AN646

Interfacing Motorola 68HC11 to Microchip SPI™ Serial EEPROMS

Author: *Shannon Poulin*
Microchip Technology Inc.

INTRODUCTION

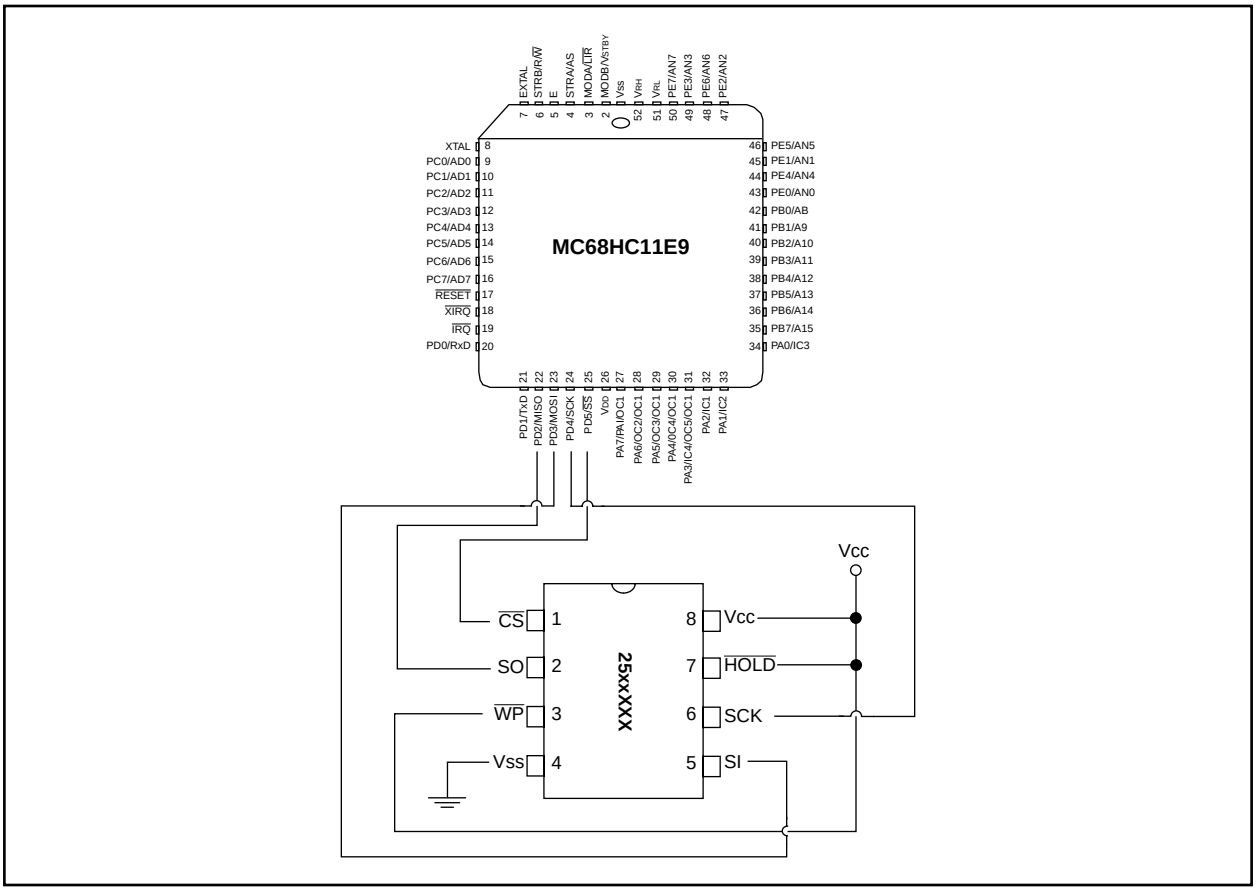
There are many different microcontrollers on the market today that are being used in embedded control applications. Many of these embedded control systems need non-volatile memory. Because of their small footprint, byte level flexibility, low I/O pin requirement, low power consumption, and low cost, Serial EEPROMs are a popular choice for non-volatile storage.

Microchip addresses these needs by offering a full line of Serial EEPROMs covering industry standard serial communication protocols for 2-wire, 3-wire, and SPI communication. Serial EEPROM devices are available in a variety of densities, operational voltage ranges, and packaging options.

Microchip realizes that its customer base is very broad, and because of this, different microcontrollers are used to interface to SPI Serial EEPROMs. One of the microcontrollers used in these applications is the Motorola 68HC11. In order to simplify the design process, Microchip has written an assembly code routine to communicate with our SPI parts that is verified and tested to function properly on a 68HC11.

Figure 1 describes the hardware schematic for the interface between Microchip's SPI devices and the Motorola 68HC11E9. The schematic shows the connections necessary between the microcontroller and the serial EEPROM, and the software was written assuming these connections. Appendix A contains a listing of the SPI source code.

FIGURE 1: CIRCUIT FOR MC68HC11E9



SPI is a trademark of Motorola.

Please check the Microchip BBS for the latest version of the source code. Microchip's Worldwide Web Address: www.microchip.com; Bulletin Board Support: MCHIPBBS using CompuServe® (CompuServe membership not required).

APPENDIX A: SOURCE CODE

```
an646.asm              Assembled with CASM   03/19/1997  16:10  PAGE 1


                 1  ;*****************************************************************
                 2  ;* This code demonstrates how the Microchip SPI Devices
                 3  ;* can be interfaced to the SPI port of the 68HC11 microcontroller.
                 4  ;* The interface uses the three SPI lines (SCK,MOSI and MISO) on
                 5  ;* the 68HC11 Microcontroller for the clock (SCK), data out (SO)
                 6  ;* and data in (SI). A chip select (CS) is generated with a
                 7  ;* general purpose port line (PD5). The 68HC11 is placed into the
                 8  ;* master mode which allows use of the slave select input (SS) for
                 9  ;* other things.  The code uses the 0 mode (CPOL = 0,  CPHA = 0)
                10  ;* of operation to define clock polarity and phase. MODE 3
                11  ;* (CPOL = 1, CPHA = 1) could also have been used.  The  baud rate
                12  ;* set at one eighth of the clock frequency (SPR0 = 0,  SPR1 = 0)
                13  ;*
                14  ;* This code shows a read and write operation, as well  as data
                15  ;* polling, setting the write enable latch, clearing the  write
                16  ;* enable latch, writing and reading the block protect
                17  ;* register
                18  ;*****************************************************************
                19
                20  ; LOAD VALUES FOR USE LATER
                21
                22
0000            23  MASKCS  EQU     $20             ;MASK THE CHIP ENABLE BIT
0000            24  DDRD    EQU     $09             ;PORTD DATA DIRECTION REGISTER
0000            25  PORTD   EQU     $08             ;PORT D ADDRESS
0000            26  SPCR    EQU     $28             ;SPI CONTROL REGISTER
0000            27  SPSR    EQU     $29             ;SPI STATUS REGISTER
0000            28  SPDR    EQU     $2A             ;SPI DATA REGISTER
0000            29  ADDRLO  EQU     $80             ;LOW ADDRESS LOCATION
0000            30  ADDRHI  EQU     $81             ;HIGH ADDRESS LOCATION
0000            31  RDATA   EQU     $82             ;READ DATA SCRATCH PAD LOCATION
0000            32  SDATA   EQU     $83             ;SEND DATA SCRATCH PAD LOCATION
                33
                34  ;*****************************************************
                35  ;* SET RESET VECTOR TO THE BEGINNING OF THE PROGRAM
                36  ;*****************************************************
                37  ;        ORG     $FFFE           ;RESET VECTOR TO PROGRAM ENTRY POINT
                38  ;        FDB     $E000
                39
                40  ;*********************************
                41  ;* BEGINNING OF PROGRAM EXECUTION
                42  ;*********************************
                43
B600            44    ORG     $B600           ;BEGINNING ADDRESS OF EXECUTABL E CODE
                45
B600 8E00FF     46  BEGIN   LDS     #$00FF          ;INITIALIZE STACK POINTER
B603 CE1000     47    LDX     #$1000          ;INITIALIZE PAGE OFFSET LOCATION
B606 863F       48    LDAA    #$3F            ;LOAD ACCUMULATOR WITH 3F HEX
B608 A709       49    STAA    DDRD,X          ;MAKE ALL PORTD PINS OUTPUTS
B60A 8650       50    LDAA    #$50            ;SPI (MODE 0,0), USE #$5C FOR (MODE 1,1)
B60C A728       51    STAA    SPCR,X          ;STORE VALUE IN THE SPI CONTROL REG
B60E A629       52    LDAA    SPSR,X          ;READ MICRO SPI STATUS REG, CLEAR SPIF
B610 8655       53    LDAA    #$55            ;LOAD ACCUMULATOR A WITH 2E HEX
B612 9780       54    STAA    ADDRLO          ;STORE THIS VALUE IN ADDRESS LOW
B614 8600       55    LDAA    #$00            ;LOAD ACCUMULATOR A WITH 00 HEX
```

```
B616 9781        56    STAA    ADDRHI          ;STORE THIS VALUE IN ADDRESS HIGH
                 57  ; SEND THE WRITE ENABLE SEQUENCE (WREN)
B618 BDB6EE      58    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW  ROUTINE
B61B 8606        59    LDAA    #$06            ;LOAD THE WRITE SELECT COMMAND
B61D 9783        60    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B61F BDB6E1      61    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B622 BDB6F2      62    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
                 63  ; SEND THE WRITE STATUS REGISTER SEQUENCE (WRSR)
B625 BDB6EE      64    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW ROUTINE
B628 8601        65    LDAA    #$01            ;LOAD THE WRITE STATUS  REGISTER COMMAND
B62A 9783        66    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B62C BDB6E1      67    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B62F 8600        68    LDAA    #$00            ;LOAD ZERO'S INTO THE STATUS REGISTER
B631 9783        69    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B633 BDB6E1      70    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B636 BDB6F2      71    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
                 72  ; WAIT THE REQUIRED 5mS FOR THE WRITE CYCLE TIMER Twc
B639 BDB6F6      73    JSR     DELAY           ;JUMP TO THE DELAY SUBROUTINE
                 74  ; SEND THE WRITE ENABLE SEQUENCE (WREN)
B63C BDB6EE      75    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW SUBROUTINE
B63F 8606        76    LDAA    #$06            ;LOAD THE WRITE ENABLE COMMAND
B641 9783        77    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B643 BDB6E1      78    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B646 BDB6F2      79    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
                 80  ; SEND THE READ STATUS REGISTER SEQUENCE (RDSR)
B649 BDB6EE      81    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW SUBROUTINE
B64C 8605        82    LDAA    #$05            ;LOAD THE READ STATUS REGISTER COMMAND
B64E 9783        83    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B650 BDB6E1      84    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B653 BDB6E1      85    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE (READ)
B656 BDB6F2      86    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
                 87  ; SEND THE WRITE SEQUENCE (WRITE)
B659 BDB6EE      88    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW SUBROUTINE
B65C 8602        89    LDAA    #$02            ;LOAD THE WRITE COMMAND
B65E 9783        90    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B660 BDB6E1      91    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
                 92  ; ***COMMENT OUT FOR USE WITH 25XX010, 25XX020, OR 25XX040*************
B663 9681        93    LDAA    ADDRHI          ;LOAD ACCA WITH ADDRHI
B665 9783        94    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B667 BDB6E1      95    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
                 96  ; ****************************************************************
B66A 9680        97    LDAA    ADDRLO          ;LOAD ACCA WITH ADDRLO
B66C 9783        98    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B66E BDB6E1      99    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B671 86AA       100    LDAA    #$AA            ;LOAD ACCA WITH HEX AA
B673 9783       101    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B675 BDB6E1     102    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B678 BDB6F2     103    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
                104  ; PERFORM DATA POLLING (RDSR BIT 0)
B67B BDB6EE     105    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW ROUTINE
B67E 8605       106    LDAA    #$05            ;LOAD THE READ STATUS REGISTER COMMAND
B680 9783       107    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B682 BDB6E1     108    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
                109  ;
B685 8630       110    LDAA    #$30            ;GIVE THE SPI DEVICE TIME TO SET WIP
B687 4A         111    LOOP    DECA            ;DECREMENT ACCUMULATOR A
B688 26FD       112    BNE     LOOP            ;BRANCH IF NOT EQUAL TO LOOP
                113  ;
B68A BDB6E1     114  POLLING JSR     OUTBYTE   ;READ THE DATA IN THE STATUS REGISTER
B68D 8601       115    LDAA    #$01            ;LOAD ACCA WITH 1 HEX
B68F 9482       116    ANDA    RDATA           ;AND ACCA WITH THE RECEIVED DATA
B691 26F7       117    BNE     POLLING         ;BRANCH IF NOT = TO POLLING
B693 BDB6F2     118    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
                119  ; SEND THE READ SEQUENCE, READ ADDRESS 0X55
B696 BDB6EE     120    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW ROUTINE
B699 8603       121    LDAA    #$03            ;LOAD THE READ COMMAND
```

```
B69B 9783       122    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B69D BDB6E1     123    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
                124  ; ***COMMENT OUT FOR USE WITH 25XX010, 25XX020, OR 25XX040*************
B6A0 9681       125    LDAA    ADDRHI          ;LOAD ACCA WITH ADDRHI
B6A2 9783       126    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B6A4 BDB6E1     127    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
                128  ; ***************************************************************
B6A7 9680       129    LDAA    ADDRLO          ;LOAD ACCA WITH ADDRLO
B6A9 9783       130    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B6AB BDB6E1     131    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B6AE BDB6E1     132    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE (READ)
B6B1 BDB6F2     133    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
                134  ; SEND THE WRITE ENABLE SEQUENCE (WREN)
B6B4 BDB6EE     135    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW ROUTINE
B6B7 8606       136    LDAA    #$06            ;LOAD THE WRITE ENABLE COMMAND
B6B9 9783       137    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B6BB BDB6E1     138    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B6BE BDB6F2     139    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
                140  ; SEND THE WRITE DISABLE SEQUENCE (WRDI)
B6C1 BDB6EE     141    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW ROUTINE
B6C4 8604       142    LDAA    #$04            ;LOAD THE WRITE DISABLE COMMAND
B6C6 9783       143    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B6C8 BDB6E1     144    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B6CB BDB6F2     145    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
                146  ; SEND THE READ STATUS REGISTER SEQUENCE
B6CE BDB6EE     147    JSR     CSLOW           ;JUMP TO THE CHIP SELECT LOW ROUTINE
B6D1 8605       148    LDAA    #$05            ;LOAD THE READ STATUS REGISTER COMMAND
B6D3 9783       149    STAA    SDATA           ;STORE THE VALUE IN THE SDATA LOCATION
B6D5 BDB6E1     150    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE
B6D8 BDB6E1     151    JSR     OUTBYTE         ;JUMP TO THE SEND BYTE ROUTINE (READ)
B6DB BDB6F2     152    JSR     CSHIGH          ;JUMP TO THE CHIP SELECT HIGH ROUTINE
B6DE 7EB600     153    JMP     BEGIN           ;LOOP UNTIL RESET
                154
                155
                156  ;*****************************************************************************
                157  ;* THIS ROUTINE SENDS A BYTE OUT TO THE SPI PART. THEN IT READS WHAT IS SENT
                158  ;* BACK ON THE DO PIN.  DATA IS SENT OUT FROM THE 68HC11 ON THE MOSI PIN AND
                159  ;*  READ IN ON THE MISO PIN.  THE CLOCK IS GENERATED BY THE SCK PIN. THIS
                160  ;*  POLLS THE SPIF BIT IN THE SPSR TO SEE IF THE TRANSFER HAS BEEN COMPLETED
                161  ;*  WHEN COMPLETE THE DATA READ IS AVAILABLE IN THE SPDR SPI DATA REGISTER
                162  ;*****************************************************************************
                163
B6E1 9683       164  OUTBYTE LDAA    SDATA             ;LOAD A WITH WHAT'S IN SDATA
B6E3 A72A       165    STAA    SPDR,X            ;SEND BYTE OUT PART
B6E5 1F2980FC   166  WAIT1   BRCLR   SPSR,X,$80,WAIT1  ;WAIT FOR BYTE TO BE SENT
B6E9 A62A       167    LDAA    SPDR,X            ;READ BYTE BACK INTO ACCA
B6EB 9782       168    STAA    RDATA             ;STORE THE READ IN LOCATION RDATA
B6ED 39         169    RTS                       ;RETURN FROM SUBROUTINE
                170
                171  ;*********************
                172  ;* SET CE (PD5) LOW
                173  ;*********************
                174
B6EE 1D0820     175  CSLOW   BCLR    PORTD,X,MASKCS          ;SET CE LOW
B6F1 39         176    RTS                       ;RETURN FROM SUBROUTINE
                177
                178  ;*********************
                179  ;* SET CE (PD5) HIGH
                180  ;*********************
                181
B6F2 1C0820     182  CSHIGH  BSET    PORTD,X,MASKCS          ;SET CE HIGH
B6F5 39         183    RTS                       ;RETURN FROM SUBROUTINE
                184
                185  ;*********************
                186  ;* DELAY SUBROUTINE
                187  ;*********************
```

```
                188  ;DELAY 250 (0xFA) * 400Ns X 50 (0x32)
                189
B6F6 8632       190  DELAY   LDAA    #$32                  ;LOAD ACCA WITH HEX 32
B6F8 C6FA       191  DEC2    LDAB    #$FA                  ;LOAD ACCB WITH FA HEX
B6FA 5A         192  DEC1    DECB                          ;DECREMENT ACCB
B6FB 26FD       193    BNE     DEC1                    ;BRANCH IF NOT EQUAL TO DEC1
B6FD 4A         194    DECA                            ;DECREMENT ACCA
B6FE 26F8       195    BNE     DEC2                    ;BRANCH IF NOT EQUAL TO DEC2
B700 39         196    RTS                             ;RETURN FROM SUBROUTINE
                197
```

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 MSC The Netherlands Accredited by the RvA — DNV — 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. The Company’s quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral 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: 480-792-7456

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

New York
150 Motor Parkway, Suite 202
Hauppauge, NY 11788
Tel: 631-273-5305 Fax: 631-273-5335

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

Toronto
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: 86-28-6766200 Fax: 86-28-6766599

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: 86-755-2350361 Fax: 86-755-2366086

Hong Kong
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: 65-334-8870 Fax: 65-334-8850

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

EUROPE

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
Gustav-Heinemann Ring 125
D-81739 Munich, 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
Arizona Microchip Technology Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5869 Fax: 44-118 921-5820

01/18/02