

The GOSH WONDERFUL ZX Spectrum ROM.

=====

Filename: gw03.rom
Version: 1.32
Stardate: 09-DEC-2004
Assembly Listing : <http://www.wearmouth.demon.co.uk/gw03.htm>
ROM Download: [gw03.rom](http://www.wearmouth.demon.co.uk/gw03.rom)

ROM modified by Geoff Wearmouth in November and December 2003.

The ZX Spectrum ROM is copyright Amstrad PLC.

Amstrad have kindly given their permission for the redistribution of their copyrighted material but retain that copyright.

The gw03.rom is forward and backward compatible with the original 16K Spectrum ROM. It is therefore very straightforward to use. You already have a manual for it but if not one is available online, again thanks to Amstrad PLC.

The gw03.rom is more compatible with the BASIC manual than the original 1982 ROM.

In addition the BASIC now has a RENUMBER and BLOCK DELETE that were standard on BASIC implementations of the 1970s and 1980s and it is much easier to keep track of FREE memory and channels.

The ROM is fully compatible with the ZX Printer and also ZX Net and Interface 1 (as of version 1.26). It works well with ResiDOS.

Version 1.29 is fully compatible with the Plus D and DISCiPLE disk interfaces. Version 1.31 rectifies a fault discovered in the tokenizer found by Andrew Owen.

Version 1.32 introduced abbreviated keyword entry as suggested by Andrew.

All such help is gratefully received at the email address on the main page.

Tokenizer

=====

The most obvious difference is in the editor as keywords have to be typed in letter-by-letter. A consequence of this is that keywords

should not be used as variable names. It is also not possible to enter PRINT CODE "BORDER" in 'L' mode.

You may wish to revert to the original editor and you can toggle between the two modes by entering 'STOP' at the command line. After loading a snapshot the keyboard mode may change and 'STOP' should be used to select the preferred keyboard mode. Games should be loaded from 'tap' files which do not affect the keyboard setup.

This method of typing in full is great especially if you work with different languages. As a shortcut a keyword can be entered in abbreviated form e.g.

10 r. gives 10 REM
10 ra. gives 10 RANDOMIZE

Extended Commands

=====

Apart from 'STOP', the other keyword that would have no meaning when entered directly is 'REM' and this has been used to add additional direct commands.

Entering 'REM' at the command line shows the commands and the version number of the ROM. The commands must be preceded by 'rem' entered in either upper or lower case.

REM streams

This is inspired by 'Stream Lister' which appeared in a microdrive book by Andy Pennell and while this is just a cut-down version, it helps keep track of which streams are in use.

An estimate of free memory appears at the top of the display and this is useful when developing programs. It is adjusted to give the same result as

```
PRINT 65536-USR 7962
```

which still works and gives the result **41473** on a standard empty Spectrum.

With Interface 1 attached, and the extra system variables created, the result is **41415**.

It is especially useful when used with Interface 1 and the M.G.T. Interfaces, and free memory will fall as extra channels are used and rise

when they are closed.

The command has revealed a new bug in the Interface 1 and if you enter

```
OPEN #4, "S"  
REM streams
```

then the new stream will be shown but you have to enter CLOSE #4 not once but twice for the stream to actually be closed.

```
REM delete {first} {last}
```

```
-----
```

The Block Delete is simple and will delete all BASIC lines between the first and last line supplied. The BASIC lines supplied must both exist in the program and be in low-high order. Only spaces are allowed as separators. The streams command will show how much memory has been reclaimed.

```
REM renumber {start} {step} {first} {last}
```

```
-----
```

This is the most powerful command and renumbers all or part of your program. Free memory will be affected one way or another and it is best to avoid using this utility when memory is low.

```
REM renumber
```

on its own renumbers the entire program starting at line 100. All instances of 'GO TO', 'GO SUB', 'RESTORE', 'LIST', 'LLIST', 'RUN' and 'SAVE LINE' will have their arguments adjusted where appropriate.

```
REM renumber {start}
```

will start renumbering at the given line number using steps of ten.

```
REM renumber {start} {step}
```

will vary the step between lines.

```
REM renumber {start} {step} {first}
```

will only renumber the last part of the program commencing at the line specified by the parameter first.

```
REM renumber {start} {step} {first} {last}
```

will restrict renumbering to a limited range of line numbers. This can

be used to smarten up a subroutine but can also be used to re-arrange sections of program. As long as they fit and don't overlap another section, then **a range of lines can be moved to a new position.**

The renumber always terminates by bubblesorting the BASIC program which is nice as bubblesorting in BASIC was one of the first things I experienced on my ZX Spectrum.

Bubblesorting a large BASIC program can take several minutes.

NON MASKABLE INTERRUPT =====

The NMI routine has had its original functionality restored and the default functionality is to perform a warm restart.

Originally the Spectrum was to use the address held in NMIADD 23728 and 23729 as a vector to the NMI routine.

These two locations are now left blank and activating the NMI jumps to address \$121C where a few bytes have been added to achieve a warm restart.

Since the Spectrum was first introduced the Interface 1 now uses 23728 and 23729 to control printer width overwriting any value you might load there when they are first paged in.

Furthermore the Plus D and DISCiPLE will suppress error messages if location 23728 is non-zero.

If no hardware conflict exist then you may activate the original functionality by making RASP (which normally holds 64) hold an odd value such as 63. The ROM will then use the address in NMIADD if it is non-zero.

Although the default Warm Reset was intended to aid the machine code developer, it enables programs like Bomb Jack (IM2) and programs on the Games microdrive to be broken into to study the programming techniques used.

However a large number of test programs were found to alter CHANS, the length bytes of BASIC lines and the address of E-LINE etc. making the success of any Warm Reset a hit-and-miss affair with anything other than one's own programs.

The Main ROM NMI is not used with the M.G.T. interfaces, ZXCF or ZXATASP which page in their own ROM's when address \$0066 is reached.

The DISCiPLE and Plus D disk interfaces

=====

To make these systems more robust the CLOSE routine now uses a look-up table with a zero end-marker and takes appropriate action if the channel identifier is not present with the standard command e.g. CLOSE #7 (instead of the M.G.T. command CLOSE #* 7).

The console input routine which accepts a single key only has been improved as the version 1.28 routine only worked with the Plus D. Channels opened by the M.G.T. Interfaces show up in the streams display with a letter 'D'.

Other Features

=====

The development has always been short of room. Some well known imperfections have been altered to give a more satisfying response and a full list follows.

Source: Understanding Your Spectrum by Dr. Ian Logan, 1982.
(12 bugs listed in appendix)

- 1) The 'division' error - is a misnomer. The inaccuracy mentioned occurs in the DEC_TO_FP routine and by switching the multiply and division operations (at \$2CDA) then 0.5 is given the floating point form 80 00 00 00 00. The suggested fix is ignored.
- 2) The '-65536' error e.g. PRINT INT -65536 gives -1. Dr. Ian Logan's fix applied (with mods) (at \$3221) and other code sections (at \$30E5) NOT removed as suggested.
- 3) The 'ZX81 program name' routine (at \$04AA) removed along with REC_EDIT. The above space has been re-used.
- 4) The 'CHR\$ 9' error corrected by calling PO_ABLE in preference to a terminal jump to CL_SET/PO_STORE.
- 5) The 'scroll?' and 'Start tape' errors corrected by new routine CONS_IN which only accepts a single ASCII key (or ENTER).
- 6) The current cursor error corrected by updating E_PPC_hi instead of E_PPC_lo.
- 7) The 'leading space error' has not been corrected and this means that

listings on screen or paper will appear exactly as they do in books and magazines. (Originally this was corrected).

- 8) The 'K-mode' error has not been corrected as the preferred mode is letter-by-letter input of tokens.
- 9) The 'CHR\$ 8' error has been corrected as suggested by Dr. Frank O'Hara.
- 10) The 'SCREEN\$' error has been corrected by substituting the suggested RET instruction. (at \$257D)
- 11) The 'STR\$' error has been corrected by removing the extra zero from the calculator stack as suggested (at \$2E25). Space was created by changing a nearby JP instruction to JR.
- 12) The 'CLOSE' error has been corrected by checking the status of the stream and issuing an error message if it is already closed. This is more in keeping with the OPEN command which issues a message if the user stream is already open. For the benefit of M.G.T. disk interfaces a look-up table with a zero end-marker is now used.

Source: The Complete Spectrum ROM Disassembly. by Dr. Ian Logan and Dr. Frank O'Hara, 1983 (various additional features listed).

- 1) The NMI bug has been corrected and the logic changed as suggested on Page 2. The new default set-up is to perform a warm reset using some original code. Setting RASP to an odd value allows the original NMIADD vector to be used.
- 2) Simple strings are not excluded when saving DATA - on Page 22. e.g. 10 LET a\$ = "dodo" : SAVE "animal" DATA a\$()
These are now rejected as they won't load back in.
(credit: First fixed by Dr. Ian Logan in the Interface 1 ROM).
- 3) There is now an end-marker for the CLOSE STREAM LOOK UP table although this was not required in the original stand-alone ROM.

Source: ZX Spectrum BASIC programming by Steven Vickers, 1982.
(discrepancies)

- 1) Line number should be optional in SAVE "some name" LINE - Page 133. Not Fixed for compatibility reasons although it is a 1-line change. If RUN can default to zero then autorun should default to the same.
- 2) CLEAR does a RESTORE (Page 124).

This seems to be an error in BASIC manual rather than ROM - ignored.
It is not clear what functionality Dr. Vickers intended. One could
make a logical case for either.

- 3) "Notice that the numbers in a DRAW statement can be negative, although those in a PLOT statement can't" - Page 92
Fixed. $0 \leq x \leq 255$. $0 \leq y \leq 175$ else Error B.
- 4) Similarly the POINT (x,y) function allowed negative coordinates.
Fixed. Error B unless $0 \leq x \leq 255$. $0 \leq y \leq 175$. Page 153.
- 5) The ATTR (y,x) function allows negative and invalid coordinates.
Fixed. Error B unless $0 \leq x \leq 31$ and $0 \leq y \leq 23$. Page 152.
- 6) The SCREEN\$ (y,x) function allows negative and invalid parameters.
Fixed. Error B unless $0 \leq x \leq 31$ and $0 \leq y \leq 23$. Page 154.

Source: The Pitman Pocket Guide to the Sinclair Spectrum by Steven Vickers,
1984. (discrepancies not previously mentioned.)

- 1) RESTORE. "Don't specify numbers > 9999, as the program may crash."
To be pedantic > 16383 - see below.
- 2) "'Statement lost' can occur with RUN, GO TO and GO SUB when the line number is between 32768 and 61439."
Fixed by new routine which checks SAVE LINE, LIST, LLIST, RUN, GO TO, GO SUB and RESTORE for invalid line numbers.
- 3) "Due to a bug, if you bring in a peripheral channel and later use a colour statement, colour controls will be sent to it by mistake."
Fixed by ensuring that the screen is first selected by CLASS_07 and CLASS_09 routines.
- 4) EDITING KEYS TABLE
When inputting from the network or RS232 or microdrive file,
code 6 (comma separator): inserted in buffer.
("This is a bug. It should work like CHR\$ 14").
Not fixed. The workaround (using ' as an output separator works O.K.)

Source: www.nonowt.com "Bugs in the ROM"
(many already covered. Some are Programming Guides rather than errors.)

- 1) The Monopolizing of IY Error.
Although not strictly an error, the manual does not mention the restriction as did the ZX81 manual. (Oh yes it does!)

Not fixed for compatibility reasons.

The interrupt routine continues to use the IY register.

- 2) The PR_CC error (credit: Dilwyn Jones 1983).
Not fixed. The system variable 23681 should not be used.
- 3) The CLEAR PRINTER BUFFER Bug.
The COPY command still needlessly clears the printer buffer but the possibility of an 'Out of screen' message with for example,
PRINT : COPY : PRINT
has been removed by setting bit 1 of FLAGS during the routine.
- 4) The MAIN-4 COPY-BUFF Error.
The possibility of an 'Out of screen' message has been removed by having CLEAR_PRB set bit 1 of FLAGS to indicate that the ZX printer position rather than the screen position should be updated. 'LPRINT "p"; : PRINT' and then 'PRINT' works OK. Minor.
- 5) The MAIN-4 HALT instruction not corrected as it is no longer an error. Its original purpose as a safe resting place for the program counter when a return is made with interrupts disabled has been restored. The NMI fix provides a clean means of continuing should the situation arise. i.e. should a programmer forget to enable interrupts before returning to BASIC. The HALT instruction was specifically put there for that purpose? [1]
- 6) The WRITE TO ROM at \$0000 by SKIP_CONS has been avoided by simply placing the unwanted values where the wanted value will go.

The WRITE TO ROM by the SCROLL routine (credit: P.Giblin) has been avoided, as suggested, by ensuring that the full 24 lines are not scrolled.
- 7) The unimplemented e-to-fp calculator instruction could be removed by assigning \$3C to 're-stack'. Five calculator routines would require alteration. This would only gain two extra bytes of ROM space and has not been done for compatibility reasons and the meagre benefit.
- 8) The INKEY\$ #0 Error. This could apply to any stream although streams 0 and 1 read from the keyboard by default. If the selected stream has been attached to the keyboard then the null string is almost always returned. Not fixed as of no practical use.

Miscellaneous BUGS and features.

- 1) In graphics mode, keys V, W, X, Y and Z give inappropriate keywords.

Fixed by not storing a key if higher than 'U'.

- 2) A typo like LIST 40000 was silently changed to LIST 7232. Fixed. As an error is now given, the modifying code (AND \$3F) has no effect and has been left in.
- 3) ED-RIGHT altered to prevent the cursor being positioned between a control character and its parameter. (credit: Andrew Owen)
- 4) Pressing ENTER while in Graphics or Extended mode did not cause a reversion to 'KLC' mode. Very minor indeed. As with most of these, I am embarrassed to mention it.
- 5) It was the intention that stream 0 should be associated with the INKEY\$ keyword and stream 1 should be associated with the INPUT keyword but the order of routines in the INPUT command routine prevented the latter and associated both commands with stream 0. This has been fixed and

```
OPEN #1,"S" : INPUT a$
```

now gives Error-J as it should have done originally.

Technical Information

=====

The Renumber will successfully renumber lines that contain machine code in REM statements in particular if they contain chr\$ 13 and chr\$ 14.

If there aren't twenty free bytes available to stack the four parameters or their defaults then the Renumber will fail at the start with the report 'Out of memory'.

The Renumber can still run out of memory if, for example, a lot of GOTO 10 statements are changed to GO TO 1000. If that occurs then it will resort to completing the number using GO TO VAL "1000" which can actually increase free memory.

The NMI checks the value of RASP 23608 and if bit 0 is set it uses the address of NMIADD (23728 and 23729) as the service routine. Normally RASP has bit 0 reset and the NMI service routine performs a Warm Reset as happens with the Enterprise Elan. Many programs use extra protection which means that you may still not arrive at the command line.

Programs that use ROM pages to vector Interrupt Mode 2 mode should work as locations \$38FF, \$39FF, \$3AFF and 3BFF address the 16 bit values \$FFFF.

The ROM can be used with most emulators.

The BACTERIA emulator uses the spare space before the character set for its help system but as long as F1 isn't pressed, I have found no problems.

The ROM will not be compatible with the Opus Discovery Disk Interface which copies substantial chunks of ROM into RAM, sticks a RET on the end and executes them there.

EXAMPLE SESSION WITH ZX Spectrum

```
goto10                ; any case, spaces are seldom required
restore 65000          ; invalid numbers can crash machine - OK now.
print int -65536       ; gives expected result
print 1/2=.5          ; evaluates as true
print int (.5+.5)     ; prints 1
rem streams           ; displays standard streams 41473 bytes free
print "A"; chr$9 ;"B" ; prints a gap between characters
print : copy : print "Ok" ; no 'Out of screen' error
print attr(255,255)   ; now gives 'Integer out of range' error
ra.                   ; gives RANDOMIZE
g.                     ; gives GO SUB
```

EXAMPLE SESSION WITH DISCiPLE or PLUS D with writable disk inserted

```
rem streams           ; displays standard streams 41473 bytes free
open #4;d1;"test" out ; open file for output attach to stream 4
rem streams           ; displays letter 'D' next to 4.
print #4;"Howdy"     ; writes text to file
close #4              ; normally crashes machine - now GDOS error.
close #*4             ; correct DISCiPLE syntax
rem streams           ; displays standard streams 41473 bytes free

open #1;d1;"test" in  ; open file for input
input a$              ; inputs from file not keyboard
```

```
print a$           ; prints 'Howdy'
print inkey$       ; INKEY$ still from keyboard
close #*1          ; INPUT from keyboard again
cat 1              ; catalog drive 1
load p1            ; loads first program
```

EXAMPLE SESSION WITH Interface 1 with writable microdrive inserted

```
rem streams        ; displays standard streams 41473 bytes free
open#7,"s"         ; associates screen with stream 7
close #7           ; stream not closed interface initialized
rem streams        ; shows 'S' against stream 7
close #7           ; This works the second time.
rem streams        ; Standard streams shown 41415 bytes free
open #9,"n";2      ; links stream 9 to networked station 2
rem streams        ; shows letter 'N' against 9 memory reduced.
close #9           ; network channel reclaimed
rem streams        ; Standard streams shown 41415 bytes free

open #0,"n";2      ; Tries to accept commands from other Spectrum
rem streams        ; shows memory leak
```

Gosh Wonderful ROM and ResiDOS

ZXCF is a Compact Flash card for the ZX Spectrum designed by Sami Vehmaa. It allows a choice of ROMS, files and snapshots to be held in non-volatile RAM within a smart interface.

ResiDOS is the clever BASIC extension, designed by Gary Lancaster, that operates the Flashcard interface and provides the optional task manager.

My experience has been through the MS-DOS RealSpec emulator. The GW03 ROM can be used directly from within the emulator. Just stick the file gw03.rom in the RS home directory and select it using F3.

ResiDOS emulation allows you to experience the gw03 ROM on a whole other level, creating what would be, in real hardware, my dream machine.

If you have the emulator working, then this is how to use the gw03 ROM from within ResiDOS.

Backup your realspec.ini file first.

Go to Sami Vehmaa's site

<http://user.tninet.se/~vjz762w/>

click on 'News' and download the command reference. Print it off for later.

Next download "config,hdd(cf)..." realspec_r14.zip which contains

realspec.ini	(realspec config)
hdd0.HDF	(16Mb emulated Flash Card)
zxcf.bin	(1 Mb emulated RAM)
allegro.cfg	(sound config - not important)

Type realspec and press RETURN when prompted

Press Ctrl-F4 and then Esc

At this point the ZXCF initialization screen should appear.

Wow! That covers normal emulation. Experiment and come back.

Next go to Gary Lancaster's ResiDOS site

<http://www/zxplus3e.plus.com/residos/index.html>

and download the tapfile for the latest upgrade - currently
residos181cf.tap

Restart the emulator initializing with Ctrl-F4 Esc as before.

Now select the tape file and load it.

When prompted, reply with 'u' for upgrade and upon
rebooting the version number should be as expected (currently 1.81).

Now use ALT-F9 to load a memory block in this case the ROM file.

Use values start = 32768, length = 16384 and tab down to
LOAD MEMORY BLOCK, press ENTER and select gw03.rom

Type SAVE % "gw03.rom" CODE 32768,16384

Type CLEAR 29999 and then RETURN.

Type LOAD % "gw03.rom" FORMAT 1

Type FORMAT % "gw03"

From now on whenever you type realspec it will use the gw03.rom after initialization with Ctrl-F4 and Esc.

The Sea Change ROM

I may as well explain this here. It's an experimental ROM that ZXCF handles. On real hardware install this as a non-Basic ROM on a Spectrum with Interface 1 attached.

After CLEAR 29999

```
Type LOAD % "sc01.rom" FORMAT 0
Then FORMAT % "sc01"
```

```
Try OPEN #7,"T"
PRINT #7,"Testing"
CLOSE #7
```

to send the text to an RS232 device. Note. that the Interface 1 ROM is never used with this setup. All that is of interest is the hardware. Network to another Spectrum and pass characters between the two. The network is not totally compatible with the Interface 1 and you will need two ResiDOS machines to pass programs between each other.

To a certain extent, it can be emulated in RealSpec. It requires that the realspec.ini file be altered to use Interface 1. The Communication Ports need to be set up with the addresses of BCHAN_IN and BCHAN_OUT in the sc01.rom. These are currently IN 1843 OUT 19A5. You can redirect output to a file SERIAL.BIN

Download the latest 16K ROM [gw03.rom](#) for use in your emulator. Also available as a zipfile [gw03.rom.zip](#)

[1] The explanation for the HALT instruction is a little fanciful. I don't think consideration was given to the assembly language programmer elsewhere. It may have been a means of communicating a signal to some external hardware. As it was never used, it could be changed to EI but it is surprising how many crashes on Interface 1 (loading a large program) and the DISCiPLE disk Interface (CLOSE #4) end up at this location with interrupts disabled. For authentic hardware crashes the HALT should remain.