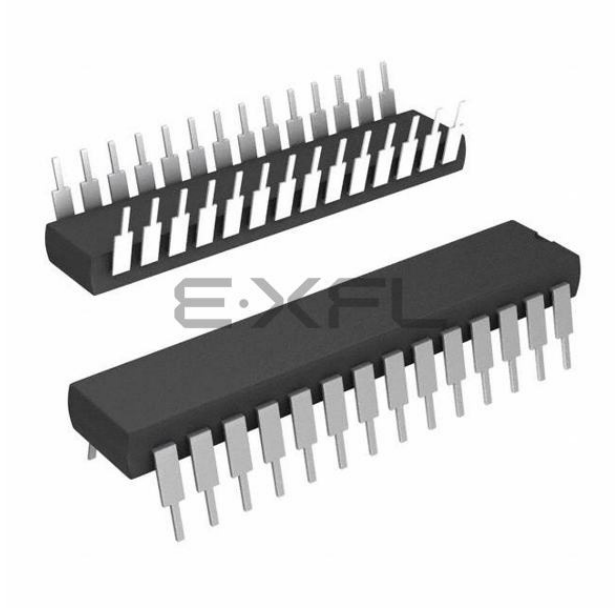


Welcome to E-XFL.COM

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	8-Bit
Speed	64MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, HLVD, POR, PWM, WDT
Number of I/O	25
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 24x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf26k42-e-sp

TABLE 2: 40/44-PIN ALLOCATION TABLE FOR PIC18(L)F4XK42

I/O	40-Pin PDIP	44-Pin TQFP	40-Pin UQFN	44-Pin QFN	ADC	Voltage Reference	DAC	Comparators	Zero Cross Detect	I^2C	SPI	UART	DSM	Timers/SMT	CCP and PWM	CWG	CLC	NCO	Clock Reference (CLKR)	Interrupt-on-Change	Basic
RC3	18	37	33	37	ANC3	—	—	—	—	SCL1(3,4)	SCK1(1)	—	—	T2IN(1)	—	—	—	—	—	IOCC3	—
RC4	23	42	38	42	ANC4	—	—	—	—	SDA1(3,4)	SDI1(1)	—	—	—	—	—	—	—	—	IOCC4	—
RC5	24	43	39	43	ANC5	—	—	—	—	—	—	—	—	T4IN(1)	—	—	—	—	—	IOCC5	—
RC6	25	44	40	44	ANC6	—	—	—	—	—	—	CTS1(1)	—	—	—	—	—	—	—	IOCC6	—
RC7	26	1	1	1	ANC7	—	—	—	—	—	—	RX1(1)	—	—	—	—	—	—	—	IOCC7	—
RD0	19	38	34	38	AND0	—	—	—	—	—(4)	—	—	—	—	—	—	—	—	—	—	—
RD1	20	39	35	39	AND1	—	—	—	—	—(4)	—	—	—	—	—	—	—	—	—	—	—
RD2	21	40	36	40	AND2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RD3	22	41	37	41	AND3	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RD4	27	2	2	2	AND4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RD5	28	3	3	3	AND5	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RD6	29	4	4	4	AND6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RD7	30	5	5	5	AND7	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RE0	8	25	23	25	ANE0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RE1	9	26	24	26	ANE1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RE2	10	27	25	27	ANE2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RE3	1	18	16	18	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IOCE3	$\overline{MCLR}$ V_{PP}
V_{DD}	11, 32	7, 28	7, 26	7, 28	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
V_{SS}	12, 31	6, 29	6, 27	6, 30	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

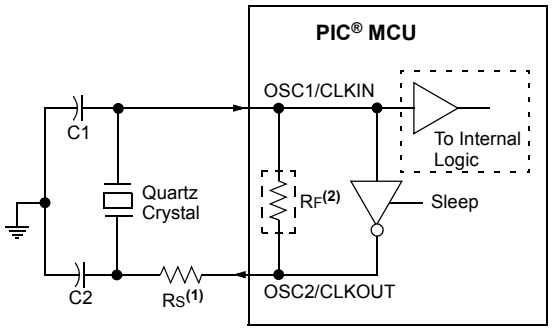
Note 1: This is a PPS remappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins.

2: All output signals shown in this row are PPS remappable.

3: This is a bidirectional signal. For normal module operation, the firmware should map this signal to the same pin in both the PPS input and PPS output registers.

4: These pins can be configured for I^2C and SMB™ 3.0/2.0 logic levels; The SCLx/SDAx signals may be assigned to any of the RB1/RB2/RC3/RC4/RD0/RD1 pins. PPS assignments to the other pins (e.g., RA5) will operate, but input logic levels will be standard TTL/ST as selected by the INLVL register, instead of the I^2C specific or SMBus input buffer thresholds.

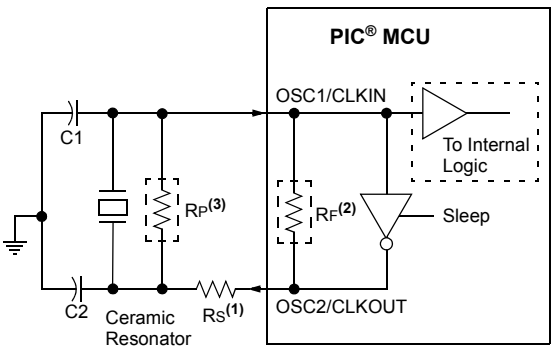
FIGURE 7-3: QUARTZ CRYSTAL OPERATION (LP, XT OR HS MODE)



Note 1: A series resistor (RS) may be required for quartz crystals with low drive level.

2: The value of RF varies with the Oscillator mode selected (typically between 2 MΩ to 10 MΩ).

FIGURE 7-4: CERAMIC RESONATOR OPERATION (XT OR HS MODE)



Note 1: A series resistor (RS) may be required for ceramic resonators with low drive level.

2: The value of RF varies with the Oscillator mode selected (typically between 2 MΩ to 10 MΩ).

3: An additional parallel feedback resistor (RP) may be required for proper ceramic resonator operation.

7.2.1.3 Oscillator Start-up Timer (OST)

If the oscillator module is configured for LP, XT or HS modes, the Oscillator Start-up Timer (OST) counts 1024 oscillations from OSC1. This occurs following a Power-on Reset (POR), or a wake-up from Sleep. The OST ensures that the oscillator circuit, using a quartz crystal resonator or ceramic resonator, has started and is providing a stable system clock to the oscillator module.

7.2.1.4 4x PLL

The oscillator module contains a 4x PLL that can be used with the external clock sources to provide a system clock source. The input frequency for the PLL must fall within specifications. See the PLL Clock Timing Specifications in Table 44-11.

The PLL can be enabled for use by one of two methods:

1. Program the RSTOSC bits in the Configuration Word 1 to 010 (enable EXTOSC with 4x PLL).
2. Write the NOSC bits in the OSCCON1 register to 010 (enable EXTOSC with 4x PLL).

REGISTER 9-21: PIE7: PERIPHERAL INTERRUPT ENABLE REGISTER 7

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0
—	—	INT2IE	CLC2IE	CWG2IE	—	CCP2IE	TMR4IE
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

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

bit 5 **INT2IE:** External Interrupt 2 Enable bit
1 = Enabled
0 = Disabled

bit 4 **CLC2IE:** CLC2 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 3 **CWG2IE:** CWG2 Interrupt Enable bit
1 = Enabled
0 = Disabled

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

bit 1 **CCP2IE:** CCP2 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 0 **TMR4IE:** TMR4 Interrupt Enable bit
1 = Enabled
0 = Disabled

REGISTER 9-28: IPR3: PERIPHERAL INTERRUPT PRIORITY REGISTER 3

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
TMR0IP	U1IP	U1EIP	U1TXIP	U1RXIP	I2C1EIP	I2C1IP	I2C1TXIP
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **TMR0IP:** TMR0 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 6 **U1IP:** UART1 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 5 **U1EIP:** UART1 Framing Error Interrupt Priority bit
1 = High priority
0 = Low priority

bit 4 **U1TXIP:** UART1 Transmit Interrupt Priority bit
1 = High priority
0 = Low priority

bit 3 **U1RXIP:** UART1 Receive Interrupt Priority bit
1 = High priority
0 = Low priority

bit 2 **I2C1EIP:** I^2C1 Error Interrupt Priority bit
1 = High priority
0 = Low priority

bit 1 **I2C1IP:** I^2C1 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 0 **I2C1TXIP:** I^2C1 Transmit Interrupt Priority bit
1 = High priority
0 = Low priority

REGISTER 11-3: WDTPSL: WWDT PRESCALE SELECT LOW BYTE REGISTER (READ-ONLY)

R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
PSCNT<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **PSCNT<7:0>:** Prescale Select Low Byte bits(1)

Note 1: The 18-bit WDT prescale value, PSCNT<17:0> includes the WDTPSL, WDTPSH and the lower bits of the WDTTMR registers. PSCNT<17:0> is intended for debug operations and should not be read during normal operation.

REGISTER 11-4: WDTPSH: WWDT PRESCALE SELECT HIGH BYTE REGISTER (READ-ONLY)

R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
PSCNT<15:8>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

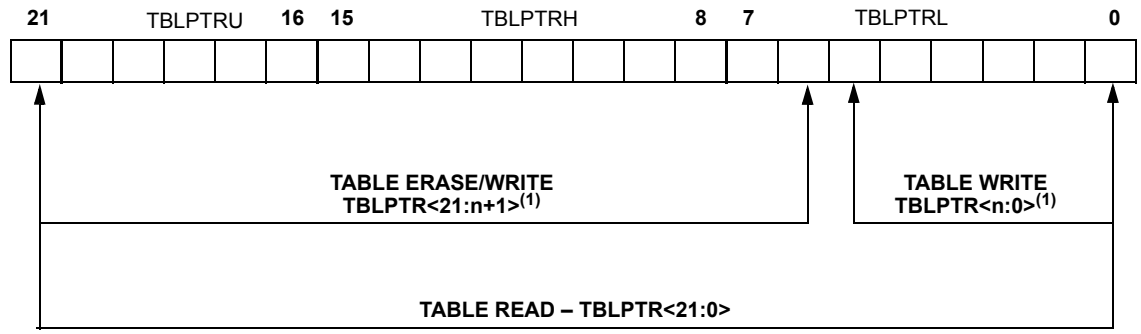
bit 7-0 **PSCNT<15:8>:** Prescale Select High Byte bits(1)

Note 1: The 18-bit WDT prescale value, PSCNT<17:0> includes the WDTPSL, WDTPSH and the lower bits of the WDTTMR registers. PSCNT<17:0> is intended for debug operations and should not be read during normal operation.

TABLE 13-2: TABLE POINTER OPERATIONS WITH TBLRD AND TBLWT INSTRUCTIONS

Example	Operation on Table Pointer
TBLRD* TBLWT*	TBLPTR is not modified
TBLRD*+ TBLWT*+	TBLPTR is incremented after the read/write
TBLRD*- TBLWT*-	TBLPTR is decremented after the read/write
TBLRD+* TBLWT+*	TBLPTR is incremented before the read/write

FIGURE 13-3: TABLE POINTER BOUNDARIES BASED ON OPERATION



Note 1: Refer to Table 5-4 for the row size values.

15.10 Reset

The DMA registers are set to the default state on any Reset. The registers are also reset to the default state when the enable bit is cleared (DMA1CON1bits.EN=0).

15.11 Power Saving Mode Operation

The DMA utilizes system clocks and it is treated as a peripheral when it comes to power saving operations. Like other peripherals, the DMA also uses Peripheral Module Disable bits to further tailor its operation in low-power states.

15.11.1 SLEEP MODE

When the device enters Sleep mode, the system clock to the module is shut down, therefore no DMA operation is supported in Sleep. Once the system clock is disabled, the requisite read and write clocks are also disabled, without which the DMA cannot perform any of its tasks.

Any transfers that may be in progress are resumed on exiting from Sleep mode. Register contents are not affected by the device entering or leaving Sleep mode. It is recommended that DMA transactions be allowed to finish before entering Sleep mode.

15.11.2 IDLE MODE

In IDLE mode, all of the system clocks (including the read and write clocks) are still operating but the CPU is not using them to save power.

Therefore, every instruction cycle is available to the system arbiter and if the bubble is granted to the DMA, it may be utilized to move data.

15.11.3 DOZE MODE

Similar to the Idle mode, the CPU does not utilize all of the available instruction cycles slots that are available to it in order to save power. It only executes instructions based on its settings from the Doze settings.

Therefore, every instruction not used by the CPU is available for system arbitration and may be utilized by the DMA if granted by the arbiter.

15.11.4 PERIPHERAL MODULE DISABLE

The Peripheral Module Disable (PMD) registers provide a method to disable DMA by gating all clock sources supplied to it. The respective DMAxMD bit needs to be set in order to disable the DMA.

15.12 DMA Register Interfaces

The DMA can transfer data to any GPR or SFR location. For better user accessibility, some of the more commonly used SFR spaces have their Mirror registers placed in Bank 64 (0x4000-0x40FF). These Mirror registers can be only accessed through the DMA Source and Destination Address registers. Refer to Table 4-3 for details about Bank 64 Registers.

16.0 I/O PORTS

The PIC18(L)F26/27/45/46/47/55/56/57K42 devices have six I/O ports, allocated as shown in Table 16-1.

TABLE 16-1: PORT ALLOCATION TABLE FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES

Device	PORTA	PORTB	PORTC	PORTD	PORTE	PORTF
PIC18(L)F26K42	•	•	•		•(1)	
PIC18(L)F27K42	•	•	•		•(1)	
PIC18(L)F45K42	•	•	•	•	•(2)	
PIC18(L)F46K42	•	•	•	•	•(2)	
PIC18(L)F47K42	•	•	•	•	•(2)	
PIC18(L)F55K42	•	•	•	•	•(2)	•
PIC18(L)F56K42	•	•	•	•	•(2)	•
PIC18(L)F57K42	•	•	•	•	•(2)	•

Note 1: Pin RE3 only.
2: Pins RE0, RE1, RE2 and RE3 only.

Each port has ten registers to control the operation. These registers are:

- PORTx registers (reads the levels on the pins of the device)
- LATx registers (output latch)
- TRISx registers (data direction)
- ANSELx registers (analog select)
- WPUx registers (weak pull-up)
- INLVLx (input level control)
- SLRCONx registers (slew rate control)
- ODCONx registers (open-drain control)

Most port pins share functions with device peripherals, both analog and digital. In general, when a peripheral is enabled on a port pin, that pin cannot be used as a general purpose output; however, the pin can still be read.

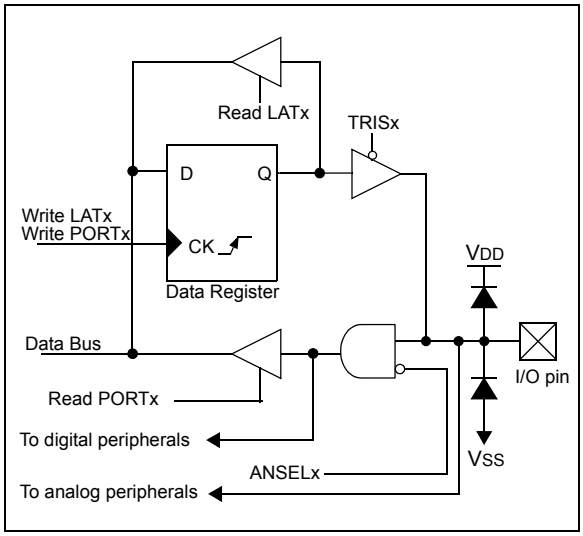
The Data Latch (LATx registers) is useful for read-modify-write operations on the value that the I/O pins are driving.

A write operation to the LATx register has the same effect as a write to the corresponding PORTx register. A read of the LATx register reads of the values held in the I/O PORT latches, while a read of the PORTx register reads the actual I/O pin value.

Ports that support analog inputs have an associated ANSELx register. When an ANSELx bit is set, the digital input buffer associated with that bit is disabled.

Disabling the input buffer prevents analog signal levels on the pin between a logic high and low from causing excessive current in the logic input circuitry. A simplified model of a generic I/O port, without the interfaces to other peripherals, is shown in Figure 16-1.

FIGURE 16-1: GENERIC I/O PORT OPERATION



16.1 I/O Priorities

Each pin defaults to the PORT data latch after Reset. Other functions are selected with the peripheral pin select logic. See **Section 17.0 "Peripheral Pin Select (PPS) Module"** for more information.

Analog input functions, such as ADC and comparator inputs, are not shown in the peripheral pin select lists. These inputs are active when the I/O pin is set for Analog mode using the ANSELx register. Digital output functions may continue to control the pin when it is in Analog mode.

Analog outputs, when enabled, take priority over digital outputs and force the digital output driver into a high-impedance state.

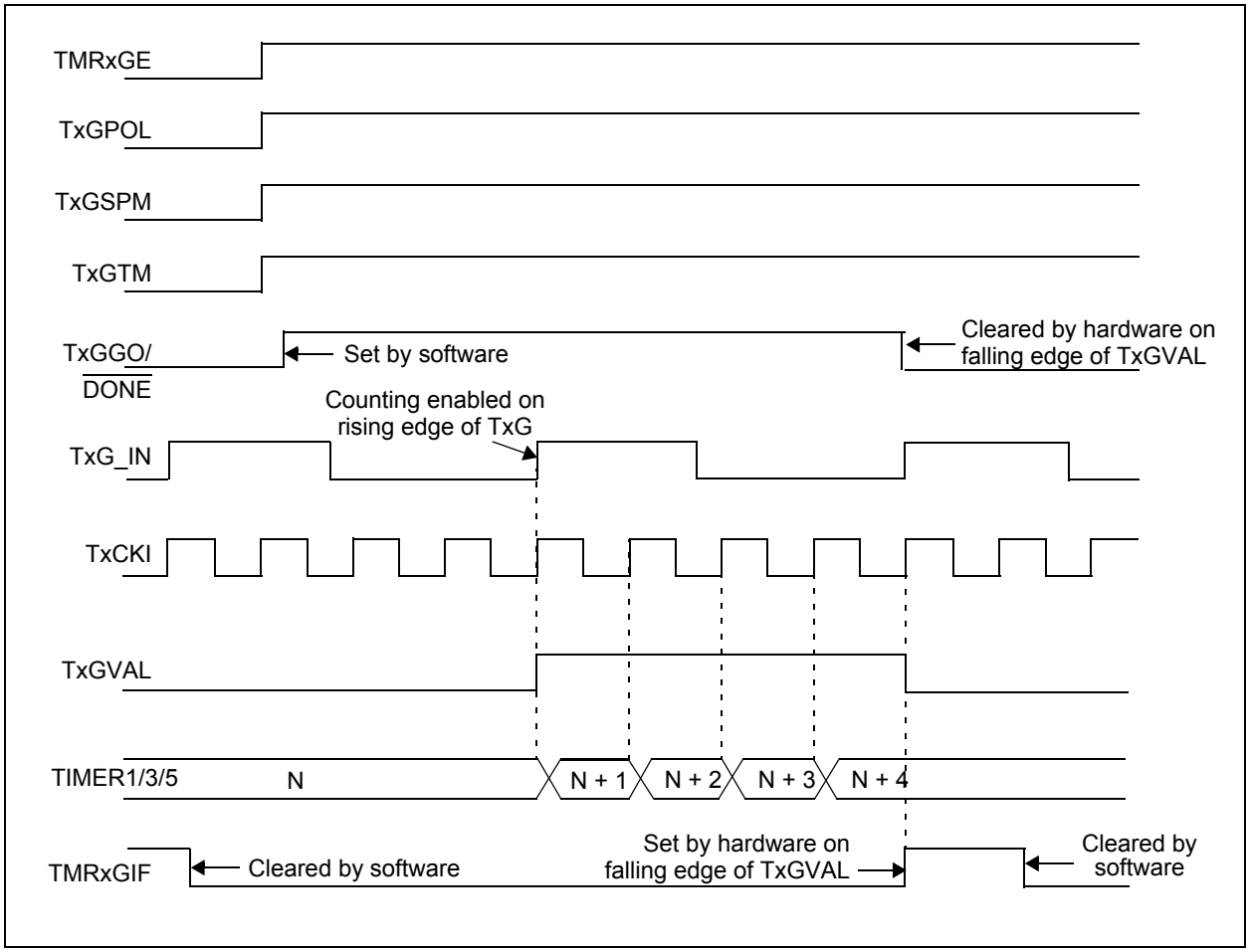
The pin function priorities are as follows:

1. Configuration bits
2. Analog outputs (disable the input buffers)
3. Analog inputs
4. Port inputs and outputs from PPS

16.2 PORTx Registers

In this section, the generic names such as PORTx, LATx, TRISx, etc. can be associated with PORTA, PORTB, and PORTC. The functionality of PORTE is different compared to other ports and is explained in a separate section.

FIGURE 21-7: TIMER1/3/5 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE



21.11 Peripheral Module Disable

When a peripheral module is not used or inactive, the module can be disabled by setting the Module Disable bit in the PMD registers. This will reduce power consumption to an absolute minimum. Setting the PMD bits holds the module in Reset and disconnects the module's clock source. The Module Disable bits for Timer1 (TMR1MD), Timer3 (TMR3MD) and Timer5 (TMR5MD) are in the respective PMD registers. See **Section 19.0 "Peripheral Module Disable (PMD)"** for more information.

FIGURE 23-1: CAPTURE MODE OPERATION BLOCK DIAGRAM

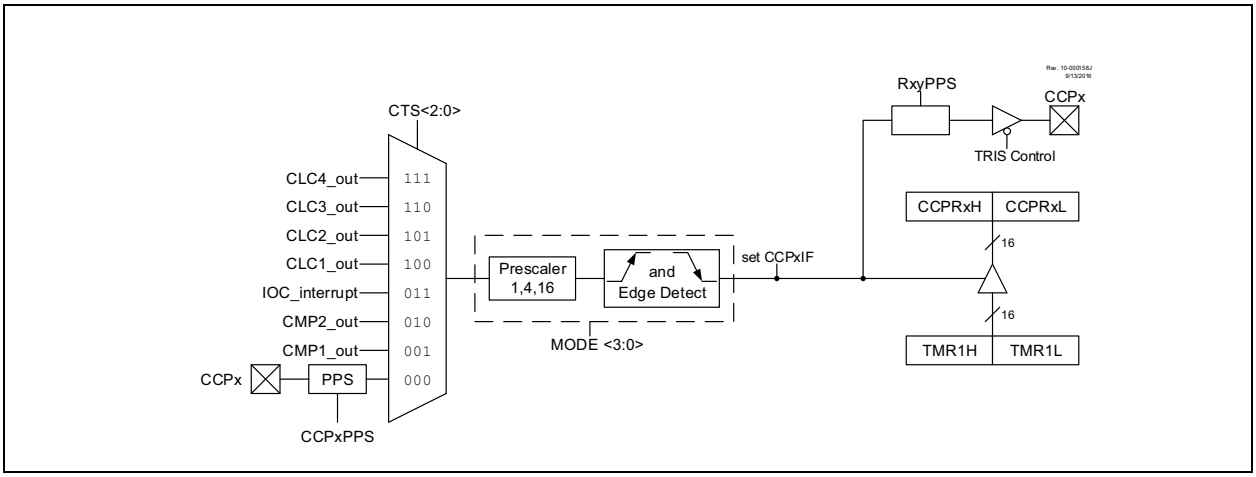


TABLE 27-1: CLCx DATA INPUT SELECTION

DyS<5:0> Value	CLCx Input Source
111111 [63]	Reserved
.	
.	
.	
110100 [52]	Reserved
110011 [51]	CWG3B_out
110010 [50]	CWG3A_out
110001 [49]	CWG2B_out
110000 [48]	CWG2A_out
101111 [47]	CWG1B_out
101110 [46]	CWG1A_out
101101 [45]	SS1
101100 [44]	SCK1
101011 [43]	SDO1
101010 [42]	Reserved
101001 [41]	UART2_tx_out
101000 [40]	UART1_tx_out
100111 [39]	CLC4_out
100110 [38]	CLC3_out
100101 [37]	CLC2_out
100100 [36]	CLC1_out
100011 [35]	DSM1_out
100010 [34]	IOC_flag
100001 [33]	ZCD_out
100000 [32]	CMP2_out
011111 [31]	CMP1_out
011110 [30]	NCO1_out
011101 [29]	Reserved
011100 [28]	Reserved
011011 [27]	PWM8_out
011010 [26]	PWM7_out
011001 [25]	PWM6_out
011000 [24]	PWM5_out
010111 [23]	CCP4_out
010110 [22]	CCP3_out
010101 [21]	CCP2_out
010100 [20]	CCP1 _out
010011 [19]	SMT1_out
010010 [18]	TMR6_out
010001 [17]	TMR5 _overflow
010000 [16]	TMR4 _out
001111 [15]	TMR3 _overflow

TABLE 27-1: CLCx DATA INPUT SELECTION (CONTINUED)

DyS<5:0> Value	CLCx Input Source
001110 [14]	TMR2 _out
001101 [13]	TMR1 _overflow
001100 [12]	TMR0 _overflow
001011 [11]	CLKR _out
001010 [10]	ADCRC
001001 [9]	SOSC
001000 [8]	MFINTOSC (32 kHz)
000111 [7]	MFINTOSC (500 kHz)
000110 [6]	LFINTOSC
000101 [5]	HFINTOSC
000100 [4]	FOSC
000011 [3]	CLCIN3PPS
000010 [2]	CLCIN2PPS
000001 [1]	CLCIN1PPS
000000 [0]	CLCIN0PPS

28.1 NCO Operation

The NCO operates by repeatedly adding a fixed value to an accumulator. Additions occur at the input clock rate. The accumulator will overflow with a carry periodically, which is the raw NCO output (NCO_overflow). This effectively reduces the input clock by the ratio of the addition value to the maximum accumulator value. See Equation 28-1.

The NCO output can be further modified by stretching the pulse or toggling a flip-flop. The modified NCO output is then distributed internally to other peripherals and can be optionally output to a pin. The accumulator overflow also generates an interrupt (NCO_overflow).

The NCO period changes in discrete steps to create an average frequency. This output depends on the ability of the receiving circuit (i.e., CWG or external resonant converter circuitry) to average the NCO output to reduce uncertainty.

EQUATION 28-1: NCO OVERFLOW FREQUENCY

$$F_{OVERFLOW} = \frac{NCO\ Clock\ Frequency \times Increment\ Value}{2^{20}}$$

28.1.1 NCO CLOCK SOURCES

Clock sources available to the NCO include:

- FOSC
- HFINTOSC
- LFINTOSC
- MFINTOSC/4 (32 kHz)
- MFINTOSC (500 kHz)
- CLC1/2/3/4_out
- CLKREF
- SOSC

The NCO clock source is selected by configuring the N1CKS<2:0> bits in the NCO1CLK register.

28.1.2 ACCUMULATOR

The accumulator is a 20-bit register. Read and write access to the accumulator is available through three registers:

- NCO1ACCL
- NCO1ACCH
- NCO1ACCU

28.1.3 ADDER

The NCO Adder is a full adder, which operates independently from the source clock. The addition of the previous result and the increment value replaces the accumulator value on the rising edge of each input clock.

28.1.4 INCREMENT REGISTERS

The increment value is stored in three registers making up a 20-bit incrementer. In order of LSB to MSB they are:

- NCO1INCL
- NCO1INCH
- NCO1INCU

When the NCO module is enabled, the NCO1INCU and NCO1INCH registers should be written first, then the NCO1INCL register. Writing to the NCO1INCL register initiates the increment buffer registers to be loaded simultaneously on the second rising edge of the NCO_clk signal.

The registers are readable and writable. The increment registers are double-buffered to allow value changes to be made without first disabling the NCO module.

When the NCO module is disabled, the increment buffers are loaded immediately after a write to the increment registers.

Note: The increment buffer registers are not user-accessible.

REGISTER 31-5: UxERRIE: UART ERROR INTERRUPT ENABLE REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TXMTIE	PERIE	ABDOVE	CERIE	FERIE	RXBKIE	RXFOIE	TXCIE
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **TXMTIE:** Transmit Shift Register Empty Interrupt Enable bit
1 = Interrupt enabled
0 = Interrupt not enabled

bit 6 **PERIE**: Parity Error Interrupt Enable bit
1 = Interrupt enabled
0 = Interrupt not enabled

bit 5 **ABDOVE:** Auto-baud Detect Overflow Interrupt Enable bit
1 = Interrupt enabled
0 = Interrupt not enabled

bit 4 **CERIE**: Checksum Error Interrupt Enable bit
1 = Interrupt enabled
0 = Interrupt not enabled

bit 3 **FERIE:** Framing Error Interrupt Enable bit
1 = Interrupt enabled
0 = Interrupt not enabled

bit 2 **RXBKIE:** Break Reception Interrupt Enable bit
1 = Interrupt enabled
0 = Interrupt not enabled

bit 1 **RXFOIE:** Receive FIFO Overflow Interrupt Enable bit
1 = Interrupt enabled
0 = Interrupt not enabled

bit 0 **TXCIE:** Transmit Collision Interrupt Enable bit
1 = Interrupt enabled
0 = Interrupt not enabled

If a BTO event occurs when the module is configured as a master and is active, (i.e., MMA bit is set), and the module immediately tries to assert a Stop condition and also sets the BTOIF bit. The actual generation of the Stop condition may be delayed if the bus is been clock stretched by some slave device. The MMA bit will be cleared only after the Stop condition is generated.

33.3.10 ADDRESS BUFFERS

The I^2C module has two address buffer registers, I2CxADB0 and I2CxADB1. Depending on the mode, these registers are used as either receive or transmit address buffers. See Table 33-2 for data flow directions in these registers. In Slave modes, these registers are only updated when there is an address match. The ADB bit in the I2CxCON2 register is used to enable/ disable the address buffer functionality. When disabled, the address data is sourced from the transmit buffer and is stored in the receive buffer.

TABLE 33-2: ADDRESS BUFFER DIRECTION AS PER I^2C MODE

Modes	MODE<2:0>	I2CxADB0	I2CxADB1
Slave (7-bit)	000	RX	—
	001	RX	—
Slave (10-bit)	010	RX	RX
	011	RX	RX
Master (7-bit)	100	—	TX
Master (10-bit)	101	TX	TX
Multi-Master (7-bit)	110	RX	TX
	111	RX	TX

33.3.10.1 Slave Mode (7-bit)

In 7-bit Slave mode, I2CxADB0 is loaded with the received matching address and R/W data. The I2CxADB1 register is ignored in this mode.

33.3.10.2 Slave Mode (10-bit)

In 10-bit Slave mode, I2CxADB0 is loaded with the lower eight bits of the matching received address. I2CxADB1 is loaded with full eight bits of the high address byte, including the R/W bit.

33.3.10.3 Master Mode (7-bit)

The I2CxADB0 register is ignored in this mode. In 7-bit Master mode, the I2CxADB1 register is used to copy address data byte, including the R/W value, to the shift register.

33.3.10.4 Master Mode (10-bit)

In 10-bit Master mode, the I2CxADB0 register stores the low address data byte value that will be copied to the shift register after the high address byte is shifted out. The I2CxADB1 register stores the high address byte value that will be copied to the shift register. It is up to the user to specify all eight of these bits, even though the I^2C specification defines the upper five bits as a constant.

33.3.10.5 Multi-Master Mode (7-bit only)

In Multi-Master mode, the device can be both master and slave depending on the sequence of events on the bus. If being addressed as a slave, the I2CxADB0 register stores the received matching slave address byte. If the device is trying to communicate as a master on the bus, the contents of the I2CxADB1 register are copied to the shift register for addressing a slave device.

33.3.11 RECEIVE AND TRANSMIT BUFFER

The receive buffer holds one byte of data while another is shifted into the SDA pin. The user can access the buffer by software (or DMA) through the I2CxRXB register. When new data is loaded into the I2CxRXB register, the receive buffer full Status bit (RXBF) is set and reading the I2CxRXB register clears this bit.

If the user tries to read I2CxRXB when it is empty (i.e., RXBF = 0), receive read error bit (RXRE) is set and a NACK will be generated. The user must clear the error bit to resume normal operation.

The transmit buffer holds one byte of data while another can be shifted out through the SDA pin. The user can access the buffer by software (or DMA) through the I2CxTXB register. When the I2CxTXB does not contain any transmit data, the transmit buffer empty status bit (TXBE) is set. At this point, the user can load another byte into the buffer.

If the user tries to write I2CxTXB when it is NOT empty (i.e. TXBE = 0), transmit write error flag bit (TXRE) is set and the new data is discarded. When TXRE is set, the user must clear this error condition to resume normal operation.

By setting the CLRBF bit in the I2CxSTAT1 register, the user can clear both receive and transmit buffers. CLRBF will also clear the I2CxRXIF and I2CxTXIF bits.

33.3.12 CLOCK STRETCHING

When a slave device has not completed processing data, it can delay the transfer of more data through the process of clock stretching. An addressed slave device may hold the SCL clock line low after receiving or sending a bit, indicating that it is not yet ready to continue. The master will attempt to raise the SCL line in order to transfer the next bit, but will detect that the clock line has not yet been released. Since the SCL connection is open-drain, the slave has the ability to hold the line low until it is ready to continue communicating. Clock stretching allows receivers that cannot keep up with a transmitter to control the flow of incoming data.

33.5.10 MASTER RECEPTION IN 7-BIT ADDRESSING MODE

This section describes the sequence of events for the I^2C module configured as an I^2C master in 7-bit Addressing mode and is receiving data. Figure 33-20 is used as a visual reference for this description.

1. Master software loads slave address in I2CxADB1 with R/W bit = d and number of bytes to be received in one sequence in I2CxCNT register.
2. Master hardware waits for BFRE bit to be set; then shifts out start and address with R/W = 1.
3. Master sends out the 9th SCL pulse for ACK, master hardware clocks in ACK from Slave
4. If ABD = 0; i.e., Address buffers are enabled

If NACK, master hardware sends Stop or sets MDR (if RSEN = 1) and waits for user software to write to S bit for restart.

If ABD = 1; i.e., Address buffers are disabled

If NACK, master hardware sends Stop or sets MDR (if RSEN = 1) and waits for user software to load the new address into I2CxTXB. Software writes to the S bit are ignored in this case.

5. If ACK, master hardware receives 7-bits of data into the shift register.
6. If the receive buffer is full (i.e., RXBF = 1), clock is stretched on 7th falling SCL edge.
7. Master software must read previous data out of I2CxRXB to clear RXBF.
8. Master hardware receives 8th bit of data into the shift register and loads it into I2CxRXB, sets I2CxRXIF and RXBF bits. I2CxCNT is decremented.
9. If I2CxCNT! = 0, master hardware clocks out ACKDT as ACK value to slave. If I2CxCNT = 0, master hardware clocks out ACKCNT as ACK value to slave. It is up to the user to set the values of ACKDT and ACKCNT correctly. If the user does not set ACKCNT to '1', the master hardware will never send a NACK when I2CxCNT becomes zero. Since a NACK was not seen on the bus, the master hardware will also not assert a Stop condition.
10. Go to step 4.

REGISTER 33-2: I2CxCON1: I²C CONTROL REGISTER 1

R/W-0	R/W-0	R-0	R-0	U-0	R/W/HS-0	R/W/HS-0	R/W-0
ACKCNT(2)	ACKDT(1,2)	ACKSTAT	ACKT	—	RXO	TXU	CSD
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	HS = Hardware set HC = Hardware clear

bit 7 **ACKCNT:** Acknowledge End of Count bit(2)
Acknowledge value transmitted after received data, when I2CCNT = 0
1 = Not Acknowledge (copied to SDA output)
0 = Acknowledge (copied to SDA output)

bit 6 **ACKDT:** Acknowledge Data bit(1,2)
Acknowledge value transmitted after matching address
Acknowledge value transmitted after received data, when I2CCNT! = 0
1 = Not Acknowledge (copied to SDA output)
0 = Acknowledge (copied to SDA output)

bit 5 **ACKSTAT:** Acknowledge Status bit (Transmission only)
1 = Acknowledge was not received for most recent transmission
0 = Acknowledge was received for most recent transmission

bit 4 **ACKT:** Acknowledge Time Status bit
1 = Indicates the I²C bus is in an Acknowledge sequence, set on 8th falling edge of SCL clock
0 = Not in Acknowledge sequence, cleared on 9th rising edge of SCL

bit 3 **Unimplemented:** Read as 1'b0

bit 2 **RXO:** Receive Overflow Status bit (MODE<2:0> = 0xx & 11x)
This bit can only be set when CSD= 1
1 = Set when SMA = 1, and a master clocks in data when RXBF = 1
0 = No slave overflow condition

bit 1 **TXU:** Transmit Underflow Status bit (MODE<2:0> = 0xx & 11x)
This bit can only be set when CSTRDIS = 1
1 = Set when SMA = 1, and a master clocks out data when TXBE = 1
0 = No slave underflow condition

bit 0 **CSD:** Clock Stretching Disable bit (MODE<2:0> = 0xx & 11x)
1 = When SMA = 1, the CSTR bit will never be set
0 = Slave clock stretching proceeds normally

Note 1: Software writes to ACKDT bit must be followed by a minimum SDA data-setup time before clearing CSTR.
2: NACK may still be generated by I²C hardware when bus errors are indicated in the I2CxSTAT1 or I2CxERR registers.

REGISTER 33-9: I2CxCNT: I²C BYTE COUNT REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
CNT<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	HS = Hardware set HC = Hardware clear

bit 7-0 **CNT<7:0>:** I²C Byte Count Register bits

If receiving data,

decremented 8th SCL edge, when a new data byte is loaded into I2CxRXB

If transmitting data,

decremented 9th SCL edge, when a new data byte is moved from I2CxTXB

CNTIF flag is set on 9th falling SCL edge, when I2CxCNT = 0. (Byte count cannot decrement past '0')

Note 1: It is recommended to write this register only when the module is IDLE (MMA = 0, SMA = 0) or when clock stretching (CSTR = 1 || MDR = 1).

MOVFFL **Move f to f (Long Range)**

Syntax: MOVFFL f_s,f_d

Operands: $0 \le f_s \le 16383$
$0 \le f_d \le 16383$

Operation: $(f_s) \rightarrow f_d$

Status Affected: None

Encoding:

1st word	0000	0000	0110	$f_sf_sf_sf_s$
2nd word	1111	$f_sf_sf_sf_s$	$f_sf_sf_sf_s$	$f_sf_sf_df_d$
3rd word	1111	$f_df_df_df_d$	$f_df_df_df_d$	$f_df_df_df_d$

Description: The contents of source register 'f_s' are moved to destination register 'f_d'. Location of source 'f_s' can be anywhere in the 16 Kbyte data space (0000h to 3FFFh). Either source or destination can be W (a useful special situation).
MOVFFL is particularly useful for transferring a data memory location to a peripheral register (such as the transmit buffer or an I/O port).
The MOVFFL instruction cannot use the PCL, TOSU, TOSH or TOSL as the destination register.

Words: 3

Cycles: 3

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	No operation	No operation
Decode	Read register 'f_s' (src)	Process data	No operation
Decode	No operation No dummy read	No operation	Write register 'f_d' (dest)

Example:
```
MOVFFL   2000h, 200Ah
```
Before Instruction
Contents of 2000h = 33h
Contents of 200Ah = 11h
After Instruction
Contents of 2000h = 33h
Contents of 200Ah = 33h

MOVLB **Move literal to BSR**

Syntax: MOVLW k

Operands: $0 \le k \le 63$

Operation: $k \rightarrow$ BSR

Status Affected: None

Encoding:

0000	0001	00kk	kkkk

Description: The 6-bit literal 'k' is loaded into the Bank Select Register (BSR<5:0>). The value of BSR<7:6> always remains '0'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write literal 'k' to BSR

Example:
```
MOVLB    5
```
Before Instruction
BSR Register = 02h
After Instruction
BSR Register = 05h

TABLE 42-1: REGISTER FILE SUMMARY FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
3A16h	RC6PPS	—	—	—	RC6PPS4	RC6PPS3	RC6PPS2	RC6PPS1	RC6PPS0	280
3A15h	RC5PPS	—	—	—	RC5PPS4	RC5PPS3	RC5PPS2	RC5PPS1	RC5PPS0	280
3A14h	RC4PPS	—	—	—	RC4PPS4	RC4PPS3	RC4PPS2	RC4PPS1	RC4PPS0	280
3A13h	RC3PPS	—	—	—	RC3PPS4	RC3PPS3	RC3PPS2	RC3PPS1	RC3PPS0	280
3A12h	RC2PPS	—	—	—	RC2PPS4	RC2PPS3	RC2PPS2	RC2PPS1	RC2PPS0	280
3A11h	RC1PPS	—	—	—	RC1PPS4	RC1PPS3	RC1PPS2	RC1PPS1	RC1PPS0	280
3A10h	RC0PPS	—	—	—	RC0PPS4	RC0PPS3	RC0PPS2	RC0PPS1	RC0PPS0	280
3A0Fh	RB7PPS	—	—	—	RB7PPS4	RB7PPS3	RB7PPS2	RB7PPS1	RB7PPS0	280
3A0Eh	RB6PPS	—	—	—	RB6PPS4	RB6PPS3	RB6PPS2	RB6PPS1	RB6PPS0	280
3A0Dh	RB5PPS	—	—	—	RB5PPS4	RB5PPS3	RB5PPS2	RB5PPS1	RB5PPS0	280
3A0Ch	RB4PPS	—	—	—	RB4PPS4	RB4PPS3	RB4PPS2	RB4PPS1	RB4PPS0	280
3A0Bh	RB3PPS	—	—	—	RB3PPS4	RB3PPS3	RB3PPS2	RB3PPS1	RB3PPS0	280
3A0Ah	RB2PPS	—	—	—	RB2PPS4	RB2PPS3	RB2PPS2	RB2PPS1	RB2PPS0	280
3A09h	RB1PPS	—	—	—	RB1PPS4	RB1PPS3	RB1PPS2	RB1PPS1	RB1PPS0	280
3A08h	RB0PPS	—	—	—	RB0PPS4	RB0PPS3	RB0PPS2	RB0PPS1	RB0PPS0	280
3A07h	RA7PPS	—	—	—	RA7PPS4	RA7PPS3	RA7PPS2	RA7PPS1	RA7PPS0	280
3A06h	RA6PPS	—	—	—	RA6PPS4	RA6PPS3	RA6PPS2	RA6PPS1	RA6PPS0	280
3A05h	RA5PPS	—	—	—	RA5PPS4	RA5PPS3	RA5PPS2	RA5PPS1	RA5PPS0	280
3A04h	RA4PPS	—	—	—	RA4PPS4	RA4PPS3	RA4PPS2	RA4PPS1	RA4PPS0	280
3A03h	RA3PPS	—	—	—	RA3PPS4	RA3PPS3	RA3PPS2	RA3PPS1	RA3PPS0	280
3A02h	RA2PPS	—	—	—	RA2PPS4	RA2PPS3	RA2PPS2	RA2PPS1	RA2PPS0	280
3A01h	RA1PPS	—	—	—	RA1PPS4	RA1PPS3	RA1PPS2	RA1PPS1	RA1PPS0	280
3A00h	RA0PPS	—	—	—	RA0PPS4	RA0PPS3	RA0PPS2	RA0PPS1	RA0PPS0	280
39FFh - 39F8h	—	Unimplemented								
39F7h	SCANPR	—	—	—	—	—	PR			31
39F6h - 39F5h	—	Unimplemented								
39F4h	DMA2PR	—	—	—	—	—	PR			31
39F3h	DMA1PR	—	—	—	—	—	PR			30
39F2h	MAINPR	—	—	—	—	—	PR			30
39F1h	ISRPR	—	—	—	—	—	PR			30
39F0h	—	Unimplemented								
39EFh	PRLOCK	—	—	—	—	—	—	—	PRLOCKED	31
39EEh - 39E7h	—	Unimplemented								
39E6h	NVMCON2	NVMCON2								211
39E5h	NVMCON1	REG		—	FREE	WRERR	WREN	WR	RD	210
39E4h	—	Unimplemented								
39E3h	NVMDAT	DAT								212
39E2h	—	Unimplemented								
39E1h	NVMADRH(4)	—	—	—	—	—	—	ADR		211
39E0h	NVMADRL	ADR								211
39DFh	OSCFRQ	—	—	—	—	FRQ				107
39DEh	OSCTUNE	—	—	TUN						108
39DDh	OSCEN	EXTOEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOEN	—	—	109
39DCh	OSCSTAT	EXTOR	HFOR	MFOR	LFOR	SOR	ADOR	—	PLLR	106
39DBh	OSCCON3	CSWHOLD	SOSCPWR	—	ORDY	NOSCR	—	—	—	105

Legend: x = unknown, u = unchanged, — = unimplemented, q = value depends on condition

Note 1: Unimplemented in LF devices.

2: Unimplemented in PIC18(L)F26/27K42.

3: Unimplemented on PIC18(L)F26/27/45/46/47K42 devices.

4: Unimplemented in PIC18(L)F45/55K42.

44.3 DC Characteristics

TABLE 44-1: SUPPLY VOLTAGE

PIC18(L)F26/27/45/46/47/55/56/57K42			Standard Operating Conditions (unless otherwise stated)				
PIC18F26/45/46/55/56K42							
Param. No.	Sym.	Characteristic	Min.	Typ.†	Max.	Units	Conditions
Supply Voltage							
D002	VDD		1.8 2.5 2.7	— — —	3.6 3.6 3.6	V V V	FOSC ≤ 16 MHz FOSC > 32 MHz FOSC > 64 MHz
D002	VDD		2.3 2.5 2.7	— — —	5.5 5.5 5.5	V V V	FOSC ≤ 16 MHz FOSC > 32 MHz FOSC > 64 MHz
RAM Data Retention(1)							
D003	VDR		1.5	—	—	V	Device in Sleep mode
D003	VDR		1.7	—	—	V	Device in Sleep mode
Power-on Reset Release Voltage(2)							
D004	VPOR		—	1.6	—	V	BOR or LPBOR disabled(3)
D004	VPOR		—	1.6	—	V	BOR or LPBOR disabled(3)
Power-on Reset Rearm Voltage(2)							
D005	VPORR		—	0.8	—	V	BOR or LPBOR disabled(3)
D005	VPORR		—	1.5	—	V	BOR or LPBOR disabled(3)
VDD Rise Rate to ensure internal Power-on Reset signal(2)							
D006	SVDD		0.05	—	—	V/ms	BOR or LPBOR disabled(3)
D006	SVDD		0.05	—	—	V/ms	BOR or LPBOR disabled(3)

† Data in "Typ." column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: This is the limit to which VDD can be lowered in Sleep mode without losing RAM data.

2: See Figure 44-3, POR and POR REARM with Slow Rising VDD.

3: See Table 44-13 for BOR and LPBOR trip point information.