



Introduction

Microcomputers released to the home market in the mid nineteen eighties came with a BASIC Interpreter. The **B**eginner's **A**ll-purpose **S**ymbolic **I**nstruction **C**ode constructed in a FORTRAN style of one-to-a-line statements. Variants evolved among home computer manufactures to become quite sophisticated and yet small enough to fit within memory constraints of the day. Computer Magazines published BASIC code lists for Games and Utilities and for a while BASIC became the de-facto standard for introducing beginners to Computer Programming.

In nineteen eighty-four during the college summer recess, I managed to get some work experience in the computing department of Aberystwyth University. I spent most of my time etching circuit boards, but it was also to be my first sighting of the Sinclair QL and much talked about multi-tasking operating system. The QL was under reviewed in particular the PSION business programs Quill (Word Processor), Abacus (Spreadsheet), Archive (Intelligent Database) and Easel (for Drawing Charts etc.). It had an external ROM with what I believe included a trial release of the SuperBASIC Interpreter.

The Sinclair QL

I bought my first QL (Quantum Leap) computer in 1985 a few months before the price dropped from £399 to £199. My experience of programming at the time was fledgling, a college introduction to machine code, a basic knowledge of some Forth commands and a few lessons of BASIC on an BBC micro. Starting a new job, we had just received our first IBM PC, the display was green characters on a black screen.

An early addition to my QL setup was a Trump card increasing RAM to 640kbytes and expanding my storage capacity with dual 3½" Floppy disk drives. It came with a release of Toolkit II which improved and extended the SuperBASIC list of Keywords.

The drawback of those times, computer platforms weren't fast enough to satisfy the growing demands of running BASIC code through the Interpreter. Writing your program in Assembly or Machine code greatly increased the speed of execution. I did try my hand at writing in Assembly Code, then along came SUPERCHARGE and Compiling SuperBASIC was a much easier and less time consuming aspect.

The QBITS Name

My first meeting with Steve Bourne was as members attending a QUANTA club held in the old school hall in a Village called Lolworth near Cambridge (UK). I gained a lot of programming advice and help from the members. Steve had just begun selling QL hardware and encouraged me by suggesting he sold copies of my fledgling Progs. I vaguely recall a conversation discussing QL Bit and Bobs and from which the name QBITS for his Trader's name and my Software ensued. The QBITS Progs became an added contribution to Steve's wares as he trawled around different QL Club venues and shows back in the late eighties and early nineties.

QBITS Software

Late 1980's display of QBITS Software that Steve carried as part of his Trading stock. As I recall I think some were Compiled with SUPERCHARGE.



QL SuperBASIC

The QL User's Guide introduces SuperBASIC and instructions on programming. Starting with **WINDOW's & BORDER's** I then began exploring the variety of ways in which to display Character fonts. I used **PRINT** with different colours and backgrounds utilising **PAPER, STRIP & INK**. A bigger impact was dropping the **AT** line/column Keyword for the more versatile **CURSOR**. Used with **CSIZE** and **OVER**, I could create different font sizes and even 3D affects. **CLS** options **FILL\$, LEN, SCROLL, PAN** added further to the variants of character displays.

QL SuperBASIC Character Strings

CURSOR x,y :**PRINT** String\$& **FILL\$**(' ',SL- **LEN** (String\$))

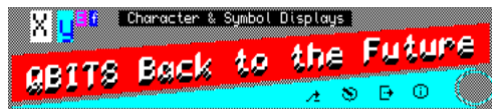
Maintains a set string length by filling EOL with 'spaces'

CURSOR x,y:**PRINT** **FILL\$**('0',4-**LEN**(number%)&number% [0000-9999]

Strings for units, ten, hundreds, thousands etc. where numbering expands from right to left.

CURSOR gx,gy,px,py:**PRINT** String\$ (graphic gx,gy coordinates px,py pixel offset)

The **CURSOR** option where **PRINT** is positioned relative to the Graphic coordinate system.



CSIZE 2.1:Str\$="QBITS back to the Future": **FOR** i=1 **TO** **LEN**(Str\$):**CURSOR** 20+i*12,60-i:**PRINT** str\$(i)

This uses the **FOR** Loop to generate a **CURSOR** offset and **PRINT** a string - character by character.

QBITS QL Symbols use

Navigate **QBITS Progs** using the Cursor keys **← ↑ ↓ →** for Actions **—** Spacebar and **↵** Enter key. **BLOCK** commands are used to provide Spacebar and Enter tail.

CURSOR24,20:**PRINT**"Select using **← ↑ ↓ → — ↵** **BLOCK**#0,12,3,130,24,5:**BLOCK**#0,2,4,198,22,5

QBITS Revival - QL Emulators

By the early nineties my QL involvement was in decline. Then in 2004 I downloaded a copy of Jimmy Montesinos **QL2K Emulator** and my interest in SuperBASIC was rekindled. Today there are QL Emulators for Desktops, Laptops, Tablets using MS, Mac or Linux Operating Systems. However, for now my choice is MS with **QPC11**.

QBITS Progs

This Collection of **QBITS Progs** was assembled with the **QPC11 Emulator**. However, most of the Progs should LRUN on other QL Platforms with expanded memory and relative Toolkits of updated and expanded Keywords. You may need to tweak some of the code and by exploring the QBITS Progs the reader will hopefully gain understanding of the simple to the more complex use of the **S/SuperBASIC Environment**.

QBITS Boot & Config

At start up the S/SuperBASIC Interpreter will Load and Run a program called '**Boot**' from the allocated default drive. This file is used to Load Extensions to the O/S and some Executable Programs to memory.

(See Page 8) Bottom left of the **QPC Configuration - Device** page - are **Boot options** - run from **FLP 1** or **2** or any of the **WIN1** to **WIN8** drives. The Default is **WIN1**, where the Boot file will need a link to **LRUN win1_QBITSBoot_bas**.

QBITSBoot_bas file when LRUN prompts for a default Source Drive **dev\$**. It then displays the generated **QBITSConfig** entires, these are common variables used by the **QBITS Programs**. Settings for **gx**, **gy** locate the QBITS backward compatible 512x256 screen size to sit within the higher screen resolutions of the QPC11 Emulator. When Exiting a QBITS Program LRUN dn\$ [set to 'dev\$&'QBITSProgs_bas'] returns to the **QBITSProgs** Menu program.



Pressing a KEY DELETE's the old and Creates a new **QBITSConfig** file and then LRUN's **dev\$&'QBITSProgs_bas** to display QBITS Progs Menu.

For those Progs using extended **Load/Save Options**: Settings for **dn%**, **dm%** and **Drv\$(dn%)** are linked with a list of Storage Devices names, 'mdv1_', 'flp1_', 'win1_', etc.

Pressing CTRL+SpaceBAR and typing ED the **QBITSConfig** entries can be Edited.



QBITSBoot_bas Code

100 REMark **QBITSBoot_bas** (QBITS Boot 2023 Review - QPC2)

120 **dev\$**='win1_':MODE 4:WINDOW 512,220,0,0:PAPER 2:BORDER 5,7:CLS:INK 7

140 CSIZE 2,1:**QBold** 190,12,'QBITSBoot':**QBold** 180,74,'QBITSConfig':CSIZE 1,1

150 BLOCK 320,90,86,100,0:**QBold** 92,42,'Select Source Drive? [ie win1_]'

160 CURSOR 356,42:INPUT dr\$:IF dr\$<>":dev\$=dr\$:END IF :**QBold** 356,42,dev\$

170 str\$="Press a KEY to Continue OR 'CTRL+SpaceBar' to Edit Settings"

180 CSIZE 0,0:CURSOR 66,194:PRINT str\$:**RESTORE** 260

:

200 REMark *** **QBITSConfig Settings** ***

210 **gx**=0:**gy**=0 :REMark SuperBASIC Screen Coordinates

220 **dn\$**=dev\$&'QBITSProgs_bas' :REMark QBITS Menu Return Path

230 **dn%**=0:FOR d=0 TO 15:**READ** d\$:IF d\$=dev\$:**dn%**=d:END FOR d

240 **dm%**=15 :REMark dn% source : dm% max

250 REMark *** **Device List** ***

260 DATA 'mdv1_', 'mdv2_', 'flp1_', 'flp2_', 'ram1_', 'ram2_', 'dos1_', 'dos2_'

270 DATA 'win1_', 'win2_', 'win3_', 'win4_', 'win5_', 'win6_', 'win7_', 'win8_'

Note: Change settings as required for default drive **dev\$** - Screen **gx gy** coordinates, return file address **dn\$** and for Drive allocations **dn% dm% drv\$(dn%)**

290 REMark *** **QBITSConfig Display** ***

300 **RESTORE** 260:STRIP 0:INK 7

310 CURSOR 100,106:PRINT 'Screen Coordinates : gx=';**gx**;;;gy=';**gy**

320 CURSOR 100,120:PRINT 'QBITS Menu Return : dn\$=';**dn\$**

330 CURSOR 100,140:PRINT 'drive default/max : dn%=';**dn%**;;;dm%=';**dm%**

340 CURSOR 100,152:PRINT 'drives dn% 0 TO 15 : drv\$(dn%)' :INK 4

350 FOR d=0 TO 7:**READ** d\$:CURSOR 100+d*36,164:PRINT d\$

360 FOR d=8 TO 15:**READ** d\$:CURSOR -188+d*36,176:PRINT d\$

380 REMark *** **QBITSConfig Format** ***

Note: Code to overwrite Config settings

390 PAUSE:**RESTORE** 260:DELETE dev\$&'QBITSConfig'

400 OPEN_NEW#9,dev\$&'QBITSConfig':PRINT#9,gx\gy\dn\$\dev\$\dn%\dm%

410 FOR d=0 TO 15:**READ** d\$:PRINT#9,d\$:END FOR d:CLOSE#9

430 REMark *** **ALTKEY Settings** ***

440 ALTKEY 'M','LRUN '&dev\$&'QBITSProgsSE_bas'&CHR\$(10)

450 ALTKEY 'f','LRUN '&dev\$&'QBITS_FTidySE_bas'&CHR\$(10)

470 REMark *** **QBITSProgs Set dev\$** ***

Note: Code to Set a Programs Default Drive

480 OPEN#9,dn\$:INPUT#9,a\$b\$c\$:CLOSE#9:c=c\$(1 TO 11)&dev\$&c\$(17 TO)

490 OPEN#9,dn\$:PRINT#9,a\$b\$c\$:CLOSE#9:LRUN dn\$:STOP

510 **DEFine PROCedure** **QBold**(x,y,str\$)

520 OVER 1:FOR i=0 TO 1:CURSOR x+i,y:PRINT str\$:END FOR i:OVER 0

530 **END DEFine**

QBITSProgs_bas

1000 REMark **QBITSProgs_bas** [QBITS Progs 2023 Review - QPC2]

1002 **dev\$='win2_':**MODE 4:gx=0:gy=0 :REMark Basic Settings

Note: Progs opening Code Lines used as File Header to facilitate use of common settings.

1004 **WHEN ERROr :CONTINUE:END WHEN**

1006 REMark **Import QBITSConfig Settings**

1007 OPEN _IN#9,**dev\$&'QBITSConfig':**INPUT#9\gx\gy\dn\$**dev\$:**CLOSE#9

1010 **QBProgs:QBMenu**

1012 **DEFine PROCedure QBProgs**

1013 DIM Prog\$(22,20):open#3,scr_10x10a10x10:WINDOW#3,512,256,gx,gy:

1014 WINDOW#2,500,204,gx+6,gy+4 :PAPER#3,0:BORDER#3,1,3:CLS#2

1015 WINDOW#1,500,204,gx+6,gy+4 :PAPER#1,0:BORDER#1,1,5:CLS#1@SCALE#1,1000,0,0

1016 WINDOW#0,500, 42,gx+6,gy+210:PAPER#0,0:BORDER#0,1,5:CLS#0

1017 OVER#1,1:CSIZE#1,2,1

1018 INK#1,2:FOR i=0 TO 1:CUSOR#1,44+i,8 :PRINT#1,'QBITS SuperBASIC Progs'

1019 INK#1,6:FOR i=0 TO 1:CUSOR#1,46,9+i :PRINT#1,'QBITS SuperBASIC Progs'

1020 INK#1,2:CSIZE 1,1 :CUSOR#1,44,158 :PRINT#1,'QPCII EMULATOR'

1021 INK#1,6:FOR i=0 TO 1:CUSOR#1,45+i,159:PRINT#1,'QPCII EMULATOR'

1022 OVER#1,0:CSIZE#1,2,0:**RESTORE 1028**

1023 FOR a=1 TO 22

1024 **READ x,y,str\$,\$:**QBold 1,5,12,x,y,str\$:Prog\$(a)=P\$

1025 END FOR a

1026 **END DEFine**

Note: Menu Title and Filename

1028 DATA 50, 36,'FTidy	','FTidySE_bas'
1029 DATA 50, 47,'FontEdit	','FontEditSE_bas'
1030 DATA 50, 58,'PIXELArt	','PIXArt_bas'
1031 DATA 50, 70,'3DGraphics	','3DGraphicsSE_bas'
1032 DATA 50, 81,'QLSounds	','QLSoundsSE_bas'
1033 DATA 50, 92,'Conundrum	','Conundrum_bas'
1034 DATA 50, 103,'Organiser	','Orgll_bas'
1035 DATA 50, 114,'Enigma	','EnigmaSE_bas'
1036 DATA 50, 125,'	','
1037 DATA 50, 136,'	','
1038 DATA 270, 36,'TicTacToe	','TTT_bas'
1039 DATA 270, 47,'MineDetect	','MDETR_bas'
1040 DATA 270, 58,'TileSlider	','Tiles_bas'
1041 DATA 270, 70,'Darts	','Darts_bas'
1042 DATA 270, 81,'Golf	','Golf_bas'
1043 DATA 270, 92,'Warehouse	','WHQPC2_bas'
1044 DATA 270, 103,'Karnak	','Maze_bas'
1045 DATA 270, 114,'Trader	','Trader_bas'
1046 DATA 270, 125,'Pandemic	','Pandemic_bas'
1047 DATA 270, 136,'Galaxy	','AD2375SE_bas'
1048 DATA 270, 147,'	','
1049 DATA 270, 158,'	','

Note: As a Program is highlighted a brief description is shown in WINDOW#0.

```

1051 DEFINE PROCEDURE QBold(ch,col,w,x,y,str$)
1052 OVER#ch,1:INK#ch,col
1053 FOR i=1 TO LEN(str$):CURSOR#ch,x+w*i,y :PRINT#ch,str$(i)
1054 FOR i=1 TO LEN(str$):CURSOR#ch,1+x+w*i,y:PRINT#ch,str$(i)
1055 OVER#ch,0
1056 END DEFINE

1058 DEFine PROCEDURE QBMenu
1059 CSIZE#1,1,0:KEsc 1,7,140,7:QBold 1,7,6,386,182,'Esc'
1060 CSIZE#0,0,0:KExit 1,7,156,8:QBold 1,7,6,430,182,'Exit'
1061 QBold 1,7,7,24,182,Select with ◀ ▶ ↕ ⏏ Cursors and Action with ⏏ Enter'
1062 CSIZE#1,3,0:BLOCK#1,2,4,324,184,7:y=1:max=7:c$='◀':x=210:col=0
1063 REPEAT Menu_lp
1064 IF x=210:RESTORE 1077+y:ELSE RESTORE 1088+y
1065 CLS#0:READ str$:sl=LEN(str$):CURSOR#0,248-6*(sl/2),10:PRINT#0,str$
1066 CURSOR x,24+y*11:PRINT c$:k=CODE(INKEY$(-1)):BLOCK 18,10,x,24+y*11,0
1067 SELECT ON k
1068 =192:IF x=260:x=210:max=7 :col= 0:c$='◀':IF y>max:y=max
1069 =200:IF x=210:x=260:max=11:col=10:c$='▶'
1070 =208:y=y-1:IF y<1 :y=max
1071 =216:y=y+1:IF y>max:y=1
1072 = 10:CLS#3:DF$=dev$&'QBITS_'&Prog$(y+col):QBDev:LRUN DF$
1073 = 27,69,101:PAPER#2,0:CLS#2:CSIZE#2,0,0:INK#2,7:EXIT Menu_lp
1074 END SELECT
1075 END REPEAT Menu_lp
1076 END DEFINE

```

Note: Menu Title Descriptions

```

1078 DATA 'A File Tidy Program - Review and Manage File Directories'
1079 DATA 'Change/Modify QL Character Fonts - Plus Retro ARCADE Games'
1080 DATA 'Explore PIXEL Art - Create Sprites & Retro Games etc.'
1081 DATA 'Exploring 3D Rotation Graphics - Plus Escape POD Rescue Game'
1082 DATA 'Exploring the Musical Attributes of the QL BEEP Commands'
1083 DATA 'Exploring the Functions of a 1980s Style Personal Organiser'
1084 DATA 'Type Correct Order of Letters that Spell out the Hidden Word'
1085 DATA 'Enigma - A simulation of the WWII Encipher/Decipher Machine'
1086 DATA "
1087 DATA "

1089 DATA 'The Coffee Break Challenge - Classic Noughts & Crosses'
1090 DATA 'Clear a Mine Field - Based on Mine Sweeper of the 1980s'
1091 DATA 'A Sliding Tile Puzzle Game with Numbers or MINIONS Picture'
1092 DATA 'Classic Darts - Play 301/501 or Around the Clock Face Game'
1093 DATA 'Compete over an 18 Hole Course - SCORECARD with Par & HandiCap'
1094 DATA 'Manage a WareHouse - Handle Invoice Requests & Stock Deliveries'
1095 DATA 'Solve the Maze - Your Mission Travel back in Time to Save Humanity'
1096 DATA 'As a Market Trader - Manage a Portfolio of Company Stocks & Shares'
1097 DATA 'As a Specialist - Lead a Team to Contain & Eradicate a Deadly Virus'
1098 DATA 'Galaxy Adventure AD2375 : The First Order - Alliance v Republic'
1099 DATA "
1100 DATA "

```

Note: QBDev overwrites the default Source Drive 'dev\$' at beginning of Selected Program. The program is then LRUN and opens the QBITSConfig file to Import current settings. (Q)uit Prog LRUN's dn\$ [ie dev\$&'QBITSProgs_bas']

```

1102 DEFine PROCEDURE QBDev
1103 OPEN#9,DF$:INPUT#9,a$b$c$.CLOSE#9
1104 c$=c$(1 TO 11)&dev$c$(17 TO) :REMark QBITS Menu - Drive
1105 OPEN#9,DF$:PRINT#9,a$b$c$.CLOSE#9
1106 END DEFine

```

```

1108 DEFine PROCEDURE KEsc(ch,col,x,y)
1109 INK#ch,col:FILL#ch,1:LINE#ch,x-1,y+2 TO x+1,y TO x+8,y-5
1110 LINE#ch TO x-1.4,y+1.6 TO x-1,y+2:FILL#ch,0:ARC#ch,x-1.5,y TO x,y+2,30
1111 END DEFine

```

```

1113 DEFine PROCEDURE KExit(ch,col,x,y)
1114 INK#ch,col:CURSOR#ch,x,y,1,-5:PRINT#ch,'½':LINE#ch,x,y TO x,y-4
1115 LINE#ch,x+1.2,y+1.8 TO x-1,y+1.8 TO x-1,y-2 TO x+1.6,y-2
1116 END DEFine

```

Symbols can be created with the Graphic commands **ARC**, **CIRCLE**, **LINE**, **POINT**:

```

2001 DEFine PROCEDURE KAlt(ch,col,x,y)
2002 INK#ch,col:CURSOR#ch,x,y,0,-5:PRINT#ch,'➡'
2003 LINE#ch,x+.5,y-2 TO x+2,y-2:LINE#ch,x-1.5,y-2 TO x-1,y-2 TO x+.2,y
2004 END DEFine

```

ALT 

```

2006 DEFine PROCEDURE KInfo(ch,col,x,y)
2007 INK#ch,col:CIRCLE#ch,x,y,2:LINE#ch,x,y-1 TO x,y+.5:POINT#ch,x,y+1
2008 END DEFine

```

Info/Help 

```

2010 DEFine PROCEDURE KCtrl(ch,col,x,y)
2011 INK#ch,col:CIRCLE#ch,x,y+.2,1.2 :LINE#ch,x,y+2.2 TO x,y-2
2012 LINE#ch,x-1.6,y+1.6 TO x+1.6,y-1.6:LINE#ch,x+1.6,y+1.6 TO x-1.6,y-1.6
2013 END DEFine

```



QPC11 Emulator

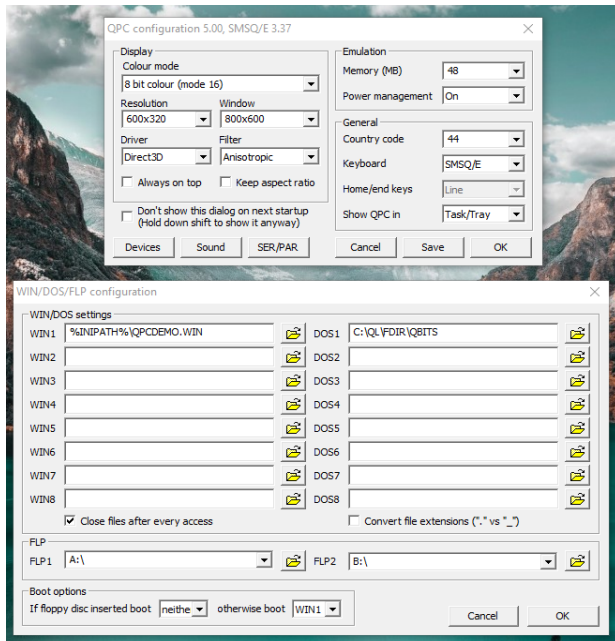
Installed and run on a Windows PC this Emulates a Sinclair QL Computer.
However, it has a far more advanced O/S with Tony Tebby's SMSQ/E
the successor to his QDOS and an updated expanded SBASIC
to the QL SuperBASIC of Jan Jones day.

Downloads: <https://www.kilgus.net/qpc/downloads/>
Also Check out: <http://www.dilwyn.me.uk/emu/index.html>

QPC11 Manual

Issued 2021 with the release of QPC11 v 5.00 it explains Installation, Concepts,
and SBASIC keywords. QPC Screen resolution and size is extended from the
original 512x256 with additional Colour Palettes.

Download and follow the documentation's instructions to Install.
Start **QPC11** and change the configuration to that shown below: -



Download **QBITS Progs** and UNZIP into a New Files Folder.
In **QPC configuration** Click on **Devices** and link **DOS1** to your
QBITSProgs Folder, press OK and then **Save**.

Press **Start** and with **QPC11** up and running exit from the demo page and in the
SuperBasic Interpreter's Command Window type: - **LRUN Dos1_QBITSBoot_bas**

Set Source Drive and **LRUN QBITSProgs_bas** to display Menu.
Select a Program with Cursor Keys and Enter.