

Interfacing Microchip PIC16C64/74 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.

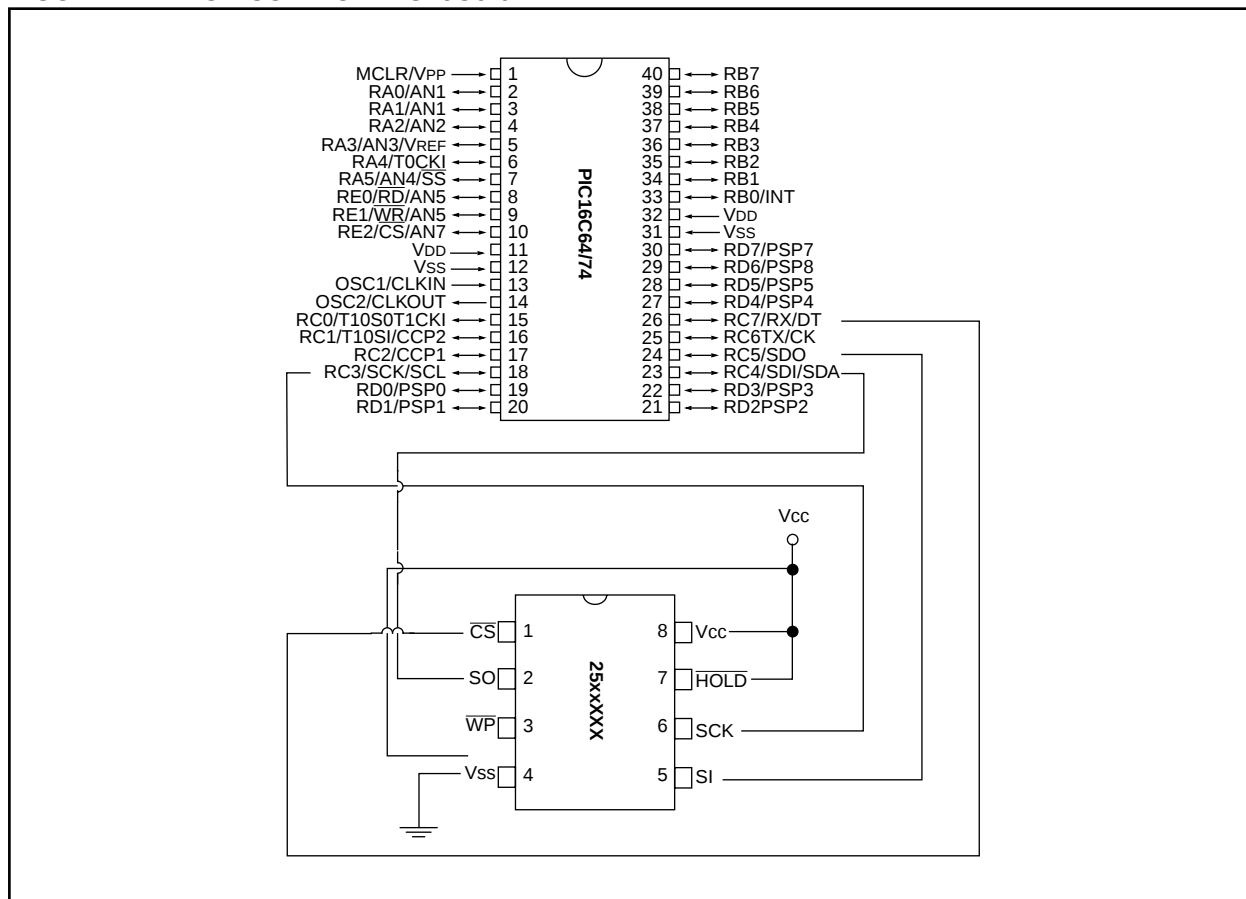
Microchip addresses these needs by offering a full line of serial EEPROMs covering industry standard serial communication protocols for two-wire, three-wire, and

SPI communication. Serial EEPROM devices are available in a variety of densities, operational voltage ranges, and packaging options.

This application note provides assistance and source code to ease the design process of interfacing a Microchip mid-range PIC16C6X/7X microcontroller and a Microchip SPI serial EEPROM. The hardware SPI port on the microcontroller provides a simple three-wire connection to the EEPROM and no external "glue" logic is required.

Figure 1 describes the hardware schematic for the interface between Microchip's SPI devices and the Microchip PIC16C64/74 Microcontroller. 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 PIC16C64/74



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

MPASM 01.40.01 Intermediate

AN647.ASM

3-20-1997 9:13:03

PAGE 1

LOC OBJECT CODE LINE SOURCE TEXT
VALUE

```

00001          LIST P=16C74
00002 ; Port C pin descriptions
00003 ; SCK bit = 3
00004 ; SDI bit = 4
00005 ; SD0 bit = 5
00006 ; CS bit = 7
00007 ;
00008 ; 10MHz crystal is being used, thus each instruction cycle = 400nS
00009 ;
00010 ;*****RAM register definitions*****
00000020      00011 rxdata equ 20h
00000021      00012 txdata equ 21h
00000022      00013 addr equ 22h
00000023      00014 loops equ 23h
00000024      00015 outbyte equ 24h
00000025      00016 temp1 equ 25h
00000026      00017 temp2 equ 26h
00018 ;*****Bit definitions*****
00000007      00019 CS equ 7
00020 ;*****Include file*****
00021          include "p16c74.inc" ; this is the include file for a PIC16C74
00001          LIST
00002 ; P16C74.INC Standard Header File, Ver. 1.00 Microchip Technology, Inc.
00318          LIST
00022 ;*****
0000      00023          org 0x000          ; set the reset vector
0000 2801      00024          goto start          ; go to the beginning of main
00025
00026 ;!!!!!!!!!!!!!!!!!!!!!!!!!!!!Begin Main Program!!!!!!!!!!!!!!!!!!!!!!!!!!!!
0001 1283      00027 start bcf STATUS,RP0          ; set to bank 0
0002 0187      00028          clrf PORTC          ; initialize portc to 0
0003 1683      00029          bsf STATUS,RP0          ; set to bank 1
0004 3010      00030          movlw 0x10          ; all bits are outputs except SDI
Message[302]: Register in operand not in bank 0. Ensure that bank bits are correct.
0005 0087      00031          movwf TRISC          ; move the value to TRIS portc
0006 1283      00032          bcf STATUS,RP0          ; set to bank0
0007 1787      00033          bsf PORTC,CS          ; make the chip select is high
Message[302]: Register in operand not in bank 0. Ensure that bank bits are correct.
0008 018C      00034          clrf PIE1          ; disable peripheral interrupts
0009 018B      00035          clrf INTCON          ; disables all interrupts
000A 1283      00036          bcf STATUS,RP0          ; set to bank 0
000B 0194      00037          clrf SSPCON          ; clear SSP control register
000C 3031      00038          movlw 0x31          ; set up spi port, SPI master,
000D 0094      00039          movwf SSPCON          ; clk/16, ckp=1 (mode 1,1)
000E 3055      00040          movlw 0x55          ; put starting address 55 in
000F 00A2      00041          movwf addr          ; addr for later use
00042 ;Send the write enable sequence (WREN)
0010 1387      00043          bcf PORTC,CS          ; clear the chip select (active)
0011 3006      00044          movlw 0x06          ; load WREN sequence
0012 00A4      00045          movwf outbyte          ; store in RAM location outbyte
0013 205F      00046          call output          ; call the SPI output routine
0014 1787      00047          bsf PORTC,CS          ; set the chip select line
00048 ;Send the write status register sequence (WRSR)
0015 1387      00049          bcf PORTC,CS          ; clear the chip select line

```

```

0016 3001      00050      movlw    0x01      ; clear all status register
0017 00A4      00051      movwf    outbyte   ; store in RAM location outbyte
0018 205F      00052      call     output   ; call the SPI output routine
0019 3000      00053      movlw    0x00      ; load up zero to send
001A 00A4      00054      movwf    outbyte   ; store in RAM location outbyte
001B 205F      00055      call     output   ; call the SPI output routine
001C 1787      00056      bsf      PORTC,CS   ; set the chip select line
                00057 ;Wait the required 5mS for the write cycle timer Twc
001D 2068      00058      call     delay    ; call the delay subroutine
                00059 ;Send the write enable sequence (WREN)
001E 1387      00060      bcf      PORTC,CS   ; clear the chip select (active)
001F 3006      00061      movlw    0x06      ; load WREN sequence
0020 00A4      00062      movwf    outbyte   ; store in RAM location outbyte
0021 205F      00063      call     output   ; call the SPI output routine
0022 1787      00064      bsf      PORTC,CS   ; set the chip select line
                00065 ;Send the read status register sequence (RDSR)
0023 1387      00066      bcf      PORTC,CS   ; clear the chip select line
0024 3005      00067      movlw    0x05      ; load RDSR sequence
0025 00A4      00068      movwf    outbyte   ; store in RAM location outbyte
0026 205F      00069      call     output   ; call the SPI output routine
0027 205F      00070      call     output   ; read the data in status reg.
0028 1787      00071      bsf      PORTC,CS   ; set the chip select line
                00072 ;Send the write sequence (WRITE)
0029 1387      00073      bcf      PORTC,CS   ; clear the chip select line
002A 3002      00074      movlw    0x02      ; load WRITE sequence
002B 00A4      00075      movwf    outbyte   ; store in RAM location outbyte
002C 205F      00076      call     output   ; call the SPI output routine
                00077 ;****Comment out for use with 25xx010, 25xx020 or 25xx040*****
002D 3000      00078      movlw    0x00      ; load high address byte
002E 00A4      00079      movwf    outbyte   ; store in RAM location outbyte
002F 205F      00080      call     output   ; call the SPI output routine
                00081 ;*****
0030 0822      00082      movf     addr,W     ; move the address into w
0031 00A4      00083      movwf    outbyte   ; store in RAM location outbyte
0032 205F      00084      call     output   ; call the SPI output routine
0033 30AA      00085      movlw    0xAA      ; load data AA into w
0034 00A4      00086      movwf    outbyte   ; store in RAM location outbyte
0035 205F      00087      call     output   ; call the SPI output routine
0036 1787      00088      bsf      PORTC,CS   ; set the chip select line
                00089 ;Perform data polling (RDSR bit 0)
0037 1387      00090      bcf      PORTC,CS   ; clear the chip select line
0038 3005      00091      movlw    0x05      ; load RDSR sequence
0039 00A4      00092      movwf    outbyte   ; store in RAM location outbyte
003A 205F      00093      call     output   ; call the SPI output routine
                00094 ;
003B 3030      00095      movlw    0x30      ; give the spi device time to
003C 00A3      00096      movwf    loops     ; set the WIP bit in the
Message[305]: Using default destination of 1 (file).
003D 0BA3      00097 wait    decfsz    loops     ; status register before coming
003E 283D      00098      goto     wait     ; back and doing data polling
                00099 ;
003F 205F      00100 polling call    output   ; read the data in status reg.
0040 1820      00101      btfsc    rxdata,0   ; test the WIP bit in status reg.
0041 283F      00102      goto     polling  ; WIP clear, loop until WIP set
0042 1787      00103      bsf      PORTC,CS   ; set the chip select line
                00104 ;Send read sequence (READ), read address 0x55
0043 1387      00105      bcf      PORTC,CS   ; clear the chip select line
0044 3003      00106      movlw    0x03      ; load READ sequence
0045 00A4      00107      movwf    outbyte   ; store in RAM location outbyte
0046 205F      00108      call     output   ; call the SPI output routine
                00109 ;****Comment out for use with 25xx010, 25xx020 or 25xx040*****
0047 3000      00110      movlw    0x00      ; load high address byte
0048 00A4      00111      movwf    outbyte   ; store in RAM location outbyte
0049 205F      00112      call     output   ; call the SPI output routine
                00113 ;*****
004A 0822      00114      movf     addr,W     ; move the address into w

```

```

004B 00A4      00115      movwf    outbyte      ; store in RAM location outbyte
004C 205F      00116      call     output      ; call the SPI output routine
004D 205F      00117      call     output      ; call output to read 1 byte
004E 1787      00118      bsf      PORTC,CS      ; set the chip select line
                00119      ;Send the write enable sequence (WREN)
004F 1387      00120      bcf      PORTC,CS      ; clear the chip select (active)
0050 3006      00121      movlw    0x06          ; load WREN sequence
0051 00A4      00122      movwf    outbyte      ; store in RAM location outbyte
0052 205F      00123      call     output      ; call the SPI output routine
0053 1787      00124      bsf      PORTC,CS      ; set the chip select line
                00125      ;Send the write disable sequence (WRDI)
0054 1387      00126      bcf      PORTC,CS      ; clear the chip select (active)
0055 3004      00127      movlw    0x04          ; load WRDI sequence
0056 00A4      00128      movwf    outbyte      ; store in RAM location outbyte
0057 205F      00129      call     output      ; call the SPI output routine
0058 1787      00130      bsf      PORTC,CS      ; set the chip select line
                00131      ;Send the read status register sequence (RDSR)
0059 1387      00132      bcf      PORTC,CS      ; clear the chip select line
005A 3005      00133      movlw    0x05          ; load RDSR sequence
005B 00A4      00134      movwf    outbyte      ; store in RAM location outbyte
005C 205F      00135      call     output      ; call the SPI output routine
005D 205F      00136      call     output      ; read the data in status reg.
                00137      ;*****Go back to main routine*****
005E 2801      00138      goto     start
                00139
                00140      ;*****SPI output subroutine*****
005F 0824      00141      output  movf      outbyte,W      ; move outbyte into w
0060 0093      00142      movwf    SSPBUF      ; place data in send buffer
0061 1683      00143      loop1   bsf      STATUS,RP0      ; set to bank 1
Message[302]: Register in operand not in bank 0. Ensure that bank bits are correct.
0062 1C14      00144      btfss    SSPSTAT,BF      ; has data been received?
0063 2861      00145      goto     loop1      ; loop if not received yet
0064 1283      00146      bcf      STATUS,RP0      ; set to bank 0
0065 0813      00147      movf      SSPBUF,W      ; empty the receive buffer
0066 00A0      00148      movwf    rxdata      ; put received byte into rxdata
0067 3400      00149      retlw    0          ; return from subroutine
                00150
                00151      ; 250 x 400nS x 50 = 5mS (plus overhead)
0068 3032      00152      delay   movlw    0x32          ; move 50 decimal into w
0069 00A5      00153      movwf    temp1      ; move 50 decimal into temp1
006A 30FA      00154      dec1     movlw    0xFA          ; move 250 decimal into w
006B 00A6      00155      movwf    temp2      ; move 250 decimal into temp2
006C 0BA6      00156      dec2     decfsz   temp2,1      ; decrement temp2, skip if zero
006D 286C      00157      goto     dec2      ; goto decrement 2 if not zero
006E 0BA5      00158      decfsz   temp1,1      ; decrement temp1, skip if zero
006F 286A      00159      goto     dec1      ; goto decrement 1 if not zero
0070 3400      00160      retlw    0          ; return both locations = 0
                00161
                00162
                00163      END

```

MEMORY USAGE MAP ('X' = Used, '-' = Unused)

```

0000 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
0040 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX X-----

```

All other memory blocks unused.

Program Memory Words Used: 113
Program Memory Words Free: 3983

Errors : 0
Warnings : 0 reported, 0 suppressed
Messages : 4 reported, 0 suppressed

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: 508-480-9990 Fax: 508-480-8575

Chicago

Microchip Technology Inc.
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

Dallas

Microchip Technology Inc.
14651 Dallas Parkway, Suite 816
Dallas, TX 75240-8809
Tel: 972-991-7177 Fax: 972-991-8588

Dayton

Microchip Technology Inc.
Two Prestige Place, Suite 150
Miamisburg, OH 45342
Tel: 937-291-1654 Fax: 937-291-9175

Los Angeles

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

New York

Microchip Technology 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: 905-405-6279 Fax: 905-405-6253

ASIA/PACIFIC

Hong Kong

Microchip Asia Pacific
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 India
No. 6, Legacy, Convent Road
Bangalore 560 025, India
Tel: 91-80-229-0061 Fax: 91-80-229-0062

Korea

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

Shanghai

Microchip Technology
RM 406 Shanghai Golden Bridge Bldg.
2077 Yan'an Road West, Hongiao District
Shanghai, PRC 200335
Tel: 86-21-6275-5700
Fax: 86 21-6275-5060

Singapore

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

Taiwan, R.O.C

Microchip Technology Taiwan
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: 44-1628-851077 Fax: 44-1628-850259

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 München, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

Italy

Arizona Microchip Technology SRL
Centro Direzionale Colleone
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-39-6899939 Fax: 39-39-6899883

JAPAN

Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shin Yokohama
Kohoku-Ku, Yokohama
Kanagawa 222 Japan
Tel: 81-4-5471- 6166 Fax: 81-4-5471-6122

5/8/97



MICROCHIP

All rights reserved. © 1997, Microchip Technology Incorporated, USA. 5/97

Information contained in this publication regarding device applications and the like is intended for suggestion only and may be superseded by updates. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights. The Microchip logo and name are registered trademarks of Microchip Technology Inc. in the U.S.A. and other countries. All rights reserved. All other trademarks mentioned herein are the property of their respective companies.
