Microchip Technology - PIC32MZ2025DAB169-I/HF Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

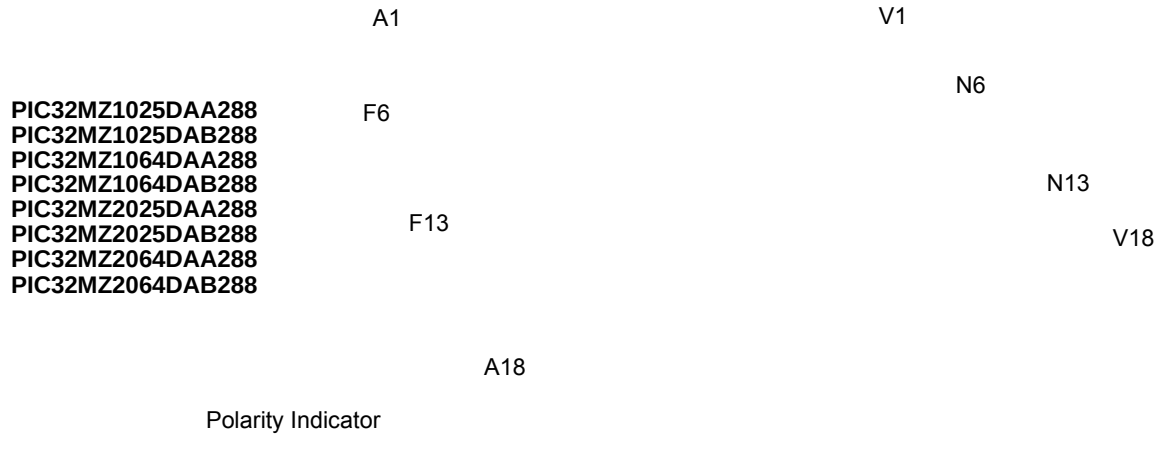
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	MIPS32® microAptiv™
Core Size	32-Bit Single-Core
Speed	200MHz
Connectivity	CANbus, EBI/EMI, Ethernet, I²C, IrDA, LINbus, PMP, SPI, SQI, UART/USART, USB OTG
Peripherals	Brown-out Detect/Reset, DMA, HLVD, I²S, POR, PWM, WDT
Number of I/O	120
Program Memory Size	2MB (2M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256K x 8
Voltage - Supply (Vcc/Vdd)	1.7V ~ 3.6V
Data Converters	A/D 45x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	169-LFBGA
Supplier Device Package	169-LFBGA (11x11)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic32mz2025dab169-i-hf

TABLE 7: PIN NAMES FOR 288-PIN DEVICES (CONTINUED)

288-PIN LFBGA (BOTTOM VIEW)



PIC32MZ1025DAA288
PIC32MZ1025DAB288
PIC32MZ1064DAA288
PIC32MZ1064DAB288
PIC32MZ2025DAA288
PIC32MZ2025DAB288
PIC32MZ2064DAA288
PIC32MZ2064DAB288

Ball/Pin Number	Full Pin Name
V3	DDRA15
V4	VDDCORE
V5	RTCC/RPD0/RD0
V6	SCK4/RD10
V7	GD6/EBIA11/RPF0/PMA11/RF0
V8	GD21/EBIA23/RH15
V9	GD3/EBIA8/RPG0/PMA8/RG0
V10	EBID2/PMD2/RE2
V11	ERXDV/ECRSDV/RH13
V12	ERXD3/RH9
V13	ETXD2/RH0
V14	ETXD0/RJ8
V15	ETXERR/RJ0
V16	ETXEN/RPD6/RD6
V17	GD1/EBID14/PMD14/RA4
V18	No Connect

Note 1: The RPn pins can be used by remappable peripherals. See Table 1 and Table 4 for the available peripherals and **12.4 "Peripheral Pin Select (PPS)"** for restrictions.

2: Every I/O port pin (RAx-RKx) can be used as a change notification pin (CNAx-CNKx). See **12.0 "I/O Ports"** for more information.

3: Shaded pins are 5V tolerant.

4: This pin must be tied to Vss through a 20k Ω resistor when DDR is not connected in the system.

5: This pin is a No Connect when DDR is not connected in the system.

6: These pins are restricted to input functions only.

Table of Contents

TABLE 1-24: JTAG, TRACE, AND PROGRAMMING/DEBUGGING PINOUT I/O DESCRIPTIONS

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	169-pin LFBGA	176-pin LQFP	288-pin LFBGA			
JTAG						
TCK	E11	160	H16	I	ST	JTAG Test Clock Input Pin
TDI	A6	28	A10	I	ST	JTAG Test Data Input Pin
TDO	C6	27	A11	O	—	JTAG Test Data Output Pin
TMS	D2	53	D4	I	ST	JTAG Test Mode Select Pin
Trace						
TRCLK	E4	54	E4	O	—	Trace Clock
TRD0	E2	64	H4	O	—	Trace Data bits 0-3
TRD1	E3	56	G4	O	—	
TRD2	E1	65	J4	O	—	
TRD3	D1	55	F4	O	—	
Programming/Debugging						
PGED1	C12	169	D18	I/O	ST	Data I/O pin for Programming/Debugging Communication Channel 1
PGEC1	B9	11	A14	I	ST	Clock input pin for Programming/Debugging Communication Channel 1
PGED2	D12	170	D17	I/O	ST	Data I/O pin for Programming/Debugging Communication Channel 2
PGEC2	D7	13	B14	I	ST	Clock input pin for Programming/Debugging Communication Channel 2
$\overline{\text{MCLR}}$	K1	85	R5	I/P	ST	Master Clear (Reset) input. This pin is an active-low Reset to the device.

Legend: CMOS = CMOS-compatible input or output; Analog = Analog input; P = Power
ST = Schmitt Trigger input with CMOS levels; O = Output; I = Input
TTL = Transistor-transistor Logic input buffer; PPS = Peripheral Pin Select

TABLE 4-4: BOOT FLASH 2 SEQUENCE AND CONFIGURATION WORDS SUMMARY

Virtual Address (BFC6_#)	Register Name	Bit Range	Bits															All Resets
			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:** See Table 41-2 for the bit descriptions.																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>																xxxx
		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:** See Table 41-1 for the bit descriptions.																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>																xxxx
		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

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

TABLE 7-3: INTERRUPT REGISTER MAP (CONTINUED)

Virtual Address (BF81_#)	Register Name(1)	Bit Range	Bits																All Resets
			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

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: All registers in this table with the exception of the OFFx registers, have corresponding CLR, SET, and INV registers at their virtual addresses, plus offsets of 0x4, 0x8 and 0xC, respectively. See **Section 12.2 "CLR, SET, and INV Registers"** for more information.

2: This bit is only available on devices with a Crypto module.

REGISTER 7-5: IFSx: INTERRUPT FLAG STATUS REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	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

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31-0 **IFS31-IFS0:** Interrupt Flag Status bits
1 = Interrupt request has occurred
0 = No interrupt request has occurred

Note: This register represents a generic definition of the IFSx register. Refer to Table 7-2 for the exact bit definitions.

REGISTER 7-6: IECx: INTERRUPT ENABLE CONTROL REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	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

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31-0 **IEC31-IEC0:** Interrupt Enable bits
1 = Interrupt is enabled
0 = Interrupt is disabled

Note: This register represents a generic definition of the IECx register. Refer to Table 7-2 for the exact bit definitions.

TABLE 12-13: PERIPHERAL PIN SELECT INPUT REGISTER MAP

Virtual Address (BF80_#)	Register Name	Bit Range	Bits																All Resets
			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

Legend: x = unknown value on Reset; — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

REGISTER 22-23: SQI1TAPCON: SQI TAP CONTROL REGISTER (CONTINUED)

bit 13-8 **SDRCLKINDLY<5:0>:** SQI Clock Input Delay in SDR Mode bits

These bits are used to add fractional delays to SQI Clock Input while sampling the incoming data in DDR mode.

111111 = 64 taps added on clock input
111110 = 63 taps added on clock input

•
•
•

000001 = 2 taps added on clock input
000000 = 1 tap added on clock input

bit 7-4 **DATAOUTDLY<3:0>:** SQI Data Output Delay bits

These bits are used to add fractional delays to SQI Data Output while writing the data to the Flash in all modes of operation.

1111 = 16 taps added on data output
1110 = 15 taps added on data output

•
•
•

0001 = 2 taps added on data output
0000 = 1 tap added on data output

bit 3-0 **CLKOUTDLY<3:0>:** SQI Clock Output Delay bits

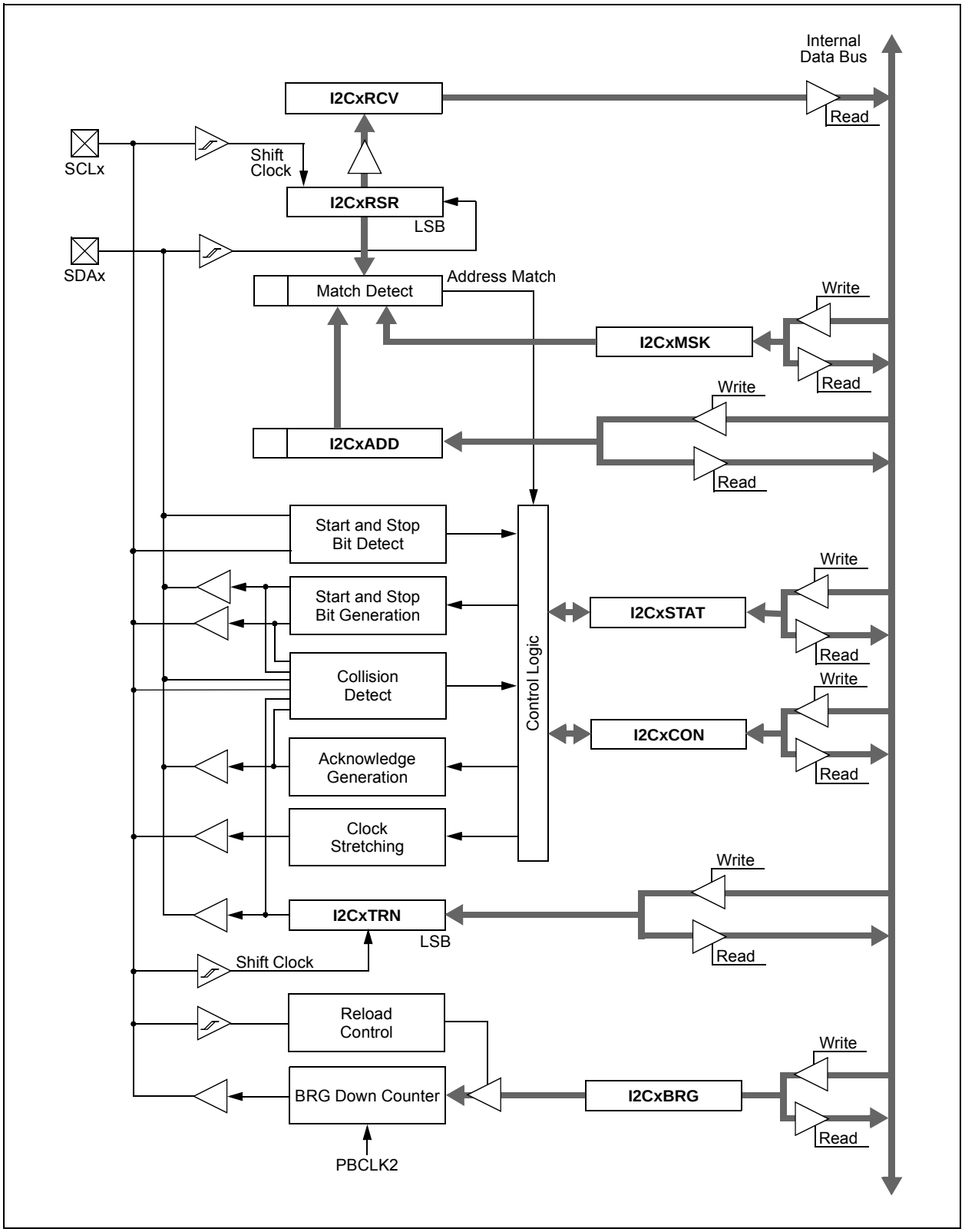
These bits are used to add fractional delays to SQI Clock Output while writing the data to the Flash in all modes of operation.

1111 = 16 taps added on clock output
1110 = 15 taps added on clock output

•
•
•

0001 = 2 taps added on clock output
0000 = 1 tap added on clock output

FIGURE 23-1: I²C BLOCK DIAGRAM



REGISTER 24-1: UxMODE: UARTx MODE REGISTER (CONTINUED)

bit 12 **IREN:** IrDA® Encoder and Decoder Enable bit
- 1 = IrDA is enabled
- 0 = IrDA is disabled

bit 11 **RTSMD:** Mode Selection for $\overline{\text{UxRTS}}$ Pin bit
- 1 = $\overline{\text{UxRTS}}$ pin is in Simplex mode
- 0 = $\overline{\text{UxRTS}}$ pin is in Flow Control mode

bit 10 **Unimplemented:** Read as '0'

bit 9-8 **UEN<1:0>:** UARTx Module Enable bits(1)
- 11 = UxTX, UxRX and UxBCLK pins are enabled and used; $\overline{\text{UxCTS}}$ pin is controlled by corresponding bits in the PORTx register
- 10 = UxTX, UxRX, $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins are enabled and used
- 01 = UxTX, UxRX and $\overline{\text{UxRTS}}$ pins are enabled and used; $\overline{\text{UxCTS}}$ pin is controlled by corresponding bits in the PORTx register
- 00 = UxTX and UxRX pins are enabled and used; $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$/UxBCLK pins are controlled by corresponding bits in the PORTx register

bit 7 **WAKE:** Enable Wake-up on Start bit Detect During Sleep Mode bit
- 1 = Wake-up enabled
- 0 = Wake-up disabled

bit 6 **LPBACK:** UARTx Loopback Mode Select bit
- 1 = Loopback mode is enabled
- 0 = Loopback mode is disabled

bit 5 **ABAUD:** Auto-Baud Enable bit
- 1 = Enable baud rate measurement on the next character – requires reception of Sync character (0x55); cleared by hardware upon completion
- 0 = Baud rate measurement disabled or completed

bit 4 **RXINV:** Receive Polarity Inversion bit
- 1 = UxRX Idle state is '0'
- 0 = UxRX Idle state is '1'

bit 3 **BRGH:** High Baud Rate Enable bit
- 1 = High-Speed mode – 4x baud clock enabled
- 0 = Standard Speed mode – 16x baud clock enabled

bit 2-1 **PDSEL<1:0>:** Parity and Data Selection bits
- 11 = 9-bit data, no parity
- 10 = 8-bit data, odd parity
- 01 = 8-bit data, even parity
- 00 = 8-bit data, no parity

bit 0 **STSEL:** Stop Selection bit
- 1 = 2 Stop bits
- 0 = 1 Stop bit

Note 1: These bits are present for legacy compatibility, and are superseded by PPS functionality on these devices (see **Section 12.4 "Peripheral Pin Select (PPS)"** for more information).

FIGURE 27-11: CRYPTO ENGINE SECURITY ASSOCIATION STRUCTURE (CONTINUED)

Name		Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 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>							
SA_ENCKEY4	31:24	ENCKEY<31:24>							
	23:16	ENCKEY<23:16>							
	15:8	ENCKEY<15:8>							
	7:0	ENCKEY<7:0>							
SA_ENCKEY5	31:24	ENCKEY<31:24>							
	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>							
SA_ENCKEY7	31:24	ENCKEY<31:24>							
	23:16	ENCKEY<23:16>							
	15:8	ENCKEY<15:8>							
	7:0	ENCKEY<7:0>							
SA_ENCKEY8	31:24	ENCKEY<31:24>							
	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>							
SA_AUTHIV2	31:24	AUTHIV<31:24>							
	23:16	AUTHIV<23:16>							
	15:8	AUTHIV<15:8>							
	7:0	AUTHIV<7:0>							
SA_AUTHIV3	31:24	AUTHIV<31:24>							
	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>							
SA_AUTHIV5	31:24	AUTHIV<31:24>							
	23:16	AUTHIV<23:16>							
	15:8	AUTHIV<15:8>							
	7:0	AUTHIV<7:0>							
SA_AUTHIV6	31:24	AUTHIV<31:24>							
	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>							
SA_AUTHIV8	31:24	AUTHIV<31:24>							
	23:16	AUTHIV<23:16>							
	15:8	AUTHIV<15:8>							
	7:0	AUTHIV<7:0>							

REGISTER 28-2: RNGCON: RANDOM NUMBER GENERATOR CONTROL REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	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(1)	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>							

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31-13 **Unimplemented:** Read as '0'

bit 12 **LOAD:** Device Select bit

This bit is self-clearing and is used to load the seed from the TRNG (i.e., the random value) as a seed to the PRNG.

bit 11 **TRNGMODE:** True Random Number Generator Mode bit(1)

1 = Enhanced TRNG mode is selected
0 = Normal TRNG mode is selected

bit 10 **CONT:** PRNG Number Shift Enable bit

1 = The PRNG random number is shifted every cycle
0 = The PRNG random number is shifted when the previous value is removed

bit 9 **PRNGEN:** PRNG Operation Enable bit

1 = PRNG operation is enabled
0 = PRNG operation is not enabled

bit 8 **TRNGEN:** TRNG Operation Enable bit

1 = TRNG operation is enabled
0 = TRNG operation is not enabled

bit 7-0 **PLEN<7:0>:** PRNG Polynomial Length bits

These bits contain the length of the polynomial used for the PRNG.

Note 1: This bit is effective only when the TRNGEN bit is set to '1'.

30.0 CONTROLLER AREA NETWORK (CAN)

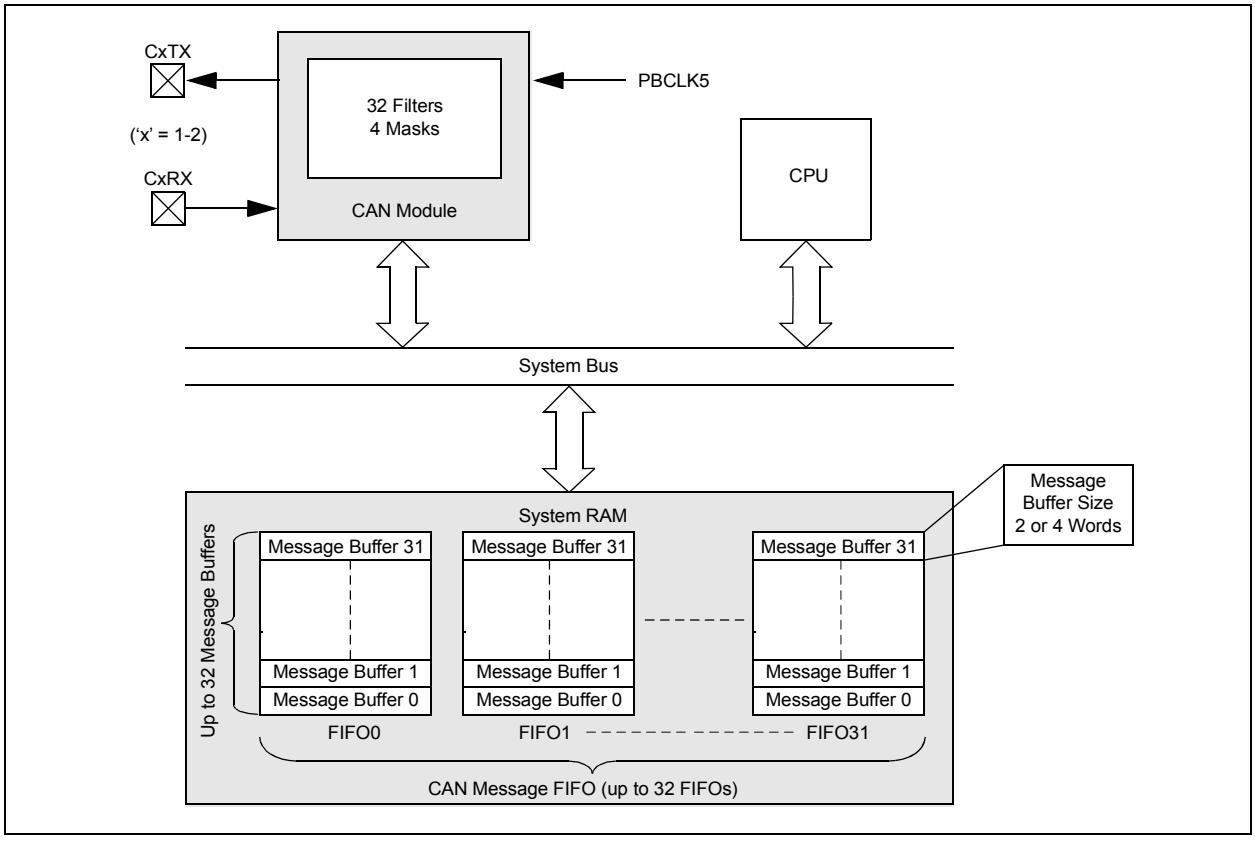
Note: This data sheet summarizes the features of the PIC32MZ Graphics (DA) Family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 34. "Controller Area Network (CAN)"** (DS60001154), which is available from the *Documentation > Reference Manual* section of the Microchip PIC32 web site (www.microchip.com/pic32).

The Controller Area Network (CAN) module supports the following key features:

- Standards Compliance:
  - Full CAN 2.0B compliance
  - Programmable bit rate up to 1 Mbps
- Message Reception and Transmission:
  - 32 message FIFOs
  - Each FIFO can have up to 32 messages for a total of 1024 messages
  - FIFO can be a transmit message FIFO or a receive message FIFO
  - User-defined priority levels for message FIFOs used for transmission
  - 32 acceptance filters for message filtering
  - Four acceptance filter mask registers for message filtering
  - Automatic response to remote transmit request
  - DeviceNet™ addressing support
- Additional Features:
  - Loopback, Listen All Messages and Listen Only modes for self-test, system diagnostics and bus monitoring
  - Low-power operating modes
  - CAN module is a bus master on the PIC32 System Bus
  - Use of DMA is not required
  - Dedicated time-stamp timer
  - Dedicated DMA channels
  - Data-only Message Reception mode

Figure 30-1 illustrates the general structure of the CAN module.

FIGURE 30-1: PIC32 CAN MODULE BLOCK DIAGRAM



REGISTER 30-2: CiCFG: CAN BAUD RATE CONFIGURATION REGISTER (CONTINUED)

bit 10-8 **PRSEG<2:0>:** Propagation Time Segment bits(4)
111 = Length is 8 x TQ
•
•
•
000 = Length is 1 x TQ

bit 7-6 **SJW<1:0>:** Synchronization Jump Width bits(3)
11 = Length is 4 x TQ
10 = Length is 3 x TQ
01 = Length is 2 x TQ
00 = Length is 1 x TQ

bit 5-0 **BRP<5:0>:** Baud Rate Prescaler bits
111111 = TQ = (2 x 64)/TPBCLK5
111110 = TQ = (2 x 63)/TPBCLK5
•
•
•
000001 = TQ = (2 x 2)/TPBCLK5
000000 = TQ = (2 x 1)/TPBCLK5

Note 1: SEG2PH ≤ SEG1PH. If SEG2PHTS is clear, SEG2PH will be set automatically.
2: 3 Time bit sampling is not allowed for BRP < 2.
3: SJW ≤ SEG2PH.
4: The Time Quanta per bit must be greater than 7 (that is, TQBIT > 7).

Note:	This register can only be modified when the CAN module is in Configuration mode (OPMOD<2:0> (CiCON<23:21>) = 100).

REGISTER 30-3: CiINT: CAN INTERRUPT REGISTER

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	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(1)	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

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31 **IVRIE:** Invalid Message Received Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 30 **WAKIE:** CAN Bus Activity Wake-up Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 29 **CERRIE:** CAN Bus Error Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 28 **SERRIE:** System Error Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 27 **RBOVIE:** Receive Buffer Overflow Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 26-20 **Unimplemented:** Read as '0'

bit 19 **MODIE:** Mode Change Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 18 **CTMRIE:** CAN Timestamp Timer Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 17 **RBIE:** Receive Buffer Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 16 **TBIE:** Transmit Buffer Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 15 **IVRIF:** Invalid Message Received Interrupt Flag bit
1 = An invalid messages interrupt has occurred
0 = An invalid message interrupt has not occurred

Note 1: This bit can only be cleared by turning the CAN module Off and On by clearing or setting the ON bit (CiCON<15>).

REGISTER 36-14: GLCDLxRES: GRAPHICS LCD CONTROLLER LAYER 'x' RESOLUTION REGISTER ('x' = 0-2)

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	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>							

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31-27 **Unimplemented:** Read as '0'

bit 26-16 **RESX<10:0>:** X Dimension Layer Pixel Resolution bits
These bits specify the layer pixel resolution in the X dimension.

bit 15-11 **Unimplemented:** Read as '0'

bit 10-0 **RESY<10:0>:** Y Dimension Layer Pixel Resolution bits
These bits specify the layer pixel resolution in the Y dimension.

REGISTER 38-6: DDRMEMCFG0: DDR MEMORY CONFIGURATION REGISTER 0

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	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>				

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31 **Unimplemented:** Read as '0'

bit 30 **APCHRGEN:** Automatic Precharge Enable bit

When set, this bit issues an auto-precharge command to close the bank at the end of every user command. If the command accesses more than one bank before completing, all banks accessed are auto-precharged.

1 = Issue an auto-precharged command

0 = Do not issue an auto-precharged command

bit 29 **Unimplemented:** Read as '0'

bit 28-24 **CLHADDR<4:0>:** Column Address Shift bits

These bits specify how many bits the controller address must be right-shifted to put the high part of the column address to the immediate left of the low part of the column address. Used in conjunction with CLADDRHMSK (DDRMEMCFG2<26:0>) and CLADDRLMASK (DDRMEMCFG3<26:0>).

bit 23-21 **Unimplemented:** Read as '0'

bit 20-16 **CSADDR<4:0>:** Chip Select Shift bits

These bits specify which bits of user address space are used to derive the Chip Select address for the DDR memory. Used in conjunction with CSADDRMASK (DDRMEMCFG4<10:8>).

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **BNKADDR<4:0>:** Bank Address Select Shift bits

These bits specify which bits of user address space are used to derive the bank address for the DDR memory. Used in conjunction with BNKADDRMASK (DDRMEMCFG4<2:0>).

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **RWADDR<4:0>:** Row Address Select Shift bits

These bits specify which bits of user address space are used to derive the row address for the DDR memory. Used in conjunction with RWADDRMSK (DDRMEMCFG1<12:0>).

REGISTER 41-4: DEVCFG1/ADEVCFG1: DEVICE CONFIGURATION WORD 1

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	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>		

Legend:	r = Reserved bit	P = Programmable bit	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31 **FDMTEN:** Deadman Timer enable bit
1 = Deadman Timer is enabled and *cannot* be disabled by software
0 = Deadman Timer is disabled and *can* be enabled by software

bit 30-26 **DMTCNT<4:0>:** Deadman Timer Count Select bits
11111 = Reserved
.
.
.
11000 = Reserved
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>:** Watchdog Timer Window Size bits
11 = Window size is 25%
10 = Window size is 37.5%
01 = Window size is 50%
00 = Window size is 75%

bit 23 **FWDTEN:** Watchdog Timer Enable bit
1 = Watchdog Timer is enabled and cannot be disabled by software
0 = Watchdog Timer is not enabled; it can be enabled in software

bit 22 **WINDIS:** Watchdog Timer Window Enable bit
1 = Watchdog Timer is in non-Window mode
0 = Watchdog Timer is in Window mode

bit 21 **WDTSPGM:** Watchdog Timer Stop During Flash Programming bit
1 = Watchdog Timer stops during Flash programming
0 = Watchdog Timer runs during Flash programming (for read/execute while programming Flash applications)

43.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16, and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

43.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

43.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

43.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Trademarks

ISBN: 978-1-5224-2532-8