# altairsim — CDOS 2.58 on a Cromemco 16FDC

2026-10-24 · 0997717

**CDOS** was the operating system of the Cromemco Z80 S-100 machines. It is a work-alike of Digital Research CP/M 1.4 with Cromemco's own extensions, and it runs many CP/M `.COM` programs unchanged. This example boots CDOS 2.58 from an 8″ diskette through a **Cromemco 16FDC** floppy controller. One board holds the disk controller, the console UART and the boot PROM.

```
cd examples/cdos
altairsim cdos.toml
```

The machine has a 4 MHz Z80 processor, 64K of RAM, one 16FDC and a host bridge (`hb0`) at port B0. The CDOS master diskette is in drive A. The machine file sets `bootstrap = true`, which arms the boot PROM. The machine then loads CDOS with no other setting.

## Boot the machine

The console shows this text:

```
Preparing to boot, ESC to abort
Standby
CDOS version 02.58
Cromemco Disk Operating System
Copyright (C) 1977, 1983 Cromemco, Inc.

A.
```

CDOS comes up to its `A.` prompt. You do not press a key. The 16FDC is set for a fixed 300-baud modem console (switch 5 of the board). This setting tells the boot PROM to skip the terminal speed measurement. On real hardware, the measurement needed a RETURN from the terminal.

The `A.` prompt shows that CDOS waits on drive A. CP/M shows `A>` for the same state. Type these commands at the prompt:

```
DIR       list the files on the disk
STAT      show the system status: memory, devices, and the disk label and date
```

`DIR` lists the 19 files on the master diskette. They include the assembler (`ASMB`, `LINK`), the editor (`EDIT`), the utilities (`XFER`, `INIT`, `DUMP`, `STAT`), `XMODEM`, the host bridge programs (`R`, `W`, `HDIR`) and CDOS itself. `STAT` reads the label sector of the disk. It prints `CDOS2.58` and the date `05-19-82`.

## Copy files to and from the host

The host bridge moves files between the host and the CDOS disk. Type these commands at the `A.` prompt:

```
HDIR [pattern]                  list the files in the host folder
R <hostfile> [cpmfile]          copy a host file to the disk
W <cpmfile> [hostfile] [B|T]    copy a file on the disk to the host
```

The host folder is the folder that you start `altairsim` from. To use a different folder, set `hostdir` on the `hb0` board in the machine file.

## How the boot works

A reset arms the **RDOS 2.52** boot PROM in the 16FDC window at C000. The `startup` list of the machine file runs the PROM. RDOS reads the boot track through the WD FD1793 controller and loads the CDOS cold loader. The loader reads `CDOS.COM` into memory and jumps to it.

CDOS then moves itself to the top of RAM and switches the PROM out with `OUT 40H`. The memory board hides the RAM under the PROM on reads only. The boot reads the PROM, and a write reaches the RAM under it. Because of this, CDOS can put running code in that RAM.

## Two authentic settings

- **4 MHz.** A real Cromemco ran at 4 MHz. The double-density read loop of RDOS only keeps up with a 500 kbit/s diskette at that speed. At 2 MHz, the controller reports Lost Data and the boot fails. Do not change `clock_hz = 4000000`.
- **The mixed-density diskette.** `CDOS258-8IN-DSDD.DSK` is an 8″ double-sided image. Track 0 is single density, because the boot PROM reads only single density. The other tracks are double density. The 16FDC finds the format from the image. You do not set it.

## Console speed

The console runs at full speed. Output appears as fast as CDOS prints it, even with the 300-baud modem setting. To get the speed of a 300-baud modem console, set the rate on the console. A `DIR` then takes many seconds.

```
[board.unit.tty]
connect = "console"
rate    = "real"    # pace the line in real time at the programmed baud
```

`rate = "full"` is the default. It does not pace the emulated line. `rate = "real"` does. When a real serial port is on the other end, that port keeps its own timing. The `rate` setting changes only the timing of the emulated console.

## The disk image

The repository tracks `CDOS258-8IN-DSDD.DSK`, and the machine file mounts it read/write in drive A, as a real machine does. Booting, `DIR` and `STAT` only read it. Before you test writes, copy the image, or add `writeprotect = true` to the drive in the machine file. In a clone, `git checkout` restores a changed image. A downloaded copy has no such way to restore it.