

WSpecEm Help

[WSpecEm v1.20 features](#)

[Copyright, program conditions and bug reports](#)

[Known problems and tips](#)

[WSpecEm user interface](#)

[How to get the latest WSpecEm version](#)

[How to read Spectrum tapes](#)

[Acknowledgements](#)

[comp.sys.sinclair ZX Spectrum FAQ](#)

[About the author](#)

[GNU General Public License](#)

WSpecEm v1.20 features

v1.20

- . Setup program for Windows'95;
- . Windows help file;
- . ZX's SNP snapshot file format;
- . BMP file saving;
- . joystick support;
- . keyboard template with key press generator;
- . Exact time delaying;
- . Windows interface improved;
- . program compiled for the 386 microprocessor.

v1.11

- . F4 as pause;

v1.10

- . definable emulator display frame rate. Essential to run it on slower machines;
- . emulator has delay configurable by user. Essential with the new routines;

v1.03

- . ghost keys (aka ghost closures -- v1.03 - they were documented but not introduced with v1.01 due to a lapse);

v1.01

- . save the current Spectrum screen as a PCX file;

- . handles .BLK tape files (a bug prevented it from working before this version);

v1.00

- . all the Z80 undocumented instructions [that I'm aware of, at least];

- . exact R register emulation;

- . exact T state cycle timing in Z80 engine;

- . kempston joystick as cursor keys with Fire as CTRL (if NUM LOCK off, the numeric keypad will work as a kempston joystick);

- . Sinclair joystick as numeric keypad if NUM LOCK on (8 up, 2 down, 4 left, 6 right, 0 and 5 are both fire);

- . recognises Windows'95 long filenames;

- . deals with ACH, PRG, SCR, SIT, SLT, SP, SNA, SNX, RAW, TAP, Z80, ZX and ROM spectrum related emulator files;

- . ED FB level loader trap supported (aka .DAT files);

- . files may be loaded by drag-and-drop, double-clicking or with the File open menu;

- . short-cut load and save keys, a' la Z80 (F2 - save and F3 - load, F5 - reset);

- . maps the Spectrum keyboard in the PC keyboard, with some extra pc-keys:

ESC : Caps-shift+SPACE (CTRL-BREAK in a Speccy)

BACKSPACE : same as Caps-shift+0 (DELETE in a Speccy)

LEFT SHIFT : same as Caps-shift

RIGHT SHIFT : same as Symbol-shift

TAB : same as Caps-shift + Symbol-shift (E Cursor)

+, -, *, / extended keys, operate as their Spectrum Plus counterparts;

- . pause, reset, nmi functions;

- . sound generation through PC speaker;

- . sound, flash, issue 3 or issue 2 control options;
- . capability of saving options;
- . WinG high-speed pixmap transfer;
- . resize to 1x1, 2x2, 3x3, 4x4 size for optimum speed under WinG;
- . poke menu;
- . easy of use;
- . enclosed DOS utility to sample Spectrum tapes in a PC as a .TAP file through a SoundBlaster or the printer port;
- . free, complete and commented sources available under GNU policy.

Copyright, program conditions and bug reports

Copyright 1996 Rui Fernando Ferreira Ribeiro

Emulator reports : wspecem@ipp.pt
Personal e-mail: rui@ipp.pt
WEB page: <http://www.idt-isep.ipp.pt/~rff-ribe>
Faculty account: i890478@dei.isep.ipp.pt

Snail mail:
Rui Ribeiro
Av. Carvalha, 280, 3D
4420 Gondomar
Portugal

[WARNING: the rff-ribe@csm.uwe.ac.uk account is no longer enabled.]

WSpecEm is a ZX Spectrum emulator that runs under Microsoft Windows. It emulates successfully a 48k model. It was developed initially over a 5-month period as a final year project, at the University of West of England, in Bristol, England.

This program is freeware, released under the GNU policy. You can use the program free of charge, but the author would appreciate a nice postcard of your homeland. I can read Portuguese, English, Spanish, French and Italian, so if this is one of your mother language(s), I hope you don't mind to use it.

Please send bug reports. Suggestions, contributions, technical discussions and job offers are more than welcome ([take a look at my CV](#)). When sending bug reports, please tell me both your Windows and the WSpecEm version number you are using.

I hope you do not get offended if I ask you to not send e-mail just to tell me you like the emulator. Also, no snapshots or software requests, please. And by all means, do not send binary pictures to replace the postcard (a few people did in the past).

I can reply to messages in English and Portuguese. My Spanish is very fragmented, I must undergo great pains to write in French and my Italian is virtually non-existent, I just manage to understand it.

All messages will receive my attention, but there's need for patience, as this is all uncompensated work and I need to make a living. If sometimes I am taking too long, please do ask me why. Snail mail is also a problem, it can take as long as 1 month, and arrives to my parent's house. I only get knowledge of urgent letters, the others are collected whenever I go there.

Known problems and tips

- . If the application is not working, first make sure you have properly installed the WinG drivers. If you did, make sure you have the multimedia drivers installed (a sure fire way of doing that is by installing the Media Player Windows utility) -- remember that I am talking about video drivers, in windows'95 you can also deinstall /install the multimedia video drivers, so make sure you get them installed -- if you still want to check it out, find a little AVI file and see if you can play it;
- . a 8 colour screen depth (256 colours) mode is recommended for optimum speed. If you want it faster and bigger, decrease the windows screen resolution;
- . run it under the menu sizes 1x1, 2x2, 3x3, 4x4 or you'll surely get very slow graphics transfer speed;
- . to get a faster emulation and smoother sound quit all applications that you can, disable screen savers and remove any windows background bitmaps;
- . to get a decent speed in the emulator, I would recommend a Pentium or equivalent processor:
- . and of course 8Mb or more for overall Windows performance;
- . if you still don't have a good speed, try decreasing the frame n/50 option at Options->Speed menu, and disable the flash option at Options menu;
- . if by the contrary, you have too much speed, put frame n/50 option at Options-Speed menu to 1, and if that doesn't still help you, go to Options-> Speed menu and start increasing the Delay number;
- . double-clicking in a .SCR file will not do any good, since WSpecEm will reset itself upon boot;
- . If you're using 16 colours, sometimes the subset of the palette chosen by Windows for WSpecEm is flawed. If this happens, you'll have to shutdown and boot Windows again. Avoid 16 colours altogether if possible, it's too slow;
- . If a key gets stuck, press it again. That should solve the problem. A key can become stuck if you press it down, select the Windows menu and release it before leaving the menu;
- . Sometimes, when you are using the extended keys, you can get only one key instead of the combination of both (e.g. when pressing DELETE you get 0). This is a Spectrum ROM related bug and was also present in the real machines;
- . If you have problems with stuck SHIFT keys trying to get the E cursor, use TAB instead to get the E cursor -- this problem is more likely to occur under Windows 95 than under Windows 3.x;

- . if you load a ROM, it will be used until you load another ROM into the emulator;
 - . if the emulator presents a message it couldn't load a ROM, that's because the spectrum.rom file is not on the directory of the WSpecem.exe file. Following that message you can load your own ROM file;
 - . to load a .TAP or .BLK contents (binary image files of Spectrum tapes), do File->Open to the corresponding file, and then tape LOAD"" (j SHIFT-P SHIFT-P ENTER).
-

WING NOTES FROM MICROSOFT

Known Bugs And Limitations

The following are known problems with or useful tidbits of information about WinG version 1.0.

- On Windows 3.1, WinGBitmaps must be 8 bits per pixel and must be created with full 256 entry color tables.
- WinGDCs are NOT palette devices. You must change their color tables using WinGSetDIBColorTable, not SelectPalette.
- WinGBitBlt and WinGStretchBlt only support bltting from WinGDCs to the screen.
- Using BitBlt and StretchBlt to blt from one WinGDC to another can be very slow when a clipping region has been selected into the destination.
- WinGBitBlt and WinGStretchBlt may return different values than StretchDIBits for identical blts.
- A few GDI APIs do not work correctly with WinGDCs:

StretchDIBits will not blt 24bpp and 16bpp DIBs into an 8bpp WinGDC.

FloodFill with a NULL brush draws incorrectly

FloodFill outside of the bounds of a WinGBitmap can flood the entire image
Brushes created with CreatePatternBrush with a WinGBitmap fault when selected into a WinGDC on Win3x - use CreateBitmap(8,8,1,8,0)

DrawIcon will crash

WinGBitBlt and WinGStretchBlt will not always blt to the correct location when you have changed the Viewport and/or Window origins using SetViewportOrg or SetWindowOrg.

- You cannot change the origin of halftone brushes created by WinG.
- Noticeable timing differences have been found while running the WinG profiler on a computer connected to a network. For accurate results, disconnect your computer from the network the first time you run a WinG application. After the profile is complete, you can plug the net in again. NOTE: Microsoft is not liable for any damage you may incur by incorrect handling of your computer hardware.
- WinGBitBlt and WinGStretchBlt use a slightly different colour matching algorithm than StretchDIBits when bltting an 8-bit image to a 4-bit planar display such as a standard VGA. Mixing WinG and StretchDIBits on these displays may produce odd results.
- WinG relies on the mmsystem timer drivers to determine display performance. If mmsystem.dll and timer.driv are not installed correctly, the results of the performance test may be

incorrect. mmsystem.dll should appear on the drivers= line of the [boot] section of system.ini, and timer=timer.drv should appear in the [drivers] section of system.ini.

- WinG version 1.0 does not yet use standard DCI because of time constraints.
- "Just In Time" debuggers install a fault handler in a chain along with the WinG display performance profiler. If your debugger reports a fault during the WinG display performance test, pass the fault on to Windows instead of invoking your debugger.

Driver-Specific Problems

WinG depends on Windows display drivers written by independent hardware manufacturers for much of its speed. Bugs or performance problems in third-party display drivers may cause problems in WinG. In many cases, the video card manufacturer has already fixed the bug, and upgrading your display driver will often clear away problems.

There are some specific "bugs" in display drivers of which you should be aware. The list below is not intended to slight the manufacturer of any particular card or driver.

A list of stress-tested configurations is available on the CompuServe WINMM forum and ftp.microsoft.com.

Some names in this list are trademarks of the respective manufacturer.

- Early drivers for Diamond Viper cards included a "Power Palette" option that is no longer supported by Diamond. They recommend that you upgrade your drivers if you have this option. WinG may be slower when power palette is enabled.
- IBM no longer supports the IBM ThinkPad 720c. There are some problems using WinG with the ThinkPad 720c display drivers.
- Cirrus drivers before version 1.43 have many known bugs which have been fixed in the more recent drivers. Be sure to upgrade your drivers if you are still running with this version.
- Some ATI drivers offer a "Crystal Fonts" option. Turning Crystal Fonts on in 8-bit modes sets up a non-palletised driver that can slow WinG significantly.
- The ATI mach8 Radical drivers cause a number of problems in both WinG and in Windows with some versions of the ATI chipset. Be aware.
- The ATI VGA Wonder drivers (W31-*.drv) will crash during a call to StretchDIBits in the profiler. Users can run the SVGA256.DRV driver that shipped with Windows.
- Many miro Crystal drivers have problems with StretchDIBits, so they crash during profiling.
- Early ATI Mach 32 PCI cards have a hardware timing problem and will hang while blting. ATI will replace these cards for no cost.
- WinG is incompatible with the #9GXE "TurboCopy" mode. Use the #9 control panel to disable TurboCopy (it is off by default).
- WinG uses a GetPixel to synchronise with display hardware when writing directly to the screen. The ATI Mach 32 driver's GetPixel does not work properly, so it is possible to use GDI to draw to the screen, then use WinG to blt to the screen and have them overwrite each other. Be careful mixing GDI drawing commands and WinG blts to the display.
- The Orchid mmtllo.drv driver for the Prodesigner IIs has duplicate system colours which prevents applications from getting an identity palette and greatly reduces the WinG blt speed. A workaround is to set SYSPAL_NOSTATIC mode or use standard the Tseng ET4000 drivers instead of the mmtllo drivers.

A Note on Speed

WinG is designed to be the absolute fastest way to blt DIBs under Windows. The goal, achieved in many cases, is to blt at memory bandwidth to the display device. On most 8bpp devices, if you use the recommended DIB format (returned by `WinGRecommendDIBFormat`) and set up correct identity palette, you should get 1:1 blt speeds comparable to `BitBlt`, which blts device dependent bitmaps (DDBs) to the display. The `timewing` sample application will show you various blt speeds on your display.

The `WinGRecommendDIBFormat` API will tell you whether to use top-down or bottom-up DIBs for fastest unclipped 1:1 identity palette blts. If you plan on using other types of blt (such as stretching or complex clipping), your application may want to time top-down versus bottom-up blts itself at run time. See the WinG help file for more information.

WSpecEm user interface

WSpecEm operation keys

Windows menu

WSpecEm operation keys

. cursor keys as joystick with Fire as CTRL (if NUM LOCK is off, the numeric keypad will work as a kempston joystick, other way as a Sinclair joystick) (8 up, 2 down, 4 left, 6 right, 0 and 5 are both fire);

. short-cut load and save keys (F2 - save, F3 - load, F5 - reset);

. Key F4 is pause;

. maps the Spectrum keyboard in the PC keyboard, with some extra pc-keys:

ESC : Caps-shift+SPACE (CTRL-BREAK in a Speccy)

BACKSPACE : same as Caps-shift+0 (DELETE in a Speccy)

LEFT SHIFT : same as Caps-shift

RIGHT SHIFT : same as Symbol-shift

TAB : same as Caps-shift + Symbol-shift (Cursor E)

+, -, *, / : extended keys, operating as their Spectrum Plus counterparts.

Windows menu

File menu

Shortcut Name	Activity
Alt-Fo Open F3	Opens a snapshot or makes the emulator point to a tape file for later use;
Alt-Fs Save	Saves the current emulator state under the name of the current snapshot;
Alt-Fa Save as F2	Saves the current emulator state asking for a name first;
Alt-Fr Reload	Reloads the current snapshot on memory;
Alt-Fx Exit	Exits the emulator

Options menu

Alt-Oo Sound	Enables or disables sound during the emulation.
Alt-Of Flash	Enables or disables flash. Flash can be disabled to gain speed.
Alt-Oi Issue 3	If ticked the emulator is emulating Spectrum issue 3, else is emulating issue 2. There are games that only work under issue 2 and others that work only under issue 3.
Alt-Oz Screen Size	

Alt-Oz-1 1x1size Client area of emulator will be 256x192

Alt-Oz-2 2x2size Client area of emulator will be 256*2x192*2

Alt-Oz-3 3x3size Client area of emulator will be 256*3x192*3

Alt-Oz-4 4x4size Client area of emulator will be 256*4x192*4

Alt-Os Speed Allows the user to customise the speed of the emulation.

AltOj Joystick off - No joystick present or not used
kempston - Joystick is emulating kempston
Sinclair - Joystick is emulating Sinclair
Cursor - Joystick is emulating Cursor

AltOk Keyboard template keyboard image and keypress generator w/ mouse

Alt-Os Save Save option at wspecem.ini, on the default windows directory.

Misc menu
=====

Alt-Mp Pause Pauses / unpauses the emulator.
F4

Alt-Mo Poke Writes a byte on a specified address.

Alt-Mr Reset Applies a reset to the emulated Z80.
F5

Alt-M NMI Asserts the NMI signal on the emulated Z80.

Help menu

=====

Alt-Hc Contents Accesses the help file.

Alt-Ha About Application credits.

How to get the latest WSpecEm version

The latest version will be always uploaded to:

<http://www.idt.ipp.pt/~rff-ribe/wspecem.html>

<ftp://ftp.idt.ipp.pt/pub/sinclair/emul/wspvXXX.zip>

<ftp://ftp.nvg.unit.no/pub/sinclair/utis/pc/wspvXXX.zip> where **XXX** stands for the corresponding version.

Unfortunately, the nvg site is almost always too busy. Try to use always regular ftp services as WWW transfers are slower. If you still can't use it properly, try to do ftp by e-mail. To see this directory, send an e-mail to:

ftp@ftp.core.ipp.pt or *ftpmail@decwrl.dec.com* with the following message:

```
open ftp.nvg.unit.no
cd /pub/sinclair/utis/pc
ls wsp*
quit
```

Be patient, this server can take as long as 24 hours to process your request. After you receive the name of the files, send this message to retrieve it:

```
open ftp.nvg.unit.no
bin
cd /pub/sinclair/utis/pc
get wspvXXX.zip
quit
```

You'll receive a message with the corresponding file uuencoded. If you want to know more details about ftp by e-mail use, send an e-mail with only the word *help* in the body of the message.

Alternatively, if you don't have Internet access, I will mail you the last version, if you send me £5 (five British pounds) or if you're in western Europe, Canada or USA the equivalent in your own currency. Please also specify which operating system you're using on the request. See here my address ([about the author](#)).

Alternatively, you can try this BBS in Portugal:

MicroRun BBS:	+351 (0)2 5104669	28800 baud
	+351 (0)2 5106214	14400 baud

Tape2Tap

INTRODUCTION

INSTRUCTIONS

THE CABLE

TIPS FOR READING TAPES WITH TAPE2TAP

KNOWN PROBLEMS

INTRODUCTION

Tape2Tap is a MS-DOS program that digitises spectrum tapes in a PC through a SoundBlaster or the printer port. This program was done as a preliminary preparation to this Spectrum emulation you are using now. It is compatible with DOS versions up from DOS v3.0 and with Windows'95.

Tape2Tap works under DOS, independently of the processor speed, since it uses the PIT to count the time, instead of depending on burning machine cycles for delays. The program will also auto detect a SoundBlaster and use it. If you do not have or do not want to use it, you can also digitise data through the printer port.

The data will be saved in the file W1.TAP at the current MS-DOS directory. If the file does not exist it will be created, else data will be appended to it. In case of a loading error the user is notified, the faulty block is not written to disk, and the tape has to be rewind to the beginning of that block. When Tape2Tap reads a tape header block successfully, it displays the familiar 'Program:' or 'Bytes:' message, followed by name stored in the block. Between tapes blocks, all data is output to disk, so I have no idea how it will behave with diskettes.

Care must be taken not to leave the program while a block is being loaded, because it will not become saved. The option 1 disables Spectrum like CRC checking, because one of the tape protection methods was to save a block with an incorrect CRC. Of course, it must be enabled again as soon as possible, to check for possible tape errors.

Tape2Tap manages to load almost all tapes at regular or turbo speed. This strength is also its weakness: to cope with this, I cannot assume transition times, so the program is more vulnerable to tape errors. There is only one kind of protection that Tape2Tap cannot cope: a silence in the middle of a block.

I did what I could with sound sampling, but I think very noisy tapes will not get much chance with this program: if your tape is original you are in your lucky day.

Of course, if you digitise turbo tapes, the emulator must recognise them or you must hack the program. Well, this program was my answer for not have to hack the program. I am hoping that a future version of WSpecEm will handle these special tape formats.

INSTRUCTIONS

You call Tape2Tap and:

. if you have a SoundBlaster, it will autodetect it and will try to use it (if for some reason you do not want to auto detect SoundBlaster use the /ds option: disable SoundBlaster detection)

. if you don't have a SoundBlaster it will try to digitise data from the Printer Port

The data will be saved in W1.TAP (in Z80 format) at the current directory. If W1.TAP does not exist, it will be created. If not, data will be appended to it. If a block gives an error it will be not saved, so you have to rewind and try to load the block again. (a recorder with a counter will be handy.)

You notice that something is being read by Tape2Tap as the old ways: the colours of the stripes change.

When Tape2Tap can read a block successfully, it will the display the familiar 'Program:' or 'Bytes:' message followed by the corresponding name. so in the very first block of the tape you should get no errors and the name right...if you did not you must try some different tape volume or something.

Inside Tape2Tap you can use 5 keys:

- ESC -- exit program (don't do that, unless you are finished with the current block)
- 'S' -- turn PC speaker on (to hear this sounds forgotten long ago by humanity)
- 'N' -- turn PC speaker off (the original Spectrum ought to do that!)
- '1' -- disable CRC checking (use with extreme care! You are in fact disabling error checking...but there are software protections that change the CRC)
- '2' -- enable CRC (when you skip the offending block, restore this as SOON AS POSSIBLE!)
- 'P' -- starting using the printer port instead the SoundBlaster as the inputting device

THE CABLE

If you are fortunate enough to have a SoundBlaster and still your Spectrum cable, you can go pick it, and plug it to SoundBlaster's Line In the corresponding place in your tape recorder (Line Out, Ear).

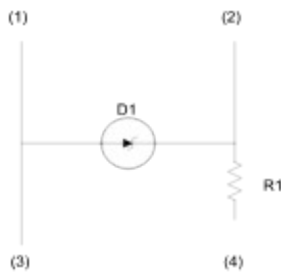
Anyway if you have a SoundBlaster, it suffices to get a cable to connect the Line IN to your tape recorder's Line Out or Ear.

To the more unfortunate people (like me):

* You can buy one tape adapter from BG services, UK; it's compatible.

Or you can build one yourself:

- . you'll need a 25-pin D-type connector (just like the one that plugs your printer to your computer)
- . a jack connector to plug to tape recorder
- . some cable



R1 - 100 Ohms

D1 - 4.7V Zener diode

- (1) - Parallel Port Pin 13 (GROUND)
- (2) - Parallel Port Pin 10 (BUSY)
- (3) - mono jack little pin
- (4) - mono jack big pin

TIPS FOR READING TAPES WITH TAPE2TAP

Tape2Tap is a bit intricate program, and got several deficiencies. In despite of that, it works perfectly with tapes on good conditions. And I'm very proud of it, despite it's limitations, since if you remember how difficult it was to load programs on the real machine, imagine doing it on a PC machine.

First of all I recommend you to use a stereo equipment of good quality with a very low noise ratio. Immediately after reading the header, the message program: Bytes:, or whatever, must appear. It's a good sign if you see the stripes, it means you are getting data. Bear in mind that you'll get better results if you have no memory managers active in memory.

Unfortunately, the volume requires a bit of experimentation. Try it a about 3/4 below the maximum, or try it at the maximum, and going a bit down after each try. If all fails, try another tape.

Please, note that Tape2Tap will also read programs saved with strange schemes, devised for fighting piracy in the old days, but the emulator can't cope with this at the moment. At my home, I succeed to read about half of my tape collection, but maybe you're not so lucky.

I must add that trying to read tapes is a very frustrating experience. If you really want a game, I suggest you first search for it on the net, and only on a last resource try to read it from a tape.

Finally, SoundBlaster for input maybe more convenient, but you will find better results with the LPT port.

KNOWN PROBLEMS

- . The mixer and your SoundBlaster will be in a different state when you leave Tape2Tap. The only workaround was not to do the sound card auto detection and ask its address, but I prefer the current approach. If you are calling it from inside the Windows' 95 shell, you won't have this problem.
- . Tapes with a lot of noise won't do very well with this current version. Try to get Warajevo from <ftp.nvg.unit.no>, they wrote a better tape digitiser based in the Tape2Tap code.

Acknowledgements

Designing a computer emulation is always difficult, and trying to implement all its unknown aspects even a greater challenge. The author would like to acknowledge here:

- all the people, who pointed a few problems and gave new ideas;
- Filipe Silva, who lent me all his Spectrum tapes, gave many suggestions through the emulator early stages, used it from the very beginnings, and exchanged Spectrum snapshots with the author;
- José Caetano Silva, for all the friendship, Windows books, discussions and contributions;
- Pedro Oliveira, for all the books, magazines and his sound card just when they were needed;
- Alexandre Ulisses, for the companionship and his PC hardware book. Without it, the utility to read Spectrum tapes would have never had seen the light of the day;
- Gilad Raz for several bug fixes (hopefully, WSpecEm's next version will be released as a jointly work by both of us);
- Axel Quentin for the floating keyboard and joystick support;
- Damien Burke for the WSpecEm icon, SLT definition, several suggestions, bug reports and the CSS faq;
- Samir Ribic for Warajevo, excellent suggestions and using TAPE2TAP's code;
- Ian Collier for our discussions in the Spectrum newsgroup;
- Francisco Cotrina for its special interest, suggestions and bug reports;
- Blood for its permanent interest, bug reports and work in the nvg site;
- James McKay for X128, problems reported and several snapshots;
- Alvin for its tech data on the TS Spectrum family and support;
- Gerton Lunter for Z80, several texts and a benchmark snapshot;
- Craig Duffy and Nigel Gunton for their continued support and advice during my stay at UWE;
- to all the Spectrum and other emulators community;
- to all my friends and my family for putting up with me;
- finally to my girlfriend, for everything we shared together.

And at last but not least, to Sir Clive Sinclair for producing such an enjoyable machine.

ZX Spectrum's ROM is copyrighted from AMSTRAD.

WING copyrighted by Microsoft.

This program and the accompanying dissertation were submitted to the Faculty of Mathematics and Computers, The University of West of England, as part of a final year project as an Erasmus student.

comp.sys.sinclair Sinclair ZX Spectrum FAQ

v3.02 (June 11 1995)

Maintained by Damien Burke
email: D.M.Burke-CSSE94@cs.bham.ac.uk
(until June 14th 1996; check <http://www.cs.bham.ac.uk/~dmb/mail.html>
for a new email address after that date)

You can find the most recent version of this document at the following
World-Wide Web (WWW) address: <http://www.cs.bham.ac.uk/~dmb/speccy/faq.html>

Please send your additions and corrections to the email address above. Much of this FAQ is the FAQ as previously maintained by Marat Fayzullin, for whom all old computer fans should give daily thanks and worship ;-)

Copyright note: if you're going to copy this, do it right. OK? I don't care if you stick this on a CD, print it and make paper aeroplanes from it, whatever -just don't modify it - tell me if changes need making.

CHANGES

CONTENTS

- I. FREQUENTLY ASKED QUESTIONS - and the answers!
- II. NET ADDRESSES - ZX-Spectrum-related Net addresses
- III. EMULATORS - comparison of existing ZX-Spectrum emulators
- IV. WHERE IS...? - what happened to famous ZX-Spectrum people
- V. FILE FORMATS - descriptions of emulator file formats
 - 1. Snapshot files - memory snapshot file formats
 - a) .SLT
 - b) .SNA (48Kb)
 - c) .SNA (128Kb)
 - d) .SP
 - e) .ZX
 - f) .Z80 (all versions)
 - 2. Tape files - virtual tape file formats
 - a) .TAP (and .BLK)
 - 3. Other files - other emulator file formats
 - a) .MDR

b) .SCR

- VI. TECHNICAL INFORMATION - info on ZX-Spectrum's hardware
1. Z80 CPU - undocumented opcodes and weird features
 - a) CBh opcodes
 - b) FDh and DDh opcodes
 - c) EDh opcodes
 - d) R register
 - e) Undocumented flags
 - f) IFF1 and IFF2
 2. Hardware - ULA and I/O ports
 3. Interface I
 - a) Port E7h
 - b) Port EFh
 4. The 128K Spectrums
 - a) Memory
 - b) Keypad
 - c) Sound Chip

- VII. PINOUTS - pinout details for various hardware
1. ULA
 2. AY-3-8912
 3. Keyboard layout

- VIII. ACKNOWLEDGEMENTS - thanks etc.

CHANGES

Changes since v3.01:

- 1) Emulators list expanded and updated.
- 2) Where is...? list expanded and updated.
- 3) Some other minor additions/corrections.

Changes since v3.0:

- 1) Updated net addresses to give descriptions to/remove all but one of the 'No description yet' URLs.
- 2) Corrected address of SAM Coupe FAQ.
- 3) Expanded description of SPECEMU emulator - thanks Rui.

Changes since v2.9:

- 1) Net Addresses list expanded.
- 2) Emulators list expanded.
- 3) Where Is...? list expanded.
- 4) File formats section expanded.
- 5) Some information on the 128K hardware added.
- 6) More Spectrum models listed.
- 7) Various minor additions/modifications.

As I am a university student and about to take a year out to go back to the real world this may be the last major update of the FAQ for some time. My university account will become defunct some time around September 1996, and I am unlikely to be able to read any mail after June 14th 1996, so email to the address listed above will probably bounce or remain unanswered. I should have a new address by September 1996; I will make a link available to the new address on the following web page: <http://www.cs.bham.ac.uk/~dmb/mail.html>

FREQUENTLY ASKED QUESTIONS

1. Just what is this newsgroup for?

This group's charter (from its original formation in December 1993) is to provide a forum for the discussion of the various brainchildren of Sir Clive (born Clive Sinclair) and related computers such as the Jupiter Ace, TK and TS series', SAM Coupe and Thor, software, peripherals, emulators for them, programming them; playing those wonderful games (6031769, anyone?); and anything else which interests the Sinclair community. The group is not moderated.

In practice, most discussion is Spectrum-related, but there's a fair bit of SAM Coupe, QL, ZX-81, etc. chat as well. This FAQ only covers the Spectrum, other documents being available seperately:

The QL FAQ at <http://ww.uni-mainz.de/~roklein/ql/>

The SAM Coupe FAQ at <http://www.soton.ac.uk/~tsp93ma/Coupe/FAQ.txt>

This FAQ covers technical information, emulator information and resources and does not attempt to cover Spectrum games and history. There is a game FAQ available -see question 18.

2. Can I post binaries to this group?

NO! The word 'binaries' does **not** appear in the newsgroup title, and this is a subtle indication that posting binaries is going to make you very unpopular very quickly. Many people have to pay for net access, and you're costing them money if you post binaries, which are substantially larger than most posts. You may also cause news providers to stop carrying the group if binaries are seen to be posted in it. If you want to distribute something Sinclair-related, upload it to an FTP site or put it on a web page and post the **address** of the site.

3. Is there a Spectrum emulator for <whatever machine>?

Probably. See section II, Emulators, below. It is split up by machine type, so you should be able to find one you can use. Of course, if you can't get one for your machine, you could always write one... ;-)

4. Where can I find <whatever game>?

The game you're after is probably available somewhere on the net. The gopher server at <gopher://gopher.nvg.unit.no> enables you to search for games (web interface to it at <http://www.nvg.unit.no/spectrum/search.html>), but an even better way to search is to use Hynek Med's snapsearch engine, located on the web at <http://drson.vse.cz/snapsearch>.

Several CDs full of games can be had:-

Speccy Sensations II: Warm Silence Software at:

<http://www.comlab.ox.ac.uk/oucl/users/robin.watts/wss/index.html>

Emulator Heaven:

Phone Mike Mee in the UK on (01745) 591704 after 6pm or write to "Anncott", Hylas Lane, Rhuddlan, Clwyd, LL18 5AG, UK or phone 'TortyFoo' also in the UK on (01922) 616324 (7pm-10pm Weekdays/ 10am-10pm Weekends) or email: emulators@tortyfoo.demon.co.uk

Bear in mind that these CDs contain a large amount of copyrighted software so they are technically illegal!

If none of those turn up the game you're after, try asking on the newsgroup, or (shock) going to a second-hand shop and digging around in their boxes of old tapes.

5. Is it legal to use ZX-Spectrum ROM images?

Yes, it is. Amstrad, who own the copyright for the ROMs, allow free distribution of ROMs for emulation purposes, although they often apply stipulations such as not charging money for software that includes these ROMs. They also allow free distribution of electronic versions of Spectrum hardware manuals.

6. Is it legal to use snapshots (i.e. games etc.)?

Formally, using snapshotted game without actually owning a copy of it on a tape is a *copyright violation*. Nevertheless, ZX-Spectrum games are not being sold any more and using snapshots doesn't harm copyright holders much, if you don't make profit from it and don't distribute snapshots on a large scale. Some companies definitely do not like to have their games distributed (e.g. Ultimate/Rare), whereas others have given explicit permission (e.g. Vortex). A list of such companies/individuals is at

<http://www.cs.bham.ac.uk/~dmb/speccy/permits.html>.

7. I have the game I was after, but in a format my emulator can't load. How do I convert snapshots between various formats?

Most snapshots can be converted into one another using SPConv program by Henk de Groot [hegr@ensae.ericsson.se] and myself. An MSDOS version of this program is supplied with the JPP emulator (source included), but is a rather old version. A more up to date version (v1.10) is available from the NVG FTP site: <ftp://ftp.nvg.unit.no/pub/sinclair/utills/generic/spcnv110.zip>

SPCONV converts files between following formats: .SIT,.SNX,.SNA,.Z80,
.PRG,.SP and RAW.

Also, the registered version of Z80 contains a converter program, and there is a program called Butcher, written by Marat Fayzullin (available at the NVG FTP site) that can split .ZX (or .SPC) files (used by the "Polish" emulator) into a series of .header/.bytes files for use with the Spectrum v1.7 emulator on the Amiga.

8. I'm trying to convert a .Z80 file into .SNA format using SPConv and it doesn't work. Why?

You are probably using SPCONV v1.05 which came with JPP distribution.

It has problems converting .Z80 files created by Z80 v2.01+. Get SPConv v1.10 (URL listed in question 7 above). It works.

SPConv cannot currently handle 128Kb snapshots, but hopefully the next version will.

9. I've got a snapshot that won't load into Z80 - it just gives an error message saying 'plusdrom.bin not found' and exits to DOS. How can I get it to work?

The snapshot was saved by someone with a registered copy including +D emulation, in +D mode. The game doesn't actually *need* the +D ROM, so you can use DEMGT:
<ftp://ftp.nvg.unit.no/pub/sinclair/utills/generci/demgt.zip> to modify the snapshot so it isn't in +D mode.

10. What is the difference between the various ROM-files available for use with Spectrum emulators?

SPECTRUM.ROM - Spectrum ROM, exactly the same as in the original Spectrum.

GROOT.ROM - Same ROM, but with lots of bug fixes and extensions by Henk de Groot [hegr@ensae.ericsson.se]. List of changes follows:

- o Bug fixes as mentioned in "The complete Spectrum Rom Disassembly" by Dr Ian Logan and Dr Frank O'Hara.
- o Support for EPSON printer via an Z80-PIO chip and Centronics connection.
- o Monitor support: start up with black screen and white characters.
- o Lightpen support: Lightpen routine for DK'Tronics lightpen is built in.

TK95.ROM - ROM from a Brazilian Spectrum clone called Micro Digital TK95.

IMC.ROM - Ian Collier's modified ROM. See the documentation that comes with xz80.

UNICORN.ROM - ROM used by Mikro-Gen's Shadow Of The Unicorn cartridge game.

11. Where can I get Microdrive cartridges?

Microdrive cartridges (as well as complete Microdrive kits) can be obtained from:

W.N. Richardson & Co.
6 Ravensmead
Chiltern Hill
Chalfont St Peter,
Bucks, SL9 0NB
United Kingdom

PHONE/FAX 01494-871319

12. How do I fix my worn out ZX-Spectrum keyboard?

If the connectors on the PCB have scratched off the silver from the ribbon, buy some so-called "conductive paint" which is a suspension of fine silver particles. If you let it dry, it conducts. Use a fine brush to replace what is scratched off from the ribbons. This paint is available from most electronics hobbyist stores (for example, from RadioShack in US).

13. What's the address of the "Spectrum Profi Club" in Germany?

SPC
c/o Haller
Ernastr. 33

D-51069 K"oln
Germany

14. What are the differences between the different Spectrum models?

- 1) 16K - original model. Rubber keys.
- 2) 48K - (most widespread) original model with RAM extension.
- 3) TS2068 - American version from Timex; had an AY-3-8912 sound chip, cartridge ports, two joystick ports and an additional 8Kb extension ROM with extra BASIC commands to support these devices. Silver plastic case and plastic keys.
- 4) TC2068 - same as TS2068; sold by a Portuguese arm of Timex away from usual Sinclair markets.
- 5) TC2048 - another one from Portuguese arm of Timex. Black with grey plastic keys. Built in Kempston joystick port, some higher resolution screen modes, 64Kb of memory with a paging system.
- 6) TK90X - clone of 48K by Micro Digital of Brazil, with a built-in joystick interface (Interface II compatible) and two new BASIC commands - TRACE, for tracing execution, and UDG for editing UDGs. This modified ROM caused some compatibility problems.
- 7) TK95 - as TK90X, but with a real keyboard (not unlike the C64's) and a more compatible ROM. Apparently a TC2048 in disguise.
- 8) 48K+ - as 48K, but had "real" keyboard.

9) 128K - 128Kb RAM, AY-3-8912 sound chip, keypad, RS232 port and "real" keyboard.
The last real Sinclair machine.

10) +2 - 128K with builtin tape deck, new (grey) case and keyboard.
First Amstrad-produced model.

11) +3 - 128K with builtin disk drive, new ROMs including a DOS and some internal changes on the bus. Back to a black case.

12) +2A - +3 with a tape deck instead of disk drive.

15. What peripherals exist?

This would be a big list, so I'll keep it to what I consider are major ones (additions welcome):

1) Interface 1 - joystick and microdrive interface, network capabilities.

2) Interface 2 - joystick and cartridge interface.

3) Microdrive - fast tape storage (requires Interface 1).

4) ZX Printer - thermal printer (aka silver bog roll burner).

5) Keypad - extension keypad for the 128K models that offered extra editing facilities in 128K BASIC.

6) Multiface - Romantic Robot's excellent device to stop the Speccy cold, enabling you to save the current state of the machine to tape or disk and enter POKEs.

7) Joysticks - many interfaces exist; Kempston, AGF/Protek, programmable interfaces (which mapped joystick positions to keypresses), etc. etc.

8) AMS Mouse - well, a mouse...

9) BetaDisk - an interface to enable attachment of a 3.5" drive.

10) Disciple/+D - MGT's interface to enable attachment of a 3.5" drive.

11) Wafadrive - not unlike a Microdrive.

16. Where is Sinclair Research now? Is Clive on email?

You could try contacting MENSA; Clive is a member.

Seamus Waldron [seamus_waldron@cix.compulink.co.uk] supplies the following from the last Profits/Loss account published by Sinclair Research (on 12th October 1995):

SINCLAIR RESEARCH LTD.
15/16 Margaret Street
London
W1N 7LE

United Kingdom
Tel: 0171 636 4488
Fax: 0171 580 6628

Sir Clive Sinclair is still the chairman; no email address is known for him (and if you were him and had one, wouldn't you keep it a secret from the slaving hordes of Speccy fans?!). Unfortunately the above address is already out of date; Sinclair has moved to somewhere near Kings Cross in London. More info welcome!

17. What the hell is a Spectrum?

The best home computer of all time. Came out in 1982 from Sinclair Research Ltd., and proceeded to capture the hearts and minds of millions of people. Based around a Z80 processor and utilising lots of neat tricks to keep the cost down, Clive (now Sir Clive) Sinclair effectively created the UK computer industry with his ZX80, ZX81 and ZX Spectrum. If you've never used one, get hold of an emulator and try out some of the classic games around (e.g. Jetpac) and see what **real** gameplay is about without needing 50 Mb of hard disk space and the type of processing power that interferes with military radar.

18. Is the Spectrum better than <other machine>?

Yes. ;-)

19. I have a question not covered here!

Well, there are other sources of information; in particular, if the question is about a particular game, your first stops should be Stephen Smith's Speccy Games FAQ and Games Database at:
<http://www.hermetica.com/technologia/sinclair/stevo/gamefaq.html> and
<http://www.hermetica.com/technologia/sinclair/stevo/gamedbs/gamedbs.html>
If they don't help, welcome to the group, ask away!

NET ADDRESSES

In between updates of this FAQ, new sites usually appear - I may have linked to them on my own page at <http://www.cs.bham.ac.uk/~dmb/speccy/> (until June 14th 1996) so take a look there if you're after something new. If you find an address in here that doesn't work any more, please let me know - and let me know if you find one that isn't listed here too!

This list is ordered by the sites' domain names, so don't feel bad if your site is listed after one you think isn't as good. If anyone thinks my description of their site is incorrect or unfair, again, let me know.

FTP Sites:

If you're after games, emulators or anything Spectrum-related, there are a number of FTP sites worth visiting, and new ones spring up now and again. bear in mind that many of the sites containing games are very busy, so you're better off trying at quieter times (e.g. very late at night in the UK), and don't use a web browser - use a proper FTP client (in which case the name of the site needs the 'ftp://' bit stripped off).

<ftp://akira.uc3m.es>
<ftp://alba.easynet.co.uk> (part time site - never seen it up myself)
<ftp://budda.tixm.tambov.ru/d/incoming/crems/speccy>
<ftp://drson.vse.cz/pub/sinclair>
<ftp://fly.cc.etf.hr>
<ftp://fly.cc.fer.hr/pub2/spectrum>
<ftp://ftp.dcc.uchile.cl/pub/OS/sinclair>
<ftp://ftp.demon.co.uk/pub/emulators/spectrum>
<ftp://ftp.enterprise.net/pub/incoming/speccy>
<ftp://ftp.enterprise.net/spectrum/>
<ftp://ftp.funet.fi/pub/misc/if-archive>
<ftp://ftp.gmd.de/if-archive>
<ftp://ftp.gui.uva.es/sinclair>
<ftp://ftp.inf.tu-dresden.de/pub/zxspectrum> (and /pub/incoming/zxspectrum)
<ftp://ftp.itu.edu.tr/pub/systems/spectrum>
<ftp://ftp.komkon.com/pub/Spectrum> (and /incoming/Spectrum)
<ftp://ftp.lanprojekt.cz/pub/sinclair>
<ftp://ftp.nvg.unit.no/pub/sinclair> (*THE* Speccy FTP site!)
<ftp://ftp.polsl.gliwice.pl/pub/systems/ZXSpectrum>
<ftp://ftp.sun.ac.za/pub/misc/sinclair> (and /pub/msdos/zx)
<ftp://ftp.zx-museum.org.ru>
ftp://maya.dei.unipd.it/pub/sinclair_QL/spectrum
<ftp://members.aol.com/zxspectrum/snaps/>
<ftp://oak.oakland.edu/pub/msdos/emulators>

<ftp://phantom.riviera.pw.edu.pl/pub/speccy> (kept by Hacker Chris)
<ftp://rcs1.urz.tu-dresden.de/pub/soft/zxspectrum>
<ftp://spodbox.ehche.ac.uk/pub/users/majik>
<ftp://tardis.soc.staffs.ac.uk/pub/sinclair>
<ftp://uhura.ijs.si/pub/zx> (used to be ftp.ijs.si)
<ftp://virgo.inesc.pt/pub/games>
<ftp://wuarchive.wustl.edu/systems/sinclair>
<ftp://www.il.ft.hse.nl/pub/sinclair/>

Web Sites

Web sites abound, and new ones appear all the time... try web searches for 'Sinclair Spectrum', 'ZX Spectrum', 'speccy', etc.

Web Sites with Reference Material

<http://www.csv.warwick.ac.uk/~mauqx/sinclair/>
Sinclair Index - links and hardware manuals, e.g. Interface 1, Multiface 128 manuals.

<http://www.mono.org/~ritchie/speccy/speccy.html>
Sinclair Spectrum Home Pages - snapshots and technical articles, and letters from Your Sinclair!

<http://www.nvg.unit.no/spectrum>
Spectrum Forever! - THE Spectrum site! A bit disorganised and rather out of date now, but still a good collection of stuff.

<http://spodbox.linux.org.uk/~majik/sinclair/>
Sinclair Microcomputer Museum - (aka Sinclair Section).

<http://www-users.informatik.rwth-aachen.de/~afw/sincfo0.html>
Sinclair ZX Spectrum Forum - primarily a BASIC reference

Web Sites with Games (Snapshots):

<http://www.accessone.com/~mccoward/>
M. Coward's SpecNet Online; some games on a nicely presented site.

<http://alba.easynet.co.uk>
John Dow's part-time page - supposedly up at 10-12PM UK time at weekends; I've never managed to get a response at any time though.

<http://homepages.enterprise.net/glyn.harper/spectrum.html>
Glyn Harper's ZX Spectrum Emulator page - basically just links to his FTP site, with around

16Mb of games.

<http://ireland.iol.ie/~sjones>

Steve Jones' Sinclair Spectrum Page - has a snapshots requests page, though hasn't been updated since it started from what I can see - apparently due to worries about copyright violation.

<http://lia01.unizar.es/curro/spectrum.htm>

Francisco Cotrina's Spectrum page - with details of his snapshots mail server and some emulators to download.

<http://members.aol.com/zxspectrum/speccy.html>

Harvey Lodder's "Most Wanted" snapshots request page.

<http://osiris.sund.ac.uk/~ca4aba/snaps.html>

Andrew Barker's Spectrum Snapshots - yet another requests page, the most active one.

<http://relcom.eu.net/zx/>

Aleksandr Babaylov's ZX Spectrum Software Museum - a large collection of snapshots and tapes; a lot of games here, but it can be a slow site.

<http://sable.ox.ac.uk/~sjoh0132/>

John Elliott's Home Page - the home of Jet Set Willy 128K, and some useful icons for Sinclair pages (as used on my page).

<http://spodbox.linux.org.uk/~blood/>

Blood's Crap Speccy Page - home of the comp.sys.sinclair crap games compilation and the Manic Miner editor.

<http://www.acropolis.gr/~icarus/spectrum.html>

Theo Devil's Spectrum page - has copies of his demos and games there, plus more you may have heard of Theo as the guy behind Hellenic Software, he had a few things published on Crash/YS cover tapes.

<http://www.cs.ucl.ac.uk/students/zcacbb1/spectrum/speccy.html>

Ben Baylis' Spectrum Stuff - including the beginning of a point and click snapshots list (using the akira FTP site), a requested snapshots page and YS Smash Tips!

<http://www-dse.doc.ic.ac.uk/~np2/spectrum/index.html>

Execute Spectrum Programs via The Web! - not really practical, but worth a look.

<http://www.io.org/~diehl/speccy.html>

Eric March's Speccy! - including what was a great requested snapshots page, but it hasn't been updated since August 1995. Should hopefully be back on form shortly, as Eric will be updating the site with more snapshots and other stuff 'soon'.

<http://www.rtc-carlow.ie/student/obrienk/spectrum.html>

Karl O'Brien's Sinclair Spectrum 48K Heaven - has a few games and a copy of Z80 on his page.

Web Sites with Game Hints/Cheats/Information:

<http://carlton.innotts.co.uk/~silthas/spectrum.html>
Sinclair Spectrum Hall of Fame - screenshots of classic games.

<http://orca.ucd.ie/~conormc/spectrum/zxspectrum.html>
Conor McCarthy's ZX Spectrum page - home of the Miner Willy Museum (tours of Manic Miner etc.), and more broken links than you can shake a stick at - all very much under construction.

<http://www-dept.cs.ucl.ac.uk/students/zcacbb1/spectrum/smashtips/>
YS Smash Tips - courtesy of Ben Baylis. Lots of tips and cheats.

<http://www.doggysoft.co.uk/chzxp.html>
Cheats Prosper - Multiface POKEs - an impressive list of cheats here.

<http://www.ftch.net/~floodnet/spectrum/spectrum.htm>
Glenn Flood's Spectrum/ZX-81 Cassette Inlay Artwork Page - Glenn did artwork for some QuikSilva games.

<http://www.gatewest.net/~cpb/lom.html>
The Moon Palace - a home page for the Lords of Midnight saga.

<http://www.hermetica.com/tecnologia/sinclair/stevo/index.html>
Stephen Smith's Master Speccy page- lots of stuff including the Spectrum games database and games FAQ. A must see!

<http://www.soton.ac.uk/~rjc394/>
The YS Tip Shop Tiptionary- courtesy of Robert Cooper. Decidedly incomplete; not updated since it was first put up.

<http://www.wintermute.co.uk/users/gargamel/ultimate.html>
Gargamel's Ultimate Play The Game Loading Screens.

Web Sites of Emulator/Utility Authors:

<http://diana40.paisley.ac.uk:80/~com40014/>
James McKay's PC Games Haters Page - home of x128, the DOS and UNIX Speccy emulator. May disappear after June 1996.

<http://fly.cc.fer.hr/~arsen/tapefix/>
TapeFix - a program to make your old Speccy tapes readable once more.

<http://julia.gns.getronics.nl/~mheide/spectrum.html>
Martijn van der Heide's Spectrum Games Database - a PC program to catalog your snapshot and

tape collection and launch JPP or Z80 to run a particular snapshot or tape. A very well done program, and it's free!

<http://web.jet.es/~tpomar/>

Toni Pomar's ZXAM homepage - home of ZXAM, oddly enough.

<http://www.comlab.ox.ac.uk/oucl/users/ian.collier/Spectrum/index.html>

Ian Collier's xz80 homepage - home of xz80, believe it or not.

<http://www.comlab.ox.ac.uk/oucl/users/robin.watts/wss/index.html>

Warm Silence Software - the producers of Z80Em.

<http://www.csd.uu.se/~richardc/home.shtml>

Richard Carlsson's home page - author of Speccylator, a Speccy emulator for the Amiga. You'll find a mention of it right at the bottom of his page.

<http://www.doggysoft.co.uk/>

DoggySoft have written several Acorn ARM Spectrum-related utilities.

<http://www.freeflight.com/fms/>

Marat Fayzullin's home page - author of emulators for the MSX, NES, Colecovision, Game Boy etc. and at least one Spectrum-related utility (Butcher).

<http://www.ugr.es/~pedrom/sinclair.htm>

Pedro Salaz's Sinclair home page - home of the Sinclair v2.00 emulator.

Web Search Engines:

<http://altavista.digital.com/>

Alta Vista - Digital's rather neat search engine. Very powerful, for instance it can do searches for pages linked to a specific page as well as search in USENET news articles and web pages.

<http://drson.vse.cz/snapsearch/>

Hynek Med's Spectrum related files search - searches over several FTP/WWW sites to match the filename you give it. Probably the most useful search engine if you're after a particular game.

<http://ftpsearch.unit.no/ftpsearch/>

FTP Search - will search for any file on the FTP sites it indexes; can also be used to find Spectrum files of course! Much like Archie, but nicer.

<http://www.infoseek.com/>

InfoSeek Guide - another generic web search engine, but as always, these are just as useful for finding Speccy stuff as finding anything else.

<http://www.nvg.unit.no/spectrum/search.html>

An interface to the gopher search engine at NVG. Unlike snapsearch, this one searches on

company and game names as well as the filename; but it doesn't have as good an index, so snapsearch often gives better results.

<http://www.yahoo.com/Computers/Hardware/Platforms/Sinclair>
Yahoo's Sinclair page - a few links, some out of date.

Magazines On The Web:

<http://jumper.mcc.ac.uk/~simonc/ys>

Your Sinclair - parts of some of the later issues up on the web courtesy of Simon Cooke, YS's tech tips editor for those issues.

<http://www.cs.bham.ac.uk/~dmb/speccy/emulate/>

Barry Plewa's Emulate! - has games, technical stuff, reviews, chart listings, and much more. The HTML version of the magazine is at this site, but plain text versions and ZIPped up copies of the HTML version are also available, e.g. from

<ftp://ftp.demon.co.uk/pub/emulators/spectrum/magazines/emulate/> and

<ftp://ftp.nvg.unit.no/pub/sinclair/snaps/discmag/emulate/>

Other Web Pages:

<http://hal.cling.gu.se/~cl0polau/zx.html>

Sinclair ZX Spectrum - a bit empty, this page.

<http://members.aol.com/jozxyqk/specmain.htm>

Spectrum Appreciation Society. Windows version of Advanced Lawnmower Simulator and some sound samples.

<http://paul.rutgers.edu/~savoIU/spectrum/>

Sinclair ZX Spectrum Switchboard - just links.

<http://sunsite.dcc.uchile.cl/sinclair/>

Sinclair Pages - all in Spanish and all very unfinished looking.

<http://stunix.matsim.udmurtia.su/~alt>

Aleksey Titov's [alt@stunix.matsim.udmurtia.su] site; apparently to be up soon, but hasn't been every time I've tried.

<http://tachyon.mono.org/~sinclair/>

Sinclair Scene - Monochrome BBS's Speccy web page; instructions on how to access Monochrome and see the actual Sinclair section within. Monochrome is a telnet-based BBS.

<http://www.8bit.com/>

8 Bit Universe. Under construction when I looked; but there will be a Spectrum area. Currently there are two German Spectrum adverts.

<http://www.afn.org/~afn09039/sinclair.html>

Andy's Sinclair Spectrum Page - just a mention of Ant Attack and Speccy repair, more to come apparently...

<http://www.aston.ac.uk/~goringgn/>

No description yet.

<http://www.dur.ac.uk/~d44abw/spectrum.html>

Spectrum Page - just links.

<http://www.earth.ox.ac.uk:80/~steve/Spectrum/index.html>

Sinclair ZX Spectrum Flyer - some old advertising bumph.

<http://www-ia.di.fct.unl.pt/~vjr/Spectrum.html>

Vitor Rocio's Spectrum stuff - just links, though he promises more in the future.

<http://www.il.ft.hse.nl/~ben/sinclair/sinclair.html>

Sinclair Page - links and info on SAM Coupe and Dutch QL user group (in Dutch!).

<http://www.nvg.unit.no/spectrum/jmg7/>

John Garner's Spectrum PD/User Groups page.

<http://www.ozemail.com.au/~martinf/>

Martin Fowler's Australian Retro Computing has links to all sorts of old computer stuff, with the bonus of local copies of emulators etc. so Australians don't have to wait ages for the download.

<http://sol.soc.staffs.ac.uk/cm5bcpmh/sinclair/index.html>

Patrick Hill's Sinclair Zone. Some stuff on the C5.

<http://www.wam.umd.edu/~audley/english/speccy.html>

Damian Audley's Speccy Page - mostly links for Mac emulators etc.

<http://zipi.fi.upm.es/~a920142/spectrum.html>

Eduardo García Gallego's Spectrum page - just links.

News Group:

Well, I'd guess you know about the newsgroup if you're reading this, but maybe not, so...

news:comp.sys.sinclair

(or just comp.sys.sinclair if you're not using a web browser)

Mailing list (the mailing list is a feed from the newsgroup):

Send mail to lserv@psg.com with the following in the BODY of the message (not the subject as previously stated in this FAQ - sorry!):

subscribe sincnews yourusername@your.domain

Don't put anything else in the body of the message.

IRC:

The channels #spectrum and #sinclair are now officially registered with DALnet; here are some DALnet servers:

liberator.dal.net (UK) (port 7000)

irc.dal.net (USA) (port 6667)

igc.dal.net (USA) (port 6667)

If you are interested, please email Theo Develegas [icarus@acropolis.net - his IRC nick is _Wind_], so that he can fix some online meetings. He also plans to have an IRC section on his home page (see above).

Other Resources:

<gopher://gopher.nvg.unit.no>

A Spectrum-related files search engine. Searches on company and game names as well as file names.

Venture BBS in Finland: +358-0-8092126

The Bus Stop BBS in England: +44 1273 774121 (go to area 34)

There is also Alchemist News, a tape-based Spectrum magazine. Past issues can be found on NVG: <ftp://ftp.nvg.unit.no/pub/sinclair/snaps/discmag/alchnews/> and you are best off reading one of them for more information.

EMULATORS

Authors, where known, are in (brackets) after the name of the emulator. Email addresses are in [square brackets]. PLEASE don't bother these guys asking for an emulator; there are plenty of places to get the emulators from, and I now list example addresses.

The features of each emulator are listed in the following set order:

- a) Emulation: 48/128/+2/+3/Multiface/Disciple/Interface 1/etc.
- b) Loadable formats - e.g. .SNA, .Z80 etc.
- c) Saveable formats.
- d) Real tape loading - type of support offered.
- e) Requirements - video, processor etc.
- f) Speed.
- g) Status: freeware/shareware/commercial/etc.
- h) Good points.
- i) Bad points.
- j) Other comments.
- k) Site available from. Please note that these are not the **only** places to get the emulator from; there are many other sites, but the directories may be different. Try a WWW/FTP search for the name of the archive file if the site listed is too slow for you.

Emulators for MS-DOS PCs:

1. Elwro 800-3 Jr v1.0 (Piotr Schmidt & Piotr Wolter)

- a) Emulates an Elwro - not 100% compatible with ZX Spectrum (the real Elwro had problems with JetPac while other games were ok) - with two disk drives accessed via * commands (e.g. SAVE * "file") with capacity of about 700kB. These are physically files.
- b) Loads files from disk. Snapshots/tape files?
- c) Saves files to disk. Snapshots/tape files?
- d) Real tape support?
- e) Requires Hercules, CGA or EGA.
- f) Speed?
- g) Commercial only.
- h) Capable of running CP/M. Changeable interrupt frequency (45, 50, 65, 70 Hz). Comes with utility to read the real Elwro Junior disks on PC.
- i) Bad points?
- j) All system messages are in Polish.
- k) Available from?

2. JPP (Arnt Gulbrandsen [agulbra@nvg.unit.no])

- a) Emulates 48K Spectrum, Kempston joystick (using PC joystick).
- b) Loads .SNA snapshots and from sampled tapes (.VOC).
- c) Saves .SNA snapshots.
- d) Tape support (read-only) via sampled sound files (.VOC).
- e) Requires VGA. Real-speed version requires 286 or above.
- f) 386/25MHz or faster is needed for the best performance. 486/25MHz or 386/44MHz is needed for good sound emulation.
- g) Freeware.
- h) Includes program for reading and converting snapshots from MGT disks. Also includes a very old version of SPConv.
- i) Screws up system screen mode on quit. Restore with MODE CO80 command. R register imperfectly emulated. No rainbowing.
- j) Two versions supplied; one for slower machines, and one (PJPP) for faster machines; PJPP has better emulation and emulates the FLASH attribute and the border. No longer being updated.
- k) Available from <ftp://ftp.nvg.unit.no/pub/sinclair/utills/pc/jpp.zip>
<http://www.nvg.unit.no/spectrum> is maintained by the author, but does not have any specific mention of JPP.

3. Sinclair v0.99 (previously SPECTRUM in this FAQ) / VGASPEC / SPEC386
(Pedro Gimeno)

- a) Emulates 48K Spectrum.
- b) Loads .SP snapshots.
- c) Saves .SP snapshots.
- d) Loads from tapes through a wire connected to the parallel port.
- e) Requires VGA, 286. (386 for version 0.99D - SPEC386.EXE)
- f) Matches real Spectrum speed at 386/25MHz.
- g) Shareware.
- h) Good points?
- i) R register and I/O (other than tape) are not emulated.
- j) VGASPEC emulator is an *illegal* prerelease of this emulator. All documentation and onscreen messages in Spanish.
- k) Available from
<ftp://ftp.nvg.unit.no/pub/sinclair/utills/pc/vgaspec.zip>
<ftp://ftp.nvg.unit.no/pub/sinclair/utills/pc/span099d.zip>

4. Sinclair v2.00 (Pedro M. R. Salaz [pedrom@goliat.ugr.es])

- a) Emulates 48K Spectrum.
- b) Loads .SIT snapshots and from .BLK tape files.
- c) Saves .SIT snapshots and to .BLK tape files.
- d) Real tape support via a separate program only supplied with registered version.
- e) Requires VGA.
- f) Speed quite good on a 486SX/25.

- g) Shareware; unregistered version has nag screen that renders the built-in debugger practically unusable. Registered version comes with utility to convert .SP snapshots (from the above similarly named emulator!) to .SIT format, can save Spectrum screen to .PCX screens and comes with a tape loader program, plus a disk full of games as .BLK files.
- h) Built-in debugger (much like Borland's Turbo Debugger) for stepping through, breakpointing, modifying etc. Z80 code (including the ROM).
- i) Speed varies with what the Spectrum program is doing.
- j) Available in both Spanish and English versions. The .BLK file format is the same as Z80's .TAP format. I don't think this emulator has any link to the other one of the same name.
- k) Available from the author's web site at <http://www.ugr.es/~pedrom/sinclair.htm> as well as <ftp://ftp.nvg.unit.no/pub/sinclair/utills/pc/sinceng2.zip>

5. SP / Polish emulator (Author?)

- a) Emulates 48K Spectrum.
- b) Doesn't load snapshots. Loads from .SPC/.ZX tape files.
- c) Doesn't save snapshots. Saves to .SPC/.ZX tape files.
- d) No real tape support.
- e) Requires CGA or EGA and a ROM file called ROM.SPC - this file is just a 16Kb ROM dump, identical to the SPECTRUM.ROM file included with JPP and xz80.
- f) Speed not known.
- g) Freeware?
- h) Built in MON-like debugger (press F8) for stepping through Z80 code.
- i) No accompanying documentation at all. Colour palette a bit odd.
- j) No other comments.
- k) Available from <ftp://ftp.nvg.unit.no/pub/sinclair/utills/pc/polish.lzh>

6. The Spectrum Emulator / SPECEM / Irish emulator (Kevin J. Phair)

- a) Emulates 48K Spectrum with Multiface 1, Kempston joystick (with cursor keys and Alt).
- b) Loads .PRG snapshots and files from 'tape' - i.e. your hard disk.
- c) Doesn't save snapshots. Saves files to hard disk.
- d) No real tape support.
- e) Requires 16MHZ 286, EGA or better.
- f) Works reasonably fast on 286 and a bit faster than necessary on 386/486 systems. 386/40MHz or 486/25MHz is needed for good sound emulation.
- g) Freeware, but a donation to the author gets you the source code plus 'various utilities'.
- h) Includes program for loading snapshots from MGT/Datel Plus-D disks.

Comes with a Z80 disassembler written in BASIC, and a speed testing program. Ran at least one snapshot (Scumball) that no other emulator could manage (JPP's documentation makes a mention of this one not working - Arnt thought it was a bad snapshot).

- i) Imperfect R register emulation. Allows modification of the ROM (though it does get refreshed to normal condition whenever loading/saving).
- j) Onscreen display of Z80 registers (can be disabled, but not removed from the screen). Attributes can be disabled for faster emulation.
- k) Available from
<ftp://ftp.nvg.unit.no/pub/sinclair/utills/pc/irish.zip>

7. SP_UKV / SP_EMU (Author?)

- a) Emulates 48K/128K Spectrum with TR-DOS (Technology Research Disk Operating System for the BetaDisk 3.5" disk drive).
- b) Loads .SNA (both standard 48K and a new 128K version) and 128K .Z80 snapshots.
- c) Saves .SNA (48/128) and 128K .Z80 snapshots.
- d) Real tape support?
- e) Requires VGA.
- f) Rather slow on a 486SX/25.
- g) Shareware. Registered version has an excellent Turbo Debugger style debugger (similar to Sinclair v2.00's one).
- h) Good points?
- i) Bad points?
- j) Usually found as SP_UKV.ARJ along with SP_EMU.ARJ; the latter being a cracked version which appears to be enable all the options only available if you register it. The help screen is identical to the one on the Polish emulator - was that an old version?
- k) Available from
ftp://ftp.nvg.unit.no/pub/sinclair/utills/pc/sp_ukv.arj
ftp://ftp.nvg.unit.no/pub/sinclair/utills/pc/sp_emu.arj

8. ZX-Spectrum 128K + TR-DOS 1.05 / SP105 (Author?)

- a) Emulates 48K/128K Spectrum with TR-DOS.
- b) Loads?
- c) Saves?
- d) Real tape support?
- e) Requires VGA.
- f) Speed?
- g) Status?
- h) Good points?
- i) Refuses point blank to run in a Windows DOS session. Actually, should this be a good point? ;)
- j) All messages and documentation in Russian... if anyone fancies doing a

translation, I'd be very grateful! Written by somebody from Moscow on FidoNet; no email address for them.

k) Available from

<ftp://ftp.nvg.unit.no/pub/sinclair/utils/pc/sp105.arj>

9. Warajevo v1.2 (Zeljko Juric [Z.JURIC@zamir-sa.ztn.zer.de] and Samir Ribic [S.RIBIC@zamir-sa.ztn.zer.de])

- a) Emulates 48K/128K/+2 Spectrum, ZX Printer, Interface 1 (only RS232), Kempston joystick (with cursor keys and + or ~ keys as fire or with PC joystick or mouse), 128's keypad.
- b) Loads .SNA and .Z80 snapshots and from near-perfect virtual tape (.TAP) format - this is not the same as Z80's .TAP format.
- c) Saves .Z80 snapshots and to .TAP tape files.
- d) Supports loading from real tapes only via an RS232 link from a real Spectrum.
- e) Requires 286, Hercules, CGA, EGA or VGA.
- f) Around 15% slower than Z80; not noticeable on anything more powerful than a 386.
- g) Freeware, though the authors welcome donations, in return for which you at least get your name listed in the documentation of the next release.
- h) Better AY-sound through PC speaker than any other PC emulator. Built-in debugger, though not as friendly as the ones in Sinclair v2.00 and SP_UKV. Can create stand-alone snapshot files that don't need to be loaded into an emulator. Comes with suite of utilities built into a single program - for converting from various formats, holding a database of games, etc. Lots of options.
- i) Colour palette looks a bit odd.
- j) Written under very poor conditions during war in Bosnia. Generally considered the second best Speccy emulator on the PC.
- k) Available from

<ftp://ftp.nvg.unit.no/pub/sinclair/utils/pc/warajevo-spectrum-120.zip>

10. WSpecem v1.11 (Rui F. Ribeiro [rff-ribe@csm.uwe.ac.uk] until June 1996)

- a) Emulates 48K Spectrum with Kempston joystick.
- b) Loads just about every snapshot and tape format there is, including SLT.
- c) Saves just about every snapshot and tape format there is, but not SLT.
- d) Real tape support provided via an extra program.
- e) Requires MS Windows 3.1, 3.11 or 95, WinG.
- f) Around 25-30% Spectrum speed on a 486SX/25 @ 800x600 res, 64K colour Cirrus video card, 8MB machine. Runs at a bit over 100% speed on a P75 with otherwise identical setup. Note these speeds are for 1x1 scaled window with a /3 frame rate. Larger sizes and full frame rate are slower.
- g) Freeware - GPL; all the author wants is a postcard.
- h) Resizable window, with quick 1, 2 and 3 times normal size settings as well. Quick save/load function for saving a snapshot and reloading it

without going through a file selector. Drag and drop snapshot loading.
Source (in C) supplied. Saves Spectrum screen to SCR and PCX files.

- i) Doesn't save SLT snapshots yet. And my machine isn't fast enough :(
- j) The very first Spectrum emulator for MS Windows.
- k) Available from
<http://lia01.unizar.es/curro/wspecem.zip>
<ftp://ftp.nvg.unit.no/pub/sinclair/utils/pc/wspv111.zip>

11. Z80 v3.03 (Gerton Lunter [gerton@math.rug.nl])

- a) Emulates 48K/128K Spectrum with Multiface 1/128, cursor, Sinclair and Kempston joysticks, AMS Mouse, Disciple disk interface, Interface 1 and Microdrives, SamRam (author's own ROM with a basic MON-like debugger).
- b) Loads .SNA and .Z80 snapshots, including level-loader support, and from .TAP tape files.
- c) Saves .Z80 snapshots and to .TAP tape files.
- d) Loads from sample tapes (.VOC) and direct from Sound Blaster MIC input, or from parallel port using a custom interface (all these only in the registered version, and the parallel port interface itself costs extra - though a diagram is provided if you want to build your own). Contrary to what several people think, this support **does** allow loading of turbo-loaders, speedlock games, etc.
- e) Requires Hercules, CGA, EGA or VGA.
- f) Can be up to 200% Spectrum speed on a 486SX/25, depending on emulation options chosen.
- g) Shareware. Registered version has real tape support and can be slowed down, and comes with several utilities (for converting from various formats) and source code (mixture of assembler and C).
- h) Most user-friendly emulator; very good menu-interface and documentation. Good high-resolution colour emulation (rainbowing!).
- i) 128K sound can be a bit wobbly.
- j) Usually considered to be the best emulator available for any platform. Earlier versions were pretty good but do not compete with version 3.03, though version 3.02, which was released on a PC Format cover disk, had speed control whereas 3.03 does not unless you register it. However, the PC Format version is not distributable - if you have it and didn't buy that issue of PC Format, their lawyers may want to talk to you! Version 3.04 expected soon; it will have, among other updates, support for .SLT snapshots. A Windows version is also in development.
- k) Available from
<ftp://ftp.nvg.unit.no/pub/sinclair/utils/pc/z80-303.zip>

12. Z80-TR-DOS 2.02 (modified version of Z80 3.02 by VitaSoft)

- a) As Z80, with addition of TR-DOS emulation.
- b) As Z80.

- c) As Z80.
- d) No real tape support; if you want that, you'll need to register a real version of Z80.
- e) As Z80.
- f) As Z80.
- g) An illegal hack.
- h) As Z80. Version 3.02 of Z80 had slightly poorer high resolution colour emulation however; and this hack is the same.
- i) As Z80.
- j) This program is an obviously modified copy of Z80 version 3.02 (as released on a PC Format cover disk) with Russian documentation. The title screen has some cyrillic text on it along with what I presume is a phone number (In Russia? Of whoever is behind VitaSoft?).
- k) Not made available at Gerton's request, nor will I reveal the site I got it from. No doubt it will soon turn up from somebody else less honest than me though...

13. ZX Spectrum-Emulator v1.0s / SPECEMU (Bernd Waschke)

- a) Emulates 48K Spectrum, Kempston joystick (using PC's analogue or digital joystick).
- b) Loads .SEM snapshots.
- c) Saves .SEM snapshots.
- d) Real tape support (registered version only) via a Soundblaster card, but you have to manually calibrate it with a number.
- e) Requires 386DX/40 and 1Mb of RAM.
- f) About 75% of normal Spectrum speed on a 486SX/25.
- g) Shareware. Some options not present in unregistered version.
- h) POKE facility. Also has what appears to be a Game-Genie/Game Wizard-like option to automatically find POKES (registered version only).
- i) Doesn't emulate flags register properly; in particular, the H flag. BRIGHT attribute not emulated.
- j) All documentation and onscreen messages in German. Anyone fancy translating it?
- k) Available from
ftp://ftp.nvg.unit.no/pub/sinclair/utils/pc/specemu_1_0s.zip

14. x128 v0.4 (James McKay [com40014@paisley.ac.uk] until June 1996)

- a) Emulates 48K/128K Spectrum, Kempston/Sinclair/Cursor joysticks.
- b) Loads .SNA, .Z80 and .SLT snapshots, and from Z80's .TAP tape files.
- c) Saves .Z80 and .SLT snapshots.
- d) No real tape support.
- e) Requires VGA.
- f) About 50% of normal Spectrum speed on a 486SX/25.
- g) Freeware.

- h) Hi-res emulation of screen and left & right border allowing colours in an 8x1 block (rainbowing). 128K sound is very good. Now has a file selector and much better user interface than previous versions.
- i) Some changeable options require re-compilation.
- j) Also available for UNIX/Linux systems with X Windows (X11).
- k) Available from author's home page (until some time in June 1996) at <http://diana40.paisley.ac.uk:80/~com40014/> as well as <ftp://ftp.nvg.unit.no/pub/sinclair/utils/pc/x128v04.zip>

Emulators for CBM Amiga:

1. KGB v1.3

- a) Emulates 48K Spectrum, cursor joystick (with cursor keys) and Kempston joystick (with Amiga joyport 2).
- b) Loads .ZX snapshots.
- c) Saves .ZX snapshots.
- d) Loads/saves from/to real tapes via a sound-digitiser.
- e) Requires monochrome or colour.
- f) Speed not known.
- g) Freeware?
- h) Has a pitch-compensated mode, that transposes the sound two octaves up (handy on slow Amigas).
- i) Doesn't multitask. Isn't completely compatible with ZX-Spectrum; several games don't work, e.g. Chronos, Starstrike II.
- j) No other comments.
- k) Available from <ftp://ftp.nvg.unit.no/pub/sinclair/utils/amiga/kgb-1.3.lha>

2. Speccylator v1.0 (Richard Carlsson [richardc@csd.uu.se])

- a) Emulates 48K Spectrum.
- b) Loads .SNA snapshots, no tape support.
- c) Saves .SNA snapshots, no tape support.
- d) No real tape support.
- e) Requires just any Amiga and around 300Kb of free memory.
- f) 'Faster than most' - except Peter McGavin's Spectrum on a 68000-based Amiga.
- g) Freeware.
- h) Fully multitasking, nice graphical interface, better sound than Peter McGavin's Spectrum. Can dump Spectrum screen to an IFF ILBM file. Preferences can be set globally and for individual snapshots. Better Z80 emulation than ZXAM.
- i) No particular bad points, unless you count lack of features - e.g.

there's no joystick support.

- j) The idea behind the project was to create a Z80 emulation kernel that could run on any 68000 machine, i.e. not using any Amiga specifics. The Z80 emulator kernel can therefore be used for an emulator on any other 68000-based machine, e.g. the Atari ST. Not updated for some time due to author being busy with other things, but he will be making some updates soon (having seen ZXAM!).
- k) Available from author's homepage:
<http://www.csd.uu.se/~richardc/>

3. ZXAM v2.0 (Toni Pomar [tpomar@jet.es])

- a) Emulates 48K Spectrum, Kempston, Sinclair II and Cursor joysticks. Also emulates the 128K's AY-8910 sound chip, so 128K-enhanced games that use only the sound chip and not the extra RAM often work and give better sound.
- b) Loads .SNA and .SP snapshots. Can also apparently load just about any format of snapshot via Rexx scripts. Loads tape files from disk.
- c) Saves .SNA (and .SP?) snapshots. Can apparently save just about any format of snapshot via Rexx scripts. Saves tape files to disk.
- d) Real tape support via a custom tape interface (diagram and instructions to make this are supplied).
- e) Kickstart 2.04 and 68020 or higher required.
- f) Apparently 'impressive' on an A1200.
- g) Shareware.
- h) Runs in a window on Workbench screen, on a separate draggable screen, or in exclusive mode. Automatic speed adjustment for fast machines. Can save snapshots to a real tape for loading on a real Spectrum. Can load PowerPacked snapshots if you have the powerpacker.library in the LIBS: drawer.
- i) At least 38 games (including Chronos and Starstrike II) do not work; so emulation obviously isn't perfected.
- j) Author is working on fixing problems that cause those games not to work, along with a built-in debugger, full 128K emulation and tape loading via sound samples. There is a thing called the 'ZX Poke Guide' which is not an emulator but instead it's an AmigaGuide by Richard Koerber [rkoerber@tfh.dssd.sub.org] full of POKEs (about 677 games). It uses the ARexx port of ZXAM to make the pokes and run/stop the emulator from the AmigaGuide itself.
- k) Available from author's homepage:
<http://web.jet.es/~tpomar/> (that last slash is very important!) as well as the 'misc' directory on AmiNet. The ZX Poke Guide (v2.0) can also be found there, e.g.:
<http://ftp.wustl.edu/pub/aminet/misc/emu/ZXPokeGuide20.lha>

4. Spectrum v1.7 (Peter McGavin [peterm@kea.grace.cri.nz])

- a) Emulates 48K Spectrum, cursor joystick (with cursor keys) and Kempston joystick (with Amiga joyport 2).
- b) Loads .SNA snapshots and tape files from separate .header and .bytes files on Amiga disks.
- c) Saves .SNA snapshots and tape files to separate .header and .bytes files on Amiga disks.
- d) Load/saves Spectrum files from/to tapes through a sound-digitiser.
- e) Toni Pomar says this one is 68000 only and incompatible with Kickstart 2.0 and up. Comment on speed, below, seems to disagree with this...
- f) Is reasonably fast on 68030/25MHz, but slow on 68000/7.14MHz machines. At least 68020/14MHz (A1200) and FAST RAM are recommended. Not as fast as ZXAM.
- g) Freeware?
- h) Multitasks.
- i) Colour palette is a bit strange.
- j) Very old, and not supported any more.
- k) Available from
gopher://micros.hensa.ac.uk:70/11/micros/amiga/dos/t/t042

5. ZX-Spectrum v4.71 (Jeroen Kwast [jkt@gesasdsc.nt.getronics.nl])

- a) Emulates 48K Spectrum.
- b) Loads?
- c) Saves?
- d) Real tape support?
- e) Requires?
- f) Speed?
- g) Status?
- h) Good points?
- i) Bad points?
- j) No other comments.
- k) Available from 'misc' directory on AmiNet, e.g.:
ftp://src.doc.ic.ac.uk/pub/aminet/misc/emu/zxspectrum4.71.lha

5. ZX Spectrum Emulator v0.10 (A.Schifano & V.Gervasi)

- a) Emulates 48K Spectrum, presumably, and Kempston joystick.
- b) Loads snapshots (type unknown) and from tape (AmigaDOS files).
- c) Presumably saves snapshots and to AmigaDOS files.
- d) Real tape support via a tape interface.
- e) Requires any Amiga.
- f) Speed? Has three video modes; monochrome, where you choose the overall PAPER and INK colour for the screen; colour 1 with limited updates of the screen and colour 2 with full updates (slower).
- g) Appears to be a commercial product. The demo does not include tape support or snapshot support, and resets after a certain amount of time.
- h) Good points?

- i) Spectrum ROM doesn't come with the emulator; apparently Amstrad denied permission (perhaps because this is a commercial product). A hacked version of the demo does include it, illegally.
- j) Information here comes from a brief Readme file in a hacked demo version of the emulator. More information welcome.
- k) Available from DigiMail in Italy:
 Digimail Srl Phone: ++39-2-427621
 Via Coronelli 10 Fax: ++39-2-427768
 20146 Milano
 ITALY

Emulators for Atari ST(e)/TT/Falcon:

1. ZX-SPECTRUM Emulator / Specci v2.07 (Christian Gandler)

- a) Emulates 48K Spectrum, Interface 1 (Microdrives and RS232), Kempston and Sinclair joysticks.
- b) Doesn't load snapshots. Loads tape files from separate files on Atari disks.
- c) Doesn't save snapshots. Saves tape files to separate files on Atari disks.
- d) No real tape support.
- e) Requires at 1Mb of memory.
- f) Speed is around 70% (at best) of real Spectrum on plain STs.
- g) Freeware?
- h) Also runs on TTs and Falcons (in ST low/high resolutions).
- i) Bad points?
- j) All documentation in German. There is a poorly-translated version of the documentation supplied (taken from a 'Pompey Pirates' disk that had this emulator on it) with at least one of the copies on the FTP sites.
- k) Available from
ftp://ftp.nvg.unit.no/pub/sinclair/utils/atari/zx_sp207.zip
<gopher://micros.hensa.ac.uk:70/11/micros/atari/tos/o/o096>
<ftp://ftp.nvg.unit.no/pub/sinclair/utils/atari/atari-speccy.zip>
<ftp://ftp.nvg.unit.no/pub/sinclair/utils/atari/atari-spectrum.zip>
 (last two may be earlier versions)

Emulators for Acorn Achimedes:

1. !MZX v1.0 (Graham Willmott)

- a) Emulates 48K Spectrum.
- b) Loads .SNA snapshots.

- c) Saves .SNA snapshots.
- d) No real tape support?
- e) Requirements?
- f) Runs at ~60% normal spectrum speed on an ARM2 machine, and considerably faster on other machines - currently there is no speed control available.
- g) Freeware?
- h) Multitasks (can also single task if required).
- i) Doesn't emulate R register correctly, or certain undocumented instructions.
- j) No other comments.
- k) Available from
<gopher://micros.hensa.ac.uk:70/11/micros/arch/riscos/b/b166>
<ftp://ftp.nvg.unit.no/pub/sinclair/utis/arm/mzx110.spark>

2. !Speccy (Karsten Witt)

- a) Emulates 48K Spectrum.
- b) Doesn't load snapshots. Loads tape files from disk in its own file format.
- c) Doesn't save snapshots. Saves tape files from disk in its own file format.
- d) Comes with application instructions for transferring tape files to/from real tapes through the serial port.
- e) Requirements?
- f) Runs at normal Spectrum speed on an ARM2 machine and very fast on an ARM3 machine. The speed is adjustable.
- g) Freeware?
- h) Has an enhanced keyboard layout - there are keyboard short cuts to Caps Lock, Extended Mode, and also the cursor keys, the keypad, and other symbols (commas, colons etc.) can be entered easily.
- i) Doesn't fully implement the R register - however this doesn't affect most programs.
- j) At the moment does not load/save in any snapshot formats. Apparently, the author is currently working on making it run in the desktop (multi-tasking) and allowing loading of .SNA files.
- k) Available from?

3. Speculator (Dave Lawrence)

- a) Emulates 48K Spectrum.
- b) Loads in its own file format.
- c) Saves in its own file format.
- d) No real tape support.
- e) Requirements?
- f) Is extremely fast on an ARM3 machine. To get normal Spectrum speed an ARM2 machine or an ARM3 machine with the cache turned off is required.

- g) Status not known - not officially released yet.
- h) It is possible to return to the desktop from the emulator, and then re-enter the emulator at the point you left off.
- i) Bad points?
- j) Is not officially available at the moment - any copies circulating are pirate copies. As a result, documentation of the file format is not complete, and the conversion programs supplied with it do not work correctly.
- k) Not yet available?

4. Z80Em (Mike Borchers [borchrds.teaching@physics.oxford.ac.uk])

- a) Emulates 48K Spectrum.
- b) Loads .RAW, .SNA and .Z80 snapshots with level-loader support. Loads tape files from Z80's .TAP files.
- c) Saves .Z80 snapshots and to .TAP tape files.
- d) Can load/save? from/to? sample tapes (.VOC).
- e) Requirements?
- f) Runs a little slowly on an A3010.
- g) Commercial - can be bought along with a CD full of games, the CD coming cheaper this way than if bought separately. However, see question 6 in the Frequently Asked Questions list (section 1 of this document).
- h) Good points?
- i) Level-loader support expects different filenames from the standard - instead of "GAME1.DAT" it wants just "1", etc.
- j) A new version is being worked on that will support v2 and v3 Z80 snapshots, 128K emulation and possibly .SLT snapshots.
- k) Available from Warm Silence Software; their homepage is at <http://www.comlab.ox.ac.uk/oucl/users/robin.watts/wss/index.html>

Emulators for UNIX/Linux/etc.:

1. Spectrum / French emulator (Jean-Francois Lozevis [lozevis@Email.ENST.Fr])

- a) Emulates 48K Spectrum.
- b) Loads .SNA and .Z80 snapshots.
- c) Saves .SNA snapshots.
- d) No real tape support.
- e) Requires X-Windows (X11) and Linux with an 80386 processor (or above) because the emulation code is written in assembler.
- f) Apparently runs at full speed on a 486DX2/66.
- g) Public domain. Author allows modification, just wants to retain his credit as the original author, and wouldn't say no to a postcard.
- h) Sound support for Linux via /dev/audio.

- i) Imperfect emulation; Rhino in Sabre Wulf problems (see the Technical Information section) and some large BASIC programs don't work. Author doesn't know why. Poor user interface.
- j) All messages in French.
- k) Available from
ftp://ftp.nvg.unit.no/pub/sinclair/utils/other/french_spectrum.zip

2. xzx v1.0 (Des Herriott [dnh@mfltd.co.uk but no reponse from this address])

- a) Emulates 48K/128K/+3 Spectrums, Kempston joystick (with keys).
- b) Loads .SNA and .Z80 snapshots with level-loader support, tape files from UNIX files.
- c) Saves .SNA and .Z80 snapshots, tape files to UNIX files.
- d) No real tape support.
- e) Requires X-Windows (X11). Makes heavy use of the MIT-SHM X11 extension, and works a good deal slower without it.
- f) Needs approx 486/33 to run at the correct speed (this is from general impressions, not exact timings). Sparc 10 or DEC Alpha will run it very fast.
- g) Freeware.
- h) Sound support for Sparc and NEC EWS workstations via /dev/audio. Sound support for Linux via direct speaker manipulation. X server bell audio for most X servers (traps ROM BEEP routine only).
- i) Poor user interface; e.g. no file selector - you have to move to your terminal window and type in name of snapshots to load/save.
- j) Author seems to have disappeared! Email address above probably no good.
- k) Available from
<http://hpux.csc.liv.ac.uk/hpux/X11/Misc/xzx-0.5.1.html>

3. xzx v2.0 - modified version (Erik Kunze [Erik.Kunze@philosys.de] & Des Herriott)

- a) Emulates 48/128/+3 with Multiface 128, Interface 1, Kempston Joystick.
- b) Loads .SNA and .Z80 (up to v3.03 format) snapshots with level-loader support, tape files from UNIX files.
- c) Saves .SNA and .Z80 snapshots, tape files from UNIX files.
- d) No real tape support.
- e) Requires X11, Linux (for AY emulation).
- f) A bit faster than original xzx 1.0.2.
- g) Still freeware.
- h) Sound support for Sparc via /dev/audio. Sound support for Linux via direct speaker manipulation. AY emulation via /dev/dsp (Linux only). X server's bell audio for most X servers (traps the ROM BEEP routine only).
- i) Still without a file selector. Some known bugs, that Erik has not yet found.
- j) Erik still needs beta testers! Mail to specy@fantasy.muc.de. This

could be the very first Spectrum +3 emulator...

k) Soon to be available from:

<ftp://ftp.nvg.unit.no/pub/sinclair/utils/other/xzx-2.0.0beta.tar.gz>

4. xz80 v0.1e (Ian Collier [Ian.Collier@comlab.ox.ac.uk])

- a) Emulates 48K Spectrum and ZX Printer (via .XBM image files!).
- b) Loads .SNA and .Z80 snapshots with level-loader support, tape files from disk or from Z80's .TAP files.
- c) Saves .SNA snapshots and tape files to disk or to Z80's .TAP files.
- d) Real tape support not known - mentions sound input (e.g. via /dev/audio device) but I am unsure what Ian means by this.
- e) Requires X-Windows (X11) or SVGALib (for Linux console version). MIT-SHM extension also probably a good idea.
- f) Faster than xzx and x128. The Linux console version runs at about 35% real Spectrum speed on a 486SX/25. True-speed emulation without hogging the CPU on fast machines.
- g) Freeware (GNU Public License).
- h) Realistic sound output via /dev/audio. Emulates 'stripey border' effect - very important ;-)
- i) Poor user interface; e.g. no file selector - you have to move to your terminal window and type in name of snapshots to load/save.
- j) No other comments.
- k) Available from author's homepage at <http://www.comlab.ox.ac.uk/oucl/users/ian.collier/Spectrum/index.html>

5. x128 v0.4 (James McKay [com40014@paisley.ac.uk] until June 1996)

- a) Emulates 48K/128K Spectrum.
- b) Loads .SNA, .Z80 and .SLT snapshots, and from Z80's .TAP tape files.
- c) Saves .Z80 and .SLT snapshots.
- d) No real tape support.
- e) Requires X-Windows (X11). MIT-SHM extension also a very good idea.
- f) Rather slow; about 25-35% of real Spectrum speed on a 486SX/25 Linux box.
- g) Freeware.
- h) Hi-res emulation of screen and left & right border allowing colours in an 8x1 block (rainbowing). Now has a file selector and much better user interface than previous versions.
- i) Some changeable options require re-compilation.
- j) Also available for DOS.
- k) Available from author's home page (until some time in June 1996) at <http://diana40.paisley.ac.uk:80/~com40014/> as well as ftp://ftp.nvg.unit.no/pub/sinclair/utils/other/x128_0.4.tar.gz

Emulators for Apple Macintosh:

1. MacSpeccy v1.1 (Danny Keogan [djkeogan@unix2.tcd.ie])

- a) Emulates 48K Spectrum.
- b) Loads .SNA and (probably) .Z80 snapshots.
- c) Saves?
- d) No real tape support?
- e) Requires 68040.
- f) Seems to be extremely slow.
- g) Freeware?
- h) Allows copying of Spectrum screen to clipboard.
- i) No flash attribute emulation.
- j) The oldest emulator for the Mac.
- k) Available from
<http://hyperarchive.lcs.mit.edu/HyperArchive/Archive/app/mac-speccy-11.hqx>

2. Mac Spectacle v1.8.2 (Guenter Woigk [kio@vanilla.nbg.sub.org])

- a) Emulates 48K/128K Spectrum, joystick (type?).
- b) Loads .SNA and .Z80 snapshots and .SCR screens, tape files from Z80's .TAP files.
- c) Saves .SNA and .Z80 snapshots, saves .SCR and .PICT screens.
- d) No real tape support; but there is a program called ZX-Loader to enable the use of PowerSpectrum's facilities for this.
- e) Needs MacOS 7.0 or higher, Color QuickDraw and at least a 68020 CPU or a PowerMAC.
- f) Speed?
- g) Freeware?
- h) Single/double/quadruple sized window, all monitor depths (2 and 4 bit slow). High-res screen & border effects. Background or exclusive execution. Better emulation than other emulators for the Mac - so more games run on this one than the others.
- i) Bad points?
- j) The newest and most regularly updated emulator for the Mac.
- k) Available from author's FTP area; get this file first:
<ftp://lst.informatik.uni-erlangen.de//incoming/kio/readme>
or
<ftp://ftp.nvg.unit.no/pub/sinclair/utils/mac/Spectacle-1.8.2.hqx>
and ZX-Loader is available from
ftp://ftp.nvg.unit.no/pub/sinclair/utils/mac/ZX_Loader_1.0.4.hqx

3. PowerSpectrum v1.0 (Bo Lindbergh [d88-bli@nada.kth.se])

- a) Emulates 48K Spectrum, Kempston joystick.

- b) Loads .SNA snapshots.
- c) Saves .SNA snapshots.
- d) Does realtime tape I/O through the microphone jack (this probably requires 44 kHz sound hardware).
- e) Requires a PowerMac - won't run on a 680x0 machine.
- f) Runs at true Spectrum speed.
- g) Freeware?
- h) 1x1, 2x2, 3x3, and 4x4 screen emulation. Has perfect sound. Supports all undocumented Z80 instructions.
- i) Bad points?
- j) No other comments.
- k) Available from
<http://hyperarchive.lcs.mit.edu/HyperArchive/Archive/app/power-spectrum-10-ppc.hqx>

Emulators for NeXTStep:

-
1. NXSpectrum (Ian Stephensen [ian.stephenson@insignia.com] & Ian Collier [Ian.Collier@comlab.ox.ac.uk])
 - a) Emulates 48K Spectrum. ZX Printer still?
 - b) As xz80.
 - c) As xz80.
 - d) As xz80.
 - e) Requirements?
 - f) Speed?
 - g) Freeware (GNU Public License).
 - h) Good points as xz80?
 - i) Bad points as xz80?
 - j) This is a quick hack of xz80 to make it run under NeXTStep. And yes, this is from the same guy who does SoftPC for Insignia!
 - k) Available from
<ftp://pitcher.peak.org/pub/next/submissions/NXSpectrum.app.NIHS.compressed>
 (directory may have changed by the time you read this)

Emulators for Commodore 64: (information from comp.sys.emulators.misc FAQ)

1. Spectrum 48

- a) Emulates Sinclair BASIC only - no Z80 emulation, so no machine code programs work. Microdrive emulated via 1541/1571 disk drive.
- b) Loads Microdrive files from disk.
- c) Saves Microdrive files to disk.
- d) No real tape support?

- e) Requires just a basic C64.
- f) Slow - well, it **is** just BASIC and it **is** on a C64!
- g) Freeware?
- h) Allows C64 users to see some of the superiority of Sinclair's masterpiece (well you try and think of a good point!). And Sinclair BASIC is somewhat more useful than Commodore's attempt.
- i) Useless for all but the most BASIC of games (pun intended).
- j) No other comments.
- k) Available from?

WHERE IS...?

1. Imagine/Denton Designs [Stonkers, Frankie goes to Hollywood, etc.]
 - o Dougie Burns (programmer)
Last seen in Glasgow as a used car salesman.
 - o Marc Dawson (programmer)
Project manager for Barbie the Video Game for software house in Manchester.
 - o John Gibson (programmer - ZZOOM/Stonkers)
Currently working in the Chester office of Psygnosis and has been for some time.
 - o Fred Gray (musician)
Became programmer in Liverpool. Works for Psygnosis, as do most of Denton Designs. Status uncertain since Sony's take over of Psygnosis.
 - o Dave Lawson (director)
Last seen running Kinetica Software in Birkenhead.
 - o Ally Noble (artist) and John Heap (programmer)
Still own and run Denton Designs.
 - * INFO BY:
Stuart Fotheringham [stuarto@stoo.demon.uk]
Andrew Toone [andrewt@smallworld.co.uk]
Ralph Ferneyhough [ralph@datel.co.uk]

2. Gargoyle Games/Faster Than Light [Heavy On The Magick, Lightforce etc.]
 - o Roy Carter
Roy Carter is a registered user of Z80 and has given spoken permission to distribute the Gargoyle games. They are (were?) doing things for Psygnosis - not all of them game writing.
 - * INFO BY:
Brian Gaff [briang@bgserv.demon.co.uk]
Andrew Toone [andrewt@smallworld.co.uk]

3. Design Design [Halls Of The Things, Dark Star etc.]
 - o Simon Brattel
Runs his own company producing system software
 - o Graham Stafford
Works for Psygnosis, as do most of Denton Designs.
 - * INFO BY:
Andrew Toone [andrewt@smallworld.co.uk]
Jon Ritman [jritman@cix.compulink.co.uk]

4. LERM [various tape copiers]
 - o Robert Evans
"Not only do I remember Lerm, I did some work for them. Lerm itself was a husband and wife company, in the back room, in the traditional fashion. The husband was a high-flying software type a.k.a my Maths teacher at

school."

"There was a tendency amongst my group of schoolfriends at the time to use LERM programs more than anything else, because some of us helped Mr Evans (our maths teacher at the time) to write them. If anyone's interested, the company name came from a mixture of Robert Evans' and his wife's initials."

* INFO BY:

David Smith [davidsm@x.co.uk]

Scott Wilding [scott@fizban.demon.co.uk]

5. Ultimate Play The Game/Ashby Computers & Graphics (ACG) [Jetpac, Knighlore etc.]

o Calling themselves Rare and producing console games for Nintendo, e.g. Donkey Kong Country and Killer Instinct for the new Nintendo Ultra 64.

* INFO BY:

Dylan Cuthbert [dylan@takoyaki.demon.co.uk]

6. Realtime Graphics [3D StarStrike, Starglider, Carrier Command etc.]

o Jez San

Developing 3D systems for anyone with with enough hardware (Nintendo et. al.)

* INFO BY:

Andrew Toone [andrewt@smallworld.co.uk]

7. Vortex [Android, TLL, Highway Encounter etc.]

o Mark Haigh-Hutchinson

Working for LucasArts, currently on Shadows of the Empire for the Nintendo 64. Mark organises the Vortex Emulation Package for several machines, including the Spectrum. This contains a number of the Vortex games and is freely distributed.

o Costa Panayi

Doing some consultancy design work for a children's toys company in the south of England (believed to be Chicco).

* INFO BY:

Arnt Gulbrandsen [agulbra@flode.nvw.unit.no]

Miguel Melo [mmelo@ea.com]

Mark Haigh-Hutchinson [mhh@lucasarts.com]

8. OTHERS:

o Marcus Berkman [YS or SU writer - aka Dr. Berkman]

Now writing for the Mail on Sunday.

* INFO BY:

Keith Willoughby [keith@cowtown.demon.co.uk]

o John F. Cain [Booty]

Still lives on the Isle of Man, as he did when he wrote Booty. He has a

gold cassette of Booty on his wall!

* INFO BY:

Mark Walker [M.S.Walker@cs.cf.ac.uk]

o Alan Cox [Blizzard Pass]

Now doing network stuff; has thought about a Linux version of Blizzard Pass. He has a home page where you can retrieve some adventure stuff (but not Blizzard Pass itself, as that is owned by whoever bought the rights from Tynesoft) - <http://shadow.cabi.net/>

* INFO BY:

Alan Cox [Alan.Cox@linux.org]

o Mel Croucher [ID]

Writing for magazines; including the Great Moments In Computing strip for Computer Shopper.

* INFO BY:

Damien Burke [D.M.Burke-CSSE94@cs.bham.ac.uk until June 14th 1996]

o Robert Erskine [anyone know anything he wrote?] - robert@smallworld.co.uk
Working for Smallworld Cambridge on GIS.

* INFO BY:

Stephen Smith [stevo@jonlan.demon.co.uk]

o Eugene Evans

Currently working in Chicago at Viacom New Media. He's been there for a good few years before which it was Icom Simulations, Inc. the people who did the CD-ROM Sherlock Holmes games, T-MON for the Mac, and Dracula Unleashed.

* INFO BY:

phyd@interaccess.com [Brian Leake]

o Glenn Flood [tape inlay art for Astroblaster and others]

Programming and drawing still; has a nice web site showing off a lot of his artwork at <http://www.fttech.net/~floodnet/>

* INFO BY:

Glenn Flood [glenn@floodnet.demon.co.uk]

o Julian Gollop and Nick Gollop [Chaos, LaserSquad, Rebelstar]

Owners and programmers of Mythos (UFO, X-COM).

* INFO BY:

Miguel Melo [mmelo@ea.com]

o Simon Goodwin [Crash Tech Tips writer]

Was still writing for Sinclair QL World when the final issue (July 94) came out. Simon Goodwin writes for Amiga magazines and Computer Shopper. He was also seen at the Last Sinclair and SAM show in Gloucester. Messing

with both Speccy emulator for Amiga and hardware Speccy emulation for IBM PC.

* INFO BY:

Jenni the Satsuma [yockneyj@cs.man.ac.uk]

Brian Gaff [briang@bgserv.demon.co.uk]

o John Hollis, Nick Lambert

John Hollis founded Hollis Research, who develop MIDI sequencers (including "Trackman" for AtariST).

* INFO BY:

Stephen K. Mulrine [robotron@moroder.demon.co.uk]

o Graeme Kidd [Crash writer]

News editor for Computer Shopper magazine. At least I think it's the same guy...

* INFO BY:

Damien Burke [D.M.Burke-CSSE94@cs.bham.ac.uk until June 14th 1996]

o Tony Mack [Gyroscope, Paper Boy, Super Sprint, Championship Sprint]

Now writes games for the Sega Saturn (currently converting Cybersled to it).

* INFO BY:

Mark Gordon [100306.3045@compuserve.com]

o Teresa Maughan [Your Sinclair editor - aka T'zer]

Now writing for The Mac magazine, either that or someone with the same name is...

* INFO BY:

Oops, I've lost the posting. Gimme a shout if you were the one who did that Alta Vista search!

o Fergus McNeill [Delta 4 - Bored Of The Rings, The Boggit, etc.]

Head of design at Sales Curve Interactive (SCi), ex-Storm. SCi have just released a Point and Click called "Kingdom 'O Magic" which was designed by Fergus. Fergus has kindly agreed to free distribution of the Delta 4 games, as long as it is *free* - no money may change hands. He cannot speak for any of his games with the Abstract Concepts logo, as they are owned by Activision.

* INFO BY:

Chris Wild [chrisw@sci.co.uk]

o Neil Pawson [Mighty Magus] - neil@smallworld.co.uk

Working for Smallworld Cambridge on GIS.

* INFO BY:

Stephen Smith [stevo@jonlan.demon.co.uk]

- o Dominic Prior [part of Gyron team] - dominic@smallworld.co.uk
Working for Smallworld Cambridge on GIS.
- * INFO BY:
Stephen Smith [stevo@jonlan.demon.co.uk]

- o Jon Ritman [co-author of Batman, Head over Heels, Matchday, etc.]
Jon Ritman is reading/posting to comp.sys.sinclair and is the director of a new(ish) videogame company in the UK called Cranberry Source. Contrary to previous assertions in this FAQ, he definitely *didn't* do a Computer Science degree at UCL and he and Bernie Drummond wrote only one Gameboy game (Monster Max). Jon doesn't care if you copy his old games, but then as he says - he doesn't own the rights to them.
- * INFO BY:
Vartan Narinian [vsa@ic.ak.uk]
Peter Watsons [pwats@cyberspace.com]
Jon Ritman [jritman@cix.compulink.co.uk]

- o Mike Singleton [Lords Of Midnight, Doomdark's Revenge etc.]
Mike Singleton is alive and well and writing Lords Of Midnight games. for the IBM PC. Chris Wild did PC versions of Lords of Midnight and Doomdark's Revenge.
- * INFO BY:
Chris Wild [chrisw@sci.co.uk]

- o Matthew Smith [Styx, Manic Miner, Jet Set Willy]
Nothing concrete on the father of the miner :(
Previous rumours range from him working in some computer store in the south of England to the latest; a motorbike mechanic called Mat living in a commune in the Netherlands; he mentioned to someone that he "wrote some video games in the 80's, got burnt out, ended up in a mental institution".
- * INFO BY:
Stephen Smith [stevo@jonlan.demon.co.uk]

- o Phil South [aka Snouty; Your Sinclair writer]
Writes for Computer Shopper magazine.
- * INFO BY:
Damien Burke [D.M.Burke-CSSE94@cs.bham.ac.uk until June 14th 1996]

- o Kevin Thoms [Football Manager]
Currently working as a Windows consultant/programmer.
- * INFO BY:
Miguel Melo [mmelo@ea.com]

- o Sandy White (Ant Attack, Zombie Zombie)
Currently working on a super new top secret engine while laying about on

a beach in Greece.

* INFO BY:

Jon Ritman [jritman@cix.compulink.co.uk

9. Spectrum people reading comp.sys.sinclair

- o Simon Cooke [csl@fs2.ee.umist.ac.uk] was Your Sinclair's technical editor, and has put up extracts from YS issues at his web site at <http://jumper.mcc.ac.uk/~simonc/ys>.
- o Alan Fothergill [afotherg@oracle.com] from Imagine.
- o Stuart Fotheringham [stuarto@stoo.demon.co.uk] did work on those beautiful graphics in Odin games Nodes of Yesod, Robin O' the Wood, Arc of Yesod, and Heartland.
- o Paul Holmes [paulh@unlimited.com] from Elite wrote and co-wrote Robotron 2084, Bomb Jack I & II, Wild West Hero, Dustman, designed and co-designed Grand National, Frank Bruno's Boxing and others.
- o Mark Haigh-Hutchinson [mhh@lucasarts.com] of Vortex pops up occasionally, mostly to mention updates to the Vortex Emulation Package, which is a collection of Vortex games (TLL, Highway Encounter etc.) and instructions for them, and is freely distributable.
- o John Ritman [jritman@cix.compulink.co.uk] of Head Over Heels fame. Also co-wrote Batman, Matchday and other excellent games with Bernie Drummond.
- o Duncan Sinclair [sinclair@dis.strath.ac.uk] worked on an unfinished Tapper with Paul Holmes.

FILE FORMATS

This section will be expanded as I get the time to do so; if you are desperate for information about a particular format right now, a good source of information is the source to SPConv v1.10; available from (among other locations):
<ftp://ftp.nvg.unit.no/pub/sinclair/utills/generic/spcnv110.zip>

Snapshot formats

Those handled by SPConv v1.10 are marked with an asterisk (*):

- .ACH or .archimedes *
Snapshots used by !Speccy. .ACH is the extension SPConv uses for these files; so that filename-challenged operating systems like DOS can handle such files for conversion purposes.
- .PRG * Program file used by Specem.
- .RAW * Raw memory dump from a real Spectrum; just the 48Kb of RAM and a CODE header on the front.
- .SEM Snapshots used by ZX Spectrum-Emulator, the German PC-based emulator.
- .SIT * Situation file used by Sinclair v2.00.
- .SLT Super level loader snapshot. Used by x128 and WSpecEm at time of writing. Basically a v2/3 Z80 snapshot with level data appended.
- .SNA or .snap or .snapshot *
Mirage Microdrive snapshot format, used by most emulators.
- .SNA 128Kb version of SNA. Distinguished by file size of 131103 bytes instead of 49179 bytes for 48Kb version.
- .SNX * Extended version of SNA, used by Speccy on the Atari ST.
- .SP * Snapshots used by SPECTRUM.
- .SP * Snapshots used by VGASPEC.
- .SP Snapshots used by ZX Spectrum (Mac emulator).
- .Z80 * Snapshots used by Z80 and several other emulators. Three versions in existence, the latest of which (v3) is not supported by all these emulators. Very flexible; support for SamRam, 128K snapshots, etc.
- .ZX * Snapshots used by KGB.

For the purposes of these descriptions, the following definitions apply:

byte = byte-sized variable; word = 2 bytes, long = 4 bytes. All stored in little-endian (Intel) format unless otherwise stated.

a) .SLT (super level loader trap used by x128 v0.4+, WSpecEm etc.)

The level loader trap has one annoying disadvantage; lots of extra files lying around for each game. The super level loader was thought up (by me, natch :-)) to replace this multi-file format with a single snapshot file containing all the level data files. It has been designed in co-operation

with James McKay (x128), Gerton Lunter (Z80), Rui Ribeiro (WSpecEm) and Darren Salt (helping with Z80Em), so is well-supported already. The format was designed with future expansion in mind, as you will see.

Size	Description
varies bytes	Z80 snapshot (version 2+)
3 bytes	Three null bytes (compatibility; see below)
3 bytes	"SLT" (signature)
---- the following blocks make up a table to access the data files ----	
2 word	data type (0 = end of table, 1 = level data)
2 word	data identifier (for type 1 this is level number)
4 long	data length
2 word	data type (0 = end of table, 1 = level data)
2 word	data identifier (for type 1 this is level number)
4 long	data length
... and so on	
---- the following blocks are the data files themselves -----	
varies bytes	data
varies bytes	data
... and so on	

The three null bytes after the end of the snapshot are for compatibility reasons; older versions of Z80 would crash if the extra data was just appended to the snapshot. With these three null bytes, they just complain about an error in the snapshot file instead. This, of course, presumes you have renamed the .SLT file to .Z80 and attempted to load it into an older emulator!

After the "SLT" signature, there is a table of data types and sizes. Only data types 0 (end of table) and 1 (level data) are supported at the moment, so if other values are encountered an emulator should ignore that data block.

To read a level data file using .SLT, the emulator should find the correct entry in the table (type = 1, identifier matching the A register when the ED/FB instruction was encountered), get its size from the table and calculate its position from the total of sizes of data blocks previous to the required one, added to the position of the end of the table. E.g., to load level 2 from a .SLT snapshot with this table:

Position	Size	Value	Description
40000	2	1	data type = level data
40002	2	1	data identifier = level 1

40004	4	256	data length = 256 bytes
40008	2	1	data type = level data
40010	2	2	data identifier = level 2
40012	4	128	data length = 128 bytes
40016	2	0	data type = end of table
40018	2	*	data identifier = unused (may as well be zero)
40020	4	*	data length = unused (may as well be zero)
40024	256	*	data block for level 1
40280	128	*	data block for level 2

 (* = could be anything)

So, the size of level 2 is 128 bytes, and its located at the end of the table (40024) + the length of all previous blocks (just 256 here) = 40280.

Level data is packed in the same way as Z80 snapshot memory banks are.

The trainspotter award seekers of you may wonder why a whole word is used for the data identifier; after all, this is the level number and is held in the A register, so could be just a byte. For level data, correct. But future expansion is better served by a word. For the same reasons, the data length is held as a long word instead of just a word; level data will never exceed 64Kb (indeed, could not even be as much as 48Kb), but future data types may do so. One example; embedding a scan of a game's inlay card in the file is possible, and that file could exceed 64Kb easily.

See <http://www.cs.bham.ac.uk/~dmb/specocy/edfb.html> for my own proposals for future data types for inclusion in .SLT snapshots.

b) .SNA, .snap or .snapshot (Mirage Microdrive format used by many emulators)

This format is the most well-supported of all snapshot formats (though Z80 is close on its heels) but has a drawback:

As the program counter is pushed onto the stack so that a RETN instruction can restart the program, 2 bytes of memory are overwritten. This will usually not matter; the game (or whatever) will have stack space that can be used for this. However, if this space is all in use when the snap is made, memory below the stack space will be corrupted. According to Rui Ribeiro, the effects of this can sometimes be avoided by replacing the corrupted bytes with zeros; e.g. take the PC from the stack pointer, replace that word with 0000 and then increment SP. This worked with snapshots of Batman, Bounder and others which had been saved at critical points. Theoretically, this problem could cause a complete crash on a real Spectrum if the stack pointer happened to be at address 16384; the push would try and write to the ROM. How different

emulators handle this is not something I know...

When the registers have been loaded, a RETN command is required to start the program. IFF2 is short for interrupt flip-flop 2, and for all practical purposes is the interrupt-enabled flag. Set means enabled.

Offset	Size	Description
0	1	byte I
1	8	word HL,DE,BC,AF
9	10	word HL,DE,BC,IY,IX
19	1	byte Interrupt (bit 2 contains IFF2, 1=EI/0=DI)
20	1	byte R
21	4	words AF,SP
25	1	byte IntMode (0=IM0/1=IM1/2=IM2)
26	1	byte BorderColor (0..7, not used by Spectrum 1.7)
27	49152	bytes RAM dump 16384..65535

Total: 49179 bytes

c) .SNA (128Kb version) (SP_EMU)

This is simply the SNA format extended to include the extra memory banks of the 128K/+2 machines, and fixes the problem with the PC being pushed onto the stack - now it is located in an extra variable in the file (and is not pushed onto the stack at all) The first 49179 bytes of the snapshot are otherwise exactly as described above, so the full description is:

Offset	Size	Description
0	27	bytes SNA header (see above)
27	16Kb	bytes RAM bank 5 \
16411	16Kb	bytes RAM bank 2 } - as standard 48Kb SNA file
32795	16Kb	bytes RAM bank n / (currently paged bank)
49179	2	word PC
49181	1	byte port 7FFD setting
49182	1	byte (unknown - padding for above byte?)
49183	16Kb	bytes remaining RAM banks in ascending order

...

Total: 131103 or 147487 bytes

The third RAM bank saved is always the one currently paged, even if this is page 5 or 2 - in this case, the bank is actually included twice. The remaining RAM banks are saved in ascending order - e.g. if RAM bank 4 is

paged in, the snapshot is made up of banks 5, 2 and 4 to start with, and banks 0, 1, 3, 6 and 7 afterwards. If RAM bank 5 is paged in, the snapshot is made up of banks 5, 2 and 5 again, followed by banks 0, 1, 3, 4, 6 and 7.

- d) .SP file format used in "ZX Spectrum", the ZX Spectrum emulator for Macintosh from Lorenzo Jose Ayuda Serrano.

Offset	Size	Description
0	2	byte "SP" (signature)
2	2	word Program length in bytes (49152 bytes)
4	2	word Program location (16384)
6	8	word BC,DE,HL,AF
14	4	word IX,IY
18	8	word BC',DE',HL',AF'
26	2	byte R,I
28	4	word SP,PC
32	2	word 0 (reserved for future use)
34	1	byte Border color
35	1	byte 0 (reserved for future use)
36	2	word Status word

Status word:

Bit	Description
15-8	Reserved for future use
7-6	Reserved for internal use (0)
5	Flash: 0=INK/1=PAPER
4	Interrupt pending for execution
3	Reserved for future use
2	IFF2 (internal use)
1	Interrupt Mode: 0=IM1/1=IM2
0	IFF1: 0=DI/1=EI

- e) .ZX (KGB v.1.2-1.3) [Contributed by Troels Norgaard]
All values stored in big-endian format; on 680x0 the most significant byte goes first.

Offset	Size	Description
0	49284	bytes RAM dump 16252..65535
49284	132	bytes unused, make 0
49416	10	word 10,10,4,1,1 (different settings)
49426	1	byte InterruptStatus (0=DI/1=EI)

49427	2	byte	0,3
49429	1	byte	ColorMode (0=BW/1=Color)
49430	4	long	0
49434	16	word	BC,BC',DE,DE',HL,HL',IX,IY
49450	2	byte	I,R
49452	2	word	0
49454	8	byte	0,A',0,A,0,F',0,F
49462	8	word	0,PC,0,SP
49470	2	word	SoundMode (0=Simple/1=Pitch/2=RomOnly)
49472	2	word	HaltMode (0=NoHalt/1=Halt)
49474	2	word	IntMode (-1=IM0/0=IM1/1=IM2)
49476	10	bytes	unused, make 0

Total: 49486 bytes

f) .Z80 (Z80) [from Z80 documentation]

The old .Z80 snapshot format (for version 1.45 and below) looks like this:

Offset	Length	Description
0	1	A register
1	1	F register
2	2	BC register pair (LSB, i.e. C, first)
4	2	HL register pair
6	2	Program counter
8	2	Stack pointer
10	1	Interrupt register
11	1	Refresh register (Bit 7 is not significant!)
12	1	Bit 0 : Bit 7 of the R-register
		Bit 1-3: Border colour
		Bit 4 : 1=Basic SamRom switched in
		Bit 5 : 1=Block of data is compressed
		Bit 6-7: No meaning
13	2	DE register pair
15	2	BC' register pair
17	2	DE' register pair
19	2	HL' register pair
21	1	A' register
22	1	F' register
23	2	IY register (Again LSB first)
25	2	IX register
27	1	Interrupt flipflop, 0=DI, otherwise EI
28	1	IFF2 (not particularly important...)
29	1	Bit 0-1: Interrupt mode (0, 1 or 2)
		Bit 2 : 1=Issue 2 emulation

- Bit 3 : 1=Double interrupt frequency
- Bit 4-5: 1=High video synchronisation
3=Low video synchronisation
0,2=Normal
- Bit 6-7: 0=Cursor/Protek/AGF joystick
1=Kempston joystick
2=Sinclair 2 Left joystick (or user defined, for version 3 .Z80 files)
3=Sinclair 2 Right joystick

Because of compatibility, if byte 12 is 255, it has to be regarded as being 1. After this header block of 30 bytes the 48K bytes of Spectrum memory follows in a compressed format (if bit 5 of byte 12 is one). The compression method is very simple: it replaces repetitions of at least five equal bytes by a four-byte code ED ED xx yy, which stands for "byte yy repeated xx times". Only sequences of length at least 5 are coded. The exception is sequences consisting of ED's; if they are encountered, even two ED's are encoded into ED ED 02 ED. Finally, every byte directly following a single ED is not taken into a block, for example ED 6*00 is not encoded into ED ED ED 06 00 but into ED 00 ED ED 05 00. The block is terminated by an end marker, 00 ED ED 00.

That's the format of .Z80 files as used by versions up to 1.45. Starting from version 2.0, a different format is used, since from then on also 128K snapshots had to be supported. This new format is used for all snapshots, either 48K or 128K.

Version 2.01 and 3.0 .Z80 files start with the same 30 byte header as old .Z80 files used. Bit 4 and 5 of the flag byte have no meaning anymore, and the program counter (byte 6 and 7) are zero to signal a version 2.01 or version 3.0 snapshot file.

After the first 30 bytes, the additional header follows:

Offset	Length	Description
* 30	2	Length of additional header block (see below)
* 32	2	Program counter
* 34	1	Hardware mode (see below)
* 35	1	If in SamRam mode, bitwise state of 74ls259. For example, bit 6=1 after an OUT 31,13 (=2*6+1) If in 128 mode, contains last OUT to 7ffd
* 36	1	Contains 0FF if Interface I rom paged
* 37	1	Bit 0: 1 if R register emulation on Bit 1: 1 if LDIR emulation on
* 38	1	Last OUT to fffd (soundchip register number)

* 39	16	Contents of the sound chip registers
55	2	Low T state counter
57	1	Hi T state counter
58	1	Flag byte used by Spectator (QL spec. emulator) Ignored by Z80 when loading, zero when saving
59	1	0FF if MGT Rom paged
60	1	0FF if Multiface Rom paged. Should always be 0.
61	1	0FF if 0-8191 is RAM
62	1	0FF if 8192-16383 is RAM
63	10	5x keyboard mappings for user defined joystick
73	10	5x ascii word: keys corresponding to mappings above
83	1	MGT type: 0=Disciple+Epson,1=Discipls+HP,16=Plus D
84	1	Disciple inhibit button status: 0=out, 0ff=in
85	1	Disciple inhibit flag: 0=rom pageable, 0ff=not

The value of the word at position 30 is 23 for version 2.01 files, and 54 for version 3.0 files. The starred fields are the ones that constitute the version 2.01 header, and their interpretation has remained unchanged except for byte 34:

Value:	Meaning in v2.01	Meaning in v3.0
0	48k	48k
1	48k + If.1	48k + If.1
2	SamRam	48k + M.G.T.
3	128k	SamRam
4	128k + If.1	128k
5	-	128k + If.1
6	-	128k + M.G.T.

The hi T state counter counts up modulo 4. Just after the ULA generates its once-in-every-20-ms interrupt, it is 3, and is increased by one every 5 emulated milliseconds. In these 1/200s intervals, the low T state counter counts down from 17472 to 0, which make a total of 69888 T states per frame.

The 5 ascii words (high byte always 0) at 73-82 are the keys corresponding to the joystick directions left, right, down (!), up (!), fire respectively. Shift, Symbol Shift, Enter and Space are denoted by [,],/,\ respectively. The ascii values are used only to display the joystick keys; the information in the 5 keyboard mapping words determine which key is actually pressed (and should correspond to the ascii values). The low byte is in the range 0-7 and determines the keyboard row. The high byte is a mask byte and determines the column. Enter for example is stored as 0x0106 (row 6 and column 1) and 'g' as 0x1001 (row 1 and column 4).

Byte 60 must be zero, because the contents of the Multiface RAM is not saved in the snapshot file. If the Multiface was paged when the snapshot was saved, the emulated program will most probably crash when loaded back.

Bytes 61 and 62 are a function of the other flags, such as byte 34, 59, 60 and 83.

Hereafter a number of memory blocks follow, each containing the compressed data of a 16K block. The compression is according to the old scheme, except for the end-marker, which is now absent. The structure of a memory block is:

Byte	Length	Description
0	2	Length of data (without this 3-byte header)
2	1	Page number of block
3	[0]	Compressed data

The pages are numbered, depending on the hardware mode, in the following way:

Page	In '48 mode	In '128 mode	In SamRam mode
0	48K rom	rom (basic)	48K rom
1	Interface I, Disciple or Plus D rom, according to setting		
2	-	rom (reset)	samram rom (basic)
3	-	page 0	samram rom (monitor,..)
4	8000-bfff	page 1	Normal 8000-bfff
5	c000-ffff	page 2	Normal c000-ffff
6	-	page 3	Shadow 8000-bfff
7	-	page 4	Shadow c000-ffff
8	4000-7fff	page 5	4000-7fff
9	-	page 6	-
10	-	page 7	-
11	Multiface rom	Multiface rom	-

In 48K mode, pages 4,5 and 8 are saved. In SamRam mode, pages 4 to 8 are saved. In '128 mode, all pages from 3 to 10 are saved. This version saves the pages in numerical order. There is no end marker.

Tape formats:

- .BLK Tape format used by Sinclair v2.00; seems to be identical to Z80's .TAP files.

- .SPC Tape format used by SP, the Polish emulator.
- .TAP Tape format used by Z80; supports headerless files and not much else.
- .TAP Tape format used by Warajevo - supports lots of features; turbo-load, headerless files, etc.
- .VOC Straight sound sample of a tape; used by several emulators.
- .ZXS Very flexible tape format, not actually used by any emulators - used to store real Spectrum tapes in a digital format. All come from the ZX Spectrum Software Museum (<http://relcom.eu.net/zx/>).

a) .TAP and .BLK (Z80, Sinclair, several others) [from Z80 documentation]

The .TAP files contain blocks of tape-saved data. All blocks start with two bytes specifying how many bytes will follow (not counting the two length bytes). Then raw tape data follows, including the flag and checksum bytes. The checksum is the bitwise XOR of all bytes including the flag byte. For example, when you execute the line SAVE "ROM" CODE 0,2 this will result:

```

|----- Spectrum-generated data -----|      |-----|
13 00 00 03 52 4f 4d 7x20 02 00 00 00 00 80 f1 04 00 ff f3 af a3

```

```

^^^^^..... first block is 19 bytes (17 bytes+flag+checksum)
^^... flag byte (A reg, 00 for headers, ff for data blocks)
^^ first byte of header, indicating a code block

```

```

file name ..^^^^^^^^^^^^^^^^^^
header info .....^^^^^^^^^^^^^^^^^^
checksum of header .....^^
length of second block .....^^^^^^
flag byte .....^^
first two bytes of rom .....^^^^^^
checksum (checkbittoggle would be a better name!).....^^

```

Note that it is possible to join .TAP files by simply stringing them together, for example COPY /B FILE1.TAP + FILE2.TAP ALL.TAP

For completeness, I'll include the structure of a tape header. A header always consists of 17 bytes:

Byte	Length	Description
0	1	Type (0,1,2 or 3)
1	10	Filename (padded with blanks)
11	2	Length of data block
13	2	Parameter 1
15	2	Parameter 2

The type is 0,1,2 or 3 for a Program, Number array, Character array or Code file. A SCREENS

file is regarded as a Code file with start address 16384 and length 6912 decimal. If the file is a Program file, parameter 1 holds the autostart line number (or a number ≥ 32768 if no LINE parameter was given) and parameter 2 holds the start of the variable area relative to the start of the program. If it's a Code file, parameter 1 holds the start of the code block when saved, and parameter 2 holds 32768. For data files finally, the byte at position 14 decimal holds the variable name.

Other formats:

- .DAT Data files used by level-loader versions of a game (Z80Em does not use a .DAT extension at all; instead files are just numbered, e.g. "1" instead of "GAME1.DAT").
- .MDR Microdrive cartridge file as used by Spectator, Carlo Delhez' Speccy emulator for the QL, and other emulators - xzx and Z80.
- .OUT OUT logs from Z80.
- .SCR Screendumps from Z80 (actually, may be any extension). This is just the 6912 bytes of the Spectrum's screen; 6144 bytes of pixel data (in that horrendous layout we all know and love) followed by 768 bytes of attribute data.

a) .MDR (Spectator, xzx, Z80) [from Z80 documentation]

The following information is adapted from Carlo's documentation. It can also be found in the 'Spectrum Microdrive Book', by Ian Logan (co-writer of the excellent 'Complete Spectrum ROM Disassembly').

A cartridge file contains 254 'sectors' of 543 bytes each, and a final byte flag which is non-zero if the cartridge is write protected, so the total length is 137923 bytes. On the cartridge tape, after a GAP of some time the Interface I writes 10 zeros and 2 FF bytes (the preamble), and then a fifteen byte header-block-with-checksum. After another GAP, it writes a preamble again, with a 15-byte record-descriptor-with-checksum (which has a structure very much like the header block), immediately followed by the data block of 512 bytes, and a final checksum of those 512 bytes. The preamble is used by the Interface I hardware to synchronise, and is not explicitly used by the software. The preamble is not saved to the microdrive file:

Offset	Length	Name	Contents

0	1	HDFLAG	Value 1, to indicate header block
1	1	HDNUMB	sector number (values 254 down to 1)
2	2		not used
4	10	HDNAME	microdrive cartridge name (blank padded)
14	1	HDCHK	header checksum (of first 14 bytes)
15	1	RECFLG	- bit 0: always 0 to indicate record block - bit 1: set for the EOF block - bit 2: reset for a PRINT file

- bits 3-7: not used (value 0)

16	1	RECNUM	data block sequence number (value starts at 0)
17	2	RECLEN	data block length (<=512, LSB first)
19	10	RECNAM	filename (blank padded)
29	1	DESCHK	record descriptor checksum (of previous 14 bytes)
30	512		data block
542	1	DCHK	data block checksum (of all 512 bytes of data block, even when not all bytes are used)

254 times

(Actually, this information is 'transparent' to the emulator. All it does is store 2 times 254 blocks in the .MDR file as it is OUTed, alternatingly of length 15 and 528 bytes. The emulator does check checksums, see below; the other fields are dealt with by the emulated Interface I software.)

A used record block is either an EOF block (bit 1 of RECFLG is 1) or contains 512 bytes of data (RECLEN=512, i.e. bit 1 of MSB is 1). An empty record block has a zero in bit 1 of RECFLG and also RECLEN=0. An unusable block (as determined by the FORMAT command) is an EOF block with RECLEN=0.

The three checksums are calculated by adding all the bytes together modulo 255; this will never produce a checksum of 255. Possibly, this is the value that is read by the Interface I if there's no or bad data on the tape.

In normal operation, all first-fifteen-byte blocks of each header or record block will have the right checksum. If the checksum is not right, the block will be treated as a GAP. For instance, if you type OUT 239,0 on a normal Spectrum with interface I, the microdrive motor starts running and the cartridge will be erased completely in 7 seconds. CAT 1 will respond with 'microdrive not ready'. Try it on the emulator...

b) .SCR (Z80, WSpecEm)

These files are just Spectrum screen dumps, and are simply the 6912 bytes of pixel and attribute data found at address 16384, stored on disk in exactly the same way as they are stored in memory.

TECHNICAL INFORMATION

Z80 CPU

I had hoped to add some information from several threads on the newsgroup (Z80 emulation related facts, BIT and the flags and Z80 interrupts) but just never got the time, and probably won't have the opportunity to do so now for several months - if the mean time someone wants to compile the relevant information together so I can just paste it in, I'd be most grateful ;)

This section is based on the text contributed by Gerton Lunter, author of the "Z80" Spectrum emulator. Marat Fayzullin made some changes which didn't change the content and Ian Collier and several others have contributed some extra information (some of which conflicts with Gertons - further contributions welcome!).

Most Z80 opcodes are one byte long, not counting a possible byte or word operand. The four opcodes CB, DD, ED and FD are shift opcodes: they change the meaning of the opcode following them.

a) CB opcodes:

There are 248 different CB opcodes.

The block CB 30 to CB 37 is missing from the official list. These instructions, usually denoted by the mnemonic SLL, Shift Left Logical, shift left the operand and make bit 0 always one. These instructions are quite commonly used. For example, Bounder and Enduro Racer use them.

b) DD and FD opcodes:

The DD and FD opcodes precede instructions using the IX and IY registers. If you look at the instructions carefully, you see how they work:

2A nn	LD HL,(nn)
DD 2A nn	LD IX,(nn)
7E	LD A,(HL)
DD 7E d	LD A,(IX+d)

A DD opcode simply changes the meaning of HL in the next instruction. If a memory byte is addressed indirectly via HL, as in the second example, a displacement byte is added. Otherwise the instruction simply acts on IX instead of HL (A notational awkwardness, that will only bother assembler and disassembler writers: JP (HL) is not indirect; it should have been denoted by JP HL). If a DD opcode precedes an instruction that doesn't use the HL register pair at all, the instruction is executed as usual. However, if the instruction uses the H or L register, it will now use the high or low halves of the IX register! Example:

44	LD B,H
FD 44	LD B,IYh

These types of 'undocumented' instructions are used in very many programs. By the way, many

DD or FD opcodes after each other will effectively be NOPs, doing nothing except repeatedly setting the flag "treat HL as IX" (or IY) and taking up 4 T states (But try to let MONS disassemble such a block.).

From Joseph S. Myers [jsm28@hermes.cam.ac.uk]:

[DD and FD have no effect on ED instructions, or on EX DE,HL. If an instruction references both (HL), and one of H and L, then (HL) becomes (IX+d) or (IY+d), but H or L stays as it is. For example, DD6601 is LD H,(IX+01).

DDCB and FDCB prefix instructions in which the next byte is a displacement byte, and the byte after that is the instruction byte. If the CB instruction references (HL), the effect is as expected; otherwise, the instruction seems to act on (IX+d) or (IY+d), and, if the instruction is such as to change the value acted on (any instruction other than BIT), the new value is loaded into the register the CB instruction would act on. Thus:

CB07 is RLC A; DDCB0107 is effectively RLC (IX+01) followed by LD A,(IX+01).]

c) ED opcodes:

There are a number of unofficial ED instructions, but none of them are very useful. The ED opcodes in the range 00-3F and 80-FF (except for the block instructions of course) do nothing at all but taking up 8 T states and incrementing the R register by 2. Most of the unlisted opcodes in the range 40-7F do have an effect, however. The complete list:

(* = not official, i.e. not listed in Zilog documentation)

(++ = documented in Spectrum manual)

ED40	IN B,(C)	ED60	IN H,(C)
ED41	OUT (C),B	ED61	OUT (C),H
ED42	SBC HL,BC	ED62	SBC HL,HL
ED43	LD (nn),BC	ED63 *	LD (nn),HL ++
ED44	NEG	ED64 *	NEG
ED45	RETN	ED65 *	RET
ED46	IM 0	ED66 *	IM 0
ED47	LD I,A	ED67	RRD
ED48	IN C,(C)	ED68	IN L,(C)
ED49	OUT (C),C	ED69	OUT (C),L
ED4A	ADC HL,BC	ED6A	ADC HL,HL
ED4B	LD BC,(nn)	ED6B *	LD HL,(nn) ++
ED4C *	NEG	ED6C *	NEG
ED4D	RETI	ED6D *	RET
ED4E *	IM 0/1	ED6E *	IM 0/1
ED4F	LD R,A	ED6F	RLD
ED50	IN D,(C)	ED70 *	IN (C) ++ as IN F,(C)
ED51	OUT (C),D	ED71 *	OUT (C),0
ED52	SBC HL,DE	ED72	SBC HL,SP
ED53	LD (nn),DE	ED73	LD (nn),SP

ED54 * NEG	ED74 * NEG
ED55 * RET	ED75 * RET
ED56 IM 1	ED76 * IM 1
ED57 LD A,I	ED77 * NOP
ED58 IN E,(C)	ED78 IN A,(C)
ED59 OUT (C),E	ED79 OUT (C),A
ED5A ADC HL,DE	ED7A ADC HL,SP
ED5B LD DE,(nn)	ED7B LD SP,(nn)
ED5C * NEG	ED7C * NEG
ED5D * RET	ED7D * RET
ED5E IM 2	ED7E * IM 2
ED5F LD A,R	ED7F * NOP

The ED70 instruction reads from port (C), just like the other instructions, but throws away the result. It does change the flags in the same way as the other IN instructions, however. The ED70 instruction *is* documented in at least two other publications beside the original Spectrum manual - one of which is an SGS Z80 data booklet.

The ED71 instruction OUTs a byte zero to port (C). In fact the instructions INI and OUTI do almost exactly the same as the proposed action for ED70 and ED71 as well as incrementing the registers. The IM 0/1 instruction puts the processor in either IM 0 or 1 (undetermined at this time). The SGS booklet says that there are two flip-flops which determine the interrupt mode and are set by the IM instructions. The possible values are:

00	IM 0
01	not used
10	IM 1
11	IM 2

...so it is entirely possible that the IM 0/1 instruction sets these registers to 01. What the Z80 does when this happens is anyone's guess.

d) About the R register:

This is not really an undocumented feature, although I have never seen any thorough description of it anywhere. The R register is a counter that is updated every instruction, where DD, FD, ED and CB are to be regarded as separate instructions. So shifted instruction will increase R by two. There's an interesting exception: doubly-shifted opcodes, the DDCB and FDCB ones, increase R by two too. LDI increases R by two, LDIR increases it by 2 times BC, as does LDDR etcetera. The sequence LD R,A/LD A,R increases A by two, except for the highest bit: this bit of the R register is never changed. This is because in the old days everyone used 16 Kbit chips. Inside the chip the bits were grouped in a 128x128 matrix, needing a 7 bit refresh cycle. Therefore ZiLOG decided to count only the lowest 7 bits. You can easily check that the R register is really crucial to memory refresh. Assemble this program:

```

ORG 32768
DI
LD B,0
L1: XOR A
LD R,A
DEC HL
LD A,H
OR L
JR NZ,L1
DJNZ L1
EI
RET

```

It will take about three minutes to run. Look at the upper 32K of memory, for instance the UDG graphics. It will have faded. Only the first few bytes of each 256 byte block will still contain zeros, because they were refreshed during the execution of the loop. The ULA took care of the refreshing of the lower 16K (This example won't work on the emulator, of course!).

e) Undocumented flags:

This undocumented "feature" of Z80 has its effect on programs like Sabre Wulf, Ghosts'n Goblins and Speedlock. Bits 3 and 5 of the F register are not used. They can contain information, as you can readily figure out by PUSHing AF onto the stack and then POPping some it into another pair of registers. Furthermore, sometimes their values change. I found the following empirical rule:

The values of bits 7, 5 and 3 follow the values of the corresponding bits of the last 8 bit result of an instruction that changed the usual flags.

For instance, after an ADD A,B those bits will be identical to the bits of the A register (Bit 7 of F is the sign flag, and fits the rule exactly). An exception is the CP x instruction (x=register, (HL) or direct argument). In this case the bits are copied from the argument.

If the instruction is one that operates on a 16 bit word, the 8 bits of the rule are the highest 8 bits of the 16 bit result - that was to be expected since the S flag is extracted from bit 15.

Ghosts'n Goblins use the undocumented flag due to a programming error. The rhino in Sabre Wulf walks backward or keeps running in little circles in a corner, if the (in this case undocumented) behaviour of the sign flag in the BIT instruction isn't right. I quote:

```

AD86 DD CB 06 7E BIT 7,(IX+6)
AD8A F2 8F AD JP P,#AD8F

```

An amazing piece of code! Speedlock does so many weird things that all must be exactly right for it to run. Finally, the '128 ROM uses the AF register to hold the return address of a subroutine for a while.

f) Interrupt flip-flops IFF1 and IFF2:

There seems to be a little confusion about these. These flip flops are simultaneously set or reset by the EI and DI instructions. IFF1 determines whether interrupts are allowed, but its value cannot be read. The value of IFF2 is copied to the P/V flag by LD A,I and LD A,R. When an NMI occurs, IFF1 is reset, thereby disallowing further (maskable) interrupts, but IFF2 is left unchanged. This enables the NMI service routine to check whether the interrupted program had enabled or disabled maskable interrupts. So, Spectrum snapshot software can only read IFF2, but most emulators will emulate both, and then the one that matters most is IFF1.

HARDWARE

At the hardware level, the Spectrum is a very simple machine. There's the 16K ROM which occupies the lowest part of the address space, and 48K of RAM which fills up the rest. An ULA which reads the lowest 6912 bytes of RAM to display the screen, and contains the logic for just one I/O port completes the machine, from a software point of view at least. Every even I/O address will address the ULA, but to avoid problems with other I/O devices only port FE should be used. If this port is written to, bits have the following meaning:

Bit	7	6	5	4	3	2	1	0
	+-----+							
				E M	Border			
	+-----+							

The lowest three bits specify the border color; a zero in bit 3 activates the MIC output, and a one in bit 4 activates the EAR output (which sounds the internal speaker). The real Spectrum also activates the MIC when the ear is written to. The upper three bits are unused. If port FE is read from, the highest eight address lines are important too. A

zero on one of these lines selects a particular half-row of five keys:

IN: Reads keys (bit 0 to bit 4 inclusive)

#FEFE	SHIFT, Z, X, C, V	#EFFE	0, 9, 8, 7, 6
#FD FE	A, S, D, F, G	#DF FE	P, O, I, U, Y
#FB FE	Q, W, E, R, T	#BF FE	ENTER, L, K, J, H
#F7 FE	1, 2, 3, 4, 5	#7F FE	SPACE, SYM SHFT, M, N,

A zero in one of the five lowest bits means that the corresponding key is pressed. If more than one address line is made low, the result is the logical AND of all single inputs, so a zero in a bit means that at least one of the appropriate keys is pressed. For example, only if each of the five lowest bits of the result from reading from port 00FE (for instance by XOR A/IN A,(FE)) is one, no key is pressed. A final remark about the keyboard. It is connected in a matrix-like fashion, with 8 rows of 5 columns, as is obvious from the above remarks. Any two keys pressed simultaneously can be uniquely decoded by reading from the IN ports. However, if more than two keys are pressed decoding may not be uniquely possible. For instance, if you press Caps shift, B and V, the Spectrum will think also the Space key is pressed, and react by giving the "Break into Program" report. Without this matrix behaviour Zynaps, for instance, won't pause

when you press 5,6,7,8 and 0 simultaneously.

Bit 5 (value 64) of IN-port FE is the ear input bit. When the line is silent, its value is zero, except in the early Model 2 of the Spectrum, where it was one. When there is a signal, this bit toggles. The Spectrum loading software is not sensitive to the polarity of this bit (which it definitely should not be, not only because of this model difference, but also because you cannot be sure the tape recorder doesn't change the polarity of the signal recorded!). Some old programs rely on the fact that bit 5 is always one (for instance Spinads).

Bits 6 and 7 are always one. The ULA with the lower 16K of RAM, and the processor with the upper 32K RAM and 16K ROM are working independently of each other. The data and address buses of the Z80 and the ULA are connected by small resistors; normally, these do effectively decouple the buses. However, if the Z80 wants to read or write the lower 16K, the ULA halts the processor if it is busy reading, and after it's finished lets the processor access lower memory through the resistors. A very fast, cheap and neat design indeed!

From Rui Ribeiro:

[If you read from a port that activates both the keyboard and a joystick port (e.g. Kempston), the joystick takes priority. Every emulator except SpecEm and WSpecEm get this wrong - see Street Hawk and Command4.]

If you run a program in the lower 16K of RAM, or read or write in that memory, the processor is halted sometimes. This part of memory is therefore somewhat slower than the upper 32K block. This is also the reason that you cannot write a sound- or save-routine in lower memory; the timing won't be exact, and the music will sound harsh. Also, INning from port FE will halt the processor, because the ULA has to supply the result. Therefore, INning from port FE is a tiny bit slower on average than INning from other ports; whilst normally an IN A,(nn) instruction would take 11 T states, it takes 12.15 T states on average if nn=FE. See below for more exact information.

If the processor reads from a non-existing IN port, for instance FF, the ULA won't stop, but nothing will put anything on the data bus. Therefore, you'll read a mixture of FF's (idle bus), and screen and ATTR data bytes (the latter being very scarce, by the way). This will only happen when the ULA is reading the screen memory, about 60% of the 1/50th second time slice in which a frame is generated. The other 40% the ULA is building the border or generating a vertical retrace. This behaviour is actually used in some programs, for instance, in Arkanoid.

Finally, there is an interesting bug in the ULA which also has to do with this split bus. After each instruction fetch cycle of the processor, the processor puts the I-R register "pair" (not the 8 bit internal Instruction Register, but the Interrupt and R registers) on the address bus. The lowest 7 bits, the R register, are used for memory refresh. However, the ULA gets confused if I is in the range 64-127, because it thinks the processor wants to read from lower 16K ram very, very often. The ULA can't cope with this read-frequency, and regularly misses a screen byte. Instead of the actual byte, the byte previously read is used to build up the video signal. The screen seems to be filled with 'snow'; however, the Spectrum won't crash, and program will continue to run normally. There's one program I know of that uses this to generate a nice effect: Vectron (which has very nice music too, by the way).

The processor has three interrupt modes, selected by the instructions IM 0, IM 1 and IM 2. In mode 1, the processor simply executes an RST #38 instruction if an interrupt is requested. This is the mode the Spectrum is normally in.

The other mode that is commonly used is IM 2. If an interrupt is requested, register (as the high byte) with whatever the interrupting device places on the data bus. The subroutine at this address is then called. Rodney Zaks in his book "Programming the Z80" states that only even bytes are allowed as low index byte, but that isn't true. The normal Spectrum contains no hardware to place a byte on the bus, and the bus will therefore always read FF (because the ULA also doesn't read the screen if it generates an interrupt), so the resulting index address is $256*I+255$. However, some not-so-neat hardware devices put things on the data bus when they shouldn't, so later programs didn't assume the low index byte was FF. These programs contain a 257 byte table of equal bytes starting at $256*I$, and the interrupt routine is placed at an address that is a multiple of 257. A useful but not so much used trick is to make the table contain FF's (or use the ROM for this) and put a byte 18 hex, the opcode for JR, at FFFF. The first byte of the ROM is a DI, F3 hex, so the JR will jump to FFF4, where a long JP to the actual interrupt routine is put.

In interrupt mode 0, the processor executes the instruction that the interrupting device places on the data bus. On a standard Spectrum this will be the byte FF, coincidentally (...) the opcode for RST #38. But for the same reasons as above, this is not really reliable.

The 50 Hz interrupt is synchronized with the video signal generation by the ULA; both the interrupt and the video signal are generated by it. Many programs use the interrupt to synchronize with the frame cycle. Some use it to generate fantastic effects, such as full-screen characters, full-screen horizon (Aquaplane) or pixel colour (Uridium for instance). Very many modern programs use the fact that the screen is "written" (or "fired") to the CRT in a finite time to do as much time-consuming screen calculations as possible without causing character flickering: although the ULA has started displaying the screen for this frame already, the electron beam will for a moment not "pass" this or that part of the screen so it's safe to change something there. So the exact time in the 1/50 second time-slice at which the screen is updated is very important. Each line takes exactly 224 T states.

After an interrupt occurs, 64 line times pass before the byte 16384 is displayed. At least the last 48 of these are actual border-lines. I could not determine whether my monitor didn't display the others or whether it was in vertical retrace, but luckily that's not really important.

From Ian Collier:

[This is probably not quite correct. I have some exact figures, but so far I have failed to bring them into the lab in order to program them into xz80. The 48K constants in there at the moment are wrong, but the +3 ones are correct. On the +3 there are exactly 14368 (well, give or take 1) cycles between the interrupt and byte 16384 being displayed, which turns out to be 63 lines and 4 cycles. On the 48K it's 14339 cycles. *I think!*

Then the 192 screen+border lines are displayed, followed by about 56 border lines again. 56.5

border lines would make up exactly 70000 T states, 1/50th of 3500000. However, I noticed that the frequency of the 50 Hz interrupt (measured in 1/T states!) changes very slightly when my Spectrum gets hot (I think it has something to do with the relative change of the frequencies of the two crystals in the Spectrum), so the time between interrupts will probably not be exactly 70000 T states.

From Ian Collier:

[This can't be true. A line contains exactly 224 cycles, and the screen contains exactly 312 lines (on the +3 and other 128K machines, 311 lines of 228 T-states). If the screen didn't contain a whole number of lines then the TV would not be able to display it correctly. This means that there are exactly 69888 cycles in a frame. Unfortunately I have never determined whether this means that the 50Hz clock is too quick or the 3.5MHz clock is too slow (or neither, which would prove me wrong - note, though, that I have written a program which depends on the correct number of cycles per frame).]

Anyway, whether the final border block is of fixed or variable length doesn't concern us either, the timings of the start and end of the screen, which are the timings of real interest, are fixed. Now for the timings of each line itself. I define a screen line to start with 256 screen pixels, then border, then horizontal retrace, and then border again.

All this takes 224 T states. Every half T state a pixel is written to the CRT, so if the ULA is reading bytes it does so each 4 T states (and then it reads two: a screen and an ATTR byte). The border is 48 pixels wide at each side. A video screen line is therefore timed as follows: 128 T states of screen, 24 T states of right border, 48 T states of horizontal retrace and 24 T states of left border.

When an interrupt occurs, the running instruction has to be completed first. So the start of the interrupt is fixed relatively to the start of the frame up to the length of the last instruction in T states. If the processor was executing a HALT (which, according to the Z80 books I read, is effectively many NOPs), the interrupt routine starts at most 3 T states away from the start of the frame. Of course the processor also needs some T states to store the program counter on the stack, read the interrupt vector and jump to the routine. In interrupt modes 0, 0/1 and 1 the total time taken to get to address 56 is 13 cycles. In interrupt mode 2 the time taken to get to the interrupt routine is 19 cycles. Interpreted as follows:

- 5 cycles: a read of the data bus (in IM 1 this is thrown away)
- 6 cycles: a read of the 2-byte interrupt vector IM 2 only)
- 8 cycles: to push the current PC and set the new one.

When an NMI occurs, the total time taken to get to address 102 is 15 cycles. This is the same as for an ordinary interrupt but with two extra cycles (reason unknown).

Now when to OUT to the border to change it at the place you want? First of all, you cannot change the border within a "byte", an 8-pixel chunk. If we forget about the screen for a moment, if you OUT to port FE after 14326 to 14329 T states (including the OUT) from the start of the IM 2 interrupt routine, the border will change at exactly the position of byte 16384 of the screen. The other positions can be computed by remembering that 8 pixels take 4 T states, and a line takes

224 T states. You would think that OUTing after 14322 to 14325 T states, the border would change at 8 pixels left of the upper left corner of the screen. This is right for 14322, 14323 and 14324 T states, but if you wait 14325 T states the ULA happens to be reading byte 16384 (or 22528, or both) and will halt the processor for a while, thereby making you miss the 8 pixels. This exception happens again after 224 T states, and again after 448, and so forth. These 192 exceptions left of the actual screen rectangle are the only ones; similar things don't happen at the right edge because the ULA don't need to read things there - it has just finished! As noted above, reading or writing in low ram (or OUTing to the ULA) causes the ULA to halt the processor. When and how much? The processor is halted each time you want to access the ULA or low memory and the ULA is busy reading. Of the 312.5 'lines' the ULA generates, only 192 contain actual screen pixels, and the ULA will only read bytes during 128 of the 224 T states of each screen line. But if it does, the processor is halted for exactly 4 T states.

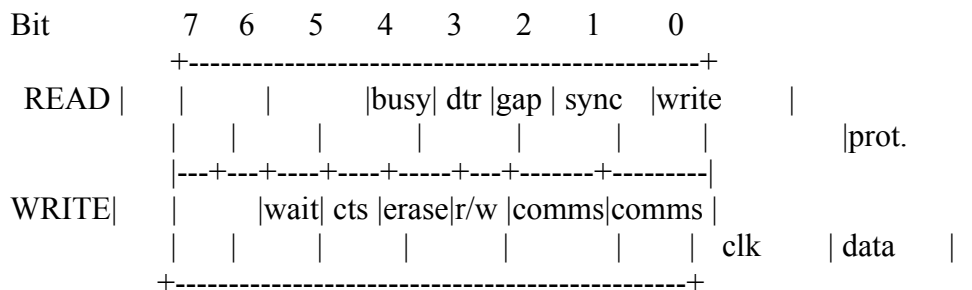
INTERFACE 1

The Interface I is quite complicated. It uses three different I/O ports, and contains logic to page and unpage an 8K ROM if new commands are used. The ROM is paged if the processor executes the instruction at ROM address 0008 or 1708 hexadecimal, the error and close# routines. It is inactivated when the Z80 executes the RET at address 0700.

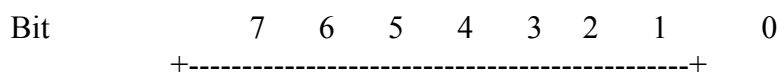
a) Port E7:

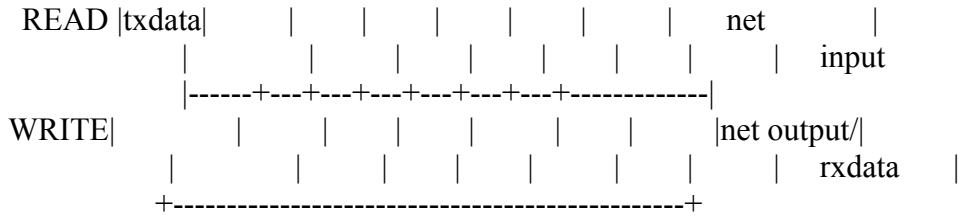
I/O port E7 is used to send or receive data to and from the microdrive. Accessing this port will halt the Z80 until the Interface I has collected 8 bits from the microdrive head; therefore, if the microdrive motor isn't running, or there is no formatted cartridge in the microdrive, the Spectrum hangs. This is the famous 'IN 0 crash'.

b) Port EF:



Bits DTR and CTS are used by the RS232 interface. The WAIT bit is used by the Network to synchronise, GAP, SYNC, WR_PROT, ERASE, R/_W, COMMS CLK and COMMS DATA are used by the microdrive system. If the microdrive is not being used, the COMMS DATA output selects the function of bit 0 of out-port F7:





TXDATA and RXDATA are the input and output of the RS232 port. COMMS DATA determines whether bit 0 of F7 is output for the RS232 or the network.

THE 128K SPECTRUMS

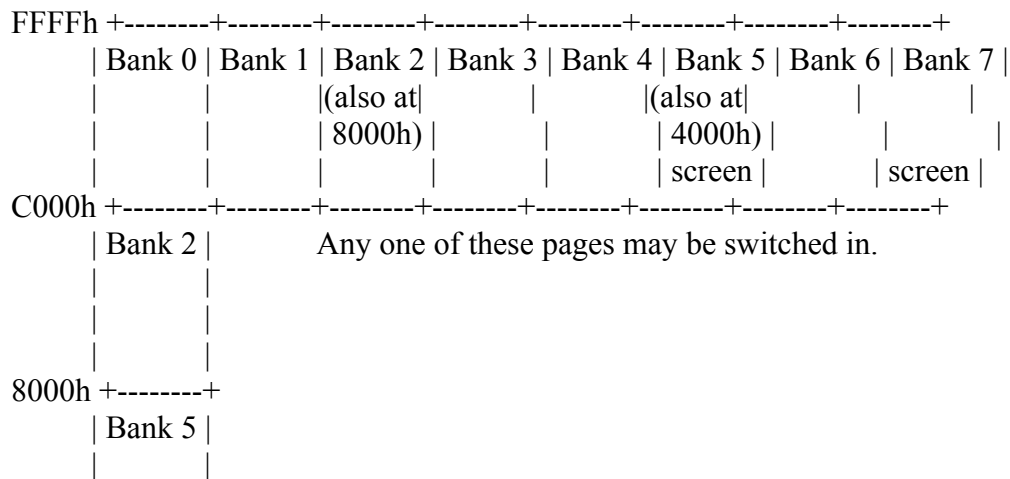
a) Memory

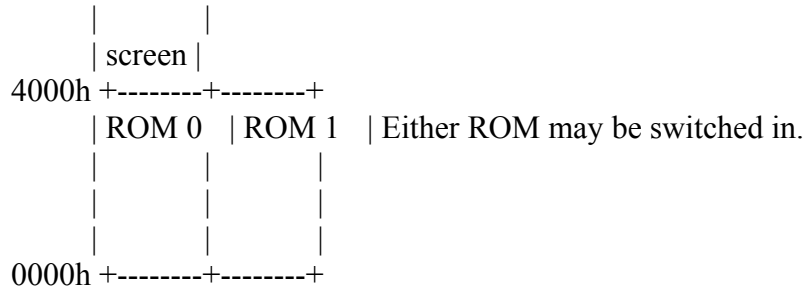
When memory is being paged, interrupts should be disabled and the stack should be in an area which is not going to change. If normal interrupt code is to run, then the system variable at 5B5Ch (23388) must be kept updated with the last value sent to port 7FFDh. It is not possible to read this port.

On the 128 and +2, memory is entirely controlled by port 7FFDh. The byte to output will be interpreted thus:

- Bits 0-2: RAM page (0-7) to map into memory at 0C000h
- Bit 3: Select normal (0) or shadow (1) screen. The normal screen is in bank 5 and normally appears at 4000h; the shadow screen is in bank 7 and can be switched in at 0C000h
- Bit 4: ROM select. ROM 0 is the 128k editor and menu system; ROM 1 contains 48k BASIC.
- Bit 5: If set, memory paging will be disabled and further output to this port will be ignored until the computer is reset.

The memory map of these computers is:





Memory banks 4-7 are contended, i.e. the processor shares them with the ULA. This reduces the speed of memory access in these banks.

b) Keypad

The 128K machine's keypad extra editing facilities are also available via the normal keyboard:

FUNCTION	KEYS
Beginning of next word	[E] [S] J
Beginning of previous word	[E] I
Up ten lines	[E] P
Down ten lines	[S] I
Start of line	[E] [S] 2
End of line	[E] M
First line	[E] N
Last line	[E] T
Screen	[E] [S] 8
Delete this character	[E] [S] K
Delete word left	[E] E
Delete word right	[E] W
Delete to start of line	[E] K
Delete to end of line	[E] J

[E] = Extended Mode

[S] = Symbol Shift

c) Sound Chip

The AY-3-8912 sound chip is a widely used one, to be found in the MSX, Vectrex, Amstrad CPC range, etc. It is controlled by two I/O ports:

- OUT (0FFFDh) - Select a register 0-14
- IN (0FFFDh) - Read the value of the selected register
- OUT (0BFFDH) - Write to the selected register

There's a guide to how to use the registers at

http://arachnid.cs.cf.ac.uk/User/K.E.W.Thacker/Amstrad/CPC_Guide/Hardware_Programming/Chips/psg.html

- this is for the CPC, so the I/O commands used will be different, and on the Spectrum register 14 is used for the serial ports rather than the keyboard as on the CPC.

PINOUTS

1. ULA pinout

He *multiplexed* address-lines.

/WR	2	39	Q	One of the +5V is decoupled through a RC-low-pass.
/RD	3	38	/MREQ	U,V are the color-difference signals.
/WE	4	37	A15	/Y is the inverted video including sync.
A0	5	36	A14	D are the data-lines, decoupled from the CPU by
A1	6	35	/RAS	resistors.
A2	7	34	/ROM CS	T are the data-lines to the keyboard (address-lines
A3	8	33	/IO-ULA	through diodes).
A4	9	32	CLOCK	SOUND is the analog-I/O-line for beep, save and load.
A5	10	31	D7	CLK is the clock-source to the CPU including the
A6	11	30	D6	inhibited T-states.
/INT	12	29	D5	IO-ULA is (A0(CPU) OR /IORQ) for the I/O-port FEh
+5V	13	28	SOUND	Q is the 14MHz-crystal, other side grounded through
+5V	14	27	D4	a capacitor
U	15	26	T4	
V	16	25	D3	
/Y	17	24	T3	
D0	18	23	T2	
T0	19	22	D2	
T1	20	21	D1	

2. AY-3-8912

SOUND C	1	28	D0	Vcc is +5V.
PORT	2	27	D1	SOUND A, B and C can be tied together.
Vcc	3	26	D2	CLOCK can be some MHz.
SOUND B	4	25	D3	
SOUND A	5	24	D4	
GND	6	23	D5	
PORT	7	22	D6	
PORT	8	21	D7	
PORT	9	20	BC1	
PORT	10	19	BC2	
PORT	11	18	BDIR	
PORT	12	17	A8	
PORT	13	16	RESET	
CLOCK	14	15	CLOCK	

3. Keyboard layout

OUTER SIDE A 15 14 8 13 12 9 10 11 INNER SIDE

D
0 B R E N C S P 0 A Q 1
1 S S L Z O 9 S W 2
2 M K X I 8 D E 3
3 N J C U 7 F R 4
4 B H V Y 6 G T 5

INNER SIDE

[BR] BREAK [EN] ENTER [CS] CAPS SHIFT [SS] SYMBOL SHIFT

In real the matrix connections are in one row on the top side of the membrane.

ACKNOWLEDGEMENTS

Thanks to:

- Richard Carlsson - for info on his Speccylator emulator for the Amiga.
- Ian Collier - xz80 emulator, advice, Z80 information and more...
- Krzysztof Czysciak - information about Elwro emulator.
- John Elliott - information about 128K machines.
- Marat Fayzullin - for creating FAQ versions before 2.7!
- Arnt Gulbrandsen - for keeping Spectrum WWW page, FTP archive and multiple contributions to this file. Also, for his IBM PC-based JPP emulator. Your WWW page *is* cool, Arnt. ;)
- Simos Hadjiyiannis - for several FTP addresses I didn't have.
- Martijn van der Heide - for his database program and Jeroen Kwast's new email address.
- Des Herriott - for his xzx emulator, which is the first and the only way to play LodeRunner on a Unix workstation! :) (ignoring the unix version of course - eh Ian?)
- A.G.Jackson - information about emulators for Acorn Archimedes.
- Bob Johnson - information about TS2068 and TK90X.
- Gerton Lunter - for the excellent information about ZX-Spectrum internals and Z80 emulator for IBM PC.
- Peter McGavin - for some useful information and (of course =:)) his excellent Spectrum emulator which actually *multitasks* under AmigaDOS and is system-friendlier than many other programs.
- Miguel Melo - information about several programmers.
- Alexander Mudretsov - information about the SP105 emulator.
- Joseph S. Myers - information on undocumented Z80 behaviour.
- Troels Norgaard - information about KGB.
- Emil Obermayr - for pinouts of ULA, AY-3-8912 and keyboard layout.
- Boudewijn Rempt - compilation of postings for "WHERE IS...?"
- Rui Ribeiro - for several contributions to "TECHNICAL INFORMATION", and WSpecem.
- Jon Ritman - information about himself!
- Adam Roach - maintainer of comp.emulators.misc FAQ; information about the Speccy BASIC emulator for the C64.
- Thorsten Roskowitz - information about Atari-based emulator.
- Marcelo Strasunas - information about TK90X and TK95.
- Stephen Smith - information about several programmers and his great games FAQ and database web pages.

... and everybody else who has contributed to the FAQ, manages FTP archives, writes Speccy emulators and generally helps to keep the Speccy alive!

Rui Fernando Ferreira Ribeiro



I sincerely hope you appreciated my piece of work. The Spectrum was my first computer, and to this days there are still some emotional ties around it. I just had to do this emulation, it was an idea I had since '87. I tried it around '89, but I have failed abysmally. Luckily, my notes survived, since I still had the grips with my Spectrum at the time. As such, my final year project and the lack of a Spectrum emulator for Windows were the perfect excuses to create WSpecEm. Now something about myself: I like reading, with topics ranging from technology, science to futurist and medieval fiction. I also prefer science-fiction films. Usually, I listen to techno, underground and modern-classic music.

The PC games I played the most were Civilisation, Masters of Magic and The Legacy. As for the Spectrum games, they were Sweevo's World and Gunfright.

Nowadays, I am working as an UNIX network administrator and would like to go back to the UK in a near future.

Curriculum vitae

Personal Details

Name: Rui Fernando Ferreira Ribeiro *Sex:* Male
Date of birth: 12 September 1970 *Nationality:* Portuguese
E-mail: rff-ribe@ipp.pt
WEB page: <http://www.idt-isep.ipp.pt/~rff-ribe>
Address: Av. Carvalha, 280 3D *Phone:* +351 (0)2 4804584
4420 Gondomar 00351 2 4804584 (from the UK)
Portugal

Education

Winter-Spring 96 University of West of England Bristol, UK
Erasmus student at B.Sc. (HONS) Computing for real time-systems

n Final year project - I enrolled as a volunteer in a student interchange program. I was given an EC grant and developed my final year project in the UK, "A Spectrum emulator for MS-WINDOWS". This project consisted of an extensive dissertation with the collected data presented in a CD, an analysis with OOA and OOD techniques, a computer emulation program written in C using the Windows 16 bit-API with the WinG library for producing graphics, another program written in DOS 8086 assembler, for reading its tapes and a final oral exam. I had a 75% grade in this project.

92 - 96 Instituto Superior Engenharia Porto, Portugal
B.Sc. Computing equivalent

n Last year incomplete.

85-88 Externato Perpetuo Socorro Porto, Portugal
College of further education

n EC Computer technician level 3 equivalence.

Work experience

Summer 94 Porto, Portugal
Freelance

n Built and sold computers, jointly with a colleague.

Winter 93 Microdigital Porto, Portugal
Consultant

n Maintained the payroll program, trained a substitute and monitored his progress.

88 - 92 Microdigital Porto, Portugal
Professional computer programmer

n Maintained and developed a payroll program in C under DOS; solved computer-related general problems and dealt with clients. Programmed in assembler, to build extra functions for the program library.

Computer Experience

C	(8 years)
Windows 16-Bit API	(2 years)
8086 Assembler	(9 years)
Real-mode 386 Assembler	(2 years)
Z80 Assembler	(10 years)
DOS	(10 years)
UNIX	(7 years)

Notions of FoxPro, Turbo Pascal, Basic.
Notions of network programming (Novell).
HTML notions and Internet experienced user.
Proficient WinWord and Excel user.
Windows'95 user.
More details available on request.

Languages

Portuguese as mother tongue.
Can communicate effectively in English.
Can understand and read Spanish.
Can read French and some Italian.

References

Provided on request.

Publications

"ZX Spectrum Emulator", in CD ZONE, PC ZONE Issue 43, October 1996.

Extracurricular Activities

Lived seven months in the UK.

Developed several programs at home.

Interests and Activities

Travelling, running, swimming, programming and computer security.

Hobbies

Listening to music, reading and computing.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright © 1989, 1991 Free Software Foundation, Inc.

675 Mass Ave, Cambridge, MA 02139, USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we

have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
 - a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
 - b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
 - c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there

is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
 - a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
 - b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
 - c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the

major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY OR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN

ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright © 19yy <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright © 19yy name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

