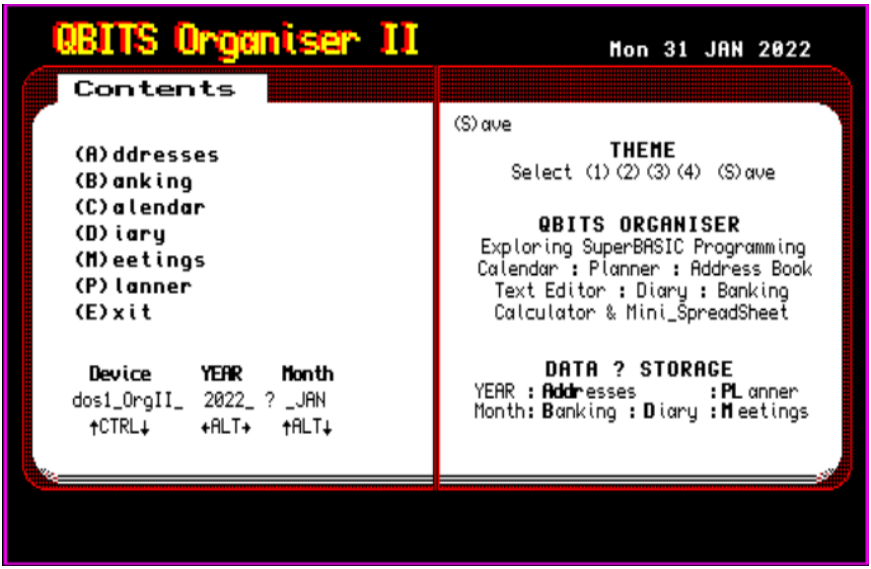
Sinclair QL Retro Computing



Sinclair QL Retro Computing

QPC11 Emulator

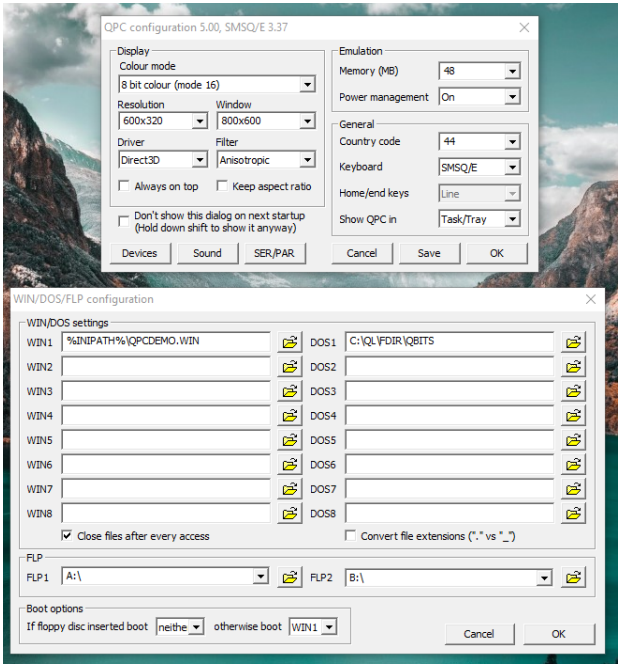
Installed and run on a Windows PC this Emulates a Sinclair QL Computer. However, it has a far more advanced O/S with Tony Tebby's SMSQ/E the successor to his QDOS and an updated expanded SBASIC to the QL SuperBASIC of Jan Jones day.

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: -



QBITS Progs

Download 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_QBITS_Boot**

The QBITS Progs Menu should now be displayed
Select a Program with Cursor Keys and Spacebar.

QBITS introduction

The release of QPCII a few years back helped in reviving my interest in SuperBASIC programming. QBITS 1980's Organiser was resurrected and I made a commitment to Review and update the code. This was seen as an opportunity to unify common areas, make it a little more robust and less complicated to handle. The performance issues and memory capacity of the QL's humble beginnings largely overcome by later versions of QL Hardware and with Software Emulators running on modern computer platforms.

So here goes?

QBITS Organiser Beginnings

This began its journey in the eighties with Progs for a Calendar and Year Planner. A Contacts list followed that utilised a simple Line Editor. An appointments page was added and a Diary Page with the continued development of the Editor to work within varying columns and rows of a Text Box. Then a Graphics display for a Calculator led to a modified Editor to work with Maths functions.

QBITS Organiser Code Review

The Contents Page provides a Menu access to each of the Organiser Sections. Apart from Calendar each Page creates information that is held in Arrays. The Saving of data is made to the default storage Device. **Organiser II** when entering a Section automatically Searches and Loads any previously Saved DATA from the current default Device. On leaving a Section Page the program prompts a Save of any DATA generated.

The Arrays holding DATA for each Organiser Section Saves in different Formats. The default Device and range of Devices is obtained from **QBITSConfig**. Following the Device name is a Prefix for the Directory Filename [DF$]. The QBITS default is **'OrgII_**', this can be changed to the users preferred file heading (see Code Line 1042).

All Sections use a Suffix of the '**Year_**'. To this is then added the Type, '**Addr**' Addresses, '**PL** Planner. '**B_**' Banking, '**D_**' Diary, '**M_**' Meetings. Appended to Banking, Diary & Meetings DATA files is the Month for example: **'JAN**'.

The default **Device**, **Year** & **Month** can be changed on the **Contents** Page.

To the righthand side of Selected Pages the Working Date is displayed. This is separate from the Current Date shown above the Organiser Page. Banking, Diary and Meetings use Left/Right Cursor keys to change the Working Day and displayed information. In addition, the Planner use Up/Down Cursors to change the Month. Calendar shows only the Year. Addresses instead of a Date displays the current record which can be changed with Left/Right Cursors. To change the Search Name Character, use the Up/Down Cursor keys.

QBITS Calendar Calculations
Calendar was one of the first Progs to be written. Using the keyword DAY$ to obtain the Weekday from a Date was found to be limiting. To cope with the expanded QBITS Year range the Gregorian Calendar and Key Value Method is used to construct a SuperBASIC PROCedure to calculate the Weekday.

See **PROCedure Cal_day (yr%,m%,d%)**.

It takes the last two digits of the year divides by 4 and drops the remainder, to which it then adds the day of the month and a key value for the month.

Jan	Feb	Mar	Apr	May	June	July	Aug	Sept	Oct	Nov	Dec
1	4	4	0	2	5	0	3	6	1	4	6

A Leap Year is one divisible by 4, but not if evenly divisible by 100 i.e. 1700,1800,1900, only centuries divisible by 400 are Leap Years. If JAN or FEB is in a Leap Year 1 is subtracted then a century code added.

1700s	1800s	1900s	2000s
4	2	0	6

To obtain codes for other centuries luckily the Gregorian Calendar repeats every 400 years. For example, 1650+400=2050 so code is 6.

Next, we again add the last two digits of the year, divide by 7 and use the remainder. The value being the day of the week, 0-Saturday, 1-Sunday, 2-Monday and so on.

QBITS TextED & Calculator
SuperBASIC uses INPUT for entering information into a program which uses its built in Line Editor. For QBITS Progs this did not allow control over variable ASCII Character sets. Written to overcome this technical detail **QBITS TextED** began as a simple Line Editor but evolved into a Text Box Editor with functions for multiple rows and variable columns. For the Calculator a separate Editor was written to handle the Maths Functions.

QBITS Mini-Spreadsheet
The Calculator can be found on the Banking Page. It can be used independently or as a ready reckoner linked to Cells of a three column mini-spreadsheet. The first column uses **TextED** to provide a heading/statement the two following used the Calculator Editor to enter monetary sums with two decimal places for cents/pennies.

QBITS File Handling
WHEN ERROR check (erk) Flags a ‘File NOT Found’, or ‘DEVICE ERROR’.

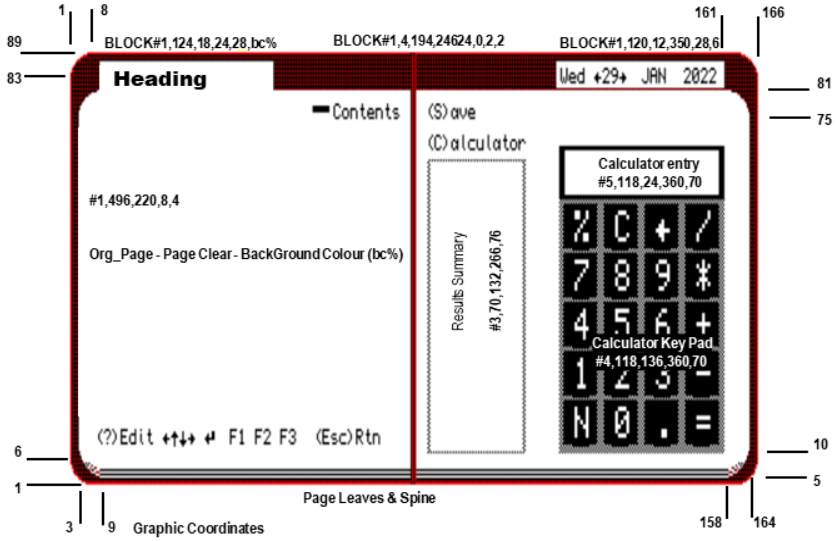
QBITSProgs
Organiser II code has been rewritten as one of a suite of QBITS SuperBASIC Progs adapted for the QPCII Emulator. It utilises common variables accessed from QBITSConfig. For Organiser II to be used independently Load the copy of **QBITSConfig** into ram2_ then COPY to and LRUN **QBIT_OrganiserII_v1** from selected storage device.

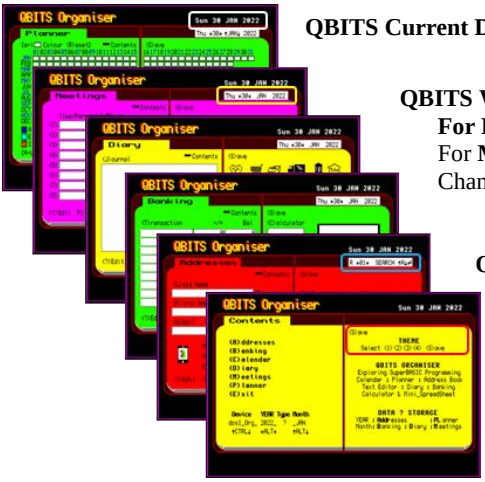
QBITS Organiser II Code 2022

```
1000 REMark QBITS_OrgII_v1 (QBITS Organiser – 2022 QPCII)

1002 OPEN_IN#9,'ram2_QBITSConfig':INPUT#9,gx\gy\dn$\dev$\dn%\dmax%      To run Organiser Copy
1003 DIM drv$(15,5):FOR d=0 TO 15:INPUT#9,drv$(d):END FOR d:CLOSE#9     QBITSConfig to ram2_

1005 REMark Appointments/Banking/Calendar/Diary/Mail List/Planner
1006 REMark TexED/Calculator/OrgII_Data File & Config

1008 WHEN ERRor
1009    eck=1:CONTINUE
1010 END WHEN

1012 REMark Arrays
1013 DIM phone$(50,6,16),name$(50,5,26),addr$(50,96),email$(50,36)   :REMark Addr
1014 DIM Tran$(31,9,3,16),Key$(4,5),Num$(9):num=0                    :REMark Banking
1015 DIM Jrnl$(31,432),Rmdr$(31,216),Wth%(31,5),CRem(31,6)           :REMark Diary
1016 DIM Meet$(31,8,32),Note$(31,256)                                :REMark Meets
1017 DIM Plnr%(12,31),Event$(12,12),Pcol%(12)                        :REMark Planner
1018 DIM wd$(7,3),mth$(12,3),dm%(12),bcol%(6),Sky$(8,10),RStr$(15)

1020                    :FOR i= 0 TO   6:bcol%(i)=7                  :REMark BkGnd Colours
1021 RESTORE 1028:FOR i= 1 TO   7:READ wd$(i)                        :REMark Days of the Week
1022 RESTORE 1029:FOR i= 1 TO 12:READ mth$(i),dm%(i)                 :REMark Month / Days-Month
1023 RESTORE 1032:FOR i= 0 TO   8:READ Sky$(i)                       :REMark Weather Patterns
1024 RESTORE 1035:FOR i= 1 TO 12:READ Event$(i),Pcol%(i)             :REMark Planner
1025 RESTORE 1039:FOR b=1 TO   5:FOR a=1 TO  4:READ Key$(a,b):END FOR a:END FOR b

1027 REMark Days of the week : Month : End of Month
1028 DATA "Sun","Mon",'Tue','Wed','Thu','Fri','Sat'
1029 DATA 'JAN',31,'FEB',28,'MAR',31,'APR',30,'MAY',31,'JUN',30
1030 DATA 'JUL',31,'AUG',31,'SEP',30,'OCT',31,'NOV',30,'DEC',31
1031 REMark Weather Patterns
1032 DATA 'Changeable','Sunny','Cloudy','Windy','Showers'
1033 DATA 'Overcast','Heavy Rain','Stormy','Snow'
1034 REMark Planner Events : Colour Codes
1035 DATA 'Anniversary',1,'Birthday',2,'Doctor',3,'Dentice',4
1036 DATA 'Shopping',5,'Meetings',6,'PayDays',41,'Holidays',48
1037 DATA 'Project 1',34,'Project 2',80,'Project 3',230,'Project 4',255
1038 REMark Calculator Keyboard
1039 DATA '%','C','←','/', '7','8','9','*', '4','5','6','+'
1040 DATA '1','2','3','-', 'N','0','.','='
1041 REMark Init Program
1042 yb%=1901:ye%=2099:DF$='OrgII_'          :REMark Year Range :Data File Prefix
1043 ct%=65:bc%=7 :rm%=50                    :REMark ct% = A : bc% BkGnd : Max Rcrds

1045 MODE 4:Init_Win:Init_Date:Org_Menu                    Initiate and Start the Program

1047 DEFine PROCedure Org_Exit
1048 STRIP bc%:INK 0:CURSOR 92,132:PRINT 'Y/N':PAUSE
1049 IF KEYROW(5)=64:LRUN dn$:ELSE BLOCK 20,10,92,144,bc%
1050 END DEFine
```

QBITS Organiser Screen Layouts

SuperBASIC WINDOW channel [#6] acts as a background. WINDOW#1 the main operating channel is set to SCALE 100,0,0. Setting x y values for drawing the Organiser Cover and Pages with rounded corners use ARC & LINE and the Graphics coordinates system. Next was location for the Page Heading and control bar for changing displayed information such Working Date or Record etc.





QBITS Current Date Generated from QL Clock.

QBITS Working Dates

For Planner change **Day** & **Month.**
For **Meetings, Diary, Banking**
Change the **Day** of the Month.

QBITS Address Records

QBITS Organiser Themes
Default Page colour is White.
Page Themes can be Selected and Saved from Contents Page.

QBITS Organiser II Control Keys

Cursor Left/Right/Up/Down <**Spacebar**> <**Enter**>

← → ↑ ↓ ▬ ↵

Edit (**?**) [key First Letter of (**H**)eading…] (**Esc**) to Return

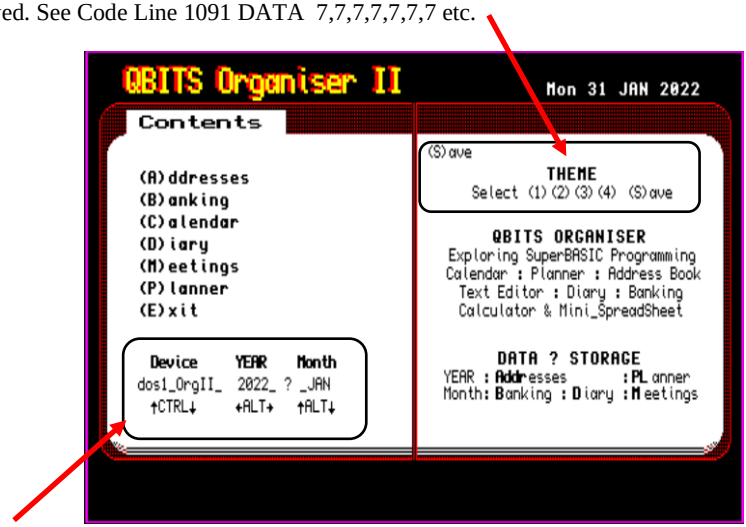
```
1052 DEFine PROCedure Init_Win
1053 LOCal a,b,c,d,e,f
1054 OPEN#6,scr_:WINDOW#6,512,256,gx,gy:BORDER#6,1,3:PAPER#6,0:CLS#6
1055 OPEN#5,scr_:WINDOW#5,118, 24,360+gx,70+gy          :REMark QB Calc Entry
1056 OPEN#4,scr_:WINDOW#4,118,136,360+gx,72+gy          :REMark QB Calc Keys
1057 OPEN#3,scr_:WINDOW#3,70,132,266+gx,76+gy           :REMark QB Calc Results
1058 WINDOW#2,496,220,gx+8,gy+4  :PAPER#2,0:INK#2,7     :REMark SBASIC EDitor
1059 WINDOW#1,496,220,gx+8,gy+4  :PAPER#2,0:INK#2,7     :REMark QB Org Page
1060 WINDOW#0,496, 20,gx+8,gy+226:PAPER#2,0:INK#2,7     :REMark SBASIC Command
1061 ch%=1:SCALE#ch%,100,0,0:chk=0
1062 FOR chk=0 TO 1
1063  IF chk=0:FILL#ch%,1:INK#ch%,16:RESTORE 1067
1064  IF chk=1:FILL#ch%,0:INK#ch%,  2:RESTORE 1067
1065  FOR i=1 TO 4:READ a,b,c,d,e,f:LINE#ch%,a,b TO c,d:ARC#ch% TO e,f,-PI/2
1066 END FOR chk
1067 DATA 6,89,161,89,166,83,166,83,166,6,161,1
1068 DATA 161,1,6,1,1,6,1,6,1,83,6,89
1069 REMark Page Leaves
1070 RESTORE 1072:BLOCK#ch%,446,5,24,210,7,0,1:INK#ch%,7
1071 FOR i=1 TO 6:READ a,b,c,d:ARC#ch%,a,b TO c,d,PI/2
1072 DATA 3,10,9,2.1,3,10,9,3,3,10,9,4,158,2.1,164,10,158,3,164,10,158,4,164,10
1073 REMark Title
1074 ch%=1:QBIT$='QBITS Organiser':OVER#ch%,1:CSIZE#ch%,2,1
1075 INK#ch%,2:FOR i=0 TO 1:CURSOR#ch%,20-i,2:PRINT#ch%,QBIT$
1076 INK#ch%,6:FOR i=0 TO 1:CURSOR#ch%,22+i,3:PRINT#ch%,QBIT$
1077 OVER#ch%,0:CSIZE#ch%,0,0
1078 END DEFine

1080 DEFine PROCedure Init_Date                    Current Date from QL Clock
1081 ch%=1:d$=DAY$(DATE):dmy$=DATE$:yr%=dmy$(1 to 4)
1082 FOR i=1 TO 12:IF dmy$(6 TO 8)==mth$(i):m%=i:EXIT i
1083 d%=dmy$(10 TO 11):IF d%<10:dy$='0'&d%
1084 OVER#ch%,1:CSIZE#ch%,1,0:INK#ch%,7
1085 FOR i=0 TO 1:CURSOR#ch%,350+i,12:PRINT#ch%,d$;' ';dy$;' ';mth$(m%);' ';yr%
1086 OVER#ch%,0:CSIZE#ch%,0,0:INK#ch%,0
1087 END DEFine

1089 DEFine PROCedure Org_Theme                    Page BkGnd Colours
1090 RESTORE 1091+(k-49):FOR i=0 TO 6:READ bcol%(i)
1091 DATA 7,7,7,7,7,7,7          :REMark Default BkGnd bc%
1092 DATA 6,2,4,7,6,3,4          :REMark BkGnds Red Green Cyan Yellow Magenta
1093 DATA 5,3,6,7,5,4,5          :REMark INK Black Blue
1094 DATA 4,6,5,7,4,5,6
1095 END DEFine

1097 DEFine PROCedure Org_Page(ch%,bc%)            Page Clear
1098 RESTORE 1101:FILL#ch%,1:INK#ch%,bc%
1099 FOR i=1 TO 4:READ a,b,c,d,e,f:LINE#ch%,a,b TO c,d:ARC#ch% TO e,f,-PI/1.5
1100 FILL#ch%,0:STRIP#ch%,bc%:INK#ch%,0
1101 DATA 10,81,156,81,164,75,164,75,164,10,158,5,158,5,10,5,3,10,3,10,3,75,10,81
1102 BLOCK#ch%,124,18,24,28,bc% :BLOCK#ch%,120,12,350,28,16
1103 BLOCK#ch%,4,194,246,24,0,2,2
1104 END DEFine
```

QBITS Organiser Contents

Init_Date takes the Current Date from the QL Clock and sets the **Year** & **Month**. Initialising the Organiser Loads **QrgII_Config** or displays 'File NOT Found' if the latter Page Backgrounds remain default white.

QBITS Organiser Themes

Select (1) (2) (3) (4) and (**S**)ave **Theme**. The Config File details the background colours to be deployed. See Code Line 1091 DATA 7,7,7,7,7,7,7 etc.



QBITS Organiser Settings

CTRL UP/Down cursor keys change the Default **Device.** Check with a **Theme** (S)ave. If Device not valid a **DEVICE ERROR** will be given.

ALT Left/Right cursor keys change the **Year**. All Data files are suffixed with this.

ALT Up/Down cursor keys change the **Month**. Banking, Diary, and Meetings are appended by this, make sure regular backup are carried out.

QBITS Organiser Load/Save

The Organiser **Year/Month** DATA Files are Searched for when entering the relevant **Organiser Page**. If found they are Loaded. When leaving an Organiser Page, or by pressing '**S**' a (**S**)ave Y/N prompt is displayed. If answer is 'Yy' (Yes) the DATA File will be Saved or a prompt to Overwrite Y/N (if already exists) or 'DEVIC ERROR' (if Device is not accessible).

QBITS DATA File Format

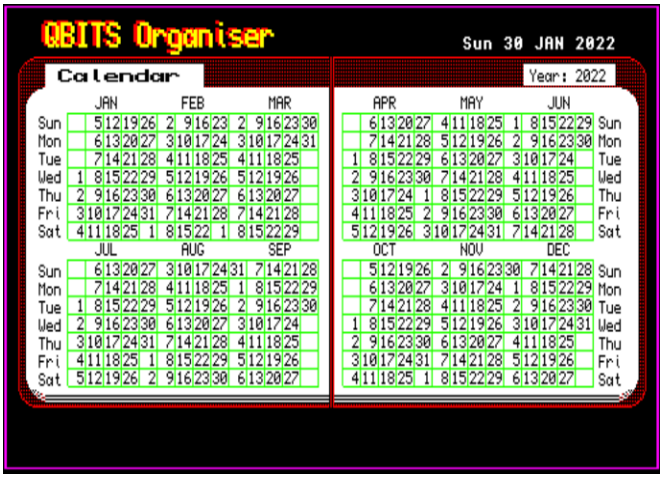
Ex. dos1_OrgII_2022_D_JAN Diary File
Ex: win1_OrgII_2021_Addr Address List File

```
1106 DEFine PROCedure Org_Menu
1107 ch%=1:Org_Page 1,7:Prn_Bold 1,256,60,'Initialising...'
1108 BLOCK 120,12,350,28,16:OLoad 0,'OrgConfig':Con=1
1109 REPeat Page_lp
1110   IF Con=1:Con=0:bc%=bcol%(0):Org_Page 1,bc%:Content_Page:FileCLS
1111   CURSOR#ch%,32,172:PRINT#ch%,drv$(dn%)&DF$&'  '&yr%&'_  ?  _'&mth$(m%)
1112   k=CODE(INKEY$(-1))
1113   SELect ON k
1114     =65,  97:Con=1:bc%=bcol%(1):Org_Page 1,bc%:Addresses
1115     =66,  98:Con=1:bc%=bcol%(2):Org_Page 1,bc%:Banking
1116     =67,  99:Con=1:bc%=bcol%(3):Org_Page 1,bc%:Calendar
1117     =68,100:Con=1:bc%=bcol%(4):Org_Page 1,bc%:Diary
1118     =77,109:Con=1:bc%=bcol%(5):Org_Page 1,bc%:Meetings
1119     =80,112:Con=1:bc%=bcol%(6):Org_Page 1,bc%:Planner
1120     =69,101:Org_Exit
1121     =210:dn%=dn%+1:IF dn%>dm%:dn%=dm%          :REMark Ctrl Up  Device
1122     =218:dn%=dn% -1:IF dn%<0  :dn%=0           :REMark Ctrl Down
1123     =193:yr% =yr%  -1:IF yr%<yb%:yr%=yb%       :REMark Alt Left   Year
1124     =201:yr% =yr% +1:IF yr%>ye%:yr%=ye%        :REMark Alt Right Year
1125     =209:m% =m%  -1:IF m%< 1:m%= 1             :REMark Alt Up     Month
1126     =217:m% =m% +1:IF m%>12:m%=12              :REMark Alt Down Month
1127     =49,50,51,52  :Con=1:Org_Theme             :REMark Select Theme
1128     =83,115  :OSave 0,DF$&'Config':FileCLS
1129   END SELect
1130 END REPeat Page_lp
1131 END DEFine

1133 DEFine PROCedure Prn_Bold(ps,px,py,str$)          Overwrite in various Font sizes
1134 CSIZE ps,0:OVER 1
1135 FOR b=0 TO 1:CURSOR px+b,py:PRINT str$
1136 CSIZE  0,0:OVER 0
1137 END DEFine

1139 DEFine PROCedure Content_Page
1140 RESTORE 1141FOR i=1 TO 8:READ px,py,hr$:CURSOR px,py:PRINT hr$
1141 DATA 280, 68,'Select (1)(2)(3)(4) (S)ave'
1142 DATA 274,106,'Exploring SuperBASIC Programming'
1143 DATA 272,116,'Calendar:Year Planner:Text Editor'`
1144 DATA 278,126,'Address Book : Diary : Banking'
1145 DATA 282,136,'Mini_SpreadSheet & Calculator'
1146 DATA  38,184,'↑CTRL↓   ←ALT→   ↑ALT↓ '
1147 DATA 270,168,'YEAR  :    esses      :  anner'
1148 DATA 270,177,'Month :  anking :  iary :  eetings'
1149:FOR i=1 TO 12:READ ps,px,py,str$:Prn_Bold ps,px,py,str$
1150 DATA 2,32,30,'Contents',1,32,60,'(A)ddresses',1,32,72,'(B)anking'
1151 DATA 1,32,84,'(C)alendar',1,32,96,'(D)iary',1,32,108,'(M)eetings'
1152 DATA 1,32,120,'(P)lanner',1,32,132,'(E)xit',1,350,58,'THEME'
1153 DATA 0,38,160,'Device   YEAR Type Month'
1154 DATA 1,308,96,'QBITS ORGANISER',1,322,158,'DATA STORAGE'
1155 DATA 0,310,168,'Addr',0,417,168,'PL'
1156 DATA 0,310,177,'B',0,370,177,'D',0417,177,'M'
1157 END DEFine
```

QBITS Organiser Calendar
Set **Year** on **Contents** Page before accessing **Calendar** Page. Press any key to Return.

The Calendar Page is divided into four quarterly sections with the Months and Days of a 13-week period. These are spread over 14 columns to cater for different start and end weekdays. The **Days** of the **week** are calculated relevant to the **YEAR** selected.

```
1159 REMark Organiser Calendar Page    Displays Year in Months and Weekdays

1161 DEFine PROCedure Calendar
1162 ch%=1:Prn_Bold 2,32,30,'Calendar'
1163 BLOCK#ch%,72,12,398,28,7:Calendar_Page:Cal_Year yr%
1164 STRIP#ch%,7:CURSOR#ch%,402,30:PRINT#ch%,'Year: ';yr%:PAUSE
1165 END DEFine
```




```
1167 DEFine PROCedure Cal_Year(yr%)
1168 ch%=1:d%=1:wk%=1:mn%=1:x%=28:y%=46:cw%=1                    :REMark 1st Quarter
1169 Cal_Day yr%,1,1:IF yr% MOD 100=0:dm%(2)=28                  :REMark 1st Jan :Weekday wkd%
1170 CSIZE#ch%,0,0:INK#ch%,0:OVER#ch%,1:IF yr% MOD 400=0:dm%(2)=29
1171 REPeat day_lp
1172   IF cw%=1:cw%=0:FOR day=1 TO 7:BLOCK#ch%,13,9,x%+wk%*14,y%+day*10,7
1173   CURSOR#ch%,x%+wk%*14,y%+wd%*10:PRINT#ch%,FILL$(' ',2-LEN(d%))&d%
1174   wd%=wd%+1:d%=d%+1:IF wd%>7:wd%=1:wk%=wk%+1:cw%=1   :REMark cw% Wk CLS
1175   IF wk%=13 AND d%=31:FOR day=1 TO 7:BLOCK#ch%,13,9,x%+14*14,y%+day*10,7
1176   IF d%>dm%(mn%):d%=1:mn%=mn%+1                             :REMark End of Month
1177   IF mn%= 4 AND d%=1:wk%=15:cw%=1:x%=  47:y%= 46            :REMark 2nd Quarter
1178   IF mn%= 7 AND d%=1:wk%=29:cw%=1:x%=-364:y%=127            :REMark 3rd Quarter
1179   IF mn%=10 AND d%=1:wk%=43:cw%=1:x%=-345:y%=127            :REMark 4th Quarter
1180   IF mn%>12:OVER#ch%,0:EXIT day_lp                          :REMark Exit
1181 END REPeat day_lp
1182 END DEFine
```

The Calendar Cells are drawn using BLOCK and the Pixel coordinates system. The chosen WINDOW size 496x220 reduces the row Cell height for Characters to 8 Pixels. To overcome this restriction the Cell is Cleared with a BLOCK command and PRINT is used with the OVER command to write the Days of the Month.

Font 6x10 Pixels 8 Pixels 23 10 Pixels

Cal_Year determines Cell BLOCK and CURSOR positioning then PRINTs the Date. The day [**d%**] and weekday [**wd%]** increments on each pass. If [**wd%**] reaches **7** it is reset to **1**, the week [**wk%**] is then incremented and clear week [**cw%**] is reset. The day [**d%**] is checked against the month [**dm%(m%)**]. If last day reached this sets the month day [**d%**] back to 1 and increments the month [**m%]**.

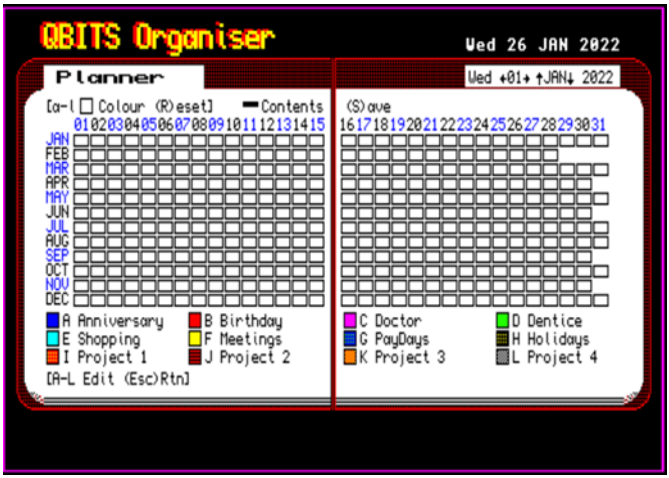
Quarters 2nd, 3rd & 4th checks, resets x%,y% coordinates for each Cell group. In certain years, the occasion arises when Jan 1st is on a Sunday and March 31st falls in week 13 for this an additional Clear of Cells in week 14 is made.

C**al_day** (Year, Month, Day) calculates which Weekday a specified Date falls on.

```
1184 DEFine PROCedure Cal_Day(yr%,m%,d%)
1185 IF yr% MOD 4=0:dm%(2)=29:ELSE dm%(2)=28                    :REMark Leap yr% check
1186 y$=yr%:yd%=y$(3 TO 4):year=yr%                             :REMark yr% & yd% values
1187 SELect ON year
1188   =1700 TO 1799:yc%=4                                      :REMark yc% Century key value
1189   =1800 TO 1899:yc%=2
1190   =1900 TO 1999:yc%=0
1191   =2000 TO 2099:yc%=6
1192 END SELect
1193 RESTORE 1218:FOR i=1 TO 12:READ a :IF m%=i:mk%=a:EXIT i
1194 DATA 1,4,4,0,2,5,0,3,6,1,4,6                               :REMark mk% Month key value
1195 IF dm%(2)=29:IF m%=1 OR m%=2:mk%=mk%-1:END IF              :REMark Leap Year adjustment
1196 wd%=((yd% DIV 4)+d%+mk%+yc%+yd%) MOD 7:IF wd%=0:wd%=7:REMark wd% Week Day
1197 END DEFine

1199 DEFine PROCedure Calendar_Page                             Draws the Quarterly Grids
1200 FOR i=1 TO  8:BLOCK#ch%,196,1,42,45+i*10,4
1201 FOR i=1 TO 15:BLOCK#ch%,1,70,27+i*14,55,4
1202 FOR i=1 TO  7:CURSOR#ch%,18,46+i*10  :PRINT#ch%,wd$(i)
1203 FOR i=1 TO  3:CURSOR#ch%,i*68-6,45   :PRINT#ch%,mth$(i)
1204 FOR i=1 TO  8:BLOCK#ch%,196,1,257,45+i*10,4
1205 FOR i=1 TO 15:BLOCK#ch%,1,70,242+i*14,55,4
1206 FOR i=1 TO  7:CURSOR#ch%,456,46+i*10 :PRINT#ch%,wd$(i)
1207 FOR i=1 TO  3:CURSOR#ch%,i*68+212,45 :PRINT#ch%,mth$(i+3)
1208 FOR i=1 TO  8:BLOCK#ch%,196,1,42,126+i*10,4
1209 FOR i=1 TO 15:BLOCK#ch%,1,70,27+i*14,136,4
1210 FOR i=1 TO  7:CURSOR#ch%,18,128+i*10 :PRINT#ch%,wd$(i)
1211 FOR i=1 TO  3:CURSOR#ch%,i*68-6,126  :PRINT#ch%,mth$(i+6)
1212 FOR i=1 TO  8:BLOCK#ch%,196,1,257,126+i*10,4
1213 FOR i=1 TO 15:BLOCK#ch%,1,70,242+i*14,136,4
1214 FOR i=1 TO  7:CURSOR#ch%,456,128+i*10:PRINT#ch%,wd$(i)
1215 FOR i=1 TO  3:CURSOR#ch%,i*68+212,126:PRINT#ch%,mth$(i+9)
1216 END DEFine
```

QBITS Organiser Planner

Select **Year** on **Contents** Page. Entering **Planner** will Load DATA file if previously Saved and display entries with **Months** in rows and **Days** in columns. Use the cursor keys to select a specific Date (Day/Month). The Cell can be now colour coded [a-l]. The Original QL had a limited number of solid colours, so Stipple patterning is used to extend the range. The Event Names can be Edited [A-L]. The Event Names are (S)aved as part of the Planner DATA File.




```
1218 REMark Organiser Planner Page      Events Chart – Days of Year Colour Coded

1220 DEFine PROCedure Planner
1221 ch%=1: Prn_Bold 2,32,30,'Planner': OLoad 6,'Org_'&yr%&'_PL':FileCLS
1222 BLOCK#ch%,120,12,350,28,7:Planner_Page:Cal_Day yr%,m%,d%:Chart_Ent
1223 REPeat Plan_lp
1224  Chg_Date:k=CODE(INKEY$(-1))
1225  SELect ON k
1226   =32:OSave 6,'Org_'&yr%&'_PL':EXIT Plan_lp
1227   =83,115:OSave 6,'Org_'&yr%&'_PL':FileCLS
1228   =97 TO 108:Plnr%(m%,d%)=k-96:Chg_Col          :REMark (a - l) Colour Code
1229   = 82,114     :Plnr%(m%,d%)=0     :Chg_Col      :REMark (R)eset Colour Code
1230   =65 TO  76 :chk=2:Chg_Event k-64               :REMark (A - L) Change Event Name
1231   =192:d% =d% -1:IF d%<1:d%=1                    :REMark Change Day
1232   =200:d% =d%+1:IF d%>dm%(m%):d%=dm%(m%)
1233   =208:m%=m%-1:IF m%<1:m%=1                      :REMark Change Month
1234   =216:m%=m%+1:IF m%>12:m%=12
1235  END SELect
1236 END REPeat Plan_lp
1237 END DEFine

1239 DEFine PROCedure Chg_Date
1240 Cal_Day yr%,m%,d%:STRIP#ch%,7
1241 CURSOR#ch%,352,30:PRINT#ch%,wd$(wd%);'← ';FILL$('0',2-LEN(d%))&d%;' → '
1242 CURSOR#ch%,404,30:PRINT#ch%,'↑ ';mth$(m%);'↓ ';yr%
1243 END DEFine
```

Note: Select Day/Month with cursors then Set with one of the Event colours [a-l]

```
1245 DEFine PROCedure Chg_Col
1246 IF d%<16:pn=34:ELSE pn=46
1247 IF Plnr%(m%,d%)=0:ec%=7:ELSE ec%=Pcol%(Plnr%(m%,d%))
1248 BLOCK#ch%,8,7,53,47,ec%:BLOCK#ch%,10,5,pn+1+d%*13,59+m%*8,ec%
1249 END DEFine
```

Use [**Shift A-L**] and access QBITS Text Editor to change any of the Twelve Event Names

```
1251 DEFine PROCedure Chg_Event(ev)
1252 SELect ON ev=1,5 , 9 :ex%=  50:ey%=164+10*(ev DIV 4)
1253 SELect ON ev=2,6,10 :ex%=160:ey%=164+10*(ev DIV 4)
1254 SELect ON ev=3,7,11 :ex%=280:ey%=164+10*(ev DIV 4)
1255 SELect ON ev=4,8,12 :ex%=398:ey%=154+10*(ev DIV 4)
1256 bc%=bcol%(6):TextEd 1,0,12,1,ex%,ey%,Event$(ev)
1257 END DEFine
```

```
1259 DEFine PROCedure Chart_Ent                    Writes the Year Chart Entries
1260 FOR mth=1 TO 12
1261  FOR day=1 TO dm%(mth)
1262    IF day<16:pn%=34:ELSE pn%=46
1263    IF Plnr%(mth,day)=0:pc%=7:ELSE pc%=Pcol%(Plnr%(mth,day))
1264    BLOCK#ch%,12,7,pn%+day*13,58+mth*8,0
1265    BLOCK#ch%,10,5,pn%+1+day*13,59+mth*8,pc%
1266  END FOR day
1267 END FOR mth
1268 IF dm%(2)=28:BLOCK#ch%,12,7,423,74,bcol%(6)
1269 END DEFine
```

```
1271 DEFine PROCedure Planner_Page                 Creates the double page display
1272 FOR mth=1 TO 12
1273   IF mth MOD 2=0:INK#ch%,0:ELSE INK#ch%,1
1274   FOR i=0 TO 1:CURSOR#ch%,24+i,56+mth*8:PRINT#ch%,mth$(mth)
1275 END FOR mth
1276 FOR n=1 TO 31
1277   IF n<10:N$='0'&n:ELSE N$=n
1278   IF n<16:pn%=34:ELSE pn%=44
1279   IF n MOD 2=0:INK#ch%,0:ELSE INK#ch%,1
1280   CURSOR#ch%,pn%+n*13,56:PRINT#ch%,N$
1281 END FOR n
1282 BLOCK#ch%,12,3,178,48,0:INK#ch%,0
1283 CURSOR#ch%,192,46:PRINT#ch%,'Contents'
1284 CURSOR#ch%,24, 46:PRINT#ch%,'[a-l   Colour (R)eset]'
1285 CURSOR#ch%,24,196:PRINT#ch%,'[A-L Edit (Esc)Rtn]'
1286:BLOCK#ch%,10,9,52,46,0:Chg_Col:RESTORE 1306:ev%=1
1287 FOR i=0 TO 11
1288    READ x:y=164+10*(i DIV 4)
1289    BLOCK#ch%,10,9,x,y,0:BLOCK#ch%,8,7,x+1,y+1,Pcol%(ev%)
1290    CURSOR#ch%,x+12,y:PRINT#ch%,CHR$(ev%+64);' ';Event$(ev%):ev%=ev%+1
1291 END FOR i
1292 DATA 26,136,256,374,26,136,256,374,26,136,256,374
1293 END DEFine
```

QBITS Organiser Addresses

Select **Year** on the **Contents** Page, entering **Addresses** Page will Load DATA file if previously Saved and display any Records. Select and Edit entries as required. When leaving Page (**S**)ave newly entered or updated records.

A Year suffix for Address DATA is kept as lists can change Year on Year. When leaving a Page the Arrays are not deleted. Beginning a New Year, change on Contents page and re-enter Addresses. Edit with new entries, delete old ones etc. then (S)ave...

```
1295 REMark Organiser Addresses Names/email/Mobile/Home/Office/Addresses

1297 DEFine PROCedure Addresses
1298 ch%=1:Prn_Bold 2,32,30,'Addresses':OLoad 1,'Org_'&yr%&'_Addr':FileCLS
1299 BLOCK#ch%,120,12,350,28,7:rn%=1:rs%=1:rf%=1:Address_Page:bc%=7
1300 REPeat Mail_lp
1301  Address_Ent rn%:k=CODE(INKEY$(-1))
1302  SELect ON k
1303   =192:rn%=rn%-1:IF rn%< 1:rn%=50                    :REMark rn% Rcd Numm
1304   =200:rn%=rn%+1:IF rn%>50:rn%=1
1305   =208:IF rs%> 1:rs%=rs%-1                           :REMark rs% Rcd Search
1306   =216:IF rs%<26:rs%=rs%+1
1307   =  9:rf%=rf%+1:IF rf%>2:rf%=1:END IF :STRIP#ch%,bcol%(1):Set_Tab
1308   = 10:rc%=1:Find_Rcd rn%
1309   =32:bc%=bcol%(1):OSave 1,'Org_'&yr%&'_Addr':EXIT Mail_lp
1310   =83,115:bc%=bcol%(1):OSave 1,'Org_'&yr%&'_Addr':FileCLS
1311   =76,108:TextEd 1,2,26,1,  20,  72,name$(rn%,1)      :REMark (L)ast Name
1312   =70,102:TextEd 1,0,26,1,  20,  98,name$(rn%,2)      :REMark (F)irst Name
1313   =69,101:TextEd 1,4,36,1,  20,124,email$(rn%)        :REMark (e)mail
1314   =65,  97:TextEd 1,0,32,3,260,  98,addr$(rn%)        :REMark (A)ddress
1315   =77,109:TextEd 1,1,16,1,140,146,mail$(rn%,1)        :REMark (M)obile
1316   =70,104:TextEd 1,1,16,1,140,160,phone$(rn%,2)       :REMark (H)ome
1317   =79,111:TextEd 1,1,16,1,140,174,phone$(rn%,3)       :REMark (O)ffice
1318   =80,112:TextEd 1,3,16,1,260,146,phone$(rn%,4)       :REMark (P)ost/Zip Code
1319   =68,100:TextEd 1,3,16,1,260,160,phone$(rn%,5)       :REMark (D)istrict
1320   =67,  99:TextEd 1,2,16,1,260,174,phone$(rn%,6)      :REMark (C)ountry
1321  END SELect
1322 END REPeat Mail_lp
1323 END DEFine
```

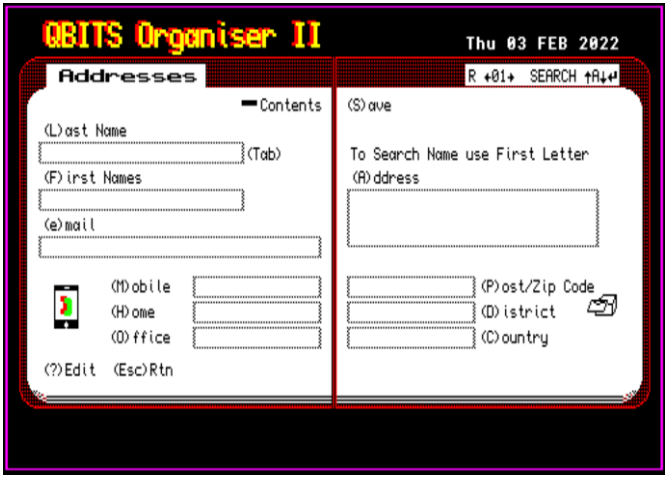
Last Name or **First Name** use **Tab** to switch between.

```
1325 DEFine PROCedure Set_Tab
1326 BLOCK#ch%,64,40,178,70,bcol%(1):CURSOR#ch%,178,47+26*rf%:PRINT#ch%,'(Tab)'
1327 END DEFine
```

Find based on the first character of the **Last Name** or **First Name**

```
1329 DEFine PROCedure Find_Rcd(rn%)
1330 REPeat rd_lp
1331   STRIP#ch%,bcol%(1):rn%=rn%+1:IF rn%>rm%:rn%=1          rn% record number : rm% max
1332   IF RS$ INSTR name$(rn%,rf%,1):RETurn :ELSE rc%=rc%+1   rc% records checked
1333   IF rc%>50:CURSOR#ch%,178,47+26*rf%:PRINT#ch%,' Not Found!':EXIT rd_lp
1334 END REPeat rd_lp
1335 PAUSE 50:Set_Tab
1336 END DEFine
```

```
1338 DEFine PROCedure Address_Ent(rn%)                Prints the Field Entries to screen
1339 RESTORE 1349:STRIP#ch%,7
1340 RS$=CHR$(rs%+64):CURSOR#ch%,448,30:PRINT#ch%,RS$
1341 CURSOR#ch%,370,30:PRINT#ch%,FILL$('0',2-LEN(rn%))&rn%
1342 sc%=26:sr%=1:sx%=  20:for i=0 to 1:sy%=72+i826:Prn_Str name$(rn%,i+1)
1343 sc%=36:sr%=1:sx%=  20:sy%=124:Prn_Str email(rn%)
1344 sc%=32:sr%=3:sx%=260:sy%=  98:Prn_Str addr$(rn%)
1345 sc%=16:sr%=1:FOR i=1 TO 6:READ sx%.sy%:Prn_Str phone$(rn%,i)
1346 DATA 146,146,146,160,146,174,260,146,260,160,260,174
1347 END DEFine
```




```
1349 DEFine PROCedure Address_Page                    Draws the Screen Layout
1350 Phone 30,150:Post 150,24:Set_Tab:RESTORE 1355
1351 FOR i=1 TO 10:READ a,b,x,y:BLOCK#ch%,a+4,b+2,x-2,y-1,248
1352 DATA 150,10,20,72,150,10,20,98,216,10,20,124        :REMark LName:FName:email
1353 DATA  90,10,260,146, 90,10,260,160,90,10,260,174    :REMark Zip Area Country
1354 DATA 90,10,146,146,90,10,146,160,90,10,146,174      :REMark Mobile:Home:Office
1355 DATA 192,30,260,98                                  :REMark Addrs
1356 FOR i=1 TO 13:READ a,b,M$ :CURSOR#ch%,a,b:PRINT#ch%,M$
1357 DATA 190,46,'Contents',256,46,'(S)ave',260,86,'(A)ddress'
1358 DATA 20,60,'(L)ast Name',20,86,'(F)irst Names',20,112,'(e)mail'
1359 DATA 72,146,'(M)obile',72,160,'(H)ome',72,174,'(O)ffice'
1360 DATA 360,146,'(P)ost/Zip Code',360,160,'(D)istrict',360,174,'(C)ountry'
1361 DATA 20,192,'(?)Edit  (Esc)Rtn',260,73,'To Search Name use First Letter'
1362 STRIP#ch%,7:CURSOR#ch%,352,30:PRINT#ch%,'R ←    →  SEARCH↑    ↓↑'
1363 BLOCK#ch%,12,3,176,48,0:BLOCK#ch%,2,4,466,32,0
1364 END DEFine
```

QBITS Organiser Contact Lists
As Contact list are an essential part of any Organiser the question was what information should the Addresses Page contain and in what format. An obvious starting point was, **Names**, Last name, First name, a nickname or non de plume. A search of names to find a related record was a consideration. As **emails** are now part of life what conventions exist over string length and allowable characters such as the **@** symbol. For Phone numbers **Mobile**, **Home** and **Office** seemed the most sensible choice.

Addresses come with possibly a Name for the Building be it Home, Apartment block or Office, maybe with a Number, then Street or Road name. The location; Village, Town or city **District**, the **Country** and for National Postal Services the use of **Post/Zip** codes.

Convention for Fields Entries
Government guidelines suggested Character fields as, **A** Alphabetic, **N** Numeric and **S** Special for example the **@** symbol. For names and addresses, field lengths were typically 30 to 35 alphanumeric characters and in the main used uppercase letters.

Surprisingly for email addresses the number of characters was 320 characters, the user's name limited to 64 plus the @ symbol and then Domain names up to 255. The special characters are Period (.), Underscore (_), Hyphen (-) and Plus sign (+).

QBITS Organiser Page Layout
What was a practicable layout that accommodated a number of fields with different sizes (string lengths), yet still met most users' requirements? The **emails** field for a start had to be limited to the Page width of 36 Alphanumeric characters A-Z, a-z, 0-9, the Special characters being period (.) underscore (_) hyphen (-) and the @ symbol.

QBITS Organiser Address Fields
To cover National Postal schemes, what was decided upon may not meet all local conventions. First a Text box with three rows for the Postal **Address** using alphanumeric and punctuation characters, then Numeric only and Upper-Case Alphabet characters for the **Post/Zip** Code, **District** and **Country** entries.

QBITS Organiser Name Search

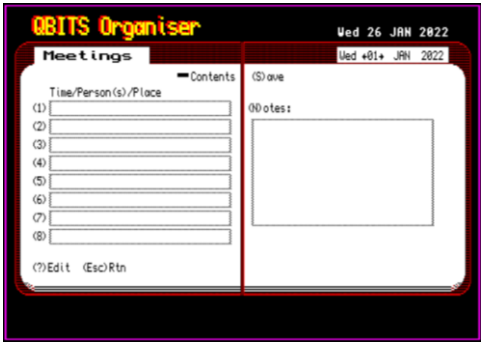
Use <**Left/Right**> Cursors to Select a **Record** or **Search Name** by selecting an Alphabet Letter A-Z with the <**Up/Down>** Cursor keys then press <**Enter**>. If no match is made '**Not Found!**' will be returned. If there are more entries with the chosen Alphabet letter, Press <**Enter**> until the one looked for is found. Alternatively, you can use the <**Left/Right>** Cursor keys to display the previous or the next sequential record.

To Edit the respective field key letter in [**?**] brackets of Selected field. When finished Exit with [**Esc**]. Records can be entered sequentially or on an add hock bases not necessarily in alphabetical order.

QBITS Organiser Meetings

Select **Year** and/or **Month** as required on the **Contents** Page. Entering the **Meetings** Page will Load DATA file if previously saved and display records. Select Day and Edit Meeting entries as required and (S)ave when leaving.

```
1366 REMark Organiser Meetings                    Meetings/Notes

1368 DEFine PROCedure Meetings
1369 ch%=1: Prn_Bold 2,32,30,'Meetings':OLoad 5,'Org_'&yr%&'_M_'&mth$(m%):FileCLS
1370 BLOCK#ch%,120,12,350,28,7:Meetings_Page:STRIP#ch%,7
1371 REPeat Appts_lp
1372  Chg_Day:Meetings_Ent:k=CODE(INKEY$(-1))
1373  SELect ON k
1374   =32      :OSave 5,'Org_'&yr%&'_M_'&mth$(m%):EXIT Appts_lp
1375   =83,115:OSave 5,'Org_'&yr%&'_M_'&mth$(m%):FileCLS
1376   =49 TO 56:n%=k-48:TextEd 1,0,32,1,42,56+n%*14,Appt$(d%,n%)
1377   =78,110                :TextEd 1,0,32,8,260,84,Note$(d%)
1378   =192:d%=d%-1:IF d%<1:d%=1
1379   =200:d%=d%+1:IF d%>dm%(m%):d%=dm%(m%)
1380  END SELect
1381 END REPeat Appts_lp
1382 END DEFine
```



Note: To Edit Select (1)(2)(3)(4)(5)(6)(7)(8)(N)

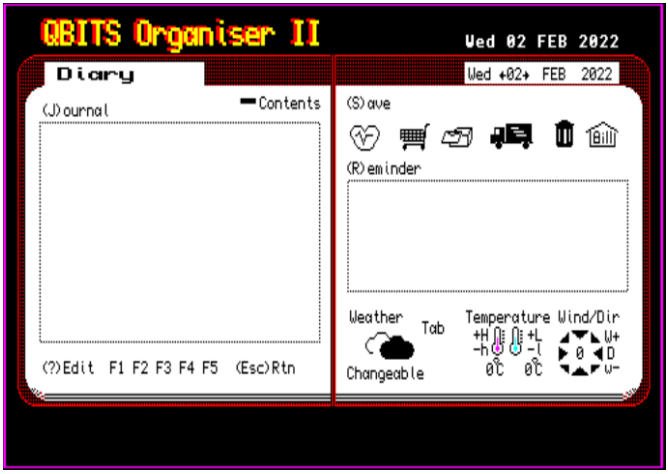
```
1384 DEFine PROCedure Chg_Day
1385 Cal_Day yr%,m%,d%:STRIP#ch%,7
1386 CURSOR#ch%,352,30:PRINT#ch%,wd$(wd%);' ← ';FILL$(0,2-LEN(d%))&d%;'→ '
1387 CURSOR#ch%,410,30:PRINT#ch%,mth$(m%);'  ';yr%
1388 END DEFine
```

Wed +01+ JAN 2022

```
1390 DEFine PROCedure Meetings_Ent
1391 sc%=32:sr%=1:sx%=42:FOR i=1 TO 8:sy%=56+i*14:Prn_Str Meet$(d%,i)
1392 IF Note$(d%)<>":sc%=1:sr%=1:sx%=1:sy%=1:Prn_Str Note$(d%)
1393 END DEFine

1395 DEFine PROCedure Meetings_Page
1396 CURSOR#ch%,192,46:PRINT#ch%,'Contents':BLOCK 12,3,178,48,0:FileCLS
1397 CURSOR#ch%, 40,58:PRINT#ch%,'Time/Person(s)/Place'
1398 FOR i=1 TO 8:CURSOR 20,56+i*14:PRINT '(';i;')'
1399 FOR i=1 TO 8:BLOCK 196,12,40,55+i*14,248
1400 CURSOR#ch%,252,70:PRINT#ch%,'(N)otes:':BLOCK#ch%,196,82,258,83,248
1401 CURSOR#ch%,20,192:PRINT#ch%,'(?)Edit   (Esc)Rtn'
1402 END DEFine
```

QBITS Organiser Diary

Select **Year** and/or **Month** as required on **Contents** Page. Entering **Diary** Page Loads DATA file if previously Saved and displays record. The layout has a (**J**)ournal for reporting on the day's events and a (**R**)eminder for all those important tasks not to be forgotten. Select Day and Edit as required, (**S**)ave when leaving.




```
1404 REMark Organiser Diary Page        Daily Journal/Reminder/Weather/Events

1406 DEFine PROCedure Diary
1407 ch%=1:Prn_Bold 2,32,30,'Diary':OLoad 4,'Org_'&yr%&'_D_'&mth$(m%):FileCLS
1408 BLOCK#ch%,120,12,350,28,7:Diary_Page:h%=0:l%=0:w%=0:wp%=0
1409 REPeat Diary_lp
1410   Chg_Day:Diary_Ent:Wth_Set:k=CODE(INKEY$(-1))
1411   SELect ON k
1412    =32:     OSave 3,'Org_'&yr%&'_D_'&mth$(m%):EXIT Diary_lp
1413    =83,115:OSave 3,'Org_'&yr%&'_D_'&mth$(m%)
1414    =74,106:TextEd 1,0,36,12,21,62,Jrnl$(d%)                    :REMark (J)ournal
1415    =82,114:TextEd 1,0,36,6,260,94,Rmdr$(d%)                    :REMark (R)eminders
1416    =9:wp%=wp%+1:IF wp%>  8:wp%=0:END IF    :Wth%(d%,1)=wp%:Wth_Set
1417    =  72:IF h% <   80:h%=h%+1:Wth%(d%,2)=h%  :Wth_Set          :REMark +(H)igh Temp
1418    =104:IF h% >  -20:h%=h% -1:Wth%(d%,2)=h%  :Wth_Set          :REMark - (h)igh Temp
1419    =  76:IF l% <    40:l% =l% +1:Wth%(d%,3)=l%  :Wth_Set       :REMark +(L)ow Temp
1420    =108:IF l% >  - 40:l% =l%  -1:Wth%(d%,3)=l%  :Wth_Set       :REMark - (l)ow Temp
1421    =  87:IF w%<200:w% =w%+1:Wth%(d%,4)=w% :Wth_Set             :REMark + Wind (F)actor
1422    =119:IF w% >   0:w% =w% -1:Wth%(d%,4)=w% :Wth_Set           :REMark - Wind (f)actor
1423    = 68,100:wp%=wp%+1:IF wp%>8:wp%=0:END IF :Wth%(d%,5)=wp%:Wth_Set
1424    =192:d%=d%-1:IF d%<1:d%=1
1425    =200:d%=d%+1:IF d%>dm%(m%):d%=dm%(m%)
1426   END SELect
1427 END REPeat Diary_lp
1428 END DEFine
```

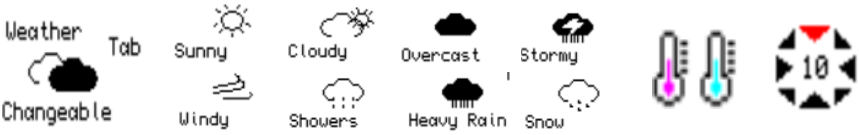
QBITS Diary Graphics

The icons were created as helpful reminders, the need of a Heath check or visit to the Gym, Shopping, Post of Letters or Parcels, expecting a Delivery, put out the Garbage or maybe a Services Bill is due Paying...

Press '**R**' to edit the (**R**)eminder textbox.

```
1430 DEFine PROCedure Diary_Ent                          Display Field Entries
1431 sc%=36:sr%=12:sx%=  21:sy%=62:Prn_Str Jrnl$(d%)     :REMark Journal entry
1432 sc%=36:sr%=  8:sx%=260:sy%=94:Prn_Str Rmdr$(d%)     :REMark Reminder entry
1433 END DEFine
```

The Weather Reporting uses Graphics to enhance the screen display and minimise the DATA storage. The Graphics are a mixture of **Vector** and **Bitmap** designs. Press the **Tab** key to scan through The Weather patterns range of Changeable, through Sunny, Cloudy, Windy, Showers, Overcast, Heavy Rain (Downpour), Stormy, to Snow.



Edit Temp and Wind speed using (+**H-h** /+**L-l** /+**W-w**), use '**Dd**' to highlight a Wind direction.

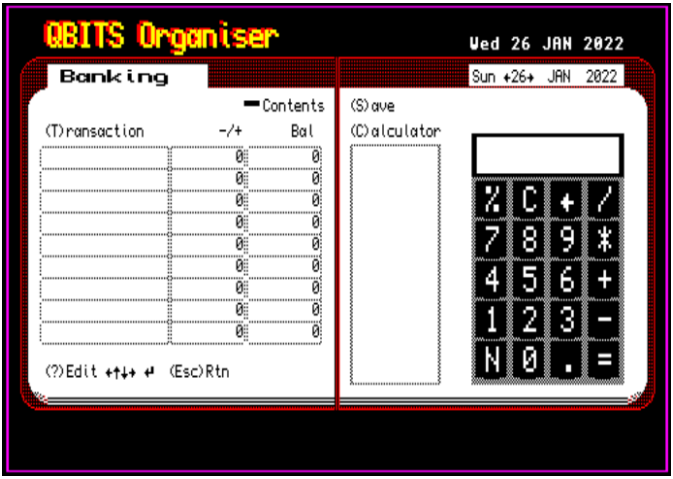
```
1435 DEFine PROCedure Wth_Set                            Weather Settings
1436 Weather Wth%(d%,1):WindDir Wth%(d%,4),Wth%(d%,5),148,14
1437 h%=Wth%(d%,2) :CURSOR#ch%,354,193:PRINT#ch%,FILL$(' ',3-LEN(h%))&h%
1438 l%=Wth%(d%,3)  :CURSOR#ch%,384,193:PRINT#ch%,FILL$(' ',3-LEN(l%))&l%
1439 END DEFine

1441 DEFine PROCedure Diary_Page
1442 ch%=1:BLOCK#ch%,218,122, 20,61,248:BLOCK#ch%,218,62, 259,93,248
1443 ch%=1:BLOCK#ch%,12,3,176,48,0:RESTORE 1452
1444 FOR i=1 TO 17:READ x%,y%,H$: CURSOR#ch%,x%,y%:PRINT#ch%,H$
1445 DATA 190,46,'Contents',256,46,'(S)ave',20,50,'(J)ournal', 256,82,'(R)eminder'
1446 DATA 260,164,'Weather',316,170,'Tab',350,164,'Temperature Wind/Dir'
1447 DATA 374,193,'C',356,174,'+H',356,182,'-h',458,175,'W+'
1448 DATA 404,193,'C',398,174,'+L',398,182,'-l',458,192,'w-'
1449 DATA 460,184,'D',20,192,'(?)Edit   F1 F2 F3 F4 F5  (Esc)Rtn'
1450 Therm 371,174,3:CIRCLE#ch%,126,12,.6:Therm 385,174,5:CIRCLE#ch%,136,12,.6
1451 Health 88,70:Shop 300,62:Post 112,66:Goods 370,62:Trash 420,60:CTax 154,70
1452 END DEFine
```

QBITS Organiser Banking

Select **Year** and **Month** as required on the **Contents** Page. Entering **Banking** Page Loads DATA file if previously Saved and displays daily entry. The Banking Page handles Transactions with a mini-spreadsheet and Calculator used as a Currency Ready Reckoner. Use Left/Right Cursors to select the Day if different from that displayed. Press <**Spacebar**> to leave or select (**S**)ave to Store the Current Data.

Enter **Transactions** by pressing (**T**). The top left Cell or last currently used will be highlighted. Navigate the various Cells with <**Cursor keys**> followed by <**Enter**> key. The Left Columns use **TextED** allowing alphanumeric entries. The two on the right act as currency columns with entries made via the Calculator **Editor**.

For independent use of the **Calculator** press (**C**). Directly type in numbers and Arithmetic Functions or use Cursors and Enter. Use (**Esc**) to Exit either Editor.




```
1454 REMark Organiser Banking Page      Daily Transactions & Calculator

1456 DEFine PROCedure Banking
1457 ch%=1::Prn_Bold 2,32,30,'Banking' :OLoad 2,'Org_'&yr%&'_B_'&mth$(m%);FileCLS
1458 BLOCK#ch%,120,12,350,28,7:Bank_Page:ch%=1:tchk=0:tl=0:oy%=y2%:om%=m2%
1459 REPeat Bank_lp
1460  Chg_Day:Bank_Ent:k=CODE(INKEY$(-1))
1461  SELect ON k
1462   =32      :OSave 2,'Org_'&yr%&'_B_'&mth$(m%):EXIT Bank_lp
1463   =83,115:OSave 2,'Org_'&yr%&'_B_'&mth$(m%):FileCLS
1464   =84,116:Transactions                    :REMark (T)ransactions
1465   =67,  99:Calculator '0'                 :REMark (C)alculator
1466   =192:d%=d%-1:IF d%<1:d%=1
1467   =200:d%=d%+1:IF d%>dm%(m%):d%=dm%(m%)
1468  END SELect
1469 END REPeat Bank_lp
1470 END DEFine
```

```
1472 DEFine PROCedure Bank_Ent
1473 FOR b=1 TO 9:FOR a=1 TO 3:bink%=248:OrgTrans a,b:END FOR a:END FOR b
1474 END DEFine

1476 DEFine PROCedure Transactions
1477 REPeat Acc_lp
1478  bink%=1:OTrans tc%,tr%:k=CODE(INKEY$(-1)):bink%=248:OrgTrans tc%,tr%
1479  SELect ON k
1480   =27:EXIT Acc_lp
1481   =10:IF tc%=1:bink%=0:TextEd 1,cm%,sc%,1,sx%,sy%,Tran$(d%,tr%,tc%)
1482      IF tc%>1:tchk=1       :Calculator Tran$(d%,tr%,tc%):tchk=0
1483   =192:IF tc%>1:tc%=tc% -1
1484   =200:IF tc%<3:tc%=tc%+1
1485   =208:IF tr% >1:tr% =tr% -1
1486   =216:IF tr% <9:tr% =tr%+1
1487  END SELect
1488 END REPeat Acc_lp
1489 END DEFine
```

OrgTrans Writes the Spreadsheet Entries, using **tc%** to set field columns (**sc%**), max string width in pixels (**sw%**), (**sx%**) for window x coordinate and **tr%** to set (**xy%**) y of row position. BLOCK commands are used to outline a Selected field. Press <**Enter**> to Edit or Exit a field. Select another field with Cursor Keys or <**Esc**> to leave spreadsheet. When Editing the letters and/or numbers is set by **cm%** character mode.

```
1491 DEFine PROCedure OrgTrans(tc%,tr%)
1492 IF tc%=1:tl%=0:sc%=16:sw%=98:sx%=20:cm%=0
1493 IF tc%>1:tl%=9-LEN(Tran$(d%,tr%,tc%)):sc%=9:sw%=56:sx%=3+tc%*59:cm%=1
1494 IF tc%>1 AND tl%=9:Tran$(d%,tr%,tc%)='0':tl%=8
1495 sy%=62+tr%*12:BLOCK#ch%,sw%,10,sx%,sy%,7
1496 BLOCK#ch%,sw%,1,sx%-1,sy%-1,bink%:BLOCK#ch%,sw%,1,sx%-1,sy%+11,bink%
1497 BLOCK#ch%,1,11,sx%-2,sy%,bink%       :BLOCK#ch%,1,11,sx%+sw%-1,sy%,bink%
1498 STRIP#ch%,7:CURSOR#ch%,sx%,sy%  :PRINT#ch%,FILL$(' ',tl%)&Tran$(d%,tr%,tc%)
1499 END DEFine

1501 DEFine PROCedure Bank_Page                    Set Up Page display
1502 ch%=1:RESTORE 1511
1503FOR i=1  TO 7:READ a,b,H$:CURSOR#ch%,a,b:PRINT#ch%,H$
1504 DATA 190, 46,'Contents',256,46,'(S)ave',256,60,'(C)alculator'
1505 DATA  20, 60,'(T)ransaction',156,60,'-/+',212,60,'Bal'
1506 DATA  20,192,'(?)Edit ←↑↓→←  F1 F2 F3  (Esc)Rtn'
1507 BLOCK#ch%,12,3,176,48,0:BLOCK#ch%,2,4,104,194,0
1508 ch%=3:PAPER#ch%,7:BORDER#ch%,1,248:INK#ch%,0:CLS#ch%:CSIZE#ch%,0,0
1509 ch%=4:PAPER#ch%,248:INK#ch%,7:STRIP#ch%,0:CLS#ch%:CSIZE#ch%,2,1
1510 FOR kr=1 TO 5
1511   FOR kc=1 TO 4
1512     BLOCK#ch%,24,20,-26+kc*29,2+kr*22,0
1513     CURSOR#ch%,-20+kc*29,3+kr*22:PRINT#ch%,Key$(kc,kr)
1514   END FOR kc
1515 END FOR kr
1516 ch%=5:PAPER#ch%,7:BORDER#ch%,2,0:INK#ch%,0:CLS#ch%:CSIZE#ch%,2,1
1517 c%=2:r%=3:k=0:km=0:Num$='0':num=0:dot=0:chk=1:slen=5:tc%=1:tr%=1
1518 END DEFine
```

QBITS Text Editor Review

Addresses, Banking, Diary, Meetings and **Planner** use different Character sets for handling field entries. To investigate code to meet this need various Editors were viewed only to produce a list each with their own twist on functionality. In the end a more practical approach was taken to obtain a acceptable code to get the job done!

QBITS Text Editor Considerations

Previously written **QBITS** code for editing a string was a starting point, this included a character **Add/Delete** function anywhere within the string. The challenge now was to introduce a **Wrap** for multiple rows, and a **New Line** function, while making sure the character string remained within a defined Text Box screen area.

Other thoughts that came to mind, generally most keyboards come with a dedicated **Delete** and **Backspace**. The **QL** Keyboard uses **CTRL Right** and **CTRL Left** Cursor keys. For the **QBITS TextED** it was decided to use the Function keys. **F1** to act as a Backspace, Deleting the character left of cursor position. **F2** to Delete character above cursor position. **F3** Delete to End of Line (opposite to a New Line) and **F4** Delete all of the current string within the Text Box.

When deleting a String (**F4**) this presented an opportunity to store it in a buffer and RESTORE using (**F5)**. This meant a character string could be transferred to other fields. However, problems arise between different field sizes and control of designated character sets. Therefore, this function is restricted, and for the most part if a string is longer than the transferred field space, the string is truncated.

The string position is identified on screen by an **underline** style cursor. This is moved horizontally by column and vertically by rows anywhere inside of the Text Box, but still restrained by the columns and rows of the current string length.

QBITS Text Editor ASCII Codes

Published International standards for Character Data fields helped in deciding the ASCII code sets for Names, Postal Address, Post/Zip Codes, Telephone numbers and emails. For control of the differing code sets, a method of scalable character groups is deployed beginning with numbers 0-9 and characters such as space, hyphen, period, then adding upper-case alphabetic characters followed by lower case, then finally building to a Full Set to include all punctuation characters, mathematical symbols and brackets etc.

‘space’ ! ” # $ % & ‘ ( ) * + , - . /	codes	32 -	47
0123456789	codes	48 -	57
: ; < = > @	codes	58 -	64
ABCDEFGHIJKLMNOPQRSTUVWXYZ	codes	65 -	90
[\ ] ^ £	codes	91 -	96
abcdefghijklmnopqrstuvwxyz	codes	97 -	122
{ | } ~	codes	123 -	126

This in General terms were the goals set for the **QBITS TextEditor**.

QBITS Text Editor Character String
In SuperBASIC an input of character codes grouped as a string variable is suffixed with '$'. Therefore, the first step is just **Add**ing characters to a string (**str$)**:

```
k$=INKEY$        str$=k$ to begin then, str$=str$&k$ to continue adding characters
```

To add characters within the string requires a little more manipulation. First, a variable to locate the current character position **cp%** then the string is added to as follows:

```
k$=INKEY$        str$=str$(1 to cp%-1)&k$&str$(cp% TO):cp%=cp%+1
```

QBITS Text Editor Character Delete
In a similar way characters in the string can be Deleted, here another variable is needed to identify the string length **sl%**. This creates a number of possible **Delete** outcomes:

```
IF cp%=1                  : str$=str$(2 TO)                        : sl%=sl%-1
IF cp%>1 AND cp%<sl%      : str$=str$(1 to cp%-1)&str$(cp%+1 TO) : sl%=sl%-1
IF cp%>1 AND cp%=sl%      : sl%=sl%-1 : str$=(1 to sl%)
IF str$=''                : sl%=0 : cp%=1
```

Further constraints are the number of columns s**c%** and rows sr**%** and maximum string length **sm%**, which set the boundaries within which string manipulation can take place.

QBITS Text Editor New Line
For a **New Line** the **cp%** character position is moved to the beginning of the next row**.** For this, variable **cn%** is introduced to count character spaces to extend the string into the next row. (While Checks that s**c%,sr%** & **sm%** boundaries are not exceeded).

```
IF sl%=0: Return
IF sl% DIV sc%<sr%-1                    Check row position sr%-1
   cn%=sc%-cp% MOD sc%+1                Count number of spaces to end of row
   IF cp%<=sl%:str$=str$(1 TO cp%-1)&FILL$(' ',cn%)&str$(cp% TO)
   IF cp% > sl%:str$=str$(1 TO cp%-1)&FILL$(' ',cn%)
   cp%=cp%+cn% : sl%=sl%+cn%
END IF
```

QBITS Text Editor Character Display
Characters are printed to screen using the attached attributes of TextED to identify **SuperBASIC** output channel **ch%**, the **ASCII** code set selected by IF **cm%** statements and acted upon by **SELect ON** k Input Key. The size of screen area calculated from columns **sc%**, rows **sr%,** and start of screen location given by **sx%**, **sy%.** The string mechanism for **Wrap** is controlled by a check row variable **cr%**.

QBITS Text Editor Position Cursor
The QBITS Editor uses the BLOCK command to provide an underscore visual indicator. CSIZE 0,0 (six pixels wide) is used to calculation current character cursor coordinates c**x%, cy%.** The variable **cp%** is now used for string position and for calculating cursor pixel position while checking against **str$** boundaries: 0, s**c%, sr%, sl%** and **sm%**.

Summary: k$, str$, cp%, sc%, sr%, sl%, sm%, cn%, sx%, sy%, cr%, cx%, cy%

```
1520 REMark QBITS Text Editor     Set channel, character mode, column, row, x, y, character string

1522 DEFine PROCedure TextEd(ch%,cm%,sc%,sr%,sx%,sy%,str$)
1523 cx%=sx%:cy%=sy%:cp%=1
1524 REPeat Edit_lp
1525  Prn_Str str$:Prn_Cur:k=CODE(INKEY$(-1))
1526  SELect ON k
1527   = 10:IF sr%>1      :Add_NIn                                  :REMark New Line
1528   = 27:EXIT Edit_lp                                            :REMark End Edit
1529   = 32 TO 125:k$=' ':Chk_Chr:IF k$>":Add_Chr                   :REMark Add Character
1530   =194,232:IF cp%>1:cp%=cp%-1     :Del_Chr                     :REMark Ctrl Left   F1 - Delete Left
1531   =202,236                        :Del_Chr                     :REMark Ctrl Right F2 - Delete Char
1532   =240                            :Del_EOL                     :REMark F3 Delete End of Line
1533   =244                            :Del_Page                    :REMark F4 Delete Page
1534   =248                            :Res_Str                     :REMark F5 Restore Page
1535   =192:IF cp%> 1         :cp%=cp%-1                            :REMark Move Left
1536   =200:IF cp%<=sl%       :cp%=cp%+1                            :REMark Move Right
1537   =208:IF cp%>sc%        :cp%=cp%-sc%                          :REMark Move Up
1538   =216:IF cp%<sl%-sc%:cp%=cp%+sc%:ELSE IF sl%>0:cp%=sl%        :REMark Move Down
1539  END SELect
1549 END REPeat Edit_lp
1541 BLOCK#ch%,bw%,bd%,sx%,sy%,bc%:STRIP#ch%,bc%:Str_Prn str$
1542 END DEFine
```

Note: Text Box Cleared and returned Character String Printed with BkGnd Colour Bc%

```
1544 DEFine PROCedure Chk_Chr                          Check Character Mode
1545  IF cm%=0                 :SELect ON k=32 TO 127     :k$=CHR$(k):RETurn
1546  IF cm%=1 OR cm%>2        :SELect ON k=32,43,45,46,48 TO  57 :k$=CHR$(k)
1547  IF cm%>1                 :SELect ON k=65 TO  90     :k$=CHR$(k)
1548  IF cm%=2 OR cm%=3        :SELect ON k=97 TO 122     :k$=CHR$(k-32)
1549  IF cm%>3                 :SELect ON k=97 TO 122     :k$=CHR$(k)
1550  IF cm%=4                 :SELect ON k=64            :k$=CHR$(k)
1551  IF cm%=5                 :SELect ON k=42,47,61      :k$=CHR$(k)
1552 END DEFine

1554 DEFine PROCedure Add_Chr                          Adds Character to String
1555 IF cp%=1 AND sl%=0:str$=str$&k$
1556 IF cp%>=1 AND cp%<sl%:str$=str$(1 TO cp%-1)&k$&str$(cp% TO sl%)
1557 IF cp%>=1 AND cp%=sl%:str$=str$(1 TO cp%-1)&k$&str$(cp%)
1558 IF cp%>1 AND cp%>sl%:str$=str$&k$
1559 IF cp%=sm%:str$(cp%)=k$
1560 IF sl%<sm%:sl%=sl%+1:ELSE sl%=sm%
1561 IF cp%<sm%:cp%=cp%+1:ELSE cp%=sm%
1562 END DEFine

1564 DEFine PROCedure Del_Chr                          Deletes a Character from String
1565 IF cp%=sl%:str$=str$(1 TO sl%-1):sl%=sl%-1
1566 IF cp%>=1 AND cp%<sl%:str$=str$(1 TO cp%-1)&str$(cp%+1 TO sl%):sl%=sl%-1
1567 IF cp%=sm%:str$=str$(1 TO sm%-1):cp%=cp%-1:sl%=sm%-1
1568 IF cp%=1 AND sl%=1:str$=" ":sl%=0
1569 END DEFine
```

```
1571 DEFine PROCedure Del_EOL                    Delete to End of Line [F3]
1572 IF cp%>=sl%:RETurn
1573 buf$=str$:cn%=1+sc%-cp% MOD sc%:IF cn%>sl%-cp%:cn%=sl%-cp%
1574 str$=FILL$(' ',sl%):Str_Prn str$
1575 str$=buf$(1 TO cp%-1)&buf$(cp%+cn% TO LEN(buf$)):sl%=LEN(str$)
1576 END DEFine
```

Note: Variable buf$ provides a buffer storage of String str$.

```
1578 DEFine PROCedure Del_Str                    Clear Text Box [F4]
1579 buf$=str$:str$="":sl%=0:cp%=1:BLOCK#ch%,sc%*6,sr%*10,sx%,sy%,7
1580 END DEFine
```

```
1582 DEFine PROCedure Res_Str                    Restore Text from Buffer [F5]
1583 IF LEN(buf$)>sm%:str$=buf$(1 TO sm%):sl%=sm%:ELSE str$=buf$
1584 END DEFine
```

```
1586 DEFine PROCedure Add_NIn            Add New line
1587 IF sl%=0:RETurn :ELSE cn%=1+sc%-cp% MOD sc%
1588 IF sl% DIV sc%<sr%-1 OR sl%+cn%<sm%
1589    IF cp%<=sl%
1590       str$=str$(1 TO cp%-1)&FILL$(' ',cn%)&str$(cp% TO sl%)
1591       cp%=cp%+cn%:sl%=LEN(str$)
1592    END IF
1593    IF cp%> sl%:str$=str$(1 TO cp%)&FILL$(' ',cn%):cp%=cp%+cn%:sl%=cp%:END IF
1594 END IF
1595 END DEFine
```

Note: INK and STRIP colours are set independently.

```
1597 DEFine PROCedure Prn_Str(str$)              Clear text Box and Print String
1598 BLOCK#ch%,sc%*6,sr*10,sx%,sy%,7:STRIP#ch%,7
1599 sm%=sc%*sr%:sl%=LEN(str$):IF sl%>sm%:sl%=sm%
1600 cr%=sl% DIV sc%:IF cr%=sr%:cr%=sr%-1
1601 FOR r=0 TO cr%
1602   IF sl%>0 AND sl% MOD sc%>=0
1603      CURSOR#ch%,sx%,sy%+r*10:PRINT#ch%,str$(1+sc%*r TO sc%+sc%*r)
1604   END IF
1605 END FOR r
1606 IF sl%=0:CURSOR#ch%,sx%,sy%:PRINT#ch%,' ':END IF
1607 END DEFine
```

```
1609 DEFine PROCedure Prn_Cur                    Display Position of Cursor
1610 BLOCK#ch%,6,1,cx%,cy%+9,7
1611 IF cp% MOD sc%=0:cx%=sx%+sc%*6-6       :cy%=sy%+cp% DIV sc%*10-10
1612 IF cp% MOD sc%>0:cx%=sx%+cp% MOD sc%*6-6:cy%=sy%+cp% DIV sc%*10
1613 IF cp%=1        :cx%=sx%               :cy%=sy%
1614 IF cp%>=sm%     :cx%=sx%+sc%*6-6       :cy%=sy%+sr*10-10
1615 BLOCK#ch%,6,1,cx%,cy%+9,2
1616 END DEFine
```

QBITS Calculator

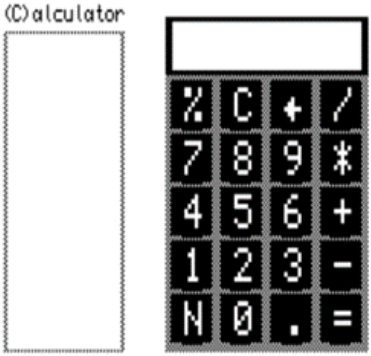
The Calculator Arithmetic Functions are [+] Add, [-] Subtract, [*] Multiply, [/] Divide. Percentage a number [*] times a value of [%]. Other keys, [=] Equal sign, [.] Decimal Point, [**N**]egate +/-number, [**C**]lear all, [🡐] Delete last character. The Calculator works independently or with mini-spreadsheet as a Currency Reckoner (two decimal places).

```
1618 REMark QBITS Calculator (NNNNN.nn)

1620 DEFine PROCedure Calculator(Num$)
1621 REPeat Calc_lp
1622  INK#4,2:Calc_Key c%,r%:PRINT#5,FILL$(' ',9-LEN(Num$))&Num$
1623  IF tchk=1:CURSOR#1,sx%,sy%:PRINT#1,FILL$(' ',9-LEN(Num$))&Num$
1624  k=CODE(INKEY$(-1)):INK#4,7:Calc_Key c%,r%:IF k=10:k=CODE(Key$(c%,r%))
1625  SELect ON k
1626    =27:INK#4,7:Calc_Key c%,r%:ch%=1:EXIT Calc_lp
1627    =194          :Calc_BkSp                  :REMark [»] Delete
1628    =67,99        :Calc_Cls                   :REMark [C]lear
1629    =78,110       :Calc_Negate                :REMark [N]egate
1630    =37,42,43,45,47,61:Calc_Func              :REMark [%/*+-=] Function
1631    =46:IF dot=0  :Calc_Dot   :chk=0:dot=1    :REMark [.] Decimal Point
1632    =48 TO 57     :Calc_Add                   :REMark [0 to 9]
1633    =192:c%=c% -1:IF c%<1:c%=4                :REMark Key column
1634    =200:c%=c%+1:IF c%>4:c%=1
1645    =216:r% =r%+1:IF r% >5:r%=1               :REMark Key row
1636    =208:r% =r% -1:IF r% <1:r%=5
1637  END SELect
1638 END REPeat Calc_lp
1639 IF Num$>0:km=32:Calc_Dec
1640 END DEFine
```

Note: REPeat Calc_lp checks for Numeric Input or Maths Function, keyed directly or via **Cursor Keys & Enter**.

Note: Printing Numbers when Growing from Right to Left

PRINT#5,FILL$(' ',9-LEN(Num$))&Num$




```
1642 DEFine PROCedure Calc_Key(c%,r%)          Updates the Highlighted Key on KeyPad
1643 ch%=4:CURSOR#ch%,-20+c%*29,3+r%*22:PRINT#ch%,Key$(c%,r%)
1644 END DEFine

1646 DEFine PROCedure Calc_BkSp                Deletes Last Character
1647 Num$=Num$(1 TO LEN(Num$)-1)
1648 IF LEN(Num$)=0:Num$='0':chk=1:dot=0:slen=5
1649 IF LEN(Num$)=1 AND '-' INSTR Num$=1:Num$='0':chk=1:dot=0:slen=5
1650 IF '.' INSTR Num$=0:dot=0:slen=5
1651 IF '.' INSTR Num$>1:slen=Num$-LEN(INT(Num$))+2
1652 END DEFine
```

```
1654 DEFine PROCedure Calc_Cls                Resets Numbers Num$,num, Clears Work Areas
1655 CLS#3:CLS#5:Num$='0':num=0:dot=0:slen=5
1656 END DEFine

1658 DEFine PROCedure Calc_Negate             Toggles +/- Value of Number
1659 IF Num$='0':RETurn
1660 IF Num$(1)='-':Num$=Num$(2 TO LEN(Num$)):ELSE Num$='-'&Num$
1661 END DEFine

1663 DEFine PROCedure Calc_Dot                Adds Decimal Point
1664 IF chk=1:Num$='0.':slen=4:RETurn
1665 IF Num$-INT(Num$)=0:Num$=Num$&'.':slen=LEN(Num$)+2
1666 END DEFine

1668 DEFine PROCedure Calc_Add                Adds a Numerical Value
1669 IF chk=1:chk=0:Num$=CHR$(k):ELSE IF LEN(Num$)<slen:Num$=Num$&CHR$(k)
1670 END DEFine
```

Note: To avoid QL SuperBASIC printing values as self-describing integers (1E6 etc.) and the QPC2 SMSQ/E apparent rounding up of 99999.99 to 100000, the QBITS Calculator is limited to string values of only eight numeric characters ranged between +99998.99 and -99998.99 exceeding these values will incur an Overflow.

```
1672 DEFine PROCedure Calc_Err
1673 IF Num$>99999.89 OR Num$<-99999.89:Num$='!':ELSE num=Num$
1674 END DEFine
```

Note: Formatting to two decimal points requires checking if a decimal point already exists within the Character String (Num$) and then to what number of decimal places.

```
1676 DEFine PROCedure Calc_Dec                Identifies and Adds Full Decimal Notation
1677 IF Num$='!':PRINT#3,'= Overflow!':Num$='0':num=0:RETurn
1678 IF '-' INSTR Num$=0 AND Num$<.1:Num$='0':RETurn    :REMark QL (1E6 etc)
1679 IF Num$-INT(Num$)=0:dec$='.00' :ELSE dec$=Num$('.' INSTR Num$ TO LEN(Num$))
1680 IF LEN(dec$)<3:dec$=dec$&'0'
1681 Num$=INT(Num$)&dec$(1 TO 3):PRINT#3,CHR$(km);FILL$(' ',10-LEN(Num$))&Num$
1682 END DEFine
```

Note: km holds the math function [**% / * + -**] num & Num$ the numeric values (n1 n2).

```
1684 DEFine PROCedure Calc_Func               Carries out the Math Functions
1685 IF km=47 AND Num$='0':chk=1:RETurn
1686 IF num=0:km=32:Calc_Err:Calc_Dec:chk=1:dot=0:slen=5:num=Num$:km=k:RETurn
1687 IF k=37:km=37:k=61                       :REMark '%' Percentages
1688 IF km<>61:Calc_Dec Num$:chk=1            :REMark '/ * + -' Functions
1689 SELect ON km
1690   =37:Num$=num*Num$/100    :REMark '%' n='n1*n2/100'
1691   =47:Num$=num/Num$        :REMark '/'   'n1/n2'
1692   =42:Num$=num*Num$        :REMark '*'   'n1*n2'
1693   =43:Num$=num+Num$        :REMark '+'   'n1+n2'
1694   =45:Num$=num-Num$        :REMark '-'   'n1-n2'
1695 END SELect
1696 Calc_Err:km=61:Calc_Dec:km=k:chk=1       :REMark Results
1697 END DEFine
```

QBIT Organiser Data Storage

Dependant on **Device**/**Year**/**Month** Selected, Data Files are Loaded if previously Saved relevant to the Organiser Page being accessed. The field entries are then displayed in accordance to the Record / Day chosen. Array fields are not Deleted or Overwritten when leaving an Organiser Page.

Note: LOAD Config DATA on Initialising Organiser or from Contents Page with change of Theme.

```
1699 REMark Organiser Data Storage

1701 DEFine PROCedure FCheck
1702 STRIP#ch%,bc%:CURSOR#ch%,256,46:PRINT#ch%,'Searching...     '
1703 PAUSE 30:eck=0:ck%=0:DELETE drv$(dn%)&'FList'
1704 OPEN_NEW#9,drv$(dn%)&'FList':DIR#9,drv$(dn%):CLOSE#9
1705 OPEN_IN#9,drv$(dn%)&'FList'
1706 REPeat dir_lp
1707   IF EOF(#9):CLOSE#9:ck=0:EXIT dir_lp
1708   INPUT#9,DFile$:IF DFile$==OrgFile$:CLOSE#9:ck=1:EXIT dir_lp
1709END REPeat dir_lp
1710 END DEFine

1712 DEFine PROCedure OLoad(OFT,OrgFile$)                    Check File exists in DIR List
1713 ch%=1:FCheck
1714 IF ck=0 OR eck=1
1715    CURSOR 256,46:PRINT 'File NOT Found':eck=0:PAUSE 50:RETurn
1716 END IF
1717 OPEN_IN#9,drv$(dn%)&OrgFile$
1718 CURSOR#ch%,256,46:PRINT#ch%,'Loading...';FILL$(' ',20)
1719 IF OFT=0:FOR a=0 TO 6:INPUT#9,bcol%(a)              :REMark Config
1720 IF OFT=1                                            :REMark Addr List
1721    FOR a=1 TO 50:FOR b=1 TO 5:INPUT#9,name$(a,b):END FOR b:END FOR a
1722    FOR a=1 TO 50:FOR b=1 TO 6:INPUT#9,mail$(a,b):END FOR b:END FOR a
1723    FOR a=1 TO 50:INPUT#9,email$(a)\addr$(a)
1724 END IF
1725 IF OFT=2                                            :REMark Banking
1726    FOR a=1 TO 31
1727      FOR b=1 TO 9:FOR c=1 TO 3:INPUT#9,Tran$(a,b,c):END FOR c:END FOR b
1728    END FOR a
1729 END IF
1730 IF OFT=4                                            :REMark Diary
1731   FOR a=1 TO 31:INPUT#9,Jrnl$(a)
1732    FOR a=1 TO 31:INPUT#9,Rmdr$(a)
1733    FOR a=1 TO 31:FOR b=1 TO 5:INPUT#9,Wth%(a,b):END FOR b:END FOR a
1734 END IF
1735 IF OFT=5                                            :REMark Meetings
1736    FOR a=1 TO 31:FOR b=1 TO 6:INPUT#9,Appt$(a,b):END FOR b:END FOR a
1737    FOR a=1 TO 31:INPUT#9,Note$(a):END FOR a
1738 END IF
1739 IF OFT=6                                            :REMark Planner
1740    FOR a=1 TO 12:FOR b=1 TO 31:INPUT#9,Plnr%(a,b):END FOR b:END FOR a
1741    FOR a=1 TO 12:INPUT#9,Pcol%(a)\Event$(a)
1742 END IF
1743 CLOSE#9:PAUSE 50:FileCLS
1744 END DEFine
```

Note: Select (**S**)ave or when leaving a Page [**Spacebar]** you are prompted to Store the current Data. Either the data will be saved as a New file or if previously saved then an **Overwrite Y/N** prompt will be given.

If the Device or Medium is not found a **DEVICE ERROR** will be given. To save Data return without Saving back to the **Contents** Page and select a Device that is attached (**Do not change Year or Month**). Return again to **Organiser Page** last accessed, the field entries should still be there so that they can now be (**S**)aved.

```
1746 DEFine PROCedure OSave(OFT,OrgFile$)
1747 ch%=1:STRIP#ch%,bc%:CURSOR#ch%,300,46:PRINT#ch%,'Y/N':PAUSE
1748 IF KEYROW(5)<>64:BLOCK#ch%,20,10,300,46,bc%:RETurn :ELSE FCheck
1749 IF eck=1
1750    CURSOR#ch%,256,46:PRINT#ch%,'DEVICE ERROR...':eck=0:PAUSE 50:RETurn
1751 END IF
1752 IF ck=1
1753    CURSOR#ch%,256,46:PRINT#ch%,'Overwrite Y/N  ':PAUSE:IF KEYROW(5)<>64:RETurn
1754 END IF
1755 DELETE drv$(dn%)&OrgFile$:OPEN_NEW#9,drv$(dn%)&OrgFile$
1756 CURSOR#ch%,256,46:PRINT#ch%,'Saving...';FILL$(' ',20)
1757 IF OFT=0:FOR a=0 TO 6:PRINT#9,bcol%(a)              :REMark Config
1758 IF OFT=1                                            :REMark Addr List
1759   FOR a=1 TO 50:FOR b=1 TO 5:PRINT#9,name$(a,b):END FOR b:END FOR a
1760   FOR a=1 TO 50:FOR b=1 TO 6:PRINT#9,mail$(a,b):END FOR b:END FOR a
1761   FOR a=1 TO 50:PRINT#9,email$(a)\addr$(a)
1762 END IF
1763 IF OFT=2                                            :REMark Banking
1764   FOR a=1 TO 31
1765     FOR b=1 TO 9:FOR c=1 TO 3:PRINT#9,Tran$(a,b,c):END FOR c:END FOR b
1766   END FOR a
1767 END IF
1768 IF OFT=4                                            :REMark Diary
1769   FOR a=1 TO 31:PRINT#9,Jrnl$(a)\Rmdr$(a)
1770   FOR a=1 TO 31:FOR b=1 TO 5:PRINT#9,Wth%(a,b):END FOR b:END FOR a
1771 END IF
1772 IF OFT=5                                            :REMark Meetings
1773   FOR a=1 TO 31:FOR b=1 TO  8:PRINT#9,Appt$(a,b):END FOR b:END FOR a
1774   FOR a=1 TO 31:PRINT#9,Note$(a):END FOR a
1775 END IF
1776 IF OFT=6                                            :REMark Planner
1777   FOR a=1 TO 12:FOR b=1 TO 31:PRINT#9,Plnr%(a,b):END FOR b:END FOR a
1778   FOR a=1 TO 12:PRINT#9,Pcol%(a)\Event$(a)
1779 END IF
1780 CLOSE#9:PAUSE 50:FileCLS
1781 END DEFine
```

File Actions

(S)ave Y/N Searching... DEVICE ERROR... Overwrite Y/N Saving...

Load Data Files File NOT Found Loading...

```
1783 DEFine PROCedure FileCLS
1784 CURSOR#ch%,256,46:PRINT#ch%,'(S)ave';FILL$(' ',15)
1785 END DEFine
```

QBITS Organiser Graphics

These use a combination of **Vector** and **Bitmap** designs. **Vectors** use SuperBasic ARC, CIRCLE, LINE and **Bitmaps** use the BLOCK command.

First the image is sketched, then redrawn over a Grid to map the coordinates. **Vectors** use **Graphics coordinates** of x y Axis for positioning, **Bitmap** images use x y top left to locate with the **Pixel coordinate** system.

```
1787 REMark QBITS Organiser Graphics

1789 DEFine PROCedure WindDir(wn%,wd%,x,y)
1790 LOCal a,b,c,d,e,f,l:RESTORE 1817:BLOCK 18,10,430,184,bcol%(4)
1791 FOR w=1 TO 8
1792    READ a,b,c,d,e,f:IF w=wd%:INK 2:ELSE INK 0
1793    FILL 1:LINE a+x,b+y TO c+x,d+y TO e+x,f+y TO a+x,b+y:FILL 0
1794 END FOR w
1795 DATA 0,3.8,-2,5.5,+2,5.5,3,3,3,5,5,3
1796 DATA 3.5,0,5.5,-1.5,5.5,1.5,3,-3,3,-5,5,-3
1797 DATA 0,-3.5,2,-5.5,-2,-5.5,-3,-3,-3,-5,-5,-3
1798 DATA -3.5,0,-5.5,-1.5,-5.5,1.5,-3,3,-3,5,-5,3
1799 l=INT(LEN(wn%)*6)/2:STRIP bcol%(4):INK 0:CURSOR x,y,-l,-4:PRINT wn%
1800 END DEFine
```

```
1802 DEFine PROCedure Weather(wp)
1803 ch%=1:INK#ch%,gc%:wx%=98:wy%=16:wr%=2
1804 STRIP#ch%,bcol%(4):BLOCK#ch%,46,30,270,174,bcol%(4)
1805 SELect ON wp
1806    =0:Cloud      ch%,wx%-2,wy%,wr%        :Overcast ch%,wx%+2,wy%-1,wr%
1807    =1:Sunny      ch%,wx%,wy%,wr%
1808    =2:Sunny      ch%,wx%+4,wy%+1,wr%*.7 :Cloud ch%,wx%,wy%-1,wr%
1809    =3:Windy      ch%,wx%,wy%,wr%
1810    =4:Showers    ch%,wx%,wy%,wr%
1811    =5:Overcast   ch%,wx%,wy%,wr%
1812    =6:Downpour   ch%,wx%,wy%,wr%
1813    =7:Stormy     ch%,wx%,wy%,wr%
1814    =8:Snow       ch%,wx%,wy%,wr%
1815 END SELect
1816 CURSOR#ch%,258,195:PRINT#ch%,Sky$(wp)&FILL$(' ',10-LEN(Sky$(wp)))
1817 END DEFine
```

Changeable

Cloudy

The Weather Patters

```
1819 DEFine PROCedure Sunny(ch%,wx%,wy%,wr%)
1820 CIRCLE#ch%,wx%,wy%,wr% :r1%=wr%*1.5:r2%=wr%*2
1821 FOR i=1 TO 9
1822   LINE#ch%,wx%+r1%*COS(i*60),wy%+r1%*SIN(i*60)
1823   LINE#ch% TO wx%+r2%*COS(i*60),wy%+r2%*SIN(i*60)
1824 END FOR i
1825 END DEFine
```

```
1827 DEFine PROCedure Cloud(ch%,wx%,wy%,wr%)
1828 ARC#ch%,wx%-wr%*1.2,wy%-wr% TO wx%-wr%,wy%+wr%,-PI
1829 ARC#ch% TO wx%+wr%,wy%+wr%,-PI*.8 TO wx%+wr%*1.2,wy%-wr%,-PI
1830 END DEFine:
```

Weather Patterns cont...

```
1832 DEFine PROCedure Windy(ch%,wx%,wy%,wr%)
1833 LINE#ch%,wx%-wr%*2,wy%+wr%/2 TO wx%,wy%+wr%:ARC#ch% TO wx%,wy%+wr%
1834 LINE#ch%,wx%-wr%*2,wy%-wr%/4 TO wx%+wr%*.6,wy%+wr%/4
1835 ARC#ch% TO wx%+wr%*.6,wy%+wr%*1.4,PI
1836 LINE#ch%,wx%-wr%,wy%-wr% TO wx%+wr%,wy%-wr%:ARC#ch% TO wx%+wr%*1.5,wy%,PI
1837 END DEFine
```

```
1839 DEFine PROCedure Overcast(ch%,wx%,wy%,wr%)
1840 FILL#ch%,1:Cloud ch%,wx%,wy%,wr%
1841 LINE#ch%,wx%-wr%,wy%-wr% TO wx%+wr%,wy%-wr%:FILL#ch%,0
1842 END DEFine
```

```
1844 DEFine PROCedure Showers(ch%,wx%,wy%,wr%)
1845 Cloud ch%,wx%,wy%,wr%
1846 FOR i=-1 TO 1:LINE#ch%,wx%+wr%*i,wy%-wr%*.8 TO wx%+wr%*i,wy%-wr%*1.2
1847 FOR i=-1 TO 1:LINE#ch%,wx%+wr%*i,wy%-wr%*1.5 TO wx%+wr%*i,wy%-wr%*2
1848 END DEFine
```

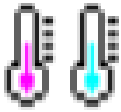
The Text with this is given as Heavy Rain

```
1850 DEFine PROCedure Downpour(ch%,wx%,wy%,wr%)
1851 Overcast ch%,wx%,wy%,wr%
1852 FOR i=-1 TO 1 STEP .4:LINE#ch%,wx%+wr%*i,wy%-wr% TO wx%+wr%*i,wy%-wr%*2
1853 END DEFine
```

```
1855 DEFine PROCedure Stormy(ch%,wx%,wy%,wr%)
1856 Downpour ch%,wx%,wy%,wr%:INK#ch%,7:FILL#ch%,1
1857 LINE#ch%,wx%,wy% TO wx%-wr%,wy% TO wx%,wy%+wr% TO wx%+wr%/2,wy%+wr%
1858 LINE#ch% TO wx%,wy%+wr%/2 TO wx%+wr%,wy%+wr%/2 TO wx%-wr%/4,wy%-wr%
1859 LINE#ch% TO wx%,wy%:FILL#ch%,0:INK#ch%,0
1860 END DEFine
```

```
1862 DEFine PROCedure Snow(ch%,wx%,wy%,wr%)
1863 Cloud ch%,wx%,wy%,wr%
1864 CIRCLE#ch%,wx%-wr%*.8,wy%-wr%*1.5,wr%/6:CIRCLE#ch%,wx%,wy%-wr%,wr%/6
1865 CIRCLE#ch%,wx%+wr%*.8,wy%-wr%*1.5,wr%/6:CIRCLE#ch%,wx%,wy%-wr%*2,wr%/6
1866 END DEFine
```

Bitmap Thermostat Symbols with Colours shown for High & Low.



```
1868 DEFine PROCedure Therm(wx,wy,wc)
1869 RESTORE 1892:FOR i=1 TO 18:READ a,b,c,d,e:BLOCK#ch%,a,b,wx+c,wy+d,e
1870 DATA 3,1,3,0,0,  1,8,2,1,0,  1,8,6,1,0,  2,1,1,9,0,  2,1,6,9,0
1871 DATA 1,4,0,10,0, 1,4,8,10,0, 2,1,1,14,0, 2,1,6,14,0, 3,1,3,15,0
1872 DATA 3,13,3,1,7, 7,3,1,10,7, 1,8,4,6,wc, 3,2,3,11,wc
1873 DATA 2,1,8,2,0,  2,1,8,4,0,  2,1,8,6,0,  2,1,8,8,0
1874 END DEFine
```

```
DEFine PROCedure Health(wx,wy)
INK#ch%,0:ARC#ch%,wx,wy-1 TO wx+3.6,wy,-PI TO wx+7.4,wy-1,-PI
LINE#ch% TO wx+5.6,wy-4.2:ARC#ch% TO wx+2,wy-4.2,-PI/3:LINE#ch% TO wx,wy-1
LINE#ch%,wx+1.2,wy-1 TO wx+1.8,wy-1 TO wx+2.2,wy TO wx+3.2,wy-3.5
LINE#ch% TO wx+4,wy-1 TO wx+6,wy-1:INK#ch%,0
END DEFine
```

```
DEFine PROCedure Shop(wx,wy)
RESTORE 1907:FOR i=1 TO 20:READ a,b,x,y,c:BLOCK#ch%,a,b,wx+x,wy+y,c
DATA 3,1,20,0,0, 2,1,19,1,0, 1,6,19,2,0, 1,6,18,6,0, 1,5,0,4,0
DATA 18,1,0,3,0, 18,1,0,5,0, 18,1,0,7,0, 16,1,2,9,0, 14,1,3,11,0
DATA 1,5,3,4,0, 1,5,6,4,0, 1,5,9,4,0, 1,5,12,4,0, 1,5,15,4,0
DATA 1,5,18,4,0, 3,2,2,12,0, 1,4,3,11,0, 3,2,16,12,0, 1,4,17,11,0
END DEFine
```

```
DEFine PROCedure Goods(wx,wy)
RESTORE 1915:FOR i=1 TO 13:READ a,b,x,y,c:BLOCK#ch%,a,b,wx+x,wy+y,c
DATA 4,5,0, 6,0, 8,8,3,2,0, 3,3, 4, 3,7, 20,10,12,0,0
DATA 5,2,4,11,0, 3,4,5,10,0, 5,2,25,11,0,  3, 4,26,10,0
DATA 15,1,9,10,0, 3,1,30,10,0, 6,1,16,2,7, 8,1,18,4,7, 8,1,20,6,7
END DEFine
```

Also used on Address page

```
DEFine PROCedure Post(wx,wy)
LINE#ch%,wx+1,wy+4 TO wx+5,wy+4 TO wx+2.5,wy+3 TO wx+1,wy+4
LINE#ch%,wx+1,wy+4 TO wx,wy+1 TO wx+4,wy+1 TO wx+5,wy+4
LINE#ch%,wx+2,wy+.5 TO wx+2,wy TO wx+6.5,wy TO wx+6.5,wy+3 TO wx+5,wy+3
LINE#ch%,wx+6.5,wy TO wx+7.5,wy+2 TO wx+7.5,wy+5 TO wx+6.5,wy+3
LINE#ch%,wx+7.5,wy+5 TO wx+4,wy+5 TO wx+2.5,wy+4
END DEFine
```

```
DEFine PROCedure Trash(wx,wy)
RESTORE 1930:FOR i=1 TO 8:READ a,b,x,y,c:BLOCK#ch%,a,b,wx+x,wy+y,c
DATA 5,1,5,0,0, 9,1,3,1,0, 15,1,0,2,0, 13,11,1,3,0, 11,1,2,14,0
DATA 1,7,3,5,7, 2,8,7,5,7, 1,7,11,5,7
END DEFine
```

```
DEFine PROCedure Bill(wx,wy)
LINE#ch%,wx-4.5,wy TO wx,wy+2 TO wx+4,wy
LINE#ch%,wx-3.5,wy TO wx-3.5,wy-4 TO wx+3,wy-4 TO wx+3,wy
CIRCLE#ch%,wx-1.8,wy-.8,.8:CIRCLE#ch%,wx-1.8,wy-2.5,.8
LINE wx,wy-2.8 TO wx,wy-1:POINT wx,wy-.5
LINE wx+1,wy-2.8 TO wx+1,wy+.2:LINE wx+2,wy-2.8 TO wx+2,wy+.2
END DEFine
```

Used on Address Page

```
DEFine PROCedure Phone(wx,wy)
RESTORE 1944:FOR i=1 TO 12:READ a,b,x,y,c:BLOCK#ch%,a,b,wx+x,wy+y,c
DATA 19,24,0,0,0, 17,16,1,3,7, 3,2,8,21,7, 1,4,9,20,7
DATA 3,10,7,6,2, 2,3,5,6,2, 2,3,5,13,2, 4,1,8,1,7
DATA 3,10,10,7,4, 1,8,13,8,4, 2,3,8,7,4, 2,3,8,14,4
END DEFine
```