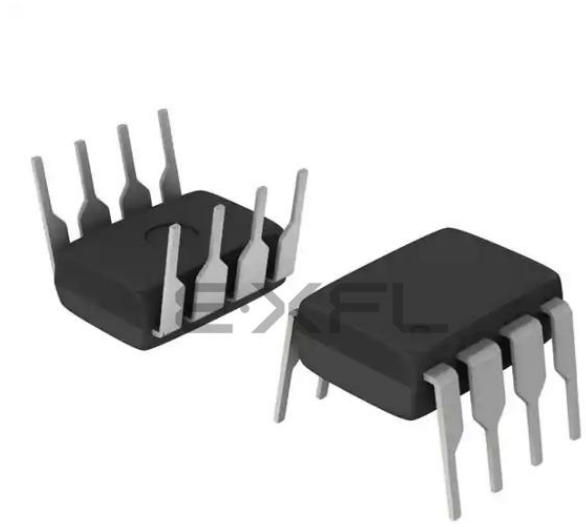


Welcome to E-XFL.COM

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	Through Hole
Package / Case	8-DIP (0.300", 7.62mm)
Supplier Device Package	8-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12lf1501-e-p

2.1 Automatic Interrupt Context Saving

During interrupts, certain registers are automatically saved in shadow registers and restored when returning from the interrupt. This saves stack space and user code. See **Section 7.5 "Automatic Context Saving"**, for more information.

2.2 16-Level Stack with Overflow and Underflow

These devices have a hardware stack memory 15 bits wide and 16 words deep. A Stack Overflow or Underflow will set the appropriate bit (STKOVF or STKUNF) in the PCON register, and if enabled, will cause a software Reset. See **Section 3.5 "Stack"** for more details.

2.3 File Select Registers

There are two 16-bit File Select Registers (FSR). FSRs can access all file registers and program memory, which allows one Data Pointer for all memory. When an FSR points to program memory, there is one additional instruction cycle in instructions using INDF to allow the data to be fetched. General purpose memory can now also be addressed linearly, providing the ability to access contiguous data larger than 80 bytes. There are also new instructions to support the FSRs. See **Section 3.6 "Indirect Addressing"** for more details.

2.4 Instruction Set

There are 49 instructions for the enhanced mid-range CPU to support the features of the CPU. See **Section 26.0 "Instruction Set Summary"** for more details.

TABLE 3-5: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 10											
50Ch to 51Fh	—	Unimplemented								—	—
Bank 11											
58Ch to 59Fh	—	Unimplemented								—	—
Bank 12											
60Ch to 610h	—	Unimplemented								—	—
611h	PWM1DCL	PWM1DCL<7:6>		—	—	—	—	—	—	00-- ----	00-- ----
612h	PWM1DCH	PWM1DCH<7:0>								xxxx xxxx	uuuu uuuu
613h	PWM1CON0	PWM1EN	PWM1OE	PWM1OUT	PWM1POL	—	—	—	—	0000 ----	0000 ----
614h	PWM2DCL	PWM2DCL<7:6>		—	—	—	—	—	—	00-- ----	00-- ----
615h	PWM2DCH	PWM2DCH<7:0>								xxxx xxxx	uuuu uuuu
616h	PWM2CON0	PWM2EN	PWM2OE	PWM2OUT	PWM2POL	—	—	—	—	0000 ----	0000 ----
617h	PWM3DCL	PWM3DCL<7:6>		—	—	—	—	—	—	00-- ----	00-- ----
618h	PWM3DCH	PWM3DCH<7:0>								xxxx xxxx	uuuu uuuu
619h	PWM3CON0	PWM3EN	PWM3OE	PWM3OUT	PWM3POL	—	—	—	—	0000 ----	0000 ----
61Ah	PWM4DCL	PWM4DCL<7:6>		—	—	—	—	—	—	00-- ----	00-- ----
61Bh	PWM4DCH	PWM4DCH<7:0>								xxxx xxxx	uuuu uuuu
61Ch	PWM4CON0	PWM4EN	PWM4OE	PWM4OUT	PWM4POL	—	—	—	—	0000 ----	0000 ----
61Dh to 61Fh	—	Unimplemented								—	—
Bank 13											
68Ch to 690h	—	Unimplemented								—	—
691h	CWG1DBR	—	—	CWG1DBR<5:0>						--00 0000	--00 0000
692h	CWG1DBF	—	—	CWG1DBF<5:0>						--xx xxxx	--xx xxxx
693h	CWG1CON0	G1EN	G1OEB	G1OEA	G1POLB	G1POLA	—	—	G1CS0	0000 0--0	0000 0--0
694h	CWG1CON1	G1ASDLB<1:0>		G1ASDLA<1:0>		—	G1IS<2:0>			0000 -000	0000 -000
695h	CWG1CON2	G1ASE	G1ARSEN	—	—	—	G1ASDSC1	G1ASDSFLT	G1ASDSCLC2	00-- -000	00-- -000
696h to 69Fh	—	Unimplemented								—	—

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'.
Note 1: PIC12F1501 only.
2: Unimplemented, read as '1'.

5.0 OSCILLATOR MODULE

5.1 Overview

The oscillator module has a wide variety of clock sources and selection features that allow it to be used in a wide range of applications while maximizing performance and minimizing power consumption. Figure 5-1 illustrates a block diagram of the oscillator module.

Clock sources can be supplied from an external clock or from one of two internal oscillators, with a choice of speeds selectable via software. Additional clock features include:

- Selectable system clock source between external or internal sources via software.
- Fast start-up oscillator allows internal circuits to power-up and stabilize before switching to the 16 MHz HFINTOSC

The oscillator module can be configured in one of the following clock modes.

1. ECL – External Clock Low-Power mode (0 MHz to 0.5 MHz)
2. ECM – External Clock Medium Power mode (0.5 MHz to 4 MHz)
3. ECH – External Clock High-Power mode (4 MHz to 20 MHz)
4. INTOSC – Internal oscillator (31 kHz to 16 MHz)

Clock Source modes are selected by the FOSC<1:0> bits in the Configuration Words. The FOSC bits determine the type of oscillator that will be used when the device is first powered.

The ECH, ECM, and ECL clock modes rely on an external logic level signal as the device clock source.

The INTOSC internal oscillator block produces a low and high-frequency clock source, designated LFINTOSC and HFINTOSC. (See Internal Oscillator Block, Figure 5-1). A wide selection of device clock frequencies may be derived from these two clock sources.

5.2.2.4 Peripheral Clock Sources

The clock sources described in this chapter and the Timer's are available to different peripherals. Table 5-1 lists the clocks and timers available for each peripheral.

TABLE 5-1: PERIPHERAL CLOCK SOURCES

	FOSC	FRC	HFINTOSC	LFINTOSC	TMR0	TMR1	TMR2
ADC	●	●					
CLC	●	●	●	●	●	●	●
COMP						●	
CWG	●		●				
NCO	●		●				
PWM	●						●
PWRT				●			
TMR0	●						
TMR1	●			●			
TMR2	●						
WDT				●			

5.2.2.5 Internal Oscillator Frequency Selection

The system clock speed can be selected via software using the Internal Oscillator Frequency Select bits IRCF<3:0> of the OSCCON register.

The postscaled output of the 16 MHz HFINTOSC and 31 kHz LFINTOSC connect to a multiplexer (see Figure 5-1). The Internal Oscillator Frequency Select bits IRCF<3:0> of the OSCCON register (Register 5-1) select the frequency output of the internal oscillators.

Note:	Following any Reset, the IRCF<3:0> bits of the OSCCON register are set to '0111' and the frequency selection is set to 500 kHz. The user can modify the IRCF bits to select a different frequency.

The IRCF<3:0> bits of the OSCCON register allow duplicate selections for some frequencies. These duplicate choices can offer system design trade-offs. Lower power consumption can be obtained when changing oscillator sources for a given frequency. Faster transition times can be obtained between frequency changes that use the same oscillator source.

5.2.2.6 Internal Oscillator Clock Switch Timing

When switching between the HFINTOSC and the LFINTOSC, the new oscillator may already be shut down to save power (see Figure 5-3). If this is the case, there is a delay after the IRCF<3:0> bits of the OSCCON register are modified before the frequency selection takes place. The OSCSTAT register will reflect the current active status of the HFINTOSC and LFINTOSC oscillators. The sequence of a frequency selection is as follows:

1. IRCF<3:0> bits of the OSCCON register are modified.
2. If the new clock is shut down, a clock start-up delay is started.
3. Clock switch circuitry waits for a falling edge of the current clock.
4. The current clock is held low and the clock switch circuitry waits for a rising edge in the new clock.
5. The new clock is now active.
6. The OSCSTAT register is updated as required.
7. Clock switch is complete.

See Figure 5-3 for more details.

If the internal oscillator speed is switched between two clocks of the same source, there is no start-up delay before the new frequency is selected. Clock switching time delays are shown in Table 5-2.

Start-up delay specifications are located in Table 27-8, "Oscillator Parameters".

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.

8.3 Register Definitions: Voltage Regulator Control

REGISTER 8-1: VREGCON: VOLTAGE REGULATOR CONTROL REGISTER(1)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-1/1
—	—	—	—	—	—	VREGPM	Reserved
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 **VREGPM:** Voltage Regulator Power Mode Selection bit
- 1 = Low-Power Sleep mode enabled in Sleep(2)
  Draws lowest current in Sleep, slower wake-up
- 0 = Normal Power mode enabled in Sleep(2)
  Draws higher current in Sleep, faster wake-up

bit 0 **Reserved:** Read as '1'. Maintain this bit set.

Note 1: PIC12F1501 only.
2: See **Section 27.0 "Electrical Specifications"**.

TABLE 8-1: SUMMARY OF REGISTERS ASSOCIATED WITH POWER-DOWN MODE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	64
IOCAF	—	—	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	103
IOCAN	—	—	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	103
IOCAP	—	—	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	103
PIE1	TMR1GIE	ADIE	—	—	—	—	TMR2IE	TMR1IE	65
PIE2	—	—	C1IE	—	—	NCO1IE	—	—	66
PIE3	—	—	—	—	—	—	CLC2IE	CLC1IE	67
PIR1	TMR1GIF	ADIF	—	—	—	—	TMR2IF	TMR1IF	68
PIR2	—	—	C1IF	—	—	NCO1IF	—	—	67
PIR3	—	—	—	—	—	—	CLC2IF	CLC1IF	70
STATUS	—	—	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	17
WDTCON	—	—	WDTPS<4:0>					SWDTEN	77

Legend: — = unimplemented, read as '0'. Shaded cells are not used in Power-Down mode.

13.3 Register Definitions: FVR Control

REGISTER 13-1: FVRCON: FIXED VOLTAGE REFERENCE CONTROL REGISTER

R/W-0/0	R-q/q	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
FVREN(1)	FVRRDY(2)	TSEN(3)	TSRNG(3)	CDAFVR<1:0>(1)		ADFVR<1:0>(1)	
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	q = Value depends on condition

bit 7 **FVREN:** Fixed Voltage Reference Enable bit(1)
- 1 = Fixed Voltage Reference is enabled
- 0 = Fixed Voltage Reference is disabled

bit 6 **FVRRDY:** Fixed Voltage Reference Ready Flag bit(2)
- 1 = Fixed Voltage Reference output is ready for use
- 0 = Fixed Voltage Reference output is not ready or not enabled

bit 5 **TSEN:** Temperature Indicator Enable bit(3)
- 1 = Temperature Indicator is enabled
- 0 = Temperature Indicator is disabled

bit 4 **TSRNG:** Temperature Indicator Range Selection bit(3)
- 1 = $V_{OUT} = V_{DD} - 4V_T$ (High Range)
- 0 = $V_{OUT} = V_{DD} - 2V_T$ (Low Range)

bit 3-2 **CDAFVR<1:0>:** Comparator FVR Buffer Gain Selection bits(1)
- 11 = Comparator FVR Buffer Gain is 4x, with output voltage = 4x V_{FVR} (4.096V nominal)(4)
- 10 = Comparator FVR Buffer Gain is 2x, with output voltage = 2x V_{FVR} (2.048V nominal)(4)
- 01 = Comparator FVR Buffer Gain is 1x, with output voltage = 1x V_{FVR} (1.024V nominal)
- 00 = Comparator FVR Buffer is off

bit 1-0 **ADFVR<1:0>:** ADC FVR Buffer Gain Selection bit(1)
- 11 = ADC FVR Buffer Gain is 4x, with output voltage = 4x V_{FVR} (4.096V nominal)(4)
- 10 = ADC FVR Buffer Gain is 2x, with output voltage = 2x V_{FVR} (2.048V nominal)(4)
- 01 = ADC FVR Buffer Gain is 1x, with output voltage = 1x V_{FVR} (1.024V nominal)
- 00 = ADC FVR Buffer is off

Note 1: To minimize current consumption when the FVR is disabled, the FVR buffers should be turned off by clearing the Buffer Gain Selection bits.

2: FVRRDY is always '1' for the PIC12F1501 devices.

3: See **Section 14.0 "Temperature Indicator Module"** for additional information.

4: Fixed Voltage Reference output cannot exceed V_{DD}.

TABLE 13-2: SUMMARY OF REGISTERS ASSOCIATED WITH THE FIXED VOLTAGE REFERENCE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR>1:0>		ADFVR<1:0>		107

Legend: Shaded cells are unused by the Fixed Voltage Reference module.

22.1.2 DATA GATING

Outputs from the input multiplexers are directed to the desired logic function input through the data gating stage. Each data gate can direct any combination of the four selected inputs.

Note:	Data gating is undefined at power-up.

The gate stage is more than just signal direction. The gate can be configured to direct each input signal as inverted or non-inverted data. Directed signals are ANDed together in each gate. The output of each gate can be inverted before going on to the logic function stage.

The gating is in essence a 1-to-4 input AND/NAND/OR/NOR gate. When every input is inverted and the output is inverted, the gate is an OR of all enabled data inputs. When the inputs and output are not inverted, the gate is an AND or all enabled inputs.

Table 22-2 summarizes the basic logic that can be obtained in gate 1 by using the gate logic select bits. The table shows the logic of four input variables, but each gate can be configured to use less than four. If no inputs are selected, the output will be zero or one, depending on the gate output polarity bit.

TABLE 22-2: DATA GATING LOGIC

CLCxGLS0	LCxG1POL	Gate Logic
0x55	1	AND
0x55	0	NAND
0xAA	1	NOR
0xAA	0	OR
0x00	0	Logic 0
0x00	1	Logic 1

It is possible (but not recommended) to select both the true and negated values of an input. When this is done, the gate output is zero, regardless of the other inputs, but may emit logic glitches (transient-induced pulses). If the output of the channel must be zero or one, the recommended method is to set all gate bits to zero and use the gate polarity bit to set the desired level.

Data gating is configured with the logic gate select registers as follows:

- Gate 1: CLCxGLS0 (Register 22-5)
- Gate 2: CLCxGLS1 (Register 22-6)
- Gate 3: CLCxGLS2 (Register 22-7)
- Gate 4: CLCxGLS3 (Register 22-8)

Register number suffixes are different than the gate numbers because other variations of this module have multiple gate selections in the same register.

Data gating is indicated in the right side of Figure 22-2. Only one gate is shown in detail. The remaining three gates are configured identically with the exception that the data enables correspond to the enables for that gate.

22.1.3 LOGIC FUNCTION

There are eight available logic functions including:

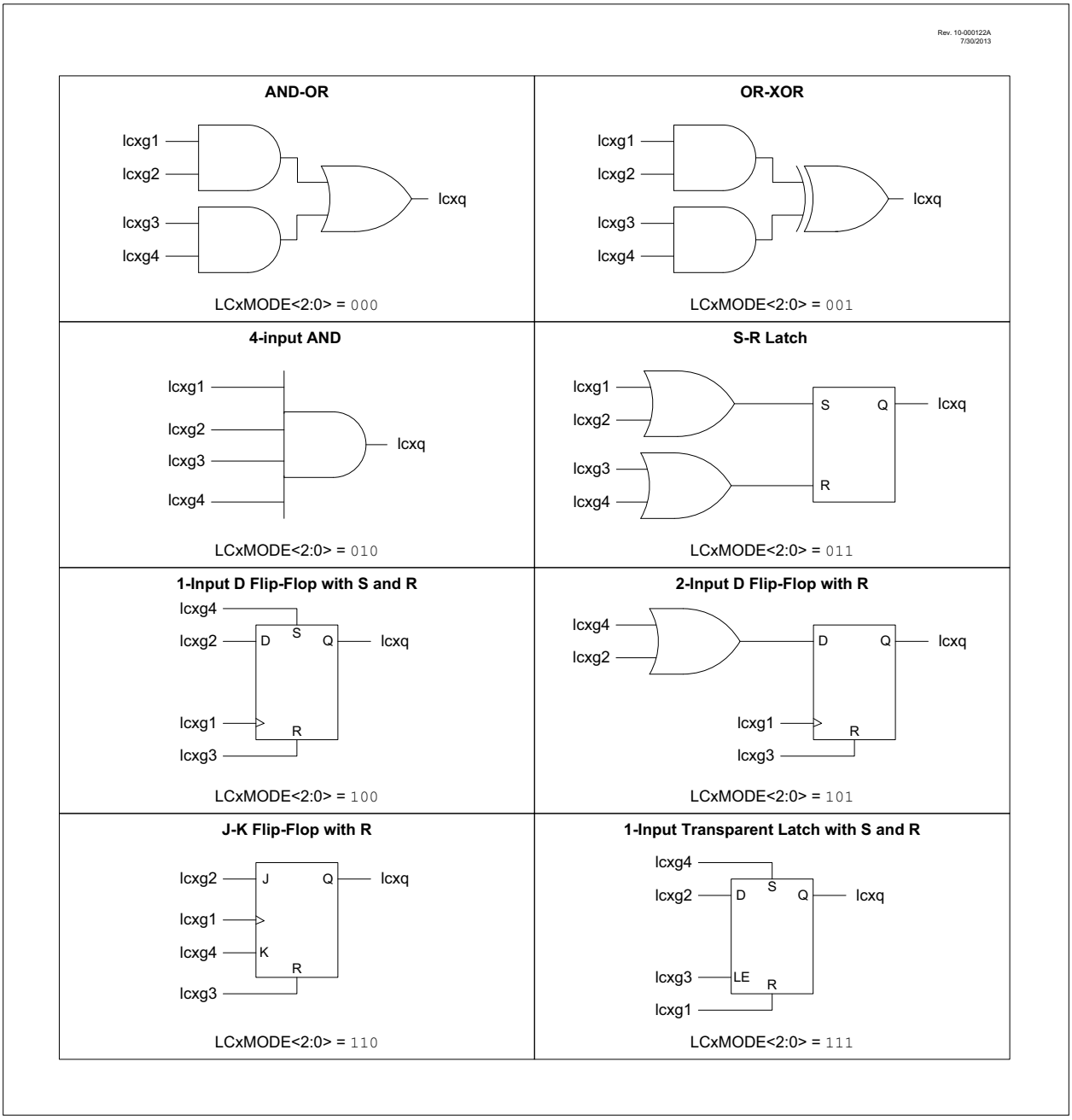
- AND-OR
- OR-XOR
- AND
- S-R Latch
- D Flip-Flop with Set and Reset
- D Flip-Flop with Reset
- J-K Flip-Flop with Reset
- Transparent Latch with Set and Reset

Logic functions are shown in Figure 22-3. Each logic function has four inputs and one output. The four inputs are the four data gate outputs of the previous stage. The output is fed to the inversion stage and from there to other peripherals, an output pin, and back to the CLCx itself.

22.1.4 OUTPUT POLARITY

The last stage in the configurable logic cell is the output polarity. Setting the LCxPOL bit of the CLCxCON register inverts the output signal from the logic stage. Changing the polarity while the interrupts are enabled will cause an interrupt for the resulting output transition.

FIGURE 22-3: PROGRAMMABLE LOGIC FUNCTIONS



REGISTER 22-5: CLCxGLS0: GATE 1 LOGIC SELECT REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
LCxG1D4T	LCxG1D4N	LCxG1D3T	LCxG1D3N	LCxG1D2T	LCxG1D2N	LCxG1D1T	LCxG1D1N
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 **LCxG1D4T:** Gate 1 Data 4 True (non-inverted) bit
1 = lcxd4T is gated into lcxg1
0 = lcxd4T is not gated into lcxg1

bit 6 **LCxG1D4N:** Gate 1 Data 4 Negated (inverted) bit
1 = lcxd4N is gated into lcxg1
0 = lcxd4N is not gated into lcxg1

bit 5 **LCxG1D3T:** Gate 1 Data 3 True (non-inverted) bit
1 = lcxd3T is gated into lcxg1
0 = lcxd3T is not gated into lcxg1

bit 4 **LCxG1D3N:** Gate 1 Data 3 Negated (inverted) bit
1 = lcxd3N is gated into lcxg1
0 = lcxd3N is not gated into lcxg1

bit 3 **LCxG1D2T:** Gate 1 Data 2 True (non-inverted) bit
1 = lcxd2T is gated into lcxg1
0 = lcxd2T is not gated into lcxg1

bit 2 **LCxG1D2N:** Gate 1 Data 2 Negated (inverted) bit
1 = lcxd2N is gated into lcxg1
0 = lcxd2N is not gated into lcxg1

bit 1 **LCxG1D1T:** Gate 1 Data 1 True (non-inverted) bit
1 = lcxd1T is gated into lcxg1
0 = lcxd1T is not gated into lcxg1

bit 0 **LCxG1D1N:** Gate 1 Data 1 Negated (inverted) bit
1 = lcxd1N is gated into lcxg1
0 = lcxd1N is not gated into lcxg1

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

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
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 = lcxd4T is gated into lcxg2
0 = lcxd4T is not gated into lcxg2

bit 6 **LCxG2D4N:** Gate 2 Data 4 Negated (inverted) bit
1 = lcxd4N is gated into lcxg2
0 = lcxd4N is not gated into lcxg2

bit 5 **LCxG2D3T:** Gate 2 Data 3 True (non-inverted) bit
1 = lcxd3T is gated into lcxg2
0 = lcxd3T is not gated into lcxg2

bit 4 **LCxG2D3N:** Gate 2 Data 3 Negated (inverted) bit
1 = lcxd3N is gated into lcxg2
0 = lcxd3N is not gated into lcxg2

bit 3 **LCxG2D2T:** Gate 2 Data 2 True (non-inverted) bit
1 = lcxd2T is gated into lcxg2
0 = lcxd2T is not gated into lcxg2

bit 2 **LCxG2D2N:** Gate 2 Data 2 Negated (inverted) bit
1 = lcxd2N is gated into lcxg2
0 = lcxd2N is not gated into lcxg2

bit 1 **LCxG2D1T:** Gate 2 Data 1 True (non-inverted) bit
1 = lcxd1T is gated into lcxg2
0 = lcxd1T is not gated into lcxg2

bit 0 **LCxG2D1N:** Gate 2 Data 1 Negated (inverted) bit
1 = lcxd1N is gated into lcxg2
0 = lcxd1N is not gated into lcxg2

REGISTER 23-6: NCOxINCL: NCOx INCREMENT REGISTER – LOW BYTE(1)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-1/1
NCOxINC<7:0>							
bit 7							bit 0

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

bit 7-0 **NCOxINC<7:0>:** NCOx Increment, Low Byte

Note 1: Write the NCOxINCH register first, then the NCOxINCL register. See **23.1.4 "Increment Registers"** for more information.

REGISTER 23-7: NCOxINCH: NCOx INCREMENT REGISTER – HIGH BYTE(1)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
NCOxINC<15:8>							
bit 7							bit 0

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

bit 7-0 **NCOxINC<15:8>:** NCOx Increment, High Byte

Note 1: Write the NCOxINCH register first, then the NCOxINCL register. See **23.1.4 "Increment Registers"** for more information.

TABLE 23-1: SUMMARY OF REGISTERS ASSOCIATED WITH NCOx

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON	CWG1BSEL	CWGA1SEL	—	—	T1GSEL	—	CLC1SEL	NCO1SEL	96
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	64
NCO1ACCH	NCO1ACC<15:8>								178
NCO1ACCL	NCO1ACC<7:0>								178
NCO1ACCU	—				NCO1ACC<19:16>				178
NCO1CLK	N1PWS<2:0>			—	—	—	N1CKS<1:0>		177
NCO1CON	N1EN	N1OE	N1OUT	N1POL	—	—	—	N1PFM	177
NCO1INCH	NCO1INC<15:8>								179
NCO1INCL	NCO1INC<7:0>								179
PIE2	—	—	C1IE	—	—	NCO1IE	—	—	66
PIR2	—	—	C1IF	—	—	NCO1IF	—	—	69
TRISA	—	—	TRISA5	TRISA4	—(1)	TRISA2	TRISA1	TRISA0	98

Legend: x = unknown, u = unchanged, – = unimplemented read as '0', q = value depends on condition. Shaded cells are not used for NCOx module.

Note 1: Unimplemented, read as '1'.

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

$$T_{DEADBAND\_UNCERTAINTY} = \frac{1}{Fcwg\_clock}$$

Example:

$$Fcwg\_clock = 16\ MHz$$

Therefore:

$$T_{DEADBAND\_UNCERTAINTY} = \frac{1}{Fcwg\_clock}$$

$$= \frac{1}{16\ MHz}$$

$$= 62.5ns$$

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.

REGISTER 24-3: CWGxCON2: CWG CONTROL REGISTER 2

R/W-0/0	R/W-0/0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
GxASE	GxARSEN	—	—	—	GxASDSC1	GxASDSFLT	GxASDSCLC2
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	q = Value depends on condition

bit 7 **GxASE:** Auto-Shutdown Event Status bit
1 = An auto-shutdown event has occurred
0 = No auto-shutdown event has occurred

bit 6 **GxARSEN:** Auto-Restart Enable bit
1 = Auto-restart is enabled
0 = Auto-restart is disabled

bit 5-3 **Unimplemented:** Read as ‘0’

bit 2 **GxASDSC1:** CWG Auto-shutdown on Comparator C1 Enable bit
1 = Shutdown when Comparator C1 output (C1OUT_async) is high
0 = Comparator C1 output has no effect on shutdown

bit 1 **GxASDSFLT:** CWG Auto-shutdown on FLT Enable bit
1 = Shutdown when $\overline{\text{CWG1FLT}}$ input is low
0 = $\overline{\text{CWG1FLT}}$ input has no effect on shutdown

bit 0 **GxASDSCLC2:** CWG Auto-shutdown on CLC2 Enable bit
1 = Shutdown when CLC2 output (LC2_out) is high
0 = CLC2 output has no effect on shutdown

MOVWI **Move W to INDFn**

Syntax: [*label*] MOVWI ++FSRn
[*label*] MOVWI --FSRn
[*label*] MOVWI FSRn++
[*label*] MOVWI FSRn--
[*label*] MOVWI k[FSRn]

Operands: n ∈ [0,1]
mm ∈ [00,01, 10, 11]
-32 ≤ k ≤ 31

Operation: W → INDFn
Effective address is determined by
- FSR + 1 (preincrement)
- FSR - 1 (predecrement)
- FSR + k (relative offset)

After the Move, the FSR value will be either:
- FSR + 1 (all increments)
- FSR - 1 (all decrements)

Unchanged

Status Affected: None

Mode	Syntax	mm
Preincrement	++FSRn	00
Predecrement	--FSRn	01
Postincrement	FSRn++	10
Postdecrement	FSRn--	11

Description: This instruction is used to move data between W and one of the indirect registers (INDFn). Before/after this move, the pointer (FSRn) is updated by pre/post incrementing/decrementing it.

Note: The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the FSRn.

FSRn is limited to the range 0000h - FFFFh. Incrementing/decrementing it beyond these bounds will cause it to wrap-around.

The increment/decrement operation on FSRn WILL NOT affect any Status bits.

NOP **No Operation**

Syntax: [*label*] NOP

Operands: None

Operation: No operation

Status Affected: None

Description: No operation.

Words: 1

Cycles: 1

Example:

```
NOP
```

OPTION **Load OPTION_REG Register with W**

Syntax: [*label*] OPTION

Operands: None

Operation: (W) → OPTION_REG

Status Affected: None

Description: Move data from W register to OPTION_REG register.

RESET **Software Reset**

Syntax: [*label*] RESET

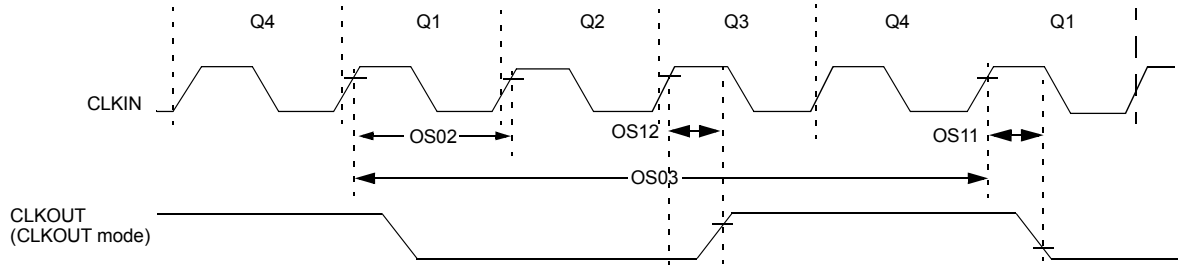
Operands: None

Operation: Execute a device Reset. Resets the nRI flag of the PCON register.

Status Affected: None

Description: This instruction provides a way to execute a hardware Reset by software.

FIGURE 27-5: CLOCK TIMING



Note: See Table 27-9.

TABLE 27-7: CLOCK OSCILLATOR TIMING REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
OS01	FOSC	External CLKIN Frequency(1)	DC	—	0.5	MHz	External Clock (ECL)
			DC	—	4	MHz	External Clock (ECM)
			DC	—	20	MHz	External Clock (ECH)
OS02	TOSC	External CLKIN Period(1)	50	—	∞	ns	External Clock (EC)
OS03	TCY	Instruction Cycle Time(1)	200	TCY	DC	ns	TCY = 4/FOSC

* 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: Instruction cycle period (TCY) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to CLKIN pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

FIGURE 28-21: IPD BASE, LOW-POWER SLEEP MODE, PIC12LF1501 ONLY

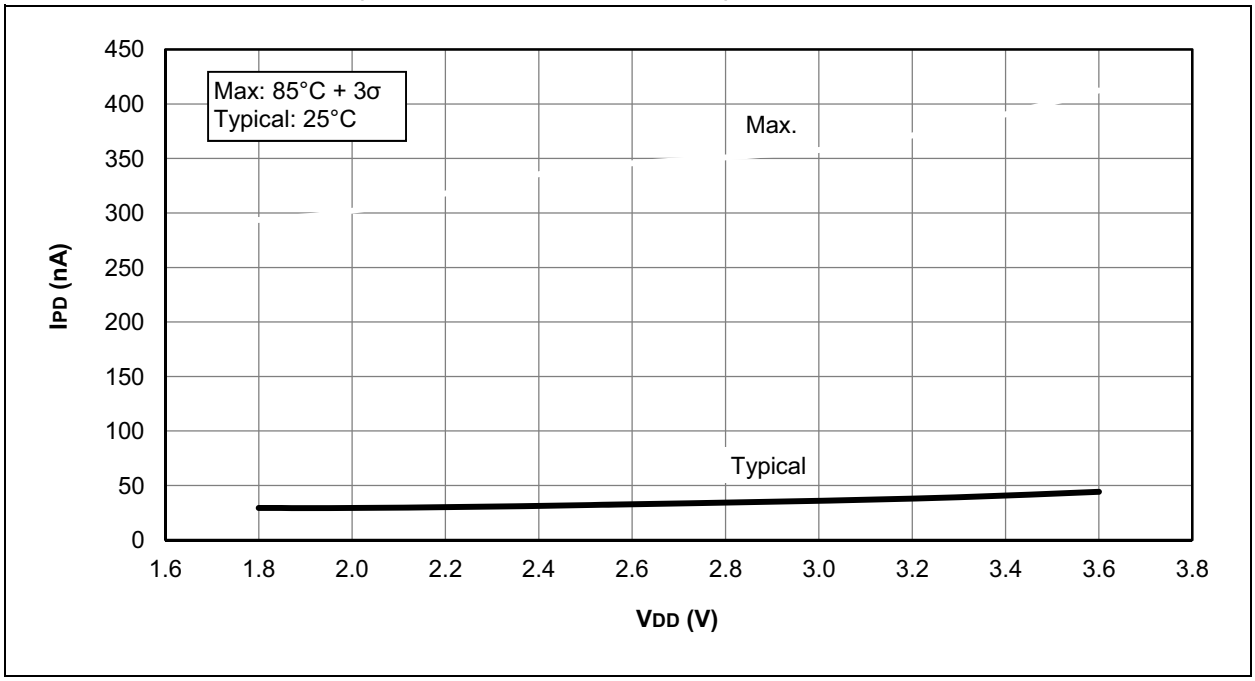


FIGURE 28-22: IPD BASE, LOW-POWER SLEEP MODE, VREGPM = 1, PIC12F1501 ONLY

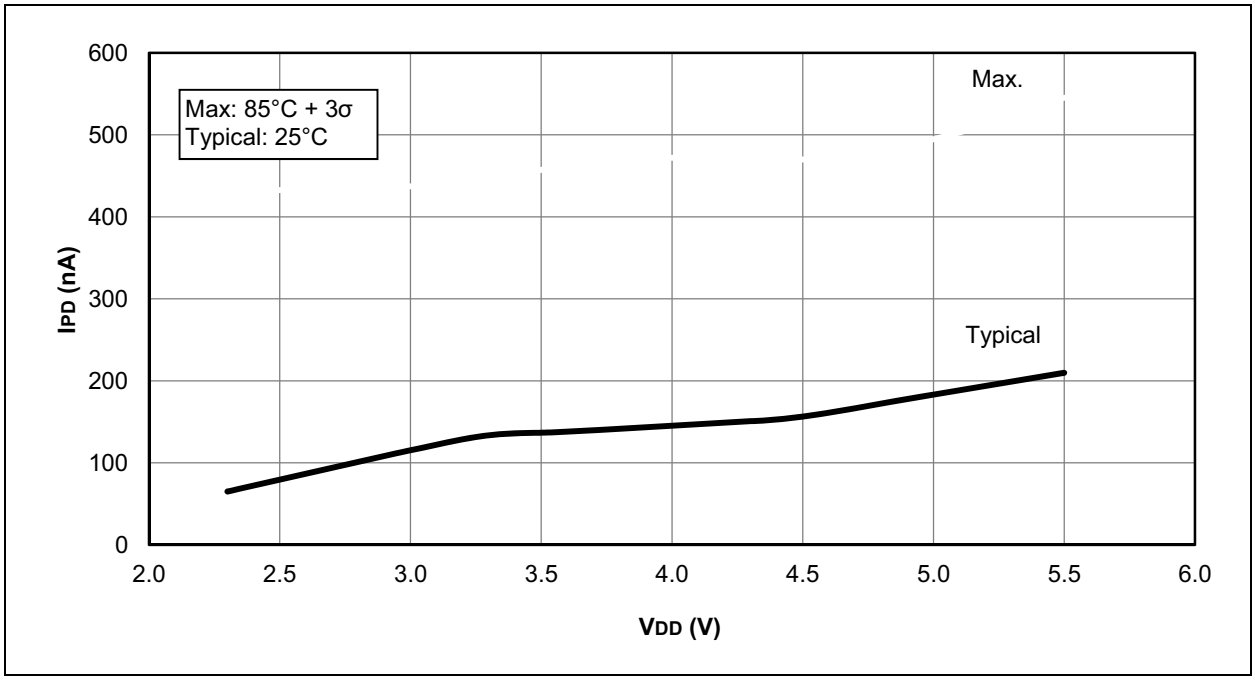


FIGURE 28-23: IPD, WATCHDOG TIMER (WDT), PIC12LF1501 ONLY

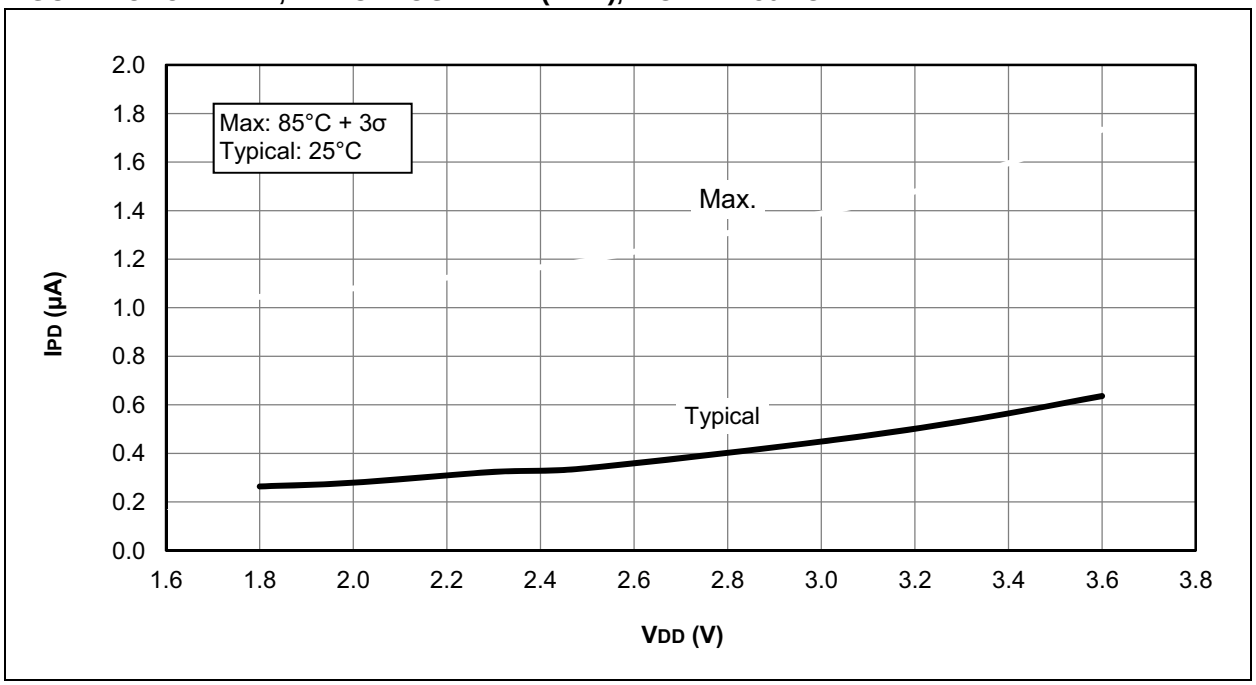


FIGURE 28-24: IPD, WATCHDOG TIMER (WDT), PIC12F1501 ONLY

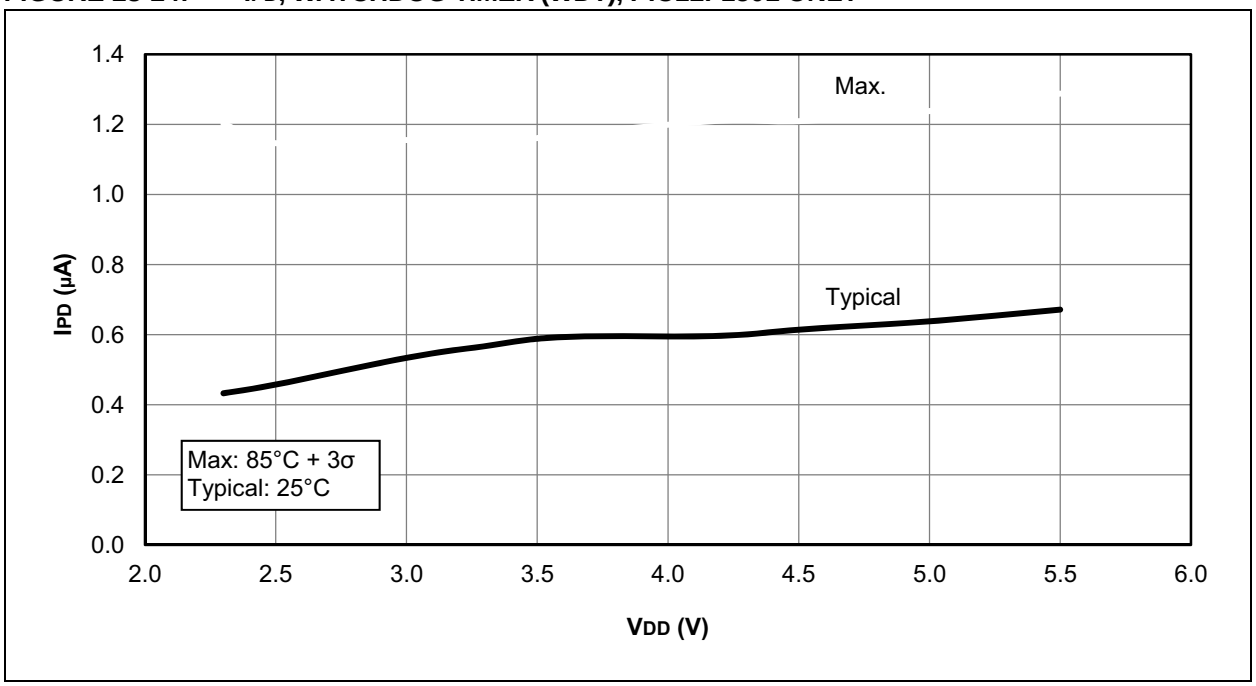
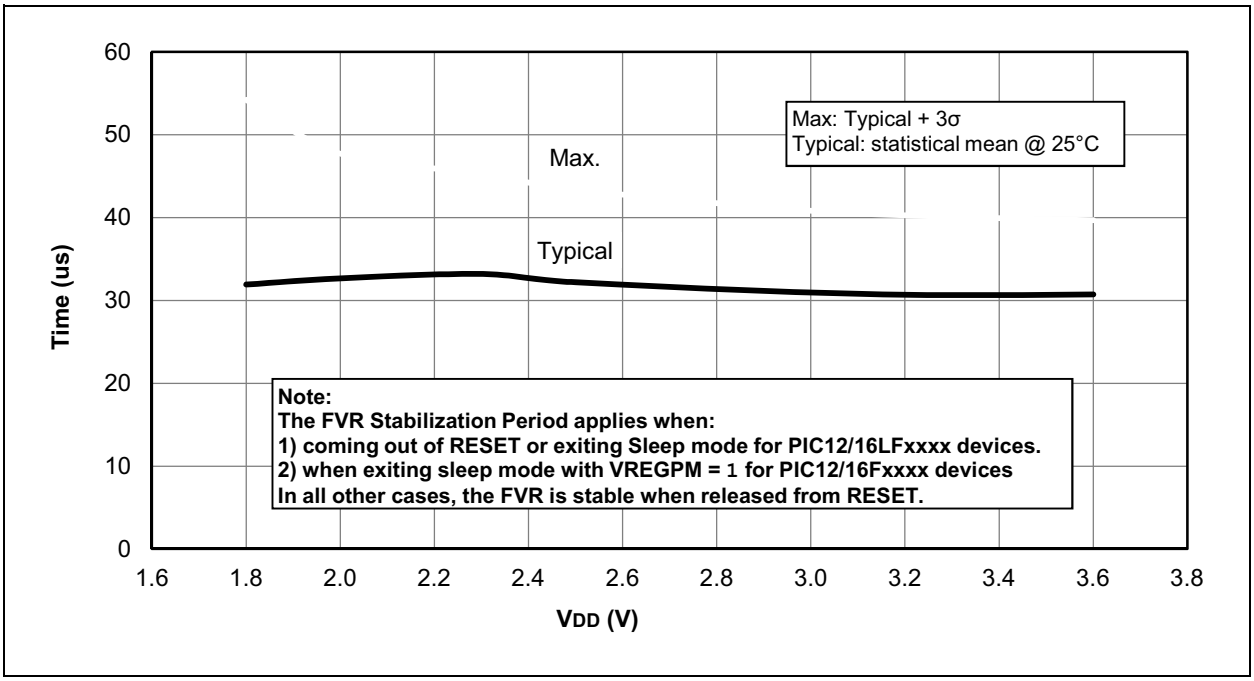
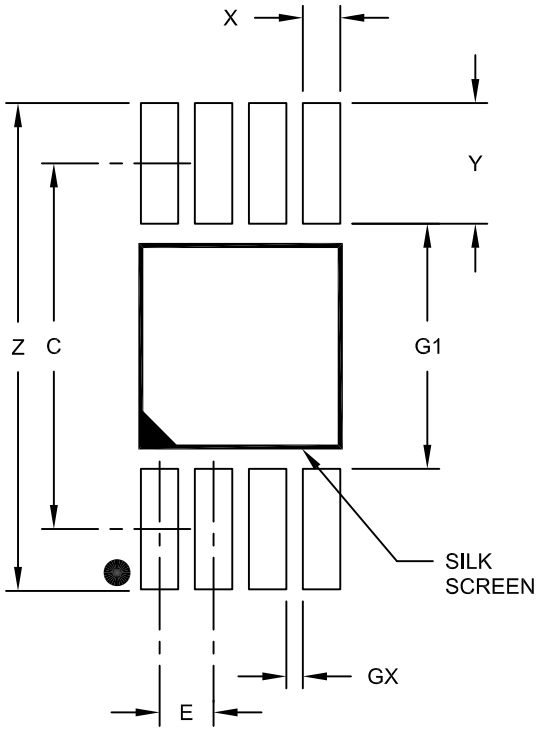


FIGURE 28-52: FVR STABILIZATION PERIOD



8-Lead Plastic Micro Small Outline Package (MS) [MSOP]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Contact Pad Spacing	C		4.40	
Overall Width	Z			5.85
Contact Pad Width (X8)	X1			0.45
Contact Pad Length (X8)	Y1			1.45
Distance Between Pads	G1	2.95		
Distance Between Pads	GX	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2111A