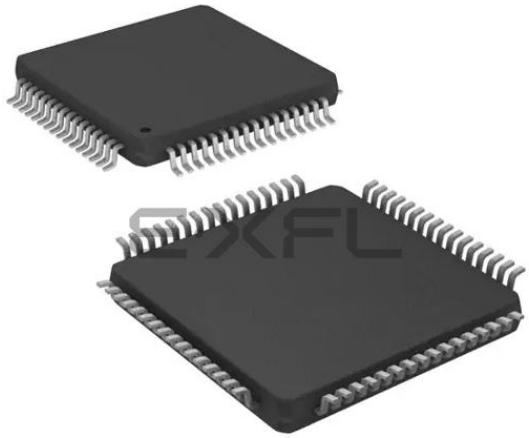


Welcome to E-XFL.COM

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	512KB (170K 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/pic24fj512ga606t-i-pt

TABLE 7: COMPLETE PIN FUNCTION DESCRIPTIONS (PIC24FJXXXGB610 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**/USBID/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	D-/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.l

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			
RF0	58	58	87	87	B6	B6	I/O	DIG/ST	PORTF Digital I/Os
RF1	59	59	88	88	A6	A6	I/O	DIG/ST	
RF2	34	—	52	52	K11	K11	I/O	DIG/ST	
RF3	33	33	51	51	K10	K10	I/O	DIG/ST	
RF4	31	31	49	49	L10	L10	I/O	DIG/ST	
RF5	32	32	50	50	L11	L11	I/O	DIG/ST	
RF6	35	—	55	—	H9	—	I/O	DIG/ST	
RF7	—	34	54	54	H8	H8	I/O	DIG/ST	
RF8	—	—	53	53	J10	J10	I/O	DIG/ST	
RF12	—	—	40	40	K6	K6	I/O	DIG/ST	
RF13	—	—	39	39	L6	L6	I/O	DIG/ST	
RG0	—	—	90	90	A5	A5	I/O	DIG/ST	PORTG Digital I/Os
RG1	—	—	89	89	E6	E6	I/O	DIG/ST	
RG2	37	37	57	57	H10	H10	I/O	DIG/ST	
RG3	36	36	56	56	J11	J11	I/O	DIG/ST	
RG6	4	4	10	10	E3	E3	I/O	DIG/ST	
RG7	5	5	11	11	F4	F4	I/O	DIG/ST	
RG8	6	6	12	12	F2	F2	I/O	DIG/ST	
RG9	8	8	14	14	F3	F3	I/O	DIG/ST	
RG12	—	—	96	96	C3	C3	I/O	DIG/ST	
RG13	—	—	97	97	A3	A3	I/O	DIG/ST	
RG14	—	—	95	95	C4	C4	I/O	DIG/ST	
RG15	—	—	1	1	B2	B2	I/O	DIG/ST	

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

TABLE 4-6: SFR MAP: 0200h BLOCK (CONTINUED)

File Name	Address	All Resets	File Name	Address	All Resets
MULTIPLE OUTPUT CAPTURE/COMPARE/PWM (CONTINUED)			**MULTIPLE OUTPUT CAPTURE/COMPARE/PWM (CONTINUED)**		
CCP2RAL	02A8	0000	CCP3PRL	02C8	FFFF
CCP2RAH	02AA	0000	CCP3PRH	02CA	FFFF
CCP2RBL	02AC	0000	CCP3RAL	02CC	0000
CCP2RBH	02AE	0000	CCP3RAH	02CE	0000
CCP2BUFL	02B0	0000	CCP3RBL	02D0	0000
CCP2BUFH	02B2	0000	CCP3RBH	02D2	0000
CCP3CON1L	02B4	0000	CCP3BUFL	02D4	0000
CCP3CON1H	02B6	0000	CCP3BUFH	02D6	0000
CCP3CON2L	02B8	0000	**COMPARATORS**		
CCP3CON2H	02BA	0100	CMSTAT	02E6	0000
CCP3CON3L	02BC	0000	CVRCON	02E8	00xx
CCP3CON3H	02BE	0000	CM1CON	02EA	0000
CCP3STATL	02C0	00x0	CM2CON	02EC	0000
CCP3STATH	02C2	0000	CM3CON	02EE	0000
CCP3TMRL	O2C4	0000	**ANALOG CONFIGURATION**		
CCP3TMRH	02C6	0000	ANCFG	02F4	0000

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

4.2.5.1 Data Read from EDS

In order to read the data from the EDS space, first, an Address Pointer is set up by loading the required EDS page number into the DSRPAG register and assigning the offset address to one of the W registers. Once the above assignment is done, the EDS window is enabled by setting bit 15 of the Working register which is assigned with the offset address; then, the contents of the pointed EDS location can be read.

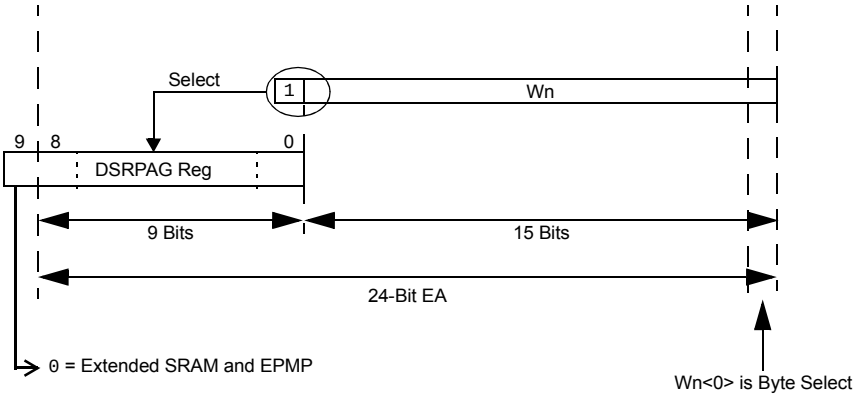
Figure 4-5 illustrates how the EDS space address is generated for read operations.

When the Most Significant bit (MSb) of EA is '1' and DSRPAG<9> = 0, the lower 9 bits of DSRPAG are concatenated to the lower 15 bits of EA to form a 24-bit EDS space address for read operations.

Example 4-1 shows how to read a byte, word and double word from EDS.

Note:	All read operations from EDS space have an overhead of one instruction cycle. Therefore, a minimum of two instruction cycles are required to complete an EDS read. For EDS reads under the REPEAT instruction; the first two accesses take three cycles and the subsequent accesses take one cycle.

FIGURE 4-5: EDS ADDRESS GENERATION FOR READ OPERATIONS



EXAMPLE 4-1: EDS READ CODE IN ASSEMBLY

```
; Set the EDS page from where the data to be read
    mov         #0x0002, w0
    mov         w0, DSRPAG      ;page 2 is selected for read
    mov         #0x0800, w1     ;select the location (0x800) to be read
    bset        w1, #15         ;set the MSB of the base address, enable EDS mode

;Read a byte from the selected location
    mov.b       [w1++], w2      ;read Low byte
    mov.b       [w1++], w3      ;read High byte

;Read a word from the selected location
    mov         [w1], w2        ;

;Read Double - word from the selected location
    mov.d       [w1], w2        ;two word read, stored in w2 and w3
```

5.1 Summary of DMA Operations

The DMA Controller is capable of moving data between addresses according to a number of different parameters. Each of these parameters can be independently configured for any transaction; in addition, any or all of the DMA channels can independently perform a different transaction at the same time. Transactions are classified by these parameters:

- Source and destination (SFRs and data RAM)
- Data size (byte or word)
- Trigger source
- Transfer mode (One-Shot, Repeated or Continuous)
- Addressing modes (fixed address or address blocks, with or without address increment/decrement)

In addition, the DMA Controller provides channel priority arbitration for all channels.

5.1.1 SOURCE AND DESTINATION

Using the DMA Controller, data may be moved between any two addresses in the Data Space. The SFR space (0000h to 07FFh), or the data RAM space (0800h to FFFFh), can serve as either the source or the destination. Data can be moved between these areas in either direction or between addresses in either area. The four different combinations are shown in Figure 5-2.

If it is necessary to protect areas of data RAM, the DMA Controller allows the user to set upper and lower address boundaries for operations in the Data Space above the SFR space. The boundaries are set by the DMAH and DMAL Limit registers. If a DMA channel attempts an operation outside of the address boundaries, the transaction is terminated and an interrupt is generated.

5.1.2 DATA SIZE

The DMA Controller can handle both 8-bit and 16-bit transactions. Size is user-selectable using the SIZE bit (DMACHn<1>). By default, each channel is configured for word-sized transactions. When byte-sized transactions are chosen, the LSb of the source and/or destination address determines if the data represents the upper or lower byte of the data RAM location.

5.1.3 TRIGGER SOURCE

The DMA Controller can use any one of the device's interrupt sources to initiate a transaction. The DMA Trigger sources are listed in reverse order of their natural interrupt priority and are shown in Table 5-1.

Since the source and destination addresses for any transaction can be programmed independently of the Trigger source, the DMA Controller can use any Trigger to perform an operation on any peripheral. This also allows DMA channels to be cascaded to perform more complex transfer operations.

5.1.4 TRANSFER MODE

The DMA Controller supports four types of data transfers, based on the volume of data to be moved for each Trigger.

- One-Shot: A single transaction occurs for each Trigger.
- Continuous: A series of back-to-back transactions occur for each Trigger; the number of transactions is determined by the DMACNTn transaction counter.
- Repeated One-Shot: A single transaction is performed repeatedly, once per Trigger, until the DMA channel is disabled.
- Repeated Continuous: A series of transactions are performed repeatedly, one cycle per Trigger, until the DMA channel is disabled.

All transfer modes allow the option to have the source and destination addresses, and counter value automatically reloaded after the completion of a transaction. Repeated mode transfers do this automatically.

5.1.5 ADDRESSING MODES

The DMA Controller also supports transfers between single addresses or address ranges. The four basic options are:

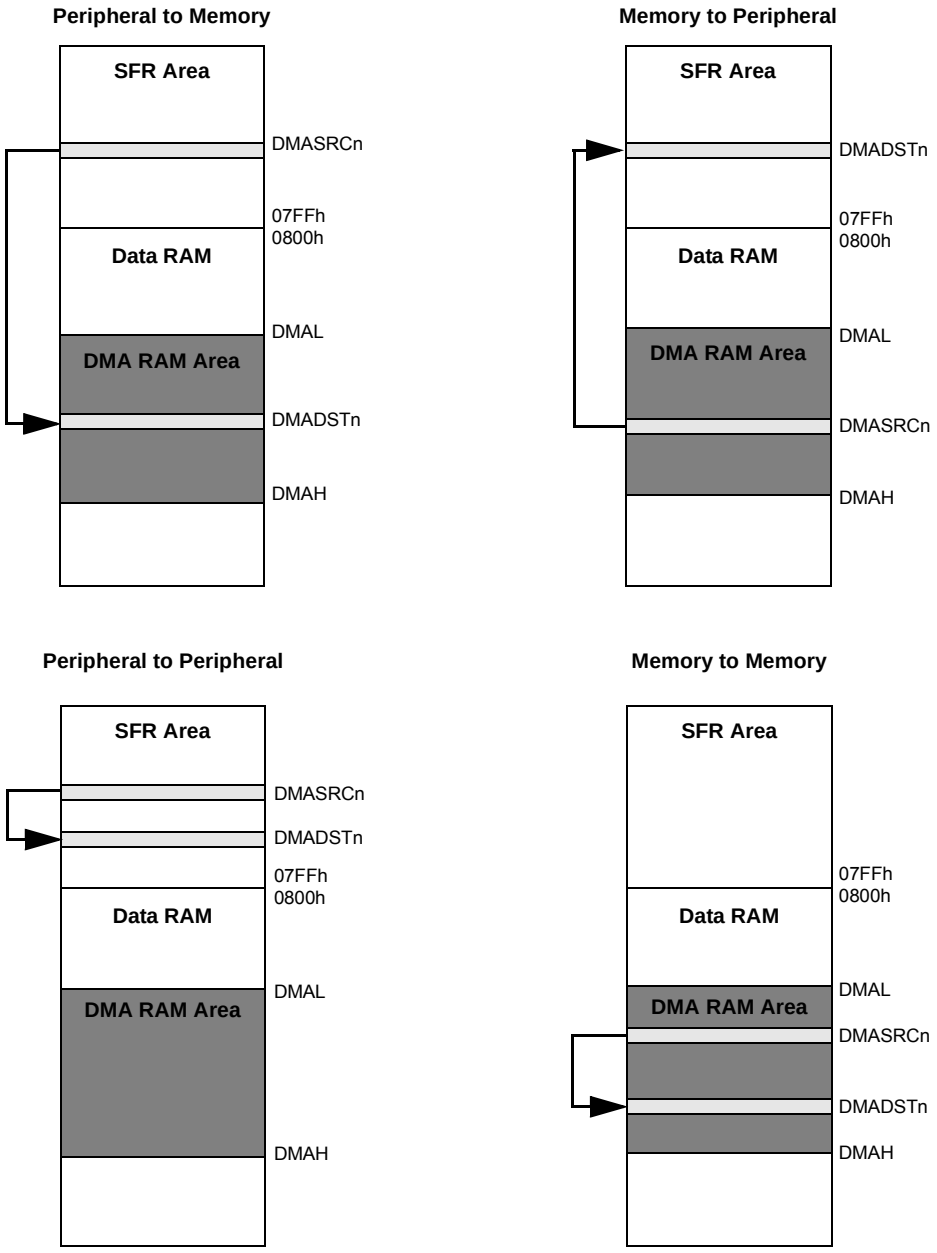
- Fixed-to-Fixed: Between two constant addresses
- Fixed-to-Block: From a constant source address to a range of destination addresses
- Block-to-Fixed: From a range of source addresses to a single, constant destination address
- Block-to-Block: From a range to source addresses to a range of destination addresses

The option to select auto-increment or auto-decrement of source and/or destination addresses is available for Block Addressing modes.

In addition to the four basic modes, the DMA Controller also supports Peripheral Indirect Addressing (PIA) mode, where the source or destination address is generated jointly by the DMA Controller and a PIA capable peripheral. When enabled, the DMA channel provides a base source and/or destination address, while the peripheral provides a fixed range offset address.

For PIC24FJ1024GA610/GB610 family devices, the 12-bit A/D Converter module is the only PIA capable peripheral. Details for its use in PIA mode are provided in **Section 25.0 "12-Bit A/D Converter with Threshold Detect"**.

FIGURE 5-2: TYPES OF DMA DATA TRANSFERS



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

REGISTER 9-7: OSCFDIV: OSCILLATOR FRACTIONAL DIVISOR REGISTER(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TRIM<0:7>							
bit 15							bit 8

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
TRIM8	—	—	—	—	—	—	—
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 **TRIM<0:8>:** Trim bits

Provides fractional additive to the DIV<14:0> value for the 1/2 period of the oscillator clock.
0000_0000_0 = 0/512 (0.0) divisor added to DIVx value
0000_0000_1 = 1/512 (0.001953125) divisor added to DIVx value
0000_0001_0 = 2/512 (0.00390625) divisor added to DIVx value
•
•
•
100000000 = 256/512 (0.5000) divisor added to DIVx value
•
•
•
1111_1111_0 = 510/512 (0.99609375) divisor added to DIVx value
1111_1111_1 = 511/512 (0.998046875) divisor added to DIVx value

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

Note 1: TRIMx values greater than zero are ONLY valid when DIVx values are greater than zero.

REGISTER 11-18: RPINR6: PERIPHERAL PIN SELECT INPUT REGISTER 6

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

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

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

bit 15-14 **Unimplemented:** Read as ‘0’

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

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

bit 5-0 **Reserved**: Maintain as ‘1’

REGISTER 11-19: RPINR7: PERIPHERAL PIN SELECT INPUT REGISTER 7

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	IC2R5	IC2R4	IC2R3	IC2R2	IC2R1	IC2R0
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
—	—	IC1R5	IC1R4	IC1R3	IC1R2	IC1R1	IC1R0
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 **IC2R<5:0>:** Assign Input Capture 2 (IC2) to Corresponding RPn or RPIn Pin bits

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

bit 5-0 **IC1R<5:0>:** Assign Input Capture 1 (IC1) to Corresponding RPn or RPIn Pin bits

REGISTER 11-48: RPOR12: PERIPHERAL PIN SELECT OUTPUT REGISTER 12

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP25R5	RP25R4	RP25R3	RP25R2	RP25R1	RP25R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP24R5	RP24R4	RP24R3	RP24R2	RP24R1	RP24R0
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 **RP25R<5:0>:** RP25 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP25 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP24R<5:0>:** RP24 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP24 (see Table 11-4 for peripheral function numbers).

REGISTER 11-49: RPOR13: PERIPHERAL PIN SELECT OUTPUT REGISTER 13

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP27R5	RP27R4	RP27R3	RP27R2	RP27R1	RP27R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP26R5	RP26R4	RP26R3	RP26R2	RP26R1	RP26R0
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 **RP27R<5:0>:** RP27 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP27 (see Table 11-4 for peripheral function numbers).

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

bit 5-0 **RP26R<5:0>:** RP26 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP26 (see Table 11-4 for peripheral function numbers).

12.0 TIMER1

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"*, **"Timers"** (DS39704), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

The Timer1 module is a 16-bit timer, which can serve as the time counter for the Real-Time Clock (RTC) or operate as a free-running, interval timer/counter. Timer1 can operate in three modes:

- 16-Bit Timer
- 16-Bit Synchronous Counter
- 16-Bit Asynchronous Counter

Timer1 also supports these features:

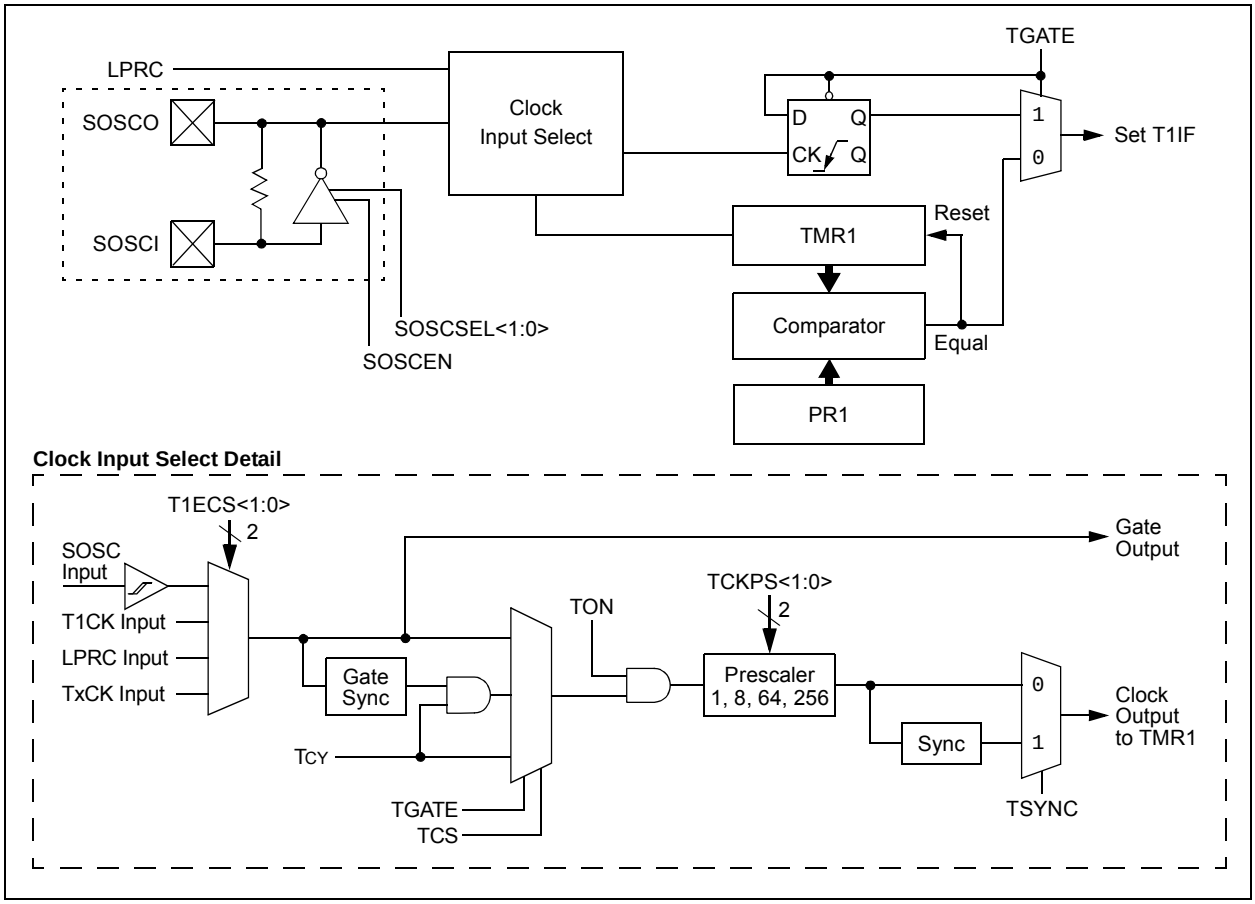
- Timer Gate Operation
- Selectable Prescaler Settings
- Timer Operation during CPU Idle and Sleep modes
- Interrupt on 16-Bit Period Register Match or Falling Edge of External Gate Signal

Figure 12-1 presents a block diagram of the 16-bit timer module.

To configure Timer1 for operation:

1. Clear the TON bit (= 0).
2. Select the timer prescaler ratio using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the TCS, TECS<1:0> and TGATE bits.
4. Set or clear the TSYNC bit to configure synchronous or asynchronous operation.
5. Load the timer period value into the PR1 register.
6. If interrupts are required, set the interrupt enable bit, T1IE. Use the priority bits, T1IP<2:0>, to set the interrupt priority.
7. Set the TON bit (= 1).

FIGURE 12-1: 16-BIT TIMER1 MODULE BLOCK DIAGRAM



15.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the OCxRS and OCxR registers. The OCxRS and OCxR registers can be written to at any time, but the duty cycle value is not latched until a match between PRy and TMRy occurs (i.e., the period is complete). This provides a double buffer for the PWM duty cycle and is essential for glitchless PWM operation.

Some important boundary parameters of the PWM duty cycle include:

- If OCxR, OCxRS and PRy are all loaded with 0000h, the OCx pin will remain low (0% duty cycle).
- If OCxRS is greater than PRy, the pin will remain high (100% duty cycle).

See Example 15-1 for PWM mode timing details. Table 15-1 and Table 15-2 show example PWM frequencies and resolutions for a device operating at 4 MIPS and 10 MIPS, respectively.

EQUATION 15-2: CALCULATION FOR MAXIMUM PWM RESOLUTION(1)

$$\text{Maximum PWM Resolution (bits)} = \frac{\log_{10}\left(\frac{F_{CY}}{FPWM \bullet (\text{Timer Prescale Value})}\right)}{\log_{10}(2)} \text{ bits}$$

Note 1: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

EXAMPLE 15-1: PWM PERIOD AND DUTY CYCLE CALCULATIONS(1)

1. Find the Timer Period register value for a desired PWM frequency of 52.08 kHz, where FOSC = 32 MHz with PLL (32 MHz device clock rate) and a Timer2 prescaler setting of 1:1.

   TCY = 2 * TOSC = 62.5 ns

   PWM Period = 1/PWM Frequency = 1/52.08 kHz = 19.2 µs

   PWM Period = (PR2 + 1) • TCY • (Timer2 Prescale Value)

   19.2 µs = (PR2 + 1) • 62.5 ns • 1

   PR2 = 306

2. Find the maximum resolution of the duty cycle that can be used with a 52.08 kHz frequency and a 32 MHz device clock rate:

   PWM Resolution = $\log_{10}(F_{CY}/F_{PWM})/\log_{10}2)$ bits

   = $(\log_{10}(16\text{ MHz}/52.08\text{ kHz})/\log_{10}2)$ bits

   = 8.3 bits

Note 1: Based on TCY = 2 * TOSC; Doze mode and PLL are disabled.

TABLE 15-1: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 4 MIPS (FCY = 4 MHz)(1)

PWM Frequency	7.6 Hz	61 Hz	122 Hz	977 Hz	3.9 kHz	31.3 kHz	125 kHz
Timer Prescaler Ratio	8	1	1	1	1	1	1
Period Register Value	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
Resolution (bits)	16	16	15	12	10	7	5

Note 1: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

TABLE 15-2: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 16 MIPS (FCY = 16 MHz)(1)

PWM Frequency	30.5 Hz	244 Hz	488 Hz	3.9 kHz	15.6 kHz	125 kHz	500 kHz
Timer Prescaler Ratio	8	1	1	1	1	1	1
Period Register Value	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
Resolution (bits)	16	16	15	12	10	7	5

Note 1: Based on FCY = FOSC/2; Doze mode and PLL are disabled.

16.4 Input Capture Mode

Input Capture mode is used to capture a timer value from an independent timer base upon an event on an input pin or other internal Trigger source. The input capture features are useful in applications requiring frequency (time period) and pulse measurement. Figure 16-6 depicts a simplified block diagram of the Input Capture mode.

Input Capture mode uses a dedicated 16/32-bit, synchronous, up counting timer for the capture function. The timer value is written to the FIFO when a capture event occurs. The internal value may be read (with a synchronization delay) using the CCPxTMRH/L registers.

To use Input Capture mode, the CCSEL bit (CCPxCON1L<4>) must be set. The T32 and MOD<3:0> bits are used to select the proper Capture mode, as shown in Table 16-3.

TABLE 16-3: INPUT CAPTURE MODES

MOD<3:0> (CCPxCON1L<3:0>)	T32 (CCPxCON1L<5>)	Operating Mode
0000	0	Edge Detect (16-bit capture)
0000	1	Edge Detect (32-bit capture)
0001	0	Every Rising (16-bit capture)
0001	1	Every Rising (32-bit capture)
0010	0	Every Falling (16-bit capture)
0010	1	Every Falling (32-bit capture)
0011	0	Every Rise/Fall (16-bit capture)
0011	1	Every Rise/Fall (32-bit capture)
0100	0	Every 4th Rising (16-bit capture)
0100	1	Every 4th Rising (32-bit capture)
0101	0	Every 16th Rising (16-bit capture)
0101	1	Every 16th Rising (32-bit capture)

FIGURE 16-6: INPUT CAPTURE x BLOCK DIAGRAM

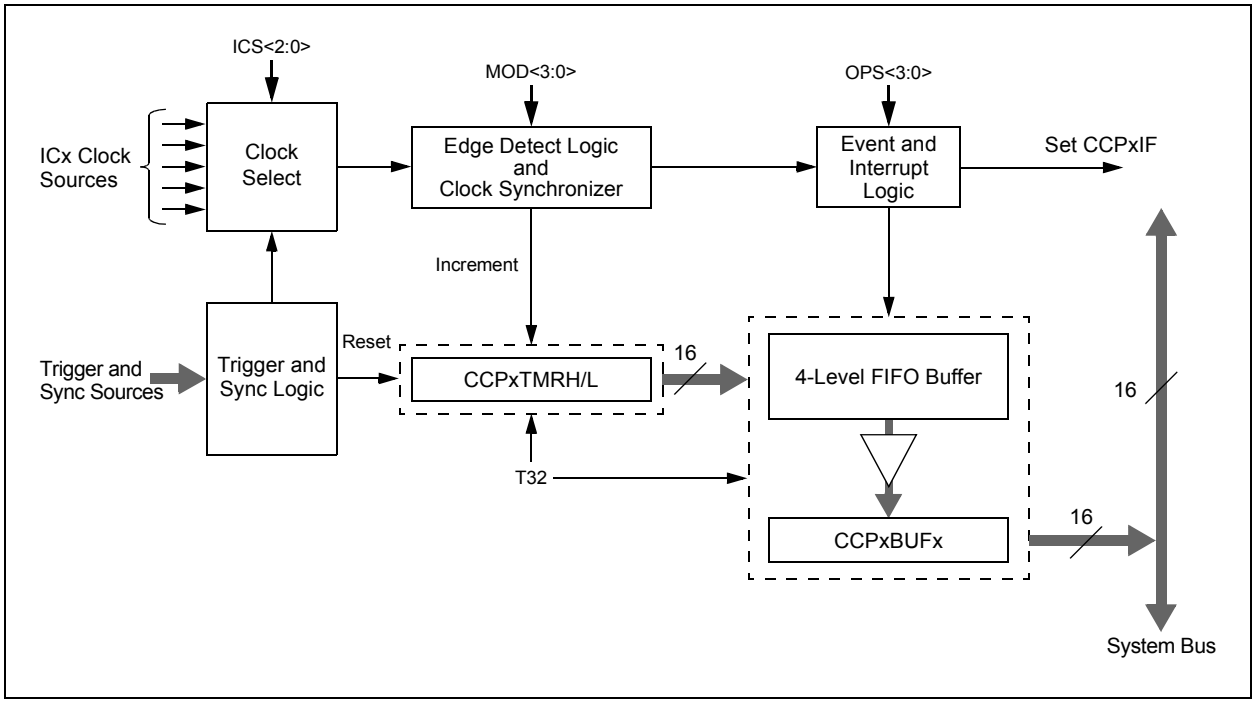


TABLE 16-5: SYNCHRONIZATION SOURCES

SYNC<4:0>	Synchronization Source
11111	None; Timer with Rollover on CCPxPR Match or FFFFh
11110	Reserved
11101	Reserved
11100	CTMU Trigger
11011	A/D Start Conversion
11010	CMP3 Trigger
11001	CMP2 Trigger
11000	CMP1 Trigger
10111	Reserved
10110	Reserved
10101	Reserved
10100	Reserved
10011	CLC4 Out
10010	CLC3 Out
10001	CLC2 Out
10000	CLC1 Out
01111	Reserved
01110	Reserved
01101	Reserved
01100	Reserved
01011	INT2 Pad
01010	INT1 Pad
01001	INT0 Pad
01000	SCCP7 Sync Out
00111	SCCP6 Sync Out
00110	SCCP5 Sync Out
00101	SCCP4 Sync Out
00100	MCCP3 Sync Out
00011	MCCP2 Sync Out
00010	MCCP1 Sync Out
00001	MCCPx/SCCPx Sync Out(1)
00000	MCCPx/SCCPx Timer Sync Out(1)

Note 1: CCP1 when connected to CCP1, CCP2 when connected to CCP2, etc.

REGISTER 17-1: SPIxCON1L: SPIx CONTROL REGISTER 1 LOW (CONTINUED)

bit 8 **CKE:** SPIx Clock Edge Select bit(1)
1 = Transmit happens on transition from active clock state to Idle clock state
0 = Transmit happens on transition from Idle clock state to active clock state

bit 7 **SSEN:** Slave Select Enable bit (Slave mode)(2)
1 = $\overline{SSx}$ pin is used by the macro in Slave mode; $\overline{SSx}$ pin is used as the slave select input
0 = $\overline{SSx}$ pin is not used by the macro ($\overline{SSx}$ pin will be controlled by the port I/O)

bit 6 **CKP:** SPIx Clock Polarity Select bit
1 = Idle state for clock is a high level; active state is a low level
0 = Idle state for clock is a low level; active state is a high level

bit 5 **MSTEN:** Master Mode Enable bit
1 = Master mode
0 = Slave mode

bit 4 **DISSDI:** Disable SDIx Input Port bit
1 = SDIx pin is not used by the module; pin is controlled by the port function
0 = SDIx pin is controlled by the module

bit 3 **DISSCK:** Disable SCKx Output Port bit
1 = SCKx pin is not used by the module; pin is controlled by the port function
0 = SCKx pin is controlled by the module

bit 2 **MCLKEN:** Master Clock Enable bit(3)
1 = REFO output is used by the BRG
0 = Peripheral clock is used by the BRG

bit 1 **SPIFE:** Frame Sync Pulse Edge Select bit
1 = Frame Sync pulse (Idle-to-active edge) coincides with the first bit clock
0 = Frame Sync pulse (Idle-to-active edge) precedes the first bit clock

bit 0 **ENHBUF:** Enhanced Buffer Mode Enable bit
1 = Enhanced Buffer mode is enabled
0 = Enhanced Buffer mode is disabled

Note 1: When AUDEN = 1, this module functions as if CKE = 0, regardless of its actual value.
2: When FRMEN = 1, SSEN is not used.
3: MCLKEN can only be written when the SPIEN bit = 0.
4: This channel is not meaningful for DSP/PCM mode as LRC follows the FRMSYPW bit.

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 21-3: PMCON3: EPMP CONTROL REGISTER 3

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
PTWREN	PTRDEN	PTBE1EN	PTBE0EN	—	AWAITM1	AWAITM0	AWAITE
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 **PTWREN:** Write/Enable Strobe Port Enable bit
1 = PMWR/PMENB port is enabled
0 = PMWR/PMENB port is disabled

bit 14 **PTRDEN:** Read/Write Strobe Port Enable bit
1 = PMRD/$\overline{\text{PMWR}}$ port is enabled
0 = PMRD/$\overline{\text{PMWR}}$ port is disabled

bit 13 **PTBE1EN:** High Nibble/Byte Enable Port Enable bit
1 = PMBE1 port is enabled
0 = PMBE1 port is disabled

bit 12 **PTBE0EN:** Low Nibble/Byte Enable Port Enable bit
1 = PMBE0 port is enabled
0 = PMBE0 port is disabled

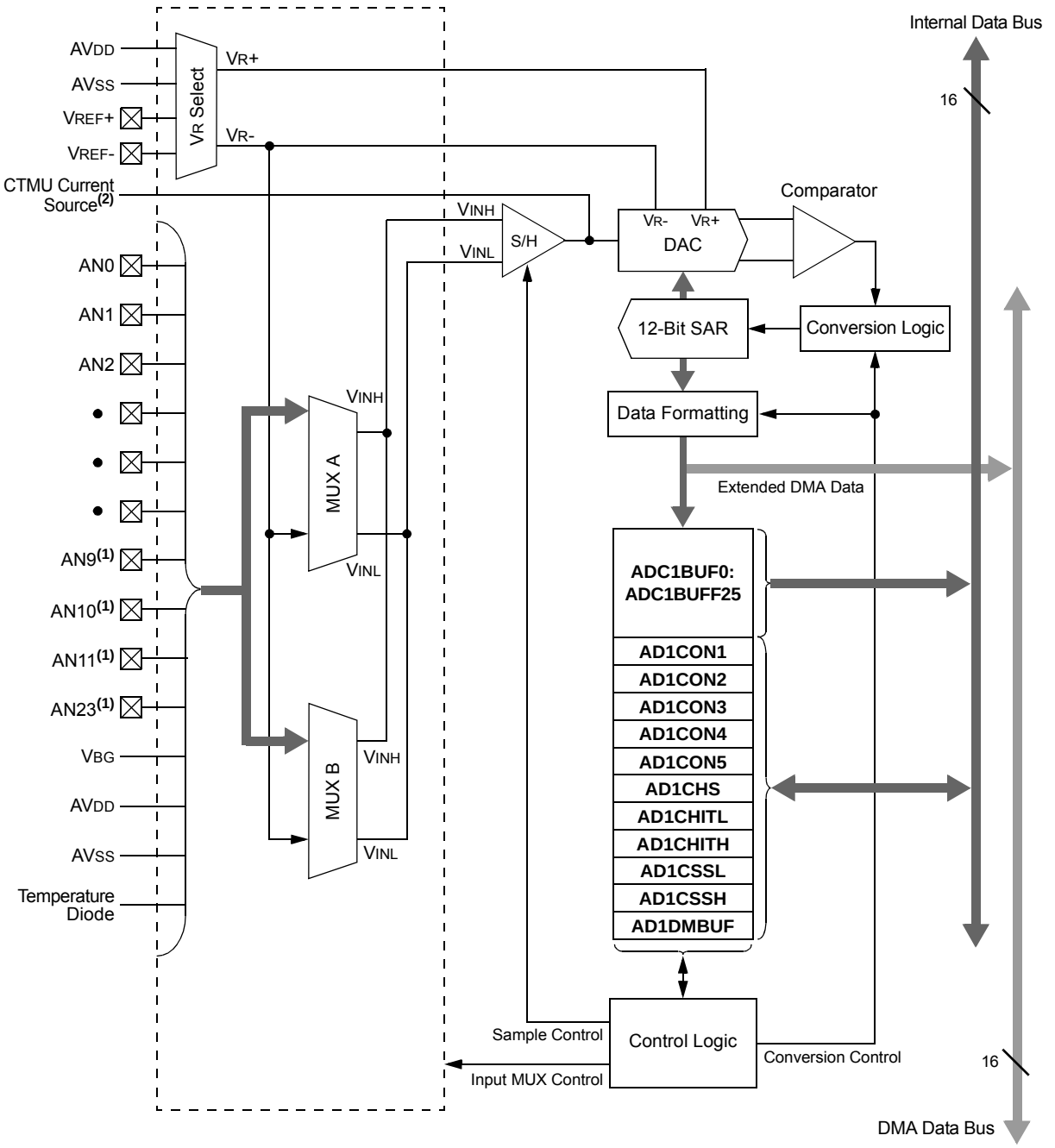
bit 11 **Unimplemented:** Read as '0'

bit 10-9 **AWAITM<1:0>:** Address Latch Strobe Wait States bits
11 = Wait of 3½ TCY
10 = Wait of 2½ TCY
01 = Wait of 1½ TCY
00 = Wait of ½ TCY

bit bit 8 **AWAITE:** Address Hold After Address Latch Strobe Wait States bits
1 = Wait of 1¼ TCY
0 = Wait of ¼ TCY

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

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



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

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

27.0 COMPARATOR VOLTAGE REFERENCE

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"*, **"Dual Comparator Module"** (DS39710), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

27.1 Configuring the Comparator Voltage Reference

The voltage reference module is controlled through the CVRCON register (Register 27-1). The comparator voltage reference provides two ranges of output voltage, each with 32 distinct levels.

The comparator reference supply voltage can come from either VDD and VSS, or the external VREF+ and VREF-. The voltage source is selected by the CVRSS bit (CVRCON<5>).

The settling time of the comparator voltage reference must be considered when changing the CVREF output.

FIGURE 27-1: COMPARATOR VOLTAGE REFERENCE BLOCK DIAGRAM

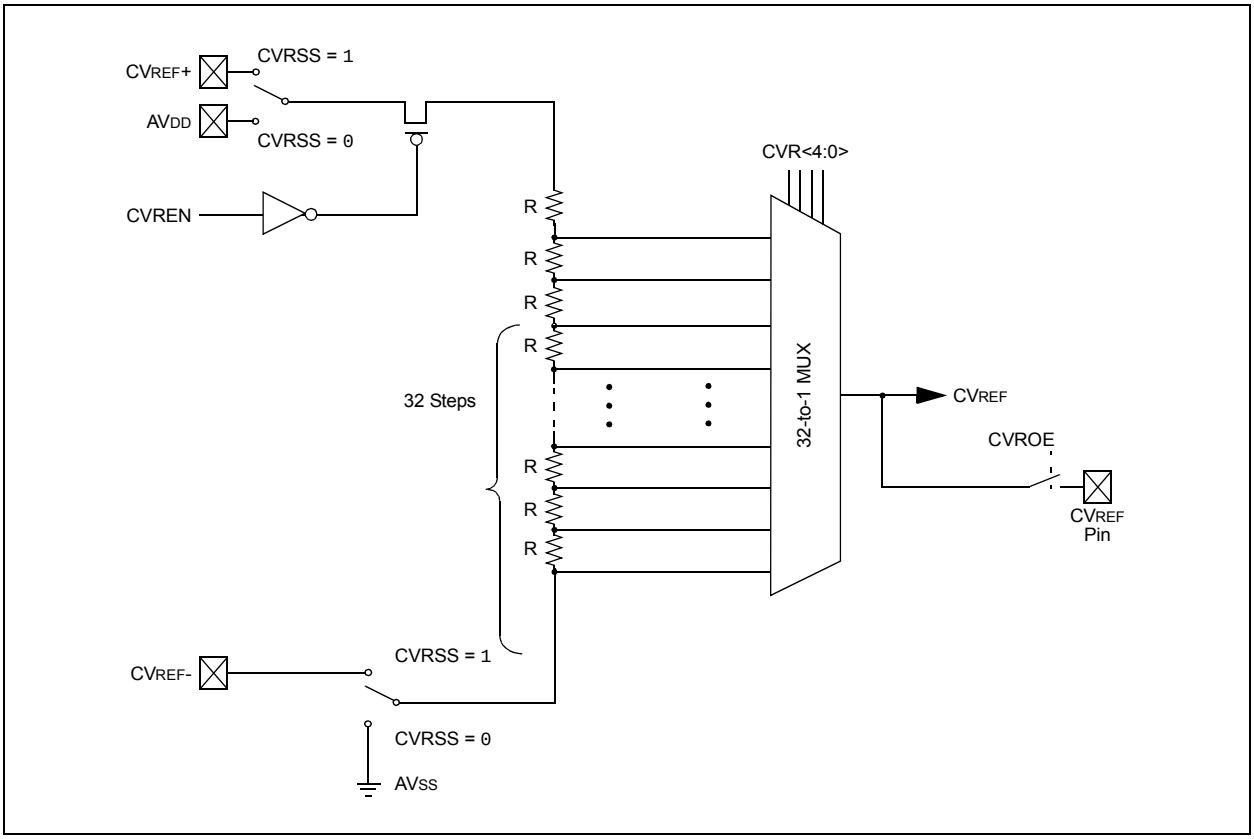
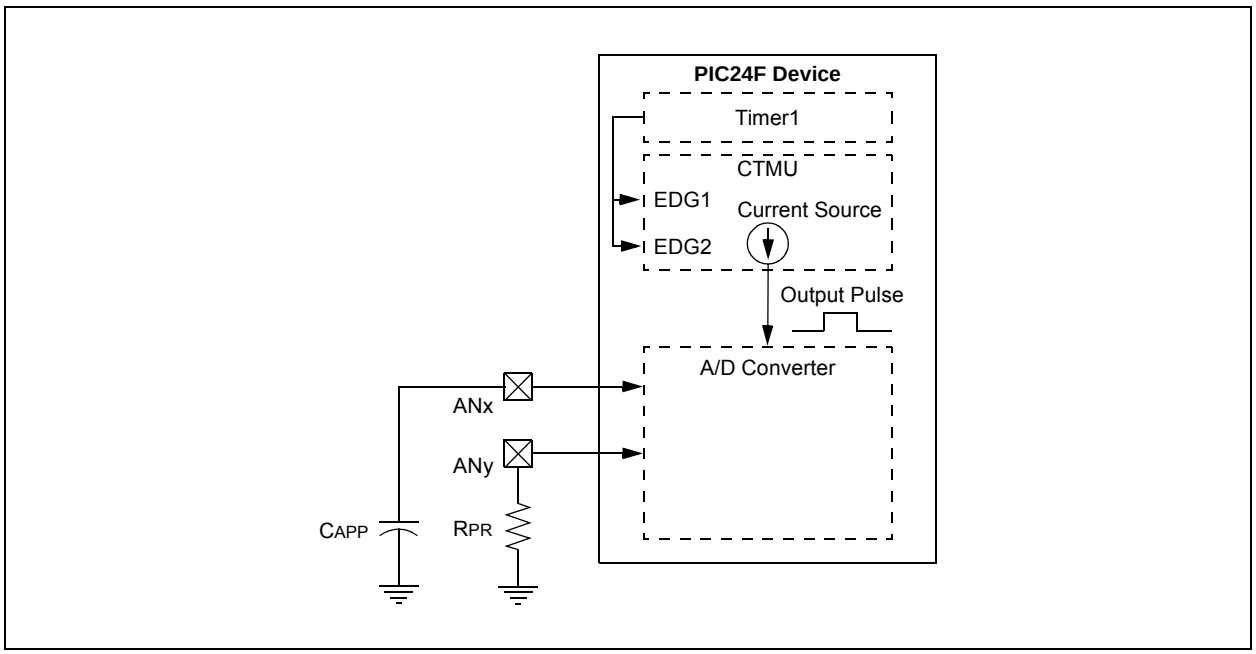


FIGURE 28-1: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR CAPACITANCE MEASUREMENT



28.2 Measuring Time/Routing Current Source to A/D Input Pin

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 28-2 displays 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 CTEDx pins, but other configurations using internal edge sources are possible.

This mode is enabled by clearing the TGEN bit (CTMUCON1L<12>). The current source is tied to the input of the A/D after the sampling switch. Therefore, the A/D bit, SAMP, must be set to '1' in order for the current to be routed through the channel selection MUX to the desired pin.

28.3 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 (CTMUCON1<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 28-3 illustrates the external connections for pulse generation, as well as the relationship of the different analog modules required. While CTED1 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 *"dsPIC33/PIC24 Family Reference Manual"*.

NOTES: