

Interfacing the 8051 with 2-wire Serial EEPROMs

Author: Mike Rosenfield
Memory Products Division

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 app note are available, as source code, for downloading from Microchip's BBS. Information on the BBS is available elsewhere in the Embedded Control Handbook. The file to download is AN61437.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.

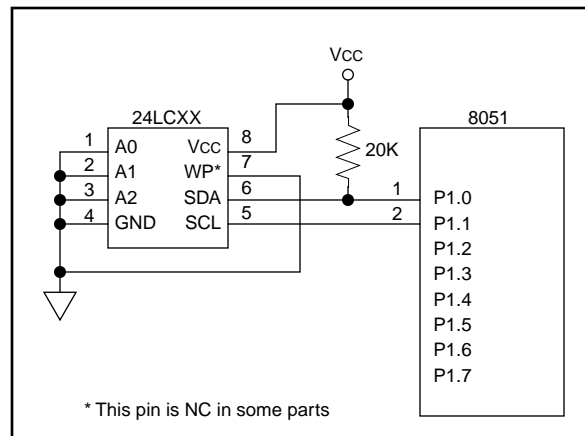
The code includes a simple loop-type shell to enable it to be executed (with an emulator) without the user having to write any other routines. The various address and data pointers must be set to the desired values by hand, before each execution cycle.

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, 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



Please check the Microchip BBS for the latest version of the source code. For BBS access information, see Section 6, Microchip Bulletin Board Service information, page 6-3.

APPENDIX A: SOURCE CODE

MCS-51 MACRO ASSEMBLER 2WIRE

05/31/95

DOS 6.0 (038-N) MCS-51 MACRO ASSEMBLER, V2.2
 OBJECT MODULE PLACED IN 2WIRE.OBJ
 ASSEMBLER INVOKED BY: C:\ICE5100\ASM51\ASM51.EXE 2WIRE.TXT

LOC	OBJ	LINE	SOURCE
		1	\$ PAGELENGTH(46) PAGEWIDTH(132) DEBUG NOPAGING XREF
		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	
		11	; PIN ASSIGNMENTS:
		12	; Port 1 bit 0 is data
		13	; Port 1 bit 1 is clock
		14	;
		15	; These routines assume chip address = 0
		16	;
		17	; The oscillator frequency assumed for this app note is 12 MHz.
		18	;
		19	; These routines use software timing loops. They may have to be
		20	; adjusted if a different oscillator frequency is used.
		21	;
		22	; NOTE 1 These NOP'S added for timing delays only on 'C' parts, OR 'LC' parts
		23	; where Vcc is less than 4.5 V. and the oscillator frequency is 12 MHz.
		24	; This allows a bit rate of 100kHz.
		25	; NOTE 2 Use these NOP's with a 16 MHz oscillator and 100kHz bit rate.
		26	; For 400kHz bit rate, the NOP's in Note 1 and Note 2 are not required.
		27	;
		28	; The EEPROM will be busy after a write cycle is initiated (by a stop condition)
		29	; for between 1ms to 10 ms per page (or per byte if a byte write). This app note
		30	; assumes the user will program appropriate wait times after a write, or check
		31	; for Busy status. A subroutine is provided to check the Busy Status.
		32	;

```

33 ;RAM DEFINITIONS
34 ;
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 ADDR L EQU 0
45 DTA EQU 55H
46 BYTCNT EQU 8
47 ;
48 ;Note3 Some chip or byte address bits are embedded in the control byte. Refer to
49 ; the data sheet for exact configuration, which varies from part to part.
50 ;
51 ;*****
52 ; This section contains test loop routines. They form a simple operating shell to
53 ; allow the 2-wire interface code to be tested in a stand-alone mode. Using an
54 ; emulator, change "NONE" to one of the four listed
55 ; routines to test that function. The address and data constants can also be set as
56 ; desired.
57 ; If using a 24XX32 or 24XX65, change the called routines by adding 'L' to the end
58 ; of the name. This is required because these parts use TWO address bytes. The 'L'
59 ; routines send out the extra address byte.
60 ;*****
61 ORG 0
62 JMP START
63
64 START: MOV P1,#0FFH ;INIT PORT 1
65 CALL NONE ;TEST LOOP INSERT PROPER ADDRESS HERE
66 JMP START
67
68 NONE: RET
69 ;*
70 ;* WRITE ONE BYTE TO EEPROM
71 ;* The Address Pointer is the address in the EEPROM. The byte to be sent to the
72 ;* EEPROM is stored in the constant 'DTA'
73 ;*
74 TESTWR: MOV R3,#ADDRH ;LOAD ADDRESS POINTER FOR 24XX32 OR 24XX65 ONLY
75 MOV R4,#ADDR L ;LOAD ADDRESS POINTER FOR ALL DEVICES
76 MOV R1,#DTA ;LOAD DATA BYTE
77 CALL BYTEW ;CALL BYTE WRITE ROUTINE
78 RET
79
0030
0030
00A0
00A1
0040
0000
0000
0055
0008
0000
0000 020003
0003 7590FF
0006 12000B
0009 80F8
000B 22
000C 7B00
000E 7C00
0010 7955
0012 120037
0015 22

```

```

80 ;
81 ;* WRITE A BLOCK OF DATA TO EEPROM
82 ;* The address pointer is the address in EEPROM where data will start. The byte
83 ;* pointer is the starting address of RAM containing the block of data to be sent.
84 ;* The byte count indicates how many bytes to send to the EEPROM.
85 ;* The number of bytes that can be sent before a STOP command is issued is
86 ;* dependant on EEPROM type. Refer to the data book for specific values.
87 ;*
0016 7B00      BLKW:  MOV   R3,#ADDRH      ;LOAD ADDRESS POINTER FOR 24XX32 OR 24XX65 ONLY
0018 7C00      MOV   R4,#ADDRL      ;LOAD ADDRESS POINTER FOR ALL DEVICES
001A 7930      MOV   R1,#BYTSTR    ;LOAD BYTE POINTER
001C 7E08      MOV   R6,#BYTCNT     ;LOAD BYTE COUNT
001E 120048    CALL  PAGEW          ;CALL PAGE WRITE ROUTINE
0021 22       RET
94
95 ;
96 ;* READ ONE BYTE FROM EEPROM
97 ;* The address pointer is the address of the desired byte in EEPROM.
98 ;* The byte will be returned in R1.
99 ;*
0022 7B00      TESTRD: MOV   R3,#ADDRH      ;LOAD ADDRESS POINTER FOR 24XX32 OR 24XX65 ONLY
0024 7C00      MOV   R4,#ADDRL      ;LOAD ADDRESS POINTER FOR ALL DEVICES
0026 120082    CALL  BYTERD        ;CALL BYTE READ ROUTINE.
0029 F9       MOV   R1,A          ;SAVE THE BYTE
002A 22       RET
105
106 ;
107 ;* READ A BLOCK FROM EEPROM
108 ;* The address pointer is the starting address of the desired data block in EEPROM.
109 ;* The data pointer is the starting address in RAM where data will be stored.
110 ;* The byte count indicates how many bytes should be read.
111 ;* The entire EEPROM may be read with one READ command this way.
112 ;*
002B 7B00      BLOKRD: MOV   R3,#ADDRH      ;LOAD ADDRESS POINTER FOR 24XX32 OR 24XX65 ONLY
002D 7C00      MOV   R4,#ADDRL      ;LOAD ADDRESS POINTER FOR ALL DEVICES
002F 7930      MOV   R1,#BYTSTR    ;LOAD DATA POINTER
0031 7E08      MOV   R6,#BYTCNT     ;LOAD BYTE COUNT
0033 12005C    CALL  BLKRD        ;CALL BLOCK READ ROUTINE
0036 22       RET
118
119 ;
120 ;*END OF TEST LOOP
121 ;*****
122
123 ;*****
124 ; This routine writes a byte of data to EEPROM
125 ; The EEPROM address is assumed to be in R4. See NOTE 3.
126

```

```

127 ; The DATA to be written is assumed to be in R1
128 ; *****
129 BYTEM: MOV A,#WTCMD ;LOAD WRITE COMMAND *****
130 CALL OUTS ;SEND IT
131 MOV A,R4 ;GET LOW BYTE ADDRESS
132 CALL OUT ;SEND IT
133 MOV A,R1 ;GET DATA
134 CALL OUT ;SEND IT
135 CALL STOP ;SEND STOP CONDITION
136 RET
137
138 ; *****
139 ; THIS ROUTINE WRITES A PAGE OF DATA TO EEPROM *****
140 ; The EEPROM start address is assumed to be in R4. See NOTE 3.
141 ; The DATA pointer is in R1
142 ; The BYTE count is in R6
143 ; The number of bytes that can be transferred depends on the EEPROM used.
144 ; *****
145 PAGEW: MOV A,#WTCMD ;LOAD WRITE COMMAND *****
146 CALL OUTS ;SEND IT
147 MOV A,R4 ;GET LOW BYTE ADDRESS
148 CALL OUT ;SEND IT
149 MOV A,@R1 ;GET DATA
150 CALL OUT ;SEND IT
151 INC R1 ;INCREMENT DATA POINTER
152 DJNZ R6,BTLP ;LOOP TILL DONE
153 CALL STOP ;SEND STOP CONDITION
154 RET
155
156 ; *****
157 ; THIS ROUTINE READS A BLOCK OF DATA FROM EEPROM AT A SPECIFIED ADDRESS *****
158 ; EEPROM address in R4. See NOTE 3.
159 ; Stores data at RAM location pointed to by R1
160 ; Byte count specified in R6. May be 1 to 256 bytes
161 ; *****
162 BLKRD: MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS *****
163 CALL OUTS ;SEND IT
164 MOV A,R4 ;GET LOW BYTE ADDRESS
165 CALL OUT ;SEND IT
166 MOV A,#RDCMD ;LOAD READ COMMAND
167 CALL OUTS ;SEND IT
168 BRDLP: CALL IN ;READ DATA
169 MOV @R1,A ;STORE DATA
170 INC R1 ;INCREMENT DATA POINTER
171 DJNZ R6,AKLP ;DECREMENT LOOP COUNTER
172 CALL STOP ;IF DONE, ISSUE STOP CONDITION
173
0037 74A0
0039 120120
003C EC
003D 120158
0040 E9
0041 120158
0044 120197
0047 22

0048 74A0
004A 120120
004D EC
004E 120158
0051 E7
0052 120158
0055 09
0056 DEF9
0058 120197
005B 22

005C 74A0
005E 120120
0061 EC
0062 120158
0065 74A1
0067 120120
006A 12017E
006D F7
006E 09
006F DE04
0071 120197

```

```

0074 22      RET                ;DONE, EXIT ROUTINE
174
175
0075 C290    CLR      P1.0        ;NOT DONE, ISSUE ACK
176
0077 D291    SETB     P1.1        ;NOTE 1
177
0079 00      NOP
178
007A 00      NOP
179
007B 00      NOP
180
007C 00      NOP
181
007D 00      NOP
182
007E C291    CLR      P1.1        ;CONTINUE WITH READS
183
0080 80E8    JMP      BRDLP
184
185
*****
186
187 ; THIS ROUTINE READS A BYTE OF DATA FROM THE EEPROM
188 ; The EEPROM address is in R4. See NOTE 3.
189 ; Returns the data byte in R1
190 ; *****
191 BYTERD: MOV   A,#WTCMD      ;LOAD WRITE COMMAND TO SEND ADDRESS
192          CALL OUTS         ;SEND IT
193          MOV   A,R4        ;GET LOW BYTE ADDRESS
194          CALL OUT          ;SEND IT
195          CALL CREAD        ;GET DATA BYTE
196          RET
197
*****
198
199 ; THIS ROUTINE READS A BYTE OF DATA FROM EEPROM
200 ; From EEPROM current address pointer.
201 ; Returns the data byte in R1
202 ; *****
203 CREAD:  MOV   A,#RDCMD      ;LOAD READ COMMAND
204          CALL OUTS         ;SEND IT
205          CALL IN           ;READ DATA
206          MOV   R1,A        ;STORE DATA
207          CALL STOP        ;SEND STOP CONDITION
208          RET
209
*****
210
211 ; The next four routines are used with the 24XX32 or 24XX65 only. These parts
212 ; require two address bytes, and these routines send the second byte out.
213 ; Other than this, these routines are the same as the previous four.
214 ; *****
215 ; THIS ROUTINE READS A BLOCK OF DATA FROM EEPROM AT A SPECIFIED ADDRESS
216 ; This routine is for the 24LC32 or 24LC64
217 ; EEPROM address in R3:R4
218 ; Stores data at RAM location pointed to by R1
219 ; Byte count specified in R6. May be 1 to 256 bytes
220 ; *****

```

```

009C 74A0      BLKRD1:  MOV     A,#WTCMD      ;LOAD WRITE COMMAND TO SEND ADDRESS
009E 120120    CALL    OUTS                 ;SEND IT
00A1 EB       MOV     A,R3                 ;GET HI BYTE ADDRESS
00A2 120158    CALL    OUT                  ;SEND IT
00A5 EC       MOV     A,R4                 ;GET LOW BYTE ADDRESS
00A6 120158    CALL    OUT                  ;SEND IT
00A9 74A1     MOV     A,#RDCMD            ;LOAD READ COMMAND
00AB 120120    CALL    OUTS                 ;SEND IT
00AE 80EA     JMP     BRDLP                ;CONTINUE WITH DATA READ
230
231 ;*****
232 ; This routine writes a byte of data to EEPROM
233 ; This routine is for the 24LC32 or 24LC64
234 ; The EEPROM address is assumed to be in R3:R4
235 ; The DATA to be written is assumed to be in R1
236 ;*****
00B0 74A0      BYTEWL:  MOV     A,#WTCMD      ;LOAD WRITE COMMAND
00B2 120120    CALL    OUTS                 ;SEND IT
00B5 EB       MOV     A,R3                 ;GET HI BYTE ADDRESS
00B6 120158    CALL    OUT                  ;SEND IT
00B9 EC       MOV     A,R4                 ;GET LOW BYTE ADDRESS
00BA 120158    CALL    OUT                  ;SEND IT
00BD E9       MOV     A,R1                 ;GET DATA
00BE 120158    CALL    OUT                  ;SEND IT
00C1 120197    CALL    STOP                 ;SEND STOP CONDITION
00C4 22       RET
246
247
248 ;*****
249 ; THIS ROUTINE WRITES A PAGE OF DATA TO EEPROM
250 ; This routine is for the 24LC32 or 24LC64
251 ; The EEPROM start address is assumed to be in R3:R4
252 ; The DATA pointer is in R1
253 ; The BYTE count is in R6
254 ; The number of bytes that can be transferred depends on the EEPROM in use.
255 ;*****
00C5 74A0      PAGEWL:  MOV     A,#WTCMD      ;LOAD WRITE COMMAND
00C7 120120    CALL    OUTS                 ;SEND IT
00CA EB       MOV     A,R3                 ;GET HI BYTE ADDRESS
00CB 120158    CALL    OUT                  ;SEND IT
00CE EC       MOV     A,R4                 ;GET LOW BYTE ADDRESS
00CF 120158    CALL    OUT                  ;SEND IT
00D2 E7       MOV     A,@R1                ;GET DATA
00D3 120158    CALL    OUT                  ;SEND IT
00D6 09       INC     R1                   ;INCREMENT DATA POINTER
00D7 DEF9     DJNZ   R6,BTLP              ;LOOP TILL DONE
00D9 120197    CALL    STOP                 ;SEND STOP CONDITION
00DC 22       RET
267

```

```

268 ; *****
269 ; THIS ROUTINE READS A BYTE OF DATA FROM THE EEPROM *****
270 ; This routine is for the 24LC32 or 24LC64
271 ; The EEPROM address is in R3:R4
272 ; Returns the data byte in R1
273 ; *****
274 ; *****
275 BYTERDL: MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS
276 CALL OUTS ;SEND IT
277 MOV A,R3 ;GET HI BYTE ADDRESS
278 CALL OUT ;SEND IT
279 MOV A,R4 ;GET LOW BYTE ADDRESS
280 CALL OUT ;SEND IT
281 CALL CREAD ;GET DATA BYTE
282 RET
283
284 ;
285 ;SUBROUTINES
286 ;
287 ; *****
288 ; This routine tests for WRITE DONE condition *****
289 ; by testing for an ACK.
290 ; This routine can be run as soon as a STOP condition
291 ; has been generated after the last data byte has been
292 ; sent to the EEPROM. No ACK will be returned until
293 ; the EEPROM is done with the write operation.
294 ; *****
295 ACKTST: MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS
296 MOV R2,#8 ;LOOP COUNT -- EQUAL TO BIT COUNT
297 CLR P1.0 ;START CONDITION -- DATA = 0
298 NOP ;NOTE 1
299 NOP
300 NOP
301 NOP ;NOTE 2
302 NOP
303 CLR P1.1 ;CLOCK = 0
304 AKTLP: RLC A ;SHIFT BIT
305 JNC AKTLS
306 SETB P1.0 ;DATA = 1
307 JMP AKTLL
308 AKTLS: CLR P1.0 ;CONTINUE
309 AKTLL: SETB P1.1 ;DATA = 0
310 NOP ;CLOCK HI
311 NOP ;NOTE 1
312 NOP
313 NOP
314 NOP ;NOTE 2
315 NOP
316 NOP
317 NOP
318 NOP
319 NOP
320 NOP
321 NOP
322 NOP
323 NOP
324 NOP
325 NOP
326 NOP
327 NOP
328 NOP
329 NOP
330 NOP
331 NOP
332 NOP
333 NOP
334 NOP
335 NOP
336 NOP
337 NOP
338 NOP
339 NOP
340 NOP
341 NOP
342 NOP
343 NOP
344 NOP
345 NOP
346 NOP
347 NOP
348 NOP
349 NOP
350 NOP
351 NOP
352 NOP
353 NOP
354 NOP
355 NOP
356 NOP
357 NOP
358 NOP
359 NOP
360 NOP
361 NOP
362 NOP
363 NOP
364 NOP
365 NOP
366 NOP
367 NOP
368 NOP
369 NOP
370 NOP
371 NOP
372 NOP
373 NOP
374 NOP
375 NOP
376 NOP
377 NOP
378 NOP
379 NOP
380 NOP
381 NOP
382 NOP
383 NOP
384 NOP
385 NOP
386 NOP
387 NOP
388 NOP
389 NOP
390 NOP
391 NOP
392 NOP
393 NOP
394 NOP
395 NOP
396 NOP
397 NOP
398 NOP
399 NOP
400 NOP
401 NOP
402 NOP
403 NOP
404 NOP
405 NOP
406 NOP
407 NOP
408 NOP
409 NOP
410 NOP
411 NOP
412 NOP
413 NOP
414 NOP
415 NOP
416 NOP
417 NOP
418 NOP
419 NOP
420 NOP
421 NOP
422 NOP
423 NOP
424 NOP
425 NOP
426 NOP
427 NOP
428 NOP
429 NOP
430 NOP
431 NOP
432 NOP
433 NOP
434 NOP
435 NOP
436 NOP
437 NOP
438 NOP
439 NOP
440 NOP
441 NOP
442 NOP
443 NOP
444 NOP
445 NOP
446 NOP
447 NOP
448 NOP
449 NOP
450 NOP
451 NOP
452 NOP
453 NOP
454 NOP
455 NOP
456 NOP
457 NOP
458 NOP
459 NOP
460 NOP
461 NOP
462 NOP
463 NOP
464 NOP
465 NOP
466 NOP
467 NOP
468 NOP
469 NOP
470 NOP
471 NOP
472 NOP
473 NOP
474 NOP
475 NOP
476 NOP
477 NOP
478 NOP
479 NOP
480 NOP
481 NOP
482 NOP
483 NOP
484 NOP
485 NOP
486 NOP
487 NOP
488 NOP
489 NOP
490 NOP
491 NOP
492 NOP
493 NOP
494 NOP
495 NOP
496 NOP
497 NOP
498 NOP
499 NOP
500 NOP
501 NOP
502 NOP
503 NOP
504 NOP
505 NOP
506 NOP
507 NOP
508 NOP
509 NOP
510 NOP
511 NOP
512 NOP
513 NOP
514 NOP
515 NOP
516 NOP
517 NOP
518 NOP
519 NOP
520 NOP
521 NOP
522 NOP
523 NOP
524 NOP
525 NOP
526 NOP
527 NOP
528 NOP
529 NOP
530 NOP
531 NOP
532 NOP
533 NOP
534 NOP
535 NOP
536 NOP
537 NOP
538 NOP
539 NOP
540 NOP
541 NOP
542 NOP
543 NOP
544 NOP
545 NOP
546 NOP
547 NOP
548 NOP
549 NOP
550 NOP
551 NOP
552 NOP
553 NOP
554 NOP
555 NOP
556 NOP
557 NOP
558 NOP
559 NOP
560 NOP
561 NOP
562 NOP
563 NOP
564 NOP
565 NOP
566 NOP
567 NOP
568 NOP
569 NOP
570 NOP
571 NOP
572 NOP
573 NOP
574 NOP
575 NOP
576 NOP
577 NOP
578 NOP
579 NOP
580 NOP
581 NOP
582 NOP
583 NOP
584 NOP
585 NOP
586 NOP
587 NOP
588 NOP
589 NOP
590 NOP
591 NOP
592 NOP
593 NOP
594 NOP
595 NOP
596 NOP
597 NOP
598 NOP
599 NOP
600 NOP
601 NOP
602 NOP
603 NOP
604 NOP
605 NOP
606 NOP
607 NOP
608 NOP
609 NOP
610 NOP
611 NOP
612 NOP
613 NOP
614 NOP
615 NOP
616 NOP
617 NOP
618 NOP
619 NOP
620 NOP
621 NOP
622 NOP
623 NOP
624 NOP
625 NOP
626 NOP
627 NOP
628 NOP
629 NOP
630 NOP
631 NOP
632 NOP
633 NOP
634 NOP
635 NOP
636 NOP
637 NOP
638 NOP
639 NOP
640 NOP
641 NOP
642 NOP
643 NOP
644 NOP
645 NOP
646 NOP
647 NOP
648 NOP
649 NOP
650 NOP
651 NOP
652 NOP
653 NOP
654 NOP
655 NOP
656 NOP
657 NOP
658 NOP
659 NOP
660 NOP
661 NOP
662 NOP
663 NOP
664 NOP
665 NOP
666 NOP
667 NOP
668 NOP
669 NOP
670 NOP
671 NOP
672 NOP
673 NOP
674 NOP
675 NOP
676 NOP
677 NOP
678 NOP
679 NOP
680 NOP
681 NOP
682 NOP
683 NOP
684 NOP
685 NOP
686 NOP
687 NOP
688 NOP
689 NOP
690 NOP
691 NOP
692 NOP
693 NOP
694 NOP
695 NOP
696 NOP
697 NOP
698 NOP
699 NOP
700 NOP
701 NOP
702 NOP
703 NOP
704 NOP
705 NOP
706 NOP
707 NOP
708 NOP
709 NOP
710 NOP
711 NOP
712 NOP
713 NOP
714 NOP
715 NOP
716 NOP
717 NOP
718 NOP
719 NOP
720 NOP
721 NOP
722 NOP
723 NOP
724 NOP
725 NOP
726 NOP
727 NOP
728 NOP
729 NOP
730 NOP
731 NOP
732 NOP
733 NOP
734 NOP
735 NOP
736 NOP
737 NOP
738 NOP
739 NOP
740 NOP
741 NOP
742 NOP
743 NOP
744 NOP
745 NOP
746 NOP
747 NOP
748 NOP
749 NOP
750 NOP
751 NOP
752 NOP
753 NOP
754 NOP
755 NOP
756 NOP
757 NOP
758 NOP
759 NOP
760 NOP
761 NOP
762 NOP
763 NOP
764 NOP
765 NOP
766 NOP
767 NOP
768 NOP
769 NOP
770 NOP
771 NOP
772 NOP
773 NOP
774 NOP
775 NOP
776 NOP
777 NOP
778 NOP
779 NOP
780 NOP
781 NOP
782 NOP
783 NOP
784 NOP
785 NOP
786 NOP
787 NOP
788 NOP
789 NOP
790 NOP
791 NOP
792 NOP
793 NOP
794 NOP
795 NOP
796 NOP
797 NOP
798 NOP
799 NOP
800 NOP
801 NOP
802 NOP
803 NOP
804 NOP
805 NOP
806 NOP
807 NOP
808 NOP
809 NOP
810 NOP
811 NOP
812 NOP
813 NOP
814 NOP
815 NOP
816 NOP
817 NOP
818 NOP
819 NOP
820 NOP
821 NOP
822 NOP
823 NOP
824 NOP
825 NOP
826 NOP
827 NOP
828 NOP
829 NOP
830 NOP
831 NOP
832 NOP
833 NOP
834 NOP
835 NOP
836 NOP
837 NOP
838 NOP
839 NOP
840 NOP
841 NOP
842 NOP
843 NOP
844 NOP
845 NOP
846 NOP
847 NOP
848 NOP
849 NOP
850 NOP
851 NOP
852 NOP
853 NOP
854 NOP
855 NOP
856 NOP
857 NOP
858 NOP
859 NOP
860 NOP
861 NOP
862 NOP
863 NOP
864 NOP
865 NOP
866 NOP
867 NOP
868 NOP
869 NOP
870 NOP
871 NOP
872 NOP
873 NOP
874 NOP
875 NOP
876 NOP
877 NOP
878 NOP
879 NOP
880 NOP
881 NOP
882 NOP
883 NOP
884 NOP
885 NOP
886 NOP
887 NOP
888 NOP
889 NOP
890 NOP
891 NOP
892 NOP
893 NOP
894 NOP
895 NOP
896 NOP
897 NOP
898 NOP
899 NOP
900 NOP
901 NOP
902 NOP
903 NOP
904 NOP
905 NOP
906 NOP
907 NOP
908 NOP
909 NOP
910 NOP
911 NOP
912 NOP
913 NOP
914 NOP
915 NOP
916 NOP
917 NOP
918 NOP
919 NOP
920 NOP
921 NOP
922 NOP
923 NOP
924 NOP
925 NOP
926 NOP
927 NOP
928 NOP
929 NOP
930 NOP
931 NOP
932 NOP
933 NOP
934 NOP
935 NOP
936 NOP
937 NOP
938 NOP
939 NOP
940 NOP
941 NOP
942 NOP
943 NOP
944 NOP
945 NOP
946 NOP
947 NOP
948 NOP
949 NOP
950 NOP
951 NOP
952 NOP
953 NOP
954 NOP
955 NOP
956 NOP
957 NOP
958 NOP
959 NOP
960 NOP
961 NOP
962 NOP
963 NOP
964 NOP
965 NOP
966 NOP
967 NOP
968 NOP
969 NOP
970 NOP
971 NOP
972 NOP
973 NOP
974 NOP
975 NOP
976 NOP
977 NOP
978 NOP
979 NOP
980 NOP
981 NOP
982 NOP
983 NOP
984 NOP
985 NOP
986 NOP
987 NOP
988 NOP
989 NOP
990 NOP
991 NOP
992 NOP
993 NOP
994 NOP
995 NOP
996 NOP
997 NOP
998 NOP
999 NOP
1000 NOP

```



```

010B C291      315      CLR      P1.1      ;CLOCK LOW
010D DAEB      316      DJNZ     R2,AKTLP    ;DECREMENT COUNTER
010F D290      317      SETB     P1.0      ;TURN PIN INTO INPUT
0111 00        318      NOP          ;NOTE 1
0112 00        319      NOP          ;NOTE 2
0113 D291      320      SETB     P1.1      ;CLOCK ACK
0115 00        321      NOP          ;NOTE 1
0116 00        322      NOP          ;NOTE 2
0117 00        323      NOP          ;NOTE 2
0118 00        324      NOP          ;NOTE 2
0119 00        325      NOP          ;NOTE 2
011A 309002    326      JNB      P1.0,EXIT    ;EXIT IF NO ACK (WRITE NOT DONE)
011D C291      327      CLR      P1.1      ;SET DONE FLAG
011F 22        328      ;
329      EXIT:  RET
330
331
332
333
334      ;*****
335      ; THIS ROUTINE SENDS OUT CONTENTS OF THE ACCUMULATOR
336      ; to the EEPROM and includes START condition. Refer to the data sheets
337      ;for discussion of START and STOP conditions.
338      ;*****
0120 7A08      339      MOV      R2,#8      ;LOOP COUNT -- EQUAL TO BIT COUNT
0122 D290      340      SETB     P1.0      ;INSURE DATA IS HI
0124 D291      341      SETB     P1.1      ;INSURE CLOCK IS HI
0126 00        342      NOP          ;NOTE 1
0127 00        343      NOP          ;NOTE 2
0128 00        344      NOP          ;NOTE 2
0129 00        345      NOP          ;NOTE 2
012A 00        346      NOP          ;NOTE 2
012B C290      347      CLR      P1.0      ;START CONDITION -- DATA = 0
012D 00        348      NOP          ;NOTE 1
012E 00        349      NOP          ;NOTE 2
012F 00        350      NOP          ;NOTE 2
0130 00        351      NOP          ;NOTE 2
0131 00        352      NOP          ;NOTE 2
0132 C291      353      CLR      P1.1      ;CLOCK = 0
0134 33        354      RLC      A          ;SHIFT BIT
0135 5005      355      JNC      BITLS      ;DATA = 1
0137 D290      356      SETB     P1.0      ;CONTINUE
0139 02013E    357      JMP      OTSL1      ;DATA = 0
013C C290      358      BITLS:  CLR      P1.0      ;CLOCK HI
013E D291      359      OTSL1:  SETB     P1.1      ;NOTE 1
0140 00        360      NOP          ;NOTE 1
0141 00        361      NOP          ;NOTE 1

```

```

0142 00 NOP
0143 00 NOP ;NOTE 2
0144 00 NOP
0145 C291 CLR P1.1 ;CLOCK LOW
0147 DAEB DJNZ R2,OTSLP ;DECREMENT COUNTER
0149 D290 SETB P1.0 ;TURN PIN INTO INPUT
014B 00 NOP ;NOTE 1
014C 00 NOP ;NOTE 2
014D 00 NOP
014E D291 SETB P1.1 ;CLOCK ACK
0150 00 NOP ;NOTE 1
0151 00 NOP
0152 00 NOP
0153 00 NOP ;NOTE 2
0154 00 NOP
0155 C291 CLR P1.1
0157 22 RET

362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408

0158 7A08 OUT: MOV R2,#8 ;LOOP COUNT -- EQUAL TO BIT COUNT
015A 33 OTLP: RLC A ;SHIFT BIT
015B 5005 JNC BITL
015D D290 SETB P1.0 ;DATA = 1
015F 020164 JMP OTL1 ;CONTINUE
0162 C290 CLR P1.0 ;DATA = 0
0164 D291 OTL1: SETB P1.1 ;CLOCK HI
0166 00 NOP ;NOTE 1
0167 00 NOP
0168 00 NOP
0169 00 NOP ;NOTE 2
016A 00 NOP
016B C291 CLR P1.1 ;CLOCK LOW
016D DAEB DJNZ R2,OTLP ;DECREMENT COUNTER
016F D290 SETB P1.0 ;TURN PIN INTO INPUT
0171 00 NOP ;NOTE 1
0172 00 NOP ;NOTE 2
0173 00 NOP
0174 D291 SETB P1.1 ;CLOCK ACK
0176 00 NOP ;NOTE 1
0177 00 NOP
0178 00 NOP
0179 00 NOP ;NOTE 2
017A 00 NOP

;*****
; THIS ROUTINE SENDS OUT CONTENTS OF ACCUMULATOR TO EEPROM
; without sending a START condition.
;*****

```

```

017B C291          CLR    P1.1
017D 22          RET

409
410
411
412 *****
413 ; THIS ROUTINE READS IN A BYTE FROM THE EEPROM
414 ; and stores it in the accumulator
415 *****
416 *****
417 IN:          MOV    R2,#8          ;LOOP COUNT
418          SETB   P1.0          ;SET DATA BIT HIGH FOR INPUT
419          INLP:  CLR    P1.1          ;CLOCK LOW
420          NOP
421          NOP
422          NOP
423          NOP
424          NOP
425          NOP
426          SETB   P1.1          ;CLOCK HIGH
427          CLR    C              ;CLEAR CARRY
428          JNB   P1.0,INL1       ;JUMP IF DATA =0
429          CPL    C              ;SET CARRY IF DATA =1
430          RLC    A              ;ROTATE DATA INTO ACCUMULATOR
431          DJNZ  R2,INLP         ;DECREMENT COUNTER
432          CLR    P1.1          ;CLOCK LOW
433          RET
434
435
436          STOP:  CLR    P1.0          ;STOP CONDITION SET DATA LOW
437          NOP
438          NOP
439          NOP
440          NOP
441          NOP
442          SETB   P1.1          ;SET CLOCK HI
443          NOP
444          NOP
445          NOP
446          NOP
447          NOP
448          SETB   P1.0          ;SET DATA HI
449          RET
450
451 *****
452 ;These routines contain special commands only for the 24LC65 SMART SERIAL EEPROM
453 ; *
454 ; SET SECURE BLOCK
455 ; ASSUMES START BLOCK 0 & BLOCK LENGTH OF 1

```

```

456 ; The numbers are implicit in the commands. Refer to the data sheet for details.
457 ; *
458 SETSEC: MOV R3,#80H ;LOAD COMMAND AND STARTING BLOCK NUMBER
459 MOV R4,#0
460 MOV R1,#81H ;SET COMMAND FOR NUMBER OF BLOCKS TO SECURE
461 CALL BYTEW ;EXECUTE
462 RET
463
464 ; *
465 ; READ SECURE BLOCK NUMBER(S)
466 ; RETURNS BLOCK NUMBER IN R1 AND BLOCK COUNT IN R2
467 ; (UPPER NIBBLES WILL BE 1'S)
468 ; *
469
470 RDSEC: MOV A,#WTCMD ;LOAD WRITE COMMAND TO SEND ADDRESS
471 CALL OUTS ;SEND IT
472 MOV A,#80H ;LOAD COMMAND
473 CALL OUT ;SEND IT
474 MOV A,#0 ;LOAD COMMAND
475 CALL OUT ;SEND IT
476 MOV A,#0C0H ;LOAD COMMAND
477 CALL OUT ;SEND IT
478 CALL IN ;READ STARTING BLOCK NUMBER
479 MOV R1,A ;STORE IT
480 NOP ;NOTE 1
481 NOP
482 NOP
483 NOP
484 NOP
485 CLR P1.0 ;ISSUE ACK
486 SETB P1.1
487 NOP ;NOTE 1
488 NOP
489 NOP
490 NOP
491 NOP ;NOTE 2
492 CLR P1.1
493 CALL IN ;READ NUMBER OF BLOCKS
494 MOV R2,A ;STORE IT
495 CALL STOP ;SEND STOP CONDITION
496 RET
497
498 ; *
499 ; SET HIGH-ENDURANCE BLOCK NUMBER
500 ; ASSUMES BLOCK 0
501 ; *
502

```

```

01DA 7B80          SETHI:  MOV   R3,#80H      ;LOAD COMMAND AND BLOCK NUMBER
01DC 7C00          MOV   R4,#0
01DE 7900          MOV   R1,#0      ;SET DATA = 0
01E0 1137          CALL  BYTEW      ;EXECUTE
01E2 22           RET
508
509
510 ; READ HIGH-ENDURANCE BLOCK NUMBER
511 ; RETURNS BLOCK NUMBER IN R1 (UPPER NIBBLE WILL BE 1'S)
512 ;
513 READHI: MOV   R3,#80H      ;LOAD COMMAND
514          MOV   R4,#0      ;EXECUTE
515          CALL  HIEND
516          RET
517
518 HIEND:  MOV   A,#WTCMD     ;LOAD WRITE COMMAND TO SEND ADDRESS
519          CALL  OUTS        ;SEND IT
520          MOV   A,R3        ;GET HI BYTE ADDRESS
521          CALL  OUT        ;SEND IT
522          MOV   A,R4        ;GET LOW BYTE ADDRESS
523          CALL  OUT        ;SEND IT
524          MOV   A,#RDEND    ;LOAD READ COMMAND
525          CALL  OUT        ;SEND IT
526          CALL  IN         ;READ DATA
527          MOV   R1,A        ;STORE DATA
528          CALL  STOP       ;SEND STOP CONDITION
529          RET
530
531 ;END of 24LC65 Routines
532 ;*****
533          END

```

ASSEMBLY COMPLETE, NO ERRORS FOUND

AN614

NOTES:

NOTES:

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.mchip.com/microchip>

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: 708 285-0071 Fax: 708 285-0075

Dallas

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

Dayton

Microchip Technology Inc.
35 Rockridge Road
Englewood, OH 45322
Tel: 513 832-2543 Fax: 513 832-2841

Los Angeles

Microchip Technology Inc.
18201 Von Karman, Suite 455
Irvine, CA 92715
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

ASIA/PACIFIC

Hong Kong

Microchip Technology
Unit No. 3002-3004, Tower 1
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T. Hong Kong
Tel: 852 2 401 1200 Fax: 852 2 401 3431

Korea

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

Singapore

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

Taiwan

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

EUROPE

United Kingdom

Arizona Microchip Technology Ltd.
Unit 6, The Courtyard
Meadow Bank, Furlong Road
Bourne End, Buckinghamshire SL8 5AJ
Tel: 44 0 1628 851077 Fax: 44 0 1628 850259

France

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

Germany

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

Italy

Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Pegaso Ingresso No. 2
Via Paracelso 23, 20041
Agrate Brianza (MI) Italy
Tel: 39 039 689 9939 Fax: 39 039 689 9883

JAPAN

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

9/22/95



MICROCHIP

All rights reserved. © 1995, Microchip Technology Incorporated, USA.

Information contained in this publication regarding device applications and the like is intended through 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. All rights reserved. All other trademarks mentioned herein are the property of their respective companies.