# MSX and Coleco

**Simon Goodwin** tests nine emulations of MSX and Coleco systems.

## On the CD and DVD

On the LXF coverdisc you'll find MSX and Coleco emulators, PD demos and games, programming tools including a free C compiler, FAQs and English instructions for Zodiac.

Coleco and MSX computers have a lot in common, because they're based on cheap, commodity parts from the early eighties. They both use the eight bit Zilog Z80 processor, running at around 4 MHz, and square wave sound chips that deliver three channels plus random noise.

The graphics chip set comes from Texas Instruments, and also surfaced in British home micro designs like the Tatung Einstein and Memotech MTX. We have yet to see those emulated in Linux, but MSX or Coleco sources could be a good starting-point for such projects. MSX and ColecoVision have so much in common that there's even a ColecoVision emulator for MSX micros, and it runs on MSX emulators too.

Coleco is short for COnnecticut LEather COmpany, but the company had far wider interests in the seventies and eighties, ranging from Cabbage Patch dolls to computers. In 1975 they launched the Telstar Home Arcade, a console with an innovative custom chip. In 1982 they attempted to take the lead again with the ColecoVision, a ROM-cartridge based system. That sold more than six million in just two years.

The ColecoVision was unusually expandable by console standards and this led Coleco to develop a computer version, the Adam, bundled with a low-cost printer and cartridge tape drives. Linux can emulate this, too.

### MSX masterplan

MSX was a collaboration between Microsoft and mainly Japanese hardware manufacturers, aimed at establishing a standard for 8 bit home computer hardware and software. It combined games console and *CP/M* business computer parts, and was a massive success in Japan, South America and much of Europe. MSX also

thrived in countries where Cyrillic and Arabic characters were preferred to the ASCII acceptable in the UK and USA.

MSX machines were solidly built but otherwise rather bland. The name stands for MicroSoft eXtended. In conjunction with the Japanese publishers ASCII, the US micro monopolists devised a standard home computer, based on their own BASIC interpreter and off-the-shelf chips from Texas, General Instruments and Zilog. The original models were sold between 1982 and 1988.

MSX rights were licensed to around 40 manufacturers who came up with variants on the MSX standard, all compatible with the same software on cassette, cartridge and – eventually – floppy disk. Sony, Yamaha, Canon, Toshiba, GoldStar, Daewoo and Philips got onto the bandwagon, and managed reasonable sales worldwide, but the hardware did not sell particularly well in the UK or USA where it was perceived as overpriced and outdated compared with locally developed micros.

Even so, the original MSX standard spawned MSX2, with more RAM and twice the graphics resolution. This enjoyed modest support in Japan and Europe. TurboR models provided the swansong, with Yamaha OPL synthesis and twice the CPU speed, but 16 bit micros eclipsed those. In all well over 200 different (but compatible) MSX computer models were made, mostly in Japan but also in France, Brazil, Hong Kong and Korea.

There are five MSX emulators for Linux. Marat Fayzullin's fMSX is the most well established, now at version 2.5 for X. This has spawned two variants, *fmsxfan* and *svgamsx*. A portable MSX emulator has been written in Brazil, based on the CPU core from the British Spectrum emulator Jasper. Zodiac is a new MSX emulator from Japan, written in C, and seems the most actively developed at the moment.

## Emulator tests

### Zodiac

*Zodiac* has been tested on FreeBSD 4.3, Win98 and MacOS. It uses SDL so suits Linux and BeOS. The well-organised, modular code has parts distributed under the GPL or BSD licences.

My first attempt to **./configure** failed for want of *libmd* but after I added the *SDL developer 1.2.0* package from SuSE 7.2 I was able to configure and make version 0.7 without problems, ending up with a 420K executable 'zodiac/src/zodiac'. To run this I only had to copy MSX.ROM into the current directory.

You need a copy of the original MSX 'DISK.ROM' to use disk images, and 'MSX2.ROM' and 'MSX2EXT.ROM' or the versions with P in the names for MSX 2 or 2+ emulation – the same files as for fMSX. It tries to load MSX2 first, then MSX2+, then MSX1. Press F12 to quit.

At the moment most of the documentation is in Japanese but the distribution structure is logical and the source is quite readable. The changelog lists lots of progress in the last year, including support for the PSG and the tricky Z80 binary coded decimal DAA instruction. We were helped by one of the developers, Wataru Kitada, who sent a quick translation of the instructions. He warns that the current version is aimed at developers; updates appear regularly on SourceForge.

Options are generally similar to those for *fMSX*. The most important are **-a** and **-b** to select disk images, **-s** and **-u** to synchronise video and **-f** for the sample rate.

Sound works well, but graphics are small, in a fixed 256 by 192 pixel window, and buggy on many programs, most notably in MSX 2 modes. I tried a couple of dozen games and found that about half worked, including Sega's *HangOn*, Taito's *Elevator Action* and Broderbund's *Loderunner 2*, though the original dropped out into MSX BASIC. Namco's *DigDug* and *Galaga* kept rebooting MSX2, and missing sprites made *Pacman* and *Tank Battalion* unplayable. Big MSX 2 ROMs didn't fare well. *Ninja*, *Gryzor* and *Solid Snake English* kept rebooting. Several others played sound but got the wrong video mode, or locked up the emulator with a one-colour screen. It is early days for *Zodiac* but it already runs many programs well.

### fMSX 2.5

*fMSX* is one of seven emulators written by Marat Fayzullin, and the most mature of them all. It comes with brief generic HTML docs that say nothing about compiling the source for Unix.

*Mission*, the ColecoVision emulator, works on *fMSX* but only in MSX2 mode – in original MSX emulation it falls back to MSXDOS with a corrupted font. *fMSX* does not support, even in scaled

windows, the 512 pixel wide mode needed for 80 columns, but otherwise it is very compatible, though you may need to try several model variations.

*fMSX* compiles at once from the makefile on Linux, but won`t run unless you are in a 16 bit video mode or tweak a switch to specify 8 or 32 bits per pixel. **-DBPP24** is not listed as a valid option but did the trick for my 24 bit G400 desktop running *XFree86 version 4* on Suse Linux 7.2. Third party display drivers are available for other types of X terminal, and various disk transfer utilities.

*fMSX* allows direct access for MSXDOS disks and *rddsk* and *wrdsk* utilities permit reading and writing of files inside MSX disk images from the Unix shell. It also supports gzipped ROM and state files via *zlib*. The emulator launches directly into MSX BASIC 2.1 The left SHIFT key worked but my right one was ignored, whether I used a USB or PS2 keyboard. The Unix port of *fMSX* interprets the right shift key as CAPS LOCK and ignores the usual CAPS LOCK key. This behaviour does not match the port of *fMSX* to other systems, or my genuine Yamaha CX-5M MSX-1 system, and I found it rather irritating.



*Mission*, the ColecoVision emulator, running on *fMSX*, running on Linux.

### Performance

fMSX can run ten times faster than the real thing on my K6/500, though it soon slows to typical MSX rates if you boost the redraw rate and expand the window with **-scale**. It seems to spend more time updating the X window than emulating the micro, which is one reason why it has spawned variants from programmers without access to Marat's fast workstations.

Keyboard entry is tricky at extreme speeds as repeat and polling code is emulated as if running on a real 3.6 MHz Z80, so keypresses repeat almost uncontrollably on a computer running *fMSX* much faster than the real thing; at the other extreme inputs lag, though without getting lost, if you use a low **-sync** value. So *fMSX* does allow speed variation, but the main focus is on running software at authentic speed. On a fast Linux box try:
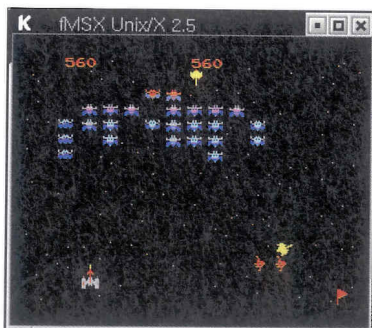
```
./fmsx -uperiod 1 -sync 50 -scale 2
```

Other switches select image files for two disk drives, one emulated tape drive, and assignments for the two joystick ports



**Sony Music Studio** was one of many MSX musical applications and add-ons.



The Coleco version of *Galaxian* on *fMSX*, thanks to *Mission*.



**MSXDOS** was the equivalent of IBM PCDOS for Z80 systems.