**Zilog Zodiac**

*Simon Goodwin scours the world for Linux-friendly Zilog micro emulators.*

This month we test a dozen emulators for home computers with Zilog's Z80 chip at their core, including the very British MGT SAM and Elan Enterprise, Tandy TRS-80 and Exidy Sorcerer, sundry Japanese Sharps, and continental micros like the P2000, Z1013 and KC85/4. The most capable of these previously unreviewed micros are Tandy TRS-80s and MGT SAMs, but others yet unmentioned put Zilog processors to good use.

**Zilog breakthrough**

Zilog's Z80 was the mainstay of home computers in the 1980s, along with the MOS Technology 6502, yet early eight bit computers favoured Motorola 6800 and Intel 8080 chips. Those large, established chip foundries did not foresee the mass market for low-priced micros, so staff left to start small companies to make the 6502 and Z80. Chuck Peddle worked on the 6800 before he set up the 6502 production line, and Federico Faggin founded Zilog, to make the 8080-compatible Z80, after working on Intel's breakthrough 4004, the first microprocessor.

The first three real home computers, aimed at mass-market users rather than hardware hackers, were the Apple 2, Commodore PET and Tandy TRS-80. Soon after Tandy adopted the Z80 that Zilog chip elbowed out 8080s from CP/M business systems and new home computers. Z80s run 8080 programs faster, with many extra instructions, and simpler hardware interfaces.

The CPC, MSX, Coleco, Master System, GameBoy and ZX computers were the most famous applications of the eight bit Z80 processor, as we've seen. Z80s also assist in NeoGeos, Megadrives and arcade systems. Before these, the TRS-80, and clones like Hong Kong's Video Genie, introduced a generation of coders and designers worldwide to Zilog's chip.

**Radio Shakers**

Electronics retailers Tandy, known as Radio Shack in the USA, were less wary of high-tech products than most chainstores, and accustomed to making hardware to fill niche markets among their technically-knowledgeable customers. Yet they were not sure that the TRS080 would take off, back in 1976 when the terms 'home' and 'computer' seemed contradictory to most people, and limited initial production so that if the machines didn't sell they could be used for stock-keeping in Tandy stores. In fact they sold more than 10,000 in the first month.

They picked the Z80 after prototyping machines with three other processors, including the 6800 and 8080; meanwhile the base specification grew from 1K RAM and 2K ROM to 4K of each, Most systems shipped with 16K RAM, expandible to 48K, and 12K of ROM, known as Level 2, with a 4K hole in the memory map for keyboard, screen and peripheral mapping.

This hole stopped the TRS-80 running the standard business operating system CP/M, although later Tandy models fixed that; the Model 2 was a CP/M business system while the Model 3 and Model 4 could run both CP/M and the original Tandy operating system, TRSDOS. Despite its nickname 'Trash-DOS', this was more sophisticated than CP/M or any 6502 micro DOS, with hashed directories, six security levels and access and update passwords on each file.

**XTRS 4.9**

XTRS is the definitive TRS-80 emulator for Unix. It delivers a TRS-80 with all the goodies we used to dream about twenty years ago, along with about 30 pages of detailed documentation. The command-heavy interface lacks icons, buttons or menus, but if you didn't like typing you'll miss many strengths of TRS-80 and Linux alike. Further TRS-80 programs, utilities and documents are plentiful online.

The GPL'd emulator is regularly updated and easy to build: unpack the archive with '`tar -xvzf`' and type '`make`'. Comment out two READLINE entries in '`Makefile.local`' by prefixing them with # characters if you lack the Gnu ReadLine extension - it's only used to ease editing of debug commands.

You need an image of the original system ROM, such as 12K Microsoft BASIC from the TRS-80 model 1 or Video Genie, to run the emulator. Specify its name on the command like after the '`-romfile`' switch. The ROM is patched so that when the TRS-80 is waiting for input it does not hog

the host CPU, though this does not fix programs that read the emulated hardware directly without calling the system.

**TRS Hardware**

The emulator supports the original 64x16 and later 80x24 character screen modes. It can emulate Microlabs HiRes graphics expansion hardware and Orchestra add-on synthesiser cards through OSS. Command line switches can scale or re-colour the display, tweak the keyboard and disk emulation, and match the font to Tandy, Genie or X Window System standards.

The XTRS real-time clock date-stamps files and is accessible through TIME$ in BASIC or the concurrent CLOCK command in DOS. The numeric pad simulates an Alpha Products joystick, and mouse emulation benefits a few Model 4 programs.

Original program and data cassette recordings at 250 (level 1), 500 (level 2) and 1500 (Model 3) baud can be read from 8 bit mono WAV files, with luck ('twas ever thus...) and even written via '`/dev/dsp`'. Serial port access uses '`/dev/tty`', and printer output is directed to '`stdout`', normally the launch terminal.

Unless you supply a Disk Operating System (DOS), you'll start in cassette BASIC at the 'Memory Size' prompt. Type an upper address like 32767 or just Enter to allocate all memory to BASIC. Later press F8 to quit, F9 to enter the powerful Z80 monitor in the Linux console or F10 to reset the TRS-80.

To boot DOS copy a floppy image to the file 'disk1-0'. The first number is the TRS-80 model, 1, 3 or 4; the big-box CP/M model 2 is not emulated, nor the Motorola-architecture TRS-80 'Color Computer'. The second digit is the drive number, 0 to 3 for 5.25" drives using single or double sides or density. Hardware permitting, links to Linux devices like '/dev/fd0' can supplant disk images.

Esoteric 8 inch floppies are emulated on drives 4 to 7, using a special LDOS driver. This later DOS replaced the TRSDOS Tandy bought outright for $2000 from Randy Reed with an extended device-independent system. For instance '`filter *ki using ki/dvr`' links in 128 characters of type-ahead, auto-repeat and screen-printing, and the KSM filter implements 26 shortcut phrase keys.

**Many DOSsers**

TRSDOS patches spawned a host of rivals including NEWDOS, VTOS, DOSPLUS and MULTIDOS. These exploited the extensible overlay structure of TRSDOS, adding many commands to DOS and BASIC and support for new hardware, rather than just Tandy's original 35 track single density single sided floppies. Capacity rose from 79 to 720K, so later formats can be read and written directly in Linux floppy drives. CP/M is also supported when running later ROMs.

LDOS is especially ambitious, running rings round 8 bit rivals like CP/M, FLEX or MSXDOS. Development continued into the 1990s without loss of compatibility. MINIDOS presaged Sidekick, offering DOS and printer commands through key combinations pressed while applications run. DO launches interpreted or compiled JCL (Job control Language) scripts, with label and parameter passing. PR/FLT and SPOOL implement disk or RAM-buffered print spooling with configurable code translation and page formatting. HELP gives brief command details.

I configured LDOS 5.31 system disks in drives 0 and 1 and the XTRS 'utility.dsk' with Unix linkage commands in drive 3, as LDOS searches all drives for commands, by default, and a model 1 does not allow double-sided disks in drive :3. 'mount' and 'umount/cmd' files on the utility disk switch images, as if you'd inserted or removed real TRS-80 floppies.

IMPORT and EXPORT allow files to be moved between Unix and emulator disk images. You can even run shell commands from within LDOS by prefixing them with UNIX, though their output is directed to the launch shell rather than the emulated TRS-80 display. The Linux command 'mkdisk' makes a blank unformatted disk image, so I made one and formatted it to 400K, then copied original TRS-80 files there from a Linux directory, with these commands in LDOS:

```
unix mkdisk disk1-2

format :2 diskname password 2 80 6
```

```
cd /home/simon/tandy

pwd

cd ..

unix ls -l

import firebird.cmd firebird/cmd:2
```

The import command copies 'firebird.cmd', moved to Linux from my ancient Video Genie, into 'FIREBIRD/CMD' in the new disk image. Typing 'FIREBIRD:2' started the game at unplayable speed. The command line switch '-autodelay' brings programs down to authentic speed, though it also slugs disk access.

The TRS-80 was well supported with applications as well as games - if you need an easy way to send and decode Morse code, for instance, TRS-80 emulation is well up to the job! The dozen Big 5 games are original TRS-80 classics, along with Scott Adams' Adventure International games, and UK releases from Molimerx and (confusingly) Kansas Software.

Given a ROM and programs, or appropriately formatted Tandy or CP/M disks,XTRS delivers a TRS-80 system that should bring a big grin to the face of anyone who remembers the original, but if you never used that you may find XTRS rather arcane.

**(( BOX - Euro-Zilogs ))**

This box discusses emulators for several Z80-based computers that were popular in Europe but less common in English-speaking countries where US and UK designs predominated. The popularity of open source on the continent means that these Euro-Zilogs are well supported by Linux.

P2000 home computers were made and sold by Philips in the 1980s. M2000 is the portable P2000 emulator. It comes with 4K system and 16K BASIC ROM images. M2000 offers 4K of video RAM and 32K of main memory, though you can add a ROM cartridge image and adjust the RAM size if you wish.

**Philips P2000**

Compile M2000 with '`make x`', '`make svga`' or '`make linux`' to build both. You need root privileges to make the SVGAlib version, and when I tried it bombed with a host of error reports about 'invalid asm statements' and 'forbidden register' usage. Switching off X86 code generation did not fix this, so I focussed on M2000x, the more portable windowed version. M2000 was developed on a 486 and there's no need for assembler to emulate a P2000 on a Pentium; 68K and PPC hosts are supported by a big-endian makefile option.

The X version should run in 8, 16 or 32 bit colour, but barfed till I downgraded my XFREE86Config from a 32 to a 16 bit mode. M2000 supports OSS audio, repeatedly described as USS in the readme, suggesting a surfeit of Star Trek. F5 toggles sound, while F11 and F12 set the volume; the '`-joystick`' switch enables Linux-compatible game controllers.

M2000x defaults to authentic P2000 timing, but can be switched with '`-cpuspeed`', in power-hungry or system-friendly ways selected with '`-savecpu`'. Other switches synchronise screen updates in 500x300 or 520x490 pixel windows; the SVGALib version prefers VGA or QVGA resolution. Cassette input and printer output can be configured on the command line or from the M2000.cfg file.

Optional debug features include ×a Z80 disassembler, breakpoint, tracing and single-stepping with a register display. The '`splittape`' command extracts individual files from serial tape images, and the Z80 disassembler is also available on the command line via '`./z80dasm`' followed by a code file name and optional start, end and offset. This is a good way to look through the supplied ROMs, but sparse documentation and software support limit the appeal of M2000.

**Z1013**

The Robotron Z1013 was a popular home computer in East Germany, and has an excellent Linux-specific emulator in SuSE's Professional distribution. The machine may be obscure, with its

mono display and U880 processor - actually an unlicensed Z80 clone - but the emulator is so well programmed and presented that it's well worth a look if you're merely curious about computing behind the Iron Curtain.

The original 2 MHz Robotron had a 32x32 character-mapped display, Commodore style preset graphic symbols as well as text, a 2K monitor and provision for up to 61K RAM. With 32 keys, cassette files and no inbuilt high-level language, the Z1013 attracted low-level programmers and electronics enthusiasts, but there's no shortage of games and hacks for the system. There is also a Java emulator, J1013.

**KCEMU**

The KC-85/4 was another German home computer, little-known in the UK but easy to test as KCEMU comes precompiled with demo programs. It ran as soon as I installed the Debian package of libg++27, opening an X window and displaying the demo disk image directory. The %menu command re-draws this and lists useful commands.

The Boulderdash game is playable even if you don't understand the German instructions. Arrow keys move the player around, amid fast monsters which you squish by dislodging boulders. Readers with slow machines will find this easiest to master. Keyboard handling is good, with upper-and lower case and sensible mappings.

F11 simulates power-cycling, F10 is a soft reset, and ESC mimics BREAK. The built-in BASIC offers almost 48K for programs. It runs at a reasonable speed but scrolling is slow. The CAOS operating system fills 12K of ROM, including pixel graphics extensions that seem to use a Spectrum or MSX-like attribute colour system; the old 8080 MBASIC uses a further 8K. Type BYE to return to CAOS and REBASIC to recover your program after using CAOS commands.

Like M2000, KCEMU uses Marat Fayzullin's portable Z80 interpreter, so ctrl-C in the launch console brings up a debug console with Z80 registers, disassembly, hex and ASCII memory dumps and facilities to write and read 256 ports. You can single step or set a breakpoint but there's no obvious way to change memory or register contents.

**(( End of EuroZ80 box ))**

**Sharp Start**

Sharp occupy the cool edge of Linux these days, with their Zaurus PDAs, but they entered the home computer market early with a range of Z80-based home computers. Their initial MZ-80 models were styled like Commodore PETs, with a built-in character-mapped screen and keyboard and the same typist-hostile orthogonal grid on the original MZ80K, though the CPU was from Zilog and most of the firmware came on tape.

Later Sharps were more modular and two of them - the MZ700 and MZ800 - have Linux emulators. These are based on Spectrum expert Ian Collier's XZ80 processor core, plus a GTK interface and MZ700 hardware emulation from NCEmu author Russell Marks.

MZ700em models a computer with 64K RAM, 4K ROM bootstrap and 4K of video RAM, and unfinished sound support. The display uses SVGAlib, and if you get past a blank screen you've done better than me. I did not persist as the MZ800 is hardware-compatible so its emulator can do all this and more, and compiled without problems.

Either way you need ROMs - 8K of bootstrap for the MZ800, plus 4K of system ROM and character set for the MZ700. The emulators do not include these but explain how to save and transfer them from a real Sharp. Most Sharp downloads come in MZ files, which consist of the characters MZF1 followed by the tape file header, then the file contents.

I didn't have much luck with those - perhaps because of ROM incompatibility - but the '`mzget`' program reads samples at CD rate from '`/dev/dsp`' and files the header and file contents as '`header.dat`' and '`out.dat`' respectively. This BASH one-liner converts those to MZF format: '`(echo -n 'MZF1';cat header.dat out.dat) >tape.mzf`'

Four Sharp emulators are on your LXF cover disk. MCE is a Java2 program that includes Sharp emulation, though it's relatively slow, like the Sharp-specific UMZE. These are worth trying if the

Linux-native programs won't work for you.

If you can access a Sharp MZ-80K, MZ700 or MZ800 family system, you should be able to get at least one of these emulators running on a Linux box, with ROMs and programs from the original. But the emulators do not come ready to run, so they're not much use without access to the real thing.

**(( End of Sharp Start (cf Sharp End?!) box ))**

**SAM's Story**

The SAM was developed by Miles Gordon Technology, a Spectrum add-on company that became a PLC in the quest to make the ultimate home micro. Alan Miles, a former Sinclair marketing man, teamed up with hardware designer Bruce Gordon, while Andy Wright wrote the system software

There are two Sam emulators for Linux. The generic Unix emulator XCoupé came first, followed by SimCoupé, now at version 0.90 on Windows and Linux, tested on PPC as well as X86 systems. XCoupé is obsolescent but SimCoupé has undergone major updates recently, trading modern host speed for cycle-perfect CPU and display timing, so it runs hacks and demos that previous SAM emulators could not.

Both owe their names to the marketing ploy that took the development name SAM (Spectrum Advanced Machine?) upmarket by tacking on a bogus sports-car suffix. Production Coupés dispensed with the ZX attribute grid, except in a legacy mode, opting for 128 colours, twice as many pixels per line, C64 and Atari-style scan-line interrupts, six-channel stereo sound, inbuilt MIDI, mouse, Kempston joystick and SCART connectors.

The processor is a 6 MHz Z80B, with 256 or 512K of internal RAM for code, BASIC or graphics. Two front-facing floppy bays support slim 800K drives and controllers; a rear Euroconnector allowed up to 4M fast RAM, serial and parallel ports, and eventually IDE hard drives.

Sam BASIC kept the fast turn-around of ZX BASIC, and full compatibility, but supported much larger programs. The first ROMs were buggy but version 2 was quite stable and version 3, now freely available, packs an amazing amount into 32K: strings as long as memory, block structure with locals and parameter passing, graphics scaling, compilation and palette interrupts.

**Installation**

Before you can build SimCoupé from source you must install SDL 1.2 and unpack, '`./configure`', '`make`' and install SAASound, the library that emulates SAM's Philips SAA1099 synth. SimCoupé also emulates SAM's one bit Spectrum compatible beeper and add-on stereo DACs, popular for sample playing though more CPU intensive than the synth. These manage a reasonable rendition of SoundTracker modules from the Amiga, though software filtering on a humble Z80 is inevitably crunchy compared with Amiga hardware. The built-in SAA1099 has much better pitch and volume resolution than earlier sound chips used on BBC, CPC, MSX, Spectrum 128 and similar systems, but it's still limited by preset waveforms and envelope shapes.

Unpack the SimCoupé archive similarly, which creates several subdirectories, with the Linux specifics in the SDL one. Type '`make all`' in that directory, wait a minute or so, then '`./simcoupe`' to launch the resultant executable, which loads the freely-distributable ROMs from the SDL directory. SAM BASIC has Spectrum strengths like syntax checking on entry, case-independent long names including spaces, continuation after editing without loss of data, plus automatic source formatting and tokenisation without forcing ZX-style keyword entry.

**Sim Controls**

Most SimCoupé keys are mapped to Linux equivalents, but SimCoupé uses Left-Alt for SAM Cntrl, and AltGr for SAM's Edit key as those are where the equivalents are on a real SAM. A click in the emulator window diverts mouse events to the SAM; press F3 to release the Linux cursor if you want to point outside the SimCoupé window.

Shift F9 saves the screen in concise PNG format, while F1 and F2 allow disk images in various formats, including .gz and zipped archives, to be selected and inserted into the two 800K emulated floppy drives. F12 resets the emulated SAM, or quits if pressed with Ctrl. F11 simulates the action of the back-panel BREAK interrupt button, and F10 overlays pretty configuration icons. SAM's original

function keys are mapped to the host's numeric keypad, so press 9 there to boot from the disk in drive 1, or type 'boot'.

The top of the emulator window shows the emulation speed in proportion to a real SAM, the host load of CPU, graphics and sound emulation are taking, and the amount left over. On my K6-500 emulating the original 6 MHz Z80-B took around 30 per cent of available CPU time but full-screen animation soon chews up the rest, with 50 Hertz refresh in an X window with four pixels for each original.

SimCoupé can restrict this by limiting the redraw size and rate; the SimCoupé 0.72 core was twice as quick, but less precise. F6 trims the render speed between 100 and 10 per cent of original frames. F8 toggles an impressive full-screen display which bypasses X, making it faster and smoother.

Graphics, sound, keyboard, mouse and floppy support are already excellent; with MIDI and hard drive emulation SimCoupé could outclass any expanded SAM setup, and the new GUI is easy to use as well as pretty. This is a very accomplished and confident emulator. Like SAM itself, it's been a long time coming but worth the wait because of the attention to detail and conceptual integrity that permeates SimCoupé.

**The Rest**

It seems that British-designed Memotech and Einstein micros lack Linux emulators, though their similarities to MSX mean most of the work has been done. But I found a couple of other Z80 emulators while researching this feature, and the archives are on your cover disk. 'jsorcerer' is a Java emulation of Exidy's Sorcerer, another early Z80 micro with a solid reputation. It arrived soon after the TRS-80 but did not benefit from Tandy's marketing clout and the resultant flood of software support.

'xent' emulates another British micro, the Elan Enterprise, which generated much interest but few sales. Enterprise ROMs and system files are freely available and the hardware was well-documented, though complex. 'xent' appears to have been cobbled together from XTRS sources, and the June 1995 version I found would not compile, even with BASIC and EXOS ROMs added in a ROM directory.

Clever hardware and software could not make up for a lack of marketing cash, and the Enterprise was eclipsed by 16 bit systems that coupled new custom chips with more powerful processors. Champion of these was Commodore's Amiga, long regarded as unemulatable because of its elaborate, tightly-coupled architecture. That's the focus of the next, penultimate, column in this series.

(( Links ))

Elan files: http://www.mumm.ac.be/~cammejpm/enterprise

Exidy files: http://www.trailingedge.com/exidy

KCEmu home: http://sourceforge.net/projects/kcemu

J1013 home: http://www.e-ilpa.de/j1013

JSorcerer: http://www.liaquay.demon.co.uk/Sorcerer

LXF emus online: http://simon.mooli.org.uk/LXF

M2000 home:  http://www.komkon.org/~dekogel

MZ80Em: http://prdownloads.sourceforge.net/mz800em

SAM demos: http://www.sskardon.fsnet.co.uk

SAM scrapbook: http://www.mono.org/~unc/coupe

Sharp links: http://www.sharpmz.computingmuseum.com

SimCoupé home: http://www.simcoupe.org

TRS80 UK: http://www.trs-80.co.uk

TRS80 files: http://www.trs-80.com

XTRS home: http://www.tim-mann.org

(( Captions - the black and white ones are chunky and can be used quite small, but it's important that at least one for each system appears, giving a feel for its looks ))

(( These eight are the most important ))

xTRS-Firebird.png "Kansas Software's Firebird managed surprisingly smooth animation on the chunky TRS-80 display"

xTRS-RobotAttack.png "The one-bit speech in Big Five's Robot Attack is as clear and threatening as ever in XTRS."

KC85-Boulderdash.gif "This Boulderdash clone comes on the KCEMU demo disk."

m2000-basic.gif "Philips P2000 was a typical 1980s BASIC home computer."

SAM-diskview.png "The SimCoupé disk requester is a joy to use."

SAM-Flash.png "The Flash paint package is a fine introduction to SAM capabilities."

SAM-OptionsGUI.png "Iconic options fine tune the SimCoupé configuration."

SAM-SoundMachineSeq.png "Revelation's Sound Machine edits tunes and waves for the SAM stereo synth."

(( These are worth including if space permits ))

xTRS-LDOS.png "LDOS was an uncommonly advanced 8 bit operating system."

xTRS-MeteorMission2.png "Meteor Mission 2 cloned Lunar Rescue in blocky TRS-80 style."

Z1013-digger.gif "Z1013 Digger breaks through the People's Anti-Fascist Defence Barrier." (( official GDR propaganda term for The Berlin Wall ))

Z1013-Wizard.gif "Wizard of Danger is one of the free games for the East German Z1013 emulator."

SimCoupe-PrinceOfPersia.png "Prince of Persia showed that SAM could compete effectively with 16 bit systems."

SAM-ManicMiner.png "The SAM version of Spectrum classic Manic Miner was an accurate but better looking port."

(( Hardware pictures - convey least information but establish some context - please do not use these or others from Future's collection at the expense of any from the 'most important' screens listed above! ))

mode1bw.jpg "Tandy squeezed the computer in the TRS-80 keyboard unit, bundling it with a 64 column mono monitor adapted from a TV."

mz-7311.jpg "The built in tape and plotter were the high points of this Sharp MZ700 variant."

sharp_mz821-good.jpg "Sharp's 821 model brought up the rear of the MZ range."

sam.gif or SAMCoupé.jpg "SAM's ergonomic front-mounted drives, inset keyboard and combined power, sound and video connection." (( I'll try to get you a photo of my own SAM to replace the hand-drawn GIF, but it's too dark to photograph that now and I don't want to sit on the copy any longer, having finally got a good SimCoupé ))