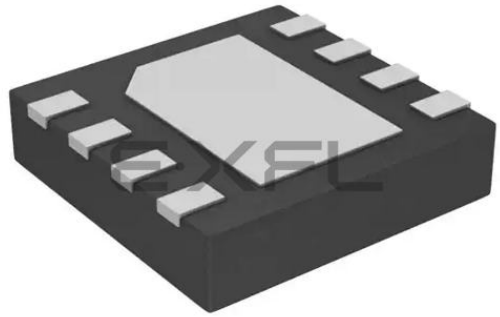


Welcome to E-XFL.COM

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	32MHz
Connectivity	LINbus, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	6
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 4x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	8-UDFN Exposed Pad
Supplier Device Package	8-UDFN (3x3)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12lf1572t-i-rf

TABLE 1-2: PIC12(L)F1571/2 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0/C1IN+/DACOUT/ TX(2)/CK(2)/CWG1B/PWM2/ ICSPDAT/ICDDAT	RA0	(3)	(4)	General purpose I/O.
	AN0			ADC channel input.
	C1IN+			Comparator positive input.
	DACOUT			Digital-to-Analog Converter output.
	TX			USART asynchronous transmit.
	CK			USART synchronous clock.
	CWG1B			CWG complementary output.
	PWM2			PWM output.
	ICSPDAT			ICSP™ data I/O.
	ICDDAT			In-circuit debug data.
RA1/AN1/VREF+/C1IN0-/RX(2)/ DT(2)/PWM1/ICSPCLK/ICDCLK	RA1	(3)	(4)	General purpose I/O.
	AN1			ADC channel input.
	VREF+			ADC Voltage Reference input.
	C1IN0-			Comparator negative input.
	RX			USART asynchronous input.
	DT			USART synchronous data.
	PWM1			PWM output.
	ICSPCLK			ICSP programming clock.
	ICDCLK			In-circuit debug clock.
RA2/AN2/C1OUT/T0CKI/ $\overline{\text{CWG1FLT}}$/CWG1A/PWM3/INT	RA2	(3)	(4)	General purpose I/O.
	AN2			ADC channel input.
	C1OUT			Comparator output.
	T0CKI			Timer0 clock input.
	CWG1FLT			Complementary Waveform Generator Fault input.
	CWG1A			CWG complementary output.
	PWM3			PWM output.
	INT			External interrupt.
RA3/VPP/T1G(1)/$\overline{\text{MCLR}}$	RA3	(3)	(4)	General purpose input with IOC and WPU.
	VPP			Programming voltage.
	T1G			Timer1 gate input.
	$\overline{\text{MCLR}}$			Master Clear with internal pull-up.
RA4/AN3/C1IN1-/T1G/TX(1,2)/ CK(1,2)/CWG1B(1)/PWM2(1)/ CLKOUT	RA4	(3)	(4)	General purpose I/O.
	AN3			ADC channel input.
	C1IN1-			Comparator negative input.
	T1G			Timer1 gate input.
	TX			USART asynchronous transmit.
	CK			USART synchronous clock.
	CWG1B			CWG complementary output.
	PWM2			PWM output.
	CLKOUT			FOSC/4 output.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C = Schmitt Trigger input with I^2C levels
HV = High Voltage XTAL = Crystal

Note 1: Alternate pin function selected with the APFCON (Register 11-1) register.
2: PIC12(L)F1572 only.
3: Input type is selected by the port.
4: Output type is selected by the port.

3.0 MEMORY ORGANIZATION

These devices contain the following types of memory:

- Program Memory:
  - Configuration Words
  - Device ID
  - User ID
  - Flash Program Memory
- Data Memory:
  - Core Registers
  - Special Function Registers
  - General Purpose RAM
  - Common RAM

The following features are associated with access and control of program memory and data memory:

- PCL and PCLATH
- Stack
- Indirect Addressing

3.1 Program Memory Organization

The enhanced mid-range core has a 15-bit Program Counter (PC) capable of addressing a 32K x 14 program memory space. Table 3-1 shows the memory sizes implemented. Accessing a location above these boundaries will cause a wraparound within the implemented memory space. The Reset vector is at 0000h and the interrupt vector is at 0004h (see Figure 3-1).

3.2 High-Endurance Flash

This device has a 128-byte section of high-endurance Program Flash Memory (PFM) in lieu of data EEPROM. This area is especially well-suited for non-volatile data storage that is expected to be updated frequently over the life of the end product. See **Section 10.2 "Flash Program Memory Overview"** for more information on writing data to PFM. See **Section 3.2.1.2 "Indirect Read with FSR"** for more information about using the FSR registers to read byte data stored in PFM.

TABLE 3-1: DEVICE SIZES AND ADDRESSES

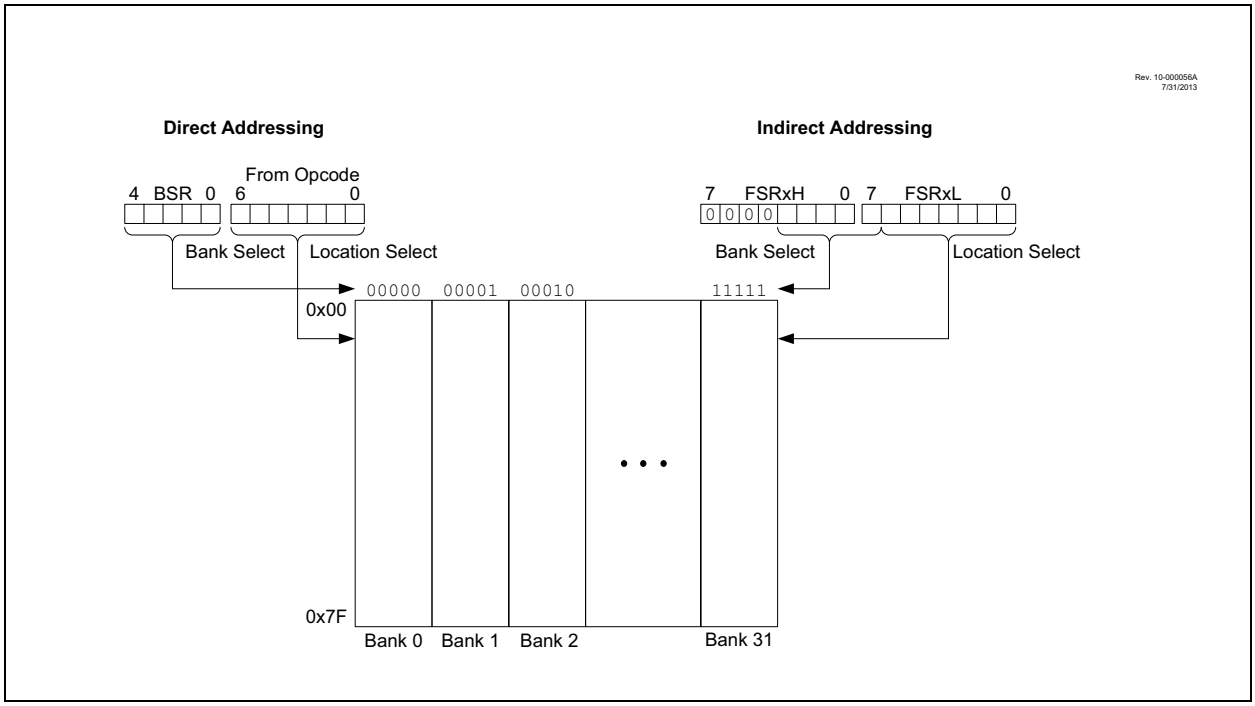
Device	Program Memory Space (Words)	Last Program Memory Address	High-Endurance Flash Memory Address Range(1)
PIC12(L)F1571	1,024	03FFh	0380h-03FFh
PIC12(L)F1572	2,048	07FFh	0780h-07FFh

Note 1: High-endurance Flash applies to the low byte of each address in the range.

3.6.1 TRADITIONAL DATA MEMORY

The traditional data memory is a region from FSR address, 0x000, to FSR address, 0xFFF. The addresses correspond to the absolute addresses of all SFR, GPR and common registers.

FIGURE 3-10: TRADITIONAL DATA MEMORY MAP



6.0 RESETS

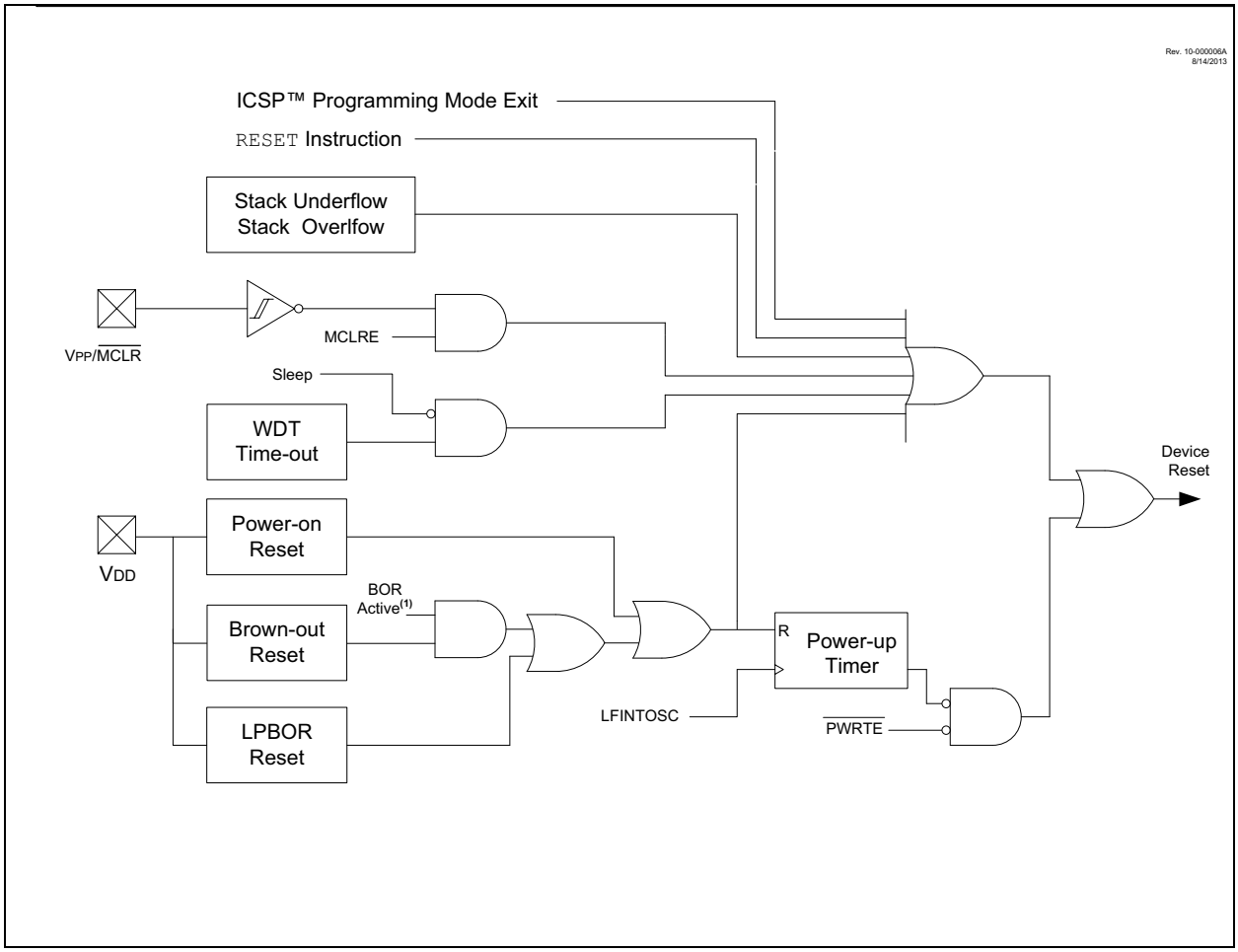
There are multiple ways to reset this device:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- Low-Power Brown-out Reset (LPBOR)
- $\overline{\text{MCLR}}$ Reset
- WDT Reset
- RESET instruction
- Stack Overflow
- Stack Underflow
- Programming mode exit

To allow V_{DD} to stabilize, an optional Power-up Timer can be enabled to extend the Reset time after a BOR or POR event.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 6-1.

FIGURE 6-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



NOTES:

EXAMPLE 10-3: WRITING TO FLASH PROGRAM MEMORY

```
; This write routine assumes the following:
; 1. 32 bytes of data are loaded, starting at the address in DATA_ADDR
; 2. Each word of data to be written is made up of two adjacent bytes in DATA_ADDR,
; stored in little endian format
; 3. A valid starting address (the Least Significant bits = 00000) is loaded in ADDRH:ADDRL
; 4. ADDRH and ADDRL are located in shared data memory 0x70 - 0x7F (common RAM)
;
        BCF         INTCON,GIE     ; Disable ints so required sequences will execute properly
        BANKSEL     PMADRH         ; Bank 3
        MOVF        ADDRH,W        ; Load initial address
        MOVWF       PMADRH         ;
        MOVF        ADDRL,W        ;
        MOVWF       PMADRL         ;
        MOVLW       LOW DATA_ADDR  ; Load initial data address
        MOVWF       FSR0L          ;
        MOVLW       HIGH DATA_ADDR ; Load initial data address
        MOVWF       FSR0H          ;
        BCF         PMCON1,CFGS    ; Not configuration space
        BSF         PMCON1,WREN    ; Enable writes
        BSF         PMCON1,LWLO    ; Only Load Write Latches
LOOP
        MOVIW       FSR0++         ; Load first data byte into lower
        MOVWF       PMDATL         ;
        MOVIW       FSR0++         ; Load second data byte into upper
        MOVWF       PMDATH         ;

        MOVF        PMADRL,W       ; Check if lower bits of address are '00000'
        XORLW       0x1F           ; Check if we're on the last of 16 addresses
        ANDLW       0x1F           ;
        BTFSC       STATUS,Z       ; Exit if last of 16 words,
        GOTO        START_WRITE    ;

  Required Sequence
        MOVLW       55h            ; Start of required write sequence:
        MOVWF       PMCON2         ; Write 55h
        MOVLW       0AAh           ;
        MOVWF       PMCON2         ; Write AAh
        BSF         PMCON1,WR      ; Set WR bit to begin write
        NOP                        ; NOP instructions are forced as processor
                                   ; loads program memory write latches
        NOP                        ;

        INCF        PMADRL,F       ; Still loading latches Increment address
        GOTO        LOOP           ; Write next latches

START_WRITE
        BCF         PMCON1,LWLO    ; No more loading latches - Actually start Flash program
                                   ; memory write

  Required Sequence
        MOVLW       55h            ; Start of required write sequence:
        MOVWF       PMCON2         ; Write 55h
        MOVLW       0AAh           ;
        MOVWF       PMCON2         ; Write AAh
        BSF         PMCON1,WR      ; Set WR bit to begin write
        NOP                        ; NOP instructions are forced as processor writes
                                   ; all the program memory write latches simultaneously
        NOP                        ; to program memory.
                                   ; After NOPs, the processor
                                   ; stalls until the self-write process in complete
                                   ; after write processor continues with 3rd instruction
        BCF         PMCON1,WREN    ; Disable writes
        BSF         INTCON,GIE     ; Enable interrupts
```

TABLE 15-1: ADC CLOCK PERIOD (T_{AD}) VS. DEVICE OPERATING FREQUENCIES

ADC Clock Period (T_{AD})		Device Frequency (F_{OSC})				
ADC Clock Source	ADCS<2:0>	20 MHz	16 MHz	8 MHz	4 MHz	1 MHz
Fosc/2	000	100 ns	125 ns	250 ns	500 ns	2.0 μs
Fosc/4	100	200 ns	250 ns	500 ns	1.0 μs	4.0 μs
Fosc/8	001	400 ns	500 ns	1.0 μs	2.0 μs	8.0 μs
Fosc/16	101	800 ns	1.0 μs	2.0 μs	4.0 μs	16.0 μs
Fosc/32	010	1.6 μs	2.0 μs	4.0 μs	8.0 μs	32.0 μs
Fosc/64	110	3.2 μs	4.0 μs	8.0 μs	16.0 μs	64.0 μs
FRC	x11	1.0-6.0 μs	1.0-6.0 μs	1.0-6.0 μs	1.0-6.0 μs	1.0-6.0 μs

Legend: Shaded cells are outside of recommended range.

Note: The T_{AD} period when using the FRC clock source can fall within a specified range (see T_{AD} parameter). The T_{AD} period when using the F_{OSC}-based clock source can be configured for a more precise T_{AD} period. However, the FRC clock source must be used when conversions are to be performed with the device in Sleep mode.

FIGURE 15-2: ANALOG-TO-DIGITAL CONVERSION T_{AD} CYCLES

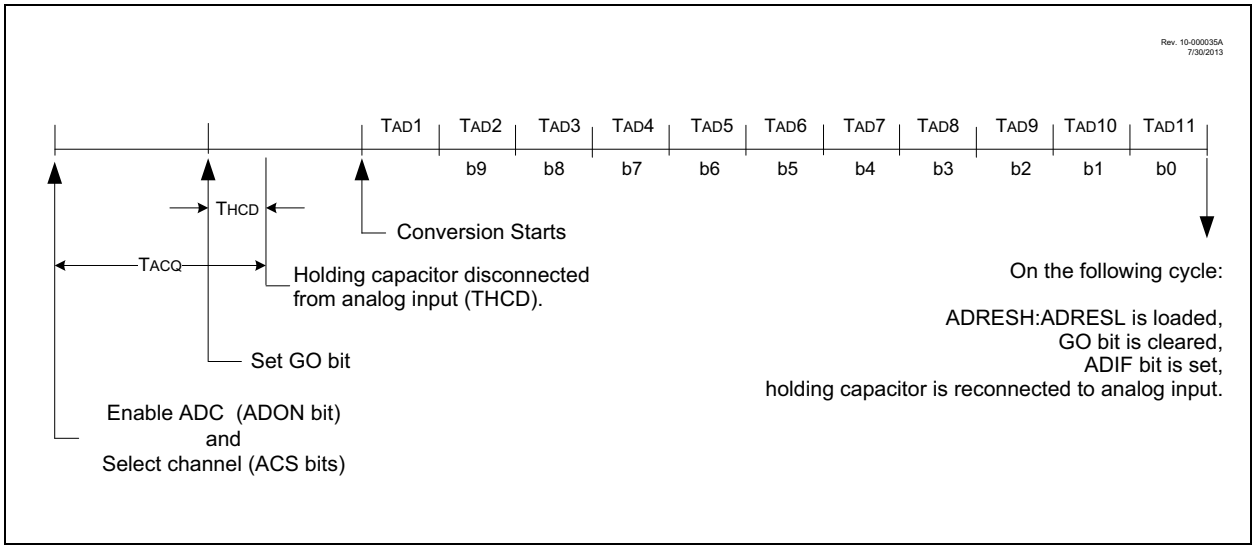


TABLE 17-3: SUMMARY OF REGISTERS ASSOCIATED WITH COMPARATOR MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	—	ANSA4	—	ANSA<2:0>			114
CM1CON0	C1ON	C1OUT	C1OE	C1POL	—	C1SP	C1HYS	C1SYNC	151
CM1CON1	C1NTP	C1INTN	C1PCH<1:0>		—	C1NCH<2:0>			152
CMOUT	—	—	—	—	—	—	—	MC1OUT	152
DAC1CON0	DACEN	—	DACOE	—	DACPSS<1:0>		—	—	145
DAC1CON1	—	—	—	DACR<4:0>					145
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		125
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	74
PIE2	—	—	C1IE	—	—	—	—	—	76
PIR2	—	—	C1IF	—	—	—	—	—	79
PORTA	—	—	RA5	RA4	RA3	RA<2:0>			113
LATA	—	—	LATA5	LATA4	—	LATA<2:0>			114
TRISA	—	—	TRISA5	TRISA4	—(1)	TRISA<2:0>			113

Legend: — = unimplemented location, read as '0'. Shaded cells are unused by the comparator module.

Note 1: Unimplemented, read as '1'.

19.3 Timer1 Prescaler

Timer1 has four prescaler options, allowing 1, 2, 4 or 8 divisions of the clock input. The T1CKPSx bits of the T1CON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

19.4 Timer1 Operation in Asynchronous Counter Mode

If control bit, $\overline{\text{T1SYNC}}$, of the T1CON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If the external clock source is selected then the timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see **Section 19.4.1 "Reading and Writing Timer1 in Asynchronous Counter Mode"**).

Note:	When switching from synchronous to asynchronous operation, it is possible to skip an increment. When switching from asynchronous to synchronous operation, it is possible to produce an additional increment.

19.4.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMR1H:TMR1L register pair.

19.5 Timer1 Gate

Timer1 can be configured to count freely or the count can be enabled and disabled using Timer1 gate circuitry. This is also referred to as Timer1 Gate Enable.

Timer1 gate can also be driven by multiple selectable sources.

19.5.1 TIMER1 GATE ENABLE

The Timer1 Gate Enable mode is enabled by setting the TMR1GE bit of the T1GCON register. The polarity of the Timer1 Gate Enable mode is configured using the T1GPOL bit of the T1GCON register.

When Timer1 Gate Enable mode is enabled, Timer1 will increment on the rising edge of the Timer1 clock source. When Timer1 Gate Enable mode is disabled, no incrementing will occur and Timer1 will hold the current count. See Figure 19-3 for timing details.

TABLE 19-3: TIMER1 GATE ENABLE SELECTIONS

T1CLK	T1GPOL	T1G	Timer1 Operation
↑	0	0	Counts
↑	0	1	Holds Count
↑	1	0	Holds Count
↑	1	1	Counts

19.5.2 TIMER1 GATE SOURCE SELECTION

Timer1 gate source selections are shown in Table 19-4. Source selection is controlled by the T1GSS<1:0> bits of the T1GCON register. The polarity for each available source is also selectable. Polarity selection is controlled by the T1GPOL bit of the T1GCON register.

TABLE 19-4: TIMER1 GATE SOURCES

T1GSS<1:0>	Timer1 Gate Source
00	Timer1 Gate Pin (T1G)
01	Overflow of Timer0 (T0_overflow) (TMR0 increments from FFh to 00h)
10	Comparator 1 Output (C1OUT_sync)(1)
11	Reserved

Note 1: Optionally synchronized comparator output.

REGISTER 21-2: RCSTA: RECEIVE STATUS AND CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-0/0
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit 7							bit 0

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

bit 7 **SPEN:** Serial Port Enable bit

1 = Serial port is enabled (configures RX/DT and TX/CK pins as serial port pins)
0 = Serial port is disabled (held in Reset)

bit 6 **RX9:** 9-Bit Receive Enable bit

1 = Selects 9-bit reception
0 = Selects 8-bit reception

bit 5 **SREN:** Single Receive Enable bit

Asynchronous mode:
Don’t care.

Synchronous mode – Master:
1 = Enables single receive
0 = Disables single receive
This bit is cleared after reception is complete.

Synchronous mode – Slave:
Don’t care.

bit 4 **CREN:** Continuous Receive Enable bit

Asynchronous mode:
1 = Enables receiver
0 = Disables receiver

Synchronous mode:
1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)
0 = Disables continuous receive

bit 3 **ADDEN:** Address Detect Enable bit

Asynchronous mode 9-bit (RX9 = 1):
1 = Enables address detection, enables interrupt and loads the receive buffer when RSR<8> is set
0 = Disables address detection, all bytes are received and ninth bit can be used as parity bit

Asynchronous mode 8-bit (RX9 = 0):
Don’t care.

bit 2 **FERR:** Framing Error bit

1 = Framing error (can be updated by reading RCREG register and receiving next valid byte)
0 = No framing error

bit 1 **OERR:** Overrun Error bit

1 = Overrun error (can be cleared by clearing bit, CREN)
0 = No overrun error

bit 0 **RX9D:** Ninth Bit of Received Data bit

This can be address/data bit or a parity bit and must be calculated by user firmware.

21.5 EUSART Synchronous Mode

Synchronous serial communications are typically used in systems with a single master and one or more slaves. The master device contains the necessary circuitry for baud rate generation and supplies the clock for all devices in the system. Slave devices can take advantage of the master clock by eliminating the internal clock generation circuitry.

There are two signal lines in Synchronous mode: a bidirectional data line and a clock line. Slaves use the external clock supplied by the master to shift the serial data into and out of their respective Receive and Transmit Shift registers. Since the data line is bidirectional, synchronous operation is half-duplex only. Half-duplex refers to the fact that master and slave devices can receive and transmit data but not both simultaneously. The EUSART can operate as either a master or slave device.

Start and Stop bits are not used in synchronous transmissions.

21.5.1 SYNCHRONOUS MASTER MODE

The following bits are used to configure the EUSART for synchronous master operation:

- SYNC = 1
- CSRC = 1
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Setting the CSRC bit of the TXSTA register configures the device as a master. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the EUSART.

21.5.1.1 Master Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a master transmits the clock on the TX/CK line. The TX/CK pin output driver is automatically enabled when the EUSART is configured for synchronous transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One clock cycle is generated for each data bit. Only as many clock cycles are generated as there are data bits.

21.5.1.2 Clock Polarity

A clock polarity option is provided for Microwire compatibility. Clock polarity is selected with the SCKP bit of the BAUDCON register. Setting the SCKP bit sets the clock Idle state as high. When the SCKP bit is set, the data changes on the falling edge of each clock. Clearing the SCKP bit sets the Idle state as low. When the SCKP bit is cleared, the data changes on the rising edge of each clock.

21.5.1.3 Synchronous Master Transmission

Data is transferred out of the device on the RX/DT pin. The RX/DT and TX/CK pin output drivers are automatically enabled when the EUSART is configured for synchronous master transmit operation.

A transmission is initiated by writing a character to the TXREG register. If the TSR still contains all or part of a previous character the new character data is held in the TXREG until the last bit of the previous character has been transmitted. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TXREG is immediately transferred to the TSR. The transmission of the character commences immediately following the transfer of the data to the TSR from the TXREG.

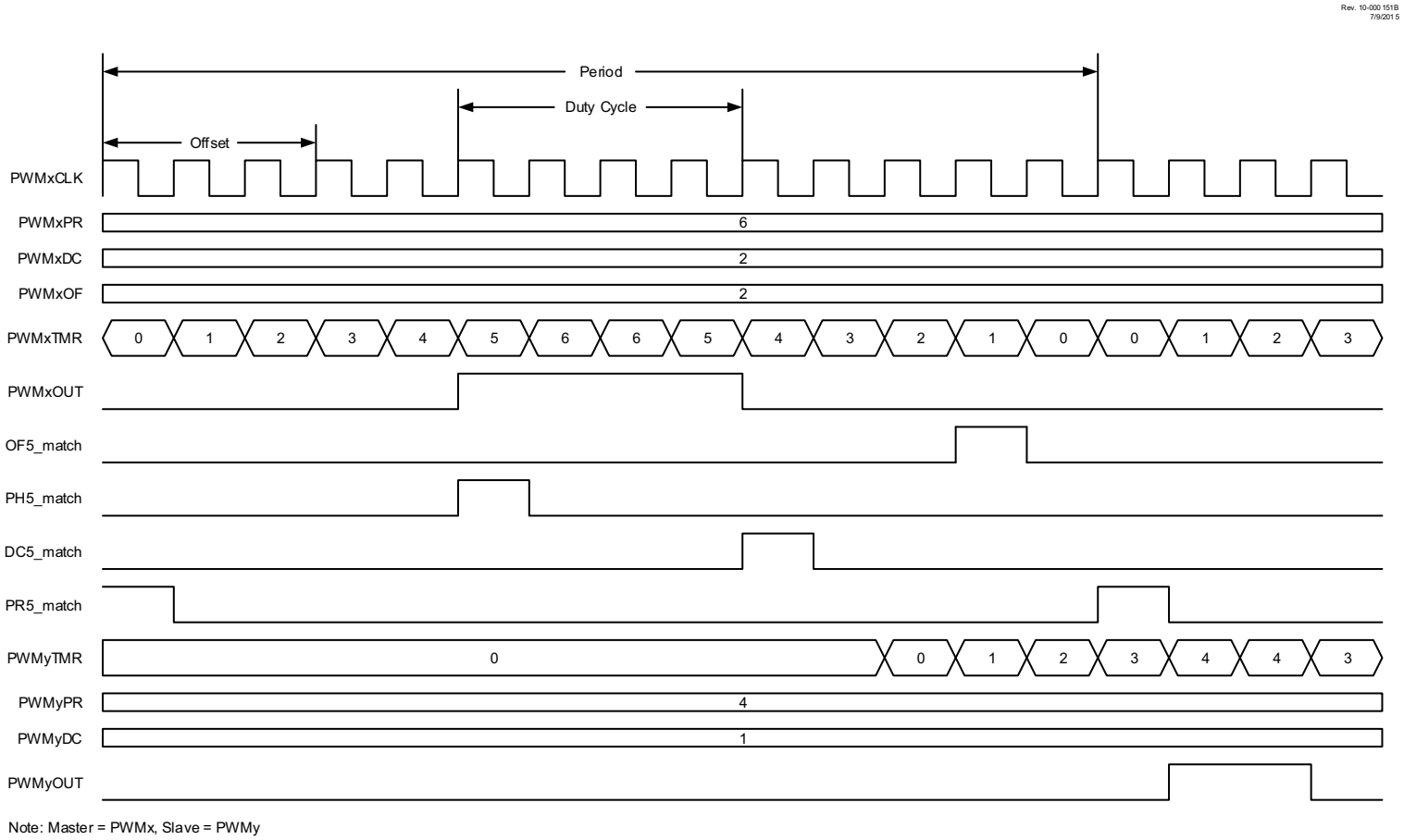
Each data bit changes on the leading edge of the master clock and remains valid until the subsequent leading clock edge.

Note:	The TSR register is not mapped in data memory, so it is not available to the user.

21.5.1.4 Synchronous Master Transmission Setup

1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 21.4 “EUSART Baud Rate Generator (BRG)”**).
2. Enable the synchronous master serial port by setting bits, SYNC, SPEN and CSRC.
3. Disable Receive mode by clearing bits, SREN and CREN.
4. Enable Transmit mode by setting the TXEN bit.
5. If 9-bit transmission is desired, set the TX9 bit.
6. If interrupts are desired, set the TXIE bit of the PIE1 register, and the GIE and PEIE bits of the INTCON register.
7. If 9-bit transmission is selected, the ninth bit should be loaded in the TX9D bit.
8. Start transmission by loading data to the TXREG register.

FIGURE 22-13: OFFSET MATCH ON DECREMENTING TIMER TIMING DIAGRAM



REGISTER 22-9: PWMxDCH: PWMx DUTY CYCLE COUNT HIGH 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
DC<15:8>							
bit 7							bit 0

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

bit 7-0 **DC<15:8>**: PWMx Duty Cycle High bits
Upper eight bits of PWM duty cycle count.

REGISTER 22-10: PWMxDCL: PWMx DUTY CYCLE COUNT LOW 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
DC<7:0>							
bit 7							bit 0

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

bit 7-0 **DC<7:0>**: PWMx Duty Cycle Low bits
Lower eight bits of PWM duty cycle count.

FIGURE 27-5: IDD TYPICAL, EC OSCILLATOR, MEDIUM POWER MODE, PIC12LF1571/2 ONLY

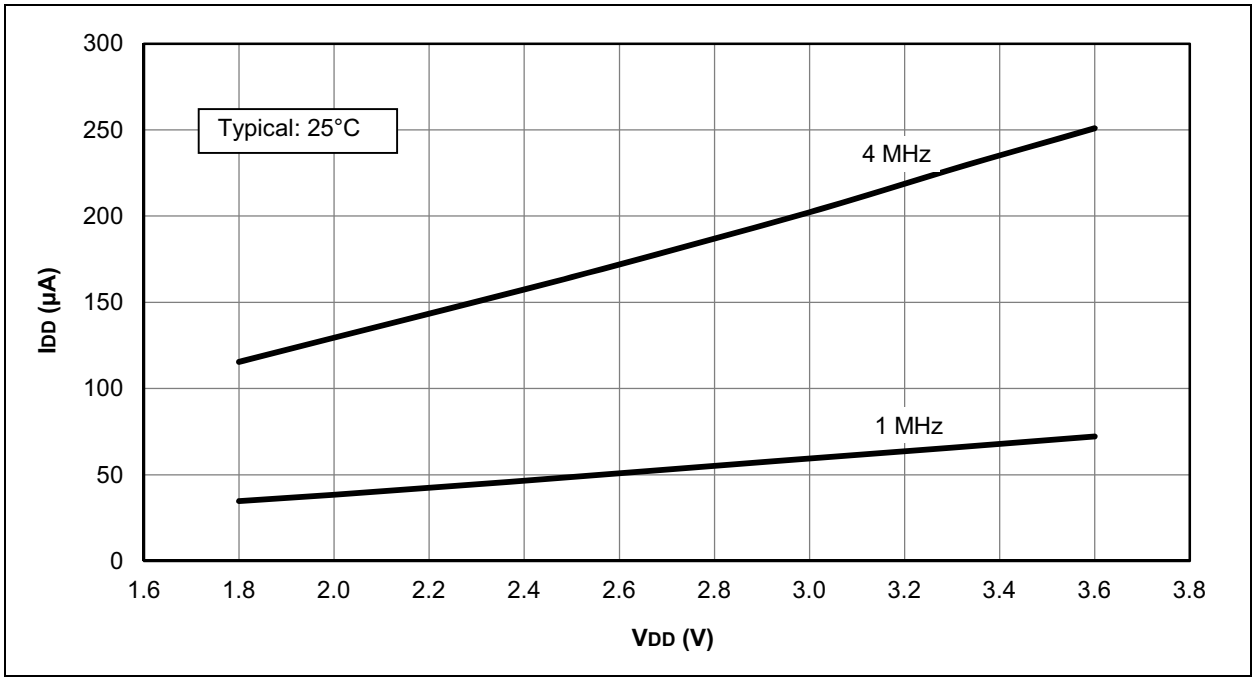


FIGURE 27-6: IDD MAXIMUM, EC OSCILLATOR, MEDIUM POWER MODE, PIC12LF1571/2 ONLY

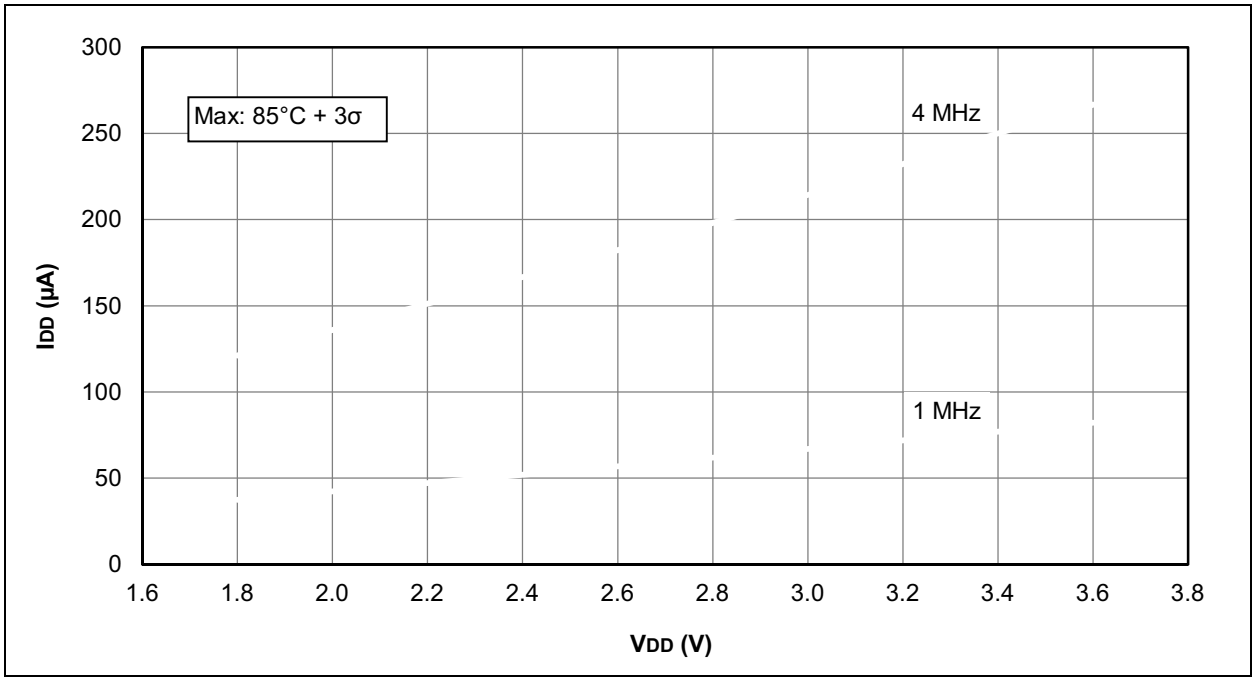


FIGURE 27-15: IDD, MFINTOSC, FOSC = 500 kHz, PIC12LF1571/2 ONLY

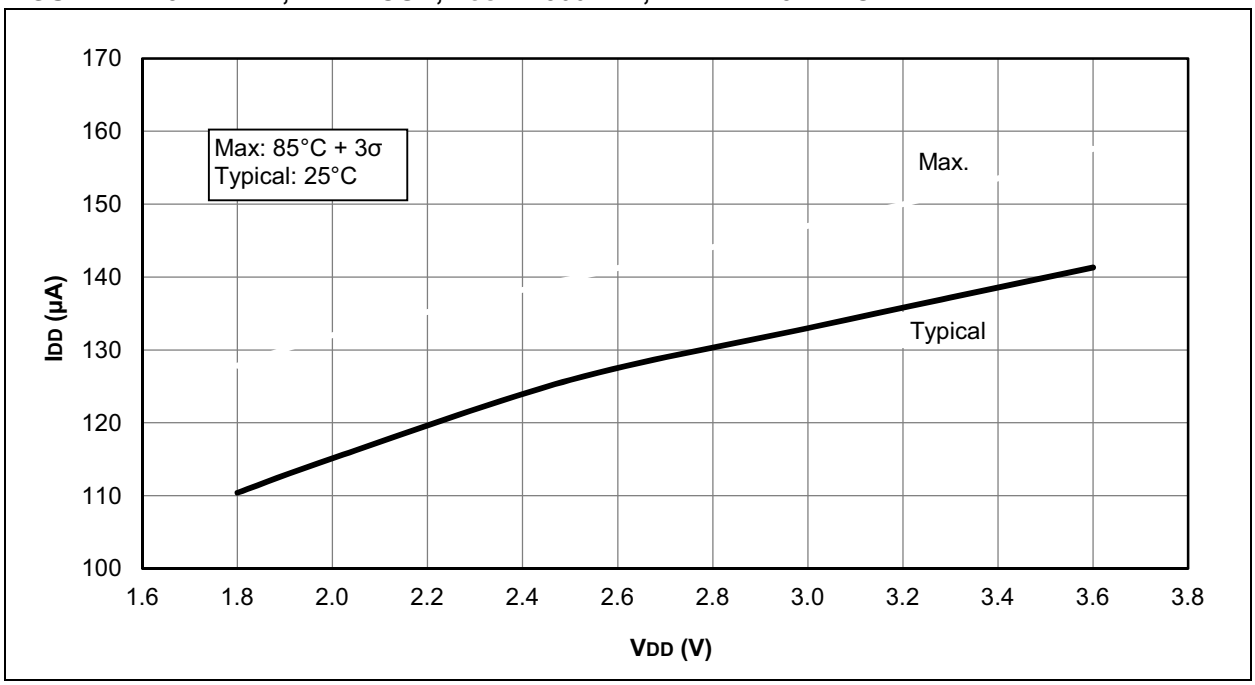


FIGURE 27-16: IDD, MFINTOSC, FOSC = 500 kHz, PIC12F1571/2 ONLY

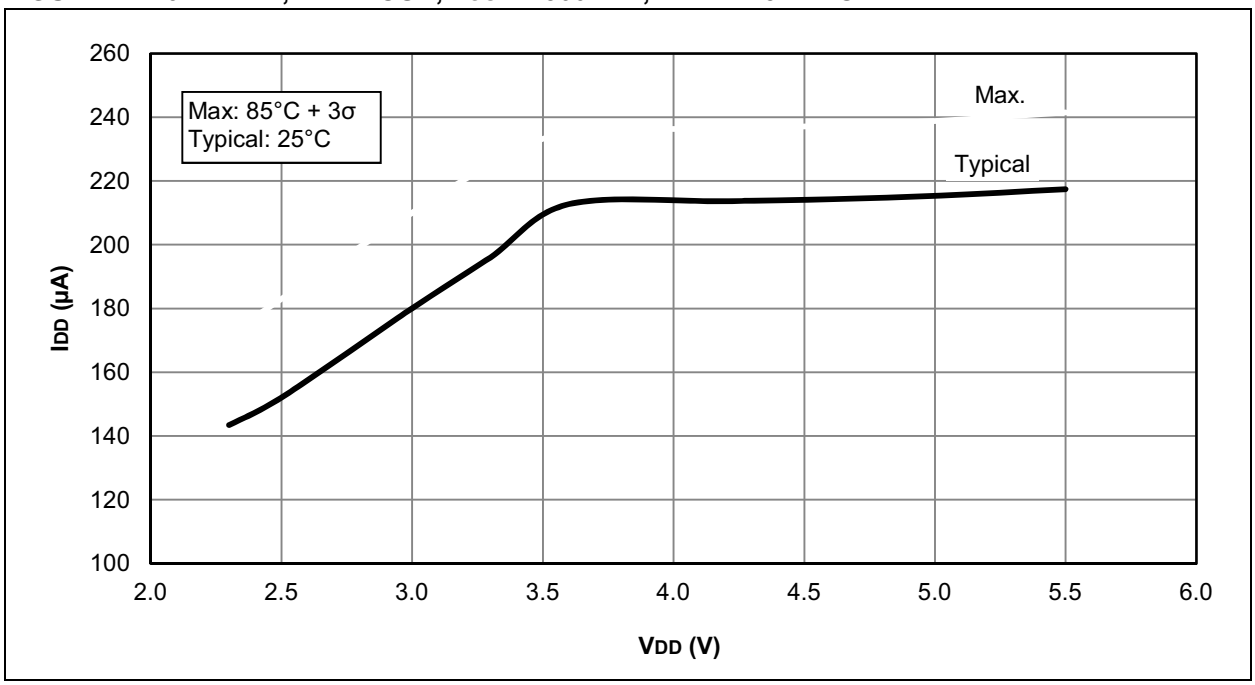


FIGURE 27-19: IDD TYPICAL, HFINTOSC, PIC12F1571/2 ONLY

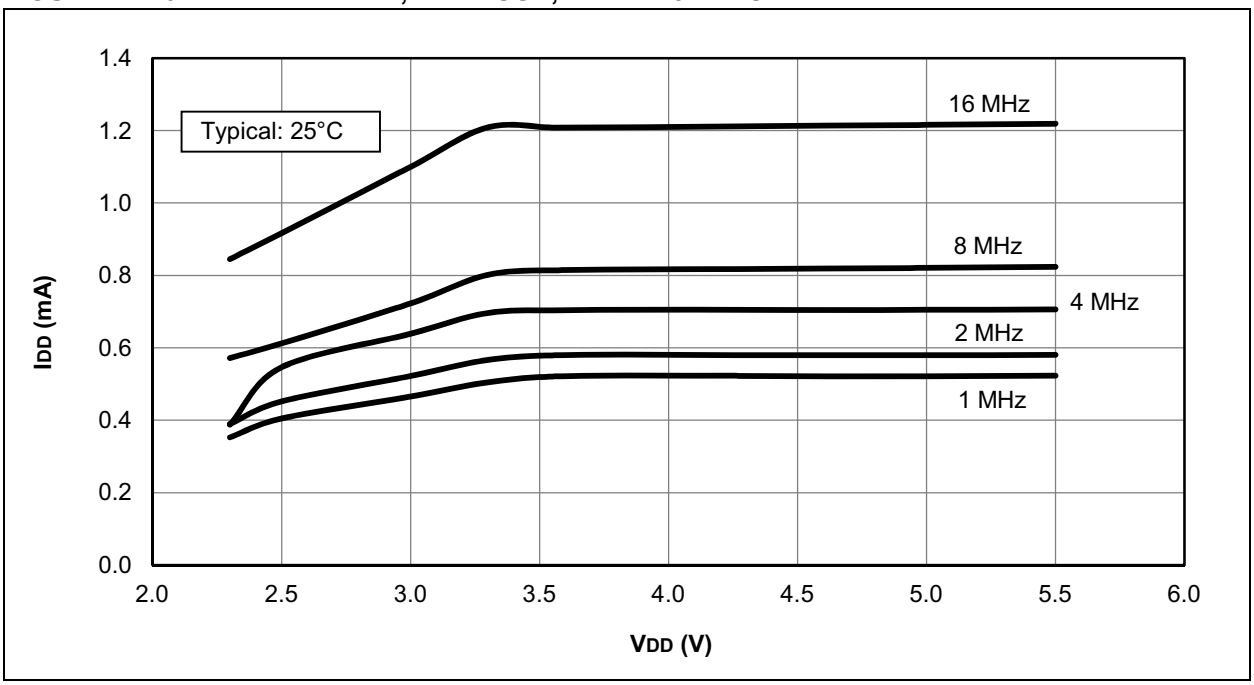


FIGURE 27-20: IDD MAXIMUM, HFINTOSC, PIC12F1571/2 ONLY

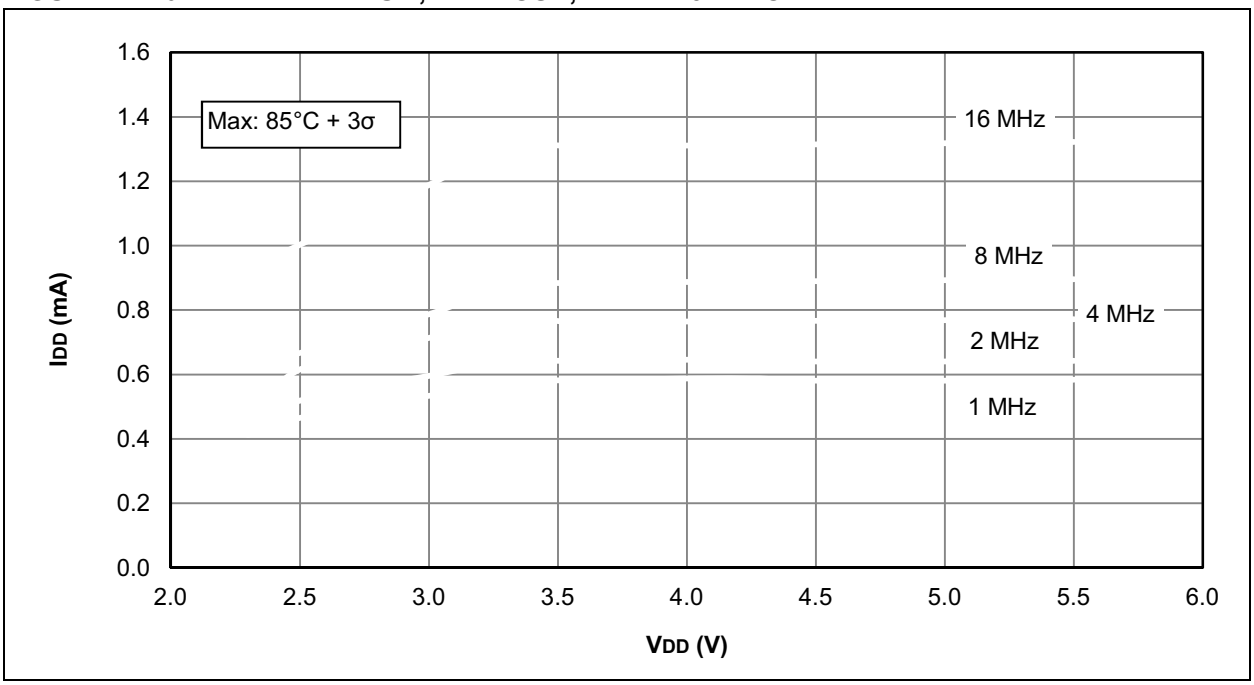


FIGURE 27-33: IPD, COMPARATOR, LOW-POWER MODE (CxSP = 0), PIC12F1571/2 ONLY

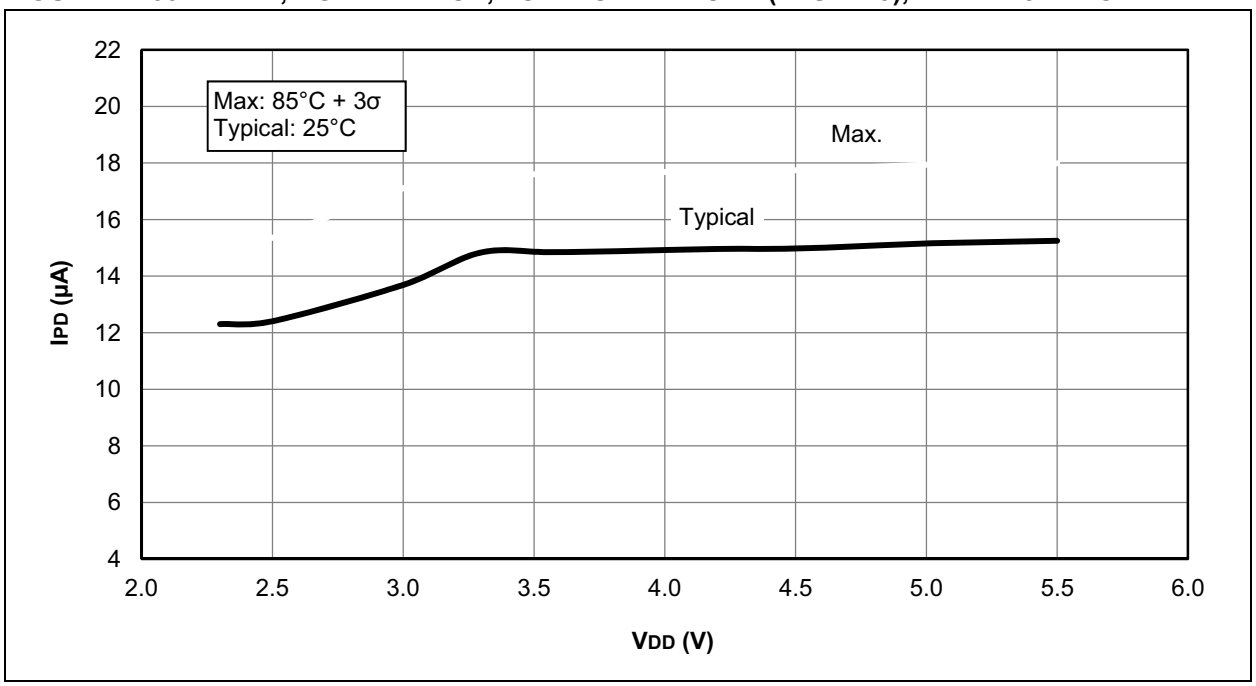
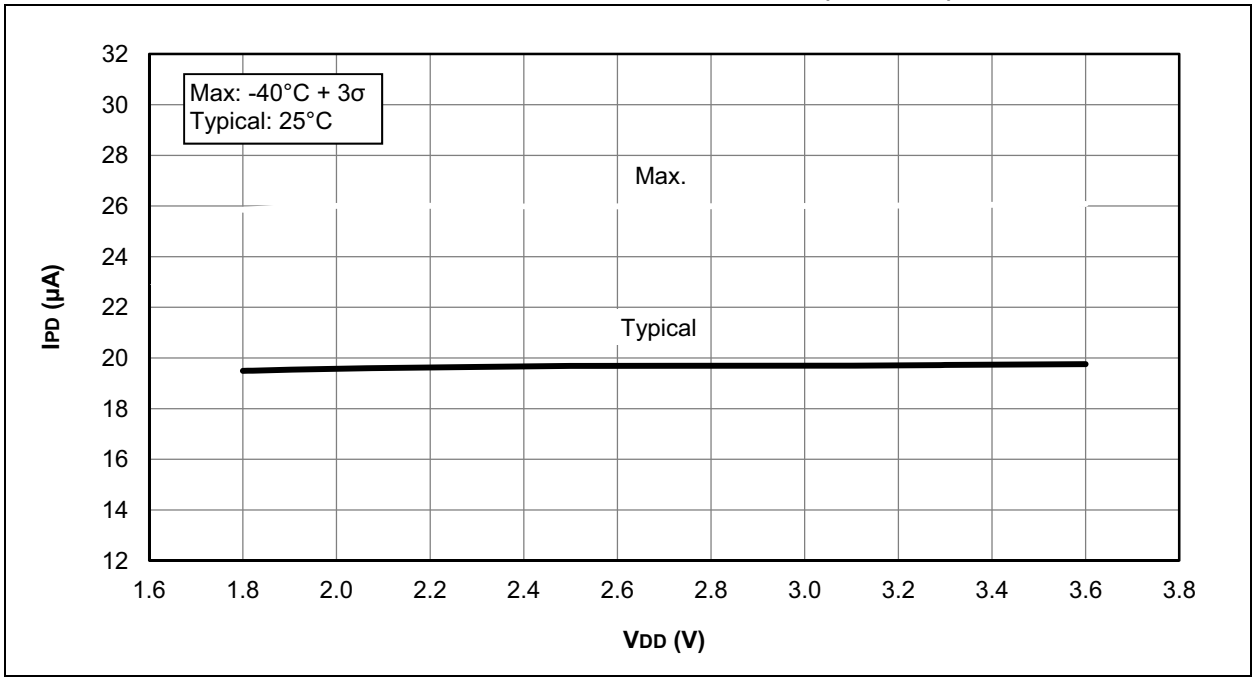


FIGURE 27-34: IPD, COMPARATOR, NORMAL POWER MODE (CxSP = 1), PIC12LF1571/2 ONLY



28.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

28.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

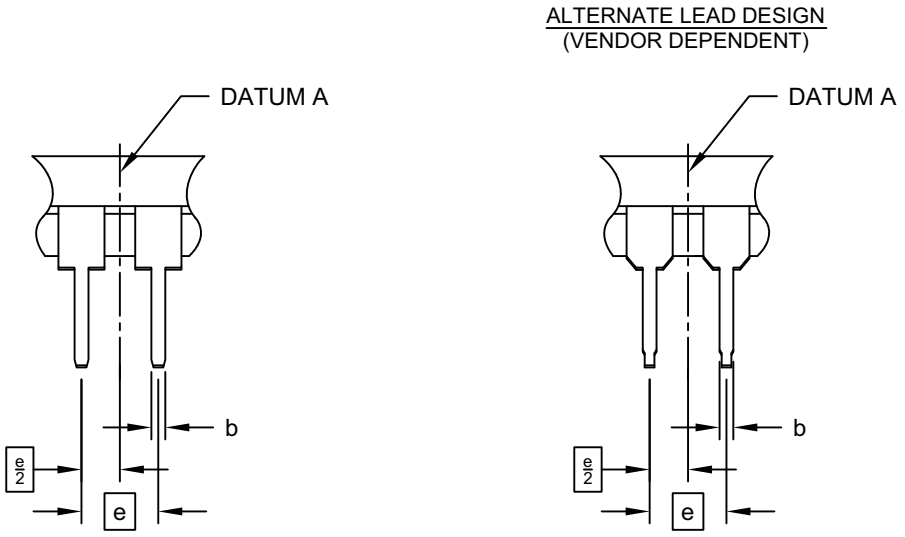
- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

8-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		8	
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	.290	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.348	.365	.400
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.040	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	-	-	.430

Notes:

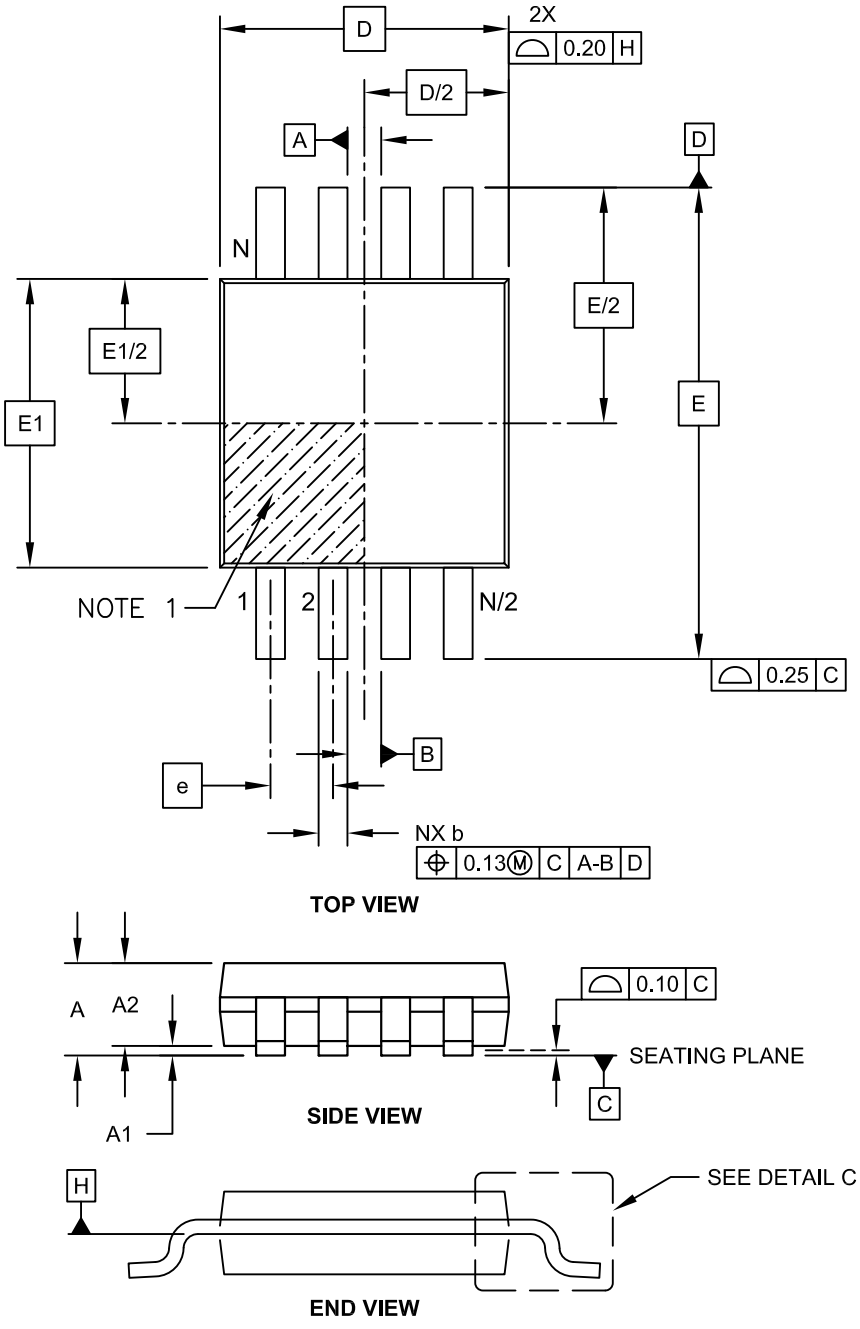
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-018D Sheet 2 of 2

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



Microchip Technology Drawing C04-111C Sheet 1 of 2