0>			AFGIEN	AFRDY	—	—	—	CHNLID<4:0>				0000	
		15:0	FLTRDATA<15:0>														0000		
B074	ADCFLTR4	31:16	AFEN	DATA16EN	DFMODE	OVRSAM<2:0>			AFGIEN	AFRDY	—	—	—	CHNLID<4:0>				0000	
		15:0	FLTRDATA<15:0>														0000		
B078	ADCFLTR5	31:16	AFEN	DATA16EN	DFMODE	OVRSAM<2:0>			AFGIEN	AFRDY	—	—	—	CHNLID<4:0>				0000	
		15:0	FLTRDATA<15:0>														0000		
B07C	ADCFLTR6	31:16	AFEN	DATA16EN	DFMODE	OVRSAM<2:0>			AFGIEN	AFRDY	—	—	—	CHNLID<4:0>				0000	
		15:0	FLTRDATA<15:0>														0000		
B080	ADCTRG1	31:16	—	—	—	TRGSRC3<4:0>			—	—	—	TRGSRC2<4:0>				0000			
		15:0	—	—	—	TRGSRC1<4:0>			—	—	—	TRGSRC0<4:0>				0000			
B084	ADCTRG2	31:16	—	—	—	TRGSRC7<4:0>			—	—	—	TRGSRC6<4:0>				0000			
		15:0	—	—	—	TRGSRC5<4:0>			—	—	—	TRGSRC4<4:0>				0000			
B088	ADCTRG3	31:16	—	—	—	TRGSRC11<4:0>			—	—	—	TRGSRC10<4:0>				0000			
		15:0	—	—	—	TRGSRC9<4:0>			—	—	—	TRGSRC8<4:0>				0000			
B0A0	ADCCMPCON1	31:16	CVDDATA<15:0>														0000		
		15:0	AINID<5:0>						ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO	0000		
B0A4	ADCCMPCON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	AINID<4:0>						ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO	0000		
B0A8	ADCCMPCON3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	AINID<4:0>						ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO	0000		

Note 1: ADC を有効にする前に、ユーザ アプリケーションは ADC 校正値を初期化する (工場 で DEVADCx フラッシュ レジスタに書き込まれた校正値に対応する ADCxCFG レジスタにコピーする) 必要があります。



表 29-1: ADC のレジスタマップ (続き)

アドレッシング レジスタ名 (BF84_#)	レジスタ名	ビット 番号	Bit															書きセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
B0AC	ADCCMPCON4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	AINID<4:0>				ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO	0000	
B0B0	ADCCMPCON5	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	AINID<4:0>				ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO	0000	
B0B4	ADCCMPCON6	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	AINID<4:0>				ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO	0000	
B0B8	ADCFSTAT	31:16	FEN	—	—	ADC4EN	ADC3EN	ADC2EN	ADC1EN	ADC0EN	FIEN	FRDY	FWROVERR	—	—	—	—	—	0000
		15:0	FCNT<7:0>							FSIGN	—	—	—	—	—	ADCID<2:0>			0000
B0BC	ADCFIFO	31:16	DATA<31:16>															0000	
		15:0	DATA<15:0>															0000	
B0C0	ADCBASE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ADCBASE<15:0>															0000	
B0D0	ADCTRGSNS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	LVL11	LVL10	LVL9	LVL8	LVL7	LVL6	LVL5	LVL4	LVL3	LVL2	LVL1	LVL0	0000
B0D4	ADC0TIME	31:16	—	—	—	ADCEIS<2:0>			SELRES<1:0>			—	ADCDIV<6:0>					0300	
		15:0	—	—	—	—	—	—	SAMC<9:0>					ADCDIV<6:0>			0000		
B0D8	ADC1TIME	31:16	—	—	—	ADCEIS<2:0>			SELRES<1:0>			—	ADCDIV<6:0>					0300	
		15:0	—	—	—	—	—	—	SAMC<9:0>					ADCDIV<6:0>			0000		
B0DC	ADC2TIME	31:16	—	—	—	ADCEIS<2:0>			SELRES<1:0>			—	ADCDIV<6:0>					0300	
		15:0	—	—	—	—	—	—	SAMC<9:0>					ADCDIV<6:0>			0000		
B0E0	ADC3TIME	31:16	—	—	—	ADCEIS<2:0>			SELRES<1:0>			—	ADCDIV<6:0>					0300	
		15:0	—	—	—	—	—	—	SAMC<9:0>					ADCDIV<6:0>			0000		
B0E4	ADC4TIME	31:16	—	—	—	ADCEIS<2:0>			SELRES<1:0>			—	ADCDIV<6:0>					0300	
		15:0	—	—	—	—	—	—	SAMC<9:0>					ADCDIV<6:0>			0000		
B0F0	ADCEIEN1	31:16	EIEN31	EIEN30	EIEN29	EIEN28	EIEN27	EIEN26	EIEN25	EIEN24	EIEN23	EIEN22	EIEN21	EIEN20	EIEN19	EIEN18	EIEN17	EIEN16	0000
		15:0	EIEN15	EIEN14	EIEN13	EIEN12	EIEN11	EIEN10	EIEN9	EIEN8	EIEN7	EIEN6	EIEN5	EIEN4	EIEN3	EIEN2	EIEN1	EIEN0	0000
B0F4	ADCEIEN2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	EIEN43	EIEN42	EIEN41	EIEN40	EIEN39	EIEN38	EIEN37	EIEN36	EIEN35	EIEN34	EIEN33	EIEN32	0000
B0F8	ADCEISTAT1	31:16	EIRDY31	EIRDY30	EIRDY29	EIRDY28	EIRDY27	EIRDY26	EIRDY25	EIRDY24	EIRDY23	EIRDY22	EIRDY21	EIRDY20	EIRDY19	EIRDY18	EIRDY17	EIRDY16	0000
		15:0	EIRDY15	EIRDY14	EIRDY13	EIRDY12	EIRDY11	EIRDY10	EIRDY9	EIRDY8	EIRDY7	EIRDY6	EIRDY5	EIRDY4	EIRDY3	EIRDY2	EIRDY1	EIRDY0	0000
B0FC	ADCEISTAT2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	EIRDY43	EIRDY42	EIRDY41	EIRDY40	EIRDY39	EIRDY38	EIRDY37	EIRDY36	EIRDY35	EIRDY34	EIRDY33	EIRDY32	0000
B100	ADCANCON	31:16	—	—	—	—	WKUPCLKCNT<3:0>				WKIEN7	—	—	WKIEN4	WKIEN3	WKIEN2	WKIEN1	WKIEN0	0000
		15:0	WKRDY7	—	—	WKRDY4	WKRDY3	WKRDY2	WKRDY1	WKRDY0	ANEN7	—	—	ANEN4	ANEN3	ANEN2	ANEN1	ANEN0	0000
B600	ADC0CFG <sup>(1)</sup>	31:16	ADCCFG<31:16>															0000	
		15:0	ADCCFG<15:0>															0000	
B604	ADC1CFG <sup>(1)</sup>	31:16	ADCCFG<31:16>															0000	
		15:0	ADCCFG<15:0>															0000	

Note 1: ADC を有効にする前に、ユーザアプリケーションは ADC 校正値を初期化する (工場 で DEVADCx フラッシュ レジスタに書き込まれた校正値を対応する ADCxCFG レジスタにコピーする) 必要があります。

表 29-1: ADC のレジスタマップ (続き)

レジスタアドレ ス (BF84_#)	レジスタ名	ビット 範囲	Bit														リセット 値	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
B608	ADC2CFG <sup>1)</sup>	31:16	ADCCFG<31:16>														0000	
		15:0	ADCCFG<15:0>														0000	
B60C	ADC3CFG <sup>1)</sup>	31:16	ADCCFG<31:16>														0000	
		15:0	ADCCFG<15:0>														0000	
B610	ADC4CFG <sup>1)</sup>	31:16	ADCCFG<31:16>														0000	
		15:0	ADCCFG<15:0>														0000	
B61C	ADC7CFG <sup>1)</sup>	31:16	ADCCFG<31:16>														0000	
		15:0	ADCCFG<15:0>														0000	
B640	ADCSYSCFG1	31:16	AN<31:16>														0000	
		15:0	AN<15:0>														0000	
B644	ADCSYSCFG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	AN<43:32>										0000	
BA00	ADCDATA0	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA04	ADCDATA1	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA08	ADCDATA2	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA0C	ADCDATA3	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA10	ADCDATA4	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA14	ADCDATA5	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA18	ADCDATA6	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA1C	ADCDATA7	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA20	ADCDATA8	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA24	ADCDATA9	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA28	ADCDATA10	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA2C	ADCDATA11	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	
BA30	ADCDATA12	31:16	DATA<31:16>														0000	
		15:0	DATA<15:0>														0000	

Note 1: ADC を有効にする前に、ユーザ アプリケーションは ADC 校正値を初期化する (工場 で DEVADCx フラッシュ レジスタに書き込まれた校正値を対応する ADCxCFG レジスタにコピーする) 必要があります。

表 29-1: ADC のレジスタマップ (続き)

レジスタアドレ ス (BF8_#)	レジスタ名	ビット レンジ	Bit														リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	
BA34	ADCDATA13	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA38	ADCDATA14	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA3C	ADCDATA15	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA40	ADCDATA16	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA44	ADCDATA17	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA48	ADCDATA18	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA4C	ADCDATA19	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA50	ADCDATA20	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA54	ADCDATA21	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA58	ADCDATA22	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA5C	ADCDATA23	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA60	ADCDATA24	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA64	ADCDATA25	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA68	ADCDATA26	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA6C	ADCDATA27	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA70	ADCDATA28	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA74	ADCDATA29	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA78	ADCDATA30	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA7C	ADCDATA31	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000

Note 1: ADC を有効にする前に、ユーザアプリケーションは ADC 校正値を初期化する (工場 で DEVADCx フラッシュ レジスタに書き込まれた校正値を対応する ADCxCFG レジスタにコピーする) 必要があります。

表 29-1: ADC のレジスタマップ ( 続き )

修正アドレ ス (BF84_#)	レジスタ名	ビット 範囲	Bit														リセット 値
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	
BA80	ADCDATA32	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA84	ADCDATA33	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA88	ADCDATA34	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA8C	ADCDATA35	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA90	ADCDATA36	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA94	ADCDATA37	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA98	ADCDATA38	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BA9C	ADCDATA39	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BAA0	ADCDATA40	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BAA4	ADCDATA41	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BAA8	ADCDATA42	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000
BAAC	ADCDATA43	31:16	DATA<31:16>														0000
		15:0	DATA<15:0>														0000

**Note** 1: ADC を有効にする前に、ユーザ アプリケーションは ADC 校正値を初期化する ( 工場 で DEVADCx フラッシュ レジスタに書き込まれた校正値を対応する ADCxCFG レジスタにコピーする ) 必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-1: ADCCON1: ADC制御レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R-0, HS, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TRBEN	TRBERR	TRBMST<2:0>			TRBSLV<2:0>		
23:16	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FRACT	SELRES<1:0>		STRGSRC<4:0>				
15:8	R/W-0	U-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0	U-0
	ON		SIDL	AICMPEN	CVDEN	FSSCLKEN	FSPBCLKEN	—
7:0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
	—	IRQVS<2:0>			STRGLVL	—	—	—

**凡例:** HC = ハードウェアクリアビット HS = ハードウェアセットビット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31 **TRBEN:** ターボチャンネル イネーブルビット

1 = ターボチャンネルを有効にする  
0 = ターボチャンネルを無効にする

bit 30 **TRBERR:** ターボチャンネル エラー ステータスビット

1 = ターボチャンネルの設定中にエラーが発生した  
TRBEN ビットを「1」にセットしていてもターボチャンネル機能は無効になります。  
0 = ターボチャンネル エラーは発生していない

**Note:** このビットのステータスは TRBEN ビットをセットした後でのみ有効です。

bit 29-27 **TRBMST<2:0>:** ターボマスタ ADCx ビット

111 = 予約済み  
110 = ADC4 をターボマスタとして選択する  
:  
:  
000 = ADC0 をターボマスタとして選択する

bit 26-24 **TRBSLV<2:0>:** ターボスレーブ ADCx ビット

111 = 予約済み  
110 = ADC4 をターボスレーブとして選択する  
:  
:  
000 = ADC0 をターボスレーブとして選択する

bit 23 **FRACT:** 小数データ出力フォーマット ビット

1 = 小数  
0 = 整数

bit 22-21 **SELRES<1:0>:** 共有 ADC (ADC7) 分解能ビット

11 = 12 ビット (既定値)  
10 = 10 ビット  
01 = 8 ビット  
00 = 6 ビット

**Note:** ADC の分解能を変更しても、対応する ADCDATAx レジスタ内の ADC 結果はシフトしません。結果は 12 ビット幅のまま、未使用の下位ビットが「0」に設定されます。例えば 6 ビット分解能の場合、ADCDATAx<5:0> は「0」に設定され、ADCDATAx<11:6> が ADC 結果を保持します。

**Note 1:** モジュール出力信号の立ち上がりエッジで ADC 変換をトリガします。詳細は、16.0「出力コンペア」内の図 16-1 と、32.0「コンパレータ」内の図 32-1 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-1: ADCCON1: ADC制御レジスタ 1 (続き)

bit 20-16 **STRGSRC<4:0>**: スキャントリガ源選択ビット

11111 = 予約済み  
11110 = 予約済み  
11101 = CTMU イベント  
11100 = 予約済み  
.  
.  
01110 = 予約済み  
01101 = CTMU イベント  
01100 = コンパレータ 2 (C2OUT) <sup>(1)</sup>  
01011 = コンパレータ 1 (C1OUT) <sup>(1)</sup>  
01010 = OCMP5 <sup>(1)</sup>  
01001 = OCMP3 <sup>(1)</sup>  
01000 = OCMP1 <sup>(1)</sup>  
00111 = TMR5 一致  
00110 = TMR3 一致  
00101 = TMR1 一致  
00100 = INT0 外部割り込み  
00011 = 予約済み  
00010 = グローバル ソフトウェア レベルトリガ (GLSWTRG)  
00001 = グローバル ソフトウェア エッジトリガ (GSWTRG)  
00000 = トリガなし

bit 15 **ON**: ADC モジュール イネーブルビット

1 = ADC モジュールを有効にする  
0 = ADC モジュールを無効にする

**Note**: ON ビットは ADC モジュールの設定を済ませてからセットする必要があります。

bit 14 **未実装**: 「0」として読み出し

bit 13 **SIDL**: アイドル中停止ビット

1 = デバイスがアイドルに移行した時にモジュールは動作を停止する  
0 = デバイスがアイドルモード中でもモジュールは動作を継続する

bit 12 **AICMPEN**: アナログ入力用チャージポンプ イネーブルビット

1 = アナログ入力用チャージポンプを有効にする  
0 = アナログ入力用チャージポンプを無効にする

**Note 1**: VDDIO が 2.5 V より低い場合、アナログ動作を適正に維持するために AICMPEN ビットと IOANCPEN (CFGCON<7>) ビットを「1」にセットする必要があります。VDDIO が 2.5 V より高い場合、これらのビットはセットしません。

**2**: AICMPEN (ADCCON1<12>) ビットと IOANCPEN (CFGCON<7>) ビットを「1」にセットした場合、ADC スループット レート性能は下表の通りに低下します。

ADC0	ADC1	ADC2	ADC3	ADC4	ADC7	Maximum combined
ON	OFF	OFF	OFF	OFF	OFF	2 MSPS
ON	ON	OFF	OFF	OFF	OFF	4 MSPS
ON	ON	ON	OFF	OFF	OFF	5 MSPS
OFF	OFF	OFF	ON	OFF	OFF	2 MSPS
OFF	OFF	OFF	ON	ON	OFF	4 MSPS
OFF	OFF	OFF	ON	ON	ON	5 MSPS
ON	ON	ON	ON	OFF	OFF	7 MSPS
ON	ON	ON	ON	ON	OFF	9 MSPS
ON	ON	ON	ON	ON	ON	10 MSPS

bit 11 **CVDEN**: 静電容量式分圧 (CVD) イネーブルビット

1 = CVD 動作を有効にする  
0 = CVD 動作を無効にする

**Note 1**: モジュール出力信号の立ち上がりエッジで ADC 変換をトリガします。詳細は、[16.0「出カコンペア」](#) 内の [図 16-1](#) と、[32.0「コンパレータ」](#) 内の [図 32-1](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-1: ADCCON1: ADC制御レジスタ 1 (続き)

- bit 10 **FSSCLKEN**: ADC 制御クロック / システムクロック高速同期ビット  
1 = ADC 制御クロックとシステムクロックの高速同期を有効にする  
0 = ADC 制御クロックとシステムクロックの高速同期を無効にする
- bit 9 **FSPBCLKEN**: ADC 制御クロック / 周辺モジュール用クロック高速同期ビット  
1 = ADC 制御クロックと周辺モジュール用クロックの高速同期を有効にする  
0 = ADC 制御クロックと周辺モジュール用クロックの高速同期を無効にする
- bit 8-7 **未実装**: 「0」として読み出し
- bit 6-4 **IRQVS<2:0>**: 割り込みベクタシフトビット  
これらのビットで指定した分だけ ADCDSTAT1 および ADCDSTAT2 レジスタ内の ARDYx ステータスビットを左へシフトさせます。このシフトした値を ADCBASE レジスタの値に加算する事で、割り込みベクタアドレスが決まります。  
割り込みベクタアドレス = ADCBASE の読み値  
ADCBASE の読み値 = ADCBASE に書き込まれた値 + x << IRQVS<2:0>  
(「x」は、ADCDSTAT1 または ADCDSTAT2 レジスタ内でアクティブな最小 (最高優先度) の入力 ID)  
111 = x を左に 7 ビットシフトする  
110 = x を左に 6 ビットシフトする  
101 = x を左に 5 ビットシフトする  
100 = x を左に 4 ビットシフトする  
011 = x を左に 3 ビットシフトする  
010 = x を左に 2 ビットシフトする  
001 = x を左に 1 ビットシフトする  
000 = x を左に 0 ビットシフトする
- bit 3 **STRGLVL**: スキャントリガ HIGH レベル / 立ち上がりエッジ検出ビット  
1 = スキャントリガを HIGH レベル検出にする  
ADCTRGx レジスタの TRGSRCx<4:0> で STRIG モードを選択すると、STRIG オプションの選択を解除するまでスキャントリガは選択した全てのアナログ入力に対して有効なままになります。  
0 = スキャントリガを立ち上がりエッジ検出にする  
ADCTRGx レジスタの TRGSRCx<4:0> で STRIG モードを選択すると、スキャントリガは 1 回だけ生成され、選択した全てのアナログ入力のスキャンが完了します。
- bit 2-0 **未実装**: 「0」として読み出し

**Note 1:** モジュール出力信号の立ち上がりエッジで ADC 変換をトリガします。詳細は、[16.0「出力コンペア」](#)内の [図 16-1](#) と、[32.0「コンパレータ」](#)内の [図 32-1](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-2: ADCCON2: ADC制御レジスタ 2

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BGVRDY	REFFLT	EOSRDY	CVDCPL<2:0>			SAMC<9:8>	
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SAMC<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
	BGVRIEN	REFFLTEN	EOSIEN	ADCEIOVR	—	ADCEIS<2:0>		
7:0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	ADCDIV<6:0>						

**凡例:** HC = ハードウェアクリアビット HS = ハードウェアセットビット r = 予約済み  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

- bit 31 **BGVRDY:** バンドギャップ電圧 /ADC 参照電圧ステータスビット  
1 = バンドギャップ電圧と ADC 参照電圧 (VREF) の両方の準備が完了している  
0 = バンドギャップ電圧と ADC 参照電圧 (VREF) の片方または両方の準備が完了していない  
ハードウェアが BGVRDY ビットをセットするまで、データ処理は有効になりません。このため、アプリケーションコードは BGVRDY ビットがセットされている事 (データが有効である事) を確認する必要があります。このビットは、ON (ADCCON1<15>) = 0 の時に「0」に設定されます。
- bit 30 **REFFLT:** バンドギャップ /VREF/AVDD BOR 異常ステータスビット  
1 = ON ビット (ADCCON1<15>) がセットされている時にバンドギャップまたは VREF 電圧に異常が発生した  
多くの場合、バンドギャップまたは VREF 異常はアナログ VDDIO 電源の BOR によって発生します。  
0 = バンドギャップと VREF 電圧は正常に動作している  
ON ビット (ADCCON1<15>) = 0 かつ BGVRDY ビット = 1 の場合、このビットはクリアされます。
- bit 29 **EOSRDY:** スキャン終了時割り込みステータスビット  
1 = ADCCSS1 および ADCCSS2 レジスタで入力スキャン向けに選択された全てのアナログ入力のスキャンが完了した  
0 = スキャンは完了していない  
このビットは、ソフトウェアで ADCCON2<31:24> を読み出した時にクリアされます。
- bit 28-26 **CVDCPL<2:0>:** 静電容量式分圧器 (CVD) 設定ビット  
111 = 7 \* 2.5 pF = 17.5 pF  
110 = 6 \* 2.5 pF = 15 pF  
101 = 5 \* 2.5 pF = 12.5 pF  
100 = 4 \* 2.5 pF = 10 pF  
011 = 3 \* 2.5 pF = 7.5 pF  
010 = 2 \* 2.5 pF = 5 pF  
001 = 1 \* 2.5 pF = 2.5 pF  
000 = 0 \* 2.5 pF = 0 pF
- bit 25-16 **SAMC<9:0>:** 共有 ADC (ADC7) サンプリング時間ビット  
1111111111 = 1025 TAD7  
.  
.  
.  
0000000001 = 3 TAD7  
0000000000 = 2 TAD7  
TAD7 = 共有 ADC (ADC7) の ADC 変換クロック周期 (ADCDIV<6:0> ビットで設定)
- bit 15 **BGVRIEN:** バンドギャップ /VREF 電圧レディ割り込みイネーブルビット  
1 = BGVRDY ビットがセットされた時に割り込みを生成する  
0 = BGVRDY ビットがセットされても割り込みを生成しない



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-2: ADCCON2: ADC制御レジスタ 2 (続き)

- bit 14     **REFFLTEN:** バンドギャップ/VREF 電圧異常割り込みイネーブルビット  
1 = REFFLT ビットがセットされた時に割り込みを生成する  
0 = REFFLT ビットがセットされても割り込みを生成しない
- bit 13     **EOSIEN:** スキャン終了時割り込みイネーブルビット  
1 = EOSRDY ビットがセットされた時に割り込みを生成する  
0 = EOSRDY ビットがセットされても割り込みを生成しない
- bit 12     **ADCEIOVR:** 早期割り込み要求オーバーライド ビット  
1 = 早期割り込み生成をオーバーライドしない(割り込みの生成はADCEIEN1およびADCEIEN2レジスタにより制御する)  
0 = 早期割り込み生成をオーバーライドする(割り込みの生成はADCGIRQEN1 およびADCGIRQEN2 レジスタにより制御する)
- bit 11     **未実装:** 「0」として読み出し
- bit 10-8   **ADCEIS<2:0>:** 共有 ADC (ADC7) 早期割り込み選択ビット  
変換が完了する前に、指定したクロック数だけ早期に割り込みを生成します。  
111 = 変換終了の 8 ADC クロック前にデータレディ割り込みを生成する  
110 = 変換終了の 7 ADC クロック前にデータレディ割り込みを生成する  
.  
.  
001 = 変換終了の 2 ADC クロック前にデータレディ割り込みを生成する  
000 = 変換終了の 1 ADC クロック前にデータレディ割り込みを生成する  
**Note:** SELRES<1:0> ビット (ADCCON1<22:21>) で 12 ビットまたは 10 ビット分解能を選択した場合、全てのオプション (「000」～「111」) が選択できます。8 ビットの分解能を選択した場合、「000」～「101」のみ選択可能です。6 ビットの分解能を選択した場合、「000」～「011」のみ選択可能です。
- bit 7     **未実装:** 「0」として読み出し
- bit 6-0   **ADCDIV<6:0>:** 共有 ADC (ADC7) クロック分周ビット  
11111111 =  $254 * TQ = TAD7$   
.  
.  
0000011 =  $6 * TQ = TAD7$   
0000010 =  $4 * TQ = TAD7$   
0000001 =  $2 * TQ = TAD7$   
0000000 = 予約済み  
ADCDIV<6:0> ビットを使って ADC 制御クロック (Tq) を分周する事で共有 ADC (ADC7) のクロック (TAD7) を生成します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-3: ADCCON3: ADC制御レジスタ 3

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADCSEL<1:0>		CONCLKDIV<5:0>					
23:16	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIGEN7	—	—	DIGEN4	DIGEN3	DIGEN2	DIGEN1	DIGEN0
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0, HS, HC	R/W-0	R-0, HS, HC
	VREFSEL<2:0>			TRGSUSP	UPDIEN	UPDRDY	SAMP <sup>(1,2,3,4)</sup>	RQCNVRT
7:0	R/W-0	R/W, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	GLSWTRG	GSWTRG	ADINSEL<5:0>					

**凡例:** HC = ハードウェアクリアビット HS = ハードウェアセットビット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-30 **ADCSEL<1:0>**: ADC クロック源 (TCLK) ビット

11 = FRC  
10 = REFCLK3  
01 = システムクロック (Tcy)  
00 = PBCLK3

bit 29-24 **CONCLKDIV<5:0>**: ADC 制御クロック (Tq) 分周ビット

111111 = 64 \* TCLK = Tq  
.  
.  
.  
000011 = 4 \* TCLK = Tq  
000010 = 3 \* TCLK = Tq  
000001 = 2 \* TCLK = Tq  
000000 = TCLK = Tq

bit 23 **DIGEN7**: 共有 ADC (ADC7) デジタル イネーブルビット

1 = ADC7 のデジタル機能を有効にする  
0 = ADC7 のデジタル機能を無効にする

bit 22-21 **未実装**: 「0」として読み出し

bit 20 **DIGEN4**: ADC4 デジタル イネーブルビット

1 = ADC4 のデジタル機能を有効にする  
0 = ADC4 のデジタル機能を無効にする

bit 19 **DIGEN3**: ADC3 デジタル イネーブルビット

1 = ADC3 のデジタル機能を有効にする  
0 = ADC3 のデジタル機能を無効にする

**Note 1:** SAMP ビットは最も高い優先度を持ちます。このビットをセットすると、クリアするまで S&H 回路はサンプリングモードのままになります。また、SAMP ビットをセットした場合、SAMC<9:0> ビット (ADCCON2<25:16>) の設定は無視されます。

**2:** SAMP ビットは、クラス 2 およびクラス 3 アナログ入力のみを共有 ADC (ADC7) に接続します。全てのクラス 1 アナログ入力は、SAMP ビットの影響を受けません。

**3:** SAMP ビットは自己クリアビットではありません。A/D 変換を開始するには、アプリケーションソフトウェアでこのビットをクリアしてから RQCNVRT ビットをセットする必要があります。

**4:** 通常、SAMP および RQCNVRT ビットをソフトウェアルーチンで使う場合、外部ハードウェアトリガがソフトウェア制御サンプリングコマンド信号 (SAMP) とソフトウェア制御トリガ (RQCNVRT) に干渉する事を防ぐため、全ての TRGSRCx<4:0> ビットと STRGSRC<4:0> ビットを「00000」に設定して外部ハードウェアトリガを全て無効にする必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-3: ADCCON3: ADC制御レジスタ 3 (続き)

bit 18 **DIGEN2:** ADC2 デジタル イネーブルビット

1 = ADC2 のデジタル機能を有効にする  
0 = ADC2 のデジタル機能を無効にする

bit 17 **DIGEN1:** ADC1 デジタル イネーブルビット

1 = ADC1 のデジタル機能を有効にする  
0 = ADC1 のデジタル機能を無効にする

bit 16 **DIGEN0:** ADC0 デジタル イネーブルビット

1 = ADC0 のデジタル機能を有効にする  
0 = ADC0 のデジタル機能を無効にする

bit 15-13 **VREFSEL<2:0>:** 参照電圧 (VREF) 入力選択ビット

VREFSEL<2:0>	ADREF+	ADREF-
111	AVDD	内部 VREFL
110	内部 VREFH	AVSS
101	内部 VREFH	外部 VREFL
100	内部 VREFH	内部 VREFL
011	内部 VREFH	外部 VREFL
010	AVDD	外部 VREFL
001	外部 VREFH	AVSS
000	AVDD	AVSS

bit 12 **TRGSUSP:** トリガサスペンド ビット

1 = トリガをブロックする (次の A/D 変換を開始しない)  
ADC モジュールは有効なままです。

0 = トリガをブロックしない

bit 11 **UPDIEN:** 更新レディ割り込みイネーブルビット

1 = ハードウェアが UPDRDY ビットをセットした時に割り込みを生成する

0 = 割り込みを生成しない

bit 10 **UPDRDY:** ADC 更新レディ ステータスビット

1 = ADC SFR は更新できる

0 = ADC SFR は更新できない

**Note:** このビットは、TRGSUSP ビットがセットされ、かつ、どの ADC モジュールも変換を実行中ではない場合にのみ有効です。

bit 9 **SAMP:** クラス 2 およびクラス 3 アナログ入力サンプリング イネーブルビット (1,2,3,4)

1 = ADC の S&H アンプをサンプル動作させる

0 = ADC の S&H アンプをホールド動作させる

bit 8 **RQCNVRT:** ADC 入力個別変換要求ビット

このビットと ADINSEL<5:0> ビットにより、ソフトウェアで個別にアナログ入力の A/D 変換を要求できます。

1 = ADINSEL<5:0> ビットで選択されている ADC 入力の変換をトリガする

0 = 変換をトリガしない

**Note:** このビットは、次の ADC クロックサイクルで自動的にクリアされます。

**Note 1:** SAMP ビットは最も高い優先度を持ちます。このビットをセットすると、クリアするまで S&H 回路はサンプリングモードのままになります。また、SAMP ビットをセットした場合、SAMC<9:0> ビット (ADCCON2<25:16>) の設定は無視されます。

**2:** SAMP ビットは、クラス 2 およびクラス 3 アナログ入力のみを共有 ADC (ADC7) に接続します。全てのクラス 1 アナログ入力は、SAMP ビットの影響を受けません。

**3:** SAMP ビットは自己クリアビットではありません。A/D 変換を開始するには、アプリケーションソフトウェアでこのビットをクリアしてから RQCNVRT ビットをセットする必要があります。

**4:** 通常、SAMP および RQCNVRT ビットをソフトウェアルーチンで使う場合、外部ハードウェアトリガがソフトウェア制御サンプリングコマンド信号 (SAMP) とソフトウェア制御トリガ (RQCNVRT) に干渉する事を防ぐため、全ての TRGSRCx<4:0> ビットと STRGSRC<4:0> ビットを「00000」に設定して外部ハードウェアトリガを全て無効にする必要があります。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-3: ADCCON3: ADC制御レジスタ 3 (続き)

bit 7 **GLSWTRG:** グローバル ソフトウェア レベルトリガ ビット  
1 = ADCTRGx レジスタの TRGSRC<4:0> ビットまたは ADCCON1 レジスタの STRGSRC<4:0> ビットで GLSWTRG をトリガ信号として選択している ADC 入力の変換をトリガする  
0 = A/D 変換をトリガしない

bit 6 **GSWTRG:** グローバル ソフトウェア エッジトリガビット  
1 = ADCTRGx レジスタの TRGSRC<4:0> ビットまたは ADCCON1 レジスタの STRGSRC<4:0> ビットで GSWTRG をトリガ信号として選択している ADC 入力の変換をトリガする  
0 = A/D 変換をトリガしない

**Note:** このビットは、次の ADC クロックサイクルで自動的にクリアされます。

bit 5-0 **ADINSEL<5:0>:** アナログ入力選択ビット

これらのビットは、RQCNVRT ビットがセットされた時に変換するアナログ入力を選択します。

一般的な定義:

111111 = 予約済み  
.  
.  
.  
101101 = 予約済み  
101100 = IVTEMP  
101011 = IVREF  
101010 = VBAT  
101000 = CTMU  
100111 = AN39  
.  
.  
.  
000001 = AN1  
000000 = AN0

- Note 1:** SAMP ビットは最も高い優先度を持ちます。このビットをセットすると、クリアするまで S&H 回路はサンプリング モードのままになります。また、SAMP ビットをセットした場合、SAMC<9:0> ビット (ADCCON2<25:16>) の設定は無視されます。
- 2: SAMP ビットは、クラス 2 およびクラス 3 アナログ入力のみを共有 ADC (ADC7) に接続します。全てのクラス 1 アナログ入力は、SAMP ビットの影響を受けません。
  - 3: SAMP ビットは自己クリアビットではありません。A/D 変換を開始するには、アプリケーション ソフトウェアでこのビットをクリアしてから RQCNVRT ビットをセットする必要があります。
  - 4: 通常、SAMP および RQCNVRT ビットをソフトウェア ルーチンで使う場合、外部ハードウェア トリガがソフトウェア制御サンプリング コマンド信号 (SAMP) とソフトウェア制御トリガ (RQCNVRT) に干渉する事を防ぐため、全ての TRGSRCx<4:0> ビットと STRGSRC<4:0> ビットを「00000」に設定して外部ハードウェア トリガを全て無効にする必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-4: ADCTRGMODE: 専用 ADC トリガモード レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	SH4ALT<1:0>	
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SH3ALT<1:0>		SH2ALT<1:0>		SH1ALT<1:0>		SH0ALT<1:0>	
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	STRGEN4	STRGEN3	STRGEN2	STRGEN1	STRGEN0
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	SSAMPEN4	SSAMPEN3	SSAMPEN2	SSAMPEN1	SSAMPEN0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-26 **未実装**: 「0」として読み出し

bit 25-24 **SH4ALT<1:0>**: ADC4 アナログ入力選択ビット

- 11 = 予約済み
- 10 = 予約済み
- 01 = AN49
- 00 = AN4

bit 23-22 **SH3ALT<1:0>**: ADC3 アナログ入力選択ビット

- 11 = 予約済み
- 10 = 予約済み
- 01 = AN48
- 00 = AN3

bit 21-20 **SH2ALT<1:0>**: ADC2 アナログ入力選択ビット

- 11 = 予約済み
- 10 = 予約済み
- 01 = AN47
- 00 = AN2

bit 19-18 **SH1ALT<1:0>**: ADC1 アナログ入力選択ビット

- 11 = 予約済み
- 10 = 予約済み
- 01 = AN46
- 00 = AN1

bit 17-16 **SH0ALT<1:0>**: ADC0 アナログ入力選択ビット

- 11 = 予約済み
- 10 = 予約済み
- 01 = AN45
- 00 = AN0

bit 15-13 **未実装**: 「0」として読み出し

bit 12 **STRGEN4**: ADC4 事前同期トリガビット

- 1 = ADC4 で事前同期トリガを使う
- 0 = ADC4 で事前同期トリガを使わない

bit 11 **STRGEN3**: ADC3 事前同期トリガビット

- 1 = ADC3 で事前同期トリガを使う
- 0 = ADC3 で事前同期トリガを使わない

bit 10 **STRGEN2**: ADC2 事前同期トリガビット

- 1 = ADC2 で事前同期トリガを使う
- 0 = ADC2 で事前同期トリガを使わない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-4: ADCTRGMODE: 専用 ADC トリガモード レジスタ (続き)

- bit 9     **STRGEN1:** ADC1 事前同期トリガビット  
          1 = ADC1 で事前同期トリガを使う  
          0 = ADC1 で事前同期トリガを使わない
- bit 8     **STRGEN0:** ADC0 事前同期トリガビット  
          1 = ADC0 で事前同期トリガを使う  
          0 = ADC0 で事前同期トリガを使わない
- bit 7-5   **未実装:** 「0」 として読み出し
- bit 4     **SSAMPEN4:** ADC4 同期サンプリング ビット  
          1 = ADC4 は、アイドルまたは無効状態後の最初のサンプリングで同期サンプリングを使う  
          0 = ADC4 は同期サンプリングを使わない
- bit 3     **SSAMPEN3:** ADC3 同期サンプリング ビット  
          1 = ADC4 は、アイドルまたは無効状態後の最初のサンプリングで同期サンプリングを使う  
          0 = ADC3 は同期サンプリングを使わない
- bit 2     **SSAMPEN2:** ADC2 同期サンプリング ビット  
          1 = ADC4 は、アイドルまたは無効状態後の最初のサンプリングで同期サンプリングを使う  
          0 = ADC3 は同期サンプリングを使わない
- bit 1     **SSAMPEN1:** ADC1 同期サンプリング ビット  
          1 = ADC4 は、アイドルまたは無効状態後の最初のサンプリングで同期サンプリングを使う  
          0 = ADC3 は同期サンプリングを使わない
- bit 0     **SSAMPEN0:** ADC0 同期サンプリング ビット  
          1 = ADC4 は、アイドルまたは無効状態後の最初のサンプリングで同期サンプリングを使う  
          0 = ADC3 は同期サンプリングを使わない

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-5: ADCIMCON1: ADC入 カモード制御レジスタ 1

ビットレ ンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF15	SIGN15	DIFF14	SIGN14	DIFF13	SIGN13	DIFF12	SIGN12
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF11	SIGN11	DIFF10	SIGN10	DIFF9	SIGN9	DIFF8	SIGN8
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF7	SIGN7	DIFF6	SIGN6	DIFF5	SIGN5	DIFF4	SIGN4
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF3	SIGN3	DIFF2	SIGN2	DIFF1	SIGN1	DIFF0	SIGN0

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット                      U = 未実装、「0」として読み出し  
 -n = POR 時の値                      1 = ビットはセット                      0 = ビットはクリア                      x = ビットは未知

- bit 31        **DIFF15:** AN15 モードビット  
             1 = AN15 を差動モードで使う  
             0 = AN15 をシングルエンドモードで使う
- bit 30        **SIGN15:** AN15 符号付きデータモードビット  
             1 = AN15 を符号付きデータモードで使う  
             0 = AN15 を符号なしデータモードで使う
- bit 29        **DIFF14:** AN14 モードビット  
             1 = AN14 を差動モードで使う  
             0 = AN14 をシングルエンドモードで使う
- bit 28        **SIGN14:** AN14 符号付きデータモードビット  
             1 = AN14 を符号付きデータモードで使う  
             0 = AN14 を符号なしデータモードで使う
- bit 27        **DIFF13:** AN13 モードビット  
             1 = AN13 を差動モードで使う  
             0 = AN13 をシングルエンドモードで使う
- bit 26        **SIGN13:** AN13 符号付きデータモードビット  
             1 = AN13 を符号付きデータモードで使う  
             0 = AN13 を符号なしデータモードで使う
- bit 25        **DIFF12:** AN12 モードビット  
             1 = AN12 を差動モードで使う  
             0 = AN12 をシングルエンドモードで使う
- bit 24        **SIGN12:** AN12 符号付きデータモードビット  
             1 = AN12 を符号付きデータモードで使う  
             0 = AN12 を符号なしデータモードで使う
- bit 23        **DIFF11:** AN11 モードビット  
             1 = AN11 を差動モードで使う  
             0 = AN11 をシングルエンドモードで使う
- bit 22        **SIGN11:** AN11 符号付きデータモードビット  
             1 = AN11 を符号付きデータモードで使う  
             0 = AN11 を符号なしデータモードで使う
- bit 21        **DIFF10:** AN10 モードビット  
             1 = AN10 を差動モードで使う  
             0 = AN10 をシングルエンドモードで使う

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-5: ADCIMCON1: ADC入カモード制御レジスタ 1 (続き)

bit 20	<b>SIGN10:</b> AN10 符号付きデータモードビット 1 = AN10 を符号付きデータモードで使う 0 = AN10 を符号なしデータモードで使う
bit 19	<b>DIFF9:</b> AN9 モードビット 1 = AN9 を差動モードで使う 0 = AN9 をシングルエンドモードで使う
bit 18	<b>SIGN9:</b> AN9 符号付きデータモードビット 1 = AN9 を符号付きデータモードで使う 0 = AN9 を符号なしデータモードで使う
bit 17	<b>DIFF8:</b> AN8 モードビット 1 = AN8 を差動モードで使う 0 = AN8 をシングルエンドモードで使う
bit 16	<b>SIGN8:</b> AN8 符号付きデータモードビット 1 = AN8 を符号付きデータモードで使う 0 = AN8 を符号なしデータモードで使う
bit 15	<b>DIFF7:</b> AN7 モードビット 1 = AN7 を差動モードで使う 0 = AN7 をシングルエンドモードで使う
bit 14	<b>SIGN7:</b> AN7 符号付きデータモードビット 1 = AN7 を符号付きデータモードで使う 0 = AN7 を符号なしデータモードで使う
bit 13	<b>DIFF6:</b> AN6 モードビット 1 = AN6 を差動モードで使う 0 = AN6 をシングルエンドモードで使う
bit 12	<b>SIGN6:</b> AN6 符号付きデータモードビット 1 = AN6 を符号付きデータモードで使う 0 = AN6 を符号なしデータモードで使う
bit 11	<b>DIFF5:</b> AN5 モードビット 1 = AN5 を差動モードで使う 0 = AN5 をシングルエンドモードで使う
bit 10	<b>SIGN5:</b> AN5 符号付きデータモードビット 1 = AN5 を符号付きデータモードで使う 0 = AN5 を符号なしデータモードで使う
bit 9	<b>DIFF4:</b> AN4 モードビット 1 = AN4 を差動モードで使う 0 = AN4 をシングルエンドモードで使う
bit 8	<b>SIGN4:</b> AN4 符号付きデータモードビット 1 = AN4 を符号付きデータモードで使う 0 = AN4 を符号なしデータモードで使う
bit 7	<b>DIFF3:</b> AN3 モードビット 1 = AN3 を差動モードで使う 0 = AN3 をシングルエンドモードで使う
bit 6	<b>SIGN3:</b> AN3 符号付きデータモードビット 1 = AN3 を符号付きデータモードで使う 0 = AN3 を符号なしデータモードで使う
bit 5	<b>DIFF2:</b> AN2 モードビット 1 = AN2 を差動モードで使う 0 = AN2 をシングルエンドモードで使う



# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-5: ADCIMCON1: ADC入 カモード制御レジスタ 1 (続 き)

- bit 4      **SIGN2:** AN2 符号付きデータモード ビット  
1 = AN2 を符号付きデータモードで使う  
0 = AN2 を符号なしデータモードで使う
- bit 3      **DIFF1:** AN1 モードビット  
1 = AN1 を差動モードで使う  
0 = AN1 をシングルエンド モードで使う
- bit 2      **SIGN1:** AN1 符号付きデータモード ビット  
1 = AN1 を符号付きデータモードで使う  
0 = AN1 を符号なしデータモードで使う
- bit 1      **DIFF0:** AN0 モードビット  
1 = AN0 を差動モードで使う  
0 = AN0 をシングルエンド モードで使う
- bit 0      **SIGN0:** AN0 符号付きデータモード ビット  
1 = AN0 を符号付きデータモードで使う  
0 = AN0 を符号なしデータモードで使う

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-6: ADCIMCON2: ADC入カモード制御レジスタ 2

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF31	SIGN31	DIFF30	SIGN30	DIFF29	SIGN29	DIFF28	SIGN28
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF27	SIGN27	DIFF26	SIGN26	DIFF25	SIGN25	DIFF24	SIGN24
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF23	SIGN23	DIFF22	SIGN22	DIFF21	SIGN21	DIFF20	SIGN20
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF19	SIGN19	DIFF18	SIGN18	DIFF17	SIGN17	DIFF16	SIGN16

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット                      U = 未実装、「0」として読み出し  
 -n = POR 時の値                      1 = ビットはセット                      0 = ビットはクリア                      x = ビットは未知

- bit 31        **DIFF31:** AN31 モードビット  
             1 = AN31 を差動モードで使う  
             0 = AN31 をシングルエンドモードで使う
- bit 30        **SIGN31:** AN31 符号付きデータモードビット  
             1 = AN31 を符号付きデータモードで使う  
             0 = AN31 を符号なしデータモードで使う
- bit 29        **DIFF30:** AN30 モードビット  
             1 = AN30 を差動モードで使う  
             0 = AN30 をシングルエンドモードで使う
- bit 28        **SIGN30:** AN30 符号付きデータモードビット  
             1 = AN30 を符号付きデータモードで使う  
             0 = AN30 を符号なしデータモードで使う
- bit 27        **DIFF29:** AN29 モードビット  
             1 = AN29 を差動モードで使う  
             0 = AN29 をシングルエンドモードで使う
- bit 26        **SIGN29:** AN29 符号付きデータモードビット  
             1 = AN29 を符号付きデータモードで使う  
             0 = AN29 を符号なしデータモードで使う
- bit 25        **DIFF28:** AN28 モードビット  
             1 = AN28 を差動モードで使う  
             0 = AN28 をシングルエンドモードで使う
- bit 24        **SIGN28:** AN28 符号付きデータモードビット  
             1 = AN28 を符号付きデータモードで使う  
             0 = AN28 を符号なしデータモードで使う
- bit 23        **DIFF27:** AN27 モードビット  
             1 = AN27 を差動モードで使う  
             0 = AN27 をシングルエンドモードで使う
- bit 22        **SIGN27:** AN27 符号付きデータモードビット  
             1 = AN27 を符号付きデータモードで使う  
             0 = AN27 を符号なしデータモードで使う

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-6: ADCIMCON2: ADC入カモード制御レジスタ 2 (続き)

bit 21	<b>DIFF26:</b> AN26 モードビット 1 = AN26 を差動モードで使う 0 = AN26 をシングルエンドモードで使う
bit 20	<b>SIGN26:</b> AN26 符号付きデータモードビット 1 = AN26 を符号付きデータモードで使う 0 = AN26 を符号なしデータモードで使う
bit 19	<b>DIFF25:</b> AN25 モードビット 1 = AN25 を差動モードで使う 0 = AN25 をシングルエンドモードで使う
bit 18	<b>SIGN25:</b> AN25 符号付きデータモードビット 1 = AN25 を符号付きデータモードで使う 0 = AN25 を符号なしデータモードで使う
bit 17	<b>DIFF24:</b> AN24 モードビット 1 = AN24 を差動モードで使う 0 = AN24 をシングルエンドモードで使う
bit 16	<b>SIGN24:</b> AN24 符号付きデータモードビット 1 = AN24 を符号付きデータモードで使う 0 = AN24 を符号なしデータモードで使う
bit 15	<b>DIFF23:</b> AN23 モードビット 1 = AN23 を差動モードで使う 0 = AN23 をシングルエンドモードで使う
bit 14	<b>SIGN23:</b> AN23 符号付きデータモードビット 1 = AN23 を符号付きデータモードで使う 0 = AN23 を符号なしデータモードで使う
bit 13	<b>DIFF22:</b> AN22 モードビット 1 = AN22 を差動モードで使う 0 = AN22 をシングルエンドモードで使う
bit 12	<b>SIGN22:</b> AN22 符号付きデータモードビット 1 = AN22 を符号付きデータモードで使う 0 = AN22 を符号なしデータモードで使う
bit 11	<b>DIFF21:</b> AN21 モードビット 1 = AN21 を差動モードで使う 0 = AN21 をシングルエンドモードで使う
bit 10	<b>SIGN21:</b> AN21 符号付きデータモードビット 1 = AN21 を符号付きデータモードで使う 0 = AN21 を符号なしデータモードで使う
bit 9	<b>DIFF20:</b> AN20 モードビット 1 = AN20 を差動モードで使う 0 = AN20 をシングルエンドモードで使う
bit 8	<b>SIGN20:</b> AN20 符号付きデータモードビット 1 = AN20 を符号付きデータモードで使う 0 = AN20 を符号なしデータモードで使う
bit 7	<b>DIFF19:</b> AN19 モードビット 1 = AN19 を差動モードで使う 0 = AN19 をシングルエンドモードで使う

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-6: ADCIMCON2: ADC入カモード制御レジスタ 2 (続 き)

bit 6	<b>SIGN19:</b> AN19 符号付きデータモード ビット 1 = AN19 を符号付きデータモードで使う 0 = AN19 を符号なしデータモードで使う
bit 5	<b>DIFF18:</b> AN18 モードビット 1 = AN18 を差動モードで使う 0 = AN18 をシングルエンド モードで使う
bit 4	<b>SIGN18:</b> AN18 符号付きデータモード ビット 1 = AN18 を符号付きデータモードで使う 0 = AN18 を符号なしデータモードで使う
bit 3	<b>DIFF17:</b> AN17 モードビット 1 = AN17 を差動モードで使う 0 = AN17 をシングルエンド モードで使う
bit 2	<b>SIGN17:</b> AN17 符号付きデータモード ビット 1 = AN17 を符号付きデータモードで使う 0 = AN17 を符号なしデータモードで使う
bit 1	<b>DIFF16:</b> AN16 モードビット 1 = AN16 を差動モードで使う 0 = AN16 をシングルエンド モードで使う
bit 0	<b>SIGN16:</b> AN16 符号付きデータモード ビット 1 = AN16 を符号付きデータモードで使う 0 = AN16 を符号なしデータモードで使う

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-7: ADCIMCON3: ADC入カモード制御レジスタ 3

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF43	SIGN43	DIFF42	SIGN42	DIFF41	SIGN41	DIFF40	SIGN40
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF39	SIGN39	DIFF38	SIGN38	DIFF37	SIGN37	DIFF36	SIGN36
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DIFF35	SIGN35	DIFF34	SIGN34	DIFF33	SIGN33	DIFF32	SIGN32

**凡例:**

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

- bit 31-24 **未実装:** 「0」として読み出し
- bit 23 **DIFF43:** AN43 モードビット  
1 = AN43 を差動モードで使う  
0 = AN43 をシングルエンドモードで使う
- bit 22 **SIGN43:** AN43 符号付きデータモードビット  
1 = AN43 を符号付きデータモードで使う  
0 = AN43 を符号なしデータモードで使う
- bit 21 **DIFF42:** AN42 モードビット  
1 = AN42 を差動モードで使う  
0 = AN42 をシングルエンドモードで使う
- bit 20 **SIGN42:** AN42 符号付きデータモードビット  
1 = AN42 を符号付きデータモードで使う  
0 = AN42 を符号なしデータモードで使う
- bit 19 **DIFF41:** AN41 モードビット  
1 = AN41 を差動モードで使う  
0 = AN41 をシングルエンドモードで使う
- bit 18 **SIGN41:** AN41 符号付きデータモードビット  
1 = AN41 を符号付きデータモードで使う  
0 = AN41 を符号なしデータモードで使う
- bit 17 **DIFF40:** AN40 モードビット  
1 = AN40 を差動モードで使う  
0 = AN40 をシングルエンドモードで使う
- bit 16 **SIGN40:** AN40 符号付きデータモードビット  
1 = AN40 を符号付きデータモードで使う  
0 = AN40 を符号なしデータモードで使う
- bit 15 **DIFF39:** AN39 モードビット  
1 = AN39 を差動モードで使う  
0 = AN39 をシングルエンドモードで使う
- bit 14 **SIGN39:** AN39 符号付きデータモードビット  
1 = AN39 を符号付きデータモードで使う  
0 = AN39 を符号なしデータモードで使う

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-7: ADCIMCON3: ADC入カモード制御レジスタ 3 (続き)

bit 13	<b>DIFF38:</b> AN38 モードビット 1 = AN38 を差動モードで使う 0 = AN38 をシングルエンドモードで使う
bit 12	<b>SIGN38:</b> AN38 符号付きデータモードビット 1 = AN38 を符号付きデータモードで使う 0 = AN38 を符号なしデータモードで使う
bit 11	<b>DIFF37:</b> AN37 モードビット 1 = AN37 を差動モードで使う 0 = AN37 をシングルエンドモードで使う
bit 10	<b>SIGN37:</b> AN37 符号付きデータモードビット 1 = AN37 を符号付きデータモードで使う 0 = AN37 を符号なしデータモードで使う
bit 9	<b>DIFF36:</b> AN36 モードビット 1 = AN36 を差動モードで使う 0 = AN36 をシングルエンドモードで使う
bit 8	<b>SIGN36:</b> AN36 符号付きデータモードビット 1 = AN36 を符号付きデータモードで使う 0 = AN36 を符号なしデータモードで使う
bit 7	<b>DIFF35:</b> AN35 モードビット 1 = AN35 を差動モードで使う 0 = AN35 をシングルエンドモードで使う
bit 6	<b>SIGN35:</b> AN35 符号付きデータモードビット 1 = AN35 を符号付きデータモードで使う 0 = AN35 を符号なしデータモードで使う
bit 5	<b>DIFF34:</b> AN34 モードビット 1 = AN34 を差動モードで使う 0 = AN34 をシングルエンドモードで使う
bit 4	<b>SIGN34:</b> AN34 符号付きデータモードビット 1 = AN34 を符号付きデータモードで使う 0 = AN34 を符号なしデータモードで使う
bit 3	<b>DIFF33:</b> AN33 モードビット 1 = AN33 を差動モードで使う 0 = AN33 をシングルエンドモードで使う
bit 2	<b>SIGN33:</b> AN33 符号付きデータモードビット 1 = AN33 を符号付きデータモードで使う 0 = AN33 を符号なしデータモードで使う
bit 1	<b>DIFF32:</b> AN32 モードビット 1 = AN32 を差動モードで使う 0 = AN32 をシングルエンドモードで使う
bit 0	<b>SIGN32:</b> AN32 符号付きデータモードビット 1 = AN32 を符号付きデータモードで使う 0 = AN32 を符号なしデータモードで使う

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-8: ADCGIRQEN1:ADC グローバル割り込みイネーブル レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	AGIEN31	AGIEN30	AGIEN29	AGIEN28	AGIEN27	AGIEN26	AGIEN25	AGIEN24
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	AGIEN23	AGIEN22	AGIEN21	AGIEN20	AGIEN19	AGIEN18	AGIEN17	AGIEN16
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	AGIEN15	AGIEN14	AGIEN13	AGIEN12	AGIEN11	AGIEN10	AGIEN9	AGIEN8
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	AGIEN7	AGIEN6	AGIEN5	AGIEN4	AGIEN3	AGIEN2	AGIEN1	AGIEN0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **AGIEN31:AGIEN0:** ADC グローバル割り込みイネーブルビット

- 1 = 選択したアナログ入力の割り込みを有効にする  
変換データが読み出し可能 (ADCDSTAT1 レジスタの ARDYx ビット (x = 31 ~ 0) = 「1」) になった時に割り込みを生成します。
- 0 = 割り込みを無効にする

レジスタ 29-9: ADCGIRQEN2: ADC グローバル割り込みイネーブル レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	AGIEN43	AGIEN42	AGIEN41	AGIEN40
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	AGIEN39	AGIEN38	AGIEN37	AGIEN36	AGIEN35	AGIEN34	AGIEN33	AGIEN32

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-12 **未実装:** 「0」として読み出し

bit 11-0 **AGIEN43:AGIEN32:** ADC グローバル割り込みイネーブルビット

- 1 = 選択したアナログ入力の割り込みを有効にする  
変換データが読み出し可能 (ADCDSTAT2 レジスタの ARDYx ビット (x = 43 ~ 32) = 「1」) になった時に割り込みを生成します。
- 0 = 割り込みを無効にする

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-10: ADCCSS1: ADC共通スキャン選択レジスタ 1

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CSS23	CSS22	CSS21	CSS20	CSS19	CSS18	CSS17	CSS16
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0      **CSS31:CSS0:** アナログ共通スキャン選択ビット  
 1 = ANx を入力スキャンに含める  
 0 = ANx は入力スキャンに含めない

**Note 1:** クラス 1 およびクラス 2 アナログ入力をスキャンに含める場合、対応する CSSx ビットをセットするだけでなく、それらの入力のトリガ源として STRIG 入力を選択する必要があります。STRIG オプションの選択については、ADCTRGx レジスタのビット定義を参照してください。

**2:** クラス 1 またはクラス 2 入力をスキャンに含める (CSSx ビットを「1」にセットし、TRGSRCx<4:0> ビットを STRIG モード ('0b11) に設定する) 場合、ユーザアプリケーションは、その入力に対して他のトリガ (ADCCON3 レジスタの RQCNVRT ビット、ハードウェア入力、デジタルフィルタを使ったトリガ) が生成される事を防ぐ必要があります。そうしないと、スキャンの挙動は予測不可能です。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-11: ADCCSS2: ADC共通スキャン選択レジスタ 2

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	CSS43	CSS42	CSS41	CSS40
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CSS39	CSS38	CSS37	CSS36	CSS35	CSS34	CSS33	CSS32

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット                      U = 未実装、「0」として読み出し  
 -n = POR 時の値                      1 = ビットはセット                      0 = ビットはクリア                      x = ビットは未知

bit 31-12    **未実装:** 「0」として読み出し

bit 11-0    **CSS43:CSS32:** アナログ共通スキャン選択ビット  
 アナログ入力 43 ~ 32 は常にクラス 3 です。そのため利用可能なトリガは 32 個のみです。  
 1 = ANx を入カスキャンに含める  
 0 = ANx は入カスキャンに含めない

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-12: ADCDSTAT1: ADCデータレディステータス レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	ARDY31	ARDY30	ARDY29	ARDY28	ARDY27	ARDY26	ARDY25	ARDY24
23:16	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	ARDY23	ARDY22	ARDY21	ARDY20	ARDY19	ARDY18	ARDY17	ARDY16
15:8	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	ARDY15	ARDY14	ARDY13	ARDY12	ARDY11	ARDY10	ARDY9	ARDY8
7:0	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	ARDY7	ARDY6	ARDY5	ARDY4	ARDY3	ARDY2	ARDY1	ARDY0

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31-0 **ARDY31:ARDY0:** アナログ入力変換データレディ ビット

1 = このビットは、データレジスタ内の変換データが読み出し可能になるとセットされます。

0 = このビットは、対応するデータレジスタが読み出されるとクリアされます。

## レジスタ 29-13: ADCDSTAT2: ADCデータレディステータス レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	—	—	—	—	ARDY43	ARDY42	ARDY41	ARDY40
7:0	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	ARDY39	ARDY38	ARDY37	ARDY36	ARDY35	ARDY34	ARDY33	ARDY32

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31-13 **未実装:** 「0」として読み出し

bit 11-0 **ARDY43:ARDY32:** アナログ入力変換データレディ ビット

1 = このビットは、データレジスタ内の変換データが読み出し可能になるとセットされます。

0 = このビットは、対応するデータレジスタが読み出されるとクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-14: ADCCMPENx: ADC デジタル コンパレータ「x」イネーブル レジスタ (x = 1 ~ 6)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CMPE31	CMPE30	CMPE29	CMPE28	CMPE27	CMPE26	CMPE25	CMPE24
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CMPE23	CMPE22	CMPE21	CMPE20	CMPE19	CMPE18	CMPE17	CMPE16
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CMPE15	CMPE14	CMPE13	CMPE12	CMPE11	CMPE10	CMPE9	CMPE8
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CMPE7	CMPE6	CMPE5	CMPE4	CMPE3	CMPE2	CMPE1	CMPE0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **CMPE31:CMPE0**: ADC デジタル コンパレータ x イネーブルビット

これらのビットをセットすると、対応するアナログ入力の変換結果がデジタル コンパレータによって処理されます。CMPE<sub>n</sub> ビットは入力 AN<sub>n</sub> に対応します。

**Note 1:** CMPE<sub>x</sub> は AN<sub>x</sub> (x = 0 ~ 31) に対応します (デジタル コンパレータ入力は AN0 ~ AN31 に制限されます)。  
**2:** デジタル コンパレータを有効 (ENDCMP = 1) にしたままこのレジスタ内のビットを変更した場合、モジュールの挙動は予測不可能です。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-15: ADCMPx: ADC デジタル コンパレータ 「x」 制限値レジスタ (x = 1 ~ 6)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DCMPHI<15:8> <sup>(1,2,3)</sup>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DCMPHI<7:0> <sup>(1,2,3)</sup>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DCMPLO<15:8> <sup>(1,2,3)</sup>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DCMPLO<7:0> <sup>(1,2,3)</sup>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット                      U = 未実装、「0」として読み出し  
 -n = POR 時の値                      1 = ビットはセット                      0 = ビットはクリア                      x = ビットは未知

bit 31-16 **DCMPHI<15:0>**: デジタル コンパレータ x 上限値ビット (1,2,3)  
 これらのビットは、デジタル コンパレータで A/D 変換データと比較する値の上限値を格納します。

bit 15-0 **DCMPLO<15:0>**: デジタル コンパレータ x 下限値ビット (1,2,3)  
 これらのビットは、デジタル コンパレータで A/D 変換データと比較する値の下限値を格納します。

- Note 1:** デジタル コンパレータを有効 (ENDCMP = 1) にしたままこれらのビットを変更した場合、モジュールの挙動が予測不可能になる事があります。
- 2:** 制限値のフォーマット (符号および小数設定) は、A/D 変換データのフォーマットと同じにする必要があります。
- 3:** CVD モードでデジタル コンパレータ 0 を使う場合、CVD 出力データは差動であり常に符号付きのため、DCMPHI<15:0> および DCMPLO<15:0> ビットは常に符号付きフォーマットで指定する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-16: ADCFLTRx: ADC デジタルフィルタ「x」レジスタ (x = 1~ 6)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0, HS, HC
	AFEN	DATA16EN	DFMODE	OVRSAM<2:0>			AFGIEN	AFRDY
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	CHNLID<4:0>				
15:8	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	FLTRDATA<15:8>							
7:0	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	FLTRDATA<7:0>							

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

- bit 31 **AFEN:** デジタルフィルタ x イネーブルビット  
 1 = デジタルフィルタを有効にする  
 0 = デジタルフィルタを無効にし、AFRDY ステータスビットをクリアする
- bit 30 **DATA16EN:** フィルタ有効データ長ビット  
 1 = フィルタ出力データの 16 ビット全てを有効にする  
 0 = 上位 12 ビットのみ有効にする (下位 4 ビットは 0)  
**Note:** このビットは、DFMODE = 1 (平均モード) または FRACT (ADCCON1<23>) = 1 (小数出力モード) の場合のみ有効です。
- bit **DFMODE:** ADC フィルタモード ビット  
 1 = フィルタ x を平均化モードで動作させる  
 0 = フィルタ x をオーバー サンプリング フィルタモードで動作させる (既定値)
- bit 28-26 **OVRSAM<2:0>:** オーバー サンプリング フィルタ レシオビット  
**DFMODE が「0」の場合:**  
 111 = 128 サンプル (積算値を 3 ビット右へシフト、出力データは 15.1 フォーマット)  
 110 = 32 サンプル (積算値を 2 ビット右へシフト、出力データは 14.1 フォーマット)  
 101 = 8 サンプル (積算値を 1 ビット右へシフト、出力データは 13.1 フォーマット)  
 100 = 2 サンプル (積算値を 0 ビット右へシフト、出力データは 12.1 フォーマット)  
 011 = 256 サンプル (積算値を 4 ビット右へシフト、出力データは 16 ビット)  
 010 = 64 サンプル (積算値を 3 ビット右へシフト、出力データは 15 ビット)  
 001 = 16 サンプル (積算値を 2 ビット右へシフト、出力データは 14 ビット)  
 000 = 4 サンプル (積算値を 1 ビット右へシフト、出力データは 13 ビット)
- DFMODE が「1」の場合:**  
 111 = 256 サンプル (256 サンプルを平均化)  
 110 = 128 サンプル (128 サンプルを平均化)  
 101 = 64 サンプル (64 サンプルを平均化)  
 100 = 32 サンプル (32 サンプルを平均化)  
 011 = 16 サンプル (16 サンプルを平均化)  
 010 = 8 サンプル (8 サンプルを平均化)  
 001 = 4 サンプル (4 サンプルを平均化)  
 000 = 2 サンプル (2 サンプルを平均化)
- bit 25 **AFGIEN:** デジタルフィルタ x 割り込みイネーブルビット  
 1 = デジタルフィルタ割り込みを有効にする (AFRDY ステータスビットにより生成)  
 0 = デジタルフィルタ割り込みを無効にする

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-16: ADCFLTRx: ADC デジタルフィルタ 「x」 レジスタ (x = 1 ~ 6) (続き)

- bit 24 **AFRDY**: デジタルフィルタ x データレディステータスビット  
1 = FLTRDATA<15:0> ビット内のデータは読み出し可能  
0 = データは読み出し可能ではない  
**Note:** このビットは、FLTRDATA<15:0> ビットを読み出すか、デジタルフィルタ モジュールを無効 (AFEN = 「0」) にするとクリアされます。
- bit 23-21 **未実装**: 「0」 として読み出し
- bit 20-16 **CHNLID<4:0>**: デジタルフィルタ アナログ入力選択ビット  
これらのビットは、オーバー サンプリング フィルタのデータ源として使うアナログ入力を指定します。  
11111 = 予約済み  
.  
.  
.  
01100 = 予約済み  
01011 = AN11  
.  
.  
.  
00001 = AN1  
00000 = AN0  
**Note:** 最初の 12 個のアナログ入力 (クラス 1 入力 (AN0 ~ AN11) とクラス 2 入力 (AN5 ~ AN11)) だけがデジタルフィルタを使えます。
- bit 15-0 **FLTRDATA<15:0>**: デジタルフィルタ x データ出力値ビット  
フィルタの出力データは、FRACT ビット (ADCCON1<23>) で設定した小数フォーマットに従います。フィルタを有効にしたまま FRACT ビットを変更しない必要があります。フィルタ動作が終了した後に FRACT ビットの設定を変更しても、FLTRDATA<15:0> ビットの値は新しいフォーマットへ更新されません。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-17: ADCTRG1: ADCトリガ源 1レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC3<4:0>				
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC2<4:0>				
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC1<4:0>				
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC0<4:0>				

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28-24 **TRGSRC3<4:0>**: アナログ入力 AN3 変換トリガ源選択ビット

- 11111 = 予約済み
- 11110 = 予約済み
- 11101 = CTMU イベント
- 11100 = 予約済み
- .
- .
- 01110 = 予約済み
- 01101 = CTMU イベント
- 01100 = コンパレータ 2 (C2OUT) <sup>(1)</sup>
- 01011 = コンパレータ 1 (C1OUT) <sup>(1)</sup>
- 01010 = OCMP5 <sup>(1)</sup>
- 01001 = OCMP3 <sup>(1)</sup>
- 01000 = OCMP1 <sup>(1)</sup>
- 00111 = TMR5 一致
- 00110 = TMR3 一致
- 00101 = TMR1 一致
- 00100 = INT0 外部割り込み
- 00011 = STRIG
- 00010 = グローバル ソフトウェア レベルトリガ (GLSWTRG)
- 00001 = グローバル ソフトウェア エッジトリガ (GSWTRG)
- 00000 = トリガなし

STRIG を選択した場合、STRGSRC<4:0> ビット (ADCCON1<20:16>) でトリガ源を選択し、さらに ADCCSSx レジスタで適切な CSS ビットをセットする必要があります。

bit 23-21 **未実装**: 「0」として読み出し

bit 20-16 **TRGSRC2<4:0>**: アナログ入力 AN2 変換トリガ源選択ビット

定義は bit 28-24 と同じです。

bit 15-13 **未実装**: 「0」として読み出し

bit 12-8 **TRGSRC1<4:0>**: アナログ入力 AN1 変換トリガ源選択ビット

定義は bit 28-24 と同じです。

bit 7-5 **未実装**: 「0」として読み出し

bit 4-0 **TRGSRC0<4:0>**: アナログ入力 AN0 変換トリガ源選択ビット

定義は bit 28-24 と同じです。

**Note 1:** モジュール出力信号の立ち上がりエッジで ADC 変換をトリガします。詳細は、[16.0「出カコンペア」](#) 内の [図 16-1](#) と、[32.0「コンパレータ」](#) 内の [図 32-1](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-18: ADCTRG2: ADCトリガ源 2レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC7<4:0>				
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC6<4:0>				
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC5<4:0>				
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC4<4:0>				

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28-24 **TRGSRC7<4:0>**: アナログ入力 AN7 変換トリガ源選択ビット

11111 = 予約済み  
 11110 = 予約済み  
 11101 = CTMU イベント  
 11100 = 予約済み  
 .  
 .  
 01110 = 予約済み  
 01101 = CTMU イベント  
 01100 = コンパレータ 2 (C2OUT) <sup>(1)</sup>  
 01011 = コンパレータ 1 (C1OUT) <sup>(1)</sup>  
 01010 = OCMP5 <sup>(1)</sup>  
 01001 = OCMP3 <sup>(1)</sup>  
 01000 = OCMP1 <sup>(1)</sup>  
 00111 = TMR5 一致  
 00110 = TMR3 一致  
 00101 = TMR1 一致  
 00100 = INT0 外部割り込み  
 00011 = STRIG  
 00010 = グローバル ソフトウェア レベルトリガ (GLSWTRG)  
 00001 = グローバル ソフトウェア エッジトリガ (GSWTRG)  
 00000 = トリガなし

STRIG を選択した場合、STRGSRC<4:0> ビット (ADCCON1<20:16>) でトリガ源を選択し、さらに ADCCSSx レジスタで適切な CSS ビットをセットする必要があります。

bit 23-21 **未実装**: 「0」として読み出し

bit 20-16 **TRGSRC6<4:0>**: アナログ入力 AN6 変換トリガ源選択ビット  
 定義は bit 28-24 と同じです。

bit 15-13 **未実装**: 「0」として読み出し

bit 12-8 **TRGSRC5<4:0>**: アナログ入力 AN5 変換トリガ源選択ビット  
 定義は bit 28-24 と同じです。

bit 7-5 **未実装**: 「0」として読み出し

bit 4-0 **TRGSRC4<4:0>**: アナログ入力 AN4 変換トリガ源選択ビット  
 定義は bit 28-24 と同じです。

**Note 1:** モジュール出力信号の立ち上がりエッジで ADC 変換をトリガします。詳細は、[16.0「出カコンペア」](#) 内の [図 16-1](#) と、[32.0「コンパレータ」](#) 内の [図 32-1](#) を参照してください。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-19: ADCTRG3: ADCトリガ源 3レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC11<4:0>				
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC10<4:0>				
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC9<4:0>				
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	TRGSRC8<4:0>				

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28-24 **TRGSRC11<4:0>**: アナログ入力 AN11 変換トリガ源選択ビット

11111 = 予約済み

11110 = 予約済み

11101 = CTMU イベント

11100 = 予約済み

⋮

⋮

01110 = 予約済み

01101 = CTMU イベント

01100 = コンパレータ 2 (C2OUT) <sup>(1)</sup>

01011 = コンパレータ 1 (C1OUT) <sup>(1)</sup>

01010 = OCMP5 <sup>(1)</sup>

01001 = OCMP3 <sup>(1)</sup>

01000 = OCMP1 <sup>(1)</sup>

00111 = TMR5 一致

00110 = TMR3 一致

00101 = TMR1 一致

00100 = INT0 外部割り込み

00011 = STRIG

00010 = グローバル ソフトウェア レベルトリガ (GLSWTRG)

00001 = グローバル ソフトウェア エッジトリガ (GSWTRG)

00000 = トリガなし

STRIG を選択した場合、STRGSRC<4:0> ビット (ADCCON1<20:16>) でトリガ源を選択し、さらに ADCCSSx レジスタで適切な CSS ビットをセットする必要があります。

bit 23-21 **未実装**: 「0」として読み出し

bit 20-16 **TRGSRC10<4:0>**: アナログ入力 AN10 変換トリガ源選択ビット

定義は bit 28-24 と同じです。

bit 15-13 **未実装**: 「0」として読み出し

bit 12-8 **TRGSRC9<4:0>**: アナログ入力 AN9 変換トリガ源選択ビット

定義は bit 28-24 と同じです。

bit 7-5 **未実装**: 「0」として読み出し

bit 4-0 **TRGSRC8<4:0>**: アナログ入力 AN8 変換トリガ源選択ビット

定義は bit 28-24 と同じです。

**Note 1:** モジュール出力信号の立ち上がりエッジで ADC 変換をトリガします。詳細は、[16.0「出力コンペア」](#) 内の [図 16-1](#) と、[32.0「コンパレータ」](#) 内の [図 32-1](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-20: ADCCMPCON1: ADC デジタル コンパレータ 1 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	CVDDATA<15:8>							
23:16	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	CVDDATA<7:0>							
15:8	U-0	U-0	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	AINID<5:0>							
7:0	R/W-0	R/W-0	R-0, HS, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31-16 **CVDDATA<15:0>**: CVD データ ステータスビット

CVD モードでは、デジタル コンパレータ 割り込みが生成されるたびに、これらのビットに CVD 差動出力データ (CVD の正と負の計測値の差分) が格納されます。これらのビット値は FRACT ビット (ADCCON1<23>) で指定したフォーマットに従い、常に符号付きです。

bit 15-14 **未実装**: 「0」として読み出し

bit 13-8 **AINID<5:0>**: デジタル コンパレータ 0 アナログ入力 ID ビット

デジタル コンパレータ イベントが発生 (DCMPED = 1) した時、これらのビットはデジタル コンパレータ 0 が監視中のアナログ入力を示します。

**Note:** 通常の ADC モードでは、アナログ入力 <31:0> のみがデジタル コンパレータ 0 で処理可能です。デジタル コンパレータ 0 は CVD モードもサポートします。CVD モードの場合、AINID<5:0> ビットはクラス 2 およびクラス 3 の全てのアナログ入力を示す事ができます。

111111 = 予約済み

⋮

101100 = 予約済み

101011 = AN43 を監視中

⋮

000001 = AN1 を監視中

000000 = AN0 を監視中

bit 7 **ENDCMP**: デジタル コンパレータ 0 イネーブルビット

1 = デジタル コンパレータ 0 を有効にする

0 = デジタル コンパレータ 0 を有効にせず、DCMPED ステータスビット (ADCCMP0CON<5>) をクリアする

bit 6 **DCMPGIEN**: デジタル コンパレータ 0 グローバル 割り込み イネーブルビット

1 = DCMPED ステータスビット (ADCCMP0CON<5>) がセットされた時にデジタル コンパレータ 0 割り込みを生成する

0 = デジタル コンパレータ 0 割り込みを無効にする

bit 5 **DCMPED**: デジタル コンパレータ 0 「真」出力 イベント ステータスビット

デジタル コンパレータ が「真」を出力する論理条件は IEBTWN、IEHIHI、IEHILO、IELOHI、IELOLO ビットで定義します。

**Note:** AINID<5:0> ビットを読み出すかデジタル コンパレータ モジュールを無効 (ENDCMP = 「0」) にすると、このビットはクリアされます。

1 = デジタル コンパレータ 0 「真」出力 イベント (コンパレータ 出力は 「1」) が発生した

0 = デジタル コンパレータ 0 出力は 「偽」 (コンパレータ 出力は 「0」)

bit 4 **IEBTWN**: LOW ~ HIGH 間 デジタル コンパレータ 0 イベント ビット

1 = DCMPLO<15:0> ≤ DATA<31:0> < DCMPHI<15:0> の場合にデジタル コンパレータ 0 イベントを生成する

0 = デジタル コンパレータ イベントを生成しない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-20: ADCCMPCON1: ADC デジタル コンパレータ 1 制御レジスタ

- bit 3 **IEHIHI:** HIGH/HIGH デジタル コンパレータ 0 イベントビット  
1 = DCMPhi<15:0> ≤ DATA<31:0> の場合にデジタル コンパレータ 0 イベントを生成する  
0 = イベントを生成しない
- bit 2 **IEHILO:** HIGH/LOW デジタル コンパレータ 0 イベントビット  
1 = DATA<31:0> < DCMPhi<15:0> の場合にデジタル コンパレータ 0 イベントを生成する  
0 = イベントを生成しない
- bit 1 **IELOHI:** LOW/HIGH デジタル コンパレータ 0 イベントビット  
1 = DCMpLo<15:0> ≤ DATA<31:0> の場合にデジタル コンパレータ 0 イベントを生成する  
0 = イベントを生成しない
- bit 0 **IELOLO:** LOW/LOW デジタル コンパレータ 0 イベントビット  
1 = DATA<31:0> < DCMpLo<15:0> の場合にデジタル コンパレータ 0 イベントを生成する  
0 = イベントを生成しない

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-21: ADCCMPCONx: ADC デジタル コンパレータ 「x」 制御レジスタ (x = 2 ~ 6)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	R-0, HS, HC AINID<4:0>				
7:0	R/W-0 ENDCMP	R/W-0 DCMPGIEN	R-0, HS, HC DCMPED	R/W-0 IEBTWN	R/W-0 IEHIHI	R/W-0 IEHILO	R/W-0 IELOHI	R/W-0 IELOLO

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31-13 **未実装:** 「0」として読み出し

bit 12-8 **AINID<4:0>:** デジタル コンパレータ x アナログ入力 ID ビット

デジタル コンパレータ イベントが発生 (DCMPED = 1) した時、これらのビットはデジタル コンパレータが監視中のアナログ入力を示します。

**Note:** アナログ入力 <31:0> のみがデジタル コンパレータ モジュール x (x = 1 ~ 5) で処理できます。

11111 = AN31 を監視中

11110 = AN30 を監視中

.

.

00001 = AN1 を監視中

00000 = AN0 を監視中

bit 7 **ENDCMP:** デジタル コンパレータ x イネーブルビット

1 = デジタル コンパレータ x を有効にする

0 = デジタル コンパレータ x を有効にせず、DCMPED ステータスビット (ADCCMPxCON<5>) をクリアする

bit 6 **DCMPGIEN:** デジタル コンパレータ x 割り込みイネーブルビット

1 = DCMPED ステータスビット (ADCCMPxCON<5>) がセットされた時にデジタル コンパレータ x 割り込みを生成する

0 = デジタル コンパレータ x 割り込みを無効にする

bit 5 **DCMPED:** デジタル コンパレータ x 「真」 出力イベント ステータスビット

デジタル コンパレータが「真」を出力する論理条件は IEBTWN、IEHIHI、IEHILO、IELOHI、IELOLO ビットで定義します。

**Note:** AINID<5:0> ビット (ADCCMP0CON<13:8>) を読み出すかデジタル コンパレータ モジュールを無効 (ENDCMP = 「0」) にすると、このビットはクリアされます。

1 = デジタル コンパレータ x 「真」 出力イベント (コンパレータ出力は 「1」) が発生した

0 = デジタル コンパレータ x 出力は 「偽」 (コンパレータ出力は 「0」)

bit 4 **IEBTWN:** LOW ~ HIGH 間デジタル コンパレータ x イベントビット

1 = DCMPILO<15:0> ≤ DATA<31:0> < DCMPHI<15:0> の場合にデジタル コンパレータ x イベントを生成する

0 = デジタル コンパレータ イベントを生成しない

bit 3 **IEHIHI:** HIGH/HIGH デジタル コンパレータ x イベントビット

1 = DCMPHI<15:0> ≤ DATA<31:0> の場合にデジタル コンパレータ x イベントを生成する

0 = イベントを生成しない

bit 2 **IEHILO:** HIGH/LOW デジタル コンパレータ x イベントビット

1 = DATA<31:0> < DCMPHI<15:0> の場合にデジタル コンパレータ x イベントを生成する

0 = イベントを生成しない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-21: ADCCMPCONx: ADC デジタル コンパレータ 「x」 制御レジスタ (x = 2~ 6) (続 き)

- bit 1     **IELOHI:** LOW/HIGH デジタル コンパレータ x イベントビット  
          1 = DCMPL0<15:0> ≤ DATA<31:0> の場合にデジタル コンパレータ x イベントを生成する  
          0 = イベントを生成しない
- bit 0     **IELOLO:** LOW/LOW デジタル コンパレータ x イベントビット  
          1 = DATA<31:0> < DCMPL0<15:0> の場合にデジタル コンパレータ x イベントを生成する  
          0 = イベントを生成しない

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-22: ADCFSTAT: ADC FIFO ステータス レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FEN	—	—	ADC4EN	ADC3EN	ADC2EN	ADC1EN	ADC0EN
23:16	R/W-0	R-0, HS, HC	R-0, HS, HC	U-0	U-0	U-0	U-0	U-0
	FIEN	FRDY	FWROVERR	—	—	—	—	—
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	FCNT<7:0>							
7:0	R-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
	FSIGN	—	—	—	—	ADCID<2:0>		

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31      **FEN:** FIFO イネーブルビット

1 = FIFO を有効にする  
 0 = FIFO を無効にする (FIFO にデータを格納しない)

bit 30-29      **未実装:** 「0」として読み出し

bit 28-24      **ADC4EN:ADC0EN:** ADCx イネーブルビット (x = 0 ~ 4)

1 = ADCx の変換出力データを FIFO に格納する  
 0 = ADCx の変換出力データを FIFO に格納しない

**Note:** FIFO の使用中も、出力データは対応する出力データレジスタ (ADCDATAx) にも格納されます。

bit 23      **FIEN:** FIFO 割り込みイネーブルビット

1 = FIFO 割り込みを有効にする (FRDY ビットがセットされた時に割り込みを生成する)  
 0 = FIFO 割り込みを無効にする

bit 22      **FRDY:** FIFO データレディ割り込みステータスビット

1 = FIFO 内に読み出せるデータが存在する  
 0 = FIFO に読み出せるデータは存在しない

**Note:** このビットは、ADCFIFO 内の FIFO 出力データが全て読み出されて FIFO 内に読み出し可能なデータが残っていない (FIFO がエンプティの) 時にクリアされます。

bit 21      **FWROVERR:** FIFO 書き込みオーバーフロー エラー ステータスビット

1 = FIFO で書き込みオーバーフロー エラーが発生した (リングバッファ)  
 0 = FIFO で書き込みオーバーフロー エラーは発生していない

**Note:** このビットは、ソフトウェアが ADCFSTAT<23:16> を読み出した後にクリアされます。

bit 15-8      **FCNT<7:0>:** FIFO データエン트리数ステータスビット

これらのビット値は FIFO 内のデータエントリの数を示します。

bit 7      **FSIGN:** FIFO 符号設定ビット

このビットは、ADCFIFO レジスタに格納されているデータの符号を示します。

bit 6-3      **未実装:** 「0」として読み出し

bit 2-0      **ADCID<2:0>:** ADCx ID ビット (x = 0 ~ 6)

これらのビットは、FIFO にどの ADC モジュールのデータが格納されているのかを示します。

111 = 予約済み  
 110 = 予約済み  
 100 = FIFO は ADC4 の変換データを格納している  
 .  
 .  
 .

000 = FIFO は ADC0 の変換データを格納している

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-23: ADCFIFO: ADC FIFO データレジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DATA<31:24>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DATA<23:16>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DATA<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DATA<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0      **DATA<31:0>**: FIFO データ出力値ビット

**Note:** 専用 ADC モジュールの入力源として代替入力を使う場合も、変換結果は既定値入力と同じデータ出力レジスタに格納されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-24: ADCBASE: ADCベースレジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADCBASE<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADCBASE<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0      **未実装:** 「0」として読み出し

bit 15-0      **ADCBASE<15:0>:** ADC ISR ベースアドレス ビット

このレジスタは、ユーザの ADC ISR ジャンプテーブルのベースアドレスを格納します。ADCDSTAT1 および ADCDSTAT2 レジスタの ARDYx ステータスビットを ADCCON1 レジスタの IRQVS<2:0> ビットが指定するビット数だけ左へシフトし、その値を ADCBASE<15:0> ビットの値に加算する事により、割り込みベクタアドレスが決まります。

割り込みベクタアドレス = ADCBASE の読み値

ADCBASE の読み値 = ADCBASE に書き込まれた値 + x << IRQVS<2:0>

(「x」は、ADCDSTAT1 または ADCDSTAT2 レジスタ内でアクティブな最小 (最高優先度) の入力 ID)



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-25: ADCDATAx: ADC出力データレジスタ「x」(x = 0 ~ 43)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DATA<31:24>							
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DATA<23:16>							
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DATA<15:8>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	DATA<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット                      U = 未実装、「0」として読み出し  
 -n = POR 時の値                      1 = ビットはセット                      0 = ビットはクリア                      x = ビットは未知

bit 31-0      **DATA<31:0>**: ADC 変換データ出力ビット

- Note 1:** 専用 ADC モジュールの入力源として代替入力を使う場合も、変換結果は既定値入力と同じデータ出力レジスタに格納されます。
- 2:** FRACT ビットを変更した後に ADCDATAx レジスタの値を読み出すと、データは FRACT ビットで指定したフォーマットに変換されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-26: ADCTRGSNS: ADCトリガ レベル/エッジ検出レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	LVL11	LVL10	LVL9	LVL8
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	LVL7	LVL6	LVL5	LVL4	LVL3	LVL2	LVL1	LVL0

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット                      U = 未実装、「0」として読み出し  
 -n = POR 時の値                      1 = ビットはセット                      0 = ビットはクリア                      x = ビットは未知

bit 31-12 **未実装:** 「0」として読み出し

bit 11-0 **LVL11:LVL0:** トリガ レベル/エッジ検出ビット

1 = アナログ入力をHIGHレベル検出でトリガする(トリガ信号がHIGHである間は繰り返しトリガする)  
 0 = アナログ入力を立ち上がりエッジ検出でトリガする(リセット後の既定値)

**Note 1:** このレジスタでは、アナログ入力0～11のトリガ検出方法を指定します。

**2:** IDが32以上のアナログ入力はクラス3に属するため、スキヤントリガだけが使えます。クラス3アナログ入力のスキヤントリガ検出方法(レベル/エッジ)はSTRGLVLビット(ADCCON1<3>)で定義します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-27: ADCxTIME: 専用 ADCx タイミング レジスタ (x = 0 ~ 4)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
	—	—	—	ADCEIS<2:0>			SELRES<1:0>	
23:16	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	ADCDIV<6:0>						
15:8	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	SAMC<9:8>	
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SAMC<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット                      U = 未実装、「0」として読み出し  
 -n = POR 時の値                      1 = ビットはセット                      0 = ビットはクリア                      x = ビットは未知

bit 31-29 **未実装:** 「0」として読み出し

bit 28-26 **ADCEIS<2:0>:** ADCx 早期割り込み選択ビット

111 = データ変換終了の 8 ADC クロックサイクル前にデータレディ割り込みを生成する

110 = データ変換終了の 7 ADC クロックサイクル前にデータレディ割り込みを生成する

・

・

001 = データ変換終了の 2 ADC クロックサイクル前にデータレディ割り込みを生成する

000 = データ変換終了の 1 ADC クロックサイクル前にデータレディ割り込みを生成する

**Note:** SELRES<1:0> ビット (ADCxTIME<25:24>) で 12 ビットまたは 10 ビット分解能を選択した場合、全てのオプション (「000」 ~ 「111」) が選択できます。8 ビット分解能を選択した場合、「000」 ~ 「101」のみ選択可能です。6 ビット分解能を選択した場合、「000」 ~ 「011」のみ選択可能です。

bit 25-24 **SELRES<1:0>:** ADCx 分解能選択ビット

11 = 12 ビット

10 = 10 ビット

01 = 8 ビット

00 = 6 ビット

**Note:** ADC の分解能を変更しても、対応する ADCDATAx レジスタ内の ADC 結果はシフトしません。結果は 12 ビット幅のまま、未使用の下位ビットが「0」に設定されます。例えば 6 ビット分解能の場合、ADCDATAx<5:0> は「0」に設定され、ADCDATAx<11:6> が ADC 結果を保持します。

bit 23 **未実装:** 「0」として読み出し

bit 22-16 **ADCDIV<6:0>:** ADCx クロック分周比ビット

これらのビットを使って ADC 制御クロック (周期 Tq) を分周する事で ADCx のクロック (TADx) を生成します。

1111111 = 254 \* Tq = TADx

・

・

0000011 = 6 \* Tq = TADx

0000010 = 4 \* Tq = TADx

0000001 = 2 \* Tq = TADx

0000000 = 予約済み

bit 15-10 **未実装:** 「0」として読み出し

bit 9-0 **SAMC<9:0>:** ADCx サンプリング時間ビット

TADx = 専用 ADC の変換クロックの周期 (ADCDIV<6:0> ビットで設定)

1111111111 = 1025 TADx

・

・

0000000001 = 3 TADx

0000000000 = 2 TADx

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-28: ADCEIEN1: ADC早期割り込みイネーブル レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EIEN31	EIEN30	EIEN29	EIEN28	EIEN27	EIEN26	EIEN25	EIEN24
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EIEN23	EIEN22	EIEN21	EIEN20	EIEN19	EIEN18	EIEN17	EIEN16
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EIEN15	EIEN14	EIEN13	EIEN12	EIEN11	EIEN10	EIEN9	EIEN8
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EIEN7	EIEN6	EIEN5	EIEN4	EIEN3	EIEN2	EIEN1	EIEN0

**凡例:** HS = ハードウェアでセット C = クリア可能ビット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-0 **EIEN31:EIEN0:** アナログ入力早期割り込みイネーブルビット

1 = 選択したアナログ入力の早期割り込みを有効にする

早期割り込みイベント (ADCEIEN1 レジスタの EIRDYx ビット (x = 31 ~ 0) = 「1」) が発生した時に割り込みを生成します。

0 = 割り込みを無効にする

## レジスタ 29-29: ADCEIEN2: ADC早期割り込みイネーブル レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	EIEN43	EIEN42	EIEN41	EIEN40
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EIEN39	EIEN38	EIEN37	EIEN36	EIEN35	EIEN34	EIEN33	EIEN32

**凡例:** HS = ハードウェアでセット C = クリア可能ビット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-11 **未実装:** 「0」として読み出し

bit 11-0 **EIEN43:EIEN32:** アナログ入力早期割り込みイネーブルビット

1 = 選択したアナログ入力の早期割り込みを有効にする

早期割り込みイベント (ADCEIEN2 レジスタの EIRDYx ビット (x = 43 ~ 32) = 「1」) が発生した時に割り込みを生成します。

0 = 割り込みを無効にする

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-30: ADCEISTAT1: ADC早期割り込みステータス レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	EIRDY31	EIRDY30	EIRDY29	EIRDY28	EIRDY27	EIRDY26	EIRDY25	EIRDY24
23:16	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	EIRDY23	EIRDY22	EIRDY21	EIRDY20	EIRDY19	EIRDY18	EIRDY17	EIRDY16
15:8	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	EIRDY15	EIRDY14	EIRDY13	EIRDY12	EIRDY11	EIRDY10	EIRDY9	EIRDY8
7:0	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC	R-0, HS, HC
	EIRDY7	EIRDY6	EIRDY5	EIRDY4	EIRDY3	EIRDY2	EIRDY1	EIRDY0

<b>凡例:</b>	HS = ハードウェアでセット	C = クリア可能ビット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

bit 31-0 **EIRDY31:EIRDY0:** アナログ入力早期割り込みレディビット

1 = このビットは、指定されたアナログ入力で早期割り込みイベントが発生した時にセットされます。ADCEIEN1 レジスタで早期割り込みを有効にしている場合、割り込みが生成されます。クラス 1 アナログ入力では、このビットは ADCxTIME レジスタの ADCEIS<2:0> ビットの設定に従ってセットされます。共有 ADC モジュールでは、このビットは ADCCON2 レジスタの ADCEIS<2:0> ビットの設定に従ってセットされます。

0 = 割り込みを無効にする

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-31: ADCEI2STAT2: ADC早期割り込みステータス レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	R-0, HS, HC EIRDY43	R-0, HS, HC EIRDY42	R-0, HS, HC EIRDY41	R-0, HS, HC EIRDY40
7:0	R-0, HS, HC EIRDY39	R-0, HS, HC EIRDY38	R-0, HS, HC EIRDY37	R-0, HS, HC EIRDY36	R-0, HS, HC EIRDY35	R-0, HS, HC EIRDY34	R-0, HS, HC EIRDY33	R-0, HS, HC EIRDY32

**凡例:** HS = ハードウェアでセット C = クリア可能ビット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-12 **未実装:** 「0」として読み出し

bit 11-0 **EIRDY43:EIRDY32:** アナログ入力早期割り込みレディビット

1 = このビットは、指定されたアナログ入力で早期割り込みイベントが発生した時にセットされます。ADCEIEN2 レジスタで早期割り込みを有効にしている場合、割り込みが生成されます。クラス 1 アナログ入力では、このビットは ADCxTIME レジスタの ADCEIS<2:0> ビットの設定に従ってセットされます。共有 ADC モジュールでは、このビットは ADCCON2 レジスタの ADCEIS<2:0> ビットの設定に従ってセットされます。

0 = 割り込みを無効にする

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 29-32: ADCANCON: ADCアナログウォームアップ制御レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0
	WKUPCLKCNT<3:0>							
23:16	R/W-0 WKIEN7	U-0 —	U-0 —	R/W-0 WKIEN4	R/W-0 WKIEN3	R/W-0 WKIEN2	R/W-0 WKIEN1	R/W-0 WKIEN0
15:8	R-0, HS, HC WKRDY7	U-0 —	U-0 —	R-0, HS, HC WKRDY4	R-0, HS, HC WKRDY3	R-0, HS, HC WKRDY2	R-0, HS, HC WKRDY1	R-0, HS, HC WKRDY0
7:0	R/W-0 ANEN7	U-0 —	U-0 —	R/W-0 ANEN4	R/W-0 ANEN3	R/W-0 ANEN2	R/W-0 ANEN1	R/W-0 ANEN0

**凡例:** HS = ハードウェアでセット C = クリア可能ビット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-28 **未実装:** 「0」として読み出し

bit 27-24 **WKUPCLKCNT<3:0>:** 復帰クロック数ビット

これらのビットは、ADC モジュールが変換を実行する前のウォームアップに必要な ADC クロック数を指定します。クロックは各 ADC に固有ですが、WKUPCLKCNT ビットは全ての ADC モジュールに共通です。

1111 =  $2^{15}$  = 32,768 クロック

⋮

0110 =  $2^6$  = 64 クロック

0101 =  $2^5$  = 32 クロック

0100 =  $2^4$  = 16 クロック

0011 =  $2^3$  = 8 クロック

0010 =  $2^2$  = 4 クロック

0001 =  $2^1$  = 2 クロック

0000 =  $2^0$  = 1 クロック

bit 23 **WKIEN7:** 共有 ADC (ADC7) 復帰割り込みイネーブルビット

1 = 割り込みを有効にする (WKRDY7 ステータスビットがセットされた時に割り込みを生成する)

0 = 割り込みを無効にする

bit 22-21 **未実装:** 「0」として読み出し

bit 20-16 **WKIEN4:WKIEN0:** ADC4 ~ ADC0 復帰割り込みイネーブルビット

1 = 割り込みを有効にする (WKRDYx ステータスビットがセットされた時に割り込みを生成する)

0 = 割り込みを無効にする

bit 15 **WKRDY7:** 共有 ADC (ADC7) 復帰ステータスビット

1 = ADC7 のアナログおよびバイアス回路の動作準備は完了している (ANEN7 を「1」にセットしてから WKUPCLKCNT<3:0> で指定したウォームアップ期間 ( $2^{WKUPEXP}$  クロック) が過ぎた)

0 = ADC7 のアナログおよびバイアス回路の動作準備は完了していない

**Note:** このビットは、ANEN7 ビットをクリアした時にハードウェアによってクリアされます。

bit 14-13 **未実装:** 「0」として読み出し

bit 12-8 **WKRDY4:WKRDY0:** ADC4 ~ ADC0 復帰ステータスビット

1 = ADCx のアナログおよびバイアス回路の動作準備は完了している (ANENx を「1」にセットしてから WKUPCLKCNT<3:0> で指定したウォームアップ期間 ( $2^{WKUPEXP}$  クロック) が過ぎた)

0 = ADCx のアナログおよびバイアス回路の動作準備は完了していない

**Note:** これらビットは、ANENx ビットをクリアした時にハードウェアによってクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 29-32: ADCANCON: ADC アナログ ウォームアップ制御レジスタ (続き)

- bit 7      **ANEN7:** 共有 ADC (ADC7) アナログおよびバイアス回路イネーブルビット  
1 = アナログおよびバイアス回路を有効にする  
アナログおよびバイアス回路を有効にした後、ADC モジュールにはウォームアップ期間 (WKUPCLKCNT<3:0> ビットで定義) が必要です。  
0 = アナログおよびバイアス回路を無効にする
- bit 5-6    **未実装:** 「0」として読み出し
- bit 4-0    **ANEN4:ANEN0:** ADC4 ~ ADC0 アナログおよびバイアス回路イネーブルビット  
1 = アナログおよびバイアス回路を有効にする  
アナログおよびバイアス回路を有効にした後、ADC モジュールにはウォームアップ期間 (WKUPCLKCNT<3:0> ビットで定義) が必要です。  
0 = アナログおよびバイアス回路を無効にする



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-33: ADCxCFG: ADCxコンフィグレーションレジスタ (x = 1~ 4, 7)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADCCFG<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADCCFG<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADCCFG<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADCCFG<7:0>							

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0      **ADCCFG<31:0>**: ADC モジュール コンフィグレーション データビット

ADC モジュールを有効にする前に (ADC の初期化中に)、DEVADCx に保存されている値をソフトウェアでこれらのレジスタに書き込む必要があります。

**Note:** これらのビットは、ADCANCON レジスタ内の対応する ANENx ビットがクリアされている場合にのみ変更できます。これらは製品検査時に設定された校正値であり、DEVADCx ヒューズビット ([レジスタ 41-8](#) 参照) によってユーザに提供されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 29-34: ADCSYSCFG1: ADCシステム コンフィグレーション レジスタ 1

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS
AN<31:23>								
23:16	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS
AN<23:16>								
15:8	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS
AN<15:8>								
7:0	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS
AN<7:0>								

<b>凡例:</b>	HS = ハードウェアでセット	C = クリア可能ビット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31-0 **AN<31:0>**: ADC アナログ入力ビット

これらのビットは、システム コンフィグレーションに従ってブートアップ中に更新されます。ユーザアプリケーションは、これらのビットを読み出す事でデバイス内の各アナログ入力を利用可能かどうか判断できます。

## レジスタ 29-35: ADCSYSCFG2: ADCシステム コンフィグレーション レジスタ 2

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
— — — — — — — —								
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
— — — — — — — —								
15:8	U-0	U-0	U-0	U-0	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS
— — — — —					AN<43:40>			
7:0	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS	R-0, HC, HS
AN<39:32>								

<b>凡例:</b>	HS = ハードウェアでセット	C = クリア可能ビット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31-13 **未実装**: 「0」として読み出し

bit 12-0 **AN<43:32>**: ADC アナログ入力ビット

これらのビットは、システム コンフィグレーションに従ってブートアップ中に更新されます。ユーザアプリケーションは、これらのビットを読み出す事でデバイス内の各アナログ入力を利用可能かどうか判断できます。

# PIC32MZ グラフィック (DA) ファミリ

## 30.0 CAN (コントローラ エリア ネットワーク)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 34. Controller Area Network (CAN)』(DS60001154) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

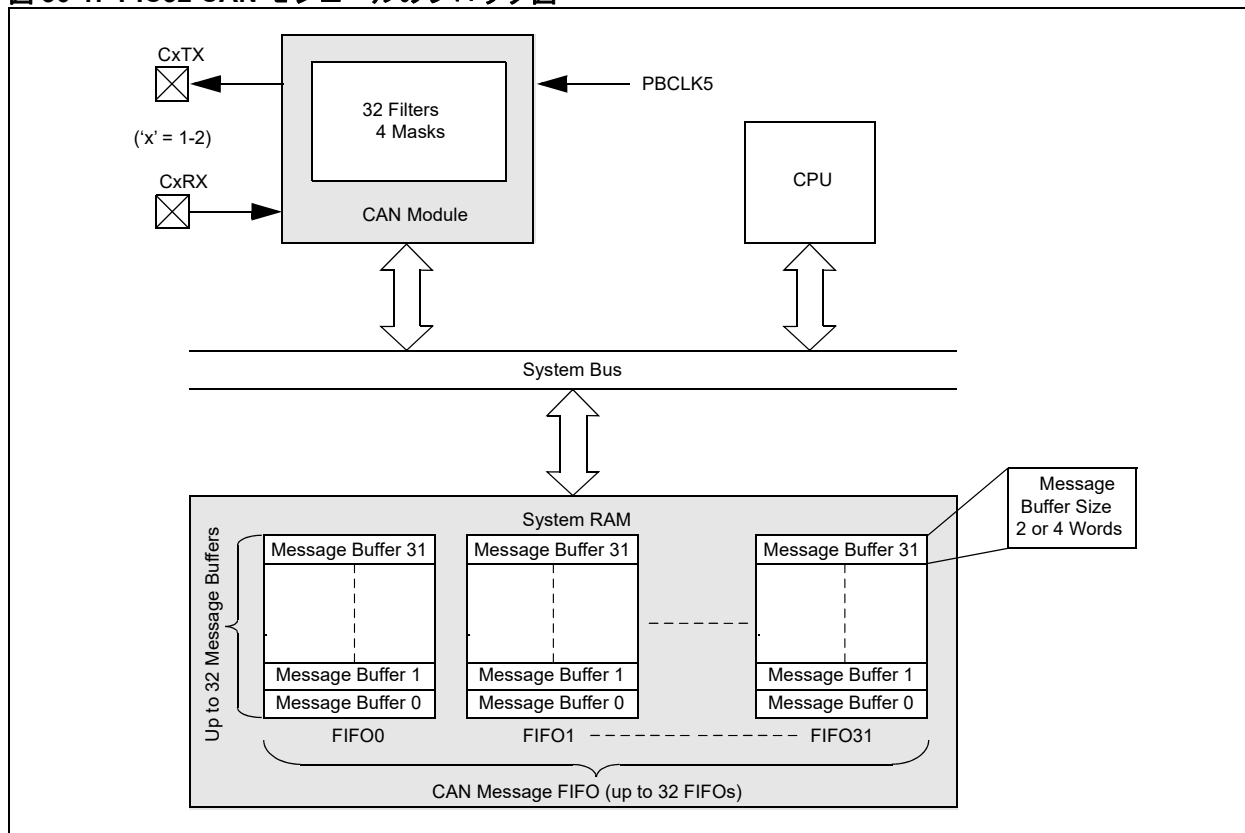
CAN モジュールの主な特長は以下の通りです。

- 規格への準拠:
  - CAN 2.0B に完全準拠
  - ビットレートを設定可能 (最大 1 Mbps)
- メッセージの送受信:
  - 32 個のメッセージ FIFO
  - 各 FIFO で最大 32 個 (合計で 1024 個) のメッセージを格納可能

- FIFO は送信メッセージ FIFO または受信メッセージ FIFO のどちらかに設定可能
- 送信に使うメッセージ FIFO の優先度はユーザ定義可能
- 32 個のアクセプタンス フィルタによるメッセージのフィルタ処理
- メッセージ フィルタ処理用の 4 個のアクセプタンス フィルタマスク レジスタ
- リモート送信要求に対する自動応答
- DeviceNet™ アドレッシングをサポート
- その他の特長:
  - セルフテスト、システム診断、バス監視用のループバック モード、リッスンオール メッセージ モード、リッスンオンリー モード
  - 低消費電力動作モード
  - CAN モジュールは PIC32 システムバス上でマスタとして機能
  - DMA は不要
  - 専用のタイムスタンプ タイマ
  - 専用の DMA チャンネル
  - データのみメッセージ受信モード

図 30-1 に、CAN モジュールの一般的な構造を示します。

図 30-1: PIC32 CAN モジュールのブロック図



## 30.1 CAN 制御レジスタ

**Note:** レジスタ名内の「i」は CAN1 または CAN2 を表します。

表 30-1: PIC32MZXXXECF/PIC32MZXXXECH の CAN1 レジスタの一覧

仮想アドレス (BF8_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																ト キ セ ツ ト		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0			
0000	C1CON	31:16	—	—	—	—	ABAT	REQOP<2:0>				OPMOD<2:0>				CANCAP	—	—	—	—	0480
		15:0	ON	—	SIDLE	—	CANBUSY	—	—	—	—	—	—	DNCNT<4:0>				0000			
0010	C1CFG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	SEG2PH<2:0>				0000	
		15:0	SEG2PHTS	SAM	SEG1PH<2:0>				PRSEG<2:0>				SJW<1:0>				BRP<5:0>				0000
0020	C1INT	31:16	IVRIE	WAKIE	CERRIE	SERRIE	RBOVIE	—	—	—	—	—	—	—	—	MODIE	CTMRIE	RBIE	TBIE	0000	
		15:0	IVRIF	WAKIF	CERRIF	SERRIF	RBOVIF	—	—	—	—	—	—	—	—	MODIF	CTMRIF	RBIF	TBIF	0000	
0030	C1VEC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	FILHIT<4:0>				—	ICODE<6:0>				0040						
0040	C1TREC	31:16	—	—	—	—	—	—	—	—	—	—	TXBO	TXBP	RXBP	TXWARN	RXWARN	EWARN	0000		
		15:0	TERRCNT<7:0>				RERRCNT<7:0>				0000										
0050	C1FSTAT	31:16	FIFOIP31	FIFOIP30	FIFOIP29	FIFOIP28	FIFOIP27	FIFOIP26	FIFOIP25	FIFOIP24	FIFOIP23	FIFOIP22	FIFOIP21	FIFOIP20	FIFOIP19	FIFOIP18	FIFOIP17	FIFOIP16	0000		
		15:0	FIFOIP15	FIFOIP14	FIFOIP13	FIFOIP12	FIFOIP11	FIFOIP10	FIFOIP9	FIFOIP8	FIFOIP7	FIFOIP6	FIFOIP5	FIFOIP4	FIFOIP3	FIFOIP2	FIFOIP1	FIFOIP0	0000		
0060	C1RXOVF	31:16	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16	0000		
		15:0	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0	0000		
0070	C1TMR	31:16	CANTS<15:0>																0000		
		15:0	CANTSPRE<15:0>																0000		
0080	C1RXM0	31:16	SID<10:0>												—	MIDE	—	EID<17:16>		xxxx	
		15:0	EID<15:0>																xxxx		
0090	C1RXM1	31:16	SID<10:0>												—	MIDE	—	EID<17:16>		xxxx	
		15:0	EID<15:0>																xxxx		
00A0	C1RXM2	31:16	SID<10:0>												—	MIDE	—	EID<17:16>		xxxx	
		15:0	EID<15:0>																xxxx		
00B0	C1RXM3	31:16	SID<10:0>												—	MIDE	—	EID<17:16>		xxxx	
		15:0	EID<15:0>																xxxx		
00C0	C1FLTCON0	31:16	FLTEN3	MSEL3<1:0>				FSEL3<4:0>				FLTEN2	MSEL2<1:0>				FSEL2<4:0>				0000
		15:0	FLTEN1	MSEL1<1:0>				FSEL1<4:0>				FLTEN0	MSEL0<1:0>				FSEL0<4:0>				0000
00D0	C1FLTCON1	31:16	FLTEN7	MSEL7<1:0>				FSEL7<4:0>				FLTEN6	MSEL6<1:0>				FSEL6<4:0>				0000
		15:0	FLTEN5	MSEL5<1:0>				FSEL5<4:0>				FLTEN4	MSEL4<1:0>				FSEL4<4:0>				0000
00E0	C1FLTCON2	31:16	FLTEN11	MSEL11<1:0>				FSEL11<4:0>				FLTEN10	MSEL10<1:0>				FSEL10<4:0>				0000
		15:0	FLTEN9	MSEL9<1:0>				FSEL9<4:0>				FLTEN8	MSEL8<1:0>				FSEL8<4:0>				0000

**凡例:** x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

**Note1:** この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

表 30-1: PIC32MZXXXECF/PIC32MZXXXECH の CAN1 レジスタの一覧 (続き)

仮想アドレス (BF8#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット 値
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
00F0	C1FLTCON3	31:16	FLTEN15	MSEL15<1:0>				FSEL15<4:0>			FLTEN14	MSEL14<1:0>			FSEL14<4:0>			0000	
		15:0	FLTEN13	MSEL13<1:0>				FSEL13<4:0>			FLTEN12	MSEL12<1:0>			FSEL12<4:0>			0000	
0100	C1FLTCON4	31:16	FLTEN19	MSEL19<1:0>				FSEL19<4:0>			FLTEN18	MSEL18<1:0>			FSEL18<4:0>			0000	
		15:0	FLTEN17	MSEL17<1:0>				FSEL17<4:0>			FLTEN16	MSEL16<1:0>			FSEL16<4:0>			0000	
0110	C1FLTCON5	31:16	FLTEN23	MSEL23<1:0>				FSEL23<4:0>			FLTEN22	MSEL22<1:0>			FSEL22<4:0>			0000	
		15:0	FLTEN21	MSEL21<1:0>				FSEL21<4:0>			FLTEN20	MSEL20<1:0>			FSEL20<4:0>			0000	
0120	C1FLTCON6	31:16	FLTEN27	MSEL27<1:0>				FSEL27<4:0>			FLTEN26	MSEL26<1:0>			FSEL26<4:0>			0000	
		15:0	FLTEN25	MSEL25<1:0>				FSEL25<4:0>			FLTEN24	MSEL24<1:0>			FSEL24<4:0>			0000	
0130	C1FLTCON7	31:16	FLTEN31	MSEL31<1:0>				FSEL31<4:0>			FLTEN30	MSEL30<1:0>			FSEL30<4:0>			0000	
		15:0	FLTEN29	MSEL29<1:0>				FSEL29<4:0>			FLTEN28	MSEL28<1:0>			FSEL28<4:0>			0000	
0140-0330	C1RXFn (n = 0-31)	31:16	SID<10:0>										—	EXID	—	EID<17:16>	xxxx		
		15:0	EID<15:0>																xxxx
0340	C1FIFOBA	31:16																	0000
		15:0	C1FIFOBA<31:0>																0000
0350	C1FIFOCOn (n = 0)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	FSIZE<4:0>				0000
		15:0	—	FRESET	UINC	DONLY	—	—	—	—	TXEN	TXABAT	TXLARB	TXERR	TXREQ	RTREN	TXPRI<1:0>		0000
0360	C1FIFOINTn (n = 0)	31:16	—	—	—	—	—	TXNFULLIE	TXHALFIE	TXEMPTYIE	—	—	—	—	RXOVFLIE	RXFULLIE	RXHALFIE	RXN EMPTYIE	0000
		15:0	—	—	—	—	—	TXNFULLIF	TXHALFIF	TXEMPTYIF	—	—	—	—	RXOVFLIF	RXFULLIF	RXHALFIF	RXN EMPTYIF	0000
0370	C1FIFOUAn (n = 0)	31:16																	0000
		15:0	C1FIFOUA<31:0>																0000
0380	C1FIFOCIn (n = 0)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	C1FIFOCIn<4:0>				0000
0390-0B40	C1FIFOCOn C1FIFOINTn C1FIFOUAn C1FIFOCIn (n = 1-31)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	FSIZE<4:0>				0000
		15:0	—	FRESET	UINC	DONLY	—	—	—	—	TXEN	TXABAT	TXLARB	TXERR	TXREQ	RTREN	TXPRI<1:0>		0000
		31:16	—	—	—	—	—	TXNFULLIE	TXHALFIE	TXEMPTYIE	—	—	—	—	RXOVFLIE	RXFULLIE	RXHALFIE	RXN EMPTYIE	0000
		15:0	—	—	—	—	—	TXNFULLIF	TXHALFIF	TXEMPTYIF	—	—	—	—	RXOVFLIF	RXFULLIF	RXHALFIF	RXN EMPTYIF	0000
		31:16																	0000
		15:0	C1FIFOUA<31:0>																0000
		31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	C1FIFOCIn<4:0>				0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 30-2: PIC32MZXXXECF/PIC32MZXXXECH の CAN2 レジスタの一覧

仮想アドレス (BF88_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0			
1000	C2CON	31:16	—	—	—	—	ABAT	REQOP<2:0>				OPMOD<2:0>				CANCAP	—	—	—	—	0480
		15:0	ON	—	SIDLE	—	CANBUSY	—	—	—	—	—	—	DNCNT<4:0>							0000
1010	C2CFG	31:16	—	—	—	—	—	—	—	—	—	—	WAKFIL	—	—	—	SEG2PH<2:0>				0000
		15:0	SEG2PHTS	SAM	SEG1PH<2:0>				PRSEG<2:0>				SJW<1:0>		BRP<5:0>						0000
1020	C2INT	31:16	IVRIE	WAKIE	CERRIE	SERRIE	RBOVIE	—	—	—	—	—	—	—	MODIE	CTMRIE	RBIE	TBIE	0000		
		15:0	IVRIF	WAKIF	CERRIF	SERRIF	RBOVIF	—	—	—	—	—	—	—	MODIF	CTMRIF	RBIF	TBIF	0000		
1030	C2VEC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	—	—	—	—	FILHIT<4:0>				—	ICODE<6:0>							0040		
1040	C2TREC	31:16	—	—	—	—	—	—	—	—	—	—	TXBO	TXBP	RXBP	TXWARN	RXWARN	EWARN	0000		
		15:0	TERRCNT<7:0>								RERRCNT<7:0>								0000		
1050	C2FSTAT	31:16	FIFOIP31	FIFOIP30	FIFOIP29	FIFOIP28	FIFOIP27	FIFOIP26	FIFOIP25	FIFOIP24	FIFOIP23	FIFOIP22	FIFOIP21	FIFOIP20	FIFOIP19	FIFOIP18	FIFOIP17	FIFOIP16	0000		
		15:0	FIFOIP15	FIFOIP14	FIFOIP13	FIFOIP12	FIFOIP11	FIFOIP10	FIFOIP9	FIFOIP8	FIFOIP7	FIFOIP6	FIFOIP5	FIFOIP4	FIFOIP3	FIFOIP2	FIFOIP1	FIFOIP0	0000		
1060	C2RXOVF	31:16	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16	0000		
		15:0	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0	0000		
1070	C2TMR	31:16	CANTS<15:0>																0000		
		15:0	CANTSPRE<15:0>															0000			
1080	C2RXM0	31:16	SID<10:0>											—	MIDE	—	EID<17:16>			xxxxx	
		15:0	EID<15:0>																xxxxx		
10A0	C2RXM1	31:16	SID<10:0>											—	MIDE	—	EID<17:16>			xxxxx	
		15:0	EID<15:0>																xxxxx		
10B0	C2RXM2	31:16	SID<10:0>											—	MIDE	—	EID<17:16>			xxxxx	
		15:0	EID<15:0>																xxxxx		
10B0	C2RXM3	31:16	SID<10:0>											—	MIDE	—	EID<17:16>			xxxxx	
		15:0	EID<15:0>																xxxxx		
1010	C2FLTCON0	31:16	FLTEN3	MSEL3<1:0>				FSEL3<4:0>				FLTEN2	MSEL2<1:0>				FSEL2<4:0>				0000
		15:0	FLTEN1	MSEL1<1:0>				FSEL1<4:0>				FLTEN0	MSEL0<1:0>				FSEL0<4:0>				0000
10D0	C2FLTCON1	31:16	FLTEN7	MSEL7<1:0>				FSEL7<4:0>				FLTEN6	MSEL6<1:0>				FSEL6<4:0>				0000
		15:0	FLTEN5	MSEL5<1:0>				FSEL5<4:0>				FLTEN4	MSEL4<1:0>				FSEL4<4:0>				0000
10E0	C2FLTCON2	31:16	FLTEN11	MSEL11<1:0>				FSEL11<4:0>				FLTEN10	MSEL10<1:0>				FSEL10<4:0>				0000
		15:0	FLTEN9	MSEL9<1:0>				FSEL9<4:0>				FLTEN8	MSEL8<1:0>				FSEL8<4:0>				0000
10F0	C2FLTCON3	31:16	FLTEN15	MSEL15<1:0>				FSEL15<4:0>				FLTEN14	MSEL14<1:0>				FSEL14<4:0>				0000
		15:0	FLTEN13	MSEL13<1:0>				FSEL13<4:0>				FLTEN12	MSEL12<1:0>				FSEL12<4:0>				0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

表 30-2: PIC32MZXXXECF/PIC32MZXXXECH の CAN2 レジスタの一覧 (続き)

仮想アドレス (BF8_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット				
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0		
1100	C2FLTCON4	31:16	FLTEN19	MSEL19<1:0>						FSEL19<4:0>								FSEL18<4:0>	0000		
		15:0	FLTEN17	MSEL17<1:0>						FSEL17<4:0>									FSEL16<4:0>	0000	
1110	C2FLTCON5	31:16	FLTEN23	MSEL23<1:0>						FSEL23<4:0>								FSEL22<4:0>	0000		
		15:0	FLTEN21	MSEL21<1:0>						FSEL21<4:0>									FSEL20<4:0>	0000	
1120	C2FLTCON6	31:16	FLTEN27	MSEL27<1:0>						FSEL27<4:0>								FSEL26<4:0>	0000		
		15:0	FLTEN25	MSEL25<1:0>						FSEL25<4:0>									FSEL24<4:0>	0000	
1130	C2FLTCON7	31:16	FLTEN31	MSEL31<1:0>						FSEL31<4:0>								FSEL30<4:0>	0000		
		15:0	FLTEN29	MSEL29<1:0>						FSEL29<4:0>									FSEL28<4:0>	0000	
1140-1330	C2RXFn (n = 0-31)	31:16	SID<10:0>										—	EXID	—	EID<17:16>			xxxxx		
		15:0	EID<15:0>																	xxxxx	
1340	C2FIFOBA	31:16	C2FIFOBA<31:0>																	0000	
		15:0	C2FIFOBA<31:0>																	0000	
1350	C2FIFOCOn (n = 0)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FSIZE<4:0>			0000
		15:0	—	FRESET	UINC	DONLY	—	—	—	—	—	TXEN	TXABAT	TXLARB	TXERR	TXREQ	RTREN	TXPRI<1:0>			0000
1360	C2FIFOINTn (n = 0)	31:16	—	—	—	—	—	—	TXNFULLIE	TXHALFIE	TXEMPTYIE	—	—	—	—	RXOVFLIE	RXFULLIE	RXHALFIE	RXN EMPTYIE	0000	
		15:0	—	—	—	—	—	—	TXNFULLIF	TXHALFIF	TXEMPTYIF	—	—	—	—	RXOVFLIF	RXFULLIF	RXHALFIF	RXN EMPTYIF	0000	
1370	C2FIFOUAn (n = 0)	31:16	C2FIFOUA<31:0>																	0000	
		15:0	C2FIFOUA<31:0>																	0000	
1380	C2FIFOCIn (n = 0)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	C2FIFOCi<4:0>			0000		
1390-1B40	C2FIFOCOn C2FIFOINTn C2FIFOUAn C2FIFOCIn (n = 1-31)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FSIZE<4:0>			0000
		15:0	—	FRESET	UINC	DONLY	—	—	—	—	TXEN	TXABAT	TXLARB	TXERR	TXREQ	RTREN	TXPRI<1:0>			0000	
		31:16	—	—	—	—	—	—	TXNFULLIE	TXHALFIE	TXEMPTYIE	—	—	—	—	RXOVFLIE	RXFULLIE	RXHALFIE	RXN EMPTYIE	0000	
		15:0	—	—	—	—	—	—	TXNFULLIF	TXHALFIF	TXEMPTYIF	—	—	—	—	RXOVFLIF	RXFULLIF	RXHALFIF	RXN EMPTYIF	0000	
		31:16	C2FIFOUA<31:0>																	0000	
		15:0	C2FIFOUA<31:0>																	0000	
		31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	C2FIFOCi<4:0>			0000			

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-1: CiCON: CAN モジュール制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	S/HC-0	R/W-1	R/W-0	R/W-0
	—	—	—	—	ABAT	REQOP<2:0>		
23:16	R-1	R-0	R-0	R/W-0	U-0	U-0	U-0	U-0
	OPMOD<2:0>			CANCAP	—	—	—	—
15:8	R/W-0	U-0	R/W-0	U-0	R-0	U-0	U-0	U-0
	ON <sup>(1)</sup>	—	SIDLE	—	CANBUSY	—	—	—
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	DNCNT<4:0>				

**凡例:** HC = ハードウェアでクリア S = セット可能ビット  
R = 読み出し可能ビット W = 書き込み可能ビット P = プログラミング可能ビット r = 予約済みビット  
U = 未実装ビット -n = POR 時のビット値 (「0」、「1」、x = ビットは未知)

bit 31-28 **未実装:** 「0」として読み出し

bit 27 **ABAT:** 保留中の全送信中止ビット

1 = 全ての送信バッファに対して送信中止を指示する  
0 = このビットは、全ての送信が中止された時にモジュールによってクリアされます。

bit 26-24 **REQOP<2:0>:** 要求動作モードビット

111 = リッスンオール メッセージ モードに設定する  
110 = 予約済み (使用禁止)  
101 = 予約済み (使用禁止)  
100 = コンフィグレーション モードに設定する  
011 = リッスンオンリー モードに設定する  
010 = ループバック モードに設定する  
001 = 無効モードに設定する  
000 = 通常動作モードに設定する

bit 23-21 **OPMOD<2:0>:** 動作モードステータスビット

111 = モジュールはリッスンオール メッセージモード中  
110 = 予約済み  
101 = 予約済み  
100 = モジュールはコンフィグレーション モード中  
011 = モジュールはリッスンオンリー モード中  
010 = モジュールはループバック モード中  
001 = モジュールは無効モード中  
000 = モジュールは通常動作モード中

bit 20 **CANCAP:** CAN メッセージ受信タイムスタンプ タイマ キャプチャ イネーブルビット

1 = 有効メッセージを受信した時に CANTMR の値をメッセージと一緒に保存する  
0 = CAN メッセージ受信タイムスタンプ タイマ キャプチャを無効にする (CANTMR を停止して消費電力を節約できます)

bit 19-16 **未実装:** 「0」として読み出し

bit 15 **ON:** CAN ON ビット<sup>(1)</sup>

1 = CAN モジュールを有効にする  
0 = CAN モジュールを無効にする

bit 14 **未実装:** 「0」として読み出し

**Note 1:** ユーザアプリケーションでこのビットをクリアした場合、CAN モジュールは処理中のトランザクションを完了してから無効になります。従って、このビットをクリアしてからモジュールが無効になるまでに複数サイクルを要する場合があります。ユーザアプリケーションは、CANBUSY ビットをポーリングする事によってモジュールが無効になったかどうか確認する必要があります。



# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 30-1: CiCON: CAN モジュール制御レジスタ ( 続き )

- bit 13 **SIDLE:** アイドル中 CAN 停止ビット  
1 = システムがアイドルに移行した時に CAN の動作を停止する  
0 = システムがアイドルに移行しても CAN の動作を継続する
- bit 12 **未実装:** 「0」 として読み出し
- bit 11 **CANBUSY:** CAN モジュール ビジービット  
1 = CAN モジュールはアクティブ  
0 = CAN モジュールは完全に無効な状態
- bit 10-5 **未実装:** 「0」 として読み出し
- bit 4-0 **DNCNT<4:0>:** DeviceNet フィルタビット番号ビット  
10011-11111 = 無効な選択 (18 ビット以下のデータのみ EID と比較可能)  
10010 = データバイト 2 の bit 6 までと EID17 (CiRXFn<17>) を比較する  
.  
.  
.  
00001 = データバイト 0 の bit 7 と EID0 (CiRXFn<0>) を比較する  
00000 = データバイトを比較しない

**Note 1:** ユーザアプリケーションでこのビットをクリアした場合、CAN モジュールは処理中のトランザクションを完了してから無効になります。従って、このビットをクリアしてからモジュールが無効になるまでに複数サイクルを要する場合があります。ユーザアプリケーションは、CANBUSY ビットをポーリングする事によってモジュールが無効になったかどうか確認する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-2: CiCFG: CAN baud レート コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	WAKFIL	—	—	—	SEG2PH<2:0> <sup>(1,4)</sup>		
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SEG2PHTS <sup>(1)</sup>	SAM <sup>(2)</sup>	SEG1PH<2:0>			PRSEG<2:0>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SJW<1:0> <sup>(3)</sup>		BRP<5:0>					

**凡例:** HC = ハードウェアでクリア S = セット可能ビット  
R = 読み出し可能ビット W = 書き込み可能ビット P = プログラミング可能ビット r = 予約済みビット  
U = 未実装ビット -n = POR 時のビット値 (「0」、 「1」、 x = ビットは未知)

bit 31-23 **未実装:** 「0」として読み出し

bit 22 **WAKFIL:** CAN バスライン フィルタ イネーブルビット

- 1 = 復帰用に CAN バスライン フィルタを使う
- 0 = 復帰用に CAN バスライン フィルタを使わない

bit 21-19 **未実装:** 「0」として読み出し

bit 18-16 **SEG2PH<2:0>:** 位相バッファ セグメント 2 ビット<sup>(1,4)</sup>

111 = 長さは 8 x T<sub>Q</sub>

•  
•  
•

000 = 長さは 1 x T<sub>Q</sub>

bit 15 **SEG2PHTS:** 位相セグメント 2 時間選択ビット<sup>(1)</sup>

- 1 = 自由に設定可能
- 0 = SEG1PH の最大値または情報処理時間のどちらか大きい方

bit 14 **SAM:** CAN バスライン サンプリング ビット<sup>(2)</sup>

- 1 = サンプルポイントでバスラインを 3 回サンプリングする
- 0 = サンプルポイントでバスラインを 1 回サンプリングする

bit 13-11 **SEG1PH<2:0>:** 位相バッファ セグメント 1 ビット<sup>(4)</sup>

111 = 長さは 8 x T<sub>Q</sub>

•  
•  
•

000 = 長さは 1 x T<sub>Q</sub>

**Note 1:** SEG2PH ≤ SEG1PHSEG2PHTS をクリアした場合、SEG2PH は自動的に設定されます。

**2:** BRP < 2 の場合、3 回のビット サンプリングはできません。

**3:** SJW ≤ SEG2PH

**4:** ビットあたりの時間単位 (Time Quanta) は 7 より大きい事 (すなわち T<sub>QBIT</sub> > 7) が必要です。

**Note:** このレジスタは、CAN モジュールがコンフィグレーション モード中 (OPMOD<2:0> (CiCON<23:21>) = 100) の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-2: CiCFG: CAN baud レート コンフィグレーション レジスタ (続き)

bit 10-8 **PRSEG<2:0>**: 伝播時間セグメントビット<sup>(4)</sup>

111 = 長さは 8 x T<sub>Q</sub>

•

•

•

000 = 長さは 1 x T<sub>Q</sub>

bit 7-6 **SJW<1:0>**: 同期ジャンプ幅ビット<sup>(3)</sup>

11 = ジャンプ幅は 4 x T<sub>Q</sub>

10 = ジャンプ幅は 3 x T<sub>Q</sub>

01 = ジャンプ幅は 2 x T<sub>Q</sub>

00 = ジャンプ幅は 1 x T<sub>Q</sub>

bit 5-0 **BRP<5:0>**: baud レート プリスケーラ ビット

111111 = T<sub>Q</sub> = (2 x 64)/TPBCLK5

111110 = T<sub>Q</sub> = (2 x 63)/TPBCLK5

•

•

•

000001 = T<sub>Q</sub> = (2 x 2)/TPBCLK5

000000 = T<sub>Q</sub> = (2 x 1)/TPBCLK5

**Note 1:** SEG2PH ≤ SEG1PHSEG2PHTS をクリアした場合、SEG2PH は自動的に設定されます。

**2:** BRP < 2 の場合、3 回のビット サンプリングはできません。

**3:** SJW ≤ SEG2PH

**4:** ビットあたりの時間単位 (Time Quanta) は 7 より大きい事 (すなわち T<sub>QBIT</sub> > 7) が必要です。

**Note:** このレジスタは、CAN モジュールがコンフィグレーション モード中 (OPMOD<2:0> (CiCON<23:21>) = 100) の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-3: CiINT: CAN 割り込みレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
	IVRIE	WAKIE	CERRIE	SERRIE	RBOVIE	—	—	—
23:16	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	MODIE	CTMRIE	RBIE	TBIE
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
	IVRIF	WAKIF	CERRIF	SERRIF <sup>(1)</sup>	RBOVIF	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	MODIF	CTMRIF	RBIF	TBIF

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

- bit 31 **IVRIE:** 無効メッセージ受信割り込みイネーブルビット  
1 = この割り込み要求を有効にする  
0 = この割り込み要求を無効にする
- bit 30 **WAKIE:** CAN バス アクティビティ 復帰割り込みイネーブルビット  
1 = この割り込み要求を有効にする  
0 = この割り込み要求を無効にする
- bit 29 **CERRIE:** CAN バス エラー 割り込みイネーブルビット  
1 = この割り込み要求を有効にする  
0 = この割り込み要求を無効にする
- bit 28 **SERRIE:** システムエラー 割り込みイネーブルビット  
1 = この割り込み要求を有効にする  
0 = この割り込み要求を無効にする
- bit 27 **RBOVIE:** 受信バッファ オーバーフロー 割り込みイネーブルビット  
1 = この割り込み要求を有効にする  
0 = この割り込み要求を無効にする
- bit 26-20 **未実装:** 「0」として読み出し
- bit 19 **MODIE:** モード変化 割り込みイネーブルビット  
1 = この割り込み要求を有効にする  
0 = この割り込み要求を無効にする
- bit 18 **CTMRIE:** CAN タイムスタンプ タイマ 割り込みイネーブルビット  
1 = この割り込み要求を有効にする  
0 = この割り込み要求を無効にする
- bit 17 **RBIE:** 受信バッファ 割り込みイネーブルビット  
1 = この割り込み要求を有効にする  
0 = この割り込み要求を無効にする
- bit 16 **TBIE:** 送信バッファ 割り込みイネーブルビット  
1 = この割り込み要求を有効にする  
0 = この割り込み要求を無効にする
- bit 15 **IVRIF:** 無効メッセージ受信割り込みフラグビット  
1 = 無効メッセージ割り込みが発生した  
0 = 無効メッセージ割り込みは発生していない

**Note 1:** このビットは、ON ビット (CiCON<15>) をクリア / セットして CAN モジュールを OFF/ON する事によってのみクリアできます。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 30-3: CiINT: CAN 割り込みレジスタ ( 続き )

- bit 14 **WAKIF:** CAN バス アクティビティ復帰割り込みフラグビット  
1 = バス アクティビティ復帰割り込みが発生した  
0 = バス アクティビティ復帰割り込みは発生していない
- bit 13 **CERRIF:** CAN バスエラー割り込みフラグビット  
1 = CAN バスエラーが発生した  
0 = CAN バスエラーは発生していない
- bit 12 **SERRIF:** システムエラー割り込みフラグビット  
1 = システムエラーが発生した ( 不正なアドレスがシステムバスに出力された等 )  
0 = システムエラーは発生していない
- bit 11 **RBOVIF:** 受信バッファ オーバーフロー割り込みフラグビット  
1 = 受信バッファ オーバーフローが発生した  
0 = 受信バッファ オーバーフローは発生していない
- bit 10-4 **未実装:** 「0」として読み出し
- bit 3 **MODIF:** CAN モード変化割り込みフラグビット  
1 = CAN モジュールのモードが変化した (REQOP に従って OPMOD<2:0> が変化した )  
0 = CAN モジュールのモードは変化していない
- bit 2 **CTMRIF:** CAN タイマ オーバーフロー割り込みフラグビット  
1 = CAN タイマ (CANTMR) がオーバーフローした  
0 = CAN タイマ (CANTMR) はオーバーフローしていない
- bit 1 **RBIF:** 受信バッファ割り込みフラグビット  
1 = 受信バッファ割り込みが保留中  
0 = 受信バッファ割り込みは保留中ではない
- bit 0 **TBIF:** 送信バッファ割り込みフラグビット  
1 = 送信バッファ割り込みが保留中  
0 = 送信バッファ割り込みは保留中ではない

**Note 1:** このビットは、ON ビット (CiCON<15>) をクリア / セットして CAN モジュールを OFF/ON する事によってのみクリアできます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-4: CiVEC: CAN 割り込みコードレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
	—	—	—	FILHIT<4:0>				
7:0	U-0	R-1	R-0	R-0	R-0	R-0	R-0	R-0
	—	ICODE<6:0> <sup>(1)</sup>						

### 凡例:

R=読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-13 **未実装**: 「0」として読み出し

bit 12-8 **FILHIT<4:0>**: フィルタヒット番号ビット

11111 = フィルタ 31

11110 = フィルタ 30

•

•

•

00001 = フィルタ 1

00000 = フィルタ 0

bit 7 **未実装**: 「0」として読み出し

bit 6-0 **ICODE<6:0>**: 割り込みフラグコードビット<sup>(1)</sup>

1001000-1111111 = 予約済み

1001000 = 無効メッセージの受信 (IVRIF)

1000111 = CAN モジュールのモード変化 (MODIF)

1000110 = CAN タイムスタンプ タイマ (CTMRIF)

1000101 = バス帯域幅エラー (SERRIF)

1000100 = アドレスエラー割り込み (SERRIF)

1000011 = 受信 FIFO オーバーフロー割り込み (RBOVIF)

1000010 = 復帰割り込み (WAKIF)

1000001 = エラー割り込み (CERRIF)

1000000 = 割り込みなし

0100000?0111111 = 予約済み

0011111 = FIFO31 割り込み (CiFSTAT<31> がセット)

0011110 = FIFO30 割り込み (CiFSTAT<30> がセット)

•

•

•

0000001 = FIFO1 割り込み (CiFSTAT<1> がセット)

0000000 = FIFO0 割り込み (CiFSTAT<0> がセット)

**Note 1:** これらのビットは有効な割り込みに対してのみ更新されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-5: CiTREC: CAN 送受信エラーカウント レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
	—	—	TXBO	TXBP	RXBP	TXWARN	RXWARN	EWARN
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	TERRCNT<7:0>							
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RERRCNT<7:0>							

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-22 **未実装**: 「0」として読み出し

bit 21 **TXBO**: トランスミッタが「バス OFF」エラー状態 (TERRCNT ≥ 256)

bit 20 **TXBP**: トランスミッタが「バスパッシブ」エラー状態 (TERRCNT ≥ 128)

bit 19 **RXBP**: レシーバが「バスパッシブ」エラー状態 (RERRCNT ≥ 128)

bit 18 **TXWARN**: トランスミッタが「警告」エラー状態 (128 > TERRCNT ≥ 96)

bit 17 **RXWARN**: レシーバが「警告」エラー状態 (128 > RERRCNT ≥ 96)

bit 16 **EWARN**: トランスミッタまたはレシーバが「警告」エラー状態

bit 15-8 **TERRCNT<7:0>**: 送信エラーカウンタ

bit 7-0 **RERRCNT<7:0>**: 受信エラーカウンタ

## レジスタ 30-6: CifSTAT: CAN FIFO ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	FIFOIP31	FIFOIP30	FIFOIP29	FIFOIP28	FIFOIP27	FIFOIP26	FIFOIP25	FIFOIP24
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	FIFOIP23	FIFOIP22	FIFOIP21	FIFOIP20	FIFOIP19	FIFOIP18	FIFOIP17	FIFOIP16
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	FIFOIP15	FIFOIP14	FIFOIP13	FIFOIP12	FIFOIP11	FIFOIP10	FIFOIP9	FIFOIP8
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	FIFOIP7	FIFOIP6	FIFOIP5	FIFOIP4	FIFOIP3	FIFOIP2	FIFOIP1	FIFOIP0

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-0 **FIFOIP<31:0>**: FIFO<sub>n</sub> 割り込み保留ビット

1 = 1 つまたは複数の有効な FIFO の割り込みが保留中

0 = FIFO 割り込みは保留中ではない

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-7: CiRXOVF: CAN 受信 FIFO オーバーフロー ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0

### 凡例:

R = 読み出し可能ビット    W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値    1 = ビットはセット    0 = ビットはクリア    x = ビットは未知

bit 31-0 **RXOVF<31:0>**: FIFO<sub>n</sub> 受信オーバーフロー割り込み保留中ビット

1 = FIFO がオーバーフローした  
 0 = FIFO はオーバーフローしていない

## レジスタ 30-8: CiTMR: CAN タイマレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CANTS<15:8>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CANTS<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CANTSPRE<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CANTSPRE<7:0>							

### 凡例:

R = 読み出し可能ビット    W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値    1 = ビットはセット    0 = ビットはクリア    x = ビットは未知

bit 31-0 **CANTS<15:0>**: CAN タイムスタンプ タイマビット

これは、CANCAP ビット (CiCON<20>) がセットされている時に CANTSPRE システムクロック周期でインクリメントするフリーランニング タイマです。

bit 15-0 **CANTSPRE<15:0>**: CAN タイムスタンプ タイマ プリスケラ ビット

1111 1111 1111 1111 = CAN タイムスタンプ タイマ (CANTS) は 65,535 システムクロックごとにインクリメントする

•  
•  
•

0000 0000 0000 0000 = CAN タイムスタンプ タイマ (CANTS) は 1 システムクロックごとにインクリメントする

**Note 1:** CANCAP = 0 の場合、CiTMR は停止します。

**2:** CiTMR プリスケラのカウンタは、CiTMR レジスタへの書き込み時にリセットされます (CANTSPRE は影響を受けません)。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-9: CiRXMn: CAN アクセプタンス フィルタマスク n レジスタ (n = 0、1、2、3)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SID<10:3>							
23:16	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0
	SID<2:0>			—	MIDE	—	EID<17:16>	
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EID<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EID<7:0>							

### 凡例:

R = 読み出し可能ビット    W = 書き込み可能ビット    U = 未実装、「0」として読み出し

-n = POR 時の値    1 = ビットはセット    0 = ビットはクリア    x = ビットは未知

bit 31-21 **SID<10:0>**: 標準 ID ビット

- 1 = フィルタ比較にこの SIDx ビットを含める
- 0 = フィルタ比較ではこの SIDx ビットを無視する

bit 20 **未実装**: 「0」として読み出し

bit 19 **MIDE**: ID 受信モードビット

- 1 = フィルタの EXID ビットに対応するメッセージタイプ (標準または拡張アドレス) のみ一致する
- 0 = 標準アドレスメッセージまたは拡張アドレスメッセージのどちらでも一致する (フィルタ SID = メッセージ SID、または、フィルタ SID/EID = メッセージ SID/EID の場合に一致)

bit 18 **未実装**: 「0」として読み出し

bit 17-0 **EID<17:0>**: 拡張 ID ビット

- 1 = フィルタ比較にこの EIDx ビットを含める
- 0 = フィルタ比較ではこの EIDx ビットを無視する

**Note:** このレジスタは、CAN モジュールがコンフィグレーションモード中 (OPMOD<2:0> (CiCON<23:21>) = 100) の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-10: CiFLTCON0: CAN フィルタ制御レジスタ 0

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN3	MSEL3<1:0>		FSEL3<4:0>				
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN2	MSEL2<1:0>		FSEL2<4:0>				
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN1	MSEL1<1:0>		FSEL1<4:0>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN0	MSEL0<1:0>		FSEL0<4:0>				

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
 -n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31 **FLTEN3:** フィルタ 3 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 30-29 **MSEL3<1:0>:** フィルタ 3 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 28-24 **FSEL3<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
•  
•

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

bit 23 **FLTEN2:** フィルタ 2 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 22-21 **MSEL2<1:0>:** フィルタ 2 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 20-16 **FSEL2<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
•  
•

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-10: CiFLTCON0: CAN フィルタ制御レジスタ 0 ( 続き )

- bit 15      **FLTEN1:** フィルタ 1 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 14-13   **MSEL1<1:0>:** フィルタ 1 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 12-8    **FSEL1<4:0>:** FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する
- bit 7        **FLTEN0:** フィルタ 0 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 6-5     **MSEL0<1:0>:** フィルタ 0 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 4-0     **FSEL0<4:0>:** FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-11: CiFLTCON1: CAN フィルタ制御レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN7	MSEL7<1:0>		FSEL7<4:0>				
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN6	MSEL6<1:0>		FSEL6<4:0>				
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN5	MSEL5<1:0>		FSEL5<4:0>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN4	MSEL4<1:0>		FSEL4<4:0>				

### 凡例:

R = 読み出し可能ビット    W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値            1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

bit 31    **FLTEN7**: フィルタ 7 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 30-29 **MSEL7<1:0>**: フィルタ 7 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 28-24 **FSEL7<4:0>**: FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
 •  
 •

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

bit 23    **FLTEN6**: フィルタ 6 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 22-21 **MSEL6<1:0>**: フィルタ 6 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 20-16 **FSEL6<4:0>**: FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
 •  
 •

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-11: CiFLTCON1: CAN フィルタ制御レジスタ 1 ( 続き )

- bit 15 **FLTEN5:** フィルタ 17 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 14-13 **MSEL5<1:0>:** フィルタ 5 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 12-8 **FSEL5<4:0>:** FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する
- bit 7 **FLTEN4:** フィルタ 4 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 6-5 **MSEL4<1:0>:** フィルタ 4 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 4-0 **FSEL4<4:0>:** FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-12: CiFLTCON2: CAN フィルタ制御レジスタ 2

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN11	MSEL11<1:0>	FSEL11<4:0>					
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN10	MSEL10<1:0>	FSEL10<4:0>					
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN9	MSEL9<1:0>	FSEL9<4:0>					
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN8	MSEL8<1:0>	FSEL8<4:0>					

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
 -n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31 **FLTEN11:** フィルタ 11 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 30-29 **MSEL11<1:0>:** フィルタ 11 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 28-24 **FSEL11<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
•  
•

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

bit 23 **FLTEN10:** フィルタ 10 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 22-21 **MSEL10<1:0>:** フィルタ 10 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 20-16 **FSEL10<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
•  
•

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-12: CiFLTCON2: CAN フィルタ制御レジスタ 2 ( 続き )

- bit 15     **FLTEN9:** フィルタ 9 イネーブルビット  
          1 = フィルタを有効にする  
          0 = フィルタを無効にする
- bit 14-13   **MSEL9<1:0>:** フィルタ 9 マスク選択ビット  
          11 = アクセプタンス マスク 3 を選択する  
          10 = アクセプタンス マスク 2 を選択する  
          01 = アクセプタンス マスク 1 を選択する  
          00 = アクセプタンス マスク 0 を選択する
- bit 12-8   **FSEL9<4:0>:** FIFO 選択ビット  
          11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
          11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
          .  
          .  
          .  
          00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
          00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する
- bit 7       **FLTEN8:** フィルタ 8 イネーブルビット  
          1 = フィルタを有効にする  
          0 = フィルタを無効にする
- bit 6-5     **MSEL8<1:0>:** フィルタ 8 マスク選択ビット  
          11 = アクセプタンス マスク 3 を選択する  
          10 = アクセプタンス マスク 2 を選択する  
          01 = アクセプタンス マスク 1 を選択する  
          00 = アクセプタンス マスク 0 を選択する
- bit 4-0     **FSEL8<4:0>:** FIFO 選択ビット  
          11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
          11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
          .  
          .  
          .  
          00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
          00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-13: CiFLTCON3:CAN フィルタ制御レジスタ 3

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN15	MSEL15<1:0>		FSEL15<4:0>				
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN14	MSEL14<1:0>		FSEL14<4:0>				
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN13	MSEL13<1:0>		FSEL13<4:0>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN12	MSEL12<1:0>		FSEL12<4:0>				

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
 -n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31 **FLTEN15:** フィルタ 15 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 30-29 **MSEL15<1:0>:** フィルタ 15 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 28-24 **FSEL15<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
•  
•

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

bit 23 **FLTEN14:** フィルタ 14 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 22-21 **MSEL14<1:0>:** フィルタ 14 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 20-16 **FSEL14<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
•  
•

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-13: CiFLTCON3:CAN フィルタ制御レジスタ 3 ( 続き )

- bit 15     **FLTEN13:** フィルタ 13 イネーブルビット  
          1 = フィルタを有効にする  
          0 = フィルタを無効にする
- bit 14-13   **MSEL13<1:0>:** フィルタ 13 マスク選択ビット  
          11 = アクセプタンス マスク 3 を選択する  
          10 = アクセプタンス マスク 2 を選択する  
          01 = アクセプタンス マスク 1 を選択する  
          00 = アクセプタンス マスク 0 を選択する
- bit 12-8   **FSEL13<4:0>:** FIFO 選択ビット  
          11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
          11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
          .  
          .  
          .  
          00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
          00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する
- bit 7       **FLTEN12:** フィルタ 12 イネーブルビット  
          1 = フィルタを有効にする  
          0 = フィルタを無効にする
- bit 6-5     **MSEL12<1:0>:** フィルタ 12 マスク選択ビット  
          11 = アクセプタンス マスク 3 を選択する  
          10 = アクセプタンス マスク 2 を選択する  
          01 = アクセプタンス マスク 1 を選択する  
          00 = アクセプタンス マスク 0 を選択する
- bit 4-0     **FSEL12<4:0>:** FIFO 選択ビット  
          11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
          11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
          .  
          .  
          .  
          00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
          00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-14: CiFLTCON4: CAN フィルタ制御レジスタ 4

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN19	MSEL19<1:0>		FSEL19<4:0>				
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN18	MSEL18<1:0>		FSEL18<4:0>				
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN17	MSEL17<1:0>		FSEL17<4:0>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN16	MSEL16<1:0>		FSEL16<4:0>				

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31 **FLTEN19:** フィルタ 19 イネーブルビット

1 = フィルタを有効にする  
0 = フィルタを無効にする

bit 30-29 **MSEL19<1:0>:** フィルタ 19 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する

bit 28-24 **FSEL19<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
•  
•

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

bit 23 **FLTEN18:** フィルタ 18 イネーブルビット

1 = フィルタを有効にする  
0 = フィルタを無効にする

bit 22-21 **MSEL18<1:0>:** フィルタ 18 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する

bit 20-16 **FSEL18<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する

•  
•  
•

00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-14: CiFLTCON4: CAN フィルタ制御レジスタ 4 (続き)

- bit 15     **FLTEN17**: フィルタ 13 イネーブルビット  
          1 = フィルタを有効にする  
          0 = フィルタを無効にする
- bit 14-13   **MSEL17<1:0>**: フィルタ 17 マスク選択ビット  
          11 = アクセプタンス マスク 3 を選択する  
          10 = アクセプタンス マスク 2 を選択する  
          01 = アクセプタンス マスク 1 を選択する  
          00 = アクセプタンス マスク 0 を選択する
- bit 12-8   **FSEL17<4:0>**: FIFO 選択ビット  
          11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
          11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
          .  
          .  
          .  
          00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
          00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する
- bit 7       **FLTEN16**: フィルタ 16 イネーブルビット  
          1 = フィルタを有効にする  
          0 = フィルタを無効にする
- bit 6-5     **MSEL16<1:0>**: フィルタ 16 マスク選択ビット  
          11 = アクセプタンス マスク 3 を選択する  
          10 = アクセプタンス マスク 2 を選択する  
          01 = アクセプタンス マスク 1 を選択する  
          00 = アクセプタンス マスク 0 を選択する
- bit 4-0     **FSEL16<4:0>**: FIFO 選択ビット  
          11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
          11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
          .  
          .  
          .  
          00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
          00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-15: CiFLTCON5:CAN フィルタ制御レジスタ 5

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN23	MSEL23<1:0>		FSEL23<4:0>				
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN22	MSEL22<1:0>		FSEL22<4:0>				
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN21	MSEL21<1:0>		FSEL21<4:0>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN20	MSEL20<1:0>		FSEL20<4:0>				

### 凡例:

R = 読み出し可能ビット    W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値    1 = ビットはセット    0 = ビットはクリア    x = ビットは未知

bit 31    **FLTEN23:** フィルタ 23 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 30-29    **MSEL23<1:0>:** フィルタ 23 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 28-24    **FSEL23<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
 .  
 .  
 00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

bit 23    **FLTEN22:** フィルタ 22 イネーブルビット

1 = フィルタを有効にする  
 0 = フィルタを無効にする

bit 22-21    **MSEL22<1:0>:** フィルタ 22 マスク選択ビット

11 = アクセプタンス マスク 3 を選択する  
 10 = アクセプタンス マスク 2 を選択する  
 01 = アクセプタンス マスク 1 を選択する  
 00 = アクセプタンス マスク 0 を選択する

bit 20-16    **FSEL22<4:0>:** FIFO 選択ビット

11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
 .  
 .  
 00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-15: CiFLTCO5:CAN フィルタ制御レジスタ 5 ( 続き )

- bit 15 **FLTEN21**: フィルタ 21 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 14-13 **MSEL21<1:0>**: フィルタ 21 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 12-8 **FSEL21<4:0>**: FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する
- bit 7 **FLTEN20**: フィルタ 20 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 6-5 **MSEL20<1:0>**: フィルタ 20 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 4-0 **FSEL20<4:0>**: FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-16: CiFLTCON6:CAN フィルタ制御レジスタ 6

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN27	MSEL27<1:0>		FSEL27<4:0>				
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN26	MSEL26<1:0>		FSEL26<4:0>				
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN25	MSEL25<1:0>		FSEL25<4:0>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN24	MSEL24<1:0>		FSEL24<4:0>				

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31 **FLTEN27:** フィルタ 27 イネーブルビット

- 1 = フィルタを有効にする
- 0 = フィルタを無効にする

bit 30-29 **MSEL27<1:0>:** フィルタ 27 マスク選択ビット

- 11 = アクセプタンス マスク 3 を選択する
- 10 = アクセプタンス マスク 2 を選択する
- 01 = アクセプタンス マスク 1 を選択する
- 00 = アクセプタンス マスク 0 を選択する

bit 28-24 **FSEL27<4:0>:** FIFO 選択ビット

- 11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する
- 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する
- 
- 
- 
- 00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する
- 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

bit 23 **FLTEN26:** フィルタ 26 イネーブルビット

- 1 = フィルタを有効にする
- 0 = フィルタを無効にする

bit 22-21 **MSEL26<1:0>:** フィルタ 26 マスク選択ビット

- 11 = アクセプタンス マスク 3 を選択する
- 10 = アクセプタンス マスク 2 を選択する
- 01 = アクセプタンス マスク 1 を選択する
- 00 = アクセプタンス マスク 0 を選択する

bit 20-16 **FSEL26<4:0>:** FIFO 選択ビット

- 11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する
- 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する
- 
- 
- 
- 00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する
- 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-16: CiFLTCON6:CAN フィルタ制御レジスタ 6 ( 続き )

- bit 15 **FLTEN25:** フィルタ 25 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 14-13 **MSEL25<1:0>:** フィルタ 25 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 12-8 **FSEL25<4:0>:** FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する
- bit 7 **FLTEN24:** フィルタ 24 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 6-5 **MSEL24<1:0>:** フィルタ 24 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 4-0 **FSEL24<4:0>:** FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-17: CiFLTCON7:CAN フィルタ制御レジスタ 7

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN31	MSEL31<1:0>		FSEL31<4:0>				
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN30	MSEL30<1:0>		FSEL30<4:0>				
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN29	MSEL29<1:0>		FSEL29<4:0>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FLTEN28	MSEL28<1:0>		FSEL28<4:0>				

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31 **FLTEN31:** フィルタ 31 イネーブルビット

- 1 = フィルタを有効にする
- 0 = フィルタを無効にする

bit 30-29 **MSEL31<1:0>:** フィルタ 31 マスク選択ビット

- 11 = アクセプタンス マスク 3 を選択する
- 10 = アクセプタンス マスク 2 を選択する
- 01 = アクセプタンス マスク 1 を選択する
- 00 = アクセプタンス マスク 0 を選択する

bit 28-24 **FSEL31<4:0>:** FIFO 選択ビット

- 11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する
- 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する
- 
- 
- 
- 00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する
- 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

bit 23 **FLTEN30:** フィルタ 30 イネーブルビット

- 1 = フィルタを有効にする
- 0 = フィルタを無効にする

bit 22-21 **MSEL30<1:0>:** フィルタ 30 マスク選択ビット

- 11 = アクセプタンス マスク 3 を選択する
- 10 = アクセプタンス マスク 2 を選択する
- 01 = アクセプタンス マスク 1 を選択する
- 00 = アクセプタンス マスク 0 を選択する

bit 20-16 **FSEL30<4:0>:** FIFO 選択ビット

- 11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する
- 11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する
- 
- 
- 
- 00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する
- 00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-17: CiFLTCON7:CAN フィルタ制御レジスタ 7 ( 続き )

- bit 15 **FLTEN29**: フィルタ 29 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 14-13 **MSEL29<1:0>**: フィルタ 29 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 12-8 **FSEL29<4:0>**: FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する
- bit 7 **FLTEN28**: フィルタ 28 イネーブルビット  
1 = フィルタを有効にする  
0 = フィルタを無効にする
- bit 6-5 **MSEL28<1:0>**: フィルタ 28 マスク選択ビット  
11 = アクセプタンス マスク 3 を選択する  
10 = アクセプタンス マスク 2 を選択する  
01 = アクセプタンス マスク 1 を選択する  
00 = アクセプタンス マスク 0 を選択する
- bit 4-0 **FSEL28<4:0>**: FIFO 選択ビット  
11111 = フィルタに一致するメッセージを FIFO バッファ 31 に格納する  
11110 = フィルタに一致するメッセージを FIFO バッファ 30 に格納する  
.  
.  
.  
00001 = フィルタに一致するメッセージを FIFO バッファ 1 に格納する  
00000 = フィルタに一致するメッセージを FIFO バッファ 0 に格納する

**Note:** このレジスタ内のビットは、対応するフィルタ イネーブルビット (FLTENn) が「0」の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 30-18: CiRxFn: CAN アクセプタンス フィルタ n レジスタ 7(n = 0 ~ 31)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	SID<10:3>							
23:16	R/W-x	R/W-x	R/W-x	U-0	R/W-0	U-0	R/W-x	R/W-x
	SID<2:0>		—	EXID	—	EID<17:16>		
15:8	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	EID<15:8>							
7:0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	EID<7:0>							

**凡例:**

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-21 **SID<10:0>**: 標準 ID ビット

- 1 = フィルタに一致するにはメッセージ アドレスビット SIDx が「1」である事が必要
- 0 = フィルタに一致するにはメッセージ アドレスビット SIDx が「0」である事が必要

bit 20 **未実装**: 「0」として読み出し

bit 19 **EXID**: 拡張 ID イネーブルビット

- 1 = 拡張 ID アドレスを持つメッセージのみ一致する
- 0 = 標準 ID アドレスを持つメッセージのみ一致する

bit 18 **未実装**: 「0」として読み出し

bit 17-0 **EID<17:0>**: 拡張 ID ビット

- 1 = フィルタに一致するにはメッセージ アドレスビット EIDx が「1」である事が必要
- 0 = フィルタに一致するにはメッセージ アドレスビット EIDx が「0」である事が必要

**Note:** このレジスタは、フィルタが無効 (FLTENn = 0) の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-19: CiFIFOBA: CAN メッセージ バッファ ベースアドレス レジスタ

ビットレ ンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CiFIFOBA<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CiFIFOBA<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CiFIFOBA<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0 <sup>(1)</sup>	R-0 <sup>(1)</sup>
	CiFIFOBA<7:0>							

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

### bit 31-0 CiFIFOBA<31:0>: CAN FIFO ベースアドレス ビット

これらのビットは、全てのメッセージ バッファのベースアドレスを定義します。個々のメッセージ バッファは、前のメッセージ バッファのサイズに基づいて配置されます。このアドレスは物理アドレスです。ビット <1:0> は読み出し専用であり、値は「0」です。これにより、メッセージはデバイス RAM 内で 32 ビットワード境界に配置されます。

**Note 1:** これらのビットは未実装であり、読み値は常に「0」です。これにより、メッセージはワード境界に配置されます。

**Note:** このレジスタは、CAN モジュールがコンフィグレーションモード中 (OPMOD<2:0> (CiCON<23:21>) = 100) の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-20: CiFIFOCONn: CAN FIFO 制御レジスタ (n = 0 ~ 31)

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	FSIZE<4:0> <sup>(1)</sup>				
15:8	U-0	S/HC-0	S/HC-0	R/W-0	U-0	U-0	U-0	U-0
	—	FRESET	UINC	DONLY <sup>(1)</sup>	—	—	—	—
7:0	R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXEN	TXABAT <sup>(2)</sup>	TXLARB <sup>(3)</sup>	TXERR <sup>(3)</sup>	TXREQ	RTREN	TXPR<1:0>	

### 凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-21 **未実装:** 「0」として読み出し

bit 20-16 **FSIZE<4:0>:** FIFO サイズビット<sup>(1)</sup>

11111 = FIFO の深さは 32 メッセージ

•

•

•

00010 = FIFO の深さは 3 メッセージ

00001 = FIFO の深さは 2 メッセージ

00000 = FIFO の深さは 1 メッセージ

bit 15 **未実装:** 「0」として読み出し

bit 14 **FRESET:** FIFO リセットビット

1 = FIFO をリセットする

このビットは、FIFO のリセット後にハードウェアによってクリアされます。このビットをセットした後は、このビットをポーリングしてクリアされている事を確認してから他の動作を実行する必要があります。

0 = 何もしない

bit 13 **UINC:** インクリメント ヘッド/テールビット

TXEN = 1: (FIFO を送信 FIFO として設定した場合)

このビットをセットすると FIFO ヘッドが 1 メッセージ分インクリメントします。

TXEN = 0: (FIFO を受信 FIFO として設定した場合)

このビットをセットすると FIFO テールが 1 メッセージ分インクリメントします。

bit 12 **DONLY:** メッセージデータのみ格納ビット<sup>(1)</sup>

TXEN = 1: (FIFO を送信 FIFO として設定した場合)

このビットは未使用 (効果なし) です。

TXEN = 0: (FIFO を受信 FIFO として設定した場合)

1 = FIFO にデータバイトのみを格納する

0 = メッセージ全体 (ID を含む) を格納する

bit 11-8 **未実装:** 「0」として読み出し

bit 7 **TXEN:** TX/RX バッファ選択ビット

1 = FIFO を送信 FIFO として使う

0 = FIFO を受信 FIFO として使う

**Note 1:** これらのビットは、CAN モジュールがコンフィグレーションモード中 (OPMOD<2:0> ビット (CiCON<23:21>) = 100) の場合にのみ変更できます。

**2:** このビットは、メッセージの完了時 (または中止時) と FIFO のリセット時に更新されます。

**3:** このビットは、このレジスタの読み出し時と FIFO のリセット時にリセットされます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-20: CiFIFOCONn: CAN FIFO 制御レジスタ (n = 0 ~ 31) (続き)

- bit 6 **TXABAT:** メッセージ中止ビット<sup>(2)</sup>  
1 = メッセージ送信は中止された  
0 = メッセージ送信は正常に完了した
- bit 5 **TXLARB:** メッセージ アービトレーション ロストビット<sup>(3)</sup>  
1 = メッセージの送信中にアービトレーション ロストが発生した  
0 = メッセージの送信中にアービトレーション ロストは発生しなかった
- bit 4 **TXERR:** 送信中エラー検出ビット<sup>(3)</sup>  
1 = メッセージの送信中にバスエラーが発生した  
0 = メッセージの送信中にバスエラーは発生しなかった
- bit 3 **TXREQ:** メッセージ送信要求  
TXEN = 1:(FIFO を送信 FIFO として設定した場合)  
このビットを「1」にセットする事でメッセージの送信を要求します。  
FIFO 内の全てのメッセージが正常に送信されると、このビットは自動的にクリアされます。  
このビットが「1」の時に「0」にクリアする事でメッセージの中止を要求します。  
TXEN = 0:(FIFO を受信 FIFO として設定した場合)  
このビットは無効です。
- bit 2 **RTREN:** 自動 RTR イネーブルビット  
1 = リモート送信を受信した時に TXREQ をセットする  
0 = リモート送信を受信しても TXREQ には影響しない
- bit 1-0 **TXPR<1:0>:** メッセージ送信優先度ビット  
11 = 最高のメッセージ優先度  
10 = 第2位のメッセージ優先度  
01 = 第3位のメッセージ優先度  
00 = 最低のメッセージ優先度

- Note 1:** これらのビットは、CAN モジュールがコンフィグレーションモード中 (OPMOD<2:0> ビット (CiCON<23:21> = 100) の場合にのみ変更できます。
- 2:** このビットは、メッセージの完了時 (または中止時) と FIFO のリセット時に更新されます。
- 3:** このビットは、このレジスタの読み出し時と FIFO のリセット時にリセットされます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-21: CiFIFOINTn: CAN FIFO 割り込みレジスタ (n = 0 ~ 31)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	TXNFULLIE	TXHALFIE	TXEMPTYIE
23:16	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	RXOVFLIE	RXFULLIE	RXHALFIE	RXEMPTYIE
15:8	U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
	—	—	—	—	—	TXNFULLIF <sup>(1)</sup>	TXHALFIF	TXEMPTYIF <sup>(1)</sup>
7:0	U-0	U-0	U-0	U-0	R/W-0	R-0	R-0	R-0
	—	—	—	—	RXOVFLIF	RXFULLIF <sup>(1)</sup>	RXHALFIF <sup>(1)</sup>	RXEMPTYIF <sup>(1)</sup>

### 凡例:

R=読み出し可能ビット W=書き込み可能ビット U = 未実装、「0」として読み出し  
 -n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-27 **未実装**: 「0」として読み出し

bit 26 **TXNFULLIE**: 送信 FIFO 非フル割り込みイネーブルビット

1 = FIFO 非フル時の割り込みを有効にする  
 0 = FIFO 非フル時の割り込みを無効にする

bit 25 **TXHALFIE**: 送信 FIFO ハーフフル割り込みイネーブルビット

1 = FIFO ハーフフル時の割り込みを有効にする  
 0 = FIFO ハーフフル時の割り込みを無効にする

bit 24 **TXEMPTYIE**: 送信 FIFO エンプティ割り込みイネーブルビット

1 = FIFO エンプティ時の割り込みを有効にする  
 0 = FIFO エンプティ時の割り込みを無効にする

bit 23-20 **未実装**: 「0」として読み出し

bit 19 **RXOVFLIE**: オーバーフロー割り込みイネーブルビット

1 = オーバーフロー発生時の割り込みを有効にする  
 0 = オーバーフロー発生時の割り込みを無効にする

bit 18 **RXFULLIE**: 受信 FIFO フル割り込みイネーブルビット

1 = FIFO フル時の割り込みを有効にする  
 0 = FIFO フル時の割り込みを無効にする

bit 17 **RXHALFIE**: 受信 FIFO ハーフフル割り込みイネーブルビット

1 = FIFO ハーフフル時の割り込みを有効にする  
 0 = FIFO ハーフフル時の割り込みを無効にする

bit 16 **RXEMPTYIE**: 受信 FIFO 非エンプティ割り込みイネーブルビット

1 = FIFO 非エンプティ時の割り込みを有効にする  
 0 = FIFO 非エンプティ時の割り込みを無効にする

bit 15-11 **未実装**: 「0」として読み出し

bit 10 **TXNFULLIF**: 送信 FIFO 非フル割り込みフラグビット<sup>(1)</sup>

TXEN = 1:(FIFO を送信バッファとして設定した場合)

1 = FIFO は非フル  
 0 = FIFO はフル

TXEN = 0:(FIFO を受信バッファとして設定した場合)

未使用、「0」として読み出し

**Note 1:** このビットは FIFO のステータスを示す読み出し専用ビットです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-21: CiFIFOINTn: CAN FIFO 割り込みレジスタ (n = 0 ~ 31) (続き)

- bit 9 **TXHALFIF**: 送信 FIFO ハーフエンプティ割り込みフラグビット<sup>(1)</sup>  
 $\text{TXEN} = 1$ : (FIFO を送信バッファとして設定した場合)  
1 = FIFO  $\leq$  ハーフフル  
0 = FIFO  $>$  ハーフフル  
 $\text{TXEN} = 0$ : (FIFO を受信バッファとして設定した場合)  
未使用、「0」として読み出し
- bit 8 **TXEMPTYIF**: 送信 FIFO エンプティ割り込みフラグビット<sup>(1)</sup>  
 $\text{TXEN} = 1$ : (FIFO を送信バッファとして設定した場合)  
1 = FIFO はエンプティ  
0 = FIFO は非エンプティ (少なくとも1つのメッセージが送信待ち)  
 $\text{TXEN} = 0$ : (FIFO を受信バッファとして設定した場合)  
未使用、「0」として読み出し
- bit 7-4 **未実装**: 「0」として読み出し
- bit 3 **RXOVFLIF**: 受信 FIFO オーバーフロー割り込みフラグビット  
 $\text{TXEN} = 1$ : (FIFO を送信バッファとして設定した場合)  
未使用、「0」として読み出し  
 $\text{TXEN} = 0$ : (FIFO を受信バッファとして設定した場合)  
1 = オーバーフロー イベントが発生した  
0 = オーバーフロー イベントは発生していない
- bit 2 **RXFULLIF**: 受信 FIFO フル割り込みフラグビット<sup>(1)</sup>  
 $\text{TXEN} = 1$ : (FIFO を送信バッファとして設定した場合)  
未使用、「0」として読み出し  
 $\text{TXEN} = 0$ : (FIFO を受信バッファとして設定した場合)  
1 = FIFO はフル  
0 = FIFO は非フル
- bit 1 **RXHALFIF**: 受信 FIFO ハーフフル割り込みフラグビット<sup>(1)</sup>  
 $\text{TXEN} = 1$ : (FIFO を送信バッファとして設定した場合)  
未使用、「0」として読み出し  
 $\text{TXEN} = 0$ : (FIFO を受信バッファとして設定した場合)  
1 = FIFO  $\geq$  ハーフフル  
0 = FIFO  $<$  ハーフフル
- bit 0 **RXNEMPTYIF**: 受信バッファ非エンプティ割り込みフラグビット<sup>(1)</sup>  
 $\text{TXEN} = 1$ : (FIFO を送信バッファとして設定した場合)  
未使用、「0」として読み出し  
 $\text{TXEN} = 0$ : (FIFO を受信バッファとして設定した場合)  
1 = FIFO は非エンプティ (少なくとも1つのメッセージが存在する)  
0 = FIFO はエンプティ

**Note 1:** このビットは FIFO のステータスを示す読み出し専用ビットです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 30-22: CiFIFOUAn: CAN FIFO ユーザアドレス レジスタ (n = 0 ~ 31)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
	CiFIFOUAn<31:24>							
23:16	R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
	CiFIFOUAn<23:16>							
15:8	R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
	CiFIFOUAn<15:8>							
7:0	R-x	R-x	R-x	R-x	R-x	R-x	R-0 <sup>(1)</sup>	R-0 <sup>(1)</sup>
	CiFIFOUAn<7:0>							

### 凡例:

R= 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-0 **CiFIFOUAn<31:0>**: CAN FIFO ユーザアドレス ビット

TXEN = 1: (FIFO を送信バッファとして設定した場合)

このレジスタを読み出すと、次のメッセージの書き込み先アドレス (FIFO ヘッド) が返されます。

TXEN = 0: (FIFO を受信バッファとして設定した場合)

このレジスタを読み出すと、次のメッセージの読み出し元アドレス (FIFO テール) が返されます。

**Note 1:** このビットの読み値は常に「0」です。これにより、メッセージはバイト境界に配置されます。

**Note:** このレジスタは、コンフィグレーション モード中に正しく読み出せない場合があります。このレジスタには、コンフィグレーション モード中ではない時にアクセスする必要があります。

## レジスタ 30-23: CiFIFOCIn: CAN モジュール メッセージ インデックス レジスタ (n = 0 ~ 31)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
	—	—	—	CiFIFOCIn<4:0>				

### 凡例:

R= 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-5 **未実装**: 「0」として読み出し

bit 4-0 **CiFIFOCIn<4:0>**: CAN 側 FIFO メッセージ インデックス ビット

TXEN = 1: (FIFO を送信バッファとして設定した場合)

このレジスタを読み出すと、FIFO が次に送信を試みるメッセージに対するインデックスが返されます。

TXEN = 0: (FIFO を受信バッファとして設定した場合)

このレジスタを読み出すと、FIFO が次のメッセージを保存するために使うインデックスが返されます。



# PIC32MZ グラフィック (DA) ファミリ

## 31.0 Ethernet コントローラ

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 35. Ethernet Controller』(DS60001155)を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

Ethernet コントローラはデバイス外部の物理層 (PHY) に接続するバスマスタ モジュールであり、システムに Ethernet ノードを実装するために使います。

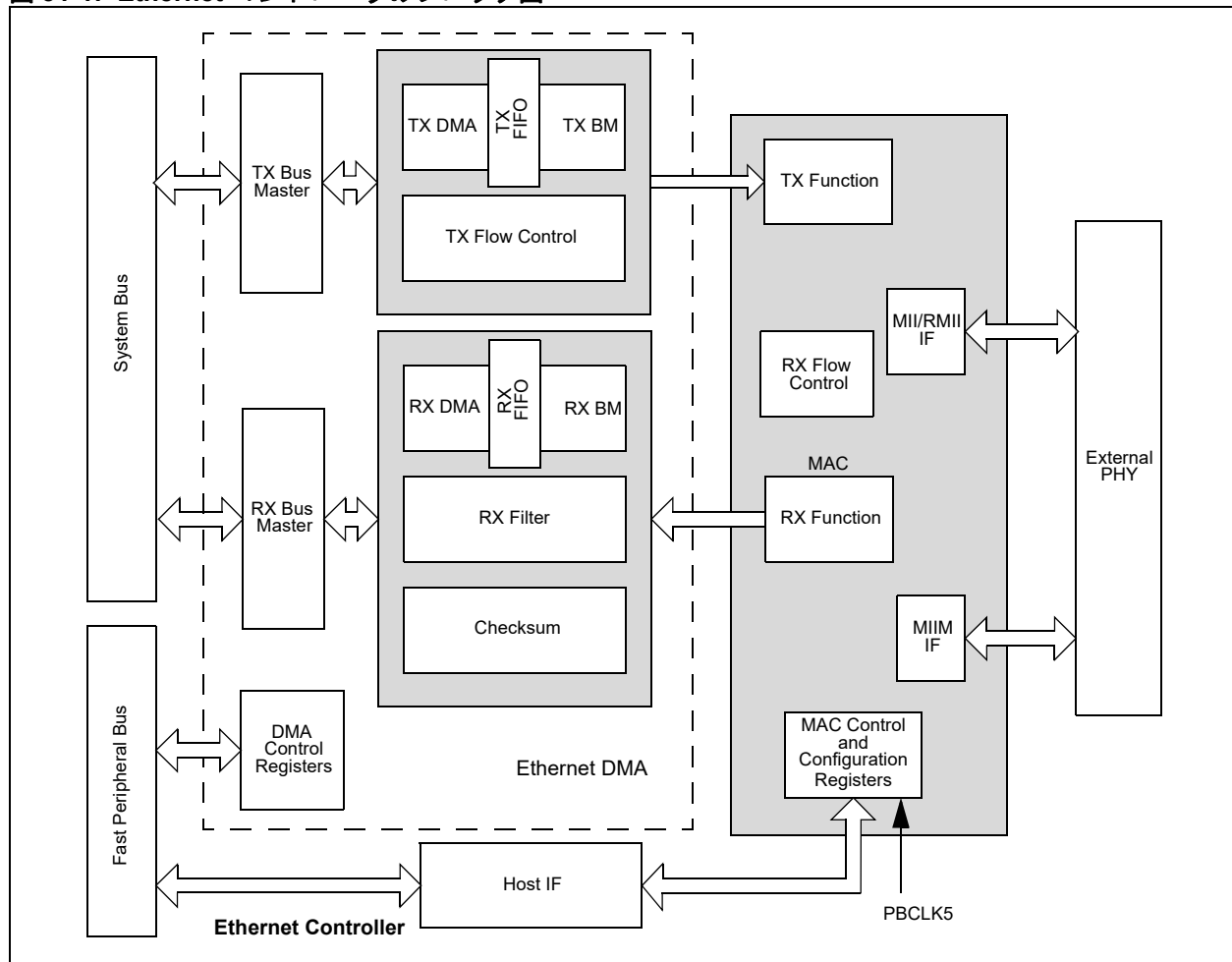
Ethernet コントローラの主な特長は以下の通りです。

- 10/100 Mbps のデータ転送レートをサポート
- 全二重および半二重動作をサポート

- RMII および MII PHY インターフェイスをサポート
- MIIM PHY マネジメント インターフェイスをサポート
- 手動および自動フロー制御をサポート
- 受信パスと送信パスの両方で RAM ディスクリプタベースの DMA 動作をサポート
- 柔軟に設定可能な割り込み
- 設定可能な受信パケットフィルタ処理
  - CRC (巡回冗長検査)
  - 64 バイトのパターンマッチ
  - ブロードキャスト、マルチキャスト、ユニキャストパケット
  - Magic Packet™
  - 64 ビットハッシュテーブル
  - Runt パケット
- パケットペイロードのチェックサム計算をサポート
- 各種のハードウェア統計カウンタをサポート

図 31-1 に Ethernet コントローラのブロック図を示します。

図 31-1: Ethernet コントローラのブロック図



# PIC32MZ グラフィック (DA) ファミリ

表 31-1 と表 31-2 に、Ethernet コントローラで使える 2 通りのインターフェイス モードとそれらのピンを示します。

表 31-1: MII モードの既定値インターフェイス 信号 (FMIEN = 1、FETHIO = 1)

ピン名	概要
EMDC	マネジメントクロック
EMDIO	マネジメント I/O
ETXCLK	送信クロック
ETXEN	送信イネーブル
ETXD0	送信データ
ETXD1	送信データ
ETXD2	送信データ
ETXD3	送信データ
ETXERR	送信エラー
ERXCLK	受信クロック
ERXDV	受信データ有効
ERXD0	受信データ
ERXD1	受信データ
ERXD2	受信データ
ERXD3	受信データ
ERXERR	受信エラー
ECRS	キャリア検出
ECOL	コリジョン表示

表 31-2: RMII モードの既定値インターフェイス 信号 (FMIEN = 0、FETHIO = 1)

ピン名	概要
EMDC	マネジメントクロック
EMDIO	マネジメント I/O
ETXEN	送信イネーブル
ETXD0	送信データ
ETXD1	送信データ
EREFCLK	参照クロック
ECRSDV	キャリア検出 - 受信データ有効
ERXD0	受信データ
ERXD1	受信データ
ERXERR	受信エラー

**Note:** 選択したインターフェイス モードで未使用の Ethernet コントローラピンは、他の周辺モジュール用に使えます。

### 31.1 Ethernet 制御レジスタ

表 31-3: Ethernet コントローラ レジスタの一覧

仮想アドレス (BF8_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
2000	ETHCON1	31:16	PTV<15:0>														0000		
		15:0	ON	—	SIDL	—	—	—	TXRTS	RXEN	AUTOFC	—	—	MANFC	—	—	—	—	BUFCDEC
2010	ETHCON2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	RXBUFSZ<6:0>										0000
2020	ETHTXST	31:16	TXSTADDR<31:16>														0000		
		15:0	TXSTADDR<15:2>											—	—	0000			
2030	ETHRXST	31:16	RXSTADDR<31:16>														0000		
		15:0	RXSTADDR<15:2>											—	—	0000			
2040	ETHHT0	31:16	HT<31:0>														0000		
		15:0															0000		
2050	ETHHT1	31:16	HT<63:32>														0000		
		15:0															0000		
2060	ETHPMM0	31:16	PMM<31:0>														0000		
		15:0															0000		
2070	ETHPMM1	31:16	PMM<63:32>														0000		
		15:0															0000		
2080	ETHPMCS	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PMCS<15:0>														0000		
2090	ETHPMO	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PMO<15:0>														0000		
20A0	ETHRXFC	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	HTEN	MPEN	—	NOTPM	PMMODE<3:0>				CRC ERREN	CRC OKEN	RUNT ERREN	RUNTEN	UCEN	NOT MEEN	MCEN	BCEN	0000
20B0	ETHRXWM	31:16	—	—	—	—	—	—	—	RXFWM<7:0>									0000
		15:0	—	—	—	—	—	—	—	RXEWM<7:0>									0000
20C0	ETHIEN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	TX BUSEIE	RX BUSEIE	—	—	—	EW MARKIE	FW MARKIE	RX DONEIE	PK TPENDIE	RX ACTIE	—	TX DONEIE	TX ABORTIE	RX BUFNAIE	RX OVFLWIE	0000
20D0	ETHIRQ	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	TXBUSE	RXBUSE	—	—	—	EWMARK	FWMARK	RXDONE	PKTPEND	RXACT	—	TXDONE	TXABORT	RXBUFNA	RXOVFLW	0000
20E0	ETHSTAT	31:16	—	—	—	—	—	—	—	BUFCNT<7:0>									0000
		15:0	—	—	—	—	—	—	—	BUSY	TXBUSY	RXBUSY	—	—	—	—	—	—	0000
2100	ETH RXOVFLOW	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RXOVFLWCNT<15:0>														0000		

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の ETHSTAT レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。  
 2: リセット値は工場書き込み値に初期化されます。

表 31-3: Ethernet コントローラ レジスタの一覧 (続き)

仮想アドレス (BF88_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
2110	ETH FRMTXOK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	FRMTXOKCNT<15:0>																0000
2120	ETH SCOLFRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SCOLFRMCNT<15:0>																0000
2130	ETH MCOLFRM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	MCOLFRMCNT<15:0>																0000
2140	ETH FRMRXOK	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	FRMRXOKCNT<15:0>																0000
2150	ETH FCSERR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	FCSERRCNT<15:0>																0000
2160	ETH ALGNERR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ALGNERRCNT<15:0>																0000
2200	EMAC1 CFG1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	SOFT RESET	SIM RESET	—	—	RESET RMCS	RESET RFUN	RESET TMCS	RESET TFUN	—	—	—	—	LOOPBACK	TXPAUSE	RXPAUSE	PASSALL	RXENABLE
2210	EMAC1 CFG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	EXCESS DFR	BP NOBKOFF	NOBKOFF	—	—	LONGPRE	PUREPRE	AUTOPAD	VLANPAD	PAD ENABLE	CRC ENABLE	DELAYCRC	HUGEFRM	LENGTHCK	FULLDPLX	4082
2220	EMAC1 IPGT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	B2BIPKTGP<6:0>																0012
2230	EMAC1 IPGR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	NB2BIPKTGP1<6:0>								—	NB2BIPKTGP2<6:0>							
2240	EMAC1 CLRT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	CWINDOW<5:0>										—	RETX<3:0>					
2250	EMAC1 MAXF	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	MACMAXF<15:0>																05EE
2260	EMAC1 SUPP	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	RESET RMII	—	—	SPEED RMII	—	—	—	—	—	—	—	—	—
2270	EMAC1 TEST	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	TESTBP	TESTPAUSE	SHRTQNTA	0000
2280	EMAC1 MCFG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	RESET MGMT	—	—	—	—	—	—	—	—	—	CLKSEL<3:0>				—	NOPRE	SCANINC
2290	EMAC1 MCMD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SCAN	READ	0000
22A0	EMAC1 MADR	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	PHYADDR<4:0>								—	REGADDR<4:0>							

## 凡例:

x = リセット時に未知の値、— = 未実装 (「0」) として読み出し、リセット値は 16 進表記

- Note
- この表内の ETHSTAT レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。
  - リセット値は工場書き込み値に初期化されます。

表 31-3: Ethernet コントローラ レジスタの一覧 (続き)

仮想アドレス (BF8_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
22B0	EMAC1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
	MWTD	15:0	MWTD<15:0>														0000		
22C0	EMAC1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
	MRDD	15:0	MRDD<15:0>														0000		
22D0	EMAC1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
	MIND	15:0	—	—	—	—	—	—	—	—	—	—	—	—	LINKFAIL	NOTVALID	SCAN	MIIMBUSY	0000
2300	EMAC1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
	SA0 <sup>(2)</sup>	15:0	STNADDR6<7:0>							STNADDR5<7:0>							xxxx		
2310	EMAC1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
	SA1 <sup>(2)</sup>	15:0	STNADDR4<7:0>							STNADDR3<7:0>							xxxx		
2320	EMAC1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
	SA2 <sup>(2)</sup>	15:0	STNADDR2<7:0>							STNADDR1<7:0>							xxxx		

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: この表内の ETHSTAT レジスタを除く全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。  
 2: リセット値は工場書き込み値に初期化されます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 31-1: ETHCON1: Ethernet コントローラ制御レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PTV<15:8>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PTV<7:0>							
15:8	R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
	ON	—	SIDL	—	—	—	TXRTS	RXEN <sup>(1)</sup>
7:0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0
	AUTOFC	—	—	MANFC	—	—	—	BUFCDEC

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-16    **PTV<15:0>**: PAUSE タイマ値ビット  
 フロー制御に使う PAUSE タイマ値です。  
 このレジスタには、RXEN (ETHCON1<8>) をセットしていない場合にのみ書き込めます。  
 これらのビットはフロー制御動作向けにのみ使います。
- bit 15        **ON**: Ethernet ON ビット  
 1 = Ethernet モジュールを有効にする  
 0 = Ethernet モジュールを無効にする
- bit 14        **未実装**: 「0」として読み出し
- bit 13        **SIDL**: アイドル中 Ethernet モジュール停止ビット  
 1 = アイドル中に Ethernet モジュールは転送を停止する  
 0 = アイドル中も Ethernet モジュールは転送を継続する
- bit 12-10    **未実装**: 「0」として読み出し
- bit 9         **TXRTS**: 送信要求ビット  
 1 = TX ロジックをアクティブにして TX EDT で定義されているパケットを送信する  
 0 = 送信を停止する(このビットがソフトウェアによってクリアされた場合)、または送信は完了した(このビットがハードウェアによってクリアされた場合)  
 このビットに「1」が書き込まれた後、送信ロジックが Ethernet ディスクリプタ テーブル (EDT) 内で定義されているパケットの送信を完了すると、このビットは「0」にクリアされます。CPU がこのビットに「0」を書き込んだ場合、送信ロジックは現在のパケットの送信を完了した後に停止します。  
 このビットは TX 動作にのみ影響します。
- bit 8         **RXEN**: 受信イネーブルビット<sup>(1)</sup>  
 1 = RX ロジックを有効にする (フィルタ設定に従って受信パケットを RX バッファに格納する)  
 0 = RX ロジックを無効にする (RX バッファにパケットを受信しない)  
 このビットは RX 動作にのみ影響します。

**Note 1:** RXEN ビットをクリアした後に RX 関連のフィールド / レジスタを変更しない事を推奨します。RX の変更を適用する前に、Ethernet コントローラを再初期化 (ON を「0」にクリア) する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-1: ETHCON1: Ethernet コントローラ制御レジスタ 1 ( 続き )

- bit 7      **AUTOFC:** 自動フロー制御ビット  
1 = 自動フロー制御を有効にする  
0 = 自動フロー制御を無効にする  
このビットをセットすると自動フロー制御が有効になります。このビットがセットされている場合、バッファのフルおよびエンプティ ウォーターマークに応じてフロー制御が自動的に有効/無効になります。受信バッファのカウント (BUFCNT (ETHSTAT<16:23>)) がフル ウォーターマークに達すると、フロー制御は自動的に有効になります。BUFCNT がエンプティ ウォーターマークまで低下すると、フロー制御は自動的に無効になります。  
このビットはフロー制御動作向けにのみ使い、TX と RX の両方の動作に影響します。
- bit 6-5    **未実装:** 「0」として読み出し
- bit 4      **MANFC:** 手動フロー制御ビット  
1 = 手動フロー制御を有効にする  
0 = 手動フロー制御を無効にする  
このビットをセットすると、手動フロー制御が有効になります。このビットがセットされている場合、フロー制御ロジックはPTVレジスタのPAUSEタイマ値に従ってPAUSEフレームを送信します。PAUSEフレームは、このビットがクリアされるまでTXクロックの  $128 * PTV<15:0>/2$  サイクルごとに繰り返し送信されます。  
**Note:** 10 Mbps 動作の場合、TXクロックは2.5 MHzで動作します。100 Mbps 動作の場合、TXクロックは25 MHzで動作します。  
このビットをクリアすると、フロー制御ロジックはPAUSEフレーム (PAUSE タイマ値 = 0x0000) を自動送信してフロー制御を無効にします。  
このビットはフロー制御動作向けにのみ使い、TX と RX の両方の動作に影響します。
- bit 3-1    **未実装:** 「0」として読み出し
- bit 0      **BUFCDEC:** ディスクリプタ バッファカウント デクリメント ビット  
BUFCDEC ビットには「1」だけが書き込み、読み値は常に「0」です。このビットに「1」が書き込まれると、ディスクリプタ バッファカウンタ BUFCNT が1つデクリメントします。このビットの書き込みと同時にRXロジックがBUFCNTをインクリメントした場合、BUFCNTの値は変化しません。このビットに「0」を書き込んでも効果はありません。  
このビットはRX動作向けにのみ使います。
- Note 1:** RXEN ビットをクリアした後にRX関連のフィールド/レジスタを変更しない事を推奨します。RXの変更を適用する前に、Ethernet コントローラを再初期化 (ON を「0」にクリア) する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-2: ETHCON2: Ethernet コントローラ制御レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	
7:0	R/W-0					RXBUF SZ<6:4>			
	RXBUF SZ<3:0>					—	—	—	—

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                    1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-11    **未実装:** 「0」として読み出し

bit 10-4    **RXBUF SZ<6:0>:** 全RXディスクリプタRXデータバッファサイズビット(16バイト刻み)

11111111 = ディスクリプタのRXデータバッファサイズを2032バイトにする

•

•

•

11000000 = ディスクリプタのRXデータバッファサイズを1536バイトにする

•

•

•

00000111 = ディスクリプタのRXデータバッファサイズを48バイトにする

00000101 = ディスクリプタのRXデータバッファサイズを32バイトにする

00000001 = ディスクリプタのRXデータバッファサイズを16バイトにする

00000000 = 予約済み

bit 3-0    **未実装:** 「0」として読み出し

**Note 1:** このレジスタはRX動作向けにのみ使います。

**2:** このレジスタ内のビットは、RXENビット(ETHCON1<8>) = 0の場合にのみ変更できます。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-3: ETHTXST: Ethernet コントローラ TX パケット ディスクリプタ開始アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXSTADDR<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXSTADDR<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TXSTADDR<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
	TXSTADDR<7:2>						—	—

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-2 **TXSTADDR<31:2>**: 最初の送信ディスクリプタ開始アドレスビット  
 送信、受信、DMA 動作の実行中は、このレジスタに書き込まない必要があります。  
 このアドレスは 4 バイト境界 (bit 1-0 は「00」) である必要があります。

bit 1-0 **未実装**: 「0」として読み出し

**Note 1**: このレジスタは TX 動作向けにのみ使います。

**2**: このレジスタは、最後に送信に成功したパケットが使った最後のディスクリプタに、ハードウェアによって自動的に更新されます。

## レジスタ 31-4: ETHRXST: Ethernet コントローラ RX パケット ディスクリプタ開始アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXSTADDR<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXSTADDR<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXSTADDR<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
	RXSTADDR<7:2>						—	—

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-2 **RXSTADDR<31:2>**: 最初の受信ディスクリプタ開始アドレスビット  
 送信、受信、DMA 動作の実行中は、このレジスタに書き込まない必要があります。  
 このアドレスは 4 バイト境界 (bit 1-0 は「00」) である必要があります。

bit 1-0 **未実装**: 「0」として読み出し

**Note 1**: このレジスタは RX 動作向けにのみ使います。

**2**: このレジスタは、最後に受信に成功したパケットが使った最後のディスクリプタに、ハードウェアによって自動的に更新されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-5: ETHHT0: Ethernet コントローラ ハッシュテーブル 0 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
HT<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
HT<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
HT<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
HT<7:0>								

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0    **HT<31:0>**: ハッシュテーブル バイト 0 ~ 3 ビット

**Note 1:** このレジスタは RX 動作向けにのみ使います。  
**2:** このレジスタ内のビットは、RXEN ビット (ETHCON1<8>) = 0 または HTEN ビット (ETHRXFC<15>) = 0 の場合にのみ変更できます。

## レジスタ 31-6: ETHHT1: Ethernet コントローラ ハッシュテーブル 1 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
HT<63:56>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
HT<55:48>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
HT<47:40>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
HT<39:32>								

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0    **HT<63:32>**: ハッシュテーブル バイト 4 ~ 7 ビット

**Note 1:** このレジスタは RX 動作向けにのみ使います。  
**2:** このレジスタ内のビットは、RXEN ビット (ETHCON1<8>) = 0 または HTEN ビット (ETHRXFC<15>) = 0 の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-7: ETHPMM0: Ethernet コントローラ パターンマッチ マスク 0 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMM<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMM<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMM<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMM<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-24    **PMM<31:24>**: パターンマッチ マスク 3 ビット  
 bit 23-16    **PMM<23:16>**: パターンマッチ マスク 2 ビット  
 bit 15-8     **PMM<15:8>**: パターンマッチ マスク 1 ビット  
 bit 7-0      **PMM<7:0>**: パターンマッチ マスク 0 ビット

**Note 1:** このレジスタは RX 動作向けにのみ使います。  
**2:** このレジスタ内のビットは、RXEN ビット (ETHCON1<8>)=0 または PMMODE ビット (ETHRXFC<11:8>)=0 の場合にのみ変更できます。

## レジスタ 31-8: ETHPMM1: Ethernet コントローラ パターンマッチ マスク 1 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMM<63:56>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMM<55:48>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMM<47:40>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMM<39:32>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-24    **PMM<63:56>**: パターンマッチ マスク 7 ビット  
 bit 23-16    **PMM<55:48>**: パターンマッチ マスク 6 ビット  
 bit 15-8     **PMM<47:40>**: パターンマッチ マスク 5 ビット  
 bit 7-0      **PMM<39:32>**: パターンマッチ マスク 4 ビット

**Note 1:** このレジスタは RX 動作向けにのみ使います。  
**2:** このレジスタ内のビットは、RXEN ビット (ETHCON1<8>)=0 または PMMODE ビット (ETHRXFC<11:8>)=0 の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-9: ETHPMCS: Ethernet コントローラ パターンマッチ チェックサム レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMCS<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMCS<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16    **未実装:** 「0」として読み出し  
 bit 15-8    **PMCS<15:8>:** パターンマッチ チェックサム 1 ビット  
 bit 7-0     **PMCS<7:0>:** パターンマッチ チェックサム 0 ビット

**Note 1:** このレジスタは RX 動作向けにのみ使います。  
**2:** このレジスタ内のビットは、RXEN ビット (ETHCON1<8>)=0 または PMMODE ビット (ETHRXFC<11:8>)=0 の場合にのみ変更できます。

## レジスタ 31-10: ETHPMO: Ethernet コントローラ パターンマッチ オフセット レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMO<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PMO<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16    **未実装:** 「0」として読み出し  
 bit 15-0    **PMO<15:0>:** パターンマッチ オフセット 1 ビット

**Note 1:** このレジスタは RX 動作向けにのみ使います。  
**2:** このレジスタ内のビットは、RXEN ビット (ETHCON1<8>)=0 または PMMODE ビット (ETHRXFC<11:8>)=0 の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-11: ETHRXFC: Ethernet コントローラ受信フィルタ コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	HTEN	MPEN	—	NOTPM	PMMODE<3:0>			
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CRCERREN	CRCKOKEN	RUNTERREN	RUNTEN	UCEN	NOTMEEN	MCEN	BCEN

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット

U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15 **HTEN**: ハッシュテーブル フィルタ処理イネーブルビット

1 = ハッシュテーブル フィルタ処理を有効にする

0 = ハッシュテーブル フィルタ処理を無効にする

bit 14 **MPEN**: Magic Packet™ イネーブルビット

1 = Magic Packet フィルタ処理を有効にする

0 = Magic Packet フィルタ処理を無効にする

bit 13 **未実装**: 「0」として読み出し

bit bit 12 **NOTPM**: パターンマッチ反転ビット

1 = パターンマッチが成立するにはパターンマッチ チェックサムが一致してはならない

0 = パターンマッチが成立するにはパターンマッチ チェックサムが一致する必要がある

このビットは、パターンマッチが成立すると判定されるためのパターンマッチ チェックサムの条件 (一致または不一致) を指定します。

bit 11-8 **PMMODE<3:0>**: パターンマッチ モード ビット

1001 = (NOTPM = 1 XOR パターンマッチ チェックサム一致) AND (パケット = Magic Packet) が真の場合、パターンマッチが成立する (1,3)

1000 = (NOTPM = 1 XOR パターンマッチ チェックサム一致) AND (ハッシュテーブル フィルタ一致) が真の場合、パターンマッチが成立する (1,1)

0111 = (NOTPM = 1 XOR パターンマッチ チェックサム一致) AND (デスティネーションアドレス = ブロードキャストアドレス) が真の場合、パターンマッチが成立する (1)

0110 = (NOTPM = 1 XOR パターンマッチ チェックサム一致) AND (デスティネーションアドレス = ブロードキャストアドレス) が真の場合、パターンマッチが成立する (1)

0101 = (NOTPM = 1 XOR パターンマッチ チェックサム一致) AND (デスティネーションアドレス = ユニキャストアドレス) が真の場合、パターンマッチが成立する (1)

0100 = (NOTPM = 1 XOR パターンマッチ チェックサム一致) AND (デスティネーションアドレス = ユニキャストアドレス) が真の場合、パターンマッチが成立する (1)

0011 = (NOTPM = 1 XOR パターンマッチ チェックサム一致) AND (デスティネーションアドレス = ステーションアドレス) が真の場合、パターンマッチが成立する (1)

0010 = (NOTPM = 1 XOR パターンマッチ チェックサム一致) AND (デスティネーションアドレス = ステーションアドレス) が真の場合、パターンマッチが成立する (1)

0001 = (NOTPM = 1 XOR パターンマッチ チェックサム一致) が真の場合、パターンマッチが成立する (1)

0000 = パターンマッチを無効にする (パターンマッチは常に不成立)

**Note 1:** XOR: 2つの条件の片方が真でもう片方が偽の場合に結果が真となる論理演算

**2:** このハッシュテーブル フィルタマッチは HTEN ビットの値に関係なくアクティブです。

**3:** この Magic Packet フィルタマッチは MPEN ビットの値に関係なくアクティブです。

**Note 1:** このレジスタは RX 動作向けにのみ使います。

**2:** このレジスタ内のビットは、RXEN ビット (ETHCON1<8>) = 0 の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-11: ETHRXFC: Ethernet コントローラ受信フィルタ コンフィグレーション レジスタ (続き)

- bit 7 **CR CERREN:** CRC エラー収集イネーブルビット  
1 = CRC が無効の受信パケットのみ承認する  
0 = CRC エラー収集フィルタ処理を無効にする  
このビットを使うと、CRC が無効のパケットを全て収集できます。
- bit 6 **CR COKEN:** CRC OK イネーブルビット  
1 = CRC が有効の受信パケットのみ承認する  
0 = CRC フィルタ処理を無効にする  
このビットを使うと、CRC が無効のパケットを全て拒否できます。
- bit 5 **RUNTERREN:** Runt エラー収集イネーブルビット  
1 = Runt の受信パケットのみ承認する  
0 = Runt エラー収集フィルタ処理を無効にする  
このビットを使うと、Runt パケットを全て収集できます。このフィルタでは、サイズが 64 バイト未満のパケット (CR COKEN = 0 の場合) またはサイズが 64 バイト未満かつ CRC が有効なパケット (CR COKEN = 1 の場合) を Runt パケットとして定義しています。
- bit 4 **RUNTEN:** Runt イネーブルビット  
1 = 非 Runt の受信パケットのみ承認する  
0 = Runt フィルタ処理を無効にする  
このビットを使うと、Runt パケットを全て拒否できます。このフィルタでは、サイズが 64 バイト未満の全てのパケットを Runt パケットとして定義しています。
- bit 3 **UCEN:** ユニキャスト イネーブルビット  
1 = ユニキャスト フィルタ処理を有効にする  
0 = ユニキャスト フィルタ処理を無効にする  
このビットを使うと、デスティネーション アドレスとステーション アドレスが一致する全てのユニキャストパケットを承認できます。
- bit 2 **NOTMEEN:** Not Me ユニキャスト イネーブルビット  
1 = Not Me ユニキャスト フィルタ処理を有効にする  
0 = Not Me ユニキャスト フィルタ処理を無効にする  
このビットを使うと、デスティネーション アドレスとステーション アドレスが一致しない全てのユニキャストパケットを承認できます。
- bit 1 **MCEN:** マルチキャスト イネーブルビット  
1 = マルチキャスト フィルタ処理を有効にする  
0 = マルチキャスト フィルタ処理を無効にする  
このビットを使うと、マルチキャスト アドレスのパケットを全て承認できます。
- bit 0 **BCEN:** ブロードキャスト イネーブルビット  
1 = ブロードキャスト フィルタ処理を有効にする  
0 = ブロードキャスト フィルタ処理を無効にする  
このビットを使うと、ブロードキャスト アドレスのパケットを全て承認できます。

- Note 1:** XOR: 2 つの条件の片方が真でもう片方が偽の場合に結果が真となる論理演算  
**2:** このハッシュテーブル フィルタマッチは HTEN ビットの値に関係なくアクティブです。  
**3:** この Magic Packet フィルタマッチは MPEN ビットの値に関係なくアクティブです。

- Note 1:** このレジスタは RX 動作向けにのみ使います。  
**2:** このレジスタ内のビットは、RXEN ビット (ETHCON1<8>) = 0 の場合にのみ変更できます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-12: ETHRXWM: Ethernet コントローラ受信ウォーターマーク レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXFWM<7:0>							
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXEWM<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-24    **未実装:** 「0」として読み出し

bit 23-16    **RXFWM<7:0>:** 受信フル ウォーターマーク ビット

ソフトウェア制御の RX バッファフル ウォーターマーク ポインタを RX BUF CNT と比較する事で、フル ウォーターマークの条件を判定します。この判定に基づき、FWMARK 割り込みを生成すると共に、自動フロー制御が有効な場合はフロー制御を有効にします。フル ウォーターマーク ポインタは、エンプティ ウォーターマーク ポインタより大きい必要があります。

bit 15-8    **未実装:** 「0」として読み出し

bit 7-0    **RXEWM<7:0>:** 受信エンプティ ウォーターマーク ビット

ソフトウェア制御の RX バッファ エンプティ ウォーターマーク ポインタを RX BUF CNT と比較する事で、エンプティ ウォーターマークの条件を判定します。この判定に基づき、EWMARK 割り込みを生成すると共に、自動フロー制御が有効な場合にはフロー制御を無効にします。エンプティ ウォーターマーク ポインタは、フル ウォーターマーク ポインタより小さい必要があります。

**Note:** このレジスタは RX 動作向けにのみ使います。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-13: ETHIEN: Ethernet コントローラ割り込みイネーブル レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	TXBUSEIE <sup>(1)</sup>	RXBUSEIE <sup>(2)</sup>	—	—	—	EWMARKIE <sup>(2)</sup>	FWMARKIE <sup>(2)</sup>
7:0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXDONEIE <sup>(2)</sup>	PKTPENDIE <sup>(2)</sup>	RXACTIE <sup>(2)</sup>	—	TXDONEIE <sup>(1)</sup>	TXABORTIE <sup>(1)</sup>	RXBUFNAIE <sup>(2)</sup>	RXOVFLWIE <sup>(2)</sup>

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット      0 = ビットはクリア              x = ビットは未知

bit 31-15 **未実装**: 「0」として読み出し

bit 14 **TXBUSEIE**: 送信 BVCI バスエラー割り込みイネーブルビット<sup>(1)</sup>

1 = TXBUS エラー割り込みを有効にする  
 0 = TXBUS エラー割り込みを無効にする

bit 13 **RXBUSEIE**: 受信 BVCI バスエラー割り込みイネーブルビット<sup>(2)</sup>

1 = RXBUS エラー割り込みを有効にする  
 0 = RXBUS エラー割り込みを無効にする

bit 12-10 **未実装**: 「0」として読み出し

bit 9 **EWMARKIE**: エンプティ ウォーターマーク割り込みイネーブルビット<sup>(2)</sup>

1 = EWMARK 割り込みを有効にする  
 0 = EWMARK 割り込みを無効にする

bit 8 **FWMARKIE**: フル ウォーターマーク割り込みイネーブルビット<sup>(2)</sup>

1 = FWMARK 割り込みを有効にする  
 0 = FWMARK 割り込みを無効にする

bit 7 **RXDONEIE**: レシーバ完了割り込みイネーブルビット<sup>(2)</sup>

1 = RXDONE 割り込みを有効にする  
 0 = RXDONE 割り込みを無効にする

bit 6 **PKTPENDIE**: パケット保留割り込みイネーブルビット<sup>(2)</sup>

1 = PKTPEND 割り込みを有効にする  
 0 = PKTPEND 割り込みを無効にする

bit 5 **RXACTIE**: RX アクティビティ割り込みイネーブルビット<sup>(2)</sup>

1 = RXACT 割り込みを有効にする  
 0 = RXACT 割り込みを無効にする

bit 4 **未実装**: 「0」として読み出し

bit 3 **TXDONEIE**: トランスミッタ完了割り込みイネーブルビット<sup>(1)</sup>

1 = TXDONE 割り込みを有効にする  
 0 = TXDONE 割り込みを無効にする

bit 2 **TXABORTIE**: トランスミッタ中止割り込みイネーブルビット<sup>(1)</sup>

1 = TXABORT 割り込みを有効にする  
 0 = TXABORT 割り込みを無効にする

bit 1 **RXBUFNAIE**: 受信バッファ使用不可割り込みイネーブルビット<sup>(2)</sup>

1 = RXBUFNA 割り込みを有効にする  
 0 = RXBUFNA 割り込みを無効にする

bit 0 **RXOVFLWIE**: 受信 FIFO オーバーフロー割り込みイネーブルビット<sup>(2)</sup>

1 = RXOVFLW 割り込みを有効にする  
 0 = RXOVFLW 割り込みを無効にする

**Note 1:** このビットは TX 動作向けにのみ使います。

**2:** このビットは RX 動作向けにのみ使います。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-14: ETHIRQ: Ethernet コントローラ割り込み要求レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	TXBUSE	RXBUSE	—	—	—	EWMARK	FWMARK
7:0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXDONE	PKTPEND	RXACT	—	TXDONE	TXABORT	RXBUFNA	RXOVFLW

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-15 **未実装:** 「0」として読み出し

bit 14 **TXBUSE:** 送信 BVCI バスエラー割り込みビット (2)

1 = BVCI バスエラーが発生した

0 = BVCI バスエラーは発生していない

このビットは、TX DMA によるメモリアクセス中に BVCI バスエラーが発生した時にセットされます。このビットは、リセット時にクリアされる他、CPU が CLR レジスタに「1」を書き込んだ時にクリアされます。

bit 13 **RXBUSE:** 受信 BVCI バスエラー割り込みビット (2)

1 = BVCI バスエラーが発生した

0 = BVCI バスエラーは発生していない

このビットは、RX DMA によるメモリアクセス中に BVCI バスエラーが発生した時にセットされます。このビットは、リセット時にクリアされる他、CPU が CLR レジスタに「1」を書き込んだ時にクリアされます。

bit 12-10 **未実装:** 「0」として読み出し

bit 9 **EWMARK:** エンプティ ウォーターマーク割り込みビット (2)

1 = エンプティ ウォーターマーク ポインタに達した

0 = この割り込みは保留中ではない

このビットは、RX ディスクリプタ バッファカウントが RXEWM ビット (ETHRXWM<0:7>) の値以下である場合にセットされます。このビットは、BUFCNT ビット (ETHSTAT<16:23>) がハードウェアによってインクリメントされる事によりクリアされます。このビットに「0」または「1」を書き込んでも効果はありません。

bit 8 **FWMARK:** フル ウォーターマーク割り込みビット (2)

1 = フル ウォーターマーク ポインタに達した

0 = この割り込みは保留中ではない

このビットは、RX ディスクリプタ バッファカウントが RXFWM ビット (ETHRXWM<16:23>) フィールドの値以上である場合にセットされます。このビットは、BUFCDEC (ETHCON1<0>) への書き込み (BUFCNT カウンタのデクリメント) によりクリアされます。このビットに「0」または「1」を書き込んでも効果はありません。

**Note 1:** このビットは TX 動作向けにのみ使います。

**2:** このビットは RX 動作向けにのみ使います。

**Note:** このレジスタ内のビットは SET、CLR、INV レジスタを使ってセット / クリアする事を推奨します。このレジスタ内のビットをセット / クリアするのは、デバッグまたはテストを目的とする場合に限りです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-14: ETHIRQ: Ethernet コントローラ割り込み要求レジスタ (続き)

bit 7	<b>RXDONE:</b> 受信完了割り込みビット <sup>(2)</sup> 1 = RX パケットの受信に成功した 0 = この割り込みは保留中ではない このビットは、RX パケットの受信に成功するとセットされます。このビットは、リセット時にクリアされる他、CPU が CLR レジスタに「1」を書き込んだ時にクリアされます。
bit 6	<b>PKTPEND:</b> パケット保留割り込みビット <sup>(2)</sup> 1 = RX パケットがメモリ内で保留中 0 = RX パケットは保留中ではない このビットは、BUFCNT カウンタの値が「0」以外の場合にセットされます。このビットは、リセットによってクリアされる他、BUFCDEC への書き込み (BUFCNT カウンタのデクリメント) によってクリアされます。このビットに「0」または「1」を書き込んででも効果はありません。
bit 5	<b>RXACT:</b> 受信アクティビティ割り込みビット <sup>(2)</sup> 1 = RX パケットデータの受信に成功した 0 = この割り込みは保留中ではない このビットは、RX パケットデータが RXBM FIFO に格納されるとセットされます。このビットは、リセット時にクリアされる他、CPU が CLR レジスタに「1」を書き込んだ時にクリアされます。
bit 4	<b>未実装:</b> 「0」として読み出し
bit 3	<b>TXDONE:</b> 送信完了割り込みビット <sup>(2)</sup> 1 = TX パケットの送信に成功した 0 = この割り込みは保留中ではない このビットは、現在処理中の TX パケットの送信が完了し、このパケットに使った最初のディスクリプタに送信ステータスペクタが読み込まれた時点でセットされます。このビットは、リセット時にクリアされる他、CPU が CLR レジスタに「1」を書き込んだ時にクリアされます。
bit 2	<b>TXABORT:</b> 送信中止条件割り込みビット <sup>(2)</sup> 1 = 直近の TX パケットで TX 中止条件が発生した 0 = この割り込みは保留中ではない このビットは、以下の理由のいずれかによって MAC が TX パケットの送信を中止した時にセットされます。 <ul style="list-style-type: none"><li>ジャンボ TX パケットによる中止</li><li>アンダーランによる中止</li><li>過度な遅れによる中止</li><li>レイトコリジョン (Late collision) による中止</li><li>過度なコリジョンによる中止</li></ul> このビットは、リセット時にクリアされる他、CPU が CLR レジスタに「1」を書き込んだ時にクリアされます。
bit 1	<b>RXBUFNA:</b> 受信バッファ使用不可割り込みビット <sup>(2)</sup> 1 = RX バッファ ディスクリプタ使用不可条件が発生した 0 = この割り込みは保留中ではない このビットは、RX バッファ ディスクリプタのオーバーラン条件によってセットされます。このビットは、リセット時にクリアされる他、CPU が CLR レジスタに「1」を書き込んだ時にクリアされます。
bit 0	<b>RXOVFLW:</b> 受信 FIFO オーバーフロー エラービット <sup>(2)</sup> 1 = RX FIFO オーバーフロー エラー条件が発生した 0 = この割り込みは保留中ではない RXOVFLW は、RX FIFO のオーバーフロー条件が発生した時に RXBM ロジックによってセットされません。このビットは、リセット時にクリアされる他、CPU が CLR レジスタに「1」を書き込んだ時にクリアされます。

- Note 1:** このビットは TX 動作向けにのみ使います。  
**Note 2:** このビットは RX 動作向けにのみ使います。

**Note:** このレジスタ内のビットは SET、CLR、INV レジスタを使ってセット / クリアする事を推奨します。このレジスタ内のビットをセット / クリアするのは、デバッグまたはテストを目的とする場合に限りです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-15: ETHSTAT: Ethernet コントローラ ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BUFCNT<7:0> <sup>(1)</sup>							
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
	ETHBUSY <sup>(5)</sup>	TXBUSY <sup>(2,6)</sup>	RXBUSY <sup>(3,6)</sup>	—	—	—	—	—

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-24 **未実装**: 「0」として読み出し

bit 23-16 **BUFCNT<7:0>**: パケットバッファ カウントビット <sup>(1)</sup>

メモリに受信したパケットバッファの数です。パケットの受信に成功すると、そのパケットが使っているディスクリプタの数に基づいてハードウェアがこのレジスタをインクリメントします。ソフトウェアは、パケットをバッファから読み出した後に、このカウンタをデクリメントします (使われているディスクリプタごとに BUFCDEC ビット (ETHCON1<0>) に書き込みます)。レジスタが既に 0xFF に達している場合、ハードウェアがさらにインクリメントしようとしても、このレジスタは 0xFF から 0x00 にロールオーバーしません。逆にレジスタが既に 0x0000 に達している場合、ソフトウェアがさらにデクリメントしようとしても、このレジスタは 0x00 から 0xFF にロールオーバーしません。ソフトウェアがカウンタをデクリメントすると同時にハードウェアがカウンタをインクリメントした場合、カウンタの値は変化しません。

自動フロー制御が有効な場合、このレジスタの値が 0xFF に達すると RX ロジックは停止し、ソフトウェアが BUFCDEC ビットに書き込んでレジスタを 0xFF よりも小さな値にデクリメントするまで待機します。

自動フロー制御が無効の場合、RXDMA は処理を継続し、BUFCNT の値は 0xFF で飽和します。

このレジスタが 0 以外の値の場合、PKTPEND ステータスビットがセットされ、ETHIEN レジスタの PKTPENDIE ビットがセットされていれば、割り込みが生成されます。

BUFCNT カウンタは、ETHRXST レジスタが書き込まれた時に自動的に 0x00 にクリアされます。

**Note:** ON が「0」に設定されている場合、BUFCNT はクリアされません。ソフトウェアは引き続きこのカウンタをデクリメントできます。

bit 15-8 **未実装**: 「0」として読み出し

bit 7 **ETHBUSY**: Ethernet モジュール ビジービット <sup>(4,5)</sup>

1 = Ethernet ロジックは有効 (ON (ETHCON1<15>) = 1) またはトランザクションの終了処理中  
 0 = Ethernet ロジックはアイドル中

このビットは、モジュールが有効である事、または無効にされた後にトランザクションの終了処理中である事を示します。

- Note 1:** このビットは RX 動作向けにのみ使います。  
**2:** このビットは TX 動作にのみ影響されます。  
**3:** このビットは RX 動作にのみ影響されます。  
**4:** このビットは TX および RX 動作に影響されます。  
**5:** このビットは ON ビット (ETHCON1<15>) = 1 の場合に**セット**されます。  
**6:** このビットは ON ビット (ETHCON1<15>) = 0 の場合に**クリア**されます。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 31-15: ETHSTAT: Ethernet コントローラ ステータス レジスタ (続き)

bit 6 **TXBUSY:** 送信ビジービット (2,6)

1 = TX ロジックはデータを送信中  
0 = TX ロジックはアイドル中

このビットは、現在パケットを送信中である事を示します。このステータスビットの変化は、必ずしも TXDONE 割り込みには反映されません (TX パケットは中止されるか MAC によって拒否される可能性があるため)。

bit 5 **RXBUSY:** 受信ビジービット (3,6)

1 = RX ロジックはデータを受信中  
0 = RX ロジックはアイドル中

このビットは、現在パケットを受信中である事を示します。このステータスビットの変化は、必ずしも RXDONE 割り込みには反映されません (RX パケットは中止されるか RX フィルタによって排除される可能性があるため)。

bit 4-0 **未実装:** 「0」として読み出し

- Note**
- 1: このビットは RX 動作向けにのみ使います。
  - 2: このビットは TX 動作にのみ影響されます。
  - 3: このビットは RX 動作にのみ影響されます。
  - 4: このビットは TX および RX 動作に影響されます。
  - 5: このビットは ON ビット (ETHCON1<15>) = 1 の場合に**セット**されます。
  - 6: このビットは ON ビット (ETHCON1<15>) = 0 の場合に**クリア**されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-16: ETHRXOVFLOW: Ethernet コントローラ受信オーバーフロー統計レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXOVFLWCNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXOVFLWCNT<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16    **未実装:** 「0」として読み出し

bit 15-0    **RXOVFLWCNT<15:0>:** 取りこぼし受信フレームカウント ビット

RX フィルタは許可したものの、その後の内部受信エラー(RXFIFO オーバーラン)によって取りこぼしたフレームを数えるインクリメント カウンタです。このイベントは、RXOVFLW ビット (ETHIRQ<0>) 割り込みフラグもセットします。

- Note 1:** このレジスタは RX 動作向けにのみ使います。
- 2:** バイト 0/1 に対するバイトイネーブルが「0」ではない場合、このレジスタは読み出し動作後にハードウェアによって自動的にクリアされます。
- 3:** このレジスタ内のビットは SET、CLR、INV レジスタを使ってセット/クリアする事を推奨します。このレジスタ内のビットをセット/クリアするのは、デバッグまたはテストを目的とする場合に限りです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-17: ETHFRMTXOK: Ethernet コントローラ フレーム送信成功統計レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FRMTXOKCNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FRMTXOKCNT<7:0>							

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット   U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **FRMTXOKCNT<15:0>**: フレーム送信成功カウントビット  
 送信に成功したフレーム数を数えるインクリメント カウンタです。

**Note 1:** このレジスタは TX 動作向けにのみ使います。

**2:** バイト 0/1 に対するバイトイネーブルが「0」ではない場合、このレジスタは読み出し動作後にハードウェアによって自動的にクリアされます。

**3:** このレジスタ内のビットは SET、CLR、INV レジスタを使ってセット/クリアする事を推奨します。このレジスタ内のビットをセット/クリアするのは、デバッグまたはテストを目的とする場合に限りです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-18: ETHSCOLFRM: Ethernet コントローラ単一コリジョン フレーム統計レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SCOLFRMCNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SCOLFRMCNT<7:0>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15-0 **SCOLFRMCNT<15:0>:** 単一コリジョン フレームカウント ビット

2 回目の試行で送信に成功したフレーム数を数えるインクリメント カウンタです。

**Note 1:** このレジスタは TX 動作向けにのみ使います。

**2:** バイト 0/1 に対するバイトイネーブルが「0」ではない場合、このレジスタは読み出し動作後にハードウェアによって自動的にクリアされます。

**3:** このレジスタ内のビットは SET、CLR、INV レジスタを使ってセット/クリアする事を推奨します。このレジスタ内のビットをセット/クリアするのは、デバッグまたはテストを目的とする場合に限りです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-19: ETHMCOLFRM: Ethernet コントローラ複数コリジョン フレーム統計レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MCOLFRMCNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MCOLFRMCNT<7:0>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **MCOLFRMCNT<15:0>**: 複数コリジョン フレームカウント ビット

複数回のコリジョンが発生した後に送信に成功したフレーム数を数えるインクリメント カウンタです。

**Note 1:** このレジスタは TX 動作向けにのみ使います。

**2:** バイト 0/1 に対するバイトイネーブルが「0」ではない場合、このレジスタは読み出し動作後にハードウェアによって自動的にクリアされます。

**3:** このレジスタ内のビットは SET、CLR、INV レジスタを使ってセット/クリアする事を推奨します。このレジスタ内のビットをセット/クリアするのは、デバッグまたはテストを目的とする場合に限りです。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-20: ETHFRMRXOK: Ethernet コントローラ フレーム受信成功統計レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FRMRXOKCNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FRMRXOKCNT<7:0>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **FRMRXOKCNT<15:0>**: フレーム受信成功カウントビット

RX フィルタが受信に成功したフレーム数を数えるインクリメント カウンタです。フレームチェックシーケンス (FCS) エラーまたはアラインメント エラー発生時はインクリメントしません。

**Note 1:** このレジスタは RX 動作向けにのみ使います。

**2:** バイト 0/1 に対するバイトイネーブルが「0」ではない場合、このレジスタは読み出し動作後にハードウェアによって自動的にクリアされます。

**3:** このレジスタ内のビットは SET、CLR、INV レジスタを使ってセット/クリアする事を推奨します。このレジスタ内のビットをセット/クリアするのは、デバッグまたはテストを目的とする場合に限りです。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 31-21: ETHFCSERR: Ethernet コントローラ フレームチェック シーケンス エラー統計レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FCSERRCNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FCSERRCNT<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15-0 **FCSERRCNT<15:0>**: FCS エラーカウンタ ビット

FCSエラー受信フレームを数えるインクリメントカウンタです。フレーム長(ビット数)は8の整数倍です。

**Note 1:** このレジスタはRX 動作向けにのみ使います。

**2:** バイト 0/1 に対するバイトイネーブルが「0」ではない場合、このレジスタは読み出し動作後にハードウェアによって自動的にクリアされます。

**3:** このレジスタ内のビットはSET、CLR、INV レジスタを使ってセット/クリアする事を推奨します。このレジスタ内のビットをセット/クリアするのは、デバッグまたはテストを目的とする場合に限りです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-22: ETHALGNERR: Ethernet コントローラ アラインメント エラー統計レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ALGNERRCNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ALGNERRCNT<7:0>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **ALGNERRCNT<15:0>**: アラインメント エラーカウンタ ビット

アラインメント エラー フレームを数えるインクリメント カウンタです。アラインメント エラー フレームとは、FCS エラーが発生したフレームおよびフレーム長が 8 ビットの整数倍ではないフレーム (ドリブルニブル) です。

**Note 1:** このレジスタは RX 動作向けにのみ使います。

**2:** バイト 0/1 に対するバイトイネーブルが「0」ではない場合、このレジスタは読み出し動作後にハードウェアによって自動的にクリアされます。

**3:** このレジスタ内のビットは SET、CLR、INV レジスタを使ってセット/クリアする事を推奨します。このレジスタ内のビットをセット/クリアするのは、デバッグまたはテストを目的とする場合に限りです。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 31-23: EMAC1CFG1: Ethernet コントローラ MAC コンフィグレーション1 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-1	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	SOFT RESET	SIM RESET	—	—	RESET RMCS	RESET RFUN	RESET TMCS	RESET TFUN
7:0	U-0	U-0	U-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-1
	—	—	—	LOOPBACK	TX PAUSE	RX PAUSE	PASSALL	RX ENABLE

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                    1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **SOFTRESET:** ソフトリセット ビット

このビットをセットする事で MACMII をリセット状態にします。既定値は「1」です。

bit 14 **SIMRESET:** シミュレーションリセットビット

このビットをセットする事で送信機能に含まれる乱数生成器をリセットします。

bit 13-12 **未実装:** 「0」として読み出し

bit 11 **RESETRMCS:** MCS/RX リセットビット

このビットをセットする事で MAC 制御副層 / 受信ドメインのロジックをリセット状態にします。

bit 10 **RESETRFUN:** RX 機能リセットビット

このビットをセットする事で MAC 受信機能のロジックをリセット状態にします。

bit 9 **RESETTMCS:** MCS/TX リセットビット

このビットをセットする事で MAC 制御副層 / TX ドメインのロジックをリセット状態にします。

bit 8 **RESETTFUN:** TX 機能リセットビット

このビットをセットする事で MAC 送信機能のロジックをリセット状態にします。

bit 7-5 **未実装:** 「0」として読み出し

bit 4 **LOOPBACK:** MAC ループバック モード ビット

1 = MAC 送信インターフェイスを MAC 受信インターフェイスにループバックする  
 0 = MAC は通常動作する

bit 3 **TXPAUSE:** MAC TX フロー制御ビット

1 = PAUSE フロー制御フレームの送信を許可する  
 0 = PAUSE フロー制御フレームの送信を許可しない

bit 2 **RXPAUSE:** MAC RX フロー制御ビット

1 = MAC は受信した PAUSE フロー制御フレームに対応する  
 0 = MAC は受信した PAUSE フロー制御フレームを無視する

bit 1 **PASSALL:** 全受信フレーム MAC 承認ビット

1 = MAC はタイプ (通常、制御) に関わらず受信した全てのフレームを承認する  
 0 = MAC は受信した制御フレームを無視する

bit 0 **RXENABLE:** MAC 受信イネーブルビット

1 = MAC によるフレームの受信を有効にする  
 0 = MAC によるフレームの受信を無効にする

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-24: EMAC1CFG2: Ethernet コントローラ MAC コンフィグレーション 2 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 25/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
	—	EXCESS DFR	BPNOBK OFF	NOBK OFF	—	—	LONGPRE	PUREPRE
7:0	R/W-1	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-1	R/W-0
	AUTO PAD <sup>(1,2)</sup>	VLAN PAD <sup>(1,2)</sup>	PAD ENABLE <sup>(1,3)</sup>	CRC ENABLE	DELAYCRC	HUGEFRM	LENGTHCK	FULLDPLX

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-15 **未実装**: 「0」として読み出し

bit 14 **EXCESSDER**: 遅延時間オーバービット

- 1 = 規格に従い、MAC は他のキャリア信号を検出した場合にメッセージの送信を無期限に遅延させる
- 0 = MAC は遅延時間リミットに達した時点で中止する

bit 13 **BPNOBKOFF**: バックプレッシャ/ノーバックオフ ビット

- 1 = バックプレッシャ中に偶発的なコリジョンが発生した場合、MAC はバックオフを適用せずにただちに再送信し、後続のコリジョンを抑制してパケットを確実に送信する
- 0 = MAC はバックオフを適用する

bit 12 **NOBKOFF**: ノーバックオフ ビット

- 1 = コリジョンの発生後、MAC はただちに再送信する (規格が定めるバイナリ エクスポネンシャルバックオフアルゴリズムを適用しない)
- 0 = コリジョンの発生後、MAC はバイナリ エクスポネンシャルバックオフアルゴリズムを適用する

bit 11-10 **未実装**: 「0」として読み出し

bit 9 **LONGPRE**: ロング プリアンブル適用ビット

- 1 = MAC はプリアンブル フィールドの長さが 12 バイト未満の受信パケットのみ許可する
- 0 = 規格に従い、MAC はプリアンブルの長さに関わらず全ての受信パケットを許可する

bit 8 **PUREPRE**: ピュア プリアンブル適用ビット

- 1 = MAC はプリアンブルの内容(0x55が含まれ、エラーがない事)を確認する(プリアンブルにエラーを含むパケットは破棄する)
- 0 = MAC はプリアンブルを一切確認しない

bit 7 **AUTOPAD**: 自動パディング検出イネーブルビット<sup>(1,2)</sup>

- 1 = MAC はフレームのタイプ ( タグの有無 ) を自動的に検出 ( ソースアドレスに続く 2 オクテットを 0x8100(VLAN プロトコル ID) と比較 ) してパディングする
- 0 = MAC は自動検出を実行しない

**Note 1:** 表 31-4 に、このレジスタの設定に基づくパディング機能を示します。

**2:** PADENABLE ビットがクリアされている場合、このビットは無視されます。

**3:** このビットは AUTOPAD および VLANPAD ビットと組み合わせて使います。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-24: EMAC1CFG2: Ethernet コントローラ MAC コンフィグレーション 2 レジスタ (続き)

- bit 6 **VLANPAD:** VLAN パディング イネーブルビット (1,2)  
 1 = MAC は短いフレームの全てを 64 バイトになるようにパディングして有効な CRC を付加する  
 0 = MAC は短いフレームのパディングを実行しない
- bit 5 **PADENABLE:** パディング /CRC イネーブルビット (1,3)  
 1 = MAC は短いフレームを全てパディングする  
 0 = MAC に供給されるフレームは有効な長さを持つ
- bit 4 **CRCENABLE:** CRC イネーブル 1 ビット  
 1 = MAC はパディングの要否に関わらず全てのフレームにCRCを付加する(PADENABLEビットをセットした場合、このビットもセットする必要があります)  
 0 = MAC に供給されるフレームには有効な CRC が付加されている
- bit 3 **DELAYCRC:** 遅延 CRC ビット  
 このビットは、IEEE 802.3 フレームの先頭に存在する独自のヘッダ情報のバイト数を指定します。  
 1 = 4 バイトのヘッダ (CRC 機能によって無視されます)  
 0 = 独自ヘッダなし
- bit 2 **HUGEFRM:** 巨大フレーム イネーブルビット  
 1 = 送受信フレームの長さを制限しない  
 0 = 巨大フレームの送受信を禁止する
- bit 1 **LENGTHCK:** フレーム長チェックビット  
 1 = 送受信両方のフレーム長を「長さ / タイプ」フィールドと比較する (「長さ / タイプ」フィールドが長さを表す場合にチェックを実行。不一致は送信 / 受信統計ベクタで報告)  
 0 = 「長さ / タイプフィールド」チェックを実行しない
- bit 0 **FULLDPLX:** 全二重動作ビット  
 1 = MAC は全二重モードで動作する  
 0 = MAC は半二重モードで動作する

- Note 1:** 表 31-4 に、このレジスタの設定に基づくパディング機能を示します。  
**2:** PADENABLE ビットがクリアされている場合、このビットは無視されます。  
**3:** このビットは AUTOPAD および VLANPAD ビットと組み合わせて使います。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

表 31-4: パディング動作

タイプ	AUTOPAD	VLANPAD	PADENABLE	動作
全て	x	x	0	パディングなし、CRC チェック
全て	0	0	1	60 バイトにパディング、CRC 付加
全て	x	1	1	64 バイトにパディング、CRC 付加
全て	1	0	1	タグが付加されていない場合： 60 バイトにパディング、CRC 付加 VLAN タグが付加されている場合： 64 バイトにパディング、CRC 付加

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-25: EMAC1IPGT: Ethernet コントローラ MAC 連続パケット間ギャップレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-1	R/W-0
	—	B2BIPKTGP<6:0>						

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-7 **未実装**: 「0」として読み出し

bit 6-0 **B2BIPKTGP<6:0>**: 連続パケット間ギャップ (IPG) ビット

この設定可能フィールドは、各送信パケットの終了から次のパケットの開始までに許容される最短の期間をニブル時間のオフセットとして表します。全二重モードの場合、このレジスタには [ 必要期間 (ニブル時間) - 3] を設定します。半二重モードの場合、このレジスタには [ 必要期間 (ニブル時間) - 6] を設定します。全二重モードの推奨設定値は 0x15 (21d) です。これは 0.96  $\mu$ s (100 Mbps の場合) または 9.6  $\mu$ s (10 Mbps の場合) の最小 IPG に相当します。半二重モードの推奨設定値は 0x12 (18d) です。これは 0.96  $\mu$ s (100 Mbps の場合) または 9.6  $\mu$ s (10 Mbps の場合) の最小 IPG に相当します。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-26: EMAC1IPGR: Ethernet コントローラ MAC 非連続パケット間ギャップレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0
	—	NB2BIPKTGP1<6:0>						
7:0	U-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-1	R/W-0
	—	NB2BIPKTGP2<6:0>						

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-15 **未実装:** 「0」として読み出し

bit 14-8 **NB2BIPKTGP1<6:0>:** 非連続パケット間ギャップパート 1 ビット

この設定可能フィールドは、IEEE 80.23 仕様のセクション 4.2.3.2.1 「Deference」で言及している carrierSense ウィンドウを表します。IPGR1 のタイミング中に他のキャリアを検出すると MAC はキャリア送出手を延期します。しかし、IPGR1 時間が経過してもキャリアを検出し続ける場合、MAC は IPGR2 時間の経過後、コリジョン発生を承知の上で送信します。これによってメディアへの公平なアクセスを確保します。値のレンジは 0x0 ~ IPGR2 です。推奨値は 0xC (12d) です。

bit 7 **未実装:** 「0」として読み出し

bit 6-0 **NB2BIPKTGP2<6:0>:** 非連続パケット間ギャップパート 2 ビット

この設定可能フィールドは、非連続パケット間ギャップを表します。推奨値は 0x12 (18d) です。これは 0.96  $\mu$ s (100 Mbps の場合) または 9.6  $\mu$ s (10 Mbps の場合) の最小 IPG に相当します。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-27: EMAC1CLRT: Ethernet コントローラ MAC コリジョンウィンドウ/リトライリミットレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	R/W-1	R/W-1	R/W-0	R/W-1	R/W-1	R/W-1
	—	—	CWINDOW<5:0>					
7:0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1
	—	—	—	—	RETX<3:0>			

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-14 **未実装**: 「0」として読み出し

bit 13-8 **CWINDOW<5:0>**: コリジョン ウィンドウ ビット

この設定可能フィールドは、適切に設定されたネットワークでコリジョンが発生するスロット時間 (コリジョン ウィンドウ) を表します。コリジョン ウィンドウは送信の開始時点から始まるため、プリアンブルと SFD が含まれます。既定値の 0x37 (55d) はウィンドウ終端におけるフレーム バイトカウントに相当します。

bit 7-4 **未実装**: 「0」として読み出し

bit 3-0 **RETX<3:0>**: 最大再送信ビット

この設定可能フィールドでは、コリジョンの発生後、過度なコリジョンによってパケットを中止する前に試みる再送信の回数を指定します。規格では、試行の最大回数 (attemptLimit) を 0xF (15d) と規定しています。既定値は「0xF」です。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-28: EMAC1MAXF: Ethernet コントローラ MAC 最大フレーム長レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-1
	MACMAXF<15:8> <sup>(1)</sup>							
7:0	R/W-1	R/W-1	R/W-1	R/W-0	R/W-1	R/W-1	R/W-1	R/W-0
	MACMAXF<7:0> <sup>(1)</sup>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **MACMAXF<15:0>**: 最大フレーム長ビット<sup>(1)</sup>

これらのフィールドは 0x05EE にリセットされます。これは、最大受信フレーム長が 1518 オクテットである事を表します。タグなしの Ethernet フレームの最大サイズは 1518 オクテットです。タグ付きフレームには 4 オクテットが追加され、全長は 1522 オクテットです。最大長の制限を変更する必要がある場合、この 16 ビットフィールドで設定します。

**Note 1:** 独自ヘッダを許可している場合、このビットを必要に応じて調整する必要があります。例えば、フレームの先頭に 4 バイトのヘッダが付加されている場合、MACMAXF を 1527 オクテットに設定できます。これによって、最大 VLAN タグ付きフレーム + 4 バイトのヘッダが使えます。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-29: EMAC1SUPP: Ethernet コントローラ MAC PHY サポートレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
	—	—	—	—	RESETRMII <sup>(1)</sup>	—	—	SPEEDRMII <sup>(1)</sup>
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-12 **未実装**: 「0」として読み出し

bit 11 **RESETRMII**: RMII ロジック リセット ビット<sup>(1)</sup>

1 = MAC RMII モジュールをリセットする

0 = 通常動作

bit 10-9 **未実装**: 「0」として読み出し

bit 8 **SPEEDRMII**: RMII 速度ビット<sup>(1)</sup>

このビットは、現在の動作速度に対する RMII ロジックの速度を設定します。

1 = RMII は 100 Mbps で動作する

0 = RMII は 10 Mbps で動作する

bit 7-0 **未実装**: 「0」として読み出し

**Note 1:** このビットは RMII モジュール向けにのみ使います。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-30: EMAC1TEST: Ethernet コントローラ MAC テストレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 TESTBP	R/W-0 TESTPAUSE <sup>(1)</sup> )	R/W-0 SHRTQNTA <sup>(1)</sup> )

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                    1 = ビットはセット    0 = ビットはクリア    x = ビットは未知

bit 31-3 **未実装:** 「0」として読み出し

bit 2 **TESTBP:** バックプレッシャ テスト ビット

- 1 = MAC はリンクにバックプレッシャをアサートする  
 バックプレッシャによってプリアンブルが送信され、キャリア検出を促します。システムからの送信  
 パケットはバックプレッシャ中に送信されます。
- 0 = 通常動作

bit 1 **TESTPAUSE:** PAUSE テストビット <sup>(1)</sup>

- 1 = あたかもポーズ時間のパラメータがゼロ以外に設定されたPAUSE受信制御フレームを受信したかのよ  
 うに MAC 制御副層が送信を禁止する
- 0 = 通常動作

bit 0 **SHRTQNTA:** PAUSE 時間単位短縮ビット <sup>(1)</sup>

- 1 = MAC は実効 PAUSE 時間単位を 64 バイト時間から 1 バイト時間に短縮する
- 0 = 通常動作

**Note 1:** このビットはテスト向けにのみ使います。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が  
 可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-31: EMAC1MCFG: Ethernet コントローラ MAC MIIM コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	RESETMGMT	—	—	—	—	—	—	—
7:0	U-0	U-0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	CLKSEL<3:0> <sup>(1)</sup>				NOPRE	SCANINC

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15 **RESETMGMT**: テストリセット MIIM ビット  
 1 = MIIM (MII Management) モジュールをリセットする  
 0 = 通常動作

bit 14-6 **未実装**: 「0」として読み出し

bit 5-2 **CLKSEL<3:0>**: MIIM クロック選択 1 ビット<sup>(1)</sup>  
 これらのビットは、クロック分周ロジックが MIIM クロック (MDC) を生成するために使います。このクロックは IEEE 802.3 仕様によって 2.5 MHz 以下と定義されています。12.5 MHz の最大クロックレートをサポートする PHY もあります。

bit 1 **NOPRE**: プリアンブル抑制ビット  
 1 = MIIMは32ビットのプリアンブル フィールドなしで読み書きサイクルを実行する(抑制されたプリアンブルをサポートする PHY もあります)  
 0 = 通常の読み書きサイクルを実行する

bit 0 **SCANINC**: スキャンインクリメント ビット  
 1 = MIIM モジュールは複数の PHY にわたって読み出しサイクルを実行する  
 読み出しサイクルはアドレス 1 から開始し、EMAC1MADR レジスタの PHYADDR ビットで設定された値まで続きます。  
 0 = 同じ PHY を繰り返し読み出す

**Note 1:** 表 31-5 に、クロック分周回路のエンコードを示します。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

表 31-5: MIIM クロックの選択

MIIM クロックの選択	EMAC1MCFG<5:2>
TPBCLK5 を 4 分周する	000x
TPBCLK5 を 6 分周する	0010
TPBCLK5 を 8 分周する	0011
TPBCLK5 を 10 分周する	0100
TPBCLK5 を 14 分周する	0101
TPBCLK5 を 20 分周する	0110
TPBCLK5 を 28 分周する	0111
TPBCLK5 を 40 分周する	1000
TPBCLK5 を 48 分周する	1001
TPBCLK5 を 50 分周する	1010
未定義	その他の全ての設定

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-32: EMAC1MCMD: Ethernet コントローラ MAC MIIM コマンド レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	SCAN	READ

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット

U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-2 **未実装:** 「0」として読み出し

bit 1 **SCAN:** MIIM スキャンモード ビット

1 = MIIM モジュールは常時読み出しサイクルを実行する (例: リンク障害の監視に使用)

0 = 通常動作

bit 0 **READ:** MIIM 読み出しコマンドビット

1 = MIIM モジュールは1回の読み出しサイクルを実行する (読み出したデータはEMAC1MRDD レジスタに返します)

0 = MIIM モジュールは書き込みサイクルを実行する (書き込むデータはEMAC1MWTD レジスタから取得します)

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-33: EMAC1MADR: Ethernet コントローラ MAC MIIM アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1
	—	—	—	PHYADDR<4:0>				
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	REGADDR<4:0>				

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-13 **未実装**: 「0」として読み出し

bit 12-8 **PHYADDR<4:0>**: MIIM PHY アドレスビット

このフィールドは、マネジメント サイクルの 5 ビットの PHY アドレス フィールドを表します。  
最大 31 個の PHY のアドレスを指定できます (0 は予約済み)。

bit 7-5 **未実装**: 「0」として読み出し

bit 4-0 **REGADDR<4:0>**: MIIM レジスタアドレス ビット

このフィールドは、マネジメント サイクルの 5 ビットのレジスタアドレス フィールドを表します。最大  
32 個のレジスタにアクセスできます。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が  
可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-34: EMAC1MWTD: Ethernet コントローラ MAC MIIM 書き込みデータレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MWTD<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MWTD<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **MWTD<15:0>**: MIIM 書き込みデータビット

これらのビットに書き込むと、その 16 ビットデータと、あらかじめ EMAC1MADR レジスタに設定された PHY アドレスおよびレジスタアドレスを使って、書き込みサイクルが実行されます。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

## レジスタ 31-35: EMAC1MRDD: Ethernet コントローラ MAC MIIM 読み出しデータレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MRDD<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MRDD<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-0 **MRDD<15:0>**: MIIM 読み出しデータビット

MIIM 読み出しサイクルの実行後、16 ビットのデータをこの位置から読み出せます。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-36: EMAC1MIND: Ethernet コントローラ MAC MIIM インジケータ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	LINKFAIL	NOTVALID	SCAN	MIIMBUSY

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-4 **未実装:** 「0」として読み出し

bit 3 **LINKFAIL:** リンク障害ビット

「1」が返された場合、リンク障害が発生した事を示します。このビットは、PHY ステータス レジスタから直近に読み出された値を反映します。

bit 2 **NOTVALID:** MIIM 読み出しデータ無効ビット

「1」が返された場合、MIIM 読み出しサイクルが未完であり、読み出しデータがまだ有効ではない事を示します。

bit 1 **SCAN:** MIIM スキャンビット

「1」が返された場合、スキャン動作 (連続した MIIM 読み出しサイクル) を実行中である事を示します。

bit 0 **MIIMBUSY:** MIIM ビジービット

「1」が返された場合、MIIM モジュールが現在 MIIM 読み出しまたは書き込みサイクルを実行中である事を示します。

**Note:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-37: EMAC1SA0: Ethernet コントローラ MAC ステーションアドレス 0 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P
	STNADDR6<7:0>							
7:0	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P
	STNADDR5<7:0>							

### 凡例:

R = 読み出し可能ビット

-n = POR 時の値

W = 書き込み可能ビット

1 = ビットはセット

P = プログラミング可能ビット

U = 未実装、「0」として読み出し

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-8 **STNADDR6<7:0>**: ステーションアドレス オクテット 6 ビット

これらのビットは、6 番目に送信されたステーションアドレスのオクテットを格納します。

bit 7-0 **STNADDR5<7:0>**: ステーションアドレス オクテット 5 ビット

これらのビットは、5 番目に送信されたステーションアドレスのオクテットを格納します。

**Note 1:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

**2:** リセット時に、工場書き込みのステーションアドレスがこのレジスタに書き込まれます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-38: EMAC1SA1: Ethernet コントローラ MAC ステーションアドレス 1 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P
	STNADDR4<7:0>							
7:0	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P
	STNADDR3<7:0>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット

P = プログラミング可能ビット

U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し

bit 15-8 **STNADDR4<7:0>**: ステーションアドレス オクテット 4 ビット

これらのビットは、4 番目に送信されたステーションアドレスのオクテットを格納します。

bit 7-0 **STNADDR3<7:0>**: ステーションアドレス オクテット 3 ビット

これらのビットは、3 番目に送信されたステーションアドレスのオクテットを格納します。

**Note 1:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。

**2:** リセット時、このレジスタは工場出荷時にプログラムされたステーションアドレスから読み込まれます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 31-39: EMAC1SA2: Ethernet コントローラ MAC ステーションアドレス 2 レジスタ

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P
STNADDR2<7:0>								
7:0	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P	R/W-P
STNADDR1<7:0>								

**凡例:** P = プログラミング可能ビット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-16 **予約済み:** 常に「0」、読み出しは無視

bit 15-8 **STNADDR2<7:0>:** ステーションアドレス オクテット 2 ビット  
これらのビットは、2 番目に送信されたステーションアドレスのオクテットを格納します。

bit 7-0 **STNADDR1<7:0>:** ステーションアドレス オクテット 1 ビット  
これらのビットは、最上位の (最初に送信された) ステーションアドレスのオクテットを格納します。

**Note 1:** これらのレジスタ (SET、CLR、INV レジスタを含む) には 16 ビットアクセスと 32 ビットアクセスの両方が可能です。8 ビットアクセスはできません (ハードウェアによって無視されます)。  
**2:** リセット時に、工場書き込みのステーションアドレスがこのレジスタに書き込まれます。

# PIC32MZ グラフィック (DA) ファミリ

## 32.0 コンパレータ

**Note:** 本書はPIC32MZグラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 19. Comparator』(DS60001110)を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

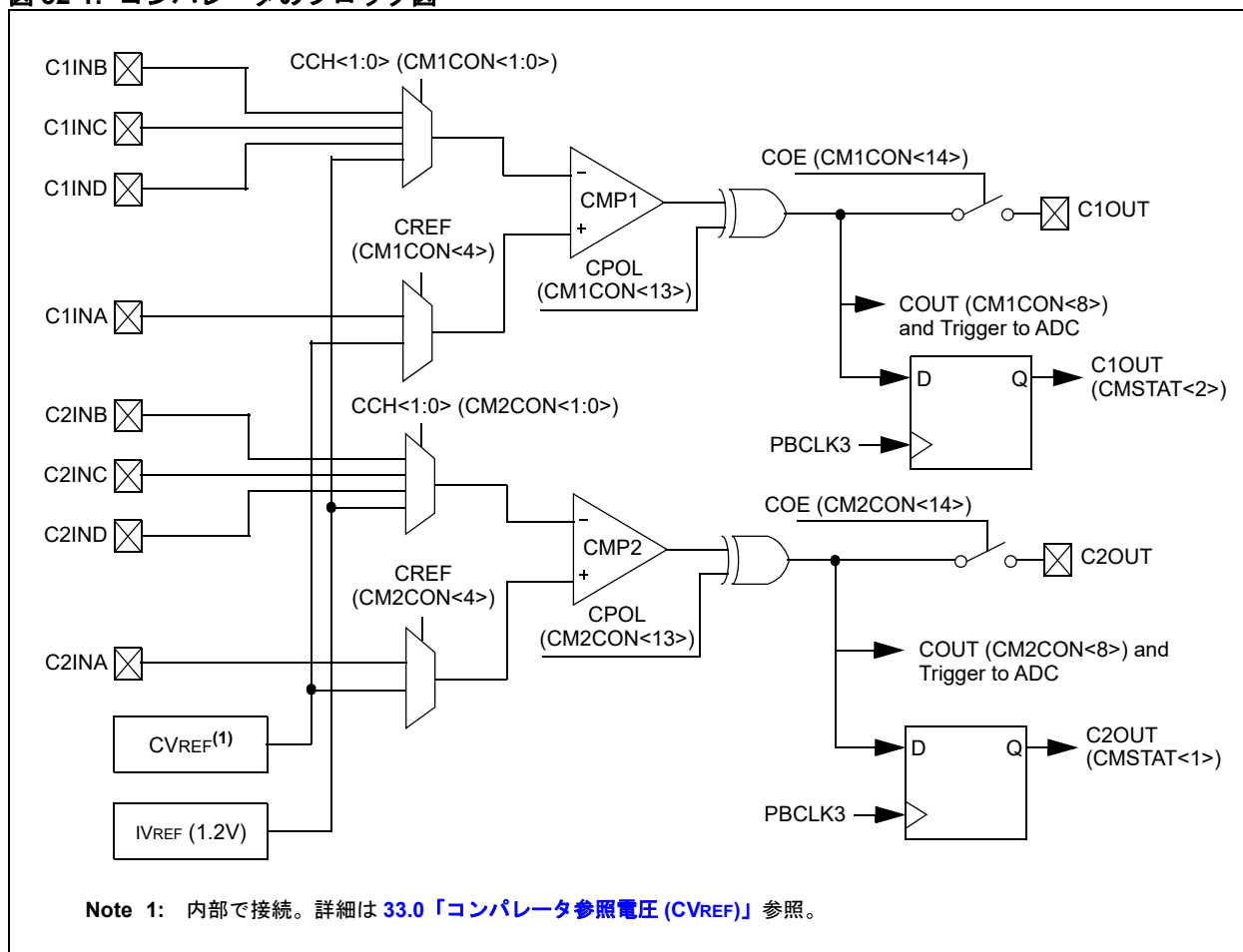
アナログ コンパレータ モジュールは、各種設定が可能な2つのコンパレータを内蔵しています。

コンパレータ モジュールの主な特長は以下の通りです。

- 差動入力
- レールツーレール動作
- 出力極性の選択
- 以下の入力の選択：
  - I/O ピンに多重化されたアナログ入力
  - デバイスの内部絶対参照電圧
  - コンパレータ参照電圧 (CVREF)
- 選択可能な割り込み生成

図 32-1 にコンパレータ モジュールのブロック図を示します。

図 32-1: コンパレータのブロック図



32.1 コンパレータ制御レジスタ

TABLE 32-1:コンパレータのレジスタマップ

仮想アドレス (BF84_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
C000	CM1CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ON	COE	CPOL	—	—	—	—	COUT	EVPOL<1:0>	—	CREF	—	—	—	CCH<1:0>	00C3	
C010	CM2CON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	ON	COE	CPOL	—	—	—	—	COUT	EVPOL<1:0>	—	CREF	—	—	—	CCH<1:0>	00C3	
C060	CMSTAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	SIDL	—	—	—	—	—	—	—	—	—	—	—	C2OUT	C1OUT	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は 12.2 「CLR、SET、INV レジスタ」を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 32-1: CMxCON: コンパレータ制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	R-0
	ON	COE	CPOL <sup>(1)</sup>	—	—	—	—	COUT
7:0	R/W-1	R/W-1	U-0	R/W-0	U-0	U-0	R/W-1	R/W-1
	EVPOL<1:0>		—	CREF	—	—	CCH<1:0>	

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-26 **未実装:** 「0」として読み出し
- bit 25-24 **未実装:** 「0」として読み出し
- bit 23-16 **未実装:** 「0」として読み出し
- bit 15    **ON:** コンパレータ ON ビット  
 1 = モジュールを有効にする (このビットをセットしても、このレジスタ内の他のビットには影響しません)  
 0 = モジュールを無効にする (このビットをクリアしても、このレジスタ内の他のビットには影響しません)
- bit 14    **COE:** コンパレータ出力イネーブルビット  
 1 = コンパレータ出力を CxOUT ピンで出力する  
 0 = コンパレータ出力を CxOUT ピンで出力しない
- bit 13    **CPOL:** コンパレータ出力反転ビット<sup>(1)</sup>  
 1 = 出力を反転する  
 0 = 出力を反転しない
- bit 12-9 **未実装:** 「0」として読み出し
- bit 8     **COUT:** コンパレータ出力ビット  
 1 = コンパレータの出力は「1」  
 0 = コンパレータの出力は「0」
- bit 7-6   **EVPOL<1:0>:** 割り込みイベント極性選択ビット  
 11 = コンパレータ出力が LOW から HIGH または HIGH から LOW に遷移した時にコンパレータ割り込みを生成する  
 10 = コンパレータ出力が HIGH から LOW に遷移した時にコンパレータ割り込みを生成する  
 01 = コンパレータ出力が LOW から HIGH に遷移した時にコンパレータ割り込みを生成する  
 00 = コンパレータ割り込みの生成を無効にする
- bit 5     **未実装:** 「0」として読み出し
- bit 4     **CREF:** コンパレータ非反転入力設定ビット  
 1 = コンパレータの非反転入力を内部 CVREF に接続する  
 0 = コンパレータの非反転入力を CxINA ピンに接続する
- bit 3-2 **未実装:** 「0」として読み出し
- bit 1-0   **CCH<1:0>:** コンパレータ反転入力選択ビット  
 11 = コンパレータの反転入力を IVREF に接続する  
 10 = コンパレータの反転入力を CxIND ピンに接続する  
 01 = コンパレータの反転入力を CxINC ピンに接続する  
 00 = コンパレータの反転入力を CxINB ピンに接続する

**Note 1:** このビットをセットすると、コンパレータ割り込みジェネレータに対する信号も反転します。これにより、割り込みは EVPOL<1:0> で選択されているのとは逆極性のエッジで生成されます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 32-2: CMSTAT: コンパレータ ステータス レジスタ

ビットレ ンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
	—	—	SIDL	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	R-0	R-0
	—	—	—	—	—	—	C2OUT	C1OUT

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-14 **未実装:** 「0」として読み出し

bit 13 **SIDL:** アイドル中停止制御ビット

1 = アイドル中は全てのコンパレータ モジュールを無効にする

0 = アイドル中も全てのコンパレータ モジュールの動作を継続する

bit 12-2 **未実装:** 「0」として読み出し

bit 1 **C2OUT:** コンパレータ 2 出力ビット

1 = コンパレータ 2 の出力は「1」

0 = コンパレータ 2 の出力は「0」

bit 0 **C1OUT:** コンパレータ 1 出力ビット

1 = コンパレータ 1 の出力は「1」

0 = コンパレータ 1 の出力は「0」



# PIC32MZ グラフィック (DA) ファミリ

## 33.0 コンパレータ参照電圧 (CVREF)

**Note:** 本書はPIC32MZグラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 20. Comparator Voltage Reference (CVREF)』(DS60001109) を参照してください。この文書は Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

CVREF モジュールは、参照電圧を選択するために使う 16 タップの抵抗ラダー回路です。このモジュールの主な用途はアナログ コンパレータに参照電圧を提供する事ですが、コンパレータとは別に使う事もできます。

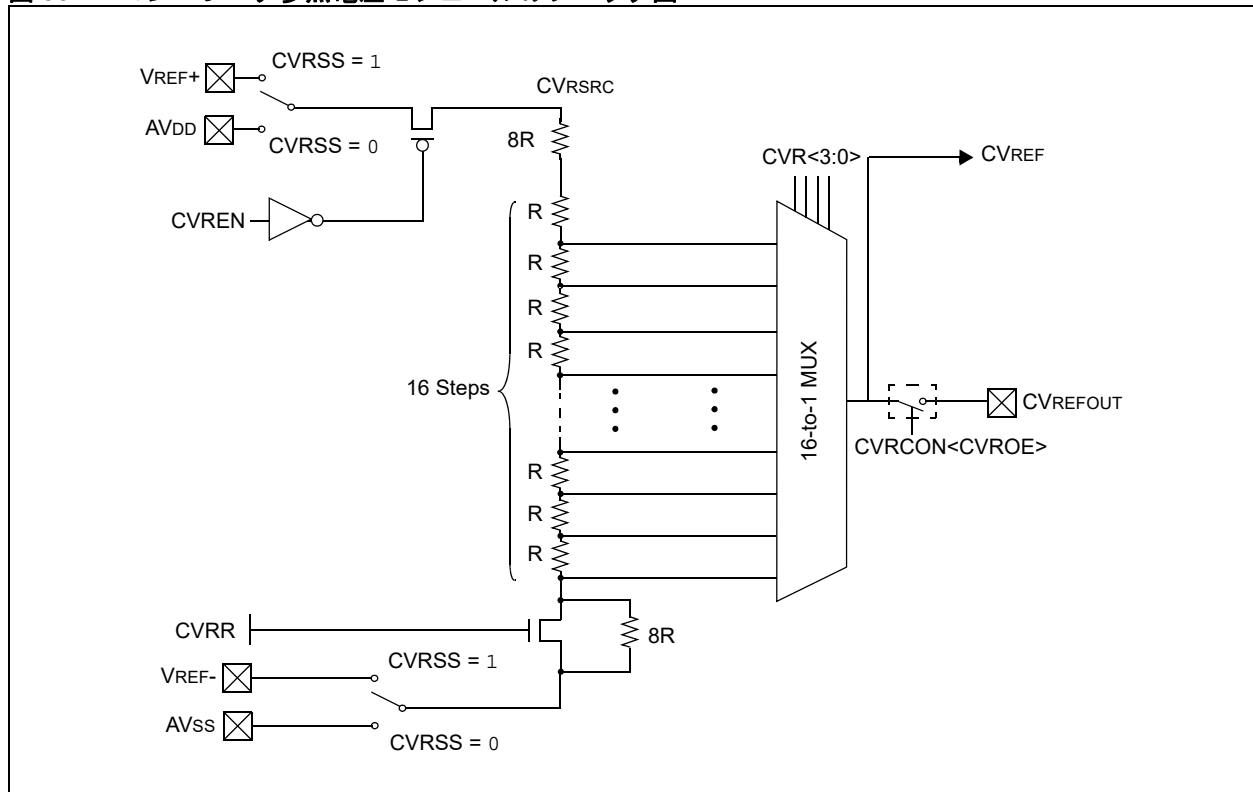
抵抗ラダーは、参照電圧の値として 2 つのレンジを提供できるように分割されています。また、未使用時の消費電力を低減するパワーダウン機能も備えています。このモジュールの電圧源には、デバイスの VDDIO/VSS または外部参照電圧のどちらかが使えます。CVREF 出力は内部でコンパレータに接続できますが、ピンに出力する事も可能です。

コンパレータ参照電圧モジュールは以下の特長を備えます。

- 高 / 低 2 つのレンジが選択可能
- 各レンジで 16 通りの電圧レベルが選択可能
- コンパレータへの内部接続によりピンを節約
- ピンに出力する事も可能

CVREF モジュールのブロック図を図 33-1 に示します。

図 33-1: コンパレータ参照電圧モジュールのブロック図



### 33.1 コンパレータ参照電圧制御レジスタ

表 33-1: コンパレータ参照電圧のレジスタマップ

仮想レジスタ (BF80_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit																全リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
0E00	CVRCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	—	—	—	—	—	—	CVROE	CVRR	CVRSS	CVR<3:0>			0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 33-1: CVRCON: コンパレータ参照電圧制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-0 ON	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	R/W-0 CVROE	R/W-0 CVRR	R/W-0 CVRSS	R/W-0	R/W-0	R/W-0	R/W-0
					CVR<3:0>			

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ON:** コンパレータ参照電圧 ON ビット

- 1 = モジュールを有効にする  
 (このビットをセットしても、レジスタ内の他のビットには影響しません)
- 0 = モジュールを無効にする (電流を消費しない)  
 (このビットをクリアしても、レジスタ内の他のビットには影響しません)

bit 14-7 **未実装:** 「0」として読み出し

bit 6 **CVROE:** CVREFOUT イネーブルビット

- 1 = モジュールの出力電圧を CVREFOUT ピンに接続する
- 0 = モジュールの出力電圧を CVREFOUT ピンから切り離す

bit 5 **CVRR:** CVREF レンジ選択ビット

- 1 = 電圧レンジ: 0 ~ 0.67 CVRSRC を選択する (電圧ステップは CVRSRC/24)
- 0 = 電圧レンジ: 0.25 CVRSRC ~ 0.75 CVRSRC を選択する (電圧ステップは CVRSRC/32)

bit 4 **CVRSS:** CVREF 電圧源選択ビット

- 1 = コンパレータ参照電圧源 CVRSRCとして (VREF+) - (VREF-) を選択する
- 0 = コンパレータ参照電圧源 CVRSRCとして AVDD - AVSS を選択する

bit 3-0 **CVR<3:0>:** CVREF 値選択ビット ( $0 \leq \text{CVR}<3:0> \leq 15$ )

CVRR = 1 の場合:  

$$\text{CVREF} = (\text{CVR}<3:0>/24) \cdot (\text{CVRSRC})$$

CVRR = 0 の場合:  

$$\text{CVREF} = 1/4 \cdot (\text{CVRSRC}) + (\text{CVR}<3:0>/32) \cdot (\text{CVRSRC})$$

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

## 34.0 HIGH/LOW 電圧検出 (HLVD)

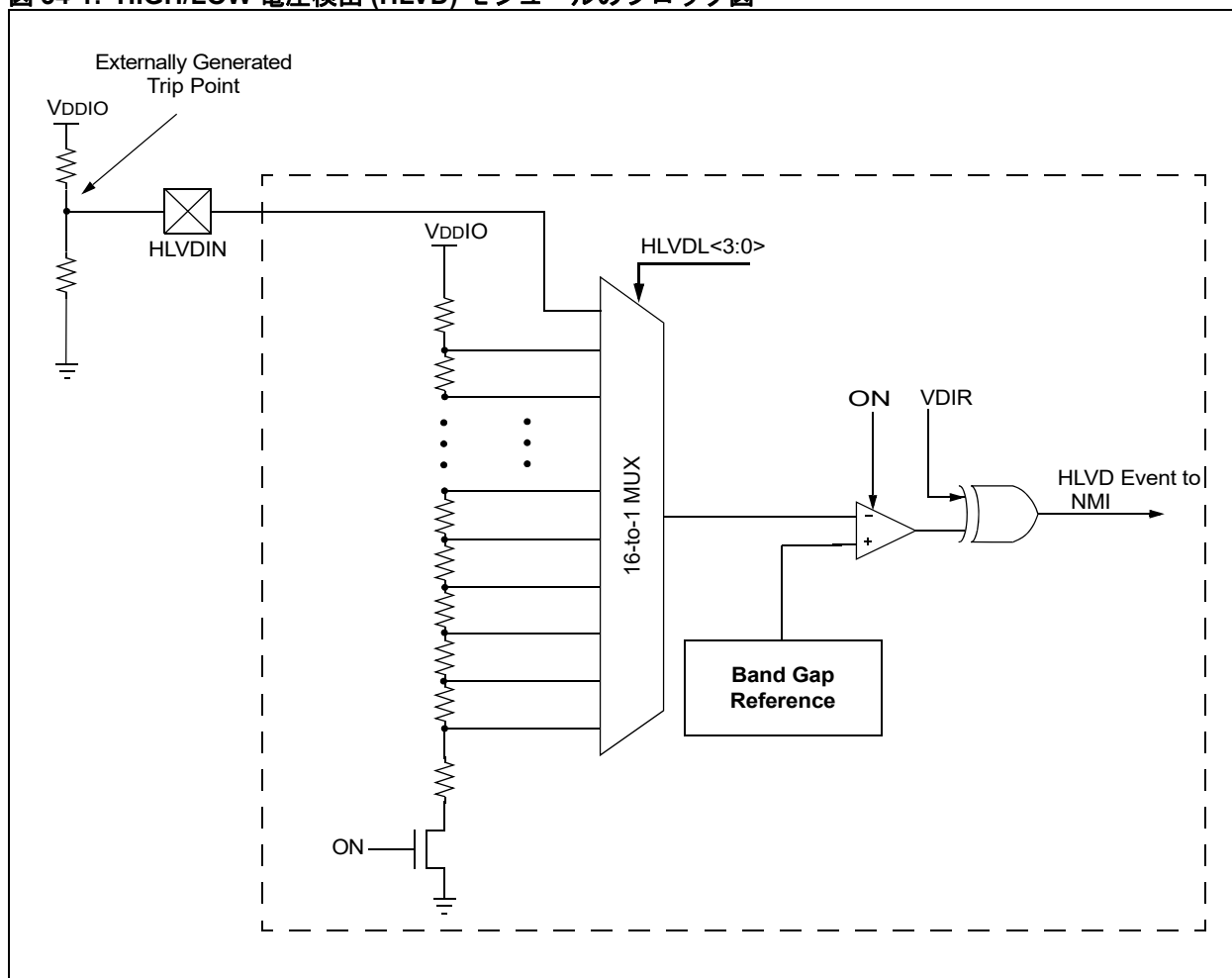
**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 38. High/Low-Voltage Detect (HLVD)』(DS60001408) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

HIGH/LOW 電圧検出モジュール (HLVD) は、デバイス電圧のトリップポイントと変化方向の両方をユーザが指定できるプログラマブルな回路です。

HLVD モジュールの特長は以下の通りです。

- ヒステリシス検出
- LOW→HIGHまたはHIGH→LOWの電圧変化の検出
- ノンマスクابل割り込み (NMI) の生成
- LVDIN ピンによる外部電圧トリップポイントの提供

図 34-1: HIGH/LOW 電圧検出 (HLVD) モジュールのブロック図



### 34.1 制御レジスタ

TABLE 34-1: HIGH/LOW 電圧検出のレジスタマップ

仮想アドレス (BF80_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														トリプル 全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1
1800	HLVDCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	ON	—	—	—	VDIR	BGVST	—	HLEVT	HLEVTOUTDIS	—	—	—	HLVDL<3:0>			0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の各レジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 34-1: HLVDCON: HIGH/LOW 電圧検出制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	R/W-0 ON	U-0 —	U-0 —	U-0 —	R/W-0 VDIR	HS,HC,R-0 BGVST	r-1 —	HS,HC,R-0 HLEVT
7:0	R/W-0 HLEVTOUTDIS <sup>(2)</sup>	U-0 —	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0
HLVDL<3:0> <sup>(1)</sup>								

<b>凡例:</b>	HS = ハードウェアでセット	HC = ハードウェアでクリア	r = 予約済みビット
	R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
	-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
			x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **ON:** HLVD モジュール イネーブルビット  
1 = HLVD モジュールを有効にする  
0 = HLVD モジュールを無効にする

bit 14-12 **未実装:** 「0」として読み出し

bit 11 **VDIR:** 電圧変化方向選択ビット  
1 = 電圧がトリップポイント (HLVDL<3:0>) 以上に上昇した時にイベントを生成する  
0 = 電圧がトリップポイント (HLVDL<3:0>) 以下に下降した時にイベントを生成する

bit 10 **BGVST:** バンドギャップ参照電圧安定ステータスビット  
1 = 内部バンドギャップ参照電圧は安定している  
0 = 内部バンドギャップ参照電圧は安定していない  
このビットは、HLVD モジュールが無効 (ON = 0) の場合に読み出せません。

bit 9 **予約済み:** 「1」として読み出し

bit 8 **HLEVT:** HIGH/LOW 電圧検出イベント ステータスビット  
1 = HLVD イベントはアクティブ  
0 = HLVD イベントは非アクティブ

bit 7 **HLEVTOUTDIS:** HIGH/LOW 電圧検出イベント出力ディセーブル ビット<sup>(2)</sup>  
1 = HLVD イベント出力を有効にする  
0 = HLVD イベント出力を無効にする

bit 6-4 **未実装:** 「0」として読み出し

**Note 1:** 誤った HLVD イベントの発生を防ぐため、全ての HLVD モジュール設定はモジュールが無効 (ON = 0) の時に変更する必要があります。実際のトリップポイント電圧については [44.0「電気的特性」](#) 内の表 44-6 を参照してください。

**2:** このビットを「1」にセットした場合、クリアするには HLVDMD ビットを使って HLVD モジュールを一度無効にしてから再度有効にする必要があります。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 34-1: HLVDCON: HIGH/LOW 電圧検出制御レジスタ

bit 3-0 HLVDL<3:0>: HIGH/LOW 電圧検出リミット選択ビット<sup>(1)</sup>

1111 = HLVDIN 上でアナログ入力を選択する  
1110 = トリップポイント 14 を選択する  
1101 = トリップポイント 13 を選択する  
1100 = トリップポイント 12 を選択する  
1011 = トリップポイント 11 を選択する  
1010 = トリップポイント 10 を選択する  
1001 = トリップポイント 9 を選択する  
1000 = トリップポイント 8 を選択する  
0111 = トリップポイント 7 を選択する  
0110 = トリップポイント 6 を選択する  
0101 = トリップポイント 5 を選択する  
0100 = トリップポイント 4 を選択する  
0011 = 予約済み (使用禁止)  
0010 = 予約済み (使用禁止)  
0001 = 予約済み (使用禁止)  
0000 = 予約済み (使用禁止)

- Note 1:** 誤った HLVD イベントの発生を防ぐため、全ての HLVD モジュール設定はモジュールが無効 (ON = 0) の時に変更する必要があります。実際のトリップポイント電圧については [44.0「電気的特性」](#) 内の [表 44-6](#) を参照してください。
- 2:** このビットを「1」にセットした場合、クリアするには HLVDMD ビットを使って HLVD モジュールを一度無効にしてから再度有効にする必要があります。



# PIC32MZ グラフィック (DA) ファミリ

## 35.0 充電時間計測ユニット (CTMU)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 37. Charge Time Measurement Unit (CTMU)』(DS60001167) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

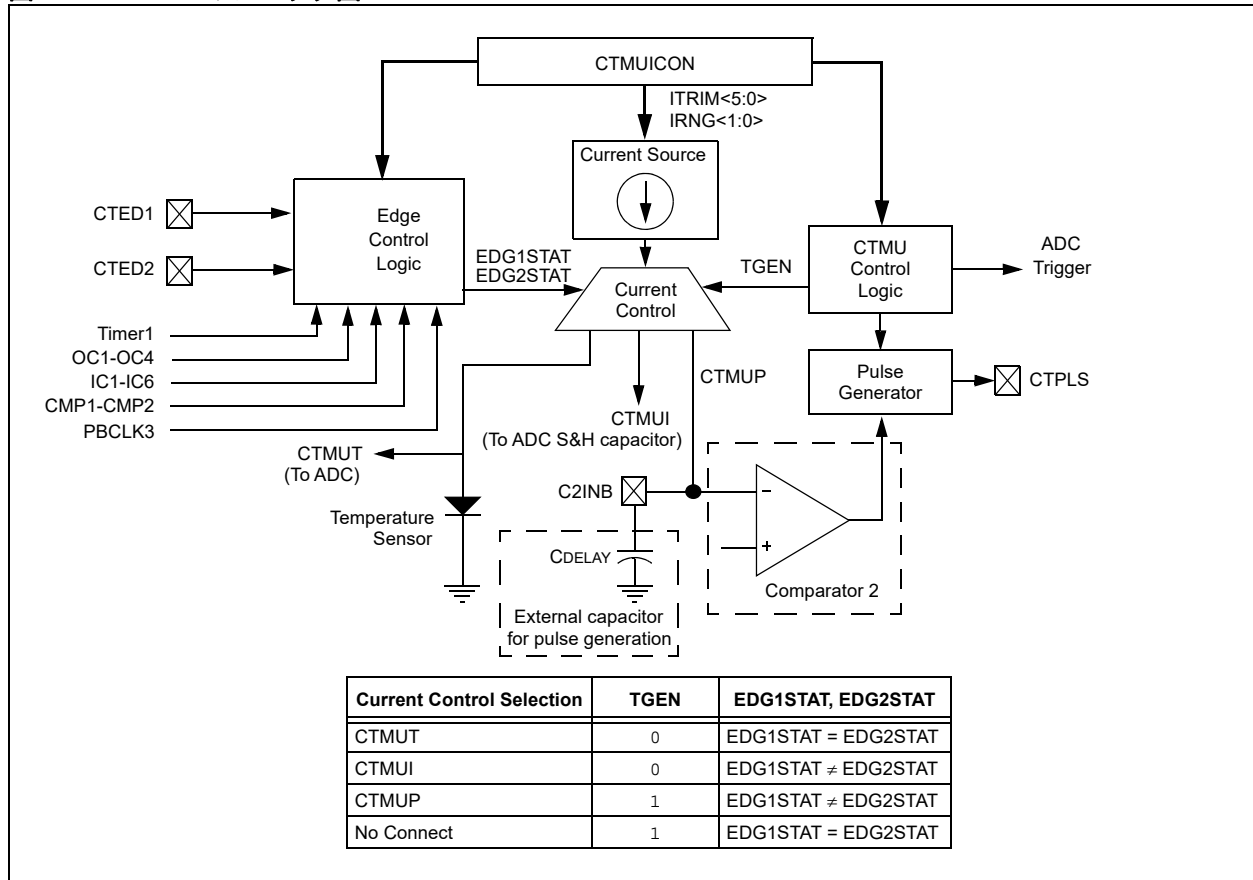
充電時間計測ユニット (CTMU) は、設定可能な電流源とデジタル設定回路で構成された柔軟なアナログモジュールです。CTMU を使うと異なるパルス源間の時間差を計測できる他、非同期パルスを生成できます。CTMU を他の内蔵アナログモジュールと組み合わせる事により、高分解能の時間計測、静電容量の計測、静電容量の相対変化の計測、一定の時間遅延を持つ出力パルスの生成が行えます。CTMU は、静電容量式センサとのインターフェイス用に理想的です。

CTMU モジュールの主な特長は以下の通りです。

- 最大 35 チャンネルの静電容量 / 時間計測入力
- 高精度電流源を内蔵
- 16 通りのエッジ入力トリガ源
- エッジ センシティブまたはレベル センシティブ入力の選択
- 各エッジ信号源の極性の制御
- エッジシーケンスの制御
- エッジに対する応答の制御
- 高精度な時間計測
- システムクロックに対して非同期の外部 / 内部信号遅延
- 温度検出ダイオードを内蔵
- 自動サンプリング中の電流源の制御
- 4 通りの電流源レンジ
- 1 ns の時間計測分解能

図 35-1 に CTMU のブロック図を示します。

図 35-1: CTMU のブロック図



### 35.1 CTMU 制御レジスタ

表 35-1: CTMU のレジスタマップ

仮想アドレス (BF84_#)	レジスタ名 <sup>(1)</sup>	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
C200	CTMUCON	31:16	EDG1MOD	EDG1POL	EDG1SEL<3:0>				EDG2STAT	EDG1STAT	EDG2MOD	EDG2POL	EDG2SEL<3:0>				—	—	0000
		15:0	ON	—	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG	ITRIM<5:0>				IRNG<1:0>			0000	

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: この表内の全てのレジスタは対応する CLR、SET、INV レジスタを持ち、それらは各レジスタの仮想アドレスからそれぞれ 0x04、0x08、0x0C だけオフセットしています。詳細は [12.2 「CLR、SET、INV レジスタ」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 35-1: CTMUCON: CTMU 制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EDG1MOD	EDG1POL	EDG1SEL<3:0>				EDG2STAT	EDG1STAT
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
	EDG2MOD	EDG2POL	EDG2SEL<3:0>				—	—
15:8	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ON	—	CTMUSIDL	TGEN <sup>(1)</sup>	EDGEN	EDGSEQEN	IDISSEN <sup>(2)</sup>	CTTRIG
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ITRIM<5:0>						IRNG<1:0>	

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31 **EDG1MOD:** エッジ1 エッジ センシティブ選択ビット

- 1 = 入力をエッジ センシティブにする
- 0 = 入力をレベル センシティブにする

bit 30 **EDG1POL:** エッジ1 極性選択ビット

- 1 = エッジ1 を立ち上がり応答に設定する
- 0 = エッジ1 を立ち下がり応答に設定する

bit 29-26 **EDG1SEL<3:0>:** エッジ1 信号源選択ビット

- 1111 = 予約済み
- 1110 = C2OUT ピンを選択する
- 1101 = C1OUT ピンを選択する
- 1100 = IC6 キャプチャ イベントを選択する
- 1011 = IC5 キャプチャ イベントを選択する
- 1010 = IC4 キャプチャ イベントを選択する
- 1001 = IC3 キャプチャ イベントを選択する
- 1000 = IC2 キャプチャ イベントを選択する
- 0111 = IC1 キャプチャ イベントを選択する
- 0110 = OC4 キャプチャ イベントを選択する
- 0101 = OC3 キャプチャ イベントを選択する
- 0100 = OC2 キャプチャ イベントを選択する
- 0011 = CTED1 ピンを選択する
- 0010 = CTED2 ピンを選択する
- 0001 = OC1 コンペアイベントを選択する
- 0000 = Timer1 イベントを選択する

bit 25 **EDG2STAT:** エッジ2 ステータスビット

- このビットはエッジ2のステータスを示します。このビットに書き込む事で、エッジ信号源を制御できます。
- 1 = エッジ2 が発生した
- 0 = エッジ2 は発生していない

**Note 1:** このビットをパルス遅延生成向けにセットした場合、EDG2SEL<2:0> ビットを「1110」(C2OUT ピンを選択)に設定する必要があります。

- 2: ADC モジュールのサンプル / ホールド コンデンサは、各サンプリング / 変換サイクルの間で自動的に放電されません。ADC を静電容量計測の一部として使うソフトウェアは、計測を行う前に ADC コンデンサを放電する必要があります。IDISSEN ビットを「1」にセットする事で、この放電を実行します。IDISSEN ビットがセットされている間 (コンデンサ アレイに放電シンクが接続されている間) は、ADC モジュールはサンプリング中である事が必要です。
- 3: 電流値については、44.0「電気的特性」内の CTMU 電流源仕様 (表 44-20) を参照してください。
- 4: この設定は、CTMU 温度計測ダイオード向けに使えません。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 35-1: CTMUCON: CTMU 制御レジスタ ( 続き )

- bit 24 **EDG1STAT:** エッジ 1 ステータスビット  
このビットはエッジ 1 のステータスを示します。このビットに書き込む事で、エッジ信号源を制御できます。  
1 = エッジ 1 が発生した  
0 = エッジ 1 は発生していない
- bit 23 **EDG2MOD:** エッジ 2 エッジ センシティブ選択ビット  
1 = 入力をエッジ センシティブにする  
0 = 入力をレベル センシティブにする
- bit 22 **EDG2POL:** エッジ 2 極性選択ビット  
1 = エッジ 2 を立ち上がり応答に設定する  
0 = エッジ 2 を立ち下がり応答に設定する
- bit 21-18 **EDG2SEL<3:0>:** エッジ 2 信号源選択ビット  
1111 = 予約済み  
1110 = C2OUT ピンを選択する  
1101 = C1OUT ピンを選択する  
1100 = PBCLK3  
1011 = IC5 キャプチャ イベントを選択する  
1010 = IC5 キャプチャ イベントを選択する  
1001 = IC5 キャプチャ イベントを選択する  
1000 = IC5 キャプチャ イベントを選択する  
0111 = IC1 キャプチャ イベントを選択する  
0110 = OC4 キャプチャ イベントを選択する  
0101 = OC3 キャプチャ イベントを選択する  
0100 = OC2 キャプチャ イベントを選択する  
0011 = CTED1 ピンを選択する  
0010 = CTED2 ピンを選択する  
0001 = OC1 コンペアイベントを選択する  
0000 = Timer1 イベントを選択する
- bit 17-16 **未実装:** 「0」 として読み出し
- bit 15 **ON:** ON イネーブルビット  
1 = モジュールを有効にする  
0 = モジュールを無効にする
- bit 14 **未実装:** 「0」 として読み出し
- bit 13 **CTMUSIDL:** アイドル中停止ビット  
1 = デバイスがアイドルに移行した時にモジュールの動作を停止する  
0 = アイドル中もモジュールの動作を継続する
- bit 12 **TGEN:** 遅延生成イネーブルビット (1)  
1 = エッジ遅延の生成を有効にする  
0 = エッジ遅延の生成を無効にする
- bit 11 **EDGEN:** エッジイネーブルビット  
1 = エッジをブロックしない  
0 = エッジをブロックする
- bit 10 **EDGSEQEN:** エッジシーケンス イネーブルビット Edge Sequence Enable bit  
1 = エッジ 1 はエッジ 2 よりも前に発生する必要がある  
0 = エッジの発生順を問わない
- bit 9 **IDISSEN:** アナログ電流源制御ビット (2)  
1 = アナログ電流源出力をグラウンドに接続する  
0 = アナログ電流源出力をグラウンドに接続しない
- Note 1:** このビットをパルス遅延生成向けにセットした場合、EDG2SEL<2:0> ビットを「1110」(C2OUT ピンを選択) に設定する必要があります。
- 2:** ADC モジュールのサンプル / ホールド コンデンサは、各サンプリング / 変換サイクルの間で自動的に放電されません。ADC を静電容量計測の一部として使うソフトウェアは、計測を行う前に ADC コンデンサを放電する必要があります。IDISSEN ビットを「1」にセットする事で、この放電を実行します。IDISSEN ビットがセットされている間 ( コンデンサ アレイに放電シンクが接続されている間 ) は、ADC モジュールはサンプリング中である事が必要です。
- 3:** 電流値については、[44.0「電気的特性」](#) 内の CTMU 電流源仕様 ( 表 44-20 ) を参照してください。
- 4:** この設定は、CTMU 温度計測ダイオード向けに使えません。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 35-1: CTMUCON: CTMU 制御レジスタ ( 続き )

- bit 8     **CTTRIG:** トリガ制御ビット  
1 = トリガ出力を有効にする  
0 = トリガ出力を無効にする
- bit 7-2   **ITRIM<5:0>:** 電流源調整ビット  
011111 = 公称電流に対する正方向の最大調整  
011110  
.  
.  
.  
000001 = 公称電流に対する正方向の最小調整  
000000 = 公称電流出力 (IRNG<1:0> で指定 )  
111111 = 公称電流に対する負方向の最小調整  
.  
.  
.  
100010  
100001 = 公称電流に対する負方向の最大調整
- bit 1-0   **IRNG<1:0>:** 電流レンジ選択ビット <sup>(3)</sup>  
11 = ベース電流の 100 倍  
10 = ベース電流の 10 倍  
01 = ベース電流レベル  
00 = ベース電流の 1000 倍 <sup>(4)</sup>

- Note 1:** このビットをパルス遅延生成向けにセットした場合、EDG2SEL<2:0> ビットを「1110」(C2OUT ピンを選択) に設定する必要があります。
- 2:** ADC モジュールのサンプル / ホールド コンデンサは、各サンプリング / 変換サイクルの間で自動的に放電されません。ADC を静電容量計測の一部として使うソフトウェアは、計測を行う前に ADC コンデンサを放電する必要があります。IDISSEN ビットを「1」にセットする事で、この放電を実行します。IDISSEN ビットがセットされている間 ( コンデンサ アレイに放電シンクが接続されている間 ) は、ADC モジュールはサンプリング中である必要があります。
- 3:** 電流値については、[44.0「電気的特性」](#) 内の CTMU 電流源仕様 ( [表 44-20](#) ) を参照してください。
- 4:** この設定は、CTMU 温度計測ダイオード向けに使えません。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

## 36.0 グラフィック LCD (GLCD) コントローラ

**Note 1:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 54. Graphics LCD Controller』(DS60001379) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

グラフィック LCD (GLCD) モジュールはディスプレイパネルと直接接続し、内蔵アナログ駆動回路を使って画面内の各画素を制御します。

GLCD コントローラは、ディスプレイ データをメモリデバイスから転送し、ディスプレイ デバイスに対応したフォーマットに変換します。メモリには内部 RAM または DDR2 が使えます。

各ピンのパラレルインターフェイスは標準の 3.3 V 出力で動作します。24 ビットカラー向けに 28 ピンが必要です。これらのピンは汎用 I/O 機能と共有されます。

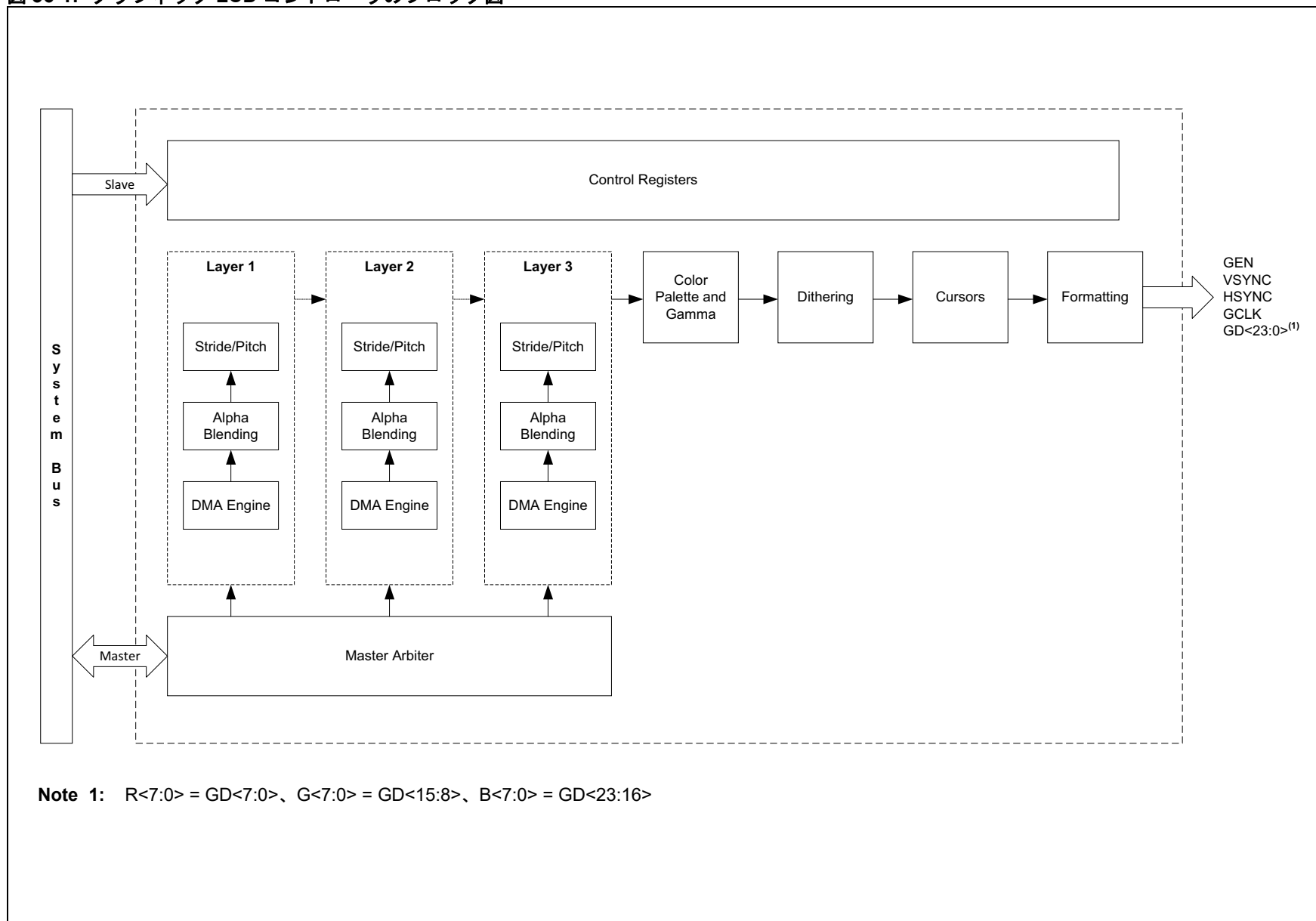
GLCD コントローラの主な特長は以下の通りです。

- 最大 50 MHz の画素クロック
- 最大 800x480 (WVGA)、オーバーレイあり (アプリケーションに依存)
- 最大 1280x1024 (SXGA) の静止画像
- 色深度 : 8、16<sup>(1)</sup>、18、24 ビット
- 最大 3 回路のタイミングレイヤ (各レイヤは以下を含む):
  - 設定可能アルファ ブレンディング
  - 設定可能ストライド / ピッチ
- 入力フォーマット : RGBA8888、ARGB8888、RGB888、RGB565、RGBA5551、YUYV、RGB332、LUT8、グレースケール
- 出力フォーマット : RGB888、RGB666、BT.656
- 18 ビット ディスプレイ向けディザ処理
- 高品質 YUV 変換
- 256 色をサポートする CLUT (Global color palette look-up table)
- グローバルなガンマ補正 / 輝度 / コントラストをサポート
- 設定可能なカーソル (16 色をサポート)
- 設定可能な HSYNC、VSYNC、DE、PCLK 極性
- 内蔵 DMA による CPU 負荷の軽減
- 設定可能な HSYNC および VSYNC 割り込み (エッジトリガとレベルトリガを選択可能)

**Note 1:** 16 ビット色深度は、GLCDMODE ビット (CFGCON2<30>) によりサポートされません。このビットをセットした場合、GD0/GD1/GD2/GD8/GD9/GD16/GD17/GD18 は汎用 I/O として使えます。

GLCD コントローラ インターフェイスのブロック図を [図 36-1](#) に示します。

図 36-1: グラフィック LCD コントローラのブロック図





## 36.1 グラフィック LCD コントローラ制御レジスタ

表 36-1: グラフィック LCD コントローラのレジスタマップ

仮アドレス (BF8E_#)	レジスタ名	ビットレンジ	Bit 31/15	Bit 30/14	Bit 29/13	Bit 28/12	Bit 27/11	Bit 26/10	Bit 25/9	Bit 24/8	Bit 23/7	Bit 22/6	Bit 21/5	Bit 20/4	Bit 19/3	Bit 118/2	Bit 17/1	Bit 16/0	全リセット
A000	GLCD MODEMode 1	31:16	LCDEN	CURSOR EN	—	VSYNC POL	HSYNC POL	DEPOL	—	DITHER	VSYNC CYC	PCLK POL	—	PGRAMP EN	FORCE BLANK	—	—	—	0000
		15:0	—	—	—	—	—	—	YUV OUTPUT	FORMAT CLK	RGBSEQ<2:0>			—	—	—	—	—	0000
A004	GLCD CLKCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0401
		15:0	LPREFETCH<5:0>									—	—	CLKDIV<5:0>					0000
A008	GLCD BGCOLOR	31:16	RED<7:0>							GREEN<7:0>							0000		
		15:0	BLUE<7:0>							ALPHA<7:0>							0000		
A00C	GLCDRES	31:16	—	—	—	—	RESX<10:0>										0000		
		15:0	—	—	—	—	RESY<10:0>										0000		
A014	GLCD FPORCH	31:16	—	—	—	—	FPORCHX<10:0>										0000		
		15:0	—	—	—	—	FPORCHY<10:0>										0000		
A018	GLCD BLANKING	31:16	—	—	—	—	BLANKINGX<10:0>										0000		
		15:0	—	—	—	—	BLANKINGY<10:0>										0000		
A01C	GLCD BPORCH	31:16	—	—	—	—	BPORCHX<10:0>										0000		
		15:0	—	—	—	—	BPORCHY<10:0>										0000		
A020	GLCD CURSOR	31:16	—	—	—	—	CURSORX<10:0>										0000		
		15:0	—	—	—	—	CURSORY<10:0>										0000		
A030	GLCD LOMODE	31:16	LAYEREN	DISA BIFIL	FORCE ALPHA	MUL ALPHA	—	—	—	—	ALPHA<7:0>							0000	
		15:0	DESTBLEND<3:0>				SRCBLEND<3:0>				—	—	—	—	COLORMODE<3:0>				0000
A034	GLCD LOSTART	31:16	—	—	—	—	STARTX<10:0>										0000		
		15:0	—	—	—	—	STARTY<10:0>										0000		
A038	GLCD LOSIZE	31:16	—	—	—	—	SIZEX<10:0>										0000		
		15:0	—	—	—	—	SIZEY<10:0>										0000		
A03C	GLCD LOBADDR	31:16	BASEADDR<31:16>																0000
		15:0	BASEADDR<15:0>																0000
A040	GLCD LOSTRIDE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	STRIDE<15:0>																0000
A044	GLCD LORES	31:16	—	—	—	—	RESX<10:0>										0000		
		15:0	—	—	—	—	RESY<10:0>										0000		
A050	GLCD L1MODE	31:16	LAYEREN	DISA BIFIL	FORCE ALPHA	MUL ALPHA	—	—	—	—	ALPHA<7:0>							0000	
		15:0	DESTBLEND<3:0>				SRCBLEND<3:0>				—	—	—	—	COLORMODE<3:0>				0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: PIXELxy ビットの x のレンジは 0 ~ 31、y のレンジは 0 ~ 31 です (例: GLCDCURDATA0 は PIXEL00 ~ PIXEL07 を格納し、PIXEL00 が最上位ニブルです)。

表 36-1: グラフィック LCD コントローラのレジスタマップ (続き)

仮想アドレス (BF8E #)	レジスタ名	ビットレンジ	Bit 31/15	Bit 30/14	Bit 29/13	Bit 28/12	Bit 27/11	Bit 26/10	Bit 25/9	Bit 24/8	Bit 23/7	Bit 22/6	Bit 21/5	Bit 20/4	Bit 19/3	Bit 118/2	Bit 17/1	Bit 16/0	リセット
A054	GLCD L1START	31:16	—	—	—	—	STARTX<10:0>											0000	
		15:0	—	—	—	—	STARTY<10:0>											0000	
A058	GLCD L1SIZE	31:16	—	—	—	—	SIZEX<10:0>											0000	
		15:0	—	—	—	—	SIZEY<10:0>											0000	
A05C	GLCD L1BADDR	31:16	BASEADDR<31:16>																0000
		15:0	BASEADDR<15:0>																0000
A060	GLCD L1STRIDE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	STRIDE<15:0>																0000
A064	GLCD L1RES	31:16	—	—	—	—	RESX<10:0>											0000	
		15:0	—	—	—	—	RESY<10:0>											0000	
A070	GLCD L2MODE	31:16	LAYEREN	DISA BIFIL	FORCE ALPHA	MUL ALPHA	—	—	—	—	ALPHA<7:0>							0000	
		15:0	DESTBLEND<3:0>				SRCBLEND<3:0>				—	—	—	—	COLORMODE<3:0>				0000
A074	GLCD L2START	31:16	—	—	—	—	STARTX<10:0>											0000	
		15:0	—	—	—	—	STARTY<10:0>											0000	
A078	GLCD L2SIZE	31:16	—	—	—	—	SIZEX<10:0>											0000	
		15:0	—	—	—	—	SIZEY<10:0>											0000	
A07C	GLCD L2BADDR	31:16	BASEADDR<31:16>																0000
		15:0	BASEADDR<15:0>																0000
A080	GLCD L2STRIDE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	STRIDE<15:0>																0000
A084	GLCDL2RES	31:16	—	—	—	—	RESX<10:0>											0000	
		15:0	—	—	—	—	RESY<10:0>											0000	
A0F8	GLCDINT	31:16	IRQCON	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	HSYNCINT	VSYNCCINT
A0FC	GLCDSTAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
A400 ~ A7FC	GLCDCLUTx ('x' = 0-255)	31:16	—	—	—	—	—	—	—	—	—	RED<7:0>							0000
		15:0	GREEN<7:0>								BLUE<7:0>								0000
A800 through A9FC	GLCD CURDATAx ('x' = 0-127)	31:16	PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>				0000
		15:0	PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>				0000
AA00 ~ AA40	GLCD CURLUTx ('x' = 0-15)	31:16	—	—	—	—	—	—	—	—	RED<7:0>							0000	
		15:0	GREEN<7:0>								BLUE<7:0>								0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記  
 Note 1: PIXELxy ビットの x のレンジは 0 ~ 31、y のレンジは 0 ~ 31 です (例: GLCDCURDATA0 は PIXEL00 ~ PIXEL07 を格納し、PIXEL00 が最上位ニブルです)。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-1: GLCDMODE: グラフィック LCD コントローラ モードレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
	LCDEN	CURSOR EN	—	VSYNC POL	HSYNC POL	DEPOL	—	DITHER
23:16	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	U-0
	VSYNC CYC	PCLKPOL	—	PGRAMP EN	FORCE BLANK	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	YUV OUTPUT	FORMAT CLK
7:0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
	RGBSEQ<2:0>			—	—	—	—	—

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

- bit 31    **LCDEN:** LCD コントローラ モジュール イネーブルビット  
 1 = LCD コントローラ モジュールを有効にする  
 0 = LCD コントローラ モジュールを有効にしない
- bit 30    **CURSOREN:** プログラマブル カーソル イネーブルビット  
 1 = プログラマブル カーソルを有効にする  
 0 = プログラマブル カーソルを無効にする
- bit 29    **未実装:** 「0」として読み出し
- bit 28    **VSYNCPOL:** 垂直同期極性ビット  
 1 = VSYNC 極性を反転する  
 0 = VSYNC 極性を反転しない
- bit 27    **HSYNCPOL:** 水平同期極性ビット  
 1 = HSYNC 極性を反転する  
 0 = HSYNC 極性を反転しない
- bit 26    **DEPOL:** DE 極性ビット  
 1 = DE 極性を反転する  
 0 = DE 極性を反転しない
- bit 25    **未実装:** 「0」として読み出し
- bit 24    **DITHER:** ディザ処理イネーブルビット  
 1 = ディザ処理を有効にする  
 0 = ディザ処理を有効にしない
- bit 23    **VSYNCCYC:** Single Cycle Per Line 向け垂直同期イネーブルビット  
 1 = Single Cycle Per Line に対して VSYNC を有効にする  
 0 = Single Cycle Per Line に対して VSYNC を有効にしない
- bit 22    **PCLKPOL:** 画素クロック出力極性ビット  
 1 = 画素クロック出力極性を反転する  
 0 = 画素クロック出力極性を反転しない
- bit 21    **未実装:** 「0」として読み出し
- bit 20    **PGRAMPEN:** パレット ガンマランプ イネーブルビット  
 1 = パレット ガンマランプを有効にする  
 0 = パレット ガンマランプを有効にしない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 36-1: GLCDMODE: グラフィック LCD コントローラ モードレジスタ (続き)

- bit 19 **FORCEBLANK:** ブランク出力強制ビット  
1 = 出力をブランクにする  
0 = 何もしない
- bit 18-10 **未実装:** 「0」として読み出し
- bit 9 **YUVOUTPUT:** YUV 出力イネーブルビット  
1 = YUV を有効にする  
0 = RGB を有効にする
- bit 8 **FORMATCLK:** フォーマット クロック分周イネーブルビット  
1 = フォーマット クロックを分周しない  
0 = フォーマット クロックを分周する
- bit 7-5 **RGBSEQ<2:0>:** RGB シーケンシャル モードビット  
111 = BT.656  
110 = YUYV  
101 = 予約済み  
100 = 予約済み  
011 = 予約済み  
010 = 予約済み  
001 = 予約済み  
000 = パラレル RGB (RGB888、RGB666、RGB332)
- bit 4-0 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-2: GLCDCLKCON: グラフィック LCD コントローラ クロック制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	LPREFETCH<5:0>					
7:0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	CLKDIV<5:0>					

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-14 **未実装:** 「0」として読み出し

bit 13-8 **LPREFETCH<5:0>:** ラインプリフェッチビット

これらのビットは、フレーム (DMA 経由) の開始前にプリフェッチされるラインの数を表します。最大値は  $2^{LPREFETCH} = 32$  です。

bit 7-6 **未実装:** 「0」として読み出し

bit 5-0 **CLKDIV<5:0>:** クロック分周比ビット

111111 = 予約済み

111110 = 予約済み

•

•

•

011111 = 31 分周する

011110 = 30 分周する

011101 = 29 分周する

•

•

•

000011 = 3 分周する

000010 = 2 分周する

000001 = 1 分周する

000000 = 0 分周する

**Note:** CLKDIV<5:0> の値が偶数である場合、GCLK = (REFCLKO5/CLKDIV)、デューティサイクル = 50% です。CLKDIV<5:0> の値が奇数である場合、PCLK = (REFCLKO5/CLKDIV)、デューティサイクル = 60 ~ 40% です。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-3: GLCDBGCOLOR: グラフィック LCD コントローラ背景色レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RED<7:0>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	GREEN<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BLUE<7:0>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ALPHA<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-24 **RED<7:0>**: 背景色赤ビット  
 これらのビットは、背景色に使う赤の値を指定します。
- bit 23-16 **GREEN<7:0>**: 背景色緑ビット  
 これらのビットは、背景色に使う緑の値を指定します。
- bit 15-8 **BLUE<7:0>**: 背景色青ビット  
 これらのビットは、背景色に使う青の値を指定します。
- bit 7-0 **ALPHA<7:0>**: 背景色アルファビット  
 これらのビットは、背景色に使うアルファ値を指定します。

**Note:** このレジスタ内の全てのビット (RED、GREEN、BLUE、ALPHA) を設定した場合、RGBA 色が背景色として使われます。

レジスタ 36-4: GLCDRES: グラフィック LCD コントローラ分解能レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	RESX<10:8>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RESX<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RESY<10:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RESY<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-27 **未実装**: 「0」として読み出し
- bit 26-16 **RESX<10:0>**: X 座標画素分解能ビット  
 これらのビットは、X 座標の画素分解能を指定します。
- bit 15-11 **未実装**: 「0」として読み出し
- bit 10-0 **RESY<10:0>**: Y 座標画素分解能ビット  
 これらのビットは、Y 座標の画素分解能を指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-5: GLCDFPORCH: グラフィック LCD コントローラ フロントポーチ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	FPORCHX<10:8>		
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FPORCHX<7:0>							
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	FPORCHY<10:8>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FPORCHY<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-27 **未実装**: 「0」として読み出し
- bit 26-16 **FPORCHX<10:0>**: X 座標フロントポーチ ライン数ビット  
 これらのビットは、X 座標のフロントポーチライン数を指定します。
- bit 15-11 **未実装**: 「0」として読み出し
- bit 10-0 **FPORCHY<10:0>**: Y 座標フロントポーチ画素クロック数ビット  
 これらのビットは、Y 座標のフロントポーチ画素クロック数を指定します。

レジスタ 36-6: GLCDBLANKING: グラフィック LCD コントローラ ブランキング レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	BLANKINGX<10:8>		
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BLANKINGX<7:0>							
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	BLANKINGY<10:8>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BLANKINGY<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-27 **未実装**: 「0」として読み出し
- bit 26-16 **BLANKINGX<10:0>**: X 座標ブランキング期間ビット  
 これらのビットは、X 座標ブランキング期間の HSYNC パルス長を指定します。
- bit 15-11 **未実装**: 「0」として読み出し
- bit 10-0 **BLANKINGY<10:0>**: Y 座標ブランキング期間ビット  
 これらのビットは、Y 座標ブランキング期間の VSYNC ライン数を指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-7: GLCDBPORCH: グラフィック LCD コントローラ バックポーチ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	BPORCHX<10:8>		
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BPORCHX<7:0>							
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	BPORCHY<10:8>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BPORCHY<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-27 **未実装:** 「0」として読み出し

bit 26-16 **BPORCHX<10:0>:** X座標バックポーチライン数ビット  
 これらのビットは、X座標のバックポーチライン数を指定します。

bit 15-11 **未実装:** 「0」として読み出し

bit 10-0 **BPORCHY<10:0>:** Y座標バックポーチ画素クロック数ビット  
 これらのビットは、Y座標のバックポーチ画素クロック数を指定します。

レジスタ 36-8: GLCDCCURSOR: グラフィック LCD コントローラ カーソルレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	CURSORX<10:8>		
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CURSORX<7:0>							
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	CURSORY<10:8>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CURSORY<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-27 **未実装:** 「0」として読み出し

bit 26-16 **CURSORX<10:0>:** X座標カーソル位置ビット  
 これらのビットは、カーソルのX座標位置を指定します。

bit 15-11 **未実装:** 「0」として読み出し

bit 10-0 **CURSORY<10:0>:** Y座標カーソル位置ビット  
 これらのビットは、カーソルのY座標位置を指定します。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-9: GLCDLxMODE: グラフィック LCD コントローラ レイヤ「x」モードレジスタ (x = 0 ~ 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
	LAYEREN	DISABIFIL	FORCE ALPHA	MUL ALPHA	—	—	—	—
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ALPHA<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	DESTBLEND<3:0>				SRCBLEND<3:0>			
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	COLORMODE<3:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31    **LAYEREN:** レイヤイネーブルビット  
 1 = レイヤを有効にする  
 0 = レイヤを無効にする
- bit 30    **DISABIFIL:** バイリニア フィルタ ディセーブル ビット  
 1 = バイリニア フィルタ処理を有効にする  
 0 = バイリニア フィルタ処理を有効にしない
- bit 29    **FORCEALPHA:** Force Alpha with Global Alpha ビット  
 1 = Force alpha with global alpha を有効にする  
 0 = Force alpha with global alpha を有効にしない
- bit 28    **MULALPHA:** プリマルチプライ イメージアルファ ビット  
 1 = プリマルチプライ イメージアルファを有効にする  
 0 = プリマルチプライ イメージアルファを無効にする
- bit 27-24 **未実装:** 「0」として読み出し
- bit 23-16 **ALPHA<7:0>:** レイヤアルファ ビット  
 これらのビットは、レイヤアルファ値 (0 ~ 0xFF) を格納します。
- bit 15-12 **DESTBLEND<3:0>:** デスティネーション ブレンド関数ビット  
 1111 = 予約済み  
 1110 = 予約済み  
 1101 = 反転したデスティネーションをブレンドする  
 1100 = 予約済み  
 1011 = 予約済み  
 1010 = アルファ デスティネーションをブレンドする  
 1001 = 予約済み  
 1000 = 予約済み  
 0111 = 反転したソースと反転したグローバルをブレンドする  
 0110 = 反転したグローバルをブレンドする  
 0101 = 反転したソースをブレンドする  
 0100 = アルファソースとアルファ グローバルをブレンドする  
 0011 = アルファ グローバルをブレンドする  
 0010 = アルファソースをブレンドする  
 0001 = 白をブレンドする  
 0000 = 黒をブレンドする

# PIC32MZ グラフィック (DA) ファミリ

---

レジスタ 36-9: GLCDLxMODE: グラフィック LCD コントローラ レイヤ「x」モードレジスタ  
(x = 0 ~ 2) ( 続き )

bit 11-8 **SRCBLEND<3:0>**: ソースブレンド関数ビット

1111 = 予約済み  
1110 = 予約済み  
1101 = 反転したデスティネーションをブレンドする  
1100 = 予約済み  
1011 = 予約済み  
1010 = アルファ デスティネーションをブレンドする  
1001 = 予約済み  
1000 = 予約済み  
0111 = 反転したソースと反転したグローバルをブレンドする  
0110 = 反転したグローバルをブレンドする  
0101 = 反転したソースをブレンドする  
0100 = アルファソースとアルファ グローバルをブレンドする  
0011 = アルファ グローバルをブレンドする  
0010 = アルファソースをブレンドする  
0001 = 白をブレンドする  
0000 = 黒をブレンドする

bit 7-4 **未実装**: 「0」として読み出し

bit 3-0 **COLORMODE<3:0>**: カラーモードビット

1111 = 予約済み  
1110 = 予約済み  
1101 = 予約済み  
1100 = 予約済み  
1011 = RGB888 カラー フォーマット  
1010 = YUYV カラー フォーマット  
1001 = L4 グレースケール/パレット フォーマット  
1000 = L1 グレースケール/パレット フォーマット  
0111 = L8 グレースケール/パレット フォーマット  
0110 = 32 ビット ARGB8888 カラー フォーマット  
0101 = 16 ビット RGB565 カラー フォーマット  
0100 = 8 ビット RGB332 カラー フォーマット  
0011 = 予約済み  
0010 = 32 ビット RGBA8888 カラー フォーマット  
0001 = 16 ビット RGBA5551 カラー フォーマット  
0000 = 8 ビット カラーパレット ルックアップ テーブル (LUT8)

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-10: GLCDLxSTART: グラフィック LCD コントローラ レイヤ「x」開始レジスタ (x = 0 ~ 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	STARTX<10:8>		
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	STARTX<7:0>							
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	STARTY<10:8>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	STARTY<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-27 **未実装**: 「0」として読み出し

bit 26-16 **STARTX<10:0>**: X 座標レイヤ開始位置ビット  
 これらのビットは、レイヤの X 座標開始位置の画素オフセットを指定します。

bit 15-11 **未実装**: 「0」として読み出し

bit 10-0 **STARTY<10:0>**: Y 座標レイヤ開始位置ビット  
 これらのビットは、レイヤの Y 座標開始位置の画素オフセットを指定します。

レジスタ 36-11: GLCDLxSIZE: グラフィック LCD コントローラ レイヤ「x」サイズレジスタ (x = 0 ~ 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	SIZEX<10:8>		
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SIZEX<7:0>							
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	SIZEY<10:8>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SIZEY<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-27 **未実装**: 「0」として読み出し

bit 26-16 **SIZEX<10:0>**: X 座標レイヤサイズ ビット  
 これらのビットは、レイヤの X 座標画素サイズを指定します。

bit 15-11 **未実装**: 「0」として読み出し

bit 10-0 **SIZEY<10:0>**: Y 座標レイヤサイズ ビット  
 これらのビットは、レイヤの Y 座標画素サイズを指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-12: GLCDLxBADDR: グラフィック LCD コントローラ レイヤ「x」ベースアドレス  
レジスタ (x = 0 ~ 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BASEADDR<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BASEADDR<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BASEADDR<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BASEADDR<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **BASEADDR<31:0>**: フレームバッファ ベースアドレス ビット  
これらのビットは、フレームバッファのベースアドレスを指定します。

レジスタ 36-13: GLCDLxSTRIDE: グラフィック LCD コントローラ レイヤ「x」ストライド レジスタ (x = 0 ~ 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	STRIDE<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	STRIDE<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値              1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装**: 「0」として読み出し  
bit 15-0 **STRIDE<15:0>**: レイヤストライド ビット  
これらのビットは、ラインの間隔をバイト数で指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-14: GLCDLxRES: グラフィック LCD コントローラ レイヤ「x」分解能レジスタ  
(x = 0 ~ 2)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	RESX<10:8>		
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RESX<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	RESY<10:8>		
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RESY<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-27 **未実装**: 「0」として読み出し

bit 26-16 **RESX<10:0>**: X 座標レイヤ画素分解能ビット

これらのビットは、レイヤの X 座標画素分解能を指定します。

bit 15-11 **未実装**: 「0」として読み出し

bit 10-0 **RESY<10:0>**: Y 座標レイヤ画素分解能ビット

これらのビットは、レイヤの Y 座標画素分解能を指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-15: GLCDINT: グラフィック LCD コントローラ割り込みレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	IRQCON	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	HSYNCINT	VSYNCINT

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31    **IRQCON:** IRQ トリガ制御ビット

- 1 = エッジトリガを有効にする
- 0 = レベルトリガを有効にする

bit 30-2    **未実装:** 「0」として読み出し

bit 1    **HSYNNCINT:** HSYNC 割り込みイネーブルビット

- 1 = HSYNC 割り込みを有効にする
- 0 = HSYNC 割り込みを有効にしない

bit 0    **VSYNCINT:** VSYNC 割り込みイネーブルビット

- 1 = VSYNC 割り込みを有効にする
- 0 = VSYNC 割り込みを有効にしない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-16: GLCDSTAT: グラフィック LCD コントローラ ステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	R-0	U-0	R-0	R-0	R-0	R-0
	—	—	LROW	—	VSYNC	HSYNC	DE	ACTIVE

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

bit 31-6 **未実装:** 「0」として読み出し

bit 5 **LROW:** 最終行ビット  
 1 = 最終行が現在表示されている  
 0 = 最終行は現在表示されていない

bit 4 **未実装:** 「0」として読み出し

bit 3 **VSYNC:** VSYNC 信号レベルビット  
 このビットは、VSYNC 信号レベルを返します。

bit 2 **HSYNC:** HSYNC 信号レベルビット  
 このビットは、HSYNC 信号レベルを返します。

bit 1 **DE:** DE 信号レベルビット  
 このビットは、DE 信号レベルを返します。

bit 0 **ACTIVE:** アクティブビット  
 1 = LCD コントローラは、アクティブな垂直ブランキング中ではない  
 0 = LCD コントローラはアクティブな垂直ブランキング中である

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-17: GLCDCLUTx: グラフィック LCD コントローラ グローバル色変換ルックアップ テーブル レジスタ x (x = 0 ~ 255)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	RED<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	GREEN<7:0>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BLUE<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-24 **未実装**: 「0」として読み出し

bit 23-16 **RED<7:0>**: グローバル色変換ルックアップ テーブル R(赤)成分ビット

bit 15-8 **GREEN<7:0>**: グローバル色変換ルックアップ テーブル G(緑)成分ビット

bit 7-0 **BLUE<7:0>**: グローバル色変換ルックアップ テーブル B(青)成分ビット



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-18: GLCDCURDATAx: グラフィック LCD コントローラ カーソルデータ 「n」 レジスタ (n = 0 ~ 127)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>			
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>			
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>			
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PIXELxy<3:0> <sup>(1)</sup>				PIXELxy<3:0> <sup>(1)</sup>			

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-28 **PIXELxy<3:0>**: 画素「xy」色変換ルックアップビット<sup>(1)</sup>

bit 27-24 **PIXELxy<3:0>**: 画素「xy」色変換ルックアップビット<sup>(1)</sup>

bit 23-20 **PIXELxy<3:0>**: 画素「xy」色変換ルックアップビット<sup>(1)</sup>

bit 19-16 **PIXELxy<3:0>**: 画素「xy」色変換ルックアップビット<sup>(1)</sup>

bit 15-12 **PIXELxy<3:0>**: 画素「xy」色変換ルックアップビット<sup>(1)</sup>

bit 11-8 **PIXELxy<3:0>**: 画素「xy」色変換ルックアップビット<sup>(1)</sup>

bit 7-4 **PIXELxy<3:0>**: 画素「xy」色変換ルックアップビット<sup>(1)</sup>

bit 3-0 **PIXELxy<3:0>**: 画素「xy」色変換ルックアップビット<sup>(1)</sup>

**Note 1:** PIXELxy ビットの x のレンジは 0 ~ 31、y のレンジは 0 ~ 31 です (例: GLCDCURDATA0 は PIXEL00 ~ PIXEL07 を格納し、PIXEL00 が最上位ニブルです)。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 36-19: GLCDCURLUTx: グラフィック LCD コントローラ カーソル LUT レジスタ「x」  
(x = 0 ~ 15)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RED<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	GREEN<7:0>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BLUE<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-24 **未実装**: 「0」として読み出し

bit 23-16 **RED<7:0>**: カーソル LUT R(赤)成分ビット

bit 15-8 **GREEN<7:0>**: カーソル LUT G(緑)成分ビット

bit 7-0 **BLUE<7:0>**: カーソル LUT B(青)成分ビット

**Note:** このレジスタ内のビットは、8ビットの RGB 色値 (0 ~ 255) を格納します。

# PIC32MZ グラフィック (DA) ファミリ

## 37.0 2D グラフィック処理ユニット (GPU)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補う文書は、Microchip 社 PIC32 ウェブサイト ([www.microchip.com/pic32](http://www.microchip.com/pic32)) の [Documentation > Reference Manual] セクションから入手できます。

2D グラフィック処理ユニットは、システム RAM または DDR2 メモリ内のフレームバッファの内容を処理する事により、最終的な画素表示のための画像の生成を高速化します。ハードウェア アクセラレーションは、各種の 2D グラフィック アプリケーション (グラフィック ユーザ インターフェイス (メニュー、オブジェクト等)、タッチスクリーン ユーザ インターフェイス、フラッシュ動画、ゲーム ソフトウェア等) で使われます。

2D GPU は垂直線、水平線、矩形の描画、画面上の異なる位置間での矩形領域のコピー、テキストの描画、既存表示データ上での圧縮データの展開等、高速なオンザフライ レンダリングも提供します。GPU のハードウェアが DMA を介してレンダリングを実行するため、レンダリングの開始後に CPU は他のタスクを実行できます。

2D グラフィック処理ユニットのブロック図を図 37-1 に示します。

**Note:** このモジュールには、ハードウェア インターフェイスに関する説明はありません。このモジュールは、付属ソフトウェア ドライバを使って制御します。

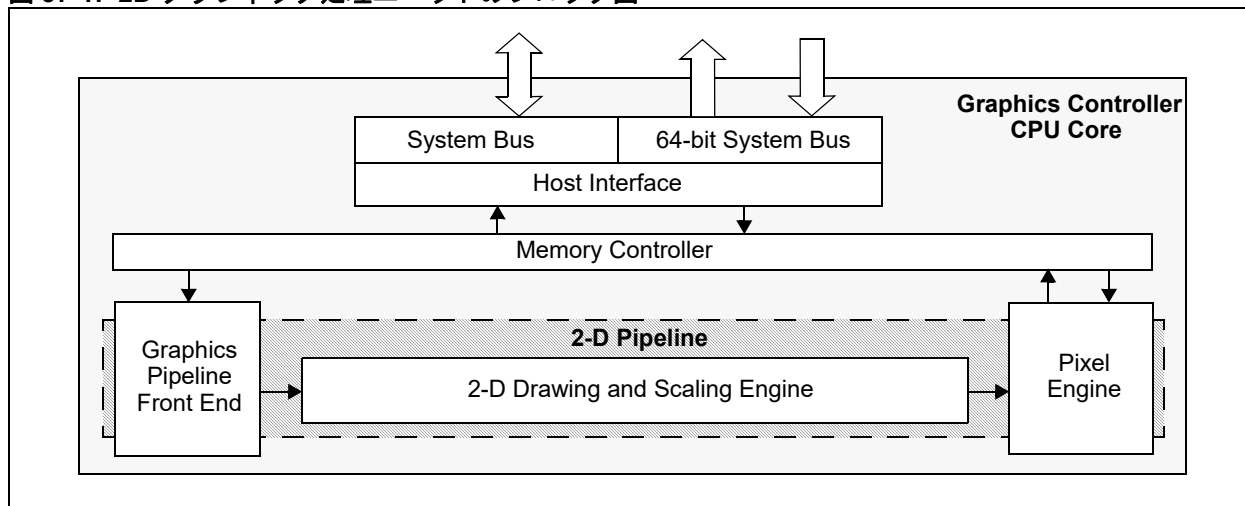
2D グラフィック処理ユニットの主な特長は以下の通りです。

- メモリに対する 64 ビット バスアクセス (高スループット)
- グローバルなクロックゲート処理 (低消費電力)
- コマンドバッファ
- プログラマブルなラスタ演算 (ROP2, ROP3, ROP4)
  - 完全なアルファ ブレンディングと透過性を提供
- 固定関数:
  - 線描画
  - 矩形塗りつぶし
  - 矩形クリア
  - ビットブロック転送 (Bit blit) (拡大 / 縮小 / フィルタ処理)
- ソースデータ フォーマット:
  - RGB888, RGB555, RGB565, RBG323, 1 ビットモノクローム
- デスティネーション データ フォーマット:
  - RGB888, RGB555, RGB565
- YUV → RGB 変換
- ディザ処理
- 回転
- クリッピング
- テキスト描画

**Note 1:** RGB ソース フォーマットについては、関連するスウィズル (swizzle) フォーマット (ARGB, RGBA, ABGR, BGRA) もサポートします。

**2:** 本 GPU は、POR 後に有効 (動作準備完了) になりますが、実行中に GPURESET ビット (CFGAPP2<0>) を使ってソフトリセットする事ができません。ソフトリセットを正しく実行するには、GPURESET ビットをトグルする前に、GPUMD ビットを「0」にクリアしてから 10  $\mu$ s 待機する必要があります。

図 37-1: 2D グラフィック処理ユニットのブロック図



# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

## 38.0 DDR2 SDRAM コントローラ

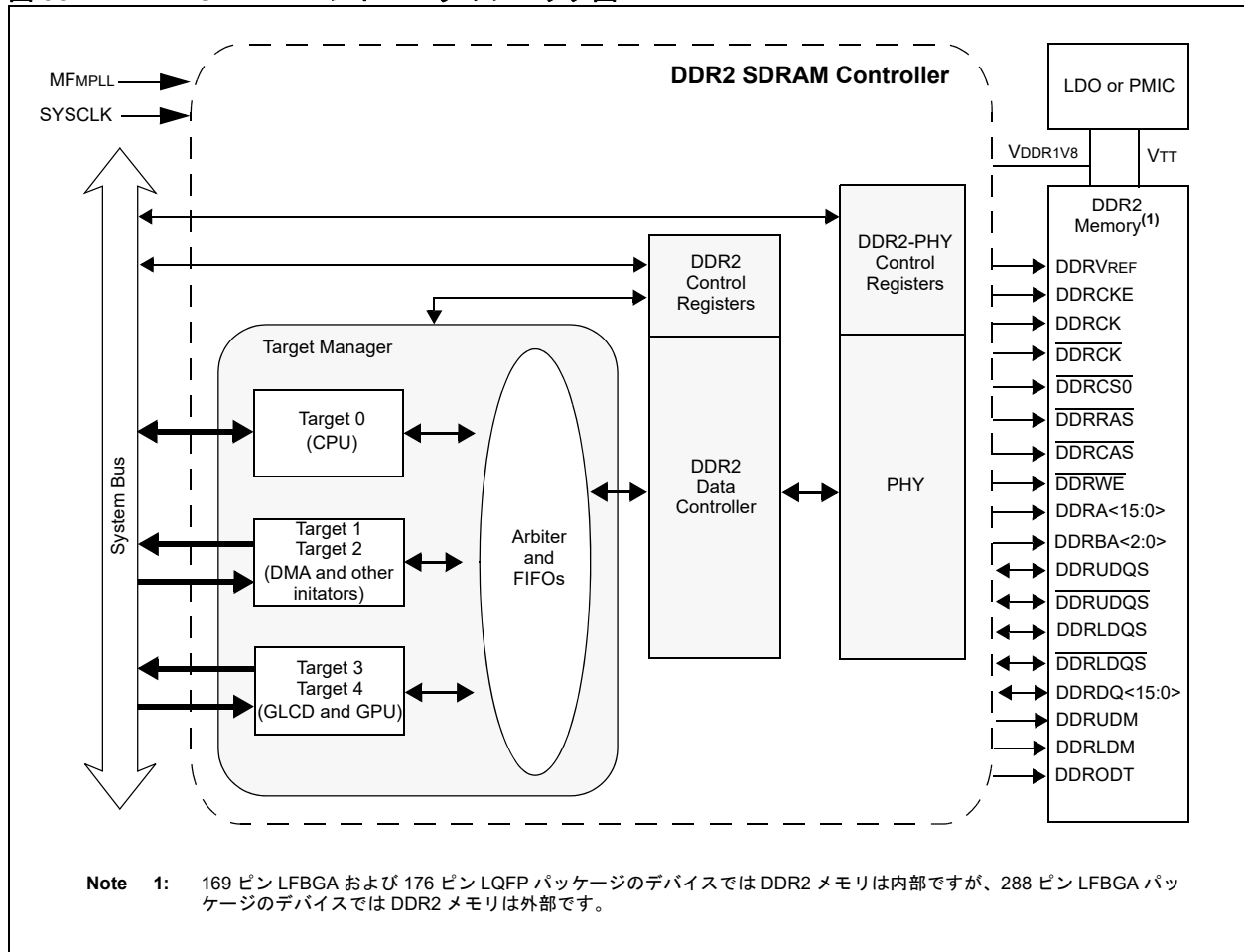
**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 55. DDR SDRAM Controller』(DS60001321) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

DDR SDRAM コントローラは、DDR (Dual Data Rate) バージョン 2 (DDR2) プロトコルを使う外部メモリバスインターフェイスと JEDEC 規格 (JESD79-2F (Nov. 2009)) に従う電氣的インターフェイスのための制御機能を実装します。

このコントローラは、オプション設定が可能な DDR2 SDRAM コントローラ コアと DDR2 物理インターフェイスにより構成されます。

図 38-1 は、これらの構成要素の接続を示したブロック図です。

図 38-1: DDR2 SDRAM コントローラのブロック図



### 38.1 制御レジスタ

表 38-1: DDR SDRAM コントローラのレジスタ一覧

アドレス (BF8E_#)	レジスタ名	ビットレンジ	Bit																リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0		
8000	DDR TSEL	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	TSEL<7:0>																0000	
8004	DDR MINLIM	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	MINLIMIT<4:0>																0000	
8008	DDR RQPER	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	RQPER<7:0>																0000	
800C	DDR MINCMD	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	MINCMD<7:0>																0000	
8010	DDR MEMCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INITDN	STINIT	0000	
8014	DDR MEMCFG0	31:16	—	AP CHRGEN	—	CLHADDR<4:0>						—	—	—	CSADDR<4:0>					0000
		15:0	—	—	—	BNKADDR<4:0>						—	—	—	RWADDR<4:0>					0000
8018	DDR MEMCFG1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	RWADDRMSK<12:0>																0000	
801C	DDR MEMCFG2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CLADDRHMSK<12:0>																0000	
8020	DDR MEMCFG3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	CLADDRMSK<12:0>																0000	
8024	DDR MEMCFG4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000	
		15:0	—	—	—	—	—	—	CSADDRMSK<2:0>				—	—	—	BNKADDRMSK<2:0>				0000
8028	DDR REFCFG	31:16	—	—	—	—	—	—	MAXREFS<2:0>				REFDLY<7:0>					0000		
		15:0	REFCNT<15:0>																0000	
802C	DDR PWRCFG	31:16	—	—	—	—	—	—	—	—	—	PCHRG PWRDN	SLFREFDLY<9:4>					0000		
		15:0	SLFREFDLY<3:0>						PWRDNDLY<7:0>						ASLF REFEN	APWR DNEN	—	—	0000	
8030	DDR DLYCFG0	31:16	RMWDLY<3:0>				R2WDLY<3:0>				W2WCSDLY<3:0>				W2WDLY<3:0>				0000	
		15:0	R2RCSPLY<3:0>				R2RDLY<3:0>				W2RCSPLY<3:0>				W2RDLY<3:0>				0000	
8034	DDR DLYCFG1	31:16	—	SLFREF EXDLY8	NXTDAT AVDLY4	W2R CSDLY4	W2R DLY4>	W2PCHRG DLY4	PWRDNEXDLY<5:0>					PWRDNMINDLY<3:0>					0000	
		15:0	SLFREFEXDLY<7:0>								SLFREFMINDLY<7:0>								0000	
8038	DDR DLYCFG2	31:16	RBENDLY<3:0>				PCHRG2RASDLY<3:0>				RAS2CASDLY<3:0>				RAS2RASDLY<3:0>				0000	
		15:0	W2PCHRGDLY<3:0>				R2PCHRGDLY<3:0>				—	—	—	—	PCHRGALLDLY<3:0>				0000	
803C	DDR DLYCFG3	31:16	—	—	—	—	—	—	—	—	—	—	FAWTDLY<5:0>					0000		
		15:0	RAS2RASSBNKDLY<5:0>								RAS2PCHRGDLY<4:0>								0000	
8040	DDR ODTCFG	31:16	—	—	—	—	—	—	—	—	—	ODTWLEN<2:0>				ODTRLEN<2:0>				0000
		15:0	ODTDLY<3:0>				ODTRDLY<3:0>				ODTSEN<7:0>								0000	
8044	DDR XFERCFG	31:16	BIGENDIA N	—	—	—	MAXBURST<3:0>					—	—	—	—	RDATENDLY<3:0>				0000
		15:0	—	—	—	—	—	—	—	—	NXTDATAVDLY<3:0>				NXTDATRQDLY<3:0>				0000	

表 38-1: DDR SDRAM コントローラのレジスタ一覧 (続き)

アドレス (BF8E_#)	レジスタ名	ビットレンジ	Bit															リセット
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	
8048	DDR CMDISSUE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	VALID	NUMHOSTCMDS<3:0>			
804C	DDR ODTENCFG	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ODTWEN
8050	DDR MEMWIDTH	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	HALF レート	—	—	—
8080	DDR CMD10	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		
8084	DDR CMD11	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		
8088	DDR CMD12	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		
808C	DDR CMD13	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		
8090	DDR CMD14	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		
8094	DDR CMD15	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		
8098	DDR CMD16	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		
809C	DDR CMD17	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		
80A0	DDR CMD18	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		
80A4	DDR CMD19	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>					CLKEN CMD1	0000		

表 38-1: DDR SDRAM コントローラのレジスタ一覧 (続き)

仮アドレス (BF8E_#)	レジスタ名	ビットレンジ	Bit															書きセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
80A8	DDR CMD110	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000	
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>						CLKEN CMD1	0000		
80AC	DDR CMD111	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000	
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>						CLKEN CMD1	0000		
80B0	DDR CMD112	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000	
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>						CLKEN CMD1	0000		
80B4	DDR CMD113	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000	
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>						CLKEN CMD1	0000		
80B8	DDR CMD114	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD<27:3>					0000	
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>						CLKEN CMD1	0000		
80BC	DDR CMD115	31:16	MDALCMD<7:0>							WEN CMD2	CASCMD2	RASCMD2	CSCMD2<7:3>					0000	
		15:0	CSCMD2<2:0>			CLKEN CMD2	WEN CMD1	CASCMD1	RASCMD1	CSCMD1<7:0>						CLKEN CMD1	0000		
80C0	DDR CMD20	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80C4	DDR CMD21	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80C8	DDR CMD22	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80CC	DDR CMD23	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80D0	DDR CMD24	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80D4	DDR CMD25	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80D8	DDR CMD26	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80DC	DDR CMD27	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80E0	DDR CMD28	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80E4	DDR CMD29	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			
80E8	DDR CMD210	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>	0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>					0000			



表 38-1: DDR SDRAM コントローラのレジスタ一覧 (続き)

アドレス (BF8E_#)	レジスタ名	ビットレンジ	Bit															リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
80EC	DDR CMD211	31:16	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>				0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>							0000	
80F0	DDR CMD212	31:16	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>				0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>							0000	
80F4	DDR CMD213	31:16	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>				0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>							0000	
80F8	DDR CMD214	31:16	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>				0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>							0000	
80FC	DDR CMD215	31:16	—	—	—	—	—	—	—	—	—	—	—	—	WAIT<8:5>				0000
		15:0	WAIT<4:0>				BNKADDRCMD<2:0>				MDADDRHCMD<7:0>							0000	
9100	DDR SCLSTART	31:16	—	—	—	SCL START	—	SCL EN	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SCL UBPASS	SCL LBPASS
910C	DDR SCLLAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	DDRCLKDLY<3:0>				CAPCLKDLY<3:0>				0000
9118	DDR SCLCFG0	31:16	—	—	—	—	—	—	—	ODTCSW	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	RCASLAT<3:0>				—	—	—	DDR2	BURST8
911C	DDR SCLCFG1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	DBL REFDLY	WCASLAT<3:0>				—	—	—	—	—	—	—	—	SCLSEN
9120	DDR PHYPADCON	31:16	—	PREAMBDLY<1:0>		R CVREN	—	—	—	—	DRVSTRPFET<3:0>				DRVSTRNFET<3:0>				0000
		15:0	—	HALF レート	WR CMD DLY	—	—	—	NOEXT DLL	EOEN CLKCYC	ODTPUCAL<1:0>		ODTPDCAL<1:0>		ADDC DRVSEL	DAT DRVSEL	ODTEN	ODTSEL	0000
9124	DDR PHYDLLR	31:16	DLYSTVAL<3:0>				—	DIS RECALIB	RECALIBCNT<17:8>										0000
		15:0	RECALIBCNT<7:0>															0000	
9140	DDR PHYCLKDLY	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	SCL UBPASS	SCL LBPASS	—	CLKDLYDELTA<2:0>		

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-1: DDRTSEL: DDR ターゲット選択レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	TSEL<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-8 **未実装:** 「0」として読み出し

bit 7-0 **TSEL<7:0>:** ターゲット選択ビット

これらのビットは、パラメータを書き込むターゲットを選択します。このフィールドは、ターゲットに調停パラメータを書き込む前に設定する必要があります。このフィールドの値は、ターゲット番号 (0 ~ 4) に調停パラメータのフィールドサイズを乗算した値を表します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-2: DDRMINLIM: DDR 最小バースト数制限レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	MINLIMIT<4:0>				

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-5 **未実装:** 「0」として読み出し

bit 4-0 **MINLIMIT<4:0>:** 最小バースト数制限ビット

これらのビットは、ターゲットが別のターゲットから割り込まれる事なくアクセスする必要がある DDR バーストの最小数 (バーストあたり 2 サイクル) を指定します。

**Note:** このレジスタを使って最小バースト数の制限値をターゲットに書き込む前に、値 [ ターゲット番号 × MINLIMIT フィールドのサイズ (5) ] を TSEL<7:0> ビット (DDRTSEL<7:0>) に書き込む必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-3: DDRQPER: DDR 要求期間レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	RQPER<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-8 **未実装:** 「0」として読み出し

bit 7-0 **RQPER<7:0>:** 要求期間ビット

これらのビットと MINCMD<7:0> ビット (DDRMINCMD<7:0>) の組み合わせにより、ターゲットに割り当てる総帯域幅の割合 (%) を指定します。[RQPER<7:0> × 4] クロックのアクセス要求期間中に MINCMD<7:0> で指定された DDR バーストの数が処理されなかった場合、この条件が満足されるまでそのターゲットの要求は高優先度で扱われます。

**Note:** このレジスタを使って最小バースト数の制限値をターゲットに書き込む前に、値 [ターゲット番号 × MILIMIT フィールドのサイズ (5)] を TSEL<7:0> ビット (DDRTSEL<7:0>) に書き込む必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-4: DDRMINCMD: DDR 最小コマンド数レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	MINCMD<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-8 **未実装:** 「0」として読み出し

bit 7-0 **MINCMD<7:0>:** 最小コマンド数ビット

これらのビットと RQPER<7:0> ビット (DDRRQPER<7:0>) の組み合わせにより、ターゲットに割り当てる総帯域幅の割合 (%) を指定します。[RQPER<7:0> × 4] クロックのアクセス要求期間中に MINCMD<7:0> で指定された DDR パーストの数が処理されなかった場合、この条件が満足されるまでそのターゲットの要求は高優先度で扱われます。

**Note:** このレジスタを使って最小パースト数の制限値をターゲットに書き込む前に、値 [ターゲット番号 × MILIMIT フィールドのサイズ (5)] を TSEL<7:0> ビット (DDRTSEL<7:0>) に書き込む必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-5: DDRMEMCON: DDR メモリ制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 INITDN	R/W-0 STINIT

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-24 **未実装:** 「0」として読み出し

bit 1 **INITDN:** メモリ初期化完了ビット

このビットは、コントローラの通常動作を可能にするため、メモリ初期化が完了した後にソフトウェアによってセットされます。

1 = 全てのコマンドが発行された (コントローラは通常動作できる)  
 0 = コントローラは通常動作できない

bit 0 **STINIT:** メモリ初期化開始ビット

このビットは、メモリの初期化を開始するために、メモリ初期化コマンドが DDRCMD レジスタに書き込まれた後にソフトウェアによってセットされます。

1 = メモリの初期化を開始する  
 0 = メモリの初期化を開始しない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-6: DDRMEMCFG0: DDR メモリ コンフィグレーションレジスタ 0

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	APCHRGEN	—	CLHADDR<4:0>				
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	CSADDR<4:0>				
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	BNKADDR<4:0>				
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	RWADDR<4:0>				

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31 **未実装:** 「0」として読み出し

bit 30 **APCHRGEN:** 自動プリチャージイネーブルビット

このビットがセットされている場合、各ユーザコマンドの最後にバンクを閉じるために、自動プリチャージコマンドが発行されます。コマンドが実行完了前に複数のバンクにアクセスする場合、アクセスされる全てのバンクが自動プリチャージされます。

1 = 自動プリチャージコマンドを発行する  
 0 = 自動プリチャージコマンドは発行しない

bit 29 **未実装:** 「0」として読み出し

bit 28-24 **CLHADDR<4:0>:** 列アドレスシフトビット

これらのビットは、列アドレスの上位部を列アドレスの下位部のすぐ左に寄せるために必要なコントローラ      アドレスの右方向へのビットシフト量を指定します。これらのビットは、CLADDRHMSK (DDRMEMCFG2<26:0>) および CLADDRMASK (DDRMEMCFG3<26:0>) と組み合わせて使います。

bit 23-21 **未実装:** 「0」として読み出し

bit 20-16 **CSADDR<4:0>:** チップセレクトシフトビット

これらのビットは、DDR メモリのチップセレクトアドレスを生成するためにユーザアドレス空間のどのビットを使うか指定します。これらのビットは CSADDRMASK (DDRMEMCFG4<10:8>) と組み合わせて使います。

bit 15-13 **未実装:** 「0」として読み出し

bit 12-8 **BNKADDR<4:0>:** バンクアドレス選択シフトビット

これらのビットは、DDR メモリのバンクアドレスを生成するためにユーザアドレス空間のどのビットを使うか指定します。これらのビットは BNKADDRMASK (DDRMEMCFG4<2:0>) と組み合わせて使います。

bit 7-5 **未実装:** 「0」として読み出し

bit 4-0 **RWADDR<4:0>:** 行アドレス選択シフトビット

これらのビットは、DDR メモリの行アドレスを生成するためにユーザアドレス空間のどのビットを使うか指定します。これらのビットは RWADDRMASK (DDRMEMCFG1<12:0>) と組み合わせて使います。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-7: DDRMEMCFG1: DDR メモリ コンフィグレーション レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	RWADDRMSK<12:8>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RWADDRMSK<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-13 **未実装**: 「0」として読み出し

bit 12-0 **RWADDRMSK<12:0>**: 行アドレス マスクビット

これらのビットは、RWADDR<4:0> ビット (DDRMEMCFG0<4:0>) と組み合わせて使う事で、DDR メモリの行アドレスを生成するためにユーザアドレス空間のどのビットを使うか指定します。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-8: DDRMEMCFG2: DDR メモリ コンフィグレーションレジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	CLADDRHMSK<12:8>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CLADDRHMSK<7:0>							

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-13 **未実装**: 「0」として読み出し

bit 12-0 **CLADDRHMSK<12:0>**: 列アドレス上位マスクビット

これらのビットは、CLADDR<4:0> ビット (DDRMEMCFG0<28:24>) および CLADDRLMASK<12:0> ビット (DDRMEMCFG3<12:0>) と組み合わせて使う事で、DDR メモリの列アドレスを生成するためにユーザアドレス空間のどのビットを使うか指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-9: DDRMEMCFG3: DDR メモリ コンフィグレーション レジスタ 3

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	CLADDRMSK<12:8>				
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CLADDRMSK<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-13 **未実装**: 「0」として読み出し

bit 12-0 **CLADDRMSK<12:0>**: 列アドレス下位マスクビット

これらのビットは、CLADDR<4:0> ビット (DDRMEMCFG0<28:24>) および CLADDRHMASK<12:0> ビット (DDRMEMCFG2<12:0>) と組み合わせて使う事で、DDR メモリの列アドレスを生成するためにユーザアドレス空間のどのビットを使うか指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-10: DDRMEMCFG4: DDR メモリ コンフィグレーションレジスタ 4

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
	—	—	—	—	—	—	—	CSADDRMSK<2>
7:0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	CSADDRMSK<1:0>		—	—	—	BNKADDRMSK<2:0>		

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
ト  
-n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-9 **未実装:** 「0」として読み出し

bit 8-6 **CSADDRMSK<2:0>:** チップセレクト アドレスマスク ビット

これらのビットは、CSADDR<4:0> ビット (DDRMEMCFG0<20:16>) と組み合わせて使う事で、DDR メモリのチップセレクト アドレスを生成するためにユーザアドレス空間のどのビットを使うか指定します。

bit 5-3 **未実装:** 「0」として読み出し

bit 2-0 **BNKADDRMSK<2:0>:** バンクアドレス マスクビット

これらのビットは、BNKADDR<4:0> ビット (DDRMEMCFG0<12:8>) と組み合わせて使う事で、DDR メモリのバンクアドレスを生成するためにユーザアドレス空間のどのビットを使うか指定します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-11: DDRREFCFG: DDR リフレッシュ コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	MAXREFS<2:0>		
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	REFDLY<7:0>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	REFCNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	REFCNT<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-27 **未実装**: 「0」として読み出し

bit 26-24 **MAXREFS<2:0>**: 保留リフレッシュ最大数ビット

これらのビットは、いつ保留しても構わないリフレッシュの最大数を指定します。1つまたは複数のリフレッシュが保留中である時にアイドル時間が存在すると、新しい要求を受け取るまで保留中リフレッシュが繰り返し発行されます。MAXREFS <2:0> で指定した数のリフレッシュが保留中である時にアイドル時間が存在しない場合、保留中リフレッシュの少なくとも1つのバーストが発行可能になるまで後続の要求は停止します。

bit 23-16 **REFDLY<7:0>**: 最小リフレッシュ間遅延ビット

これらのビットは、リフレッシュと次のリフレッシュの間に必要な最小クロック数を指定します。

bit 15-0 **REFCNT<15:0>**: リフレッシュ カウントビット

これらのビットは、リフレッシュの平均周期 ( 間隔 ) に対応するクロックサイクル数を指定します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-12: DDRPWRCFG: DDR 省電力モード コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	PCHRGPWDN	SLFREFDLY<9:4>					
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SLFREFDLY<3:0>				PWDNDLY<7:4>			
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
	PWDNDLY<3:0>				ASLFREFEN	APWRDNEN	—	—

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-23 **未実装**: 「0」として読み出し

bit 22 **PCHRGPWDN**: プリチャージ省電力モードのみ許可ビット

プリチャージ省電力モードへのみ自動的な移行を許可します (アクティブ省電力モードへの移行は禁止)。いずれかの行が開いている場合、それらの行は DDR SDRAM がプリチャージ省電力モードへ移行する前にプリチャージされます。

1 = プリチャージ省電力モードへの自動的な移行を許可する

0 = プリチャージ省電力モードへの自動的な移行を許可しない

bit 21-12 **SLFREFDLY<9:0>**: 自己リフレッシュ遅延ビット

これらのビットは、自己リフレッシュモードへの自動的な移行の前にコントローラが待機する必要があるアイドル時間の最小クロックサイクル数を指定します。この値を 1024 倍した値がクロックサイクル数を表します。

111111111 = 2111452 クロック

....

000000001 = 1024 クロック

bit 11-4 **PWDNDLY<7:0>**: リフレッシュ カウントビット

これらのビットは、省電力モード (アクティブまたはプリチャージ) への自動的な移行の前にコントローラが待機する必要があるアイドル時間の最小クロックサイクル数を指定します。この値を 4 倍した値がクロックサイクル数を表します。

11111111 = 1020 クロック

....

000000001 = 4 クロック

bit 3 **ASLFREFEN**: 自己リフレッシュ自動移行イネーブルビット

1 = 自己リフレッシュモードへの自動的な移行を許可する

0 = 自己リフレッシュモードへの自動的な移行を許可しない

bit 2 **APWRDNEN**: 省電力モード自動移行イネーブルビット

1 = 省電力モードへの自動的な移行を許可する

0 = 省電力モードへの自動的な移行を許可しない

bit 1-0 **未実装**: 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-13: DDRDLYCFG0: DDR 遅延コンフィグレーションレジスタ 0

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RMWDLY<3:0>				R2WDLY<3:0>			
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	W2WCSDLY<3:0>				W2WDLY<3:0>			
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	R2RCSPLY<3:0>				R2RDLY<3:0>			
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	W2RCSPLY<3:0>				W2RDLY<3:0>			

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-28 **RMWDLY<3:0>**: 読み出し - 変更 - 書き込み間遅延ビット  
 これらのビットは、読み出し - 変更 - 書き込み動作のために発行される 2 つの読み書きコマンドの間に必要な最小クロック数を指定します。
- bit 27-24 **R2WDLY<3:0>**: 読み出し - 書き込み間遅延ビット  
 これらのビットは、読み出しコマンドと書き込みコマンドの間に必要な最小クロック数を指定します。読み出しコマンドと書き込みコマンドは、同じチップセレクトに対する物であっても互いに異なるチップセレクトに対する物であっても構いません。
- bit 23-20 **W2WCSDLY<3:0>**: チップセレクトの異なる書き込み - 書き込み間遅延ビット  
 これらのビットは、異なるチップセレクトに対する 2 つの書き込みコマンドの間に必要な最小クロック数を指定します。
- bit 19-16 **W2WDLY<3:0>**: 書き込み - 書き込み間遅延ビット  
 これらのビットは、同じチップセレクトに対する 2 つの書き込みコマンドの間に必要な最小クロック数を指定します。
- bit 15-12 **R2RCSPLY<3:0>**: チップセレクトの異なる読み出し - 読み出し間遅延ビット  
 これらのビットは、異なるチップセレクトに対する 2 つの読み出しコマンドの間に必要な最小クロック数を指定します。
- bit 11-8 **R2RDLY<3:0>**: 読み出し - 読み出し間遅延ビット  
 これらのビットは、同じチップセレクトに対する 2 つの読み出しコマンドの間に必要な最小クロック数を指定します。
- bit 7-4 **W2RCSPLY<3:0>**: チップセレクトの異なる書き込み - 読み出し間遅延ビット  
 これらのビットは、異なるチップセレクトに対する書き込みコマンドと読み出しコマンドの間に必要な最小クロック数を指定します。
- bit 3-0 **W2RDLY<3:0>**: 書き込み - 読み出し間遅延ビット  
 これらのビットは、同じチップセレクトに対する書き込みコマンドと読み出しコマンドの間に必要な最小クロック数を指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-14: DDRDLYCFG1: DDR 遅延コンフィグレーションレジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	R/W-0 SLFREF EXDLY<8>	R/W-0 NXTDAT AVDLY<4>	R/W-0 W2RCS DLY<4>	R/W-0 W2RDLY<4>	R/W-0 W2PCHRG DLY<4>	R/W-0 PWRDNEXDLY<5:4>	
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PWRDNEXDLY<3:0>				PWRDNMINDLY <3:0>			
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SLFREFEXDLY<7:0>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SLFREFMINDLY<7:0>							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31    **未実装:** 「0」として読み出し
- bit 30    **SLFREFEXDLY:** 自己リフレッシュ終了遅延ビット 8  
このビットは、自己リフレッシュモードを終了してから通常動作を始める前に必要な最小クロック数を指定します。
- bit 29    **NXTDATAVDLY:** 次のデータが利用可能になるまでの遅延ビット 4  
これらのビットは、書き込みコマンドと書き込みデータ転送ハンドシェイク信号「next data request」の間に必要な最小クロックサイクル数を指定します。NXTDATAVDLY<3:0> ビット (DDRXFRCFG<7:4>) の説明も参照してください。
- bit 28    **W2RCS DLY:** チップセレクトの異なる書き込み - 読み出し間遅延ビット 4  
このビットは、異なるチップセレクトに対する書き込みコマンドと読み出しコマンドの間に必要な最小クロック数を指定します。W2RCS DLY<3:0> ビット (DDR DLYCFG0<7:4>) の説明も参照してください。
- bit 27    **W2RDLY:** 書き込み - 読み出し間遅延ビット 4  
このビットは、同じチップセレクトに対する書き込みコマンドと読み出しコマンドの間に必要な最小クロック数を指定します。W2RDLY<3:0> ビット (DDR DLYCFG0<3:0>) の説明も参照してください。
- bit 26    **W2PCHRG DLY:** 書き込み - プリチャージ間遅延ビット 4  
これらのビットは、書き込みコマンドから同じバンクに対する書き込みとしてのプリチャージコマンドまでの間に必要な最小クロック数を指定します。W2PCHRG DLY<3:0> ビット (DDR DLYCFG2<15:12>) の説明も参照してください。
- bit 25-20 **PWRDNEXDLY<5:0>:** 省電力モード終了遅延ビット  
これらのビットは、省電力モードを終了してから通常動作を始める前に必要な最小クロック数を指定します。
- bit 19-16 **PWRDNMINDLY<3:0>:** 省電力モード最小遅延ビット  
これらのビットは、省電力モードに移行した後にそのモードを持続する最小クロック数を指定します。
- bit 15-8    **SLFREFEXDLY<7:0>:** 自己リフレッシュ終了遅延ビット  
これらのビットは、自己リフレッシュモードを終了してから通常動作を始める前に必要な最小クロック数を指定します。
- bit 7-0    **SLFREFMINDLY<7:0>:** 自己リフレッシュ最小遅延ビット  
これらのビットは、自己リフレッシュモードに移行した後にそのモードを持続する最小クロック数を指定します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-15: DDRDLYCFG2: DDR 遅延コンフィグレーションレジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RBENDDLY<3:0>				PCHRG2RASDLY<3:0>			
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RAS2CASDLY<3:0>				RAS2RASDLY <3:0>			
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	W2PCHRGDLY<3:0>				R2PCHRGDLY<3:0>			
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	PCHRGALLDLY<3:0>			

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-28 **RBENDDLY<3:0>**: 読み出し - バースト終了間遅延ビット  
 これらのビットは、読み出しコマンドを発行してからデータバーストの読み出しが完了するまでに必要な最小クロック数を指定します。
- bit 27-24 **PCHRG2RASDLY<3:0>**: プリチャージ-RAS 間遅延ビット  
 これらのビットは、プリチャージコマンドから同じバンクに対する RAS コマンドまでの間に必要な最小クロック数を指定します。
- bit 23-20 **RAS2CASDLY<3:0>**: RAS-CAS 間遅延ビット  
 これらのビットは、RAS コマンドから同じバンクに対す CAS コマンドまでの間に必要な最小クロック数を指定します。
- bit 19-16 **RAS2RASDLY<3:0>**: 書き込み - 読み出し間遅延ビット  
 これらのビットは、RAS コマンドから同じチップセレクトの異なるバンクに対する RAS コマンドまでの間に必要な最小クロック数を指定します。
- bit 15-12 **W2PCHRGDLY<3:0>**: 書き込み - プリチャージ間遅延ビット 3-0  
 これらのビットは、書き込みコマンドから同じバンクに対する書き込みとしてのプリチャージ コマンドまでの間に必要な最小クロック数を指定します。  
 15クロックサイクルを超える遅延のためにオーバーフロービット(DDRDLYCFG1<26>)が提供されます。
- bit 11-8 **R2PCHRGDLY<3:0>**: 読み出し - プリチャージ間遅延ビット  
 これらのビットは、読み出しコマンドから同じバンクに対する読み出しとしてのプリチャージ コマンドまでの間に必要な最小クロック数を指定します。
- bit 7-4 **未実装**: 「0」として読み出し
- bit 3-0 **PCHRGALLDLY<3:0>**: 全バンク プリチャージ遅延ビット  
 これらのビットは、全バンク プリチャージ コマンドからアクティベートまたはリフレッシュ コマンドまでの間に必要な最小クロック数を指定します。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-16: DDRDLYCFG3: DDR 遅延コンフィグレーションレジスタ 3

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FAWTDLY<5:0>							
15:8	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RAS2RASSBNKDLY<5:0>							
7:0	U-0 —	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RAS2PCHRGDLY<4:0>							

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-22 **未実装**: 「0」として読み出し

bit 21-16 **FAWTDLY<5:0>**: 4バンク アクティベート ウィンドウ時間遅延ビット

これらのビットは、4つのバンクを開くために必要な最小クロック数を指定します。

bit 15-14 **未実装**: 「0」として読み出し

bit 13-8 **RAS2RASSBNKDLY<5:0>**: 同じバンクの RAS-RAS 間遅延ビット

これらのビットは、同じバンクに対する2つの RAS コマンドの間に必要な最小クロック数を指定します。

bit 7-5 **未実装**: 「0」として読み出し

bit 4-0 **RAS2PCHRGDLY<4:0>**: RAS- プリチャージ間遅延ビット

これらのビットは、RAS コマンドから同じバンクに対するプリチャージ コマンドまでの間に必要な最小クロック数を指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-17: DDRODTCFG: DDR オンダイターミネーションコンフィグレーションレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	R/W-0 ODTWLEN<2:0>	R/W-0 ODTWLEN<2:0>	R/W-0 ODTWLEN<2:0>	U-0 —	R/W-0 ODTRLEN<2:0>	R/W-0 ODTRLEN<2:0>	R/W-0 ODTRLEN<2:0>
15:8	R/W-0 OTDWDLY<3:0>	R/W-0 OTDWDLY<3:0>	R/W-0 OTDWDLY<3:0>	R/W-0 OTDWDLY<3:0>	R/W-0 OTDRDLY<3:0>	R/W-0 OTDRDLY<3:0>	R/W-0 OTDRDLY<3:0>	R/W-0 OTDRDLY<3:0>
7:0	R/W-0 OTDCSEN<7:0>	R/W-0 OTDCSEN<7:0>	R/W-0 OTDCSEN<7:0>	R/W-0 OTDCSEN<7:0>	R/W-0 OTDCSEN<7:0>	R/W-0 OTDCSEN<7:0>	R/W-0 OTDCSEN<7:0>	R/W-0 OTDCSEN<7:0>

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-23 **未実装:** 「0」として読み出し
- bit 22-20 **ODTWLEN<2:0>:** オンダイターミネーション書き込み長ビット  
 これらのビットは、書き込みのためにオンダイターミネーション (ODT) を ON にするクロック数を指定します。
- bit 19 **未実装:** 「0」として読み出し
- bit 18-16 **ODTRLEN<2:0>:** オンダイターミネーション読み出し長ビット  
 これらのビットは、読み出しのためにオンダイターミネーション (ODT) を ON にするクロック数を指定します。
- bit 15-12 **OTDWDLY<3:0>:** オンダイターミネーション書き込み遅延ビット  
 これらのビットは、書き込みコマンドの後に DDR に対して ODT を ON にするまでのクロック数を指定します。
- bit 11-8 **OTDRDLY<3:0>:** オンダイターミネーション読み出し遅延ビット  
 これらのビットは、読み出しコマンドの後に DDR に対して ODT を ON にするまでのクロック数を指定します。
- bit 7-0 **OTDCSEN<7:0>:** オンダイターミネーションチップセレクトイネーブルビット  
 これらのビットは、DDRODTCFG レジスタ ( [レジスタ 38-20](#) ) と組み合わせて使う事で、各チップセレクトに対する ODT 制御を設定します。このフィールドの値は、[ チップセレクトの数 × 設定するチップセレクトの番号 ] を表します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-18: DDRXFERCFG: DDR 転送コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	U-0	U-0	U-0	R/W-0	R/W-1	R/W-0	R/W-0
	BIGENDIAN	—	—	—	MAXBURST<3:0>			
23:16	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	RDATENDLY<3:0>			
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	NXTDATAVDLY<3:0>				NXTDATRQDLY<3:0>			

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31    **BIGENDIAN:** ビッグエンディアン ビット

1 = データはビッグエンディアン形式  
 0 = データはリトルエンディアン形式

bit 30-28 **未実装:** 「0」として読み出し

bit 27-24 **MAXBURST<3:0>:** 最大コマンドバースト数ビット

これらのビットは、バーストモードで DDR コントローラへ書き込む事のできるコマンドの最大数を指定します。

bit 23-20 **未実装:** 「0」として読み出し

bit 19-16 **RDATENDLY<3:0>:** PHY データ読み出しイネーブル遅延ビット

これらのビットは、PHY に対して読み出しコマンドを発行してから PHY に対する「read data enable」信号がアサートされるまでに必要な最小クロック数を指定します。

bit 15-8 **未実装:** 「0」として読み出し

bit 7-4 **NXTDATAVDLY<3:0>:** 次のデータが利用可能になるまでの遅延ビット

これらのビットは、読み出しコマンドを発行してから読み出しデータを受け取るまでに必要な最小クロック数を指定します。

bit 3-0 **NXTDATRQDLY<3:0>:** 次のデータ要求までの遅延ビット

これらのビットは、書き込みコマンドの発行と書き込みデータ転送ハンドシェイク信号「next data request」の間に必要な最小クロックサイクル数を指定します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-19: DDRCMDISSUE: DDR コマンド発行レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	U-0 —	U-0 —	R/W-0, HC VALID	R/W-0	R/W-0	R/W-0	R/W-0
					NUMHOSTCMDS<3:0>			

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-5 **未実装:** 「0」として読み出し

bit 4 **VALID:** ホストコマンド有効ビット

「1」が書き込まれた場合、このビットはホストコマンド レジスタ内のデータは有効である (SDRAM へ転送すべきである) という事をコントローラに知らせます。全てのデータが転送されると、このビットはハードウェアによってクリアされます。

bit 3-0 **NUMHOSTCMDS<3:0>:** ホストコマンド数ビット

SDRAM へ転送するホストコマンドの数を表します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-20: DDRODTENCFG: DDR オンダイ ターミネーション イネーブル コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 ODTWEN
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 ODTREN

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-17 **未実装:** 「0」として読み出し

bit 16 **ODTWEN:** オンダイ ターミネーション書き込みイネーブルビット

- 1 = OTDCSEN<7:0> ビット (DDRODTCFG<7:0>) によって表されるチップセレクトはデータ読み出しに対して有効な ODT を持つ
- 0 = OTDCSEN<7:0> ビット (DDRODTCFG<7:0>) によって表されるチップセレクトはデータ読み出しに対して無効な ODT を持つ

bit 15-1 **未実装:** 「0」として読み出し

bit 0 **ODTREN:** オンダイ ターミネーション読み出しイネーブルビット

- 1 = OTDCSEN<7:0> ビット (DDRODTCFG<7:0>) によって表されるチップセレクトはデータ書き込みに対して有効な ODT を持つ
- 0 = OTDCSEN<7:0> ビット (DDRODTCFG<7:0>) によって表されるチップセレクトはデータ書き込みに対して無効な ODT を持つ

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-21: DDRMEMWIDTH: DDR メモリ幅レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
	—	—	—	—	HALFRATE	—	—	—

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-4 **未実装:** 「0」として読み出し

bit 3 **HALFRATE:** ハーフレートモードビット

PIC32 は常にハーフレートモードで動作します。このビットは初期化中にセットする必要があります。

1 = ハーフレートモード

0 = フルレートモード

bit 2-0 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-22: DDRCMD1x: DDR ホストコマンド 1 レジスタ「x」(「x」= 0 ~ 15)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MDALCMD<7:0>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	WENCMD2	CASCMD2	RASCMD2	CSCMD2<7:3>				
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CSCMD2<2:0>			CLKENCMD2	WENCMD1	CASCMD1	RASCMD1	CSCMD1<7>
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CSCMD1<6:0>							CLKENCMD1

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-24 **MDALCMD<7:0>**: モードアドレス下位コマンドビット  
 これらのビットは、コマンド発行時に SDRAM アドレスの bit 7-0 に対して駆動する値を指定します。
- bit 23 **WENCMD2**: 書き込みイネーブル コマンド 2 ビット  
 このビットは、コマンド発行の 2 サイクル目と後続のサイクルで WE\_N に対して駆動する値を指定します。
- bit 22 **CASCMD2**: 列アドレス ストロープ コマンド 2 ビット  
 このビットは、コマンド発行の 2 サイクル目と後続のサイクルで CAS\_N に対して駆動する値を指定します。
- bit 21 **RASCMD2**: 行アドレス ストロープ コマンド 2 ビット  
 このビットは、コマンド発行の 2 サイクル目と後続のサイクルで RAS\_N に対して駆動する値を指定します。
- bit 20-13 **CSCMD2<7:0>**: チップセレクト コマンド 2 ビット  
 これらのビットは、コマンド発行の 2 サイクル目と後続のサイクルで CS\_N 信号 (最大 8) に対して駆動する値を指定します。
- bit 12 **CLKENCMD2**: クロック イネーブル コマンド 2 ビット  
 このビットは、コマンド発行の 2 サイクル目と後続のサイクルで CKE に対して駆動する値を指定します。
- bit 11 **WENCMD1**: 書き込みイネーブル コマンド 1 ビット  
 このビットは、コマンド発行の最初のサイクルで WE\_N に対して駆動する値を指定します。
- bit 10 **CASCMD1**: 列アドレス ストロープ コマンド 1 ビット  
 このビットは、コマンド発行の最初のサイクルで CAS\_N に対して駆動する値を指定します。
- bit 9 **RASCMD1**: 行アドレス ストロープ コマンド 1 ビット  
 このビットは、コマンド発行の最初のサイクルで RAS\_N に対して駆動する値を指定します。
- bit 8-1 **CSCMD1<7:0>**: チップセレクト コマンド 1 ビット  
 これらのビットは、コマンド発行の最初のサイクルで CS\_N 信号 (最大 8) に対して駆動する値を指定します。
- bit 0 **CLKENCMD1**: クロック イネーブル コマンド 1 ビット  
 このビットは、コマンド発行の最初のサイクルで CKE に対して駆動する値を指定します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-23: DDRCMD2x: DDR ホストコマンド 2 レジスタ 「x」 (「x」= 0 ~ 15)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	WAIT<8:5>			
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	WAIT<4:0>				BNKADDRCMD<2:0>			
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	MDADDRHCMD<7:0>							

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-20 **未実装**: 「0」として読み出し

bit 19-11 **WAIT<8:0>**: コマンド待機ビット

これらのビットは、コマンド発行後に次のコマンドを発行するまでの待機クロックサイクル数を指定します。

bit 10-8 **BNKADDRCMD<2:0>**: バンクアドレス コマンドビット

これらのビットは、コマンド発行時にバンクアドレス ビットに対して駆動する値を指定します。

bit 7-0 **MDADDRHCMD<7:0>**: モードアドレス上位コマンドビット

これらのビットは、コマンド発行時に SDRAM アドレスの bit 15-8 に対して駆動する値を指定します。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-24: DDRSCLSTART: DDL 自己校正ロジック開始レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/ 5	Bit 28/20/12/4	Bit 27/19/11/ 3	Bit 26/18/10/ 2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	R/W-0 SCLSTART	U-0 —	W-0 SCLLEN	U-0 —	R/W-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	R/W-0 —	R/W-0 —	R/W-0 —	R/W-0 —	U-0 —	U-0 —	R-0 SCLUBPASS <sup>(1)</sup>	R-0 SCLLBPASS <sup>(1)</sup>

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-29 **未実装**: 「0」として読み出し

bit 28 **SCLSTART**: 自己校正ロジック開始ビット

1 = 自己校正を開始する  
 0 = 自己校正を開始しない

bit 27 **未実装**: 「0」として読み出し

bit 26 **SCLLEN**: 自己校正ロジック イネーブルビット

1 = 動的自己校正ロジックを有効にする  
 0 = 動的自己校正ロジックを無効にする

bit 25-2 **未実装**: 「0」として読み出し

bit 1 **SCLUBPASS**: 自己校正ロジック上位データバイト ステータスビット<sup>(1)</sup>

1 = 上位データバイトの自己校正ロジックは合格した  
 0 = 上位データバイトの自己校正ロジックは失敗した

bit 0 **SCLLBPASS**: 自己校正ロジック下位データバイト ステータスビット<sup>(1)</sup>

1 = 下位データバイトの自己校正ロジックは合格した  
 0 = 下位データバイトの自己校正ロジックは失敗した

**Note 1:** 自己校正ロジック (SCL) が合格して完了すると、このビットはハードウェアによってセットされます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-25: DDRSCLLAT: DDL 自己校正ロジック レイテンシ レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/ 5	Bit 28/20/12/4	Bit 27/19/11/ 3	Bit 26/18/10/ 2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	R/W-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	R/W-0	R/W-1	R/W-1	R/W-0	U-0	U-0	R/W-1	R/W-0
DDRCLKDLY<3:0>				CAPCLKDLY<3:0>				

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット

U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア x = ビットは未知

bit 31-8 **未実装**: 「0」として読み出し

bit 7-4 **DDRCLKDLY<3:0>**: DDR クロック遅延ビット

推奨値は 4 です。

bit 3-0 **CAPCLKDLY<3:0>**: キャプチャ クロック遅延ビット

推奨値は 3 です。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 38-26: DDRSCLCFG0: DDR SCL コンフィグレーションレジスタ 0

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-1
	—	—	—	—	—	—	—	ODTCSW
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R/W-1	R/W-0	R/W-1	R/W-1	U-0	U-0	R/W-0	R/W-1
	RCASLAT<3:0>				—	—	DDR2	BURST8

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

- bit 31-25 **未実装:** 「0」として読み出し
- bit 24 **ODTCSW:** オンダイターミネーションチップセレクト書き込みビット  
 1 = ODT は、SCL による書き込み実行中に CS0 上の DRAM に対して ON になる  
 0 = ODT は、SCL による書き込み実行中に CS0 上の DRAM に対して OFF になる
- bit 23-8 **未実装:** 「0」として読み出し
- bit 7-4 **RCASLAT<3:0>:** CAS 読み出しレイテンシビット  
 DRAM の CAS 読み出しレイテンシ (クロックサイクル数) を表します。
- bit 3-2 **未実装:** 「0」として読み出し
- bit 1 **DDR2:** DDR2 ビット  
 1 = DDR2 は接続されている  
 0 = DDR2 は接続されていない
- bit 0 **BURST8:** PHY バースト 8 ビット  
 1 = SCL テストの実行中に DRAM はバースト 8 モードで動作する  
 0 = SCL テストの実行中に DRAM はバースト 4 モードで動作する

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-27: DDRSCLCFG1: DDR SCL コンフィグレーションレジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0
	—	—	—	DBLREFDLY	WCASLAT<3:0>			
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-1
	—	—	—	—	—	—	—	SCLCSEN

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-13 **未実装**: 「0」として読み出し

bit bit 12 **DBLREFDLY**: 基準遅延 2 倍ビット

肯定応答後に PHY が SCL 動作を遅延させる期間を基準の 2 倍にするかどうか指定します。基準の期間はハードウェアによって決まります。

1 = SCL 動作の遅延期間を 2 倍にする

0 = SCL 動作の遅延期間を 2 倍にしない

bit 11-8 **WCASLAT<3:0>**: CAS 書き込みレイテンシビット

DRAM の CAS 書き込みレイテンシ (クロックサイクル数) を表します。

bit 7-1 **未実装**: 「0」として読み出し

bit 0 **SCLCSEN**: SCL チップセレクトイネーブルビット

1 = チップセレクト 0 で SCL を実行する

0 = チップセレクト 0 で SCL を実行しない

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-28: DDRPHYPADCON: DDR PHY パッド制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	R/W-1	R/W-0	R/W-1	U-0	U-0	U-0	U-0
	—	PREAMBDLY<1:0>		R/VREN	—	—	—	—
23:16	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0
	DRVSTRPFET<3:0>				DRVSTRNFET<3:0>			
15:8	U-0	R/W-1	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	HALFRATE	WR CMDLY	—	—	—	NOEXTDLL	EOEN CLKCYC
7:0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1
	ODTPUCAL<1:0>		ODTPDCAL<1:0>		ADDC DRVDLY	DAT DRVSEL	ODTEN	ODTSEL

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31 **未実装:** 「0」として読み出し

bit 30-29 **PREAMBDLY<1:0>:** プリアンブル遅延ビット

このビットは、書き込みのためのプリアンブルの長さを制御します。

11 = 予約済み  
 10 = 1 サイクルのプリアンブル  
 01 = 1.5 サイクルのプリアンブル  
 00 = 2 サイクルのプリアンブル

bit 28 **R/VREN:** レシーバイネーブルビット

1 = 双方向 I/O 上のパッドレシーバを ON にする  
 0 = 双方向 I/O 上のパッドレシーバを OFF にする

bit 27-24 **未実装:** 「0」として読み出し

bit 23-20 **DRVSTRPFET<3:0>:** PFET 駆動強度ビット

パッド PFET ドライバ出力のインピーダンス調整を制御します。

1111 = 最大駆動強度

⋮

0000 = 最小駆動強度

bit 19-16 **DRVSTRNFET<3:0>:** NFET 駆動強度ビット

パッド NFET ドライバ出力のインピーダンス調整を制御します。

1111 = 最大駆動強度

⋮

0000 = 最小駆動強度

bit 15 **未実装:** 「0」として読み出し

bit 14 **HALFRATE:** ハーフレートビット

1 = コントローラ クロックは PHY に対してハーフレートで動作している  
 0 = コントローラ クロックは PHY に対してフルレートで動作している

bit 13 **WRCMDLY:** 書き込みコマンド遅延ビット

書き込みレイテンシ (WL) が偶数値である場合、このビットを「1」にセットする必要があります。

1 = 書き込みコマンドを遅延させる  
 0 = 書き込みコマンドを遅延させない

bit 12-10 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 38-28: DDRPHYPADCON: DDR PHY パッド制御レジスタ (続き)

- bit 9 **NOEXTDLL**: 外部 DLL 不使用ビット  
1 = 内部デジタル DLL を使う  
0 = 外部 DLL を使う
- bit 8 **EOENCLKCYC**: 追加出力イネーブルビット  
1 = 書き込みバースト後の追加クロックサイクルでの駆動パッド出力を有効にする  
0 = 書き込みバースト後の追加クロックサイクルでの駆動パッド出力を有効にしない
- bit 7-6 **ODTPUCAL<1:0>**: オンダイ ターミネーション プルアップ校正ビット  
11 = 最大 ODT インピーダンス  
.  
.  
00 = 最小 ODT インピーダンス
- bit 5-4 **ODTPFDCAL<1:0>**: オンダイ ターミネーション プルダウン校正ビット  
11 = 最大 ODT インピーダンス  
.  
.  
00 = 最小 ODT インピーダンス
- bit 3 **ADDCCRVSSEL**: アドレスおよび制御パッド駆動強度選択ビット  
1 = フル駆動  
0 = 60% 駆動
- bit 2 **DATDRVSEL**: データパッド駆動強度選択ビット  
1 = フル駆動  
0 = 60% 駆動
- bit 1 **ODTEN**: オンダイ ターミネーション イネーブルビット  
1 = ODT を有効にする  
0 = ODT を無効にする
- bit 0 **ODTSEL**: オンダイ ターミネーション選択ビット  
1 = 150 Ω のオンダイ ターミネーション  
0 = 75 Ω のオンダイ ターミネーション

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-29: DDRPHYDLLR: DDR PHY DLL 再校正レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
	DLYSTVAL<3:0>				—	DISRECALIB	RECALIBCNT<17:16>	
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RECALIBCNT<15:8>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RECALIBCNT<7:0>							
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット 0 = ビットはクリア

x = ビットは未知

bit 31-28 **DLYSTVAL<3:0>**: 遅延開始値ビット

デジタル DLL マスタ遅延ラインの開始値です。値「0011」を推奨します。

bit 27 **未実装**: 「0」として読み出し

bit 26 **DISRECALIB**: 再校正ディセーブル ビット

1 = 初回校正後にデジタル DLL を再校正しない

0 = RECALIBCNT<17:0> ビットの値に従ってデジタル DLL を再校正する

bit 25-8 **RECALIBCNT<17:0>**: 再校正カウントビット

デジタル DLL の再校正の期間を [256 × PHY クロックサイクル数] 単位で指定します。

bit 7-0 **未実装**: 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 38-30: DDRPHYCLKDLY: DDR クロック デルタ遅延レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	U-0 —	R-0 SCLUBPASS <sup>(1)</sup>	R-0 SCLLBPASS <sup>(1)</sup>	U-0 —	R/W-0 —	R/W-0 —	R/W-0 —
						CLKDLYDELTA<2:0>		

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット

U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-6 **未実装:** 「0」として読み出し

bit 5 **SCLUBPASS:** 自己校正ロジック上位データバイト ステータスビット

1 = 上位データバイトの自己校正ロジックは合格した

0 = 上位データバイトの自己校正ロジックは失敗した

bit 4 **SCLLBPASS:** 自己校正ロジック下位データバイト ステータスビット

1 = 下位データバイトの自己校正ロジックは合格した

0 = 下位データバイトの自己校正ロジックは失敗した

bit 3 **未実装:** 「0」として読み出し

bit 2-0 **CLKDLYDELTA<2:0>:** DDR クロック遅延デルタビット

これらのビットは、バイトレーンごとに設定された SCL レイテンシを示します。

111 = 7 DDR クロック

110 = 6 DDR クロック

⋮

⋮

⋮

000 = 0 DDR クロック

**Note:** これらのビットは SCL ロジックによって自動的に設定されますが、ユーザが設定する事も可能です。このビットは、SCL が撤退する場合に特に便利です。

**Note 1:** これらのビットは、SCLLBPASS (DDRSCLESTART<0>) および SCLUBPASS (DDRSCLESTART<0>) ビットと同じステータスを示します。



# PIC32MZ グラフィック (DA) ファミリ

## 39.0 セキュア デジタルホスト コントローラ (SDHC)

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 57. Secure Digital Host Controller (SDHC)』(DS60001334) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

SDHC モジュールは、32 ビットシステム バスマスタおよびスレーブ インターフェイスを使って、ホストシステムとデバイス側の標準カード インターフェイスを接続します。

コアは DMA コントローラを内蔵するため、CPU の介入を必要とせずに、システムメモリと SD/SDIO/eMMC カードの間でデータを自動的に転送できます。

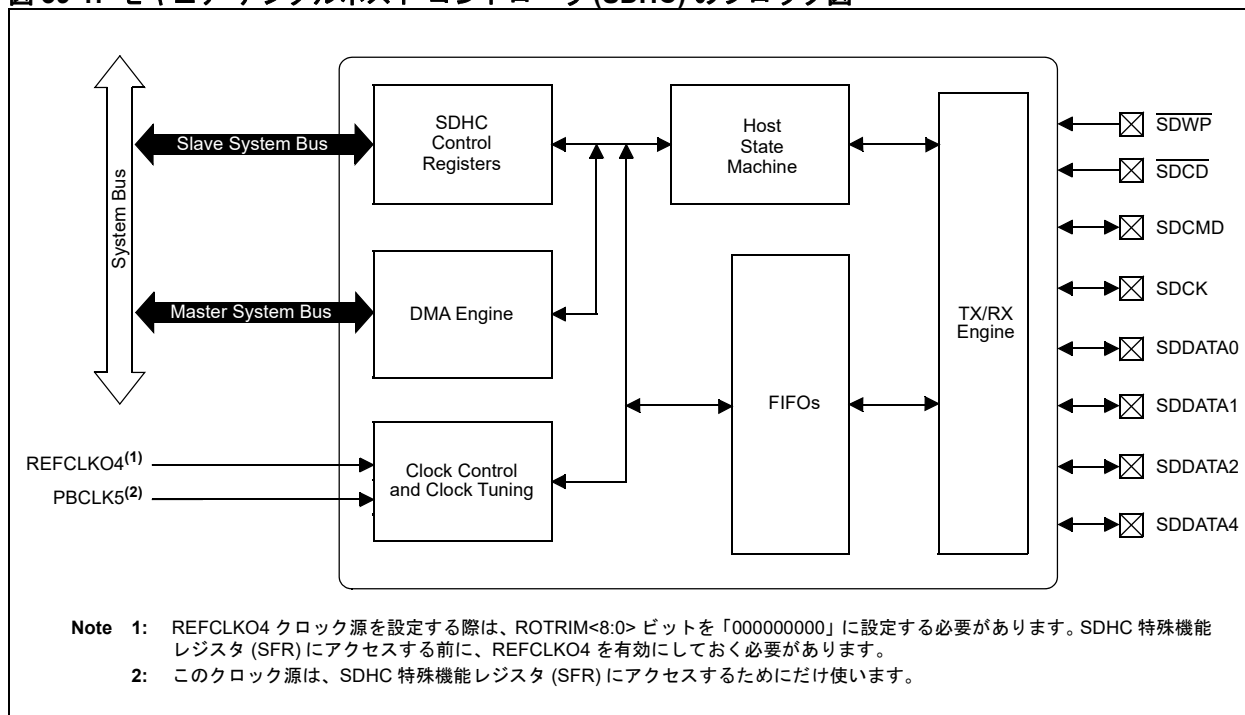
SDHC モジュールの特長は以下の通りです。

- 以下の SD アソシエーション仕様に準拠：
  - SD Host Controller Simplified Specification (version 2.00)
  - Physical Layer Simplified Specification (version 2.00)
  - SDIO Simplified Specification (version 2.00)
- eMMC 規格 : JESD84-A441
- ノーマルスピード モードとハイスピード モード
- 1 ビットまたは 4 ビットのデータ転送
- クロック分周器を内蔵
- PIO および ADMA モードのデータ転送
- 3.3V 動作
- 割り込みをサポート
- ブロックギャップでの停止

図 39-1 に、SDHC モジュールのブロック図を示します。

**Note:** ADMA モードにおける送信および受信バッファのアドレスは、ワード境界に整列させる必要があります。複数のディスクリプタを使って 1 つのブロックを転送する場合、最後のディスクリプタを除く全てのディスクリプタの転送サイズは 4 の倍数である必要があります。

図 39-1: セキュア デジタルホスト コントローラ (SDHC) のブロック図



### 39.1 制御レジスタ

表 39-1: SDHC SFR の一覧

制御アドレ ス (BF8E_#)	レジスタ名	ビットレンジ	Bit																ト ク ン タ 全	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0		
C004	SDHC BLKCON	31:16	BCOUNT<15:0>																0000	
		15:0	BSIZE<9:0>																0000	
C008	SDHC ARG	31:16	ARG<31:16>																0000	
		15:0	ARG<15:0>																0000	
C00C	SDHC MODEM <sub>0</sub> de 1	31:16	CIDX<5:0>								CTYPE<1:0>			DPSSEL	CIDXCEN	CCRCCEN	—		RESPTYPE<1:0>	0000
		15:0	—								—			BSEL	DTXDSEL	ACEN<1:0>		BCEN	DMAEN	0000
C010	SHDC RESP0	31:16	RESP<31:16>																0000	
		15:0	RESP<15:0>																0000	
C014	SHDC RESP1	31:16	RESP<31:16>																0000	
		15:0	RESP<15:0>																0000	
C018	SHDC RESP2	31:16	RESP<31:16>																0000	
		15:0	RESP<15:0>																0000	
C01C	SHDC RESP3	31:16	RESP<31:16>																0000	
		15:0	RESP<15:0>																0000	
C020	SHDC DATA	31:16	DATA<31:16>																0000	
		15:0	DATA<15:0>																0000	
C024	SDHC STAT1	31:16	—	—	—	—	—	—	—	CMDSLVL	DATA3SLVL	DATA2SLVL	DATA1SLVL	DATA0SLVL	WPSLVL	CDSLVL	CARDST	CARDINS	0000	
		15:0	—	—	—	—	BREN	BWEN	RDACTIVE	WRACTIVE	—	—	—	—	—	DLACTIVE	CINHDT	CINHCMD	0000	
C028	SDHC CON1	31:16	—	—	—	—	—	WKONREM	WKONINS	WKONINT	—	—	—	—	INTBG	RDWTCON	CONTREQ	SBGREQ	0000	
		15:0	—	—	—	—	—	—	—	SDBP	CDSSSEL	CDTLVL	—	DMASEL<1:0>		HSEN	DTXWIDTH	—	0000	
C02C	SDHC CON2	31:16	—	—	—	—	—	SWRDATA	SWRCMD	SWRALL	—	—	—	—	DTC<3:0>				0000	
		15:0	SDCLKDIV<7:0>								—	—	—	—	—	SDCLKEN	ICLKSTABLE	ICLKEN	0000	
C030	SDHC INTSTAT	31:16	—	—	—	—	—	ADEIF	ACEIF	CLEIF	DEBEIF	DCRCEIF	DTOEIF	CIDXEIF	CEBEIF	CCRCEIF	CTOEIF	0000		
		15:0	EIF	—	—	—	—	—	CARDIF	CARDRIF	CARDIIF	BRRDYIF	BWRDYIF	DMAIF	BGIF	TXCIF	CCIF	0000		
C034	SDHC INTEN	31:16	—	—	—	—	—	ADEIE	AACEIE	CLEIE	DEBEIE	DCRCEIE	DTOEIE	CIDXEIE	CDEBEIE	CCRCEIE	CTOEIE	0000		
		15:0	FTZIE	—	—	—	—	—	CARDIE	CARDRIE	CARDIIE	BRRDYIE	BWRDYIE	DMAIE	BGIE	TXCIE	CCE	0000		
C038	SDHC INTSEN	31:16	—	—	—	—	—	ADEISE	ACEISE	CLEISE	DEBEISE	DCRCEISE	DTOEISE	CIDXEISE	CEBEISE	CEBEISE	CCRCEISE	0000		
		15:0	FTZEISE	—	—	—	—	—	CARDISE	CARDRISE	CARDIISE	BRRDYISE	BWRDYISE	DMAISE	BGISE	TXCISE	CCISE	0000		
C03C	SDHC STAT2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	—	—	—	—	—	—	—	CNISSE	—	—	—	ACIDXE	ACEBE	ACCRC	ACTOE	ACNEXEC	0000	
C040	SDHC CAP	31:16	SLOTTYPE<1:0>		ASYNCINT	—	—	—	VOLT3V3	SRESUME	—	HISPEED	—	ADMA2	—	MBLEN<1:0>		0000		
		15:0	BASECLK<7:0>								TOCLKU	—	TOCLKFREQ<5:0>						0000	
C048	SDHC MAXCAP	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000		
		15:0	—	—	—	—	—	—	—	MC3V3<7:0>										0000
C050	SDHC FE	31:16	—	—	—	—	—	FEAE	FEACE	FECLE	FEDEBE	FEDCRCE	FEDTOE	FEIDX	FECEBE	FECRC	FECTOE	0000		
		15:0	—	—	—	—	—	—	—	FECNIACE	—	—	FEACIDX	FEACEBE	FEACRC	FEACTOE	FEACNEE	0000		

凡例: — = 未実装、「0」として読み出し

表 39-1: SDHC SFR の一覧 ( 続き )

アドレス (BF8E_#)	レジスタ名	ビットレンジ	Bit															全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
C054	SDHC AESTAT	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ALMERR	AERRST<1:0>	0000
C058	SDHC AADDR	31:16	ADDR<31:16>															0000	
		15:0	ADDR<15:0>															0000	

凡例： — = 未実装、「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 39-1: SDHCBLKCON: SDHC ブロック制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BCOUNT<15:8> <sup>(1)</sup>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BCOUNT<7:0> <sup>(1)</sup>								
15:8	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—							BSIZE<9:8> <sup>(2)</sup>	
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BSIZE<7:0> <sup>(2)</sup>								

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **BCOUNT<31:0>**: 実行中転送のブロック カウントビット<sup>(1)</sup>

これらのビットはブロックの数を表します。ソフトウェアはこのビットを 1 ~ 65,535 の値に設定します。SDHCは1つのブロックを転送するたびに値を1つデクリメントし、この値が0になると転送を停止します。  
 0xFFFF = 65,535 ブロック  
 0x0002 = 2 ブロック  
 0x0001 = 1 ブロック  
 0x0000 = このカウントビットのカウントを停止する

bit 15-10 **未実装**: 「0」として読み出し

bit 9-0 **BSIZE<9:0>**: 転送ブロックサイズ ビット<sup>(2)</sup>

これらのビットは CMD17、CMD18、CMD24、CMD25、CMD53 向けデータ転送のブロックサイズを指定します。  
 0x200 = 512 バイト  
 0x1FF = 511 バイト  
 .  
 .  
 .  
 0x002 = 2 バイト  
 0x001 = 1 バイト  
 0x000 = データを転送しない

- Note 1:** これらのビットは、BCEN ビット (SDHCMODE<1>) が「1」にセットされている時にのみ使われ、マルチブロック転送に対してのみ有効です。BSEL ビット (SDHCMODE<5>) が「0」にクリアされている場合、BCOUNT<15:0> ビットを設定する必要はありません。
- 2:** これらのビットには、トランザクションが進行中ではない時にのみアクセスできます。転送中のこれらのビットへの読み出し動作は無効値を返し、これらのビットへの書き込み動作は無視されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-2: SDHCARG: SDHC 引数レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ARG<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ARG<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ARG<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ARG<7:0>								

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-0 ARG<31:0>: コマンド引数ビット

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-3: SDHCMODE: SDHC モードレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CIDX<5:0> <sup>(1)</sup>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
	CTYPE<1:0>		DPSEL	CIDXCEN <sup>(2)</sup>	CCRCEN <sup>(3)</sup>	—	RESPTYPE<1:0>	
15:8	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
7:0	U-0 —	U-0 —	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
			BSEL	DTXDSEL	ACEN<1:0>		BCEN	DMAEN

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット

U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-30 **未実装**: 「0」として読み出し

bit 29-24 **CIDX<5:0>**: コマンドインデックス ビット<sup>(1)</sup>  
これらのビットはコマンド番号 (0 ~ 63) を表します。

bit 23-22 **CTYPE<1:0>**: コマンドタイプ ビット

- 11 = 中止
- 10 = 再開
- 01 = 一時停止
- 00 = 通常

bit 21 **DPSEL**: データあり / なし選択ビット

- 1 = データあり
- 0 = データなし

bit 20 **CIDXCEN**: コマンドインデックス チェック イネーブルビット<sup>(2)</sup>

- 1 = コマンド インデックス チェックを有効にする
- 0 = コマンド インデックス チェックを無効にする

bit 19 **CCRCEN**: コマンド CRC チェック イネーブルビット<sup>(3)</sup>

- 1 = コマンド CRC チェックを有効にする
- 0 = コマンド CRC チェックを無効にする

bit 18 **未実装**: 「0」として読み出し

bit 17-16 **RESPTYPE<1:0>**: 応答タイプ選択ビット

- 11 = 応答長は 48 ( 応答後に busy をチェック )
- 10 = 応答長は 48
- 01 = 応答長は 136
- 00 = 応答しない

bit 15-6 **未実装**: 「0」として読み出し

bit 5 **BSEL**: マルチ / シングル ブロック選択ビット

- 1 = マルチブロック (DAT ラインを使って複数の転送コマンドを発行する場合にこのビットをセット)
- 0 = シングルブロック

**Note 1**: 「SD Host Controller Simplified Specification」(version 2.00) 内のコマンド フォーマットの bit 45-40 を参照してください。

**2**: このビットが「1」にセットされている場合、SDHC は応答内のインデックス フィールドの値が CIDX<5:0> ビットと同じかどうか確認し、同じではない場合にコマンド インデックス エラーを報告します。

**3**: このビットが「1」にセットされている場合、SDHC は応答内の CRC フィールドを確認し、CRC エラーを検出した場合にコマンド CRC エラーを報告します。

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 39-3: SDHCMODE: SDHC モードレジスタ (続き)

- bit 4     **DTXDSEL:** データ転送方向選択ビット  
          1 = 読み出し (カードから SDHC への転送)  
          0 = 書き込み (SDHC からカードへの転送)
- bit 3-2   **ACEN<1:0>:** 自動 CMD12 イネーブルビット  
          自動 CMD12 は、マルチブロック読み書き動作を停止するために使います。  
          11 = 予約済み  
          10 = 予約済み  
          01 = 自動 CMD12 を有効にする  
          00 = 自動 CMD12 を無効にする
- bit 1     **BCEN:** ブロックカウント イネーブルビット  
          1 = ブロックカウントを有効にする  
          0 = ブロックカウントを無効にする
- bit 0     **DMAEN:** DMA イネーブルビット  
          1 = データ転送に DMA (ADMA) を使う  
          0 = データ転送に CPU を使う

- Note 1:** 「SD Host Controller Simplified Specification」(version 2.00) 内のコマンド フォーマットの bit 45-40 を参照してください。
- 2:** このビットが「1」にセットされている場合、SDHC は応答内のインデックス フィールドの値が CIDX<5:0> ビットと同じかどうか確認し、同じではない場合にコマンド インデックス エラーを報告します。
- 3:** このビットが「1」にセットされている場合、SDHC は応答内の CRC フィールドを確認し、CRC エラーを検出した場合にコマンド CRC エラーを報告します。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 39-4: SDHCRESPx: SDHC 応答レジスタ「x」(x = 0 ~ 3)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RESP<31:24>								
23:16	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RESP<23:16>								
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RESP<15:8>								
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RESP<7:0>								

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア        x = ビットは未知

bit 31-0 **RESP<31:0>**: 応答ビット

これらのビットは、「SD Host Controller Simplified Specification」(version 2.00) 内で定義されている Responses [31:0] のビット位値を示します。完全なビット定義については表 39-2 を参照してください。

表 39-2: 各応答タイプの応答ビット定義

応答タイプ (Note 1 参照)	応答の意味	応答レジスタ
R1, R1b (通常の応答)	カードステータス	SDHCRESP0<31:0>
R1b (自動 CMD12 応答)	自動 CMD12 のカードステータス	SDHCRESP3<31:0>
R2 (CID、CSD レジスタ)	CID または CSD レジスタ	SDHCRESP0<31:0> SDHCRESP1<31:0> SDHCRESP2<31:0> SDHCRESP3<31:0>
R3 (OCR レジスタ)	メモリ向け OCR レジスタ	SDHCRESP0<31:0>
R4 (OCR レジスタ)	I/O 等向け OCR レジスタ	SDHCRESP0<31:0>
R5, R5b	SDIO 応答	SDHCRESP0<31:0>
R6 (発行された RCA 応答)	新しく発行された RCA<31:16> 等	SDHCRESP0<31:0>

**Note 1:** その他の情報については、「SD Host Controller Simplified Specification」(version 2.00)、「Physical Layer Simplified Specification」(version 2.00)、「SDIO Simplified Specification」(version 2.00) を参照してください。これらの文書は、以下の SD アソシエーションのウェブサイトからダウンロードできます。  
[http://www.sdcard.org/downloads/pls/simplified\\_specs/archive/index.html](http://www.sdcard.org/downloads/pls/simplified_specs/archive/index.html)



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-5: SDHCDATA: SDHC データレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DATA<31:24>								
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DATA<23:16>								
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DATA<15:8>								
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DATA<7:0>								

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-0 **DATA<31:0>**: バッファデータ ビット

これらのビットは、内部データバッファの bit 31-0 にアクセスするために使います。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-6: SDHCSTAT1: SDHC ステータス レジスタ 1

ビットレンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-x, HC
	—	—	—	—	—	—	—	CMDSLVL
23:16	R-x, HC	R-x, HC	R-x, HC	R-x, HC	R-x, HC	R-x, HC	R-x, HC	R-x, HC
	DATA3SLVL	DATA2SLVL	DATA1SLVL	DATA0SLVL	WPSLVL	CDSLVL	CARDST	CARDINS
15:8	U-0	U-0	U-0	U-0	R-0, HC	R-0, HC	R-0, HC	R-0, HC
	—	—	—	—	BREN	BWEN	RDACTIVE	WRACTIVE
7:0	U-0	U-0	U-0	U-0	U-0	R-0, HC	R-0, HC	R-0, HC
	—	—	—	—	—	DLACTIVE	CINH DAT	CINH CMD

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット
-n = POR 時の値	1 = ビットはセット
	U = 未実装、「0」として読み出し
	0 = ビットはクリア
	x = ビットは未知

- bit 31-25 **未実装:** 「0」として読み出し
- bit 24 **CMDSLVL:** コマンドライン信号レベルビット
  - 1 = CMD ラインは HIGH
  - 0 = CMD ラインは LOW
- bit 23 **DATA3SLVL:** DATA3 信号レベルビット
  - 1 = DAT3 ラインは HIGH
  - 0 = DAT3 ラインは LOW
- bit 22 **DATA2SLVL:** DATA2 信号レベルビット
  - 1 = DAT2 ラインは HIGH
  - 0 = DAT2 ラインは LOW
- bit 21 **DATA1SLVL:** DATA1 信号レベルビット
  - 1 = DAT1 ラインは HIGH
  - 0 = DAT1 ラインは LOW
- bit 20 **DATA0SLVL:** DATA0 信号レベルビット
  - 1 = DAT0 ラインは HIGH
  - 0 = DAT0 ラインは LOW
- bit 19 **WPSLVL:** 書き込み保護信号レベルビット
  - 1 = 書き込み保護は無効
  - 0 = 書き込み保護は有効
- bit 18 **CDSLVL:** カード検出信号レベルビット
  - 1 = カードは存在しない
  - 0 = カードは存在する
- bit 17 **CARDST:** カードステート安定ビット
  - 1 = カードは存在しない (または挿入されていない)
  - 0 = リセットまたはデバウンス中
- bit 16 **CARDINS:** カード挿入ビット
  - 1 = カードは挿入されている
  - 0 = リセットまたはデバウンス中、またはカードは存在しない
- bit 15-12 **未実装:** 「0」として読み出し
- bit 11 **BREN:** バッファ読み出しイネーブルビット
  - 1 = バッファ読み出しは有効
  - 0 = バッファ読み出しは無効

**Note:** このレジスタは、エラーからの復帰用とデバッグ用に使います。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-6: SDHCSTAT1: SDHC ステータス レジスタ 1 ( 続き )

- bit 10    **BWEN:** バッファ書き込みイネーブルビット  
1 = バッファ書き込みは有効  
0 = バッファ書き込みは無効
- bit 9    **RDACTIVE:** 読み出し転送アクティブビット  
1 = データを転送中  
0 = 有効データなし
- bit 8    **WRACTIVE:** 書き込み転送アクティブビット  
1 = データを転送中  
0 = 有効データなし
- bit 7-3   **未実装:** 「0」として読み出し
- bit 2    **DLACTIVE:** DAT ライン アクティブビット  
1 = DAT ラインはアクティブ  
0 = DAT ラインは非アクティブ
- bit 1    **CINH DAT:** コマンド抑止 (DAT) ビット  
1 = DAT ラインを使うコマンドは発行禁止  
0 = DAT ラインを使うコマンドは発行可能
- bit 0    **CINH CMD:** コマンド抑止 (CMD) ビット  
1 = コマンドは発行禁止  
0 = CMD ラインを使うコマンドだけ発行可能

**Note:** このレジスタは、エラーからの復帰用とデバッグ用に使います。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 39-7: SDHCCON1: SDHC 制御レジスタ 1

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	WKONREM	WKONINS	WKONINT
23:16	U-0	U-0	U-0	U-0	R/W-0	R/W-0	HC, R/W-0	R/W-0
	—	—	—	—	INTBG	RDWTCON	CONTREQ	SBGREQ
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
	—	—	—	—	—	—	—	SDBP
7:0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
	CDSSEL	CDTLVL	—	DMASEL<1:0>		HSEN	DTXWIDTH	—

**凡例:**

HC = ハードウェアでクリア

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31-27 **未実装:** 「0」として読み出し

bit 26 **WKONREM:** SD カード取り外し時復帰イベント イネーブルビット

1 = 復帰イベントを有効にする

0 = 復帰イベントを有効にする

bit 25 **WKONINS:** SD カード挿入時復帰イベント イネーブルビット

1 = 復帰イベントを有効にする

0 = 復帰イベントを有効にする

bit 24 **WKONINT:** SD カード割り込み時復帰イベント イネーブルビット

1 = 復帰イベントを有効にする

0 = 復帰イベントを有効にする

bit 23-20 **未実装:** 「0」として読み出し

bit 19 **INTBG:** ブロックギャップでの割り込みビット

1 = 割り込みを有効にする

0 = 割り込みを無効にする

bit 18 **RDWTCON:** 読み出し待機制御ビット

1 = 読み出し待機制御を有効にする

0 = 読み出し待機制御を無効にする

bit 17 **CONTREQ:** 再開要求ビット

STOPREQ が「1」にセットされている場合、このビットへの書き込みは無視されます。

1 = 再開する

0 = 何もしない

bit 16 **SBGREQ:** ブロックギャップでの停止要求ビット

1 = 停止する

0 = 転送する

bit 15-9 **未実装:** 「0」として読み出し

bit 8 **SDBP:** SD バスパワービット

1 = バスパワーを ON にする

0 = バスパワーを OFF にする

bit 7 **CDSSEL:** カード検出信号選択ビット

1 = カード検出試験レベル (CDTLVL) を選択する (検査用)

0 = SDCDx を選択する (通常動作)

bit 6 **CDTLVL:** カード検出試験レベルビット

1 = カード挿入

0 = カード非挿入

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 39-7: SDHCCON1: SDHC 制御レジスタ 1 ( 続き )

- bit 5 **未実装**: 「0」として読み出し
- bit 4-3 **DMASEL<1:0>**: DMA 選択ビット
- 11 = 予約済み
  - 10 = 32 ビットアドレス ADMA2 を選択する
  - 01 = 予約済み
  - 00 = 予約済み
- bit 2 **HSEN**: ハイスピード イネーブルビット
- 1 = ハイスピード モードを有効にする
  - 0 = ノーマルスピード モードを有効にする
- bit 1 **DTXWIDTH**: データ転送幅ビット
- 1 = 4 ビットモード
  - 0 = 1 ビットモード
- bit 0 **未実装**: 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-8: SDHCCON2: SDHC 制御レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0, HC	R/W-0, HC	R/W-0, HC
	—	—	—	—	—	SWRDATA	SWRCMD	SWRALL
23:16	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	DTCO<3:0>			
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SDCLKDIV<7:0>							
7:0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	SDCLKEN	ICLK STABLE	ICLKEN

<b>凡例:</b>		HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

bit 31-27 **未実装:** 「0」として読み出し

bit 26 **SWRDATA:** DATA ライン ソフトウェア リセットビット

1 = DMA とデータロジックの一部をリセットする

0 = 動作を継続する

bit 25 **SWRCMD:** CMD ライン ソフトウェア リセットビット

1 = 現在のステート、割り込みステータス レジスタ、CMD ビットをクリアする

0 = 動作を継続する

bit 24 **SWRALL:** 全ソフトウェア リセットビット

1 = リセットコマンドを発行し、SD カードを再初期化する

0 = 分周クロックモードを選択する

bit 23-20 **未実装:** 「0」として読み出し

bit 19-16 **DTCO<3:0>:** データ タイムアウト カウンタ値ビット

1111 = 予約済み

1110 = タイムアウト クロック  $\times 2^{27}$

.

.

0001 = タイムアウト クロック  $\times 2^{14}$

0000 = タイムアウト クロック  $\times 2^{13}$

bit 15-8 **SDCLKDIV<7:0>:** SDCLK 分周比選択ビット

8 ビット分周クロックモードを選択している場合:

0x80 - ベースクロックを 256 分周する

0x40 - ベースクロックを 128 分周する

0x20 - ベースクロックを 64 分周する

0x10 - ベースクロックを 32 分周する

0x08 - ベースクロックを 16 分周する

0x04 - ベースクロックを 8 分周する

0x02 - ベースクロックを 4 分周する

0x01 - ベースクロックを 2 分周する

0x00 - ベースクロック

bit 7-3 **未実装:** 「0」として読み出し

bit 2 **SDCLKEN:** SD クロック イネーブルビット

1 = SD クロックを有効にする

0 = SD クロックを無効にする

bit 1 **ICLKSTABLE:** 内部クロック安定ビット

1 = 内部クロックの準備は完了している

0 = 内部クロックの準備は未完了

bit 0 **ICLKEN:** 内部クロック イネーブルビット

1 = 発振

0 = 停止

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 39-9: SDHCINTSTAT: SDHC 割り込みステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HC	R/W-0, HC
	—	—	—	—	—	—	ADEIF	ACEIF
23:16	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
	CLEIF	DEBEIF	DCRCEIF	DTOEIF	CIDXEIF	CEBEIF	CCRCEIF	CTOEIF
15:8	R-0, HC	U-0	U-0	U-0	U-0	U-0	U-0	R-0, HC
	EIF	—	—	—	—	—	—	CARDIF
7:0	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
	CARDRIF	CARDIIF	BRRDYIF	BWRDYIF	DMAIF	BGIF	TXCIF	CEIF

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット
-n = POR 時の値	U = 未実装、「0」として読み出し
	1 = ビットはセット
	0 = ビットはクリア
	x = ビットは未知

bit 31-26 **未実装:** 「0」として読み出し

bit 25 **ADEIF:** ADMA エラー割り込みフラグビット

1 = ADMA エラーが発生した  
0 = ADMA エラーは発生していない

bit 24 **ACEIF:** 自動 CMD12 エラー割り込みフラグビット

1 = 自動 CMD12 エラーが発生した  
0 = 自動 CMD12 エラーは発生していない

bit 23 **CLEIF:** 電流制限エラー割り込みフラグビット

1 = 電流制限エラーが発生した  
0 = 電流制限エラーは発生していない

bit 22 **DEBEIF:** データ エンドビット エラー割り込みフラグビット

1 = データ エンドビット エラーが発生した  
0 = データ エンドビット エラーは発生していない

bit 21 **DCRCEIF:** データ CRC エラー割り込みフラグビット

1 = データ CRC エラーが発生した  
0 = データ CRC エラーは発生していない

bit 20 **DTOEIF:** データ タイムアウト エラー割り込みフラグビット

1 = データ タイムアウト エラーが発生した  
0 = データ タイムアウト エラーは発生していない

bit 19 **CIDXEIF:** コマンド インデックス エラー割り込みフラグビット

1 = コマンド インデックス エラーが発生した  
0 = コマンド インデックス エラーは発生していない

bit 18 **CEBEIF:** コマンド エンドビット エラー割り込みフラグビット

1 = エンドビット エラーが発生した  
0 = エンドビット エラーは発生していない

bit 17 **CCRCEIF:** コマンド CRC エラー割り込みフラグビット

1 = コマンド CRC エラーが発生した  
0 = コマンド CRC エラーは発生していない

bit 16 **CTOEIF:** コマンド タイムアウト エラー割り込みフラグビット

1 = コマンド タイムアウト エラーが発生した  
0 = コマンド タイムアウト エラーは発生していない

bit 15 **EIF:** エラー割り込みフラグビット

このビットは、このレジスタ内の bit 0 ~ 9 のいずれか (または全て) のビットがセットされた場合にセットされます。  
1 = エラーを検出した  
0 = エラーは検出していない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 39-9: SDHCINTSTAT: SDHC 割り込みステータス レジスタ ( 続き )

- bit 14-9 **未実装**: 「0」として読み出し
- bit 8 **CARDIF**: カード割り込みステータスビット  
1 = カード割り込みが発生した  
0 = カード割り込みは発生していない
- bit 7 **CARDRIF**: カード取り外し割り込みフラグビット  
1 = カードが取り外された  
0 = カードの状態は安定している、またはデバウンス中
- bit 6 **CARDIIF**: カード挿入割り込みフラグビット  
1 = カードが挿入された  
0 = カードの状態は安定している、またはデバウンス中
- bit 5 **BRRDYIF**: バッファ読み出し準備完了割り込みフラグビット  
1 = バッファの読み出し準備は完了している  
0 = バッファの読み出し準備は未完了
- bit 4 **BWRDYIF**: バッファ書き込み準備完了割り込みフラグビット  
1 = バッファの書き込み準備は完了している  
0 = バッファの書き込み準備は未完了
- bit 3 **DMAIF**: DMA 割り込みステータスビット  
1 = DMA 割り込みが発生した  
0 = DMA 割り込みは発生していない
- bit 2 **BGIF**: ブロックギャップ割り込みフラグビット  
1 = ブロックギャップでトランザクションが停止した  
0 = ブロックギャップ イベントは発生していない
- bit 1 **TXEIF**: 転送完了割り込みフラグビット  
1 = 転送は完了した  
0 = 転送は未完了
- bit 0 **CEIF**: コマンド完了割り込みフラグビット  
1 = コマンドは完了した  
0 = コマンドは未完了



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-10: SDHCINTEN: SDHC 割り込みフラグ イネーブル レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HC	R/W-0, HC
	—	—	—	—	—	—	ADEFIE	ACEFIE
23:16	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
	CLEFIE	DEBEFIE	DCRCEFIE	DTOEFIE	CIDXFIE	CDEBEFIE	CCRCEFIE	CTOEFIE
15:8	R-0, HC	U-0	U-0	U-0	U-0	U-0	U-0	R-0, HC
	FTZIE	—	—	—	—	—	—	CARDIE
7:0	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
	CARDRIE	CARDIIE	BRRDYIE	BWRDYIE	DMAIE	BGIE	TXEIE	CEIE

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 21-26 **未実装:** 「0」として読み出し

bit 25 **ADEFIE:** ADMA エラー割り込みフラグ イネーブルビット

- 1 = ADMA エラー割り込みフラグを有効にする
- 0 = ADMA エラー割り込みフラグをマスクする

bit 24 **ACEFIE:** 自動 CMD12 エラー割り込みフラグ イネーブルビット

- 1 = 自動 CMD12 エラー割り込みフラグを有効にする
- 0 = 自動 CMD12 エラー割り込みフラグをマスクする

bit 23 **CLEFIE:** 電流制限エラー割り込みフラグ イネーブルビット

- 1 = 電流制限エラー割り込みフラグを有効にする
- 0 = 電流製源エラー割り込みフラグをマスクする

bit 22 **DEBEFIE:** データ エンドビット エラー割り込みフラグ イネーブルビット

- 1 = データ エンドビット エラー割り込みフラグを有効にする
- 0 = データ エンドビット エラー割り込みフラグをマスクする

bit 21 **DCRCEFIE:** データ CRC エラー割り込みフラグ イネーブルビット

- 1 = データ CRC エラー割り込みフラグを有効にする
- 0 = データ CRC エラー割り込みフラグをマスクする

bit 20 **DTOEFIE:** データ タイムアウト エラー割り込みフラグ イネーブルビット

- 1 = データ タイムアウト エラー割り込みフラグを有効にする
- 0 = データ タイムアウト エラー割り込みフラグをマスクする

bit 19 **CIDXFIE:** コマンド インデックス エラー割り込みフラグ イネーブルビット

- 1 = コマンド インデックス エラー割り込みフラグを有効にする
- 0 = コマンド インデックス エラー割り込みフラグをマスクする

bit 18 **CDEBEFIE:** コマンド エンドビット エラー割り込みフラグ イネーブルビット

- 1 = コマンド エンドビット エラー割り込みフラグを有効にする
- 0 = コマンド エンドビット エラー割り込みフラグをマスクする

bit 17 **CCRCEFIE:** コマンド CRC エラー割り込みフラグ イネーブルビット

- 1 = コマンド CRC エラー割り込みフラグを有効にする
- 0 = コマンド CRC エラー割り込みフラグをマスクする

bit 16 **CTOEFIE:** コマンド タイムアウト エラー割り込みフラグ イネーブルビット

- 1 = コマンド タイムアウト エラー割り込みフラグを有効にする
- 0 = コマンド タイムアウト エラー割り込みフラグをマスクする

bit 15 **FTZIE:** ゼロ固定割り込みフラグ イネーブルビット

- このビットは、このレジスタ内の bit 0 ~ 9 のいずれか (または全て) のビットがセットされた場合にセットされます。
- 1 = エラーを検出した
- 0 = エラーは検出していない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 39-10: SDHCINTEN: SDHC 割り込みフラグ イネーブル レジスタ (続き)

bit 14-9 未実装: 「0」として読み出し

bit 8 **CARDIE:** カード割り込みフラグ イネーブルビット

- 1 = カード割り込みフラグを有効にする
- 0 = カード割り込みフラグをマスクする

bit 7 **CARDRIE:** カード取り外し割り込みフラグ イネーブルビット

- 1 = カード取り外し割り込みフラグを有効にする
- 0 = カード取り外し割り込みフラグをマスクする

bit 6 **CARDIIE:** カード挿入割り込みフラグ イネーブルビット

- 1 = カード挿入割り込みフラグを有効にする
- 0 = カード挿入割り込みフラグをマスクする

bit 5 **BRRDYIE:** バッファ読み出し準備完了割り込みフラグ イネーブルビット

- 1 = バッファ読み出し準備完了割り込みフラグを有効にする
- 0 = バッファ読み出し準備完了割り込みフラグをマスクする

bit 4 **BWRDYIE:** バッファ書き込み準備完了割り込みフラグ イネーブルビット

- 1 = バッファ書き込み準備完了割り込みフラグを有効にする
- 0 = バッファ書き込み準備完了割り込みフラグをマスクする

bit 3 **DMAIE:** DMA 割り込みフラグ イネーブルビット

- 1 = DMA 割り込みフラグを有効にする
- 0 = DMA 割り込みフラグをマスクする

bit 2 **BGIE:** ブロックギャップ割り込みフラグ イネーブルビット

- 1 = ブロックギャップ イベント割り込みフラグを有効にする
- 0 = ブロックギャップ イベント割り込みフラグをマスクする

bit 1 **TXEIE:** 転送完了割り込みフラグ イネーブルビット

- 1 = 転送完了割り込みフラグを有効にする
- 0 = 転送完了割り込みフラグをマスクする

bit 0 **CEIE:** コマンド完了割り込みフラグ イネーブルビット

- 1 = コマンド完了割り込みフラグを有効にする
- 0 = コマンド完了割り込みフラグをマスクする

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-11: SDHCINTSEN: SDHC 割り込み信号イネーブル レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HC	R/W-0, HC
	—	—	—	—	—	—	ADEISE	ACEISE
23:16	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
	CLEISE	DEBEISE	DCRCEISE	DTOEISE	CIDXEISE	CEBEISE	CCRCEISE	CTOEISE
15:8	R-0, HC	U-0	U-0	U-0	U-0	U-0	U-0	R-0, HC
	FTZEISE	—	—	—	—	—	—	CARDISE
7:0	R/W-1, HC	R/W-1, HC	R/W-1, HC	R/W-1, HC	R/W-1, HC	R/W-1, HC	R/W-1, HC	R/W-1, HC
	CARDRISE	CARDIISE	BRRDYISE	BWRDYISE	DMAISE	BGISE	TXEISE	CEISE

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-26 **未実装:** 「0」として読み出し

bit 25 **ADEISE:** ADMA エラー割り込み信号イネーブルビット

- 1 = ADMA エラー信号を有効にする
- 0 = ADMA エラー信号をマスクする

bit 24 **ACEISE:** 自動 CMD12 エラー割り込み信号イネーブルビット

- 1 = 自動 CMD12 エラー信号を有効にする
- 0 = 自動 CMD12 エラー信号をマスクする

bit 23 **CLEISE:** 電流制限エラー割り込み信号イネーブルビット

- 1 = 電流制限エラー信号を有効にする
- 0 = 電流制限エラー信号をマスクする

bit 22 **DEBEISE:** データ エンドビット エラー割り込み信号イネーブルビット

- 1 = データ エンドビット エラー信号を有効にする
- 0 = データ エンドビット エラー信号をマスクする

bit 21 **DCRCEISE:** データ CRC エラー割り込み信号イネーブルビット

- 1 = データ CRC エラー信号を有効にする
- 0 = データ CRC エラー信号をマスクする

bit 20 **DTOEISE:** データ タイムアウト エラー割り込み信号イネーブルビット

- 1 = データ タイムアウト エラー信号を有効にする
- 0 = データ タイムアウト エラー信号をマスクする

bit 19 **CIDXEISE:** コマンド インデックス エラー割り込み信号イネーブルビット

- 1 = コマンド インデックス エラー信号を有効にする
- 0 = コマンド インデックス エラー信号をマスクする

bit 18 **CEBEISE:** コマンド エンドビット エラー割り込み信号イネーブルビット

- 1 = コマンド エンドビット エラー信号を有効にする
- 0 = コマンド エンドビット エラー信号をマスクする

bit 17 **CCRCEISE:** コマンド CRC エラー割り込み信号イネーブルビット

- 1 = コマンド CRC エラー信号を有効にする
- 0 = コマンド CRC エラー信号をマスクする

bit 16 **CTOEISE:** コマンド タイムアウト エラー割り込み信号イネーブルビット

- 1 = コマンド タイムアウト エラー信号を有効にする
- 0 = コマンド タイムアウト エラー信号をマスクする

bit 15 **FTZEISE:** ゼロ固定エラー割り込み信号イネーブルビット

- このビットは、このレジスタ内の bit 0 ~ 9 のいずれか (または全て) のビットがセットされた場合にセットされます。
- 1 = エラーを検出した
- 0 = エラーは検出していない

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 39-11: SDHCINTSEN: SDHC 割り込み信号イネーブル レジスタ (続き)

bit 14-9 **未実装**: 「0」として読み出し

bit 8 **CARDISE**: カード割り込み信号イネーブルビット

1 = カード割り込み信号を有効にする  
0 = カード割り込み信号をマスクする

bit 7 **CARDRISE**: カード取り外し割り込み信号イネーブルビット

1 = カード取り外し信号を有効にする  
0 = カード取り外し信号をマスクする

bit 6 **CARDIISE**: カード挿入割り込み信号イネーブルビット

1 = カード挿入信号を有効にする  
0 = カード挿入信号をマスクする

bit 5 **BRRDYISE**: バッファ読み出し準備完了割り込み信号イネーブルビット

1 = バッファ読み出し準備完了信号を有効にする  
0 = バッファ読み出し準備完了信号をマスクする

bit 4 **BWRDYISE**: バッファ書き込み準備完了割り込み信号イネーブルビット

1 = バッファ書き込み準備完了信号を有効にする  
0 = バッファ書き込み準備完了信号をマスクする

bit 3 **DMAISE**: DMA 割り込み信号イネーブルビット

1 = DMA 割り込み信号を有効にする  
0 = DMA 割り込み信号をマスクする

bit 2 **BGISE**: ブロックギャップ割り込み信号イネーブルビット

1 = ブロックギャップ イベント信号を有効にする  
0 = ブロックギャップ イベント信号をマスクする

bit 1 **TXEISE**: 転送完了割り込み信号イネーブルビット

1 = 転送完了信号を有効にする  
0 = 転送完了信号をマスクする

bit 0 **CEISE**: コマンド完了割り込み信号イネーブルビット

1 = コマンド完了信号を有効にする  
0 = コマンド完了信号をマスクする

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-12: SDHCSTAT2: SDHC ステータス レジスタ 2

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R-0, HC	U-0	U-0	R-0, HC	R-0, HC	R-0, HC	R-0, HC	R-0, HC
	CNISSE	—	—	ACIDXE	ACEBE	ACCRCE	ACTOE	ACNEXEC

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット
-n = POR 時の値	U = 未実装、「0」として読み出し
	1 = ビットはセット
	0 = ビットはクリア
	x = ビットは未知

- bit 31-8 **未実装:** 「0」として読み出し
- bit 7 **CNISSE:** 自動 CMD12 によるコマンド未発行エラービット
  - 1 = コマンドは発行されなかった
  - 0 = エラーなし
- bit 6-5 **未実装:** 「0」として読み出し
- bit 4 **ACIDXE:** 自動 CMD12 インデックス エラービット
  - 1 = インデックス エラーが発生した
  - 0 = インデックス エラーは発生していない
- bit 3 **ACEBE:** 自動 CMD12 エンドビット エラービット
  - 1 = エンドビット エラーが発生した
  - 0 = エンドビット エラーは発生していない
- bit 2 **ACCRCE:** 自動 CMD12 CRC エラービット
  - 1 = CRC エラーが発生した
  - 0 = CRC エラーは発生していない
- bit 1 **ACTOE:** 自動 CMD12 タイムアウト エラービット
  - 1 = タイムアウト エラーが発生した
  - 0 = タイムアウト エラーは発生していない
- bit 0 **ACNEXEC:** 自動 CMD12 未実行ビット
  - 1 = 自動 CMD12 が実行されなかった
  - 0 = 自動 CMD12 は実行された

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 39-13: SDHCCAP: SDHC 機能レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-1, HS
	—	—	—	—	—	—	—	VOLT3V3
23:16	R-x, HS	U-0	R-x, HS	U-0	R-x, HS	U-0	R-0, HS	R-0, HS
	SRESUME	—	HISPEED	—	ADMA2	—	MBLEN<1:0>	
15:8	U-0	U-0	R-x, HS	R-x, HS	R-x, HS	R-x, HS	R-x, HS	R-x, HS
	—	—	BASECLK<5:0>					
7:0	R-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
	TOCLKU	—	TOCLKFREQ<5:0>					

**凡例:** HS = ハードウェア セット可能ビット  
R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
-n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-25 **未実装:** 「0」として読み出し
- bit 24 **VOLT3V3:** 3.3 V 電圧サポートビット  
1 = 3.3 V をサポートする
- bit 23 **SRESUME:** 一時停止 / 再開サポートビット  
1 = 一時停止 / 再開をサポートする  
0 = 一時停止 / 再開はサポートしない
- bit 22 **未実装:** 「0」として読み出し
- bit 21 **HISPEED:** ハイスピード サポートビット  
1 = ハイスピードをサポートする  
0 = ハイスピードはサポートしない
- bit 20 **未実装:** 「0」として読み出し
- bit 19 **ADMA2:** ADMA2 サポートビット  
1 = ADMA2 をサポートする  
0 = ADMA2 はサポートしない
- bit 18 **未実装:** 「0」として読み出し
- bit 17-16 **MBLEN<1:0>:** 最大ブロック長ビット  
11 = 予約済み  
10 = 2048  
01 = 1024  
00 = 512
- bit 15-14 **未実装:** 「0」として読み出し
- bit 13-8 **BASECLK<5:0>:** SDCLK ベースクロック周波数ビット  
111111 = 63 MHz  
111110 = 62 MHz  
111101 = 61 MHz  
.  
.  
.  
000010 = 2 MHz  
000001 = 1 MHz  
000000 = 予約済み
- bit 7 **TOCLKU:** タイムアウト クロック単位ビット  
1 = タイムアウト クロックの単位は kHz  
0 = タイムアウト クロックの単位は MHz
- bit 6 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 39-13: SDHCCAP: SDHC 機能レジスタ (続き)

bit 5-0 **TOCLKFREQ<5:0>**: タイムアウト クロック周波数ビット

この値の単位 (kHz または MHz) は、TOCLKU ビットによって定義されます。

111111 = 63 kHz または 63 MHz

111110 = 62 kHz または 62 MHz

111101 = 61 kHz または 61 MHz

⋮

⋮

000010 = 2 kHz または 2 MHz

000001 = 1 kHz または 1 MHz

000000 = 予約済み

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 39-14: SDHCMAXCAP: SDHC 許容最大電流レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R-x, HS	R-x, HS	R-x, HS	R-x, HS	R-x, HS	R-x, HS	R-x, HS	R-x, HS
	MC3V3<7:0>							

<p><b>凡例:</b></p> <p>R = 読み出し可能ビット</p> <p>-n = POR 時の値</p>	<p>HS = ハードウェアでセット</p> <p>W = 書き込み可能ビット</p> <p>1 = ビットはセット</p>	<p>U = 未実装、「0」として読み出し</p> <p>0 = ビットはクリア</p>	<p>x = ビットは未知</p>
--	--	--	-------------------

bit 31-8 **未実装**: 「0」として読み出し

bit 7-0 **MC3V3<7:0>**: 3.3 V 向け最大電流ビット

11111111 = 1020 mA

11111110 = 1016 mA

11111101 = 1012 mA

⋮

00000011 = 12 mA

00000010 = 8 mA

00000001 = 4 mA

00000000 = 予約済み



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 39-15: SDHCFE: SDHC イベント強制レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	W-0, HC	W-0, HC
	—	—	—	—	—	—	FEADE	FEACE
23:16	W-0, HC	W-0, HC	W-0, HC	W-0, HC	W-0, HC	W-0, HC	W-0, HC	W-0, HC
	FECLE	FEDEBE	FEDCRCE	FEDTOE	FEIDX	FECEBE	FECRCRCE	FECTOE
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	W-0	U-0	U-0	W-0	W-0	W-0	W-0	W-0
	FECNIACE	—	—	FEACIDX	FEACEBE	FEACRCRCE	FEACTOE	FEACNEE

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

- bit 31-26 **未実装:** 「0」として読み出し
- bit 25 **FEADE:** ADMA エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 24 **FEACE:** 自動 CMD12 エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 23 **FECLE:** 電流制限エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 22 **FEDEBE:** データ エンドビット エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 21 **FEDCRCE:** データ CRC エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 20 **FEDTOE:** データ タイムアウト エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 19 **FEIDX:** コマンド インデックス エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 18 **FECEBE:** コマンド エンドビット エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 17 **FECRCRCE:** CRC エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 16 **FECTOE:** コマンド タイムアウト エラーに対するイベント強制ビット  
 1 = 割り込みを強制的に発生させる  
 0 = 割り込みを強制的に発生させる事はしない
- bit 15-8 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 39-15: SDHCFE: SDHC イベント強制レジスタ (続き)

- bit 7     **FECNIACE:** 自動 CMD12 エラーによるコマンド未発行エラーに対するイベント強制ビット  
1 = 割り込みを強制的に発生させる  
0 = 割り込みを強制的に発生させる事はしない
- bit 6-5   **未実装:** 「0」として読み出し
- bit 4     **FEACIDX:** 自動 CMD12 インデックス エラーに対するイベント強制ビット  
1 = 割り込みを強制的に発生させる  
0 = 割り込みを強制的に発生させる事はしない
- bit 3     **FEACEBE:** 自動 CMD12 エンドビット エラーに対するイベント強制ビット  
1 = 割り込みを強制的に発生させる  
0 = 割り込みを強制的に発生させる事はしない
- bit 2     **FEACRC:** 自動 CMD12 CRC エラーに対するイベント強制ビット
- bit 1     **FEACTOE:** 自動 CMD12 タイムアウト エラーに対するイベント強制ビット  
1 = 割り込みを強制的に発生させる  
0 = 割り込みを強制的に発生させる事はしない
- bit 0     **FEACNEE:** 自動 CMD12 未実行エラーに対するイベント強制ビット  
1 = 割り込みを強制的に発生させる  
0 = 割り込みを強制的に発生させる事はしない

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 39-16: SDHCADESTAT: SDHC ADMA エラーステータス レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	R-0, HC	R-0, HC	R-0, HC
	—	—	—	—	—	ADLMERR	ADERRST<1:0>	

<b>凡例:</b>	HC = ハードウェアでクリア
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

- bit 31-3 **未実装:** 「0」として読み出し
- bit 2 **ADLMERR:** ADMA 長さ不適合エラービット  
 1 = 長さ不適合エラーが発生した  
 0 = 長さ不適合エラーは発生していない
- bit 1-0 **ADERRST<1:0>:** ADMA エラーステータス ビット  
 11 = データ転送エラー  
 10 = 予約済み  
 01 = ディスクリプタ フェッチエラー  
 00 = DMA 停止エラー

レジスタ 39-17: SDHCAADDR: SDHC ADMA アドレスレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADDR<31:24>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADDR<23:16>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADDR<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ADDR<7:0>							

<b>凡例:</b>	
R = 読み出し可能ビット	W = 書き込み可能ビット U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット 0 = ビットはクリア x = ビットは未知

- bit 31-0 **ADDR<31:0>:** ADMA アドレスレジスタ ビット  
 これらのビットは、実行中コマンドの ADMA ディスクリプタ アドレスを格納します。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

## 40.0 省電力機能

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 10. Power-Saving Features』(DS60001130) を参照してください。この文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

以下では、PIC32MZ DA の省電力機能について説明します。これらのデバイスは複数の電源ドメインを備え、消費電力と性能のバランスを調整するための各種省電力機能とモードを提供します。

### 40.1 CPU 実行中の省電力機能

CPU 実行中の消費電力は、CPU クロック周波数または PBCLK7 速度を低減するか、より低消費電力のクロック源 (LPRC または Sosc) を選択する事により制御できます。

加えて、各周辺モジュールバスに対して周辺モジュールバス スケーリング モードが使えます。このモードを使うと、対応する PBCLKx の分周比を大きくする事で周辺モジュールのクロック速度を低減するか、クロックを完全に停止させる事ができます。

### 40.2 CPU 停止による省電力機能

周辺モジュールと CPU を停止または無効化する事で、消費電力をさらに低減できます。

#### 40.2.1 スリープモード

スリープモードは、消費電力が最も低くなる省電力モードです。CPU と大部分の周辺モジュールは停止し、それらに対応するクロックは停止します。一部の周辺モジュールは動作を継続でき、それらを使ってデバイスをスリープから復帰させる事ができます。スリープ中の周辺モジュールの挙動については、本書内の各周辺モジュールの説明を参照してください。

スリープモードの特性は以下の通りです。

- オシレータの選択に応じて復帰時の遅延時間は異なる
- フェイルセーフ クロック監視 (FSCM) 機能はスリープ中に動作しない
- BOR 回路はスリープ中も動作する
- WDT が有効な場合、スリープへの移行前に WDT は自動的にクリアされない

- 一部の周辺モジュールは、機能は制限されるもののスリープ中に動作できます。これらの周辺モジュールには、入力信号の状態変化を検出する I/O ピン、WDT/ADC/UART モジュール、外部クロック入力または内部 LPRC オシレータを使う周辺モジュール (RTCC、Timer1、入力キャプチャ等) が含まれます。
- I/O ピンは、非スリープ時と同様に、電流のシンクまたはソースを継続します。

プロセッサは、以下のいずれかのイベントが発生した時にスリープから復帰します。

- スリープ中に動作する割り込み要因からの割り込み (割り込みの優先度は、その時の CPU 優先度よりも高い事が必要)
- 全てのデバイスリセット (種類を問わず)
- WDT タイムアウト

割り込み優先度が CPU 優先度以下である場合、CPU は停止したままですが、周辺モジュールバス クロックが動作を開始し、デバイスはアイドルモードに移行します。

#### 40.2.2 アイドルモード

アイドルモードでは CPU は停止しますが、全てのクロックは有効なままです。このため周辺モジュールは動作を継続できます。個々の周辺モジュールの SIDL ビットをセットする事により、アイドルへの移行時に一部のモジュールを選択的に停止させる事ができます。アイドル中に CPU オシレータは動作するため、アイドルからは非常に小さなレイテンシで復帰できます。

SLPEN ビット (OSCCON<4>) をクリアした状態で WAIT 命令を実行すると、デバイスはアイドルに移行します。

プロセッサは、以下のイベントでアイドルから復帰します。

- 有効にされた割り込み要因による割り込みイベント (割り込みイベントの優先度はその時点の CPU 優先度よりも高い事が必要) 割り込みイベントの優先度が CPU 優先度以下である場合、デバイスはアイドルから復帰しません (CPU は停止したまま)。
- 全てのデバイスリセット (種類を問わず)
- WDT タイムアウト割り込み

# PIC32MZ グラフィック (DA) ファミリ

## 40.2.3 ディープスリープモード

ディープスリープモードは、外部スイッチでデバイスの電源を遮断する事なくデバイスの消費電力を最小にします。

### • ディープスリープ

このモードでは、CPU および RAM とほとんどの周辺モジュールへの電力供給が停止します。DSGPR0 レジスタと、RTCC/DSWDT/DSGPR1 ~ DSGPR32 レジスタの1つまたは複数への電力供給は維持されます。

どの周辺モジュールがアクティブになるかは、ディープスリープに移行する際の以下のレジスタビットの状態によって決まります。

### • RTCDIS (DSCON<12>)

ディープスリープ中に RTCC を無効にするには、このビットをセットする必要があります (レジスタ 40-1 参照)。

### • DSWDTEN (DEVCFG2<27>)

ディープスリープ中に DSWDT レジスタを有効にするには、このコンフィグレーションビットをセットする必要があります (レジスタ 41-5 参照)。

### • DSGPREN (DSCON<13>)

ディープスリープ中に DSGPR1 ~ DSGPR32 レジスタを有効にするには、このビットをセットする必要があります (レジスタ 40-1 参照)。

**Note:** ディープスリープ制御レジスタにアクセスする前に、システムのロック解除シーケンスを実行する必要があります。また、ディープスリープ制御レジスタは2回書き込む必要があります。

上記の条件付きで有効になる周辺モジュールに加えて、MCLR フィルタと INTO ピンがディープスリープ中に有効です。

## 40.2.4 VBAT モード

VBAT モードはディープスリープモードに似ていますが、VBAT ピンからデバイスに電力を供給するという点で異なります。VBAT モードは完全にハードウェアにより制御され、ソフトウェアは一切介入しません。VDDCORE パワーオンリセットが発生するとデバイスは VBAT モードに移行します (VPORCORE と VBATSW の定義は表 44-4 参照)。VDDIO/VDDCORE への電力供給が停止して VBAT モードへの移行が発生する前に、外部電源を VBAT ピンに接続しておく必要があります。VBAT モードは、RTCC の動作を維持できる最低消費電力のバッテリー駆動モードです。VDDIO/VDDCORE への電力供給が再開した場合にのみ VBAT モードから復帰できます。この復帰は、デバイスの他の部分からは POR のように見えます。

VBAT モードでは、ディープスリープウォッチドッグタイマは無効になります。RTCC レジスタと DSGPR1 ~ DSGPR32 レジスタは、それぞれ RTCDIS ビット (DSCON<12>) と DSGPREN ビット (DSCON<13>) の設定により有効または無効にできます。ディープスリープ時データ保持汎用レジスタ 0 (DSGPR0) は、VBAT モード中も常に有効です。

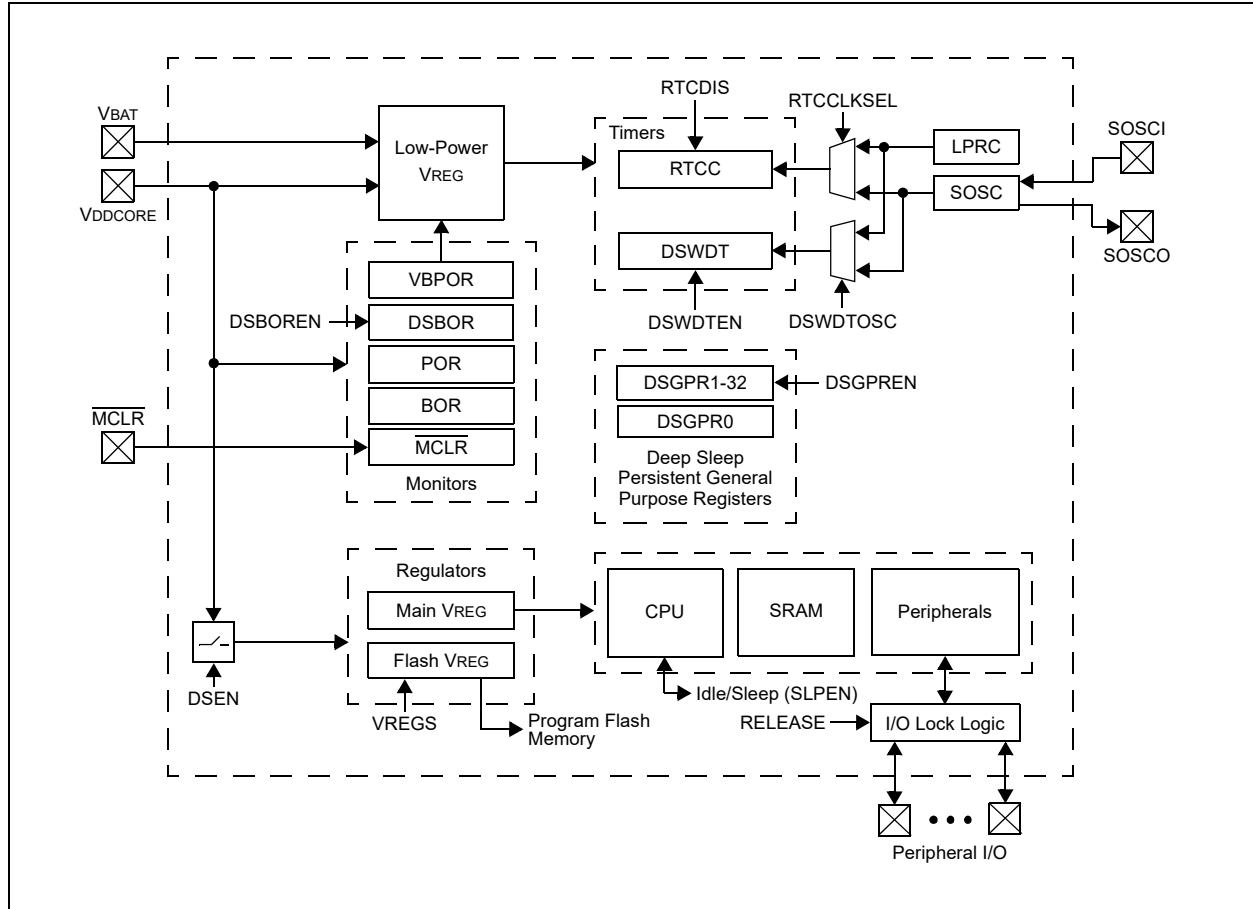
## 40.2.5 XLP 省電力モード

図 40-1 に、XLP デバイスのシステム領域と、関連する省電力機能のブロック図を示します。各種のブロックは、以下のコンフィグレーションビットの設定と SFR により制御されます。

- DSBORN (DEVCFG2<20>)
- DSEN (DSCON<15>)
- DSGPREN (DSCON<13>)
- DSWDTEN (DEVCFG2<27>)
- DSWDTOSC (DEVCFG2<26>)
- RELEASE (DSCON<0>)
- RTCCLKSEL (RTCCON <9:8>)
- RTCDIS (DSCON<12>)
- SLPEN (OSCCON<4>)
- VREGS (PWRCON<0>)

# PIC32MZ グラフィック (DA) ファミリ

図 40-1: XLP デバイスのブロック図



### 40.3 ディープスリープ (DSCTRL) 制御レジスタ

表 40-1: 省電力モードレジスタの一覧

仮アドレ (BF8C_#)	レジスタ名 <sup>(2)</sup>	ビットレンジ	Bit															全リセット <sup>(1)</sup>	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0
0200	DSCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	DSEN	—	DSGPREN	RTCDIS	—	—	—	—	RTCCWDIS	—	—	—	—	—	—	DSBOR	RELEASE
0210	DSWAKE	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0220	DSGPR0 <sup>(1)</sup>	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
0240	DSGPR1	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
0244	DSGPR2	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
0248	DSGPR3	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
024C	DSGPR4	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
0250	DSGPR5	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
0254	DSGPR6	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
0258	DSGPR7	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
025C	DSGPR8	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
0260	DSGPR9	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
0264	DSGPR10	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	
0268	DSGPR11	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000	
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000	

凡例: — = 未実装、「0」として読み出し

- Note
- 1: DSGPR0 レジスタは、全てのデバイス動作モードで保持されます。
  - 2: ディープスリープ制御レジスタにアクセスする前に、システムのロック解除シーケンスを実行する必要があります。これらのレジスタは、2 回書き込む必要があります。



表 40-1: 省電力モードレジスタの一覧

仮アドレ ス (BF8C_#)	レジスタ名 <sup>(2)</sup>	ビットレンジ	Bit														全リセット <sup>(1)</sup>
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	
026C	DSGPR12	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
0270	DSGPR13	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
0274	DSGPR14	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
0278	DSGPR15	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
027C	DSGPR16	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
0280	DSGPR17	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
0284	DSGPR18	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
0288	DSGPR19	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
028C	DSGPR20	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
0290	DSGPR21	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
0294	DSGPR22	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
0298	DSGPR23	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
029C	DSGPR24	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
02A0	DSGPR25	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000
02A4	DSGPR26	31:16	ディープスリープ時データ保持汎用ビット <31:16>														0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>														0000

凡例: — = 未実装、「0」として読み出し

- Note
- 1: DSGPR0 レジスタは、全てのデバイス動作モードで保持されます。
  - 2: ディープスリープ制御レジスタにアクセスする前に、システムのロック解除シーケンスを実行する必要があります。これらのレジスタは、2回書き込む必要があります。

表 40-1: 省電力モードレジスタの一覧

アドレス (BF8C_#)	レジスタ名 <sup>(2)</sup>	ビットレンジ	Bit															全リセット <sup>(1)</sup>
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	
02A8	DSGPR27	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000
02AC	DSGPR28	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000
02B0	DSGPR29	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000
02B4	DSGPR30	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000
02B8	DSGPR31	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000
02BC	DSGPR32	31:16	ディープスリープ時データ保持汎用ビット <31:16>															0000
		15:0	ディープスリープ時データ保持汎用ビット <15:0>															0000

凡例: — = 未実装、「0」として読み出し

- Note
- 1: DSGPR0 レジスタは、全てのデバイス動作モードで保持されます。
  - 2: ディープスリープ制御レジスタにアクセスする前に、システムのロック解除シーケンスを実行する必要があります。これらのレジスタは、2回書き込む必要があります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 40-1: DSCON: ディープスリープ制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
23:16	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —	U-0 —
15:8	HC, R/W-y	U-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0
	DSEN <sup>(1)</sup>	—	DSGPREN	RTCDIS	—	—	—	RTCCWDIS
7:0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	DSBOR <sup>(2)</sup>	RELEASE

**凡例:** HC = ハードウェアでクリア      y = POR 時にコンフィグレーション ビットからの値に設定  
R = 読み出し可能ビット      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
-n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-16 **未実装:** 「0」として読み出し

bit 15 **DSEN:** ディープスリープ イネーブルビット<sup>(1)</sup>

1 = WAIT 命令でディープスリープに移行する  
0 = WAIT 命令でスリープに移行する

bit 14 **未実装:** 「0」として読み出し

bit 13 **DSGPREN:** 汎用レジスタ イネーブルビット

1 = ディープスリープ中に汎用レジスタのデータを保持する  
0 = ディープスリープ中に汎用レジスタのデータを保持しない

bit bit 12 **RTCDIS:** RTCC モジュール ディセーブルビット

1 = RTCC モジュールを無効にする  
0 = RTCC モジュールを有効にする

bit 11-9 **未実装:** 「0」として読み出し

bit 8 **RTCCWDIS:** RTCC 復帰ディセーブル ビット

1 = RTCC からの復帰を無効にする  
0 = RTCC からの復帰を有効にする

bit 7-2 **未実装:** 「0」として読み出し

bit 1 **DSBOR:** ディープスリープ BOR イベント ステータスビット<sup>(2)</sup>

1 = DSBOREN が有効な状態でディープスリープ中に VDDCORE が DSBOR しきい値よりも低下した<sup>(2)</sup>  
0 = DSBOREN が無効であった、または有効であったがディープスリープ中に VDDCORE は DSBOR しきい値よりも低下しなかった

bit 0 **RELEASE:** I/O ピン状態リリースビット

1 = ディープスリープからの復帰時に I/O ピンは以前の状態を保持する  
0 = I/O ピンをリリースし、対応する TRIS および LAT ビットで I/O ピンの状態を制御する

**Note 1:** ディープスリープに移行するには、DSEN ビットをセットしてからスリープへの移行を実行する必要があります。

**2:** ディープスリープ ブラウンアウト リセット (BOR) イベントは、他の全てのイベントとは異なり、ディープスリープからの復帰をトリガしません。このビットはステータスビットとしてのみ機能します。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 40-2: DSWAKE: ディープスリープ復帰要因レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HS
	—	—	—	—	—	—	—	DSINT0
7:0	R/W-0, HS	U-0	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	U-0	U-0
	DSFLT	—	—	DSWDT	DSRTC	DSMCLR	—	—

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値              1 = ビットはセット              0 = ビットはクリア              x = ビットは未知  
 HS = ハードウェアでセット

bit 31-9 **未実装:** 「0」として読み出し

bit 8 **DSINT0:** 入力状態変化時割り込みビット

1 = ディープスリープ中に入力状態変化割り込みがアサートされた  
 0 = ディープスリープ中に入力状態変化割り込みはアサートされなかった

bit 7 **DSFLT:** ディープスリープ中異常検出ビット

1 = ディープスリープ中に異常が発生し、一部のディープスリープ コンフィグレーション設定が破損した可能性がある  
 0 = ディープスリープ中に異常は発生しなかった

bit 6-5 **未実装:** 「0」として読み出し

bit 4 **DSWDT:** ディープスリープ ウォッチドッグ タイマ タイムアウトビット

1 = ディープスリープ中にディープスリープ ウォッチドッグ タイマがタイムアウトした  
 0 = ディープスリープ中にディープスリープ ウォッチドッグ タイマはタイムアウトしなかった

bit 3 **DSRTC:** リアルタイム クロック / カレンダー アラームビット

1 = ディープスリープ中にリアルタイム クロック / カレンダーがアラームをトリガした  
 0 = ディープスリープ中にリアルタイム クロック / カレンダーはアラームをトリガしなかった

bit 2 **DSMCLR:** MCLR イベントビット

1 = アクティブであった MCLR ピンがディープスリープ中にアサートされた  
 0 = MCLR ピンはアクティブではなかった、またはアクティブであったがディープスリープ中にアサートされなかった

bit 1-0 **未実装:** 「0」として読み出し

**Note:** DSEN ビット (DSCON<15>) をセットすると、このレジスタ内の全てのビットがクリアされます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 40-3: DSGPRx: ディープスリープ時データ保持汎用レジスタ「x」(x = 0 ~ 32)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	ディープスリープ時データ保持汎用ビット							
23:16	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	ディープスリープ時データ保持汎用ビット							
15:8	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	ディープスリープ時データ保持汎用ビット							
7:0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	ディープスリープ時データ保持汎用ビット							

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 ディープスリープ時データ保持汎用ビット

**Note:** DSGPR0 レジスタの内容は、ディープスリープモードと VBAT モード中に保持されます。ディープスリープおよび VBAT モード時の DSPGR1 ~ DSPGR32 レジスタのデータ保持機能は既定値では無効ですが、DSGPREN ビット (DSCON<13>) を使って有効にできます。全てのレジスタビットは、ディープスリープモードとは無関係の VDDCORE パワーオンリセット (POR) イベントの場合にのみリセットされます。

# PIC32MZ グラフィック (DA) ファミリ

---

## 40.4 周辺モジュールの無効化

周辺モジュール ディセーブル (PMD) レジスタを使うと、周辺モジュールに供給されるクロック源を全て停止する事によってモジュールを無効にできます。PMD 制御ビットを使って無効にされたモジュールは、最小消費電力状態になります。その周辺モジュールに対応する制御およびステータス レジスタも無効になります。それらのレジスタに書き込んでも効果はなく、それらのレジスタからの読み値は無効です。

周辺モジュールを無効にするには、対応する PMD<sub>x</sub> ビットを「1」にセットする必要があります。周辺モジュールを有効にするには、対応する PMD<sub>x</sub> ビットを「0」(既定値)にクリアする必要があります。詳細は [表 40-2](#) を参照してください。

**Note:** ON ビットがセットされたままの周辺モジュールをPMD<sub>x</sub>ビットで無効にした場合の動作は未定義です。PMD<sub>x</sub> ビットを使って周辺モジュールを無効にする前に、そのモジュールの ON ビットをクリアしておく必要があります。

TABLE 40-2: 周辺モジュール ディセーブル レジスタの一覧

仮想アドレス (BF80_#)	レジスタ名	ビットレンジ	Bit															全リセット (1)				
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1		16/0			
0040	PMD1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	HLVDMD	—	—	—	—	0000		
		15:0	—	—	—	CVRMD	—	—	—	—	CTMUMD	—	—	—	—	—	—	—	—	—	ADCMD	0000
0050	PMD2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMP2MD	CMP1MD	0000
0060	PMD3	31:16	—	—	—	—	—	—	—	—	OC9MD	OC8MD	OC7MD	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	—	—	0000
		15:0	—	—	—	—	—	—	—	—	IC9MD	IC8MD	IC7MD	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	—	—	0000
0070	PMD4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
		15:0	—	—	—	—	—	—	—	—	T9MD	T8MD	T7MD	T6MD	T5MD	T4MD	T3MD	T2MD	T1MD	—	—	0000
0080	PMD5	31:16	—	—	CAN2MD	CAN1MD	—	—	—	—	USBMD	—	—	—	I2C5MD	I2C4MD	I2C3MD	I2C2MD	I2C1MD	—	—	0000
		15:0	—	—	SPI6MD	SPI5MD	SPI4MD	SPI3MD	SPI2MD	SPI1MD	—	—	—	U6MD	U5MD	U4MD	U3MD	U2MD	U1MD	—	—	0000
0090	PMD6	31:16	—	—	—	ETHMD	—	—	—	—	SQI1MD	—	—	SDHCMD	GLCDMD	—	GPUMD	EBIMD	PMPMD	—	—	0000
		15:0	—	—	—	REFO5MD	REFO4MD	REFO3MD	REFO2MD	REFO1MD	—	—	—	—	—	—	—	—	—	—	—	0000
00A0	PMD7	31:16	—	—	—	DDR2CMD	—	—	—	—	—	—	CRYPTMD	—	RNGMD	—	—	—	—	—	—	1000
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	DMAMD	—	—	—	—	—	—	0000

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

Note 1: リセット値はデバイスのタイプによって異なります。

# PIC32MZ グラフィック (DA) ファミリ

表 40-3: 周辺モジュール ディセーブル ビットの名称と位置

周辺モジュール	PMDx ビット名	レジスタ名とビット位値
ADC	ADCMD	PMD1<0>
CTMU	CTMUMD	PMD1<8>
コンパレータ参照電圧	CVRMD	PMD1<12>
HIGH/LOW 電圧検出	HLVDM	PMD1<20>
コンパレータ 1	CMP1MD	PMD2<0>
コンパレータ 2	CMP2MD	PMD2<1>
入力キャプチャ 1	IC1MD	PMD3<0>
入力キャプチャ 2	IC2MD	PMD3<1>
入力キャプチャ 3	IC3MD	PMD3<2>
入力キャプチャ 4	IC4MD	PMD3<3>
入力キャプチャ 5	IC5MD	PMD3<4>
入力キャプチャ 6	IC6MD	PMD3<5>
入力キャプチャ 7	IC7MD	PMD3<6>
入力キャプチャ 8	IC8MD	PMD3<7>
入力キャプチャ 9	IC9MD	PMD3<8>
出力コンペア 1	OC1MD	PMD3<16>
出力コンペア 2	OC2MD	PMD3<17>
出力コンペア 3	OC3MD	PMD3<18>
出力コンペア 4	OC4MD	PMD3<19>
出力コンペア 5	OC5MD	PMD3<20>
出力コンペア 6	OC6MD	PMD3<21>
出力コンペア 7	OC7MD	PMD3<22>
出力コンペア 8	OC8MD	PMD3<23>
出力コンペア 9	OC9MD	PMD3<24>
Timer1	T1MD	PMD4<0>
Timer2	T2MD	PMD4<1>
Timer3	T3MD	PMD4<2>
Timer4	T4MD	PMD4<3>
Timer5	T5MD	PMD4<4>
Timer6	T6MD	PMD4<5>
Timer7	T7MD	PMD4<6>
Timer8	T8MD	PMD4<7>
Timer9	T9MD	PMD4<8>
UART1	U1MD	PMD5<0>
UART2	U2MD	PMD5<1>
UART3	U3MD	PMD5<2>
UART4	U4MD	PMD5<3>
UART5	U5MD	PMD5<4>
UART6	U6MD	PMD5<5>
SPI1	SPI1MD	PMD5<8>
SPI2	SPI2MD	PMD5<9>

**Note 1:** USBMD ビットをセットする前に、USB モジュールは対応する ON ビットのクリア後に非ビジーである必要があります。

**2:** 一部のデバイスはこの周辺モジュールを実装していません。デバイスがこの周辺モジュールを実装しているかどうかは、ピン機能表 (表 2 ~ 表 4) を参照してください。



# PIC32MZ グラフィック (DA) ファミリ

表 40-3: 周辺モジュール ディセーブルビットの名称と位置 ( 続き )

周辺モジュール	PMDx ビット名	レジスタ名とビット位値
SPI3	SPI3MD	PMD5<10>
SPI4	SPI4MD	PMD5<11>
SPI5	SPI5MD	PMD5<12>
SPI6	SPI6MD	PMD5<13>
I2C1	I2C1MD	PMD5<16>
I2C2	I2C2MD	PMD5<17>
I2C3	I2C3MD	PMD5<18>
I2C4	I2C4MD	PMD5<19>
I2C5	I2C5MD	PMD5<20>
USB <sup>(1)</sup>	USBMD	PMD5<24>
CAN1	CAN1MD	PMD5<28>
CAN2	CAN2MD	PMD5<29>
参照クロック出力 1	REFO1MD	PMD6<8>
参照クロック出力 2	REFO2MD	PMD6<9>
参照クロック出力 3	REFO3MD	PMD6<10>
参照クロック出力 4	REFO4MD	PMD6<11>
参照クロック出力 5	REFO5MD	PMD6<12>
PMP	PMPMD	PMD6<16>
EBI	EBIMD	PMD6<17>
2-D GPU	GPUMD	PMD6<18>
GLCD	GLCDMD	PMD6<20>
SDHC	SDHCMD	PMD6<21>
SQI1	SQI1MD	PMD6<23>
Ethernet	ETHMD	PMD6<28>
DMA	DMAMD	PMD7<4>
RNG	RNGMD	PMD7<20>
暗号エンジン <sup>(2)</sup>	CRYPTMD	PMD7<22>
DDR2 SDRAM コントローラ <sup>(2)</sup>	DDR2CMD	PMD7<28>

**Note 1:** USBMD ビットをセットする前に、USB モジュールは対応する ON ビットのクリア後に非ビジーである必要があります。

**2:** 一部のデバイスはこの周辺モジュールを実装していません。デバイスがこの周辺モジュールを実装しているかどうかは、ピン機能表 ( 表 2 ~ 表 4 ) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

---

## 40.4.1 コンフィグレーション変更の制限

周辺モジュールは実行時に無効にできるため、誤った無効化を防ぐための制限措置が必要です。周辺モジュールの有効化 / 無効化を制限するため、PIC32MZ DA は以下の 2 つの機能を備えています。

- 制御レジスタのロックシーケンス
- コンフィグレーション ビット選択のロック

### 40.4.1.1 制御レジスタのロックシーケンス

通常動作中に PMD<sub>x</sub> レジスタに書き込む事はできません。書き込みを試みると、正常に実行されたかのように見えますが、実際にはレジスタの内容は変更されません。これらのレジスタを変更するには、ハードウェアでレジスタのロックを解除する必要があります。レジスタのロックは PMDLOCK コンフィグレーション ビット (CFGCON<12>) で制御します。PMDLOCK ビットをセットすると制御レジスタへの書き込みが禁止され、クリアすると書き込みが許可されます。

PMDLOCK ビットをセットまたはクリアするには、ロック解除シーケンスを実行する必要があります。詳細は『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

### 40.4.1.2 コンフィグレーション ビット選択のロック

さらに安全性を高めるため、PMD<sub>x</sub> レジスタに対して複数回の書き込みを防ぐようにデバイスを設定できます。PMDL1WAY コンフィグレーション ビット (DEVCFG3<28>) は、PMDLOCK ビットが一度セットされた後でクリアされる事を防ぎます。PMDLOCK ビットがセットされたままの場合、レジスタロック解除シーケンスは実行されず、PPS 制御レジスタに書き込む事はできません。このビットをクリアして PMD 機能を再び有効にするには、デバイスリセットを実行する必要があります。

# PIC32MZ グラフィック (DA) ファミリ

## 41.0 その他の特殊機能

**Note:** 本書は PIC32MZ グラフィック (DA) ファミリの機能をまとめて記載したものであり、包括的な参照資料の提供を意図したものではありません。本書の内容を補足する情報は、『PIC32 Family Reference Manual, Section 32. Configuration』(DS60001124) と『PIC32 Family Reference Manual, Section 33. Programming and Diagnostics』(DS60001129) を参照してください。これらの文書は、Microchip 社 PIC32 ウェブページ ([www.microchip.com/pic32](http://www.microchip.com/pic32)) 内の [Documentation > Reference Manual] セクションから入手できます。

PIC32MZ DA は、アプリケーションの柔軟性と信頼性を最大限に高めると共に外付け部品を不要にしてコストを最小限に抑えるための各種機能を備えています。これらの機能には以下が含まれます。

- 柔軟なデバイス コンフィグレーション
- JTAG (Joint Test Action Group) インターフェイス
- インサーキット シリアル プログラミング (ICSP™)
- 内部温度センサ

### 41.1 コンフィグレーション ビット

PIC32MZ DA は 2 つのブートフラッシュメモリ (ブートフラッシュ 1 とブートフラッシュ 2) を内蔵し、各メモリにコンフィグレーション空間が割り当てられています。これらのコンフィグレーション空間には、各種のデバイス コンフィグレーションが書き込めます。下位ブートエイリアス メモリ領域によってエイリアスされるコンフィグレーション空間を使う事で、以下のコンフィグレーションレジスタに値を提供します。詳細は 4.1.1 「ブートフラッシュのシーケンス空間とコンフィグレーション空間」を参照してください。

- DEVSIGN0/ADEVSIGN0: デバイス シグネチャワード 0 レジスタ
- DEVCP0/ADEVCP0: デバイスコード保護 0 レジスタ
- DEVCFG0/ADEVCFG0: デバイス コンフィグレーションワード 0 レジスタ
- DEVCFG1/ADEVCFG1: デバイス コンフィグレーションワード 1 レジスタ
- DEVCFG2/ADEVCFG2: デバイス コンフィグレーションワード 2 レジスタ
- DEVCFG3/ADEVCFG3: デバイス コンフィグレーションワード 3 レジスタ
- DEVCFG4/ADEVCFG4: デバイス コンフィグレーションワード 4 レジスタ
- DEVADCx: デバイス ADC 校正ワード「x」レジスタ (x = 0 ~ 4、7)

実行時に設定可能な以下のコンフィグレーションレジスタは、追加のコンフィグレーション制御を提供します。

- CFGCON: コンフィグレーション制御レジスタ
- CFGEBIA: 外部バス インターフェイス アドレスピンコンフィグレーションレジスタ
- CFGEBIC: 外部バス インターフェイス制御ピンコンフィグレーションレジスタ
- CFGPG: パーミッション グループ コンフィグレーションレジスタ
- CFGCON2: コンフィグレーション制御レジスタ 2 レジスタ
- CFGMPLL: メモリ PLL コンフィグレーションレジスタ

加えて、DEVID レジスタ (レジスタ 41-15 参照) はデバイスおよびビジョン情報を提供し、DEVSN0 および DEVSN1 レジスタはデバイスの一意シリアル番号を格納します (レジスタ 41-16 参照)。

**Note:** 以下に記載するデバイスワードのプログラミングにはワード書き込み動作 (NVMOP<3:0> = 0001) を使わないでください。

## 41.2 レジスタ

表 41-1: DEVCFG: デバイス コンフィグレーション ワードの一覧

仮アドレス (BFCO_#)	レジスタ名	ビットレンジ	Bit																全リセット	
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0		
FFBC	DEVCFG4	31:16	—	—	—	SWDTPS<4:0>						—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFC0	DEVCFG3	31:16	—	—	IOL1WAY	PMDL1WAY	PGL1WAY	—	FETHIO	FMIEN	—	—	—	—	EXTDDRSIZE<3:0>				xxxx	
		15:0	USERID<15:0>																xxxx	
FFC4	DEVCFG2	31:16	—	UPLLFSEL	—	FDSEN	DSWDTEN	DSWDTOSC	DSWDTPS<4:0>				DSBOREN	VBATBOREN	FPLLIDIV<2:0>			xxxx		
		15:0	FPLLMULT<6:0>						FPLLICK	FPLL RNG<2:0>			—	FPLLIDIV<2:0>			xxxx			
FFC8	DEVCFG1	31:16	FDMTEN	DMTCNT<4:0>				FWDTWINSZ<1:0>		FWDTEN	WINDIS	WDTSPGM	WDTPS<4:0>				xxxx			
		15:0	FCKSM<1:0>		—	—	—	OSCI OFNC	POSCMOD<1:0>		IESO	FSOSCEN	DMTINTV<2:0>		FNOSC<2:0>		xxxx			
FFCC	DEVCFG0	31:16	—	EJTAGBEN	—	—	POSCAGC	—	POSCTYPE<1:0>		—	—	POSCBOOST	POSCGAIN<1:0>		SOSCBOOST	SOSCGAIN<1:0>		xxxx	
		15:0	SMCLR	DBGPER<2:0>				—	FSLEEP	FECCCON<1:0>		—	BOOTISA	TRCEN	ICESEL<1:0>		JTAGEN	DEBUG<1:0>		xxxx
FFD0	DEVCP3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFD4	DEVCP2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFD8	DEVCP1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFDC	DEVCP0	31:16	—	—	—	CP	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFE0	DEVSIGN3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFE4	DEVSIGN2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFE8	DEVSIGN1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
FFEC	DEVSIGN0	31:16	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx	
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は 16 進表記

表 41-2: ADEVCFG: 代替デバイスコンフィグレーションワードの一覧

アドレス (BFC0_#)	レジスタ名	ビットレンジ	Bit														全リセット		
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2		17/1	16/0
FF3C	DEVCFG4	31:16	—	—	—	SWDTPS<4:0>				—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF40	ADEVCFG3	31:16	—	—	IOL1WAY	PMDL1WAY	PGL1WAY	—	FETHIO	FMIIEN	—	—	—	—	EXTDDRSIZE<3:0>				xxxx
		15:0	USERID<15:0>														xxxx		
FF44	ADEVCFG2	31:16	—	UPLLFSEL	—	FDSSEN	DSWDTEN	DSWDTOSC	DSWDTPS<4:0>				DSBOREN	VBATBOREN	FPLLIDIV<2:0>			xxxx	
		15:0	FPLLMULT<6:0>						FPLLICKL	FPLL RNG<2:0>		—	FPLLIDIV<2:0>			xxxx			
FF48	ADEVCFG1	31:16	FDMTEN	DMTCNT<4:0>				FWDTWINSZ<1:0>	FWDTEN	WINDIS	WDTSPGM	WDTPS<4:0>				xxxx			
		15:0	FCKSM<1:0>		—	—	—	OSCI0FNC	POSCMOD<1:0>	IESO	FSOCEN	DMTINTV<2:0>		FNOSC<2:0>			xxxx		
FF4C	ADEVCFG0	31:16	—	EJTAGBEN	—	—	—	POSCAGC	—	POSCTYPE<1:0>	—	—	POSCBOOST	POSCGAIN<1:0>	SOSCB00ST	SOSCGAIN<1:0>	—	—	xxxx
		15:0	SMCLR	DBGPER<2:0>			—	FSLEEP	FECCON<1:0>		—	BOOTISA	TRCEN	ICESEL<1:0>		JTAGEN	DEBUG<1:0>		xxxx
FF50	ADEVCP3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF54	ADEVCP2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF58	ADEVCP1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF5C	ADEVCP0	31:16	—	—	—	CP	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF60	ADEVSIGN3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF64	ADEVSIGN2	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF68	ADEVSIGN1	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF6C	ADEVSIGN0	31:16	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	xxxx
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

凡例: x = リセット時に未知の値、— = 未実装 (「0」として読み出し)、リセット値は16進表記



表 41-5: デバイス ADC 校正レジスタの一覧

仮想アドレス (BFC5_#)	レジスタ名	ビットレンジ	Bit														全リセット (1)
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	
4000	DEVADC0	31:16	ADC 校正データ <31:16>														xxxxx
		15:0	ADC 校正データ <15:0>														xxxxx
4004	DEVADC1	31:16	ADC 校正データ <31:16>														xxxxx
		15:0	ADC 校正データ <15:0>														xxxxx
4008	DEVADC2	31:16	ADC 校正データ <31:16>														xxxxx
		15:0	ADC 校正データ <15:0>														xxxxx
400C	DEVADC3	31:16	ADC 校正データ <31:16>														xxxxx
		15:0	ADC 校正データ <15:0>														xxxxx
4010	DEVADC4	31:16	ADC 校正データ <31:16>														xxxxx
		15:0	ADC 校正データ <15:0>														xxxxx
401C	DEVADC7	31:16	ADC 校正データ <31:16>														xxxxx
		15:0	ADC 校正データ <15:0>														xxxxx

凡例: x = リセット時に未知の値  
 Note 1: リセット値はデバイスのタイプによって異なります。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-1: DEVSIGN0/ADEVSIGN0: デバイス シグネチャワード 0 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	r-0	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—
23:16	r-1	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—
15:8	r-1	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—
7:0	r-1	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—

**凡例:** r = 予約済みビット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31 予約済み: 「0」として書き込み

bit 30-0 予約済み: 「1」として書き込み

**Note:** DEVSIGN1 ~ DEVSIGN3 レジスタと ADEVSIGN1 ~ ADEVSIGN3 レジスタは、DEVSIGN0/ADESIGN0 レジスタに対するクワッドワードプログラミング動作に使用します。これらのレジスタは有効な情報を格納しません。

レジスタ 41-2: DEVCP0/ADEVCP0: デバイスコード保護 0 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	r-1	r-1	r-1	R/P	r-1	r-1	r-1	r-1
	—	—	—	CP	—	—	—	—
23:16	r-1	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—
15:8	r-1	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—
7:0	r-1	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—

**凡例:** r = 予約済みビット P = プログラム可能ビット  
R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
-n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-29 予約済み: 「1」として書き込み

bit 28 CP: コード保護ビット

外部プログラミング デバイスによるブートおよびプログラム フラッシュメモリの読み出しまたは変更を防ぎます。

1 = 保護を無効にする

0 = 保護を有効にする

bit 27-0 予約済み: 「1」として書き込み

**Note:** DEVCP1 ~ DEVCP3 レジスタと ADEVCP1 ~ ADEVCP3 レジスタは、DEVCP0/ADEVCP0 レジスタに対するクワッドワードプログラミング動作に使用します。これらのレジスタは有効な情報を格納しません。



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-3: DEVCFG0/ADEVCFG0: デバイス コンフィグレーションワード0 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	r-x —	R/P EJTAGBEN	r-1 —	r-1 —	r-1 POSCACG	r-1 —	r-1 POSCTYPE<1:0>	r-1 —
23:16	r-1 —	r-1 —	R/P POSCBOOST	R/P POSCGAIN<1:0>	R/P —	R/P SOSCBOOST	R/P SOSCGAIN<1:0>	R/P —
15:8	R/P SMCLR	R/P —	R/P DBGPER<2:0>	R/P —	r-y —	R/P FSLEEP	R/P FECCCON<1:0>	R/P —
7:0	r-1 —	R/P BOOTISA	R/P TRCEN	R/P ICESEL<1:0>	R/P —	R/P JTAGEN <sup>(1)</sup>	R/P —	R/P DEBUG<1:0>

<b>凡例:</b>	r = 予約済みビット	y = POR 時にコンフィグレーション ビットからの値に設定
R = 読み出し可能ビット	P = プログラミング可能ビット	U = 未実装ビット、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

bit 31 **予約済み:** このビットのリセット値は DEVSIGN0<31> と同じです。

bit 30 **EJTAGBEN:** EJTAG ブート イネーブルビット

- 1 = 標準の EJTAG 機能
- 0 = 削減した EJTAG 機能

bit 29-28 **予約済み:** 「1」として書き込み

bit 27 **POSCACG:** プライマリ オシレータ 自動ゲイン制御ビット

- 1 = POSC 自動ゲイン制御を有効にする
- 0 = POSC 自動ゲイン制御を無効にする (手動ゲイン制御を有効にする)

bit 26 **予約済み:** 「1」として書き込み

bit 25-24 **POSCTYPE:** プライマリ オシレータ タイプビット

- NOTE: これらのビットは、水晶振動子周波数に応じたゲインループの制御用に使います。
- 11 = 12 MHz 水晶振動子
  - 10 = 24 MHz 水晶振動子
  - 01 = レゾネータ
  - 00 = 8 MHz 水晶振動子

bit 23-22 **予約済み:** 「1」として書き込み

bit 21 **POSCBOOST:** プライマリ オシレータ キックスタート ブーストビット

- 1 = オシレータのキックスタートをブーストする
- 0 = オシレータを通常の方法で起動する

bit 20-19 **POSCGAIN<1:0>:** プライマリ オシレータ ゲイン制御ビット

- 11 = ゲインレベル 3 (最高)
- 10 = ゲインレベル 2
- 01 = ゲインレベル 1
- 00 = ゲインレベル 0 (最低)

bit 18 **SOSCBOOST:** セカンダリ オシレータ キックスタート ブースト イネーブルビット

- 1 = オシレータのキックスタートをブーストする
- 0 = オシレータを通常の方法で起動する

bit 17-16 **SOSCGAIN<1:0>:** セカンダリ オシレータ ゲイン制御ビット

- 11 = ゲインレベル 3 (最高)
- 10 = ゲインレベル 2
- 01 = ゲインレベル 1
- 00 = ゲインレベル 0 (最低)

bit 15 **SMCLR:** ソフト マスタクリア イネーブルビット

- 1 = MCLR ピンは通常のシステムリセットを生成する
- 0 = MCLR ピンは POR を生成する

**Note 1:** このビットは、CFGCON レジスタの JTAGEN ビットの値を設定します。起動時にこのヒューズが「0」に設定された場合、CFGCON レジスタの JTAGEN ビットは実行時に無視されます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 41-3: DEVCFG0/ADEVCFG0: デバイス コンフィグレーションワード 0 レジスタ (続き)

bit 14-12 **DBGPER<2:0>**: デバッグモード CPU アクセス パーミッション ビット

1xx = パーミッション グループ 2 領域への CPU アクセスを許可する  
x1x = パーミッション グループ 1 領域への CPU アクセスを許可する  
xx1 = パーミッション グループ 0 領域への CPU アクセスを許可する  
0xx = パーミッション グループ 2 領域への CPU アクセスを拒否する  
x0x = パーミッション グループ 1 領域への CPU アクセスを拒否する  
xx0 = パーミッション グループ 0 領域への CPU アクセスを拒否する

CPU がデバッグモード中の場合、CPU1PG<1:0> ビット (CFGPG<1:0>) を DBGPER<2:0> で拒否されているパーミッション グループに設定した場合、トランザクション要求はパーミッション グループ 3 に割り当てられます。

bit 11 **予約済み**: このビットはデバッガ / エミュレータ開発ツールによって制御されます。このビットは変更しないでください。

bit 10 **FSLEEP**: フラッシュ スリープモード ビット

1 = デバイスがスリープ中の時にフラッシュの電源を OFF にする  
0 = フラッシュの電源は VREGS ビット (PWRCON<0>) によって制御する

bit 9-8 **FECCCON<1:0>**: 動的フラッシュ ECC コンフィグレーション ビット

デバイスリセット時に、これらのビットの値が ECCCON<1:0> ビット (CFGCON<5:4>) にコピーされます。  
11 = ECC および動的 ECC を無効にする (ECCCON<1:0> ビットは書き込み可能)  
10 = ECC および動的 ECC を無効にする (ECCCON<1:0> ビットをロック)  
01 = 動的フラッシュ ECC を有効にする (ECCCON<1:0> ビットをロック)  
00 = フラッシュ ECC を有効にする (ECCCON<1:0> ビットをロック、ワードフラッシュ書き込みは無効)

bit 7 **予約済み**: 「1」として書き込み

bit 6 **BOOTISA**: ブート ISA 選択ビット

1 = ブートコードと例外コードは MIPS32 を使う  
CP0 CONFIG3 レジスタ内で ISAONEXC ビットが「0」にクリアされ、ISA<1:0> ビットが「10」に設定されます。  
0 = ブートコードと例外コードは microMIPS を使う  
CP0 CONFIG3 レジスタ内で ISAONEXC ビットが「1」にセットされ、ISA<1:0> ビットが「11」に設定されます。

bit 5 **TRCEN**: トレース イネーブルビット

1 = CPU でトレース機能を有効にする  
0 = CPU でトレース機能を無効にする

bit 4-3 **ICESEL<1:0>**: インサーキット エミュレータ / デバッガ通信チャンネル選択ビット

11 = PGEC1/PGED1 ペアを使う  
10 = PGEC2/PGED2 ペアを使う  
01 = 予約済み  
00 = 予約済み

bit 2 **JTAGEN**: JTAG イネーブルビット<sup>(1)</sup>

1 = JTAG を有効にする  
0 = JTAG を無効にする

bit 1-0 **DEBUG<1:0>**: バックグラウンド デバッガ イネーブルビット (コード保護が有効な場合、「11」に固定)

1x = デバッガを有効にする  
0x = デバッガを無効にする

**Note 1:** このビットは、CFGCON レジスタの JTAGEN ビットの値を設定します。起動時にこのヒューズが「0」に設定された場合、CFGCON レジスタの JTAGEN ビットは実行時に無視されます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-4: DEVCFG1/ADEVCFG1: デバイス コンフィグレーションワード1 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	FDMTEN	DMTCNT<4:0>					FWDTWINSZ<1:0>	
23:16	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	FWDTEN	WINDIS	WDTSPGM	WDTPS<4:0>				
15:8	R/P	R/P	r-1	r-1	r-1	R/P	R/P	R/P
	FCKSM<1:0>		—	—	—	OSCIOFNC	POSCMOD<1:0>	
7:0	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	IESO	FSOSCEN	DMTINV<2:0>			FNOSC<2:0>		

<b>凡例:</b>	r = 予約済みビット	P = プログラミング可能ビット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア x = ビットは未知

- bit 31 **FDMTEN:** デッドマンタイム イネーブルビット  
 1 = デッドマンタイムを有効にする (ソフトウェアで無効にできません)  
 0 = デッドマンタイムを無効にする (ソフトウェアで有効にできます)
- bit 30-26 **DMTCNT<4:0>:** デッドマンタイム カウント選択ビット  
 11111 = 予約済み  
 .  
 .  
 11000 = 予約済み  
 10111 =  $2^{31}$  (2147483648)  
 10110 =  $2^{30}$  (1073741824)  
 10101 =  $2^{29}$  (536870912)  
 10100 =  $2^{28}$  (268435456)  
 .  
 .  
 00001 =  $2^9$  (512)  
 00000 =  $2^8$  (256)
- bit 25-24 **FWDTWINSZ<1:0>:** ウォッチドッグ タイマ ウィンドウサイズ ビット  
 11 = ウィンドウサイズは 25%  
 10 = ウィンドウサイズは 37.5%  
 01 = ウィンドウサイズは 50%  
 00 = ウィンドウサイズは 75%
- bit 23 **FWDTEN:** ウォッチドッグ タイマ イネーブルビット  
 1 = ウォッチドッグ タイマを有効にする (ソフトウェアで無効にできません)  
 0 = ウォッチドッグ タイマを無効にする (ソフトウェアで有効にできます)
- bit 22 **WINDIS:** ウォッチドッグ タイマ ウィンドウ イネーブルビット  
 1 = ウォッチドッグ タイマは非ウィンドウモード中  
 0 = ウォッチドッグ タイマはウィンドウモード中
- bit 21 **WDTSPGM:** フラッシュ プログラミング中ウォッチドッグ タイマ停止ビット  
 1 = フラッシュ プログラミング中にウォッチドッグ タイマは停止する  
 0 = フラッシュ プログラミング中にウォッチドッグ タイマは動作する (フラッシュ プログラミング中もアプリケーションの読み出し / 実行のために動作する)

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 41-4: DEVCFG1/ADEVCFG1: デバイス コンフィグレーション ワード 1 レジスタ ( 続き )

bit 20-16 **WDTPS<4:0>**: ウォッチドッグ タイマ ポストスケール選択ビット

10100 = 1:1048576  
10011 = 1:524288  
10010 = 1:262144  
10001 = 1:131072  
10000 = 1:65536  
01111 = 1:32768  
01110 = 1:16384  
01101 = 1:8192  
01100 = 1:4096  
01011 = 1:2048  
01010 = 1:1024  
01001 = 1:512  
01000 = 1:256  
00111 = 1:128  
00110 = 1:64  
00101 = 1:32  
00100 = 1:16  
00011 = 1:8  
00010 = 1:4  
00001 = 1:2  
00000 = 1:1

上記以外の値に設定した場合の動作は、「10100」に設定した場合と同じです。

bit 15-14 **FCKSM<1:0>**: クロック切り換え / 監視選択コンフィグレーション ビット

11 = クロック切り換えとクロック監視の両方を有効にする  
10 = クロック切り換えを無効にし、クロック監視を有効にする  
01 = クロック切り換えを有効にし、クロック監視を無効にする  
00 = クロック切り換えとクロック監視の両方を無効にする

bit 13-11 **予約済み**: 「1」として書き込み

bit 10 **OSCIOFNC**: CLKO イネーブル コンフィグレーション ビット

1 = CLKO 出力を無効にする  
0 = OSCOピンでCLKO出力信号をアクティブにする(CLKOをアクティブにするには、POSCMOD<1:0>を  
プライマリ オシレータ無効(=11)または外部クロック(EC)モード(=00)に設定する必要があります)

bit 9-8 **POSCMOD<1:0>**: プライマリ オシレータ コンフィグレーション ビット

11 = Posc を無効にする  
10 = HS オシレータモードを選択する  
01 = 予約済み  
00 = EC モードを選択する

bit 7 **IESO**: 内部 / 外部切り換えビット

1 = 内部 / 外部切り換えモードを有効にする (2 段階起動を有効にする )  
0 = 内部 / 外部切り換えモードを無効にする (2 段階起動を無効にする )

bit 6 **FSOSCEN**: セカンダリ オシレータ イネーブルビット

1 = Sosc を有効にする  
0 = Sosc を無効にする

bit 5-3 **DMTINV<2:0>**: デッドマンタイマ カウント ウィンドウ インターバル ビット

111 = ウィンドウ / インターバル値は 127/128 カウンタ値  
110 = ウィンドウ / インターバル値は 63/64 カウンタ値  
101 = ウィンドウ / インターバル値は 31/32 カウンタ値  
100 = ウィンドウ / インターバル値は 15/16 カウンタ値  
011 = ウィンドウ / インターバル値は 7/8 カウンタ値  
010 = ウィンドウ / インターバル値は 3/4 カウンタ値  
001 = ウィンドウ / インターバル値は 1/2 カウンタ値  
000 = ウィンドウ / インターバル値は 0

# PIC32MZ グラフィック (DA) ファミリ

---

レジスタ 41-4: DEVCFG1/ADEVCFG1: デバイス コンフィグレーションワード1 レジスタ (続き)

bit 2-0 **FNOSC<2:0>**: オシレータ選択ビット

111 = SPLL

110 = 予約済み

101 = LPRC

100 = Sosc

011 = 予約済み

010 = Posc (HS、EC)

001 = SPLL

000 = FRC を FRCDIV<2:0> ビット (FRCDIV) により分周

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-5: DEVCFG2/ADEVCFG2: デバイス コンフィグレーションワード 2 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	r-1 —	R/P UPLLFSEL	r-1 —	R/P FDSEN	R/P DSWDTEN	R/P DSWDTOSC	R/P DSWDTPS<4:3>	R/P
23:16	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	DSWDTPS<2:0>			DSBOREN	VBATBOREN	FPLLIDIV<2:0>		
15:8	r-1 —	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	FPLLMULT<6:0>							
7:0	R/P	R/P	R/P	R/P	r-1	R/P	R/P	R/P
	FPLLICK	FPLL RNG<2:0>			—	FPLLIDIV<2:0>		

<b>凡例:</b>	r = 予約済みビット	P = プログラミング可能ビット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア
		x = ビットは未知

- bit 31   **予約済み:** 「1」として書き込み
- bit 30   **UPLLFSEL:** USB PLL 入力周波数選択ビット  
1 = UPLL 入力クロックは 24 MHz  
0 = UPLL 入力クロックは 12 MHz
- bit 29   **予約済み:** 「1」として書き込み
- bit 28   **FDSEN:** ディープスリープ イネーブルビット  
1 = WAIT 命令でディープスリープに移行する  
0 = WAIT 命令でスリープに移行する
- bit 27   **DSWDTEN:** ディープスリープ ウォッチドッグ タイマ イネーブルビット  
1 = ディープスリープ中にディープスリープ ウォッチドッグ タイマ (DSWDT) を有効にする  
0 = ディープスリープ中に DSWDT を無効にする
- bit 26   **DSWDTOSC:** ディープスリープ ウォッチドッグ タイマ参照クロック選択ビット  
1 = LPRC オシレータを DSWDT 参照クロックとして選択する  
0 = セカンダリ オシレータを DSWDT 参照クロックとして選択する

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 41-5: DEVCFG2/ADEVCFG2: デバイス コンフィグレーションワード2 レジスタ (続き)

bit 25-21 **DSWDTPS<4:0>**: ディープスリープ ウォッチドッグ タイマ ポストスケール選択ビット

11111 = 1:2<sup>36</sup>  
11110 = 1:2<sup>35</sup>  
11101 = 1:2<sup>34</sup>  
11100 = 1:2<sup>33</sup>  
11011 = 1:2<sup>32</sup>  
11010 = 1:2<sup>31</sup>  
11001 = 1:2<sup>30</sup>  
11000 = 1:2<sup>29</sup>  
10111 = 1:2<sup>28</sup>  
10110 = 1:2<sup>27</sup>  
10101 = 1:2<sup>26</sup>  
10100 = 1:2<sup>25</sup>  
10011 = 1:2<sup>24</sup>  
10010 = 1:2<sup>23</sup>  
10001 = 1:2<sup>22</sup>  
10000 = 1:2<sup>21</sup>  
01111 = 1:2<sup>20</sup>  
01110 = 1:2<sup>19</sup>  
01101 = 1:2<sup>18</sup>  
01100 = 1:2<sup>17</sup>  
01011 = 1:2<sup>16</sup>  
01010 = 1:2<sup>15</sup>  
01001 = 1:2<sup>14</sup>  
01000 = 1:2<sup>13</sup>  
00111 = 1:2<sup>12</sup>  
00110 = 1:2<sup>11</sup>  
00101 = 1:2<sup>10</sup>  
00100 = 1:2<sup>9</sup>  
00011 = 1:2<sup>8</sup>  
00010 = 1:2<sup>7</sup>  
00001 = 1:2<sup>6</sup>  
00000 = 1:2<sup>5</sup>

bit 20 **DSBOREN**: ディープスリープ BOR イネーブルビット

1 = ディープスリープ中に BOR を有効にする  
0 = ディープスリープ中に BOR を無効にする

bit 19 **VBATBOREN**: VBAT BOR イネーブルビット

1 = VBAT モード中に BOR を有効にする  
0 = VBAT モード中に BOR を無効にする

bit 18-16 **FPLL0DIV<2:0>**: 既定値システム PLL 出力分周比ビット

111 = PLL 出力を 32 分周する  
110 = PLL 出力を 32 分周する  
101 = PLL 出力を 32 分周する  
100 = PLL 出力を 16 分周する  
011 = PLL 出力を 8 分周する  
010 = PLL 出力を 4 分周する  
001 = PLL 出力を 2 分周する  
000 = PLL 出力を 2 分周する

bit 15 **予約済み**: 「1」として書き込み

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 41-5: DEVCFG2/ADEVCFG2: デバイス コンフィグレーションワード2 レジスタ (続き)

bit 14-8 **FPLLMULT<6:0>**: システム PLL フィードバック分周比ビット

11111111 = 128 通倍する  
11111110 = 127 通倍する  
11111101 = 126 通倍する  
11111100 = 125 通倍する

•  
•  
•

00000000 = 1 通倍する

bit 7 **FPLLICKL**: システム PLL 入カクロック選択ビット

1 = FRC をシステム PLL への入力として選択する  
0 = Posc をシステム PLL への入力として選択する

bit 6-4 **FPLLNRNG<2:0>**: システム PLL 分周後入カクロック周波数レンジビット

111 = 予約済み  
110 = 予約済み  
101 = 34 ~ 64 MHz  
100 = 21 ~ 42 MHz  
011 = 13 ~ 26 MHz  
010 = 8 ~ 16 MHz  
001 = 5 ~ 10 MHz  
000 = バイパス

bit 3 **予約済み**: 「1」として書き込み

bit 2-0 **FPLLIDIV<2:0>**: PLL 入力分周比ビット

111 = 8 分周する  
110 = 7 分周する  
101 = 6 分周する  
100 = 5 分周する  
011 = 4 分周する  
010 = 3 分周する  
001 = 2 分周する  
000 = 1 分周する



# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-6: DEVCFG3/ADEVCFG3: デバイス コンフィグレーションワード3 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	r-1 —	r-1 —	R/P IOL1WAY	R/P PMDL1WAY	R/P PGL1WAY	r-1 —	R/P FETHIO	R/P FMIEN
23:16	r-1 —	r-1 —	r-1 —	r-1 —	R/P EXTDDRSIZE<3:0>	R/P	R/P	R/P
15:8	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	USERID<15:8>							
7:0	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	USERID<7:0>							

<b>凡例:</b>	r = 予約済みビット	P = プログラミング可能ビット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア x = ビットは未知

bit 31-30 **予約済み:** 「1」として書き込み

bit 29 **IOL1WAY:** ペリフェラルピンセレクトコンフィグレーションビット

1 = 1 回の設定変更のみ許可する

0 = 複数回の設定変更を許可する

bit 28 **PMDL1WAY:** 周辺モジュール無効化コンフィグレーションビット

1 = 1 回の設定変更のみ許可する

0 = 複数回の設定変更を許可する

bit 27 **PGL1WAY:** パーミッショングループコンフィグレーションビット

1 = 1 回の設定変更のみ許可する

0 = 複数回の設定変更を許可する

bit 26 **予約済み:** 「1」として書き込み

bit 25 **FETHIO:** Ethernet I/O ピン選択ビット

1 = 既定値 Ethernet I/O ピン

0 = 代替 Ethernet I/O ピン

デバイスが代替 Ethernet ピンを備えていない場合、このビットは無視されます。

bit 24 **FMIEN:** Ethernet MII イネーブルビット

1 = MII を有効にする

0 = RMII を有効にする

bit 23-20 **予約済み:** 「1」として書き込み

bit 19-16 **EXTDDRSIZE<3:0>:** 外部 DDR2 SDRAM 容量ビット

このフィールドは、DDR2 メモリマップのコンフィグレーション用に使います。アドレス割り当ての詳細は表 4-1 を参照してください。

1111 = 128 MB

1110 = 128 MB

.

.

.

0111 = 128 MB

0110 = 64 MB

0101 = 32 MB

0100 = 16 MB

0011 = 8 MB

0010 = 4 MB

0001 = 2 MB

0000 = 1 MB

bit 15-0 **USERID<15:0>:** これはユーザが定義する 16 ビットのユーザ ID 値です。この値は ICSP™ および JTAG を介して読み出せます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-7: DEVCFG4/ADEVCFG4: デバイス コンフィグレーションワード 4 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	r-1 —	r-1 —	r-1 —	R/P —	R/P —	R/P —	R/P —	R/P —
	SWDTPS<4:0>							
23:16	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —
15:8	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —
7:0	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —	r-1 —

<b>凡例:</b>	r = 予約済みビット	P = プログラム可能ビット
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装、「0」として読み出し
-n = POR 時の値	1 = ビットはセット	0 = ビットはクリア      x = ビットは未知

bit 31-29 **予約済み**: 「1」として書き込み

bit 29-24 **SWDTPS<4:0>**: スリープウォッチドッグタイマポストスケール選択ビット

10100 = 1:1048576  
 10011 = 1:524288  
 10010 = 1:262144  
 10001 = 1:131072  
 10000 = 1:65536  
 01111 = 1:32768  
 01110 = 1:16384  
 01101 = 1:8192  
 01100 = 1:4096  
 01011 = 1:2048  
 01010 = 1:1024  
 01001 = 1:512  
 01000 = 1:256  
 00111 = 1:128  
 00110 = 1:64  
 00101 = 1:32  
 00100 = 1:16  
 00011 = 1:8  
 00010 = 1:4  
 00001 = 1:2  
 00000 = 1:1

上記以外の値に設定した場合の動作は、「10100」に設定した場合と同じです。

bit 31-29 **予約済み**: 「1」として書き込み

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-8: DEVADCx: デバイス ADC 校正ワード「x」レジスタ (x = 0 ~ 4、7)

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R	R	R	R	R	R	R	R
	ADCFG<31:24>							
23:16	R	R	R	R	R	R	R	R
	ADCFG<23:16>							
15:8	R	R	R	R	R	R	R	R
	ADCFG<15:8>							
7:0	R	R	R	R	R	R	R	R
	ADCFG<7:0>							

**凡例:**

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装、「0」として読み出し  
 -n = POR 時の値 1 = ビットはセット 0 = ビットはクリア x = ビットは未知

bit 31-0 **ADCFG<31:0>**: ADC モジュール校正データビット  
 このデータは、対応する ADCxCFG レジスタにコピーする必要があります。詳細は [29.0 「高速 12 ビット逐次比較型レジスタ \(SAR\) A/D コンバータ \(ADC\)」](#) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-9: CFGCON: コンフィグレーション制御レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	ICACLK <sup>(1)</sup>	OCACLK <sup>(1)</sup>
15:8	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
	—	—	IOLOCK <sup>(1)</sup>	PMDLOCK <sup>(1)</sup>	PGLOCK <sup>(1)</sup>	—	—	USBSEN <sup>(1)</sup>
7:0	R/W-0	U-0	R/W-1	R/W-1	R/W-1	R/W-0	U-0	R/W-1
	IOANCPEN	—	ECCCON<1:0>		JTAGEN <sup>(2)</sup>	TROEN	—	TDOEN

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-18 **未実装:** 「0」として読み出し

bit 17 **ICACLK:** 入力キャプチャ代替クロック選択ビット<sup>(1)</sup>

1 = 入力キャプチャ モジュールは代替タイマペアをタイムベース クロックとして使う  
 0 = 全ての入力キャプチャ モジュールは Timer2/3 をタイムベース クロックとして使う

bit 16 **OCACLK:** 出力コンペア代替クロック選択ビット<sup>(1)</sup>

1 = 出力コンペア モジュールは代替タイマペアをタイムベース クロックとして使う  
 0 = 全ての出力コンペア モジュールは Timer2/3 をタイムベース クロックとして使う

bit 15-14 **未実装:** 「0」として読み出し

bit 13 **IOLOCK:** ペリフェラル ピンセレクト ロック ビット<sup>(1)</sup>

1 = ペリフェラル ピンセレクト機能をロックする (PPS レジスタへの書き込みを許可しない)  
 0 = ペリフェラル ピンセレクト機能をロックしない (PPS レジスタへの書き込みを許可する)

bit 12 **PMDLOCK:** 周辺モジュール無効化ロックビット<sup>(1)</sup>

1 = 周辺モジュールの無効化をロックする (PMD レジスタへの書き込みを許可しない)  
 0 = 周辺モジュールの無効化をロックしない (PMD レジスタへの書き込みを許可する)

bit 11 **PGLOCK:** パーミッショングループロックビット<sup>(1)</sup>

1 = パーミッショングループレジスタをロックする (PG レジスタへの書き込みを許可しない)  
 0 = パーミッショングループレジスタをロックしない (PG レジスタへの書き込みを許可する)

bit 10-9 **未実装:** 「0」として読み出し

bit 8 **USBSEN:** スリープ中 USB サスペンド イネーブルビット<sup>(1)</sup>

スリープ中に USB PHY クロック シャットダウン機能を有効にします。

1 = スリープ中に USB PHY クロックはシャットダウンする  
 0 = スリープ中に USB PHY クロックは動作を継続する

**Note 1:** このビットを変更するには、ロック解除シーケンスを実行する必要があります。『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

**2:** 起動時に JTAGEN (DEVCFG0<2>) ヒューズビットがセットされた場合にのみ、この JTAGEN ビットを実行時に使う事ができます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 41-9: CFGCON: コンフィグレーション制御レジスタ ( 続き )

bit 7 **IOANCPEN:** I/O アナログ チャージポンプ イネーブルビット

- 1 = チャージポンプを有効にする
- 0 = チャージポンプを無効にする

**Note 1:** 2.5V を下回る VDD で適正なアナログ動作を維持するには、AICPMPEN ビット (ADCCON1<12>) と IOANCPEN ビットを「1」にセットする必要がありますが、チャージポンプの消費電流は増加します。VDD が 2.5V を上回る場合、これらのビットをセットする必要はありません。

**2:** ADCCON1<AICPMPEN> = 1 かつ CFGCON<IOANCPEN> = 1 の場合、ADC のスループットレートは下表の通りに低下します。

ADC0	ADC1	ADC2	ADC3	ADC4	ADC7	Maximum combined
ON	OFF	OFF	OFF	OFF	OFF	2 MSPS
ON	ON	OFF	OFF	OFF	OFF	4 MSPS
ON	ON	ON	OFF	OFF	OFF	5 MSPS
OFF	OFF	OFF	ON	OFF	OFF	2 MSPS
OFF	OFF	OFF	ON	ON	OFF	4 MSPS
OFF	OFF	OFF	ON	ON	ON	5 MSPS
ON	ON	ON	ON	OFF	OFF	7 MSPS
ON	ON	ON	ON	ON	OFF	9 MSPS
ON	ON	ON	ON	ON	ON	10 MSPS

bit 6 **未実装:** 「0」として読み出し

bit 5-4 **ECCCON<1:0>:** フラッシュ ECC コンフィグレーション ビット

11 = ECC および動的 ECC を無効にする (ECCCON<1:0> ビットは書き込み可能)

10 = ECC および動的 ECC を無効にする (ECCCON<1:0> ビットをロック)

01 = 動的フラッシュ ECC を有効にする (ECCCON<1:0> ビットをロック)

00 = フラッシュ ECC を有効にする (ECCCON<1:0> ビットをロック、ワードフラッシュ書き込みは無効)

bit 3 **JTAGEN:** JTAG ポート イネーブルビット<sup>(2)</sup>

1 = JTAG ポートを有効にする

0 = JTAG ポートを無効にする

bit 2 **TROEN:** トレース出力イネーブルビット

1 = トレース出力を有効にし、トレースクロックを起動する (トレースプローブが必要です)

0 = トレース出力を無効にし、トレースクロックを停止する

bit 1 **未実装:** 「0」として読み出し

bit 0 **TDOEN:** 2 線式 JTAG TDO イネーブルビット

1 = 2 線式 JTAG プロトコルは TDO を使う

0 = 2 線式 JTAG プロトコルは TDO を使わない

**Note 1:** このビットを変更するには、ロック解除シーケンスを実行する必要があります。『PIC32 ファミリ リファレンス マニュアル、セクション 42. 拡張 PLL を備えたオシレータ』(DS60001250) を参照してください。

**2:** 起動時に JTAGEN (DEVCFG0<2>) ヒューズビットがセットされた場合にのみ、この JTAGEN ビットを実行時に使う事ができます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-10: CFGEBIA: 外部バス インターフェイス アドレスピン コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EBIA23EN	EBIA22EN	EBIA21EN	EBIA20EN	EBIA19EN	EBIA18EN	EBIA17EN	EBIA16EN
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EBIA15EN	EBIA14EN	EBIA13EN	EBIA12EN	EBIA11EN	EBIA10EN	EBIA9EN	EBIA8EN
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EBIA7EN	EBIA6EN	EBIA5EN	EBIA4EN	EBIA3EN	EBIA2EN	EBIA1EN	EBIA0EN

**凡例:**

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                                1 = ビットはセット            0 = ビットはクリア            x = ビットは未知

bit 31-24 **未実装:** 「0」として読み出し

bit 23-0 **EBIA23EN:EBIA0EN:** EBI アドレスピン イネーブルビット

1 = EBIAx ピンを EBI モジュール用に使う  
 0 = EBIAx ピンは汎用ピンとして使える

**Note:** EBIMD = 1 に設定した場合、このレジスタ内のビットは無視され、対応するピンは他の用途に使えます。

# PIC32MZ グラフィック (DA) ファミリ

レジスタ 41-11: CFGEBIC: 外部バス インターフェイス制御ピン コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0
	EBI RDYINV3	EBI RDYINV2	EBI RDYINV1	—	EBI RDYEN3	EBI RDYEN2	EBI RDYEN1	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
	—	—	—	—	—	—	EBIRDYLVL	EBIRPEN
15:8	U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
	—	—	EBIWEEN	EBIOEEN	—	—	EBIBSEN1	EBIBSEN0
7:0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
	EBICSEN3	EBICSEN2	EBICSEN1	EBICSEN0	—	—	EBIDEN1	EBIDEN0

**凡例:**

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31 **EBIRDYINV3:** EBIRDY3 反転制御ビット

1 = EBIRDY3 ピンを使用前に反転する  
0 = EBIRDY3 ピンを使用前に反転しない

bit 30 **EBIRDYINV2:** EBIRDY2 反転制御ビット

1 = EBIRDY2 ピンを使用前に反転する  
0 = EBIRDY2 ピンを使用前に反転しない

bit 29 **EBIRDYINV1:** EBIRDY1 反転制御ビット

1 = EBIRDY1 ピンを使用前に反転する  
0 = EBIRDY1 ピンを使用前に反転しない

bit 28 **未実装:** 「0」として読み出し

bit 27 **EBIRDYEN3:** EBIRDY3 ピン イネーブルビット

1 = EBIRDY3 ピンを EBI モジュール用に使う  
0 = EBIRDY3 ピンは汎用ピンとして使える

bit 26 **EBIRDYEN2:** EBIRDY2 ピン イネーブルビット

1 = EBIRDY2 ピンを EBI モジュール用に使う  
0 = EBIRDY2 ピンは汎用ピンとして使える

bit 25 **EBIRDYEN1:** EBIRDY1 ピン イネーブルビット

1 = EBIRDY1 ピンを EBI モジュール用に使う  
0 = EBIRDY1 ピンは汎用ピンとして使える

bit 24-18 **未実装:** 「0」として読み出し

bit 17 **EBIRDYLVL:** EBIRDYx ピン レベル/エッジ センシティブ制御ビット

1 = EBIRDYx ピンにレベル検出を適用する  
0 = EBIRDYx ピンにエッジ検出を適用する

bit 16 **EBIRPEN:** EBIRP ピン イネーブルビット

1 = EBIRP ピンを EBI モジュール用に使う  
0 = EBIRP ピンは汎用ピンとして使える

bit 15-14 **未実装:** 「0」として読み出し

bit 13 **EBIWEEN:** EBIWE ピン イネーブルビット

1 = EBIWE ピンを EBI モジュール用に使う  
0 = EBIWE ピンは汎用ピンとして使える

**Note:** EBIMD = 1 に設定した場合、このレジスタ内のビットは無視され、対応するピンは他の用途に使えます。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 41-11: CFGBIC: 外部バス インターフェイス制御ピン コンフィグレーション レジスタ ( 続き

- bit 12 **EBIOEEN:**  $\overline{\text{EBIOE}}$  ピン イネーブルビット  
1 =  $\overline{\text{EBIOE}}$  ピンを EBI モジュール用に使う  
0 =  $\overline{\text{EBIOE}}$  ピンは汎用ピンとして使える
- bit 11-10 **未実装:** 「0」 として読み出し
- bit 9 **EBIBSEN1:**  $\overline{\text{EBIBS1}}$  ピン イネーブルビット  
1 =  $\overline{\text{EBIBS1}}$  ピンを EBI モジュール用に使う  
0 =  $\overline{\text{EBIBS1}}$  ピンは汎用ピンとして使える
- bit 8 **EBIBSEN0:**  $\overline{\text{EBIBS0}}$  ピン イネーブルビット  
1 =  $\overline{\text{EBIBS0}}$  ピンを EBI モジュール用に使う  
0 =  $\overline{\text{EBIBS0}}$  ピンは汎用ピンとして使える
- bit 7 **EBICSEN3:**  $\overline{\text{EBICS3}}$  ピン イネーブルビット  
1 =  $\overline{\text{EBICS3}}$  ピンを EBI モジュール用に使う  
0 =  $\overline{\text{EBICS3}}$  ピンは汎用ピンとして使える
- bit 6 **EBICSEN2:**  $\overline{\text{EBICS2}}$  ピン イネーブルビット  
1 =  $\overline{\text{EBICS2}}$  ピンを EBI モジュール用に使う  
0 =  $\overline{\text{EBICS2}}$  ピンは汎用ピンとして使える
- bit 5 **EBICSEN1:**  $\overline{\text{EBICS1}}$  ピン イネーブルビット  
1 =  $\overline{\text{EBICS1}}$  ピンを EBI モジュール用に使う  
0 =  $\overline{\text{EBICS1}}$  ピンは汎用ピンとして使える
- bit 4 **EBICSEN0:**  $\overline{\text{EBICS0}}$  ピン イネーブルビット  
1 =  $\overline{\text{EBICS0}}$  ピンを EBI モジュール用に使う  
0 =  $\overline{\text{EBICS0}}$  ピンは汎用ピンとして使える
- bit 3-2 **未実装:** 「0」 として読み出し
- bit 1 **EBIDEN1:** EBI データ上位バイトピン イネーブルビット  
1 = EBID<15:8> ピンを EBI モジュール用に使う  
0 = EBID<15:8> ピンは汎用ピンとして使える
- bit 0 **EBIDEN01:** EBI データ下位バイトピン イネーブルビット  
1 = EBID<7:0> ピンを EBI モジュール用に使う  
0 = EBID<7:0> ピンは汎用ピンとして使える

**Note:** EBIMD = 1 に設定した場合、このレジスタ内のビットは無視され、対応するピンは他の用途に使えます。



# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 41-12: CFGPG: パーミッショングループコンフィグレーションレジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	GPUPG<1:0>		GLCDPG<1:0>		CRYPTPG<1:0>	
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	FCPG<1:0>		SQ1PG<1:0>		SDHCPG<1:0>		ETHPG<1:0>	
15:8	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
	CAN2PG<1:0>		CAN1PG<1:0>		—	—	USBPG<1:0>	
7:0	U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
	—	—	DMPAG<1:0>		—	—	CPUPG<1:0>	

### 凡例:

R = 読み出し可能ビット                      W = 書き込み可能ビット    U = 未実装、「0」として読み出し  
 -n = POR 時の値                              1 = ビットはセット        0 = ビットはクリア

bit 31-30 **未実装**: 「0」として読み出し

bit 29-28 **GPUPG<1:0>**: 2D グラフィック処理ユニット パーミッション グループビット

11 = イニシエータをパーミッショングループ3に割り当てる  
 10 = イニシエータをパーミッショングループ2に割り当てる  
 01 = イニシエータをパーミッショングループ1に割り当てる  
 00 = イニシエータをパーミッショングループ0に割り当てる

bit 27-26 **GLCDPG<1:0>**: グラフィック LCD コントローラ パーミッション グループビット

定義は bit 29-28 と同じです。

bit 25-24 **CRYPTPG<1:0>**: 暗号エンジン パーミッション グループビット

定義は bit 29-28 と同じです。

bit 23-22 **FCPG<1:0>**: フラッシュ制御パーミッション グループビット

定義は bit 29-28 と同じです。

bit 21-20 **SQ1PG<1:0>**: SQI モジュール パーミッション グループビット

定義は bit 29-28 と同じです。

bit 19-18 **SDHCPG<1:0>**: セキュア デジタルホスト コントローラ パーミッション グループビット

定義は bit 29-28 と同じです。

bit 17-16 **ETHPG<1:0>**: Ethernet モジュール パーミッション グループビット

定義は bit 29-28 と同じです。

bit 15-14 **CAN2PG<1:0>**: CAN2 モジュール パーミッション グループビット

定義は bit 29-28 と同じです。

bit 13-12 **CAN1PG<1:0>**: CAN1 モジュール パーミッション グループビット

定義は bit 29-28 と同じです。

bit 11-10 **未実装**: 「0」として読み出し

bit 9-8 **USBPG<1:0>**: USB モジュール パーミッション グループビット

定義は bit 29-28 と同じです。

bit 7-6 **未実装**: 「0」として読み出し

bit 5-4 **DMPAG<1:0>**: DMA モジュール パーミッション グループビット

定義は bit 29-28 と同じです。

bit 3-2 **未実装**: 「0」として読み出し

bit 1-0 **CPUPG<1:0>**: CPU パーミッション グループビット

定義は bit 29-28 と同じです。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 41-13: CFGCON2: コンフィグレーション制御レジスタ 2 レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
	GLCDPINEN	GLCDMODE <sup>(1)</sup>	SDCDEN	SDWPEN	—	—	SDWRFTHR<9:8>	
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	SDWRFTHR<7:0>							
15:8	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	SDRDFTHR<9:4>					
7:0	R/W-0	R/W-0	R/W-0	R/W-0	r-1	R/W-0	U-0	R/W-0
	SDRDFTHR<3:0>				—	SDWPPOL	—	GPURESET

### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット U = 未実装、「0」として読み出し

-n = POR 時の値

1 = ビットはセット

0 = ビットはクリア

x = ビットは未知

bit 31 **GLCDPINEN:** グラフィック ディスプレイピン イネーブルビット

1 = GLCD モジュールは GLCD ピンを使う

0 = GLCD ピンは汎用ピンとして使える

bit 30 **GLCDMODE:** グラフィック ディスプレイ モードビット<sup>(1)</sup>

1 = GLCD ピンを RGB565 モードに設定する (他の GDx ピンは汎用ピンとして使えます)

0 = GLCD ピンを RGB888 モードに設定する

bit 29 **SDCDEN:** SD カード検出ピン イネーブルビット

1 = SDCD ピンを SDHC モジュール用に使う

0 = SDCD ピンは汎用ピンとして使える

bit 28 **SDWPEN:** SD カード書き込み保護イネーブルビット

1 = SDWP ピンを SDHC モジュール用に使う

0 = SDWP ピンは汎用ピンとして使える

bit 27-26 **未実装:** 「0」として読み出し

bit 25-16 **SDWRFTHR<9:0>:** SDHC FIFO 書き込みしきい値ビット

SDHC FIFO しきい値をバイト数で指定します (FIFO のサイズは 512 バイト)。

bit 15-14 **未実装:** 「0」として読み出し

bit 13-4 **SDRDFTHR<9:0>:** SDHC FIFO 読み出ししきい値ビット

SDHC FIFO しきい値をバイト数で指定します (FIFO のサイズは 512 バイト)。

bit 3 **予約済み:** 「1」として読み出し

bit 2 **SDWPPOL:** SD カード書き込み保護極性ビット

1 = SDWP ピンをアクティブ HIGH にする

0 = SDWP ピンをアクティブ LOW にする

**Note:** このビットは、書き込み保護の極性が異なる SD カードをサポートするために使います。

bit 1 **未実装:** 「0」として読み出し

bit 0 **GPURESET:** GPU リセットビット

1 = GPU モジュールに対する RESET を解放する

0 = GPU を RESET に保持する

**Note:** このビットは、実行時に GPU 機能を有効または無効にする場合にのみ使います。このビットに書き込むには、GPUMD ビット (PMD6<18>) が「0」(GPU は有効) に設定されている必要があります。

**Note 1:** GLCD を RGB888 モードで使う場合、GLCDMODE ビットを「0」に設定する必要があります。これにより、他の 6 本のピンの汎用 I/O 機能は無効になります。GDx ピンの共有については、[デバイスピン一覧](#)内の対応するパッケージのテーブルを参照してください。

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 41-14: CFGMPLL: メモリ PLL コンフィグレーション レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	MPLLRDY	MPLLDIS	MPLLLODIV2<2:0>			MPLLLODIV1<2:0>		
23:16	R-0	R/W-1	U-0	U-0	U-0	U-0	U-0	U-0
	MPLL VREGRDY	MPLL VREGDIS	—	—	—	—	—	—
15:8	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	MPLLMULT<7:0>							
7:0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	INTVREFCON<1:0>		MPLLIDIV<5:0>					

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31 **MPLLRDY:** メモリ PLL ステータスビット

1 = MPLL クロックは安定している (動作準備は完了している)  
 0 = MPLL クロックの動作準備は未完了  
 クロックの動作準備が未完了の時に DDR2 SDRAM を初期化した場合の動作は未確定です。

bit 30 **MPLLDIS:** MPLL ディセーブル ビット

1 = MPLL を無効にする  
 0 = MPLL を有効にする

**Note:** このビットは、MPLLVREGRDY ビットが「1」にセットされた後にクリアする必要があります。

bit 29-27 **MPLLLODIV2<2:0>:** MPLL 出力分周比 2 ビット

111 = MPLL の第 2 段出力を 7 分周する  
 110 = MPLL の第 2 段出力を 6 分周する  
 101 = MPLL の第 2 段出力を 5 分周する  
 100 = MPLL の第 2 段出力を 4 分周する  
 011 = MPLL の第 2 段出力を 3 分周する  
 010 = MPLL の第 2 段出力を 2 分周する  
 001 = MPLL の第 1 段出力を 1 分周する  
 000 = 予約済み

**Note:** このビットフィールドの値は、MPLLLODIV1 より小さい必要があります。クロックジッタを低減するため、必要な場合を除き、これらのビットは「001」(MPLL の第 2 段出力を 1 分周する) に設定します。

bit 26-24 **MPLLLODIV1<2:0>:** MPLL 出力分周比 1 ビット

定義は bit 29-27 と同じです。

bit 23 **MPLLVREGRDY:** MPLL 電圧レギュレータ レディービット

1 = MPLL 電圧レギュレータの動作準備は完了している  
 0 = MPLL 電圧レギュレータの動作準備は未完了

bit 22 **MPLLVREGDIS:** MPLL 電圧レギュレータ ディセーブル ビット

1 = MPLL 電圧レギュレータを無効にする  
 0 = MPLL 電圧レギュレータを有効にする

bit 21-16 **未実装:** 「0」として読み出し

# PIC32MZ グラフィック (DA) ファミリ

---

## レジスタ 41-14: CFGMPLL: メモリ PLL コンフィグレーション レジスタ ( 続き )

bit 15-8 **MPLLMULT<7:0>**: MPLL 通倍比ビット

11111111 = 予約済み  
11111110 = 予約済み  
.  
.  
.  
10100001 = 予約済み  
10100000 = 160 通倍する  
10011111 = 159 通倍する  
.  
.  
.  
00010000 = 16 通倍する  
00001111 = 予約済み  
.  
.  
.  
00000000 = 予約済み

bit 7-6 **INTVREFCON<1:0>**: 内部 DDRVREF 制御ビット

11 = 内部 DDRVREF 回路を有効にする  
10 = 内部 DDRVREF 回路を無効にし、DDRVREF ピンを V<sub>SS1V8</sub> に駆動する  
01 = 内部 DDRVREF 回路を無効にし、DDRVREF ピンを V<sub>DDR1V8</sub> に駆動する  
00 = 外部の DDRVREF 回路を使う

**Note:** INTVREFCON<1:0> ビットは、V<sub>DDR1V8</sub> を印加する前に設定する必要があります。

bit 5-0 **MPLLIDIV<5:0>**: MPLL 入力分周比ビット

111111 = MPLL 入力クロックを 63 分周する  
111110 = MPLL 入力クロックを 62 分周する  
.  
.  
.  
000001 = MPLL 入力クロックを 1 分周する  
000000 = 予約済み

# PIC32MZ グラフィック (DA) ファミリ

## レジスタ 41-15: DEVID: デバイスおよびリビジョン ID レジスタ

ビット レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R	R	R	R	R	R	R	R
	VER<3:0> <sup>(1)</sup>				DEVID<27:24> <sup>(1)</sup>			
23:16	R	R	R	R	R	R	R	R
	DEVID<23:16> <sup>(1)</sup>							
15:8	R	R	R	R	R	R	R	R
	DEVID<15:8> <sup>(1)</sup>							
7:0	R	R	R	R	R	R	R	R
	DEVID<7:0> <sup>(1)</sup>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-28 **VER<3:0>**: リビジョン ID ビット <sup>(1)</sup>

bit 27-0 **DEVID<27:0>**: デバイス ID ビット <sup>(1)</sup>

**Note 1:** リビジョン ID 値とデバイス ID 値の一覧は『PIC32 フラッシュ プログラミング仕様』(DS60001145) に記載しています。

## レジスタ 41-16: DEVSNx: デバイス シリアル番号レジスタ「x」(x = 0、1)

ビットレ レンジ	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R	R	R	R	R	R	R	R
	SN<31:24>							
23:16	R	R	R	R	R	R	R	R
	SN<23:16>							
15:8	R	R	R	R	R	R	R	R
	SN<15:8>							
7:0	R	R	R	R	R	R	R	R
	SN<7:0>							

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装、「0」として読み出し  
 -n = POR 時の値      1 = ビットはセット      0 = ビットはクリア      x = ビットは未知

bit 31-0 **SN<31:0>**: デバイス一意シリアル番号ビット

# PIC32MZ グラフィック (DA) ファミリ

## 41.3 VDDR1V8 に対する高電圧検出 (HVD1V8)

高電圧検出 (HVD) モジュールは、VDDR1V8 電源電圧 (1.8 V) で DDR2 PHY 電圧を監視します。危険な高電圧が検出されると、HVD 条件が続く限りデバイスはリセット状態に保たれます。

HVD イベントからの復帰は、HVD1V8R ビット (RCON<29>) によって示されます。

## 41.4 内蔵電圧レギュレータ

全ての PIC32MZ DA デバイスのコアおよびデジタルロジックは、公称 1.8 V で動作するように設計されています。システム設計を簡潔にするため、PIC32MZ DA ファミリは VDDIO から必要なコア電圧を生成するための電圧レギュレータを内蔵しています。

### 41.4.1 内蔵レギュレータと POR

内蔵レギュレータによる出力電圧の生成には一定の遅延が適用されます。この遅延時間 (TPU) が過ぎるまで、コード実行は無効にされます。TPU は、デバイスが全ての形態のパワーダウン (スリープを含む) から動作を再開するたびに適用されます。

### 41.4.2 内蔵レギュレータと BOR

PIC32MZ DA は、簡潔なブラウンアウト機能も備えています。レギュレータに供給される電圧が、出力電圧を適正に維持するために必要なレベルを下回ると、レギュレータ リセット回路によってブラウンアウト リセット (BOR) が発生します。このイベントが発生した事は、BOR フラグビット (RCON<1>) によって示されます。ブラウンアウトの電圧レベルは [44.1 「DC 特性」](#) を参照してください。

## 41.5 内蔵温度センサ

PIC32MZ DA は、デバイスの接合部温度を正確に計測するための温度センサを内蔵しています ([44.2 「AC 特性とタイミングパラメータ」](#) 参照)。

温度センサは ADC モジュールに接続しており、共有 S&H 回路を使って計測できます ([29.0 「高速 12 ビット逐次比較型レジスタ \(SAR\) A/D コンバータ \(ADC\)」](#) 参照)。

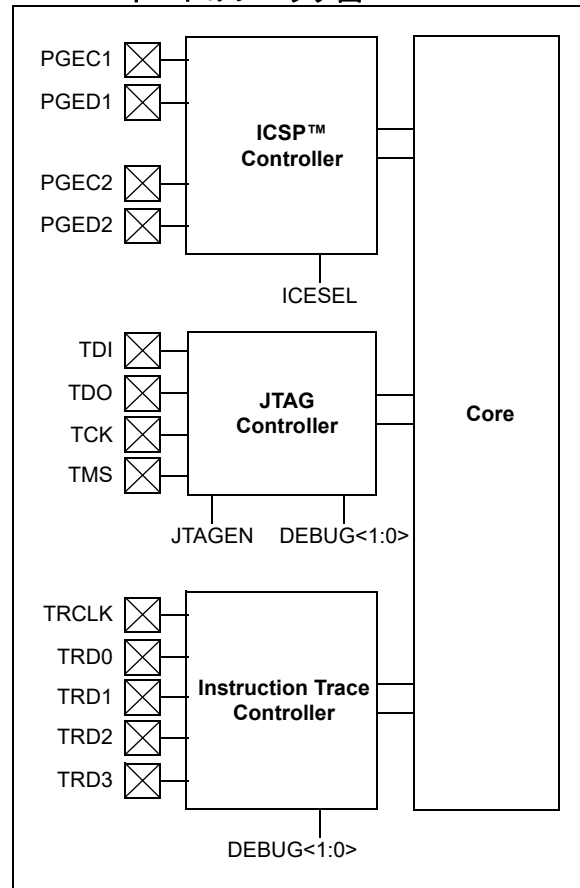
## 41.6 プログラミングと診断

PIC32MZ DA は、アプリケーションの柔軟性を高めるための各種プログラミングおよび診断機能を備えています。これらの機能により、以下が可能です。

- 2 線式インサーキット シリアル プログラミング (ICSP™) インターフェイスを使った簡潔なフィールドプログラミング
- ICSP を使ったデバッグ
- EJTAG ( 拡張 JTAG ) を使ったプログラミングとデバッグ
- JTAG バウンダリ スキャンを使ったデバイスとボードの診断

PIC32 は、2 つのプログラミング / 診断モジュールとトレース コントローラを内蔵する事で、アプリケーション開発に役立つ各種機能を提供します。

図 41-1: プログラミング、デバッグ、トレースポートのブロック図



## 42.0 命令セット

PIC32MZ グラフィック (DA) ファミリファミリの命令セットは、MIPS32<sup>®</sup> リリース 2 命令セットアーキテクチャに準拠します。PIC32MZ DA デバイスファミリは以下の機能をサポートしません。

- コア拡張命令
- コプロセッサ 2 命令

**Note:** 詳細は『MIPS32<sup>®</sup> Architecture for Programmers  
Volume II: The MIPS32<sup>®</sup> Instruction Set』  
([www.imgtec.com](http://www.imgtec.com)) を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:



# PIC32MZ グラフィック (DA) ファミリ

## 43.0 開発サポート

PIC<sup>®</sup> マイクロコントローラ (MCU) と dsPIC<sup>®</sup> デジタルシグナル コントローラ (DSC) は、以下に示す各種ソフトウェア / ハードウェア開発ツールによってサポートされています。

- 統合開発環境 (Integrated Development Environment)
  - MPLAB<sup>®</sup> X IDE ソフトウェア
- コンパイラ / アセンブラ / リンカ
  - MPLAB XC コンパイラ
  - MPASM<sup>™</sup> アセンブラ
  - MPLINK<sup>™</sup> オブジェクト リンカ / MPLIB<sup>™</sup> オブジェクト ライブラリアン
  - 各種デバイスファミリ用 MPLAB アセンブラ / リンカ / ライブラリアン
- シミュレータ
  - MPLAB X SIM ソフトウェア シミュレータ
- エミュレータ
  - MPLAB REAL ICE<sup>™</sup> インサーキット エミュレータ
- インサーキット デバッガ / プログラマ
  - MPLAB ICD 3
  - PICKit<sup>™</sup> 3
- デバイス プログラマ
  - MPLAB PM3 デバイス プログラマ
- 低コストのデモボード、開発ボード、評価キット、スタータキット
- サードパーティ製開発ツール

## 43.1 MPLAB X 統合開発環境ソフトウェア

MPLAB X IDE は、Microchip 社およびサードパーティのソフトウェア / ハードウェア開発ツールに対する共通のグラフィカル ユーザ インターフェイスで、Windows<sup>®</sup>、Linux、Mac OS<sup>®</sup> X で動作します。NetBeans IDE をベースにした MPLAB X IDE は、無償のソフトウェア コンポーネントとプラグインを豊富に揃えた全く新しい IDE で、高性能アプリケーションの開発とデバッグを実行できます。ツール間の移行も、ソフトウェア シミュレータからハードウェア デバッグ / プログラミング ツールへのアップグレードも、このシームレスなユーザ インターフェイスで簡単に行えます。

プロジェクト管理、視覚的なコールグラフ、設定可能なウォッチ ウィンドウ、多機能なエディタ (コード補完、コンテキスト メニュー等) を完備した MPLAB X IDE は、柔軟性に富み新規のユーザでも簡単に使えます。複数のプロジェクトを複数のツールで開きながら同時にデバッグを行う機能も備える等、MPLAB X IDE はベテランユーザのニーズにも対応します。

多機能エディタ：

- 構文の色分け表示
- 入力中に候補とヒントを表示するスマートコード補完
- ユーザ定義のルールに基づくコードの自動フォーマット処理
- リアルタイムの構文解析

カスタマイズ可能で使いやすいインターフェイス：

- 完全にカスタマイズ可能なツールバー、ツールバーのボタン、ウィンドウ、ウィンドウ配置等のインターフェイス
- コールグラフ ウィンドウ

プロジェクト ベースのワークスペース：

- 複数プロジェクト
- 複数ツール
- 複数設定
- 同時デバッグ セッション

ファイル履歴とバグトラッキング：

- ローカルファイル履歴機能
- Bugzilla (バグ追跡システム) をサポート

# PIC32MZ グラフィック (DA) ファミリ

## 43.2 MPLAB XC コンパイラ

MPLAB XC コンパイラは、Microchip 社の 8/16/32 ビット MCU および DSC デバイス全てに対応する完全な ANSI C コンパイラです。これらのコンパイラは強力な統合機能と優れたコード最適化機能を備え、容易に使えます。MPLAB XC コンパイラは Windows、Linux、Mac OS X で動作します。

また、MPLAB X IDE 用に最適化されたデバッグ情報を出力できるため、ソースレベルのデバッグも容易です。

この MPLAB XC コンパイラの Free エディションは、全てのデバイスとコマンドをサポートし、使用期間とメモリ量に制約はありません。また、ほとんどのアプリケーションに十分なコード最適化機能も備えています。

MPLAB XC コンパイラは、アセンブラ、リンカ、ユーティリティを含みます。このアセンブラが生成した再配置可能なオブジェクト ファイルをアーカイブまたは他の再配置可能なオブジェクト ファイルとリンクして、実行ファイルを作成します。MPLAB XC コンパイラはこのアセンブラを使ってオブジェクト ファイルを生成します。このアセンブラには、主に以下の特長があります。

- デバイスの全命令セットのサポート
- 固定 / 浮動小数点データのサポート
- コマンドライン インターフェイス
- 豊富なディレクティブ セット
- 柔軟なマクロ言語
- MPLAB X IDE との互換性

## 43.3 MPASM アセンブラ

MPASM アセンブラは、PIC10/12/16/18 MCU に対応したフル機能の汎用マクロアセンブラです。

MPASM アセンブラは、MPLINK オブジェクト リンカ用の再配置可能なオブジェクト ファイル、Intel® 標準 HEX ファイル、メモリ使用状況とシンボル参照を詳述する MAP ファイル、ソース行と生成後のマシンコードを含む絶対 LST ファイル、デバッグ用の COFF ファイルを生成します。

MPASM アセンブラには、以下の特長があります。

- MPLAB X IDE プロジェクトへの統合
- ユーザ定義マクロによるアセンブリコードの最適化アセンブリコード
- 多用途ソースファイルに対応する条件付きアセンブリ  
#TOIN#
- アセンブリ プロセスを完全に制御できるディレクティブ

## 43.4 MPLINK オブジェクト リンカ / MPLIB オブジェクト ライブラリアン

MPLINK オブジェクト リンカは、MPASM アセンブラで生成された再配置可能なオブジェクトを結合します。このオブジェクト リンカは、リンカスクリプトのディレクティブを使って、プリコンパイル済みのライブラリ内の再配置可能なオブジェクトをリンクできます。

MPLIB オブジェクト ライブラリアンは、プリコンパイル済みコードのライブラリ ファイルの作成と変更を管理します。ライブラリのルーチンをソースファイルから呼び出すと、そのルーチンが含まれているモジュールのみがアプリケーションとリンクされます。これにより、大きなライブラリを各種アプリケーションで効率的に使えます。

オブジェクト リンカ / ライブラリには、以下の特長があります。

- 多数の小さなファイルをリンクするのではなく、1 つのライブラリを効率的にリンクする
- 関連するモジュールをグループ化する事により、コードの保守性が向上する
- モジュールのリスト作成、置換、削除、抽出が容易なライブラリを柔軟に作成する

## 43.5 各種デバイスファミリ用 MPLAB アセンブラ / リンカ / ライブラリアン

MPLAB アセンブラは、PIC24、PIC32、dsPIC DSC 用のシンボリック アセンブリ言語から、再配置可能なマシンコードを作成します。MPLAB XC コンパイラはこのアセンブラを使ってオブジェクト ファイルを生成します。このアセンブラが生成した再配置可能なオブジェクト ファイルをアーカイブまたは他の再配置可能なオブジェクト ファイルとリンクして、実行ファイルを作成します。このアセンブラには、主に以下の特長があります。

- デバイスの全命令セットのサポート
- 固定 / 浮動小数点データのサポート
- コマンドライン インターフェイス
- 豊富なディレクティブ セット
- 柔軟なマクロ言語
- MPLAB X IDE との互換性

# PIC32MZ グラフィック (DA) ファミリ

## 43.6 MPLAB X SIM ソフトウェア シミュレータ

MPLAB X SIM ソフトウェア シミュレータには、PIC MCU と dsPIC DSC を命令レベルでシミュレートする機能があり、PC 環境でコードを開発できます。任意の命令に対してデータ領域を検証または変更でき、総合的なスティミュラス コントローラから外部信号を印加できます。レジスタをファイルに記録して、より詳細な実行時解析が可能で、また、トレースバッファとロジックアナライザ ディスプレイを使って、プログラムの実行、I/O アクティビティ、ほとんどの周辺モジュールと内部レジスタの記録と確認ができるため、シミュレータとしての完成度がより向上しています。

MPLAB X SIM ソフトウェア シミュレータは、MPLAB XC コンパイラ、MPASM/MPLAB アセンブラを使ったシンボリック デバッグを完全サポートしています。このソフトウェア シミュレータは、ハードウェアラボ環境外での柔軟なコード開発とデバッグを可能にする経済的で優れたソフトウェア開発ツールです。

## 43.7 MPLAB REAL ICE インサーキット エミュレータ システム

MPLAB REAL ICE インサーキット エミュレータ システムは、Microchip 社製フラッシュ DSC/MCU デバイス用の次世代型高速エミュレータです。このエミュレータでは、MPLAB X IDE の強力な使いやすいグラフィカル ユーザー インターフェイスを利用して、8/16/32 ビット MCU および DSC の全てをデバッグ/プログラムできます。

このエミュレータをハイスピード USB 2.0 インターフェイスで設計エンジニアの PC に接続し、ターゲット デバイスとはインサーキット デバッグシステムと互換の RJ-11 コネクタか、高速で耐ノイズ性に優れた最新の LVDS インターフェイス (CAT5) で接続します。

このエミュレータの更新用ファームウェアは、MPLAB X IDE からダウンロードできます。MPLAB REAL ICE は、競合エミュレータと比べると大きく優れています。例えばフルスピードでのエミュレーション、実行時変数監視、パターン解析、複雑なブレイクポイント、論理プローブ、耐久性の高いプローブ インターフェイス、長いケーブル (最長 3 m) を使える点が挙げられます。

## 43.8 MPLAB ICD 3 インサーキット デバッグシステム

MPLAB ICD 3 インサーキット デバッグシステムは、Microchip 社製フラッシュ DSC および MCU に対応した、非常に対費用効果の高い高速ハードウェア デバッグ/プログラマです。このデバッグを使うと、MPLAB IDE の使いやすいパワフルなグラフィカル ユーザー インターフェイスで PIC MCU と dsPIC DSC のデバッグとプログラミングを実行できます。

MPLAB ICD 3 インサーキット デバッグのプローブは、設計エンジニアの PC との接続にハイスピード USB 2.0 インターフェイスを使い、ターゲット デバイスとの接続には MPLAB ICD 2/MPLAB REAL ICE システムと互換のコネクタ (RJ-11) を使います。MPLAB ICD 3 は全ての MPLAB ICD 2 ヘッダをサポートしています。

## 43.9 PICKit 3 インサーキット デバッグ/プログラマ

MPLAB PICKit 3 により、MPLAB IDE の優れたユーザー インターフェイスを使って非常に安価に PIC および dsPIC フラッシュ マイクロコントローラのデバッグとプログラミングが行えます。MPLAB PICKit 3 と設計エンジニアの PC の接続にはフルスピード USB インターフェイスを使います。ターゲット デバイスとの接続には、MPLAB ICD 3/MPLAB REAL ICE と互換の Microchip 社のデバッグコネクタ (RJ-11) を使います。このコネクタは 2 本のデバイス I/O ピンとリセットラインを使って、インサーキット デバッグとインサーキット シリアル プログラミング™ (ICSP™) を実現します。

## 43.10 MPLAB PM3 デバイス プログラマ

MPLAB PM3 デバイス プログラマは CE 準拠のユニバーサル デバイス プログラマであり、VDDIOMIN と VDDIOMAX でのプログラマブルな電圧検証によって最大限の信頼性を確保します。このデバイス プログラマは、メニューとエラーメッセージを表示する大型 LCD (128 x 64) と、各種パッケージタイプに対応するための脱着可能なモジュール式ソケット アセンブリを備えています。ICSP ケーブル アセンブリは標準で付属しています。スタンドアロン モードでは、MPLAB PM3 デバイス プログラマを PC に接続せずに、PIC MCU の読み出し、検証、プログラムが可能です。このモードでコード保護も設定できます。MPLAB PM3 とホスト PC との接続には、RS-232 または USB ケーブルを使います。さらに、大容量メモリデバイスの高速プログラミングを可能にする高速通信と最適化されたアルゴリズムを備え、ファイル保存とデータ アプリケーションのための MMC カードを内蔵しています。

# PIC32MZ グラフィック (DA) ファミリ

---

## 43.11 デモボード、開発ボード、評価用キット、スタータキット

各種 PIC MCU と dsPIC DSC には多彩なデモボード、開発ボード、評価用ボードを使用でき、完全に機能するシステム上でアプリケーションを迅速に開発できます。ほとんどのボードは、カスタム回路を追加するためのプロトタイプ領域を備えています。また、付属のアプリケーション ファームウェアとソースコードを使って動作を評価できます。これらを編集して使う事もできます。

これらのボードは LED、温度センサ、スイッチ、スピーカ、RS-232 インターフェイス、LCD、ポテンショメータ、増設 EEPROM メモリ等の各種機能をサポートします。

デモボードと開発ボードは、カスタム回路の試作と各種マイクロコントローラ アプリケーションの学習教材として使う事ができます。

PICDEM™ と dsPICDEM™ デモ/開発ボードシリーズの回路の他に、Microchip 社ではアナログフィルタ設計、KEELOQ® セキュリティ IC、CAN、IrDA®、PowerSmart バッテリ管理、SEEVAl® 評価システム、 $\Delta\Sigma$  型 ADC、流量検出等、各種アプリケーションに対応する評価キットとデモソフトウェアを取り揃えています。

また、特定のデバイスを体験するために必要なもの全てを収めたスタータキットも提供しています。スタータキットは通常、1 つのアプリケーションとデバッグ機能の全てを 1 つのボードに搭載した形で提供します。

デモボード、開発ボード、評価キットの一覧は、Microchip 社のウェブサイト ([www.microchip.com](http://www.microchip.com)) でご覧ください。

## 43.12 サードパーティ製の開発ツール

Microchip 社は、サードパーティ製のツールも数多く提供しています。これらのツール、他では得られない便利な機能を提供します。

- デバイス プログラマ/ギャング プログラマ: SoftLog 社、CCS 社等の製品
- ソフトウェア ツール: Gimpel 社、Trace Systems 社等の製品
- プロトコル アナライザ: Saleae 社、Total Phase 社等の製品
- デモボード: MikroElektronika社、Digilent®社、Olimex 社等の製品
- 組み込み Ethernet ソリューション: EZ Web Lynx 社、WIZnet 社、IPLogika® 社等の製品

# PIC32MZ グラフィック (DA) ファミリ

## 44.0 電気的特性

以下では、PIC32MZ DAの電気的特性の概要を説明します。今後新たに追加される情報は、本書の改訂版に記載します。

以下は PIC32MZ DA の絶対最大定格の一覧です。長期間にわたる最大定格条件での動作や保管は、デバイスの信頼性に影響する可能性があります。本書の動作表に示す条件または上記から外れた条件でのデバイスの運用は想定していません。

### 絶対最大定格

(Note1 参照)

通電中の周囲温度 .....	-40 ~ +85 °C
保管温度 .....	-65 ~ +150 °C
VDDIO/VDDCORE/VBAT 電圧 (VSS 基準) .....	-0.3 ~ +4.0 V
VDDR1V8 ピン電圧 (VSS1V8 基準) .....	-0.5 ~ +1.98 V
DDR2 ピン電圧 (VSS1V8 基準) .....	-0.3 V ~ (VDDR1V8 + 0.3 V)
5 V 対応ピン電圧 (VSS 基準)(Note 3) .....	-0.3 ~ (VDDIO + 0.3 V)
5 V 対応ピン電圧 (VSS 基準、VDDIO ≥ 2.2 V 時)(Note 3) .....	-0.3 ~ +5.5 V
5 V 対応ピン電圧 (VSS 基準、VDDIO < 2.2 V 時)(Note 3) .....	-0.3 ~ +3.6 V
D+/D- ピン電圧 (VUSB3V3 基準) .....	-0.3 V ~ (VUSB3V3 + 0.3 V)
VBUS 電圧 (VSS 基準) .....	-0.3 ~ +5.5 V
VSS ピン最大電流 .....	200 mA
VDDIO ピン最大電流 (Note 2) .....	200 mA
DDR2 ピンによる最大シンク / ソース電流 .....	22 mA
任意の 4 本の I/O ピンによる最大シンク / ソース電流 (Note 4) .....	15 mA
任意の 8 本の I/O ピンによる最大シンク / ソース電流 (Note 4) .....	25 mA
任意の 12 本の I/O ピンによる最大シンク / ソース電流 (Note 4) .....	33 mA
全ポートによる最大シンク電流 (Note 5) .....	150 mA
全ポートによる最大ソース電流 (Note 2、Note 5) .....	150 mA

- Note 1:** 「絶対最大定格」を超える条件は、デバイスに恒久的な損傷を生じる可能性があります。これはストレス定格です。本書の動作表に示す条件外でのデバイス運用は想定していません。絶対最大定格条件を超えて長期間曝露させると、デバイスの信頼性に影響する可能性があります。
- 2:** 最大許容電流は、デバイスの最大消費電力によって決まります (表 44-2 参照)。
- 3:** 5 V 対応ピンについては表 5 ~ 表 7 を参照してください。
- 4:** 特性データであり、製造時の検査は実施していません。4 本、8 本、12 本の I/O ピンの特性については DO10、DO20、DO20a を参照してください。
- 5:** DDR2 ピンを除きます。

# PIC32MZ グラフィック (DA) ファミリ

## 44.1 DC 特性

表 44-1: 電圧に対する動作 MIPS

特性	V <sub>DDIO</sub> レンジ (V) (Note 1)	V <sub>DDCORE</sub> レンジ (V) (Note 1)	温度レンジ (°C)	最大周波数	コメント
				PIC32MZ DA デバイス	
DC5	2.2 ~ 3.6 V	1.7 ~ 1.9 V	-40 ~ +85 °C	200 MHz	—

**Note 1:** 動作電圧を下回る場合、デバイスの全般的な動作はリセットが発生するまで保証されますが、特性は保証できません。動作電圧を下回る場合、有効にされた全てのアナログ モジュールは機能しますが、性能は低下します。リセット値については表 44-5 を参照してください。

表 44-2: 動作温度条件

定格	記号	Min.	Typ.	Max.	単位
<b>産業用温度レンジ品</b>					
動作時接合部温度レンジ	TJ	-40	—	+125	°C
動作時周囲温度レンジ	TA	-40	—	+85	°C
消費電力: デバイス内部の消費電力: $P_{INT} = V_{DDIO} \times (I_{DD} - S I_{OH})$	PD	PINT + PI/O			W
I/O ピンの消費電力: $P_{I/O} = S ((V_{DDIO} - V_{OH}) \times I_{OH}) + S (V_{OL} \times I_{OL})$					
最大許容消費電力	PD <sub>MAX</sub>	$(T_J - T_A)/q_{JA}$			W

表 44-3: パッケージ熱特性

特性	記号	Typ.	Max.	単位	Note
パッケージ熱抵抗、169 ピン LFBGA (11x11x1.4 mm)	θ <sub>JA</sub>	25	—	°C /W	1
パッケージ熱抵抗、169 ピン LFBGA (11x11x1.56 mm)	θ <sub>JA</sub>	24	—	°C /W	1、2
パッケージ熱抵抗、176 ピン LFBGA (20x20x1.45 mm)	θ <sub>JA</sub>	17	—	°C /W	1
パッケージ熱抵抗、176 ピン LQFP (20x20x1.45 mm)	θ <sub>JA</sub>	19	—	°C /W	1、2
パッケージ熱抵抗、288 ピン LFBGA (15x15x1.4 mm)	θ <sub>JA</sub>	22	—	°C /W	1

**Note 1:** 接合部～大気の熱抵抗 (θ<sub>JA</sub>) は、パッケージのシミュレーションから求めた値です。

**2:** 内部 DDR2 SDRAM を備えたデバイス

# PIC32MZ グラフィック (DA) ファミリ

表 44-4: DC 温度 / 電圧仕様

DC 特性			標準動作条件: V <sub>DDIO</sub> = 2.2 ~ 3.6 V, V <sub>VDDCORE</sub> = 1.7 ~ 1.9 V (特に明記しない場合) 動作温度 -40 °C ≤ T <sub>A</sub> ≤ +85 °C (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
<b>動作電圧</b>							
DC10	V <sub>DDIO</sub>	I/O 電源電圧 (Note 1)	2.2	—	3.6	V	—
DC11	V <sub>VDDCORE</sub>	コア電源電圧 (Note 1)	1.7	1.8	1.9	V	—
DC12	S <sub>VDDIO</sub> /S <sub>VDDCORE</sub>	V <sub>DDIO</sub> /V <sub>VDDCORE</sub> 立ち上がり速度 (内部パワーオンリセット信号を保証する立ち上がり速度) (Note 2)	0.000011	—	1.1	V/μs	300 ms ~ 3 μs @ 3.3 V
DC13	V <sub>BAT</sub>	バッテリー電源電圧	2.2	—	3.6	V	—
DC14	V <sub>VDDR1V8</sub>	DDR メモリ電源電圧	1.7	1.8	1.9	V	—
DC15	V <sub>DDRVREF</sub>	DDR 参照電圧	0.49 x V <sub>VDDR1V8</sub>	0.50 x V <sub>VDDR1V8</sub>	0.51 x V <sub>VDDR1V8</sub>	V	—

- Note 1:** 動作電圧を下回る場合、デバイスの全般的な動作はリセットが発生するまで保証されますが、特性は保証できません。動作電圧を下回る場合、有効にされた全てのアナログモジュールは機能しますが、性能は低下します。リセット値については表 44-5 を参照してください。
- 2:** V<sub>DDIO</sub> は、パワーアップ中常に V<sub>VDDCORE</sub> 以上である必要があります。

表 44-5: 電気的特性: リセット

DC 特性 (Note1)			標準動作条件: V <sub>DDIO</sub> = 2.2 ~ 3.6 V, V <sub>VDDCORE</sub> = 1.7 ~ 1.9 V (特に明記しない場合) 動作温度 -40 °C ≤ T <sub>A</sub> ≤ +85 °C (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
RST10	V <sub>PORIO</sub>	V <sub>DDIO</sub> POR 電圧 (Note 2)	V <sub>SS</sub> + 0.3	—	1.75	V	—
RST11	V <sub>PORCORE</sub> /V <sub>BATSW</sub>	V <sub>VDDCORE</sub> POR 電圧 (Note 2) V <sub>VDDCORE</sub> から V <sub>BAT</sub> への切り換え電圧 (Note 3)	V <sub>SS</sub> + 0.3	—	1.7	V	—
RST12	V <sub>BORIO</sub>	V <sub>DDIO</sub> が HIGH から LOW へ遷移した時の BOR イベント (Note 4)	1.92	—	2.2	V	—
RST13	V <sub>PORBAT</sub>	V <sub>BAT</sub> での POR イベント (Note 4)	1.35	—	2.2	V	—
RST14	V <sub>HVD1V8</sub>	V <sub>VDDR1V8</sub> ピンでの高電圧検出	2.16	—	2.24	V	—

- Note 1:** パラメータは設計上の目安であり、製品によるテストは実施していません。
- 2:** これは、パワーオンリセットを保証するためのリミット値です。V<sub>DDIO</sub>/V<sub>VDDCORE</sub> はこの電圧まで低下する必要があります。
- 3:** V<sub>VDDCORE</sub> でパワーオンリセットが発生すると、デバイスは V<sub>BAT</sub> モードに移行します。
- 4:** 動作電圧を下回る場合、デバイスの全般的な動作はリセットが発生するまで保証されますが、特性は保証できません。動作電圧を下回る場合、有効にされた全てのアナログモジュールは機能しますが、性能は低下します。

# PIC32MZ グラフィック (DA) ファミリ

表 44-6: 低電圧検出の特性

DC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)						
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件	
LV10	VHLVD	V <sub>DDIO</sub> 遷移時の HLVD 電圧	HLVDL<3:0> = 0100 <sup>(1)</sup>	—	3.52	—	V	—
			HLVDL<3:0> = 0101	—	3.29	—	V	—
			HLVDL<3:0> = 0110	—	3.00	—	V	—
			HLVDL<3:0> = 0111	—	2.79	—	V	—
			HLVDL<3:0> = 1000	—	2.70	—	V	—
			HLVDL<3:0> = 1001	—	2.50	—	V	—
			HLVDL<3:0> = 1010	—	2.40	—	V	—
			HLVDL<3:0> = 1011	—	2.30	—	V	—
			HLVDL<3:0> = 1100	—	2.20	—	V	—
			HLVDL<3:0> = 1101	—	2.12	—	V	—
			HLVDL<3:0> = 1110	—	2.00	—	V	—
LV11	VTHL	HLVDIN ピン遷移時の電圧	HLVDL<3:0> = 1111	—	1.20	—	V	—

**Note 1:** LVD<3:0> 値 = 0000 ~ 0011 に対応するトリップポイントは実装していません。



# PIC32MZ グラフィック (DA) ファミリ

表 44-7: DC 特性 : 動作電流 (IDD = IDDIO + IDDCORE)

DC 特性 <sup>(1,2)</sup>		標準動作条件 : VDDIO = 2.2 ~ 3.6 V, VDDCORE = 1.7 ~ 1.9 V (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ)		
パラメータ No.	Typ. <sup>(3)</sup>	Max.	単位	条件
<b>I/O 動作電流 (IDDIO): 周辺モジュール有効時 (PMDx = 0, ON(PBxDIV&lt;15&gt;) = 1)</b>				
DC20	1.4	2.1	mA	8 MHz
DC21	3.5	4.1	mA	100 MHz <sup>(4)</sup>
DC22	5.6	6.5	mA	200 MHz
DC23	5.6	6.5	mA	200 MHz (L1 キャッシュおよびプリフェッチ モジュール無効) <sup>(4)</sup>
<b>I/O 動作電流 (IDDCORE): 周辺モジュール有効時 (PMDx = 0, ON(PBxDIV&lt;15&gt;) = 1)</b>				
DC20a	20	34	mA	8 MHz
DC21a	97	118	mA	100 MHz <sup>(4)</sup>
DC22a	152	180	mA	200 MHz
DC23a	128	153	mA	200 MHz (L1 キャッシュおよびプリフェッチ モジュール無効) <sup>(4)</sup>
<b>I/O 動作電流 (IDDIO): 周辺モジュール無効時 (PMDx = 1, ON(PBxDIV&lt;15&gt;) = 0)</b>				
DC24	1.4	2.1	mA	8 MHz
DC25	3.5	4.1	mA	100 MHz <sup>(4)</sup>
DC26	5.6	6.5	mA	200 MHz
DC27	5.6	6.5	mA	200 MHz (L1 キャッシュおよびプリフェッチ モジュール無効) <sup>(4)</sup>
<b>I/O 動作電流 (IDDCORE): 周辺モジュール無効時 (PMDx = 1, ON(PBxDIV&lt;15&gt;) = 0)</b>				
DC24a	19	33	mA	8 MHz
DC25a	90	109	mA	100 MHz <sup>(4)</sup>
DC26a	146	177	mA	200 MHz
DC27a	121	147	mA	200 MHz (L1 キャッシュおよびプリフェッチ モジュール無効) <sup>(4)</sup>

**Note 1:** デバイスの IDD 電源電流は、主に動作電圧と周波数によって決まります。その他の要因 (周辺モジュールバスクロック (PBCLK) 周波数、有効な周辺モジュールの数、内部コード実行パターン、I/O ピンのローディングおよびスイッチング レート、オシレータ タイプ、温度等) も消費電流に影響する可能性があります。

**2:** IDD の計測条件は以下の通りです。

- VDDR1V8 = 1.8 V
- オシレータモードは EC (8 MHz 以下の場合) または EC+PLL (8 MHz を超える場合)
- OSC1 は外部の矩形波 (レールツーレール) により駆動 (OSC1 クロック入力のオーバーシュート / アンダーシュートは 100 mV 未満とする)
- OSC2/CLKO は I/O 入力ピンとして設定
- USB PLL は無効 (USBMD = 1)、VUSB3V3 は Vss に接続
- CPU、プログラム フラッシュ、SRAM データメモリは動作可能 (プログラム フラッシュメモリ待機状態は 2)
- 全ての周辺モジュールは無効 (ON ビット = 0)
- 特に明記しない場合、L1 キャッシュおよびプリフェッチ モジュールは有効
- 周辺モジュールは全て無効 (ON ビット = 0)
- WDT、DMT、クロック切り換え、ファイルセーフ クロック監視、セカンダリ オシレータは無効
- 全 I/O ピンは入力として設定 (Vss にプルダウン)
- MCLR = VDDIO
- CPU はフラッシュから while(1) 命令文を実行
- RTCC と JTAG は無効
- I/O アナログ チャージポンプは無効 (IOANCPEN ビット (CFGCON<7>) = 0)
- ADC 入力チャージポンプは無効 (AICMPEN ビット (ADCCON1<12>) = 0)
- PBCLK7 を除く全ての周辺モジュールバスクロックは無効 (ON ビット (PBxDIV<15>) = 0、x = 2 ~ 6)

**3:** 特に明記しない場合、「Typ.」列のデータは指定された動作周波数における 3.3 V/+25 °C での値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。

**4:** このパラメータは特性評価で検証していますが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-8: DC 特性 : アイドル電流 (I<sub>IDLE</sub>)

DC 特性		標準動作条件 : V <sub>DDIO</sub> = 2.2 ~ 3.6 V、 V <sub>DDCORE</sub> = 1.7 ~ 1.9 V (特に明記しない場合) 動作温度 -40 °C ≤ T <sub>A</sub> ≤ +85 °C (産業用温度レンジ)		
パラメータ No.	Typ. <sup>(2)</sup>	Max.	単位	条件
<b>アイドル電流 (I<sub>IDLE</sub>): コア OFF/ クロック ON 時のベース電流<sup>(1)</sup></b>				
DC30	19	35	mA	8 MHz <sup>(3)</sup>
DC31	55	70	mA	100 MHz <sup>(3)</sup>
DC32	90	123	mA	200 MHz

**Note 1:** I<sub>IDLE</sub> 電流の計測条件は以下の通りです。

- V<sub>DDR1V8</sub> = 1.8 V
  - オシレータモードは EC (8 MHz 以下の場合) または EC+PLL (8 MHz を超える場合)
  - OSC1 は外部の矩形波 (レールツーレール) により駆動 (OSC1 クロック入力のオーバーシュート / アンダーシュートは 100 mV 未満とする)
  - OSC2/CLKO は I/O 入力ピンとして設定
  - USB PLL は無効 (USBPMD = 1)、V<sub>USB3V3</sub> は V<sub>SS</sub> に接続、PBCLK<sub>x</sub> 分周比 = 1:2 (x ≠ 7)
  - CPU はアイドル中 (CPU コアは停止)
  - 全ての周辺モジュールは無効 (ON ビット = 0)、対応する PMD ビットはクリア (USBPMD を除く)
  - WDT、DMT、クロック切り換え、ファイルセーフ クロック監視、セカンダリ オシレータは無効
  - 全 I/O ピンは入力として設定 (V<sub>SS</sub> にプルダウン)
  - MCLR = V<sub>DDIO</sub>
  - RTCC と JTAG は無効
  - I/O アナログ チャージポンプは無効 (IOANCPEN ビット (CFGCON<7>) = 0)
  - ADC 入力チャージポンプは無効 (AICMPEN ビット (ADCCON1<12>) = 0)
- 2: 特に明記しない場合、「Typ.」列のデータは 3.3 V/+25 °Cでの値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。
- 3: このパラメータは特性評価で検証していますが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-9: DC 特性 : 省電力時の電流 (IPD)

DC 特性 <sup>(1,2)</sup>		標準動作条件 : $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)			
パラメータ No.	Typ. <sup>(2)</sup>	Max.	単位	条件	
<b>省電力時の電流 (IPD) (Note 1)</b>					
DC40k	9	14	mA	-40 °C	スリープ <sup>(1)</sup>
DC40l	9.5	14	mA	+25 °C	
DC40m	15	25	mA	+85 °C	
<b>モジュール差動電流</b>					
DC44a	50	350	mA	3.6 V	ウォッチドッグ タイマ電流 : DIWDT <sup>(3)</sup>
DC44b	3.5	5	mA	3.6 V	ADC 電流 : DIADC <sup>(3,4)</sup>
DC44c	50	350	μA	3.6 V	デッドマンタイマ電流 : DIDMT

**Note 1:** IPD 電流の計測条件は以下の通りです。

- オシレータモードは EC (8 MHz 以下の場合) または EC+PLL (8 MHz を超える場合)
  - OSC1 は外部の矩形波 (レールツーレール) により駆動 (OSC1 クロック入力のオーバーシュート / アンダーシュートは 100 mV 未満とする)
  - OSC2/CLKO は I/O 入力ピンとして設定
  - USB PLL は無効 (USBMD = 1)、 $V_{USB3V3}$  は  $V_{SS}$  に接続
  - CPU はスリープ中
  - L1 キャッシュおよびプリフェッチ モジュールは無効
  - 全ての周辺モジュールは無効 (ON ビット = 0)、対応する PMD ビットはセット全てのクロックは無効 (ON ビット (PBxDIV<15>) = 0 ( $x \neq 1,7$ ))
  - WDT、DMT、クロック切り換え、ファイルセーフ クロック監視、セカンダリ オシレータは無効
  - 全 I/O ピンは入力として設定 ( $V_{SS}$  にプルダウン)
  - MCLR =  $V_{DDIO}$
  - RTCC と JTAG は無効
  - 電圧レギュレータはスタンバイモード中 ( $V_{REGS} = 0$ 、 $IOANCPEN = 0$ )
- 2: 特に明記しない場合、「typ.」列のデータは 3.3 V での値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。
- 3: Δ 電流は、モジュールを有効にした時の消費電流増加量です。この電流をベース IPD 電流に加算する必要があります。
- 4: 電圧レギュレータは動作可能 ( $V_{REGS} = 1$ )

# PIC32MZ グラフィック (DA) ファミリ

表 44-10: DC 特性 : I/O ピン入力仕様

DC 特性		標準動作条件 : $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40 \text{ }^{\circ}\text{C} \leq T_A \leq +85 \text{ }^{\circ}\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性	Min.	Typ. <sup>(1)</sup>	Max.	単位	条件
DI10 DI18 DI19	V <sub>IL</sub>	入力電圧 LOW					
		PMP を備えた I/O ピン	V <sub>SS</sub>	—	0.15*V <sub>DDIO</sub>	V	—
		I/O ピン	V <sub>SS</sub>	—	0.2*V <sub>DDIO</sub>	V	—
		SDAx、SCLx	V <sub>SS</sub>	—	0.3*V <sub>DDIO</sub>	V	SMBus 無効時 (Note 4)
DI19		SDAx、SCLx	V <sub>SS</sub>	—	0.8	V	SMBus 有効時 (Note 4)
DI20 DI28a DI29a DI28b DI29b	V <sub>IH</sub>	入力電圧 HIGH					
		I/O ピン (5 V 非対応) <sup>(5)</sup>	0.65*V <sub>DDIO</sub>	—	V <sub>DDIO</sub>	V	(Note 4)
		I/O ピン (5 V 対応、PMP あり) <sup>(5)</sup>	0.65*V <sub>DDIO</sub>	—	5.5	V	(Note 4)
		I/O ピン (5 V 対応) <sup>(5)</sup>	0.65*V <sub>DDIO</sub>	—	5.5	V	—
		5 V 非対応ピンでの SDAx、SCLx <sup>(5)</sup>	0.65*V <sub>DDIO</sub>	—	V <sub>DDIO</sub>	V	SMBus 無効時 (Note 4)
		5 V 非対応ピンでの SDAx、SCLx <sup>(5)</sup>	2.1	—	V <sub>DDIO</sub>	V	SMBus 有効時、 2.2 V ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> (Note 4)
		5 V 対応ピンでの SDAx、SCLx <sup>(5)</sup>	0.65*V <sub>DDIO</sub>	—	5.5	V	SMBus 無効時 (Note 4)
DI29b		5 V 対応ピンでの SDAx、SCLx <sup>(5)</sup>	2.1	—	5.5	V	SMBus 有効時、 2.2V ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> (Note 4)
DI30	ICNPU	状態変化通知 ブルアップ電流	-400	-300	-50	mA	V <sub>DDIO</sub> = 3.3 V、 V <sub>PIN</sub> = V <sub>SS</sub>
DI31	ICNPD	状態変化通知 ブルダウン電流 <sup>(4)</sup>	50	175	400	μA	V <sub>DDIO</sub> = 3.3 V、 V <sub>PIN</sub> = V <sub>DDIO</sub>
DI50 DI51 DI55 DI56	I <sub>IL</sub>	入力リーク電流 (Note 3)					
		I/O ポート	—	—	±1	mA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DDIO</sub> 、 ハイインピーダンス状態のピン
		アナログ入力ピン	—	—	±1	mA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DDIO</sub> 、 ハイインピーダンス状態のピン
		MCLR <sup>(2)</sup>	—	—	±1	mA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DDIO</sub>
DI56		OSC1	—	—	±1	mA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DDIO</sub> 、 HS モード

**Note 1:** 特に明記しない場合、「Typ.」列のデータは 3.3 V/+25 °C での値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。

**2:** MCLR ピンのリーク電流は、印加する電圧レベルによって大きく変化します。仕様値のレベルは通常の動作条件での値を表します。入力電圧が異なると、より大きなリーク電流が計測される事があります。

**3:** 負の電流値は、ピンからのソース電流として定義しています。

**4:** このパラメータは特性評価で検証していますが、製造時の検査は実施していません。

**5:** 5 V 対応ピンについては表 5 ~ 表 7 を参照してください。

# PIC32MZ グラフィック (DA) ファミリ

表 44-11: DC 特性 : I/O ピン出力仕様

DC 特性			標準動作条件 : $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ	記号	特性	Min.	Typ.	Max.	単位	条件 <sup>(1)</sup>
DO10	VOL	出力電圧 LOW I/O ピン 4本のシンクドライバピン - RA0-RA3、RA9、RA10、RA14、RA15 RB0、RB4、RB6、RB7、RB10、RB11、 RB12、RB14 RC12-RC15 RD6、RD7、RD11、RD14 RE8、RE9 RF2、RF3、RF8、RF12 RG15 RH0、RH1、RH4-RH14 RJ0-RJ2、RJ8、RJ9、RJ11	—	—	0.4	V	$I_{OL} \leq 10 \text{ mA}$ 、 $V_{DDIO} = 3.3 \text{ V}$
		出力電圧 LOW I/O ピン: 8本のシンクドライバピン - RA4、RA5 RB2、RB3、RB5、RB8、RB9、RB13、RB14、 RB15 RC1-RC4 RD0-RD3、RD9、RD10、RD12、RD13 RE0-RE7 RF0、RF1、RF4、RF5、RF13 RG0、RG1、RG6、RG7、RG8、RG9 RH2、RH3、RH7、RH15 RJ3-RJ7、RJ10、RJ12-RJ15、 RK0-RK7	—	—	0.4	V	$I_{OL} \leq 15 \text{ mA}$ 、 $V_{DDIO} = 3.3 \text{ V}$
		出力電圧 LOW I/O ピン: 12本のシンクドライバピン - RA6、RA7 RD4、RD5 RG12-RG14	—	—	0.4	V	$I_{OL} \leq 20 \text{ mA}$ 、 $V_{DDIO} = 3.3 \text{ V}$

Note 1: これらのパラメータは特性データであり、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-11: DC 特性 : I/O ピン出力仕様 ( 続き )

DC 特性		標準動作条件 : $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ ( 特に明記しない場合 ) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ ( 産業用温度レンジ )					
パラメータ	記号	特性	Min.	Typ.	Max.	単位	条件 <sup>(1)</sup>
DO20	VOH	出力電圧 HIGH I/O ピン 4 本のシンクドライバピン - RA0-RA3, RA9, RA10, RA14, RA15 RB0, RB4, RB6, RB7, RB10, RB11, RB12, RB14 RC12-RC15 RD6, RD7, RD11, RD14 RE8, RE9 RF2, RF3, RF8, RF12 RG15 RH0, RH1, RH4-RH14 RJ0-RJ2, RJ8, RJ9, RJ11	2.4	—	—	V	$I_{OH} \geq -10 \text{ mA}$ 、 $V_{DDIO} = 3.3 \text{ V}$
		出力電圧 HIGH I/O ピン: 8 本のシンクドライバピン - RA4, RA5 RB2, RB3, RB5, RB8, RB9, RB13, RB14, RB15 RC1-RC4 RD0-RD3, RD9, RD10, RD12, RD13 RE0-RE7 RF0, RF1, RF4, RF5, RF13 RG0, RG1, RG6, RG7, RG8, RG9 RH2, RH3, RH7, RH15 RJ3-RJ7, RJ10, RJ12-RJ15, RK0-RK7	2.4	—	—	V	$I_{OH} \geq -15 \text{ mA}$ 、 $V_{DDIO} = 3.3 \text{ V}$
		出力電圧 HIGH I/O ピン: 12 本のソースドライバピン - RA6, RA7 RD4, RD5 RG12-RG14	2.4	—	—	V	$I_{OH} \geq -20 \text{ mA}$ 、 $V_{DDIO} = 3.3 \text{ V}$

**Note 1:** これらのパラメータは特性データであり、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-11: DC 特性 : I/O ピン出力仕様 (続き)

DC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ	記号	特性	Min.	Typ.	Max.	単位	条件 <sup>(1)</sup>
DO20a	VOH1	<b>出力電圧 HIGH</b> I/O ピン 4本のシンクドライバピン - RA0-RA3, RA9, RA10, RA14, RA15 RB0, RB4, RB6, RB7, RB10, RB11, RB12, RB14 RC12-RC15 RD6, RD7, RD11, RD14 RE8, RE9 RF2, RF3, RF8, RF12 RG15 RH0, RH1, RH4-RH14 RJ0-RJ2, RJ8, RJ9, RJ11	1.5	—	—	V	$I_{OH} \geq -14 \text{ mA}$ , $V_{DDIO} = 3.3 \text{ V}$
			2.0	—	—	V	$I_{OH} \geq -12 \text{ mA}$ , $V_{DDIO} = 3.3 \text{ V}$
			3.0	—	—	V	$I_{OH} \geq -7 \text{ mA}$ , $V_{DDIO} = 3.3 \text{ V}$
		<b>出力電圧 HIGH</b> I/O ピン: 8本のシンクドライバピン - RA4, RA5 RB2, RB3, RB5, RB8, RB9, RB10, RB13, RB14, RB15 RC1-RC4 RD0-RD3, RD9, RD10, RD12, RD13 RE0-RE7 RF0, RF1, RF4, RF5, RF13 RG0, RG1, RG6, RG7, RG8, RG9 RH2, RH3, RH7, RH15 RJ3-RJ7, RJ10, RJ12-RJ15, RK0-RK7	1.5	—	—	V	$I_{OH} \geq -22 \text{ mA}$ , $V_{DDIO} = 3.3 \text{ V}$
			2.0	—	—	V	$I_{OH} \geq -18 \text{ mA}$ , $V_{DDIO} = 3.3 \text{ V}$
			3.0	—	—	V	$I_{OH} \geq -10 \text{ mA}$ , $V_{DDIO} = 3.3 \text{ V}$
		<b>出力電圧 HIGH</b> I/O ピン: 12本のソースドライバピン - RA6, RA7 RD4, RD5 RG12-RG14	1.5	—	—	V	$I_{OH} \geq -32 \text{ mA}$ , $V_{DDIO} = 3.3 \text{ V}$
			2.0	—	—	V	$I_{OH} \geq -25 \text{ mA}$ , $V_{DDIO} = 3.3 \text{ V}$
			3.0	—	—	V	$I_{OH} \geq -14 \text{ mA}$ , $V_{DDIO} = 3.3 \text{ V}$

**Note 1:** これらのパラメータは特性データであり、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-12: DC 特性 : I/O ピン入力注入電流仕様

DC 特性		標準動作条件 : $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性	Min.	Typ. <sup>(1)</sup>	Max.	単位	条件
DI60a	IICL	入力注入電流 LOW	0	—	-5 <sup>(2,5)</sup>	mA	このパラメータは、RB10 を除く全てのピンに適用されます。この例外に対する最大 IICL 電流は 0 mA です。
DI60b	IICH	入力注入電流 HIGH	0	—	+5 <sup>(3,4,5)</sup>	mA	このパラメータは 5 V 対応ピン、SOSCI ピン、RB10 ピンを除く全てのピンに適用されます。これらの例外に対する最大 IICH 電流は 0 mA です。
DI60c	$\Sigma I_{ICT}$	総入力注入電流 (全ての I/O ピンと制御ピンの合計)	-20 <sup>(6)</sup>	—	+20 <sup>(6)</sup>	mA	全 I/O ピンの正負の入力注入電流の純粋な瞬時値の合計 ( $ I_{ICL}  +  I_{ICH}  \leq \Sigma I_{ICT}$ )

**Note 1:** 特に明記しない場合、「Typ.」列のデータは 3.3 V/+25 °Cでの値です。これらのパラメータは設計上の目安であり、テストはしていません。

**2:**  $V_{IL \text{ source}} < (V_{SS} - 0.3)$ 、特性データであり、製造時の検査は実施していません。

**3:** 非 5 V 対応ピンのみ、 $V_{IH \text{ source}} > (V_{DDIO} + 0.3 \text{ V})$  です。

**4:** 5 V 対応デジタルピンは、 $V_{DDIO}$  に対する内部ハイサイド ダイオードを備えていません。従って、「正極性」の入力注入電流を一切許容しません。

**5:** 注入電流が 0 ではない (すなわち  $V_{IH \text{ Source}} > (V_{DDIO} + 0.3 \text{ V})$  または  $V_{IL \text{ source}} < (V_{SS} - 0.3 \text{ V})$  の場合)、ADC 結果に 4 ~ 6 カウントの影響が及びます。

**6:** 全てのピンからの入力注入電流の「純粋な瞬時値」の計算上の合計が仕様の制限値を超えない限り、 $I_{ICL}$  または  $I_{ICH}$  条件下の場合も含めて、I/O ピンの任意の数および / または組み合わせが許容されます。

**Note 2** の場合、 $I_{ICL} = ((V_{SS} - 0.3 \text{ V}) - V_{IL \text{ source}}) / R_s$  です。**Note 3** の場合、 $I_{ICH} = ((I_{ICH \text{ source}} - (V_{DDIO} + 0.3 \text{ V})) / R_s)$  です。 $R_s$  は、入力電圧源とデバイスピンの間の抵抗です。 $(V_{SS} - 0.3 \text{ V}) \leq V_{SOURCE} \leq (V_{DDIO} + 0.3 \text{ V})$  の場合、注入電流は 0 です。



# PIC32MZ グラフィック (DA) ファミリ

表 44-13: DDR2 SDRAM コントローラ I/O 仕様

DC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
DDR1	VOH	出力電圧 HIGH	$V_{DDR1V8} - 0.28$	—	—	V	—
DDR2	VOL	出力電圧 LOW	—	—	0.28	V	—
DDR5	VIH	入力電圧 HIGH	$DDR_{VREF} + 0.125$	—	$V_{DDR1V8} + 0.3$	—	—
DDR6	VIL	入力電圧 LOW	0.3	—	$DDR_{VREF} - 0.125$	—	—

Note 1: これらのパラメータは特性データであり、製造時の検査は実施していません。

表 44-14: SD ホスト コントローラ I/O 仕様

DC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
SD10	VOH	出力電圧 HIGH	2.4	—	—	V	$I_{OH} \geq 20 mA$ 、 $V_{DDIO} = 3.3 V$
SD11	VOL	出力電圧 LOW	—	—	0.4	V	$I_{OL} \leq 20 mA$ 、 $V_{DDIO} = 3.3 V$
SD12	VIH	入力電圧 HIGH	$0.65 \cdot V_{DDIO}$	—	$V_{DDIO}$	V	—
SD13	VIL	入力電圧 LOW	$V_{SS}$	—	$0.2 \cdot V_{DDIO}$	V	—

# PIC32MZ グラフィック (DA) ファミリ

表 44-15: DC 特性 : プログラムメモリ<sup>3)</sup>

DC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ. <sup>(1)</sup>	Max.	単位	条件
D130a	EP	セル書き込み耐性	10,000	—	—	E/W	ECC なし
D130b			20,000	—	—	E/W	ECC あり
D131	VPR	読み出し用 $V_{DDCORE}$	$V_{DDCOREMIN}$	—	$V_{DDCOREMAX}$	V	—
D132	VPEW	消去 / 書き込み用 $V_{DDCORE}$	$V_{DDCOREMIN}$	—	$V_{DDCOREMAX}$	V	—
D134a	TRETD	データ保持期間	10	—	—	年	ECC なし
D134b			20	—	—	年	ECC あり
D135	IDDP	プログラミング中の消費電流	—	—	30	mA	—
D136	TRW	行書き込みサイクル時間 (Note 2, 4)	—	66813	—	FRC サイクル	—
D137	TQWW	クワードワード書き込みサイクル時間 (Note 4)	—	773	—	FRC サイクル	—
D138	TWW	ワード書き込みサイクル時間 (Note 2, 4)	—	383	—	FRC サイクル	—
D139	TCE	デバイス消去サイクル時間 (Note 2, 4)	—	515373	—	FRC サイクル	—
D140	TPFE	全プログラムフラッシュ (上位および下位領域) 消去サイクル時間 (Note 4)	—	256909	—	FRC サイクル	—
D141	TPBE	プログラムフラッシュ (上位または下位領域のみ) 消去サイクル時間 (Note 4)	—	128453	—	FRC サイクル	—
D142	TPGE	ページ消去サイクル時間 (Note 2, 4)	—	128453	—	FRC サイクル	—

Note 1: 特に明記しない場合、「Typ.」列のデータは  $3.3 \text{ V}/+25 \text{ }^\circ\text{C}$  での値です。

2: 行プログラミングの最小 PBCLK5 は 4 MHz です。

3: プログラミングおよび消去中の動作条件については『PIC32 フラッシュ プログラミング仕様』(DS60001145) を参照してください。

4: このパラメータは、FRCの精度(表44-27参照)と調整値(OSCTUNレジスタ(レジスタ8-2参照))の影響を受けます。

表 44-16: DC 特性 : プログラムフラッシュメモリ待機ステート

DC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)		
要求されるフラッシュ待機ステート <sup>(1)</sup>	SYSCLK	単位	条件	
<b>ECC あり:</b>				
0待機ステート	$0 < \text{SYSCLK} \leq 60$	MHz	—	
1待機ステート	$60 < \text{SYSCLK} \leq 120$			
2待機ステート	$120 < \text{SYSCLK} \leq 200$			
<b>ECC あり:</b>				
0待機ステート	$0 < \text{SYSCLK} \leq 74$	MHz	—	
1待機ステート	$74 < \text{SYSCLK} \leq 140$			
2待機ステート	$140 < \text{SYSCLK} \leq 200$			

Note 1: 待機ステートを使う場合、プリフェッチ モジュールを有効 (PREFEN<1:0>  $\neq$  00) にし、PFMWS<2:0> ビットに必要な待機ステート値を書き込む必要があります。

# PIC32MZ グラフィック (DA) ファミリ

表 44-17: DC 特性 : DDR2 SDRAM メモリ

DC 特性			標準動作条件 : $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No. (Note 1)	記号	特性	Min.	Typ.	Max.	単位	条件
DDRM12	IDD0	動作電流、1バンクアクティブプリチャージ	—	—	90	mA	Note 2
DDRM13	IDD1	動作電流、1バンクアクティブ読み出しプリチャージ	—	—	100	mA	Note 2
DDRM14	IDD2	プリチャージパワーダウン電流	—	—	8	mA	Note 3
DDRM15	IDD3	プリチャージスタンバイ電流	—	—	45	mA	Note 2
DDRM16	IDD4	プリチャージ静止スタンバイ電流	—	—	35	mA	Note 4
DDRM17	IDD5	アクティブパワーダウン電流	—	—	12	mA	Note 3
DDRM18	IDD6	アクティブスタンバイ電流	—	—	65	mA	Note 2
DDRM19	IDD7	バースト読み出し動作電流	—	—	140	mA	Note 2
DDRM20	IDD8	バースト書き込み動作電流	—	—	165	mA	Note 2
DDRM21	IDD9	バーストリフレッシュ電流	—	—	95	mA	Note 2
DDRM22	IDD10	自己リフレッシュ電流	—	—	6	mA	Note 5
DDRM23	IDD11	バンクインターリーブ読み出し動作電流	—	—	200	mA	Note 6

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。仕様値はメモリの初期化後にのみ有効です。

- 2: DDRCKE は HIGH。DDRCS0 は有効なコマンドとコマンドの間で HIGH。アドレスバス、制御バス、データバス入力はスイッチング。
- 3: DDRCKE は LOW。他の制御およびアドレス入力は安定。データバス入力はフローティング。
- 4: DDRCKE は HIGH。DDRCS0 は HIGH。他の制御およびアドレス入力は安定。データバス入力はフローティング。
- 5: DDRCKE は LOW。DDRCK/DDRCK は LOW。他の制御およびアドレス入力はフローティング。データバス入力はフローティング。
- 6: DDRCKE は HIGH。DDRCS0 は有効なコマンドとコマンドの間で HIGH。アドレスバス入力は安定。データバス入力はスイッチング。

# PIC32MZ グラフィック (DA) ファミリ

表 44-18: コンパレータ仕様

DC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	コメント
D300	VIOFF	入力オフセット電圧	—	$\pm 10$	—	mV	$AV_{DD} = V_{DDIO}$ 、 $AV_{SS} = V_{SS}$
D301	VICM	入力コモンモード電圧	0	—	2.5	V	$AV_{DD} = V_{DDIO}$ 、 $AV_{SS} = V_{SS}$ (Note 2)
D302	CMRR	コモンモード除去比	55	—	—	dB	Max Vicm = $(V_{DDIO} - 1)\text{V}$ (Note 2)
D303	TRESP	応答時間	—	150	—	ns	$AV_{DD} = V_{DDIO}$ 、 $AV_{SS} = V_{SS}$ (Note 1,2)
D304	ON2OV	コンパレータ有効化から出力確定までの時間	—	—	10	ms	コンパレータ モジュールはコンパレータ ON ビットがセットされる前にコンフィグレーション済み (Note 2)
D305	IVREF	内部参照電圧	—	1.2	—	V	—
D306	VHYST	入力ヒステリシス電圧	48	120	192	mV	—

- Note 1:** 応答時間は、コンパレータの片方の入力を  $(V_{DDIO} - 1.5)/2$  とし、もう片方の入力を  $V_{SS}$  から  $V_{DDIO}$  へ遷移させて計測しています。
- 2:** これらのパラメータは特性データであり、製造時の検査は実施していません。
- 3:** コンパレータ モジュールは  $V_{BORIOMIN} < V_{DDIO} < V_{DDIOMIN}$  の条件で動作しますが、性能は低下します。特に明記しない場合、モジュールの動作は保証されますが、特性は保証できません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-19: コンパレータ参照電圧仕様

DC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	コメント
D312	TSET	内部 4 ビット DAC コンパレータ参照電圧セトリングタイム	—	—	10	$\mu\text{s}$	Note 1 参照
D313	DACREFH	CVREF 入力参照電圧レンジ	AVSS	—	AVDD	V	CVRSSRC (CVRSS = 0 の場合)
			VREF-	—	VREF+	V	CVRSSRC (CVRSS = 1 の場合)
D314	DVREF	CVREF プログラマブル出力電圧レンジ	0	—	$0.625 \times \text{DACREFH}$	V	0 ~ $0.625 \text{ DACREFH}$ (DACREFH/24 刻み)
			$0.25 \times \text{DACREFH}$	—	$0.719 \times \text{DACREFH}$	V	$0.25 \text{ DACREFH} \sim 0.719 \text{ DACREFH}$ (DACREFH/32 刻み)
D315	DACRES	分解能	—	—	DACREFH/24		CVRCON<CVRR> = 1
			—	—	DACREFH/32		CVRCON<CVRR> = 0
D316	DACACC	絶対精度 <sup>(2)</sup>	—	—	1, 4	LSB	DACREFH/24、CVRCON<CVRR> = 1
			—	—	1, 2	LSB	DACREFH/32、CVRCON<CVRR> = 0

**Note 1:** セトリングタイムは、CVRR = 1 として CVR<3:0> を「0000」から「1111」へ変更する事により計測しています。このパラメータは特性評価で検証していますが、製造時の検査は実施していません。

**2:** これらのパラメータは特性データであり、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-20: CTMU 電流源仕様

DC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
<b>CTMU電流源</b>							
CTMUI1	IOUT1	ベースレンジ <sup>(1)</sup>	—	0.55	—	$\mu\text{A}$	CTMUICON<9:8> = 01
CTMUI2	IOUT2	10x レンジ <sup>(1)</sup>	—	5.5	—	$\mu\text{A}$	CTMUICON<9:8> = 10
CTMUI3	IOUT3	100x レンジ <sup>(1)</sup>	—	55	—	$\mu\text{A}$	CTMUICON<9:8> = 11
CTMUI4	IOUT4	1000x レンジ <sup>(1)</sup>	—	550	—	$\mu\text{A}$	CTMUICON<9:8> = 00
CTMUFV1	VF	温度ダイオード順方向電圧 <sup>(1,2)</sup>	—	0.598	—	V	$T_A = +25 \text{ }^\circ\text{C}$ 、 CTMUICON<9:8> = 01
			—	0.658	—	V	$T_A = +25 \text{ }^\circ\text{C}$ 、 CTMUICON<9:8> = 10
			—	0.721	—	V	$T_A = +25 \text{ }^\circ\text{C}$ 、 CTMUICON<9:8> = 11
CTMUFV2	VFVR	温度ダイオードの電圧変化率 <sup>(1,2)</sup>	—	-1.92	—	$\text{mV}/^\circ\text{C}$	CTMUICON<9:8> = 01
			—	-1.74	—	$\text{mV}/^\circ\text{C}$	CTMUICON<9:8> = 10
			—	-1.56	—	$\text{mV}/^\circ\text{C}$	CTMUICON<9:8> = 11

- Note 1:** 電流調整レンジの中央 (CTMUICON<15:10> = 000000) における公称電圧です。
- 2:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。計測条件は以下の通りです。
- $V_{REF+} = AV_{DD} = 3.3 \text{ V}$
  - ADC モジュールの変換速度は 500 ksps に設定
  - 全ての PMD ビットはクリア (PMDx = 0)
  - `while(1)` 命令文を実行
  - デバイスは FRC (PLL 併用せず) を使って動作

表 44-21: GLCD コントローラ DC 仕様

DC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
GD10	VOH	出力電圧 HIGH	2.4	—	—	V	$I_{OH} \geq 20 \text{ mA}$ 、 $V_{DDIO} = 3.3 \text{ V}$
GD11	VOL	出力電圧 LOW	—	—	0.4	V	$I_{OL} \leq 20 \text{ mA}$ 、 $V_{DDIO} = 3.3 \text{ V}$
GD12	VIH	入力電圧 HIGH	$0.65 \cdot V_{DDIO}$	—	$V_{DDIO}$	V	—
GD13	VIL	入力電圧 LOW	$V_{SS}$	—	$0.2 \cdot V_{DDIO}$	V	—

# PIC32MZ グラフィック (DA) ファミリ

## 44.2 AC 特性とタイミングパラメータ

以下に、PIC32MZ DA の AC 特性とタイミングパラメータの定義を示します。

図 44-1: デバイス タイミング仕様に対する負荷条件

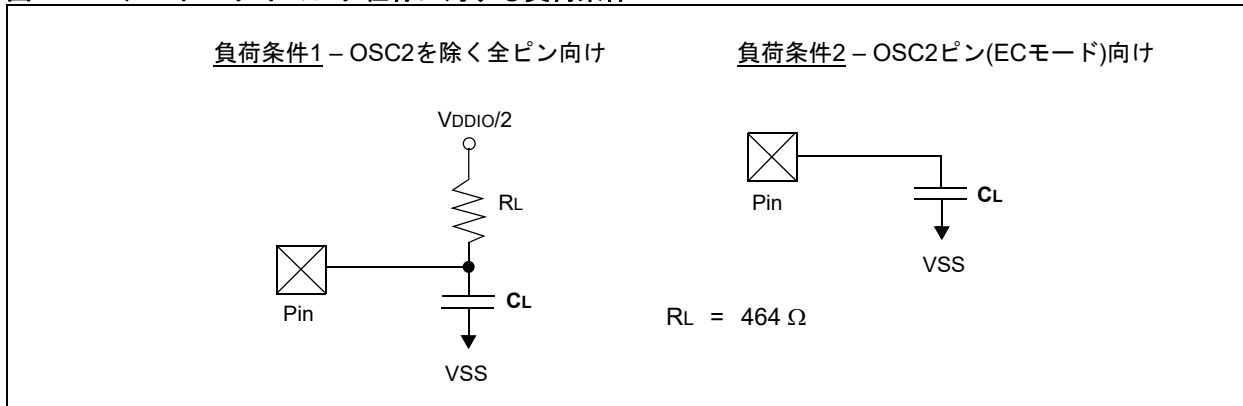


表 44-22: 出力ピンの容量性負荷要件

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性	Min.	Typ. <sup>(1)</sup>	Max.	単位	条件
DO50	CosCO	OSC2 ピン	—	—	15	pF	HS モード、外部クロックで OSC1 を駆動時
DO56	CL	全 I/O ピン	—	—	50	pF	OSC2 に対して EC モード
DO58	CB	SCLx、SDAx	—	—	400	pF	I <sup>2</sup> C モード
DO59	CsQI	全 SQI ピン	—	—	10	pF	—

**Note 1:** 特に明記しない場合、「Typ.」列のデータは  $3.3 \text{ V}/+25 \text{ }^\circ\text{C}$  での値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

図 44-2: 外部クロック タイミング

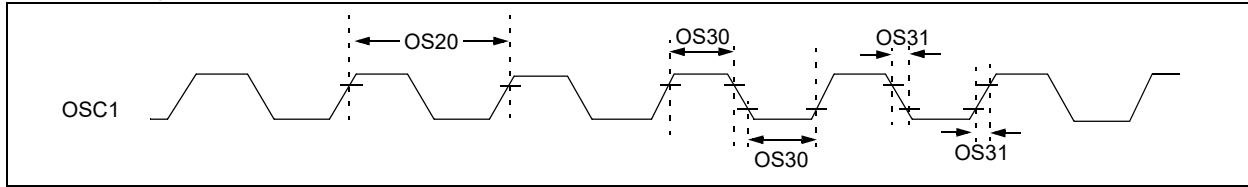


表 44-23: 外部クロック タイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^{\circ}\text{C} \leq T_A \leq +85 \text{ }^{\circ}\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ. <sup>(1)</sup>	Max.	単位	条件
OS10	Fosc	外部 CLKI 周波数 (外部クロックは EC および ECPLL モードでのみ許容)	DC	—	64	MHz	EC (Note 2)
OS13		オシレータ水晶振動子周波数	4	—	32	MHz	HS (Note 2)
OS15			32	32.768	100	kHz	Sosc (Note 2)
OS20	Tosc	$Tosc = 1/Fosc$	—	—	—	—	Fosc の値についてはパラメータ OS10 参照
OS30	TosL、 TosH	外部クロック入力 (OSC1) HIGH または LOW 時間	$0.375 \times Tosc$	—	—	ns	EC (Note 2)
OS31	TosR、 TosF	外部クロック入力 (OSC1) 立ち上がりまたは立ち下がり 時間	—	—	7.5	ns	EC (Note 2)
OS40	TOST	オシレータ起動タイム期間 (HS、HSPLL、Sosc クロック オシレータモードにのみ適用)	—	1024	—	Tosc	(Note 2)
OS41	TfSCM	プライマリクロックフェイル セーフタイムアウト期間	—	2	—	ms	(Note 2)
OS42	GM	外部オシレータ相互コンダクタンス	—	400	—	$\mu\text{A/V}$	$V_{DDIO} = 3.3\text{V}$ 、 $T_A = +25 \text{ }^{\circ}\text{C}$ (Note 2)

Note 1: 特に明記しない場合、「Typ.」列のデータは  $3.3 \text{ V}/+25 \text{ }^{\circ}\text{C}$  での値です。これらのパラメータは特性データであり、製造時の検査は実施していません。

2: このパラメータは特性評価で検証していますが、製造時の検査は実施していません。



# PIC32MZ グラフィック (DA) ファミリ

表 44-24: システム タイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
OS51	FSYS	システム周波数	DC	—	200	MHz	USB モジュールは無効
			30	—	200	MHz	USB モジュールは有効
OS55a	FPB	周辺モジュールバス周波数	DC	—	100	MHz	PBCLKx (x < 7) に対して
OS55b			DC	—	200	MHz	PBCLK7 に対して
OS56	FREF	参照クロック周波数	—	—	50	MHz	REFCLK1、REFCLK3、REFCLK4、REFCLK01、REFCLK3、REFCLK4 ピン に対して

表 44-25: SPLL クロック タイミング仕様

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	記号	特性 (1)	Min.	Typ.	Max.	単位	条件
OS50	FIN	PLL 入力周波数レンジ	5	—	64	MHz	ECPLL、HSPLL、FRCPLL モード
OS52	TLOCK	PLL 起動時間 (ロック時間)	—	—	100	$\mu s$	—
OS53	DCLK	CLKO 安定性 (2) (周期ジッタまたは累積ジッタ)	-0.25	—	+0.25	%	100 ms の期間で計測
OS54	FVco	PLL Vco 周波数レンジ	350	—	700	MHz	—
OS54a	FPLL	PLL 出力周波数レンジ	10	—	200	MHz	—

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

**2:** このジッタ仕様値は、クロックサイクルごとの計測に基づきます。通信クロックでの個々のタイムベースに対する実効ジッタは、下式により求められます。

$$EffectiveJitter = \frac{D_{CLK}}{\sqrt{\frac{PBCLK2}{CommunicationClock}}}$$

例: PBCLK2 = 100 MHz/SPIビットレート = 50 MHzの場合:

$$EffectiveJitter = \frac{D_{CLK}}{\sqrt{\frac{100}{50}}} = \frac{D_{CLK}}{1.41}$$

# PIC32MZ グラフィック (DA) ファミリ

表 44-26: MPLL クロック タイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性 <sup>(1)</sup>	Min.	Typ.	Max.	単位	条件
MP10	MFIN	MPLL 入力周波数	8	—	64	MHz	—
MP11	MFVCO	MPLL $V_{CO}$ 周波数レンジ	400	—	1600	MHz	—
MP12	MFMPLL	MPLL 出力周波数	8	—	400	MHz	—
MP13	MLOCK	MPLL 起動時間 (ロック時間)	—	—	$1500 \times 1/MFIN$	$\mu\text{s}$	—
MP14	MPJ	MPLL 周期ジッタ	—	—	0.015	%	—
MP15	MCJ	MPLL サイクル間ジッタ	—	—	0.02	%	—
MP16	MLTJ	MPLL 長期ジッタ	—	—	0.5	%	—

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-27: 内部 FRC 精度

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	特性	Min.	Typ.	Max.	単位	条件
内部 FRC 精度 (@8.00 MHz) <sup>(1)</sup>						
F20	FRC	-5	—	+5	%	$0^{\circ}C \leq T_A \leq +85^{\circ}C$
		-8	—	+8	%	$-40^{\circ}C \leq T_A \leq +85^{\circ}C$

Note 1: 周波数は  $+25^{\circ}C / 3.3 V$  の条件で校正されています。温度ドリフトは TUN ビットを使って補償できます。

表 44-28: 内部 LPRC 精度

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	特性	Min.	Typ.	Max.	単位	条件
LPRC (31.25 kHz) <sup>(1)</sup>						
F21	LPRC	-8	—	+8	%	$0^{\circ}C \leq T_A \leq +85^{\circ}C$
		-25	—	+25	%	$-40^{\circ}C \leq T_A \leq +85^{\circ}C$

Note 1: LPRC 周波数は  $V_{DDIO}$  によって変化します。

表 44-29: 内部バックアップ FRC (BFRC) 精度

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	特性	Min.	Typ.	Max.	単位	条件
内部 BFRC 精度 (@8 MHz) <sup>(1)</sup>						
F22	BFRC	-30	—	+30	%	—

# PIC32MZ グラフィック (DA) ファミリ

図 44-3: I/O タイミング特性

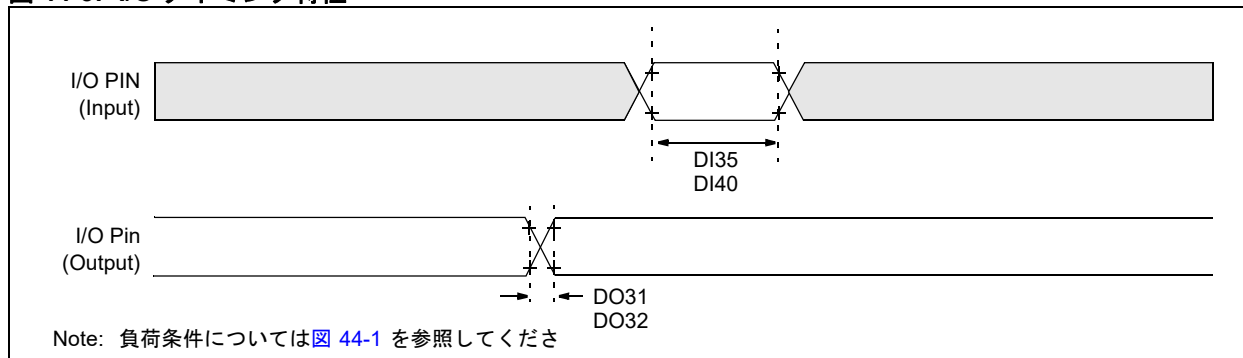


表 44-30: I/O タイミング要件

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)					
パラメータ No.	記号	特性 <sup>(2)</sup>	Min.	Typ. <sup>(1)</sup>	Max.	単位	条件
DO31	TioR	ポート出力立ち上がり時間 I/O ピン: 4本のソースドライバピン - RA3, RA9, RA10, RA14, RA15 RB0-7, RB11, RB13 RC12-RC15 RD0, RD6-RD7, RD11, RD14 RE8, RE9 RF2, RF3, RF8, RG15 RH0, RH1, RH4-RH6, RH8- RH13 RJ0-RJ2, RJ8, RJ9, RJ11	—	—	9.5	ns	CLOAD = 50 pF
		ポート出力立ち上がり時間 I/O ピン: 8本のソースドライバピン - RA0-RA2, RA4, RA5 RB8-RB10, RB12, RB14, RB15 RC1-RC4 RD1-RD5, RD9, RD10, RD12, RD13, RD15 RE4-RE7 RF0, RF4, RF5, RF12, RF13 RG0, RG1, RG6-RG9 RH2, RH3, RH7, RH14, RH15 RJ3-RJ7, RJ10, RJ12-RJ15, RK0-RK7	—	—	6	ns	CLOAD = 20 pF
		ポート出力立ち上がり時間 I/O ピン: 8本のソースドライバピン - RA0-RA2, RA4, RA5 RB8-RB10, RB12, RB14, RB15 RC1-RC4 RD1-RD5, RD9, RD10, RD12, RD13, RD15 RE4-RE7 RF0, RF4, RF5, RF12, RF13 RG0, RG1, RG6-RG9 RH2, RH3, RH7, RH14, RH15 RJ3-RJ7, RJ10, RJ12-RJ15, RK0-RK7	—	—	8	ns	CLOAD = 50 pF
		ポート出力立ち上がり時間 I/O ピン: 12本のソースドライバピン - RA6, RA7 RE0-RE3 RF1 RG12-RG14	—	—	3.5	ns	CLOAD = 50 pF
			—	—	2	ns	CLOAD = 20 pF

Note 1: 特に明記しない場合、「Typ.」列のデータは  $3.3 V/+25^{\circ}C$  での値です。

2: このパラメータは特性評価で検証していますが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-30: I/O タイミング要件 (続き)

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)					
パラメータ No.	記号	特性 <sup>(2)</sup>	Min.	Typ. <sup>(1)</sup>	Max.	単位	条件
DO32	TioF	ポート出力立ち下がり時間 I/O ピン: 4本のソースドライバピン - RA3, RA9, RA10, RA14, RA15 RB0-7, RB11, RB13 RC12-RC15 RD0, RD6-RD7, RD11, RD14 RE8, RE9 RF2, RF3, RF8, RG15 RH0, RH1, RH4-RH6, RH8- RH13 RJ0-RJ2, RJ8, RJ9, RJ11	—	—	9.5	ns	CLOAD = 50 pF
		ポート出力立ち下がり時間 I/O ピン: 8本のソースドライバピン - RA0-RA2, RA4, RA5 RB8-RB10, RB12, RB14, RB15 RC1-RC4 RD1-RD5, RD9, RD10, RD12, RD13, RD15 RE4-RE7 RF0, RF4, RF5, RF12, RF13 RG0, RG1, RG6-RG9 RH2, RH3, RH7, RH14, RH15 RJ3-RJ7, RJ10, RJ12-RJ15, RK0-RK7	—	—	6	ns	CLOAD = 20 pF
		ポート出力立ち下がり時間 I/O ピン: 8本のソースドライバピン - RA0-RA2, RA4, RA5 RB8-RB10, RB12, RB14, RB15 RC1-RC4 RD1-RD5, RD9, RD10, RD12, RD13, RD15 RE4-RE7 RF0, RF4, RF5, RF12, RF13 RG0, RG1, RG6-RG9 RH2, RH3, RH7, RH14, RH15 RJ3-RJ7, RJ10, RJ12-RJ15, RK0-RK7	—	—	8	ns	CLOAD = 50 pF
		ポート出力立ち下がり時間 I/O ピン: 12本のソースドライバピン - RA6, RA7 RE0-RE3 RF1 RG12-RG14	—	—	6	ns	CLOAD = 20 pF
		ポート出力立ち下がり時間 I/O ピン: 12本のソースドライバピン - RA6, RA7 RE0-RE3 RF1 RG12-RG14	—	—	3.5	ns	CLOAD = 50 pF
			—	—	2	ns	CLOAD = 20 pF
DI35	TINP	INTx ピンの HIGH/LOW 時間	5	—	—	ns	—
DI40	TRBP	CNx の HIGH/LOW 時間 (入力)	5	—	—	ns	—

**Note 1:** 特に明記しない場合、「Typ.」列のデータは 3.3 V/+25 °Cでの値です。

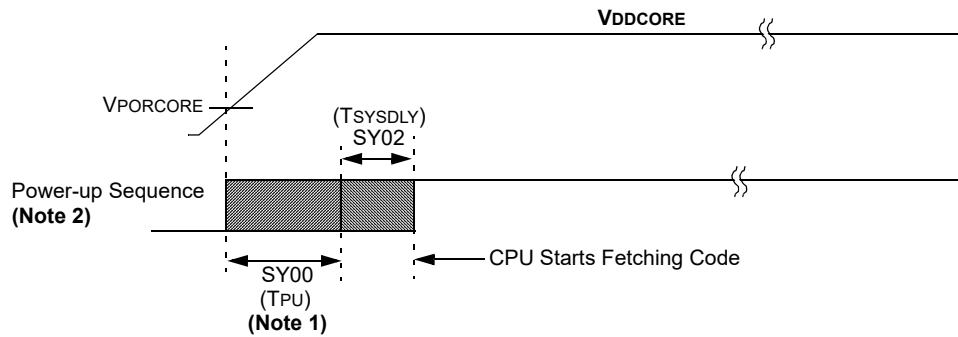
**2:** このパラメータは特性評価で検証していますが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

図 44-4: パワーオンリセット タイミング特性

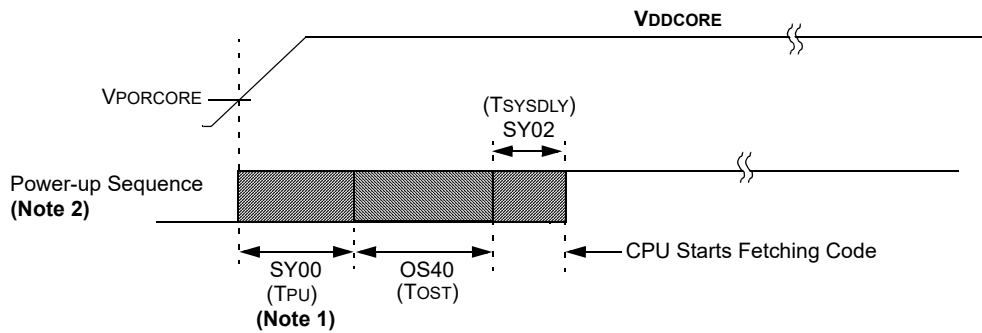
Internal Voltage Regulator Enabled

Clock Sources = (FRC, FRCDIV, FRCDIV16, FRCPLL, EC, ECPLL and LPRC)



Internal Voltage Regulator Enabled

Clock Sources = (HS, HSPLL, and Sosc)



**Note 1:** BOR ( $V_{DDIO} < V_{DDIOMIN}$ ) が終了する前に電源投入シーケンスが完了した場合、電源投入期間は延長されます。

**2:** これには電圧レギュレータの安定化遅延を含みます。

# PIC32MZ グラフィック (DA) ファミリ

図 44-5: 外部リセット タイミング特性

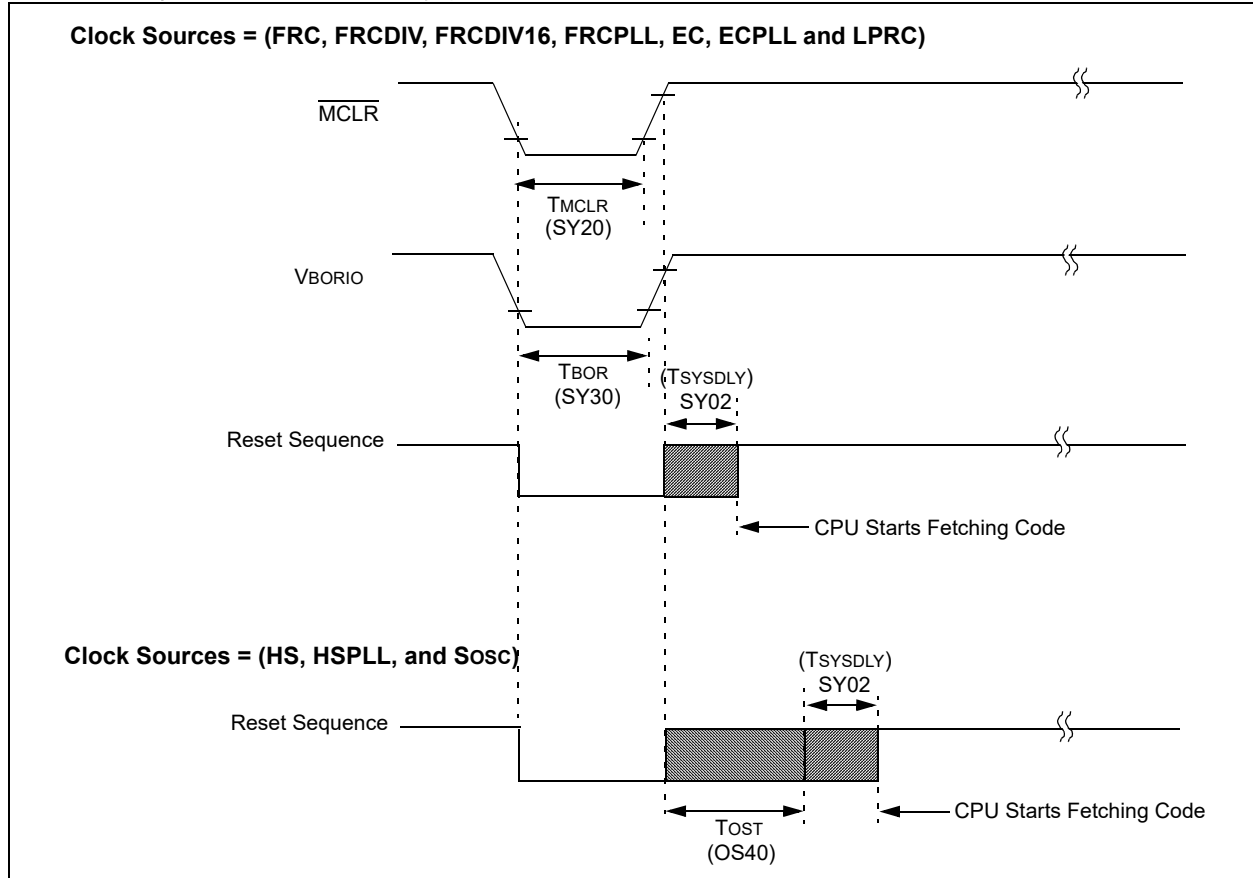


表 44-31: リセット タイミング

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	記号	特性 <sup>(1)</sup>	Min.	Typ. <sup>(2)</sup>	Max.	単位	条件
SY00	TPU	起動時間 内部電圧レギュレータは有効	—	400	600	ms	—
SY02	TSYSDLY	システム遅延期間: デバイス コンフィグレーション ヒューズのリロードに要する時間 と最初の命令をフェッチするまで の SYSCLK 遅延の合計	—	1 ms + 8 SYSCLK サイクル	—	—	—
SY20	TMCLR	MCLR パルス幅 (LOW)	2	—	—	ms	—
SY30	TBOR	BOR パルス幅 (LOW)	—	1	—	ms	—

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

**Note 2:** 特に明記しない場合、「Typ.」列のデータは  $3.3 V/+25^{\circ}C$  での値です。特性評価済みですが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

図 44-6: Timer1 ~ 9 外部クロック タイミング特性

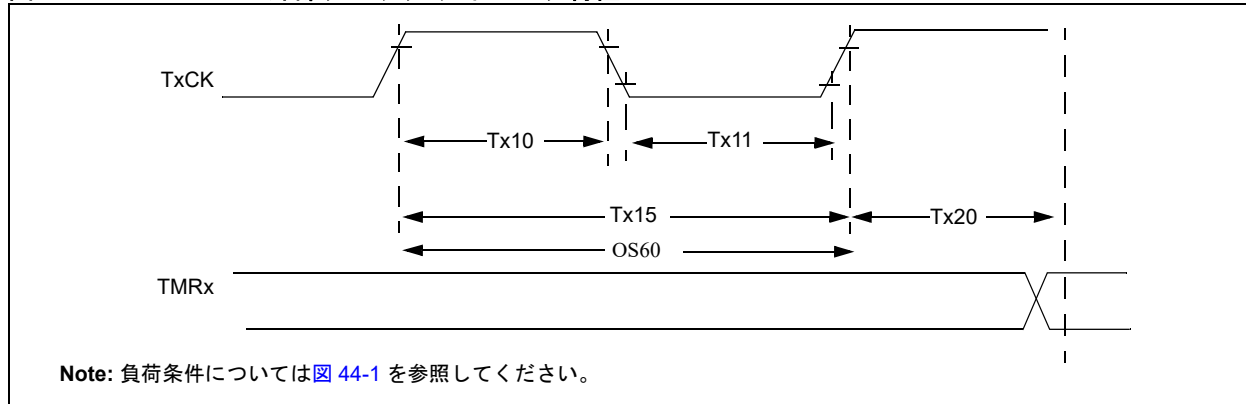


表 44-32: Timer1 外部クロック タイミング要件 (1)

AC 特性		標準動作条件: VDDIO = 2.2 ~ 3.6 V, VDDCORE = 1.7 ~ 1.9 V (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ)						
パラメータ No.	記号	特性 (2)		Min.	Typ.	Max.	単位	条件
TA10	TtxH	TxCK HIGH 時間	同期、プリスケールあり	$[(12.5 \text{ ns または } 1 \text{ TPBCLK3}) / N] + 20 \text{ ns}$	—	—	ns	パラメータ TA15 も満たす必要あり (Note 3)
			非同期、プリスケールあり	10	—	—	ns	—
TA11	TtxL	TxCK LOW 時間	同期、プリスケールあり	$[(12.5 \text{ ns または } 1 \text{ TPBCLK3}) / N] + 20 \text{ ns}$	—	—	ns	パラメータ TA15 も満たす必要あり (Note 3)
			非同期、プリスケールあり	10	—	—	ns	—
TA15	TtxP	TxCK 入力周期	同期、プリスケールあり	$[(20 \text{ ns または } 2 \text{ TPBCLK3 の大きい方}) / N] + 30 \text{ ns}$	—	—	ns	VDDIO > 2.7 V (Note 3)
				$[(20 \text{ ns または } 2 \text{ TPBCLK3 の大きい方}) / N] + 50 \text{ ns}$	—	—	ns	VDDIO < 2.7 V (Note 3)
			非同期、プリスケールあり	20	—	—	ns	VDDIO > 2.7 V
				50	—	—	ns	VDDIO < 2.7 V
OS60	Ft1	SOSC1/T1CK オシレータ入力周波数レンジ (TCS ビット (T1CON<1>) をセットする事でオシレータを有効化)		32	—	50	kHz	—
TA20	TCKEXTMRL	外部 TxCK クロックエッジからタイマ インクリメントまでの遅延		—	—	1	TPBCLK3	—

Note 1: Timer1 はタイプ A です。

2: このパラメータは特性評価で検証していますが、製造時の検査は実施していません。

3: N = プリスケール値 (1、8、64、256)



# PIC32MZ グラフィック (DA) ファミリ

表 44-33: Timer2 ~ 9 外部クロック タイミング要件

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ , $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性 (1)		Min.	Max.	単位	条件
TB10	TtXH	TxCK HIGH 時間	同期、プリスケールあり	$[(12.5 \text{ ns または } 1 \text{ TPBCLK}_3) / N] + 25 \text{ ns}$	—	ns	パラメータ TB15 も満たす必要あり N = プリスケール値 (1、2、4、8、16、32、64、256)
TB11	TtXL	TxCK LOW 時間	同期、プリスケールあり	$[(12.5 \text{ ns または } 1 \text{ TPBCLK}_3) / N] + 25 \text{ ns}$	—	ns	
TB15	TtXP	TxCK 入力期間	同期、プリスケールあり	$[(25 \text{ ns または } 2 \text{ TPBCLK}_3 \text{ の大きい方}) / N] + 30 \text{ ns}$	—	ns	
				$[(25 \text{ ns または } 2 \text{ TPBCLK}_3 \text{ の大きい方}) / N] + 50 \text{ ns}$	—	ns	$V_{DDIO} < 2.7 \text{ V}$
TB20	TcKEXTMRL	外部 TxCK クロックエッジからタイマインクリメントまでの遅延		—	1	TPBCLK <sub>3</sub>	—

Note 1: これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

図 44-7: 入力キャプチャ (CAPx) タイミング特性

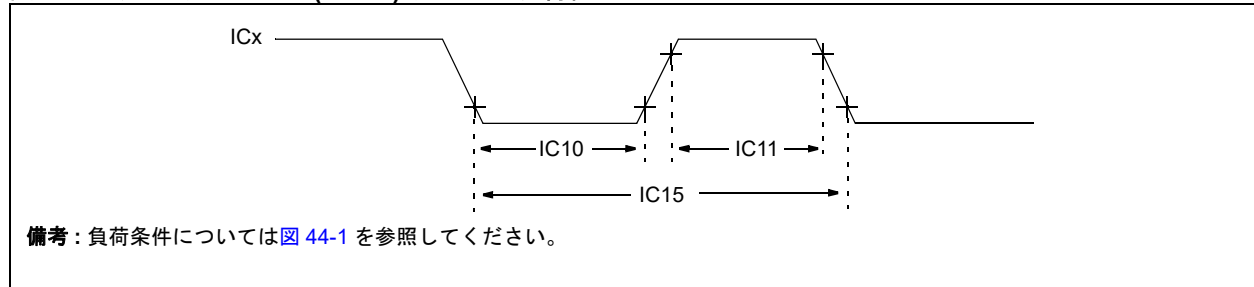


表 44-34: 入力キャプチャ モジュール タイミング要件

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ , $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性 (1)		Min.	Max.	単位	条件
IC10	TcCL	ICx 入力 LOW 時間		$[(12.5 \text{ ns または } 1 \text{ TPBCLK}_3) / N] + 25 \text{ ns}$	—	ns	パラメータ IC15 も満たす必要あり N = プリスケール値 (1、4、16)
IC11	TcCH	ICx 入力 HIGH 時間		$[(12.5 \text{ ns または } 1 \text{ TPBCLK}_3) / N] + 25 \text{ ns}$	—	ns	
IC15	TcCP	ICx 入力周期		$[(25 \text{ ns または } 2 \text{ TPBCLK}_3) / N] + 50 \text{ ns}$	—	ns	

Note 1: これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

図 44-8: 出力コンペア モジュール (OCx) タイミング特性

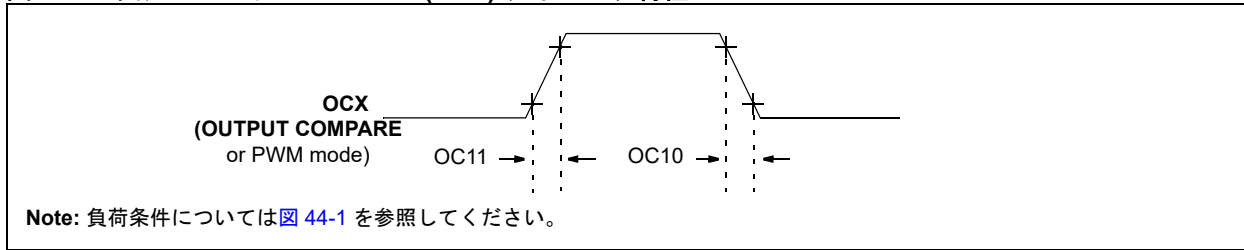


表 44-35: 出力コンペア モジュール タイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性 (1)	Min.	Typ. (2)	Max.	単位	条件
OC10	TccF	OCx 出力立ち下がり時間	—	—	—	ns	パラメータ DO32 参照
OC11	TccR	OCx 出力立ち上がり時間	—	—	—	ns	パラメータ DO31 参照

- Note 1: これらのパラメータは特性評価済みですが、製造時の検査は実施していません。  
 2: 特に明記しない場合、「Typ.」列のデータは  $3.3 \text{ V}/+25 \text{ }^\circ\text{C}$  での値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。

図 44-9: OCx/PWM モジュール タイミング特性

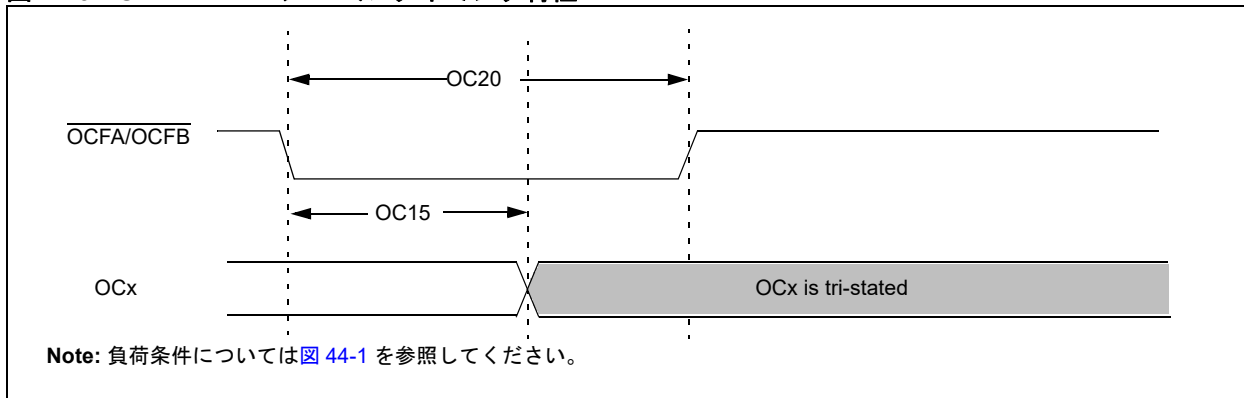


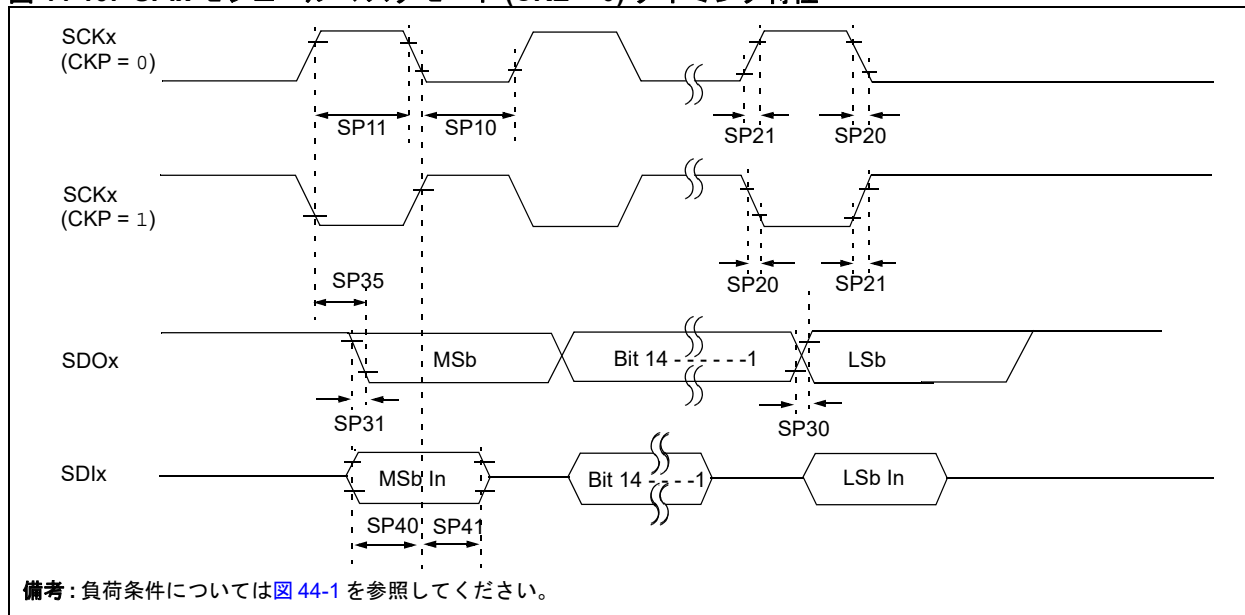
表 44-36: 単純 OCx/PWM モード タイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性 (1)	Min.	Typ. (2)	Max.	単位	条件
OC15	TfD	フォルト入力から PWM I/O が変化するまでの時間	—	—	50	ns	—
OC20	TFLT	フォルト入力パルス幅	50	—	—	ns	—

- Note 1: これらのパラメータは特性評価済みですが、製造時の検査は実施していません。  
 2: 特に明記しない場合、「Typ.」列のデータは  $3.3 \text{ V}/+25 \text{ }^\circ\text{C}$  での値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

図 44-10: SPIx モジュール マスタモード (CKE = 0) タイミング特性



# PIC32MZ グラフィック (DA) ファミリ

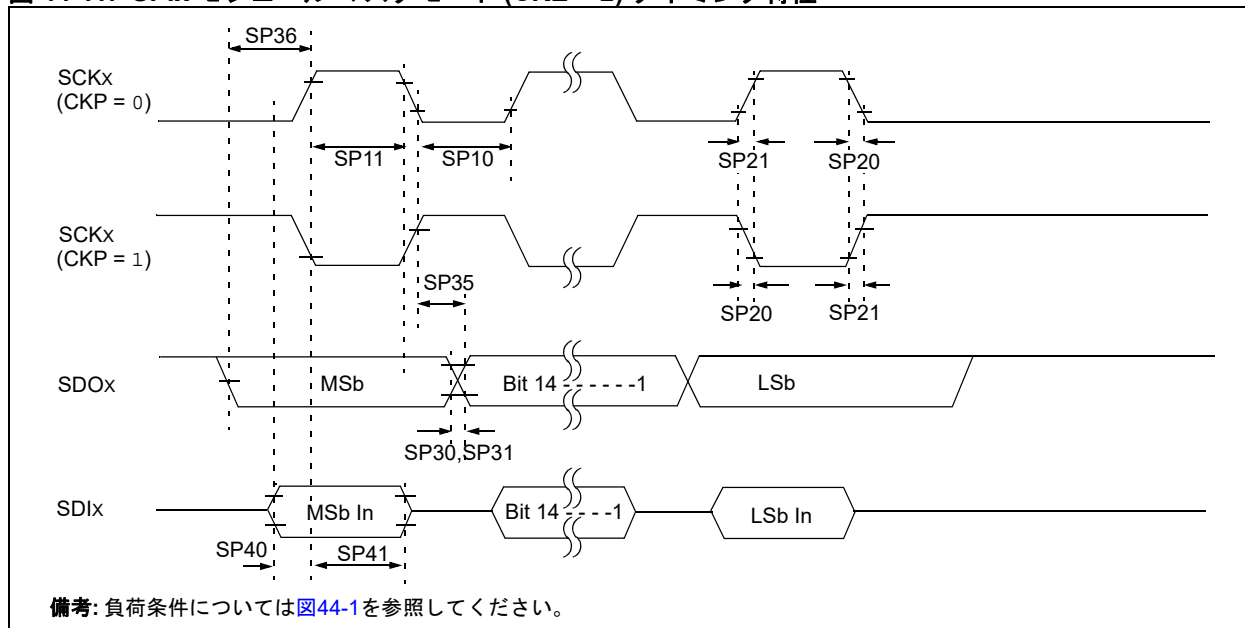
表 44-37: SPIx マスタモード (CKE = 0) タイミング要件

AC 特性			標準動作条件: V <sub>DDIO</sub> = 2.2 ~ 3.6 V、 V <sub>DDCORE</sub> = 1.7 ~ 1.9 V (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ)				
パラメータ No.	記号	特性 <sup>(1)</sup>	Min.	Typ. <sup>(2)</sup>	Max.	単位	条件
SP10	TsCL	SCKx 出力 LOW 時間 (Note 3)	T <sub>SCK</sub> /2	—	—	ns	Note 5
SP11	TsCH	SCKx 出力 HIGH 時間 (Note 3)	T <sub>SCK</sub> /2	—	—	ns	Note 5
SP15	TsCK	SPI クロック速度 (Note 5)	—	—	25	MHz	SPI1、SPI3、SPI4、SPI6
					50	MHz	SPI2 (RPG7、RPG8)
					25	MHz	SPI2 (他の I/O)
					50	MHz	SPI5 (RPC1、RPC4)
					25	MHz	SPI5 (他の I/O)
SP20	TsCF	SCKx 出力立ち下がり時間 (Note 4)	—	—	—	ns	パラメータ DO32 参照
SP21	TsCR	SCKx 出力立ち上がり時間 (Note 4)	—	—	—	ns	パラメータ DO31 参照
SP30	TDoF	SDOx データ出力立ち下がり時間 (Note 4)	—	—	—	ns	パラメータ DO32 参照
SP31	TDoR	SDOx データ出力立ち上がり時間 (Note 4)	—	—	—	ns	パラメータ DO31 参照
SP35	TsCH2DoV、 TsCL2DoV	SCKx エッジから SDOx データ出力確定までの時間	—	—	7	ns	V <sub>DDIO</sub> > 2.7 V
			—	—	10	ns	V <sub>DDIO</sub> < 2.7 V
SP40	TdIV2sCH、 TdIV2sCL	SCKx エッジに対する SDIx データ入力セットアップ時間	5	—	—	ns	—
SP41	TsCH2dIL、 TsCL2dIL	SCKx エッジに対する SDIx データ入力ホールド時間	5	—	—	ns	—

- Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。
- 2:** 特に明記しない場合、「Typ.」列のデータは 3.3 V/+25 °Cでの値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。
- 3:** SCKx の最小クロック周期は 20 ns です。マスタモードで生成するクロックはこの仕様に違反しない事が必要です。
- 4:** 全ての SPIx ピンに 30 pF の負荷を想定しています。
- 5:** 最大データレートを達成するには、V<sub>DDIO</sub> が 3.0 V 以上あり、SMP ビット (SPIxCON<9>) が「1」にセットされている必要があります。

# PIC32MZ グラフィック (DA) ファミリ

図 44-11: SPIx モジュール マスタモード (CKE = 1) タイミング特性



# PIC32MZ グラフィック (DA) ファミリ

表 44-38: SPIx モジュール マスタモード (CKE = 1) タイミング要件

AC 特性			標準動作条件: V <sub>DDIO</sub> = 2.2 ~ 3.6 V, V <sub>DDCORE</sub> = 1.7 ~ 1.9 V (特に明記しない場合) 動作温度 -40 °C ≤ T <sub>A</sub> ≤ +85 °C (産業用温度レンジ)				
パラメータ No.	記号	特性 <sup>(1)</sup>	Min.	Typ. <sup>(2)</sup>	Max.	単位	条件
SP10	TscL	SCKx 出力 LOW 時間 (Note 3)	T <sub>SCK</sub> /2	—	—	ns	Note 5
SP11	TschH	SCKx 出力 HIGH 時間 (Note 3)	T <sub>SCK</sub> /2	—	—	ns	Note 5
SP15	TscK	SPI クロック速度 (Note 5)	—	—	25	MHz	SSPI1、SPI3、SPI4、SPI6 SPI2 (RPG7、RPG8) SPI2 (他の I/O) SPI5 (RPC1、RPC4) SPI5 (他の I/O)
			—	—	50	MHz	
			—	—	25	MHz	
			—	—	50	MHz	
			—	—	25	MHz	
SP20	TscF	SCKx 出力立ち下がり時間 (Note 4)	—	—	—	ns	パラメータ DO32 参照
SP21	TscR	SCKx 出力立ち上がり時間 (Note 4)	—	—	—	ns	パラメータ DO31 参照
SP30	TdoF	SDOx データ出力立ち下がり時間 (Note 4)	—	—	—	ns	パラメータ DO32 参照
SP31	TdoR	SDOx データ出力立ち上がり時間 (Note 4)	—	—	—	ns	パラメータ DO31 参照
SP35	Tsch2doV、 TscL2doV	SCKx エッジから SDOx データ出力確定までの時間	—	—	7	ns	V <sub>DDIO</sub> > 2.7 V
			—	—	10		V <sub>DDIO</sub> < 2.7 V
SP36	TdoV2sc、 TdoV2scl	最初の SCKx エッジに対する SDOx データ出力セットアップ時間	7	—	—	ns	—
SP40	TdiV2sch、 TdiV2scl	SCKx エッジに対する SDIx データ入力セットアップ時間	7	—	—	ns	V <sub>DDIO</sub> > 2.7 V
			10				V <sub>DDIO</sub> < 2.7 V
SP41	Tsch2diL、 TscL2diL	SCKx エッジに対する SDIx データ入力ホールド時間	7	—	—	ns	V <sub>DDIO</sub> > 2.7 V
			10	—	—	ns	V <sub>DDIO</sub> < 2.7 V

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

**2:** 特に明記しない場合、「Typ.」列のデータは 3.3 V/+25 °Cでの値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。

**3:** SCKx の最小クロック周期は 20 ns です。マスタモードで生成するクロックはこの仕様に違反しない事が必要です。

**4:** 全ての SPIx ピンに 30 pF の負荷を想定しています。

**5:** 最大データレートを達成するには、V<sub>DDIO</sub> が 3.0 V 以上あり、SMP ビット (SPIxCON<9>) が「1」にセットされている必要があります。

# PIC32MZ グラフィック (DA) ファミリ

図 44-12: SPIx モジュール スレーブモード (CKE = 0) タイミング特性

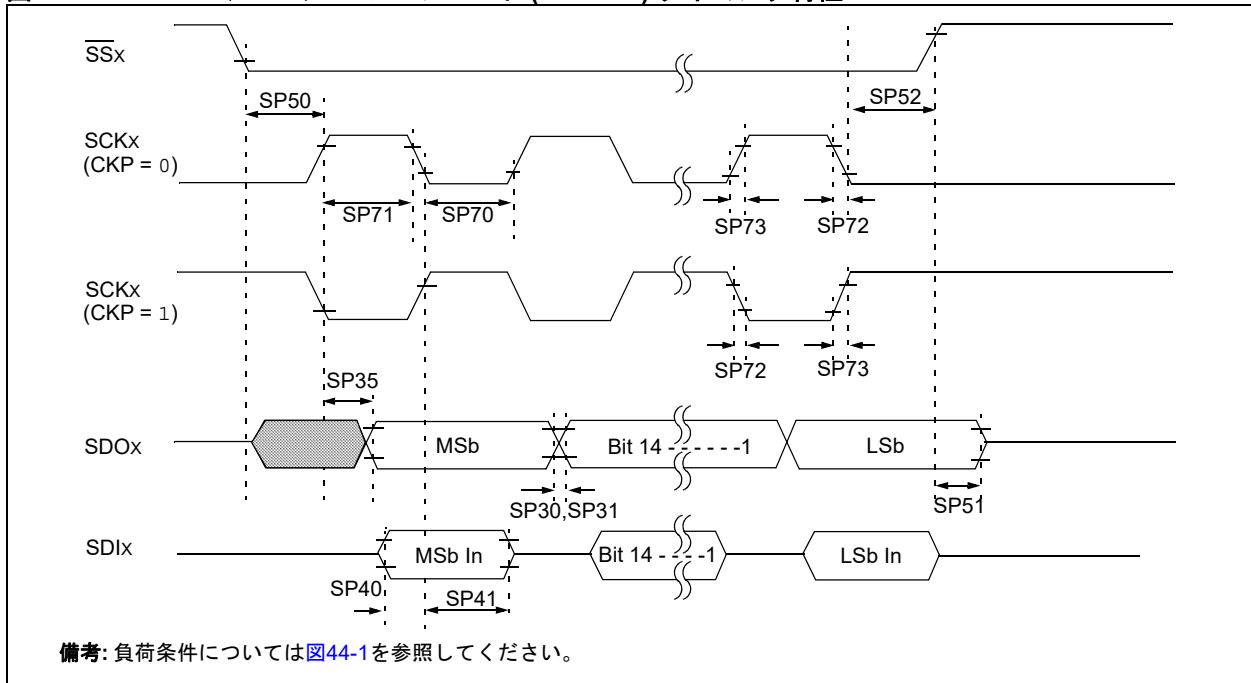


表 44-39: SPIx モジュール スレーブモード (CKE = 0) タイミング要件

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ , $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性 (1)	Min.	Typ. (2)	Max.	単位	条件
SP70	TscL	SCKx 入力 LOW 時間 (Note 3)	$T_{SCK}/2$	—	—	ns	Note 5
SP71	TscH	SCKx 入力 HIGH 時間 (Note 3)	$T_{SCK}/2$	—	—	ns	Note 5
SP72	TscF	SCKx 入力立ち下がり時間	—	—	—	ns	パラメータ DO32 参照
SP73	TscR	SCKx 入力立ち上がり時間	—	—	—	ns	パラメータ DO31 参照
SP30	TdoF	SDOx データ出力立ち下がり時間 (Note 4)	—	—	—	ns	パラメータ DO32 参照
SP31	TdoR	SDOx データ出力立ち上がり時間 (Note 4)	—	—	—	ns	パラメータ DO31 参照
SP35	Tsch2doV, TscL2doV	SCKx エッジから SDOx データ出力確定までの時間	—	—	7	ns	$V_{DDIO} > 2.7 \text{ V}$
			—	—	10	ns	$V_{DDIO} < 2.7 \text{ V}$
SP40	TdiV2scH, TdiV2scL	SCKx エッジに対する SDIx データ入力セットアップ時間	5	—	—	ns	—
SP41	Tsch2diL, TscL2diL	SCKx エッジに対する SDIx データ入力ホールド時間	5	—	—	ns	—
SP50	Tssl2scH, Tssl2scL	$\overline{SSx}$ ↓ から SCKx ↑ または SCKx 入力までの時間	88	—	—	ns	—
SP51	Tssh2doZ	$\overline{SSx}$ ↑ から SDOx 出力ハイインピーダンスまでの時間 (Note 3)	2.5	—	12	ns	—

Note 1: これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

Note 2: 特に明記しない場合、「Typ.」列のデータは  $3.3 \text{ V}/+25 \text{ }^\circ\text{C}$  での値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。

Note 3: SCKx の最小クロック周期は 20 ns です。

Note 4: 全ての SPIx ピンに 10 pF の負荷を想定しています。

Note 5:  $T_{SCK}$  は、SPI1/SPI3/SPI4/SPI6 に対して 40 ns、SPI2/SPI5 に対して 20 ns です。

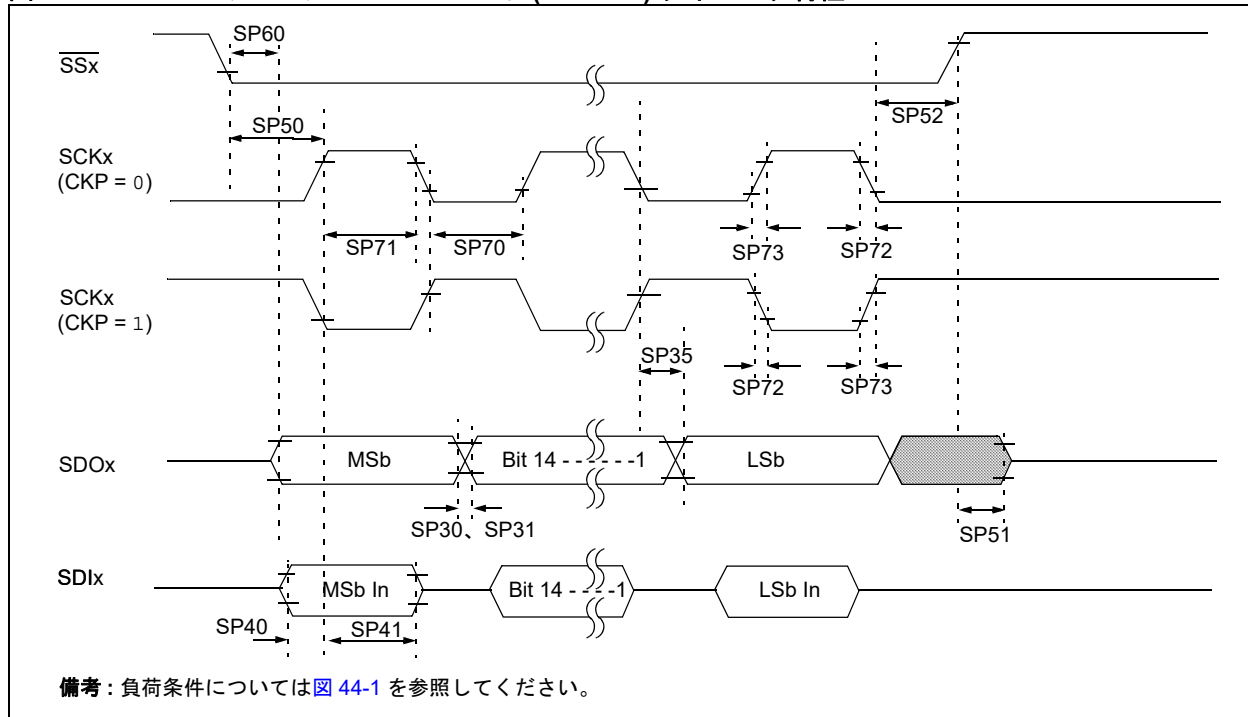
# PIC32MZ グラフィック (DA) ファミリ

表 44-39: SPIx モジュール スレープモード (CKE = 0) タイミング要件 (続き)

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ , $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性 (1)	Min.	Typ. (2)	Max.	単位	条件
SP52	Tsch2ssH TscL2ssH	SCKx エッジから SSx までの時間	10	—	—	ns	—

- Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。  
**2:** 特に明記しない場合、「Typ.」列のデータは 3.3 V/+25 °C での値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。  
**3:** SCKx の最小クロック周期は 20 ns です。  
**4:** 全ての SPIx ピンに 10 pF の負荷を想定しています。  
**5:** T<sub>sck</sub> は、SPI1/SPI3/SPI4/SPI6 に対して 40 ns、SPI2/SPI5 に対して 20 ns です。

図 44-13: SPIx モジュール スレープモード (CKE = 1) タイミング特性





# PIC32MZ グラフィック (DA) ファミリ

表 44-40: SPIx モジュール スレーブモード (CKE = 1) タイミング要件 (x = 1, 3, 4, 6)

AC 特性			標準動作条件: V <sub>DDIO</sub> = 2.2 ~ 3.6 V, V <sub>DDCORE</sub> = 1.7 ~ 1.9 V (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ)				
パラメータ No.	記号	特性 <sup>(1)</sup>	Min.	Typ. <sup>(2)</sup>	Max.	単位	条件
SP70	TscL	SCKx 入力 LOW 時間 (Note 3)	Tsck/2	—	—	ns	Note 5
SP71	TscH	SCKx 入力 HIGH 時間 (Note 3)	Tsck/2	—	—	ns	Note 5
SP72	TscF	SCKx 入力立ち下がり時間	—	—	10	ns	—
SP73	TscR	SCKx 入力立ち上がり時間	—	—	10	ns	—
SP30	TdoF	SDOx データ出力立ち下がり時間 (Note 4)	—	—	—	ns	パラメータ DO32 参照
SP31	TdoR	SDOx データ出力立ち上がり時間 (Note 4)	—	—	—	ns	パラメータ DO31 参照
SP35	Tsch2doV, TscL2doV	SCKx エッジから SDOx データ出力確定までの時間	—	—	10	ns	V <sub>DDIO</sub> > 2.7 V
			—	—	15	ns	V <sub>DDIO</sub> < 2.7 V
SP40	TdiV2sch, TdiV2scl	SCKx エッジに対する SDIx データ入力セットアップ時間	0	—	—	ns	—
SP41	Tsch2dil, TscL2dil	SCKx エッジに対する SDIx データ入力ホールド時間	7	—	—	ns	—
SP50	TssL2sch, TssL2scl	$\overline{SSx}$ ↓ から SCKx ↓ または SCKx ↑ 入力までの時間	88	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx}$ ↑ から SDOx 出力ハイインピーダンスまでの時間 (Note 4)	2.5	—	12	ns	—
SP52	Tsch2ssh, TscL2ssh	SCKx エッジから $\overline{SSx}$ ↑ までの時間	10	—	—	ns	—
SP60	TssL2doV	$\overline{SSx}$ エッジから SDOx データ出力確定までの時間	—	—	12.5	ns	—

- Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。
- 2:** 特に明記しない場合、「Typ.」列のデータは 3.3 V/+25 °C での値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。
- 3:** SCKx の最小クロック周期は 20 ns です。
- 4:** 全ての SPIx ピンに 10 pF の負荷を想定しています。
- 5:** Tsck は、SPI1/SPI3/SPI4/SPI6 に対して 40 ns、SPI2/SPI5 に対して 20 ns です。

# PIC32MZ グラフィック (DA) ファミリ

図 44-14: SQI シリアル入カタイミング特性

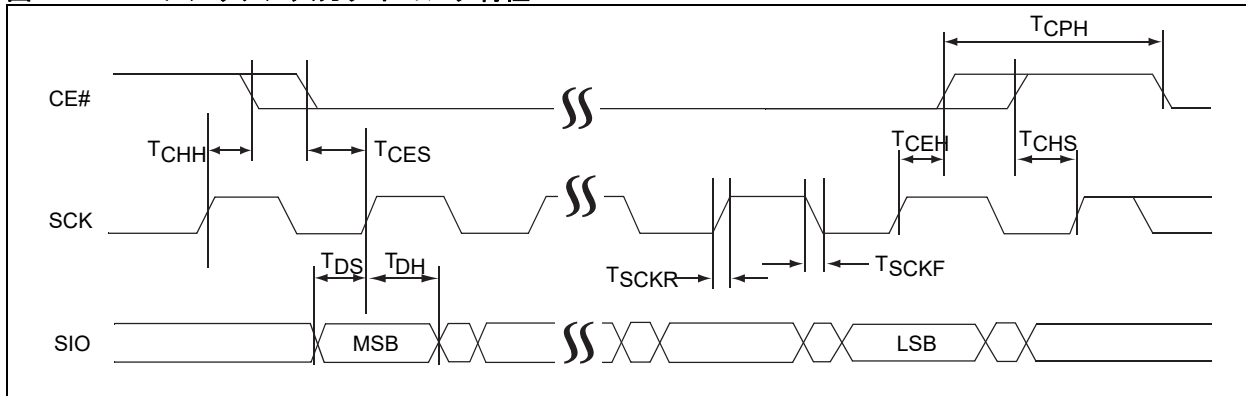


図 44-15: SQI シリアル出カタイミング特性

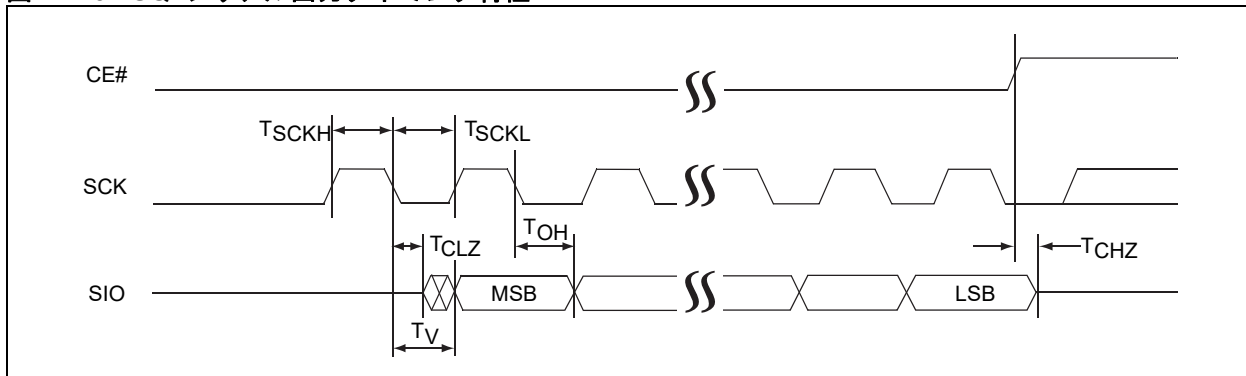


表 44-41: SQI タイミング要件

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ , $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
SQ10	FCLK	シリアルクロック周波数 (1/Tsqi)	—	—	80	MHz	DMA 読み出しモード、SDR モード
			—	—	66	MHz	DMA 読み出しモード、DDR モード
			—	—	100	MHz	PIO 書き込みモード、SDR モード
SQ11	T_SCKH	シリアルクロック HIGH 時間	6	—	—	ns	—
SQ12	T_SCKL	シリアルクロック LOW 時間	6	—	—	ns	—
SQ13	T_SCKR	シリアルクロック立ち上がり時間	0.25	—	—	ns	—
SQ14	T_SCKF	シリアルクロック立ち下がり時間	0.25	—	—	ns	—
SQ15	T_CSS (T_CES)	$\overline{\text{CS}}$ アクティブセットアップ時間	5	—	—	ns	—
SQ16	T_CSH (T_CEH)	$\overline{\text{CS}}$ アクティブ ホールド時間	5	—	—	ns	—
SQ17	T_CHS	$\overline{\text{CS}}$ 非アクティブ セットアップ時間	3	—	—	ns	—
SQ18	T_CHH	$\overline{\text{CS}}$ 非アクティブ ホールド時間	3	—	—	ns	—
SQ19	T_CPH	$\overline{\text{CS}}$ HIGH 時間	6	—	—	ns	—
SQ20	T_CHZ	$\overline{\text{CS}}$ HIGH から High-Z データ出力までの時間	—	—	6	ns	—
SQ21	T_CLZ	SCK LOW から Low-Z データ出力までの時間	0	—	—	ns	—
SQ22	T_DS	データ入力セットアップ時間	3	—	—	ns	—
SQ23	T_DH	データ入力ホールド時間	4	—	—	ns	—
SQ24	T_OH	データ出力ホールド時間	0	—	—	ns	—
SQ25	T_OV (T_V)	データ出力確定	—	—	6	ns	—

# PIC32MZ グラフィック (DA) ファミリ

図 44-16: I2Cx バス START/STOP ビットタイミング特性 (マスタモード)

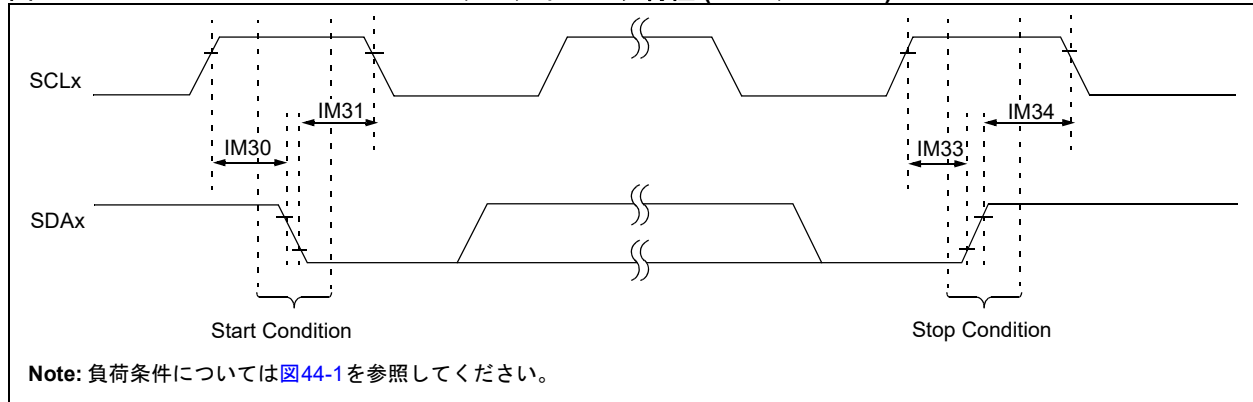


図 44-17: I2Cx バスデータ タイミング特性 (マスタモード)

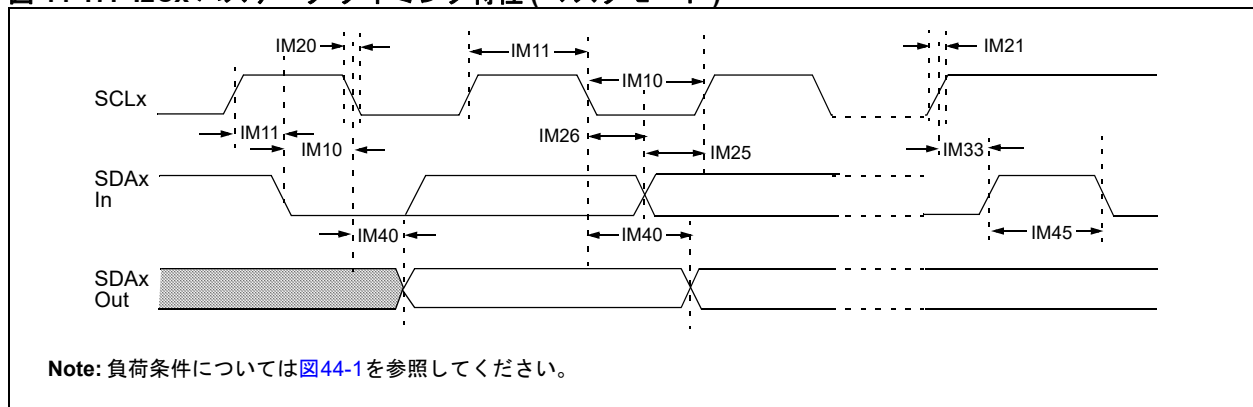


表 44-42: I2Cx バスデータ タイミング要件 (マスタモード)

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ , $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^{\circ}\text{C} \leq T_A \leq +85 \text{ }^{\circ}\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性	Min. <sup>(1)</sup>	Max.	単位	条件	
IM10	TLO:SCL	クロック LOW 時間	100 kHz モード	TPBCLK2 * (BRG + 2)	—	ms	—
			400 kHz モード	TPBCLK2 * (BRG + 2)	—	ms	—
			1 MHz モード (Note 2)	TPBCLK2 * (BRG + 2)	—	ms	—
IM11	THI:SCL	クロック HIGH 時間	100 kHz モード	TPBCLK2 * (BRG + 2)	—	ms	—
			400 kHz モード	TPBCLK2 * (BRG + 2)	—	ms	—
			1 MHz モード (Note 2)	TPBCLK2 * (BRG + 2)	—	ms	—
IM20	TF:SCL	SDAx と SCLx の立ち下がり時間	100 kHz モード	—	300	ns	Cb は 10 ~ 400 pF に指定
			400 kHz モード	$20 + 0.1 C_B$	300	ns	
			1 MHz モード (Note 2)	—	100	ns	

Note 1: BRG は I<sup>2</sup>C baud レート ジェネレータの値です。

2: 全ての I2Cx ピンの最大静電容量は 10 pF (1 MHz モードのみ) です。

3: このパラメータの代表値は 104 ns です。

# PIC32MZ グラフィック (DA) ファミリ

表 44-42: I2Cx バスデータ タイミング要件 (マスタモード) (続き)

AC 特性				標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)			
パラメータ No.	記号	特性	Min. <sup>(1)</sup>	Max.	単位	条件	
IM21	Tr:SCL	SDAx と SCLx の立ち上がり時間	100 kHz モード	—	1000	ns	Cb は 10 ~ 400 pF に指定
			400 kHz モード	$20 + 0.1 \text{ Cb}$	300	ns	
			1 MHz モード (Note 2)	—	300	ns	
IM25	TSU:DAT	データ入力セットアップ時間	100 kHz モード	250	—	ns	—
			400 kHz モード	100	—	ns	
			1 MHz モード (Note 2)	100	—	ns	
IM26	THD:DAT	データ入力ホールド時間	100 kHz モード	0	—	ms	—
			400 kHz モード	0	0.9	ms	
			1 MHz モード (Note 2)	0	0.3	ms	
IM30	TSU:STA	START 条件セットアップ時間	100 kHz モード	$TPBCLK2 * (BRG + 2)$	—	ms	反復スタート条件の場合のみ該当
			400 kHz モード	$TPBCLK2 * (BRG + 2)$	—	ms	
			1 MHz モード (Note 2)	$TPBCLK2 * (BRG + 2)$	—	ms	
IM31	THD:STA	START 条件ホールド時間	100 kHz モード	$TPBCLK2 * (BRG + 2)$	—	ms	この期間後に、最初のクロックパルスを生成
			400 kHz モード	$TPBCLK2 * (BRG + 2)$	—	ms	
			1 MHz モード (Note 2)	$TPBCLK2 * (BRG + 2)$	—	ms	
IM33	TSU:STO	STOP 条件セットアップ時間	100 kHz モード	$TPBCLK2 * (BRG + 2)$	—	ms	—
			400 kHz モード	$TPBCLK2 * (BRG + 2)$	—	ms	
			1 MHz モード (Note 2)	$TPBCLK2 * (BRG + 2)$	—	ms	
IM34	THD:STO	STOP 条件ホールド時間	100 kHz モード	$TPBCLK2 * (BRG + 2)$	—	ns	—
			400 kHz モード	$TPBCLK2 * (BRG + 2)$	—	ns	
			1 MHz モード (Note 2)	$TPBCLK2 * (BRG + 2)$	—	ns	
IM40	TAA:SCL	クロックから出力確定までの時間	100 kHz モード	—	3500	ns	—
			400 kHz モード	—	1000	ns	
			1 MHz モード (Note 2)	—	350	ns	
IM45	TBF:SDA	バスフリー時間	100 kHz モード	4.7	—	ms	次の伝送が開始可能になるまでに必要なバスフリー時間
			400 kHz モード	1.3	—	ms	
			1 MHz モード (Note 2)	0.5	—	ms	
IM50	CB	バス負荷容量	—	—	pF	パラメータ DO58 参照	
IM51	TPGD	パルスゴブラー遅延	52	312	ns	Note 3 参照	

Note 1: BRG は I<sup>2</sup>C baud レート ジェネレータの値です。

2: 全ての I2Cx ピンの最大静電容量は 10 pF (1 MHz モードのみ) です。

3: このパラメータの代表値は 104 ns です。

# PIC32MZ グラフィック (DA) ファミリ

図 44-18: I2Cx バス START/STOP ビット タイミング特性 (スレーブモード)

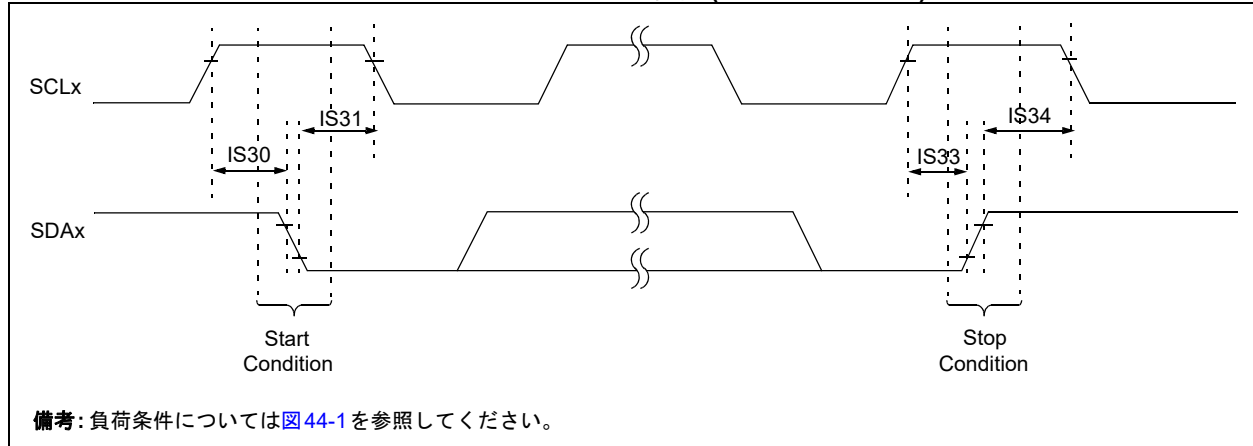


図 44-19: I2Cx バスデータ タイミング特性 (スレーブモード)

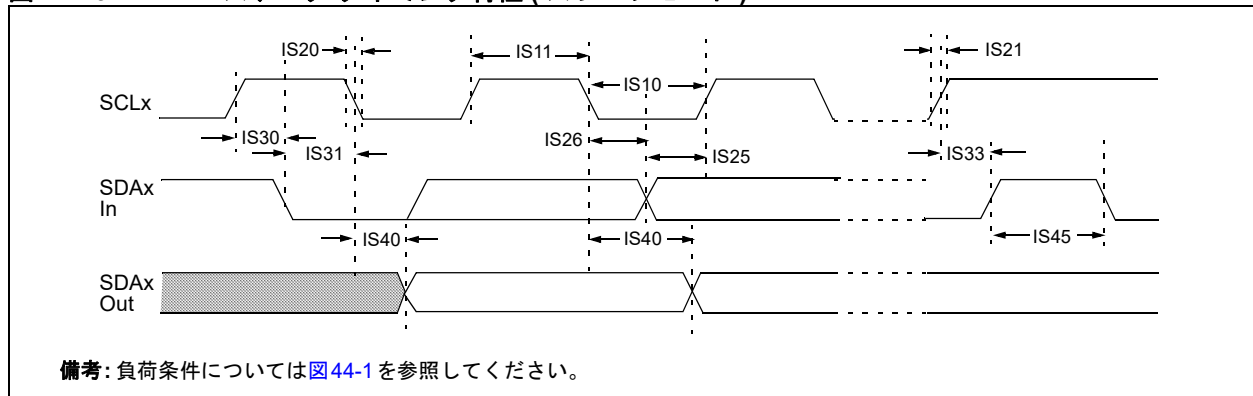


表 44-43: I2Cx バスデータ タイミング要件 (スレーブモード)

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ , $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性	Min.	Max.	単位	条件	
IS10	TLO:SCL	クロック LOW 時間	100 kHz モード	4.7	—	ms	PBCLK は 800 kHz 以上で動作する事が必要
			400 kHz モード	1.3	—	ms	PBCLK は 3.2 MHz 以上で動作する事が必要
			1 MHz モード (Note 1)	0.5	—	ms	—
IS11	THI:SCL	クロック HIGH 時間	100 kHz モード	4.0	—	ms	PBCLK は 800 kHz 以上で動作する事が必要
			400 kHz モード	0.6	—	ms	PBCLK は 3.2 MHz 以上で動作する事が必要
			1 MHz モード (Note 1)	0.5	—	ms	—

Note 1: 全ての I2Cx ピンの最大静電容量 = 10 pF (1 MHz モードのみ) です。

# PIC32MZ グラフィック (DA) ファミリ

表 44-43: I2Cx バスデータ タイミング要件 (スレーブモード) (続き)

AC 特性				標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)			
パラメータ No.	記号	特性	Min.	Max.	単位	条件	
IS20	TF:SCL	SDAx と SCLx の立ち下がり時間	100 kHz モード	—	300	ns	Cb は 10 ~ 400 pF に指定
			400 kHz モード	$20 + 0.1 \text{ Cb}$	300	ns	
			1 MHz モード (Note 1)	—	100	ns	
IS21	TR:SCL	SDAx と SCLx の立ち上がり時間	100 kHz モード	—	1000	ns	Cb は 10 ~ 400 pF に指定
			400 kHz モード	$20 + 0.1 \text{ Cb}$	300	ns	
			1 MHz モード (Note 1)	—	300	ns	
IS25	TSU:DAT	データ入力セットアップ時間	100 kHz モード	250	—	ns	—
			400 kHz モード	100	—	ns	
			1 MHz モード (Note 1)	100	—	ns	
IS26	THD:DAT	データ入力ホールド時間	100 kHz モード	0	—	ns	—
			400 kHz モード	0	0.9	ms	
			1 MHz モード (Note 1)	0	0.3	ms	
IS30	TSU:STA	START 条件セットアップ時間	100 kHz モード	4700	—	ns	反復スタート条件の場合のみ該当
			400 kHz モード	600	—	ns	
			1 MHz モード (Note 1)	250	—	ns	
IS31	THD:STA	START 条件ホールド時間	100 kHz モード	4000	—	ns	この期間後に最初のクロックパルスを生成
			400 kHz モード	600	—	ns	
			1 MHz モード (Note 1)	250	—	ns	
IS33	TSU:STO	STOP 条件セットアップ時間	100 kHz モード	4000	—	ns	—
			400 kHz モード	600	—	ns	
			1 MHz モード (Note 1)	600	—	ns	
IS34	THD:STO	STOP 条件ホールド時間	100 kHz モード	4000	—	ns	—
			400 kHz モード	600	—	ns	
			1 MHz モード (Note 1)	250	—	ns	
IS40	TAA:SCL	クロックから出力確定までの時間	100 kHz モード	0	3500	ns	—
			400 kHz モード	0	1000	ns	
			1 MHz モード (Note 1)	0	350	ns	
IS45	TBF:SDA	バスフリー時間	100 kHz モード	4.7	—	ms	次の伝送が開始可能になるまでに必要なバスフリー時間
			400 kHz モード	1.3	—	ms	
			1 MHz モード (Note 1)	0.5	—	ms	
IS50	CB	バス負荷容量	—	—	pF	パラメータ DO58 参照	

Note 1: 全ての I2Cx ピンの最大静電容量 = 10 pF (1 MHz モードのみ) です。

# PIC32MZ グラフィック (DA) ファミリ

図 44-20: CANx モジュール I/O タイミング特性

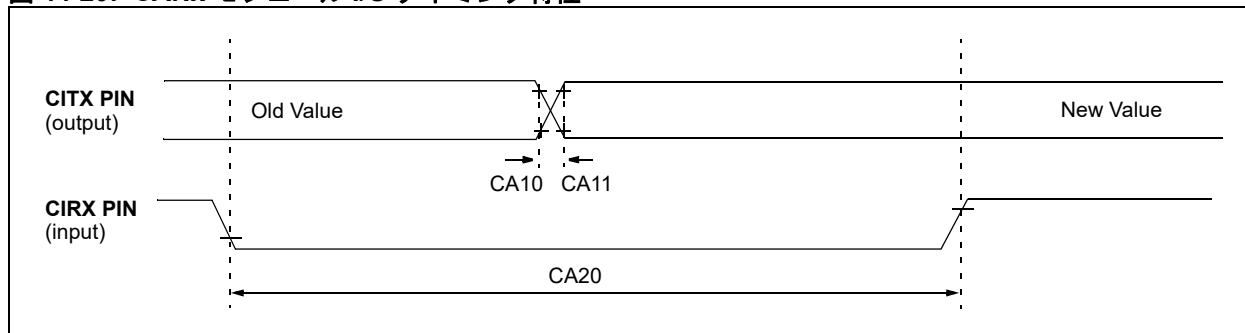


表 44-44: CANx モジュール I/O 要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性 (1)	Min.	Typ. (2)	Max.	単位	条件
CA10	TioF	ポート出力立ち下がり時間	—	—	—	ns	パラメータ DO32 参照
CA11	TioR	ポート出力立ち上がり時間	—	—	—	ns	パラメータ DO31 参照
CA20	Tcwf	CAN 復帰フィルタをトリガするためのパルス幅	700	—	—	ns	—

**Note 1:** これらのパラメータは特性評価済みですが、製品によるテストは実施していません。

**2:** 「Typ」列のデータは、特に明記のない限り 3.3 V、+25 °Cでの値です。これらのパラメータは設計上の目安であり、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-45: ADC モジュール仕様

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
<b>デバイス電源</b>							
AD01	AVDD	モジュールの $V_{DD}$ 電源	$(V_{DDIO} - 0.3)$ または 2.3 の大きい方	—	$(V_{DDIO} + 0.3)$ または 3.6 の小さい方	V	—
AD02	AVSS	モジュール $V_{SS}$ 電源	$V_{SS}$	—	$V_{SS} + 0.3$	V	—
<b>参照入力</b>							
AD05	VREFH	参照電圧 HIGH	$V_{REFL} + 1.8$	—	AVDD	V	(Note 1)
AD06	VREFL	参照電圧 LOW	AVSS	—	$V_{REFH} - 1.8$	V	(Note 1)
AD07	VREF	絶対参照電圧 ( $V_{REFH} - V_{REFL}$ )	1.8	—	AVDD	V	(Note 2)
AD08	IREF	消費電流	—	102	—	$\mu\text{A}$	ADC は動作中またはスタンバイ中
<b>アナログ入力</b>							
AD12	VINH-VINL	フルスケール入カスパン	$V_{REFL}$	—	$V_{REFH}$	V	—
AD13	VINL	絶対 $V_{INL}$ 入力電圧	AVSS	—	$V_{REFL}$	V	—
AD14	VINH	絶対 $V_{INH}$ 入力電圧	AVSS	—	$V_{REFH}$	V	—
<b>ADC 精度 - 外部 <math>V_{REF+}/V_{REF-}</math> を使って計測</b>							
AD20c	Nr	分解能	6	—	12	bit	6/8/10/12 ビット分解能を選択可能
AD21c	INL	積分非直線性	—	$\pm 3$	—	LSb	$V_{INL} = AV_{SS} = V_{REFL} = 0 \text{ V}$ 、 $AV_{DD} = V_{REFH} = 3.3 \text{ V}$
AD22c	DNL	微分非直線性	—	$\pm 1$	—	LSb	$V_{INL} = AV_{SS} = V_{REFL} = 0 \text{ V}$ 、 $AV_{DD} = V_{REFH} = 3.3 \text{ V}$
AD23c	GERR	ゲイン誤差	—	$\pm 8$	—	LSb	$V_{INL} = AV_{SS} = V_{REFL} = 0 \text{ V}$ 、 $AV_{DD} = V_{REFH} = 3.3 \text{ V}$
AD24c	EOFF	オフセット誤差	—	$\pm 2$	—	LSb	$V_{INL} = AV_{SS} = 0 \text{ V}$ 、 $AV_{DD} = 3.3 \text{ V}$
AD25c	—	単調性	—	—	—	—	保証 (Note 2)
<b>動的性能</b>							
AD31b	SINAD	SINAD (Signal-to-Noise and Distortion Ratio)	—	65	—	dB	シングルエンド (Note 2, 3)
AD34b	ENOB	有効ビット数	—	10.5	—	bit	(Note 2, 3)

**Note 1:** これらのパラメータは特性評価済みではなく、製造時の検査は実施していません。

**2:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

**3:** 1 kHz 正弦波で特性を計測。

**4:** ADC モジュールは  $V_{BORIOMIN} < V_{DDIO} < V_{DDIOMIN}$  の条件で動作しますが、性能は低下します。特に明記しない場合、モジュールの動作は保証されますが、特性は保証できません。



# PIC32MZ グラフィック (DA) ファミリ

表 44-46: A/D 変換タイミング要件

AC 特性 <sup>(2)</sup>		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ , $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性	Min.	Typ. <sup>(1)</sup>	Max.	単位	条件
<b>クロック パラメータ</b>							
AD50	TAD	ADC クロック周期	20	—	6250	ns	—
<b>スループットレート</b>							
AD51	FTP	ADC0 ~ ADC4 のサンプリングレート (クラス 1 入力)	—	—	3.125	Msp	12 ビット分解能のソース インピーダンス $\leq 200\Omega$
			—	—	3.57	Msp	10 ビット分解能のソース インピーダンス $\leq 200\Omega$
			—	—	4.16	Msp	8 ビット分解能のソース インピーダンス $\leq 200\Omega$
			—	—	5	Msp	6 ビット分解能のソース インピーダンス $\leq 200\Omega$
	ADC7 のサンプリングレート (クラス 2 およびクラス 3 入力)	—	—	2.94	Msp	12 ビット分解能のソース インピーダンス $\leq 200\Omega$	
		—	—	3.33	Msp	10 ビット分解能のソース インピーダンス $\leq 200\Omega$	
<b>タイミング パラメータ</b>							
AD60	TSAMP	ADC0 ~ ADC4 のサンプリング時間 (クラス 1 入力)	3	—	—	TAD	ソース インピーダンス $\leq 200\Omega$ 、最大 ADC クロック ソース インピーダンス $\leq 500\Omega$ 、最大 ADC クロック ソース インピーダンス $\leq 1 \text{ K}\Omega$ 、最大 ADC クロック ソース インピーダンス $\leq 5 \text{ K}\Omega$ 、最大 ADC クロック
			4	—	—		
			5	—	—		
13	—		—				
ADC7 のサンプリング時間 (クラス 2 およびクラス 3 入力)	4	—	—	TAD	ソース インピーダンス $\leq 200\Omega$ 、最大 ADC クロック ソース インピーダンス $\leq 500\Omega$ 、最大 ADC クロック ソース インピーダンス $\leq 1 \text{ K}\Omega$ 、最大 ADC クロック ソース インピーダンス $\leq 5 \text{ K}\Omega$ 、最大 ADC クロック		
	5	—	—				
	6	—	—				
14	—	—	—	—			
ADC7 のサンプリング時間 (クラス 2 およびクラス 3 入力)	表 44-47 参照	—	—	—	TAD	CVDEN (ADCCON1<11>) = 1	
		—	—	—			
		—	—	—			
AD62	TCONV	変換時間 (サンプリング時間終了後)	—	—	13	TAD	12 ビット分解能 10 ビット分解能 8 ビット分解能 6 ビット分解能
			—	—	11		
			—	—	9		
			—	—	7		
AD65	TWAKE	低消費電力モードからの復帰時間	—	500	—	TAD	500 TAD または $20 \mu\text{s}$ の小さい方
			—	20	—		

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

**Note 2:** ADC モジュールは  $V_{BORIOMIN} < V_{DDIO} < V_{DDIOMIN}$  の条件で動作しますが、性能は低下します。特に明記しない場合、モジュールの動作は保証されますが、特性は保証できません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-47: ADC サンプリング時間 (CVD 有効時)

AC 特性 <sup>(2)</sup>			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ. <sup>(1)</sup>	Max.	単位	条件
AD60a	TSAMP	ADC7 (クラス 2 およびクラス 3 入力) のサンプリング時間 CVDEN ビット (ADCCON1<11>) = 1	8			TAD	ソース インピーダンス $\leq 200 \Omega$ CVDCPL<2:0> (ADCCON2<28:26>) = 001
			9				CVDCPL<2:0> (ADCCON2<28:26>) = 010
			11				CVDCPL<2:0> (ADCCON2<28:26>) = 011
			12	—	—		CVDCPL<2:0> (ADCCON2<28:26>) = 100
			14				CVDCPL<2:0> (ADCCON2<28:26>) = 101
			16				CVDCPL<2:0> (ADCCON2<28:26>) = 110
			17				CVDCPL<2:0> (ADCCON2<28:26>) = 111
			10			TAD	ソース インピーダンス $\leq 500 \Omega$ CVDCPL<2:0> (ADCCON2<28:26>) = 001
			12				CVDCPL<2:0> (ADCCON2<28:26>) = 010
			14				CVDCPL<2:0> (ADCCON2<28:26>) = 011
			16	—	—		CVDCPL<2:0> (ADCCON2<28:26>) = 100
			18				CVDCPL<2:0> (ADCCON2<28:26>) = 101
			19			CVDCPL<2:0> (ADCCON2<28:26>) = 110	
			21			CVDCPL<2:0> (ADCCON2<28:26>) = 111	
			13			TAD	ソース インピーダンス $\leq 1 \text{ K}\Omega$ CVDCPL<2:0> (ADCCON2<28:26>) = 001
			16				CVDCPL<2:0> (ADCCON2<28:26>) = 010
18			CVDCPL<2:0> (ADCCON2<28:26>) = 011				
21	—	—	CVDCPL<2:0> (ADCCON2<28:26>) = 100				
23			CVDCPL<2:0> (ADCCON2<28:26>) = 101				
26			CVDCPL<2:0> (ADCCON2<28:26>) = 110				
28			CVDCPL<2:0> (ADCCON2<28:26>) = 111				
41			TAD	ソース インピーダンス $\leq 5 \text{ K}\Omega$ CVDCPL<2:0> (ADCCON2<28:26>) = 001			
48				CVDCPL<2:0> (ADCCON2<28:26>) = 010			
56				CVDCPL<2:0> (ADCCON2<28:26>) = 011			
63	—	—		CVDCPL<2:0> (ADCCON2<28:26>) = 100			
70				CVDCPL<2:0> (ADCCON2<28:26>) = 101			
78				CVDCPL<2:0> (ADCCON2<28:26>) = 110			
85			CVDCPL<2:0> (ADCCON2<28:26>) = 111				

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

**2:** ADC モジュールは  $V_{BORIOMIN} < V_{DDIO} < V_{DDIOMIN}$  の条件で動作しますが、性能は低下します。特に明記しない場合、モジュールの動作は保証されますが、特性は保証できません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-48: 温度センサ仕様

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
TS10	VTS	電圧変化率	—	5	—	mV/°C	—
TS11	TR	分解能	-2	—	+2	°C	—
TS12	IVTEMP	電圧レンジ	0.5	—	1.5	V	—
TS13	TMIN	最小温度	—	-40	—	°C	IVTEMP = 0.5 V
TS14	TMAX	最大温度	—	160	—	°C	IVTEMP = 1.5 V

**Note 1:** 温度センサモジュールは  $V_{BORIOMIN} < V_{DDIO} < V_{DDIOMIN}$  の条件で動作しますが、性能は低下します。特に明記しない場合、モジュールの動作は保証されますが、特性は保証できません。

# PIC32MZ グラフィック (DA) ファミリ

図 44-21: パラレルスレーブポート タイミング

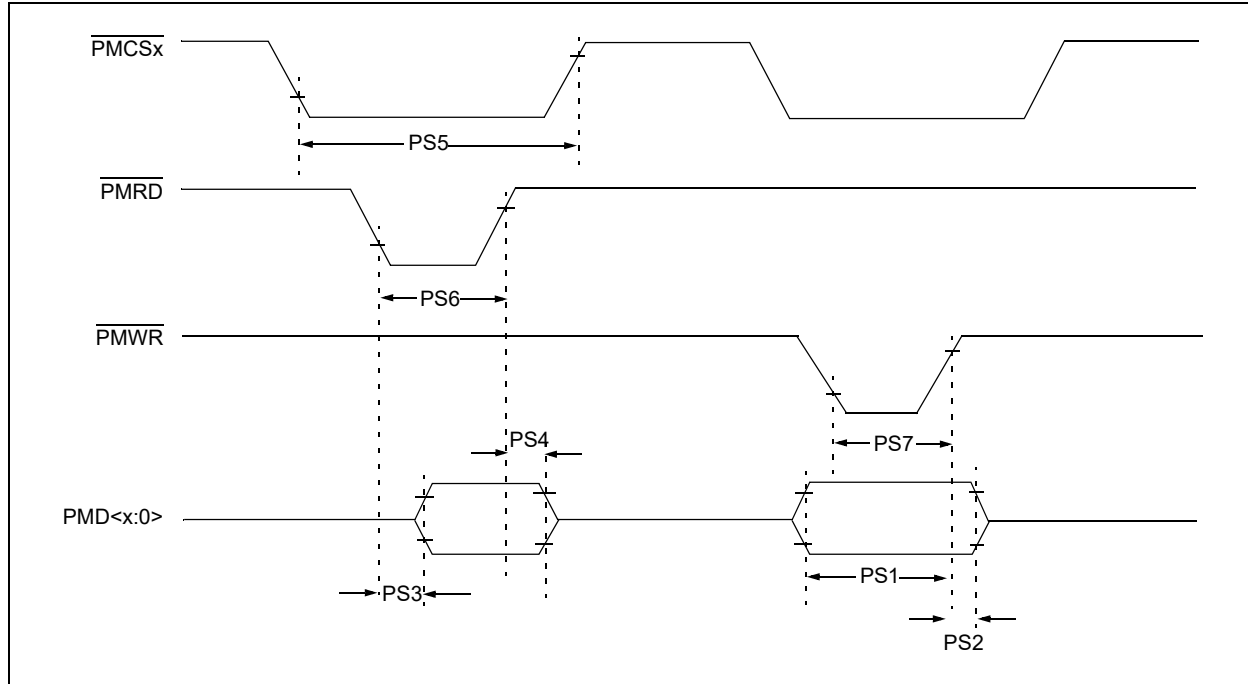


表 44-49: パラレルスレーブポート要件

AC 特性				標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)			
パラメータ No.	記号	特性 (1)	Min.	Typ.	Max.	単位	条件
PS1	TdtV2wrH	PMWR または $\overline{\text{PMCSx}}$ 非アクティブまでのデータ入力有効時間 (セットアップ時間)	20	—	—	ns	—
PS2	TwrH2dtl	PMWR または $\overline{\text{PMCSx}}$ 非アクティブからデータ入力無効までの時間 (ホールド時間)	40	—	—	ns	—
PS3	TrdL2dtV	$\overline{\text{PMRD}}$ および $\overline{\text{PMCSx}}$ アクティブからデータ出力有効までの時間	—	—	60	ns	—
PS4	TrdH2dtl	$\overline{\text{PMRD}}$ アクティブまたは $\overline{\text{PMCSx}}$ 非アクティブからデータ出力無効までの時間	0	—	10	ns	—
PS5	Tcs	$\overline{\text{PMCSx}}$ アクティブ時間	$TPBCLK2 + 40$	—	—	ns	—
PS6	TWR	$\overline{\text{PMWR}}$ アクティブ時間	$TPBCLK2 + 25$	—	—	ns	—
PS7	TRD	$\overline{\text{PMRD}}$ アクティブ時間	$TPBCLK2 + 25$	—	—	ns	—

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

図 44-22: パラレルマスタ ポート読み出しタイミング図

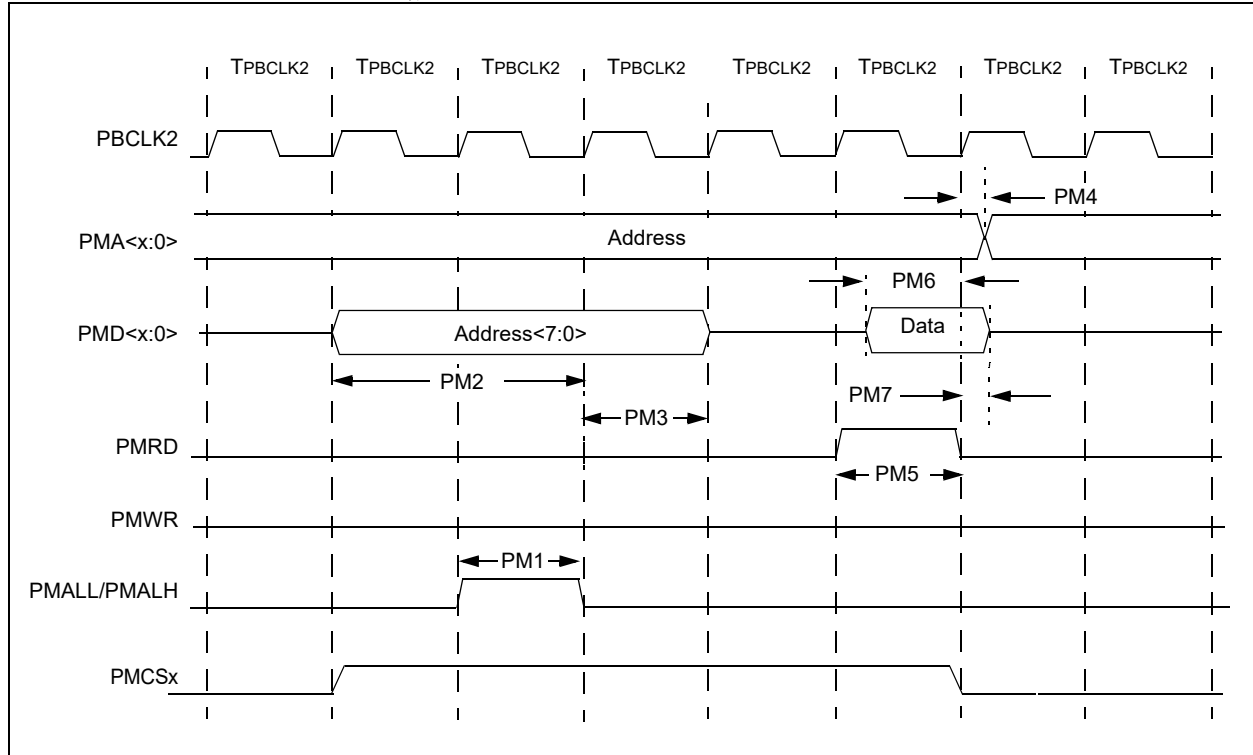


表 44-50: パラレルマスタ ポート読み出しタイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	記号	特性 <sup>(1)</sup>	Min.	Typ.	Max.	単位	条件
PM1	TLAT	PMALL/PMALH パルス幅	—	1 TPBCLK2	—	—	—
PM2	TADSU	アドレス出力有効から PMALL/PMALH 無効までの時間 (アドレス セットアップ時間)	—	2 TPBCLK2	—	—	—
PM3	TADHOLD	PMALL/PMALH 無効からアドレス出力無効までの時間 (アドレス ホールド時間)	—	1 TPBCLK2	—	—	—
PM4	TAHOLD	PMRD 非アクティブからアドレス出力無効までの時間 (アドレス ホールド時間)	5	—	—	ns	—
PM5	TRD	PMRD パルス幅	—	1 TPBCLK2	—	—	—
PM6	TDSU	PMRD または PMENB アクティブからデータ入力有効までの時間 (データ セットアップ時間)	15	—	—	ns	—
PM7	TDHOLD	PMRD または PMENB 非アクティブからデータ入力無効までの時間 (データ ホールド時間)	5	—	—	ns	—

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

図 44-23: パラレルマスタ ポート書き込みタイミング図

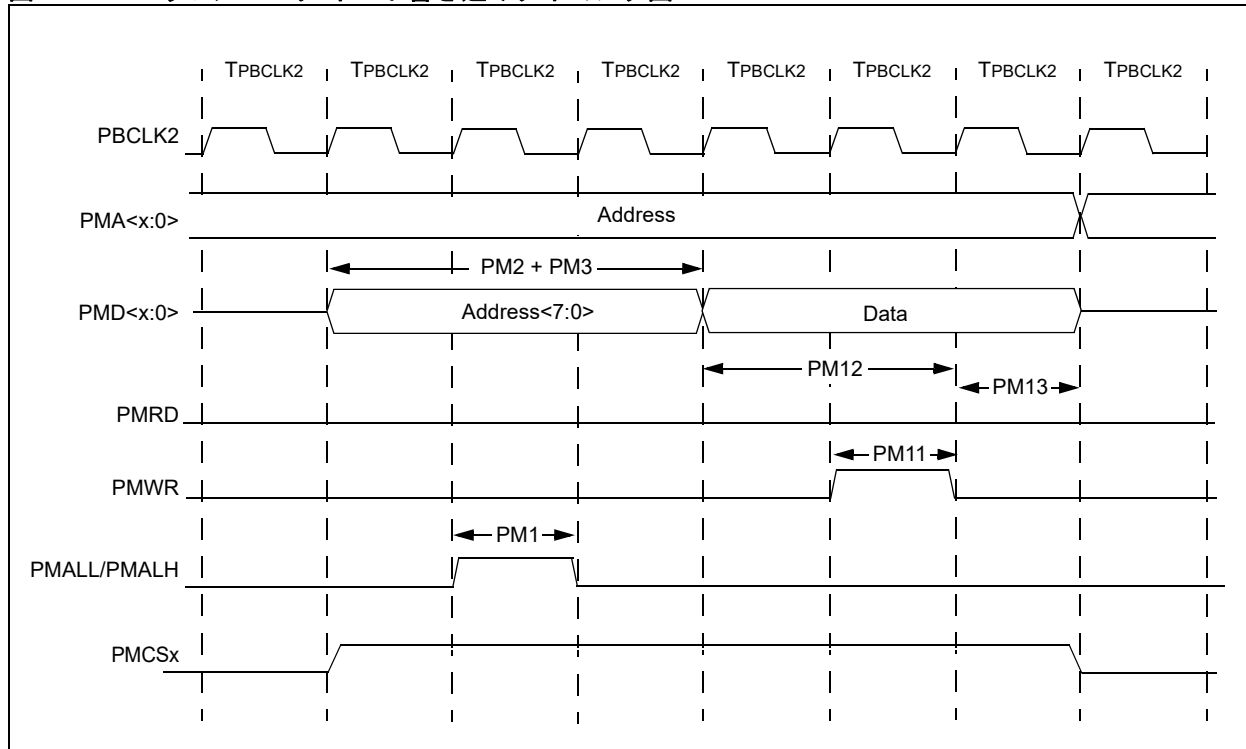


表 44-51: パラレルマスタ ポート書き込みタイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性 (1)	Min.	Typ.	Max.	単位	条件
PM11	TWR	PMWR パルス幅	—	1 TPBCLK2	—	—	—
PM12	TdVSU	PMWR または PMENB 非アクティブまでのデータ出力有効時間 (データ セットアップ時間)	—	2 TPBCLK2	—	—	—
PM13	TdVHOLD	PMWR または PMENB 非アクティブからデータ出力無効までの時間 (データホールド時間)	—	1 TPBCLK2	—	—	—

Note 1: これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-52: USB OTG 電氣的仕様

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^{\circ}\text{C} \leq T_A \leq +85 \text{ }^{\circ}\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性 (1)	Min.	Typ.	Max.	単位	条件
USB313	Vusb3v3	USB 電圧	3.0	—	3.6	V	USB が正常に動作するには、 $V_{USB3V3}$ 電圧がこのレンジ内である必要があります。
<b>ロースピード / フルスピード モード</b>							
USB315	Vilusb	USB バッファ入力電圧 LOW	—	—	0.8	V	—
USB316	Vihusb	USB バッファ入力電圧 HIGH	2.0	—	—	V	—
USB318	Vdifs	差動入力感度	0.2	—	—	V	VCM の条件を満たしている時、D+ と D- の電圧差はこの値より大きい必要があります
USB319	VCM	差動コモンモード レンジ	0.8	—	2.5	V	—
USB321	Vol	電圧出力 LOW	0.0	—	0.3	V	1.425 k $\Omega$ の負荷を $V_{USB3V3}$ に接続
USB322	Voh	電圧出力 HIGH	2.8	—	3.6	V	14.25 k $\Omega$ の負荷をグラウンドに接続
<b>ハイスピード モード</b>							
USB323	VHSDI	差動入力信号レベル	150	—	—	mV	—
USB324	VHSSQ	SQ 検出しきい値	100	—	150	mV	—
USB325	VHSCM	コモンモード電圧レンジ	-50	—	500	mV	—
USB326	VHSH	データ信号 HIGH	360	—	440	mV	—
USB327	VHSOL	データ信号 LOW	-10	—	10	mV	—
USB328	VCHIRPJ	Chirp J レベル	700	—	1100	mV	—
USB329	Vchirpk	Chirp K レベル	-900	—	-500	mV	—
USB330	Zhsdrv	ドライバ出力抵抗	—	45	—	$\Omega$	—

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-53: Ethernet モジュール仕様

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	特性	Min.	Typ.	Max.	単位	条件
<b>MIIM タイミング要件</b>						
ET1	MDC デューティサイクル	40	—	60	%	—
ET2	MDC 周期	400	—	—	ns	—
ET3	MDIO 出力セットアップ/ホールド	10	—	10	ns	図 44-24 参照
ET4	MDIO 入力セットアップ/ホールド	0	—	300	ns	図 44-25 参照
<b>MII タイミング要件</b>						
ET5	TX クロック周波数	—	25	—	MHz	—
ET6	TX クロック デューティサイクル	35	—	65	%	—
ET7	ETXD <sub>x</sub> 、ETEN、ETXERR 出力遅延	0	—	25	ns	図 44-26 参照
ET8	RX クロック周波数	—	25	—	MHz	—
ET9	RX クロック デューティサイクル	35	—	65	%	—
ET10	ERXD <sub>x</sub> 、ERXDV、ERXERR セットアップ/ホールド	10	—	30	ns	図 44-27 参照
<b>RMII タイミング要件</b>						
ET11	参照クロック周波数	—	50	—	MHz	—
ET12	参照クロック デューティサイクル	35	—	65	%	—
ET13	ETXD <sub>x</sub> 、ETEN、セットアップ/ホールド	2	—	16	ns	—
ET14	ERXD <sub>x</sub> 、ERXDV、ERXERR セットアップ/ホールド	2	—	16	ns	—

図 44-24: PIC32 をソースとする場合の MDIO

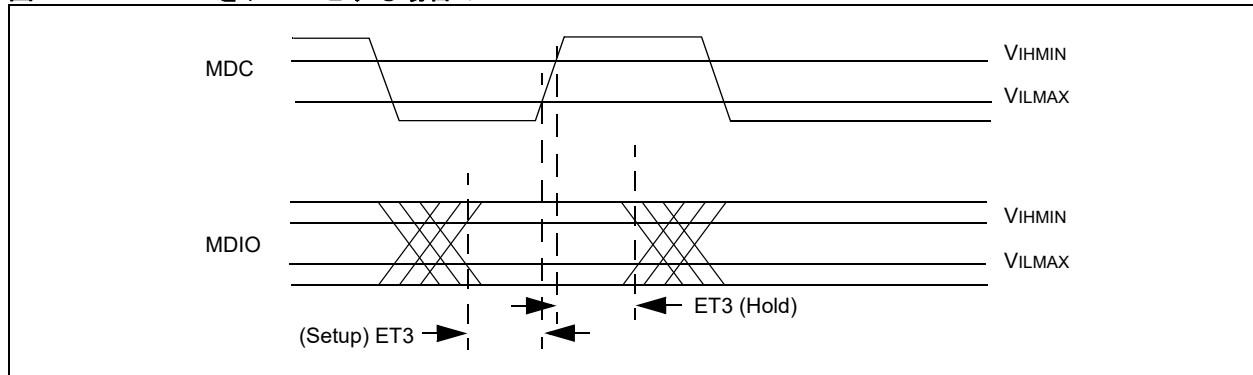
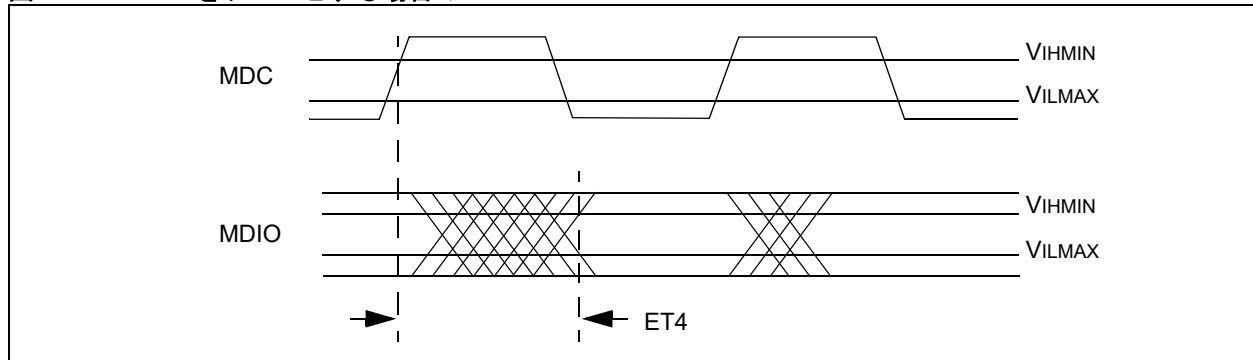


図 44-25: PHY をソースとする場合の MDIO





# PIC32MZ グラフィック (DA) ファミリ

図 44-26: MII における信号送信のタイミング関係

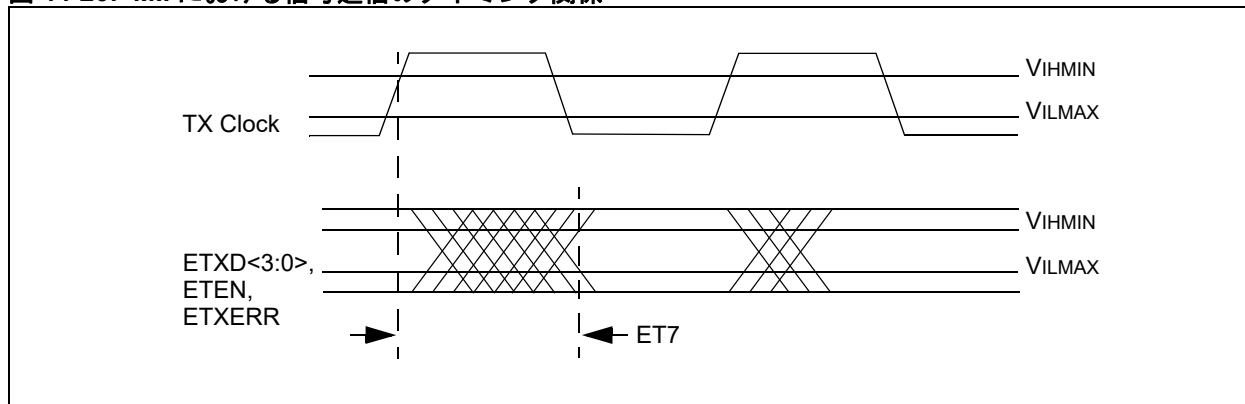
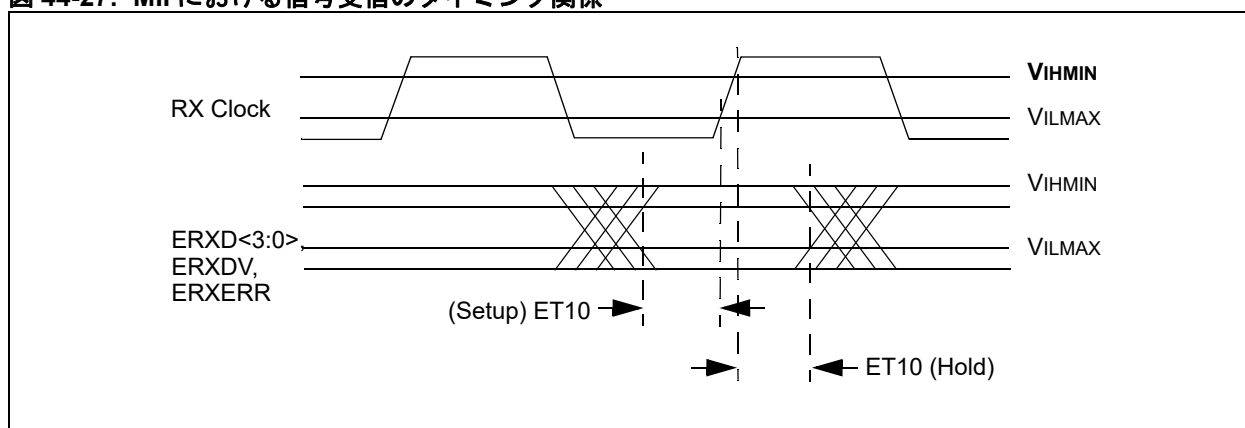


図 44-27: MII における信号受信のタイミング関係



# PIC32MZ グラフィック (DA) ファミリ

図 44-28: EBI ページ読み出しタイミング

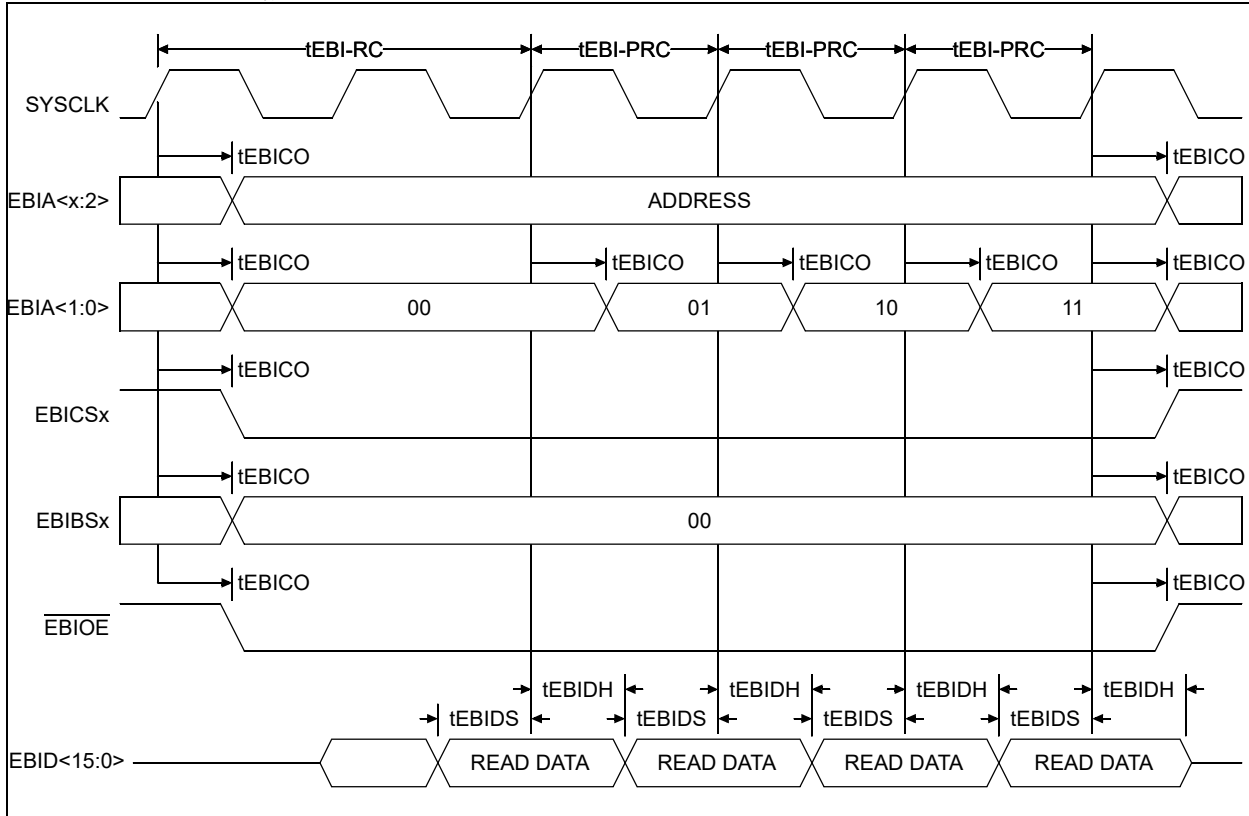
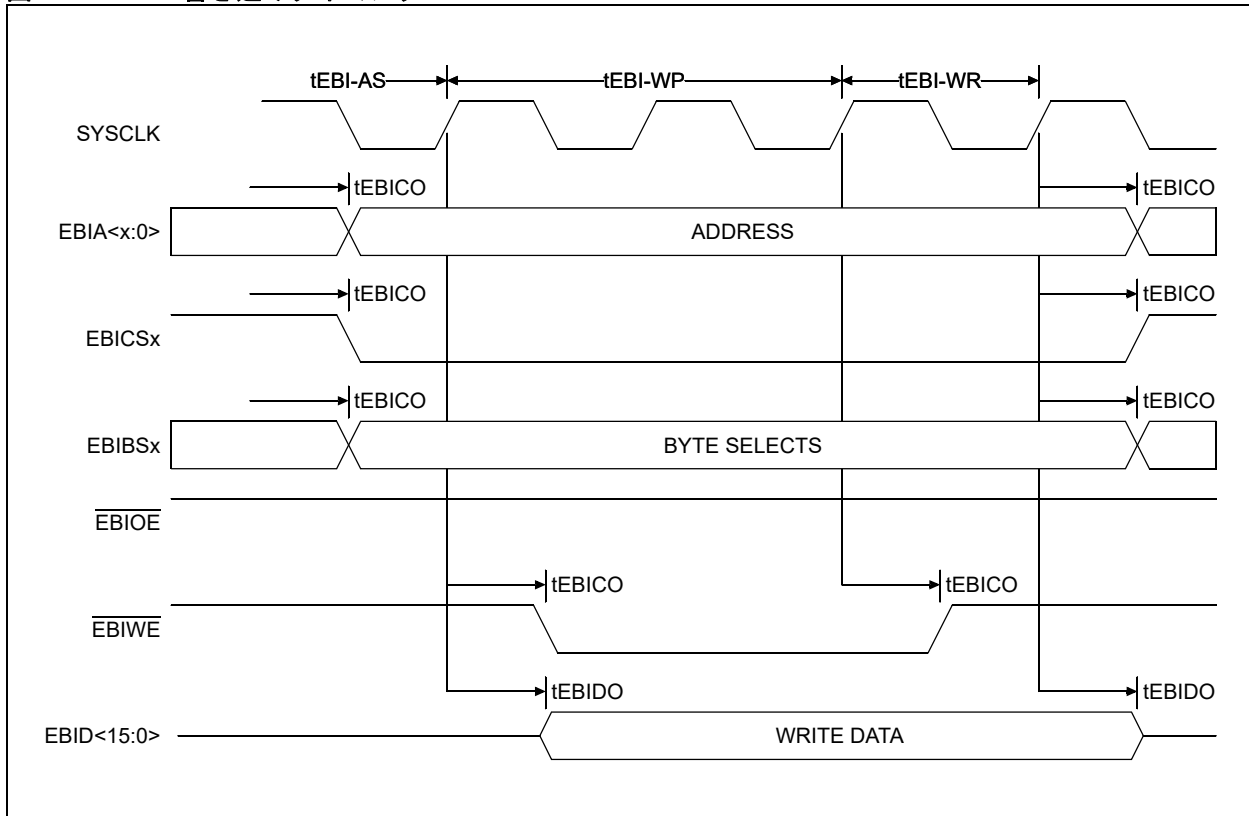


図 44-29: EBI 書き込みタイミング



# PIC32MZ グラフィック (DA) ファミリ

表 44-54: EBI タイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
EB10	TEBICLK	内部 EBI クロック周期 (SYSCLK)	5	—	—	ns	—
EB11	TEBIRC	EBI 読み出しサイクル時間 (TRC<5:0>)	10	—	—	ns	—
EB12	TEBIPRC	EBI ページ読み出しサイクル時間 (TPRC<3:0>)	10	—	—	ns	—
EB13	TEBIAS	EBI アドレス書き込みセットアップ時間 (TAS<1:0>)	5	—	—	ns	—
EB14	TEBIWP	EBI 書き込みパルス幅 (TWP<5:0>)	5	—	—	ns	—
EB15	TEBIWR	EBI 書き込みリカバリ時間 (TWR<1:0>)	5	—	—	ns	—
EB16	TEBICO	EBI 出力制御信号遅延	—	—	5	ns	Note 1 参照
EB17	TEBIDO	EBI 出力データ信号遅延	—	—	5	ns	Note 1 参照
EB18	TEBIDS	EBI 入力データ セットアップ時間	2.5	—	—	ns	Note 1 参照
EB19	TEBIDH	EBI 入力データホールド時間	1.5	—	—	ns	Note 1、2 参照

Note 1: 最大ピン容量 = 10 pF

2: EBI アドレス変更からのホールド時間は 0 ns です。

表 44-55: GLCD コントローラ タイミング仕様

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^\circ\text{C} \leq T_A \leq +85 \text{ }^\circ\text{C}$ (産業用温度レンジ)				
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
GD20	tGCLK	画素クロック周波数	—	—	50	MHz	—

# PIC32MZ グラフィック (DA) ファミリ

表 44-56: DDR2 SDRAM コントローラ タイミング仕様

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)				
パラメータ No.	記号	特性 <sup>(1)</sup>	Min.	Typ.	Max.	単位	条件
DDR10	tCK	クロック周波数	—	5	—	ns	—
DDR11	tDUTY	デューティサイクル	48	50	52	%	—
DDR12	tRCD	アクティベート - 読み書きコマンド間の遅延時間	20	—	—	ns	—
DDR13	tRP	プリチャージ - アクティベートコマンド間の遅延時間	20	—	—	ns	—
DDR14	tRC	アクティベート - Ref/ アクティベートコマンド間の遅延時間	110	—	120	ns	—
DDR15	tRAS	アクティベート - プリチャージコマンド間の遅延時間	40	70000	—	ns	Note 1
DDR16	tRFC	オートリフレッシュ - アクティベート / オートリフレッシュコマンド間の遅延時間	130	—	—	ns	Note 2
DDR17	tREFI	周期的リフレッシュの平均間隔	—	—	7.8	$\mu s$	Note 2
DDR18	tCKE	DDRCKE の最小 HIGH/LOW パルス幅	6	—	—	ntCK	—
DDR19	tRRD	アクティベート - アクティベートコマンド間の遅延時間 (1 KB ページサイズ)	10	—	—	ns	Note 3
DDR20	tFAW	4バンク アクティベート ウィンドウ (1 KB ページサイズ)	35	—	—	ns	—
DDR21	tWR	書き込みリカバリ時間	25	—	—	ns	—
DDR22	tWTR	内部書き込み - 読み出しコマンド間の遅延時間	10	—	—	ns	Note 4
DDR23	tRTP	内部読み出し - プリチャージコマンド間の遅延時間	10	—	—	ns	Note 1
DDR24	tXSRD	セルフリフレッシュ終了 - 読み出しコマンド間の遅延時間	200	—	—	ntCK	—
DDR25	tXP	プリチャージパワーダウン終了 - 任意コマンド間の遅延時間	6	—	—	ntCK	—
DDR26	tMRD	モードレジスタ設定コマンドのサイクル時間	4	—	—	ntCK	—
DDR27	RL	読み出しレイテンシ	CL	—	—	ntCK	—
DDR28	CL	CAS レイテンシ	3	—	4	ntCK	—
DDR29	WL	書き込みレイテンシ	RL - 1	—	—	ntCK	—
DDR30	BL	バースト長	8	—	—	ntCK	—

**Note 1:** これは最小要件です。tRTP と tRAS(min) の要件を満たしている場合、読み出しからプリチャージまでの最小時間は  $AL + BL / 2$  です。

**2:** リフレッシュ タイミングが違反した場合、データが破損する可能性があります。その場合、有効なデータが再書き込みされるまで、有効な読み出しは実行できません。

**3:** 動作周波数に関係なく、2クロック ( $2 * ntCK$ ) 以上が必要です。

**4:** 動作周波数に関係なく、tWTR は 2クロック ( $2 * ntCK$ ) 以上です。

# PIC32MZ グラフィック (DA) ファミリ

表 44-57: SD ホストコントローラ既定値モード タイミング仕様

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
SD20	tSDCK	クロック周波数	—	—	25	MHz	—
SD21	tDUTY	デューティサイクル	—	50	—	%	—
SD22	tHIGH	クロック HIGH 時間	10	—	—	ns	—
SD23	tLOW	クロック LOW 時間	10	—	—	ns	—
SD24	tRISE	クロック立ち上がり時間	—	10	—	ns	—
SD25	tFALL	クロック立ち下がり時間	—	10	—	ns	—
SD26	tSETUP	入力セットアップ時間	5	—	—	ns	—
SD27	tHOLD	入力ホールド時間	5	—	—	ns	—

表 44-58: SD ホストコントローラ ハイスピードモード タイミング仕様

AC 特性		標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$ (産業用温度レンジ)					
パラメータ No.	記号	特性	Min.	Typ.	Max.	単位	条件
SD30	tSDCK	クロック周波数	—	—	50	MHz	—
SD31	tDUTY	デューティサイクル	—	50	—	%	—
SD32	tHIGH	クロック HIGH 時間	7	—	—	ns	—
SD33	tLOW	クロック LOW 時間	7	—	—	ns	—
SD34	tRISE	クロック立ち上がり時間	—	3	—	ns	—
SD35	tFALL	クロック立ち下がり時間	—	3	—	ns	—
SD36	tSETUP	入力セットアップ時間	6	—	—	ns	—
SD37	tHOLD	入力ホールド時間	2	—	—	ns	—

# PIC32MZ グラフィック (DA) ファミリ

図 44-30: EJTAG タイミング特性

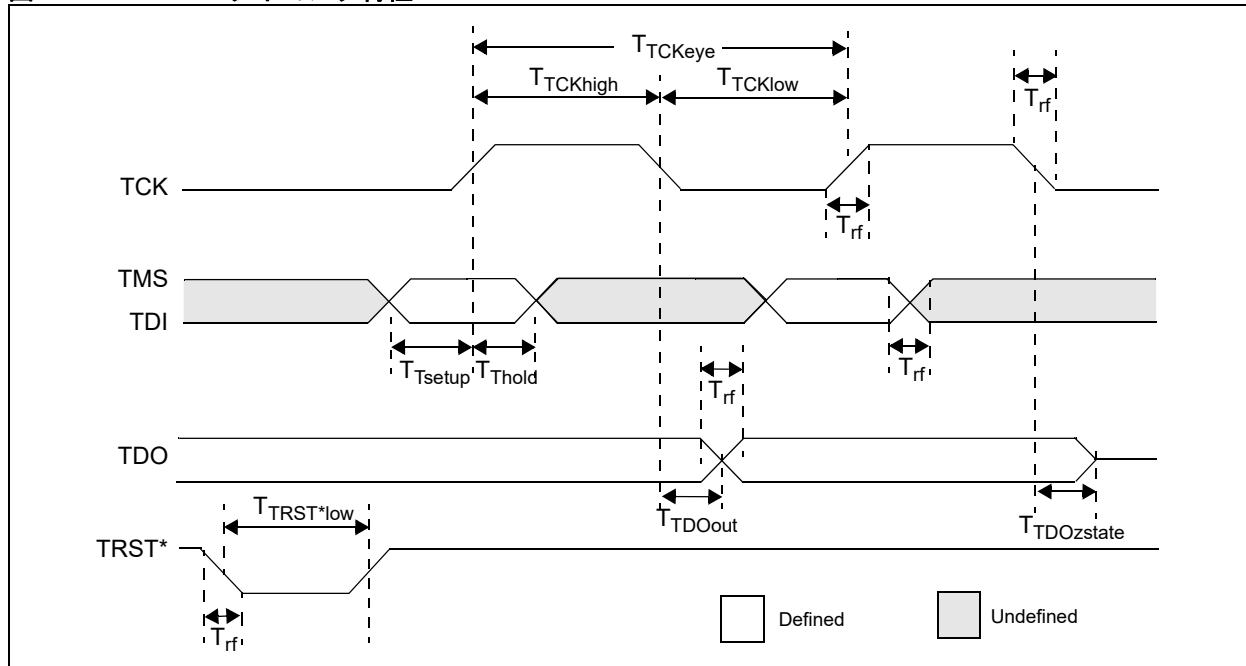


表 44-59: EJTAG タイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 V$ 、 $V_{DDCORE} = 1.7 \sim 1.9 V$ (特に明記しない場合) 動作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$ (産業用温度レンジ)			
パラメータ No.	記号	概要 (1)	Min.	Max.	単位	条件
EJ1	TTCKCYC	TCK サイクル時間	25	—	ns	—
EJ2	TTCKHIGH	TCK HIGH 時間	10	—	ns	—
EJ3	TTCKLOW	TCK LOW 時間	10	—	ns	—
EJ4	TTSETUP	TCK 立ち上がり前の TAP 信号 セットアップ時間	5	—	ns	—
EJ5	TTHOLD	TCK 立ち上がり後の TAP 信号 ホールド時間	3	—	ns	—
EJ6	TTDOOUT	TCK 立ち下がりからの TDO 出 力遅延時間	—	5	ns	—

Note 1: これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

表 44-59: EJTAG タイミング要件

AC 特性			標準動作条件: $V_{DDIO} = 2.2 \sim 3.6 \text{ V}$ 、 $V_{DDCORE} = 1.7 \sim 1.9 \text{ V}$ (特に明記しない場合) 動作温度 $-40 \text{ }^{\circ}\text{C} \leq T_A \leq +85 \text{ }^{\circ}\text{C}$ (産業用温度レンジ)			
パラメータ No.	記号	概要 <sup>(1)</sup>	Min.	Max.	単位	条件
EJ7	TDOZSTATE	TCK 立ち下がりからの TDO 3 ステート遅延時間	—	5	ns	—
EJ8	TTRSTLOW	TRST LOW 時間	25	—	ns	—
EJ9	TRF	TAP 信号立ち上がり / 立ち下がり 時間 (全入出力)	—	—	ns	—

**Note 1:** これらのパラメータは特性評価済みですが、製造時の検査は実施していません。

# PIC32MZ グラフィック (DA) ファミリ

---

---



## 45.0 AC/DC 特性グラフ

**Note:** 以下の図表は限られたサンプル数に基づく統計的な結果であり、設計指標の提供のみを目的としています。ここに記載する性能特性は検査されておらず、保証されません。グラフの一部には、仕様動作レンジ外で計測されたデータも含まれます (例: 仕様レンジ外の電源を使用)。従ってこれらのデータは保証範囲外です。

図 45-1: VOH - 4x ドライバピン

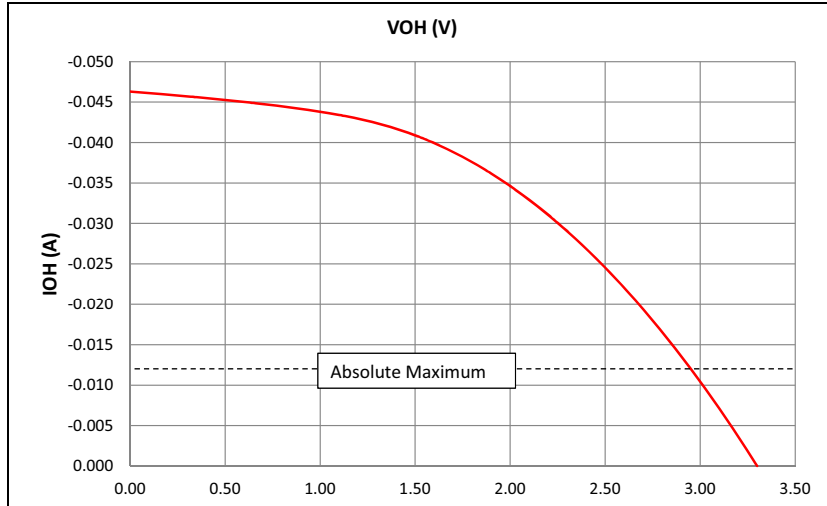


図 45-3: VOH - 8x ドライバピン

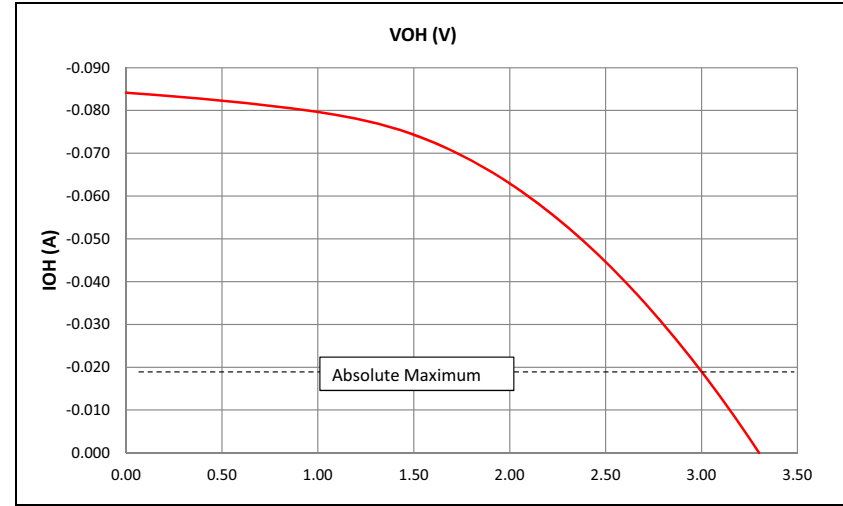


図 45-2: VOL - 4x ドライバピン

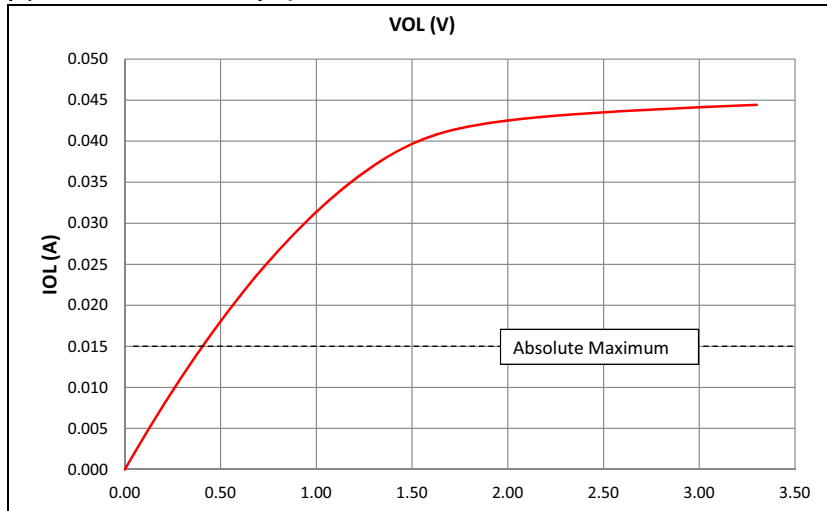


図 45-4: VOL - 8x ドライバピン

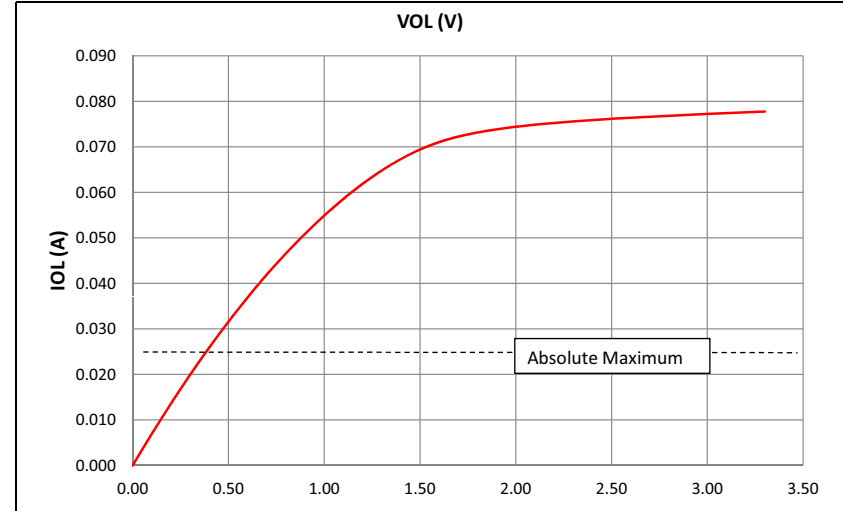


図 45-5: VOH - 12x ドライバピン

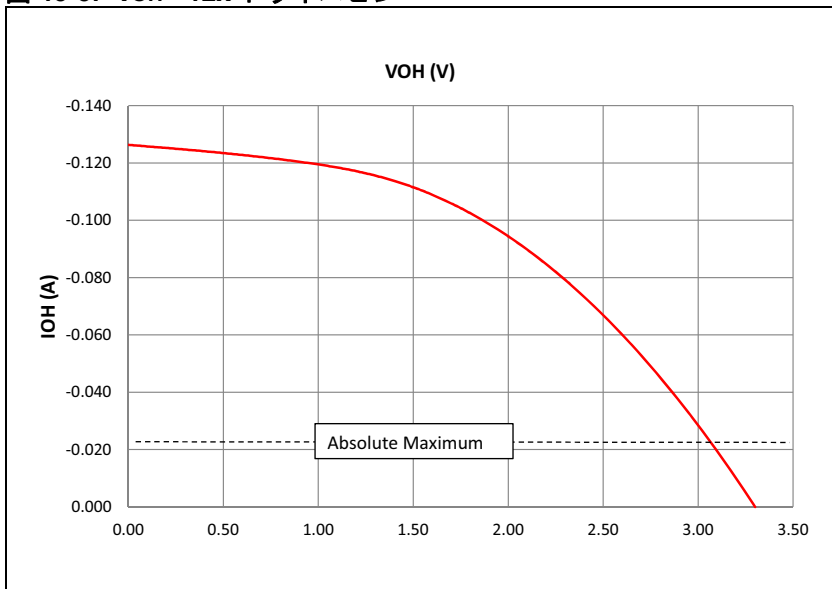


図 45-6: VOL - 12x ドライバピン

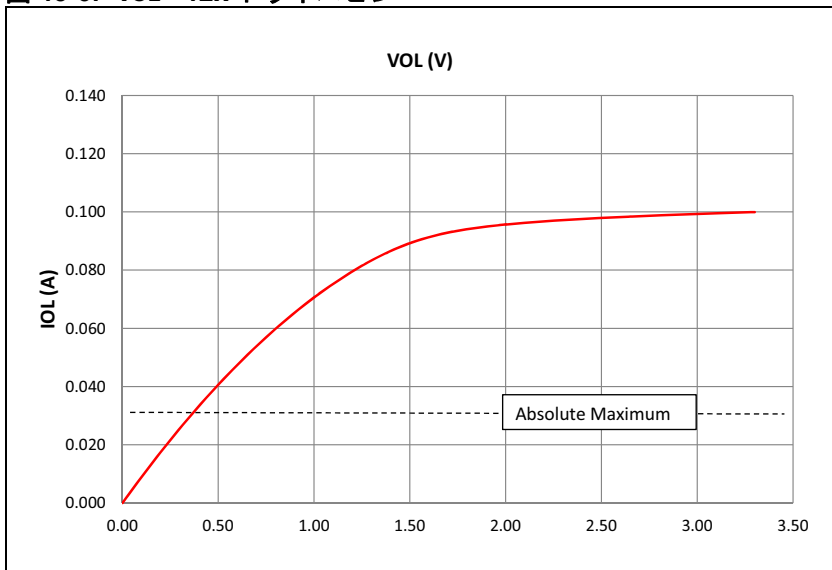
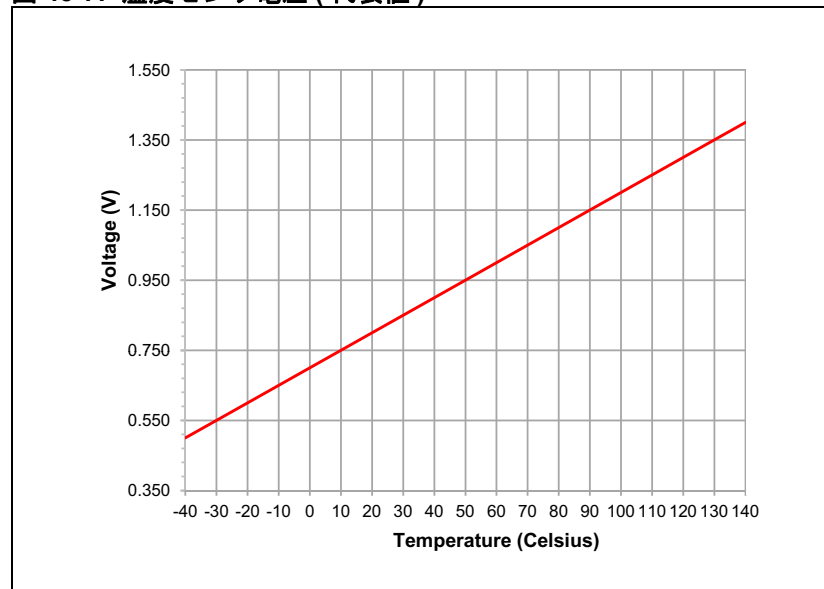


図 45-7: 温度センサ電圧 (代表値)

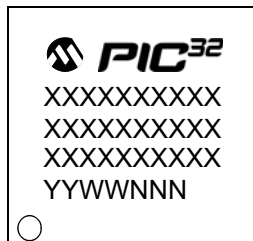


# PIC32MZ グラフィック (DA) ファミリ

## 46.0 パッケージ情報

### 46.1 パッケージのマーキング情報

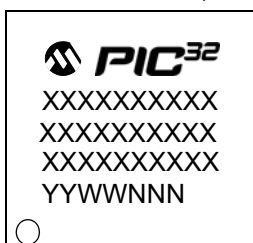
169ピンLFBGA (11x11x1.56 mm)



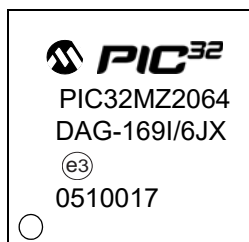
例:



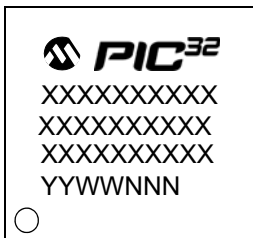
169ピンLFBGA (11x11 mm)



例:



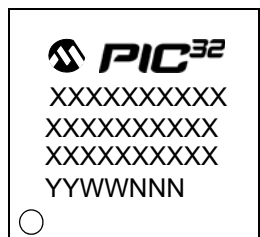
176ピンLQFP (20x20x1.45 mm)



例:



288ピンLFBGA (15x15x1.4 mm)



例:



**凡例:** XX...X お客様固有情報  
Y 年コード (西暦の下1桁)  
YY 年コード (西暦の下2桁)  
WW 週コード (1月1日の週を「01」とする)  
NNN 英数字のトレーサビリティコード  
無光沢スズ (Sn) めっきの使用を示す鉛フリー JEDEC マーク  
\* 本パッケージは鉛フリーです。鉛フリー JEDEC マーク ( e3 ) は外箱に表記しています。

**備考:** Microchip 社の製品番号が1行に収まりきらない場合は複数行を使います。その場合お客様固有情報に使える文字数が制限されます。

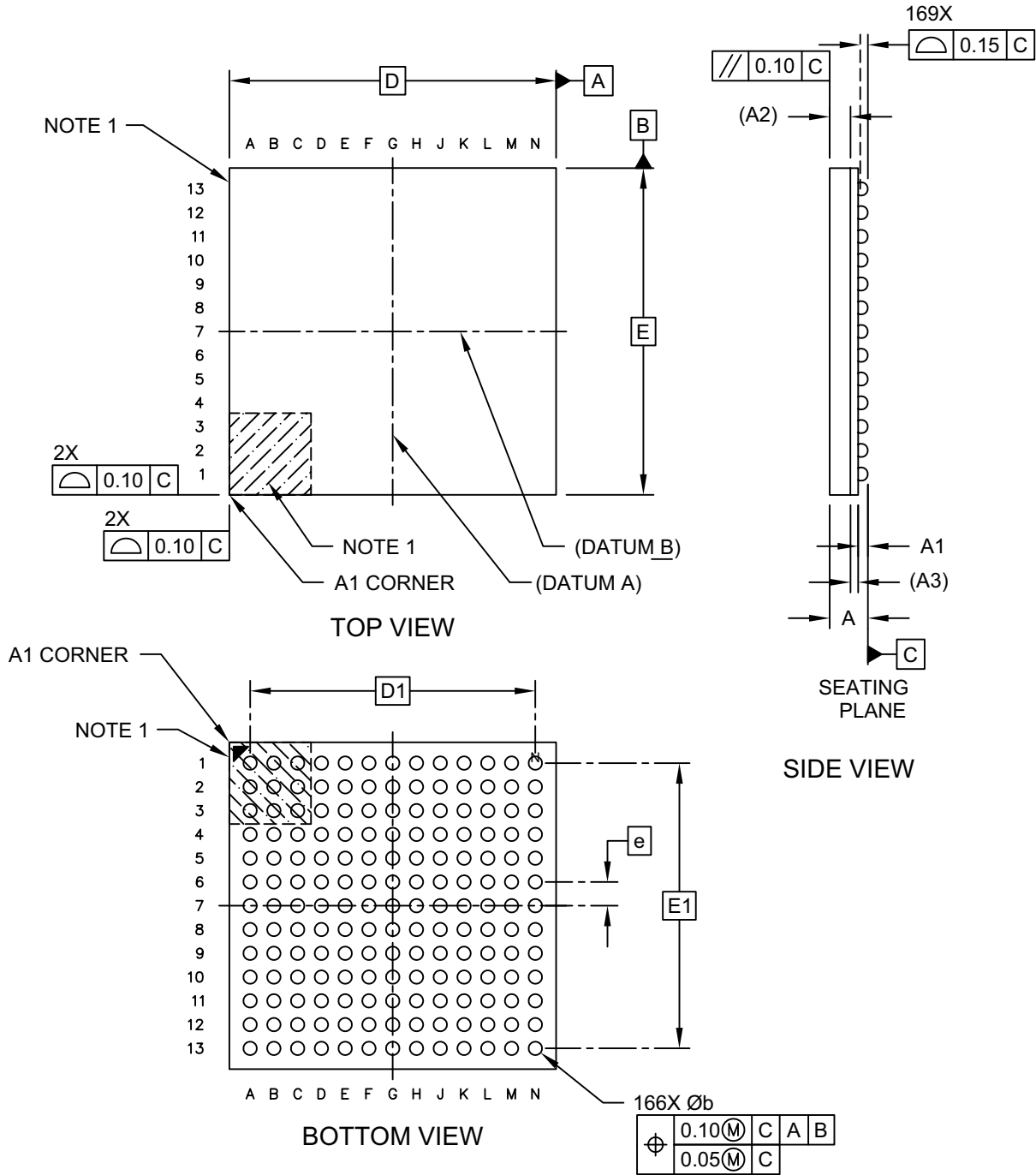
# PIC32MZ グラフィック (DA) ファミリ

## 46.2 パッケージの詳細

以下に各パッケージの技術的詳細を記載します。

### 169-Ball Low Profile Fine Pitch Ball Grid Array (HF) - 11x11x1.4 mm Body [LFBGA]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>

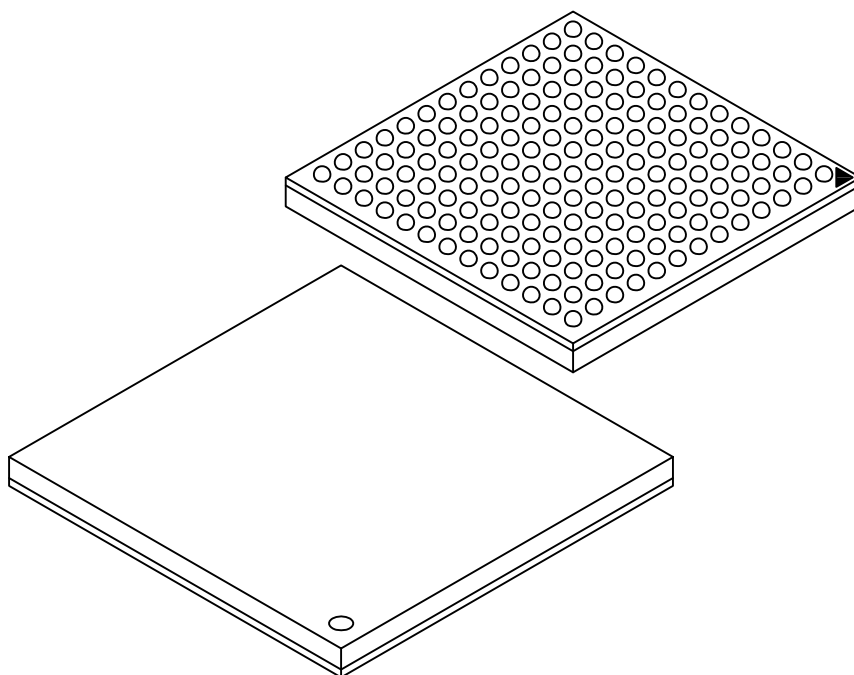


Microchip Technology Drawing C04-365B Sheet 1 of 2

# PIC32MZ グラフィック (DA) ファミリ

## 169-Ball Low Profile Fine Pitch Ball Grid Array (HF) - 11x11x1.4 mm Body [LFBGA]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Terminals (Balls)	N	169		
Pitch	e	0.80 BSC		
Overall Height	A	1.17	1.285	1.40
Terminal (Ball) Height	A1	0.25	0.325	0.40
Mold Cap Thickness	(A2)	0.70 REF		
Substrate Thickness	(A3)	0.26 REF		
Overall Length	D	11.00 BSC		
Overall Width	E	11.00 BSC		
Overall Ball Pitch	D1	9.60		
Overall Ball Pitch	E1	9.60		
Ball Diameter	b	0.40	0.45	0.50

**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

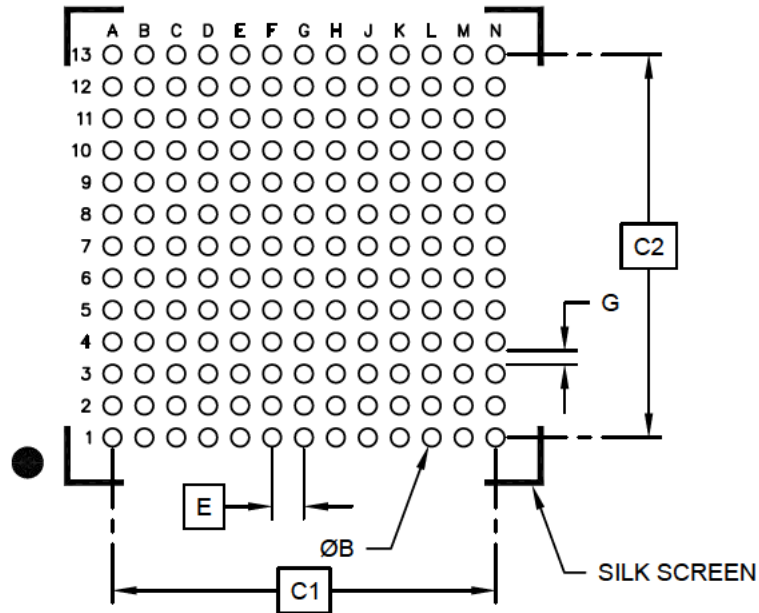
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-365B Sheet 2 of 2

# PIC32MZ グラフィック (DA) ファミリ

## 169 Ball Low Profile Fine Pitch Ball Grid Array (HF) - 11x11x1.4 mm Body [LFBGA]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Contact Pad Spacing	C1	9.60 BSC		
Contact Pad Spacing	C2	9.60 BSC		
Contact Pad Diameter (X169)	B	0.40	0.45	0.50
Pad-to-Pad Clearance	G	0.30		

**Notes:**

1. Dimensioning and tolerancing per ASME Y14.5M

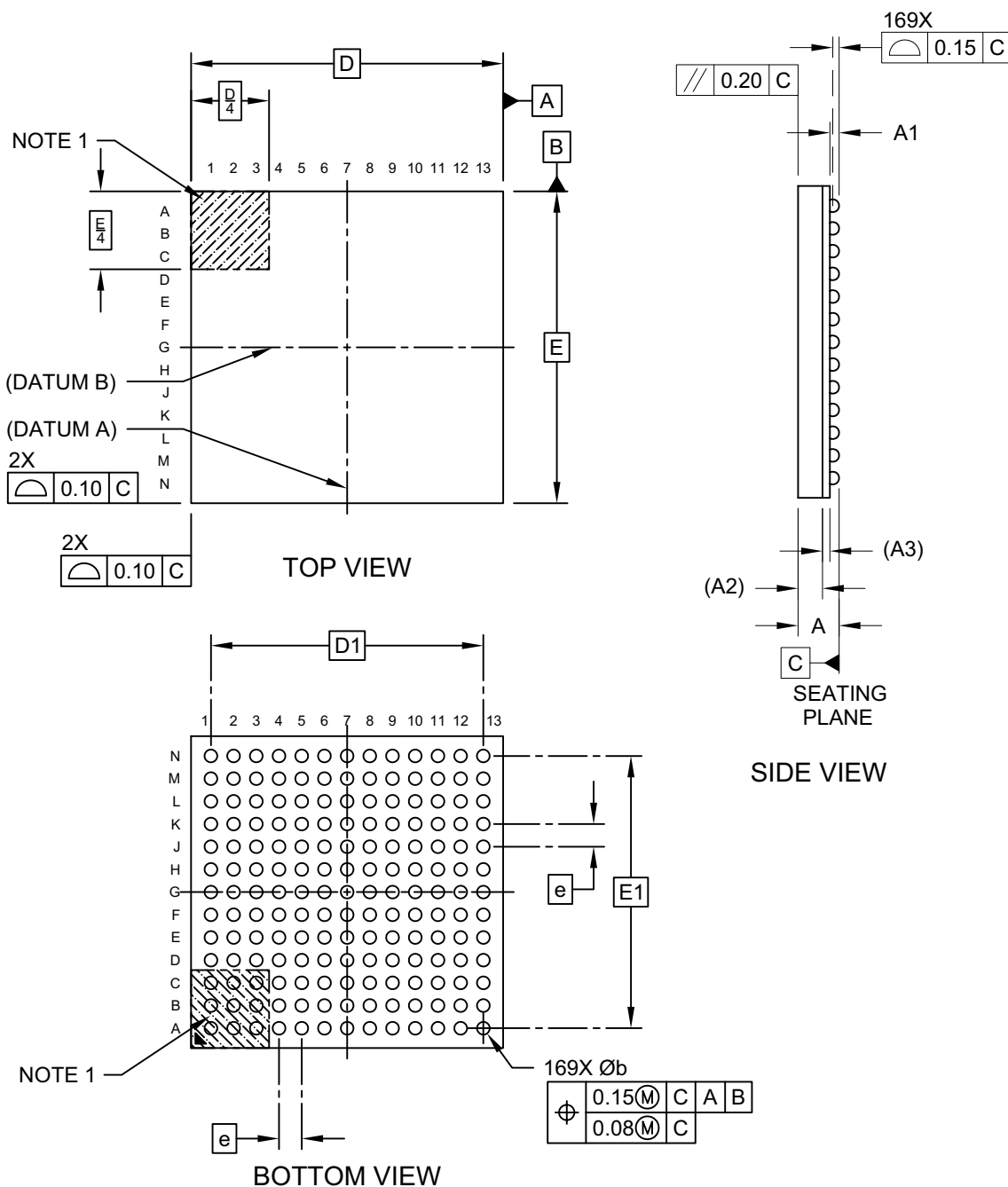
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2365B

# PIC32MZ グラフィック (DA) ファミリ

## 169-Ball Low Profile Ball Grid Array (6JX) - 11x11 mm Body [LFBGA]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>

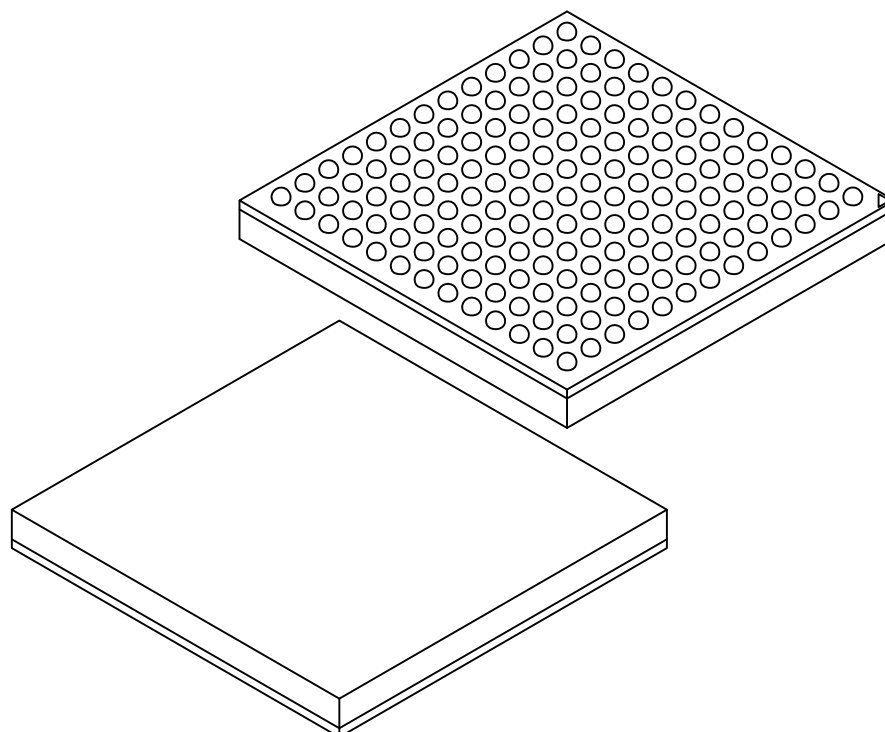


Microchip Technology Drawing C04-439A Sheet 1 of 2

# PIC32MZ グラフィック (DA) ファミリ

## 169-Ball Low Profile Ball Grid Array (6JX) - 11x11 mm Body [LFBGA]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



		Units	MILLIMETERS		
Dimension Limits			MIN	NOM	MAX
Number of Terminals	N		169		
Pitch	e		0.80 BSC		
Overall Height	A	1.33	1.445	1.56	
Standoff	A1	0.40	0.45	0.50	
Mold Thickness	A2	0.86 REF			
Substrate Thickness	A3	0.26 REF			
Overall Length	D	11.00 BSC			
Overall Terminal Spacing	D1	9.60 BSC			
Overall Width	E	11.00 BSC			
Overall Terminal Spacing	E1	9.60 BSC			
Terminal Diameter	b	0.40	0.45	0.50	

**Notes:**

- Pin A1 visual index feature may vary, but must be located within the hatched area.
- Dimensioning and tolerancing per ASME Y14.5M  
 BSC: Basic Dimension. Theoretically exact value shown without tolerances.  
 REF: Reference Dimension, usually without tolerance, for information purposes only.

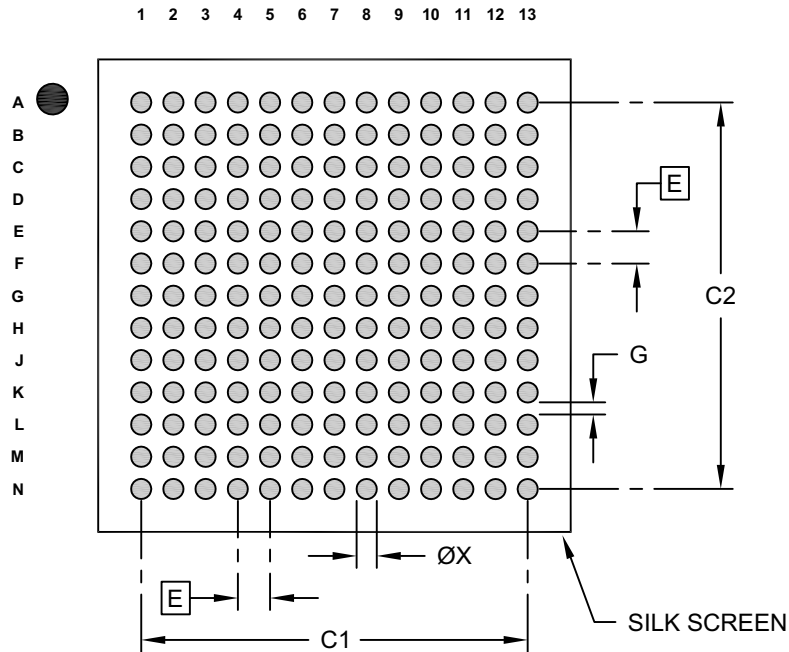
Microchip Technology Drawing C04-439A Sheet 2 of 2



# PIC32MZ グラフィック (DA) ファミリ

## 169-Ball Low Profile Ball Grid Array (6JX) - 11x11 mm Body [LFBGA]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Overall Contact Pad Spacing	C1		9.60	
Overall Contact Pad Spacing	C2		9.60	
Contact Pad Width (X169)	X1			0.50
Contact Pad to Contact Pad	G	0.30		

**Notes:**

1. Dimensioning and tolerancing per ASME Y14.5M

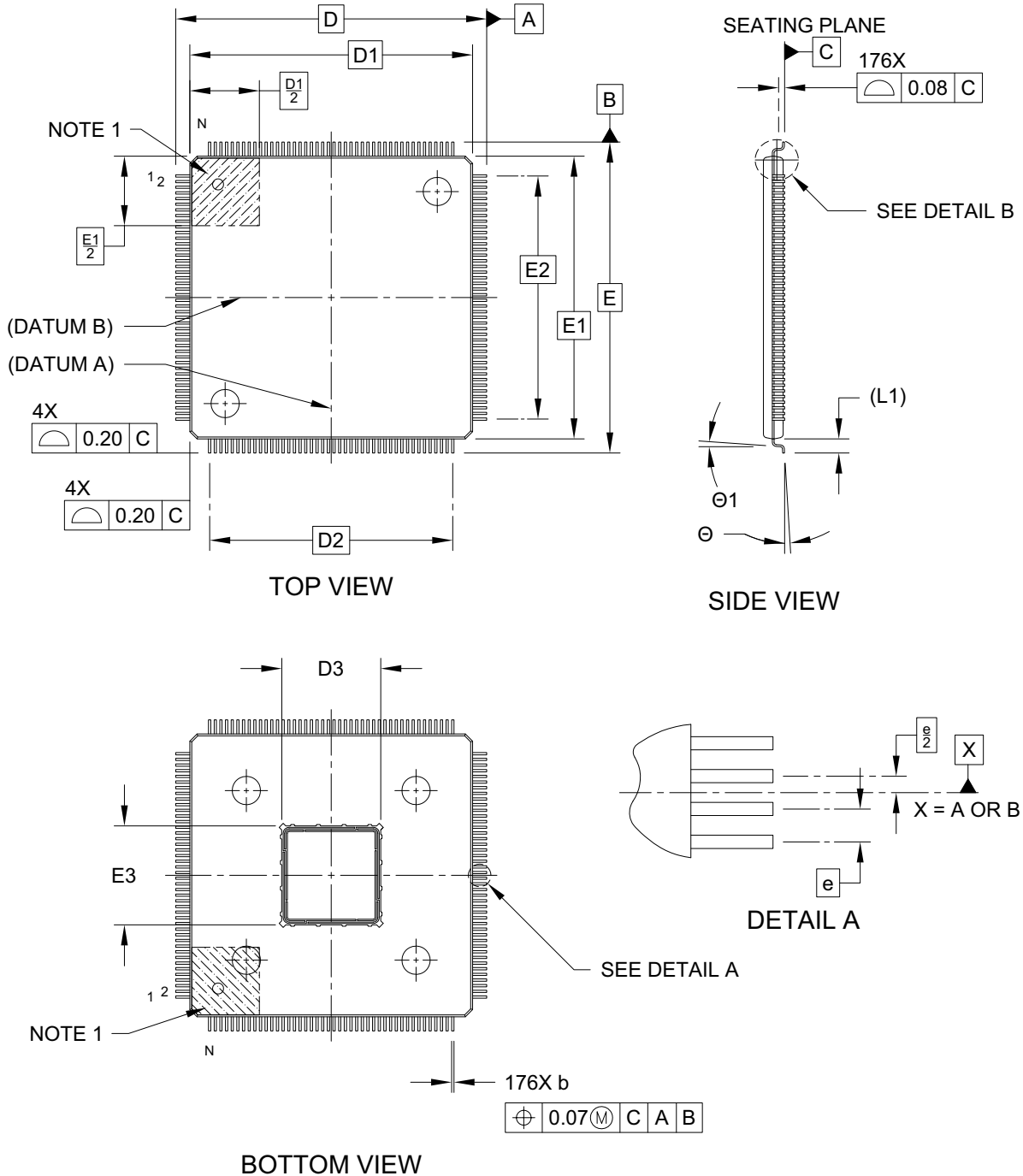
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2439A

# PIC32MZ グラフィック (DA) ファミリ

## 176-Lead Low Profile Quad Flat Pack (2J) - 20x20x1.4 mm Body [LQFP] With 7x7 mm Exposed Pad

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>

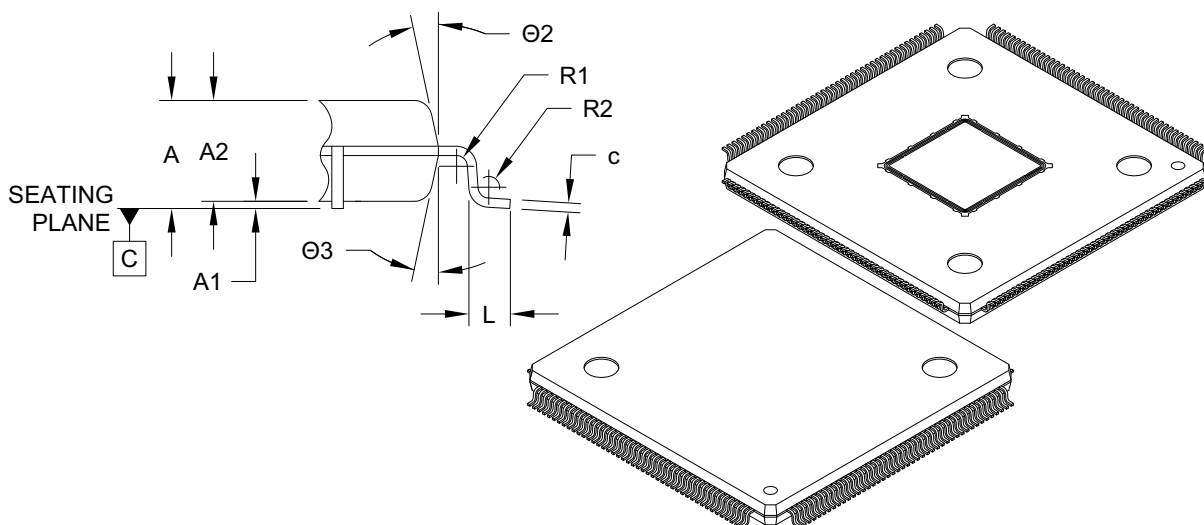


Microchip Technology Drawing C04-367A Sheet 1 of 2

# PIC32MZ グラフィック (DA) ファミリ

## 176-Lead Low Profile Quad Flat Pack (2J) - 20x20x1.4 mm Body [LQFP] With 7x7 mm Exposed Pad

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	176		
Pitch	e	0.40 BSC		
Overall Height	A	-	-	1.60
Standoff	A1	0.05	-	0.15
Molded Package Height	A2	1.35	1.40	1.45
Overall Length	D	22.00 BSC		
Molded Package Length	D1	20.00 BSC		
Overall Lead Pitch	D2	17.20 BSC		
Exposed Pad Length	D3	6.90	7.00	7.10
Overall Width	E	22.00 BSC		
Molded Package Width	E1	20.00 BSC		
Overall Lead Pitch	E2	17.20 BSC		
Exposed Pad Width	E3	6.90	7.00	7.10

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Lead Width	b	0.13	0.16	0.23
Lead Thickness	c	0.09	-	0.20
Lead Length	L	0.45	0.60	0.75
Footprint	(L1)	1.00 REF		
Bend Radius	R1	0.08	-	-
Bend Radius	R2	0.08	-	0.20
Foot Angle	Θ	0°	3.5°	7°
Lead Angle	Θ1	0°	-	-
Mold Draft Angle	Θ2	11°	12°	13°
Mold Draft Angle	Θ3	11°	12°	13°

**Notes:**

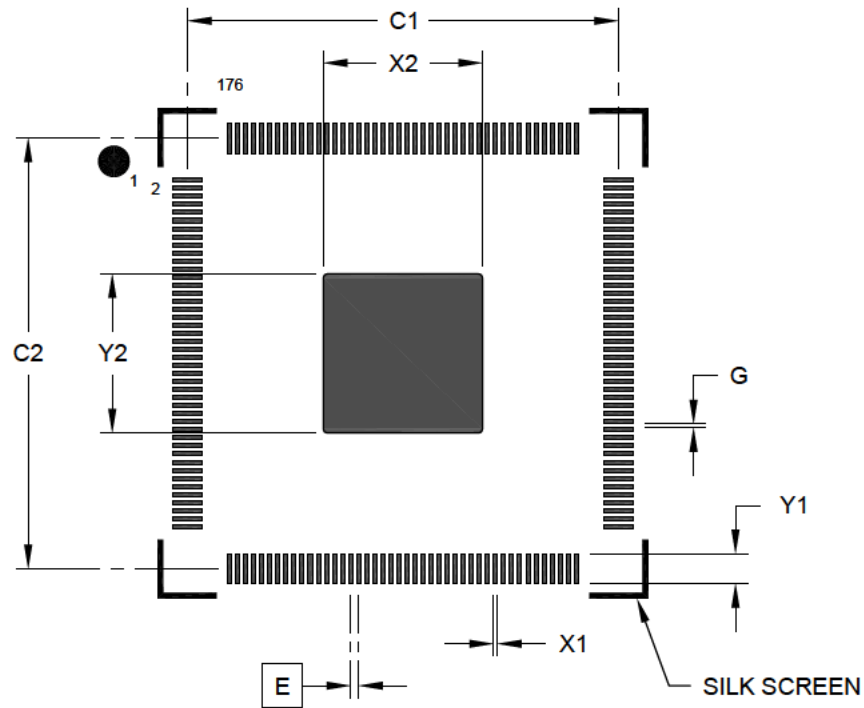
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Dimensions D1 and E1 do not include mold protrusion. Allowable Protrusion is 0.25mm per side. D1 and E1 are maximum body size dimensions including mold mismatch.
- Dimension b does not include dambar protrusion. Allowable dam bar protrusion shall not cause the lead width to exceed the maximum b dimension by more than 0.08mm. Dambar cannot be located on the lower radius or the foot. Minimum space between protrusion and adjacent lead is 0.07mm for 0.40mm pitch packages.
- Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.  
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-367A Sheet 2 of 2

# PIC32MZ グラフィック (DA) ファミリ

## 176-Lead Low Profile Quad Flat Pack (2J) - 20x20x1.4 mm Body [LQFP] With 7x7 mm Exposed Pad

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.40 BSC		
Contact Pad Spacing	C1		21.40	
Contact Pad Spacing	C2		21.40	
Contact Pad Width (X176)	X1			0.20
Contact Pad Length (X176)	Y1			1.50
Center Pad Width	X2			7.90
Center Pad Length	Y2			7.90
Contact Pad to Pad (X172)	G	0.20		

**Notes:**

1. Dimensioning and tolerancing per ASME Y14.5M

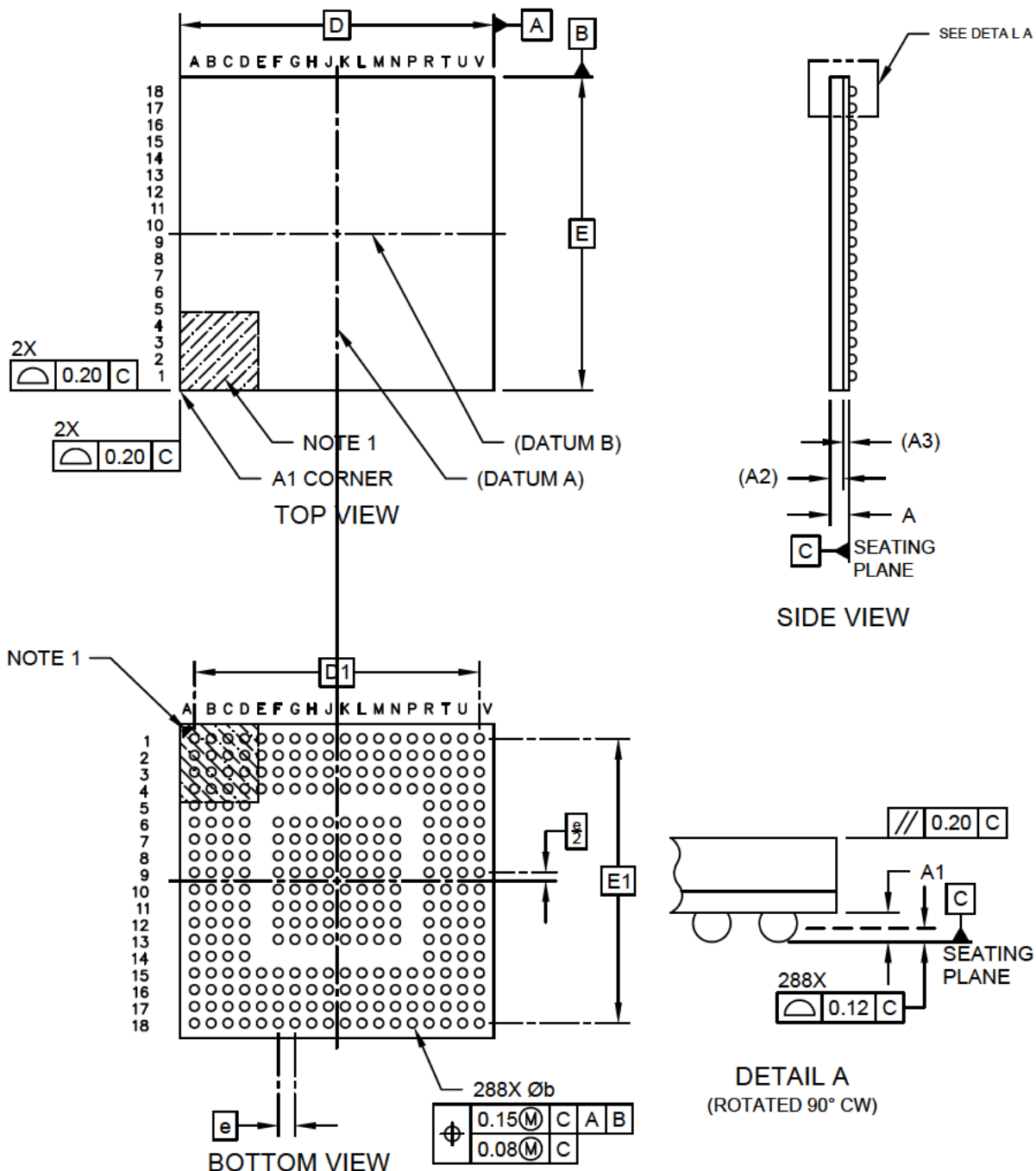
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2367B

# PIC32MZ グラフィック (DA) ファミリ

## 288 Ball Low Profile Fine Pitch Ball Grid Array (4J) - 15x15x1.4 mm Body [LFBGA]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>

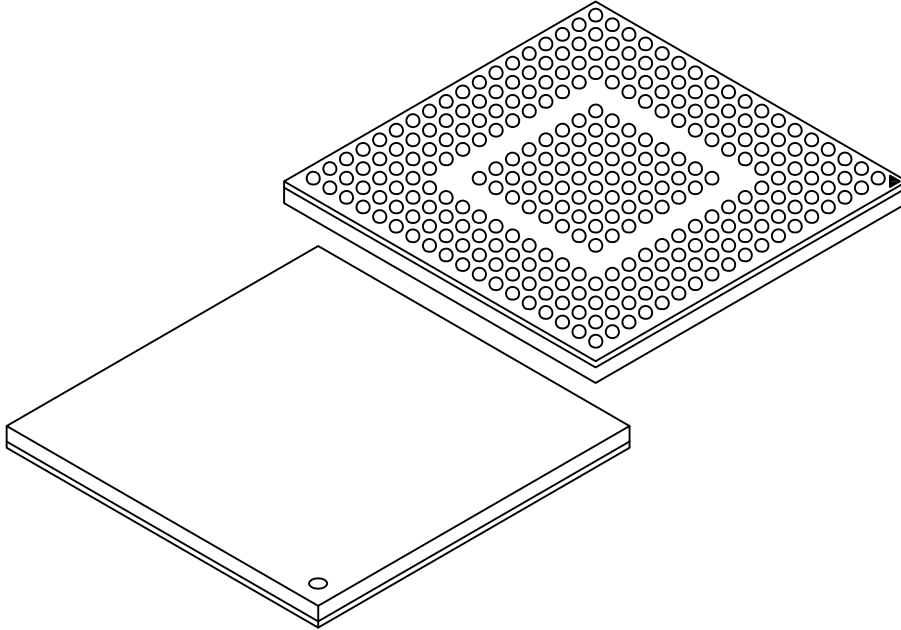


Microchip Technology Drawing C04-366B Sheet 1 of 2

# PIC32MZ グラフィック (DA) ファミリ

## 288 Ball Low Profile Fine Pitch Ball Grid Array (4J) - 15x15x1.4 mm Body [LFBGA]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Terminals (Balls)	N	288		
Pitch	e	0.80 BSC		
Overall Height	A	-	-	1.40
Terminal (Ball) Height	A1	0.30	0.35	0.40
Mold Cap Height	(A2)	0.70 REF		
Substrate Thickness	(A3)	0.26 REF		
Overall Length	D	15.00 BSC		
Overall Ball Pitch	D1	13.60 BSC		
Overall Width	E	15.00 BSC		
Overall Ball Pitch	E1	13.60 BSC		
Ball Diameter	b	0.40	0.45	0.50

**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

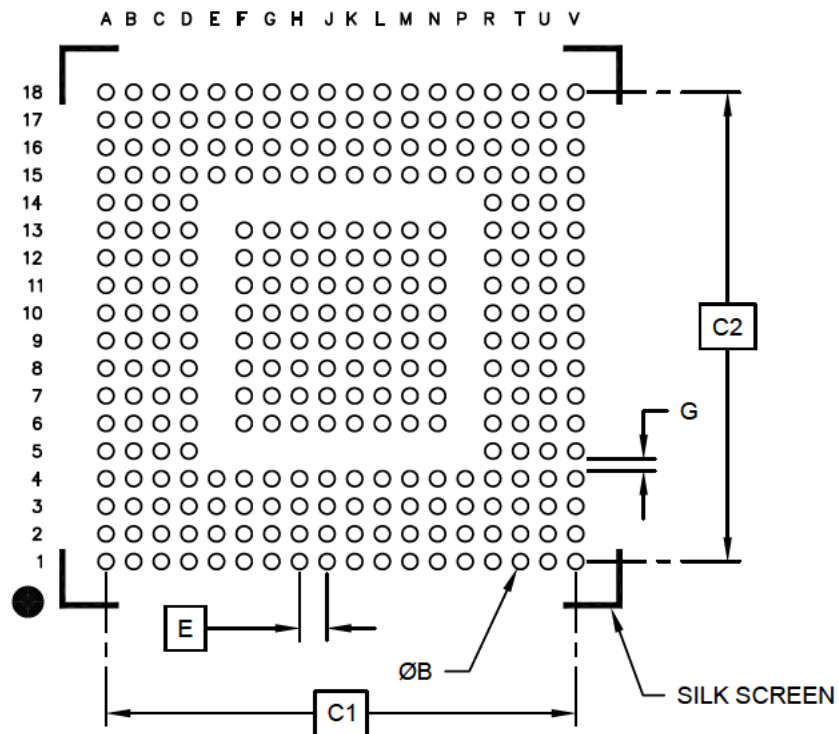
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-366B Sheet 2 of 2

# PIC32MZ グラフィック (DA) ファミリ

## 288 Ball Low Profile Fine Pitch Ball Grid Array (4J) - 15x15x1.4 mm Body [LFBGA]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Contact Pad Spacing	C1	13.60 BSC		
Contact Pad Spacing	C2	13.60 BSC		
Contact Pad Diameter (X288)	B	0.40	0.45	0.50
Pad-to-Pad Clearance	G	0.30		

**Notes:**

- Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2366B

# PIC32MZ グラフィック (DA) ファミリ

---

---



# PIC32MZ グラフィック (DA) ファミリ

## 補遺 A: 改訂履歴

### リビジョン A (2015 年 7 月)

本書の初版です。

### リビジョン B (2015 年 11 月)

本書のステータスを「Advance Information」から「Preliminary」に更新しました。

このリビジョンの主な変更点を表 A-1 に記載します。

加えて、文章と体裁の変更等、本書全体の細部を修正しました。

表 A-1: 主な変更点

項目タイトル	更新内容
XLP テクノロジ採用 32 ビット グラフィック アプリケーション MCU(最大 2 MB のライブアップデートフラッシュ、640 KB の SRAM、32 MB の DDR2 SDRAM 内蔵)	169 ピンデバイスのピン名を更新しました(表 5 参照)。 288 ピンデバイスのピン名を更新しました(表 7 参照)。
4.0 「メモリ構成」	ブートフラッシュ シーケンスとコンフィグレーションワードの一覧を更新しました(表 4-3 と表 4-4 参照)。 BFxSEQ3/ABFxSEQ3: ブートフラッシュ「x」シーケンスワード 0 レジスタを更新しました(レジスタ 4-1 参照)。
6.0 「リセット」	リセット レジスタマップ内の RCON レジスタの全てのリセット値を更新しました(表 6-1 参照)。
7.0 「CPU 例外と割り込みコントローラ」	割り込みレジスタマップに OFF199 レジスタを追加しました(表 7-3 参照)。
8.0 「オシレータ コンフィグレーション」	オシレータ レジスタマップ内の OSCON および PB6DIV レジスタの全てのリセット値を更新しました(表 8-2 参照)。 SPLLCON レジスタの PLLDIV<2:0> ビット値を更新しました(レジスタ 8-3 参照)。
10 「ダイレクトメモリアクセス(DMA)コントローラ」	DMA チャンネル 0 ~ 7 レジスタマップ内の全てのリセット値を更新しました(表 10-3 参照)。
11.0 「Hi-Speed USB with On-The-Go (OTG)」	USB レジスタマップ 1 内の USBOTG レジスタの bit 15:0 の全てのリセット値を更新しました(表 11-1 参照)。 USBCRCON レジスタの bit 24 と 13 の POR 時の値を更新しました(レジスタ 11-30)。
12.0 「I/O ポート」	PORTC レジスタマップの TRISC ビットを更新しました(表 12-5 参照)。 PORTH レジスタマップ内の ANSELH レジスタに ANSH3 ビットを追加しました(表 12-10 参照)。 ペリフェラルピンセレクト出力レジスタマップから RPD15R レジスタを削除しました(表 12-14 参照)。
18.0 「ウォッチドッグ タイマ(WDT)」	ウォッチドッグ レジスタマップ内の WDTCON レジスタの bit 15:0 の全てのリセット値を更新しました(表 18-1 参照)。
21.0 「シリアルペリフェラルインターフェイス(SPI)とI <sup>2</sup> C(Inter-IC Sound)インターフェイス」	ウォッチドッグ タイマ レジスタマップ内の SPI1STAT および SPI1CON2 レジスタの bit 15:0 の全ビットのリセット値を更新しました(表 21-1 参照)。
22.0 「SQI (Serial Quad Interface)」	SQI レジスタマップ内の SQI1XCON1 レジスタの bit 15:0 の全てのリセット値を更新しました(表 22-1 参照)。
25.0 「パラレル マスタポート(PMP)」	パラレル マスタポート レジスタマップ内の PMSTAT レジスタの bit 15:0 の全てのリセット値を更新しました(表 25-1 参照)。
26.0 「外部バス インターフェイス(EBI)」	EBI レジスタマップ内の全てのリセット値を更新しました(表 26-2 参照)。
29.0 「高速 12 ビット逐次比較型レジスタ(SAR) A/D コンバータ(ADC)」	ADC レジスタマップ内の ADCCON1 および ADCxTIME レジスタの全てのリセット値と ADCxCFG、ADCSYSCFGx、ADCDATAx レジスタの仮想アドレス値を更新しました(表 29-1 参照)。

# PIC32MZ グラフィック (DA) ファミリ

表 A-1: 主な変更点 (続き)

項目タイトル	更新内容
34.0 「HIGH/LOW 電圧検出 (HLVD)」	項目の表題を変更し、導入部を更新しました。 HLVDCON レジスタを更新しました (表 34-1 とレジスタ 34-1 参照)。 HIGH/LOW 電圧検出 (HLVD) モジュールのブロック図を更新しました (図 34-1 参照)。
36.0 「グラフィック LCD (GLCD) コントローラ」	グラフィック LCD コントローラ レジスタマップを更新しました (表 36-1 参照)。 以下のレジスタを更新しました。 <ul style="list-style-type: none"> <li>レジスタ 36-2: GLCDCLKCON: グラフィック LCD コントローラ クロック制御レジスタ</li> <li>レジスタ 36-4: GLCDRES: グラフィック LCD コントローラ分解能レジスタ</li> <li>レジスタ 36-5: GLCDFPORCH: グラフィック LCD コントローラ フロントポート レジスタ</li> <li>レジスタ 36-6: GLCDBLANKING: グラフィック LCD コントローラ ブランキング レジスタ</li> <li>レジスタ 36-7: GLCDBPORCH: グラフィック LCD コントローラ バックポート レジスタ</li> <li>レジスタ 36-8: GLCDCCURSOR: グラフィック LCD コントローラ カーソルレジスタ</li> <li>レジスタ 36-10: GLCDLxstart: グラフィック lcd コントローラ レイヤ「x」開始レジスタ (x = 0 ~ 2)</li> <li>レジスタ 36-11: GLCDLxsize: グラフィック lcd コントローラ レイヤ「x」サイズレジスタ (x = 0 ~ 2)</li> <li>レジスタ 36-14: GLCDLxres: グラフィック lcd コントローラ レイヤ「x」分解能レジスタ (x = 0 ~ 2)</li> </ul>
37.0 「2D グラフィック処理ユニット (GPU)」	導入部を更新しました。
39.0 「セキュア デジタルホストコントローラ (SDHC)」	SDHC のブロック図を更新しました (図 39-1 参照)。 SDHC レジスタマップを更新しました (表 39-1 参照)。 SDHCSTAT1 レジスタの CDSLVL ビットの値を更新しました (レジスタ 39-6 参照)。 SDHCCAP レジスタを更新しました (レジスタ 39-13)。
40.0 「省電力機能」	40.2.3 「ディープスリープモード」を更新しました。 PMD レジスタ一覧 (表 40-2) と PMD ビットとその位置 (表 40-3) 内の高電圧検出に関する記述を削除しました。
41.0 「その他の特殊機能」	CFGCON2 レジスタを更新しました (表 41-3 とレジスタ 41-12 参照)。
44.0 「電氣的特性」	以下の表を更新しました。 <ul style="list-style-type: none"> <li>表 44-1. 電圧に対する動作 MIPS</li> <li>表 44-3. パッケージ熱特性</li> <li>表 44-4. DC 温度 / 電圧仕様</li> <li>表 44-8. DC 特性: 動作電流 (I<sub>dd</sub>)</li> <li>表 44-9. DC 特性: アイドル電流 (I<sub>idle</sub>)</li> <li>表 44-10. DC 特性: パワーダウン電流 (I<sub>pd</sub>)</li> <li>表 44-12. DC 特性: I/O ピン出力仕様</li> <li>表 44-38. SPIx マスタモード (CKE = 0) タイミング要件</li> <li>表 44-39. SPIx モジュール マスタモード (CKE = 1) タイミング要件</li> <li>表 44-53. USB OTG 電氣的仕様</li> </ul>

# PIC32MZ グラフィック (DA) ファミリ

## リビジョン C (2016 年 10 月 /11 月)

本書全体を通して「VDD1V8」を以下の通りに変更しました。VDDR1V8 と VDD を VDDIO に変更しました。

本書全体を通して V-Temp 仕様を削除しました。

このリビジョンの主な変更点を表 A-2 に記載します。

加えて、文章と体裁の変更等、本書全体の細部を修正しました。

表 A-2: 主な変更点

セクション名	更新説明
<b>XLP テクノロジ採用 32 ビット グラフィック アプリケーション 向け MCU (最大 2 MB ライブ アップデートフラッシュ / 640 KB SRAM / 32 MB DDR2 SDRAM 内蔵)</b>	動作条件を 2.0 ~ 3.6 V から 2.2 ~ 3.6 V に変更しました。 全てのデバイスピン一覧表 (表 5 ~ 表 7) を更新しました。
<b>1.0 「デバイスの概要」</b>	Note 1 を Timer1 ~ Timer9 および RTCC ピン配置 I/O の説明に追加しました (表 1-7 参照)。 電源、グランド、参照電圧のピン配置 I/O の説明の Note 2 とピン番号を更新しました (表 1-23 参照)。
<b>2.0 「32 ビット マイクロコントローラを使うためのガイドライン」</b>	推奨最小接続図 (図 2-1) を更新しました。 <b>2.9.1.3 「EMI/EMC/EFT (IEC 61000-4-4、IEC 61000-4-2) を抑制するための配慮」</b> を追加しました。
<b>3.0 「CPU」</b>	コンフィグレーションレジスタ (CP0 レジスタ 16、セレクト 0) (レジスタ 3-1 参照) 内の SB ビットを更新しました。
<b>4.0 「メモリ構成」</b>	<b>4.3 「タイミングパラメータ」</b>
<b>6.0 「リセット」</b>	Note 1 をリセットレジスタマップ (表 6-1 参照) に追加しました。
<b>8.0 「オシレータの設定」</b>	CLKSTAT レジスタから DIVSPLLRDY ビットを削除しました (表 8-2、レジスタ 8-8 参照)。 bit 5-0 の中心周波数値を -2 ~ -4% から +2 ~ +4% に更新しました (レジスタ 8-2 参照)。
<b>12.0 「I/Oポート」</b>	ポートレジスタマップ内の CNCON レジスタを更新しました (表 12-3 ~ 表 12-12 参照)。 CNCONx レジスタから SIDL ビットを削除しました (レジスタ 12-3 参照)。
<b>20.0 「リアルタイムクロック / カレンダー (RTCC)」</b>	RTCC ピンに関連する Note を主な特長に追加しました。
<b>22.0 「SQI (SERIAL QUAD INTERFACE)」</b>	SQI モジュールのブロック図内の Note 1 を更新しました (図 22-1 参照)。
<b>26.0 「外部バス インターフェイス (EBI)」</b>	EBI モジュールを使う場合のグラフィック LCD (GLCD) コントローラに関する Note 2 を追加しました。 表 26-1: EBI モジュール機能を削除しました。 外部バス インターフェイス アドレスマスク レジスタ (レジスタ 26-2) から Note 1 を削除し、他の Note を追加しました。

# PIC32MZ グラフィック (DA) ファミリ

表 A-2: 主な変更点 (続き)

セクション名	更新説明
29.0 「高速 12ビット逐次比較型レジスタ (SAR) A/Dコンバータ (ADC)」	<p><b>式 29-1: “ADCのスループット レート”</b>            Note 1 を ADC レジスタマップに追加しました (表 29-1 参照)。            ADCCON1 レジスタおよび ADCxTIME レジスタの SELRES ビットに Note を追加しました (レジスタ 29-1、レジスタ 29-27 参照)。            ADC 制御レジスタ 1 に AICMPEN ビットを追加しました (表 29-1、レジスタ 29-1 参照)。            ADCFLTRx レジスタの CHNLID ビットの値と Note を更新しました (レジスタ 29-16 参照)。            ADCFSTAT レジスタの ADCID ビットのビット値を更新しました (レジスタ 29-22 参照)。            ADCxCFG レジスタの ADCCFG ビットの定義と Note を更新しました (レジスタ 29-33 参照)。</p>
34.0 「HIGH/LOW電圧検出 (HLVD)」	HLVDCON レジスタから SDIL ビットを削除しました (表 34-1、レジスタ 34-1 参照)。
36.0 「グラフィック LCD (GLCD) コントローラ」	GLCDLxMODE レジスタの FORCEALPHA ビットと DISABIFIL ビットの位置を入れ替えました (表 36-1、レジスタ 36-9 参照)。
38.0 「DDR2 SDRAM コントローラ」	<p>DDRPADCON レジスタを更新しました (表 38-1、レジスタ 38-28 参照)。            以下のレジスタの POR 時の値を更新しました。</p> <ul style="list-style-type: none"> <li>• レジスタ 38-18</li> <li>• レジスタ 38-25</li> <li>• レジスタ 38-26</li> <li>• レジスタ 38-28</li> </ul>
39.0 「セキュア デジタルホスト コントローラ (SDHC)」	セキュア デジタルホスト コントローラ (SDHC) のブロック図内の Note 1 を更新しました (図 39-1 参照)。
40.0 「省電力機能」	ディープスリープ制御レジスタから WAKEDIS ビットを削除しました (表 40-1、レジスタ 40-1 参照)。
41.0 「その他の特殊機能」	<p>デバイス シリアル番号レジスタの一覧に DEVS2 と DEVS3 を追加しました (表 41-4 参照)。            デバイス ADC 校正レジスタの一覧を更新しました (表 41-5 参照)。            CFGCON レジスタの JTAGEN ビットに Note2 を追加しました (レジスタ 41-9 参照)。</p>
44.0 「電気的特性」	<p>本書全体を通して、VDDIO の動作条件を 2.0 ~ 3.6 V から 2.2 ~ 3.6 V に変更し、VDDCORE の動作条件を 1.7 ~ 1.9 V に更新しました。            絶対最大定格を更新しました。            VDDIO 値を 0.8*VDDIO から 0.65*VDDIO に更新しました (表 44-10、表 44-15、表 44-22 参照)。            パッケージ熱特性を更新しました (表 44-3 参照)。            DC 特性値を更新しました (表 44-7 参照)。            SD ホスト コントローラ タイミング仕様の Min. 標準動作条件を更新しました (表 44-58、表 44-59 参照)。            全ての表を更新しました。</p>
46.0 「パッケージ情報」	<p>パッケージの寸法を更新しました (46.1 「パッケージのマーキング情報」参照)。            6JX 派ケージの情報を追加しました (46.1 「パッケージのマーキング情報」、46.2 「パッケージの詳細」参照)。</p>
製品識別システム	V-Temp デバイスのパッケージ マーキングを V に変更しました。

# PIC32MZ グラフィック (DA) ファミリ

## リビジョン D (2017 年 3 月)

このリビジョンの主な変更点を表 A-3 に記載します。

加えて、文章と体裁の変更等、本書全体の細部を修正しました。

表 A-3: 主な変更点

項目タイトル	更新内容
XLP テクノロジ採用 32 ビット グラフィック アプリケーション向け MCU (最大 2 MB ライブ アップデートフラッシュ / 640 KB SRAM / 32 MB DDR2 SDRAM 内蔵)	表 5 で、ピン B4 を VDDCORE に更新し、B6 を VDDIO に更新しました。
4.0 「メモリ構成」	図 4-1 で、KSEG3 を「cacheable」から「not cacheable」に変更しました。
6.0 「リセット」	図 6-1 を更新しました。
8.0 「オシレータの設定」	表 8-1 で、EBI に SYSCLK を追加しました。
26.0 「外部バス インターフェイス (EBI)」	図 26-1 で、PBCLK8 を SYSCLK に変更しました。
29.0 「高速 12 ビット逐次比較型レジスタ (SAR) A/D コンバータ (ADC)」	レジスタ 29-1 で、bit 12 の Note を更新し、表を追加しました。
38.0 「DDR2 SDRAM コントローラ」	表 38-1 で、レジスタ名を DRVSTRPFET と DRVSTRNFET の間で入れ替えました。 表 38-1 で、オフセットアドレス 9140 を追加しました。 レジスタ 38-28 で、DRVSTRPFET および DRVSTRNFET レジスタの名前と定義を入れ替えました。 レジスタ 38-30 を追加しました。
40.0 「省電力機能」	レジスタ 40-1 で、「command」を「instruction」に更新し、40.2.4 「VBATモード」を更新しました。
41.0 「その他の特殊機能」	レジスタ 41-5 で、「command」を「instruction」に更新しました。 レジスタ 41-9 で、bit 7 の Note を更新し、表を追加しました。
44.0 「電气的特性」	44.1 「DC特性」を更新しました。 表 44-4 と表 44-5 を更新しました。 表 44-18 で、パラメータ D306 を追加しました。 表 44-56 で、パラメータ DDR10、DDR19、DDR22、DDR23 の値を更新しました。

# PIC32MZ グラフィック (DA) ファミリ

## リビジョン E (2017 年 5 月)

このリビジョンの主な変更点を表 A-4 に記載します。

加えて、文章と体裁の変更等、本書全体の細部を修正しました。

表 A-4: 主な変更点

項目タイトル	更新内容
XLP テクノロジ採用 32 ビット グラフィック アプリケーション 向け MCU (最大 2 MB ライブ アップデートフラッシュ / 640 KB SRAM / 32 MB DDR2 SDRAM 内蔵)	表 6 で、ピン 168 の値を「CVREFOUT/AN5/RPB10/RB10」から「AN5/RPB10/RB10」へ更新しました。
25.0 「パラレルマスタ ポート (PMP)」	表 25-1 で、仮想アドレス列のヘッダ行内の値を BF80 ~ BF82 に更新しました。仮想アドレスを 70xx から E0xx に更新しました。
36.0 「グラフィック LCD (GLCD) コントローラ」	主な特長の中の分解能を更新しました。
39.0 「セキュア デジタルホスト コントローラ (SDHC)」	主な特長に「eMMC 規格 : JESD84-A441」を追加しました。
44.0 「電気的特性」	表 44-7、表 44-8、表 44-9、表 44-10、表 44-11、表 44-16、表 44-18 内の各種 DC 特性パラメータを更新しました。 表 44-27、表 44-28、表 44-29 内の各種 AC 特性パラメータを更新しました。
補遺 A: 「改訂履歴」	リビジョン C の導入部で最初に現れる VDDR1V8 を VDD1V8 に更新しました。

# PIC32MZ グラフィック (DA) ファミリ

## 索引

### A

AC 特性 .....	735
A/D 変換タイミング要件 .....	761
ADC 仕様 .....	760
EBI タイミング要件 .....	771
EJTAG タイミング要件 .....	774
Ethernet .....	768
GLCD コントローラ タイミング仕様 .....	771
MPLL クロック タイミング要件 .....	738
PLL クロック タイミング .....	737
SD ホスト コントローラ ハイスピード モード .....	773
SD ホスト コントローラ 既定値モード .....	773
USB OTG 電氣的仕様 .....	767
内部 BFRC 精度 .....	739
内部 LPRC 精度 .....	739
内部 RC 精度 .....	739
パラレルスレーブ ポート要件 .....	764
パラレルマスタ ポート書き込み要件 .....	766
パラレルマスタ ポート読み出し要件 .....	765

### B

Block Diagrams	
Prefetch Module .....	173

### C

CAN( コントローラ エリア ネットワーク ) .....	483
CPU	
EJTAG デバッグサポート .....	50
アーキテクチャの概要 .....	47
管理 .....	50
コア例外タイプ .....	124
コプロセッサ 0 レジスタ .....	48
CPU モジュール .....	39, 45
Customer Notification Service .....	805
C コンパイラ	
MPLAB C18 .....	714

### D

DC 特性 .....	718
I/O ピン入力仕様 .....	724
I/O ピン入力注入電流仕様 .....	728
I/O ピン出力仕様 .....	725
アイドル電流 (idle) .....	722
温度 / 電圧仕様 .....	719
省電力時の電流 (I <sub>pd</sub> ) .....	723
プログラム フラッシュメモリ 待機ステート .....	730
プログラムメモリ .....	730
DDR2 SDRAM コントローラ .....	605
I/O 仕様 .....	729

### E

EBI ( 外部バスインターフェイス ) .....	391
Ethernet コントローラ .....	521

### G

GLCD コントローラ	
DC 仕様 .....	734

### I

I/O ピン機能 ( 一覧 ) .....	18, 19, 21, 25, 26, 27, 29, 30, 31, 32, 33, 34, 36, 38
I/O ポート .....	251
パラレル I/O (PIO) .....	252
読み書きタイミング .....	252
I <sup>2</sup> C (Inter-Integrated Circuit) .....	361

### M

Microchip 社のウェブサイト .....	805
MPLAB ASM30 アセンブラ、リンカ、ライブラリアン .....	714
MPLAB PM3 デバイス プログラマ .....	715
MPLAB REAL ICE インサーキット エミュレータ システム .....	715
MPLAB SIM ソフトウェア シミュレータ .....	715
MPLAB 統合開発環境ソフトウェア .....	713
MPLINK オブジェクト リンカ /MPLIB オブジェクト ライブラリアン .....	714

### O

OTG (On-The-Go) 対応のハイスピード USB .....	201
-------------------------------------	-----

### S

SD ホスト コントローラ	
I/O 仕様 .....	729
SPI (Serial Peripheral Interface) .....	323
SQI (Serial Quad Interface) .....	333

### T

Timer1 モジュール .....	279
Timer2/3, Timer4/5, Timer6/7, Timer8/9 モジュール .....	283

### U

UART .....	369
USB インターフェイス ブロック図 .....	202

### W

WWW アドレス .....	805
WWW オンラインサポート .....	15

### あ

アセンブラ	
MPASM アセンブラ .....	714
アナログ ポートピンの設定 .....	252
暗号エンジン .....	399
SA_CTRL のフォーマット .....	418
セキュリティ アソシエーション構造 .....	415
バッファ ディスクリプタ .....	410
インターネット アドレス .....	805
エラッタ .....	15
オシレータの設定 .....	157

### か

改訂履歴 .....	779
開発サポート .....	713
外部クロック	
Timer1 タイミング要件 .....	744
Timer2 ~ 9 タイミング要件 .....	745
タイミング要件 .....	736
カスタマサポート .....	805
グラフィック LCD (GLCD) コントローラ .....	583
高電圧検出 (HVD) .....	117
顧客変更通知サービス .....	805
コンパレータ	
仕様 .....	732
コンパレータ モジュール .....	565
コンパレータ参照電圧	
仕様 .....	733
コンパレータ参照電圧 (CV <sub>ref</sub> ) .....	569
コンフィグレーション ビット .....	683

### さ

出力コンペア .....	293
省電力モード SFR の一覧 .....	672
省電力機能 .....	669

# PIC32MZ グラフィック (DA) ファミリ

CPU 実行中 .....	669	DDR2 SDRAM コントローラ .....	605
その他の特殊機能 .....	683	DMA .....	177
<b>た</b>		Ethernet コントローラ .....	521
タイミシング仕様		High/Low 電圧検出 (HLVD).....	573
CAN I/O 要件 .....	759	I2C (Inter-Integrated Circuit).....	362
I2Cx バスデータ要件 (スレーブモード).....	757	JTAG プログラミング、デバッグ、トレースポート .....	710
I2Cx バスデータ要件 (マスタモード).....	755	PMP ピン配置と外部デバイスへの接続 .....	377
SPIx スレーブモード (CKE = 1) 要件 .....	753	RTCC .....	313
SPIx スレーブモード要件 (CKE = 0) .....	751	SPI モジュール .....	323
SPIx マスタモード (CKE = 0) 要件 .....	748	SQL.....	333
SPIx マスタモード (CKE = 1) 要件 .....	750	Timer1 .....	279
出力コンペア要件 .....	746	Timer2/3/4/5 (16 ビット).....	283
単純 OCx/PWM モード要件 .....	746	UART .....	369
入力キャプチャ要件 .....	745	WDT とパワーアップ タイマ .....	307, 311
<b>タイミシング図</b>		暗号エンジン .....	399
CAN I/O .....	759	グラフィック LCD (GLCD) コントローラ .....	584
EJTAG .....	774	コンパレータ I/O 動作モード .....	565
I/O 特性 .....	740	コンパレータ参照電圧モジュール .....	569
I2Cx バス START/STOP ビット (スレーブモード).....	757	出力コンペアモジュール .....	293
I2Cx バス START/STOP ビット (マスタモード).....	755	代表的な多重化ポート構造 .....	251
I2Cx バスデータ (スレーブモード).....	757	入力キャプチャ .....	289
I2Cx バスデータ (マスタモード).....	755	プリフェッチモジュール .....	173
OCx/PWM .....	746	乱数生成器 (RNG) .....	421
SPIx スレーブモード (CKE = 0) .....	751	リセットシステム .....	115
SPIx スレーブモード (CKE = 1) .....	752	割り込みコントローラ .....	123
SPIx マスタモード (CKE = 0) .....	747		
SPIx マスタモード (CKE = 1).....	749	<b>ま</b>	
Timer1 ~ 9 外部クロック .....	744	命令セット .....	711
UART 受信 .....	376	メモリマップ	
UART 送信 (8 ビットまたは 9 ビットデータ).....	376	2048 KB プログラムメモリのデバイス .....	56
外部クロック .....	736	メモリ構成 .....	55
出力コンペア (OCx).....	746	レイアウト .....	55
入力キャプチャ (CAPx).....	745		
パラレルスレーブポート .....	764	<b>ら</b>	
パラレルマスタポート書き込み .....	766	乱数生成器 (RNG) .....	421
パラレルマスタポート読み出し .....	765	リアルタイムクロック / カレンダー (RTCC).....	313
<b>タイミシング要件</b>		レジスタ	
I/O .....	740	AD1CON1 (A/D 制御 1) .....	322
ダイレクトメモリアクセス (DMA) コントローラ .....	177	AD1CON1 (ADC 制御 1).....	322
電圧レギュレータ (内蔵) .....	710	ADCANCON (ADC アナログウォームアップ制御レジスタ .....	479
電気的特性 .....	717	).....	479
AC .....	735	ADCBASE (ADC ベース) .....	472
High/Low-Voltage Detect .....	720	ADCCMP1CON (ADC デジタルコンパレータ 1 制御).....	466
		ADCCMPENx (ADC デジタルコンパレータ「x」イネーブル .....	459
<b>な</b>		レジスタ (x = 1 ~ 6)) .....	459
入力状態変化通知 .....	252	ADCCMPx (ADC デジタルコンパレータ「x」制限値レジ .....	460
		スタ (x = 1 ~ 6)) .....	460
<b>は</b>		ADCCMPxCON (ADC デジタルコンパレータ「x」制御レ .....	468
パッケージ .....	785	ジスタ (x = 1 ~ 9)) .....	468
マーキング .....	785	ADCCON1 (ADC 制御レジスタ 1).....	437
詳細 .....	786	ADCCON2 (ADC 制御レジスタ 2).....	440
パラレルマスタポート (PMP).....	377	ADCCON3 (ADC 制御レジスタ 3).....	442
パワーオンリセット (POR) と内蔵電圧レギュレータ .....	710	ADCCSS1 (ADC 共通スキャン選択レジスタ 1).....	456
		ADCCSS2 (ADC 共通スキャン選択レジスタ 2).....	457
ピン配置 I/O の説明 (表) .....	20, 24, 28, 32	ADCDATAx (ADC 出力データレジスタ (x = 0 ~ 44)).....	473
ブラウンアウトリセット (BOR) と内蔵電圧レギュレータ .....	710	ADCDCSTAT1 (ADC データレディステータスレジスタ 1) .....	458
		.....	458
フラッシュプログラムメモリ .....	105, 115	ADCDCSTAT2 (ADC データレディステータスレジスタ 2) .....	458
RTSP .....	105	.....	458
プリフェッチモジュール .....	173	ADCEIEN1 (ADC 早期割り込みイネーブルレジスタ 1) .....	476
ブロック図		.....	476
2D グラフィック処理ユニット .....	603	ADCEIEN2 (ADC 早期割り込みイネーブルレジスタ 2).....	476
CAN モジュール .....	483	ADCEIEN2 (ADC 早期割り込みステータスレジスタ 2).....	478
CPU .....	46	.....	478
CTMU コンフィグレーション		ADCFLTRx (ADC デジタルフィルタ「x」レジスタ (x = 1 ~ .....	461
時間計測 .....	577	6)) .....	461
		ADCGIRQEN1 (ADC 割り込みイネーブルレジスタ 1).....	455



# PIC32MZ グラフィック (DA) ファミリ

ADCIMCON1 (ADC 入力モード制御レジスタ 1).....	447	.....	615
ADCIMCON2 (ADC 入力モード制御レジスタ 2).....	450	DDRMEMCFG1 (DDR メモリ コンフィグレーション 1)....	616
ADCIMCON3 (ADC 入力モード制御レジスタ 3).....	453	.....	616
ADCIRQEN2 (ADC 割り込みイネーブルレジスタ 2)....	455	DDRMEMCFG2 (DDR メモリ コンフィグレーション 2)....	617
ADCSYSCFG1 (ADC システム コンフィグレーションレジスタ 1).....	482	.....	617
ADCSYSCFG2 (ADC システム コンフィグレーションレジスタ 2).....	482	DDRMEMCFG3 (DDR メモリ コンフィグレーション 3)....	618
ADCTRG1 (ADC トリガ源 1).....	463	.....	619
ADCTRG2 (ADC トリガ源 2 レジスタ).....	464	DDRMEMCON (DDR メモリ制御レジスタ).....	614
ADCTRG3 (ADC トリガ源 3 レジスタ).....	465	DDRMINCMD (DDR 最小コマンド数).....	613
ADCTRGMODE (専用 ADC トリガモード).....	445	DDRMINLIM (DDR 最小バースト数制限).....	611
ADCTRGSNS (ADC トリガ レベル / エッジ検出).....	474	DDRODTCFG (DDR オンダイ ターミネーション コンフィグレーション).....	626
ADCxCFG (ADCx コンフィグレーションレジスタ「x」(x = 1 ~ 4, 7)).....	481	DDRODTENCFG (DDR オンダイ ターミネーションイネーブル コンフィグレーション).....	629, 630
ADCxTIME (専用 ADCx タイミングレジスタ「x」(x = 0 ~ 4)).....	475	DDRPHYDLLR (DDR PHY DLL 再校正).....	639
ALRMDATE (アラーム日付値).....	322	DDRPYPADCON (DDR PHY パッド制御).....	637
ALRMDATECLR (アラーム日付クリア).....	322	DDRPWRCFG (DDR 省電力モード コンフィグレーション).....	621
ALRMDATESET (アラーム日付設定).....	322	.....	621
ALRMTIME (アラーム時刻値).....	321	DDRREFCFG (DDR リフレッシュ コンフィグレーション).....	620
ALRMTIMECLR (アラーム時刻クリア).....	322	.....	620
ALRMTIMEINV (アラーム時刻反転).....	322	DDRRQPER (DDR 要求期間).....	612
ALRMTIMESET (アラーム時刻設定).....	322	DDRSCLCFG0 (DDR SCL コンフィグレーション 0)....	635
ALFGCON (コンフィグレーション制御).....	700	DDRSCLCFG1 (DDR SCL コンフィグレーション 1)....	636
CFGCON2 (コンフィグレーション制御 2).....	706	DDRTSEL (DDR ターゲット選択).....	610
CFGBIA (外部バス インターフェイス アドレスピン コンフィグレーション).....	702	DDRXFERCFG (DDR 転送コンフィグレーション).....	627
CFGBIC (外部バス インターフェイス制御ピン コンフィグレーション).....	703	DEVCFG0/ADEVCFG0 (デバイス コンフィグレーションワード 0).....	689
CFGPLL (メモリ PLL コンフィグレーション).....	707	DEVCFG1/ADEVCFG1 (デバイス コンフィグレーションワード 1).....	691
CHECON (キャッシュ制御).....	176	DEVCFG2/ADEVCFG2 (デバイス コンフィグレーションワード 2).....	694
CMSTAT (コンパレータ ステータス).....	568	DEVCFG3/ADEVCFG2 (デバイス コンフィグレーションワード 3).....	697
CMxCON (コンパレータ x 制御).....	567	DEVCFG4/ADEVCFG4 (デバイス コンフィグレーションワード 4).....	698
CNCONx (PORTx 向け状態変更通知制御).....	278	DEVCP0/ADEVCP0 (デバイスコード保護 0).....	688
CONFIG (CP0 レジスタ 16、Select 0).....	51	DEVID (デバイスおよびリビジョン ID).....	62, 709
CONFIG1 (CP0 レジスタ 16、Select 1).....	52	DEVSIGN0/ADEVSIGN0 (デバイス シグネチャワード 0).....	688
CONFIG2 (CP0 レジスタ 16、Select 7).....	54	.....	688
CONFIG2 (CP0 レジスタ 16、Select 5).....	54	DMAADDR (DMA アドレス).....	186
CONFIG3 (CP0 レジスタ 16、Select 3).....	53	DMAADDR (DMR アドレス).....	186
CTMUCON (CTMU 制御).....	579	DMACON (DMA コントローラ制御).....	185
CVRCON (コンパレータ参照電圧制御).....	571	DMASTAT (DMA ステータス).....	186
DCHxCON (DMA チャンネル x 制御).....	190	DMSTAT (デッドマンタイマステータス).....	303
DCHxCPTR (DMA チャンネル x セルポインタ).....	198	DMTCLR (デッドマンタイムクリア).....	302
DCHxCSIZ (DMA チャンネル x セルサイズ).....	198	DMTCNT (デッドマンタイムカウンタ).....	304
DCHxDAT (DMA チャンネル x パターンデータ).....	199	DMTCON (デッドマンタイム制御).....	301
DCHxDPTR (チャンネル x デスティネーションポインタ).....	197	DMTPRECLR (デッドマンタイムプリスケアラ).....	301
DCHxDSA (DMA チャンネル x デスティネーション開始アドレス).....	195	DSCON (ディーブスリープ制御).....	675
DCHxDSIZ (DMA チャンネル x デスティネーションサイズ).....	196	EBICSx (外部バス インターフェイス チップセレクト).....	393, 396
DCHxECON (DMA チャンネル x イベント制御).....	192	.....	396
DCHxINT (DMA チャンネル x 割り込み制御).....	193	EBIMSKx (外部バス インターフェイス アドレスマスク).....	394
DCHxSPTR (DMA チャンネル x ソースポインタ).....	197	.....	394
DCHxSSA (DMA チャンネル x ソース開始アドレス).....	195	EBISMCON (外部バス インターフェイス 静的メモリ制御).....	397
DCHxSSIZ (DMA チャンネル x ソースサイズ).....	196	.....	397
DCRCCON (DMA CRC 制御).....	187	EBISMTx (外部バス インターフェイス 静的メモリ タイミング).....	395
DCRCDATA (DMA CRC データ).....	189	.....	395
DCRCXOR (DMA CRCXOR イネーブル).....	189	EMAC1CFG1 (Ethernet コントローラ MAC コンフィグレーション 1).....	548
DDRCMD1x (DDR ホストコマンド 1 レジスタ「x」).....	631	.....	548
DDRCMD2x (DDR ホストコマンド 2「x」).....	632	EMAC1CFG2 (Ethernet コントローラ MAC コンフィグレーション 2).....	549
DDRCMDISSUE (DDR コマンド発行).....	628	.....	549
DDRDLYCFG0 (DDR 遅延コンフィグレーション 0).....	622	EMAC1CLRT (Ethernet コントローラ MAC コリジョンウィンドウ / リトライ リミット).....	553
DDRDLYCFG1 (DDR 遅延コンフィグレーション 1).....	623	.....	553
DDRDLYCFG2 (DDR 遅延コンフィグレーション 2).....	624	EMAC1IPGR (Ethernet コントローラ MAC 非連続パケット間ギャップ).....	552
DDRDLYCFG3 (DDR 遅延コンフィグレーション 3).....	625	.....	552
DDRMEMCFG0 (DDR メモリ コンフィグレーション 0).....	615	EMAC1IPGT (Ethernet コントローラ MAC 連続パケット間	.....

# PIC32MZ グラフィック (DA) ファミリ

ギャップ).....	551	ク制御).....	589
EMAC1MADR (Ethernet コントローラ MAC MIIM アドレス).....	559	GLCDCCLUTx(グラフィック LCD コントローラ グローバル色変換ルックアップ テーブル).....	600
EMAC1MAXF (Ethernet コントローラ MAC 最大フレーム長).....	554	GLCDCCURSOR (グラフィック LCD コントローラ カーソル).....	592
EMAC1MCFG (Ethernet コントローラ MAC MIIM コンフィグレーション).....	557	GLCDFPDRCH (グラフィック LCD コントローラ フロントポート).....	591
EMAC1MCMD (Ethernet コントローラ MAC MIIM コマンド).....	558	GLCDINT (グラフィック LCD コントローラ カーソル LUT).....	602
EMAC1MIND (Ethernet コントローラ MAC MIIM インジケータ).....	561	GLCDINT (グラフィック LCD コントローラ カーソルデータ).....	601
EMAC1MRDD (Ethernet コントローラ MAC MIIM 読み出しデータ).....	560	GLCDINT (グラフィック LCD コントローラ 割り込み).....	598
EMAC1MWTD (Ethernet コントローラ MAC MIIM 書き込みデータ).....	560	GLCDLxBADDR (グラフィック LCD コントローラ レイヤ「X」 ベースアドレス).....	596
EMAC1SA0 (Ethernet コントローラ MAC ステーションアドレス 0).....	562	GLCDLxMODE (グラフィック LCD コントローラ レイヤ「x」 モード).....	593
EMAC1SA1 (Ethernet コントローラ MAC ステーションアドレス 1).....	563	GLCDLxRES (グラフィック LCD コントローラ レイヤ「X」 分解能).....	597
EMAC1SA2 (Ethernet コントローラ MAC ステーションアドレス 2).....	564	GLCDLxSIZE (グラフィック LCD コントローラ レイヤ「x」 サイズ).....	595
EMAC1SUPP (Ethernet コントローラ MAC PHY サポート).....	555	GLCDLxSTART (グラフィック LCD コントローラ レイヤ「X」 開始).....	595
EMAC1TEST (Ethernet コントローラ MAC テスト).....	556	GLCDLxSTRIDE (グラフィック LCD コントローラ レイヤ「X」 ストライド).....	596
ETHALGNERR (Ethernet コントローラ アラインメントエラー統計).....	547	GLCDRES (グラフィック LCD コントローラ 分解能).....	590
ETHCON1 (Ethernet コントローラ 制御 1).....	526	GLCDSTAT (グラフィック LCD コントローラ ステータス).....	599
ETHCON2 (Ethernet コントローラ 制御 2).....	528	GLDCMODE (グラフィック LCD コントローラ モード).....	587
ETHFCSERR (Ethernet コントローラ フレームチェックシーケンス エラー統計).....	546	I2CxCON (I2C 制御).....	365
ETHFRMRXOK (Ethernet コントローラ フレーム受信成功統計).....	545	I2CxSTAT (I2C ステータス).....	367
ETHFRMTXOK (Ethernet コントローラ フレーム送信成功統計).....	542	ICxCON (入力キャプチャ x 制御).....	292
ETHHT0 (Ethernet コントローラ ハッシュテーブル 0).....	530	IFSx (割り込みフラグステータス).....	153
ETHHT1 (Ethernet コントローラ ハッシュテーブル 1).....	530	INTCON (割り込み制御).....	149
ETHIEN (Ethernet コントローラ 割り込みイネーブル).....	536	INTSTAT (割り込みステータス).....	152
ETHIRQ (Ethernet コントローラ 割り込み要求).....	537	IPC0 (割り込み優先度制御 0).....	154
ETHMCOLFRM (Ethernet コントローラ 複数コリジョンフレーム統計).....	544	IPTMR (割り込み近接タイマ).....	152
ETHPM0 (Ethernet コントローラ パターンマッチ オフセット).....	532	NVMADDR (フラッシュ アドレス).....	109
ETHPMCS (Ethernet コントローラ パターンマッチ チェックサム).....	532	NVMBWP (フラッシュブート (ページ) 書き込み保護).....	112
ETHPMM0 (Ethernet コントローラ パターンマッチ マスク 0).....	531	NVMCON (プログラミング制御).....	107
ETHPMM1 (Ethernet コントローラ パターンマッチ マスク 1).....	531	NVMCON2 (プログラミング制御レジスタ 2).....	114
ETHRXFC (Ethernet コントローラ 受信フィルタ コンフィグレーション).....	533	NVMDATA (フラッシュデータ).....	110
ETHRXOVFLOW (Ethernet コントローラ 受信オーバーフロー統計).....	541	NVMKEY (プログラミング ロック解除).....	109
ETHRXST (Ethernet コントローラ RX パケット ディスクリプタ開始アドレス).....	529	NVMPWP (プログラム フラッシュ書き込み保護).....	111
ETHRXWM (Ethernet コントローラ 受信ウォーターマーク).....	535	NVMSRCADDR (ソースデータ アドレス).....	110
ETHSCOLFRM (Ethernet コントローラ 単一コリジョンフレーム統計).....	543	OCxCON (出力コンペア x 制御).....	297
ETHSTAT (Ethernet コントローラ ステータス).....	539	OSCCON (オシレータ制御).....	163
ETHTXST (Ethernet コントローラ TX パケット ディスクリプタ開始アドレス).....	529	OSCTUN (FRC 調整).....	165
GLCDBGCOLOR (グラフィック LCD コントローラ 背景色).....	590	PMADDR (パラレルポート アドレス).....	383
GLCDBLANKING (グラフィック LCD コントローラ ブランキング).....	591	PMAEN (パラレルポート ピンイネーブル).....	386
GLCDBPORCH (グラフィック LCD コントローラ バックポート).....	592	PMCON (パラレルポート制御).....	379
GLCDCCLKCON (グラフィック LCD コントローラ クロ		PMDIN (パラレルポート 入力データ).....	385, 390
		PMDOUT (パラレルポート 出力データ).....	384
		PMODE (パラレルポート モード).....	381
		PMRADDR (パラレルポート 読み出しアドレス).....	389
		PMSTAT (パラレルポート ステータス (スリープモード専用)).....	387
		PMWADDR (パラレルポート 書き込みアドレス).....	388
		PRECON (プリフェッチ モジュール制御).....	175
		PRIS (優先度シャドール選択).....	150
		PSCNT (ポストステータス DMT カウント設定ステータス).....	304
		PSINTV (ポストステータス DMT インターバル設定ステータス レジスタ).....	305
		REFOxCON (参照オシレータ制御 (x = 1 ~ 4)).....	168
		REFOxTRIM (参照オシレータ調整 (x = 1 ~ 4)).....	169
		RPhR (ペリフェラル ビンセレクト出力).....	277

# PIC32MZ グラフィック (DA) ファミリ

RSWRST (ソフトウェアリセット).....	119, 120, 122	USBTMCON1 (USB タイミング制御 1).....	244
RTCALRM (RTC アラーム制御).....	317	USBTMCON2 (USB タイミング制御 2).....	244
RTCCON (RTCC 制御).....	315	WDTCON (ウォッチドッグタイマ制御).....	309
RTCDATE (RTC 日付値).....	320	レジスタマップ	
RTCTIME (RTC 時刻値).....	319	[pin name]R (ペリフェラルピンセレクト入力).....	277
SBFLAG (システムバスステータスフラグ).....	93, 94, 95, 96	DMA CRC.....	178
SBTxECLRM (システムバスターゲット「x」複数エラー クリア).....	100	DMA グローバル.....	178
SBTxECLRS (システムバスターゲット「x」単一エラー クリア).....	100	DMA チャンネル 0 ~ 7.....	179
SBTxECON (システムバスターゲット「x」エラー制御).....	99	EBI.....	392
SBTxELOG1 (システムバスターゲット「x」エラーログ 1) .....	97	HIGH/LOW 電圧検出.....	574
SBTxELOG2 (システムバスターゲット「x」エラーログ 2) .....	99	I2C1 ~ I2C5.....	363
SBTxRDy (システムバスターゲット「x」領域「y」読み 出しパーミッション).....	102	PORTA.....	260
SBTxREGy (システムバスターゲット「x」領域「y」).....	101	PORTB.....	261
SBTxWRy (システムバスターゲット「x」領域「y」書き 込みパーミッション).....	103	PORTH.....	267
SDHCBLKCON (SDHC ブロック制御).....	644	PORTJ.....	268
SDHCSTAT2 (SDHC ステータス 2).....	661	PORTK.....	269
SPIxCON (SPI 制御).....	326	RTCC.....	314
SPIxCON2 (SPI 制御 2).....	329	SPI1 ~ SPI6.....	324
SPIxSTAT (SPI ステータス).....	330	Timer1.....	280
SPLLCON (システム PLL 制御).....	166	Timer2 ~ Timer9.....	285
SQI1XCON1 (SQI XIP 制御 1).....	336	UART1-5.....	370
SQI1XCON2 (SQI XIP 制御レジスタ 2).....	338	USB.....	203, 209
T1CON (タイプ A タイマ制御).....	281	ウォッチドッグタイマ.....	308
TxCON (タイプ B タイマ制御).....	287	オシレータ設定.....	161
USBCRCON (USB クロック / リセット制御).....	249	コンパレータ.....	566
USBCSR0 (USB 制御ステータス 0).....	210	コンパレータ参照電圧.....	570
USBCSR1 (USB 制御ステータス 1).....	212	システムバス.....	71
USBCSR2 (USB 制御ステータス 2).....	213	システムバスターゲット 0.....	72
USBCSR3 (USB 制御ステータス 3).....	215	システムバスターゲット 1.....	89, 90, 91
USBDMAINT (USB DMA 割り込み).....	240	システムバスターゲット 10.....	85
USBDMAxA (USB DMA チャンネル「x」メモリアドレス) .....	242	システムバスターゲット 11.....	86
USBDMAxC (USB DMA チャンネル「x」制御).....	241	システムバスターゲット 12.....	87
USBDMAxN (USB DMA チャンネル「x」カウント).....	242	システムバスターゲット 13.....	88
USBDPBFD (USB ダブルパケットバッファディセーブル) .....	243	システムバスターゲット 2.....	75
USBEOFRST (USB EOF / ソフトリセット制御).....	237	システムバスターゲット 3.....	76
USBExRPC (USB エンドポイント「x」要求パケットカウ ント (ホストモード専用)).....	243	システムバスターゲット 4.....	77
USBExRXA (USB エンドポイント「x」受信アドレス).....	239	システムバスターゲット 5.....	79
USBExTXA (USB エンドポイント「x」送信アドレス).....	238	システムバスターゲット 6.....	81
USBFIFOA (USB FIFO アドレス).....	234	システムバスターゲット 7.....	82
USBHWVER (USB ハードウェアバージョン).....	235	システムバスターゲット 8.....	83
USBIE0CSR0 (USB インデックス付きエンドポイント制御 ステータス 0 (エンドポイント 0)).....	217	システムバスターゲット 9.....	84
USBIE0CSR2 (USB インデックス付きエンドポイント制御 ステータス 2 (エンドポイント 0)).....	219	システム制御.....	116
USBIE0CSR3 (USB インデックス付きエンドポイント制御 ステータス 3 (エンドポイント 0)).....	220	出力コンペア 1 ~ 9.....	295
USBIECSR0 (USB インデックス付きエンドポイント制御 ステータス 0 (エンドポイント 1 ~ 7)).....	221	代替デバイスコンフィグレーションワード.....	685
USBIECSR1 (USB インデックス付きエンドポイント制御 ステータス 1 (エンドポイント 1 ~ 7)).....	224	デッドマンタイマ.....	300
USBIECSR2 (USB インデックス付きエンドポイント制御 ステータス 2 (エンドポイント 1 ~ 7)).....	227	デバイスコンフィグレーションワード.....	684
USBIECSR3 (USB インデックス付きエンドポイント制御 ステータス 3 (エンドポイント 1 ~ 7)).....	228	デバイス ADC 校正.....	687
USBINFO (USB 情報).....	236	デバイスシリアル番号.....	686
USBLPMR1 (USB リンク電力管理制御 1).....	245	ニュー入力キャプチャ 1 ~ 9.....	291
USBLPMR2 (USB リンク電力管理制御 2).....	247	パラレルマスタポート.....	378
		フラッシュコントローラ.....	106
		プリフェッチ.....	174
		ペリフェラルピンセレクト入力.....	270
		ペリフェラルピンセレクト出力.....	274
		割り込み.....	134
		わ	
		割り込みコントローラ	
		IRG、ベクタとビットの位置.....	126

# PIC32MZ グラフィック (DA) ファミリ

---

NOTE:

# PIC32MZ グラフィック (DA) ファミリ

---

## Microchip 社のウェブサイト

Microchip 社はウェブサイト ([www.microchip.com](http://www.microchip.com)) でオンラインサポートを提供しています。このウェブサイトを通じて、お客様はファイルと情報を簡単に入手できます。インターネット ブラウザから以下の内容がご覧になれます。

- **製品サポート** - データシートとエラッタ、アプリケーションノートとサンプル プログラム、設計リソース、ユーザガイドとハードウェア サポート文書、最新のソフトウェアと過去のソフトウェア
- **一般的技術サポート** - よく寄せられる質問 (FAQ)、技術サポートのご依頼、オンライン ディスカッショングループ、Microchip 社のコンサルタント プログラムおよびメンバーリスト
- **ご注文とお問い合わせ** - 製品セレクトと注文ガイド、最新プレスリリース、セミナー / イベントの一覧、お問い合わせ先 (営業所 / 販売代理店) の一覧

## 顧客変更通知サービス

Microchip 社のお客様向け変更通知サービスは、お客様に Microchip 社製品の最新情報をお届けするサービスです。ご興味のある製品ファミリまたは開発ツールに関する変更、更新、リビジョン、エラッタ情報をいち早くメールにてお知らせします。

Microchip 社ウェブサイト ([www.microchip.com](http://www.microchip.com)) にアクセスし、[DESIGN SUPPORT] メニューの下の [Product Change Notification] からご登録ください。

## カスタマサポート

Microchip 社製品をお使いのお客様は、以下のチャンネルからサポートをご利用になれます。

- 販売代理店または販売担当者
- 各地の営業所
- フィールド アプリケーション エンジニア (FAE)
- 技術サポート

サポートは販売代理店までお問い合わせください。各地の営業所もご利用になれます。本書の最後のページには各国の営業所の一覧を記載しています。

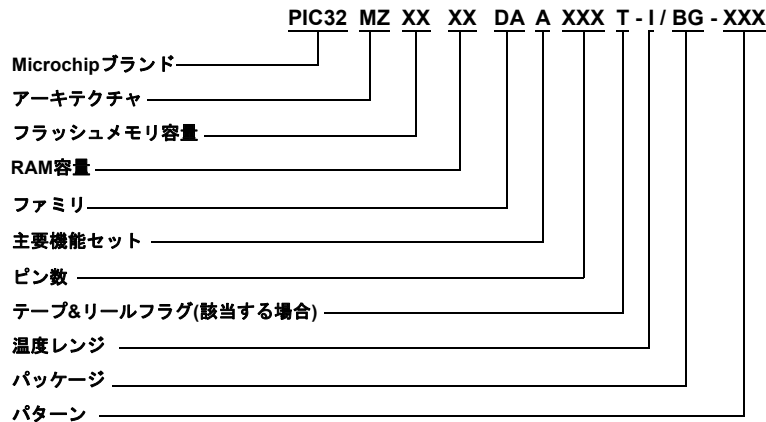
技術サポートは以下のウェブページからもご利用になれます。

<http://microchip.com/support>

# PIC32MZ グラフィック (DA) ファミリ

## 製品識別システム

ご注文や製品の価格、納期につきましては弊社または販売代理店にお問い合わせください。



例:

PIC32MZ0512DAA176-I/2J:  
グラフィック MCU ファミリ、PIC32、  
MIPS32<sup>®</sup> microAptiv<sup>™</sup> MPU コア、  
512 KB プログラムメモリ  
176 ピン、産業用温度レンジ  
LQFP パッケージ

### フラッシュメモリ ファミリ

アーキテクチャ MZ = MIPS32<sup>®</sup> microAptiv<sup>™</sup> MPU コア

フラッシュメモリ容量 10 = 1024 KB  
20 = 2048 KB

RAM 容量 25 = 256 KB  
64 = 640 KB

ファミリ DA = グラフィック MCU ファミリ

主な機能 A = PIC32 DA ファミリ機能、暗号エンジンなし、DDR メモリなし  
B = PIC32 DA ファミリ機能、暗号エンジンあり、DDR メモリなし  
G = PIC32 DA ファミリ機能、暗号エンジンなし、DDR メモリあり  
H = PIC32 DA ファミリ機能、暗号エンジンあり、DDR メモリあり

ピン数 169 = 169 ピン  
176 = 176 ピン  
288 = 288 ピン

温度レンジ I = -40 ~ +85 °C ( 産業用温度レンジ )

パッケージ HF = 169 ピン (11x11x1.4 mm) LFBGA (Low Profile Fine Pitch Ball Grid Array)  
6J = 169 ピン (11x11x1.56 mm) LFBGA (Low Profile Fine Pitch Ball Grid Array)  
2J = 176 ピン (22x22x1.4 mm) LQFP (Low Profile Quad Flat Pack)  
4J = 288 ピン (15x15x1.4 mm) LFBGA (Low Profile Fine Pitch Ball Grid Array)

パターン 3 桁の QTP、SQTP、その他のコード等 ( または空白のまま )  
ES = エンジニアリング サンプル

---

---

**Microchip 社製デバイスのコード保護機能に関して次の点にご注意ください。**

- Microchip 社製品は、該当する Microchip 社データシートに記載の仕様を満たしています。
- Microchip 社では、通常の条件ならびに仕様に従って使用した場合、Microchip 社製品のセキュリティ レベルは、現在市場に流通している同種製品の中でも最も高度であると考えています。
- しかし、コード保護機能を解除するための不正かつ違法な方法が存在する事もまた事実です。弊社の理解ではこうした手法は、Microchip 社データシートにある動作仕様書以外の方法で Microchip 社製品を使用する事になります。このような行為は知的財産権の侵害に該当する可能性が非常に高いと言えます。
- Microchip 社は、コードの保全性に懸念を抱くお客様と連携し、対応策に取り組んでいきます。
- Microchip 社を含む全ての半導体メーカーで、自社のコードのセキュリティを完全に保証できる企業はありません。コード保護機能とは、Microchip 社が製品を「解読不能」として保証するものではありません。

**コード保護機能は常に進歩しています。Microchip 社では、常に製品のコード保護機能の改善に取り組んでいます。Microchip 社のコード保護機能の侵害は、デジタル ミレニアム著作権法に違反します。そのような行為によってソフトウェアまたはその他の著**

---

本書に記載されているデバイス アプリケーション等に関する情報は、ユーザの便宜のためにのみ提供されているものであり、更新によって無効とされる事があります。お客様のアプリケーションが仕様を満たす事を保証する責任は、お客様にあります。Microchip 社は、明示的、暗黙的、書面、口頭、法定のいずれであるかを問わず、本書に記載されている情報に関して、状態、品質、性能、商品性、特定目的への適合性をはじめとする、いかなる類の表明も保証も行いません。Microchip 社は、本書の情報およびその使用に起因する一切の責任を否認します。Microchip 社の明示的な書面による承認なしに、生命維持装置あるいは生命安全用途に Microchip 社の製品を使用する事は全て購入者のリスクとし、また購入者はこれによって発生したあらゆる損害、クレーム、訴訟、費用に関して、Microchip 社は擁護され、免責され、損害をうけない事に同意するものとします。暗黙的あるいは明示的を問わず、Microchip 社が知的財産権を保有しているライセンスは一切譲渡されません。

#### 商標

Microchip 社の名称と Microchip ロゴ、dsPIC、FlashFlex、KEELOQ、KEELOQ ロゴ、MPLAB、PIC、PICmicro、PICSTART、PIC<sup>32</sup> ロゴ、rfPIC、SST、SST ロゴ、SuperFlash、UNI/O は、米国およびその他の国における Microchip Technology Incorporated の登録商標です。

FilterLab、Hampshire、HI-TECH C、Linear Active Thermistor、MTP、SEEVAL、Embedded Control Solutions Company は、米国における Microchip Technology Incorporated の登録商標です。

Silicon Storage Technology は、その他の国における Microchip Technology Incorporated の登録商標です。

Analog-for-the-Digital Age、Application Maestro、BodyCom、chipKIT、chipKIT ロゴ、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPF、MPLAB 認証ロゴ、MPLIB、MPLINK、mTouch、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICkit、PICtail、REAL ICE、rfLAB、Select Mode、SQL、Serial Quad I/O、Total Endurance、TSHARC、UniWinDriver、WiperLock、ZENA、Z-Scale は、米国およびその他の国における Microchip Technology Incorporated の登録商標です。

SQTP は、米国における Microchip Technology Incorporated のサービスマークです。

GestIC と ULPP は、その他の国における Microchip Technology Germany II GmbH & Co. & KG (Microchip Technology Incorporated の子会社) の登録商標です。

その他、本書に記載されている商標は各社に帰属します。

©2013, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-5224-2058-3

**QUALITY MANAGEMENT SYSTEM  
CERTIFIED BY DNV  
= ISO/TS 16949 =**

Microchip 社では、Chandler および Tempe (アリゾナ州)、Gresham (オレゴン州)の本部、設計部およびウェハ製造工場そしてカリフォルニア州とインドのデザインセンターが ISO/TS-16949:2009 認証を取得しています。Microchip 社の品質システム プロセスおよび手順は、PIC<sup>®</sup> MCU および dsPIC<sup>®</sup> DSC、KEELOQ<sup>®</sup> コード ホッピング デバイス、シリアル EEPROM、マイクロペリフェラル、不揮発性メモリ、アナログ製品に採用されています。さらに、開発システムの設計と製造に関する Microchip 社の品質システムは ISO 9001:2000 認証を取得しています。

## 各国の営業所とサービス

### 北米

#### 本社

2355 West Chandler Blvd.

Chandler, AZ 85224-6199

Tel:480-792-7200

Fax:480-792-7277

技術サポート:

<http://www.microchip.com/support>

URL:

[www.microchip.com](http://www.microchip.com)

#### アトランタ

Duluth, GA

Tel:678-957-9614

Fax:678-957-1455

#### オースティン (TX)

Tel:512-257-3370

#### ボストン

Westborough, MA

Tel:774-760-0087

Fax:774-760-0088

#### シカゴ

Itasca, IL

Tel:630-285-0071

Fax:630-285-0075

#### クリーブランド

Independence, OH

Tel:216-447-0464

Fax:216-447-0643

#### ダラス

Addison, TX

Tel:972-818-7423

Fax:972-818-2924

#### デトロイト

Novi, MI

Tel:248-848-4000

#### ヒューストン (TX)

Tel:281-894-5983

#### インディアナポリス

Noblesville, IN

Tel:317-773-8323

Fax:317-773-5453

#### ロサンゼルス

Mission Viejo, CA

Tel:949-462-9523

Fax:949-462-9608

#### ニューヨーク (NY)

Tel:631-435-6000

#### サンノゼ (CA)

Tel:408-735-9110

#### カナダ - トロント

Tel:905-673-0699

Fax:905-673-6509

### アジア/太平洋

#### アジア太平洋支社

Suites 3707-14, 37th Floor

Tower 6, The Gateway

Harbour City, Kowloon

Hong Kong

Tel:852-2943-5100

Fax:852-2401-3431

#### オーストラリア - シドニー

Tel:61-2-9868-6733

Fax:61-2-9868-6755

#### 中国 - 北京

Tel:86-10-8569-7000

Fax:86-10-8528-2104

#### 中国 - 成都

Tel:86-28-8665-5511

Fax:86-28-8665-7889

#### 中国 - 重慶

Tel:86-23-8980-9588

Fax:86-23-8980-9500

#### 中国 - 東莞

Tel:86-769-8702-9880

#### 中国 - 杭州

Tel:86-571-8792-8115

Fax:86-571-8792-8116

#### 中国 - 香港 SAR

Tel:852-2943-5100

Fax:852-2401-3431

#### 中国 - 南京

Tel:86-25-8473-2460

Fax:86-25-8473-2470

#### 中国 - 青島

Tel:86-532-8502-7355

Fax:86-532-8502-7205

#### 中国 - 上海

Tel:86-21-5407-5533

Fax:86-21-5407-5066

#### 中国 - 瀋陽

Tel:86-24-2334-2829

Fax:86-24-2334-2393

#### 中国 - 深圳

Tel:86-755-8864-2200

Fax:86-755-8203-1760

#### 中国 - 武漢

Tel:86-27-5980-5300

Fax:86-27-5980-5118

#### 中国 - 西安

Tel:86-29-8833-7252

Fax:86-29-8833-7256

### アジア/太平洋

#### 中国 - 厦門

Tel:86-592-2388138

Fax:86-592-2388130

#### 中国 - 珠海

Tel:86-756-3210040

Fax:86-756-3210049

#### インド - バンガロール

Tel:91-80-3090-4444

Fax:91-80-3090-4123

#### インド - ニューデリー

Tel:91-11-4160-8631

Fax:91-11-4160-8632

#### インド - プネ

Tel:91-20-3019-1500

#### 日本 - 大阪

Tel:81-6-6152-7160

Fax:81-6-6152-9310

#### 日本 - 東京

Tel:81-3-6880-3770

Fax:81-3-6880-3771

#### 韓国 - 大邱

Tel:82-53-744-4301

Fax:82-53-744-4302

#### 韓国 - ソウル

Tel:82-2-554-7200

Fax:82-2-558-5932 または

82-2-558-5934

#### マレーシア - クアラルンプール

Tel:60-3-6201-9857

Fax:60-3-6201-9859

#### マレーシア - ペナン

Tel:60-4-227-8870

Fax:60-4-227-4068

#### フィリピン - マニラ

Tel:63-2-634-9065

Fax:63-2-634-9069

#### シンガポール

Tel:65-6334-8870

Fax:65-6334-8850

#### 台湾 - 新竹

Tel:886-3-5778-366

Fax:886-3-5770-955

#### 台湾 - 高雄

Tel:886-7-213-7828

#### 台湾 - 台北

Tel:886-2-2508-8600

Fax:886-2-2508-0102

#### タイ - バンコク

Tel:66-2-694-1351

Fax:66-2-694-1350

### ヨーロッパ

#### オーストリア - ヴェルス

Tel:43-7242-2244-39

Fax:43-7242-2244-393

#### デンマーク - コペンハーゲン

Tel:45-4450-2828

Fax:45-4485-2829

#### フランス - パリ

Tel:33-1-69-53-63-20

Fax:33-1-69-30-90-79

#### ドイツ - デュッセルドルフ

Tel:49-2129-3766400

#### ドイツ - ミュンヘン

Tel:49-89-627-144-0

Fax:49-89-627-144-44

#### ドイツ - プフォルツハイム

Tel:49-7231-424750

#### イタリア - ミラノ

Tel:39-0331-742611

Fax:39-0331-466781

#### イタリア - ベニス

Tel:39-049-7625286

#### オランダ - ドリュエネン

Tel:31-416-690399

Fax:31-416-690340

#### ポーランド - ワルシャワ

Tel:48-22-3325737

#### スペイン - マドリッド

Tel:34-91-708-08-90

Fax:34-91-708-08-91

#### スウェーデン - ストックホルム

Tel:46-8-5090-4654

#### イギリス - ウォーキングム

Tel:44-118-921-5800

Fax:44-118-921-5820