S
PE
CTA
TO
R

version 1.00

**** SPECTATOR: ZX SPECTRUM EMULATOR -- Official Version 1.00 ****

by Carlo Delhez, October 1992

CONTENTS

PROGRAM DESCRIPTION

Spectator is - stated very accurately - a Z80 Emulator in a 48k Sinclair ZX Spectrum environment, written for the Sinclair QL and its clones. The Z80 Emulator is able to execute programs written in the Z80 machine code language. The ZX Spectrum environment includes the original ZX Spectrum ROM plus ZX Interface 1 ROM (edition 2) and additionally caters for "hardware compatibility" such as keyboard reading, screen output control and redirecting of signals originally heading for devices such as tape, Microdrive and RS232. The Z80 Emulator and the ZX Spectrum environment are joined in a single block of 68000 machine code, which - as a whole - shall be referred to as Spectrum Emulator, and is named Spectator. Spectator supports multitasking and is designed to be compatible with all versions of the QL ROM, including Minerva. The policy behind Spectator is to create a highly compatible, flexible, multitasking, well documented, no-nonsense Spectrum emulator for the QL.

NOTE: Spectator contains the original ZX Spectrum and ZX Interface 1 ROM codes. These codes are copyright software. Therefore, Spectator may ONLY be used by you if you own a ZX Spectrum plus ZX Interface 1 YOURSELF. In that case, you are IN PRINCIPLE able to LEGALLY transport the ROM codes from your Spectrum to your QL. Any usage of Spectator by people NOT owning a ZX Spectrum and ZX Interface 1 is in conflict with copyright laws.

REGISTRATION

Spectator is distributed on a ShareWare basis. You are allowed to make copies for other QL users, provided you copy ALL the files and do NOT make changes in any one of these files.

If you like to use Spectator, please express your appreciation and support my efforts by becoming a REGISTERED user. You will then receive a nicely printed copy of this manual, you will be informed about new releases, and can update to a more recent version for a small fee to cover my costs. Registration also includes free technical support by the author and the possibility to obtain a custom-made version of Spectator to suit your specific hardware configuration.

To become a registered Spectator user, send a EuroCheque worth Hfl 50 (Dutch Guilders), plus a note stating your name and address to me (address at the bottom of this document). The most recent version of Spectator on a 3.5" DS/DD floppy (3M), a laser-printed copy of the manual and a disk filled with interesting Public Domain / ShareWare Spectrum programs ready-to-load will be dispatched you.

For cheques other than EuroCheques, please add Hfl 20 for my cashing costs. If you wish, you may also send cash (banknotes) instead of a cheque. In case you send banknotes in a currency other than Dutch Guilders, please use an acceptable rate of exchange and add the equivalent of Hfl 10 for my exchange costs.

COMPATIBILITY

Since the original ZX Spectrum ROM and ZX Interface 1 Shadow ROM (edition 2) in Z80 code are included, almost complete software compatibility is guaranteed. Almost, since only some VERY specific software can obviously NOT be emulated. Application programs which properly use the Hook codes of ZX Interface 1 will function just fine. Programs which use their own INs and OUTs for microdrive access, serial communication, tape reading or any other non-standard I/O activity will certainly fail to operate.
Apart from these exceptional applications, ANY program running on the ZX Spectrum should ALSO run on Spectator.

COMPARISON

Spectator has some advantages and disadvantages with respect to the ZX Spectrum. To name just a few advantages: safe housing in a well-stabilized computer, improved keyboard and screen quality, full 64k RAM available, reliable and fast data storage (both tape and microdrive), easy reset etc. Additionally, Spectator happily MultiTasks, so you can run several copies of Spectator at one time, e.g. one copy to draw beautiful fractals, another copy to play a game meanwhile (provided you have enough memory in your QL).

If you have standard 8 MHz 68008-based QL, a main disadvantage is SPEED! A common property of all emulators known to man is the general lack of speed. Since single Z80 instructions are being translated in at least five and sometimes as much as tens of 68000 instuctions, Spectator is bound to be slow. The Spectrum emulator cannot do better than about 8% of the original ZX Spectrum speed. I therefore recommend NOT to run Spectator on a 'normal' QL. You will only get irritated by the low speed!

Much better performance is obtained on a 16 MHz 68000-based QL, and the speed increases to about 35% (one third of original Spectrum speed). Although 35% may still seem terribly slow, this speed is sufficient for general editting purposes and for running most (machine code) games and utilities. In fact, the ZX Spectrum is quite a fast computer and many machine-code games use fixed delay-loops or HALTs for proper human operation. Taking out the delay-loops (or even better: making them processor speed independent) will make most games as fast and exciting as they were on the ZX Spectrum.

In this manual it is assumed you are running Spectator on a 16 MHz 68000-based QL. More details on the effective clock frequency are given later on.

STARTING SPECTATOR

For running Spectator, the following hardware is required:

* A QL (or clone) with at least 300k of FREE RAM memory to store the program and the required work space, preferably fitted with a 16 MHz 68000 processor (Gold Card).

The following software is optional, but HIGHLY recommended:

* The well known Window Manager (and Pointer Interface) environment to have maximum flexibility in MultiTasking (e.g. QPAC2).

* Toolkit-II by Tony Tebby, mainly for Job Management.

Compatible hard- or software is ofcourse allowed as well. Obviously, for reasons of copyright, the Window Manager and Toolkit are not included in the Spectator package.

If you do not use the Window Manager environment, many of the possibilities discussed below do not apply for you. You should use EW (or EXEC_W) instead of EX, but then you cannot use CTRL-C or change priorities. However, Spectator runs a little bit faster!

Before starting Spectator, make sure Toolkit-II and the Window Manager are present (or create a simple boot-program to do this). Then type:

```
EX FLP1_SPECTATOR_EXE
```

(or use an equivalent hotkey/button-command when you are using the QJUMP extended environment)

Spectator will perform the regular RST 0, first showing a completely black screen. Almost immediately, the border should go white while the central part stays black. After a while you should see the memory test come by (red bars vertically across the screen). Next, these disappear again, and some time later the screen should go completely white and the well-known Sinclair copyright message appears at the bottom of the main screen in black ink. Now, Spectator is ready for use.

Before typing anything, return to SuperBasic by pressing CTRL-C and type the command JOBS. You will see that Spectator consists of four jobs, viz.:

* Spectator : This is the main part of the emulator, performing the actual translation from Z80 to 68000 code.

* Spc_Scr : This job is concerned with printing of the Spectrum screen only.

* Spc_Key : Keyboard decoding and CTRL keypress interception are catered for by this job

* Spc_Vita : The Spirit of Spectator; as soon as this job is removed, Spectator kills itself (more about that later).

The first three jobs have default priorities of 127, 16 and 16, respectively. This setting seems optimal for a QL with a Gold Card. However, the values can be adapted to suit another computing/display ratio. The priority of the latter job should always be 0. Just for fun, try:

```
          SPJOB SPC_VITA,127
          JOBS
```

Besides the above four jobs, also a 50 Hz interrupt has been linked to keep track of the ZX Spectrum internal clock (Frames, 3 bytes at $5C78). Because of this interrupt, keep in mind NEVER to delete either the Spectator-, Spc_Scr- or the Spc_Key-job. The ONLY proper way to stop Spectator is

```
          RJOB SPC_VITA       ( or equivalent )
```

Depending on the current keyboard interpreter setting, Spectator will either kill itself immediately, closing all channels, releasing its work-space and also unlinking the 50 Hz interrupt, or do this at the very moment it is forced to the foreground e.g. by CTRL-C. Any other way of removing Spectator will eventually crash your QL!

In Pointer Interface terms, Z80 emulation (Spectator job) is completely "unlocked" as it is not connected to any window or keyboard queue. So, when Spectator is temporarily buried by another job, the Z80 happily keeps on running, unless it needs to wait for a keypress which it requests from the (suspended) Spc_Key job. Please note that the ZX Spectrum ROM prints to the Spectrum Display File while Spc_Scr caters for the conversion from Spectrum Display File to QL screen!

Spectator is REALLY multitasking and neither the reading of the keyboard nor the printing of the screen will interfere with other jobs (provided 'WMan' and 'Ptr_Gen' are loaded).

Having read all this, switch back to the Spectrum screen (CTRL-C) and start having fun!

COMMANDSTRING QUALIFIERS

Instead of starting Spectator as described above, it is also possible to add a commandstring with one or more qualifiers as to influence the initial settings of Spectator. A qualifier is a single-letter command, preceeded by any one of the characters '-', '+' or '/' (there is no effective difference between these three prefixes). The commandstring must be passed as follows:

```
EX SPECTATOR_EXE;<commandstring>        ( or similar )
```

with <commandstring> a character expression, normally a quoted string. The commandstring may contain one or more qualifiers, which may (but need not be) separated by one or more spaces. The Spectator commandstring interpreter is case-insensitive, so upper or lower case text may be used at will. Qualifiers are interpreted left to right, later settings superseding earlier settings.

Currently, the following qualifiers are supported:

/#	force keyrow mode
/I	force inverted screen colours (EOR 7)
/N	force sound via QL network
/S	force sound via QL beeper

Example:

```
EX SPECTATOR_EXE;'/# /N'        : start Spectator in keyrow mode
                                  and generate sound via network
```

Future versions of Spectator will support many more commandstring qualifiers in order to to set things like default drives, RS232 redirection etc. Any invalid qualifier in the commandstring will result in an error message being printed in #0 or #1 and execution of Spectator will be aborted without any QDOS error.

SCREEN OUTPUT

The screen output frequency is determined by the priority of Spc_Scr. The ZX Spectrum FLASH attribute is emulated by software, as the QL hardware flash works in a completely different fashion. Spectator fixes the flash frequency to about 2 Hz, but if the priority of Spc_Scr is too low, a frequency of less than this value is used (in that case you get one flash-flip per sceen update). It is advised not to increase the priority of Spc_Scr too much, as this slows down Z80 emulation considerably!
The ZX Spectrum BRIGHT attribute is neglected as the QL has only 8 colours. Only in rare cases, this causes information on the screen to be invisible.

KEYBOARD CONSIDERATIONS

Spectrum BASIC is tokenized. This is handy in the sense that you need less keystrokes for often used commands like 'PRINT': you merely have to press 'P' when the cursor is a 'K' and the whole keyword appears. A disadvantage is the fact there are no less than 91 tokens and these are all hidden under no more than 40 keys. These 40 keys also have to provide the regular ASCII set (96 characters), special control characters (25 in total) and a set of (user defined) graphics (37). This makes a a grand total of 249. So, each of the forty keys can be used for an average of 6.2 functions (some keys have only one function, many have 7, some 8 or even 9).

Obviously, it is quite impossible to know by heart where all these functions are located. The original Spectrum keyboard has the functions printed on the keyboard itself. The QL obviously does not provide this facility; I therefore have added a help-screen which is simply a copy of the Spectrum keyboard.

When you press CTRL-F1 you will hear a friendly beep and the help screen appears. Press ESC to return to the Spectrum screen.
It is not the purpose of this manual to explain how to use the Spectrum keyboard. Please consult a Spectrum handbook to learn this.

As said before, the 48k Spectrum keyboard has only 40 keys. All of these keys are also available on the QL:

* 26 letter keys (A thru Z)
* 10 number keys (0 thru 9)
* 1 Space key (Space bar on the QL)
* 1 Enter key (same on the QL)
* 1 Capital-Shift key (Shift on the QL)
* 1 Symbol-Shift key (CTRL on the QL)

It is important to note the difference between CAPITAL SHIFT (QL SHIFT key) and SYMBOL SHIFT (QL CTRL key)!

These 40 keys on the QL should be sufficient to operate Spectator. But there are some queer incompatibilities. Just some examples:

- To get an uppercase letter on the QL, one presses Shift plus the letter key (e.g. Shift-'p' gives 'P'); on the Spectrum one has to press Capital-Shift plus the letter key, i.e. just the same. BUT: to get the symbols of the numeric keys of the QL, one also needs to press Shift plus the number key (e.g. Shift-'4' gives '$'). However, on the Spectrum one has to press Symbol-Shift (NOT Capital-Shift!) plus the number key, so this means: you have to press CTRL-'4' on the QL to get the dollar. Pressing SHIFT-'4' on the QL (i.e. Capital-Shift '4') produces 'Inverse Video Mode'.

- Having read the above example, one would press CTRL-'8' to get an asterisk (the '*' symbol). Strangely enough, this keystroke produces the opening bracket '(': the symbols on the Spectrum keyboard are arranged in a strange non-standard fashion:

key:	1	2	3	4	5	6	7	8	9	0
CTRL-key:	!	@	#	$	%	&	'	(	)	_

  The asterisk is 'hidden' under CTRL-B and the '^' (hat) symbol can be typed as CTRL-H. Other symbols are distributed over the various letter keys:

key:	B	C	H	J	K	L	M	N	O	P	R	T	V	X	Z
CTRL-key:	*	?	^	-	+	=	.	,	;	"	<	>	/	Pound	:

  CTRL plus any one of the remaining letter keys produce a special token (e.g. CTRL-A gives 'STOP').

It is obvious that it is not quite easy to use the QL keyboard as a Spectrum typewriter. An additional problem is the fact that the Spectrum does not have a 'type ahead buffer' (keyboard queue). So, the keyboard is not scanned until a previous keypress has been fully processed. On the normal Spectrum this is not really a problem, unless you are editting very long lines of BASIC. On Spectator, this is a distinct problem as it operates 3 (or even 12) times slower than the Spectrum. Using the QL keyboard queue might be a solution, but then many games don't work since these allow you to press many keys at once for complicated steering mechanisms; these multiple keypresses are ignored by the QL keyboard queue handler. Additionally, pressing down a single key continuously cannot be 'seen' through the queue handler, as it starts to auto-repeat with a finite repetition frequency.

Spectator provides TWO keyboard emulation modes:

(1) Keyboard-queue-mode (indicated by a 'K' in the top-righthand corner)
(default when Spectator is started)

Advantages:
- Type-ahead buffer, so you don't have to wait until a keypress has been processed
- Excellent keyboard mapping: all ASCII symbols on the QL keyboard can be typed (just using Shift, not CTRL)
- All-time autorepeat (origin: QL keyboard handler)
- No unwanted keyboard input when job is buried
- Very well suited for serious applications

Disadvantages:
- Tokens less easily accessible
- Poor emulation of Spectrum 'keyrow' map
- No emulation of original Spectrum key repetition
- Delayed recognition of special CTRL keystrokes
- Not suited for playing games

(2) Keyrow-mode (indicated by a '#' in the top-righthand corner)

Advantages:
- Excellent emulation of Spectrum keyrow map
- Original Spectrum keyrepeat emulated correctly
- Tokens more easily accessible
- Immediate recognition of special CTRL keystrokes
- Very well suited for playing games

Disadvantages:
- No keyboard queue, dull typing
- ASCII symbols not easily accessible
- Only auto-repeats when the Spectrum ROM code is used
- May 'see' unwanted keystrokes while job is buried
- Not suited for serious applications

You can switch from one mode to another by typing CTRL-F3. You should hear a friedly beep and the symbol 'K' changes into '#' or vice versa.

Ad (1) : KEYBOARD-QUEUE-MODE

All the letters and symbols can be typed on the QL keyboard as though you were using a QL editor. You don't have to use CTRL: both capitals and symbols on the number keys can be obtained by pressing the Shifted key.

Dedicated symbol keys (like '.', '/' and ';') can be used as on the QL (also Shifted). There are a few exceptions: the keys '[', ']', '{', '}', '\', '|', '~' and '(copyright)' can only typed when the Spectrum is in extended mode (E-cursor). First put the Spectrum in extended mode by pressing F5 and then press the appropriate key(s) for the symbol you need.

Specials: arrows work just fine

Shift-Space is break

CTRL-left produces 'delete' (equivalent to CTRL-0)

CTRL-Capslock switches Spectrum CAPSLOCK mode but does NOT alter QL CAPSLOCK (equivalent to CTRL-2); take care NOT to switch on QL CAPSLOCK as this will disable a large number of the keyboard functions

F4 produces 'edit' (equivalent to CTRL-1)

F5 flips extended mode (equivalent to pressing CTRL and SHIFT together, but that is not seen through the queue handler!). You can also use Shift-Enter.

Further examples showing how to obtain tokens and special functions:
(please take a look at the help screen for more information!)

cursor	keypress	result	notes
K	Q	PLOT	
K	Shift-Q	PLOT	
K/L	CTRL-Q	<=	CTRL-key works ONLY for tokens
L	Q	q	
L	Shift-Q	Q	
E	Q	SIN	
E	Shift-Q	ASN	don't use CTRL-Q
G	Q	udg-Q	
G	Shift-Q	udg-Q	udg = user definable graphic
K/L	1	1	
K/L	Shift-1	!	
K/L	CTRL-1	edit	
E	1	blue paper	
E	CTRL-1	blue ink	
E	Shift-1	DEF FN	
G	1	graphic	
G	CTRL-1	inverse graphic	
G	Shift-1	inverse graphic	

The keywords of the numeric keys '6' thru '0' need some further explanation. Similarly to Shift-'1' in extended mode for 'DEF FN', you are tempted to press Shift-'9' in extended mode for 'CAT'; this however is not right: Shift-'9' is converted to SymbolShift-'8' for the sake of the opening bracket '(', so you get 'POINT'. To get 'CAT', press Shift-'0' in extended mode, as this is converted to SymbolShift-'9'. It is good to remember that the RED symbols on each Spectrum key are also on the SAME QL key. You therefore have to press the single quote key in extended mde to get the 'ERASE' command, while the underscore key ('_') is used for 'FORMAT'.

Finally a WARNING: ALL keypresses are seen and queued. If you're the type of person that frantically hits all the keys of the keyboard when a program doesn't react quickly enough to your commands, keep in mind that ALL these useless keypresses are remembered and executed, sometimes with disasterous results...

Ad (2) : KEYROW-MODE

All the letters and symbols must be typed as though you were using a Spectrum keyboard. Use Shifted letter keys for uppercase, but press CTRL plus a numeric key to get the symbol on that key.

Dedicated symbol keys (like '.', '/' and ';') can be used as on the QL, but you cannot Shift them (for example: Shift-'/' still produces '/', not '?') There is no special provision for extended-mode symbols.

Specials: arrows work just fine
Shift-Space is break
CTRL-left produces 'delete' (equivalent to Shift-0)
Capslock switches Spectrum CAPSLOCK mode but ALSO QL CAPSLOCK mode (it is better to use Shift-2)
F4 produces 'edit' (equivalent to Shift-1)
F5 flips extended mode (equivalent to pressing CTRL and SHIFT together). You can also use Shift-Enter.

Further examples showing how to obtain tokens and special functions:
(please take a look at the help screen for more information!)

cursor	keypress	result	notes
K	Q	PLOT	
K	Shift-Q	PLOT	
K/L	CTRL-Q	<=	CTRL-key works also for symbols
L	Q	q	
L	Shift-Q	Q	
E	Q	SIN	
E	Shift-Q	ASN	don't use CTRL-Q
G	Q	udg-Q	
G	Shift-Q	udg-Q	udg = user definable graphic
K/L	1	1	
K/L	Shift-1	edit	\
K/L	CTRL-1	!	| Note the differences with
E	1	blue paper	> respect to the Keyboard-
E	CTRL-1	DEF FN	| Queue-Mode!
E	Shift-1	blue ink	/
G	1	graphic	
G	CTRL-1	inverse graphic	
G	Shift-1	inverse graphic	

To get the '?' symbol you would need to press CTRL-C. This is not possible as this keypress is intercepted by the job manager. The backslash key '\' produces a question mark without switching jobs. For the backslash itself, press Shift-'D' in extended mode.
Using CTRL-C for switching jobs MAY be seen by Spectator and causes a repetition of '?' characters in the period that Spectator remains buried; this cannot easily be avoided.

None of the keywords or functions are converted to deviating key presses, so you can directly use all the keys shown on the help screen.

There is no queue, so type slowly and firmly. It may be good practice to adapt job priorities or to change the Spectrum repetition system variables for better response (REPDEL and REPPER are at $5C09 and $5C0A respectively).

KEYBOARD CONTROL

There are several special Control keypresses. These are not available on the normal Spectrum but may be quite useful for the emulator. They are summed up below:

F1	Pop up Spectator main menu; see section 'Pop-Up Menu System'
CTRL-F1	Show keyboard Help screen; return with ESC
CTRL-F2	Negative screen colours (EOR 7); press CTRL-F2 again to return to the normal colours
CTRL-F3	Toggle keyboard mode (Keyboard-Queue-Mode is indicated by a 'K', Keyrow-Mode by a '#', see previous section)
CTRL-ALT-F3	Toggle sound; sound off is indicated by 's' (lower case), sound via QL beeper on is indicated by 'S' (upper case) and sound via network port by 'N'. See section 'Beeping' for more details.
CTRL-ESC	Refresh Spectrum ROM and force a Break in the current program (also in machine code); this may crash your Spectrum (in case of a crash: press CTRL-ALT-ESC to recover).
CTRL-ALT-ESC	Refresh Spectrum ROM and perform a RST 0 (Z80 reset) (the safest way to stop protected games or to recover from a crash; works like the Non-Maskable Interrupt on a Spectrum). This also sets the keyboard mode to 'K' (queue mode) as this mode seems most suited for Spectrum BASIC.

In Keyboard Queue Mode, a CTRL-keypress is stored in the queue and is not seen until the Spectrum ROM or a Spectrum program reads the keyboard. This may cause a delayed recognition of the keypress. In case of a crash, the keypress may not be seen at all. Remember that pressing the key more than once does not cure this!
In Keyrow Mode, the CTRL-keypresses are always seen immediately, whatever the Spectrum is doing at that moment. You will hear a friendly beep as soon

as the keypress has been recognized. Pressing CTRL-ESC in the Spectrum editor won't do any harm, but the effect in other editors is unpredictable.

POP-UP MENU SYSTEM

To make life for Spectator users even easier, a pop-up menu system has been implemented. The system of menus offers many facilities by pressing down just a few keys. The menu-system was first introduced in the pre-release of Spectator 1.00 and will be further expanded in future versions.

The main menu can be popped up by pressing F1 while the emulator is running. A list with the available options is shown. In order to execute any one of the options, simply press the key printed in front of the desired option. Depending on the chosen option, either new menus are popped up or the required action is performed immediately.

The main menu already offers the following facilities:

```
D - make screendump of Spectrum screen as QL MODE-8 file
    (filename is requested; overwrite confirmation and full
    error trapping);
H - pop up the keyboard help screen (return by ESC);
I - show general program info (return by ESC);
K - present overview of main keyboard functions (return by ESC);
Q - quit Spectator (confirmation requested);
R - reset Z80 (hence Spectrum);
S - save memory contents as compressed Z80 snapshot (filename
    is requested; overwrite confirmation & full error trapping).
```

In general, the menus disappear by pressing ESC. Only when a text needs to be editted (e.g. a filename), ESC won't work; in that case, press the up or down cursor movement key instead.

BUTTONING

If you want to put Spectator in a button, it is advantageous to call up the keyboard help screen or the menu system first (using CTRL-F1 or F1). This has the effect that Spectator is almost entirely suspended until you press ESC. Buttoning Spectator this way results in a significant reduction of CPU-time consumption while Spectator is in the buttoned state and additionally the program you were running (e.g. a game) is halted very effectively. After unbuttoning, press ESC to return to the main Spectrum screen and the program is properly resumed.

BEEPING

By default, Spectator produces no sound (lower case 's' in the top-righthand corner of the screen). Sound can be generated in two ways: via the QL's internal beeper or via the QL's net port. Pressing CTRL-ALT-F3 will change the lower case 's' into an upper case 'S', indicating sound generation via the QL beeper. Pressing CTRL-ALT-F3 once more, will display the letter 'N' to indicate sound generation via the net port. Press CTRL-ALT-F3 once more to disable sound again.

Although the generated sounds may sometimes be a real pain in the ear, it can also be quite advantageous to enable sound. Some games seem to have crashed on a mute QL, while they are actually 'amusing' the player with some lengthy melody before commencing. In such a case, 'sound on' can be helpful for determining what the program is actually doing.

(1) 'S' ... sound via QL beeper

The Spectrum can produce sound by regularly activating and deactivating the electromagnet in the loudspeaker. On the QL, sound is produced in a similar way, but it is fully controlled by the 8049 coprocessor and therefore unaccessible directly from 68000 machine code. Nevertheless, Spectator does try to mimic the Spectrum sound. For this, it registers the output heading for the Spectrum speaker during a period of 20 ms, and then sends a command to the IPC to generate a tone which resembles the tone originally produced by the Spectrum. As this involves an averaging procedure, quick variations occuring within the 20 ms period cannot be noticed and are lost in the produced sound.

(2) 'N' ... sound via Net port

Much better sound can be generated via the QL net port. The way this port is accessed by the QL is very similar to the way the Spectrum accesses its internal loudspeaker. Therefore, the output of the net port will sound just the same as on the Spectrum, without loss of quality.
There's just one draw-back: Spectator can only control the net port from the Z80-emulation job ('Spectator' job). Although this job has a very high priority, the scheduler will need to switch to other jobs (e.g. 'Spc_Scr') once every while. When this happens, the sound stops abruptly and is only resumed when 'Spectator' gets CPU access again from the scheduler.
Due to these silent periods, however short they may be, the output sounds stammering and faltering. You can hear the real quality of net-sound when you press CTRL-F5 (Scroll Lock): this will suspend any screen and keyboard activities and the sound is no longer interrupted!
Obviously, frequencies are not reproduced but lessened proportionally to the operating speed of Spectator with respect to the real Spectrum. It is my experience that this has little influence on the quality of the sound.

In order to hear sound via the net port, you must connect it to your audio set (amplifier). There are several ways you can do this:

(a) Connect a small loudspeaker (or set of headphones) directly to the net port. This is a simple but somewhat blunt solution. It may blow-up your net circuit due to the large currents flowing.

(b) Connect the net port to the microphone input of your amplifier directly. This should do no harm to your QL net circuit.

(c) The same as (b), but with an additional low-pass or middle-pass RC-filter inbetween QL and amplifier in order to smoothen the sharp edges of the pulses produced in the net circuit.

In case you didn't know it already: the net ports are on the outer left when facing the back of your QL; they are marked 'NET' (how appropriate!). It doesn't matter which of both ports you use.

INTERRUPTS

Interrupts are partially emulated by Spectator. The ZX Spectrum usually operates in Interrupt Mode 1 (IM 1). The Maskable Interrupt Server at address $0038 is called 50 times per second to update the internal clock (frames) and to read the keyboard. When Spectator would have to do this in the same way, execution speed would go down rather drastically, so IM 1 is neglected. Instead, Spectator uses a QL interrupt to update the clock at 16.7 Hz (1/3rd of 50 Hz) and a separate job to read the keyboard ('Spc_Key').

Many programs enable Interrupt Mode 2 (IM 2) which allows for user interrupt routines. In this mode, the interrupt address is fetched from the address whose high part is formed by the I register and whose low part is read from the data line (always $FF in Spectator). Spectator DOES support this type of interrupt, but not at a rate of 50 Hz as this would also slow down the emulator too much. Spectator tries to generate Z80 interrupts at a frequency of about 16 Hz on a QL with Gold Card. Because of this lower frequency, time-critical routines are likely to fail.
An IM-2 server may end its execution with a jump to address $38 so that the timer and keyboard are updated. This may conflict with the regular keyboard job, leading to less accurate keyboard response. Programs suffering from this effect are (for example) Beta Basic and PopKey.

Interrupt Mode 0 (IM 0, default after reset but hardly ever used) executes the instruction on the data bus. On Spectator this will always be $FF (i.e. RST 38h), so IM 0 is just equal to IM 1 and is emulated as such (see above).

The Non-Maskable Interrupt (high-priority call to address $0066) is not supported by Spectator, so the second interrupt flipflop (IFF2) is always equal to IFF1 (0 or 1).

The instructions EI and DI (enable/disable interrupts) function as expected, but, contrary to the original Z80, they take effect immediately. The instructions 'LD A,I' and 'LD A,R' will properly set the parity/overflow flag to the value of IFF2 (=IFF1!).

HALT waits for an interrupt; on Spectator it operates differently in the various interrupt modes:

```
     if IM 0/1 : HALT waits for the next timer tick (50/3 Hz)
     if IM 2   : HALT really waits for the next interrupt, which is exucted
                 and RETurns to the byte after the HALT instruction;
```

The instructions 'RETI' and 'RETN' are emulated as a regular 'RET' (but they are used next to never, even in the Spectrum ROM!).

ZX INTERFACE 1

Spectator supports both the software and the hardware of ZX Interface 1.
When the Z80 interpreter reaches the byte $2A at address $0008 or the byte $23 at address $1708, the Interface 1 Shadow ROM (edition 2) is paged in (in the first 8k of memory only) and the Z80 interpreter continues emulation in this ROM. When the Z80 interpreter reaches the byte $C9 at address $0700, the original contents of the first 8k of the Spectrum ROM are restored and the emulator resumes execution of the Spectrum ROM.
You can verify the presence of the ZX Interface 1 Shadow ROM by typing the command 'RANDOMIZE USR 8' which should return 'Hook code error' (a Spectrum without Interface 1 would return the cryptic '5 M' message).
The main hardware components contained within ZX Interface 1 are:

(1) 8k Shadow ROM BASIC Extension	: see above
(2) A Microdrive Controller	: see section 'File Management'
(3) An RS232 Interface	: see section 'RS232 Serial Port'
(4) A Local Area Network connection	: not yet implemented on Spectator

Application programs accessing these hardware components by way of the standard Hook Codes of the ZX Interface 1 Shadow ROM will work just fine. However, programs using their own INs and OUTs to achieve the same thing will most certainly fail to operate...

FILE MANAGEMENT

On the 'real' ZX Spectrum, there are several ways for storing and retrieving your data. First of all, the regular SAVE and LOAD commands with a quoted name which write to and read from a tape recorder. Secondly, if you have installed ZX Interface 1 with one or more Microdrives, you can use the more advanced SAVE *"m" and LOAD *"m" commands which access the microdrives. Thirdly, several diskinterfaces exist which all have their own specific commands for accessing the drives.

Spectator supports all of the above possibilities (and more!).

(1) TAPE support

The SAVE commands with just a quoted name will write data directly to a QL disk (currently FLP1_ only). The header (17 bytes) and the actual data block are stored together in a single file which is called

```
          FLP1_SPT_<n>_<name>           { SPT meaning: SPectrum Tape }
```

with <n> the filetype (0, 1, 2 or 3) and <name> the name supplied in the SAVE command (following ZX Spectrum standards, the name is 1 up to 10 characters long). The filetype represents the type of data contained within the file:

0	: BASIC lines and variables
1	: numerical array
2	: string array
3	: code block (including screen$)

```
Examples:     SAVE "Test"                 produces  FLP1_SPT_0_Test
              SAVE "Test2" LINE 0                   FLP1_SPT_0_Test2
              SAVE "Income" DATA x()                FLP1_SPT_1_Income
              SAVE "DBase" DATA d$()                FLP1_SPT_2_DBase
              SAVE "Pattern"SCREEN$                 FLP1_SPT_3_Pattern
              SAVE "Tables"CODE 65000,512           FLP1_SPT_3_Tables
```

Since the filetype is part of the name of the produced file, it is possible to give a basic file (type 0) and a code file (type 3) the same name, although this is not advised.

It is obvious that files are stored more reliably on disk than on tape. Nonetheless, the VERIFY command is supported by Spectator, as you may want to compare a file on disk with resident data. Just as on the normal ZX Spectrum, VERIFYing a 'SCREEN$' will always fail due to the messages printed to the screen. Also, blocks of unequal length fail to verify. The message 'TAPE LOADING ERROR' is given in all these cases.

LOADing a file must be done by supplying the required name. The frequently used command LOAD "" (i.e. no name, load first program on tape) does not work and returns a TAPE LOADING ERROR. The same happens when you try to LOAD a file which doesn't exist.

```
Examples:     LOAD ""                   loads  nothing: TAPE LOADING ERROR
              LOAD "Test"                      FLP1_SPT_0_Test
              LOAD "Test2"                     FLP1_SPT_0_Test2
              LOAD "Income" DATA x()           FLP1_SPT_1_Income
              LOAD "DBase" DATA d$()           FLP1_SPT_2_DBase
              LOAD "Pattern"SCREEN$            FLP1_SPT_3_Pattern
              LOAD "Pattern"CODE               FLP1_SPT_3_Pattern
              LOAD "Tables"CODE                FLP1_SPT_3_Tables
              LOAD "Tables"CODE 33000          FLP1_SPT_3_Tables
              LOAD "Tables"CODE 65000,536      FLP1_SPT_3_Tables
              LOAD "Tables"CODE 65000,500      nothing: TAPE LOADING ERROR
```

{ Note: the clever ones amongst you may figure out how get rid of the 'TAPE LOADING ERROR' report at LOAD ""; do you know which file is opened? }

There are several other circumstances in which a 'TAPE LOADING ERROR' can occur. First, if Spectator cannot open the file you have requested, this error will be displayed. Secondly, there are some cases in which Spectator CAN open the file, but - after having read the header info - the ZX Spectrum ROM may decide that the contents are no good (e.g. because a file is too long to be stored in the buffer space you have available).

Spectator will never exceed address 65535 during LOAD. The Spectrum ROM would "wrap" the rest of the file to address 0 onwards, but Spectator simply stops LOADing. This protects the ROM area against an accidental overwrite.

On the normal ZX Spectrum, LOADing and SAVEing to/from tape is case-sensitive, so, in the above examples, LOAD "test" would fail to LOAD as it was SAVEd with an upper-case 'T'. Spectator does some clever things to make sure that this case-sensitivity remains unnoticed, so you can just as well type

```
LOAD "test"         or
LOAD "TEST"         or
LOAD "tEsT"         etc.
```

All of these commands will properly LOAD the file 'FLP1_SPT_0_Test'.

Everything that has been said for LOAD applies equally well for MERGE, except ofcourse that you can only MERGE 'type 0' files (basic lines & variables).

Application programs in machine code which call the Spectrum ROM code to SAVE data to tape will work just fine. However, LOADing data from machine coded application programs will not always work (even when they call the ROM) as Spectator has no possibility to work out the name of the required file. For LOAD, the Spectrum ROM uses a buffer of 2 times 17 bytes; the first part contains the required header info, the second part is used to store a header loaded from tape. When an application program mimics this procedure, the LOAD will be successful (provided the file actually exists). In all other cases, a 'Tape Loading Error' is bound to occur.

(2) MICRODRIVE support

The MicroDrive subroutines and matching Hook codes of the ZX Interface 1 Shadow ROM are fully emulated. Nonetheless, it is not (yet) possible to read ZX Spectrum microcartridges directly via the QL microdrives. The files on your Spectrum microcartridges must be transported to the QL in another way, e.g. via the serial line (see next section).

All eight microdrives can be used. Spectator puts the contents of each 'microcartridge' in a single file of about 135 kilobytes (270 QL-sectors). The names of the accessed files are as follows:

```
*"m";1;    accesses    FLP1_MicroDrive_1
*"m";2;       "        FLP1_MicroDrive_2
*"m";3;       "        FLP1_MicroDrive_3
*"m";4;       "        FLP1_MicroDrive_4

*"m";5;    accesses    FLP2_MicroDrive_1
*"m";6;       "        FLP2_MicroDrive_2
*"m";7;       "        FLP2_MicroDrive_3
*"m";8;       "        FLP2_MicroDrive_4
```

As can be seen, Spectator stores up to 4 'microdrives' on a single disk; this requires a capacity of 4*270 = 1080 QL-sectors, or 540 kilobytes.
It is NOT necessary to use all 4 microdrives, e.g. a 360k disk can also contain only the files 'FLP1_MicroDrive_1' and 'FLP1_MicroDrive_2'. In that case, do not access microdrives 3 or 4 as this will cause unpredictable errors.

A microdrive-file is generated as soon as a microdrive is accessed from Spectator. Initially, a pre-formatted, empty cartridge (read: file) with the name 'Nomenescio' will be produced ('Nomen Nescio' means: 'name unknown'). The actual data sectors are filled with rubbish, but the headers and record descriptors are clean and the full capacity of 127 kilobytes is available.

I strongly advise to re-format this cartridge yourself using the proper Interface 1 routines, so the most sensible command to begin with is 'FORMAT'. Insert an empty, formatted QL disk in diskdrive 1 and type (in Spectrum BASIC ofcourse!)

```
FORMAT "m";1;"<Name>"
```

with <Name> the cartridge name (1 up to 10 characters). This will properly set up the file 'FLP1_MicroDrive_1'. The formatting and checking is fully performed from the Interface 1 ROM and takes about 50 seconds (on a QL with Gold Card and 3.5" DD disks; note that the 'OK'-message already appears after about half a minute, but the Gold Card spooler still needs to flush the diskfile).

You can ensure yourself that the format has succeeded by typing:

```
CAT 1
```

This should display the cartridge name and the number of kilobytes available for file storage (should be 126!). Obviously, no files are present as yet. Now, microdrive 1 can be used for SAVE, VERIFY, LOAD, MERGE, OPEN, CLOSE, MOVE, ERASE and any other file manipulations.

Note that microdrive filenames are case-sensitive; contrary to the case of SPT-files, this sensitivity is NOT cured by Spectator in any way.

The contents of each microdrive file are as follows:

```
            /   15 bytes   header             \
            |   15 bytes   record descriptor  |
  254 times <                                 > equals 137922 bytes
            |  512 bytes   data block         |
            \    1 byte    data checksum      /
```

The 12-byte preambles of each header and record descriptor (entirely useless for our purpose) are stripped off, saving a total of 6096 bytes.
It is obvious that these microdrive files always offer 254 free sectors (or 127 kilobytes, the maximum amount that can handled by the ZX Interface 1 ROM). This is quite a lot more than the 160 sectors (80k) guaranteed by Sinclair for the real thing!

There is no objection whatsoever against copying or renaming microdrive files. You can clone Microdrive 1 to Microdrive 2 simply by typing:

```
COPY FLP1_MicroDrive_1 TO FLP1_MicroDrive_2
```

This makes backing-up quite easy (or doesn't it!). I also want to remind you that MicroDrive_1 effectively becomes MicroDrive_5 when you insert the disk in your QL diskdrive 2 (and vice versa).

There is no emulation of the 'write protection tab' of the original micro-cartridges; the QL files are always write-enabled. Do NOT make your QL disks write-protected!

(3) Z80 compressed file support

A ZX Spectrum emulator called 'Z80' has been written by Gerton Lunter for the MS/DOS PC. This emulator is quite popular and a huge number of ZX Spectrum programs in MS/DOS format can be obtained from BBS 'Tatort' (NL 50-264840). These files are in fact compressed 'snapshots' of the full 48k RAM of the Spectrum, including screen, system variables, basic listing, basic variables, machine code, machine stack and any data above RAMtop. A small header contains information about the current values of all Z80 registers (including PC and SP), so the program can be properly resumed after it has been LOADed.

Spectator supports these special files (which have extension .Z80 in MS/DOS). After having transported a Z80-file called '<name>_Z80' from MS/DOS to QDOS (using, for example, XOVER, ATR device driver, AtariDOS extensions or MultiDISCover) RENAME it to 'Z80_<name>'. Once in the Spectrum emulator, you can LOAD this file by typing:

```
LOAD "/<name>"
```

The forward slash ('/') tells Spectator that a Z80 file is wanted. Please note that <name> itself cannot exceed 9 characters, but this should be no problem since the MS/DOS limit is 8 characters.

```
Example:            LOAD "/Booty"       loads FLP1_Z80_Booty
```

When the Z80 file cannot be opened, you will get a 'TAPE LOADING ERROR'. Any extension to the LOAD command (e.g. CODE or DATA) will be ignored. When the file is found, Spectator automatically switches to Keyrow-mode since most of the Z80-files are games. Press CTRL-F3 for Keyboard-Queue-Mode. Spectator does not have a facility to create Z80-files (yet).

The header of a Z80-file created by Z80 version 1.40 and onwards contains a bit that indicates whether or not an Issue-2 Spectrum is to be emulated. The main difference between Issue-2 and later Spectrums is the keyboard handler and some older games may only work on an Issue-2 Spectrum. Spectator reads the Issue-2-bit and will apply the proper keyboard decoding. Issue-2 decoding remains active until Spectator is reset by way of CTRL-ALT-ESC.

Besides reading them, Spectator can also WRITE compressed Z80 files. This implies that you can easily transfer Spectrum programs running in Spectator to the Z80-emulator for the PC or to any other Spectrum emulator supporting the Z80 file format. Please refer to the section 'Pop-Up Menu System' for more details about Z80 snapshot saving.

On the Spectator disk you will find two utilities which may be helpful to examine the Z80 files:

Z80_Header_bas : Reads and displays the header info of a Z80 file.

Z80_Expand_bas : Although the Z80 files ported from the PC are usually compressed, Spectator can also handle non-compressed Z80 files. For examining the data contents of a Z80 file, it may be advantageous to expand it to full size. That can be done with this utility. A compressed Z80 file is expanded and stored as a new (non-compressed) Z80 file which can also be loaded directly into Spectator (if you want to). Note that the Spectrum emulator for the PC should also be able to cope with these non-compressed files, which are 48k + 30 bytes in length (=49182 bytes).

To demonstrate the decoding of the Z80 files, both programs are written in SuperBASIC, hence rather slow. However, they can easily be compiled or be converted to C or machine code for improved performance.

(4) Magic Box and DISCiPLE memory snapshot support

The "Magic Box" is a Russian disk interface for the Spectrum. Amongst other things, it can write a full 48k memory snapshot to disk. A small part of the 1k header contains information about the current values of most Z80 registers, so a program can be properly resumed after it has been LOADed. Magic Box Files can be loaded by Spectator if they are stored on disk with the prefix 'MBF_'.

The well-known DISCiPLE disk interface has a similar option. It can write SNAP files to Disciple disks. These files can be converted to QL 'SNP' files by way of the utility 'ddConvert' (see section 'Conversion Utilities'). This creates files with the prefix 'SNP_' and these can be loaded directly into Spectator.

To read such files into Spectator, type:

```
          LOAD "?<name>"      for an  MBF  file
          LOAD ".<name>"      for an  SNP  file

Example:            LOAD "?FireLord"    loads 'FLP1_MBF_FireLord'
                    LOAD ".SNAP C"      loads 'FLP1_SNP_SNAP C'
```

When the MBF- or SNP-file cannot be opened, you will get a 'TAPE LOADING ERROR'. Any extension to the LOAD command (e.g. CODE or DATA) will be ignored. When the file is found, Spectator automatically switches to Keyrow-mode.

CONVERSION UTILITIES   (!!! registered versions only !!!)

Another way to get Spectrum files on QL disks is to use one of the conversion utilities written by Jack Raats, which are part of a registered Spectator package. These allow you to transfer files from your original Spectrum disks to QL disks in the SPT_ format of Spectator. Also, utilities to convert the P-files of the Dutch HCC/SGG/Impuls association to SPT-files and vice versa are included.

```
P2SPT_exe           : unpacks a P-file, creating a number of SPT-files
                    : usage:  EW P2SPT_exe;'dev1_pfile dev2_'
                              dev1_pfile : source device and P-file name
                              dev2_      : destination device
```

```
SPT2P_exe           : packs a number of SPT-files in a single P-file
                    : usage:  EW P2SPT_exe
                              and answer the questions appropriately

bdConvert_exe       : converts the files on a BETA disk to SPT files
                    : usage:  EW bdConvert_exe;'dev1_ dev2_'
                              dev1_ : device containing Beta disk
                              dev2_ : device containing QDOS disk

ddConvert_exe       : converts the files on a DISCIPLE disk to SPT files;
                      also converts SNAP files, which can be loaded into
                      Spectator immediately (see section 'File Management').
                    : usage:  EW ddConvert_exe;'dev1_ dev2_ step'
                              dev1_ : device containing Disciple disk
                              dev2_ : device containing QDOS disk
                              step  : number of tracks to step
                                      1 for 40-track disk in 40-track drive
                                      2 for 40-track disk in 80-track drive
                                      1 for 80-track disk in 80-track drive

odConvert_exe       : converts the files on an OPUS disk to SPT files
                    : usage:  EW odConvert_exe;'dev1_ dev2_'
                              dev1_ : device containing Opus disk
                              dev2_ : device containing QDOS disk
```

RS232 SERIAL PORT

The ZX Interface 1 contains a serial port which can be accessed by way of the "b" and "t" channels. Spectator supports these channels, simply by redirecting all Spectrum RS232 output to the QL serial port (SER1) and redirecting all the input on this same port to Spectator. It is not yet possible to use other QL ports (e.g. SER2, a parallel port or even a file) for this purpose.

The baud rate of the serial port is set according to the delay-period held in the two-byte system variable BAUD (address $5CC3). The baud rate of the QL is fixed to standard values, so illegate baud rates will be rounded to the nearest legal one. The available baud rates are 300, 600, 1200, 2400, 4800, 9600 and 19200. The rates 50 and 110 are not possible on the QL and are replaced by 75 baud. Also, take care with 19200 baud, as the QL can only transmit at this rate (i.e. not receive)!

It is NEVER sufficient to set the required baud rate OUTSIDE Spectator before printing or communicating; it is ALWAYS necessary to use a FORMAT "b" command within Spectator to set the proper baud rate used by Spectator.

You can use the "t" channel to print texts or to list your programs to a printer connected to SER1. As ZX Interface 1 edition 2 is included, you can set the page width of your paper by POKEing address 23729; default value is 80 columns.

You can use the "b" channel to send and receive binary data to and from SER1. If you have a Spectrum with Interface 1, you can connect it to the QL and, after having set them to equal Baud rate (preferably 9600 baud), transmit programs from the Spectrum directly into Spectator (and subsequently store them on disk), e.g. by typing

```
          SAVE *"b"           on the Spectrum
          LOAD *"b"           on Spectator
```

and similarly for array- or code-files. This is not the fastest file tranfer you can think of, but it works and a simple cable is all you need.

STOPPING

There are three legitimate ways to end your Spectator session.
Directly from Spectrum BASIC, you may type the following command:

```
         LOAD " STOP "
```

with STOP typed as a token (SymbolShift-A on Spectrum, so CTRL-A on QL). Alternatively, you can pop up the main menu (press F1) and use the Quit option (press Q). Note that an additional confirmation is requested.

If you wish to stop Spectator from OUTSIDE the program itself, this can be accomplished by removing the job 'Spc_Vita'. In SuperBASIC this can be done by typing

```
          RJOB SPC_VITA
```

You can also use the RJOB option of QPAC-2 (or any equivalent program). Take care NOT to delete any other Spectator job as this will crash your system! Next, switch to the Spectator screen by CTRL-C. This will clean up the QL, removing all channels, jobs and interrupts owned by Spectator. If Spectator was set to Keyboard-Queue Mode, the CTRL-C is NOT needed as Spectator will be fully removed directly after the RJOB.

CLOCK FREQUENCY

The effective clock frequency can be determined in serveral ways. First of all, you can RUN a (long) BASIC program which does not have any INPUT, INKEY$, RND, PAUSE or similar commands (i.e. execution time must be fully determinate and may not contain processor-independent loops). You can then compare the execution time on Spectator to the time which this same program requires on a ZX Spectrum.
The ZX Spectrum runs at 3.5 MHz. Supposing a program on Spectator takes three times longer to execute than on the ZX Spectrum, the effective clock frequency is (3.5 MHz)/3 = approx. 1.2 MHz.

Another way to find the clock frequency is to use a machine code test loop, embracing a single instruction which is executed many (e.g. 65536) times.
If you then count the number of frames needed (and subtract the number of frames needed for just the empty loop) you can get an idea of the clock frequency of that specific instruction, provided you know the number of T-cycles the Z80 would normally need for that same instruction.

Z80 EMULATION

As far as I have been able to test, all Z80 instructions are emulated correctly: since the ZX Spectrum ROM is the basis for Spectator to function, the latter statement should be true.

Some brief remarks for Z80 freaks:

* The refresh register R is incremented by a slowly-varing random number between 4 and 15 (both inclusive) each time it is requested. Note that bit 7 remains unchanged (can be set to 0 or 1 by the user).

* The Half Carry and Subtract flag of the Z80 flag-register (f) are NOT emulated. Nearly all Z80 instructions alter these flags, but there is only one instruction which uses the result, viz. DAA. Since DAA is used only once in the ZX Spectrum ROM and since proper emulation of the Half Carry greatly reduces operation speed of Spectator, I think this exclusion is a sensible decision.

* Despite the previous remarks regarding the half-carry flag, DAA is still PARTIALLY emulated. In most cases, DAA will be preceeded by an instruction which adds data to or subtracts data from the accumulator (A), the decimal adjust taking place immediately after. Instead of working out the half-carry flag setting caused by these instructions, the 'class' of the instruction and (when applicable) the original value of the accumulator, the argument and the original setting of the carry-flag are stored in such a way that a subsequent DAA can always reconstuct the initial conditions and perform the proper 68000 'ABCD' or 'SBCD' directly (instead of using the cumbersome half-carry method). Currently, 7 instruction classes are supported: 'ADD A', 'ADC A', 'INC A', 'SUB', 'SBC', 'DEC A' and 'NEG'. Any other instruction preceeding the DAA will cause unpredictable results.
In the Spectrum ROM, the DAA occurs after an 'ADC A,A' and is therefore emulated properly.

* Many Spectrum games use unofficial Z80 instructions; the unofficial instructions that are emulated by Spectator are summed up below:

  - CBh.30h thru CBh.37h, DDh.CBh.36h and FDh.CBh.36h;
    should be 'SLL' (shift left logical); working out this instuction, it would be equal to 'SLA' (shift left arithmetic). However, the Z80 sets bit 0 of the destination after the shift. Spectator emulates 'SLL' this way.

  - DDh.??h and FDh.??h
    when an instruction has a DDh or FDh prefix, the instruction operates on IX or IY instead of HL; if the instruction contains (HL), a signed byte offset can be added, i.e. (IX+d) or (IY+d); if the instruction contains just H or L, these are interpreted as IX[hi] and IX[lo] or IY[hi] and IY[lo]; if the instruction contains no reference to HL, (HL), H or L, a warning message is displayed. If there is a reference to both (HL) and H or L, only (HL) is translated to (IX+d) or (IY+d).

- EDh.??h
  some programs use unofficial EDh prefixed instructions, which seem to have no effect at all when they are performed on a 'real' Z80. These instructions are simply skipped by Spectator.

- The undocumented behaviour of the Z80 'bit 7' instructions is properly emulated as some games seem to make use of this feature.

* It seems that many Spectrum games try to change the ROM area for some reason or another (maybe piracy protection). These ROM pokes are ignored by Spectator. The ROM protection covers virtually all memory operations.
For reasons of speed, an exception has been made for stack manipulations: these are not 'protected', so you could change the ROM by setting the SP in the ROM area and PUSHing registerpairs. I assume/hope that no game will actually do this...
By the way, the Spectrum ROM also pokes into its own code! This is obviously protected as well.

OTHER SPECTRUM EMULATORS & ADDRESSES

The following list of Spectrum emulators was taken from the manual of JPP (see below) and appended where possible. The 'I_heard_a_rumour' emulators are not included in the list. I welcome any information about existing Spectrum emulators not mentioned below.

name	system	author	remarks
JPP	PC	Arnt Gulbrandsen	needs at least 386 and VGA
SIMULATOR	C64	Whitby Software	emulates Spectrum BASIC only
SPECTATOR	QL	Carlo Delhez	you're reading its manual right now!
SPECTRUM	Amiga	Peter McGavin	needs 25 MHz Amiga for full speed
unknown	Archimed	D. Lawrence	70% of full speed on an ARM2
VGASPEC	PC	Alfonso Olloqui	needs at least 286 and VGA, no manual
Z80	PC	Gerton Lunter	copes with any processor/video card
ZM-1/2/3	QL	Ergon Development	ZM-3 is fast but less compatible
ZX	QL	Andrew Lavrov	file-formats compatible to Spectator

As quite a lot of Spectator users have asked me where they can get either JPP or Z80 (both for the PC), I have included the addresses below. Note that JPP is FreeWare (with English manual), Z80 is ShareWare (with Dutch manual).

JPP from Arnt Gulbrandsen,
Kometv. 8,
N-7036 Trondheim,
Norway.

Z80 from Gerton Lunter,
Aweg 11a,
NL-9718 CT Groningen,
Netherlands.

I have included these addresses here for your benefit. So, you can do me a favour in return: when you write to either one of them, please mention Spectator and/or my name somehow!

TESTED PROGRAMS & WARRANTY

During the development of Spectator, quite a lot of existing Spectrum programs have been loaded into Spectator as to test their behaviour in the emulator environment.

The table below lists the programs that have been extensively tested so far. All these programs have been found to run just fine on the current version of Spectator. The programs are written entirely or mainly in machine code and were commericially marketed or published in one of the many magazines. Programs written entirely in BASIC are omitted in the list below, as such programs will always function OK on Spectator.

If you should find that any of the programs below does not work properly on your machine with the current version of Spectator, I'd be most glad to hear from you! If Spectator is to blame, you'll get a repaired update for free. This way you are helping me locating bugs and helping all other Spectator users in getting a better emulator! If you have troubles with programs not mentioned below, the same applies, but please send me a disk with the program(s) (plus brief instructions) on it! (And the disk will ofcourse be sent back to you.)

'Address Manager 80+', 'Alien 8', 'Ant Attack', 'Arkanoid', 'Art Studio', 'Artist-II', 'Atic Atac', 'Automania', 'Batman', 'Bazooka', 'BCs Quest for Tires', 'Beta Basic 3.0', 'Blue Thunder', 'BMX Simulator', 'Bobby Bearing', 'Booty', 'Bounder', 'Cauldron', 'Cobra', 'Combat Lynx', 'Commando', 'Complete Bastard', 'Cookie', 'CP Forth', 'Critical Mass', 'Daley Thompsons Decathlon 1+2', 'Donkey Kong', 'Doomsday Castle', 'Dun Darach', 'Dynamite Dan', 'Dynamite Dan II', 'Enduro Racer', 'Enigma Force', 'Exolon', 'Fighter Pilot', 'Finance Manager 80+', 'Firelord', 'Fist II', 'Fractals', 'Fred', 'Freezbee', 'Full Throttle', 'Gazeteer', 'Glass', 'Head over Heels', 'Highway Encounter', 'HiSoft C', 'HiSoft Pascal', 'Hobbit', 'Hunchback', 'Hungry Horace', 'Jack and the Beanstalk', 'Jack the Nipper', 'Jet Set Willy', 'Kong Strikes Back', 'Lord of the Rings', 'Manic Miner', 'Match Day', 'Mr. Wimpy', 'Nagels Dammen', 'Nightshade', 'Paperboy of the Month', 'Paws', 'Pool', 'PopKey', 'Project Future', 'Pyjamarama', 'Raiders', 'Sabre Wulf', 'Sagittarian Pinball', 'Sinclair Logo', 'Smashout', 'Spinads', 'Stanley and the Wallbangers', 'Subway Vigilante', 'Super Sleuth', 'Supertrux', 'Tasword Two', 'Technician Ted & the Chip Factory', 'Ten Little Indians', 'Three Weeks in Paradise', 'Thunder', 'Tornado Assembler', 'Vu 3D', 'Vu File', 'Zaxxan', 'Zombies', 'ZX Forth'.

PAPERWARE

The following books were used during the development of Spectator:

- Zakboekje Z80 (in Dutch) by J.B. Vonk; Kluwer Technische Boeken 1985; ISBN 90-201-1808-0.

- 68000 deel 2, Anatomie van een super-microprocessor (in Dutch) by L. Nachtmann; Elektuur BV 1986; ISBN 90-70160-42-0.

- QL User Guide by Sinclair Research Limited & PSION Limited 1984.

- QL Advanced Users Guide by Adrian Dickens; Adder Publishing 1984; ISBN 0-947929-00-2.

- Sinclair ZX Spectrum BASIC programming by Steven Vickers; first published 1982 by Sinclair Research Ltd.

- The complete Spectrum ROM disassembly by Dr. Ian Logan and Dr. Frank O'Hara; Melbourne House Publishers 1983; ISBN 0-86161-116-0.

- ZX Interface 1 and ZX Microdrive Handbook by Cambridge Communication Limited; first edition 1983 by Sinclair Research Ltd.

- The Spectrum Shadow ROM Disassembly by Gianluca Carri; Melbourne House Publishers 1985; ISBN 0-86161-191-8.

ABOUT THE AUTHOR

I was born in 1967 and I bought my first computer, a ZX81, in 1982. With no prior programming experience at all, I started writing simple Basic programs. after about one year, it appeared to me that Basic just couldn't solve the problems I devised. So, I turned to Z80 machine code (with obliging help of Toni Baker; that is, of her book). A whole new world of possibilities opened up before my very eyes. High-speed compact code and access to all system utilities at lowest level, what else does a programmer need? In 1987, I decided to buy a QL (during sales) and found this younger brother (sister?) just as interesting as the ZX81. It also gave me a good excuse for learning 68000 machine code. As I saw that less and less people were actually using the ZX81, I sought a way for elongating its existence. I thought that (amongst other things) a ZX81 emulator for the QL would be a good way to do so, and I started writing this program in the end of 1989. Once this program was working OK, it was suggested to me that it would be a 'piece of cake' to rewrite the ZX81 emulator to become a Spectrum emulator. Without ever having touched a Spectrum before in my life, I started with the conversion. Within a few days, I implemented the most relevant Spectrum-specific routines, and already at the first test (in September 1991) the copyright message appeared and I could enter a simple Basic program. Encouraged by this successful start, the program was improved quickly and after a year (in October 1992) the first official version was released, highly compatible to the real Spectrum and packed with features. The Spectrum ROM and the vast majority of utilities and games written for it fascinated me a lot, so I now own a 'real' Spectrum 128 with ZX-IF-1 as well - better late than never! My present-day occupation as Ph.D. student in Physics offers me the opportunity to work with the latest technology in computers, but at home I still prefer to use those absorbing machines made by Sinclair...

SPECTATOR RELEASE HISTORY

Version	Date	Remarks
0.10	September 1991	First documented pre-release
0.xx	Oct'91 - Sep'92	Successive preliminary versions
1.00	October 1992	Current version (first official release)

FINAL NOTES

All rights of Spectator and this manual reserved by law (copyright);
Consequently: UNAUTHORIZED copying, hiring and lending prohibited
(please refer to section "Registration" for more details).
Speed tests performed on a QL fitted with a 16 MHz Gold Card, Spectator
running as single-tasking job at default priority.
Just to keep the lawyers happy: ZX, ZX80, ZX81, ZX Spectrum, ZX Microdrive,
ZX Interface, ZX Net, Microdrive, Microdrive Cartridge, ZX Printer, QL,
QL Net, QDOS and QL Microdrive are all Trade Marks of Sinclair Research Ltd.
The Gold Card is available from Miracle Systems Ltd, York, U.K.
Thanks to Jack Raats for providing me with useful hardware, paperware,
software, conversionware and brainware.
For more information on Spectator, do not hesitate to contact me at subjoined
address in writing:

Carlo Delhez,
Emmastraat 3,
4651 BV Steenbergen,
Netherlands.

via E-Mail on Internet:
tnndcarlo@cycl.phys.tue.nl (until May 1994)

Thank you for reading the manual & for using Spectator !!

PLEASE READ THE FOLLOWING MESSAGE ONCE MORE !!!

Spectator contains the original ZX Spectrum and ZX Interface 1
ROM codes. These codes are copyright software. Therefore, Spectator
may ONLY be used by you if you own a ZX Spectrum plus ZX Interface 1
YOURSELF. In that case, you are IN PRINCIPLE able to LEGALLY
transport the ROM codes from your Spectrum to your QL. Any usage
of Spectator by people NOT owning a ZX Spectrum and ZX Interface 1
is in conflict with copyright laws.

```
-----------------------------------------------------------------------------

*******  SHAREWARE RELEASES from the same author:  **************************

>>>>>>>  XTRICATOR, a ZX81 emulator for the QL; speed about 35% on an 8 MHz
         68008 QL and up to 200% on a 16 MHz 68000 QL; send 2 QDOS-formatted
         3.5" DS/DD brand disks or 5 IRCs for a test version.
         Or register immediately for only Hfl 50.

>>>>>>>  XTENDER, a ZX81 emulator for PC/XT/AT/386/486 MS-DOS machines, any
         video card; preliminary benchmark: from 60% on an 8 MHz 8088 up to
         900% on a 33 MHz 80386dx machine. Send 2 MS-DOS-formatted 3.5" DS/DD
         or DS/HD brand disks or 5 IRCs for a test version.
         Or register immediately for only Hfl 50.

-----------------------------------------------------------------------------
```