AUCKLAND COLOUR GENIE USERS GROUP

MEETING HELD ON 28 MAY, 1984

A very good attendance at this meeting - 45 people came, which really filled the hall. This is marvellous, it certainly shows that 'us Genie people' really want to learn about our machine.

Andy and Ken showed the Group 'Peeks' and 'Pokes' a synopsis of which is included at the end of this report.

We should now have a mention in the 'Club Contacts' section of 'Bits and Bytes' - a letter was written at the beginning of this month.

A member also thought it would be a good idea to mention the magazines that occasionally had articles about the Colour Genie, as not too many magazines feature our machine. Ones known are:

(a) Computing Today (monthly magazine)
(b) Personal Computing News (weekly magazine)

If anyone knows of any other magazines, please let us know.

Re New Roms New roms will be available soon from the Group for around $12.00 to $15.00 each. These Roms contain the disk commands and will be available from Andy and Ken. More on this in the next Newsletter.

Can anyone help one of our members; he wishes to know what [A3 42] in Super 80 programs means, both square brackets and figures. If anyone can help, please contact Ron Burton, Box 208, Te Aroha, or your Committee.

Other Possible Software - this time from New Zealand

We have a friend of one of our members, a Mr Allan Clarke, who has quite a few TRS80/System 80 programs that can be converted on to the Colour Genie, with quite a bit of effort involved, and he would be willing to sell these programs at probably between $10.00 and $15.00, certainly no more than $15.00. Under these circumstances, he would need an indication of what interest there would be in his programs, so I am including a list of what he could convert.

Compilers
Word Processor (basic, but with M/C routine for data saving)
Monitors
Statistics
Serial Port Driver (up and running)
Data Base
Editor Assembler

Disassembler
PILOT language
Program compressor
PERT
Ledger Keeper
General Utility (screen dump, move memory, auto edit, recover BASIC program, merge)
Program Search

I feel personally, that these utility programs would be well worth having. The same sort of programs are going to cost equally as much, if not more, than Allan's programs. However, I do feel that we would need such programs NOW, not in the future, which is the one benefit with the overseas programs, they are ready now. So, if we can get some idea on who wants what, we can get Allan cracking on producing these programs for the Genie without delay!!

Please do remember, though, that your Committee members are working their tails off to get this software off the ground, and we can do nothing without you all getting in touch and letting us know your software preferences. So come on, stop reading this for a moment, and write me a note.

Software

A big thank you to the 10 people who have so far responded to my request for a note on what software everyone is interested in. And everyone else, please put pen to paper.

We have now heard from J. Brier (two machine code arcade type games) and Gumboot Software (see software pages in last month's newsletter). Both of these software houses are willing to let us become the New Zealand agents for their software, so we are going ahead and ordering the masters from these two companies.

We will let you know for sure later, but prices should be from $9.00 to $12.00 per tape.

There is also a couple of programs to be added to last month's list:

Gumboot Software

40. Flying Bytes — Flight simulator for a 32K machine - supposed to be very good.

41. Droids — Arcade game. (Using a laser tank, droids and mines have to be cleared. Original and great fun,so the review in the U.K. magazine goes.)

We have also ordered a Technical Manual from Gumboot as these are now available. We will then be able to photocopy them and charge accordingly.

Gumboot are also willing to let us have quantities of the U.K. magazine - at a reasonable price, we are awaiting the exact cost.

So far, we have not heard from any of the other software houses, and I have sent out a further 17 letters to other software houses also advertising in the magazine.

Well, that's about it for this month, remember that the next meeting is on:-

25 June 1984

and hopefully we will see at least 50 people there.

- Nola Huggins

PEEKS AND POKES

As I promised, the following will cover what I spoke about at our last meeting. The first program looked at, scanned the keyboard memory and responded when a Y or an N was pressed.

For those of you who don't have a map of the Keyboard Memory System, there is a copy attached.

The first program was:

```
10  CLS:G=PEEK(-2046):H=PEEK(-2040)
20  IF G AND 64 THEN PRINT@ 500,"NO" ELSE
    IF H AND 02 THEN PRINT@ 500,"YES" ELSE GOTO 10
30  FOR X=1 TO 500:NEXT:GOTO 10
```

In line 10 G is assigned a value from Memory location(-2046) or &HF802. This val ue can be from 0 to 255 and the value that G assumes will depend on the key(s) depressed on the computer keyboard. The same applies to H, the only difference being the memory location being looked at. In this case (-2040) &HF808.

In line 20, using an IF-THEN test, we establish whether the required keyboard input was made or not. If it was, then we print a response to show the system working. If there is no input from the keyboard then the program goes back to line 10 to look again. The delay loop in line 30 just keeps the response on the screen so we get to see it.

The tests in line 20 could cause branching to subroutines or returns to a main program instead of how it appears in this program which was for demonstration purposes only.

Disabling Break

Poke 16396,23

- To re-enable - Poke 16396,201

(it is a good idea to do this at the end of a program if BREAK was disabled at the start of the program.)

Cursor

To change the cursor - Poke 16410,(value)

Value	1-7	static	1 =	(block)
			7 = _	(underline)
	64-71	fast	(64 same as 1)	
			(71 same as 7)	
	96-103	slow	(96 same as 1)	
			(103 same as 7)	

Low Res Screen

This screen starts at &H4400 (17408 dec) and finishes at &H47FF (18431) but locations after &H47BF (18367) are not displayed.

To see if something is printed on the screen PEEK the required location and test to see if the value returned is what you were looking for. More on this later in the second program.

High Res Screen

Starts at &H4800 (18432 dec)
Stops at &H57FF (22527 dec)

Disabling The Keyboard

This could be used to speed up sorting or mathematical calculations during a program where no keyboard input is required. Once the computations are complete, re-enable the keyboard to continue program execution.

Poke 16405,0 (disable)
Poke 16405,1 (able)

Second Program

```
10  CLS:X=17408:Y=&HAC00
20  G=PEEK(-1984)
30  IF G AND 64 THEN Z=X:X=X+1:C=4
40  IF G AND 32 THEN Z=X:X=X-1:C=6
50  IF G AND 16 THEN Z=X:X=X+40:C=1
60  IF G AND 08 THEN Z=X:X=X-40:C=5
70  IF (X <17408 OR X >18367) THEN X=7
80  POKE X,199:POKE X+7,C
90  V=PEEK(17900):IF V=199 THEN CLS
100 GOTO 20
```

The program was used to show the keyboard memory and screen memory can be used by PEEKing and POKing to do something on the screen.

Line 10 - Sets X to first screen location. Sets Y to the colour RAM area so that the colour of the object to be printed at X can be set during the program.

Line 20 - Scans the keyboard, namely the arrow keys, all other things being ignored.

Lines 30-60 - determine which arrow key was pressed, adjusts the value of X and sets a new colour if required.

Line 70 - tests to see if X is less than or greater than the text screen size. If you don't do this, disastrous things can happen. (Try taking it out and see!!)

Line 80 - Pokes character 199 to the screen at location X and sets the colour of that location.

Line 90 - Here we are looking at the screen at a specific location. If that location has a character of 199 (ASCII) printed on it, then the test continues to clear the screen.

Line 100 - returns to 20 to start again.

This is just one simple use of this type of programming. I used a similar technique to change the viewing direction in Haunted House. Have a look and see if you can understand all that is happening.

Blanking Lines in a Program

As an example I will blank the following line:

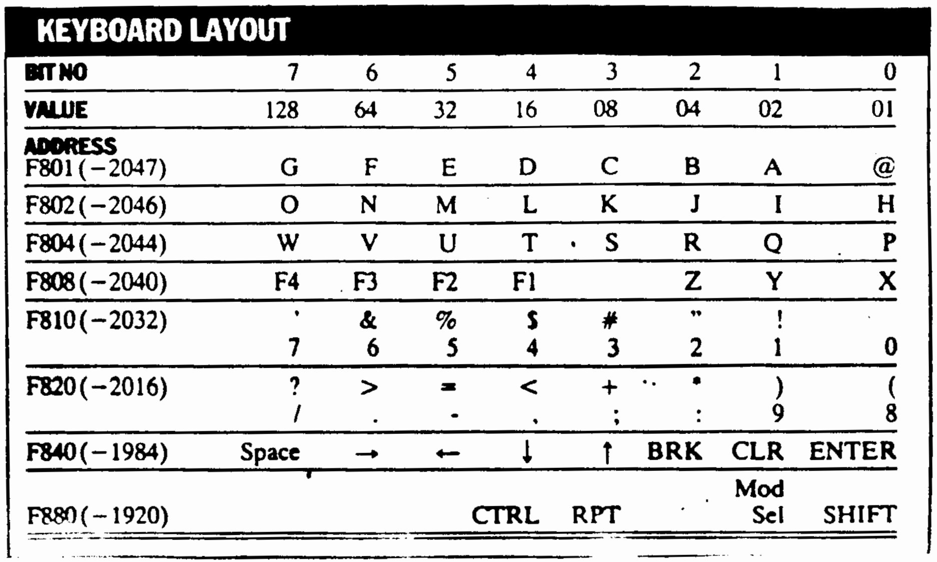
```
10  CLS:POKE 16410,103:POKE 16396,23
```

Enter the line normally, then go EDIT 10. Count the characters/spaces used in the line. In this case 32. Press X which will put you at thhe end of Line 10. then enter 32 blank characters (space bar). Leave the edit mode.

Now re-enter with EDIT 10. Press 32 then the space bar. This will put the cursor at the first blank space after '23'. Now press 32 then C then -- (back space arrow key). The line should disappear. Press Return to exit the edit mode. Now LIST 10 and you should see '10 (blank)'.

Next Month "Using Block Graphics".

Happy Computing. - Andy



KEYBOARD LAYOUT

BIT NO	7	6	5	4	3	2	1	0
VALUE	128	64	32	16	08	04	02	01
ADDRESS F801(−2047)	G	F	E	D	C	B	A	@
F802(−2046)	O	N	M	L	K	J	I	H
F804(−2044)	W	V	U	T	S	R	Q	P
F808(−2040)	F4	F3	F2	F1		Z	Y	X
F810(−2032)	' 7	& 6	% 5	$ 4	# 3	" 2	! 1	0
F820(−2016)	? /	> .	= -	< ,	+ ;	* :	) 9	(8
F840(−1984)	Space	→	←	↓	↑	BRK	CLR	ENTER
F880(−1920)			CTRL	RPT			Mod Sel	SHIFT

ONE LINERS

Here is a pair of one liners (or should it be one two-liner?) that you can use as subroutines in your programs to convert to and from hexadecimal numbers. Try them in the stand alone form given below to see how they work.

```
5   'Hex to Decimal Converter
10  INPUT H$:CO=0
20  C=C*16+ASC(H$)-48+7*(ASC(H$) >57):H$=MID$(H$.2):IF LEN(H$) THEN 20
30  PRINT C:GOTO 10
40  '
100 'Decimal to Hex Converter
110 INPUT N:L$=""
120 L=N-INT(N/16)*16:L$=CHR$(L+48-7*(L >9))+L$:N=INT(N/16):IF N THEN 120
130 PRINT L$:GOTO 110
```

The work is done in lines 20 and 120. The other lines are just there to process inputs and outputs. The program(s) will handle numbers of any length.

- Allan Clarke

COLOUR GENIE G-MON MONITOR CORRECTION

This is the correction which will get the 32K version of G-MON (Computing Today - Feb 84) working properly. The author incorporated two boo-boos and didn't explain the need to reserve memory.

The steps needed are:

1. Reserve memory of 48000 when answering MEM SIZE?

2. CLOAD G-MON (16K) and RUN it.

3. Carry out the patch procedure as in the article but do not, repeat not, alter BDD9 or BDF3 to C0H. (Delete these two lines in your copy of the article.)

4. Check that BF87 has been changed from C04 to 80H. This correction was in a later issue of Computing Today.

5. Now make a SYSTEM tape of the 32K version as in the article.

It is necessary to reserve memory when using machine code programs at the top of memory. The BASIC interpreter uses this area to handle strings and this will clobber any machine code4 if you have not protected high memory. The string working area is moved just below the memory size figure that you set at MEM SIZE?.

I would also suggest that you make a 16K SYSTEM tape of G-MON, as there will be occasions when the 32K version is in conflict with a SYSTEM program you wish to examine. You can call these two versions GMON-HI and GMON-LO.

Note: You can lower your program running time by adding the following line to the original listing:

```
5 DEFINT A,I,T.
```

BASIC processes integers faster than real numbers.

- Allan Clarke

LIST OF FINANCIAL COLOUR GENIE GROUP MEMBERS AS AT 18/6/84

ALVAREZ.Fred	49 Astley Ave. New Lynn	873 120	
AVIS.Stephen	12 Ruru Crescent.Putaruru	7750	Country
BARKER.Anne & family	1 Carbery Place.Manurewa	267 4833	
BERMAN.Robert	16 Asbury Cres.Campbells Bay	478 4022	Student
BISHOP.Chris	6 Jenanne Place. Glenfield	444 5301	
BURTON.Ron	P O Box 208. Te Aroha	625 Man.	Country
BYRNES.Ces	31 Rosehill Drive.Papakura	298 4107	Country
CARIAN.Craig	42A Comins Cres.Mission Bay Postal Box 4326.Auckland	583 061	
DONALDSON.David & family	80 Woolfield Rd.Papatoetoe	278 7598	
DUNNINGHAM.Neville	5 Kingswood Rd.Papatoetoe	278 3105	
DYER.Ray	5 Browne Street.Kawerau	6199 Kaw.	Country
EDWARDS. D.G.	10 Fairview Ave.Feilding		Country
EUSAK.Alex & Kevyn	14 Awakino Place. Manurewa	266 7423	
FISHER.Peter	64 Old Wairoa Rd.Papakura	298 4654	
GEBBIE.George	141 Flanshaw Rd.Te Atatu Sth	83 46028	
GOLDIE.Willie & Andrew	24 Douglas Ave. Mt Albert	867 533	
GORDON.Terry	99 Reeves Rd.Pakuranga	566 564	
GREEN.Gordon & family	62 Rajkot Tce.Khandallah.Wgn4.		Country
GREEN.Mike	49 Astley Ave.New Lynn	873 120	Student
GRUSNING.H.N.	48 Beatty Rd. Pukekohe	86 712	Country
GURNEY.P.A.	10 Norfolk Rise. Waiuku	59 682	Country
HAMILL.family	26 Ashlynne Ave.Papatoetoe	278 9585	
HARNESS.Justin	18 Wynyard Rd.Mt Eden	602 189	Student
HARRIS.Bert & family	15 Seakens Way. Glen Eden	818 4660	
HILL.Gerrard & family	86 Wallace Rd.Papatoetoe	278 3446	
HOLES.Freda & family	10 Rowan Terrace.Te Atatu Sth	83 45244	
* **HUGGINS**.Nola	612 Mt Albert Rd.Royal Oak	655 718	
* **HYNDS**.Ken	13 Ngahue Cres. Whenuapai	416 7404	
IRVINE.Robyn	430 Massey Rd.Mangere East	275 7007	
KAY.Ross	2/9 Longreach Drive.Glen Eden	818 4818	
LEWIS.Stuart	39 Hillcrest Ave.Rotorua		Country
LIDDEL.family	91 Taikata Rd.Te Atatu Nth	834 7129	
LOCKERBIE.Claire & Roger	63 Grampian Road.St Heliers	580 270	
MacALPINE.Mr E.	45 Mutu St.Te Awamutu		Country
McGILL.Keith	15 Manapouri Place.Pakuranga	565 643	
METCALFE.E.K.	Box 13031.University of Waikato P.O., Hamilton		Country
MILLAR.Robert	17 Onewa Rd. Northcote	486 504	Student
MILLS.Andrew	59 Blackbeech St.Akatarawa. Lower Hutt.		Country
MITCHELL.Geoff	Box 95.Mangawhai.Northland.		Country
MULLEN.Peter	54 Park Road.Glenfield	444 9155	
MUZYKA.George	33 Vermont St. Ponsonby	789 176	Student

* **Committee members**

LIST OF FINANCIAL MEMBERS continued......

	PETERS.D.R.	6 Tone Lane. Whakatane		Country
	RICHMOND.Bob	181 Browns Bay Rd.Browns Bay	478 4745	
	ROBINSON.Ian	4 Kingsley Street.Gisborne		Country
*	**ROOTS**.Ernie	512 Glenfield Rd.Glenfield	444 9669	
*	**RUSSELL**.Andy	2 Takitimu St. Whenuapai	416 6249	
	SAUNDERS.A.M.	96 Oriel Ave.Tawa.Wellington		Country
	SAVILLE.family	7A Southlynn Rd.Titirangi	817 6491	
	SCOTHERN.D.J.	64 Princess Road. Tauranga		Country
	SIMPSON.Ron	25 Renata Cres.Te Atatu Nth	834 6987	
	SMITH.Barry	23 Eliot St.New Plymouth		Country
	SPRATT.Dave	11 Ball Place. Mt Roskill	675 676	
	STARKE.Mrs L.M.	Urquhart Rd.Karaka.R.D.1. Papakura		Country
	STEPHENS.Frances & family	35 Raurenga Ave.Royal Oak	654 120	
	STRANAGHAN.family	29 Haseler Cres.Howick	535 7450	
	TODD. A.J.	46A Hutton Street. Otahuhu	276 7886	
	TRUE.Murray & Anne	4/503 Oliphant Rd.Hastings		Country
	TUAVERA.Jeffery	732 Sandringham Rd.Sand.	693 986	
	WALKER.Barry & Irene	20 Ellesmere Cres.Palm.North		Country
	WALKER.David	Kiwitea.No 7 R.D.Feilding		Country
	WAWMAN.Don	5/53 Bellevue Rd.Mt Eden	797 440 X685(work)	
	WESSELING.Ralph	112 Pt View Drive.Howick	535 6134	
*	**WILLIAMS**.Olwen	3/26A West End Rd.Herne Bay	761 954	

* -**Committee Members**

```
1 REM       THIS IS A COMPLETELY ALTERED VERSION OF WURM AS PUBLISHED IN A RECENT NEWSLETTER
2 REM       IT HAS SEEN ADDED TO AND CHANGED BY BARRY AND IRENE WALKER OF FEILDING
3 REM -------------------------------------------------------------------------------------------
10 REM WORM
20 SX=1500:SZ=0:COLOUR7
30 CLS:SS=0:PRINT@163,"DO YOU WANT INSTRUCTIONS":PRINT@257,"(Y/N)"
40 IF SC>SZ THEN SZ=SC
50 SC=0
60 PRINT@760,"HIGHEST SCORE   ";SZ
70 Q$=INKEY$:IF Q$<>"Y" AND Q$<>"N" THEN   70 ELSE IF Q$="Y" THEN 450
80 PRINT@329,"OK NO INSTRUCTIONS"
90 FOR N=1 TO 5:FOR E=1 TO 200:NEXT E,N
100 CLS:COLOUR7:SP=0
110 PRINT@1,STRING$(38,243)
120 PRINT@891,STRING$(38,243)
130 FOR DR=41 TO 841  STEP 40
140 PRINT@DR,CHR$(243):PRINT@DR+37,CHR$(243)
150 NEXTDR
160 COLOUR1
170 A=803
180 PRINT@A.CHR$(230)
190 COLOUR1
200 BB=PEEK(&HF840)
210 IF BB=32 THEN A=A-1:GOSUB  550
220  IF BB=64 THEN A=A+1:GOSUB  550
230  IF BB=16 THEN A=A+40:GOSUB 550
240  IF BB= 8 THEN A=A-40:GOSUB 550
250  IF BB=128 THEN GOSUB 520
260 IF SP>=755 THEN SS=SS-1 :SC=SC+SX :    SX=1500:GOTO 100
270 IFPEEK(17408+A)=243 THEN 340
280 IFBB>7 AND BB<65 THEN GOSUB 540
290 SX=SX-1:IF SX=0 GOTO 340
300 IF SX =0 GOTO 440
310 PRINT@921,"SCORE";SC;
320 PRINT@932,"LIVES LEFT";2-SS;" TIME ";SX;
330 GOTO 180
340 COLOUR5:PRINT@A,CHR$(254)+CHR$(249)
350 SOUND8,15
360 SOUND7,7
370 SOUND8,16
380 SOUND9,16
390 SOUND10,16
400 SOUND12,56
410 SOUND13,0
420 FOR D =0 TO 1000:NEXT
430 COLOUR 4
440 SX=1500:SS=SS+1:IF SS>=3 GOTO 30ELSE GOTO 100
450 CLS:PRINT"THIS IS A SIMPLE VERSION OF WORM."
455 PRINT"THE AIM IS TO FILL ALL THE BLANK SPACES. YOU GET POINTS FOR DOING THIS."
460 PRINT:PRINT:PRINT"USE THE ARROW KEYS TO MOVE."
470 PRINT:PRINT:PRINT"IF YOU GET LOST PRESSING THE SPACE BAR "
480 PRINT"WILL CHANGE THE COLOUR OF YOUR PLACE "
490 PRINT "BUT YOU COULD LOSE ALL YOUR POINTS !"
500 PRINT@608,"PRESS RETURN TO CONTINUE"    :INPUTX
510  CLS:GOTO100
520 SC=SC-(RND(SC+1)-1):COLOUR(RND(15))
530 RETURN
540 SOUND7,248:SOUND8,15:SOUND13,6:        SOUND7,255:RETURN
550 IF PEEK(17408+A)=32 THEN SP=SP+1:         SC=SC+1
560 RETURN
```

WORM is a modified form of the program in Issue 3 of the Newsletter. The main reason it is included is that it is an example of a program using the arrow keys, and the space bar. If you disable the break key it can be used too.

The basis of it is:

```
A=PEEK(&HF840)

if A=1    Return key was pressed
if A=2    Clear  key was pressed
if A=4    Break  key was pressed
if A=16   Down Arrow was pressed
if A=32   Left Arrow was pressed
if A=64   RightArrow was pressed
if A=128  Spacebar   was pressed
```

Demonstration program - Barry + Irene Walker

```
10 CLS
20 COLOUR8:PRINT"PRESS 1 WHEN PAGE IS FULL FOR ANOTHER  MULTICOLOURED PAGE"
30 PRINT"PRESS Z FOR A WORD
40 FRINT"PRESSING ANY OTHER KEY BRINGS YOU BACK  TO THE NORMAL SCREEN"
50 PRINT"PRESS BREAK TO END PROGRAM"
60 PRINT"PRESS ANY KEY TO CONTINUE"
70 X$=INKEY$:IFX$="" THEN  GOTO 70
80 BGRD
90 FOR I =0 TO 959
100 PRINT@I, CHR$(RND(200)+40);
110 COLOUR(RND(16))
120 NEXT I
130 A$=INKEY$:IF A$=""THEN GOTO 130
140 IF A$="1"THEN CLS:GOTO 80
150 IF A$="Z" THEN GOSUB 170:GOTO 130
160 NBGRD:GOTO 130
170 CLS
180 PRINT:PRINT
190 FRINT"            Y   Y EEE  SSSS
            Y Y  E    S
             Y   EE   SSSS
             Y   E       S
             Y   EEE  SSSS "
200 RETURN
```

DEMONSTRATION program

prints random colours or spells YES.
This mode could have possibiliies for program headings.

The page only stays on while program is running or until NBGRD.

RANDOM NUMBERS - A lot of computers use RND(1) where the Genie` uses RND(0) to do the same job. Enclosed is a photocopy of a section on using ERROR as a form of GOSUB. . The program listing in it works on a Colour Genie. It is from " BASIC PROGRAMMING PRIMER" Second Edition, by Mitchell Waite and Michael Pardee. We have found this to be a helpful book for learning basic.

To prevent my children from accumulating exorbitant amounts of chips a
Blackjack, by betting all on virtually unbeatable hands, and embarrassi
the computor, I modified the program as follows:

```
320COLOUR2:INPUT"BET,HIT,STAND,OR REVIEW CARDS <B / H / S / R
>";YN$:GOTO330
325COLOUR2:INPUT"BET OR HIT ( B / H )";YN$
326IFYN$="B"ORYN$="H"THEN330ELSEGOTO325
590COLOUR2:PRINT"YOU HAVE"SC"CHIPS":INPUT"PLAY ANOTHER HAND (Y/N)";YN$
```

This may be of interest to members.

Don Edwards, Feilding.

ERROR Used as an Extended Form of GOSUB

ERROR can also be used in another way, one the designers of BASIC probably didn't think of. In the following program we will use ERROR not to imitate an error, but almost as if it were a normal GOSUB statement. The advantage of using ERROR Instead of GOSUB is that when an ERROR statement is executed, ERR is automatically set to the error number N (specified by the number N in "ERROR N"), and ERL is set to the line number where the error occurred. It can be very useful for a subroutine to be able to figure out the line number from which it was called, and it can also be useful to be able to pass an "argument" (a number with different values) to the subroutine from the program which calls it.

Here's a program which shows how this might be done. The program asks for the user's name and age. If the name is longer than 15 characters it prints "PLEASE USE A SHORTER NAME." If the age is less than 3 years or greater than 80, it assumes the user has made a nonserious reply and prints "PLEASE BE HONEST." (If you really are more than 80 or less than 3 years old, you can change line 50 accordingly!)

```
10 ON ERROR GOTO 80
20 INPUT"PLEASE ENTER YOUR LAST NAME"; LN$
30 IF LEN(LN$) > 15 THEN ERROR 38
40 INPUT"AND YOUR AGE"; AG
50 IF AG<3.OR AG>80 THEN ERROR 38
60 PRINT"THANK YOU."
70 STOP
80 REM — ERROR-TRAPPING ROUTINE
90 IF ERR <> 38 THEN ON ERROR GOTO 0      'UNEXPECTED ERROR
100 IF ERL=30 THEN PRINT"USE A SHORTER NAME." : RESUME 20
110 IF ERL=50 THEN PRINT "PLEASE BE HONEST." : RESUME 40
120 ON ERROR GOTO 0      'UNEXPECTED LINE NUMBER
```

In this program we use the ERROR statement to branch to the "error-trapping" routine at line 80 if either the name typed in is too long, or the age typed in is less than 3 or more than 80. Neither of these conditions is really an "error," of course. We are simply using the error-handling statements to permit the routine to figure out which part of the main program called the routine.

Let's go through this step by step for the case of the user typing in an inappropriate age: 120, for example.

The first line of the program makes BASIC aware that, in the event of an error, control should go to line 80. Then the program asks for the user's last name and his age. If, when he is asked his age, the user types (for example) 120, then in line 50 the program will see that AG is greater than 80 and will go on to the next part of the statement, which is ERROR 38. (Actually, any valid error number could be used here.) Executing this statement takes us immediately to the error-trapping routine on line 90. There, on line 90, we first verify that the error is the one we expect (that is, number 38). If it isn't, we assume a "real" error (that is, one we hadn't anticipated, such as a syntax error) has taken place, and execute the ON ERROR GOTO 0 statement so that control will return to BASIC to deal with the error in the normal way. However, if the error is number 38, we then go on to check what line number it occurred on using the ERL function. In this case, it occurred on line number 50, so we will execute the second part of line 110, which is

```
PRINT"PLEASE BE HONEST." : RESUME 40
```

Control will then go back to line 40, where the user will again be asked his age. If the error routine has not been called from either of the expected line numbers, 30 or 50, then the routine goes on to execute the "ON ERROR GOTO 0" statement in line 120, which will return control to BASIC to deal with a normal error.

IRENE and BARRY WALKER of Feilding.

Chart makes conversion easy

A unique chart designed to be used by anyone in the computer industry has been produced by Control Microcomputers of Auckland.

A handy reference for programmers and microcomputer users, it shows character conversions into decimal, octal and hexadecimal codes, conversions between the bases and structure of common characters and the essential parts of the common RS-232-C and Centronic interfaces.

The chart, which took the company's staff many hours to prepare, is pictured on the facing page.

Control Microcomputers' chief executive, Basil Orr, explains how to use it:

Interfacing charts

These charts have been designed to assist in interfacing printers and other peripherals to computer systems.

Code conversion table

Printers require control sequences to enable their special features, such as changing print sizes, to be implemented. This chart shows all the standard characters together with their various numeric representations. When using BASIC, characters can be represented by a decimal number e.g. A = CHR$(65). Operating systems, monitors and word processor initialisation often require hexidecimal (Base 16) numbers. Here A = '41. The minicomputer world often uses octal numbers. In this case A = "101. This chart also converts numbers up to 255 between the three bases.

The control codes, used for controlling data transmission and peripheral features are given with explanations. They can be generated by using the control key and the appropriate upper case character e.g. <CTRL> G gives the Bell, and <CTRL> M is the same as a carriage return. The ESC character (CHR$(27), hexadecimal '1B, octal "033) is often used for enabling command sequences.

Centronics parallel interface

This interface system is often used with printers. The computer sets the eight data lines to the byte required. The data strobe line which is normally about +5 volts is momentarily pulsed to 0 volts to transfer the character to the printer.

The printer tells the computer when it can send another character by pulsing the acknowledge line. Alternatively, the computer can keep sending characters until the printer sets the busy line to + volts. Paper empty, select status and fault conditions can be sensed by the computer using the designated lines.

RS-232-C (V24) simplified

This serial communications system is used extensively by terminals and by some printers. Two-way communication requires only one line for data transfer to the computer, and one from the computer as well as the signal ground. The data bits of a character are transmitted sequentially, prefixed by a start bit, and followed sometimes by a parity bit to check for correct transmission, and always by stop bits. the characters are transmitted sequentially in this format. The Baud rate is the number of bits transmitted per second. The control lines carry status information including busy and ready between computer and terminal. Seldom are they all used together. In many applications control lines are not required. The chart correlates the various circuit names used in manuals with the pin number, function, source of the signal and description.

CODE CONVERSION TABLE

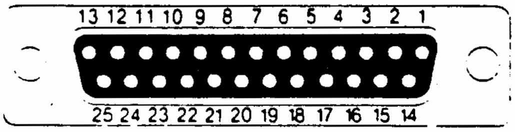
H = Hexadecimal. O = Octal. D = Decimal. CC = Control Code.
C = ASCII Character. (American Standard Code for Information Interchange)

H	O	D	CC	EXPLANATION	H	O	D	C	H	O	D	C	H	O	D	C	H	O	D	H	O	D	H	O	D	H	O	D
00	000	0	NUL	Null	20	040	32		40	100	64	@	60	140	96	`	80	200	128	A0	240	160	C0	300	192	E0	340	224
01	001	1	SOH	Start of Heading	21	041	33	!	41	101	65	A	61	141	97	a	81	201	129	A1	241	161	C1	301	193	E1	341	225
02	002	2	STX	Start of Text	22	042	34	"	42	102	66	B	62	142	98	b	82	202	130	A2	242	162	C2	302	194	E2	342	226
03	003	3	ETX	End of Text	23	043	35	#	43	103	67	C	63	143	99	c	83	203	131	A3	243	163	C3	303	195	E3	343	227
04	004	4	EOT	End Transmission	24	044	36	$	44	104	68	D	64	144	100	d	84	204	132	A4	244	164	C4	304	196	E4	344	228
05	005	5	ENQ	Enquiry	25	045	37	%	45	105	69	E	65	145	101	e	85	205	133	A5	245	165	C5	305	197	E5	345	229
06	006	6	ACK	Acknowledge	26	046	38	&	46	106	70	F	66	146	102	f	86	206	134	A6	246	166	C6	306	198	E6	346	230
07	007	7	BEL	Bell	27	047	39	'	47	107	71	G	67	147	103	g	87	207	135	A7	247	167	C7	307	199	E7	347	231
08	010	8	BS	Backspace	28	050	40	(	48	110	72	H	68	150	104	h	88	210	136	A8	250	168	C8	310	200	E8	350	232
09	011	9	HT	Horizontal Tab	29	051	41	)	49	111	73	I	69	151	105	i	89	211	137	A9	251	169	C9	311	201	E9	351	233
0A	012	10	LF	Line Feed	2A	052	42	*	4A	112	74	J	6A	152	106	j	8A	212	138	AA	252	170	CA	312	202	EA	352	234
0B	013	11	VT	Vertical Tab	2B	053	43	+	4B	113	75	K	6B	153	107	k	8B	213	139	AB	253	171	CB	313	203	EB	353	235
0C	014	12	FF	Form Feed	2C	054	44	,	4C	114	76	L	6C	154	108	l	8C	214	140	AC	254	172	CC	314	204	EC	354	236
0D	015	13	CR	Carriage Return	2D	055	45	-	4D	115	77	M	6D	155	109	m	8D	215	141	AD	255	173	CD	315	205	ED	355	237
0E	016	14	SO	Shift-Out	2E	056	46	.	4E	116	78	N	6E	156	110	n	8E	216	142	AE	256	174	CE	316	206	EE	356	238
0F	017	15	SI	Shift-In	2F	057	47	/	4F	117	79	O	6F	157	111	o	8F	217	143	AF	257	175	CF	317	207	EF	357	239
10	020	16	DLE	Data Link Escape	30	060	48	0	50	120	80	P	70	160	112	p	90	220	144	B0	260	176	D0	320	208	F0	360	240
11	021	17	DC1	X-ON	31	061	49	1	51	121	81	Q	71	161	113	q	91	221	145	B1	261	177	D1	321	209	F1	361	241
12	022	18	DC2	Device Control 2	32	062	50	2	52	122	82	R	72	162	114	r	92	222	146	B2	262	178	D2	322	210	F2	362	242
13	C23	19	DC3	X-OFF	33	063	51	3	53	123	83	S	73	163	115	s	93	223	147	B3	263	179	D3	323	211	F3	363	243
14	024	20	DC4	Device Control 4	34	064	52	4	54	124	84	T	74	164	116	t	94	224	148	B4	264	180	D4	324	212	F4	364	244
15	025	21	NAK	Not Acknowlege	35	065	53	5	55	125	85	U	75	165	117	u	95	225	149	B5	265	181	D5	325	213	F5	365	245
16	026	22	SYN	Synchronous Idle	36	066	54	6	56	126	86	V	76	166	118	v	96	226	150	B6	266	182	D6	326	214	F6	366	246
17	027	23	ETB	End Trans. Block	37	067	55	7	57	127	87	W	77	167	119	w	97	227	151	B7	267	183	D7	327	215	F7	367	247
18	030	24	CAN	Cancel	38	070	56	8	58	130	88	X	78	170	120	x	98	230	152	B8	270	184	D8	330	216	F8	370	248
19	031	25	EM	End of Medium	39	071	57	9	59	131	89	Y	79	171	121	y	99	231	153	B9	271	185	D9	331	217	F9	371	249
1A	032	26	SUB	Substitute Char	3A	072	58	:	5A	132	90	Z	7A	172	122	z	9A	232	154	BA	272	186	DA	332	218	FA	372	250
1B	033	27	ESC	Escape	3B	073	59	;	5B	133	91	[	7B	173	123	{	9B	233	155	BB	273	187	DB	333	219	FB	373	251
1C	034	28	FS	File Separator	3C	074	60	<	5C	134	92	\	7C	174	124	|	9C	234	156	BC	274	188	DC	334	220	FC	374	252
1D	035	29	GS	Group Separator	3D	075	61	=	5D	135	93	]	7D	175	125	}	9D	235	157	BD	275	189	DD	335	221	FD	375	253
1E	036	30	RS	Record Separator	3E	076	62	>	5E	136	94	^	7E	176	126	~	9E	236	158	BE	276	190	DE	336	222	FE	376	254
1F	037	31	US	Unit Separator	3F	077	63	?	5F	137	95	_	7F	177	127		9F	237	159	BF	277	191	DF	337	223	FF	377	255

To generate use Control Key <CTRL> with

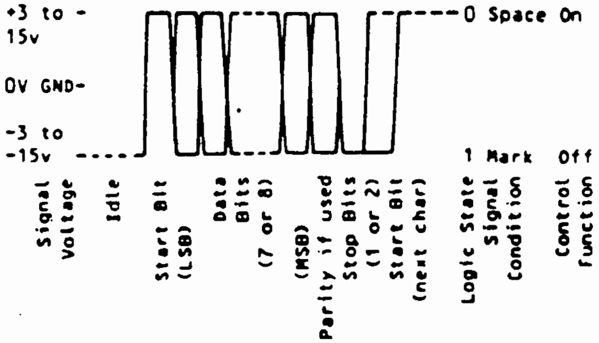
RS 232-C (V24) SIMPLIFIED

25 pin connector using round terminals
View of front of Socket (female connector)



DTE = Data Terminal Eqpt (Printer,Modem port)
DCE = Data Communications Eqpt (Printer port,Modem)

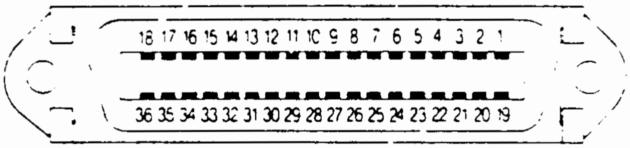
PIN	CIRCUIT		FUNC	SRC	CODE	DESCRIPTION
1	AA	101	Gnd	-	FG	Frame Ground
2	BA	103	Data	DTE	TD	Transmit Data
3	BB	104	Data	DCE	RD	Receive Data
4	CA	105	Cont	DTE	RTS	Ready to Send
5	CB	106	Cont	DCE	CTS	Clear to Send
6	CC	107	Cont	DCE	DSR	Data Set Ready
7	AB	102	Gnd	-	SG	Signal Ground
8	CF	109	Cont	DCE	CD	Carrier Detect
20	CD	108	Cont	DTE	DTR	Data Terminal Ready
21	CG	110	Cont	DCE	SQD	Signal Quality Detect
22	CE	125	Cont	DCE	RI	Ring Indicator
23	CH	111	Cont	DTE	DRS	Data Rate Select



Control signals must be on to allow data transfer

CENTRONICS PARALLEL INTERFACE

36 pin connector using flat contacts.
View of front of socket (female - groove down middle)



PIN	SIGNAL	SRC	PIN	SIGNAL	SRC
1	* Data strobe pulse	Comp	19	Signal ground	-
2	Data bit 1 (LSB)	Comp	20	Signal ground	-
3	Data bit 2	Comp	21	Signal ground	-
4	Data bit 3	Comp	22	Signal ground	-
5	Data bit 4	Comp	23	Signal ground	-
6	Data bit 5	Comp	24	Signal ground	-
7	Data bit 6	Comp	25	Signal ground	-
8	Data bit 7	Comp	26	Signal ground	-
9	Data bit 8 (MSB)	Comp	27	Signal ground	-
10	* Acknowledge pulse	Prt	28	Signal ground	-
11	Busy	Prt	29	Signal ground	-
12	Paper Empty	Prt	30	Signal ground	-
13	Select	Prt	31	* Input prime	Comp
14	Signal ground	-	32	* Fault	Prt
15	No connection	-	33	Signal ground	-
16	Signal ground	-	34	* Ext prime	Comp
17	Chassis ground	-	35	No connection	-
18	+ 5 volts DC	Prt	36	Input busy	Prt

Signals TTL levels: Active +2 to +5v, False +0 to +0.4V
* Inverted signals: Active +0 to +0.4V, False +2 to +5v

Compiled by
CONTROL
MICROCOMPUTERS
P O Box 68-474 Auckland. Pr (09) 600-687

THANKS TO BARRY & IRENE WALKER

LUNAR LANDING -- *converted for the Colour Genie by George Muzyka*

```
1 CLS
2 PRINT@17,"LEM":PRINT@55,"*******"
4 PRINT:PRINT:PRINT "CREATIVE COMPUTING  MORRISTOWN,        NEW JERSEY"
7 REM ROCKT2 IS AN INTERACTIVE GAME THAT SIMULATES A LUNAR
8 REM LANDING IS SIMILAR TO THAT OF THE APOLLO PROGRAM.
9 REM THERE IS ABSOLUTELY NO CHANCE INVOLVED
10 Z$="GO"
15 B1=1
20 M=17.95
25 F1=5.25
30 N=7.5
35 R0=926
40 V0=1.29
45 T=0
50 H0=60
55 R=R0+H0
60 A=-3.425
65 R1=0
70 A1=8.84361E-04
75 R3=0
80 A3=0
85 M1=7.45
90 M0=M1
95 B=750
100 T1=0
105 F=0
110 P=0
115 N=1
120 M2=0
125 S=0
130 C=0
135 IF Z$="YES" THEN 1150
140 PRINT
145 PRINT"LUNAR LANDING SIMULATION":PRINT"===== ======= ==========":PRINT
150 PRINT
155 PRINT"HAVE YOU FLOWN AN APOLLO/LEM MISSION    BEFORE";
160 PRINT"         (YES OR NO)";
165 INPUT Q$
170 IF Q$="YES" THEN 190
175 IF Q$="NO" THEN 205
180 PRINT"JUST ANSWER THE GUESTION, PLEASE, ";
185 GOTO 160
190 PRINT
191 PRINT" 1=METRIC    0=ENGLISH"
195 PRINT"INPUT MEASUREMENT OPTION NUMBER";
200 GOTO 225
205 PRINT
210 PRINT"WHICH SYSTEM OF MEASUREMENT DO YOU      PREFER?"
215 PRINT" 1=METRIC    0=ENGLISH"
220 PRINT"ENTER THE APPROPRIATE NUMBER";
225 INPUT K
230 PRINT
235 IF K=0 THEN 280
240 IF K=1 THEN 250
245 GOTO 220
250 Z=1852.8
255 M$="METRES"
260 G3=3.6
265 N$=" KILOMETRES"
270 G5=1000
275 GOTO 305
280 Z=6080
```

```
285 M$="FEET"
290 G3=.592
295 N$="N.MILES"
300 G5=2
305 IF B1=3 THEN 670
310 IF Q$="YES" THEN 485
312 CLS:PRINT:PRINT:PRINT
315 PRINT
320 PRINT">YOU ARE ON A LUNAR LANDING MISSION<":PRINT:PRINT
321 PRINT"  AS THE PILOT OF THE LUNAR EXCURSION"
325 PRINT"MODULE. YOU WILL BE EXPECTED TO"
330 PRINT"GIVE CERTAIN COMMANDS TO THE MODULE"
331 PRINT"NAVIGATION SYSTEM."
335 PRINT"  THE ON-BOARD COMPUTER WILL GIVE A"
336 PRINT"RUNNING ACCOUNT OF INFORMATION"
340 PRINT"NEEDED TO NAVIGATE THE SHIP."
342 PRINT:PRINT
345 PRINT:PRINT
355 INPUT" PRESS <RETURN> TO CONTINUE...":ZZ$
360 CLS
365 PRINT:PRINT"  THE ATTITUDE ANGLE CALLED FOR IS":PRINT"DESCRIBED AS FOLLOWS:"
370 PRINT"                    +++++++++++++++++":PRINT"  + OR -180 DEG. IS   +    -180,180   +"
375 PRINT"DIRECTLY AWAY FROM   +       ▲       +"
376 PRINT"THE MOON.            + -90 < -+- > 90+"
378 PRINT"                     +       ■T      +"
380 PRINT"  -90 DEG. IS ON A   +       0       +"
382 PRINT"TANGENT IN THE       +               +"
385 PRINT"DIRECTION OF ORBIT.  +<< DIRECTION <<+"
387 PRINT"                     +    OF ORBIT   +"
390 PRINT"                     +               +"
400 PRINT"  +90 DEG. IS ON A   +SURFACE OF MOON+""
405 PRINT"TANGENT FROM THE     +///////////////+"
410 PRINT"DIRECTION OF ORBIT.  +++++++++++++++++"
415 PRINT
420 PRINT"  0 (ZERO) DEG. IS"
425 PRINT"DIRECTLY TOWARD THE MOON."
427 PRINT:PRINT:INPUT"   PRESS <RETURN> TO CONTINUE...":ZX$
428 CLS
435 PRINT"  ALL ANGLES BETWEEN -180 AND 180       DEGREES ARE ACCEPTED."
440 PRINT
445 PRINT"::1 FUEL UNIT = 1 SEC. AT MAX. THRUST::":PRINT:PRINT
450 PRINT"  ANY DISCREPANCIES ARE ACCOUNTED FOR INTHE USE OF FUEL FOR AN ATTITUDE CHANGE."
460 PRINT"  AVAILABLE ENGINE POWER: 0 (ZERO) AND  ANY VALUE BETWEEN 10 AND 100 PERCENT."
470 PRINT:PRINT
475 PRINT">NEGATIVE THRUST OR TIME IS PROHIBITED<"
480 PRINT
482 PRINT:PRINT:PRINT:PRINT:INPUT"   PRESS <RETURN> TO CONTINUE...":ZY$
485 CLS
490 PRINT"INPUT:":PRINT"  TIME INTERVAL IN SECONDS ------ (T)"
495 PRINT"  PERCENTAGE OF THRUST ---------- (P)"
500 PRINT"  ATTITUDE ANGLE IN DEGREES ----- (A)"
505 PRINT
510 IF Q$="YES" THEN 535
515 PRINT"FOR EXAMPLE:"
520 PRINT"  T,P,A? 10,65,-60"
525 PRINT"TO ABORT THE MISSION AT ANY TIME,      ENTER 0,0,0"
530 PRINT
535 PRINT"OUTPUT:*TOTAL TIME IN ELAPSED SECONDS"
540 PRINT"       *HEIGHT IN ":M$
545 PRINT"       *DISTANCE FROM LANDING SITE IN          ":M$
550 PRINT"       *VERTICAL VELOCITY IN ":M$:"/           SECOND"
555 PRINT"       *HORIZONTAL VELOCITY IN ":M$:"/         SECOND"
560 PRINT"       *FUEL UNITS REMAINING"
```

```
565 PRINT
567 PRINT:PRINT:INPUT" PRESS <RETURN> TO START MISSION....";Z$
568 CLS
570 GOTO 670
575 PRINT
580 PRINT"  T,P,A";
585 INPUT T1,F,P
590 F=F/100
595 IF T1<0 THEN 905
600 IF T1=0 THEN 1090
605 IF ABS(F-.05)>1 THEN 945
610 IF ABS(F-.05)<.05 THEN 945
615 IF ABS(P)>180 THEN 925
620 N=20
625 IF T1<400 THEN 635
630 N=T1/20
635 T1=T1/N
640 P=P*3.14159/180
645 S=SIN(P)
650 C=COS(P)
655 M2=M0*T1*F/B
660 R3=-.5*R0*((V0/R)[2)+R*A1*A1
665 A3=-2*R1*A1/R
670 FOR I=1 TO N
675 IF M1=0 THEN 715
680 M1=M1-M2
685 IF M1>0 THEN 725
690 F=F*(1*M1/M2)
695 M2=M1+M2
700 PRINT"YOU ARE OUT OF FUEL...."
705 M1=0
710 GOTO 725
715 F=0
720 M2=0
725 M=M-.5*M2
730 R4=R3
735 R3=-.5*R0*((V0/R)[2)+R*A1*A1
740 R2=(3*R3-R4)/2+.00526*F1*F*C/M
745 A4=A3
750 A3=-2*R1*A1/R
755 A2=(3*A3-A4)/2+.0056*F1*F*S/(M*R)
760 X=R1*T1+.5*R2*T1*T1
765 R=R+X
770 H0=H0+X
775 R1=R1+R2*T1
780 A=A+A1*T1+.5*A2*T1*T1
785 A1=A1+A2*T1
790 M=M-.5*M2
795 T=T+T1
800 IF H0<3.287828E-04 THEN 810
805 NEXT I
810 H=H0*Z
815 H1=R1*Z
820 D=R0*A*Z
825 D1=R*A1*Z
830 T2=M1*B/M0
832 PRINT
833 IF M$="METRES" THEN LET X$="M"ELSE LET X$="F"
835 PRINT"TOTAL TIME (S) -------- ";T:PRINT"HEIGHT (";X$;") ------------ ";H:PRINT"MOON RANGE (";X$;") -------- ";D:PRINT"VERT. VELO
CITY (";X$;"/S) -- ";H1:PRINT"HOR. VELOCITY (";X$;"/S) --- ";D1:PRINT"FUEL UNITS REMAINING -- ";T2
845 IF H0<3.287828E-04 THEN 880
850 IF R0*A>164.4736 THEN 1050
855 IF M1>0 THEN 580
860 T1=20
```

```
865 F=0
870 P=0
875 GOTO 620
880 IF R1<-8.21957E-04 THEN 1020
885 IF ABS(R*A1)>4.931742E-04 THEN 1020
890 IF H0<-3.287828E-04 THEN 1020
895 IF ABS(D)>10*Z THEN 1065
900 GOTO 995
905 PRINT
910 PRINT"THIS SPACECRAFT IS NOT ABLE TO VIOLATE THE SPACE-";
915 PRINT"TIME CONTINUUM."
920 GOTO 575
925 PRINT
930 PRINT"IF YOU WANT TO SPIN AROUND, GO OUTSIDE THE MODULE"
935 PRINT"FOR AN E.V.A."
940 GOTO 575
945 PRINT
950 PRINT"IMPOSSIBLE THRUST VALUE ";
955 IF F<0 THEN 985
960 IF F-.5<.05 THEN 975
965 PRINT"TOO LARGE"
970 GOTO 575
975 PRINT"TOO SMALL"
980 GOTO 575
985 PRINT"NEGATIVE"
990 GOTO 575
995 PRINT
1000 PRINT"TRANQUILITY BASE HERE -- THE EAGLE HAS LANDED"
1005 PRINT"CONGRADULATIONS -- THERE WAS NO SPACECRAFT DAMAGE"
1010 PRINT"YOU MAY NOW PROCEED WITH SURFACE EXPLORATION."
1015 GOTO 1100
1020 PRINT
1025 PRINT"CRASH !!!!!!!!!!"
1030 PRINT"YOUR IMPACT CREATED A CRATER ";ABS(H);M$;" DEEP."
1035 X1=SQR(D1*D1+H1*H1)*G3
1040 PRINT"AT CONTACT YOU WERE TRAVELLING ";X1;N$;"/HR"
1045 GOTO 1100
1050 PRINT
1055 PRINT"YOU HAVE BEEN LOST IN SPACE WITH NO HOPE OF RECOVERY."
1060 GOTO 1100
1065 PRINT"YOU ARE DOWN SAFELY -"
1075 PRINT
1080 PRINT"BUT MISSED THE LANDING SITE BY ";ABS(D/G5);N$
1085 GOTO 1100
1090 PRINT
1095 PRINT"MISSION ABORTED"
1100 PRINT
1105 PRINT"DO YOU WANT TO TRY IT AGAIN (YES/NO)?"
1110 INPUT Z$
1115 IF Z$="YES" THEN 20
1120 IF Z$="NO" THEN 1130
1125 GOTO 1105
1130 PRINT
1135 PRINT"TOO BAD, THE SPACE PROGRAM HATES TO LOSE EXPERIENCED"
1140 PRINT"ASTRONAUTS."
1145 STOP
1150 PRINT
1155 PRINT"OK, DO YOU WANT THE COMPLETE";PRINT"INSTRUCTIONS OR THE INPUT-OUTPUT STATEMENTS?"
1160 PRINT
1165 PRINT"1=COMPLETE INSTRUCTIONS"
1170 PRINT"2=INPUT-OUTPUT STATEMENTS"
1175 PRINT"3=NEITHER"
1180 INPUT B1
1185 Q$="NO"
```

```
1190 IF B1=1 THEN 205
1195 Q$="YES"
1200 IF B1=2 THEN 190
1205 IF B1=3 THEN 190
1210 GOTO 1165
1215 END
```

HISTOGRAM PROGRAM

I based this program on the histogram from the demonstration program which I received with the machine. Some friends of mine helped in redesigning the program so that the information can be read from data statements.

I can fit up to 18 individual bars across the X-axis and up to a height of 20 units max. on the Y-axis. The units on the Y-axis can be of any value, but inaccuracie occur in very large numbers when the actual bars are produced for very small X values (i.e. Y value of 6 and Y value 2 will be same height if max Y value = 100 units.

Terry Gordon, Pakuranga.

HISTOGRAM PROGRAM.

by. J. Dwen c/f JAMES 'COOK HIGH SCHOOL.
& T. GORDON.

VA = The values of the heights of the histogram bars, the number must not exceed 20, otherwise the histogram bar will not fit the screen.

XA$ = The units for the x-axis, these must not exceed a factor of 20 units, otherwise the screen becomes overloaded on the x-axis.

MA = TOTAL POSSIBLE MARK OR SCORE VALUE FOR Y-AXIS.

NU = The number of histogram bars, up to 18 seems to fit the screen.

```
100 CLS
200 DIM VA(20)
300 DIM XA$(20)
400 REM DX=STARTING POINT
500 DX=807
600 READ MA
700 READ NU
800 FOR T=1TO NU
900 READ VA(T)
1000 NEXT
1100 FORT=1  TO NU
1200 READ XA$(T)
1300 NEXT T
1400 INPUT"UNITS Y";Y$ - Yaxis units - Asks for the name of units for Y-axis
1500 INPUT"UNITS X";X$ - Xaxis units - Asks for the name of units for X-axis.
1550 CLS
1600 F=LEN(Y$)
1610 FOR D=1  TO 34                              1610 - 1630 } Prints the — of the x-axis.
1620 COLOUR7:PRINT@845+D,"S"; — ASCII code 211 or SHIFT/MOD SEL/I
1630 NEXTD
1640 FOR U=1  TO20                               1640 - 1660 } Prints the | of the y-axis.
1650 COLOUR7:PRINT@845-(U*40),"L"; - ASCII code 219 or SHIFT/MOD SEL/Q.
1660 NEXTU
1700 Q=(INT((20-F)/2))*40
1800 FORT=1  TOF
1900 COLOUR8:PRINT@Q,MID$(Y$,T,1)   Prints y-axis units "name"
2000 Q=Q+40
2100 NEXTT
2200 SP=INT(MA/20):IF  MA<20  THEN  SP=(MA/20)
2300 FOR  T=0  TO  19
2400 IF  T=0  THEN  PRINT@41,MA
2500 COLOUR4:IF  T<>0  THEN  PRINT@T*40+41,MA-T*SP  • prints y-axis units
2600 NEXTT
2700 QX=LEN(X$)                        2700 - 2800 } Prints x-axis 'units name"
2800 COLOUR6:PRINT@926+INT((33-QX)/2),X$
2900 FOR  T=1  TO NU                   2900 - 3100 } Prints x-axis units from data 10700
3000 COLOUR2:PRINT@846+(T-1)*2,XA$(T)
3100 NEXTT
                                       3200 - 3600 Prints bars of the histogram.
3200 FOR  T=1  TO NU                                                       ■
3300 FOR  AA=1TO  VA(T)/SP
3400 COLOUR1:PRINT@(DX-40*AA)+2*(T-1),"J" ASCII code 202 or SHIF/MOD SEL/©
3500 NEXTAA
3600 NEXTT
9000 K$=INKEY$:IFK$=""THEN  9000
10000 REM OUT OF SCORE
10100 DATA 100
10200 REM NUMBER OF BARS
10300 DATA 10
10400 REM VALUES OF HEIGHTS OF BARS,(VA)
10500 DATA 5,10,16,19,5,50,11,16,6,2
10600 REM X AXIS NAMES XA$(T)
10700 DATA 01,.2,.3,.4,.5,.6,.7,.8,.9,10
```

```
10 REM THE ERROR HANDLING ROUTINE IS          THERE TO TRAP THE CONDITION            CAUSED BY INPUTING A BLANK              STATE
MENT.IE JUST PUSHING RETURN        KEY.
20 ON ERROR GOTO 640
30 CLS
40 COLOUR7:PRINTTAB(22);"UNDER OR OVER"
50 REM THIS PROGAM HAS BEEN MODIFIED          FROM THE PROGRAM IN 'MORE BASIC         COMPUTER GAMES',PAGE 170.
60 REM BY                                     BARRY & IRENE WALKER
70 PRINT:PRINT:PRINT
80 PRINT"THIS IS A GAME OF UNDER OR OVER."
90 PRINT"THE OBJECT IS TO PICK A NUMBER BETWEEN  2 AND 12."
100 PRINT" IF YOUR NUMBER IS BETWEEN 2 AND 6 YOUR BET IS UNDER"
110 PRINT" IF YOUR NUMBER IS 7 YOUR BET IS EVEN."
120 PRINT" IF YOUR BET IS BETWEEN 8 AND 12 YOUR  BET IS OVER."
130 PRINT"IF THE NUMBER TURNED UP IS YOUR NUMBER  YOU WIN FOUR TIMES YOUR BET."
140 PRINT"IF YOUR NUMBER IS THE SAME TYPE AS THE  NUMBER TURNED UP YOU WIN EVEN MONEY."
150 PRINT"YOU HAVE $100 TO START WITH.GOOD LUCK!":PRINT:PRINT
160 A=100:COLOUR8
170 RESTORE
180 REM SET UP TYPE NAMES
190 READ A$,B$,C$
200 DATA"UNDER","OVER","EVEN"
210 PRINT"INPUT THE NUMBER (2-12)AND AMOUNT OF    YOUR BET."
220 B1$="A":C1$="1"
230 INPUT B1$,C1$
240 REM CHECK THAT THE INPUTED VALUES          ARE NUMBERS AND IN RANGE
250 IF ASC(B1$) <48 OR ASC(B1$) > 57 OR         ASC(C1$) <48 OR ASC(C1$) > 57  THEN CLS: PRINT "PLEASE USE NUMBERS ONLY":ELSE GOTO
270
260 PRINT:PRINT"YOU HAVE GOT $";A :         GOTO 210
270 B=VAL(B1$):C=VAL(C1$)
280 IF B<2 OR B>12 THEN CLS:PRINT"YOU HAVE GOT $";A:PRINT "NUMBERS BETWEEN 2 AND 12 PLEASE":GOTO 210
290 IF C>A THEN COLOUR1:PRINT"CHEAT!YOU DON'T HAVE THAT MUCH MONEY":COLOUR8:GOTO 210
300 REM PRINT OUTPUT
310 CLS:PRINT:PRINT"THE DICE HAVE NOW BEEN THROWN."
320 PRINT"THE RESULTS ARE AS FOLLOWS :"
330 REM  THE DICE IS THROWN
340 Q=INT(RND(6)):RANDOM:R=INT(RND(6)):     RANDOM
350 PRINT " DIE #1","DIE #2","SUM","YOUR#"
360 IF Q+R>7 THEN420 ELSE IF Q+R=7 THEN 460
370 PRINTQ,R,Q+R,B,"TOSS - ";A$:PRINT
380 REM FIND OUT IF YOU WIN OR LOSE
390 IF B=Q+R THEN 490
400 IF B<7 THEN   520
410 GOTO550
420 PRINTQ,R,Q+R,B,"TOSS - ";B$:PRINT
430 IF B=Q+R THEN 490
440 IF B>7   THEN 520
450 GOTO 550
460 PRINTQ,R,Q+R,B,"TOSS - "C$:PRINT
470 IF B=Q+R THEN 490
480 GOTO 550
490 A=A+(4*C)
500 PRINTTAB(6);"********YOU WIN 4 TO 1*******"
510 PRINT"THE AMOUNT YOU NOW HAVE IS $";A:PRINT:GOTO 580
520 A=A+C
530 PRINTTAB(5);"*****YOU WIN EVEN MONEY*****"
540 PRINT"THE AMOUNT YOU NOW HAVE IS $";A:PRINT:GOTO 580
550 A=A-C
560 PRINTTAB(10)"!!!!!YOU LOSE!!!!!"
570 PRINT"THE AMOUNT YOU NOW HAVE IS $";A:PRINT
580 IF A>0 THEN 170
590 PRINT:PRINT:PRINT:COLOUR4:PRINT"THE GAME IS OVER.SORRY MATE YOU ARE     FLAT BROKE."
600 PRINT"THIS PROVES IT IS NOT GOOD TO GAMBLE."
610 PRINT"DO YOU WANT TO PLAY AGAIN?";     :INPUTW$
620 IF LEFT$(W$,1)="Y" THEN CLS:PRINT"YOU HAVE $100.GOOD LUCK THIS TIME!":GOTO    160
630 END
640 IF ERR= 5 OR ERL=250 THEN B1$="ERROR": C1$ ="ERROR":RESUME
650 ON ERROR GOTO 0
660 REM HAND CONTROL BACK TO BASIC IF          AN UNEXPECTED ERROR OCCURS
```

```
10 '**********  ANAGRAMS  **************
20 '
30 '
40 '
50 '  BY MICHAEL MERRYLEES  AGED 12-13
60 '
65 '  ADAPTED  TO COLOUR  GENIE
70 '
80 '  BY DON EDWARDS AGED 120-130
90 '
100 '
110 '
120 '
130 '***********************************
180 '********* INSTRUCTIONS ************
190 CLS:CLEAR2100:N9=1
195 COLOUR1
200 PRINTTAB(12)"A N A G R A M S"
210 PRINT:PRINT
230 PRINT:PRINT:PRINT"IF INSTRUCTIONS ARE NEEDED TYPE";CHR$(34);"I";CHR$(34);"EL
SE  TYPE";CHR$(34);"N";CHR$(34)
240 A$=INKEY$:IFA$=""THEN240
250 IFLEFT$(A$,1)="I"THEN270
260 IFLEFT$(A$,1)="N"THEN365ELSEGOTO240
270 PRINT$0,CHR$(30):PRINT$12,"A N A G R A M S":PRINT
280 PRINT"  ANAGRAMS ARE WORDS  IN  WHICH  ALL   THE LETTERS ARE JUMBLED UP  AN
D YOU  ARE GIVEN  THE TASK OF  UNJUMBLING THE     LETTERS IN  A  LIMITED  TIME.
I HAVE    TAKEN THIS  OLD GAME  AND WITH THE HELP OF DATA AND RANDOM NUMBER STAT
EMENTS  I"
290 PRINT"HAVE  CONVERTED  IT INTO  A COMPUTOR    PROGRAM.";CHR$(10);"    THERE A
RE FOUR  LEVELS IN  THE       PROGRAM:";CHR$(10);"        1    = 4-6 LETTER WORDS
";CHR$(10);"         2    = 7-8 LETTER WORDS"
300 PRINT"         3    = 9-13 LETTER  WORDS";CHR$(10);"   AND   4    = WHICH HAS  WOR
DS RANGING"      ;CHR$(10);"              FROM 4-13 LETTERS IN THEM";
310 PRINT:PRINT:PRINT:PRINT"PRESS ANY  KEY TO  CONTINUE";
320 A$=INKEY$:IFA$=""THEN320
330 CLS:PRINT$12,"A N A G R A M S":PRINT:PRINT:PRINT"     IN EACH ONE OF THESE LE
VELS THE     COMPUTOR  WILL SHOW YOU  ONE  OF   THE    MANY";
340 PRINT" WORDS POSSIBLE AND YOU WILL HAVE   TO WORK IT OUT AND TYPE  IT IN  DU
RING A  LIMITED TIME.  IF YOU CANNOT WORK THE ANAGRAM OUT AND  YOU WOULD LIKE
EITHER TO  STOP OR TRY ANOTHER ANAGRAM THEN   TYPE IN '/'."CHR$(30)
350 PRINT"    THERE ARE ALSO TWO LEVELS OF        DIFFICULTY IN  THE PROGRAM";CH
R$(10);"        (P)   --  PROFESSIONAL";CHR$(10);"        (A)   -- AMATEUR"
360 PRINT" N.B. PLEASE DO  NOT PRESS RETURN  AFTER";CHR$(10);"      YOU ENTER YOU
R ANSWER  !!"
365 COLOUR5
370 PRINT:INPUT"ENTER THE LEVEL YOU WANT(1-4)";U:IFU>4ORU<1THEN370
380 PRINT$880,CHR$(30);:PRINT$880,;:INPUT"(P)ROFESSIONAL    OR    (A)MATEUR";TT$:
IFTT$="P"THENTT=50ELSETT=150
385 COLOUR1
390 '******** DATA INPUT ***************
400 CLS:PRINT:PRINT:PRINT:PRINT"PRESS RETURN FOR COMPUTOR TO START DATA INPUT";
410 A$=INKEY$:IFA$=""THEN410
420 PRINT:PRINT"   ENTERING  COUNTERS....."
430 READV,W,X
440 Y=V+W+X
450 IFU=1THENZ=V:M=6:DIMD$(V)
460 IFU=2THENZ=W+V:M=8:DIMD$(W)
470 IFU=3THENZ=Y:M=13:DIMD$(Y)
480 IFU=4THENZ=Y:M=13:DIMD$(Y)
490 T=1
500 PRINT:PRINT"   ENTERING  WORDS  :-"CHR$(10);:PRINT:PRINT"    THIS MAY  TAKE
 A WHILE ACCORDING TO";CHR$(10);"    WHICH LEVEL YOU PICKED  SO WATCH THE";CHR$(
10);"   SHOW WHILE YOUR COMPUTOR  LISTENS TO"CHR$(10);"    SOME MUSIC  ..."
510 FORN2=1TO2000:NEXT
520 FORB=1TOZSTEP5:READD$(T),D$(T+1),D$(T+2),D$(T+3),D$(T+4)
530 IFU=1THENT=T+5
540 IFU=2ANDB>VTHENT=T+5
550 IFU=3ANDB>WTHENT=T+5
```

```
570 COLOURRND(8):FORWW=1TO5:PRINT$RND(950),CHR$(42):NEXT
580 NEXTB:FORB=1TOT-1
590 IFLEN(D$(B))<8THENN3=INT(LEN(D$(B))/2):D$(B)=RIGHT$(D$(B),N3)+LEFT$(D$(B),LE
N(D$(B))-N3)ELSEN3=INT(LEN(D$(B))/3):D$(B)=RIGHT$(D$(B),N3)+MID$(D$(B),LEN(D$ (B
))-2*N3+1,N3)+LEFT$(D$(B),LEN(D$(B))-2*N3)
600 COLOURRND(8):PRINT$RND(959),CHR$(42):NEXTB
610 RANDOM
620 PRINT$776," FINISHED !!!"
630 PRINT"I AM NOW  WORKING OUT A FEW                    A N A G R A M S  FOR Y
OU";:FORA=0TO20:NEXT
640 '** THE MAKINGS OF AN ANAGRAM ******
650 S=RND(T-1):IFU=2ANDS<VTHEN650
660 IFU=3ANDS<WTHEN650
670 A$=D$(S)
680 A=LEN(A$)
690 IFL=1THEN720
700 IFN5=1THEN720
710 DIMA$(M),A(M),B(M)
720 FORB=1TOA:A$(B)=MID$(A$,B,1):NEXT
730 N=1:NN=1
740 FORB=1TOA
750 A(B)=RND(A)
760 FORC=1TOB
770 IFA(B)=B(C) THENNN=NN+1:IFNN>20THENN5=1:GOTO650ELSEGOTO750
780 NEXTC
790 B(B)=A(B)
800 NEXTB
810 FORB=1TOA:B$=B$+A$(A(B)):NEXT
820 N=N+1:IFN>5THENRANDOM:GOTO650
830 IFB$=A$THEN720
840 '******* MAIN SCREEN LAYOUT ********
845 COLOUR2
850 IFN9<>1THENPRINT$333,"----------------";:GOTO900
860 CLS:PRINT$57,"ANAGRAMS"
870 PRINT$129,".........................";:PRINT$215,"YOUR ANAGRAM";:PRINT$259, S
TRING$(4,92);:PRINT$329,"XXXX----------------";STRING$ (4,93);:PRINT$450,"SCORE:
";:PRINT$532,"***";:PRINT$468,"TIME:";:PRINT$548,"**** MINS";
880 PRINT$771,"********************";:PRINT$532,"000";
890 FORN8=1TO1000:NEXT
900 N7=INT(28-A+512):IFN7/2<>INT(N7/2)THENN7=N7-1
905 COLOUR6
910 PRINT$337,B$;:COLOUR3:PRINT$547,USING"###.#";A*TT/500;:COLOUR6:PRINT$133,"ST
ART";
920 '**** LISTENING TO YOUR ANSWERS ****
930 C=1:N7=N7+384
940 FORB=1TOTT*A:G=TT*A-B:GOSUB1110:COLOUR6:C$=C$+INKEY$
950 IFRIGHT$(C$,1)=CHR$(8)ANDC$<>CHR$(8)THENC$=LEFT$(C$,LEN(C$)-2):PRINT$697,C$;
"**";
960 IFC$=CHR$(8)THENC$=""
970 IFRIGHT$(C$,1)=CHR$(10)ORRIGHT$(C$,1)=CHR$(13)THENC$=LEFT$(C$,LEN(C$)-1)
980 PRINT$697,C$;
990 IFRIGHT$(C$,1)="/" THENPRINT$132,"THE SOLVED ANAGRAM:";:PRINT$337   ,A$;:FOR
N8=1TO200:NEXT:GOTO1050
1000 IFLEN(C$)+1>ATHEN1030
1010 NEXTB
1020 PRINT$132,"OUT OF  TIME - IT  WAS..":PRINT$337,A$;:GOTO1050

1030 IFC$=A$THENPRINT$132,"CORRECT ! MY ANAGRAM -";:PRINT$337   ,A$;:N6=N6+1:N6$
=STR$(N6):COLOUR3:PRINT$535-LEN(N6$),N6$;:COLOUR6:GOTO1050
    0
1040 PRINT$132,"WRONG,BAD LUCK IT WAS";:PRINT$337   ,A$;:GOTO1050
1050 FORN8=1TO2000:NEXT:PRINT$132,   " 1 FOR MORE, 2 TO END";
1060 Z$=INKEY$:IFZ$=""THEN1060
1064 COLOUR3
1065 Z=VAL(Z$):IFZ<1ORZ>2THENPRINT$890,"WRONG CHOICE,TRY AGAIN":FORN=1TO500:NEXT
:PRINT$880,CHR$(30):GOTO1060"
1066 COLOUR6
```

```
1070 IFZ$="1" THENPRINT$337,STRING$(LEN(B$),"*");:A$="":B$="":C$="":L=1:N9=2:PRI
NT$335,"I'M  THINKING";:COLOUR2:PRINT$132,".......................";:PRINT$690,"
          ":COLOUR2:PRINT$549,"****";:RANDOM:GOTO650
1080 CLEAR50
1085 COLOUR6
1090 PRINT$132,"THANKS FOR PLAYING,SO LONG";:FORA=1TO1000:NEXT:FORA=0TO43:PRINT:
NEXT:END
1100 '****  TIME SUB-ROUTNE ***********
1110 COLOUR3:PRINT$548,USING"###.#";G/500;:RETURN



1120 '*** DATA STATMENTS WITH ENCODED          ANAGRAMS **********************
1130 DATA40,60,70
1140 DATAHEAC,REHI,GEUR,TESI,BTDE,ALOV,INVA,WNLA,OMWH,ITKN,IPTR
1150 DATACLECY,ULTFA,ROLEN,ORDSW,UGEGA,IZESE,OKYSM,DTHWI,CHTYA,EUEQU,RIPST
1160 DATAIONMOT,EUMMUS,RCEFIE,MITCOM,IREDES,RORMIR,CLEMUS,OURLAB,AGEGAR,IALSER,E
ALORD,NEDBUR,PITPUL,ISHPUN,PETPUP,IFYPUR,ESSSTR,IPESTR
1170 DATAYCLEBIC,MNEYCHI,DEMNCOM,CENDDES,MACHSTO,SAGESAU,ACCOTOB,ICLEVEH,NESSWIT
,EIPTREC,ENSEIMM,UGHTDRO,REMEEXT,IENTANC,UIREINQ,TURYCEN,IZENCIT,CERTCON,LINGCEI
,TIFYCER,PARECOM,EOUSHID,OINEHER,LTHYHEA,ATRETHE,RIERTER,RSTYTHI,THERWHE,STLEWHI
1175 DATAPPEDWRA,STLEWRE
1180 DATAINGSSLGO
1190 DATAAPSELLCO,ROUSMOHU,UNCENORN,BBLEUASQ,NDAHRAVE,TIONXATA,LLELRAPA,ICALYSPH
,GIONLIRE,ERVEESPR,NITEFIDE,SENEROKE,  CENTNOIN,ERSEIVUN,OUGHORTH,ATORDIRA,NTICMA
RO,STERGIRE,ARCHSERE,LYSERAPA,YPUSATPL,HLETMPPA,LIARCUPE,RAITRTPO,EIVERCPE,ACRES
SMA,
1195 DATAIFULRCME,CIANSIMU
1200 DATANCEULAAMB,GUERLOCAT,NZALUEINF,URENATSIG,MMEGRAPRO,ENTMANPER,IZEOGNREC,E
NDOMMREC,SALEARREH,OUSIGIREL,NIAUMOPNE,EGEVILPRI,URECEDPRO,LERPELPRO,REDFERPRE,B
LEERAMIS
1210 DATASIONMISCOM,ABLEOURFAV,ENCESCICON,RATEUSTILL,TIONOLUREV,NERYTIOSTA,ITERE
WRTYP,IOUSTORVIC,TIONUPAOCC,ULARTICPAR,MENTLIAPAR,URRAKABKOO ,DUALIVIIND,ANCEORT
IMP
1220 DATAAGANTRAVEXT,ATIONOCIASS,NDENTEPEIND,UNITYORTOPP,ATIONPARPRE,SIBLEPONRES
,RHAGEMORHAE,OITREONNREC,CABLECTIPRA ,ORATEROBCOR,ATIONORMINF,PMENTELODEV,RFEITN
TECOU,CTIONTINDIS,ANCESITTREM,CIOUSONSUNC
1230 DATANATECTIOAFFE,TIONERSACONV,HIEFKERCHAND,ALLYSIONOCCA,IBLEEMPTCONT,ONEREC
TICONF,CITYNTRIECCE ,ELEDRALLUNPA,TINGUCIAEXCR,ATEDEMORCOMM,ATESUNICCOMM,IOUSNTA
TOSTE,TONECE-SPUMI
1240 DATAATIONMMODACCO,ATIONRMINDETE,ATELYRTUNUNFO,EMENTRTISADVE,TIOUSCIENCONS,A
TIONUNCIPRON,ANCESUMSTCIRC,ATIONSSINASSA,ATIONIDERCONS,SMENTRRASEMBA,IENCENVENIN
CO
1250 RETURN
1260 '************************************
1270 '******* T H A N K  Y O U *********
```