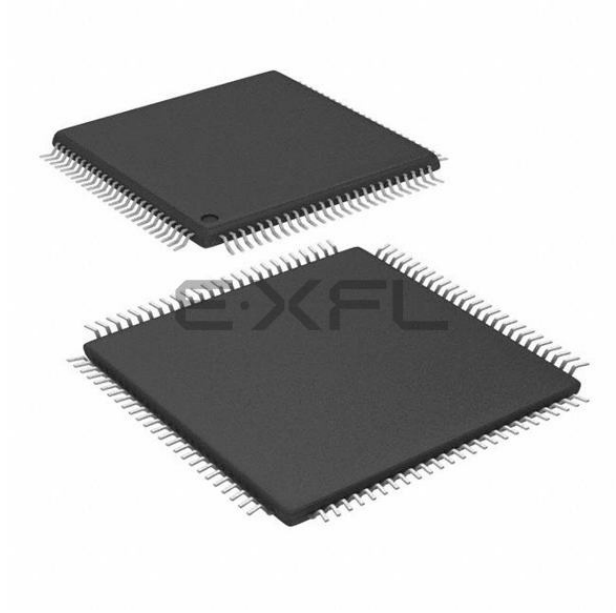
Microchip Technology - PIC24FJ1024GA610-I/PT Datasheet



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	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, LINbus, PMP, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, HLVD, POR, PWM, WDT
Number of I/O	85
Program Memory Size	1MB (341.5K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 24x10/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj1024ga610-i-pt

Referenced Sources

This device data sheet is based on the following individual chapters of the *"dsPIC33/PIC24 Family Reference Manual"*. These documents should be considered as the general reference for the operation of a particular module or device feature.

Note 1: To access the documents listed below, browse to the documentation section of the PIC24FJ1024GA610/GB610 product page of the Microchip web site (www.microchip.com) or select a family reference manual section from the following list.

In addition to parameters, features and other documentation, the resulting page provides links to the related family reference manual sections.

- **"CPU with Extended Data Space (EDS)"** (DS39732)
- **"Data Memory with Extended Data Space (EDS)"** (DS39733)
- **"Direct Memory Access Controller (DMA)"** (DS39742)
- **"PIC24F Flash Program Memory"** (DS30009715)
- **"Reset"** (DS39712)
- **"Interrupts"** (DS70000600)
- **"Power-Saving Features"** (DS39698)
- **"I/O Ports with Peripheral Pin Select (PPS)"** (DS39711)
- **"Timers"** (DS39704)
- **"Input Capture with Dedicated Timer"** (DS70000352)
- **"Output Compare with Dedicated Timer"** (DS70005159)
- **"Capture/Compare/PWM/Timer (MCCP and SCCP)"** (DS33035A)
- **"Serial Peripheral Interface (SPI) with Audio Codec Support"** (DS70005136)
- **"Inter-Integrated Circuit (I^2C)"** (DS70000195)
- **"UART"** (DS39708)
- **"USB On-The-Go (OTG)"** (DS39721)
- **"Enhanced Parallel Master Port (EPMP)"** (DS39730)
- **"RTCC with Timestamp"** (DS70005193)
- **"RTCC with External Power Control"** (DS39745)
- **"32-Bit Programmable Cyclic Redundancy Check (CRC)"** (DS30009729)
- **"12-Bit A/D Converter with Threshold Detect"** (DS39739)
- **"Scalable Comparator Module"** (DS39734)
- **"Dual Comparator Module"** (DS39710)
- **"Charge Time Measurement Unit (CTMU) and CTMU Operation with Threshold Detect"** (DS30009743)
- **"High-Level Integration with Programmable High/Low-Voltage Detect (HLVD)"** (DS39725)
- **"Watchdog Timer (WDT)"** (DS39697)
- **"CodeGuard™ Intermediate Security"** (DS70005182)
- **"High-Level Device Integration"** (DS39719)
- **"Programming and Diagnostics"** (DS39716)
- **"Dual Partition Flash Program Memory"** (DS70005156)

2.4 Voltage Regulator Pin (VCAP)

Note: This section applies only to PIC24FJ devices with an on-chip voltage regulator.

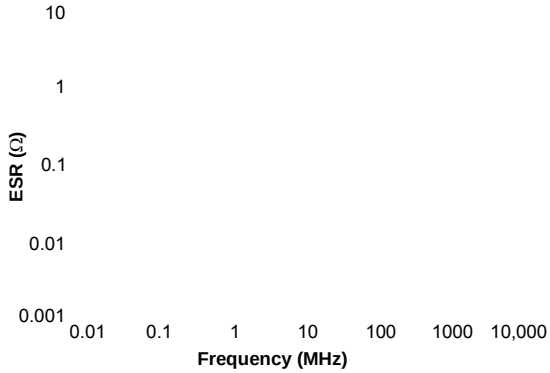
Refer to **Section 30.3 "On-Chip Voltage Regulator"** for details on connecting and using the on-chip regulator.

A low-ESR (< 5Ω) capacitor is required on the VCAP pin to stabilize the voltage regulator output voltage. The VCAP pin must not be connected to VDD and must use a capacitor of 10 µF connected to ground. The type can be ceramic or tantalum. Suitable examples of capacitors are shown in Table 2-1. Capacitors with equivalent specifications can be used.

Designers may use Figure 2-3 to evaluate the ESR equivalence of candidate devices.

The placement of this capacitor should be close to VCAP. It is recommended that the trace length not exceed 0.25 inch (6 mm). Refer to **Section 33.0 "Electrical Characteristics"** for additional information.

FIGURE 2-3: FREQUENCY vs. ESR PERFORMANCE FOR SUGGESTED VCAP



Note: Typical data measurement at +25°C, 0V DC bias.

TABLE 2-1: SUITABLE CAPACITOR EQUIVALENTS (0805 CASE SIZE)

Make	Part #	Nominal Capacitance	Base Tolerance	Rated Voltage
TDK	C2012X5R1E106K085AC	10 µF	±10%	25V
TDK	C2012X5R1C106K085AC	10 µF	±10%	16V
Kemet	C0805C106M4PACTU	10 µF	±10%	16V
Murata	GRM21BR61E106KA3L	10 µF	±10%	25V
Murata	GRM21BR61C106KE15	10 µF	±10%	16V

4.0 MEMORY ORGANIZATION

As Harvard architecture devices, PIC24F microcontrollers feature separate program and data memory spaces and buses. This architecture also allows direct access of program memory from the Data Space during code execution.

4.1 Program Memory Space

The program address memory space of the PIC24FJ1024GA610/GB610 family devices is 4M instructions. The space is addressable by a 24-bit value derived from either the 23-bit Program Counter (PC) during program execution, or from table operation or Data Space remapping, as described in **Section 4.3 "Interfacing Program and Data Memory Spaces"**.

User access to the program memory space is restricted to the lower half of the address range (000000h to 7FFFFFh). The exception is the use of TBLRD/TBLWT operations, which use TBLPAG<7> to permit access to the Configuration bits and customer OTP sections of the configuration memory space.

The PIC24FJ1024GA610/GB610 family of devices supports a Single Partition mode and two Dual Partition modes. The Dual Partition modes allow the device to be programmed with two separate applications to facilitate bootloading or to allow an application to be programmed at run time without stalling the CPU.

Memory maps for the PIC24FJ1024GA610/GB610 family of devices are shown in Figure 4-1.

TABLE 4-4: SFR MAP: 0000h BLOCK

File Name	Address	All Resets	File Name	Address	All Resets
CPU CORE			**INTERRUPT CONTROLLER (CONTINUED)**		
WREG0	0000	0000	IEC1	009A	0000
WREG1	0002	0000	IEC2	009C	0000
WREG2	0004	0000	IEC3	009E	0000
WREG3	0006	0000	IEC4	00A0	0000
WREG4	0008	0000	IEC5	00A2	0000
WREG5	000A	0000	IEC6	00A4	0000
WREG6	000C	0000	IEC7	00A6	0000
WREG7	000E	0000	IPC0	00A8	4444
WREG8	0010	0000	IPC1	00AA	4444
WREG9	0012	0000	IPC2	00AC	4444
WREG10	0014	0000	IPC3	00AE	4444
WREG11	0016	0000	IPC4	00B0	4444
WREG12	0018	0000	IPC5	00B2	4404
WREG13	001A	0000	IPC6	00B4	4444
WREG14	001C	0000	IPC7	00B6	4444
WREG15	001E	0800	IPC8	00B8	0044
SPLIM	0020	xxxx	IPC9	00BA	4444
PCL	002E	0000	IPC10	00BC	4444
PCH	0030	0000	IPC11	00BE	4444
DSRPAG	0032	0000	IPC12	00C0	4444
DSWPAG	0034	0000	IPC13	00C2	0440
RCOUNT	0036	xxxx	IPC14	00C4	4400
SR	0042	0000	IPC15	00C6	4444
CORCON	0044	0004	IPC16	00C8	4444
DISICNT	0052	xxxx	IPC17	00CA	4444
TBLPAG	0054	0000	IPC18	00CC	0044
INTERRUPT CONTROLLER			IPC19	00CE	0040
INTCON1	0080	0000	IPC20	00D0	4440
INTCON2	0082	8000	IPC21	00D2	4444
INTCON4	0086	0000	IPC22	00D4	4444
IFS0	0088	0000	IPC23	00D6	4400
IFS1	008A	0000	IPC24	00D8	4444
IFS2	008C	0000	IPC25	00DA	0440
IFS3	008E	0000	IPC26	00DC	0400
IFS4	0090	0000	IPC27	00DE	4440
IFS5	0092	0000	IPC28	00E0	4444
IFS6	0094	0000	IPC29	00E2	0044
IFS7	0096	0000	INTTREG	00E4	0000
IEC0	0098	0000			

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

4.3 Interfacing Program and Data Memory Spaces

The PIC24F architecture uses a 24-bit wide program space and 16-bit wide Data Space. The architecture is also a modified Harvard scheme, meaning that data can also be present in the program space. To use this data successfully, it must be accessed in a way that preserves the alignment of information in both spaces.

Aside from normal execution, the PIC24F architecture provides two methods by which program space can be accessed during operation:

- Using table instructions to access individual bytes or words anywhere in the program space
- Remapping a portion of the program space into the Data Space (Program Space Visibility)

Table instructions allow an application to read or write to small areas of the program memory. This makes the method ideal for accessing data tables that need to be updated from time to time. It also allows access to all bytes of the program word. The remapping method allows an application to access a large block of data on a read-only basis, which is ideal for look-ups from a large table of static data. It can only access the least significant word of the program word.

4.3.1 ADDRESSING PROGRAM SPACE

Since the address ranges for the data and program spaces are 16 and 24 bits, respectively, a method is needed to create a 23-bit or 24-bit program address from 16-bit data registers. The solution depends on the interface method to be used.

For table operations, the 8-bit Table Memory Page Address register (TBLPAG) is used to define a 32K word region within the program space. This is concatenated with a 16-bit EA to arrive at a full 24-bit program space address. In this format, the MSBs of TBLPAG are used to determine if the operation occurs in the user memory (TBLPAG<7> = 0) or the configuration memory (TBLPAG<7> = 1).

For remapping operations, the 10-bit Extended Data Space Read register (DSRPAG) is used to define a 16K word page in the program space. When the Most Significant bit (MSb) of the EA is '1', and the MSb (bit 9) of DSRPAG is '1', the lower 8 bits of DSRPAG are concatenated with the lower 15 bits of the EA to form a 23-bit program space address. The DSRPAG<8> bit decides whether the lower word (when bit is '0') or the higher word (when bit is '1') of program memory is mapped. Unlike table operations, this strictly limits remapping operations to the user memory area.

Table 4-14 and Figure 4-8 show how the program EA is created for table operations and remapping accesses from the data EA. Here, P<23:0> refers to a program space word, whereas D<15:0> refers to a Data Space word.

TABLE 4-14: PROGRAM SPACE ADDRESS CONSTRUCTION

Access Type	Access Space	Program Space Address				
		<23>	<22:16>	<15>	<14:1>	<0>
Instruction Access (Code Execution)	User	0	PC<22:1>			0
		0xx xxxx xxxx xxxx xxxx xxx0				
TBLRD/TBLWT (Byte/Word Read/Write)	User	TBLPAG<7:0>		Data EA<15:0>		
		0xxx xxxx		xxxx xxxx xxxx xxxx		
	Configuration	TBLPAG<7:0>		Data EA<15:0>		
		1xxx xxxx		xxxx xxxx xxxx xxxx		
Program Space Visibility (Block Remap/Read)	User	0	DSRPAG<7:0>(2)		Data EA<14:0>(1)	
		0	xxxx xxxx		xxx xxxx xxxx xxxx	

Note 1: Data EA<15> is always '1' in this case, but is not used in calculating the program space address. Bit 15 of the address is DSRPAG<0>.

2: DSRPAG<9> is always '1' in this case. DSRPAG<8> decides whether the lower word or higher word of program memory is read. When DSRPAG<8> is '0', the lower word is read, and when it is '1', the higher word is read.

TABLE 8-2: INTERRUPT VECTOR DETAILS (CONTINUED)

Interrupt Source	IRQ #	IVT Address	Interrupt Bit Location		
			Flag	Enable	Priority
I2C1BC – I2C1 Bus Collision	84	0000BCh	IFS5<4>	IEC5<4>	I2C1BCInterrupt
I2C2BC – I2C2 Bus Collision	85	0000BEh	IFS5<5>	IEC5<5>	I2C2BCInterrupt
USB1 – USB1 Interrupt	86	0000C0h	IFS5<6>	IEC5<6>	USB1Interrupt
U4E – UART4 Error	87	0000C2h	IFS5<7>	IEC5<7>	U4ErrInterrupt
U4RX – UART4 Receiver	88	0000C4h	IFS5<8>	IEC5<8>	U4RXInterrupt
U4TX – UART4 Transmitter	89	0000C6h	IFS5<9>	IEC5<9>	U4TXInterrupt
SPI3 – SPI3 General	90	0000C8h	IFS5<10>	IEC5<10>	SPI3Interrupt
SPI3TX – SPI3 Transfer Done	91	0000CAh	IFS5<11>	IEC5<11>	SPI3TXInterrupt
—	92	92	—	—	—
—	93	93	—	—	—
CCP3 – Capture/Compare 3	94	0000D0h	IFS5<14>	IEC5<14>	CCP3Interrupt
CCP4 – Capture/Compare 4	95	0000D2h	IFS5<15>	IEC5<15>	CCP4Interrupt
CLC1 – Configurable Logic Cell 1	96	0000D4h	IFS6<0>	IEC6<0>	CLC1Interrupt
CLC2 – Configurable Logic Cell 2	97	0000D6h	IFS6<1>	IEC6<1>	CLC2Interrupt
CLC3 – Configurable Logic Cell 3	98	0000D8h	IFS6<2>	IEC6<2>	CLC3Interrupt
CLC4 – Configurable Logic Cell 4	99	0000DAh	IFS6<3>	IEC6<3>	CLC4Interrupt
—	100	—	—	—	—
CCT1 – Capture/Compare Timer1	101	0000DEh	IFS6<5>	IEC6<5>	CCT1Interrupt
CCT2 – Capture/Compare Timer2	102	0000E0h	IFS6<6>	IEC6<6>	CCT2Interrupt
—	103	—	—	—	—
—	104	—	—	—	—
—	105	—	—	—	—
FST – FRC Self-Tuning Interrupt	106	0000E8h	IFS6<10>	IEC6<10>	FSTInterrupt
—	107	—	—	—	—
—	108	—	—	—	—
I2C3BC – I2C3 Bus Collision	109	0000EEh	IFS6<13>	IEC6<13>	I2C3BCInterrupt
RTCCTS – Real-Time Clock Timestamp	110	0000F0h	IFS6<14>	IEC6<14>	RTCCTSInterrupt
U5RX – UART5 Receiver	111	0000F2h	IFS6<15>	IEC6<15>	U5RXInterrupt
U5TX – UART5 Transmitter	112	0000F4h	IFS7<0>	IEC7<0>	U5TXInterrupt
U5E – UART5 Error	113	0000F6h	IFS7<1>	IEC7<1>	U5ErrInterrupt
U6RX – UART6 Receiver	114	0000F8h	IFS7<2>	IEC7<2>	U6RXInterrupt
U6TX – UART6 Transmitter	115	0000FAh	IFS7<3>	IEC7<3>	U6TXInterrupt
U6E – UART6 Error	116	0000FCh	IFS7<4>	IEC7<4>	U6ErrInterrupt
JTAG – JTAG	117	0000FEh	IFS7<5>	IEC7<5>	JTAGInterrupt

REGISTER 9-8: REFOCONL: REFERENCE OSCILLATOR CONTROL REGISTER LOW

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-0
ROEN	—	ROSIDL	ROOUT	ROSLP	—	ROSWEN	ROACTIVE
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	ROSEL3	ROSEL2	ROSEL1	ROSEL0
bit 7							bit 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 15 **ROEN:** Reference Oscillator Enable bit
1 = Reference Oscillator module is enabled
0 = Reference Oscillator is disabled

bit 14 **Unimplemented:** Read as '0'

bit 13 **ROSIDL:** REFO Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **ROOUT:** Reference Clock Output Enable bit
1 = Reference clock is driven out on the REFO pin
0 = Reference clock is not driven out on the REFO pin

bit 11 **ROSLP:** Reference Oscillator Output Stop in Sleep bit
1 = Reference Oscillator continues to run in Sleep
0 = Reference Oscillator is disabled in Sleep

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

bit 9 **ROSWEN:** Reference Clock RODIV<14:0>/ROTRIM<0:8> Switch Enable bit
1 = Switch clock divider; clock divider switching is currently in progress
0 = Clock divider switch has been completed

bit 8 **ROACTIVE:** Reference Clock Request Status bit
1 = Reference clock is active (user should not change the REFO settings)
0 = Reference clock is inactive (user can update the REFO settings)

bit 7-4 **Unimplemented:** Read as '0'

bit 3-0 **ROSEL<3:0>:** Reference Clock Source Select bits
1111-1001 = Reserved
1000 = REFI pin
0111 = Reserved
0110 = PLL (4/6/8x or 96 MHz)
0101 = SOSC
0100 = LPRC
0011 = FRC
0010 = POSC
0001 = Peripheral clock
0000 = Oscillator clock

REGISTER 11-22: RPINR12: PERIPHERAL PIN SELECT INPUT REGISTER 12

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	TCKIBR5	TCKIBR4	TCKIBR3	TCKIBR2	TCKIBR1	TCKIBR0
bit 15							bit 8

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	TCKIAR5	TCKIAR4	TCKIAR3	TCKIAR2	TCKIAR1	TCKIAR0
bit 7							bit 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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **TCKIBR<5:0>:** Assign MCCP/SCCP Clock Input B to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **TCKIAR<5:0>:** Assign MCCP/SCCP Clock Input A to Corresponding RPn or RPIn Pin bits

REGISTER 11-23: RPINR14: PERIPHERAL PIN SELECT INPUT REGISTER 14

U-0	U-0	r-1	r-1	r-1	r-1	r-1	r-1
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	r-1	r-1	r-1	r-1	r-1	r-1
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:	r = Reserved 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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **Reserved**: Maintain as ‘1’

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **Reserved**: Maintain as ‘1’

REGISTER 11-42: RPOR6: PERIPHERAL PIN SELECT OUTPUT REGISTER 6

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP13R5	RP13R4	RP13R3	RP13R2	RP13R1	RP13R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP12R5	RP12R4	RP12R3	RP12R2	RP12R1	RP12R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP13R<5:0>:** RP13 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP13 (see Table 11-4 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RP12R<5:0>:** RP12 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP12 (see Table 11-4 for peripheral function numbers).

REGISTER 11-43: RPOR7: PERIPHERAL PIN SELECT OUTPUT REGISTER 7

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP15R5(1)	RP15R4(1)	RP15R3(1)	RP15R2(1)	RP15R1(1)	RP15R0(1)
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP14R5	RP14R4	RP14R3	RP14R2	RP14R1	RP14R0
bit 7							bit 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 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP15R<5:0>:** RP15 Output Pin Mapping bits(1)
Peripheral Output Number n is assigned to pin, RP15 (see Table 11-4 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RP14R<5:0>:** RP14 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP14 (see Table 11-4 for peripheral function numbers).

Note 1: This pin is not available on 64-pin devices.

REGISTER 15-1: OCxCON1: OUTPUT COMPARE x CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2(2)	ENFLT1(2)
bit 15							bit 8

R/W-0	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
ENFLT0(2)	OCFLT2(2,3)	OCFLT1(2,4)	OCFLT0(2,4)	TRIGMODE	OCM2(1)	OCM1(1)	OCM0(1)
bit 7							bit 0

Legend:	HSC = Hardware Settable/Clearable 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 15-14 **Unimplemented:** Read as '0'

bit 13 **OCSIDL:** Output Compare x Stop in Idle Mode Control bit
- 1 = Output Compare x halts in CPU Idle mode
- 0 = Output Compare x continues to operate in CPU Idle mode

bit 12-10 **OCTSEL<2:0>:** Output Compare x Timer Select bits
- 111 = Peripheral clock (FCY)
- 110 = Reserved
- 101 = Reserved
- 100 = Timer1 clock (only synchronous clock is supported)
- 011 = Timer5 clock
- 010 = Timer4 clock
- 001 = Timer3 clock
- 000 = Timer2 clock

bit 9 **ENFLT2:** Fault Input 2 Enable bit(2)
- 1 = Fault 2 (Comparator 1/2/3 out) is enabled(3)
- 0 = Fault 2 is disabled

bit 8 **ENFLT1:** Fault Input 1 Enable bit(2)
- 1 = Fault 1 (OCFB pin) is enabled(4)
- 0 = Fault 1 is disabled

bit 7 **ENFLT0:** Fault Input 0 Enable bit(2)
- 1 = Fault 0 (OCFA pin) is enabled(4)
- 0 = Fault 0 is disabled

bit 6 **OCFLT2:** Output Compare x PWM Fault 2 (Comparator 1/2/3) Condition Status bit(2,3)
- 1 = PWM Fault 2 has occurred
- 0 = No PWM Fault 2 has occurred

bit 5 **OCFLT1:** Output Compare x PWM Fault 1 (OCFB pin) Condition Status bit(2,4)
- 1 = PWM Fault 1 has occurred
- 0 = No PWM Fault 1 has occurred

Note 1: The OCx output must also be configured to an available RPn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

2: The Fault input enable and Fault status bits are valid when OCM<2:0> = 111 or 110.

3: The Comparator 1 output controls the OC1-OC3 channels, Comparator 2 output controls the OC4-OC6 channels, Comparator 3 output controls the OC7-OC9 channels.

4: The OCFA/OCFB Fault inputs must also be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 "Peripheral Pin Select (PPS)"**.

REGISTER 17-8: SPIxBRGL: SPIx BAUD RATE GENERATOR REGISTER LOW

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	BRG<12:8>(1)				
bit 15							bit 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
BRG<7:0>(1)							
bit 7							bit 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 15-13 **Unimplemented:** Read as '0'

bit 12-0 **BRG<12:0>:** SPIx Baud Rate Generator Divisor bits(1)

Note 1: Changing the BRG value when SPIEN = 1 causes undefined behavior.

REGISTER 17-10: SPIxIMSKH: SPIx INTERRUPT MASK REGISTER HIGH

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RXWIEN	—	RXMSK5(1)	RXMSK4(1,4)	RXMSK3(1,3)	RXMSK2(1,2)	RXMSK1(1)	RXMSK0(1)
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TXWIEN	—	TXMSK5(1)	TXMSK4(1,4)	TXMSK3(1,3)	TXMSK2(1,2)	TXMSK1(1)	TXMSK0(1)
bit 7							bit 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 15 **RXWIEN:** Receive Watermark Interrupt Enable bit
1 = Triggers receive buffer element watermark interrupt when RXMSK<5:0> ≤ RXELM<5:0>
0 = Disables receive buffer element watermark interrupt

bit 14 **Unimplemented:** Read as '0'

bit 13-8 **RXMSK<5:0>:** RX Buffer Mask bits(1,2,3,4)
RX mask bits; used in conjunction with the RXWIEN bit.

bit 7 **TXWIEN:** Transmit Watermark Interrupt Enable bit
1 = Triggers transmit buffer element watermark interrupt when TXMSK<5:0> = TXELM<5:0>
0 = Disables transmit buffer element watermark interrupt

bit 6 **Unimplemented:** Read as '0'

bit 5-0 **TXMSK<5:0>:** TX Buffer Mask bits(1,2,3,4)
TX mask bits; used in conjunction with the TXWIEN bit.

Note 1: Mask values higher than FIFODEPTH are not valid. The module will not trigger a match for any value in this case.
2: RXMSK2 and TXMSK2 bits are only present when FIFODEPTH = 8 or higher.
3: RXMSK3 and TXMSK3 bits are only present when FIFODEPTH = 16 or higher.
4: RXMSK4 and TXMSK4 bits are only present when FIFODEPTH = 32.

REGISTER 19-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (CONTINUED)

bit 7-6 **URXISEL<1:0>:** UARTx Receive Interrupt Mode Selection bits
11 = Interrupt is set on an RSR transfer, making the receive buffer full (i.e., has 4 data characters)
10 = Interrupt is set on an RSR transfer, making the receive buffer 3/4 full (i.e., has 3 data characters)
0x = Interrupt is set when any character is received and transferred from the RSR to the receive buffer; receive buffer has one or more characters

bit 5 **ADDEN:** Address Character Detect bit (bit 8 of received data = 1)
1 = Address Detect mode is enabled (if 9-bit mode is not selected, this does not take effect)
0 = Address Detect mode is disabled

bit 4 **RIDLE:** Receiver Idle bit (read-only)
1 = Receiver is Idle
0 = Receiver is active

bit 3 **PERR:** Parity Error Status bit (read-only)
1 = Parity error has been detected for the current character (the character at the top of the receive FIFO)
0 = Parity error has not been detected

bit 2 **FERR:** Framing Error Status bit (read-only)
1 = Framing error has been detected for the current character (the character at the top of the receive FIFO)
0 = Framing error has not been detected

bit 1 **OERR:** Receive Buffer Overrun Error Status bit (clear/read-only)
1 = Receive buffer has overflowed
0 = Receive buffer has not overflowed (clearing a previously set OERR bit, 1 → 0 transition); will reset the receive buffer and the RSR to the empty state

bit 0 **URXDA:** UARTx Receive Buffer Data Available bit (read-only)
1 = Receive buffer has data, at least one more character can be read
0 = Receive buffer is empty

Note 1: The value of this bit only affects the transmit properties of the module when the IrDA® encoder is enabled (IREN = 1).

2: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn/RPIn pin. For more information, see **Section 11.4 “Peripheral Pin Select (PPS)”**.

BDs have a fixed relationship to a particular endpoint, depending on the buffering configuration. Table 20-2 provides the mapping of BDs to endpoints. This relationship also means that gaps may occur in the BDT if endpoints are not enabled contiguously. This, theoretically, means that the BDs for disabled endpoints could be used as buffer space. In practice, users should avoid using such spaces in the BDT unless a method of validating BD addresses is implemented.

20.2.1 BUFFER OWNERSHIP

Because the buffers and their BDs are shared between the CPU and the USB module, a simple semaphore mechanism is used to distinguish which is allowed to update the BD and associated buffers in memory. This is done by using the UOWN bit as a semaphore to distinguish which is allowed to update the BD and associated buffers in memory. UOWN is the only bit that is shared between the two configurations of BDnSTAT.

When UOWN is clear, the BD entry is “owned” by the microcontroller core. When the UOWN bit is set, the BD entry and the buffer memory are “owned” by the USB peripheral. The core should not modify the BD or its corresponding data buffer during this time. Note that the microcontroller core can still read BDnSTAT while the SIE owns the buffer and vice versa.

The Buffer Descriptors have a different meaning based on the source of the register update. Register 20-1 and Register 20-2 show the differences in BDnSTAT depending on its current “ownership”.

When UOWN is set, the user can no longer depend on the values that were written to the BDs. From this point, the USB module updates the BDs as necessary, overwriting the original BD values. The BDnSTAT register is updated by the SIE with the token PID and the transfer count is updated.

20.2.2 DMA INTERFACE

The USB OTG module uses a dedicated DMA to access both the BDT and the endpoint data buffers. Since part of the address space of the DMA is dedicated to the Buffer Descriptors, a portion of the memory connected to the DMA must comprise a contiguous address space, properly mapped for the access by the module.

TABLE 20-2: ASSIGNMENT OF BUFFER DESCRIPTORS FOR THE DIFFERENT BUFFERING MODES

Endpoint	BDs Assigned to Endpoint							
	Mode 0 (No Ping-Pong)		Mode 1 (Ping-Pong on EP0 RX)		Mode 2 (Ping-Pong on All EPs)		Mode 3 (Ping-Pong on All Other EPs, Except EP0)	
	RX	TX	RX	TX	RX	TX	RX	TX
0	0	1	0 (E), 1 (O)	2	0 (E), 1 (O)	2 (E), 3 (O)	0	1
1	2	3	3	4	4 (E), 5 (O)	6 (E), 7 (O)	2 (E), 3 (O)	4 (E), 5 (O)
2	4	5	5	6	8 (E), 9 (O)	10 (E), 11 (O)	6 (E), 7 (O)	8 (E), 9 (O)
3	6	7	7	8	12 (E), 13 (O)	14 (E), 15 (O)	10 (E), 11 (O)	12 (E), 13 (O)
4	8	9	9	10	16 (E), 17 (O)	18 (E), 19 (O)	14 (E), 15 (O)	16 (E), 17 (O)
5	10	11	11	12	20 (E), 21 (O)	22 (E), 23 (O)	18 (E), 19 (O)	20 (E), 21 (O)
6	12	13	13	14	24 (E), 25 (O)	26 (E), 27 (O)	22 (E), 23 (O)	24 (E), 25 (O)
7	14	15	15	16	28 (E), 29 (O)	30 (E), 31 (O)	26 (E), 27 (O)	28 (E), 29 (O)
8	16	17	17	18	32 (E), 33 (O)	34 (E), 35 (O)	30 (E), 31 (O)	32 (E), 33 (O)
9	18	19	19	20	36 (E), 37 (O)	38 (E), 39 (O)	34 (E), 35 (O)	36 (E), 37 (O)
10	20	21	21	22	40 (E), 41 (O)	42 (E), 43 (O)	38 (E), 39 (O)	40 (E), 41 (O)
11	22	23	23	24	44 (E), 45 (O)	46 (E), 47 (O)	42 (E), 43 (O)	44 (E), 45 (O)
12	24	25	25	26	48 (E), 49 (O)	50 (E), 51 (O)	46 (E), 47 (O)	48 (E), 49 (O)
13	26	27	27	28	52 (E), 53 (O)	54 (E), 55 (O)	50 (E), 51 (O)	52 (E), 53 (O)
14	28	29	29	30	56 (E), 57 (O)	58 (E), 59 (O)	54 (E), 55 (O)	56 (E), 57 (O)
15	30	31	31	32	60 (E), 61 (O)	62 (E), 63 (O)	58 (E), 59 (O)	60 (E), 61 (O)

Legend: (E) = Even transaction buffer, (O) = Odd transaction buffer

REGISTER 20-20: U1EIE: USB ERROR INTERRUPT ENABLE REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BTSEE	—	DMAEE	BTOEE	DFN8EE	CRC16EE	CRC5EE EOFEE	PIDEE
bit 7							bit 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 15-8 **Unimplemented:** Read as '0'

bit 7 **BTSEE:** Bit Stuff Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 6 **Unimplemented:** Read as '0'

bit 5 **DMAEE:** DMA Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 4 **BTOEE:** Bus Turnaround Time-out Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

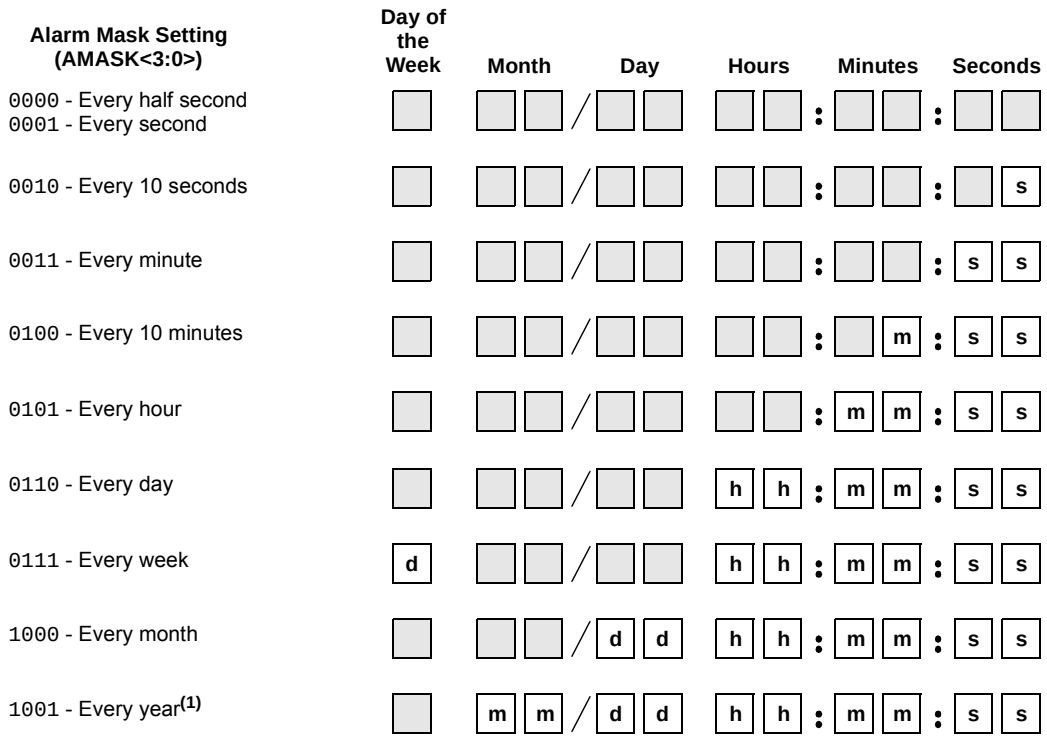
bit 3 **DFN8EE:** Data Field Size Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 2 **CRC16EE:** CRC16 Failure Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 1 For Device mode:
CRC5EE: CRC5 Host Error Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled
For Host mode:
EOFEE: End-of-Frame (EOF) Error interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

bit 0 **PIDEE:** PID Check Failure Interrupt Enable bit
1 = Interrupt is enabled
0 = Interrupt is disabled

FIGURE 22-2: ALARM MASK SETTINGS



Note 1: Annually, except when configured for February 29.

22.6 Power Control

The RTCC includes a power control feature that allows the device to periodically wake-up an external device, wait for the device to be stable before sampling wake-up events from that device and then shut down the external device. This can be done completely autonomously by the RTCC, without the need to wake-up from the current lower power mode.

To use this feature:

1. Enable the RTCC (RTCEN = 1).
2. Set the PWCEN bit (RTCCON1L<10>).
3. Configure the RTCC pin to drive the PWC control signal (RTCOE = 1 and OUTSEL<2:0> = 011).

The polarity of the PWC control signal may be chosen using the PWCPOL bit (RTCCON1L<9>). An active-low or active-high signal may be used with the appropriate external switch to turn on or off the power to one or more external devices. The active-low setting may also be used in conjunction with an open-drain setting on the RTCC pin, in order to drive the ground pin(s) of the external device directly (with the appropriate external VDD pull-up device), without the need for external switches. Finally, the CHIME bit should be set to enable the PWC periodicity.

Once the RTCC and PWC are enabled and running, the PWC logic will generate a control output and a sample gate output. The control output is driven out on the RTCC pin (when RTCOE = 1 and OUTSEL<2:0> = 011) and is used to power up or down the device, as described above.

Once the control output is asserted, the stability window begins, in which the external device is given enough time to power up and provide a stable output.

Once the output is stable, the RTCC provides a sample gate during the sample window. The use of this sample gate depends on the external device being used, but typically, it is used to mask out one or more wake-up signals from the external device.

Finally, both the stability and the sample windows close after the expiration of the sample window and the external device is powered down.

REGISTER 23-3: CRCXORL: CRC XOR POLYNOMIAL REGISTER, LOW BYTE

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
X<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
X<7:1>							—
bit 7							bit 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 15-1 **X<15:1>:** XOR of Polynomial Term x^n Enable bits

bit 0 **Unimplemented:** Read as ‘0’

REGISTER 23-4: CRCXORH: CRC XOR POLYNOMIAL REGISTER, HIGH BYTE

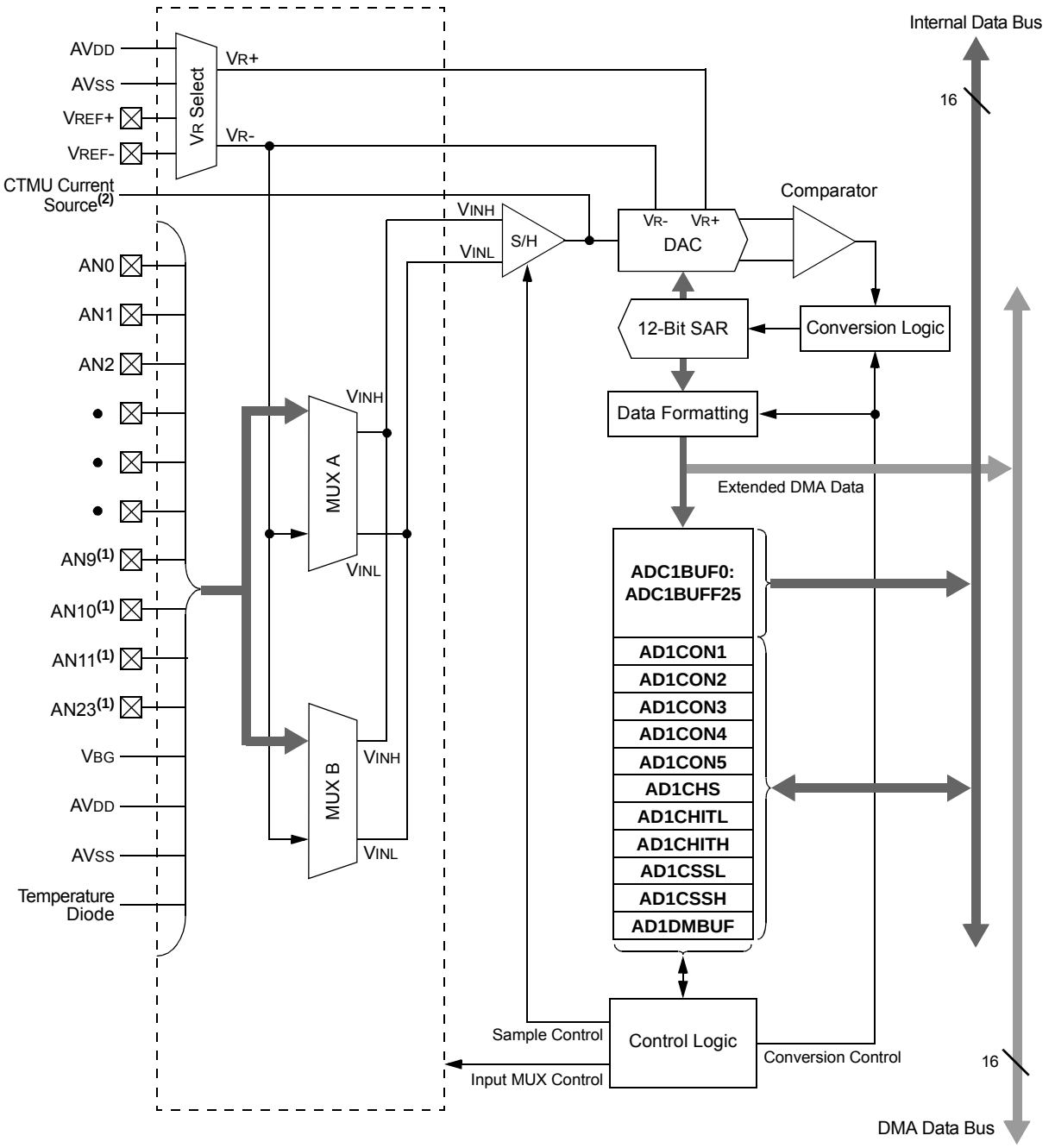
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
X<31:24>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
X<23:16>							
bit 7							bit 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 15-0 **X<31:16>:** XOR of Polynomial Term x^n Enable bits

FIGURE 25-1: 12-BIT A/D CONVERTER BLOCK DIAGRAM (PIC24FJ1024GA610/GB610 FAMILY)

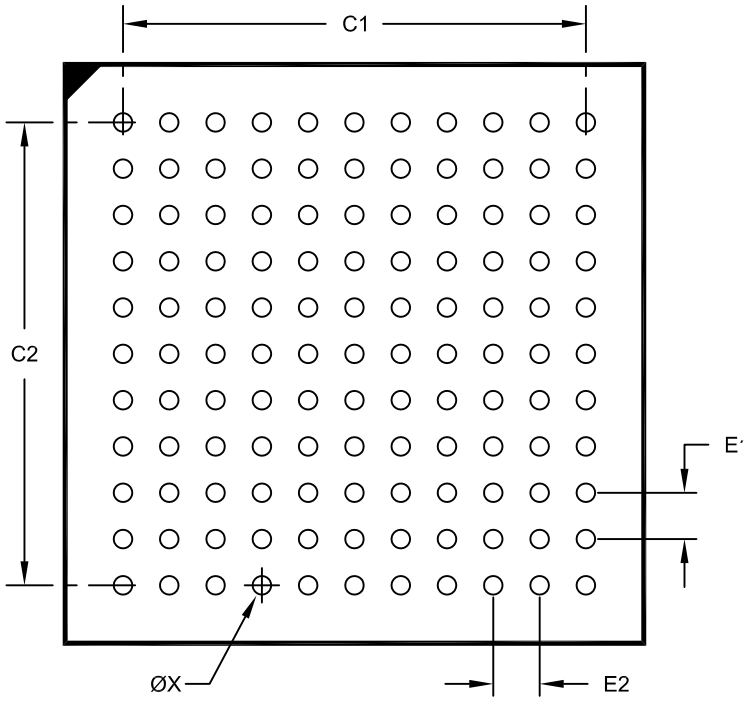


Note 1: AN16 through AN23 are not implemented on 64-pin devices.

2: CTMU current source is routed to the selected ANx pin when SAMP = 1 and TGEN = 0. See **Section 28.0 "Charge Time Measurement Unit (CTMU)"** for details.

121-Lead Plastic Thin Profile Ball Grid Array (BG) - 10x10x1.10 mm Body [TFBGA--Formerly XBGA]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E1		0.80 BSC	
Contact Pitch	E2		0.80 BSC	
Contact Pad Spacing	C1		8.00	
Contact Pad Spacing	C2		8.00	
Contact Pad Diameter (X121)	X			0.32

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2148 Rev D