```
VASM ROM ASSEMBLY          REV.  6/81A            HP-82182A TIMER MODULE

OPTIONS: L C S          COCONUT TIMER             ADDRESSES @50000-51777

     2                  FILE   BWTMB1             COCONUT TIMER Q1 = TM0
*
*************************************************************************
* COCONUT TIMER ROM FUNCTIONS LIST                                      *
*************************************************************************
*
     8    0          32 CON    26                 ROM ID 0026
     9    1          36 CON    30                 # OF FUNCTIONS
    10    2           0 DEFR4K HEADER        00 - MODULE ID: -TIME- C
    10    3           0
    11    4           0 DEFR4K ADATE         01 - DATE TO ALPHA REGISTER
    11    5           0
    12    6           0 DEFR4K ALMCAT        02 - ALARM CATALOG FUNCTION
    12    7           0
    13   10           0 DEFR4K ALMNOW        03 - RUN OLDEST PD LBL ALRM
    13   11           0
    14   12           0 DEFR4K ATIME         04 - TIME TO ALPHA REGISTER
    14   13           0
    15   14           0 DEFR4K ATIM24        05 - 24HR FMT TIME TO ALPHA
    15   15           0
    16   16           0 DEFR4K CLK12         06 - SET 12-HOUR CLK FORMAT
    16   17           0
    17   20           0 DEFR4K CLK24         07 - SET 24-HOUR CLK FORMAT
    17   21           0
    18   22           0 DEFR4K CLKT          08 - SET TIME ONLY FORMAT
    18   23           0
    19   24           0 DEFR4K CLKTD         09 - SET TIME & DATE FORMAT
    19   25           0
    20   26           0 DEFR4K CLOCK         10 - START THE CLOCK MODE
    20   27           0
    21   30           0 DEFR4K CORECT        11 - CORRECT TIME ACCURACY
    21   31           0
    22   32           0 DEFR4K DATE          12 - DISPLAY CURRENT DATE
    22   33           0
    23   34           0 DEFR4K DATE+         13 - ADD TO CURRENT DATE
    23   35           0
    24   36           0 DEFR4K DDAYS         14 - # DAYS BETWEEN DATES
    24   37           0
    25   40           0 DEFR4K DMY           15 - FORMAT DAY,MONTH,YEAR
    25   41           0
    26   42           0 DEFR4K DOW           16 - DAY OF THE WEEK (0-6)
    26   43           0
    27   44           0 DEFR4K MDY           17 - FORMAT MONTH,DAY,YEAR
    27   45           0
    28   46           0 DEFR4K RCLAF         18 - RCL ACCURACY FACTOR
    28   47           0
    29   50           0 DEFR4K RCLSW         19 - RCL STOPWATCH CONTENT
    29   51           0
    30   52           0 DEFR4K RUNSW         20 - RUN THE STOPWATCH
    30   53           0
    31   54           0 DEFR4K SETAF         21 - SET ACCURACY FACTOR
    31   55           0
    32   56           0 DEFR4K SDATE         22 - SET THE CURRENT DATE
    32   57           0
    33   60           0 DEFR4K SETIME        23 - SET THE CURRENT TIME
```

```
    33   61           0
    34   62           0 DEFR4K SETSW         24 - SET THE STOPWATCH
    34   63           0
    35   64           0 DEFR4K STOPSW        25 - STOP THE STOPWATCH
    35   65           0
    36   66           0 DEFR4K SW            26 - DISPLAY STOPWATCH
    36   67           0
    37   70           0 DEFR4K T+X           27 - ADD TO CURRENT TIME
    37   71           0
    38   72           0 DEFR4K TIME          28 - DISPLAY CURRENT TIME
    38   73           0
    39   74           0 DEFR4K XYZALM        29 - PROGRAMMABLE ALARM FN
    39   75           0
    40   76           0 NOP                  ** END OF FUNCTION TABLE **
    41   77           0 NOP                  ** DON'T PURGE TWO NOPS! **
*******************************************************************************
*
    44  100         203 CON    @203          C
    45  101          40 CON    @40
    46  102          55 CON    @55           -
    47  103           5 CON    @05           E
    48  104          15 CON    @15           M
    49  105          11 CON    @11           I
    50  106          24 CON    @24           T
    51  107          55 CON    @55           -
    52                  ENTRY  HEADER
    53  110 HEADER  253 CON    @253          +               2-3-81 RSW
    54  111           5 CON    @05           E
    55  112          24 CON    @24           T
    56  113           1 CON    @01           A
    57  114           4 CON    @04           D
    58                  ENTRY  DATE+
    59  115 DATE+     1 GOSUB  CHECKX        ERROR IF X= ALPHA DATA
    59  116           0                       *TIMER ROM:  TM0, @0240
    60  117        1204 S7=    0              NOT "DDAYS"
    61  120         210 S5=    1              INTEGER PART
    62  121           1 GOSUB  INTFRC         GET INTEGER PART OF X
    62  122           0                       *MAINFRAME:  CN6, @0473
    63                                        IN: C= NORMALIZED F.P. NUM.
    64                                        ASSUME: S5= 1 FOR INTEGER
    65                                                S5= 0 FOR FRACTION
    66                                                DECIMAL MODE
    67                                        OUT: C= NORMALIZED F.P. NO.
    68                                             A= EXPONENT & SIGN
    69                                             B= 13-DIGIT MANTISSA
    70                                        USES: A,B,C,M,ACT PT, +1SUB
    71                                          (NO ST, NO DADD, NO PFAD)
    72  123        1276 C=-C-1 S              COMPLEMENT SIGN OF # DAYS
    73  124 DT+10   530 M=C                   SAVE INTEGER PART IN M
    74  125         460 LDI                   LOAD LOW 12 BITS OF C WITH
    75  126           2 CON    2              2=USE DATE FROM Y REGISTER
    76  127           1 GOSUB  CHECK          ERROR IF Y = ALPHA DATA
    76  130           0                       *TIMER ROM:  TM0, @0242
    77  131        1610 S0=    1              TAKE DATE FROM Y
    78  132        1410 S1=    1              ADD X OR Y TO DATA ERROR
    79  133           1 GOSUB  YMDDAY         CONVERT IT TO JULIAN DAYS
    79  134           0                       *TIMER ROM:  TM0, @0253
    80  135           1 GOSUB  NDAYS          A= POS NORMAL F.P. # DAYS
    80  136           0                       *TIMER ROM:  TM3, @0011
    81  137         676 A=A-1  S              A= NEGATIVE NUMBER OF DAYS
```

```
    82  140         630 C=M                   GET THE INTEGER BACK TO C
    83  141           1 GOSUB  AD2-10         ADD INTEGER TO JULIAN DAYS
    83  142           0                       *MAINFRAME:  CN6, @0007
*  NO OVERFLOW POSSIBLE SINCE THE DAY NUMBER CALCULATED FROM THE DATE IS
*  AT MOST 999999 WHICH CAN'T CAUSE THE MAXIMUM NUMBER IN X
*  (9.99999999 E99) TO OVERFLOW.
*     NO UNDERFLOW IS POSSIBLE SINCE ONLY THE INTEGER PART OF X IS USED,
*  AND THE DAY NUMBER CALCULATED FROM THE DATE IS AN INTEGER.
    89                                        IN: A & C= NORMALIZED FP ##
    90                                        ASSUME: DECMODE
    91                                        OUT: C= A+-C= NORMALIZED FP
    92                                            A[X&S]= RESULT EXP/SIGN
    93                                            B[12:0]= 13-DIG MANT.
    94                                        USES: A,B,C,M,
    95                                              ACTIVE PT ONLY
    96  143        1214 ?S7=1                 CALLED FROM "DDAYS"?
    97  144         247 GOC    TNFRXY ( 170) YES, OUTPUT NUMBER OF DAYS
    98  145        1376 ? C#0  S              NEGATIVE RESULT?
    99  146          37 GOC    DT+15  ( 151) YES
   100  147          16 A=0                    NO, OUTPUT LOW END OF CAL.
   101  150         133 GOTO   DT+24  ( 163)
   102  151 DT+15   416 A=C                   A= -(DAYS SINCE 10-15-1582)
   103  152         460 LDI                   LOAD LOW 12 BITS OF C WITH
   104  153           5 CON    5              5= 6 DIGITS LEFT OF DEC PT
   105  154           1 GOSUB  UNNORM         UNNORMALIZE
   105  155           0                       *TIMER ROM:  TM3, @0025
   106  156          73 GOTO   DT+25  ( 165) (P+1) OK, A= #DDDDDD0000000
   107  157        1240 SETDEC               (P+2) BEYOND CALENDAR RANGE
   108  160          16 A=0
   109  161         656 A=A-1                 A= DDDDDD= 999999
   110  162           0 NOP
   111  163 DT+24     1 GOSUB  RNGERR         RTN ONLY IF RNG ERR IGNOR=1
   111  164           0                       *TIMER ROM:  TM2, @0464
   112  165 DT+25  1756 A SL
   113  166           1 GOSUB  DAYMD          C= POSITIVE NORMAL FP DATE
   113  167           0                       *TIMER ROM:  TM0, @1642
   114
   115  170 TNFRXY    1 GOLONG NFRXY          NORMAL FUNCTION RETURN XY
   115  171           2                       *MAINFRAME:  CN0, @0332
   116                                        IN: C= NEW VALUE OF X
   117                                             = NORMALIZED F.P. NUM.
   118                                            CHIP 0 ENABLE, P SELECT
   119                                            PERIPHERALS DISABLED
*********************************************************************
   121  172         223 CON    @223           S               2-3-81 RSW
   122  173          31 CON    @31            Y
   123  174           1 CON    @01            A
   124  175           4 CON    @04            D
   125  176           4 CON    @04            D
   126                ENTRY  DDAYS
   127  177 DDAYS  1410 S1=     1             ADD X OR Y TO "DATA ERROR"
   128  200           1 GOSUB  X-YMDD         CONVERT X TO JULIAN DAYS
   128  201           0                       *TIMER ROM:  TM0, @0250
   129  202           1 GOSUB  NDAYS          NORMALIZE NUMBER OF DAYS
   129  203           0                       *TIMER ROM:  TM3, @0011
   130  204        1210 S7=     1             CALLED FROM "DDAYS"
   131  205        1173 GOTO   DT+10  ( 124)
   132
*********************************************************************
   134  206         227 CON    @227           W              1-23-81 RSW
```

```
   135  207          17 CON    @17           O
   136  210           4 CON    @04           D
   137                  ENTRY  DOW
   138  211 DOW    1404 S1=    0             DON'T ADD X TO "DATA ERROR"
   139  212           1 GOSUB  X-YMDD        CONVERT X TO JULIAN DAYS
   139  213           0                      *TIMER ROM:  TM0, @0250
   140  214           1 GOSUB  WKDAYS        CONV. DAYS INTO DAY OF WEEK
   140  215           0                      *TIMER ROM:  TM0, @0576
   141  216           1 GOSUB  RUNSST        RUNNING OR SST?
   141  217           0                      *TIMER ROM:  TM0, @1032
   142  220         143 GOTO   DOW30  ( 234) (P+1) YES, DON'T DISPLAY
   143  221           1 GOSUB  CLLCDE        (P+2) ENABLE & CLR DISPLAY
   143  222           0                      *MAINFRAME: CN11, @0360
   144                                       IN & ASSUME: NOTHING
   145                                       OUT: DSP ENABLE RAM DISABLE
   146                                       USES: C[11:0], ACTIVE PT
   147                                        DADD, PFAD, NO ST, +0 SUB
   148                                        LEVEL, NO ARITHMODE
   149  223         630 C=M                  C= 0D000000000000
   150  224          36 A=0    S
   151  225         576 A=A+1  S             A.S= 1
   152                  LEGAL                (CLEAR THE CARRY FLAG)
   153  226           1 GOSUB  DSWEEK        DISPLAY DOW IN ENGLISH
   153  227           0                      *TIMER ROM:  TM3, @0115
   154  230           1 GOSUB  LEFTJ         LEFT-JUSTIFY DISPLAY
   154  231           0                      *MAINFRAME: CN10, @1767
   155                                       IN & ASSUME: LCD ENABLED
   156                                                    RAM DISABLED
*  !!  DISPLAY MUST NOT CONTAIN ALL BLANKS!! "LEFTJ" WILL NEVER RETURN!!
   158                                       OUT: DISPLAY LEFT-JUSTIFIED
   159                                       USES: A.X,C, ACTIVE PT
   160                                        (NO ST, +0 SUB LEVELS)
   161  232           1 GOSUB  TMSG          PRINT MSG IN NORM & TRACE
   161  233           0                      *TIMER ROM:  TM2, @0431
   162                                       AND SET MESSAGE FLAG
   163  234 DOW30  630 C=M                   C = 0D000000000000
   164  235           1 GOLONG NFRX          NORMAL FUNCTION RETURN X
   164  236           2                      *MAINFRAME:  CN0, @0314
   165                                       IN: CHIP 0 ENABLED, PERIPH
   166                                           DISABLED, P SELECTED
*****************************************************************
* CHECK = CHECKS DATA INPUT FOR LEGALITY                 1-30-81 RSW
*     CHECKS THE C.X SPECIFIED REGISTER FOR LEGAL DATA AND RETURNS ONLY
* FOR NUMERIC DATA.   BOTH THE REGISTER AND THE OUTPUT OF "CHECK" WILL
* BE A FLOATING POINT NUMBER (C.S & C.XS = 0 OR 9) WITH ALL BCD DIGITS.
*
* IN: C.X= REGISTER ADDRESS  !!! THE REGISTER MUST EXIST !!!
*          !!! WILL GET MEMORY LOST IF IT DOES NOT EXIST AND S9= 0 !!!
* ASSUME: PERIPHERALS DISABLED
* OUT:    C= LEGAL NORMALIZED FLOATING POINT NUMBER
*         DECIMAL MODE, THE C.X REGISTER IS ENABLED
* USES:   A,B,C,  ACTIVE PT, DADD, ARITH MODE, +1 SUB LEVEL
*          (NO ST, NO TIMER CHIP ACCESS)
*
   181                  ENTRY  CHKXM
   182                  ENTRY  CHECKX
   183                  ENTRY  CHECK
   184  237 CHKXM  530 M=C
   185  240 CHECKX 460 LDI                   LOAD LOW 12 BITS OF C WITH
   186  241           3 CON    3             ADDRESS OF "X" REGISTER
```

```
   187  242 CHECK  1160 DADD=C
   188  243           1 GOSUB  P6RTN         B= MASSAGED REG CONTENTS
   188  244           0                      *MAINFRAME:  CN5, @1160
*   P6RTN  MASSAGES REGISTER TO CONTAIN EITHER ALPHA DATA OR A LEGAL
*   NORMALIZED FLOATING PT NUM (SIGN & EXP = 0 OR 9, ALL BCD DIGITS)
   191                                       IN: C.X= REGISTER ADDRESS
   192                                           !THIS REG MUST EXIST!
   193                                       ASSUME: C.X REG ENABLED
   194                                           PERIPHERALS DISABLED
   195                                       OUT: B= MASSAGED REGISTER
   196                                            CONTENTS, C.X REG IS
   197                                            ENABLED, HEX MODE
   198                                       USES: A,B,C, ACTIVE PT,
   199                                            ARITH MODE
   200                                        (+0 SUB LEVELS, NO ST)
   201
   202  245         316 C=B
   203  246           1 GOLONG CHK#S         ERROR IF ALPHA DATA
   203  247           2                      *MAINFRAME:  CN5, @0330
   204                                       IN: C= REGISTER CONTENTS
   205                                       ASSUME: NOTHING
   206                                       OUT: C= REG CONTENTS
   207                                        (UNCHANGED), DEC MODE
   208                                       USES: ARITH MODE ONLY
   209
*************************************************************************
* YMDDAY = YEAR, MONTH, DAY TO DAY NUMBER                    1-23-81 RSW
*  TAKES A DATE IN "C" AND COMPUTES THE DAY NUMBER.
*  THEN CALCULATES A DATE FROM THE DAY NUMBER AND COMPARES IT AGAINST
*  A DATE IN "X" OR "Y" AS SPECIFIED BY S0.
*
* IN:  C= NORMALIZED FLOATING POINT DATE
*         !! MUST BE A VALID FLOATING POINT NUMBER !!
* ASSUME:  S0=  1  (0)  --  CHECK THE DATE AGAINST "Y"  ("X")
*          S1=  1  (0)  --  IF ERROR, THE "DATA ERROR" MESSAGE SHOULD
*                           SPECIFY "Y" ("X") REGISTER ERROR
* OUT:  NO ERROR --
*          C= DDDDDD00000000= DAY NUMBER SINCE OCT 15, 1582
*          CHIP 0 ENABLED, HEXMODE
*          A= FLOATING POINT DATE (SAME AS DATE IN X OR Y AS SPECIFIED
*             BY S0)
*       ERROR --       DOES NOT RETURN
* USES: A,B,C,N, R8[13:6], ACTIVE PT, +2 SUB LEVELS, ARITH MODE, DADD,
*       PFAD (NO TIMER CHIP ACCESS)
*
*
* X-YMDD -- TAKES A DATE FROM "X" AND COMPUTES THE DAY NUMBER
*  IN: X= NORMALIZED FLOATING POINT DATE
*      PERIPHERALS DISABLED
*  ASSUME: NOTHING
*  OUT:  SAME AS YMDDAY BUT ALSO: S0= 0
*  USES: SAME AS YMDDAY BUT ALSO: S0, S9
*
   238                   ENTRY  X-YMDD
   239                   ENTRY  YMDDAY
   240  250 X-YMDD      1 GOSUB  CHECKX       ERROR IF ALPHA DATA
   240  251           0                      *TIMER ROM:  TM0, @0240
   241  252        1604 S0=    0             DATE FROM X REGISTER
   242  253 YMDDAY     1 GOSUB  C-YMDD        C REG -- Y,M,D TO DAY#
   242  254           0                      *TIMER ROM:  TM0, @1565
```

```
*
*
*********************************************************************
* DATECK = DATE CHECK                                      1-23-81 RSW
* COMPARES THE DATE IN X OR Y AGAINST A DATE IN "A"
*
* IN & ASSUME:
*      A= FLOATING POINT MORMALIZED DATE
*      CHIP 0 ENABLED
*      S1= 1 (0)     DO (NOT) ADD "X" OR "Y" TO DATA ERROR IF ERR OCCURS
*      S0= 1 (0)     TAKE COMPARISON DATE FROM "Y"  ("X")
*      R8[13:8]= DDDDDD= DAY NUMBER SINCE OCT 15, 1582
* OUT: NO ERROR --
*         C= DDDDDD00000000= DAY NUMBER SINCE OCT 15, 1582
*      ERROR CASE --
*         DOES NOT RETURN
* USES:  C, ACTIVE PT ONLY           (FOR NO ERROR CASE)
*
*
   262                    ENTRY  DATECK
   263  255 DATECK  370 C=REGN 3             C = X REG
   264  256        1614 ?S0=1                USE Y REG?
   265  257          23 GONC   DATCK2 ( 261) NO, USE X
   266  260         270 C=REGN 2
   267  261 DATCK2 1556 ? A#C
   268  262         323 GONC   DATCK4 ( 314) DATE IS OK
*
*
*********************************************************************
* TERROR = TIMER ERROR
*
* IN:  S1= 1 (0)     DO (NOT) ADD "X" OR "Y" TO "DATA ERROR"
*         IF S1=1, THEN:   S0= 1 (0)     ADD "Y" ("X") TO "DATA ERROR"
* ASSUME: NOTHING
*     !!!!! DOES NOT RETURN !!!!!
*
*
* TERRXY --- SAME AS TERROR EXCEPT IGNORES S1 AND ALWAYS ADDS "X" OR "Y"
*            TO "DATA ERROR"
*
*
   284                    ENTRY  TERROR
   285  263 TERROR  460 LDI                  LOAD LOW 12 BITS OF C WITH
   286  264          40 CON    @40           BLANK
   287  265        1414 ?S1=1                ADD X OR Y TO "DATA ERROR"?
   288  266          63 GONC   TERR20 ( 274) NO
   289  267         460 LDI                  LOAD LOW 12 BITS OF C WITH
   290  270          30 CON2   1      8       X
   291  271        1614 ?S0=1                Y REGISTER ERROR?
   292  272          23 GONC   TERR20 ( 274) NO, X REG ERROR
   293  273        1046 C=C+1  X              Y=  19  (HEX)
*********************************************************************
*                                                          1-23-81 RSW
* IN:  C.X= LCD FORMAT CHARACTER TO ADD TO "DATA ERROR"
* ASSUME: NOTHING
*  !! DOES NOT RETURN !!
*
   300                    ENTRY  TERR20
   301  274 TERR20  346 BC EX  X              B.X= CHARACTER TO ADD
   302  275           1 GOSUB  ERRSUB         RTN ONLY IF ERR IGN FLAG=0
```

```
   302  276           0                       *MAINFRAME:  CN8, @1350
   303                                        IN & ASSUME: NOTHING
   304                                        OUT: HEXMODE, CHIP 0 ENABLE
   305                                         S8=1, PERIPHERALS DISABLED
   306                                         FLAGS CLEARED: DATA ENTRY,
   307                                         CATALOG, SHIFT, MESSAGE
   308                                        USES: C, S0-S8, DADD, PFAD,
   309                                         ARITH MODE, +2 SUB LEVELS
   310                                         (NO PT)
   311
   312  277         404 S8=    0              DON'T PRINT YET
   313  300           1 GOSUB  MSGA           SEND "DATA ERROR" TO DISP
   313  301           0                       *MAINFRAME:  CN7, @0154
   314                                        IN: S8= 1 (0) DO (NOT) SET
   315                                            MESSAGE FLAG AND PRINT
   316                                        ASSUME: HEXMODE
   317                                        OUT:  S8=0: CHIP 0 ENABLED
   318                                        USES: S8=0: A.C. ACTIVE PT.
   319                                         +1 SUB, ARITH, DADD, PFAD
   320
   321
   322  302           0 XDEF   MSGDE          "DATA ERROR"
   323  303           1 GOSUB  ENLCD          ENABLE DISPLAY, DISABLE RAM
   323  304           0                       *MAINFRAME:  CN1, @1766
   324                                        IN & ASSUME: NOTHING
   325                                        USES: C.X, DADD, PFAD ONLY
   326
   327  305        1670 FRSABC
   328  306         306 C=B    X
   329  307        1750 SLSABC                 ADD BLANK,X,Y, OR Z TO DISP
   330
   331                  ENTRY  TERR50
   332  310 TERR50    1 GOSUB  TMSG           PRINT AND SET MESSAGE FLAG
   332  311           0                       *TIMER ROM:  TM2, @0431
   333  312           1 GOLONG ERR110         ERROR EXIT
   333  313           2                       *MAINFRAME:  CN8, @1373
   334                                        IN: CHIP 0 ENABLED, HEXMODE
   335                                            STATUS SET 0 UP
   336                                        !!! DOES NOT RETURN !!!
   337  314 DATCK4 1070 C=REGN 8               C= DDDDDD........
   338  315        1234 PT=    7
   339  316         112 C=0    WPT             C= DDDDDD00000000
   340  317        1740 RTN
*******************************************************************************
* IDVD - TWO REGISTER INTEGER DIVIDE                          1-15-81 RSW
*
*   IDVD RETURNS A QUOTIENT AND A REMAINDER SEPARATED BY A ZERO, WITH
* THE POINTER POINTING TO THAT ZERO.
*   C CONTAINS 10'S COMPLEMENT OF DIVISOR WITH AT LEAST ONE LEADING 9.
*  ADDING A 10'S COMPLEMENT (NEGATIVE) NUMBER TO A POSITIVE NUMBER
*  PRODUCES A CARRY EVERY TIME THAT THE RESULT IS STILL POSITIVE, SO
*  "IDVD" BUILDS THE QUOTIENT IN "A" BY ADDING "C" TO "A" AND
*  ACCUMULATING THE CARRIES AS THE QUOTIENT.
*           [ IF N-M >= 0    THEN    N+(10-M) = (N-M)+10 >= 10   ]
*  C.S IS USED AS A LOOP COUNTER ( NUMBER OF DIGITS IN ANSWER ), AND
*  MUST BE LESS THAN I (FOR C.S= 9, ONLY 1 DIVISION LOOP IS DONE).
*
* IN:    1. A.M= DIVIDEND,  POSITIONED ANYWHERE IN A[10:3] WITH ZEROS
*                           IN UNUSED DIGIT POSITIONS
*        (MUST HAVE A[12:11]=0 FOR CARRIES, OR QUOTIENT MAY BE WRONG)
```

```
*        2. C.M= DIVISOR,  (BCD NUMBER) WITH MOST SIGNIFICANT DIGIT
*                (MSD) IN THE SAME POSITION AS MSD OF THE DIVIDEND
*                !!!!! DIVISOR MUST NOT BE ZERO, AND IN GENERAL THE
*                      SIZE OF THE DIVISOR MUST BE KNOWN AS WELL AS
*                      THE SIZE OF THE RESULTING QUOTIENT !!!!!
*        3. C.X= 000
*        4. POINTER= (MSD OF DIVIDEND) + 1
*        5. C.S= ( NUMBER OF QUOTIENT DIGITS IN ANSWER ) - 2
*            !!! WARNING: IF DIVISOR DIGITS GET SHIFTED OFF RIGHT END
*                         OF C REGISTER, SOME ACCURACY WILL BE LOST !!!
*            !!! MUST BE A BCD DIGIT < 9 !!!
*
* ASSUME:  DEC MODE
*
* OUT:   1. "A" CONTAINS THE QUOTIENT WITH ALL UNUSED DIGITS = 0.
*           THE LEFTMOST DIGIT (IT MAY BE A ZERO) OF THE QUOTIENT WILL
*           BE 2 DIGITS TO THE LEFT OF THE MST OF THE INPUT DIVIDEND.
*        2. "C" CONTAINS THE REMAINDER, WITH ALL UNUSED DIGITS= 0.
*           THE LEFTMOST DIGIT (IT MAY BE A ZERO) OF THE REMAINDER IS 2
*           DIGITS TO THE RIGHT OF THE RIGHTMOST QUOTIENT DIGIT.
*           (THERE IS 1 EMPTY DIGIT BETWEEN THE QUOTIENT AND REMAINDER)
*        3. POINTER POINTING TO THE LEFTMOST DIGIT (IT MAY BE A ZERO)
*           OF THE REMAINDER.
*           (THE PT IS DECREMENTED ONCE FOR EACH DIGIT OF THE QUOTIENT)
* USES:  A, C, ACTIVE PT
*        (NO ST, +0 SUB LEVELS, NO DADD, NO PFAD, NO TIMER CHIP ACCESS)
*
* IDVD4 -- SAME AS IDVD EXCEPT C.S= (NUMBER OF QUOTIENT DIGITS) - 4
*
*
   388                  ENTRY  IDVD4
   389                  ENTRY  IDVD
   390  320 IDVD4  1076 C=C+1  S
   391  321        1076 C=C+1  S
   392  322 IDVD   1076 C=C+1  S             INCREMENT LOOP COUNTER
   393  323        1212 C=-C   WPT
   394  324 IDVDL    36 A=0    S
   395  325         516 A=A+C                PERFORM SUBTRACTION
   396  326        1542 ? A#C  PT            OVERFLOW ? (TOO MANY SUBS?)
   397  327        1757 GOC    IDVDL  ( 324) NO, A[PT] # 9
   398  330         716 A=A-C                YES, RECOVER EXTRA SUBTRACT
   399  331        1712 C SR   WPT           DIVIDE BY 10
   400  332        1724 DEC PT
   401  333        1176 C=C-1  S             CARRIES WHEN DONE
   402  334        1703 GONC   IDVDL  ( 324)
   403  335          36 A=0    S
   404  336         116 C=0
   405  337         252 AC EX  WPT           C= REMAINDER
   406  340        1740 RTN
*****************************************************************
* ENTMR= ENABLE TIMER                                    1-6-81 RSW
* DISABLES RAM AND ENABLES TIMER CHIP
*
* IN & ASSUME: NOTHING
* USES:  C.X, DADD, PFAD, TIMER PT
*        (NO 41C PT, NO ST, +0 SUB LEVELS, NO ARITH MODE)
* OUT:   TIMER CHIP ENABLED, RAM DISABLED, TIMER PT=A
*
*
* ENTMRS - SAME AS ENTMR EXCEPT RESTORES S0-S7 FROM C[1:0] BEFORE
```

```
*          DESTROYING C[1:0], SO "ENTMRS" USES S0-S7
*
   420                    ENTRY  ENTMRS
   421                    ENTRY  ENTMR
   422  341 ENTMRS 1530 ST=C
   423  342 ENTMR   460 LDI                  LOAD LOW 12 BITS OF C WITH
   424  343          20 CON2   1      0      NONEXISTENT RAM ADDRESS
   425  344        1160 DADD=C
   426  345         460 LDI                  LOAD LOW 12 BITS OF C WITH
   427  346         373 CON2   15     11     TIMER CHIP ADDRESS = FB
   428  347        1760 PFAD=C               ENABLE TIMER
   429  350        1750 PT=A
   430  351        1740 RTN
*
*****************************************************************
* TIME  - TIME FUNCTION                                 2-3-81 RSW
*         PUT THE CURRENT TIME IN HMS FORM TO X.
*         IF NOT RUNNING, DISPLAY THE TIME AS A MESSAGE.
*****************************************************************
*
   438  352          205 CON    @205          E
   439  353           15 CON    @15           M
   440  354           11 CON    @11           I
   441  355           24 CON    @24           T
   442                    ENTRY  TIME
   443  356 TIME       1 GOSUB  IGDHMS        INIT, GET DAY-HOUR-MIN-SEC
   443  357            0                      *TIMER ROM:  TM2, @1266
   444  360          216 B=A                  A= B= C= DDDDDHHMMSSCC
   445  361         1174 RCR    9             C= .HHMMSSCC.....
   446  362          136 C=0    S
   447  363            1 GOSUB  NORMC         A= C= NORMALIZED F.P. TIME
   447  364            0                      *TIMER ROM:  TM0, @0530
   448  365            1 GOSUB  RUNSST        RUNNING OR SINGLE-STEPPING?
   448  366            0                      *TIMER ROM:  TM0, @1032
   449  367          273 GOTO   DATX37 ( 416) (P+1) YES, DON'T SHOW TIME
   450  370          156 AB EX                (P+2) NO, DISPLAY THE TIME
   451  371            1 GOSUB  DSPTIM        DISPLAY THE TIME
   451  372            0                      *TIMER ROM:  TM3, @0066
   452  373          213 GOTO   DATX30 ( 414)
*
*****************************************************************
* DATE  - DATE FUNCTION                                 2-3-81 RSW
*         PUT THE DATE IN MM.DDYYYY FORM TO X.
*         IF NOT RUNNING, DISPLAY THE DATE AS A MESSAGE.
*
*****************************************************************
*
   461  374          205 CON    @205          E
   462  375           24 CON    @24           T
   463  376            1 CON    @01           A
   464  377            4 CON    @04           D
   465                    ENTRY  DATE
   466  400 DATE       1 GOSUB  IGDHMS        A= DDDDDHHMMSSCC
   466  401            0                      *TIMER ROM:  TM2, @1266
   467  402            1 GOSUB  DAYMDF        A= C= POS NORMAL F.P. DATE
   467  403            0                      *TIMER ROM:  TM0, @1635
   468                                        R8[13:8]= DAY# > 10/15/1582
   469  404            1 GOSUB  RUNSST        RUNNING OR SINGLE STEPPING?
   469  405            0                      *TIMER ROM:  TM0, @1032
   470  406          103 GOTO   DATX37 ( 416) (P+1) YES, DON'T SHOW DATE
```

```
   471  407           1 GOSUB  CLLCDE        (P+2) NO, DISPLAY THE DATE
   471  410           0                      *MAINFRAME: CN11, @0360
   472                                       IN & ASSUME: NOTHING
   473                                       OUT: DISP ENA, RAM DISABLED
   474                                       USES: C[11:0], ACTIVE PT,
   475                                             DADD, PFAD (NO ST, +0
   476                                             SUB LEVEL, NO ARITH)
   477  411        1010 S2=    1             DISPLAY THE YEAR
   478  412           1 GOSUB  DSPDTA        DISPLAY DATA (YEAR)
   478  413           0                      *TIMER ROM:  TM0, @1375
   479  414 DATX30    1 GOSUB  TMSG          PRINT DISP IN NORM & TRACE
   479  415           0                      *TIMER ROM:  TM2, @0431
   480  416 DATX37  630 C=M
   481  417 DATX40  356 BC EX                B= NEW VALUE OF "X"
   482  420           1 GOLONG RCL           MAINFRAME RECALL LOGIC
   482  421           2                      *MAINFRAME:  CN4, @1056
   483                                       IN: B= NEW VALUE OF X
   484                                           (NORMALIZED F.P. FORM)
   485                                       ASSUME: PERIPHERALS DISABLE
   486                                       OUT: NO RETURN, GOTO NFRPR
*
*****************************************************************************
* RCLSW= LOAD THE TIME IN STOPWATCH TO X-REGISTER           2-3-81 RSW
*
* THE TIME WILL BE IN H.M.S FORM AND LESS THAN 99.595999
* THE TIME CAN EITHER BE POSITIVE OR NEGATIVE
*
*****************************************************************************
*
   496  422         227 CON    @227          W
   497  423          23 CON    @23           S
   498  424          14 CON    @14           L
   499  425           3 CON    @03           C
   500  426          22 CON    @22           R
   501                ENTRY  RCLSW
   502  427 RCLSW     1 GOSUB  INITMR        INIT TIMER IF NECESSARY
   502  430           0                      *TIMER ROM:  TM0, @1524
   503  431         404 S8=    0             IGNORE KEYBOARD
   504  432           1 GOSUB  GETMR         GET TIME FROM TIMER
   504  433           0                      *TIMER ROM:  TM0, @0471
   505  434        1633 GOTO   DATX40 ( 417)
*****************************************************************************
* RCLAF= RECALL ACCURACY FACTOR                             1-9-81 RSW
*****************************************************************************
   509  435         206 CON    @206          F
   510  436           1 CON    @01           A
   511  437          14 CON    @14           L
   512  440           3 CON    @03           C
   513  441          22 CON    @22           R
   514                ENTRY  RCLAF
   515  442 RCLAF     1 GOSUB  GETAF         GET ACCURACY FACTOR
   515  443           0                      *TIMER ROM:  TM0, @0445
   516  444        1533 GOTO   DATX40 ( 417) (TIMER DISABLES ON DADD=C)
*****************************************************************************
* GETAF= GET ACCURACY FACTOR                                1-9-81 RSW
* READS THE ACCURACY FACTOR FROM THE TIMER CHIP AND FORMATS IT
*
* IN & ASSUME: NOTHING
* OUT:   C= NORMALIZED FLOATING POINT ACCURACY FACTOR
*        HEXMODE, TIMER PT=B, TIMER CHIP ENABLED, RAM DISABLED
```

```
* USES:  A,C, S0-S7, ACTIVE PT, ARITH MODE, +2 SUB LEVELS, DADD, PFAD,
*        TIMER PT, (NO TIMER ST)
*
   527                    ENTRY  GETAF
   528  445 GETAF     1 GOSUB  INITMR        INIT TIMER IF NECESSARY, PT
   528  446           0                      *TIMER ROM:  TM0, @1524
   529  447         370 RDSTS                C= AF= 000000000SDDD0
   530  450         274 RCR    5             C= SDDD0000000000
   531  451           1 GOSUB  NORM          C= NORMALIZED AF
   531  452           0                      *TIMER ROM:  TM0, @0532
   532  453        1376 ? C#0  S             NEGATIVE?
   533  454        1640 RTN NC               NO
   534  455        1334 PT=    13            YES
   535  456        1120 LC     9             FIX SIGN DIGIT
   536  457        1740 RTN
*************************************************************************
* RUNSW= RUN STOPWATCH                                          1-8-81 RSW
*************************************************************************
   540  460         227 CON    @227          W
   541  461          23 CON    @23           S
   542  462          16 CON    @16           N
   543  463          25 CON    @25           U
   544  464          22 CON    @22           R
   545                    ENTRY  RUNSW
   546  465 RUNSW     1 GOSUB  INITMR        INIT TIMER IF NECESSARY, PT
   546  466           0                      *TIMER ROM:  TM0, @1524
   547  467        1550 STARTC               START THE STOPWATCH
   548  470        1740 RTN                  NOTE: TIMER CHIP AUTOMATIC
   549                                       DISABLE WHEN DADD=C EXEC'D
*
*************************************************************************
* GETMR - GET THE TIME OF TIMER                                 2-2-81 RSW
* IF THE TIME => 100 HOURS, THE OUTPUT WILL BE (TIME)MOD(100) BUT THE
* TIMER TIME WILL NOT BE CLEARED.
*
* IN:     NOTHING
* ASSUME: S8=  1 (0)   TO CHECK (IGNORE) KEYBOARD
*             IF S8=1, THEN:  S9= 1 (0) RETURN ON KEY UP (DOWN)
* OUTPUT: !!! SEE GTMR30 COMMENTS ABOUT GARBAGE OUT OR JUMP TO TMRKEY !!
*         B= UNNORMALIZED TIME=  #HHMMSSCC.....  (#= 0 FOR POSITIVE,
*                                                    9 FOR NEGATIVE)
*         A= C= SIGNED NORMALIZED FLOATING POINT TIME
*         HEXMODE, P SELECTED, Q= 13
*         TIMER CHIP ENABLED, TIMER PT=B
* USES: A,B[13:3],C, P,Q, +1 SUB LEVEL, ARITH MODE, DADD, PFAD, TIMER PT
*                    (NO ST)
*
*
* GETMRC-- IN: C= TIME IN SECS, SO DOESN'T ENABLE OR ACCESS TIMER CHIP
*
*
   572                    ENTRY  GETMR
   573                    ENTRY  GETMRC
   574  471 GETMR     1 GOSUB  ENTMR         ENABLE TIMER
   574  472           0                      *TIMER ROM:  TM0, @0342
   575  473        1650 PT=B                 SELECT STOPWATCH CLOCK
   576  474          70 RDTIME               READ STOPWATCH TIME
   577  475 GETMRC 1240 SETDEC               389 >= MAX EXEC TIME >= 220
   578  476        1376 ? C#0  S             COMPLEMENTED?
   579  477          33 GONC   GTMR15 ( 502) NO
```

```
   580  500         1534 PT=    12            YES, KEEP C.S= 9
   581  501         1212 C=-C   WPT
   582  502 GTMR15   416 A=C
   583  503          236 B=A    S             SAVE THE SIGN
   584  504           36 A=0    S
   585  505            1 GOSUB  36000         C= 00000036000000
   585  506            0                      *TIMER ROM:  TM2, @0454
   586  507          516 A=A+C                YES, RESTORE IT
   587
*                                                              2-3-81 RSW
* GETMR30= CONVERT STOPWATCH TIME TO H.MS NORMALIZED FLOATING POINT FORM
*
*
* IN:  A= #00SSSSSSSSSCC= STOPWATCH TIME (S= SECONDS, C= CENTISECONDS)
* ASSUME: S8= 1 (0)     TO CHECK (IGNORE) KEYBOARD
*             IF S8= 1, THEN:     S9= 1 (0)    RETURN ON KEY UP (DOWN)
*         B.S= SIGN OF THE TIME (0= POSITIVE,  9= NEGATIVE)
* OUT:    IF S8=1 & S9=1 AND THE KEY GOES UP, OUTPUT= GARBAGE
*         IF S8=1 & S9=0 AND A KEY GOES DOWN, JUMP DIRECTLY TO "TMRKEY"
*         OTHERWISE:
*           A= C= SIGNED NORMALIZED FLOATING POINT TIME
*           HEXMODE,  P SELECTED
* USES:   A, B.M, C,  P,Q, +1 SUB LEVEL,  ARITH MODE
*         (NO ST, NO DADD, NO PFAD, NO TIMER CHIP ACCESS)
*
   604                   ENTRY  GTMR30
   605  510 GTMR30    1 GOSUB  KEYCHK        CHECK KEYBOARD IF S8= 1
   605  511            0                     *TIMER ROM:  TM1, @1466
   606  512          106 C=0    X             A= STOPWATCH TIME (100 HRS)
   607  513          534 PT=    6             A= 00SSSSSSSSSCC
   608  514            1 GOSUB  SDHMSK        A= 00000DHHMMSCC
   608  515            0                     *TIMER ROM:  TM2, @1306
   609  516            1 GOSUB  X20Q8         A= (DAYS) * 20
   609  517            0                     *TIMER ROM:  TM3, @0043
   610  520         1716 C SR                 C= (DAYS) * 4
   611  521         1016 C=A+C                C= (DAYS) * 24
   612  522         1534 PT=    12
   613  523         1174 RCR    9             C= 0HHMMSSCC.....
   614  524          352 BC EX  WPT           B= #HHMMSSCC.....
   615  525          316 C=B
   616  526            1 GOSUB  KEYCHK        CHECK KEYBOARD IF S8= 1
   616  527            0                     *TIMER ROM:  TM1, @1466
*  FALLS INTO NORMC!!!!!!!!!!!!!!
   618
   619
*
*********************************************************************
* NORM=    NORMALIZE                                        1-6-81 RSW
*
* INPUT:   C=FLOATING POINT !UNNORMALIZED! NUMBER WITH THE EXPONENT
*            DECREMENTED 1 BELOW ITS CORRECT VALUE (SO THAT TIMES AND
*            DATES CAN BE INPUT WITH C.X= 0)
*               THE EXPONENT MUST BE >= -91 OR "NORM" MAY DECREMENT
*               PAST -99 !!!!!!!
* ASSUME:  NOTHING
* OUT:     HEXMODE
*          A= C= NORMALIZED FLOATING POINT NUMBER
*          THE SIGN OF THE NUMBER IS PRESERVED EXCEPT FOR NEGATIVE ZERO,
*           WHICH IS SET POSITIVE.
* USES:    A, C, ACTIVE PT, ARITH MODE
```

```
*          (NO ST, +0 SUB LVLS, NO DADD, NO PFAD, NO TIMER CHIP ACCESS)
*
*
* NORMC=   SAME AS NORM EXCEPT SETS C[4:0]= 0000
*
*********************************************************************
*
   642                    ENTRY  NORMC
   643  530 NORMC   134 PT=    4
   644  531         112 C=0    WPT
   645
   646                    ENTRY  NORM
   647  532 NORM   1372 ? C#0  M              VALUE= 0?
   648  533          37 GOC    NORM00 ( 536) NO
   649  534         116 C=0                   YES, NORMALIZE IT TO 0
   650  535         133 GOTO   NORM20 ( 550)
   651  536 NORM00 1240 SETDEC
   652  537         416 A=C
   653  540         546 A=A+1  X
   654  541        1534 PT=    12
   655  542          33 GOTO   NORM10 ( 545)
   656  543 NORM05 1772 A SL   M              SHIFT OUT LEADING ZEROS
   657  544         646 A=A-1  X
   658  545 NORM10 1502 ? A#0  PT
   659  546        1753 GONC   NORM05 ( 543)
   660  547         256 AC EX                 C= NORMALIZED NUMBER
   661  550 NORM20  416 A=C
   662  551        1140 SETHEX
   663  552        1740 RTN
*
*
*********************************************************************
*                                                          1-8-81 RSW
* MDY   - SET THE DATE FORMAT TO MONTH.DAY.YEAR  (CLR FLAG.31)
* DMY   - SET THE DATE FORMAT TO DAY.MONTH.YEAR  (SET FLAG.31)
*
* WHEN MAINFRAME FLAG.31 IS SET, ALL I/O
* OF DATE IS ASSUMED DD.MMYYYY FORMAT.  WHEN FLAG.31 IS CLEARED,
* ALL I/O OF DATE IS ASSUMED MM.DDYYYY FORMAT.
*
* IN & ASSUME: CHIP 0 ENABLED, PERIPHERALS DISABLED
*********************************************************************
*
   678  553         231 CON    @231          Y
   679  554           4 CON    @04           D
   680  555          15 CON    @15           M
   681                    ENTRY  MDY
   682  556 MDY    1670 C=REGN 14
   683  557         274 RCR    5
   684  560        1730 CST EX
   685  561         104 S4=    0             CLEAR FLAG 31
   686  562 MDY10  1730 CST EX
   687  563        1174 RCR    9
   688  564        1650 REGN=C 14
   689  565        1740 RTN
*
*    *    *    *    *    *    *    *    *    *    *    *    *    *    *
   692  566         231 CON    @231          Y
   693  567          15 CON    @15           M
   694  570           4 CON    @04           D
```

```
   695
   696  571 DMY    1670 C=REGN 14
   697  572         274 RCR    5
   698  573        1730 CST EX
   699  574         110 S4=    1             SET FLAG 31
   700  575        1653 GOTO   MDY10  ( 562)
*
*********************************************************************
*                                                          1-15-81 RSW
* WKDAYS - CONVERT DAYS SINCE OCT 15, 1582 INTO DAY OF WEEK
*
* INPUT:  C = DDDDDD00000000 = DAYS SINCE OCT 15, 1582
*         WHERE OCT 15, 1582 = 000000
*               SEP 10, 4320 = 999999
* ASSUME: NOTHING
* OUTPUT: C = 0D000000000000 = DAYS SINCE OCT 15, 1582
*         WHERE D IS THE NUMBER OF THE CURRENT DAY OF THE WEEK
*         0 = SUNDAY
*         :                   ( OCT 15, 1582 = FRI = 5 )
*         :
*         6 = SATURDAY
* USES:   A, C, ACTIVE PT, +1 SUB LEVEL, ARITH MODE
*         (NO ST, NO DADD, NO PFAD, NO TIMER CHIP ACCESS)
*
*********************************************************************
*
   721                  ENTRY  WKDAYS
   722  576 WKDAYS  416 A=C
   723  577        1616 A SR
   724  600        1616 A SR                 A= 00DDDDDD000000
   725  601        1240 SETDEC
   726  602         116 C=0
   727  603         534 PT=    6
   728  604         520 LC     5              ADJUST DAY# SO DAY OF WEEK
   729                                         COMES OUT RIGHT
   730  605         516 A=A+C                 MAYBE A= 0 1DDDDDD000 000
*   THE QUOTIENT MAY BE WRONG, BUT ONLY THE REAMAINDER IS USED.
   732  606         116 C=0
   733  607         634 PT=    11
   734  610        1076 C=C+1  S
   735  611        1076 C=C+1  S              6 ITERATIONS IN DIVISION
   736  612         720 LC     7              C= 2 0700000000 000
   737  613        1534 PT=    12
   738  614           1 GOSUB  IDVD4          REMAINDER = DAY-OF-WEEK
   738  615           0                       *TIMER ROM:  TM0, @0320
   739  616        1140 SETHEX               REMAINDER= C[6], REST 0
   740  617         474 RCR    8              C= 0D000000000000
   741  620        1740 RTN
*********************************************************************
* DSAMSG - DISPLAY ALARM MESSAGE                           1-27-81 RSW
*
* THIS ROUTINE GETS THE MESSAGE FROM THE ALARM CATALOG AND DISPLAYS IT.
* IF THE ALARM DOESN'T HAVE A MESSAGE, THE DISPLAY WILL BE CLEARED.
* IF THE MESSAGE IS SHORTER THAN 12 CHARACTERS, THE MESSAGE WILL BE
* DISPLAYED LEFT-JUSTIFIED.
* IF THE MESSAGE IS LONGER THAN 12 CHARS, S8 CAN BE USED AS FOLLOWS:
* S8 = 0 - ONLY DISPLAY THE FIRST 12 CHARACTERS
* S8 = 1 - DISPLAY THE FIRST 12 CHARACTERS AND THEN WAIT FOR KEY UP
*          TO DISPLAY THE REST OF THE MESSAGE.
*
```

```
* IN: M.X= ADDRESS OF CURRENT ALARM (!! MUST BE A VALID ADDRESS
*                                    -- NOT TRAILER REG ADDRESS!!)
*     S8= 0    TO DISPLAY ONLY THE FIRST 12 CHARACTERS
*     S8= 1    TO DISPLAY THE FIRST 12 CHARACTERS AND THEN WAIT FOR
*              KEY UP TO DISPLAY THE REST OF THE MESSAGE
* ASSUME: NO ALARM MESSAGE REGISTERS MAY CONTAIN ALL NULLS (REG= 0)
*         (IT WON'T LOCK UP, BUT THE DISPLAY WILL BE FUNNY)
*         HEXMODE
* OUT:  P SELECTED, Q= 13   (IN ALL CASES)
*      IF RETURN TO (P+1):   (NO MESSAGE)
*          ALARM TIME & DATE REGISTER ENABLED, PERIPHERALS DISABLED
*      IF RETURN TO (P+2):   (MESSAGE DISPLAYED)
*          CHIP 0 ENABLED, PERIPHERALS DISABLED
*          S8= 0 IF S8= 0 ON INPUT
*          B.S = 0
*          B[12] = (NUMBER OF BYTES LEFT IN LAST MSG REGISTER) - 1
*          B[11] = (NUMBER OF MSG REGS NOT YET FULLY DISPLAYED) - 1
*         B[5:3] = ADDRESS OF CURRENT MESSAGE REGISTER
*              N = CURRENT MSG REG WITH NEXT CHARACTER IN N[13:12]
* USES: A, B, C, N,  P, Q,  S3, +1 SUB LEVEL, ARITH MODE, DADD, PFAD
*       IF THE MESSAGE >= 12 CHARS AND S8=1 ON INPUT, THEN SETS S8=0
*              (NO TIMER CHIP ACCESS)
*******
*
* DSA2ND - ENTRY POINT TO DISPLAY THE 2ND PART OF THE ALARM MESSAGE
*          !! MUST NOT BE USED TO DISPLAY THE FIRST HALF, SINCE IT
*          DOESN'T SKIP LEADING NULLS !!
* IN:      B.S = 0
*          B[12] = (NUMBER OF BYTES LEFT IN CURRENT MESSAGE REG) - 1
*          B[11] = (NUMBER OF MSG REGS NOT YET FULLY DISPLAYED) - 1
*         B[5:3] = ADDRESS OF CURRENT MESSAGE REGISTER
*              N = CURRENT MSG REG WITH NEXT CHAR IN N[13:12]
* ASSUME: P SELECTED, Q= 13, HEXMODE
* OUT:    IF RETURN TO (P+1): NO MORE CHARACTERS
*              DISPLAY UNCHANGED, CHIP ENABLE UNCHANGED
*              (USES ONLY THE ACTIVE PT)
*         IF RETURN TO (P+2): MORE CHARACTERS IN MESSAGE
*              CHIP 0 ENABLED, PERIPHERALS DISABLED,  S3= S8= 0
* USES:   SAME AS DSAMSG EXCEPT ALWAYS USES S8
*************************************************************************
*
   795                  ENTRY  DSAMS0
   796                  ENTRY  DSAMSG
   797  621 DSAMS0  404 S8=    0
   798  622 DSAMSG   10 S3=    1             SET "FIRST REGISTER" FLAG
   799  623         340 SEL Q
   800  624        1334 PT=    13
   801  625         240 SEL P
   802  626         106 C=0    X
   803  627        1760 PFAD=C               DISABLE ANY PERIIPHERAL
   804  630         630 C=M                  C.X= ADDR OF CURRENT ALARM
   805  631         406 A=C    X
   806  632        1160 DADD=C
   807  633          70 C=DATA               LOAD CURRENT ALARM
   808  634        1366 ? C#0  XS            ALARM HAS RESET INTERVAL?
   809  635          23 GONC   AMG10  ( 637) NO
   810  636         546 A=A+1  X
   811  637 AMG10   546 A=A+1  X             A.X= 1ST MESSAGE REGISTER
   812  640        1474 RCR    1             C.X= MESSAGE LENGTH
   813  641        1176 C=C-1  S             C.S= # OF MSG REG - 1
```

```
   814  642         1540 RTN C                NO MESSAGE
   815  643          274 RCR    5             C= X XXXXRXXXXX XXX
   816  644          246 AC EX  X             (R= MSG REG COUNTER)
   817  645          674 RCR    11            C= X XRXXXXXAAA XXX
* USE A.M RATHER THAN A.X FOR MSG REG ADDR   (A= MSG REG ADDRESS)
* SINCE PIL PRTR "PRTLCD" ROUTINE USES B.X (WHERE A.X WOULD BE STORED)
   820  646         1334 PT=    13
   821  647         1420 LC     12            C.S= # UNUSED LCD CHAR POS
   822  650          416 A=C                  A[13]= # UNUSED LCD CHARS
   823                                        A[11]= MSG LENGTH (IN REGS)
   824                                        A[5:3]= MSG REG ADDRESS
   825  651            1 GOSUB  CLLCDE        ENABLE & CLEAR DISPLAY
   825  652            0                      *MAINFRAME: CN11, @0360
   826                                        IN & ASSUME: NOTHING
   827                                        OUT: DISP ENA, RAM DISABLED
   828                                        USES: C[11:0], ACTIVE PT,
   829                                        DADD, PFAD (NO ST, +0 SUB
   830                                        LEVELS, NO ARITHMODE)
   831  653 DSAM20   256 AC EX                C= COUNTERS
   832  654         1534 PT=    12
   833  655          620 LC     6
   834  656          416 A=C                  A[12]= BYTE CTR IN REG= 6
   835                                           ( 7 BYTES/REGISTER )
   836  657          106 C=0    X             C= X 6RXXXXXAAA 000
   837  660         1760 PFAD=C               DISABLE DISPLAY
   838  661           74 RCR    3             C.X= MSG REGISTER ADDRESS
   839  662         1160 DADD=C
   840  663           70 C=DATA               LOAD ONE MESSAGE REGISTER
   841  664           14 ?S3=1                FIRST REGISTER OF MESSAGE?
   842  665          103 GONC   DSAM30 ( 675) NO, DON'T SKIP EMB'D NULLS
   843  666         1534 PT=    12            YES, SKIP LEADING NULLS
   844  667            4 S3=    0             CLEAR FIRST REGISTER FLAG
   845  670 DSAM25 1362 ? C#0   PQ            ANY LEADING NULL ?
   846  671           47 GOC    DSAM30 ( 675) NO
   847  672         1574 RCR    12            SKIP LEADING NULLS
   848  673          642 A=A-1  PT            DECREMENT BYTE COUNTER
   849  674         1743 GONC   DSAM25 ( 670)
*   !!  ASSUME THAT THE REGISTER WILL NOT BE ALL NULLS !!!!!!!
*       THERE MUST BE AT LEAST 1 NON-NULL CHARACTER!
   852  675 DSAM30   160 N=C                  SAVE MESSAGE REGSTER IN N
   853  676            1 GOSUB  ENLCD         ENABLE LCD, DISABLE RAM
   853  677            0                      *MAINFRAME:  CN1, @1766
   854                                        IN:  NOTHING
   855                                        ASSUME: NOTHING
   856                                        OUT: DISP ENA, RAM DISABLED
   857                                        USES: C.X, DADD, PFAD ONLY
   858  700 DSAM35   216 B=A                  SAVE ALL COUNTERS IN B
   859                                        B.S= UNUSED LCD CHAR COUNT
   860                                        B[12] = REG BYTE COUNTER
   861                                        B[11] = MSG REG. COUNTER
   862                                        B[5:3] = MSG REG ADDRESS
   863  701          260 C=N
   864  702         1574 RCR    12            C[1:0]= NEXT ASCII CHAR
   865  703          160 N=C
   866  704            1 GOSUB  ASCLCD        SEND C[1:0] CHAR TO DISPLAY
   866  705            0                      *MAINFRAME: CN11, @0135
   867                                        IN: C[1:0]= ASCII CHARACTER
   868                                            B.S= # UNUSED LCD CHARS
   869                                        ASSUME: DISP ENA, HEXMODE
   870                                                RAM DISABLED
```

```
                                                  OUT: IF B.S= 0, DON'T DECR.
                                                       IF B.S# 0, DECREMENT
                                                       WHENEVER ANOTHER LCD
                                                       CHAR POSITION IS USED.
                                                       C.X= LCD CHAR CODE
                                                  USES: A.X&S, B.S, C, ACT PT
                                                       (NO ST, +0 SUB LEVELS)

*  NOTE: A "NOP" FOLLOWING "GOSUB ASCLCD" WILL CAUSE ANY CHAR EXCEPT
*        PUNCTUATION NOT TO BE SENT TO THE DISPLAY.

  706 DSAM37   156 A=B                    GET COUNTERS BACK TO A
  707          216                        (INSERTED BY ASSEMBLER)
  710         1536 ? A#0   S              DISPLAY FULL ?
  711          113 GONC    DSAM50 ( 722)  YES
  712         1534 PT=     12
  713          642 A=A-1   PT             END OF ONE REGISTER ?
  714         1643 GONC    DSAM35 ( 700)  NO
  715 DSAM45   634 PT=     11
  716          642 A=A-1   PT             ALL MSG REGS FINISHED ?
  717          257 GOC     DSAM60 ( 744)  YES, ALL CHARACTERS OUT
  720          572 A=A+1   M              POINT TO NEXT MSG REG
                   LEGAL                  (CLEAR THE CARRY FLAG)
  721         1323 GOTO    DSAM20 ( 653)
  722 DSAM50   414 ?S8=1                  DISPLAY REST OF MSG ?
  723          343 GONC    DSAM80 ( 757)  NO, (DSP FULL, NO L-JUST)
  724            1 GOSUB   RSTKB          WAIT FOR KEY UP (DEBOUNCE)
  725            0                        *MAINFRAME:  CN0, @0230
                                          IN: HEXMODE
                                          ASSUME: HEXMODE
                                          USES: C.X ONLY
  726          634 PT=     11
  727         1322 ? B#0   PQ             ANY MORE CHARACTERS ?
  730          273 GONC    DSAM80 ( 757)  NO

                   ENTRY   DSA2ND
  731 DSA2ND   634 PT=     11
  732         1322 ? B#0   PQ             ANY MORE MSG CHARACTERS ?
  733         1640 RTN NC                 NO, DONE
  734            1 GOSUB   CLLCDE         ENABLE & CLEAR DISPLAY
  735            0                        *MAINFRAME: CN11, @0360
                                          IN & ASSUME: NOTHING
                                          OUT: DISP ENA, RAM DISABLED
                                          USES: C[11:0], ACTIVE PT,
                                                PFAD, DADD (NO ST,
                                                +0 SUB, NO ARITHMODE)
  736         1334 PT=     13
  737         1420 LC      12
  740          376 BC EX   S              B.S= 12= WHOLE DISP EMPTY
  741          404 S8=     0              REMEMBER 2ND HALF OF MSG
  742            4 S3=     0              CLEAR "FIRST REGISTER" FLAG
  743         1433 GOTO    DSAM37 ( 706)  DISPLAY REST OF THE MESSAGE
*  NOW LEFT JUSTIFY THE DISPLAY, ALLOWING IT TO BE ALL BLANKS
  744 DSAM60   460 LDI                    LOAD LOW 12 BITS OF C WITH
  745         1013 CON     @1013          @1013 = 001000001011 BINARY
  746         1474 RCR     1
  747          416 A=C                    A.X= @40= BLANK, A.S= 11
  750         1434 PT=     1
  751 DSAM65  1770 RABCL                  FETCH LEFTMOST DISPLAY CHAR
  752         1552 ? A#C   WPT            BLANK?
```

```
   928  753           37 GOC    DSAM70 ( 756) NO, DONE
   929  754          676 A=A-1  S              DISPLAY= ALL BLANKS ?
   930  755         1743 GONC   DSAM65 ( 751) MAYBE NOT, KEEP LOOKING
   931  756 DSAM70 1670 RABCR                 RESTORE THE DISPLAY
   932  757 DSAM80    1 GOSUB  ENCP00         ENABLE CHIP 0, DISABLE RAM
   932  760           0                       *MAINFRAME:  CN2, @0522
   933                                        IN: NOTHING
   934                                        ASSUME: NOTHING
   935                                        OUT: C.X= 0
   936                                        USES: C.X, DADD, PFAD ONLY
   937  761          76 B=0    S
   938  762 DSAM90    1 GOLONG RTNP+2         RETURN TO (P+2)
   938  763           2                       *TIMER ROM:  TM2, @1230
*
*********************************************************************
* PUGALM - PURGE AN ALARM                                  12-15-80 RSW
* INPUT : M.X= ADDRESS OF ALARM TO BE PURGED
*    !!!!! THIS MUST BE A VALID ALARM REG ADDRESS, NOT THE ADDRESS OF
*    !!!!! THE TRAILER REG (OR RESET INTERVAL OR MESSAGE REGISTER)
* ASSUME: HEXMODE
* OUT:    P SELECTED, Q= 13, PERIPHERALS DISABLED
*        IF THERE ARE STILL 1 OR MORE ALARMS LEFT, RETURN TO P+2
*           IF THE PURGED ALARM WAS THE HIGHEST ADDRESSED ALARM IN
*           THE ALARM STACK, THEN M.X WILL POINT TO THE NEW HIGHEST
*           ADDRESSED ALARM IN THE STACK, AND A.S= 0
*
*           OTHERWISE, M.X WILL BE UNCHANGED (SO IT WILL POINT TO WHAT
*           WAS THE NEXT HIGHEST ADDRESSED ALARM IN THE STACK) & A.S= F
*
*        IF THE ALARM PURGED WAS THE ONLY ALARM, RETURN TO P+1
*           THE WHOLE BUFFER WILL BE PURGED.
*
* USES:  A.B[X&S], C,N, MAY UPDATE M.X, P,Q, S3,S8, +2 SUB LVLS,
*        DADD, PFAD (NO TIMER CHIP ACCESS)
*
*********************************************************************
*
   963                 ENTRY  PUGALM
   964  764 PUGALM    1 GOSUB  SRHBUF         A.X=  BEGINNING OF BUFFER
   964  765           0                       *TIMER ROM:  TM2, @1141
   965                                        (ALSO SETS Q= 13)
   966                                        (P+1)
   967                                        (P+2) ASSUME NO ERROR POSS.
   968  766           1 GOSUB  GETM.X         C= ALARM REG
   968  767           0                       *TIMER ROM:  TM1, @0532
   969  770         160 N=C                   SET N= ALARM REGISTER
   970  771          10 S3=    1              PURGING THE ALARM
   971  772           1 GOSUB  CHKBUF         FIND END OF I/O BUFFER AREA
   971  773           0                       *TIMER ROM:  TM2, @1052
   972                                          (+1 SUB LEVEL)
   973  774         346 BC EX  X              B.X= LAST REG OF I/O BUFFER
   974  775         406 A=C    X              A= ADR OF TIMER BUF HDR REG
   975  776        1160 DADD=C
   976  777          70 C=DATA
   977 1000        1434 PT=    1
   978 1001         374 RCR    10
   979 1002        1152 C=C-1  WPT            CHECK FOR EMPTY BUF (2 REG)
   980 1003        1152 C=C-1  WPT
   981 1004          36 A=0    S
   982 1005        1152 C=C-1  WPT
```

```
   983 1006          63 GONC   PUGA30 (1014) DON'T PURGE THE BUFFER
   984 1007         576 A=A+1  S              PURGE THE HEADER AND
   985 1010         576 A=A+1  S               TRAILER REGISTERS ALSO
   986 1011         630 C=M
   987 1012         246 AC EX  X
   988 1013         530 M=C                   M.X= ADDR OF HEADER REG
   989 1014 PUGA30  260 C=N                   C= ALARM TO BE PURGED
   990 1015           1 GOSUB  SHFTDN         SHIFT REGISTER BLOCK DOWN
   990 1016           0                       *TIMER ROM:  TM0, @1140
*  NOTE: THE MAXIMUM TIME TO SHIFT THE I/O BUFFERS TO PURGE AN ALARM IS
*        ABOUT 3.6 SECS, ASSUMING A 6-REG ALARM SHIFTED 319 REGISTERS.
   993 1017           1 GOSUB  SRHBUF         A.X=BUF START ADDR, A.S= 0
   993 1020           0                       *TIMER ROM:  TM2, @1141
   994 1021          23 GOTO   PUGA40 (1023) (P+1) STILL ALARMS LEFT
   995 1022        1740 RTN                  (P+2) BUFFER PURGED
   996 1023 PUGA40    1 GOSUB  GETM.X         C= NEXT ALARM
   996 1024           0                       *TIMER ROM:  TM1, @0532
   997 1025         676 A=A-1  S              OUTPUT A.S= F IF THIS WAS
   998                                          NOT LAST ALARM IN STACK
   999 1026        1076 C=C+1  S              IS IT TRAILER REGISTER ?
  1000 1027           1 GSUBC  ALMBST         YES, BACKSTEP TO PREV ALARM
  1000 1030           1                       *TIMER ROM:  TM2, @0400
  1001 1031 PUGA50 1313 GOTO   DSAM90 ( 762)
*******************************************************************************
* RUNSST = RUNNING OR SINGLE STEPPING                          1-9-81 RSW
*
* IN:     DATA IN "C" WHICH IS SAVED IN "M"
* ASSUME: NOTHING
* OUT:    CHIP 0 ENABLED, STATUS SET 0 UP, PERIPHERALS DISABLED.
*         M= COPY OF "C" AT INPUT
*         RETURN TO --
*         P+1:  RUNNING OR SINGLE STEPPING
*         P+2:  NOT RUNNING, NOT SST (HEXMODE SET)
* USES:   C,M, S0-S7, DADD, PFAD, +1 SUB LEVEL, ARITH MODE IF RTN TO P+2
*         (NO PT, NO TIMER CHIP ACCESS)
*
  1015                   ENTRY  RUNSST
  1016 1032 RUNSST  530 M=C                  ("M=C" SAVES CODE - IT
  1017                                         ISN'T USED IN "RUNSST")
  1018 1033           1 GOSUB  LDSST0        LOAD STATUS SET 0
  1018 1034           0                      *MAINFRAME:  CN1, @1627
  1019                                       IN & ASSUME: NOTHING
  1020                                       OUT: S0-S7= STATUS SET 0
  1021                                         C=REG 14, CHIP 0 ENABLED
  1022                                         PERIPHERALS DISABLED
  1023                                       USES: C, S0-S7, DADD, PFAD
  1024                                        (NO PT, +0 SUB LEVELS,
  1025                                         NO ARITH MODE)
  1026 1035        1314 ?S13=1               RUNNING ?
  1027 1036        1540 RTN C                YES
  1028 1037         114 ?S4=1                SINGLE STEPPING ?
  1029 1040        1540 RTN C                YES
  1030 1041        1703 GOTO   PUGA50 (1031) NO
*
*******************************************************************************
* RSTALM - RESET AN ALARM BY ITS AUTO RESET INTERVAL.          1-14-81 RSW
*   TRIES TO RESET THE ALARM TO ITS NEXT FUTURE OCCURRENCE, BUT DUE TO
*   THE TIME TAKEN TO SHIFT THE ALARM TO ITS NEW LOCATION IN THE ALARM
*   STACK AND THE TIME TO DO OTHER OVERHEAD AFTER CALLING "RSTALM", THE
*   ALARM MAY BE SET TO THE PAST!!!!!
```

```
*
* INPUT:  M.X= ADDRESS OF CURRENT ALARM (MUST BE A VALID ALARM ADDR!!)
*         HEXMODE
*         RESET INTERVAL EXPONENT FIELD = 000 WITH INTERVAL STORED IN
*            THIS FORMAT:      0 0SSSSSSSST 000
*            WHERE "T" = TENTHS OF SECONDS
* ASSUME: PERIPHERALS DISABLED
* OUT:    (P+1): ALARM HAS RESET INTERVAL
*         C.X= NEW ADDRESS OF CURRENT ALARM (REST OF C IS COPY OF M)
*         P SELECTED, Q= 13, HEXMODE, TIMER PT=A
*         (P+2): NO RESET INTERVAL  (USES ONLY: C, DADD, ARITH MODE)
*         HEXMODE
* USES:   A,B,C,N, P,Q, S3, +2 SUB LVL, DADD, PFAD, ARITH MODE, TIMER PT
*
*********************************************************************
*
  1054                    ENTRY  RSTALM
  1055 1042 RSTALM    1 GOSUB  GETM.X        C= ALARM TIME & INFO
  1055 1043             0                    *TIMER ROM:  TM1, @0532
  1056 1044          1366 ? C#0  XS            ALARM HAS RESET INTERVAL ?
  1057 1045          1643 GONC   PUGA50 (1031) NO, RETURN TO P+2
  1058 1046           340 SEL Q
  1059 1047          1334 PT=    13
  1060 1050           240 SEL P
  1061 1051            34 PT=    3
  1062 1052           356 BC EX               B= ALARM TIME & INFO
  1063 1053           630 C=M
  1064 1054          1046 C=C+1  X            C.X= ADDR OF RESET INTERVAL
  1065 1055          1160 DADD=C
  1066 1056            70 C=DATA              C= 0 0SSSSSSSST 000= RESET
  1067                                        INTERVAL (T=TENTHS OF SECS)
  1068 1057           416 A=C
  1069 1060          1240 SETDEC
  1070 1061 RSTA05 1762 A SL   PQ            LEFT-JUSTIFY RESET INTERVAL
  1071 1062           646 A=A-1  X
  1072 1063          1536 ? A#0  S
  1073 1064          1753 GONC   RSTA05 (1061)
  1074 1065           646 A=A-1  X
  1075 1066           162 AB EX  PQ           A[13:3]= ALARM TIME
  1076                                        B[13:3]= LEFT-JUST RST INTV
  1077 1067             1 GOSUB  ENTMR        ENA TIMER, DIS RAM, PT=A
  1077 1070             0                    *TIMER ROM:  TM0, @0342
  1078 1071 RSTA10    70 RDTIME              C= CUR TIME= 00SSSSSSSSSCC
  1079                                        TIMER PT=A
  1080 1072          1374 RCR    13           C= 0SSSSSSSSSCC0
  1081 1073          1062 C=C+1  PQ
  1082 1074          1062 C=C+1  PQ
  1083 1075          1062 C=C+1  PQ           MAKE TIME 3 SECS IN FUTURE
  1084 1076          1374 RCR    13           C= SSSSSSSSSCC00
  1085 1077 RSTA15   462 A=A+B   PQ           ADD RESET INTERVAL TO ALARM
  1086 1100            37 GOC    RSTA17 (1103) OVERFLOW
  1087 1101          1422 ? A<C  PQ           ALARM STILL IN THE PAST ?
  1088 1102          1757 GOC    RSTA15 (1077) YES
  1089 1103 RSTA17   546 A=A+1   X            DONE ?
  1090 1104            47 GOC    RSTA20 (1110) YES
  1091 1105           622 A=A-B   PQ           REMOVE EXTRA ADD
  1092 1106          1662 B SR    PQ           SHIFT RESET INTERVAL
  1093 1107          1623 GOTO    RSTA10 (1071)
  1094 1110 RSTA20   222 B=A     PQ            B= NEW ALARM TIME & INFO
  1095 1111          1140 SETHEX
```

```
  1096 1112          630 C=M                   C.X= ALARM ADDRESS
  1097 1113            1 GOSUB  NEWLSK         A.X= ADR OF 1ST ALM>NEW ALM
  1097 1114            0                       *TIMER ROM:  TM2, @1107
  1098                                           OR A.X= TRAILER REG ADDR.
  1099 1115          646 A=A-1  X
  1100 1116            4 S3=    0              NOT PURGING THE ALARM
  1101 1117          630 C=M                   C.X= ADDR OF CURRENT ALARM
  1102 1120         1160 DADD=C
  1103 1121          316 C=B                   C= NEW ALARM TIME & INFO
  1104 1122         1434 PT=    1
  1105 1123          102 C=0    PT             UNMARK THE ALARM
  1106 1124         1360 DATA=C                UPDATE ALARM TIME
  1107 1125          206 B=A    X              B.X=ADRRESS OF HIGHEST
  1108 1126           36 A=0    S               REGISTER TO BE SHIFTED
  1109 1127            1 GOSUB  SHFTDN         SHIFT ALARM TO NEW LOCATION
  1109 1130            0                       *TIMER ROM:  TM0, @1140
*   NOTE: THE MAXIMUM TIME TO SHIFT A RESET ALARM TO ITS NEW LOCATION IS
*         ABOUT 2.9 SECONDS, ASSUMING A 6-REGISTER ALARM SHIFTED 253
*         REGISTERS.
  1113 1131          176 AB EX  S              A.S= (# ALARM REG) - 1
  1114 1132          630 C=M
  1115 1133          306 C=B    X              C.X= LST REG OF SHIFTED ALM
  1116 1134 RSTA50   676 A=A-1  S
  1117 1135         1540 RTN C
  1118 1136         1146 C=C-1  X              C.X= 1ST REG OF SHIFTED ALM
  1119                   LEGAL                 (CLEAR THE CARRY FLAG)
  1120 1137         1753 GOTO   RSTA50 (1134)
*
*******************************************************************
*                                                        12-15-80 RSW
* SHFTDN - SHIFT A BLOCK OF REGS DOWN, ROTATE BOTTOM ALARM TO TOP
* INPUT:  B.X= ADDR OF HIGHEST ADDRESSED REG TO BE SHIFTED DOWN
*         [TO PURGE AN ALARM, THIS MUST BE THE LAST REGISTER OF THE LAST
*         (HIGHEST ADDRESSED) I/O BUFFER]
*         C.X= EXPONENT FIELD OF ALARM TIME & DATE REGISTER (ALARM INFO)
*         M.X= ADDR OF CURRENT ALM (LOWEST ADDRESSED REG TO BE SHIFTED)
*         A.S= 0 UNLESS PURGING THE WHOLE BUFFER, WHEN A.S= 2
*         S3=1:  CALLED FROM PUGALM (PURGES CURRENT ALARM)
*         S3=0:  CALLED FROM RSTALM (ROTATES CURRENT ALARM TO TOP OF
*                                    THE SHIFTED AREA)
* ASSUME: PERIPHERALS DISABLED, HEXMODE
* OUT:    B.X PRESERVED, S3 PRESERVED
*         B.S= (NUMBER OF REGISTERS IN THE CURRENT ALARM) - 1
* USES:   A[X&S], B.S, C,N, DADD
*           (NO PT, NO TIMER CHIP ACCESS, +0 SUB LEVELS, NO PFAD)
*******************************************************************
*
  1141                   ENTRY  SHFTDN
  1142 1140 SHFTDN  1366 ? C#0  XS             RESET INTERVAL ?
  1143 1141           23 GONC   SHFTD2 (1143) NO
  1144 1142         1046 C=C+1  X
  1145 1143 SHFTD2  1474 RCR    1              C.X= # REGS (MSG+AUTO INC)
  1146 1144          536 A=A+C  S
  1147 1145          236 B=A    S
  1148 1146 SHFTD4   630 C=M
  1149 1147         1160 DADD=C
  1150 1150           70 C=DATA
  1151 1151           14 ?S3=1                 PURGING ALARM ?
  1152 1152           23 GONC   SHFTD6 (1154) NO
  1153 1153          116 C=0                   YES, PURGE CURRENT ALARM
```

```
  1154 1154 SHFTD6  160 N=C                  N= REGISTER TO BE ROTATED
  1155 1155         630 C=M
  1156 1156         123 GOTO   SHFTD8 (1170)
  1157 1157 SHFTD7 1046 C=C+1  X
  1158 1160        1160 DADD=C
  1159 1161          70 C=DATA
  1160 1162         246 AC EX  X              A.X= DATA, C.X= ADDRESS
  1161 1163        1160 DADD=C
  1162 1164         246 AC EX  X              A.X= ADDR, C.X= DATA
  1163 1165        1360 DATA=C
  1164 1166         546 A=A+1  X
  1165 1167         246 AC EX  X
  1166 1170 SHFTD8  406 A=C    X
  1167 1171        1446 ? A<B  X              MORE REGISTERS TO SHIFT ?
  1168 1172        1657 GOC    SHFTD7 (1157) YES
  1169 1173         306 C=B    X
  1170 1174        1160 DADD=C
  1171 1175         260 C=N
  1172 1176        1360 DATA=C
  1173 1177         676 A=A-1  S
  1174 1200        1463 GONC   SHFTD4 (1146)
  1175 1201        1740 RTN
*
*************************************************************************
* FNDMSG = FIND MESSAGE                                        1-9-81 RSW
* FINDS THE FIRST NON-NULL REGISTER IN THE ALPHA REGISTER
*
* IN & ASSUME: PERIPHERALS DISABLED, HEXMODE
* OUT:       IF THE ALPHA REGISTER IS NOT EMPTY--
*               C= FIRST NON-NULL REGISTER (THAT REGISTER IS ENABLED)
*               A.X= ADDRESS OF THE FIRST NON-NULL REGISTER
*               A.S= (MESSAGE LENGTH IN REGISTERS) - 1
*
*            IF THE ALPHA REGISTER IS EMPTY--
*               C= 0
*               A.X= 4
*               A.S= F
* USES:      A[X&S], C, ACTIVE PT, DADD
*             (NO ST, +0 SUB LEVELS, NO PFAD, NO TIMER CHIP ACCESS)
*************************************************************************
*
  1195                    ENTRY  FNDMSG
  1196 1202 FNDMSG  460 LDI                  LOAD LOW 12 BITS OF C WITH
  1197 1203          10 CON    8             ADDRESS OF FIRST ALPHA REG
  1198 1204        1160 DADD=C               ENABLE REGISTER 8
  1199 1205         246 AC EX  X             A.X= 8 = REGISTER ADDRESS
*   NOTE:  COULD SAVE A STATE BY SETTING C.S=3 AND DOING "AC EX   W" AT
*          THE EXPENSE OF DESTROYING A.M
  1202 1206        1334 PT=    13
  1203 1207         320 LC     3
  1204 1210        1234 PT=    7
  1205 1211         436 A=C    S             A.S= C.S= 3
  1206 1212          70 C=DATA
  1207 1213         574 RCR    6
  1208 1214         112 C=0    WPT           CLEAR REG 8 SCRATCH AREA
  1209 1215         474 RCR    8
  1210 1216 FNDM10 1356 ? C#0
  1211 1217        1540 RTN C
  1212 1220         646 A=A-1  X             DECREMENT REGISTER ADDRESS
  1213 1221         676 A=A-1  S             ALPHA REGISTER EMPTY ?
```

```
  1214 1222         1540 RTN C                YES
  1215 1223          246 C=A    X
  1215 1224          406                      (INSERTED BY ASSEMBLER)
  1216 1225         1160 DADD=C
  1217 1226           70 C=DATA
  1218 1227         1673 GOTO   FNDM10 (1216)
*
*********************************************************************
*                                                          1-12-81 RSW
* CHKALM - CHECK ALARM STACK TO SEE IF THERE ARE ANY PAST DUE ALARMS
*          AND UNMARK ALL FUTURE ALARMS
*
* IN:     A.X= ADDR OF FIRST REGISTER (HEADER REG) IN THE TIMER BUFFER
* ASSUME: HEXMODE, P SELECTED, Q= 13
* OUT:    TIMER PT=A    (IN ALL CASES)
*         B.XS= ZERO    (NON-ZERO)  ---  THERE ARE (NO) PAST DUE ALARMS
*         B[1:0]= ZERO  (NON-ZERO)  ---  THERE ARE (NO) UNDISPLAYED
*                                          PAST DUE ALARMS
* NOTE:   CHKALM READS THE TIME ONCE AT THE BEGINNING AND USES THAT
*         TIME AS THE REFERENCE FOR PAST DUE VERSUS FUTURE ALARMS.
*
* USES:   A, B.X, C, ACTIVE PT, +1 SUB LEVEL, DADD, PFAD
*         (NO ST)
*
*********************************************************************
*
  1239                    ENTRY  CHKALM
  1240 1230 CHKALM     1 GOSUB  ENTMR          ENA TMR, DISABLE RAM, PT=A
  1240 1231             0                      *TIMER ROM:  TM0, @0342
  1241 1232            70 RDTIME               C= CURRENT TIME
  1242 1233          1574 RCR    12            C= SSSSSSSSSCC00
  1243 1234           106 C=0    X
  1244 1235          1146 C=C-1  X
  1245 1236           346 BC EX  X             B.X SET NON-ZERO
  1246 1237           246 AC EX  X
  1247 1240          1046 C=C+1  X             C.X= 1ST ALARM ADDRESS
  1248                                          (OR MAYBE TRAILER REG ADR)
  1249 1241           416 A=C
  1250 1242           113 GOTO   CHKA35 (1253)
  1251 1243 CHKA20     66 B=0    XS            AT LEAST 1 PAST DUE ALARM
  1252 1244          1434 PT=    1
  1253 1245          1342 ? C#0  PT            THIS ALARM BEEN DISPLAYED ?
  1254 1246            27 GOC    CHKA25 (1250) YES
  1255 1247            52 B=0    WPT           B[1:0]= 0= REM UNDISP ALARM
  1256 1250 CHKA25    246 AC EX  X             C.X= ALARM ADDRESS
  1257 1251             1 GOSUB  SKPALC        A.X= C.X= NXT ALARM ADDRESS
  1257 1252             0                      *TIMER ROM:  TM2, @1122
  1258 1253 CHKA35   1160 DADD=C
  1259 1254            70 C=DATA
  1260 1255          1076 C=C+1  S             END OF ALARM STACK ?
  1261 1256          1540 RTN C                (TRAILER REGISTER ?) YES
  1262 1257          1176 C=C-1  S
  1263 1260            34 PT=    3
  1264 1261          1422 ? A<C  PQ            IS THIS A FUTURE ALARM ?
  1265 1262          1613 GONC   CHKA20 (1243) NO
  1266 1263          1434 PT=    1             YES
  1267 1264           102 C=0    PT            UNMARK FUTURE ALARMS
  1268 1265          1360 DATA=C
  1269 1266          1623 GOTO   CHKA25 (1250)
*
```

```
*********************************************************************
* SWPM&D - SWAP MONTH AND DAY                          12-18-80 RSW
*       IF FLAG 31= 0 (M.DY), SWAPS MONTH & DAY
*       IF FLAG 31= 1 (D.MY), DOES NOT SWAP MONTH & DAY
*    SO THE OUTPUT IS ALWAYS D.MY
*
* IN:     A= . MMDDYYYY.. ... AND FLAG 31= 0 OR
*         A= . DDMMYYYY.. ... AND FLAG 31= 1
*         WHERE "." MEANS "DON'T CARE"
* ASSUME: NOTHING
* OUT:    C= 0 DDMMYYYY00 000
*         CHIP 0 ENABLED, PT= 6, HEXMODE
* USES:   A, C, ACTIVE PT, DADD, PFAD, ARITH MODE
*         (NO ST, +0 SUB LEVELS, NO TIMER CHIP ACCESS)
*
*   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .
*
* GFLG31 = GET FLAG 31
*
* IN:     A.X MUST BE NON-ZERO
* ASSUME: HEXMODE
* OUT:    C[6]= 0 IF FLAG 31= 0   (MDY)
*         C[6]= 1 IF FLAG 31= 1   (DMY)
*         CHIP 0 ENABLED, PT=6
* USES:   C, ACTIVE PT, DADD, PFAD
*         (NO ST, +0 SUB LEVELS, NO TIMER CHIP ACCESS)
*
*********************************************************************
*
  1300                   ENTRY  SWPM&D
  1301                   ENTRY  GFLG31
  1302 1267 SWPM&D  134 PT=    4            SWAP MONTH & DAY
  1303 1270          12 A=0    WPT
  1304 1271        1140 SETHEX
  1305 1272 GFLG31  106 C=0    X            GET FLAG 31
  1306 1273        1760 PFAD=C
  1307 1274        1160 DADD=C
  1308 1275        1670 C=REGN 14
  1309 1276         534 PT=    6            FLAG 15 IN DIGIT.6, BIT.3
  1310 1277         742 C=C+C  PT
  1311 1300         742 C=C+C  PT
  1312 1301         742 C=C+C  PT
  1313 1302        1506 ? A#0  X            GET FLAG 31 ONLY ??
  1314 1303        1540 RTN C               YES, DONE
  1315 1304         742 C=C+C  PT           FLAG 31 SET ?
  1316 1305          33 GONC   SWPMD4 (1310) NO, MDY, SWAP MONTH & DAY
  1317 1306         256 AC EX                YES, C= DMY DATE
  1318 1307         113 GOTO   SWPMD8 (1320)
  1319 1310 SWPMD4  256 AC EX                ASSUME, FOR EXAMPLE, MDY:
  1320 1311         674 RCR    11           C= D DYYYY00000 0MM
  1321 1312         406 A=C    X
  1322 1313        1574 RCR    12           C= Y YYY000000M MDD
  1323 1314         246 AC EX  X            C.X= 0MM, A.X= MDD
  1324 1315        1074 RCR    2            C= M MYYYY00000 0M0
  1325 1316         246 AC EX  X
  1326 1317          74 RCR    3
  1327 1320 SWPMD8  136 C=0    S            C= 0 DDMMYYYY00 000
  1328 1321        1740 RTN
*********************************************************************
* CLK12                                                 1-6-81 RSW
```

```
*************************************************************************
  1332 1322          262 CON    @262          2
  1333 1323           61 CON    @61           1
  1334 1324           13 CON    @13           K
  1335 1325           14 CON    @14           L
  1336 1326            3 CON    @03           C
  1337                   ENTRY  CLK12
  1338 1327 CLK12      1 GOSUB  ITMRST        GET TIMER SOFTWARE STATUS
  1338 1330            0                      *TIMER ROM:  TM0, @1367
  1339 1331          504 S6=    0             CLEAR 24 HR FORMAT BIT
  1340                                         (TO GET 12-HOUR FORMAT)
  1341 1332 CLKST   1630 C=ST
  1342 1333          450 WRSCR                 UPDATE SCRATCH REGISTER B
  1343 1334         1740 RTN
*************************************************************************
* CLK24                                                        1-6-81 RSW
*************************************************************************
  1347 1335          264 CON    @264          4
  1348 1336           62 CON    @62           2
  1349 1337           13 CON    @13           K
  1350 1340           14 CON    @14           L
  1351 1341            3 CON    @03           C
  1352                   ENTRY  CLK24
  1353 1342 CLK24      1 GOSUB  ITMRST        GET TIMER SOFTWARE STATUS
  1353 1343            0                      *TIMER ROM:  TM0, @1367
  1354 1344          510 S6=    1             SET 24 HR FORMAT BIT
  1355 1345         1653 GOTO   CLKST  (1332)
*************************************************************************
* CLKT                                                         1-6-81 RSW
*************************************************************************
  1359 1346          224 CON    @224          T
  1360 1347           13 CON    @13           K
  1361 1350           14 CON    @14           L
  1362 1351            3 CON    @03           C
  1363                   ENTRY  CLKT
  1364 1352 CLKT       1 GOSUB  ITMRST        GET TIMER SOFTWARE STATUS
  1364 1353            0                      *TIMER ROM:  TM0, @1367
  1365 1354         1204 S7=    0             CLEAR "TIME & DATE" BIT
  1366 1355         1553 GOTO   CLKST  (1332)
*************************************************************************
* CLKTD                                                        1-6-81 RSW
*************************************************************************
  1370 1356          204 CON    @204          D
  1371 1357           24 CON    @24           T
  1372 1360           13 CON    @13           K
  1373 1361           14 CON    @14           L
  1374 1362            3 CON    @03           C
  1375                   ENTRY  CLKTD
  1376 1363 CLKTD      1 GOSUB  ITMRST        GET TIMER SOFTWARE STATUS
  1376 1364            0                      *TIMER ROM:  TM0, @1367
  1377 1365         1210 S7=    1             SET "TIME & DATE" BIT
  1378 1366         1443 GOTO   CLKST  (1332) (TO SHOW TIME & DATE)
*************************************************************************
* ITMRST= INITIALIZE AND PUT UP TIMER STATUS                   1-6-81 RSW
*
* IN:     WARM START CONSTANT IN ALARM B REGISTER
*         SOFTWARE STATUS BITS IN SCRATCH REG B
* ASSUME: NOTHING
* OUT:    TIMER CHIP ENABLED, TIMER PT=B, TIMER SOFTWARE STATUS UP,
*         RAM DISABLED, HEXMODE
```

```
* USES:   A, C, S0-S7, TIMER PT, DADD, PFAD, +2 SUB LEVELS, ARITH MODE
*         (NO 41C PT)
*
  1390                    ENTRY  ITMRST
  1391 1367 ITMRST     1 GOSUB  INITMR         INIT TIMER IF NECESSARY
  1391 1370             0                      *TIMER ROM:  TM0, @1524
  1392 1371             1 GOLONG TMRST          TIMER STATUS ROUTINE
  1392 1372             2                      *TIMER ROM:  TM3, @0253
*
*************************************************************************
* DSPDTA - DISPLAY THE DATE IN "A"                              1-28-81 BW
* SHIFTS THE DATE CHARACTERS LEFT INTO THE DISPLAY
*     (THE DISPLAY IS NOT CLEARED)
*
* IN:     DISPLAY PROPERLY INIT'ED. DISPLAY !!MUST!! BE CLEARED IF S2=1
*         (DSPDTA DOES NOT CLEAR THE DISPLAY !!!!!)
*         A= POSITIVE NORMALIZED F.P. DATE (WITH EXPONENT= 0 OR 1)
*         !! IT MUST BE A VALID DATE -- NO ERROR CHECKING IS DONE !!
* ASSUME: S2 = 0  TO DISPLAY ONLY MONTH AND DAY
*         S2 = 1  TO DISPLAY MONTH, DAY, YEAR, AND DAY OF WEEK
*            IF S2= 1 MUST ALSO HAVE R8[13:8]= DATE # SINCE OCT 15, 1582
* OUT:    CHIP 0 ENABLED, PERIPHERALS DISABLED, HEXMODE
*         S5= 1 (0)  FOR 24 (12) HOUR FORMAT
*         IF S2= 1, DISPLAY IS LEFT-JUSTIFIED, AND P SELECTED
*         IF S2= 0, DISPLAY HAS NOT BEEN LEFT-JUSTIFIED, AND POINTER
*                   SELECT IS SAME AS INPUT
* USES:   A,B,C,G, (P,Q IF S2=1)/(ACTIVE PT IF S2=0), S0,S3-S5, +2 SUB
*         LEVELS (NO TIMER CHIP ACCESS)
*
*    *    *    *    *    *    *
* DSPDT -- SAME AS DSPDTA EXCEPT:
* IN:     A= DDDDDDD........= DAY NUMBER SINCE 1/1/1900
* ASSUME: SAME AS DSPDTA
* OUT:    SAME AS DSPDTA, BUT ALSO R8[13:8]= DAY # SINCE OCT 15, 1582
* USES:   A,B,C,G,N,R8[13:6], (P,Q IF S2=1)/(ACTIVE PT IF S2=0), S0,
*         S3-S5, +2 SUB LEVELS, ARITH MODE, DADD, PFAD (NO TIMER ACCESS)
*************************************************************************
*
  1423                    ENTRY  DSPDT
  1424                    ENTRY  DSPDTA
  1425
  1426 1373 DSPDT      1 GOSUB  DAYMDF         A= DATE, R8[13:8] = DAY NUM
  1426 1374             0                      *TIMER ROM:  TM0, @1635
  1427 1375 DSPDTA  1506 ? A#0  X              MONTH(DAY) ONLY 1 DIGIT ?
  1428 1376           27 GOC    DSDT10 (1400) NO, MONTH(DAY) IS 2 DIGITS
  1429 1377         1616 A SR                  A= 00MDDYYYY00000
  1430 1400 DSDT10   460 LDI                   LOAD LOW 12 BITS OF C WITH
  1431 1401            2 CON    2             2= DISPLAY MONTH & DAY
  1432 1402         1014 ?S2=1                 DISPLAY YEAR ALSO ?
  1433 1403          153 GONC   DSDT20 (1420) NO
  1434 1404          340 SEL Q
  1435 1405          434 PT=    8
  1436 1406          240 SEL P
  1437 1407          434 PT=    8
  1438 1410          120 LC     1
  1439 1411         1120 LC     9
  1440 1412         1234 PT=    7
  1441 1413          460 LDI                   LOAD LOW 12 BITS OF C WITH
  1442 1414            5 CON    5             5 TO DISPLAY 4-DIGIT YEAR
  1443 1415         1562 ? A#C  PQ             CENTURY = 19 ?
```

```
  1444 1416          27 GOC    DSDT20 (1420) NO, DISPLAY 4-DIGIT YEAR
  1445 1417        1146 C=C-1  X             C.X= 4 TO DISP 2-DIGIT YEAR
  1446 1420 DSDT20 1610 S0=    1             DOING A DATE
  1447 1421         110 S4=    1             FORMATTING FOR DISPLAY
  1448 1422         204 S5=    0             USE CLK12/CLK24 BIT = AM/PM
  1449 1423           1 GOSUB  DATEIN        DATE TO DISPLAY
  1449 1424           0                      *TIMER ROM:  TM2, @1567
  1450 1425        1014 ?S2=1                DISPLAY DAY OF WEEK ?
  1451 1426         413 GONC   DSDT70 (1467) NO
  1452 1427           1 GOSUB  LEFTJ         LEFT-JUSTIFY DISPLAY
  1452 1430           0                      *MAINFRAME: CN10, @1767
  1453                                       IN & ASSUME: LCD ENABLED
  1454                                                    RAM DISABLED
* !!!! DISPLAY MUST NOT CONTAIN ALL BLANKS!   "LEFTJ" WILL NEVER RTN!!!!
  1456                                       OUT: DISPLAY LEFT-JUSTIFIED
  1457                                       USES: A.X,C, ACTIVE PT
  1458                                        (NO ST, +0 SUB LEVELS)
  1459
  1460 1431          116 C=0
  1461 1432          460 LDI                 LOAD LOW 12 BITS OF C WITH
  1462 1433           40 CON    @40          @40= ASCII BLANK
* WOULD PROBABLY SAVE CODE BY PUTTING THIS CHECK FOR BLANKS IN "DSWEEK".
  1464 1434          416 A=C                 A.S=0, A.X=@40
  1465 1435        1670 FRSABC
  1466 1436        1670 FRSABC               WILL ALWAYS BE 2 BLANKS
  1467 1437        1670 FRSABC
  1468 1440        1546 ? A#C  X             IS THIS CHARACTER USED ?
  1469 1441          57 GOC    DSDT45 (1446) YES
  1470 1442         576 A=A+1  S             3 CHARACTERS AVAILABLE
  1471 1443        1670 FRSABC               LOOK AT NEXT CHARACTER
  1472 1444        1546 ? A#C  X             NEXT CHARACTER USED ?
  1473 1445          43 GONC   DSDT50 (1451) NO, IT'S A BLANK
  1474 1446 DSDT45 1730 CST EX
  1475 1447        1210 S7=    1             ADD A COLON
  1476 1450        1730 CST EX
  1477 1451 DSDT50 1750 SLSABC               RESTORE THE CHARACTER
  1478 1452           1 GOSUB  ENCP00        ENA CHIP 0, DIS PERIPHERALS
  1478 1453           0                      *MAINFRAME:  CN0, @0230
  1479                                       IN:     HEXMODE
  1480                                       ASSUME: HEXMODE
  1481                                       USES:   C.X ONLY
  1482 1454        1070 C=REGN 8             C[13:8]=DAY # > 10/15/1582
  1483 1455           1 GOSUB  ENLCD         ENABLE DISP, DISABLE RAM
  1483 1456           0                      *MAINFRAME:  CN1, @1766
  1484                                       IN:     NOTHING
  1485                                       ASSUME: NOTHING
  1486                                       OUT:    LCD ENA, RAM DIS
  1487                                       USES:   C.X, DADD, PFAD
  1488 1457        1234 PT=    7
  1489 1460         112 C=0    WPT
  1490 1461         236 B=A    S              SAVE A.S IN B.S
  1491 1462           1 GOSUB  WKDAYS         DAYS SINCE 10/15/1582->DOW
  1491 1463           0                      *TIMER ROM:  TM0, @0576
  1492 1464         176 AB EX  S              A.S= 0 (1) FOR 2 (3) CHARS
  1493 1465           1 GOSUB  DSWEEK         DISP DAY OF WEEK IN ENGLISH
  1493 1466           0                      *TIMER ROM:  TM3, @0115
  1494 1467 DSDT70    1 GOLONG ENCP00         ENA CHIP 0, DIS PERIPHERALS
  1494 1470           2                      *MAINFRAME:  CN0, @0230
  1495                                       SEE ABOVE COMMENTS
*************************************************************************
```

```
* TGLSHF= TOGGLE SHIFT ANNUNCIATOR                          1-12-81 RSW
*
* IN & ASSUME: S7= EXISTING STATE OF SHIFT ANNUNCIATOR (1= ON, 0= OFF)
* OUT:  CHIP 0 ENABLED, S7 IN OPPOSITE STATE WITH MATCHING ANNUNCIATOR
* USES: C, S7, +1 SUB LEVEL, DADD, PFAD
*       ( NO PT, NO ARITH MODE, NO TIMER CHIP ACCESS )
*
  1504                    ENTRY  TGLSHF        (26 MAX INC GSB & RTN)
  1505 1471 TGLSHF    1 GOSUB  ENLCD          ENABLE LCD, DISABLE RAM
  1505 1472           0                       *MAINFRAME:  CN1, @1766
  1506                                        IN:     NOTHING
  1507                                        ASSUME: NOTHING
  1508                                        OUT:    LCD ENA, RAM DIS
  1509                                        USES:   C.X, DADD, PFAD
  1510 1473         570 READEN                READ ANNUNCIATORS
  1511 1474        1214 ?S7=1                 SHIFT SET ?
  1512 1475          53 GONC   TGLS10 (1502) NO, SET IT
  1513 1476        1204 S7=    0              YES, CLEAR IT
  1514 1477        1730 CST EX
  1515 1500        1204 S7=    0              CLEAR SHIFT ANNUNCIATOR
  1516 1501          43 GOTO   TGLS20 (1505)
  1517 1502 TGLS10 1210 S7=    1              SET SHIFT
  1518 1503        1730 CST EX
  1519 1504        1210 S7=    1              SET SHIFT ANNUNCIATOR
  1520 1505 TGLS20 1730 CST EX
  1521 1506 TGLS30 1360 WRTEN                 WRITE OUT ANNUNCIATORS
  1522 1507 ENCP0J 1603 GOTO   DSDT70 (1467) ENA CHIP 0, DIS PERIPHERALS
*******************************************************************
* CHKLB = CHECK LOW BATTERY                                  2-6-81 RSW
*
* IN & ASSUME: HEXMODE
* OUT:         LOW BATTERY ANNUNCIATOR SET IF LOW BATTERY IS DETECTED
*              (CHIP ENABLE= CHIP 0 OR SAME AS INPUT)
* USES:        C, DADD, PFAD, +1 SUB LEVEL
*              (NO ST, NO PT, NO TIMER CHIP ACCESS)
*
* EXEC TIME: 28 WORD TIMES INCLUDING GSB & RTN IF BATTERY IS LOW
*
*
  1535                    ENTRY  CHKLB
  1536 1510 CHKLB   540 ?LLD                  LOW BATTERY ?
  1537 1511        1640 RTN NC                NO
  1538 1512           1 GOSUB  ENLCD          ENABLE THE DISPLAY
  1538 1513           0                       *MAINFRAME:  CN1, @1766
  1539                                        IN:     NOTHING
  1540                                        ASSUME: NOTHING
  1541                                        OUT:    LCD ENA, RAM DIS
  1542                                        USES:   C.X, DADD, PFAD
  1543 1514         570 READEN                READ ANNUNCIATORS
  1544 1515        1474 RCR    1
  1545 1516        1730 CST EX
  1546 1517        1210 S7=    1              SET LOW BATT ANNUNCIATOR
  1547 1520        1730 CST EX
  1548 1521        1374 RCR    13
  1549 1522        1643 GOTO   TGLS30 (1506)
  1550
*******************************************************************
*                                                            4-2-81 RSW
* INITMR= INITIALIZE TIMER (IF IT HAS JUST POWERED UP)
*       IF HARDWARE POWER UP STATUS BIT IS SET OR THE WARM START CONSTANT
```

```
*      IN ALARM B REGISTER IS WRONG THEN "INITMR" DOES THE FOLLOWING:
*         - CLEARS & STARTS MAIN CLOCK
*         - CLEARS & STOPS STOPWATCH CLOCK
*         - DISABLES ALARMS
*         - CLEARS SCRATCH REGISTERS A & B
*         - CLEARS ALARM STATUS BITS, POWER UP STATUS BIT, AND
*           ACCURACY FACTOR
*         - STOPS INTERVAL TIMER
*         - DISABLES BOTH TEST MODES
*         - SETS ALARM B REG= 0 9999999999 000= WARM START CONSTANT
*
* IN:     WARM START CONSTANT IN ALARM B REGISTER
* ASSUME: NOTHING
* OUT:    TIMER CHIP ENABLED, RAM DISABLED, TIMER PT=B, HEXMODE
*              (DON'T TRUST THE HARDWARE STATUS SET THAT IS UP)
* USES:   A, C, ST[7:0], TIMER PT, DADD, PFAD, +1 SUB LEVEL, ARITH MODE
*         (NO 41C PT)
* MINIMUM EXECUTION TIME= 26 WORD TIMES (INCLUDING GOSUB & RETURN)
*
*  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
*
* INITM1 - SAME AS INITMR EXCEPT EXPECTS TIMER CHIP ENABLED,
*          RAM DISABLED, AND TIMER PT=A.
* USES:    A, C, S0-S7, TIMER PT, ARITH MODE
*           (NO DADD, NO PFAD, NO 41C PT, +0 SUB LEVELS)
*
  1581                  ENTRY  INITMM
  1582                  ENTRY  INITMR
  1583                  ENTRY  INITM1
  1584 1523 INITMM  530 M=C
  1585 1524 INITMR    1 GOSUB  ENTMR          SAVE A SUB LEVEL, PT=A
  1585 1525           0                       *TIMER ROM:  TM0, @0342
  1586 1526        1550 STARTC                START MAIN CLOCK
  1587 1527 INITM1  370 RDSTS                READ HARDWARE STATUS
  1588 1530        1530 ST=C                  ST= HARDWARE STATUS
  1589 1531        1650 PT=B
  1590 1532         270 RDALM                 ALM B= WARM START CONSTANT
  1591 1533         416 A=C                   A= WARM START CONSTANT
  1592 1534         116 C=0
  1593 1535        1240 SETDEC
  1594 1536        1172 C=C-1  M              C= 09999999999000
  1595 1537        1140 SETHEX
  1596 1540        1556 ? A#C                 WARM START CONSTANT WRONG ?
  1597 1541          37 GOC    INITM3 (1544) YES, INITIALIZE TIMER CHIP
  1598 1542         214 ?S5=1                 HARDWARE POWER UP BIT SET ?
  1599 1543        1640 RTN NC                NO
  1600 1544 INITM3  250 WRALM                ALARM B= 09999999999000
  1601 1545         116 C=0
  1602 1546        1050 DSWKUP                DISABLE TEST MODE B
  1603 1547        1450 STOPC                 STOP STOPWATCH CLOCK
  1604 1550        1250 DSALM                 DISABLE CLOCK B ALARM
  1605 1551          50 WRTIME                CLEAR STOPWATCH CLOCK
  1606 1552         450 WRSCR                 CLEAR SCRATCH REGISTER B
  1607 1553         350 WRSTS                 CLEAR ACCURACY FACTOR
  1608 1554        1750 PT=A
  1609 1555        1050 DSWKUP                DISABLE TEST MODE A
  1610 1556         750 STPINT                STOP INTERVAL TIMER
  1611 1557        1250 DSALM                 DISABLE ALARM B
  1612 1560          50 WRTIME                CLEAR MAIN CLOCK
  1613 1561         450 WRSCR                 CLEAR LAST TIME SET
```

```
  1614 1562         350 WRSTS               CLEAR ALL ALARMS
  1615 1563        1650 PT=B
  1616 1564        1740 RTN
  1617
*********************************************************************
*********************************************************************
*                           CALENDAR ROUTINES
*
*
* Let MONTH = month number (1,2,...,12)
* Let DAY   = day number in month = (1,2,...,31)
* Let YEAR  = year number = (1582,1583,...,4320)
* Let DAY#  = day number since October 15, 1582 [oct 15, 1582 = day 0]
*
* Now define the following conditionally depending on value of MONTH:
*     if MONTH < 3 then let  M = MONTH + 13 and let Y = YEAR - 1.
*     if MONTH >= 3 then let M = MONTH + 1  and let Y = YEAR.
*
* Also define the following functions:
*     SUM3(Y) = int(Y * 365.25) - int(Y/100) + int(Y/400)
*     M306(M) = int(M & 30.6001)
*
*
* Mapping DATE to DAY# :
*     DAY#(MONTH,DAY,YEAR) = SUM3(Y) + M306(M) + DAY - 578164
*
*
* Mapping DAY# to Date :
*     Calculate the value of Y0 as follows:
*        Y0 = int( [(DAY# + 578164) - 121.5] / 365.2425)
*        This is an approximation of the correct year.
*     Now calculate M0 as follows:
*        M0 = int( [(DAY# + 578164) - SUM3(Y0)] / 30.6001)
*     If this M0 is less than 4 then Y0 was one too high
*        therefore let Y0 = Y0 - 1 and recalculate M0 using the new Y0.
*     Once M0 >= 4, the values of MONTH, DAY and YEAR are:
*            DAY = [(DAY# + 578164) - SUM3(Y0)] - M306(M0)
*            If M0 >= 14 then MONTH = M0 - 13 and  YEAR = Y0 + 1
*            If M0 < 14  then MONTH = M0 - 1  and  YEAR = Y0
*********************************************************************
*********************************************************************
* C-YMDD = C REGISTER INPUT -- YEAR,MONTH,DAY TO DAY#       1-23-81 RSW
*
* IN:     C= FLOATING POINT DATE (!!MUST BE VALID F.P. NUMBER!!)
*           THE LEGAL DATE RANGE IS OCT 15, 1582 TO SEPT 10, 4320.
*           DATES OUTSIDE THIS RANGE, MONTH= 0 OR 13-99 AND ILLEGAL
*           DAY WILL BE DETECTED.
* ASSUME: NOTHING
* OUT:    R8[13:8]= DAY NUMBER SINCE OCT 15, 1582= DDDDDD........
*             (NOT TRUE FOR ERROR CASE)
*         A= FLOATING POINT DATE        ( M.DY OR D.MY )
*            !!! THIS DATE MUST BE COMPARED TO THE INPUT DATE.  IF THEY
*                ARE NOT IDENTICAL F.P. NUMBERS (EXCEPT FOR SIGN) THE
*                INPUT DATE IS NOT VALID (SO R8[13:8] IS TRASH)
*         CHIP 0 ENABLED   (IN ALL CASES)
*         HEXMODE          (IF NO ERROR OCCURRED)
* USES:   A,B,C,N,R8[13:6], ACTIVE PT, +1 SUB LEVEL, ARITH, PFAD, DADD
*                      (NO ST, NO TIMER CHIP ACCESS)
*
  1673                 ENTRY  C-YMDD
```

```
  1674 1565 C-YMDD    1 GOSUB  UNNOR1        UNNORMALIZE THE DATE
  1674 1566           0                      *TIMER ROM:  TM3, @0022
  1675 1567          23 GOTO   DAYS20 (1571) (P+1)
  1676                                       (P+2) ERROR
  1677 1570        1173 GOTO   ENCP0J (1507) ENABLE CHIP 0, DISABLE RAM
*
*   NOTE:  NOW A[X]= 1 WHICH IS DIFFERENT FROM THE INPUT DATE, SO THE
*          DATE COMPARISON AFTER "GOSUB C-YMDD" WILL CATCH THE ERROR.
  1681
  1682
  1683 1571 DAYS20    1 GOSUB  SWPM&D        SWAP MONTH & DAY IF FL31= 0
  1683 1572           0                      *TIMER ROM:  TM0, @1267
  1684 1573        1240 SETDEC               (NOW HAVE DMY)
  1685 1574          16 A=0
  1686 1575        1074 RCR    2             C= 0 00DDMMYYYY 000
*      ...... PT= 6 FROM SWPM&D .............
  1688 1576         252 AC EX  WPT           A= 0000000YYYY000
  1689 1577         574 RCR    6             C= 0 00000000DD MM0
  1690 1600        1706 C SR   X
  1691 1601         246 AC EX  X             A.X= 0MM, C.X= 0
  1692 1602         160 N=C                  N= 0 00000000DD 000
  1693 1603         546 A=A+1  X             M= M+1
  1694 1604         460 LDI                  LOAD LOW 12 BITS OF C WITH
  1695 1605           4 CON    4             4TH MONTH FOR COMPARISON
  1696 1606        1406 ? A<C  X             M < 4 ?
  1697 1607          53 GONC   YMDD40 (1614) NO
  1698 1610         460 LDI                  LOAD LOW 12 BITS OF C WITH
  1699 1611          22 CON2   1      2       @22 = DECIMAL 18
  1700 1612         506 A=A+C  X             MONTH + 12 (GET "MTH + 13")
  1701 1613         672 A=A-1  M             YEAR - 1
  1702 1614 YMDD40  232 B=A    M             B.M= 000000YYYY
  1703 1615         116 C=0
  1704 1616         246 AC EX  X             C.X= 0MM, A.X= 000
  1705 1617         674 RCR    11            C= 0 00000000MM 000
  1706 1620           1 GOSUB  M306          CALCULATE DAYS PER MONTH
  1706 1621           0                      *TIMER ROM:  TM3, @0000
  1707 1622         416 A=C                  A= M*30.6
  1708 1623         260 C=N                  C= 0 00000000DD 000
  1709 1624        1032 C=A+C  M             C.M= DAY + INT[M * 30.6]
  1710 1625        1232 C=-C   M
  1711 1626         160 N=C                  N.M= -(DAY + M306[M])
  1712 1627         332 C=B    M             C.M= 000000YYYY
  1713 1630           1 GOSUB  SUM3D5        CALC N.M + 578164 - SUM3(Y)
  1713 1631           0                      *TIMER ROM:  TM3, @0351
  1714 1632        1232 C=-C   M          C.M=-[-(DAY+M306)+578164-SUM3]
  1715                                        =SUM3(Y)+M306(M)+DAY-578164
  1716 1633        1174 RCR    9             C= DDDDDDXXXX0000
*  NOTE:  FOR DATES BEYOND SEPT 10, 4320 (DAY#= 999999) THE DAY# WILL BE
*         7 DIGITS. THE SEVENTH DIGIT WILL NOW BE IN C[0] WHERE IT WILL
*         BE TRUNCATED SO THE RECONSTRUCTED DATE WILL NOT MATCH THE
*         INPUT DATE.
*
  1722 1634          53 GOTO   DAYYMD (1641)
*************************************************************************
* DAYMDF -- SAME AS DAYYMD EXCEPT:
*    IN:  A= DDDDDD........= DAY NUMBER SINCE 1/1/1900
*
  1727                  ENTRY  DAYMDF
  1728 1635 DAYMDF    1 GOSUB  115860        C= 11586000000000
  1728 1636           0                      *TIMER ROM:  TM2, @1440
```

```
  1729 1637        1240 SETDEC
  1730 1640        1016 C=A+C                ADD 115860 TO DAY NUMBER
  1731
*************************************************************************
* DAYYMD = DAY NUMBER TO YEAR,MONTH,DAY                      1-22-81 RSW
* CALCULATES THE DATE FROM THHE DAY NUMBER
*
* IN:     DECMODE
*         C= DDDDDD........= DAY NUMBER SINCE 10-15-1582
*            WHERE 10-15-1582 = 000000,  9-10-4320 = 999999
* ASSUME: NOTHING
* OUT:    A= C= POSITIVE FLOATING POINT DATE   (M.DY OR D.MY)
*         R8= DDDDDD........= COPY OF "C" ON INPUT
*         HEXMODE, CHIP 0 ENABLED
* USES:   A,B,C,N,R8[13:6], ACTIVE PT, +1 SUB LEVEL, ARITH, PFAD, DADD
*                 (NO ST, NO TIMER CHIP ACCESS)
*
*
* DAYMD -- SAME AS DAYYMD EXCEPT:   INPUT= A= DDDDDD........
*
  1749                    ENTRY  DAYMD
  1750 1641 DAYYMD  416 A=C
  1751 1642 DAYMD     1 GOSUB  ENCP00         ENABLE CHIP 0, DISABLE RAM
  1751 1643           0                       *MAINFRAME:  CN2, @0522
  1752                                        IN:     NOTHING
  1753                                        ASSUME: NOTHING
  1754                                        USES:   C.X, DADD, PFAD
  1755 1644        1070 C=REGN 8
  1756 1645        1234 PT=    7
  1757 1646         412 A=C    WPT
  1758 1647         256 AC EX                 C= DDDDDD........
  1759 1650        1050 REGN=C 8              SAVE DAY# IN R8
  1760 1651         112 C=0    WPT
  1761 1652         174 RCR    4              C= 0 000DDDDDD0 000
  1762 1653         416 A=C                   A= 0 000DDDDDD0 000
  1763 1654         116 C=0
  1764 1655        1134 PT=    9              "A+C" MAY CARRY TO DIGIT 10
  1765 1656         520 LC     5
  1766 1657         720 LC     7
  1767 1660        1020 LC     8              C= 0 0005780425 000
  1768 1661          20 LC     0
  1769 1662         420 LC     4
  1770 1663         220 LC     2
  1771 1664         520 LC     5              (578164-121.5)*10
  1772 1665         516 A=A+C                 [MAX= 0 0015780415 000]
  1773 1666        1134 PT=    9
  1774 1667         320 LC     3
  1775 1670         620 LC     6
  1776 1671         520 LC     5
  1777 1672         220 LC     2              C[9:3]= 365.2425
  1778 1673         334 PT=    10
*   BECAUSE MSD OF DIVIDEND (DIGIT 11) IS AT MOST = 1, MSD OF DIVISOR IS
*   POSITIONED AT DIGIT 10.          Y0= INT[(D#+5788164-121.5)/365.2425]
  1781 1674           1 GOSUB  IDVD4          CALCULATE Y0 AS SHOWN ABOVE
  1781 1675           0                       *TIMER ROM:  TM0, @0320
*   MSD OF THE QUOTIENT IS IN DIGIT 11 DUE TO THE SPECIAL POSITIONING OF
*   THE DIVISOR AND THE SMALL MSD (MSD <= 1) OF THE DIVIDEND.
  1784 1676         256 AC EX                 C= 0 0YYYY00000 000
  1785 1677         274 RCR    5
  1786 1700         416 A=C                   A= 0 900000YYYY 000 = Y0
```

```
 1787 1701 DAYY20 1070 C=REGN 8
 1788 1702        1234 PT=    7
 1789 1703         112 C=0    WPT           C= DDDDDD00000000
 1790 1704         274 RCR    5
 1791 1705         160 N=C                  N= DAY#= 0 0000DDDDDD 000
 1792 1706         272 AC EX  M             C.M= 000000YYYY (R=RESULT)
 1793 1707           1 GOSUB  SUM3D5         C= 0 0000000RRR 000 ^^^^^
 1793 1710           0                      *TIMER ROM:  TM3, @0351
 1794 1711         160 N=C                  N.M= DAY#+578164-SUM3(Y0)
 1795 1712         674 RCR    11            MOVE FOR "IDVD" BY 30.6001
 1796 1713         416 A=C                  A.M= DAY#+578164-SUM3(Y0)
 1797 1714         116 C=0
 1798 1715         434 PT=    8
 1799 1716         320 LC     3
 1800 1717          20 LC     0
 1801 1720         620 LC     6
 1802 1721        1072 C=C+1  M             C.M= 0000306001
 1803 1722        1134 PT=    9
 1804 1723           1 GOSUB  IDVD          DIVIDE BY 30.6001
 1804 1724           0                      *TIMER ROM:  TM0, @0322
 1805 1725         116 C=0                  A= 0 00MM000000 000
 1806 1726        1134 PT=    9
 1807 1727         420 LC     4
 1808 1730        1416 ? A<C                MONTH < 4 ?
 1809 1731          53 GONC   DAYY40 (1736) NO
 1810 1732         316 C=B                  C= R 000000YYYY RRR
 1811 1733        1172 C=C-1  M             Y= Y-1
 1812 1734         432 A=C    M             A= 0 000000YYYY 000
 1813 1735        1443 GOTO   DAYY20 (1701)
 1814
*  IN:  N.M=  DAY# + 578164 - SUM3(Y0)
*       B.M=  000000YYYY
 1817 1736 DAYY40  256 AC EX               C= 0 00MM000000 000
 1818 1737        1174 RCR    9
 1819 1740         406 A=C    X             A.X= 0MM
 1820 1741         260 C=N
 1821 1742         246 AC EX  X
 1822 1743         160 N=C                  SAVE M0 IN N.X
 1823 1744         132 C=0    M             C= 0 0000000000 0MM
 1824 1745           1 GOSUB  M306          CALCULATE DAYS PER MONTH
 1824 1746           0                      *TIMER ROM:  TM3, @0000
 1825 1747         674 RCR    11            C.M= INT[M * 30.6]
 1826 1750         460 LDI                  LOAD LOW 12 BITS OF C WITH
 1827 1751          22 CON2   1      2       @22 = 18 DECIMAL
 1828 1752         416 A=C                  A.M= INT[M * 30.6], A.X= 12
 1829 1753         260 C=N                  C.M= DAY#+578164-SUM3(Y0)
 1830
 1831 1754         272 AC EX  M
 1832 1755        1132 C=A-C  M             C.M= DAY= 00000000DD
 1833 1756         172 AB EX  M             A.M= Y0=  000000YYYY
 1834 1757        1146 C=C-1  X             M0 - 1
 1835 1760        1406 ? A<C  X             12 < M0-1 ? (M0 >= 14 ?)
 1836 1761          43 GONC   DAYY60 (1765) NO
 1837 1762         572 A=A+1  M             YEAR= Y0+1
 1838 1763         246 AC EX  X             C.X= 012, A.X= M0 - 1
 1839 1764        1106 C=A-C  X             MONTH= M0-1-12= M0-13
 1840 1765 DAYY60 1374 RCR    13            C= 0 00000 00DD0 MM0
 1841 1766        1732 C SR   M
 1842 1767         474 RCR    8
 1843 1770         534 PT=    6
```

```
  1844 1771         252 AC EX  WPT           C= 0 00DDMMYYYY ...
  1845 1772        1574 RCR    12
  1846 1773         416 A=C                  A= 0 DDMMYYYY.. .00
  1847 1774           1 GOSUB  SWPM&D        SWAP MTH & DAY IF FLAG 31=0
  1847 1775           0                      *TIMER ROM:  TM0, @1267
  1848 1776           1 GOLONG NORM          NORMALIZE THE RESULT
  1848 1777           2                      *TIMER ROM:  TM0, @0532
  1849                  FILLTO END
  1850                  END

ERRORS :        0
```

SYMBOL TABLE (BWTMB1 = TIMER ROM QUAD 1 = TM0 = ADDRESSES @50000-51777)

```
AMG10    637   -   635
C-YMDD  1565   -
CHECK    242   -
CHECKX   240   -
CHKA20  1243   -  1262
CHKA25  1250   -  1266 1246
CHKA35  1253   -  1242
CHKALM  1230   -
CHKLB   1510   -
CHKXM    237   -
CLK12   1327   -
CLK24   1342   -
CLKST   1332   -  1366 1355 1345
CLKT    1352   -
CLKTD   1363   -
DATCK2   261   -   257
DATCK4   314   -   262
DATE     400   -
DATE+    115   -
DATECK   255   -
DATX30   414   -   373
DATX37   416   -   406  367
DATX40   417   -   444  434
DAYMD   1642   -
DAYMDF  1635   -
DAYS20  1571   -  1567
DAYY20  1701   -  1735
DAYY40  1736   -  1731
DAYY60  1765   -  1761
DAYYMD  1641   -  1634
DDAYS    177   -
DMY      571   -
DOW      211   -
DOW30    234   -   220
DSA2ND   731   -
DSAM20   653   -   721
DSAM25   670   -   674
DSAM30   675   -   671  665
DSAM35   700   -   714
DSAM37   706   -   743
DSAM45   715   -
DSAM50   722   -   711
DSAM60   744   -   717
DSAM65   751   -   755
DSAM70   756   -   753
DSAM80   757   -   730  723
DSAM90   762   -  1031
DSAMS0   621   -
DSAMSG   622   -
DSDT10  1400   -  1376
DSDT20  1420   -  1416 1403
DSDT45  1446   -  1441
DSDT50  1451   -  1445
DSDT70  1467   -  1507 1426
DSPDT   1373   -
DSPDTA  1375   -
```

```
DT+10    124   -   205
DT+15    151   -   146
DT+24    163   -   150
DT+25    165   -   156
ENCP0J  1507   -  1570
ENTMR    342   -
ENTMRS   341   -
FNDM10  1216   -  1227
FNDMSG  1202   -
GETAF    445   -
GETMR    471   -
GETMRC   475   -
GFLG31  1272   -
GTMR15   502   -   477
GTMR30   510   -
HEADER   110   -
IDVD     322   -
IDVD4    320   -
IDVDL    324   -   334  327
INITM1  1527   -
INITM3  1544   -  1541
INITMM  1523   -
INITMR  1524   -
ITMRST  1367   -
MDY      556   -
MDY10    562   -   575
NORM     532   -
NORM00   536   -   533
NORM05   543   -   546
NORM10   545   -   542
NORM20   550   -   535
NORMC    530   -
PUGA30  1014   -  1006
PUGA40  1023   -  1021
PUGA50  1031   -  1045 1041
PUGALM   764   -
RCLAF    442   -
RCLSW    427   -
RSTA05  1061   -  1064
RSTA10  1071   -  1107
RSTA15  1077   -  1102
RSTA17  1103   -  1100
RSTA20  1110   -  1104
RSTA50  1134   -  1137
RSTALM  1042   -
RUNSST  1032   -
RUNSW    465   -
SHFTD2  1143   -  1141
SHFTD4  1146   -  1200
SHFTD6  1154   -  1152
SHFTD7  1157   -  1172
SHFTD8  1170   -  1156
SHFTDN  1140   -
SWPM&D  1267   -
SWPMD4  1310   -  1305
SWPMD8  1320   -  1307
TERR20   274   -   272  266
TERR50   310   -
TERROR   263   -
TGLS10  1502   -  1475
```

```
TGLS20  1505   -  1501
TGLS30  1506   -  1522
TGLSHF  1471   -
TIME     356   -
TNFRXY   170   -   144
WKDAYS   576   -
X-YMDD   250   -
YMDD40  1614   -  1607
YMDDAY   253   -
```

ENTRY TABLE (BWTMB1 = TIMER ROM QUAD 1 = TM0 = ADDRESSES @50000-51777)

C-YMDD	1565	-
CHECK	242	-
CHECKX	240	-
CHKALM	1230	-
CHKLB	1510	-
CHKXM	237	-
CLK12	1327	-
CLK24	1342	-
CLKT	1352	-
CLKTD	1363	-
DATE	400	-
DATE+	115	-
DATECK	255	-
DAYMD	1642	-
DAYMDF	1635	-
DDAYS	177	-
DMY	571	-
DOW	211	-
DSA2ND	731	-
DSAMS0	621	-
DSAMSG	622	-
DSPDT	1373	-
DSPDTA	1375	-
ENTMR	342	-
ENTMRS	341	-
FNDMSG	1202	-
GETAF	445	-
GETMR	471	-
GETMRC	475	-
GFLG31	1272	-
GTMR30	510	-
HEADER	110	-
IDVD	322	-
IDVD4	320	-
INITM1	1527	-
INITMM	1523	-
INITMR	1524	-
ITMRST	1367	-
MDY	556	-
NORM	532	-
NORMC	530	-
PUGALM	764	-
RCLAF	442	-
RCLSW	427	-
RSTALM	1042	-
RUNSST	1032	-
RUNSW	465	-
SHFTDN	1140	-
SWPM&D	1267	-
TERR20	274	-
TERR50	310	-
TERROR	263	-
TGLSHF	1471	-
TIME	356	-
WKDAYS	576	-
X-YMDD	250	-

YMDDAY 253 -

EXTERNAL REFERENCES (BWTMB1 = TIMER ROM QUAD 1 = TM0 = ADR @50000-51777)

115860	1635			
115860	1636			
36000	505			
36000	506			
AD2-10	141			
AD2-10	142			
ADATE	5			
ADATE	4			
ALMBST	1027			
ALMBST	1030			
ALMCAT	7			
ALMCAT	6			
ALMNOW	11			
ALMNOW	10			
ASCLCD	704			
ASCLCD	705			
ATIM24	15			
ATIM24	14			
ATIME	13			
ATIME	12			
C-YMDD	253			
C-YMDD	254			
CHECK	127			
CHECK	130			
CHECKX	115	250		
CHECKX	116	251		
CHK#S	246			
CHK#S	247			
CHKBUF	772			
CHKBUF	773			
CLK12	13			
CLK12	12			
CLK24	21			
CLK24	20			
CLKT	23			
CLKT	22			
CLKTD	25			
CLKTD	24			
CLLCDE	221	407	651	734
CLLCDE	222	410	652	735
CLOCK	27			
CLOCK	26			
CORECT	31			
CORECT	30			
DATE	33			
DATE	32			
DATE+	35			
DATE+	34			
DATEIN	1423			
DATEIN	1424			
DAYMD	166			
DAYMD	167			
DAYMDF	402	1373		
DAYMDF	403	1374		
DDAYS	37			
DDAYS	36			

```
WKDAYS   215  1463
X-YMDD   200   212
X-YMDD   201   213
X20Q8    516
X20Q8    517
XYZALM    75
XYZALM    74
YMDDAY   133
YMDDAY   134

End of VASM assembly
********************************************************************************
VASM ROM ASSEMBLY          REV.  6/81A            HP-82182A TIMER MODULE

OPTIONS: L C S          COCONUT TIMER             ADDRESSES @52000-53777

     2                  FILE   BWTMB2             COCONUT TIMER Q2 = TM1
*
*
     5
********************************************************************************
* CORRECT = CORRECT TIME & ACCURACY FACTOR                      2-3-81 RSW
********************************************************************************
     9    0          224 CON    @224           T
    10    1            3 CON    @03            C
    11    2            5 CON    @05            E
    12    3           22 CON    @22            R
    13    4           22 CON    @22            R
    14    5           17 CON    @17            O
    15    6            3 CON    @03            C
    16                   ENTRY  CORECT
    17    7 CORECT 1110 S9=     1              ADJUST ACCURACY FACTOR
    18   10          103 GOTO   XTIM00 (  20)
    19
********************************************************************************
* SETIME = SET THE CURRENT TIME                                 2-3-81 RSW
********************************************************************************
    23   11          205 CON    @205           E
    24   12           15 CON    @15            M
    25   13           11 CON    @11            I
    26   14           24 CON    @24            T
    27   15            5 CON    @05            E
    28   16           23 CON    @23            S
    29                   ENTRY  SETIME
    30   17 SETIME 1104 S9=     0              DON'T CHG ACCURACY FACTOR
    31   20 XTIM00  116 C=0                    NO DATE ENTERED
    32   21            1 GOSUB  CHKXM          ERROR IF X= ALPHA DATA
    32   22            0
    33   23          160 N=C                   N= H.MS NORMALIZED TIME
    34   24            1 GOSUB  R9=T           C= R9= TIME
    34   25            0                       *TIMER ROM:  TM2, @1463
    35   26            1 GOSUB  C=T+D          C= TIME + DATE
    35   27            0                       *TIMER ROM:  TM1, @0222
    36   30         1474 RCR    1              ADD 0.1 SEC TO COMP. FOR
    37   31         1056 C=C+1                 KEY ASSIGNMENT SEARCH TIME
    38   32         1374 RCR    13
    39   33          273 GOTO   TE10   (  62)
    40
********************************************************************************
* SETDATE = SET THE CURRENT DATE                                2-3-81 RSW
```

```
*************************************************************************
    44   34          205 CON    @205           E
    45   35           24 CON    @24            T
    46   36            1 CON    @01            A
    47   37            4 CON    @04            D
    48   40           24 CON    @24            T
    49   41            5 CON    @05            E
    50   42           23 CON    @23            S
    51                 ENTRY  SDATE
    52   43 SDATE    116 C=0
    53   44         1176 C=C-1  S
    54   45          160 N=C                   N.S= F TO USE CURRENT TIME
    55   46            1 GOSUB  CHECKX          ERROR IF X= ALPHA DATA
    55   47            0                        *TIMER ROM:  TM0, @0240
    56   50         1356 ? C#0                 IS X=0 ?
    57   51            1 GOLNC  ERRDE           YES, NOT A VALID DATE
    57   52            2                        *MAINFRAME: CN10, @0055
    58   53          530 M=C                   M= M.DY (D.MY) DATE
    59   54            1 GOSUB  R9=T            C= R9= TIME
    59   55            0                        *TIMER ROM:  TM2, @1463
    60   56         1104 S9=    0              DON'T CHG ACCURACY FACTOR
    61   57         1604 S0=    0              USE X FOR DATE COMPARISON
    62   60            1 GOSUB  C=T+D           C= TIME + DATE
    62   61            0                        *TIMER ROM:  TM1, @0222
    63
    64   62 TE10     160 N=C                   N= NEW ENTERED TIME
    65   63            1 GOSUB  T=T+TP          STORE IT
    65   64            0                        *TIMER ROM:  TM3, @0321
    66   65          470 RDSCR                 READ "LAST TIME SET"
    67   66          416 A=C                   A= LAST TIME SET
    68   67          260 C=N                   C= NEW ENTERED TIME
    69   70          450 WRSCR                 UPDATE "LAST TIME SET"
    70   71         1114 ?S9=1                 ADJUST ACCURACY FACTOR ?
    71   72          723 GONC   ADJ100 ( 164) NO, BEEP IF PAST DUE ALARMS
    72
    73
* IN:   N= NEW ENTERED TIME= 00SSSSSSSSSSCC
*       REG 9= INCORRECT (CLOCK)  TIME= OLD TIME= 00SSSSSSSSSSCC
*       A=     OLD "LAST TIME SET"= LTS= 00SSSSSSSSSSCC
*       DECMODE
*
    79   73            1 GOSUB  ENCP00          ENABLE CHIP 0, DISABLE RAM
    79   74            0                        *MAINFRAME:  CN2, @0522
    80                                          IN:     NOTHING
    81                                          ASSUME: NOTHING
    82                                          OUT:    C.X= 0
    83                                          USES:   C.X, DADD, PFAD
    84   75         1170 C=REGN 9              C= WRONG (CLK) TIME= OLD TM
    85   76          256 AC EX                 C= LTS, C.S= 0, A= OLD TIME
    86   77          716 A=A-C                 A=OLD TIME - LTS= OLD - LTS
    87  100          460 LDI                   LOAD LOW 12 BITS OF C WITH
    88  101           20 CON2   1      0       MAXIMUM EXPONENT=10 (SECS)
    89  102            1 GOSUB  MPY150          NORMALIZE TO 13-DIGIT FORM
    89  103            0                        *MAINFRAME:  CN6, @0145
    90                                          IN: A= 13-DIG MANT A[12:0]
    91                                              C[X&S]= EXPONENT & SIGN
    92                                          ASSUME: DECMODE
    93                                          OUT: C=NORMALIZED FP NUMBER
    94                                               A[X&S]= EXP & SIGN
    95                                               B=13-DIGIT MANTISSA
```

```
 96                                          USES: A,B,C,M, ACTIVE PT
 97
 98  104        1170 C=REGN 9                C= OLD TIME
 99  105         256 AC EX                   A= OLD TIME, C= EXP & SIGN
100  106        1150 REGN=C 9                REG 9= EXP & SIGN (OLD-LTS)
101  107         316 C=B
102  110         360 CN EX                   C= NEW TIME SET= LTS
103                                          N= 13-DIGIT MANT (OLD-LTS)
104  111         256 AC EX                   A= NEW, C= OLD
105  112        1534 PT=    12
106  113        1116 C=A-C                   C= NEW - OLD
107  114          23 GONC   ADJ20  ( 116)    (IF CARRY, C.S= 9)
108  115        1212 C=-C   WPT              CLOCK FAST, OLD > NEW
109  116 ADJ20   412 A=C    WPT              A= 13-DIGIT MANTISSA
110  117         460 LDI                     LOAD LOW 12 BITS OF C WITH
111  120          20 CON2   1      0         MAXIMUM EXPONENT = 10
112  121           1 GOSUB  MPY150           NORMALIZE TO 13-DIGIT FORM
112  122           0                         *MAINFRAME:  CN6, @0145
113  123        1170 C=REGN 9                C= EXP & SIGN (OLD-LTS)
114  124         530 M=C                     M= EXP & SIGN (OLD-LTS)
115  125         260 C=N                     C= 13-DIGIT MANT (OLD-LTS)
116  126           1 GOSUB  DV2-13           (NEW-OLD)/(OLD-LTS)
116  127           0                         *MAINFRAME:  CN6, @0235
117                                          IN: A[X&S]= EXP&SIGN DIVDND
118                                              B= 13-DIGIT MANT DIVDND
119                                              C= 13-DIGIT DIVISOR
120                                              M[X&S]= EXP&SIGN DIVISR
121                                          ASSUME: DECMODE
122                                          OUT: C= NORM. F.P. RESULT
123                                               A[X&S]= EXP&SIGN RES
124                                               B= 13-DIG MANT RESULT
125                                          USES: A,B,C,M, ACTIVE PT
126
*    NO OVERFLOW/UNDERFLOW POSSIBLE
*       MAX "NEW-OLD"= 24 HR= 86400 SECONDS
*       MIN "OLD-LTS"= 0.01 SECONDS
*      THEREFORE MAX (NEW-OLD)/(OLD-LTS) = 8640000
*
*       MIN "NEW-OLD"= 0.01 SECONDS
*       MAX "OLD-LTS"= 1 E11
*      THEREFORE MIN (NEW-OLD)/(OLD-LTS) = 1 E-13
*    ASSUME THAT THE "OLD TIME" CANNOT EQUAL THE "LAST TIME SET"
136  130         116 C=0
137  131         460 LDI                     LOAD LOW 12 BITS OF C WITH
138  132           4 CON    4                EXPONENT OF 4 FOR 10240
139  133         530 M=C                     M= 0...004= E&S FOR 10240
140  134        1072 C=C+1  M
141  135         460 LDI                     LOAD LOW 12 BITS OF C WITH
142  136          44 CON2   2      4         C= 0 0000000001 024
143  137         274 RCR    5                C&M= 10240 IN 13-DIGIT FORM
144  140           1 GOSUB  MP2-13           (NEW-OLD)*10240/(OLD-LTS)
144  141           0                         *MAINFRAME:  CN6, @0122
145                                          IN: A[X&S]= EXP & SIGN #1
146                                              B= 13-DIGIT MANTISSA #1
147                                              C= 13-DIGIT MANTISSA #2
148                                              M[X&S]= EXP & SIGN #2
149                                          ASSUME: DECMODE
150                                          OUT: C= NORMAL F.P. RESULT
151                                               A[X&S]= EXP&SGN RESULT
152                                               B= 13-DIG MANT. RESULT
```

```
   153                                        USES: A,B,C,M, ACTIVE PT
   154
*  NO OVERFLOW SHOULD BE POSSIBLE SINCE MAX= (9 E6)*(10240)= 9 E11
*  MINIMUM= 1 E-9
   157  142          160 N=C                  N= (NEW-OLD)10240/(OLD-LTS)
   158  143            1 GOSUB  GETAF         C= F.P. ACCURACY FACTOR
   158  144            0                      *TIMER ROM:  TM0, @0445
   159  145         1240 SETDEC
   160  146         1356 ? C#0                EXISTING AF = 0 ?
   161  147           37 GOC    ADJ40  ( 152) NO, 1/X WILL WORK
   162  150          640 CLRABC              A= B= 0
   163  151           33 GOTO   ADJ42  ( 154)
   164  152 ADJ40      1 GOSUB  1/X10         INVERT OLD AF= 1/OAF
   164  153            0                      *MAINFRAME:  CN6, @0213
   165                                        IN: C= NORMAL F.P. NUMBER
   166                                        ASSUME: DECMODE
   167                                        OUT: C= NORMAL F.P. RESULT
   168                                        !! WON'T RETURN IF C=0
   169                                           (DIVIDE BY 0)!!
   170                                           A[X&S]=EXP & SIGN RESULT
   171                                           B= 13-DIGIT MANT. RESULT
   172                                        USES: A,B,C,M, ACTIVE PT
*  NO UNDERFLOW/OVERFLOW SINCE AF= 01 TO 99.9
   174  154 ADJ42   260 C=N                   C= 10240(NEW-OLD)/(OLD-LTS)
   175                                         = 1/(NEW ACCURACY FACTOR)
   176  155            1 GOSUB  AD1-10        ADD THE CORRECTIONS
   176  156            0                      *MAINFRAME:  CN6, @0007
   177                                        IN: C= NORMALIZED F.P. #1
   178                                            B= 13-DIGIT MANTISSA #2
   179                                            A[X&S]= EXP & SIGN #2
   180                                        ASSUME: DECMODE
   181                                        OUT: C= NORMAL FP RES= A+-C
   182                                             A[X&S]= EXP & SIGN RES
   183                                             B= 13-DIGIT MANT. RES
   184                                        USES: A,B,C,M, ACTIVE PT
   185
*
*  NO OVERFLOW:  MAX= 9 E11
*  NO UNDERFLOW: MIN< 1/OAF >= 0.01 SO MINIMUM DIFFERENCE = 1 E-15
*
   190  157         1356 ? C#0                RESULT = 0 ?
   191  160            1 GSUBC  1/X13         NO, INVERT TO GET RESULT AF
   191  161            1                      *MAINFRAME,  CN6, @0216
   192                                        IN: B= 13-DIGIT MANTISSA
   193                                            A[X&S]= EXPONENT & SIGN
   194                                        ASSUME: DECMODE
   195                                        OUT: C= NORMAL F.P. NUMBER
   196                                             A[X&S]= EXP & SIGN
   197                                             B= 13-DIGIT MANTISSA
   198                                        USES: A,B,C,M, ACTIVE PT
   199
   200
   201  162            1 GOSUB  SETAF0        FORMAT, ROUND & STORE AF
   201  163            0                      *TIMER ROM:  TM2, @1410
   202
*                                                              3-10-81 RSW
* ADJ100 -- SETS THE NEW HARDWARE ALARM, AND BEEPS IF THERE ARE ANY
*           PAST-DUE ALARMS
*
* IN & ASSUME: NOTHING
```

```
* OUT:  HEXMODE, P SELECTED, Q= 13
*       PERIPHERALS DISABLED (EXCEPT TIMER CHIP)
* USES: A,B.X,C,M.X, P,Q, S8, +3 SUB LEVELS, DADD, PFAD, ARITH MODE,
*       TIMER PT
*
   213                   ENTRY  ADJ100
   214  164 ADJ100 1704 CLR ST
   215  165           1 GOSUB  NXTALM        SET NEW HARDWARE ALARM
   215  166           0                      *TIMER ROM:  TM3, @1475
   216  167           1 GOSUB  SRHBUF        SEARCH FOR ALARM STACK
   216  170           0                      *TIMER ROM:  TM2, @1141
   217  171          23 GOTO   ADJ110 ( 173) (P+1) ALARM STACK FOUND
   218  172        1740 RTN                  (P+2) NO ALARMS FOUND
   219
*                                                              3-4-81 RSW
* ADJ110 -- BEEPS IF THERE ARE ANY PAST DUE ALARMS
*
* IN:     A.X= ADDRESS OF FIRST REGISTER IN TIMER BUFFER
* ASSUME: HEXMODE, P SELECTED, Q= 13
* OUT:    TIMER PT= A
*         PERIPHERALS DISABLED (EXCEPT TIMER CHIP)
* USES:   A,B,X,C, ACTIVE PT, S8, +2 SUB LEVELS, DADD, PFAD, TIMER PT
*              (NO TIMER ST)
*
   230  173 ADJ110    1 GOSUB  CHKALM        CHECK FOR PAST-DUE ALARMS
   230  174           0                      *TIMER ROM:  TM0, @1230
   231  175        1434 PT=    1
   232  176        1312 ? B#0  WPT            ANY UNDISPLAYED P-D ALARMS?
   233  177           1 GOLNC  BEEP2          YES, BEEP TWICE
   233  200           2                      *TIMER ROM:  TM3, @0176
   234  201        1740 RTN
*
*************************************************************************
* TO24H = CONVERT TO 24-HOUR FORMAT                          1-26-81 RSW
*  CONVERTS FROM 12 OR 24-HOUR USER INPUT FORMAT TO 24-HOUR FORM.
*  EXITS WITH "A" ROTATED SO THAT HOUR IS IN A.X
*
* IN:     A= #HHMMSSCC.....    WHERE "#"= 0 FOR AM OR 24-HOUR INPUT
*                                       = NON-ZERO FOR PM
*                                AND "."= DON'T CARE
* ASSUME: NOTHING
* OUT:    A= MMSSCC.....0HH    WITH HH IN 24-HOUR FORM
*         DECMODE, C.X= 012
*         PT= 1
* USES:   INPUT A[13:11]= OUTPUT A.X, C, ACTIVE PT, ARITH MODE
*           (NO ST, +0 SUB LEVELS, NO DADD, NO PFAD)
*
*
   252                   ENTRY  TO24H
   253  202 TO24H   256 AC EX                 C= #HHMMSSCC......
   254  203         674 RCR    11
   255  204         416 A=C                    A= MMSSCC......#HH
   256  205        1240 SETDEC
   257  206        1434 PT=    1
   258  207         460 LDI                    LOAD LOW 12 BITS OF C WITH
   259  210          22 CON2   1      2        @22 (TREATED AS DECIMAL 12)
   260  211        1412 ? A<C  WPT            HOUR < 12 ?
   261  212          63 GONC   T24H20 ( 220) NO
   262  213        1512 ? A#0  WPT            HOUR = 1-11 ?
   263  214          43 GONC   T24H20 ( 220) NO, HOUR = 00
```

```
   264  215         1526 ? A#0  XS            PM ?
   265  216           23 GONC   T24H20 ( 220) NO, AM
   266  217          512 A=A+C  WPT           ADD 12 HOURS
   267  220 T24H20    26 A=0    XS
   268  221         1740 RTN
*
*********************************************************************
* C=T+D0 = C= TIME + DATE                                  1-29-81 RSW
*  COMBINES TIME & DATE TO GET A POINT ON THE TIME LINE.  THE INPUT
*  DATE CAN BE= 0 TO USE CURRENT DATE.  IF A DATE IS SPECIFIED, THEN
*  S0 MUST INDICATE WHETHER IT IS FROM X OR Y (FOR ERROR CHECKING).
*
* IN:     N= H.MS NORMALIZED FLOATING POINT TIME
*              (N.S= F  TO USE CURRENT TIME)
*         M= M.DY (D.MY)  NORMALIZED FLOATING POINT DATE
*              (M= 0 TO USE CURRENT DATE)
*            IF A DATE IS GIVEN (M NON-ZERO) THEN:
*              S0= 1 (0) TO TAKE COMPARISON DATE FROM "Y" ("X")
*         C= CLOCK TIME
*            IF A DATE IS GIVEN (M NON-ZERO) THEN:
*              S0= 1 (0) TO TAKE COMPARISON DATE FROM "Y" ("X")
* ASSUME: S1= 1 (0)  DO (NOT) ADD "X" OR "Y" TO "DATE ERROR"
* OUTPUT: C= 100TH'S OF SECONDS SINCE 1/1/1900 = 00SSSSSSSSSSCC
*            WHERE "S"= SECONDS, "C"= CENTISECONDS
*            AND C[13:12]= 00 FOR DATES FROM 1/1/1900 TO 12/31/2199
*         DECMODE, Q= 13, P SELECTED
*   !!!!! DOES NOT RETURN IF THERE IS AN ERROR IN TIME OR DATE.
*
* USES:   A,B,C,M,N,R8[13:6], S8, P,Q, +2 SUB LEVELS, ARITH MODE,
*          (IF A DATE IS GIVEN, MEANING M#0, THEN ALSO: DADD, PFAD)
*                (NO TIMER CHIP ACCESS)
*
   296                 ENTRY  C=T+D
   297                 ENTRY  C=T+D0
   298  222 C=T+D   1404 S1=    0             DON'T ADD X OR Y TO
   299  223 C=T+D0   340 SEL Q                   "DATA ERROR"
   300  224         1334 PT=    13
   301  225          240 SEL P
   302  226            1 GOSUB  SDHMSC        A= DAY, HR, MIN, SEC
   302  227            0                      *TIMER ROM:  TM2, @1301
   303  230          216 B=A                  B= DDDDDDHHMMSSCC
   304  231          260 C=N                  C= H.MS TIME
   305  232         1076 C=C+1  S             ANY ENTRY IN TIME ?
   306  233          167 GOC    STMN25 ( 251) NO, USE CLOCK TIME
   307  234         1176 C=C-1  S             RESTORE "C"
   308                 LEGAL                  (CLEAR THE CARRY FLAG)
   309  235            1 GOSUB  UNNOR1        UNNORMALIZE THE TIME
   309  236            0                      *TIMER ROM:  TM3, @0022
   310  237           23 GOTO   STMN23 ( 241) (P+1) OK, A= #HHMMSSCC.....
   311  240          163 GOTO   STMN27 ( 256) (P+2) X REGISTER ERROR
   312  241 STMN23     1 GOSUB  TO24H         CONVERT HOURS TO 24-HR FORM
   312  242            0                      *TIMER ROM:  TM1, @0202
   313                                        A= MMSSCC.....0HH, PT=1,
   314                                        C.X= 012, DECMODE
   315  243          752 C=C+C  WPT           C[1:0]= 24
   316  244         1412 ? A<C  WPT           HOUR < 24 ?
   317  245          113 GONC   STMN27 ( 256) NO, ERROR
   318  246          256 AC EX
   319  247          474 RCR    8
   320  250          416 A=C                  A= .....0HHMMSSCC
```

```
   321  251 STMN25  434 PT=    8
   322  252          22 A=0    PQ            A= 000000HHMMSSCC
   323  253           1 GOSUB  HMSS40        C= 00SSSSSSSSSCC
   323  254           0                      *TIMER ROM:  TM1, @0364
   324  255          33 GOTO   STMN30 ( 260) (P+1) VALID TIME
   325  256 STMN27 1604 S0=    0              (P+2) X REGISTER ERROR
   326                                        ASSUME NO ERROR POSSIBLE
   327  257         253 GOTO   STMNER ( 304)  WHEN USING CURRENT TIME.
   328  260 STMN30  730 CM EX                 M= TIME= 00SSSSSSSSSCC
   329                                        C= M.DY (D.MY) DATE
   330  261        1356 ? C#0                 ANY ENTRY IN DATE ?
   331  262          47 GOC    STMN40 ( 266) YES
   332  263         434 PT=    8              NO, USE CURRENT DATE
   333  264         322 C=B    PQ             C= DDDDDD00000000
   334  265         323 GOTO   STMN60 ( 317) C= DAY NUM. SINCE 1/1/1900
   335  266 STMN40    1 GOSUB  C-YMDD         A= POSITIVE NORM F.P. DATE
   335  267           0                      *TIMER ROM:  TM0, @1565
   336  270           1 GOSUB  DATECK         C= DAY#= DDDDDD00000000
   336  271           0                      *TIMER ROM:  TM0, @0255
   337  272         356 BC EX                B= DAY# SINCE OCT 15, 1582
   338  273           1 GOSUB  UNNOR2         UNNORMALIZE THE DATE
   338  274           0                      *TIMER ROM:  TM3, @0023
*         !!! ASSUME NO ERROR POSSIBLE !!!!! (P+1)(P+2)
   340  275         116 C=0
   341  276         434 PT=    8
   342  277         120 LC     1
   343  300        1120 LC     9              C= 0 0000190000 000
   344  301         434 PT=    8
   345  302        1412 ? A<C  WPT            YEAR < 1900 ?
   346  303          33 GONC   STMN55 ( 306) NO, OK
   347
   348  304 STMNER    1 GOLONG TERROR         IN: S1=0 (0) DO (NOT) + X|Y
   348  305           2                      *TIMER ROM:  TM0, @0263
   349                                          IF S1= 1 THEN:
   350                                             S0= 0 (1) ADD Y (X) TO
   351                                                       "DATA ERROR"
   352                                        ASSUME: NOTHING
   353  306 STMN55  220 LC     2
   354  307         220 LC     2
   355  310         434 PT=    8
   356  311        1412 ? A<C  WPT            YEAR < 2200 ?
   357  312        1723 GONC   STMNER ( 304) NO, ERROR
   358  313         156 AB EX                 A= DDDDDD00000000, B= DATE
   359  314           1 GOSUB  115860         C= 11586000000000
   359  315           0                      *TIMER ROM:  TM2, @1440
   360  316        1116 C=A-C                 SUBTRACT 115860 DAYS= DAY#
   361                                          SINCE 1/1/1900
*
   363  317 STMN60  416 A=C                   A= DDDDDD00000000
   364  320           1 GOSUB  X20Q8          A= DAYS * 20, C= DAYS * 40
   364  321           0                      *TIMER ROM:  TM3, @0043
   365  322        1716 C SR                  C= DAYS * 4
   366  323         516 A=A+C                 A= DAYS * 24= HOURS=
   367                 LEGAL                  (CLR CARRY)  0HHHHHHH000000
   368  324           1 GOSUB  HM-SC          CONVERT HMS TO SEC & 100THS
   368  325           0                      *TIMER ROM:  TM1, @0377
   369  326        1240 SETDEC
   370  327         630 C=M                   C= TIME= 00SSSSSSSSSCC
   371  330        1016 C=A+C                 C=TIME+DATE= 00SSSSSSSSSCC
   372  331        1740 RTN
```

```
*************************************************************************
*                                                              1-12-81 RSW
* HMSSEC = HOURS, MINUTES, SECONDS TO SECONDS
*  CONVERTS A FLOATING POINT NORMALIZED H.MS NUMBER TO 100THS OF SECONDS
*
*                                (10 TO KEYBOARD CHECK INC GSB)
*                                (47 MAX TO EXIT AFTER KEY TRANSITION)
*         ERRORS (RTN TO P+2) IF:
*                    HOUR > 99
*                    MINUTES OR SECONDS > 59
*
* IN:     A= FLOATING POINT NORMALIZED H.MS TIME < 100 HOURS
*            (WITH VALID EXPONENT, BCD DIGITS)
* ASSUME: S8= 1 (0)   TO CHECK (IGNORE) KEYBOARD
*            IF S8= 1 THEN: S9= 1 (0)   RETURN ON KEY UP (DOWN)
* OUT:    IF S8=1 & S9=0, JUMPS TO "TMRKEY" ON KEY DOWN (GARBAGE OUT)
*         IF RTN TO P+1--    (NORMAL CASE)
*            IF S8= 1 AND KEY UP IS DETECTED, OUTPUT= GARBAGE !!!
*            OTHERWISE:
*             A= C= 00SSSSSSSSSCC   ( CC = CENTISECONDS )
*             P SELECTED, Q= 13, HEXMODE
*         IF RTN TO P+2--    (ERROR CASE)
*             HEXMODE
*
* USES:   A,C, P,Q, +1 SUB LEVEL, ARITH MODE
*            (NO ST, NO DADD, NO PFAD, NO TIMER CHIP ACCESS)
*
* HMSSCB - SAME AS HMSSEC EXCEPT SAVES THE SIGN OF THE NUMBER IN B.S
* HMSEC1 - SAME AS HMSSEC EXCEPT ALLOWS HOUR =< 9999, AND SETS S8=0
*              SO IT USES S8 !!!!!!!
* HMSS20 - SAME AS HMSSEC EXCEPT:
*           IN:  A= UNNORMALIZED HMS NUMBER (OUTPUT OF "UNNOR2")
*                PT= 12
*
*************************************************************************
*
  409                   ENTRY  HMSSCB
  410                   ENTRY  HMSSEC
  411                   ENTRY  HMSS20
  412  332 HMSSCB  236 B=A    S
  413  333 HMSSEC    1 GOSUB  KEYCHK        CHECK KEYBOARD IF S8= 1
  413  334            0                     *TIMER ROM:  TM1, @1466
  414  335            1 GOSUB  UNNOR2        ALLOW HOUR < 100
  414  336            0                     *TIMER ROM:  TM3, @0023
  415  337           23 GOTO   HMSS10 ( 341) (P+1) A= #HHMMSSCC.....
  416  340          143 GOTO   HMSTER ( 354) (P+2) ERROR, HOUR > 99
  417  341 HMSS10    1 GOSUB  KEYCHK        CHECK KEYBOARD IF S8= 1
  417  342            0                     *TIMER ROM:  TM1, @1466
  418  343 HMSS20 1612 A SR   WPT
  419  344        1612 A SR   WPT
  420  345         113 GOTO   HMSS35 ( 356)
  421
  422                   ENTRY  HMSEC1
  423  346 HMSEC1  460 LDI                   LOAD LOW 12 BITS OF C WITH
  424  347            3 CON    3             3 TO ALLOW HOUR < 10000
  425  350          404 S8=    0             IGNORE KEYBOARD
  426  351            1 GOSUB  UNNORM        UNNORMALIZE
  426  352            0                     *TIMER ROM:  TM3, @0025
  427  353           33 GOTO   HMSS35 ( 356) (P+1) OK
  428  354 HMSTER    1 GOLONG RTNP+2         (P+2) ERROR CASE
```

```
   428  355           2                       *TIMER ROM:  TM2, @1230
   429
   430  356 HMSS35 1612 A SR   WPT
   431  357        1612 A SR   WPT
   432  360        1612 A SR   WPT
   433  361          36 A=0    S               A= 0000HHHHMMSSCC
   434  362           1 GOSUB  KEYCHK          CHECK KEYBOARD IF S8= 1
   434  363           0                       *TIMER ROM:  TM1, @1466
*
*                                                              1-12-81 RSW
*
* HMSS40 -- CONVERTS HMS NUMBER TO 100TH'S OF SECONDS
*             ERRORS (RTN TO P+2) IF MINUTES OR SECONDS > 59
*
* IN:     A= 0000HHHHMMSSCC  (COULD HAVE 6 DIGITS OF HOURS, MAYBE MORE)
* ASSUME: NOTHING
* OUT:    SAME AS HMSSEC
* USES:   SAME AS HMSSEC
*
   446                  ENTRY  HMSS40          (40 MAX INC RTN [NOT GSB])
   447  364 HMSS40 1240 SETDEC
   448  365         116 C=0
   449  366         234 PT=    5
   450  367         620 LC     6               C= 00000000600000
   451  370         234 PT=    5
   452  371        1412 ? A<C  WPT             MINUTES < 60 ?
   453  372        1623 GONC   HMSTER ( 354)   NO, ERROR
   454  373          34 PT=    3
   455  374        1074 RCR    2               C= 00000000006000
   456  375        1412 ? A<C  WPT             SECONDS < 60 ?
   457  376        1563 GONC   HMSTER ( 354)   NO, ERROR
   458
*
*                                                              1-12-81 RSW
* HM-SC = HOURS & MINUTES TO SECONDS
*
* IN:     DECMODE, A= 00HHHHHHMMSSCC WITH MIINUTES & SECONDS < 60
* ASSUME: NOTHING
* OUT:    HEXMODE, P SELECTED, Q= 13
*         A= C= 00SSSSSSSSSSCC     (CC= CENTISECONDS)
*           (ONLY RETURNS TO P+1, NO ERROR CHECK !!!)
* USES:   SAME AS HMSSEC
*
   470                  ENTRY  HM-SC           (29 MAX INC RTN, NOT GSB)
   471  377 HM-SC   240 SEL P
   472  400         534 PT=    6               PT= LEAST SIG. HOURS DIGIT
   473  401           1 GOSUB  X20Q            Q= 13, HOURS * 20
   473  402           0                       *TIMER ROM:  TM3, @0046
   474  403         516 A=A+C                  A= 00MMMMMMMMSSCC (HR * 60)
   475  404         134 PT=    4               PT= LEAST SIG. MINS DIGIT
   476  405           1 GOSUB  X20             MINUTES * 20
   476  406           0                       *TIMER ROM:  TM3, @0051
   477  407        1016 C=A+C                  A= 00SSSSSSSSSSCC (MIN*60)
   478  410         416 A=C
   479  411        1140 SETHEX
   480  412        1740 RTN
*****************************************************************************
* DIFF= DIFFERENCE                                             2-11-81 RSW
*
* "DIFF" CALCULATES THE DIFFERENCE BETWEEN THE M[11:9] SPLIT AND THE
```

```
* PREVIOUS SPLIT AND EXITS WITH IT READY FOR DISPLAY BY "DSPTMR".
*    (IF M[11:9]= REG 0, THEN DIFF OUTPUTS THE INPUT SPLIT)
*    (IF THE DIFFERENCE IS INVALID ON THE "ENTER" KEY PATH, DIFF
*    OUTPUTS THE INPUT SPLIT)
*
* TO CALCULATE A DIFF, IT CONVERTS NEGATIVE SPLITS TO 10'S COMPLEMENT
* SSSSSSSSSSCC FORM, DOES THE SUBTRACT, AND CONVERTS BACK TO HMS.
*
* IN:      C= MOST RECENT SPLIT (H.MS FLOATING POINT NORMALIZED NUMBER)
*          M.X= REG 0 ADDRESS
*          M[11:9]= ADDRESS OF MOST RECENT SPLIT
*              (!!THIS REG MUST EXIST, OR MAY GET A "MEMORY LOST"!!)
*          B.S= SIGN OF THE MOST RECENT SPLIT
* ASSUME:  PERIPHERALS DISABLED
*          S8= 1 (0)      TO CHECK (IGNORE) KEYBOARD
*              IF S8= 1, THEN:  S9= 1 (0) RETURN ON KEY UP (DOWN)
* OUT:     IF S8=1 & S9=0 AND A KEY GOES DOWN, JUMP TO "TMRKEY"!!!!
*          IF RETURN TO (P+1): [NORMAL CASE]
*            IF S8= 1 AND KEY UP OCCURS, OUTPUT= GARBAGE!!!!!
*            IF S8= 0 OR NO KEY TRANSITION:
*              A= C= SIGNED NORMALIZED F.P. TIME      (MAY BE A SPLIT)
*              B= UNNORMALIZED TIME= #HHMMSSCC......  (MAY BE A SPLIT)
*              HEXMODE, P SELECTED
*          IF RETURN TO (P+2): [CURR SPLIT NOT VALID, OR DIFF >= 100 HR]
*              HEXMODE
* USES:    A,B,C, P,Q,  +2 SUB LEVELS, DADD, ARITH MODE
*          (NO ST, NO PFAD, NO TIMER CHIP ACCESS)
*
   513                  ENTRY  DIFF
   514  413 DIFF    416 A=C
   515  414           1 GOSUB  HMSSCB        C= SSSSSSSSSSSSCC
   515  415           0                      *TIMER ROM:  TM1, @0332
   516  416          23 GOTO   DIFF20 ( 420) (P+1)
   517  417 DIFFEX 1353 GOTO   HMSTER ( 354) (P+2) ERROR
   518  420 DIFF20    1 GOSUB  KEYCHK        CHECK KEYBOARD IF S8= 1
   518  421           0                      *TIMER ROM:  TM1, @1466
   519  422         336 C=B    S              RESTORE THE SIGN
   520  423         160 N=C                   N= UNCODED TIMER TIME
   521  424         630 C=M                   C.X= REG 0 ADDRESS
   522  425         406 A=C    X
   523  426        1174 RCR    9              C.X= ACTIVE REG ADDRESS
   524  427        1546 ? A#C  X              OK TO USE NEXT LOWER REG ?
   525  430          47 GOC    DIFF30 ( 434) YES
   526  431 DIFF25  260 C=N                   NO, SHOW SPLIT, NOT DIFF.
   527  432         416 A=C
   528  433         453 GOTO   DIFF65 ( 500)
   529  434 DIFF30 1146 C=C-1  X
   530  435        1160 DADD=C
*  !!!! IF THIS REGISTER DOESN'T EXIST, MAY GET A "MEMORY LOST"
   532  436           1 GOSUB  P6RTN         MASSAGE REGISTER CONTENTS
   532  437           0                      *MAINFRAME:  CN5, @1160
   533                                       IN: C.X= REGISTER ADDRESS
   534                                       ASSUME: C.X REG ENABLED
   535                                          PERIPHERALS DISABLED
   536                                       OUT: C.X PRESERVED, HEXMODE
   537                                          B=LEGALIZED C.X REG CONT
   538                                       USES: A,B,C, ACTIVE PT,
   539                                          DADD, (+0 SUB LEVELS)
   540
   541  440         316 C=B                   C= REGISTER CONTENTS
```

```
   542  441           1 GOSUB  KEYCHK        CHECK KEYBOARD IF S8= 1
   542  442           0                      *TIMER ROM:  TM1, @1466
   543  443        1376 ? C#0  S              NEGATIVE OR ALPHA ?
   544  444          43 GONC   DIFF35 ( 450) NO, POSITIVE NUMBER
   545  445        1240 SETDEC
   546  446        1076 C=C+1  S              NEGATIVE NUMBER?
   547  447          53 GONC   DIFF38 ( 454) NO, ALPHA DATA
   548  450 DIFF35  156 AB EX                A= H.MS TIME OF PREV. SPLIT
   549  451           1 GOSUB  HMSSEC         A= SSSSSSSSSSSSCC
   549  452           0                      *TIMER ROM:  TM1, @0333
   550  453          43 GOTO   DIFF40 ( 457) (P+1)
   551  454 DIFF38 1114 ?S9=1                (P+2) ENTER KEY PATH ?
   552  455        1547 GOC    DIFF25 ( 431) YES, DISPLAY SPLIT BECAUSE
   553                                       THE DIFFERENCE IS NOT VALID
   554  456        1413 GOTO   DIFFEX ( 417) ERROR
   555
   556  457 DIFF40    1 GOSUB  KEYCHK        CHECK KEYBOARD IF S8= 1
   556  460           0                      *TIMER ROM:  TM1, @1466
   557  461         336 C=B    S              C= PREVIOUS SPLIT
   558  462        1240 SETDEC
   559  463        1534 PT=    12
   560  464        1376 ? C#0  S              NEGATIVE ?
   561  465          23 GONC   DIFF50 ( 467) NO
   562  466        1212 C=-C   WPT            YES, MAKE 10'S COMPLEMENT
   563  467 DIFF50  416 A=C                   A= PREVIOUS SPLIT
   564  470         260 C=N                   C= MOST RECENT SPLIT
   565  471        1376 ? C#0  S              NEGATIVE ?
   566  472          23 GONC   DIFF60 ( 474) NO
   567  473        1212 C=-C   WPT            MAKE IT 10'S COMPLEMENT
   568  474 DIFF60  256 AC EX                 A= CURRENT, C=PREV. SPLIT
   569  475         716 A=A-C                 DIFF= CURRENT - PREVIOUS
   570  476        1536 ? A#0  S              NEGATIVE RESULT ?
   571  477        1557 GOC    DIFF38 ( 454) YES, BAD DATA
   572  500 DIFF65  236 B=A    S
   573  501           1 GOLONG GTMR30         STOPWATCH TIME TO H.MS TIME
   573  502           2                      *TIMER ROM:  TM0, @0510
*******************************************************************
* RSTKBT -- TIMER RESET KEYBOARD                          2-24-81 RSW
*
* IN & ASSUME: HEXMODE
* USES: C.X ONLY
*
   580                 ENTRY  RSTKBT
   581  503 RSTKBT  460 LDI                   LOAD LOW 12 BITS OF C WITH
   582  504         172 CON    122            DECIMAL 122 FOR COUNTDOWN
   583  505        1146 C=C-1  X              DO 40 MSEC DOWN DEBOUNCE
   584  506        1773 GONC   *-1    ( 505)
   585  507           1 GOLONG RSTKB          CLEAR KEYBOARD
   585  510           2                      *MAINFRAME:  CN0, @0230
   586                                       IN:     HEXMODE
   587                                       ASSUME: HEXMODE
   588                                       USES:   C.X ONLY
*******************************************************************
*                                                         1-5-81 RSW
* CALCRA = CALCULATE NEWW ACTIVE (STO/RCL) REGISTER ADDRESS
*
* FETCHES THE REGISTER NUMBER FROM "MM" AND COMPUTES THE NEW ACTIVE
* REGISTER ADDRESS.
*
* IN:     M.X= REG 0 ADDRESS.  M[5:3]= STO REG NUMBER (BCD)
```

```
*         M[8:6]= RCL REG NUMBER (BCD)
* ASSUME: HEXMODE
* OUT:    M[11:9]= NEW ACTIVE REGISTER ADDRESS
* USES:   A.X, C, M[11:9], 2 ADDITIONAL SUB LEVELS
*           (NO PT, NO DADD, NO PFAD, NO TIMER CHIP ACCESS)
*
*
* R-TO-S= RCL TO STO:  SAME AS CALCRA EXCEPT SETS TO STO MODE (S2=0)
*                      SO IT USES S2 !!!!!
*
* CALCRC: SAME AS CALCRA EXCEPT EXPECTS C= M REGISTER CONTENTS
*
   609                   ENTRY  R-TO-S
   610                   ENTRY  CALCRA
   611                   ENTRY  CALCRC
   612  511 R-TO-S 1004 S2=    0             SWITCH OUT OF RCL
   613  512           4 S3=    0             DON'T SUPPRESS REG NUMBER
   614  513 CALCRA  630 C=M
   615  514 CALCRC    1 GOSUB  GETR#         C.X= ACTIVE REGISTER NUMBER
   615  515           0                      *TIMER ROM:  TM1, @1507
   616  516         132 C=0    M             C[4:3]= 00 FOR "GOTINT"
   617  517           1 GOSUB  GOTINT        CONVERT IT INTO BINARY
   617  520           0                      *MAINFRAME:  CN0, @1370
   618                                       IN: C.X= BCD NUMBER
   619                                           C[4:3]= 00
   620                                       ASSUME: HEXMODE
   621                                       OUT: C.X= BINARY NUMBER
   622                                       USES: A.X, C, +1 SUB LEVEL
   623                                        (NO: ST, PT, DADD, PFAD)
   624  521         406 A=C    X             A.X= BINARY REGISTER NUMBER
   625  522         630 C=M                  C.X= REG 0 ADDRESS
   626  523         506 A=A+C  X             A.X= NEW ACTIVE REG ADDRESS
   627  524        1174 RCR    9
   628  525         246 AC EX  X             C.X= NEW ACTIVE REG ADDRESS
   629  526         274 RCR    5
   630  527         530 M=C
   631  530        1740 RTN
*
*******************************************************************
*                                                        1-15-81 RSW
* GETM.X = GET REGISTER WHOSE ADDRESS IS IN M.X
*
* IN:     M.X= REGISTER ADDRESS
* ASSUME: PERIPHERALS DISABLED
* OUT:    C= REGISTER CONTENTS, THAT REGISTER IS ENABLED
* USES:   C, DADD         (NO PT, NO ST, +0 SUB LEVELS)
*         (NO ARITH MODE, NO PFAD)
*
   643                   ENTRY  GETMXP
   644                   ENTRY  GETM.X
   645  531 GETMXP 1434 PT=    1
   646  532 GETM.X  630 C=M                  C.X= REGISTER ADDRESS
   647  533        1160 DADD=C
   648  534          70 C=DATA
   649  535        1740 RTN
*******************************************************************
*                    STOPWATCH                           2-24-81 RSW
*
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*
```

```
*           S7=  1  (0)     SHIFT (NOT) SET
*           S6=  1  (0)     (NOT) ENTERING REGISTER NUMBER
*           S5              SCRATCH  [SUPPRESS AM/PM IN TIME DISPLAY]
*           S4=  1  (0)     (2)/3 DIGIT REG NUMBER
*           S3=  1  (0)     DON'T (D0) DISPLAY REGISTER NUMBER
*           S2=  1  (0)     RCL  (STO)  MODE
*           S0=  1  (0)     DIFFERENCE/(SPLIT)  MODE
*
* POINTERS WHICH ARE PRESERVED THROUGHOUT THE STOPWATCH:
*           M.X=     REG 0 ADDRESS
*           M[5:3]=  STO REGISTER NUMBER
*           M[8:6]=  RCL REGISTER NUMBER
*           M[11:9]= ACTIVE (STO/RCL) REGISTER ADDRESS
*           M[12]=   CURRENT TENTH OF SECOND
*
   670  536          227 CON    @227          W
   671  537           23 CON    @23           S
   672                   ENTRY  SW
   673  540 SW      1570 C=REGN 13
   674  541           74 RCR    3             C.X= REG 0 ADDRESS
   675  542          346 BC EX  X             B.X= REG 0 ADDRESS
   676  543          116 C=0
   677  544          306 C=B    X
   678  545         1174 RCR    9
   679  546          306 C=B    X
   680  547         1074 RCR    2
   681  550          356 BC EX
   682  551            1 GOSUB  INITMR        INIT TIMER IF NECESSARY
   682  552            0                      *TIMER ROM:  TM0, @1524
   683  553          170 RCTIME               READ & START HOLDING COUNT
   684  554          416 A=C
   685  555         1434 PT=    1
   686  556          202 B=A    PT            SAVE CURR TENTH OF SECOND
   687  557         1376 ? C#0  S             SW TIME COMPLEMENTED ?
   688  560           47 GOC    TM20   ( 564) YES, "SETSW" SCREENS LEGAL
   689  561            1 GOSUB  36000         C= 00000036000000
   689  562            0                      *TIMER ROM:  TM2, @0454
*   NOTE: IF THE STOPWATCH TIME HAS BEEN SCRAMBLED BUT NEITHER THE
*         TIMER WARM START CONSTANT NOR THE HARDWARE POWER UP BIT
*         DETECTED A PROBLEM, THE KEYBOARD COULD BE LOCKED OUT FOR
*         UP TO 15 MINUTES HERE AND ON EVERY ACCESS TO THE STOPWATCH
*         TIME.
*
   696  563         1016 C=A+C               TIME MUST BE < 100 HOURS
   697  564 TM20     150 WDTIME              CORRECT TIME EXACTLY
*   NOTE: FOR TIMES > 88 DAYS, WDTIME WILL WRITE A TIME THAT IS SLIGHTLY
*         IN ERROR.  THE ERROR WILL BE (1+N)/100 SECONDS, WHERE
*           N= INT([(HOURS SINCE LAST USE OF SW) - 2100]/3000)
*         THIS IS AN ERROR OF ABOUT (1/100 SEC)/(3000 HR)(3600 SEC/HR)
*         OR ABOUT 0.001 PPM WHICH SHOULD NOT BE NOTICEABLE COMPARED
*         WITH THE TIMEBASE ERROR.
*
   705  565         1140 SETHEX
   706  566         1704 CLR ST
   707  567          316 C=B
   708  570           74 RCR    3
   709  571          530 M=C                  M.X= REG 0 ADDRESS
   710                                        M[5:3]=  STO REG NUMBER= 0
   711                                        M[8:6]=  RCL REG NUMBER= 0
   712                                        M[11:9]= ACTV REG ADDR=M.X
```

```
713                                          M[12]=   CURR TENTH OF SEC
714  572           1 GOSUB  CLRALM           CLEAR EXTRANEOUS ALARMS
714  573           0                         *TIMER ROM:  TM2, @1456
715
716                  ENTRY  TMR00
717                  ENTRY  TMR01
718  574 TMR00       1 GOSUB  RSTKB           CLEAR KEYBOARD
718  575           0                         *MAINFRAME:  CN0, @0230
719                                          IN: HEXMODE
720                                          ASSUME: HEXMODE
721                                          USES: C.X ONLY
722                                          TIME: 44 MIN (INC GSB&RTN)
723  576 TMR01    1714 CHK KB                ANOTHER KEY DOWN ?
724  577           1 GOLC   TMRKEY           YES
724  600           3                         *TIMER ROM:  TM1, @0773
725  601         410 S8=    1                CHECK KEYBOARD
726  602        1014 ?S2=1                   RCL MODE ?
727  603         177 GOC    TMR10  ( 622)    YES
728  604        1104 S9=    0                RETURN ON KEY DOWN
729  605           1 GOSUB  GETMR            B=UNNORMALIZED TIMER TIME
729  606           0                         *TIMER ROM:  TM0, @0471
730  607        1714 CHK KB                  KEY DOWN ?
731  610         717 GOC    TMRCHK ( 701)    YES
732  611         340 SEL Q
733  612         134 PT=    4                DISPLAY TENTHS OF SECONDS
734  613           1 GOSUB  HWSTS            PUT UP HARDWARE STATUS
734  614           0                         *TIMER ROM:  TM3, @0255
735  615        1214 ?S7=1                   RUNNING?
736  616          27 GOC    TMR07  ( 620)    YES
737  617          34 PT=    3                NO, DISPLAY 100TH'S
738  620 TMR07  1730 CST EX                  RESTORE TIMER STATUS
739  621         523 GOTO   TMR24  ( 673)
740
741  622 TMR10   106 C=0    X
742  623        1760 PFAD=C                  DISABLE PERIPHS (DISPLAY)
743  624         630 C=M
744  625        1174 RCR    9                C.X= RCL REG ADDRESS
745  626           1 GOSUB  CHKADR           ERROR FOR INVALID ADDRESS
745  627           0                         *MAINFRAME:  CN5, @1156
746                                          IN: C.X= REGISTER ADDRESS
747                                          ASSUME: PERIPHS DISABLED
748                                          OUT: B=MASSAGED REG CONTENT
749                                               HEXMODE, S9= 1
750                                          USES: A,B,C, S9, ACTIVE PT
751                                               DADD, ARITH MODE
752                                               (+0 SUB LEVELS)
753
754  630        1104 S9=    0                RETURN ON KEY DOWN
755  631         316 C=B                     C= MASSAGED REG CONTENTS
756  632        1714 CHK KB                  KEY DOWN ?
757  633         467 GOC    TMRCHK ( 701)    YES
758  634        1376 ? C#0  S                NEGATIVE OR ALPHA ?
759  635          53 GONC   TMR12  ( 642)    NO, POSITIVE NUMBER
760  636        1240 SETDEC
761  637        1076 C=C+1  S
762  640        1176 C=C-1  S                NEGATIVE NUMBER ?
763  641         123 GONC   TNRDE  ( 653)    NO, ALPHA DATA
764  642 TMR12  1614 ?S0=1                   IN DIFFERENCE MODE ?
765  643          53 GONC   TMR15  ( 650)    NO, SPLIT MODE
766  644           1 GOSUB  DIFF             CALCULATE DIFFERENCE
```

```
766  645           0                          *TIMER ROM:  TM1, @0413
767  646         213 GOTO   TMR18  ( 667) (P+1)
768  647          43 GOTO   TMRDE  ( 653) (P+2) "DATA ERROR"
769  650 TMR15     1 GOSUB  UNNOR1         UNNORMALIZE THE SPLIT
769  651           0                          *TIMER ROM:  TM3, @0022
770  652         143 GOTO   TMR17  ( 666) (P+1) LEGAL SIZE SPLIT
771  653 TMRDE   404 S8=    0               (P+2) ERROR, DON'T PRINT
772  654        1140 SETHEX
773  655           1 GOSUB  MSGA           PUT "DATA ERROR" IN DISPLAY
773  656           0                          *MAINFRAME:  CN7, @0154
774  657           0 XDEF   MSGDE          "DATA ERROR"
775                                         IN: S8= 1 SET MSG FLAG&PRT
776                                             S8= 0 JUST DISPLAY
777                                         ASSUME: HEXMODE
778                                         OUT: CHIP 0 ENABLED
779                                         USES: FOR S8= 0, A,C,
780                                               ACTIVE PT,
781                                               +1 SUB LEVEL, DADD
782                                               PFAD, ARITH MODE
783  660         410 S8=    1               CHECK KEYBOARD
784  661           1 GOSUB  ENLCD           ENABLE DISPLAY, DISABLE RAM
784  662           0                          *MAINFRAME:  CN1, @1766
785                                         IN & ASSUME: NOTHING
786                                         OUT: DISPLAY ENABLED,
787                                              RAM DISABLED
788                                         USES: C.X, DADD, PFAD ONLY
789  663           1 GOSUB  REG#            SHIFT REG # LEFT INTO DISP.
789  664           0                          *TIMER ROM:  TM1, @1522
790  665         143 GOTO   TMPCHK ( 701)
791  666 TMR17   216 B=A                    B= UNNORMALIZED SPLIT
792  667 TMR18  1714 CHK KB                 KEY DOWN ?
793  670         117 GOC    TMRCHK ( 701)
794  671         340 SEL Q
795  672          34 PT=    3               RCL MODE, DISPLAY 100TH'S
796  673 TMR24     1 GOSUB  CHKLB           CHECK LOW BATTERY
796  674           0                          *TIMER ROM:  TM0, @1510
797  675        1714 CHK KB                 KEY DOWN ?
798  676           1 GSUBNC DSPTMR          NO, DISP W/R# IF NECESSARY
798  677           0                          *TIMER ROM:  TM1, @1575
799  700         240 SEL P
800
801                 ENTRY  TMRCHK
802  701 TMRCHK 1714 CHK KB                 KEY DOWN ?
803  702         717 GOC    TMRKEY ( 773) YES
804  703        1140 SETHEX
805  704           1 GOSUB  ENTMR           ENABLE TIMER, DISABLE RAM
805  705           0                          *TIMER ROM:  TM0, @0342
806  706        1554 ALARM?
807  707         233 GONC   TMR70  ( 732) NO
808  710         370 RDSTS               C= HARDWARE STATUS
809  711        1634 PT=    0
810  712         742 C=C+C  PT              TIMER COUNTED THROUGH 0 ?
811  713         173 GONC   TMR70  ( 732) NO
812  714           1 GOSUB  BEEPK           BEEP
812  715           0                          *TIMER ROM:  TM3, @0222
813  716        1714 CHK KB                 KEY DOWN ?
814  717           1 GSUBNC BEEPKP          NO, BEEP
814  720           0                          *TIMER ROM:  TM3, @0207
815  721        1714 CHK KB                 KEY PRESSED DURING BEEP ?
816  722         517 GOC    TMRKEY ( 773) YES, BEEP ABORT, LEAVE ALM
```

```
 817  723           1 GOSUB  ENTMR          ENA TMR, DISABLE RAM, PT= 6
 817  724           0                       *TIMER ROM:  TM0, @0342
 818  725         460 LDI                   DON'T CLEAR ALM UNTIL SURE
 819  726          61 CON    @61              OF SOUNDING 2 BEEPS
 820  727         350 WRSTS                 CLR DTZ B, DTZ A, ALM B
 821  730           1 GOLONG TMR01          UPD LCD TO RMV OLD TENTHS
 821  731           2                       *TIMER ROM:  TM1, @0576
 822  732 TMR70  1434 PT=    1
 823  733         370 RDSTS                 C= HARDWARE STATUS
 824  734         752 C=C+C  WPT            RUNNING ?
 825  735        1443 GONC   TMRCHK ( 701)  NO
 826  736        1650 PT=B                  SELECT STOPWATCH CLOCK
 827  737          70 RDTIME                C= STOPWATCH TIME
 828  740         416 A=C
 829  741         630 C=M
 830  742         674 RCR    11             C[1]= LAST TENTH OF SECOND
 831  743        1542 ? A#C  PT             TIME TO UPDATE DISPLAY ?
 832  744        1353 GONC   TMRCHK ( 701)  NO
 833  745        1714 CHK KB                KEY DOWN ?
 834  746         257 GOC    TMRKEY ( 773)  YES
 835  747         242 C=A    PT
 835  750         402                       (INSERTED BY ASSEMBLER)
 836  751          74 RCR    3
 837  752         530 M=C                   M[12]= CURR TENTH OF SECOND
 838  753        1014 ?S2=1                 IN RCL MODE ?
 839  754        1257 GOC    TMRCHK ( 701)  YES, DON'T CHANGE DISPLAY
 840  755         410 S8=    1              RETURN ON KEY DOWN
 841  756        1104 S9=    0
 842  757         256 AC EX                 C= TIMER TIME
 843  760           1 GOSUB  GETMRC         CONVERT TIME TO HMS
 843  761           0                       *TIMER ROM:  TM0, @0475
 844  762        1714 CHK KB                KEY DOWN ?
 845  763         107 GOC    TMRKEY ( 773)  YES
 846  764           1 GOSUB  CHKLB          CHECK LOW BATTERY
 846  765           0                       *TIMER ROM:  TM0, @1510
 847  766         340 SEL Q
 848  767         134 PT=    4
 849  770           1 GOSUB  DSPTMM         DISPLAY TIMER TIME
 849  771           0                       *TIMER ROM:  TM1, @1604
 850  772        1073 GOTO   TMRCHK ( 701)
 851
 852
 853
 854                ENTRY  TMRKEY
 854  773 TMRKEY 1140 SETHEX
 855  774         106 C=0    X
 857  775        1760 PFAD=C                DISABLE PERIPHERALS
 858  776         460 LDI                   LOAD LOW 12 BITS OF C WITH
 859  777          23 CON    19             DECIMAL 19 FOR BRANCH
 860 1000           1 GOSUB  KEY-FC         BRANCH TO FUNCTIONS
 860 1001           0                       *TIMER ROM:  TM2, @0434
*       EXECUTION TIME= 7(KEY) + 12  INCLUDING GSB & RTN
 862 1002         207 CON2   8      7       R/S
 863 1003          23 CON2   1      3       ENTER
 864 1004         204 CON2   8      4       "9"
 865 1005         164 CON2   7      4       "8"
 866 1006          64 CON2   3      4       "7"
 867 1007         205 CON2   8      5       "6"
 868 1010         165 CON2   7      5       "5"
 869 1011          65 CON2   3      5       "4"
```

```
   870 1012          206 CON2   8       6       "3"
   871 1013          166 CON2   7       6       "2"
   872 1014           66 CON2   3       6       "1"
   873 1015           67 CON2   3       7       "0"
   874 1016           30 CON2   1       8       "OFF" KEY
   875 1017          202 CON2   8       2       "RCL"
   876 1020          302 CON2   12      2       "SST"
   877 1021           22 CON2   1       2       "SHIFT"
   878 1022          203 CON2   8       3       "EEX" KEY
   879 1023          303 CON2   12      3       BACK ARROW
   880 1024          163 CON2   7       3       "CHS" KEY
   881 1025            0 CON    0
   882 1026          443 GOTO   TMR/S   (1072)  R/S
   883 1027          413 GOTO   TMENT   (1070)  ENTER
   884 1030         1066 C=C+1  XS
   885 1031         1066 C=C+1  XS
   886 1032         1066 C=C+1  XS
   887 1033         1066 C=C+1  XS
   888 1034         1066 C=C+1  XS
   889 1035         1066 C=C+1  XS
   890 1036         1066 C=C+1  XS
   891 1037         1066 C=C+1  XS
   892 1040         1066 C=C+1  XS
   893                   LEGAL                  (CLEAR THE CARRY FLAG)
   894 1041          423 GOTO   ADENT   (1103)  ADDRESS ENTRY
   895 1042          513 GOTO   TMROFF  (1113)  0FF
   896 1043          443 GOTO   TMRCL   (1107)  RCL
   897 1044          463 GOTO   TMSST   (1112)  SST
   898 1045          773 GOTO   TMSHF   (1144)  SHIFT
   899 1046          373 GOTO   TMEEX   (1105)  EEX
   900 1047          553 GOTO   TMRBAK  (1124)  BACK ARROW
   901 1050          413 GOTO   TMCHS   (1111)
*.....................................................................
   903
   904                   ENTRY  TMR00K
   905 1051 TMR00K   460 LDI                    LOAD LOW 12 BITS OF C WITH
   906 1052          156 CON    110             DEBOUNCE TIME COUNTER
   907 1053         1146 C=C-1  X               WAIT FOR KEY DOWN DEBOUNCE
   908 1054         1773 GONC   *-1     (1053)   (MIN OF 40 MILLISECONDS)
   909 1055            1 GOLONG TMR00           CHECK FOR KEY DOWN
   909 1056            2                        *TIMER ROM:  TM1, @0574
*.....................................................................
   911
*********************************************************************
* CLOCK                                                    6-30-81 RSW
*********************************************************************
   915 1057          213 CON    @213            K
   916 1060            3 CON    @03             C
   917 1061           17 CON    @17             O
   918 1062           14 CON    @14             L
   919 1063            3 CON    @03             C
   920                   ENTRY  CLOCK
   921 1064 CLOCK   1204 S7=     0
   922 1065            1 GOSUB  TGLSHF          TURN ON SHIFT ANNUNCIATOR
   922 1066            0                        *TIMER ROM:  TM0, @1471
   923 1067          313 GOTO   TOFF    (1120)
   924
   925 1070 TMENT      1 GOLONG TMRENT          READ TIME AFTER "ENTER" KEY
   925 1071            2                        *TIMER ROM:  TM1, @1346
   926 1072 TMR/S    673 GOTO   TMRR/S (1161)   HANDLE R/S KEY
```

```
   927
   928 1073 TMREX     1 GOSUB  TGLSHF         TURN OFF SHIFT ANNUNCIATOR
   928 1074           0                       *TIMER ROM:  TM0, @1471
   929
   930                  ENTRY  TMEXIT
   931 1075 TMEXIT    1 GOSUB  RSTKBT         CLEAR KEYBOARD
   931 1076           0                       *TIMER ROM:  TM1, @0503
   932 1077           1 GOSUB  ENCP00         ENA CHIP 0, DIS PERIPHERALS
   932 1100           0                       *MAINFRAME:  CN2, @0522
   933
   934 1101           1 GOLONG CLDSP          CLR MSG FLAG, PUT UP GOOSE
   934 1102           2                       *MAINFRAME:  CN4, @0340
   935                                         (FOR RUNNING PROGRAM CASE)
   936                                        IN: CHIP 0 ENABLED
   937                                            PERIPHERALS DISABLED
   938                                        OUT: NO RETURN, GOTO NFRPU
   939 1103 ADENT     1 GOLONG ADRENT         ADDRESS ENTRY ROUTINE
   939 1104           2                       *TIMER ROM:  TM1, @1304
   940 1105 TMEEX     1 GOLONG TMREEX         HANDLE EEX KEY ENTRY
   940 1106           2                       *TIMER ROM:  TM1, @1256
   941 1107 TMRCL     1 GOLONG TMRRCL         SWITCH TO RECALL MODE
   941 1110           2                       *TIMER ROM:  TM1, @1241
   942 1111 TMCHS   673 GOTO   TMRCHS (1200)
   943 1112 TMSST   763 GOTO   TMRSST (1210)
*
* TMROFF -- SET UP FOR JUMP TO "OFF"
*
* IN: HEXMODE      (DOES NOT RETURN)
*
   949                  ENTRY  TMROFF
   950 1113 TMROFF    1 GOSUB  TMRSTS         READ TIMER HARDWARE STATUS
   950 1114           0                       *TIMER ROM:  TM3, @0244
   951 1115           1 GOSUB  CLRALS         STOP CLOCK MODE
   951 1116           0                       *TIMER ROM:  TM2, @1451
   952
   953                  ENTRY  CLKOFF
   954 1117 CLKOFF  240 SEL P
   955 1120 TOFF      1 GOSUB  LDSST0         PUT UP STATUS SET 0
   955 1121           0                       *MAINFRAME:  CN1, @1627
   956                                        IN & ASSUME: NOTHING
   957                                        OUT: S0-S7= STATUS SET 0
   958                                         C= REG 14, CHIP 0 ENABLED
   959                                         PERIPHERALS DISABLED
   960                                        USES: C, S0-S7, DADD, PFAD
   961                                          (NO PT, +0 SUB LEVELS)
   962 1122           1 GOLONG OFF            POWER OFF
   962 1123           2                       *MAINFRAME:  CN4, @0710
   963                                        IN: CHIP 0 ENABLED, PERIPH
   964                                            DISABLED, STATUS SET 0
   965                                            UP, P SELECTED, HEXMODE
   966
   967
   968 1124 TMRBAK 1214 ?S7=1                 SHIFT SET ?
   969 1125        1467 GOC    TMREX  (1073)  YES, EXIT "SW"
   970 1126         514 ?S6=1                 ENTERING ADDRESS ?
   971 1127         163 GONC   TMBK20 (1145)  NO
   972 1130           1 GOSUB  GETR#M         C.X= STO/RCL REG NUMBER
   972 1131           0                       *TIMER ROM:  TM1, @1506
   973 1132         406 A=C    X
   974 1133        1434 PT=    1
```

```
   975 1134          542 A=A+1  PT            A[1]= DIGIT ?
   976 1135           23 GONC   TMBK10 (1137) YES
   977 1136         1034 PT=    2             NO, SO A.XS= LAST DIGIT
   978 1137 TMBK10   102 C=0    PT            "F" OUT LAST DIGIT
   979 1140         1142 C=C-1  PT
   980 1141            1 GSUBC  PUTR#         !!! WILL ALWAYS CARRY !!!
   980 1142            1                      *TIMER ROM:  TM1, @1514
   981 1143 TMROKJ 1063 GOTO   TMR00K (1051)
   982
   983 1144 TMSHF   723 GOTO   TMRSHF (1236)
   984
   985 1145 TMBK20    1 GOSUB  ENTMR         ENABLE TIMER CHIP, PT=A
   985 1146           0                      *TIMER ROM:  TM0, @0342
   986 1147         370 RDSTS                C= HARDWARE STATUS
   987 1150        1434 PT=    1
   988 1151        1014 ?S2=1               RCL MODE ?
   989 1152         777 GOC    TMRC15 (1251) YES, SWITCH TO STO MODE
   990 1153         752 C=C+C  WPT           RUNNING ? (?S7=1)
   991 1154        1677 GOC    TMR0KJ (1143) YES, IGNORE THE KEY
   992 1155         116 C=0                  NOT RUNNING
   993 1156        1650 PT=B
   994 1157          50 WRTIME              CLEAR THE STOPWATCH
   995 1160        1633 GOTO   TMR0KJ (1143)
   996
   997
*.....................................................................
   999
  1000 1161 TMRR/S  514 ?S6=1               ENTERING ADDRESS ?
  1001 1162        1617 GOC    TMR0KJ (1143) YES, IGNORE THE KEY
  1002 1163           1 GOSUB  HWSTS         PUT UP HARDWARE STATUS
  1002 1164           0                      *TIMER ROM:  TM3, @0255
  1003 1165        1650 PT=B                 SELECT TIMER/STOPWATCH CLK
  1004 1166        1214 ?S7=1                RUNNING ?
  1005 1167          33 GONC   TMRS20 (1172) NO
  1006 1170        1450 STOPC                YES, STOP
  1007 1171          23 GOTO   TMRS30 (1173)
  1008
  1009 1172 TMRS20 1550 STARTC               START TIMER/STOPWATCH
  1010 1173 TMRS30 1530 ST=C                 RESTORE STATUS
  1011 1174        1014 ?S2=1                IN RCL MODE ?
  1012 1175           1 GSUBC  R-TO-S        YES, SWITCH TO STO MODE
  1012 1176           1                      *TIMER ROM:  TM1, @0511
  1013 1177         363 GOTO   CLRSHF (1235) BE SURE "SHIFT" IS OFF
  1014
*.....................................................................
  1016
  1017 1200 TMRCHS 1214 ?S7=1               SHIFT SET ?
  1018 1201        1427 GOC    TMR0KJ (1143) YES, IGNORE THE KEY
  1019 1202        1614 ?S0=1                DIFFERENCE MODE ?
  1020 1203          37 GOC    TMRCH3 (1206) YES
  1021 1204        1610 S0=    1             NO, GO TO DIFFERENCE MODE
  1022 1205         473 GOTO   TMRC30 (1254)
  1023 1206 TMRCH3 1604 S0=    0             GO TO SPLIT MODE
  1024 1207         453 GOTO   TMRC30 (1254)
  1025
*.....................................................................
  1027
  1028 1210 TMRSST  514 ?S6=1               ENTERING ADDRESS ?
  1029 1211        1327 GOC    TMR0KJ (1143) YES, IGNORE THE KEY
  1030 1212         460 LDI                  LOAD LOW 12 BITS OF C WITH
```

```
  1031 1213           62 CON    50             50 DECIMAL
  1032 1214            1 GOSUB  GTR#MC         C.X= REG NUMBER
  1032 1215            0                       *TIMER ROM:  TM1, @1505
  1033 1216         1240 SETDEC
  1034 1217         1214 ?S7=1                 SHIFT SET ?
  1035 1220           57 GOC    TMSST2 (1225) YES, DO A BST
  1036 1221 TMSST1   646 A=A-1  X              WAIT 100 WORD TIMES FOR
  1037 1222         1773 GONC   TMSST1 (1221)  40 MSEC TOT DOWN DEBOUNCE
  1038 1223            1 GOLONG TENT35         DO A SST
  1038 1224            2                       *TIMER ROM:  TM1, @1420
  1039 1225 TMSST2 1146 C=C-1  X              DO A BST
  1040 1226           23 GONC   TMSST4 (1230)
  1041 1227          106 C=0    X              CAN'T BST, AT REG 0 ALREADY
  1042
  1043 1230 TMSST4 1140 SETHEX
  1044 1231            1 GOSUB  PUTR#          UPDATE REG NUMBER
  1044 1232            0                       *TIMER ROM:  TM1, @1514
  1045 1233 TMSST6     1 GOSUB  CALCRC         CALC NEW ACTIVE REG ADDRESS
  1045 1234            0                       *TIMER ROM:  TM1, @0514
  1046 1235 CLRSHF 1210 S7=     1
  1047
*    FALLS INTO TMRSHF TO CLEAR SHIFT ANNUNCIATOR
*.....................................................................
  1050
  1051                  ENTRY  TMRSHF
  1052       TMRSHF
  1053 1236 TMR00S     1 GOSUB  TGLSHF         TOGGLE SHIFT ANNUNCIATOR
  1053 1237            0                       *TIMER ROM:  TM0, @1471
  1054 1240          253 GOTO   TMRKJ1 (1265)
  1055
*.....................................................................
  1057
  1058                  ENTRY  TMRRCL
  1059 1241 TMRRCL 1214 ?S7=1                 SHIFT SET ?
  1060 1242          237 GOC    TMRKJ1 (1265) YES, IGNORE THE KEY
  1061 1243          514 ?S6=1                ENTERING ADDRESS ?
  1062 1244          217 GOC    TMRKJ1 (1265) YES, IGNORE THE KEY
  1063 1245         1014 ?S2=1                IN RCL MODE ?
  1064 1246           37 GOC    TMRC15 (1251) YES
  1065 1247         1010 S2=    1             NO, SWITCH TO RCL MODE
  1066 1250           23 GOTO   TMRC20 (1252)
  1067
  1068 1251 TMRC15 1004 S2=     0             SWITCH TO STO MODE
  1069 1252 TMRC20     1 GOSUB  CALCRA         CALC NEW ACTIVE REG ADDRESS
  1069 1253            0                       *TIMER ROM:  TM1, @0513
  1070 1254 TMRC30     4 S3=    0             DON'T SUPPRESS REG NUMBER
  1071 1255          103 GOTO   TMRKJ1 (1265)
  1072
*.....................................................................
  1074
  1075                  ENTRY  TMREEX
  1076 1256 TMREEX   514 ?S6=1                ENTERING ADDRESS ?
  1077 1257           67 GOC    TMRKJ1 (1265) YES, IGNORE THE KEY
  1078 1260         1214 ?S7=1                SHIFT SET ?
  1079 1261           67 GOC    TMEE20 (1267) YES
  1080 1262           14 ?S3=1                SUPPRESSING REG NUMBER ?
  1081 1263         1717 GOC    TMRC30 (1254) YES
  1082 1264           10 S3=    1             SUPPRESS REG NUMBER
  1083 1265 TMRKJ1     1 GOLONG TMR00K         DEBOUNCE AND CHECK KEY
  1083 1266            2                       *TIMER ROM:  TM1, @1051
```

```
  1084
  1085 1267 TMEE20    4 S3=    0             DON'T SUPPRESS REG NUMBER
  1086 1270          114 ?S4=1                3-DIGIT REG NUMBER ?
  1087 1271           37 GOC    TMEE30 (1274) YES
  1088 1272          110 S4=    1             NO, SET TO 3 DIGITS
  1089 1273 CLRSHJ 1423 GOTO   CLRSHF (1235)
  1090 1274 TMEE30  104 S4=    0             SET TO 2-DIGIT REG NUMBER
  1091 1275         630 C=M
  1092 1276         234 PT=    5
  1093 1277         102 C=0    PT            2-DIGIT STO POINTER
  1094 1300         434 PT=    8
  1095 1301         102 C=0    PT            2-DIGIT RCL POINTER
  1096 1302         530 M=C
  1097 1303        1303 GOTO   TMSST6 (1233)
*
*.....................................................................
  1100
  1101                ENTRY  ADRENT
  1102 1304 ADRENT    1 GOSUB  GTR#MC        C.X= ACTIVE REGISTER NUMBER
  1102 1305           0                      *TIMER ROM:  TM1, @1505
  1103 1306           4 S3=    0             DON'T SUPPRESS REG NUMBER
  1104 1307         514 ?S6=1                ALREADY ENTERING REG NO. ?
  1105 1310          67 GOC    ADRE20 (1316) YES
  1106 1311         106 C=0    X             NO, START NEW REG NUMBER
  1107 1312        1146 C=C-1  X             C.X= FFF
  1108 1313 ADRE15  266 AC EX  XS            C.X= DFF
  1109 1314         510 S6=    1             SET REG NUMBER ENTRY FLAG
  1110 1315         223 GOTO   ADRE50 (1337)
* NOTE: BACKARROW MAY HAVE CLEARED ALL THE DIGITS SO THE FOLLOWING TEST
*       IS NEEDED
  1113 1316 ADRE20 1066 C=C+1  XS
  1114 1317        1166 C=C-1  XS            FIRST DIGIT ENTERED YET ?
  1115 1320        1737 GOC    ADRE15 (1313) NO, C.XS= F
  1116 1321        1434 PT=    1
  1117 1322        1606 A SR   X             A.X= 0D0
  1118 1323        1042 C=C+1  PT            SECOND DIGIT ENTERED YET ?
  1119 1324          63 GONC   ADRE30 (1332) YES
  1120 1325         242 AC EX  PT            NO, ADD IT
  1121 1326         114 ?S4=1                3-DIGIT REGISTER NUMBER ?
  1122 1327         107 GOC    ADRE50 (1337) YES
  1123 1330        1706 C SR   X             NO, 2 DIGITS, C.X= 0DD
  1124 1331          53 GOTO   ADRE40 (1336)
  1125 1332 ADRE30 1142 C=C-1  PT            FIX UP THE DIGIT
  1126 1333        1634 PT=    0
  1127 1334        1606 A SR   X             A.X= 00D
  1128 1335         242 AC EX  PT            ADD 3RD DIGIT
  1129 1336 ADRE40  504 S6=    0             TERMINATE ADDRESS ENTRY
  1130 1337 ADRE50    1 GOSUB  PUTR#         UPDATE REGISTER NUMBER
  1130 1340           0                      *TIMER ROM:  TM1, @1514
  1131 1341         514 ?S6=1                STILL ENTERING REG NUMBER ?
  1132 1342        1317 GOC    CLRSHJ (1273) YES
  1133 1343           1 GOSUB  CALCRC        NO, CALC ACTIVE REG ADDRESS
  1133 1344           0                      *TIMER ROM:  TM1, @0514
  1134 1345         673 GOTO   TENT45 (1434) BEEP IF REG DOESN'T EXIST
  1135
*.....................................................................
* WANT TO READ TIME 47 WORD TIMES AFTER DETECTING "ENTER" KEY SINCE THE
* "R/S" KEY TAKES 47 WORD TIMES TO START/STOP THE STOPWATCH
*
* MAY BE DEAD TIME BETWEEN "ENTER"S WHEN IN STO MODE IS 0.071 SEC WITH
```

```
* SLOWEST 41C CLOCK.
  1142
  1143                    ENTRY  TMRENT
  1144 1346 TMRENT  514 ?S6=1                  ENTERING ADDRESS ?
  1145 1347        1167 GOC    TMRKJ1 (1265) YES, IGNORE THE KEY
  1146 1350           1 GOSUB  ENTMR         ENABLE TIMER, DISABLE RAM
  1146 1351           0                      *TIMER ROM:  TM0, @0342
  1147 1352        1650 PT=B
  1148 1353          70 RDTIME               C= STOPWATCH TIME
  1149 1354         160 N=C                  SAVE THE TIME
  1150 1355        1014 ?S2=1                RCL MODE ?
  1151 1356           1 GSUBC  R-TO-S        YES, SET TO STO MODE
  1151 1357           1                      *TIMER ROM:  TM1, @0511
  1152 1360         630 C=M
  1153 1361        1174 RCR    9             C.X= STO REGISTER ADDRESS
  1154 1362           1 GOSUB  CHKADR        ERROR IF REG DOESN'T EXIST
  1154 1363           0                      *MAINFRAME:  CN5, @1156
  1155                                       IN: C.X= REGISTER ADDRESS
  1156                                       ASSUME: PERIPHS DISABLED
  1157                                       OUT: C.X PRESERVED, HEXMODE
  1158                                            C.X REGISTER ENABLED,
  1159                                            B= LEGALIZED CONTENT
  1160                                       USES: A,B,C,S9,ACTIVE PT,
  1161                                             DADD
  1162                                            (NO ADDL SUB LEVELS)
  1163 1364         260 C=N                  C= CURRENT TIME
  1164 1365         404 S8=    0             IGNORE KEYBOARD
  1165 1366           1 GOSUB  GETMRC        CONVERT TO H.MS
  1165 1367           0                      *TIMER ROM:  TM0, @0475
  1166 1370         630 C=M
  1167 1371        1174 RCR    9             C.X= STO REGISTER ADDRESS
  1168 1372        1160 DADD=C
  1169 1373         256 AC EX                C= H.MS TIME
  1170 1374        1360 DATA=C               STORE THE SPLIT
  1171 1375        1710 RST KB
  1172 1376        1714 CHK KB               KEY STILL DOWN ?
  1173 1377         173 GONC   TENT30 (1416) NO
  1174 1400         410 S8=    1             YES, CHECK KEYBOARD
  1175 1401        1110 S9=    1             RETURN ON KEY UP
  1176 1402        1614 ?S0=1                IN DIFFERENCE MODE ?
  1177 1403           1 GSUBC  DIFF          YES, CALC THE DIFFERENCE
  1177 1404           1                      *TIMER ROM:  TM1, @0413
  1178                                         IF IN DIFFERENCE MODE
  1179                                       (P+1)
  1180                                       (P+2) ASSUME NO ERROR POSS.
  1181 1405        1710 RST KB
  1182 1406        1714 CHK KB               KEY STILL DOWN ?
  1183 1407          73 GONC   TENT30 (1416) NO, DON'T DISPLAY
  1184 1410         340 SEL Q
  1185 1411          34 PT=    3             DISPLAY 100TH'S
  1186 1412           1 GOSUB  DSPTMR        DISPLAY SPLIT/DIFFERENCE
  1186 1413           0                      *TIMER ROM:  TM1, @1575
  1187 1414         106 C=0    X             DISABLE PERIPHERALS FOR
  1188 1415        1760 PFAD=C               "KEY UP" ABORT CASE
  1189 1416 TENT30  630 C=M
  1190 1417          74 RCR    3
  1191
*
* IN: PERIPHERALS DISABLED
*     C= COPY OF "M" ROTATED SO THAT C.X= ACTIVE REGISTER NUMBER
```

```
*     !!! MUST NOT BE ENTERING THE REGISTER NUMBER !!!!
*
  1197                    ENTRY  TENT35
  1198 1420 TENT35 1240 SETDEC
  1199 1421        1046 C=C+1  X             INCREMENT REGISTER NUMBER
  1200 1422        1140 SETHEX
  1201 1423        1366 ? C#0  XS            REGISTER NUMBER > 99 ?
  1202 1424          23 GONC   TENT40 (1426) NO
  1203 1425         110 S4=    1             YES, SET TO 3 DIGITS
  1204 1426 TENT40   1 GOSUB  PUTR#          UPDATE REGISTER NUMBER
  1204 1427          0                       *TIMER ROM:  TM1, @1514
  1205 1430        1174 RCR    9
  1206 1431        1046 C=C+1  X             INCREMENT ACTIVE REG ADDR
  1207 1432         274 RCR    5
  1208 1433         530 M=C                  STORE INCREMENTED ADDRESS
  1209 1434 TENT45 1214 ?S7=1               SHIFT SET ?
  1210 1435          1 GSUBC  TGLSHF        YES, TURN IT OFF
  1210 1436          1                      *TIMER ROM:  TM0, @1471
  1211 1437          1 GOSUB  RSTKB         WAIT FOR KEY UP
  1211 1440          0                      *MAINFRAME:  CN0, @0230
  1212                                      IN & ASSUME: HEXMODE
  1213                                      USES: C.X ONLY
  1214                                      44 MINIMUM INC GSB & RTN
  1215
  1216 1441         630 C=M
  1217 1442        1174 RCR    9
  1218 1443        1046 C=C+1  X             INCREMENT ADDRESS AGAIN
  1219 1444          26 A=0    XS
  1220 1445         566 A=A+1  XS
  1221 1446        1426 ? A<C  XS
  1222 1447         127 GOC    TENT60 (1461) YES, INVALID ADDRESS
  1223 1450        1160 DADD=C
  1224 1451          70 C=DATA               CHECK EXISTENCE OF NEXT REG
  1225 1452         256 AC EX                SAVE IN "A"
  1226 1453         630 C=M                  USE "M" AS DATA PATTERN
  1227 1454        1360 DATA=C
  1228 1455          70 C=DATA
  1229 1456         256 AC EX                C= ORIGINAL REG CONTENTS
  1230                                      A= DATA READ BACK
  1231 1457        1360 DATA=C               RESTORE ORIG REG CONTENTS
  1232 1460         630 C=M
  1233 1461 TENT60 1556 ? A#C               DOES THE NEXT REG EXIST ?
  1234 1462          1 GSUBC  BEEPK         NO, SOUND WARNING BEEP
  1234 1463          1                      *TIMER ROM:  TM3, @0222
  1235 1464          1 GOLONG TMR01         UPDATE LCD DISPLAY
  1235 1465          2                      *TIMER ROM:  TM1, @0576
  1236
*****************************************************************
* KEYCHK = KEY CHECK                                  2-6-81 RSW
*  THIS ROUTINE IS INTENDED TO TRY TO GIVE 1/100 SEC ACCURACY TO THE
* STOPWATCH BY ABORTING NON-ESSENTIAL ROUTINES WHEN A KEY TRANSITION
* IS DETECTED.
*
* IN & ASSUME: S8= 1 (0)   TO CHECK (IGNORE) KEYBOARD
*                IF S8=1, THEN: S9= 1 (0)   RTN ON KEY UP (DOWN)
* OUT:  IF S8=1 & S9=0 AND A KEY GOES DOWN, JUMP DIRECTLY TO "TMRKEY"
*       IF S8=1 & S9=1 AND THE KEY GOES UP, POPS STACK AND RETURNS
*       IF S8=0 OR NO KEY TRANSITIONS, RETURNS NORMALLY
* USES: NOTHING (EXCEPT POPPING A LEVEL OFF RETURN STACK)
*
```

```
  1250                  ENTRY  KEYCHK
  1251 1466 KEYCHK  414 ?S8=1               CHECK KEYBOARD ?
  1252 1467        1640 RTN NC              NO
  1253 1470        1114 ?S9=1               RETURN ON KEY UP ?
  1254 1471         103 GONC   KEYCK4 (1501) NO, ON KEY DOWN
  1255 1472        1710 RST KB
  1256 1473        1714 CHK KB              KEY STILL DOWN ?
  1257 1474        1540 RTN C               YES, RETURN NORMALLY
  1258 1475         640 CLRABC              CLEAN UP GARBAGE OUTPUT
  1259 1476        1140 SETHEX              AVOID ADDR CALC PROBLEMS
  1260 1477          40 SPOPHD
  1261 1500        1740 RTN                 EXIT THE CALLING ROUTINE
  1262 1501 KEYCK4 1714 CHK KB              KEY DOWN YET ?
  1263 1502           1 GOLC   TMRKEY       YES, GO CHECK IT OUT
  1263 1503           3                     *TIMER ROM:  TM1, @0773
  1264 1504        1740 RTN                 RETURN NORMALLY
*************************************************************************
* GETR# = GET REGISTER NUMBER                                 1-5-81 RSW
*
* IN:     C= M REGISTER CONTENTS
*         S2= 0 FOR STO MODE, S2= 1 FOR RCL MODE
* ASSUME: NOTHING
* OUT:    C= ROTATED M REGISTER CONTENTS
*         IF S2= 0 ON INPUT, C.X= STO REGISTER NUMBER
*         IF S2= 1 ON INPUT, C.X= RCL REGISTER NUMBER
* USES:   C ONLY
*
  1276                  ENTRY  GTR#MC
  1277                  ENTRY  GETR#M
  1278                  ENTRY  GETR#
  1279 1505 GTR#MC  406 A=C    X
  1280 1506 GETR#M  630 C=M
  1281 1507 GETR#    74 RCR    3             C.X= STO REGISTER NUMBER
  1282 1510        1014 ?S2=1                IN RECALL MODE ?
  1283 1511        1640 RTN NC               NO, DONE
  1284 1512          74 RCR    3             C.X= RCL REGISTER NUMBER
  1285 1513        1740 RTN
  1286
*************************************************************************
* PUTR# = PUT REGISTER NUMBER                                 1-5-81 RSW
*
* IN:     C= ROTATED M REGISTER CONTENTS
*         IF S2= 0 ON INPUT, C.X= STO REGISTER NUMBER
*         IF S2= 1 ON INPUT, C.X= RCL REGISTER NUMBER
* ASSUME: NOTHING
* OUT   : M REGISTER UPDATED, C=M
* USES: : C ONLY
*
  1297                  ENTRY  PUTR#
  1298 1514 PUTR#   474 RCR    8             POSITION FOR RCL MODE
  1299 1515        1014 ?S2=1                IN RCL MODE ?
  1300 1516          27 GOC    PUTR10 (1520) YES, SO DONE
  1301 1517          74 RCR    3             POSITION FOR STO MODE
  1302 1520 PUTR10  530 M=C
  1303 1521        1740 RTN
*************************************************************************
* REG# = REGISTER NUMBER                                      4-15-81 RSW
*  SHIFTS REGISTER NUMBER LEFT INTO DISPLAY
*
* IN & ASSUME: DISPLAY ENABLED, HEXMODE
```

```
*         S8= 1 (0)    CHECK (IGNORE) KEYBOARD
*            IF S8= 1  THEN:   S9= 1 (0)    RETURN ON KEY UP (DOWN)
*         S6= 1 (0)    (NOT) ENTERING REGISTER NUMBER
*         S4= 1 (0)    FOR 3 (2) DIGIT REGISTER NUMBER
*         S2= 1 (0)    FOR RCL (STO) MODE
*         S0= 1 (0)    FOR DIFFERENCE (SPLIT) MODE
*
*         M REGISTER POINTERS ACCURATE
*           IF THE USER IS ENTERING THE REGISTER NUMBER, THE NOT-YET-
*           SPECIFIED DIGITS MUST BE F'S AND MUST BE AS SHOWN:
*                              2-DIGIT REGISTER NUMBER:
*                                   0DD   DFX   FFX   (X= DON'T CARE)
*                              3-DIGIT REGISTER NUMBER:
*                                   0DD   DDF   DFF   FFF
* OUT:  HEXMODE
* USES: A.X, C, +1 SUB LEVEL, ARITH MODE
*       (NO ST, NO PT, NO TIMER CHIP ACCESS)
*
  1327                   ENTRY  REG#
  1328 1522 REG#    460 LDI                  ASSUME STO MODE
  1329 1523          56 CON    @56           RIGHT ARROW
  1330 1524        1014 ?S2=1                 IN RCL MODE ?
  1331 1525          33 GONC   DSTM10 (1530) NO, IN STO MODE
  1332 1526         460 LDI                  LOAD LOW 12 BITS OF C WITH
  1333 1527          75 CON    @75           =
  1334 1530 DSTM10 1750 SLSABC
  1335 1531         460 LDI                  ASSUME SPLIT MODE
  1336 1532          22 CON    @22           R
  1337 1533        1614 ?S0=1                 IN DIFFERENCE MODE ?
  1338 1534          33 GONC   DSTM20 (1537) NO, SPLIT MODE
  1339 1535         460 LDI                  LOAD LOW 12 BITS OF C WITH
  1340 1536           4 CON    @04           D
  1341 1537 DSTM20 1750 SLSABC
  1342 1540           1 GOSUB  GETR#M        C.X= REGISTER NUMBER
  1342 1541           0                      *TIMER ROM:  TM1, @1506
  1343 1542           1 GOSUB  KEYCHK        CHECK KEYBOARD IF S8= 1
  1343 1543           0                      *TIMER ROM:  TM1, @1466
  1344 1544         406 A=C    X
  1345 1545         136 C=0    S
  1346 1546        1076 C=C+1  S
  1347 1547         114 ?S4=1                 3-DIGIT REGISTER NUMBER ?
  1348 1550          33 GONC   DSTM30 (1553) NO, 2-DIGIT
  1349 1551        1076 C=C+1  S
  1350                LEGAL                  (CLEAR THE CARRY FLAG)
  1351 1552          43 GOTO   DSTM35 (1556)
  1352 1553 DSTM30  514 ?S6=1                ENTERING REGISTER NUMBER ?
  1353 1554          27 GOC    DSTM35 (1556) YES, A.X= FDF
  1354 1555        1746 A SL   X              NO,  A.X= 0D0
  1355 1556 DSTM35  460 LDI                  LOAD LOW 12 BITS OF C WITH
  1356 1557          60 CON2   3      0      LCD FORMAT DIGIT
  1357 1560 DSTM40 1574 RCR    12
  1358 1561         266 C=A    XS            C[4:2]= LCD FORMAT DIGIT
  1358 1562         426                      (INSERTED BY ASSEMBLER)
  1359 1563        1074 RCR    2
  1360 1564         566 A=A+1  XS            PROMPT NEEDED ?
  1361 1565          33 GONC   DSTM50 (1570) NO, THIS DIGIT WAS ENTERED
  1362 1566         460 LDI                  YES
  1363 1567          37 CON    @37           PROMPT CHARACTER
  1364 1570 DSTM50 1750 SLSABC
  1365 1571        1746 A SL   X              A.XS= NEXT REG NUMBER DIGIT
```

```
  1366 1572        1176 C=C-1  S             DONE ?
  1367 1573        1653 GONC   DSTM40 (1560) NO
  1368 1574        1740 RTN
*
*********************************************************************
* DSPTMR - DISPLAY TIME OF TIMER (STOPWATCH)                2-2-81 RSW
* DSPTMR FIRST CLEARS THE DISPLAY.
* IF THE REGISTER NUMBER IS BEING DISPLAYED, THE REGISTER NUMBER
* (WITH PROMPTS IF IT IS NOT COMPLETE) IS LEFT-SHIFTED INTO THE DISPLAY.
* THEN THE TIME FROM "B" IS PLACED IN THE LEFT SIDE OF THE DISPLAY.
*
* IN:     Q= 3     TO TRY TO DISPLAY 100TH'S
*            4     TO TRY TO DISPLAY TENTHS
*         B= #HHMMSSCC.... WHERE #= 0 FOR POSITIVE OR 9 FOR NEGATIVE
* ASSUME: S8= 1 (0)    CHECK (IGNORE) KEYBOARD
*            IF S8= 1  THEN:   S9= 1 (0)    RETURN ON KEY UP (DOWN)
*         S6= 1 (0)    (NOT) ENTERING REGISTER NUMBER
*         S4= 1 (0)    FOR 3 (2) DIGIT REGISTER NUMBER
*         S3= 1 (0)    TO SUPPRESS (DISPLAY) THE REGISTER NUMBER
*         S2= 1 (0)    FOR RCL (STO) MODE
*         S0= 1 (0)    FOR DIFFERENCE (SPLIT) MODE
*
*         M REGISTER POINTERS ACCURATE
*           IF THE USER IS ENTERING THE REGISTER NUMBER, THE NOT-YET-
*           SPECIFIED DIGITS MUST BE F'S AND MUST BE AS SHOWN:
*                              2-DIGIT REGISTER NUMBER:
*                                   0DD   DFX   FFX   (X= DON'T CARE)
*                              3-DIGIT REGISTER NUMBER:
*                                   0DD   DDF   DFF   FFF
* OUT:    IF S8=1 & S9=0 AND A KEY GOES DOWN, JUMP DIRECTLY TO "TMRKEY"
*         IF S8=0 OR NO KEY TRANSITION: P SELECTED, PERIPHERALS DISABLED
*                                       HEXMODE, S5= 1
*         IF S8=1 & S9=1 AND THE KEY GOES UP, OUTPUT= GARBAGE
* USES:   A,B,C, P,Q, S5, +2 SUB LEVELS, DADD, PFAD, ARITH MODE
*                (NO TIMER CHIP ACCESS)
*
*                              56= MAX TO EXIT AFTER KEY TRANSITION
  1403                   ENTRY  DSPTMR        MAX TIME= 213 FOR 3 PROMPTS
  1404 1575 DSPTMR  240 SEL P                         = 205 FOR 3 DIGITS
  1405 1576           1 GOSUB  CLLCDE        ENABLE & CLEAR DISPLAY
  1405 1577           0                      *MAINFRAME: CN11, @0360
  1406                                       IN & ASSUME: NOTHING
  1407                                       OUT: DISPLAY ENA, RAM DIS
  1408                                       USES: C[11:0], ACTIVE PT,
  1409                                             DADD, PFAD (NO ST,
  1410                                             +0 SUB LVL, NO ARITH)
  1411 1600        1140 SETHEX
  1412 1601          14 ?S3=1               SUPPRESS REGISTER NUMBER ?
  1413 1602           1 GSUBNC REG#         NO, PUT REG NUM IN DISPLAY
  1413 1603           0                     *TIMER ROM:  TM1, @1522
  1414
*
*.....................................................................
* DSPTMM                                                    2-2-81 RSW
*  ASSUMES THAT THE REGISTER NUMBER IS ALREADY IN THE DISPLAY
*  (IF DISPLAYING REG NUMBER) SO "DSPTMM" JUST UPDATES THE TIME
*
* IN:     HEXMODE
*         B= #HHMMSSCC.....  WHERE "#" = 0 FOR POSITIVE
*                                      = NON-ZERO FOR NEGATIVE
```

```
*         Q= 3     TO TRY TO DISPLAY 100TH'S
*            4     TO TRY TO DISPLAY TENTHS
* ASSUME: S3= 1 (0)    TO SUPPRESS (DISPLAY) THE REGISTER NUMBER
*         S4= 1 (0)    FOR 3 (2) DIGIT REGISTER NUMBER
*         S8= 1 (0)    CHECK (IGNORE) KEYBOARD
*            IF S8= 1  THEN:   S9= 1 (0)    RETURN ON KEY UP (DOWN)
* OUT:    IF S8=1 & S9=0 AND A KEY GOES DOWN, JUMP DIRECTLY TO "TMRKEY"
*         IF S8=0 OR NO KEY TRANSITION: P SELECTED, PERIPHERALS DISABLED
*                                       HEXMODE, S5= 1
*         IF S8=1 & S9=1 AND THE KEY GOES UP, OUTPUT= GARBAGE
* USES:   A,B,C, P,Q, S5, +1 SUB LEVEL, DADD, PFAD, ARITH MOED
*                (NO TIMER CHIP ACCESS)
*
*
  1438                   ENTRY  DSSPTMM       55 MAX TO EXIT AFTER A KEY
  1439 1604 DSPTMM     1 GOSUB  KEYCHK        CHECK KEYBOARD IF S8= 1
  1439 1605            0                      *TIMER ROM:  TM1, @1466
  1440 1606          156 AB EX                A= HMS TIME
  1441 1607          340 SEL Q
  1442 1610           14 ?S3=1                SUPPRESSING REG NUMBER ?
  1443 1611           53 GONC   DSTM55 (1616) NO
  1444 1612         1536 ? A#0  S              NEGATIVE ?
  1445 1613          113 GONC   DSTM59 (1624) NO
  1446 1614         1724 DEC PT               YES
  1447                   LEGAL                (CLEAR THE CARRY FLAG)
  1448 1615           73 GOTO   DSTM59 (1624)
  1449 1616 DSTM55 1536 ? A#0  S              NEGATIVE ?
  1450 1617           23 GONC   DSTM58 (1621) NO
  1451 1620           34 PT=    3              YES, ONLY ROOM FOR TENTHS
  1452 1621 DSTM58   114 ?S4=1                3-DIGIT REGISTER NUMBER ?
  1453 1622           23 GONC   DSTM59 (1624) NO, 2-DIGIT
  1454 1623          134 PT=    4              YES, LEAVE 5 CHAR REG NUM
  1455 1624 DSTM59   453 GOTO   DSTM64 (1671)
  1456
  1457
*************************************************************************
* DSPINT = DISPLAY RESET INTERVAL                              2-2-81 RSW
*
* IN:     C= .HHHHMMSSCC...
* ASSUME: NOTHING
* OUT:    P SELECTED, PERIPHERALS DISABLED, HEXMODE
*         S8= 0, S5= 1
* USES:   A,B[S] & B[4:0],C, P,Q, S5,S8, +1 SUB LEVEL, DADD, PFAD, ARITH
*         (NO TIMER CHIP ACCESS)
*
*
  1469                   ENTRY  DSPINT
  1470 1625 DSPINT   340 SEL Q
  1471 1626         1034 PT=    2              DISPLAY TENTHS OF SECONDS
  1472 1627          240 SEL P
  1473 1630          136 C=0    S              MAKE IT POSITIVE
  1474 1631          416 A=C
  1475 1632            1 GOSUB  CLLCDE         ENABLE & CLEAR DISPLAY
  1475 1633            0                       *MAINFRAME: CN11, @0360
  1476                                         IN & ASSUME: NOTHING
  1477                                         OUT: DISPLAY ENABLED AND
  1478                                              CLEARED, RAM DISABLED
  1479                                         USES: C[11:0], ACTIVE PT,
  1480                                               PFAD, DADD, (NO ST,
  1481                                               +0 SUB LVL, NO ARITH)
```

```
  1482 1634         1076 C=C+1  S             C.S= 1
  1483 1635          404 S8=    0             DON'T CHECK KEYBOARD
  1484 1636          210 S5=    1             DON'5 SHOW AM/PM
  1485 1637         1534 PT=    12
  1486 1640 DSPT10  1502 ? A#0  PT            LEADING ZERO ?
  1487 1641           33 GONC   DSPT20 (1644) YES, REMOVE IT
  1488 1642         1076 C=C+1  S
  1489                   LEGAL                (CLEAR THE CARRY FLAG)
  1490 1643          303 GOTO   DSTM70 (1673)
  1491 1644 DSPT20  1752 A SL   WPT
  1492 1645          340 SEL Q
  1493 1646         1734 INC PT
  1494 1647          240 SEL P
  1495 1650         1176 C=C-1  S             DOWN TO 2 HOUR DIGITS ?
  1496 1651         1673 GONC   DSPT10 (1640) NO
  1497 1652          203 GOTO   DSTM65 (1672) YES
  1498
*.....................................................................
*
* DSPTM = DISPLAY TIME                                      1-28-81 RSW
*  THE RIGHT-HAND SIDE OF THE DISPLAY (Q:0) IS ASSUMED TO HAVE BEEN
*  APPROPRIATELY INITIALIZED. "DSPTM" PUTS THE TIME IN THE LEFT-HAND
*  SIDE AND LEAVES THE RIGHT-HAND SIDE UNCHANGED.
*  !! THIS MEANS THE DISPLAY MUST BE CLEARED IF NECESSARY, SINCE "DSPTM"
*     DOES NOT CLEAR THE DISPLAY !!
*
* IN:     A= 24-HOUR FORM OF TIME (UNNORMALIZED), WITH A.S= 0.
*                                  [ EXAMPLE:  A= 0HHMMSSCC..... ]
*         P SELECTED
*         Q= (RIGHTMOST TIME DIGIT TO BE DISPLAYED) - 1   WHERE THE
*            LEFTMOST DISPLAY CHARACTER= DIGIT 11 AND RIGHTMOST= DIGIT 0
* ASSUME: "CLK24"/"CLK12" BIT IN TIMER CHIP IS IN PROPER STATE
* OUT:    P SELECTED, PERIPHERALS DISABLED, HEXMODE
*         S8= 0, ST= 1 (0)  FOR 24 (12) HOUR DISPLAY
* USES:   A, B[S] & B[Q:0], C, P,Q, S5,S6,S8, +1 SUB LEVEL, DADD, PFAD,
*         ARITH MODE, TIMER PT
*
  1519                   ENTRY  DSPTM
  1520 1653 DSPTM    404 S8=    0             DON'T CHECK KEYBOARD
  1521 1654            1 GOSUB  TMRSTS        PUT UP SOFTWARE STATUS
  1521 1655            0                      *TIMER ROM:  TM3, @0244
  1522 1656          514 ?S6=1                24-HOUR DISPLAY
  1523 1657          117 GOC    DSTM63 (1670) YES
  1524 1660         1730 CST EX               NO, 12-HOUR
  1525 1661            1 GOSUB  TO12H         CONVERT TO 12-HOUR, S5=0.
  1525 1662            0                      *TIMER ROM:  TM3, @0262
  1526                                        S6 INITIALIZED
  1527
  1528 1663         1534 PT=    12
  1529 1664         1502 ? A#0  PT            2-DIGIT HOUR ?
  1530 1665           57 GOC    DSTM65 (1672) YES
  1531 1666           76 B=0    S             1-HOUR DIGIT
  1532 1667           63 GOTO   DSTM75 (1675)
  1533
  1534 1670 DSTM63  1730 CST EX
  1535 1671 DSTM64   210 S5=    1             NO AM/FM
  1536 1672 DSTM65   136 C=0    S
  1537 1673 DSTM70  1076 C=C+1  S
  1538 1674          376 BC EX  S             B.S= (# OF HRS DIGITS) - 1
*
```

```
*.....................................................................
* DSTM75                                                  1-28-81 RSW
*  THE RIGHTHAND SIDE OF THE DISPLAY (Q:0) IS ASSUMED TO HAVE BEEN
*  APPROPRIATELY INITIALIZED (CLEARED, FOR EXAMPLE). THIS CODE PUTS THE
*  TIME IN THE LEFTHAND SIDE AND LEAVES THE RIGHTHAND SIDE UNCHANGED.
*
* IN:     A= #HHMMMSSCC.....= UNNORMALIZED TIME TO DISPLAY
*            A.S= 0 FOR POSITIVE TIME
*            A.S= NON-ZERO FOR NEGATIVE TIME
*                    (SIGN-MAGNITUDE, NOT 10'S COMPLEMENT)
*         Q= (RIGHTMOST TIME DIGIT TO BE DISPLAYED) - 1     WHERE THE
*            LEFTMOST DISPLAY CHARACTER= DIGIT 11, RIGHTMOST= DIGIT 0
*            !!NOTE: IF AM/PM IS SELECTED (S5=0) IT WILL BE ADDED
*                    STARTING AT Q!!
* ASSUME: B.S=  (NUMBER OF HOURS DIGITS) - 1  [B.S= 3 IS EXPECTED MAX]
*         S8=   1 (0) DO (NOT) CHECK KEYBOARD
*            IF S8= 1, THEN:  S9= 1 (0)  RETURN ON KEY UP (DOWN)
*         S5=   1 (0) DON'T (DO) ADD AM/PPM
*            IF S5= 0, THEN:  S6= 1 (0)  FOR PM (AM)
* OUT:    IF S8=1 & S9=0 AND A KEY GOES DOWN, JUMP DIRECTLY TO "TMRKEY"
*         IF S8=0 OR NO KEY TRANSITION: P SELECTED, PERIPHERALS DISABLED
*                                       HEXMODE
*         IF S8=1 & S9=1 AND THE KEY GOES UP, OUTPUT= GARBAGE!!!!!!
* USES:   A,B[Q:0],C, P,Q, +1 SUB LEVEL, DADD, PFAD, ARITH MODE
*         (NO ST, NO TIMER CHIP ACCESS)

*  33 WORD TIMES MAX TO EXIT AFTER LAST KEY CHECK FOR STOPWATCH DELAY
  1567 1675 DSTM75    1 GOSUB  ENLCD          ENABLE DISPLAY, DISABLE RAM
  1567 1676            0                      *MAINFRAME,  CN1, @1766
  1568                                        IN & ASSUME: NOTHING
  1569                                        OUT: DISPLAY ENABLED
  1570                                             RAM DISABLED
  1571                                        USES: C.X, DADD, PFAD ONLY
  1572 1677          170 FLLDB                READ DISPLAY BACKWARDS
  1573 1700          150 SRLDB                REVERSE DISPLAY
  1574 1701          170 FLLDB                C[11:0]= DISPLAY REG B
  1575 1702          150 SRLDB                RESTORE DISPLAY
  1576 1703         1140 SETHEX
  1577 1704          340 SEL Q
  1578 1705          352 BC EX  WPT           SAVE RIGHT SIDE OF LCD IN B
  1579 1706          240 SEL P
  1580 1707          634 PT=    11
  1581 1710         1632 A SR   M             A[11]= LEADING HOUR DIGIT
  1582 1711         1536 ? A#0  S             NEGATIVE?
  1583 1712           63 GONC   DSTM85 (1720) NO
  1584 1713         1632 A SR   M             OPEN A[11] FOR MINUS SIGN
  1585 1714         1520 LC     13            MINUS [A[11]= D, C[11]= 2]
  1586 1715          634 PT=    11
  1587 1716          402 A=C    PT
  1588 1717          220 LC     2             MINUS SIGN
  1589 1720 DSTM85   336 C=B    S             C.S= HOURS COUNTER
  1590 1721         1336 ? B#0  S             MORE THAN 1 HOUR DIGIT ?
  1591 1722           27 GOC    DSTM87 (1724) YES
  1592 1723          220 LC     2             NO, ADD LEADING BLANK
  1593 1724 DSTM87    1 GOSUB  KEYCHK         CHECK KEYBOARD IF S8= 1
  1593 1725            0                      *TIMER ROM:  TM1, @1466
  1594 1726           23 GOTO   DSTM92 (1730)
  1595 1727 DSTM90   320 LC     3
  1596 1730 DSTM92  1176 C=C-1  S
  1597 1731         1763 GONC   DSTM90 (1727)
```

```
  1598 1732         1320 LC     11             DIGIT WITH COLON  (HH:)
  1599 1733          320 LC     3              DIGIT
  1600 1734         1320 LC     11             DIGIT WITH COLON  (MM:)
  1601 1735          320 LC     3              DIGIT
  1602 1736          720 LC     7              DIGIT WITH DECIMAL PT (SS.)
  1603 1737          320 LC     3              DIGIT
  1604 1740          320 LC     3              DIGIT  (CC)
  1605 1741          340 SEL Q
  1606 1742         1734 INC PT
  1607 1743          320 LC     3              REMV PUNCTUATION LAST DIGIT
  1608 1744          214 ?S5=1                 ADD AM/PM?
  1609 1745          177 GOC    DTM150 (1764)  NO
  1610 1746           12 A=0    WPT
  1611 1747         1224 ? PT=  7              TIME & DATE DISP ? (+HH:MM)
  1612 1750           27 GOC    DTM105 (1752)  YES, PUT AM/PM NEXT TO TIME
  1613 1751          220 LC     2              ADD BLANK [A[PT]=0,C[PT]=2]
  1614 1752 DTM105   514 ?S6=1                 PM ?
  1615 1753           47 GOC    DTM110 (1757)  YES
  1616 1754          542 A=A+1  PT
  1617 1755           20 LC     0              ADD "A" [A[PT]=1, C[PT]=0]
  1618 1756           23 GOTO   DTM120 (1760)
  1619 1757 DTM110   120 LC     1              ADD "P" [A[PT]=0, C[PT]=1]
  1620 1760 DTM120  1520 LC     13
  1621 1761         1734 INC PT
  1622 1762          402 A=C    PT
  1623 1763           20 LC     0              ADD "M" [A[PT]=D, C[PT]=0]
  1624 1764 DTM150   312 C=B    WPT            RESTORE REG NUMBER STUFF
  1625 1765          150 SRLDB                 UPDATE DISPLAY B REGISTER
  1626 1766           70 FLLDA
  1627 1767           50 SRLDA
  1628 1770           70 FLLDA
  1629 1771          412 A=C    WPT            ADD REG NUMBER STUFF TO HMS
  1630 1772          256 AC EX
  1631 1773           50 SRLDA
  1632 1774          240 SEL P
  1633 1775          106 C=0    X
  1634 1776         1760 PFAD=C               DISABLE PERIPHERALS
  1635 1777         1740 RTN
  1636                   UNLIST
  1639                   END

ERRORS :        0
```

SYMBOL TABLE (BWTMB2 = TIMER ROM QUAD 2 = TM1 = ADDRESSES @52000-53777)

```
ADENT   1103   -  1041
ADJ100   164   -    72
ADJ110   173   -   171
ADJ20    116   -   114
ADJ40    152   -   147
ADJ42    154   -   151
ADRE15  1313   -  1320
ADRE20  1316   -  1310
ADRE30  1332   -  1324
ADRE40  1336   -  1331
ADRE50  1337   -  1327 1315
ADRENT  1304   -
C=T+D    222   -
C=T+D0   223   -
CALCRA   513   -
CALCRC   514   -
CLKOFF  1117   -
CLOCK   1064   -
CLRSHF  1235   -  1273 1177
CLRSHJ  1273   -  1342
CORECT     7   -
DIFF     413   -
DIFF20   420   -   416
DIFF25   431   -   455
DIFF30   434   -   430
DIFF35   450   -   444
DIFF38   454   -   477  447
DIFF40   457   -   453
DIFF50   467   -   465
DIFF60   474   -   472
DIFF65   500   -   433
DIFFEX   417   -   456
DSPINT  1625   -
DSPT10  1640   -  1651
DSPT20  1644   -  1641
DSPTM   1653   -
DSPTMM  1604   -
DSPTMR  1575   -
DSTM10  1530   -  1525
DSTM20  1537   -  1534
DSTM30  1553   -  1550
DSTM35  1556   -  1554 1552
DSTM40  1560   -  1573
DSTM50  1570   -  1565
DSTM55  1616   -  1611
DSTM58  1621   -  1617
DSTM59  1624   -  1622 1615 1613
DSTM63  1670   -  1657
DSTM64  1671   -  1624
DSTM65  1672   -  1665 1652
DSTM70  1673   -  1643
DSTM75  1675   -  1667
DSTM85  1720   -  1712
DSTM87  1724   -  1722
DSTM90  1727   -  1731
DSTM92  1730   -  1726
```

```
DTM105  1752   -  1750
DTM110  1757   -  1753
DTM120  1760   -  1756
DTM150  1764   -  1745
GETM.X   532   -
GETMXP   531   -
GETR#   1507   -
GETR#M  1506   -
GTR#MC  1505   -
HM-SC    377   -
HMSEC1   346   -
HMSS10   341   -   337
HMSS20   343   -
HMSS35   356   -   353  345
HMSS40   364   -
HMSSCB   332   -
HMSSEC   333   -
HMSTER   354   -   417  376  372  340
KEYCHK  1466   -
KEYCK4  1501   -  1471
PUTR#   1514   -
PUTR10  1520   -  1516
R-TO-S   511   -
REG#    1522   -
RSTKBT   503   -
SDATE     43   -
SETIME    17   -
STMN23   241   -   237
STMN25   251   -   233
STMN27   256   -   245  240
STMN30   260   -   255
STMN40   266   -   262
STMN55   306   -   303
STMN60   317   -   265
STMNER   304   -   312  257
SW       540   -
T24H20   220   -   216  214  212
TE10      62   -    33
TENT30  1416   -  1407 1377
TENT35  1420   -
TENT40  1426   -  1424
TENT45  1434   -  1345
TENT60  1461   -  1447
TM20     564   -   560
TMBK10  1137   -  1135
TMBK20  1145   -  1127
TMCHS   1111   -  1050
TMEE20  1267   -  1261
TMEE30  1274   -  1271
TMEEX   1105   -  1046
TMENT   1070   -  1027
TMEXIT  1075   -
TMR/S   1072   -  1026
TMR00    574   -
TMR00K  1051   -  1143
TMR00S  1236   -
TMR01    576   -
TMR07    620   -   616
TMR0KJ  1143   -  1211 1201 1162 1160 1154
TMR10    622   -   603
```

```
TMR12    642   -   635
TMR15    650   -   643
TMR17    666   -   652
TMR18    667   -   646
TMR24    673   -   621
TMR70    732   -   713  707
TMRBAK  1124   -  1047
TMRC15  1251   -  1246 1152
TMRC20  1252   -  1250
TMRC30  1254   -  1263 1207 1205
TMRCH3  1206   -  1203
TMRCHK   701   -   772  754  744  735  670  665  633  610
TMRCHS  1200   -  1111
TMRCL   1107   -  1043
TMRDE    653   -   647  641
TMREEX  1256   -
TMRENT  1346   -
TMREX   1073   -  1125
TMRKEY   773   -   763  746  722  702
TMRKJ1  1265   -  1347 1257 1255 1244 1242 1240
TMROFF  1113   -  1042
TMRR/S  1161   -  1072
TMRRCL  1241   -
TMRS20  1172   -  1167
TMRS30  1173   -  1171
TMRSHF  1236   -  1144
TMRSST  1210   -  1112
TMSHF   1144   -  1045
TMSST   1112   -  1044
TMSST1  1221   -  1222
TMSST2  1225   -  1220
TMSST4  1230   -  1226
TMSST6  1233   -  1303
TO24H    202   -
TOFF    1120   -  1067
XTIM00    20   -    10
```

ENTRY TABLE (BWTMB2 = TIMER ROM QUAD 2 = TM1 = ADDRESSES @52000-53777)

ADJ100	164	-
ADRENT	1304	-
C=T+D	222	-
C=T+D0	223	-
CALCRA	513	-
CALCRC	514	-
CLKOFF	1117	-
CLOCK	1064	-
CORECT	7	-
DIFF	413	-
DSPINT	1625	-
DSPTM	1653	-
DSPTMM	1604	-
DSPTMR	1575	-
GETM.X	532	-
GETMXP	531	-
GETR#	1507	-
GETR#M	1506	-
GTR#MC	1505	-
HM-SC	377	-
HMSEC1	346	-
HMSS20	343	-
HMSS40	364	-
HMSSCB	332	-
HMSSEC	333	-
KEYCHK	1466	-
PUTR#	1514	-
R-TO-S	511	-
REG#	1522	-
RSTKBT	503	-
SDATE	43	-
SETIME	17	-
SW	540	-
TENT35	1420	-
TMEXIT	1075	-
TMR00	574	-
TMR00K	1051	-
TMR01	576	-
TMRCHK	701	-
TMREEX	1256	-
TMRENT	1346	-
TMRKEY	773	-
TMROFF	1113	-
TMRRCL	1241	-
TMRSHF	1236	-
TO24H	202	-

EXTERNAL REFERENCES (BWTMB2 = TIMER ROM QUAD 2 = TM1 = ADR @52000-53777)

1/X10	152	
1/X10	153	
1/X13	160	
1/X13	161	
115860	314	
115860	315	
36000	561	
36000	562	
AD1-10	155	
AD1-10	156	
ADRENT	1103	
ADRENT	1104	
BEEP2	177	
BEEP2	200	
BEEPK	714	1462
BEEPK	715	1463
BEEPKP	717	
BEEPKP	720	
C-YMDD	266	
C-YMDD	267	
C=T+D	26	60
C=T+D	27	61
CALCRA	1252	
CALCRA	1253	
CALCRC	1233	1343
CALCRC	1234	1344
CHECKX	46	
CHECKX	47	
CHKADR	626	1362
CHKADR	627	1363
CHKALM	173	
CHKALM	174	
CHKLB	673	764
CHKLB	674	765
CHKXM	21	
CHKXM	22	
CLDSP	1101	
CLDSP	1102	
CLLCDE	1576	1632
CLLCDE	1577	1633
CLRALM	572	
CLRALM	573	
CLRALS	1115	
CLRALS	1116	
DATECK	270	
DATECK	271	
DIFF	644	1403
DIFF	645	1404
DSPTMM	770	
DSPTMM	771	
DSPTMR	676	1412
DSPTMR	677	1413
DV2-13	126	
DV2-13	127	
ENCP00	73	1077
ENCP00	74	1100

```
REG#     664  1603
RSTKB    507   574  1437
RSTKB    510   575  1440
RSTKBT  1075
RSTKBT  1076
RTNP+2   354
RTNP+2   355
SDHMSC   226
SDHMSC   227
SETAF0   162
SETAF0   163
SRHBUF   167
SRHBUF   170
T=T+TP    63
T=T+TP    64
TENT35  1223
TENT35  1224
TERROR   304
TERROR   305
TGLSHF  1065  1073  1236  1435
TGLSHF  1066  1074  1237  1436
TMR00   1055
TMR00   1056
TMR00K  1265
TMR00K  1266
TMR01    730  1464
TMR01    731  1465
TMREEX  1105
TMREEX  1106
TMRENT  1070
TMRENT  1071
TMRKEY   577  1502
TMRKEY   600  1503
TMRRCL  1107
TMRRCL  1110
TMRSTS  1113  1654
TMRSTS  1114  1655
TO12H   1661
TO12H   1662
TO24H    241
TO24H    242
UNNOR1   235   650
UNNOR1   236   651
UNNOR2   273   335
UNNOR2   274   336
UNNORM   351
UNNORM   352
X20      405
X20      406
X20Q     401
X20Q     402
X20Q8    320
X20Q8    321

End of VASM assembly
********************************************************************************
VASM ROM ASSEMBLY          REV.  6/81A            HP-82182A TIMER MODULE

OPTIONS: L C S          COCONUT TIMER             ADDRESSES @54000-55777
```

```
     2                  FILE   BWTMB3             COCONUT TIMER Q3 = TM2
*
*
*
*********************************************************************
*
*   ALARM CATALOG
*
*********************************************************************
*  INTERNAL STATUS USE:
*              S5=  1   (0)             (DON'T)  PRINT  [TEMPORARY USE]
*              S7=  1   (0)            SHIFT  (NOT)  SET
*        M[S]= NON-ZERO (0)            CATALOG (NOT) RUNNING
*
    16    0          224 CON    @224          T
    17    1            1 CON    @01           A
    18    2            3 CON    @03           C
    19    3           15 CON    @15           M
    20    4           14 CON    @14           L
    21    5            1 CON    @01           A
    22                   ENTRY  ALMCAT
    23    6 ALMCAT     1 GOSUB  SRHBFI        SEARCH FOR TIMER BUFFER
    23    7            0                      *TIMER ROM:  TM2, @1137
    24   10           23 GOTO   ACT110 (  12) (P+1) FOUND IT
    25   11         1740 RTN                  (P+2) CATALOG EMPTY
    26   12 ACT110   546 A=A+1  X             A.X= ADDR OF 1ST ALARM
    27   13          246 AC EX  X
    28   14          136 C=0    S
    29   15         1076 C=C+1  S             CATALOG RUNNING
    30   16          530 M=C                  M.X= 1ST ALARM ADDRESS
    31   17         1704 CLR ST
    32   20         1104 S9=    0             NO PRINTER ERRORS
*     !!! S9 MUST BE PRESERVED THROUGHOUT ALMCAT SINCE IT IS TESTED BY
*     !!! THE PRINTER SUBROUTINE CALLS.
    35
* IN:  HEXMODE, PERIPHERALS DISABLED, M REG POINTERS
    37
    38                   ENTRY  ACT120
    39   21 ACT120     1 GOSUB  RSTKBT        RESET KEYBOARD
    39   22            0                      *TIMER ROM:  TM1, @0503
    40
* IN:  PERIPHERALS DISABLED, M REG POINTERS
    42
    43                   ENTRY  ACT125
    44   23 ACT125 1340 DISOFF                TURN DISPLAY OFF
    45   24            1 GOSUB  DSTMDA        DISPLAY TIME & DATE
    45   25            0                      *TIMER ROM:  TM3, @0060
    46   26          204 S5=    0             NOT PRINTING
    47   27            1 GOSUB  WAITKD        WAIT 0.6 SECONDS
    47   30            0                      *TIMER ROM:  TM2, @0417
    48   31          404 S8=    0             PRINT ONLY IN TRACE
    49   32            1 GOSUB  IAUALL        OK TO PRINT ?
    49   33            0                      *ILPRINTER:  PL3, @0663
    50                                        IN: S8= 0 PRINT TRACE ONLY
    51                                            S8= 1 PRT NORM & TRACE
    52                                        IN & ASSUME: HEXMODE, CHIP
    53                                            0 ENABLED, PERIPHERALS
    54                                            DISABLED
    55                                        OUT: (P+1) DON'T PRINT
    56                                             (P+2) OK TO PRINT
```

```
 57                                          USES: A,C,N, S0-S9, ACTIVE
 58                                                PT,  +2 SUB LEVELS
 59                                          (INCLUDING PIL PRINTER USE)
 60   34          213 GOTO   ACT140 (  55) (P+1) DON'T PRINT
 61   35            1 GOSUB  OUTPCT        (P+2) SEND PAPER ADVANCE
 61   36            0                      *ILPRINTER:  PL0, @1735
 62                                          IN: S9=0 -- IF NO PRTR ERRS
 63                                                    HAVE OCCURRED
 64                                              S9=1 -- ERRORS OCCURRED
 65                                              !!! DOES NOT RETURN !!!
 66                                          ASSUME: HEXMODE, PERIPH'LS
 67                                                  DISABLED
 68                                          OUT: S9=0 IF S9=0 ON INPUT
 69                                          USES: A,C,N, S0-S7, +2 SUB
 70                                                LEVELS (NO PT)
 71                                          (INCLUDING PIL PRINTER USE)
 72   37         1340 DISOFF               TURN DISPLAY OFF
 73   40            1 GOSUB  TMSG          PRINT ALARM TIME & DATE
 73   41            0                      *TIMER ROM:  TM2, @0431
 74   42         1340 DISOFF               TURN DISPLAY OFF
 75   43          510 S6=    1             (P+2) OK TO PRINT
 76   44            1 GOSUB  INTVAL        DISPLAY RESET INTERVAL
 76   45            0                      *TIMER ROM:  TM2, @1234
 77   46           53 GOTO   ACT135 (  53) (P+1) NO RESET INTERVAL
 78   47            1 GOSUB  TMSG          (P+2) PRINT RESET INTERVAL
 78   50            0                      *TIMER ROM:  TM2, @0431
 79   51            1 GOSUB  WAITK6        WAIT 0.6 SECONDS
 79   52            0                      *TIMER ROM:  TM2, @0420
 80   53 ACT135  615 CON    @615           GOSUB TO BECHK & PECHK
 81   54         1474 CON    @1474          *ILPRINTER:  PL0, @1737
 82                                          (DISPLAY ANY PRTR ERRORS)
 83                                          IN: S9=0 -- IF NO PRTR ERRS
 84                                                    HAVE OCCURRED
 85                                              S9=1 -- ERRORS OCCURRED
 86                                              !!! DOES NOT RETURN !!!
 87                                          ASSUME: HEXMODE, PERIP DIS
 88                                          OUT: S9=0 IF S9=0 ON INPUT
 89                                          USES: A,C,N, S0-S7, +2 SUB
 90                                                LEVELS (NO PT)
 91                                          (INCLUDING PIL PRINTER USE)
 92   55 ACT140 1340 DISOFF                 TURN DISPLAY OFF
 93   56            1 GOSUB  DSAMS0        SHOW 1ST 12 CHAR OF ALM MSG
 93   57            0                      *TIMER ROM:  TM0, @0621
 94   60          353 GOTO   ACT175 ( 115) (P+1) NO MESSAGE
 95   61          260 C=N
 96   62         1150 REGN=C 9              SAVE MESSAGE REG IN REG 9
*    S8= 0 FROM "DSAMS0" !!!!
 98   63            1 GOSUB  IAUALL        OK TO PRINT ?
 98   64            0                      *ILPRINTER:  PL3, @0663
 99                                          IN: S8= 0 PRINT TRACE ONLY
100                                              S8= 1 PRT NORM & TRACE
101                                         IN & ASSUME: HEXMODE, CHIP
102                                              0 ENABLED, PERIPHERALS
103                                              DISABLED
104                                         OUT: (P+1) DON'T PRINT
105                                               (P+2) OK TO PRINT
106                                         USES: A,C,N, S0-S9, ACTIVE
107                                               PT,  +2 SUB LEVELS
108                                         (INCLUDING PIL PRINTER USE)
109   65           53 GOTO   ACT150 (  72) (P+1) DON'T PRINT
```

```
110   66           1 GOSUB  PRTLCD         (P+2) PRINT CONTENTS OF LCD
110   67           0                       *ILPRINTER:  PL2, @1671
111                                        IN & ASSUME: HEXMODE
112                                        OUT: CHIP 0 ENABLED, PERIPH
113                                             DISABLED, LCD TO PRTR
114                                             WITHOUT EOL
115                                        USES: A[X&S],B.X,C,N, S0,S9
116                                              ACTIVE PT, +1 SUB LVL
117                                        (INCLUDING PIL PRINTER USE)
118   70         210 S5=    1              PRINTING
119   71          23 GOTO   ACT155 (  73)
120   72 ACT150  204 S5=    0              NOT PRINTING
121   73 ACT155 1170 C=REGN 9
122   74         160 N=C                   RESTORE MESSAGE REGISTER
123   75           1 GOSUB  WAITKD         WAIT 0.6 SECONDS
123   76           0                       *TIMER ROM:  TM2, @0417
124   77        1340 DISOFF                TURN DISPLAY OFF
125  100           1 GOSUB  DSA2ND         SHOW LAST 12 CHARS ALM MSG
125  101           0                       *TIMER ROM:  TM0, @0731
126  102          73 GOTO   ACT165 ( 111)  (P+1) NO MORE CHARS
127  103         214 ?S5=1                 (P+2) PRINT?
128  104           1 GSUBC  PRTLCD         YES, PRINT DISPLAY CONTENTS
128  105           1                       *ILPRINTER:  PL2, @1671
129                                        PRINTER: SEE ABOVE COMMENTS
130  106           1 GOSUB  WAITKD         WAIT 0.6 SECS, CHK KEYBOARD
130  107           0                       *TIMER ROM:  TM2, @0417
131  110          23 GOTO   ACT170 ( 112)
132  111 ACT165 1440 DISTOG                TURN DISPLAY ON
133
134                 ENTRY  ACT170
135  112 ACT170  214 ?S5=1                 PRINTING ?
136  113           1 GSUBC  OUTPCT         YES, EOLL, BECHK, PECHK
136  114           1                       *ILPRINTER:  PL0, @1735
137                                        IN: S9=0 -- IF NO PRINTER
138                                                    ERRORS OCCURRED
139                                            S9=1 -- ERROR OCCURRED
140                                            !!! DOES NOT RETURN !!!
141                                        ASSUME: HEXMODE, PERIPH'LS
142                                                DISABLED
143                                        OUT: S9=0 IF S9=0 ON INPUT
144                                        USES: A,C,N, S0-S7, +2 SUB
145                                              LEVELS (NO PT)
146                                        [INCLUDES PIL PRINTER USE]
147  115 ACT175 1340 DISOFF                TURN DISPLAY OFF
148  116        1440 DISTOG                TURN DISPLAY ON
149  117        1204 S7=    0              CLEAR "SHIFT" BIT
150  120          63 GOTO   ACT190 ( 126)
151
152                 ENTRY  ACT180
153                 ENTRY  ACT185
154  121 ACT180  630 C=M                   STOP THE CATALOG
155  122         136 C=0    S
156  123 ACT185  530 M=C
157  124           1 GOSUB  RSTKBT         RESET KEYBOARD
157  125           0                       *TIMER ROM:  TM1, @0503
158  126 ACT190    1 GOSUB  CHKLB          CHECK LOW BATTERY
158  127           0                       *TIMER ROM:  TM0, @1510
159  130           1 GOSUB  ENCP00         ENA CHIP 0, DIS PERIPHERALS
159  131           0                       *MAINFRAME:  CN2, @0522
160                                        IN: NOTHING
```

```
   161                                          ASSUME: NOTHING
   162                                          OUT: C.X= 0
   163                                          USES: C.X, DADD, PFAD ONLY
   164  132         630 C=M
   165  133        1376 ? C#0  S                 CATALOG RUNNING ?
   166  134         453 GONC   ACT300 ( 201)     NO
   167  135        1714 CHK KB                   ANY KEY DOWN ?
   168  136         157 GOC    ACT250 ( 153)     YES
   169  137 ACT240    1 GOSUB  ALMSST            SINGLE STEP TO NEXT ALARM
   169  140           0                         *TIMER ROM:  TM2, @1205
   170  141         103 GOTO   ACT242 ( 151)     (P+1) CONT RUNNING CATALOG
   171  142         460 LDI                      (P+2) NO MORE ALARMS
   172  143        1446 CON    806
*   WAIT 0.5 SECONDS BEFORE EXITING TO GIVE USER A CHANCE TO HIT R/S
   174  144 ACT241 1714 CHK KB                   ANY KEY DOWN ?
   175  145          67 GOC    ACT250 ( 153)     YES
   176  146        1146 C=C-1  X                 DECREMENT C REG COUNTER
   177  147        1753 GONC   ACT241 ( 144)     CONTINUE WAITING LOOP
   178  150         443 GOTO   ACTEXT ( 214)     TIMEOUT, EXIT CATALOG
   179  151 ACT242    1 GOLONG ACT125            ALARM CATALOG, NO HEXMODE
   180  152           2                         *TIMER ROM:  TM2, @0023
*
   181  153 ACT250 1040 C=KEYS                   LOAD THE KEY
   182  154          74 RCR    3
   183  155         126 C=0    XS
   184  156         406 A=C    X                 A.X= KEY CODE
   185  157         460 LDI                      LOAD LOW 12 BITS OF C WITH
   186  160          30 CON2   1      8          OFF KEY CODE
   187  161        1546 ? A#C  X                 OFF KEY ?
   188  162 ACTOFF    1 GOLNC  TMROFF            YES
   188  163           2                         *TIMER ROM:  TM1, @1113
   189  164         460 LDI                      LOAD LOW 12 BITS OF C WITH
   190  165         207 CON2   8      7          R/S KEY CODE
   191  166        1546 ? A#C  X                 R/S KEY ?
   192  167 A180J1 1323 GONC   ACT180 ( 121)     YES, STOP CATALOG
   193  170         460 LDI                      LOAD LOW 12 BITS OF C WITH
   194  171         623 CON    403               WAIT 0.125 SECONDS
   195  172 ACT265 1146 C=C-1  X                 DECREMENT C REG COUNTER
   196  173        1773 GONC   ACT265 ( 172)     CONTINUE WAITING LOOP
   197  174        1710 RST KB                   RESET KEYBOARD
   198  175        1714 CHK KB                   KEY STILL DOWN ?
   199  176           1 GSUBNC RSTKB             NO, DO UP DEBOUNCE
   199  177           0                         *MAINFRAME:  CN0, @0230
   200                                          IN: HEXMODE
   201                                          ASSUME: HEXMODE
   202                                          USES: C.X ONLY
   203  200        1373 GOTO   ACT240 ( 137)     GET NEXT ALARM
*
*      IDLE LOOP WHEN CATALOG IS NOT RUNNING
*
   207  201 ACT300  116 C=0
   208  202        1234 PT=    7
   209  203         220 LC     2                 LOAD 2 MIN TIMEOUT COUNTER
   210  204 ACT310 1714 CHK KB                   ANY KEY DOWN ?
   211  205         247 GOC    ACT320 ( 231)     YES
   212  206         540 ?LLD                     LOW BATTERY ?
   213  207          33 GONC   ACT312 ( 212)     NO
   214  210        1172 C=C-1  M                 YES, MAKE TIME OUT FASTER
   215  211          37 GOC    ACTEXT ( 214)     TIMEOUT, EXIT ALMCAT
   216  212 ACT312 1172 C=C-1  M
```

```
   217  213         1713 GONC   ACT310 ( 204)
   218  214 ACTEXT    1 GOLONG TMEXIT         CLEAR KEYBOARD & MSG FLAGS
   218  215           2                       *TIMER ROM:  TM1, @1075
*  THE WHOLE RETURN STACK HAS BEEN BLOWN, SO MUST DO GOLONG TO NFRPU
   220
   221  216 ACTRST    1 GOSUB  RSTALM         RESET THE ALARM
   221  217           0                       *TIMER ROM:  TM0, @1042
   222  220         530 M=C                   (P+1) ALM HAD RST INTERVAL
   223  221 ACTR10  630 C=M                   (P+2)
   224  222         160 N=C                   SAVE M IN N
   225  223        1704 CLR ST                CLEAR RUN LABEL BIT
   226                                        NOT POWEROFF
   227  224           1 GOSUB  NXTALM         SET NEW HARDWARE ALARM
   227  225           0                       *TIMER ROM:  TM3, @1475
   228  226         260 C=N
   229  227         530 M=C
   230  230         463 GOTO   ACTRTJ ( 276)
   231
   232  231 ACT320   34 PT=    3
   233  232        1040 C=KEYS                C[4:3] = KEYCODE
   234  233         742 C=C+C  PT              ON KEY ?
   235  234        1267 GOC    ACTOFF ( 162) YES, TURN OFF W/O CLEARING
   236                                         INPUT SO CLOCK MODE WORKS
   237  235        1214 ?S7=1                 SHIFT SET ?
   238  236         513 GONC   ACT340 ( 307) NO
   239  237           1 GOSUB  TGLSHF         TOGGLE SHIFT ANNUNC. OFF
   239  240           0                       *TIMER ROM:  TM0, @1471
   240  241         460 LDI                   LOAD LOW 12 BITS OF C WITH
   241  242           6 CON    6              TABLE LENGTH - 1 = 6
   242  243           1 GOSUB  KEY-FC         PROCESS THE KEY
   242  244           0                       *TIMER ROM:  TM2, @0434
   243  245          64 CON2   3      4       "R"
   244  246         160 CON2   7      0       "C"
   245  247         302 CON2   12     2       "BST"
   246  250         204 CON2   8      4       "T"
   247  251          22 CON2   1      2       SHIFT
   248  252         303 CON2   12     3       BACK ARROW
   249  253           0 CON    0               END OF TABLE
   250  254        1423 GOTO   ACTRST ( 216) RESET THE ALARM
   251  255          63 GOTO   PURGA  ( 263) PURGE ALARM
   252  256         163 GOTO   ACTBST ( 274) BACK STEP
   253  257         103 GOTO   CURNTT ( 267) CURRENT TIME
   254  260 A180J2 1073 GOTO   A180J1 ( 167) SHIFT IS ALREADY OFF
   255  261        1333 GOTO   ACTEXT ( 214) EXIT ALARM CATALOG
   256  262         223 GOTO   SHFTON ( 304) IGNORE OTHER KEYS
   257                                        (TURN "SHIFT" BACK ON)
   258
   259  263 PURGA     1 GOSUB  PUGALM         PURGE THE ALARM
   259  264           0                       *TIMER ROM:  TM0, @0764
   260  265 ACTEXJ 1273 GOTO   ACTEXT ( 214) (P+1) ALARM CATALOG EMPTY
   261  266        1333 GOTO   ACTR10 ( 221) (P+2)
   262
   263  267 CURNTT    1 GOSUB  GDHMS          GET DAYS-HOURS-MIN-SEC
   263  270           0                       *TIMER ROM:  TM2, @1270
   264  271         340 SEL Q
   265  272         234 PT=    5              DISPLAY SECONDS
   266  273         543 GOTO   DST10  ( 347)
   267
   268  274 ACTBST    1 GOSUB  ALMBST         BACKSTEP TO PREVIOUS ALARM
   268  275           0                       *TIMER ROM:  TM2, @0400
```

```
269  276 ACTRTJ  433 GOTO   ACTRTN ( 341)
270
271  277 ACTR/S  630 C=M                   START CATALOG
272  300         136 C=0    S
273  301        1076 C=C+1  S
274                 LEGAL                 (CLEAR THE CARRY FLAG)
275  302           1 GOSUB  ACT185         CHECK FOR KEY, GET NEXT ALM
275  303           2                       *TIMER ROM:  TM2, @0123
276
277  304 SHFTON    1 GOSUB  TGLSHF         TOGGLE SHIFT ANNUNCIATOR ON
277  305           0                       *TIMER ROM:  TM0, @1471
278  306        1523 GOTO   A180J2 ( 260)
279
280
281  307 ACT340  460 LDI                   LOAD LOW 12 BITS OF C WITH
282  310          10 CON    8              TABLE LENGTH - 1 = 8
283  311           1 GOSUB  KEY-FC         PROCESS THE KEY
283  312           0                       *TIMER ROM:  TM2, @0434
284  313         200 CON2   8      0       "D"
285  314          64 CON2   3      4       "R"
286  315         202 CON2   8      2       "M"
287  316         302 CON2   12     2       "SST"
288  317         204 CON2   8      4       "T"
289  320         207 CON2   8      7       "R/S"
290  321          22 CON2   1      2       SHIFT
291  322         303 CON2   12     3       BACK ARROW
292  323           0 CON    0               END OF KEY CODE TABLE
293  324         303 GOTO   ALDATE ( 354) ALARM DATE
294  325         373 GOTO   RESETI ( 364) ALARM RESET INTERVAL
295  326          73 GOTO   ALMMSG ( 335) ALARM MESSAGE
296  327          73 GOTO   ACTSST ( 336) SINGLE STEP
297  330         133 GOTO   ALTIME ( 343) ALARM TIME
298  331        1463 GOTO   ACTR/S ( 277) RUN STOP
299  332        1523 GOTO   SHFTON ( 304) SHIFT ON
300  333        1323 GOTO   ACTEXJ ( 265) EXIT ALMCAT
301  334         153 GOTO   ACTM20 ( 351) UNDEFINED KEY
302
303  335 ALMMSG  343 GOTO   ALMSG  ( 371) DISPLAY ALARM MESSAGE
304  336 ACTSST    1 GOSUB  ALMSST         SINGLE STEP TO NEXT ALARM
304  337           0                       *TIMER ROM:  TM2, @1205
305  340          13 GOTO   ACTRTN ( 341) (P+1) M.X= ADDR OF NXT ALM
306  341 ACTRTN    1 GOLONG ACT120         (P+2) M.X UNCHANGED
306  342           2                       *TIMER ROM:  TM2, @0021
307
308  343 ALTIME    1 GOSUB  A-DHMS         GET&CONVERT ALM TO D.H.M.S
308  344           0                       *TIMER ROM:  TM2, @1274
309  345         340 SEL Q
310  346         134 PT=    4             DISPLAY SECONDS & TENTHS
311
312                 ENTRY  DST10
313  347 DST10     1 GOSUB  DSPTMP         DISP TIME (LEFT-JUSTIFIED)
313  350           0                       *TIMER ROM:  TM3, @0070
314  351 ACTM20 1204 S7=    0
315  352           1 GOLONG ACT180         STOP CATALOG, GET KEY PRESS
315  353           2                       *TIMER ROM:  TM2, @0121
316
317  354 ALDATE    1 GOSUB  A-DHMS         GET&CONVERT ALM TO D.H.M.S
317  355           0                       *TIMER ROM:  TM2, @1274
318  356           1 GOSUB  CLLCDE         ENABLE & CLEAR DISPLAY
318  357           0                       *MAINFRAME: CN11, @0360
```

```
   319                                          IN & ASSUME: NOTHING
   320                                          OUT: DISP ENA, RAM DISABLED
   321                                          USES: C[11:0], ACTIVE PT,
   322                                          DADD, PFAD (NO ST, +0 SUB
   323                                          LEVEL, NO ARITH MODE)
   324  360        1010 S2=    1
   325  361           1 GOSUB  DSPDT            DISPLAY THE DATE
   325  362           0                         *TIMER ROM:  TM0, @1373
   326  363        1663 GOTO   ACTM20 ( 351)
   327
   328  364 RESETI  504 S6=    0                DISP 00:00:00 = NO INTERVAL
   329  365           1 GOSUB  INTVAL           DISPLAY RESET INTERVAL
   329  366           0                         *TIMER ROM:  TM2, @1234
   330  367           0 NOP                     (P+1)
   331  370        1613 GOTO   ACTM20 ( 351) (P+2)
   332
   333  371 ALMSG     1 GOSUB  CLLCDE           ENABLE & CLEAR DISPLAY
   333  372           0                         *MAINFRAME: CN11, @0360
   334                                          SEE ABOVE COMMENTS
   335  373         410 S8=    1
   336  374           1 GOSUB  DSAMSG           DISPLAY IT
   336  375           0                         *TIMER ROM:  TM0, @0622
   337  376           0 NOP                     (P+1) NO MESSAGE
   338  377        1523 GOTO   ACTM20 ( 351) (P+2)
*****************************************************************
* ALMBST = BACK STEP TO PREVIOUS ALARM                    4-21-81 RSW
* INPUT :  M.X= ADDRESS OF CURRENT ALARM
*               (ADDRESS OF TRAILER REG IS OK AS LONG AS THERE IS AT
*               LEAST ONE ALARM IN THE BUFFER)
* ASSUME:  HEXMODE, TIMER BUFFER DOES EXIST
* OUTPUT:  B.X= C.X= M.X= ADDRESS OF PREVIOUS ALARM
*          A.S= 0
* USES:    A, B.X, C, M.X, S8, P,Q, +1 SUB LEVEL, DADD, PFAD
*          (NO TIMER CHIP ACCESS)
*
   350                  ENTRY  ALMBST
   351  400 ALMBST    1 GOSUB  SRHBUF           A.X= BUF START ADDR, A.S= 0
   351  401           0                         *TIMER ROM:  TM2, @1141
   352  402         546 A=A+1  X                A.X= ADDR OF FIRST ALARM
   353  403         206 B=A    X
   354  404 ABST10  630 C=M                     C.X= ADDR OF CURRENT ALARM
   355  405        1546 ? A#C  X                REACH CURRENT ALARM ?
   356  406          63 GONC   ABST20 ( 414) YES
   357  407         206 B=A    X
   358  410         306 C=B    X
   359  411           1 GOSUB  SKPALM           A.X= C.X= NEXT ALARM ADDR
   359  412           0                         *TIMER ROM:  TM2, @1123
   360  413        1713 GOTO   ABST10 ( 404)
   361  414 ABST20  306 C=B    X                C.X= ADDR OF PREVIOUS ALARM
   362  415         530 M=C
   363  416        1740 RTN
*****************************************************************
* WAITK6= WAIT 0.8 SEC WHILE CHECKING FOR KEY DOWN         1-15-81 RSW
*
* IN:     KEYBOARD CLEARED UNLESS IT IS NECESSARY TO RECOGNIZE PAST KEYS
*         DISPLAY TURNED ON    (DISPLAY OFF FOR "WAITKD")
* ASSUME: HEXMODE
* OUT:    KEY DETECTED - JUMPS TO ACT155
*             DISPLAY TURNED ON
*         NO KEY DETECTED - RETURNS
```

```
*            DISPLAY TURNED OFF
* USES:   C.X (NO ST, NO PT, +0 SUB LEVELS, NO DADD, NO PFAD)
*
   376                  ENTRY  WAITK6
   377                  ENTRY  WAITKD
   378  417 WAITKD 1440 DISTOG               TURN DISPLAY ON
   379  420 WAITK6  460 LDI                  LOAD LOW 12 BITS OF C WITH
   380  421        1361 CON    1777          VALUE TO WAIT 0.8 SECONDS
   381  422 WAITK  1714 CHK KB               KEY DOWN ?
   382  423           1 GOLC   ACT170        YES
   382  424           3
   383  425        1146 C=C-1  X             TIMEOUT ?
   384  426        1743 GONC   WAITK  ( 422) NO
   385  427        1340 DISOFF              TURN DISPLAY BACK OFF
   386  430        1740 RTN                 YES
*************************************************************************
* TMSG = TIMER MESSAGE
*  PRINTS THE DISPLAY IN NORM & TRACE, AND SETS MESSAGE FLAG
*
* IN & ASSUME: HEXMODE
* OUT: CHIP 0 ENABLED, PERIPHERALS DISABLED, STATUS SET 0 UP
* USES: A,C,G,N,B[12] FOR PIL PRINTER, S0-S8, ACTIVE PT, +2 SUB LEVELS
*
   395                  ENTRY  TMSG
   396  431 TMSG    410 S8=    1             PRINT & SET MESSAGE FLAG
   397  432           1 GOLONG MSG105        PRINT DISP IN NORM & TRACE
   397  433           2                      *MAINFRAME:  CN7, @0200
   398                                        AND SET MESSAGE FLAG
   399                                       IN & ASSUME: S8=1, HEXMODE
   400                                       OUT: CHIP 0 ENA, PERIPH DIS
   401                                            STATUS SET 0 UP
   402                                       USES: A,C,G,N,B[12] FOR PIL
   403                                             PRINTER, S0-S7,
   404                                             ACTIVE PT, +2 SUB LVL
*
*************************************************************************
*                                                              1-5-81 RSW
* KEY-FC : READ THE KEY CODE TO MATCH THE KEY CODE IN KEY CODE TABLE
*          AND JUMP TO THE CORRESPONDING POINT AFTER THE KEY CODE
*          TABLE.
*
* IN:      KEY DOWN, C[1:0]= TABLE LENGTH - 1, C.XS= 0
*      !!! NOTE !!!! THE LAST CONSTANT IN THE KEY CODE TABLE MUST BE 0
*          TO MARK THE END OF THE TABLE.
* ASSUME:  HEXMODE
* OUT:     C.X= 0 (NEEDED FOR DIGIT KEYS)
* USED:    A[X&M], C[X&M]
*          (NO  ST, NO PT, NO DADD, NO PFAD, +0 SUB LVLS, NO TIMER CHIP)
*
*
* TIME= 7 (KEY POSITION IN TABLE) + 12  [INCLUDING GSB & RTN]
   422                  ENTRY  KEY-FC
   423  434 KEY-FC  132 C=0    M
   424  435         674 RCR    11            C[4:3]= TABLE LENGTH
   425  436         432 A=C    M
   426  437        1040 C=KEYS
   427  440          74 RCR    3
   428  441         406 A=C    X             A.X= KEY CODE
   429  442         660 C=STK                GET ADDR OF KEY CODE TABLE
   430  443 KYFC10 1460 CXISA                LOAD A KEY CODE FROM TABLE
```

```
   431  444        1072 C=C+1  M
   432  445        1346 ? C#0  X             REACH END OF TABLE ?
   433  446          43 GONC   KYFC30 ( 452) YES, UNDEFINED KEY
   434  447        1546 ? A#C  X             MATCH KEY CODE IN TABLE ?
   435  450        1737 GOC    KYFC10 ( 443) NO
   436  451         106 C=0    X
   437  452 KYFC30 1032 C=A+C  M
   438  453         740 GOTOC
   439
*************************************************************************
*                                                              4-21-81 RSW
* 36000 - LOAD CONSTANT 0 0000036000 000 TO REG C
*
* IN:     A= POSITIVE STOPWATCH TIME IN 100TH'S OF SECONDS FORMAT
* ASSUME: NOTHING
* OUT:    DEC MODE, PT= 5, C= 0 0000036000 000
*         A= (STOPWATCH TIME) MOD (100 HOURS) WITH 1 TOO MANY SUBTRACTS
* USES:   C, ACTIVE PT, ARITH MODE
*         (NO ST, NO DADD, NO PFAD, NO TIMER CHIP ACCESS, +0 SUB LVLS)
*
*
*
   453                  ENTRY  36000
   454  454 36000  1240 SETDEC
   455  455        1234 PT=    7
   456  456         116 C=0
   457  457         320 LC     3
   458  460         620 LC     6
   459  461 36000A  716 A=A-C                SUBTRACT 100 HOURS
   460  462        1773 GONC   36000A ( 461)
   461  463        1740 RTN
*************************************************************************
* RNGERR = RANGE ERROR                                          4-7-81 RSW
* RETURNS ONLY IF THE RANGE ERROR IGNORE FLAG IS SET
*
* IN & ASSUME: CHIP 0 ENABLED, PERIPHERALS DISABLED
* OUT:  INPUT S0-S7 PRESERVED IN C[1:0]
* USES: C, S0-S7
*       (NO PT, +0 SUB LEVELS, NO ARITH MODE, NO TIMER CHIP ACCESS)
*
   471                  ENTRY  RNGERR
   472  464 RNGERR 1670 C=REGN 14
   473  465         574 RCR    6
   474  466        1730 CST EX
   475  467        1214 ?S7=1               RANGE ERR IGNORE FLAG SET ?
   476  470        1540 RTN C               YES
   477  471           1 GOLONG ERROF        NO, "OUT OF RANGE"
   477  472           2                     *MAINFRAME:  CN0, @0242
   478                                      IN:     NOTHING
   479                                      ASSUME: NOTHING
   480                                      !! DOES NOT RETURN !!
   481
*************************************************************************
* XYZALM                                                        2-19-81 RSW
*************************************************************************
* XYZALM = FULLY PROGRAMMABLE SET ALARM FUNCTION
*
* IN:    X-REG= ALARM TIME IN H.MS FORM
*        Y-REG= ALARM DATE IN M.DY (D.MY) FORM
*        Z-REG= ALARM AUTO-RESET INTERVAL IN H.MS FORM
```

```
*        ALPHA REG= ALARM MESSAGE
*
* IF ALARM DATE= 0 THEN XYZALM WILL USE TODAY'S DATE
*                 (ILLEGAL DATE GIVES "DATA ERROR Y")
* IF Z-REG IS ZERO THERE WILL BE NO RESET INTERVAL
*                 (ILLEGAL INTERVAL GIVES "DATA ERROR")
* IF ALPHA REG IS CLEAR THE ALARM WILL HAVE NO MESSAGE
*
*
*
   500  473          215 CON    @215          M
   501  474           14 CON    @14           L
   502  475            1 CON    @01           A
   503  476           32 CON    @32           Z
   504  477           31 CON    @31           Y
   505  500           30 CON    @30           X
   506                 ENTRY  XYZALM
   507  501 XYZALM   460 LDI                   LOAD LOW 12 BITS OF C WITH
   508  502            2 CON    2             2 TO TAKE DATE FROM Y-REG
   509  503            1 GOSUB  CHECK         ERROR IF ALPHA DATE
   509  504            0                      *TIMER ROM:  TM0, @0242
   510
   511                                        M= M.DY (D.MY) DATE
   512  505            1 GOSUB  CHKXM         ERROR IF X= ALPHA DATA
   512  506            0                      *TIMER ROM:  TM0, @0237
   513  507          160 N=C                  N= H.MS TIME
   514  510            1 GOSUB  INITMR        INITIALIZE TIMER IF NEEDED
   514  511            0                      *TIMER ROM:  TM0, @1524
   515  512         1750 PT=A
   516  513           70 RDTIME              C= CURRENT TIME
   517  514         1610 S0=    1             USE Y FOR DATE COMPARE
   518  515         1410 S1=    1             ADD X OR Y TO "DATA ERROR"
   519  516            1 GOSUB  C=T+D0        C= TIME + DATE IN SECONDS
   519  517            0                      *TIMER ROM:  TM1, @0223
   520  520         1574 RCR    12            C= SSSSSSSSSSCC00
   521  521          106 C=0    X
   522  522         1356 ? C#0                 WHOLE REGISTER = 0 ?
   523  523           27 GOC    XYZA20 ( 525) NO, IT'S OK
   524  524         1072 C=C+1  M             YES, ADD 0.1 SEC TO TIME TO
   525                                        AVOID 67/97 CARD READ BUG
   526  525 XYZA20   160 N=C                  N= SSSSSSSSSSC000
   527  526         1604 S0=    0             ASSUME NO RESET INTERVAL
   528  527         1046 C=C+1  X             C.X= 1= ADDRESS OF Z REG
   529                 LEGAL                  (CLEAR THE CARRY FLAG)
   530  530            1 GOSUB  CHECK         ERROR IF Z= ALPHA DATA
   530  531            0                      *TIMER ROM:  TM0, @0242
   531  532          416 A=C
   532  533         1356 ? C#0                 ANY RESET INTERVAL GIVEN ?
   533  534          153 GONC   XYZA40 ( 551) NO
   534  535         1610 S0=    1             THERE IS A RESET INTERVAL
   535  536            1 GOSUB  HMSEC1        ->SECS (A= 0000SSSSSSSSCC)
   535  537            0                      *TIMER ROM:  TM1, @0346
* THE MAXIMUM RESET INTERVAL OF 9999.595999 FITS IN 8 DIGITS OF SECONDS
   537  540           53 GOTO   XYZA30 ( 545) (P+1) INTERVAL O.K.
   538  541 XYZA25   460 LDI                   (P+2) BAD INTERVAL, "ERROR"
   539  542           32 CON    @32           Z
   540  543            1 GOLONG TERR20        PRINT DATA ERROR
   540  544            2                      *TIMER ROM:  TM0, @0274
   541  545 XYZA30   116 C=0
   542  546         1072 C=C+1  M
```

```
   543  547         1416 ? A<C                 RESET INTERVAL < 10 SECS ?
   544  550         1717 GOC    XYZ125 ( 541) YES, ERROR
   545  551 XYZA40   256 AC EX                 C= 0 000SSSSSSS SCC
   546  552         1574 RCR    12             C= 00SSSSSSSSCC00
   547  553          106 C=0    X              REMOVE 100TH'S OF SECONDS
   548  554         1150 REGN=C 9              REG 9= RESET INTERVAL
*
*          STORE AN ALARM
*
* IN:     N[13:3] = ALARM TIME IN 10TH'S OF SECONDS SINCE JAN. 1, 1900
*         N.X = 000
*    IF S0 = 0 NO AUTO-RESET INTERVAL WITH THIS ALARM
*    IF S0 = 1 THE AUTO-RESET INTEERVAL IS STORED IN REG.9
* PROCEDURE TO STORE AN ALARM :
* 1. FIND OUT AFTER THE CHAIN HEAD IF THERE IS ENOUGH EMPTY MEMORY
*    TO EXPAND THE TIMER I/O BUFFER. THE REQUIRED EMPTY MEMORY = 1 +
*    ALARM MESSAGE LENGTH IN REGISTERS.
* 2. SEARCH THE FIRST REGISTER OF THE TIMER BUFFER. IF NOT FOUND,
*    INITIALIZE AN I/O BUFFER WITH THE SIZE OF 2 REGISTERS. THIS
*    INITIAL TIMER I/O BUFFER DOESN'T HAVE ANY ALARM IN IT.
* 3. GO THRU THE ALARM STACK TO DETERMINE WHERE TO STORE THIS NEW ALARM.
* 4. LIFT UP A HOLE IN THE TIMER BUFFER FOR STORING THE NEW ALARM TIME
*    AND ITS MESSAGE, IF IT HAS ANY.
* 5. STORE THE ALARM AND ITS MESSAGE INTO THE HOLE JUST GENERATED.
*    THE MESSAGE IS STORED THE SAME WAY AS IT IS IN THE ALPHA REGISTER.
*.....................................................................
*    THE EXP FIELD OF THE ALARM TIME CARRIES THE FOLLOWING INFORMATION:
*    DIGIT 0 : LENGTH OF ALARM MESSAGE IN REGISTERS
*    DIGIT 2 : NON-ZERO IF THE ALARM HAS AN AUTO-RESET INTERVAL
*
*
*
   575  555         1140 SETHEX
   576  556            1 GOSUB  FNDMSG         FIND 1ST NON-NULL IN ALPHA
   576  557            0                       *TIMER ROM:  TM0, @1202
   577  560          260 C=N
   578  561          460 LDI                   LOAD LOW 12 BITS OF C WITH
   579  562            4 CON    4              C= SSSSSSSSSSC004
   580  563         1106 C=A-C  X              C= A= # REG OF MESSAGE
   581  564          406 A=C    X
   582  565         1614 ?S0=1                 ANY AUTO INCREMENT?
   583  566           33 GONC   STA100 ( 571) NO
   584  567          546 A=A+1  X              YES, 1 MORE REG REQUIRED
   585  570         1066 C=C+1  XS             MARK AUTO RESET ALARM
   586  571 STA100   160 N=C                    N= ALARM & INFO
   587  572          546 A=A+1  X              COUNT ALARM REG
   588  573          246 AC EX  X              C.X= # OF REG REQUIRED
   589  574          530 M=C
   590  575            1 GOSUB  MEMLFT         CHECK MEMORY LEFT
   590  576            0                       *MAINFRAME:  CN1, @0641
   591                                         IN: HEXMODE
   592                                         ASSUME: HEXMODE
   593                                         OUT: C.X= MEMORY LEFT
   594                                         USES: A,C, +1 SUB LEVEL
   595                                               (NO ST, NO PT)
   596  577          406 A=C    X              A.X= # UNUSED REG LEFT
   597  600          630 C=M                   C.X= # OF REG REQUIRED
   598  601         1406 ? A<C  X              IS THERE A SPACE PROBLEM ?
   599  602          147 GOC    NOROOM ( 616) YES
   600  603         1610 S0=    1              ASSUME ENOUGH ROOM
```

```
   601  604         1046 C=C+1  X              MAY NEED 2 REGS FOR BUFFER
   602  605         1046 C=C+1  X               HEADER & TRAILER
   603  606         1406 ? A<C  X              IS THERE A SPACE PROBLEM ?
   604  607           23 GONC   STA150 ( 611) NO, STILL OK
   605  610         1604 S0=    0              YES, NO ROOM TO CREATE BUF
   606  611 STA150     1 GOSUB  SRHBUF         SEARCH FOR TIMER BUFFER
   606  612            0                       *TIMER ROM:  TM2, @1141
   607                                         A.X= BEGINNING OF BUFFER
   608  613          443 GOTO   STA200 ( 657) (P+1) BUFFER FOUND
*
*
*          SET UP THE TIMER ROM I/O BUFFER IN THE UNUSED MEMORY
*
* IN:  HEXMODE, PERIPHERALS DISABLED
*      A.X= ADDRESS OF FIRST UNUSED REGISTER AFTER I/O BUFFERS
*
*         BUFFER SIZE  = 2 REGISTERS
*         1ST REGISTER = AA020000000000
*         2ND REGISTER = F0000000000000
*
*
*
   622  614         1614 ?S0=1                 ROOM TO CREATE BUFFER ?
   623  615          227 GOC    STA160 ( 637) YES
   624
   625
* NOROOM  ---  IF THE ERROR IGNORE FLAG IS CLEARED, PUT UP "NO ROOM"
*              MESSAGE AND JUMP TO ERR110 IN 41C MAINFRAME.
*
* IN & ASSUME: NOTHING
*                    !!! DOES NOT RETURN !!!
*
   632                   ENTRY  NOROOM
   633  616 NOROOM     1 GOSUB  ERRSUB         RTN ONLY IF ERR IGNORE = 0
   633  617            0                       *MAINFRAME:  CN8, @1350
   634                                         IN & ASSUME: NOTHING
   635                                         OUT: HEXMODE, CHIP 0 ENABLE
   636                                          S8=1, FLAGS CLEARED: DATA
   637                                          ENTRY, CATALOG, SHIFT, MSG
   638                                         USES: C, S0-S8, DADD, PFAD,
   639                                          +2 SUB LVLS, ARITH MODE
   640                                          (NO PT)
   641  620            1 GOSUB  CLLCDE         ENABLE AND CLEAR DISPLAY
   641  621            0                       *MAINFRAME: CN11, @0360
   642  622            1 GOSUB  MESSL          SHIFT MSG LEFT INTO DISPLAY
   642  623            0                       *MAINFRAME:  CN1, @1757
   643                                         IN & ASSUME: DISP ENABLED
   644                                                      RAM DISABLED
   645                                                      HEXMODE
   646                                         OUT: MESSAGE TO DISPLAY
   647                                         USES: C[6:0] ONLY
   648  624           16 CON    @16            N
   649  625           17 CON    @17            O
   650  626           40 CON    @40              SPACE
   651  627           22 CON    @22            R
   652  630           17 CON    @17            O
   653  631           17 CON    @17            O
   654  632         1015 CON    @1015          M
   655  633          470 FLLABC                ROTATE DISP LEFT 4 CHARS
   656  634         1770 RABCL                 LEFT JUSTIFY DISPLAY
```

```
   657  635           1 GOLONG TERR50        PRT MSG FLAG, ERROR EXIT
   657  636           2                      *TIMER ROM:  TM0, @0310
   658
   659  637 STA160  206 B=A    X            B.X= ADDR OF 1ST UNUSED REG
   660  640         306 C=B    X
   661  641        1046 C=C+1  X
   662  642        1160 DADD=C
   663  643         116 C=0
   664  644        1176 C=C-1  S            C.S= F (TO MARK LAST REG)
   665  645        1360 DATA=C
   666  646         306 C=B    X            C.X= FIRST REGISTER ADDRESS
   667  647        1160 DADD=C
   668  650         116 C=0
   669  651        1334 PT=    13
   670  652        1220 LC     10
   671  653        1220 LC     10
   672  654          20 LC     0
   673  655         220 LC     2
   674  656        1360 DATA=C
*
* 1. MAKE SURE THE TIMER BUFFER WILL NOT BE OVERFLOWED AND GET THE
*    ADDRESS OF THE LAST REGISTER OF LAST I/O BUFFER
*
   679  657 STA200    4 S3=    0            NOT PURGING ALARM
   680  660           1 GOSUB  CHKBUF       UPDATE TIMER BUFFER LENGTH
   680  661           0                      *TIMER ROM:  TM2, @1052
   681                                       B.X= HEADER REG OF TMR BUFF
   682  662         530 M=C                  M.X= LAST REG OF I/O BUFFER
*
* 2. DETERMINE WHERE TO INSERT THE NEW ALARM INTO THE ALARM STACK
*
   686  663         260 C=N                  C= ALARM TIME & INFO
   687  664         306 C=B    X             C.X= ADDR OF 1ST REGISTER
   688  665        1046 C=C+1  X             C.X= ADDR OF 1ST ALARM
   689  666         416 A=C                  A= ALM TIME & 1ST ALM ADDR
   690  667           1 GOSUB  NEWLOC        A.X= PLACE TO INSERT ALARM
   690  670           0                      *TIMER ROM:  TM2, @1111
*
* IN: A.X= ADDR TO STORE AT,  M= LAST REG & ETC
* NEED TO LIFT OTHER I/O BUFFERS EVEN IF STORING AT END OF TIMER BUFFER
*
   695  671         206 B=A    X             B.X= START LIFTING ADDRESS
   696  672         630 C=M                  C.X= LAST REG OF LAST BUF
   697  673         406 A=C    X
   698  674         260 C=N                  C= ALARM & INFO
   699  675           1 GOSUB  SKPAL1        A.X= NEW ADDR OF LAST REG
   699  676           0                      *TIMER ROM:  TM2, @1125
   700  677         630 C=M                  C.X= OLD LAST BUF LAST REG
   701  700         246 AC EX  X             A.X= OLD LAST REG ADDRESS
   702  701         530 M=C                  M.X= NEW LAST REG ADDRESS
*
* OPEN AN EMPTY SPACE IN THE TIMER I/O BUFFER
*
*
* IN:  A.X= ADDRESS OF LAST REGISTER IN LAST I/O BUFFER
*      M.X= NEW ADDRESS FOR LAST REG IN LAST I/O BUFFER
*      B.X= ADDRESS OF LOWEST ADDRESSED REG TO BE MOVED
*
*
   712  702 LFTMEM  246 AC EX  X
```

```
   713  703        1160 DADD=C
   714  704        1146 C=C-1  X
   715  705         406 A=C    X
   716  706          70 C=DATA
   717  707         730 CM EX
   718  710        1160 DADD=C
   719  711        1146 C=C-1  X
   720  712         730 CM EX
   721  713        1360 DATA=C
   722  714        1446 ? A<B  X             DONE LIFTING ?
   723  715        1653 GONC   LFTMEM (702)  NO
*
*
* 4. STORE THE NEW ALARM TIME AND ITS MESSAGE INTO THE EMPTY SPACE
*
   728  716         306 C=B    X             C.X= INSERTING ADDRESS
   729  717        1160 DADD=C
   730  720        1046 C=C+1  X
   731  721         530 M=C
   732  722         260 C=N                  C= ALARM & INFORMATION
   733  723        1366 ? C#0  XS            HAS AUTO-RESET INTERVAL ?
   734  724         133 GONC   STA325 ( 737) NO
   735  725        1360 DATA=C               STORE THE ALARM TIME
   736  726         106 C=0    X
   737  727        1160 DADD=C
   738  730        1170 C=REGN 9             LOAD RESET INT FROM REG.9
   739  731         416 A=C
   740  732         630 C=M
   741  733        1160 DADD=C
   742  734        1046 C=C+1  X
   743  735         530 M=C
   744  736         256 AC EX                C= AUTO-RESET INTERVAL
   745  737 STA325 1360 DATA=C
   746  740         260 C=N                  C[0]= MSG LEN IN REGISTERS
   747  741        1634 PT=    0
   748  742        1142 C=C-1  PT            MESSAGE LENGTH = 0 ?
   749  743         237 GOC    STA350 ( 766) YES, ALL DONE
   750  744           1 GOSUB  FNDMSG        FIND 1ST CHAR IN ALPHA REG
   750  745           0                      *TIMER ROM:  TM0, @1202
   751  746 STA340 1356 ? C#0                ANY CHARS IN THIS REG ?
   752  747          27 GOC    STA342 ( 751) YES
   753  750        1072 C=C+1  M             NO, SO MAKE IT NON-ZERO
   754
*  NOTE: THE ADVANCED PROGRAMMING ROM ALLOWS LONG EMBEDDED NULL STRINGS
*        IN THE ALPHA REGISTER. DUE TO THE 67/97 CARD READER BUG, AN I/O
*        BUFFER CANNOT HAVE ANY REGISTERS = 0, SO WHOLE REGISTERS OF
*        NULLS MUST BE MODIFIED TO BE NON-ZERO.
   759  751 STA342  730 CM EX                M= MESSAGE REGISTER
   760  752        1160 DADD=C               C.X= BUFF ADDR TO STORE MSG
   761  753        1046 C=C+1  X
   762  754         730 CM EX
   763  755        1360 DATA=C
   764  756 STA345  676 A=A-1  S             ALL MESSAGE MOVED ?
   765  757          77 GOC    STA350 ( 766) YES, ALL DONE
   766  760         246 AC EX  X
   767  761        1146 C=C-1  X
   768  762        1160 DADD=C
   769  763         406 A=C    X
   770  764          70 C=DATA               LOAD NEXT MESSAGE REGISTER
   771  765        1613 GOTO   STA340 ( 746)
```

```
   772
   773  766 STA350    1 GOLONG ADJ100          SET NEW HARDW ALARM & BEEP
   773  767           2                        *TIMER ROM:  TM1, @0164
   774                                         IF ANY PAST DUE ALARMS
*********************************************************************
*  T+X                                                    2-23-81 RSW
*********************************************************************
   778  770         230 CON    @230            X
   779  771          53 CON    @53             +
   780  772          24 CON    @24             T
   781                ENTRY  T+X
   782  773 T+X       1 GOSUB  CHECKX          ERROR IF X= ALPHA DATA
   782  774           0                        *TIMER ROM:  TM0, @0240
   783  775           1 GOSUB  R9=T            C= REG9= CLOCK TIME, A=0
   783  776           0                        *TIMER ROM:  TM2, @1463
   784  777         456 A=A+B                  A= B= ENTERED  TIME
   785 1000         530 M=C                    M= CLOCK TIME
   786 1001           1 GOSUB  HMSEC1          H.MS TIME TO 100TH'S OF SEC
   786 1002           0                        *TIMER ROM:  TM1, @0346
   787 1003          33 GOTO   PLUS20 (1006)   (P+1) ENTRY LEGAL
   788                                         (P+2)
   789 1004           1 GOLONG ERRDE           *DATA ERROR*
   789 1005           2                        *MAINFRAME: CN10, @0055
   790 1006 PLUS20 1240 SETDEC
   791 1007        1336 ? B#0  S               SUBTRACT ?
   792 1010          33 GONC   PLUS25 (1013)   NO, ADD
   793 1011        1216 C=-C
   794 1012         416 A=C                    A= ENTERED INTERVAL
   795 1013 PLUS25  160 N=C                    N= ENTERED INTERVAL
   796 1014         630 C=M                    C= CURRENT TIME
   797 1015         516 A=A+C
   798 1016        1536 ? A#0  S               RESULTING YEAR < 1900 ?
   799 1017          63 GONC   PLUS30 (1025)   NO, OK
   800
   801 1020          16 A=0                    SET TO 1/1/1900
   802 1021 PLUSER    1 GOSUB  RNGERR          RTN IF RANGE ERR IGNORE= 1
   802 1022           0                        *TIMER ROM:  TM2, @0464
   803 1023         256 AC EX                  C= TIME TO SET
   804 1024         163 GOTO   PLUS40 (1042)
   805 1025 PLUS30  116 C=0
   806 1026         634 PT=    11
   807 1027        1120 LC     9               (BEG OF 300 YR= 1/1/1900)
   808 1030         420 LC     4                12/31/2199= 109572 DAYS+1
   809 1031         620 LC     6                     = 9467107200 SECONDS
   810 1032         720 LC     7
   811 1033         120 LC     1
   812 1034          20 LC     0
   813 1035         720 LC     7
   814 1036         220 LC     2
   815 1037         256 AC EX                  A= MAX, C= NEW TIME
   816 1040        1416 ? A<C                  MAX < NEW TIME ?
   817 1041        1607 GOC    PLUSER (1021)   YES, "OUT OF RANGE"
   818 1042 PLUS40    1 GOSUB  T=T+TP          SET NEW TIME
   818 1043           0                        *TIMER ROM:  TM3, @0321
   819 1044         470 RDSCR
   820 1045         416 A=C                    A= LAST TIME SET
   821 1046         260 C=N                    C= ENTERED INTERVAL
   822 1047        1016 C=A+C
   823 1050         450 WRSCR                  UPDATE "LAST TIME SET"
   824 1051        1153 GOTO   STA350 ( 766)   BEEP 2X IF PAST DUE ALARMS
```

```
* IF THE TIME IS READ WITHIN 1/100 SEC OF THE WRITE & CORRECT IN
* "T=T+TP" IT MAY  BE 1/100 SECOND LOW !!!!!!!
*
*********************************************************************
*                                                          12-11-80 RSW
* CHKBUF - CHECK IF THE BUFFER SIZE WILL OVERFLOW AFTER ADDING A NEW
*          ALARM AND UPDATE THE BUFFER SIZE. CHECK IF THERE IS ANY
*          OTHER I/O BUFFER ON TOP OF TIMER I/O BUFFER.
*
* INPUT:   A.X= ADDRESS OF FIRST REGISTER OF THE TIMER I/O BUFFER
*          N.X= EXPONENT FIELD OF ALARM RESTIER
*          S3= 1  PURGING THE ALARM
*          S3= 0  ADDING AN ALARM
* ASSUME:  PERIPHERALS DISABLED, HEXMODE
* OUTPUT:  C.X= ADDRESS OF LAST REGISTER OF LAST I/O BUFFER
*          B.X= ADDRESS OF FIRST REGISTER OF THE TIMER BUFFER
*          TIMER ROM I/O BUFFER SIZE WILL BE UPDATED BY ADDING
*          (SUBTRACTING FOR S3=1) THE REQUIRED REGS TO ORIG BUFF SIZE.
*          RETURN TO "NO ROOM" ROUTINE IF BUFFER SIZE WILL OVERFLOW
*
* USES:    A, B.X, C, S8, ACTIVE PT, DADD, +1 SUB LEVEL
*          (NO TIMER CHIP ACCESS)
*
*
*
  850                    ENTRY  CHKBUF
  851 1052 CHKBUF  206 B=A    X             B.X= ADDR OF 1ST REG IN BUF
  852 1053           6 A=0    X
  853 1054         260 C=N
  854 1055           1 GOSUB  SKPAL1        A.X= ALARM LENGTH
  854 1056           0                      *TIMER ROM:  TM2, @1125
  855 1057         306 C=B    X
  856 1060        1160 DADD=C
  857 1061          70 C=DATA               C= 1ST REG IN TIMER BUFFER
  858 1062         374 RCR    10
  859 1063        1434 PT=    1
  860 1064         252 AC EX  WPT            A.X= OLD BUFFER LENGTH
  861                                        C[1:0]= ALARM LENGTH
  862 1065          14 ?S3=1                 PURGING ALARM ?
  863 1066          57 GOC    CHKB20 (1073) YES
  864 1067        1012 C=A+C  WPT            C[1:0]= NEW BUFFER LENGTH
  865 1070           1 GOLC   NOROOM         BUFFER OVERFLOW
  865 1071           3                       *TIMER ROM:  TM2, @0616
  866 1072          23 GOTO   CHKB30 (1074)
  867 1073 CHKB20 1112 C=A-C  WPT            ADJUST BUFFER LENGTH
  868 1074 CHKB30  174 RCR    4
  869 1075        1360 DATA=C               UPDATE BUFFER LENGTH
  870 1076         446 A=A+B  X
  871 1077         246 AC EX  X              C.X= LAST TMR BUF REG + 1
*  C.X MUST POINT THERE SO THE "? A#C  PT" TEST IN "FNDEOB" WON'T RUN
  873 1100         404 S8=    0              DON'T ALLOW RETURN TO P+2
  874 1101           1 GOSUB  FNDEOB         A.X= (LST REG OF LST I/O)+1
  874 1102           0                       *TIMER ROM:  TM2, @1151
  875 1103         646 A=A-1  X
  876 1104         246 AC EX  X              C.X= ADDR OF LAST REG OF
  877                                              LAST I/O BUFFER
  878 1105        1740 RTN
*********************************************************************
* NEWLOC= NEW LOCATION (FOR ALARM)                        12-15-80 RSW
* NEWLSK= NEW LOCATION, SKIP FIRST ALARM
```

```
* GIVEN AN ALARM TIME IN B[13:3], FINDS THE PLACE IN THE ALARM STACK
* TO INSERT THE NEW ALARM.
*
* IN:   (NEWLOC) A.X= C.X= ADDRESS AT WHICH TO START SEARCHING
*                (TRAILER REG ADDR IS OK, OTHERWISE IT MUST BE A VALID
*                 ALARM ADDRESS)
*       (NEWLSK) SAME EXCEPT SETS A.X= C.X, AND SKIPS THAT ALARM
*                BEFORE STARTING THE SEARCH, SO C.X MUST BE A VALID
*                ALARM ADDRESS.
* ASSUME:   A[13:3]= NEW ALARM TIME IN TENTHS OF SECONDS
*           Q= 13, P SELECTED, HEXMODE, PERIPHERALS DISABLED
* OUT:      A.X= ADDRESS OF TRAILER REGISTER OF ALARM STACK
*       OR  A.X= ADDRESS OF FIRST ALARM > NEW ALARM
*           PT= 3
* USES:     A, C, ACTIVE PT, +1 SUB LEVEL, DADD
*           (NO ST, NO PFAD, NO TIMER CHIP ACCESS)
*
   899                  ENTRY  NEWLSK
   900                  ENTRY  NEWLOC
   901 1106 NEWL00  246 AC EX  X             C.X= CURRENT ALARM ADDRESS
   902 1107 NEWLSK    1 GOSUB  SKPALC        GET A.X= C.X= NEXT ALM ADDR
   902 1110           0                      *TIMER ROM:  TM2, @1122
   903 1111 NEWLOC 1160 DADD=C
   904 1112          70 C=DATA
   905 1113          34 PT=    3
   906 1114        1076 C=C+1  S             END OF BUFFER ?
   907 1115        1540 RTN C                YES, SHOW LAST ALARM
   908 1116        1176 C=C-1  S             RESTORE C.S
   909 1117        1422 ? A<C  PQ            GIVEN ALARM < CURR ALARM ?
   910 1120        1663 GONC   NEWL00 (1106) YES, KEEP LOOKING
   911 1121        1740 RTN                  NO, DONE
*
*******************************************************************************
*                                                                    1-9-81 RSW
* SKPALM - ADVANCE THE ADDRESS OF CURRENT ALARM TO NEXT ALARM ADDRESS
*
* INPUT :  (SKPALM) A.X= C.X= CURRENT ALARM ADDRESS
*          !! NOTE - ALARM ADDRESS MUST A REAL ALARM ADDRESS, NOT THE
*                    ADDRESS OF THE TRAILER REGISTER OR OTHER REGISTER.
*          (SKPAL1) A.X= CURRENT ALM ADDR, C.X= EXP FIELD OF CURR ALARM
*
* ASSUME:  HEXMODE, PERIPHERALS DISABLED
* OUT:     A.X= C.X= NEXT ALM ADDR, (CURR ALM REG ENABLED FOR SKPALM)
*          !! THIS MAY BE THE ADDRESS OF THE TRAILER REGISTER !!
* USES:    A.X, C, DADD
*          (NO PT, NO ST, +0 SUB LEVELS, NO PFAD, NO TIMER CHIP ACCESS)
*
*
   929                  ENTRY  SKPALC
   930                  ENTRY  SKPALM
   931                  ENTRY  SKPAL1
   932 1122 SKPALC  406 A=C    X
   933 1123 SKPALM 1160 DADD=C
   934 1124          70 C=DATA               LOAD CURRENT ALARM
   935 1125 SKPAL1 1366 ? C#0  XS            ALARM HAS RESET INTERVAL ?
   936 1126          23 GONC   SKPAL2 (1130) NO
   937 1127         546 A=A+1  X             YES, COUNT THE INTERVAL REG
   938 1130 SKPAL2  546 A=A+1  X             COUNT TIME & DATE REGISTER
   939 1131        1474 RCR    1             C.S= NUMBER OF MESSAGE REGS
   940 1132         106 C=0    X             CLR RESET INTERVAL FLAG AND
```

```
   941                                          "ALREADY ACKNOWLEDGED" FLAG
   942 1133        1374 RCR    13
   943 1134        1006 C=A+C  X                C.X= ADDRESS OF NEXT ALARM
   944 1135         406 A=C    X
   945 1136        1740 RTN
*
*****************************************************************************
*                                                                    4-2-81 RSW
* SRHBUF - SEARCH TIMER ROM I/O BUFFER IN THE USER MEMORY AFTER
*          CHAIN HEAD
*
* IN & ASSUME: HEXMODE
* OUT:         P SELECTED, P = 12, PERIPHERALS DISABLED
*              Q = 13
*       RETURN TO P+1 IF I/O BUFFER FOUND
*              A.S= 0
*              A.X= ADDR OF 1ST REG OF TIMER BUFFER, WITH REG ENABLED
*              C= CONTENTS OF FIRST REGISTER OF BUFFER
*       RETURN TO P+2 IF I/O BUFFER NOT FOUND
*              IF C.X= 0, THEN A.X= ADDR OF 1ST UNUSED REG AFTER I/O
*
* USED:        A,C, S8, P, Q, DADD, PFAD
*              (NO TIMER CHIP ACCESS, +0 SUB LEVELS)
*
*****************************
* SRHBFI - SAME AS SRHBUF EXCEPT IT INITIALIZES THE TIMER CHIP IF THE
*          POWER LOST STATUS BIT IS SET OR THE WARM START CONSTANT IS
*          NOT THERE.
*
* IN & ASSUME: NOTHING
* OUT:     P SELECTED, P= 12, Q= 13, PERIPHERALS DISABLED, HEXMODE
*          (P+1) & (P+2) OUTPUT SAME AS SRHBUF
* USES:    A,C, S0-S8, P,Q, +2 SUB LEVELS, DADD, PFAD, ARITH MODE
*
   975                  ENTRY  SRHBFI
   976                  ENTRY  SRHBUF
   977 1137 SRHBFI    1 GOSUB  INITMR        INIT TIMER IF NECESSARY
   977 1140           0                      *TIMER ROM:  TM0, @1524
   978 1141 SRHBUF  116 C=0
   979 1142        1760 PFAD=C
   980 1143         340 SEL Q
   981 1144        1334 PT=    13            Q= 13
   982 1145         240 SEL P
   983 1146         460 LDI                   LOAD LOW 12 BITS OF C WITH
   984 1147         300 CON2   12     0       ADDR OF LOW END OF MEMORY
   985 1150         410 S8=    1              ALLOW RETURN TO P+2
*
*                                                              12-11-80 RSW
* THE FOLLOWING COMMENTS APPLY WHEN "FNDEOB" IS CALLED FROM "CHKBUF"
*
* IN:      C.X= (ADDRESS OF LAST REG OF TIMER BUFFER) + 1
*          S8= 0 TO PREVENT RETURN TO P+2
* ASSUME: HEXMODE, PERIPHERALS DISABLED
* OUT:     A.X= (ADDRESS OF LAST REG IN LAST I/O BUFFER) + 1
* USES:    A,C, ACTIVE PT, DADD  (NO TIMER CHIP ACCESS, +0 SUB LEVELS)
*
   996                  ENTRY  FNDEOB
   997 1151 FNDEOB 1534 PT=    12
   998 1152        1220 LC     10
   999 1153        1534 PT=    12
```

```
  1000 1154          416 A=C
  1001 1155           23 GOTO   SRBF10 (1157)
  1002 1156 SRBF08   546 A=A+1  X              POINT TO NEXT REGISTER
  1003 1157 SRBF10   116 C=0
  1004 1160         1160 DADD=C                ENABLE CHIP 0
  1005 1161         1570 C=REGN 13             C.X= LOW ADDR OF PROG MEM
  1006 1162         1406 ? A<C  X              REACHED PROGRAM MEMORY YET?
  1007 1163          173 GONC   SRBF30 (1202) YES, 1/O BUFFER NOT FOUND
  1008 1164          246 C=A    X
  1008 1165          406                       (INSERTED BY ASSEMBLER)
  1009 1166         1160 DADD=C
  1010 1167           70 C=DATA
  1011 1170         1076 C=C+1  S              IS THIS A KEY REGISTER ?
  1012 1171         1657 GOC    SRBF08 (1156) YES
  1013 1172         1176 C=C-1  S              RESTORE C.S (FOR C=0 TEST)
  1014 1173         1542 ? A#C  PT             IS THIS 1ST TIMER BUF REG ?
  1015 1174         1640 RTN NC                YES, WE FOUND TIMER BUFFER
  1016 1175          374 RCR    10
  1017 1176          126 C=0    XS             C.X= BUFFER SIZE
  1018 1177          506 A=A+C  X              JUMP OVER THIS I/O BUFFER
  1019 1200         1356 ? C#0                 IS THIS AN I/O BUFFER ?
  1020 1201         1567 GOC    SRBF10 (1157) YES
  1021 1202 SRBF30   414 ?S8=1                 ALLOW RETURN TO P+2 ?
  1022 1203         1640 RTN NC                NO
  1023 1204          243 GOTO   RTNP+2 (1230)
*
*************************************************************************
* ALMSST - SINGLE STEP TO NEXT ALARM                          10-9-80 RSW
*
* INPUT:  M.X= ADDRESS OF CURRENT ALARM
* !!!NOTE - ALARM ADDRESS MUST REALLY BE AN ALARM ADDRESS, NOT THE
*           ADDRESS OF THE TRAILER REGISTER OR SOME OTHE REGISTER.
*
* ASSUME: HEXMODE, PERIPHERALS DISABLED
* OUT:    (P+1):   C.X= M.X= NEXT ALARM ADDRESS, A.X= ORIGINAL ADDRESS
*         (P+2):   THERE IS NO NEXT ALARM (M.X  UNCHANGED)
* USES:   A.X, C, M.X, DADD, +1 SUB LEVEL (NO ST, NO PT, NO PFAD)
*
*****************************
* NEWM.X -- PUTS A.X INTO M.X
*
* IN:     A.X= ALARM ADDRESS TO BE PLACED IN M.X
* ASSUME: NOTHING
* OUT:    C.X= M.X= INPUT A.X
* USES:   A.X, C, M.X ONLY
*
*
  1046                 ENTRY  ALMSST
  1047 1205 ALMSST   630 C=M
  1048 1206            1 GOSUB  SKPALC        SKIP OVER CURRENT ALARM
  1048 1207            0                      *TIMER ROM:  TM2, @1122
  1049 1210         1160 DADD=C
  1050 1211           70 C=DATA                LOAD NEXT ALARM
  1051 1212         1076 C=C+1  S              END OF TIMER BUFFER ?
  1052 1213          157 GOC    RTNP+2 (1230) YES, RETURN TO P+2
  1053
  1054                 ENTRY  NEWM.X
  1055 1214 NEWM.X   630 C=M                  C.X= M.X= NEXT ALARM ADDR
  1056 1215          246 AC EX  X
  1057 1216          530 M=C
```

```
  1058 1217        1740 RTN
*
*********************************************************************
* ACKALM - ACKNOWLEDGE AN ALARM                              2-9-81 RSW
*  IF THE ALARM HAS A RESET INTERVAL, RESET IT TO NEXT (HOPEFULLY)
*  FUTURE OCCURRENCE.  IF THE ALARM HAS NO RESET INTERVAL, PURGE IT.
*
* INPUT:  M.X= ALARM ADDRESS
* ASSUME: HEXMODE, PERIPHERALS DISABLED
*         RESET INTERVAL STORED IN THIS FORMAT:  0 0SSSSSSSST 000
*           WHERE "T"= TENTHS OF SECONDS
* OUT:    P SELECTED, Q= 13
*         RTN TO (P+1):  [NO ALARMS LEFT, STACK PURGED]
*                        [SOME ALARMS LEFT, NO HIGHER ADDRESSED ALARMS]
*         RTN TO (P+2):  AT LEAST 1 HIGHER ADDRESSED ALARM LEFT
*           M.X= NEXT (TOWARD FUTURE) ALARM ADDRESS
* USES:   A,B,C,N, MAY UPDATE M.X, P,Q, S3,S8, +3 SUB LEVELS, DADD, PFAD
*         ARITH MODE, TIMER PT   (NO TIMER ST)
*
*
  1078                   ENTRY  ACKALM
  1079 1220 ACKALM     1 GOSUB  RSTALM        YES, RESET THIS ALARM
  1079 1221            0                      *TIMER ROM:  TM0, @1042
  1080 1222           63 GOTO   RTNP+2 (1230) (P+1) ALM HAS RESET INTERVL
  1081 1223 ACKL10     1 GOSUB  PUGALM        (P+2) PURGE THIS ALARM
  1081 1224            0                      *TIMER ROM:  TM0, @0764
  1082                                              (NO RESET INTERVAL)
  1083 1225        1740 RTN                    (P+1) NO ALMS, STACK PURGED
  1084 1226         576 A=A+1  S               ANY HIGHER ADDRESSED ALMS ?
  1085 1227        1640 RTN NC                 NO, DONE
  1086
  1087
  1088                   ENTRY  RTNP+2
  1089 1230 RTNP+2 1140 SETHEX                 NEED HEXMODE FOR "C=STK"
  1090 1231         660 C=STK                  RETURN TO (P+2)
  1091 1232        1072 C=C+1  M
  1092 1233         740 GOTOC
*********************************************************************
*                                                            2-17-81 RSW
* INTVAL - RESET INTERAL VALUE
*          IF S6=0: DISPLAYS INTERVAL VALUE OR 00:00:00 FOR NO INTERVAL
*                   (ALWAYS RETURNS TO P+2)
*          IF S6=1: NO INTERVAL -- NO DISPLAY, RETURNS TO P+1
*                   INTERVAL -- DISPLAY INTERVAL AND RETURN TO P+2
*
* IN:      S6 INITIALIZED PROPERLY
*          PERIPHERALS DISABLED
* ASSUME:  M.X= ADDRESS OF CURRENT ALARM, HEXMODE
* OUT:     (P+1) RETURNS HERE IF S6= 1 AND THE ALARM HAS NO RESET
*                INTERVAL.  PERIPHERALS DISABLED, S6= 1
*          (P+2) RETURNS HERE IF (S6= 0) OR (S6= 1 AND ALARM HAS A
*                                            RESET INTERVAL)
*                DISPLAY ENABLED, RAM DISABLED
* USES:    A,B,C, P,Q, S5,S8,, +2 SUB LEVELS, DADD, PFAD, ARITH MODE
*          (NO TIMER CHIP ACCESS)
*
  1112                   ENTRY  INTVAL
  1113 1234 INTVAL     1 GOSUB  GETM.X        C= CURRENT ALARM
  1113 1235            0                      *TIMER ROM:  TM1, @0532
  1114 1236        1366 ? C#0  XS             HAS AUTO RESET INTERVAL ?
```

```
  1115 1237          233 GONC   INTV20 (1262) NO
  1116 1240          630 C=M
  1117 1241         1046 C=C+1  X              POINT TO RESET INTERVAL
  1118 1242         1160 DADD=C
  1119 1243           70 C=DATA                C= SSSSSSSSSSC000
  1120 1244         1074 RCR    2              C= 00SSSSSSSSSSC0
  1121 1245            1 GOSUB  SDHMSC         A= DDDDDDHHMMSSCC
  1121 1246            0                       *TIMER ROM:  TM2, @1301
  1122 1247            1 GOSUB  X20Q8          A= (DAYS) X 20
  1122 1250            0                       *TIMER ROM:  TM3, @0043
  1123 1251         1716 C SR                  C= (DAYS) X 4
  1124 1252         1016 C=A+C                 C= (DAYS) X 24
  1125 1253          674 RCR    11             TRUNCATE TO 4 HOUR DIGITS
  1126 1254 INTV10     1 GOSUB  DSPINT         DISPLAY RESET INTERVAL
  1126 1255            0                       *TIMER ROM:  TM1, @1625
  1127 1256            1 GOSUB  ENLCD          ENABLE DISPLAY, DISABLE RAM
  1127 1257            0                       *MAINFRAME:  CN1, @1766
  1128                                         IN & ASSUME: NOTHING
  1129                                         OUT: DISPLAY ENABLED
  1130                                              RAM DISABLED
  1131                                         USES: C.X, DADD, PFAD ONLY
  1132 1260         1670 RABCR                 ROTATE 1 PLACE TO THE RIGHT
  1133 1261         1473 GOTO   RTNP+2 (1230)
  1134 1262 INTV20   514 ?S6=1
  1135 1263         1540 RTN C
  1136 1264          116 C=0                   NO RESET INTERVAL
  1137 1265         1673 GOTO   INTV10 (1254)
*******************************************************************
*                                                        1-19-81 RSW
* IGDHMS - INITIALIZE & GET DAYS, HOURS, MINUTES, SECONDS
*
* IN:      NOTHING
* ASSUME:  VALID CLOCK TIME (NO MORE THAN 1 YR PAST 12/2199, NO GARBAGE)
* OUT:     TIMER CHIP ENABLED, RAM DISABLED, TIMER PT=A
*          A= C= DDDDDDHHMMSSCC,   HEXMODE
* USES:    A, B.M, C, ACTIVE PT, S0-S8, +2 SUB LVLS, DADD, PFAD,
*          ARITH MODE, TIMER PT         (NO TIMER ST)
*
*
* GDHMS -- SAME AS IGDHMS EXCEPT:
*            - ASSUMES THE TIMER CHIP DOESN'T NEED TO BE INITIALIZED
*            - DOESN'T USE S0-S7
*            - ONLY USES +1 SUB LEVEL
*
  1155                 ENTRY  IGDHMS
  1156                 ENTRY  GDHMS
  1157 1266 IGDHMS     1 GOSUB  INITMR         INIT TIMER IF NECESSARY
  1157 1267            0                       *TIMER ROM:  TM0, @1524
  1158 1270 GDHMS      1 GOSUB  ENTMR          ENABLE TIMER, DIS RAM, PT=A
  1158 1271            0                       *TIMER ROM:  TM0, @0342
  1159 1272           70 RDTIME                C= CLOCK TIME
  1160 1273           63 GOTO   SDHMSC (1301) A= C= DDDDDDHHMMSSCC
*
*******************************************************************
*                                                        1-15-81 RSW
* A-DHMS - CONVERT ALARM TIME (FROM ALARM STACK) TO DAYS, HOURS, MINUTES
*          AND SECONDS
* INPUT:   M.X= ADDRESS OF CURRENT ALARM (! MUST BE VALID ALARM ADDRESS,
*                                         NOT TRAILER REGISTER !!)
* ASSUME:  PERIPHERALS DISABLED
```

```
* OUTPUT:  A= C= DDDDDDHHMMSSCC, HEXMODE
*          ALARM REGISTER ENABLED, S8= 0
*
* USES:    A,B[13:3],C, ACTIVE PT, S8, ARITH MODE, DADD
*          (+0 SUB LEVELS, NO PFAD, NO TIMER CHIP ACCESS)
*
*
*
  1177                  ENTRY  A-DHMS
  1178 1274 A-DHMS  630 C=M
  1179 1275        1160 DADD=C
  1180 1276          70 C=DATA              C= SSSSSSSSSSCXXX
  1181 1277         106 C=0    X
  1182 1300        1074 RCR    2             C= 00SSSSSSSSSSC0
*
*********************************************************************
*                                                          1-15-81 RSW
* SDHMSK - CONVERT SECONDS INTO DAYS, HOURS, MINUTES, SECONDS, 100TH'S
*
*  !! IF S8=1 & S9=0, JUMPS DIRECTLY TO TMRKEY ON KEY DOWN!!!
*
*
* INPUT:   A= 00SSSSSSSSSSCC
*          C.X AND THE ACTIVE PT MUST BE RELATED AS FOLLOWS:
*          C.X  ACTIVE PT  "A"  CONTAINS NUMBER OF SECONDS EQUIVALENT TO
*          ***  *********  ********************************************
*           0       6                    0-9 DAYS   (1 DIGIT)
*           1       7                  10-99 DAYS   (2 DIGITS)
*           2       8                100-999 DAYS   (3 DIGITS)
*           3       9              1000-9999 DAYS   (4 DIGITS)
*           4      10            10000-99999 DAYS   (5 DIGITS)
*           5      11          100000-999999 DAYS   (6 DIGITS)
*           IF S8= 1, THEN KEYBOARD MUST BE CLEARED RIGHT BEFORE CALLING
*              SDHMSK IF YOU ONLY WANT TO SEE KEYS PRESSED DURING SDHMSK
* ASSUME:  S8= 0  TO IGNORE KEYBOARD  (S9= DON'T CARE)
*          S8= 1  TO CHECK KEYBOARD
*              AND S9= 1 (0) RETURN ON KEY UP (DOWN)
* OUT:     IF (S8= 0) OR (S8= 1 AND NO KEY TRANSITIONS DETECTED) THEN:
*              A= C= DDDDDDHHMMSSCC, HEXMODE, PT=3
*          IF S8= 1 AND A KEY TRANSITION HAS BEEN DETECTED THEN:
*              DECMODE
*  !! NOTE: SINCE SDHMSK DOESN'T CLEAR THE KEYBOARD, THE KEY COULD HAVE
*          BEEN PUSHED BEFORE EVER CALLING SDHMSK
* USES:    A,B.M,C, ACTIVE PT, ARITH MODE
*          (+0 SUB LEVELS, NO DADD, NO PFAD, NO TIMER CHIP ACCESS)
*
*
* SDHMSA -- SAME AS SDHMSK EXCEPT:
*           SETS C.X= 5 AND PT= 11        (6 DIGITS  OF DAY)
*           SETS S8= 0 TO IGNORE KEYBOARD
*                !!! SO USES S8 !!!
*
* SDHMSC -- SAME AS SDHMSA EXCEPT THE SECONDS ARE INPUT IN C
*     NOTE: TAKES ABOUT 341 WORD TIMES WORST CASE FOR DATES IN THE RANGE
*           1/1/1900 - 12/31/2199  (99999 DAYS, 19 HOURS, 59 MINUTES)
*
*
  1226                  ENTRY  SDHMSK
  1227                  ENTRY  SDHMSC
  1228 1301 SDHMSC  416 A=C
```

```
1229 1302          404 S8=    0             IGNORE KEYBOARD
1230 1303          460 LDI                  LOAD LOW 12 BITS OF C WITH
1231 1304            5 CON    5             REPEAT FOR 6 DIGITS OF DAY
1232 1305          634 PT=    11
1233 1306 SDHMSK   132 C=0    M
1234 1307         1020 LC     8
1235 1310          620 LC     6
1236 1311          420 LC     4             86400 SECONDS= 1 DAY
1237 1312         1734 INC PT
1238 1313         1734 INC PT
1239 1314         1734 INC PT
1240 1315         1734 INC PT
1241 1316         1240 SETDEC
1242 1317          372 BC EX  M
1243 1320          132 C=0    M
1244 1321           23 GOTO   DHMS25 (1323)
1245 1322 DHMS20  1042 C=C+1  PT
1246 1323 DHMS25   632 A=A-B  M
1247 1324         1763 GONC   DHMS20 (1322)
1248 1325          472 A=A+B  M
1249 1326          414 ?S8=1                CHECK KEYBOARD ?
1250 1327          123 GONC   DHMS28 (1341) NO
1251 1330         1114 ?S9=1                RETURN ON KEY UP ?
1252 1331           53 GONC   DHMS27 (1336) NO, ON KEY DOWN
1253 1332         1710 RST KB
1254 1333         1714 CHK KB
1255 1334         1640 RTN NC               NO, ABORT
1256 1335           43 GOTO   DHMS28 (1341)
1257 1336 DHMS27  1714 CHK KB
1258 1337            1 GOLC   TMRKEY        YES, ABORT
1258 1340            3                      *TIMER ROM:  TM1, @0773
1259 1341 DHMS28  1146 C=C-1  X
1260 1342           47 GOC    DHMS30 (1346)
1261 1343         1724 DEC PT
1262 1344         1672 B SR   M
1263 1345         1563 GOTO   DHMS25 (1323)
1264 1346 DHMS30  1224 ? PT=  7             JUST FINISHED DAYS?
1265 1347          133 GONC   DHMS40 (1362) NO, FINISHED HRS OR MINUTES
1266                                        YES, SET UP FOR HOURS
1267 1350          372 BC EX  M             SAVE C IN B
1268 1351          132 C=0    M
1269 1352          534 PT=    6
1270 1353          320 LC     3
1271 1354          620 LC     6             3600 SECONDS IN 1 HOUR
1272 1355          534 PT=    6             C.M= 0000003600= 10 HOURS
1273 1356 DHMS35   372 BC EX  M             RESTORE C, B= NEW CONSTANT
1274 1357          106 C=0    X
1275 1360         1046 C=C+1  X             REPEAT FOR 2 DIGITS
1276                LEGAL                   (CLEAR THE CARRY FLAG)
1277 1361         1423 GOTO   DHMS25 (1323)
1278 1362 DHMS40   224 ? PT=  5             JUST FINISHED HOURS ?
1279 1363           43 GONC   DHMS50 (1367) NO, JUST FINISHED MINUTES
1280                                        YES, SET UP FOR MINUTES
1281 1364          372 BC EX  M             C.M= 0000000360, SAVE C->B
1282 1365           20 LC     0             C.M= 0000000060= 10 MINUTES
1283 1366         1703 GOTO   DHMS35 (1356)
1284 1367 DHMS50  1374 RCR    13            C= DDDDDDHHMM0000
1285 1370          252 AC EX  WPT           C= DDDDDDHHMMSSCC
1286 1371          416 A=C
1287 1372         1140 SETHEX
```

```
  1288 1373        1740 RTN                  13 EXIT AFTER LAST KEY CHK
*****************************************************************
* SETAF =  SET ACCURACY FACTOR                          2-4-81 RSW
*****************************************************************
  1292 1374         206 CON    @206          F
  1293 1375           1 CON    @01           A
  1294 1376          24 CON    @24           T
  1295 1377           5 CON    @05           E
  1296 1400          23 CON    @23           S
  1297                  ENTRY  SETAF
  1298                  ENTRY  SETAF0
  1299 1401 SETAF     1 GOSUB  INITMR        INIT SO NEW A.F WON'T CLEAR
  1299 1402           0                      *TIMER ROM:  TM0, @1524
  1300 1403        1750 PT=A                 SELECT MAIN CLOCK
  1301 1404          70 RDTIME               C= CURRENT TIME
  1302 1405         450 WRSCR                UPDATE "LAST TIME SET"
  1303 1406           1 GOSUB  CHECKX        ERROR IF X= ALPHA DATA
  1303 1407           0                      *TIMER ROM:  TM0, @0240
*
*.................................................................
* SETAF0 -- SET (STORE) ACCURACY FACTOR                 2-4-81 RSW
*
* IN:       C= FLOATING POINT NORMALIZED ACCURACY FACTOR
* ASSUME:   NOTHING
* OUT:      NEW ACCURACY FACTOR STORED IN AF REGISTER IN TIMER CHIP
*           TIMER CHIP ENABLED, RAM DISABLED
* USES:     A, C, ACTIVE PT, +1 SUB LEVEL, ARITH MODE,
*           DADD, PFAD, TIMER PT      (NO TIMER ST)
*
*
  1316 1410 SETAF0 1356 ? C#0               AF= 0 EXACTLY ?
  1317 1411         203 GONC   SETA40 (1431) YES, DON'T ROUND  TO 0.1
  1318 1412           1 GOSUB  UNNOR1        UNNORMALIZE AF
  1318 1413           0                      *TIMER ROM:  TM3, @0022
  1319 1414          33 GOTO   SETA10 (1417) (P+1) OK, A=#DDD........000
  1320 1415         116 C=0                  (P+2) AF => 100, SET TO
  1321                                             MINIMUM CORRECTION
  1322 1416         133 GOTO   SETA40 (1431) DON'T SET AF OF 0 TO 0.1
  1323 1417 SETA10 1334 PT=    13
  1324 1420         520 LC     5
  1325 1421         256 AC EX               C= UNNORMALIZED AF, A.S= 5
  1326 1422         374 RCR    10           C.X= AF
  1327 1423        1036 C=A+C  S             NEED TO ROUND UP AF?
  1328 1424          33 GONC   SETA30 (1427) NO
  1329 1425 SETA20 1046 C=C+1  X             ROUND UP
  1330 1426          37 GOC    SETA40 (1431) +-99.95 => +-0 (NO CORRECT)
  1331 1427 SETA30 1346 ? C#0  X             AF = 0 ?
  1332 1430        1753 GONC   SETA20 (1425) YES, ROUND TO 0.1
  1333 1431 SETA40  674 RCR    11           C= .......SDDD...
  1334 1432           1 GOSUB  ENTMR         ENABLE TIMER, DISABLE RAM
  1334 1433           0                      *TIMER ROM:  TM0, @0342
  1335 1434        1074 RCR    2            C= .........SSDD.
  1336 1435        1650 PT=B                SELECT AF
  1337 1436         350 WRSTS               STORE ACCURACY FACTOR
  1338 1437        1740 RTN
*****************************************************************
* 115860 - SET C= 11586000000000                       2-19-81 RSW
*
* IN & ASSUME: NOTHING
* OUT:         C= 11586000000000
```

```
* USES:        C, ACTIVE PT ONLY
*
  1346                   ENTRY  115860
  1347 1440 115860  116 C=0
  1348 1441        1334 PT=    13
  1349 1442         120 LC     1
  1350 1443         120 LC     1
  1351 1444         520 LC     5
  1352 1445        1020 LC     8
  1353 1446         620 LC     6
  1354 1447        1740 RTN
*************************************************************************
* CLRALM - CLEAR ALARM  (HARDWARE ALARM BITS)              1-15-81 RSW
*
* IN & ASSUME: TIMER CHIP ENABLED, RAM DISABLED
* OUT:     TIMER PT=A
*          HARDWARE ALARM BITS DTZA, ALMB, AND DTZIT CLEARED
*            DTZA=  DECREMENT THROUGH ZERO ON CLOCK A (SHOULDN'T HAPPEN)
*            DTZIT= DECREMENT THROUGH ZERO ON INTERVAL TIMER
*            ALMB=  ALARM ON CLOCK B (SHOULDN'T HAPPEN)
*   !! NOTE: DTZIT COULD BE SET AGAIN ON EXIT IF INTERVAL TIMER RUNNING
*
* USES:    C.X, TIMER PT ONLY
*
*
* CLRAL0 - SAME AS CLRALM EXCEPT ASSUMES TIMER PT=A
*          USES ONLY C.X
*
* CLRALW - WRITES THE CONTENTS OOF "C" TO TIMER SCRATCH REG B, THEN
*          CLEARS DTZA, ALMB, DTZIT.
*          IN:     C[1:0]= UPDATED TIMER SOFTWARE STATUS, TIMER PT=B
*          ASSUME: TIMER CHIPE ENABLED, RAM DISABLED
*          OUT:    TIMER SCRATCH REG B UPDATED, + CLRALM OUTPUT
*          USES:   C.X, TIMER PT ONLY
*
*
* CLRALS - STOPS THE INTERVAL TIMER, CLEARS CLOCK DISPLAY BITS AND
*          UPDATES THE SOFTWARE STATUS BIT PATTERN IN TIMER SCRATCH
*          REGISTER B
*          IN:     S0-S7= TIMER SOFTWARE STATUS BITS, TIMER PT=B
*          ASSUME: TIMER CHIP ENABLED, RAM DISABLED
*          OUT:    SAME AS CLRALW BUT ALSO:    S0-S7= INPUT C[1:0]
*                                              INTERVAL TIMER STOPPED
*          USES:   C.X, S0-S7, TIMER PT ONLY
*
  1389                   ENTRY  CLRALD
  1390                   ENTRY  CLRALS
  1391                   ENTRY  CLRALW
  1392                   ENTRY  CLRALM
  1393                   ENTRY  CLRAL0
  1394 1450 CLRALD 1604 S0=    0             CLEAR DSWKNO BIT
  1395 1451 CLRALS  750 STPINT               STOP INTERVAL TIMER
  1396 1452           4 S3=    0
  1397 1453         104 S4=    0
  1398 1454        1730 CST EX
  1399 1455 CLRALW  450 WRSCR
  1400 1456 CLRALM 1750 PT=A
  1401 1457 CLRAL0  460 LDI                   LOAD LOW 12 BITS OF C WITH
  1402 1460          51 CON    @51            NO CLEAR ALM A, DTZB, PUS
  1403 1461         350 WRSTS                 CLEAR ALL OTHER ALARMS
```

```
  1404 1462        1740 RTN
  1405
*************************************************************************
* R9=T  -  REGISTER 9 = TIME                                  12/4/80 RSW
*
* IN:      C= FLOATING POINT NUMBER
* ASSUME:  NOTHING
* OUT:     C= REG9= MAIN CLOCK TIME, B= COPY OF INPUT CONTENTS OF "C"
*          A= 0
*          TIMER PT=A, CHIP 0 ENABLED, PERIPHERALS DISABLED, DEC MODE
* USES:    A,B,C, S0-S7, TIMER PT, DADD, PFAD, ARITH MODE, +2 SUB LVLS
*          (NO 41C PT)
*
  1417                    ENTRY  R9=T
  1418 1463 R9=T    356 BC EX                B=H.MS TIME (FOR T+X)
  1419 1464           1 GOSUB  INITMR        INIT TIMER IF NECESSARY
  1419 1465           0                      *TIMER ROM:  TM0, @1524
  1420 1466        1750 PT=A
  1421 1467          70 RDTIME              C= MAIN CLOCK TIME
  1422 1470         416 A=C
  1423 1471         116 C=0
  1424 1472        1160 DADD=C
  1425 1473         256 AC EX
  1426 1474        1150 REGN=C 9            REG 9= MAIN CLOCK TIME
  1427 1475        1240 SETDEC
  1428 1476        1740 RTN
*
*************************************************************************
* SETSW - USE X REGISTER TO SET THE TIME IN THE STOPWATCH   1-30-81 RSW
*
*************************************************************************
*
  1435 1477         227 CON    @227          W
  1436 1500          23 CON    @23           S
  1437 1501          24 CON    @24           T
  1438 1502           5 CON    @05           E
  1439 1503          23 CON    @23           S
  1440                  ENTRY  SETSW
  1441 1504 SETSW     1 GOSUB  UNNORX         CHECK ALPHA DATA, UNNORM X
  1441 1505           0                      *TIMER ROM:  TM3, @0020
  1442 1506          23 GOTO   XTMR05 (1510) (P+1) A= #HHMMSSCC.....
  1443 1507          63 GOTO   SETSDE (1515) (P+2) ERROR
  1444
  1445 1510 XTMR05  236 B=A    S              SAVE THE SIGN
  1446 1511         404 S8=    0              IGNORE KEYBOARD
  1447 1512           1 GOSUB  HMSS20         HOURS, MINS, SECS TO SECS
  1447 1513           0                      *TIMER ROM:  TM1, @0343
  1448 1514          33 GOTO   XTMR10 (1517) (P+1) C= 00SSSSSSSSSSCC
  1449 1515 SETSDE    1 GOLONG ERRDE          (P+2) DATA ERROR
  1449 1516           2                      *MAINFRAME: CN10, @0055
  1450 1517 XTMR10    1 GOSUB  INITMM         INITIALIZE, TIMER PT=B
  1450 1520           0                      *TIMER ROM:  TM0, @1523
  1451 1521         630 C=M
  1452
  1453 1522        1336 ? B#0  S              NEGATIVE TIME ?
  1454 1523          33 GONC   XTMR30 (1526) NO, POSITIVE
  1455 1524        1240 SETDEC
  1456 1525        1216 C=-C
  1457 1526 XTMR30   50 WRTIME               STORE TIME IN STOPWATCH
  1458 1527        1740 RTN
```

```
*********************************************************************
  1460 1530          264 CON    @264           4               1-29-81 RSW
  1461 1531           62 CON    @62            2
  1462 1532            5 CON    @05            E
  1463 1533           15 CON    @15            M
  1464 1534           11 CON    @11            I
  1465 1535           24 CON    @24            T
  1466 1536            1 CON    @01            A
  1467                   ENTRY  ATIM24
  1468 1537 ATIM24 1604 S0=     0              DOING A TIME
  1469 1540          210 S5=     1
  1470 1541          173 GOTO   ATD007 (1560)
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
  1472 1542          205 CON    @205           E               1-29-81 RSW
  1473 1543           15 CON    @15            M
  1474 1544           11 CON    @11            I
  1475 1545           24 CON    @24            T
  1476 1546            1 CON    @01            A
  1477                   ENTRY  ATIME
  1478 1547 ATIME  1604 S0=     0              DOING A TIME
  1479 1550           73 GOTO   TIMEIN (1557)
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
  1481 1551          205 CON    @205           E               1-29-81 RSW
  1482 1552           24 CON    @24            T
  1483 1553            1 CON    @01            A
  1484 1554            4 CON    @04            D
  1485 1555            1 CON    @01            A
  1486                   ENTRY  ADATE
  1487 1556 ADATE  1610 S0=     1              DOING A DATE
  1488 1557 TIMEIN  204 S5=     0              ASSUME 12-HOUR FORMAT
  1489 1560 ATD007  104 S4=     0              NOT FORMATTING FOR DISPLAY
  1490 1561            1 GOSUB  UNNORX         UNNORMALIZE X REGISTER
  1490 1562            0                       *TIMER ROM:  TM3, @0020
  1491 1563           23 GOTO   ATD010 (1565) (P+1) OK
  1492 1564         1313 GOTO   SETSDE (1515) (P+2) "DATA ERROR"
  1493 1565 ATD010 1670 C=REGN 14
  1494 1566          174 RCR    4              C[0]= NUMBER OF DISP DIGITS
*
*
* DATEIN = DATE IN     (DATE DISPLAY ENTRY POINT)           1-26-81 RSW
*
*    INTERNAL POINTERS & STATUS BITS
*             S0=  1 (0)    DOING A DATE (TIME)
*                IF S0= 1, THEN:    S3=  1 (0)    DMY (MDY)
*             S4=  1 (0)    FORMATTING FOR DISPLAY (ALPHA REGISTER)
*             S5=  1 (0)    24 (12) HOUR FORMAT
*             S6=  1 (0)    PM (AM)  !! FOR TIME ONLY, NOT DATES !!
*
*             A[0]=  FORMAT COUNTER= NUMBER OF DIGITS TO OUTPUT
*             A[1]=  FIELD COUNTER
*             A[2]=  DIGITS SINCE LAST SEPARATOR COUNTER
*
* IN: TIME --     A= #HHMMSSCC.....    WHERE "#"= 0 FOR AM
*                                           = NON-ZERO FOR PM
*     LATE --     A= 0MMDDYYYY.....  OR   DDMMYYYY.....
*     FOR BOTH -- C[0]= NUMBER OF DIGITS TO THE RIGHT OF THE DECIMAL
*                    POINT IN "FIX" MODE
*                   (ODD NUMBERS WILL BE ROUNDED UP, NUMBERS >=6
*                    WILL BE SET= 6)
*                 S5= 1   FOR UNCONDITIONAL 24-HOUR FORMAT
```

```
*                S5= 0   TO USE THE "CLK12"/"CLK24" BIT
* ASSUME:      S0= 1     TO CREATE A DATE FORMAT
*                S4= 1   TO SEND OUTPUT TO DISPLAY.  THE CHARACTERS
*                        ARE SHIFTED LEFT INTO THE DISPLAY, SO THE
*                        DISPLAY MUST BE PROPERLY INITIALIZED.
*                        (!! DATEIN DOES NOT CLEAR THE DISPLAY !!!)
*                S4= 0   TO SEND OUTPUT TO ALPHA REGISTER
*              S0= 0     TO CREATE A TIME FORMAT
*                S4= 0   IS REQUIRED FOR TIME FORMAT !!!
* OUT:      HEXMODE
*           S5=  1  (0)    FOR 24 (12) HOUR FORMAT
*           IF TIME (S0=0) IS BEING FORMATTED: S6= 1 (0) FOR PM (AM)
*           IF DATE (S0=1) IS BEING FORMATTED: S3= 1 (0) FOR DMY (MDY)
*           IF S4= 1, THEN DISPLAY ENABLED, RAM DISABLED
*           IF S4= 0, THEN CHIP 0  ENABLED, PERIPHERALS DISABLED
* USES:     A,B,C,G, S3 IF S0=1, S5,S6 IF S0=0, ACTIVE PT, +1 SUB LEVEL,
*           ARITH MODE, PFAD, DADD (NO TIMER CHIP ACCESS)
*
  1536                    ENTRY  DATEIN
  1537 1567 DATEIN  1434 PT=    1             A[1]=5 FOR CENTURY DELETE
  1538 1570          520 LC     5                    (DATES)
  1539 1571         1140 SETHEX                NOW PT=0
  1540 1572          406 A=C    X
  1541 1573            1 GOSUB  TMRSTS         READ TIMER HARDWARE STATUS
  1541 1574            0                       *TIMER ROM:  TM3, @0244
  1542 1575          514 ?S6=1                 24-HOUR FORMAT ?
  1543 1576           43 GONC   ATD015 (1602)  NO, 12-HOUR FORMAT
  1544 1577         1730 CST EX
  1545 1600          210 S5=    1             REMEMBER 24-HOUR FORMAT
  1546 1601           23 GOTO   ATD017 (1603)
  1547 1602 ATD015  1730 CST EX
  1548 1603 ATD017     1 GOSUB  ENCP00         ENA CHIP 0, DISABLE PERIPHS
  1548 1604            0                       *MAINFRAME:  CN2, @0522
  1549                                         IN:     NOTHING
  1550                                         ASSUME: NOTHING
  1551                                         OUT:    C.X= 0
  1552                                         USES:   C.X, DADD, PFAD
  1553 1605          460 LDI                   LOAD LOW 12 BITS OF C WITH
  1554 1606          446 CON    @446           DIGIT CTR= 1, FIELD CTR= 2
  1555                                           FORMAT CTR= 6
  1556 1607         1402 ? A<C  PT             DISPLAY FORMAT < 6 ?
  1557 1610           23 GONC   ATD020 (1612)  NO, SET FORMAT = 6
  1558 1611          242 AC EX  PT
  1559 1612 ATD020  1042 C=C+1  PT             ROUND FORMAT UP
  1560 1613         1730 CST EX
  1561 1614         1610 S0=    1             ADD 2 DIGITS LEFT OF D.P.
  1562 1615         1730 CST EX
  1563 1616         1614 ?S0=1                DOING A DATE
  1564 1617          173 GONC   ATD032 (1636) NO, A TIME
  1565 1620         1606 A SR   X              A[0]= 5
  1566 1621         1542 ? A#C  PT             FORMAT + 1 = 5 ?
  1567 1622           47 GOC    ATD030 (1626) NO, DON'T DELETE CENTURY
  1568 1623          434 PT=    8
  1569 1624         1752 A SL   WPT
  1570 1625         1752 A SL   WPT
  1571 1626 ATD030   406 A=C    X              A.X= FORMAT, FIELD, & DIGIT
  1572                                            = 12X           |COUNTERS
  1573 1627            4 S3=    0              ASSUME MDY
  1574 1630            1 GOSUB  GFLG31         GET FLAG 31
  1574 1631            0                       *TIMER ROM:  TM0, @1272
```

```
  1575 1632          742 C=C+C  PT             DMY ?
  1576 1633          263 GONC   ATD061 (1661) NO, MDY
  1577 1634           10 S3=    1              REMEMBER DMY
  1578 1635          243 GOTO   ATD061 (1661)
  1579 1636 ATD032   406 A=C    X              A.X= FMT,FIELD,DGT CTRS=12X
  1580 1637          214 ?S5=1                 24-HOUR FORMAT ?
  1581 1640          103 GONC   ATD050 (1650) NO, 12-HOUR
  1582 1641            1 GOSUB  TO24H          CONVERT HR TO 24-HR FORMAT
  1582 1642            0                       *TIMER ROM:  TM1, @0202
  1583 1643         1140 SETHEX
  1584 1644          256 AC EX                 C= MMSSCC.....0HH
  1585 1645           74 RCR    3
  1586 1646          416 A=C                   A= 0HHMMSSCC..12N WHERE
  1587                                            N= #DIGITS TO OUTPUT - 1
  1588 1647          123 GOTO   ATD061 (1661)
  1589 1650 ATD050     1 GOSUB  TO12H          NO, CONVERT TM TO 12-HR FMT
  1589 1651            0                       *TIMER ROM:  TM1, @0262
  1590 1652         1534 PT=    12
  1591 1653         1502 ? A#0  PT             2-DIGIT HOUR/MONTH ?
  1592 1654           57 GOC    ATD061 (1661) YES
  1593 1655         1334 PT=    13
  1594 1656          220 LC     2
  1595 1657           63 GOTO   ATD070 (1665) LEADING BLANK TO ALPHA REG
  1596
  1597 1660 ATD060 1772 A SL    M              A[12]= NEXT DIGIT
  1598 1661 ATD061  114 ?S4=1                  FORMATTING FOR DISPLAY ?
  1599 1662          107 GOC    ATD080 (1672) YES
  1600 1663         1334 PT=    13
  1601 1664          320 LC     3
  1602 1665 ATD070   242 AC EX  PT
  1603 1666            1 GOSUB  ATD120         APPEND DIGIT TO ALPHA REG
  1603 1667            0                       *TIMER ROM:  TM2, @1727
  1604 1670          156 AB EX
  1605 1671          113 GOTO   ATD090 (1702)
  1606
  1607 1672 ATD080     1 GOSUB  ENLCD          ENABLE DISPLAY, DISABLE RAM
  1607 1673            0                       *MAINFRAME:  CN1, @1766
  1608                                         IN & ASSUME: NOTHING
  1609                                         OUT: DISPLAY ENABLED
  1610                                              RAM DISABLED
  1611                                         USES: C.X, DADD, PFAD ONLY
  1612 1674         1634 PT=    0
  1613 1675           20 LC     0
  1614 1676          320 LC     3
  1615 1677          242 AC EX  PT             C[12]= A[12]
  1616 1700         1574 RCR    12             C.X= LCD FORMAT DIGIT
  1617 1701         1750 SLSABC
  1618 1702 ATD090 1634 PT=     0              CHECK FORMAT COUNTER
  1619 1703          642 A=A-1  PT             END OF DIGITS ?
  1620 1704          273 GONC   ATD150 (1733) NO
  1621 1705         1614 ?S0=1                 DOING A DATE ?
  1622 1706         1540 RTN C                 YES, DONE
  1623 1707          214 ?S5=1                 24-HOUR FORMAT ?
  1624 1710         1540 RTN C                 YES, DONE
*   MUST BE FORMATTING A TIME, SO NOT FORMATTING FOR DISPLAY,
*   SO CHIP 0 IS ENABLED.
  1627 1711          460 LDI                   LOAD LOW 12 BITS OF C WITH
  1628 1712           40 CON2   2      0       BLANK
  1629 1713            1 GOSUB  ATD125         APPEND BLANK TO ALPHA REG
  1629 1714            0                       *TIMER ROM:  TM2, @1730
```

```
1630 1715          460 LDI                    LOAD LOW 12 BITS OF C WITH
1631 1716          101 CON2   4       1       ASSUME AM (ASCII "A")
1632 1717          514 ?S6=1                  PM ?
1633 1720           33 GONC   ATD110 (1723)  NO, AM
1634 1721          460 LDI                    LOAD LOW 12 BITS OF C WITH
1635 1722          120 CON2   5       0       ASCII "P"
1636 1723 ATD110     1 GOSUB  ATD125          APPEND "A" OR "P" TO ALPHA
1636 1724            0                        *TIMER ROM:  TM2, @1730
1637 1725          460 LDI                    LOAD LOW 12 BITS OF C WITH
1638 1726          115 CON2   4       13      ASCII "M"
1639
1640                   ENTRY  ATD120
1641                   ENTRY  ATD125
1642 1727 ATD120   156 AB EX                  SAVES CODE (B= TIME/DATE)
1643 1730 ATD125   130 G=C                              (AND COUNTERS)
1644 1731            1 GOLONG APNDNW          APPEND CHAR TO ALPHA REG
1644 1732            2                        *MAINFRAME: CN11, @0424
1645                                          IN: G= ASCII CHARACTER
1646                                          ASSUME: CHIP 0 ENABLED
1647                                                  PERIPH DISABLED
1648                                          OUT:    PT= 0
1649                                          USES:   A, C, ACTIVE PT
1650                                          (NO ST, +0 SUB LEVELS,
1651                                           NO ARITH MODE)
1652 1733 ATD150   666 A=A-1  XS              2 DGT SINCE LAST SEPARATOR?
1653 1734 ATD152  1243 GONC   ATD060 (1660)  NO, SEND 2ND DIGIT
1654 1735         1434 PT=    1
1655 1736         1614 ?S0=1                 DOING A DATE ?
1656 1737          177 GOC    ATD190 (1756)  YES
1657 1740          642 A=A-1  PT              JUST SEND SECONDS ?
1658 1741          123 GONC   ATD180 (1753)  NO
1659 1742          220 LC     2
1660 1743         1620 LC     14             2D = ASCII DECIMAL POINT
1661 1744 ATD160  1634 PT=    0
1662 1745            1 GOSUB  ATD120          APPEND IT TO ALPHA REG
1662 1746            0                        *TIMER ROM:  TM2, @1727
1663 1747          156 AB EX                  A= TIME DATE & COUNTERS
1664 1750 ATD170    26 A=0    XS
1665 1751          566 A=A+1  XS              DIGIT COUNTER= 1
1666                   LEGAL                  (CLEAR THE CARRY FLAG)
1667 1752         1623 GOTO   ATD152 (1734)
1668
1669 1753 ATD180   320 LC     3
1670 1754         1220 LC     10             3A= ASCII COLON
1671 1755         1673 GOTO   ATD160 (1744)
1672 1756 ATD190   642 A=A-1  PT              JUST SEND FIRST 2 DIGITS
1673                                            OF YEAR ?
1674 1757         1717 GOC    ATD170 (1750)  YES, DON'T SEND SEPARATOR
1675 1760          460 LDI                    LOAD LOW 12 BITS OF C WITH
1676 1761           57 CON2   2       15      2F= ASCII OR LCD "/"
1677 1762           14 ?S3=1                  DMY ?
1678 1763          113 GONC   ATD200 (1774)  NO, MDY -- USE "/"
1679 1764         1146 C=C-1  X               C.X= 02E= ASCII "."
1680 1765          114 ?S4=1                  FORMATTING FOR DISPLAY ?
1681 1766         1563 GONC   ATD160 (1744)  NO
1682 1767         1670 FRSABC                FETCH RIGHTMOST CHARACTER
1683 1770         1730 CST EX
1684 1771          510 S6=    1               ADD "."
1685 1772         1730 CST EX
1686 1773           33 GOTO   ATD210 (1776)
```

```
1687 1774 ATD200  114 ?S4=1                FORMATTING FOR DISPLAY ?
1688 1775        1473 GONC   ATD160 (1744) NO
1689 1776 ATD210 1750 SLSABC               YES, SEND IT TO LCD
1690 1777        1513 GOTO   ATD170 (1750)
1691                  END

ERRORS :      0
```

SYMBOL TABLE (BWTMB3 = TIMER ROM QUAD 3 = TM2 = ADDRESSES @54000-55777)

```
115860  1440   -
36000    454   -
36000A   461   -   462
A-DHMS  1274   -
A180J1   167   -   260
A180J2   260   -   306
ABST10   404   -   413
ABST20   414   -   406
ACKALM  1220   -
ACKL10  1223   -
ACT110    12   -    10
ACT120    21   -
ACT125    23   -
ACT135    53   -    46
ACT140    55   -    34
ACT150    72   -    65
ACT155    73   -    71
ACT165   111   -   102
ACT170   112   -   110
ACT175   115   -    60
ACT180   121   -   167
ACT185   123   -
ACT190   126   -   120
ACT240   137   -   200
ACT241   144   -   147
ACT242   151   -   141
ACT250   153   -   145  136
ACT265   172   -   173
ACT300   201   -   134
ACT310   204   -   213
ACT312   212   -   207
ACT320   231   -   205
ACT340   307   -   236
ACTBST   274   -   256
ACTEXJ   265   -   333
ACTEXT   214   -   265  261  211  150
ACTM20   351   -   377  370  363  334
ACTOFF   162   -   234
ACTR/S   277   -   331
ACTR10   221   -   266
ACTRST   216   -   254
ACTRTJ   276   -   230
ACTRTN   341   -   340  276
ACTSST   336   -   327
ADATE   1556   -
ALDATE   354   -   324
ALMBST   400   -
ALMCAT     6   -
ALMMSG   335   -   326
ALMSG    371   -   335
ALMSST  1205   -
ALTIME   343   -   330
ATD007  1560   -  1541
ATD010  1565   -  1563
ATD015  1602   -  1576
ATD017  1603   -  1601
```

```
ATD020  1612   -  1610
ATD030  1626   -  1622
ATD032  1636   -  1617
ATD050  1650   -  1640
ATD060  1660   -  1734
ATD061  1661   -  1654 1647 1635 1633
ATD070  1665   -  1657
ATD080  1672   -  1662
ATD090  1702   -  1671
ATD110  1723   -  1720
ATD120  1727   -
ATD125  1730   -
ATD150  1733   -  1704
ATD152  1734   -  1752
ATD160  1744   -  1775 1766 1755
ATD170  1750   -  1777 1757
ATD180  1753   -  1741
ATD190  1756   -  1737
ATD200  1774   -  1763
ATD210  1776   -  1773
ATIM24  1537   -
ATIME   1547   -
CHKB20  1073   -  1066
CHKB30  1074   -  1072
CHKBUF  1052   -
CLRAL0  1457   -
CLRALD  1450   -
CLRALM  1456   -
CLRALS  1451   -
CLRALW  1455   -
CURNTT   267   -   257
DATEIN  1567   -
DHMS20  1322   -  1324
DHMS25  1323   -  1361 1345 1321
DHMS27  1336   -  1331
DHMS28  1341   -  1335 1327
DHMS30  1346   -  1342
DHMS35  1356   -  1366
DHMS40  1362   -  1347
DHMS50  1367   -  1363
DST10    347   -   273
FNDEOB  1151   -
GDHMS   1270   -
IGDHMS  1266   -
INTV10  1254   -  1265
INTV20  1262   -  1237
INTVAL  1234   -
KEY-FC   434   -
KYFC10   443   -   450
KYFC30   452   -   446
LFTMEM   702   -   715
NEWL00  1106   -  1120
NEWLOC  1111   -
NEWLSK  1107   -
NEWM.X  1214   -
NOROOM   616   -   602
PLUS20  1006   -  1003
PLUS25  1013   -  1010
PLUS30  1025   -  1017
PLUS40  1042   -  1024
```

```
PLUSER  1021   -  1041
PURGA    263   -   255
R9=T    1463   -
RESETI   364   -   325
RNGERR   464   -
RTNP+2  1230   -  1261 1222 1213 1204
SDHMSC  1301   -  1273
SDHMSK  1306   -
SETA10  1417   -  1414
SETA20  1425   -  1430
SETA30  1427   -  1424
SETA40  1431   -  1426 1416 1411
SETAF   1401   -
SETAF0  1410   -
SETSDE  1515   -  1564 1507
SETSW   1504   -
SHFTON   304   -   332  262
SKPAL1  1125   -
SKPAL2  1130   -  1126
SKPALC  1122   -
SKPALM  1123   -
SRBF08  1156   -  1171
SRBF10  1157   -  1201 1155
SRBF30  1202   -  1163
SRHBFI  1137   -
SRHBUF  1141   -
STA100   571   -   566
STA150   611   -   607
STA160   637   -   615
STA200   657   -   613
STA325   737   -   724
STA340   746   -   765
STA342   751   -   747
STA345   756   -
STA350   766   -  1051  757  743
T+X      773   -
TIMEIN  1557   -  1550
TMSG     431   -
WAITK    422   -   426
WAITK6   420   -
WAITKD   417   -
XTMR05  1510   -  1506
XTMR10  1517   -  1514
XTMR30  1526   -  1523
XYZA20   525   -   523
XYZA25   541   -   550
XYZA30   545   -   540
XYZA40   551   -   534
XYZALM   501   -
```

ENTRY TABLE (BWTMB3 = TIMER ROM QUAD 3 = TM2 = ADDRESSES @54000-55777)

115860	1440	-
36000	454	-
A-DHMS	1274	-
ACKALM	1220	-
ACT120	21	-
ACT125	23	-
ACT170	112	-
ACT180	121	-
ACT185	123	-
ADATE	1556	-
ALMBST	400	-
ALMCAT	6	-
ALMSST	1205	-
ATD120	1727	-
ATD125	1730	-
ATIM24	1537	-
ATIME	1547	-
CHKBUF	1052	-
CLRAL0	1457	-
CLRALD	1450	-
CLRALM	1456	-
CLRALS	1451	-
CLRALW	1455	-
DATEIN	1567	-
DST10	347	-
FNDEOB	1151	-
GDHMS	1270	-
IGDHMS	1266	-
INTVAL	1234	-
KEY-FC	434	-
NEWLOC	1111	-
NEWLSK	1107	-
NEWM.X	1214	-
NOROOM	616	-
R9=T	1463	-
RNGERR	464	-
RTNP+2	1230	-
SDHMSC	1301	-
SDHMSK	1306	-
SETAF	1401	-
SETAF0	1410	-
SETSW	1504	-
SKPAL1	1125	-
SKPALC	1122	-
SKPALM	1123	-
SRHBFI	1137	-
SRHBUF	1141	-
T+X	773	-
TMSG	431	-
WAITK6	420	-
WAITKD	417	-
XYZALM	501	-

EXTERNAL REFERENCES (BWTMB3 = TIMER ROM QUAD 3 = TM2 = ADR @54000-55777)

A-DHMS	343	354	
A-DHMS	344	355	
ACT120	341		
ACT120	342		
ACT125	151		
ACT125	152		
ACT170	423		
ACT170	424		
ACT180	352		
ACT180	353		
ACT185	302		
ACT185	303		
ADJ100	766		
ADJ100	767		
ALMBST	274		
ALMBST	275		
ALMSST	137	336	
ALMSST	140	337	
APNDNW	1731		
APNDNW	1732		
ATD120	1666	1745	
ATD120	1667	1746	
ATD125	1713	1723	
ATD125	1714	1724	
C=T+D0	516		
C=T+D0	517		
CHECK	503	530	
CHECK	504	531	
CHECKX	773	1406	
CHECKX	774	1407	
CHKBUF	660		
CHKBUF	661		
CHKLB	126		
CHKLB	127		
CHKXM	505		
CHKXM	506		
CLLCDE	356	371	620
CLLCDE	357	372	621
DSA2ND	100		
DSA2ND	101		
DSAMS0	56		
DSAMS0	57		
DSAMSG	374		
DSAMSG	375		
DSPDT	361		
DSPDT	362		
DSPINT	1254		
DSPINT	1255		
DSPTMP	347		
DSPTMP	350		
DSTMDA	24		
DSTMDA	25		
ENCP00	130	1603	
ENCP00	131	1604	
ENLCD	1256	1672	
ENLCD	1257	1673	

```
SDHMSC  1245
SDHMSC  1246
SKPAL1   675  1055
SKPAL1   676  1056
SKPALC  1107  1206
SKPALC  1110  1207
SKPALM   411
SKPALM   412
SRHEFI     6
SRHEFI     7
SRHBUF   400   611
SRHBUF   401   612
T=T+TP  1042
T=T+TP  1043
TERR20   543
TERR20   544
TERR50   635
TERR50   636
TGLSHF   237   304
TGLSHF   240   305
TMEXIT   214
TMEXIT   215
TMRKEY  1337
TMRKEY  1340
TMROFF   162
TMROFF   163
TMRSTS  1573
TMRSTS  1574
TMSG      40    47
TMSG      41    50
TO12H   1650
TO12H   1651
TO24H   1641
TO24H   1642
UNNOR1  1412
UNNOR1  1413
UNNORX  1504  1561
UNNORX  1505  1562
WAITX6    51
WAITX6    52
WAITXD    27    75   106
WAITXD    30    76   107
X20Q8   1247
X20Q8   1250

End of VASM assembly
********************************************************************************
VASM ROM ASSEMBLY          REV.  6/81A            HP-82182A TIMER MODULE

OPTIONS: L C S          COCONUT TIMER             ADDRESSES @56000-57777

     2                  FILE   BWTMB4             COCONUT TIMER Q4 = TM3
*
*
********************************************************************************
* M306 = MONTH * 30.6                                              1-14-81 RSW
*
* IN: C= MONTH   [NOT IN C[S] OR C[0], AND THE REST OF C SHOULD BE
*                 CLEARED TO BE SAFE]
* ASSUME:  DEC MODE
```

```
* OUT:     C= 30.6 * C
* !! THIS INCLUDES FRACTIONAL PART -- THE CALLING ROUTINE MUST CLEAN UP!
* USES:    A, C    (NO PT, NO ST, NO DADD, NO PFAD, NO ARITH MODE,
*                   +0 SUB LEVELS, NO TIMER CHIP ACCESS)
*
    16                  ENTRY  M306
    17    0 M306    416 A=C
    18    1         756 C=C+C
    19    2        1016 C=A+C               C= 3*C
    20    3         416 A=C
    21    4         756 C=C+C               C= 6*C
    22    5        1716 C SR                C= 0.6*C
    23    6        1756 A SL                A= 30*C
    24    7        1016 C=A+C               C=30.6*C
    25   10        1740 RTN
*
*************************************************************************
*                                                              1-6-81 RSW
* NDAYS - CONVERT INTEGER NUM OF DAYS TO NORMALIZED FLOATING POINT FORM
*
* IN:      C= DDDDDD00000000 ( 6 DIGITS OF DAYS )
* ASSUME:  NOTHING
* OUT:     A= C= POSITIVE NORMALIZED FLOATING POINT NUMBER OF DAYS
*          DEC MODE
* USES:    A, C, ACTIVE PT, +1 SUB LEVEL, ARITH MODE
*          (NO ST, NO DADD, NO PFAD, NO TIMER CHIP ACCESS)
*
*
    39                  ENTRY  NDAYS
    40   11 NDAYS  1716 C SR                C= 0 DDDDDD0000 000
    41   12         460 LDI                 LOAD LOW 12 BITS OF C WITH
    42   13           4 CON    4            EXPONENT= 5 - 1
    43   14           1 GOSUB  NORM         NORMALIZE THE NUMBER
    43   15           0                     *TIMER ROM:  TM0, @0532
    44   16        1240 SETDEC              A= C= NORMALIZED # OF DAYS
    45   17        1740 RTN
*************************************************************************
* UNNORM = UNNORMALIZE                                         1-5-81 RSW
*
* IN:     A=   NORMALIZED FLOATING POINT NUMBER (EXPONENT MUST BE VALID;
*              NEGATIVE ZERO, A.XS= 1-8, OR NON-BCD DIGITS WON'T WORK)
*         C.X= POSITIVE EXPONENT TO WHICH TO NORMALIZE
*            = [NUMBER OF DIGITS TO LEFT OF DECIMAL POINT] - 1
*              (NEG EXPONENT CAN GIVE TROUBLE AS WHEN C.X=977 & A.X=085)
*
* !!!! WARNING:  IF C.X < A.X, AN ERROR WILL RESULT!
*
* ASSUME:  NOTHING
* OUT:     P+1: A= UNNORMALIZED NUMBER (WITH LEADING ZEROS)
*                 NOTE: UNNORM MAY OUTPUT A NEGATIVE ZERO!!!!!
*               DEC MODE, PT= 12
*          P+2: ERROR EXIT WITH HEX MODE SET (INPUT NUMBER WAS TOO BIG)
* USES:    A[12:0], C.X, C[6:3] USED ON ERR EXIT, ACTIVE PT, ARITH MODE
*          (NO ST, NO DADD, NO PFAD, +0 SUB LVLS, NO TIMER CHIP ACCESS)
*
*
*    *     *     *     *     *     *     *     *     *     *     *     *
* UNNORX = UNNORMALIZE THE X REGISTER CONTENTS                 1-20-81 RSW
*
* IN & ASSUME: PERIPHERALS DISABLED
```

```
*              "X" CONTAINS A FLOATING POINT NUMBER
* OUT:         SAME AS UNNORM BUT ALSO:  CHIP 0 ENABLED, S9=1
* USES:        A,B,C, ACTIVE PT, S9, +2 SUB LEVELS, ARITH MODE, DADD
*              (NO PFAD, NO TIMER CHIP ACCESS)
*
*
    76                  ENTRY  UNNORX
    77   20 UNNORX    1 GOSUB  CHECKX        ERROR IF X= ALPHA DATA
    77   21           0                      *TIMER ROM:  TM0, @0240
    78
    79                  ENTRY  UNNOR1
    80                  ENTRY  UNNOR2
    81                  ENTRY  UNNORM
    82   22 UNNOR1  416 A=C
    83   23 UNNOR2  460 LDI                    LOAD LOW 12 BITS OF C WITH
    84   24           1 CON    1              ERROR IF HH > 99
    85   25 UNNORM 1240 SETDEC                 35 MAX FOR .01 SEC INPUT
    86   26          246 AC EX  X              A= MAX EXP, C= ACTUAL EXP
    87   27         1534 PT=    12
    88   30         1106 C=A-C  X              C.X= MAX - ACTUAL
    89   31         1066 C=C+1  XS             MAX < ACTUAL ?
    90   32            1 GOLC   RTNP+2         YES, DATA ERROR
    90   33            3                       *TIMER ROM:  TM2, @1230
    91   34         1166 C=C-1  XS             NO, FIX C.X
    92   35            6 A=0    X              CLEAN UP X FIELD
    93   36           23 GOTO   UNORM6 (  40)
    94   37 UNORM4 1612 A SR    WPT            ADD LEADING ZEROS
    95   40 UNORM6 1146 C=C-1   X              DONE ?
    96   41         1763 GONC   UNORM4 (  37) NO
    97   42         1740 RTN
*
*******************************************************************************
*                                                                  1-12-81 RSW
* X20 - INTEGER MULTIPLY BY 20
*       THIS SPECIAL PURPOSE ROUTINE SETS UP FOR MULTIPLICATION BY 24 OR
*       60. THE PQ FIELD IN "A" IS MOVED TO C, SHIFTED (TO MULT BY 10)
*       AND DOUBLED, THEN ADDED TO "A" TO ADD 20 TIMES THE PQ FIELD TO
*       THE 2 MSD OF THE REST OF "A". C IS DOUBLED AGAIN BEFORE RETURN-
*       ING SO ANOTHER ADDITION WILL RESULT IN MULTIPLICATION BY 60.
*       SHIFTING THEN ADDING RESULTS IN THE PQ FIELD OF "A" BEING
*       MULTIPLIED BY 24 AND ADDED TO THE 2 MSD OF THE REST OF "A".
*
* IN:   A= INPUT DATA
* ASSUME:
*       1. DECIMAL
*       2. Q = 13
*       3. POINTER P AT RIGHTMOST DIGIT OF UPPER FIELD (P >= 1)
* OUT:
*       1. A = 20 * PQ FIELD + 2ND FIELD
*       2. C = 40 * PQ FIELD
*       3. POINTER UNCHANGED
* USES: A.C             (NO ST, NO PT, +0 SUB LEVELS, NO DADD, NO PFAD)
*
* X20Q -- SAME AS X20 EXCEPT EXPECTS P SELECTED & SETS Q= 13, SO USES Q
*
*
*
   125                  ENTRY  X20Q8
   126                  ENTRY  X20Q
   127                  ENTRY  X20
```

```
   128   43 X20Q8  1240 SETDEC
   129   44         240 SEL P
   130   45         434 PT=    8
   131   46 X20Q    340 SEL Q
   132   47        1334 PT=    13
   133   50         240 SEL P
   134   51 X20     116 C=0
   135   52         262 AC EX  PQ            UPPER DATA TO C
   136   53        1716 C SR                 = 10X DATA IN A
   137   54         756 C=C+C                = 20X
   138   55         516 A=A+C                A = 20 * UPPER + LOWER
   139   56         756 C=C+C                = 40X
   140   57        1740 RTN
*
*************************************************************************
* DSPTMP - DISPLAY TIME (LEFT JUSTIFIED)                    1-29-81 RSW
*
* IN:      A= ......HHMMSSCC WHERE HH= 24-HOUR FORM, "."= DON'T CARE
*          Q= (RIGHTMOST TIME DIGIT TO BE DISPLAYED) - 1, WHERE THE
*             LEFTMOST DISPLAY CHARACTER= DIGIT 11, RIGHTMOST= DIGIT 0
* ASSUME:  "CLK12"/"CLK24" BIT IN TIMER CHIP IS IN PROPER STATE
*          (12/24-HOUR DISPLAY)
* OUT:     P SELECTED, PERIPHERALS DISABLED, HEXMODE
*          S5= 1 (0)   FOR 24 (12) HOUR DISPLAY
*          S2= 1
* USES:    A, B[S] & B[Q:0], C,N, P,Q, S2,S5,S6,S8, +2 SUB LEVELS, DADD
*          PFAD, ARITH MODE, TIMER PT
*
*
*
* DSPTMD - DISPLAY TIME & DATE
*
* IN:      A= DDDDDDHHMMSSCC  (LIKE OUTPUT OF SDHMSC)
* ASSUME:  SAME AS DSPTMP
* OUT:     P SELECTED, CHIP 0 ENABLED, PERIPHERALS DISABLED, HEXMODE
*          S5= 1 (0)   FOR 24 (12) HOUR FORMAT
*          R8[13:8]= DAY NUMBER SINCE OCT 15, 1582
* USES:    A,B,C,G,N,R8[13:6], P,Q, S0,S2-S6,S8, +2 SUB LEVELS, ARITH
*          MODE, DADD, PFAD, TIMER PT
*
*
*
* DSTMDA - DISPLAY TIME & DATE OF AN ALARM
*          SAME AS DSPTMD EXCEPT:
* IN:      M.X= ALARM ADDRESS (!! MUST BE VALID ALARM ADDRESS !!)
*          PERIPHERALS DISABLED
*
*
   176                 ENTRY  DSTMDA
   177                 ENTRY  DSPTMD
   178   60 DSTMDA    1 GOSUB  A-DHMS          GET&CONVERT ALM TO D.H.M.S
   178   61           0                        *TIMER ROM:  TM2, @1274
   179   62 DSPTMD 1004 S2=    0
   180   63         340 SEL Q
   181   64        1234 PT=    7               DISPLAY MINUTES
   182   65          43 GOTO   DSPTM5 (  71)
   183
   184                 ENTRY  DSPTIM
   185                 ENTRY  DSPTMP
   186   66 DSPTIM  340 SEL Q
```

```
   187   67         234 PT=    5             DISPLAY SECONDS
   188   70 DSPTMP 1010 S2=    1
   189   71 DSPTM5  240 SEL P                A= DDDDDDHHMMSSCC
   190   72           1 GOSUB  CLLCDE        ENABLE & CLEAR DISPLAY
   190   73           0                      *MAINFRAME: CN11, @0360
   191                                       IN & ASSUME: NOTHING
   192                                       OUT: DISP ENA, RAM DISABLED
   193                                       USES: C[11:0], ACTIVE PT,
   194                                             DADD, PFAD. (NO ST,
   195                                             +0 SUB LVL, NO ARITH)
   196   74         256 AC EX
   197   75         160 N=C                 SAVE DAY# IN N
   198   76        1174 RCR    9            C= DHHMMSSCCDDDDD
   199   77         136 C=0    S
   200  100         416 A=C                 A= 0HHMMSSCCDDDDD
   201  101           1 GOSUB  DSPTM         DISPLAY TIME
   201  102           0                      *TIMER ROM:  TM1, @1653
   202  103        1014 ?S2=1                DISPLAY TIME ONLY ?
   203  104        1540 RTN C                YES, DONE
   204  105           1 GOSUB  ENLCD         ENABLE DISPLAY, DISABLE RAM
   204  106           0                      *MAINFRAME:  CN1, @1766
   205                                       IN & ASSUME: NOTHING
   206                                       OUT: DISPLAY ENABLED,
   207                                            RAM DISABLED
   208                                       USES: C.X, DADD, PFAD ONLY
   209  107         370 FLLDAB                ROTATE DISPLAY LEFT 6 CHARS
   210  110        1770 RABCL
   211  111         260 C=N
   212  112         416 A=C                 A= DDDDDD...= DAY# SINCE
   213  113           1 GOLONG DSPDT         ADD THE DATE  | 1/1/1900
   213  114           2                      *TIMER ROM:  TM0, @1373
*
*************************************************************************
* DSWEEK - DISPLAY DAY OF THE WEEK IN ENGLISH              1-14-81 RSW
* SHIFTS THE DAY OF WEEK INTO THE RIGHT SIDE OF THE DISPLAY
*
* INPUT : C= 0D000000000000
*         WHERE D IS DAY OF THE WEEK.  SUNDAY=0, ... , SATURDAY=6
*         A.S = 0 TO DISPLAY TWO LETTERS
*         A.S = 1 TO DISPLAY THREE LETTERS
* ASSUME: LCD ENABLED, RAM DISABLED, HEXMODE
* USES:   A[13:3], C, +1 SUB LEVEL
*         (NO PT, NO ST, NO DADD, NO PFAD, NO TIMER CHIP ACCESS)
*
*
   228                   ENTRY  DSWEEK
   229  115 DSWEEK  576 A=A+1  S
   230  116        1174 RCR    9            C= 0 000000000D 000
   231  117         432 A=C    M
   232  120         772 C=C+C  M
   233  121         532 A=A+C  M            A.M= 3*(DAY OF WEEK)
   234                 LEGAL                (CLEAR THE CARRY FLAG)
   235  122           1 GOSUB  DSWEKA        DISPLAY DAY OF WEEK ALPHA
   235  123           0                      *TIMER ROM:  TM3, @0151
   236  124          23 CON    @23           S
   237  125          25 CON    @25           U
   238  126          16 CON    @16           N
   239  127          15 CON    @15           M
   240  130          17 CON    @17           O
   241  131          16 CON    @16           N
```

```
   242  132          24 CON    @24           T
   243  133          25 CON    @25           U
   244  134           5 CON    @05           E
   245  135          27 CON    @27           W
   246  136           5 CON    @05           E
   247  137           4 CON    @04           D
   248  140          24 CON    @24           T
   249  141          10 CON    @10           H
   250  142          25 CON    @25           U
   251  143           6 CON    @06           F
   252  144          22 CON    @22           R
   253  145          11 CON    @11           I
   254  146          23 CON    @23           S
   255  147           1 CON    @01           A
   256  150          24 CON    @24           T
   257
   258                  ENTRY  DSWEKA
   259  151 DSWEKA  660 C=STK                C.M= ADDR OF TOP OF TABLE
   260  152        1032 C=A+C  M              C.M= TOP OF TABLE + 3*(DAY)
   261  153 DSWK30 1460 CXISA
   262  154        1750 SLSABC
   263  155        1072 C=C+1  M
   264  156         676 A=A-1  S
   265  157        1743 GONC   DSWK30 ( 153)
   266  160        1740 RTN
   267
*************************************************************************
* BEPI = BEEP INITIALIZE ROUTINE                              4-28-81 RSW
*
* IN & ASSUME: HEXMODE
* OUT:   CHIP 0 ENABLED, RAM DISABLED, A[1:0]= INPUT S7-S0
*        IF AUDIO ENABLE FLAG IS NOT SET:
*           S7-S0 = 00000000
*        IF AUDIO ENABLE FLAG IS SET:
*           S7-S0 = 00000001
* USES:  A.X, C, S0-S7, DADD, PFAD, +1 SUB LEVEL
*           (NO PT, NO TIMER CHIP ACCESS)
*
   280                  ENTRY  BEPI
   281  161 BEPI   1630 C=ST
   282  162         406 A=C    X             SAVE S7-S0 IN A[1:0]
   283  163        1704 CLR ST
   284  164           1 GOSUB  ENCP00        ENABLE CHIP 0
   284  165           0                      *MAINFRAME:  CN2, @0522
   285                                       IN:     NOTHING
   286                                       ASSUME: NOTHING
   287                                       OUT: CHIP 0 ENABLED, C.X= 0
   288                                            PERIPHERALS DISABLED
   289                                       USES:   C.X ONLY
   290
   291  166        1670 C=REGN 14           C= USER FLAG REGISTER
   292  167         274 RCR    5
   293  170         766 C=C+C  XS
   294  171         766 C=C+C  XS
   295  172         766 C=C+C  XS            AUDIO ENABLE FLAG SET ?
   296  173        1640 RTN NC               NO, DON'T BEEP (S7-S0=0..0)
   297  174        1610 S0=    1             YES, S7-S0= 00000001, BEEP
   298  175        1740 RTN
*
*************************************************************************
```

```
* BEEP2K - ROUTINE SOUNDS TWO BEEPS                          1-6-81 RSW
*          IF THE AUDIO ENABLE FLAG IS NOT SET, IT WILL NOT BEEP
*          IF THE AUDIO ENABLE FLAG IS SET:
*             1.  BEEP
*             2.  PAUSE BRIEFLY, CHECKING KEYBOARD IF S8= 0
*             3.  BEEP A SECOND TIME, CHECKING KEYBOARD IF S8= 1
*
*             IF S8= 1 AND KEY IS DETECTED, BEEP2K RETURNS IMMEDIATELY
*             AFTER CLEARING THE FLAG OUT REGISTER AND RESTORING THE
*             INPUT STATUS BITS.
*
*             !!!! CAUTION !!!!   SINCE THE KEYBOARD IS NOT CLEARED IN
*             "BEEP2K", AN EARLIER KEYSTROKE COULD CAUSE A RETURN !!!!
*
*
* IN & ASSUME:  S8= 1     TO CHECK KEYBOARD
*               S8= 0     TO IGNORE KEYBOARD
*               HEXMODE
*               FLAG OUT REGISTER = 0
* OUT:          CHIP 0 ENABLED, FLAG OUT REGISTER CLEARED
*               INPUT STATUS BITS RESTORED
*               !! THE KEYBOARD IS NOT CLEARED !!
* USES:         A,X, C, +2 SUB LEVELS, DADD, PFAD
*               (NO PT, S0-S7 RESTORED, NO TIMER CHIP ACCESS)
*
*  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
*
* BEEP2  - SAME AS BEEP2K EXCEPT SETS S8= 0 SO IT IGNORES THE KEYBOARD
*
* BEEPKP - PAUSE, THEN BEEP
*          SAME AS BEEP2K EXCEPT IT DOESN'T DO THE FIRST BEEP.
*
* BEEPK  - SAME AS BEEP2K EXCEPT IT ONLY DOES THE SECOND BEEP
*          (IT SETS S8= 1 SO THE KEYBOARD WILL BE CHECKED)
*
*************************************************************************
*
  338                   ENTRY  BEEP2
  339                   ENTRY  BEEP2K
  340  176 BEEP2   404 S8=    0             DON'T CHECK KEYBOARD
  341  177 BEEP2K    1 GOSUB  BEPI          BEEP INITIALIZE ROUTINE
  341  200            0                     *TIMER ROM:  TM3, @0161
  342  201          460 LDI                 LOAD LOW 12 BITS OF C WITH
  343  202         1131 CON    601          ..... FIRST BEEP .....
  344  203 BEP210 1330 FEXSB               (NO KEYBOARD CHECK)
  345  204         1146 C=C-1  X
  346  205         1763 GONC   BEP210 ( 203)
  347  206           33 GOTO   BEPK05 ( 211)
  348
  349                   ENTRY  BEEPKP
  350  207 BEEPKP    1 GOSUB  BEPI          BEEP INITIALIZE ROUTINE
  350  210            0                     *TIMER ROM:  TM3, @0161
  351  211 BEPK05   460 LDI                 LOAD LOW 12 BITS OF C WITH
  352  212           36 CON    30           ... PAUSE BETWEEN BEEPS ...
  353  213 BEPK10   414 ?S8=1               CHECK KEYBOARD ?
  354  214           33 GONC   BEPK20 ( 217) NO
  355  215         1714 CHK KB               KEY DOWN ?
  356  216          207 GOC    BEPK50 ( 236) YES
  357  217 BEPK20 1146 C=C-1  X
  358  220         1733 GONC   BEPK10 ( 213)
```

```
   359  221          43 GOTO   BEPK28 ( 225)
   360
   361                 ENTRY  BEEPK
   362                 ENTRY  BEEPNK
   363  222 BEEPK   410 S8=    1              CHECK THE KEYBOARD
   364  223 BEEPNK    1 GOSUB  BEPI           BEEP INITIALIZE ROUTINE
   364  224           0                       *TIMER ROM:  TM3, @0161
   365  225 BEPK28  460 LDI                   LOAD LOW 12 BITS OF C WITH
   366  226         455 CON    301            ..... SECOND BEEP .....
   367  227 BEPK30 1330 FEXSB
   368  230         414 ?S8=1                 CHECK THE KEYBOARD ?
   369  231          33 GONC   BEPK40 ( 234) NO
   370  232        1714 CHK KB                YES, KEY DOWN ?
   371  233          37 GOC    BEPK50 ( 236) YES
   372  234 BEPK40 1146 C=C-1  X
   373  235        1723 GONC   BEPK30 ( 227)
   374  236 BEPK50 1704 CLR ST
   375  237        1130 F=SB                  CLEAR FLAG OUT REGISTER
   376  240         246 C=A    X
   376  241         406                       (INSERTED BY ASSEMBLER)
   377  242        1530 ST=C                  RESTORE INPUT STATUS
   378  243        1740 RTN
*************************************************************************
* TMRSTS= TIMER STATUS                                          1-5-81 RSW
* GETS TIMER SOFTWARE STATUS FROM TIMER SCRATCH REG B AND PUTS IT
* IN S[7:0], SAVING THE OLD STATUS IN C[1:0]
*
* IN & ASSUME: TIMER SOFTWARE STATUS BITS IN SCRATCH REG B
* OUT:  ST[7:0]= TIMER STATUS BITS 7-0, INPUT S0-S7 SAVED IN C[1:0]
*       TIMER CHIP ENABLED, RAM DISABLED, PT=B,
*       C[13:2]= SCRATCH REGISTER B[13:2]
* USES: C, ST[7:0], DADD, PFAD, TIMER PT
*       (NO 41C PT, NO ARITH MODE, +0 SUB LEVELS)
*
*   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .
*
* TMRST -
*      IN: TIMER SOFTWARE STATUS BITS IN SCRATCH REGISTER B
*      ASSUME: TIMER CHIP ENABLED, RAM DISABLED, TIMER PT=B
*      OUT:    S0-S7= TIMER SOFTWARE STATUS BITS 0-7
*              C[1:0]= INPUT S0-S7, C[13:2]= TIMER SCRATCH REG B[13:2]
*      USES:   C, S0-S7 (NO PT, NO ARITH MODE, +0 SUB LEVELS)
*
   400                 ENTRY  TMRSTS
   401                 ENTRY  TMRST
   402  244 TMRSTS  460 LDI                   LOAD LOW 12 BITS OF C WITH
   403  245          20 CON2   1      0       10H = DISABLE RAM CHIP
   404  246        1160 DADD=C                 DISABLE RAM
   405  247         460 LDI                   LOAD LOW 12 BITS OF C WITH
   406  250         373 CON2   15     11      FBH = ENABLE TIMER CHIP
   407  251        1760 PFAD=C                 ENABLE TIMER
   408  252        1650 PT=B                   SELECT SCRATCH REGISTER B
   409  253 TMRST   470 RDSCR                  READ IT
   410  254          43 GOTO   CSTRTN ( 260)
*************************************************************************
* HWSTS= HARDWARE STATUS                                        1-5-81 RSW
* GETS TIMER CHIP HARDWARE STATUS BITS
*
* IN & ASSUME: NOTHING
* OUT:  S[7:0]= TIMER HARDWARE STATUS, INPUT S0-S7 SAVED IN C[1:0]
```

```
*       TIMER CHIP ENABLED, RAM DISABLED, PT=A
* USES: C, ST[7:0], TIMER PT, DADD, PFAD, +1 SUB LEVEL
*       (NO 41C PT, NO ARITH MODE)
*
   421                    ENTRY  HWSTS
   422  255 HWSTS     1 GOSUB  ENTMR          ENABLE TIMER CHIP, PT=A
   422  256            0                      *TIMER ROM:  TM0, @0342
   423  257          370 RDSTS                READ IT
   424  260 CSTRTN  1730 CST EX
   425  261         1740 RTN
*************************************************************************
* TO12H = TO 12-HOUR
* CONVERTS 12- OR 24-HOUR INPUT TO 12-HOUR OUTPUT (UNLESS HOUR > 23)
*
* IN:       A= #HHMMSSCC.....    WHERE "#" =   0 FOR AM
*                                              1-9 FOR PM
*                                AND "." = DON'T CARE
* ASSUME:   NOTHING
* OUT:      A= C= #HHMMSSCC..... WHERE HH= 12-HOUR FORMAT (UNLESS HH>23)
*           HEXMODE
*           S5= 0 (12-HOUR FORMAT) FOR HOUR < 24
*           S5= 1 (24-HOUR FORMAT) FOR HOUR >= 24
*                 (S6= 1 ALSO, AND THE HOUR WILL BE UNCHANGED)
*           S6= 0 FOR AM
*           S6= 1 FOR PM
*           PT= 1
* USES:     A[13:11], C, S5,S6, ACTIVE PT, ARITH MODE
*           (+0 SUB LEVELS, NO PFAD, NO DADD, NO TIMER CHIP ACCESS)
*
*
   446                    ENTRY  TO12H
   447  262 TO12H    256 AC EX
   448  263          204 S5=    0             ASSUME 12-HOUR FORMAT
   449  264          504 S6=    0             ASSUME "AM"
   450  265          674 RCR    11
   451  266          416 A=C                  A= MMSSCC.....#HH
   452  267          460 LDI                  LOAD LOW 12 BITS OF C WITH
   453  270           22 CON2   1      2       12 FOR AM/PM COMPARISON
   454  271         1434 PT=    1
   455  272         1412 ? A<C  WPT            HOUR < 12 ?
   456  273          167 GOC    T12H30 ( 311) YES
   457  274         1552 ? A#C  WPT            HOUR = 12 ?
   458  275          123 GONC   T12H25 ( 307) YES
   459  276         1240 SETDEC
   460  277          712 A=A-C  WPT            PM HOUR= HOUR - 12
*  THIS CODE IS FOR "ATIME" TO MAKE HOUR > 23 SWITCH TO 24-HOUR FORMAT
   462  300         1412 ? A<C  WPT            PM HOUR < 12 ?
   463  301           67 GOC    T12H25 ( 307) YES
   464  302          512 A=A+C  WPT            NO, RESTORE HOUR
   465  303          210 S5=    1              24-HOUR FORMAT
*  END OF CODE FOR "ATIME"
   467  304           33 GOTO   T12H25 ( 307)
   468  305 T12H20 1526 ? A#0  XS             PM?
   469  306           63 GONC   T12H40 ( 314) NO
   470  307 T12H25   510 S6=    1              REMEMBER PM
   471  310           43 GOTO   T12H40 ( 314) NO
   472  311 T12H30 1512 ? A#0  WPT            HOUR= 00?
   473  312         1737 GOC    T12H20 ( 305) NO
   474  313          406 A=C    X              YES, SET HOUR = 12
   475  314 T12H40 1140 SETHEX
```

```
   476  315          256 AC EX
   477  316           74 RCR    3
   478  317          416 A=C
   479  320         1740 RTN
*********************************************************************
* T=T+TP = TIME + PROCESSING TIME                          1-5-81 RSW
*
* ADDS THE PROCESSING TIME TO THE NEW TIME&DATE AND SETS THE TIME&DATE.
*
* IN:      C= CALCULATED NEW TIME= "TC" = 00SSSSSSSSSCC
*          PERIPHERALS (OTHER THAN TIMER) DISABLED
* ASSUME:  REG 9= CLOCK TIME AT START OF PROCESSING
* OUT:     DEC MODE, C= NEW CORRECTED TIME (MAY BE 1/100 SEC SLOW)
*          TIMER CHIP ENABLED, RAM DISABLED, TIMER PT=A
* USES:    A,B,C, ACTIVE PT, TMR PT, +1 SUB LVL, DADD, PFAD, ARITH MODE
*          (NO ST)
*
* !!!! CAUTION !!!!   DOESN'T CHECK FOR TC+TP > 12/31/2199
*
*
   496                   ENTRY  T=T+TP
   497  321 T=T+TP  416 A=C
   498  322          116 C=0
   499  323         1160 DADD=C
   500  324         1170 C=REGN 9             C= OLD CLOCK TIME
   501  325          356 BC EX                B= OLD CLOCK TIME (BEGIN)
   502  326            1 GOSUB  ENTMR         ENABLE TIMER CHIP, PT=A
   502  327            0                      *TIMER ROM:  TM0, @0342
   503  330          170 RCTIME               C= CURR TIME (START CORR'N)
   504  331         1240 SETDEC
   505  332          256 AC EX                A= CURR TIME, C= CALC TIME
   506  333          616 A=A-B                A= PROCESSING TIME
   507  334         1016 C=A+C                C= CORRECTED TIME
   508  335          150 WDTIME               STORE TIME & CORRECT IF NEC
   509  336         1740 RTN
*********************************************************************
* STOPSW = STOP STOPWATCH                                  1-8-81 RSW
*********************************************************************
   513  337          227 CON    @227          W
   514  340           23 CON    @23           S
   515  341           20 CON    @20           P
   516  342           17 CON    @17           O
   517  343           24 CON    @24           T
   518  344           23 CON    @23           S
   519
   520                   ENTRY  STOPSW
   521  345 STOPSW    1 GOSUB  INITMR         INIT TIMER IF NECESS, PT=B
   521  346            0                      *TIMER ROM:  TM0, @1524
   522  347         1450 STOPC                STOP STOPWATCH
   523  350         1740 RTN                  NOTE: THE TIMER CHIP
   524                                       AUTOMATICALLY DISABLES WHEN
   525                                       A RAM CHIP IS ENABLED.
*********************************************************************
* SUM3D5                                                   1-22-81 RSW
*
* THIS ROUTINE IS BASED ON THE DATE ALGORITHM GIVEN ABOVE.
* IT CALCULATES N.M + 578164 - SUM3(Y).
*
* IN:      C.M= 000000YYYY = YEAR
*          N.M= ACCUMULATING TOTAL
```

```
* ASSUME: NOTHING
* OUT:    C.M= INPUT ACCUMULATING TOTAL + 578164 - SUM3(Y)
*              (RIGHT JUSTIFIED IN C.M)
*         DEC MODE
*         B.M= 000000YYYY = COPY OF INPUT YEAR
* USES:   A,B,C,N,M, ACTIVE PT, ARITH MODE
*         (NO ST, +0 SUB LVLS SINCE SAVES A SUB LVL, NO DADD, NO PFAD
*          NO TIMER CHIP ACCESS)
*
   543                    ENTRY  SUM3D5            (X= DON'T CARE)
   544  351 SUM3D5 1140 SETHEX                C= X 000000YYYY XXX
   545  352         374 RCR    10             C= 0 00YYYYXXXX 000
   546  353         660 C=STK                 SAVE RETURN ADDRESS
   547  354         174 RCR    4              C= R 000000YYYY RRR
   548  355        1240 SETDEC
   549  356         356 BC EX                 SAVE YEAR & RETURN ADDRESS
   550  357          16 A=0
   551  360        1034 PT=     2
   552  361         320 LC      3
   553  362         620 LC      6
   554  363         520 LC      5             C= R 000000YYYY 365
   555  364         246 AC EX   X             A.X= 365, C.X= 0
   556  365        1066 C=C+1   XS            3-DIGIT MULTIPLIER
*   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .
*
* INTEGER MULTIPLY
*     START WITH A[M]= 0 AND A[X]= MULTIPLIER.  THE YEAR IN THE
*     MANTISSA OF C IS ADDED TO A[M] FOR EACH COUNT OF A[0].  ALL OF "A"
*     IS SHIFTED TO POSITION THE NEXT DIGIT OF THE MULTIPLIER IN A[0].
*     THE NUMBER OF DIGITS OF A.X USED IS (C.XS+2)
*     (SO C.XS=0 USES A[1:0], AND C.XS=1 USES A[2:0])
* RESULT: A= A[X] * C[M]
*
* INPUT:  1. A[M]= 0
*         2. MULTIPLICAND IN MANTISSA OF B
*         3. MULTIPLIER RIGHT-JUSTIFIED IN A.X
*         4. C.XS= (NUMBER OF DIGITS IN "A.X" TO USE AS MULTIPLIER) - 2
* ASSUME: DECMODE
* OUT:
*         1. ANSWER IN A (OPTIONAL POSITION)
*         2. PT= 0
*         3. C.XS= 9, REST OF C UNCHANGED
* USES:   A, C.XS, ACTIVE PT
*
*
   579  366        1066 C=C+1   XS            C.XS=2 (3-DIGIT MULTIPLIER)
   580  367        1634 PT=     0
   581  370          23 GOTO    IMPY3   ( 372)
*
   583  371 IMPY2   472 A=A+B   M             ADD CONSTANT TO A
   584  372 IMPY3   642 A=A-1   PT            DECREMENT A[0]
   585  373        1763 GONC    IMPY2   ( 371)
*
   587  374        1616 A SR                 POSITION FOR NEXT PASS
   588  375        1166 C=C-1   XS            USED ALL SPECIFIED DIGITS ?
   589  376        1743 GONC    IMPY3   ( 372) NO, DO NEXT ONE
   590
*   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .   .
   592                                        (D...D= Y*365)
   593  377         256 AC EX                 C= 0 00000DDDDD DDD
```

```
   594  400         674 RCR    11            C= 0 00DDDDDDDD 000
   595  401         416 A=C
   596  402         332 C=B    M             C= 0 000000YYYY 000
   597  403        1074 RCR    2             C.M= INT[Y/100]
   598  404         732 A=A-C  M             A.M= 365*Y - INT[Y/100]
   599  405         260 C=N                  C= TOTAL
   600  406         272 AC EX  M             C.M= Y*365 - INT[Y/100],
   601  407        1132 C=A-C  M             A.M= TOTAL
   602  410         160 N=C                  STORE NEW TOTAL
   603  411         172 A=B    M             A= 0 000000YYYY 000
   603  412         232                      (INSERTED BY ASSEMBLER)
   604  413         116 C=0
   605  414         534 PT=    6
   606  415         420 LC     4             C= DIVISOR=0 0000004000 000
   607  416        1234 PT=    7
   608  417           1 GOSUB  IDVD4         A= Y/4 (4-DIGIT ANSWER)
   608  420           0                      *TIMER ROM:  TM0, @0320
   609  421         256 AC EX                C= 0 0000QQQQ00 000
   610  422        1074 RCR    2             C.M= INT[Y/4]
   611  423         416 A=C                  A= 0 000000QQQQ 000
   612  424        1074 RCR    2             C= 0 00000000QQ QQ0
   613  425         532 A=A+C  M             C.M= INT[Y/4] + INT[Y/400]
   614  426         116 C=0
   615  427         434 PT=    8
   616  430         520 LC     5
   617  431         720 LC     7
   618  432        1020 LC     8
   619  433         120 LC     1
   620  434         620 LC     6
   621  435         420 LC     4
   622  436         256 AC EX                A.M= 0000578164
   623                                      C= INT[Y/4] + INT[Y/400]
   624  437         732 A=A-C  M
   625  440         316 C=B                  C= R 000000YYYYRRR
   626  441         374 RCR    10
   627  442         560 STK=C                ! WORKS IN DECMODE !
   628  443         260 C=N                  C.M= TOTAL
   629  444        1032 C=A+C  M
   630  445        1740 RTN
*****************************************************************
* CLRFLG = CLEAR FLAGS                                 2-25-81 RSW
*
* CLEARS THE FOLLOWING FLAGS:  ALPHA, PARTIAL KEY SEQUENCE, PAUSING,
*                              CATALOG, SHIFT, DATA ENTRY, MESSAGE.
* AND CLEARS THE SHIFT & ALPHA ANNUNCIATORS
*
* IN & ASSUME: HEXMODE
* OUT:  STATUS SET 0 UP, CHIP 0 ENABLED, PERIPHERALS DISABLED
*       AND THE ABOVE MENTIONED FLAGS CLEARED.
* USES: A,C, S0-S7, (MAYBE S13), +1 SUB LEVEL, DADD, PFAD
*       (NO PT, NO TIMER CHIP ACCESS)
*
   644                   ENTRY  CLRFLG
   645  446 CLRFLG       1 GOSUB  LDSST0       LOAD STATUS SET 0
   645  447              0                     *MAINFRAME:  CN1, @1627
   646                                         IN & ASSUME: NOTHING
   647                                         OUT: STATUS SET 0 UP,
   648                                          C= REG 14, CHIP 0 ENABLED,
   649                                          PERIPHERALS DISABLED.
   650                                         USES: C, S0-S7, DADD, PFAD
```

```
*
* NOTE:  THE "TRUN" CODE CAN BE ACCESSED DURING A PAUSE AS THE PAUSE OR
*        ANY DIGIT KEY DURING A PAUSE PASSES THROUGH THE NFRS ON THE
*        WAY TO "PAUSLP".
*        THE PAUSE FLAG CAN ONLY BE SET DURING A RUNNING PROGRAM, BUT
*        S13 WILL BE CLEARED, SO THIS CODE SETS S13=1 AGAIN.
*
   658  450         1414 ?S1=1                 PAUSING ?
   659  451           23 GONC   CLRF10 ( 453)  NO
   660  452         1310 S13=   1              YES, SET RUNNING FLAG
   661  453 CLRF10  1204 S7=    0              CLEAR ALPHA FLAG
   662  454         1630 C=ST
   663  455         1650 REGN=C 14
   664  456            1 GOSUB  RSTSQ          CLEAR PARTIAL KEY SEQUENCE,
   664  457            0                       *MAINFRAME:  CN0, @1605
   665                                         PAUSING, CATALOG, SHIFT,
   666                                         DATA ENTRY & MESSAGE FLAGS
   667                                         IN & ASSUME: CHIP 0 ENABLED
   668                                                PERIPHERALS DISABLED
   669                                         OUT: STATUS SET 0 UP,
   670                                             C=REG 14, THE 6 ABOVE-
   671                                             NAMED FLAGS CLEARED
   672                                         uSES: C, S0-S7, FLAGS ABOVE
   673                                           (NO PT, +0 SUB LEVELS, NO
   674                                            DADD, NO PFAD, NO ARITH)
   675  460            1 GOLONG ANNOUT         CLEAR SHIFT & ALPHA ANNUNS
   675  461            2                       *MAINFRAME:  CN1, @1534
   676                                         (DOES NOT TURN DISPLAY ON)
   677                                         IN & ASSUME: HEXMODE
   678                                         OUT: STATUS SET 0 UP,
   679                                             C=REG 14, CHIP 0 EN-
   680                                             ABLED, PERIPH DISABLED
   681                                         USES: A,C,S0-S7, PFAD, DADD
   682                                             (NO PT, +0 SUB LEVELS)
   683
*******************************************************************************
* CLKDSP = CLOCK DISPLAY                                          3-18-81 RSW
*
* IN:     S0-S7= TIMER SOFTWARE STATUS
* ASSUME: NOTHING
* OUT:    P SELECTED, HEXMODE, PERIPHERALS DISABLED
* USES:   A,B,C,G,N,R8[13:6], P,Q, S0,S2-S6,S8, +3 SUB LVLS,
*         ARITH MODE, DADD, PFAD, TIMER PT
*
   693                 ENTRY  CLKDSP
   694  462 CLKDSP     1 GOSUB  GDHMS          A= C= DDDDDDHHMMSSCC
   694  463            0                       *TIMER ROM:  TM2, @1270
   695  464         1214 ?S7=1
   696  465            1 GOLNC  DSPTIM         NO, DISP HOURS, MIN & DATE
   696  466            2                       *TIMER ROM:  TM3, @0066
   697  467         1340 DISOFF                TURN DISPLAY OFF
   698  470            1 GOSUB  DSPTMD         DISPLAY HOURS, MIN & DATE
   698  471            0                       *TIMER ROM:  TM3, @0062
   699  472         1440 DISTOG                TURN DISPLAY ON
   700  473         1740 RTN
*******************************************************************************
* TIME+1 = GET INCREMENTED TIME                                   3-16-81 RSW
*
* IN & ASSUME: NOTHING
* OUT:  C= SLIGHTLY FUTURE TIME
```

```
*       HEXMODE
*       TIMER CHIP ENABLED, RAM DISABLED, TIMER PT=A
* USES: C, +1 SUB LEVEL, DADD, PFAD, ARITH MODE
*       (NO 41C, PT, NO ST)
*
***************
* FTIME -- SAME AS TIME+1 EXCEPT:
*          IN & ASSUME:  TIMER CHIP ENABLED, RAM DISABLED
*                         TIMER PT=A
*
   716                    ENTRY  TIME+1
   717                    ENTRY  FTIME
   718  474 TIME+1    1 GOSUB  ENTMR         ENA TIMER, DIS RAM, PT=A
   718  475           0                      *TIMER ROM:  TM0, @0342
   719  476 FTIME    70 RDTIME               C= CURRENT TIME
   720  477        1240 SETDEC
   721  500        1056 C=C+1
   722  501        1056 C=C+1
   723  502        1140 SETHEX
   724  503        1740 RTN
   725
   726                    FILLTO @530
        504        0000
        505        0000
        506        0000
        507        0000
        510        0000
        511        0000
        512        0000
        513        0000
        514        0000
        515        0000
        516        0000
        517        0000
        520        0000
        521        0000
        522        0000
        523        0000
        524        0000
        525        0000
        526        0000
        527        0000
        530        0000
*********************************************************************
* TRUN -- "NFR" ENTRY POINT                                3-10-81 RSW
*         (ONLY ACCESSED WHEN F13=1 OR I/O FLAG=1)
*
* THIS ENTRY POINT IS ACCESSED FROM THE NFRS IF SOME HARDWARE IS
* PULLING ON F13 OF THE FLAG IN LINE OR IF THE I/O FLAG IS SET.
* PARTIAL KEY SEQUENCES CAN ENTER HERE IF A KEY (OR KEYS) ARE HIT
* AND AN ALARM OCCURS BEFORE PASSING THROUGH THE NFRS.
   735                    ENTRY  TRUN
   736  531 TRUN    410 S8=    1
   737  532         413 GOTO   LSWK01  ( 573)
   738
*
* TRUN10
*
* IN: S0-S7= TIMER SOFTWARE STATUS
*     TIMER CHIP ENABLED, RAM DISABLED, TIMER PT=B
```

```
*     C[11]= TIMER SCRATCH REGISTER B[11]= USER FLAGS 8-11
*
   746  533 TRUN10 1554 ALARM?
   747  534         273 GONC   TRUN40 ( 563) NO, SOME OTHER PERIPHERAL
   748  535        1614 ?S0=1                JUST CAME FROM DSWKNO ?
   749  536         253 GONC   TRUN40 ( 563) NO, ALM CODE DURING FUNCION
   750  537         416 A=C                  A[11]= USER FLAGS 8-11
   751  540         116 C=0
   752  541        1160 DADD=C
   753  542        1670 C=REGN 14
   754  543         634 PT=    11
   755  544         242 AC EX  PT            RESTORE USER FLAGS 8-11
   756  545        1650 REGN=C 14
   757  546           1 GOSUB  HWSTS         S0-S7= HARDWARE STATUS
   757  547           0                      *TIMER ROM:  TM3, @0255
   758                                       C[1:0]= SOFTWARE STATUS
   759  550        1614 ?S0=1                ALARM A?
   760  551          33 GONC   TRUN35 ( 554) NO
   761  552        1530 ST=C                 S2= "RUN LABEL" BIT
   762  553         133 GOTO   TRUN55 ( 566)
   763  554 TRUN35  114 ?S4=1                INT TIMER ALARM ?
   764  555         103 GONC   TRUN50 ( 565) NO
   765  556           1 GOSUB  CLLCDE        YES, CLEAR DISPLAY
   765  557           0                      *MAINFRAME: CN11, @0360
   766  560         510 S6=    1             DON'T PUT UP DEFAULT DISP
   767  561        1004 S2=    0             CLEAR "RUN LABEL" BIT
   768  562          53 GOTO   TRUN60 ( 567)
   769  563 TRUN40    1 GOSUB  CLRALD        CLR "DSWKNO" BIT, STOP CLK
   769  564           0                      *TIMER ROM:  TM2, @1450
   770  565 TRUN50 1004 S2=    0             CLEAR "RUN LABEL" BIT
   771  566 TRUN55  504 S6=    0             DON'T SET MESSAGE FLAG
   772  567 TRUN60 1604 S0=    0             NOT LIGHT SLEEP
   773  570           1 GOLONG LSWK90        LEAVE DISP ON, WAKE UP
   773  571           2                      *TIMER ROM:  TM3, @0730
*************************************************************************
   775                 ENTRY  LSWK00
   776  572 LSWK00  404 S8=    0
*  MUST NOT USE MORE THAN 1 SUB LEVEL DUE TO PARTIAL KEY SEQUENCES
   778  573 LSWK01  530 M=C
   779  574           1 GOSUB  ENTMR         ENABLE TIMER CHIP, PT=A
   779  575           0                      *TIMER ROM:  TM0, @0342
   780  576           1 GOSUB  INITM1        INIT TIMER IF NECESSARY
   780  577           0                      *TIMER ROM:  TM0, @1527
   781  600         470 RDSCR
   782  601        1530 ST=C                 PUT UP SOFTWARE STATUS
   783  602         414 ?S8=1                CALLED FROM TRUN ?
   784  603        1307 GOC    TRUN10 ( 533) YES
   785  604        1554 ALARM?
   786  605          57 GOC    LSWK10 ( 612) YES
   787  606         114 ?S4=1                DOING CLOCK DISPLAY ?
   788  607 LSWK05    1 GOLNC  RMRT01        NO
   788  610           2                      *TIMER ROM:  TM2, @1624
   789  611         733 GOTO   LSWK6J ( 704) YES, END IT (SAVE SUB LVL,
   790  612 LSWK10 1750 PT=A                  PARTIAL KEY SEQUENCE)
   791  613         370 RDSTS
   792  614        1730 CST EX               S0-S7= HARDWARE STATUS
   793                                       C[1:0]= SOFTWARE STATUS
   794  615         114 ?S4=1                INT TIIMER ALARM ?
   795  616           1 GOLNC  LSWK80        NO, SOME OTHER ALARM
   795  617           2                      *TIMER ROM:  TM3, @0723
```

```
796  620         1730 CST EX
797  621           14 ?S3=1                  SETTING TO CLOCK MODE ?
798  622           37 GOC    LSWK15 ( 625) YES
799  623          114 ?S4=1                  DOING CLOCK DISPLAY ?
800  624          703 GONC   LSWK65 ( 714) NO
801  625 LSWK15   106 C=0    X              YES, CHECK FOR EVIDENCE
802  626         1160 DADD=C                    OF A KEY DOWN
803  627         1670 C=REGN 14
804  630         1474 RCR    1
805  631         1730 CST EX
806  632         1414 ?S1=1                 MESSAGE FLAG CLEARED ?
807  633          563 GONC   LSWK58 ( 711) YES, A KEY WENT DOWN
808  634          214 ?S5=1                 PARTIAL KEY SEQUENCE ?
809  635          547 GOC    LSWK58 ( 711) YES, A KEY WENT DOWN
810  636          514 ?S6=1                 DATA ENTRY FLAG SET ?
811  637          527 GOC    LSWK58 ( 711) YES, A KEY WENT DOWN
812  640         1730 CST EX
813  641           14 ?S3=1                  SETTING TO CLOCK MODE ?
814  642          573 GONC   LSWK70 ( 721) NO, ALREADY ESTABLISHED
815  643            1 GOSUB  CLKDSP         YES, PUT UP CLOCK DISPLAY
815  644            0                       *TIMER ROM:  TM3, @0462
816  645            1 GOSUB  TMRSTS         S0-S7= TIMER SW STATUS
816  646            0                       *TIMER ROM:  TM3, @0244
817  647          110 S4=     1
818  650         1630 C=ST
819  651         1650 PT=B
820  652          450 WRSCR                 UPDATE SOFTWARE STATUS
821                                          (IN CASE OF AN ALARM)
822  653         1750 PT=A
823  654 LSWK20    70 RDTIME                C= CURRENT TIME
824  655         1434 PT=    1
825  656         1352 ? C#0  WPT             ON SECONDS TRANSITION ?
826  657          127 GOC    LSWK25 ( 671) NO, HUNDREDTHS NON-ZERO
827  660         1214 ?S7=1                 YES, DISPLAY MINUTES ?
828  661          243 GONC   LSWK28 ( 705) NO, START SECONDS WAKEUPS
829  662            1 GOSUB  SDHMSC         C= DDDDDDHHMMSSCC, PT=3
829  663            0                       *TIMER ROM:  TM2, @1301
830  664         1352 ? C#0  WPT             ON MINUTES TRANSITION ?
831  665           47 GOC    LSWK25 ( 671) NO, SECONDS NON-ZERO
832  666          116 C=0                   YES, START MINUTES WAKEUPS
833  667          620 LC     6              C= 60.00 SECONDS
834  670          173 GOTO   LSWK30 ( 707)
835  671 LSWK25   370 RDSTS                C= HARDWARE STATUS
836  672         1474 RCR    1              C.S= DIGIT 0 OF HW STATUS
837  673          776 C=C+C  S              TIMER/STOPWATCH ALARM ?
838  674          317 GOC    LSWK85 ( 725) YES
839  675         1376 ? C#0  S              ALARM A ?
840  676          277 GOC    LSWK85 ( 725) YES
841  677          540 ?LLD                  LOW BATTERY ?
842  700            1 GOLC   CLKOFF         YES, TURN OFF
842  701            3                       *TIMER ROM:  TM1, @1117
843  702         1714 CHK KB                KEY DOWN ?
844  703         1513 GONC   LSWK20 ( 654) NO
845  704 LSWK6J    73 GOTO   LSWK60 ( 713) END CLOCK MODE
846
847  705 LSWK28   116 C=0
848  706         1066 C=C+1  XS             C= 1.00 SECONDS
849  707 LSWK30   550 WSINT                 WRITE & START INTVL TIMER
850  710           53 GOTO   LSWK67 ( 715)
851
```

```
   852  711 LSWK58    1 GOSUB  ENTMRS        ENABLE TIMER, SAVE S0-S7
   852  712           0                      *TIMER ROM:  TM0, @0341
   853
   854  713 LSWK60  104 S4=    0             CLEAR CLOCK DISPLAY BIT
   855  714 LSWK65  750 STPINT               STOP INTERVAL TIMER
   856  715 LSWK67    4 S3=    0             CLR SETTING TO CLK MODE BIT
   857  716        1730 CST EX
   858  717        1650 PT=B
   859  720         450 WRSCR                UPDATE SOFTWARE STATUS
   860  721 LSWK70    1 GOSUB  ENTMR         ENABLE TIMER, DIS RAM, PT=A
   860  722           0                      *TIMER ROM:  TM0, @0342
   861
   862                 ENTRY  LSWK80
   863  723 LSWK80    1 GOSUB  CLRAL0        CLR INT TIMER/GARBAGE ALMS
   863  724           0                      *TIMER ROM:  TM2, @1547
   864  725 LSWK85 1610 S0=    1             LEAVE DISPLAY ON
   865  726        1004 S2=    0             CLEAR "RUN LABEL" BIT
   866  727         504 S6=    0
   867
   868                 ENTRY  LSWK90
   869  730 LSWK90 1410 S1=    1             LEAVE DISPLAY ON
   870  731        1204 S7=    0             NOT PWOFF WITH NO KEY
   871
   872
*********************************************************************
* ALM000= ALARM WAKEUP ROUTINE
*
* IN:  S0=  1   (0)  (NOT) CALLED FROM IOSERV ENTRY POINT [LIGHT SLEEP]
*      S1=  1   (0)  TURN DISPLAY ON (OFF) AT EXIT
*      S2=  1   (0)  DO (NOT) RUN OLDEST LABEL ALARM = "RUN LABEL" BIT
*      S6=  1   (0)  DO (NOT) SET MESSAGE FLAG
*      S7=  1   (0)  CALLED (NOT CALLED) FROM PWOFF
* OUT: DOES  NOT RETURN, GOES TO RMCK10 !!!!!!!!!
*      !!! MUST PRESERVE M[10:3]= ROMCHK STUFF !!!
*
* INTERNAL USE:
*      S3            SCRATCH
*      S4=  1   (0)  (NOT) TIMER ALARM
*      S5=  1   (0)  THE DISPLAY HAS (NOT) BEEN CHANGED
*      S8,S9         SCRATCH
*
*
   891  732 ALM000  104 S4=    0             NOT TIMER ALARM
   892  733         204 S5=    0             DISPLAY NOT CHANGED YET
   893  734        1140 SETHEX
   894  735           1 GOSUB  ENTMR         SAVE SUB LVL (PARTIAL KEY)
   894  736           0                      *TIMER ROM:  TM0, @0342
   895  737         370 RDSTS                READ HARDWARE STATUS
   896  740        1730 CST EX
   897  741        1614 ?S0=1                MAIN CLOCK ALARM ?
   898  742          73 GONC   ALM032 ( 751) NO
   899  743        1730 CST EX
   900  744         460 LDI                  LOAD LOW 12 BITS OF C WITH
   901  745          70 CON    @70           CLEAR ALARM A, DTZ A, ALM B
   902  746         350 WRSTS                CLEAR ALARM A
   903  747           1 GOLONG ALM185        GO TO START OF ALARM BUFFER
   903  750           2                      *TIMER ROM:  TM3, @1477
   904
   905  751 ALM032   14 ?S3=1                TIMER COUNTED THROUGH 0 ?
   906  752          47 GOC    ALM033 ( 756) YES
```

```
   907  753         1730 CST EX
   908  754            1 GOLONG ALM230         DISPLAY ALARM MESSAGE
   908  755            2                       *TIMER ROM:  TM3, @1565
   909  756 ALM033 1530 ST=C
   910  757          110 S4=     1             REMEMBER TIMER ALARM
   911  760          460 LDI                   LOAD LOW 12 BITS OF C WITH
   912  761           61 CON     @61           CLR TMR ALARM, DTZ A, ALM B
   913  762          350 WRSTS
   914  763            1 GOLONG ALM135         DISPLAY ALARM CODE
   914  764            2                       *TIMER ROM:  TM3, @1235
   915
*************************************************************************
* GTALBL - GET THE ALPHA LABEL ALARM MESSAGE                2-24-81 RSW
*
* IN:     NOTHING
* ASSUME: M.X= ALARM ADDRESS
*         HEXMODE, PERIPHERALS DISABLED
*
* OUT:    JUMP TO ALM116 IF THE ALARM IS NOT A LABEL ALARM
*         JUMP TO ALM045 IF THE  ALARM IS A LABEL ALARM WITH :
*         C= FIRST 7 CHARACTERS OF THE ALARM MESSAGE
*         (THE MESSAGE IN C-REG IS READ FROM RIGHT TO LEFT AND IS RIGHT-
*          JUSTIFIED [WITH ZEROS IN UNUSED CHARACTER POSITIONS], WHICH
*          IS THE FORMAT REQUIRED BY THE ALPHA SEARCH ROUTINE)
*         P SELECTED, Q= 13
*
* USES:   A,C, P,Q, DADD
*         (NO ST, +0 SUB LEVELS, NO PFAD, NO TIMER CHIP ACCESS)
*
   934                  ENTRY  GTALBL
   935  765 GTALBL  630 C=M                   GET ALARM ADDRESS
   936  766          406 A=C     X
   937  767         1160 DADD=C
   938  770           70 C=DATA                LOAD THE ALARM
   939  771         1634 PT=     0
   940  772         1142 C=C-1   PT            DOES ALARM HAVE MESSAGE ?
   941  773          317 GOC     GALB28 (1024) NO, NOT A LABEL ALARM
   942  774         1366 ? C#0   XS            HAS RESET INTERVAL ?
   943  775           23 GONC    GALB05 ( 777) NO
   944  776          546 A=A+1   X
   945  777 GALB05   546 A=A+1   X             POINT TO 1ST MESSAGE REG
   946 1000         1474 RCR     1
   947 1001          436 A=C     S             A.S= MESSAGE LENGTH - 1
   948 1002          246 C=A     X
   948 1003          406                       (INSERTED BY ASSEMBLER)
   949 1004         1160 DADD=C
   950 1005           70 C=DATA
   951 1006          256 AC EX                 A= 1ST MSG REG, C= LEN&ADDR
   952 1007         1176 C=C-1   S             MESSAGE MORE THAN 1 REG ?
   953 1010           33 GONC    GALB10 (1013) YES
   954 1011          116 C=0                   MESSAGE ONLY ONE REG LONG
   955 1012           43 GOTO    GALB20 (1016)
   956 1013 GALB10 1046 C=C+1   X
   957 1014         1160 DADD=C
   958 1015           70 C=DATA                LOAD 2ND MESSAGE REGISTER
   959 1016 GALB20   256 AC EX                C= 1ST MSG REG, A= 2ND MREG
   960 1017          340 SEL Q
   961 1020         1334 PT=     13
   962 1021          240 SEL P
   963 1022         1534 PT=     12
```

```
  964 1023         1356 ? C#0                 WHOLE REGISTER= NULLS ?
  965 1024 GALB28     1 GOLNC  ALM116         YES, AVOID LOCKING UP
  965 1025            2                       *TIMER ROM:  TM3, @1227
  966 1026 GALB30  1362 ? C#0   PQ            FOUND FIRST CHARACTER ?
  967 1027           67 GOC     GALB40 (1035) YES
  968 1030          262 AC EX   PQ
  969 1031         1756 A SL
  970 1032         1756 A SL
  971 1033         1574 RCR    12            MOVE CHAR/NULL TO C[1:0]
  972 1034         1723 GOTO   GALB30 (1026)
  973 1035 GALB40  1574 RCR    12            C[1:0] = 1ST CHAR OF MSG
  974 1036          406 A=C    X
  975 1037          460 LDI                  LOAD LOW 12 BITS OF C WITH
  976 1040          136 CON2   5      14      ASCII CODE FOR UP-ARROW
  977 1041          246 AC EX  X
  978 1042         1434 PT=    1
  979 1043         1552 ? A#C  WPT           IS 1ST CHAR AN UP-ARROW ?
  980 1044         1607 GOC    GALB28 (1024) NO, NOT A LABEL ALARM
  981 1045         1074 RCR    2
  982 1046         1534 PT=    12
  983 1047          262 AC EX  PQ            REPLACE "^" WITH NEXT CHAR
  984 1050         1574 RCR    12
  985 1051          416 A=C                  REVERSE THE STRING ORDER
  986 1052          116 C=0
  987 1053 GALB50   262 AC EX  PQ
  988 1054         1756 A SL
  989 1055         1756 A SL
  990 1056         1522 ? A#0  PQ            ALL MOVED ?
  991 1057           33 GONC   GALB55 (1062) YES
  992 1060         1074 RCR    2
  993 1061         1723 GOTO   GALB50 (1053)
  994 1062 GALB55  1434 PT=    1            RIGHT-JUSTIFY STRING IN C
  995 1063         1356 ? C#0                ANY LABEL GIVEN ?
  996 1064            1 GSUBC  RTJLBL        YES, RT-JUSTIFY ALPHA LABEL
  996 1065            1                      *MAINFRAME:  CN5, @0311
  997                                        IN & ASSUME: C= ALPHA LABEL
  998                                         !!! C MUST BE NON-ZERO !!!
  999                                        OUT: C= RIGHT-JUSTIFIED
 1000                                             ALPHA LABEL, PT= 1
 1001                                        USES: C, ACTIVE PT ONLY
 1002
 1003
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
*
* ALM045 -- THE ALARM IS A LABEL ALARM                    3-11-81 RSW
*
* IN: HEXMODE
*     C= ALPHA LABEL IN RIGHT-TO-LEFT ORDER AND RIGHT-JUSTIFIED
*        I.E., FOR "ABC" C= 0000CBA.
*     S13= 1 (0)  RUNNING (NOT RUNNING)
*     S10= 1 (0)  CURRENT PC IN ROM (RAM)
*     M.X= VALID ALARM ADDRESS (OF THE LABEL ALARM)
*
 1015
 1016 1066 ALM045   416 A=C                  A= ALPHA LABEL
 1017 1067            1 GOSUB  TMRSTS        S0-S7=SW ST, C[1:0]=TEMP ST
 1017 1070            0                      *TIMER ROM:  TM3, @0244
 1018 1071         1004 S2=    0             CLEAR "RUN LABEL" BIT
 1019 1072          346 B=C    X             B[1:0]= ALARM TEMP STATUS
 1019 1073          306                      (INSERTED BY ASSEMBLER)
```

```
  1020 1074         1434 PT=    1
  1021 1075          752 C=C+C  WPT            POWOFF? (?S7=1)
  1022 1076          113 GONC   ALM050 (1107) NO
  1023 1077         1340 DISOFF                TURN DISPLAY OFF
  1024 1100         1010 S2=    1              SET "RUN LABEL" BIT
  1025 1101 ALM047     1 GOSUB  TIME+1         C= SLIGHTLY FUTURE TIME
  1025 1102            0                       *TIMER ROM:  TM3, @0474
  1026 1103          250 WRALM
  1027 1104         1350 ENALM                 ENABLE ALARM
  1028 1105            1 GOLONG RMRT02         CLEAR "DSWKNO" BIT & RTN
  1028 1106            2                       *TIMER ROM:  TM3, @1625
  1029
  1030 1107 ALM050  1474 RCR    1              C[S]= ALM TEMP STAT DIGIT 0
  1031 1110          460 LDI                   LOAD LOW 12 BITS OF C WITH
  1032 1111          136 CON2   5      14      CODE FOR UP ARROW
  1033 1112         1552 ? A#C  WPT            2ND UP ARROW?
  1034 1113          107 GOC    ALM052 (1123) NO
  1035 1114         1616 A SR
  1036 1115         1616 A SR                  ELIMINATE 2ND UP ARROW
  1037 1116          614 ?S11=1                STACK LIFT FLAG SET?
  1038 1117          277 GOC    ALM060 (1146) YES, OK TO RUN
  1039 1120         1314 ?S13=1                NO, RUNNING?
  1040 1121         1607 GOC    ALM047 (1101) YES, WAIT FOR S11=1
  1041 1122          243 GOTO   ALM060 (1146)
  1042 1123 ALM052   776 C=C+C  S              RUN LABEL ALARM?
  1043 1124          227 GOC    ALM060 (1146) YES
  1044 1125          306 C=B    X              C.X= ALARM TEMP STATUS
  1045 1126         1614 ?S0=1                 DEEP SLEEP WAKEUP?
  1046 1127          177 GOC    ALM060 (1146) YES, RUN LABEL ALARM
  1047 1130          114 ?S4=1                 CLOCK DISPLAY?
  1048 1131          157 GOC    ALM060 (1146) YES, RUN LABEL ALARM
  1049 1132         1730 CST EX                ST= ALARM TEMP STATUS
  1050 1133          104 S4=    0              NOT A TIMER ALARM
  1051 1134         1314 ?S13=1                RUNNING A PROGRAM?
  1052 1135          723 GONC   ALM116 (1227) NO
  1053 1136            1 GOSUB  GETMXP         YES, C= ALARM REG, PT=1
  1053 1137            0                       *TIMER ROM:  TM1, @0531
  1054 1140         1720 LC     15             MARK THE ALARM AS ACTIVATED
  1055 1141         1360 DATA=C
  1056 1142            1 GOSUB  BEEP2          YES, BEEP TWICE
  1056 1143            0                       *TIMER ROM:  TM3, @0176
  1057 1144 ALM055     1 GOLONG ALM200         GO TO NEXT ALARM
  1057 1145            2                       *TIMER ROM:  TM3, @1503
  1058
*****************************************************************
* ALM060 -- RUN A LABEL ALARM                          3-12-81 RSW
*
* IN:  S0-S7= TIMER SOFTWARE STATUS
*      TIMER PT=B, HEXMODE
*      TIMER CHIP ENABLED, RAM DISABLED
*      A= ALPHA LABEL IN RIGHT TO LEFT ORDER AND RIGHT JUSTIFIED.
*        I.E., FOR "ABC", A= 0000CBA.
*      B.X= ALARM TEMPORARY STATUS
*      S13= 1 (0)   RUNNING (NOT RUNNING)
*      S10= 1 (0)   CURRENT PC IN ROM (RAM)
*      M.X= VALID ALARM ADDRESS (OF THE LABEL ALARM)
*
  1072 1146 ALM060   306 C=B    X              C[1:0]= ALARM TEMP STATUS
  1073 1147            1 GOSUB  CLRALD         CLR DSWKNO BIT, END CLK MDE
  1073 1150            0                       *TIMER ROM:  TM2, @1450
```

```
  1074 1151           1 GOSUB  ENCP00        ENA CHIP 0, DIS PERIPHERALS
  1074 1152           0                      *MAINFRAME:  CN2, @0522
  1075                                       IN & ASSUME: NOTHING
  1076                                       OUT:  CHIP 0 ENABLED,
  1077                                             C.X= 0,
  1078                                             PERIPHERALS DISABLED
  1079                                       USES: C.X ONLY
  1080 1153        1014 ?S2=1                CALLED FROM ALMNOW?
  1081 1154          67 GOC    ALM063 (1162) YES, HONOR SINGLE-STEP FLAG
*  S2 CAN ALSO BE SET WHEN RUNNING PAST-DUE LABEL ALARM AFTER POWER OFF,
*  BUT THE DEEP SLEEP WAKEUP WILL HAVE CLEARED THE SST FLAG ALREADY.
*
  1085 1155        1670 C=REGN 14
  1086 1156        1730 CST EX               PUT UP STATUS SET 0
  1087 1157         104 S4=    0             CLEAR THE SINGLE-STEP FLAG
  1088 1160        1730 CST EX
  1089 1161        1650 REGN=C 14            UPDATE REGISTER 14
  1090 1162 ALM063  256 AC EX                C= ALPHA LABEL
*****************************************************************
*                                                        3-2-81 RSW
*
* IN:  HEXMODE, CHIP 0 ENABLED, PERIPHERALS DISABLED
*      C= ALPHA LABEL IN RIGHT TO LEFT ORDER AND RIGHT JUSTIFIED,
*        I.E., FOR "ABC", C= 0000CBA.
*      S13= 1 (0)   RUNNING (NOT RUNNING)
*      S10= 1 (0)   CURRENT PC IN ROM (RAM)
*      M.X= VALID ALARM ADDRESS (OF THE LABEL ALARM)
*
  1101
  1102 1163        1150 REGN=C 9             REG 9= LABEL
  1103 1164           1 GOSUB  ACKALM        ACKNOWLEDGE ALARM
  1103 1165           0                      *TIMER ROM:  TM2, @1220
  1104 1166          73 GOTO   ALM070 (1175) (P+1) ALM CATALOG EMPTY OR
  1105                                             NO HIGHER ADDR ALARMS
  1106 1167         114 ?S4=1                (P+2) CALLED FROM "ALMNOW"?
  1107 1170          57 GOC    ALM070 (1175) YES, ACTIVATE NO MORE ALMS
  1108 1171           1 GOSUB  TIME+1         C= SLIGHTLY FUTURE TIME
  1108 1172           0                      *TIMER ROM:  TM3, @0474
  1109 1173         250 WRALM                SET HARDWARE ALARM
  1110 1174        1350 ENALM                ENABLE ALARM
* !!! MUST EITHER SET THIS ALARM OR SET THE NEXT FUTURE ALARM SINCE
*     "ACKALM" DOES NOT SET THE NEXT FUTURE ALARM.
  1113
*
* IN: HEXMODE, R9= ALPHA LABEL
*
  1117 1175 ALM070 1314 ?S13=1               RUNNING?
  1118 1176           1 GSUBNC CLRFLG        NO, CLEAR MISC FLAGS
  1118 1177           0                      *TIMER ROM:  TM3, @0446
  1119 1200           1 GOSUB  LDSST0        PUT UP STATUS SET 0
  1119 1201           0                      *MAINFRAME:  CN1, @1627
  1120                                       IN & ASSUME: NOTHING
  1121                                       OUT: S0-S7= STATUS SET 0,
  1122                                         C= REG 14, CHIP 0 ENABLED
  1123                                         PERIPHERALS DISABLED
  1124                                       USES: C, S0-S7, DADD, PFAD
  1125                                         (NO PT, +0 SUB LEVELS)
  1126 1202          14 ?S3=1                IN PROGRAM MODE?
  1127 1203          63 GONC   ALM080 (1211) NO
  1128 1204           4 S3=    0             YES, EXIT PROGRAM MODE
```

```
  1129 1205         1630 C=ST
  1130 1206         1650 REGN=C 14
  1131 1207            1 GOSUB  DECMPL        DECOMPILE PROGRAM MEMORY
  1131 1210            0                      *MAINFRAME: CN11, @1302
  1132                                        IN: PERIPHERALS DISABLED
  1133                                        ASSUME: HEXMODE
  1134                                        OUT: CHIP 0 ENA, SST 0 UP
  1135                                             C=REG 14, PERIPH DISA
  1136                                        USES: A,B,C,N, ACTIVE PT
  1137                                          (P AND Q ?) S0-S9, DADD,
  1138                                          (PFAD?) SUPPOSEDLY +3
  1139                                          SUB LEVELS
  1140 1211 ALM080   674 RCR    11
  1141 1212         1530 ST=C
  1142 1213         1604 S0=     0             CLEAR USER FLAG 11
  1143 1214         1630 C=ST
  1144 1215           74 RCR    3
  1145 1216         1650 REGN=C 14
  1146 1217         1170 C=REGN 9
  1147 1220          530 M=C                  M= ALPHA LABEL
  1148 1221         1356 ? C#0                ANY LABEL GIVEN?
  1149 1222            1 GOLNC  RUN           NO, RUN AT CURRENT PC
  1149 1223            2                      *MAINFRAME:  CN1, @1702
  1150                                        IN: CHIP 0 ENA, PERIPH DISA
  1151                                            PC POSITIONED PROPERLY
  1152                                            HEXMODE
  1153
  1154 1224         1104 S9=     0
  1155 1225            1 GOLONG AXEQ          FALL INTO MAINFM ALPHA XEQ
  1155 1226            2                      *MAINFRAME:  CN4, @0265
  1156                                        IN: S10= 1/0 PC IN ROM/RAM
  1157                                            S9=  1/0 ALPHA SEARCH
  1158                                                 HAS/NOT BEEN DONE
  1159                                            IF S9=1 : M[3:0] = ADDR
  1160                                            IF S9=0 : R9= M= ALPHA
  1161                                             LABEL IN R-L ORDER
  1162                                             (FOR ABC, M= 0000CBA)
  1163                                          S13 (RUNNING FLAG) IN
  1164                                          PROPER STATE, CHIP 0
  1165                                          ENABLED, PERIPHERALS
  1166                                          DISABLED, HEXMODE
*
* IN: HEXMODE, PERIPHERALS DISABLED
*     ALARM TEMPORARY STATUS SET UP
*     M.X= !VALID! CURRENT ALARM ADDRESS
*
  1172
  1173                   ENTRY  ALM116
  1174 1227 ALM116 1014 ?S2=1                RUN LABEL ALARM?
  1175 1230         1147
  1176 1231            1 GOSUB  GETMXP        C=ALARM TIME & INFO, PT=1
  1176 1232            0                      *TIMER ROM:  TM1, @0531
  1177 1233         1720 LC     15            MARK THE ALARM
  1178 1234         1360 DATA=C
  1179
  1180                   ENTRY  ALM135
  1181 1235 ALM135   630 C=M
  1182 1236          210 S5=     1             REMEMBER THAT DISP CHANGED
  1183 1237          674 RCR    11
  1184 1240         1630 C=ST
```

```
  1185 1241          74 RCR    3
  1186 1242         530 M=C                  M[12:11]= ALARM TEMP STATUS
*************************************************************************
*                                                              2-25-81 RSW
*             ALARM DISPLAY CODE
*
*    THIS CODE DOES THE BEEPING & FLASHING WHEN AN ALARM GOES OFF.
* IF THE USER PUSHES A KEY, S6 IS SET = 0 (DON'T SET MSG FLAG) SINCE
* THE ALARM MESSAGE SHOULD GO AWAY IF THE ALARM IS ACKNOWLEDGED.  IF
* A TIMEOUT OCCURS BEFORE A KEY IS PUSHED, THE CLOCK DISPLAY IS
* DISABLED (INTERVAL TIMER STOPPED & CLOCK DISPLAY BITS S3=S4= 0)
* AND S6=1 SO THE ALARM MESSAGE WILL REMAIN IN THE DISPLAY.
*
* IN: M[12:11]= ALARM TEMPORARY STATUS,
*      WHERE S4= 1 (0) MEANS ALARM IS (NOT) A TIMER/STOPWATCH ALARM.
* ASSUME: HEXMODE
*         IF S4= 0, THEN M.X= VALID ALARM ADDRESS.
* OUT:    PERIPHERALS DISABLED, ALARM TEMPORARY STATUS RESTORED
*       IF NO KEY:  ALARM TEMPORARY STATUS BIT S6=1 TO SET MESSAGE FLAG
*                   CLEAR S2-S4 OF TIMER SOFTWARE STATUS
*                   STOP INTERVAL TIMER & CLEAR ITS ALARM
*                   JUMP TO ALM160
*       IF KEY DOWN: ALARM TEMP STATUS BIT S6=0 (DON'T SET MSG FLAG)
*                   G=KEYCODE
* USES: A,B,C,G,M[S],N,R8[13:6], P,Q, S3,S6 OF ALARM TEMP STATUS, S8,S9,
*       (MAYBE S13=1), +3 SUB LEVELS, DADD, PFAD, ARITH MODE, TIMER PT,
*       IF NO KEY DOWN, CLEARS S2-S4 OF TIMER ST
*
*
*
  1215 1243           1 GOSUB  CLRFLG        CLEAR MISC FLAGS
  1215 1244           0                      *TIMER ROM:  TM3, @0446
  1216 1245        1340 DISOFF
  1217 1246        1440 DISTOG               BE SURE DISPLAY IS ON
  1218 1247           1 GOSUB  BEEP2         BEEP TWICE
  1218 1250           0                      *TIMER ROM:  TM3, @0176
  1219 1251         630 C=M
  1220 1252         674 RCR    11
  1221 1253        1530 ST=C
  1222 1254         114 ?S4=1                TIMER ALARM?
  1223 1255         223 GONC   DSPA10 (1277) NO
  1224 1256           1 GOSUB  CLLCDE        ENA & CLR DISP, DISABLE RAM
  1224 1257           0                      *MAINFRAME: CN11, @0360
  1225                                       IN & ASSUME: NOTHING
  1226                                       OUT: LCD ENABLED & CLEARED
  1227                                       USES: C[11:0], ACTIVE PT,
  1228                                           DADD, PFAD, NO ST, +0
  1229                                           SUB LEVELS, NO ARITH
  1230 1260           1 GOSUB  MESSL         SHIFT MSG LEFT INTO DISPLAY
  1230 1261           0                      *MAINFRAME:  CN1, @1757
  1231                                       IN: DISPLAY CLEARED
  1232                                       ASSUME: DISPLAY ENABLED
  1233                                               RAM DISABLED
  1234                                               HEXMODE
  1235                                       OUT: MESSAGE TO DISPLAY
  1236                                       USES: C[6:0] ONLY
  1237 1262          24 CON    @24           T
  1238 1263          11 CON    @11           I
  1239 1264          15 CON    @15           M
  1240 1265           5 CON    @05           E
```

```
  1241 1266          22 CON    @22           R
  1242 1267          40 CON    @40               SPACE
  1243 1270           1 CON    @01           A
  1244 1271          14 CON    @14           L
  1245 1272           1 CON    @01           A
  1246 1273          22 CON    @22           R
  1247 1274          15 CON    @15           M
  1248 1275        1040 CON    @1040             TRAILING BLANK
  1249 1276         123 GOTO   DSPA20 (1310)
  1250
  1251 1277 DSPA10    1 GOSUB  DSAMS0        DISP 1ST 12 CHARS OF MSG
  1251 1300           0                      *TIMER ROM:  TM0, @0621
  1252 1301          23 GOTO   DSPA15 (1303) (P+1) NOT MSG/LABEL ALARM
  1253 1302          63 GOTO   DSPA20 (1310) (P+2) MSG/LABEL ALARM
  1254 1303 DSPA15    1 GOSUB  DSTMDA        DISPLAY TIME & DATE
  1254 1304           0                      *TIMER ROM:  TM3, @0060
  1255 1305         630 C=M
  1256 1306         674 RCR    11
  1257 1307        1530 ST=C                 RESTORE ALM TEMP STATUS
  1258 1310 DSPA20  106 C=0    X
  1259 1311        1760 PFAD=C               DISABLE DISPLAY
  1260 1312        1146 C=C-1  X
  1261 1313 DSPA21 1146 C=C-1  X
  1262 1314        1773 GONC   DSPA21 (1313) WAIT 1.3 SECONDS
  1263 1315           1 GOSUB  RSTKB         CLEAR KEYBOARD
  1263 1316           0                      *MAINFRAME:  CN0, @0230
  1264                                       IN: HEXMODE
  1265                                       ASSUME: HEXMODE
  1266                                       USES: C.X ONLY
  1267 1317         630 C=M
  1268 1320        1334 PT=    13
  1269 1321         420 LC     4             4 SEC FLASH BEFORE BEEPING
  1270 1322        1104 S9=    0             NOT BEEPING
  1271
*************************************************************************
* IN: HEXMODE
*     C.S= TIMER COUNTER, REST OF C= COPY OF M
*     S4= 1 (0)  ALARM IS (NOT) A TIMER/STOPWATCH ALARM
*         IF S4=1, THEN:  M.X= VALID ALARM ADDRESS
*     S9= 1 (0)  (NOT) BEEPING
*
  1279 1323 DSPA25 1340 DISOFF              TURN DISPLAY OFF
  1280 1324         530 M=C                 M.S= TIMEOUT COUNTER
  1281 1325         460 LDI                 LOAD LOW 12 BITS OF C WITH
  1282 1326         361 CON    241          WAIT 0.15 SEC W/DISPLAY OFF
  1283 1327 DSPA30 1714 CHK KB               KEY DOWN?
  1284 1330         377 GOC    DSPA80 (1367) YES, CHECK IT OUT
  1285 1331        1146 C=C-1  X
  1286 1332        1753 GONC   DSPA30 (1327)
  1287 1333        1440 DISTOG               TURN DISPLAY ON
  1288 1334         460 LDI                 LOAD LOW 12 BITS OF C WITH
  1289 1335        1446 CON    806
  1290 1336        1114 ?S9=1                BEEPING?
  1291 1337          53 GONC   DSPA40 (1344) NO, WAIT 0.5 SEC W/DISP ON
  1292 1340         410 S8=    1             CHECK KEYBOARD
  1293 1341           1 GOSUB  BEEP2K        BEEP TWICE
  1293 1342           0                      *TIMER ROM:  TM3, @0177
  1294 1343         106 C=0    X             DON'T WAIT, JUST CHK KYBD
  1295 1344 DSPA40 1714 CHK KB               KEY DOWN?
  1296 1345         237 GOC    DSPA81 (1370) YES, CHECK IT OUT
```

```
  1297 1346         1146 C=C-1  X
  1298 1347         1753 GONC   DSPA40 (1344)
  1299 1350          630 C=M
  1300 1351         1176 C=C-1  S              TIMEOUT?
  1301 1352         1513 GONC   DSPA25 (1323) NO
  1302 1353         1114 ?S9=1                 YES, BEEPING?
  1303 1354           37 GOC    DSPA60 (1367) YES
  1304 1355         1110 S9=    1              NO, BEEP
  1305 1356         1453 GOTO   DSPA25 (1323) C.S= F FROM PREV TIMEOUT
  1306                                          SO BEEP&FLASH FOR 15 SECS
  1307 1357 DSPA60   510 S6=    1              NO KEY DOWN, SET MSG FLAG
  1308 1360         1214 ?S7=1                 POWEROFF?
  1309 1361          627 GOC    ALM160 (1443) YES, DON'T STOP CLOCK MODE
  1310 1362            1 GOSUB  TMRSTS         READ TIMER HARDWARE STATUS
  1310 1363            0                       *TIMER ROM:  TM3, @0244
  1311 1364            1 GOSUB  CLRALS         CLEAR ANY INT TIMER ALARM
  1311 1365            0                       *TIMER ROM:  TM2, @1451
  1312                                          & STOP CLOCK MODE
  1313 1366          553 GOTO   ALM160 (1443)
*    !!!! THIS IS THE "NO KEY" EXIT FROM DSPALM !!!!
  1315
  1316 1367 DSPA80 1440 DISTOG               TURN DISPLAY ON
  1317 1370 DSPA81 1040 C=KEYS
  1318 1371          504 S6=    0            DON'T SET MSG FLAG,
  1319 1372           34 PT=    3             ALARM ACKNOWLEDGED
  1320 1373          130 G=C                 SAVE KEYCODE IN G
  1321 1374           74 RCR    3
  1322 1375         1434 PT=    1
  1323 1376          412 A=C    WPT          A= KEYCODE
  1324 1377         1374 RCR    13
  1325 1400          742 C=C+C  PT           OFF KEY?
  1326 1401          467 GOC    ALM169 (1447) YES
  1327 1402          460 LDI                 LOAD LOW 12 BITS OF C WITH
  1328 1403          303 CON2   12     3     BACK ARROW
  1329 1404         1552 ? A#C  WPT          BACK ARROW?
  1330 1405          223 GONC   DSPA97 (1427) YES
  1331 1406           56 B=0                 ASSUME NO MESSAGE
  1332 1407          114 ?S4=1               TIMER ALARM?
  1333 1410           47 GOC    DSPA85 (1414) YES
  1334 1411            1 GOSUB  DSAMS0        DISPLAY ALARM MESSAGE
  1334 1412            0                      *TIMER ROM:  TM0, @0621
  1335 1413            0 NOP                  (P+1) NO MESSAGE, B=0
  1336 1414 DSPA85     1 GOSUB  RSTKBT        (P+2) WAIT FOR KEY UP
  1336 1415            0                      *TIMER ROM:  TM1, @0503
  1337 1416            1 GOSUB  DSA2ND        DISP REST OF MSG (IF ANY)
  1337 1417            0                      *TIMER ROM:  TM0, @0731
  1338 1420            0 NOP                  (P+1)
  1339 1421          106 C=0    X            (P+2)
  1340 1422         1146 C=C-1  X
  1341 1423 DSPA95 1714 CHK KB               KEY DOWN?
  1342 1424         1447 GOC    DSPA81 (1370) YES
  1343 1425         1146 C=C-1  X
  1344 1426         1753 GONC   DSPA95 (1423)
  1345 1427 DSPA97     1 GOSUB  RSTKBT        CLEAR KEYBOARD
  1345 1430            0                      *TIMER ROM:  TM1, @0503
*    !!! THIS IS THE "KEY DOWN" EXIT FROM DSPALM !!!!
*          (EXCEPT THE "ON" KEY BRANCH TO ALM169)
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
  1349 1431          114 ?S4=1               WAS IT A TIMER ALARM?
  1350 1432          137 GOC    ALM168 (1445) YES
```

```
  1351 1433         1634 PT=    0
  1352 1434          230 C=G
  1353 1435          126 C=0    XS
  1354 1436          406 A=C    X              A.X= HARDWARE KEYCODE
  1355 1437          460 LDI                   LOAD LOW 12 BITS OF C WITH
  1356 1440          162 CON2   7      2       "STO" KEYCODE
  1357 1441         1546 ? A#C  X              STO KEY HIT?
  1358 1442          137 GOC    ALM170 (1455) NO
  1359 1443 ALM160   114 ?S4=1                 TIMER ALARM?
  1360 1444          373 GONC   ALM200 (1503) NO
  1361 1445 ALM168     1 GOLONG ALM230         YES, TIMER ALARM
  1361 1446            2                       *TIMER ROM:  TM3, @1565
  1362
  1363 1447 ALM169  1214 ?S7=1                 POWEROFF PATH?
  1364 1450         1573 GONC   DSPA97 (1427) NO
  1365 1451          110 S4=    1              DON'T DISP ANY MORE ALARMS
  1366 1452            1 GOSUB  ALM200         SET NEW HARDWARE ALARM
  1366 1453            0                       *TIMER ROM:  TM3, @1503
  1367                                          (IF NO ALARM IS GOING OFF)
  1368 1454         1713 GOTO   ALM168 (1445) RETURN TO ROMCHK
  1369
  1370 1455 ALM170     1 GOSUB  ACKALM         RESET ALM IF HAS INTERVAL
  1370 1456            0                       *TIMER ROM:  TM2, @1220
*
*    !! NOTE: ACKALM USES 4 SUB LEVELS, SO MUST NOT DISPLAY ALARMS IF
*       IT IS DESIRED TO EXIT THIS ALARM CODE WITH A RTN (S4=1) !!!!
*
  1375 1457          273 GOTO   ALM210 (1506) (P+1) NO MORE ALARMS, OR NO
  1376                                              HIGHER ADDRESSED ALMS
  1377 1460          630 C=M                  (P=2) C.X= CURR ALARM ADDR
  1378 1461          343 GOTO   ALM220 (1515)
  1379
*******************************************************************************
* ALMNOW
*******************************************************************************
  1383 1462          227 CON    @227          W
  1384 1463           17 CON    @17           O
  1385 1464           16 CON    @16           N
  1386 1465           15 CON    @15           M
  1387 1466           14 CON    @14           L
  1388 1467            1 CON    @01           A
  1389                   ENTRY  ALMNOW
  1390                   ENTRY  NXTALM
  1391 1470 ALMNOW     1 GOSUB  INITMR         INIT TIMER IF NECESSARY
  1391 1471            0                       *TIMER ROM:  TM0, @1524
  1392 1472          610 S11=   1              SET STACK LIFT FLAG
  1393 1473         1704 CLR ST               NOT POWOFF WITH NO KEY DOWN
  1394 1474         1010 S2=    1              SET "RUN LABEL" BIT
  1395 1475 NXTALM   110 S4=    1              RETURN AND DON'T DISPLAY
  1396 1476         1140 SETHEX
*
* .    .     .     .     .     .     .     .     .     .     .     .
*
  1400
  1401                   ENTRY  ALM185
  1402 1477 ALM185     1 GOSUB  SRHBUF         NO, FIND START OF ALM STACK
  1402 1500            0                       *TIMER ROM:  TM2, @1141
  1403 1501          113 GOTO   ALM215 (1512) (P+1) BUFFER FOUND
  1404 1502           43 GOTO   ALM210 (1506) (P+2) BUFFER NOT FOUND
  1405
```

```
  1406                    ENTRY  ALM200
  1407 1503 ALM200    1 GOSUB  ALMSST          SINGLE-STEP TO NEXT ALARM
  1407 1504           0                       *TIMER ROM:  TM2, @1205
  1408 1505         103 GOTO   ALM220 (1515) (P+1) C.X=M.X=NEXT ALM ADDR
  1409                                        (P+2)
  1410 1506 ALM210    1 GOSUB  ENTMR          ENABLE TIMER CHIP, PT=A
  1410 1507           0                       *TIMER ROM:  TM0, @0342
  1411 1510          16 A=0                   SET ALM TO 0, NEVER GO OFF
  1412 1511         333 GOTO   ALM225 (1544)
  1413
  1414 1512 ALM215  546 A=A+1  X              A.X= ADDRESS OF FIRST ALARM
  1415                LEGAL                   (CLEAR THE CARRY FLAG)
  1416 1513           1 GOSUB  NEWM.X         C.X= M.X= FIRST ALARM ADDR
  1416 1514           0                       *TIMER ROM:  TM2, @1214
  1417
  1418 1515 ALM220 1160 DADD=C
  1419 1516          70 C=DATA
  1420 1517         346 BC EX  X              B.X= ALARM INFORMATION
  1421 1520         106 C=0    X
  1422 1521        1074 RCR    2
  1423 1522         416 A=C                   A= ALM TIME= 00SSSSSSSSSS00
  1424 1523           1 GOSUB  ENTMR          ENABLE TIMER, DIS RAM, PT=A
  1424 1524           0                       *TIMER ROM:  TM0, @0342
*   SAVE SUB LEVELS FOR PARTIAL KEY SEQUENCES !!!!!!!!
  1426 1525           1 GOSUB  FTIME          C= SLIGHTLY FUTURE TIME
  1426 1526           0                       *TIMER ROM:  TM3, @0476
  1427 1527        1416 ? A<C                 IS ALARM REALLY PAST DUE?
  1428 1530         143 GONC   ALM225 (1544) NO, SET FUTURE ALARM
  1429 1531        1014 ?S2=1                 RUN LABEL ALARM?
  1430 1532         107 GOC    ALM222 (1542) YES
  1431 1533         114 ?S4=1                 CALLED FROM XYZALM?
  1432 1534        1477 GOC    ALM200 (1503) YES, DON'T DISPLAY ALARM
  1433 1535        1214 ?S7=1                 POWEROFF?
  1434 1536          47 GOC    ALM222 (1542) YES, DISPLAY PAST DUE ALARM
  1435 1537        1434 PT=    1
  1436 1540        1302 ? B#0  PT             ALARM ALREADY DISPLAYED?
  1437 1541        1427 GOC    ALM200 (1503) YES, DON'T DISPLAY IT AGAIN
  1438 1542 ALM222    1 GOLONG GTALBL         GET ALPHA LABEL ALARM MESG
  1438 1543           2                       *TIMER ROM:  TM3, @0765
*****************************************************************
*         SET ALARM IN TIMER CHIP                        3-19-81 RSW
*
  1442
* IN:  A= 00SSSSSSSSSSC0 = ALARM TIME
*      TIMER CHIP ENABLED, RAM DISABLED, TIMER PT=A, HEXMODE
*      TEMPORARY ALARM STATUS SET UP (S0,S2,S4,S6 SPECIFICALLY)
*
  1447 1544 ALM225  114 ?S4=1                 SUPPRESSING ALARM DISPLAY?
  1448 1545         103 GONC   ALM226 (1555) NO, DISPLAYING ALARMS
  1449 1546         370 RDSTS                C= HARDWARE STATUS
*
*    NOTE: AN ALARM CAN'T GO OFF AT THIS POINT AND BE MISSED.  IT WOULD
*          HAVE TO HAVE BEEN JUDGED TO BE PAST DUE IN THE PREVIOUS TEST
*          (BUT REALLY WAS 1/100 SEC IN THE FUTURE) AND IT TAKES MORE
*          THAN 1/100 SEC TO SINGLE-STEP TO THE NEXT ALARM AND GET HERE
*          TO SET IT.
*
  1457 1547        1730 CST EX
  1458 1550        1614 ?S0=1                 MAIN CLOCK ALARM?
  1459 1551          33 GONC   AL225B (1554) NO
```

```
1460 1552         1530 ST=C
1461 1553         1740 RTN
1462 1554 AL225B  1530 ST=C                 RESTORE STATUS
1463 1555 ALM226   256 AC EX                C= ALARM
1464 1556          250 WRALM                STORE ALARM
1465 1557         1350 ENALM                ENABLE THE ALARM
1466 1560 ALM227   460 LDI                  LOAD LOW 12 BITS OF C WITH
1467 1561           70 CON    @70           CLEAR ALL ALARMS EXCEPT
1468 1562          350 WRSTS                 TIMER, INT TIMER, & PUS
1469 1563          114 ?S4=1                CALLED FROM XYZALM?
1470 1564         1540 RTN C                YES, DONE
1471
1472
*                                                              3-19-81 RSW
* IN:  HEXMODE, ALARM TEMPORARY STATUS SET UP (S0,S2,S6 SPECIFICALLY)
*      M[10:3]= COPY OF C[10:3] AT ENTRY FROM ROMCHK
*
1477                    ENTRY  ALM230
1478 1565 ALM230  1340 DISOFF               DISPLAY OFF
1479 1566         1414 ?S1=1                TURN DISPLAY ON?
1480 1567           23 GONC   ALM232 (1571)
1481 1570         1440 DISTOG               YES, DISPLAY ON
1482 1571 ALM232     1 GOSUB  ENCP00        ENA CHIP 0, DIS PERIPHERALS
1482 1572            0                      *MAINFRAME:  CN2, @0522
1483                                        IN: NOTHING
1484                                        ASSUME: NOTHING
1485                                        OUT: C.X= 0
1486                                        USES: C.X, DADD, PFAD ONLY
1487 1573          514 ?S6=1                NO, DISPLAYING ALARM MSG?
1488 1574          247 GOC    ALM270 (1620) YES, SET MESSAGE FLAG
1489 1575         1314 ?S13=1               CALLED IN RUNNING PROGRAM?
1490 1576            1 GOLC   RUN           PUT GOOSE BACK IN DISPLAY
1490 1577            3                      *MAINFRAME:  CN1, @1702
1491                                        !!!! MESSAGE FLAG MUST BE
1492                                             CLEARED !!!!
1493                                        IN: CHIP 0 ENABLED, HEXMODE
1494                                            PC POSITIONED PROPERLY
1495                                        OUT: GOLONG TO NFRPU
1496 1600         1614 ?S0=1                CALLED FROM LIGHT SLEEP?
1497 1601          213 GONC   RMRT00 (1622) NO
1498 1602            1 GOSUB  TMRSTS        YES, PUT UP SOFTWARE STATUS
1498 1603            0                      *TIMER ROM:  TM3, @0244
1499 1604          114 ?S4=1                DO CLOCK DISPLAY?
1500 1605           67 GOC    ALM260 (1613) YES
1501 1606         1730 CST EX
1502 1607          214 ?S5=1                HAS DISPLAY BEEN CHANGED?
1503 1610          123 GONC   RMRT00 (1622) NO
1504 1611            1 GOLONG NFRC          YES, REFRESH THE DISPLAY
1504 1612            2                      *MAINFRAME:  CN0, @0361
1505                                        IN: PERIPHERALS DISABLED
1506
1507
1508 1613 ALM260   540 ?LLD                 LOW BATTERY ?
1509 1614            1 GOLC   CLKOFF        YES, TURN OFF
1509 1615            3                      *TIMER ROM:  TM1, @1117
1510 1616            1 GOSUB  CLKDSP        PUT UP CLOCK DISPLAY
1510 1617            0                      *TIMER ROM:  TM1, @0462
1511 1620 ALM270     1 GOSUB  STMSGF        SET MESSAGE FLAG
1511 1621            0                      *MAINFRAME:  CN0, @1576
1512                                        IN & ASSUME: PERIPHERALS
```

```
  1513                                                    DISABLED
  1514                                          OUT: MSG FLAG SET, SST 0 UP
  1515                                            CHIP 0 ENABLED, C= REG 14
  1516                                          USES: C, S0-S7 ONLY
  1517
*
* IN: M[10:3]= COPY OF C[10:3] AT ENTRY FROM ROMCHK
*
  1521
  1522                    ENTRY  RMRT01
  1523                    ENTRY  RMRT02
  1524 1622 RMRT00     1 GOSUB  TMRSTS        READ TIMER HARDWARE STATUS
  1524 1623             0                     *TIMER ROM:  TM3, @0244
  1525 1624 RMRT01  1004 S2=     0            CLEAR "RUN LABEL" BIT
  1526 1625 RMRT02  1604 S0=     0            CLEAR "DSWKNO" BIT
  1527 1626         1630 C=ST
  1528 1627         1650 PT=B
  1529 1630          450 WRSCR
  1530 1631 ROMRTN 1140 SETHEX
  1531 1632          240 SEL P
  1532 1633            1 GOSUB  LDSST0        PUT UP STATUS SET 0
  1532 1634            0                     *MAINFRAME:  CN1, @1627
  1533                                       IN & ASSUME: NOTHING
  1534                                       OUT: S0-S7= STATUS SET 0
  1535                                         C=REG 14, CHIP 0 ENABLED
  1536                                       USES: C, S0-S7, DADD, PFAD
  1537                                         (NO PT, +0 SUB LEVELS)
  1538
  1539 1635 RMRT05   630 C=M
  1540 1636            1 GOLONG RMCK10        ROM CHECK SUBROUTINE
  1540 1637            2                     *MAINFRAME:  CN9, @1763
*
*******************************************************************************
*
* DEEP SLEEP WAKE UP WITH NO KEY DOWN                          2-24-81 RSW
*
*******************************************************************************
*
* IF THERE IS AN ALARM:
*       - SETS I/O FLAG TO STAY AWAKE
*       - SETS DSWKNO BIT IN TIMER SOFTWARE STATUS
*       - SAVES USER FLAGS 8-11 IN SCRATCH REG B[11]
*       - CLEARS USER FLAGS 8-11
*
  1554                    ENTRY  DSWKNO
  1555 1640 DSWKNO   530 M=C                   SAVE C IN M
  1556 1641         1670 C=REGN 14
  1557 1642          356 BC EX                 B= REG 14
  1558 1643            1 GOSUB  INITMR         INIT TIMER IF NECESSARY
  1558 1644            0                       *TIMER ROM:  TM0, @1524
  1559 1645         1554 ALARM?
  1560 1646         1633 GONC   ROMRTN (1631)  RETURN TO ROMCHK
  1561 1647          470 RDSCR                 C= SOFTWARE STATUS
  1562 1650         1530 ST=C                  PUT UP SOFTWARE STATUS
  1563 1651         1610 S0=     1             REMEMBER DSWKNO
  1564 1652         1630 C=ST
  1565 1653          634 PT=     11
  1566 1654          302 C=B     PT            SAVE USER FLAGS 8-11
  1567 1655          450 WRSCR                   (11= AUTO RUN FLAG)
  1568 1656            1 GOSUB  LDSST0         PUT UP STATUS SET 0
```

```
  1568 1657           0                       *MAINFRAME:  CN1, @1627
  1569                                        IN & ASSUME: NOTHING
  1570                                        OUT: S0-S7= STATUS SET 0,
  1571                                          C=REG 14, CHIP 0 ENABLED
  1572                                        USES: C, S0-S7, DADD, PFAD
  1573                                          (NO PT, +0 SUB LEVELS)
  1574
  1575 1660         1010 S2=    1             SET I/O FLAG TO STAY AWAKE
  1576 1661         1630 C=ST
  1577 1662          102 C=0    PT            CLEAR USER FLAGS 8-11
  1578 1663         1650 REGN=C 14
  1579 1664         1513 GOTO   RMRT05 (1635)
  1580
*******************************************************************
* POWER OFF ENTRY POINT                                  3-19-81 RSW
*******************************************************************
*
  1585                   ENTRY  PWOF00
  1586 1665 PWOF00     1 GOSUB  INITMM         INIT TIMER IF NECESSARY
  1586 1666            0                       *TIMER ROM:  TM0, @1523
  1587 1667          470 RDSCR                 C= TIMER SOFTWARE STATUS
  1588 1670         1530 ST=C                  PUT UP SOFTWARE STATUS
  1589 1671         1004 S2=    0             CLEAR "RUN LABEL" BIT
  1590 1672         1604 S0=    0             CLEAR "DSWKNO" BIT
  1591 1673            1 GOSUB  ENLCD         ENABLE DISP, DISABLE RAM
  1591 1674            0                       *MAINFRAME:  CN1, @1766
  1592 1675          570 READEN                READ ANNUNCIATORS
  1593 1676         1074 RCR    2             MOVE SHIFT ANNUN TO C[S]
  1594 1677            1 GOSUB  ENTMR         ENABLE TIMER, DISABLE RAM
  1594 1700            0                       *TIMER ROM:  TM0, @0342
  1595 1701         1650 PT=B
  1596 1702          776 C=C+C  S              SHIFT ANNUNCIATOR SET?
  1597 1703          143 GONC   PWOF20 (1717)  NO
  1598 1704           10 S3=    1             YES, SETTING TO CLOCK MODE
  1599 1705          116 C=0
  1600 1706         1056 C=C+1                 ALARM IN 0.01 SEC
  1601 1707          550 WSINT                 WRITE & START INTVL TIMER
  1602 1710 PWOF10  1630 C=ST
  1603 1711          450 WRSCR
  1604 1712         1704 CLR ST
  1605                                        CLEAR "RUN LABEL" BIT
  1606                                        TURN DISPLAY OFF AT EXIT
  1607                                        NOT LIGHT SLEEP WAKEUP
  1608 1713         1210 S7=    1             REMEMBER PWOFF
  1609 1714         1304 S13=   0             CLEAR RUNNING FLAG
  1610 1715            1 GOLONG ALM185        GO TO START OF ALARM BUFFER
  1610 1716            2                       *TIMER ROM:  TM3, @1477
  1611
  1612 1717 PWOF20   114 ?S4=1                 DOING CLOCK DISPLAY?
  1613 1720         1703 GONC   PWOF10 (1710) NO
  1614 1721            1 GOSUB  CLRALS         END CLOCK MODE
  1614 1722            0                       *TIMER ROM:  TM2, @1451
  1615 1723 RMRTN   1063 GOTO   ROMRTN (1631)
  1616
*
*******************************************************************
* WAKE UP FROM DEEP SLEEP  (POLLED EVERY TIME)            3-4-81 RSW
*******************************************************************
*
  1622 1724 DSWKON   530 M=C                  SAVE REG-C IN M
```

```
*
* RECLAIM THE TIMER I/O BUFFER
*
  1626 1725           1 GOSUB  SRHBFI         SEARCH FOR TIMER BUFFER
  1626 1726           0                       *TIMER ROM:  TM2, @1137
  1627 1727          23 GOTO   TDSWK3 (1731) (P+1) FOUND TIMER BUFFER
  1628 1730        1733 GOTO   RMRTN  (1723) (P+2)
  1629
  1630 1731 TDSWK3 1334 PT=    13
  1631 1732        1220 LC     10           C= AA....
  1632 1733        1360 DATA=C              RECLAIM TIMER I/O BUFFER
  1633 1734           1 GOSUB  TMRSTS         PUT UP SOFTWARE STATUS
  1633 1735           0                       *TIMER ROM:  TM3, @0244
  1634 1736        1614 ?S0=1               FROM DSWKNO?
  1635 1737        1647 GOC    RMRTN  (1723) YES, DON'T BEEP
  1636 1740           1 GOSUB  CHKALM         LOOK FOR PAST DUE ALARMS
  1636 1741           0                       *TIMER ROM:  TM0, @1230
  1637 1742        1326 ? B#0  XS           ANY PAST DUE ALARMS?
  1638 1743           1 BEEP2                 YES, BEEP TWICE
  1638 1744           0                       *TIMER ROM:  TM3, @0176
  1639 1745        1563 GOTO   RMRTN  (1723)
  1640
  1641 1746 TMRUN     1 GOLONG TRUN           RUN THE TIMER (NFR ENTRY)
  1641 1747           2                       *TIMER ROM:  TM3, @0531
  1642 1750 DPWKNK    1 GOLONG DSWKNO         DEEP SLEEP WAKE UP NO KEY
  1642 1751           2                       *TIMER ROM:  TM3, @1640
  1643                  FILLTO @1751
*   ENTRY POINT FOR PIL PRINTER TO PRINT TIME & DATE         3-3-81 RSW
*
* THE TIMER CODE MUST ONLY USE:
*    A,B,C,G,M,N,R8[13:6], P,Q, S0-S8, +3 SUB LEVELS
*
* IN:   NOTHING
*
  1651 1752           1 GOSUB  IGDHMS         INIT: GET DAYS-HRS-MIN-SEC
  1651 1753           0                       *TIMER ROM:  TM2, @1266
  1652 1754           1 GOSUB  DSPTMD         DISPLAY TIME AND DATE
  1652 1755           0                       *TIMER ROM:  TM3, @0062
*      !! MUST BE IN HEXMODE AT THIS POINT !!!!!!!!!!
  1654 1756         561 CON    @561           GOLONG TO 67134 TO PRINT
  1655 1757         672 CON    @672           DISPLAY (PIL PRINTER ONLY)
  1656 1760 PWROFF    1 GOLONG PWOF00         POWER OFF ENTRY POINT
  1656 1761           2                       *TIMER ROM:  TM3, @1665
  1657 1762 LSWKUP    1 GOLONG LSWK00         LIGHT SLEEP WAKE UP
  1657 1763           2                       *TIMER ROM:  TM3, @0572
*
  1659                  FILLTO @1763
*
* COMMUNICATION ENTRY LOCATION & CHECKSUM
*
  1663 1764           0 NOP                   PAUSE LOOP
  1664 1765        1613 GOTO   TMRUN  (1746) MAIN RUNNING LOOP
  1665 1766        1623 GOTO   DPWKNK (1750) DEEP SLEEP WAKEUP W/NO KEY
  1666 1767        1713 GOTO   PWROFF (1760) POWER OFF ENTRY LOCATION
  1667 1770        1723 GOTO   LSWKUP (1762) I/O SERVICE ENTRY LOCATION
  1668 1771        1333 GOTO   DSWKON (1724) DEEP SLEEP STARTUP ENTRY
  1669 1772           0 NOP                   COLD START ENTRY LOCATION
  1670 1773           3 CON    @03            C
  1671 1774          61 CON    @61            1
  1672 1775          15 CON    @15            M
```

```
1673 1776          24 CON    @24           T (TM-1C)
1674                                       BITS 9&8= 00 MEANS 1 CHIP
1675 1777           0 NOP                  10-BIT CHECKSUM
1676                  END

ERRORS :      0
```

SYMBOL TABLE (BWTMB4 = TIMER ROM QUAD 4 = TM3 = ADDRESSES @56000-57777)

```
AL225B  1554   -  1551
ALM000   732   -
ALM032   751   -   742
ALM033   756   -   752
ALM045  1066   -
ALM047  1101   -  1121
ALM050  1107   -  1076
ALM052  1123   -  1113
ALM055  1144   -  1230
ALM060  1146   -  1131 1127 1124 1122 1117
ALM063  1162   -  1154
ALM070  1175   -  1170 1166
ALM080  1211   -  1203
ALM116  1227   -  1135
ALM135  1235   -
ALM160  1443   -  1366 1361
ALM168  1445   -  1454 1432
ALM169  1447   -  1401
ALM170  1455   -  1442
ALM185  1477   -
ALM200  1503   -  1541 1534 1444
ALM210  1506   -  1502 1457
ALM215  1512   -  1501
ALM220  1515   -  1505 1461
ALM222  1542   -  1536 1532
ALM225  1544   -  1530 1511
ALM226  1555   -  1545
ALM227  1560   -
ALM230  1565   -
ALM232  1571   -  1567
ALM260  1613   -  1605
ALM270  1620   -  1574
ALMNOW  1470   -
BEEP2    176   -
BEEP2K   177   -
BEEPK    222   -
BEEPKP   207   -
BEEPNK   223   -
BEP210   203   -   205
BEPI     161   -
BEPK05   211   -   206
BEPK10   213   -   220
BEPK20   217   -   214
BEPK28   225   -   221
BEPK30   227   -   235
BEPK40   234   -   231
BEPK50   236   -   233  216
CLKDSP   462   -
CLRF10   453   -   451
CLRFLG   446   -
CSTRTN   260   -   254
DPWKNK  1750   -  1766
DSPA10  1277   -  1255
DSPA15  1303   -  1301
DSPA20  1310   -  1302 1276
DSPA21  1313   -  1314
```

DSPA25	1323	-	1356	1352	
DSPA30	1327	-	1332		
DSPA40	1344	-	1347	1337	
DSPA60	1357	-	1354		
DSPA80	1367	-	1330		
DSPA81	1370	-	1424	1345	
DSPA85	1414	-	1410		
DSPA95	1423	-	1426		
DSPA97	1427	-	1450	1405	
DSPTIM	66	-			
DSPTM5	71	-	65		
DSPTMD	62	-			
DSPTMP	70	-			
DSTMDA	60	-			
DSWEEK	115	-			
DSWEKA	151	-			
DSWK30	153	-	157		
DSWKNO	1640	-			
DSWKON	1724	-	1771		
FTIME	476	-			
GALB05	777	-	775		
GALB10	1013	-	1010		
GALB20	1016	-	1012		
GALB28	1024	-	1044	773	
GALB30	1026	-	1034		
GALB40	1035	-	1027		
GALB50	1053	-	1061		
GALB55	1062	-	1057		
GTALBL	765	-			
HWSTS	255	-			
IMPY2	371	-	373		
IMPY3	372	-	376	370	
LSWK00	572	-			
LSWK01	573	-	532		
LSWK05	607	-			
LSWK10	612	-	605		
LSWK15	625	-	622		
LSWK20	654	-	703		
LSWK25	671	-	665	657	
LSWK28	705	-	661		
LSWK30	707	-	670		
LSWK58	711	-	637	635	633
LSWK60	713	-	704		
LSWK65	714	-	624		
LSWK67	715	-	710		
LSWK6J	704	-	611		
LSWK70	721	-	642		
LSWK80	723	-			
LSWK85	725	-	676	674	
LSWK90	730	-			
LSWKUP	1762	-	1770		
M306	0	-			
NDAYS	11	-			
NXTALM	1475	-			
PWOF00	1665	-			
PWOF10	1710	-	1720		
PWOF20	1717	-	1703		
PWROFF	1760	-	1767		
RMRT00	1622	-	1610	1601	
RMRT01	1624	-			

```
RMRT02  1625   -
RMRT05  1635   -  1664
RMRTN   1723   -  1745 1737 1730
ROMRTN  1631   -  1723 1646
STOPSW   345   -
SUM3D5   351   -
T12H20   305   -   312
T12H25   307   -   304  301  275
T12H30   311   -   273
T12H40   314   -   310  306
T=T+TP   321   -
TDSWK3  1731   -  1727
TIME+1   474   -
TMRST    253   -
TMRSTS   244   -
TMRUN   1746   -  1765
TO12H    262   -
TRUN     531   -
TRUN10   533   -   603
TRUN35   554   -   551
TRUN40   563   -   536  534
TRUN50   565   -   555
TRUN55   566   -   553
TRUN60   567   -   562
UNNOR1    22   -
UNNOR2    23   -
UNNORM    25   -
UNNORX    20   -
UNORM4    37   -    41
UNORM6    40   -    36
X20       51   -
X20Q      46   -
X20Q8     43   -
```

ENTRY TABLE (BWTMB4 = TIMER ROM QUAD 4 = TM3 = ADDRESSES @56000-57777)

ALM116	1227	-
ALM135	1235	-
ALM185	1477	-
ALM200	1503	-
ALM230	1565	-
ALMNOW	1470	-
BEEP2	176	-
BEEP2K	177	-
BEEPK	222	-
BEEPKP	207	-
BEEPNK	223	-
BEPI	161	-
CLKDSP	462	-
CLRFLG	446	-
DSPTIM	66	-
DSPTMD	62	-
DSPTMP	70	-
DSTMDA	60	-
DSWEEK	115	-
DSWEKA	151	-
DSWKNO	1640	-
FTIME	476	-
GTALBL	765	-
HWSTS	255	-
LSWK00	572	-
LSWK80	723	-
LSWK90	730	-
M306	0	-
NDAYS	11	-
NXTALM	1475	-
PWOF00	1665	-
RMRT01	1624	-
RMRT02	1625	-
STOPSW	345	-
SUM3D5	351	-
T=T+TP	321	-
TIME+1	474	-
TMRST	253	-
TMRSTS	244	-
TO12H	262	-
TRUN	531	-
UNNOR1	22	-
UNNOR2	23	-
UNNORM	25	-
UNNORX	20	-
X20	51	-
X20Q	46	-
X20Q8	43	-

EXTERNAL REFERENCES (BWTMB4 = TIMER ROM QUAD 4 = TM3 = ADR @56000-57777)

A-DHMS	60		
A-DHMS	61		
ACKALM	1164	1455	
ACKALM	1165	1456	
ALM116	1024		
ALM116	1025		
ALM135	763		
ALM135	764		
ALM185	747	1715	
ALM185	750	1716	
ALM200	1144	1452	
ALM200	1145	1453	
ALM230	754	1445	
ALM230	755	1446	
ALMSST	1503		
ALMSST	1504		
ANNOUT	460		
ANNOUT	461		
AXEQ	1225		
AXEQ	1226		
BEEP2	1142	1247	1743
BEEP2	1143	1250	1744
BEEP2K	1341		
BEEP2K	1342		
BEPI	177	207	223
BEPI	200	210	224
CHECKX	20		
CHECKX	21		
CHKALM	1740		
CHKALM	1741		
CLKDSP	643	1616	
CLKDSP	644	1617	
CLKOFF	700	1614	
CLKOFF	701	1615	
CLLCDE	72	556	1256
CLLCDE	73	557	1257
CLRAL0	723		
CLRAL0	724		
CLRALD	563	1147	
CLRALD	564	1150	
CLRALS	1364	1721	
CLRALS	1365	1722	
CLRFLG	1176	1243	
CLRFLG	1177	1244	
DECMPL	1207		
DECMPL	1210		
DSA2ND	1416		
DSA2ND	1417		
DSAMSG	1277	1411	
DSAMSG	1300	1412	
DSPDT	113		
DSPDT	114		
DSPTIM	465		
DSPTIM	466		
DSPTM	101		
DSPTM	102		

```
DSPTMD   470  1754
DSPTMD   471  1755
DSTMDA  1303
DSTMDA  1304
DSWEKA   122
DSWEKA   123
DSWKNO  1750
DSWKNO  1751
ENCP00   164  1151  1571
ENCP00   165  1152  1572
ENLCD    105  1673
ENLCD    106  1674
ENTMR    255   326   474   574   721   735  1506  1523  1677
ENTMR    256   327   475   575   722   736  1507  1524  1700
ENTMRS   711
ENTMRS   712
FTIME   1525
FTIME   1526
GDHMS    462
GDHMS    463
GETMXP  1136  1231
GETMXP  1137  1232
GTALBL  1542
GTALBL  1543
HWSTS    546
HWSTS    547
IDVD4    417
IDVD4    420
IGDHMS  1752
IGDHMS  1753
INITM1   576
INITM1   577
INITMM  1665
INITMM  1666
INITMR   345  1470  1643
INITMR   346  1471  1644
LDSST0   446  1200  1633  1656
LDSST0   447  1201  1634  1657
LSWK00  1762
LSWK00  1763
LSWK80   616
LSWK80   617
LSWK90   570
LSWK90   571
MESSL   1260
MESSL   1261
NEWM.X  1513
NEWM.X  1514
NFRC    1611
NFRC    1612
NORM      14
NORM      15
PWOF00  1760
PWOF00  1761
RMCK10  1636
RMCK10  1637
RMRT01   607
RMRT01   610
RMRT02  1105
RMRT02  1106
```

```
RSTKB   1315
RSTKB   1316
RSTKBT  1414  1427
RSTKBT  1415  1430
RSTSQ    456
RSTSQ    457
RTJLBL  1064
RTJLBL  1065
RTNP+2    32
RTNP+2    33
RUN     1222  1576
RUN     1223  1577
SDHMSC   662
SDHMSC   663
SRHEFI  1725
SRHEFI  1726
SRHBUF  1477
SRHBUF  1500
STMSGF  1620
STMSGF  1621
TIME+1  1101  1171
TIME+1  1102  1172
TMRSTS   645  1067  1362  1602  1622  1734
TMRSTS   646  1070  1363  1603  1623  1735
TRUN    1746
TRUN    1747

End of VASM assembly
```