

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	20MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	5
Program Memory Size	1.75KB (1K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	8-VDFN Exposed Pad
Supplier Device Package	8-DFN (3x3)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12lf1501-e-mf

PIC12(L)F1501

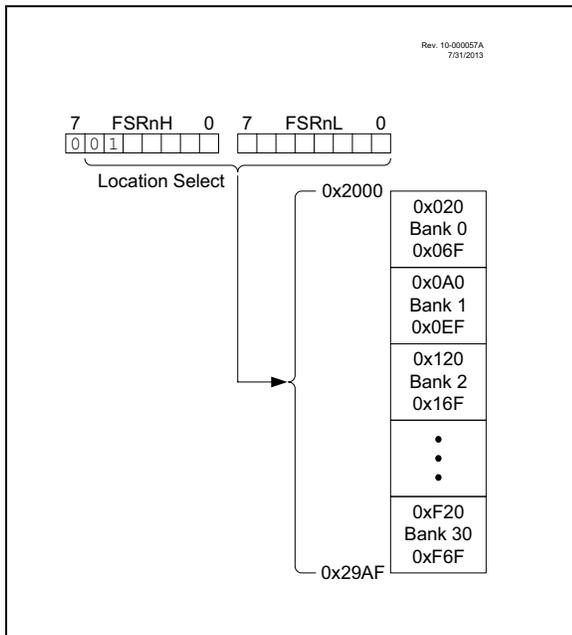
3.6.2 LINEAR DATA MEMORY

The linear data memory is the region from FSR address 0x2000 to FSR address 0x29AF. This region is a virtual region that points back to the 80-byte blocks of GPR memory in all the banks.

Unimplemented memory reads as 0x00. Use of the linear data memory region allows buffers to be larger than 80 bytes because incrementing the FSR beyond one bank will go directly to the GPR memory of the next bank.

The 16 bytes of common memory are not included in the linear data memory region.

FIGURE 3-10: LINEAR DATA MEMORY MAP



3.6.3 PROGRAM FLASH MEMORY

To make constant data access easier, the entire program Flash memory is mapped to the upper half of the FSR address space. When the MSb of FSRnH is set, the lower 15 bits are the address in program memory which will be accessed through INDF. Only the lower eight bits of each memory location is accessible via INDF. Writing to the program Flash memory cannot be accomplished via the FSR/INDF interface. All instructions that access program Flash memory via the FSR/INDF interface will require one additional instruction cycle to complete.

FIGURE 3-11: PROGRAM FLASH MEMORY MAP

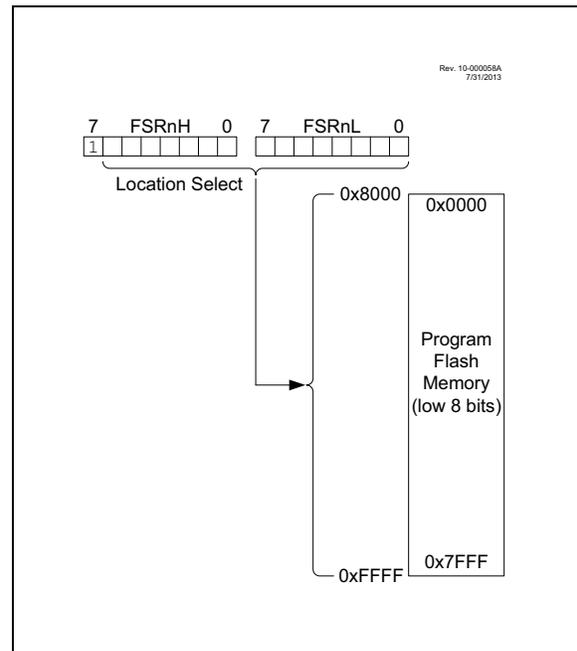
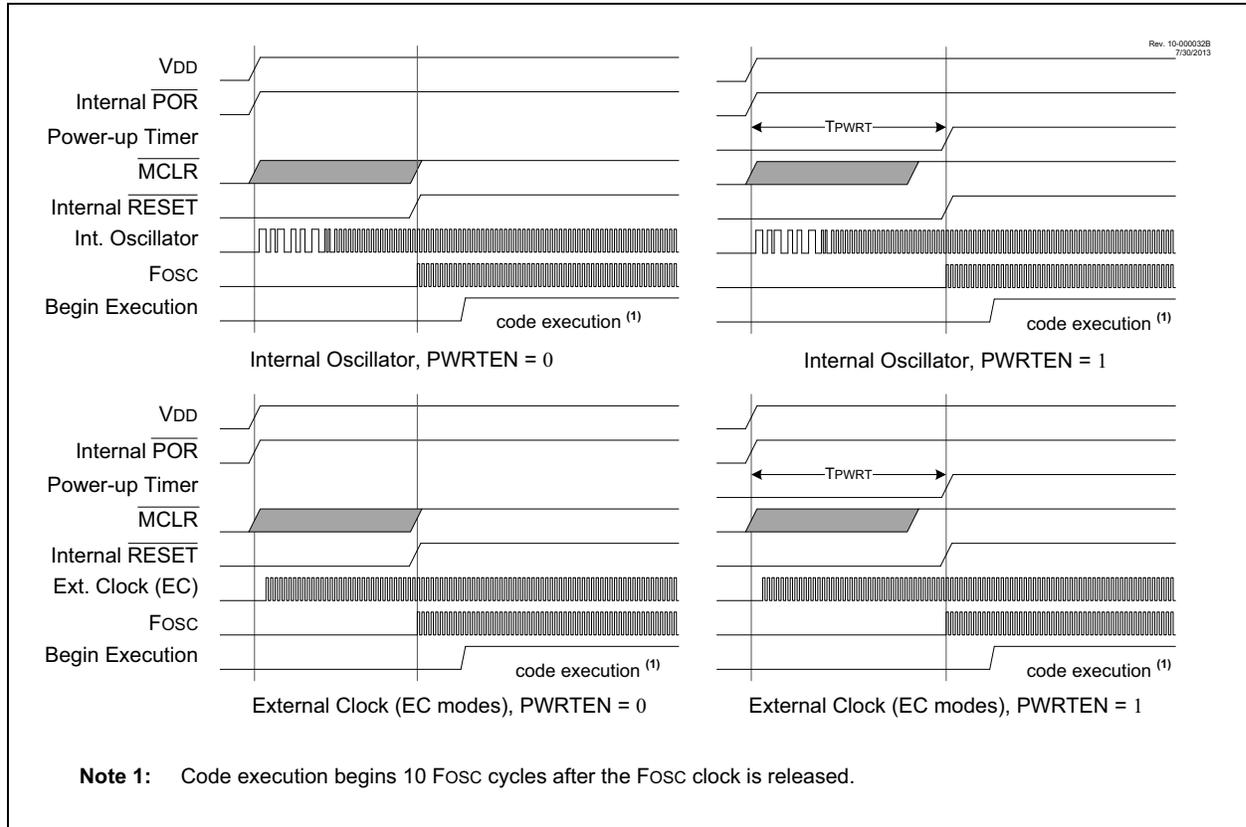


FIGURE 6-3: RESET START-UP SEQUENCE



7.3 Interrupts During Sleep

Some interrupts can be used to wake from Sleep. To wake from Sleep, the peripheral must be able to operate without the system clock. The interrupt source must have the appropriate Interrupt Enable bit(s) set prior to entering Sleep.

On waking from Sleep, if the GIE bit is also set, the processor will branch to the interrupt vector. Otherwise, the processor will continue executing instructions after the `SLEEP` instruction. The instruction directly after the `SLEEP` instruction will always be executed before branching to the ISR. Refer to **Section 8.0 “Power-Down Mode (Sleep)”** for more details.

7.4 INT Pin

The INT pin can be used to generate an asynchronous edge-triggered interrupt. This interrupt is enabled by setting the INTE bit of the INTCON register. The INTEDG bit of the OPTION_REG register determines on which edge the interrupt will occur. When the INTEDG bit is set, the rising edge will cause the interrupt. When the INTEDG bit is clear, the falling edge will cause the interrupt. The INTF bit of the INTCON register will be set when a valid edge appears on the INT pin. If the GIE and INTE bits are also set, the processor will redirect program execution to the interrupt vector.

7.5 Automatic Context Saving

Upon entering an interrupt, the return PC address is saved on the stack. Additionally, the following registers are automatically saved in the shadow registers:

- W register
- STATUS register (except for \overline{TO} and \overline{PD})
- BSR register
- FSR registers
- PCLATH register

Upon exiting the Interrupt Service Routine, these registers are automatically restored. Any modifications to these registers during the ISR will be lost. If modifications to any of these registers are desired, the corresponding shadow register should be modified and the value will be restored when exiting the ISR. The shadow registers are available in Bank 31 and are readable and writable. Depending on the user's application, other registers may also need to be saved.

REGISTER 7-4: PIE3: PERIPHERAL INTERRUPT ENABLE REGISTER 3

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0
—	—	—	—	—	—	CLC2IE	CLC1IE
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 7-2 **Unimplemented:** Read as '0'

bit 1 **CLC2IE:** Configurable Logic Block 2 Interrupt Enable bit

1 = Enables the CLC 2 interrupt

0 = Disables the CLC 2 interrupt

bit 0 **CLC1IE:** Configurable Logic Block 1 Interrupt Enable bit

1 = Enables the CLC 1 interrupt

0 = Disables the CLC 1 interrupt

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

PIC12(L)F1501

10.2.2 FLASH MEMORY UNLOCK SEQUENCE

The unlock sequence is a mechanism that protects the Flash program memory from unintended self-write programming or erasing. The sequence must be executed and completed without interruption to successfully complete any of the following operations:

- Row Erase
- Load program memory write latches
- Write of program memory write latches to program memory
- Write of program memory write latches to User IDs

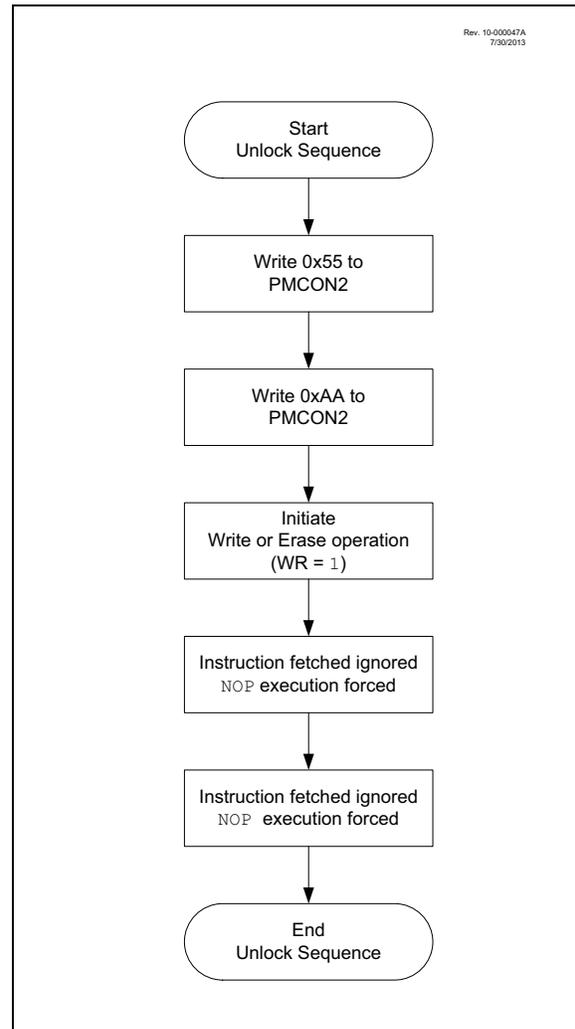
The unlock sequence consists of the following steps:

1. Write 55h to PMCON2
2. Write AAh to PMCON2
3. Set the WR bit in PMCON1
4. NOP instruction
5. NOP instruction

Once the WR bit is set, the processor will always force two NOP instructions. When an Erase Row or Program Row operation is being performed, the processor will stall internal operations (typical 2 ms), until the operation is complete and then resume with the next instruction. When the operation is loading the program memory write latches, the processor will always force the two NOP instructions and continue uninterrupted with the next instruction.

Since the unlock sequence must not be interrupted, global interrupts should be disabled prior to the unlock sequence and re-enabled after the unlock sequence is completed.

FIGURE 10-3: FLASH PROGRAM MEMORY UNLOCK SEQUENCE FLOWCHART



REGISTER 15-4: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 0

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
ADRES<9:2>							
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 **ADRES<9:2>**: ADC Result Register bits
Upper eight bits of 10-bit conversion result

REGISTER 15-5: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0

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
ADRES<1:0>		—	—	—	—	—	—
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 7-6 **ADRES<1:0>**: ADC Result Register bits
Lower two bits of 10-bit conversion result

bit 5-0 **Reserved**: Do not use.

17.0 COMPARATOR MODULE

Comparators are used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. Comparators are very useful mixed signal building blocks because they provide analog functionality independent of program execution. The analog comparator module includes the following features:

- Independent comparator control
- Programmable input selection
- Comparator output is available internally/externally
- Programmable output polarity
- Interrupt-on-change
- Wake-up from Sleep
- Programmable Speed/Power optimization
- PWM shutdown
- Programmable and fixed voltage reference

17.1 Comparator Overview

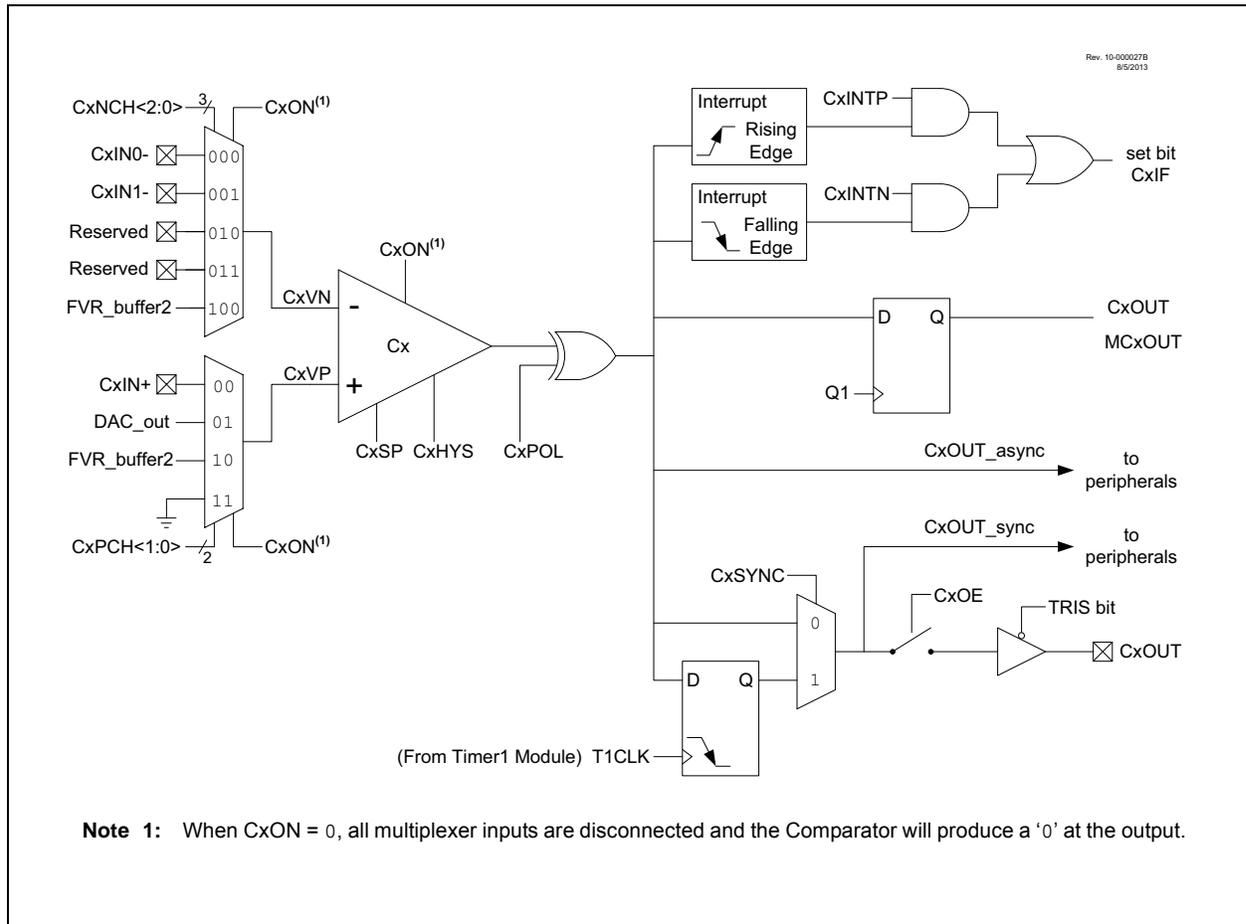
A single comparator is shown in Figure 17-2 along with the relationship between the analog input levels and the digital output. When the analog voltage at V_{IN+} is less than the analog voltage at V_{IN-} , the output of the comparator is a digital low level. When the analog voltage at V_{IN+} is greater than the analog voltage at V_{IN-} , the output of the comparator is a digital high level.

The comparators available for this device are listed in Table 17-1.

TABLE 17-1: AVAILABLE COMPARATORS

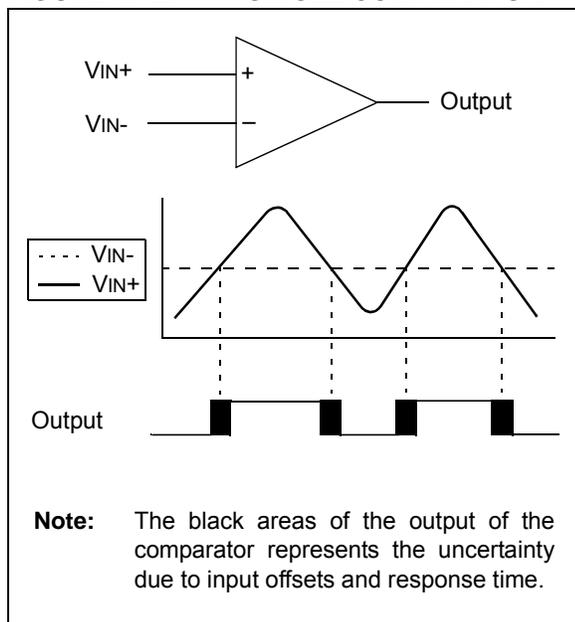
Device	C1	C2
PIC12(L)F1501	•	

FIGURE 17-1: COMPARATOR MODULE SIMPLIFIED BLOCK DIAGRAM



PIC12(L)F1501

FIGURE 17-2: SINGLE COMPARATOR



17.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 registers (see Register 17-1) contain Control and Status bits for the following:

- Enable
- Output selection
- Output polarity
- Speed/Power selection
- Hysteresis enable
- Output synchronization

The CMxCON1 registers (see Register 17-2) contain Control bits for the following:

- Interrupt enable
- Interrupt edge polarity
- Positive input channel selection
- Negative input channel selection

17.2.1 COMPARATOR ENABLE

Setting the CxON bit of the CMxCON0 register enables the comparator for operation. Clearing the CxON bit disables the comparator resulting in minimum current consumption.

17.2.2 COMPARATOR POSITIVE INPUT SELECTION

Configuring the CxPCH<1:0> bits of the CMxCON1 register directs an internal voltage reference or an analog pin to the non-inverting input of the comparator:

- CxIN+ analog pin
- DAC1_output
- FVR_buffer2
- Vss

See **Section 13.0 “Fixed Voltage Reference (FVR)”** for more information on the Fixed Voltage Reference module.

See **Section 16.0 “5-Bit Digital-to-Analog Converter (DAC) Module”** for more information on the DAC input signal.

Any time the comparator is disabled (CxON = 0), all comparator inputs are disabled.

17.2.3 COMPARATOR NEGATIVE INPUT SELECTION

The CxNCH<2:0> bits of the CMxCON0 register direct one of the input sources to the comparator inverting input.

Note: To use CxIN+ and CxINx- pins as analog input, the appropriate bits must be set in the ANSEL register and the corresponding TRIS bits must also be set to disable the output drivers.

17.2.4 COMPARATOR OUTPUT SELECTION

The output of the comparator can be monitored by reading either the CxOUT bit of the CMxCON0 register or the MCxOUT bit of the CMOUT register. In order to make the output available for an external connection, the following conditions must be true:

- CxOE bit of the CMxCON0 register must be set
- Corresponding TRIS bit must be cleared
- CxON bit of the CMxCON0 register must be set

The synchronous comparator output signal (CxOUT_sync) is available to the following peripheral(s):

- Configurable Logic Cell (CLC)
- Analog-to-Digital Converter (ADC)
- Timer1

The asynchronous comparator output signal (CxOUT_async) is available to the following peripheral(s):

- Complementary Waveform Generator (CWG)

Note 1: The CxOE bit of the CMxCON0 register overrides the PORT data latch. Setting the CxON bit of the CMxCON0 register has no impact on the port override.

2: The internal output of the comparator is latched with each instruction cycle. Unless otherwise specified, external outputs are not latched.

19.8 Register Definitions: Timer1 Control

REGISTER 19-1: T1CON: TIMER1 CONTROL REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	U-0	R/W-0/u	U-0	R/W-0/u
TMR1CS<1:0>		T1CKPS<1:0>		—	$\overline{T1SYNC}$	—	TMR1ON
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 **TMR1CS<1:0>**: Timer1 Clock Source Select bits
 - 11 = Timer1 clock source is LFINTOSC
 - 10 = Timer1 clock source is T1CKI pin (on the rising edge)
 - 01 = Timer1 clock source is system clock (Fosc)
 - 00 = Timer1 clock source is instruction clock (Fosc/4)
- bit 5-4 **T1CKPS<1:0>**: Timer1 Input Clock Prescale Select bits
 - 11 = 1:8 Prescale value
 - 10 = 1:4 Prescale value
 - 01 = 1:2 Prescale value
 - 00 = 1:1 Prescale value
- bit 3 **Unimplemented**: Read as '0'
- bit 2 **T1SYNC**: Timer1 Synchronization Control bit
 - 1 = Do not synchronize asynchronous clock input
 - 0 = Synchronize asynchronous clock input with system clock (Fosc)
- bit 1 **Unimplemented**: Read as '0'
- bit 0 **TMR1ON**: Timer1 On bit
 - 1 = Enables Timer1
 - 0 = Stops Timer1 and clears Timer1 gate flip-flop

PIC12(L)F1501

21.1.9 SETUP FOR PWM OPERATION USING PWMx PINS

The following steps should be taken when configuring the module for PWM operation using the PWMx pins:

1. Disable the PWMx pin output driver(s) by setting the associated TRIS bit(s).
2. Clear the PWMxCON register.
3. Load the PR2 register with the PWM period value.
4. Clear the PWMxDCH register and bits <7:6> of the PWMxDCL register.
5. Configure and start Timer2:
 - Clear the TMR2IF interrupt flag bit of the PIR1 register. See note below.
 - Configure the T2CKPS bits of the T2CON register with the Timer2 prescale value.
 - Enable Timer2 by setting the TMR2ON bit of the T2CON register.
6. Enable PWM output pin and wait until Timer2 overflows, TMR2IF bit of the PIR1 register is set. See note below.
7. Enable the PWMx pin output driver(s) by clearing the associated TRIS bit(s) and setting the PWMxOE bit of the PWMxCON register.
8. Configure the PWM module by loading the PWMxCON register with the appropriate values.

Note 1: In order to send a complete duty cycle and period on the first PWM output, the above steps must be followed in the order given. If it is not critical to start with a complete PWM signal, then move Step 8 to replace Step 4.

2: For operation with other peripherals only, disable PWMx pin outputs.

PIC12(L)F1501

REGISTER 22-6: CLCxGLS1: GATE 2 LOGIC SELECT REGISTER

| R/W-x/u |
|----------|----------|----------|----------|----------|----------|----------|----------|
| LCxG2D4T | LCxG2D4N | LCxG2D3T | LCxG2D3N | LCxG2D2T | LCxG2D2N | LCxG2D1T | LCxG2D1N |
| 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 **LCxG2D4T:** Gate 2 Data 4 True (non-inverted) bit
 1 = lcx4T is gated into lcxg2
 0 = lcx4T is not gated into lcxg2
- bit 6 **LCxG2D4N:** Gate 2 Data 4 Negated (inverted) bit
 1 = lcx4N is gated into lcxg2
 0 = lcx4N is not gated into lcxg2
- bit 5 **LCxG2D3T:** Gate 2 Data 3 True (non-inverted) bit
 1 = lcx3T is gated into lcxg2
 0 = lcx3T is not gated into lcxg2
- bit 4 **LCxG2D3N:** Gate 2 Data 3 Negated (inverted) bit
 1 = lcx3N is gated into lcxg2
 0 = lcx3N is not gated into lcxg2
- bit 3 **LCxG2D2T:** Gate 2 Data 2 True (non-inverted) bit
 1 = lcx2T is gated into lcxg2
 0 = lcx2T is not gated into lcxg2
- bit 2 **LCxG2D2N:** Gate 2 Data 2 Negated (inverted) bit
 1 = lcx2N is gated into lcxg2
 0 = lcx2N is not gated into lcxg2
- bit 1 **LCxG2D1T:** Gate 2 Data 1 True (non-inverted) bit
 1 = lcx1T is gated into lcxg2
 0 = lcx1T is not gated into lcxg2
- bit 0 **LCxG2D1N:** Gate 2 Data 1 Negated (inverted) bit
 1 = lcx1N is gated into lcxg2
 0 = lcx1N is not gated into lcxg2

24.8 Dead-Band Uncertainty

When the rising and falling edges of the input source triggers the dead-band counters, the input may be asynchronous. This will create some uncertainty in the dead-band time delay. The maximum uncertainty is equal to one CWG clock period. Refer to Equation 24-1 for more detail.

EQUATION 24-1: DEAD-BAND UNCERTAINTY

$$T_{DEADBAND_UNCERTAINTY} = \frac{1}{F_{cwg_clock}}$$

Example:

$$F_{cwg_clock} = 16 \text{ MHz}$$

Therefore:

$$\begin{aligned} T_{DEADBAND_UNCERTAINTY} &= \frac{1}{F_{cwg_clock}} \\ &= \frac{1}{16 \text{ MHz}} \\ &= 62.5 \text{ ns} \end{aligned}$$

24.9 Auto-Shutdown Control

Auto-shutdown is a method to immediately override the CWG output levels with specific overrides that allow for safe shutdown of the circuit. The shutdown state can be either cleared automatically or held until cleared by software.

24.9.1 SHUTDOWN

The shutdown state can be entered by either of the following two methods:

- Software generated
- External Input

24.9.1.1 Software Generated Shutdown

Setting the GxASE bit of the CWGxCON2 register will force the CWG into the shutdown state.

When auto-restart is disabled, the shutdown state will persist as long as the GxASE bit is set.

When auto-restart is enabled, the GxASE bit will clear automatically and resume operation on the next rising edge event. See Figure 24-6.

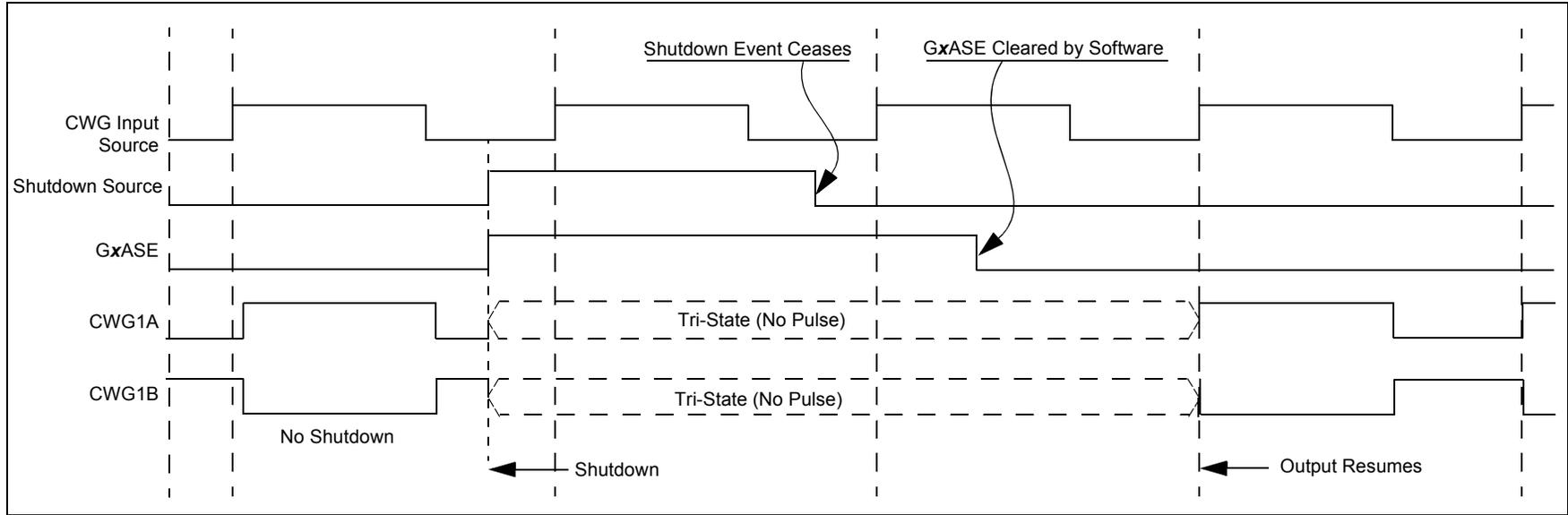
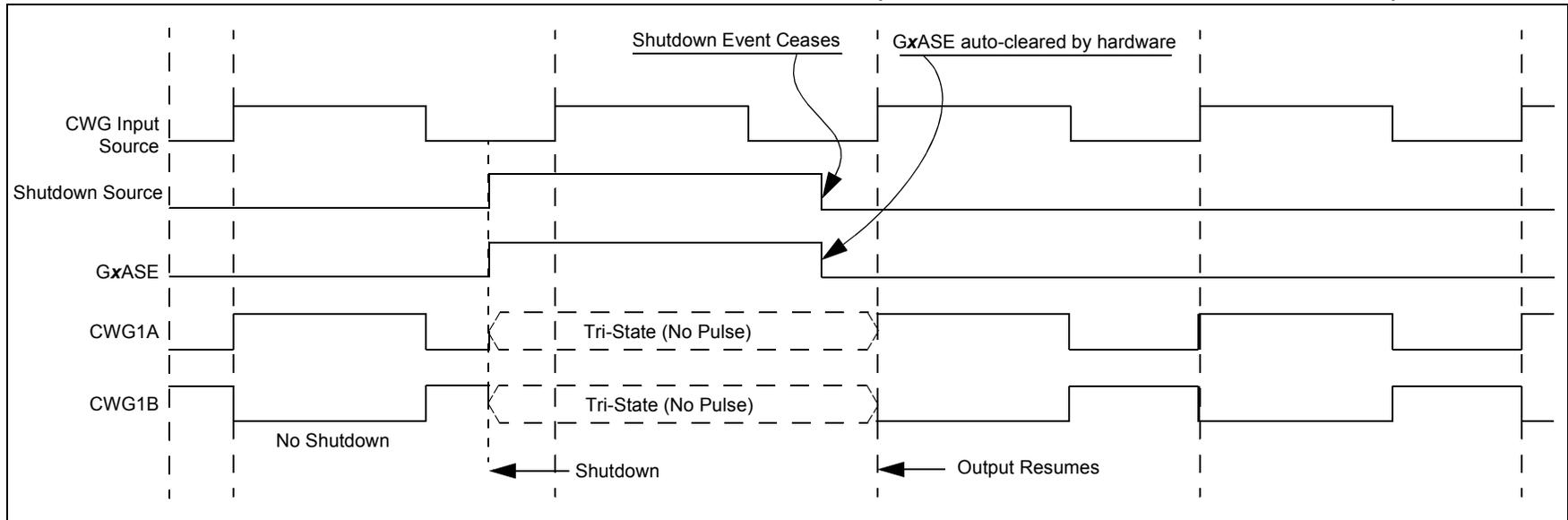
24.9.1.2 External Input Source

External shutdown inputs provide the fastest way to safely suspend CWG operation in the event of a Fault condition. When any of the selected shutdown inputs goes active, the CWG outputs will immediately go to the selected override levels without software delay. Any combination of two input sources can be selected to cause a shutdown condition. The sources are:

- Comparator C1 – C1OUT_async
- CLC2 – LC2_out
- $\overline{\text{CWG1FLT}}$

Shutdown inputs are selected in the CWGxCON2 register. (Register 24-3).

Note: Shutdown inputs are level sensitive, not edge sensitive. The shutdown state cannot be cleared, except by disabling auto-shutdown, as long as the shutdown input level persists.

FIGURE 24-5: SHUTDOWN FUNCTIONALITY, AUTO-RESTART DISABLED ($GxARSEN = 0$, $GxASDLA = 01$, $GxASDLB = 01$)

FIGURE 24-6: SHUTDOWN FUNCTIONALITY, AUTO-RESTART ENABLED ($GxARSEN = 1$, $GxASDLA = 01$, $GxASDLB = 01$)


PIC12(L)F1501

TABLE 27-3: POWER-DOWN CURRENTS (I_{PD})^(1,2)

PIC12LF1501		Operating Conditions: (unless otherwise stated) Low-Power Sleep Mode						
PIC12F1501		Low-Power Sleep Mode, VREGPM = 1						
Param. No.	Device Characteristics	Min.	Typ†	Max. +85°C	Max. +125°C	Units	Conditions	
							VDD	Note
D022	Base I _{PD}	—	0.020	1.0	8.0	μA	1.8	WDT, BOR, FVR and SOSC disabled, all Peripherals inactive
		—	0.03	2.0	9.0	μA	3.0	
D022	Base I _{PD}	—	0.25	3.0	10	μA	2.3	WDT, BOR, FVR and SOSC disabled, all Peripherals inactive, Low-Power Sleep mode
		—	0.30	4.0	12	μA	3.0	
		—	0.40	6.0	15	μA	5.0	
D022A	Base I _{PD}	—	10	16	18	μA	2.3	WDT, BOR, FVR and SOSC disabled, all Peripherals inactive, Normal Power Sleep mode, VREGPM = 0
		—	11	18	20	μA	3.0	
		—	12	21	26	μA	5.0	
D023		—	0.26	2.0	9.0	μA	1.8	WDT Current
		—	0.44	3.0	10	μA	3.0	
D023		—	0.43	6.0	15	μA	2.3	WDT Current
		—	0.53	7.0	20	μA	3.0	
		—	0.64	8.0	22	μA	5.0	
D023A		—	15	28	30	μA	1.8	FVR Current
		—	18	30	33	μA	3.0	
D023A		—	18	33	35	μA	2.3	FVR Current
		—	19	35	37	μA	3.0	
		—	20	37	39	μA	5.0	
D024		—	6.0	17	20	μA	3.0	BOR Current
D024		—	7.0	17	30	μA	3.0	BOR Current
		—	8.0	20	40	μA	5.0	
D24A		—	0.1	4.0	10	μA	3.0	LPBOR Current
D24A		—	0.35	5.0	14	μA	3.0	LPBOR Current
		—	0.45	8.0	17	μA	5.0	

* 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.

- Note** 1: The peripheral Δ current can be determined by subtracting the base I_{PD} current from this limit. Max. values should be used when calculating total current consumption.
- 2: 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 V_{SS}.
- 3: ADC clock source is FRC.

FIGURE 27-12: ADC CONVERSION TIMING (ADC CLOCK Fosc-BASED)

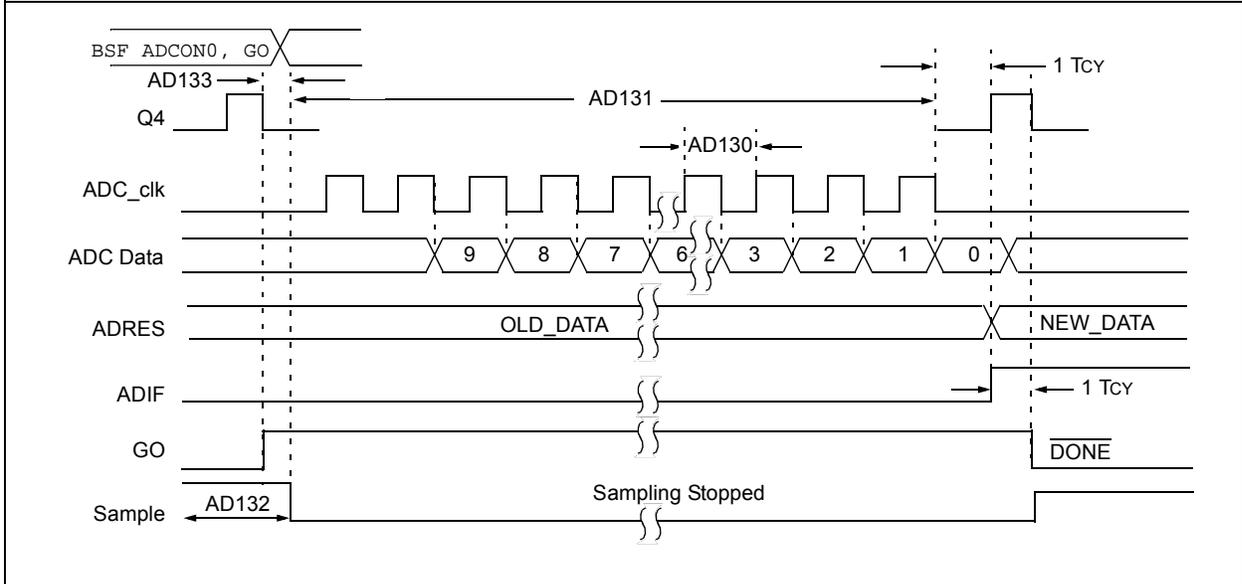
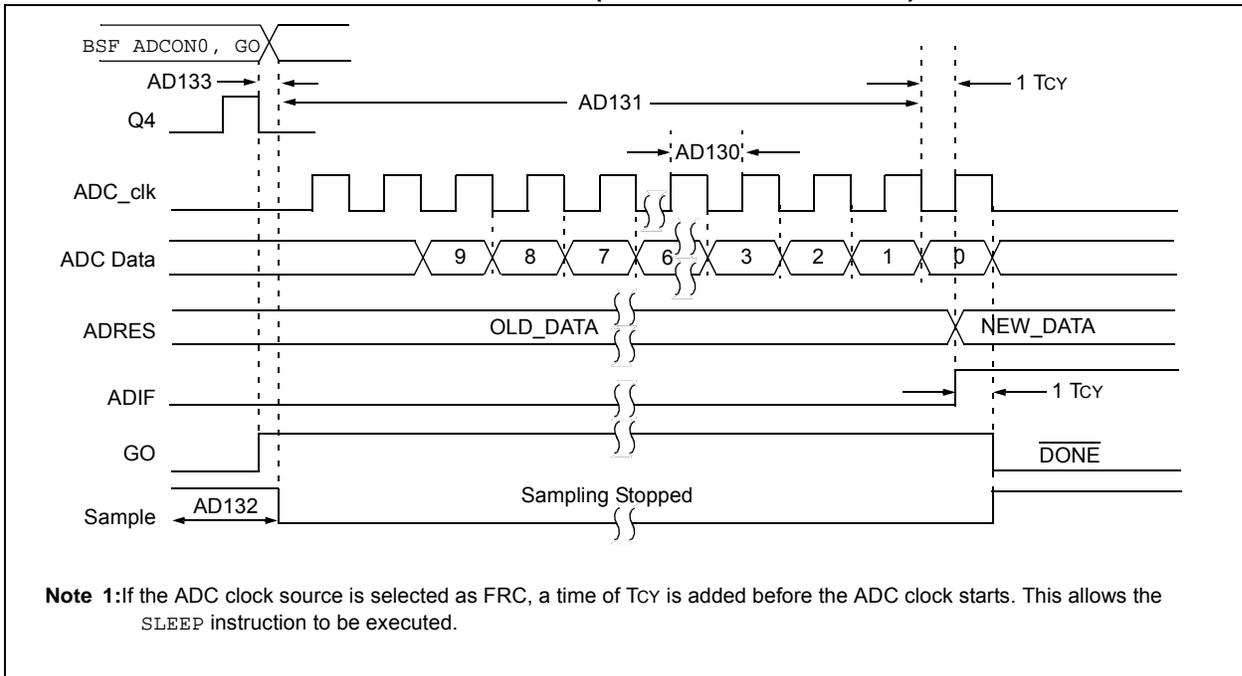


FIGURE 27-13: ADC CONVERSION TIMING (ADC CLOCK FROM FRC)



PIC12(L)F1501

FIGURE 28-3: I_{DD} , EXTERNAL CLOCK (ECL), LOW-POWER MODE, $F_{osc} = 500$ kHz, PIC12LF1501 ONLY

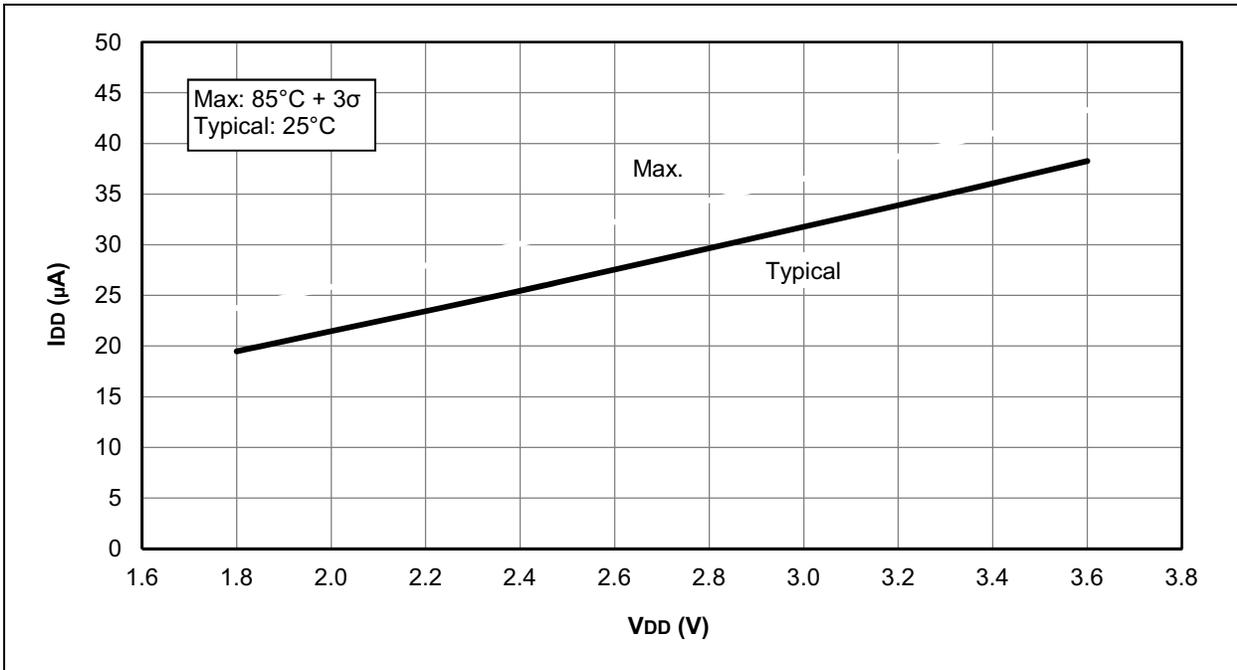
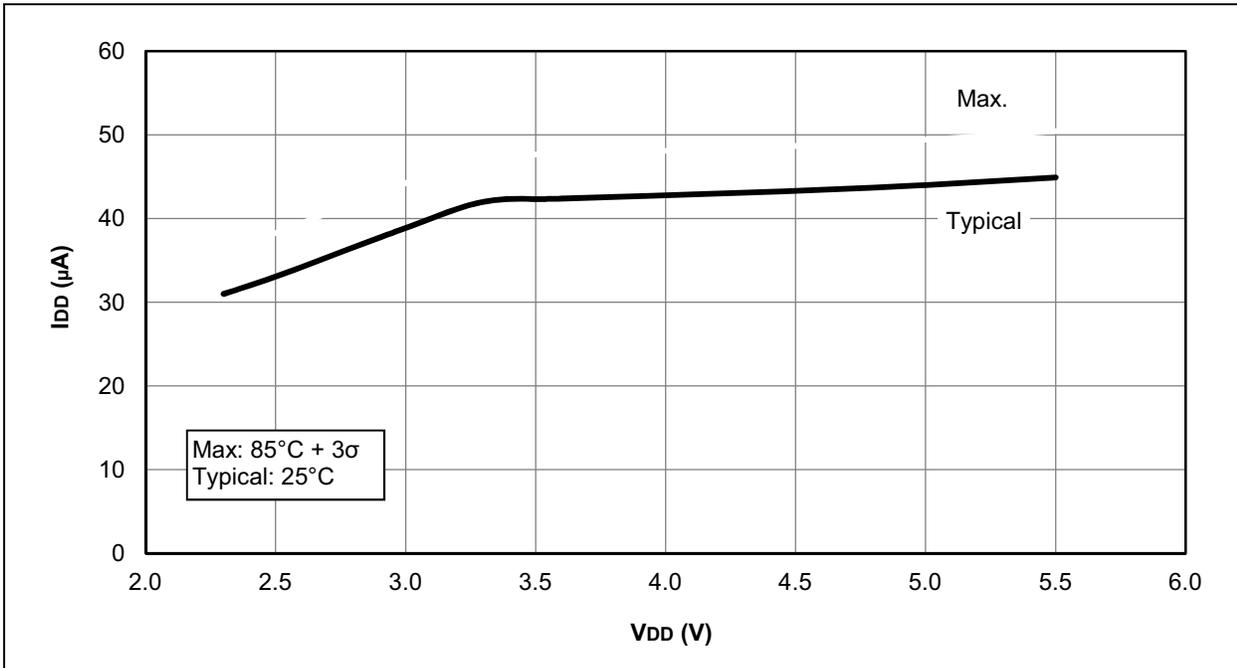


FIGURE 28-4: I_{DD} , EXTERNAL CLOCK (ECL), LOW-POWER MODE, $F_{osc} = 500$ kHz, PIC12F1501 ONLY



PIC12(L)F1501

FIGURE 28-11: I_{DD} TYPICAL, EXTERNAL CLOCK (ECH), HIGH-POWER MODE, PIC12F1501 ONLY

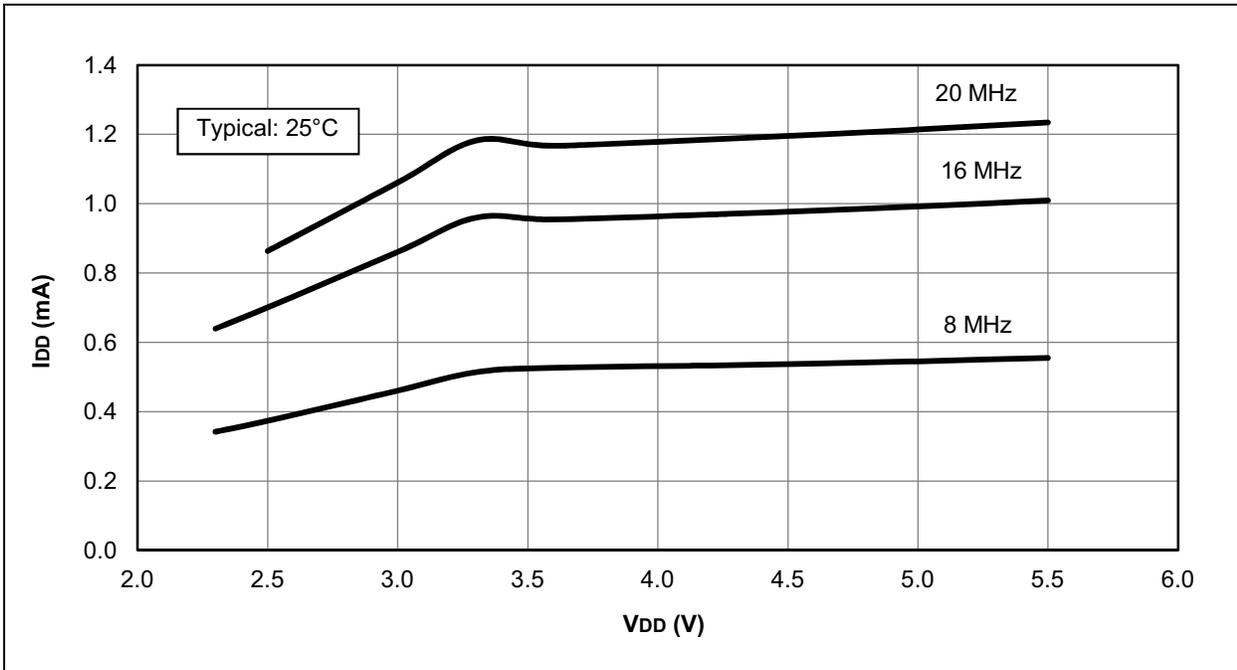
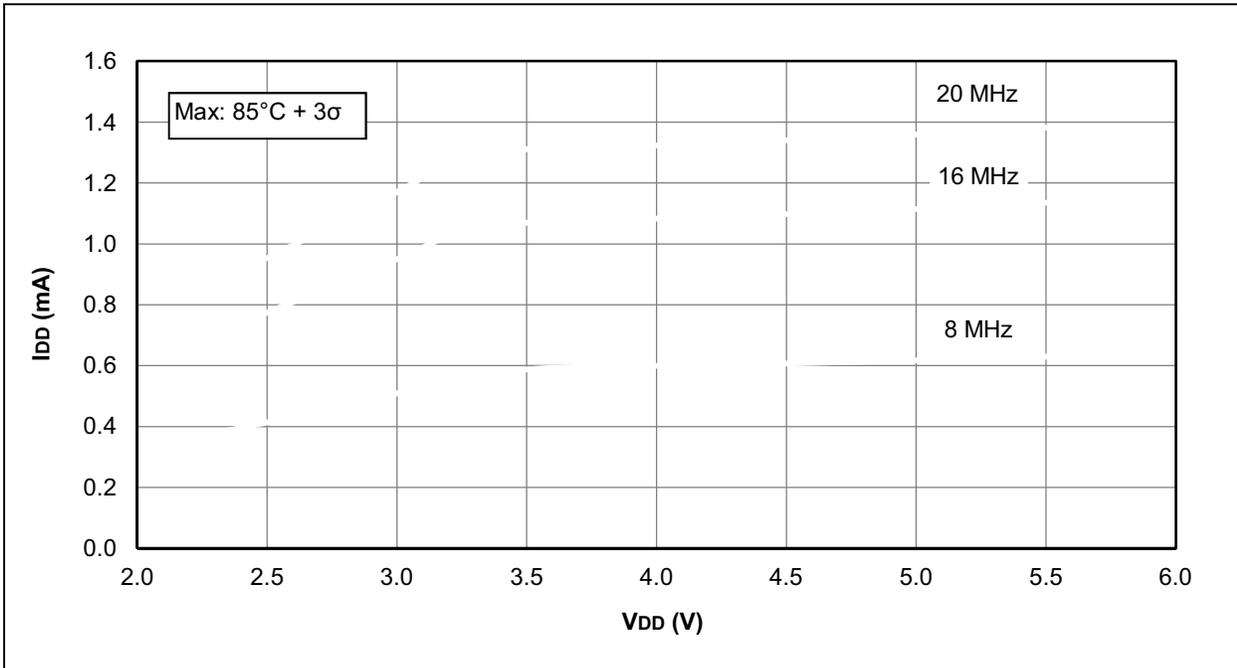


FIGURE 28-12: I_{DD} MAXIMUM, EXTERNAL CLOCK (ECH), HIGH-POWER MODE, PIC12F1501 ONLY



29.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

29.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

29.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

29.9 PICkit 3 In-Circuit Debugger/Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

29.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

TABLE 30-1: 8-LEAD 2x3 DFN (MC) TOP MARKING

Part Number	Marking
PIC12F1501-E/MC	BAK
PIC12F1501-I/MC	BAL
PIC12LF1501-E/MC	BAM
PIC12LF1501-I/MC	BAP

TABLE 30-2: 8-LEAD 3x3 DFN (MF) TOP MARKING

Part Number	Marking
PIC12F1501-E/MF	MFA1
PIC12F1501-I/MF	MFB1
PIC12LF1501-E/MF	MFC1
PIC12LF1501-I/MF	MFD1

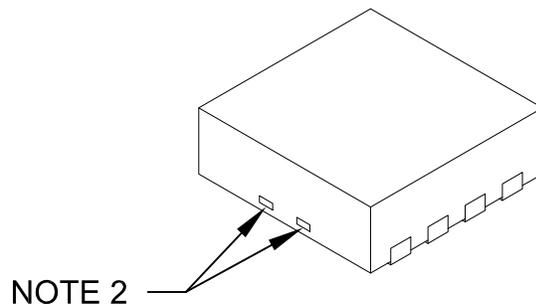
TABLE 30-3: 8-LEAD 2X3 UDFN (MU) TOP MARKING

Part Number	Marking
PIC12F1501-E/MU	BAR
PIC12F1501-I/MU	BAQ
PIC12LF1501-E/MU	BAT
PIC12LF1501-I/MU	BAS

PIC12(L)F1501

8-Lead Plastic Dual Flat, No Lead Package (MF) - 3x3x0.9mm Body [DFN]

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



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	8		
Pitch	e	0.65 BSC		
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3	0.20 REF		
Overall Length	D	3.00 BSC		
Exposed Pad Width	E2	1.34	-	1.60
Overall Width	E	3.00 BSC		
Exposed Pad Length	D2	1.60	-	2.40
Contact Width	b	0.25	0.30	0.35
Contact Length	L	0.20	0.30	0.55
Contact-to-Exposed Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package may have one or more exposed tie bars at ends.
3. Package is saw singulated
4. Dimensioning and tolerancing per ASME Y14.5M

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

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing No. C04-062C Sheet 2 of 2