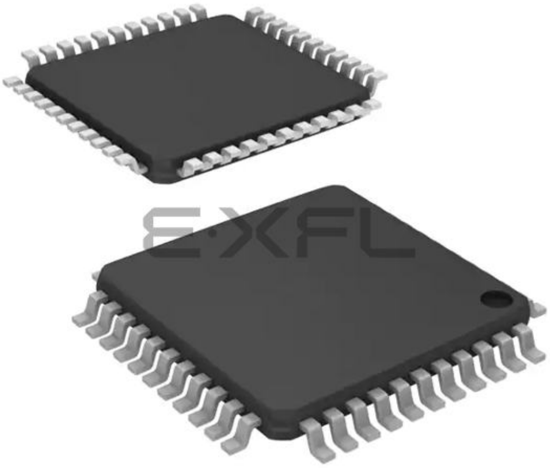


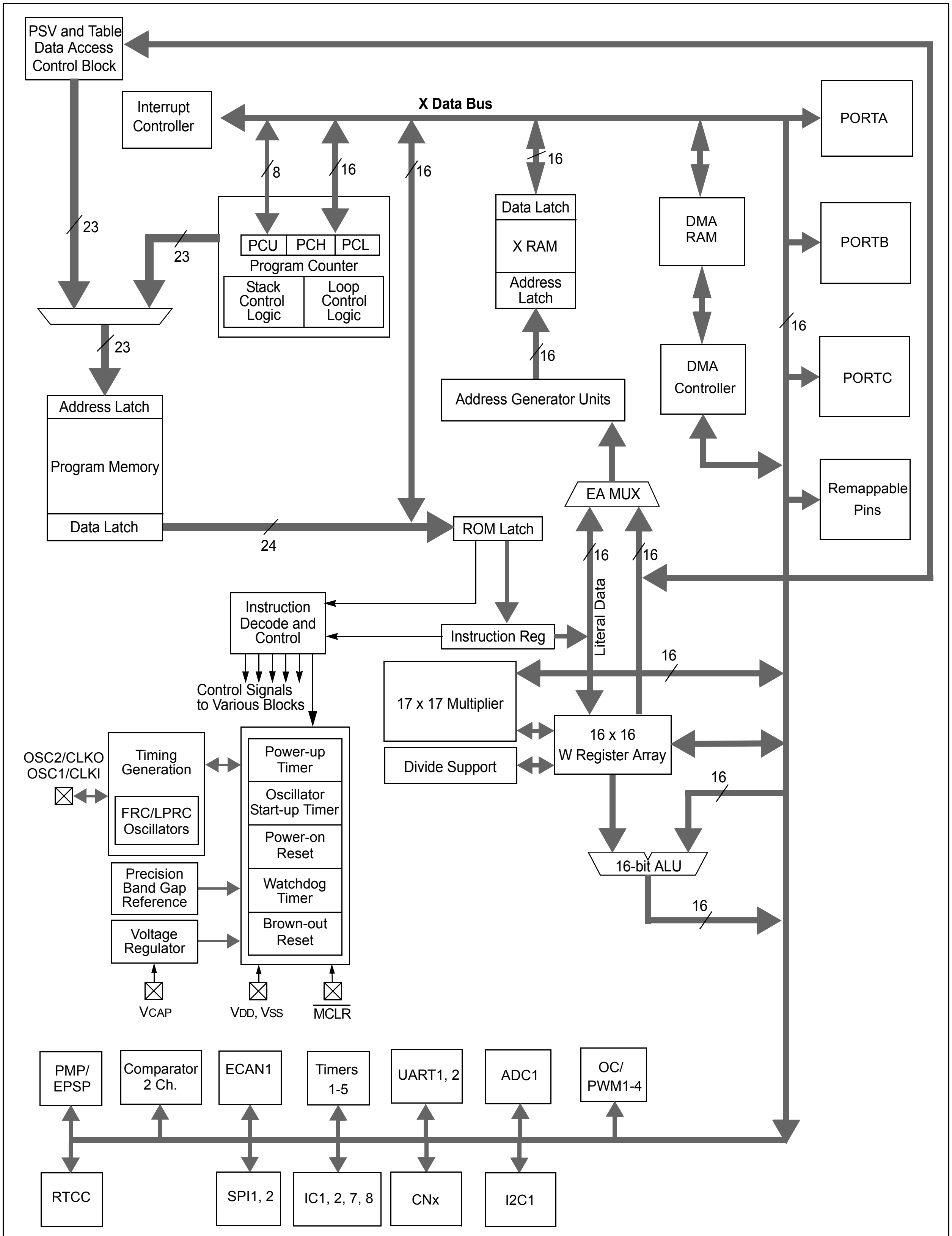
What is "Embedded - Microcontrollers"?

"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	40 MIPs
Connectivity	CANbus, I²C, IrDA, LINbus, PMP, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	35
Program Memory Size	128KB (43K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 13x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24hj128gp504-e-pt

FIGURE 1-1: PIC24HJ32GP302/304, PIC24HJ64GPX02/X04 AND PIC24HJ128GPX02/X04 BLOCK DIAGRAM



Note: Not all pins or features are implemented on all device pinout configurations. See **"Pin Diagrams"** for the specific pins and features present on each device.

2.5 ICSP Pins

The PGECx and PGEDx pins are used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes. It is recommended to keep the trace length between the ICSP connector and the ICSP pins on the device as short as possible. If the ICSP connector is expected to experience an ESD event, a series resistor is recommended, with the value in the range of a few tens of Ohms, not to exceed 100 Ohms.

Pull-up resistors, series diodes, and capacitors on the PGECx and PGEDx pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits and pin input voltage high (V_{IH}) and input low (V_{IL}) requirements.

Ensure that the "Communication Channel Select" (i.e., PGECx/PGEDx pins) programmed into the device matches the physical connections for the ICSP to MPLAB® ICD 3 or MPLAB REAL ICE™.

For more information on ICD 3 and REAL ICE connection requirements, refer to the following documents that are available on the Microchip web site.

- *"Using MPLAB® ICD 3"* (poster) DS51765
- *"MPLAB® ICD 3 Design Advisory"* DS51764
- *"MPLAB® REAL ICE™ In-Circuit Emulator User's Guide"* DS51616
- *"Using MPLAB® REAL ICE™"* (poster) DS51749

2.6 External Oscillator Pins

Many MCUs have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator (refer to **Section 9.0 "Oscillator Configuration"** for details).

The oscillator circuit should be placed on the same side of the board as the device. Also, place the oscillator circuit close to the respective oscillator pins, not exceeding one-half inch (12 mm) distance between them. The load capacitors should be placed next to the oscillator itself, on the same side of the board. Use a grounded copper pour around the oscillator circuit to isolate them from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed. A suggested layout is shown in Figure 2-3. Recommendations for crystals and ceramic resonators are provided in Table 2-1 and Table 2-2, respectively.

FIGURE 2-3: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT

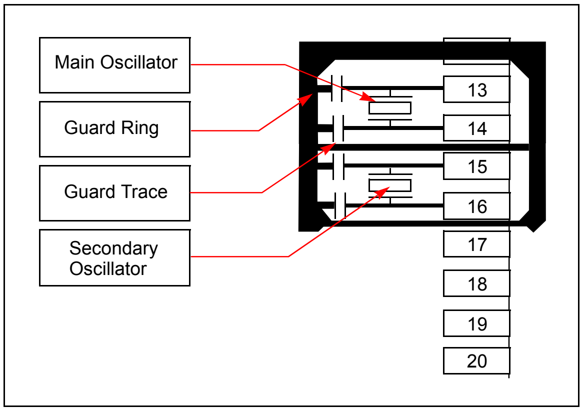


TABLE 2-1: CRYSTAL RECOMMENDATIONS

Part Number	Vendor	Freq.	Load Cap.	Package Case	Frequency Tolerance	Mounting Type	Operating Temperature
ECS-40-20-4DN	ECS Inc.	4 MHz	20 pF	HC49/US	±30 ppm	TH	-40°C to +85°C
ECS-80-18-4DN	ECS Inc.	8 MHz	18 pF	HC49/US	±30 ppm	TH	-40°C to +85°C
ECS-100-18-4-DN	ECS Inc.	10 MHz	18 pF	HC49/US	±30 ppm	TH	-40°C to +85°C
ECS-200-20-4DN	ECS Inc.	20 MHz	20 pF	HC49/US	±30 ppm	TH	-40°C to +85°C
ECS-40-20-5G3XDS-TR	ECS Inc.	4 MHz	20 pF	HC49/US	±30 ppm	SM	-40°C to +125°C
ECS-80-20-5G3XDS-TR	ECS Inc.	8 MHz	20 pF	HC49/US	±30 ppm	SM	-40°C to +125°C
ECS-100-20-5G3XDS-TR	ECS Inc.	10 MHz	20 pF	HC49/US	±30 ppm	SM	-40°C to +125°C
ECS-200-20-5G3XDS-TR	ECS Inc.	20 MHz	20 pF	HC49/US	±30 ppm	SM	-40°C to 125°C
NX3225SA 20MHZ AT-W	NDK	20 MHz	8 pF	3.2 mm x 2.5 mm	±50 ppm	SM	-40°C to 125°C

Legend: TH = Through Hole SM = Surface Mount

4.1.1 PROGRAM MEMORY ORGANIZATION

The program memory space is organized in word-addressable blocks. Although it is treated as 24 bits wide, it is more appropriate to think of each address of the program memory as a lower and upper word, with the upper byte of the upper word being unimplemented. The lower word always has an even address, while the upper word has an odd address, as shown in Figure 4-2.

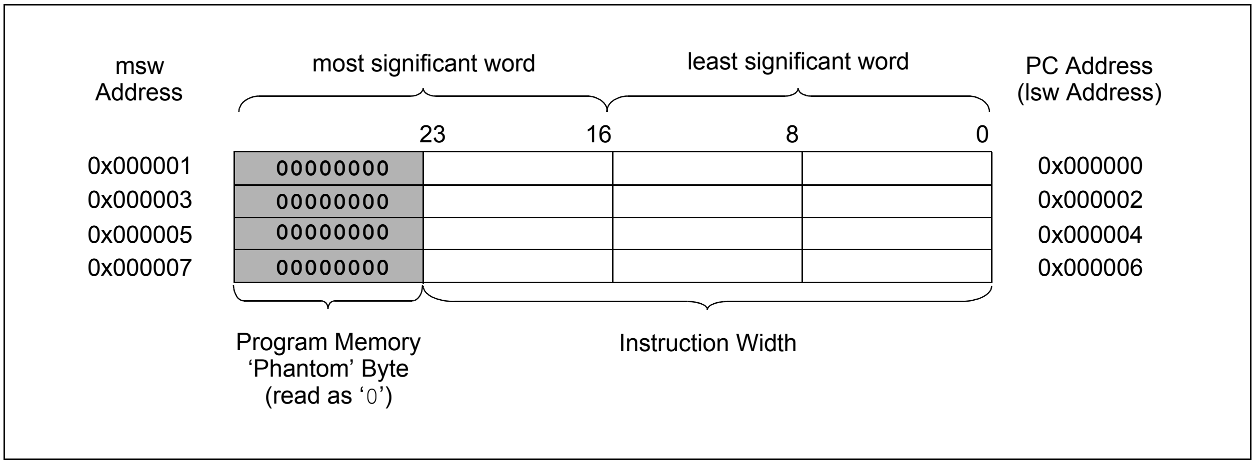
Program memory addresses are always word-aligned on the lower word, and addresses are incremented or decremented by two during code execution. This arrangement provides compatibility with data memory space addressing and makes data in the program memory space accessible.

4.1.2 INTERRUPT AND TRAP VECTORS

All PIC24HJ32GP302/304, PIC24HJ64GPX02/X04 and PIC24HJ128GPX02/X04 devices reserve the addresses between 0x00000 and 0x000200 for hard-coded program execution vectors. A hardware Reset vector is provided to redirect code execution from the default value of the PC on device Reset to the actual start of code. A `GOTO` instruction is programmed by the user application at 0x000000, with the actual address for the start of code at 0x000002.

PIC24HJ32GP302/304, PIC24HJ64GPX02/X04 and PIC24HJ128GPX02/X04 devices also have two interrupt vector tables, located from 0x000004 to 0x0000FF and 0x000100 to 0x0001FF. These vector tables allow each of the device interrupt sources to be handled by separate Interrupt Service Routines (ISRs). A more detailed discussion of the interrupt vector tables is provided in **Section 7.1 "Interrupt Vector Table"**.

FIGURE 4-2: PROGRAM MEMORY ORGANIZATION



EXAMPLE 5-2: LOADING THE WRITE BUFFERS

```
; Set up NVMCON for row programming operations
        MOV     #0x4001, W0               ;
        MOV     W0, NVMCON                ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
        MOV     #0x0000, W0               ;
        MOV     W0, TBLPAG                ; Initialize PM Page Boundary SFR
        MOV     #0x6000, W0               ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
        MOV     #LOW_WORD_0, W2           ;
        MOV     #HIGH_BYTE_0, W3          ;
        TBLWTL  W2, [W0]                  ; Write PM low word into program latch
        TBLWTH  W3, [W0++]                ; Write PM high byte into program latch
; 1st_program_word
        MOV     #LOW_WORD_1, W2           ;
        MOV     #HIGH_BYTE_1, W3          ;
        TBLWTL  W2, [W0]                  ; Write PM low word into program latch
        TBLWTH  W3, [W0++]                ; Write PM high byte into program latch
;  2nd_program_word
        MOV     #LOW_WORD_2, W2           ;
        MOV     #HIGH_BYTE_2, W3          ;
        TBLWTL  W2, [W0]                  ; Write PM low word into program latch
        TBLWTH  W3, [W0++]                ; Write PM high byte into program latch
        •
        •
        •
; 63rd_program_word
        MOV     #LOW_WORD_31, W2          ;
        MOV     #HIGH_BYTE_31, W3         ;
        TBLWTL  W2, [W0]                  ; Write PM low word into program latch
        TBLWTH  W3, [W0++]                ; Write PM high byte into program latch
```

EXAMPLE 5-3: INITIATING A PROGRAMMING SEQUENCE

```
        DISI    #5                        ; Block all interrupts with priority <7
                                          ; for next 5 instructions
        MOV     #0x55, W0
        MOV     W0, NVMKEY                ; Write the 55 key
        MOV     #0xAA, W1                 ;
        MOV     W1, NVMKEY                ; Write the AA key
        BSET    NVMCON, #WR               ; Start the erase sequence
        NOP                               ; Insert two NOPs after the
        NOP                               ; erase command is asserted
```

REGISTER 8-2: DMAxREQ: DMA CHANNEL x IRQ SELECT REGISTER

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
FORCE[1]	—	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	IRQSEL<6:0>[2]						
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 **FORCE:** Force DMA Transfer bit[1]

`1` = Force a single DMA transfer (Manual mode)
`0` = Automatic DMA transfer initiation by DMA request

bit 14-7 **Unimplemented:** Read as ‘`0`’

bit 6-0 **IRQSEL<6:0>:** DMA Peripheral IRQ Number Select bits[2]

`0000000-1111111` = DMAIRQ0-DMAIRQ127 selected to be Channel DMAREQ

Note 1: The FORCE bit cannot be cleared by the user. The FORCE bit is cleared by hardware when the forced DMA transfer is complete.

2: Refer to Table 7-1 for a complete listing of IRQ numbers for all interrupt sources.

REGISTER 8-7: DMACS0: DMA CONTROLLER STATUS REGISTER 0 (CONTINUED)

bit 3	**XWCOL3:** Channel 3 DMA RAM Write Collision Flag bit 1 = Write collision detected 0 = No write collision detected
bit 2	**XWCOL2:** Channel 2 DMA RAM Write Collision Flag bit 1 = Write collision detected 0 = No write collision detected
bit 1	**XWCOL1:** Channel 1 DMA RAM Write Collision Flag bit 1 = Write collision detected 0 = No write collision detected
bit 0	**XWCOL0:** Channel 0 DMA RAM Write Collision Flag bit 1 = Write collision detected 0 = No write collision detected

NOTES:

10.5 Power-Saving Resources

Many useful resources related to power-saving modes are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this link, contains the latest updates and additional information.

Note:	In the event you are not able to access the product page using the link above, enter this URL in your browser: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en532315

10.5.1 KEY RESOURCES

- **Section 9. "Watchdog Timer and Power-Saving Modes"** (DS70196)
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- Development Tools

11.6 Peripheral Pin Select

Peripheral pin select configuration enables peripheral set selection and placement on a wide range of I/O pins. By increasing the pinout options available on a particular device, programmers can better tailor the microcontroller to their entire application, rather than trimming the application to fit the device.

The peripheral pin select configuration feature operates over a fixed subset of digital I/O pins. Programmers can independently map the input and/or output of most digital peripherals to any one of these I/O pins. Peripheral pin select is performed in software, and generally does not require the device to be reprogrammed. Hardware safeguards are included that prevent accidental or spurious changes to the peripheral mapping, once it has been established.

11.6.1 AVAILABLE PINS

The peripheral pin select feature is used with a range of up to 26 pins. The number of available pins depends on the particular device and its pin count. Pins that support the peripheral pin select feature include the designation “RPn” in their full pin designation, where “RP” designates a remappable peripheral and “n” is the remappable pin number.

11.6.2 CONTROLLING PERIPHERAL PIN SELECT

Peripheral pin select features are controlled through two sets of special function registers: one to map peripheral inputs, and another one to map outputs. Because they are separately controlled, a particular peripheral’s input and output (if the peripheral has both) can be placed on any selectable function pin without constraint.

The association of a peripheral to a peripheral selectable pin is handled in two different ways, depending on whether an input or output is being mapped.

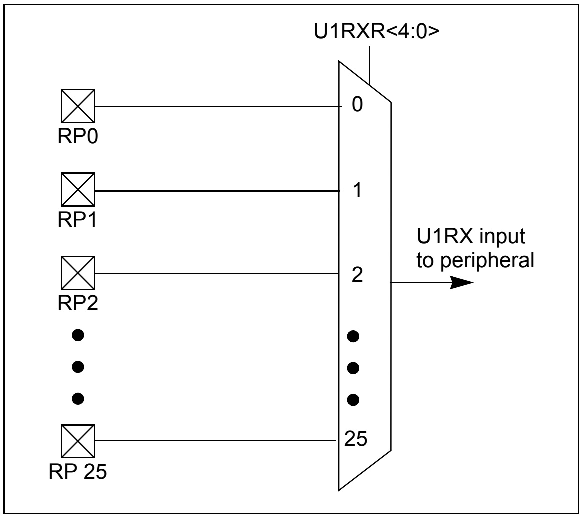
11.6.2.1 Input Mapping

The inputs of the peripheral pin select options are mapped on the basis of the peripheral. A control register associated with a peripheral dictates the pin it is mapped to. The RPINRx registers are used to configure peripheral input mapping (see Register 11-1 through Register 11-14). Each register contains sets of 5-bit fields, with each set associated with one of the remappable peripherals. Programming a given peripheral’s bit field with an appropriate 5-bit value maps the RPn pin with that value to that peripheral. For any given device, the valid range of values for any bit field corresponds to the maximum number of peripheral pin selections supported by the device.

Figure 11-2 illustrates remappable pin selection for U1RX input.

Note:	For input mapping only, the Peripheral Pin Select (PPS) functionality does not have priority over the TRISx settings. Therefore, when configuring the RPx pin for input, the corresponding bit in the TRISx register must also be configured for input (i.e., set to ‘`1`’).

FIGURE 11-2: REMAPPABLE MUX INPUT FOR U1RX



12.2 Timer1 Control Register

REGISTER 12-1: T1CON: TIMER1 CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
—	TGATE	TCKPS<1:0>		—	TSYNC	TCS	—
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 **TON:** Timer1 On bit
`1` = Starts 16-bit Timer1
`0` = Stops 16-bit Timer1

bit 14 **Unimplemented:** Read as ‘`0`’

bit 13 **TSIDL:** Stop in Idle Mode bit
`1` = Discontinue module operation when device enters Idle mode
`0` = Continue module operation in Idle mode

bit 12-7 **Unimplemented:** Read as ‘`0`’

bit 6 **TGATE:** Timer1 Gated Time Accumulation Enable bit
When TCS = `1`:
This bit is ignored.
When TCS = `0`:
`1` = Gated time accumulation enabled
`0` = Gated time accumulation disabled

bit 5-4 **TCKPS<1:0>:** Timer1 Input Clock Prescaler Select bits
`11` = 1:256
`10` = 1:64
`01` = 1:8
`00` = 1:1

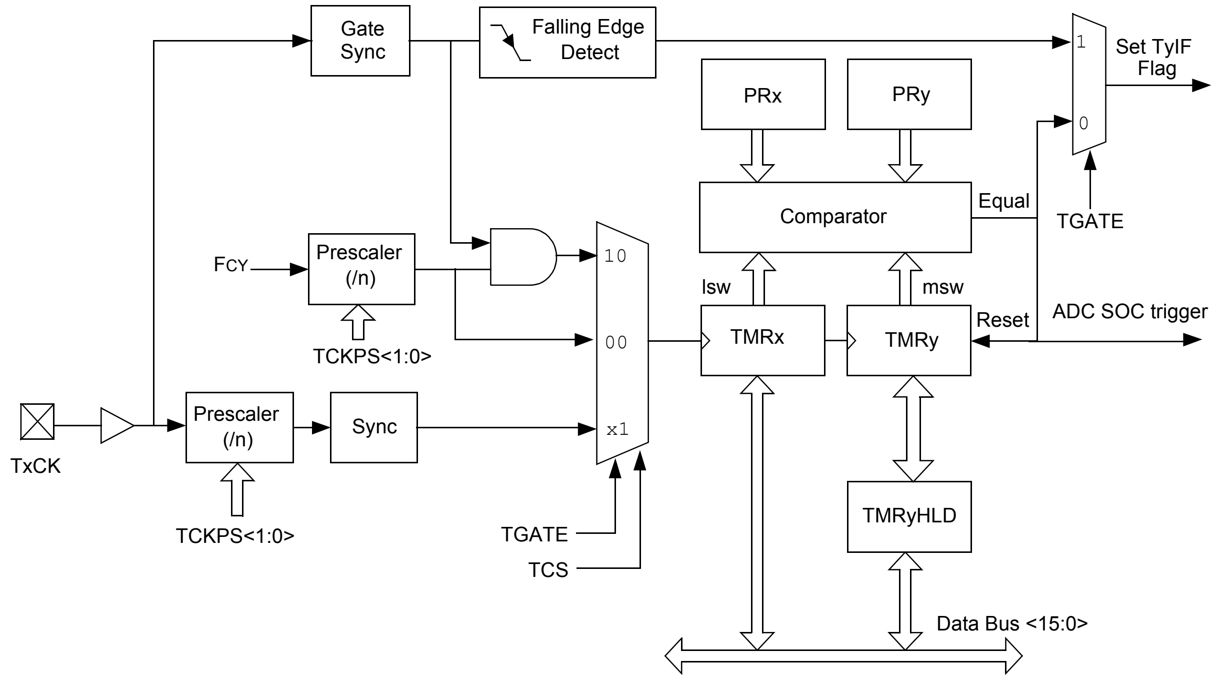
bit 3 **Unimplemented:** Read as ‘`0`’

bit 2 **TSYNC:** Timer1 External Clock Input Synchronization Select bit
When TCS = `1`:
`1` = Synchronize external clock input
`0` = Do not synchronize external clock input
When TCS = `0`:
This bit is ignored.

bit 1 **TCS:** Timer1 Clock Source Select bit
`1` = External clock from pin T1CK (on the rising edge)
`0` = Internal clock (FCY)

bit 0 **Unimplemented:** Read as ‘`0`’

FIGURE 13-3: 32-BIT TIMER BLOCK DIAGRAM



Note 1: ADC trigger is available only on TMR3:TMR2 and TMR5:TMR2 32-bit timers.

2: Timer x is a Type B Timer (x = 2 and 4).

3: Timer y is a Type C Timer (y = 3 and 5).

13.3 Timer Resources

Many useful resources related to Timers are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page, which can be accessed using this link, contains the latest updates and additional information.

Note: In the event you are not able to access the product page using the link above, enter this URL in your browser: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en532315

13.3.1 KEY RESOURCES

- **Section 11. "Timers"** (DS70205)
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All related dsPIC33F/PIC24H Family Reference Manuals Sections
- Development Tools

REGISTER 16-2: SPIxCON1: SPIx CONTROL REGISTER 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE(1)
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEN(3)	CKP	MSTEN	SPRE<2:0>(2)			PPRE<1:0>(2)	
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 **DISSCK:** Disable SCKx pin bit (SPI Master modes only)
`1` = Internal SPI clock is disabled, pin functions as I/O
`0` = Internal SPI clock is enabled

bit 11 **DISSDO:** Disable SDOx pin bit
`1` = SDOx pin is not used by module; pin functions as I/O
`0` = SDOx pin is controlled by the module

bit 10 **MODE16:** Word/Byte Communication Select bit
`1` = Communication is word-wide (16 bits)
`0` = Communication is byte-wide (8 bits)

bit 9 **SMP:** SPIx Data Input Sample Phase bit
Master mode:
`1` = Input data sampled at end of data output time
`0` = Input data sampled at middle of data output time
Slave mode:
SMP must be cleared when SPIx is used in Slave mode.

bit 8 **CKE:** SPIx Clock Edge Select bit(1)
`1` = Serial output data changes on transition from active clock state to Idle clock state (see bit 6)
`0` = Serial output data changes on transition from Idle clock state to active clock state (see bit 6)

bit 7 **SSEN:** Slave Select Enable bit (Slave mode)(3)
`1` = $\overline{\text{SSx}}$ pin used for Slave mode
`0` = $\overline{\text{SSx}}$ pin not used by module. Pin controlled by port function

bit 6 **CKP:** 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

Note 1: The CKE bit is not used in the Framed SPI modes. Program this bit to ‘`0`’ for the Framed SPI modes (FRMEN = `1`).

2: Do not set both Primary and Secondary prescalers to a value of 1:1.

3: This bit must be cleared when FRMEN = `1`.

REGISTER 16-3: SPIxCON2: SPIx CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
FRMEN	SPIFSD	FRMPOL	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	FRMDLY	—
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	**FRMEN:** Framed SPIx Support bit 1 = Framed SPIx support enabled ($\overline{\text{SSx}}$ pin used as frame sync pulse input/output) 0 = Framed SPIx support disabled
bit 14	**SPIFSD:** Frame Sync Pulse Direction Control bit 1 = Frame sync pulse input (slave) 0 = Frame sync pulse output (master)
bit 13	**FRMPOL:** Frame Sync Pulse Polarity bit 1 = Frame sync pulse is active-high 0 = Frame sync pulse is active-low
bit 12-2	**Unimplemented:** Read as ‘0’
bit 1	**FRMDLY:** Frame Sync Pulse Edge Select bit 1 = Frame sync pulse coincides with first bit clock 0 = Frame sync pulse precedes first bit clock
bit 0	**Unimplemented:** This bit must not be set to ‘1’ by the user application

REGISTER 19-4: CiFCTRL: ECAN™ FIFO CONTROL REGISTER

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
DMABS<2:0>			—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	FSA<4:0>				
bit 7							bit 0

Legend:	C = Writeable bit, but only ‘0’ can be written to clear the 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 15-13 **DMABS<2:0>:** DMA Buffer Size bits

`111` = Reserved
`110` = 32 buffers in DMA RAM
`101` = 24 buffers in DMA RAM
`100` = 16 buffers in DMA RAM
`011` = 12 buffers in DMA RAM
`010` = 8 buffers in DMA RAM
`001` = 6 buffers in DMA RAM
`000` = 4 buffers in DMA RAM

bit 12-5 **Unimplemented:** Read as ‘`0`’

bit 4-0 **FSA<4:0>:** FIFO Area Starts with Buffer bits

`11111` = Read buffer RB31
`11110` = Read buffer RB30
•
•
•
`00001` = TX/RX buffer TRB1
`00000` = TX/RX buffer TRB0

REGISTER 19-19: CiFMSKSEL2: ECAN™ FILTER 15-8 MASK SELECTION REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F15MSK<1:0>		F14MSK<1:0>		F13MSK<1:0>		F12MSK<1:0>	
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F11MSK<1:0>		F10MSK<1:0>		F9MSK<1:0>		F8MSK<1:0>	
bit 7							bit 0

Legend:	C = Writeable bit, but only ‘0’ can be written to clear the 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	**F15MSK<1:0>:** Mask Source for Filter 15 bit `11` = No mask `10` = Acceptance Mask 2 registers contain mask `01` = Acceptance Mask 1 registers contain mask `00` = Acceptance Mask 0 registers contain mask
bit 13-12	**F14MSK<1:0>:** Mask Source for Filter 14 bit (same values as bit 15-14)
bit 11-10	**F13MSK<1:0>:** Mask Source for Filter 13 bit (same values as bit 15-14)
bit 9-8	**F12MSK<1:0>:** Mask Source for Filter 12 bit (same values as bit 15-14)
bit 7-6	**F11MSK<1:0>:** Mask Source for Filter 11 bit (same values as bit 15-14)
bit 5-4	**F10MSK<1:0>:** Mask Source for Filter 10 bit (same values as bit 15-14)
bit 3-2	**F9MSK<1:0>:** Mask Source for Filter 9 bit (same values as bit 15-14)
bit 1-0	**F8MSK<1:0>:** Mask Source for Filter 8 bit (same values as bit 15-14)

REGISTER 20-1: AD1CON1: ADC1 CONTROL REGISTER 1 (CONTINUED)

bit 3	**SIMSAM:** Simultaneous Sample Select bit (only applicable when CHPS<1:0> = 01 or 1x) **When AD12B = 1, SIMSAM is: U-0, Unimplemented, Read as '0'** 1 = Samples CH0, CH1, CH2, CH3 simultaneously (when CHPS<1:0> = 1x); or Samples CH0 and CH1 simultaneously (when CHPS<1:0> = 01) 0 = Samples multiple channels individually in sequence
bit 2	**ASAM:** ADC Sample Auto-Start bit 1 = Sampling begins immediately after last conversion. SAMP bit is auto-set 0 = Sampling begins when SAMP bit is set
bit 1	**SAMP:** ADC Sample Enable bit 1 = ADC sample/hold amplifiers are sampling 0 = ADC sample/hold amplifiers are holding If ASAM = 0, software can write '1' to begin sampling. Automatically set by hardware if ASAM = 1. If SSRC = 000, software can write '0' to end sampling and start conversion. If SSRC ≠ 000, automatically cleared by hardware to end sampling and start conversion.
bit 0	**DONE:** ADC Conversion Status bit 1 = ADC conversion cycle is completed 0 = ADC conversion not started or in progress Automatically set by hardware when ADC conversion is complete. Software can write '0' to clear DONE status (software not allowed to write '1'). Clearing this bit does NOT affect any operation in progress. Automatically cleared by hardware at start of a new conversion.

REGISTER 20-7: AD1CSSL: ADC1 INPUT SCAN SELECT REGISTER LOW[1,2]

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	CSS12	CSS11	CSS10	CSS9	CSS8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0
bit 7							bit 0

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

bit 15-13	**Unimplemented:** Read as ‘`0`’
bit 12-0	**CSS<12:0>:** ADC Input Scan Selection bits `1` = Select ANx for input scan `0` = Skip ANx for input scan

Note 1: On devices without 13 analog inputs, all AD1CSSL bits can be selected by user application. However, inputs selected for scan without a corresponding input on device converts VREF-.

2: CSSx = ANx, where x = 0 through 12.

REGISTER 20-8: AD1PCFGL: ADC1 PORT CONFIGURATION REGISTER LOW[1,2,3]

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

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

bit 15-13	**Unimplemented:** Read as ‘`0`’
bit 12-0	**PCFG<12:0>:** ADC Port Configuration Control bits `1` = Port pin in Digital mode, port read input enabled, ADC input multiplexer connected to AVSS `0` = Port pin in Analog mode, port read input disabled, ADC samples pin voltage

Note 1: On devices without 13 analog inputs, all PCFG bits are R/W by user. However, PCFG bits are ignored on ports without a corresponding input on device.

2: PCFGx = ANx, where x = 0 through 12.

3: PCFGx bits have no effect if ADC module is disabled by setting ADxMD bit in the PMDx register. In this case, all port pins multiplexed with ANx will be in Digital mode.

TABLE 28-6: DC CHARACTERISTICS: IDLE CURRENT (IIDLE)

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Parameter No.(3)	**Typical(2)**	**Max**	**Units**	**Conditions**		
Idle Current (IIDLE): Core OFF Clock ON Base Current(1)						
DC40d	8	10	mA	-40°C	3.3V	10 MIPS
DC40a	8	10	mA	+25°C		
DC40b	9	10	mA	+85°C		
DC40c	10	13	mA	+125°C		
DC41d	13	15	mA	-40°C	3.3V	16 MIPS
DC41a	13	15	mA	+25°C		
DC41b	13	16	mA	+85°C		
DC41c	13	19	mA	+125°C		
DC42d	15	18	mA	-40°C	3.3V	20 MIPS
DC42a	16	18	mA	+25°C		
DC42b	16	19	mA	+85°C		
DC42c	17	22	mA	+125°C		
DC43a	23	27	mA	+25°C	3.3V	30 MIPS
DC43d	23	26	mA	-40°C		
DC43b	24	28	mA	+85°C		
DC43c	25	31	mA	+125°C		
DC44d	31	42	mA	-40°C	3.3V	40 MIPS
DC44a	31	36	mA	+25°C		
DC44b	32	39	mA	+85°C		
DC44c	34	43	mA	+125°C		

Note 1: Base IIDLE current is measured as follows:

- CPU core is off (i.e., Idle mode), oscillator is configured in EC mode and external clock active, OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)
- CLKO is configured as an I/O input pin in the Configuration word
- External Secondary Oscillator disabled (i.e., SOSCO and SOSCI pins configured as digital I/O inputs)
- All I/O pins are configured as inputs and pulled to VSS
- $\overline{\text{MCLR}}$ = VDD, WDT and FSCM are disabled
- No peripheral modules are operating; however, every peripheral is being clocked (defined PMDx bits are set to zero)
- JTAG is disabled

2: Data in “Typ” column is at 3.3V, +25ºC unless otherwise stated.

3: These parameters are characterized but not tested in manufacturing.

TABLE 28-34: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 1, SMP = 0) TIMING REQUIREMENTS

AC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	**Symbol**	**Characteristic[1]**	**Min**	**Typ[2]**	**Max**	**Units**	**Conditions**
SP70	TscP	Maximum SCK Input Frequency	—	—	15	MHz	See **Note 3**
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See parameter DO32 and **Note 4**
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See parameter DO31 and **Note 4**
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	—
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{\text{SSx}}$ ↓to SCKx ↑ or SCKx Input	120	—	—	ns	—
SP51	TssH2doZ	$\overline{\text{SSx}}$ ↑ to SDOx Output High-Impedance[4]	10	—	50	ns	—
SP52	TscH2ssH TscL2ssH	$\overline{\text{SSx}}$ after SCKx Edge	1.5 TCY + 40	—	—	ns	See **Note 4**

Note 1: These parameters are characterized, but are not tested in manufacturing.

2: Data in “Typ” column is at 3.3V, 25°C unless otherwise stated.

3: The minimum clock period for SCKx is 66.7 ns. Therefore, the SCK clock generated by the Master must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.