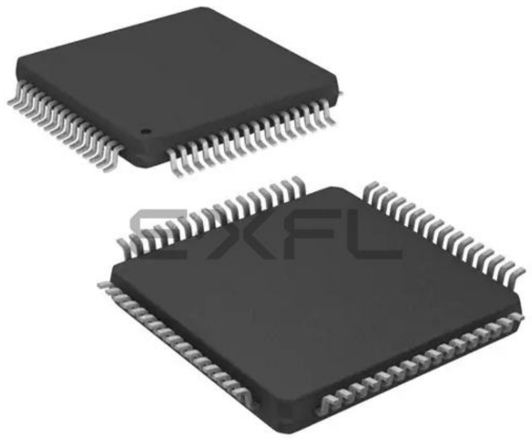


Welcome to E-XFL.COM

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, USB OTG
Peripherals	Brown-out Detect/Reset, DMA, HLVD, POR, PWM, WDT
Number of I/O	53
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 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj1024gb606-i-pt

Peripheral Features

- Peripheral Pin Select (PPS) – Allows Independent I/O Mapping of Many Peripherals
- Up to 5 External Interrupt Sources
- Configurable Interrupt-on-Change on All I/O Pins:
  - Each pin is independently configurable for rising edge or falling edge change detection
- Eight-Channel DMA Supports All Peripheral modules:
  - Minimizes CPU overhead and increases data throughput
- Five 16-Bit Timers/Counters with Prescalers:
  - Can be paired as 32-bit timers/counters
- Six Input Capture modules, Each with a Dedicated 16-Bit Timer
- Six Output Compare/PWM modules, Each with a Dedicated 16-Bit Timer
- Four Single Output CCPs (SCCPs) and Three Multiple Output CCPs (MCCPs):
  - Independent 16/32-bit time base for each module
  - Internal time base and period registers
  - Legacy PIC24F Capture and Compare modes (16 and 32-bit)
  - Special Variable Frequency Pulse and Brushless DC Motor Output modes
- Enhanced Parallel Master/Slave Port (EPMP/EPSP)
- Hardware Real-Time Clock/Calendar (RTCC) with Timestamping
- Three 3-Wire/4-Wire SPI modules:
  - Support 4 Frame modes
  - 8-level FIFO buffer
  - Support I^2S operation
- Three I^2C modules Support Multi-Master/Slave mode and 7-Bit/10-Bit Addressing
- Six UART modules:
  - Support RS-485, RS-232 and LIN/J2602
  - On-chip hardware encoder/decoder for IrDA®
  - Auto-wake-up on Auto-Baud Detect (ABD)
  - 4-level deep FIFO buffer
- Programmable 32-Bit Cyclic Redundancy Check (CRC) Generator
- Four Configurable Logic Cells (CLCs):
  - Two inputs and one output, all mappable to peripherals or I/O pins
  - AND/OR/XOR logic and D/JK flip-flop functions
- High-Current Sink/Source (18 mA/18 mA) on All I/O Pins
- Configurable Open-Drain Outputs on Digital I/O Pins
- 5.5V Tolerant Inputs on Multiple I/O Pins

1.0 DEVICE OVERVIEW

This document contains device-specific information for the following devices:

- PIC24FJ1024GB610
- PIC24FJ512GB610
- PIC24FJ256GB610
- PIC24FJ128GB610
- PIC24FJ1024GB606
- PIC24FJ512GB606
- PIC24FJ256GB606
- PIC24FJ128GB606
- PIC24FJ1024GA610
- PIC24FJ512GA610
- PIC24FJ256GA610
- PIC24FJ128GA610
- PIC24FJ1024GA606
- PIC24FJ512GA606
- PIC24FJ256GA606
- PIC24FJ128GA606

The PIC24FJ1024GA610/GB610 family introduces many new analog features to the extreme low-power Microchip devices. This is a 16-bit microcontroller family with a broad peripheral feature set and enhanced computational performance. This family also offers a new migration option for those high-performance applications which may be outgrowing their 8-bit platforms, but do not require the numerical processing power of a Digital Signal Processor (DSP).

Table 1-3 lists the functions of the various pins shown in the pinout diagrams.

1.1 Core Features

1.1.1 16-BIT ARCHITECTURE

Central to all PIC24F devices is the 16-bit modified Harvard architecture, first introduced with Microchip's dsPIC® Digital Signal Controllers (DSCs). The PIC24F CPU core offers a wide range of enhancements, such as:

- 16-bit data and 24-bit address paths with the ability to move information between data and memory spaces
- Linear addressing of up to 12 Mbytes (program space) and 32 Kbytes (data)
- A 16-element Working register array with built-in software stack support
- A 17 x 17 hardware multiplier with support for integer math
- Hardware support for 32 by 16-bit division
- An instruction set that supports multiple addressing modes and is optimized for high-level languages, such as 'C'
- Operational performance up to 16 MIPS

1.1.2 POWER-SAVING TECHNOLOGY

The PIC24FJ1024GA610/GB610 family of devices includes Retention Sleep, a low-power mode with essential circuits being powered from a separate low-voltage regulator.

This new low-power mode also supports the continuous operation of the low-power, on-chip Real-Time Clock/ Calendar (RTCC), making it possible for an application to keep time while the device is otherwise asleep.

Aside from this new feature, PIC24FJ1024GA610/GB610 family devices also include all of the legacy power-saving features of previous PIC24F microcontrollers, such as:

- On-the-Fly Clock Switching, allowing the selection of a lower power clock during run time
- Doze Mode Operation, for maintaining peripheral clock speed while slowing the CPU clock
- Instruction-Based Power-Saving Modes, for quick invocation of the Idle and the Sleep modes

1.1.3 OSCILLATOR OPTIONS AND FEATURES

All of the devices in the PIC24FJ1024GA610/GB610 family offer six different oscillator options, allowing users a range of choices in developing application hardware. These include:

- Two Crystal modes
- Two External Clock (EC) modes
- A Phase-Locked Loop (PLL) frequency multiplier, which allows clock speeds of up to 32 MHz
- A Digitally Controlled Oscillator (DCO) with multiple frequencies and fast wake-up time
- A Fast Internal Oscillator (FRC), a nominal 8 MHz output, with multiple frequency divider options
- A separate Low-Power Internal RC Oscillator (LPRC), 31 kHz nominal, for low-power, timing-insensitive applications.

The internal oscillator block also provides a stable reference source for the Fail-Safe Clock Monitor (FSCM). This option constantly monitors the main clock source against a reference signal provided by the internal oscillator and enables the controller to switch to the internal oscillator, allowing for continued low-speed operation or a safe application shutdown.

1.1.4 EASY MIGRATION

Regardless of the memory size, all devices share the same rich set of peripherals, allowing for a smooth migration path as applications grow and evolve. The consistent pinout scheme used throughout the entire family also aids in migrating from one device to the next larger device, or even in jumping from 64-pin to 100-pin devices.

The PIC24F family is pin-compatible with devices in the dsPIC33 family, and shares some compatibility with the pinout schema for PIC18 and dsPIC30. This extends the ability of applications to grow from the relatively simple, to the powerful and complex, yet still selecting a Microchip device.

TABLE 1-3: PIC24FJ1024GA610/GB610 FAMILY PINOUT DESCRIPTIONS (CONTINUED)

Pin Function	Pin Number/Grid Locator						I/O	Input Buffer	Description
	GA606 64-Pin QFN/TQFP/ QFP	GB606 64-Pin QFN/ TQFP/QFP	GA610 100-Pin TQFP/ QFP	GB610 100-Pin TQFP/ QFP	GA612 121-Pin BGA	GB612 121-Pin BGA			
RA0	—	—	17	17	G3	G3	I/O	DIG/ST	PORTA Digital I/Os
RA1	—	—	38	38	J6	J6	I/O	DIG/ST	
RA2	—	—	58	58	H11	H11	I/O	DIG/ST	
RA3	—	—	59	59	G10	G10	I/O	DIG/ST	
RA4	—	—	60	60	G11	G11	I/O	DIG/ST	
RA5	—	—	61	61	G9	G9	I/O	DIG/ST	
RA6	—	—	91	91	C5	C5	I/O	DIG/ST	
RA7	—	—	92	92	B5	B5	I/O	DIG/ST	
RA9	—	—	28	28	L2	L2	I/O	DIG/ST	
RA10	—	—	29	29	K3	K3	I/O	DIG/ST	
RA14	—	—	66	66	E11	E11	I/O	DIG/ST	
RA15	—	—	67	67	E8	E8	I/O	DIG/ST	
RB0	16	16	25	25	K2	K2	I/O	DIG/ST	PORTB Digital I/Os
RB1	15	15	24	24	K1	K1	I/O	DIG/ST	
RB2	14	14	23	23	J2	J2	I/O	DIG/ST	
RB3	13	13	22	22	J1	J1	I/O	DIG/ST	
RB4	12	12	21	21	H2	H2	I/O	DIG/ST	
RB5	11	11	20	20	H1	H1	I/O	DIG/ST	
RB6	17	17	26	26	L1	L1	I/O	DIG/ST	
RB7	18	18	27	27	J3	J3	I/O	DIG/ST	
RB8	21	21	32	32	K4	K4	I/O	DIG/ST	
RB9	22	22	33	33	L4	L4	I/O	DIG/ST	
RB10	23	23	34	34	L5	L5	I/O	DIG/ST	
RB11	24	24	35	35	J5	J5	I/O	DIG/ST	
RB12	27	27	41	41	J7	J7	I/O	DIG/ST	
RB13	28	28	42	42	L7	L7	I/O	DIG/ST	
RB14	29	29	43	43	K7	K7	I/O	DIG/ST	
RB15	30	30	44	44	L8	L8	I/O	DIG/ST	
RC1	—	—	6	6	D1	D1	I/O	DIG/ST	PORTC Digital I/Os
RC2	—	—	7	7	E4	E4	I/O	DIG/ST	
RC3	—	—	8	8	E2	E2	I/O	DIG/ST	
RC4	—	—	9	9	E1	E1	I/O	DIG/ST	
RC12	39	39	63	63	F9	F9	I/O	DIG/ST	
RC13	47	47	73	73	C10	C10	I/O	DIG/ST	
RC14	48	48	74	74	B11	B11	I/O	DIG/ST	
RC15	40	40	64	64	F11	F11	I/O	DIG/ST	

Legend: TTL = TTL input buffer; ST = Schmitt Trigger input buffer
ANA = Analog level input/output; I^2C = I^2C/SMBus input buffer
DIG = Digital input/output; XCVR = Dedicated Transceiver

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

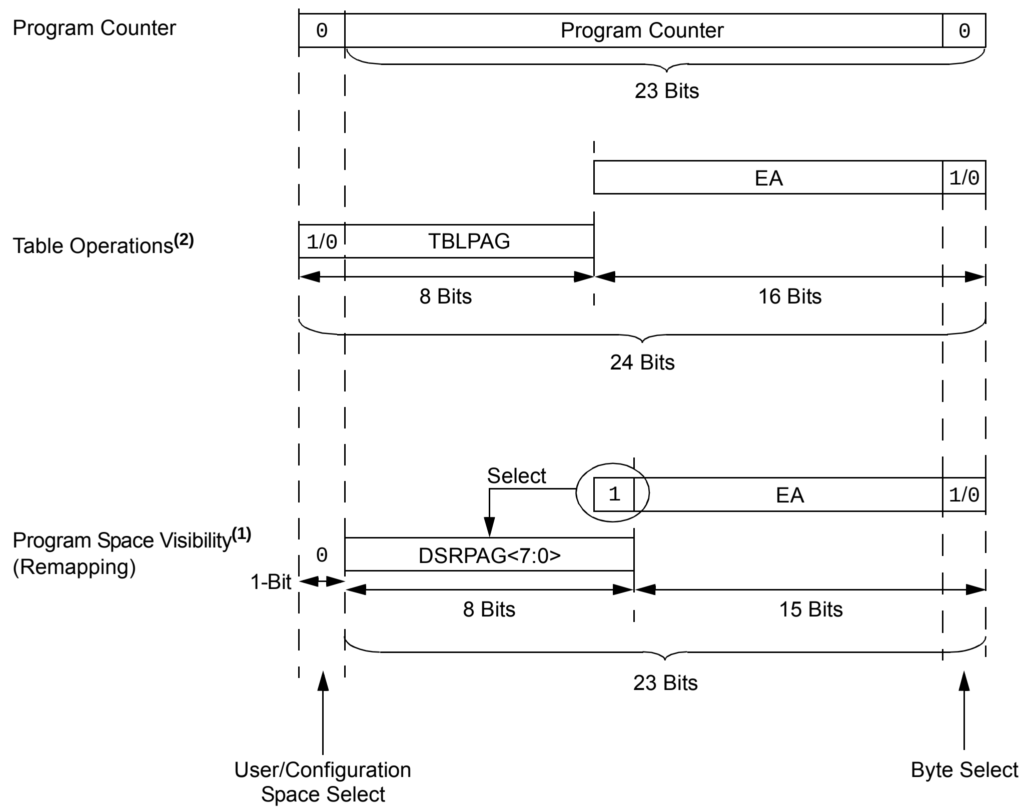
The program address memory space of the PIC24FJ1024GA610/GB610 family devices is 4M instructions. The space is addressable by a 24-bit value derived from either the 23-bit Program Counter (PC) during program execution, or from table operation or Data Space remapping, as described in **Section 4.3 "Interfacing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

FIGURE 4-8: DATA ACCESS FROM PROGRAM SPACE ADDRESS GENERATION



Note 1: DSRPAG<8> acts as word select. DSRPAG<9> should always be '1' to map program memory to data memory.

2: The instructions, `TBLRDH/TBLWTH/TBLRDL/TBLWTL`, decide if the higher or lower word of program memory is accessed. `TBLRDH/TBLWTH` instructions access the higher word and `TBLRDL/TBLWTL` instructions access the lower word. Table Read operations are permitted in the configuration memory space.

5.0 DIRECT MEMORY ACCESS CONTROLLER (DMA)

Note:	This data sheet summarizes the features of the PIC24FJ1024GA610/GB610 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the *"dsPIC33/PIC24 Family Reference Manual"*, **"Direct Memory Access Controller (DMA)"** (DS39742), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

The Direct Memory Access Controller (DMA) is designed to service high-throughput data peripherals operating on the SFR bus, allowing them to access data memory directly and alleviating the need for CPU intensive management. By allowing these data intensive peripherals to share their own data path, the main data bus is also deloaded, resulting in additional power savings.

The DMA Controller functions both as a peripheral and a direct extension of the CPU. It is located on the microcontroller data bus between the CPU and DMA-enabled peripherals, with direct access to SRAM. This partitions the SFR bus into two buses, allowing the DMA Controller access to the DMA capable peripherals located on the new DMA SFR bus. The controller serves as a master device on the DMA SFR bus, controlling data flow from DMA capable peripherals.

The controller also monitors CPU instruction processing directly, allowing it to be aware of when the CPU requires access to peripherals on the DMA bus and automatically relinquishing control to the CPU as needed. This increases the effective bandwidth for handling data without DMA operations causing a processor stall. This makes the controller essentially transparent to the user.

The DMA Controller has these features:

- Eight Multiple Independent and Independently Programmable Channels
- Concurrent Operation with the CPU (no DMA caused Wait states)
- DMA Bus Arbitration
- Five Programmable Address modes
- Four Programmable Transfer modes
- Four Flexible Internal Data Transfer modes
- Byte or Word Support for Data Transfer
- 16-Bit Source and Destination Address Register for Each Channel, Dynamically Updated and Reloadable
- 16-Bit Transaction Count Register, Dynamically Updated and Reloadable
- Upper and Lower Address Limit Registers
- Counter Half-Full Level Interrupt
- Software Triggered Transfer
- Null Write mode for Symmetric Buffer Operations

A simplified block diagram of the DMA Controller is shown in Figure 5-1.

FIGURE 5-1: DMA FUNCTIONAL BLOCK DIAGRAM

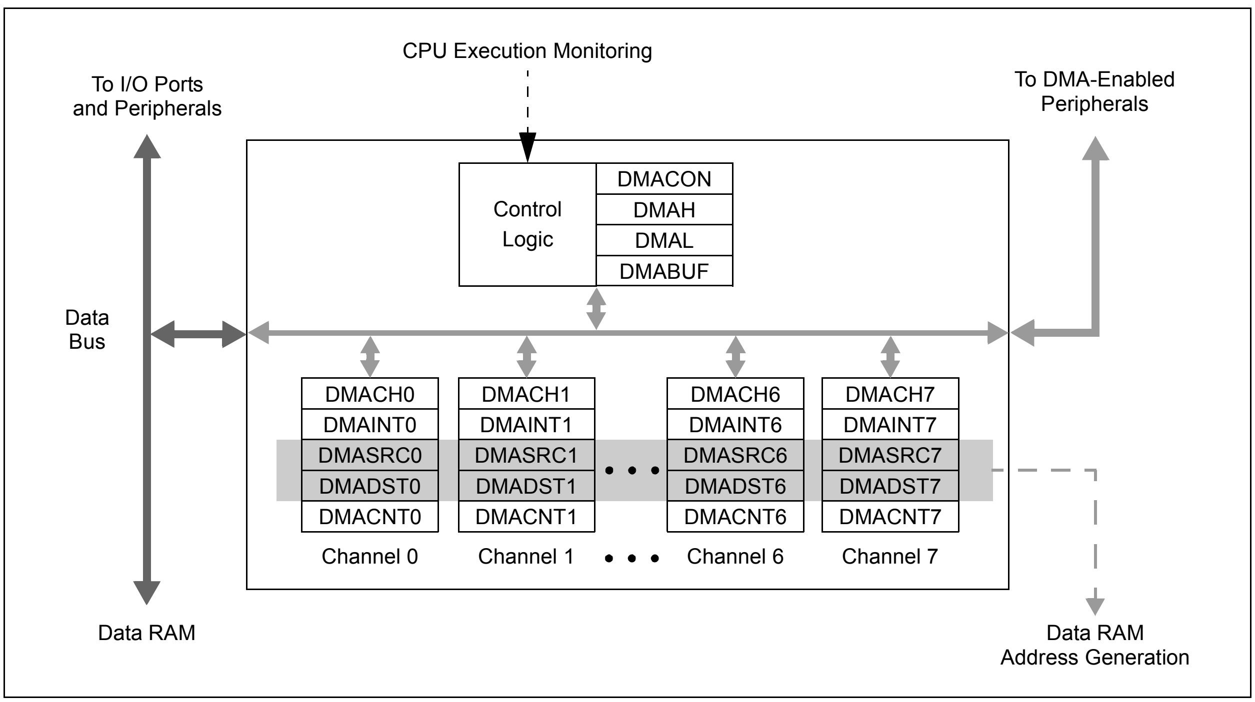
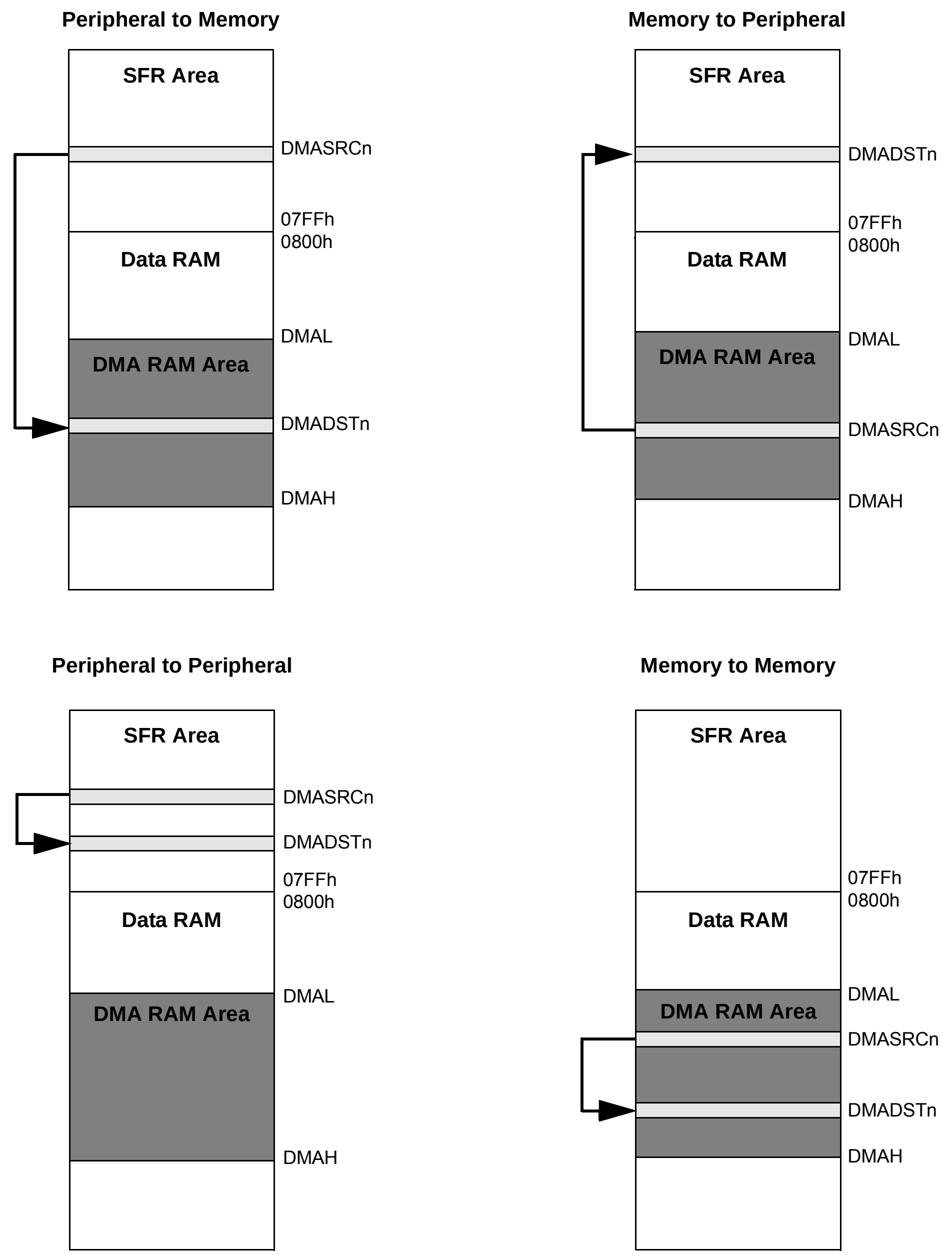


FIGURE 5-2: TYPES OF DMA DATA TRANSFERS



Note: Relative sizes of memory areas are not shown to scale.

REGISTER 8-4: INTCON2: INTERRUPT CONTROL REGISTER 2

R/W-1	R-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
GIE	DISI	SWTRAP	—	—	—	—	AIVTEN
bit 15							bit 8

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
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	**GIE:** Global Interrupt Enable bit 1 = Interrupts and associated interrupt enable bits are enabled 0 = Interrupts are disabled, but traps are still enabled
bit 14	**DISI:** DISI Instruction Status bit 1 = DISI instruction is active 0 = DISI instruction is not active
bit 13	**SWTRAP:** Software Trap Status bit 1 = Software trap is enabled 0 = Software trap is disabled
bit 12-9	**Unimplemented:** Read as ‘0’
bit 8	**AIVTEN:** Alternate Interrupt Vector Table Enable bit 1 = Use Alternate Interrupt Vector Table (if enabled in Configuration bits) 0 = Use standard Interrupt Vector Table (default)
bit 7-5	**Unimplemented:** Read as ‘0’
bit 4	**INT4EP:** External Interrupt 4 Edge Detect Polarity Select bit 1 = Interrupt on negative edge 0 = Interrupt on positive edge
bit 3	**INT3EP:** External Interrupt 3 Edge Detect Polarity Select bit 1 = Interrupt on negative edge 0 = Interrupt on positive edge
bit 2	**INT2EP:** External Interrupt 2 Edge Detect Polarity Select bit 1 = Interrupt on negative edge 0 = Interrupt on positive edge
bit 1	**INT1EP:** External Interrupt 1 Edge Detect Polarity Select bit 1 = Interrupt on negative edge 0 = Interrupt on positive edge
bit 0	**INT0EP:** External Interrupt 0 Edge Detect Polarity Select bit 1 = Interrupt on negative edge 0 = Interrupt on positive edge

9.4 Clock Switching Operation

With few limitations, applications are free to switch between any of the five clock sources (POSC, SOSC, FRC, DCO and LPRC) under software control and at any time. To limit the possible side effects that could result from this flexibility, PIC24F devices have a safeguard lock built into the switching process.

Note:	The Primary Oscillator mode has three different submodes (XT, HS and EC), which are determined by the POSCMD<1:0> Configuration bits. While an application can switch to and from Primary Oscillator mode in software, it cannot switch between the different primary submodes without reprogramming the device.

9.4.1 ENABLING CLOCK SWITCHING

To enable clock switching, the FCKSM<1> Configuration bit in FOSC must be programmed to ‘0’. (Refer to **Section 30.1 “Configuration Bits”** for further details.) If the FCKSM<1> Configuration bit is unprogrammed (‘1’), the clock switching function and Fail-Safe Clock Monitor function are disabled; this is the default setting.

The NOSC<2:0> control bits (OSCCON<10:8>) do not control the clock selection when clock switching is disabled. However, the COSC<2:0> bits (OSCCON<14:12>) will reflect the clock source selected by the FNOSC<2:0> Configuration bits.

The OSWEN control bit (OSCCON<0>) has no effect when clock switching is disabled; it is held at ‘0’ at all times.

9.4.2 OSCILLATOR SWITCHING SEQUENCE

At a minimum, performing a clock switch requires this basic sequence:

1. If desired, read the COSC<2:0> bits (OSCCON<14:12>) to determine the current oscillator source.
2. Perform the unlock sequence to allow a write to the OSCCON register high byte.
3. Write the appropriate value to the NOSC<2:0> bits (OSCCON<10:8>) for the new oscillator source.
4. Perform the unlock sequence to allow a write to the OSCCON register low byte.
5. Set the OSWEN bit to initiate the oscillator switch.

Once the basic sequence is completed, the system clock hardware responds automatically as follows:

1. The clock switching hardware compares the COSC<2:0> bits with the new value of the NOSC<2:0> bits. If they are the same, then the clock switch is a redundant operation. In this case, the OSWEN bit is cleared automatically and the clock switch is aborted.
2. If a valid clock switch has been initiated, the LOCK (OSCCON<5>) and CF (OSCCON<3>) bits are cleared.
3. The new oscillator is turned on by the hardware if it is not currently running. If a crystal oscillator must be turned on, the hardware will wait until the OST expires. If the new source is using the PLL, then the hardware waits until a PLL lock is detected (LOCK = 1).
4. The hardware waits for 10 clock cycles from the new clock source and then performs the clock switch.
5. The hardware clears the OSWEN bit to indicate a successful clock transition. In addition, the NOSC<2:0> bits values are transferred to the COSC<2:0> bits.
6. The old clock source is turned off at this time, with the exception of LPRC (if WDT or FSCM is enabled) or SOSC (if SOSCEN remains set).

Note 1:	The processor will continue to execute code throughout the clock switching sequence. Timing-sensitive code should not be executed during this time.
2:	Direct clock switches between any Primary Oscillator mode with PLL and FRCPLL mode are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transitional clock source between the two PLL modes.

REGISTER 9-9: REFOCONH: REFERENCE OSCILLATOR CONTROL REGISTER HIGH

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>						
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
RODIV<7:0>							
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 **Unimplemented:** Read as ‘0’

bit 14-0 **RODIV<14:0>:** Reference Clock Divider bits

Specifies 1/2 period of the reference clock in the source clocks
(ex: Period of Output = [Reference Source * 2] * RODIV<14:0>; this equation does not apply to RODIV<14:0> = 0).
111111111111111 = REFO clock is the base clock frequency divided by 65,534 (32,767 * 2)
111111111111110 = REFO clock is the base clock frequency divided by 65,532 (32,766 * 2)
•
•
•
000000000000011 = REFO clock is the base clock frequency divided by 6 (3 * 2)
000000000000010 = REFO clock is the base clock frequency divided by 4 (2 * 2)
000000000000001 = REFO clock is the base clock frequency divided by 2 (1 * 2)
000000000000000 = REFO clock is the same frequency as the base clock (no divider)

REGISTER 9-10: REFOTRIML: REFERENCE OSCILLATOR TRIM REGISTER LOW

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<0:7>							
bit 15							bit 8

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
ROTRIM8	—	—	—	—	—	—	—
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 15-7 **ROTRIM<0:8>:** REFO Trim bits

These bits provide a fractional additive to the RODIV<14:0> value for the 1/2 period of the REFO clock.
`000000000` = 0/512 (0.0 divisor added to the RODIV<14:0> value)
`000000001` = 1/512 (0.001953125 divisor added to the RODIV<14:0> value)
`000000010` = 2/512 (0.00390625 divisor added to the RODIV<14:0> value)
•
•
•
`100000000` = 256/512 (0.5000 divisor added to the RODIV<14:0> value)
•
•
•
`111111110` = 510/512 (0.99609375 divisor added to the RODIV<14:0> value)
`111111111` = 511/512 (0.998046875 divisor added to the RODIV<14:0> value)

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

10.2.2 IDLE MODE

Idle mode has these features:

- The CPU will stop executing instructions.
- The WDT is automatically cleared.
- The system clock source remains active. By default, all peripheral modules continue to operate normally from the system clock source, but can also be selectively disabled (see **Section 10.4 “Selective Peripheral Module Control”**).
- If the WDT or FSCM is enabled, the LPRC will also remain active.

The device will wake from Idle mode on any of these events:

- Any interrupt that is individually enabled.
- Any device Reset.
- A WDT time-out.

On wake-up from Idle, the clock is reapplied to the CPU and instruction execution begins immediately, starting with the instruction following the `PWRSAV` instruction or the first instruction in the ISR.

10.2.3 INTERRUPTS COINCIDENT WITH POWER SAVE INSTRUCTIONS

Any interrupt that coincides with the execution of a `PWRSAV` instruction will be held off until entry into Sleep or Idle mode has completed. The device will then wake-up from Sleep or Idle mode.

10.2.4 LOW-VOLTAGE RETENTION REGULATOR

PIC24FJ1024GA610/GB610 family devices incorporate a second on-chip voltage regulator, designed to provide power to select microcontroller features at 1.2V nominal. This regulator allows features, such as data RAM and the WDT, to be maintained in power-saving modes where they would otherwise be inactive, or maintain them at a lower power than would otherwise be the case.

Retention Sleep uses less power than standard Sleep mode, but takes more time to recover and begin execution. An additional 20-35 µS (typical) is required to charge VCAP from 1.2V to 1.8V and start to execute instructions when exiting Retention Sleep.

The VREGS bit allows the control of speed to exit from the Sleep modes (regular and Retention) at the cost of more power. The regulator band gaps are enabled when VREGS = `1`, which increases the current but reduces time to recover from Sleep by ~10 µS.

The low-voltage retention regulator is only available when Sleep mode is invoked. It is controlled by the $\overline{\text{LPCFG}}$ Configuration bit (FPOR<2>) and in firmware by the RETEN bit (RCON<12>). $\overline{\text{LPCFG}}$ must be programmed (= `0`) and the RETEN bit must be set (= `1`) for the regulator to be enabled.

10.2.5 EXITING FROM LOW-VOLTAGE RETENTION SLEEP

All of the standard methods for exiting from standard Sleep also apply to Retention Sleep ($\overline{\text{MCLR}}$, INT0, etc.). However, in order to allow the regulator to switch from 1.8V (operating) to Retention mode (1.2V), there is a hardware ‘lockout timer’ from the execution of Retention Sleep until Retention Sleep can be exited. During the ‘lockout time’, the only method to exit Retention Sleep is a POR or $\overline{\text{MCLR}}$. Interrupts that are asserted (such as INT0) during the ‘lockout time’ are masked. The lockout timer then sets a minimum interval from when the part enters Retention Sleep until it can exit from Retention Sleep. Interrupts are not ‘held pending’ during lockout; they are masked and in order to exit after the lockout expires, the exiting source must assert after the lockout time.

The lockout timer is derived from the LPRC clock, which has a wide (untrimmed) frequency tolerance. The lockout time will be one of the following two cases:

- If the LPRC was not running at the time of Retention Sleep, the lockout time is 2 LPRC periods + LPRC wake-up time
- If the LPRC was running at the time of Retention Sleep, the lockout time is 1 LPRC period

Refer to Table 33-20 and Table 33-21 in the AC Electrical Specifications for the LPRC timing.

10.2.6 SUMMARY OF LOW-POWER SLEEP MODES

The RETEN bit and the VREGS bit (RCON<8>) allow for four different Sleep modes, which will vary by wake-up time and power consumption. Refer to Table 10-1 for a summary of these modes. Specific information about the current consumption and wake times can be found in **Section 33.0 “Electrical Characteristics”**.

TABLE 10-1: LOW-POWER SLEEP MODES

RETEN	VREGS	Mode	Relative Power (1 = Lowest)
0	0	Sleep	3
0	1	Fast Wake-up	4
1	0	Retention Sleep	1
1	1	Fast Retention	2

10.3 Doze Mode

Generally, changing clock speed and invoking one of the power-saving modes are the preferred strategies for reducing power consumption. There may be circumstances, however, where this is not practical. For example, it may be necessary for an application to maintain uninterrupted synchronous communication, even while it is doing nothing else. Reducing system clock speed may introduce communication errors, while using a power-saving mode may stop communications completely.

Doze mode is a simple and effective alternative method to reduce power consumption while the device is still executing code. In this mode, the system clock continues to operate from the same source and at the same speed. Peripheral modules continue to be clocked at the same speed while the CPU clock speed is reduced. Synchronization between the two clock domains is maintained, allowing the peripherals to access the SFRs while the CPU executes code at a slower rate.

Doze mode is enabled by setting the DOZEN bit (CLKDIV<11>). The ratio between peripheral and core clock speed is determined by the DOZE<2:0> bits (CLKDIV<14:12>). There are eight possible configurations, from 1:1 to 1:256, with 1:1 being the default.

It is also possible to use Doze mode to selectively reduce power consumption in event driven applications. This allows clock-sensitive functions, such as synchronous communications, to continue without interruption while the CPU Idles, waiting for something to invoke an interrupt routine. Enabling the automatic return to full-speed CPU operation on interrupts is enabled by setting the ROI bit (CLKDIV<15>). By default, interrupt events have no effect on Doze mode operation.

10.4 Selective Peripheral Module Control

Idle and Doze modes allow users to substantially reduce power consumption by slowing or stopping the CPU clock. Even so, peripheral modules still remain clocked, and thus, consume power. There may be cases where the application needs what these modes do not provide: the allocation of power resources to CPU processing with minimal power consumption from the peripherals.

PIC24F devices address this requirement by allowing peripheral modules to be selectively disabled, reducing or eliminating their power consumption. This can be done with two control bits:

- The Peripheral Enable bit, generically named, “XXXEN”, located in the module’s main control SFR.
- The Peripheral Module Disable (PMD) bit, generically named, “XXXMD”, located in one of the PMD Control registers.

Both bits have similar functions in enabling or disabling their associated module. Setting the PMD bit for a module disables all clock sources to that module, reducing its power consumption to an absolute minimum. In this state, the control and status registers associated with the peripheral will also be disabled, so writes to those registers will have no effect and read values will be invalid. Many peripheral modules have a corresponding PMD bit.

In contrast, disabling a module by clearing its XXXEN bit disables its functionality, but leaves its registers available to be read and written to. This reduces power consumption, but not by as much as setting the PMD bit does. Most peripheral modules have an enable bit; exceptions include input capture, output compare and RTCC.

To achieve more selective power savings, peripheral modules can also be selectively disabled when the device enters Idle mode. This is done through the control bit of the generic name format, “XXXIDL”. By default, all modules that can operate during Idle mode will do so. Using the disable on Idle feature allows further reduction of power consumption during Idle mode, enhancing power savings for extremely critical power applications.

REGISTER 11-26: RPINR18: PERIPHERAL PIN SELECT INPUT REGISTER 18

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	U1CTSR5	U1CTSR4	U1CTSR3	U1CTSR2	U1CTSR1	U1CTSR0
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
—	—	U1RXR5	U1RXR4	U1RXR3	U1RXR2	U1RXR1	U1RXR0
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 **U1CTSR<5:0>:** Assign UART1 Clear-to-Send ($\overline{\text{U1CTS}}$) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **U1RXR<5:0>:** Assign UART1 Receive (U1RX) to Corresponding RPn or RPIn Pin bits

REGISTER 11-27: RPINR19: PERIPHERAL PIN SELECT INPUT REGISTER 19

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	U2CTSR5	U2CTSR4	U2CTSR3	U2CTSR2	U2CTSR1	U2CTSR0
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
—	—	U2RXR5	U2RXR4	U2RXR3	U2RXR2	U2RXR1	U2RXR0
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 **U2CTSR<5:0>:** Assign UART2 Clear-to-Send ($\overline{\text{U2CTS}}$) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **U2RXR<5:0>:** Assign UART2 Receive (U2RX) to Corresponding RPn or RPIn Pin bits

15.0 OUTPUT COMPARE WITH DEDICATED TIMERS

Note:	This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"dsPIC33/PIC24 Family Reference Manual",* **"Output Compare with Dedicated Timer"** (DS70005159), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

All devices in the PIC24FJ1024GA610/GB610 family feature six independent output compare modules. Each of these modules offers a wide range of configuration and operating options for generating pulse trains on internal device events, and can produce Pulse-Width Modulated (PWM) waveforms for driving power applications.

Key features of the output compare module include:

- Hardware-Configurable for 32-Bit Operation in all modes by Cascading Two Adjacent modules
- Synchronous and Trigger modes of Output Compare Operation with up to 31 User-Selectable Sync/Trigger Sources Available
- Two Separate Period registers (a main register, OCxR, and a secondary register, OCxRS) for Greater Flexibility in Generating Pulses of Varying Widths
- Configurable for Single Pulse or Continuous Pulse Generation on an Output Event or Continuous PWM Waveform Generation
- Up to 6 cLock Sources Available for each module, Driving a Separate Internal 16-Bit Counter

15.1 General Operating Modes

15.1.1 SYNCHRONOUS AND TRIGGER MODES

When the output compare module operates in a Free-Running mode, the internal 16-bit counter, OCxTMR, runs counts up continuously, wrapping around from 0xFFFF to 0x0000 on each overflow. Its period is synchronized to the selected external clock source. Compare or PWM events are generated each time a match between the internal counter and one of the Period registers occurs.

In Synchronous mode, the module begins performing its compare or PWM operation as soon as its selected clock source is enabled. Whenever an event occurs on the selected Sync source, the module's internal counter is reset. In Trigger mode, the module waits for a Sync event from another internal module to occur before allowing the counter to run.

Free-Running mode is selected by default or any time that the SYNCSEL<4:0> bits (OCxCON2<4:0>) are set to '00000'. Synchronous or Trigger modes are selected any time the SYNCSELx bits are set to any value except '00000'. The OCTRIG bit (OCxCON2<7>) selects either Synchronous or Trigger mode; setting the bit selects Trigger mode operation. In both modes, the SYNCSELx bits determine the Sync/Trigger source.

15.1.2 CASCADED (32-BIT) MODE

By default, each module operates independently with its own set of 16-Bit Timer and Duty Cycle registers. To increase resolution, adjacent even and odd modules can be configured to function as a single 32-bit module. (For example, Modules 1 and 2 are paired, as are Modules 3 and 4, and so on.) The odd numbered module (OCx) provides the Least Significant 16 bits of the 32-bit register pairs and the even numbered module (OCy) provides the Most Significant 16 bits. Wrap-arounds of the OCx registers cause an increment of their corresponding OCy registers.

Cascaded operation is configured in hardware by setting the OC32 bit (OCxCON2<8>) for both modules. For more details on cascading, refer to the *"dsPIC33/PIC24 Family Reference Manual"*, **"Output Compare with Dedicated Timer"** (DS70005159).

REGISTER 17-11: SPIxURDTL: SPIx UNDERRUN DATA REGISTER LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
URDATA<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	R/W-0
URDATA<7:0>							
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 **URDATA<15:0>:** SPIx Underrun Data bits

These bits are only used when URDTEN = 1. This register holds the data to transmit when a Transmit Underrun condition occurs.

When the MODE<32,16> or WLENGTH<4:0> bits select 16 to 9-bit data, the SPIx only uses URDATA<15:0>. When the MODE<32,16> or WLENGTH<4:0> bits select 8 to 2-bit data, the SPIx only uses URDATA<7:0>.

REGISTER 17-12: SPIxURDTH: SPIx UNDERRUN DATA REGISTER HIGH

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
URDATA<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
URDATA<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 **URDATA<31:16>:** SPIx Underrun Data bits

These bits are only used when URDTEN = 1. This register holds the data to transmit when a Transmit Underrun condition occurs.

When the MODE<32,16> or WLENGTH<4:0> bits select 32 to 25-bit data, the SPIx only uses URDATA<15:0>. When the MODE<32,16> or WLENGTH<4:0> bits select 24 to 17-bit data, the SPIx only uses URDATA<7:0>.

REGISTER 20-6: U1STAT: USB STATUS REGISTER

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

R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	U-0	U-0
ENDPT3	ENDPT2	ENDPT1	ENDPT0	DIR	PPBI(1)	—	—
bit 7							bit 0

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-4 **ENDPT<3:0>:** Number of the Last Endpoint Activity bits
(Represents the number of the BDT updated by the last USB transfer.)
1111 = Endpoint 15
1110 = Endpoint 14
•
•
•
0001 = Endpoint 1
0000 = Endpoint 0

bit 3 **DIR:** Last BD Direction Indicator bit
1 = The last transaction was a transmit transfer (TX)
0 = The last transaction was a receive transfer (RX)

bit 2 **PPBI:** Ping-Pong BD Pointer Indicator bit(1)
1 = The last transaction was to the odd BD bank
0 = The last transaction was to the even BD bank

bit 1-0 **Unimplemented:** Read as ‘0’

Note 1: This bit is only valid for endpoints with available even and odd BD registers.

REGISTER 21-2: PMCON2: EPMP CONTROL REGISTER 2

R-0, HSC	U-0	R/C-0, HS	R/C-0, HS	U-0	U-0	U-0	U-0
BUSY	—	ERROR	TIMEOUT	—	—	—	—
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
RADDR23(1)	RADDR22(1)	RADDR21(1)	RADDR20(1)	RADDR19(1)	RADDR18(1)	RADDR17(1)	RADDR16(1)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown
C = Clearable bit	HS = Hardware Settable bit	HSC = Hardware Settable/Clearable bit	

bit 15 **BUSY:** Busy bit (Master mode only)
1 = Port is busy
0 = Port is not busy

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **ERROR:** Error bit
1 = Transaction error (illegal transaction was requested)
0 = Transaction completed successfully

bit 12 **TIMEOUT:** Time-out bit
1 = Transaction timed out
0 = Transaction completed successfully

bit 11-8 **Unimplemented:** Read as ‘0’

bit 7-0 **RADDR<23:16>:** Parallel Master Port Reserved Address Space bits(1)

Note 1: If RADDR<23:16> = 00000000, then the last EDS address for Chip Select 2 will be FFFFFFh.

REGISTER 21-8: PMSTAT: EPMP STATUS REGISTER (SLAVE MODE ONLY)

R-0, HSC	R/W-0, HS	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
IBF	IBOV	—	—	IB3F(1)	IB2F(1)	IB1F(1)	IB0F(1)
bit 15							bit 8

R-1, HSC	R/W-0, HS	U-0	U-0	R-1, HSC	R-1, HSC	R-1, HSC	R-1, HSC
OBE	OBUF	—	—	OB3E	OB2E	OB1E	OB0E
bit 7							bit 0

Legend:	HS = Hardware Settable bit	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 **IBF:** Input Buffer Full Status bit
1 = All writable Input Buffer registers are full
0 = Some or all of the writable Input Buffer registers are empty

bit 14 **IBOV:** Input Buffer Overflow Status bit
1 = A write attempt to a full Input register occurred (must be cleared in software)
0 = No overflow occurred

bit 13-12 **Unimplemented:** Read as ‘0’

bit 11-8 **IB3F:IB0F:** Input Buffer x Status Full bits(1)
1 = Input buffer contains unread data (reading the buffer will clear this bit)
0 = Input buffer does not contain unread data

bit 7 **OBE:** Output Buffer Empty Status bit
1 = All readable Output Buffer registers are empty
0 = Some or all of the readable Output Buffer registers are full

bit 6 **OBUF:** Output Buffer Underflow Status bit
1 = A read occurred from an empty Output Buffer register (must be cleared in software)
0 = No underflow occurred

bit 5-4 **Unimplemented:** Read as ‘0’

bit 3-0 **OB3E:OB0E:** Output Buffer x Status Empty bit
1 = Output Buffer x is empty (writing data to the buffer will clear this bit)
0 = Output Buffer x contains untransmitted data

Note 1: Even though an individual bit represents the byte in the buffer, the bits corresponding to the word (Byte 0 and 1, or Byte 2 and 3) get cleared, even on byte reading.

REGISTER 28-1: CTMUCON1L: CTMU CONTROL REGISTER 1 LOW (CONTINUED)

bit 1-0	**IRNG<1:0>:** Current Source Range Select bits If IRNGH = 0: 11 = 55 μA range 10 = 5.5 μA range 01 = 550 nA range 00 = 550 μA range If IRNGH = 1: 11 = Reserved 10 = Reserved 01 = 2.2 mA range 00 = 550 μA range