

Date and Time for the CP/M

By W. C. Hoffer

Nearly all computer systems must know the date and the time of day. The software described here provides an interface between the CP/M operating system from Digital Research and the COMPU/TIME board manufactured in Huntington Beach, California. Since CP/M is hardware independent, I will not go into the details of that system.

The code shown in the listing titled SETIME.PRN provides the ability to set the date and time on the board. When the board is purchased, some software routines are provided. I have taken these and added the wants and needs of CP/M along with my own preferences. Particularly important is that the board accepts invalid dates and times and then hangs up with no indication that there is a problem. Instead of writing all of the code required for input validity checks, I have chosen to warn the user and provide a program abort capability.

Some code in this program is duplicated (READ DATE and READ TIME sections). Initially, this program was written for the purpose of lifting sections of code that could "stand alone" to be used in other programs. This method has worked well for me since hardly any change is required after the initial development. The program checks to see if there is a board in the system, and if there is none, it writes eight zeros into the output buffer. When a board is present, the date and time are set and held. The user is then prompted to strike a key for ZERO SECONDS SYNC, allowing the clock to be set accurately using a known time source. The complete setting procedure takes approximately two minutes and forty seconds. This means that you should input a time that is about two minutes later than the current time. If the program has not prompted for ZERO SECONDS SYNC after about two minutes, you should abort the mission by striking any key and check your input. Please note that all of the listings provided are full of comments that can answer many of your questions. An actual setting is shown in Figure 1.

```
A>SETIME
INPUT MONTH, DAY,HOURS,MINUTES
IN THE FORM MM,DD,HH,MM (MUST BE TWO DIGITS EACH)
INVALID INPUT IS NOT CHECKED AND WILL HANGUP THE BOARD
02,27,12,48
STRIKE ANY KEY TO ABORT THIS PROGRAM AND RETURN TO CP/M
STRIKE KEY FOR ZERO SECONDS SYNC

NOW CHECK THE SETTING

DATE 02/27/78 TIME 12:48:00

A>
```

FIGURE 1 "SETIME" EXAMPLE

The program titled TIME.PRN shows the software that displays the date and time on the console device. The appropriate routines were taken from the SETIME program. Figure 2 shows a request and the result.

```
A>TIME
DATE 02/27/78 TIME 13:02:31

A>
```

FIGURE 2 "TIME" EXAMPLE

TIMESUBS.PRN is a listing of the READ DATE and READ TIME software set up as subroutines which I have stored in PROM beginning at E000H. This allows the user to call for the date or time from a running program.

The calling program must point at the starting address of an 8-byte buffer with the D and E registers. The routines will return an eight byte ASCII date or time.

Now that the date and time can be set and the results can be displayed on the console, we can look at some applications. First, CP/M should display the date and time during a cold boot and a warm boot. These modifications dictate the need for the source code of BIOS. Since the TIMESUBS software is located outside of BIOS, I set aside two vectors in the ENTRY POINT TABLE. Thus, no changes to applications software will be required if TIMESUBS has to be relocated. I left room for expansion in the ENTRY POINT TABLE before establishing my vectors.

I have the IMSAI supplied Version 1.33 of CP/M, and the modifications I have made work very well. If you do not have the same version, you should have no problem incorporating the changes into your BIOS, assuming you have the source code for BIOS and have some insight as to its relation to the CP/M system.

My vector for DATE is at the beginning of BIOS + 45H, and my vector for TIME is at BIOS + 48H. Both vectors point to the TIMESUBS entry points at E000H and E003H. Figure 3 shows the vectors and all of the code that can be placed anywhere in BIOS. The code in Figure 4 must be placed such that it will be executed each time there is a jump to the WARM boot entry point.

```
BD00 C38DBF ENTAB: JMP COLD ;COLD START RETURN
BD03 C36EBF JMP WBOOT ;COME HERE FOR REBOOT (VIA 0)
BD06 C3F2BD JMP CONSTAT
BD09 C30ABE JMP CONIN
BD00 C318BE JMP CONOUT
BD0F C326BE JMP LIST
BD12 C336BE JMP PUNCH
BD15 C347BE JMP READER
BD18 C3CBBD JMP HOME
BD1B C3E4BD JMP SELDSK
BD1E C3D1BD JMP SETTRK
BD21 C3D6BD JMP SETSEC
BD24 C2DBBD JMP SETDMA
BD27 C392BD JMP READ
BD2A C39C8D JMP WRITE
BD2D C350BF JMP NXM ;FOR RESTART 7: GIVE ERR MESSAGE
BD30 C395BF JMP WARM ;WARM BOOT RETURN - FINISH INIT
BD33 DS 18 ;ROOM FOR EXPANSION VECTORS

;
;ANY CHANGES TO THESE VECTORS RELATIVE TO THEIR POSITION
;IN THE ENTRY POINT TABLE/AFFECT THE LIST PROGRAMS
;AND ANY OTHER PROGRAMS THAT USE THESE VECTORS FOR DATE & TIME
;
BD45 C300 GDATE: JMP DATE ;8 BYTE DATE MM/DD/YY
BD48 C303E0 GTIME: JMP TIME ;8 BYTE TIME HH:MM:SS
;
; SIGN-ON MESSAGE, TYPED AFTER RETURN FROM BOOT
;
BD4B 0D0A34384BMESSAGE: DB CR,LF,'48K CP/M EXPERIMENTAL VERS 1.33.3 ',0
;
;DATE & TIME MESSAGE
;
BD71 0D0A444154DATEMESS: DB CR,LF,'DATE'
BD78 58582F5858DATES: DB 'XX/XX/XX TIME '
BD87 58583A5858TIMES: DB 'XX:XX:XX',CR,LF,'S'
```

FIGURE 3 "BIOS" ADDITIONS

```
;
; COLD:
BF8D 2148BD LXI H,MESSAGE ;COLD START SIGN ON
BF90 CDBABF CALL CONMSG
BF93 0E00 MVI C,0 ;COLD STARTS FROM DRIVE ZERO
;
; WARM:
; "BOOT"RETURNS HERE AFTER WARM START
;
;GET TIME & DATE AND PRINT IT
;
BF95 C5 PUSH B ;PRESERVE IOBYTE & SELECTED DISK
BF96 1178BD LXI D,DATES
BF99 CD45BD CALL GDATE ;GET THE DATE
BF90 1187BD LXI D,TIMES
BF9F CD48BD CALL GTIME ;GET THE TIME
BFA2 1171BD LXI D,DATEMESS
BFA5 0E09 MVI C,9
BFA7 CD0500 CALL ENTRY ;PRINT IT
BFAA C1 POP B
```

FIGURE 4 "BIOS" MODIFICATIONS

Disk Operating System

When these modifications have been incorporated into the system, the date and time will be displayed on the console device each time either a cold boot or warm boot is initiated. Anytime you want the date and time you need only to warm boot (control C). This means that you no longer need the TIME program described earlier. Figure 5 is an example of a cold boot and a warm boot.

Everyone involved in software development is plagued with keeping track of the latest listing of the program under development. Many times, I have thrown away the wrong listing and ended up having to get a new listing just to be sure. Having the date and time along with the name of the program at the top of each page of a listing will save a good deal of time.

```
48K CP/M EXPERIMENTAL VERS 1.33.3
DATE 02/27/78 TIME 13:08:56
```

```
A>
```

```
"COLD BOOT"
```

```
^C
```

```
DATE 02/27/78 TIME 13:09:10
```

```
A>
```

```
"WARM BOOT"
```

FIGURE 5 CP/M COLD BOOT AND WARM BOOT EXAMPLES

Incorporating the date and time into a LIST program is also a good application for my new software. The LIST program I use was provided by IMSAI and since it bears a copyright message, I won't supply the listing. If you have Version 1.3 of LIST, you can incorporate these changes directly. If you do not have this Version, you can apply the techniques and some of the code. The code in Figure 6 is self-explanatory and can be placed anywhere in the program. Figure 7 shows TITLBUF in its original form, and Figure 8 shows the changes to make and the addition of the storage area DATEBUF. Figure 9 shows the calls to DATE and TIME which are placed at CIB9 and the two statements that must be added to the PAGE NUMBER routine. The code in Figure 10 can be inserted anywhere where it won't be executed since it is used as a subroutine. This code determines the size of the CP/M system that is currently running and then jumps to the proper place in the ENTRY POINT TABLE of BIOS. This method can be used anytime a program uses the ENTRY POINT TABLE directly. The program listings included in this article were prepared using my list program with date and time.

```
;
; LOCATIONS 1 & 2 CONTAIN THE ADDRESS OF THE WARM START VECTOR
;
0001 =   LOCL EQU 1 ;LOCATION CONTAINING THE ADDRESS OF BIOS+3
;
; OFFSET IN BIOS FOR DATE & TIME VECTORS
; BECAUSE WE CAN ONLY DETERMINE THE START OF BIOS+3
; THE BELOW MUST BE ADJUSTED BY 3
;
0042 =   DADDR EQU 42H ;ACTUALLY IT'S 3 MORE
0045 =   TADDR EQU 45H ;DITTO
```

FIGURE 6 "LIST" ADDITIONS

```
0386 00      TITLBUF: DB 0          ;SAYS NO TITLE HERE YET
0387          DS 100H         ;REST OF TITLE BUFFER
```

FIGURE 7 ORIGINAL "TITLBUF"

```
03D9          TITLBUF: DS 100H         ;TITLE BUFFER
04D9 000A204441 DATEBUF: DB CRC,LPC,' DATE XX/XX/XX TIME XX:XX:XX',CRC,LPC,0
```

FIGURE 8 CHANGES TO "TITLBUF" AND ADDITION OF "DATEBUF"

```
CIB9:
;
; GET TIME & DATE
;
01A0 11E104          LXI    D,DATEBUF+8
01A3 CD3E01          CALL  DATE
01A6 11F004          LXI    D,DATEBUF+23
01A9 CD4801          CALL  TIME

;PAGE NUMBER
033A 3A3B01          LDA  PAGNUMELAG
033D B7              ORA  A
033E CA6003          JZ  NOPAGNUM
;SPACE TO COLUMN
0341 3E20          PAGN1: MVI A,' '
0343 CD8C02          CALL LSTCH ;MINIMUM ONE SPACE
0346 3AF504          LDA  COL
0349 DF41          SET  PAGNUMCOL
034B FA4103          JM  PAGN1
; "PAGE" TEXT
034E 217C03          LXI  H,PAGETXT
0351 CD8203          CALL LSTRING
;NUMBER
0354 2AF704          LHD  PAGE
0357 CD8E03          CALL DECPR

;THE NEXT TWO STATEMENTS WERE ADDED TO GET DATE & TIME
;AT THE TOP OF THE PAGE
035A 21D004          LXI  H,DATEBUF
035D CD8203          CALL  LSTRING
```

FIGURE 9 "LIST" ADDITIONS

```
;
; DATE & TIME ROUTINES-DETERMINES WHERE BIOS IS LOCATED
; AND VECTORS APPROPRIATELY
;
013E D5          DATE: PUSH  D ;
013F 114200          LXI  D,DADDR;
0142 2A0100          LHLD LOCL ;
0145 19          DAD  D ;
0146 D1          POP  D ;
0147 E9          PCHL ;JUMP TO BIOS+DADDR
0148 D5          TIME: PUSH  D ;
0149 114500          LXI  D,TADDR;
014C 2A0100          LHLD LOCL ;
014F 19          DAD  D ;
0150 D1          POP  D ;
0150 F9          PCHL ;JUMP TO BIOS+TADDR
```

FIGURE 10 "LIST" DATE AND TIME SUBROUTINES

PROGRAM LISTING 1

```

;
; THIS PROGRAM WILL SET THE TIME AND DATE ON THE COMPU/TIME
; BOARD MANUFACTURED AT B532 HAMILTON AVE.,
; HUNTINGTON BEACH, CA 92642, (714)536-9962
;
; THE BOARD MUST BE ADDRESSSED BEGINNING AT '00' OR YOU MUST
; CHANGE THE CONTROL & DATA PORT ASSIGNMENTS BELOW
;
; THIS SOFTWARE EXECUTES ON THE CP/M OPERATING SYSTEM
;
; WRITTEN BY V. C. HOPPER-2721 N. WANDA-SIMI VALLEY, CA. -92645
; ORG 100H
%100
; SET UP THE STACK
LXI H,0
DAD SP
SHLD OLDSP
LXI SP,STACK
; ASSIGNMENTS
ENTRY: EQU 5 ;CP/M ENTRY TO BDOS
CR: EQU 02H ;CARRIAGE RETURN
LF: EQU 0AH ;LINE FEED
BCONT: EQU 0C5H ;PORT A CONTROL
ADATA: EQU 0C7H ;PORT A DATA
BDATA: EQU 0C6H ;PORT B DATA
; CHECK FOR BOARD PRESENT
IN ADATA ;READ PORT A DATA
CPI 0FFH ;BOARD PRESENT?
JNZ BEGIN ;YES
RSTCPM: SHLD OLDSP ;NO-RETURN TO CP/M
RST
; SET DATE & TIME
BEGIN:
CALL CLKRES ;RESET CLOCK
CALL CLKINT ;INIT THE CLOCK
CALL GETIMP ;GET THE INPUT
LXI H,MONI ;LOAD B&C WITH MONTH
MOV B,M ;
INX H ;
MOV M,M ;
LXI A,77H ;
CALL SETDIG ;SET MONTH
INX H ;SKIP OVER COMMA
INX H ;
MOV B,M ;
INX H ;
MOV C,M ;
LXI A,57H ;SET DAY
CALL SETDIG ;SET DAY
LXI H,HOURI ;LOAD B&C WITH HOUR
MOV B,M ;
INX H ;
MOV M,M ;
LXI A,77H ;
CALL SETDIG ;SET MONTH
INX H ;SKIP OVER COMMA
INX H ;
MOV B,M ;
INX H ;
MOV C,M ;
LXI A,5EH ;
CALL SETDIG ;SET MINUTES
CALL SETSEC ;SYNC ZERO SEC
; DISPLAY DATE & TIME FOR SET CHECK
LXI D,SETMES
MVI C,9
CALL ENTRY
DISPLAY:
LXI H,MON
CALL DATE ;GET IT
LXI H,HOUR
CALL TIME
LXI D,PDAT; DISPLAY THE WHOLE THING
MVI C,9
CALL ENTRY
RSTCPM ;RETURN TO CP/M
MVI C,1101
DB CR,LF,'NOW CHECK THE SETTING ',CR,LF,'$'
; CONSOLE INPUT ROUTINES
GETINP: LXI D,DMESS ;GET INPUT USING CP/M BUFFERED I/O
MVI C,9
CALL ENTRY ;ASK FOR INPUT
LXI D,BUFF ;GET THE INPUT
MVI C,10
CALL ENTRY
LXI D,DMESS1 ;TELL 'EM HOW TO ABORT
MVI C,9
CALL ENTRY
RST
DB CR,LF,'INPUT MONTH, DAY, HOUR, MINUTES ',CR,LF
DB 'IN THE FORM MM, DD, HH, MM (MUST BE TWO DIGITS EACH) ',CR,LF
DB 'INVALID INPUT IS NOT CHECKED AND WILL HANGUP THE BOARD
DB CR,LF,'$'
2232 00A535452DMESS1: DB CR,LF,'STRIKE ANY KEY TO ABORT THIS PROGRAM AND RETURN TO
CP/M ',CR,LF,'$'
; SET SECONDS SYNCHRONIZATION ROUTINE
SETSEC: LXI D,SMESS
MVI C,9
CALL ENTRY ;SYNC MESSAGE
KWAIT: MVI C,11
CALL ENTRY ;WAIT FOR ANY KEY
RAR
JNC KWAIT ;NOTHING WAITING
RST ;SET TIME
2221 00A535452SMESS: DB CR,LF,'STRIKE KEY FOR ZERO SECONDS SYNC ',CR,LF,'$'
; RESET CLOCK/CALENDER PORT
CLKRES: MVI A,0 ;RESET PORT A
OUT ACONT ;RESET PORT B
22A8 D3C5
22A4 D3C7
22AC C9
; INITIALIZE CLOCK/CALENDER PORT
CLKINT: MVI A,76H ;STORE 76H AT PORT A DATA REGISTER
OUT ADATA
MVI A,77H
OUT BDATA ;STORE A 77H AT PORT B DATA REGISTER
MVI A,14H ;INTERRUPT CODE
OUT ACONT
MVI A,0AH ;INTERRUPT CODE
OUT BCONT
RST
; 1 HZ WAIT ROUTINE
WAIT: IN BDATA ;RESET INTERRUPT
WAIT: IN BCONT
ANI 80H ;CHECK FOR 1 HZ INTERRUPT
RZ ;RETURN IF YES
JMP WAIT ;LOOP IF NOT
; READ A DIGIT ROUTINE
RDIGIT: MOV A,D ;SELECT DIGIT
OUT ADATA
IN ADATA ;RESET INTERRUPT
DWAIT: IN ACONT ;TEST FOR DIGIT PRESENT
ANI 80H ;ANYTHING THERE?
JZ DWAIT ;LOOP UNTIL INTERRUPT
IN ADATA ;READ A DIGIT
ANI 0FH ;MASK ZONE
ORI 30H ;SET ASCII
RST
; READ FOUR DIGITS ROUTINE
READ4: MVI D,0 ;SET TO SELECT FIRST DIGIT
NEXT: CALL RDIGIT ;DELAY ONE DIGIT SCAN
CALL RSDIG ;READ & STORE DIGIT
2223 7A
2224 FE2
222E C2F902
22F9 79
22FA F202
22FC C2F402
22FE 32F
22FF C3F602
2300 3331
2301 CDF003
2302 7A
2303 F340
2304 C9
2305 C9
; STORE A DIGIT ROUTINE
SDIGIT: MOV M,A ;STORE A DIGIT
INX H ;INCR H&L
RST
; READ DATE ROUTINE
DATE: CALL BOARD ;IS THERE A BOARD
JZ NOBOARD ;NOPE
CALL CLKINT ;GET TIME
MVI A,0 ;SET DATE DISPLAY MODE
OUT BDATA
C,0 ;TELL READ4 THIS IS DATE
CALL READ4 ;GET 4 DIGITS
MVI A,'/' ;
CALL SDIGIT ;
MVI A,'7' ;SET TENS OF YEARS
CALL SDIGIT ;
MVI A,'0' ;SET UNITS OF YEARS
CALL SDIGIT ;
RST
; READ THE TIME
TIME: CALL BOARD ;CHECK FOR BOARD PRESENT
JZ NOBOARD ;NOPE
CALL CLKINT ;INITIALIZE THE BOARD
CALL A,40H ;SET TIME DISPLAY MODF
OUT BDATA
MVI C,1 ;TELL READ4 THIS IS TIME
CALL READ4 ;GET 4 DIGITS
MVI A,'/' ;
CALL SDIGIT ;STORE A DIGIT
CALL RSDIG ;READ & STORE A DIGIT
; READ AND STORE A DIGIT
RSDIG: CALL RDIGIT ;
CALL SDIGIT ;
MOV A,D ;
ADI 10H ;STEP TO NEXT DIGIT
MOV D,A ;
RST ;
; NO BOARD IN THE SYSTEM
NOBOARD:
MVI A,'0' ;STUFF 0 ZEROS
CALL SDIGIT ;
CALL SDIGIT ;
CALL SDIGIT ;
CALL SDIGIT ;
CALL SDIGIT ;
CALL SDIGIT ;
CALL SDIGIT ;
CALL SDIGIT ;
CALL SDIGIT ;
CALL SDIGIT ;
RST
; SEE IF BOARD PRESENT
BOARD: IN ADATA ;CHECK FOR BOARD PRESENT
CPI 0FFH
RST
; DIGIT SET ROUTINE
SETDIG:
OUT BDATA ;CLOCK SET MODE
MOV E,A ;
ANI 20H ;TEST BIT 5(1=HRS/MON;0=MIN/DATS)
JZ MINDAY ;
MVI D,0 ;SET D TO ZERO IF BIT 5=1
JMP HCALL ;SET D TO 20H IF BIT 5=0
MINDAY: MVI D,20H
HCALL: CALL HWAIT ;CHECK FOR ABORT WANTED
CALL RDIGIT ;READ A DIGIT
CMP B ;CHECK 'TENS' DIGIT
JNZ HCALL ;LOOP ON NO MATCH
MVI A,10H ;
ADD D ;STEP TO UNITS DIGIT
MOV D,A
CALL CHKABORT ;
CALL RDIGIT ;
CMP C ;CHECK "UNITS" DIGIT
MVI A,B ;
JNZ TEST ;
MVI A,40H ;SET TIME HOLD MODE
OUT BDATA ;
RST ;
; CHECK FOR ABORT WANTED
CHKABORT:
PUSH B ;SAVE THE REGISTERS
PUSH H ;
PUSH D ;
MVI C,11 ;SEE IF A KEY HAS BEEN DEPRESSSED ON CONSOLE
CALL ENTRY ;
RAR ;
JNC GOAHEAD ;NO ABORT WANTED
LXI D,ARMESS ;
MVI C,9 ;
CALL ENTRY ;PRINT ABORT MESSAGE
JMP DISPLAY ;DISPLAY THE ABORTED MFS5
GOAHEAD:
POP D
POP H
POP B
RST
23B5 00A53524PARMESS: DB CR,LF,'PROGRAM ABORTED DURING SET-RESULTS UNCERTAIN ',CR,LF,'$'
; STORAGE AREA

```

```

03E7 0D0A444154PDATE DB CR,LF,'DATE
03E8 5858 MON DB 'XX',MONTH
03F0 DS 1
03F1 5858 DAY DB 'XX',DAY
03F2 DS 1
03F3 2020544494DPTIME DB 'XX',TIME
03F4 5858 HOUR DB 'XX',HOURS
03F5 DS 1
0402 5858 MIN DB 'XX',MINUTES
0403 5858 SEC DB 'XX',SECONDS
0404 010A24 DB CR,LF,'$'
0405 010A24 DB 13,0
0406 MONI DS 2
0407 DS 1
0408 DS 2
0409 DS 1
0410 HOURI DS 1
0411 DS 1
0412 DS 1
0413 DS 2
0414 DS 2
0415 OLDSP DS 2
0416 DS 100H
0417 DS 0
0517 00 STACK: DB 0
    
```

```

0194 3E3A MVI A,0
0195 CD5B01 CALL SDIGIT
0199 CD9C01 CALL RSDIG
019C CD2301 RSDIG: CALL RDIGIT
019F CD5B01 CALL SDIGIT
01A2 7A MOV A,D
01A3 CE10 ADI 10H
01A5 57 MOV D,A
01A6 C9 RET
    
```

```

01A7 3F30 MVI A,0 ;STUFF 8 ZEROS
01A9 CD5B01 CALL SDIGIT
01AC CD5B01 CALL SDIGIT
01AF CD5B01 CALL SDIGIT
01B2 CD5B01 CALL SDIGIT
01B5 CD5B01 CALL SDIGIT
01B8 CD5B01 CALL SDIGIT
01BB CD5B01 CALL SDIGIT
01BF CD5B01 CALL SDIGIT
01C1 C9 RET
    
```

```

01D0 0D0A444154PDATE DB CR,LF,'DATE
01D1 5858 MON DB 'XX',MONTH
01E1 DS 1
01E2 5858 DAY DB 'XX',DAY
01E3 DS 1
01E4 2020544494DPTIME DB 'XX',TIME
01E5 5858 HOUR DB 'XX',HOURS
01E6 DS 1
01E7 5858 MIN DB 'XX',MINUTES
01E8 DS 1
01E9 5858 SEC DB 'XX',SECONDS
01EA 010A24 DB CR,LF,'$'
01EB 010A24 DB 13,0
01EC MONI DS 2
01ED DS 1
01EE DS 2
01EF HOURI DS 1
01F0 DS 1
01F1 DS 1
01F2 DS 2
01F3 DS 2
01F4 DS 100H
01F5 DS 0
03E0 00 STACK: DB 0
    
```

PROGRAM LISTING 2

```

; THIS PROGRAM WILL READ & DISPLAY THE TIME AND DATE FROM A
; COMPU/TIME BOARD MANUFACTURED AT 8532 HAMILTON AVE.,
; HUNTINGTON BEACH, CA. 92646, (714) 536-9967
;
; THE BOARD MUST BE ADDRESSED BEGINNING AT 'C0' OR YOU
; MUST CHANGE THE PORT ASSIGNMENTS BELOW
;
; THIS SOFTWARE EXECUTES ON THE CP/M OPERATING SYSTEM
; WRITTEN BY W.C.HOFFER-2721 N. WANDA-SIMI VALLEY, CA.-93065
0100 ORG 100H
; SET UP THE STACK
0100 210020 LXI H,0
0103 39 DAD SP
0104 220602 SHLD SP,STACK
0107 310503 LXI SP,STACK
; ASSIGNMENTS
0005 = ENTRY: EQU 5
000D = CR: EQU 0DH
000A = LF: EQU 0AH
0005 = ACONT: EQU 0C5H ;PORT A CONTROL
0007 = BCONT: EQU 0C7H ;PORT B CONTROL
0004 = ADATA: EQU 0C4H ;PORT A DATA
0006 = BDATA: EQU 0C6H ;PORT B DATA
; DISPLAY DATE & TIME
010A 11DF01 LXI D,MON
010D CD5E01 CALL DATE ;GET IT
0110 11E001 LXI D,HOUR
0113 CD8101 CALL TIME
0116 11DB01 LXI D,PDATE ;DISPLAY THE WHOLE THING
0119 E890 MVI C,9
011B CD0500 CALL ENTRY
; RETURN TO CMP THRU CCP
011E 2A0602 LHLD OLDSP
0121 F9 SPHL
0122 C9 RET
; READ A DIGIT ROUTINE
0123 7A RDIGIT: MOV A,D ;SELECT DIGIT
0124 D304 OUT ADATA
0126 DBC4 IN ADATA ;RESET INTERRUPT
0128 DBC5 DWAIT: IN ACONT ;TEST FOR DIGIT PRESENT
012A F600 ANI 80H ;ANYTHING THERE?
012C CA2B01 JZ DWAIT ;LOOP UNTIL INTERRUPT
012F DBC4 IN ADATA ;READ A DIGIT
0131 F00F ANI 0FH ;MASK ZONE
0133 F630 ORI 30H ;SET ASCII
0135 C9 RET
; READ FOUR DIGITS ROUTINE
0136 1E00 READ4: MVI D,0 ;SET TO SELECT FIRST DIGIT
0138 CD3301 CALL RDIGIT ;DELAY ONE DIGIT SCAN
0139 CD9C01 NEXT: CALL RSDIG ;READ & STORE DIGIT
013E 7A MOV A,D
013F FE20 CPI 20H ;TEST IF 2 DIGITS DONE
0141 C95401 JNZ SKIP ;SKIP A PLACE
0144 90 MOV A,C ;SEE IF TIME OR DATE
0145 FF00 CPI 0
0147 C24F01 JNZ COLON ;IT'S FOR TIME
014A 3E2F MVI A, '/'
014C CD5101 JMP DOIT
014F 3E3A COLON: MVI A, ':'
0151 CD5B01 DOIT: CALL SDIGIT
0154 7A SKIP: MOV A,D ;TEST FOR ALL DIGITS DONE
0155 FE40 CPI 40H
0157 C9 RET ;ALL 4 DONE
0158 C33F01 JMP NEXT ;GET ANOTHER DIGIT
; STORE A DIGIT ROUTINE
015F 77 SDIGIT: MOV M,A ;STORE A DIGIT
0160 2C INX H ;INCR H&L
0162 C9 RET
; READ DATE ROUTINE
; DATE:
0165 CDC001 CALL BOARD ;IS THERE A BOARD
0167 C4701 JZ NOBOARD ;NOPE
0169 CD0701 CALL CLKINT ;INIT THE BOARD
016F 7F XCHG ;PUT DESTINATION OF DATE IN H&L
0170 3F00 MVI A,0 ;SET DATE DISPLAY MODE
0171 DBC6 OUT BDATA
0172 F100 MVI C,0 ;TELL READ4 THIS IS DATE
0173 CD3F01 CALL READ3 ;GET 4 DIGITS
0174 3E2F MVI A, '/'
0175 CD5B01 CALL SDIGIT
0176 3E27 MVI A, 'Y' ;SET TENS OF YEARS
0178 CD5E01 CALL SDIGIT
0179 3E26 MVI A, 'E' ;SET UNITS OF YEARS
017D CD5B01 CALL SDIGIT
0180 C9 RET
; READ TIME & READ AND STORE DIGIT ROUTINES
; TIME:
0181 CD0201 CALL BOARD ;CHECK FOR BOARD PRESENT
0184 C4701 JZ NOBOARD ;NOPE
0187 CD0701 CALL CLKINT ;INIT THE BOARD
018A EF XCHG ;PUT DESTINATION OF TIME IN H&L
018B 3F40 MVI A,40H ;SET TIME DISPLAY MODE
018D DBC6 OUT BDATA
018F FE01 MVI C,1
0191 CD3E01 CALL READ4 ;GET 4 DIGITS
    
```

```

; SEE IF BOARD PRESENT
; BOARD:
01C2 DBC4 IN ADATA
01C4 FFFH CPI 0FFH
01C6 C9 RET
CLKINT: MVI A,70H
01C9 D3C4 OUT ADATA
01CB 3E77 MVI A,77H
01CD D306 OUT BDATA
01CF 3E14 MVI A,14H
01D1 D3C5 OUT ACONT
01D3 3F04 MVI A,04H
01D5 D3C7 OUT BCONT
01D7 C9 RET
; STORAGE AREA
01D0 0D0A444154PDATE DB CR,LF,'DATE
01D1 5858 MON DB 'XX',MONTH
01E1 DS 1
01E2 5858 DAY DB 'XX',DAY
01E3 DS 1
01E4 2020544494DPTIME DB 'XX',TIME
01E5 5858 HOUR DB 'XX',HOURS
01E6 DS 1
01E7 5858 MIN DB 'XX',MINUTES
01E8 DS 1
01E9 5858 SEC DB 'XX',SECONDS
01EA 010A24 DB CR,LF,'$'
01EB 010A24 DB 13,0
01EC MONI DS 2
01ED DS 1
01EE DS 2
01EF HOURI DS 1
01F0 DS 1
01F1 DS 1
01F2 DS 2
01F3 DS 2
01F4 DS 100H
01F5 DS 0
03E0 00 STACK: DB 0
; THESE SUBROUTINES GET THE DATE & TIME FROM
; A COMPU/TIME BOARD MANUFACTURED AT 8532 HAMILTON AVE.,
; HUNTINGTON BEACH, CA. 92646, (714) 536-9967
;
; THE BOARD MUST BE ADDRESSED BEGINNING AT 'C0' OR YOU MUST
; CHANGE THE PORT ASSIGNMENTS BELOW
;
; THE ADDRESS OF THE DESTINATION OF THE DATE & TIME MUST BE IN
; THE DSE REGISTER WHEN THESE ROUTINES ARE CALLED
;
; A CALL TO TIME RETURNS HH:MM:SS (P ASCII BYTES)
; A CALL TO DATE RETURNS MM/DD/YY (DITTO)
;
; THIS SOFTWARE EXECUTES ON THE CP/M OPERATING SYSTEM
; WRITTEN BY W.C.HOFFER-2721 N. WANDA-SIMI VALLEY, CA.-93065
0000 ORG 0E00H
; ASSIGNMENTS
0005 = ACONT: EQU 0C5H ;PORT A CONTROL
0007 = BCONT: EQU 0C7H ;PORT B CONTROL
0004 = ADATA: EQU 0C4H ;PORT A DATA
0006 = BDATA: EQU 0C6H ;PORT B DATA
; READ A DIGIT ROUTINE
000E 7A RDIGIT: MOV A,D ;SELECT DIGIT
000F D304 OUT ADATA
0011 DBC4 IN ADATA ;RESET INTERRUPT
0013 DBC5 DWAIT: IN ACONT ;TEST FOR DIGIT PRESENT
0015 F600 ANI 80H ;ANYTHING THERE?
0017 CA2B01 JZ DWAIT ;LOOP UNTIL INTERRUPT
001A DBC4 IN ADATA ;READ A DIGIT
001C F00F ANI 0FH ;MASK ZONE
001E ORI 30H ;SET ASCII
001F C9 RET
; READ FOUR DIGITS ROUTINE
0020 1E00 READ4: MVI D,0 ;SET TO SELECT FIRST DIGIT
0022 CD3301 CALL RDIGIT ;DELAY ONE DIGIT SCAN
0023 CD9C01 NEXT: CALL RSDIG ;READ & STORE DIGIT
0025 7A MOV A,D
0027 FE20 CPI 20H ;TEST IF 2 DIGITS DONE
0029 C95401 JNZ SKIP ;SKIP A PLACE
002B 90 MOV A,C ;SEE IF TIME OR DATE
002D FF00 CPI 0
002F C24F01 JNZ COLON ;IT'S FOR TIME
0030 3E2F MVI A, '/'
0032 CD5101 JMP DOIT
0035 3E3A COLON: MVI A, ':'
0037 CD5B01 DOIT: CALL SDIGIT
0039 7A SKIP: MOV A,D ;TEST FOR ALL DIGITS DONE
003B FE40 CPI 40H
003D C9 RET ;ALL 4 DONE
003E C33F01 JMP NEXT ;GET ANOTHER DIGIT
; STORE A DIGIT ROUTINE
0040 77 SDIGIT: MOV M,A ;STORE A DIGIT
0041 2C INX H ;INCR H&L
0042 C9 RET
; READ DATE ROUTINE
; DATE:
0045 CDC001 CALL BOARD ;IS THERE A BOARD
0047 C4701 JZ NOBOARD ;NOPE
0049 CD0701 CALL CLKINT ;INIT THE BOARD
004F 7F XCHG ;PUT ADDRESS OF DESTINATION IN H&L
0050 3F00 MVI A,0 ;SET DATE DISPLAY MODE
0051 DBC6 OUT BDATA
0052 F100 MVI C,0 ;TELL READ4 THIS IS DATE
0053 CD3F01 CALL READ3 ;GET 4 DIGITS
0055 3E2F MVI A, '/'
0057 CD5B01 CALL SDIGIT
0058 3E27 MVI A, 'Y' ;SET TENS OF YEARS
005A CD5E01 CALL SDIGIT
005B 3E26 MVI A, 'E' ;SET UNITS OF YEARS
005D CD5B01 CALL SDIGIT
005F C9 RET
; STORE A DIGIT ROUTINE
0060 77 SDIGIT: MOV M,A ;STORE A DIGIT
0061 2C INX H ;INCR H&L
0062 C9 RET
; READ DATE ROUTINE
; DATE:
0065 CDC001 CALL BOARD ;IS THERE A BOARD
0067 C4701 JZ NOBOARD ;NOPE
0069 CD0701 CALL CLKINT ;INIT THE BOARD
006F 7F XCHG ;PUT ADDRESS OF DESTINATION IN H&L
0070 3F40 MVI A,40H ;SET TIME DISPLAY MODE
0071 DBC6 OUT BDATA
0072 FE01 MVI C,1
0074 CD3E01 CALL READ4 ;GET 4 DIGITS
    
```

PROGRAM LISTING 3

YOUR S-100 BUS NEEDS FILLED!

* Main Frames * Disk Systems
* Memory Boards * Interface Boards

	Credit Card Price	Cash Discount Price
Dynabyte 250ns 16K RAM (asm.)	\$ 520.	\$ 500.
Dynabyte Naked Terminal (asm.)	\$ 328.	\$ 315.
Cromemco 250ns 16K RAM Kit	\$ 464.	\$ 446.
Cromemco Bytesaver Kit	\$ 136.	\$ 131.
Cromemco Z-2 Computer Kit	\$ 557.	\$ 536.
Cromemco Z-2D Computer Kit	\$1399.	\$1345.
IMSAI PCS-80/15 Kit	\$ 731.	\$ 703.
IMSAI 8080 Kit with 22-slot M.B.	\$ 593.	\$ 570.
IMSAI VIO-C Kit	\$ 297.	\$ 286.
Trace 16K Static RAM (asm. on 32K board)	\$ 529.	\$ 509.
Trace 32K Static RAM (asm.)	\$ 883.	\$ 849.
North Star Micro Disk System Kit	\$ 623.	\$ 599.
North Star MDS Second drive Kit	\$ 395.	\$ 350.
North Star Horizon 1 computer Kit	\$1497.	\$1439.
North Star Horizon Second Drive	\$ 395.	\$ 380.
North Star Horizon 1 (asm.)	\$1777.	\$1709.
8" Floppy Diskettes (IBM soft-sectored)	\$ 6.	\$ 5.50
5" Floppy Diskettes (North Star format)	\$ 4.50	\$ 4.50
TDL ZPU (asm.)	\$ 183.	\$ 176.
TDL D-32 RAM (asm.)	\$ 789.	\$ 739.
TDL Software Package A (with 1 2K BASIC)		
For North Star Disk	\$ 228.	\$ 219.
On Paper Tape	\$ 186.	\$ 179.
On Tarbell Cassette	\$ 168.	\$ 161.
Lear Siegler ADM-3A Kit	write or call for prices	
Seals Electronics 8K 250ns RAM (asm.)	\$ 187.	\$ 180.
George Risk Model 756 ASC II Keyboard (asm.)	\$ 71.	\$ 68.
Metal Cabinet for Model 756	\$ 28.	\$ 27.

Shipping charges: \$10 per CPU on larger units; \$1.50 per kit. \$2.00 min. per order.
Delivery is stock to 30 days on most items. Shipment is immediate for payment by cashier's check, money order or charge card. Allow 3 weeks for personal checks to clear. N.Y. State residents add approx. sales tax. Availability, prices and specs may change without notice.

Write or Call
computer enterprises™

P.O. Box 71
Fayetteville, N.Y. 13066
Phone (315) 637-6208 Today!

Operating Hours:
M-W 10-5 E.S.T.
Th-F 10-9 E.S.T.
Closed Sat. & Sun.

CIRCLE INQUIRY NO. 66

```

FC4E 4F00      MVI  A,0      ;SET DATE DISPLAY MODF
FC4D 430E      OUT  FDATA
FC4F 430E      MVI  C,0      ;TELL READ4 THIS IS DATE
FC51 0D10E0    CALL READ4    ;GET 4 DIGITS
FC54 3E2F      MVI  A,'?'
FC56 CD7FE0    CALL SDIGIT
FC5C 4F07      MVI  A,'?'    ;SET TENS OF YEARS
FC5F CD7FE0    CALL SDIGIT
FC62 3E2F      MVI  A,'?'    ;SET UNITS OF YEARS
FC65 CD3FF0    CALL SDIGIT
FC63 05        RET
;READ TIME & READ AND STORE DIGIT ROUTINES
;
;TIME: CALL BOARD ;CHECK FOR BOARD PRESENT
FC74 C7AEF0    JZ     NOBOARD ;NOPE
FC77 CA8A2     XCHG ;PUT ADDRESS OF DESTINATION IN H&L
FC7A FB        CALL CLKINT ;INIT THE BOARD
FC7F CDAAE0    MVI  A,40H   ;SET TIME DISPLAY MODF
FC82 3E04      OUT  FDATA
FC85 D30E      MVI  C,1
FC87 0F01      CALL READ4   ;TELL READ4 THIS IS TIME
FC8A 0D10E0    CALL READ4   ;GET 4 DIGITS
FC8D 3E5A      MVI  A,'?'
FC90 CD3FE0    CALL SDIGIT
FC93 CD7FE0    CALL RSDIGIT
FC96 CD2EE0    CALL RDIGIT
FC99 CD3FE0    CALL SDIGIT
FC9C 7A        MOV  A,D
FC9F 0F10      ADI  10H
FA02 05        MOV  D,A
FA05 05        RET
;
;NO BOARD IN THE SYSTEM
;
NOBOARD:
FCFA 3E32      MVI  A,'0'   ;STUFF 6 ZEROS
FCFB CD3FE0    CALL SDIGIT
FCFD CD3FE0    CALL SDIGIT
FCFF CD3FE0    CALL SDIGIT
FC02 CD3FE0    CALL SDIGIT
FC05 CD3FE0    CALL SDIGIT
FC08 CD3FE0    CALL SDIGIT
FC0B CD3FE0    CALL SDIGIT
FC0E CD3FE0    CALL SDIGIT
FC11 CD3FE0    CALL SDIGIT
FC14 05        RET
;
;SEE IF BOARD PRESENT
;BOARD:
FA45 DFC4      IN   ADATA
FA48 F1FF      CPI  0FFE
FA4B 05        RBT
FA4E 3E70      MVI  A,70H
FA51 D30E      OUT  ADATA
FA54 3E70      MVI  A,70H
FA57 D30E      OUT  FDATA
FA5A 0F14      MVI  A,14H
FA5D D30E      OUT  ACONT
FA60 3E74      MVI  A,94H
FA63 D307      OUT  BCONT
FA66 05        RET

```

Many users prefer to employ a higher level language to communicate with their systems. Once the TIME-SUBS software has been added and the modifications have been made to BIOS, you can get the date and time from BASIC or FORTRAN or any other language you use, as long as you have the ability to interface with assembly language routines. I have used the date and time with both BASIC and FORTRAN, but since there are so many versions, I won't go into the details.

In conclusion, I believe that no system is complete without date and time. The uses are only limited by the need. I'm sure many of you have the need for date and time, and I urge you to try the software I've described. □

