PIC16F87XA

FLASH Memory Programming Specification

This document includes programming specifications for the following devices:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- PIC16F877A

1.0 PROGRAMMING THE PIC16F87XA

The PIC16F87XA is programmed using a serial method. The Serial mode will allow the PIC16F87XA to be programmed while in the user's system. This allows for increased design flexibility. This programming specification applies to PIC16F87XA devices in all packages.

1.1 Programming Algorithm Requirements

The programming algorithm used depends on the operating voltage (V_{DD}) of the PIC16F87XA device, or whether internal or external timing is desired.

Algorithm #	V_{DD} Range	Timing
1	$2.0V \le V_{DD} < 5.5V$	Internal; 4 ms/op
2	$4.5V \le V_{DD} \le 5.5V$	External; 1 ms/op

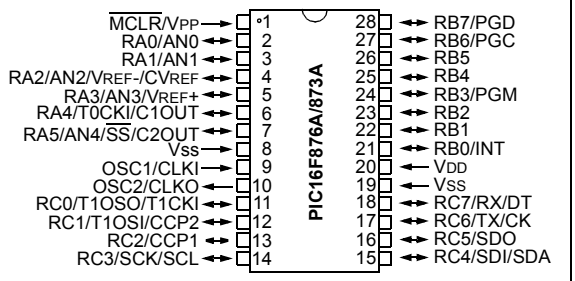
Both algorithms can be used with the two available programming entry methods. The first method follows the normal Microchip Programming mode entry of holding pins RB6 and RB7 low, while raising $\overline{MCLR}$ pin from V_{IL} to V_{IHH} (13V ± 0.5V). The second method, called Low Voltage ICSP™ or LVP for short, applies V_{DD} to $\overline{MCLR}$ and uses the I/O pin RB3 to enter Programming mode. When RB3 is driven to V_{DD} from ground, the PIC16F87XA device enters Programming mode.

1.2 Programming Mode

The Programming mode for the PIC16F87XA allows programming of user program memory, data memory, special locations used for ID, and the configuration word.

Pin Diagrams

PDIP, SOIC



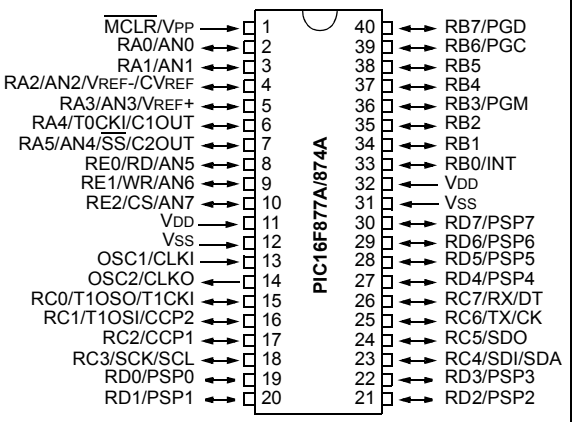


TABLE 1-1: PIN DESCRIPTIONS (DURING PROGRAMMING): PIC16F87XA

Pin Name	During Programming		
	Function	Pin Type	Pin Description
RB3	PGM	I	Low voltage ICSP programming input if LVP configuration bit equals '1'
RB6	CLOCK	I	Clock input
RB7	DATA	I/O	Data input/output
$\overline{MCLR}$	$V_{TEST\ MODE}$	P*	Program Mode Select
V_{DD}	V_{DD}	P	Power Supply
V_{SS}	V_{SS}	P	Ground

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

* To activate the Programming mode, high voltage needs to be applied to the $\overline{MCLR}$ input. Since $\overline{MCLR}$ is used for a level source, this means that $\overline{MCLR}$ does not draw any significant current.

2.0 PROGRAM MODE ENTRY

2.1 User Program Memory Map

The user memory space extends from 0000h to 1FFFh (8 K words). In Programming mode, the program memory space extends from 0000h to 3FFFh, with the first half (0000h - 1FFFh) being user program memory and the second half (2000h - 3FFFh) being configuration memory. The PC will increment from 0000h to 1FFFh and wrap around to 0000h. From 2000h, the PC will increment up to 3FFFh and wrap around to 2000h (not to 0000h). 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 2.4.

In the configuration memory space, 2000h - 200Fh are physically implemented. However, only locations 2000h through 2007h are available. Other locations are reserved. Locations beyond 200Fh will physically access user memory (see Figure 2-1).

2.2 Data EEPROM Memory

The EEPROM data memory space is a separate block of high endurance memory that the user accesses, using a special sequence of instructions. The amount of data EEPROM memory depends on the device and is shown below in number of bytes.

Device	# of Bytes
PIC16F873A	128
PIC16F874A	128
PIC16F876A	256
PIC16F877A	256

The contents of data EEPROM memory have the capability to be embedded into the HEX file.

The programmer should be able to read data EEPROM information from a HEX file and conversely (as an option), write data EEPROM contents to a HEX file, along with program memory information and configuration bit information.

The 256 data memory locations are logically mapped starting at address 2100h. The format for data memory storage is one data byte per address location, LSB aligned.

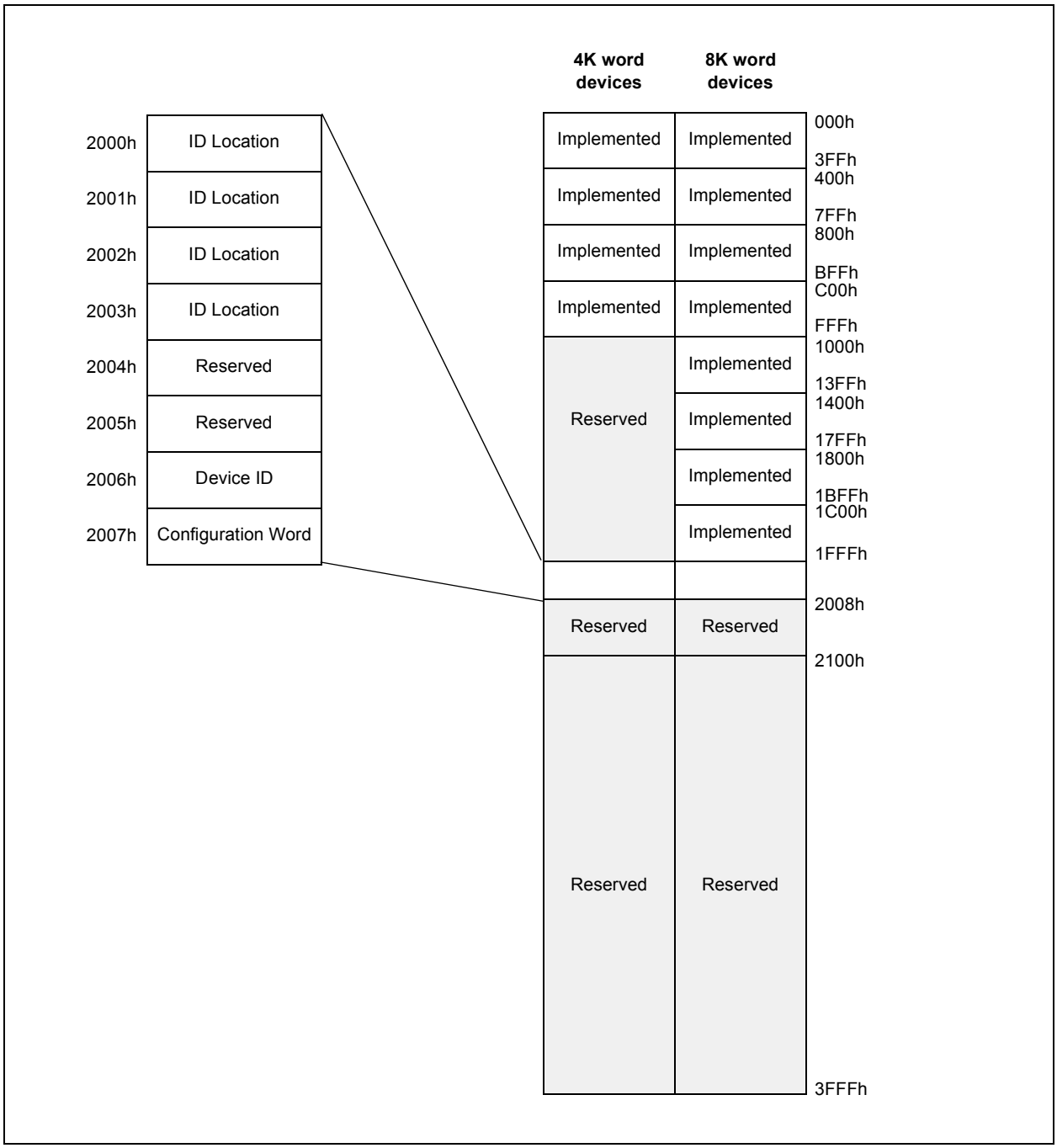
2.3 ID Locations

A user may store identification information (ID) in four ID locations. The ID locations are mapped in addresses 2000h - 2003h. It is recommended that the user use only the four Least Significant bits of each ID location. In some devices, the ID locations read out in an unscrambled fashion after code protection is enabled.

For these devices, it is recommended that ID location is written as "11 1111 1000 bbbb", where 'bbbb' is ID information.

In other devices, the ID locations read out normally, even after code protection. To understand how the devices behave, refer to Table 5-1.

FIGURE 2-1: PIC16F87XA PROGRAM MEMORY MAPPING



2.4 Program/Verify Mode

The Program/Verify mode is entered by holding pins RB6 and RB7 low, while raising MCLR pin from VIL to VIHH (high voltage). In this mode, the state of the RB3 pin does not effect programming. Low Voltage ICSP Programming mode is entered by raising RB3 from VIL to VDD, and then applying VDD to MCLR. Once in this mode, the user program memory and the configuration memory can be accessed and programmed in serial fashion. The mode of operation is serial, and the memory accessed is the user program memory. RB6 and RB7 are Schmitt Trigger inputs in this mode.

Note:	The OSC must not have 72 osc clocks while the device MCLR is between VIL and VIHH.

The sequence that enters the device into the Programming/Verify mode places all other logic into the RESET state (the MCLR pin was initially at VIL). This means all I/O are in the RESET state (high impedance inputs).

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 2000h. The available commands are shown in Table 2-1.

The normal sequence for programming eight program memory words at a time is as follows:

1. Load a word at the current program memory address using the 'Load Data' command.
2. Issue an 'Increment Address' command.
3. Load a word at the current program memory address using the 'Load Data' command.
4. Repeat Step 2 and Step 3 six times.
5. Issue a 'Begin Programming' command to begin programming.
6. Wait tprog (about 1 ms).
7. Issue an 'End Programming' command.
8. Increment to the next address.
9. Repeat this sequence as required to write program and configuration memory.

The alternative sequence for programming one program memory word at a time is as follows:

1. Set a word for the current memory location using the 'Load Data' command.
2. Issue a 'Begin Programming Only' command to begin programming.
3. Wait tprog.
4. Issue an 'End Programming' command.
5. Increment to the next address.
6. Repeat this alternative sequence as required to write program and configuration memory.

The address and program counter are reset to 0000h by resetting the device (taking MCLR below VIL) and re-entering Programming mode. Program and configuration memory may then be read or verified using the 'Read Data' and 'Increment Address' commands.

2.4.1 LOW VOLTAGE ICSP PROGRAMMING MODE

Low Voltage ICSP Programming mode allows a PIC16F87XA device to be programmed using VDD only. However, when this mode is enabled by a configuration bit (LVP), the PIC16F87XA device dedicates RB3 to control entry/exit into Programming mode.

When LVP bit is set to '1', the low voltage ICSP programming entry is enabled. Since the LVP configuration bit allows low voltage ICSP programming entry in its erased state, an erased device will have the LVP bit enabled at the factory. While LVP is '1', RB3 is dedicated to low voltage ICSP programming. Bring RB3 and then, MCLR to VDD to enter Programming mode. All other specifications for high voltage ICSP apply.

To disable Low Voltage ICSP mode, the LVP bit must be programmed to '0'. This must be done while entered with the High Voltage Entry mode (LVP bit = '1'). RB3 is now a general purpose I/O pin.

2.4.2 SERIAL PROGRAM/VERIFY OPERATION

The RB6 pin is used as a clock input pin, and the RB7 pin is used to enter command bits, and to input or output data during serial operation. To input a command, the clock pin (RB6) is cycled six times. Each command bit is latched on the falling edge of the clock, with the Least Significant bit (LSb) of the command being input first. The data on RB7 is required to have a minimum setup (tset1) and hold (thold1) time (see AC/DC specifications), with respect to the falling edge of the clock. Commands with associated data (read and load) are specified to have a minimum delay (tdly1) 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 (0) and the last cycle being a STOP bit (0). Data is transferred LSb first.

During a read operation, the LSb will be transmitted onto RB7 on the rising edge of the second cycle, and during a load operation, the LSb will be latched on the falling edge of the second cycle. A minimum 1 μs delay (tdly2) is specified between consecutive commands.

All commands and data words are transmitted LSb first. The data is transmitted on the rising edge, and latched on the falling edge of the clock. To allow decoding of commands and reversal of data pin configuration, a time separation of at least 1 μs (tdly1) is required between a command and a data word, or another command.

The available commands are described in the following paragraphs and listed in Table 2-1.

2.4.2.1 Load Configuration

After receiving this command, the program counter (PC) will be set to 2000h. By then applying 16 cycles to the clock pin, the chip will load 14 bits in a “data word,” as described above, to be programmed into the configuration memory. A description of the memory mapping schemes of the program memory for normal operation and configuration mode operation is shown in Figure 2-1. After the configuration memory is entered, the only way to get back to the user program memory is to exit the Program/Verify Test mode by taking $\overline{MCLR}$ low (V_{IL}).

2.4.2.2 Load Data for Program Memory

After receiving this command, the chip will load one word (with 14 bits as a “data word”) to be programmed into user program memory when 16 cycles are applied. A timing diagram for this command is shown in Figure 6-1.

2.4.2.3 Load Data for Data Memory

After receiving this command, the chip will load in a 14-bit “data word” when 16 cycles are applied. However, the data memory is only 8-bits wide, and thus, only the first 8 bits of data after the START bit will be programmed into the data memory. It is still necessary to cycle the clock the full 16 cycles in order to allow the internal circuitry to reset properly. The data memory contains up to 256 bytes. If the device is code protected, the data is read as all zeros. A timing diagram for this command is shown in Figure 6-2.

2.4.2.4 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 RB7 pin will go into Output mode on the second rising clock edge, and it will revert back to Input mode (hi-impedance) after the 16th rising edge. A timing diagram of this command is shown in Figure 6-3.

2.4.2.5 Read Data from Data Memory

After receiving this command, the chip will transmit data bits out of the data memory, starting with the second rising edge of the clock input. The RB7 pin will go into Output mode on the second rising edge, and it will revert back to Input mode (hi-impedance) after the 16th rising edge. As previously stated, the data memory is 8-bits wide, and therefore, only the first 8 bits that are output are actual data. A timing diagram for this command is shown in Figure 6-4.

2.4.2.6 Increment Address

The PC is incremented when this command is received. A timing diagram of this command is shown in Figure 6-5.

2.4.2.7 Begin Erase/Program Cycle

Eight locations must be loaded before every ‘Begin Erase/Programming’ command. After this command is received and decoded, eight words of program memory will be erased and programmed with the values contained in the program data latches. The PC address will decode which eight words are programmed. The lower three bits of the PC are ignored, so if the PC points to address 003h, then all eight locations from 000h to 007h are written.

An internal timing mechanism executes an erase before write. The user must allow the combined time for erase and programming, as specified in the electrical specs, for programming to complete. No ‘End Programming’ command is required.

1. If the address is pointing to user memory, the user memory alone will be affected.
2. If the address is pointing to the physically implemented test memory (2000h - 201Fh), test memory will be written. The configuration word will not be written unless the address is specifically pointing to 2007h.

This command can be used to perform programming over the entire V_{DD} range of the device.

Note 1:	The code protect bits cannot be erased with this command.
2:	All Begin Erase/Programming operations can take place over the entire V_{DD} range.

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

2.4.2.8 Begin Programming Only

Note:	Begin Programming Only operations must take place at the 4.5V to 5.5V V_{DD} range.

This command is similar to the ‘Erase/Programming Cycle’ command, except that a word erase is not done, and the internal timer is not used. Programming of program and data memory will begin after this command is received and decoded. The user must allow the time for programming, as specified in the electrical specs, for programming to complete. An ‘End Programming’ command is required.

The internal timer is not used for this command, so the ‘End Programming’ command must be used to stop programming.

1. If the address is pointing to user memory, the user memory alone will be affected.
2. If the address is pointing to the physically implemented test memory (2000h - 201Fh), the test memory will be written. The configuration word will not be written unless the address is specifically pointing to 2007h.

A timing diagram for this command is shown in Figure 6-7.

2.4.2.9 End Programming

After receiving this command, the chip stops programming the memory (test program memory or user program memory) that it was programming at the time.

Note:	This command will also set the write data shift latches to all '1's to avoid issues with downloading only one word before the write.

TABLE 2-1: COMMAND MAPPING FOR PIC16F87XA

Command	Mapping (MSB … LSB)					Data	Voltage Range
Load Configuration	0	0	0	0	0	0, data (14), 0	2.2V - 5.5V
Load Data for Program Memory	0	0	0	1	0	0, data (14), 0	2.2V - 5.5V
Read Data from Program Memory	0	0	1	0	0	0, data (14), 0	2.2V - 5.5V
Increment Address	0	0	1	1	0		2.2V - 5.5V
Begin Erase/Programming Cycle	0	1	0	0	0	4 ms typical, internally timed	2.2V - 5.5V
Begin Programming Only Cycle	1	1	0	0	0	1 ms typical, externally timed	4.5V - 5.5V
Bulk Erase Program Memory	0	1	0	0	1	4 ms typical, internally timed	4.5V - 5.5V
Bulk Erase Data Memory	0	1	0	1	1	4 ms typical, internally timed	4.5V - 5.5V
Chip Erase	1	1	1	1	1	4 ms typical, internally timed	4.5V - 5.5V
Load Data for Data Memory	0	0	0	1	1	0, data (14), 0	2.2V - 5.5V
Read Data from Data Memory	0	0	1	0	1	0, data (14), 0	2.2V - 5.5V
End Programming	1	0	1	1	1		

2.5 Erasing Program and Data Memory

Depending on the state of the code protection bits, program and data memory will be erased using different methods. The first two commands are used when both program and data memories are not code protected. The third command is used when either memory is code protected, or if you want to also erase the fuse locations, including the code protect bits. A device programmer should determine the state of the code protection bits and then apply the proper command to erase the desired memory.

2.5.1 ERASING NON-CODE PROTECTED PROGRAM AND DATA MEMORY

When both program and data memories are not code protected, they must be individually erased using the following commands. The only way that both memories are erased using a single command is if code protection is enabled for one of the memories. These commands do not erase the configuration word or ID locations.

2.5.1.1 Bulk Erase Program Memory

When this command is performed, and is followed by a 'Begin Erase/Programming' command, the entire program memory will be erased.

If the address is pointing to user memory, only the user memory will be erased.

If the address is pointing to the test program memory (2000h - 201Fh), then both the user memory and the test memory will be erased. The configuration word will not be erased, even if the address is pointing to location 2007h.

Previously, a load data with 0FFh command was recommended before any Bulk Erase. On these devices, this will not be required.

The Bulk Erase command is disabled when the CP bit is programmed to '0' enabling code protect.

A timing diagram for this command is shown in Figure 6-8.

2.5.1.2 Bulk Erase Data Memory

When this command is performed, and is followed by a 'Begin Erase/Programming' command, the entire data memory will be erased.

The Bulk Erase Data command is disabled when the CPD bit is programmed to '0' enabling protected data memory. A timing diagram for this command is shown in Figure 6-9.

Note:	All Bulk Erase operations must take place at the 4.5V to 5.5V VDD range.

2.5.1.3 Chip Erase

This command, when performed, will erase the program memory, EE data memory, and all of the fuse locations, including the code protection bits. All on-chip FLASH and EEPROM memory is erased, regardless of the address contained in the PC.

When a Chip Erase command is issued and the PC points to (0000h - 1FFFh), the configuration word and the user program memory will be erased, but not the test row (see Section 2.5.2.1). Chip Erase can also be used to erase code protected memory, as described in Section 2.5.2.

This command will also erase the code protect and code protect data fuses if they are programmed. This is the only command that allows a user to erase the code protect fuses.

The Chip Erase is internally self-timed to ensure that all program and data memory is erased before the code protect bits are erased. A timing diagram for this command is shown in Figure 6-10.

Note:	The Chip Erase operation must take place at the 4.5V to 5.5V VDD range.

2.5.2 ERASING CODE PROTECTED MEMORY

For the PIC16F87XA devices, once code protection is enabled, all protected program and data memory locations read all '0's and further programming is disabled. The ID locations and configuration word read out unscrambled and can be reprogrammed normally. The only command to erase a code protected PIC16F87XA device is the Chip Erase. This erases program memory, data memory, configuration bits and ID locations. **Since all data within the program and data memory will be erased when this command is executed, the security of the data or code is not compromised.**

2.5.2.1 Chip Erase

This command, when performed, will erase the program memory, data EEPROM, and all of the fuse locations, including the code protection bits, code protect fuses, and code protect data fuses. All on-chip FLASH and EEPROM memory is erased, regardless of the address contained in the PC.

If the PC points to user memory, the test row (2000h through 201Fh) is not erased with a Chip Erase command, except for the configuration word (at 2007h). If the test row is to be completely erased, the address in the PC must point to configuration memory.

When the PC points to 2000h - 201Fh, the configuration word, test program memory, and the user program memory will all be erased with a Chip Erase command. This allows the user to erase all program and configuration content, including the code protect bits, without compromising the user ID bits (2000h through 2004h), or any pass codes stored in the test row.

The Chip Erase is internally self-timed to ensure that all program and data memory is erased before the code protect bits are erased.

A timing diagram for this command is shown in Figure 6-10.

Note:	The Chip Erase operation must take place at the 4.5V to 5.5V VDD range.

FIGURE 2-2: ALGORITHM 1 FLOW CHART – PROGRAM MEMORY (2.0V ≤ VDD < 5.5V)

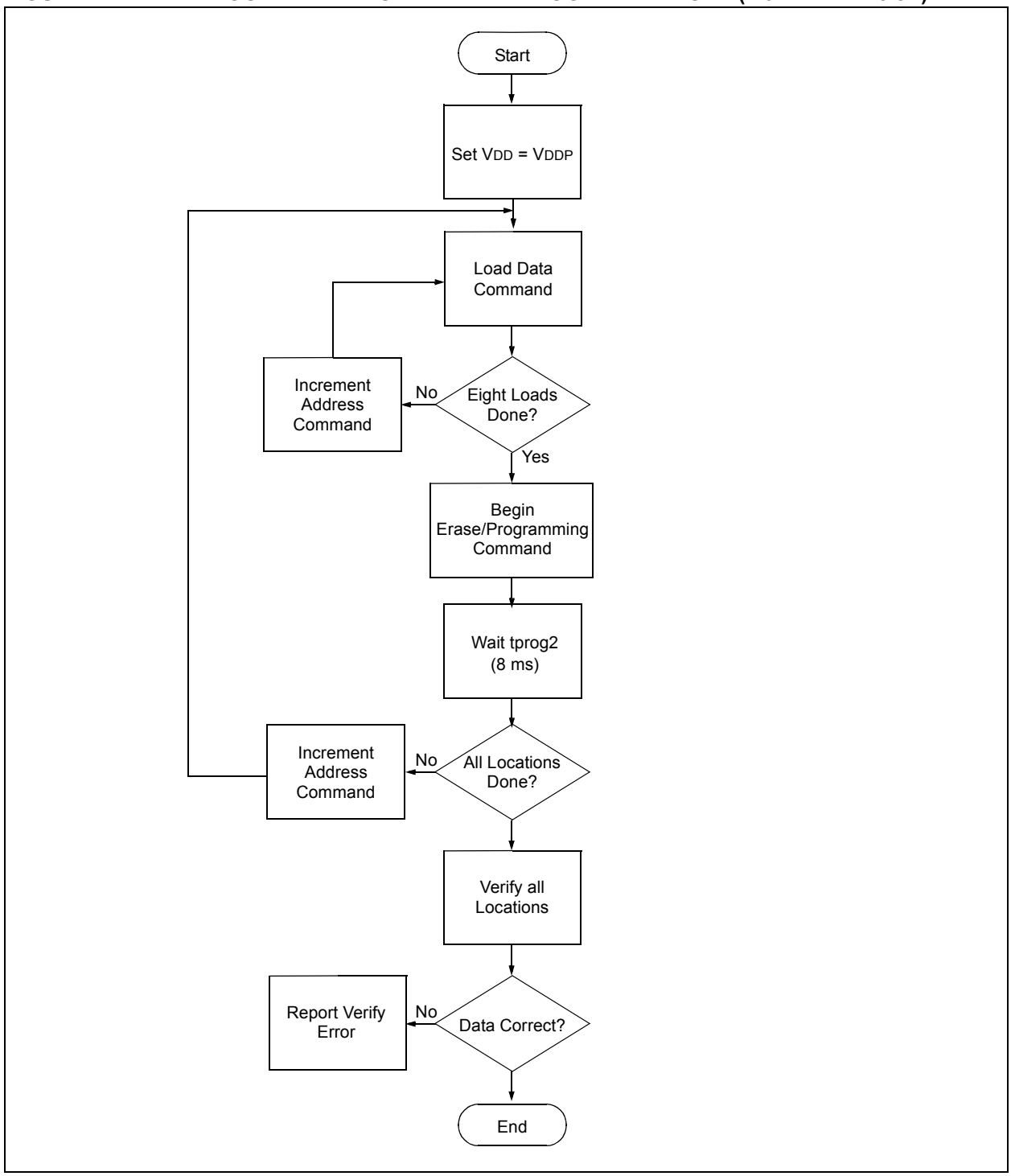


FIGURE 2-3: ALGORITHM 2 FLOW CHART – PROGRAM MEMORY (4.5V ≤ VDD ≤ 5.5V)

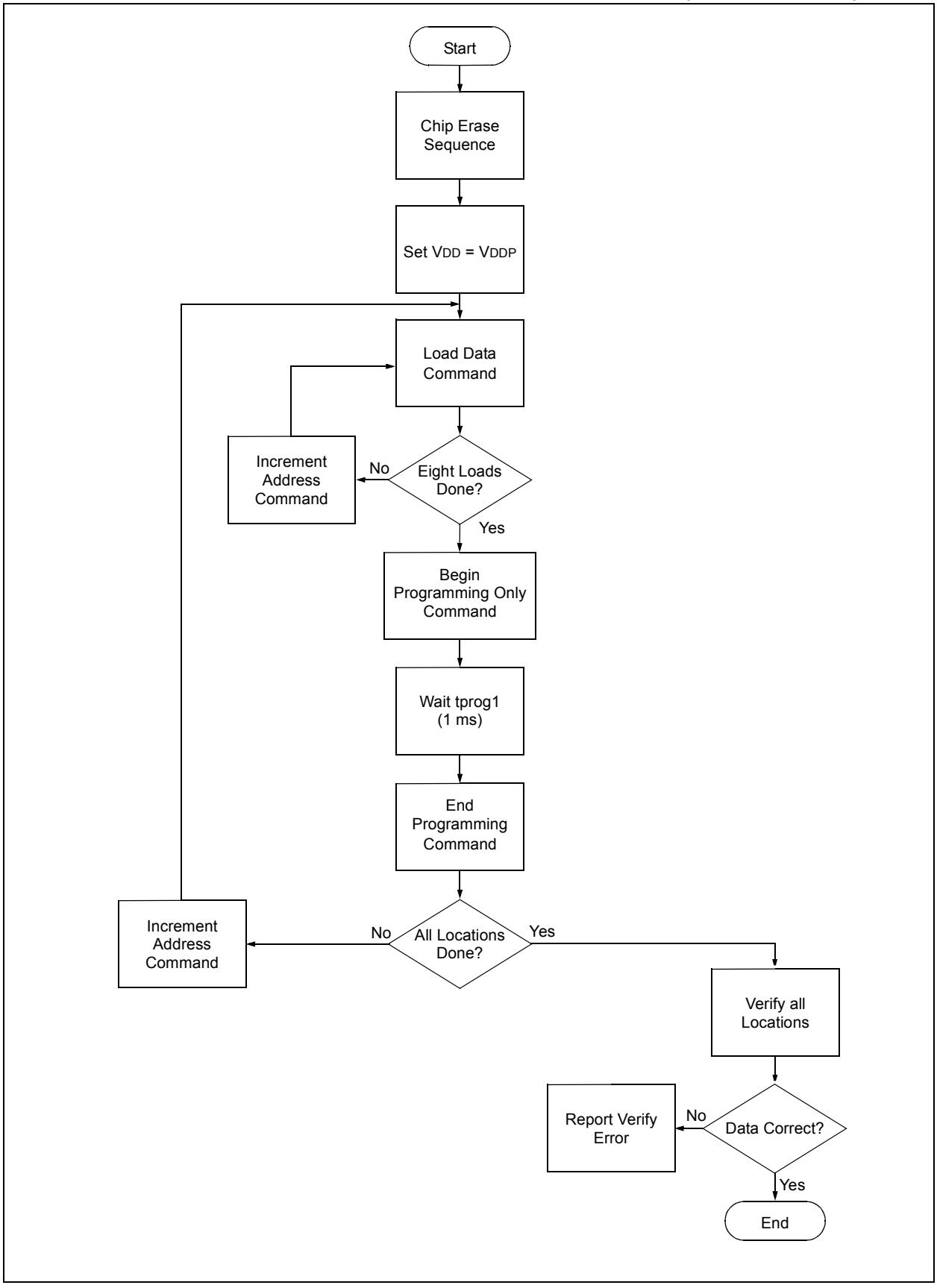


FIGURE 2-4: FLOW CHART – PIC16F87XA CONFIGURATION MEMORY (2.0V ≤ VDD < 5.5V)

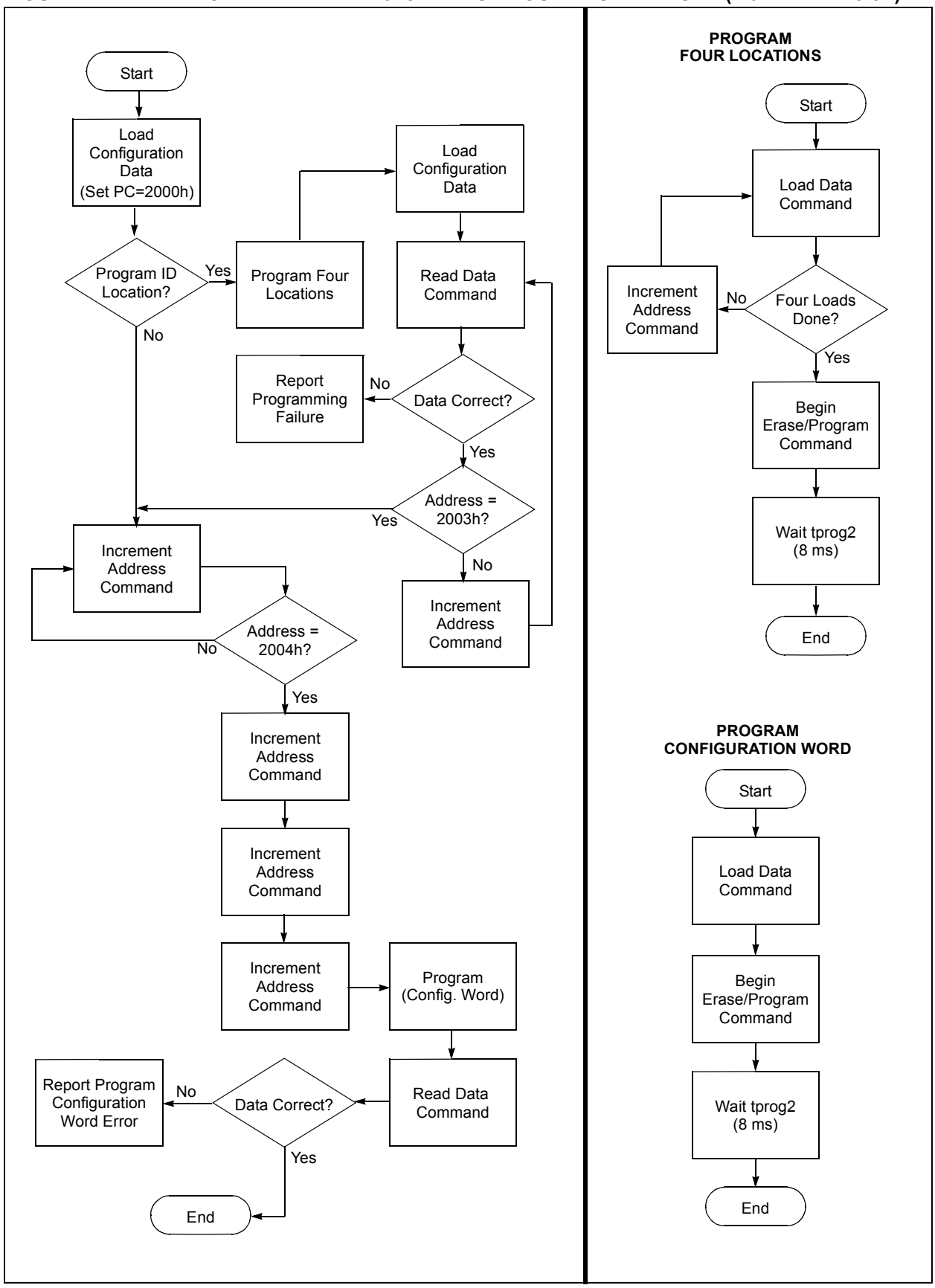
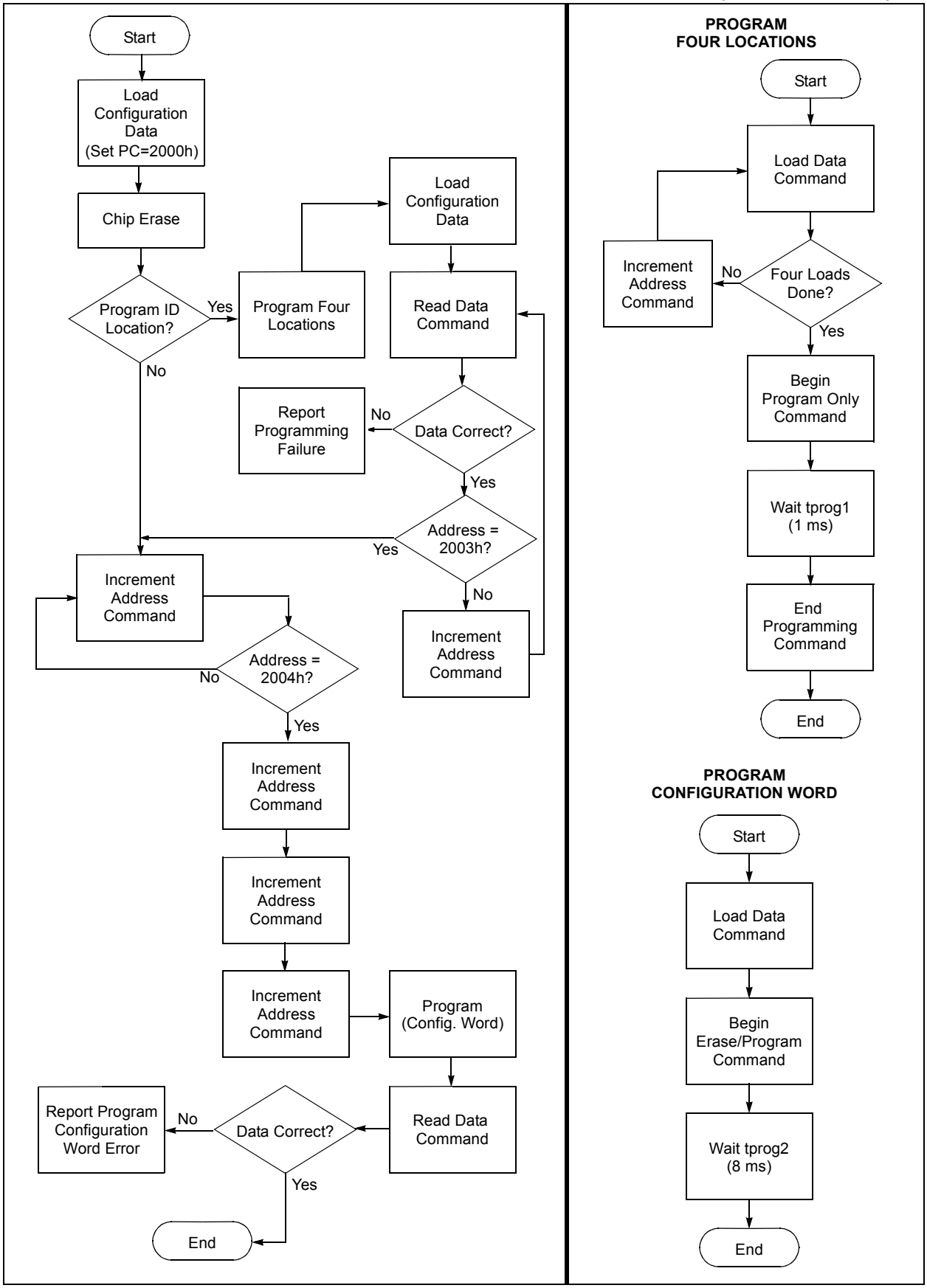


FIGURE 2-5: FLOW CHART – PIC16F87XA CONFIGURATION MEMORY (4.5V ≤ VDD ≤ 5.5V)



3.0 CONFIGURATION WORD

The PIC16F87XA has several configuration bits. These bits can be set (reads '0'), or left unchanged (reads '1'), to select various device configurations.

3.1 Device ID Word

The device ID word for the PIC16F87XA is located at 2006h.

TABLE 3-1: DEVICE ID VALUE

Device	Device ID Value	
	Dev	Rev
PIC16F873A	00 1110 0100	XXXX
PIC16F874A	00 1110 0110	XXXX
PIC16F876A	00 1110 0000	XXXX
PIC16F877A	00 1110 0010	XXXX

REGISTER 3-1: CONFIGURATION WORD REGISTER

R/P-1	U-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	U-1	U-1	R/P-1	R/P-1	R/P-1	R/P-1
CP	—	DEBUG	WRT1	WRT0	CPD	LVP	BOREN	—	—	$\overline{\text{PWRTEN}}$	WDTEN	FOSC1	FOSC0
bit 13													bit 0

bit 13 **CP:** FLASH Program Memory Code Protection bit
(PIC16F877A/876A):
1 = Code protection off
0 = 0000h to 1FFFh code protected
(PIC16F874A/873A):
1 = Code protection off
0 = 0000h to 0FFFh code protected
1000h to 1FFFh wraps to 0000h to 0FFFh

bit 12 **Unimplemented:** Read as '1'

bit 11 **DEBUG:** Background Debugger Mode bit
1 = Background debugger functions not enabled
0 = Background debugger functional

bit 10-9 **WRT<1:0>:** FLASH Program Memory Write Enable bits
(PIC16F877A/876A):
11 = Write protection off
10 = 0000h to 00FFh write protected, 0100h to 1FFFh may be modified by EECON control
01 = 0000h to 07FFh write protected, 0800h to 1FFFh may be modified by EECON control
00 = 0000h to 0FFFh write protected, 1000h to 1FFFh may be modified by EECON control
(PIC16F874A/873A):
11 = Write protection off
10 = 0000h to 00FFh write protected, 0100h to 0FFFh may be modified by EECON control
01 = 0000h to 03FFh write protected, 0400h to 0FFFh may be modified by EECON control
00 = 0000h to 07FFh write protected, 0800h to 1FFFh may be modified by EECON control

bit 8 **CPD:** Data EE Memory Code Protection bit
1 = Code protection off
0 = Data EE memory code protected

bit 7 **LVP:** Low Voltage Programming Enable bit
1 = RB3/PGM pin has PGM function, low voltage programming enabled
0 = RB3 is digital I/O, HV on $\overline{\text{MCLR}}$ must be used for programming

bit 6 **BOREN:** Brown-out Reset Enable bit
1 = BOR enabled
0 = BOR disabled

bit 5-4 **Unimplemented:** Read as '1'

bit 3 **$\overline{\text{PWRTEN}}$:** Power-up Timer Enable bit
1 = PWRT disabled
0 = PWRT enabled

bit 2 **WDTEN:** Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled

bit 1-0 **FOSC<1:0>:** Oscillator Selection bits
11 = RC oscillator
10 = HS oscillator
01 = XT oscillator
00 = LP oscillator

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '1'	
-n = Default value	1 = Bit is erased	0 = Bit is programmed	x = Bit is unknown

4.0 EMBEDDING CONFIGURATION WORD AND ID INFORMATION IN HEX FILE

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

Specifically for the PIC16F87XA, the EEPROM data memory should also be embedded in the HEX file (see Section 2.2).

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

5.0 CHECKSUM COMPUTATION

Checksum is calculated by reading the contents of the PIC16F87XA memory locations and adding up the opcodes up to the maximum user addressable location, e.g., 0x1FF for the PIC16F87XA. Any carry bits exceeding 16-bits are neglected. Finally, the configuration word (appropriately masked) is added to the checksum. Checksum computation for each member of the PIC16F87XA devices is shown in Table 5-1.

The checksum is calculated by summing the following:

- The contents of all program memory locations
- The configuration word, appropriately masked
- Masked ID locations (when applicable)

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

The following table 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 differently depending on the code protect setting, the table describes how to manipulate the actual program memory values to simulate the 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 ID locations can always be read.

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

TABLE 5-1: CHECKSUM COMPUTATION

Device	Code Protect	Checksum*	Blank Value	25E6h at 0 and max address
PIC16F873A	OFF	SUM[0000:0FFF] + (CFGW & 2FCF)	1FCF	EB9D
	ON	(CFGW & 2FCF) + SUM_ID	4F9E	1B6C
PIC16F874A	OFF	SUM[0000:0FFF] + (CFGW & 2FCF)	1FCF	EB9D
	ON	(CFGW & 2FCF) + SUM_ID	4F9E	1B6C
PIC16F876A	OFF	SUM[0000:1FFF] + (CFGW & 2FCF)	0FCF	DB9D
	ON	(CFGW & 2FCF) + SUM_ID	1F9E	EB6C
PIC16F877A	OFF	SUM[0000:1FFF] + (CFGW & 2FCF)	0FCF	DB9D
	ON	(CFGW & 2FCF) + SUM_ID	1F9E	EB6C

Legend: CFGW = Configuration Word
SUM[a:b] = [Sum of locations a to b inclusive]
SUM_ID = ID locations masked by 0Fh then made into a 16-bit value with ID0 as the most significant nibble. For example, ID0 = 01h, ID1 = 02h, ID3 = 03h, ID4 = 04h, then SUM_ID = 1234h
*Checksum = [Sum of all the individual expressions] **MODULO** [FFFFh]
\+ = Addition
& = Bitwise AND

6.0 PROGRAM/VERIFY MODE ELECTRICAL CHARACTERISTICS

TABLE 6-1: TIMING REQUIREMENTS FOR PROGRAM/VERIFY MODE

AC/DC CHARACTERISTICS POWER SUPPLY PINS	**Standard Operating Procedure (unless otherwise stated)** Operating temperature $0 \le T_A \le +70°C$ Operating Voltage $2.0V \le V_{DD} \le 5.5V$					
Characteristics	**Sym**	**Min**	**Typ**	**Max**	**Units**	**Conditions/Comments**
General						
V_{DD} level for Begin Erase/Program operations and EECON write of program memory	V_{DD}	2.0		5.5	V	
V_{DD} level for Begin Erase/Program operations and EECON write of data memory	V_{DD}	2.0		5.5	V	
V_{DD} level for Bulk Erase/Write, Chip Erase, and Begin Program operations, of program and data memory	V_{DD}	4.5		5.5	V	
Begin Programming Only cycle time	tprog1	1			ms	Externally Timed
Begin Erase/Programming	tprog2		4	8	ms	Internally Timed
Chip Erase cycle time	tprog3		4	8	ms	Internally Timed
High voltage on $\overline{MCLR}$ and RA4/T0CKI for Test mode entry	V_{IHH}	V_{DD} + 3.5		13.5	V	
$\overline{MCLR}$ rise time (V_{SS} to V_{HH}) for Test mode entry	tVHHR			1.0	μs	
(RB6, RB7) input high level	V_{IH1}	0.8 V_{DD}			V	Schmitt Trigger input
(RB6, RB7) input low level	V_{IL1}	0.2 V_{DD}			V	Schmitt Trigger input
RB<7:4> setup time before $\overline{MCLR}$↑ (Test mode selection pattern setup time)	tset0	100			ns	
RB<7:4> hold time after $\overline{MCLR}$↑ (Test mode selection pattern setup time)	thld0	5			μs	
Serial Program/Verify						
Data in setup time before clock↓	tset1	100			ns	
Data in hold time after clock↓	thld1	100			ns	
Data input not driven to next clock input (delay required between command/data or command/command)	tdly1	1.0			μs	$2.0V \le V_{DD} < 4.5V$
		100			ns	$4.5V \le V_{DD} \le 5.5V$
Delay between clock↓ to clock↑ of next command or data	tdly2	1.0			μs	$2.0V \le V_{DD} < 4.5V$
		100			ns	$4.5V \le V_{DD} \le 5.5V$
Clock↑ to data out valid (during read data)	tdly3	80			ns	

FIGURE 6-1: LOAD DATA FOR USER PROGRAM MEMORY COMMAND (PROGRAM/VERIFY)

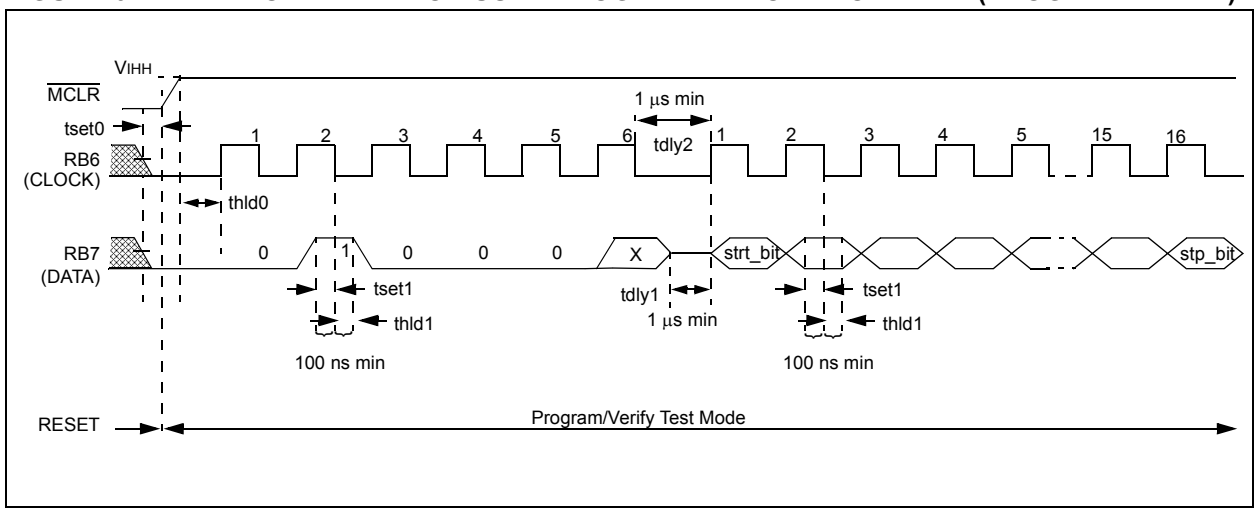


FIGURE 6-2: LOAD DATA FOR USER DATA MEMORY COMMAND (PROGRAM/VERIFY)

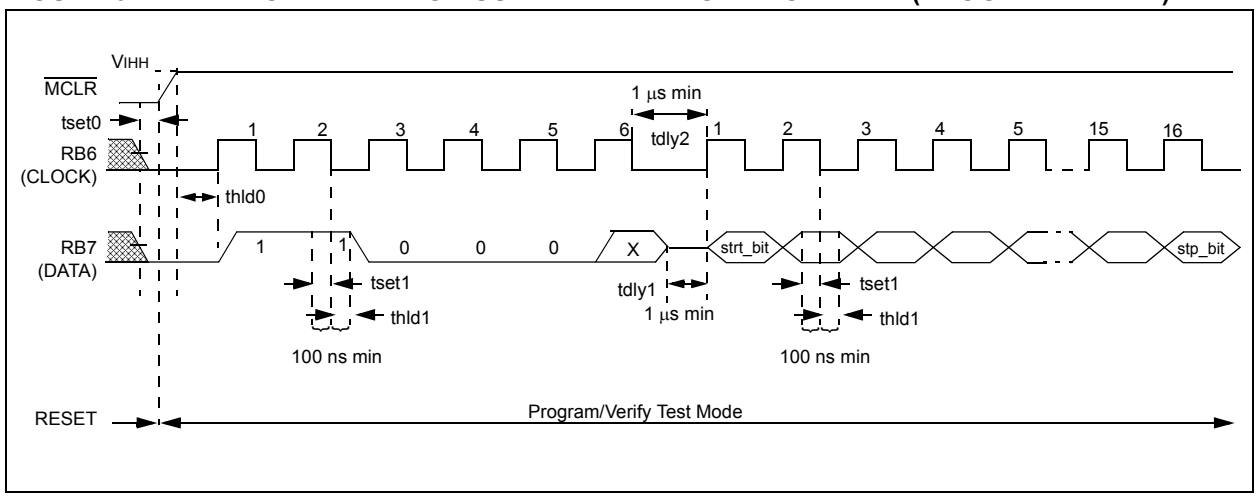


FIGURE 6-3: READ DATA FROM PROGRAM MEMORY COMMAND (PROGRAM/VERIFY)

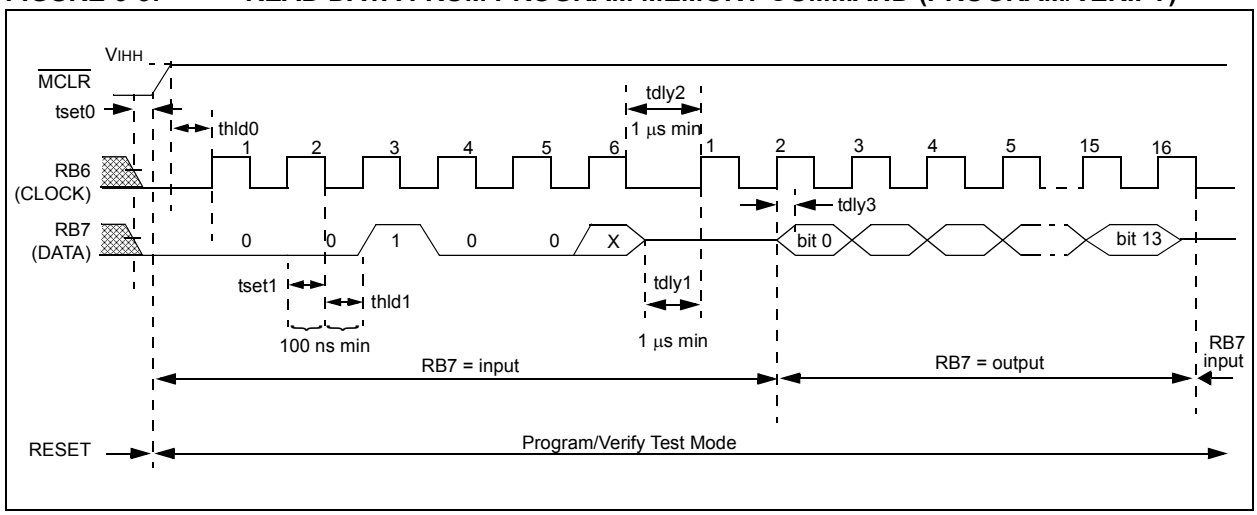


FIGURE 6-4: READ DATA FROM DATA MEMORY COMMAND (PROGRAM/VERIFY)

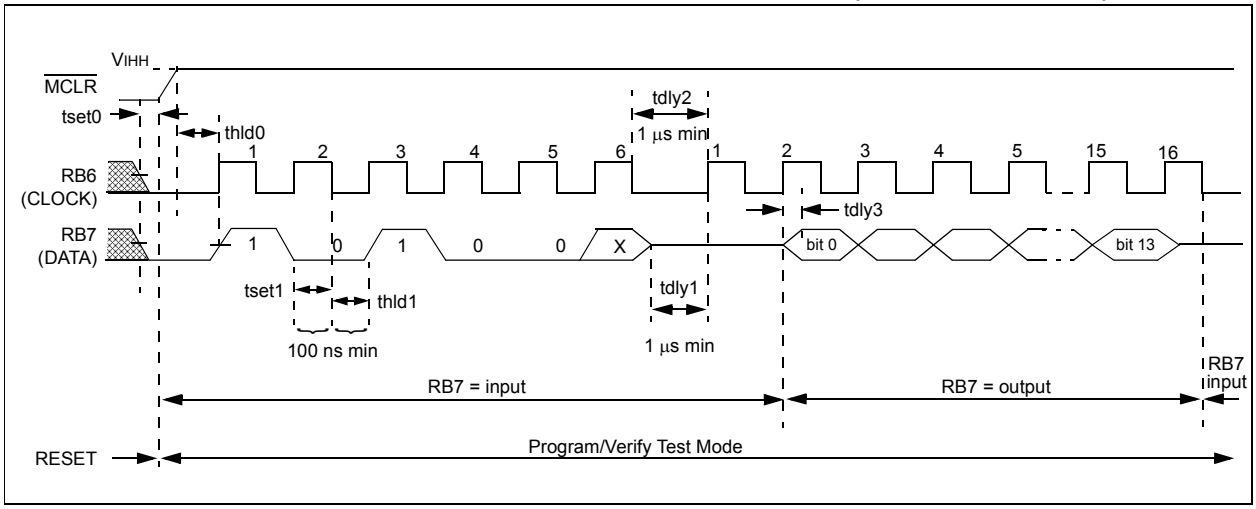


FIGURE 6-5: INCREMENT ADDRESS COMMAND (PROGRAM/VERIFY)

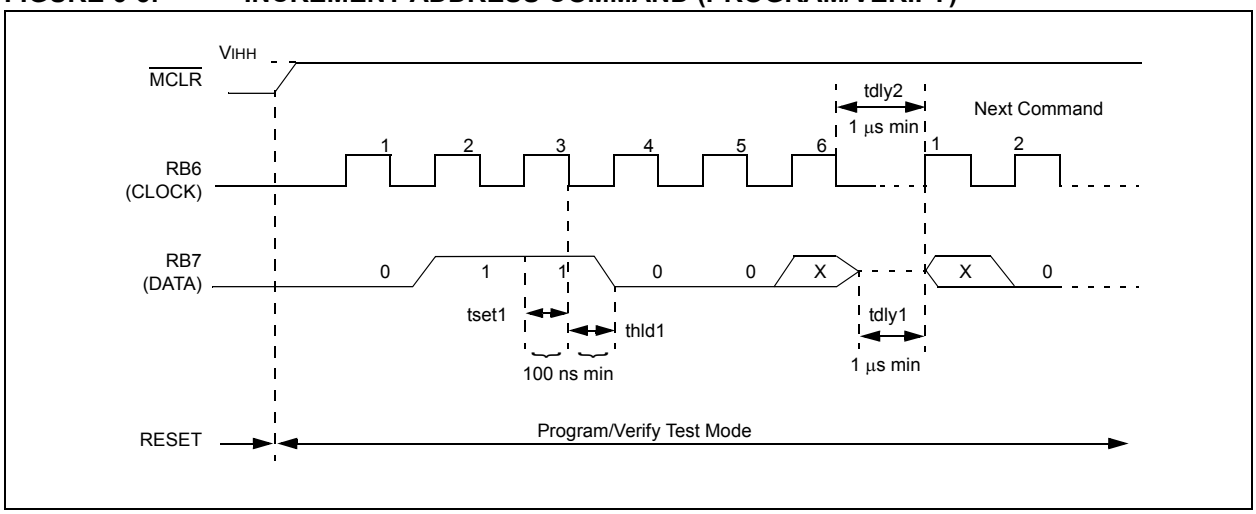


FIGURE 6-6: BEGIN ERASE/PROGRAMING COMMAND (PROGRAM/VERIFY)

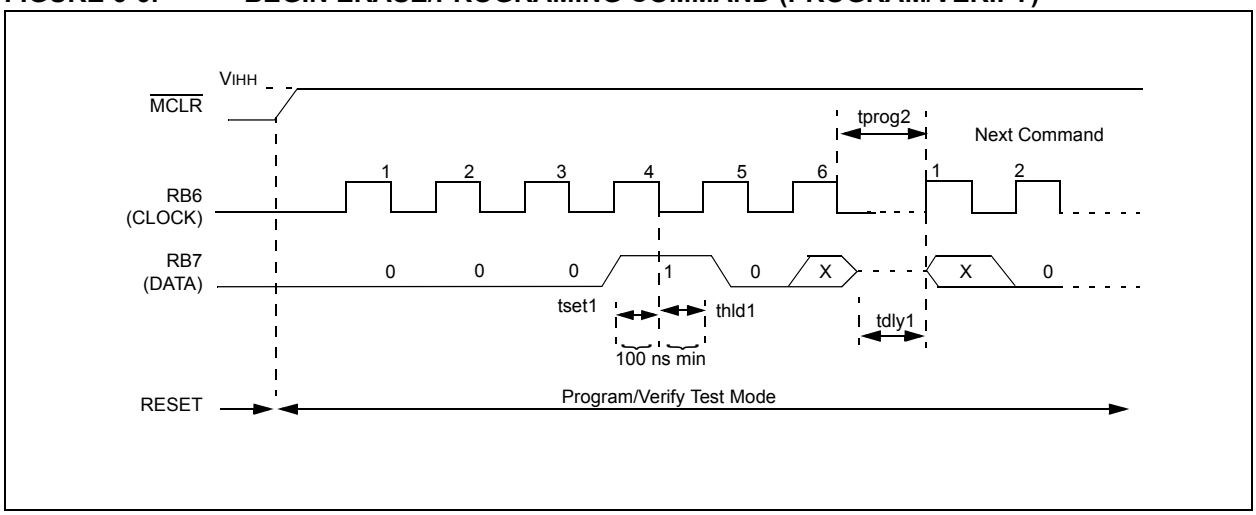


FIGURE 6-7: BEGIN PROGRAMING ONLY COMMAND (PROGRAM/VERIFY)

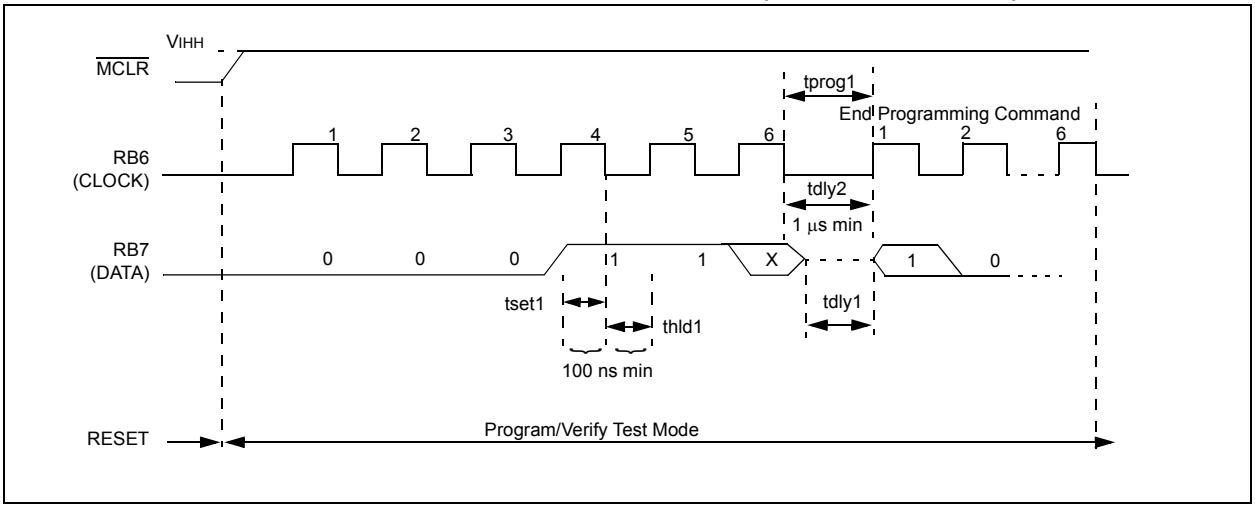


FIGURE 6-8: BULK ERASE PROGRAM MEMORY COMMAND (PROGRAM/VERIFY)

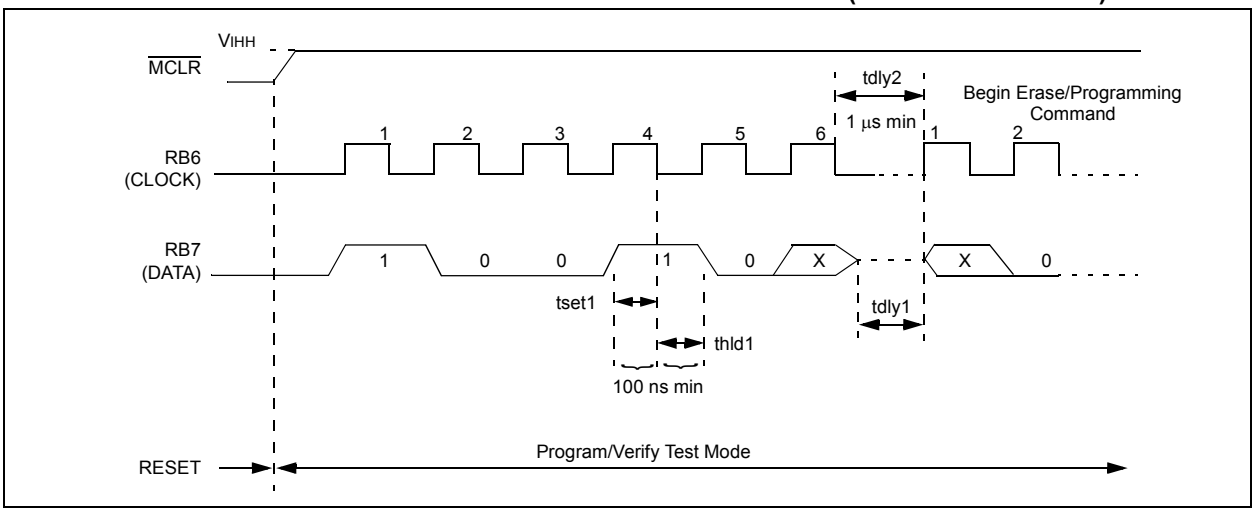


FIGURE 6-9: BULK ERASE DATA MEMORY COMMAND (PROGRAM/VERIFY)

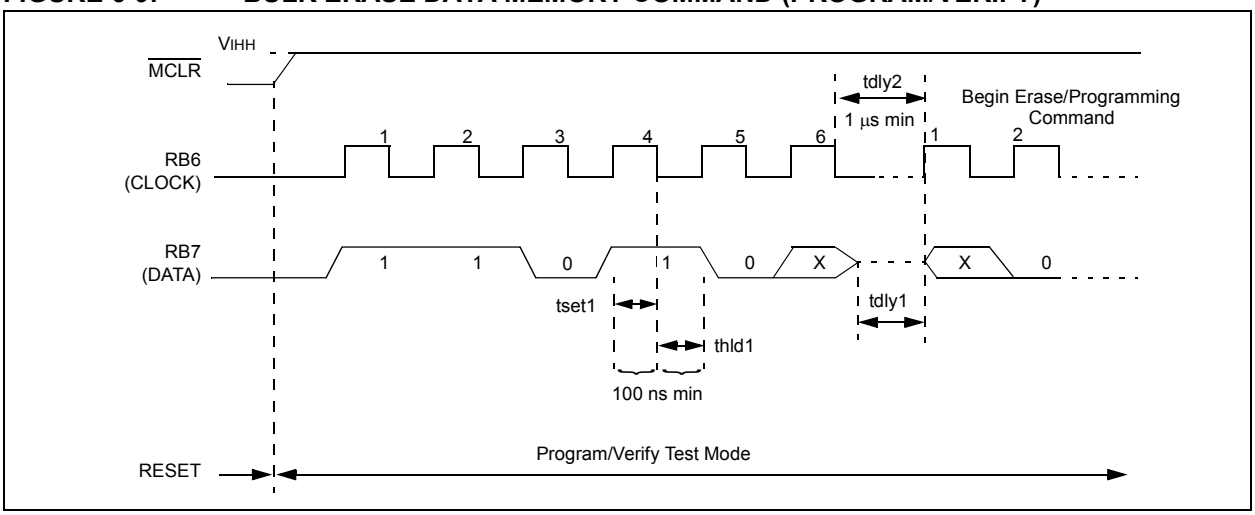
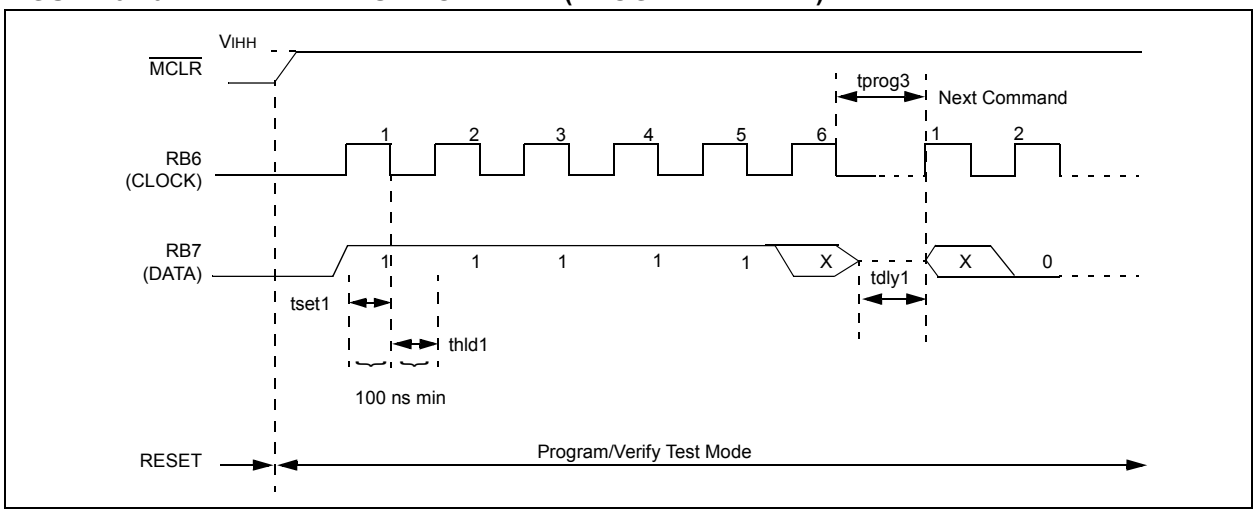


FIGURE 6-10: CHIP ERASE COMMAND (PROGRAM/VERIFY)



Note the following details of the code protection feature on PICmicro® MCUs.

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products 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 PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be 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 product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

DNV Certification, Inc. USA — ANSI-RAB QMS ACCREDITED — DNV — DNV MSC The Netherlands Accredited by the RvA — ISO 9001 / QS-9000 REGISTERED FIRM

Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999 and Mountain View, California in March 2002. The Company’s quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, non-volatile memory and analog products. In addition, Microchip’s quality system for the design and manufacture of development systems is ISO 9001 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: 480-792-7627
Web Address: http://www.microchip.com

Rocky Mountain
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7966 Fax: 480-792-4338

Atlanta
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770-640-0034 Fax: 770-640-0307

Boston
2 Lan Drive, Suite 120
Westford, MA 01886
Tel: 978-692-3848 Fax: 978-692-3821

Chicago
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

Dallas
4570 Westgrove Drive, Suite 160
Addison, TX 75001
Tel: 972-818-7423 Fax: 972-818-2924

Detroit
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

Kokomo
2767 S. Albright Road
Kokomo, Indiana 46902
Tel: 765-864-8360 Fax: 765-864-8387

Los Angeles
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

San Jose
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

Toronto
6285 Northam Drive, Suite 108
Mississauga, Ontario L4V 1X5, Canada
Tel: 905-673-0699 Fax: 905-673-6509

ASIA/PACIFIC

Australia
Microchip Technology Australia Pty Ltd
Suite 22, 41 Rawson Street
Epping 2121, NSW
Australia
Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

China - Beijing
Microchip Technology Consulting (Shanghai) Co., Ltd., Beijing Liaison Office
Unit 915
Bei Hai Wan Tai Bldg.
No. 6 Chaoyangmen Beidajie
Beijing, 100027, No. China
Tel: 86-10-85282100 Fax: 86-10-85282104

China - Chengdu
Microchip Technology Consulting (Shanghai) Co., Ltd., Chengdu Liaison Office
Rm. 2401, 24th Floor,
Ming Xing Financial Tower
No. 88 TIDU Street
Chengdu 610016, China
Tel: 86-28-86766200 Fax: 86-28-86766599

China - Fuzhou
Microchip Technology Consulting (Shanghai) Co., Ltd., Fuzhou Liaison Office
Unit 28F, World Trade Plaza
No. 71 Wusi Road
Fuzhou 350001, China
Tel: 86-591-7503506 Fax: 86-591-7503521

China - Shanghai
Microchip Technology Consulting (Shanghai) Co., Ltd.
Room 701, Bldg. B
Far East International Plaza
No. 317 Xian Xia Road
Shanghai, 200051
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

China - Shenzhen
Microchip Technology Consulting (Shanghai) Co., Ltd., Shenzhen Liaison Office
Rm. 1315, 13/F, Shenzhen Kerry Centre,
Renminnan Lu
Shenzhen 518001, China
Tel: 86-755-82350361 Fax: 86-755-82366086

China - Hong Kong SAR
Microchip Technology Hongkong Ltd.
Unit 901-6, Tower 2, Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2401-1200 Fax: 852-2401-3431

India
Microchip Technology Inc.
India Liaison Office
Divyasree Chambers
1 Floor, Wing A (A3/A4)
No. 11, O'Shaugnessey Road
Bangalore, 560 025, India
Tel: 91-80-2290061 Fax: 91-80-2290062

Japan
Microchip Technology Japan K.K.
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa, 222-0033, Japan
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

Korea
Microchip Technology Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea 135-882
Tel: 82-2-554-7200 Fax: 82-2-558-5934

Singapore
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore, 188980
Tel: 65-6334-8870 Fax: 65-6334-8850

Taiwan
Microchip Technology (Barbados) Inc.,
Taiwan Branch
11F-3, No. 207
Tung Hua North Road
Taipei, 105, Taiwan
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

EUROPE

Austria
Microchip Technology Austria GmbH
Durisolstrasse 2
A-4600 Wels
Austria
Tel: 43-7242-2244-399
Fax: 43-7242-2244-393

Denmark
Microchip Technology Nordic ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

France
Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

Germany
Microchip Technology GmbH
Steinheilstrasse 10
D-85737 Ismaning, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

Italy
Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-039-65791-1 Fax: 39-039-6899883

United Kingdom
Microchip Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5869 Fax: 44-118 921-5820

10/18/02