Microchip Technology - PIC16LC782-I/SS 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	20MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	13
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x8b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	20-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lc782-i-ss

Peripheral Features (Continued):

- Dual Analog Comparator module with:
  - Individual enable and interrupt bits
  - Programmable speed and output polarity
  - Fully configurable inputs and outputs
  - Reference from DAC, or VREF1/VREF2
  - Low input offset voltage.
- VR voltage reference module:
  - 3.072V +/- 0.7% @25°C, AVDD = 5V
  - Configurable output to ADC reference, DAC reference, and VR pin
  - 5 mA sink/source
- Programmable Switch Mode Controller module:
  - PWM and PSM modes
  - Programmable switching frequency
  - Configurable for either single or dual feedback inputs
  - Configurable single or dual outputs
  - Slope compensation output available in single output mode

Key Features PIC® Mid-Range Reference Manual (DS33023)	PIC16C781	PIC16C782
Operating Frequency	DC - 20 MHz	DC - 20 MHZ
RESETS (and Delays)	POR, BOR, $\overline{\text{MCLR}}$, WDT (PWRT, OST)	POR, BOR, $\overline{\text{MCLR}}$, WDT (PWRT, OST)
Program Memory (14 bit words)	1K	2K
Data Memory (bytes)	128	128
Interrupts	8	8
I/O Ports	13 + 3 Input only	13 + 3 Input only
Timers	2	2
Programmable Switch Mode Controller	1	1
8-bit Analog-to-Digital Module	1	1
ADC channels	8 External, 2 Internal	8 External, 2 Internal
8-bit Digital-to-Analog Module	1	1
Comparators	2	2
Comparator Channels	4 (AN<7:4>)	4 (AN<7:4>)
Operational Amplifier	1	1
Voltage Reference	1	1
Brown-out Reset	Yes	Yes
Programmable Low Voltage Detect	Yes	Yes
Instruction Set	35 Instructions	35 Instructions

TABLE 1-1: PIC16C781/782 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0/OPA+	RA0	ST	N/A	Port Input
	AN0	AN	—	ADC Input
	OPA+	AN	—	OPAMP Non-inverting Input
RA1/AN1/OPA-	RA1	ST	N/A	Port Input
	AN1	AN	—	ADC Input
	OPA-	AN	—	OPAMP Inverting Input
RA2/AN2/VREF2	RA2	ST	CMOS	Bi-directional I/O
	AN2	AN	—	ADC Input
	VREF2	AN	—	Comparator 2 Voltage Reference Input
RA3/AN3/VREF1	RA3	ST	CMOS	Bi-directional I/O
	AN3	AN	—	ADC Input
	VREF1	AN	—	Comparator 1, ADC, DACREF Input
RA4/T0CKI	RA4	ST	OD	Bi-directional I/O
	TOCKI	ST	—	Timer0 Clock Input
RA5/$\overline{\text{MCLR}}$/VPP	RA5	ST	N/A	Port Input
	$\overline{\text{MCLR}}$	ST	—	Master Clear Input
	VPP	Power	—	Programming Voltage
RA6/OSC2/CLKOUT/T1CKI	RA6	ST	CMOS	Bi-directional I/O
	OSC2	—	XTAL	Crystal/Resonator
	CLKOUT	—	CMOS	Fosc/4 Output
	T1CKI	ST	—	Timer1 Clock Input
RA7/OSC1/CLKIN	RA7	ST	CMOS	Bi-directional I/O
	OSC1	XTAL	—	Crystal/Resonator
	CLKIN	ST	—	External Clock Input
RB0/INT/AN4/VR	RB0	TTL	CMOS	Bi-directional I/O
	INT	ST	—	External Interrupt
	AN4	AN	—	ADC, Comparator Input
	VR	—	AN	Internal Voltage Reference Output
RB1/AN5/VDAC	RB1	TTL	CMOS	Bi-directional I/O
	AN5	AN	—	ADC, Comparator Input
	VDAC	—	AN	DAC Output
RB2/AN6	RB2	TTL	CMOS	Bi-directional I/O
	AN6	AN	—	ADC, Comparator Input
RB3/AN7/OPA	RB3	TTL	CMOS	Bi-directional I/O
	AN7	AN	—	ADC, Comparator Input
	OPA	—	AN	OPAMP Output
RB4	RB4	TTL	CMOS	Bi-directional I/O
RB5	RB5	TTL	CMOS	Bi-directional I/O
RB6/C1/PSMC1A	RB6	TTL	CMOS	Bi-directional I/O
	C1	—	CMOS	Comparator 1 Output
	PSMC1A	—	CMOS	PSMC Output 1A

2.4 OPTION_REG Register

The OPTION_REG register is a readable and writable register which contains various control bits to configure:

- TMR0 prescaler/WDT postscaler (single assignable register known also as the prescaler)
- External INT interrupt
- TMR0
- Weak pull-ups on PORTB

Note:	To achieve a 1:1 prescaler assignment for the TMR0 register, assign the prescaler to the Watchdog Timer.

REGISTER 2-2: OPTION REGISTER (OPTION_REG: 81h, 181h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit7							bit0

bit 7 $\overline{\text{RBPU}}$: PORTB Pull-up Enable bit(1)
1 = PORTB weak pull-ups are disabled
0 = PORTB weak pull-ups are enabled by the WPUB register

bit 6 INTEDG: Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

bit 5 T0CS: TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI pin
0 = Internal instruction cycle clock (FOSC/4)

bit 4 T0SE: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI pin
0 = Increment on low-to-high transition on RA4/T0CKI pin

bit 3 PSA: Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

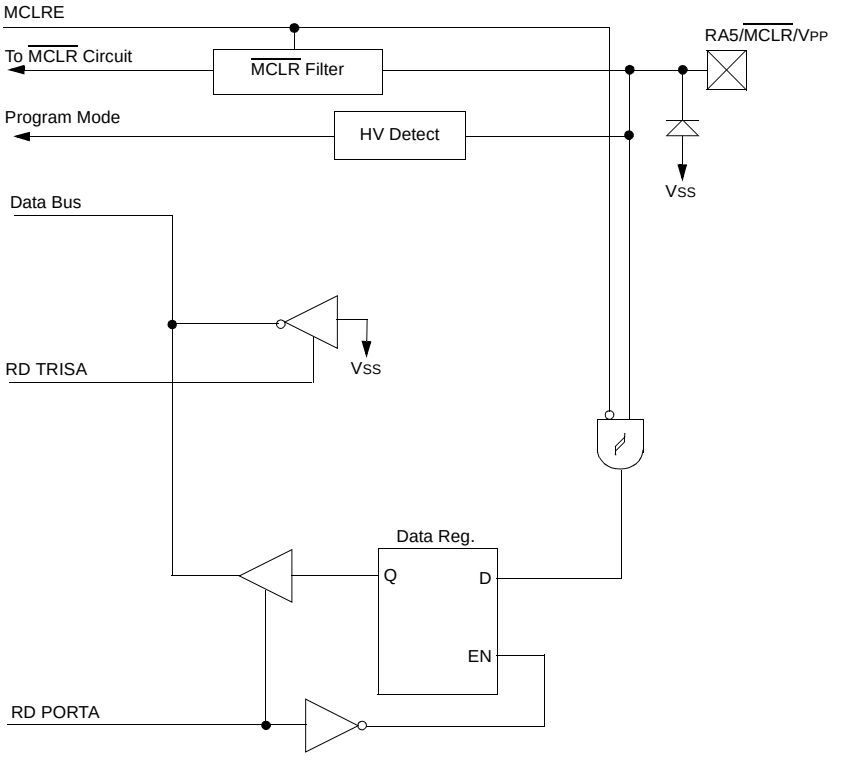
bit 2-0 PS<2:0>: Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Note 1: Individual weak pull-ups on RB pins can be enabled/disabled from the weak pull-up PORTB register (WPUB).

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

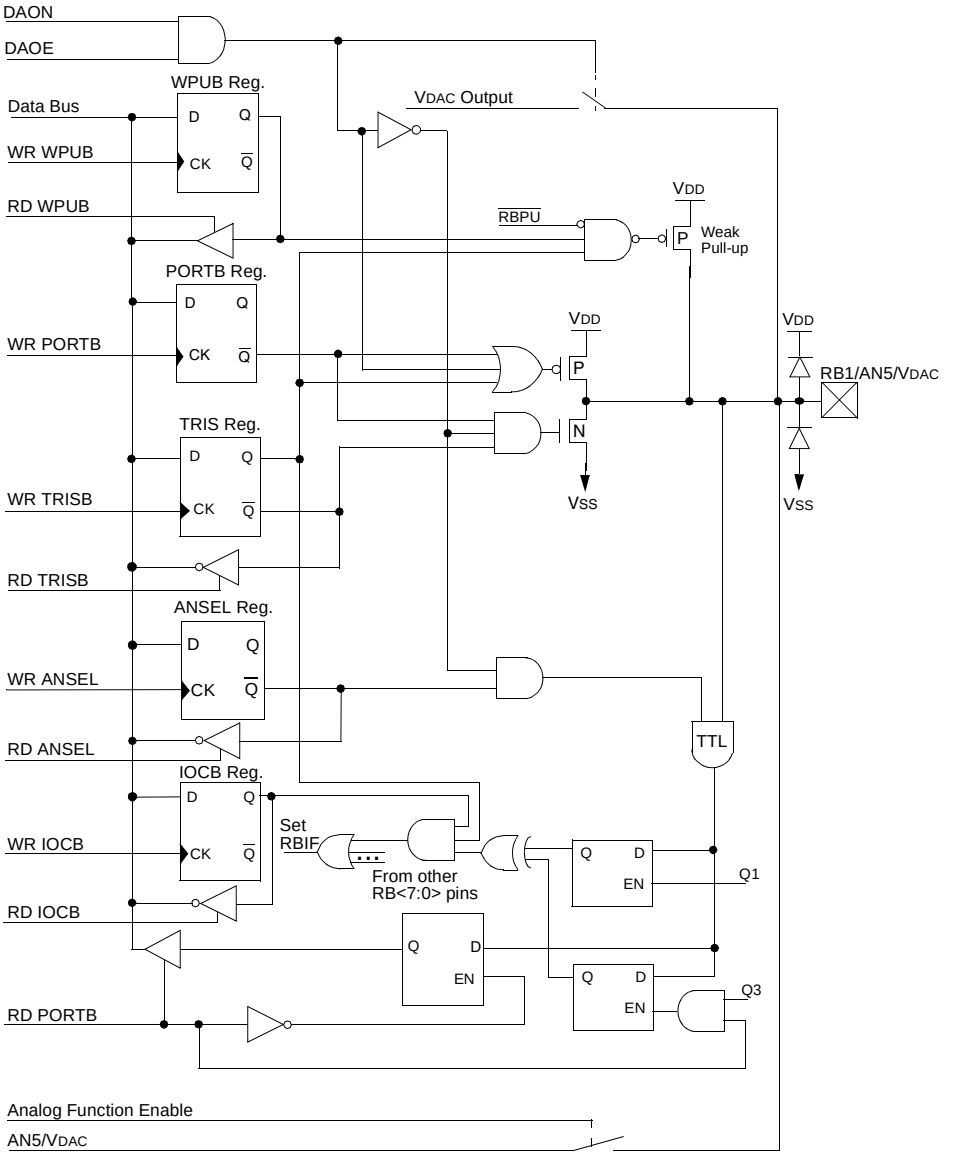
FIGURE 3-6: BLOCK DIAGRAM OF RA5/$\overline{\text{MCLR}}$/V_{PP} PIN



MCLRE(1)	TRISA<5>	FUNCTION	PORTA<5> READ
Internal	x	Digital In	Pin
External	x	$\overline{\text{MCLR}}$	0

Note 1: See Configuration Word <5>, Register 14-1.

FIGURE 3-10: BLOCK DIAGRAM OF RB1/AN5/V DAC PIN



DAON & DAOE	ANSEL<5>	TRISB<1>	FUNCTION	PORTB<1> READ
0	0	1	Digital In	Pin
0	0	0	Digital Out	Pin
0	1	x	Analog In	0
1	x	x	Analog Out	0

4.0 PROGRAM MEMORY READ (PMR)

Program memory is readable during normal operation (full VDD range). It is read by indirect addressing through the following Special Function Registers:

- PMCON1: Control
- PMDATH: Data High
- PMDATL: Data Low
- PMADRH: Address High
- PMADRL: Address Low

When interfacing to the program memory block, the PMDATH and PMDATL registers form a 2-byte word, which holds the 14-bit data. The PMADRH and PMADRL registers form a 2-byte word, which holds the 12-bit address of the program memory location being accessed. Mid-range devices have up to 8K words of program EPROM with an address range from 0h to 3FFFh. When the device contains less memory than the full address range of the PMADRH:PMARDL registers, the Most Significant bits of the PMADRH register are ignored.

4.1 PMCON1 Register

PMCON1 is the control register for program memory accesses.

Control bit RD initiates a read operation. This bit cannot be cleared, only set, in software. It is cleared in hardware at completion of the read operation.

4.2 PMDATH and PMDATL Registers

The PMDATH:PMDATL registers are loaded with the contents of program memory addressed by the PMADRH and PMADRL registers upon completion of a Program Memory Read command.

REGISTER 4-1: PROGRAM MEMORY READ CONTROL REGISTER 1 (PMCON1: 18Ch)

R-1	U-0	U-0	U-0	U-0	U-0	U-0	R/S-0
Reserved	—	—	—	—	—	—	RD
bit7							bit0

bit 7 Reserved: Read as ‘1’

bit 6-1 **Unimplemented:** Read as '0

bit 0 **RD**: Read Control bit

1 = Initiates a Program memory read (read takes 2 cycles, RD is cleared in hardware)
0 = Reserved

Legend:			
S = Settable bit			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
- n = Value at POR	’1’ = Bit is set	’0’ = Bit is cleared	x = Bit is unknown

REGISTER 4-2: PROGRAM MEMORY DATA HIGH (PMDATH: 10Eh)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	PMD13	PMD12	PMD11	PMD10	PMD9	PMD8
bit7							bit0

bit 7-6 Unimplemented: Read as '0

bit 5-0 PMD<13:8>: Program Memory Data bits
The value of the program memory word pointed to by PMADRH and PMADRL after a program memory read command.

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
- n = Value at POR	’1’ = Bit is set	’0’ = Bit is cleared	x = Bit is unknown

8.0 PROGRAMMABLE LOW VOLTAGE DETECT MODULE (PLVD)

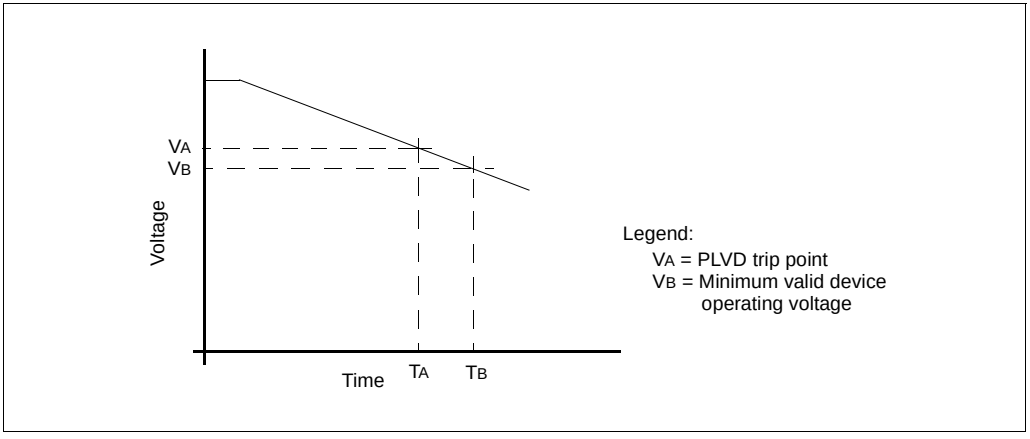
The PLVD module monitors the VDD power supply of the microcontroller and signals the microcontroller whenever VDD drops below its trip voltage. The signal acts as an 'early warning' of power-down, allowing the microcontroller to finish any critical 'housekeeping' tasks prior to completing power-down.

Figure 8-1 demonstrates a potential application of the PLVD module (typical battery operation). At time TA, the VDD supply voltage (VA) has fallen below the PLVD reference voltage. The PLVD voltage comparator then sets the LVDIF bit (PIR<7>), indicating a low voltage condition. The time between TA and TB is then available to the microcontroller for completing a 'graceful' power-down before VDD falls below VB.

Figure 8-2 is a simplified block diagram for the PLVD module, showing the VDD resistor ladder, control register, and voltage comparator.

Note:	For low power applications, current drain can be minimized by enabling the module only during regular polled testing. When not in use, the module is disabled by clearing the LVDEN bit (LVDCON<4>), which also powers down the resistor ladder between VDD and Vss.

FIGURE 8-1: TYPICAL LOW VOLTAGE DETECT APPLICATION



8.1 Control Register

The PLVD module is controlled via the LVDCON register shown in Register 8-1.

To enable the module for testing, the LVDEN bit (LVDCON<4>) must be set. This will enable the on-board voltage reference and connect the resistor ladder between VDD and Vss. Clearing LVDEN will disable the module and disconnect the resistor ladder from Vss.

The trip voltage is set by programming the LVDL<3:0> bit (LVDCON<3.0>). The voltages available are listed in Register 8-1. Note that voltages below 2.5V and above 4.75V are not available and should not be used.

The BGST bit (LVDCON<5>) is a status bit indicating that the internal reference voltage bandgap has stabilized. No test should be performed until this bit is set.

The low voltage output flag for the PLVD module is the LVDIF bit (PIR1<6>).

9.2.4 INITIATING A CONVERSION

The Analog-to-Digital conversion is initiated by setting the GO/$\overline{\text{DONE}}$ bit in ADCON0 register. When the conversion is complete, the ADC module:

- Clears the GO/$\overline{\text{DONE}}$ bit
- Sets the ADIF flag in the PIR1 register
- Generates an interrupt if the ADIE, PEIE, and GIE bits are set.

If the conversion must be aborted, the GO/$\overline{\text{DONE}}$ bit can be cleared in software. The ADRES register will not be updated with the partially completed ADC conversion sample. Instead, the ADRES will contain the value from the last completed conversion. After an aborted conversion, a 2TAD delay is required before another acquisition/conversion can be initiated. Following the delay, an input acquisition is automatically started on the selected channel.

Note:	The GO/$\overline{\text{DONE}}$ bit should NOT be set in the same instruction that turns on the ADC.

9.3 ADC Acquisition Requirements

For the ADC module to meet its specified accuracy, the internal Sample-and-Hold capacitor (CHOLD) must be allowed to charge to within ½ LSb of the voltage present on the input channel (see analog input model in Figure 9-2). The analog source resistance (RS) and the internal sampling switch resistance (RSS) will directly affect the time required to charge CHOLD. In addition, RSS will vary over the power supply voltage range (AVDD), and RS will affect the input offset voltage at the analog input (due to pin leakage current). Therefore:

1. The maximum recommended impedance for any analog sources is 10 kOhms.
2. Following any change in the analog input channel selection, a minimum acquisition delay must be observed before another conversion can begin (see Equation 9-1).

To calculate the minimum acquisition time, Equation 9-1 may be used. This equation calculates the acquisition time to within ½ LSb error, assuming an 8-bit conversion (512 steps for the PIC16C781/782 ADC). The ½ LSb error is the maximum error allowed for the ADC to meet its specified accuracy.

EQUATION 9-1: ADC MINIMUM CHARGING TIME

$$V_{HOLD} = (ADC_{REF}-(ADC_{REF}/512))\bullet(1-e^{-T_{CAP}/C_{HOLD}(R_{IC}+R_{SS}+R_S)})$$

Given: $V_{HOLD} = (ADC_{REF}/512)$, for 1/2LSb resolution

The above equation reduces to:

$$T_{CAP} = -(51.2\ pF)(1\ k\Omega + R_{SS} + R_S)\ Ln(1/511)$$

Example 9-1 shows the calculation of the minimum required acquisition time TACQ. This calculation is based on the following system assumptions.

CHOLD = 51.2 pF

RS = 10kΩ

1/2 LSb error

RSS = 7kΩ @ VDD = 5V

Note 1:	The reference voltage (ADCREF) has no effect on the equation, since it cancels itself out.
2:	The charge holding capacitor (CHOLD) is not discharged after each conversion.
3:	The maximum recommended impedance for analog sources is 10kΩ. This is required to meet the pin leakage specification.
4:	After a conversion has completed, a 1.0TAD delay must be completed before acquisition can begin again. During this time the holding capacitor is not connected to the selected ADC input channel.

EXAMPLE 9-1: CALCULATING THE MINIMUM REQUIRED ACQUISITION TIME

T_{ACQ} = Amplifier Setting Time + Holding Capacitor Charging Time + Temperature Coefficient

$T_{ACQ} = 5\ \mu s + T_{CAP} + [(Temp - 25°C)(0.05\ \mu s/°C)]$

$T_{CAP} = -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ In(1/511)$

$-51.2\ pF\ (1\ k\Omega + 7k\Omega + 10k\Omega)\ In(0.0020)$

$-51.2\ pF\ (18\ k\Omega)\ In(0.0020)$

$-0.921\ \mu s\ (-6.2364)$

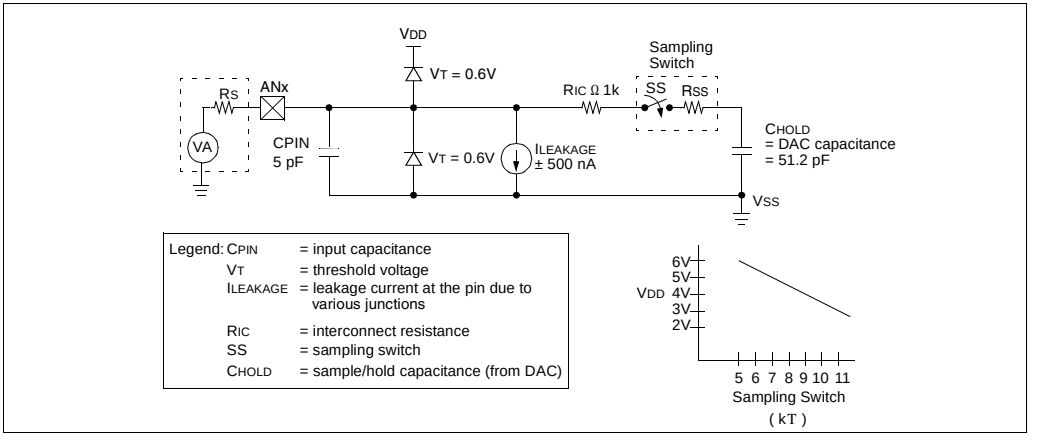
$5.747\ \mu s$

$T_{ACQ} = 5\ \mu s + 5.747\ \mu s + [(50°C - 25°C)(0.05 \mu s/°C)]$

$10.747\ \mu s + 1.25\ \mu s$

$11.997\ \mu s$

FIGURE 9-2: ANALOG INPUT MODEL



9.4 ADC Configuration and Conversion

Example 9-2 demonstrates an ADC conversion. The RA0/AN0 pin is configured as the analog input. The reference voltage selected is the device AVDD. The ADC interrupt is enabled, and the ADC conversion clock is ADRC.

Clearing the GO/$\overline{\text{DONE}}$ bit during a conversion aborts the current conversion. The ADRES register is NOT updated with the partially completed ADC conversion sample. That is, the ADRES register continues to contain the value of the last completed conversion (or the last value written to the ADRES register). After the ADC conversion is aborted, a 2TAD wait period is required before the next acquisition is started. After this 2TAD wait period, an acquisition is automatically started on the selected channel.

EXAMPLE 9-2: ADC CONVERSION

```
;*****************************************
;*  This code block will configure the ADC
;*  for polling, AVDD as reference, RC clock
;*  and RA0 input.
;*
;*  Conversion start & wait for complete
;*  polling code included.
;*
    BANKSEL    ADCON1      ; Select Bank 1
    CLRF       ADCON1      ; AVDD as VREF
    BSF        TRISA,0     ; Set RA0 as input
    BSF        ANSEL,0     ; Set RA0 as analog

    BANKSEL    ADCON0      ; Select Bank0
    MOVLW      B'11000001'
    MOVWF      ADCON0      ; RC, Ch 0, ADC on

;*****************************************
;*  Start & Wait for ADC complete, assumes
;*  minimum acquisition delay from
;*  configuration.

ADC_CNVRT
    BANKSEL    ADCON0      ; Select Bank 0
    BSF        ADCON0,GO   ; Start convert
ADC_CN_LOOP
    BTFSC      ADCON0,GO   ; Test for end
    GOTO       ADC_CN_LOOP ; If not, wait
    MOVF       ADRES,W     ; Get result
```


10.3 DAC Configuration

Example 10-1 shows a sample configuration for the DAC module. The port pin is configured, AVDD is selected for the voltage reference, and the DAC output is enabled.

EXAMPLE 10-1: DAC CONFIGURATION

```
;*  This code block will configure the DAC
;*  for AVDD Voltage Ref, and RB1/AN5/VDAC as
;*  output.

    BANKSEL TRISB          ; Select bank 1
    BSF     TRISB,1        ; Set RB1 input
    BSF     ANSEL,1        ; Set RB1 as analog

    BANKSEL DACON0         ; Select Bank 2
    CLRF    DAC            ; DAC to 00
    MOVLW   B'11000000'    ; Enable DAC output
    MOVWF   DACON0         ; Set REF = VDD

    MOVLW   DAC_VALUE
    MOVWF   DAC            ; Set DAC output
```

10.4 Effects of RESET

A device RESET forces all registers to their RESET state. This forces the following conditions:

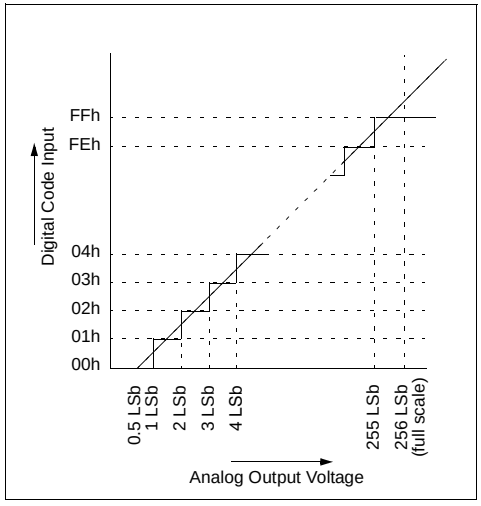
- DAC module is off
- Reference input to AVDD
- Output disabled
- DAC register is cleared

10.5 DAC Module Accuracy/Error

The accuracy/error specified for the DAC includes:

- Integral non-linearity error
- Differential non-linearity error
- Gain error
- Offset error
- Monotonicity

FIGURE 10-2: DAC TRANSFER FUNCTION



Offset error measures the first actual transition of a code versus the first ideal transition of a code. Offset error shifts the entire transfer function. Offset error can be calibrated out of a system or introduced into a system through the interaction of the output drive capability with the load impedance.

Gain error measures the maximum deviation of the last actual transition and the last ideal transition adjusted for offset error. This error appears as a change in slope of the transfer function. The difference in gain error to full scale error is that full scale does not take offset error into account. Gain error can be calibrated out by adjusting the reference voltage.

Linearity error refers to the uniformity of the voltage change with code change. Linearity errors cannot be calibrated out of the system. Integral non-linearity error measures the actual voltage output versus the ideal voltage output adjusted by the gain error for each code.

Differential non-linearity error measures the maximum actual voltage step versus the ideal voltage step. This measure is unadjusted.

TABLE 10-1: REGISTERS/BITS ASSOCIATED WITH DAC

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
11Fh	DACON0	DAON	DAOE	—	—	—	—	DARS1	DARS0	00-- --00	00-- --00
11Eh	DAC	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0	0000 0000	0000 0000
86h	TRISB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	1111 1111	1111 1111
9Dh	ANSEL	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used for DAC conversion.

NOTES:

TABLE 11-1: REGISTERS ASSOCIATED WITH THE OPA MODULE

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
11Ch	OPACON	OPAON	CMPEN	—	—	—	—	—	GBWP	00-- ---0	00-- ---0
110h	CALCON	CAL	CALERR	CALREF	—	—	—	—	—	000- ----	000- ----
9Dh	ANSEL	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	1111 1111	1111 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
85h	TRISA	PORTA Data Direction Register								1111 1111	1111 1111
11Eh	DAC	DA7	DA6	DA5	DA4	DA3	DA1	DA1	DA0	0000 0000	0000 0000
11Fh	DACON0	DAON	DAOE	—	—	—	—	DARS1	DARS0	00-- --00	00-- --00

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used for the OPA module.

NOTES:

12.1.2 COMPARATOR C2 CONTROL REGISTERS

The CM2CON0 register is a functional copy of the CM1CON0 register described in Section 12.1.1. A second control register, CM2CON1, is also present for control of an additional synchronizing feature, as well as mirrors of both comparator outputs.

12.1.2.1 Control Register CM2CON0

The CM2CON0 register, shown in Register 12-2, contains the control and status bits for Comparator C2.

Setting C2ON (CM2CON0<7>) enables Comparator C2 for operation.

Bits C2CH<1:0> (CM2CON0<1:0>) select the comparator input from the four analog pins, AN<7:4>.

Note 1: To use AN<7:4> as analog inputs, the appropriate bits must be programmed in the ANSEL register.

C2R (CM2CON0<2>) selects the reference to be used with the comparator. Setting C2R (CM2CON0<2>) selects the output of the DAC module as the reference for the comparator. Clearing C2R selects the VREF2 input on the RA2/AN2/VREF2 pin.

The output of the comparator is available internally via the C2OUT bit (CM2CON0<6>). To make the output available for an external connection, the C2OE bit (CM2CON0<5>) must be set.

Note 1: The internal output of the comparator is latched at the end of each instruction cycle. External outputs are not latched.

2: The C2 interrupt will operate correctly with C2OE set or cleared. An external output is not required for the C2 interrupt.

3: For C2 output on RB7/C2/PSMC1B/$\overline{T1G}$: (C2OE=1) & (C2ON=1) & (TRISB<7>=0) & ((SMCON=0) or ((SMCOM=0) & (SCEN=0))).

The comparator output, C2OUT, can be inverted by setting the C2POL bit (CM2CON0<4>). Clearing C2POL results in a non-inverted output. A complete table showing the output state versus input conditions and the polarity bit is shown in Table 12-3.

C2SP (CM2CON0<3>) configures the speed of the comparator. When C2SP is set, the comparator operates at its normal speed. Clearing C2SP operates the comparator in low power mode.

EXAMPLE 12-3: WINDOW COMPARATOR

```
;*  Example of Low Power Window Comparator C1
;*  This code block will configure Comparator
;*   C1 and C2 for slow speed, C1 non invert,
;*  C2 invert, input on AN4, and external
;*  References
;*
;*  Interrupt service routine included
;*
    BANKSEL     TRISA       ; Select Bank 1
    BSF         TRISA,2     ; RA2 input
    BSF         TRISA,3     ; RA3 input
    BSF         TRISB,0     ; Set RB0

    BSF         ANSEL,AN2   ; RA2 analog
    BSF         ANSEL,AN3   ; RA3 analog
    BSF         ANSEL,AN4   ; RB4 analog

    BANKSEL     CM1CON0     ; Select Bank 2
    MOVLW       B'10000000'; C1: no output
    MOVWF       CM1CON0     ; VREF1, AN4
    MOVLW       B'10010000'; C2: no output
    MOVWF       CM2CON0     ; invert,VREF1,AN4

    BANKSEL     PIE1        ; Select Bank 1
    BCF         INTCON,GIE  ; Disable Int
    BSF         PIE1,C1IE   ; Enabl C1&C2 Ints
    BSF         PIE1,C2IE
    BSF         INTCON,PEIE
    BSF         INTCON,GIE  ; Enabl Global Ints
;*********************************************
;*  WINDOW COMPARATOR ISR with context save

WC_INT_SRV_R
    MOVWF       W_SAVE      ; Save W & STATUS
    SWAPF       STATUS,W
    MOVWF       STATUS_SAV

    BANKSEL     PIR1        ; Select Bank 0
    MOVLW       B'00110000'; Save Int
    ANDWF       PIR1,W
    MOVWF       WIN_INT

;*** CLEAR C1 INTERRUPT

    BTFSS       WIN_INT,C1IF; C1 Int ?
    GOTO        TST_C2_INT
    BANKSEL     CM1CON0     ; Select Bank 2
    MOVF        CM1CON0,F   ; Clear C2 mismatch
    BANKSEL     PIR1        ; Select Bank 0
    BCF         PIR1,C1IF   ; Clear C2 Int

;*** CLEAR C2 INTERRUPT

TXT_C2_INT
    BTFSS       WIN_INT,C2IF; C2 int?
    GOTO        USER_ISR
    BANKSEL     CM2CON0     ; Select Bank 2
    MOVF        CM2CON0,F   ; Clear C2 mismatch
    BANKSEL     PIR1        ; Select Bank 0
    BCF         PIR1,C1IF   ; Clear C2 int

USER_ISR
;*** USER INTERRUPT ROUTING
;*

    SWAPF       STATUS_SAVE,W; Restore W &
                            ; STATUS
    MOVWF       STATUS
    SWAPF       W_SAVE,F
    SWAPF       W_SAVE,W
    RETFIE                  ; Return
```

15.1 Instruction Descriptions

ADDLW	Add Literal and W
Syntax:	[*label*] ADDLW k
Operands:	0 Ω k Ω 255
Operation:	(W) + k ⇓ (W)
Status Affected:	C, DC, Z
Description:	The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register.

ADDWF	Add W and f
Syntax:	[*label*] ADDWF f,d
Operands:	0 Ω f Ω 127 d ⊂ ₧7:8β
Operation:	(W) + (f) ⇓ (destination)
Status Affected:	C, DC, Z
Description:	Add the contents of the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

ANDLW	AND Literal with W
Syntax:	[*label*] ANDLW k
Operands:	0 Ω k Ω 255
Operation:	(W) .AND. (k) ⇓ (W)
Status Affected:	Z
Description:	The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register.

ANDWF	AND W with f
Syntax:	[*label*] ANDWF f,d
Operands:	0 Ω f Ω 127 d ⊂ ₧7:8β
Operation:	(W) .AND. (f) ⇓ (destination)
Status Affected:	Z
Description:	AND the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

BCF	Bit Clear f
Syntax:	[*label*] BCF f,b
Operands:	0 Ω f Ω 127 0 Ω b Ω 7
Operation:	0 ⇓ (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is cleared.

BSF	Bit Set f
Syntax:	[*label*] BSF f,b
Operands:	0 Ω f Ω 127 0 Ω b Ω 7
Operation:	1 ⇓ (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is set.

BTFSS	Bit Test f, Skip if Set
Syntax:	[*label*] BTFSS f,b
Operands:	0 Ω f Ω 127 0 Ω b < 7
Operation:	skip if (f<b>) = 1
Status Affected:	None
Description:	If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2TCY instruction.

16.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK object linker combines relocatable objects created by the MPASM assembler and the MPLAB C17 and MPLAB C18 C compilers. It can also link relocatable objects from pre-compiled libraries, using directives from a linker script.

The MPLIB object librarian is a librarian for pre-compiled code to be used with the MPLINK object linker. When a routine from a library is called from another source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications. The MPLIB object librarian manages the creation and modification of library files.

The MPLINK object linker features include:

- Integration with MPASM assembler and MPLAB C17 and MPLAB C18 C compilers.
- Allows all memory areas to be defined as sections to provide link-time flexibility.

The MPLIB object librarian features include:

- Easier linking because single libraries can be included instead of many smaller files.
- Helps keep code maintainable by grouping related modules together.
- Allows libraries to be created and modules to be added, listed, replaced, deleted or extracted.

16.5 MPLAB SIM Software Simulator

The MPLAB SIM software simulator allows code development in a PC-hosted environment by simulating the PIC series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file, or user-defined key press, to any of the pins. The execution can be performed in single step, execute until break, or trace mode.

The MPLAB SIM simulator fully supports symbolic debugging using the MPLAB C17 and the MPLAB C18 C compilers and the MPASM assembler. The software simulator offers the flexibility to develop and debug code outside of the laboratory environment, making it an excellent multi-project software development tool.

16.6 MPLAB ICE High Performance Universal In-Circuit Emulator with MPLAB IDE

The MPLAB ICE universal in-circuit emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PIC microcontrollers (MCUs). Software control of the MPLAB ICE in-circuit emulator is provided by the MPLAB Integrated Development Environment (IDE), which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 2000 is a full-featured emulator system with enhanced trace, trigger and data monitoring features. Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB ICE in-circuit emulator allows expansion to support new PIC microcontrollers.

The MPLAB ICE in-circuit emulator system has been designed as a real-time emulation system, with advanced features that are generally found on more expensive development tools. The PC platform and Microsoft® Windows environment were chosen to best make these features available to you, the end user.

16.7 ICEPIC In-Circuit Emulator

The ICEPIC low cost, in-circuit emulator is a solution for the Microchip Technology PIC16C5X, PIC16C6X, PIC16C7X and PIC16CXXX families of 8-bit One-Time-Programmable (OTP) microcontrollers. The modular system can support different subsets of PIC16C5X or PIC16CXXX products through the use of interchangeable personality modules, or daughter boards. The emulator is capable of emulating without target application circuitry being present.

16.8 MPLAB ICD In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD, is a powerful, low cost, run-time development tool. This tool is based on the FLASH PIC MCUs and can be used to develop for this and other PIC microcontrollers. The MPLAB ICD utilizes the in-circuit debugging capability built into the FLASH devices. This feature, along with Microchip's In-Circuit Serial Programming™ protocol, offers cost-effective in-circuit FLASH debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time.

16.9 PRO MATE II Universal Device Programmer

The PRO MATE II universal device programmer is a full-featured programmer, capable of operating in stand-alone mode, as well as PC-hosted mode. The PRO MATE II device programmer is CE compliant.

The PRO MATE II device programmer has programmable V_{DD} and V_{PP} supplies, which allow it to verify programmed memory at V_{DD} min and V_{DD} max for maximum reliability. It has an LCD display for instructions and error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode, the PRO MATE II device programmer can read, verify, or program PIC devices. It can also set code protection in this mode.

16.10 PICSTART Plus Entry Level Development Programmer

The PICSTART Plus development programmer is an easy-to-use, low cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient.

The PICSTART Plus development programmer supports all PIC devices with up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus development programmer is CE compliant.

16.11 PICDEM 1 Low Cost PIC MCU Demonstration Board

The PICDEM 1 demonstration board is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The user can program the sample microcontrollers provided with the PICDEM 1 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The user can also connect the PICDEM 1 demonstration board to the MPLAB ICE in-circuit emulator and download the firmware to the emulator for testing. A prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push button switches and eight LEDs connected to PORTB.

16.12 PICDEM 2 Low Cost PIC16CXX Demonstration Board

The PICDEM 2 demonstration board is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM 2 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 2 demonstration board to test firmware. A prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push button switches, a potentiometer for simulated analog input, a serial EEPROM to demonstrate usage of the I^2C™ bus and separate headers for connection to an LCD module and a keypad.

NOTES:

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7828
Fax: 886-7-330-9305

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

11/29/12

