INTERFACING MICROCHIP SERIAL EEPROMS TO THE INTEL 8051 FAMILY OF MICROCONTROLLERS

Many designers today are implementing embedded systems that require low cost non-volatile memory. Microchip has addressed this need with a full line of serial EEPROMs, in a variety of memory configurations, using the industry-standard 2 or 3-wire communication protocols. The theory and application of these protocols are addressed in detail in Microchip's application note AN536.

Microchip recognizes that its broad customer base uses a variety of micro-controllers; many firmware related questions have been asked concerning interfacing the 8051 family and its derivatives.

The purpose of this app note is to provide assembly language examples of 8051 code for the various serial EEPROMs available from Microchip. These routines are intended to provide the basic operating kernels for storing data to or retrieving data from a serial EEPROM.

All of the routines in this application note are available, as source code, for downloading from Microchip’s worldwide website at www.microchip.com. The file to download is 00614.zip.

This app note covers all of Microchip’s 2-wire serial devices. Note that some devices have features not supported in others, and therefore, some sections of the code presented here may not be applicable to a particular part. We have attempted to label those special sections to minimize confusion.

The code includes a simple loop-type shell to enable it to be executed (with an emulator) without the user having to write any other routines. The various address and data pointers must be set to the desired values by hand, before each execution cycle.

TIMING DATA

Clock and data timing is accomplished by software. There are two sets of timing specifications: 100 kHz and 400 kHz. Assuming a 12 MHz 8051 clock, extra NOP's have been added to slow timing down to 100 kHz. See Note 1 in the listing. If a 16 MHz clock is used, additional NOP's are required for 100 kHz operation. See Note 2 in the listing. For 400 kHz operation, the NOP's labelled Note 1 or Note 2 are not needed. If not needed the NOP's may be left out.

Below is the connection diagram used for this app note. Do not forget the pull-up resistor!

WIRING DIAGRAM

* This pin is NC in some parts
APPENDIX A: SOURCE CODE

1  $MOD51
2  ;REGISTER ASSIGNMENTS
3  ;
4  ;R1 DATA OR DATA POINTER
5  ;R2 LOOP COUNTER REGISTER
6  ;R3 ADDRESS, HI BYTE
7  ;R4 ADDRESS, LOW BYTE
8  ;R5
9  ;R6 BYTE COUNT FOR PAGE OPERATIONS
10  ;PIN ASSIGNMENTS:
11  ;Port 1 bit 0 is data
12  ;Port 1 bit 1 is clock
13  ;
14  ;These routines assume chip address = 0
15  ;
16  ;The oscillator frequency assumed for this app note is 12 MHz
17  ;
18  ;These routines use software timing loops. They may have to be
19  ;adjusted if a different oscillator frequency is used.
20  ;
21  ;NOTE 1 These NOP's added for timing delays on 'C' parts, OR 'LC' parts
22  ;       where Vcc < 4.5V and the oscillator frequency is 12 MHz.
23  ;       This allows a bit rate of 100kHz.
24  ;NOTE 2 Use these NOP's with a 16MHz oscillator and 100kHz bit rate.
25  ;       For 400kHz bit rate, the NOP's in Note 1 and Note 2 are not req'd.
26  ;
27  ;The EEPROM will be busy after a write cycle is initiated
28  ;for up to 10mS per page (or per byte if a byte write). This app note
29  ;assumes the user will delay the appropriate time after write, or check
30  ;for Busy status. A subroutine is provided to check the Busy Status.
31  ;
32  ;RAM DEFINITIONS
33  ---
34  DSEG
35  ORG 30H
36  BYTSTR: DS 20H ;STORAGE FOR READ DATA
37  ;
38  ;CONSTANTS -- REDEFINE AS NECESSARY
39  ;
40  WTCMD EQU     10100000B ;WRITE DATA COMMAND Note 3
41  RDCMD  EQU     10100001B   ;READ DATA COMMAND Note 3
42  RDEND EQU     01000000B   ;READ HIGH-ENDURANCE BLOCK NUMBER COMMAND
43  ADDRH EQU     0
44  ADDRL EQU     0
45  DTA EQU     55H
46  BYTCNTEQU  8
47  ;
48  ;Note 3 Some chip or byte address bits are embedded in the control byte.
49  ;Refer to the data sheet for exact configuration, which varies from part
50  ;to part.
51  ;
52  ;************************************************************************
53  ; This section contains test loop routines. They form a simple operating
54  ; shell to allow the 2-wire interface code to be tested in a stand-alone
55  ; mode. Using an emulator, change "NONE" to one of the four listed
56  ; routines to test that function. The address and data constants can
57  ; also be set as desired.
58  ; If using a 32Kbit or higher density serial EEPROM, change the called
59  ; routines by adding 'L' to the end of the name. This is required
60  ; because these parts use TWO address bytes. The 'L' routines send out
61  ; the extra address byte.
62  ;************************************************************************
--- 63 CSEG
0000 64 ORG 0
0000 020003 65 JMP START
0003 7590FF 66 START: MOV P1,#0FFH ;INIT PORT1
0006 12000B 67 CALL NONE ;TEST LOOP INSERT PROPER ADDRESS HERE
0009 80F8 68 JMP START
69
000B 22 70 NONE: RET
71 ;*
72 ;* WRITE ONE BYTE TO EEPROM
73 ;* The Address Pointer is the address in the EEPROM. The byte to be sent
74 ;* to the EEPROM is stored in the constant 'DTA'
75 ;*
000C 7B00 76 TESTWR: MOV R3,#ADDRH ;LOAD ADDRESS POINTER FOR HIGH DENSITY ONLY
000E 7C00 77 MOV R4,#ADDRL ;LOAD ADDRESS POINTER FOR ALL DEVICES
0010 7955 78 MOV R1,#DTA ;LOAD DATA BYTE
0012 120037 79 CALL BYTWR ;CALL PAGE WRITE ROUTINE
0015 22 80 RET
81
82 ;*
83 ;* WRITE A BLOCK OF DATA TO EEPROM
84 ;* The address pointer is the address in EEPROM where data will start.
85 ;* The byte pointer is the starting address of RAM containing the block
86 ;* of data to be sent. The byte count indicates how many bytes to send to
87 ;* the EEPROM. The number of bytes that can be sent before a STOP command
88 ;* is issued is dependent on EEPROM type. Refer to the data book for
89 ;* specific values.
90 ;*
0016 7B00 91 BLKWR: MOV R3,#ADDRH ;LOAD ADDRESS POINTER FOR HIGH DENSITY ONLY
0018 7C00 92 MOV R4,#ADDRL ;LOAD ADDRESS POINTER FOR ALL DEVICES
001A 7930 93 MOV R1,#BYTSTR ;LOAD BYTE POINTER
001C 7E08 94 MOV R6,#BYTCNT ;LOAD BYTE COUNT
001E 120048 95 CALL PAGEW ;CALL PAGE WRITE ROUTINE
0021 22 96 RET
96
97
98 ;*
99 ;* READ ONE BYTE FROM EEPROM
100 ;* The address pointer is the address of the desired byte in EEPROM.
101 ;* The byte will be returned in R1.
102 ;*
0022 7B00 103 TESTRD: MOV R3,#ADDRH ;LOAD ADDRESS POINTER FOR HIGH DENSITY ONLY
0024 7C00 104 MOV R4,#ADDRL ;LOAD ADDRESS POINTER FOR ALL DEVICES
0026 12000A 105 CALL BYTERD ;CALL BYTE READ ROUTINE.
0029 F9 106 MOV R1,A ;SAVE THE BYTE
002A 22 107 RET
107
108
109 ;*
110 ;* READ A BLOCK FROM EEPROM
111 ;* The address pointer is the starting address of the desired data block
112 ;* in EEPROM. The data pointer is the starting address in RAM where data
113 ;* will be stored. The byte count indicates how many bytes should be read
114 ;* The entire EEPROM may be read with one READ command this way.
115 ;*
002B 7B00 116 BLOKRD: MOV R3,#ADDRH ;LOAD ADDR POINTER FOR HIGH DENSITY ONLY
002D 7C00 117 MOV R4,#ADDRL ;LOAD ADDRESS POINTER FOR ALL DEVICES
002F 7930 118 MOV R1,#BYTSTR ;LOAD DATA POINTER
0031 7E0A 119 MOV R6,#BYTCNT ;LOAD BYTE COUNT
0033 12005C 120 CALL BLOCK ;CALL BLOCK READ ROUTINE
0036 22 121 RET
122
123 ;* END OF TEST LOOP
124 ;**************************************************************************************************
125
126
127 ;**************************************************************************************************
128 ; This routine writes a byte of data to EEPROM
; The EEPROM address is assumed to be in R4. See NOTE 3
130; The DATA to be written is assumed to be in R1
131;*******************************************************************
0037 74A0 132 BYTEW: MOV A,#WTCMD ;LOAD WRITE COMMAND
0039 12012F 133 CALL OUTS ;SEND IT
003C EC 134 MOV A,R4 ;GET BYTE ADDRESS
003D 120167 135 CALL OUT ;SEND IT
0040 E9 136 MOV A,R1 ;GET DATA
0041 120167 137 CALL OUT ;SEND IT
0044 1201A6 138 CALL STOP ;SEND STOP CONDITION
0047 22 139 RET
140
141;*******************************************************************
142; THIS ROUTINE WRITES A PAGE OF DATA TO EEPROM
143; The EEPROM start address is assumed to be in R4. See NOTE 3
144; The DATA pointer is in R1
145; The BYTE count is in R6
146; The number of bytes that can be transferred depends upon the
147; EEPROM used
148;*******************************************************************
0048 74A0 149 PAGEW: MOV A,#WTCMD ;LOAD WRITE COMMAND
004A 12012F 150 CALL OUTS ;SEND IT
004D EC 151 MOV A,R4 ;GET LOW BYTE ADDRESS
004E 120167 152 CALL OUT ;SEND IT
0051 E7 153 BTLP: MOV A,@R1 ;GET DATA
0052 120167 154 CALL OUT ;SEND IT
0055 09 155 INC R1 ;INCREMENT DATA POINTER
0056 DEF9 156 DJNZ R6,BTLP ;LOOP TILL DONE
0058 1201A6 157 CALL STOP ;SEND STOP CONDITION
005B 22 158 RET
159
160
161;*******************************************************************
162; THIS ROUTINE READS A BLOCK OF DATA FROM EEPROM AT A SPECIFIED ADDRESS
163; EEPROM address in R4. See NOTE 3.
164; Stores data at RAM location pointed to by R1
165; Byte count specified in R6. May be 1 to 256 bytes
166;*******************************************************************
005C 74A0 167 BLKRD: MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS
005E 12012F 168 CALL OUTS ;SEND IT
0061 EC 169 MOV A,R4 ;GET LOW BYTE ADDRESS
0062 120167 170 CALL OUT ;SEND IT
0065 74A1 171 MOV A,#RDCMD ;LOAD READ COMMAND
0067 12012F 172 CALL OUTS ;SEND IT
006A 12018D 173 BRDLP: CALL IN ;READ DATA
006D F7 174 MOV @R1,a ;STORE DATA
006E 09 175 INC R1 ;INCREMENT DATA POINTER
006F DE04 176 DJNZ R6,AKLP ;DECREMENT LOOP COUNTER
0071 1201A6 177 CALL STOP ;IF DONE, ISSUE STOP CONDITION
0074 22 178 RET ;DONE, EXIT ROUTINE
179
0075 C290 180 AKLP: CLR P1.0 ;NOT DONE, ISSUE ACK
0077 D291 181 SETB P1.1
0079 00 182 NOP ;NOTE 1
007A 00 183 NOP
007B 00 184 NOP
007C 00 185 NOP ;NOTE 2
007D 00 186 NOP
007E C291 187 CLR P1.1
0080 80E8 188 JMP BRDLP ;CONTINUE WITH READS
189
190;*******************************************************************
191; THIS ROUTINE READS A BYTE OF DATA FROM THE EEPROM
192; The EEPROM address is in R4. See Note 3
193; Returns the data byte in R1
194;*******************************************************************
AN614

0082 74A0 195 BYTERD: MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS
0084 12012F 196 CALL OUTS ;SEND IT
0087 EC 197 MOV A,R4 ;GET LOW BYTE ADDRESS
0088 120167 198 CALL OUT ;SEND IT
008B 12008F 199 CALL CREAD ;GET DATA BYTE
008E 22 200 RET

008F 74A1 207 CREAD: MOV A,#RDCMD ;LOAD READ COMMAND
0091 12012F 208 CALL OUTS ;SEND IT
0094 12018D 209 CALL IN ;READ DATA
0097 F9 210 MOV R1,A ;STORE DATA
0098 1201A6 211 CALL STOP ;SEND STOP CONDITION
009B 22 212 RET

009C 74A0 216 BLKRDL: MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS
009E 12012F 217 CALL OUTS ;SEND IT
00A1 EB 218 MOV A,R3 ;GET HI BYTE ADDRESS
00A2 120167 219 CALL OUT ;SEND IT
00A5 EC 220 MOV A,R4 ;GET LO BYTE ADDRESS
00A6 120167 221 CALL OUT ;SEND IT
00A9 74A1 222 MOV A,#RDCMD ;LOAD READ COMMAND
00AB 12012F 223 CALL OUTS ;SEND IT
00AE 80BA 224 JMP BRDLP ;CONTINUE WITH DATA READ

00B0 74A0 225 BYTEWL: MOV A,#WTCMD ;LOAD WRITE COMMAND
00B2 12012F 226 CALL OUTS ;SEND IT
00B5 EB 227 MOV A,R3 ;GET HI BYTE ADDRESS
00B6 120167 228 CALL OUT ;SEND IT
00B9 EC 229 MOV A,R4 ;GET LO BYTE ADDRESS
00BA 120167 230 CALL OUT ;SEND IT
00BD E9 231 MOV A,R1 ;GET DATA
00BE 120167 232 CALL OUT ;SEND IT
00C1 1201A6 233 CALL STOP ;SEND STOP CONDITION
00C4 22 234 RET

00C6 235 ;******************************************************************************
00C8 236 ;THIS ROUTINE READS A BYTE OF DATA FROM EEPROM
00CA 237 ;From EEPROM current address pointer.
00CB 238 ;Returns the data byte in R1
00CD 239 ;******************************************************************************
00CE 74A1 240 CREAD: MOV A,#RDCMD ;LOAD READ COMMAND
00D0 12012F 241 CALL OUTS ;SEND IT
00D3 12018D 242 CALL IN ;READ DATA
00D5 F9 243 MOV R1,A ;STORE DATA
00D6 1201A6 244 CALL STOP ;SEND STOP CONDITION
00D9 22 245 RET

00E1 246 ;***********************************************************************
00E3 247 ;This routine writes a byte of data to EEPROM
00E5 248 ;This routine is for the 24LC32 or 24LC65
00E7 249 ;The EEPROM address is assumed to be in R3:R4
00E9 250 ;The DATA to be written is assumed to be in R1
00EB 251 ;******************************************************************************
00EC 74A0 252 BYTEWL: MOV A,#WTCMD ;LOAD WRITE COMMAND
00ED 12012F 253 CALL OUTS ;SEND IT
00F1 EB 254 MOV A,R3 ;GET HI BYTE ADDRESS
00F2 120167 255 CALL OUT ;SEND IT
00F5 EC 256 MOV A,R4 ;GET LO BYTE ADDRESS
00F6 120167 257 CALL OUT ;SEND IT
00F9 E9 258 MOV A,R1 ;GET DATA
00FA 120167 259 CALL OUT ;SEND IT
00FC 1201A6 260 CALL STOP ;SEND STOP CONDITION
00FF 22 261 RET

0101 262 ;******************************************************************************
0103 263 ;This routine writes a page of data to EEPROM
0105 264 ;This routine is for the 24LC32 or 24LC65
0107 265 ;The EEPROM start address is assumed to be in R3:R4
0109 266 ;The DATA pointer is in R1
010B 267 ;The BYTE count is in R6
010D 268 ;The number of bytes that can be transfered depends on the
010F 269 ;EEPROM in use
0111 270 ;******************************************************************************

© 1998 Microchip Technology Inc.
00C5 74A0  262  PAGEWL: MOV A,#WTCMD ;LOAD WRITE COMMAND
00C7 12012F 263 CALL OUTS ;SEND IT
00CA EB  264 MOV A,R3 ;GET HIYTE ADDRESS
00CB 120167 265 CALL OUT ;SEND IT
00CE EC  266 MOV A,R4 ;GET LOW BYTE ADDRESS
00CF 120167 267 CALL OUT ;SEND IT
00D2 E7  268 BTLPL: MOV A,@R1 ;GET DATA
00D3 1201A6 272 CALL STOP ;SEND STOP CONDITION
00D9 120167 272 CALL OUT ;SEND IT
00DC 22  273 RET

274 ;*******************************************************************************
275 ; THIS ROUTINE READS A BYTE OF DATA FROM THE EEPROM
277 ; This routine is for the 24CL32 or 24LC65
278 ; The EEPROM address is in R3:R4
279 ; Returns the data byte in R1
280 ;*******************************************************************************
00DD 281 BYTERDL:
00DD 74A0  282 MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS
00DF 12012F 283 CALL OUTS ;SEND IT
00E2 EB  284 MOV A,R3 ;GET HI BYTE ADDRESS
00E3 120167 285 CALL OUT ;SEND IT
00E6 EC  286 MOV A,R4 ;GET LOW BYTE ADDRESS
00E7 120167  287 CALL OUT ;SEND IT
00EA 118F  288 CALL CREAD ;GET DATA BYTE
00EC 22  289 RET

290 ; SUBROUTINES
292 ;*******************************************************************************
294 ; This routine test for WRITE DONE condition
297 ; by testing for an ACK.
299 ; This routine can be run as soon as a STOP condition
300 ; has been generated after the last data byte has been sent
302 ; to the EEPROM. The routine loops until an ACK is received from
304 ; the EEPROM. No ACK will be received until the EEPROM is done with
306 ; the write operation.
308 ;*******************************************************************************
00ED 74A0  302 ACKTST: MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS
00EF 7A08  303 R2,#8 ;LOOP COUNT -- EQUAL TO BIT COUNT
00F3 00  304 CLR P1.0 ;START CONDITION -- DATA = 0
00F4 00  305 NOP ;NOTE 1
00F5 00  306 NOP
00F6 00  307 NOP
00F7 00  308 NOP ;NOTE 2
00F8 C291  309 CLR P1.1 ;CLOCK = 0
00FA 33  310 AKTLPS: RLC A ;SHIFT BIT
00FB 5005 312 JNC AKTLS
00FD D290  313 SETB P1.0 ;DATA = 1
00FF 020104 314 JMP AKTL1 ;CONTINUE
0102 C290  315 AKTLS: CLR P1.0 ;DATA = 0
0104 D291  316 AKTL1: SETB P1.1 ;CLOCK HI
0106 00  317 NOP ;NOTE 1
0107 00  318 NOP
0108 00  319 NOP
0109 00  320 NOP ;NOTE 2
010A 00  321 NOP
010B C291  322 CLR P1.1 ;CLOCK LOW
010D D290  323 DJNZ R2,AKTLPS ;DECREMENT COUNTER
010F D290  324 SETB P1.0 ;TURN PIN INTO INPUT
0111 00  325 NOP ;NOTE 1
0112 00  326 NOP ;NOTE 2
0113 D291  327 SETB P1.1 ;CLOCK ACK
0115 00 328 NOP ;NOTE 1
0116 00 329 NOP
0117 00 330 NOP
0118 00 331 NOP ;NOTE 2
0119 00 332 NOP
011A 309002 333 JNB P1.0,EXIT ;EXIT IF ACK (WRITE DONE)
011D 80CE 334 JMP ACKTST ;START OVER
011F C291 335 EXIT: CLR P1.1 ;CLOCK LOW
0121 C290 336 CLR P1.0 ;DATA LOW
0123 00 337 NOP ;NOTE 1
0124 00 338 NOP
0125 00 339 NOP
0126 00 340 NOP ;NOTE 2
0127 00 341 NOP
0128 D291 342 SETB P1.1 ;CLOCK HIGH
012A 00 343 NOP
012B 00 344 NOP
012C D290 345 SETB P1.0 ;STOP CONDITION
012E 22 346 RET

***********************************************************************
THIS ROUTINE SENDS OUT CONTENTS OF THE ACCUMULATOR TO THE EEPROM AND INCLUDES START CONDITION. REFER TO THE DATA SHEETS FOR DISCUSSION OF START AND STOP CONDITIONS.
***********************************************************************

012F 7A08 356 OUTS: MOV R2,#8 ;LOOP COUNT -- EQUAL TO BIT COUNT
0131 D290 357 SETB P1.0 ;INSURE DATA IS HI
0133 D291 358 SETB P1.1 ;INSURE CLOCK IS HI
0135 00 359 NOP ;NOTE 1
0136 00 360 NOP
0137 00 361 NOP
0138 00 362 NOP ;NOTE 2
0139 00 363 NOP
013A C290 364 CLR P1.0 ;START CONDITION -- DATA = 0
013C 00 365 NOP ;NOTE 1
013D 00 366 NOP
013E 00 367 NOP
013F 00 368 NOP ;NOTE 2
0140 00 369 NOP
0141 C291 370 CLR P1.1 ;CLOCK = 0
0143 33 371 OTSLP: RLC A ;SHIFT BIT
0144 5005 372 JNC BITLS
0146 D290 373 SETB P1.0 ;DATA = 1
0148 02814D 374 JMP OTSL1 ;CONTINUE
014B C290 375 BITLS: CLR P1.0 ;DATA = 0
014D D291 376 OTSL1: SETB P1.1 ;CLOCK HI
014F 00 377 NOP ;NOTE 1
0150 00 378 NOP
0151 00 379 NOP
0152 00 380 NOP ;NOTE 2
0153 00 381 NOP
0154 C291 382 CLR P1.1 ;CLOCK LOW
0156 DAEB 383 DJNZ R2,OTSLP ;DECREMENT COUNTER
0158 D290 384 SETB P1.0 ;TURN PIN INTO INPUT
015A 00 385 NOP ;NOTE 1
015B 00 386 NOP ;NOTE 2
015C 00 387 NOP
015D D291 388 SETB P1.1 ;CLOCK ACK
015F 00 389 NOP ;NOTE 1
0160 00 390 NOP
0161 00 391 NOP
0162 00 392 NOP ;NOTE 2
0163 00 393 NOP
0164 C291 394 CLR P1.1
;**********************************************************************
; THIS ROUTINE SENDS OUT CONTENTS OF ACCUMULATOR TO EEPROM
; without sending a START condition.
;**********************************************************************

0167 7A08 402 OUT:        MOV     R2,#8            ;LOOP COUNT -- EQUAL TO BIT COUNT
0169 33 403 OTLP:      RLC     A               ;SHIFT BIT
016A 5005 404 JNC     BITL
016C D290 405 SETB    P1.0            ;DATA = 1
016E 020173 406 JMP     OTL1            ;CONTINUE
0171 C290 407 BITL:      CLR     P1.0            ;DATA = 0
0173 D291 408 OTL1:      SETB    P1.1            ;CLOCK HI
0175 00 409 NOP            ;NOTE 1
0176 00 410 NOP
0177 00 411 NOP
0178 00 412 NOP            ;NOTE 2
0179 00 413 NOP
017A C291 414 CLR     P1.1            ;CLOCK LOW
017C DAEB 415 DJNZ    R2,OTLP            ;DECREMENT COUNTER
017E D290 416 SETB    P1.0            ;TURN PIN INTO INPUT
0180 00 417 NOP            ;NOTE 1
0181 00 418 NOP            ;NOTE 2
0182 00 419 NOP
0183 D291 420 SETB    P1.1            ;CLOCK ACK
0185 00 421 NOP            ;NOTE 1
0186 00 422 NOP
0187 00 423 NOP
0188 00 424 NOP            ;NOTE 2
0189 00 425 NOP
018A C291 426 CLR     P1.1
018C 22 427 RET

;**********************************************************************
; THIS ROUTINE READS IN A BYTE FROM THE EEPROM
; and stores it in the accumulator
;**********************************************************************

018D 7A08 434 IN: MOV     R2,#8           ;LOOP COUNT
018F D290 435 SETB    P1.0            ;SET DATA BIT HIGH FOR INPUT
0191 C291 436 INLP:    CLR     P1.1            ;CLOCK LOW
0193 00 437 NOP            ;NOTE 1
0194 00 438 NOP
0195 00 439 NOP
0196 00 440 NOP
0197 00 441 NOP            ;NOTE 2
0198 00 442 NOP
0199 D291 443 SETB    P1.1            ;CLOCK HIGH
019B C3 444 CLR     C            ;CLEAR CARRY
019C 309001 445 JNB     P1.0,INL1            ;JUMP IF DATA = 0
019F B3 446 CPL     C            ;SET CARRY IF DATA = 1
01A0 33 447 INL1:    RLC     A            ;ROTATE DATA INTO ACCUMULATOR
01A1 DAEB 448 DJNZ    R2,INLP            ;DECREMENT COUNTER
01A3 C291 449 CLR     P1.1            ;CLOCK LOW
01A5 22 450 RET

01A6 C290 453 STOP:    CLR     P1.0            ;STOP CONDITION SET DATA LOW
01A8 00 454 NOP            ;NOTE 1
01A9 00 455 NOP
01AA 00 456 NOP
01AB 00 457 NOP            ;NOTE 2
01AC 00 458 NOP
01AD D291 459 SETB    P1.1            ;SET CLOCK HI
01AF 00 460 NOP            ;NOTE 1
**These routines contain special commands for the 24LC65 SMART SERIAL EEPROM**

**SET SECURE BLOCK ASSUMES**

**START BLOCK 0 & BLOCK LENGTH OF 1. The numbers are implicit in the commands. Refer to the data sheet for details.**

**SET SECURE BLOCK NUMBER(S)**: Returns block number in R1 and blockcount in R2 (upper nibbles will be 1's)

**SET HIGH-ENDURANCE BLOCK NUMBER**: Assumes block 0
527 ;*
528 ; READ HIGH ENDURANCE BLOCK NUMBER
529 ; RETURNS BLOCK NUMBER IN R1 (UPPER NIBBLE WILL BE 1'S)
530 ;*
01F2 7B80 531 READHI: MOV R3,#080H ;LOAD COMMAND
01F4 7C00 532 MOV R4,#0
01F6 1201FA 533 CALL HIEND ;EXECUTE
01F9 22 534 RET

01FA 74A0 536 HIEND: MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS
01FC 312F 537 CALL OUTS ;SEND IT
01FE EB 538 MOV A,R3 ;GET HI BYTE ADDRESS
01FF 3167 539 CALL OUT ;SEND IT
0201 EC 540 MOV A,R4 ;GET LOW BYTE ADDRESS
0202 3167 541 CALL OUT ;SEND IT
0204 7440 542 MOV A,#RDEND ;LOAD READ COMMAND
0206 3167 543 CALL OUT ;SEND IT
0208 318D 544 CALL IN ;READ DATA
020A F9 545 MOV R1,A ;STORE DATA
020B 31A6 546 CALL STOP ;SEND STOP CONDITION
020D 22 547 RET

548 ;END of 24LC65 Routines
549 ;******************************************************
550 ;END
551 VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND
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.

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. 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.

Trademarks

The Microchip name and logo, the Microchip logo, FilterLab, KEELOQ, microID, MPLAB, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

dsPIC, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, MXDEV, PICC, PICDEM, PICDEM.net, rIPIC, Select Mode and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Turn Programming (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.

© 2002, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.
### 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: 770-640-0075

**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: 972-818-792 Fax: 972-818-2924

**Detroit**  
Tri-Axia 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 92812  
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  
Beijing, 100027, No. China  
Tel: 86-10-85282100 Fax: 86-10-85282104

**China - Chengdu**  
Microchip Technology Consulting (Shanghai) Co., Ltd., Chengdu Liaison Office  
Rm. 3401, 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  
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 Xian Road  
Shanghai, 200051, China  
Tel: 86-21-6275-5070 Fax: 86-21-6275-5060

**China - Shenzhen**  
Microchip Technology Consulting (Shanghai) Co., Ltd., Shenzhen Liaison Office  
Rm. 1301, 13/F, Shenzhen Kerry Centre, Renminnan Lu  
Shenzhen 518001, China  
Tel: 86-755-2366086 Fax: 86-755-2366086

**China - 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

**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, 189880  
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  
Lastrup høj 1-3  
Ballrup 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 – 1er Etage  
91300 Massy, France  
Tel: 33-1-99-53-63-20 Fax: 33-1-99-30-90-79

**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  
Windersh Triangle  
Wokingham  
Berkshire, England RG41 5TU  
Tel: 44 118 921 5869 Fax: 44-118 921-5820

01/18/02