



Sinclair QL retro-Computing



Sinclair QL retro-Computing



Returning to The Sinclair QL

As an owner of a QL I guess I'm somewhat biased due to my own background involvement, but to my mind the QL makes an excellent choice for those wishing to engage in a bit of computer programming and dabbling in early gaming software. This hankering for such I'm reliably informed by some is called retro-computing.

A first in what was to follow in Personal Computing the QL computer produced in 1983/4 came with a multi-tasking operating system called QDOS and a SuperBASIC interpreter for users to write their own programs. Bundled with the QL were four business applications; Quill a word processor, Abacus a spreadsheet, Archive an intelligent database and Easel a graphic package for charts etc.

Sadly my own QL microdrives and add-ons, floppy drives, extended memory etc. reside in the loft, not used since the TV/Monitor bought in the late eighties gave up the ghost. After a long gap, the last time I had the QL working was over ten years ago, the floppy drives were on their last legs and most of the floppies I had were unreadable.

However, searching online I came across the QL emulators and today I use copies of **QL2K emulator** loaded on my Win 7 desktop and Win 10 Tablet.

The QL user's manual

This original documentation as an introduction to the Sinclair QL, its operating system, programming in the SuperBASIC language, concepts etc. it is still a worthwhile read.

QL Emulators

There are several available for the original QL as well as its later spin offs. You can download these and run them on PC's, Desktops Laptops and Tablets under the Windows, Mac or Linux operating systems. Then there are the additional ROM's and toolkit extensions and an extensive number of useful programs all with plenty of helpful documentation available.

Check out Dilwyn's web site below for downloads, helpful information and links to other suppliers of QL software and documentation.

<http://www.dilwyn.me.uk/>

Updated: 07.09.2015

QBFTidy 2017



QL Retro Computing

At college in nineteen eighty-four during the summer recess I managed to get some work experience in the computing department of Aberystwyth University. Most of my time was spent etching circuit boards, but it was also to be my first sighting of the Sinclair QL. It was under review and being tested in particular the 4 PSION business programs Quill, Abacus, Archive & Easel.

My experience of programming at the time was fledgling an introduction to machine code, Forth, and a few lessons on an early BBC micro. And yes that primary accomplishment of how to program a print statement - 'Hello World'.

I bought my first QL in 1985 a few months before the price dropped from £399 to £199. At work we had just received the first IBM PC the screen display green print on a black background. My discovery with the QL was printing to screen came in a variety of colours and combinations of backgrounds using the SuperBASIC INK, PAPER commands. The AT command for row and column positioning and the CSIZE command creating different font sizes.

The QL SuperBASIC made an even bigger impact on my limited programming skills when I broadened my horizons with the CURSOR, and OVER commands. I could produce bold and highlighted type even giving the print a 3-D effect.

My programming skills quickly developed into writing these screen prints with the CURSOR offsets as simple FOR loops, and then grouping them into PROCedures. In constantly exploring new ways of presenting characters sets to the screen I was soon adding a few graphics.

QBFTidy 2017

Exploring Character and Graphics displays

Presentation is everything I was once told. So I believe screen displays should create a little bit of a zing to give an aesthetic tingle of enticement. Back in the eighties when I bought my first QL and was investigating SuperBASIC I was enthralled as to the multiplicity of combination that could be evoked just in creating a line of text.



114 DEFine PROCedure Display

```
116 ch=1:WINDOW#ch,512,200,0,0:PAPER#ch,0,7,3:CLS#ch
118 CSIZE#ch,3,1:STRIP#ch,7:INK#ch,0:CURSOR#ch,38,10:PRINT#ch,'A'
120 CSIZE#ch,2,1:STRIP#ch,4:INK#ch,2:CURSOR#ch,58,10:PRINT#ch,'B'
122 CSIZE#ch,1,0:STRIP#ch,2:INK#ch,4:CURSOR#ch,76,10:PRINT#ch,'C'
124 CSIZE#ch,0,0:STRIP#ch,2:INK#ch,7:CURSOR#ch,88,10:PRINT#ch,'D'
126 CURSOR#ch,104,10:STRIP#ch,0:PRINT#ch,'Exploring Character
Graphics'
128 str$=' QBITS Back to the Future '
130 FILL#ch,1:INK#ch,2
132 LINE#ch,15,78 TO 136,90 TO 136,76 TO 15,64 TO 15,78
134 FILL#ch,0:CSIZE#ch,3,1:OVER#ch,1
136 FOR char=1 TO 25
138  FOR p=1 TO 3
140   CURSOR#ch,20+p+char*13,54-p-char:INK#ch,0:PRINT#ch,str$(char)
142   CURSOR#ch,20+p+char*13,52-p-char:INK#ch,7:PRINT#ch,str$(char)
144  END FOR p
146 END FOR char
148 OVER#ch,0:FILL#ch,1:LINE#ch,98,60 TO 16,60 TO 98,68
150 ARC#ch,98,68 TO 98,60,PI/2:FILL#ch,0
152 FOR c=1 TO
2:FILL#ch,1:INK#ch,4:CIRCLE#ch,92+c*12,63+c,3+c:FILL#ch,0
153 k$=INKEY$(-1)
154 END DEFine
```

QBFTidy 2017

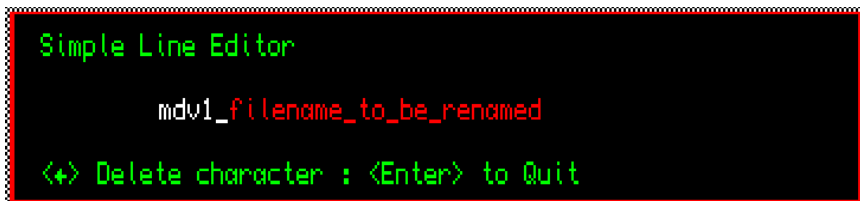
QL SuperBASIC Simple Line Editor

Merely evoking screen prints led me to wanting a means of editing my type in real time not using the SuperBASIC EDIT command and running the altered lines in my program.

I needed control over the characters in a string being able to delete or add. Here I discovered the power of the SELEct ON command used with numbers set of the ASCII Keyboard. This provided a very useful programming tool for my Line Editor.

216 DEFine PROCEDURE Line_ed

```
217 WINDOW#ch,360,60,40,120,0:BORDER#ch,1,2:PAPER#ch,0:CLS#ch
218 str$='filename_to_be_renamed':char%=23:cmax%=32
220 CSIZE#ch,0,0:INK#ch,4
222 CURSOR#ch,10, 5:PRINT#ch,'Simple Line Editor'
224 CURSOR#ch,10,45:PRINT#ch,'<¼> Delete character : <Enter> to Quit'
225 CURSOR#ch,60,25:INK#ch,7:PRINT#ch,Dev$(Dnum)
226 REPEAT ed_lp
228 FILL#ch,1:STRIP#ch,0:INK#ch,2
230 CURSOR#ch,90,25 :PRINT#ch,str$:CLS#ch,4
232 k$=INKEY$(#ch,-1):k=CODE(k$)
234 SELEct ON k
236   =48 TO 57,65 TO 90,95,97 TO 122
238     IF char%<cmax%:char%=char%+1:str$=str$&k$:END IF
240     =192:IF char%>=1:char%=char%-1:END IF
242     IF char%=0:str$=":ELSE str$=str$(1 TO char%):END IF
244     = 10:RETurn
246 END SELEct
248 END REPEAT ed_lp
250 END DEFine
```



```
Simple Line Editor
mdw1_filename_to_be_renamed
<¼> Delete character : <Enter> to Quit
```

At this point my collection of Files was growing somewhat out of hand. What I needed was a means of managing my files and so coding for QBFTidy began with a review of the programs I had written.

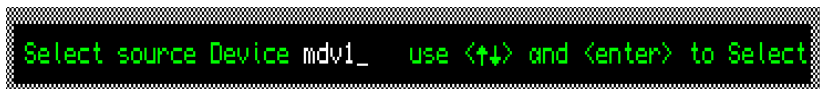
QBFTidy 2017

QL Device Selection

The SuperBASIC DIR command output as a single vertical list of filenames wasn't enough.

The QL comes with two microdrives storage devices, mdv1_ & mdv2_ and it wasn't long before external twin floppy drives were available. So if I was going to access these different devices I would need a simple selection method rather than having to type a five character devicename every time.

```
160 DIM Dev$(4,5):RESTORE
162 DATA 'mdv1_', 'mdv2_', 'flp1_', 'flp2_'
164 FOR d=1 TO 4:READ Dev$(d)
166 SelDrv
168 :
170 DEFine PROCedure SelDrv
171 BLOCK#ch,360,20,40,95,0
172 CSIZE#ch,0,0:INK#ch,4:CURSOR#ch,44,100
174 PRINT#ch,'Select source Device      use <↑↓> and <enter> to Select'
176 INK#ch,7:px%=170:py%=100:DDrv:SDR$=Dev$(Dnum)
178 END DEFine
180 :
182 DEFine PROCedure DDrv
184 ch=1:Dnum=3
186 REPEAT Dev_lp
188 k$=INKEY$(#ch,10):k=CODE(k$)
190 CURSOR#ch,px%,py%:PRINT#ch,Dev$(Dnum)
192 SELEct ON k
194   =10 :RETurn                :REMark Enter
196   =208:Dnum=Dnum-1:IF Dnum<1 THEN Dnum=4:REMark Up
198   =216:Dnum=Dnum+1:IF Dnum>4 THEN Dnum=1:REMark Down
200 END SELEct
202 END REPEAT Dev_lp
204 END DEFine
```



Note: The code for page 2,3 & 4 are to be found in the file FTidy_dev with together with a Boot file and the QBFTidy file are in QBFTidy.zip

QBFTidy 2017

QL SuperBasic Programming

So this is my journey into programming with SuperBASIC and the methodology I followed in devising my programming skills.

Opening Remarks

The first few lines of a program to identify its name, a reference to its history (version date etc), ownership, maybe a little on its concept and the requirement it is fulfilling.

Example:

```
100 Remark FTidy_bas (1986 version 1.00 QBITS)
120 :
130 Remark This program is to aid file management of storage devices.
140 Remark Selection of Device allowing different source & target devices
150 Remark Provides multiple file COPY / DELETE function
160 Remark DIR of source device, file LRUN / RENAME / VIEW
170 :
```

Definitions

It is wise to assign runtime variables and strings at the beginning of a program so that if they need to be altered they can be found quickly.

Example: 200 DIM Dev\$(16,5) :REMark Device names

As a program develops further, Procedures are added to carry out more actions. These maybe requests from Menu inputs, updating screen displays that might be messages or help prompts etc. or calls to handle any number of different actions to be carried out.

Procedures and Functions

A program is therefore a collection of interactive sets of code, procedures and/or functions. First to initialise the environment and load any strings or arrays, then a Menu that allows keyboard input to select further actions. When the program is finished a means to close down unwanted jobs, channels, windows, release RAM etc and restore the system back to its previous settings.

Program Start/Finish

This consist of a line calling the main **PROC**edures.

Example: 240 **Prog_Init : Main_Menu :Clean_Up**

QBFTidy 2017

QBFTidy Concepts

To begin with an opening display with QBITS File Tidy Title and a help screen showing the commands with a brief description of their function. The first action to select a source device, then display the directory filenames in several columns across the screen.

File DIR

Device selection envisaged as a simple scrolling up or down through an array of possible choices. Then inputting the DIR filenames from the selected device to an FLlist. This would then be used to generate the filename columns to screen.

QL Technical Guide identifies a QL filename as being up to 36 bytes in length or the equivalent to the same number in ASCII characters. For four columns of filenames, only the first 16 characters are displayed. However, selecting any of the other commands the full filename would need to be displayed in the command window below.

COPY / DELETE

Here the idea was to batch a number of files to be Copied or for Deletion. First filename needed to be highlighted, but then the ability to move on through the files listed leaving the selected ones highlighted. A final requirement was to go through the selected list with a 'y/n' before any action was carried out.

RENAME

The simple concept here was to Copy an existing filename with a new filename and then Delete the old filename. The existing filename (file\$) to be copied into a new string (str\$) then passed to a simple line editor. Again a 'y/n' option to be given before carrying out any final action.

VIEW

Filenames not always being recognisable for what they were, being able to read their contents especially the first few line seemed a logical addition. Opening a selected file and reading it requires a little fineness. If a byte file, a wrap around new line is required after so many characters. For a SuperBASIC or ASCII character file then acknowledgment of an Enter for each new line.

LRUN

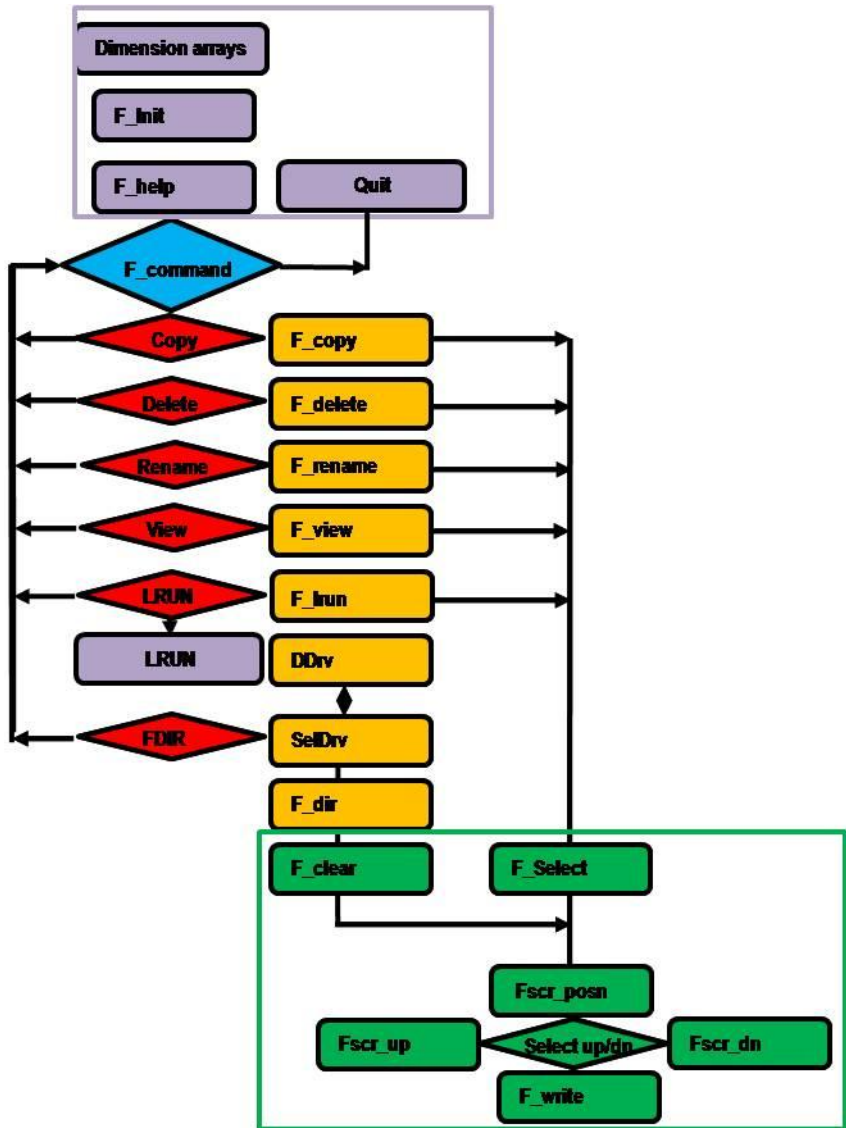
Last but not least, to be able to load and run a selected file. LBYTES or EXEC EXEC_W files I deemed not a necessary requirement at this time as I was only writing SuperBASIC code.

QBFTidy 2017



QBFTidy 2017

QBFTidy FlowChart



QBFTidy 2017

QBFTidy PROCedures

Dimension Arrays and Variables

01	F_Init	Initialise programs screens and consoles
02	F_help	Display help screen
03	F_command	Main program loop
04	F_copy	Batch / single file copy source to target device
05	F_Delete	Batch / single file delete in source device
06	F_dir	Device selection then display of files
07	F_lrun	LRUN of a selected file
80	F_rename	Rename of a selected file
09	F_edlin	Line editor
10	F_veiw	View line contents of a selected file
11	F_quit	Clear screen, Close channels
12	F_Tittle	Display title
13	SelDrv	Select source or target drive
14	DDrv	Scroll to selected drive
15	F_select	Select a filename for highlight
16	fscr_posn	Calculate a screen position for filename display
17	Fscr_up	Filename up
18	Fscr_dn	Filename down
19	F_write	Write filename to screen
20	F_clear	Clears highlighted filenames

QBFTidy 2017

100 REMark QBFTidy (update version 2017.02 QBITS)

102 DIM mes\$(4,54),Com\$(8,50),Title\$(36),Time\$(20)
103 DIM Command\$(60),str\$(32),an\$(1),k\$(1)
104 DIM Dev\$(16,5),SDR\$(5),TDR\$(5)
105 DIM Volume\$(10),Sector\$(20),file\$(200,32),Sett(200)
106 RESTORE :num%=200:lgh%=32:PLen%=0:PCol%=3

108 REMark - change as required ie win1_nfa1_dos1_ram1_etc
109 DATA "mdv1_", "mdv2_", "mdv3_", "mdv4_", "mdv5_", "mdv6_", "mdv7_", "mdv8_"
110 DATA "flp1_", "flp2_", "flp3_", "flp4_", "flp5_", "flp6_", "flp7_", "flp8_"
112 FOR Dnum=0 TO 15:READ Dev\$(Dnum)
114 MODE 4:**F_init:F_help:F_command**

116 DEFine PROCEDURE F_init

117 OPEN #5,scr_512x256a0x0 :PAPER #5,0:CLS #5:CLOSE #5
118 OPEN #5,con_448x32a32x220:PAPER #5,0:CLS #5:BORDER #5,1,4:CSIZE #5,0,0
119 OPEN #6,con_448x202a32x16:PAPER #6,0:CLS #6:CSIZE #6,1,0
120 OPEN #7,scr_448x162a32x44:PAPER #7,0:CLS #7:BORDER #7,1,2:CSIZE #7,0,0
121 OPEN #8,scr_140x28a32x12 :PAPER #8,4:CLS #8:BORDER #8,1,7:CSIZE #8,2,1
122 Command\$=' Copy Delete FDIR LRUN Rename View '
123 **END DEFine**

125 DEFine PROCEDURE F_help

126 **F_title** 'QBFTidy'
127 CURSOR #6,148,0:CLS #6,4:CURSOR #6,324,12:CLS #6,4
128 mes\$(1)=' Select COMMAND using cursor keys (← →) then ENTER'
129 mes\$(2)=' Select FILES using cursor keys / MARK with SPACEBAR'
130 mes\$(3)=' ALT-↑↓ page Up/Down Esc To Quit"
131 mes\$(4)=' To show this HELP panel press the 'h' key"
132 Com\$(1)='Copy - marked Files from Source to Target'
133 Com\$(2)='Delete - marked Files from Source Device'
134 Com\$(3)='FDIR - File Directory of Source Device'
135 Com\$(4)='LRUN - Load and Run SuperBASIC program'
136 Com\$(5)='Rename - File from Source Device'
137 Com\$(6)='View - File from Source Device'
138 CSIZE #7,1,0:OVER #7,1:CLS #7
139 FOR c=1 TO 6
140 INK #7,7
141 FOR i=0 TO 1:CURSOR #7,36+i,14+c*10:PRINT #7,Com\$(c,1 TO 7);
142 INK #7,4:PRINT #7,Com\$(c,8 TO)
143 END FOR c
144 INK #7,182
145 FOR i=0 TO 2
146 CURSOR #7,i, 8:PRINT #7,mes\$(1)
147 CURSOR #7,i,100:PRINT #7,mes\$(2)
148 CURSOR #7,i,116:PRINT #7,mes\$(3)
149 END FOR i
150 INK #7,2
151 FOR i=0 TO 1:CURSOR #7,i,140:PRINT #7,mes\$(4)
152 CSIZE #7,0,0:OVER #7,0:F=0
153 **END DEFine**

QBFTidy 2017

155 DEFine PROCEDURE F_command

```
156 INK #6,7:c=2:F=0
157 REPEAT Comm_lp
158 Time$=DATE$:CURSOR #6,148,12
159 PRINT #6,Time$(13 TO 17)&' '&DAY$&' '&Time$(6 TO 11)&' '&Time$(1 TO 4)
160 x%=c*56+24:y%=192:CURSOR #6,24,y%:PRINT #6,Command$
161 STRIP #6,4:INK #6,0:CURSOR #6,x%,y%:PRINT #6,Command$(c*7+1 TO c*7+6)
162 STRIP #6,0:INK #6,7:k=CODE(INKEY$(#5,25))
163 SElect ON k
164 =192:c=c-1:IF c<0:c=5 :REMark ← left cursor
165 =200:c=c+1:IF c>5:c=0 :REMark → right cursor
166 =72,104 :F_help:c=2 :REMark key h or H for help
167 =27:F_quit :REMark 27 = ESC /10 = Enter
168 =10:SElect ON c
169 =0:IF F>0 :F_copy :CLS #5
170 =1:IF F>0 :F_delete :CLS #5
171 =2:SelDrv :F_dir :CLS #5
172 =3:IF F>0 :F_lrun :CLS #5
173 =4:IF F>0 :F_rename :CLS #5
174 =5:IF F>0 :F_view :CLS #5
175 END SElect
176 END SElect
177 END REPEAT Comm_lp
178 END DEFine
```

180 DEFine PROCEDURE F_quit

```
181 CLOSE #8:CLOSE #7:CLOSE #6
182 WINDOW #5,512,256,0,0:PAPER #5,0:CLS #5:CLOSE #5
183 END DEFine
```

185 DEFine PROCEDURE F_title (Title\$)

```
186 PAPER #8,4:CLS #8:OVER #8,1
187 INK #8,0:FOR i=0 TO 3:CURSOR #8,4+i,1+i:PRINT #8,Title$
188 INK #8,0:FOR i=0 TO 3:CURSOR #8,6+i,1+i:PRINT #8,Title$
189 INK #8,0:FOR i=0 TO 3:CURSOR #8,7+i,1+i:PRINT #8,Title$
190 INK #8,7:FOR i=1 TO 2:CURSOR #8,4+i,2 :PRINT #8,Title$
191 END DEFine
```



QBFTidy 2017

193 **DEFine PROCedure SelDrv**

194 CLS #5:INK #5,4:CURSOR #5,18,6

195 PRINT #5,'Select source Device use <↕↘> and <Enter> to Select'

196 INK #5,7:px%=148:py%=6:**DDrv**:SDR\$=Drv\$(Dnum)

197 **END DEFine**

199 **DEFine PROCedure DDrv**

200 Dnum=8 :REMark default device - flp1_

201 REPEAT Dev_lp

202 k=CODE(INKEY\$(#5,10))

203 CURSOR #5,px%,py%:PRINT #5,Drv\$(Dnum)

204 SELEct ON k

205 =10 :RETurn :REMark Enter

206 =208:Dnum=Dnum-1:IF Dnum<0 THEN Dnum=15 :REMark Up

207 =216:Dnum=Dnum+1:IF Dnum>15 THEN Dnum=0 :REMark Down

208 END SELEct

209 END REPEAT Dev_lp

210 **END DEFine**

212 **DEFine PROCedure F_dir**

213 n%=1:ftot%=0:CLS #7:F_title 'DIR '&SDR\$

214 CLS #5:CURSOR #5,24,6:PRINT #5,'Files being selected...'

215 DELETE SDR\$&'FList'

216 OPEN_NEW #4,SDR\$&'FList':DIR #4,SDR\$:CLOSE #4

217 OPEN_IN #4,SDR\$&'FList':INPUT #4,Volume\$:INPUT #4,Sector\$

218 REPEAT DIR_lp

219 IF EOF(#4) OR n%>num%:ftot%=n%-1:CLOSE #4:EXIT DIR_lp

220 INPUT #4,str\$

221 CURSOR #5,px%,py%:PRINT #5,Dev\$(Dnum)

222 IF PLen%>0

223 IF SDR\$(6 TO 5+PLen%) INSTR str\$=1

224 file\$(n%)=str\$:Sett(n%)=4:n%=n%+1

225 END IF

226 ELSE

227 file\$(n%)=str\$:Sett(n%)=4:n%=n%+1

228 END IF

229 END REPEAT DIR_lp

230 CURSOR #6,328,0:CLS #6,4:OVER #6,1:l=LEN(Sector\$)

231 INK #6,2:FOR i=0 TO 1:CURSOR #6,148+i,0:PRINT #6,'Volume':CLS #6,4

232 INK #6,7:CURSOR #6,210,0:PRINT #6,Volume\$

233 INK #6,2:FOR i=0 TO 1:CURSOR #6,324+i,0:PRINT #6,'Sectors'

234 INK #6,7:CURSOR #6,386,0:PRINT #6,Sector\$(1 TO l-8)

235 CURSOR #6,336,12:CLS #6,4

236 INK #6,2:FOR i=0 TO 1:CURSOR #6,324+i,12:PRINT #6,'Files'

237 INK #6,7:CURSOR #6,386,12:PRINT #6,ftot%

238 OVER #6,0:INK #6,7:lptr%=0

239 IF ftot%<1:SDR\$=SDR\$(1 TO 5):F=0:PLen%=0:RETurn :END IF

240 **F_clear**:F=1

241 **END DEFine**

QBFTidy 2017

243 **DEFine PROCEDURE Fscr_posn**

244 IF n<1:n=1

245 IF n>ftot%:n=ftot%

246 fptr%=n-1:frow%=(fptr% DIV 4)

247 IF frow%>(15+lptr%):**Fscr_up**:n=fptr%+1:**Fscr_posn**

248 IF frow%<(0+lptr%):**Fscr_dn**:n=fptr%+1:**Fscr_posn**

249 scr_row%=frow%-lptr%:scr_col%=(fptr% MOD 4)*18+1

250 **END DEFine**

252 **DEFine PROCEDURE Fscr_up**

253 lptr%=lptr%+1:SCROLL #7,-10:n=(lptr%+15)*4:scr_row%=15

254 FOR i=0 TO 3:scr_col%=i*18+1:n=n+1:**F_write**

255 **END DEFine**

257 **DEFine PROCEDURE Fscr_dn**

258 lptr%=lptr%-1:SCROLL #7,10:n=(lptr%)*4:scr_row%=0

259 FOR i=0 TO 3:scr_col%=i*18+1:n=n+1:**F_write**

260 **END DEFine**

262 **DEFine PROCEDURE F_write**

263 IF n>ftot% OR n<1:RETurn

264 flgth%=LEN(file\$(n)):IF flgth%-PLen%>15:flgth%=15+PLen%

265 INK #7,Sett(n):CURSOR #7,scr_col%*6,scr_row%*10

266 PRINT #7,file\$(n,1+PLen% TO flgth%)&FILL\$(' ',16+PLen%-flgth%)

267 CURSOR #5,px%,py%:PRINT #5,file\$(n):CLS #5,4

268 **END DEFine**

270 **DEFine PROCEDURE F_select**

271 REPEAT Sel_lp

272 **Fscr_posn**:Sett(n)=7:**F_write**:Sett(n)=4

273 k=CODE(INKEY\$(#5,50))

274 SElect ON k

275 =192:**Fscr_posn**:**F_write**:n=n-1 :REMark back 1

276 =200:**Fscr_posn**:**F_write**:n=n+1 :REMark forward 1

277 =208:**Fscr_posn**:**F_write**:n=n-4 :REMark up 1 row

278 =216:**Fscr_posn**:**F_write**:n=n+4 :REMark down 1 row

279 =209:**Fscr_posn**:**F_write**:n=n-60 :REMark up 1 page

280 =217:**Fscr_posn**:**F_write**:n=n+60 :REMark down 1 page

281 = 32:Sett(n)=mark%:**F_write**:n=n+1 :REMark mark filename

282 = 10:Sett(n)=7:RETurn

283 **END SElect**

284 **END REPEAT Sel_lp**

285 **END DEFine**

287 **DEFine PROCEDURE F_clear**

288 CLS #5:CURSOR #5,24,6:PRINT #5,' Please wait..'

289 FOR s=1 TO ftot%:Sett(s)=4

290 n1=(lptr%*4)+1:n2=(lptr%+16)*4:IF n2>ftot%:n2=ftot%

291 CLS #7:INK #5,0:px%=200:py%=5

292 FOR n=n1 TO n2:**Fscr_posn**:**F_write**

293 INK #5,7:CLS #5

294 **END DEFine**

QBFTidy 2017

296 **DEFine PROCedure F_copy**

```
297 CLS #5:mark%=7:px%=84:py%=6:CURSOR #5,24,6:PRINT#5,'Copy '&SDR$
298 F_select:flgth%=LEN(file$(n))
299 CURSOR #5,px%,py%:PRINT #5,file$(n)&' to '
300 px%=108+flgth%*6:py%=6:DDrv:TDR$=Dev$(Dnum)
301 IF TDR$=SDR$:F_clear:RETurn
302 CURSOR #5,px%,py%:PRINT #5,TDR$&' Overwrite <y/n>'
303 px%=84:py%=6:Fnum%=n:k$=INKEY$(#5,-1)
304 IF k$='y' OR k$=CHR$(10)
305 FOR n=1 TO ftot%
306 IF Sett(n)=7
307 CURSOR #5,px%,py%:PRINT #5,file$(n)&' TO ' &TDR$:CLS #5,4
308 DELETE TDR$&file$(n):COPY SDR$&file$(n) TO TDR$&file$(n)
309 INK #5,7:Sett(n)=0:Fscr_posn:F_write
310 END IF
311 END FOR n
312 ELSE
313 F_clear
314 END IF
315 n=Fnum%:SDR$=TDR$:F_dir
316 END DEFine
```

318 **DEFine PROCedure F_delete**

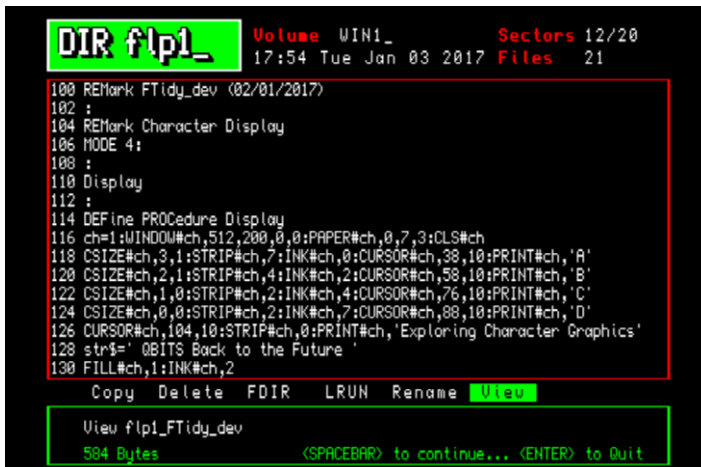
```
319 CLS #5:Fdel%=ftot%:px%=84:py%=6:mark%=7
320 CURSOR #5,12,py%:PRINT #5,'Delete '&SDR$:F_select:Fnum%=n
321 FOR n=1 TO ftot%
322 IF Sett(n)=7
323 CURSOR #5,px%,py%:PRINT #5,file$(n)&' (y/n) ':CLS #5,4
324 k$=INKEY$(#5,-1)
325 IF k$='y'
326 DELETE SDR$&file$(n):Fdel%=Fdel%-1
327 Sett(n)=0:Fscr_posn:F_write
328 END IF
329 Sett(n)=4:Fscr_posn:F_write
330 END IF
331 END FOR n
332 n=Fnum%:IF Fdel%=ftot%:RETurn :ELSE F_dir
333 END DEFine
```

335 **DEFine PROCedure F_lrun**

```
336 CLS #5:px%=84:py%=6:mark%=4:char%=0:cline%=0:count%=0
337 CURSOR #5,24,6:PRINT #5,'LRUN '&SDR$:F_select
338 CURSOR #5,px%,py%:PRINT #5,file$(n)&' (y/n) ':CLS #5,4
339 k$=INKEY$(#5,-1)
340 IF k$='y':LRUN SDR$&file$(n)
341 Sett(n)=4:Fscr_posn:F_write
342 END DEFine
```


QBFTidy 2017

```
344 DEFine PROCedure F_view
345 CLS #5:px%=84:py%=6:mark%=4:char%=0:cline%=0:count%=0
346 CURSOR #5,24,6:PRINT #5,'View '&SDR$:F_select
347 CURSOR #5,px%,py%:PRINT #5,file$(n)&' (y/n)':CLS #5,4
348 Fnum%=n:k$=INKEY$(#5,-1)
349 IF k$='y' OR k$=CHR$(10)
350 CURSOR #5,px%,py%:PRINT #5,file$(n):CLS #5,4
351 CURSOR #5,180,20 :INK #5,4
352 PRINT #5,'<SPACEBAR> to continue... <Enter> to Quit'
353 OPEN_IN #4,SDR$&file$(n):CLS #7
354 REPEAT View_lp
355 k$=INKEY$(#4,-1):count%=count%+1:char%=char%+1
356 IF EOF(#4)
357 CLOSE #4:CURSOR #5,24,20:PRINT #5,count%,' Bytes [end]':CLS #5,4
358 PAUSE:EXIT View_lp
359 END IF
360 IF char%>=74 AND k$<>CHR$(10):char%=0:cline%=cline%+1:END IF
361 IF k$=CHR$(10):char%=0:cline%=cline%+1:END IF
362 IF cline%>15
363 CURSOR #5,24,20:PRINT #5,count%,' Bytes'
364 cline%=0:an$=INKEY$(#5,-1)
365 IF an$=CHR$(10):CLOSE #4:EXIT View_lp:END IF
366 END IF
367 PRINT #7,k$;
368 END REPEAT View_lp
369 INK #5,7:F_clear:n=Fnum%
370 END IF
371 Sett(n)=4:Fscr_posn:F_write
372 END DEFine
```



```
DIR flp1_ Volume WIN1_ Sectors 12/20
17:54 Tue Jan 03 2017 Files 21

100 REMark FTidy_dev (02/01/2017)
102 :
104 REMark Character Display
106 MODE 4:
108 :
110 Display
112 :
114 DEFine PROCedure Display
116 ch=1:WINDOW#ch,512,200,0,0:PAPER#ch,0,7,3:CLS#ch
118 CSIZE#ch,3,1:STRIP#ch,7:INK#ch,0:CURSOR#ch,30,10:PRINT#ch,'A'
120 CSIZE#ch,2,1:STRIP#ch,4:INK#ch,2:CURSOR#ch,50,10:PRINT#ch,'B'
122 CSIZE#ch,1,0:STRIP#ch,2:INK#ch,4:CURSOR#ch,70,10:PRINT#ch,'C'
124 CSIZE#ch,0,0:STRIP#ch,2:INK#ch,7:CURSOR#ch,80,10:PRINT#ch,'D'
126 CURSOR#ch,104,10:STRIP#ch,0:PRINT#ch,'Exploring Character Graphics'
128 str1=' 0BITS Back to the Future '
130 FILL#ch,1:INK#ch,2

Copy Delete FDIR LRUN Rename View
View flp1_FTidy_dev
584 Bytes (SPACEBAR) to continue... (ENTER) to Quit
```

QBFTidy 2017

```
374 DEFine PROCEDURE F_rename
375 CLS #5:px%=84:py%=6:cmax%=32:mark%=4
376 CURSOR #5,12,6:PRINT #5,'Rename '&SDR$:F_select
377 CURSOR #5,px%,py%:PRINT #5,file$(n)&' (y/n)':CLS #5,4
378 k$=INKEY$(#5,-1)
379 IF k$='y' OR k$=CHR$(10)
380   str$=file$(n):char%=LEN(str$):F_edlin
381   IF str$=":GO TO 388:END IF
382   FOR n1=1 TO ftot%
383     IF str$=file$(n1):GO TO 388:END IF
384   END FOR n1
385   COPY SDR&&file$(n) TO SDR$(1 TO 5)&str$
386   DELETE SDR&&file$(n):F_dir
387 END IF
388 Sett(n)=4:Fscr_posn:F_write:RETurn
389 END DEFine
```

```
391 DEFine PROCEDURE F_edlin
392 INK #5,4:CURSOR #5,200,20
393 PRINT #5,'<␣ > Delete character : <Enter> to Quit'
394 REPEAT ed_lp
395 INK #5,2:CURSOR #5,px%,py%:PRINT #5,str$:CLS #5,4
396 k$=INKEY$(#5,-1):k=CODE(k$)
397 SElect ON k
398   =48 TO 57,65 TO 90,95,97 TO 122
399     IF char%<cmax%:char%=char%+1:str$=str$&k$:END IF
400   =192:IF char%>=1:char%=char%-1:END IF
401     IF char%=0:str$=":ELSE str$=str$(1 TO char%):END IF
402   =10:INK #5,7:CURSOR #5,170,20:CLS #5,4:RETurn
403 END SElect
404 END REPEAT ed_lp
405 END DEFine
```

```
DIR f:\p1_ Volume WIN1_ Sectors 12/20
18:28 Tue Jan 03 2017 Files 20
boot FLlist Radar QBFTidy
QBGolf_data QBWordGen Storeman_Sam QBWH2K
Wrd_GenWord1 Wrd_Country1 Wrd_Early_G1 Wrd_BETA
Wrd_GAMMA Wrd_OEMO Wrd_SIZES Wrd_DELTA
QBGolf_2016_12 QBWord_Gen QBConundrum FTidy_dev

Copy Delete FDIR LRUN Rename View
Rename f:\p1_FTidy_dev
<+> Delete character : <Enter> to Quit
```

QBFTidy

Having obtained a copy of **QBFTidy** SuperBASIC code and loaded it into a recognised QL device. Use the QDOS command LRUN, as shown:-

LRUN flp1_QBFTidy.

All being well you will find this a useful tool for your file management...

Notes on QL2K emulator

Both the **QLAY & QL2K emulators** use an application tool to create a QDOS directory file and append or delete files in it. Creating a new qlay.dir file first open a Windows **Command Prompt** (Win 7 Press Start button in *search programs and files* box type **command prompt**: Win 10 in *ask me anything* box type **command prompt**.)

Activate the command prompt window then navigate with DOS commands to the drive and Windows File Directory folder that holds your QL Files.

i.e C:\>**chdir** H:\QL\FDIR\WIN1_ H:\QL\FDIR\WIN1_>dir

This will list the files as a DOS directory. This needs to also contain a copy of **QLAYT-86.EXE** or **QLAY-X64.EXE** downloaded with **QLAY** or **QL2K**

At the DOS prompt now enter this command: -

i.e. H:\QL\FDIR\WIN1_>**qlayt-x64.exe -c qlay.dir**

This should create a directory file qlay.dir to which you can now append files. For example:-

i.e. H:\QL\FDIR\WIN1_>**qlayt-x64.exe -i Boot**

This will append the File named '**Boot**' to the qlay.dir .

Once you have appended your files you can use the following command to list them:-

i.e. H:\QL\FDIR\WIN1_> **qlayt-x64.exe -l**

A list of files should now be shown contained within the qlay.dir



```
Command Prompt
H:\QL\FDIR\WIN1_>qlayt-x64.exe -l
boot
FList
Golf
QBFTidy
QBGo lf
QBGo lf_04_12_16
QBGo lf_data
Found 7 files in directory qlay.dir
H:\QL\FDIR\WIN1_>
```

To remove a file at the DOS prompt type:-

i.e. H:\QL\FDIR\WIN1_>**qlayt-x64.exe -r Golf**

This will remove the File named '**Golf**' from the qlay.dir .

Running the **QL2K emulator** the files listed in WIN1_ should now be readable by the QDOS DIR command; however these files will not be loadable or run if not compatible with the QDOS operating system you have. This even applies to QL software that does not work with or only works with certain versions of QL ROM's or with added Toolkit extensions.

```
QLZK
QL Commands Screen Language Help

DIR flp1_ Volume WIN1_ Sectors 12/20
18:06 Wed Jan 04 2017 Files 20

FList Boot QBFTidy QBGolf_data
Boot_2016 QBWordGEen Storeman_Sam QBWH2K
Wrd_GenWord1 Wrd_Country1 Wrd_Early_G1 Wrd_BETA
Wrd_GAMMA Wrd_DEMO Wrd_SIZES Wrd_DELTA
QBGolf_2016_12 QBWord_Gen QBConundrum FTidy_dev

Copy Delete FDIR LRUN Rename View

Copy flp1_FTidy_dev to flp2_ Overwrite <y/n>
```

```
QLZK
QL Commands Screen Language Help

DIR flp1_ Volume WIN1_ Sectors 12/20
18:07 Wed Jan 04 2017 Files 20

FList Boot QBFTidy QBGolf_data
Boot_2016 QBWordGEen Storeman_Sam QBWH2K
Wrd_GenWord1 Wrd_Country1 Wrd_Early_G1 Wrd_BETA
Wrd_GAMMA Wrd_DEMO Wrd_SIZES Wrd_DELTA
QBGolf_2016_12 QBWord_Gen QBConundrum FTidy_dev

Copy Delete FDIR LRUN Rename View

Delete flp1_Boot_2016 <y/n>
```