

RESOURCES

IEEE
696-1983

THE ALTAIR'S S-100 INTERNAL BUS DESIGN BECAME AN IEEE STANDARD, RETIRED IN 1994



RESOURCES_HANDS ON

The MITS Altair 8800 was the first commercially successful personal computer.

Created by Ed Roberts in 1974, it was purchased by the thousands via mail order, proving there was a huge demand for computers outside universities and large corporations. Its influence was

immense: For example, after seeing the Altair featured on the cover of the January 1975 issue of *Popular*

Electronics, Bill Gates and Paul Allen founded Microsoft (then Micro-Soft) in order to write a Basic interpreter for the new machine. • The Altair sold for US\$439 in kit form. Original machines are now collectors' items that trade for thousands of dollars. Fortunately, there are some cheaper alternatives for people who want to get a direct understanding of the Altair computing experience. Modern kits that replicate the Altair hardware as faithfully as possible are available, as are purely virtual online simulators. Falling somewhere between a replica and a simulation is the \$149 Altairduino kit from Chris Davis. The Altairduino duplicates the front panel of the Altair in all its LED- and switch-festooned glory while emulating the internal hardware (including some once fantastically expensive peripherals), using an Arduino Due. • The Altairduino is derived from David Hansel's work on cloning the Altair with the Arduino Due and Arduino Mega 2560. If you want to build one of Hansel's designs from scratch, you can do so by following his free instructions on hackster.io. The advantage of Davis's kit is that it provides all the components, including a nice bamboo case and plastic front panel, along with a custom printed circuit board (PCB) that greatly simplifies construction. • The original Altair's relatively large size means that most of the components are fairly well spaced out on the PCB. Even a beginner ▶

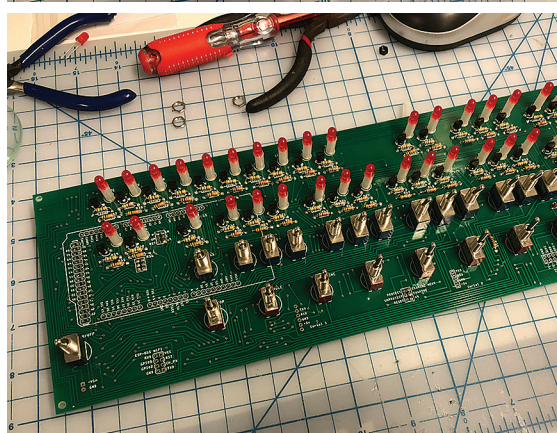
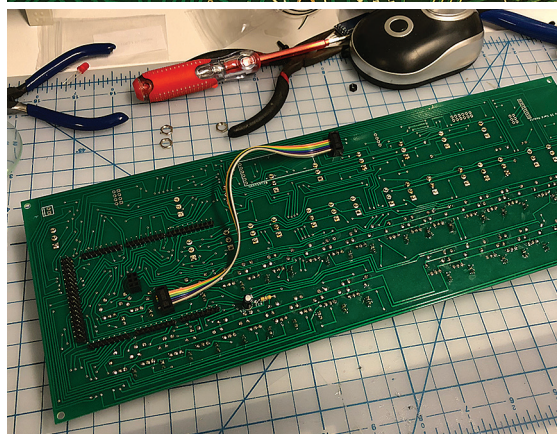
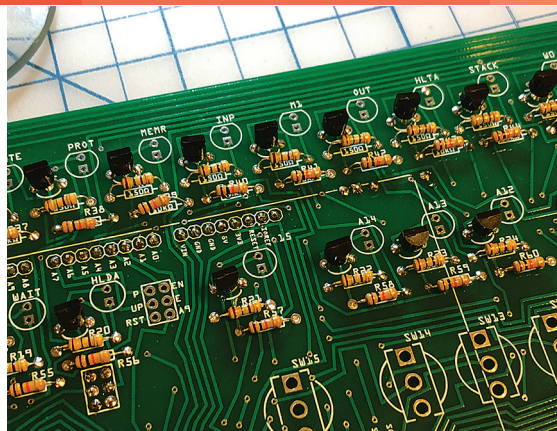
THE ALTAIR,
REINCARNATED
AN ARDUINO-
BASED KIT
TAKES YOU
BACK
TO 1974

could do much of the soldering, although a little bit more experience is required for trickier areas, such as the headers that connect to the Due. The fiddliest bit is adding the LED indicator lights. These are mounted on spacers, and it's best to put the front panel in place to ensure alignment, which can put you in one of those situations where you really wish you had an extra pair of hands to hold the panel, LED, and PCB tightly together while you solder. The online instructions are detailed and well illustrated, but I would recommend skipping forward and making sure you solder all the resistors in the kit before proceeding to add other, taller, components.

The Altairduino improves on the original Altair in two important respects. First, it offers modern interface options. You can connect an old-school terminal using an optional DB-9 connector (which I will stipulate should properly be called a DE-9 connector, so no need to send me letters this time!), but you can also use a soft terminal running on a computer via a USB connection, or even Bluetooth.

You do the initial configuration of the Altairduino via USB. The instructions are written for Microsoft Windows, so I had to do a little poking around the forums on the Altairduino site to figure out how to get my Mac to talk to the USB interface. Setting the baud rate to 115200 when launching the “screen” terminal command did the trick, and once I set the Bluetooth connection up as the power-on default, it was all smooth sailing.

The second big improvement is that the Altairduino comes loaded with a lot of software. You can call up some programs purely by flipping various front panel switches, such as *Kill the Bit*, a game that hacked the Altair's memory-address indicator lights to act as a 1-dimensional display. Other programs are



LIGHTS, SWITCHES, ACTION: A printed circuit board contains additional circuitry for driving LEDs [top], but most of the work is handed off to an Arduino Due plugged into a rear connector [middle]. The finished PCB [bottom] is mounted in the case.



called up with a combination of switch throws and terminal commands. You can quickly fire up Microsoft's very first 4-kilobyte Basic (which gives you the option, on startup, to disable its sine, random number, and square root

functions to save a little memory), or its more advanced 16-KB Basic. The latter has a number of programs you can load from the Altairduino's memory, including early computer game classics such as *Lunar Lander*, *Star Trek*, and *Hunt the Wumpus*.

You can put additional Altair software on a microSD card, which you plug into a reader that's soldered to the PCB (Davis conveniently offers a one-stop bundle on his website). Once the kit is assembled, you can't access the reader to swap out the card without unscrewing the case, but since the universe of Altair 8800 software isn't growing that rapidly, I'll manage. (That said, I did just see someone announce they had gotten a Forth compiler running on the Altairduino that I'd like to try.)

The card reader emulates the 88-HDSK hard disk that was available in the business version of the Altair, which sold in the late 1970s for \$11,450 to \$15,950. A number of disk images are available in the software bundle, including one with the CP/M operating system. The CP/M software also comes with a bunch of software, including parts 1, 2, and 3 of *Zork*, a pioneering text adventure game.

The Altairduino is a lovely kit that's an enormous amount of fun—it is surprisingly satisfying to control a computer by

flipping switches versus, say, mouse clicks. More seriously, simply looking at pictures of early personal computers, with their blinking rows of lights and bulky cases, can leave the impression they were little more than toys. But engaging with this incarnation quickly demonstrates that the Altair was a capable system, and it becomes much clearer why it was that *this* machine came to be so critical in establishing the value of personal computing.

—STEPHEN CASS

POST YOUR COMMENTS at <https://spectrum.ieee.org/altairduino0318>