SIMPLE DISPLAY AND OPERATING PROGRAM

BY JACK DOLLHAUSEN

THE simple display and operating system described in this article allows any 1802 user to input machine-language programs, and as a bonus, provide a display readout with any Elf using an 1861 video chip.

The program requires 1K bytes of RAM; ½K for display buffer storage, and ½K for program and subroutines that do not alter themselves. The I/O commands are compatible with an expanded Elf using an 1861 TV chip. An EF3 flag is required, and this can be supplied by grounding that input through a toggle switch.

The Program. Load the program shown in the Listing starting at M0000. Flip the RUN switch on and enter any two-byte address. The video display will be a column of eight 4-digit addresses with their corresponding data bytes. Set EF3 to logic 0, insert 00 via the INPUT toggle switches and note that when the INPUT switch is turned on, the display scrolls upward through memory. Entering 01 on the switches will produce a down scroll, and 02 will single-step up for each operation of the INPUT switch. To jump the display anywhere in memory, enter 03 and the two-byte address.

Note that the input address is displayed at the bottom of the CRT screen. This is the "active" position, and all operations are performed from this point.

Address an empty memory location (keep in mind that M0200-M03FF is display buffer storage), and make EF3=1. Now with each operation of the INPUT switch, the byte on the toggle switches will be sequentially input into memory. A pointer reminds you that memory is being changed. When finished, return EF3 to logic 0.

To execute a program from any point in memory, set the display to the beginning address of the program to be run and enter 04. The 1861 is disabled by an 04 command, and the machine is running outside the operating program. To return, flip the RUN switch off/on and enter an address. The program you are

```
INITIALIZATION:
    0000  F8 01 B1 B2 B3 B8
    0006  F8 C9 A1 F8 EA A2
    000C  F8 81 A3

    000F  F8 02 B6 F8 00 A6
    0015  F8 00 56
    0018  16 96 FB 04                 clear display buffer
    001C  3A 15

    001E  F8 00 B4 F8 25 A4
    0024  D4                          R4 is "main" pgm. ctr.

MAIN PROGRAM:
    0025  E2 69                       TV on
    0027  37 27 3F 29                 ENTER high byte
    002B  6C BE                       memory location displayed
    002D  37 2D 3F 2F                 ENTER low byte
    0031  6C AE                       in RE
    0033  37 33
    0035  8E FF 07 AE 33 3F
    003B  9E FF 01 BE                 RE has top of display
    003F  F8 02 B6 F8 00 A6           R6 is display buffer pointer
    0045  16                          display loop:
    0046  9E 7A D3 9E 7B D3           R3 is pgm. ctr. for digit
    004C  8E 7A D3 8E 7B D3           configuration subroutine
    0052  16
    0053  0E 7A D3 4E 7B D3           one display line
    0059  86 FB C0 3A 45              loop for eight lines
    005E  F8 8B A3 F8 10 D3           display filled

    0064  3F 64 3E 77                 ENTER opcode or EF=3
    0068  F8 8B A3 F8 8D A6           EF3=1, put flag and change
    006E  F8 11 D3 2E 6C 5E           byte
    0074  1E 30 33                    inc display and loop for more

    0077  F8 8B A3 F8 8D A6
    007D  F8 12 D3
    0080  6C FB 00 3A 87              opcode 00
    0085  30 35                       shift display up

    0087  6C FB 01 3A 90              opcode 01
    008C  2E 2E 30 35                 shift display down

    0090  6C FB 02 3A 97              opcode 02
    0095  30 33                       single'step display

    0097  6C FB 03 3A 9E              opcode 03
    009C  30 27                       change display address

    009E  6C FB 04 3A B3              opcode 04 (TV off)
    00A3  61                          run program at :
    00A4  2E 9E B0 8E A0              ENTER high address
    00A9  37 A9
    00AB  D0                          ENTER low address
    00AC  00 00 00 00
    00B0  00 00 00                    R0 is pgm. ctr.

    00B3  6C FB 05 3A 64              opcode 05 (TV off)
    00B8  F8 01 BA F8 EB AA           move block of memory
```

FOR THE EXPANDED ELF

Permits easy machine-language input to an 1802-based system

```
00BE 61 22 EA 37 C1
00C3 3F C3 37 C5                  ENTER beginning add. of data
00C7 6C B9                        to be moved (high byte)
00C9 3F C9 37 CB                  ENTER (low byte)
00CD 6C A9
00CF 3F CF 37 D1                  ENTER last add. of data to
00D3 6C 1A                        be moved (high byte)
00D5 3F D5 37 D7                  ENTER (low byte)
00D9 6C 2E
00DB 49 5E 1E 89 F3 3A DB
00E2 2A 99 F3 1A 3A DB
00E8 49 5E 30 25                  return for display

TABLE: DIGIT CONFIGURATION
0100 35 2B 2F 39 27 31 41 51
0108 43 45 56 49 3D 4D 20 24
0110 5B 60 66 65 00 00 00 00
0118 00 00 00 00 00 00 00 00
0120 F0 80 C0 80 F0 80 C0 80
0128 80 A0 F0 20 60 20 20 70
0130 10 F0 80 F0 10 F0 90 90
0138 90 F0 10 70 10 F0 80 80
0140 80 F0 80 F0 90 F0 90 F0
0148 10 F0 50 70 50 F0 50 50
0150 50 F0 10 20 40 40 F0 90
0158 F0 90 90 2F 25 25 A5 EF
0160 F8 7C 3E 7C F8 00 00 00
0168 00 00 AA 00 00 00 00 00
0170 00 00 00 00 00 00 00 00
0178 00 00 00 00 00 00 00 00

SUBROUTINE: DIGIT MAKER
0180 D4
0181 39 87                        Q state identifies hi/lodigit
0183 FE FE FE FE
0187 F6 F6 F6 F6
018B A8 08 A8                     enter here for single digit
018E F8 05 A7                     R5 counts 5 lines per digit
0191 48 56
0193 86 FC 08 A6 3B 9D
0199 96 FC 01 B6
019D 27 87 3A 91
01A1 86 FF 27 A6 33 AB
01A7 96 FF 01 B6
01AB 86 FE FE FE FE
01B0 32 B6 FB 80 3A C0
01B6 86 FC 30 A6 3B C0
01BC 96 FC 01 B6
01C0 96 FB 04 3A 80

SUBROUTINE: TV INTERRUPT
01C7 72 70
01C9 C4 22 78 22 52
01CE F8 02 B0 F8 00 A0
01D1 C4 C4 E2 80
01D8 E2 20 A0 E2
01DC 3C D7
01DE 80 E2 20 A0 2F
01E3 34 DE 30 C7
```

creating may "eat" the operating program space, so keep the operating program on cassette.

To move a block of memory, address the first memory position to be changed and enter 05. Note that the display blanks. Enter the two-byte beginning address of the data to be moved, and then the two-byte ending address. The display will return when the transfer is complete. Enter a two-byte address to get back in the operating program.

The program uses two subroutines. The TV interrupt routine (M01C7) is a standard 512-byte display for the 1861 chip. The digit maker routine (M0180) provides functions useful in any display requiring hex digits, and has two entry points. If entered at M0181, it will display a digit corresponding to the high or low half byte present in the D register. The main program sets buffer pointer R6 to the position of the upper left corner of the digit in the display, and sets the Q line to specify whether the high or the low digit is to be displayed. Before a D3 is executed, R6, D, and Q must be set and the subroutine leaves R6 pointing to the next digit position in the display. The main program uses the subroutine at M003F-FD to create the display. The routine may also be entered at M018B to produce a symbol or digit of your own design. Following the operations for the pointer at M0068-71 will reveal how this works, and space is provided in the configuration table at M0170-7F.

This program does not alter itself and could be put into ROM. There are, however, three bytes of storage at M01E8-EA which would need to be moved. Putting them at the bottom of the display buffer M03xx will add a line of dancing dots and dashes to the display. Registers R2 and RA point to this storage.

The ability to scan memory and to move stacks makes machine language easier to edit and debug. Keep your loop addresses and X designators straight and you can say almost anything to the 1802 . . . in its own language. ◇