



Putting Your **Commodore 64** to Work

by **Chris Callender**



**Word processing,
personal database,
spreadsheet calculations,
home accounts and more!**

Putting Your Commodore 64 To Work

by Chris Callender



Interface Publications, London and Melbourne

THE
FEDERAL
BUREAU OF INVESTIGATION
OF THE
DEPARTMENT OF JUSTICE
INVESTIGATION OF THE
ACTS OF
TERRORISM

REPORT OF THE
COMMISSION ON THE
ACTS OF TERRORISM
AND THE
FEDERAL BUREAU OF INVESTIGATION

UNITED STATES GOVERNMENT PRINTING OFFICE: 1975

Chris Callender is an experienced programmer in the sphere of games and utility software. Now, he turns his skills to the field of business applications and proves that imagination and know-how are equally advantageous at home and in business.

Fascinated by the possibilities of 'artificial intelligence', Chris enjoys adventure and exploration not just through computers, but also in his leisure pursuits. He is an enthusiastic yachtsman and sailboarder on the waters around his home by Scotland's Loch Long. Water-sports apart, he is also keen on motorcycling and camping.

First published in the UK by:
Interface Publications,
9-11 Kensington High Street,
London W8 5NP

Copyright © Chris Callender, 1983

ISBN 0 907563 56 2

The programs in this book have been included for their instructional value. They have been tested with care, but are not guaranteed for any particular purpose. Whilst every care has been taken, the publishers cannot be held responsible for any running mistakes which may occur.

ALL RIGHTS RESERVED

No use whatsoever may be made of the contents of this volume - programs and/or text - except for private study by the purchaser of this volume, without the prior written permission of the copyright holder.

Reproduction in any form or for any purpose is forbidden.

Books published by Interface Publications are distributed in the UK by WHS Distributors, St. John's House, East Street, Leicester LE1 6NE (0533 551196) and in Australia and New Zealand by PITMAN PUBLISHING. Any queries regarding the contents of this volume should be directed by mail to Interface Publications, 9-11 Kensington High Street, London W8 5NP.

1st Printing - December 1983

2nd Printing - May 1984

Typeset by Twickenham Printers, Albion Road, Twickenham, Middlesex.

Printed by J. W. Dunn (Printers) Ltd., Sutton, Surrey.

Note: - BE WARY OF []

TRY () INSTEAD.

IMPASSEMENT: →

Contents

Wordscreen	1
Bargraphs	5
Database	7
Cardfile	11
Home Accounts	14
Mailing List	18
Matrix	22
Planner	28
Calendar	32 ✓ () (NOT 1
SpreadCalc	34
Stock Control	38
Telephone Directory	43
Work Control	47
BOSS - Business Orientated Software System	52



1. Introduction
2. Background
3. Methodology
4. Results
5. Discussion
6. Conclusion
7. References
8. Appendix
9. Bibliography
10. Index

© 2000 by the Author. All rights reserved.

Typing In Programs

Control codes in listings are shown on the printouts as underlined characters. Here is a complete copy of them.

```
10 rem printer control codes
20 print "s - Home"
30 print "S - CLR"
40 print "g - Crsr Down"
50 print "Q - Crsr Up"
60 print "△ - Crsr Right"
→ 70 print "I - Crsr Left" (↑) AMEND ]
80 print "r - RUS On"
90 print "R - RUS Off"
100 print "e - White"
110 print "E - f1"
```

In addition to these, graphics have been listed as letters on the printer. So if you come across some strange characters in a print statement, look for the notes on graphics at the end of the program instructions.

Using Disc Drives

Programs which do not make use of the tape recorder will run without alteration on a disc system. In other programs, lines which include open 1,1,0 should have the statement replaced by open 1,8,0. In a similar manner, open 1,1,1 should be replaced by open 1,8,1.

THE HISTORY OF THE

... ..

... ..

... ..

... ..

... ..

Wordscreen

One of the foremost uses of a computer in every business must surely be word processing. Using Wordscreen, a word processing program, documents can be prepared quickly and easily, stored on cassette, and edited with the minimum of fuss.

When Wordscreen is run, the computer will display a flashing cursor. The first line of the document can be input and finished by pressing the Return key. If a line is started with an "*" character, then that line will be interpreted by Wordscreen as a command, instead of a document line. There are 10 valid commands which can be entered:

- ***exit** This command stops the program.
- ***load** After asking the name of the document to be loaded from tape, this command will load it from tape.
- ***save** This command will ask for the name of the document in the computer's memory, and then save the document to tape.
- ***edit** This command allows you to change any line of the document. You are first asked the number of the line, and then what you want to change that line to.
- ***address** This command will put your address at the start of a letter. You will have to put your address between lines 2500 and 2540, of course!
- ***centre** This command centres the last line that you typed in to the middle of the screen.
- ***see** This command lets you see the document in the computer's memory with line numbers.
- ***insert** This command allows you to insert a new line between two lines that have already been typed in. You are

asked which line you wish to insert after, and then what you want to insert.

***delete** This command deletes a line, after you supply its line number.

***print** This command prints the document in the computer's memory.

```
10 print chr$(14);chr$(8)"S_~~~~~  
_~~~~ppppppppppppWordscreen"  
20 for a=0to500:next:print "S"  
30 dim w$(100)  
40 l=0  
50 open 1,0,0:input#1,a$:close1:print  
60 if left$(a$,1)="x" then 110  
70 w$(l)=a$  
80 l=l+1  
90 if l>100 then print "Memory is full..  
":l=100  
100 goto 50  
110 for a=1 to len(a$)  
120 if mid$(a$,a,1)>="A" and mid$(a$,a,1  
)<="Z"then gosub 260  
130 next a  
140 if a$="*exit"then print "Bye-bye!":e  
nd  
150 if a$="*load" then gosub 1000:goto 5  
0  
160 if a$="*save" then gosub 1500:goto 5  
0  
170 if a$="*edit" then gosub 2000:goto 5  
0  
180_ if a$="*address"then gosub 2500:goto  
50  
190 if a$="*centre"then gosub 3500:goto  
50
```

```

200 if a$="*see"then gosub 4000:goto 50
210 if a$="*insert"then gosub 4500:goto
50
220 if a$="*delete"then gosub 5000:goto
50
230 if a$="*print"then gosub 5500:goto 5
0
240 print "Command not valid-try again"
250 goto 50
260 a$=left$(a$,a-1)+chr$(asc(mid$(a$,a,
1))-128)+mid$(a$,a+1,len(a$)):return
1000 input "Filename";f$
1010 open 1,1,0,f$
1020 for a=0 to 100:input#1,w$(a):next:1
input#1,l:close1
1030 return
1500 input "Filename";f$
1510 open 1,1,1,f$
1520 for a=0 to 100:print#1,w$(a):next:p
rint#1,l:close1
1530 return
2000 input "Edit which line";e
2010 print w$(e)
2020 open 1,0,0:input#1,w$(e):close1
2030 return
2500 w$(0)="          Y
our address"
2510 w$(1)="          Y
our address"
2520 w$(2)="          Y
our address"
2530 w$(3)="          Y
our address"
2540 w$(4)="          Y
our address"
2550 l=5:return

```

```

3500 c$=w$(i-1)
3510 s=(40-len(c$))/2
3520 for a=1 to s:c$=" "+c$:next
3530 w$(i-1)=c$
3540 return
4000 print "S"
4010 for t=0 to l
4020 print t;" ":"
4030 print w$(t)
4040 next
4050 return
4500 input "Insert after line no";i
4510 for t=l to i
4520 w$(t)=w$(t-1)
4530 next t
4540 Input "Insert what";w$(i+1)
4550 l=l+1:return
5000 input "Delete line no.";d
5010 for t=d to l
5020 w$(t)=w$(t+1)
5030 next t
5040 l=l-1:return
5500 open 4,4,6:print#4,1:close4:open 4,
6
5510 for a=0 to l:print#4,w$(a):print#4:
next a:close 4
5520 return

```

Bargraphs

This program will draw bargraphs, automatically scaling them to fit on the screen. Using bargraphs could not be easier - you simply type the vertical and horizontal headings in, followed by up to 15 numbers. Then you leave the Commodore 64 to make a graph of them. The graph is then drawn up on the screen.

```
10 poke 53280,0
20 poke 53281,0
30 print "Se"
40 input "vertical label";v$
50 input "horizontal label";h$
60 print
70 dim d(15)
80 for a=1 to 15
90 print"data item ";a;:input d(a)
100 next a
110 rem now correct for scale
120 m=0
130 for a=1 to 15
140 if d(a)>m then m=d(a)
150 next a
160 rem m is maximum value
170 s=20/m:rem s is scale factor
180 for a=1 to 15
190 d(a)=d(a)*s
200 next a
210 rem now draw graph
220 print"S:";for a=1 to len(v$)
230 printmid$(v$,a,1);"q:";
240 next a
```


Database

Using the program, Database, you will be able to store all your personal and business records on computer, and there will be no excuse for not being able to find any piece of information.

Data is stored by the computer in the form of a table. When the program is run, a 'menu' of options will be displayed, and you simply key in the option that you want. The options are:

- 1. Create file** When you press the 1 key, you will first be asked the name of the file of data that you are about to create. Then, you will be asked to key in the first column, followed by the second, and so on. To return to the main menu, type '@E' instead of a record; and to move on to the next column, type '@N'. There is a maximum of 50 rows by 50 columns available.
- 2. See file** This command will display the file that you have just typed in. Pressing the @ key will stop the display scrolling up, and pressing the @ key followed by the E key will return you to the main menu. Typing '@N' will move you on to the next column.
- 3. Edit file** This allows you to edit a file created using option 1. The computer will ask for the row and column to be edited, and will then ask you what you want to change it to.
- 4. Load file** This will ask you for the name of the file which is about to be loaded. You will then be able to load a previously stored file from cassette.
- 5. Save file** This will store the file which is in the computer's memory on the cassette.
- 6. Search file** Once a file has been created, any record in it can be called up on screen by typing the row and column headings under which it is found.
- 7. Exit** This command stops the program.

Note that in lines 1060, 1070 and 2080, the '-' symbol should be typed in as an '@' character.

```
10 print chr$(8);chr$(14);"Sxxxxxxxxxxxxxxxxx
   Database"
20 for a=0 to 500:next
30 print "S"
40 dim a$(50,50)
50 print spc(15);"Options"
60 print
70 print"Create file.....
   .....1":print
80 print"See file.....
   .....2":print
90 print"Edit file.....
   .....3":print
100 print"Load file.....
   .....4":print
110 print"Save file.....
   .....5":print
120 print"Search for Record.....
   .....6":print
130 print "      rKey In desired option
R"
140 get a$:if a$="" then 140
150 if a$<"1" or a$>"6"then 140
160 print"S":on val(a$) gosub 1000,2000,
3000,4000,5000,6000
170 print"S"
180 goto 50
1000 rem create file
1010 input "Sfilename ";file$
1020 c=0
1030 r=0
1040 print "S"
```

```

1050 print r;:open 1,0,0:input#1,1$:close1:print
1060 if 1$="e" then return
1070 if 1$="n" then r=50:goto 1090
1080 a$(c,r)=1$
1090 r=r+1
1100 if r<51 then 1050
1110 c=c+1:r=0
1120 if c<51 then 1040
1130 return
2000 input "Printout";p$
2010 if left$(p$,1)="y" or left$(p$,1)="Y" then open 4,4:cmd 4
2020 c=0
2030 r=0
2040 print "S"
2050 print "Column ";c,c+1,c+2
2060 print r;" ";a$(c,r),a$(c+1,r),a$(c+2,r)
2070 c$=""
*2080 get e$:if e$="=" then gosub 2170
2090 if c$="e" then return
2100 if c$="n" then r=50
2110 r=r+1
2120 if r<51 then goto 2060
2130 c=c+1:r=0
2140 if c<49 then 2040
2150 close4
2160 return
2170 get c$:if c$="" then 2170
2180 return
3000 input "Row";r
3010 input "Column";c
3020 print a$(c,r);" becomes ";:input a$(c,r)
3030 return

```

```

4000 input "Filename";file$
4010 open 4,1,0,file$
4015 for c=0 to 50:for r=0 to 50:input#4
.a$(c,r):next r:next c
4020 close 4
4030 return
5000 open 4,1,1,file$
5010 for c=0 to 50:for r=0 to 50:input#4
,a$(c,r):next r:next c
5020 close 4
5030 return
6000 input "Enter column heading";c$
6010 input "Enter row heading";r$
6020 c=-1
6030 for c1=0 to 50
6040 if a$(c1,0)=c$ then c=c1
6050 next c1
6060 r=-1
6070 for r1=0 to 50
6080 if a$(0,r1)=r$ then r=r1
6090 next r1
6100 if r=-1 and c=-1 then print "Row and
Column not found":goto 6000
6110 if r=-1 then print "Row not found"
:goto 6000
6120 if c=-1 then print "Column not found"
:goto 6000
6130 print "Record found=":print:print
6140 printa$(c,r)
6150 get a$:if a$="" then 6150
6160 return

```

SAVE "DDEK"

VERIFY "DDEK"

LOAD SHIFT RUN STOP. OR.

LOAD "DDEK"

Cardfile

Cardfile will replace a conventional card filing system, with the added advantage of easier, faster searching than even the most efficient of card filing systems.

When the program is run, a card will be shown which looks just like its paper counterpart – it even has lines on it to make it easier to read! Eight commands may be entered and these are outlined below:

input This command allows you to input a card. Simply type 'input', and then type on top of the card provided.

load This command loads the whole card filing system from tape.

save This command saves the entire card filing system to tape.

n This command moves on to the next card.

p This command moves back to the previous card.

c This command allows the user to move to a card, the number of which should be input.

search This command allows the user to input the title of a card. The computer will then search for, and display that card.

exit This command stops the program.

Note that in lines 60 and 90, the 'D' character is a shifted D, the '*' character is a 'Commodore W', and the number '4' is a 'Commodore Q'.

```
10 poke 53280,6:poke 53281,6:print "Se":d
im c$(100,8)
20 c=0
40 rem draw card
```

```

50 for a=0 to 8
60 print "xXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXX4"
70 print "x
      4"
80 next a
90 print "xXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXX4"
100 print "sggq card#";c,"title ";c$(c,0)
110 for a=1 to 8:print "q△";c$(c,a):next
a
120 print "ggq>";:open 1,0,0:input#1,d$:
print:close 1
130 if d$="input" then gosub 1000
140 if d$="load" then gosub 2000
150 if d$="save" then gosub 3000
160 if d$="n" then c=c+1
170 if d$="p" then c=c-1
180 if d$="c" then input c
190 if d$="search" then gosub 4000
210 if d$="exit" then print "Sbye-bye!":e
nd
220 print "S":goto 40
1000 rem input
1010 print "sgq","△△△△△";:gosub5000:c$(c
,0)=q$:print
1020 for a=1 to 8:print "q△";:gosub5000:c
$(c,a)=q$:print:next a
1030 return
2000 input "filename";f$
2010 open 1,1,0,f$
2020 for b=0 to 100
2030 for a=0 to 8
2040 input#1,c$(b,a)
2050 next a
2060 next b

```

```

2070 close1: return
3000 input "filename";f$
3010 open 1,1,1,f$
3020 for b=0 to 100
3030 for a=0 to 8
3040 print#1,c$(b,a)
3050 next a
3060 next b
3070 close1:return
4000 input "title";t$
4010 c1=-1
4020 for b=0 to 100
4030 if c$(b,0)=t$ then c1=b
4040 next b
4050 if c1=-1 then print "no such title"
:goto 4000
4060 c=c1:return
5000 q$=""
5010 get k$
5020 if k$=chr$(13) then print " ";:retur
n
5030 printk$;"_r_Rl";
5040 q$=q$+k$
5050 goto5010

```

Home Accounts

Home accounts will help you to keep track of your spending by producing a neatly laid out accounts sheet, showing all the year's income and expenditure. One of the greatest advantages of the program is that it can be customised to suit your own requirements.

The program is customised by changing lines 9000 and 9010. You must work out what you most often spend money on, and then put this in the data statements in these lines. There are only three minor limitations: the data must start with 'Income'; it should be ended with the word 'end' to tell the computer where the end is; and there is only room on the screen for about 20 entries (excluding 'end'). Example data statements have been listed at lines 9000-9010 to give you an idea of what is required.

When the program is run, it will first ask you which month you want to work on. Then, an accounts sheet will be printed on the screen and a '>' prompt will be displayed, showing you that the computer is expecting a command. There are four valid commands. These are:

new month This command allows you change to the month on which you are working.

input This command will ask you what you want to input. If you typed 'income', the computer would proceed to ask you what your income was for the month on which you are working.

load This command loads an accounts sheet from tape.

save This command saves an accounts sheet to tape.


```

330 next a
340 x=0:y=23:gosub 500:print ">";:open 1
,0,0:input#1,c$:close 1
345 x=0:y=23:gosub 500:print "
";
350 if c$="new month" or c$="n" then pri
nt"S":goto 50
360 if c$="input" or c$="i" then goto 40
0
370 if c$="load" or c$="l" then goto 540
380 if c$="save" then goto 580
390 goto 340
400 x=0:y=23:gosub 500:print "What ?";:op
en 1,0,0:input#1,w$:close 1
410 restore
420 d=0
430 for a=1 to c
440 read s$
445 if left$(s$,len(w$))=w$ then d=a
450 next a
460 if d=0 then x=0:y=23:gosub 500:print
'No such heading":goto 400
470 x=7+(m-s)*4:y=d:gosub 500:print "r
R"
475 x=0:y=23:gosub 500:print "
";
480 x=0:y=23:gosub 500:print "=";:open 1,
0,0:input#1,a(d,m):close 1
490 print "S":goto 70
500 print "s";
510 if y<>0 then for y1=0 to y:print "q";
:next y1
520 if x<>0 then for x1=0 to x:print "△";
:next x1
530 return
540 print "S":input "Filename":f$

```

```
550 open 1,1,0,f$
560 for a=1 to 20:for b=0 to 12:input#1,
a(a,b):next b:nexta
570 print"S":goto 70
580 print"S":input "Filename":f$
590 open 1,1,1,f$
600 for a=1 to 20:for b=0 to 12:print#1,
a(a,b):next b:nexta
610 print"S":goto 70
9000 data "Income","Mort.","Phone","Gas"
,"Elect.","Car","Insur.,"Rates"
9010 data "Clothes","Food","Savings","TV
Ren.,"HP"
9020 data "end"
```

Mailing List

One area where computers can be used to great effect is to store mailing lists. Once the mailing list is typed into the computer; it can be stored on cassette and recalled and printed, whenever it is required. This speeds up the process of sending a letter to everyone on a mailing list, and greatly reduces the effort required.

When the program is run, a menu will show you the options available. Each option has a number, and to select an option, you must type in that number. The seven available options are:

- 1. Add address to mailing list** You simply select option 1, and type the address. The mailing list can store up to 200 addresses.
- 2. Delete address from mailing list** This asks you for the first line of the address to be deleted, and then deletes that address.
- 3. Edit address** This again asks you for the first line of an address, but instead of deleting it, allows you to edit it.
- 4. See mailing list** This displays the mailing list on the screen.
- 5. Print mailing list** This prints the mailing list on the printer.
- 6. Load mailing list** This loads a mailing list from tape.
- 7. Save mailing list** This saves a mailing list to tape.

Note that the number '8' in line 4020 is a 'Commodore U' and the "" symbol is a 'Commodore I'.

```
10 poke 53280,6:poke 53281,6:print "Se";c  
hr$(14);chr$(8);"qqqqqqqqqqqqqqqqqqqq";
```

```

20 print "Mailing List"; for p=0 to 500; n
ext p
30 print "S"
40 dim a$(200,5)
50 print tab(10); "Options"; print tab(10);
#####"
60 print
70 print
90 print "Add address to mailing list....
.....1"
100 print
110 print "Delete address from mailing li
st.....2"
120 print
130 print "Edit address.....
.....3"
140 print
150 print "See list.....
.....4"
160 print
170 print "Print address.....
.....5"
180 print
190 print "Load mailing list.....
.....6"
200 print
210 print "Save mailing list.....
.....7"
220 print
230 print
240 print "rKey in desired optionR";
245 print " ";
250 get b$; if b$="" then 250
260 if b$<"1" or b$>"7" then goto 250
270 print b$
280 for p=0 to 100

```

```

290 print "S"
300 on val(b$) gosub 1000,2000,3000,4000
,5000,6000,7000
310 print "S"
320 goto 50
1000 p1=0:f=0
1010 for p=0 to 200
1020 if f=0 and a$(p,1)="" then f=1:p1=p
1030 next p
1040 if f=0 then print "Memory Full":for
p=0 to 1000:next p:return
1050 for a=0 to 5
1060 open 1,0,0:input#1,b$:close 1
1070 print
1080 a$(p1,a)=b$
1090 next a
1100 return
2000 print "First line of address to be d
eleted":input d$
2010 for p=0 to 200
2020 if a$(p,0)=d$ then for a=0 to 5:a$(
p,a)="" :next a
2030 next p
2040 return
3000 print "First line of address to be e
dited"
3010 input d$
3020 for p=0 to 200
3030 if a$(p,0)=d$ then gosub 3060
3040 next p
3050 return
3060 for a=0 to 5
3070 print a$(p,a):open 1,0,0:input#1,x$:
close 1
3080 if x$="" then x$=a$(p,a)
3090 a$(p,a)=x$

```


Matrix

Matrix performs the operations of addition, subtraction and multiplication, with an order of up to 12 by 12. A matrix is a group of numbers in rows and columns, similar to a table. To add matrices, the corresponding elements of each matrix are simply added, and a similar rule applies to subtraction. Multiplication cannot, however, have such a simple definition; it is in multiplying matrices that the program will be most useful. Multiplying two 12 by 12 matrices would be very prone to error, time consuming and boring – two by two matrices are hard enough to multiply together!

When the program is run, a menu of options will be displayed. The various options are as follows:

- 1. Input a matrix** First, you must specify the name of the matrix that you wish to enter. The matrices are labelled A to Z. Then, you must input column 1, followed by column 2, and so on. To move on to the next column, you type 'N', and when you are finished, you type 'E'. This returns you to the main menu.
- 2. Edit a matrix** You must first input the name of the matrix that you wish to edit. Then you must enter the row and column of that matrix. Finally, you input what you want to change that element to.
- 3. Add/subtract/multiply two matrices** This option performs the various arithmetic operations on matrices. First, you enter the two matrices the operation is to be carried out on, and then you input the matrix that the result is to be stored in. If, for instance, you wished to multiply matrix A, by matrix Q and store the result in matrix M, you would type 'A' (Return) 'Q' (Return) and 'M' (Return). Then a 'sub-menu' will give you the option of addition, subtraction or multiplication. In this case,

you would select multiplication. There is a short delay, and then you return to the main menu.

4. Save matrix This option asks you which matrix you wish to save, and then saves it to tape.

5. Load matrix This option asks you which matrix you want to load, and then loads the matrix (which has been saved using option 4).

6. Print a matrix This option prints a matrix, or part of a matrix on the screen. The screen acts like a window on the matrix. Pressing the less than (<) key moves the window left, and pressing the greater than (>) key moves the window right. The Shift key is not required. Typing 'E' returns to the main menu.

```
10 poke 53280,6:poke 53281,6:printchr$(8
);chr$(14);"Seqqqqqqqqq~Matrices
20 for p=0 to 500:next p
30 print "S"
40 dim m(26,15,15)
50 print tab(10);"Options"
60 print
70 print
80 print
90 print
100 print
110 print
120 print "Input matrix.....
.....1"
130 print "Edit Matrix.....
.....2"
140 print "Add/Subtract/Multiply Matrix.
.....3"
150 print "Save a matrix.....
.....4"
160 print "Load a matrix.....
.....5"
```

```

170 print "See a matrix.....
.....6"
180 print "Clear matrix.....
.....7"
190 print "Exit program.....
.....8"
200 print
210 print
220 print"RKey in desired optionR"
240 get k$:if k$="" then 240
250 if k$<"1" or k$>"8" then 240
260 print"S"
270 on val(k$) gosub 1000,2000,3000,4000
,5000,6000,7000,8000
280 print"S"
290 goto 50
300 rem fn-convert(a$)
310 for a=1 to ien(a$)
320 if mid$(a$,a,1)>="A" and mid$(a$,a,1
)<="Z" then gosub 350
330 next a
340 return
350 a$=left$(a$,a-1)+chr$(asc(mid$(a$,a,
1))-128)+right$(a$,len(a$)-a)
360 return
1000 rem input
1010 input "Input which matrix(a-z)";m$
1020 a$=m$:gosub 300:m$=a$
1030 a=asc(m$)-64
1040 c=0
1050 r=0
1060 print"S"
1070 print "Column ";c
1080 print r;" ";:open 1,0,0:input#1,m$:
close1:print
1090 if m$="N" or m$="n" then r=15:goto
1120

```

```

1100 if m$="E" or m$="e" then r=15:c=15:
goto 1120
1110 m(a,c,r)=val(m$)
1120 r=r+1
1130 if r<16 then goto 1080
1140 c=c+1:r=0
1150 if c<16 then goto 1060
1160 return
2000 rem valdit
2010 input "Which Matrix";m$
2020 a$=m$:gosub 300:m$a=a$:m=asc(m$)-64
2030 input "Column,Row";c,r
2040 print m$(m,c,r);"=";;input l
2050 m$(m,c,r)=l
2060 return
3000 input "Matrix 1";a$
3010 input "Matrix 2";b$
3020 input "Result";c$
3030 print "1.Add ";a$;" and ";b$
3040 print "2.Subtract ";a$;" and ";b$
3050 print "3.Multiply ";a$;" and ";b$
3060 print
3070 print
3080 print "r_          Key on desired optio
n          R_"
3090 get d$:if d$="" then 3090
3100 if d$<"1" or d$>"3" then goto 3090
3110 on val(d$) gosub 3200,3400,3600
3120 return
3200 gosub 300:d=asc(a$)-64
3210 a$=b$:gosub 300:b=asc(a$)-64
3220 a$=c$:gosub 300:c=asc(a$)-64
3230 print "Adding...wait a minute"
3240 print:for x=0 to 15
3250 for r=0 to 15
3260 print "QCColumn ";x;" Row ";r;"t_"

```

```

3270 m(c,x,r)=m(d,x,r)+m(b,x,r)
3280 next r
3290 next x
3300 return
3400 gosub 300:d=asc(a$)-64
3410 a$=b$:gosub 300:b=asc(a$)-64
3420 a$=c$:gosub 300:c=asc(a$)-64
3430 print "Subtracting...wait a minute"
3440 print:for x=0 to 15
3450 for r=0 to 15
3460 print "QColumn ";x;" Row ";r;"t"
3470 m(c,x,r)=m(d,x,r)-m(b,x,r)
3480 next r
3490 next x
3500 return
3600 print "Multiplying...wait a minute"
3610 gosub 300:d=asc(a$)-64
3620 a$=b$:gosub 300:b=asc(a$)-64
3630 a$=c$:gosub 300:c=asc(a$)-64
3640 for x=0 to 15
3650 for r=0 to 15
3660 s=0
3670 for n=0 to 15
3680 s=s+m(d,n,r)*m(b,x,n)
3690 next n
3700 m(c,x,r)=s
3710 print "QColumn ";x;" Row ";r;"t"
3720 next r
3730 next x
3740 return
4000 input "Save which matrix";m$
4010 a$=m$:gosub 300:m=asc(a$)-64
4020 open 1,1,1,"Matrix "+m$
4030 for x=0 to 15
4040 for r=0 to 15
4050 print#1,m(m,x,r)

```

```

4060 next r
4070 next x
4080 close1:return
5000 input "Load which matrix";m$
5010 a$=m$:gosub 300:m=asc(a$)-64
5020 open 1,1,0,"Matrix "+m$
5030 for x=0 to 15
5040 for r=0 to 15
5050 input#1,m(m,x,r)
5060 next r
5070 next x
5080 close1:return
6000 input "See which matrix";m$
6010 a$=m$:gosub 300:m=asc(a$)-64
6020 c=0
6030 print"S"
6040 for r=0 to 15
6050 for x=c to c+3
6070 print tab[(x-c)*5];m(m,x,r);
6080 next x
6090 for a=0 to 25:next
6100 print:next r
6110 get d$:if d$="" then 6110
6120 if d$="," or d$("<" and c<>0 then c
=c-1
6130 if d$="," or d$(">)" and c<>12 then
c=c+1
6140 if d$("<")"e" then goto 6030
6150 return
7000 input "Clear which matrix";m$
7010 a$=m$:gosub 300:m=asc(a$)-64
7020 for x=0 to 15:for r=0 to 15:m(m,x,r
)=0:next r:next x:return
8000 print"Bye-Bye"
8010 end

```

Planner

Planner will tell you what you are doing at any time of the week. It works like a very detailed diary, into which you could enter everything you are going to do during a week.

When the program is run, a menu of options will be displayed. These options are:

- 1. Load week's appointments** This option will load the week's appointments, which have been saved (using option 2).
- 2. Save week's appointments** This command allows you to save the week's appointments to tape.
- 3. See/alter appointments** This command will ask you the day on which you want to work. It will then display the appointments that you have on that day, and ask you if you want to alter them. If you do, you will be asked what time of day you want to alter and finally, what you want to do at that time of day.
- 4. Exit** This option stops the program.

Note that the '#' symbol in lines 50 and 3160 should be input as the 'T' key pressed at the same time as the 'Commodore' key.

```
10 poke 53280,6:poke 53281,6:printchr$(8
);chr$(14);"Seqqqqqqqqqqqqqqqqqqqqq";
15 print"Planner"
20 for p=0 to 500:next p
30 print"S"
40 dim d$(7,19)
50 print tab(10);"Options":printtab(10);
"#####"
60 print
```

```

70 print
80 print "Load appointments form tape...
.....1"
90 print "Save appointments on tape.....
.....2"
100 print "See/alter any day.....
.....3"
110 print "Exit.....
.....4"
120 print
130 print
140 print
150 print "Key in desired option"
160 get a$: if a$="" then 160
170 if a$<"1" or a$>"4" then 160
180 print "S"
190 on val(a$) gosub 1000,2000,3000,4000
200 print "S"
210 goto 50
300 for l=1 to len(a$)
310 if mid$(a$,l,1)>="A" and mid$(a$,l,1
)<="Z" then gosub 340
320 next l
330 return
340 a$=left$(a$,l-1)+chr$(asc(mid$(a$,l,
1))-128)+right$(a$,len(a$)-l):return
1000 open 1,1,1,"planner"
1010 for a=0 to 7:for b=0 to 19:print#1,
d$(a,b):next b:next a:close 1:return
2000 open 1,1,0,"planner"
2010 for a=0 to 7:for b=0 to 19:input#1,
d$(a,b):next b:next a:close 1:return
3000 input "Which day ";a$
3010 d=8
3020 gosub 300
3030 if a$="monday" then d=1

```

```

3040 if a$="tuesday" then d=2
3050 if a$="wednesday" then d=3
3060 if a$="thursday" then d=4
3070 if a$="friday" then d=5
3080 if a$="saturday" then d=6
3090 if a$="sunday" then d=7
3100 if d=8 then print "What???:goto 30
00
3110 input "Copy to printer";p$;a$=p$;go
sub 300;p$=a$
3120 if left$(p$,1)="y" then open 1,4,7;
cmd 1
3130 restore
3140 for e=1 to 19
3145 get k$; if k$="h" then gosub 4010
3150 read t$
3160 print "#####
#####"
3170 printt$
3180 printd$(d,e)
3190 next e
3200 if left$(p$,1)="y" then close 1
3210 input "Alter ";a$
3220 gosub 300
3230 if left$(a$,1)<>"y" and left$(a$,1)
<>"n" then print "What??":goto 3210
3235 if left$(a$,1)="n" then return
3240 input "Time ";t$
3250 restore;f=0
3260 for e=1 to 19
3270 read s$
3280 if s$=t$ then f=e
3290 next e
3300 if f=0 then print "What??":goto 3240
3310 printd$(d,f);input d$(d,f)
3320 print "O.K."

```

```
3330 goto 3000
4000 print "Bye-Bye":end
4010 get k$:if k$="" then 4010
4020 return
9000 data "9.00"
9010 data "9.30"
9020 data "10.00"
9030 data "10.30"
9040 data "11.00"
9050 data "11.30"
9060 data "12.00"
9070 data "12.30"
9080 data "13.00"
9090 data "13.30"
9100 data "14.00"
9110 data "14.30"
9120 data "15.00"
9130 data "15.30"
9140 data "16.00"
9150 data "16.30"
9160 data "17.00"
9170 data "17.30"
9180 data "18.00"
```

* = ERROR. (SYNTAX)

NOTE: CALENDAR IS

ONE YEAR 'ADVANCED'
REDUCE BY ONE YEAR

e.g:-
① $\frac{1987}{1986}$

Calendar (WHEN ASKED ? YEAR.)

Calendar will produce a calendar for any year from 1901 onwards. Apart from finding out what day Christmas was on in 1904, the program could be useful for businesses, such as travel agents, which would benefit from being able to call up the calendar for any year required.

The program is easy to use - simply type in the year that you want and up comes a calendar. Pressing the 'h' key will stop the screen scrolling up.

Note that the 'C' in lines 170 and 260 should be input as a shifted C.

```

10 poke 53280,6
20 print "e"
* 30 input "year";y
40 if y<1900 then goto 30
( ) * 50 if y<>int(y) then goto 30
60 d=1
70 for a=1901 to y-1
( ) * 80 if a/4=int(a/4) and a/100<>int(a/100)
0] then d=d+365:goto 100
90 d=d+366
100 next a
110 rem d is no of days since 1st Jan
1900
( ) * 120 e=d-7*int(d/7)
130 for m=1 to 12
* 140 read m$,L (L = Lower case (not 1 or L)
150 if y/4=int(y/4) and y/100<>int(y/100)
( ) 00] and m$="february" then l=29

```

```
( ) 160 print tab(15);m$;" ";y
170 print "CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCC"
180 print "    mon tue wed thu fri sat
sun"
* 190 for a=1 to 1 (L = Lower case)(NOT 1 OR L)
200 get h$:if h$="h" then gosub 1000
( ) 210 print tab(e*4+5);a;:if e=6 then prin
t
220 e=e+1
230 if e=7 then e=0
240 next a
250 print
260 print "CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCC"
270 print:print:print
280 next m
290 end
300 data "january",31
310 data "february",28
320 data "march",31
330 data "april",30
340 data "may",31
350 data "june",30
360 data "july",31
370 data "august",31
380 data "september",30
390 data "october",31
400 data "november",30
410 data "december",31
1000 get h$:if h$="" then 1000
1010 return
```

SpreadCalc

SpreadCalc is a spreadsheet program which can be used for a great many applications. Imagine a piece of paper divided up into cells, each of which could have written in it a word, a number or a formula written in terms of other cells. A formula could, for example, be in cell (3,2) and say 'cell (1,1) * cell (2,2)'. The value of cell (3,2) would change if cell (1,1) or cell (2,2) was changed, and the command 'recalc' was typed in.

When the program is run, some of the blank cells will be displayed. It would be impossible to show all the hundreds of cells that there are, so the display must act as a 'window' on the cells. There are six commands which can be entered at the top of the screen. These are:

input After typing 'input', the prompt 'At?' will be displayed. When this prompt is displayed, you type the cell that you wish to input something into. If you wanted to input into the cell 5 along and 7 down, you would type '5,7'. Then you type in what you want to input, if it's a number or a word. You type '↑' if you require to input a formula and the prompt 'Formula' is displayed. You then type in the formula. To reference a cell, such a cell (9,3) in the formula, you would use 'n(9,3)'.

save This command saves all the cells to tape.

load This command loads all the cells from tape.

clear This command clears all the cells.

exit This command stops the program.

recalc This command re-calculates all the formulae on the spreadsheet.

It should be noted that commands should be entered in lower case.


```

320 if y2<>0 then for y1=0 to y2:print "q
";:next y1
330 return
420 rem get command
430 print "sq":for n=1 to 2:print "
";:next n
440 open 1,0,0:print "Q":input#1,i$:close 1
590 rem interpret command
600 if i$="input" then gosub 700
610 if i$="save" then gosub 1090
620 if i$="load" then gosub 1110
640 if i$="clear" then run
650 if i$="exit" then print "SBye bye!":end
660 if i$="recalc" then gosub 1200
670 return
700 rem input
710 print "sAt?":gosub 420
720 print "s  ":gosub 940
730 if p=0 then goto 700
740 a=val(left$(i$,p-1))
750 b=val(right$(i$,len(i$)-p))
760 if a<0 or a>25 or b<0 or b>25 then 7
10
770 x=a-2
780 y=b-10
790 if x<1 then x=1
800 if x>22 then x=22
810 if y<1 then y=1
820 if y>7 then y=7
830 gosub 150
840 print "sWhat?"
850 gosub 420
860 if i$="π" then gosub 1020:goto 880
865 s$(a,b)=i$:n(a,b)=val(i$)

```

```

880 goto 670
940 rem function equivalent to:
      p=instr$(i$,";")
950 p=0
960 p=p+1
970 p$=mid$(i$,p,1)
980 if p$<>";"and p<>len(i$) then 960
990 if p=len(i$) then p=0
1000 return
1020 rem formula
1030 print"sFormula"
1040 gosub 420
1050 i$=" "+i$
1055 gosub 940
1056 if p<>0 then i$=left$(i$,p-1)+","+r
ight$(i$,len(i$)-p):goto 1055
1060 s$(a,b)=i$:q$=i$:gosub 110
1070 n(a,b)=q
1080 return
1090 print"sFilename":gosub 420:open 1,1
,1,f$:for v=0 to 25:for w=0 to 25
1100 print#1,s$(v,w):nextw:nextv:close1:
return
1110 print"sFilename":gosub 420:open 1,1
,0,f$:for v=0 to 25:for w=0 to 25
1120 input#1,s$(v,w):nextw:nextv:close1:
gosub 1200:return
1200 rem recalc
1210 for p=0 to 1:for v=0 to 25:for w=0
to 25
1220 if left$(s$(v,w),1)<>" " then 1240
1230 q$=s$(v,w):gosub 110:n(v,w)=q
1240 next w:next v:next p
1250 return

```

Stock Control

Stock Control will help businesses, such as shops, by keeping track of the levels of stock of up to 300 different goods and listing those which have to be re-ordered. The program will also store the suppliers and re-order codes for each item.

When the program is run, the information on each of the goods will have to be input. This is done by selecting option 5 on the menu and pressing the Return key in response to the prompt 'Part Name?' You will then be asked for the re-order code, name of suppliers, re-order level, and the stocks currently held of the item. This process can be repeated until data on all the goods is entered. You can then use the other options on the menu.

Here are the options you have available on Stock Control:

- 1. Load current stocks** This command will load stocks which have been saved using option 2.
- 2. Save current stocks** This command will save what is in the computer to tape, so that it can be re-loaded using option 1.
- 3. List current stocks** This command will list all the data in the computer's memory, showing the items which should be re-ordered in reverse type.
- 4. List stocks to be re-ordered** This command will list all stocks to be re-ordered on a printer.
- 5. Change data on part** This command allows for the editing of the data in the computer's memory. It first asks for the part name to be edited, and then for the new part name, re-order code, supplier, re-order level and stock level. If the Return key is pressed in response to any of these questions, the computer will assume that there is no change to be made to that piece

of data.

6. Input number of parts sold This command is intended to be used every time something is sold. The part name, and then the number sold are to be entered.

7. Exit This command stops the program.

It should be noted that in line 4080, the '#' symbol should be entered as a 'Commodore T' and in line 5120, the '@' symbol should be entered as a 'Commodore Y'.

```
10 poke 53280,6:poke 53281,6:print "Segggg
ggggggggg~~~~~";chr$(14);chr$(8);
20 print "Stock Control":for p=0 to 500:n
exp
30 dim s$(300,4)
40 print "S"
50 print tab(10);"Options":printtab(10);
'#####'
60 print
70 print "Load current stocks.....
.....1"
80 print
90 print "Save current stocks.....
.....2"
100 print
110 print "List current stocks.....
.....3"
120 print
130 print "List stocks to be re-ordered..
.....4"
140 print
150 print "Change data on part.....
.....5"
160 print
170 print "Input no. of parts sold.....
.....6"
```

```

180 print
190 print "Exit program.....
.....7"
200 print
210 print "r.          Key in desired optio
n          R"
220 get a$
240 if a$<"1" or a$>"7" then 220
250 print "S"
260 on val(a$) gosub 1000,2000,3000,4000
.5000,6000,7000
270 goto 40
1000 open 2,1,0,"stock"
1010 for a=0 to 100
1020 for b=0 to 4
1030 input#2,s$(a,b)
1040 next b
1050 next a
1060 return
2000 open 2,1,1,"stock"
2010 for a=0 to 300
2020 for b=0 to 4
2030 print#2,s$(a,b)
2040 next b
2050 next a
2060 return
3000 for a=0 to 300
3010 if s$(a,0)=" " then 3090
3020 print "S"
3030 print "Item: ";s$(a,0)
3040 print "Order Code: ";s$(a,1)
3050 print "Suppliers: ";s$(a,2)
3060 print "Re-order level: ";s$(a,3)
3065 if val(s$(a,4))<val(s$(a,3)) then p
rint "rStocks left: ";s$(a,4);"R"

```

```

3070 if val(s$(a,4))>val(s$(a,3)) then p
rint"Stocks left:";s$(a,4)
3080 get a$:if a$="" then 3080
3090 next a
3100 return
4000 open 4,4:cmd 4
4005 for a=0 to 300
4010 if s$(a,0)= then 4090
4020 if val(s$(a,4))>val(s$(a,3)) then 4
090
4030 print"Item:";s$(a,0)
4040 print"Order Code:";s$(a,1)
4050 print"Suppliers:";s$(a,2)
4060 print"Re-order level:";s$(a,3)
4070 print"Stocks left:";s$(a,4)
4080 print "#####
#####"
4090 next a
4100 close4:return
5000 open 1,0,0:print"Part name":input#1
,a$:print
5020 d=-1
5030 for a=0 to 300
5040 if s$(a,0)=a$ and d=-1 then d=a
5050 next a
5060 if d=-1 then print "Part Name not f
ound":close 1:goto 5000
5065 a=d
5070 print "Part name:";s$(a,0)
5080 print"Order Code:";s$(a,1)
5090 print"Suppliers:";s$(a,2)
5100 print"Re-order level:";s$(a,3)
5110 print"Stocks left:";s$(a,4)
5120 print"@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@"
5130 print"New part name:";:input#1,a$:p
rint

```

```

5140 if a$="" then a$=s$(a,0)
5150 s$(a,0)=a$
5160 print "New order code:";:input#1,a$;
print
5170 if a$="" then a$=s$(a,1)
5180 s$(a,1)=a$
5190 print "New suppliers:";:input#1,a$;p
rint
5200 if a$="" then a$=s$(a,2)
5210 s$(a,2)=a$
5220 print "New Re-order level:";:input#1
,a$;print
5230 if a$="" then a$=s$(a,3)
5240 s$(a,3)=a$
5250 print "New Stock left:";:input#1,a$;
print
5260 if a$="" then a$=s$(a,4)
5270 s$(a,4)=a$
5280 close 1
5290 return
6000 open 1,0,0;print "Part name:";:input
#1,a$;print
6010 d=-1
6020 for a=0 to 300
6030 if s$(a,0)=a$ then d=a
6040 next a
6050 if d=-1 then print "No such part";c
lose 1;goto 6000
6060 a=d
6070 print "Stocks previously held:";s$(
a,4)
6080 print "Number sold ";input#1,n;close
1
6090 s$(a,4)=str$(val(s$(a,4))-n)
6100 return
7000 print "Bye-Bye";end

```

Telephone Directory

Telephone Directory will hold the telephone numbers of up to 100 people. By simply typing in a name, a telephone number will appear on the screen. No more turning over pages!

Telephone Directory is menu driven, and is simple to use. The options available are:

- 1. Input new number** This command will ask you for a new name and number; the computer will then store it in the directory.
- 2. Delete telephone number** This command will ask you whose number you wish to delete, and then delete it.
- 3. Search directory for number** This command will ask you for someone's name and once input, will enable the computer to display their number.
- 4. Load directory** This command allows the directory to be loaded into the computer's memory from tape.
- 5. Save directory** This command allows the directory to be saved from the computer's memory to tape.
- 6. Print all numbers** This command prints the whole directory.
- 7. Exit** This command stops the program.

Note that in line 4060 the '8' should be entered as a 'Commodore U' and the "" as a 'Commodore I'.

```
10 poke 53280,6:poke 53281,6:printchr$(8
);chr$(14);"Seqqqqqqqqqqqqqqqqqq";
20 print "Telephone Directory":for p=0 t
o 500:next p
```

```

30 print "S"
40 dim t$(100,1)
50 print tab(15); "Options": print tab(15);
  "#####"
60 print
70 print "Input new telephone number.....
  .....1"
80 print
90 print "Delete telephone number.....
  .....2"
100 print
110 print "Search Directory for number...
  .....3"
120 print
130 print "Load Directory.....
  .....4"
140 print
150 print "Save Directory.....
  .....5"
160 print
170 print "Print all numbers.....
  .....6"
180 print
190 print "Exit.....
  .....7"
200 print
210 print "L           Key in desired option
         R"
220 get k$
230 if k$="" then 220
240 if k$<"1" or k$>"7" then 220
245 print "S"
250 on val(k$) gosub 1000,2000,3000,5000
  ,6000,4000,7000
260 print "S": goto 50
1000 p=0: f=-1

```

```

1010 if t$(p,0)="" then f=p:p=100
1020 p=p+1
1030 if p<>101 then 1010
1040 if f=-1 then print "Memory Full":fo
r p=0 to 1000:nextp:return
1050 open 1,0,0:print "Name ";;input#1,t$
(f,0):print
1060 print "Number ";;input#1,t$(f,1):pri
nt:close1
1070 return
2000 open 1,0,0:print "Name to be deleted
";input#1,d$:print:close1
2010 s=-1
2020 for a=0 to 100
2030 if t$(a,0)=d$ then s=a
2040 next a
2050 if s=-1 then print "Name not found":
goto2000
2060 t$(s,0)="":t$(s,1)=""
2070 return
3000 open 1,0,0:print "Name ";;input#1,n$
:close 1:print
3010 s=-1
3020 for a=0 to 100
3030 if t$(a,0)=n$ then s=a
3040 next a
3050 if s=-1 then print "Name not found"
:goto 3000
3060 print "Name ";t$(s,0)
3070 print "Number ";t$(s,1)
3080 print:print:print "Press any key"
3090 get k$:if k$="" then 3090
3100 return
4000 input "Copy to printer":p$
4010 if left$(p$,1)="y" then open 4,4,2:
cmd 4

```


Work Control

Work Control will keep track of who has various jobs in a business. First, each job is assigned a code. Then the code, the person who has the job, and any comments are fed into the computer.

When the program is run, a security code must be entered. At present, this is 4y02f, but this may be changed by altering line 50. It is certainly not impossible to get round the security code, but it would probably stop someone who knew little about computers. Having entered the code, a menu of options is displayed. These are:

- 1. Load current work** This command loads all the work from the cassette recorder to the computer.
- 2. Save current work** This command saves all the work on cassette, so that it can be re-loaded using option 1.
- 3. List current work/job** After selecting this option, the question 'Work or job?' will be asked. If 'job' is input, the computer will ask for the job code, and list a particular job. If 'work' is entered, the computer will list all the work in its memory.
- 4. Alter records of any job** After selecting this option, the question, 'Search for which code?' will be asked. If a job code is typed in, the computer will search for that code; or if the Return key is pressed, the computer will search for the first free space. Then, it will ask for the new job code, who the job is with, and any new comments.
- 5. Job finished** This command deletes all records of a job once it is finished. The computer asks you for the code of the job to be deleted.
- 6. Exit** This command stops the program.


```

240 get a$
250 if a$<"1" or a$>"6" then 240
260 print "S"
270 on val(a$) gosub 1000,2000,3000,4000
.5000,6000
280 print "S"
290 goto 70
1000 open 2,1,0,"work"
1010 for a=0 to 400
1020 for b=0 to 2
1030 input#2,w$(a,b)
1040 next b
1050 next a
1060 close 2
1070 return
2000 open 2,1,1,"work"
2010 for a=0 to 400
2020 for b=0 to 2
2030 print#2,w$(a,b)
2040 next b
2050 next a
2060 close 2
2070 return
3000 input "Work or job";d$
3005 if d$="job" or d$="j" then 3090
3008 for a=0 to 400
3010 if w$(a,0)=" " then 3060
3020 print "Code:-";printw$(a,0)
3030 print "With:-";printw$(a,1)
3040 print "comments:-";printw$(a,2)
3050 print "#####"
#####
3055 get a$;if a$="h" then 3055
3060 next a
3070 get a$;if a$="" then 3070
3080 return

```

```

3090 input "Code ";c$
3100 for a=0 to 400
3110 if w$(a,0)=c$ then print "With:-";pr
intw$(a,1)
3120 if w$(a,1)=c$ then print "Comments:-
";printw$(a,2)
3130 next a
3135 print "Press any key"
3140 get a$; if a$="" then 3140
3150 return
4000 input "Code";c$
4010 b=-1
4020 for a=0 to 400
4030 if w$(a,0)=c$ and b=-1 then b=a
4040 nexta
4050 if b=-1 then print "Code not found"
:goto 4000
4060 open 1,0,0
4070 print "Code:-";:input#1,w$(b,0):prin
t
4080 print "With:-";:input#1,w$(b,1):prin
t
4090 print "Comments:-";:input#1,w$(b,2):
print:close1
4100 return
5000 input "Code";c$
5010 b=-1
5020 for a=0 to 400
5030 if w$(a,0)=c$ and b=-1 then b=a
5040 nexta
5050 if b=-1 then print "Code not found"
:goto 5000
5060 w$(b,0)="" :w$(b,1)="" :w$(b,2)=""
5070 return
6000 print "Bye-Bye!";end
6999 end

```

```
7000 poke 54296,0:poke 54276,0:poke 5427
7,0:poke 54278,0
7005 poke 54296,15:poke 54277,128:poke 5
4278,130:poke 54276,17
7010 poke 53280,0:poke 53281,0:print "P":
rem screen black
7020 poke 54273,20
7030 for a=0 to 500:next a
7040 poke 54273,100:poke 53280,1:poke 53
281,1:print "e"
7050 for a=0 to 500:next a:goto 7010
```

BOSS – Business Orientated Software System

BOSS shows how several programs can be chained together, and information passed between them – and for people who often switch from one program to another it will save a great deal of time.

There is only one small problem – it will take a long time to type in. When typing in any long program, it is far easier to input 10 or 20 lines at a time than to type in all 500 or more in one great programming session. Not only is this easier for you, but it also means that you will lose only a few lines in the event of a LOAD/SAVE error, or a power failure.

When you do finally get BOSS typed into your Commodore 64, and with great relief manage to gather up the strength to type 'RUN', you will be confronted with the program menu which will allow you to select the program you want to use. To select a program, you position the cursor over the first letter of the name of the program, and press the Return key. The required program will then be called.

The instructions for the program are as follows:

Writing Paper See Wordscreen.

Drawing Paper The cursor is moved using the cursor control keys. Any character pressed will appear at the cursor position. Pressing the 'f1' key will return you to the main program menu.

List Paper See Database.

Planner Paper See Planner.

Calculator Paper This makes the computer perform as a

calculator with 25 memories and 25 formulae. To input a formula, type 'input' followed by 'f n' where n is the number of the formula that you want to enter; you can then enter the formula. To reference a memory in a formula, e.g. memory 3, use m (3). To input a number to one of the memories, type 'input' followed by 'm n' where n is the memory number. To calculate a formula, type 'calculate' followed by the number of the formula to be calculated. The formulae may be loaded and saved using the commands 'load' and 'save'.

Accounts Paper See Home Accounts.

It should be noted that in lines 20 - 230 and line 4200 the following symbols represent the graphics listed below:

- O - Shift O
- 7 - Commodore Y
- P - Shift P
- % - Commodore H
- ' - Commodore N
- R - Shift R
- N - Shift N
- L - Shift L
- / - Commodore P
- : - Shift @
- _ - £
- 9 - Commodore U

Also, in lines 3140 and 3150, the "-" symbol should be input as an "@".

```
i0 poke 53280,6:poke 53281,6:print "e":dl
m s$(25,25):dim n(25,12):l=0
20 print chr$(142); "SQ7777777777P 077777
7777P 07777777777P"
```

```

30 print "%      RRR'   4      NG'  %RR
RRRRRRRR'"
40 print "%      RRR'   4      N G'  %RR
RRRRRRRR'"
50 print "%      RRR'   4      N G'  %RR
RRRRRRRR'"
60 print "%          '   4      N  G'  %RR
RRRRRRRR'"
70 print "%RRRRRRRRR'   4  N    G'  %RR
RRRRRRRR'"
80 print "%RRRRRRRRR'   4  N    G'  %RR
RRRRRRRR'"
90 print "%RRRRRRRRR'   4  ##### ' %RR
RRRRRRRR'"
120 print "L/////////:  L/////////:  L/
////////:/"
130 print:print "writing      drawing
      list"
140 print "  paper          paper
paper":print
150 print "0777777777P   0777777777P  07
77777777P"
160 print "% ????????'   %3*↑πcos( ' %_
|
170 print "% ????????'   %a)*s in(a)' %_
|
180 print "% ????????'   %          ' %_
|
190 print "% ????????'   %5*rnd(1)*' %_
|
200 print "% ????????'   %n/6 ;n=3*' %_
|
210 print "% ????????'   %p|π-3    ' %_
|
220 print "% ????????'   %          ' %_
|

```

```

230 print "L/////////: L/////////: L/
/////////:"
240 print "planner calculator ac
counts"
245 print " paper paper
paper"
250 geta$: if a$="" then 250
260 x=0:y=0
270 poke55296+40*x+y,x,0:a=peek(1024+40*x+y
x):poke 1024+40*x+y,x,160
290 get a$: if a$="" then 290
300 poke55296+40*x+y,x,1:poke 1024+40*x+y,
a
310 if a$="I" and x>0 then x=x-1
320 if a$="A" and x<39 then x=x+1
330 if a$="Q" and y<25 then y=y+1
340 if a$="Q" and y>0 then y=y-1
350 if a$<>chr$(13) then 270
360 if a<>23 and a<>4 and a<>12 and a<>1
6 and a<>3 and a<>1 then 270
370 if a=23 then goto 1000
380 if a=4 then goto 2000
390 if a=12 then goto 3000
400 if a=16 then goto 4000
410 if a=3 then goto 5000
420 if a=1 then goto 6000
1000 rem wordscreen
1002 printchr$(14); "S"
1005 open 1,0,0:input#1,a$:print:close 1
1010 if left$(a$,1)="x" then gosub 1035:
goto 1005
1015 s$(1,1)=a$
1020 l=l+1
1025 if l>25 then print"memory full":l=2
5
1030 goto 1005

```

```

1035 if a$="*print" then gosub 1085
1040 if a$="*load" then gosub 1105
1045 if a$="*save" then gosub 1135
1050 if a$="*exit" then goto 20
1055 if a$="*insert" then gosub 1165
1060 if a$="*centre" then gosub 1205
1065 if a$="*clear" then l=0:s$(0,1)=" "
1070 if a$="*address" then gosub 1230
1075 if a$="*edit" then gosub 1265
1080 if a$="*see" then gosub 1290
1082 goto 1005
1085 open 4,4,6:print#4,1:close 4:open 4
,4
1090 for a=0 to 25
1095 print#4,s$(a,1):next a
1100 close4:return
1105 input "Filename";f$
1110 open 1,1,0,f$
1115 for a=0 to 25
1120 print#1,s$(a,1)
1125 next a
1130 close1:return
1135 input "Filename";f$
1140 open 1,1,1,f$
1145 for a=0 to 25
1150 input#1,s$(a,1)
1155 next a
1160 close1:return
1165 input "After which line ";n
1170 n=n+1
1175 for t=l to n step -1
1180 s$(t+1,1)=s$(t,1)
1185 next t
1190 print "Insert what":input s$(n,1)
1195 l=l+1
1200 return

```

```

1205 c$=s$(l-1,1)
1210 c=40-len(c$):c=c/2
1215 for s=1 to c:c$=" "+c$:next s
1220 s$(l-1,1)=c$
1225 return
1230 s$(0,1)="
Your Address"
1235 s$(1,1)="
Your Address"
1240 s$(2,1)="
Your Address"
1245 s$(3,1)="
Your Address"
1250 s$(4,1)="
Your Address"
1255 l=5
1260 return
1265 input"Which line":n
1270 prints$(n,1)
1275 print"becomes"
1280 open 1,0,0:input#1,s$(n,1):print:cl
ose1
1285 return
1290 for a=0 to l
1295 printa;" ":"prints$(a,1)
1300 nexta
1310 return
2000 rem drwing paper
2010 print"S"
2020 x=0:y=0
2030 a=peek(1024+40*y+x):b=peek(55296+40
*y+x):x1=x:y1=y
2040 poke 1024+40*y+x,160:poke 55296+x+4
0*y,1
2050 get a$:if a$="" then 2050
2060 if a$<>"q" and a$<>"Q" and a$<>"△"
and a$<>"┘" then gosub 2130:x=x+1:a=-1

```

```

2070 if a$="g" and y<23 then y=y+1
2080 if a$="Q" and y>0 then y=y-1
2090 if a$="A" and x<39 then x=x+1
2100 if a$="I" and x>0 then x=x-1
2105 if a$="E" then goto 20
2110 if a<>-1 then poke 1024+x1+40*y1,a:
poke 55296+x1+40*y1,b
2120 goto 2030
2130 poke 781,y:poke782,x:sys 65520:prin
ta$
2140 return
3000 rem list paper
3005 print "S";chr$(14)
3010 printtab(10);"Options":printtab(10)
;"88888888"
3015 print
3020 print "Create file....."
.....1"
3025 print
3030 print "See file....."
.....2"
3035 print
3040 print "Save file....."
.....3"
3045 print
3050 print "Load file....."
.....4"
3055 print
3060 print "Save file....."
.....5"
3065 print
3070 print "Search for record....."
.....6"
3075 print
3080 print "Return to programs menu....."
.....7"

```

```

3085 print
3090 print "r          Key in desired optio
n          R"
3095 get k$:if k$="" then 3095
3100 if k$<"1" or k$>"7" then 3095
3105 print "S":on val(k$) goto 3110,3180,
3245,3265,3285,3300,20
3110 input "Filename";f$
3115 c=0
3120 r=0
3125 print "S"
3130 print "Column ";c
3135 print r;" "":open 1,0,0:input#1,i$:
print:close1
3140 if i$="-e" then c=25:r=25:goto 3155
3145 if i$="-n" then r=25:goto 3155
3150 s$(c,r)=i$
3155 r=r+1
3160 if r<>26 then 3135
3165 c=c+1:r=0
3170 if c<>26 then 3125
3175 goto 3000
3180 c=0:r=0
3185 print "SColumn ";c
3190 print r;s$(c,r)
3195 for a=0 to 100:next a
3200 get a$
3205 if a$="n" then r=25
3210 if a$="e" then r=25:c=25
3215 if a$="h" then gosub 3235
3220 r=r+1:if r<>26 then 3190
3225 r=0:c=c+1:ifc<>26 then goto3185
3230 goto 3000
3235 getk$:if k$="" then 3235
3240 return
3245 input "Column";c

```

```

3250 input "Row ";r
3255 input "becomes";s$(c,r)
3260 return
3265 input "Filename";f$
3270 open 1,1,0,f$
3275 for c=0 to 25:for r=0 to 25:input#1
,s$(c,r):next c:next r:close 1
3280 goto 3000
3285 open 1,1,1,f$
3290 for c=0 to 25:for r=0 to 25:print#1
,s$(c,r):next c:next r:close 1
3295 goto 3000
3300 input "Column heading";c$
3305 input "Row heading";r$
3310 c=-1:r=-1
3315 for x=0 to 25
3320 if s$(x,0)=c$ then c=x
3325 if s$(0,x)=r$ then r=x
3330 next x
3335 if c=-1 and r=-1 then print"Column
& Row not found":goto 3300
3340 if c=-1 then print "Column not foun
d":goto 3300
3345 if r=-1 then print "Row not found":
goto 3300
3350 print"Record found:"
3355 print:print
3360 prints$(c,r)
3365 print:print
3370 print"Press any key"
3375 get k$:if k$="" then 3375
3380 goto 3000
4000 rem planner paper
4005 printchr$(14);""
4010 printtab(10);"Options"
4015 print

```

```

4020 print "Load weeks appointments from
tape.....1"
4030 print "Save appointments on tape....
.....2"
4035 print "See/Alter any day.....
.....3"
4040 print "Return to programs menu.....
.....4"
4045 print:print:print
4050 print "Key in desired option"
4055 get k$:if k$="" then 4055
4060 if k$<"1" or k$>"4" then 4055
4065 on val(k$) goto 4070,4085,4130,20
4070 open 1,1,0,"planner"
4075 for a=1 to 7:for b=1 to 19:input#1,
s$(a,b):nextb:nexta:close 1
4080 goto 4000
4085 open 1,1,1,"planner"
4090 for a=1 to 7:for b=1 to 19:print#1,
s$(a,b):nextb:nexta:close 1
4095 goto 4000
4100 rem lower case subroutine
4110 for l=1 to len(a$)
4115 if mid$(a$,l,1)>="A" and mid$(a$,l,
1)<="Z" then gosub 4125
4120 next l:return
4125 a$=left$(a$,l-1)+chr$(asc(mid$(a$,l
,1))-128)+right$(a$,len(a$)-l):return
4130 input "Which day":a$
4135 d=8
4140 gosub 4100
4145 if a$="monday" then d=1
4150 if a$="tuesday" then d=2
4155 if a$="wednesday" then d=3
4160 if a$="thursday" then d=4
4165 if a$="friday" then d=5

```



```

5030 nexta:print
5040 print "memories:";for a=1 to 25
5050 print "m";a;"=";n(a,1)
5060 nexta
5070 open 1,0,0:print ">";:input#1,a$:pri
nt:close1
5080 if a$="list" then goto 5000
5090 if a$="clear" then for a=1 to 25:s$
(a,1)="" :n(a,1)=0:next a:goto 5000
5100 if a$="calculate" then goto 5160
5105 if a$="exit" then goto 20
5110 input "what";w$
5120 n=val(right$(w$,2))
5130 if left$(w$,1)="m" then open 1,0,0:
print "=";:input#1,n(n,1):close1:goto5000
5140 if left$(w$,1)<>"f" then goto 5110
5150 print "f";n;"=";:open 1,0,0:input#1
,s$(n,1):print:close 1:goto 5000
5160 input "calculate which formula";k
5170 f$=s$(k,1)
5180 a=0
5190 a=a+1
5200 if mid$(f$,a,1)="m" and mid$(f$,a+1
,1)="(" then gosub 5280
5210 if a<>len(f$) then goto 5190
5220 print "Sqqqqqr=";f$;":goto 5240";
5230 poke 631,13:poke 198,1:print "QQ";:e
nd
5240 print "S"
5250 print "result=";r
5260 get a$:if a$="" then 5260
5270 goto 5000
5280 n$=mid$(f$,a+2,2)
5290 n$="n("+n$+" ,1"
5300 f$=left$(f$,a-1)+n$+right$(f$,len(f
$)-a-3)

```

```

5310 return
6000 print "S";chr$(14)
6005 input "Month(1-12)";m
6010 if m<1 or m>12 or m<>int(m) then 60
05
6015 m$="Jan:Feb:Mar:Apr:May:Jun:Jul:Aug
:Sept:Oct:Nov:Dec"
6020 b$=": : : : : : : "
6025 if m<=6 then m$=left$(m$,24)
6030 if m>=7 then m$=right$(m$,24)
6032 print "S"
6035 restore:for z=1 to 19:readx$:nextz
6040 c=0
6045 read s$
6050 if s$="end" then 6065
6055 prints$;tab(7);b$:c=c+1
6060 goto 6045
6065 if m<=6 then s=1
6070 if m>=7 then s=7
6075 for a=s to s+5
6080 for b=1 to c
6085 poke 781,b:poke 782,7+(a-s)*4:sys 6
5520
6090 if n(b,a)<>0 then printn(b,a)
6095 next b
6100 next a
6105 poke 781,c+1:poke 782,0:sys 65520:p
rint "TOTAL"
6110 for a=s to s+5
6115 t=n(1,a)
6120 for b=2 to c
6125 t=t-n(b,a)
6130 next b
6135 poke 781,c+1:poke 782,7+(a-s)*4:sys
65520:printt
6140 next a

```

```

6145 print "△";:open 1,0,0:input#1,c$:pri
nt:close 1
6150 if c$="new month" or c$="n" then 60
00
6155 if c$="input" then 6175
6160 if c$="load" then 6225
6165 if c$="save" then 6235
6170 if c$="exit" then 20
6175 input "QWhat          ]]]]]]]]]]]";w$
6180 restore:for a=1 to 19:read x$:nexta
6185 d=0:for a=1 to c
6190 read s$
6195 if s$=w$ then d=a
6200 next a
6205 if d=0 then 6175
6210 poke 781,d:poke 782,8+(m-s)*4:sys 6
5520:print "r   R"
6212 poke 781,c+2:poke 782,0:sys 65520
6215 print "=          ]]]]]]]]]]]]]";
:open 1,0,0:input#1,n(d,m):print:close1
6220 print "S":goto6035
6225 open 1,1,0,"accounts"
6230 for a=0 to 25:for b=0 to 12:input#1
,n(a,b):next b:nexta:close1:goto6035
6235 open 1,1,1,"accounts"
6240 for a=0 to 25:for b=0 to 12:print#1
,n(a,b):next b:nexta:close1:goto6035
6250 data "Income","Mort.","Phone","Gas"
,"Elect.","Car","Insur.","Rates"
6255 data "Cheques","Clothes","Food","Sa
vings","TV Ren.","HP","end"

```


Now you can put your Commodore 64 to work, with the aid of this great collection of handy programs from Chris Callender.

If you want to use your computer as a word processor, we have the program, WORDSCREEN, to do it for you. You can keep your accounts in order with the HOME ACCOUNTS program, organise your life with PLANNER and keep your numbers under control with TELEPHONE DIRECTORY.

The 15 major programs in this book will allow you to employ your Commodore 64 in a wide variety of ways. The final program, BOSS (Business Orientated Software System) gives you a massive integrated program which brings together the most useful programs from the book. BOSS is organised to allow you to share data between programs, and switch easily and quickly from task to task.

Chris Callender has done a great job, so now you can, indeed 'put your Commodore 64 to work'.

Another great book from

INTERFACE
PUBLICATIONS 

£4.95

ISBN 0-907563-56-2



9 780907 563563