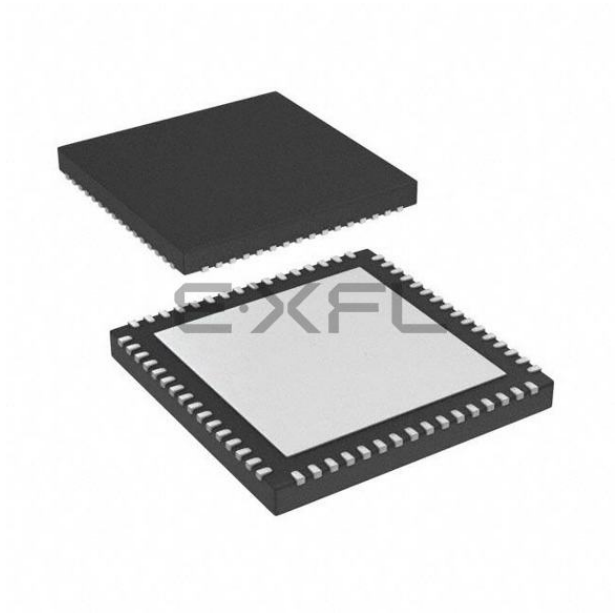
Microchip Technology - PIC24FJ1024GA606-I/MR Datasheet



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	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-VFQFN Exposed Pad
Supplier Device Package	64-QFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj1024ga606-i-mr

PIC24FJ1024GA610/GB610 FAMILY PRODUCT FAMILIES

The device names, pin counts, memory sizes and peripheral availability of each device are listed in Table 1. Their pinout diagrams appear on the following pages.

TABLE 1: PIC24FJ1024GA610/GB610 GENERAL PURPOSE FAMILIES

Device	Memory		Pins		Analog			Digital								RTCC	USB OTG
	Program (bytes)	Data (bytes)	Total	I/O	10/12-Bit A/D (ch)	Comparator	CTMU	16/32-Bit Timer	IC/OC/PWM	MCCP/SCCP	I^2C	SPI	UART w/IrDA®	EPMP/EPSP	CLC		
PIC24FJ128GA606	128K	32K	64	53	16	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	N
PIC24FJ256GA606	256K	32K	64	53	16	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	N
PIC24FJ512GA606	512K	32K	64	53	16	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	N
PIC24FJ1024GA606	1024K	32K	64	53	16	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	N
PIC24FJ128GA610	128K	32K	100	85	24	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	N
PIC24FJ256GA610	256K	32K	100	85	24	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	N
PIC24FJ512GA610	512K	32K	100	85	24	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	N
PIC24FJ1024GA610	1024K	32K	100	85	24	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	N
PIC24FJ128GB606	128K	32K	64	53	16	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	Y
PIC24FJ256GB606	256K	32K	64	53	16	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	Y
PIC24FJ512GB606	512K	32K	64	53	16	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	Y
PIC24FJ1024GB606	1024K	32K	64	53	16	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	Y
PIC24FJ128GB610	128K	32K	100	85	24	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	Y
PIC24FJ256GB610	256K	32K	100	85	24	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	Y
PIC24FJ512GB610	512K	32K	100	85	24	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	Y
PIC24FJ1024GB610	1024K	32K	100	85	24	3	Y	5/2	6/6	3/4	3	3	6/2	Y	4	Y	Y

TABLE 6: COMPLETE PIN FUNCTION DESCRIPTIONS (PIC24FJXXXGA610 BGA) (CONTINUED)

Pin	Full Pin Name	Pin	Full Pin Name
J1	AN3/C2INA/RB3	K7	AN14/**RP14**/CTED5/CTPLS/PMA1/PMALH/RB14
J2	AN2/CTCMP/C2INB/**RP13**/CTED13/RB2	K8	VDD
J3	PGED2/AN7/**RP7**/U6TX/RB7	K9	**RP5**/RD15
J4	AVDD	K10	**RP16**/RF3
J5	AN11/REFI/PMA12/RB11	K11	**RP30**/RF2
J6	TCK/RA1	L1	PGEC2/AN6/**RP6**/RB6
J7	AN12/U6RX/CTED2/PMA11/RB12	L2	CVREF-/VREF-/PMA7/RA9
J8	N/C	L3	AVSS
J9	N/C	L4	AN9/$\overline{\text{TMPR}}$/**RP9**/T1CK/RB9
J10	**RP15**/RF8	L5	CVREF/AN10/PMA13/RB10
J11	SDA1/RG3	L6	**RP31**/RF13
K1	PGEC1/ALTCVREF-/ALTVREF-/AN1/**RP1**/CTED12/RB1	L7	AN13/CTED1/PMA10/RB13
K2	PGED1/ALTCVREF+/ALTVREF+/AN0/**RP0**/RB0	L8	AN15/**RP29**/CTED6/PMA0/PMALL/RB15
K3	CVREF+/VREF+/PMA6/RA10	L9	**RPI43**/RD14
K4	AN8/**RP8**/PWRGT/RB8	L10	**RP10**/PMA9/RF4
K5	N/C	L11	**RP17**/PMA8/RF5
K6	**RPI32**/CTED7/PMA18/RF12		

Legend: **RPn** and **RPIn** represent remappable pins for Peripheral Pin Select (PPS) functions.

Note: Pinouts are subject to change.

TABLE 4-7: SFR MAP: 0300h BLOCK

File Name	Address	All Resets	File Name	Address	All Resets
SINGLE OUTPUT CAPTURE/COMPARE/PWM			**SINGLE OUTPUT CAPTURE/COMPARE/PWM (CONTINUED)**		
CCP4CON1L	0300	0000	CCP6STATH	0356	0000
CCP4CON1H	0302	0000	CCP6TMRL	0358	0000
CCP4CON2L	0304	0000	CCP6TMRH	035A	0000
CCP4CON2H	0306	0100	CCP6PRL	035C	FFFF
CCP4CON3L	0308	0000	CCP6PRH	035E	FFFF
CCP4CON3H	030A	0000	CCP6RAL	0360	0000
CCP4STATL	030C	00x0	CCP6RAH	0362	0000
CCP4STATH	030E	0000	CCP6RBL	0364	0000
CCP4TMRL	0310	0000	CCP6RBH	0366	0000
CCP4TMRH	0312	0000	CCP6BUFL	0368	0000
CCP4PRL	0314	FFFF	CCP6BUFH	036A	0000
CCP4PRH	0316	FFFF	CCP7CON1L	036C	0000
CCP4RAL	0318	0000	CCP7CON1H	036E	0000
CCP4RAH	031A	0000	CCP7CON2L	0370	0000
CCP4RBL	031C	0000	CCP7CON2H	0372	0100
CCP4RBH	031E	0000	CCP7CON3L	0374	0000
CCP4BUFL	0320	0000	CCP7CON3H	0376	0000
CCP4BUFH	0322	0000	CCP7STATL	0378	00x0
CCP5CON1L	0324	0000	CCP7STATH	037A	0000
CCP5CON1H	0326	0000	CCP7TMRL	037C	0000
CCP5CON2L	0328	0000	CCP7TMRH	037E	0000
CCP5CON2H	032A	0100	CCP7PRL	0380	FFFF
CCP5CON3L	032C	0000	CCP7PRH	0382	FFFF
CCP5CON3H	032E	0000	CCP7RAL	0384	0000
CCP5STATL	0330	00x0	CCP7RAH	0386	0000
CCP5STATH	0332	0000	CCP7RBL	0388	0000
CCP5TMRL	0334	0000	CCP7RBH	038A	0000
CCP5TMRH	0336	0000	CCP7BUFL	038C	0000
CCP5PRL	0338	FFFF	CCP7BUFH	038E	0000
CCP5PRH	033A	FFFF	**UART**		
CCP5RAL	033C	0000	U1MODE	0398	0000
CCP5RAH	033E	0000	U1STA	039A	0110
CCP5RBL	0340	0000	U1TXREG	039C	x0xx
CCP5RBH	0342	0000	U1RXREG	039E	0000
CCP5BUFL	0344	0000	U1BRG	03A0	0000
CCP5BUFH	0346	0000	U1ADMD	03A2	0000
CCP6CON1L	0348	0000	U2MODE	03AE	0000
CCP6CON1H	034A	0000	U2STA	03B0	0110
CCP6CON2L	034C	0000	U2TXREG	03B2	xxxx
CCP6CON2H	034E	0100	U2RXREG	03B4	0000
CCP6CON3L	0350	0000	U2BRG	03B6	0000
CCP6CON3H	0352	0000	U2ADMD	03B8	0000
CCP6STATL	0354	00x0	U3MODE	03C4	0000

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

REGISTER 9-2: CLKDIV: CLOCK DIVIDER REGISTER

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE2	DOZE1	DOZE0	DOZEN(1)	RCDIV2	RCDIV1	RCDIV0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
CPDIV1	CPDIV0	PLLEN	—	—	—	—	—
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 **ROI:** Recover on Interrupt bit
1 = Interrupts clear the DOZEN bit and reset the CPU peripheral clock ratio to 1:1
0 = Interrupts have no effect on the DOZEN bit

bit 14-12 **DOZE<2:0>:** CPU Peripheral Clock Ratio Select bits
111 = 1:128
110 = 1:64
101 = 1:32
100 = 1:16
011 = 1:8 (default)
010 = 1:4
001 = 1:2
000 = 1:1

bit 11 **DOZEN:** Doze Enable bit(1)
1 = DOZE<2:0> bits specify the CPU peripheral clock ratio
0 = CPU peripheral clock ratio is set to 1:1

bit 10-8 **RCDIV<2:0>:** System Frequency Divider Clock Source Select bits
000 = Fast RC Oscillator (FRC)
001 = Fast RC Oscillator (FRC) with PLL module (FRCPLL)
010 = Primary Oscillator (XT, HS, EC)
011 = Primary Oscillator (XT, HS, EC) with PLL module (XTPLL, HSPLL, ECPLL)
100 = Secondary Oscillator (SOSC)
101 = Low-Power RC Oscillator (LPRC)
110 = Digitally Controlled Oscillator (DCO)
111 = Reserved; do not use

bit 7-6 **CPDIV<1:0>:** System Clock Select bits (postscaler select from 96 MHz PLL, 32 MHz clock branch)
11 = 4 MHz (divide-by-8)(2)
10 = 8 MHz (divide-by-4)(2)
01 = 16 MHz (divide-by-2)
00 = 32 MHz (divide-by-1)

bit 5 **PLLEN:** USB PLL Enable bit
1 = PLL is always active
0 = PLL is only active when a PLL Oscillator mode is selected (OSCCON<14:12> = 011 or 001)

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

Note 1: This bit is automatically cleared when the ROI bit is set and an interrupt occurs.
2: This setting is not allowed while the USB module is enabled.

9.8 Secondary Oscillator

9.8.1 BASIC SOSC OPERATION

PIC24FJ1024GA610/GB610 family devices do not have to set the SOSCEN bit to use the Secondary Oscillator. Any module requiring the SOSC (such as RTCC or Timer1) will automatically turn on the SOSC when the clock signal is needed. The SOSC, however, has a long start-up time (as long as 1 second). To avoid delays for peripheral start-up, the SOSC can be manually started using the SOSCEN bit.

To use the Secondary Oscillator, the SOSCSEL bit (FOSC<3>) must be set to '1'. Programming the SOSCSEL bit to '0' configures the SOSC pins for Digital mode, enabling digital I/O functionality on the pins.

9.8.2 CRYSTAL SELECTION

The 32.768 kHz crystal used for the SOSC must have the following specifications in order to properly start up and run at the correct frequency when in High-Power mode:

- 12.5 pF loading capacitance
- 1.0 pF shunt capacitance
- A typical ESR of 35K; 50K maximum

In addition, the two external crystal loading capacitors should be in the range of 18-22 pF, which will be based on the PC board layout. The capacitors should be C0G, 5% tolerance and rated 25V or greater.

The accuracy and duty cycle of the SOSC can be measured on the REFO pin, and is recommended to be in the range of 40-60% and accurate to ±0.65 Hz.

9.8.3 LOW-POWER SOSC OPERATION

The Secondary Oscillator can operate in two distinct levels of power consumption based on device configuration. In Low-Power mode, the oscillator operates in a low drive strength, low-power state. By default, the oscillator uses a higher drive strength, and therefore, requires more power. Low-Power mode is selected by Configuration bit, SOSCHP (FDEVOPT1<3>). The lower drive strength of this mode makes the SOSC more sensitive to noise and requires a longer start-up time. This mode can be used with lower load capacitance crystals (6 pF-9 pF) having higher ESR ratings (50K-80K) to reduce Sleep current in the RTCC. When Low-Power mode is used, care must be taken in the design and layout of the SOSC circuit to ensure that the oscillator starts up and oscillates properly. PC board layout issues, stray capacitance and other factors will need to be carefully controlled in order for the crystal to operate.

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.

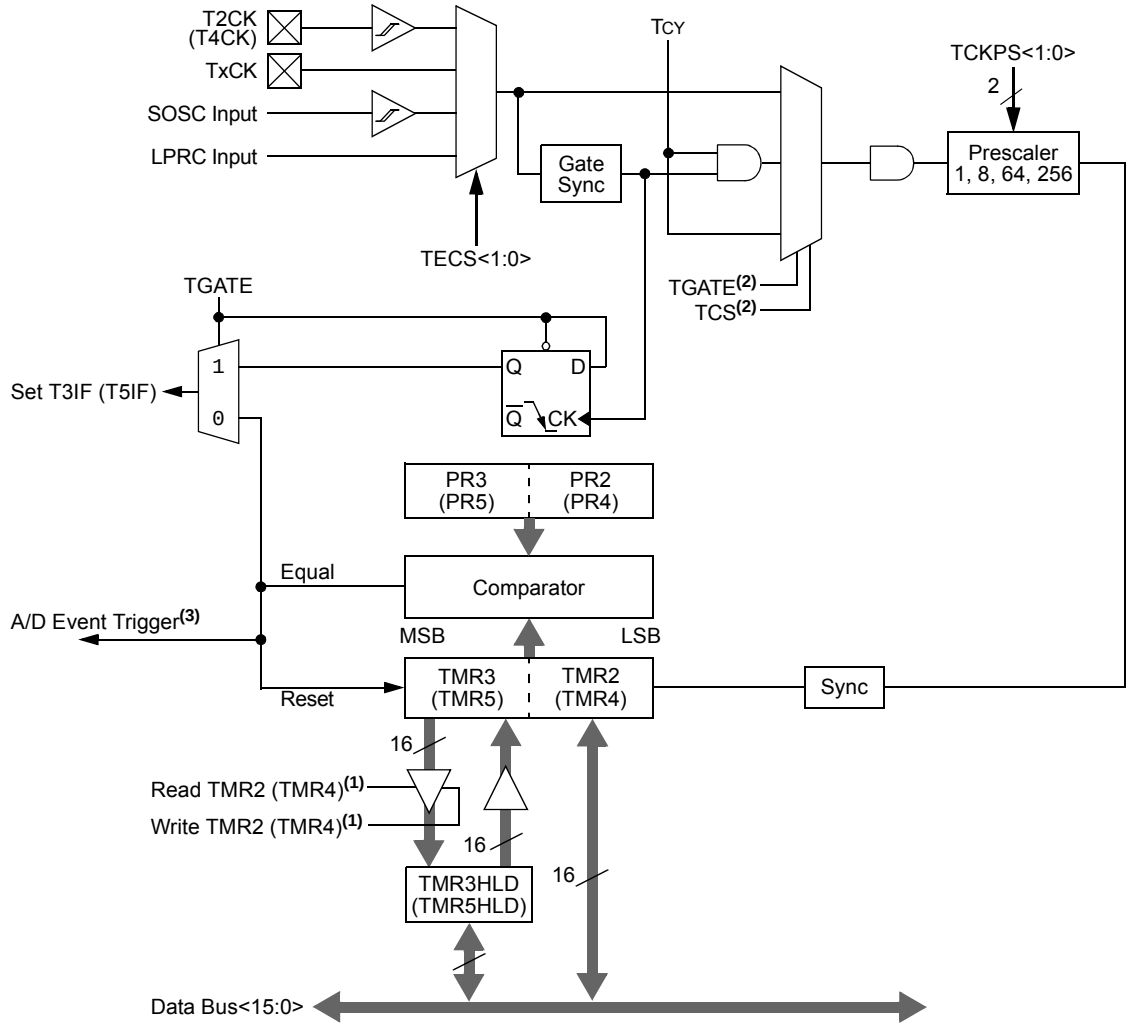
Both bits have similar functions in enabling or disabling their associated module. Setting the PMD bit for a module disables all clock sources to that module, reducing its power consumption to an absolute minimum. In this state, the control and status registers associated with the peripheral will also 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.

NOTES:

FIGURE 13-1: TIMER2/3 AND TIMER4/5 (32-BIT) BLOCK DIAGRAM



Note 1: The 32-Bit Timer Configuration bit, T32, must be set for 32-bit timer/counter operation. All control bits are respective to the T2CON and T4CON registers.

2: The timer clock input must be assigned to an available RPn/RPIn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

3: The A/D Event Trigger is available only on Timer2/3 in 32-bit mode and Timer3 in 16-bit mode.

REGISTER 17-9: SPIxIMSKL: SPIx INTERRUPT MASK REGISTER LOW

U-0	U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	—	—	FRMERREN	BUSYEN	—	—	SPITUREN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0
SRMTEN	SPIROVEN	SPIRBEN	—	SPITBEN	—	SPITBFEN	SPIRBFEN
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 **FRMERREN:** Enable Interrupt Events via FRMERR bit
1 = Frame error generates an interrupt event
0 = Frame error does not generate an interrupt event

bit 11 **BUSYEN:** Enable Interrupt Events via SPIBUSY bit
1 = SPIBUSY generates an interrupt event
0 = SPIBUSY does not generate an interrupt event

bit 10-9 **Unimplemented:** Read as '0'

bit 8 **SPITUREN:** Enable Interrupt Events via SPITUR bit
1 = Transmit Underrun (TUR) generates an interrupt event
0 = Transmit Underrun does not generate an interrupt event

bit 7 **SRMTEN:** Enable Interrupt Events via SRMT bit
1 = Shift Register Empty (SRMT) generates interrupt events
0 = Shift Register Empty does not generate interrupt events

bit 6 **SPIROVEN:** Enable Interrupt Events via SPIROV bit
1 = SPIx Receive Overflow generates an interrupt event
0 = SPIx Receive Overflow does not generate an interrupt event

bit 5 **SPIRBEN:** Enable Interrupt Events via SPIRBE bit
1 = SPIx RX Buffer Empty generates an interrupt event
0 = SPIx RX Buffer Empty does not generate an interrupt event

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

bit 3 **SPITBEN:** Enable Interrupt Events via SPITBE bit
1 = SPIx Transmit Buffer Empty generates an interrupt event
0 = SPIx Transmit Buffer Empty does not generate an interrupt event

bit 2 **Unimplemented:** Read as '0'

bit 1 **SPITBFEN:** Enable Interrupt Events via SPITBF bit
1 = SPIx Transmit Buffer Full generates an interrupt event
0 = SPIx Transmit Buffer Full does not generate an interrupt event

bit 0 **SPIRBFEN:** Enable Interrupt Events via SPIRBF bit
1 = SPIx Receive Buffer Full generates an interrupt event
0 = SPIx Receive Buffer Full does not generate an interrupt event

20.5.2 COMPLETE A CONTROL TRANSACTION TO A CONNECTED DEVICE

1. Follow the procedure described in **Section 20.5.1 "Enable Host Mode and Discover a Connected Device"** to discover a device.
2. Set up the Endpoint Control register for bidirectional control transfers by writing 0Dh to U1EP0 (this sets the EPCONDIS, EPTXEN and EPHSHK bits).
3. Place a copy of the device framework setup command in a memory buffer. See Chapter 9 of the *"USB 2.0 Specification"* for information on the device framework command set.
4. Initialize the Buffer Descriptor (BD) for the current (even or odd) TX EP0 to transfer the eight bytes of command data for a device framework command (i.e., `GET DEVICE DESCRIPTOR`):
   a) Set the BD Data Buffer Address (BD0ADR) to the starting address of the 8-byte memory buffer containing the command.
   b) Write 8008h to BD0STAT (this sets the UOWN bit and sets a byte count of 8).
5. Set the USB device address of the target device in the address register (U1ADDR<6:0>). After a USB bus Reset, the device USB address will be zero. After enumeration, it will be set to another value between 1 and 127.
6. Write D0h to U1TOK; this is a SETUP token to Endpoint 0, the target device's default control pipe. This initiates a SETUP token on the bus, followed by a data packet. The device handshake is returned in the PID field of BD0STAT after the packets are complete. When the USB module updates BD0STAT, a Token Complete Interrupt Flag is asserted (the TRNIF flag is set). This completes the setup phase of the setup transaction, as referenced in Chapter 9 of the *"USB 2.0 Specification"*.
7. To initiate the data phase of the setup transaction (i.e., get the data for the `GET DEVICE DESCRIPTOR` command), set up a buffer in memory to store the received data.
8. Initialize the current (even or odd) RX or TX (RX for IN, TX for OUT) EP0 BD to transfer the data.
   a) Write C040h to BD0STAT. This sets the UOWN, configures the Data Toggle bit (DTS) to DATA1 and sets the byte count to the length of the data buffer (64 or 40h in this case).
   b) Set BD0ADR to the starting address of the data buffer.
9. Write the Token register with the appropriate IN or OUT token to Endpoint 0, the target device's default control pipe (e.g., write 90h to U1TOK for an IN token for a `GET DEVICE DESCRIPTOR` command). This initiates an IN token on the bus, followed by a data packet from the device to the host. When the data packet completes, the BD0STAT is written and a Token Complete Interrupt Flag is asserted (the TRNIF flag is set). For control transfers with a single packet data phase, this completes the data phase of the setup transaction, as referenced in Chapter 9 of the *"USB 2.0 Specification"*. If more data needs to be transferred, return to Step 8.
10. To initiate the status phase of the setup transaction, set up a buffer in memory to receive or send the zero length status phase data packet.
11. Initialize the current (even or odd) TX EP0 BD to transfer the status data:
    a) Set the BDT buffer address field to the start address of the data buffer.
    b) Write 8000h to BD0STAT (set UOWN bit, configure DTS to DATA0 and set byte count to 0).
12. Write the Token register with the appropriate IN or OUT token to Endpoint 0, the target device's default control pipe (e.g., write 01h to U1TOK for an OUT token for a `GET DEVICE DESCRIPTOR` command). This initiates an OUT token on the bus, followed by a zero length data packet from the host to the device. When the data packet completes, the BD is updated with the handshake from the device and a Token Complete Interrupt Flag is asserted (the TRNIF flag is set). This completes the status phase of the setup transaction, as described in Chapter 9 of the *"USB 2.0 Specification"*.

Note:	Only one control transaction can be performed per frame.

REGISTER 20-16: U1IR: USB INTERRUPT STATUS REGISTER (DEVICE MODE ONLY)

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

R/K-0, HS	U-0	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS
STALLIF	—	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	URSTIF
bit 7							bit 0

Legend:	U = Unimplemented bit, read as '0'		
R = Readable bit	K = Write '1' to Clear bit	HS = Hardware Settable 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 **STALLIF:** STALL Handshake Interrupt bit
- 1 = A STALL handshake was sent by the peripheral during the handshake phase of the transaction in Device mode
- 0 = A STALL handshake has not been sent

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

bit 5 **RESUMEIF:** Resume Interrupt bit
- 1 = A K-state is observed on the D+ or D- pin for 2.5 μs (differential '1' for low speed, differential '0' for full speed)
- 0 = No K-state is observed

bit 4 **IDLEIF:** Idle Detect Interrupt bit
- 1 = Idle condition is detected (constant Idle state of 3 ms or more)
- 0 = No Idle condition is detected

bit 3 **TRNIF:** Token Processing Complete Interrupt bit
- 1 = Processing of the current token is complete; read the U1STAT register for endpoint information
- 0 = Processing of the current token is not complete; clear the U1STAT register or load the next token from STAT (clearing this bit causes the STAT FIFO to advance)

bit 2 **SOFIF:** Start-of-Frame Token Interrupt bit
- 1 = A Start-of-Frame token is received by the peripheral or the Start-of-Frame threshold is reached by the host
- 0 = No Start-of-Frame token is received or threshold reached

bit 1 **UERRIF**: USB Error Condition Interrupt bit
- 1 = An unmasked error condition has occurred; only error states enabled in the U1EIE register can set this bit
- 0 = No unmasked error condition has occurred

bit 0 **URSTIF:** USB Reset Interrupt bit
- 1 = Valid USB Reset has occurred for at least 2.5 μs; Reset state must be cleared before this bit can be reasserted
- 0 = No USB Reset has occurred; individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

NOTES:

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.

NOTES:

22.3 Registers

22.3.1 RTCC CONTROL REGISTERS

REGISTER 22-1: RTCCON1L: RTCC CONTROL REGISTER 1 (LOW)

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
RTCEN	—	—	—	WRLOCK	PWCEN	PWCPOL	PWCPOE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0
RTCOE	OUTSEL2	OUTSEL1	OUTSEL0	—	—	—	TSAEN
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 **RTCEN:** RTCC Enable bit
1 = RTCC is enabled and counts from selected clock source
0 = RTCC is not enabled

bit 14-12 **Unimplemented:** Read as ‘0’

bit 11 **WRLOCK:** RTCC Register Write Lock
1 = RTCC registers are locked
0 = RTCC registers may be written to by user

bit 10 **PWCEN:** Power Control Enable bit
1 = Power control is enabled
0 = Power control is disabled

bit 9 **PWCPOL:** Power Control Polarity bit
1 = Power control output is active-high
0 = Power control output is active-low

bit 8 **PWCPOE:** Power Control Output Enable bit
1 = Power control output pin is enabled
0 = Power control output pin is disabled

bit 7 **RTCOE:** RTCC Output Enable bit
1 = RTCC output is enabled
0 = RTCC output is disabled

bit 6-4 **OUTSEL<2:0>:** RTCC Output Signal Selection bits
111 = Unused
110 = Unused
101 = Unused
100 = Timestamp A event
011 = Power control
010 = RTCC input clock
001 = Second clock
000 = Alarm event

bit 3-1 **Unimplemented:** Read as ‘0’

bit 0 **TSAEN:** Timestamp A Enable bit
1 = Timestamp event will occur when a low pulse is detected on the $\overline{\text{TMPR}}$ pin
0 = Timestamp is disabled

REGISTER 30-11: FDEVOPT1 CONFIGURATION REGISTER

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
—	—	—	—	—	—	—	—
bit 23							bit 16

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
—	—	—	—	—	—	—	—
bit 15							bit 8

U-1	U-1	U-1	R/PO-1	R/PO-1	R/PO-1	R/PO-1	U-1
—	—	—	ALTVREF	SOSCHP(1)	TMPRPIN	ALTCMPI	—
bit 7							bit 0

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

bit 23-5 **Unimplemented:** Read as '1'

bit 4 **ALTVREF:** Alternate Voltage Reference Location Enable bit (100-pin and 121-pin devices only)
1 = VREF+ and CVREF+ on RA10, VREF- and CVREF- on RA9
0 = VREF+ and CVREF+ on RB0, VREF- and CVREF- on RB1

bit 3 **SOSCHP:** SOSC High-Power Enable bit (valid only when SOSCSEL = 1)(1)
1 = SOSC High-Power mode is enabled
0 = SOSC Low-Power mode is enabled

bit 2 **TMPRPIN:** Tamper Pin Enable bit
1 = $\overline{\text{TMPR}}$ pin function is disabled
0 = $\overline{\text{TMPR}}$ pin function is enabled

bit 1 **ALTCMPI:** Alternate Comparator Input Enable bit
1 = C1INC, C2INC and C3INC are on their standard pin locations
0 = C1INC, C2INC and C3INC are on RG9

bit 0 **Unimplemented:** Read as '1'

Note 1: High-Power mode is for crystals with 35K ESR (typical). Low-Power mode is for crystals with more than 65K ESR.

TABLE 30-2: DEVICE ID REGISTERS

Address	Name	Bit															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FF0000h	DEVID	FAMID<7:0>								DEV<7:0>							
FF0002h	DEVREV	—												REV<3:0>			

TABLE 30-3: DEVICE ID BIT FIELD DESCRIPTIONS

Bit Field	Register	Description
FAMID<7:0>	DEVID	Encodes the family ID of the device.
DEV<7:0>	DEVID	Encodes the individual ID of the device.
REV<3:0>	DEVREV	Encodes the sequential (numerical) revision identifier of the device.

TABLE 30-4: PIC24FJ1024GA610/GB610 FAMILY DEVICE IDs

Device	DEVID
PIC24FJ128GA606	6000h
PIC24FJ256GA606	6008h
PIC24FJ512GA606	6010h
PIC24FJ1024GA606	6018h
PIC24FJ128GA610	6001h
PIC24FJ256GA610	6009h
PIC24FJ512GA610	6011h
PIC24FJ1024GA610	6019h
PIC24FJ128GB606	6004h
PIC24FJ256GB606	600Ch
PIC24FJ512GB606	6014h
PIC24FJ1024GB606	601Ch
PIC24FJ128GB610	6005h
PIC24FJ256GB610	600Dh
PIC24FJ512GB610	6015h
PIC24FJ1024GB610	601Dh

30.2 Unique Device Identifier (UDID)

All PIC24FJ1024GA610/GB610 family devices are individually encoded during final manufacturing with a Unique Device Identifier, or UDID. The UDID cannot be erased by a bulk erase command or any other user-accessible means. This feature allows for manufacturing traceability of Microchip Technology devices in applications where this is a requirement. It may also be used by the application manufacturer for any number of things that may require unique identification, such as:

- Tracking the device
- Unique serial number
- Unique security key

The UDID comprises five 24-bit program words. When taken together, these fields form a unique 120-bit identifier.

The UDID is stored in five read-only locations, located between 801600h and 801608h in the device configuration space. Table 30-5 lists the addresses of the identifier words.

TABLE 30-5: UDID ADDRESSES

UDID	Address	Description
UDID1	801600	UDID Word 1
UDID2	801602	UDID Word 2
UDID3	801604	UDID Word 3
UDID4	801606	UDID Word 4
UDID5	801608	UDID Word 5

TABLE 33-3: DC CHARACTERISTICS: TEMPERATURE AND VOLTAGE SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
Operating Voltage							
DC10	VDD	**Supply Voltage**	2.0	—	3.6	V	BOR is disabled
			VBOR	—	3.6	V	BOR is enabled
DC12	VDR	**RAM Data Retention Voltage(1)**	Greater of: VPORREL or VBOR	—	—	V	VBOR is used only if BOR is enabled (BOREN = 1)
DC16	VPOR	**VDD Start Voltage** to Ensure Internal Power-on Reset Signal	VSS	—	—	V	**(Note 2)**
DC17A	SVDD	**Recommended VDD Rise Rate** to Ensure Internal Power-on Reset Signal	1V/20 ms	—	1V/10 µS	sec	**(Note 2, Note 4)**
DC17B	VBOR	**Brown-out Reset Voltage** on VDD Transition, High-to-Low	2.0	2.1	2.2	V	**(Note 3)**

Note 1: This is the limit to which VDD may be lowered and the RAM contents will always be retained.

2: If the VPOR or SVDD parameters are not met, or the application experiences slow power-down VDD ramp rates, it is recommended to enable and use BOR.

3: On a rising VDD power-up sequence, application firmware execution begins at the higher of the VPORREL or VBOR level (when BOREN = 1).

4: VDD rise times outside this window may not internally reset the processor and are not parametrically tested.

33.2 AC Characteristics and Timing Parameters

The information contained in this section defines the PIC24FJ1024GA610/GB610 family AC characteristics and timing parameters.

TABLE 33-16: TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC

AC CHARACTERISTICS	**Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for Industrial Operating voltage VDD range as described in **Section 33.1 "DC Characteristics"**.

FIGURE 33-2: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

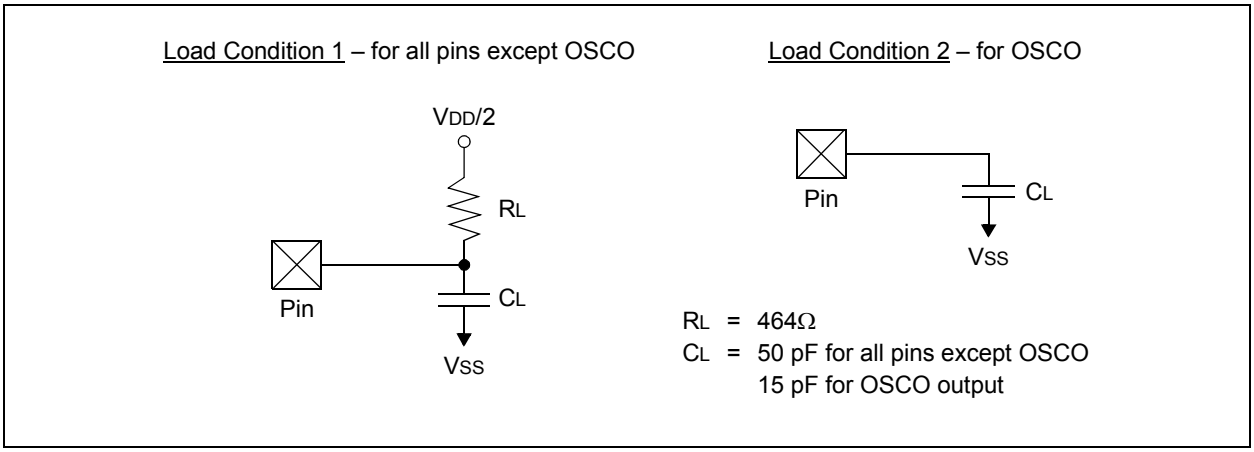


TABLE 33-17: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS

Param No.	Symbol	Characteristic	Min	Typ(1)	Max	Units	Conditions
DO50	COSCO	OSCO/CLKO Pin	—	—	15	pF	In XT and HS modes when external clock is used to drive OSCI
DO56	CIO	All I/O Pins and OSCO	—	—	50	pF	EC mode
DO58	CB	SCLx, SDAx	—	—	400	pF	In I^2C mode

Note 1: Data in the "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

TABLE 33-20: INTERNAL RC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
F20	FRC Accuracy @ 8 MHz	-1.5	+0.15	1.5	%	2.0V ≤ VDD ≤ 3.6V, 0°C ≤ TA ≤ +85°C **(Note 1)**
		-2.0	—	2.0	%	2.0V ≤ VDD ≤ 3.6V, -40°C ≤ TA ≤ 0°C
		-0.20	+0.05	-0.20	%	2.0V ≤ VDD ≤ 3.6V, 0°C ≤ TA ≤ +85°C, self-tune is enabled and locked **(Note 2)**
F21	LPRC @ 31 kHz	-20	—	20	%	VCAP Output Voltage = 1.8V
F22	OSCTUN Step-Size	—	0.05	—	%/bit	
F23	FRC Self-Tune Lock Time	—	5	8	ms	**(Note 3)**

Note 1: To achieve this accuracy, physical stress applied to the microcontroller package (ex., by flexing the PCB) must be kept to a minimum.

2: Accuracy is measured with respect to the reference source.

3: Time from reference clock stable, and in range, to FRC tuned within range specified by F20 (with self-tune).

TABLE 33-21: RC OSCILLATOR START-UP TIME

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
FR0	TFRC	FRC Oscillator Start-up Time	—	15	—	μs	
FR1	TLPRC	Low-Power RC Oscillator Start-up Time	—	50	—	μs	