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	48MHz
Connectivity	I²C, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	14
Program Memory Size	8KB (4K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	20-DIP (0.300", 7.62mm)
Supplier Device Package	20-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf13k50-i-p

PIC18(L)F1XK50 Family Types

Device	Data Sheet Index	Program Memory		Data Memory		I/O(1)	10-bit A/D (ch)(2)	ECCP (PWM)	MSSP		EUSART	Comp.	Timers 8/16-bit	USB
		Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)				SPI	Master I²C™				
PIC18F13K50/ PIC18LF13K50	**(A)**	8K	4096	512(3)	256	15	11	1	Y	Y	1	2	1/3	Y
PIC18F14K50/ PIC18LF14K50	**(A)**	16K	8192	768(3)	256	15	11	1	Y	Y	1	2	1/3	Y

Note 1: One pin is input only.

2: Channel count includes internal Fixed Voltage Reference (FVR) and Programmable Voltage Reference (CV_{REF}) channels.

3: Includes the dual port RAM used by the USB module which is shared with the data memory.

Data Sheet Index: (Unshaded devices are described in this document)

A. DS40001350 PIC18(L)F1XK50 Data Sheet, 20-Pin USB Flash Microcontrollers with XLP Technology.

Note: For other small form-factor package availability and marking information, please visit **http://www.microchip.com/packaging** or contact your local sales office.

TABLE 1-2: PIC18(L)F1XK50 PINOUT I/O DESCRIPTIONS

Pin Name	Pin Number	Pin Type	Buffer Type	Description
RA0/D+/PGD	19			
RA0		I	TTL	Digital input
D+		I/O	XCVR	USB differential plus line (input/output)
PGD		I/O	ST	ICSP™ programming data pin
RA1/D-/PGC	18			
RA1		I	TTL	Digital input
D-		I/O	XCVR	USB differential minus line (input/output)
PGC		I/O	ST	ICSP™ programming clock pin
RA3/$\overline{\text{MCLR}}$/V_{PP}	4			Master Clear (input) or programming voltage (input)
RA3		I	ST	Digital input
$\overline{\text{MCLR}}$		I	ST	Active-low Master Clear with internal pull-up
V_{PP}		P	—	High voltage programming input
RA4/AN3/OSC2/CLKOUT	3			
RA4		I/O	TTL	Digital I/O
AN3		I	Analog	ADC channel 3
OSC2		O	XTAL	Oscillator crystal output. Connect to crystal or resonator in Crystal Oscillator mode
CLKOUT		O	CMOS	In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate
RA5/OSC1/CLKIN	2			
RA5		I/O	TTL	Digital I/O
OSC1		I	XTAL	Oscillator crystal input or external clock input ST buffer when configured in RC mode; analog other wise
CLKIN		I	CMOS	External clock source input. Always associated with the pin function OSC1 (See related OSC1/CLKIN, OSC2, CLKOUT pins
RB4/AN10/SDI/SDA	13			
RB4		I/O	TTL	Digital I/O
AN10		I	Analog	ADC channel 10
SDI		I	ST	SPI data in
SDA		I/O	ST	I^2C™ data I/O
RB5/AN11/RX/DT	12			
RB5		I/O	TLL	Digital I/O
AN11		I	Analog	ADC channel 11
RX		I	ST	EUSART asynchronous receive
DT		I/O	ST	EUSART synchronous data (see related RX/TX)
RB6/SCK/SCI	11			
RB6		I/O	TLL	Digital I/O
SCK		I/O	ST	Synchronous serial clock input/output for SPI mode
SCI		I/O	ST	Synchronous serial clock input/output for I^2C™ mode
RB7/TX/CK	10			
RB7		I/O	TLL	Digital I/O
TX		O	CMOS	EUSART asynchronous transmit
CK		I/O	ST	EUSART synchronous clock (see related RX/DT)

Legend: TTL = TTL compatible input
ST = Schmitt Trigger input
O = Output
XTAL= Crystal Oscillator
CMOS = CMOS compatible input or output
I = Input
P = Power
XCVR = USB Differential Transceiver

3.0 MEMORY ORGANIZATION

There are three types of memory in PIC18 Enhanced microcontroller devices:

- Program Memory
- Data RAM
- Data EEPROM

As Harvard architecture devices, the data and program memories use separate busses; this allows for concurrent access of the two memory spaces. The data EEPROM, for practical purposes, can be regarded as a peripheral device, since it is addressed and accessed through a set of control registers.

Additional detailed information on the operation of the Flash program memory is provided in **Section 4.0 "Flash Program Memory"**. Data EEPROM is discussed separately in **Section 5.0 "Data EEPROM Memory"**.

3.1 Program Memory Organization

PIC18 microcontrollers implement a 21-bit program counter, which is capable of addressing a 2-Mbyte program memory space. Accessing a location between the upper boundary of the physically implemented memory and the 2-Mbyte address will return all '0's (a NOP instruction).

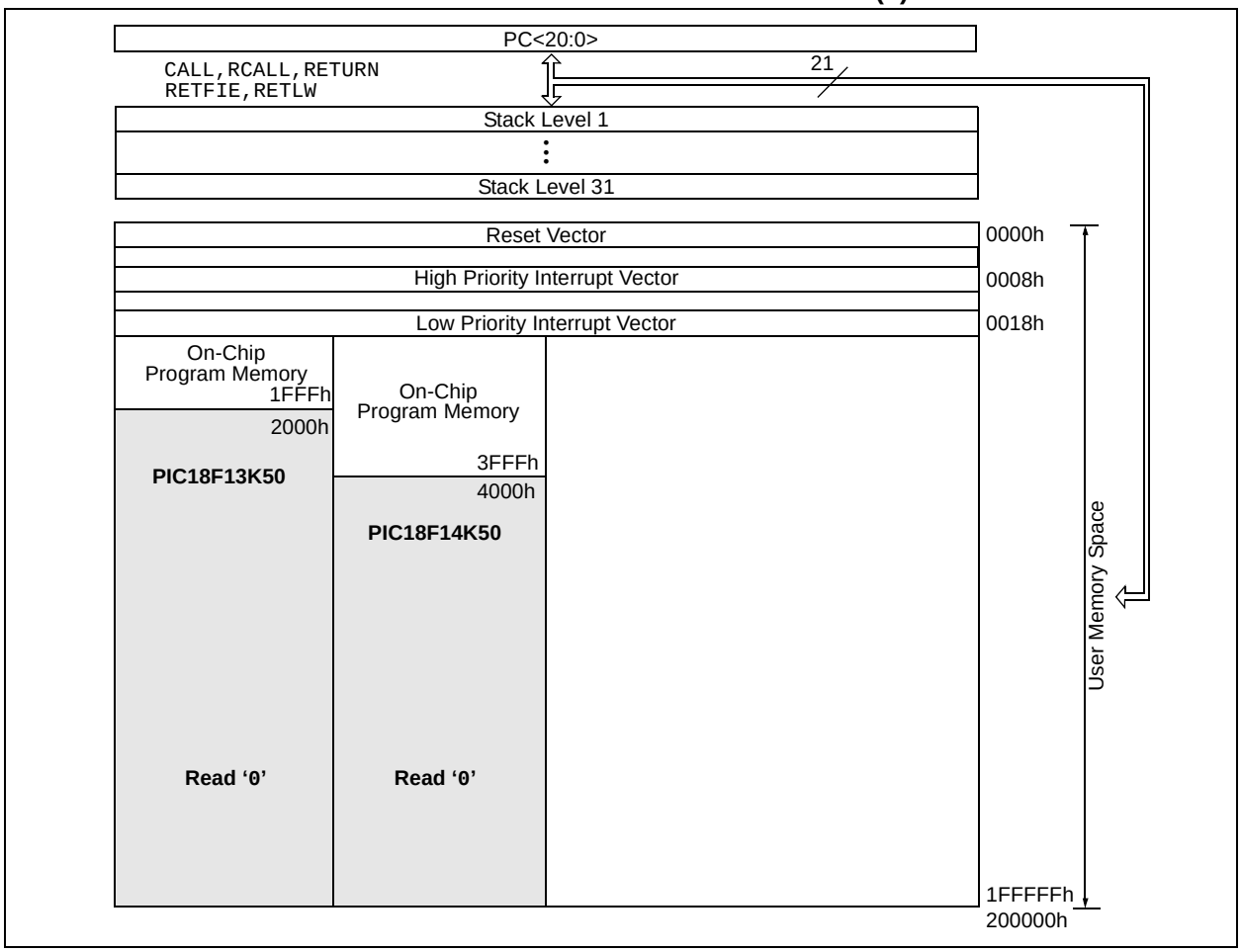
This family of devices contain the following:

- PIC18F13K50: 8 Kbytes of Flash memory, up to 4,096 single-word instructions
- PIC18F14K50: 16 Kbytes of Flash memory, up to 8,192 single-word instructions

PIC18 devices have two interrupt vectors and one Reset vector. The Reset vector address is at 0000h and the interrupt vector addresses are at 0008h and 0018h.

The program memory map for PIC18(L)F1XK50 devices is shown in Figure 3-1. Memory block details are shown in Figure 24-2.

FIGURE 3-1: PROGRAM MEMORY MAP AND STACK FOR PIC18(L)F1XK50 DEVICES



3.2.3 INSTRUCTIONS IN PROGRAM MEMORY

The program memory is addressed in bytes. Instructions are stored as either two bytes or four bytes in program memory. The Least Significant Byte (LSB) of an instruction word is always stored in a program memory location with an even address (LSb = 0). To maintain alignment with instruction boundaries, the PC increments in steps of 2 and the LSb will always read '0' (see **Section 3.1.1 "Program Counter"**).

Figure 3-4 shows an example of how instruction words are stored in the program memory.

The CALL and GOTO instructions have the absolute program memory address embedded into the instruction. Since instructions are always stored on word boundaries, the data contained in the instruction is a word address. The word address is written to PC<20:1>, which accesses the desired byte address in program memory. Instruction #2 in Figure 3-4 shows how the instruction GOTO 0006h is encoded in the program memory. Program branch instructions, which encode a relative address offset, operate in the same manner. The offset value stored in a branch instruction represents the number of single-word instructions that the PC will be offset by. **Section 25.0 "Instruction Set Summary"** provides further details of the instruction set.

FIGURE 3-4: INSTRUCTIONS IN PROGRAM MEMORY

			LSB = 1	LSB = 0	Word Address ↓
Program Memory Byte Locations →					000000h
					000002h
					000004h
					000006h
Instruction 1:	MOVLW	055h	0Fh	55h	000008h
Instruction 2:	GOTO	0006h	EFh	03h	00000Ah
			F0h	00h	00000Ch
Instruction 3:	MOVFF	123h, 456h	C1h	23h	00000Eh
			F4h	56h	000010h
					000012h
					000014h

3.2.4 TWO-WORD INSTRUCTIONS

The standard PIC18 instruction set has four two-word instructions: CALL, MOVFF, GOTO and LSFR. In all cases, the second word of the instruction always has '1111' as its four Most Significant bits (MSb); the other 12 bits are literal data, usually a data memory address.

The use of '1111' in the 4 MSbs of an instruction specifies a special form of NOP. If the instruction is executed in proper sequence – immediately after the first word – the data in the second word is accessed and used by the instruction sequence. If the first word is skipped for some reason and the second word is executed by itself, a NOP is executed instead. This is necessary for cases when the two-word instruction is preceded by a conditional instruction that changes the PC. Example 3-4 shows how this works.

Note:	See **Section 3.6 "PIC18 Instruction Execution and the Extended Instruction Set"** for information on two-word instructions in the extended instruction set.

EXAMPLE 3-4: TWO-WORD INSTRUCTIONS

CASE 1:			
Object Code	**Source Code**		
0110 0110 0000 0000	TSTFSZ	REG1	; is RAM location 0?
1100 0001 0010 0011	MOVFF	REG1, REG2	; No, skip this word
1111 0100 0101 0110			; Execute this word as a NOP
0010 0100 0000 0000	ADDWF	REG3	; continue code
CASE 2:			
Object Code	**Source Code**		
0110 0110 0000 0000	TSTFSZ	REG1	; is RAM location 0?
1100 0001 0010 0011	MOVFF	REG1, REG2	; Yes, execute this word
1111 0100 0101 0110			; 2nd word of instruction
0010 0100 0000 0000	ADDWF	REG3	; continue code

TABLE 3-2: REGISTER FILE SUMMARY (PIC18(L)F1XK50) (CONTINUED)

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Details on page:
SPBRGH	EUSART Baud Rate Generator Register, High Byte								0000 0000	277, 175
SPBRG	EUSART Baud Rate Generator Register, Low Byte								0000 0000	277, 175
RCREG	EUSART Receive Register								0000 0000	277, 176
TXREG	EUSART Transmit Register								0000 0000	277, 175
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	277, 184
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	277, 185
EEADR	EEADR7	EEADR6	EEADR5	EEADR4	EEADR3	EEADR2	EEADR1	EEADR0	0000 0000	277, 50, 58
EEDATA	EEPROM Data Register								0000 0000	277, 50, 58
EECON2	EEPROM Control Register 2 (not a physical register)								0000 0000	277, 50, 58
EECON1	EEPGD	CFGS	—	FREE	WRERR	WREN	WR	RD	xx-0 x000	277, 51, 58
IPR2	OSCFIP	C1IP	C2IP	EEIP	BCLIP	USBIP	TMR3IP	–	1111 111-	278, 74
PIR2	OSCFIF	C1IF	C2IF	EEIF	BCLIF	USBIF	TMR3IF	–	0000 000-	278, 70
PIE2	OSCFIE	C1IE	C2IE	EEIE	BCLIE	USBIE	TMR3IE	–	0000 000-	278, 72
IPR1	–	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	-111 1111	278, 73
PIR1	–	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	278, 69
PIE1	–	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	278, 71
OSCTUNE	INTSRC	SPLLEN	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	0000 0000	20, 278
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	278, 89
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	–	–	–	–	1111 ----	278, 84
TRISA	–	–	TRISA5	TRISA4	–	–	–	–	--11 ----	278, 78
LATC	LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	xxxx xxxx	278, 89
LATB	LATB7	LATB6	LATB5	LATB4	–	–	–	–	xxxx ----	278, 84
LATA	–	–	LATA5	LATA4	–	–	–	–	--xx ----	278, 78
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	278, 89
PORTB	RB7	RB6	RB5	RB4	–	–	–	–	xxxx ----	278, 84
PORTA	–	–	RA5	RA4	RA3(2)	–	RA1(3)	RA0(3)	--xx x-xx	278, 78
ANSELH	—	—	—	—	ANS11	ANS10	ANS9	ANS8	---- 1111	278, 94
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	—	—	—	1111 1---	278, 93
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	—	—	—	—	0000 ----	278, 84
IOCA	—	—	IOCA5	IOCA4	IOCA3	—	IOCA1	IOCA0	--00 0-00	278, 78
WPUB	WPUB7	WPUB6	WPUB5	WPUB4	—	—	—	—	1111 ----	278, 84
WPUA	—	—	WPUA5	WPUA4	WPUA3	—	—	—	--11 1---	275, 84
SLRCON	—	—	—	—	—	SLRC	SLRB	SLRA	---- -111	278, 95
SSPMSK	MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0	1111 1111	278, 154
CM1CON0	C1ON	C1OUT	C1OE	C1POL	C1SP	C1R	C1CH1	C1CH0	0000 1000	278, 221
CM2CON1	MC1OUT	MC2OUT	C1RSEL	C2RSEL	C1HYS	C2HYS	C1SYNC	C2SYNC	0000 0000	278, 222
CM2CON0	C2ON	C2OUT	C2OE	C2POL	C2SP	C2R	C2CH1	C2CH0	0000 1000	278, 222
SRCON1	SRSPE	SRSCKE	SRSC2E	SRSC1E	SRRPE	SRRCKE	SRRC2E	SRRC1E	0000 0000	278, 235
SRCON0	SRLEN	SRCLK2	SRCLK1	SRCLK0	SRQEN	SRNQEN	SRPS	SRPR	0000 0000	278, 234
UCON	—	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	—	-0x0 000-	278, 242

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

Note 1: The SBOREN bit is only available when the BOREN<1:0> Configuration bits = 01; otherwise it is disabled and reads as '0'. See **Section 23.4 "Brown-out Reset (BOR)"**.

2: The RA3 bit is only available when Master Clear Reset is disabled (MCLRE Configuration bit = 0). Otherwise, RA3 reads as '0'. This bit is read-only.

3: Bits RA0 and RA1 are available only when USB is disabled.

Operations on the FSRs with POSTDEC, POSTINC and PREINC affect the entire register pair; that is, roll-overs of the FSRnL register from FFh to 00h carry over to the FSRnH register. On the other hand, results of these operations do not change the value of any flags in the STATUS register (e.g., Z, N, OV, etc.).

The PLUSW register can be used to implement a form of indexed addressing in the data memory space. By manipulating the value in the W register, users can reach addresses that are fixed offsets from pointer addresses. In some applications, this can be used to implement some powerful program control structure, such as software stacks, inside of data memory.

3.4.3.3 Operations by FSRs on FSRs

Indirect addressing operations that target other FSRs or virtual registers represent special cases. For example, using an FSR to point to one of the virtual registers will not result in successful operations. As a specific case, assume that FSR0H:FSR0L contains FE7h, the address of INDF1. Attempts to read the value of the INDF1 using INDF0 as an operand will return 00h. Attempts to write to INDF1 using INDF0 as the operand will result in a NOP.

On the other hand, using the virtual registers to write to an FSR pair may not occur as planned. In these cases, the value will be written to the FSR pair but without any incrementing or decrementing. Thus, writing to either the INDF2 or POSTDEC2 register will write the same value to the FSR2H:FSR2L.

Since the FSRs are physical registers mapped in the SFR space, they can be manipulated through all direct operations. Users should proceed cautiously when working on these registers, particularly if their code uses indirect addressing.

Similarly, operations by indirect addressing are generally permitted on all other SFRs. Users should exercise the appropriate caution that they do not inadvertently change settings that might affect the operation of the device.

3.5 Data Memory and the Extended Instruction Set

Enabling the PIC18 extended instruction set (XINST Configuration bit = 1) significantly changes certain aspects of data memory and its addressing. Specifically, the use of the Access Bank for many of the core PIC18 instructions is different; this is due to the introduction of a new addressing mode for the data memory space.

What does not change is just as important. The size of the data memory space is unchanged, as well as its linear addressing. The SFR map remains the same. Core PIC18 instructions can still operate in both Direct and Indirect Addressing mode; inherent and literal instructions do not change at all. Indirect addressing with FSR0 and FSR1 also remain unchanged.

3.5.1 INDEXED ADDRESSING WITH LITERAL OFFSET

Enabling the PIC18 extended instruction set changes the behavior of indirect addressing using the FSR2 register pair within Access RAM. Under the proper conditions, instructions that use the Access Bank – that is, most bit-oriented and byte-oriented instructions – can invoke a form of indexed addressing using an offset specified in the instruction. This special addressing mode is known as Indexed Addressing with Literal Offset, or Indexed Literal Offset mode.

When using the extended instruction set, this addressing mode requires the following:

- The use of the Access Bank is forced (‘a’ = 0) and
- The file address argument is less than or equal to 5Fh.

Under these conditions, the file address of the instruction is not interpreted as the lower byte of an address (used with the BSR in direct addressing), or as an 8-bit address in the Access Bank. Instead, the value is interpreted as an offset value to an Address Pointer, specified by FSR2. The offset and the contents of FSR2 are added to obtain the target address of the operation.

3.5.2 INSTRUCTIONS AFFECTED BY INDEXED LITERAL OFFSET MODE

Any of the core PIC18 instructions that can use direct addressing are potentially affected by the Indexed Literal Offset Addressing mode. This includes all byte-oriented and bit-oriented instructions, or almost one-half of the standard PIC18 instruction set. Instructions that only use Inherent or Literal Addressing modes are unaffected.

Additionally, byte-oriented and bit-oriented instructions are not affected if they do not use the Access Bank (Access RAM bit is ‘1’), or include a file address of 60h or above. Instructions meeting these criteria will continue to execute as before. A comparison of the different possible addressing modes when the extended instruction set is enabled is shown in Figure 3-9.

Those who desire to use byte-oriented or bit-oriented instructions in the Indexed Literal Offset mode should note the changes to assembler syntax for this mode. This is described in more detail in **Section 25.2.1 “Extended Instruction Syntax”**.

7.4 INTCON Registers

The INTCON registers are readable and writable registers, which contain various enable, priority and flag bits.

Note:	Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global 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 7-1: INTCON: INTERRUPT CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RABIE	TMR0IF	INT0IF	RABIF
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 7 **GIE/GIEH:** Global Interrupt Enable bit
When IPEN = 0:
1 = Enables all unmasked interrupts
0 = Disables all interrupts including peripherals
When IPEN = 1:
1 = Enables all high priority interrupts
0 = Disables all interrupts including low priority.

bit 6 **PEIE/GIEL:** Peripheral Interrupt Enable bit
When IPEN = 0:
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts
When IPEN = 1:
1 = Enables all low priority interrupts
0 = Disables all low priority interrupts

bit 5 **TMR0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 overflow interrupt
0 = Disables the TMR0 overflow interrupt

bit 4 **INT0IE:** INT0 External Interrupt Enable bit
1 = Enables the INT0 external interrupt
0 = Disables the INT0 external interrupt

bit 3 **RABIE:** RA and RB Port Change Interrupt Enable bit(2)
1 = Enables the RA and RB port change interrupt
0 = Disables the RA and RB port change interrupt

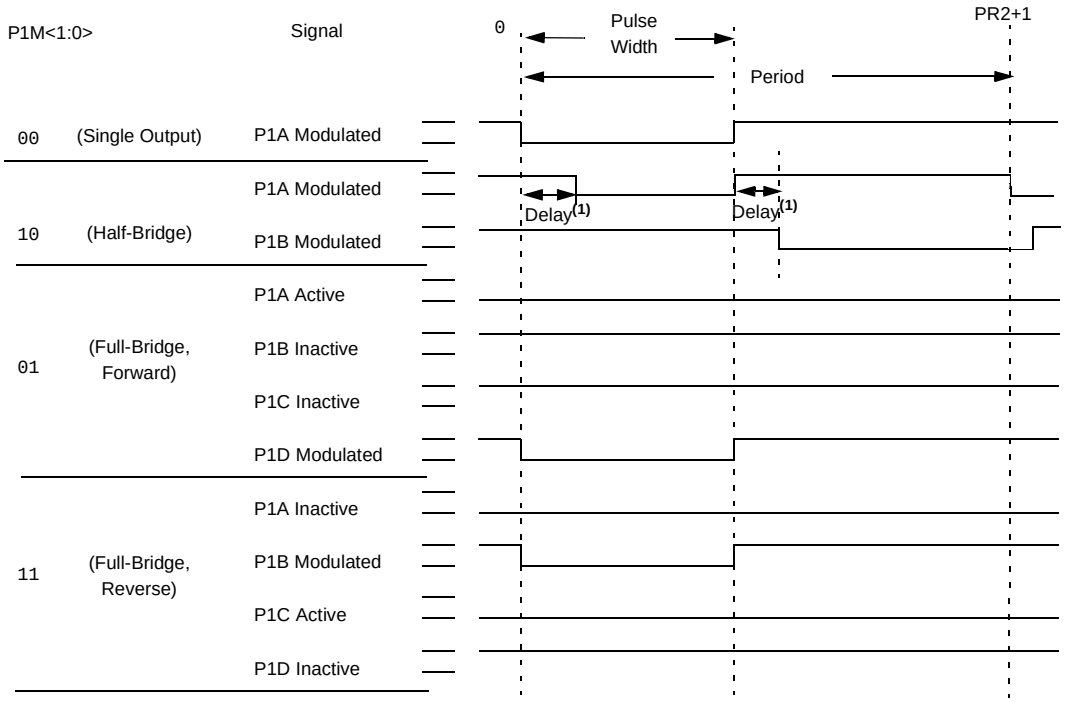
bit 2 **TMR0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared by software)
0 = TMR0 register did not overflow

bit 1 **INT0IF:** INT0 External Interrupt Flag bit
1 = The INT0 external interrupt occurred (must be cleared by software)
0 = The INT0 external interrupt did not occur

bit 0 **RABIF:** RA and RB Port Change Interrupt Flag bit(1)
1 = At least one of the RA <5:3> or RB<7:4> pins changed state (must be cleared by software)
0 = None of the RA<5:3> or RB<7:4> pins have changed state

Note 1: A mismatch condition will continue to set the RABIF bit. Reading PORTA and PORTB will end the mismatch condition and allow the bit to be cleared.
2: RA and RB port change interrupts also require the individual pin IOCA and IOCB enable.

FIGURE 14-5: EXAMPLE ENHANCED PWM OUTPUT RELATIONSHIPS (ACTIVE-LOW STATE)



Relationships:

- Period = 4 * Tosc * (PR2 + 1) * (TMR2 Prescale Value)
- Pulse Width = Tosc * (CCPR1L<7:0>:CCP1CON<5:4>) * (TMR2 Prescale Value)
- Delay = 4 * Tosc * (PWM1CON<6:0>)

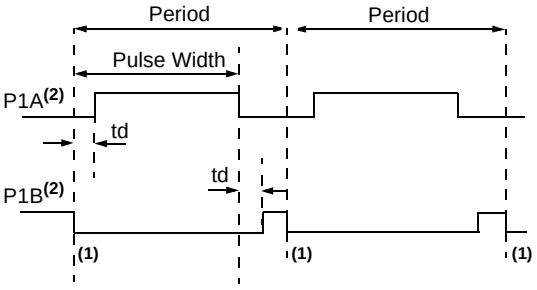
Note 1: Dead-band delay is programmed using the PWM1CON register (**Section 14.4.6 "Programmable Dead-Band Delay mode"**).

14.4.6 PROGRAMMABLE DEAD-BAND DELAY MODE

In Half-Bridge applications where all power switches are modulated at the PWM frequency, the power switches normally require more time to turn off than to turn on. If both the upper and lower power switches are switched at the same time (one turned on, and the other turned off), both switches may be on for a short period of time until one switch completely turns off. During this brief interval, a very high current (shoot-through current) will flow through both power switches, shorting the bridge supply. To avoid this potentially destructive shoot-through current from flowing during switching, turning on either of the power switches is normally delayed to allow the other switch to completely turn off.

In Half-Bridge mode, a digitally programmable dead-band delay is available to avoid shoot-through current from destroying the bridge power switches. The delay occurs at the signal transition from the non-active state to the active state. See Figure 14-14 for illustration. The lower seven bits of the associated PWM1CON register (Register 14-3) sets the delay period in terms of microcontroller instruction cycles (T_{CY} or 4 T_{OSC}).

FIGURE 14-14: EXAMPLE OF 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 14-15: EXAMPLE OF HALF-BRIDGE APPLICATIONS

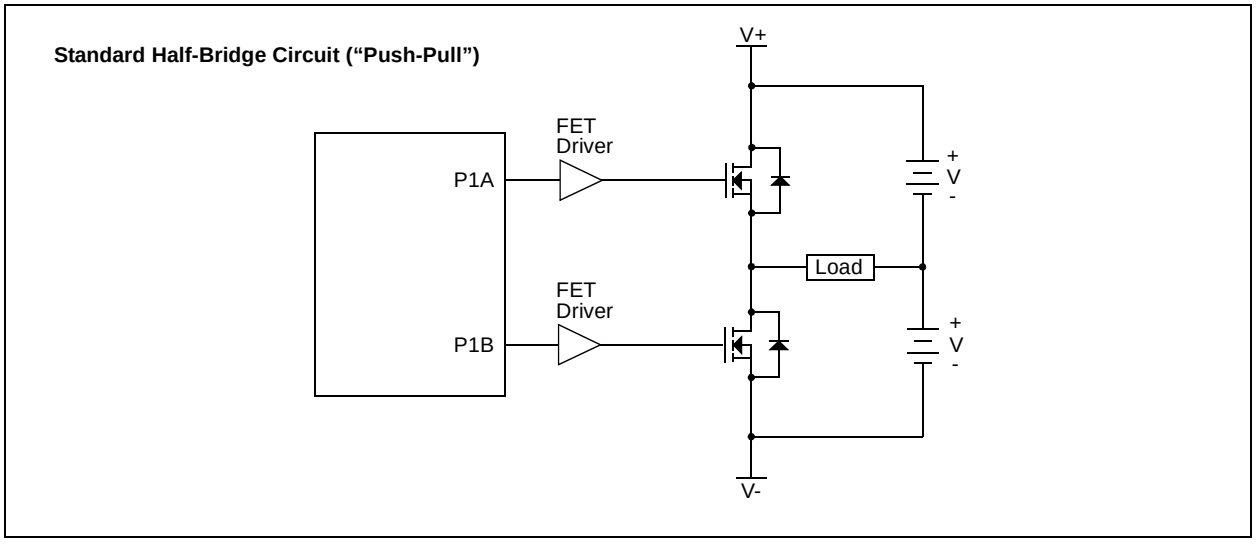
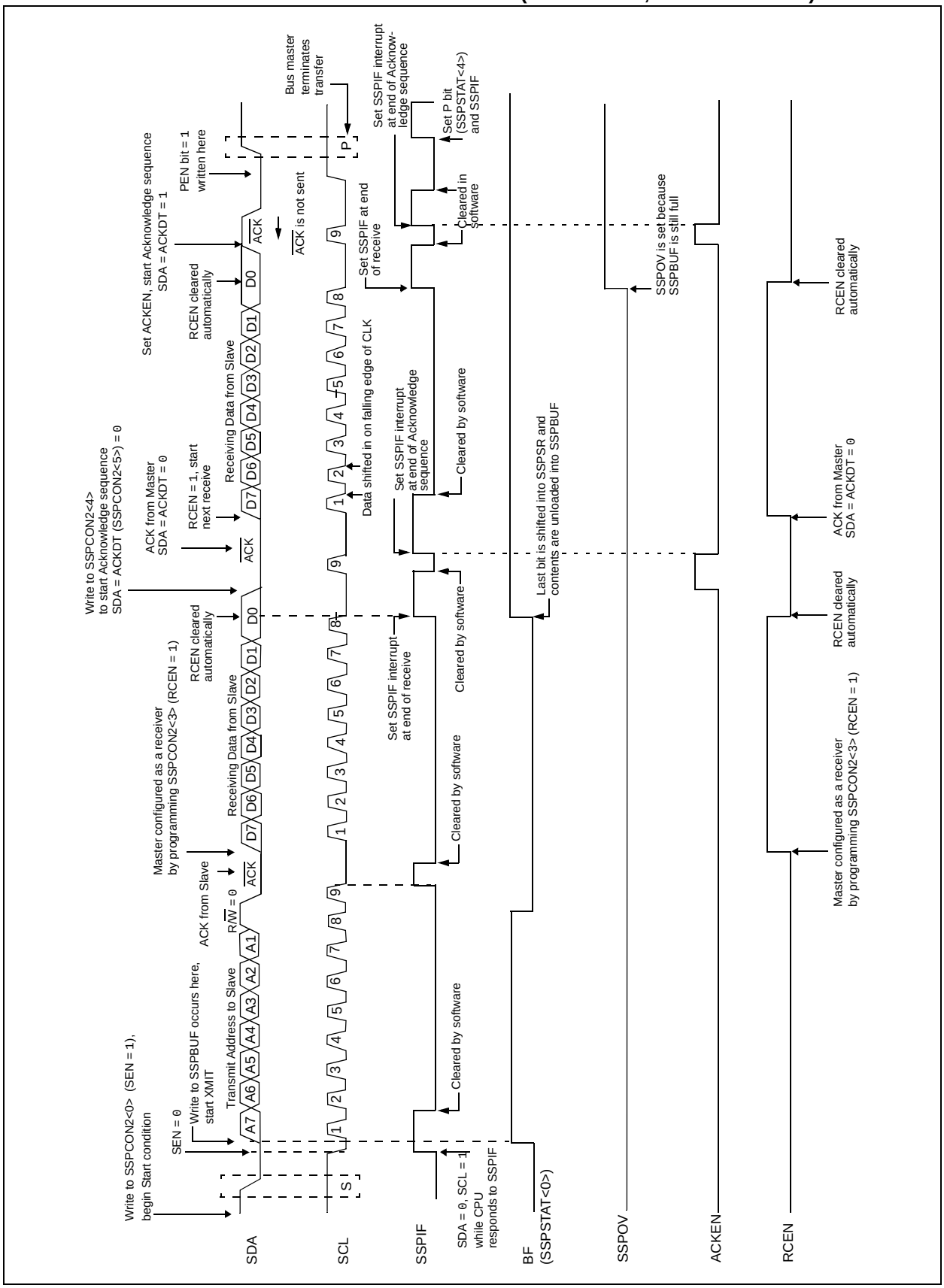


FIGURE 15-22: I²C™ MASTER MODE WAVEFORM (RECEPTION, 7-BIT ADDRESS)



REGISTER 18-3: CM2CON1: COMPARATOR 2 CONTROL REGISTER 1

R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MC1OUT	MC2OUT	C1RSEL	C2RSEL	C1HYS	C2HYS	C1SYNC	C2SYNC
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 7 **MC1OUT:** Mirror Copy of C1OUT bit

bit 6 **MC2OUT:** Mirror Copy of C2OUT bit

bit 5 **C1RSEL:** Comparator C1 Reference Select bit
1 = FVR routed to C1VREF input
0 = CVREF routed to C1VREF input

bit 4 **C2RSEL:** Comparator C2 Reference Select bit
1 = FVR routed to C2VREF input
0 = CVREF routed to C2VREF input

bit 3 **C1HYS:** Comparator C1 Hysteresis Enable bit
1 = Comparator C1 hysteresis enabled
0 = Comparator C1 hysteresis disabled

bit 2 **C2HYS:** Comparator C2 Hysteresis Enable bit
1 = Comparator C2 hysteresis enabled
0 = Comparator C2 hysteresis disabled

bit 1 **C1SYNC:** C1 Output Synchronous Mode bit
1 = C1 output is synchronous to rising edge to TMR1 clock
0 = C1 output is asynchronous

bit 0 **C2SYNC:** C2 Output Synchronous Mode bit
1 = C2 output is synchronous to rising edge to TMR1 clock
0 = C2 output is asynchronous

20.0 SR LATCH

The module consists of a single SR Latch with multiple Set and Reset inputs as well as selectable latch output. The SR Latch module includes the following features:

- Programmable input selection
- SR Latch output is available internally/externally
- Selectable Q and $\overline{Q}$ output
- Firmware Set and Reset

20.1 Latch Operation

The latch is a Set-Reset latch that does not depend on a clock source. Each of the Set and Reset inputs are active-high. The latch can be Set or Reset by CxOUT, INT1 pin, or variable clock. Additionally the SRPS and the SRPR bits of the SRCON0 register may be used to Set or Reset the SR Latch, respectively. The latch is reset-dominant, therefore, if both Set and Reset inputs are high the latch will go to the Reset state. Both the SRPS and SRPR bits are self resetting which means that a single write to either of the bits is all that is necessary to complete a latch Set or Reset operation.

20.2 Latch Output

The SRQEN and SRNQEN bits of the SRCON0 register control the latch output selection. Only one of the SR latch's outputs may be directly output to an I/O pin at a time. Priority is determined by the state of bits SRQEN and SRNQEN in registers SRCON0.

TABLE 20-1: SR LATCH OUTPUT CONTROL

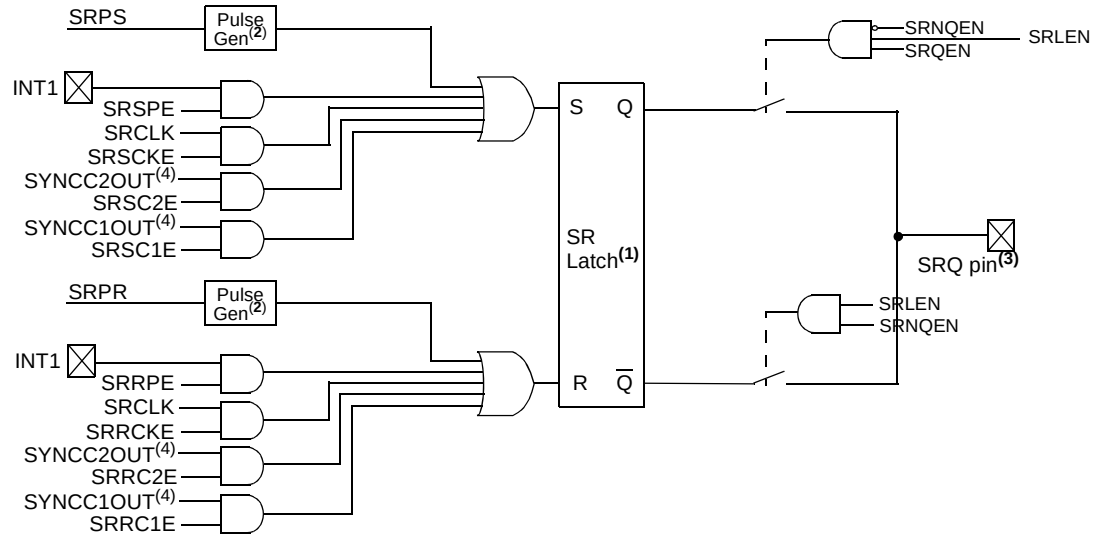
SRLEN	SRQEN	SRNQEN	SR Latch Output to Port I/O
0	X	X	I/O
1	0	0	I/O
1	0	1	$\overline{Q}$
1	1	0	Q
1	1	1	$\overline{Q}$

The applicable TRIS bit of the corresponding port must be cleared to enable the port pin output driver.

20.3 Effects of a Reset

Upon any device Reset, the SR latch is not initialized. The user's firmware is responsible to initialize the latch output before enabling it to the output pins.

FIGURE 20-1: SR LATCH SIMPLIFIED BLOCK DIAGRAM



Note 1: If R = 1 and S = 1 simultaneously, Q = 0, $\overline{Q}$ = 1
2: Pulse generator causes a 2 Q-state pulse width.
3: Output shown for reference only. See I/O port pin block diagram for more detail.
4: Name denotes the source of connection at the comparator output.

TABLE 24-3: SUMMARY OF CODE PROTECTION REGISTERS

File Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
300008h	CONFIG5L	—	—	—	—	—	—	CP1	CP0
300009h	CONFIG5H	CPD	CPB	—	—	—	—	—	—
30000Ah	CONFIG6L	—	—	—	—	—	—	WRT1	WRT0
30000Bh	CONFIG6H	WRTD	WRTB	WRTC	—	—	—	—	—
30000Ch	CONFIG7L	—	—	—	—	—	—	EBTR1	EBTR0
30000Dh	CONFIG7H	—	EBTRB	—	—	—	—	—	—

Legend: Shaded cells are unimplemented.

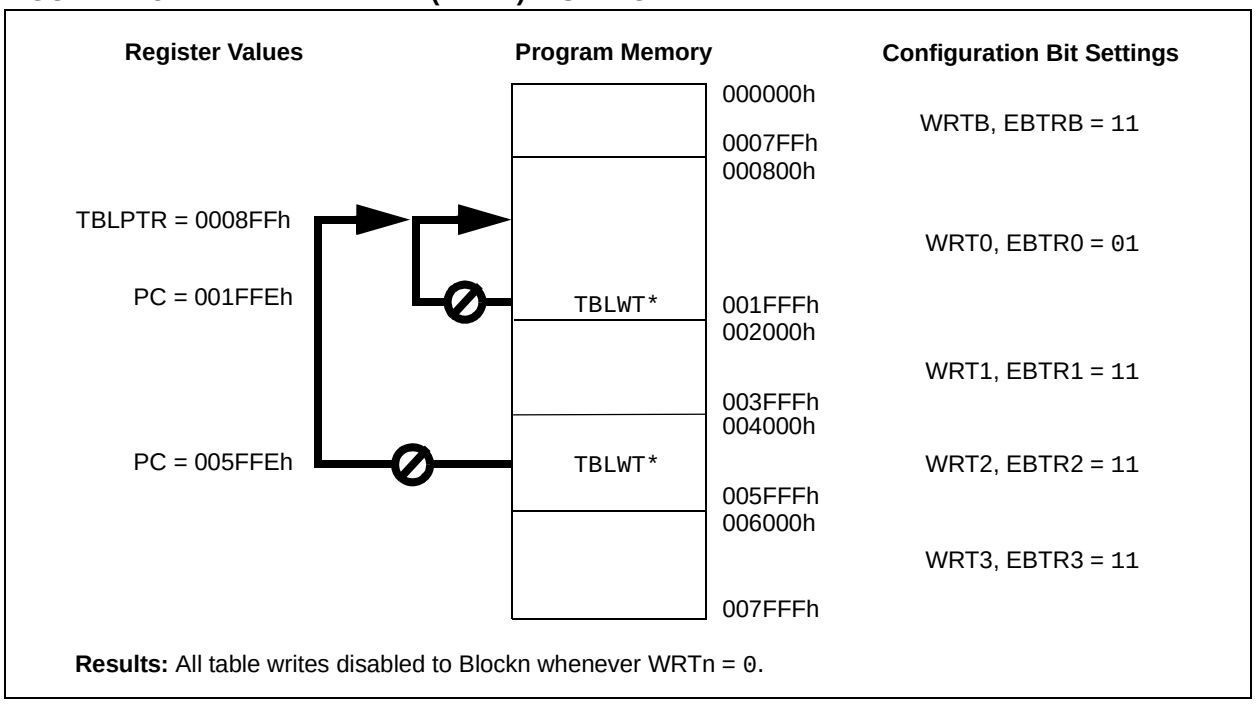
24.3.1 PROGRAM MEMORY CODE PROTECTION

The program memory may be read to or written from any location using the table read and table write instructions. The device ID may be read with table reads. The Configuration registers may be read and written with the table read and table write instructions.

In normal execution mode, the CPn bits have no direct effect. CPn bits inhibit external reads and writes. A block of user memory may be protected from table writes if the WRTn Configuration bit is '0'. The EBTRn bits control table reads. For a block of user memory with the EBTRn bit cleared to '0', a table READ instruction that executes from within that block is allowed to read. A table read instruction that executes from a location outside of that block is not allowed to read and will result in reading '0's. Figures 24-3 through 24-5 illustrate table write and table read protection.

Note:	Code protection bits may only be written to a '0' from a '1' state. It is not possible to write a '1' to a bit in the '0' state. Code protection bits are only set to '1' by a full chip erase or block erase function. The full chip erase and block erase functions can only be initiated via ICSP or an external programmer.

FIGURE 24-3: TABLE WRITE (WRTn) DISALLOWED



Results: All table writes disabled to Blockn whenever WRTn = 0.

BTG **Bit Toggle f**

Syntax: BTG f, b {,a}

Operands: $0 \le f \le 255$
$0 \le b < 7$
$a \in [0,1]$

Operation: $(\overline{f<b>}) \rightarrow f<b>$

Status Affected: None

Encoding:

0111	bbba	ffff	ffff

Description: Bit 'b' in data memory location 'f' is inverted.
If 'a' is '0', the Access Bank is selected.
If 'a' is '1', the BSR is used to select the GPR bank (default).
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See **Section 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register 'f'

Example: `BTG PORTC, 4, 0`

Before Instruction:
PORTC = 0111 0101 [75h]
After Instruction:
PORTC = 0110 0101 [65h]

BOV **Branch if Overflow**

Syntax: BOV n

Operands: $-128 \le n \le 127$

Operation: if OVERFLOW bit is '1'
$(PC) + 2 + 2n \rightarrow PC$

Status Affected: None

Encoding:

1110	0100	nnnn	nnnn

Description: If the OVERFLOW bit is '1', then the program will branch.
The 2's complement number '2n' is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a 2-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:
If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example: `HERE BOV Jump`

Before Instruction
PC = address (HERE)
After Instruction
If OVERFLOW = 1;
PC = address (Jump)
If OVERFLOW = 0;
PC = address (HERE + 2)

DAW **Decimal Adjust W Register**

Syntax: DAW

Operands: None

Operation: If [W<3:0> > 9] or [DC = 1] then
(W<3:0>) + 6 → W<3:0>;
else
(W<3:0>) → W<3:0>;

If [W<7:4> + DC > 9] or [C = 1] then
(W<7:4>) + 6 + DC → W<7:4> ;
else
(W<7:4>) + DC → W<7:4>

Status Affected: C

Encoding:

0000	0000	0000	0111

Description: DAW adjusts the 8-bit value in W, resulting from the earlier addition of two variables (each in packed BCD format) and produces a correct packed BCD result.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register W	Process Data	Write W

Example1:

```
DAW
```

Before Instruction
W = A5h
C = 0
DC = 0
After Instruction
W = 05h
C = 1
DC = 0

Example 2:

Before Instruction
W = CEh
C = 0
DC = 0
After Instruction
W = 34h
C = 1
DC = 0

DECF **Decrement f**

Syntax: DECF f {,d {,a}}

Operands: $0 \le f \le 255$
$d \in [0,1]$
$a \in [0,1]$

Operation: (f) – 1 → dest

Status Affected: C, DC, N, OV, Z

Encoding:

0000	01da	ffff	ffff

Description: Decrement register ‘f’. If ‘d’ is ‘0’, the result is stored in W. If ‘d’ is ‘1’, the result is stored back in register ‘f’ (default).
If ‘a’ is ‘0’, the Access Bank is selected.
If ‘a’ is ‘1’, the BSR is used to select the GPR bank (default).
If ‘a’ is ‘0’ and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 95 (5Fh). See **Section 25.2.3 “Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode”** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register ‘f’	Process Data	Write to destination

Example: DECF CNT, 1, 0

Before Instruction
CNT = 01h
Z = 0
After Instruction
CNT = 00h
Z = 1

LFSR **Load FSR**

Syntax: LFSR f, k

Operands: $0 \le f \le 2$
$0 \le k \le 4095$

Operation: $k \rightarrow FSRf$

Status Affected: None

Encoding:

1110 1111	1110 0000	00ff $k_7$kkk	k_{11}kkk kkkk

Description: The 12-bit literal 'k' is loaded into the File Select Register pointed to by 'f'.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k' MSB	Process Data	Write literal 'k' MSB to FSRfH
Decode	Read literal 'k' LSB	Process Data	Write literal 'k' to FSRfL

Example: `LFSR 2, 3ABh`

After Instruction
FSR2H = 03h
FSR2L = ABh

MOVF **Move f**

Syntax: MOVF f {,d {,a}}

Operands: $0 \le f \le 255$
$d \in [0,1]$
$a \in [0,1]$

Operation: $f \rightarrow dest$

Status Affected: N, Z

Encoding:

0101	00da	ffff	ffff

Description: The contents of register 'f' are moved to a destination dependent upon the status of 'd'. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed back in register 'f' (default). Location 'f' can be anywhere in the 256-byte bank.
If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default).
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See **Section 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write W

Example: `MOVF REG, 0, 0`

Before Instruction
REG = 22h
W = FFh
After Instruction
REG = 22h
W = 22h

TABLE 27-6: USB MODULE SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
D313	VUSB	USB Voltage	3.0	—	3.6	V	Voltage on VUSB pin must be in this range for proper USB operation
D314	IIL	Input Leakage on pin	—	—	± 1	µA	VSS ≤ VPIN ≤ VDD pin at high impedance
D315	VILUSB	Input Low Voltage for USB Buffer	—	—	0.8	V	For VUSB range
D316	VIHUSB	Input High Voltage for USB Buffer	2.0	—	—	V	For VUSB range
D318	VDIFS	Differential Input Sensitivity	—	—	0.2	V	The difference between D+ and D- must exceed this value while VCM is met
D319	VCM	Differential Common Mode Range	0.8	—	2.5	V	
D320	ZOUT	Driver Output Impedance(1)	28	—	44	Ω	
D321	VOL	Voltage Output Low	0.0	—	0.3	V	1.5 kΩ load connected to 3.6V
D322	VOH	Voltage Output High	2.8	—	3.6	V	1.5 kΩ load connected to ground

Note 1: The D+ and D- signal lines have been built-in impedance matching resistors. No external resistors, capacitors or magnetic components are necessary on the D+/D- signal paths between the PIC18(L)F1XK50 family device and USB cable.

TABLE 27-7: THERMAL CONSIDERATIONS

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C					
Param. No.	**Sym.**	**Characteristic**	**Typ.**	**Units**	**Conditions**
TH01	θJA	Thermal Resistance Junction to Ambient	62.2	°C/W	20-pin PDIP package
			77.7	°C/W	20-pin SOIC package
			87.3	°C/W	20-pin SSOP package
			36.1	°C/W	20-pin QFN 5x5mm package
TH02	θJC	Thermal Resistance Junction to Case	27.5	°C/W	20-pin PDIP package
			23.1	°C/W	20-pin SOIC package
			31.1	°C/W	20-pin SSOP package
			12.2	°C/W	20-pin QFN 5x5mm package
TH03	TJMAX	Maximum Junction Temperature	150	°C	
TH04	PD	Power Dissipation	—	W	PD = PINTERNAL + PI/O
TH05	PINTERNAL	Internal Power Dissipation	—	W	PINTERNAL = IDD x VDD(1)
TH06	PI/O	I/O Power Dissipation	—	W	PI/O = Σ (IOL * VOL) + Σ (IOH * (VDD - VOH))
TH07	PDER	Derated Power	—	W	PDER = PDMAX (TJ - TA)/θJA(2)

Note 1: IDD is current to run the chip alone without driving any load on the output pins.

2: TA = Ambient Temperature, TJ = Junction Temperature

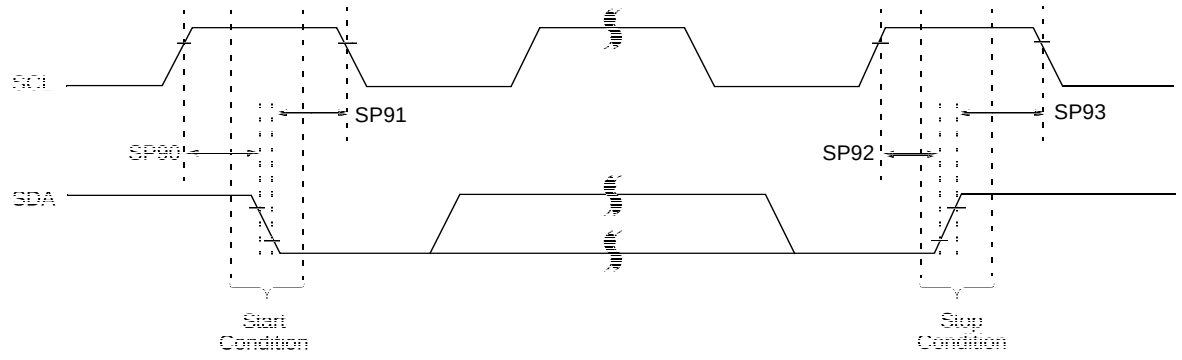
TABLE 27-22: SPI MODE REQUIREMENTS

Param. No.	Symbol	Characteristic		Min.	Typ.†	Max.	Units	Conditions
SP70*	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ to SCK$\downarrow$ or SCK$\uparrow$ input		Tcy	—	—	ns	
SP71*	TscH	SCK input high time (Slave mode)		Tcy + 20	—	—	ns	
SP72*	TscL	SCK input low time (Slave mode)		Tcy + 20	—	—	ns	
SP73*	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge		100	—	—	ns	
SP74*	TscH2diL, TscL2diL	Hold time of SDI data input to SCK edge		100	—	—	ns	
SP75*	TdoR	SDO data output rise time	3.0-5.5V	—	10	25	ns	
			1.8-5.5V	—	25	50	ns	
SP76*	TdoF	SDO data output fall time		—	10	25	ns	
SP77*	TssH2doZ	$\overline{SS}\uparrow$ to SDO output high-impedance		10	—	50	ns	
SP78*	TscR	SCK output rise time (Master mode)	3.0-5.5V	—	10	25	ns	
			1.8-5.5V	—	25	50	ns	
SP79*	TscF	SCK output fall time (Master mode)		—	10	25	ns	
SP80*	TscH2doV, TscL2doV	SDO data output valid after SCK edge	3.0-5.5V	—	—	50	ns	
			1.8-5.5V	—	—	145	ns	
SP81*	TdoV2scH, TdoV2scL	SDO data output setup to SCK edge		Tcy	—	—	ns	
SP82*	TssL2doV	SDO data output valid after $\overline{SS}\downarrow$ edge		—	—	50	ns	
SP83*	TscH2ssH, TscL2ssH	$\overline{SS}\uparrow$ after SCK edge		1.5Tcy + 40	—	—	ns	

* These parameters are characterized but not tested.

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

FIGURE 27-19: I²C™ BUS START/STOP BITS TIMING



Note: Refer to Figure 27-2 for load conditions.

Unless otherwise noted, V_{IN} = 5V, F_{OSC} = 300 kHz, C_{IN} = 0.1 µF, T_A = 25°C.

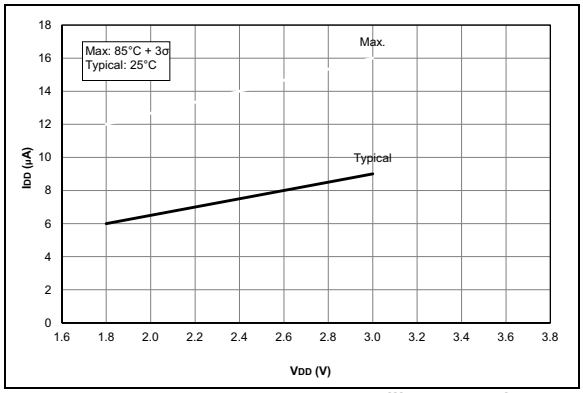


FIGURE 28-1: *IDD, LP Oscillator Mode, FOSC = 32 kHz, PIC18LF1XK50 Only.*

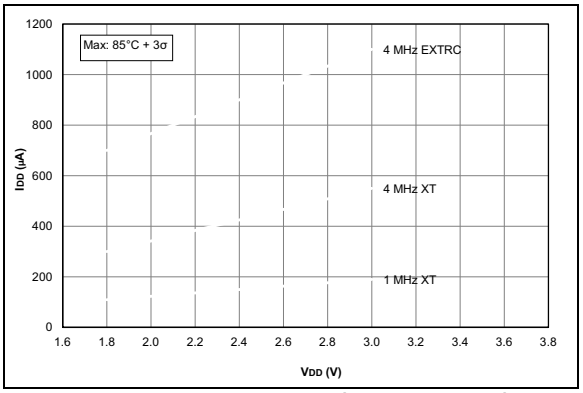


FIGURE 28-4: *IDD Maximum, XT and EXTRC Oscillator, PIC18LF1XK50 Only.*

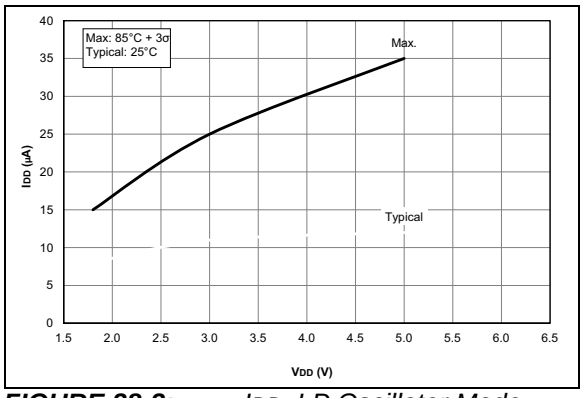


FIGURE 28-2: *IDD, LP Oscillator Mode, FOSC = 32 kHz, PIC18F1XK50 Only.*

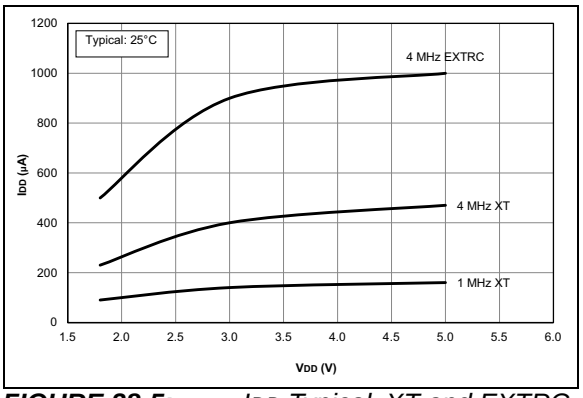


FIGURE 28-5: *IDD Typical, XT and EXTRC Oscillator, PIC18F1XK50 Only.*

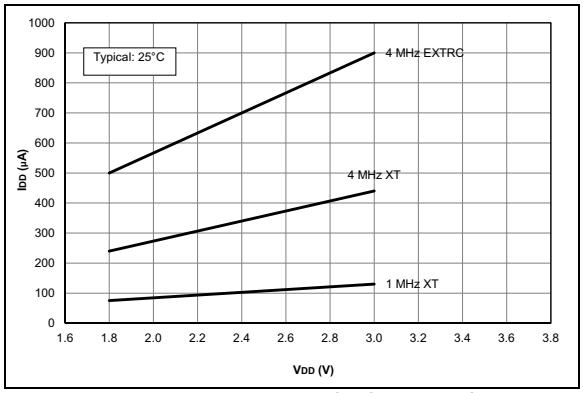


FIGURE 28-3: *IDD Typical, XT and EXTRC Oscillator, PIC18LF1XK50 Only.*

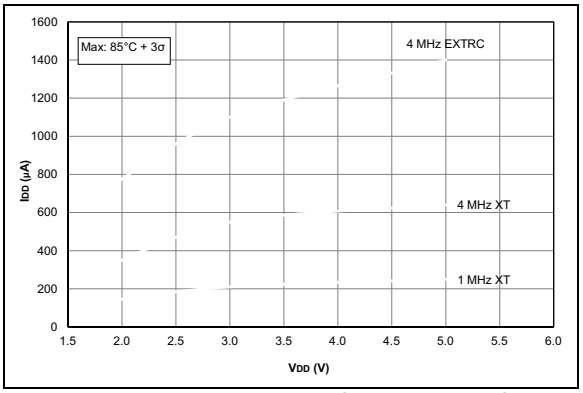


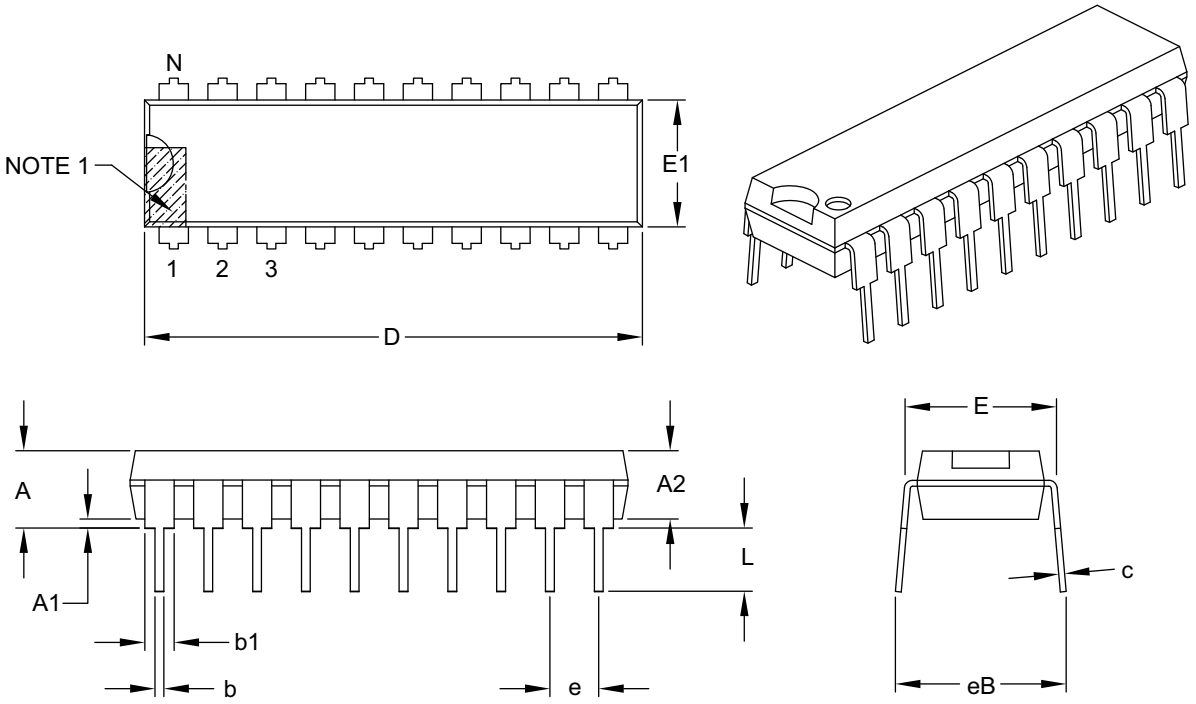
FIGURE 28-6: *IDD Maximum, XT and EXTRC Oscillator, PIC18F1XK50 Only.*

29.2 Package Details

The following sections give the technical details of the packages.

20-Lead Plastic Dual In-Line (P) – 300 mil Body [PDIP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		20	
Pitch	e		.100 BSC	
Top to Seating Plane	A	–	–	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.300	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.980	1.030	1.060
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-019B