PIC12C5XX

EPROM Memory Programming Specification

This document includes the programming specifications for the following devices:

- PIC12C508
- PIC12C509

1.0 PROGRAMMING THE PIC12C5XX

The PIC12C5XX can be programmed using a serial method. Due to this serial programming, the PIC12C5XX can be programmed while in the user's system increasing design flexibility. This programming specification applies to PIC12C5XX devices in all packages.

1.1 Hardware Requirements

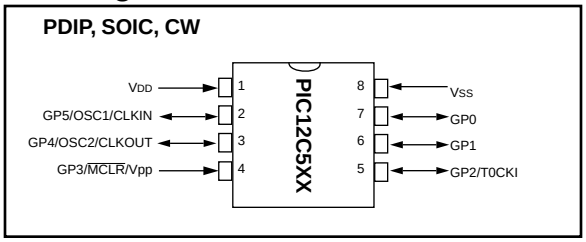
The PIC12C5XX requires two programmable power supplies, one for VDD (2.0V to 6.5V recommended) and one for VPP (12V to 14V). Both supplies should have a minimum resolution of 0.25V.

1.2 Programming Mode

The programming mode for the PIC12C5XX allows programming of user program memory, special locations used for ID, and the configuration word for the PIC12C5XX.

Pin Diagram

PDIP, SOIC, CW



PIN DESCRIPTIONS (DURING PROGRAMMING): PIC12C508/509

Pin Name	During Programming		
	Pin Name	Pin Type	Pin Description
GP1	CLOCK	I	Clock input
GP0	DATA	I/O	Data input/output
GP3/MCLR/VPP	VPP	P	Programming Power
VDD	VDD	P	Power Supply
VSS	VSS	P	Ground

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

2.0 PROGRAM MODE ENTRY

The program/verify test mode is entered by holding pins DB0 and DB1 low while raising $\overline{MCLR}$ pin from V_{IL} to V_{IHH}. Once in this test mode the user program memory and the test program memory can be accessed and programmed in a serial fashion. The first selected memory location is the fuses. **GP0 and GP1 are Schmitt trigger inputs in this mode.**

Incrementing the PC once (using the increment address command) selects location 0x000 of the regular program memory. Afterwards all other memory locations from 0x001-01FF (PIC12C508), 0x001-03FF (PIC12C509) can be addressed by incrementing the PC.

If the program counter has reached the last user program location and is incremented again, the on-chip special EPROM area will be addressed. (See Figure 2-2 to determine where the special EPROM area is located for the various PIC12C5XX devices).

2.1 Programming Method

The programming technique is described in the following section. It is designed to guarantee good programming margins. It does, however, require a variable power supply for V_{CC}.

2.1.1 PROGRAMMING METHOD DETAILS

Essentially, this technique includes the following steps:

1. Perform blank check at $V_{DD} = V_{DD}$min. Report failure. The device may not be properly erased.
2. Program location with pulses and verify after each pulse at $V_{DD} = V_{DDP}$:
   where $V_{DDP} = V_{DD}$ range required during programming (4.5V - 5.5V).

a) Programming condition:

   V_{PP} = 13.0V to 13.25V

   $V_{DD} = V_{DDP}$ = 4.5V to 5.5V

   V_{PP} must be $\geq V_{DD}$ + 7.25V to keep “programming mode” active.

b) Verify condition:

   $V_{DD} = V_{DDP}$

   $V_{PP} \geq V_{DD}$ + 7.5V but not to exceed 13.25V

   If location fails to program after “N” pulses, (suggested maximum program pulses of 25) then report error as a programming failure.

Note:	Device must be verified at minimum and maximum specified operating voltages as specified in the data sheet.

3. Once location passes “Step 2", apply 3X over-programming, i.e., apply three times the number of pulses that were required to program the location. This will guarantee a solid programming margin. The overprogramming should be made “software programmable” for easy updates.
4. Program all locations.
5. Verify all locations (using speed verify mode) at $V_{DD} = V_{DD}$min
6. Verify all locations at $V_{DD} = V_{DD}$max

   V_{DD}min is the minimum operating voltage spec. for the part. V_{DD}max is the maximum operating voltage spec. for the part.

2.1.2 SYSTEM REQUIREMENTS

Clearly, to implement this technique, the most stringent requirements will be that of the power supplies:

V_{PP}: V_{PP} can be a fixed 13.0V to 13.25V supply. It must not exceed 14.0V to avoid damage to the pin and should be current limited to approximately 100mA.

V_{DD}: 2.0V to 6.5V with 0.25V granularity. Since this method calls for verification at different V_{DD} values, a programmable V_{DD} power supply is needed.

Current Requirement: 40mA maximum

Microchip may release devices in the future with different V_{DD} ranges which make it necessary to have a programmable V_{DD}.

It is important to verify an EPROM at the voltages specified in this method to remain consistent with Microchip's test screening. For example, a PIC12C5XX specified for 4.5V to 5.5V should be tested for proper programming from 4.5V to 5.5V.

Note:	Any programmer not meeting the programmable V_{DD} requirement and the verify at V_{DD}max and V_{DD}min requirement may only be classified as “prototype” or “development” programmer but not a production programmer.

2.1.3 SOFTWARE REQUIREMENTS

Certain parameters should be programmable (and therefore easily modified) for easy upgrade.

a) Pulse width

b) Maximum number of pulses, present limit 25.

c) Number of over-programming pulses: should be = (A • N) + B, where N = number of pulses required in regular programming. In our current algorithm A = 3, B = 0.

2.2 Programming Pulse Width

Program Memory Cells: When programming one word of EPROM, a programming pulse width (T_{PW}) of 100µs is recommended.

The maximum number of programming attempts should be limited to 25 per word.

After the first successful verify, the same location should be over-programmed with 3X over-programming.

Configuration Word: The configuration word for oscillator selection, WDT (watchdog timer) disable and code protection, and MCLR enable, requires a programming pulse width (T_{PWF}) of 10ms. A series of 100µs pulses is preferred over a single 10ms pulse.

FIGURE 2-1: PROGRAMMING METHOD FLOWCHART

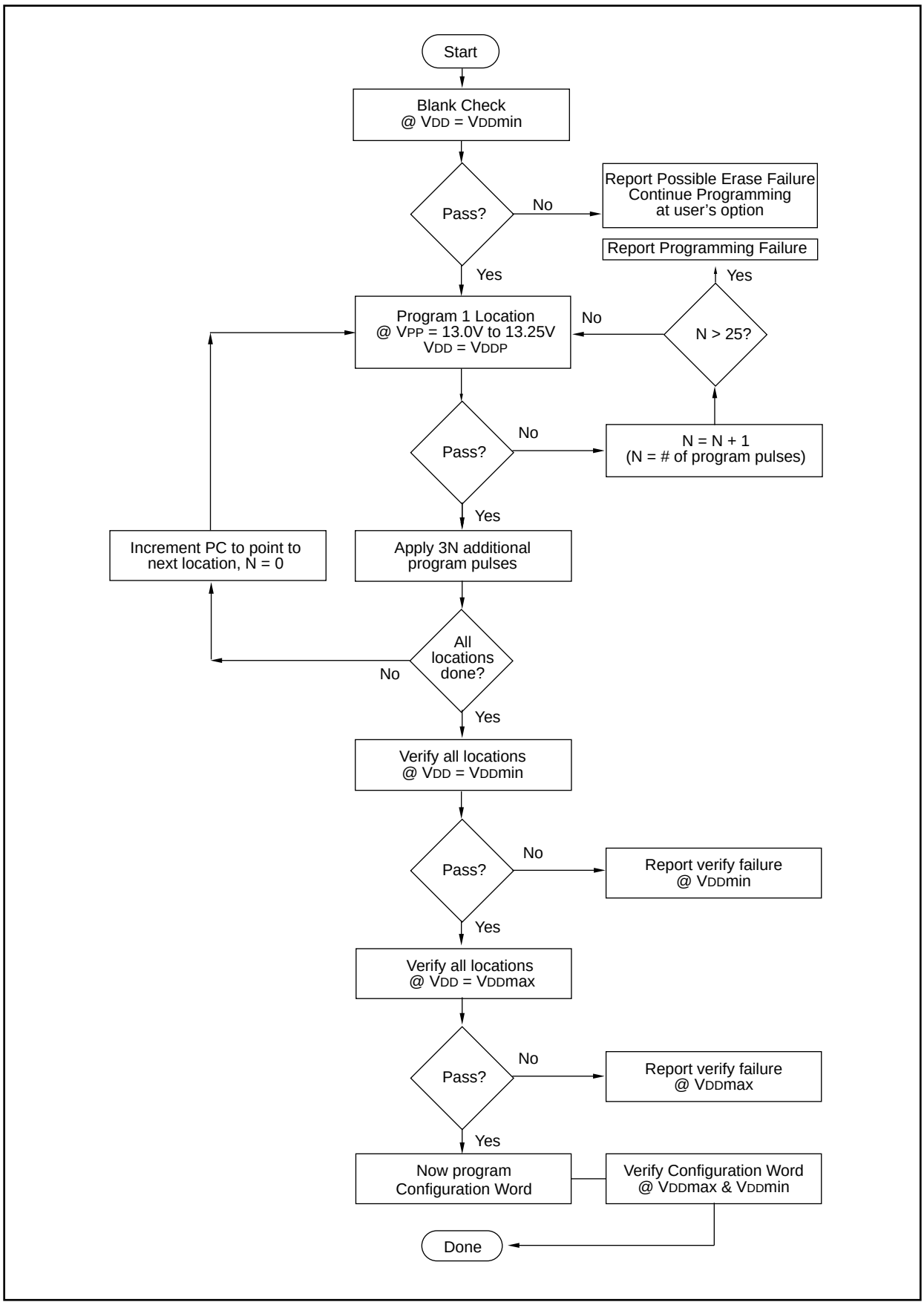
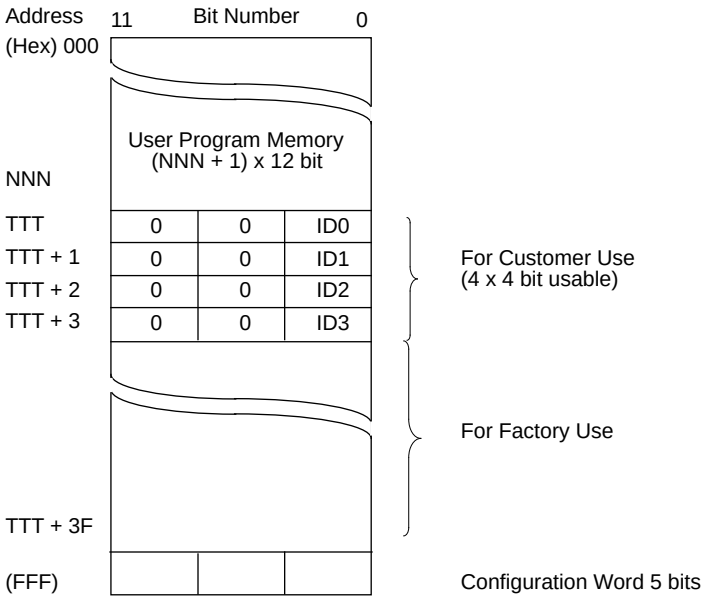


FIGURE 2-2: PIC12C5XX SERIES PROGRAM MEMORY MAP IN PROGRAM/VERIFY MODE



NNN Highest normal EPROM memory address. NNN = 0x1FF for PIC12C508.
NNN = 0x3FF for PIC12C509 .
Note that some versions will have an oscillator calibration value programmed at NNN
TTT Start address of special EPROM area and ID Locations.

2.3 Special Memory Locations

The highest address of program memory space is reserved for the internal RC oscillator calibration value. This location should not be overwritten except when this location is blank, and it should be verified, when programmed, that it is a `MOVLW XX` instruction.

The ID Locations area is only enabled if the device is in a test or programming/verify mode. Thus, in normal operation mode only the memory location 0x000 to 0xNNN will be accessed and the Program Counter will just roll over from address 0xNNN to 0x000 when incremented.

The configuration word can only be accessed immediately after $\overline{MCLR}$ going from V_{IL} to V_{HH}. The Program Counter will be set to all '1's upon $\overline{MCLR} = V_{IL}$. Thus, it has the value “0xFFF” when accessing the configuration EPROM. Incrementing the Program Counter once causes the Program Counter to roll over to all '0's. Incrementing the Program Counter 4K times after reset ($\overline{MCLR} = V_{IL}$) does not allow access to the configuration EPROM.

2.3.1 CUSTOMER ID CODE LOCATIONS

Per definition, the first four words (address TTT to TTT + 3) are reserved for customer use. It is recommended that the customer use only the four lower order bits (bits 0 through 3) of each word and filling the eight higher order bits with '0's.

A user may want to store an identification code (ID) in the ID locations and still be able to read this code after the code protection bit was programmed. This is possible if the ID code is only four bits long per memory location, is located in the least significant nibble boundary of the 12-bit word, and the remaining eight bits are all '0's.

EXAMPLE 2-1:CUSTOMER CODE 0xD1E2

The Customer ID code “0xD1E2” should be stored in the ID locations 200-203 like this:

```
200:   0000 0000 1101
201:   0000 0000 0001
202:   0000 0000 1110
203:   0000 0000 0010
```

Reading these four memory locations, even with the code protection bit programmed would still output on Port A the bit sequence “1101”, “0001”, “1110”, “0010” which is “0xD1E2”.

2.4 Program/Verify Mode

The program/verify mode is entered by holding pins GP1 and GP0 low while raising $\overline{MCLR}$ pin from V_{IL} to V_{IHH} (high voltage). 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 that is accessed is the user program memory. GP1 is a Schmitt Trigger input in this mode.

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 V_{IL}). This means that all I/O are in the reset state (High impedance inputs).

Note:	The $\overline{MCLR}$ pin should be raised from V_{IL} to V_{IHH} within 9 ms of V_{DD} rise. This is to ensure that the device does not have the PC incremented while in valid operation range.

2.4.1 PROGRAM/VERIFY OPERATION

The GP1 pin is used as a clock input pin, and the GP0 pin is used for entering command bits and data input/output during serial operation. To input a command, the clock pin (GP1) 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 pin GP0 is required to have a minimum setup and hold time (see AC/DC specs) 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 1µs 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. Data is also input and output LSB first. Therefore, during a read operation the LSB will be transmitted onto pin GP0 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 1µs delay is also specified between consecutive commands.

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

The commands that are available are listed in Table 2-1.

TABLE 2-1: COMMAND MAPPING

Command	Mapping (MSB ... LSB)						Data
Load Data	0	0	0	0	1	0	0, data(14), 0
Read Data	0	0	0	1	0	0	0, data(14), 0
Increment Address	0	0	0	1	1	0	
Begin programming	0	0	1	0	0	0	
End Programming	0	0	1	1	1	0	

Note: The clock must be disabled during in-circuit programming.

2.4.1.1 LOAD DATA

After receiving this command, the chip will load in a 14-bit “data word” when 16 cycles are applied, as described previously. Because this is a 12 bit core, the two msb’s of the data word are ignored. A timing diagram for the load data command is shown in Figure 5-1.

2.4.1.2 READ DATA

After receiving this command, the chip will transmit data bits out of the memory currently accessed starting with the second rising edge of the clock input. The GP0 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 5-2.

2.4.1.3 INCREMENT ADDRESS

The PC is incremented when this command is received. A timing diagram of this command is shown in Figure 5-3.

2.4.1.4 BEGIN PROGRAMMING

A load data command must be given before every begin programming command. Programming of the appropriate memory (test program memory or user program memory) will begin after this command is received and decoded. Programming should be performed with a series of 100µs programming pulses. A programming pulse is defined as the time between the begin programming command and the end programming command.

2.4.1.5 END PROGRAMMING

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

2.5 Programming Algorithm Requires Variable VDD

The PIC12C5XX uses an intelligent algorithm. The algorithm calls for program verification at VDDmin as well as VDDmax. Verification at VDDmin guarantees good “erase margin”. Verification at VDDmax guarantees good “program margin”.

The actual programming must be done with VDD in the VDDP range (4.75 - 5.25V).

VDDP = VCC range required during programming.

VDD min. = minimum operating VDD spec for the part.

VDDmax = maximum operating VDD spec for the part.

Programmers must verify the PIC12C5XX at its specified VDDmax and VDDmin levels. Since Microchip may introduce future versions of the PIC12C5XX with a broader VDD range, it is best that these levels are user selectable (defaults are ok).

Note:	Any programmer not meeting these requirements may only be classified as “prototype” or “development” programmer but not a “production” quality programmer.

3.0 CONFIGURATION WORD

The PIC12C5XX family members have several configuration bits. These bits can be programmed (reads '0') or left unprogrammed (reads '1') to select various device configurations. Figure 3-1 provides an overview of configuration bits.

FIGURE 3-1: CONFIGURATION WORD BIT MAP

Bit Number:	11	10	9	8	7	6	5	4	3	2	1	0
PIC12C5XX	—	—	—	—	—	—	—	MCLRE	CP	WDTE	FOSC1	FOSC0

bit 11-5:**Reserved**, '–' write as '0' for PIC12C5XX

bit 4: **MCLRE**, Master Clear pin Enable Bit
0 = MCLR internally connected to Vdd
1 = MCLR pin enabled

bit 3: **CP**, Code Protect Enable Bit
1 = Code Memory Unprotected
0 = Code Memory Protected

bit 2: **WDTE**, WDT Enable Bit
1 = WDT enabled
0 = WDT disabled

bit 1-0: **FOSC<1:0>**, Oscillator Selection Bit
11: ExtRC oscillator
10: IntRC oscillator
01: XT oscillator
00: LP oscillator

4.0 CODE PROTECTION

The program code written into the EPROM can be protected by writing to the CP0 bit of the configuration word.

In PIC12C5XX it is still possible to program and read locations 0x000 through 0x03F, after code protection. Once code protection is enabled, all protected segments read '0's (or “garbage values”) and are prevented from further programming. All unprotected segments, including ID locations and configuration word, read normally. These locations can be programmed.

4.1 Embedding Configuration Word and ID Information in the 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.

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

TABLE 4-1: CODE PROTECTION

PIC12C508

To code protect:

- (CP enable pattern: XXXXXXXX0XXX)

Program Memory Segment	R/W in Protected Mode	R/W in Unprotected Mode
Configuration Word (0xFFF)	Read enabled, Write Enabled	Read enabled, Write Enabled
[0x00:0x3F]	Read enabled, Write Enabled	Read enabled, Write Enabled
[0x40:0x1FF]	Read disabled (all 0’s), Write Disabled	Read enabled, Write Enabled
ID Locations (0x200 : 0x203)	Read enabled, Write Enabled	Read enabled, Write Enabled

PIC12C509

To code protect:

- (CP enable pattern: XXXXXXXX0XXX))

Program Memory Segment	R/W in Protected Mode	R/W in Unprotected Mode
Configuration Word (0xFFF)	Read enabled, Write Enabled	Read enabled, Write Enabled
[0x00:0x3F]	Read enabled, Write Enabled	Read enabled, Write Enabled
[0x40:0x3FF]	Read disabled (all 0’s), Write Disabled	Read enabled, Write Enabled
ID Locations (0x400 : 0x403)	Read enabled, Write Enabled	Read enabled, Write Enabled

4.2 Checksum

4.2.1 CHECKSUM CALCULATIONS

Checksum is calculated by reading the contents of the PIC12C5XX memory locations and adding up the opcodes up to the maximum user addressable location, (not including the last location which is reserved for the oscillator calibration value) e.g., 0x1FE for the PIC12C508. 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 PIC12C50X family is shown in Table 4-2.

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 is 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 4-2: CHECKSUM COMPUTATION

Device	Code Protect	Checksum*	Blank Value	0x723 at 0 and max address
PIC12C508	OFF	SUM[0x000:0x1FE] + CFGW & 0x001F	EE20	DC68
	ON	SUM[0x000:0x03F] + CFGW & 0x001F	EDF7	D363
PIC12C509	OFF	SUM[0x000:0x3FE] + CFGW & 0x001F	EC20	DA68
	ON	SUM[0x000:0x03F] + CFGW & 0x001F	EBF7	D163

Legend: CFGW = Configuration Word
SUM[a:b] = [Sum of locations a through b inclusive]
SUM_ID = ID locations masked by 0xF then made into a 16-bit value with ID0 as the most significant nibble.
For example,
ID0 = 0x12, ID1 = 0x37, ID2 = 0x4, ID3 = 0x26, then SUM_ID = 0x2746.
*Checksum = [Sum of all the individual expressions] **MODULO** [0xFFFF]
+ = Addition
& = Bitwise AND

5.0 PROGRAM/VERIFY MODE ELECTRICAL CHARACTERISTICS

TABLE 5-1: AC/DC CHARACTERISTICS TIMING REQUIREMENTS FOR PROGRAM/VERIFY TEST MODE

Standard Operating Conditions Operating Temperature: +10°C ≤ TA ≤ +40°C, unless otherwise stated, (20°C recommended) Operating Voltage: 4.5V ≤ VDD ≤ 5.5V, unless otherwise stated.							
Parameter No.	**Sym.**	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
General							
PD1	VDDP	Supply voltage during programming	4.75	5.0	5.25	V	
PD2	IDDP	Supply current (from VDD) during programming			20	mA	
PD3	VDDV	Supply voltage during verify	VDDmin		VDDmax	V	Note 1
PD4	VIHH1	Voltage on $\overline{MCLR}$/VPP during programming	12.75		13.25	V	Note 2
PD5	VIHH2	Voltage on $\overline{MCLR}$/VPP during verify	VDD + 4.0		13.5		
PD6	IPP	Programming supply current (from VPP)			50	mA	
PD9	VIH1	(GP1, GP0) input high level	0.8 VDD			V	Schmitt Trigger input
PD8	VIL1	(GP1, GP0) input low level	0.2 VDD			V	Schmitt Trigger input
Serial Program Verify							
P1	TR	$\overline{MCLR}$/VPP rise time (VSS to VHH) for test mode entry			8.0	µs	
P2	Tf	$\overline{MCLR}$ Fall time			8.0	µs	
P3	Tset1	Data in setup time before clock ↓	100			ns	
P4	Thld1	Data in hold time after clock ↓	100			ns	
P5	Tdly1	Data input not driven to next clock input (delay required between command/data or command/command)	1.0			µs	
P6	Tdly2	Delay between clock ↓ to clock ↑ of next command or data	1.0			µs	
P7	Tdly3	Clock ↑ to date out valid (during read data)	200			ns	
P8	Thld0	Hold time after $\overline{MCLR}$ ↑	2			µs	

Note 1: Program must be verified at the minimum and maximum VDD limits for the part.
Note 2: VIHH must be greater than VDD + 4.5V to stay in programming/verify mode.

FIGURE 5-1: LOAD DATA COMMAND (PROGRAM/VERIFY)

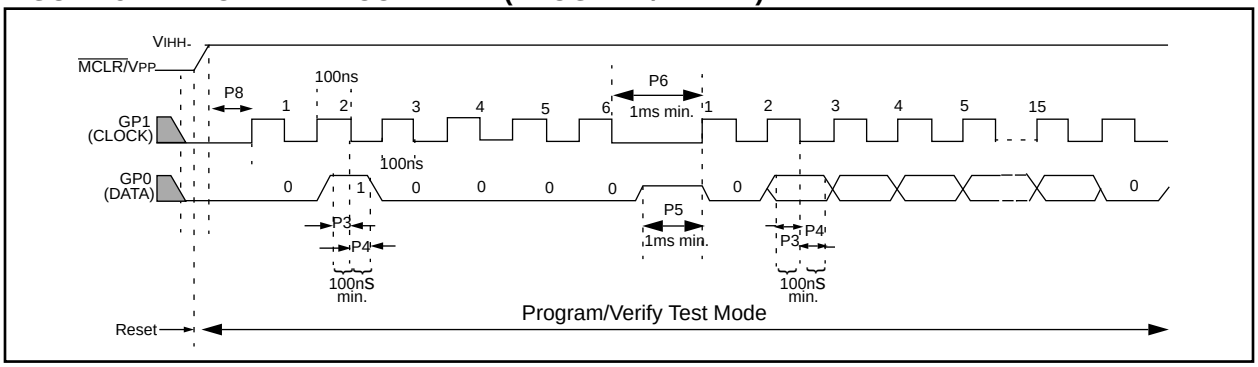


FIGURE 5-2: READ DATA COMMAND (PROGRAM/VERIFY)

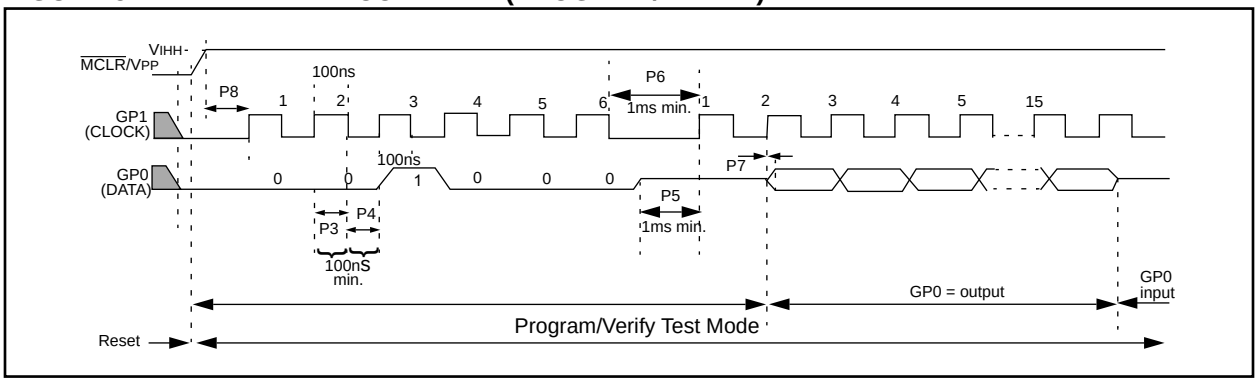
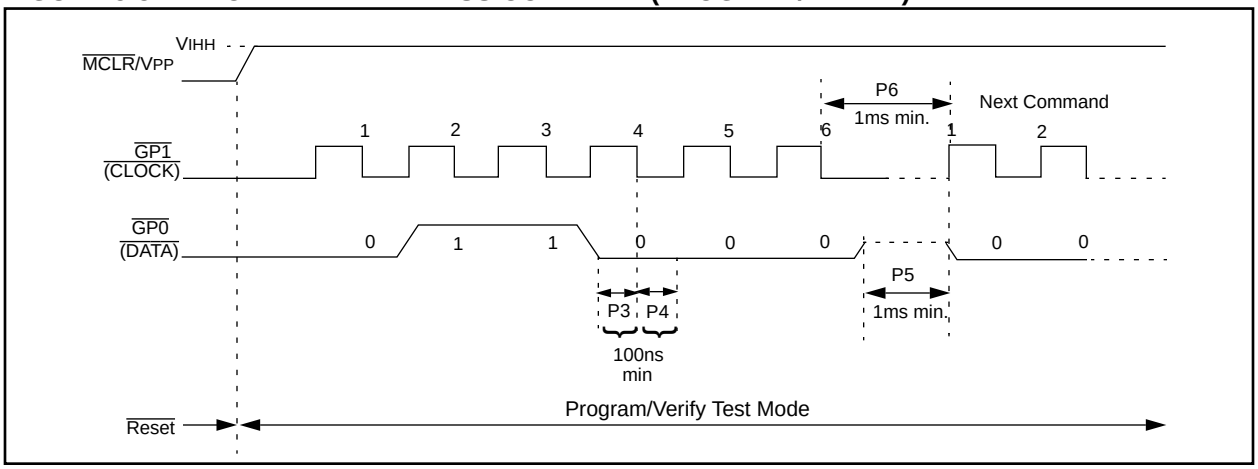


FIGURE 5-3: INCREMENT ADDRESS COMMAND (PROGRAM/VERIFY)



WORLDWIDE SALES & SERVICE

AMERICAS

Corporate Office
Microchip Technology Inc.
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 602 786-7200 Fax: 602 786-7277
Technical Support: 602 786-7627
Web: http://www.microchip.com

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

Boston
Microchip Technology Inc.
5 Mount Royal Avenue
Marlborough, MA 01752
Tel: 508 480-9990Fax: 508 480-8575

Chicago
Microchip Technology Inc.
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 708 285-0071 Fax: 708 285-0075

Dallas
Microchip Technology Inc.
14651 Dallas Parkway, Suite 816
Dallas, TX 75240-8809
Tel: 214 991-7177 Fax: 214 991-8588

Dayton
Microchip Technology Inc.
Suite 150
Two Prestige Place
Miamisburg, OH 45342
Tel: 513 291-1654 Fax: 513 291-9175

Los Angeles
Microchip Technology Inc.
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 714 263-1888 Fax: 714 263-1338

New York
Microchip Technmgy Inc.
150 Motor Parkway, Suite 416
Hauppauge, NY 11788
Tel: 516 273-5305 Fax: 516 273-5335

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

Toronto
Microchip Technology Inc.
5925 Airport Road, Suite 200
Mississauga, Ontario L4V 1W1, Canada
Tel: 905 405-6279Fax: 905 405-6253

ASIA/PACIFIC

Hong Kong
Microchip Technology
RM 3801B, Tower Two
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T. Hong Kong
Tel: 852 2 401 1200 Fax: 852 2 401 3431

India
Microchip Technology
No. 6, Legacy, Convent Road
Bangalore 560 025 India
Tel: 91 80 526 3148 Fax: 91 80 559 9840

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

Singapore
Microchip Technology
200 Middle Road
#10-03 Prime Centre
Singapore 188980
Tel: 65 334 8870 Fax: 65 334 8850

Shanghai
Microchip Technology
Unit 406 of Shanghai Golden Bridge Bldg.
2077 Yan'an Road West, Hongiao District
Shanghai, Peoples Republic of China
Tel: 86 21 6275 5700
Fax: 011 86 21 6275 5060

Taiwan
Microchip Technology
10F-1C 207
Tung Hua North Road
Taipei, Taiwan, ROC
Tel: 886 2 717 7175 Fax: 886 2 545 0139

EUROPE

United Kingdom
Arizona Microchip Technology Ltd.
Unit 6, The Courtyard
Meadow Bank, Furlong Road
Bourne End, Buckinghamshire SL8 5AJ
Tel: 44 1628 850303 Fax: 44 1628 850178

France
Arizona Microchip Technology SARL
Zone Industrielle de la Bonde
2 Rue du Buisson aux Fraises
91300 Massy - France
Tel: 33 1 69 53 63 20 Fax: 33 1 69 30 90 79

Germany
Arizona Microchip Technology GmbH
Gustav-Heinemann-Ring 125
D-81739 Muenchen, Germany
Tel: 49 89 627 144 0 Fax: 49 89 627 144 44

Italy
Arizona Microchip Technology SRL
Centro Direzionale Colleone Pas Taurus 1
Viale Colleoni 1
20041 Agrate Brianza
Milan Italy
Tel: 39 39 6899939 Fax: 39 39 689 9883

JAPAN

Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shin Yokohama
Kohoku-Ku, Yokohama
Kanagawa 222 Japan
Tel: 81 45 471 6166 Fax: 81 45 471 6122

8/13/96

MICROCHIP