



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	Obsolete
Core Processor	dsPIC
Core Size	16-Bit
Speed	20 MIPS
Connectivity	CANbus, I <sup>2</sup> C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, Motor Control PWM, QEI, POR, PWM, WDT
Number of I/O	68
Program Memory Size	144KB (48K x 24)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	80-TQFP
Supplier Device Package	80-TQFP (14x14)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/dspic30f6010t-20e-pf">https://www.e-xfl.com/product-detail/microchip-technology/dspic30f6010t-20e-pf</a>

# dsPIC30F Flash Programming Specification

## 2.2 Pins Used During Programming

The pins identified in [Table 2-1](#) are used for device programming. Refer to the appropriate device data sheet for complete pin descriptions.

**TABLE 2-1: dsPIC30F PIN DESCRIPTIONS DURING PROGRAMMING**

Pin Name	Pin Type	Pin Description
MCLR/VPP	P	Programming Enable
VDD	P	Power Supply
VSS	P	Ground
PGC	I	Serial Clock
PGD	I/O	Serial Data

**Legend:** I = Input, O = Output, P = Power

## 2.3 Program Memory Map

The program memory space extends from 0x0 to 0xFFFFFE. Code storage is located at the base of the memory map and supports up to 144 Kbytes (48K instruction words). Code is stored in three, 48 Kbyte memory panels that reside on-chip. [Table 2-2](#) shows the location and program memory size of each device.

Locations 0x800000 through 0x8005BE are reserved for executive code memory. This region stores either the programming executive or debugging executive. The programming executive is used for device programming, while the debug executive is used for in-circuit debugging. This region of memory cannot be used to store user code.

Locations 0xF80000 through 0xF8000E are reserved for the Configuration registers. The bits in these registers may be set to select various device options, and are described in [Section 5.7 “Configuration Bits Programming”](#).

Locations 0xFF0000 and 0xFF0002 are reserved for the Device ID registers. These bits can be used by the programmer to identify what device type is being programmed and are described in [Section 10.0 “Device ID”](#). The device ID reads out normally, even after code protection is applied.

[Figure 2-2](#) illustrates the memory map for the dsPIC30F devices.

## 2.4 Data EEPROM Memory

The Data EEPROM array supports up to 4 Kbytes of data and is located in one memory panel. It is mapped in program memory space, residing at the end of User Memory Space (see [Figure 2-2](#)). [Table 2-2](#) shows the location and size of data EEPROM in each device.

**TABLE 2-2: CODE MEMORY AND DATA EEPROM MAP AND SIZE**

Device	Code Memory map (Size in Instruction Words)	Data EEPROM Memory Map (Size in Bytes)
dsPIC30F2010	0x000000-0x001FFE (4K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F2011	0x000000-0x001FFE (4K)	None (0K)
dsPIC30F2012	0x000000-0x001FFE (4K)	None (0K)
dsPIC30F3010	0x000000-0x003FFE (8K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F3011	0x000000-0x003FFE (8K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F3012	0x000000-0x003FFE (8K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F3013	0x000000-0x003FFE (8K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F3014	0x000000-0x003FFE (8K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F4011	0x000000-0x007FFE (16K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F4012	0x000000-0x007FFE (16K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F4013	0x000000-0x007FFE (16K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F5011	0x000000-0x00AFFE (22K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F5013	0x000000-0x00AFFE (22K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F5015	0x000000-0x00AFFE (22K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F5016	0x000000-0x00AFFE (22K)	0x7FFC00-0x7FFFFE (1K)
dsPIC30F6010	0x000000-0x017FFE (48K)	0x7FF000-0x7FFFFE (4K)
dsPIC30F6010A	0x000000-0x017FFE (48K)	0x7FF000-0x7FFFFF (4K)
dsPIC30F6011	0x000000-0x015FFE (44K)	0x7FF800-0x7FFFFE (2K)
dsPIC30F6011A	0x000000-0x015FFE (44K)	0x7FF800-0x7FFFFE (2K)
dsPIC30F6012	0x000000-0x017FFE (48K)	0x7FF000-0x7FFFFE (4K)
dsPIC30F6012A	0x000000-0x017FFE (48K)	0x7FF000-0x7FFFFE (4K)
dsPIC30F6013	0x000000-0x015FFE (44K)	0x7FF800-0x7FFFFE (2K)
dsPIC30F6013A	0x000000-0x015FFE (44K)	0x7FF800-0x7FFFFE (2K)
dsPIC30F6014	0x000000-0x017FFE (48K)	0x7FF000-0x7FFFFE (4K)
dsPIC30F6014A	0x000000-0x017FFE (48K)	0x7FF000-0x7FFFFE (4K)
dsPIC30F6015	0x000000-0x017FFE (48K)	0x7FF000-0x7FFFFE (4K)

# dsPIC30F Flash Programming Specification

## 5.0 DEVICE PROGRAMMING

### 5.1 Overview of the Programming Process

Once the programming executive has been verified in memory (or loaded if not present), the dsPIC30F can be programmed using the command set shown in [Table 5-1](#). A detailed description for each command is provided in [Section 8.0 “Programming Executive Commands”](#).

**TABLE 5-1: COMMAND SET SUMMARY**

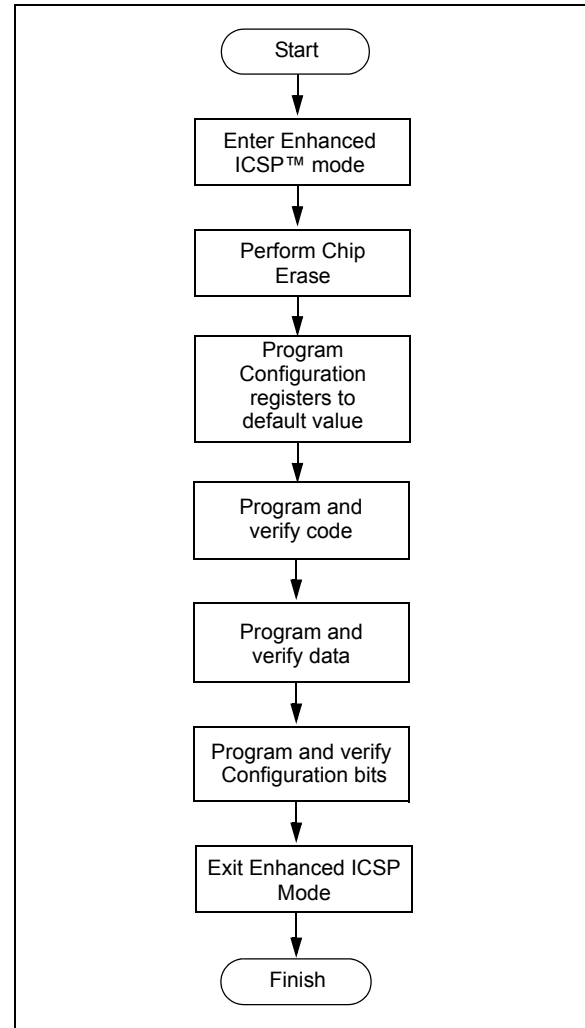
Command	Description
SCHECK	Sanity check
READD	Read data EEPROM, Configuration registers and device ID
READP	Read code memory
PROGD	Program one row of data EEPROM and verify
PROGP	Program one row of code memory and verify
PROGC	Program Configuration bits and verify
ERASEB	Bulk Erase, or erase by segment
ERASED	Erase data EEPROM
ERASEP	Erase code memory
QBLANK	Query if the code memory and data EEPROM are blank
QVER	Query the software version

A high-level overview of the programming process is illustrated in [Figure 5-1](#). The process begins by entering Enhanced ICSP mode. The chip is then bulk erased, which clears all memory to ‘1’ and allows the device to be programmed. The Chip Erase is verified before programming begins. Next, the code memory, data Flash and Configuration bits are programmed. As these memories are programmed, they are each verified to ensure that programming was successful. If no errors are detected, the programming is complete and Enhanced ICSP mode is exited. If any of the verifications fail, the procedure should be repeated, starting from the Chip Erase.

If Advanced Security features are enabled, then individual Segment Erase operations need to be performed, based on user selections (i.e., based on the specific needs of the user application). The specific operations that are used typically depend on the order in which various segments need to be programmed for a given application or system.

[Section 5.2 “Entering Enhanced ICSP Mode”](#) through [Section 5.8 “Exiting Enhanced ICSP Mode”](#) describe the programming process in detail.

**FIGURE 5-1: PROGRAMMING FLOW**



# dsPIC30F Flash Programming Specification

**TABLE 5-5: FOSC CONFIGURATION BITS DESCRIPTION FOR dsPIC30F4011/4012 AND dsPIC30F5011/5013**

Bit Field	Register	Description
FCKSM<1:0>	FOSC	<b>Clock Switching Mode</b> 1x = Clock switching is disabled, Fail-Safe Clock Monitor is disabled 01 = Clock switching is enabled, Fail-Safe Clock Monitor is disabled 00 = Clock switching is enabled, Fail-Safe Clock Monitor is enabled
FOS<1:0>	FOSC	<b>Oscillator Source Selection on POR</b> 11 = Primary Oscillator 10 = Internal Low-Power RC Oscillator 01 = Internal Fast RC Oscillator 00 = Low-Power 32 kHz Oscillator (Timer1 Oscillator)
FPR<3:0>	FOSC	<b>Primary Oscillator Mode</b> 1111 = ECIO w/PLL 16X – External Clock mode with 16X PLL. OSC2 pin is I/O 1110 = ECIO w/PLL 8X – External Clock mode with 8X PLL. OSC2 pin is I/O 1101 = ECIO w/PLL 4X – External Clock mode with 4X PLL. OSC2 pin is I/O 1100 = ECIO – External Clock mode. OSC2 pin is I/O 1011 = EC – External Clock mode. OSC2 pin is system clock output (Fosc/4) 1010 = FRC w/PLL 8x – Internal fast RC oscillator with 8x PLL. OSC2 pin is I/O 1001 = ERC – External RC Oscillator mode. OSC2 pin is system clock output (Fosc/4) 1000 = ERCIO – External RC Oscillator mode. OSC2 pin is I/O 0111 = XT w/PLL 16X – XT Crystal Oscillator mode with 16X PLL 0110 = XT w/PLL 8X – XT Crystal Oscillator mode with 8X PLL 0101 = XT w/PLL 4X – XT Crystal Oscillator mode with 4X PLL 0100 = XT – XT Crystal Oscillator mode (4 MHz-10 MHz crystal) 0011 = FRC w/PLL 16x – Internal fast RC oscillator with 16x PLL. OSC2 pin is I/O 0010 = HS – HS Crystal Oscillator mode (10 MHz-25 MHz crystal) 0001 = FRC w/PLL 4x – Internal fast RC oscillator with 4x PLL. OSC2 pin is I/O 0000 = XTL – XTL Crystal Oscillator mode (200 kHz-4 MHz crystal)

# dsPIC30F Flash Programming Specification

**TABLE 5-6: FOSC CONFIGURATION BITS DESCRIPTION FOR dsPIC30F2011/2012, dsPIC30F3010/3011/3012/3013/3014, dsPIC30F4013, dsPIC30F5015/5016, dsPIC30F6010A/6011A/6012A/6013A/6014A AND dsPIC30F6015 (CONTINUED)**

Bit Field	Register	Description
FPR<4:0>	FOSC	<b>Alternate Oscillator Mode (when FOS&lt;2:0&gt; = 011b)</b> 1xxxx = Reserved (do not use) 0111x = Reserved (do not use) 01101 = Reserved (do not use) 01100 = ECIO – External clock. OSC2 pin is I/O 01011 = EC – External clock. OSC2 pin is system clock output (Fosc/4) 01010 = Reserved (do not use) 01001 = ERC – External RC oscillator. OSC2 pin is system clock output (Fosc/4) 01000 = ERCIO – External RC oscillator. OSC2 pin is I/O 00111 = Reserved (do not use) 00110 = Reserved (do not use) 00101 = Reserved (do not use) 00100 = XT – XT crystal oscillator (4 MHz-10 MHz crystal) 00010 = HS – HS crystal oscillator (10 MHz-25 MHz crystal) 00001 = Reserved (do not use) 00000 = XTL – XTL crystal oscillator (200 kHz-4 MHz crystal)

**TABLE 5-10: dsPIC30F CONFIGURATION REGISTERS (FOR dsPIC30F2011/2012, dsPIC30F3010/3011/3012/3013/3014, dsPIC30F4013 AND dsPIC30F5015/5016)**

Address	Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xF80000	FOSC	FCKSM<1:0>		—	—	—	FOS<2:0>			—	—	—	FPR<4:0>				
0xF80002	FWDT	FWDTEN	—	—	—	—	—	—	—	—	—	FWPSA<1:0>		FWPSB<3:0>			
0xF80004	FBORPOR	MCLREN	—	—	—	—	PWMPIN <sup>(1)</sup>	HPOL <sup>(1)</sup>	LPOL <sup>(1)</sup>	BOREN	—	BORV<1:0>		—	—	FPWRT<1:0>	
0xF80006	FBS	—	—	Reserved <sup>(2)</sup>		—	—	—	Reserved <sup>(2)</sup>	—	—	—	—	Reserved <sup>(2)</sup>			
0xF80008	FSS	—	—	Reserved <sup>(2)</sup>		—	—	Reserved <sup>(2)</sup>		—	—	—	—	Reserved <sup>(2)</sup>			
0xF8000A	FGS	—	—	—	—	—	—	—	—	—	—	—	—	—	Reserved <sup>(3)</sup>	GCP	GWRP
0xF8000C	FICD	BKBUG	COE	—	—	—	—	—	—	—	—	—	—	—	—	ICS<1:0>	

**Note** 1: On the 2011, 2012, 3012, 3013, 3014 and 4013, these bits are reserved (read as '1' and must be programmed as '1').  
2: Reserved bits read as '1' and must be programmed as '1'.  
3: The FGS<2> bit is a read-only copy of the GCP bit (FGS<1>).

**TABLE 5-11: dsPIC30F CONFIGURATION REGISTERS (FOR dsPIC30F6010A/6011A/6012A/6013A/6014A AND dsPIC30F6015)**

Address	Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xF80000	FOSC	FCKSM<1:0>		—	—	—	FOS<2:0>			—	—	—	FPR<4:0>				
0xF80002	FWDT	FWDTEN	—	—	—	—	—	—	—	—	—	FWPSA<1:0>		FWPSB<3:0>			
0xF80004	FBORPOR	MCLREN	—	—	—	—	PWMPIN <sup>(1)</sup>	HPOL <sup>(1)</sup>	LPOL <sup>(1)</sup>	BOREN	—	BORV<1:0>		—	—	FPWRT<1:0>	
0xF80006	FBS	—	—	RBS<1:0>		—	—	—	EBS	—	—	—	—	BSS<2:0>			BWRP
0xF80008	FSS	—	—	RSS<1:0>		—	—	ESS<1:0>		—	—	—	—	SSS<2:0>			SWRP
0xF8000A	FGS	—	—	—	—	—	—	—	—	—	—	—	—	—	GSS<1:0>		GWRP
0xF8000C	FICD	BKBUG	COE	—	—	—	—	—	—	—	—	—	—	—	—	ICS<1:0>	

**Note** 1: On the 6011A, 6012A, 6013A and 6014A, these bits are reserved (read as '1' and must be programmed as '1').

# dsPIC30F Flash Programming Specification

## 8.0 PROGRAMMING EXECUTIVE COMMANDS

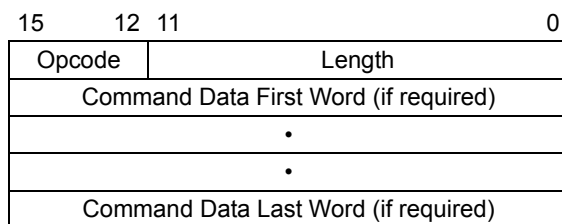
### 8.1 Command Set

The programming executive command set is shown in [Table 8-1](#). This table contains the opcode, mnemonic, length, time out and description for each command. Functional details on each command are provided in the command descriptions (see [Section 8.5 “Command Descriptions”](#)).

### 8.2 Command Format

All programming executive commands have a general format consisting of a 16-bit header and any required data for the command (see [Figure 8-1](#)). The 16-bit header consists of a 4-bit opcode field, which is used to identify the command, followed by a 12-bit command length field.

**FIGURE 8-1: COMMAND FORMAT**



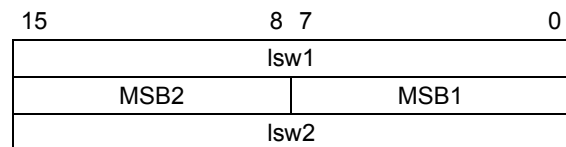
The command opcode must match one of those in the command set. Any command that is received which does not match the list in [Table 8-1](#) will return a “NACK” response (see [Section 9.2.1 “Opcode Field”](#)).

The command length is represented in 16-bit words since the SPI operates in 16-bit mode. The programming executive uses the Command Length field to determine the number of words to read from the SPI port. If the value of this field is incorrect, the command will not be properly received by the programming executive.

### 8.3 Packed Data Format

When 24-bit instruction words are transferred across the 16-bit SPI interface, they are packed to conserve space using the format shown in [Figure 8-2](#). This format minimizes traffic over the SPI and provides the programming executive with data that is properly aligned for performing table write operations.

**FIGURE 8-2: PACKED INSTRUCTION WORD FORMAT**



lswx: Least significant 16 bits of instruction word

MSBx: Most Significant Byte of instruction word

**Note:** When the number of instruction words transferred is odd, MSB2 is zero and lsw2 cannot be transmitted.

### 8.4 Programming Executive Error Handling

The programming executive will “NACK” all unsupported commands. Additionally, due to the memory constraints of the programming executive, no checking is performed on the data contained in the Programmer command. It is the responsibility of the programmer to command the programming executive with valid command arguments, or the programming operation may fail. Additional information on error handling is provided in [Section 9.2.3 “QE\\_Code Field”](#).

# dsPIC30F Flash Programming Specification

## 8.5.5 PROGP COMMAND

15	12	11	8	7	0
Opcode		Length			
Reserved			Addr_MSB		
Addr_LS					
D_1					
D_2					
...					
D_N					

Field	Description
Opcode	0x5
Length	0x33
Reserved	0x0
Addr_MSB	MSB of 24-bit destination address
Addr_LS	LS 16 bits of 24-bit destination address
D_1	16-bit data word 1
D_2	16-bit data word 2
...	16-bit data word 3 through 47
D_48	16-bit data word 48

The **PROGP** command instructs the programming executive to program one row of code memory (32 instruction words) to the specified memory address. Programming begins with the row address specified in the command. The destination address should be a multiple of 0x40.

The data to program to memory, located in command words D\_1 through D\_48, must be arranged using the packed instruction word format shown in [Figure 8-2](#).

After all data has been programmed to code memory, the programming executive verifies the programmed data against the data in the command.

### Expected Response (2 words):

0x1500  
0x0002

**Note:** Refer to [Table 5-2](#) for code memory size information.

## 8.5.6 PROGC COMMAND

15	12	11	8	7	0
Opcode		Length			
Reserved			Addr_MSB		
Addr_LS					
Data					

Field	Description
Opcode	0x6
Length	0x4
Reserved	0x0
Addr_MSB	MSB of 24-bit destination address
Addr_LS	LS 16 bits of 24-bit destination address
Data	Data to program

The **PROGC** command programs data to the specified Configuration register and verifies the programming. Configuration registers are 16 bits wide, and this command allows one Configuration register to be programmed.

### Expected Response (2 words):

0x1600  
0x0002

**Note:** This command can only be used for programming Configuration registers.



# dsPIC30F Flash Programming Specification

**TABLE 10-3: DEVICE ID BITS DESCRIPTION**

Bit Field	Register	Description
DEVID<15:0>	DEVID	Encodes the device ID.
PROC<3:0>	DEVREV	Encodes the process of the device (always read as 0x001).
REV<5:0>	DEVREV	Encodes the major revision number of the device. 000000 = A 000001 = B 000010 = C
DOT<5:0>	DEVREV	Encodes the minor revision number of the device. 000000 = 0 000001 = 1 000010 = 2 000011 = 3
<p>Examples:</p> <p>Rev A.1 = 0000 0000 0000 0001</p> <p>Rev A.2 = 0000 0000 0000 0010</p> <p>Rev B.0 = 0000 0000 0100 0000</p> <p>This formula applies to all dsPIC30F devices, with the exception of the following:</p> <ul style="list-style-type: none"><li>• dsPIC30F6010</li><li>• dsPIC30F6011</li><li>• dsPIC30F6012</li><li>• dsPIC30F6013</li><li>• dsPIC30F6014</li></ul> <p>Refer to <a href="#">Table 10-1</a> for the actual revision IDs.</p>		

# dsPIC30F Flash Programming Specification

## 11.0 ICSP™ MODE

### 11.1 ICSP Mode

ICSP mode is a special programming protocol that allows you to read and write to the dsPIC30F programming executive. The ICSP mode is the second (and slower) method used to program the device. This mode also has the ability to read the contents of executive memory to determine whether the programming executive is present. This capability is accomplished by applying control codes and instructions serially to the device using pins PGC and PGD.

In ICSP mode, the system clock is taken from the PGC pin, regardless of the device's oscillator Configuration bits. All instructions are first shifted serially into an internal buffer, then loaded into the Instruction register and executed. No program fetching occurs from internal memory. Instructions are fed in 24 bits at a time. PGD is used to shift data in and PGC is used as both the serial shift clock and the CPU execution clock.

Data is transmitted on the rising edge and latched on the falling edge of PGC. For all data transmissions, the Least Significant bit (LSb) is transmitted first.

**Note 1:** During ICSP operation, the operating frequency of PGC must not exceed 5 MHz.

**2:** Because ICSP is slower, it is recommended that only Enhanced ICSP (E-ICSP) mode be used for device programming, as described in [Section 5.1 "Overview of the Programming Process"](#).

### 11.2 ICSP Operation

Upon entry into ICSP mode, the CPU is idle. Execution of the CPU is governed by an internal state machine. A 4-bit control code is clocked in using PGC and PGD, and this control code is used to command the CPU (see [Table 11-1](#)).

The SIX control code is used to send instructions to the CPU for execution, while the REGOUT control code is used to read data out of the device via the VISI register. The operation details of ICSP mode are provided in [Section 11.2.1 "SIX Serial Instruction Execution"](#) and [Section 11.2.2 "REGOUT Serial Instruction Execution"](#).

**TABLE 11-1: CPU CONTROL CODES IN ICSP™ MODE**

4-bit Control Code	Mnemonic	Description
0000b	SIX	Shift in 24-bit instruction and execute.
0001b	REGOUT	Shift out the VISI register.
0010b-1111b	N/A	Reserved.

#### 11.2.1 SIX SERIAL INSTRUCTION EXECUTION

The SIX control code allows execution of dsPIC30F assembly instructions. When the SIX code is received, the CPU is suspended for 24 clock cycles as the instruction is then clocked into the internal buffer. Once the instruction is shifted in, the state machine allows it to be executed over the next four clock cycles. While the received instruction is executed, the state machine simultaneously shifts in the next 4-bit command (see [Figure 11-2](#)).

**Note 1:** Coming out of the ICSP entry sequence, the first 4-bit control code is always forced to SIX and a forced NOP instruction is executed by the CPU. Five additional PGC clocks are needed on start-up, thereby resulting in a 9-bit SIX command instead of the normal 4-bit SIX command. After the forced SIX is clocked in, ICSP operation resumes as normal (the next 24 clock cycles load the first instruction word to the CPU). See [Figure 11-1](#) for details.

**2:** TBLRDH, TBLRDL, TBLWTH and TBLWTL instructions must be followed by a NOP instruction.

# dsPIC30F Flash Programming Specification

Table 11-4 shows the ICSP programming process for bulk-erasing program memory. This process includes the ICSP command code, which must be transmitted (for each instruction) to the Least Significant bit first using the PGC and PGD pins (see Figure 11-2).

If an individual Segment Erase operation is required, the NVMCON value must be replaced by the value for the corresponding Segment Erase operation.

**Note:** Program memory must be erased before writing any data to program memory.

**TABLE 11-4: SERIAL INSTRUCTION EXECUTION FOR BULK ERASING PROGRAM MEMORY (ONLY IN NORMAL-VOLTAGE SYSTEMS)**

Command (Binary)	Data (Hexadecimal)	Description
<b>Step 1: Exit the Reset vector.</b>		
0000	040100	GOTO 0x100
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 2: Set NVMCON to program the FBS Configuration register.<sup>(1)</sup></b>		
0000	24008A	MOV #0x4008, W10
0000	883B0A	MOV W10, NVMCON
<b>Step 3: Initialize the TBLPAG and write pointer (W7) for TBLWT instruction for Configuration register.<sup>(1)</sup></b>		
0000	200F80	MOV #0xF8, W0
0000	880190	MOV W0, TBLPAG
0000	200067	MOV #0x6, W7
<b>Step 4: Load the Configuration Register data to W6.<sup>(1)</sup></b>		
0000	EB0300	CLR W6
0000	000000	NOP
<b>Step 5: Load the Configuration Register write latch. Advance W7 to point to next Configuration register.<sup>(1)</sup></b>		
0000	BB1B86	TBLWTL W6, [W7++]
<b>Step 6: Unlock the NVMCON for programming the Configuration register.<sup>(1)</sup></b>		
0000	200558	MOV #0x55, W8
0000	200AA9	MOV #0xAA, W9
0000	883B38	MOV W8, NVMKEY
0000	883B39	MOV W9, NVMKEY
<b>Step 7: Initiate the programming cycle.<sup>(1)</sup></b>		
0000	A8E761	BSET NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
—	—	Externally time 2 ms
0000	000000	NOP
0000	000000	NOP
0000	A9E761	BCLR NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
<b>Step 8: Repeat steps 5-7 one time to program 0x0000 to RESERVED2 Configuration register.<sup>(1)</sup></b>		
<b>Step 9: Set the NVMCON to erase all Program Memory.</b>		
00000	2407FA	MOV #0x407F, W10
0000	883B0A	MOV W10, NVMCON
<b>Step 10: Unlock the NVMCON for programming.</b>		

**Note 1:** Steps 2-8 are only required for the dsPIC30F5011/5013 devices. These steps may be skipped for all other devices in the dsPIC30F family.

# dsPIC30F Flash Programming Specification

**TABLE 11-4: SERIAL INSTRUCTION EXECUTION FOR BULK ERASING PROGRAM MEMORY (ONLY IN NORMAL-VOLTAGE SYSTEMS) (CONTINUED)**

Command (Binary)	Data (Hexadecimal)	Description
0000	200558	MOV #0x55, W8
0000	883B38	MOV W8, NVMKEY
0000	200AA9	MOV #0xAA, W9
0000	883B39	MOV W9, NVMKEY
<b>Step 11: Initiate the erase cycle.</b>		
0000	A8E761	BSET NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
—	—	Externally time 'P13a' ms (see <a href="#">Section 13.0 “AC/DC Characteristics and Timing Requirements”</a> )
0000	000000	NOP
0000	000000	NOP
0000	A9E761	BCLR NVMCON, #WR
0000	000000	NOP
0000	000000	NOP

**Note 1:** Steps 2-8 are only required for the dsPIC30F5011/5013 devices. These steps may be skipped for all other devices in the dsPIC30F family.

# dsPIC30F Flash Programming Specification

## 11.6 Erasing Program Memory in Low-Voltage Systems

The procedure for erasing program memory (all code memory and data memory) in low-voltage systems (with VDD between 2.5 volts and 4.5 volts) is quite different than the procedure for erasing program memory in normal-voltage systems. Instead of using a Bulk Erase operation, each region of memory must be individually erased by row. Namely, all of the code memory, executive memory and data memory must be erased one row at a time. This procedure is detailed in [Table 11-5](#).

Due to security restrictions, the FBS, FSS and FGS register cannot be erased in low-voltage systems. Once any bits in the FGS register are programmed to '0', they can only be set back to '1' by performing a Bulk Erase in a normal-voltage system. Alternatively, a Segment Erase operation can be performed instead of a Bulk Erase.

Normal-voltage systems can also be used to erase program memory as shown in [Table 11-5](#). However, since this method is more time-consuming and does not clear the code-protect bits, it is not recommended.

**Note:** Program memory must be erased before writing any data to program memory.

**TABLE 11-5: SERIAL INSTRUCTION EXECUTION FOR ERASING PROGRAM MEMORY (EITHER IN LOW-VOLTAGE OR NORMAL-VOLTAGE SYSTEMS)**

Command (Binary)	Data (Hexadecimal)	Description
<b>Step 1: Exit the Reset vector.</b>		
0000	040100	GOTO 0x100
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 2: Initialize NVMADR and NVMADRU to erase code memory and initialize W7 for row address updates.</b>		
0000	EB0300	CLR W6
0000	883B16	MOV W6, NVMADR
0000	883B26	MOV W6, NVMADRU
0000	200407	MOV #0x40, W7
<b>Step 3: Set NVMCON to erase 1 row of code memory.</b>		
0000	24071A	MOV #0x4071, W10
0000	883B0A	MOV W10, NVMCON
<b>Step 4: Unlock the NVMCON to erase 1 row of code memory.</b>		
0000	200558	MOV #0x55, W8
0000	883B38	MOV W8, NVMKEY
0000	200AA9	MOV #0xAA, W9
0000	883B39	MOV W9, NVMKEY
<b>Step 5: Initiate the erase cycle.</b>		
0000	A8E761	BSET NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
—	—	Externally time 'P13a' ms (see <a href="#">Section 13.0 “AC/DC Characteristics and Timing Requirements”</a> )
0000	000000	NOP
0000	000000	NOP
0000	A9E761	BCLR NVMCON, #WR
0000	000000	NOP
0000	000000	NOP

# dsPIC30F Flash Programming Specification

**TABLE 11-5: SERIAL INSTRUCTION EXECUTION FOR ERASING PROGRAM MEMORY  
(EITHER IN LOW-VOLTAGE OR NORMAL-VOLTAGE SYSTEMS) (CONTINUED)**

Command (Binary)	Data (Hexadecimal)	Description
<b>Step 6:</b> Update the row address stored in NVMADRU:NVMADR. When W6 rolls over to 0x0, NVMADRU must be incremented.		
0000	430307	ADD W6, W7, W6
0000	AF0042	BTSC SR, #C
0000	EC2764	INC NVMADRU
0000	883B16	MOV W6, NVMADR
<b>Step 7:</b> Reset device internal PC.		
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 8:</b> Repeat Steps 3-7 until all rows of code memory are erased.		
<b>Step 9:</b> Initialize NVMADR and NVMADRU to erase executive memory and initialize W7 for row address updates.		
0000	EB0300	CLR W6
0000	883B16	MOV W6, NVMADR
0000	200807	MOV #0x80, W7
0000	883B27	MOV W7, NVMADRU
0000	200407	MOV #0x40, W7
<b>Step 10:</b> Set NVMCON to erase 1 row of executive memory.		
0000	24071A	MOV #0x4071, W10
0000	883B0A	MOV W10, NVMCON
<b>Step 11:</b> Unlock the NVMCON to erase 1 row of executive memory.		
0000	200558	MOV #0x55, W8
0000	883B38	MOV W8, NVMKEY
0000	200AA9	MOV #0xAA, W9
0000	883B39	MOV W9, NVMKEY
<b>Step 12:</b> Initiate the erase cycle.		
0000	A8E761	BSET NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
—	—	Externally time 'P13a' ms (see <a href="#">Section 13.0 “AC/DC Characteristics and Timing Requirements”</a> )
0000	000000	NOP
0000	000000	NOP
0000	A9E761	BCLR NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
<b>Step 13:</b> Update the row address stored in NVMADR.		
0000	430307	ADD W6, W7, W6
0000	883B16	MOV W6, NVMADR
<b>Step 14:</b> Reset device internal PC.		
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 15:</b> Repeat Steps 10-14 until all 24 rows of executive memory are erased.		
<b>Step 16:</b> Initialize NVMADR and NVMADRU to erase data memory and initialize W7 for row address updates.		
0000	2XXXX6	MOV #<lower 16-bits of starting Data EEPROM address>, W6
0000	883B16	MOV W6, NVMADR
0000	2007F6	MOV #0x7F, W6
0000	883B16	MOV W6, NVMADRU
0000	200207	MOV #0x20, W7
<b>Step 17:</b> Set NVMCON to erase 1 row of data memory.		
0000	24075A	MOV #0x4075, W10
0000	883B0A	MOV W10, NVMCON

# dsPIC30F Flash Programming Specification

**TABLE 11-5: SERIAL INSTRUCTION EXECUTION FOR ERASING PROGRAM MEMORY  
(EITHER IN LOW-VOLTAGE OR NORMAL-VOLTAGE SYSTEMS) (CONTINUED)**

Command (Binary)	Data (Hexadecimal)	Description
<b>Step 18:</b> Unlock the NVMCON to erase 1 row of data memory.		
0000	200558	MOV #0x55, W8
0000	883B38	MOV W8, NVMKEY
0000	200AA9	MOV #0xAA, W9
0000	883B39	MOV W9, NVMKEY
<b>Step 19:</b> Initiate the erase cycle.		
0000	A8E761	BSET NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
—	—	Externally time 'P13a' ms (see <a href="#">Section 13.0 “AC/DC Characteristics and Timing Requirements”</a> )
0000	000000	NOP
0000	000000	NOP
0000	A9E761	BCLR NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
<b>Step 20:</b> Update the row address stored in NVMADR.		
0000	430307	ADD W6, W7, W6
0000	883B16	MOV W6, NVMADR
<b>Step 21:</b> Reset device internal PC.		
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 22:</b> Repeat Steps 17-21 until all rows of data memory are erased.		

# dsPIC30F Flash Programming Specification

**TABLE 11-7: SERIAL INSTRUCTION EXECUTION FOR WRITING CONFIGURATION REGISTERS (CONTINUED)**

Command (Binary)	Data (Hexadecimal)	Description
<b>Step 6:</b> Write the Configuration register data to the write latch and increment the write pointer.		
0000	BB1B96	TBLWTL W6, [W7++]
0000	000000	NOP
0000	000000	NOP
<b>Step 7:</b> Unlock the NVMCON for programming.		
0000	200558	MOV #0x55, W8
0000	883B38	MOV W8, NVMKEY
0000	200AA9	MOV #0xAA, W9
0000	883B39	MOV W9, NVMKEY
<b>Step 8:</b> Initiate the write cycle.		
0000	A8E761	BSET NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
—	—	Externally time 'P12a' ms (see <a href="#">Section 13.0 “AC/DC Characteristics and Timing Requirements”</a> )
0000	000000	NOP
0000	000000	NOP
0000	A9E761	BCLR NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
<b>Step 9:</b> Reset device internal PC.		
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 10:</b> Repeat steps 3-9 until all 7 Configuration registers are cleared.		



# dsPIC30F Flash Programming Specification

**TABLE 11-10: SERIAL INSTRUCTION EXECUTION FOR READING CODE MEMORY (CONTINUED)**

Command (Binary)	Data (Hexadecimal)	Description
<b>Step 4: Output W0:W5 using the VISI register and REGOUT command.</b>		
0000	883C20	MOV W0, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
0000	883C21	MOV W1, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
0000	883C22	MOV W2, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
0000	883C23	MOV W3, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
0000	883C24	MOV W4, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
0000	883C25	MOV W5, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
<b>Step 5: Reset the device internal PC.</b>		
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 6: Repeat steps 3-5 until all desired code memory is read.</b>		

# dsPIC30F Flash Programming Specification

## 11.12 Reading Data Memory

The procedure for reading data memory is similar to that of reading code memory, except that 16-bit data words are read instead of 24-bit words. Since less data is read in each operation, only working registers W0:W3 are used as temporary holding registers for the data to be read.

Table 11-12 shows the ICSP programming details for reading data memory. Note that the TBLPAG register is hard-coded to 0x7F (the upper byte address of all locations of data memory).

**TABLE 11-12: SERIAL INSTRUCTION EXECUTION FOR READING DATA MEMORY**

Command (Binary)	Data (Hexadecimal)	Description
<b>Step 1: Exit the Reset vector.</b>		
0000	040100	GOTO 0x100
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 2: Initialize TBLPAG and the read pointer (W6) for TBLRD instruction.</b>		
0000	2007F0	MOV #0x7F, W0
0000	880190	MOV W0, TBLPAG
0000	2xxxx6	MOV #<SourceAddress15:0>, W6
<b>Step 3: Initialize the write pointer (W7) and store the next four locations of code memory to W0:W5.</b>		
0000	EB0380	CLR W7
0000	000000	NOP
0000	BA1BB6	TBLRDL [W6++], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BA1BB6	TBLRDL [W6++], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BA1BB6	TBLRDL [W6++], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BA1BB6	TBLRDL [W6++], [W7++]
0000	000000	NOP
0000	000000	NOP
<b>Step 4: Output W0:W5 using the VISI register and REGOUT command.</b>		
0000	883C20	MOV W0, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
0000	883C21	MOV W1, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
0000	883C22	MOV W2, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
0000	883C23	MOV W3, VISI
0000	000000	NOP
0001	<VISI>	Clock out contents of VISI register
0000	000000	NOP
<b>Step 5: Reset device internal PC.</b>		
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 6: Repeat steps 3-5 until all desired data memory is read.</b>		

# dsPIC30F Flash Programming Specification

## 12.0 PROGRAMMING THE PROGRAMMING EXECUTIVE TO MEMORY

Storing the programming executive to executive memory is similar to normal programming of code memory. The executive memory must first be erased, and then the programming executive must be programmed 32 words at a time. This control flow is summarized in [Table 12-1](#).

### 12.1 Overview

If it is determined that the programming executive does not reside in executive memory (as described in [Section 4.0 “Confirming the Contents of Executive Memory”](#)), it must be programmed into executive memory using ICSP and the techniques described in [Section 11.0 “ICSP™ Mode”](#).

**TABLE 12-1: PROGRAMMING THE PROGRAMMING EXECUTIVE**

Command (Binary)	Data (Hexadecimal)	Description
<b>Step 1: Exit the Reset vector and erase executive memory.</b>		
0000	040100	GOTO 0x100
0000	040100	GOTO 0x100
0000	000000	NOP
<b>Step 2: Initialize the NVMCON to erase executive memory.</b>		
0000	24072A	MOV #0x4072, W10
0000	883B0A	MOV W10, NVMCON
<b>Step 3: Unlock the NVMCON for programming.</b>		
0000	200558	MOV #0x55, W8
0000	883B38	MOV W8, NVMKEY
0000	200AA9	MOV #0xAA, W9
0000	883B39	MOV W9, NVMKEY
<b>Step 4: Initiate the erase cycle.</b>		
0000	A8E761	BSET NVMCON, #15
0000	000000	NOP
0000	000000	NOP
—	—	Externally time 'P13a' ms (see <a href="#">Section 13.0 “AC/DC Characteristics and Timing Requirements”</a> )
0000	000000	NOP
0000	000000	NOP
0000	A9E761	BCLR NVMCON, #15
0000	000000	NOP
0000	000000	NOP
<b>Step 5: Initialize the TBLPAG and the write pointer (W7).</b>		
0000	200800	MOV #0x80, W0
0000	880190	MOV W0, TBLPAG
0000	EB0380	CLR W7
0000	000000	NOP
0000	000000	NOP
<b>Step 6: Initialize the NVMCON to program 32 instruction words.</b>		
0000	24001A	MOV #0x4001, W10
0000	883B0A	MOV W10, NVMCON
<b>Step 7: Load W0:W5 with the next 4 words of packed programming executive code and initialize W6 for programming. Programming starts from the base of executive memory (0x800000) using W6 as a read pointer and W7 as a write pointer.</b>		
0000	2<LSW0>0	MOV #<LSW0>, W0
0000	2<MSB1:MSB0>1	MOV #<MSB1:MSB0>, W1
0000	2<LSW1>2	MOV #<LSW1>, W2
0000	2<LSW2>3	MOV #<LSW2>, W3
0000	2<MSB3:MSB2>4	MOV #<MSB3:MSB2>, W4
0000	2<LSW3>5	MOV #<LSW3>, W5

# dsPIC30F Flash Programming Specification

## APPENDIX A: DEVICE-SPECIFIC INFORMATION

### A.1 Checksum Computation

The checksum computation is described in [Section 6.8 “Checksum Computation”](#). [Table A-1](#) shows how this 16-bit computation can be made for each dsPIC30F device. Computations for read code protection are shown both enabled and disabled. The checksum values assume that the Configuration registers are also erased. However, when code protection is enabled, the value of the FGS register is assumed to be 0x5.

### A.2 dsPIC30F5011 and dsPIC30F5013

#### A.2.1 ICSP PROGRAMMING

The dsPIC30F5011 and dsPIC30F5013 processors require that the FBS and FSS registers be programmed with 0x0000 before the device is chip erased. The steps to perform this action are shown in [Table 11-4](#).

#### A.2.2 ENHANCED ICSP PROGRAMMING

The dsPIC30F5011 and dsPIC30F5013 processors require that the FBS and FSS registers be programmed with 0x0000 using the `PROGC` command before the `ERASEB` command is used to erase the chip.

**TABLE A-1: CHECKSUM COMPUTATION**

Device	Read Code Protection	Checksum Computation	Erased Value	Value with 0xAAAAAA at 0x0 and Last Code Address
dsPIC30F2010	Disabled	CFGB+SUM(0:001FFF)	0xD406	0xD208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F2011	Disabled	CFGB+SUM(0:001FFF)	0xD406	0xD208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F2012	Disabled	CFGB+SUM(0:001FFF)	0xD406	0xD208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F3010	Disabled	CFGB+SUM(0:003FFF)	0xA406	0xA208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F3011	Disabled	CFGB+SUM(0:003FFF)	0xA406	0xA208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F3012	Disabled	CFGB+SUM(0:003FFF)	0xA406	0xA208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F3013	Disabled	CFGB+SUM(0:003FFF)	0xA406	0xA208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F3014	Disabled	CFGB+SUM(0:003FFF)	0xA406	0xA208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F4011	Disabled	CFGB+SUM(0:007FFF)	0x4406	0x4208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F4012	Disabled	CFGB+SUM(0:007FFF)	0x4406	0x4208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F4013	Disabled	CFGB+SUM(0:007FFF)	0x4406	0x4208
	Enabled	CFGB	0x0404	0x0404
dsPIC30F5011	Disabled	CFGB+SUM(0:00AFFF)	0xFC06	0xFA08
	Enabled	CFGB	0x0404	0x0404
dsPIC30F5013	Disabled	CFGB+SUM(0:00AFFF)	0xFC06	0xFA08
	Enabled	CFGB	0x0404	0x0404
dsPIC30F5015	Disabled	CFGB+SUM(0:00AFFF)	0xFC06	0xFA08
	Enabled	CFGB	0x0404	0x0404

#### Item Description:

**SUM(a:b)** = Byte sum of locations a to b inclusive (all 3 bytes of code memory)

**CFGB** = **Configuration Block (masked)** = Byte sum of ((FOSC&0xC10F) + (FWDTC&0x803F) + (FBORPOR&0x87B3) + (FBS&0x310F) + (FSS&0x330F) + (FGS&0x0007) + (FICD&0xC003))

## 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://support.microchip.com>  
Web Address:  
[www.microchip.com](http://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

**Kokomo**  
Kokomo, IN  
Tel: 765-864-8360  
Fax: 765-864-8387

**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: 86-10-8528-2100  
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 - Hong Kong SAR**  
Tel: 852-2401-1200  
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: 86-755-8203-2660  
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 - Yokohama**  
Tel: 81-45-471- 6166  
Fax: 81-45-471-6122

**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: 886-3-6578-300  
Fax: 886-3-6578-370

**Taiwan - Kaohsiung**  
Tel: 886-7-213-7830  
Fax: 886-7-330-9305

**Taiwan - Taipei**  
Tel: 886-2-2500-6610  
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 - Druenen**  
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