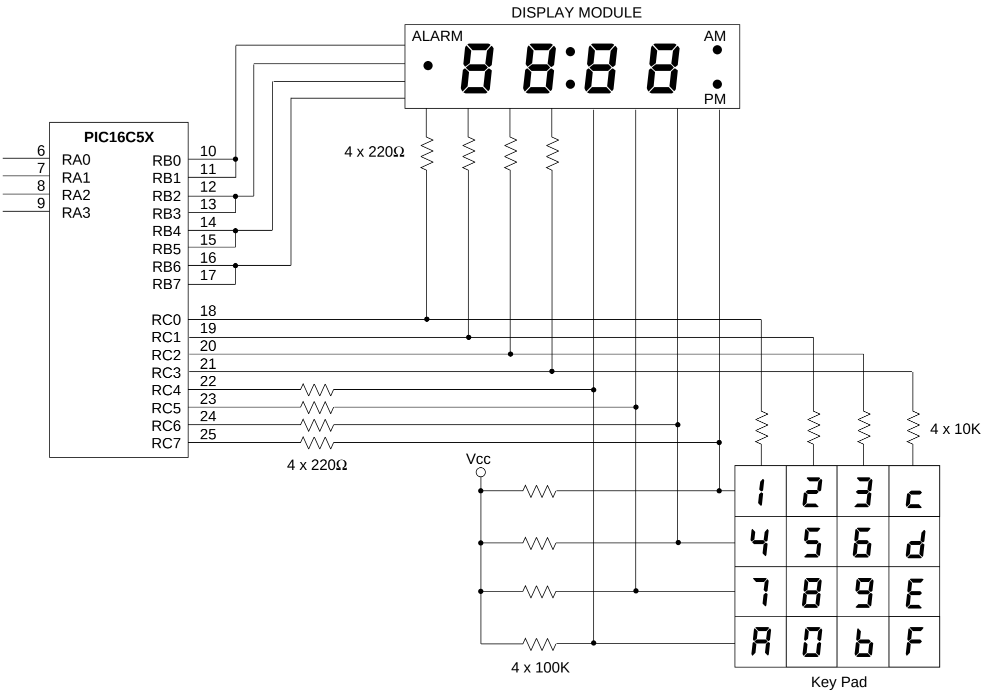
Multiplexing LED Drive and a 4x4 Keypad Sampling

INTRODUCTION

Many applications require driving LEDs along with an interface to a keypad. Implementing such designs usually involves using up significant amounts of the processors I/O lines. This application note describes a method which uses only 16 I/O pins of a PIC16C5X microcontroller to sample a 4x4 keypad matrix, and directly drive four 7 segment LEDs (see Figure 1). Direct drive of the LEDs is possible, because of the high sink and source capabilities of the PIC16C5X microcontroller, thus eliminating the use of external drive transistor, and resulting in reduced cost and complexity of the overall circuit.

Typically applications having LEDs and keypads also keep track of real time, in order to synchronize certain key events. An Industrial Clock/Timer example has been used in this application note as a demonstration of this technique. The software overhead to keep track of real time is minimal and the user can modify the code to significantly expand the functionality of this circuit.

FIGURE 1 - PIC16C5X INTERFACE TO A SEGMENT DISPLAY AND 4X4 KEYPAD



PART A: 4X4 KEY MATRIX SAMPLING

Implementation

The 4x4 Key Matrix is connected to port C of the PIC16C5X (Figure 2a). The four columns are connected to RC0-RC3 and the four rows are connected to RC4-RC7. Each digit is refreshed every 20 ms. with a 5 ms pulse. The keypad is sampled every 20 ms with four 3 µs pulses (Figure 3).

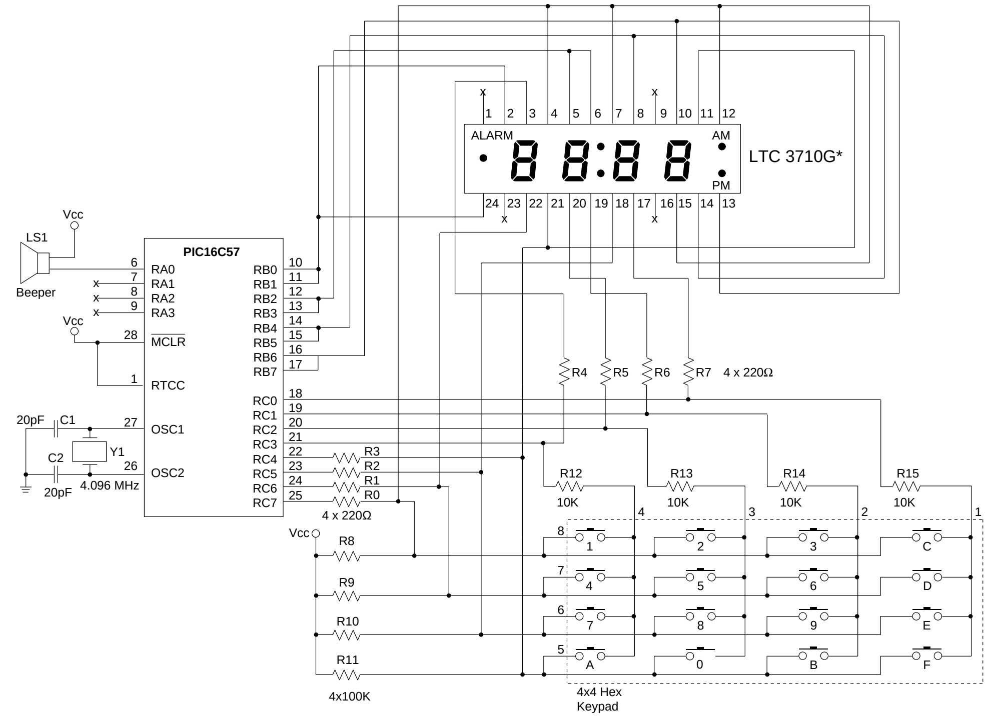
The keypad sampling is as follows:

1. The columns are connected to output pins, and the rows are connected to input pins.
2. Each column is sequentially driven to a low voltage while at the same instance the four rows are sampled. Since the rows are all held high with pull-up resistors, all four inputs will normally be high. If a key is pressed in a column which is at a low level, that low level will be conducted to the input pin through the closed key and the corresponding row will be sensed as a low.
3. Before a new column is brought low, care should be taken to discharge the input pins (see code section for details).
4. A 50 ms key debounce technique has been implemented in the software, in order to eliminate multiple key strokes.

Notes:

1. Resistors R8-R11 and R12-R14 have been selected such that their ratio is 1:10. This will insure a 0.5 Volt level at the input, when a key is pressed. Also R8-R14 should have a value such that their current contribution to the LEDs segments is negligible.
2. In circuits where there is substantial interference between the key matrix and the LED drive circuit, the alternative circuit (Figure 2b) should be utilized. Diodes in the path of all pins connected to the keypad insure that there is minimal interference from the keypad, when it is not being sampled.

FIGURE 2A - PIC16C5X INDUSTRIAL CLOCK/TIMER SCHEMATIC



* Each LED segment should draw no more than 7mA.

PART B: INDUSTRIAL CLOCK/TIMER

Clock Selection

The 4.096 MHz crystal oscillator is the time base. The PIC16C5X internally divides the clock by 4 to give an internal clock of 1.024 MHz. This clock is further divided by 32 (by the prescaler in the OPTIONS register) to give a clock of 32 KHz which is used to increment the RTCC in the PIC16C5X. If the RTCC is initialized to 96, it would overflow to 0 in 5 ms.

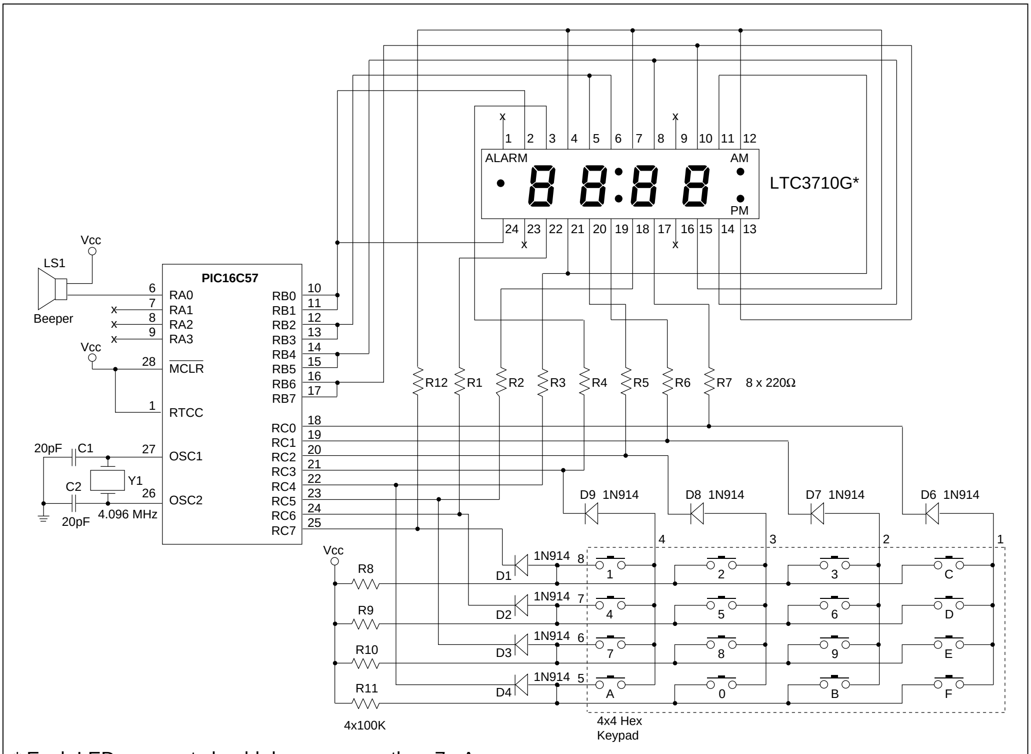
(256-96) x (1/32000) = 5.000 ms

This 5 ms is used to count the seconds, minutes and hours in the clock/timer. It is also used as a time base to update the display digits and sample the keyboard. The clock speed being 4.096 MHz, each instruction will execute in 1 µs. Therefore in 5 ms, approximately 5000 instructions can be executed. This gives sufficient time to execute a large section of code and not miss the overflow in the RTCC.

Using a 3.579545 MHz color burst crystal oscillator as a time base

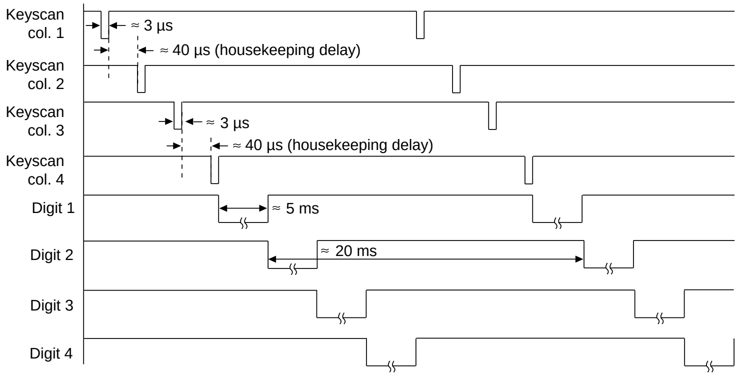
Some users may want to use a color burst crystal oscillator as a time base, because of its low cost. If a 3.579545 MHz crystal is used, then the internal clock will be 1.117 µs. If this is prescaled by 32, the RTCC will be incremented every 35.758 µs. Initializing the RTCC with 116 will cause it to overflow to 0 in 5.006 ms. giving an error of 0.12%. This error can be corrected in software by making time adjustments every minute and/or every hour.

FIGURE 2B - PIC16C5X ALARM CLOCK SCHEMATIC (USING DIODES)



* Each LED segment should draw no more than 7mA.

FIGURE 3 - KEY SCAN AND LED DIGIT SELECT TIMING



FEATURES

The Flow Chart (Figure 4) shows the sequence of events in the clock/timer software. The clock has the following features:

1. 12 hour clock with a.m./p.m.
2. 12 hour alarm with a.m./p.m.
3. Full function Hex keypad (Figure 5).
4. AA audible alarm for 1 minute.
5. 10 minute alarm disable.

SETTING CLOCK/TIMER FUNCTIONS

Function	Key Sequence to Activate Function
Set Real Time	Set → Hours (tens) → Hours → Minutes (tens) → Minutes → AM/PM → Set
View Alarm Time	Alarm (alarm time is displayed for 5 seconds)
Set Alarm Time	Alarm → Set (must be pressed when alarm LED is flashing) → Hours (tens) → Hours → Minutes (tens) → Minutes → AM/PM → Set
Enable/Disable Alarm	Alarm → Alarm (toggles alarm status)
Disable AA alarm	Disable Alarm (disable audible beep for 10 minutes)
Clear Alarm	Clear Alarm (clears audible alarm)
Abort Entry	Clear Entry (aborts data entry mode when setting real and alarm time)

Notes: 1. Valid key strokes will be acknowledged with a beep.

2. Hours and minutes used above correspond to digits 0 - 9 on the keypad.

FIGURE 4 - TIMER/CLOCK FLOW CHART

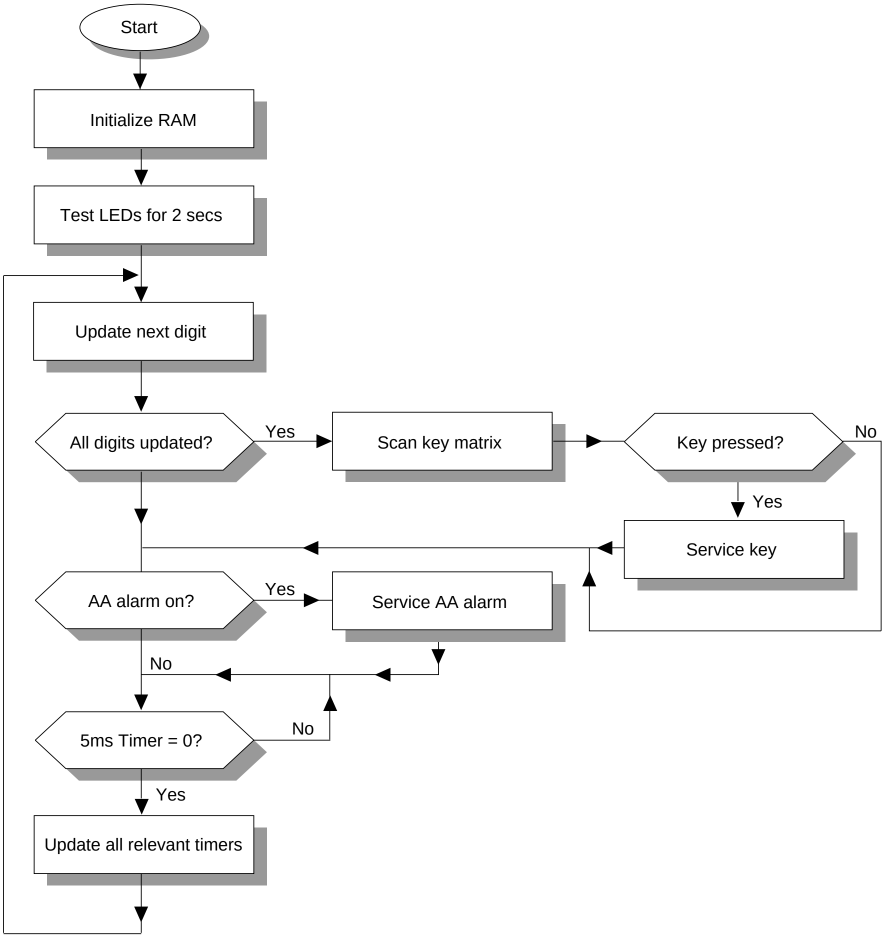
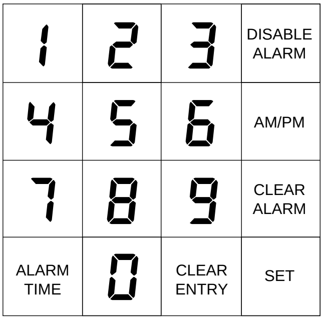
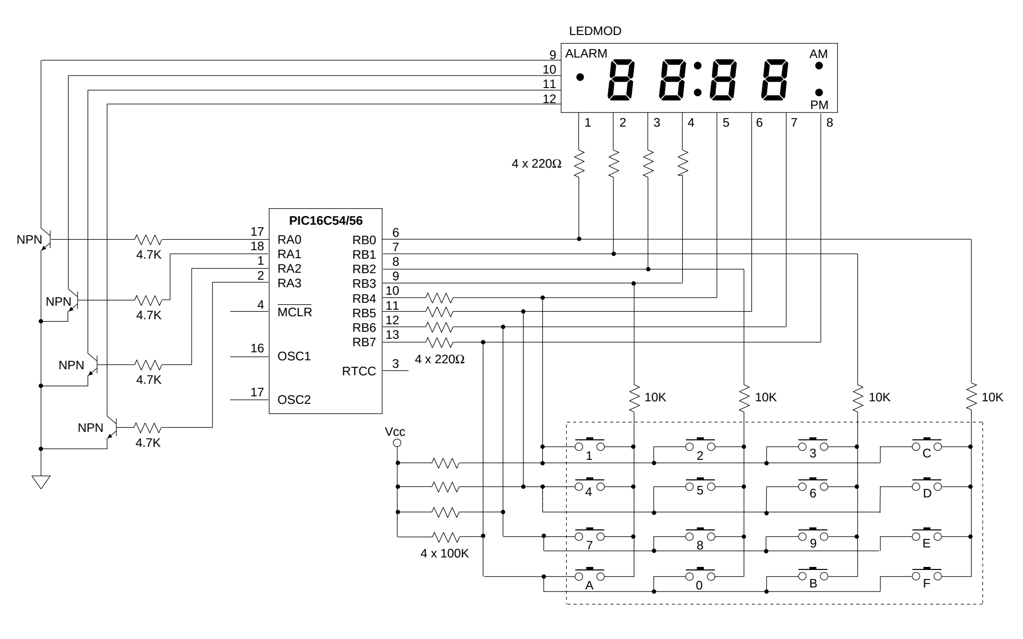


FIGURE 5 - KEYPAD



1	2	3	DISABLE ALARM
4	5	6	AM/PM
7	8	9	CLEAR ALARM
ALARM TIME	0	CLEAR ENTRY	SET

FIGURE 6 - INTERFACE TO PIC16C54/56



SUMMARY

This Application Note demonstrates a simple method of interfacing the PIC16C5X to 7-segment LEDs and a keypad. The key features of the PIC16C5X which made this possible are:

1. High sink/source of the I/O ports.
2. Fast instruction cycle for quick key-scan.
3. RISC processor allowing minimal overhead for real time clock maintenance.
4. Reconfigurable I/O ports, enabling dual functionality of ports.

Figure 6 depicts a block diagram connecting a PIC16C54/56 to a 4-digit, 7-segment LED display and a 4x4 hex keypad. Since only 12 I/O pins are available in the PIC16C54/56, external npn transistor will have to be utilized to sink the current from each digit.

CODE SIZE

Key scan → 97 bytes

Display update → 113 bytes

Author: Stan D'Souza
Logic Products Division

APPENDIX A: CODE LISTING

```
MPASM 1.00 Released       CLK.ASM   7-15-1994  13:15:10                 PAGE  1
Alarm Clock

LOC  OBJECT CODE     LINE SOURCE TEXT

                     0001                TITLE        “Alarm Clock”
                     0002         LIST  P = 16C57,f=inhx8m
                     0003 ;
                     0004 ;Define Equates:
                     0005 ;
07FF                 0006         PIC57   EQU     7FFH
                     0007
;************************************************************************
                     0008 ;External Ossc. used = 4.096Mhz. Prescaler of 32 used, which gives a
                     0009 ;31.25 microSec increment of the RTCC. If RTCC is intially loaded with
96,
                     0010 ;it would overflow to 0 in 5.000 milliSecs. Giving a 0.00% error.
0060                 0011         MSEC5   EQU     D’96'
                     0012
;*************************************************************************
0000                 0013         C       EQU     0
0000                 0014         BEP     EQU     0
0000                 0015         RTATS   EQU     0
0001                 0016         DC      EQU     1
0001                 0017         HR10    EQU     1
0002                 0018         Z       EQU     2
0002                 0019         HR      EQU     2
0003                 0020         MIN10   EQU     3
0004                 0021         MIN     EQU     4
0004                 0022         FLASH   EQU     4
0005                 0023         PA0     EQU     5
0005                 0024         KEY_BEEP EQU    5
0005                 0025         AMPM    EQU     5
0006                 0026         PA1     EQU     6
0000                 0027         F0      EQU     0
0006                 0028         KEY_HIT EQU     6
0006                 0029         ALED    EQU     6
0007                 0030         AM_PM   EQU     7
0003                 0031         COLON   EQU     3
0002                 0032         ALRMLED EQU     2
0007                 0033         SERVICED EQU    7
0000                 0034         ALONOF  EQU     0
0001                 0035         INAL    EQU     1
0002                 0036         SILNC   EQU     2
0003                 0037         INAA    EQU     3
0005                 0038         INKEYBEP EQU    5
                     0039 ;
                     0040 ;DEFINE RAM LOCATIONS:
0001                 0041         RTCC    EQU     1
0002                 0042         PC      EQU     2
0003                 0043         STATUS  EQU     3
0004                 0044         FSR     EQU     4
0005                 0045         PORT_A  EQU     5
0006                 0046         PORT_B  EQU     6
0007                 0047         PORT_C  EQU     7
                     0048 ;DEFINE REAL TIME MODE REGS (RTM)
0008                 0049         MSTMR   EQU     8       ;MILLI SEC. TIMER
0009                 0050         STMR    EQU     9       ;SEC. TIMER
                     0051 ;*****************************************************
                     0052 ;DO NOT CHANGE RELATIVE POSITION OF NEXT 6 BYTES
000A                 0053         MTMR    EQU     0A      ;MIN. TIMER
000B                 0054         HTMR    EQU     0B      ;HOUR TIMER
                     0055 ;DEFINE ALARM TIME MODE REGS (ATM)
000C                 0056         MALARM  EQU     0C      ;MIN. ALARM
000D                 0057         HALARM  EQU     0D      ;HOUR ALARM
                     0058 ;DEFINE DATA ENTRY MODE REGS (DEM)
```

```
000E              MENTRY  EQU     0E      ;MIN. ENTRY
000F              HENTRY  EQU     0F      ;HOUR ENTRY
          ;*****************************************************
          ;
          ;DEFINE FLAG REG AND FUNCTION:
0010              FLAG    EQU     10
          ;       BIT # 7|6|5|4|3|2|1|0|
          ;———————|-|-|-|-|-|-|-|
          ;             X|X|X|X|X|X|0|0| —> REAL TIME MODE (RTM)
          ;             X|X|X|X|X|X|0|1| —> ALARM TIME MODE(ATM)
          ;             X|X|X|X|X|X|1|0| —> DATA ENTRY M0DE(DEM)
          ;             X|X|X|X|X|X|1|1| —> TEST MODE (TM)
          ;             X|X|X|X|X|Y|X|X| —> ALRMLED ON/OFF
          ;             X|X|X|X|Y|X|X|X| —> COLON LED ON/OFF
          ;             X|X|X|Y|X|X|X|X| —> FLASH DISPLAY
          ;             X|X|Y|X|X|X|X|X| —> KEY_BEEP
          ;             X|Y|X|X|X|X|X|X| —> KEY_HIT (0/1)
          ;             Y|X|X|X|X|X|X|X| —> SERVICED
          ; X = DEFINED ELSEWHERE IN TABLE
          ; Y = DEFINED AS SHOWN (0/1)
          ;
0011              TEMP    EQU     11
0012              DIGIT   EQU     12
0013              NEW_KEY EQU     13
0014              KEY_NIBL EQU    14
0015              DEBOUNCE EQU    15
0016              MIN_SEC EQU     16      ;MIN/SECONDS TIMER
0017              ENTFLG  EQU     17
          ;flag dedicated to the key entry mode
          ;       BIT # 7|6|5|4|3|2|1|0|
          ;———————|-|-|-|-|-|-|-|
          ;             X|X|X|X|X|X|X|Y| —> REAL/ALARM TIME STATUS
          ;             X|X|X|X|X|X|Y|X| —> HR10 DONE
          ;             X|X|X|X|X|Y|X|X| —> HR DONE
          ;             X|X|X|X|Y|X|X|X| —> MIN10 DONE
          ;             X|X|X|Y|X|X|X|X| —> MIN DONE
          ;             X|X|Y|X|X|X|X|X| —> INKEYBEP
          ;             X|Y|X|Y|X|X|X|X| —> NOT USED
          ;             Y|X|X|X|X|X|X|X| —> NOT USED
          ;
          ;
0018              ALFLAG  EQU     18
          ;flag dedicated to the alarm
          ;       BIT # 7|6|5|4|3|2|1|0|
          ;———————|-|-|-|-|-|-|-|
          ;             X|X|X|X|X|X|X|Y| —> ALONOF
          ;             X|X|X|X|X|X|Y|X| —> INAL
          ;             X|X|X|X|X|Y|X|X| —> SILNC
          ;             X|X|X|X|Y|X|X|X| —> INAA
          ;             X|X|X|Y|X|X|X|X| —> NOT USED
          ;             X|X|Y|X|X|X|X|X| —> NOT USED
          ;             X|Y|X|Y|X|X|X|X| —> NOT USED
          ;             Y|X|X|X|X|X|X|X| —> NOT USED
          ;
0019              AAFLAG  EQU     19
          ;flag dedicated to the AA alarm
001A              AATMR   EQU     1A
          ;
          ;Port pin definitions:
          ;
          ;PORT_A:
          ;       BIT 0   —> BEEPER (ACTIVE LOW) OUTPUT
          ;       BIT 1-3 —> unused I/O
          ;
          ;PORT_B: ALL OUTPUTS
          ;       BIT 0&4 —> MSB DIGIT COMMON CATHODE & ALARM
          ;       BIT 1&5 —> 2ND DIGIT COMMOM CATHODE & COLON
          ;       BIT 2&6 —> 3RD DIGIT COMMON CATHODE & PM
          ;       BIT 3&7 —> LSB DIGIT COMMON CATHODE & AM
```

```
            ;
            ;PORT_C:
            ;IN DISPLAY MODE ALL SEG/ANNN SET AS OUTPUTS
            ;IN KEY SCAN MODE COLS ARE OUTPUTS ROWS ARE INPUTS
            ;       BIT 0   —> SEGMENT A & COL 4
            ;       BIT 1   —> SEGMENT B & COL 3
            ;       BIT 2   —> SEGMENT C & COL 2
            ;       BIT 3   —> SEGMENT D & COL 1
            ;       BIT 4   —> SEGMENT E & ROW 4
            ;       BIT 5   —> SEGMENT F & ROW 3
            ;       BIT 6   —> SEGMENT G & ROW 2
            ;       BIT 7   —> CA OF ALL ANNUNCIATORS & ROW 1
            ;
            ;
            ;
                    ORG     0
            START
0000 0AFC           GOTO    INIT_CLK        ;INITIALIZE CLOCK
            ;THIS ROUTINE RUNS A TEST ON THE LEDS.
            ;ALL THE RELEVENT LEDS ARE LIT UP FOR 2 SECS.
            ;
            TEST_HARDWARE
0001 0C02           MOVLW   d’02'           ;DISPLAY FOR 2 SECS
0002 0036           MOVWF   MIN_SEC         ;       /
            ;
            ;
            TEST_LOOP
0003 0216           MOVF    MIN_SEC,W       ;GET MIN/SEC
0004 0643           BTFSC   STATUS,Z        ;NOT 0 THEN SKIP
0005 0A0B           GOTO    NORM_TIME       ;ELSE NORMAL TIME
0006 0925           CALL    UPDATE_DISPLAY  ;UPDATE DISPLAY
0007 05A3           BSF     STATUS,PA0      ;GOTO PAGE 1
0008 0900           CALL    UPDATE_TIMERS   ;WAIT AND UPDATE
0009 04A3           BCF     STATUS,PA0      ;RESET PAGE MARKER
000A 0A03           GOTO    TEST_LOOP       ;LOOP BACK
            NORM_TIME
000B 0410           BCF     FLAG,0          ;PUT IN REAL TIME
000C 0430           BCF     FLAG,1
            TIME_LOOP
000D 0925           CALL    UPDATE_DISPLAY
000E 05C3           BSF     STATUS,PA1      ;GOTO PAGE 2
000F 0900           CALL    SERVICE_KEYS
0010 05A3           BSF     STATUS,PA0      ;GOTO PAGE 3
0011 0900           CALL    SOUND_AA        ;CHECK ALARM
0012 04C3           BCF     STATUS,PA1      ;GOTO PAGE 1
0013 0900           CALL    UPDATE_TIMERS   ;WAIT AND UPDATE TIMERS
0014 04A3           BCF     STATUS,PA0      ;RESET PAGE MARKER
0015 04C3           BCF     STATUS,PA1      ;       /
0016 0210           MOVF    FLAG,W          ;SEE IF IN ATM
0017 0E03           ANDLW   B’00000011'     ;       /
0018 0F01           XORLW   B’00000001'     ;      /
0019 0643           BTFSC   STATUS,Z        ;SKIP IF NOT
001A 091C           CALL    RESET_ATM
001B 0A0D           GOTO    TIME_LOOP
            ;
            RESET_ATM
001C 0216           MOVF    MIN_SEC,W       ;GET MIN/SEC
001D 0E0F           ANDLW   B’00001111'     ;       /
001E 0743           BTFSS   STATUS,Z        ;Z THEN SKIP
001F 0800           RETLW   0               ;ELSE RETURN
0020 0410           BCF     FLAG,0          ;SET TO RTM
0021 0450           BCF     FLAG,ALRMLED    ;CLEAR LED
0022 0618           BTFSC   ALFLAG,ALONOF   ;TEST STAT
0023 0550           BSF     FLAG,ALRMLED    ;SET LED
0024 0800           RETLW   0               ;RETURN
            ;
            ;
            UPDATE_DISPLAY
0025 0C00           MOVLW   B’00000000'     ;CLEAR SEG DRIVE
```

```
0026 0027            0200         MOVWF   PORT_C          ;       /
0027 0C3F            0201         MOVLW   B’00111111'     ;SEE IF LAST DIGIT
0028 0186            0202         XORWF   PORT_B,0        ;       /
0029 0643            0203         BTFSC   STATUS,Z        ;NO THEN SKIP
002A 0A6F            0204         GOTO    SCAN_KP         ;ELSE SCAN KEYPAD
                     0205 UP_DSP_1
                     0206 ;SELECT DIGIT TO BE DISPLAYED
002B 0246            0207         COMF    PORT_B,0        ;GET COMPL. PORT B IN W
002C 0643            0208         BTFSC   STATUS,Z        ;NO DIGIT SELECTED?
002D 0CC0            0209         MOVLW   B’11000000'     ;THEN SELECT DEFAULT
002E 0031            0210         MOVWF   TEMP            ;SAVE IN TEMP
002F 0271            0211         COMF    TEMP            ;COMPLEMENT VALUE
0030 0503            0212         BSF     STATUS,C        ;SET CARRY
0031 0371            0213         RLF     TEMP            ;SHIFT LEFT
0032 0703            0214         BTFSS   STATUS,C        ;IF C=1 THEN SKIP
0033 0371            0215         RLF     TEMP            ;ELSE 3 TIMES...
0034 0371            0216         RLF     TEMP            ;THRU CARRY
0035 0211            0217         MOVF    TEMP,0          ;GET IN W
0036 0026            0218         MOVWF   PORT_B          ;OUTPUT TO PORT
                     0219 ;NOW THAT DIGIT IS SELECTED, SELECT SEG VALUES FOR THAT DIGIT
                     0220 ;FIRST FIND MODE OF OPERATION.
0037 0C0A            0221         MOVLW   MTMR            ;LOAD FSR WITH MTMR
0038 0024            0222         MOVWF   FSR             ;       /
0039 0210            0223         MOVF    FLAG,0          ;GET FLAG IN W
003A 0E03            0224         ANDLW   B’00000011'     ;MASK OTHER BITS
003B 0031            0225         MOVWF   TEMP            ;SAVE IN TEMP
003C 0F03            0226         XORLW   B’00000011'     ;IN TEST MODE
003D 0643            0227         BTFSC   STATUS,Z        ;NO THEN SKIP
003E 0A4B            0228         GOTO    DO_TM           ;ELSE TEST MODE
003F 0403            0229         BCF     STATUS,C        ;CLEAR CARRY
0040 0371            0230         RLF     TEMP            ;LEFT SHIFT TEMP
0041 0211            0231         MOVF    TEMP,0          ;GET IN W
0042 01E4            0232         ADDWF   FSR             ;CHANGE INDIRECT POINTER
0043 0954            0233         CALL    GET_7_SEG       ;GET 7 SEG DATA IN W
0044 0032            0234         MOVWF   DIGIT           ;SAVE IN DIGIT LOC.
0045 09D1            0235         CALL    MASK_ANNC       ;MASK ANNC TO DIGIT
0046 0690            0236         BTFSC   FLAG,FLASH      ;NO FLASH THEN SKIP
0047 094E            0237         CALL    CHK_HALF_SEC    ;ELSE CHK. IF ON
0048 0212            0238         MOVF    DIGIT,0         ;GET BACK DIGIT
0049 0027            0239         MOVWF   PORT_C          ;OUTPUT TO PORT
004A 0800            0240         RETLW   0               ;RETURN
                     0241 ;
                     0242 DO_TM
004B 0CFF            0243         MOVLW   B’11111111'     ;LIGHT ALL SEGMENTS
004C 0027            0244         MOVWF   PORT_C          ;       /
004D 0800            0245         RETLW   0               ;RETURN FROM UPDATE DISPLAY
                     0246 ;
                     0247 CHK_HALF_SEC
004E 0770            0248         BTFSS   FLAG,COLON      ;IF COLON ON THEN DO
004F 0A51            0249         GOTO    BLANK_DSP       ;ELSE BLANK DISPLAY
0050 0800            0250         RETLW   0
                     0251 BLANK_DSP
0051 0C00            0252         MOVLW   B’00000000'     ;MAKE PORT C LOW
0052 0032            0253         MOVWF   DIGIT
0053 0800            0254         RETLW   0
                     0255 ;
                     0257 ;
                     0258 ;ON ENTRY FSR POINTS TO THE REAL TIME MODE’S MINUTES REGISTER.
                     0259 ;ON RETURN FSR POINTS TO THE TIMER REGISTER TO BE DISPLAYED.
                     0260 ;W REG. CONTAINS THE DECODED 7 SEG. INFO OF THE DIGIT
                     0261 ;TO BE DISPLAYED
                     0262 ;
                     0263 GET_7_SEG
0054 0246            0264         COMF    PORT_B,0        ;COMPLEMENT B —> W
0055 0EF0            0265         ANDLW   B’11110000'     ;MASK LO NIBBLE
0056 0643            0266         BTFSC   STATUS,Z        ;NZ THEN SKIP
0057 02A4            0267         INCF    FSR             ;INC POINTER
0058 0200            0268         MOVF    F0,0            ;MOVE INDIRECT TO W
0059 0031            0269         MOVWF   TEMP            ;GET INTO TEMP
```

```
005A 0246                   COMF    PORT_B,0        ;COMPL.B —> W
005B 0EF0                   ANDLW   B’11110000'     ;MASK LO NIBBLE
005C 0643                   BTFSC   STATUS,Z        ;IF D1/2 THEN
005D 04F1                   BCF     TEMP,AM_PM      ;CLEAR AM/PM BIT
005E 0246                   COMF    PORT_B,0        ;GET PORT B AGAIN
005F 0ECC                   ANDLW   B’11001100'     ;SEE IF D2 OR D4
0060 0643                   BTFSC   STATUS,Z        ;YES THEN SKIP
0061 03B1                   SWAPF   TEMP            ;SWAP TEMP
0062 0C0F                   MOVLW   B’00001111'     ;MASK HI NIBBLE
0063 0151                   ANDWF   TEMP,0
0064 01E2                   ADDWF   PC              ;ADD TO PC
0065 083F                   RETLW   B’00111111'     ;CODE FOR 0
0066 0806                   RETLW   B’00000110'     ;CODE FOR 1
0067 085B                   RETLW   B’01011011'     ;CODE FOR 2
0068 084F                   RETLW   B’01001111'     ;CODE FOR 3
0069 0866                   RETLW   B’01100110'     ;CODE FOR 4
006A 086D                   RETLW   B’01101101'     ;CODE FOR 5
006B 087D                   RETLW   B’01111101'     ;CODE FOR 6
006C 0807                   RETLW   B’00000111'     ;CODE FOR 7
006D 087F                   RETLW   B’01111111'     ;CODE FOR 8
006E 0867                   RETLW   B’01100111'     ;CODE FOR 9
                    ;
                    ;This routine scans the 4x4 hex key pad for a key hit.
                    ;If key is pressed, KEY_HIT flag is set and the value of
                    ;the hex key is returned in reg NEW_KEY
                    ;If no key is detected, then a 0xff value is returned in
                    ;register NEW_KEY and the flag KEY_HIT is reset.
                    ;
                    SCAN_KP
006F 06D0                   BTFSC   FLAG,KEY_HIT    ;KEY UNDER SERVICE?
0070 0A2B                   GOTO    UP_DSP_1        ;YES SKIP ROUTINE
0071 0CFF                   MOVLW   B’11111111'     ;SET DIGIT SINKS ...
0072 0026                   MOVWF   PORT_B          ;TO HIGH
0073 0CF7                   MOVLW   B’11110111'     ;SET KEY COL LOW
0074 0031                   MOVWF   TEMP            ;SAVE IN TEMP
                    SKP1
0075 0C00                   MOVLW   B’00000000'     ;SET PORT C AS OUTPUTS
0076 0007                   TRIS    PORT_C          ;       /
0077 0211                   MOVF    TEMP,W
0078 0E0F                   ANDLW   B’00001111'     ;DISCHARGE PINS
0079 0027                   MOVWF   PORT_C          ;       /
007A 0CF0                   MOVLW   B’11110000'     ;SET AS I/O
007B 0007                   TRIS    PORT_C          ;   /
007C 0211                   MOVF    TEMP,W          ;GET OLD VALUE
007D 0027                   MOVWF   PORT_C          ;OUTPUT TO PORT
007E 0207                   MOVF    PORT_C,W        ;INPUT PORT VALUE
007F 0EF0                   ANDLW   B’11110000'     ;MASK LO BYTE
0080 0FF0                   XORLW   B’11110000'     ;SEE IF KEY HIT
0081 0743                   BTFSS   STATUS,Z        ;NO KEY THEN SKIP
0082 0A8D                   GOTO    DET_KEY         ;LOAD KEY VALUE
                    SKP3
0083 0503                   BSF     STATUS,C        ;SET CARRY
0084 0331                   RRF     TEMP            ;MAKE NEXT COL. LOW
0085 0603                   BTFSC   STATUS,C        ;ALL DONE THEN SKIP
0086 0A75                   GOTO    SKP1
0087 0073                   CLRF    NEW_KEY         ;SET NEW_KEY = FF
0088 00F3                   DECF    NEW_KEY         ;       /
                    SKP2
0089 0067                   CLRF    PORT_C          ;SETPORT C AS ...
008A 0C00                   MOVLW   B’00000000'     ;OUTPUTS
008B 0007                   TRIS    PORT_C          ;       /
008C 0A2B                   GOTO    UP_DSP_1        ;RETURN
                    DET_KEY
                    ;key is detected
008D 0293                   INCF    NEW_KEY,W       ;CHK IF KEY ...
008E 0743                   BTFSS   STATUS,Z        ;WAS RELEASED
008F 0A89                   GOTO    SKP2            ;NO THEN RETURN
0090 0207                   MOVF    PORT_C,W        ;GET RAW KEY...
0091 0D0F                   IORLW   B’00001111'     ;VALUE.
```

```
0092 0151            0339         ANDWF   TEMP,W          ;   /
0093 0033            0340         MOVWF   NEW_KEY         ;SAVE IN NEW_KEY
0094 0998            0341         CALL    GET_KEY_VAL     ;GET ACTUAL KEY ...
0095 0033            0342         MOVWF   NEW_KEY         ;VALUE
0096 05D0            0343         BSF     FLAG,KEY_HIT    ;SET KEY HIT FLAG
0097 0A89            0344         GOTO    SKP2            ;RETURN
                     0345 ;
                     0347 ;This routine decodes the hex value from the “raw” data got
                     0348 ;from scanning the rows and cols.
                     0349 ;   actual key value          raw hex value
                     0350 ;       ONE              EQU     77
                     0351 ;       TWO              EQU     7B
                     0352 ;       THREE            EQU     7D
                     0353 ;       C                EQU     7E
                     0354 ;       FOUR             EQU     0B7
                     0355 ;       FIVE             EQU     0BB
                     0356 ;       SIX              EQU     0BD
                     0357 ;       D                EQU     0BE
                     0358 ;       SEVEN            EQU     0D7
                     0359 ;       EIGHT            EQU     0DB
                     0360 ;       NINE             EQU     0DD
                     0361 ;       E                EQU     0DE
                     0362 ;       A                EQU     0E7
                     0363 ;       ZERO             EQU     0EB
                     0364 ;       B                EQU     0ED
                     0365 ;       F                EQU     0EE
                     0366 ;
                     0367 ;
                     0368 GET_KEY_VAL
0098 0E0F            0369         ANDLW   B’00001111'     ;GET LO NIBBLE
0099 0034            0370         MOVWF   KEY_NIBL        ;SAVE
009A 0C04            0371         MOVLW   4               ;SET COUNT TO 4
009B 0031            0372         MOVWF   TEMP            ;       /
                     0373 GKV1
009C 0503            0374         BSF     STATUS,C        ;SET CARRY
009D 0334            0375         RRF     KEY_NIBL        ;ROTATE NIBBLE
009E 0703            0376         BTFSS   STATUS,C        ;SKIP IF NOT Z
009F 0AA5            0377         GOTO    GET_HI_KEY      ;GOTO NEXT PART
00A0 02F1            0378         DECFSZ  TEMP            ;DEC COUNT
00A1 0A9C            0379         GOTO    GKV1            ;LOOP
                     0380 GO_RESET
00A2 05A3            0381         BSF     STATUS,PA0      ;SET MSB
00A3 05C3            0382         BSF     STATUS,PA1      ;       /
00A4 0BFF            0383         GOTO    SYS_RESET       ;ELSE BIG ERROR
                     0384 GET_HI_KEY
00A5 00F1            0385         DECF    TEMP            ;REDUCE BY 1
00A6 0393            0386         SWAPF   NEW_KEY,W       ;GET HI NIBBLE
00A7 0E0F            0387         ANDLW   B’00001111'     ;       /
00A8 0034            0388         MOVWF   KEY_NIBL        ;SAVE
00A9 0211            0389         MOVF    TEMP,W          ;GET OFFSET TO TBL
00AA 01E2            0390         ADDWF   PC              ;LOAD IN PC
00AB 0AAF            0391         GOTO    GET147A         ;JUMP TO NEXT PART
00AC 0AB8            0392         GOTO    GET2580         ;       /
00AD 0ABA            0393         GOTO    GET369B         ;     /
00AE 0ABC            0394         GOTO    GETCDEF         ;   /
                     0395 ;
                     0396 GET147A
00AF 0C04            0397         MOVLW   4               ;SET COUNT TO 4
                     0398 GETCOM
00B0 0031            0399         MOVWF   TEMP            ;
                     0400 GETCOM1
00B1 0503            0401         BSF     STATUS,C        ;SET CARRY
00B2 0334            0402         RRF     KEY_NIBL        ;ROTATE RIGHT
00B3 0703            0403         BTFSS   STATUS,C        ;CHECK IF DONE
00B4 0ABE            0404         GOTO    KEY_TBL         ;JUMP TO TABLE
00B5 02F1            0405         DECFSZ  TEMP            ;DEC COUNT
00B6 0AB1            0406         GOTO    GETCOM1         ;LOOP
00B7 0AA2            0407         GOTO    GO_RESET        ;ELSE ERROR
                     0408 ;
                     0409 GET2580
00B8 0C08            0410         MOVLW   8               ;SET COUNT TO 8
00B9 0AB0            0411         GOTO    GETCOM
                     0412 ;
                     0413 GET369B
00BA 0C0C            0414         MOVLW   D’12'           ;SET COUNT TO 12
00BB 0AB0            0415         GOTO    GETCOM
                     0416 ;
                     0417 GETCDEF
00BC 0C10            0418         MOVLW   D’16'           ;SET COUNT TO 16
00BD 0AB0            0419         GOTO    GETCOM
                     0421 ;
                     0422 KEY_TBL
00BE 00F1            0423         DECF    TEMP            ;REDUCE BY 1
00BF 0211            0424         MOVF    TEMP,W          ;GET IN W
00C0 01E2            0425         ADDWF   PC              ;JUMP TO TABLE
00C1 0801            0426         RETLW   1               ;KEY 1
00C2 0804            0427         RETLW   4               ;KEY 4
00C3 0807            0428         RETLW   7               ;KEY 7
00C4 080A            0429         RETLW   0A              ;KEY A
00C5 0802            0430         RETLW   2               ;KEY 2
00C6 0805            0431         RETLW   5               ;KEY 5
00C7 0808            0432         RETLW   8               ;KEY 8
00C8 0800            0433         RETLW   0               ;KEY 0
00C9 0803            0434         RETLW   3               ;KEY 3
00CA 0806            0435         RETLW   6               ;KEY 6
00CB 0809            0436         RETLW   9               ;KEY 9
00CC 080B            0437         RETLW   0B              ;KEY B
00CD 080C            0438         RETLW   0C              ;KEY C
00CE 080D            0439         RETLW   0D              ;KEY D
00CF 080E            0440         RETLW   0E              ;KEY E
00D0 080F            0441         RETLW   0F              ;KEY F
                     0442 ;
                     0444 ;
                     0445 MASK_ANNC
00D1 0CFC            0446         MOVLW   B’11111100'     ;CHK IF DIGIT 1
00D2 0186            0447         XORWF   PORT_B,0        ;       /
00D3 0643            0448         BTFSC   STATUS,Z        ;NO THEN SKIP
00D4 0AE5            0449         GOTO    MASK_ALARM      ;ELSE MASK ALARM
00D5 0CF3            0450         MOVLW   B’11110011'     ;CHK IF DIGIT 2
00D6 0186            0451         XORWF   PORT_B,0        ;       /
00D7 0643            0452         BTFSC   STATUS,Z        ;NO THEN SKIP
00D8 0AE8            0453         GOTO    MASK_COLON      ;ELSE MASK COLON
00D9 0CCF            0454         MOVLW   B’11001111'     ;CHK IF DIGIT 3
00DA 0186            0455         XORWF   PORT_B,0        ;       /
00DB 0643            0456         BTFSC   STATUS,Z        ;NO THEN SKIP
00DC 0AE1            0457         GOTO    MASK_PM         ;ELSE MASK PM
                     0458 MASK_AM
00DD 02A4            0459         INCF    FSR             ;INC FSR
00DE 07E0            0460         BTFSS   F0,AM_PM        ;IF 0 THEN AM
00DF 05F2            0461         BSF     DIGIT,7         ;SET MSB
00E0 0AEB            0462         GOTO    BLNK_LEAD_0     ;NEXT
                     0463 MASK_PM
00E1 02A4            0464         INCF    FSR             ;INC FSR
00E2 06E0            0465         BTFSC   F0,AM_PM        ;IF 1 THEN PM
00E3 05F2            0466         BSF     DIGIT,7         ;SET MSB
00E4 0AEB            0467         GOTO    BLNK_LEAD_0     ;NEXT
                     0468 MASK_ALARM
00E5 0650            0469         BTFSC   FLAG,ALRMLED    ;1 THEN LIGHT LED
00E6 05F2            0470         BSF     DIGIT,7         ;      /
00E7 0AEB            0471         GOTO    BLNK_LEAD_0     ;NEXT
                     0472 MASK_COLON
00E8 0670            0473         BTFSC   FLAG,COLON      ;1 THEN LIGHT LED
00E9 05F2            0474         BSF     DIGIT,7         ;       /
00EA 0AEB            0475         GOTO    BLNK_LEAD_0     ;NEXT
                     0476 ;
                     0477 BLNK_LEAD_0
00EB 0210            0478         MOVF    FLAG,W          ;GET IN W
00EC 0E03            0479         ANDLW   B’00000011'     ;SEE IF IN DEM
```

```
00ED 0F02           0480         XORLW   B’00000010'     ;CHECK
00EE 0643           0481         BTFSC   STATUS,Z        ;NO THEN DO
00EF 0800           0482         RETLW   0               ;ELSE RETURN
00F0 0CFC           0483         MOVLW   B’11111100'     ;SEE IF DIGIT 1
00F1 0186           0484         XORWF   PORT_B,0        ;       /
00F2 0743           0485         BTFSS   STATUS,Z        ;YES THEN SKIP
00F3 0800           0486         RETLW   0               ;RETURN
00F4 0C3F           0487         MOVLW   B’00111111'     ;ELSE MASK G AND ANUNC
00F5 0152           0488         ANDWF   DIGIT,0         ;GET IN W
00F6 0F3F           0489         XORLW   B’00111111'     ;SEE IF 0
00F7 0743           0490         BTFSS   STATUS,Z        ;YES THEN SKIP
00F8 0800           0491         RETLW   0               ;RETURN
00F9 0C80           0492         MOVLW   B’10000000'     ;ELSE BLANK D1
00FA 0172           0493         ANDWF   DIGIT           ;       /
00FB 0800           0494         RETLW   0               ;RETURN
                    0495 ;
                    0496 ;
                    0497 ;
                    0499 ;
                    0500 ;THIS ROUTINE SETS UP PORTS A,B,C AND THE INTERNAL
                    0501 ;REAL TIME CLOCK COUNTER.
                    0502 INIT_CLK
00FC 0C0F           0503         MOVLW   B’00001111'     ;MAKE ACTIVE HIGH
00FD 0025           0504         MOVWF   PORT_A          ;       /
00FE 0C00           0505         MOVLW   B’00000000'     ;SET PORT A AS OUTPUTS
00FF 0005           0506         TRIS    PORT_A
                    0507 ;
0100 0CFF           0508         MOVLW   B’11111111'     ;SET LEVELS HIGH
0101 0026           0509         MOVWF   PORT_B          ;       /
0102 0C00           0510         MOVLW   B’00000000'     ;SET PORT B AS OUTPUTS
0103 0006           0511         TRIS    PORT_B
                    0512 ;
0104 0C00           0513         MOVLW   B’00000000'     ;SET LEVELS LOW
0105 0027           0514         MOVWF   PORT_C          ;       /
0106 0C00           0515         MOVLW   B’00000000'     ;SET PORT C AS OUTPUTS
0107 0007           0516         TRIS    PORT_C          ;       /
                    0517 ;
0108 0C04           0518         MOVLW   B’00000100'     ;SET UP PRESCALER
0109 0002           0519         OPTION                  ;       /
                    0520 ;
010A 0C60           0521         MOVLW   MSEC5           ;RTCC = 5 mSEC
010B 0021           0522         MOVWF   RTCC            ;       /
010C 0068           0523         CLRF    MSTMR           ;CLEAR MSTMR
010D 0069           0524         CLRF    STMR            ; & SEC TMR
010E 006A           0525         CLRF    MTMR            ;& MINUTES
010F 0C12           0526         MOVLW   12H             ;MAKE HRS = 12
0110 002B           0527         MOVWF   HTMR            ;     /
0111 002D           0528         MOVWF   HALARM          ;MAKE HRS = 12
0112 006C           0529         CLRF    MALARM          ;       /
0113 0C03           0530         MOVLW   B’00000011'     ;SET TO TEST MODE
0114 0030           0531         MOVWF   FLAG            ;       /
0115 0078           0532         CLRF    ALFLAG          ;CLEAR ALL FLAG
0116 0079           0533         CLRF    AAFLAG          ;       /
0117 0077           0534         CLRF    ENTFLG          ;      /
0118 0A01           0535         GOTO    TEST_HARDWARE
                    0536 ;
                    0537 ;All routines related to timer updates are located at
                    0538 ;address 200 and above.
                    0540         ORG     0200
                    0541 ;
                    0542 UPDATE_TIMERS
0200 0201           0543         MOVF    RTCC,W          ;SEE IF RTCC = 0
0201 0743           0544         BTFSS   STATUS,Z        ;IF 0 THEN SKIP
0202 0A00           0545         GOTO    UPDATE_TIMERS   ;ELSE LOOP
0203 0C60           0546         MOVLW   MSEC5           ;RTCC = 5 mSEC
0204 0021           0547         MOVWF   RTCC            ;       /
0205 02A8           0548         INCF    MSTMR           ;INC 5 MILLI SEC
0206 06D0           0549         BTFSC   FLAG,KEY_HIT    ;NO KEY HIT THEN SKIP
0207 0A70           0550         GOTO    CHK_DE_BOUNCE   ;ELSE DEBOUNCE
```

```
                     0551 UP_TMR_1
0208 0210            0552         MOVF    FLAG,W          ;ALARM MODE?
0209 0E03            0553         ANDLW   B’00000011'     ;       /
020A 0F01            0554         XORLW   B’00000001'     ;       /
020B 0743            0555         BTFSS   STATUS,Z        ;SKIP IF YES
020C 0A14            0556         GOTO    UP_TMR_2        ;DO NEXT
020D 0550            0557         BSF     FLAG,ALRMLED    ;LIGHT LED
020E 0570            0558         BSF     FLAG,COLON      ;       /
020F 0C64            0559         MOVLW   D’100'          ;IF 1/2 SEC
0210 0088            0560         SUBWF   MSTMR,0         ; BLINK
0211 0703            0561         BTFSS   STATUS,C        ;  /
0212 0450            0562         BCF     FLAG,ALRMLED    ;ALARM LED
0213 0A19            0563         GOTO    UP_TMR_3        ;SKIP
                     0564 UP_TMR_2
0214 0570            0565         BSF     FLAG,COLON      ;TURN ON
0215 0C64            0566         MOVLW   D’100'          ;<100 BLINK COLON
0216 0088            0567         SUBWF   MSTMR,0         ;       /
0217 0703            0568         BTFSS   STATUS,C        ;YES THEN SKIP
0218 0470            0569         BCF     FLAG,COLON      ;ELSE TURN OFF
                     0570 UP_TMR_3
0219 0208            0571         MOVF    MSTMR,0         ;GET MSTMR IN W
021A 0FC8            0572         XORLW   D’200'          ;= 200 THEN SKIP
021B 0743            0573         BTFSS   STATUS,Z        ;       /
021C 0800            0574         RETLW   0
                     0575 ;INC SECONDS COUNT
021D 0068            0576         CLRF    MSTMR           ;CLEAR MS_TMR
021E 0216            0577         MOVF    MIN_SEC,W       ;GET MIN_SEC TIMER
021F 0E0F            0578         ANDLW   B’00001111'     ;MASK MINUTES
0220 0743            0579         BTFSS   STATUS,Z        ;ZERO THEN SKIP
0221 00F6            0580         DECF    MIN_SEC         ;REDUCE SECONDS
0222 0C09            0581         MOVLW   STMR            ;LOAD FSR WITH S_TMR
0223 0024            0582         MOVWF   FSR             ;       /
0224 0955            0583         CALL    INC_60          ;INC SECONDS
0225 0D00            0584         IORLW   0               ;DO AN OPERATION
0226 0743            0585         BTFSS   STATUS,Z        ;IF RETURN = 0 SKIP
0227 0A38            0586         GOTO    CHK_AL_TIM      ;CHK ALRM
                     0587 ;INC MINUTES COUNT
0228 03B6            0588         SWAPF   MIN_SEC         ;SWAP MIN SEC
0229 0216            0589         MOVF    MIN_SEC,W       ;GET MIN_SEC IN W
022A 0E0F            0590         ANDLW   B’00001111'     ;MASK SECONDS
022B 0743            0591         BTFSS   STATUS,Z        ;SKIP IF NOT SET
022C 00F6            0592         DECF    MIN_SEC         ;ELSE DEC
022D 03B6            0593         SWAPF   MIN_SEC         ;SWAP BACK
022E 0966            0594         CALL    CHK_SILNC_TIM   ;SILNCE ON?
022F 0C0A            0595         MOVLW   MTMR            ;INC MINUTES
0230 0024            0596         MOVWF   FSR             ;       /
0231 0955            0597         CALL    INC_60          ;       /
0232 0D00            0598         IORLW   0               ;DO AN OPERATION
0233 0743            0599         BTFSS   STATUS,Z        ;IF 0 THEN SKIP
0234 0A38            0600         GOTO    CHK_AL_TIM      ;CHECK ALRAM TIME
                     0601 ;INC HOUR COUNT
0235 0C0B            0602         MOVLW   HTMR            ;GET HTMR IN FSR
0236 0024            0603         MOVWF   FSR
0237 0989            0604         CALL    INC_HR          ;INC HOURS
                     0605 ;
                     0606 CHK_AL_TIM
0238 0718            0607         BTFSS   ALFLAG,ALONOF   ;IF OFF QUIT
0239 0800            0608         RETLW   0               ;       /
023A 0658            0609         BTFSC   ALFLAG,SILNC    ;RET IF IN SILENCE
023B 0800            0610         RETLW   0
023C 0638            0611         BTFSC   ALFLAG,INAL     ;ALREADY DONE
023D 0A4D            0612         GOTO    CHK_1_MIN       ;SEE IF 1 MIN UP
                     0613 ;        RETLW   0               ;YES THEN QUIT
023E 020D            0614         MOVF    HALARM,W        ;CHK HRS
023F 018B            0615         XORWF   HTMR,W          ;EQUAL?
0240 0743            0616         BTFSS   STATUS,Z        ;YES THEN SKIP
0241 0800            0617         RETLW   0               ;ELSE RET
0242 020C            0618         MOVF    MALARM,W        ;CHK MIN
0243 018A            0619         XORWF   MTMR,W          ;EQUAL?
```

```
0244 0743            0620         BTFSS   STATUS,Z        ;YES THEN SKIP
0245 0800            0621         RETLW   0               ;ELSE RET
0246 0209            0622         MOVF    STMR,W          ;SEE IF SEC=0
0247 0743            0623         BTFSS   STATUS,Z        ;YES THEN SKIP
0248 0800            0624         RETLW   0               ;NO THEN RET
0249 0538            0625         BSF     ALFLAG,INAL     ;SET IN ALARM FLAG
024A 0C10            0626         MOVLW   10              ;SET 1 MIN TIMER
024B 0036            0627         MOVWF   MIN_SEC         ;       /
024C 0800            0628         RETLW   0
                     0629 ;
                     0630 CHK_1_MIN
024D 0396            0631         SWAPF   MIN_SEC,W       ;SWAP IN W
024E 0E0F            0632         ANDLW   B’00001111'     ;CHK MINUTES
024F 0743            0633         BTFSS   STATUS,Z        ;0 THEN SKIP
0250 0800            0634         RETLW   0               ;ELSE RET
0251 0438            0635         BCF     ALFLAG,INAL     ;CLR IN ALARM
0252 0478            0636         BCF     ALFLAG,INAA     ;CLR IN AA
0253 0505            0637         BSF     PORT_A,BEP      ;STOP BEEPER
0254 0800            0638         RETLW   0
                     0639 ;
                     0640 INC_60
0255 02A0            0641         INCF    F0              ;INC AND GET IN W
0256 0200            0642         MOVF    F0,0            ;       /
0257 0E0F            0643         ANDLW   B’00001111'     ;MASK HI BITS
0258 0F0A            0644         XORLW   B’00001010'     ;= 10 THEN MAKE IT 0
0259 0743            0645         BTFSS   STATUS,Z        ;       /
025A 0801            0646         RETLW   1               ;ELSE RETURN NON ZERO
025B 0CF0            0647         MOVLW   B’11110000'     ;ZERO LSB
025C 0160            0648         ANDWF   F0              ;  /
025D 03A0            0649         SWAPF   F0              ;SWAP INDIRECT
025E 02A0            0650         INCF    F0              ;INC
025F 0200            0651         MOVF    F0,0            ;GET IN W
0260 03A0            0652         SWAPF   F0              ;SWAP F0 BACK
0261 0F06            0653         XORLW   D’6'            ;=6 THEN SKIP
0262 0743            0654         BTFSS   STATUS,Z        ;       /
0263 0801            0655         RETLW   1               ;ELSE RETURN NZ
0264 0060            0656         CLRF    F0              ;       /
0265 0800            0657         RETLW   0               ;RET 0
                     0658 ;
                     0660 ;
                     0661 CHK_SILNC_TIM
0266 0758            0662         BTFSS   ALFLAG,SILNC    ;CHK IF IN SILENCE
0267 0800            0663         RETLW   0               ;NO THEN SKIP
0268 0396            0664         SWAPF   MIN_SEC,W       ;GET MIN IN W
0269 0E0F            0665         ANDLW   B’00001111'     ;MASK SECS
026A 0743            0666         BTFSS   STATUS,Z        ;ZERO?
026B 0800            0667         RETLW   0               ;NO THEN RET
026C 0458            0668         BCF     ALFLAG,SILNC    ;RESET SILENCE
026D 0C10            0669         MOVLW   10              ;SET I MIN TIMER
026E 0036            0670         MOVWF   MIN_SEC         ;       /
026F 0800            0671         RETLW   0
                     0672 ;
                     0673 ;
                     0674 CHK_DE_BOUNCE
0270 06B7            0675         BTFSC   ENTFLG,INKEYBEP ;IN KEY BEEP?
0271 0A76            0676         GOTO    CHK_DEB_1       ;YES THEN DEC TIMER
0272 07B0            0677         BTFSS   FLAG,KEY_BEEP   ;KEY BEEP SET?
0273 0A7F            0678         GOTO    CHK_SERV        ;NO, SEE IF SERVICED
0274 0678            0679         BTFSC   ALFLAG,INAA     ;IN AA?
0275 0A86            0680         GOTO    CHK_BEP_ON      ;YES THEN SEE IF ON
                     0681 CHK_DEB_1
0276 05B7            0682         BSF     ENTFLG,INKEYBEP ;SET FLAG
0277 0215            0683         MOVF    DEBOUNCE,W      ;GET IN W
0278 0643            0684         BTFSC   STATUS,Z        ;NZ THEN SKIP
0279 0C14            0685         MOVLW   D’20'           ;ELSE DB 100 mSEC
027A 0035            0686         MOVWF   DEBOUNCE        ;       /
027B 0405            0687         BCF     PORT_A,BEP      ;TURN ON BEEPER
027C 02F5            0688         DECFSZ  DEBOUNCE        ;DEC AND CHK
027D 0A08            0689         GOTO    UP_TMR_1        ;GO BACK
```

```
027E 0505                    BSF     PORT_A,BEP      ;TURN OFF BEEPER
                     CHK_SERV
                     ;        CLRF    DEBOUNCE
                     ;        BSF     PORT_A,BEP
027F 07F0                    BTFSS   FLAG,SERVICED   ;SERVICED THEN SKIP
0280 0A08                    GOTO    UP_TMR_1        ;GO BACK
0281 04F0                    BCF     FLAG,SERVICED   ;ELSE CLEAR FLAGS
0282 04D0                    BCF     FLAG,KEY_HIT    ;       /
0283 04B0                    BCF     FLAG,KEY_BEEP   ;RESET FLAG
0284 04B7                    BCF     ENTFLG,INKEYBEP ;    /
0285 0A08                    GOTO    UP_TMR_1        ;GO BACK
                     ;
                     CHK_BEP_ON
0286 0705                    BTFSS   PORT_A,BEP      ;IF OFF THEN SKIP
0287 0A08                    GOTO    UP_TMR_1        ;ELSE WAIT
0288 0A76                    GOTO    CHK_DEB_1       ;RETURN
                     ;
                     ;
                     INC_HR
0289 02A0                    INCF    F0              ;INC HOUR TIMER
028A 0200                    MOVF    F0,W            ;GET HR TMR IN W
028B 0031                    MOVWF   TEMP            ;SAVE IN TEMP
028C 0E0F                    ANDLW   B’00001111'     ;CHK LO BYTE = 10
028D 0F0A                    XORLW   D’10'           ;       /
028E 0743                    BTFSS   STATUS,Z        ;YES THEN SKIP
028F 0A93                    GOTO    INC_AM_PM       ;ELSE CHK 12
0290 0C10                    MOVLW   B’00010000'     ;LOAD 1 IN MSB
0291 0020                    MOVWF   F0
0292 0AA3                    GOTO    RESTORE_AM_PM   ;RESTORE AM/PM
                     INC_AM_PM
0293 04E0                    BCF     F0,AM_PM        ;CLEAR AM/PM
0294 0200                    MOVF    F0,W            ;GET IN W
0295 0F12                    XORLW   12H             ;SEE IF 12 HEX
0296 0743                    BTFSS   STATUS,Z        ;YES THEN SKIP
0297 0A9D                    GOTO    CHK_13          ;ELSE CHK 13
0298 07F1                    BTFSS   TEMP,AM_PM      ;IF SET, SKIP
0299 0A9C                    GOTO    SET_AM_PM       ;ELSE SET
029A 04E0                    BCF     F0,AM_PM        ;CLEAR FLAG
029B 0800                    RETLW   0               ;RETURN
                     SET_AM_PM
029C 05E0                    BSF     F0,AM_PM        ;SET FLAG
                     CHK_13
029D 0200                    MOVF    F0,W            ;GET IN W
029E 0F13                    XORLW   13H             ;SEE IF 13
029F 0743                    BTFSS   STATUS,Z        ;YES THEN SKIP
02A0 0AA3                    GOTO    RESTORE_AM_PM
                     SET_1_HR
02A1 0C01                    MOVLW   B’00000001'     ;SET TO 1
02A2 0020                    MOVWF   F0
                     RESTORE_AM_PM
02A3 06F1                    BTFSC   TEMP,AM_PM      ;SKIP IF AM
02A4 05E0                    BSF     F0,AM_PM        ;ELSE SET TO PM
02A5 0800                    RETLW   0
                     ;
                     ;
                     ;
                             ORG     400
                     ;
                     ;KEY DEFINITIONS
000A                         ALARM_KEY       EQU     0A
000B                         CE_KEY          EQU     0B
000C                         SNOOZE_KEY      EQU     0C
000D                         AM_PM_KEY       EQU     0D
000E                         CLR_ALARM_KEY   EQU     0E
000F                         SET_KEY         EQU     0F
                     ;
                     SERVICE_KEYS
0400 07D0                    BTFSS   FLAG,KEY_HIT    ;NO KEY HIT THEN ...
0401 0800                    RETLW   0               ;RETURN
```

```
0402 06F0                   BTFSC   FLAG,SERVICED   ;IF NOT SERVICED SKIP
0403 0800                   RETLW   0               ;ELSE RETURN
0404 05F0                   BSF     FLAG,SERVICED   ;SET SERVICED FLAG
0405 0210                   MOVF    FLAG,W          ;GET MODE OF OPERATION
0406 0E03                   ANDLW   B’00000011'     ;       /
0407 0643                   BTFSC   STATUS,Z        ;00 THEN RTM
0408 0A10                   GOTO    RTMKS           ;RTM KEY SERVICE
0409 0031                   MOVWF   TEMP            ;SAVE IN TEMP
040A 02F1                   DECFSZ  TEMP            ;REDUCE TEMP
040B 0A0D                   GOTO    SK1             ;SKIP
040C 0A1D                   GOTO    ATMKS           ;01, DO ALARM MODE
                    SK1
040D 02F1                   DECFSZ  TEMP            ;REDUCE TEMP
040E 0800                   RETLW   0               ;11 THEN RETURN
040F 0A2A                   GOTO    DEMKS           ;10, DATA ENTRY MODE
                    ;
                    ;REAL TIME MODE KEY SERVICE
                    RTMKS
0410 09BA                   CALL    CHK_AL_KEYS     ;CHK ALARM KEYS
0411 0D00                   IORLW   0               ;SEE IF NZ RET
0412 0643                   BTFSC   STATUS,Z        ;NZ THEN SKIP
0413 0800                   RETLW   0               ;ELSE RETURN
0414 0C0F                   MOVLW   SET_KEY         ;SEE IF SET KEY
0415 0193                   XORWF   NEW_KEY,W       ;       /
0416 0643                   BTFSC   STATUS,Z        ;NO THEN SKIP
0417 0A91                   GOTO    SERV_SET_RTM    ;SERVICE SET KEY
0418 0C0A                   MOVLW   ALARM_KEY       ;ALARM KEY?
0419 0193                   XORWF   NEW_KEY,W       ;    /
041A 0643                   BTFSC   STATUS,Z        ;NO THEN SKIP
041B 0AAB                   GOTO    SERV_ALARM_RTM  ;ELSE SERVICE ALARM
                    IGNORE_KEY
041C 0800                   RETLW   0               ;ELSE RETURN
                    ;
                    ;ALARM TIME MODE KEY SERVICE
                    ATMKS
041D 09BA                   CALL    CHK_AL_KEYS     ;CHECK ALRM KEYS
041E 0D00                   IORLW   0               ;CHECK IF 0
041F 0643                   BTFSC   STATUS,Z        ;NZ THEN SKIP
0420 0800                   RETLW   0               ;ELSE RETURN
0421 0C0F                   MOVLW   SET_KEY         ;SEE IF SET KEY
0422 0193                   XORWF   NEW_KEY,W       ;       /
0423 0643                   BTFSC   STATUS,Z        ;NO THEN SKIP
0424 0A9C                   GOTO    SERV_SET_ATM
0425 0C0A                   MOVLW   ALARM_KEY       ;GET ALARM KEY
0426 0193                   XORWF   NEW_KEY,W       ;SEE IF HIT
0427 0643                   BTFSC   STATUS,Z        ;NO THEN SKIP
0428 0AA2                   GOTO    SERV_ALARM_ATM  ;ELSE SERVICE
0429 0A1C                   GOTO    IGNORE_KEY
                    ;
                    ;DATA ENTRY MODE KEY SERVICE
                    DEMKS
042A 09BA                   CALL    CHK_AL_KEYS     ;CHECK ALARM KEYS
042B 0D00                   IORLW   0               ;CHK IF 0
042C 0643                   BTFSC   STATUS,Z        ;NZ THEN SKIP
042D 0800                   RETLW   0               ;ELSE RETURN
042E 0C0F                   MOVLW   SET_KEY         ;IF SET KEY THEN END
042F 0193                   XORWF   NEW_KEY,W       ;       /
0430 0643                   BTFSC   STATUS,Z        ;NO THEN SKIP
0431 0A3F                   GOTO    DEMKS_END       ;GOTO END
0432 0C0B                   MOVLW   CE_KEY          ;IF CLEAR ENTRY
0433 0193                   XORWF   NEW_KEY,W       ;       /
0434 0643                   BTFSC   STATUS,Z        ;SKIP IF NO
0435 0A48                   GOTO    DEMKS_END_1     ;ABANDON ENTRY
0436 0737                   BTFSS   ENTFLG,HR10     ;10’S HRS DONE?
0437 0A54                   GOTO    ENT_HR_10       ;NO THEN GET
0438 0757                   BTFSS   ENTFLG,HR       ;HRS DONE?
0439 0A5F                   GOTO    ENT_HRS         ;NO THEN GET
043A 0777                   BTFSS   ENTFLG,MIN10    ;10’S MIN. DONE?
043B 0A72                   GOTO    ENT_MIN_10      ;NO THEN GET
```

```
043C 0797            0829        BTFSS   ENTFLG,MIN      ;MIN DONE?
043D 0A7F            0830        GOTO    ENT_MIN         ;NO THEN GET
043E 0A87            0831        GOTO    ENT_AM_PM       ;NO THEN GET
                     0832 DEMKS_END
043F 0717            0833        BTFSS   ENTFLG,RTATS    ;GET OLD STATUS
0440 0A4D            0834        GOTO    LD_RTM          ;LOAD IN TIME
0441 020E            0835        MOVF    MENTRY,W        ;LD IN ALARM
0442 002C            0836        MOVWF   MALARM          ;       /
0443 020F            0837        MOVF    HENTRY,W        ;     /
0444 002D            0838        MOVWF   HALARM          ;   /
0445 0450            0839        BCF     FLAG,ALRMLED    ;CLEAR FLAG
0446 0618            0840        BTFSC   ALFLAG,ALONOF   ;SEE IF ON-OFF
0447 0550            0841        BSF     FLAG,ALRMLED    ;ELSE SET
                     0842 DEMKS_END_1
0448 0410            0843        BCF     FLAG,0          ;RTM MODE
0449 0430            0844        BCF     FLAG,1          ;    /
044A 0490            0845        BCF     FLAG,FLASH      ;STOP FLASH
                     0846 SERV_COM_RET
044B 05B0            0847        BSF     FLAG,KEY_BEEP
044C 0800            0848        RETLW   0               ;RETURN
                     0849 ;
                     0850 LD_RTM
044D 020E            0851        MOVF    MENTRY,W        ;LD IN RTM
044E 002A            0852        MOVWF   MTMR            ;       /
044F 020F            0853        MOVF    HENTRY,W        ;     /
0450 002B            0854        MOVWF   HTMR            ;   /
0451 0068            0855        CLRF    MSTMR           ;CLR TIME
0452 0069            0856        CLRF    STMR            ;  /
0453 0A48            0857        GOTO    DEMKS_END_1     ;GO BACK
                     0858 ;
                     0859 ENT_HR_10
0454 0213            0860        MOVF    NEW_KEY,W       ;SEE IF 0
0455 0643            0861        BTFSC   STATUS,Z        ;NZ THEN SKIP
0456 0A5C            0862        GOTO    LD_HENTRY_0     ;LOAD 0
0457 02D3            0863        DECFSZ  NEW_KEY,0       ;1 THE SKIP
0458 0A1C            0864        GOTO    IGNORE_KEY      ;ELSE IGNORE KEY
0459 058F            0865        BSF     HENTRY,4        ;SET TO 1
045A 0537            0866        BSF     ENTFLG,HR10     ;SET FLAG
045B 0A4B            0867        GOTO    SERV_COM_RET    ;GO GET NEXT
                     0868 LD_HENTRY_0
045C 048F            0869        BCF     HENTRY,4        ;SET TO 0
045D 0537            0870        BSF     ENTFLG,HR10
045E 0A4B            0871        GOTO    SERV_COM_RET    ;
                     0872 ENT_HRS
045F 0C0F            0873        MOVLW   HENTRY          ;USE INDIRECT ADDR.
0460 0024            0874        MOVWF   FSR             ;       /
0461 068F            0875        BTFSC   HENTRY,4        ;SEE IF 0
0462 0A6D            0876        GOTO    ALLOW0_2        ;YES THEN 0,1&2
0463 0C0A            0877        MOVLW   D’10'           ;SEE IF 0 - 9
0464 0093            0878        SUBWF   NEW_KEY,W       ;       /
0465 0603            0879        BTFSC   STATUS,C        ;IF C THEN SKIP
0466 0A1C            0880        GOTO    IGNORE_KEY      ;ELSE IGNORE
                     0881 ENT_LO_COM1
0467 0557            0882        BSF     ENTFLG,HR       ;SET FLAG
                     0883 ENT_LO_COM
0468 0200            0884        MOVF    F0,W            ;LD HRS
0469 0EF0            0885        ANDLW   B’11110000'     ;MASK LO NIBL
046A 0113            0886        IORWF   NEW_KEY,W       ;OR NEW KEY
046B 0020            0887        MOVWF   F0              ;SAVE BACK
046C 0A4B            0888        GOTO    SERV_COM_RET    ;GET NEXT
                     0889 ALLOW0_2
046D 0C03            0890        MOVLW   D’3'            ;SEE IF 0 - 2
046E 0093            0891        SUBWF   NEW_KEY,W       ;       /
046F 0603            0892        BTFSC   STATUS,C        ;<3 THEN SKIP
0470 0A1C            0893        GOTO    IGNORE_KEY
0471 0A67            0894        GOTO    ENT_LO_COM1     ;       /
                     0895 ;
                     0896 ENT_MIN_10
0472 0C0E            0897        MOVLW   MENTRY          ;DO INDIRECT ADDR.
```

```
0473 0024                 MOVWF   FSR             ;       /
0474 0C06                 MOVLW   D’6'            ;ALLOW 0 - 5
0475 0093                 SUBWF   NEW_KEY,W       ;       /
0476 0603                 BTFSC   STATUS,C        ;IF C THEN SKIP
0477 0A1C                 GOTO    IGNORE_KEY      ;ELSE IGNORE
0478 0380                 SWAPF   F0,W            ;SWAP AND GET
0479 0EF0                 ANDLW   B’11110000'     ;MASK LO NIBL
047A 0113                 IORWF   NEW_KEY,W       ;OR NEW KEY
047B 0020                 MOVWF   F0              ;SAVE BACK
047C 03A0                 SWAPF   F0              ;SWAP BACK
047D 0577                 BSF     ENTFLG,MIN10
047E 0A4B                 GOTO    SERV_COM_RET    ;GET NEXT
                  ;
                  ENT_MIN
047F 0C0E                 MOVLW   MENTRY          ;DO INDIRECT
0480 0024                 MOVWF   FSR             ;       /
0481 0C0A                 MOVLW   D’10'           ;ALLOW 0 - 9
0482 0093                 SUBWF   NEW_KEY,W       ;SEE IF >
0483 0603                 BTFSC   STATUS,C        ;NO THEN SKIP
0484 0A1C                 GOTO    IGNORE_KEY      ;ELSE IGNORE
0485 0597                 BSF     ENTFLG,MIN      ;SET FLAG
0486 0A68                 GOTO    ENT_LO_COM      ;       /
                  ;
                  ENT_AM_PM
0487 0C0D                 MOVLW   AM_PM_KEY       ;AM/PM KEY?
0488 0193                 XORWF   NEW_KEY,W       ;    /
0489 0743                 BTFSS   STATUS,Z        ;YES THEN SKIP
048A 0A1C                 GOTO    IGNORE_KEY
048B 07EF                 BTFSS   HENTRY,AM_PM    ;TEST BIT
048C 0A8F                 GOTO    SETAMPM         ;ELSE SET
048D 04EF                 BCF     HENTRY,AM_PM    ;CLEAR FLAG
048E 0A4B                 GOTO    SERV_COM_RET    ;GOTO END
                  SETAMPM
048F 05EF                 BSF     HENTRY,AM_PM    ;SET FLAG
0490 0A4B                 GOTO    SERV_COM_RET
                  ;
                  ;
                  ;
                  SERV_SET_RTM
0491 020A                 MOVF    MTMR,W          ;TRANSFER TIME
0492 002E                 MOVWF   MENTRY          ;TO DATA ENTRY
0493 020B                 MOVF    HTMR,W          ;       /
0494 002F                 MOVWF   HENTRY          ;     /
                  SERV_COM
0495 0210                 MOVF    FLAG,W          ;SAVE IN W
0496 0E01                 ANDLW   B’00000001'     ;ATM OR RTM MODE?
0497 0037                 MOVWF   ENTFLG          ;SAVE IN ENTFLG
0498 0CF2                 MOVLW   B’11110010'     ;FORCE 1S
0499 0130                 IORWF   FLAG            ;    /
049A 0410                 BCF     FLAG,0          ;   /
049B 0800                 RETLW   0
                  ;
                  SERV_SET_ATM
049C 020C                 MOVF    MALARM,W        ;TRANSFER ALARM
049D 002E                 MOVWF   MENTRY          ;TO DATA ENTRY
049E 020D                 MOVF    HALARM,W        ;       /
049F 002F                 MOVWF   HENTRY          ;     /
04A0 0518                 BSF     ALFLAG,ALONOF   ;SET FLAG
04A1 0A95                 GOTO    SERV_COM        ;GOTO COMMON
                  ;
                  SERV_ALARM_ATM
04A2 0718                 BTFSS   ALFLAG,ALONOF   ;TEST ON/OFF
04A3 0AA6                 GOTO    SET_ALONOF      ;SET ON/OF FLG
04A4 0418                 BCF     ALFLAG,ALONOF   ;CLEAR FLAG
04A5 0AA7                 GOTO    SERV_ATM_COM    ;RET THRO COM
                  SET_ALONOF
04A6 0518                 BSF     ALFLAG,ALONOF   ;SET FLAG
                  SERV_ATM_COM
04A7 05B0                 BSF     FLAG,KEY_BEEP   ;BEEP
```

```
04A8 0CF0              MOVLW   B’11110000'     ;CLEAR SEC COUNT
04A9 0176              ANDWF   MIN_SEC         ;       /
04AA 0800              RETLW   0               ;RETURN
               ;
               SERV_ALARM_RTM
04AB 05B0              BSF     FLAG,KEY_BEEP   ;SET BEEP FLAG
04AC 0510              BSF     FLAG,0          ;SET TO ALARM TIME
04AD 0430              BCF     FLAG,1          ;       /
04AE 0C05              MOVLW   D’05'           ;SAVE 5 IN MIN_SEC
04AF 0036              MOVWF   MIN_SEC         ;       /
04B0 0800              RETLW   0
               ;
               SERV_SNOOZE
04B1 0CA0              MOVLW   0A0             ;SNOOZE FOR 10 MINS
04B2 0036              MOVWF   MIN_SEC         ;       /
04B3 0558              BSF     ALFLAG,SILNC    ;SET FLAG
               CLR_AL_COM
04B4 05B0              BSF     FLAG,KEY_BEEP   ;SET BEEP FLAG
04B5 007A              CLRF    AATMR           ;RESET AA TIMER
04B6 0079              CLRF    AAFLAG          ;CLEAR AA FLAGS
04B7 0478              BCF     ALFLAG,INAA     ;RESET INAA FLAG
04B8 0505              BSF     PORT_A,BEP      ;TURN OFF BEEPER
04B9 0800              RETLW   0               ;RET
               ;
               CHK_AL_KEYS
04BA 0718              BTFSS   ALFLAG,ALONOF   ;ALARM ON?
04BB 0801              RETLW   1               ;NO THEN RET
04BC 0738              BTFSS   ALFLAG,INAL     ;IN ALARM?
04BD 0801              RETLW   1               ;NO THEN SKIP
04BE 0C0E              MOVLW   CLR_ALARM_KEY   ;CHECK IF CLR ALARM
04BF 0193              XORWF   NEW_KEY,W       ;       /
04C0 0643              BTFSC   STATUS,Z        ;NO THEN SKIP
04C1 0AC7              GOTO    CLR_ALARM       ;ELSE CLEAR ALARM
04C2 0C0C              MOVLW   SNOOZE_KEY      ;SEE IF SNOOZE HIT
04C3 0193              XORWF   NEW_KEY,W       ;       /
04C4 0743              BTFSS   STATUS,Z        ;YES THEN SKIP
04C5 0801              RETLW   1
04C6 0AB1              GOTO    SERV_SNOOZE
               ;
               CLR_ALARM
04C7 0438              BCF     ALFLAG,INAL     ;CLEAR ALARM
04C8 0458              BCF     ALFLAG,SILNC    ;CLEAR SILENCE
04C9 0C0F              MOVLW   B’00001111'     ;CLEAR MINS
04CA 0176              ANDWF   MIN_SEC         ;       /
04CB 0AB4              GOTO    CLR_AL_COM
               ;
                       ORG     600
               ;If the AA alarm is set, then this routine takes care of
               ;the timing in sounding the alarm.
               ;
               SOUND_AA
0600 0738              BTFSS   ALFLAG,INAL     ;SKIP IF IN ALRM
0601 0800              RETLW   0               ;ELSE RETURN
0602 0658              BTFSC   ALFLAG,SILNC    ;SKIP IF NOT IN SIL
0603 0800              RETLW   0               ;ELSE RET
0604 06B7              BTFSC   ENTFLG,INKEYBEP ;SKIP IF NOT IN KEY BEP
0605 0A55              GOTO    CHK_COLSN       ;CHK COLLISION
               SND_AA_0
0606 0778              BTFSS   ALFLAG,INAA     ;SKIP IF IN AA
               SND_AA_1
0607 0919              CALL    INIT_AA         ;INIT ALL
0608 0719              BTFSS   AAFLAG,0        ;SKIP IF DONE
0609 0A21              GOTO    DO_CYCL0        ;DO FIRST CYCL
060A 0739              BTFSS   AAFLAG,1        ;SKIP IF DONE
060B 0A29              GOTO    DO_CYCL1        ;ELSE 2ND CYCLE
060C 0759              BTFSS   AAFLAG,2        ;SKIP IF DONE
060D 0A31              GOTO    DO_CYCL2        ;ELSE DO 3RD CYCLE
060E 0779              BTFSS   AAFLAG,3        ;SKIP IF DONE
060F 0A39              GOTO    DO_CYCL3        ;DO CYCLE 4
```

```
0610 0799                         BTFSS   AAFLAG,4        ;SKIP IF DONE
0611 0A3E                         GOTO    DO_CYCL4        ;DO CYCLE 5
0612 07B9                         BTFSS   AAFLAG,5        ;SKIP IF DONE
0613 0A43                         GOTO    DO_CYCL5        ;DO CYCLE 6
0614 07D9                         BTFSS   AAFLAG,6        ;SKIP IF DONE
0615 0A48                         GOTO    DO_CYCL6        ;DO CYCLE 6
0616 07F9                         BTFSS   AAFLAG,7        ;SKIP IF DONE
0617 0A50                         GOTO    DO_CYCL7        ;DO CYCLE 7
0618 0A07                         GOTO    SND_AA_1        ;GO BACK
                          ;
                          INIT_AA
0619 0079                         CLRF    AAFLAG          ;CLEAR ALL FLAGS
061A 0578                         BSF     ALFLAG,INAA     ;SET IN AA FLAG
061B 0A2D                         GOTO    PUT_ON_100      ;ON 100 MSECS
                          ;
                          DEC_AA_TMR
061C 00FA                         DECF    AATMR           ;REDUCE TIMER
061D 021A                         MOVF    AATMR,W         ;GET IN W
061E 0743                         BTFSS   STATUS,Z        ;CHECK IF Z
061F 0801                         RETLW   1               ;NO THEN NZ
0620 0800                         RETLW   0               ;ELSE 0
                          ;
                          DO_CYCL0
0621 091C                         CALL    DEC_AA_TMR      ;REDUCE TIMER
0622 0743                         BTFSS   STATUS,Z        ;IF NZ THEN RET
0623 0800                         RETLW   0
0624 0519                         BSF     AAFLAG,0        ;SET DONE FLAG
                          PUT_OFF_100
0625 0505                         BSF     PORT_A,BEP      ;TURN OFF BEEPER
0626 0C14                         MOVLW   D’20'           ;FOR 100 MSECS
0627 003A                         MOVWF   AATMR           ;       /
0628 0800                         RETLW   0
                          ;
                          DO_CYCL1
0629 091C                         CALL    DEC_AA_TMR      ;REDUCE TIMER
062A 0743                         BTFSS   STATUS,Z        ;IF NZ THEN RET
062B 0800                         RETLW   0
062C 0539                         BSF     AAFLAG,1        ;SET DONE FLAG
                          PUT_ON_100
062D 0405                         BCF     PORT_A,BEP      ;TURN ON BEEPER
062E 0C14                         MOVLW   D’20'           ;FOR 100 MSECS
062F 003A                         MOVWF   AATMR           ;       /
0630 0800                         RETLW   0
                          ;
                          DO_CYCL2
0631 091C                         CALL    DEC_AA_TMR      ;REDUCE TIMER
0632 0743                         BTFSS   STATUS,Z        ;IF NZ THEN RET
0633 0800                         RETLW   0               ;       /
0634 0559                         BSF     AAFLAG,2        ;SET DONE FLAG
0635 0505                         BSF     PORT_A,BEP      ;TURN OFF BEEPER
0636 0C64                         MOVLW   D’100'          ;FOR 500 MSECS
0637 003A                         MOVWF   AATMR           ;       /
0638 0800                         RETLW   0
                          ;
                          DO_CYCL3
0639 091C                         CALL    DEC_AA_TMR      ;REDUCE TIMER
063A 0743                         BTFSS   STATUS,Z        ;IF NZ THEN RET
063B 0800                         RETLW   0               ;       /
063C 0579                         BSF     AAFLAG,3        ;SET DONE FLAG
063D 0A2D                         GOTO    PUT_ON_100      ;DO NEXT CYCLE
                          ;
                          DO_CYCL4
063E 091C                         CALL    DEC_AA_TMR      ;REDUCE TIMER
063F 0743                         BTFSS   STATUS,Z        ;IF NZ THEN RET
0640 0800                         RETLW   0               ;       /
0641 0599                         BSF     AAFLAG,4        ;SET DONE FLAG
0642 0A25                         GOTO    PUT_OFF_100     ;DO NEXT CYCLE
                          ;
                          DO_CYCL5
```

```
0643 091C                 CALL    DEC_AA_TMR      ;REDUCE TIMER
0644 0743                 BTFSS   STATUS,Z        ;IF NZ THEN RET
0645 0800                 RETLW   0               ;       /
0646 05B9                 BSF     AAFLAG,5        ;SET DONE FLAG
0647 0A2D                 GOTO    PUT_ON_100      ;DO NEXT CYCLE
                  ;
                  DO_CYCL6
0648 091C                 CALL    DEC_AA_TMR      ;REDUCE TIMER
0649 0743                 BTFSS   STATUS,Z        ;IF NZ THEN RET
064A 0800                 RETLW   0               ;       /
064B 05D9                 BSF     AAFLAG,6        ;SET DONE FLAG
064C 0505                 BSF     PORT_A,BEP      ;TURN OFF BEEPER
064D 0CC8                 MOVLW   D'200'          ;FOR 1000 MSECS
064E 003A                 MOVWF   AATMR           ;       /
064F 0800                 RETLW   0
                  ;
                  DO_CYCL7
0650 091C                 CALL    DEC_AA_TMR      ;REDUCE TIMER
0651 0743                 BTFSS   STATUS,Z        ;IF NZ THEN RET
0652 0800                 RETLW   0               ;       /
0653 05F9                 BSF     AAFLAG,7        ;SET DONE FLAG
0654 0A2D                 GOTO    PUT_ON_100      ;DO NEXT CYCLE
                  ;
                  CHK_COLSN
0655 0605                 BTFSC   PORT_A,BEP      ;IF ON THEN SKIP
0656 0A06                 GOTO    SND_AA_0        ;ELSE RET
0657 021A                 MOVF    AATMR,W         ;GET TIMER
0658 0643                 BTFSC   STATUS,Z        ;NZ THEN SKIP
0659 0A5C                 GOTO    LD_AAT_1        ;LOAD A 1 IN TMR
065A 00FA                 DECF    AATMR           ;REDUCE TIMER
065B 0800                 RETLW   0               ;RETURN
                  LD_AAT_1
065C 02BA                 INCF    AATMR           ;INC TIMER
065D 0800                 RETLW   0               ;RET
                  ;
                          ORG     PIC57
                  SYS_RESET
07FF 0A00                 GOTO    START
                  ;
                          END
```

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

0000 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
0040 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX

0080 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
00C0 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX

0100 : XXXXXXXXXXXXXXXX XXXXXXXXX——— ————————— —————————
0140 : ————————— ————————— ————————— —————————

0200 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
0240 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX

0280 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXX————— —————————
02C0 : ————————— ————————— ————————— —————————

0400 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
0440 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX

0480 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
04C0 : XXXXXXXXXXXX—— ————————— ————————— —————————
```

```
0600 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
0640 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXX— ———————— ————————

0780 : ———————— ———————— ———————— ————————
07C0 : ———————— ———————— ———————— ———————X

All other memory blocks unused.


Errors   :    0
Warnings :    0
```

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/microhip

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

AMERICAS (continued)

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

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.