This document includes the programming specifications for the following devices:

- PIC16F1946, PIC16F1947, PIC16LF1902
- PIC16LF1903, PIC16LF1904, PIC16LF1906

1.0 OVERVIEW

The device can be programmed using either the high-voltage In-Circuit Serial Programming™ (ICSP™) method or the low-voltage ICSP method.

1.1 Hardware Requirements

1.1.1 HIGH-VOLTAGE ICSP PROGRAMMING

In High-Voltage ICSP mode, the device requires two programmable power supplies: one for VDD and one for the MCLR/VPP pin.

1.1.2 LOW-VOLTAGE ICSP PROGRAMMING

In Low-Voltage ICSP mode, the PIC16F193X/194X and PIC16LF193X/194X/190X devices can be programmed using a single VDD source in the operating range. The MCLR/VPP pin does not have to be brought to a different voltage, but can instead be left at the normal operating voltage.

1.1.2.1 Single-Supply ICSP Programming

The LVP bit in Configuration Word 2 enables single-supply (low-voltage) ICSP programming. The LVP bit defaults to a ‘1’ (enabled) from the factory. The LVP bit may only be programmed to ‘0’ by entering the High-Voltage ICSP mode, where MCLR/VPP pin is raised to VIHH. Once the LVP bit is programmed to a ‘0’, only the High-Voltage ICSP mode is available and only the High-Voltage ICSP mode can be used to program the device.

Note 1: The High-Voltage ICSP mode is always available, regardless of the state of the LVP bit, by applying VIHH to the MCLR/VPP pin.

2: While in Low-Voltage ICSP mode, MCLR is always enabled, regardless of the MCLRE bit, and the port pin can no longer be used as a general purpose input.

1.2 Pin Utilization

Five pins are needed for ICSP programming. The pins are listed in Table 1-1 and Table 1-2.

### TABLE 1-1: PIN DESCRIPTIONS DURING PROGRAMMING FOR PIC16F193X/LF193X/LF190X

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>During Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Function</td>
</tr>
<tr>
<td>RB6</td>
<td>ICSPCLK</td>
</tr>
<tr>
<td>RB7</td>
<td>ICSPDAT</td>
</tr>
<tr>
<td>RE3/MCLR/VPP</td>
<td>Program/Verify mode</td>
</tr>
<tr>
<td>VDD</td>
<td>VDD</td>
</tr>
<tr>
<td>VSS</td>
<td>VSS</td>
</tr>
</tbody>
</table>

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

Note 1: The programming high voltage is internally generated. To activate the Program/Verify mode, high voltage needs to be applied to MCLR input. Since the MCLR is used for a level source, MCLR does not draw any significant current.
### TABLE 1-2: PIN DESCRIPTIONS DURING PROGRAMMING FOR PIC16F194X/LF194X

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>Function</th>
<th>Pin Type</th>
<th>Pin Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RB6</td>
<td>ICSPCLK</td>
<td>I</td>
<td>Clock Input – Schmitt Trigger Input</td>
</tr>
<tr>
<td>RB7</td>
<td>ICSPDAT</td>
<td>I/O</td>
<td>Data Input/Output – Schmitt Trigger Input</td>
</tr>
<tr>
<td>RG5/MCLR/VPP</td>
<td>Program/Verify mode</td>
<td>P(1)</td>
<td>Program Mode Select/Programming Power Supply</td>
</tr>
<tr>
<td>VDD</td>
<td>VDD</td>
<td>P</td>
<td>Power Supply</td>
</tr>
<tr>
<td>VSS</td>
<td>VSS</td>
<td>P</td>
<td>Ground</td>
</tr>
</tbody>
</table>

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

**Note 1:** The programming high voltage is internally generated. To activate the Program/Verify mode, high voltage needs to be applied to MCLR input. Since the MCLR is used for a level source, MCLR does not draw any significant current.
Table of Contents

1.0 Overview ..................................................................................................................................................................  1
2.0 Device Pinouts .........................................................................................................................................................  4
3.0 Memory Map ............................................................................................................................................................  8
4.0 Program/Verify Mode ............................................................................................................................................. 18
5.0 Programming Algorithms ........................................................................................................................................ 27
6.0 Code Protection ........................................................................................................................................................ 32
7.0 Hex File Usage ...................................................................................................................................................... 32
8.0 Electrical Specifications ......................................................................................................................................... 39
2.0 DEVICE PINOUTS

The pin diagrams for the PIC16F193X/LF193X/PIC16F194X/LF194X/PIC16LF190X family are shown in Figure 2-1 through Figure 2-6. The pins that are required for programming are listed in Table 1-1 and shown in bold lettering in the pin diagrams.


40-pin PDIP

VPP/MCLR/RE3  |  RB7/ICSPDAT  
1             |  40           
RA0           |  36           
RA1           |  38           
RA2           |  37           
RA3           |  36           
RA4           |  35           
RA5           |  36           
RE0           |  33           
RE1           |  32           
RE2           |  31           
VDD           |  32           
VSS           |  31           
RA7           |  28           
RA6           |  27           
RC0           |  26           
RC1           |  25           
RC2           |  24           
RC3           |  23           
RD0           |  22           
RD1           |  21           


44-pin QFN

RC7  |  RA6  
41   |  33  
RD4  |  RA7  
32   |  31  
RD5  |  VSS  
3   |  2  
RD6  |  VSS  
4    |  3   
RD7  |  NC   
5    |  29  
VSS  |  VDD  
6    |  28  
VDD  |  VDD  
7    |  27  
VDD  |  RE2  
8    |  26  
RE1  |  RE1  
9    |  25  
RE0  |  RE0  
10   |  24  
RA5  |  RA5  
11   |  23  
RA4  |  RA4  
12   |  22  
RA3  |  RA3  
13   |  21  
RA2  |  RA2  
14   |  20  
RA1  |  RA1  
15   |  19  
RB0  |  RB0  
16   |  18  
RB1  |  RB1  
17   |  17  
RB2  |  RB2  
18   |  16  
RB3  |  RB3  
19   |  15  
RB4  |  RB4  
20   |  14  
RB5  |  RB5  
21   |  13  
RB6  |  RB6  
22   |  12  
RB7  |  RB7  
23   |  11  
ICSPDAT/RB7 
24   |  10  
ICSPCLK/RB6 
25   |  9   
ICSPCLK/RB6 
26   |  8   
ICSPDAT/RB7 
27   |  7   
VPP/MCLR/RE3 
28   |  6   
VPP/MCLR/RE3 
29   |  5   
VPP/MCLR/RE3 
30   |  4   
VPP/MCLR/RE3 
31   |  3   
VPP/MCLR/RE3 
32   |  2   
VPP/MCLR/RE3 
33   |  1   
VPP/MCLR/RE3 
34   |   1  
VPP/MCLR/RE3 
35   |   2  
VPP/MCLR/RE3 
36   |   3  
VPP/MCLR/RE3 
37   |   4  
VPP/MCLR/RE3 
38   |   5  
VPP/MCLR/RE3 
39   |   6  
VPP/MCLR/RE3 
40   |   7  
VPP/MCLR/RE3 
41   |   8  
VPP/MCLR/RE3 
42   |   9  
VPP/MCLR/RE3 
43   |  10  
VPP/MCLR/RE3 
44   |  11  
VPP/MCLR/RE3


Note: QFN package orientation is the same. No leads are present on the QFN package.
3.0 MEMORY MAP

The memory is broken into two sections: program memory and configuration memory. Only the size of the program memory changes between devices, the configuration memory remains the same.

FIGURE 3-1: PIC16LF1902 PROGRAM MEMORY MAPPING
FIGURE 3-2: PIC16F1933/PIC16LF1933, PIC16F1934/PIC16LF1934, PIC16LF1903/
PIC16LF1904 PROGRAM MEMORY MAPPING

- 8000h-80FFh: User ID Location
- 8000h-80FFh: Implemented
- 0000h-0FFFh: Maps to 0-0FFF
- 8000h: Device ID
- 8007h: Configuration Word 1
- 8008h: Configuration Word 2
- 8009h: Calibration Word 1
- 800Ah: Calibration Word 2
- 800Bh-81FFh: Reserved
- 7FFFh-8000h: Implemented
- 8200h: Maps to 8000-81FF
- FFFFh: Program Memory
- 0000h-0FFFh: Maps to 0-0FFF
- 8000-81FF: User ID Location
- 8000h-80FFh: Implemented
PIC16LF1906/PIC16LF1907 PROGRAM MEMORY MAPPING

0000h
1FFFh
8000h
7FFFh
8200h
FFFFh

8 KW
Implemented
Maps to
0-1FFF
Program Memory
Maps to
8000-81FF
Configuration Memory

8000h
User ID Location
8001h
User ID Location
8002h
User ID Location
8003h
User ID Location
8004h
Reserved
8005h
Reserved
8006h
Device ID
8007h
Configuration Word 1
8008h
Configuration Word 2
8009h
Calibration Word 1
800Ah
Calibration Word 2
800Bh-81FFh
Reserved

![Diagram of Program Memory Mapping]

- **User ID Location**: From 8000h to 8007h
- **Device ID**: 8006h
- **Configuration Word 1**: 8007h
- **Configuration Word 2**: 8008h
- **Calibration Word 1**: 8009h
- **Calibration Word 2**: 800Ah
- **Reserved**: 800Bh-81FFh

Program Memory implemented from 0000h to 3FFFh

Configuration Memory maps to 0x8000 to 0x81FFh

16KW maps to 0x8200 to FFFFh
3.1 User ID Location

A user may store identification information (user ID) in four designated locations. The user ID locations are mapped to 8000h-8003h. Each location is 14 bits in length. Code protection has no effect on these memory locations. Each location may be read with code protection enabled or disabled.

Note: MPLAB® IDE only displays the 7 Least Significant bits (LSb) of each user ID location, the upper bits are not read. It is recommended that only the 7 LSb’s be used if MPLAB IDE is the primary tool used to read these addresses.

3.2 Device ID

The device ID word is located at 8006h. This location is read-only and cannot be erased or modified.

REGISTER 3-1: DEVICEID: DEVICE ID REGISTER(1)

<table>
<thead>
<tr>
<th>Bit Location</th>
<th>Bit Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>bit 13-5</td>
<td>DEV&lt;8:0&gt;: Device ID bits</td>
</tr>
<tr>
<td></td>
<td>These bits are used to identify the part number.</td>
</tr>
<tr>
<td>bit 4-0</td>
<td>REV&lt;4:0&gt;: Revision ID bits</td>
</tr>
<tr>
<td></td>
<td>These bits are used to identify the revision.</td>
</tr>
</tbody>
</table>

Note 1: This location cannot be written.
TABLE 3-1: DEVICE ID VALUES

<table>
<thead>
<tr>
<th>DEVICE</th>
<th>DEV</th>
<th>REV</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIC16F1933</td>
<td>10 0011 000</td>
<td>x</td>
</tr>
<tr>
<td>PIC16F1934</td>
<td>10 0011 010</td>
<td>x</td>
</tr>
<tr>
<td>PIC16F1936</td>
<td>10 0011 011</td>
<td>x</td>
</tr>
<tr>
<td>PIC16F1937</td>
<td>10 0011 100</td>
<td>x</td>
</tr>
<tr>
<td>PIC16F1938</td>
<td>10 0011 101</td>
<td>x</td>
</tr>
<tr>
<td>PIC16F1939</td>
<td>10 0011 110</td>
<td>x</td>
</tr>
<tr>
<td>PIC16F1946</td>
<td>10 0101 000</td>
<td>x</td>
</tr>
<tr>
<td>PIC16F1947</td>
<td>10 0101 001</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1933</td>
<td>10 0100 000</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1934</td>
<td>10 0100 010</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1936</td>
<td>10 0100 011</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1937</td>
<td>10 0100 100</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1938</td>
<td>10 0100 101</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1939</td>
<td>10 0100 110</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1946</td>
<td>10 0101 100</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1947</td>
<td>10 0101 101</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1902</td>
<td>10 1100 001</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1903</td>
<td>10 1100 000</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1904</td>
<td>10 1100 100</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1906</td>
<td>10 1100 011</td>
<td>x</td>
</tr>
<tr>
<td>PIC16LF1907</td>
<td>10 1100 010</td>
<td>x</td>
</tr>
</tbody>
</table>

3.3 Configuration Words
The device has two Configuration Words, Configuration Word 1 (8007h) and Configuration Word 2 (8008h). The individual bits within these Configuration Words are used to enable or disable device functions such as the Brown-out Reset, code protection and Power-up Timer.

3.4 Calibration Words
The internal calibration values are factory calibrated and stored in Calibration Words 1 and 2 (8009h and 800Ah).

The Calibration Words do not participate in erase operations. The device can be erased without affecting the Calibration Words.
### REGISTER 3-2: CONFIGURATION WORD 1

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| 13  | FCMEN (Fail-Safe Clock Monitor Enable) | 1 = Fail-Safe Clock Monitor is enabled  
0 = Fail-Safe Clock Monitor is disabled |
| 12  | IESO (Internal External Switchover)   | 1 = Internal/External Switchover mode is enabled  
0 = Internal/External Switchover mode is disabled |
| 11  | CLKOUTEN (Clock Out Enable)           | 1 = CLKOUT function is enabled. I/O or oscillator function on RA6/CLKOUT  
0 = CLKOUT function is enabled on RA6/CLKOUT |
| 10-9| BOREN<1:0> (Brown-out Reset Enable)   | 11 = BOR enabled  
10 = BOR enabled during operation and disabled in Sleep  
01 = BOR controlled by SBOREN bit of the PCON register  
00 = BOR disabled |
| 8   | CPD (Data Code Protection)            | 1 = Data memory code protection is disabled  
0 = Data memory code protection is enabled |
| 7   | CP (Code Protection)                  | 1 = Program memory code protection is disabled  
0 = Program memory code protection is enabled |
| 6   | MCLRE (MCLR/VPP Pin Function Select)  | If LVP bit = 1:  
This bit is ignored.  
If LVP bit = 0:  
1 = MCLR/VPP pin function is MCLR; Weak pull-up enabled.  
0 = MCLR/VPP pin function is digital input; MCLR internally disabled; Weak pull-up under control of port pin's WPU control bit. |
| 5   | PWRTE (Power-up Timer Enable)         | 1 = PWRT disabled  
0 = PWRT enabled |

**Legend:**  
R = Readable bit  
W = Writable bit  
U = Unimplemented bit, read as '0'  
'0' = Bit is cleared  
‘1’ = Bit is set  
x = Bit is unknown

- **Note 1:** Enabling Brown-out Reset does not automatically enable Power-up Timer.
- **Note 2:** The entire data EEPROM will be erased when the code protection is turned off during an erase.
- **Note 3:** The entire program memory will be erased when the code protection is turned off.
- **Note 4:** Unimplemented on PIC16LF190X devices. This bit reads as '1'.
- **Note 5:** For PIC16LF190X only.
REGISTER 3-2: CONFIGURATION WORD 1 (CONTINUED)

bit 4-3  WDTE<1:0>: Watchdog Timer Enable bit
11 = WDT enabled
10 = WDT enabled while running and disabled in Sleep
01 = WDT controlled by the SWDTEN bit in the WDTCON register
00 = WDT disabled

bit 2-0  FOSC<2:0>: Oscillator Selection bits
111 = ECH: External Clock, High-Power mode: CLKIN on RA7/OSC1/CLKIN
110 = ECM: External Clock, Medium-Power mode: CLKIN on RA7/OSC1/CLKIN
101 = ECL: External Clock, Low-Power mode: CLKIN on RA7/OSC1/CLKIN
100 = INTOSC oscillator: I/O function on RA7/OSC1/CLKIN
011 = EXTRC oscillator: RC function on RA7/OSC1/CLKIN
010 = HS oscillator: High-speed crystal/resonator on RA6/OSC2/CLKOUT pin and RA7/OSC1/CLKIN
001 = XT oscillator: Crystal/resonator on RA6/OSC2/CLKOUT pin and RA7/OSC1/CLKIN
000 = LP oscillator: Low-power crystal on RA6/OSC2/CLKOUT pin and RA7/OSC1/CLKIN

bit 2<5>  Unimplemented: Read as ‘1’

bit 1-0<5>  FOSC<1:0>: Oscillator Selection bits
00 = INTOSC Oscillator: I/O function on RA7/CLKIN
01 = ECL: External Clock, Low-Power mode: CLKIN on RA7/OSC1/CLKIN
10 = ECM: External Clock, Medium-Power mode: CLKIN on RA7/OSC1/CLKIN
11 = ECH: External Clock, High-Power mode: CLKIN on RA7/CLKIN

Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.
2: The entire data EEPROM will be erased when the code protection is turned off during an erase.
3: The entire program memory will be erased when the code protection is turned off.
4: Unimplemented on PIC16LF190X devices. This bit reads as ‘1’.
5: For PIC16LF190X only.
### REGISTER 3-3: CONFIGURATION WORD 2

<table>
<thead>
<tr>
<th>R/P-1</th>
<th>R/P-1</th>
<th>R/P-1&lt;sup&gt;(5)&lt;/sup&gt;</th>
<th>R/P-1</th>
<th>R/P-1</th>
<th>R/P-1</th>
<th>U-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>LVP</td>
<td>DEBUG</td>
<td>—</td>
<td>BORV</td>
<td>STVREN</td>
<td>PLLEN</td>
<td>—</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>bit 13</th>
<th>bit 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>U-1</td>
<td>U-1</td>
</tr>
<tr>
<td>VCAPEN&lt;sup&gt;(1)&lt;/sup&gt;</td>
<td>VCAPEN&lt;sup&gt;(3)&lt;/sup&gt;</td>
</tr>
<tr>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>WRT1</td>
<td>WRT0</td>
</tr>
</tbody>
</table>

**Legend:**

- **U** = Unimplemented bit, read as ‘0’
- **R** = Readable bit
- **W** = Writable bit
- ‘0’ = Bit is cleared
- ‘1’ = Bit is set
- ‘x’ = Bit is unknown

**bit 13**  
**LVP**: Low-Voltage Programming Enable bit<sup>(3)</sup>  
1 = Low-voltage programming enabled  
0 = MCLR/VPP must be used for programming high voltage

**bit 12**  
**DEBUG**: In-Circuit Debugger Mode bit  
1 = In-Circuit Debugger disabled, RB6/ICSPCLK and RB7/ICSPDAT are general purpose I/O pins  
0 = In-Circuit Debugger enabled, RB6/ICSPCLK and RB7/ICSPDAT are dedicated to the debugger

**bit 11**  
**Unimplemented**: Read as ‘1’

**bit 11<sup>(5)</sup>**  
**ULPBOR**: Ultra Low-Power BOR Enable bit  
1 = Ultra low-power BOR is disabled  
0 = Ultra low-power BOR is enabled

**bit 10**  
**BORV**: Brown-out Reset Voltage Selection bit  
1 = Brown-out Reset voltage set to 1.9V  
0 = Brown-out Reset voltage set to 2.7V

**bit 9**  
**STVREN**: Stack Overflow/Underflow Reset Enable bit  
1 = Stack overflow or underflow will cause a Reset  
0 = Stack overflow or underflow will not cause a Reset

**bit 8<sup>(4)</sup>**  
**PLLEN**: PLL Enable bit  
1 = 4xPLL enabled  
0 = 4xPLL disabled

**bit 7-6**  
**Unimplemented**: Read as ‘1’

**bit 5-4<sup>(1)</sup>**  
For the PIC16F1933/1934/1936/1937/1938/1939:  
**VCAPEN<sup>1:0</sup>**: Voltage Regulator Capacitor Enable bits  
PIC16LF193x:  
These bits are unimplemented. All VCAP pin functions are disabled.

PIC16F193x:  
00 = VCap functionality is enabled on RA0  
01 = VCap functionality is enabled on RA5  
10 = VCap functionality is enabled on RA6  
11 = All VCap pin functions are disabled

**bit 5**  
**Unimplemented**: Read as ‘1’

**Note 1**: For PIC16F193X only.

**Note 2**: For PIC16F194X only.

**Note 3**: The LVP bit cannot be programmed to ‘0’ when Programming mode is entered via LVP.

**Note 4**: Unimplemented on PIC16LF190X devices. This bit reads as ‘1’.

**Note 5**: For PIC16LF190X only.
REGISTER 3-3: CONFIGURATION WORD 2 (CONTINUED)

bit 4\(^{(2,4)}\) For the PIC16F1946/1947:

VCAPEN\(^{(2)}\): Voltage Regulator Capacitor Enable bits

PIC16LF194x:
This bit is unimplemented. All VCAP pin functions are disabled.

PIC16F194x:
0 = VCAP functionality is enabled on RF0
1 = All VCAP pin functions are disabled

bit 3-2 Unimplemented: Read as ‘1’

bit 1-0 WRT<1:0>: Flash Memory Self-write Protection bits

4 kW Flash memory (PIC16F1933/PIC16LF1933 and PIC16F1934/PIC16LF1934 only):

11 = Write protection off
10 = 000h to 1FFh write protected, 200h to FFFh may be modified by EECON control
01 = 000h to 7FFh write protected, 800h to FFFh may be modified by EECON control
00 = 000h to FFFh write protected, no addresses may be modified by EECON control

PIC16LF1946):

11 = Write protection off
10 = 000h to 1FFh write protected, 200h to 1FFFh may be modified by EECON control
01 = 000h to FFFh write protected, 2000h to 1FFFh may be modified by EECON control
00 = 000h to 1FFFh write protected, no addresses may be modified by EECON control

PIC16LF1947):

11 = Write protection off
10 = 000h to 1FFh write protected, 200h to 3FFFh may be modified by EECON control
01 = 000h to FFFh write protected, 2000h to 3FFFh may be modified by EECON control
00 = 000h to 3FFFh write protected, no addresses may be modified by EECON control

bit 1-0\(^{(5)}\) WRT<1:0>: Flash Memory Self-Write Protection bits

2 kW Flash memory: PIC16LF1902:

11 = Write protection off
10 = 000h to 1FFh write protected, 200h to 7FFh may be modified by PMCON control
01 = 000h to 3FFh write protected, 400h to 7FFh may be modified by PMCON control
00 = 000h to 7FFh write protected, no addresses may be modified by PMCON control

4 kW Flash memory: PIC16LF1903/1904:

11 = Write protection off
10 = 000h to 1FFh write protected, 200h to FFFh may be modified by PMCON control
01 = 000h to 7FFh write protected, 800h to FFFh may be modified by PMCON control
00 = 000h to FFFh write protected, no addresses may be modified by PMCON control

8 kW Flash memory: PIC16LF1906/1907:

11 = Write protection off
10 = 000h to 1FFh write protected, 200h to 1FFFh may be modified by PMCON control
01 = 000h to FFFh write protected, 1000h to 1FFFh may be modified by PMCON control
00 = 000h to 1FFFh write protected, no addresses may be modified by PMCON control

Note 1: For PIC16F193X only.
2: For PIC16F194X only.
3: The LVP bit cannot be programmed to ‘0’ when Programming mode is entered via LVP.
4: Unimplemented on PIC16LF190X devices. This bit reads as ‘1’.
5: For PIC16LF190X only.
4.0 PROGRAM/VERIFY MODE

In Program/Verify mode, the program memory and the configuration memory can be accessed and programmed in serial fashion. ICSPDAT and ICSPCLK are used for the data and the clock, respectively. All commands and data words are transmitted LSb first. Data changes on the rising edge of the ICSPCLK and latched on the falling edge. In Program/Verify mode both the ICSPDAT and ICSPCLK are Schmitt Trigger inputs. The sequence that enters the device into Program/Verify mode places all other logic into the Reset state. Upon entering Program/Verify mode, all I/O’s are automatically configured as high-impedance inputs and the address is cleared.

4.1 High-Voltage Program/Verify Mode Entry and Exit

There are two different methods of entering Program/Verify mode via high-voltage:

• VPP – First entry mode
• VDD – First entry mode

4.1.1 VPP – FIRST ENTRY MODE

To enter Program/Verify mode via the VPP-first method the following sequence must be followed:

1. Hold ICSPCLK and ICSPDAT low. All other pins should be unpowered.
2. Raise the voltage on MCLR from 0V to VIHH.
3. Raise the voltage on V DD from 0V to the desired operating voltage.

The VPP-first entry prevents the device from executing code prior to entering Program/Verify mode. For example, when the Configuration Word has MCLR disabled (MCLRE = 0), the power-up time is disabled (PWRTE = 0), the internal oscillator is selected (FOSC = 100), and RB6 and RB7 are driven by the user application, the device will execute code. Since this may prevent entry, VPP-first entry mode is strongly recommended. See the timing diagram in Figure 8-3.

4.1.2 VDD – FIRST ENTRY MODE

To enter Program/Verify mode via the VDD-first method the following sequence must be followed:

1. Hold ICSPCLK and ICSPDAT low.
2. Raise the voltage on V DD from 0V to the desired operating voltage.
3. Raise the voltage on MCLR from V DD or below to VIHH.

The VDD-first method is useful when programming the device when V DD is already applied, for it is not necessary to disconnect V DD to enter Program/Verify mode. See the timing diagram in Figure 8-2.

4.1.3 PROGRAM/VERIFY MODE EXIT

To exit Program/Verify mode take MCLR to V DD or lower (VIL). See Figures 8-4 and 8-5.

4.2 Low-Voltage Programming (LVP) Mode

The Low-Voltage Programming mode allows the PIC16F193X/LF193X/PIC16F194X/LF194X/ PIC16LF190X devices to be programmed using V DD only, without high voltage. When the LVP bit of the Configuration Word 2 register is set to ‘1’, the low-voltage ICSP programming entry is enabled. To disable the Low-Voltage ICSP mode, the LVP bit must be programmed to ‘0’. This can only be done while in the High-Voltage Entry mode.

Entry into the Low-Voltage ICSP Program/Verify modes requires the following steps:

1. MCLR is brought to VIL.
2. A 32-bit key sequence is presented on ICSPDAT, while clocking ICSPCLK.

The key sequence is a specific 32-bit pattern, *0100 1101 0100 0011 0100 1000 0101 0000* (more easily remembered as MCHP in ASCII). The device will enter Program/Verify mode only if the sequence is valid. The Least Significant bit of the Least Significant nibble must be shifted in first.

Once the key sequence is complete, MCLR must be held at VIL for as long as Program/Verify mode is to be maintained.

For low-voltage programming timing, see Figures 8-9 and 8-10.

Exiting Program/Verify mode is done by no longer driving MCLR to VIL. See Figures 8-9 and 8-10.

Note: To enter LVP mode, the LSB of the Least Significant nibble must be shifted in first. This differs from entering the key sequence on other parts.
4.3 Program/Verify Commands

The PIC16F193X/194X and PIC16LF193X/194X/190X implement 13 programming commands, each six bits in length. The commands are summarized in Table 4-1.

Commands that have data associated with them are specified to have a minimum delay of TDLY between the command and the data. After this delay 16 clocks are required to either clock in or clock out the 14-bit data word. The first clock is for the Start bit and the last clock is for the Stop bit.

<table>
<thead>
<tr>
<th>Command</th>
<th>Mapping Data/Note</th>
<th>Binary (MSb ... LSb)</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Configuration</td>
<td></td>
<td>x 0 0 0 0 0 0 0</td>
<td>00h</td>
</tr>
<tr>
<td>Load Data For Program Memory</td>
<td></td>
<td>x 0 0 0 0 0 0 1 0</td>
<td>02h</td>
</tr>
<tr>
<td>Load Data For Data Memory</td>
<td></td>
<td>x 0 0 0 0 0 0 0 1 0</td>
<td>03h</td>
</tr>
<tr>
<td>Read Data From Program Memory</td>
<td></td>
<td>x 0 0 0 0 0 0 0 1</td>
<td>04h</td>
</tr>
<tr>
<td>Read Data From Data Memory</td>
<td></td>
<td>x 0 0 0 0 0 0 0 0 0</td>
<td>05h</td>
</tr>
<tr>
<td>Increment Address</td>
<td></td>
<td>x 0 0 0 1 1 0 0</td>
<td>06h</td>
</tr>
<tr>
<td>Reset Address</td>
<td></td>
<td>x 0 0 1 0 0 0 1</td>
<td>10h</td>
</tr>
<tr>
<td>Begin Internally Timed Programming</td>
<td></td>
<td>x 0 0 0 0 0 0 0 0 0</td>
<td>08h</td>
</tr>
<tr>
<td>Begin Externally Timed Programming</td>
<td></td>
<td>x 0 0 0 0 0 0 0 0 0</td>
<td>0Ah</td>
</tr>
<tr>
<td>End Externally Timed Programming</td>
<td></td>
<td>x 0 0 0 0 0 0 0 0 0</td>
<td>08h</td>
</tr>
<tr>
<td>Bulk Erase Program Memory</td>
<td>Internally Timed</td>
<td>x 0 0 0 0 0 0 0 0 0</td>
<td>09h</td>
</tr>
<tr>
<td>Bulk Erase Data Memory</td>
<td>Internally Timed</td>
<td>x 0 0 0 0 0 0 0 0 0</td>
<td>11h</td>
</tr>
<tr>
<td>Row Erase Program Memory</td>
<td>Internally Timed</td>
<td>x 0 0 0 0 0 0 0 0 0</td>
<td>11h</td>
</tr>
</tbody>
</table>
4.3.1 LOAD CONFIGURATION

The Load Configuration command is used to access the configuration memory (User ID Locations, Configuration Words, Calibration Words). The Load Configuration command sets the address to 8000h and loads the data latches with one word of data (see Figure 4-1).

After issuing the Load Configuration command, use the Increment Address command until the proper address to be programmed is reached. The address is then programmed by issuing either the Begin Internally Timed Programming or Begin Externally Timed Programming command.

**Note:** Externally timed writes are not supported for Configuration and Calibration bits. Any externally timed write to the Configuration or Calibration Word will have no effect on the targeted word.

The only way to get back to the program memory (address 0) is to exit Program/Verify mode or issue the Reset Address command after the configuration memory has been accessed by the Load Configuration command.

![Figure 4-1: LOAD CONFIGURATION](image1)

![Figure 4-2: LOAD DATA FOR PROGRAM MEMORY](image2)

4.3.2 LOAD DATA FOR PROGRAM MEMORY

The Load Data for Program Memory command is used to load one 14-bit word into the data latches. The word programs into program memory after the Begin Internally Timed Programming or Begin Externally Timed Programming command is issued (see Figure 4-2).
4.3.3 LOAD DATA FOR DATA MEMORY

The Load Data for Data Memory command will load 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 (see Figure 4-3).

FIGURE 4-3: LOAD DATA FOR DATA MEMORY COMMAND

4.3.4 READ DATA FROM PROGRAM MEMORY

The Read Data from Program Memory command will transmit data bits out of the program memory map currently accessed, starting with the second rising edge of the clock input. The ICSPDAT pin will go into Output mode on the first falling clock edge, and it will revert to Input mode (high-impedance) after the 16th falling edge of the clock. If the program memory is code-protected (CP), the data will be read as zeros (see Figure 4-4).

FIGURE 4-4: READ DATA FROM PROGRAM MEMORY
4.3.5 READ DATA FROM DATA MEMORY

The Read Data from Data Memory command will transmit data bits out of the data memory starting with the second rising edge of the clock input. The ICSPDAT pin will go into Output mode on the second rising edge, and it will revert to Input mode (high-impedance) after the 16th rising edge. The data memory is 8 bits wide, and therefore, only the first 8 bits that are output are actual data. If the data memory is code-protected, the data is read as all zeros. A timing diagram of this command is shown in Figure 4-5.

FIGURE 4-5: READ DATA FROM DATA MEMORY COMMAND

4.3.6 INCREMENT ADDRESS

The address is incremented when this command is received. It is not possible to decrement the address. To reset this counter, the user must use the Reset Address command or exit Program/Verify mode and re-enter it.

If the address is incremented from address 7FFFh, it will wrap around to location 0000h. If the address is incremented from FFFFh, it will wrap around to location 8000h.

FIGURE 4-6: INCREMENT ADDRESS
4.3.7 RESET ADDRESS

The Reset Address command will reset the address to 0000h, regardless of the current value. The address is used in program memory or the configuration memory.

**FIGURE 4-7: RESET ADDRESS**

4.3.8 BEGIN INTERNALLY TIMED PROGRAMMING

A Load Configuration or Load Data for Program Memory command must be given before every Begin Programming command. Programming of the addressed memory will begin after this command is received. An internal timing mechanism executes the write. The user must allow for the program cycle time, \( TP_{INT} \), for the programming to complete.

The End Externally Timed Programming command is not needed when the Begin Internally Timed Programming is used to start the programming.

The program memory address that is being programmed is not erased prior to being programmed. However, the EEPROM memory address that is being programmed is erased prior to being programmed with internally timed programming.

**FIGURE 4-8: BEGIN INTERNALLY TIMED PROGRAMMING**
4.3.9 BEGIN EXTERNALLY TIMED PROGRAMMING

A Load Configuration, Load Data for Program Memory or Load Data for Data Memory command must be given before every Begin Programming command. Programming of the addressed memory will begin after this command is received. To complete the programming, the End Externally Timed Programming command must be sent in the specified time window defined by TPEXT. No internal erase is performed for the data EEPROM, therefore, the device should be erased prior to executing this command (see Figure 4-9).

FIGURE 4-9: BEGIN EXTERNALLY TIMED PROGRAMMING

4.3.10 END EXTERNALLY TIMED PROGRAMMING

This command is required after a Begin Externally Timed Programming command is given. This command must be sent within the time window specified by TPEXT after the Begin Externally Timed Programming command is sent.

After sending the End Externally Timed Programming command, an additional delay (TDIS) is required before sending the next command. This delay is longer than the delay ordinarily required between other commands (see Figure 4-10).

FIGURE 4-10: END EXTERNALLY TIMED PROGRAMMING
4.3.11 BULK ERASE PROGRAM MEMORY

The Bulk Erase Program Memory command performs two different functions dependent on the current state of the address.

Address 0000h-7FFFh:
- Program Memory is erased
- Configuration Words are erased
- If $\text{CPD} = 0$, Data Memory is erased

Address 8000h-8008h:
- Program Memory is erased
- Configuration Words are erased
- User ID Locations are erased
- If $\text{CPD} = 0$, Data Memory is erased

A Bulk Erase Program Memory command should not be issued when the address is greater than 8008h.

**FIGURE 4-11: BULK ERASE PROGRAM MEMORY**

4.3.12 BULK ERASE DATA MEMORY

To perform an erase of the data memory, after a Bulk Erase Data Memory command, wait a minimum of $\text{TERAB}$ to complete Bulk Erase.

To erase data memory when data code-protect is active ($\text{CPD} = 0$), the Bulk Erase Program Memory command should be used.

**FIGURE 4-12: BULK ERASE DATA MEMORY COMMAND**

Note: The code protection Configuration bit ($\text{CP}$) has no effect on the Bulk Erase Program Memory command.

Note: Data memory will not erase if code-protected ($\text{CPD} = 0$).
4.3.13 ROW ERASE PROGRAM MEMORY

The Row Erase Program Memory command will erase an individual row. A row of program memory consists of 32 consecutive 14-bit words. A row is addressed by the address PC<15:5>. If the program memory is code-protected, the Row Erase Program Memory command will be ignored. When the address is 8000h-8008h the Row Erase Program Memory command will only erase the user ID locations regardless of the setting of the CP Configuration bit.

After receiving the Row Erase Program Memory command the erase will not complete until the time interval, TERAR, has expired.

**FIGURE 4-13: ROW ERASE PROGRAM MEMORY**

![Diagram of Row Erase Program Memory]
5.0 PROGRAMMING ALGORITHMS

The devices have the capability of storing eight 14-bit words in its data latches. The data latches are internal and are only used for programming. The data latches allow the user to program up to eight program words with a single Begin Externally Timed Programming or Begin Internally Timed Programming command. The Load Program Data or the Load Configuration command is used to load a single data latch. The data latch will hold the data until the Begin Externally Timed Programming or Begin Internally Timed Programming command is given.

The data latches are aligned with the 3 LSb of the address. The address at the time the Begin Externally Timed Programming or Begin Internally Timed Programming command is given will determine which location(s) in memory are written. Writes cannot cross a physical eight-word boundary. For example, attempting to write from address 0002h-0009h will result in data being written to 0008h-000Fh.

If more than 8 data latches are written without a Begin Externally Timed Programming or Begin Internally Timed Programming command the data in the data latches will be overwritten. The following figures show the recommended flowcharts for programming.
FIGURE 5-1: DEVICE PROGRAM/VERIFY FLOWCHART

Note 1: See Figure 5-2.
2: See Figure 5-5.
3: See Figure 5-6.
**FIGURE 5-2: PROGRAM MEMORY FLOWCHART**

Start

Bulk Erase Program Memory\(^{(1, 2)}\)

Program Cycle\(^{(3)}\)

Read Data from Program Memory

Data Correct? No

Report Programming Failure

Yes

Increment Address Command

All Locations Done? No

Yes

Done

---

**Note 1:** This step is optional if 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 device per Figure 5-8.

3: See Figure 5-3 or Figure 5-4.
FIGURE 5-3: ONE-WORD PROGRAM CYCLE

Program Cycle

Load Data for Program Memory

Begin Programming Command (Internally timed)

Begin Programming Command (Externally timed)\(^{(1)}\)

Wait TPINT

Wait TPEXT

End Programming Command

Wait TDIS

\textbf{Note 1:} Externally timed writes are not supported for Configuration and Calibration bits.
FIGURE 5-4: MULTIPLE-WORD PROGRAM CYCLE

Program Cycle

Load Data for Program Memory

Latch 1

Increment Address Command

Load Data for Program Memory

Latch 2

•

•

•

Increment Address Command

Load Data for Program Memory

Latch 8

Begin Programming Command (Internally timed)

Wait TPINT

Begin Programming Command (Externally timed)

Wait TPEXT

End Programming Command

Wait TDIS
FIGURE 5-5: CONFIGURATION MEMORY PROGRAM FLOWCHART

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

**Note 2:** See Figure 5-3.
FIGURE 5-6: DATA MEMORY PROGRAM FLOWCHART

Note 1: See Figure 5-7.
FIGURE 5-7: DATA MEMORY PROGRAM CYCLE

Program Cycle

Load Data for Data Memory

Begin Programming Command (Internally timed)

Wait TPINT

Begin Programming Command (Externally timed)

Wait TPEXT

End Programming Command

Wait TDIS

FIGURE 5-8: ERASE FLOWCHART

Start

Load Configuration

Bulk Erase Program Memory

Bulk Erase Data Memory

Done

Note: This sequence does not erase the Calibration Words.
6.0 CODE PROTECTION

Code protection is controlled using the CP bit in Configuration Word 1. When code protection is enabled, all program memory locations (0000h-7FFFh) read as all ‘0’. Further programming is disabled for the program memory (0000h-7FFFh). Program memory can still be programmed and read during program execution.

Data memory is protected with its own Code-Protect bit (CPD). When data code protection is enabled (CPD = 0), all data memory locations read as ‘0’. Further programming is disabled for the data memory. Data memory can still be programmed and read during program execution.

The user ID locations and Configuration Words can be programmed and read out regardless of the code protection settings.

6.1 Program Memory

Code protection is enabled by programming the CP bit in Configuration Word 1 register to ‘0’.

The only way to disable code protection is to use the Bulk Erase Program Memory command.

6.2 Data Memory

Data memory protection is enabled by programming the CPD bit in Configuration Word 1 register to ‘0’.

The only way to disable code protection is to use the Bulk Erase Program Memory command.

Note: To ensure system security, if CPD bit = 0, the Bulk Erase Program Memory command will also erase data memory.

7.0 HEX FILE USAGE

In the hex file there are two bytes per program word stored in the Intel® INHX32 hex format. Data is stored LSB first, MSB second. Because there are two bytes per word, the addresses in the hex file are 2x the address in program memory. (Example: The Configuration Word 1 is stored at 8007h on the PIC16F193X/LF193X/PIC16F194X/LF194X/ PIC16LF190X. In the hex file this will be referenced as 1000Eh-1000Fh).

7.1 Configuration Word

To allow portability of code, it is strongly recommended that the programmer is able to read the Configuration Words and user ID locations from the hex file. If the Configuration Words information was not present in the hex file, a simple warning message may be issued. Similarly, while saving a hex file, Configuration Words and user ID information should be included.

7.2 Device ID and Revision

If a device ID is present in the hex file at 1000Ch-1000Dh (8006h on the part), the programmer should verify the device ID (excluding the revision) against the value read from the part. On a mismatch condition the programmer should generate a warning message.

7.3 Data EEPROM

The programmer should be able to read data memory information from a hex file and write data memory contents to a hex file.

The physical address range of the 256 data memory is 0000h-00FFh. However, these addresses are logically mapped to address 1E000h-1E1FFh in the hex file. This provides a way of differentiating between the data and program memory locations in this range. The format for data memory storage is one data byte per address location, LSb aligned.
7.4 Checksum Computation

The checksum is calculated by two different methods dependent on the setting of the CP Configuration bit.

### 7.4.1 PROGRAM CODE PROTECTION DISABLED

With the program code protection disabled, the checksum is computed by reading the contents of the PIC16F193X/LF193X/PIC16F194X/LF194X/ PIC16LF190X program memory locations and adding up the program memory data starting at address 0000h, up to the maximum user addressable location (e.g., 1FFFh for the PIC16F1936). Any Carry bit exceeding 16 bits are ignored. Additionally, the relevant bits of the Configuration Words are added to the checksum. All unimplemented Configuration bits are masked to ‘0’.

**Note:** Data memory does not effect the checksum.

### Table 7-1: Configuration Word Mask Values

<table>
<thead>
<tr>
<th>Device</th>
<th>Config. Word 1 Mask</th>
<th>Config. Word 2 Mask</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIC16F1933</td>
<td>3FFFh</td>
<td>3733h</td>
</tr>
<tr>
<td>PIC16LF1933</td>
<td>3FFFh</td>
<td>3703h</td>
</tr>
<tr>
<td>PIC16F1934</td>
<td>3FFFh</td>
<td>3733h</td>
</tr>
<tr>
<td>PIC16LF1934</td>
<td>3FFFh</td>
<td>3703h</td>
</tr>
<tr>
<td>PIC16F1936</td>
<td>3FFFh</td>
<td>3733h</td>
</tr>
<tr>
<td>PIC16LF1936</td>
<td>3FFFh</td>
<td>3703h</td>
</tr>
<tr>
<td>PIC16F1937</td>
<td>3FFFh</td>
<td>3733h</td>
</tr>
<tr>
<td>PIC16LF1937</td>
<td>3FFFh</td>
<td>3703h</td>
</tr>
<tr>
<td>PIC16F1938</td>
<td>3FFFh</td>
<td>3733h</td>
</tr>
<tr>
<td>PIC16LF1938</td>
<td>3FFFh</td>
<td>3703h</td>
</tr>
<tr>
<td>PIC16F1939</td>
<td>3FFFh</td>
<td>3733h</td>
</tr>
<tr>
<td>PIC16LF1939</td>
<td>3FFFh</td>
<td>3703h</td>
</tr>
<tr>
<td>PIC16F1946</td>
<td>3FFFh</td>
<td>3733h</td>
</tr>
<tr>
<td>PIC16LF1946</td>
<td>3FFFh</td>
<td>3703h</td>
</tr>
<tr>
<td>PIC16F1947</td>
<td>3FFFh</td>
<td>3713h</td>
</tr>
<tr>
<td>PIC16LF1947</td>
<td>3FFFh</td>
<td>3703h</td>
</tr>
<tr>
<td>PIC16LF1902</td>
<td>0EFBh</td>
<td>3E03h</td>
</tr>
<tr>
<td>PIC16LF1903</td>
<td>0EFBh</td>
<td>3E03h</td>
</tr>
<tr>
<td>PIC16LF1904</td>
<td>0EFBh</td>
<td>3E03h</td>
</tr>
<tr>
<td>PIC16LF1906</td>
<td>0EFBh</td>
<td>3E03h</td>
</tr>
<tr>
<td>PIC16LF1907</td>
<td>0EFBh</td>
<td>3E03h</td>
</tr>
</tbody>
</table>

**Example 7-1: Checksum Computed with Program Code Protection Disabled
PIC16F1936, Blank Device

PIC16F1936  
Sum of Memory addresses 0000h-1FFFh  
= E000h

Configuration Word 1  
= 3FFFh

Configuration Word 1 mask  
= 3FFFh

Configuration Word 2  
= 3FFFh

Configuration Word 2 mask(1)  
= 3733h

Checksum  
= E000h + (3FFFh and 3FFFh) + (3FFFh and 3733h)  
= E000h + 3FFFh + 3733h  
= 5732h

**Note 1:** In PIC16F194X devices, the VCAPEN<1> bit is not implemented in Configuration Word 2 and the Configuration Word 2 mask is 3713h.
EXAMPLE 7-2: CHECKSUM COMPUTED WITH PROGRAM CODE PROTECTION DISABLED
PIC16LF1936, 00AAh AT FIRST AND LAST ADDRESS

<table>
<thead>
<tr>
<th>PIC16LF1936</th>
<th>Sum of Memory addresses 0000h-1FFFh</th>
<th>6156h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration Word 1</td>
<td>3FFFh</td>
<td></td>
</tr>
<tr>
<td>Configuration Word 1 mask</td>
<td>3FFFh</td>
<td></td>
</tr>
<tr>
<td>Configuration Word 2</td>
<td>3FFFh</td>
<td></td>
</tr>
<tr>
<td>Configuration Word 2 mask</td>
<td>3703h</td>
<td></td>
</tr>
<tr>
<td>Checksum = 6156h + (3FFFh and 3FFFh) + (3FFFh and 3703h)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>= 6156h + 3FFFh + 3703h</td>
<td></td>
<td></td>
</tr>
<tr>
<td>= D858h</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note 1: In PIC16F194X devices, the VCAPEN<1> bit is not implemented in Configuration Word 2 and the Configuration Word 2 mask is 3713h.

7.4.2 PROGRAM CODE PROTECTION ENABLED

With the program code protection enabled the checksum is computed in the following manner. The Least Significant nibble of each user ID is used to create a 16-bit value. The masked value of user ID location 8000h is the Most Significant nibble. This sum of user IDs is summed with the Configuration Words (all unimplemented Configuration bits are masked to '0').

Note: Data memory does not effect the checksum.

EXAMPLE 7-3: CHECKSUM COMPUTED WITH PROGRAM CODE PROTECTION ENABLED
PIC16F1936, BLANK DEVICE

<table>
<thead>
<tr>
<th>PIC16F1936</th>
<th>Configuration Word 1</th>
<th>3F7Fh</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration Word 1 mask</td>
<td>3FFFh</td>
<td></td>
</tr>
<tr>
<td>Configuration Word 2</td>
<td>3FFFh</td>
<td></td>
</tr>
<tr>
<td>Configuration Word 2 mask</td>
<td>3733h</td>
<td></td>
</tr>
<tr>
<td>User ID (8000h)</td>
<td>0005h</td>
<td></td>
</tr>
<tr>
<td>User ID (8001h)</td>
<td>0007h</td>
<td></td>
</tr>
<tr>
<td>User ID (8002h)</td>
<td>0003h</td>
<td></td>
</tr>
<tr>
<td>User ID (8003h)</td>
<td>0002h</td>
<td></td>
</tr>
<tr>
<td>Sum of User IDs = (0005h and 000Fh) &lt;&lt; 12 + (0007h and 000Fh) &lt;&lt; 8 + (0003h and 000Fh) &lt;&lt; 4 + (0002h and 000Fh)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>= 5000h + 0700h + 0030h + 0002h</td>
<td></td>
<td></td>
</tr>
<tr>
<td>= 5732h</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Checksum = (3F7Fh and 3FFFh) + (3FFFh and 3733h) + Sum of User IDs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>= 3F7Fh + 3773h + 5732h</td>
<td></td>
<td></td>
</tr>
<tr>
<td>= CDE4h</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note 1: In PIC16F194X devices, the VCAPEN<1> bit is not implemented in Configuration Word 2 and the Configuration Word 2 mask is 3713h.
EXAMPLE 7-4:  
CHECKSUM COMPUTED WITH PROGRAM CODE PROTECTION ENABLED
PIC16LF1936, 00AAh AT FIRST AND LAST ADDRESS

<table>
<thead>
<tr>
<th>PIC16LF1936</th>
<th>Configuration Word 1</th>
<th>3F7Fh</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Configuration Word 1 mask</td>
<td>3FFFh</td>
</tr>
<tr>
<td></td>
<td>Configuration Word 2</td>
<td>3FFFh</td>
</tr>
<tr>
<td></td>
<td>Configuration Word 2 mask(1)</td>
<td>3703h</td>
</tr>
<tr>
<td></td>
<td>User ID (8000h)</td>
<td>000Dh</td>
</tr>
<tr>
<td></td>
<td>User ID (8001h)</td>
<td>0008h</td>
</tr>
<tr>
<td></td>
<td>User ID (8002h)</td>
<td>0005h</td>
</tr>
<tr>
<td></td>
<td>User ID (8003h)</td>
<td>0008h</td>
</tr>
<tr>
<td></td>
<td>Sum of User IDs</td>
<td>D000h + 0800h + 0050h + 0008h</td>
</tr>
<tr>
<td></td>
<td>Checksum</td>
<td>(3F7Fh and 3FFFh) + (3FFFh and 3703h) + Sum of User IDs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>= 4EDAh</td>
</tr>
</tbody>
</table>

**Note 1:** In PIC16F194X devices, the VCAPEN<1> bit is not implemented in Configuration Word 2 and the Configuration Word 2 mask is 3713h.
## 8.0 ELECTRICAL SPECIFICATIONS

Refer to device specific data sheet for absolute maximum ratings.

### TABLE 8-1: AC/DC CHARACTERISTICS TIMING REQUIREMENTS FOR PROGRAM/VERIFY MODE

<table>
<thead>
<tr>
<th>AC/DC CHARACTERISTICS</th>
<th>Standard Operating Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Production tested at 25°C</td>
</tr>
<tr>
<td><strong>Sym.</strong></td>
<td><strong>Characteristics</strong></td>
</tr>
<tr>
<td><strong>VDD</strong></td>
<td><strong>Vdd</strong></td>
</tr>
<tr>
<td>Read/Write and Row Erase operations</td>
<td>PIC16F193X/ PIC16F194X</td>
</tr>
<tr>
<td></td>
<td>PIC16LF193X/ PIC16LF194X</td>
</tr>
<tr>
<td></td>
<td>PIC16LF190X</td>
</tr>
<tr>
<td>Bulk Erase operations</td>
<td>PIC16F193X/ PIC16F194X</td>
</tr>
<tr>
<td></td>
<td>PIC16LF193X/ PIC16LF194X</td>
</tr>
<tr>
<td></td>
<td>PIC16LF190X</td>
</tr>
<tr>
<td>IDDI</td>
<td>Current on Vdd, Idle</td>
</tr>
<tr>
<td>IDDP</td>
<td>Current on Vdd, Programming</td>
</tr>
<tr>
<td><strong>IPP</strong></td>
<td><strong>Vpp</strong></td>
</tr>
<tr>
<td>Current on MCLR/VPP</td>
<td></td>
</tr>
<tr>
<td><strong>VIHH</strong></td>
<td>High voltage on MCLR/VPP for Program/Verify mode entry</td>
</tr>
<tr>
<td><strong>TVHHR</strong></td>
<td>MCLR rise time (VIL to VIHH) for Program/Verify mode entry</td>
</tr>
<tr>
<td><strong>I/O pins</strong></td>
<td><strong>VIH</strong></td>
</tr>
<tr>
<td>(ICSPCLK, ICSPDAT, MCLR/VPP) input high level</td>
<td></td>
</tr>
<tr>
<td><strong>VIL</strong></td>
<td>(ICSPCLK, ICSPDAT, MCLR/VPP) input low level</td>
</tr>
<tr>
<td><strong>VOH</strong></td>
<td>ICSPDAT output high level</td>
</tr>
<tr>
<td>Vdd-0.7</td>
<td>Vdd-0.7</td>
</tr>
<tr>
<td><strong>VOL</strong></td>
<td>ICSPDAT output low level</td>
</tr>
<tr>
<td>Vss+0.6</td>
<td>Vss+0.6</td>
</tr>
<tr>
<td>Programming mode entry and exit</td>
<td><strong>TENTS</strong></td>
</tr>
<tr>
<td></td>
<td>Programing mode entry hold time: ICSPCLK, ICSPDAT hold time after VDD or MCLR</td>
</tr>
<tr>
<td><strong>Serial Program/Verify</strong></td>
<td><strong>TCKL</strong></td>
</tr>
<tr>
<td><strong>TCKH</strong></td>
<td>Clock High Pulse Width</td>
</tr>
<tr>
<td><strong>TDS</strong></td>
<td>Data in setup time before clock↓</td>
</tr>
<tr>
<td><strong>TDH</strong></td>
<td>Data in hold time after clock↓</td>
</tr>
<tr>
<td><strong>TCO</strong></td>
<td>Clock↑ to data out valid (during a Read Data command)</td>
</tr>
<tr>
<td><strong>TLZD</strong></td>
<td>Clock↓ to data low-impedance (during a Read Data command)</td>
</tr>
<tr>
<td><strong>THZD</strong></td>
<td>Clock↓ to data high-impedance (during a Read Data command)</td>
</tr>
<tr>
<td><strong>TDLY</strong></td>
<td>Data input not driven to next clock input (delay required between command/data or command/ command)</td>
</tr>
<tr>
<td><strong>TERAB</strong></td>
<td>Bulk Erase cycle time</td>
</tr>
<tr>
<td><strong>TERAR</strong></td>
<td>Row Erase cycle time</td>
</tr>
</tbody>
</table>
### 8.1 AC Timing Diagrams

**FIGURE 8-2: PROGRAMMING MODE ENTRY – VDD FIRST**

**FIGURE 8-3: PROGRAMMING MODE ENTRY – VPP FIRST**

**FIGURE 8-4: PROGRAMMING MODE EXIT – VPP LAST**

**FIGURE 8-5: PROGRAMMING MODE EXIT – VDD LAST**

<table>
<thead>
<tr>
<th>AC/DC CHARACTERISTICS</th>
<th>Standard Operating Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Production tested at 25°C</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sym.</th>
<th>Characteristics</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
<th>Conditions/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPINT</td>
<td>Internally timed programming operation time</td>
<td>—</td>
<td>—</td>
<td>2.5</td>
<td>ms</td>
<td>Program memory</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
<td></td>
<td>Configuration words</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
<td></td>
<td>Data EEPROM</td>
</tr>
<tr>
<td>TPEXT</td>
<td>Externally timed programming pulse</td>
<td>1.0</td>
<td>—</td>
<td>2.1</td>
<td>ms</td>
<td>Note 1</td>
</tr>
<tr>
<td>TDIS</td>
<td>Time delay from program to compare (HV discharge time)</td>
<td>300</td>
<td>—</td>
<td>—</td>
<td>µs</td>
<td></td>
</tr>
<tr>
<td>TEXIT</td>
<td>Time delay when exiting Program/Verify mode</td>
<td>1</td>
<td>—</td>
<td>—</td>
<td>µs</td>
<td></td>
</tr>
</tbody>
</table>

**Note 1:** Externally timed writes are not supported for Configuration and Calibration bits.
FIGURE 8-6: CLOCK AND DATA TIMING

FIGURE 8-7: WRITE COMMAND-PAYLOAD TIMING
FIGURE 8-8: READ COMMAND-PAYLOAD TIMING

![Diagram showing ICSPCLK, ICSPDAT, and TDLY timing for read command-payload timing.](image)

FIGURE 8-9: LVP ENTRY (POWERING UP)

![Diagram showing Vdd, MCLR, ICSPCLK, and ICSPDAT timing for LVP entry during powering up.](image)
Note 1: Sequence matching can start with no edge on MCLR first.
APPENDIX A: REVISION HISTORY

Revision A (09/2009)
Original release of this document.

Revision B (08/2010)
Revised Pin Diagrams; Added Notes to sections 4.3.1; Revised 4.3.9; Added Note 1 to Figure 5-3; Added Note 1 to Table 8-1; Other minor corrections; Added PIC16LF190X devices.
Note the following details of the code protection feature on Microchip devices:

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

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

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

Trademark

The Microchip name and logo, the Microchip logo, dsPIC, KEELOG, KEELOG logo, MPLAB, PIC, PICmicro, PICSTART, PIC® logo, rPIC and Uni/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

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

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

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

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

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

Printed on recycled paper.


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

### AMERICAS

<table>
<thead>
<tr>
<th>Corporate Office</th>
<th>Address</th>
<th>Phone Numbers</th>
<th>Fax Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corporate Office</td>
<td>2355 West Chandler Blvd. Chandler, AZ 85224-6199</td>
<td>Tel: 480-792-7200</td>
<td>Fax: 480-792-7277</td>
</tr>
<tr>
<td>Technical Support</td>
<td><a href="http://support.microchip.com">http://support.microchip.com</a></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Web Address:</td>
<td><a href="http://www.microchip.com">www.microchip.com</a></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ATLANTA</td>
<td>Duluth, GA</td>
<td>Tel: 678-957-9614</td>
<td>Fax: 678-957-1455</td>
</tr>
<tr>
<td>BOSTON</td>
<td>Westborough, MA</td>
<td>Tel: 774-760-0087</td>
<td>Fax: 774-760-0088</td>
</tr>
<tr>
<td>CHICAGO</td>
<td>Itasca, IL</td>
<td>Tel: 630-285-0071</td>
<td>Fax: 630-285-0075</td>
</tr>
<tr>
<td>CLEVELAND</td>
<td>Independence, OH</td>
<td>Tel: 216-447-0464</td>
<td>Fax: 216-447-0643</td>
</tr>
<tr>
<td>DALLAS</td>
<td>Addison, TX</td>
<td>Tel: 972-818-7423</td>
<td>Fax: 972-818-2924</td>
</tr>
<tr>
<td>DETROIT</td>
<td>Farmington Hills, MI</td>
<td>Tel: 248-538-2250</td>
<td>Fax: 248-538-2260</td>
</tr>
<tr>
<td>KOKOMO</td>
<td>Kokomo, IN</td>
<td>Tel: 765-864-8360</td>
<td>Fax: 765-864-8387</td>
</tr>
<tr>
<td>LOS ANGELES</td>
<td>Mission Viejo, CA</td>
<td>Tel: 949-462-9523</td>
<td>Fax: 949-462-9608</td>
</tr>
<tr>
<td>SANTA CLARA</td>
<td>Santa Clara, CA</td>
<td>Tel: 408-961-6444</td>
<td>Fax: 408-961-6445</td>
</tr>
<tr>
<td>TORONTO</td>
<td>Mississauga, Ontario, Canada</td>
<td>Tel: 905-673-0699</td>
<td>Fax: 905-673-6509</td>
</tr>
</tbody>
</table>

### ASIA/PACIFIC

<table>
<thead>
<tr>
<th>Asia Pacific Office</th>
<th>Address</th>
<th>Phone Numbers</th>
<th>Fax Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asia Pacific Office</td>
<td>Suites 3707-14, 37th Floor Tower 6, The Gateway Harbour City, Kowloon Hong Kong</td>
<td>Tel: 852-2401-1200</td>
<td>Fax: 852-2401-3431</td>
</tr>
<tr>
<td>China - Beijing</td>
<td>Tel: 86-10-8528-2100</td>
<td>Fax: 86-10-8528-2104</td>
<td></td>
</tr>
<tr>
<td>China - Chengdu</td>
<td>Tel: 86-28-8665-5511</td>
<td>Fax: 86-28-8665-7889</td>
<td></td>
</tr>
<tr>
<td>China - Chongqing</td>
<td>Tel: 86-23-8890-9588</td>
<td>Fax: 86-23-8890-9500</td>
<td></td>
</tr>
<tr>
<td>China - Hong Kong SAR</td>
<td>Tel: 852-2401-1200</td>
<td>Fax: 852-2401-3431</td>
<td></td>
</tr>
<tr>
<td>China - Nanjing</td>
<td>Tel: 86-25-8473-2460</td>
<td>Fax: 86-25-8473-2470</td>
<td></td>
</tr>
<tr>
<td>China - Qingdao</td>
<td>Tel: 86-532-8502-7355</td>
<td>Fax: 86-532-8502-7205</td>
<td></td>
</tr>
<tr>
<td>China - Shanghai</td>
<td>Tel: 86-21-5407-5533</td>
<td>Fax: 86-21-5407-5066</td>
<td></td>
</tr>
<tr>
<td>China - Shenyang</td>
<td>Tel: 86-24-2334-2829</td>
<td>Fax: 86-24-2334-2393</td>
<td></td>
</tr>
<tr>
<td>China - Shenzhen</td>
<td>Tel: 86-755-8203-2660</td>
<td>Fax: 86-755-8203-1760</td>
<td></td>
</tr>
<tr>
<td>China - Wuhan</td>
<td>Tel: 86-27-5980-5300</td>
<td>Fax: 86-27-5980-5118</td>
<td></td>
</tr>
<tr>
<td>China - Xian</td>
<td>Tel: 86-29-8833-7252</td>
<td>Fax: 86-29-8833-7256</td>
<td></td>
</tr>
<tr>
<td>China - Xiamen</td>
<td>Tel: 86-592-2388138</td>
<td>Fax: 86-592-2388130</td>
<td></td>
</tr>
<tr>
<td>China - Zhuhai</td>
<td>Tel: 86-756-3210040</td>
<td>Fax: 86-756-3210049</td>
<td></td>
</tr>
</tbody>
</table>

### ASIA/PACIFIC

| India - Bangalore               | Tel: 91-80-3090-4444                        | Fax: 91-80-3090-4123       |                     |
| India - New Delhi               | Tel: 91-11-4160-8631                        | Fax: 91-11-4160-8632       |                     |
| India - Pune                     | Tel: 91-20-2566-1512                        | Fax: 91-20-2566-1513       |                     |
| Japan - Yokohama                | Tel: 81-45-471-6166                         | Fax: 81-45-471-6122        |                     |
| Korea - Daegu                    | Tel: 82-53-744-4301                         | Fax: 82-53-744-4302        |                     |
| Korea - Seoul                    | Tel: 82-2-554-7200                          | Fax: 82-2-558-5932 or 82-2-558-5934 |                     |
| Malaysia - Kuala Lumpur          | Tel: 60-3-6201-9857                         | Fax: 60-3-6201-9859        |                     |
| Malaysia - Penang                | Tel: 60-4-227-8870                         | Fax: 60-4-227-4068         |                     |
| Philippines - Manila            | Tel: 63-2-634-9065                         | Fax: 63-2-634-9069         |                     |
| Singapore                        | Tel: 65-6334-8870                          | Fax: 65-6334-8850          |                     |
| Taiwan - Hsin Chu               | Tel: 886-3-6578-300                         | Fax: 886-3-6578-370        |                     |
| Taiwan - Kaohsiung               | Tel: 886-7-213-7830                         | Fax: 886-7-330-9305        |                     |
| Taiwan - Taipei                 | Tel: 886-2-2500-6610                        | Fax: 886-2-2508-0102       |                     |
| Thailand - Bangkok              | Tel: 66-2-694-1351                         | Fax: 66-2-694-1350         |                     |

### EUROPE

| Austria - Wels                    | Tel: 43-7242-2244-39                      | Fax: 43-7242-2244-393     |                     |
| Denmark - Copenhagen             | Tel: 45-4450-2828                         | Fax: 45-4485-2829         |                     |
| France - Paris                   | Tel: 33-1-69-53-63-20                     | Fax: 33-1-69-30-90-79     |                     |
| Germany - Munich                 | Tel: 49-89-627-144-0                      | Fax: 49-89-627-144-44     |                     |
| Italy - Milan                    | Tel: 39-0331-742611                       | Fax: 39-0331-466781       |                     |
| Netherlands - 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      |                     |

08/04/10