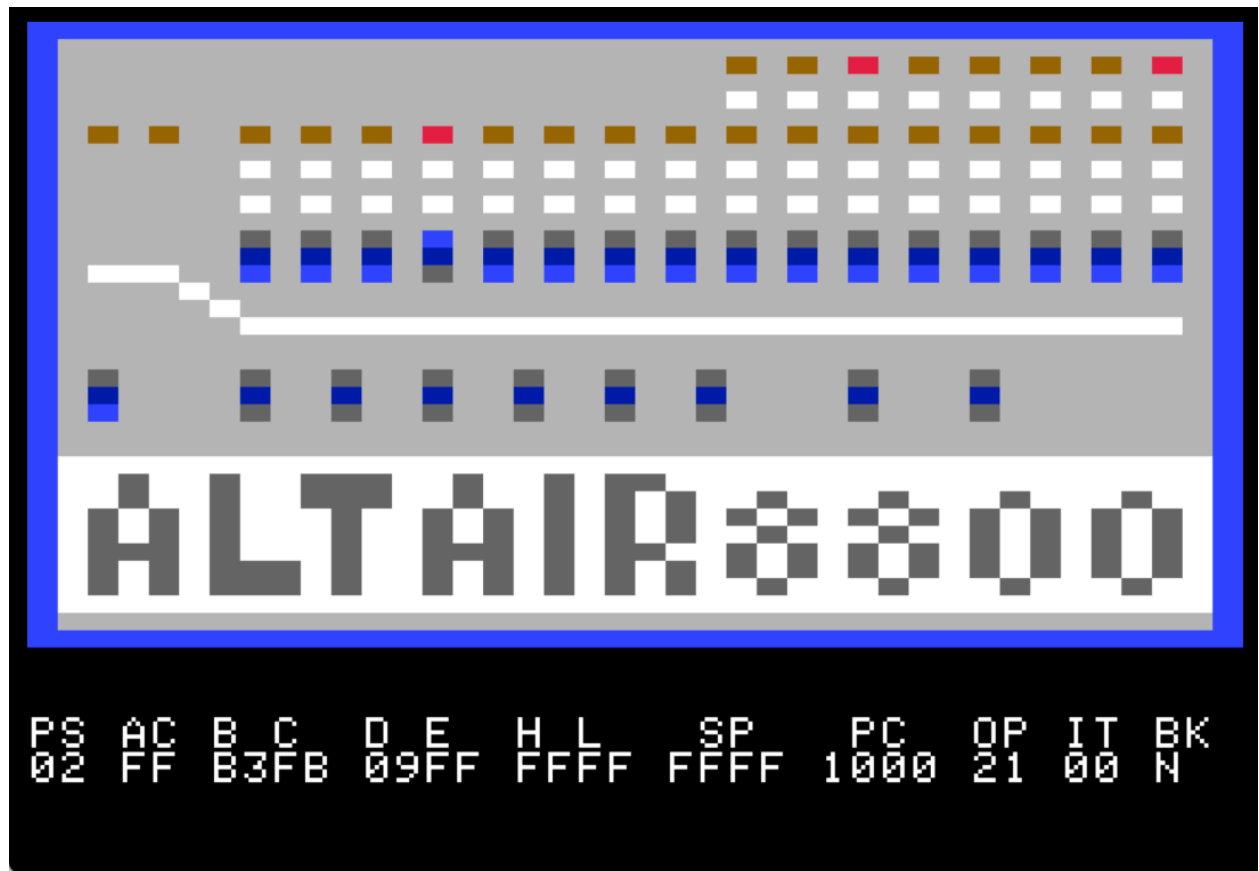


SIM-8800: Altair 8800 emulator for Apple II



Sim-8800 is based on APPLE-80 ©1979 by Dann McCreary.

Additional operating instructions can be found in the APPLE-80 manual.

Release 1.1b

Startup:

At startup, Sim-8800 will ask you to choose a program to load. See below for a summary of the programs, and for links to more information. It will then display a condensed set of instructions. Press the space bar to continue to the main Altair interface mode. Press **I** any time the virtual Altair is paused to view the instruction screen again.

Keyboard commands:

0-9, A-F TOGGLES SWITCHES 0-15

Number keys 0-9 and letters A-F will toggle the corresponding switch on the Altair, numbered 15-0 from left to right. Switches will operate when the Altair is stopped, or during **Trace** (see below).

X TO EXAMINE MEMORY

The byte in memory at the address indicated by the address switches is displayed in the data LEDs. e.g, when the program starts, the address is set to \$0000, so pressing X will display the byte at \$0000.

N FOR NEXT ADDRESS, **^** FOR PREVIOUS

Pressing **N** will advance the program counter address by one, and display the byte at that address. The caret (^) will decrement the address and display the corresponding byte.

P TO DEPOSIT BYTE INTO MEMORY

To write a byte to memory, set switches 15 through 0 to the corresponding address, and e**X**amine that byte. Then set switches 7 through 0 to the byte value you want to write, and press **P** to deposit it into memory.

O TO DEPOSIT NEXT BYTE INTO MEMORY

Once a byte is set into memory, another can be deposited at the next address in memory with the **O** key. This increments the program counter and writes the byte value set with switches 7 through 0 into memory.

SPACE OR **S** TO EXECUTE 1 8080 PROGRAM STEP

This will execute the opcode at the currently displayed program counter address, then pause the machine. The results of the operation are displayed in the status line and the LEDs will update with the new address and data bytes.

T TO START INTERACTIVE TRACE MODE

This will execute a program, one opcode at a time, while updating the LED display and the status line. While tracing a program, you can toggle the switches (emulated input 00). The trace will continue until interrupted with **Control-C** or the program reaches a breakpoint.

G TO RUN PROGRAM IN NON-INTERACTIVE MODE

In this mode, the program will run, full speed, until interrupted by control-C or a breakpoint is reached. Switches and LEDs will not be active during this “non-interactive” mode.

CONTROL-F TO RUN FULL SCREEN MODE

The virtual Altair is connected to an emulated teletype for input and output. In normal operation, you can see only the last 4 lines of text at the bottom of the screen. Press **Control-F** to expand the text to the full screen, and optionally view in 80-column text. Use **Control-C** to pause the machine and return to normal view.

CONTROL-C TO STOP RUN/TRACE MODE

While in Run or Trace mode, the virtual machine can be paused with **Control-C**, and the current program counter and register status will be displayed in the status line and LEDs.

K TO SET BREAKPOINTS

Pressing **K** while the Altair is paused will display the word “Breakpoints” below the status line. You may then enter or edit several memory addresses to be breakpoints for your program separated by spaces. Type in the addresses, or use the arrow keys to move the cursor over an existing breakpoint. Press **Return** when finished. When the program reaches a breakpoint address while tracing or running, it will pause the Altair and display the registers and program counter status.

R TO EDIT REGISTERS

While the Altair is paused, the contents of the registers and accumulator can be edited. Use the arrow keys to move the cursor over the register, and type the new value. Press **Return** to accept the new register values.

I FOR INSTRUCTIONS

When the virtual Altair is paused and not running a program, pressing **I** will display the instructions screen. Pressing **Space** will hide the instructions and return to the virtual Altair display.

Programs

A number of pre-loaded programs are available in this release of Sim-8800. They are each loaded into memory at \$0000 by choosing the appropriate response from the menu when Sim-8800 starts up. You may also start with a blank program, containing only the random bytes that remain in memory from when your Apple II started up.

Altair 4K BASIC

Version 3.2 of Micro-Soft 4K BASIC by Bill Gates, Pul Allen and Monte Davidoff, from 1975.

MITS 8K BASIC

Altair 8K BASIC Version 4.0 from 1976.

When BASIC starts up, it will display

MEMORY SIZE?

Any number (in bytes) can be entered, from 4096 (4K version) or 8192 (8K version) up to 32768 (32K max memory).

It will then ask for the terminal width. The default is 72, so you should enter 40, unless you are running in full-screen mode (press Control-F) at 80 columns, in which case you can choose any width up to 80.

To access more memory, some features of BASIC can be disabled by answering N to the questions “WANT SIN?”, “WANT SQR?” and “WANT RND?”. Those functions will not be available, and a program using them will cause an error.

More information, and a complete manual for Altair BASIC, can be found here: https://www.autometer.de/unix4fun/z80pack/ftp/altair/AltairBasic_1975.pdf

Microchess

Microchess: A Chess playing program for the 8080 Microcomputer, written by Peter Jennings in 1977.

More information and a complete manual can be found here: <https://www.computerhistory.org/chess/doc-434fe976ea550/>

Kill the Bit

Classic front panel lights "game" by Dean McDaniel, May 15, 1975.

```
;          classic "kill the bit"

; 0000                org 0000
; 0000    21    00    00    lxi h,0          ;initialize counter
; 0003    16    80          mvi d,080h       ;set up initial display bit
; 0005    01    0E    00    lxi b,0eh       ;higher value = faster
; 0008    1A          beg: ldax d            ;display bit pattern on
; 0009    1A          ldax d                ;...upper 8 address lights
; 000A    1A          ldax d
; 000B    1A          ldax d
; 000C    09          dad b                ;increment display counter
; 100D    D2    08    00    jnc beg
; 1010    DB    00          in 00h          ;input data from sense switches
; 1012    AA          xra d                ;exclusive or with A
; 1013    0F          rrc                  ;rotate display right one bit
; 1014    57          mov d,a              ;move data to display reg
; 1015    C3    08    00    jmp beg        ;repeat sequence
; 1018                end

21 00 10 16 80 00 0E 80 1A 1A 1A 1A 09 D2 08 00 DB 00 AA 0F 57 C3 08 00
```

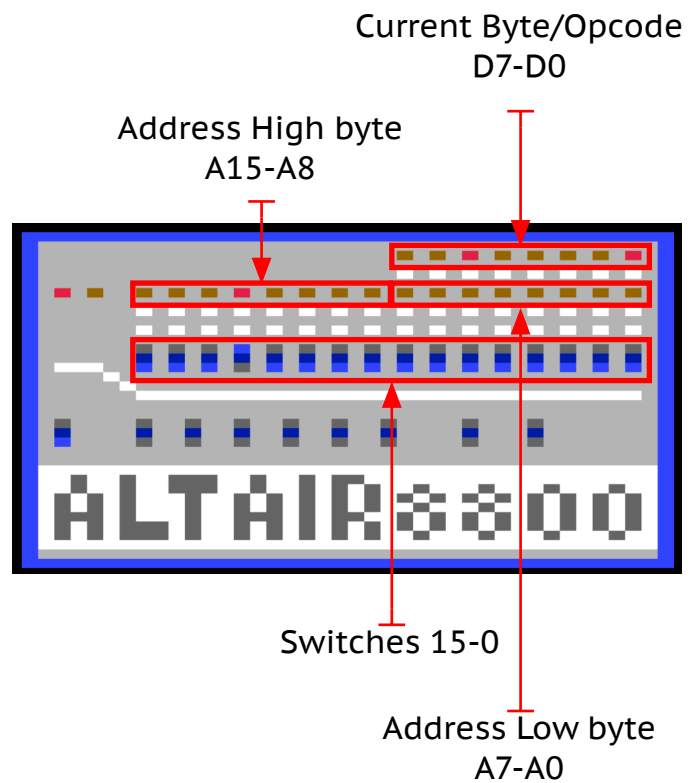
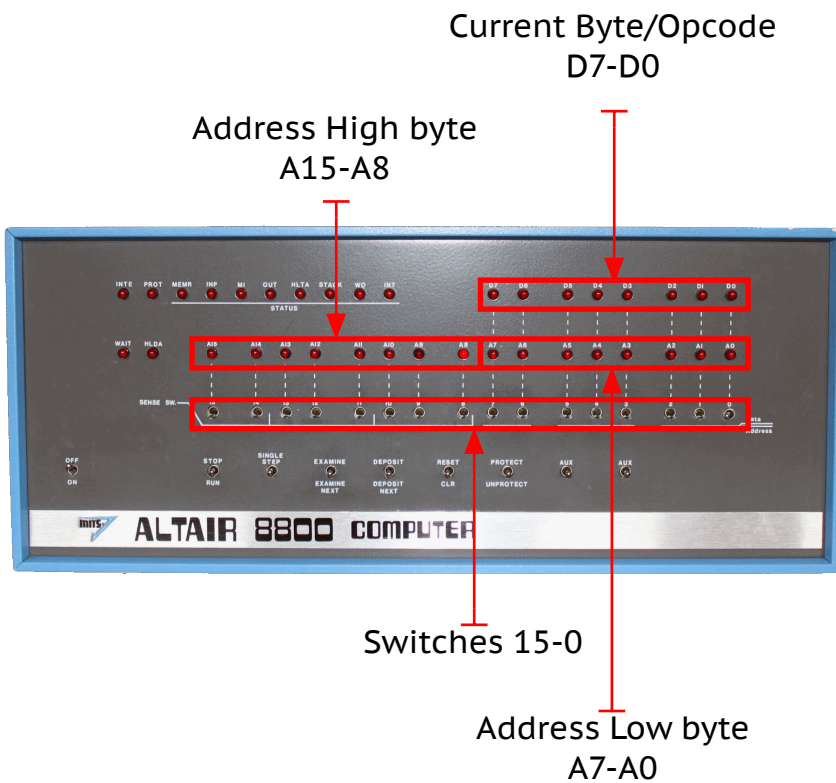
Before tracing the program, set switches 15-8 (F-8) off. As the program traces, you will see one LED in the Address High Byte bank of LEDs light up and move across. The goal is to "kill" the bit by flipping up the switch under the lit LED while it is still lit, then flip it down again once the LED turns off. If you miss, another bit is added to the pattern for you to kill.

Additional Altair 8800 programs can be found online that take advantage of the switches and LEDs, or a teletype connected to SIO card for input and output. To get them to run on the virtual Altair, a few considerations must be made:

- The emulator simulates a teletype connected via a simplified 88-SIO serial interface card. Serial status is on input 00, serial data on input/output port 01.

- Sim-8800 LEDs are output FF, otherwise they are set to display the byte in register D. This comes in particularly handy for the "Kill the Bit" game.

SIM-8800 Altair display mode



8080 Status line

