

## Multiplexing LED Drive and a 4x4 Keypad Sampling

Author: Stan D'Souza  
Microchip Technology Inc.

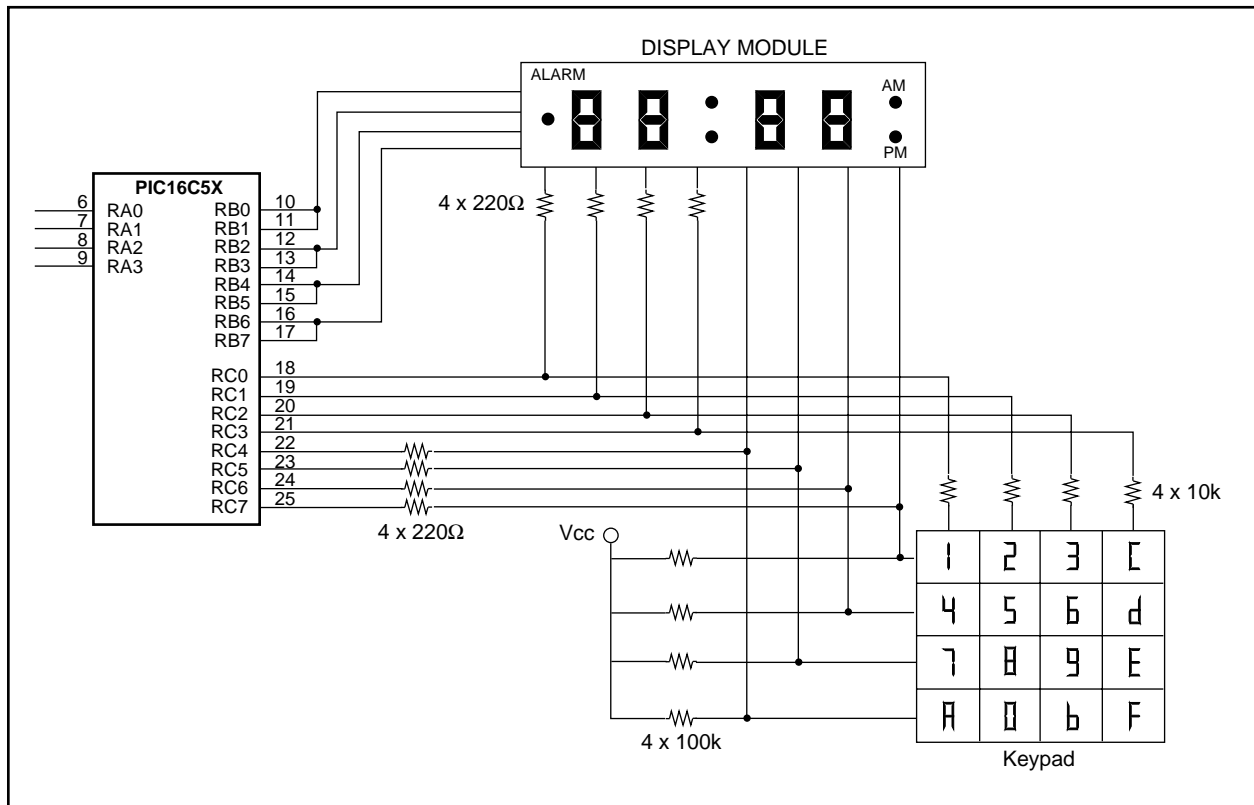
### INTRODUCTION

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 (Figure 1). Direct

drive of the LEDs is possible, because of the high sink and source capabilities of PIC16C5X microcontroller, thus eliminating the use of an external drive transistor, and results in a reduction in both cost and complexity of the overall circuit.

Typically applications having LEDs and keypads also keep track of real-time, in order to synchronize certain related events. An Industrial Clock/Timer example has been used in this application note as a demonstration of these functions. 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 PORTC of the PIC16C5X (Figure 2). 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 4).

The keypad sampling is as follows:

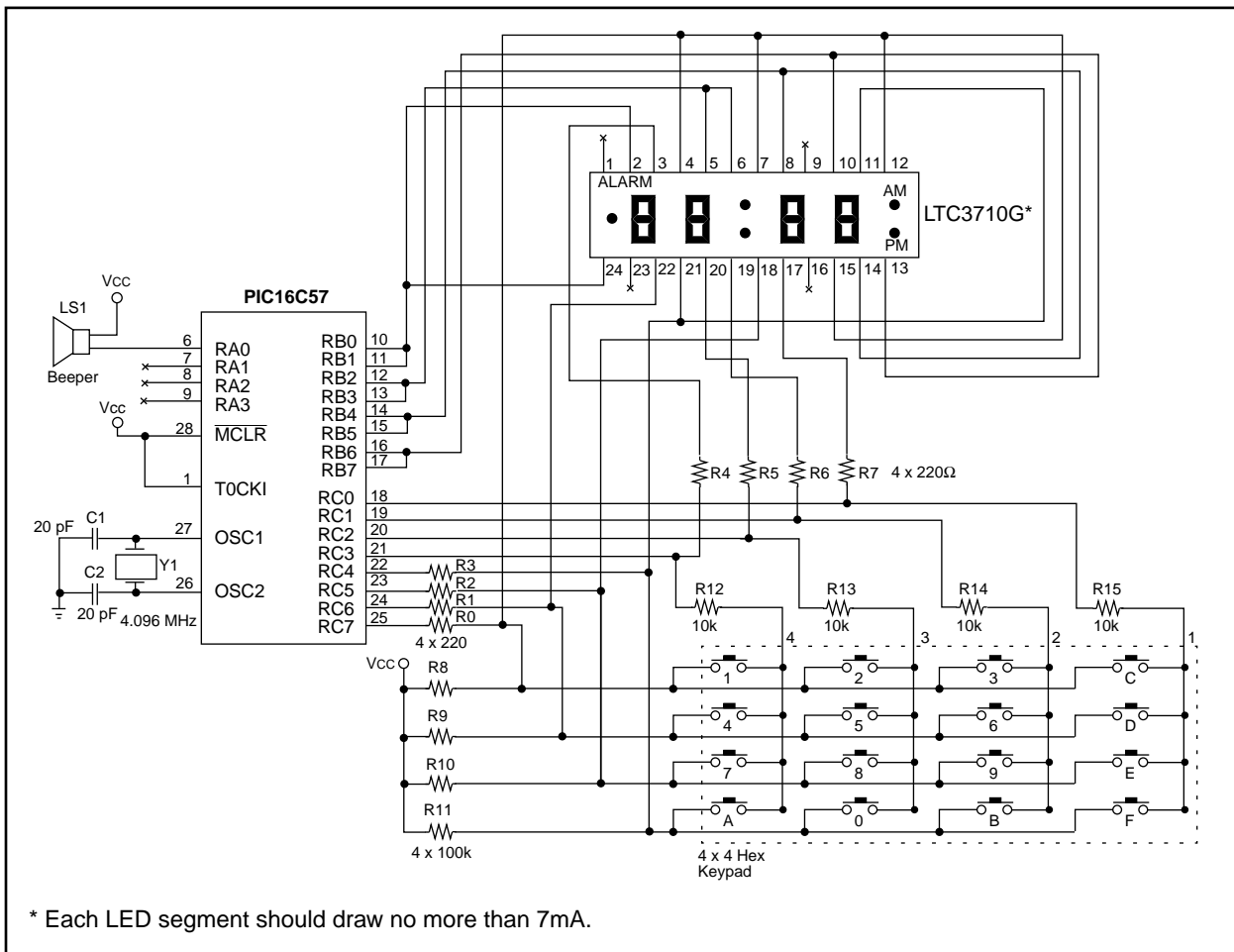
- The columns are connected to output pins, and the rows are connected to input pins.
- Each column is sequentially driven to a low voltage while at the same instance the four rows are sampled. Since the rows are all held high with pull-up resistors, all four inputs will normally be high. If a key is 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.

- Before a new column is brought low, care should be taken to discharge the input pins (see code section for details).
- A 50 ms key debounce technique has been implemented in the software, in order to eliminate multiple key strokes.

**Note 1:** Resistors R8-R11 and R12-R14 have been selected such that their ratio is 1:10. This will insure a 0.5V 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.

**Note 2:** In circuits where there is substantial interference between the key matrix and the LED drive circuit, the alternative circuit of Figure 3 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 2: PIC16C5X INDUSTRIAL CLOCK/TIMER SCHEMATIC



## 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 OPTION register) to give a clock of 32 kHz which is used to increment the TMR0 in the PIC16C5X. If the TMR0 is initialized to 96, it would overflow to zero in 5 ms.

$$(256-96) \times (1/32000) = 5.000 \text{ ms}$$

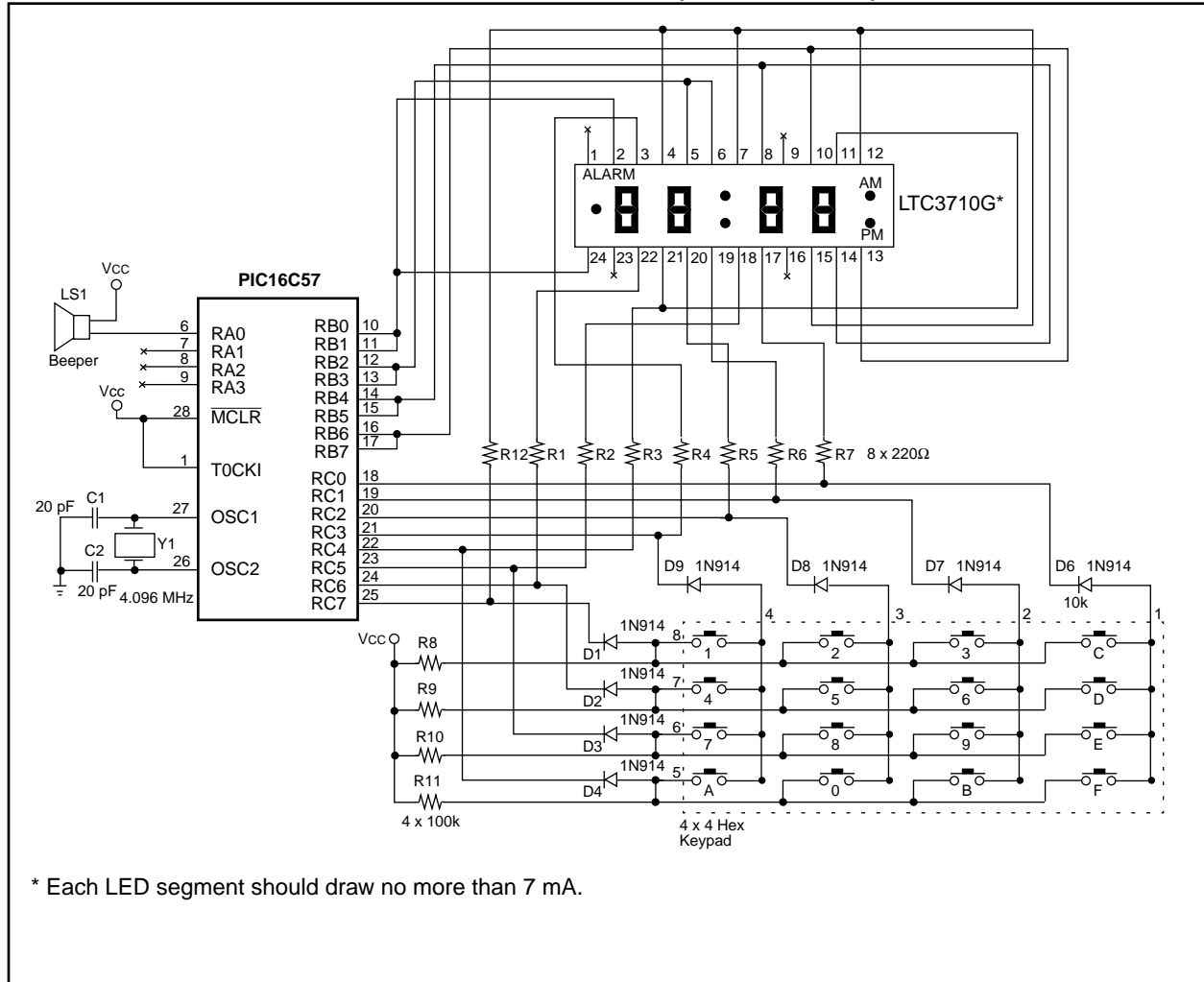
This 5 ms is used to count the seconds, minutes and hours in the clock/timer. It is also used as a time-base to update the display digits and sample the keyboard. The clock speed being 4.096 MHz, each instruction will execute in 1  $\mu$ s. Therefore in 5 ms, approximately 5000

instructions can be executed. This gives sufficient time to execute a large section of code and not miss the overflow in TMR0.

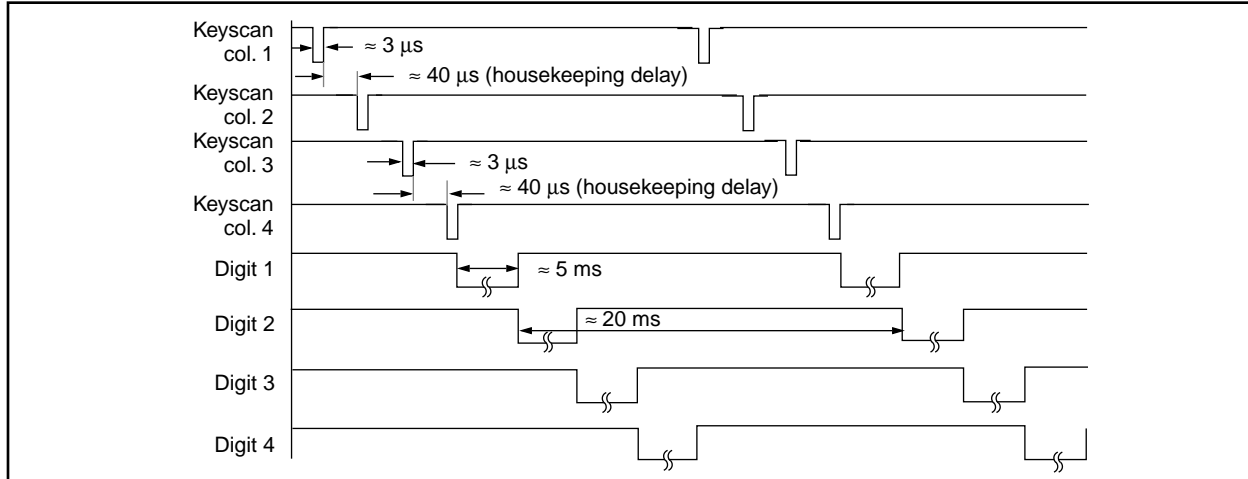
### Using a 3.579545 MHz colorburst crystal oscillator as a time-base

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  $\mu$ s. If this is prescaled by 32, TMR0 will be incremented every 35.758  $\mu$ s. Initializing the TMR0 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 3: PIC16C5X ALARM CLOCK SCHEMATIC (USING DIODES)**



**FIGURE 4: KEY SCAN AND LED DIGIT SELECT TIMING**



## FEATURES

The Flowchart (Figure 5) shows the sequence of events in the clock/timer software. The clock has the following features:

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

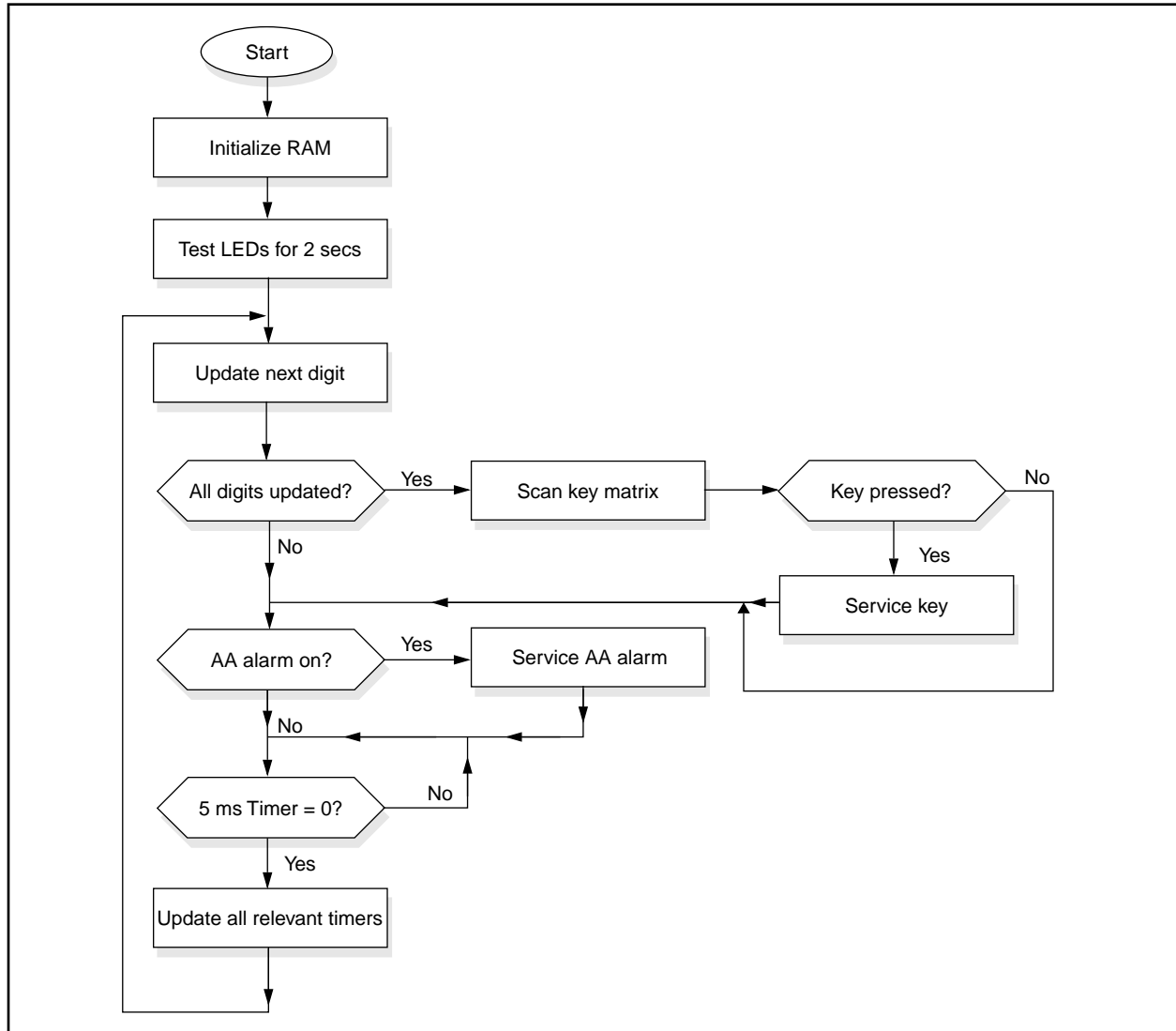
## SETTING CLOCK/TIMER FUNCTIONS

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 alarm time)

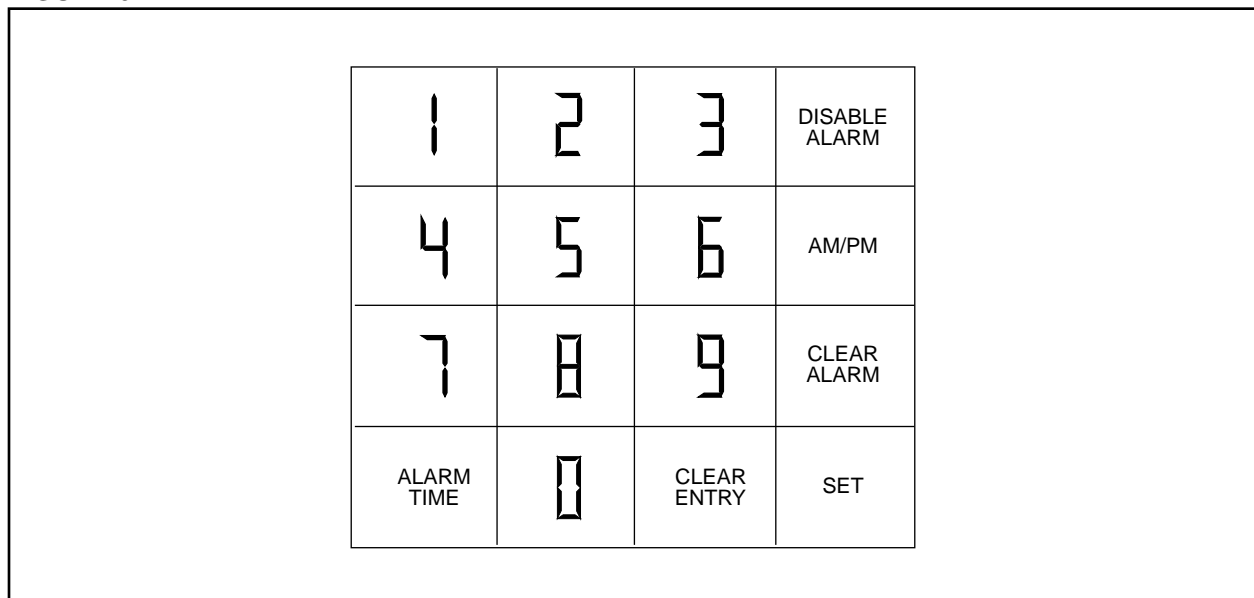
**Note 1:** Valid key strokes will be acknowledged with a beep.

**Note 2:** Hours and minutes used above correspond to digits 0 - 9 on the keyboard.

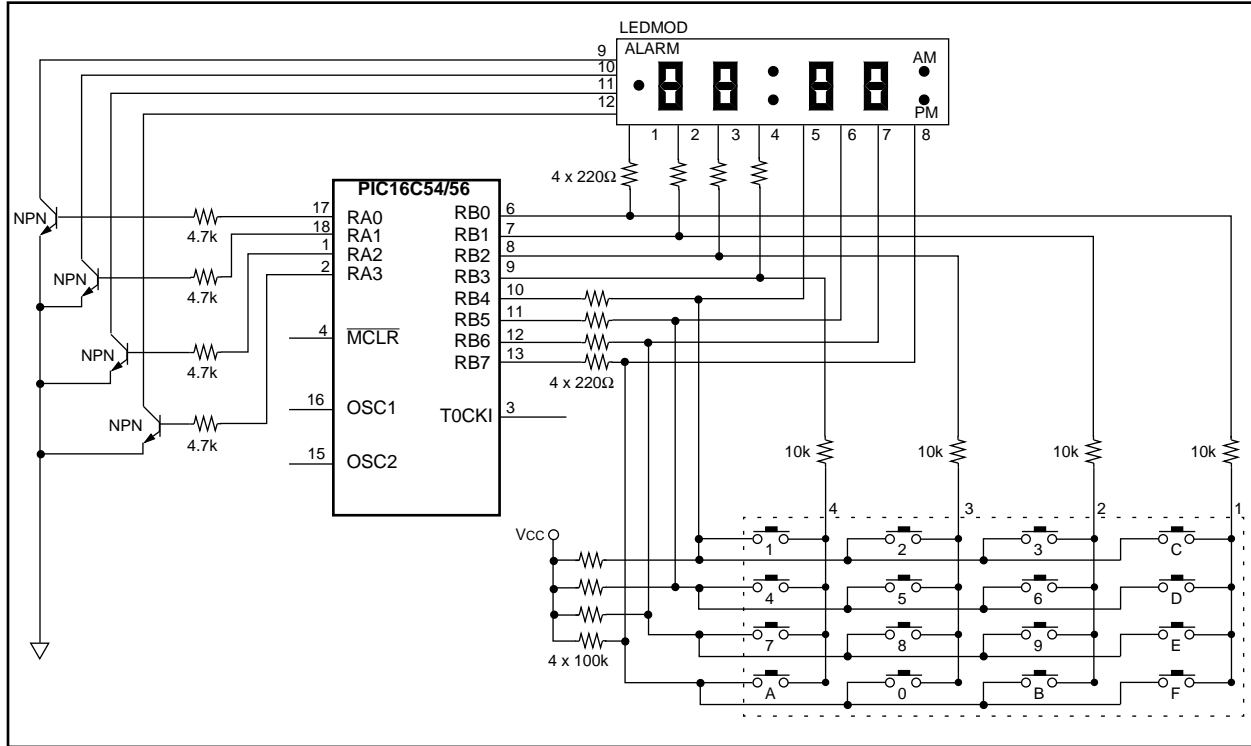
**FIGURE 5: TIMER/CLOCK FLOWCHART**



**FIGURE 6: KEYPAD**



**FIGURE 7: INTERFACE TO PIC16C54/56**



## CODE SIZE

Key scan → 97 bytes  
 Display update → 113 bytes

## SUMMARY

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

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

Figure 7 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, an external npn transistor will have to be utilized to sink the current from each digit.

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

## APPENDIX A: CODE LISTING

MPASM 01.40 Released

CLK.ASM 1-16-1997 13:07:37

PAGE 1

```

LOC  OBJECT CODE      LINE SOURCE TEXT
VALUE
                                00001          TITLE          "Alarm Clock"
                                00002      LIST  P = 16C57
                                00003      ERRORLEVEL  -306
                                00004 ;
                                00005 ;Define Equates:
                                00006 ;
000007FF      00007 PIC57  EQU      7FFH
                                00008 ;*****
                                00009 ;
                                00010 ;External Ossc. used = 4.096Mhz. Prescaler of 32 used, which gives a
                                00011 ;31.25 microSec increment of the TMR0.If TMR0 is intially loaded with
                                00012 ;96, it would overflow to 0 in 5.000 milliSecs. Giving a 0.00% error.
                                00013 ;
00000060      00014 MSEC5   EQU      D'96'
                                00015 ;
                                00016 ;
                                00017 ;      Program:          CLK.ASM
                                00018 ;      Revision Date:
                                00019 ;                  1-13-97      Compatibility with MPASMWIN 1.40
                                00020 ;
                                00021 ;*****
                                00022 ;
00000000      00023 C          EQU      0
00000000      00024 BEP     EQU      0
00000000      00025 RTATS   EQU      0
00000001      00026 DC      EQU      1
00000001      00027 HR10    EQU      1
00000002      00028 Z       EQU      2
00000002      00029 HR      EQU      2
00000003      00030 MIN10   EQU      3
00000004      00031 MIN     EQU      4
00000004      00032 FLASH   EQU      4
00000005      00033 PA0     EQU      5
00000005      00034 KEY_BEEP EQU      5
00000005      00035 AMPM    EQU      5
00000006      00036 PA1     EQU      6
00000000      00037 F0      EQU      0
00000006      00038 KEY_HIT  EQU      6
00000006      00039 ALED    EQU      6
00000007      00040 AM_PM   EQU      7
00000003      00041 COLON   EQU      3
00000002      00042 ALRMLED  EQU      2
00000007      00043 SERVICED EQU      7
00000000      00044 ALONOF   EQU      0
00000001      00045 INAL    EQU      1
00000002      00046 SILNC   EQU      2
00000003      00047 INAA    EQU      3
00000005      00048 INKEYBEP EQU      5
                                00049 ;
                                00050 ;DEFINE RAM LOCATIONS:
00000001      00051 TMR0    EQU      1
00000002      00052 PC      EQU      2
00000003      00053 STATUS  EQU      3
00000004      00054 FSR     EQU      4

```

```

00000005      00055 PORT_A   EQU     5
00000006      00056 PORT_B   EQU     6
00000007      00057 PORT_C   EQU     7
00058 ;
00000001      00059 F         EQU     1
00060 ;
00061 ;DEFINE REAL TIME MODE REGS (RTM)
00000008      00062 MSTMR    EQU     8      ;MILLI SEC. TIMER
00000009      00063 STMR     EQU     9      ;SEC. TIMER
00064 ;
00065 ;*****
00066 ;DO NOT CHANGE RELATIVE POSITION OF NEXT 6 BYTES
0000000A      00067 MTMR     EQU     0A     ;MIN. TIMER
0000000B      00068 HTMR     EQU     0B     ;HOUR TIMER
00069 ;
00070 ;DEFINE ALARM TIME MODE REGS (ATM)
0000000C      00071 MALARM   EQU     0C     ;MIN. ALARM
0000000D      00072 HALARM   EQU     0D     ;HOUR ALARM
00073 ;
00074 ;DEFINE DATA ENTRY MODE REGS (DEM)
0000000E      00075 MENTRY   EQU     0E     ;MIN. ENTRY
0000000F      00076 HENTRY   EQU     0F     ;HOUR ENTRY
00077 ;*****
00078 PAGE
00079 ;
00080 ;DEFINE FLAG REG AND FUNCTION:
00000010      00081 FLAG     EQU     10
00082 ;
00083 ;          BIT # 7|6|5|4|3|2|1|0|
00084 ;-----|---|---|---|---|
00085 ;          X|X|X|X|X|0|0|    --> REAL TIME MODE (RTM)
00086 ;          X|X|X|X|X|0|1|    --> ALARM TIME MODE(ATM)
00087 ;          X|X|X|X|X|1|0|    --> DATA ENTRY MODE(DEM)
00088 ;          X|X|X|X|X|1|1|    --> TEST MODE (TM)
00089 ;          X|X|X|X|X|Y|X|X|    --> ALRMLD ON/OFF
00090 ;          X|X|X|X|Y|X|X|X|    --> COLON LED ON/OFF
00091 ;          X|X|X|Y|X|X|X|X|    --> FLASH DISPLAY
00092 ;          X|X|Y|X|X|X|X|X|    --> KEY_BEEP
00093 ;          X|Y|X|X|X|X|X|X|    --> KEY_HIT (0/1)
00094 ;          Y|X|X|X|X|X|X|X|    --> SERVICED
00095 ; X = DEFINED ELSEWHERE IN TABLE
00096 ; Y = DEFINED AS SHOWN (0/1)
00097 ;
00000011      00098 TEMP     EQU     11
00000012      00099 DIGIT    EQU     12
00000013      00100 NEW_KEY   EQU     13
00000014      00101 KEY_NIBL  EQU     14
00000015      00102 DEBOUNCE  EQU     15
00000016      00103 MIN_SEC   EQU     16     ;MIN/SECONDS TIMER
00000017      00104 ENTFLG    EQU     17
00105 ;
00106 ;flag dedicated to the key entry mode
00107 ;          BIT # 7|6|5|4|3|2|1|0|
00108 ;-----|---|---|---|---|
00109 ;          X|X|X|X|X|X|Y|    --> REAL/ALARM TIME STATUS
00110 ;          X|X|X|X|X|Y|X|    --> HR10 DONE
00111 ;          X|X|X|X|X|Y|X|X|    --> HR DONE
00112 ;          X|X|X|X|Y|X|X|X|    --> MIN10 DONE
00113 ;          X|X|X|Y|X|X|X|X|    --> MIN DONE
00114 ;          X|X|Y|X|X|X|X|X|    --> INKEYBEP
00115 ;          X|Y|X|Y|X|X|X|X|    --> NOT USED
00116 ;          Y|X|X|X|X|X|X|X|    --> NOT USED
00117 ;
00118 ;
00000018      00119 ALFLAG    EQU     18
00120 ;

```



```

00121 ;flag dedicated to the alarm
00122 ;      BIT # 7|6|5|4|3|2|1|0|
00123 ;-----|---|---|---|---|
00124 ;      X|X|X|X|X|X|Y|  --> ALONOF
00125 ;      X|X|X|X|X|X|Y|X|  --> INAL
00126 ;      X|X|X|X|X|Y|X|X|  --> SILNC
00127 ;      X|X|X|X|Y|X|X|X|  --> INAA
00128 ;      X|X|X|Y|X|X|X|X|  --> NOT USED
00129 ;      X|X|Y|X|X|X|X|X|  --> NOT USED
00130 ;      X|Y|X|Y|X|X|X|X|  --> NOT USED
00131 ;      Y|X|X|X|X|X|X|X|  --> NOT USED
00132 ;
00000019 00133 AAFLAG    EQU    19
00134 ;
0000001A 00135 ;flag dedicated to the AA alarm
00136 AATMR      EQU    1A
00137 ;
00138 ;Port pin definitions:
00139 ;
00140 ;PORT_A:
00141 ;      BIT 0    --> BEEPER (ACTIVE LOW) OUTPUT
00142 ;      BIT 1-3  --> unused I/O
00143 ;
00144 ;PORT_B: ALL OUTPUTS
00145 ;      BIT 0&4  --> MSB DIGIT COMMON CATHODE & ALARM
00146 ;      BIT 1&5  --> 2ND DIGIT COMMOM CATHODE & COLON
00147 ;      BIT 2&6  --> 3RD DIGIT COMMON CATHODE & PM
00148 ;      BIT 3&7  --> LSB DIGIT COMMON CATHODE & AM
00149 ;
00150 ;PORT_C:
00151 ;IN DISPLAY MODE ALL SEG/ANNN SET AS OUTPUTS
00152 ;IN KEY SCAN MODE COLS ARE OUTPUTS ROWS ARE INPUTS
00153 ;      BIT 0    --> SEGMENT A & COL 4
00154 ;      BIT 1    --> SEGMENT B & COL 3
00155 ;      BIT 2    --> SEGMENT C & COL 2
00156 ;      BIT 3    --> SEGMENT D & COL 1
00157 ;      BIT 4    --> SEGMENT E & ROW 4
00158 ;      BIT 5    --> SEGMENT F & ROW 3
00159 ;      BIT 6    --> SEGMENT G & ROW 2
00160 ;      BIT 7    --> CA OF ALL ANNUNCIATORS & ROW 1
00161 ;
00162 ;
00163      PAGE
00164 ;
0000      00165      ORG      0
0000      00166 START
0000 0AFC      00167      GOTO    INIT_CLK      ;INITIALIZE CLOCK
00168 ;THIS ROUTINE RUNS A TEST ON THE LEDS.
00169 ;ALL THE RELEVANT LEDS ARE LIT UP FOR 2 SECS.
00170 ;
0001      00171 TEST_HARDWARE
0001 0C02      00172      MOVLW   d'02'          ;DISPLAY FOR 2 SECS
0002 0036      00173      MOVWF   MIN_SEC      ;      /
00174 ;
00175 ;
0003      00176 TEST_LOOP
0003 0216      00177      MOVF    MIN_SEC,W      ;GET MIN/SEC
0004 0643      00178      BTFSC   STATUS,Z      ;NOT 0 THEN SKIP
0005 0A0B      00179      GOTO    NORM_TIME     ;ELSE NORMAL TIME
0006 0925      00180      CALL    UPDATE_DISPLAY ;UPDATE DISPLAY
0007 05A3      00181      BSF     STATUS,PA0     ;GOTO PAGE 1
0008 0900      00182      CALL    UPDATE_TIMERS ;WAIT AND UPDATE
0009 04A3      00183      BCF     STATUS,PA0     ;RESET PAGE MARKER
000A 0A03      00184      GOTO    TEST_LOOP     ;LOOP BACK
000B      00185 NORM_TIME
000B 0410      00186      BCF     FLAG,0        ;PUT IN REAL TIME

```

```

000C 0430      00187      BCF      FLAG,1
000D           00188      TIME_LOOP
000D 0925      00189      CALL     UPDATE_DISPLAY
000E 05C3      00190      BSF     STATUS,PA1      ;GOTO PAGE 2
000F 0900      00191      CALL     SERVICE_KEYS
0010 05A3      00192      BSF     STATUS,PA0      ;GOTO PAGE 3
0011 0900      00193      CALL     SOUND_AA      ;CHECK ALARM
0012 04C3      00194      BCF     STATUS,PA1      ;GOTO PAGE 1
0013 0900      00195      CALL     UPDATE_TIMERS ;WAIT AND UPDATE TIMERS
0014 04A3      00196      BCF     STATUS,PA0      ;RESET PAGE MARKER
0015 04C3      00197      BCF     STATUS,PA1      ;
0016 0210      00198      MOVF    FLAG,W          ;SEE IF IN ATM
0017 0E03      00199      ANDLW   B'00000011'     ;
0018 0F01      00200      XORLW   B'00000001'     ;
0019 0643      00201      BTFSC   STATUS,Z        ;SKIP IF NOT
001A 091C      00202      CALL     RESET_ATM
001B 0A0D      00203      GOTO    TIME_LOOP
00204 ;
001C           00205      RESET_ATM
001C 0216      00206      MOVF    MIN_SEC,W      ;GET MIN/SEC
001D 0E0F      00207      ANDLW   B'00001111'     ;
001E 0743      00208      BTFSS   STATUS,Z        ;Z THEN SKIP
001F 0800      00209      RETLW   0              ;ELSE RETURN
0020 0410      00210      BCF     FLAG,0          ;SET TO RTM
0021 0450      00211      BCF     FLAG,ALRMLD     ;CLEAR LED
0022 0618      00212      BTFSC   ALFLAG,ALONOF  ;TEST STAT
0023 0550      00213      BSF     FLAG,ALRMLD     ;SET LED
0024 0800      00214      RETLW   0              ;RETURN
00215      PAGE
00216 ;
00217 ;
0025           00218      UPDATE_DISPLAY
0025 0C00      00219      MOVLW   B'00000000'     ;CLEAR SEG DRIVE
0026 0027      00220      MOVWF   PORT_C          ;
0027 0C3F      00221      MOVLW   B'00111111'     ;SEE IF LAST DIGIT
0028 0186      00222      XORWF   PORT_B,0        ;
0029 0643      00223      BTFSC   STATUS,Z        ;NO THEN SKIP
002A 0A6F      00224      GOTO    SCAN_KP        ;ELSE SCAN KEYPAD
002B           00225      UP_DSP_1
002B 0246      00226      ;SELECT DIGIT TO BE DISPLAYED
002C 0643      00227      COMF    PORT_B,0        ;GET COMPL. PORT B IN W
002D 0CC0      00228      BTFSC   STATUS,Z        ;NO DIGIT SELECTED?
002E 0031      00229      MOVLW   B'11000000'     ;THEN SELECT DEFAULT
002F 0271      00230      MOVWF   TEMP            ;SAVE IN TEMP
0030 0503      00231      COMF    TEMP, F         ;COMPLEMENT VALUE
0031 0371      00232      BSF     STATUS,C        ;SET CARRY
0032 0703      00233      RLF     TEMP, F         ;SHIFT LEFT
0033 0371      00234      BTFSS   STATUS,C        ;IF C=1 THEN SKIP
0034 0371      00235      RLF     TEMP, F         ;ELSE 3 TIMES...
0035 0211      00236      RLF     TEMP, F         ;THRU CARRY
0036 0026      00237      MOVF    TEMP,0          ;GET IN W
0037           00238      MOVWF   PORT_B          ;OUTPUT TO PORT
0038           00239      ;NOW THAT DIGIT IS SELECTED, SELECT SEG VALUES FOR THAT DIGIT
0039           00240      ;FIRST FIND MODE OF OPERATION.
0037 0C0A      00241      MOVLW   MTMR            ;LOAD FSR WITH MTMR
0038 0024      00242      MOVWF   FSR             ;
0039 0210      00243      MOVF    FLAG,0          ;GET FLAG IN W
003A 0E03      00244      ANDLW   B'00000011'     ;MASK OTHER BITS
003B 0031      00245      MOVWF   TEMP            ;SAVE IN TEMP
003C 0F03      00246      XORLW   B'00000011'     ;IN TEST MODE
003D 0643      00247      BTFSC   STATUS,Z        ;NO THEN SKIP
003E 0A4B      00248      GOTO    DO_TM           ;ELSE TEST MODE
003F 0403      00249      BCF     STATUS,C        ;CLEAR CARRY
0040 0371      00250      RLF     TEMP, F         ;LEFT SHIFT TEMP
0041 0211      00251      MOVF    TEMP,0          ;GET IN W
0042 01E4      00252      ADDWF   FSR, F          ;CHANGE INDIRECT POINTER

```

```

0043 0954      00253      CALL      GET_7_SEG      ;GET 7 SEG DATA IN W
0044 0032      00254      MOVWF    DIGIT          ;SAVE IN DIGIT LOC.
0045 09D1      00255      CALL      MASK_ANNC     ;MASK ANNC TO DIGIT
0046 0690      00256      BTFSC   FLAG,FLASH     ;NO FLASH THEN SKIP
0047 094E      00257      CALL      CHK_HALF_SEC  ;ELSE CHK. IF ON
0048 0212      00258      MOVF     DIGIT,0        ;GET BACK DIGIT
0049 0027      00259      MOVWF    PORT_C        ;OUTPUT TO PORT
004A 0800      00260      RETLW   0               ;RETURN
                                00261 ;
004B           00262 DO_TM
004B 0CFF      00263      MOVLW   B'11111111'    ;LIGHT ALL SEGMENTS
004C 0027      00264      MOVWF    PORT_C        ;
004D 0800      00265      RETLW   0               ;RETURN FROM UPDATE DISPLAY
                                00266 ;
004E           00267 CHK_HALF_SEC
004E 0770      00268      BTFSS   FLAG,COLON     ;IF COLON ON THEN DO
004F 0A51      00269      GOTO    BLANK_DSP      ;ELSE BLANK DISPLAY
0050 0800      00270      RETLW   0
0051           00271 BLANK_DSP
0051 0C00      00272      MOVLW   B'00000000'    ;MAKE PORT C LOW
0052 0032      00273      MOVWF    DIGIT
0053 0800      00274      RETLW   0
                                00275 ;
                                00276 PAGE
                                00277 ;
00278 ;ON ENTRY FSR POINTS TO THE REAL TIME MODE'S MINUTES REGISTER.
00279 ;ON RETURN FSR POINTS TO THE TIMER REGISTER TO BE DISPLAYED.
00280 ;W REG. CONTAINS THE DECODED 7 SEG. INFO OF THE DIGIT
00281 ;TO BE DISPLAYED
00282 ;
0054           00283 GET_7_SEG
0054 0246      00284      COMF    PORT_B,0       ;COMPLEMENT B --> W
0055 0EF0      00285      ANDLW   B'11110000'    ;MASK LO NIBBLE
0056 0643      00286      BTFSC   STATUS,Z       ;NZ THEN SKIP
0057 02A4      00287      INCF    FSR, F         ;INC POINTER
0058 0200      00288      MOVF    F0,0           ;MOVE INDIRECT TO W
0059 0031      00289      MOVWF    TEMP          ;GET INTO TEMP
005A 0246      00290      COMF    PORT_B,0       ;COMPL.B --> W
005B 0EF0      00291      ANDLW   B'11110000'    ;MASK LO NIBBLE
005C 0643      00292      BTFSC   STATUS,Z       ;IF D1/2 THEN
005D 04F1      00293      BCF     TEMP,AM_PM     ;CLEAR AM/PM BIT
005E 0246      00294      COMF    PORT_B,0       ;GET PORT B AGAIN
005F 0ECC      00295      ANDLW   B'11001100'    ;SEE IF D2 OR D4
0060 0643      00296      BTFSC   STATUS,Z       ;YES THEN SKIP
0061 03B1      00297      SWAPF   TEMP, F       ;SWAP TEMP
0062 0C0F      00298      MOVLW   B'00001111'    ;MASK HI NIBBLE
0063 0151      00299      ANDWF   TEMP,0         ;
0064 01E2      00300      ADDWF   PC, F          ;ADD TO PC
0065 083F      00301      RETLW   B'00111111'    ;CODE FOR 0
0066 0806      00302      RETLW   B'00000110'    ;CODE FOR 1
0067 085B      00303      RETLW   B'01011011'    ;CODE FOR 2
0068 084F      00304      RETLW   B'01001111'    ;CODE FOR 3
0069 0866      00305      RETLW   B'01100110'    ;CODE FOR 4
006A 086D      00306      RETLW   B'01101101'    ;CODE FOR 5
006B 087D      00307      RETLW   B'01111101'    ;CODE FOR 6
006C 0807      00308      RETLW   B'00000111'    ;CODE FOR 7
006D 087F      00309      RETLW   B'01111111'    ;CODE FOR 8
006E 0867      00310      RETLW   B'01100111'    ;CODE FOR 9
                                00311 ;
00312 ;This routine scans the 4x4 hex key pad for a key hit.
00313 ;If key is pressed, KEY_HIT flag is set and the value of
00314 ;the hex key is returned in reg NEW_KEY
00315 ;If no key is detected, then a 0xff value is returned in
00316 ;register NEW_KEY and the flag KEY_HIT is reset.
00317 ;
006F           00318 SCAN_KP

```

```

006F 06D0          00319          BTFSC  FLAG,KEY_HIT      ;KEY UNDER SERVICE?
0070 0A2B          00320          GOTO   UP_DSP_1        ;YES SKIP ROUTINE
0071 0CFF          00321          MOVLW  B'11111111'     ;SET DIGIT SINKS ...
0072 0026          00322          MOVWF  PORT_B          ;TO HIGH
0073 0CF7          00323          MOVLW  B'11110111'     ;SET KEY COL LOW
0074 0031          00324          MOVWF  TEMP            ;SAVE IN TEMP
0075              00325  SKP1
0075 0C00          00326          MOVLW  B'00000000'     ;SET PORT C AS OUTPUTS
0076 0007          00327          TRIS   PORT_C          ;
0077 0211          00328          MOVF   TEMP,W          ;
0078 0E0F          00329          ANDLW  B'00001111'     ;DISCHARGE PINS FOR MEMBRANE KEYPADS
0079 0027          00330          MOVWF  PORT_C          ;
007A 0CF0          00331          MOVLW  B'11110000'     ;SET AS I/O
007B 0007          00332          TRIS   PORT_C          ;
007C 0211          00333          MOVF   TEMP,W          ;GET OLD VALUE
007D 0027          00334          MOVWF  PORT_C          ;OUTPUT TO PORT
007E 0207          00335          MOVF   PORT_C,W        ;INPUT PORT VALUE
007F 0EF0          00336          ANDLW  B'11110000'     ;MASK LO BYTE
0080 0FF0          00337          XORLW  B'11110000'     ;SEE IF KEY HIT
0081 0743          00338          BTFSS  STATUS,Z        ;NO KEY THEN SKIP
0082 0A8D          00339          GOTO   DET_KEY        ;LOAD KEY VALUE
0083              00340  SKP3
0083 0503          00341          BSF    STATUS,C         ;SET CARRY
0084 0331          00342          RRF    TEMP, F         ;MAKE NEXT COL. LOW
0085 0603          00343          BTFSC  STATUS,C         ;ALL DONE THEN SKIP
0086 0A75          00344          GOTO   SKP1
0087 0073          00345          CLRFB  NEW_KEY         ;SET NEW_KEY = FF
0088 00F3          00346          DECF   NEW_KEY, F      ;
0089              00347  SKP2
0089 0067          00348          CLRFB  PORT_C          ;SETPORT C AS ...
008A 0C00          00349          MOVLW  B'00000000'     ;OUTPUTS
008B 0007          00350          TRIS   PORT_C          ;
008C 0A2B          00351          GOTO   UP_DSP_1        ;RETURN
008D              00352  DET_KEY
008D              00353 ;key is detected
008D 0293          00354          INCF   NEW_KEY,W       ;CHK IF KEY ...
008E 0743          00355          BTFSS  STATUS,Z        ;WAS RELEASED
008F 0A89          00356          GOTO   SKP2            ;NO THEN RETURN
0090 0207          00357          MOVF   PORT_C,W        ;GET RAW KEY...
0091 0D0F          00358          IORLW  B'00001111'     ;VALUE.
0092 0151          00359          ANDWF  TEMP,W          ;
0093 0033          00360          MOVWF  NEW_KEY         ;SAVE IN NEW_KEY
0094 0998          00361          CALL  GET_KEY_VAL      ;GET ACTUAL KEY ...
0095 0033          00362          MOVWF  NEW_KEY         ;VALUE
0096 05D0          00363          BSF    FLAG,KEY_HIT    ;SET KEY HIT FLAG
0097 0A89          00364          GOTO   SKP2            ;RETURN
00365 ;
00366 PAGE
00367 ;This routine decodes the hex value from the "raw" data got
00368 ;from scanning the rows and cols.
00369 ; actual key value raw hex value
00370 ; ONE EQU 77
00371 ; TWO EQU 7B
00372 ; THREE EQU 7D
00373 ; C EQU 7E
00374 ; FOUR EQU 0B7
00375 ; FIVE EQU 0BB
00376 ; SIX EQU 0BD
00377 ; D EQU 0BE
00378 ; SEVEN EQU 0D7
00379 ; EIGHT EQU 0DB
00380 ; NINE EQU 0DD
00381 ; E EQU 0DE
00382 ; A EQU 0E7
00383 ; ZERO EQU 0EB
00384 ; B EQU 0ED

```

```

00385 ;      F      EQU      0EE
00386 ;
00387 ;
0098      00388 GET_KEY_VAL
0098 0E0F      00389      ANDLW  B'00001111'      ;GET LO NIBBLE
0099 0034      00390      MOVWF  KEY_NIBL      ;SAVE
009A 0C04      00391      MOVLW  4      ;SET COUNT TO 4
009B 0031      00392      MOVWF  TEMP      ; /
009C      00393 GKV1
009C 0503      00394      BSF    STATUS,C      ;SET CARRY
009D 0334      00395      RRF    KEY_NIBL, F      ;ROTATE NIBBLE
009E 0703      00396      BTFSS  STATUS,C      ;SKIP IF NOT Z
009F 0AA5      00397      GOTO   GET_HI_KEY      ;GOTO NEXT PART
00A0 02F1      00398      DECF   TEMP, F      ;DEC COUNT
00A1 0A9C      00399      GOTO   GKV1      ;LOOP
00A2      00400 GO_RESET
00A2 05A3      00401      BSF    STATUS,PA0      ;SET MSB
00A3 05C3      00402      BSF    STATUS,PA1      ; /
00A4 0BFF      00403      GOTO   SYS_RESET      ;ELSE BIG ERROR
00A5      00404 GET_HI_KEY
00A5 00F1      00405      DECF  TEMP, F      ;REDUCE BY 1
00A6 0393      00406      SWAPF NEW_KEY,W      ;GET HI NIBBLE
00A7 0E0F      00407      ANDLW  B'00001111'      ; /
00A8 0034      00408      MOVWF  KEY_NIBL      ;SAVE
00A9 0211      00409      MOVF  TEMP,W      ;GET OFFSET TO TBL
00AA 01E2      00410      ADDWF  PC, F      ;LOAD IN PC
00AB 0AAF      00411      GOTO   GET147A      ;JUMP TO NEXT PART
00AC 0AB8      00412      GOTO   GET2580      ; /
00AD 0ABA      00413      GOTO   GET369B      ; /
00AE 0ABC      00414      GOTO   GETCDEF      ; /
00AF      00415 ;
00AF      00416 GET147A
00AF 0C04      00417      MOVLW  4      ;SET COUNT TO 4
00B0      00418 GETCOM
00B0 0031      00419      MOVWF  TEMP      ;
00B1      00420 GETCOM1
00B1 0503      00421      BSF    STATUS,C      ;SET CARRY
00B2 0334      00422      RRF    KEY_NIBL, F      ;ROTATE RIGHT
00B3 0703      00423      BTFSS  STATUS,C      ;CHECK IF DONE
00B4 0ABE      00424      GOTO   KEY_TBL      ;JUMP TO TABLE
00B5 02F1      00425      DECF   TEMP, F      ;DEC COUNT
00B6 0AB1      00426      GOTO   GETCOM1      ;LOOP
00B7 0AA2      00427      GOTO   GO_RESET      ;ELSE ERROR
00B8      00428 ;
00B8      00429 GET2580
00B8 0C08      00430      MOVLW  8      ;SET COUNT TO 8
00B9 0AB0      00431      GOTO   GETCOM
00BA      00432 ;
00BA      00433 GET369B
00BA 0C0C      00434      MOVLW  D'12'      ;SET COUNT TO 12
00BB 0AB0      00435      GOTO   GETCOM
00BC      00436 ;
00BC      00437 GETCDEF
00BC 0C10      00438      MOVLW  D'16'      ;SET COUNT TO 16
00BD 0AB0      00439      GOTO   GETCOM
00BE      00440 PAGE
00BE      00441 ;
00BE      00442 KEY_TBL
00BE 00F1      00443      DECF  TEMP, F      ;REDUCE BY 1
00BF 0211      00444      MOVF  TEMP,W      ;GET IN W
00C0 01E2      00445      ADDWF  PC, F      ;JUMP TO TABLE
00C1 0801      00446      RETLW  1      ;KEY 1
00C2 0804      00447      RETLW  4      ;KEY 4
00C3 0807      00448      RETLW  7      ;KEY 7
00C4 080A      00449      RETLW  0A      ;KEY A
00C5 0802      00450      RETLW  2      ;KEY 2

```

```

00C6 0805      00451      RETLW  5          ;KEY 5
00C7 0808      00452      RETLW  8          ;KEY 8
00C8 0800      00453      RETLW  0          ;KEY 0
00C9 0803      00454      RETLW  3          ;KEY 3
00CA 0806      00455      RETLW  6          ;KEY 6
00CB 0809      00456      RETLW  9          ;KEY 9
00CC 080B      00457      RETLW  0B        ;KEY B
00CD 080C      00458      RETLW  0C        ;KEY C
00CE 080D      00459      RETLW  0D        ;KEY D
00CF 080E      00460      RETLW  0E        ;KEY E
00D0 080F      00461      RETLW  0F        ;KEY F
                00462 ;
                00463     PAGE
                00464 ;
00D1           00465 MASK_ANNC
00D1 0CFC      00466      MOVLW  B'11111100' ;CHK IF DIGIT 1
00D2 0186      00467      XORWF  PORT_B,0   ;      /
00D3 0643      00468      BTFSC  STATUS,Z   ;NO THEN SKIP
00D4 0AE5      00469      GOTO   MASK_ALARM ;ELSE MASK ALARM
00D5 0CF3      00470      MOVLW  B'11110011' ;CHK IF DIGIT 2
00D6 0186      00471      XORWF  PORT_B,0   ;      /
00D7 0643      00472      BTFSC  STATUS,Z   ;NO THEN SKIP
00D8 0AE8      00473      GOTO   MASK_COLON ;ELSE MASK COLON
00D9 0CCF      00474      MOVLW  B'11001111' ;CHK IF DIGIT 3
00DA 0186      00475      XORWF  PORT_B,0   ;      /
00DB 0643      00476      BTFSC  STATUS,Z   ;NO THEN SKIP
00DC 0AE1      00477      GOTO   MASK_PM    ;ELSE MASK PM
00DD           00478 MASK_AM
00DD 02A4      00479      INCF   FSR, F     ;INC FSR
00DE 07E0      00480      BTFSS  F0,AM_PM   ;IF 0 THEN AM
00DF 05F2      00481      BSF    DIGIT,7    ;SET MSB
00E0 0AEB      00482      GOTO   BLNK_LEAD_0 ;NEXT
00E1           00483 MASK_PM
00E1 02A4      00484      INCF   FSR, F     ;INC FSR
00E2 06E0      00485      BTFSS  F0,AM_PM   ;IF 1 THEN PM
00E3 05F2      00486      BSF    DIGIT,7    ;SET MSB
00E4 0AEB      00487      GOTO   BLNK_LEAD_0 ;NEXT
00E5           00488 MASK_ALARM
00E5 0650      00489      BTFSC  FLAG,ALRMLD ;1 THEN LIGHT LED
00E6 05F2      00490      BSF    DIGIT,7    ;      /
00E7 0AEB      00491      GOTO   BLNK_LEAD_0 ;NEXT
00E8           00492 MASK_COLON
00E8 0670      00493      BTFSC  FLAG,COLON ;1 THEN LIGHT LED
00E9 05F2      00494      BSF    DIGIT,7    ;      /
00EA 0AEB      00495      GOTO   BLNK_LEAD_0 ;NEXT
                00496 ;
00EB           00497 BLNK_LEAD_0
00EB 0210      00498      MOVF   FLAG,W     ;GET IN W
00EC 0E03      00499      ANDLW  B'00000011' ;SEE IF IN DEM
00ED 0F02      00500      XORLW  B'00000010' ;CHECK
00EE 0643      00501      BTFSC  STATUS,Z   ;NO THEN DO
00EF 0800      00502      RETLW  0          ;ELSE RETURN
00F0 0CFC      00503      MOVLW  B'11111100' ;SEE IF DIGIT 1
00F1 0186      00504      XORWF  PORT_B,0   ;      /
00F2 0743      00505      BTFSS  STATUS,Z   ;YES THEN SKIP
00F3 0800      00506      RETLW  0          ;RETURN
00F4 0C3F      00507      MOVLW  B'00111111' ;ELSE MASK G AND ANUNC
00F5 0152      00508      ANDWF  DIGIT,0    ;GET IN W
00F6 0F3F      00509      XORLW  B'00111111' ;SEE IF 0
00F7 0743      00510      BTFSS  STATUS,Z   ;YES THEN SKIP
00F8 0800      00511      RETLW  0          ;RETURN
00F9 0C80      00512      MOVLW  B'10000000' ;ELSE BLANK D1
00FA 0172      00513      ANDWF  DIGIT, F   ;      /
00FB 0800      00514      RETLW  0          ;RETURN
                00515 ;
                00516 ;

```

```

00517 ;
00518     PAGE
00519 ;
00520 ;THIS ROUTINE SETS UP PORTS A,B,C AND THE INTERNAL
00521 ;REAL TIME CLOCK COUNTER.
00522 INIT_CLK
00FC      00FC      00523     MOVLW   B'00001111'     ;MAKE ACTIVE HIGH
00FD      0025      00524     MOVWF   PORT_A           ; /
00FE      0C00      00525     MOVLW   B'00000000'     ;SET PORT A AS OUTPUTS
00FF      0005      00526     TRIS    PORT_A
00527 ;
0100      0CFF      00528     MOVLW   B'11111111'     ;SET LEVELS HIGH
0101      0026      00529     MOVWF   PORT_B           ; /
0102      0C00      00530     MOVLW   B'00000000'     ;SET PORT B AS OUTPUTS
0103      0006      00531     TRIS    PORT_B
00532 ;
0104      0C00      00533     MOVLW   B'00000000'     ;SET LEVELS LOW
0105      0027      00534     MOVWF   PORT_C           ; /
0106      0C00      00535     MOVLW   B'00000000'     ;SET PORT C AS OUTPUTS
0107      0007      00536     TRIS    PORT_C           ; /
00537 ;
0108      0C04      00538     MOVLW   B'00000100'     ;SET UP PRESCALER
0109      0002      00539     OPTION  ; /
00540 ;
010A      0C60      00541     MOVLW   MSEC5           ;TMR0 = 5 mSEC
010B      0021      00542     MOVWF   TMR0           ; /
010C      0068      00543     CLRF   MSTMR           ;CLEAR MSTMR
010D      0069      00544     CLRF   STMR            ; & SEC TMR
010E      006A      00545     CLRF   MTMR            ;& MINUTES
010F      0C12      00546     MOVLW   12H           ;MAKE HRS = 12
0110      002B      00547     MOVWF   HTMR           ; /
0111      002D      00548     MOVWF   HALARM         ;MAKE HRS = 12
0112      006C      00549     CLRF   MALARM         ; /
0113      0C03      00550     MOVLW   B'00000011'   ;SET TO TEST MODE
0114      0030      00551     MOVWF   FLAG           ; /
0115      0078      00552     CLRF   ALFLAG         ;CLEAR ALL FLAG
0116      0079      00553     CLRF   AAFLAG         ; /
0117      0077      00554     CLRF   ENTFLG         ; /
0118      0A01      00555     GOTO   TEST_HARDWARE
00556 ;
00557 ;All routines related to timer updates are located at
00558 ;address 200 and above.
00559     PAGE
0200      00560     ORG     0200
00561 ;
0200      00562     UPDATE_TIMERS
0200      0201      00563     MOVF   TMR0,W         ;SEE IF TMR0 = 0
0201      0743      00564     BTFSS  STATUS,Z       ;IF 0 THEN SKIP
0202      0A00      00565     GOTO   UPDATE_TIMERS ;ELSE LOOP
0203      0C60      00566     MOVLW   MSEC5         ;TMR0 = 5 mSEC
0204      0021      00567     MOVWF   TMR0           ; /
0205      02A8      00568     INCF   MSTMR, F       ;INC 5 MILLI SEC
0206      06D0      00569     BTFSC  FLAG,KEY_HIT   ;NO KEY HIT THEN SKIP
0207      0A70      00570     GOTO   CHK_DE_BOUNCE ;ELSE DEBOUNCE
0208      00571     UP_TMR_1
0208      0210      00572     MOVF   FLAG,W         ;ALARM MODE?
0209      0E03      00573     ANDLW  B'00000011'   ; /
020A      0F01      00574     XORLW  B'00000001'   ; /
020B      0743      00575     BTFSS  STATUS,Z       ;SKIP IF YES
020C      0A14      00576     GOTO   UP_TMR_2       ;DO NEXT
020D      0550      00577     BSF    FLAG,ALRMLED   ;LIGHT LED
020E      0570      00578     BSF    FLAG,COLON     ; /
020F      0C64      00579     MOVLW  D'100'        ;IF 1/2 SEC
0210      0088      00580     SUBWF  MSTMR,0        ; BLINK
0211      0703      00581     BTFSS  STATUS,C       ; /
0212      0450      00582     BCF    FLAG,ALRMLED   ;ALARM LED

```

```

0213 0A19      00583      GOTO      UP_TMR_3      ;SKIP
0214          00584      UP_TMR_2
0214 0570      00585          BSF      FLAG,COLON      ;TURN ON
0215 0C64      00586          MOVLW   D'100'          ;<100 BLINK COLON
0216 0088      00587          SUBWF   MSTMR,0          ; /
0217 0703      00588          BTFSS  STATUS,C          ;YES THEN SKIP
0218 0470      00589          BCF     FLAG,COLON      ;ELSE TURN OFF
0219          00590      UP_TMR_3
0219 0208      00591          MOVF   MSTMR,0          ;GET MSTMR IN W
021A 0FC7      00592          XORLW  D'199'          ;= 199 THEN SKIP
021B 0743      00593          BTFSS  STATUS,Z          ; /
021C 0800      00594          RETLW  0
0219          00595      ;INC SECONDS COUNT
021D 0068      00596          CLRF   MSTMR          ;CLEAR MS_TMR
021E 0216      00597          MOVF   MIN_SEC,W      ;GET MIN_SEC TIMER
021F 0E0F      00598          ANDLW  B'00001111'    ;MASK MINUTES
0220 0743      00599          BTFSS  STATUS,Z          ;ZERO THEN SKIP
0221 00F6      00600          DECF   MIN_SEC, F      ;REDUCE SECONDS
0222 0C09      00601          MOVLW  STMR           ;LOAD FSR WITH S_TMR
0223 0024      00602          MOVWF  FSR             ; /
0224 0955      00603          CALL   INC_60          ;INC SECONDS
0225 0D00      00604          IORLW  0               ;DO AN OPERATION
0226 0743      00605          BTFSS  STATUS,Z          ;IF RETURN = 0 SKIP
0227 0A38      00606          GOTO   CHK_AL_TIM     ;CHK ALRM
0219          00607      ;INC MINUTES COUNT
0228 03B6      00608          SWAPF  MIN_SEC, F      ;SWAP MIN SEC
0229 0216      00609          MOVF   MIN_SEC,W      ;GET MIN_SEC IN W
022A 0E0F      00610          ANDLW  B'00001111'    ;MASK SECONDS
022B 0743      00611          BTFSS  STATUS,Z          ;SKIP IF NOT SET
022C 00F6      00612          DECF   MIN_SEC, F      ;ELSE DEC
022D 03B6      00613          SWAPF  MIN_SEC, F      ;SWAP BACK
022E 0966      00614          CALL   CHK_SILNC_TIM   ;SILENCE ON?
022F 0C0A      00615          MOVLW  MTMR           ;INC MINUTES
0230 0024      00616          MOVWF  FSR             ; /
0231 0955      00617          CALL   INC_60          ; /
0232 0D00      00618          IORLW  0               ;DO AN OPERATION
0233 0743      00619          BTFSS  STATUS,Z          ;IF 0 THEN SKIP
0234 0A38      00620          GOTO   CHK_AL_TIM     ;CHECK ALARM TIME
0219          00621      ;INC HOUR COUNT
0235 0C0B      00622          MOVLW  HTMR           ;GET HTMR IN FSR
0236 0024      00623          MOVWF  FSR             ; /
0237 0989      00624          CALL   INC_HR          ;INC HOURS
0219          00625      ;
0238          00626      CHK_AL_TIM
0238 0718      00627          BTFSS  ALFLAG,ALONOF   ;IF OFF QUIT
0239 0800      00628          RETLW  0               ; /
023A 0658      00629          BTFSC  ALFLAG,SILNC    ;RET IF IN SILENCE
023B 0800      00630          RETLW  0
023C 0638      00631          BTFSC  ALFLAG,INAL     ;ALREADY DONE
023D 0A4D      00632          GOTO   CHK_1_MIN      ;SEE IF 1 MIN UP
0219          00633      ; RETLW  0 ;YES THEN QUIT
023E 020D      00634          MOVF   HALARM,W        ;CHK HRS
023F 018B      00635          XORWF  HTMR,W          ;EQUAL?
0240 0743      00636          BTFSS  STATUS,Z          ;YES THEN SKIP
0241 0800      00637          RETLW  0               ;ELSE RET
0242 020C      00638          MOVF   MALARM,W        ;CHK MIN
0243 018A      00639          XORWF  MTMR,W          ;EQUAL?
0244 0743      00640          BTFSS  STATUS,Z          ;YES THEN SKIP
0245 0800      00641          RETLW  0               ;ELSE RET
0246 0209      00642          MOVF   STMR,W          ;SEE IF SEC=0
0247 0743      00643          BTFSS  STATUS,Z          ;YES THEN SKIP
0248 0800      00644          RETLW  0               ;NO THEN RET
0249 0538      00645          BSF    ALFLAG,INAL     ;SET IN ALARM FLAG
024A 0C10      00646          MOVLW  10              ;SET 1 MIN TIMER
024B 0036      00647          MOVWF  MIN_SEC          ; /
024C 0800      00648          RETLW  0

```



```

00649 ;
024D 024D 0396 00650 CHK_1_MIN
024E 0E0F 00651 SWAPF MIN_SEC,W ;SWAP IN W
024F 0743 00652 ANDLW B'00001111' ;CHK MINUTES
0250 0800 00653 BTFSS STATUS,Z ;0 THEN SKIP
0251 0438 00654 RETLW 0 ;ELSE RET
0252 0478 00655 BCF ALFLAG,INAL ;CLR IN ALARM
0253 0505 00656 BCF ALFLAG,INAA ;CLR IN AA
0254 0800 00657 BSF PORT_A,BEP ;STOP BEEPER
00658 RETLW 0
00659 ;
0255 00660 INC_60
0255 02A0 00661 INCF F0, F ;INC AND GET IN W
0256 0200 00662 MOVF F0,0 ; /
0257 0E0F 00663 ANDLW B'00001111' ;MASK HI BITS
0258 0F0A 00664 XORLW B'00001010' ;= 10 THEN MAKE IT 0
0259 0743 00665 BTFSS STATUS,Z ; /
025A 0801 00666 RETLW 1 ;ELSE RETURN NON ZERO
025B 0CF0 00667 MOVLW B'11110000' ;ZERO LSB
025C 0160 00668 ANDWF F0, F ; /
025D 03A0 00669 SWAPF F0, F ;SWAP INDIRECT
025E 02A0 00670 INCF F0, F ;INC
025F 0200 00671 MOVF F0,0 ;GET IN W
0260 03A0 00672 SWAPF F0, F ;SWAP F0 BACK
0261 0F06 00673 XORLW D'6' ;=6 THEN SKIP
0262 0743 00674 BTFSS STATUS,Z ; /
0263 0801 00675 RETLW 1 ;ELSE RETURN NZ
0264 0060 00676 CLRF F0 ; /
0265 0800 00677 RETLW 0 ;RET 0
00678 ;
00679 PAGE
00680 ;
0266 00681 CHK_SILNC_TIM
0266 0758 00682 BTFSS ALFLAG,SILNC ;CHK IF IN SILENCE
0267 0800 00683 RETLW 0 ;NO THEN SKIP
0268 0396 00684 SWAPF MIN_SEC,W ;GET MIN IN W
0269 0E0F 00685 ANDLW B'00001111' ;MASK SECS
026A 0743 00686 BTFSS STATUS,Z ;ZERO?
026B 0800 00687 RETLW 0 ;NO THEN RET
026C 0458 00688 BCF ALFLAG,SILNC ;RESET SILENCE
026D 0C10 00689 MOVLW 10 ;SET 1 MIN TIMER
026E 0036 00690 MOVWF MIN_SEC ; /
026F 0800 00691 RETLW 0
00692 ;
00693 ;
0270 00694 CHK_DE_BOUNCE
0270 06B7 00695 BTFSC ENTFLG,INKEYBEP ;IN KEY BEEP?
0271 0A76 00696 GOTO CHK_DEB_1 ;YES THEN DEC TIMER
0272 07B0 00697 BTFSS FLAG,KEY_BEEP ;KEY BEEP SET?
0273 0A7F 00698 GOTO CHK_SERV ;NO, SEE IF SERVICED
0274 0678 00699 BTFSC ALFLAG,INAA ;IN AA?
0275 0A86 00700 GOTO CHK_BEP_ON ;YES THEN SEE IF ON
0276 00701 CHK_DEB_1
0276 05B7 00702 BSF ENTFLG,INKEYBEP ;SET FLAG
0277 0215 00703 MOVF DEBOUNCE,W ;GET IN W
0278 0643 00704 BTFSC STATUS,Z ;NZ THEN SKIP
0279 0C14 00705 MOVLW D'20' ;ELSE DB 100 mSEC
027A 0035 00706 MOVWF DEBOUNCE ; /
027B 0405 00707 BCF PORT_A,BEP ;TURN ON BEEPER
027C 02F5 00708 DECFSZ DEBOUNCE, F ;DEC AND CHK
027D 0A08 00709 GOTO UP_TMR_1 ;GO BACK
027E 0505 00710 BSF PORT_A,BEP ;TURN OFF BEEPER
027F 00711 CHK_SERV
00712 ; CLRF DEBOUNCE
00713 ; BSF PORT_A,BEP
027F 07F0 00714 BTFSS FLAG,SERVICED ;SERVICED THEN SKIP

```

# AN529

```
0280 0A08      00715      GOTO      UP_TMR_1      ;GO BACK
0281 04F0      00716      BCF       FLAG,SERVICED ;ELSE CLEAR FLAGS
0282 04D0      00717      BCF       FLAG,KEY_HIT  ;      /
0283 04B0      00718      BCF       FLAG,KEY_BEEP ;RESET FLAG
0284 04B7      00719      BCF       ENTFLG,INKEYBEP ;      /
0285 0A08      00720      GOTO      UP_TMR_1      ;GO BACK
00721 ;
0286          00722      CHK_BEP_ON
0286 0705      00723      BTFSS    PORT_A,BEP     ;IF OFF THEN SKIP
0287 0A08      00724      GOTO      UP_TMR_1      ;ELSE WAIT
0288 0A76      00725      GOTO      CHK_DEB_1      ;RETURN
00726 ;
00727 ;
0289          00728      INC_HR
0289 02A0      00729      INCF     F0, F          ;INC HOUR TIMER
028A 0200      00730      MOVF     F0,W           ;GET HR TMR IN W
028B 0031      00731      MOVWF    TEMP           ;SAVE IN TEMP
028C 0E0F      00732      ANDLW   B'00001111'    ;CHK LO BYTE = 10
028D 0F0A      00733      XORLW   D'10'          ;      /
028E 0743      00734      BTFSS    STATUS,Z       ;YES THEN SKIP
028F 0A93      00735      GOTO      INC_AM_PM     ;ELSE CHK 12
0290 0C10      00736      MOVLW   B'00010000'    ;LOAD 1 IN MSB
0291 0020      00737      MOVWF    F0             ;
0292 0AA3      00738      GOTO      RESTORE_AM_PM ;RESTORE AM/PM
0293          00739      INC_AM_PM
0293 04E0      00740      BCF      F0,AM_PM       ;CLEAR AM/PM
0294 0200      00741      MOVF     F0,W           ;GET IN W
0295 0F12      00742      XORLW   12H            ;SEE IF 12 HEX
0296 0743      00743      BTFSS    STATUS,Z       ;YES THEN SKIP
0297 0A9D      00744      GOTO      CHK_13        ;ELSE CHK 13
0298 07F1      00745      BTFSS    TEMP,AM_PM     ;IF SET, SKIP
0299 0A9C      00746      GOTO      SET_AM_PM     ;ELSE SET
029A 04E0      00747      BCF      F0,AM_PM       ;CLEAR FLAG
029B 0800      00748      RETLW   0               ;RETURN
029C          00749      SET_AM_PM
029C 05E0      00750      BSF      F0,AM_PM       ;SET FLAG
029D          00751      CHK_13
029D 0200      00752      MOVF     F0,W           ;GET IN W
029E 0F13      00753      XORLW   13H            ;SEE IF 13
029F 0743      00754      BTFSS    STATUS,Z       ;YES THEN SKIP
02A0 0AA3      00755      GOTO      RESTORE_AM_PM ;
02A1          00756      SET_1_HR
02A1 0C01      00757      MOVLW   B'00000001'    ;SET TO 1
02A2 0020      00758      MOVWF    F0             ;
02A3          00759      RESTORE_AM_PM
02A3 06F1      00760      BTFSC    TEMP,AM_PM     ;SKIP IF AM
02A4 05E0      00761      BSF      F0,AM_PM       ;ELSE SET TO PM
02A5 0800      00762      RETLW   0               ;
00763 ;
00764 ;
00765 ;
00766      PAGE
0400          00767      ORG      400
00768 ;
00769 ;KEY DEFINITIONS
0000000A      00770      ALARM_KEY EQU 0A
0000000B      00771      CE_KEY EQU 0B
0000000C      00772      SNOOZE_KEY EQU 0C
0000000D      00773      AM_PM_KEY EQU 0D
0000000E      00774      CLR_ALARM_KEY EQU 0E
0000000F      00775      SET_KEY EQU 0F
00776 ;
0400          00777      SERVICE_KEYS
0400 07D0      00778      BTFSS    FLAG,KEY_HIT   ;NO KEY HIT THEN ...
0401 0800      00779      RETLW   0               ;RETURN
0402 06F0      00780      BTFSC    FLAG,SERVICED ;IF NOT SERVICED SKIP
```

```

0403 0800          00781      RETLW    0              ;ELSE RETURN
0404 05F0          00782      BSF     FLAG,SERVICED ;SET SERVICED FLAG
0405 0210          00783      MOVF    FLAG,W        ;GET MODE OF OPERATION
0406 0E03          00784      ANDLW  B'00000011'   ;
0407 0643          00785      BTFSC  STATUS,Z      ;00 THEN RTM
0408 0A10          00786      GOTO   RTMKS         ;RTM KEY SERVICE
0409 0031          00787      MOVWF  TEMP          ;SAVE IN TEMP
040A 02F1          00788      DECFSZ TEMP, F      ;REDUCE TEMP
040B 0A0D          00789      GOTO   SK1           ;SKIP
040C 0A1D          00790      GOTO   ATMKS         ;01, DO ALARM MODE
040D              00791      SK1
040D 02F1          00792      DECFSZ TEMP, F      ;REDUCE TEMP
040E 0800          00793      RETLW  0              ;11 THEN RETURN
040F 0A2A          00794      GOTO   DEMKS         ;10, DATA ENTRY MODE
00795 ;
00796 ;REAL TIME MODE KEY SERVICE
00797 RTMKS
0410 09BA          00798      CALL   CHK_AL_KEYS   ;CHK ALARM KEYS
0411 0D00          00799      IORLW  0              ;SEE IF NZ RET
0412 0643          00800      BTFSC  STATUS,Z      ;NZ THEN SKIP
0413 0800          00801      RETLW  0              ;ELSE RETURN
0414 0C0F          00802      MOVLW  SET_KEY       ;SEE IF SET KEY
0415 0193          00803      XORWF  NEW_KEY,W     ;
0416 0643          00804      BTFSC  STATUS,Z      ;NO THEN SKIP
0417 0A91          00805      GOTO   SERV_SET_RTM  ;SERVICE SET KEY
0418 0C0A          00806      MOVLW  ALARM_KEY     ;ALARM KEY?
0419 0193          00807      XORWF  NEW_KEY,W     ;
041A 0643          00808      BTFSC  STATUS,Z      ;NO THEN SKIP
041B 0AAB          00809      GOTO   SERV_ALARM_RTM ;ELSE SERVICE ALARM
041C              00810      IGNORE_KEY
041C 0800          00811      RETLW  0              ;ELSE RETURN
00812 ;
00813 ;ALARM TIME MODE KEY SERVICE
00814 ATMKS
041D 09BA          00815      CALL   CHK_AL_KEYS   ;CHECK ALARM KEYS
041E 0D00          00816      IORLW  0              ;CHECK IF 0
041F 0643          00817      BTFSC  STATUS,Z      ;NZ THEN SKIP
0420 0800          00818      RETLW  0              ;ELSE RETURN
0421 0C0F          00819      MOVLW  SET_KEY       ;SEE IF SET KEY
0422 0193          00820      XORWF  NEW_KEY,W     ;
0423 0643          00821      BTFSC  STATUS,Z      ;NO THEN SKIP
0424 0A9C          00822      GOTO   SERV_SET_ATM  ;SERVICE SET ATM
0425 0C0A          00823      MOVLW  ALARM_KEY     ;GET ALARM KEY
0426 0193          00824      XORWF  NEW_KEY,W     ;SEE IF HIT
0427 0643          00825      BTFSC  STATUS,Z      ;NO THEN SKIP
0428 0AA2          00826      GOTO   SERV_ALARM_ATM ;ELSE SERVICE
0429 0A1C          00827      GOTO   IGNORE_KEY
00828 ;
00829 ;DATA ENTRY MODE KEY SERVICE
00830 DEMKS
042A 09BA          00831      CALL   CHK_AL_KEYS   ;CHECK ALARM KEYS
042B 0D00          00832      IORLW  0              ;CHK IF 0
042C 0643          00833      BTFSC  STATUS,Z      ;NZ THEN SKIP
042D 0800          00834      RETLW  0              ;ELSE RETURN
042E 0C0F          00835      MOVLW  SET_KEY       ;IF SET KEY THEN END
042F 0193          00836      XORWF  NEW_KEY,W     ;
0430 0643          00837      BTFSC  STATUS,Z      ;NO THEN SKIP
0431 0A3F          00838      GOTO   DEMKS_END     ;GOTO END
0432 0C0B          00839      MOVLW  CE_KEY        ;IF CLEAR ENTRY
0433 0193          00840      XORWF  NEW_KEY,W     ;
0434 0643          00841      BTFSC  STATUS,Z      ;SKIP IF NO
0435 0A48          00842      GOTO   DEMKS_END_1   ;ABANDON ENTRY
0436 0737          00843      BTFSS  ENTFLG,HR10   ;10'S HRS DONE?
0437 0A54          00844      GOTO   ENT_HR_10     ;NO THEN GET
0438 0757          00845      BTFSS  ENTFLG,HR     ;HRS DONE?
0439 0A5F          00846      GOTO   ENT_HRS       ;NO THEN GET

```

```

043A 0777      00847      BTFSS  ENTFLG,MIN10    ;10'S MIN. DONE?
043B 0A72      00848      GOTO   ENT_MIN_10   ;NO THEN GET
043C 0797      00849      BTFSS  ENTFLG,MIN    ;MIN DONE?
043D 0A7F      00850      GOTO   ENT_MIN      ;NO THEN GET
043E 0A87      00851      GOTO   ENT_AM_PM    ;NO THEN GET
043F          00852  DEMKS_END
043F 0717      00853      BTFSS  ENTFLG,RTATS  ;GET OLD STATUS
0440 0A4D      00854      GOTO   LD_RTM       ;LOAD IN TIME
0441 020E      00855      MOVF   MENTRY,W     ;LD IN ALARM
0442 002C      00856      MOVWF  MALARM       ;
0443 020F      00857      MOVF   HENTRY,W     ;
0444 002D      00858      MOVWF  HALARM       ;
0445 0450      00859      BCF    FLAG,ALRMLD  ;CLEAR FLAG
0446 0618      00860      BTFSC  ALFLAG,ALONOF;SEE IF ON-OFF
0447 0550      00861      BSF    FLAG,ALRMLD  ;ELSE SET
0448          00862  DEMKS_END_1
0448 0410      00863      BCF    FLAG,0       ;RTM MODE
0449 0430      00864      BCF    FLAG,1       ;
044A 0490      00865      BCF    FLAG,FLASH   ;STOP FLASH
044B          00866  SERV_COM_RET
044B 05B0      00867      BSF    FLAG,KEY_BEEP
044C 0800      00868      RETLW  0            ;RETURN
044C          00869  ;
044D          00870  LD_RTM
044D 020E      00871      MOVF   MENTRY,W     ;LD IN RTM
044E 002A      00872      MOVWF  MTMR         ;
044F 020F      00873      MOVF   HENTRY,W     ;
0450 002B      00874      MOVWF  HTMR         ;
0451 0068      00875      CLRF   MSTMR        ;CLR TIME
0452 0069      00876      CLRF   STMR         ;
0453 0A48      00877      GOTO   DEMKS_END_1 ;GO BACK
0453          00878  ;
0454          00879  ENT_HR_10
0454 0213      00880      MOVF   NEW_KEY,W    ;SEE IF 0
0455 0643      00881      BTFSC  STATUS,Z     ;NZ THEN SKIP
0456 0A5C      00882      GOTO   LD_HENTRY_0  ;LOAD 0
0457 02D3      00883      DECFSZ NEW_KEY,0    ;1 THE SKIP
0458 0A1C      00884      GOTO   IGNORE_KEY   ;ELSE IGNORE KEY
0459 058F      00885      BSF    HENTRY,4     ;SET TO 1
045A 0537      00886      BSF    ENTFLG,HR10 ;SET FLAG
045B 0A4B      00887      GOTO   SERV_COM_RET ;GO GET NEXT
045C          00888  LD_HENTRY_0
045C 048F      00889      BCF    HENTRY,4     ;SET TO 0
045D 0537      00890      BSF    ENTFLG,HR10 ;
045E 0A4B      00891      GOTO   SERV_COM_RET ;
045F          00892  ENT_HRS
045F 0C0F      00893      MOVLW  HENTRY       ;USE INDIRECT ADDR.
0460 0024      00894      MOVWF  FSR          ;
0461 068F      00895      BTFSC  HENTRY,4     ;SEE IF 0
0462 0A6D      00896      GOTO   ALLOW0_2    ;YES THEN 0,1&2
0463 0C0A      00897      MOVLW  D'10'       ;SEE IF 0 - 9
0464 0093      00898      SUBWF  NEW_KEY,W    ;
0465 0603      00899      BTFSC  STATUS,C     ;IF C THEN SKIP
0466 0A1C      00900      GOTO   IGNORE_KEY   ;ELSE IGNORE
0467          00901  ENT_LO_COM1
0467 0557      00902      BSF    ENTFLG,HR    ;SET FLAG
0468          00903  ENT_LO_COM
0468 0200      00904      MOVF   F0,W         ;LD HRS
0469 0EF0      00905      ANDLW  B'11110000' ;MASK LO NIBL
046A 0113      00906      IORWF  NEW_KEY,W    ;OR NEW KEY
046B 0020      00907      MOVWF  F0           ;SAVE BACK
046C 0A4B      00908      GOTO   SERV_COM_RET ;GET NEXT
046D          00909  ALLOW0_2
046D 0C03      00910      MOVLW  D'3'        ;SEE IF 0 - 2
046E 0093      00911      SUBWF  NEW_KEY,W    ;
046F 0603      00912      BTFSC  STATUS,C     ;<3 THEN SKIP

```

```

0470 0A1C          00913          GOTO    IGNORE_KEY
0471 0A67          00914          GOTO    ENT_LO_COM1      ;      /
                                00915          ;
0472              00916 ENT_MIN_10
0472 0C0E          00917          MOVLW  MENTRY           ;DO INDIRECT ADDR.
0473 0024          00918          MOVWF  FSR              ;      /
0474 0C06          00919          MOVLW  D'6'            ;ALLOW 0 - 5
0475 0093          00920          SUBWF  NEW_KEY,W       ;      /
0476 0603          00921          BTFSC  STATUS,C       ;IF C THEN SKIP
0477 0A1C          00922          GOTO   IGNORE_KEY     ;ELSE IGNORE
0478 0380          00923          SWAPF  F0,W           ;SWAP AND GET
0479 0EF0          00924          ANDLW  B'11110000'    ;MASK LO NIBL
047A 0113          00925          IORWF  NEW_KEY,W      ;OR NEW KEY
047B 0020          00926          MOVWF  F0              ;SAVE BACK
047C 03A0          00927          SWAPF  F0, F          ;SWAP BACK
047D 0577          00928          BSF    ENTFLG,MIN10
047E 0A4B          00929          GOTO   SERV_COM_RET   ;GET NEXT
                                00930          ;
047F              00931 ENT_MIN
047F 0C0E          00932          MOVLW  MENTRY           ;DO INDIRECT
0480 0024          00933          MOVWF  FSR              ;      /
0481 0C0A          00934          MOVLW  D'10'          ;ALLOW 0 - 9
0482 0093          00935          SUBWF  NEW_KEY,W       ;SEE IF >
0483 0603          00936          BTFSC  STATUS,C       ;NO THEN SKIP
0484 0A1C          00937          GOTO   IGNORE_KEY     ;ELSE IGNORE
0485 0597          00938          BSF    ENTFLG,MIN     ;SET FLAG
0486 0A68          00939          GOTO   ENT_LO_COM1   ;      /
                                00940          ;
0487              00941 ENT_AM_PM
0487 0C0D          00942          MOVLW  AM_PM_KEY      ;AM/PM KEY?
0488 0193          00943          XORWF  NEW_KEY,W       ;      /
0489 0743          00944          BTFSS  STATUS,Z       ;YES THEN SKIP
048A 0A1C          00945          GOTO   IGNORE_KEY
048B 07EF          00946          BTFSS  HENTRY,AM_PM   ;TEST BIT
048C 0A8F          00947          GOTO   SETAMPM        ;ELSE SET
048D 04EF          00948          BCF    HENTRY,AM_PM   ;CLEAR FLAG
048E 0A4B          00949          GOTO   SERV_COM_RET   ;GOTO END
048F              00950 SETAMPM
048F 05EF          00951          BSF    HENTRY,AM_PM   ;SET FLAG
0490 0A4B          00952          GOTO   SERV_COM_RET
                                00953          ;
                                00954          PAGE
                                00955          ;
0491              00956 SERV_SET_RTM
0491 020A          00957          MOVF   MTMR,W         ;TRANSFER TIME
0492 002E          00958          MOVWF  MENTRY         ;TO DATA ENTRY
0493 020B          00959          MOVF   HTMR,W         ;      /
0494 002F          00960          MOVWF  HENTRY         ;      /
0495              00961 SERV_COM
0495 0210          00962          MOVF   FLAG,W         ;SAVE IN W
0496 0E01          00963          ANDLW  B'00000001'    ;ATM OR RTM MODE?
0497 0037          00964          MOVWF  ENTFLG        ;SAVE IN ENTFLG
0498 0CF2          00965          MOVLW  B'11110010'    ;FORCE 1S
0499 0130          00966          IORWF  FLAG, F       ;      /
049A 0410          00967          BCF    FLAG,0        ;      /
049B 0800          00968          RETLW  0
                                00969          ;
049C              00970 SERV_SET_ATM
049C 020C          00971          MOVF   MALARM,W       ;TRANSFER ALARM
049D 002E          00972          MOVWF  MENTRY         ;TO DATA ENTRY
049E 020D          00973          MOVF   HALARM,W      ;      /
049F 002F          00974          MOVWF  HENTRY         ;      /
04A0 0518          00975          BSF    ALFLAG,ALONOF  ;SET FLAG
04A1 0A95          00976          GOTO   SERV_COM      ;GOTO COMMON
                                00977          ;
04A2              00978 SERV_ALARM_ATM

```

```

04A2 0718      00979      BTFSS  ALFLAG,ALONOF  ;TEST ON/OFF
04A3 0AA6      00980      GOTO   SET_ALONOF   ;SET ON/OF FLG
04A4 0418      00981      BCF    ALFLAG,ALONOF ;CLEAR FLAG
04A5 0AA7      00982      GOTO   SERV_ATM_COM ;RET THRO COM
04A6           00983      SET_ALONOF
04A6 0518      00984      BSF    ALFLAG,ALONOF ;SET FLAG
04A7           00985      SERV_ATM_COM
04A7 05B0      00986      BSF    FLAG,KEY_BEEP ;BEEP
04A8 0CF0      00987      MOVLW B'11110000'    ;CLEAR SEC COUNT
04A9 0176      00988      ANDWF  MIN_SEC, F    ; /
04AA 0800      00989      RETLW  0              ;RETURN
                00990      ;
04AB           00991      SERV_ALARM_RTM
04AB 05B0      00992      BSF    FLAG,KEY_BEEP ;SET BEEP FLAG
04AC 0510      00993      BSF    FLAG,0         ;SET TO ALARM TIME
04AD 0430      00994      BCF    FLAG,1         ; /
04AE 0C05      00995      MOVLW  D'05'         ;SAVE 5 IN MIN_SEC
04AF 0036      00996      MOVWF  MIN_SEC       ; /
04B0 0800      00997      RETLW  0              ;
                00998      ;
04B1           00999      SERV_SNOOZE
04B1 0CA0      01000      MOVLW  0A0           ;SNOOZE FOR 10 MINS
04B2 0036      01001      MOVWF  MIN_SEC       ; /
04B3 0558      01002      BSF    ALFLAG,SILNC  ;SET FLAG
04B4           01003      CLR_AL_COM
04B4 05B0      01004      BSF    FLAG,KEY_BEEP ;SET BEEP FLAG
04B5 007A      01005      CLRFB  AATMR         ;RESET AA TIMER
04B6 0079      01006      CLRFB  AAFLAG        ;CLEAR AA FLAGS
04B7 0478      01007      BCF    ALFLAG,INAA   ;RESET INAA FLAG
04B8 0505      01008      BSF    PORT_A,BEP    ;TURN OFF BEEPER
04B9 0800      01009      RETLW  0              ;RET
                01010      ;
04BA           01011      CHK_AL_KEYS
04BA 0718      01012      BTFSS  ALFLAG,ALONOF ;ALARM ON?
04BB 0801      01013      RETLW  1              ;NO THEN RET
04BC 0738      01014      BTFSS  ALFLAG,INAL   ;IN ALARM?
04BD 0801      01015      RETLW  1              ;NO THEN SKIP
04BE 0C0E      01016      MOVLW  CLR_ALARM_KEY ;CHECK IF CLR ALARM
04BF 0193      01017      XORWF  NEW_KEY,W     ; /
04C0 0643      01018      BTFSC  STATUS,Z      ;NO THEN SKIP
04C1 0AC7      01019      GOTO   CLR_ALARM     ;ELSE CLEAR ALARM
04C2 0C0C      01020      MOVLW  SNOOZE_KEY    ;SEE IF SNOOZE HIT
04C3 0193      01021      XORWF  NEW_KEY,W     ; /
04C4 0743      01022      BTFSS  STATUS,Z      ;YES THEN SKIP
04C5 0801      01023      RETLW  1              ;
04C6 0AB1      01024      GOTO   SERV_SNOOZE   ;
                01025      ;
04C7           01026      CLR_ALARM
04C7 0438      01027      BCF    ALFLAG,INAL   ;CLEAR ALARM
04C8 0458      01028      BCF    ALFLAG,SILNC  ;CLEAR SILENCE
04C9 0C0F      01029      MOVLW  B'00001111'   ;CLEAR MINS
04CA 0176      01030      ANDWF  MIN_SEC, F    ; /
04CB 0AB4      01031      GOTO   CLR_AL_COM
                01032      ;
0600           01033      ORG    600
                01034      ;If the AA alarm is set, then this routine takes care of
                01035      ;the timing in sounding the alarm.
                01036      ;
0600           01037      SOUND_AA
0600 0738      01038      BTFSS  ALFLAG,INAL   ;SKIP IF IN ALRM
0601 0800      01039      RETLW  0              ;ELSE RETURN
0602 0658      01040      BTFSC  ALFLAG,SILNC  ;SKIP IF NOT IN SILENCE
0603 0800      01041      RETLW  0              ;ELSE RET
0604 06B7      01042      BTFSC  ENTFLG,INKEYBEP ;SKIP IF NOT IN KEY BEEP
0605 0A55      01043      GOTO   CHK_COLSN     ;CHK COLLISION
0606           01044      SND_AA_0

```

```

0606 0778      01045      BTFSS  ALFLAG,INAA      ;SKIP IF IN AA
0607           01046  SND_AA_1
0607 0919      01047      CALL   INIT_AA          ;INIT ALL
0608 0719      01048      BTFSS  AAFLAG,0        ;SKIP IF DONE
0609 0A21      01049      GOTO   DO_CYCLO        ;DO FIRST CYCL
060A 0739      01050      BTFSS  AAFLAG,1        ;SKIP IF DONE
060B 0A29      01051      GOTO   DO_CYCL1       ;ELSE 2ND CYCLE
060C 0759      01052      BTFSS  AAFLAG,2        ;SKIP IF DONE
060D 0A31      01053      GOTO   DO_CYCL2       ;ELSE DO 3RD CYCLE
060E 0779      01054      BTFSS  AAFLAG,3        ;SKIP IF DONE
060F 0A39      01055      GOTO   DO_CYCL3       ;DO CYCLE 4
0610 0799      01056      BTFSS  AAFLAG,4        ;SKIP IF DONE
0611 0A3E      01057      GOTO   DO_CYCL4       ;DO CYCLE 5
0612 07B9      01058      BTFSS  AAFLAG,5        ;SKIP IF DONE
0613 0A43      01059      GOTO   DO_CYCL5       ;DO CYCLE 6
0614 07D9      01060      BTFSS  AAFLAG,6        ;SKIP IF DONE
0615 0A48      01061      GOTO   DO_CYCL6       ;DO CYCLE 6
0616 07F9      01062      BTFSS  AAFLAG,7        ;SKIP IF DONE
0617 0A50      01063      GOTO   DO_CYCL7       ;DO CYCLE 7
0618 0A07      01064      GOTO   SND_AA_1       ;GO BACK
                01065 ;
0619           01066  INIT_AA
0619 0079      01067      CLRFB  AAFLAG          ;CLEAR ALL FLAGS
061A 0578      01068      BSFB  ALFLAG,INAA     ;SET IN AA FLAG
061B 0A2D      01069      GOTO   PUT_ON_100     ;ON 100 MSECS
                01070 ;
061C           01071  DEC_AA_TMR
061C 00FA      01072      DECF  AATMR, F        ;REDUCE TIMER
061D 021A      01073      MOVFB AATMR,W         ;GET IN W
061E 0743      01074      BTFSS STATUS,Z        ;CHECK IF Z
061F 0801      01075      RETLW 1                ;NO THEN NZ
0620 0800      01076      RETLW 0                ;ELSE 0
                01077 ;
0621           01078  DO_CYCLO
0621 091C      01079      CALL  DEC_AA_TMR       ;REDUCE TIMER
0622 0743      01080      BTFSS STATUS,Z        ;IF NZ THEN RET
0623 0800      01081      RETLW 0                ;
0624 0519      01082      BSFB  AAFLAG,0        ;SET DONE FLAG
0625           01083  PUT_OFF_100
0625 0505      01084      BSFB  PORT_A,BEP      ;TURN OFF BEEPER
0626 0C14      01085      MOVLW D'20'           ;FOR 100 MSECS
0627 003A      01086      MOVWF AATMR           ;
0628 0800      01087      RETLW 0                ;
                01088 ;
0629           01089  DO_CYCL1
0629 091C      01090      CALL  DEC_AA_TMR       ;REDUCE TIMER
062A 0743      01091      BTFSS STATUS,Z        ;IF NZ THEN RET
062B 0800      01092      RETLW 0                ;
062C 0539      01093      BSFB  AAFLAG,1        ;SET DONE FLAG
062D           01094  PUT_ON_100
062D 0405      01095      BCFB  PORT_A,BEP      ;TURN ON BEEPER
062E 0C14      01096      MOVLW D'20'           ;FOR 100 MSECS
062F 003A      01097      MOVWF AATMR           ;
0630 0800      01098      RETLW 0                ;
                01099 ;
0631           01100  DO_CYCL2
0631 091C      01101      CALL  DEC_AA_TMR       ;REDUCE TIMER
0632 0743      01102      BTFSS STATUS,Z        ;IF NZ THEN RET
0633 0800      01103      RETLW 0                ;
0634 0559      01104      BSFB  AAFLAG,2        ;SET DONE FLAG
0635 0505      01105      BSFB  PORT_A,BEP      ;TURN OFF BEEPER
0636 0C64      01106      MOVLW D'100'          ;FOR 500 MSECS
0637 003A      01107      MOVWF AATMR           ;
0638 0800      01108      RETLW 0                ;
                01109 ;
0639           01110  DO_CYCL3

```

# AN529

```
0639 091C      01111      CALL      DEC_AA_TMR      ;REDUCE TIMER
063A 0743      01112      BTFSS    STATUS,Z        ;IF NZ THEN RET
063B 0800      01113      RETLW    0                ; /
063C 0579      01114      BSF      AAFLAG,3        ;SET DONE FLAG
063D 0A2D      01115      GOTO     PUT_ON_100      ;DO NEXT CYCLE
01116 ;
063E          01117 DO_CYCL4
063E 091C      01118      CALL      DEC_AA_TMR      ;REDUCE TIMER
063F 0743      01119      BTFSS    STATUS,Z        ;IF NZ THEN RET
0640 0800      01120      RETLW    0                ; /
0641 0599      01121      BSF      AAFLAG,4        ;SET DONE FLAG
0642 0A25      01122      GOTO     PUT_OFF_100     ;DO NEXT CYCLE
01123 ;
0643          01124 DO_CYCL5
0643 091C      01125      CALL      DEC_AA_TMR      ;REDUCE TIMER
0644 0743      01126      BTFSS    STATUS,Z        ;IF NZ THEN RET
0645 0800      01127      RETLW    0                ; /
0646 05B9      01128      BSF      AAFLAG,5        ;SET DONE FLAG
0647 0A2D      01129      GOTO     PUT_ON_100      ;DO NEXT CYCLE
01130 ;
0648          01131 DO_CYCL6
0648 091C      01132      CALL      DEC_AA_TMR      ;REDUCE TIMER
0649 0743      01133      BTFSS    STATUS,Z        ;IF NZ THEN RET
064A 0800      01134      RETLW    0                ; /
064B 05D9      01135      BSF      AAFLAG,6        ;SET DONE FLAG
064C 0505      01136      BSF      PORT_A,BEP      ;TURN OFF BEEPER
064D 0CC8      01137      MOVLW   D'200'          ;FOR 1000 MSECS
064E 003A      01138      MOVWF   AATMR           ; /
064F 0800      01139      RETLW    0
01140 ;
0650          01141 DO_CYCL7
0650 091C      01142      CALL      DEC_AA_TMR      ;REDUCE TIMER
0651 0743      01143      BTFSS    STATUS,Z        ;IF NZ THEN RET
0652 0800      01144      RETLW    0                ; /
0653 05F9      01145      BSF      AAFLAG,7        ;SET DONE FLAG
0654 0A2D      01146      GOTO     PUT_ON_100      ;DO NEXT CYCLE
01147 ;
0655          01148 CHK_COLSN
0655 0605      01149      BTFSC   PORT_A,BEP      ;IF ON THEN SKIP
0656 0A06      01150      GOTO     SND_AA_0        ;ELSE RET
0657 021A      01151      MOVF    AATMR,W          ;GET TIMER
0658 0643      01152      BTFSC   STATUS,Z        ;NZ THEN SKIP
0659 0A5C      01153      GOTO     LD_AAT_1        ;LOAD A 1 IN TMR
065A 00FA      01154      DECF    AATMR, F        ;REDUCE TIMER
065B 0800      01155      RETLW    0                ;RETURN
065C          01156 LD_AAT_1
065C 02BA      01157      INCF    AATMR, F        ;INC TIMER
065D 0800      01158      RETLW    0                ;RET
01159 ;
07FF          01160      ORG     PIC57
07FF          01161 SYS_RESET
07FF 0A00      01162      GOTO     START
01163 ;
01164      END
```



```
0000 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
0040 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
0080 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
00C0 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
0100 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXX----- -----X
0200 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
0240 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
0280 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXX----- -----X
0400 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
0440 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
0480 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
04C0 : XXXXXXXXXXXXXXX----- -----X
0600 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
0640 : XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXX----- -----X
07C0 : -----X
```

All other memory blocks unused.

Program Memory Words Used: 746

Program Memory Words Free: 1302

```
Errors : 0
Warnings : 0 reported, 0 suppressed
Messages : 0 reported, 6 suppressed
```

---

---

**Note the following details of the code protection feature on PICmicro® MCUs.**

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable”.
- Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

---

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

#### **Trademarks**

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

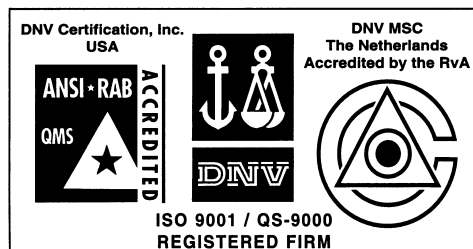
dsPIC, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, MXDEV, PICC, PICDEM, PICDEM.net, rPIC, Select Mode and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Turn Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

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

 Printed on recycled paper.



*Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.*



# MICROCHIP

## WORLDWIDE SALES AND SERVICE

### AMERICAS

#### Corporate Office

2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7200 Fax: 480-792-7277  
Technical Support: 480-792-7627  
Web Address: <http://www.microchip.com>

#### Rocky Mountain

2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7966 Fax: 480-792-7456

#### Atlanta

500 Sugar Mill Road, Suite 200B  
Atlanta, GA 30350  
Tel: 770-640-0034 Fax: 770-640-0307

#### Boston

2 Lan Drive, Suite 120  
Westford, MA 01886  
Tel: 978-692-3848 Fax: 978-692-3821

#### Chicago

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

#### Dallas

4570 Westgrove Drive, Suite 160  
Addison, TX 75001  
Tel: 972-818-7423 Fax: 972-818-2924

#### Detroit

Tri-Atria Office Building  
32255 Northwestern Highway, Suite 190  
Farmington Hills, MI 48334  
Tel: 248-538-2250 Fax: 248-538-2260

#### Kokomo

2767 S. Albright Road  
Kokomo, Indiana 46902  
Tel: 765-864-8360 Fax: 765-864-8387

#### Los Angeles

18201 Von Karman, Suite 1090  
Irvine, CA 92612  
Tel: 949-263-1888 Fax: 949-263-1338

#### New York

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

#### San Jose

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

#### Toronto

6285 Northam Drive, Suite 108  
Mississauga, Ontario L4V 1X5, Canada  
Tel: 905-673-0699 Fax: 905-673-6509

### ASIA/PACIFIC

#### Australia

Microchip Technology Australia Pty Ltd  
Suite 22, 41 Rawson Street  
Epping 2121, NSW  
Australia  
Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

#### China - Beijing

Microchip Technology Consulting (Shanghai)  
Co., Ltd., Beijing Liaison Office  
Unit 915  
Bei Hai Wan Tai Bldg.  
No. 6 Chaoyangmen Beidajie  
Beijing, 100027, No. China  
Tel: 86-10-85282100 Fax: 86-10-85282104

#### China - Chengdu

Microchip Technology Consulting (Shanghai)  
Co., Ltd., Chengdu Liaison Office  
Rm. 2401, 24th Floor,  
Ming Xing Financial Tower  
No. 88 TIDU Street  
Chengdu 610016, China  
Tel: 86-28-6766200 Fax: 86-28-6766599

#### China - Fuzhou

Microchip Technology Consulting (Shanghai)  
Co., Ltd., Fuzhou Liaison Office  
Unit 28F, World Trade Plaza  
No. 71 Wusi Road  
Fuzhou 350001, China  
Tel: 86-591-7503506 Fax: 86-591-7503521

#### China - Shanghai

Microchip Technology Consulting (Shanghai)  
Co., Ltd.  
Room 701, Bldg. B  
Far East International Plaza  
No. 317 Xian Xia Road  
Shanghai, 200051  
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

#### China - Shenzhen

Microchip Technology Consulting (Shanghai)  
Co., Ltd., Shenzhen Liaison Office  
Rm. 1315, 13/F, Shenzhen Kerry Centre,  
Renminnan Lu  
Shenzhen 518001, China  
Tel: 86-755-2350361 Fax: 86-755-2366086

#### Hong Kong

Microchip Technology Hongkong Ltd.  
Unit 901-6, Tower 2, Metroplaza  
223 Hing Fong Road  
Kwai Fong, N.T., Hong Kong  
Tel: 852-2401-1200 Fax: 852-2401-3431

#### India

Microchip Technology Inc.  
India Liaison Office  
Divyasree Chambers  
1 Floor, Wing A (A3/A4)  
No. 11, O'Shaugnessey Road  
Bangalore, 560 025, India  
Tel: 91-80-2290061 Fax: 91-80-2290062

### Japan

Microchip Technology Japan K.K.  
Benex S-1 6F  
3-18-20, Shinyokohama  
Kohoku-Ku, Yokohama-shi  
Kanagawa, 222-0033, Japan  
Tel: 81-45-471-6166 Fax: 81-45-471-6122

### Korea

Microchip Technology Korea  
168-1, Youngbo Bldg. 3 Floor  
Samsung-Dong, Kangnam-Ku  
Seoul, Korea 135-882  
Tel: 82-2-554-7200 Fax: 82-2-558-5934

### Singapore

Microchip Technology Singapore Pte Ltd.  
200 Middle Road  
#07-02 Prime Centre  
Singapore, 188980  
Tel: 65-6334-8870 Fax: 65-6334-8850

### Taiwan

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

### EUROPE

#### Denmark

Microchip Technology Nordic ApS  
Regus Business Centre  
Lautrup høj 1-3  
Ballerup DK-2750 Denmark  
Tel: 45 4420 9895 Fax: 45 4420 9910

#### France

Microchip Technology SARL  
Parc d'Activite du Moulin de Massy  
43 Rue du Saule Trapu  
Batiment A - 1er Etage  
91300 Massy, France  
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

#### Germany

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

#### Italy

Microchip Technology SRL  
Centro Direzionale Colleoni  
Palazzo Taurus 1 V. Le Colleoni 1  
20041 Agrate Brianza  
Milan, Italy  
Tel: 39-039-65791-1 Fax: 39-039-6899883

#### United Kingdom

Arizona Microchip Technology Ltd.  
505 Eskdale Road  
Winnersh Triangle  
Wokingham  
Berkshire, England RG41 5TU  
Tel: 44 118 921 5869 Fax: 44-118 921-5820

03/01/02