

Release notes for Poly Emulator 5.0 (PM)

Bob Bybee, September 2025

What you will need:

- A system running DOS, or a DOS emulator such as DOSbox or VMware. I use DOSbox version 0.74-3 with the configuration items mentioned below.
- A display card, or an emulated display, compatible with VGA. This allows the Poly Emulator to download a character set closely matching the original Poly Video Terminal Interface (VTI). If you're not using a VGA-compatible display, you can change a setting in the **polydev.dat** file so it works without VGA.
- Set your display's resolution to 1280x720. Larger resolutions will work, but the Poly Emulator display window may be too small for comfortable viewing.
- Some or all of the files provided in ZIP form, listed below.

For instructions on configuring DOSbox to run PM, visit the website mentioned below (or unzip your own copy of the site from the **Website.zip** file). Click on the heading item **Emulator**, and scroll down to the section **Running PM in DOS, Windows, and DOSbox**. My Poly website can be found at <https://deramp.com/polymorphic-computers/> and also in a ZIP file packaged with this release (and described below).

Keep the **Emulator** page open in your browser. You'll want to refer to it while setting up PM.

Other quick tips:

- Poly commands and filenames are case-sensitive. Since most of them are upper case, it's helpful to turn on Caps Lock and use Shift to change to lower case occasionally.
- One exception is L which lists files. The letter L can be upper or lower case.
- Don't confuse L with DIR. Both will list directories. L sends its output to the screen, DIR goes to the printer. You can configure the emulator's printer to go to the BIOS print function or to a PC file.
- The emulator defaults to emulating a PolyMorphic Systems 8813 (disk-based system). If you're more interested in emulating a Poly-88, you can change this setting, either on the DOS command-line or in the POLYDEV.DAT file.

This release is packaged in several ZIP archive files as follows. For best results, unzip any (or all) of these ZIP files into the same folder (except as noted), and make that folder the default directory for the DOS window you're working in.

PM-main.zip

This ZIP file contains the minimum set of files needed to run the emulator. Use these files, along with the Emulator page on the website, to get started.

- **PM.EXE** – the PolyMorphic Emulator executable program.
- **POLYDEV.DAT** – its configuration file, described in the website under **Emulator**.
- **VGA.FNT** – downloadable font file to make the VGA character set resemble the one on Poly's video card.
- ***.PM** – several “virtual disk” (vdisk) images of Poly diskettes. **SYSTEM.PM** is the bootable disk. There are several other vdisks in the **Vdisks.zip** archive.

Vdisks.zip

This archive includes vdisks suitable for use with the emulator in its default 8813 emulation mode. The files are named *.PM and *.PMS. It also includes these subfolders:

- **RALPH** – disk images contributed by Ralph Kenyon. This includes PolyLetter Disk-of-the-Month (DOM) images, plus Ralph's own contributions (named PGL*).
- **DWIGHT2** – files contributed by Dwight Elvey. I have not worked with these; they may or may not be directly usable as vdisks in the emulator.

You can find a brief summary of each vdisk in the file named

_Vdisk file descriptions.pdf.

Tapes.zip

This archive contains *.CAS files, which are images of various Poly-88 cassette tapes. The CAS file format is described on the Emulator web page. It also includes:

- **DWIGHT** – files contributed by Dwight Elvey. Dwight created these from actual Poly tapes and stored them in *.IMG files. Those files were not directly readable by the Emulator, since they didn't match all the bytes expected by the Poly-88 tape loader. This folder also contains several notes from Dwight, and a C program I wrote to convert IMG to CAS files.

Manuals.zip

Copies of all the original PolyMorphic Systems manuals that I'm aware of. There may be others out on the web, on various retro-computing sites. Pay particular attention to the one named **_PolyQuickRef.pdf** since it summarizes many of the others.

Borland.zip

To build or make changes to the Poly Emulator you'll need Borland C++, version 3.1. It's now in the public domain. It's available at several places online, and also in this ZIP file. Unzip this file into the base directory of your DOSbox installation. This will create subdirectories **bc**, **bc\bin**, **bc\include**, and **bc\lib**. Make sure your DOS PATH variable contains **\bc\bin**. Note: you will not need this or **Source.zip** if you don't want to recompile the Poly Emulator program.

Source.zip

Source code files to rebuild PM and several utilities. The file **PM.PRJ** is a Borland C project file. After installing Borland C and these files, type the command:
BC PM.PRJ to open the project. The PM executable is built using the Compact code model. That selection is saved in the project file.

Website.zip

Unzip this into its own folder and open **index.html** in your favorite browser. This website contains nearly everything I know about the Poly, plus a lot of notes and photos. It also contains an earlier version of the Poly Emulator. For best results, you should use the emulator version packaged with this set of release files.

Note: This website appears online at <https://deramp.com/polymorphic-computers/>
This ZIP file is primarily a backup in case the online version goes away.

PU and PMU, utility programs

In the Source.zip folder is the PU program (source and executable). PU is a DOS program which allows you to do a few things with a Poly vdisk.

PU system.pm (to work with the vdisk named system.pm)

Once it's running, you'll see an Exec-style prompt, the dollar sign. Within the PU program you can use the following commands. They can be upper or lower case.

h or help	to see this list of commands
l or list	for a directory listing
en	change to ENabled mode (long directory listing)
disa	change to DISAbled mode (short listing)
t or type <file>	display a Poly file
get <file>	copy a Poly file to a PC file
size <len>	change the Poly vdisk size to <len> sectors
q or quit	exit the PU program

PU notes:

- Don't use it while the PM program is running. The Poly operating system will not know that you've changed the contents of a disk, leading to serious errors.
- PU doesn't work with Poly subdirectories – only with the main directory.

There is a similar program called **PMU.GO** which is loaded on your **system.pm** vdisk. It runs on the Poly side of things. Here is an example session with PMU.

```
$PMU
(PMU/1.2 - 10/12/90)
pmu: h
PMU commands are:
SHOW           displays drive assignments
CONNECT drive PCfile   changes drive assignments
SIZE drive nsec      changes drive size
COPY src dest        moves files PC<->Poly
PRINTER device      selects PC printer
DOS              brings up a DOS shell
FPB ON or OFF       floating-point emulation on/off
HELP             displays this list
QUIT             exits to Exec
pmu: q
$
```

PM command-line arguments

On the DOS command line, type **PM** alone, or follow it with any of these.

-norun Don't immediately start running the emulator. This is useful if you want to start by single-stepping.

-match Try to match the emulator's speed to an original Poly CPU's speed. This can make certain programs (games!) run more like they did on a Poly. This uses a delay loop after each opcode. It executes the delay loop a number of times, gradually homing in on a reasonable estimate after several seconds.

-matchNN Same as above, but fixes the delay loop count at NN.

-capture Capture all characters sent to the Poly screen via WH1 (wormhole 1) into a file called **wh1.out**.

-88 Start in Poly-88 emulation mode. The default is 8813 emulation. This affects the ROMs and screen memory address.

-tbasic Load the Tiny Basic ROMs. Only useful with the **-88** flag.

Any command-line argument not starting with a dash is assumed to be the name of a vdisk. The first such argument will be attached to Poly drive 1, then drive 2, etc.

POLYDEV.DAT options

These commands can be placed in the **POLYDEV.DAT** file, which is read when the emulator starts up. (Any line beginning with a semicolon is ignored.)

drive N filename Connects Poly drive number N to the vdisk file specified. Drive numbers can be 1 through 7.

printer filename Connects the Poly printer to the device or filename specified. **NUL** means “no printer.”

colors col1 col2 col3 col4 Specifies the emulator’s screen colors. Type each color name in lower case such as **white**, **black**, **blue**, etc. To make a foreground color bright, prefix the name with **b-** such as **b-white**. **col1** and **col2** are the foreground and background colors of the Poly’s 64x16 character screen. **col3** and **col4** are the foreground/background of the rest of the screen.

artc off [or **on**] Turns the adaptive real-time clock off or on. The default is **on**.

fpb off [or **on**] Turns the North*Star hardware floating point board emulation off or on. The default is **off**. If present, this board was used by Poly 8813 BASIC to speed up calculations.

font vga.fnt Loads the specified file into the VGA card (actual or emulated graphics hardware). If you leave this line in place, you’ll get a pretty good emulation of the Poly video graphics card. Comment it out with a semicolon if your DOS emulation doesn’t support VGA.

match on [or **NN**] See the **-match** flag in command-line arguments.

poly88 on Starts the emulator in Poly 88 mode, like the **-88** command-line argument.

Examples (8813 disk-based system)

PM Start the emulator using defaults listed in **polydev.dat**.

PM advent.pm Boot from the vdisk named **advent.pm**.

While the emulator is running, try these commands at the **\$** prompt:

L (upper or lower case) List the files on the current drive.

L 2 List the files on drive 2.

<2<MAZE Loads and runs the **MAZE.GO** program (which should be on drive 2). To start a new maze drawing, press **Tab**. To exit, press **Ctrl-Y**. Note that the filename MAZE is in uppercase, so you must type it that way.

To start the BASIC interpreter, type **BASIC**. While BASIC is running, the prompt changes to **>**. Poly's BASIC isn't exactly the same as some other dialects, but the simple things remain the same. Consult the BASIC manual for details.

To interrupt a running BASIC program, press **Ctrl-Y**. The prompt changes to **>>**.

To clear a program you entered, type **CLEAR**. The prompt changes back to **>**.

To exit from BASIC, type **BYE**.

Other things you can try:

Ctrl-Y Interrupts a running Poly program and returns to the **\$** prompt.

EN ENable changes the prompt to **\$\$**.

Ctrl-Z Opens the software Front Panel.

Ctrl-Alt-F1 Interrupts the emulator and returns to the **[pm]** prompt at the bottom of the screen. Then, press **q** and **Enter** to exit from the emulator.

Examples (Poly 88 cassette-based system)

PM -88	Start the emulator in Poly-88 mode.
P [or B] BASIC	Tells the emulator to load from a cassette tape. Follow this with the name of the program to load (such as BASIC). You will then be prompted, at the bottom of the screen, to enter the name of the file to be loaded. This file must match the name of the program you typed earlier.
BASIC.CAS	Here, you would type BASIC.CAS since the BASIC interpreter program is stored in that file.

The loader will read the CAS file and display block numbers as each block is loaded. If you have successfully loaded the BASIC interpreter, you will see

Poly 88 BASIC version A00. 44568 bytes free.

Now that you've loaded BASIC, you can load in a program that was created using Poly 88 BASIC. You'll need to know what program to load. Let's try the CRAPS game. Type the BASIC command:

LOAD CRAPS

Again you'll be prompted for the name of the CAS file to load. In this case it would be **CRAPS.CAS**.

If successful, the program will load and start running.

See the files in the **Tapes.zip** archive for more *.CAS files which are suitable for running on the Poly-88 emulator. Some require you to load BASIC first, as described above. Others load and run directly, as BASIC does. The file **_README1.TXT** in that archive goes into more detail.

To run PM in Poly-88 mode and load the Tiny Basic ROMs, use:

PM -88 -tbasic Then press **Ctrl-C** to enter Tiny Basic.

Suggested “to do” items for future development

Add “write to tape” for Poly-88 emulation so it can save programs to a file. This could probably be done by adding a space and “W” when entering a filename to the ’88 emulator. Currently the ’88 emulator can only read (load) tapes.

Final notes

If you read the source code files or notice some of the file dates, you’ll see that I started this project in 1990. I’ve learned a lot since then. I would probably do many things differently if I had started this project later in my career.

The biggest change has been this: I originally coded the main 8080 emulation loop in 8086 assembly language, since I was doing this on a 1990-era 80386 machine. Assembly was the only way to get reasonable performance from the emulator; plus, I wanted to learn about 8086 assembly.

Later, I decided to rewrite the assembly pieces in C. Newer, faster machines allowed this to work just fine. So, version 5.0 has no *.ASM files in the source code. It still runs much faster than the original Poly CPU card, which had an 8080 processor running at 1.8432 MHz. (Why this number? Because it could be divided down to produce common baud rates for the serial port.) And yes, there are a few remaining pieces of assembly language, mostly video BIOS calls, but these are now written as inline “asm” calls so the C compiler can handle them without a separate TASM pass.

I hope you’ll enjoy this bit of retro-computing.

Bob Bybee
rbybee@mindspring.com