

Welcome to [E-XFL.COM](https://www.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.

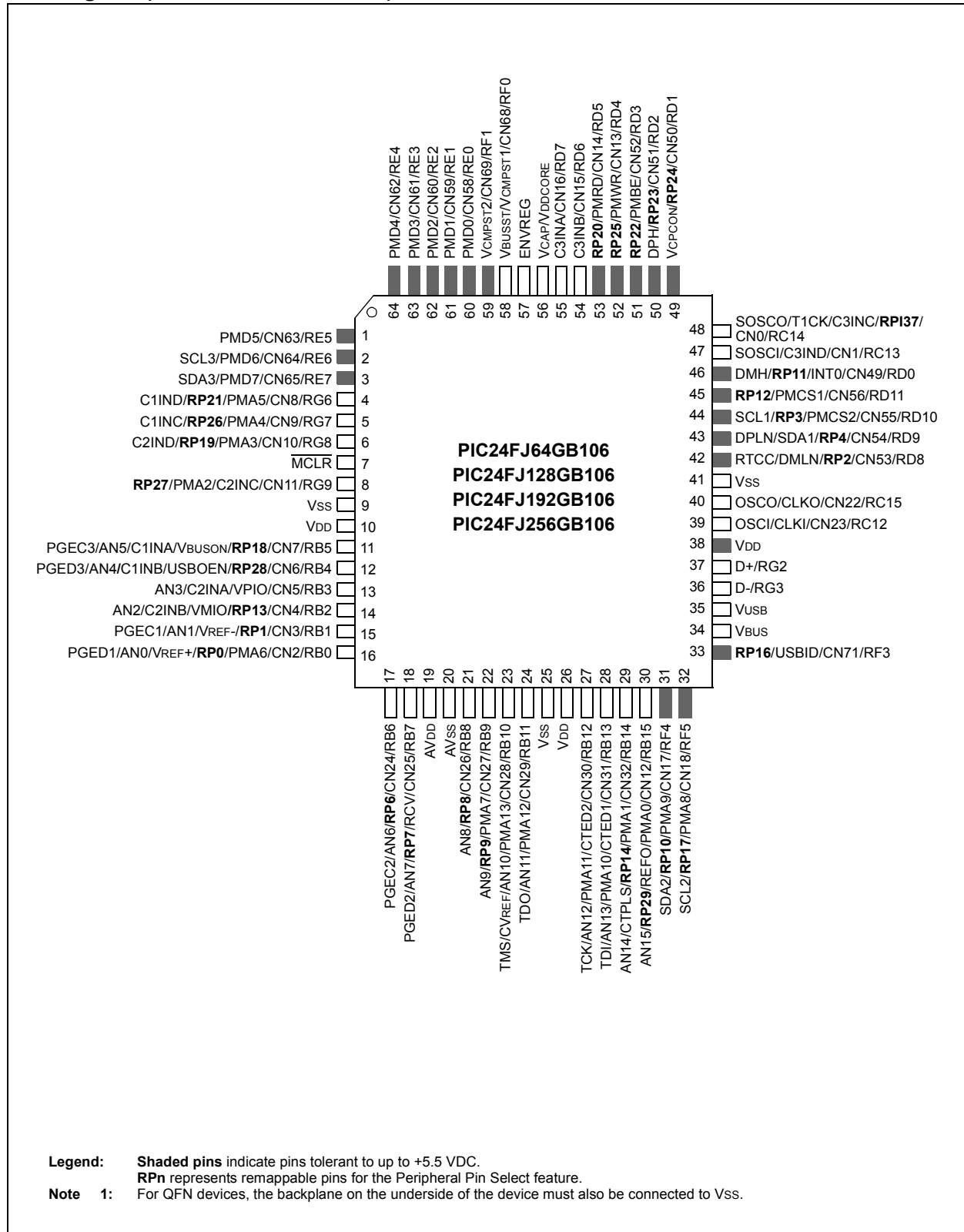
Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Active
Core Processor	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I ² C, SPI, UART/USART, USB OTG
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	83
Program Memory Size	128KB (43K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K 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	100-TQFP
Supplier Device Package	100-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj128gb110t-i-pt

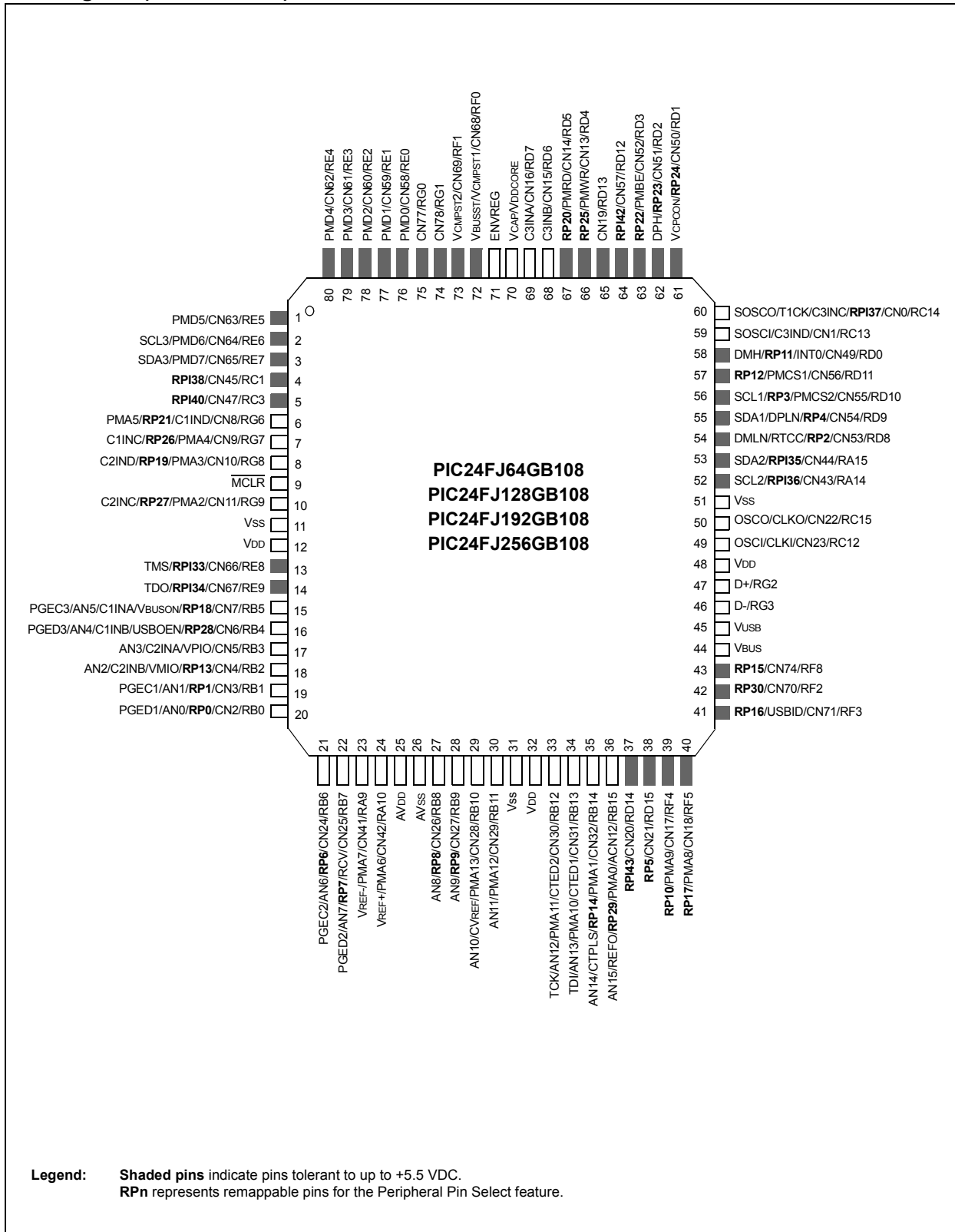
PIC24FJ256GB110 FAMILY

Pin Diagram (64-Pin TQFP and QFN)



PIC24FJ256GB110 FAMILY

Pin Diagram (80-Pin TQFP)



PIC24FJ256GB110 FAMILY

2.0 GUIDELINES FOR GETTING STARTED WITH 16-BIT MICROCONTROLLERS

2.1 Basic Connection Requirements

Getting started with the PIC24FJ256GB110 family of 16-bit microcontrollers requires attention to a minimal set of device pin connections before proceeding with development.

The following pins must always be connected:

- All VDD and VSS pins (see **Section 2.2 “Power Supply Pins”**)
- All AVDD and AVSS pins, regardless of whether or not the analog device features are used (see **Section 2.2 “Power Supply Pins”**)
- MCLR pin (see **Section 2.3 “Master Clear (MCLR) Pin”**)
- ENVREG/DISVREG and VCAP/VDDCORE pins (PIC24FJ devices only) (see **Section 2.4 “Voltage Regulator Pins (ENVREG/DISVREG and VCAP/VDDCORE)”**)

These pins must also be connected if they are being used in the end application:

- PGECx/PGEDx pins used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes (see **Section 2.5 “ICSP Pins”**)
- OSCI and OSCO pins when an external oscillator source is used (see **Section 2.6 “External Oscillator Pins”**)

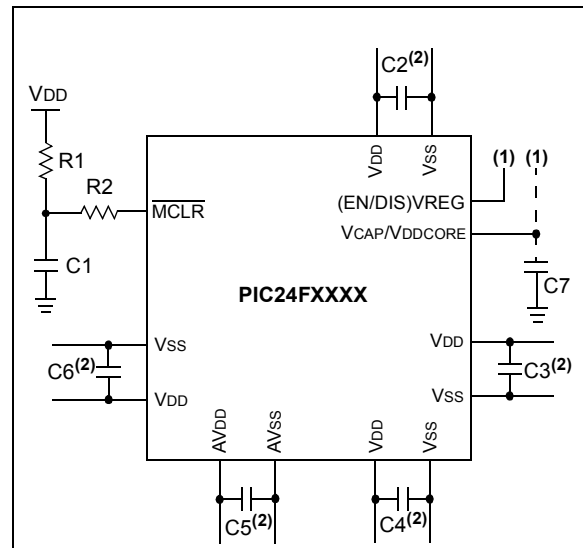
Additionally, the following pins may be required:

- VREF+/VREF- pins used when external voltage reference for analog modules is implemented

Note: The AVDD and AVSS pins must always be connected, regardless of whether any of the analog modules are being used.

The minimum mandatory connections are shown in Figure 2-1.

FIGURE 2-1: RECOMMENDED MINIMUM CONNECTIONS



Key (all values are recommendations):

C1 through C6: 0.1 μ F, 20V ceramic

C7: 10 μ F, 6.3V or greater, tantalum or ceramic

R1: 10 k Ω

R2: 100 Ω to 470 Ω

Note 1: See **Section 2.4 “Voltage Regulator Pins (ENVREG/DISVREG and VCAP/VDDCORE)”** for explanation of ENVREG/DISVREG pin connections.

2: The example shown is for a PIC24F device with five VDD/VSS and AVDD/AVSS pairs. Other devices may have more or less pairs; adjust the number of decoupling capacitors appropriately.

TABLE 4-7: INPUT CAPTURE REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
IC1CON1	0140	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC1CON2	0142	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC1BUF	0144	Input Capture 1 Buffer Register																0000
IC1TMR	0146	Timer Value 1 Register																xxxx
IC2CON1	0148	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC2CON2	014A	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC2BUF	014C	Input Capture 2 Buffer Register																0000
IC2TMR	014E	Timer Value 2 Register																xxxx
IC3CON1	0150	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC3CON2	0152	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC3BUF	0154	Input Capture 3 Buffer Register																0000
IC3TMR	0156	Timer Value 3 Register																xxxx
IC4CON1	0158	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC4CON2	015A	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC4BUF	015C	Input Capture 4 Buffer Register																0000
IC4TMR	015E	Timer Value 4 Register																xxxx
IC5CON1	0160	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC5CON2	0162	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC5BUF	0164	Input Capture 5 Buffer Register																0000
IC5TMR	0166	Timer Value 5 Register																xxxx
IC6CON1	0168	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC6CON2	016A	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC6BUF	016C	Input Capture 6 Buffer Register																0000
IC6TMR	016E	Timer Value 6 Register																xxxx
IC7CON1	0170	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC7CON2	0172	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC7BUF	0174	Input Capture 7 Buffer Register																0000
IC7TMR	0176	Timer Value 7 Register																xxxx
IC8CON1	0178	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC8CON2	017A	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC8BUF	017C	Input Capture 8 Buffer Register																0000
IC8TMR	017E	Timer Value 8 Register																xxxx
IC9CON1	0180	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC9CON2	0182	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC9BUF	0184	Input Capture 9 Buffer Register																0000
IC9TMR	0186	Timer Value 9 Register																xxxx

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

PIC24FJ256GB110 FAMILY

5.0 FLASH PROGRAM MEMORY

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 “PIC24F Family Reference Manual”, **Section 4. “Program Memory”** (DS39715).

The PIC24FJ256GB110 family of devices contains internal Flash program memory for storing and executing application code. It can be programmed in four ways:

- In-Circuit Serial Programming™ (ICSP™)
- Run-Time Self-Programming (RTSP)
- JTAG
- Enhanced In-Circuit Serial Programming (Enhanced ICSP)

ICSP allows a PIC24FJ256GB110 family device to be serially programmed while in the end application circuit. This is simply done with two lines for the programming clock and programming data (which are named PGECx and PGEDx, respectively), and three other lines for power (VDD), ground (VSS) and Master Clear (MCLR). This allows customers to manufacture boards with unprogrammed devices and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

RTSP is accomplished using TBLRD (table read) and TBLWT (table write) instructions. With RTSP, the user may write program memory data in blocks of 64 instructions (192 bytes) at a time, and erase program memory in blocks of 512 instructions (1536 bytes) at a time.

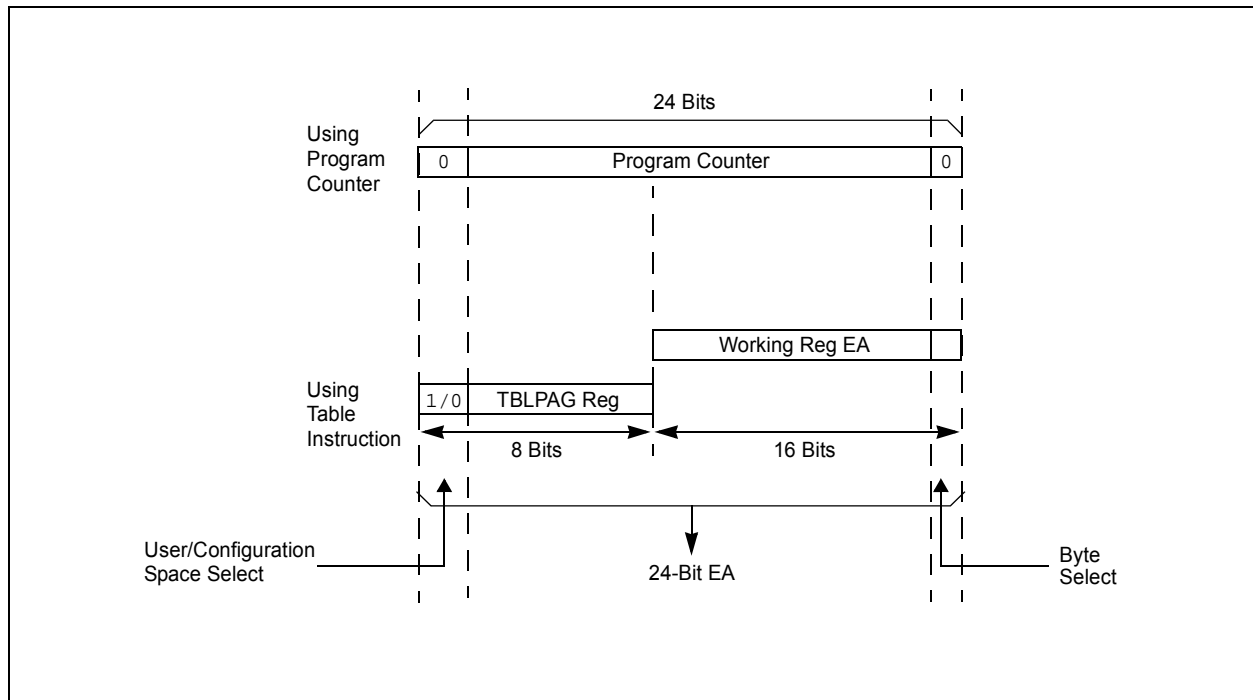
5.1 Table Instructions and Flash Programming

Regardless of the method used, all programming of Flash memory is done with the table read and table write instructions. These allow direct read and write access to the program memory space from the data memory while the device is in normal operating mode. The 24-bit target address in the program memory is formed using the TBLPAG<7:0> bits and the Effective Address (EA) from a W register specified in the table instruction, as shown in Figure 5-1.

The TBLRDL and the TBLWTL instructions are used to read or write to bits<15:0> of program memory. TBLRDL and TBLWTL can access program memory in both Word and Byte modes.

The TBLRDH and TBLWTH instructions are used to read or write to bits<23:16> of program memory. TBLRDH and TBLWTH can also access program memory in Word or Byte mode.

FIGURE 5-1: ADDRESSING FOR TABLE REGISTERS



PIC24FJ256GB110 FAMILY

EXAMPLE 5-4: LOADING THE WRITE BUFFERS (C LANGUAGE CODE)

```
// C example using MPLAB C30

#define NUM_INSTRUCTION_PER_ROW 64
unsigned int offset;
unsigned int i;
unsigned long progAddr = 0XXXXXXX;           // Address of row to write
unsigned int progData[2*NUM_INSTRUCTION_PER_ROW]; // Buffer of data to write

//Set up NVMCON for row programming
NVMCON = 0x4001;                             // Initialize NVMCON

//Set up pointer to the first memory location to be written
TBLPAG = progAddr>>16;                       // Initialize PM Page Boundary SFR
offset = progAddr & 0xFFFF;                  // Initialize lower word of address

//Perform TBLWT instructions to write necessary number of latches
for(i=0; i < 2*NUM_INSTRUCTION_PER_ROW; i++)
{
    __builtin_tblwtl(offset, progData[i++]); // Write to address low word
    __builtin_tblwth(offset, progData[i]);   // Write to upper byte
    offset = offset + 2;                     // Increment address
}
```

EXAMPLE 5-5: INITIATING A PROGRAMMING SEQUENCE (ASSEMBLY LANGUAGE CODE)

```
DISI    #5                                ; Block all interrupts with priority <7
                                           ; for next 5 instructions

MOV     #0x55, W0                         ; Write the 55 key
MOV     W0, NVMKEY
MOV     #0xAA, W1                         ;
MOV     W1, NVMKEY                       ; Write the AA key
BSET    NVMCON, #WR                      ; Start the erase sequence
NOP
NOP
BTSC    NVMCON, #15                      ; and wait for it to be
BRA     $-2                              ; completed
```

EXAMPLE 5-6: INITIATING A PROGRAMMING SEQUENCE (C LANGUAGE CODE)

```
// C example using MPLAB C30

asm("DISI #5");                          // Block all interrupts with priority < 7
                                           // for next 5 instructions

__builtin_write_NVM();                    // Perform unlock sequence and set WR
```

PIC24FJ256GB110 FAMILY

REGISTER 8-2: CLKDIV: CLOCK DIVIDER REGISTER

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

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
CPDIV1	CPDIV0	—	—	—	—	—	—
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
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>:** FRC Postscaler Select bits
111 = 31.25 kHz (divide-by-256)
110 = 125 kHz (divide-by-64)
101 = 250 kHz (divide-by-32)
100 = 500 kHz (divide-by-16)
011 = 1 MHz (divide-by-8)
010 = 2 MHz (divide-by-4)
001 = 4 MHz (divide-by-2)
000 = 8 MHz (divide-by-1)
- bit 7-6 **CPDIV<1:0>:** USB System Clock Select bits (postscaler select from 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 5-0 **Unimplemented:** Read as '0'

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

PIC24FJ256GB110 FAMILY

REGISTER 10-19: RPINR27: PERIPHERAL PIN SELECT INPUT REGISTER 27

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	U4CTSR5	U4CTSR4	U4CTSR3	U4CTSR2	U4CTSR1	U4CTSR0
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
—	—	U4RXR5	U4RXR4	U4RXR3	U4RXR2	U4RXR1	U4RXR0
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 **U4CTSR<5:0>:** Assign UART4 Clear to Send (U4CTS) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **U4RXR<5:0>:** Assign UART4 Receive (U4RX) to Corresponding RPn or RPIn Pin bits

REGISTER 10-20: RPINR28: PERIPHERAL PIN SELECT INPUT REGISTER 28

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	SCK3R5	SCK3R4	SCK3R3	SCK3R2	SCK3R1	SCK3R0
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
—	—	SDI3R5	SDI3R4	SDI3R3	SDI3R2	SDI3R1	SDI3R0
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 **SCK3R<5:0>:** Assign SPI3 Clock Input (SCK3IN) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **SDI3R<5:0>:** Assign SPI3 Data Input (SDI3) to Corresponding RPn or RPIn Pin bits

PIC24FJ256GB110 FAMILY

REGISTER 14-1: OCxCON1: OUTPUT COMPARE x CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	—	—
bit 15						bit 8	

R/W-0	U-0	U-0	R/W-0, HCS	R/W-0	R/W-0	R/W-0	R/W-0
ENFLT0	—	—	OCFLT0	TRIGMODE	OCM2 ⁽¹⁾	OCM1 ⁽¹⁾	OCM0 ⁽¹⁾
bit 7						bit 0	

Legend:

R = Readable bit

W = Writable bit

HCS = Hardware Clearable/Settable 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:** Stop Output Compare x 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 = System Clock

110 = Reserved

101 = Reserved

100 = Timer1

011 = Timer5

010 = Timer4

001 = Timer3

000 = Timer2

bit 9-8 **Unimplemented:** Read as '0'

bit 7 **ENFLT0:** Fault 0 Input Enable bit

1 = Fault 0 input is enabled

0 = Fault 0 input is disabled

bit 6-5 **Unimplemented:** Read as '0'

bit 4 **OCFLT0:** PWM Fault Condition Status bit

1 = PWM Fault condition has occurred (cleared in HW only)

0 = No PWM Fault condition has occurred (this bit is only used when OCM<2:0> = 111)

bit 3 **TRIGMODE:** Trigger Status Mode Select bit

1 = TRIGSTAT (OCxCON2<6>) is cleared when OCxRS = OCxTMR or in software

0 = TRIGSTAT is only cleared by software

bit 2-0 **OCM<2:0>:** Output Compare x Mode Select bits⁽¹⁾

111 = Center-aligned PWM mode on OCx⁽²⁾

110 = Edge-aligned PWM Mode on OCx⁽²⁾

101 = Double Compare Continuous Pulse mode: Initialize OCx pin low, toggle OCx state continuously on alternate matches of OCxR and OCxRS

100 = Double Compare Single-Shot mode: Initialize OCx pin low, toggle OCx state on matches of OCxR and OCxRS for one cycle

011 = Single Compare Continuous Pulse mode: Compare events continuously toggle OCx pin

010 = Single Compare Single-Shot mode: Initialize OCx pin high, compare event forces OCx pin low

001 = Single Compare Single-Shot mode: Initialize OCx pin low, compare event forces OCx pin high

000 = Output compare channel is disabled

Note 1: The OCx output must also be configured to an available RPN pin. For more information, see **Section 10.4 "Peripheral Pin Select"**.

2: OCFA pin controls the OC1-OC4 channels; OCFB pin controls the OC5-OC9 channels. OCxR and OCxRS are double-buffered only in PWM modes.

PIC24FJ256GB110 FAMILY

FIGURE 15-5: SPI MASTER, FRAME MASTER CONNECTION DIAGRAM

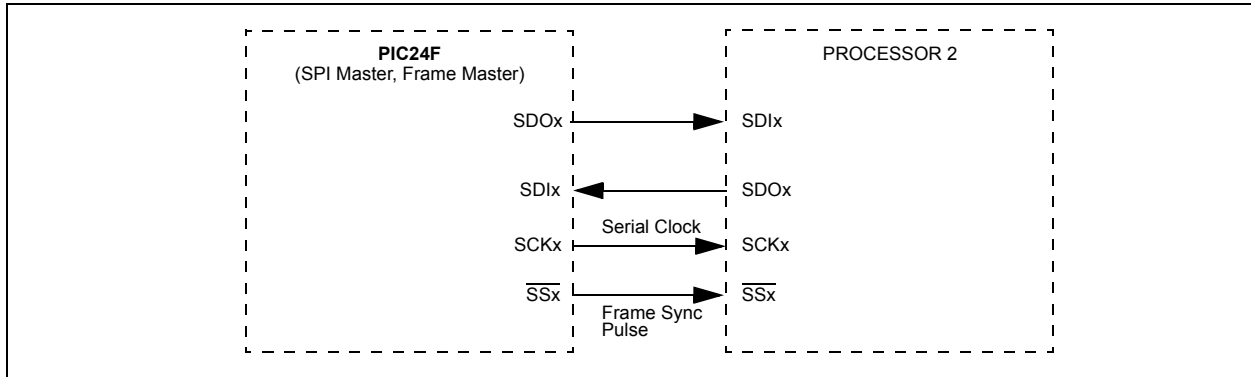


FIGURE 15-6: SPI MASTER, FRAME SLAVE CONNECTION DIAGRAM

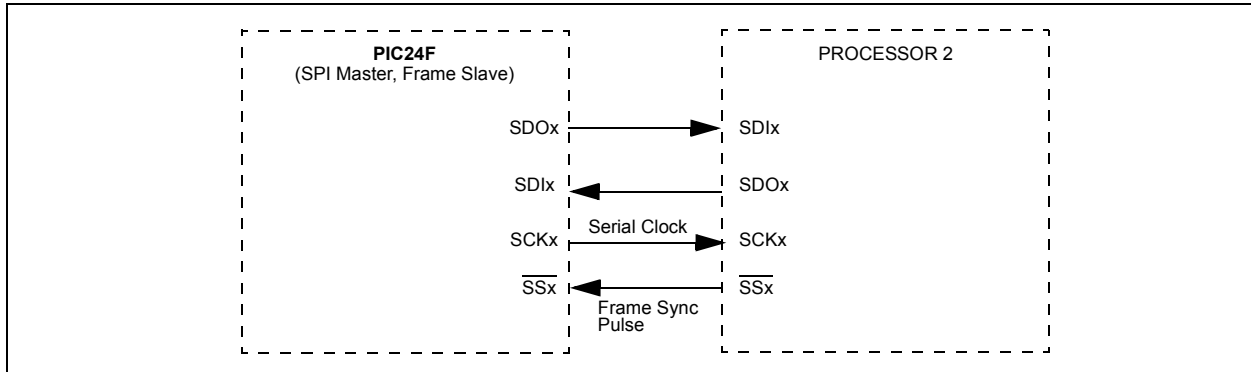


FIGURE 15-7: SPI SLAVE, FRAME MASTER CONNECTION DIAGRAM

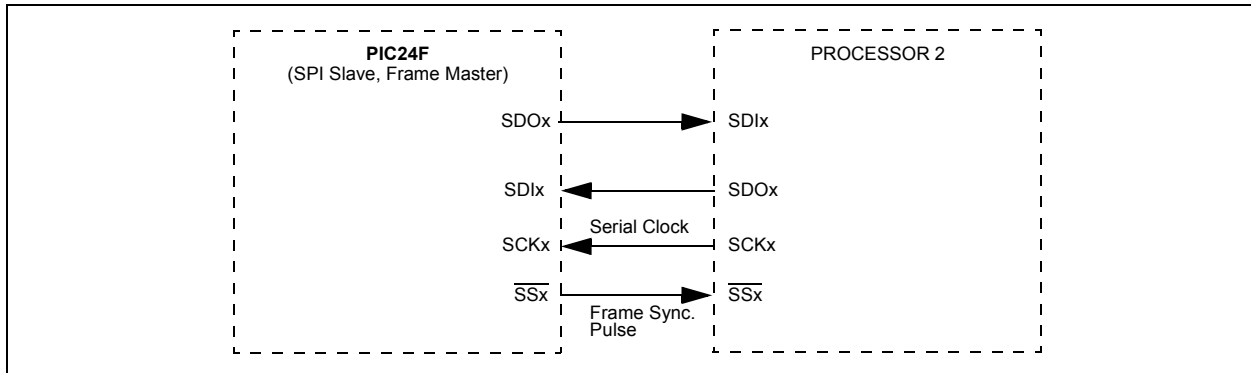
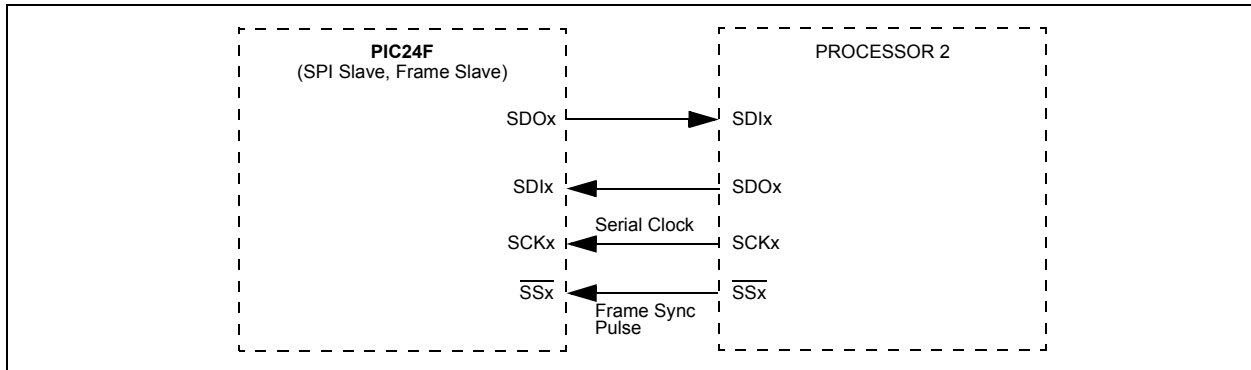


FIGURE 15-8: SPI SLAVE, FRAME SLAVE CONNECTION DIAGRAM



PIC24FJ256GB110 FAMILY

NOTES:

PIC24FJ256GB110 FAMILY

REGISTER 18-17: U1IR: USB INTERRUPT STATUS REGISTER (HOST 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	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R-0	R/K-0, HS
STALLIF	ATTACHIF	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	DETACHIF
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 device during the handshake phase of the transaction in Device mode
 0 = A STALL handshake has not been sent
- bit 6 **ATTACHIF:** Peripheral Attach Interrupt bit
 1 = A peripheral attachment has been detected by the module; set if the bus state is not SE0 and there has been no bus activity for 2.5 μ s
 0 = No peripheral attachment detected
- bit 5 **RESUMEIF:** Resume Interrupt bit
 1 = A K-state is observed on the D+ or D- pin for 2.5 μ s (differential '1' for low speed, differential '0' for full speed)
 0 = No K-state observed
- bit 4 **IDLEIF:** Idle Detect Interrupt bit
 1 = Idle condition detected (constant Idle state of 3 ms or more)
 0 = No Idle condition detected
- bit 3 **TRNIF:** Token Processing Complete Interrupt bit
 1 = Processing of current token is complete; read U1STAT register for endpoint information
 0 = Processing of current token not complete; clear U1STAT register or load next token from U1STAT
- bit 2 **SOFIF:** Start-Of-Frame Token Interrupt bit
 1 = A Start-Of-Frame token received by the peripheral or the Start-Of-Frame threshold reached by the host
 0 = No Start-Of-Frame token 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 **DETACHIF:** Detach Interrupt bit
 1 = A peripheral detachment has been detected by the module; Reset state must be cleared before this bit can be reasserted
 0 = No peripheral detachment detected. 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 bit-wise 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.

PIC24FJ256GB110 FAMILY

REGISTER 18-19: U1EIR: USB ERROR INTERRUPT STATUS REGISTER

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
BTSEF	—	DMAEF	BTOEF	DFN8EF	CRC16EF	CRC5EF EOFEF	PIDEF
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	BTSEF: Bit Stuff Error Flag bit 1 = Bit stuff error has been detected 0 = No bit stuff error
bit 6	Unimplemented: Read as '0'
bit 5	DMAEF: DMA Error Flag bit 1 = A USB DMA error condition detected; the data size indicated by the BD byte count field is less than the number of received bytes. The received data is truncated. 0 = No DMA error
bit 4	BTOEF: Bus Turnaround Time-out Error Flag bit 1 = Bus turnaround time-out has occurred 0 = No bus turnaround time-out
bit 3	DFN8EF: Data Field Size Error Flag bit 1 = Data field was not an integral number of bytes 0 = Data field was an integral number of bytes
bit 2	CRC16EF: CRC16 Failure Flag bit 1 = CRC16 failed 0 = CRC16 passed
bit 1	<u>For Device mode:</u> CRC5EF: CRC5 Host Error Flag bit 1 = Token packet rejected due to CRC5 error 0 = Token packet accepted (no CRC5 error) <u>For Host mode:</u> EOFEF: End-Of-Frame Error Flag bit 1 = End-Of-Frame error has occurred 0 = End-Of-Frame interrupt disabled
bit 0	PIDEF: PID Check Failure Flag bit 1 = PID check failed 0 = PID check passed. 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.

PIC24FJ256GB110 FAMILY

18.7.4 USB V_{Bus} POWER CONTROL REGISTER

REGISTER 18-22: U1PWMCON: USB V_{Bus} PWM GENERATOR CONTROL REGISTER

R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
PWMEN	—	—	—	—	—	PWMPOL	CNTEN
bit 15						bit 8	

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-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 **PWMEN:** PWM Enable bit
1 = PWM generator is enabled
0 = PWM generator is disabled; output is held in Reset state specified by PWMPOL
- bit 14-10 **Unimplemented:** Read as '0'
- bit 9 **PWMPOL:** PWM Polarity bit
1 = PWM output is active-low and resets high
0 = PWM output is active-high and resets low
- bit 8 **CNTEN:** PWM Counter Enable bit
1 = Counter is enabled
0 = Counter is disabled
- bit 7-0 **Unimplemented:** Read as '0'

PIC24FJ256GB110 FAMILY

REGISTER 20-1: RCFGAL: RTCC CALIBRATION AND CONFIGURATION REGISTER⁽¹⁾ (CONTINUED)

bit 7-0 **CAL<7:0>**: RTC Drift Calibration bits
 01111111 = Maximum positive adjustment; adds 508 RTC clock pulses every one minute
 ...
 00000001 = Minimum positive adjustment; adds 4 RTC clock pulses every one minute
 00000000 = No adjustment
 11111111 = Minimum negative adjustment; subtracts 4 RTC clock pulses every one minute
 ...
 10000000 = Maximum negative adjustment; subtracts 512 RTC clock pulses every one minute

- Note 1:** The RCFGAL register is only affected by a POR.
2: A write to the RTCEN bit is only allowed when RTCWREN = 1.
3: This bit is read-only. It is cleared to '0' on a write to the lower half of the MINSEC register.

REGISTER 20-2: PADCFG1: PAD CONFIGURATION CONTROL REGISTER

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

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	RTSECSEL ⁽¹⁾	PMPTTL
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 15-2 **Unimplemented:** Read as '0'
 bit 1 **RTSECSEL:** RTCC Seconds Clock Output Select bit⁽¹⁾
 1 = RTCC seconds clock is selected for the RTCC pin
 0 = RTCC alarm pulse is selected for the RTCC pin
 bit 0 **PMPTTL:** PMP Module TTL Input Buffer Select bit
 1 = PMP module inputs (PMDx, PMCS1) use TTL input buffers
 0 = PMP module inputs use Schmitt Trigger input buffers

- Note 1:** To enable the actual RTCC output, the RTCOE (RCFGAL<10>)) bit must also be set.

PIC24FJ256GB110 FAMILY

20.2 Calibration

The real-time crystal input can be calibrated using the periodic auto-adjust feature. When properly calibrated, the RTCC can provide an error of less than 3 seconds per month. This is accomplished by finding the number of error clock pulses for one minute and storing the value into the lower half of the RCFGAL register. The 8-bit signed value loaded into the lower half of RCFGAL is multiplied by four and will either be added or subtracted from the RTCC timer, once every minute. Refer to the steps below for RTCC calibration:

1. Using another timer resource on the device, the user must find the error of the 32.768 kHz crystal.
2. Once the error is known, it must be converted to the number of error clock pulses per minute and loaded into the RCFGAL register.

EQUATION 20-1: RTCC CALIBRATION

Error (clocks per minute) = (Ideal Frequency [†] – Measured Frequency) * 60 [†] Ideal frequency = 32,768 Hz

3. a) If the oscillator is faster than ideal (negative result from step 2), the RCFGAL register value needs to be negative. This causes the specified number of clock pulses to be subtracted from the timer counter once every minute.
b) If the oscillator is slower than ideal (positive result from step 2) the RCFGAL register value needs to be positive. This causes the specified number of clock pulses to be added to the timer counter once every minute.
4. Divide the number of error clocks per minute by 4 to get the correct CAL value and load the RCFGAL register with the correct value.
(Each 1-bit increment in CAL adds or subtracts 4 pulses).

Writes to the lower half of the RCFGAL register should only occur when the timer is turned off, or immediately after the rising edge of the seconds pulse.

Note: It is up to the user to include, in the error value, the initial error of the crystal, drift due to temperature and drift due to crystal aging.
--

20.3 Alarm

- Configurable from half second to one year
- Enabled using the ALRMEN bit (ALCFGRPT<15>, Register 20-3)
- One-time alarm and repeat alarm options available

20.3.1 CONFIGURING THE ALARM

The alarm feature is enabled using the ALRMEN bit. This bit is cleared when an alarm is issued. Writes to ALRMVAL should only take place when ALRMEN = 0.

As shown in Figure 20-2, the interval selection of the alarm is configured through the AMASK bits (ALCFGRPT<13:10>). These bits determine which and how many digits of the alarm must match the clock value for the alarm to occur.

The alarm can also be configured to repeat based on a preconfigured interval. The amount of times this occurs once the alarm is enabled is stored in the ARPT bits, ARPT<7:0> (ALCFGRPT<7:0>). When the value of the ARPT bits equals 00h and the CHIME bit (ALCFGRPT<14>) is cleared, the repeat function is disabled and only a single alarm will occur. The alarm can be repeated up to 255 times by loading ARPT<7:0> with FFh.

After each alarm is issued, the value of the ARPT bits is decremented by one. Once the value has reached 00h, the alarm will be issued one last time, after which the ALRMEN bit will be cleared automatically and the alarm will turn off.

Indefinite repetition of the alarm can occur if the CHIME bit = 1. Instead of the alarm being disabled when the value of the ARPT bits reaches 00h, it rolls over to FFh and continues counting indefinitely while CHIME is set.

20.3.2 ALARM INTERRUPT

At every alarm event, an interrupt is generated. In addition, an alarm pulse output is provided that operates at half the frequency of the alarm. This output is completely synchronous to the RTCC clock and can be used as a trigger clock to other peripherals.

Note: Changing any of the registers, other than the RCFGAL and ALCFGRPT registers and the CHIME bit while the alarm is enabled (ALRMEN = 1), can result in a false alarm event leading to a false alarm interrupt. To avoid a false alarm event, the timer and alarm values should only be changed while the alarm is disabled (ALRMEN = 0). It is recommended that the ALCFGRPT register and CHIME bit be changed when RTCSYNC = 0.

PIC24FJ256GB110 FAMILY

FIGURE 22-2: 10-BIT A/D CONVERTER ANALOG INPUT MODEL

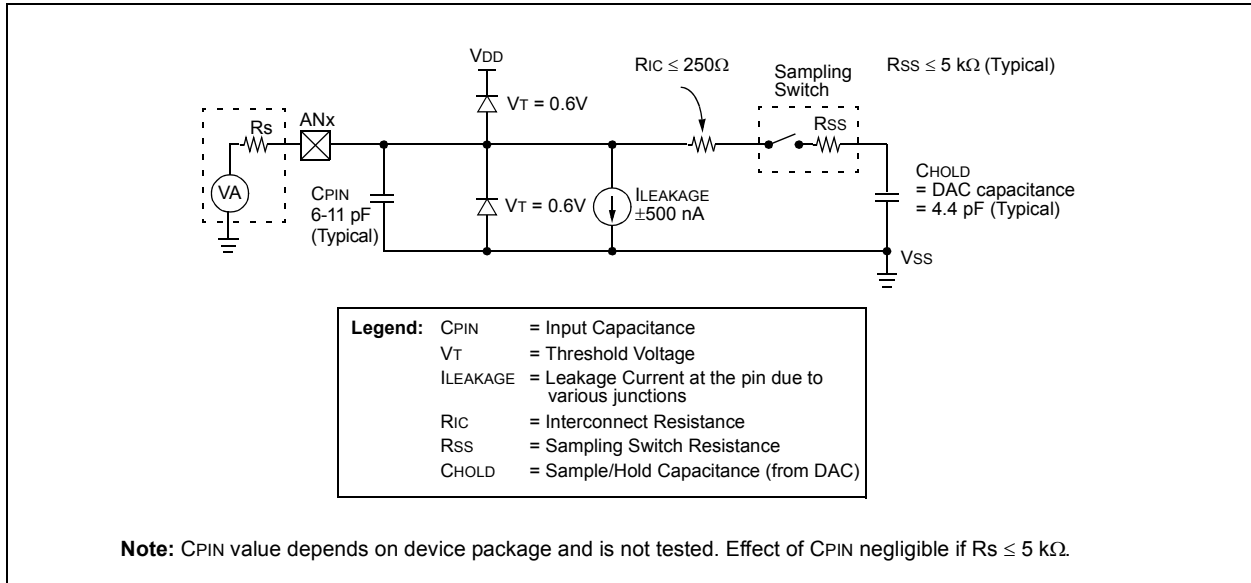
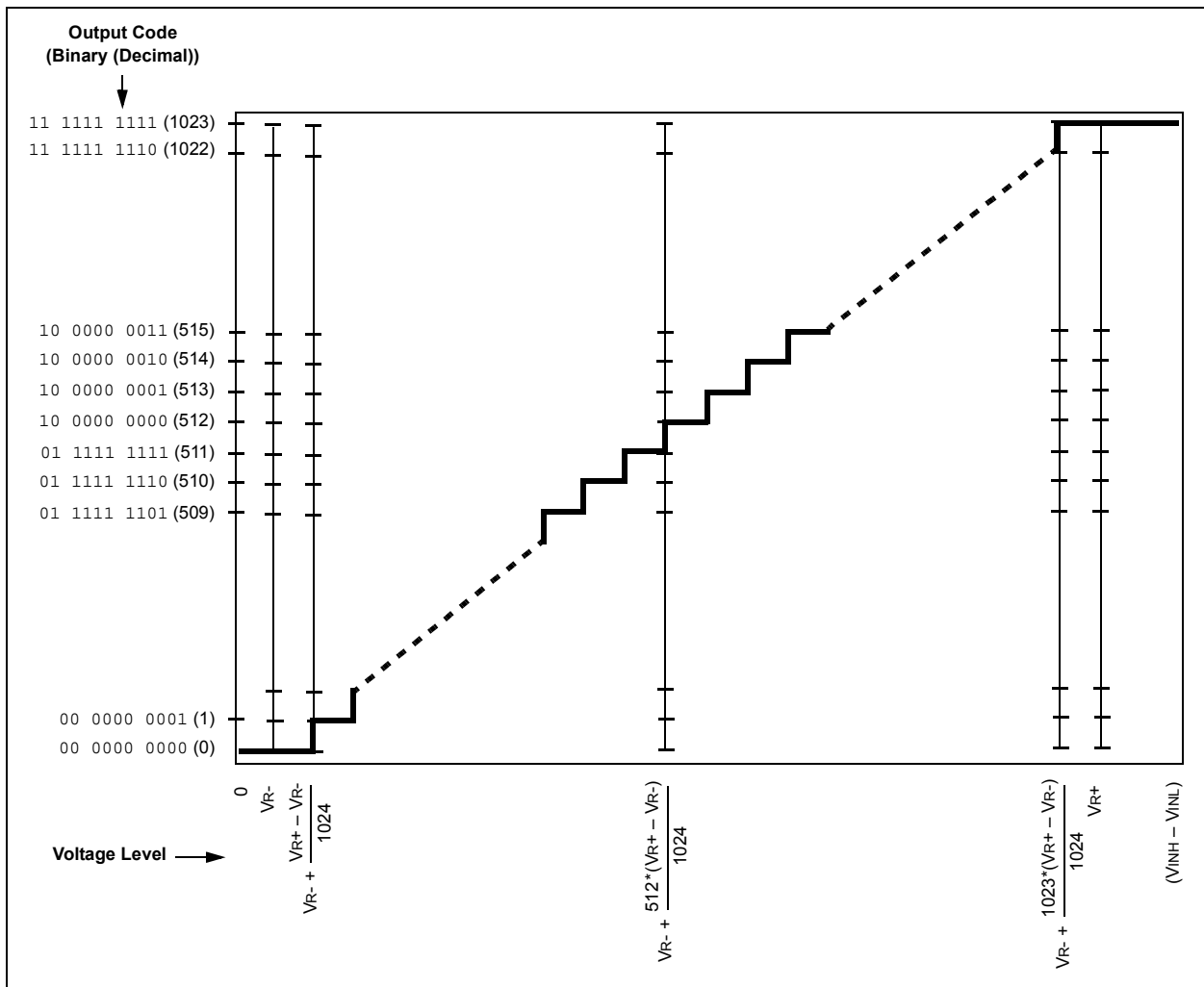


FIGURE 22-3: A/D TRANSFER FUNCTION



PIC24FJ256GB110 FAMILY

25.2 Measuring Time

Time measurements on the pulse width can be similarly performed, using the A/D module's internal capacitor (CAD) and a precision resistor for current calibration. Figure 25-2 shows the external connections used for time measurements, and how the CTMU and A/D modules are related in this application. This example also shows both edge events coming from the external CTEDG pins, but other configurations using internal edge sources are possible. A detailed discussion on measuring capacitance and time with the CTMU module is provided in the "PIC24F Family Reference Manual".

25.3 Pulse Generation and Delay

The CTMU module can also generate an output pulse with edges that are not synchronous with the device's system clock. More specifically, it can generate a pulse with a programmable delay from an edge event input to the module.

When the module is configured for pulse generation delay by setting the TGEN bit (CTMUCON<12>), the internal current source is connected to the B input of Comparator 2. A capacitor (CDELAY) is connected to the Comparator 2 pin, C2INB, and the comparator voltage reference, CVREF, is connected to C2INA. CVREF is then configured for a specific trip point. The module begins to charge CDELAY when an edge event is detected. When CDELAY charges above the CVREF trip point, a pulse is output on CTPLS. The length of the pulse delay is determined by the value of CDELAY and the CVREF trip point.

Figure 25-3 shows the external connections for pulse generation, as well as the relationship of the different analog modules required. While CTEDG1 is shown as the input pulse source, other options are available. A detailed discussion on pulse generation with the CTMU module is provided in the "PIC24F Family Reference Manual".

FIGURE 25-2: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR TIME MEASUREMENT

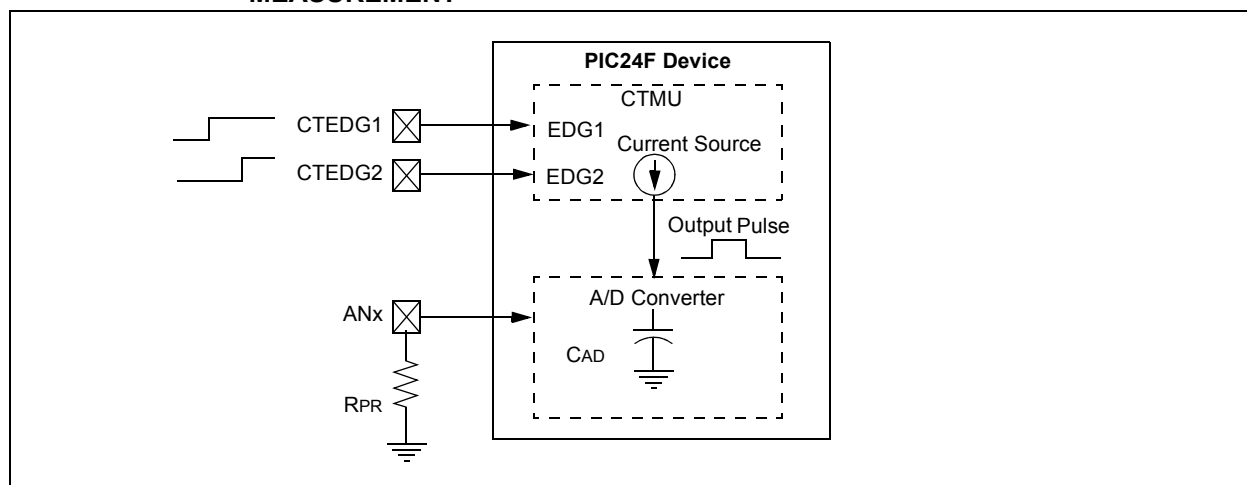
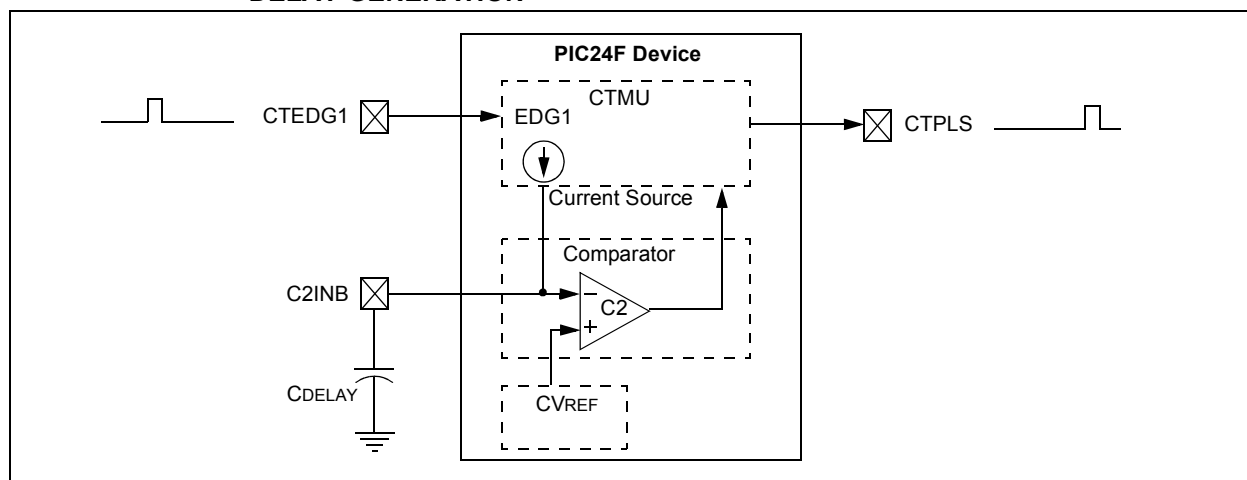


FIGURE 25-3: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR PULSE DELAY GENERATION



PIC24FJ256GB110 FAMILY

TABLE 28-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
BTSS	BTSS $f, \#bit4$	Bit Test f , Skip if Set	1	1 (2 or 3)	None
	BTSS $Ws, \#bit4$	Bit Test Ws , Skip if Set	1	1 (2 or 3)	None
BTST	BTST $f, \#bit4$	Bit Test f	1	1	Z
	BTST.C $Ws, \#bit4$	Bit Test Ws to C	1	1	C
	BTST.Z $Ws, \#bit4$	Bit Test Ws to Z	1	1	Z
	BTST.C Ws, Wb	Bit Test $Ws < Wb >$ to C	1	1	C
	BTST.Z Ws, Wb	Bit Test $Ws < Wb >$ to Z	1	1	Z
BTSTS	BTSTS $f, \#bit4$	Bit Test then Set f	1	1	Z
	BTSTS.C $Ws, \#bit4$	Bit Test Ws to C, then Set	1	1	C
	BTSTS.Z $Ws, \#bit4$	Bit Test Ws to Z, then Set	1	1	Z
CALL	CALL $lit23$	Call Subroutine	2	2	None
	CALL Wn	Call Indirect Subroutine	1	2	None
CLR	CLR f	$f = 0x0000$	1	1	None
	CLR WREG	WREG = $0x0000$	1	1	None
	CLR Ws	$Ws = 0x0000$	1	1	None
CLRWDT	CLRWDT	Clear Watchdog Timer	1	1	WDTO, Sleep
COM	COM f	$f = \bar{f}$	1	1	N, Z
	COM $f, WREG$	WREG = \bar{f}	1	1	N, Z
	COM Ws, Wd	$Wd = \overline{Ws}$	1	1	N, Z
CP	CP f	Compare f with WREG	1	1	C, DC, N, OV, Z
	CP $Wb, \#lit5$	Compare Wb with $lit5$	1	1	C, DC, N, OV, Z
	CP Wb, Ws	Compare Wb with Ws ($Wb - Ws$)	1	1	C, DC, N, OV, Z
CP0	CP0 f	Compare f with $0x0000$	1	1	C, DC, N, OV, Z
	CP0 Ws	Compare Ws with $0x0000$	1	1	C, DC, N, OV, Z
CPB	CPB f	Compare f with WREG, with Borrow	1	1	C, DC, N, OV, Z
	CPB $Wb, \#lit5$	Compare Wb with $lit5$, with Borrow	1	1	C, DC, N, OV, Z
	CPB Wb, Ws	Compare Wb with Ws , with Borrow ($Wb - Ws - C$)	1	1	C, DC, N, OV, Z
CPSEQ	CPSEQ Wb, Wn	Compare Wb with Wn , Skip if =	1	1 (2 or 3)	None
CPSGT	CPSGT Wb, Wn	Compare Wb with Wn , Skip if >	1	1 (2 or 3)	None
CPSLT	CPSLT Wb, Wn	Compare Wb with Wn , Skip if <	1	1 (2 or 3)	None
CPSNE	CPSNE Wb, Wn	Compare Wb with Wn , Skip if \neq	1	1 (2 or 3)	None
DAW	DAW.b Wn	$Wn =$ Decimal Adjust Wn	1	1	C
DEC	DEC f	$f = f - 1$	1	1	C, DC, N, OV, Z
	DEC $f, WREG$	WREG = $f - 1$	1	1	C, DC, N, OV, Z
	DEC Ws, Wd	$Wd = Ws - 1$	1	1	C, DC, N, OV, Z
DEC2	DEC2 f	$f = f - 2$	1	1	C, DC, N, OV, Z
	DEC2 $f, WREG$	WREG = $f - 2$	1	1	C, DC, N, OV, Z
	DEC2 Ws, Wd	$Wd = Ws - 2$	1	1	C, DC, N, OV, Z
DISI	DISI $\#lit14$	Disable Interrupts for k Instruction Cycles	1	1	None
DIV	DIV.SW Wm, Wn	Signed 16/16-bit Integer Divide	1	18	N, Z, C, OV
	DIV.SD Wm, Wn	Signed 32/16-bit Integer Divide	1	18	N, Z, C, OV
	DIV.UW Wm, Wn	Unsigned 16/16-bit Integer Divide	1	18	N, Z, C, OV
	DIV.UD Wm, Wn	Unsigned 32/16-bit Integer Divide	1	18	N, Z, C, OV
EXCH	EXCH Wns, Wnd	Swap Wns with Wnd	1	1	None
FF1L	FF1L Ws, Wnd	Find First One from Left (MSb) Side	1	1	C
FF1R	FF1R Ws, Wnd	Find First One from Right (LSb) Side	1	1	C

PIC24FJ256GB110 FAMILY

29.1 DC Characteristics

FIGURE 29-1: PIC24FJ256GB110 FAMILY VOLTAGE-FREQUENCY GRAPH (INDUSTRIAL)

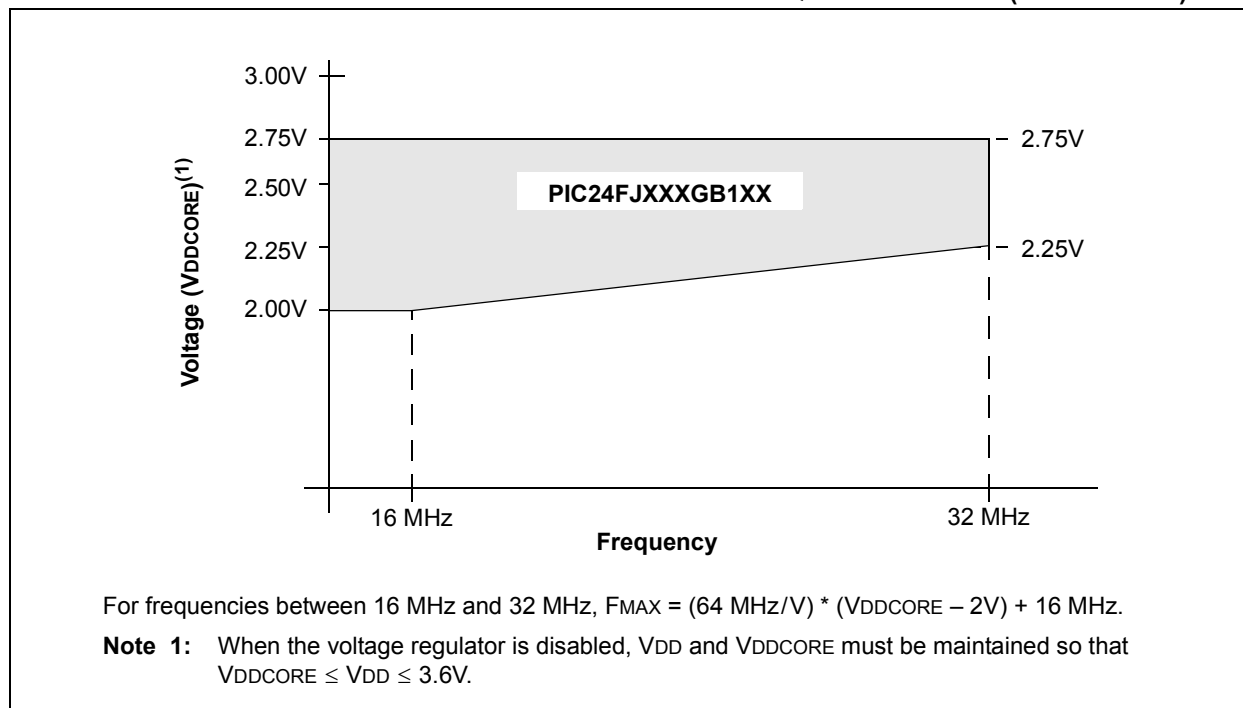


TABLE 29-1: THERMAL OPERATING CONDITIONS

Rating	Symbol	Min	Typ	Max	Unit
PIC24FJ256GB110 Family:					
Operating Junction Temperature Range	TJ	-40	—	+125	°C
Operating Ambient Temperature Range	TA	-40	—	+85	°C
Power Dissipation: Internal Chip Power Dissipation: $P_{INT} = V_{DD} \times (I_{DD} - \sum I_{OH})$ I/O Pin Power Dissipation: $P_{I/O} = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$	PD	PINT + PI/O			W
Maximum Allowed Power Dissipation	PDMAX	$(T_J - T_A)/\theta_{JA}$			W

TABLE 29-2: THERMAL PACKAGING CHARACTERISTICS

Characteristic	Symbol	Typ	Max	Unit	Notes
Package Thermal Resistance, 14x14x1 mm TQFP	θ_{JA}	50.0	—	°C/W	(Note 1)
Package Thermal Resistance, 12x12x1 mm TQFP	θ_{JA}	69.4	—	°C/W	(Note 1)
Package Thermal Resistance, 10x10x1 mm TQFP	θ_{JA}	76.6	—	°C/W	(Note 1)
Package Thermal Resistance, 9x9x0.9 mm QFN	θ_{JA}	28.0	—	°C/W	(Note 1)

Note 1: Junction to ambient thermal resistance, Theta-JA (θ_{JA}) numbers are achieved by package simulations.