



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	30 MIPS
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	12
Program Memory Size	24KB (8K x 24)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 8x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic30f3012t-30i-ml

dsPIC30F Flash Programming Specification

5.5 Code Memory Programming

5.5.1 OVERVIEW

The Flash code memory array consists of 512 rows of thirty-two, 24-bit instructions. Each panel stores 16K instruction words, and each dsPIC30F device has either 1, 2 or 3 memory panels (see [Table 5-2](#)).

TABLE 5-2: DEVICE CODE MEMORY SIZE

Device	Code Size (24-bit Words)	Number of Rows	Number of Panels
dsPIC30F2010	4K	128	1
dsPIC30F2011	4K	128	1
dsPIC30F2012	4K	128	1
dsPIC30F3010	8K	256	1
dsPIC30F3011	8K	256	1
dsPIC30F3012	8K	256	1
dsPIC30F3013	8K	256	1
dsPIC30F3014	8K	256	1
dsPIC30F4011	16K	512	1
dsPIC30F4012	16K	512	1
dsPIC30F4013	16K	512	1
dsPIC30F5011	22K	704	2
dsPIC30F5013	22K	704	2
dsPIC30F5015	22K	704	2
dsPIC30F5016	22K	704	2
dsPIC30F6010	48K	1536	3
dsPIC30F6010A	48K	1536	3
dsPIC30F6011	44K	1408	3
dsPIC30F6011A	44K	1408	3
dsPIC30F6012	48K	1536	3
dsPIC30F6012A	48K	1536	3
dsPIC30F6013	44K	1408	3
dsPIC30F6013A	44K	1408	3
dsPIC30F6014	48K	1536	3
dsPIC30F6014A	48K	1536	3
dsPIC30F6015	48K	1536	3

5.5.2 PROGRAMMING METHODOLOGY

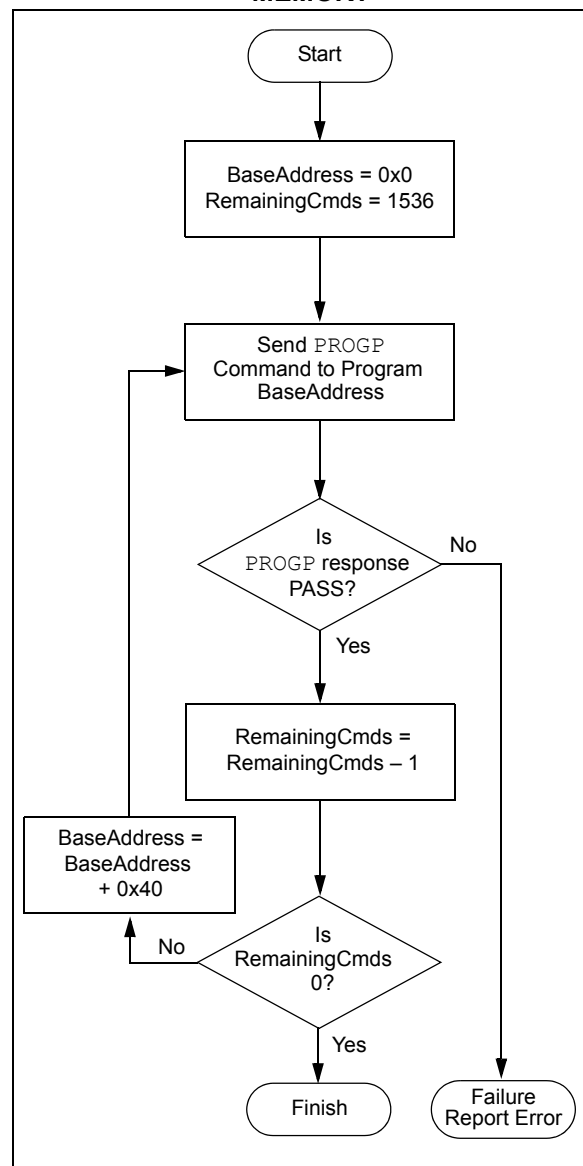
Code memory is programmed with the `PROGP` command. `PROGP` programs one row of code memory to the memory address specified in the command. The number of `PROGP` commands required to program a device depends on the number of rows that must be programmed in the device.

A flowchart for programming of code memory is illustrated in [Figure 5-3](#). In this example, all 48K instruction words of a dsPIC30F6014A device are programmed. First, the number of commands to send (called 'RemainingCmds' in the flowchart) is set to 1536 and the destination address (called 'BaseAddress') is set to '0'.

Next, one row in the device is programmed with a `PROGP` command. Each `PROGP` command contains data for one row of code memory of the dsPIC30F6014A. After the first command is processed successfully, 'RemainingCmds' is decremented by 1 and compared to 0. Since there are more `PROGP` commands to send, 'BaseAddress' is incremented by 0x40 to point to the next row of memory.

On the second `PROGP` command, the second row of each memory panel is programmed. This process is repeated until the entire device is programmed. No special handling must be performed when a panel boundary is crossed.

FIGURE 5-3: FLOWCHART FOR PROGRAMMING dsPIC30F6014A CODE MEMORY



dsPIC30F Flash Programming Specification

5.6.3 PROGRAMMING VERIFICATION

Once the data EEPROM is programmed, the contents of memory can be verified to ensure that the programming was successful. Verification requires the data EEPROM to be read back and compared against the copy held in the programmer's buffer. The `READD` command reads back the programmed data EEPROM.

Alternatively, the programmer can perform the verification once the entire device is programmed using a checksum computation, as described in [Section 6.8 "Checksum Computation"](#).

Note: `TBLRDL` instructions executed within a `REPEAT` loop must not be used to read from Data EEPROM. Instead, it is recommended to use PSV access.

5.7 Configuration Bits Programming

5.7.1 OVERVIEW

The dsPIC30F has Configuration bits stored in seven 16-bit registers. These bits can be set or cleared to select various device configurations. There are two types of Configuration bits: system-operation bits and code-protect bits. The system-operation bits determine the power-on settings for system-level components such as the oscillator and Watchdog Timer. The code-protect bits prevent program memory from being read and written.

The FOSC Configuration register has three different register descriptions, based on the device. The FOSC Configuration register description for the dsPIC30F2010 and dsPIC30F6010/6011/6012/6013/6014 devices are shown in [Table 5-4](#).

Note: If user software performs an erase operation on the configuration fuse, it must be followed by a write operation to this fuse with the desired value, even if the desired value is the same as the state of the erased fuse.

The FOSC Configuration register description for the dsPIC30F4011/4012 and dsPIC30F5011/5013 devices is shown in [Table 5-5](#).

The FOSC Configuration register description for all remaining devices (dsPIC30F2011/2012, dsPIC30F3010/3011/3012/3013, dsPIC30F3014/4013, dsPIC30F5015 and dsPIC30F6011A/6012A/6013A/6014A) is shown in [Table 5-6](#). Always use the correct register descriptions for your target processor.

The `FWDT`, `FBORPOR`, `FBS`, `FSS`, `FGS` and `FICD` Configuration registers are not device-dependent. The register descriptions for these Configuration registers are shown in [Table 5-7](#).

The Device Configuration register maps are shown in [Table 5-8](#) through [Table 5-11](#).

TABLE 5-4: FOSC CONFIGURATION BITS DESCRIPTION FOR dsPIC30F2010 AND dsPIC30F6010/6011/6012/6013/6014

Bit Field	Register	Description
FCKSM<1:0>	FOSC	Clock Switching Mode 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	Oscillator Source Selection on POR 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	Primary Oscillator Mode 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 = Reserved (do not use) 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) 001x = HS – HS Crystal Oscillator mode (10 MHz-25 MHz crystal) 000x = 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	Alternate Oscillator Mode (when FOS<2:0> = 011b) 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)

dsPIC30F Flash Programming Specification

TABLE 5-7: CONFIGURATION BITS DESCRIPTION (CONTINUED)

Bit Field	Register	Description
EBS	FBS	Boot Segment Data EEPROM Code Protection (only present in dsPIC30F5011/5013/6010A/6011A/6012A/6013A/6014A/6015) 1 = No Data EEPROM is reserved for Boot Segment 0 = 128 bytes of Data EEPROM are reserved for Boot Segment in dsPIC30F5011/5013, and 256 bytes in dsPIC30F6010A/6011A/6012A/6013A/6014A/6015
BSS<2:0>	FBS	Boot Segment Program Memory Code Protection (only present in dsPIC30F5011/5013/6010A/6011A/6012A/6013A/6014A/6015) 111 = No Boot Segment 110 = Standard security; Small-sized Boot Program Flash [Boot Segment starts after BS and ends at 0x0003FF] 101 = Standard security; Medium-sized Boot Program Flash [Boot Segment starts after BS and ends at 0x000FFF] 100 = Standard security; Large-sized Boot Program Flash [Boot Segment starts after BS and ends at 0x001FFF] 011 = No Boot Segment 010 = High security; Small-sized Boot Program Flash [Boot Segment starts after BS and ends at 0x0003FF] 001 = High security; Medium-sized Boot Program Flash [Boot Segment starts after BS and ends at 0x000FFF] 000 = High security; Large-sized Boot Program Flash [Boot Segment starts after BS and ends at 0x001FFF]
BWRP	FBS	Boot Segment Program Memory Write Protection (only present in dsPIC30F5011/5013/6010A/6011A/6012A/6013A/6014A/6015) 1 = Boot Segment program memory is not write-protected 0 = Boot Segment program memory is write-protected
RSS<1:0>	FSS	Secure Segment Data RAM Code Protection (only present in dsPIC30F5011/5013/6010A/6011A/6012A/6013A/6014A/6015) 11 = No Data RAM is reserved for Secure Segment 10 = Small-sized Secure RAM [(256 – N) bytes of RAM are reserved for Secure Segment] 01 = Medium-sized Secure RAM [(768 – N) bytes of RAM are reserved for Secure Segment in dsPIC30F5011/5013, and (2048 – N) bytes in dsPIC30F6010A/6011A/6012A/6013A/6014A/6015] 00 = Large-sized Secure RAM [(1024 – N) bytes of RAM are reserved for Secure Segment in dsPIC30F5011/5013, and (4096 – N) bytes in dsPIC30F6010A/6011A/6012A/6013A/6014A/6015] where N = Number of bytes of RAM reserved for Boot Sector.
ESS<1:0>	FSS	Secure Segment Data EEPROM Code Protection (only present in dsPIC30F5011/5013/6010A/6011A/6012A/6013A/6014A/6015) 11 = No Data EEPROM is reserved for Secure Segment 10 = Small-sized Secure Data EEPROM [(128 – N) bytes of Data EEPROM are reserved for Secure Segment in dsPIC30F5011/5013, and (256 – N) bytes in dsPIC30F6010A/6011A/6012A/6013A/6014A/6015] 01 = Medium-sized Secure Data EEPROM [(256 – N) bytes of Data EEPROM are reserved for Secure Segment in dsPIC30F5011/5013, and (512 – N) bytes in dsPIC30F6010A/6011A/6012A/6013A/6014A/6015] 00 = Large-sized Secure Data EEPROM [(512 – N) bytes of Data EEPROM are reserved for Secure Segment in dsPIC30F5011/5013, (1024 – N) bytes in dsPIC30F6011A/6013A, and (2048 – N) bytes in dsPIC30F6010A/6012A/6014A/6015] where N = Number of bytes of Data EEPROM reserved for Boot Sector.

TABLE 5-8: dsPIC30F CONFIGURATION REGISTERS (FOR dsPIC30F2010, dsPIC30F4011/4012 AND dsPIC30F6010/ 6011/6012/6013/ 6014)

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<1:0>		—	—	—	—	FPR<3:0>			
0xF80002	FWDT	FWDTEN	—	—	—	—	—	—	—	—	—	FWPSA<1:0>		FWPSB<3:0>			
0xF80004	FBORPOR	MCLREN	—	—	—	—	PWMPIN ⁽¹⁾	HPOL ⁽¹⁾	LPOL ⁽¹⁾	BOREN	—	BORV<1:0>		—	—	FPWRT<1:0>	
0xF80006	FBS	—	—	Reserved ⁽²⁾		—	—	—	Reserved ⁽²⁾	—	—	—	—	Reserved ⁽²⁾			
0xF80008	FSS	—	—	Reserved ⁽²⁾		—	—	Reserved ⁽²⁾		—	—	—	—	Reserved ⁽²⁾			
0xF8000A	FGS	—	—	—	—	—	—	—	—	—	—	—	—	—	Reserved ⁽²⁾	GCP	GWRP
0xF8000C	FICD	BKBUG	COE	—	—	—	—	—	—	—	—	—	—	—	—	ICS<1:0>	

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

Note 2: Reserved bits read as '1' and must be programmed as '1'.

TABLE 5-9: dsPIC30F CONFIGURATION REGISTERS (FOR dsPIC30F5011/5013)

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<1:0>		—	—	—	—	FPR<3:0>			
0xF80002	FWDT	FWDTEN	—	—	—	—	—	—	—	—	—	FWPSA<1:0>		FWPSB<3:0>			
0xF80004	FBORPOR	MCLREN	—	—	—	—	Reserved ⁽¹⁾			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: Reserved bits read as '1' and must be programmed as '1'.

dsPIC30F Flash Programming Specification

5.7.2 PROGRAMMING METHODOLOGY

System operation Configuration bits are inherently different than all other memory cells. Unlike code memory, data EEPROM and code-protect Configuration bits, the system operation bits cannot be erased. If the chip is erased with the `ERASEB` command, the system-operation bits retain their previous value. Consequently, you should make no assumption about the value of the system operation bits. They should always be programmed to their desired setting.

Configuration bits are programmed as a single word at a time using the `PROGC` command. The `PROGC` command specifies the configuration data and Configuration register address. When Configuration bits are programmed, any unimplemented bits must be programmed with a '0', and any reserved bits must be programmed with a '1'.

Four `PROGC` commands are required to program all the Configuration bits. Figure 5-5 illustrates the flowchart of Configuration bit programming.

Note: If the General Code Segment Code Protect (GCP) bit is programmed to '0', code memory is code-protected and cannot be read. Code memory must be verified before enabling read protection. See Section 5.7.4 "Code-Protect Configuration Bits" for more information about code-protect Configuration bits.

5.7.3 PROGRAMMING VERIFICATION

Once the Configuration bits are programmed, the contents of memory should be verified to ensure that the programming was successful. Verification requires the Configuration bits to be read back and compared against the copy held in the programmer's buffer. The `READD` command reads back the programmed Configuration bits and verifies whether the programming was successful.

Any unimplemented Configuration bits are read-only and read as '0'.

5.7.4 CODE-PROTECT CONFIGURATION BITS

The FBS, FSS and FGS Configuration registers are special Configuration registers that control the size and level of code protection for the Boot Segment, Secure Segment and General Segment, respectively. For each segment, two main forms of code protection are provided. One form prevents code memory from being written (write protection), while the other prevents code memory from being read (read protection).

The BWRP, SWRP and GWRP bits control write protection; and BSS<2:0>, SSS<2:0> and GSS<1:0> bits control read protection. The Chip Erase `ERASEB` command sets all the code protection bits to '1', which allows the device to be programmed.

When write protection is enabled, any programming operation to code memory will fail. When read protection is enabled, any read from code memory will cause a '0x0' to be read, regardless of the actual contents of code memory. Since the programming executive always verifies what it programs, attempting to program code memory with read protection enabled will also result in failure.

It is imperative that all code protection bits are '1' while the device is being programmed and verified. Only after the device is programmed and verified should any of the above bits be programmed to '0' (see Section 5.7 "Configuration Bits Programming").

In addition to code memory protection, parts of data EEPROM and/or data RAM can be configured to be accessible only by code resident in the Boot Segment and/or Secure Segment. The sizes of these "reserved" sections are user-configurable, using the EBS, RBS<1:0>, ESS<1:0> and RSS<1:0> bits.

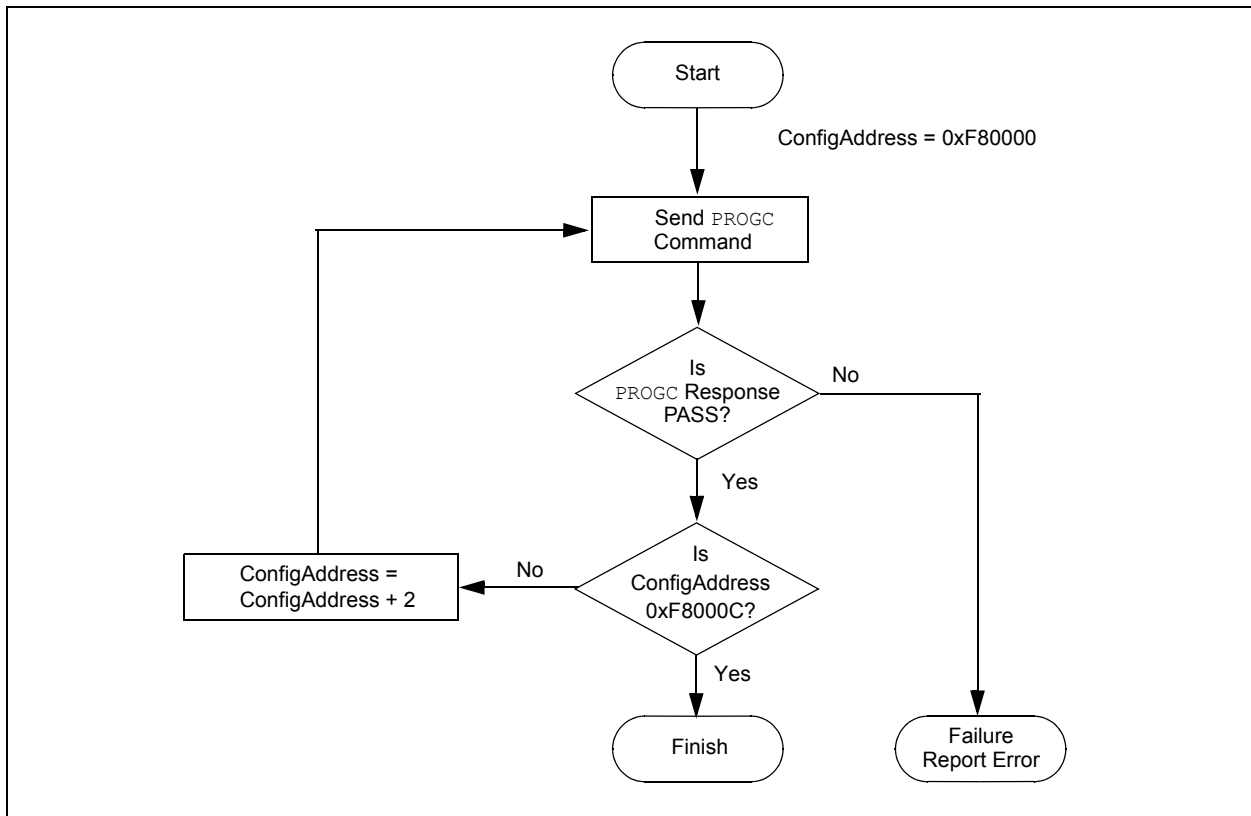
Note 1: All bits in the FBS, FSS and FGS Configuration registers can only be programmed to a value of '0'. `ERASEB` is the only way to reprogram code-protect bits from ON ('0') to OFF ('1').

2: If any of the code-protect bits in FBS, FSS, or FGS are clear, the entire device must be erased before it can be reprogrammed.

5.8 Exiting Enhanced ICSP Mode

The Enhanced ICSP mode is exited by removing power from the device or bringing MCLR to V_{IL}. When normal user mode is next entered, the program that was stored using Enhanced ICSP will execute.

FIGURE 5-5: CONFIGURATION BIT PROGRAMMING FLOW



dsPIC30F Flash Programming Specification

6.0 OTHER PROGRAMMING FEATURES

6.1 Erasing Memory

Memory is erased by using an `ERASEB`, `ERASED` or `ERASEP` command, as detailed in [Section 8.5 “Command Descriptions”](#). Code memory can be erased by row using `ERASEP`. Data EEPROM can be erased by row using `ERASED`. When memory is erased, the affected memory locations are set to ‘1’s.

`ERASEB` provides several Bulk Erase options. Performing a Chip Erase with the `ERASEB` command clears all code memory, data EEPROM and code protection registers. Alternatively, `ERASEB` can be used to selectively erase either all code memory or data EEPROM. Erase options are summarized in [Table 6-1](#).

TABLE 6-1: ERASE OPTIONS

Command	Affected Region
<code>ERASEB</code>	Entire chip ⁽¹⁾ or all code memory or all data EEPROM, or erase by segment
<code>ERASED</code>	Specified rows of data EEPROM
<code>ERASEP</code> ⁽²⁾	Specified rows of code memory

- Note 1:** The system operation Configuration registers and device ID registers are not erasable.
- 2:** `ERASEP` cannot be used to erase code-protect Configuration bits. These bits must be erased using `ERASEB`.

6.2 Modifying Memory

Instead of bulk-erasing the device before programming, it is possible that you may want to modify only a section of an already programmed device. In this situation, Chip Erase is not a realistic option.

Instead, you can erase selective rows of code memory and data EEPROM using `ERASEP` and `ERASED`, respectively. You can then reprogram the modified rows with the `PROGP` and `PROGD` command pairs. In these cases, when code memory is programmed, single-panel programming must be specified in the `PROGP` command.

For modification of Advanced Code Protection bits for a particular segment, the entire chip must first be erased with the `ERASEB` command. Alternatively, on devices that support Advanced Security, individual segments (code and/or data EEPROM) may be erased, by suitably changing the MS (Memory Select)

field in the `ERASEB` command. The code-protect Configuration bits can then be reprogrammed using the `PROGC` command.

Note: If read or write code protection is enabled for a segment, no modifications can be made to that segment until code protection is disabled. Code protection can only be disabled by performing a Chip Erase or by performing a Segment Erase operation for the required segment.

6.3 Reading Memory

The `READD` command reads the data EEPROM, Configuration bits and device ID of the device. This command only returns 16-bit data and operates on 16-bit registers. `READD` can be used to return the entire contents of data EEPROM.

The `READP` command reads the code memory of the device. This command only returns 24-bit data packed as described in [Section 8.3 “Packed Data Format”](#). `READP` can be used to read up to 32K instruction words of code memory.

Note: Reading an unimplemented memory location causes the programming executive to reset. All `READD` and `READP` commands **must** specify only valid memory locations.

6.4 Programming Executive Software Version

At times, it may be necessary to determine the version of programming executive stored in executive memory. The `QVER` command performs this function. See [Section 8.5.11 “QVER Command”](#) for more details about this command.

6.5 Data EEPROM Information in the Hexadecimal File

To allow portability of code, the programmer must read the data EEPROM information from the hexadecimal file. If data EEPROM information is not present, a simple warning message should be issued by the programmer. Similarly, when saving a hexadecimal file, all data EEPROM information must be included. An option to not include the data EEPROM information can be provided.

Microchip Technology Inc. believes that this feature is important for the benefit of the end customer.

dsPIC30F Flash Programming Specification

clocked out. The programmer can begin to clock out the response 20 μ sec after PGD is brought low, and it must provide the necessary amount of clock pulses to receive the entire response from the programming executive.

Once the entire response is clocked out, the programmer should terminate the clock on PGC until it is time to send another command to the programming executive. This protocol is illustrated in Figure 7-2.

7.3 SPI Rate

In Enhanced ICSP mode, the dsPIC30F operates from the fast internal RC oscillator, which has a nominal frequency of 7.37 MHz. This oscillator frequency yields an effective system clock frequency of 1.84 MHz. Since the SPI module operates in Slave mode, the programmer must limit the SPI clock rate to a frequency no greater than 1 MHz.

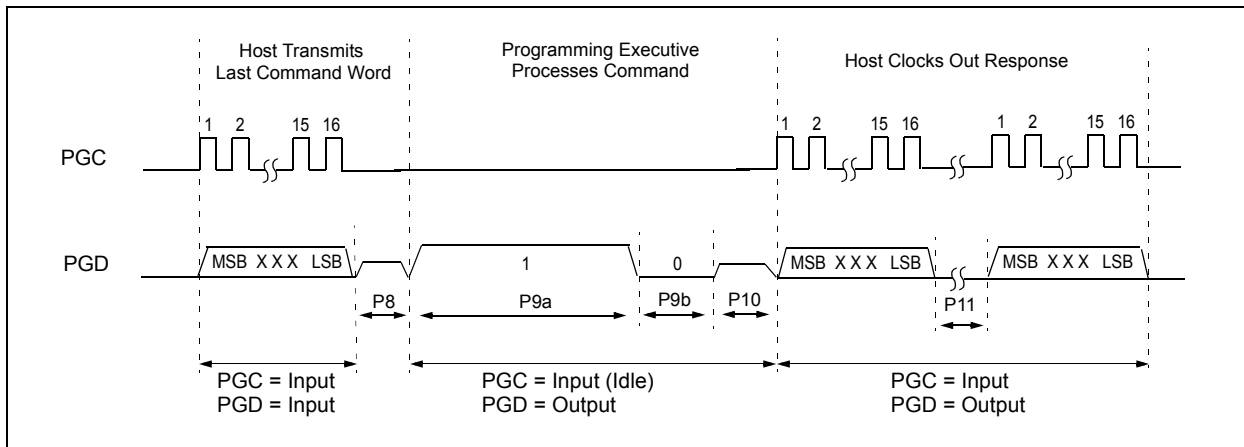
Note: If the programmer provides the SPI with a clock faster than 1 MHz, the behavior of the programming executive will be unpredictable.

7.4 Time Outs

The programming executive uses no Watchdog Timer or time out for transmitting responses to the programmer. If the programmer does not follow the flow control mechanism using PGC, as described in [Section 7.2 “Communication Interface and Protocol”](#), it is possible that the programming executive will behave unexpectedly while trying to send a response to the programmer. Since the programming executive has no time out, it is imperative that the programmer correctly follow the described communication protocol.

As a safety measure, the programmer should use the command time outs identified in [Table 8-1](#). If the command time out expires, the programmer should reset the programming executive and start programming the device again.

FIGURE 7-2: PROGRAMMING EXECUTIVE – PROGRAMMER COMMUNICATION PROTOCOL



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

10.0 DEVICE ID

The device ID region is 2 x 16 bits and can be read using the `READD` command. This region of memory is read-only and can also be read when code protection is enabled.

Table 10-1 shows the device ID for each device, Table 10-2 shows the device ID registers and Table 10-3 describes the bit field of each register.

TABLE 10-1: DEVICE IDS

Device	DEVID	Silicon Revision							
		A0	A1	A2	A3	A4	B0	B1	B2
dsPIC30F2010	0x0040	0x1000	0x1001	0x1002	0x1003	0x1004	—	—	—
dsPIC30F2011	0x0240	—	0x1001	—	—	—	—	—	—
dsPIC30F2012	0x0241	—	0x1001	—	—	—	—	—	—
dsPIC30F3010	0x01C0	0x1000	0x1001	0x1002	—	—	—	—	—
dsPIC30F3011	0x01C1	0x1000	0x1001	0x1002	—	—	—	—	—
dsPIC30F3012	0x00C1	—	—	—	—	—	0x1040	0x1041	—
dsPIC30F3013	0x00C3	—	—	—	—	—	0x1040	0x1041	—
dsPIC30F3014	0x0160	—	0x1001	0x1002	—	—	—	—	—
dsPIC30F4011	0x0101	—	0x1001	0x1002	0x1003	0x1003	—	—	—
dsPIC30F4012	0x0100	—	0x1001	0x1002	0x1003	0x1003	—	—	—
dsPIC30F4013	0x0141	—	0x1001	0x1002	—	—	—	—	—
dsPIC30F5011	0x0080	—	0x1001	0x1002	0x1003	0x1003	—	—	—
dsPIC30F5013	0x0081	—	0x1001	0x1002	0x1003	0x1003	—	—	—
dsPIC30F5015	0x0200	0x1000	—	—	—	—	—	—	—
dsPIC30F5016	0x0201	0x1000	—	—	—	—	—	—	—
dsPIC30F6010	0x0188	—	—	—	—	—	—	0x1040	0x1042
dsPIC30F6010A	0x0281	—	—	0x1002	0x1003	0x1004	—	—	—
dsPIC30F6011	0x0192	—	—	—	0x1003	—	—	0x1040	0x1042
dsPIC30F6011A	0x02C0	—	—	0x1002	—	—	0x1040	0x1041	—
dsPIC30F6012	0x0193	—	—	—	0x1003	—	—	0x1040	0x1042
dsPIC30F6012A	0x02C2	—	—	0x1002	—	—	0x1040	0x1041	—
dsPIC30F6013	0x0197	—	—	—	0x1003	—	—	0x1040	0x1042
dsPIC30F6013A	0x02C1	—	—	0x1002	—	—	0x1040	0x1041	—
dsPIC30F6014	0x0198	—	—	—	0x1003	—	—	0x1040	0x1042
dsPIC30F6014A	0x02C3	—	—	0x1002	—	—	0x1040	0x1041	—
dsPIC30F6015	0x0280	—	—	0x1002	0x1003	0x1004	—	—	—

TABLE 10-2: dsPIC30F DEVICE ID REGISTERS

Address	Name	Bit															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFF0000	DEVID	DEVID<15:0>															
0xFF0002	DEVREV	PROC<3:0>				REV<5:0>						DOT<5:0>					

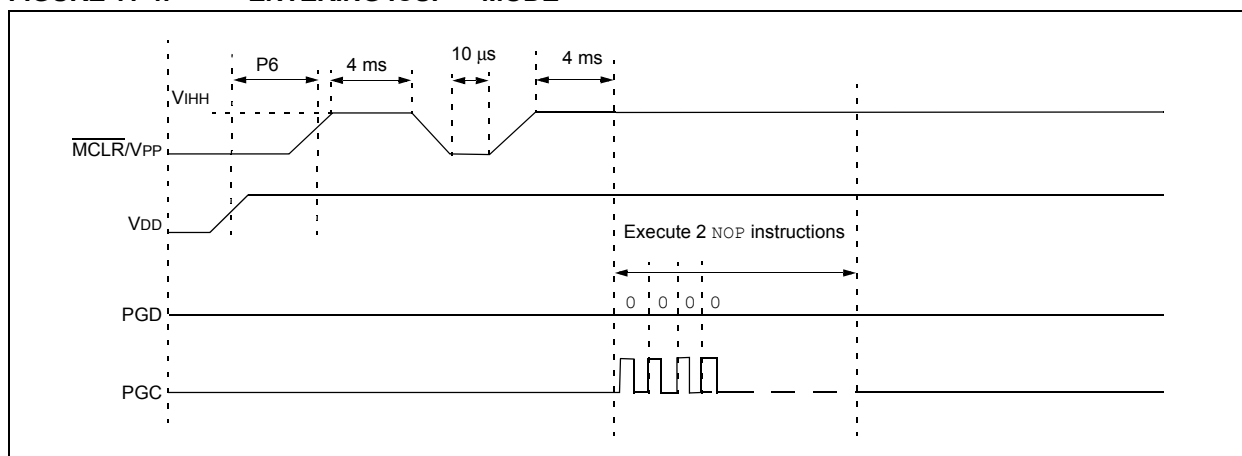
dsPIC30F Flash Programming Specification

11.3 Entering ICSP Mode

The ICSP mode is entered by holding PGC and PGD low, raising MCLR/VPP to VIH (high voltage), and then performing additional steps as illustrated in [Figure 11-4](#).

- Note 1:** The sequence that places the device into ICSP mode places all unused I/O pins to the high-impedance state.
- 2:** Once ICSP mode is entered, the PC is set to 0x0 (the Reset vector).
- 3:** Before leaving the Reset vector, execute two `GOTO` instructions, followed by a single `NOP` instruction must be executed.

FIGURE 11-4: ENTERING ICSP™ MODE



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

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

dsPIC30F Flash Programming Specification

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

Command (Binary)	Data (Hexadecimal)	Description
Step 6: 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
Step 7: 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
Step 8: Initiate the write cycle.		
0000	A8E761	BSET NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
—	—	Externally time 'P12a' ms (see Section 13.0 “AC/DC Characteristics and Timing Requirements”)
0000	000000	NOP
0000	000000	NOP
0000	A9E761	BCLR NVMCON, #WR
0000	000000	NOP
0000	000000	NOP
Step 9: Reset device internal PC.		
0000	040100	GOTO 0x100
0000	000000	NOP
Step 10: Repeat steps 3-9 until all 7 Configuration registers are cleared.		

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
Step 1: Exit the Reset vector.		
0000	040100	GOTO 0x100
0000	040100	GOTO 0x100
0000	000000	NOP
Step 2: Initialize TBLPAG and the read pointer (W6) for TBLRD instruction.		
0000	2007F0	MOV #0x7F, W0
0000	880190	MOV W0, TBLPAG
0000	2xxxx6	MOV #<SourceAddress15:0>, W6
Step 3: Initialize the write pointer (W7) and store the next four locations of code memory to W0:W5.		
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
Step 4: Output W0:W5 using the VISI register and REGOUT command.		
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
Step 5: Reset device internal PC.		
0000	040100	GOTO 0x100
0000	000000	NOP
Step 6: Repeat steps 3-5 until all desired data memory is read.		

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
Step 1: Exit the Reset vector and erase executive memory.		
0000	040100	GOTO 0x100
0000	040100	GOTO 0x100
0000	000000	NOP
Step 2: Initialize the NVMCON to erase executive memory.		
0000	24072A	MOV #0x4072, W10
0000	883B0A	MOV W10, NVMCON
Step 3: 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
Step 4: Initiate the erase cycle.		
0000	A8E761	BSET NVMCON, #15
0000	000000	NOP
0000	000000	NOP
—	—	Externally time 'P13a' ms (see Section 13.0 “AC/DC Characteristics and Timing Requirements”)
0000	000000	NOP
0000	000000	NOP
0000	A9E761	BCLR NVMCON, #15
0000	000000	NOP
0000	000000	NOP
Step 5: Initialize the TBLPAG and the write pointer (W7).		
0000	200800	MOV #0x80, W0
0000	880190	MOV W0, TBLPAG
0000	EB0380	CLR W7
0000	000000	NOP
0000	000000	NOP
Step 6: Initialize the NVMCON to program 32 instruction words.		
0000	24001A	MOV #0x4001, W10
0000	883B0A	MOV W10, NVMCON
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.		
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

12.2 Programming Verification

After the programming executive has been programmed to executive memory using ICSP, it must be verified. Verification is performed by reading out the contents of executive memory and comparing it with the image of the programming executive stored in the programmer.

Reading the contents of executive memory can be performed using the same technique described in [Section 11.10 “Reading Code Memory”](#). A procedure for reading executive memory is shown in [Table 12-2](#). Note that in Step 2, the TBLPAG register is set to 0x80 such that executive memory may be read.

TABLE 12-2: READING EXECUTIVE MEMORY

Command (Binary)	Data (Hexadecimal)	Description
Step 1: Exit the Reset vector.		
0000	040100	GOTO 0x100
0000	040100	GOTO 0x100
0000	000000	NOP
Step 2: Initialize TBLPAG and the read pointer (W6) for TBLRD instruction.		
0000	200800	MOV #0x80, W0
0000	880190	MOV W0, TBLPAG
0000	EB0300	CLR W6
Step 3: Initialize the write pointer (W7), and store the next four locations of executive memory to W0:W5.		
0000	EB0380	CLR W7
0000	000000	NOP
0000	BA1B96	TBLRDL [W6], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BADBB6	TBLRDH.B [W6++], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BADBD6	TBLRDH.B [W6++], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BA1BB6	TBLRDL [W6++], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BA1B96	TBLRDL [W6], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BADBB6	TBLRDH.B [W6++], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BADBD6	TBLRDH.B [W6++], [W7++]
0000	000000	NOP
0000	000000	NOP
0000	BA1BB6	TBLRDL [W6++], [W7]
0000	000000	NOP
0000	000000	NOP

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC³² logo, rfPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.


FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniscent Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICTail, REAL ICE, rLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2010, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

ISBN: 978-1-60932-636-4

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

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

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