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	40MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	4KB (2K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	4.2V ~ 5.5V
Data Converters	A/D 13x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f4221-i-ml

TABLE 1-1: DEVICE FEATURES

Features	PIC18F2221	PIC18F2321	PIC18F4221	PIC18F4321
Operating Frequency	DC – 40 MHz	DC – 40 MHz	DC – 40 MHz	DC – 40 MHz
Program Memory (Bytes)	4096	8192	4096	8192
Program Memory (Instructions)	2048	4096	2048	4096
Data Memory (Bytes)	512	512	512	512
Data EEPROM Memory (Bytes)	256	256	256	256
Interrupt Sources	19	19	20	20
I/O Ports	Ports A, B, C, (E)	Ports A, B, C, (E)	Ports A, B, C, D, E	Ports A, B, C, D, E
Timers	4	4	4	4
Capture/Compare/PWM Modules	2	2	1	1
Enhanced Capture/Compare/ PWM Modules	0	0	1	1
Serial Communications	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART
Parallel Communications (PSP)	No	No	Yes	Yes
10-bit Analog-to-Digital Module	10 Input Channels	10 Input Channels	13 Input Channels	13 Input Channels
Resets (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT
Programmable Low-Voltage Detect	Yes	Yes	Yes	Yes
Programmable Brown-out Reset	Yes	Yes	Yes	Yes
Instruction Set	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled
Packages	28-pin SPDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	28-pin SPDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	40-pin PDIP 44-pin QFN 44-pin TQFP	40-pin PDIP 44-pin QFN 44-pin TQFP

NOTES:

REGISTER 3-2:   **OSCCON: OSCILLATOR CONTROL REGISTER**

R/W-0	R/W-1	R/W-0	R/W-0	R[(1)]	R-0	R/W-0	R/W-0
IDLEN	IRCF2	IRCF1	IRCF0	OSTS	IOFS	SCS1	SCS0
bit 7							bit 0

bit 7   **IDLEN:** Idle Enable bit

1 = Device enters an Idle mode when a SLEEP instruction is executed
0 = Device enters Sleep mode when a SLEEP instruction is executed

bit 6-4   **IRCF<2:0>:** Internal Oscillator Frequency Select bits

111 = 8 MHz (INTOSC drives clock directly)
110 = 4 MHz
101 = 2 MHz
100 = 1 MHz[(3)]
011 = 500 kHz
010 = 250 kHz
001 = 125 kHz
000 = 31 kHz (from either INTOSC/256 or INTRC directly)[(2)]

bit 3   **OSTS:** Oscillator Start-up Time-out Status bit[(1)]

1 = Oscillator Start-up Timer (OST) time-out has expired; primary oscillator is running
0 = Oscillator Start-up Timer (OST) time-out is running; primary oscillator is not ready

bit 2   **IOFS:** INTOSC Frequency Stable bit

1 = INTOSC frequency is stable
0 = INTOSC frequency is not stable

bit 1-0   **SCS<1:0>:** System Clock Select bits

1x = Internal oscillator block
01 = Secondary (Timer1) oscillator
00 = Primary oscillator

> **Note 1:** Reset state depends on state of the IESO Configuration bit.
>
> **2:** Source selected by the INTSRC bit (OSCTUNE<7>), see text.
>
> **3:** Default output frequency of INTOSC on Reset.

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

EXAMPLE 7-3:   WRITING TO FLASH PROGRAM MEMORY

```
                MOVLW    D'64'                  ; number of bytes in erase block
                MOVWF    COUNTER
                MOVLW    BUFFER_ADDR_HIGH       ; point to buffer
                MOVWF    FSR0H
                MOVLW    BUFFER_ADDR_LOW
                MOVWF    FSR0L
                MOVLW    CODE_ADDR_UPPER        ; Load TBLPTR with the base
                MOVWF    TBLPTRU                ; address of the memory block
                MOVLW    CODE_ADDR_HIGH
                MOVWF    TBLPTRH
                MOVLW    CODE_ADDR_LOW          ; 6 LSB = 0
                MOVWF    TBLPTRL
READ_BLOCK
                TBLRD*+                         ; read into TABLAT, and inc
                MOVF     TABLAT, W              ; get data
                MOVWF    POSTINC0               ; store data and increment FSR0
                DECFSZ   COUNTER                ; done?
                BRA      READ_BLOCK             ; repeat
MODIFY_WORD
                MOVLW    DATA_ADDR_HIGH         ; point to buffer
                MOVWF    FSR0H
                MOVLW    DATA_ADDR_LOW
                MOVWF    FSR0L
                MOVLW    NEW_DATA_LOW           ; update buffer word and increment FSR0
                MOVWF    POSTINC0
                MOVLW    NEW_DATA_HIGH          ; update buffer word
                MOVWF    INDF0
ERASE_BLOCK
                MOVLW    CODE_ADDR_UPPER        ; load TBLPTR with the base
                MOVWF    TBLPTRU                ; address of the memory block
                MOVLW    CODE_ADDR_HIGH
                MOVWF    TBLPTRH
                MOVLW    CODE_ADDR_LOW          ; 6 LSB = 0
                MOVWF    TBLPTRL
                BCF      EECON1, CFGS           ; point to PROG/EEPROM memory
                BSF      EECON1, EEPGD          ; point to Flash program memory
                BSF      EECON1, WREN           ; enable write to memory
                BSF      EECON1, FREE           ; enable Row Erase operation
                BCF      INTCON, GIE            ; disable interrupts
                MOVLW    55h                    ; Required sequence
                MOVWF    EECON2                 ; write 55h
                MOVLW    0AAh
                MOVWF    EECON2                 ; write AAh
                BSF      EECON1, WR             ; start erase (CPU stall)
                NOP
                BSF      INTCON, GIE            ; re-enable interrupts
WRITE_BUFFER_BACK
                MOVLW    8                      ; number of write buffer groups of 8 bytes
                MOVWF    COUNTER_HI
                MOVLW    BUFFER_ADDR_HIGH       ; point to buffer
                MOVWF    FSR0H
                MOVLW    BUFFER_ADDR_LOW
                MOVWF    FSR0L
PROGRAM_LOOP
                MOVLW    8                      ; number of bytes in holding register
                MOVWF    COUNTER
WRITE_WORD_TO_HREGS
                MOVF     POSTINC0, W            ; get low byte of buffer data and increment FSR0
                MOVWF    TABLAT                 ; present data to table latch
                TBLWT+*                         ; short write
                                                ; to internal TBLWT holding register, increment
                                                ; TBLPTR
                DECFSZ   COUNTER                ; loop until buffers are full
                GOTO     WRITE_WORD_TO_HREGS
```

8.5   Operation During Code-Protect

Data EEPROM memory has its own code-protect bits in Configuration Words. External read and write operations are disabled if code protection is enabled.

The microcontroller itself can both read and write to the internal data EEPROM, regardless of the state of the code-protect Configuration bit. Refer to **Section 24.0 "Special Features of the CPU"** for additional information.

8.6   Protection Against Spurious Write

To protect against spurious EEPROM writes, various mechanisms have been implemented. On power-up, the WREN bit is cleared. In addition, writes to the EEPROM are blocked during the Power-up Timer period (T$_{PWRT}$, parameter 33).

The write initiate sequence and the WREN bit together help prevent an accidental write during Brown-out Reset, power glitch or software malfunction.

8.7   Using the Data EEPROM

The data EEPROM is a high-endurance, byte addressable array that has been optimized for the storage of frequently changing data. Such data is typically updated at least one time within the number of writes defined by specification, D124. If any location storing data is not written at least this often, the data EEPROM array must be refreshed. For this reason, values that change infrequently, or not at all, should be stored in Flash program memory.

A simple data EEPROM refresh routine is shown in Example 8-3.

Note:	If data EEPROM is only used to store constants and/or data that changes often, an array refresh is likely not required. See specification, D124.

EXAMPLE 8-3:   DATA EEPROM REFRESH ROUTINE

```
        CLRF    EEADR               ; Start at address 0
        BCF     EECON1, CFGS        ; Set for memory
        BCF     EECON1, EEPGD       ; Set for Data EEPROM
        BCF     INTCON, GIE         ; Disable interrupts
        BSF     EECON1, WREN        ; Enable writes
LOOP                                ; Loop to refresh array
        BSF     EECON1, RD          ; Read current address
        MOVLW   55h                 ;
        MOVWF   EECON2              ; Write 55h
        MOVLW   0AAh                ;
        MOVWF   EECON2              ; Write 0AAh
        BSF     EECON1, WR          ; Set WR bit to begin write
        BTFSC   EECON1, WR          ; Wait for write to complete
        BRA     $-2
        INCFSZ  EEADR, F            ; Increment address
        BRA     LOOP                ; Not zero, do it again

        BCF     EECON1, WREN        ; Disable writes
        BSF     INTCON, GIE         ; Enable interrupts
```

REGISTER 10-3: **INTCON3: INTERRUPT CONTROL REGISTER 3**

R/W-1	R/W-1	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
INT2IP	INT1IP	—	INT2IE	INT1IE	—	INT2IF	INT1IF
bit 7							bit 0

bit 7 **INT2IP:** INT2 External Interrupt Priority bit

1 = High priority
0 = Low priority

bit 6 **INT1IP:** INT1 External Interrupt Priority bit

1 = High priority
0 = Low priority

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

bit 4 **INT2IE:** INT2 External Interrupt Enable bit

1 = Enables the INT2 external interrupt
0 = Disables the INT2 external interrupt

bit 3 **INT1IE:** INT1 External Interrupt Enable bit

1 = Enables the INT1 external interrupt
0 = Disables the INT1 external interrupt

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

bit 1 **INT2IF:** INT2 External Interrupt Flag bit

1 = The INT2 external interrupt occurred (must be cleared in software)
0 = The INT2 external interrupt did not occur

bit 0 **INT1IF:** INT1 External Interrupt Flag bit

1 = The INT1 external interrupt occurred (must be cleared in software)
0 = The INT1 external interrupt did not occur

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

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global interrupt enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt. This feature allows for software polling.

REGISTER 10-9: **IPR2: PERIPHERAL INTERRUPT PRIORITY REGISTER 2**

R/W-1	R/W-1	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
OSCFIP	CMIP	—	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP
bit 7							bit 0

bit 7 **OSCFIP:** Oscillator Fail Interrupt Priority bit

1 = High priority
0 = Low priority

bit 6 **CMIP:** Comparator Interrupt Priority bit

1 = High priority
0 = Low priority

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

bit 4 **EEIP:** Data EEPROM/Flash Write Operation Interrupt Priority bit

1 = High priority
0 = Low priority

bit 3 **BCLIP:** Bus Collision Interrupt Priority bit

1 = High priority
0 = Low priority

bit 2 **HLVDIP:** High/Low-Voltage Detect Interrupt Priority bit

1 = High priority
0 = Low priority

bit 1 **TMR3IP:** TMR3 Overflow Interrupt Priority bit

1 = High priority
0 = Low priority

bit 0 **CCP2IP:** CCP2 Interrupt Priority bit

1 = High priority
0 = Low priority

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

11.0 I/O PORTS

Depending on the device selected and features enabled, there are up to five ports available. Some pins of the I/O ports are multiplexed with an alternate function from the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Each port has three registers for its operation. These registers are:

• TRIS register (Data Direction register)
• PORT register (reads the levels on the pins of the device)
• LAT register (Data Latch register)

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

A simplified model of a generic I/O port, without the interfaces to other peripherals, is shown in Figure 11-1.

FIGURE 11-1: GENERIC I/O PORT OPERATION

Note 1: I/O pins have diode protection to VDD and VSS.
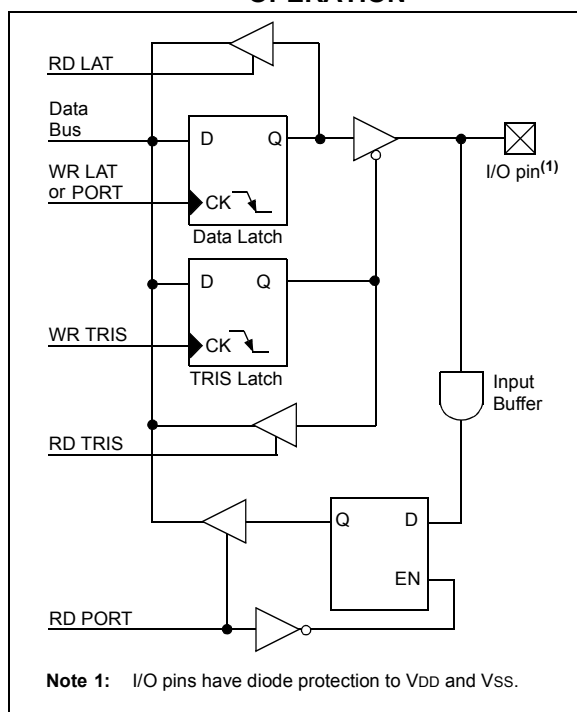
11.1 PORTA, TRISA and LATA Registers

PORTA is an 8-bit wide, bidirectional port. The corresponding Data Direction register is TRISA. Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., put the contents of the output latch on the selected pin).

Reading the PORTA register reads the status of the pins, whereas writing to it, will write to the port latch.

The Data Latch (LATA) register is also memory mapped. Read-modify-write operations on the LATA register read and write the latched output value for PORTA.

The RA4 pin is multiplexed with the Timer0 module clock input and one of the comparator outputs to become the RA4/T0CKI/C1OUT pin. Pins RA6 and RA7 are multiplexed with the main oscillator pins. They are enabled as oscillator or I/O pins by the selection of the main oscillator in the Configuration register (see **Section 24.1 "Configuration Bits"** for details). When they are not used as port pins, RA6 and RA7 and their associated TRIS and LAT bits are read as '0'.

The other PORTA pins are multiplexed with analog inputs, the analog VREF+ and VREF- inputs and the comparator voltage reference output. The operation of pins RA<3:0> and RA5 as A/D converter inputs is selected by clearing or setting the control bits in the ADCON1 register (A/D Control Register 1).

Pins RA0 through RA5 may also be used as comparator inputs or outputs by setting the appropriate bits in the CMCON register. To use RA<3:0> as digital inputs, it is also necessary to turn off the comparators.

> **Note:** On a Power-on Reset, RA5 and RA<3:0> are configured as analog inputs and read as '0'. RA4 is configured as a digital input.

The RA4/T0CKI/C1OUT pin is a Schmitt Trigger input. All other PORTA pins have TTL input levels and full CMOS output drivers.

The TRISA register controls the direction of the PORTA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

EXAMPLE 11-1: INITIALIZING PORTA

```
CLRF    PORTA   ; Initialize PORTA by
                ; clearing output
                ; data latches
CLRF    LATA    ; Alternate method
                ; to clear output
                ; data latches
MOVLW   0Fh     ; Configure all A/D
MOVWF   ADCON1  ; for digital inputs
MOVWF   07h     ; Configure comparators
MOVWF   CMCON   ; for digital input
MOVLW   0CFh    ; Value used to
                ; initialize data
                ; direction
MOVWF   TRISA   ; Set RA<7:6,3:0> as inputs
                ; RA<5:4> as outputs
```

11.4 PORTD, TRISD and LATD Registers

Note:	PORTD is only available on 40/44-pin devices.

PORTD is an 8-bit wide, bidirectional port. The corresponding Data Direction register is TRISD. Setting a TRISD bit (= 1) will make the corresponding PORTD pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISD bit (= 0) will make the corresponding PORTD pin an output (i.e., put the contents of the output latch on the selected pin).

The Data Latch register (LATD) is also memory mapped. Read-modify-write operations on the LATD register read and write the latched output value for PORTD.

All pins on PORTD are implemented with Schmitt Trigger input buffers. Each pin is individually configurable as an input or output.

Three of the PORTD pins are multiplexed with outputs P1B, P1C and P1D of the Enhanced CCP module. The operation of these additional PWM output pins is covered in greater detail in **Section 17.0 "Enhanced Capture/Compare/PWM (ECCP) Module"**.

Note:	On a Power-on Reset, these pins are configured as digital inputs.

PORTD can also be configured as an 8-bit wide microprocessor port (Parallel Slave Port) by setting control bit, PSPMODE (TRISE<4>). In this mode, the input buffers are TTL. See **Section 11.6 "Parallel Slave Port"** for additional information on the Parallel Slave Port (PSP).

Note:	When the Enhanced PWM mode is used with either dual or quad outputs, the PSP functions of PORTD are automatically disabled.

EXAMPLE 11-4: INITIALIZING PORTD

```
CLRF    PORTD   ; Initialize PORTD by
                ; clearing output
                ; data latches
CLRF    LATD    ; Alternate method
                ; to clear output
                ; data latches
MOVLW   0CFh    ; Value used to
                ; initialize data
                ; direction
MOVWF   TRISD   ; Set RD<3:0> as inputs
                ; RD<5:4> as outputs
                ; RD<7:6> as inputs
```

17.4.4 HALF-BRIDGE MODE

In the Half-Bridge Output mode, two pins are used as outputs to drive push-pull loads. The PWM output signal is output on the P1A pin, while the complementary PWM output signal is output on the P1B pin (Figure 17-4). This mode can be used for half-bridge applications, as shown in Figure 17-5, or for full-bridge applications where four power switches are being modulated with two PWM signals.

In Half-Bridge Output mode, the programmable dead-band delay can be used to prevent shoot-through current in half-bridge power devices. The value of bits, PDC<6:0>, sets the number of instruction cycles before the output is driven active. If the value is greater than the duty cycle, the corresponding output remains inactive during the entire cycle. See **Section 17.4.6 "Programmable Dead-Band Delay"** for more details of the dead-band delay operations.

Since the P1A and P1B outputs are multiplexed with the PORTC<2> and PORTD<5> data latches, the TRISC<2> and TRISD<5> bits must be cleared to configure P1A and P1B as outputs.

FIGURE 17-4: HALF-BRIDGE PWM OUTPUT

td = Dead-Band Delay

Note 1: At this time, the TMR2 register is equal to the PR2 register.
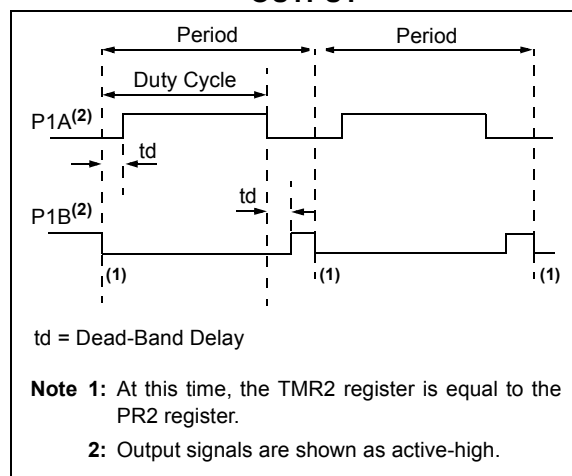
2: Output signals are shown as active-high.
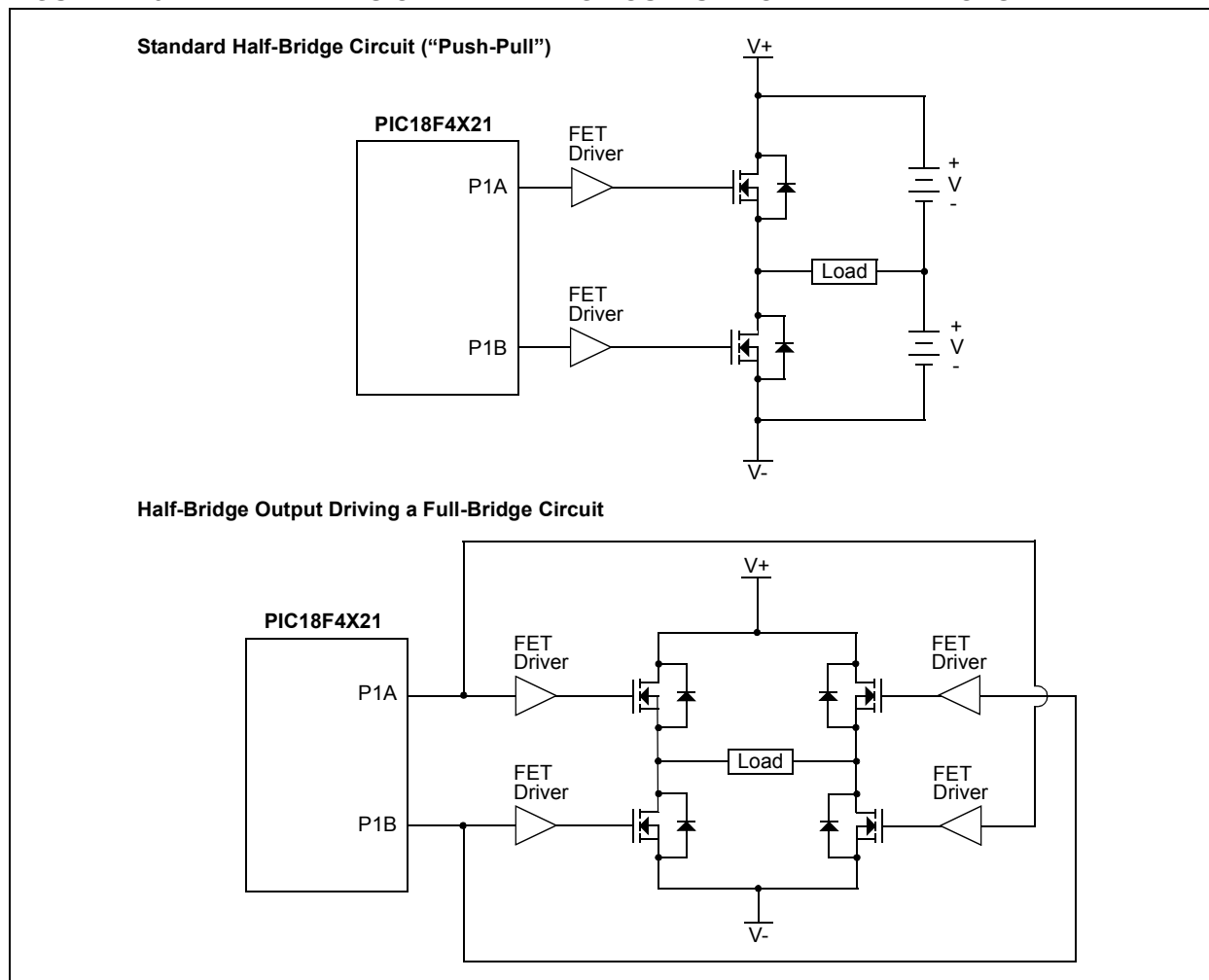
FIGURE 17-5: EXAMPLES OF HALF-BRIDGE OUTPUT MODE APPLICATIONS

FIGURE 18-8: I²C™ SLAVE MODE TIMING WITH SEN = 0 (RECEPTION, 7-BIT ADDRESSING)

18.4.6.1 I²C Master Mode Operation

The master device generates all of the serial clock pulses and the Start and Stop conditions. A transfer is ended with a Stop condition or with a Repeated Start condition. Since the Repeated Start condition is also the beginning of the next serial transfer, the I²C bus will not be released.

In Master Transmitter mode, serial data is output through SDA, while SCL outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the Read/Write (R/$\overline{\text{W}}$) bit. In this case, the R/$\overline{\text{W}}$ bit will be logic '0'. Serial data is transmitted 8 bits at a time. After each byte is transmitted, an Acknowledge bit is received. Start and Stop conditions are output to indicate the beginning and the end of a serial transfer.

In Master Receive mode, the first byte transmitted contains the slave address of the transmitting device (7 bits) and the R/$\overline{\text{W}}$ bit. In this case, the R/$\overline{\text{W}}$ bit will be logic '1'. Thus, the first byte transmitted is a 7-bit slave address followed by a '1' to indicate the receive bit. Serial data is received via SDA, while SCL outputs the serial clock. Serial data is received 8 bits at a time. After each byte is received, an Acknowledge bit is transmitted. Start and Stop conditions indicate the beginning and end of transmission.

The Baud Rate Generator used for the SPI mode operation is used to set the SCL clock frequency for either 100 kHz, 400 kHz or 1 MHz I²C operation. See **Section 18.4.7 "Baud Rate"** for more detail.

A typical transmit sequence would go as follows:

1. The user generates a Start condition by setting the Start Enable bit, SEN (SSPCON2<0>).
2. SSPIF is set. The MSSP module will wait the required start time before any other operation takes place.
3. The user loads the SSPBUF with the slave address to transmit.
4. Address is shifted out the SDA pin until all 8 bits are transmitted.
5. The MSSP module shifts in the ACK bit from the slave device and writes its value into the SSPCON2 register.
6. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPIF bit.
7. The user loads the SSPBUF with eight bits of data.
8. Data is shifted out the SDA pin until all 8 bits are transmitted.
9. The MSSP module shifts in the ACK bit from the slave device and writes its value into the SSPCON2 register.
10. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPIF bit.
11. The user generates a Stop condition by setting the Stop Enable bit, PEN (SSPCON2<2>).
12. Interrupt is generated once the Stop condition is complete.

18.4.9 I²C MASTER MODE REPEATED START CONDITION TIMING

A Repeated Start condition occurs when the RSEN bit (SSPCON2<1>) is programmed high and the I²C logic module is in the Idle state. When the RSEN bit is set, the SCL pin is asserted low. When the SCL pin is sampled low, the Baud Rate Generator is loaded with the contents of SSPADD<5:0> and begins counting. The SDA pin is released (brought high) for one Baud Rate Generator count (T$_{BRG}$). When the Baud Rate Generator times out, if SDA is sampled high, the SCL pin will be deasserted (brought high). When SCL is sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<6:0> and begins counting. SDA and SCL must be sampled high for one T$_{BRG}$. This action is then followed by assertion of the SDA pin (SDA = 0) for one T$_{BRG}$ while SCL is high. Following this, the RSEN bit (SSPCON2<1>) will be automatically cleared and the Baud Rate Generator will not be reloaded, leaving the SDA pin held low. As soon as a Start condition is detected on the SDA and SCL pins, the S bit (SSPSTAT<3>) will be set. The SSPIF bit will not be set until the Baud Rate Generator has timed out.

Note 1:	If RSEN is programmed while any other event is in progress, it will not take effect.	
2:	A bus collision during the Repeated Start condition occurs if:	
	• SDA is sampled low when SCL goes from low-to-high.	
	• SCL goes low before SDA is asserted low. This may indicate that another master is attempting to transmit a data '1'.	

Immediately following the SSPIF bit getting set, the user may write the SSPBUF with the 7-bit address in 7-bit mode, or the default first address in 10-bit mode. After the first eight bits are transmitted and an ACK is received, the user may then transmit an additional eight bits of address (10-bit mode) or eight bits of data (7-bit mode).

18.4.9.1 WCOL Status Flag

If the user writes the SSPBUF when a Repeated Start sequence is in progress, the WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

Note:	Because queueing of events is not allowed, writing of the lower 5 bits of SSPCON2 is disabled until the Repeated Start condition is complete.

FIGURE 18-22:   REPEATED START CONDITION WAVEFORM

TABLE 25-2: PIC18FXXXX INSTRUCTION SET (CONTINUED)

Mnemonic, Operands		Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
				MSb			LSb		
LITERAL OPERATIONS									
ADDLW	k	Add Literal and WREG	1	0000	1111	kkkk	kkkk	C, DC, Z, OV, N	
ANDLW	k	AND Literal with WREG	1	0000	1011	kkkk	kkkk	Z, N	
IORLW	k	Inclusive OR Literal with WREG	1	0000	1001	kkkk	kkkk	Z, N	
LFSR	f, k	Move Literal (12-bit) 2nd Word	2	1110	1110	00ff	kkkk	None	
		to FSR(f) 1st Word		1111	0000	kkkk	kkkk		
MOVLB	k	Move Literal to BSR<3:0>	1	0000	0001	0000	kkkk	None	
MOVLW	k	Move Literal to WREG	1	0000	1110	kkkk	kkkk	None	
MULLW	k	Multiply Literal with WREG	1	0000	1101	kkkk	kkkk	None	
RETLW	k	Return with Literal in WREG	2	0000	1100	kkkk	kkkk	None	
SUBLW	k	Subtract WREG from Literal	1	0000	1000	kkkk	kkkk	C, DC, Z, OV, N	
XORLW	k	Exclusive OR Literal with WREG	1	0000	1010	kkkk	kkkk	Z, N	
DATA MEMORY ↔ PROGRAM MEMORY OPERATIONS									
TBLRD*		Table Read	2	0000	0000	0000	1000	None	
TBLRD*+		Table Read with Post-Increment		0000	0000	0000	1001	None	
TBLRD*-		Table Read with Post-Decrement		0000	0000	0000	1010	None	
TBLRD+*		Table Read with Pre-Increment		0000	0000	0000	1011	None	
TBLWT*		Table Write	2	0000	0000	0000	1100	None	
TBLWT*+		Table Write with Post-Increment		0000	0000	0000	1101	None	
TBLWT*-		Table Write with Post-Decrement		0000	0000	0000	1110	None	
TBLWT+*		Table Write with Pre-Increment		0000	0000	0000	1111	None	

Note 1: When a PORT register is modified as a function of itself (e.g., MOVF PORTB, 1, 0), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and where applicable, 'd' = 1), the prescaler will be cleared if assigned.

3: If the Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

4: Some instructions are two-word instructions. The second word of these instructions will be executed as a NOP unless the first word of the instruction retrieves the information embedded in these 16 bits. This ensures that all program memory locations have a valid instruction.

POP	Pop Top of Return Stack
Syntax:	POP
Operands:	None
Operation:	(TOS) → bit bucket
Status Affected:	None

Encoding:

0000	0000	0000	0110

Description: The TOS value is pulled off the return stack and is discarded. The TOS value then becomes the previous value that was pushed onto the return stack. This instruction is provided to enable the user to properly manage the return stack to incorporate a software stack.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	POP TOS value	No operation

Example: POP
            GOTO   NEW

Before Instruction
TOS   =   0031A2h
Stack (1 level down)   =   014332h

After Instruction
TOS   =   014332h
PC   =   NEW

PUSH	Push Top of Return Stack
Syntax:	PUSH
Operands:	None
Operation:	(PC + 2) → TOS
Status Affected:	None

Encoding:

0000	0000	0000	0101

Description: The PC + 2 is pushed onto the top of the return stack. The previous TOS value is pushed down on the stack. This instruction allows implementing a software stack by modifying TOS and then pushing it onto the return stack.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	PUSH PC + 2 onto return stack	No operation	No operation

Example: PUSH

Before Instruction
TOS   =   345Ah
PC   =   0124h

After Instruction
PC   =   0126h
TOS   =   0126h
Stack (1 level down)   =   345Ah

PIC18F2221/2321/4221/4321 FAMILY

25.2.2 EXTENDED INSTRUCTION SET

ADDFSR	**Add Literal to FSR**
Syntax:	ADDFSR f, k
Operands:	$0 \le k \le 63$ $f \in [\,0, 1, 2\,]$
Operation:	FSR(f) + k $\rightarrow$ FSR(f)
Status Affected:	None
Encoding:	1110 1000 ffkk kkkk
Description:	The 6-bit literal 'k' is added to the contents of the FSR specified by 'f'.
Words:	1
Cycles:	1

Q Cycle Activity:

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

<u>Example:</u>   ADDFSR 2, 23h

    Before Instruction
        FSR2   =   03FFh
    After Instruction
        FSR2   =   0422h

ADDULNK	**Add Literal to FSR2 and Return**
Syntax:	ADDULNK k
Operands:	$0 \le k \le 63$
Operation:	FSR2 + k $\rightarrow$ FSR2, (TOS) $\rightarrow$ PC
Status Affected:	None
Encoding:	1110 1000 11kk kkkk
Description:	The 6-bit literal 'k' is added to the contents of FSR2. A RETURN is then executed by loading the PC with the TOS. The instruction takes two cycles to execute; a NOP is performed during the second cycle. This may be thought of as a special case of the ADDFSR instruction, where f = 3 (binary '11'); it operates only on FSR2.
Words:	1
Cycles:	2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write to FSR
No Operation	No Operation	No Operation	No Operation

<u>Example:</u>   ADDULNK 23h

    Before Instruction
        FSR2   =   03FFh
        PC   =   0100h
    After Instruction
        FSR2   =   0422h
        PC   =   (TOS)

Note:	All PIC18 instructions may take an optional label argument preceding the instruction mnemonic for use in symbolic addressing. If a label is used, the instruction syntax then becomes: {label} instruction argument(s).

27.2 DC Characteristics: Power-Down and Supply Current
PIC18F2221/2321/4221/4321 (Industrial)
PIC18LF2221/2321/4221/4321 (Industrial) (Continued)

PIC18LF2221/2321/4221/4321 (Industrial)	Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial				
PIC18F2221/2321/4221/4321 (Industrial, Extended)	**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				

Param No.	Device	Typ	Max	Units	Conditions	
	Supply Current (IDD)[2]					
	PIC18LF2X21/4X21	13	19	μA	-40°C	VDD = 2.0V
		13	19	μA	+25°C	
		13	17	μA	+85°C	
	PIC18LF2X21/4X21	41	45	μA	-40°C	VDD = 3.0V
		34	38	μA	+25°C	
		27	30	μA	+85°C	
	All Devices	104	115	μA	-40°C	VDD = 5.0V
		86	95	μA	+25°C	
		67	75	μA	+85°C	
	Extended Devices Only	68	100	μA	+125°C	
	PIC18LF2X21/4X21	0.31	0.35	mA	-40°C	VDD = 2.0V
		0.31	0.35	mA	+25°C	
		0.31	0.35	mA	+85°C	
	PIC18LF2X21/4X21	0.55	0.60	mA	-40°C	VDD = 3.0V
		0.51	0.60	mA	+25°C	
		0.47	0.60	mA	+85°C	
	All Devices	1.0	1.3	mA	-40°C	VDD = 5.0V
		0.94	1.3	mA	+25°C	
		0.88	1.2	mA	+85°C	
	Extended Devices Only	0.88	1.2	mA	+125°C	

Condition for first block (31 kHz rows): FOSC = 31 kHz (**RC_RUN** mode, INTRC source)

Condition for second block (1 MHz rows): FOSC = 1 MHz (**RC_RUN** mode, INTOSC source)

Legend: Shading of rows is to assist in readability of the table.

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD or VSS, and all features that add delta current disabled (such as WDT, Timer1 Oscillator, BOR, etc.).

2: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD or VSS;
$\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

3: Low-power, Timer1 oscillator is selected unless otherwise indicated, where LPT1OSC (CONFIG3H<2>) = 1.

4: BOR and HLVD enable internal band gap reference. With both modules enabled, current consumption will be less than the sum of both specifications.

5: When operation below -10°C is expected, use T1OSC High-Power mode, where LPT1OSC (CONFIG3H<2>) = 0.
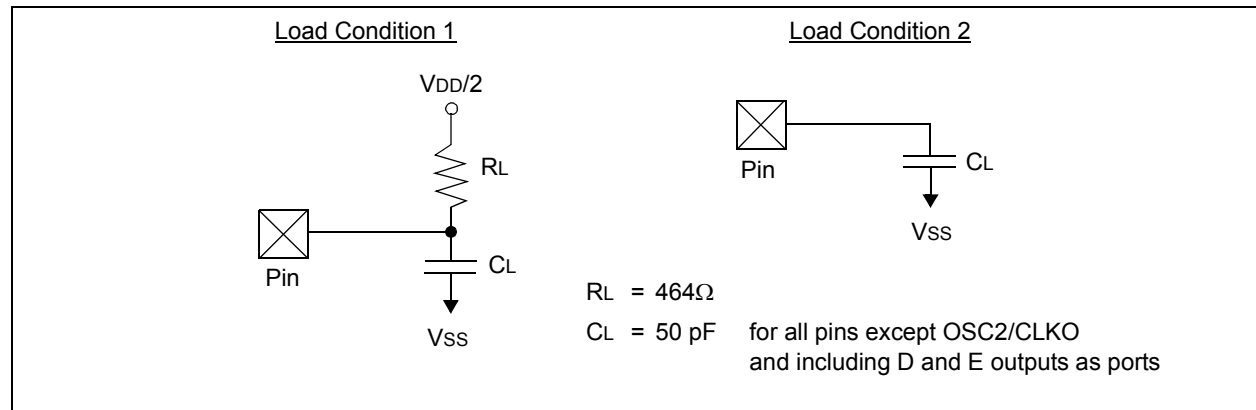
27.4.2   TIMING CONDITIONS

The temperature and voltages specified in Table 27-5 apply to all timing specifications unless otherwise noted. Figure 27-5 specifies the load conditions for the timing specifications.

> **Note:** Because of space limitations, the generic terms "PIC18FXXXX" and "PIC18LFXXXX" are used throughout this section to refer to the PIC18F2221/2321/4221/4321 and PIC18LF2221/2321/4221/4321 families of devices specifically and only those devices.

TABLE 27-5:   TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC

AC CHARACTERISTICS	**Standard Operating Conditions (unless otherwise stated)**
	Operating temperature $-40°C \leq T_A \leq +85°C$ for industrial
	$-40°C \leq T_A \leq +125°C$ for extended
	Operating voltage V_{DD} range as described in DC spec **Section 27.1** and **Section 27.3**.
	LF parts operate for industrial temperatures only.

FIGURE 27-5:   LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

Load Condition 1

$V_{DD}/2$

R_L

Pin

C_L

V_{SS}

Load Condition 2

Pin

C_L

V_{SS}

R_L = 464Ω

C_L = 50 pF   for all pins except OSC2/CLKO
and including D and E outputs as ports

NOTES:

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com, click on Customer Change Notification and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://support.microchip.com