Microchip Technology - PIC12F1571T-I/SN Datasheet

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	-
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	6
Program Memory Size	1.75KB (1K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
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-SOIC (0.154", 3.90mm Width)
Supplier Device Package	8-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12f1571t-i-sn

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^{2}C = Schmitt Trigger input with I^{2}C 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.

TABLE 3-10: SPECIAL FUNCTION REGISTER SUMMARY

Addr	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 0											
00Ch	PORTA	—	—	RA<5:0>						`--xx xxxx`	`--xx xxxx`
00Dh	—	Unimplemented								—	—
00Eh	—	Unimplemented								—	—
00Fh	—	Unimplemented								—	—
010h	—	Unimplemented								—	—
011h	PIR1	TMR1GIF	ADIF	RCIF[2]	TXIF[2]	—	—	TMR2IF	TMR1IF	`0000 --00`	`0000 --00`
012h	PIR2	—	—	C1IF	—	—	—	—	—	`--0- ----`	`--0- ----`
013h	PIR3	—	PWM3IF	PWM2IF	PWM1IF	—	—	—	—	`-000 ----`	`-000 ----`
014h	—	Unimplemented								—	—
015h	TMR0	Holding Register for the 8-Bit Timer0 Count								`xxxx xxxx`	`uuuu uuuu`
016h	TMR1L	Holding Register for the Least Significant Byte of the 16-Bit TMR1 Count								`xxxx xxxx`	`uuuu uuuu`
017h	TMR1H	Holding Register for the Most Significant Byte of the 16-Bit TMR1 Count								`xxxx xxxx`	`uuuu uuuu`
018h	T1CON	TMR1CS<1:0>		T1CKPS<1:0>		—	$\overline{\text{T1SYNC}}$	—	TMR1ON	`0000 -0-0`	`uuuu -u-u`
019h	T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/$\overline{\text{DONE}}$	T1GVAL	T1GSS<1:0>		`0000 0x00`	`uuuu uxuu`
01Ah	TMR2	Timer2 Module Register								`0000 0000`	`0000 0000`
01Bh	PR2	Timer2 Period Register								`1111 1111`	`1111 1111`
01Ch	T2CON	—	T2OUTPS<3:0>				TMR2ON	T2CKPS<1:0>		`-000 0000`	`-000 0000`
01Dh	—	Unimplemented								—	—
01Eh	—	Unimplemented								—	—
01Fh	—	Unimplemented								—	—
Bank 1											
08Ch	TRISA	—	—	TRISA<5:4>		—[2]	TRISA<2:0>			`--11 1111`	`--11 1111`
08Dh	—	Unimplemented								—	—
08Eh	—	Unimplemented								—	—
08Fh	—	Unimplemented								—	—
090h	—	Unimplemented								—	—
091h	PIE1	TMR1GIE	ADIE	RCIE[2]	TXIE[2]	—	—	TMR2IE	TMR1IE	`0000 --00`	`0000 --00`
092h	PIE2	—	—	C1IE	—	—	—	—	—	`--0- ----`	`--0- ----`
093h	PIE3	—	PWM3IE	PWM2IE	PWM1IE	—	—	—	—	`-000 ----`	`-000 ----`
094h	—	Unimplemented								—	—
095h	OPTION_REG	$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			`1111 1111`	`1111 1111`
096h	PCON	STKOVF	STKUNF	—	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	`00-1 11qq`	`qq-q qquu`
097h	WDTCON	—	—	WDTPS<4:0>					SWDTEN	`--01 0110`	`--01 0110`
098h	OSCTUNE	—	—	TUN<5:0>						`--00 0000`	`--00 0000`
099h	OSCCON	SPLLEN	IRCF<3:0>				—	SCS<1:0>		`0011 1-00`	`0011 1-00`
09Ah	OSCSTAT	—	PLLR	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS	`-0q0 0q00`	`-qqq qqqq`
09Bh	ADRESL	ADC Result Register Low								`xxxx xxxx`	`uuuu uuuu`
09Ch	ADRESH	ADC Result Register High								`xxxx xxxx`	`uuuu uuuu`
09Dh	ADCON0	—	CHS<4:0>					GO/$\overline{\text{DONE}}$	ADON	`-000 0000`	`-000 0000`
09Eh	ADCON1	ADFM	ADCS<2:0>			—	—	ADPREF<1:0>		`0000 --00`	`0000 --00`
09Fh	ADCON2	TRIGSEL<3:0>				—	—	—	—	`0000 ----`	`0000 ----`

Legend: x = unknown; u = unchanged; q = value depends on condition; — = unimplemented; `r` = reserved. Shaded locations are unimplemented, read as ‘`0`’.

Note 1: PIC12F1571/2 only.

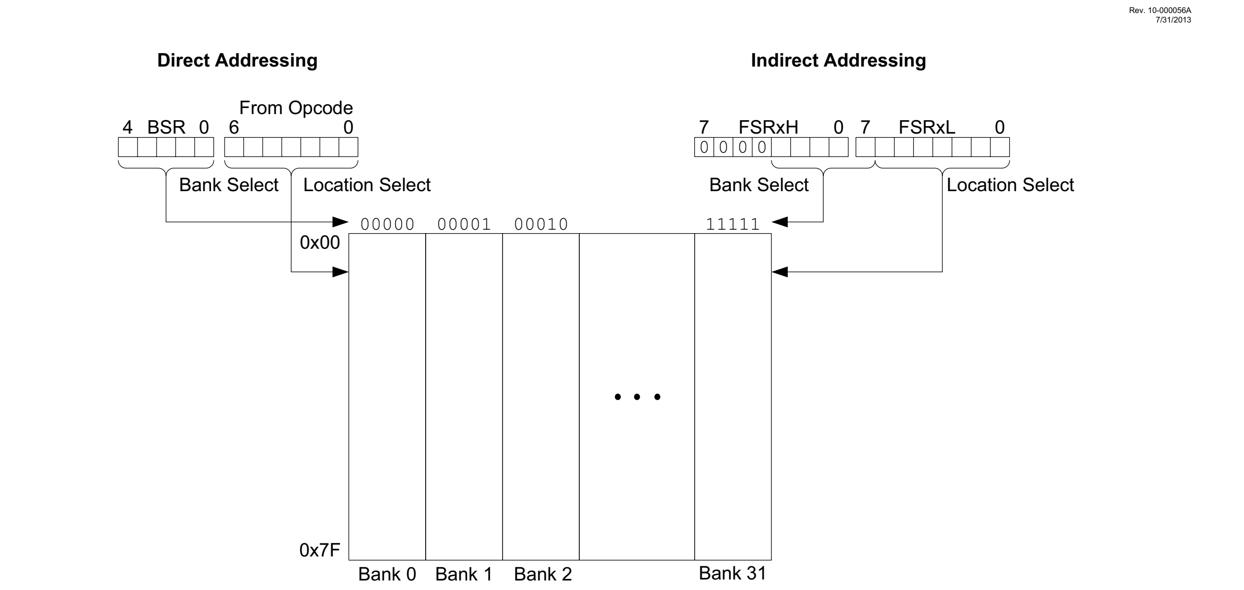
2: PIC12(L)F1572 only.

3: Unimplemented, read as ‘`1`’.

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



REGISTER 7-5: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

R/W-0/0	R/W-0/0	R-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
TMR1GIF	ADIF	RCIF[1]	TXIF[1]	—	—	TMR2IF	TMR1IF
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 **TMR1GIF:** Timer1 Gate Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 6 **ADIF:** ADC Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 5 **RCIF:** USART Receive Interrupt Flag bit[1]
1 = Interrupt is pending
0 = Interrupt is not pending

bit 4 **TXIF:** USART Transmit Interrupt Flag bit[1]
1 = Interrupt is pending
0 = Interrupt is not pending

bit 3-2 **Unimplemented:** Read as ‘0’

bit 1 **TMR2IF:** Timer2 to PR2 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 0 **TMR1IF:** Timer1 Overflow Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

Note 1: PIC12(L)F1572 only.

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Interrupt Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

8.0 POWER-DOWN MODE (SLEEP)

The Power-Down mode is entered by executing a SLEEP instruction.

Upon entering Sleep mode, the following conditions exist:

1. WDT will be cleared but keeps running if enabled for operation during Sleep.
2. $\overline{PD}$ bit of the STATUS register is cleared.
3. $\overline{TO}$ bit of the STATUS register is set.
4. CPU clock is disabled.
5. 31 kHz LFINTOSC is unaffected and peripherals that operate from it may continue operation in Sleep.
6. Timer1 and peripherals that operate from Timer1 continue operation in Sleep when the Timer1 clock source selected is:
   - LFINTOSC
   - T1CKI
   - Timer1 oscillator
7. ADC is unaffected if the dedicated FRC oscillator is selected.
8. I/O ports maintain the status they had before SLEEP was executed (driving high, low or high-impedance).
9. Resets other than WDT are not affected by Sleep mode.

Refer to individual chapters for more details on peripheral operation during Sleep.

To minimize current consumption, the following conditions should be considered:

- I/O pins should not be floating
- External circuitry sinking current from I/O pins
- Internal circuitry sourcing current from I/O pins
- Current draw from pins with internal weak pull-ups
- Modules using 31 kHz LFINTOSC
- CWG module using HFINTOSC

I/O pins that are high-impedance inputs should be pulled to V_{DD} or V_{SS} externally to avoid switching currents caused by floating inputs.

Examples of internal circuitry that might be sourcing current include the FVR module. See **Section 13.0 "Fixed Voltage Reference (FVR)"** for more information on this module.

8.1 Wake-up from Sleep

The device can wake-up from Sleep through one of the following events:

1. External Reset input on $\overline{MCLR}$ pin if enabled.
2. BOR Reset if enabled.
3. POR Reset.
4. Watchdog Timer if enabled.
5. Any external interrupt.
6. Interrupts by peripherals capable of running during Sleep (see individual peripheral for more information)

The first three events will cause a device Reset. The last three events are considered a continuation of program execution. To determine whether a device Reset or wake-up event occurred, refer to **Section 6.12 "Determining the Cause of a Reset"**.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is prefetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be enabled. Wake-up will occur regardless of the state of the GIE bit. If the GIE bit is disabled, the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is enabled, the device executes the instruction after the SLEEP instruction, the device will then call the Interrupt Service Routine. In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

The WDT is cleared when the device wakes up from Sleep, regardless of the source of wake-up.

8.1.1 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a SLEEP instruction:
  - SLEEP instruction will execute as a NOP.
  - WDT and WDT prescaler will not be cleared
  - $\overline{TO}$ bit of the STATUS register will not be set
  - $\overline{PD}$ bit of the STATUS register will not be cleared.
- If the interrupt occurs **during or after** the execution of a SLEEP instruction:
  - SLEEP instruction will be completely executed
  - Device will immediately wake-up from Sleep
  - WDT and WDT prescaler will be cleared
  - $\overline{TO}$ bit of the STATUS register will be set
  - $\overline{PD}$ bit of the STATUS register will be cleared

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

10.2.4 WRITING TO FLASH PROGRAM MEMORY

Program memory is programmed using the following steps:

1. Load the address in PMADRH:PMADRL of the row to be programmed.
2. Load each write latch with data.
3. Initiate a programming operation.
4. Repeat Steps 1 through 3 until all data is written.

Before writing to program memory, the word(s) to be written must be erased or previously unwritten. Program memory can only be erased one row at a time. No automatic erase occurs upon the initiation of the write.

Program memory can be written one or more words at a time. The maximum number of words written at one time is equal to the number of write latches. See Figure 10-5 (row writes to program memory with 16 write latches) for more details.

The write latches are aligned to the Flash row address boundary defined by the upper 11 bits of PMADRH:PMADRL (PMADRH<6:0>:PMADRL<7:4>), with the lower 4 bits of PMADRL (PMADRL<3:0>) determining the write latch being loaded. Write operations do not cross these boundaries. At the completion of a program memory write operation, the data in the write latches is reset to contain 0x3FFF.

The following steps should be completed to load the write latches and program a row of program memory. These steps are divided into two parts. First, each write latch is loaded with data from the PMDATH:PMDATL using the unlock sequence with LWLO = 1. When the last word to be loaded into the write latch is ready, the LWLO bit is cleared and the unlock sequence executed. This initiates the programming operation, writing all the latches into Flash program memory.

Note:	The special unlock sequence is required to load a write latch with data or initiate a Flash programming operation. If the unlock sequence is interrupted, writing to the latches or program memory will not be initiated.

1. Set the WREN bit of the PMCON1 register.
2. Clear the CFGS bit of the PMCON1 register.
3. Set the LWLO bit of the PMCON1 register. When the LWLO bit of the PMCON1 register is '1', the write sequence will only load the write latches and will not initiate the write to Flash program memory.
4. Load the PMADRH:PMADRL register pair with the address of the location to be written.
5. Load the PMDATH:PMDATL register pair with the program memory data to be written.
6. Execute the unlock sequence (**Section 10.2.2 "Flash Memory Unlock Sequence"**). The write latch is now loaded.
7. Increment the PMADRH:PMADRL register pair to point to the next location.
8. Repeat Steps 5 through 7 until all but the last write latch has been loaded.
9. Clear the LWLO bit of the PMCON1 register. When the LWLO bit of the PMCON1 register is '0', the write sequence will initiate the write to Flash program memory.
10. Load the PMDATH:PMDATL register pair with the program memory data to be written.
11. Execute the unlock sequence (**Section 10.2.2 "Flash Memory Unlock Sequence"**). The entire program memory latch content is now written to Flash program memory.

Note:	The program memory write latches are reset to the blank state (0x3FFF) at the completion of every write or erase operation. As a result, it is not necessary to load all the program memory write latches. Unloaded latches will remain in the blank state.

An example of the complete write sequence is shown in Example 10-3. The initial address is loaded into the PMADRH:PMADRL register pair; the data is loaded using Indirect Addressing.

15.2 ADC Operation

15.2.1 STARTING A CONVERSION

To enable the ADC module, the ADON bit of the ADCON0 register must be set to a ‘1’. Setting the GO/$\overline{\text{DONE}}$ bit of the ADCON0 register to a ‘1’ will start the Analog-to-Digital conversion.

> **Note:** The GO/$\overline{\text{DONE}}$ bit should not be set in the same instruction that turns on the ADC. Refer to **Section 15.2.6 “ADC Conversion Procedure”**.

15.2.2 COMPLETION OF A CONVERSION

When the conversion is complete, the ADC module will:

- Clear the GO/$\overline{\text{DONE}}$ bit
- Set the ADIF Interrupt Flag bit
- Update the ADRESH and ADRESL registers with new conversion result

15.2.3 TERMINATING A CONVERSION

If a conversion must be terminated before completion, the GO/$\overline{\text{DONE}}$ bit can be cleared in software. The ADRESH and ADRESL registers will be updated with the partially complete Analog-to-Digital conversion sample. Incomplete bits will match the last bit converted.

> **Note:** A device Reset forces all registers to their Reset state. Thus, the ADC module is turned off and any pending conversion is terminated.

15.2.4 ADC OPERATION DURING SLEEP

The ADC module can operate during Sleep. This requires the ADC clock source to be set to the FRC option. Performing the ADC conversion during Sleep can reduce system noise. If the ADC interrupt is enabled, the device will wake-up from Sleep when the conversion completes. If the ADC interrupt is disabled, the ADC module is turned off after the conversion completes, although the ADON bit remains set.

When the ADC clock source is something other than FRC, a SLEEP instruction causes the present conversion to be aborted and the ADC module is turned off, although the ADON bit remains set.

15.2.5 AUTO-CONVERSION TRIGGER

The auto-conversion trigger allows periodic ADC measurements without software intervention. When a rising edge of the selected source occurs, the GO/$\overline{\text{DONE}}$ bit is set by hardware.

The auto-conversion trigger source is selected with the TRIGSEL<3:0> bits of the ADCON2 register.

Using the auto-conversion trigger does not assure proper ADC timing. It is the user’s responsibility to ensure that the ADC timing requirements are met.

The PWM module can trigger the ADC in two ways, directly through the PWMx_OFx_match or through the interrupts generated by all four match signals. See **Section 22.0 “16-Bit Pulse-Width Modulation (PWM) Module”**. If the interrupts are chosen, each enabled interrupt in PWMxINTE will trigger a conversion. Refer to Figure 15-4 for more information.

See Table 15-2 for auto-conversion sources.

FIGURE 15-4: 16-BIT PWM INTERRUPT BLOCK DIAGRAM

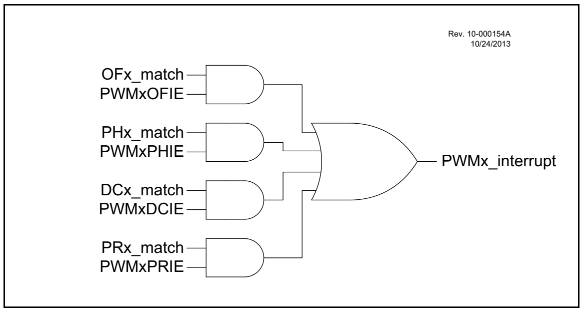


TABLE 15-2: AUTO-CONVERSION SOURCES

Source Peripheral	Signal Name
Timer0	T0_overflow
Timer1	T1_overflow
Timer2	T2_match
Comparator C1	C1OUT_sync
PWM1	PWM1_OF_match
PWM1	PWM1_interrupt
PWM2	PWM2_OF_match
PWM2	PWM2_interrupt
PWM3	PWM3_OF_match
PWM3	PWM3_interrupt

16.0 5-BIT DIGITAL-TO-ANALOG CONVERTER (DAC) MODULE

The Digital-to-Analog Converter supplies a variable voltage reference, ratiometric with the input source, with 32 selectable output levels.

The positive input source (VSOURCE+) of the DAC can be connected to the:

- External VREF+ pin
- VDD supply voltage
- FVR buffered output

The negative input source (VSOURCE-) of the DAC can be connected to the:

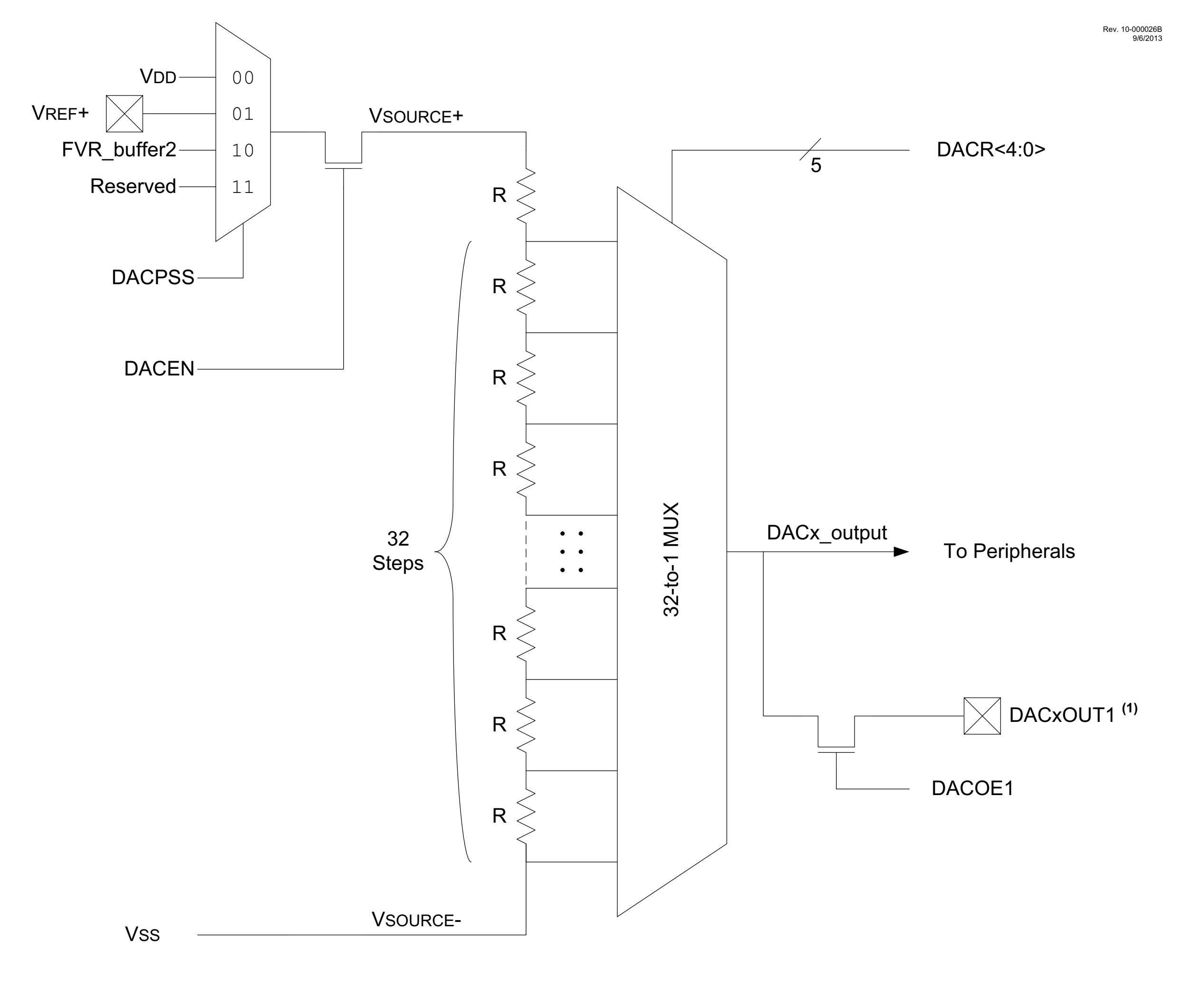
- Vss

The output of the DAC (DACx_output) can be selected as a reference voltage to the following:

- Comparator positive input
- ADC input channel
- DACxOUT1 pin

The Digital-to-Analog Converter (DAC) can be enabled by setting the DACEN bit of the DACxCON0 register.

FIGURE 16-1: DIGITAL-TO-ANALOG CONVERTER BLOCK DIAGRAM



Note 1: The unbuffered DACx_output is provided on the DACxOUT pin(s).

17.8 Register Definitions: Comparator Control

REGISTER 17-1: CMxCON0: COMPARATOR Cx CONTROL REGISTER 0

R/W-0/0	R-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
CxON	CxOUT	CxOE	CxPOL	—	CxSP	CxHYS	CxSYNC
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 **CxON:** Comparator Enable bit
1 = Comparator is enabled
0 = Comparator is disabled and consumes no active power

bit 6 **CxOUT:** Comparator Output bit
If CxPOL = 1 (inverted polarity):
1 = CxVP < CxVN
0 = CxVP > CxVN
If CxPOL = 0 (non-inverted polarity):
1 = CxVP > CxVN
0 = CxVP < CxVN

bit 5 **CxOE:** Comparator Output Enable bit
1 = CxOUT is present on the CxOUT pin; requires that the associated TRISx bit be cleared to actually drive the pin, not affected by CxON
0 = CxOUT is internal only

bit 4 **CxPOL:** Comparator Output Polarity Select bit
1 = Comparator output is inverted
0 = Comparator output is not inverted

bit 3 **Unimplemented:** Read as ‘0’

bit 2 **CxSP:** Comparator Speed/Power Select bit
1 = Comparator mode is in Normal Power, Higher Speed mode
0 = Comparator mode is in Low-Power, Low-Speed mode

bit 1 **CxHYS:** Comparator Hysteresis Enable bit
1 = Comparator hysteresis is enabled
0 = Comparator hysteresis is disabled

bit 0 **CxSYNC:** Comparator Output Synchronous Mode bit
1 = Comparator output to Timer1 and I/O pin is synchronous to changes on Timer1 clock source; output updated on the falling edge of Timer1 clock source
0 = Comparator output to Timer1 and I/O pin is asynchronous

18.2 Register Definitions: Option Register

REGISTER 18-1: OPTION_REG: OPTION REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>		
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 **$\overline{\text{WPUEN}}$:** Weak Pull-Up Enable bit
1 = All weak pull-ups are disabled (except $\overline{\text{MCLR}}$ if it is enabled)
0 = Weak pull-ups are enabled by individual WPUx latch values

bit 6 **INTEDG:** Interrupt Edge Select bit
1 = Interrupt on rising edge of INT pin
0 = Interrupt on falling edge of INT pin

bit 5 **TMR0CS:** Timer0 Clock Source Select bit
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock (FOSC/4)

bit 4 **TMR0SE:** Timer0 Source Edge Select bit
1 = Increment on high-to-low transition on T0CKI pin
0 = Increment on low-to-high transition on T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
1 = Prescaler is not assigned to the Timer0 module
0 = Prescaler is assigned to the Timer0 module

bit 2-0 **PS<2:0>:** Prescaler Rate Select bits

Bit Value	Timer0 Rate
000	1 : 2
001	1 : 4
010	1 : 8
011	1 : 16
100	1 : 32
101	1 : 64
110	1 : 128
111	1 : 256

TABLE 18-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER0

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ADCON2	TRIGSEL<3:0>				—	—	—	—	137
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	74
OPTION_REG	$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			157
TMR0	Holding Register for the 8-bit Timer0 Count								155*
TRISA	—	—	TRISA5	TRISA4	—(1)	TRISA2	TRISA1	TRISA0	113

Legend: — = unimplemented location, read as ‘0’. Shaded cells are not used by the Timer0 module.
* Page provides register information.

Note 1: Unimplemented, read as ‘1’.

21.1.1.5 TSR Status

The TRMT bit of the TXSTA register indicates the status of the TSR register. This is a read-only bit. The TRMT bit is set when the TSR register is empty and is cleared when a character is transferred to the TSR register from the TXREG. The TRMT bit remains clear until all bits have been shifted out of the TSR register. No interrupt logic is tied to this bit, so the user has to poll this bit to determine the TSR status.

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

21.1.1.6 Transmitting 9-Bit Characters

The EUSART supports 9-bit character transmissions. When the TX9 bit of the TXSTA register is set, the EUSART will shift nine bits out for each character transmitted. The TX9D bit of the TXSTA register is the ninth and Most Significant data bit. When transmitting 9-bit data, the TX9D data bit must be written before writing the eight Least Significant bits into the TXREG. All nine bits of data will be transferred to the TSR register immediately after the TXREG is written.

A special 9-Bit Address mode is available for use with multiple receivers. See **Section 21.1.2.7 "Address Detection"** for more information on this mode.

21.1.1.7 Asynchronous 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 asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
3. If 9-bit transmission is desired, set the TX9 control bit. A set ninth data bit will indicate that the eight Least Significant data bits are an address when the receiver is set for address detection.
4. Set the SCKP bit if inverted transmit is desired.
5. Enable the transmission by setting the TXEN control bit. This will cause the TXIF interrupt bit to be set.
6. If interrupts are desired, set the TXIE interrupt enable bit of the PIE1 register. An interrupt will occur immediately provided that the GIE and PEIE bits of the INTCON register are also set.
7. If 9-bit transmission is selected, the ninth bit should be loaded into the TX9D data bit.
8. Load 8-bit data into the TXREG register. This will start the transmission.

FIGURE 21-3: ASYNCHRONOUS TRANSMISSION

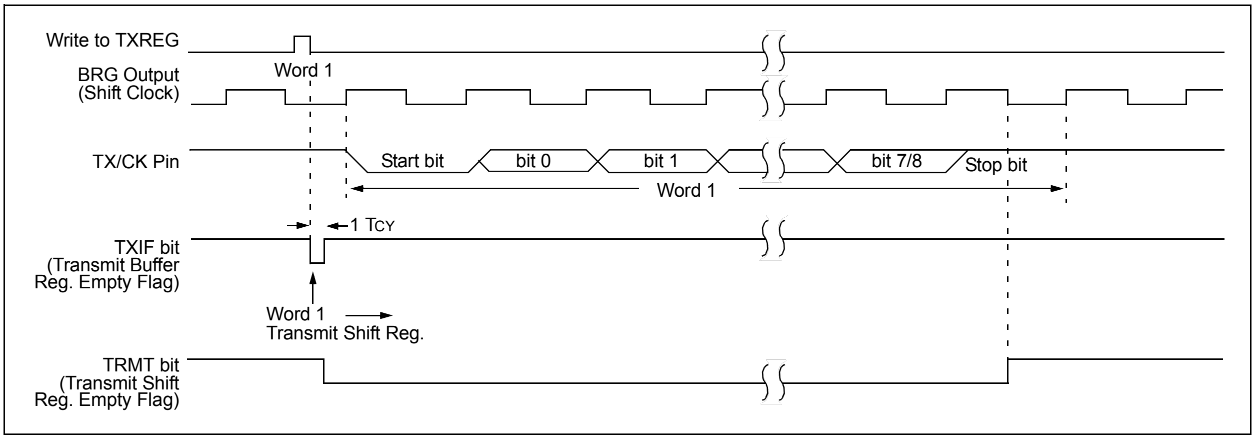
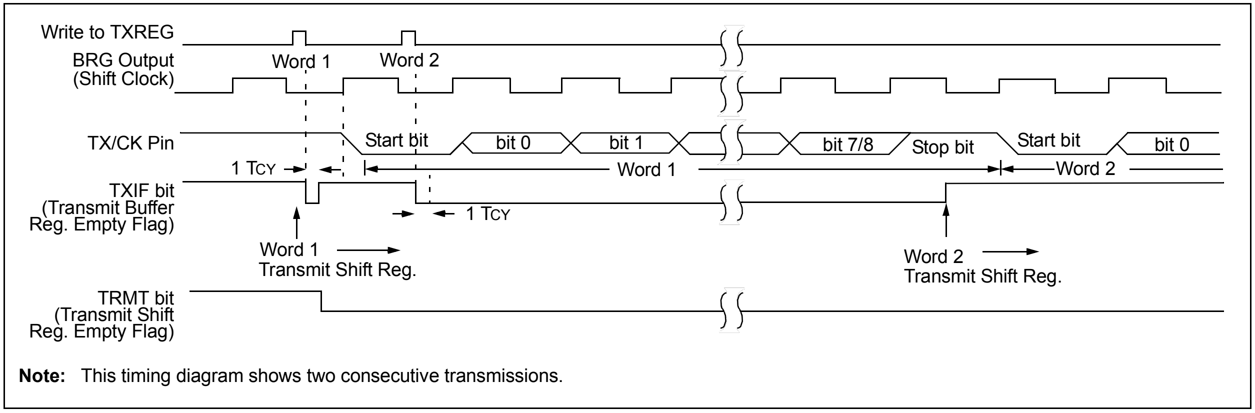
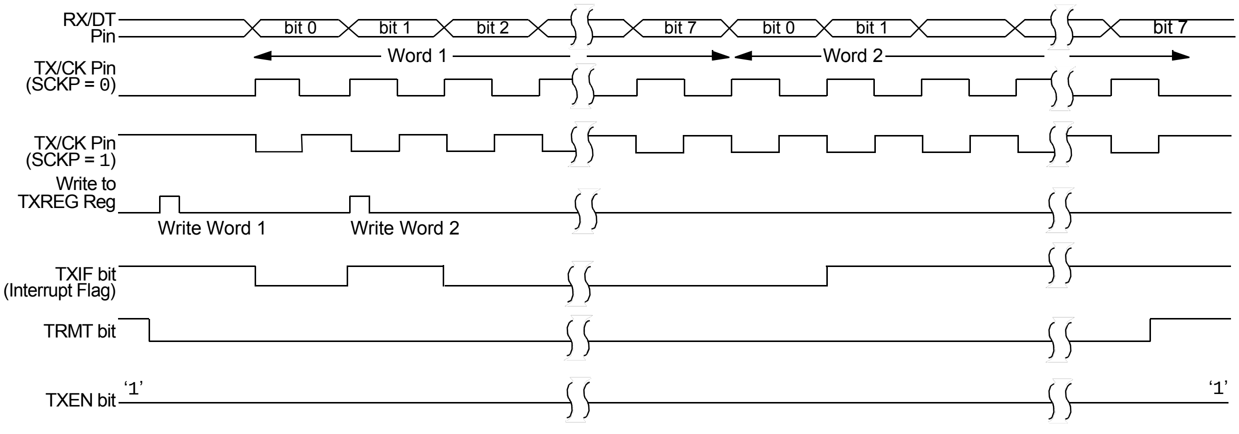

Write to TXREG
BRG Output (Shift Clock)
Word 1
TX/CK Pin
Start bit
bit 0
bit 1
bit 7/8
Stop bit
Word 1
1 TCY
TXIF bit (Transmit Buffer Reg. Empty Flag)
Word 1 → Transmit Shift Reg.
TRMT bit (Transmit Shift Reg. Empty Flag)

FIGURE 21-4: ASYNCHRONOUS TRANSMISSION (BACK-TO-BACK)


Write to TXREG
BRG Output (Shift Clock)
Word 1
Word 2
TX/CK Pin
Start bit
bit 0
bit 1
bit 7/8
Stop bit
Start bit
bit 0
1 TCY
Word 1
Word 2
TXIF bit (Transmit Buffer Reg. Empty Flag)
1 TCY
Word 1 → Transmit Shift Reg.
Word 2 → Transmit Shift Reg.
TRMT bit (Transmit Shift Reg. Empty Flag)

Note: This timing diagram shows two consecutive transmissions.

FIGURE 21-10: SYNCHRONOUS TRANSMISSION



Note: Sync Master mode, SPBRGL = 0, continuous transmission of two 8-bit words.

FIGURE 21-11: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)

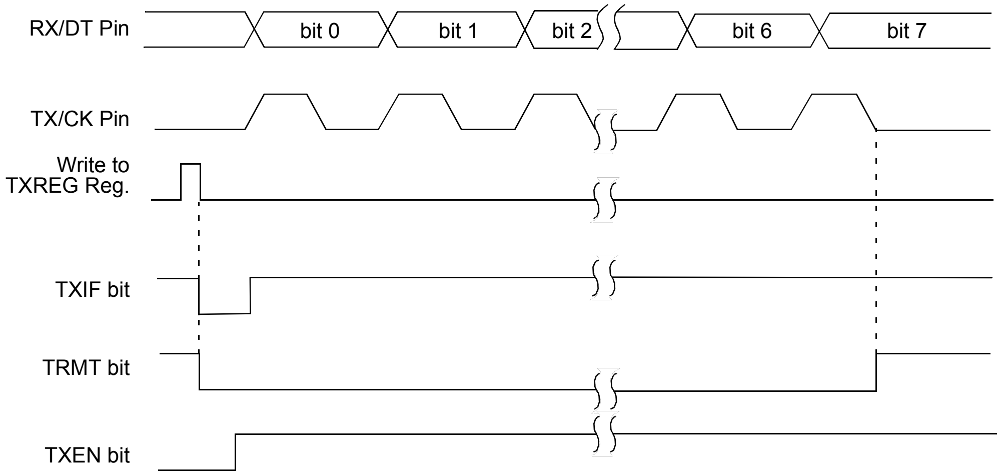


TABLE 21-7: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	186
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	74
PIE1	TMR1GIE	ADIE	RCIE(1)	TXIE(1)	—	—	TMR2IE	TMR1IE	75
PIR1	TMR1GIF	ADIF	RCIF(1)	TXIF(1)	—	—	TMR2IF	TMR1IF	78
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	185
SPBRGL	BRG<7:0>								187*
SPBRGH	BRG<15:8>								187*
TXREG	EUSART Transmit Data Register								177*
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	184

Legend: — = unimplemented location, read as '0'. Shaded cells are not used for synchronous master transmission.
* Page provides register information.

Note 1: PIC12(L)F1572 only.

21.5.1.5 Synchronous Master Reception

Data is received at the RX/DT pin. The RX/DT pin output driver is automatically disabled when the EUSART is configured for synchronous master receive operation.

In Synchronous mode, reception is enabled by setting either the Single Receive Enable bit (SREN of the RCSTA register) or the Continuous Receive Enable bit (CREN of the RCSTA register).

When SREN is set and CREN is clear, only as many clock cycles are generated as there are data bits in a single character. The SREN bit is automatically cleared at the completion of one character. When CREN is set, clocks are continuously generated until CREN is cleared. If CREN is cleared in the middle of a character, the CK clock stops immediately and the partial character is discarded. If SREN and CREN are both set, then SREN is cleared at the completion of the first character and CREN takes precedence.

To initiate reception, set either SREN or CREN. Data is sampled at the RX/DT pin on the trailing edge of the TX/CK clock pin and is shifted into the Receive Shift Register (RSR). When a complete character is received into the RSR, the RCIF bit is set and the character is automatically transferred to the two-character receive FIFO. The Least Significant eight bits of the top character in the receive FIFO are available in RCREG. The RCIF bit remains set as long as there are unread characters in the receive FIFO.

Note:	If the RX/DT function is on an analog pin, the corresponding ANSELx bit must be cleared for the receiver to function.

21.5.1.6 Slave Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a slave receives the clock on the TX/CK line. The TX/CK pin output driver is automatically disabled when the device is configured for synchronous slave 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 data bit is transferred for each clock cycle. Only as many clock cycles should be received as there are data bits.

Note:	If the device is configured as a slave and the TX/CK function is on an analog pin, the corresponding ANSELx bit must be cleared.

21.5.1.7 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before RCREG is read to access the FIFO. When this happens, the OERR bit of the RCSTA register is set. Previous data in the FIFO will not be overwritten. The two characters in the FIFO buffer can be read, however, no additional characters will be received until the error is cleared. The OERR bit can only be cleared by clearing the overrun condition. If the overrun error occurred when the SREN bit is set and CREN is clear, then the error is cleared by reading RCREG. If the overrun occurred when the CREN bit is set, then the error condition is cleared by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.

21.5.1.8 Receiving 9-Bit Characters

The EUSART supports 9-bit character reception. When the RX9 bit of the RCSTA register is set the EUSART will shift 9 bits into the RSR for each character received. The RX9D bit of the RCSTA register is the ninth, and Most Significant, data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the eight Least Significant bits from the RCREG.

21.5.1.9 Synchronous Master Reception Setup

1. Initialize the SPBRGH/SPBRGL register pair for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
2. Clear the ANSELx bit for the RX pin (if applicable).
3. Enable the synchronous master serial port by setting bits, SYNC, SPEN and CSRC.
4. Ensure bits, CREN and SREN, are clear.
5. If interrupts are desired, set the RCIE bit of the PIE1 register, and the GIE and PEIE bits of the INTCON register.
6. If 9-bit reception is desired, set bit, RX9.
7. Start reception by setting the SREN bit or for continuous reception, set the CREN bit.
8. Interrupt flag bit, RCIF, will be set when reception of a character is complete. An interrupt will be generated if the enable bit, RCIE, was set.
9. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
10. Read the 8-bit received data by reading the RCREG register.
11. If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.

FIGURE 23-5: SHUTDOWN FUNCTIONALITY, AUTO-RESTART DISABLED (GxARSEN = `0`, GxASDLA = `01`, GxASDLB = `01`)

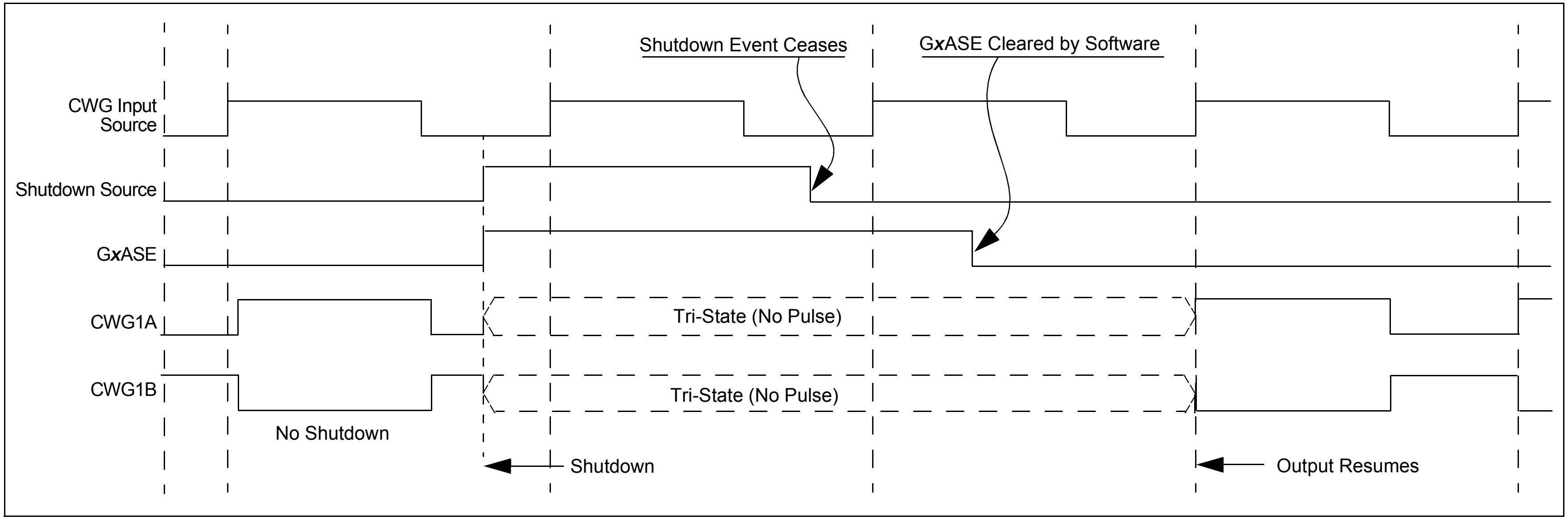


FIGURE 23-6: SHUTDOWN FUNCTIONALITY, AUTO-RESTART ENABLED (GxARSEN = `1`, GxASDLA = `01`, GxASDLB = `01`)

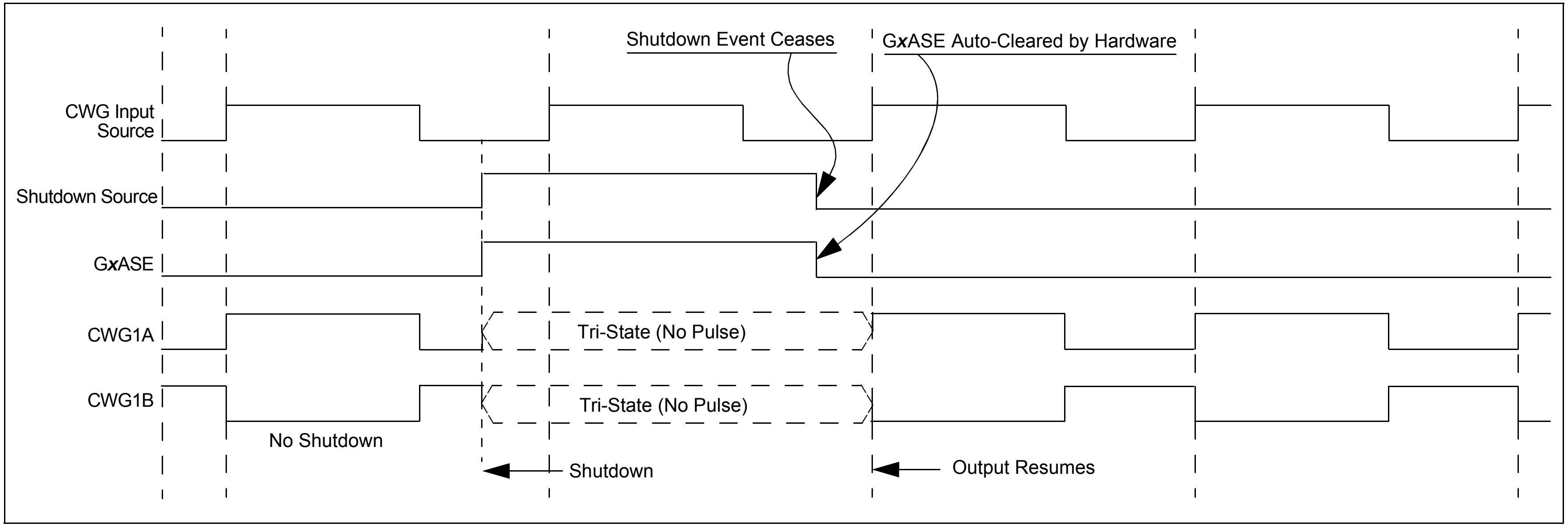
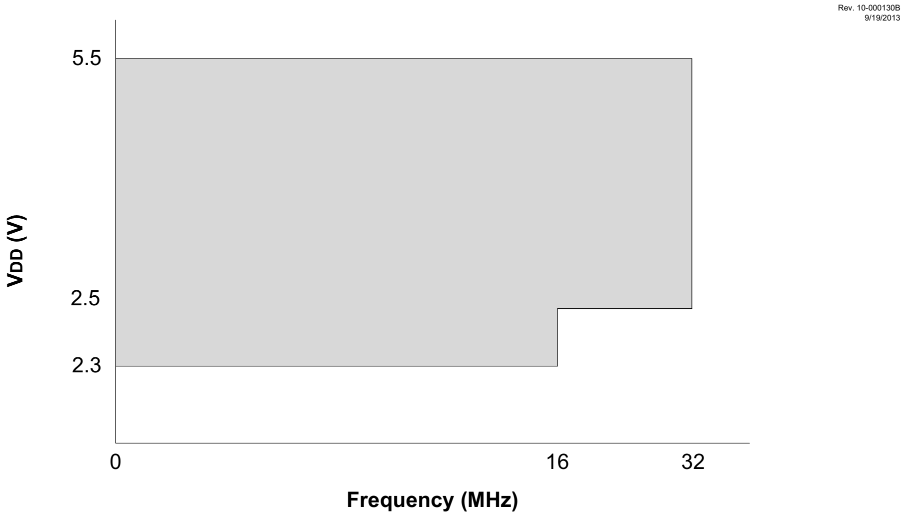


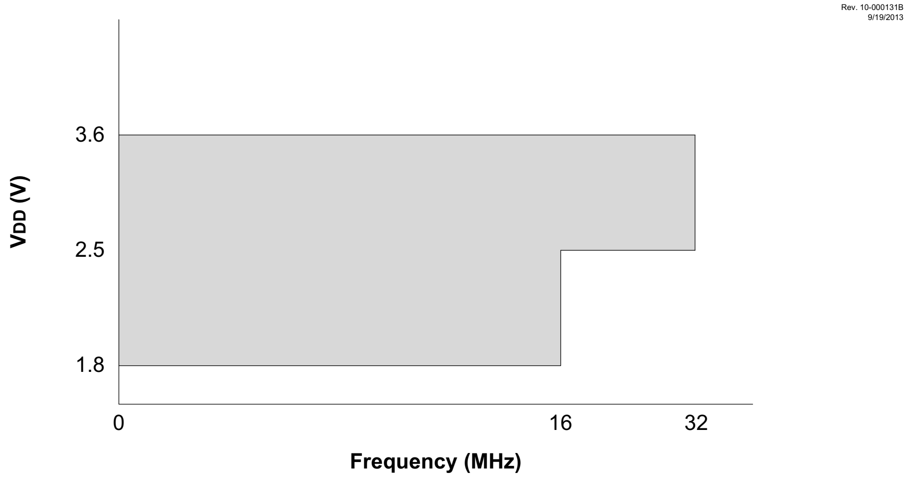
FIGURE 26-1: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC12F1571/2 ONLY



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

2: Refer to Table 26-7 for each Oscillator mode’s supported frequencies.

FIGURE 26-2: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC12LF1571/2 ONLY



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

2: Refer to Table 26-7 for each Oscillator mode’s supported frequencies.

FIGURE 26-7: CLKOUT AND I/O TIMING

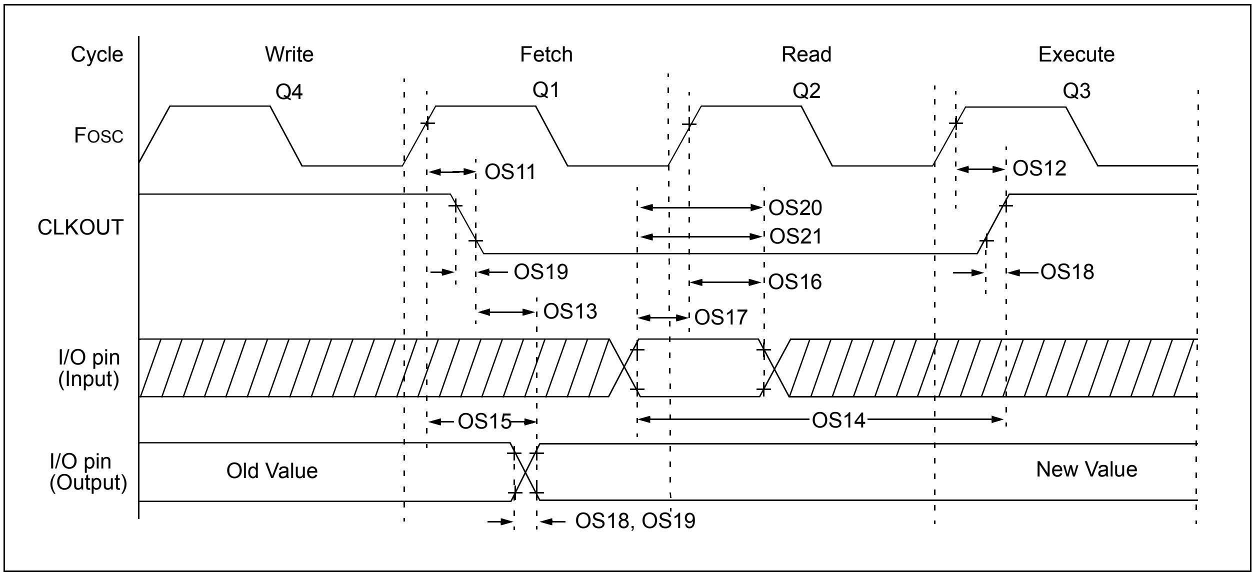


TABLE 26-10: CLKOUT AND I/O TIMING PARAMETERS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
OS11	TosH2ckL	FOSC↑ to CLKOUT↓(1)	—	—	70	ns	3.3V ≤ VDD ≤ 5.0V
OS12	TosH2ckH	FOSC↑ to CLKOUT↑(1)	—	—	72	ns	3.3V ≤ VDD ≤ 5.0V
OS13	TckL2ioV	CLKOUT↓ to Port Out Valid(1)	—	—	20	ns	
OS14	TioV2ckH	Port Input Valid Before CLKOUT↑(1)	TOSC + 200 ns	—	—	ns	
OS15	TosH2ioV	Fosc↑ (Q1 cycle) to Port Out Valid	—	50	70*	ns	3.3V ≤ VDD ≤ 5.0V
OS16	TosH2ioI	Fosc↑ (Q2 cycle) to Port Input Invalid (I/O in setup time)	50	—	—	ns	3.3V ≤ VDD ≤ 5.0V
OS17	TioV2osH	Port Input Valid to Fosc↑ (Q2 cycle) (I/O in setup time)	20	—	—	ns	
OS18*	TioR	Port Output Rise Time	— —	40 15	72 32	ns	VDD = 1.8V, 3.3V ≤ VDD ≤ 5.0V
OS19*	TioF	Port Output Fall Time	— —	28 15	55 30	ns	VDD = 1.8V, 3.3V ≤ VDD ≤ 5.0V
OS20*	Tinp	INT Pin Input High or Low Time	25	—	—	ns	
OS21*	Tioc	Interrupt-On-Change New Input Level Time	25	—	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, +25°C unless otherwise stated.

Note 1: Measurements are taken in EXTRC mode where CLKOUT output is 4 x TOSC.

FIGURE 27-9: IDD TYPICAL, EC OSCILLATOR, HIGH-POWER MODE, PIC12LF1571/2 ONLY

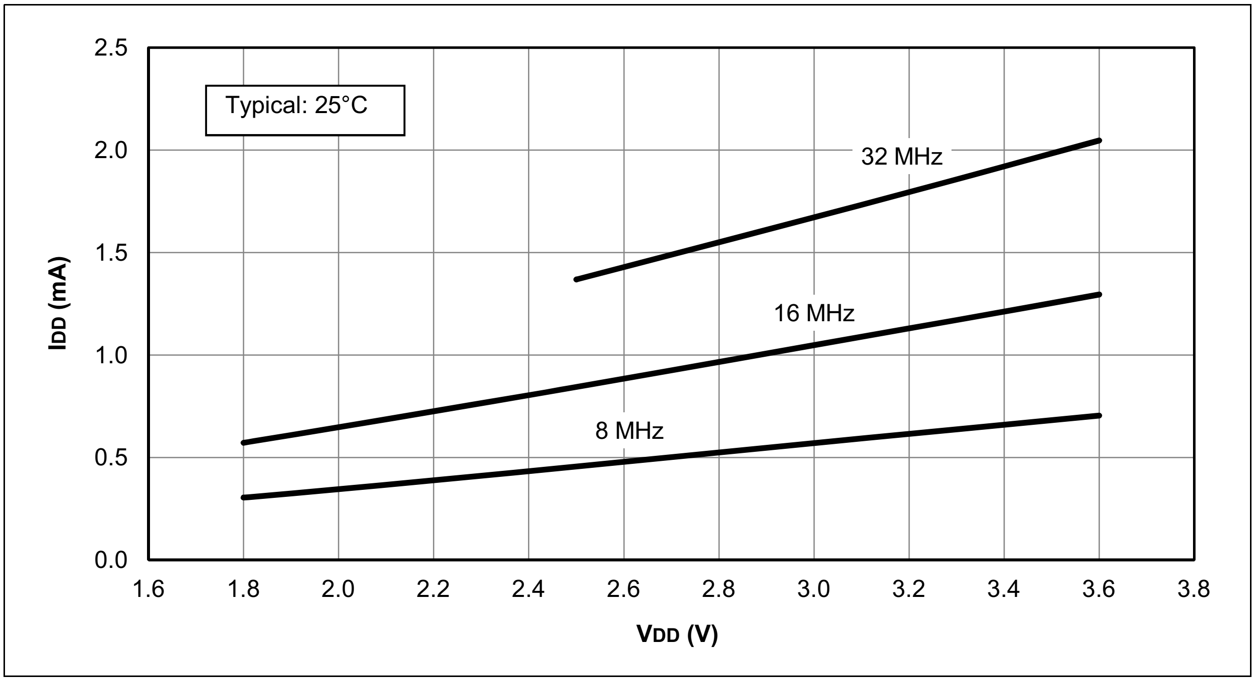


FIGURE 27-10: IDD MAXIMUM, EC OSCILLATOR, HIGH-POWER MODE, PIC12LF1571/2 ONLY

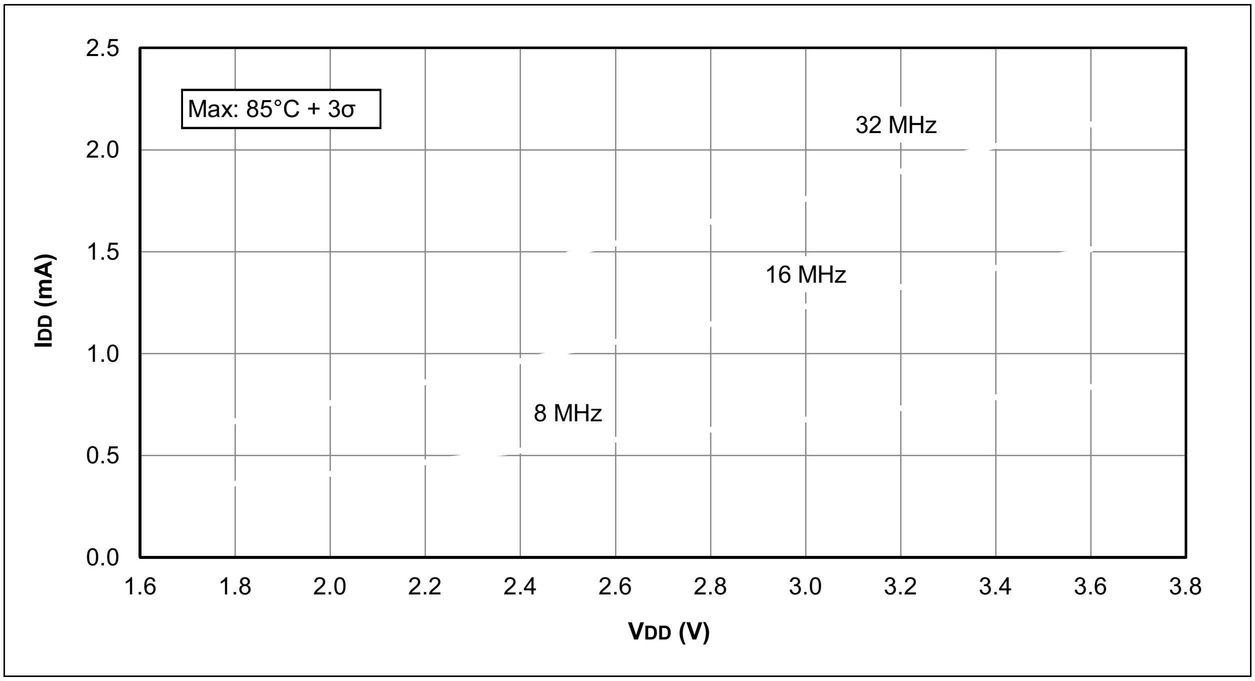


FIGURE 27-31: IPD, ADC NON-CONVERTING, PIC12LF1571/2 ONLY

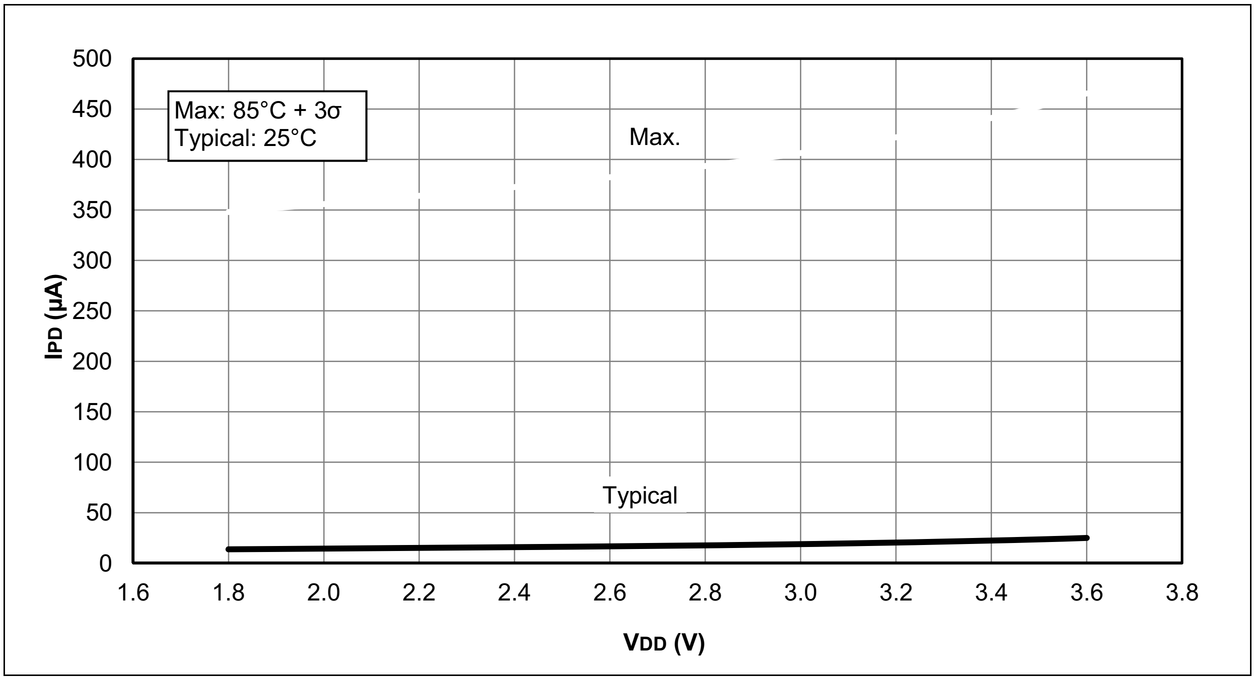
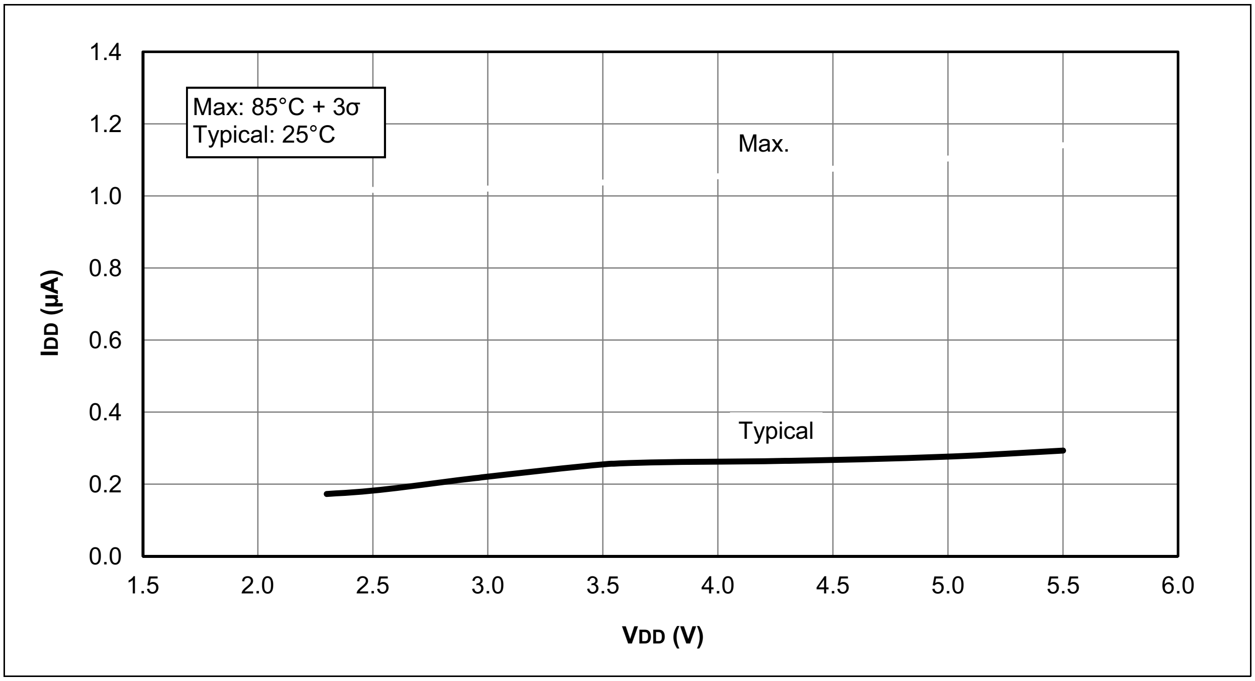
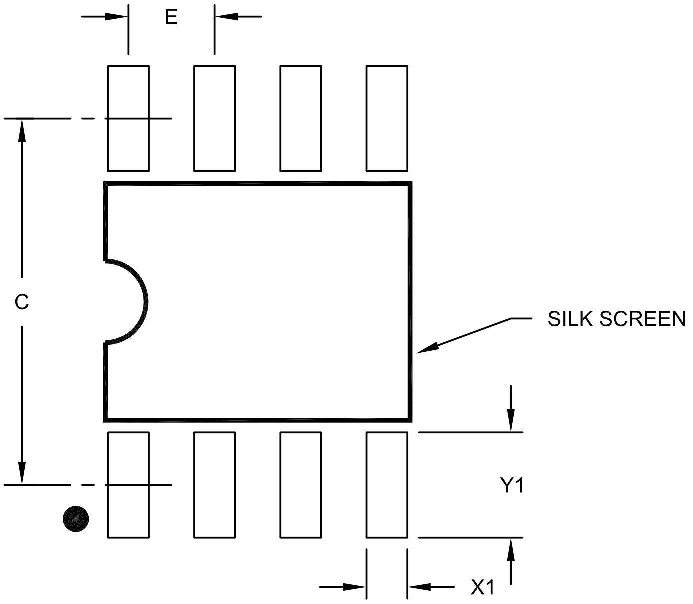


FIGURE 27-32: IPD, ADC NON-CONVERTING, PIC12F1571/2 ONLY



8-Lead Plastic Small Outline (SN) – Narrow, 3.90 mm Body [SOIC]

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	1.27 BSC		
Contact Pad Spacing	C		5.40	
Contact Pad Width (X8)	X1			0.60
Contact Pad Length (X8)	Y1			1.55

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2057A

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (10/2013)

Original release of this document.

Revision B (2/2014)

Updated PIC12(L)F1571/2 Family Types table Program Memory Flash heading (*words* to *K words*).

Revision C (8/2014)

Updated PWM chapter. Changed to Final data sheet. Updated IDD and IPD parameters in the Electrical Specification chapter. Added Characterization Graphs.

Added Section 1.1: Register and Bit Naming Conventions.

Updated Figures 5-3 and 15-5. Updated Tables 3-1, 3-7, and 3-10. Updated Section 15.2.5. Updated Equation 15-1.

Revision D (8/2015)

Updated Clocking Structure, Memory, Low-Power Features, Family Types table and Pin Diagram Table on cover pages.

Added Sections 3.2: High-Endurance Flash and 5.4: Clock Switching Before Sleep. Added Table 29-2 and 8-pin UDFN packaging.

Updated Examples 3-2 and 15-1.

Updated Figures 8-1, 21-1, 22-8 through 22-13 and 23-1.

Updated Registers 7-5, 8-1, 22-6 and 23-3.

Updated Sections 8.2.2, 15.2.6, 16.0, 21.0, 21.4.2, 22.3.3, 23.9.1.2, 23.11.1, 26.1 and 29.1.

Updated Tables 1, 3-3, 3-4, 3-10, 5-1, 16-1, 17-3, 22-2, 23-2, 26-6, 26-8 and 29-1.