PIC12F60X/12F61X/16F61X

Memory Programming Specification

1.0 DEVICE OVERVIEW

This document includes the programming specifications for the following devices:

- PIC12F615
- PIC12HV615
- PIC12F609
- PIC12HV609
- PIC16F616
- PIC16HV616
- PIC16F610
- PIC16HV610

Note 1: All references to the PIC12F615 parts refer to the PIC12HV615 parts as well (unless otherwise specified).

2: All references to the PIC16F616 parts refer to the PIC16HV616 as well (unless otherwise specified).

3: All references to the PIC12F609 parts refer to the PIC12HV609 as well (unless otherwise specified).

4: All references to the PIC16F610 parts refer to the PIC16HV610 as well (unless otherwise specified).

5: Any references in this programming specification to PORTA and RAn refer to GPIO and GPn, respectively.

2.0 PROGRAMMING THE PIC12F60X/12F61X/16F61X DEVICES

The PIC12F60X/12F61X/16F61X devices are programmed using a serial method. The Serial mode will allow the PIC12F60X/12F61X/16F61X devices to be programmed while in the user's system. This programming specification applies to the PIC12F60X/12F61X/16F61X devices in all packages.

2.1 Hardware Requirements

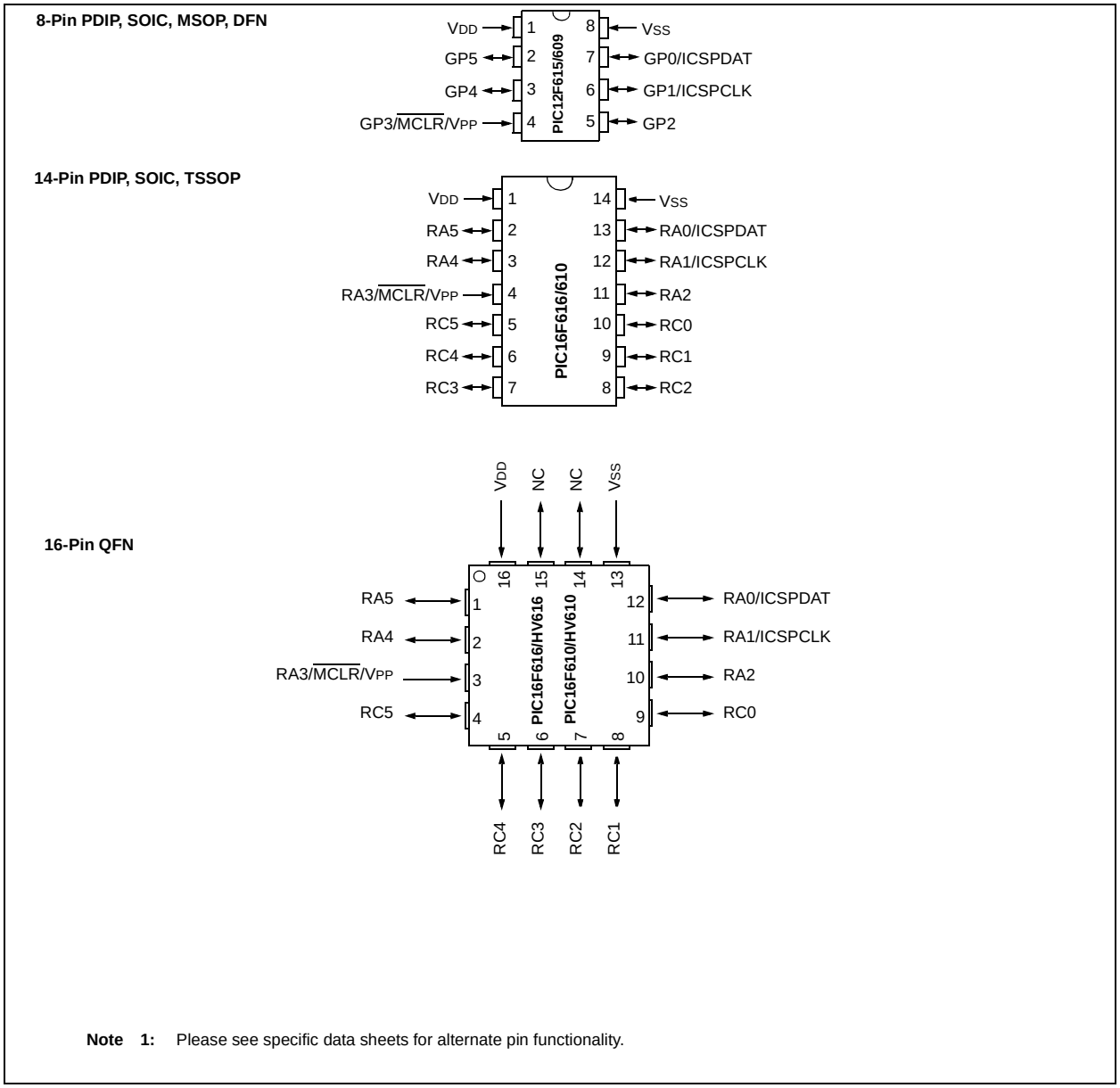
PIC12F60X/12F61X/16F61X devices require one power supply for V_{DD} (5.0V) and one for V_{PP} (12.0V).

The PIC12HV60X/12HV61X/16HV61X devices require one power supply for V_{DD} (4.5V) and one for V_{PP} (12V). V_{DD} is lower for the 'HV' parts to avoid possible contention between the shunt regulator and an unrestricted supply current.

2.2 Program/Verify Mode

The Program/Verify mode for the PIC12F60X/12F61X/16F61X devices allow programming of user program memory, user ID locations, Calibration Word and the Configuration Word.

FIGURE 2-1: 8-PIN, 14-PIN, AND 16-PIN PROGRAMMING PINS DIAGRAM FOR PIC12F60X/12F61X/16F61X(1)



Note 1: Please see specific data sheets for alternate pin functionality.

TABLE 2-1: PIN DESCRIPTIONS IN PROGRAM/VERIFY MODE: PIC12F60X/12F61X/16F61X

Pin Name	During Programming		
	Function	Pin Type	Pin Description
GP1/RA1	ICSPCLK	I	Clock input – Schmitt Trigger input
GP0/RA0	ICSPDAT	I/O	Data input/output – Schmitt Trigger input
$\overline{MCLR}$	Program/Verify mode	P(1)	Program Mode Select
VDD	VDD	P	Power Supply
VSS	VSS	P	Ground

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

Note 1: In the PIC12F60X/12F61X/16F61X, the programming high voltage is internally generated. To activate the Program/Verify mode, voltage of VIHH and a current of IIHH (see Table 7-1) needs to be applied to $\overline{MCLR}$ input.

3.0 MEMORY DESCRIPTION

3.1 Program Memory Map

The user memory space extends from 0x0000 to 0x1FFF. In Program/Verify mode, the program memory space extends from 0x0000 to 0x3FFF, with the first half (0x0000-0x1FFF) being user program memory and the second half (0x2000-0x3FFF) being configuration memory. The PC will increment from 0x0000 to 0x1FFF and wrap to 0x000. If the PC is between 0x2000 to 0x3FFF it will wrap around to 0x2000 (not to 0x0000). Once in configuration memory, the highest bit of the PC stays a '1', thus always pointing to the configuration memory. The only way to point to user program memory is to reset the part and re-enter Program/Verify mode as described in **Section 4.0 "Program/Verify Mode"**.

For the PIC12F60X/12F61X/16F61X devices, the configuration memory space, 0x2000 to 0x2008, are physically implemented. However, only locations 0x2000 to 0x2003, 0x2007 and 0x2008 are available. Other locations are reserved.

3.2 User ID Locations

A user may store identification information (user ID) in four designated locations. The user ID locations are mapped in 0x2000 to 0x2003. It is recommended that the user use only the seven Least Significant bits (LSb) of each user ID location. The user ID locations read out normally, even after code protection is enabled. It is recommended that ID locations are written as 'xx xxxx xbbb bbbb' where 'bbb bbbb' is user ID information.

The 14 bits may be programmed, but only the 7 LSb's are displayed by MPLAB® IDE. The xxxx's are "don't care" bits and are not read by MPLAB® IDE.

3.3 Calibration Word

For the PIC12F60X/12F61X/16F61X devices, the 4/8 MHz Internal Oscillator (INTOSC) module is factory calibrated. This value is stored in the Calibration Word (0x2008). See the applicable device data sheet for more information.

The Calibration Word does not necessarily participate in the erase operation unless a specific procedure is executed. Therefore, the device can be erased without affecting the Calibration Word. This simplifies the erase procedure, for these values do not need to be read and restored after the device is erased.

FIGURE 3-1: PIC12F615/HV615, PIC12F609/HV609, PIC16F610/HV610 PROGRAM MEMORY MAPPING

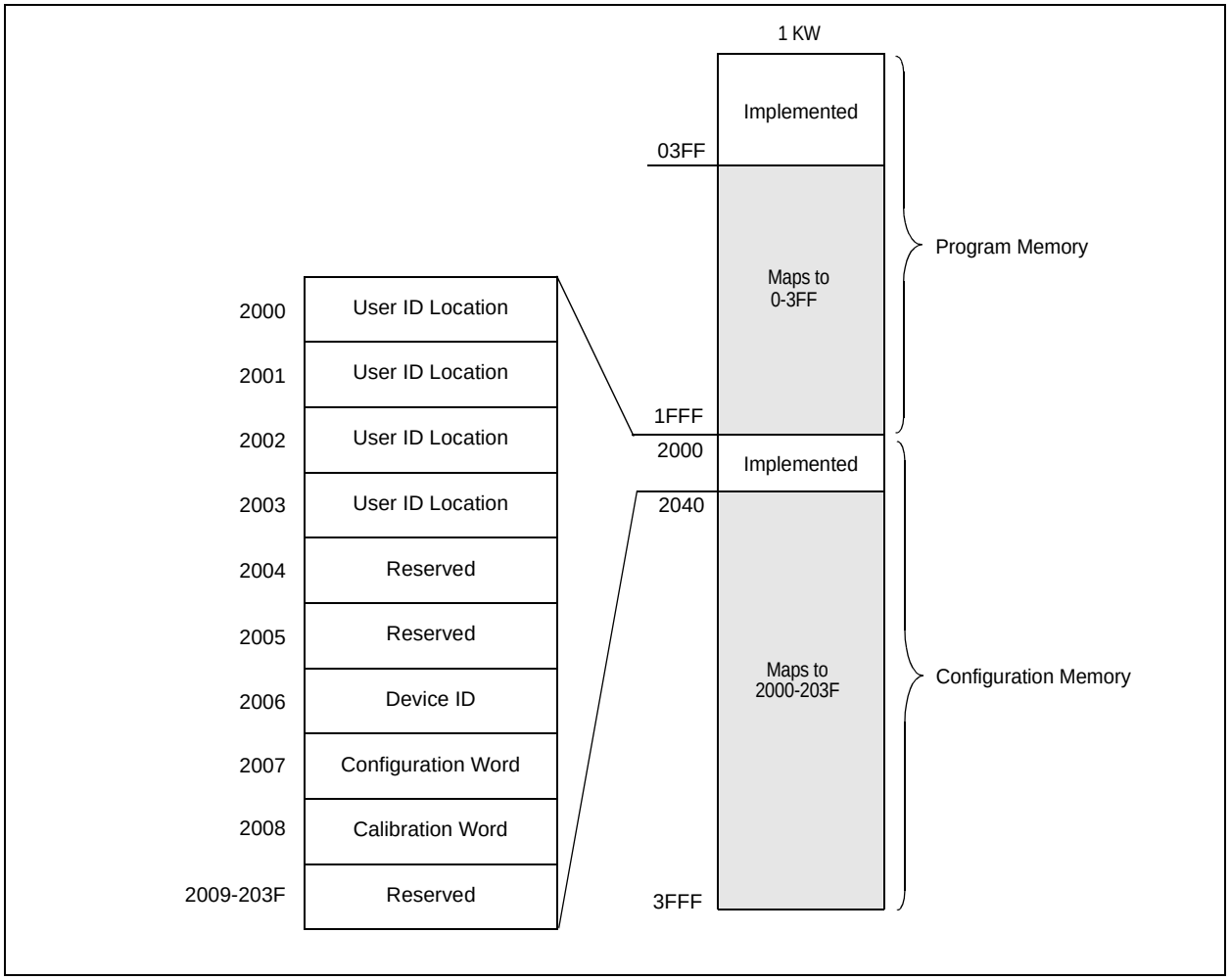
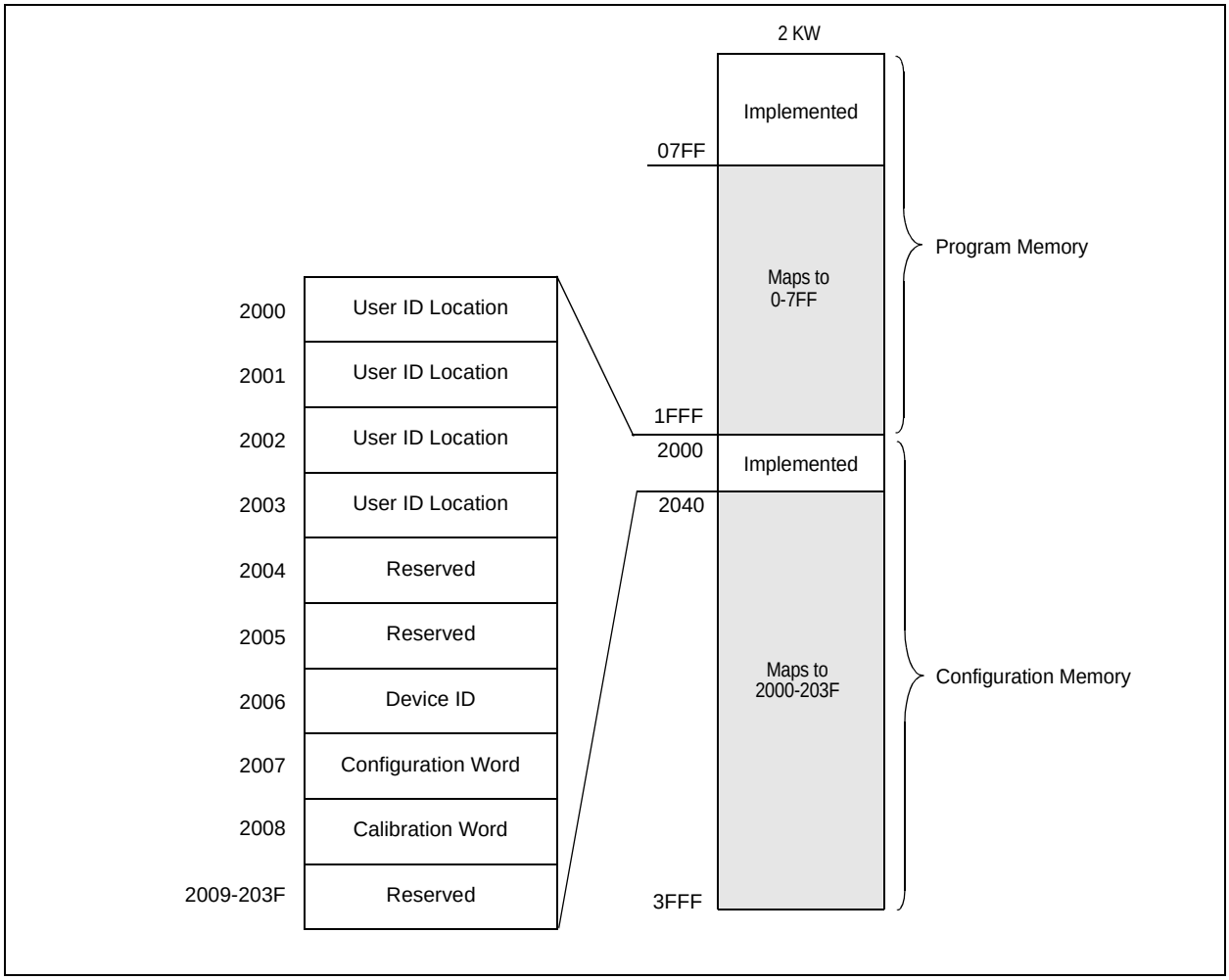


FIGURE 3-2: PIC16F616/HV616 PROGRAM MEMORY MAPPING



4.0 PROGRAM/VERIFY MODE

Two methods are available to enter Program/Verify mode. The "VPP-first" is entered by holding ICSPDAT and ICSPCLK low while raising MCLR pin from VIL to VIHH (high voltage), then applying VDD and data. This method can be used for any Configuration Word selection and **must** be used if the INTOSC and internal MCLR options are selected (FOSC<2:0> = 100 or 101 and MCLRE = 0). The VPP-first entry prevents the device from executing code prior to entering Program/Verify mode. See the timing diagram in Figure 4-1.

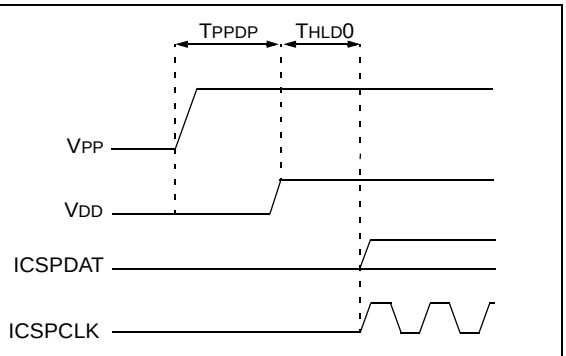
The second entry method, "VDD-first", is entered by applying VDD, holding ICSPDAT and ICSPCLK low, then raising MCLR pin from VIL to VIHH (high voltage), followed by data. This method can be used for any Configuration Word selection **except** when INTOSC and internal MCLR options are selected (FOSC<2:0> = 100 or 101 and MCLRE = 0). This technique is useful when programming the device when VDD is already applied, for it is not necessary to disconnect VDD to enter Program/Verify mode. See the timing diagram in Figure 4-2.

Once in Program/Verify mode, the program memory and configuration memory can be accessed and programmed in serial fashion. ICSPDAT and ICSPCLK are Schmitt Trigger inputs in this mode. RA4 is tri-state regardless of fuse setting.

The sequence that enters the device into the Program/Verify mode places all other logic into the Reset state (the MCLR pin was initially at VIL). Therefore, all I/O's are in the Reset state (high-impedance inputs) and the Program Counter (PC) is cleared.

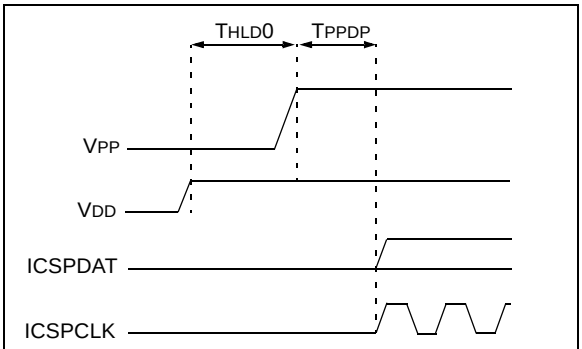
To prevent a device configured with INTOSC and internal MCLR from executing after exiting Program/Verify mode, VDD needs to power down before VPP. See Figure 4-3 for the timing.

FIGURE 4-1: VPP-FIRST PROGRAM/ VERIFY MODE ENTRY



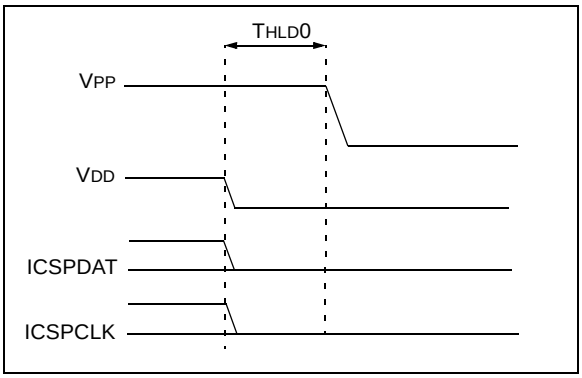
Note: This method of entry is valid, regardless of Configuration Word selected.

FIGURE 4-2: VDD-FIRST PROGRAM/ VERIFY MODE ENTRY



Note: This method of entry is valid if INTOSC and internal MCLR are not selected.

FIGURE 4-3: PROGRAM/VERIFY MODE EXIT



4.1 Program/Erase Algorithms

The PIC16F616 program memory may be written in two ways. The fastest method writes four words at a time. However, one-word writes are also supported for backward compatibility with previous 8-pin and 14-pin Flash devices. The four-word algorithm is used to program the program memory only. The one-word algorithm can write any available memory location (i.e., program memory, configuration memory and calibration memory).

Note: The PIC12F615, PIC12F609, PIC16F610 program memories must be written in one-word writes only.

After writing the array, the PC may be reset and read back to verify the write. It is not possible to verify immediately following the write because the PC can only increment, not decrement.

A device Reset will clear the PC and set the address to '0'. The Increment Address command will increment the PC. The Load Configuration command will set the PC to 0x2000. The available commands are shown in Table 4-1.

4.1.1 ONE-WORD PROGRAMMING

The PIC12F615, PIC12F609, PIC16F616 and PIC16F610 program memories can be written one word at a time to allow compatibility with other 8-pin and 14-pin Flash PIC® devices. Configuration memory (>0x2000) must be written one word (or byte) at a time.

The sequences for programming one word of program memory at a time is as follows:

1. Load a word at the current program memory address using the Load Data For Program Memory command.
2. Issue a Begin Programming (externally timed) command.
3. Wait TPROG1.
4. Issue End Programming command.
5. Issue an Increment Address command.
6. Repeat this sequence as required to write program, calibration or configuration memory.

See Figure 4-11 for more information.

4.1.2 FOUR-WORD PROGRAMMING

The PIC16F616 program memory can be written four words at a time using the four-word algorithm. Configuration memory (addresses >0x2000) and non-aligned (addresses modulo 4 not equal to zero) starting addresses must use the one-word programming algorithm.

This algorithm writes four sequential addresses in program memory. The four addresses must point to a four-word block which address modulo 4 of 0, 1, 2 and 3. For example, programming address 4 through 7 can be programmed together. Programming addresses 2 through 5 will create an unexpected result.

The sequence for programming four words of program memory at a time is as follows:

1. Load a word at the current program memory address using the Load Data For Program Memory command. This location must be address modulo 4 equal to 0.
2. Issue an Increment Address command to point to the next address in the block.
3. Load a word at the current program memory address using the Load Data For Program Memory command.
4. Issues an Increment Address command to point to the next address in the book.
5. Load a word at the current program memory address using the Load Data For Programming Memory command.
6. Issue and Increment Address command to point to the next address in the book.
7. Load a word at the current program memory address using the Load Data For Programming Memory command.
8. Issue a Begin Programming command externally timed.
9. Wait TPROG1.
10. Issue End Programming.
11. Wait TDIS.
12. Issue an Increment Address command to point to the start of the next block of addresses.
13. Repeat steps 1 through 12 as required to write the desired range of program memory.

See Figure 4-12 for more information.

Note:	Only the PIC16F616 program memory can be written to using the four-word programming algorithm.

4.1.3 ERASE ALGORITHMS

The PIC12F60X/12F61X/16F61X devices will erase different memory locations depending on the Program Counter (PC) and $\overline{CP}$. The following sequences can be used to erase noted memory locations. To erase the program memory and Configuration Word (0x2007), the following sequence must be performed. Note the Calibration Word (0x2008) and User ID (0x2000-0x2003) **will not** be erased.

1. Do a Bulk Erase Program Memory command.
2. Wait TERA to complete erase.

To erase the User ID (0x2000-0x2003), Configuration Word (0x2007) and program memory, use the following sequence. Note that the Calibration Word (0x2008) **will not** be erased.

1. Perform Load Configuration with dummy data to point the Program Counter (PC) to 0x2000.
2. Perform a Bulk Erase Program Memory command.
3. Wait TERA to complete erase.

To erase the User ID (0x2000-0x2003), Configuration Word (0x2007), Calibration Word (0x2008) and program memory, use the following sequence. Note that the Calibration Word (0x2008) **will** be erased.

1. Perform Load Configuration with dummy data to point the Program Counter (PC) to 0x2000.
2. Perform 8 Increment Address commands to point the PC to the Calibration Word at 0x2008.
3. Do a Bulk Erase Program Memory command.
4. Wait TERA to complete erase.

4.1.4 SERIAL PROGRAM/VERIFY OPERATION

The ICSPCLK pin is used as a clock input and the ICSPDAT pin is used for entering command bits and data input/output during serial operation. To input a command, ICSPCLK is cycled six times. Each command bit is latched on the falling edge of the clock with the LSb of the command being input first. The data input onto the ICSPDAT pin is required to have a minimum setup and hold time (see Table 7-1), with respect to the falling edge of the clock. Commands that have data associated with them (Read and Load) are specified to have a minimum delay of 1 µs between the command and the data. After this delay, the clock pin is cycled 16 times with the first cycle being a Start bit and the last cycle being a Stop bit.

During a Read operation, the LSb will be transmitted onto ICSPDAT pin on the rising edge of the second cycle. For a Load operation, the LSb will be latched on the falling edge of the second cycle. A minimum 1 µs delay is also specified between consecutive commands, except for the End Programming command, which requires a 100 µs TDIS.

All commands and data words are transmitted LSb first. Data is transmitted on the rising edge and latched on the falling edge of the ICSPCLK. To allow for decoding of commands and reversal of data pin configuration, a time separation of at least 1 µs is required between a command and a data word.

The commands that are available are described in Table 4-1.

TABLE 4-1: COMMAND MAPPING FOR PIC12F60X/12F61X/16F61X

Command	Mapping (MSb ... LSb)						Data
Load Configuration	x	x	0	0	0	0	0, data (14), 0
Load Data for Program Memory	x	x	0	0	1	0	0, data (14), 0
Read Data from Program Memory	x	x	0	1	0	0	0, data (14), 0
Increment Address	x	x	0	1	1	0	
Begin Programming	x	1	1	0	0	0	Externally Timed
End Programming	x	0	1	0	1	0	
Bulk Erase Program Memory	x	x	1	0	0	1	Internally Timed

4.1.4.1 Load Configuration

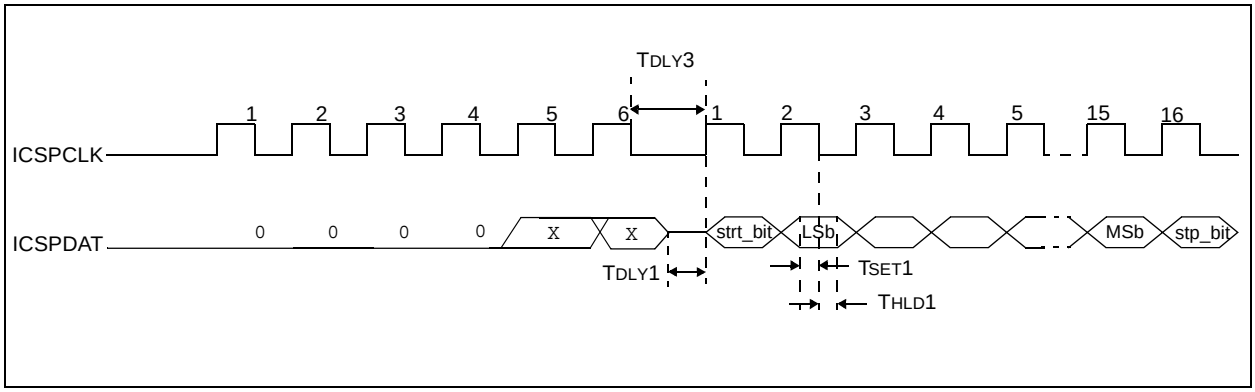
The Load Configuration command is used to access the Configuration Word (0x2007), User ID (0x2000-0x2003) and Calibration Word (0x2008). This command sets the Program Counter (PC) to address 0x2000 and loads the data latches with one word of data.

To access the configuration memory, send the Load Configuration command. Individual words within the configuration memory can be accessed by sending Increment Address commands and using load or read data for program memory.

After the 6-bit command is input, ICSPCLK pin is cycled an additional 16 times for the Start bit, 14 bits of data and Stop bit (see Figure 4-4).

After the configuration memory is entered, the only way to get back to the program memory is to exit the Program/Verify mode by taking MCLR low (V_{IL}).

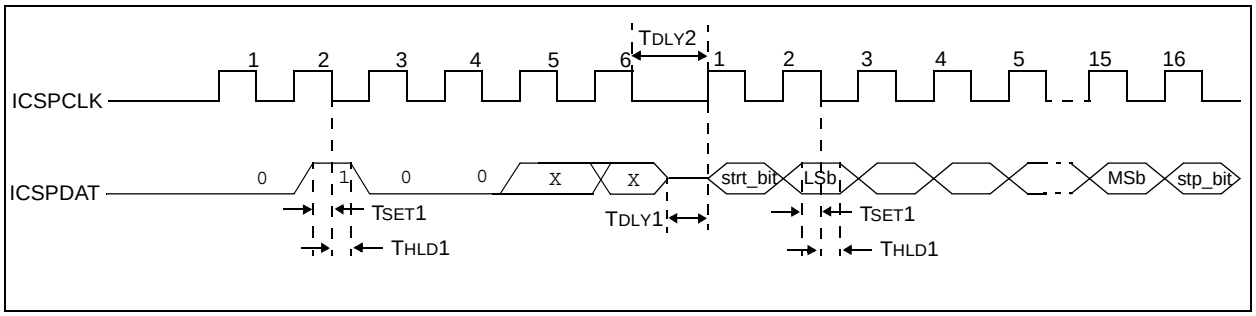
FIGURE 4-4: LOAD CONFIGURATION COMMAND



4.1.4.2 Load Data For Program Memory

After receiving this command, the chip will load in a 14-bit "data word" when 16 cycles are applied, as described previously. A timing diagram of this command is shown in Figure 4-5.

FIGURE 4-5: LOAD DATA FOR PROGRAM MEMORY COMMAND



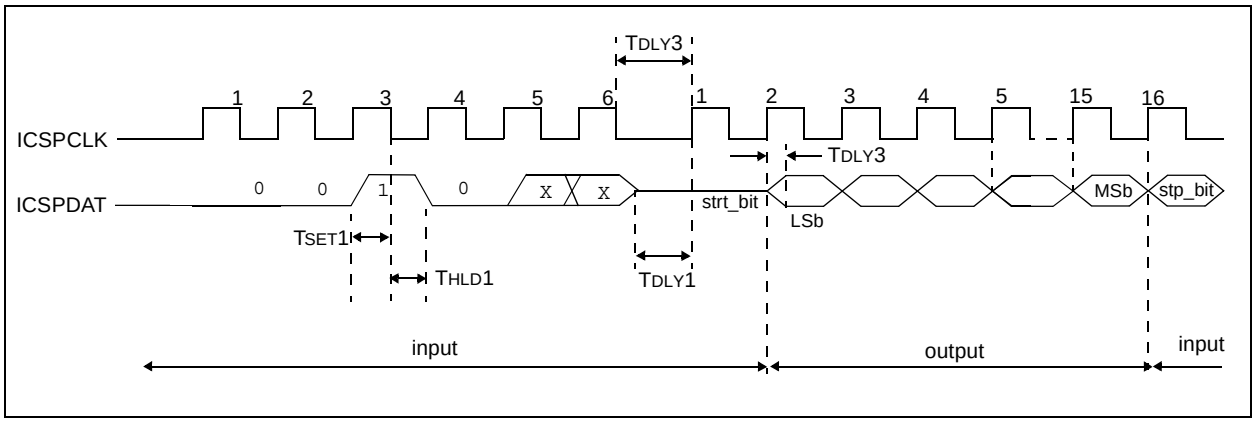
4.1.4.3 Read Data From Program Memory

After receiving this command, the chip will transmit data bits out of the program memory (user or configuration) currently accessed, starting with the second rising edge of the clock input. The data pin will go into Output mode on the second rising clock edge, and it will revert to Input mode (high-impedance) after the 16th rising edge.

If the program memory is code-protected ($\overline{CP}$ = 0), the data is read as zeros.

A timing diagram of this command is shown in Figure 4-6.

FIGURE 4-6: READ DATA FROM PROGRAM MEMORY COMMAND

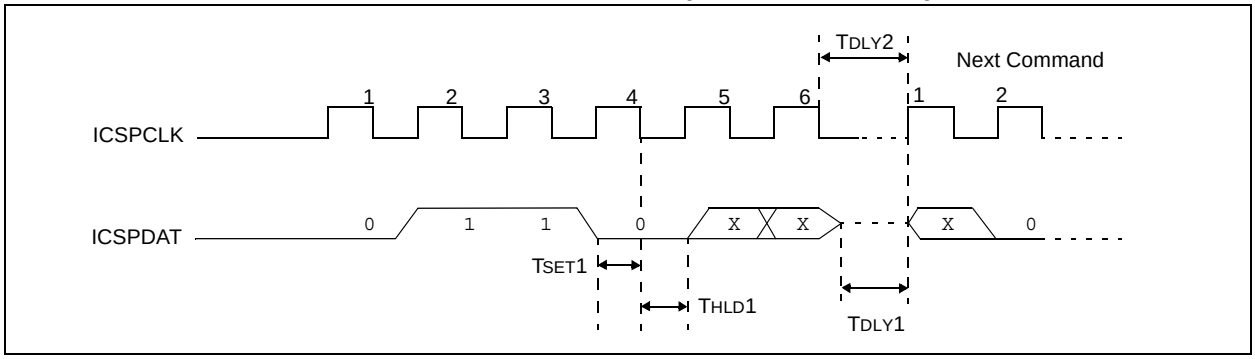


4.1.4.4 Increment Address

The PC is incremented when this command is received. A timing diagram of this command is shown in Figure 4-7.

It is not possible to decrement the address counter. To reset this counter, the user should exit and re-enter Program/Verify mode.

FIGURE 4-7: INCREMENT ADDRESS COMMAND (PROGRAM/VERIFY)

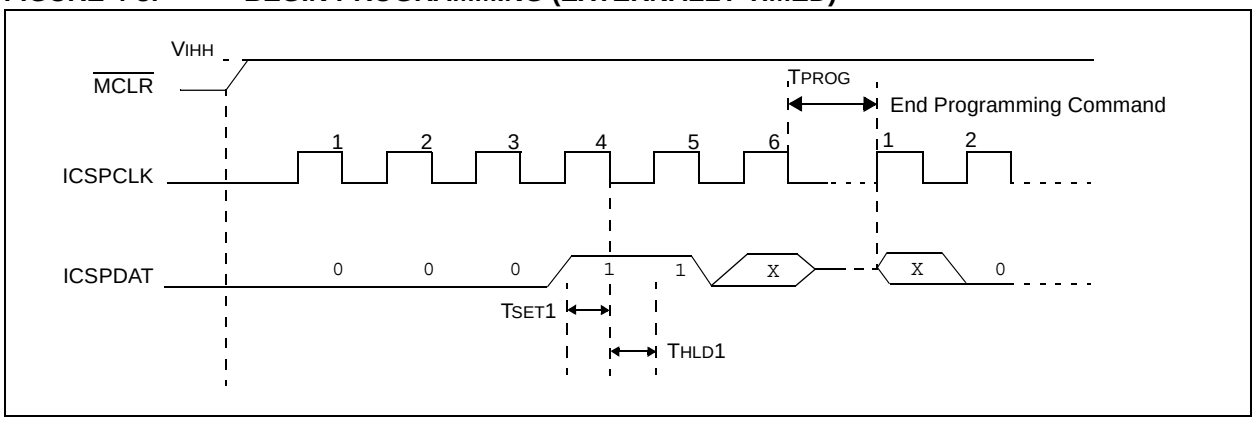


4.1.4.5 Begin Programming (Externally Timed)

A Load command must be given before every Begin Programming command. Programming of the appropriate memory (program memory, configuration or calibration memory) will begin after this command is received and decoded. Programming requires (TPROG) time and is terminated using an End Programming command. A timing diagram for this command is shown in Figure 4-8.

The addressed locations are not erased before programming.

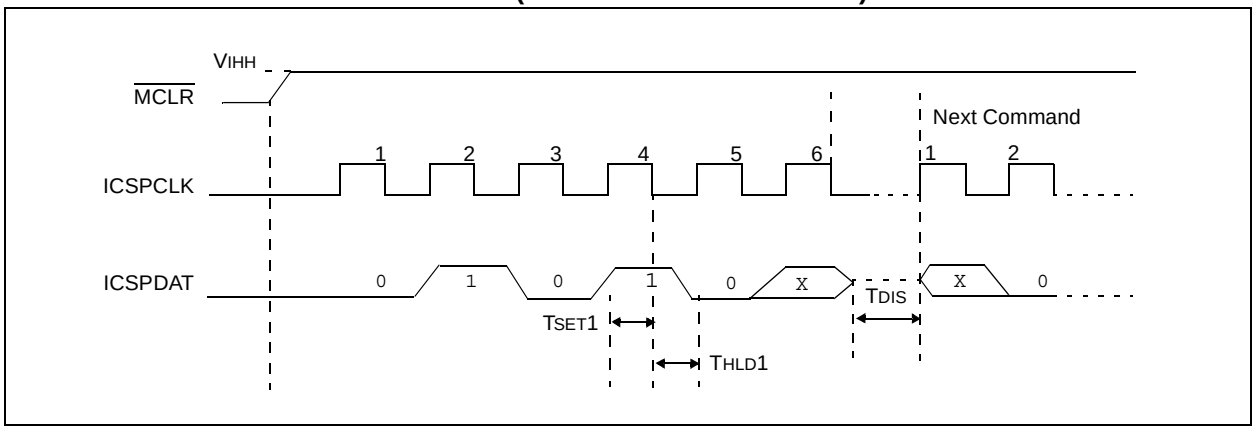
FIGURE 4-8: BEGIN PROGRAMMING (EXTERNALLY TIMED)



4.1.4.6 End Programming

After this command is performed, the write procedure will stop. A timing diagram of this command is shown in Figure 4-9.

FIGURE 4-9: END PROGRAMMING (SERIAL PROGRAM/VERIFY)



4.1.4.7 Bulk Erase Program Memory

After this command is performed, the entire program memory and Configuration Word (0x2007) is erased. The user ID and calibration memory may also be erased, depending on the value of the PC. See **Section 4.1.3 "Erase Algorithms"** for erase sequences. A timing diagram for this command is shown in Figure 4-10.

FIGURE 4-10: BULK ERASE PROGRAM MEMORY COMMAND

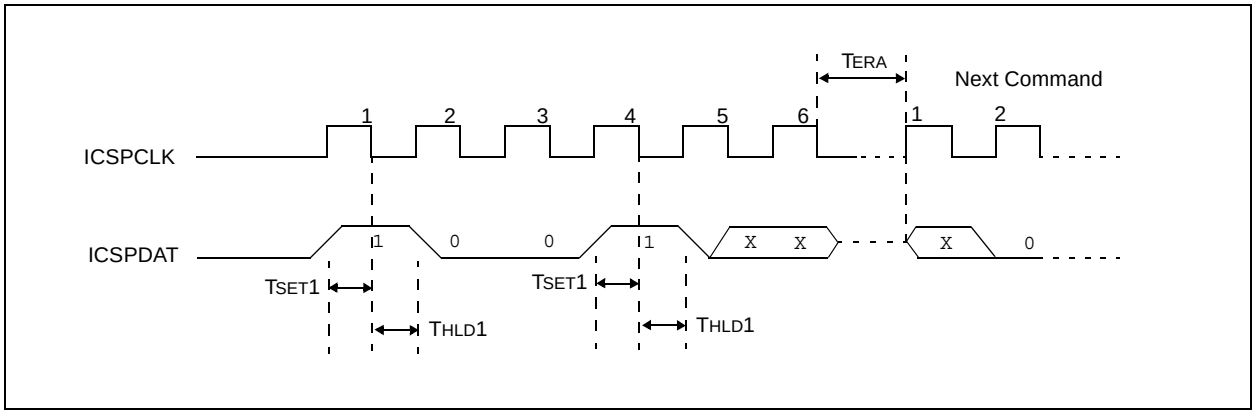
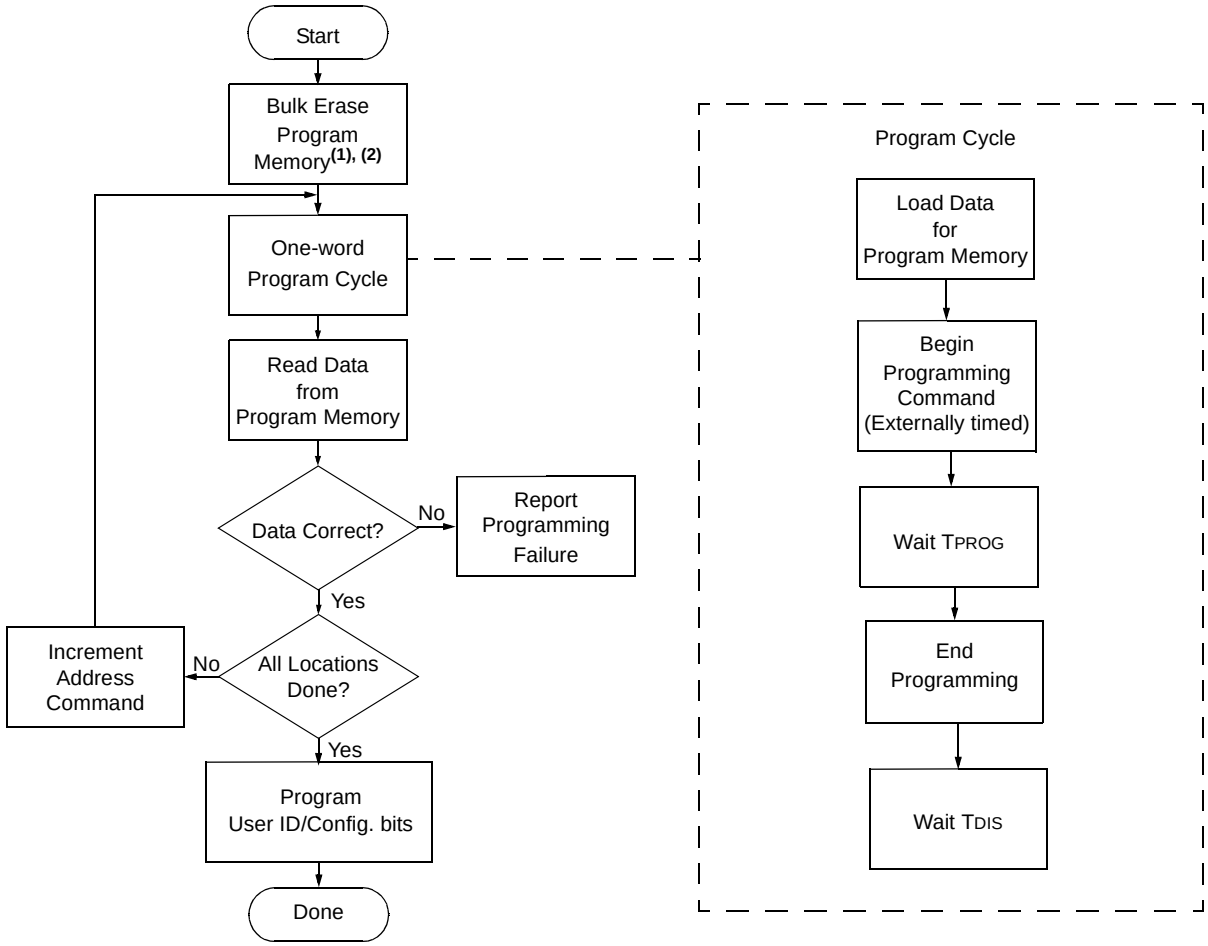


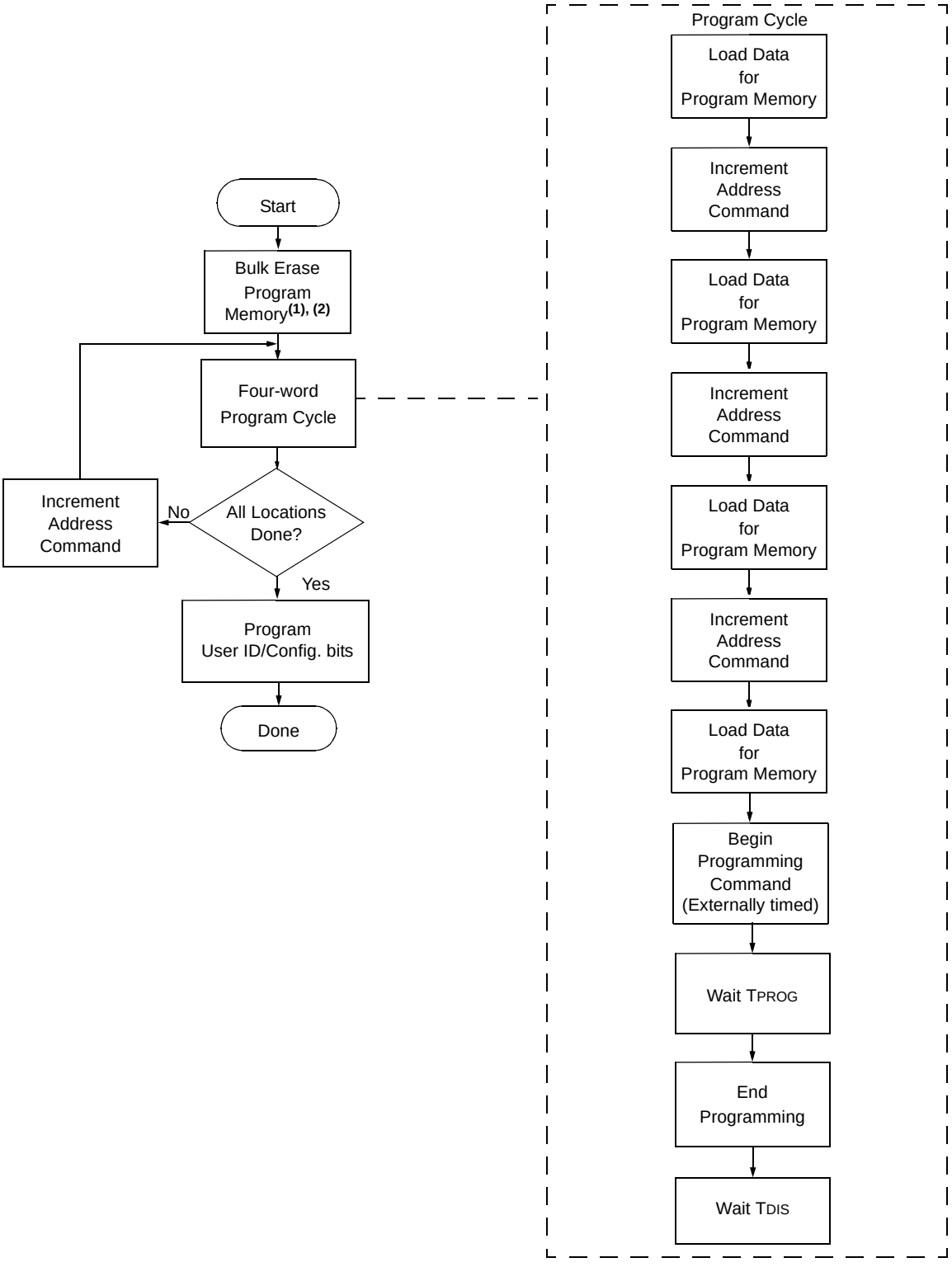
FIGURE 4-11: ONE-WORD PROGRAMMING FLOWCHART (PIC12F61X/16F61X AND PIC12F609)



Note 1: This step is optional if the device has already been erased or has not been previously programmed.

2: If the device is code-protected or must be completely erased, then bulk erase the device per Figure 4-14.

FIGURE 4-12: FOUR-WORD PROGRAMMING FLOWCHART (PIC16F616)



Note 1: This step is optional if the device is erased or not previously programmed.

2: If the device is code-protected or must be completely erased, then bulk erase the device per Figure 4-14.

FIGURE 4-13: PROGRAM FLOWCHART – CONFIGURATION MEMORY

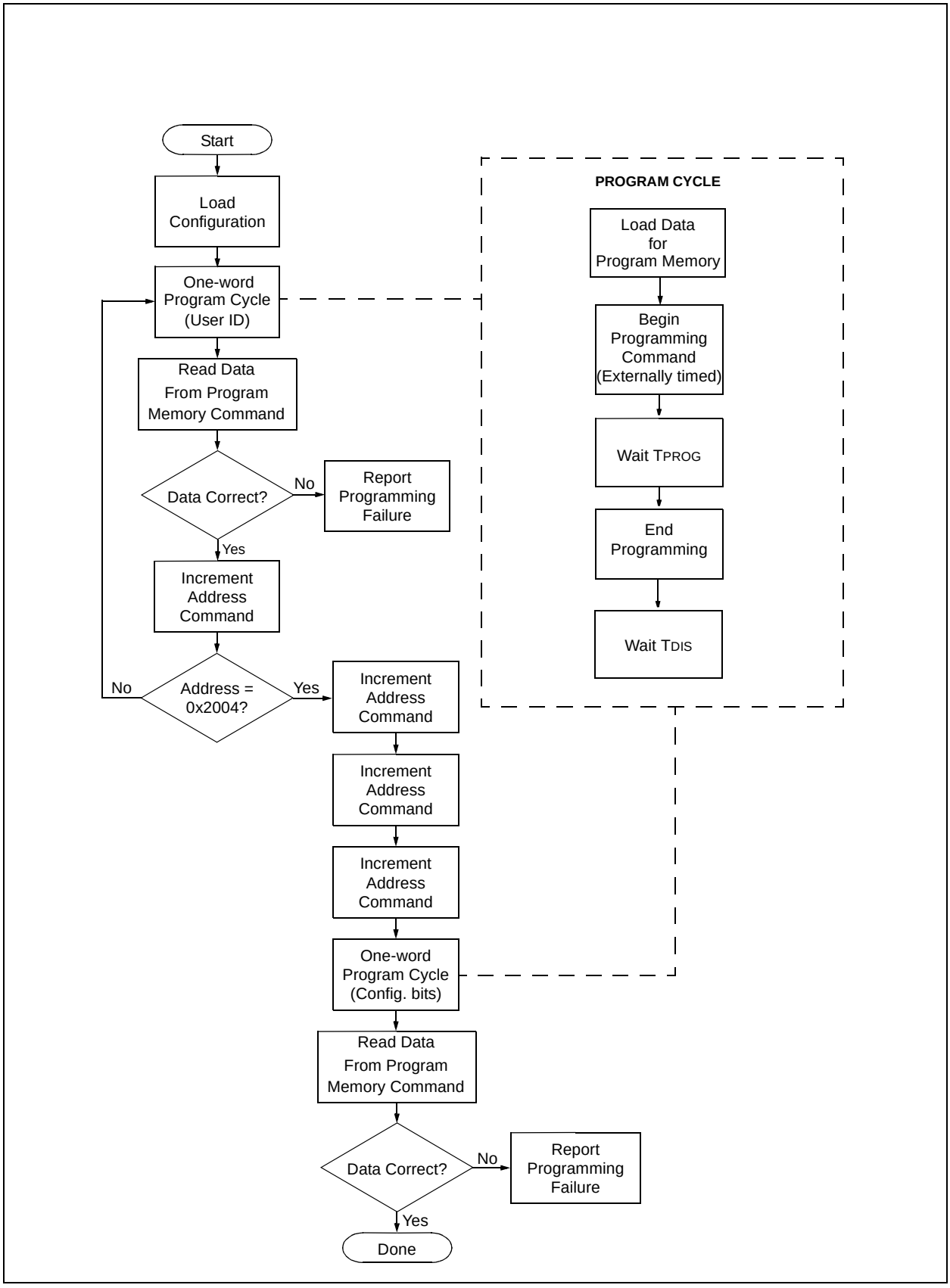
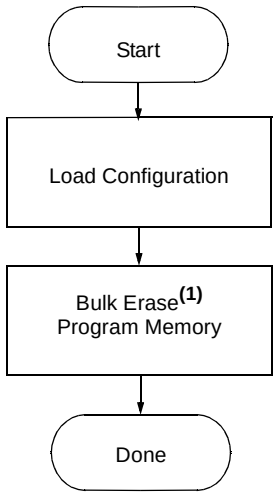


FIGURE 4-14: PROGRAM FLOWCHART – ERASE FLASH DEVICE



Note 1: See **Section 4.1.4.7 "Bulk Erase Program Memory"** for more information on the bulk erase procedure.

5.0 CONFIGURATION WORD

The PIC12F60X/12F61X/16F61X has several Configuration bits. These bits can be programmed (reads '0') or left unchanged (reads '1'), to select various device configurations.

REGISTER 5-1: CONFIG: CONFIGURATION WORD (ADDRESS: 2007h)

U-1	U-1	U-1	U-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	—	BOREN1	BOREN0	IOSCFS	$\overline{CP}$	MCLRE	$\overline{PWRTE}$	WDTE	FOSC2	FOSC1	FOSC0
bit 13													bit 0

bit 13-10 **Unimplemented:** Read as '1'

bit 9-8 **BOREN<1:0>:** Brown-out Reset Enable bits
11 = BOR enabled
10 = BOR enabled while running and disabled in Sleep
0x = BOR disabled

bit 7 **IOSCFS:** Internal Oscillator Frequency Select bit
1 = 8 MHz
0 = 4 MHz

bit 6 **$\overline{CP}$:** Code Protection bit
1 = Program memory is not code-protected
0 = Program memory is external read and write-protected

bit 5 **MCLRE:** $\overline{MCLR}$ Pin Function Select bit
1 = $\overline{MCLR}$ pin is $\overline{MCLR}$ function and weak internal pull-up is enabled
0 = $\overline{MCLR}$ pin is alternate function, $\overline{MCLR}$ function is internally disabled

bit 4 **$\overline{PWRTE}$:** Power-up Timer Enable bit(1)
1 = PWRT disabled
0 = PWRT enabled

bit 3 **WDTE:** Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled

bit 2-0 **FOSC<2:0>:** Oscillator Selection bits
000 = LP oscillator: Low-power crystal on RA5/GP5 and RA4/GP4
001 = XT oscillator: Crystal/resonator on RA5/GP5 and RA4/GP4
010 = HS oscillator: High-speed crystal/resonator on RA5/GP5 and RA4/GP4
011 = EC: I/O function on RA4/GP4, CLKIN on RA5/GP5
100 = INTOSCIO oscillator: I/O function on RA4/GP4, I/O function on RA5/GP5
101 = INTOSC oscillator: CLKOUT function on RA4/GP4, I/O function on RA5/GP5
110 = EXTRCIO oscillator: I/O function on RA4/GP4, RC on RA5/GP5
111 = EXTRC oscillator: CLKOUT function on RA4/GP4, RC on RA5/GP5

Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.

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

REGISTER 5-2: CALIB: CALIBRATION WORD (ADDRESS: 2008h)

U-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	U-1	U-1	U-1	U-1	U-1	U-1
—	FCAL6	FCAL5	FCAL4	FCAL3	FCAL2	FCAL1	FCAL0	—	—	—	—	—	—
bit 13													bit 0

bit 13 **Unimplemented:** Read as '1'

bit 12-6 **FCAL<6:0>:** Internal Oscillator Calibrations bits
0111111 = Maximum frequency
•
•
0000001
0000000 = Center frequency
1111111
•
•
1000000 = Minimum frequency

bit 5-0 **Unimplemented:** Read as '1'

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

5.1 Device ID Word

The device ID word for the PIC12F60X/12F61X/16F61X is loaded at 2006h. This location can not be erased.

TABLE 5-1: DEVICE ID VALUES

Device	Device ID Values	
	Dev	Rev
PIC12F615	10 0001 100	x xxxx
PIC12HV615	10 0001 101	x xxxx
PIC16F616	01 0010 010	x xxxx
PIC16HV616	01 0010 011	x xxxx
PIC12F609	10 0010 010	x xxxx
PIC12HV609	10 0010 100	x xxxx
PIC16F610	10 0010 011	x xxxx
PIC16HV610	10 0010 101	x xxxx

6.0 CODE PROTECTION

For PIC12F60X/12F61X/16F61X, once the $\overline{CP}$ bit is programmed to '0', all program memory locations read all '0's. The user ID locations and the Configuration Word read out in an unprotected fashion. Further programming is disabled for the entire program memory.

The user ID locations and the Configuration Word can be programmed regardless of the state of the $\overline{CP}$ bit.

6.1 Disabling Code Protection

It is recommended to use the procedure in Figure 4-14 to disable code protection of the device. This sequence will erase the program memory, Configuration Word (0x2007) and user ID locations (0x2000-0x2003). The Calibration Word (0x2008) **will not** be erased.

6.2 Embedding Configuration Word and User ID Information in the Hex File

To allow portability of code, the programmer is required to read the Configuration Word and user ID locations from the hex file when loading the hex file. If Configuration Word information was not present in the hex file, a simple warning message may be issued. Similarly, while saving a hex file, Configuration Word and user ID information must be included. An option to not include this information may be provided.

Microchip Technology Incorporated feels strongly that this feature is important for the benefit of the end customer.

6.3 Checksum Computation

6.3.1 CHECKSUM

Checksum is calculated by reading the contents of the PIC12F60X/12F61X/16F61X memory locations and adding up the opcodes up to the maximum user addressable location (e.g., 0x7FF for the PIC16F616). Any carry bits exceeding 16 bits are neglected. Finally, the Configuration Word (appropriately masked) is added to the checksum. Checksum computation for the PIC12F60X/12F61X/16F61X devices is shown in Table 6-1.

The checksum is calculated by summing the following:

- The contents of all program memory locations
- The Configuration Word, appropriately masked
- Masked user ID locations (when applicable)

The Least Significant 16 bits of this sum is the checksum.

Table 6-1 describes how to calculate the checksum for each device. Note that the checksum calculation differs depending on the code-protect setting. Since the program memory locations read out zeroes when code-protected, the table describes how to manipulate the actual program memory values to simulate values that would be read from a protected device. When calculating a checksum by reading a device, the entire program memory can simply be read and summed. The Configuration Word and user ID locations can always be read regardless of code-protect setting.

Note:	Some older devices have an additional value added in the checksum. This is to maintain compatibility with older device programmer checksums.

TABLE 6-1: CHECKSUM COMPUTATIONS

Device	Code Protect	Checksum*	Blank Value	0x25E6 at 0 and Max. Address
PIC12F615/HV615	$\overline{CP}$ = 1 $\overline{CP}$ = 0	SUM[0x000:0x03FF] + (CFGW & 03FF) (CFGW & 1FFF) + SUM_ID	0xFFFF 0x03BE	0xCBCD 0xCF8C
PIC12F609/HV609	$\overline{CP}$ = 1 $\overline{CP}$ = 0	SUM[0x000:0x03FF] + (CFGW & 03FF) (CFGW & 1FFF) + SUM_ID	0xFFFF 0x03BE	0xCBCD 0xCF8C
PIC16F610/HV610	$\overline{CP}$ = 1 $\overline{CP}$ = 0	SUM[0x000:0x03FF] + (CFGW & 03FF) (CFGW & 1FFF) + SUM_ID	0xFFFF 0x03BE	0xCBCD 0xCF8C
PIC16F616/HV616	$\overline{CP}$ = 1 $\overline{CP}$ = 0	SUM[0x000:0x07FF] + (CFGW & 03FF) (CFGW & 0x0FFF) + SUM_ID	0xFBFF 0xFFBE	0xC7CD 0xCB8C

Legend: CFGW = Configuration Word. Example calculations assume Configuration Word is erased (all '1's).
SUM[a:b] = [Sum of locations a to b inclusive]
SUM_ID = User ID locations masked by 0xF then made into a 16-bit value with ID0 as the Most Significant nibble.
For example, ID0 = 0x1, ID1 = 0x2, ID3 = 0x3, ID4 = 0x4, then SUM_ID = 0x1234.
The 4 LSb's of the unprotected checksum is used for the example calculations.
*Checksum = [Sum of all the individual expressions] MODULO [0xFFFF]
+ = Addition
& = Bitwise AND

7.0 PROGRAM/VERIFY MODE ELECTRICAL CHARACTERISTICS

TABLE 7-1: AC/DC CHARACTERISTICS TIMING REQUIREMENTS FOR PROGRAM/VERIFY MODE

AC/DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +85°C Operating Voltage 4.5V ≤ VDD ≤ 5.5V				
Sym.	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions/Comments**
General						
VDD	VDD level for read/write operations, program and data memory	2.0 2.0	— —	5.5 4.9(1)	V V	PIC16F616/F610, PIC12F615/F609 PIC16HV616/HV610, PIC12HV615/HV609
	VDD level for bulk erase operations, program and data memory	4.5 4.5	— —	5.5 4.9(1)	V V	PIC12F60X/12F61X/ 16F61X PIC16HV616/HV610, PIC12HV615/HV609
VIHH	High voltage on $\overline{MCLR}$ for Program/Verify mode entry	10	—	13	V	
IIHH	$\overline{MCLR}$ current during programming	—	300	1000	µA	
TVHHR	$\overline{MCLR}$ rise time (VSS to VHH) for Program/Verify mode entry	—	—	1.0	µs	
TPPDP	Hold time after VPP changes	5	—	—	µs	
VIH1	(ICSPCLK, ICSPDAT) input high level	0.8 VDD	—	—	V	
VIL1	(ICSPCLK, ICSPDAT) input low level	0.2 VDD	—	—	V	
TSET0	ICSPCLK, ICSPDAT setup time before $\overline{MCLR}$↑ (Program/Verify mode selection pattern setup time)	100	—	—	ns	
THLD0	Hold time after VDD changes	5	—	—	µs	
Serial Program/Verify						
TSET1	Data in setup time before clock↓	100	—	—	ns	
THLD1	Data in hold time after clock↓	100	—	—	ns	
TDLY1	Data input not driven to next clock input (delay required between command/data or command/ command)	1.0	—	—	µs	
TDLY2	Delay between clock↓ to clock↑ of next command or data	1.0	—	—	µs	
TDLY3	Clock↓ to data out valid (during a Read Data command)	—	—	80	ns	
TERA	Erase cycle time	—	5	6	ms	
TPROG	Programming cycle time	3	—	—	ms	10°C ≤ TA ≤ +40°C
TDIS	Time delay from program to compare (HV discharge time)	100	—	—	µs	

Note 1: Exceeding the maximum voltage may cause the shunt regulator to draw excessive current and damage the device.

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.

Trademarks

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

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona, Gresham, Oregon and Mountain View, California. The Company's quality system processes and procedures are for its PIC® 8-bit MCUs, 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.

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
Alpharetta, GA
Tel: 770-640-0034
Fax: 770-640-0307

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

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

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
Habour 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 - Fuzhou
Tel: 86-591-8750-3506
Fax: 86-591-8750-3521

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

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 - Shunde
Tel: 86-757-2839-5507
Fax: 86-757-2839-5571

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

China - Xian
Tel: 86-29-8833-7250
Fax: 86-29-8833-7256

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

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 - Gumi
Tel: 82-54-473-4301
Fax: 82-54-473-4302

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

Malaysia - Penang
Tel: 60-4-646-8870
Fax: 60-4-646-5086

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: 886-3-572-9526
Fax: 886-3-572-6459

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

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 - 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

10/19/06

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for 8-bit Microcontrollers - MCU ***category:***

Click to view products by Microchip ***manufacturer:***

Other Similar products are found below :

009936B CY8C20524-12PVXIT CY8C28433-24PVXIT MB95F012KPFT-G-SNE2 MB95F013KPMC-G-SNE2 MB95F263KPF-G-SNE2 MB95F264KPFT-G-SNE2 MB95F398KPMC-G-SNE2 MB95F478KPMC2-G-SNE2 MB95F562KPF-G-SNE2 MB95F564KPF-G-SNE2 MB95F634KPMC-G-SNE2 MB95F636KWQN-G-SNE1 MB95F696KPMC-G-SNE2 MB95F698KPMC1-G-SNE2 MB95F698KPMC2-G-SNE2 MB95F698KPMC-G-SNE2 MB95F818KPMC1-G-SNE2 MC908JK1ECDWER MC9S08PA32AVLD MC9S08PT60AVLD R5F1076CMSPV0 R5F5631ECDFBV0 C8051F389-B-GQ C8051F392-A-GMR ISD-ES1600_USB_PROG 901015X S9S08SL8F1CTJR STM8TL53G4U6 PIC16F877-04/P-B R5F10Y17ASP#30 CY8C3MFIDOCK-125 403708R MB95F354EPF-G-SNE2 MB95F564KPFT-G-SNE2 MB95F564KWQN-G-SNE1 MB95F636KP-G-SH-SNE2 MB95F636KPMC-G-SNE2 MB95F694KPMC-G-SNE2 MB95F778JPMC1-G-SNE2 MB95F818KPMC-G-SNE2 MC908QY8CDWER MC9S08PT16AVLD MC9S08PT32AVLH MC9S08PT60AVLC MC9S08PT60AVLH C8051F500-IQR 400801H LC87F0G08AUJA-AH 026923G