The Looking Glass ROM - LG18.rom
Documentation ROM Version 0.07
Date: 29th July 2018

Welcome to the Looking Glass ROM for the 16K and 48K ZX Spectrum.
This ROM is a development of the Gosh Wonderful ROM GW03.
That ROM is a development of the original ZX Spectrum ROM which is the work of Steve Vickers and John Grant. I haven't altered the BASIC at all. Respect.
The reason for the name change is that the GW03 ROM is still viable and maintained but I have had to remove the renumber and block delete from this ROM to add all the features that should be present in a fully tokenised ROM.

While the GW03 ROM had a mode that used single letter entry of keywords, Extended Mode was necessary when using keywords in response to the INPUT prompt.

The LG ROM presents two enchanting and opposing environments in which to edit and run Sinclair ZX Spectrum BASIC. To switch between the two worlds enter 'stop' (4 letter word) or 'STOP' (Symbol Shift A) at the command line. Which one to use depends on the mode you are in.

Both modes are completely compatible with the original ZX Spectrum and in keyword mode you will be hard pressed to tell the difference.

One clue is that all of the imperfections have been corrected. There weren't many but try
PRINT INT -65536 or PRINT INT -65535.5. This gives the correct answer but the standard Spectrum returns spurious answers such as -1. Note. If you hold down the 'T' key then the screen will quickly fill with repetitions of the command RANDOMIZE. I wasn't going to take all the fun away.

Another clue is that if you enter 'REM' or 'rem' as a direct command you will see a simple screen showing the new options.

```
THE LOOKING GLASS ROM

Enter REM k for Keys map.

Enter REM s for Streams map.

Enter REM for this screen.

Enter STOP or stop to toggle
      tokenizer.




 0 OK, 0:1
```

If in keyword mode, you enter 'STOP' then you enter single Letter mode. Now you enter ALL BASIC words and symbols as single letters , numbers, graphics or symbols. In this mode SYMBOL-SHIFT is much simpler and produces the red symbol associated with the key, either on the key or below it. The exception is the letter 'I' which has two words associated and this takes the copyright symbol associated with the 'P' key which has two single letter symbols. For wider compatibility 'Q' has the © symbol too and keys 'W' and 'E' must be blank as no tokens should be available as a single keypress.

Extended Mode is available but restricted to the colour control codes on the digit keys and in that respect is used exactly as on the original Spectrum.

Originally the ZX Spectrum had 10 symbols on the digit keys and 26 symbols associated with the alpha keys. However for ease of typing keywords such as 'AT' and 'THEN' replaced the symbols which were then relegated to the faceplate below the keys.

With keywords removed the single character symbols can regain their rightful place.

'REM k' shows the keymap for the symbol keys.

With keyword mode you see

The symbols coloured magenta are accessed via Extended Mode.

With single letter mode you see
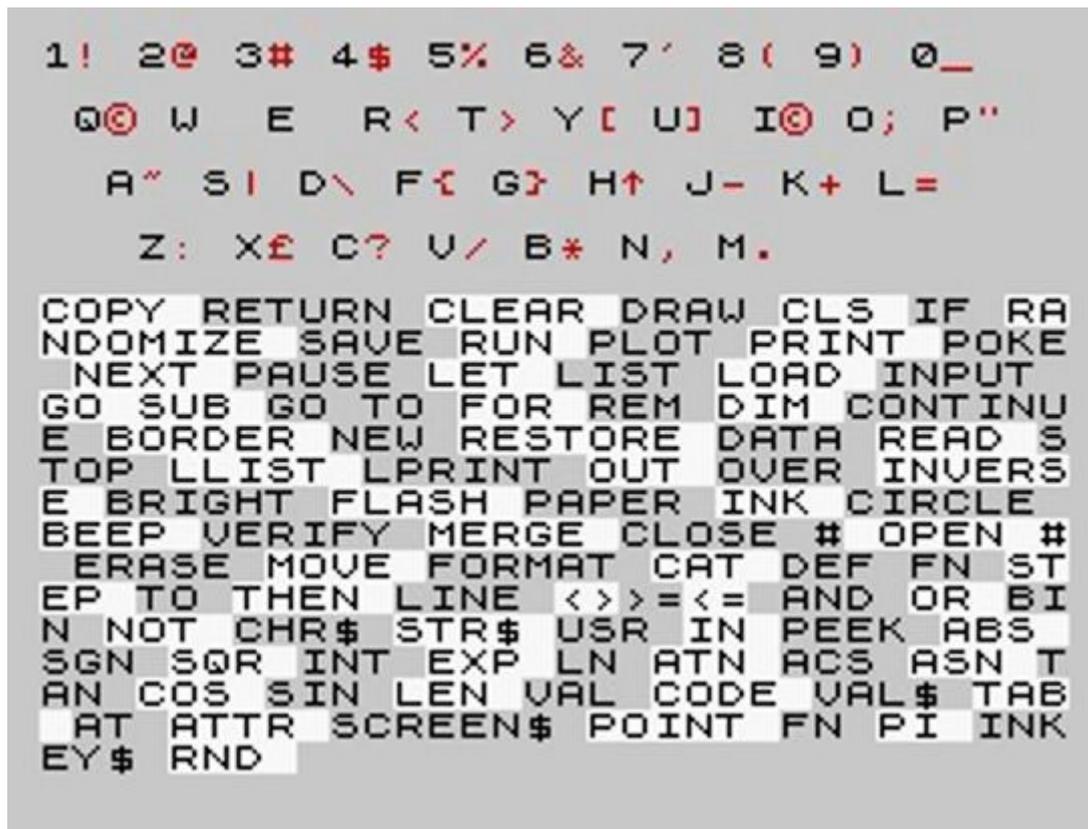
Much better. All the symbols are accessed with Symbol Shift.

Both maps have a list of the BASIC keywords. This is a bit like the command 'vlist' on the Jupiter Ace.
I always found the list of commands and functions displayed on the original ZX Spectrum enticing. I just had to learn what each one does.
For someone who knows a little BASIC then this keymap is very useful, even more so if using a PS/2 keyboard and a bare board.

From the Steve Vickers manual try the expression

10 PRINT val"val""val""""2"""""""

The same expression can also be used in response to INPUT.

The final screen is the Stream Lister and I owe the inspiration for this to Andy Pennell who wrote software, magazine articles and books for the ZX Spectrum. Mine is just a simple version of his utility and shows the letter of any channel attached to an open stream.

You are free to alter streams 0 to 15 and for this reason the ZX Spectrum has three of its own which you can't alter -

'K' for the lower screen and keyboard e.g. the 'scroll?" prompt

'S' for the upper screen e.g. tape messages

'R' for Writing to RAM. e.g. STR$ writes a number to RAM workspace and then copies the result to a variable for instance.

At the top of this screen is the version number alongside an estimate of free memory.


```
 LG v1.07        Free: 41473

253              K
254              S
255              R

0                K
1                K
2                S
3                P
4
5
6
7
8
9
10
11
12
13
14
15

0 OK, 0:1
```

If you get stuck then just enter REM at the command line.

I haven't altered the BASIC so the original 1982 BASIC Programming Manual by Steven Vickers is as relevant as it ever was.

Have fun,

Geoff Wearmouth.

I am indebted to Phoebus Dokos and Garry Lancaster for help and guidance with the ROM.