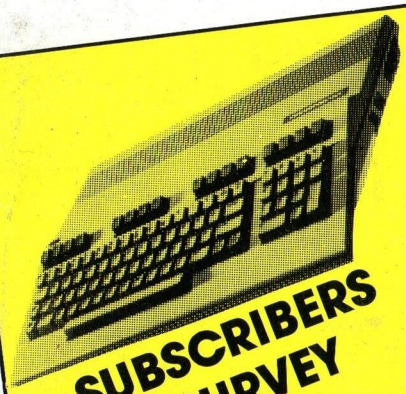
COMMODORE MAGAZINE

VOL 5 NO 5

AUSTRALIAN $4.95
N.Z. $5.20
(Recommended Retail Price)

Issue 33

HOLIDAY ISSUE



CHANCE TO WIN A COMMODORE 128 & DISK DRIVE

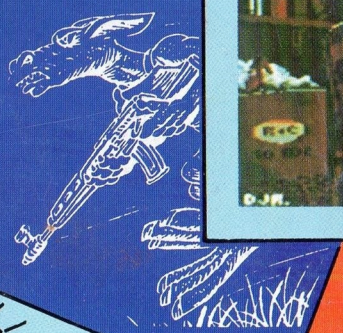
SUBSCRIBERS SURVEY

GRAPHICS LIBRARY

"THE CHESSPLAYER"

complete GRAPHICS GENERATOR

ADVENTURE HELP & HINTS



BOOOM!

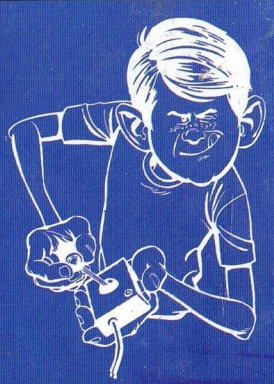
SOUND EFFECTS GENERATOR

SPECIAL OFFER

RACE - CAL

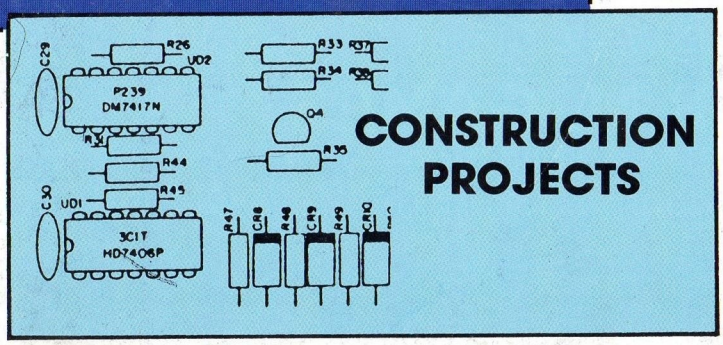
OMPLETE PUNTERS SYSTEM

COMPACTIN(MORE DATA INTO MEMORY



HOLIDAY LISTINGS

- DRAW E.T.
- LOTTO GENERATOR
- ADDRESS BOOK & OTHERS



CONSTRUCTION PROJECTS

ISSN - 8814 - 5741
Registered by Australia Post
Publication No. NBP4288

S.F./FANTASY GAMING & MODELING

PANDORA

MAGAZINE ISSUE 3 Vol 1 No 3

AUSTRALIAN $3.95
N.Z. $4.95
(Recommended Retail Price)

COMPUTER ART
Art On Your Home Computer

AUSSIECON II REPORT
S.F. World Comes To Australia

P.B.M. GAMES
P.B.M. Publications
Indepth Look At The Keys Of Bled

ISSN No: 0815 – 4228
Registered by Australia Post
Publication No. NBQ7439

COMMODORE MAGAZINE

Commodore Magazine is published 6 issue per volume currently 10/12 issues a year (this may alter without notice.) Produced by Mervyn Beamish Graphics Pty Ltd through its division Kim Books. The cost of a one volume subscription (6 issues including p&p) is $A30 within Australia, $NZ36 within New Zealand and $A38 elsewhere. Overseas airmail (including New Zealand) $A8 extra. Subscriptions are available from the Publisher and individual copies from News Agents, dealers and other retail outlets.

EDITOR
Mervyn Beamish

ASSISTANT EDITOR
Lawrence Hulse (IMAGISTICS)

PRODUCTION & DESIGN
Denise Elkins

TECHNICAL WRITERS
Greg Perry & Paul Blair

EDITORIAL ASSISTANCE
Michael Spiteri & David Roth

ADVERTISING
Webster Media Sales
(02) 331 4777

DISTRIBUTION ENQUIRIES
Bookshops & Newsagents
Gordon & Gotch
Computer Stores & Others
KIM BOOKS
(02) 439 1827

PUBLISHER
KIM BOOKS
82 Alexander St., Crows Nest 2065
(02) 439 1827

PRINTER
LANGRIDGE PRESS
52 Gibbes St., Chatswood 2067
(02) 406 6266

PROUDLY PRINTED IN AUSTRALIA

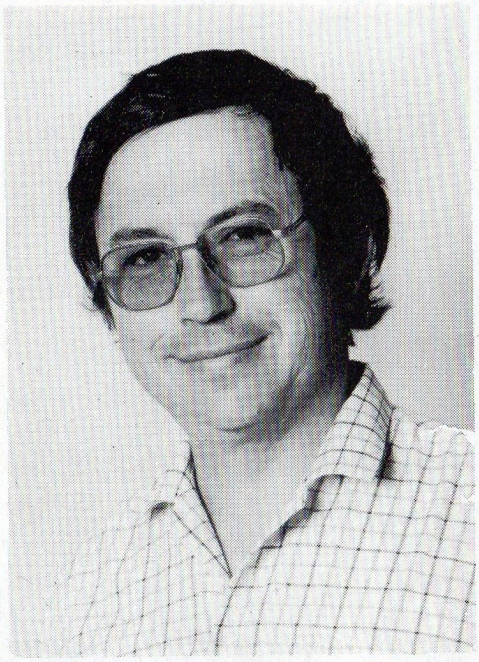
COMMODORE USERS MAGAZINE VOLUME 5 NUMBER 5 ISSUE 33

CONTENTS **PAGE**

ADVERTISERS **PAGE**

EDITORIAL

MERVYN BEAMISH



Mervyn Beamish

The festive season is upon us again and here at Kim Books we're in a total panic. The printer has given us four days to get the artwork to him if we want the magazine out before Christmas. By noting the date that this issue was received you'll know if we made the deadline or not.

What is happening to Commodore?? – the rumour mungers have been really alive this year. AMIGA was supposed to have been taken by Tramiel to ATARI, Commodore was going bust because of loan repayments, Nigel Shepherd was resigning, the Commodore Magazine was in trouble, the Commodore Magazine was being taken over by its competition or the Commodore Magazine was taking over its competition.

Well Commodore is still the leading marketer of home computers in Australia. It is to release the AMIGA and some are already in the country. Nigel Shepherd is still with us and had a heafty push up the promotions ladder. My bank manager is still smiling at me (I think) and Kim Books doesn't own the competition,and the competition does not own Kim books.

All in all things seem to be much the same. That is why we are giving you, the subscriber, a chance to tell us what you like, dislike or would like to see in our magazine. Over the next three issues we will be publishing subscriber survey forms. Not only will these forms give you a chance to put your 'bit' to the editor, but also a chance to win a Commodore 128 and diskdrive.

So if you ever wanted to tell me where to go (figuratively speaking of course!) – here is a golden opportunity.

And a Happy Christmas to you all – thanks for your support.

Mervyn Beamish
Editor

NEXT ISSUE

- ★ Our Annual Adventurer's Issue - Hints, Tips, & Reviews
- ★ A Reset Pulse Switch for the C64
- ★ Measuring the readability of your Prose.
- ★ Sound Effects Generator Number 3
- ★ Graphics Library Picture
- ★ 2nd Survey Sheet

(Chance to win a Commodore 128 & Disk Drive)

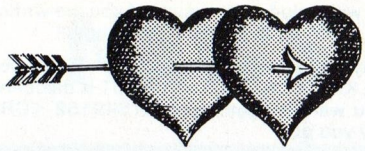
Commodore is spending just over $1 million on a TV campaign, $345,000 on daily press advertising and a further $50,000 on bookings in special interest magazines. Its consumer campaign was developed by agency Beeman, Mayrhofer and Stott, while its business advertising is co-ordinated by Pope, Keirnan and Black.

Much of Commodore's advertising is centred on its $499 Commodore 64 family package, which includes a computer that plugs into a TV set, a cassette data recorder and six software packages.

In the under-$1,000 market, Mr Wood expects Commodore to dominate. The company's research focuses only on dealer sales but a number of units will be shifted through other major retail chains.

The Commodore advertising and promotions manager, Mr Gil Avenaim, said the company expected to sell about 76,000 systems – 40,000 family packs, 20,000 Commodore 16s, 15,000 Commodore 128s and 1,000 PC-10s.

Financial Review December 10th 1985



The perfect match...

COMMODORE RITEMAN

COMPARE THESE SPECS BEFORE YOU BUY...

FEATURES	RITEMAN C+	COMMODORE PRINTERS				
		MPS 801	MPS 802	MPS 803	VIC1525	VIC1526
PRINT SPEED (CPS)	105	50	60	60	50	60
BIDIRECTIONAL PRINT	YES	NO	YES	YES	NO	YES
(COLUMN WIDTH)						
40 CHARACTERS PER LINE	YES	YES	YES	YES	YES	YES
80 CHARACTERS PER LINE	YES	YES	YES	YES	YES	YES
66 CHARACTERS PER LINE	YES					
132 CHARACTERS PER LINE	YES					
(PAPER HANDLING)						
FRONT LOADING FOR EASY PAPER SETTINGS	YES					
BUILT-IN PRINTER STAND	YES					
PRINT ON POST CARDS	YES					
(SOFTWARE COMMANDS)						
DOUBLE STRIKE	YES					
EMPHASIZED	YES					
COMPRESSED	YES					
UNDERLINE	YES					
SUPER/SUBSCRIPTS	YES					
ITALICS	YES					
DOUBLE DENSITY BIT IMAGE	YES					
(CHARACTERS)						
9X9 FONT	YES					
TRUE DESCENDERS	YES					
ITALICS	YES					
COMMODORE GRAPHICS	YES	YES	YES	YES	YES	YES
(OTHER FEATURES)						
SINGLE DENSITY BIT IMAGE	YES	YES	NO	YES	YES	NO
EXPANDED	YES	YES	YES	YES	YES	YES
REVERSE	YES	YES	YES	YES	YES	YES

NO

Plug-compatible with Commodore* computers. 2 software built-in: Commodore* & Epson** compatibility.

If you own a Commodore computer...or are thinking about getting one...you're going to want the Riteman C+ dot matrix printer. You'll really appreciate that added convenience, versatility and economy. Its unique front loading design lets you use plain paper of any thickness, eliminates positioning and aligning problems and keeps continuous-feed paper away from entangling cables and connectors. Just compare the spec. table...complete with a built-in Commodore interface and all necessary cables and connectors...the Riteman C+ is the RIGHT printer for your Commodore system.

*Commodore is a registered trademark of Commodore Business Machines Inc.
**Epson is a registered trademark of Epson America Inc.

Distributed exclusively in Australia by:

anitech
A DIVISION OF THE ANI CORPORATION LTD (INC IN N.S.W.)

ADELAIDE: (08) 356 7333 **BRISBANE:** (07) 275 1766
HOBART: (002) 34 4511 **MELBOURNE:** (03) 795 9011 & 795 5111
PERTH: (09) 277 7000 & 277 1944
SYDNEY: (02) 648 1711 & 648 4088 **AUCKLAND:** (09) 444 2645

WF981/81

READ THIS PAGE
Before entering listings printed in this magazine.

If you want to enter any of our programs without HELPOUT, omit the checksum at each line.

HELPOUT

HELPOUT is a program checker that makes it easier to correctly enter C64 programs from the Commodore Magazine. Once HELPOUT is loaded enable it with SYS49152, type TEST <CR>.

A typical line would be:

6045 NEXT:GOSUB 6300'CFSI

You type ALL of it, remembering to use a single quote ('), not "talking marks". You may use shorthand typing (GO then SHIFT and S for GOSUB, for example), and put in or omit spaces as you like (except, of course, inside quotes).

If you do it wrong, one of six error messages will appear, and a fog-horn will blow.

NO CHECKSUM: You probably forgot the apostrophe, or some or all of the four character checksum. Cursor to the end of the line, enter the checksum as shown, and press Return.

QUOTE: HELPOUT checks to ensure that quotes come in pairs. Either you left one out, or got a bit carried away and added one of your own.

PARENTHESIS: This is like QUOTE, except that HELPOUT looks for pairs of (and).

KEYWORD: You may have misspelled a Basic keyword (GOSLOB instead of GOSUB) or simply left one out. Recheck the line.

#OF CHARACTERS: Ignoring spaces outside quotes, you have typed too many or too few characters. This could even be a typo in a keyword, so check the line carefully.

UNIDENTIFIED: The cowards way out. The line is not right, but the error could be any of a number of things- wrong line number, wrong checksum, or you just mussed it up. Read the line carefully to find the error.

If you do get an error, the line WILL NOT be entered into your program. You will have to take some action to correct it before that can happen. To clear the error message, press any key (the line underneath the error message will not be affected if you use, say, the space bar), then get to work to fix the problem.

Then the line is OK, HELPOUT strips off the checksum, then enters the line into your program. There is no program space overhead from using HELPOUT.

We like HELPOUT. It comes from Mark Robin, and is the program that Commodore uses in its homegrown magazines. With a pedigree like that, and with the very considerable help it gives, we know it will make your programming more enjoyable.

If you want to add checksums to a program of your own, say for this magazine or a User Group publication, load HELPOUT and get it all going. But this time, enter ADD <CR> instead of TEST.

If your program is already written, load it and LIST it to the screen. Using the cursor keys, put the cursor on each line in turn and press Return. You will get a chirp of sound at each line, but nothing will appear on the screen until you type LIST again.

ADD will include checksums if you are writing new lines. Nothing could be more simple!!

Lastly, you may turn HELPOUT off to suit yourself. Type KILL <CR>, and HELPOUT is disconnected. If you want it again, type SYS49152 <CR> and away you go.

```
1000 REM: HELPOUT '64
1010 REM:
1020 REM: COMMODORE MAGAZINE
1030 REM:
1040 SA=49152:FA=50052
1050 FORI=SATOFA:READA:S=S+A:POKEI,A:NEXT
1060 IFS<>103233THENPRINT"->ERROR.. CHECK AGAIN!!":END
1070 PRINT"OK":NEW
1080 DATA  76, 35,192,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0, 13
1090 DATA   0, 88,193, 94,193,102,193,118,193,131,193,143,193,234,234
1100 DATA  76,131,192,162,  5,189, 29,192,149,115,202, 16,248, 96,160,  2
1110 DATA 185,  0,  2,217, 60,193,208, 11,136, 16,245,169,  1,141, 16,192
1120 DATA  76, 31,193, 96,160,  3,185,  0,  2,217, 56,193,208,224,136, 16
1130 DATA 245,169,  0,141, 16,192, 76, 31,193, 96,160,  3,185,  0,  2,217
1140 DATA  52,193,208,224,136, 16,245,160,  5,185,162,227,153,115,  0,136
1150 DATA  16,247,169,  0,141, 24,212, 76, 31,193,230,122,208,  2,230,123
1160 DATA  76,121,  0,165,157,240,243,165,122,201,255,208,237,165,123,201
1170 DATA   1,208,231, 32, 90,192,173,  0,  2, 32,163,192,144,220,160,  0
1180 DATA  76,234,193,201, 48, 48,  6,201, 58, 16,  2, 56, 96, 24, 96,200
1190 DATA 177,122,201, 32,208,  3,200,208,247,177,122, 96, 24,200,177,122
1200 DATA 240, 53,201, 34,240,245,109,  5,192,141,  5,192,173,  6,192,105
1210 DATA   0,141,  6,192, 76,189,192, 24,109,  7,192,141,  7,192,144,  3
1220 DATA 238,  8,192,238, 11,192, 96, 24,109, 10,192,141, 10,192,144,  3
1230 DATA 238,  9,192,238, 12,192, 96, 10,168,185, 17,192,133,251,185, 18
1240 DATA 192,133,252,160,  0,169, 18, 32,210,255,177,251,240,  6, 32,210
1250 DATA 255,200,208,246, 32, 84,195, 32,126,195, 32,228,255,240,251,160
1260 DATA  27,185, 63,193, 32,210,255,136, 16,247,104,104,169,  0,141,  0
1270 DATA   2, 76,116,164, 75, 73, 76, 76, 84, 69, 83, 84, 65, 68, 68,145
1280 DATA 145, 13, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
1290 DATA  32, 32, 32, 32, 32, 32,145, 13, 81, 85, 79, 84, 69,  0, 75, 69
1300 DATA  89, 87, 79, 82, 68,  0, 35, 32, 79, 70, 32, 67, 72, 65, 82, 65
1310 DATA  67, 84, 69, 82, 83,  0, 85, 78, 73, 68, 69, 78, 84, 73, 70, 73
1320 DATA  69, 68,  0, 78, 79, 32, 67, 72, 69, 67, 75, 83, 85, 77,  0, 80
1330 DATA  65, 82, 69, 78, 84, 72, 69, 83, 73, 83,  0,200,177,122,208,251
1340 DATA 132,253,192,  9, 16,  3, 76,199,193,136,136,136,136,136,177,122
1350 DATA 201, 39,208, 19,169,  0,145,122,200,162,  0,177,122,157, 60,  3
1360 DATA 200,232,224,  4,208,245, 96, 76,242,194,160,  0,185,  0,  2,153
1370 DATA  64,  3,240,242,200,208,245,160,  0,185, 64,  3,240,232,153,  0
1380 DATA   2,200,208,245, 32,215,193, 76, 86,194,160, 11,169,  0,153,  3
1390 DATA 192,141, 60,  3,136, 16,247,169,128,133,  2, 32, 27,195,160,  0
1400 DATA  32,155,193, 32,202,193, 32, 49,194,230,122,230,123, 32,124,165
1410 DATA 160,  0, 32,175,192,240,205, 36,  2,240,  6, 32,215,192, 76, 18
1420 DATA 194,201, 34,208,  6, 32,188,192, 76, 18,194, 32,231,192, 76, 18
1430 DATA 194,160,  0,185,  0,  2, 32,163,192,200,144, 10, 24,109,  9,192
1440 DATA 141,  9,192, 76, 51,194,136,162,  0,185,  0,  2,157,  0,  2,240
1450 DATA   4,232,200,208,244, 96, 24,173, 11,192,105, 65,141, 11,192, 56
1460 DATA 173, 12,192,233, 25,144,  6,141, 12,192, 76, 96,194,173, 12,192
1470 DATA 105, 65,141, 12,192,173,  5,192,109,  7,192, 72,173,  6,192,109
1480 DATA   8,192,141, 14,192,104,109, 10,192,141, 13,192,173, 14,192,109
1490 DATA   9,192,141, 14,192, 56,233, 25,144,  6,141, 14,192, 76,150,194
1500 DATA 173, 14,192,105, 65,141, 14,192,173, 13,192,233, 25,144,  6,141
1510 DATA  13,192, 76,171,194,173, 13,192,105, 65,141, 13,192,160,  1,173
1520 DATA  11,192,205, 60,  3,208, 32,200,173, 12,192,205, 61,  3,208, 23
1530 DATA 200,173, 13,192,205, 62,  3,208, 14,173, 14,192,205, 63,  3,208
1540 DATA   6, 32,100,195, 76,122,192,173, 16,192,208, 17,152, 72,104, 76
1550 DATA 247,192,173, 16,192,240,  1, 96,169,  4, 76,247,192,164,253,169
1560 DATA  39,145,122,162,  0,200,189, 11,192,145,122,200,232,224,  4,208
1570 DATA 245,169,  0,145,122, 32,100,195, 76,122,192,160,  0,185,  0,  2
1580 DATA 240, 17,201, 40,208,  3,238,  3,192,201, 41,208,  3,238,  4,192
1590 DATA 200,208,234,173,  3,192,205,  4,192,208,  1, 96,169,  5, 76,247
1600 DATA 192,169, 32,141,  0,212,141,  1,212,169,  9,141,  5,212,169, 15
1610 DATA 141, 24,212, 96, 32, 65,195,169,129, 32,119,195,169,128, 32,119
1620 DATA 195, 76,113,195, 32, 65,195,169, 17, 32,119,195,169, 16, 32,119
1630 DATA 195,169,  0,141,  4,212, 96,141,  4,212,162,112,160,  0,136,208
1640 DATA 253,202,208,250, 96
```

NICE LISTER CONVENTIONS

1. All control, colour, function, and shifted and Commodore key graphics are converted to 'words' (or the abbreviations as represented on the keyboard) enclosed in square brackets [ ].
For example, [DOWN], [CLR] and so on.

2. Multiple cursor controls are represented by one word plus a number. For example, [DOWN 15].

3. Shifted graphics (right-hand symbol on key) are converted to the corresponding alphabet character enclosed in square brackets. A shifted 'S' heart character becomes [S].

4. Any character accessed by the Commodore (C=) key is indicated by further enclosing the alphabet character inside the symbols <>.
A Commodore 'A' becomes [<A>].

5. With multiple characters, the redundant brackets][are replaced by a comma as:-
[CLR,DOWN5,WHT,<A>]

6. With multiple shifted graphics, the alphabet character is simply repeated, numbers are not used as [AAAAAAAAAAA].

7. Multiple Commodore graphics are repeated as [<A>, <A>, <A>, <A>].

8. Spaces and shifted spaces within quotes are represented by the words [SPACE] or [SHSPACE] followed by a number if required.
For example, [SPACE15].

9. Extra words are used for the following control characters.

Keyword	CHR$	
DEL (CRTL-T)	20	
INS	148	
TEXT (CTRL-N)	14	converts character set to upper/lowercase mode.
GRAPH	142	converts character set to uppercase/graphics mode.
LOCK (CTRL-H)	8	disables the C = key and locks the keyboard in the current character mode.
UNLOCK (CTRL-I)	9	enables the C = key sequence.

NEWS & VIEWS

C64 INVOLVED WITH NEW INTELLIGENCE TEST

Doctor Stankov (right) with Research Assistants John Crawford (seated) & Brett Myers. Data from his attention tests are recorded on a C64 for later evaluation. Photo courtesy of University of Sydney-Department of Photography

A Commodore C-64 is an important tool in research at the University of Sydney's Psychology Department which indicates that measuring attention maybe a better way of measuring intelligence than the old written intelligence tests.

A team led by Dr Lazar Stankov, Senior Lecturer in the department, has been investigating the role of attention as a factor of intelligence.

In the experiments, volunteers are faced with complex audio and visual tasks, such as recognising tones or responding to visual clues on a C-64, at the same time. The results are recorded on the C-64 so that accurate data on response times and correct responses can be obtained and evaluated.

Dr Stankov says, "Divided attention tests are much more demanding, than written tests, as they call for more resources to be put into them and they draw out more effort."

Part of the research involves studying how performance may be influenced by practising the divided attention tasks, and the effects of prolonged training in creative thinking on intelligence tests.

It has been discovered that fluid intelligence (problem solving ability which does not depend on formal education) declines relatively rapidly after the age of 30, but crystallised intelligence (intelligence which depends heavily on education) doesn't decline until 60 or 70.

"As people get older they are less able to perform demanding tasks, but the tasks that involved vocabulary skills don't decline at all until the 65-70," he said.

PRIMARY SCHOOLS AND COMMODORE

Yarrambat Primary School and Watsonia Heights Primary School have produced a booklet entitled "A Suggested Approach for Using Commodore Computers In Primary Schools".

The booklet covers many questions associated with computer use in Schools and concentrates in particular on a list of 27 skills/concepts in getting children started.

Much of the booklet is applicable to all brands of computers, but this booklet is also machine specific to the point that it concentrates on Commodore software and details programs seen as appropriate to the skills/concepts being taught.

Cost of the booklet is $6.00 (including postage).

For further information contact
Brendon G. Smith
Yarrambat Public School
Yan Yean Rd., Yarrambat VIC. 3091.

NEW FOR DLM

The Developmental Learning Materials Early Childhood programs, Number Farm, Alphabet Circus and Shape & Color Rodeo, were award winners and proved most popular on the Australian market.

Three new programs have been added to this range and each is just as exciting and educationally valuable as the first three.

Comparison Kitchen makes learning how to compare sizes, shapes and colours an exciting adventure. Children become acquainted with the concepts of large and small, greater than/less than, and same or different.

Animal Photo Fun helps children identify animals, classify animals by habitats and strengthens memory skills. One and two player games are enhanced with music and colourful graphics.

Fish Scales makes it easy to learn how to measure height, length and distance, and to use measurements to compare sizes and distances.

Each of these packages contains six different games, all of which are motivating and fun. Music is lively and familiar and colourful graphics attract the attention of every little learner.

DLM software is distributed in Australia by Dataflow Computer Services and packages are available from your local computer store or school supplier.

SURESHOT SUPREME

The SUPREME version of the popular SureShot joystick has now been released in Australia. The SureShot SUPREME incorporates a CENTRE FIRE button in addition to the long life features built into every SureShot. These features include positive micro switch action, steel shaft, phosphor bronze bearing, nylon actuator, high impact case, and left and right hand fire buttons.

The N.S.W. Department of Main Roads recently chose SureShot joysticks to remotely control the closed circuit television cameras monitoring traffic flow on the Sydney Harbour Bridge. SureShot joysticks were chosen in preference to industrial products costing up to 10 times as much as the SureShot.

Further information,
Dolphin Computers,(02) 438 4933

SureShot Supreme Joystick
Distributed by Dolphin Computers (02) 438 4933

HELP FOR CANCER SUFFERERS

Sydney — Three of Australia's leading computer companies – Imagineering, Datachecker/DTS and Dick Smith Electronics, have joined forces to donate computer equipment and games to NSW children suffering from cancer.

Imagineering, Australia's largest distributor of microcomputer software and peripherals, has donated a Commodore 64 home computer and 30 games to Alpha Committee for the Malcolm Sargent Cancer Fund for Children in Australia. Datachecker/DTS, Australia's leading supplier of point-of-sale equipment, and Dick Smith Electronics, Australia's leading electronic components retailer, have joined to donate a Dick Smith Challenger IBM - compatible personal computer to the same committee.

Both computers and the games were donated to Sydney's Prince of Wales Children's Hospital, where they are being used by teenagers undergoing chemotherapy.

The donations co-incided with a gala fashion parade held on Oct. 10 at the Art Gallery of N.S.W., and organised by the Alpha Committee to help raise funds for the Malcolm Sargent Cancer Fund. A spokesman for the committee, Neil Matthews, said it was the first time computer companies in Australia had assisted this fund.

"We think it's great that these three companies have donated equipment and games which will help teenagers suffering from cancer," Matthews said. "To date, teenagers in either the Prince of Wales Childrens Hospital or the Royal Alexandra Hospital for Children have had little to occupy their time while undergoing treatment."

Imagineering, Datachecker/DTS and Dick Smith Electronics were formally thanked for their donations at the fashion parade - a glittering event at which three of Australia's leading fashion designers, Anne

CONTINUED OVERLEAF

NEWS & VIEWS

CONTINUED FROM PREVIOUS PAGE

Present at the donation at the Prince of Wales Children's Hospital were (from left) Alan Bowman, Jeanne Rockey, Neil Matthews, Joe Harper & Prof. John Beveridge.

Lewin, Lydia Pearson and Ewaldo, showed their coming Summer collections of Lingerie, day and evening wear respectively to an audience of more than 300 people.

Jeanne Rockey, The Malcolm Sargent Cancer Fund's administrator, said more than 71 families had been helped financially in the past two years. "The Fund is primarily interested in providing services for children suffering from cancer, and for their families," she said.

"Although we so not get involved in funding cancer research, we provide things like travelling allowances for children undergoing treatment at either children's hospital, and for their families to visit them. We also provide games and equipment the children can use while in hospital. The fund also sponsors outside activities, like camps, for the children.

"We are particularly grateful to Imagineering, Datachecker/DTS and Dick Smith Electronics because most of the children we help come from battling families, and we know the computers and games will be put to good use by kids wanting to learn as much as they can about computers."

The Malcolm Sargent Cancer Fund was originally formed in the U.K. in 1967 in memory of world-renowned musical conductor Malcolm Sargent, who died of cancer.

MINITAB FOR PC'S

At a recent marketing course at Karingai College I came across MINITAB, simple to use but a mighty powerful data analysis software (from a company called Data Analysis - devilishly imaginative these software companies!)

For most microcomputers, Minitab is available in two forms: Fundamental Minitab and Standard Minitab. Particularly useful in instruction and for preliminary analyses, *Fundamental Minitab* produces scatterplots and histograms and performs the following operations; descriptive statistics, simple and multiple regression, analysis of variance, nonparametrics, crosstabulations, random data generation, and macro and looping capabilities.

Standard Minitab is the same version of Minitab that's available for mainframe and minicomputers. It performs time series analysis, stepwise regression, exploratory data analysis, and matrix operations as well as all the operations found in Fundamental Minitab.

Minitab will run off IBM compatibles. It requires 256K and uses MS-DOS (all versions) operating system. It is hard disk supported and the Maths processor is 8087 (supported). Approximate worksheet size is 8,000 numbers.

For further information:
Data Analysis
215 Pond Laboratory
University Park, PA 16802 USA
Or you could try Wadsworth Publishing who distribute the manuals in Australia.

NEW AUTOMATIC DISK COPYING

Nashua, are now distributing a series of machines which offer solutions to diskette copying problems.

The first is the Nashua Automatic Diskette Copier. The Computer based machine operates as a high speed diskette duplication and verification machine. It takes a blank diskette, verifies it to see it has no manufacturing errors, formats it correctly for the computer concerned, writes information to it from the original diskette and then, having done all that, verifies it to make sure that it is 100% correct.

The machine has a hundred diskette hopper which automatically feeds in the diskette.

The Nashua Automatic Diskette Copier is the ideal machine for copying diskettes for one computer using one format in commercial quantities.

If you are using a multiplicity of formats Nashua have the Series One Duplicator which can be configured in various formats: 5.25" with 48 tracks per inch, 5.25" with 96 tracks per inch, 8" or 3.5" drives and the new high density IBM AT drives.

With a single side IBM diskette the Series One Duplicator can make an exact copy in 36 seconds. A double sided IBM DISKETTE takes 60 seconds, and Apple diskette 24 seconds and a Commodore diskette about the same amount of time.

This is twenty times the speed of manual duplication on an ordinary computer.

The recording uses a technique called HRDT (High Definition Recording Technology) which places the information with great precision at the precise point on the diskette as dictated by the format laid down by the computer manufacturer.

The Nashua Diskette Copier can also be programmed to add protection devices at one of three levels as the diskettes are made.

Nashua also have a new board/software combination called the Formaster Sprint which works with the IBM XT to allow it to produce disk copies in IBM, Commodore or Apple formats. Which means that a software house can use an IBM XT as a host and produce Apple, IBM or Commodore disks on that single machine.

Nashua will either sell the machines outright to major users or make it available through service bureaus in capital cities throughout Australia for people who only need occasional use of such a service.

NEWSROOM

The Newsroom software package, distributed in Australia by Dataflow Computer Services, provides the means for journalists of any age to create stylish, sophisticated publications.

The two disk program alows one to develop newspapers, newsletters, brochures, flyers, information packets and motivating assignment presentations.

Using the built in word processor, two type sizes and five type styles wrap neatly around photos of artwork one has prepared. 600 pieces of artwork are provided on the clip art disk. Each of these can be flipped and, using the powerful graphic tools, and endless variety of pictures can be created.

The comprehensive manual not only gives detailed instructions on how to use the many features of The Newsroom, but also is a complete guide to creating a newspaper, from organising a newspaper staff to telling a story with pictures.

The Newroom is mouse an joystick supporting and, via modem text and graphics can be transferred between previously incompatible computers.

The Newsroom is available now the Apple II+/IIe/IIc and IBM PC machines and is expected for release on the Commodore 64 in January, 1986.

NEWS & VIEWS

POWER CONDITIONING

Online Control Pty Ltd of Artarmon, N.S.W. have appointed an authorized stocking distributor of the Square D Topaz range of power conditioning products for computers and electronic equipment.

Computer systems and other electronic devices are susceptible to malfunction and permanent damage caused by commonly occurring problems on the incoming power supply. These problems include electrical noise, voltage fluctuation and black out.

The Topaz range of power conditioning products provides a cure for power supply problems. The range includes ultra isolators for electrical noise suppression, power conditioners which provide noise suppression and voltage fluctuation protection, and uninterruptable power supplies for black out protection.

Topaz power conditioners are available to protect all sizes of equipment, from 125VA personal computers through to large main frame or industrial electronic installations requiring over 100KVA.

Enquiries: Peter Warde
(02) 43 1313

CLASSROOM COMPUTING

ASHTON SCHOLASTIC are to produce a magazine which they claim will cater for a *complete teaching program.* To achieve this their two popular teacher publications CLASSROOM and CLASSROOM COMPUTING, have been amalgamated thus creating a complete resource magazine for Australian K-7 teachers.

As computers are an accepted part of the school curriculum they feel it appropriate to treat them as an integrated part of the curriculum.

The first new-look CLASSROOM incorporating CLASSROOM COMPUTING will be published in February 1986. To accommodate all the regular articles and special features which appear in both CLASSROOM and CLASSROOM COMPUTING, the new magazine will have 72 pages.

For Further information:
Telephone: (043) 28 3555

THE VIATEL TANGO

A new service enabling dealers and suppliers of computer products to provide information on Viatel, thus significantly reducing their current dealer communication charges as well as dramatically reducing turnaround, is now available through Creative Communications.

Creative Communications, a major videotex consultancy and implementation group based in Sydney, have been developing a range of videotex services for over two years during which time they have helped develop databases for organisations such as American Express, Qantas, Apple, Commodore and Yellow Pages.

Their own large database on Viatel is called Tango.

By simply selecting page *424# to Tango, Viatel users gain access to information on the latest hardware, software and computer shop specials for Apple, Apricot, BBC, Commodore and Olivetti computer products.

There are also special Bulletin Boards for authorised dealers only and directories for the above microcomputer companies providing an extensive computer shop access service of prices, promotions, peripherals and configurations.

This means that all VIATEL users now have direct access to the latest information from suppliers and can message their computer shops who are using their dealer-only Bulletin Boards. A PC Chat Board for all Viatel users provides a facility for specific questions or problems to be answered by the PC Shops.

"Any microcomputer dealer not providing their own information on Viatel is missing out on a unique opportunity to generate well qualified sales leads and communicate directly with prospective customers" says Gerrard Sayes director at Creative Communications

"For around $1,350 per year, a dealer can advertise their services, promote different products, advertise specials each month, take orders via the system and change price details as and when applicable" says Mr Sayes.

For a directory listing, a one page bulletin, updated every quarter is available for $200 per year.

"Unlike a magazine, videotex offers a direct response facility, statistics on how many people have viewed the information and the facility to update the information when applicable, for very little cost".

Mr Sayes stressed that because Creative Communications worked for most of the large suppliers, computer shop information may, within certain guidelines, be linked and integrated with the suppliers master bulletin boards and shop directories on the Tango database.

"Also, Tango is now a known central reference point for micro owners, computer shops and Viatel users, with some frames recording over 300 enquiries per day" said Mr Sayes.

To promote this new service Creative Communications are launching a Summer Buyers Guide for Christmas specials for computer shops, complete with an ordering service, as well as a public trading board for buying and selling PC's.

"These services will be available by the end of November," said Mr Sayes.

Any dealer wishing to find out more information can press *424# to Tango on Viatel or Tel: Mr Sayes on (02) 908 4099

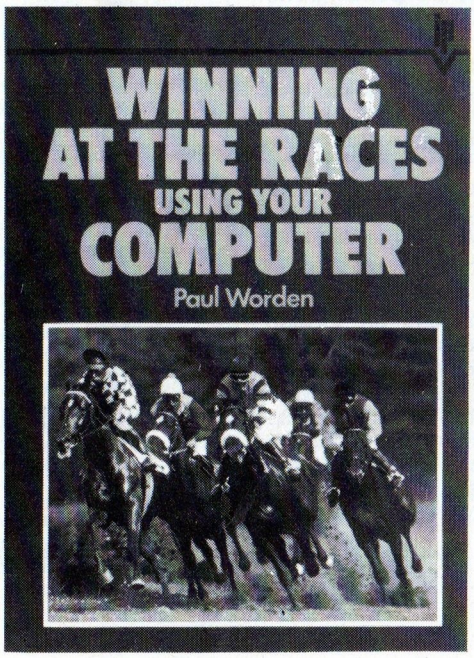
FROM NAGS TO RICHES!!?

WINNING AT THE RACES USING YOUR COMPUTER is an all Australian book on racing systems disigned specificaly for Australian racing conditions.

It is not a book about computers but it does provide insight into how computerised number crunching can discover patterns to turn the odd losing tipple into an educational winning scheme.

Author, Paul Worden bought all the racing systems available, assessed their potential and then developed his own system based upon the systems that he found to be most reliable. All programs are written in microsoft basic and graphics and syntax have been avoided. The programs have been run and tested and even if you have never owned a computer, you will find it totally understandable.

You will be able to analyse odds structures, write systemettes, learn to make a profit out of betting of several horses, write complex evaluation programs and much more. The book contains completed programs on speed rating, multiple betting, equine evaluation and betting simulation as well as short utility programs. Each chapter deals with a different aspect of horse racing as it is applied to personal computers.



The only way to beat the bookmaker is to have better information and process it faster, hence the utilisation of a computer. Worden explains the rules of the systems and application of information and appropriate systems.

While Worden states that it is not foolproof, with an 80% success rate in the past 18 months, it certainly provides an edge on most betting systems used to select the winning combination.

WINNING AT THE RACES WITH YOUR COMPUTER is not written just for computer owners or horse race enthusiasts, it is written for anyone who is interested in game simulation wher an outcome is governed by a large number of factors. R.R.P. $12.95

PAUL WORDEN was born in Plymouth, England in 1943 and moved to Australia in 1959. After a varied career as a meat inspector, technician and more recently, a photographer, Worden is now self employed writing computer programs and a few magazine articles.

He is married and lives in Portland, Victoria and his main ambition is to be self sufficient from his computer racing analysis!

For further information:
Pitman Publishing
158 Bouverie Street
Carlton, Victoria 3053
Tel: (03) 347 3055

PC 86 DATES

Some dates which you may care to put in your memory system occur in March and June 1986.

The Sixth Australian Personal Computer Show will be at the Centrepoint Tower in Sydney between the 12th and 15th March.

Meanwhile, the Seventh Australian Personal Computer Show is being staged at Melbourne's Royal Exhibition Building from June 1-4. Being staged concurrently will be Communications 86 and Office Technology 86.

USER GROUP COLUMN

An Active Time For User Groups

Lawrence Hulse

Although close at hand, the call of the wilds has yet to be heard and while spring fever is being felt throughout the land, there remains high turn out at user group monthly meetings.

However, many group newsletter editors are echoing the familiar cry to group members, "Please submit stories!" An alternative is a letter to the editor. A sure fire way to see your name in print is to tell the disheartened old editor that you really do read the newsletter and all of his time and efforts are appreciated.

FROM QUEENSLAND — It must have been an interesting time at the Mermaid (Business) Users Group (MUGS) recent meeting. Their newsletter reports that, "...meeting held at Col Vernon's residence was a buzz of comings and goings with the subjects varying from the latest in computer desks to wallpapering to the way of Superbase." They may have originated an innovated development in home computers – the colour co-ordinated computer case, which blends with its owner's lounge room furnishings.

The Townsville group has been viewing alot of hardware and software during recent meetings. They have seen the Epson GX80 printer, an opposition machine with a mouse by the tail, as well as the Epyx Fast Load Cartridge, and the "Copy Files" software by the Southport Group. An important achievement is the group has been given free access to the CCUGO's bulletin board.

FROM WESTERN AUSTRALIA — One of the largest independent Commodore user groups in Australia, already with a current membership of about 400 is expanding. The Vic-ups Computer User Group, now with five groups, will soon be comprised of members from six Perth metropolitan groups and country areas. Each group has its own software library and printer. With such a diverse organisation, Chairman and newsletter editor, Alan Stuart is trying to organise lectures or demonstrations about each two months for all members. His idea is to hold total group get together on a Sunday afternoon.

FROM VICTORIA — The Shepparton group's VIATEL demonstration, which members held in conjunction with the local Library, was a great success. "Many people came and saw VIATEL in action and went away very happy with what hey had seen. ... Special thanks to the Library for holding the demonstration with us and also for offering to pay for the afternoon's VIATEL bill," writes Communicator editor Stephen Meddings.

Colin Donald, Waverley Group of Victorian Commodore 64 User's Group, writes of The Age Consumer Electronics Show, "A soccer match in the Exhibition Buildings? Plus a jet Dogfight? As well as a rock concert? yes, I saw all these things at once ... thanks to dealers demonstrating Commodore software. Commodore Business Machines had one of the largest displays, ..." He reports on the software, 'Mind Prober", "This analyses the personality of a subject, e.g. business competitor, and will reveal strengths, weaknesses and likely behaviour in a number of situations."

FROM NEW SOUTH WALES — The 3Gosford Commodore Users' Group (GOSCOM) was recently formed and reports a membership of fifty. So far they have 150 public domain programs available on tape or disk. As publicity manager, Richard Bridge writes, "All Commodore Users are invited to attend, whether they are PET, Vic20, C-64, PC-10 or whatever!"

From the Wollongong Group's newsletter comes some reasons why a computer makes an ideal partner at home. Among the reasons are: every time you turn it on, it's always 'ready'; it obeys your every command, without an argument; a computer responds to logical reasoning and it doesn't snore." There are plenty more, reasons too.

Editor of the Newcastle Technical College's Compu-Tech Computer Club's newsletter 'Victim', Mark Hopkins, reports that the printer evening was a real success. Members present were able to compare such things as speed, size, print quality, interfacing problems and draw on the experience of others. Members have been invited to submit their ideas for a new cover design for the newsletter, and a prize will be awarded to the winning designer. In an editorial, Mark says, "With human nature being what it is, people are always looking for new challenges and new ideas. For those of you who have explored all the facets of software perhaps you should consider a new avenue such as learning assembly language or trying to interface your computer to the real world through some hardware. I guess the main point I am trying to make is the same old one. That is that computers are only limited by our imaginations, ..."

FROM A.C.T — A new editor has taken the helm of the group's newsletter. Greg Weller replaces Dave Hamer and Gavin Lee.

FROM NEW ZEALAND — 'Connection', the Christchurch Commodore Users' Group magazine, reports Dick Anderson, Commodore N.Z.'s Managing Director, saying that an agreement for manufacturing rights to Superbase and Supertype is a real coup and that Commodore N.Z. is the only company in the world to have such rights.

Schools in the Northern Region of Melbourne have set up a Commodore Computers Education Users Group which concentrates on issues relating to computers in Primary Schools.

Dear Sir,
I am writing to you on behalf of Camp Pelican, the camping program of our Church.

I have been appointed as Director of our first "COMPUTER CAMP", to be held in January 1986. We have access to several Commodore C64's, disk drives and printers, but need to build a software library, and need guidance in establishing our program for this camp.

As this is a totally new project for Camp Pelican, we would be grateful for any assistance your company may provide in advising us, or by donating software.
I trust you may see fit to support our Camp Pelican Activities.

Yours Faithfully
Rev. J. G. Noble
St. Mark's Rectory Tarcutta N.S.W.

TEXT COMPACTION

by Paul Markowski

This article is a result of reading a similar article in the British games magazine "The War Machine". The article described a text compaction program written in BASIC and a corresponding decode routine written in 6502 assembler for the BBC microcomputer. I have adapted the decode routine to suit the Commodore 64 and because I don't like to hang around for BASIC, have also implemented the encode routine (which does the actual text compaction) in 6502 assembler as well. All this has been packaged up into a text compaction sytem which will allow you to create text compacted files for use with your own programs. The files consisting of the decode routine and compacted text are loaded under the KERNAL and BASIC ROMs. The files could contain all those messages used in adventure games and thus would free up a lot of RAM for actual program code.

First a little bit about text compaction and the method used in this system. The program uses the Huffman encoding technique to compact and decode the text. As you all know all characters (ASCII ones any way) are rep resented by 1 byte (8 bits) in your comuter. One possible way of saving space would be to store each character as 4 bits so that you could store 2 characters in each byte. This would mean however that you would be limited to only 16 possible combinations of 4 bits and thus could only use 16 characters. What the Huffman technique does is to store the characters that occur most often in the smallest number of bits and to use more bits for those characters that occur least. To accomplish this you need to know the frequency that characters occur in the particular piece of text you want to compact. I have chosen the frequency that letters occur in normal speech as a good compromise for this system.

As you can see from the table, "SPACE" and "E" appear most often, while "Z" and "?" appear the least. This table is also the list of the allowable characters that will be compacted, so if you require numbers you will have to spell them out. What we need to do now is analyse the frequencies in this table and allocate the the smallest number of bits to the characters that occur most often. To do this we build a frequency tree with the least frequent characters at the bottom and the more frequent ones at the top. Diagram 1 represents this tree and this is what we use to generate the binary codes which will represent our characters. Starting at the top we have to go left (this is a "0") first, and then right (this is a "1") and then left again (another "0") to arrive at the "SPACE". Thus the code for a space is "010". By doing this for all the characters we arrive at the following table.

LETTER FREQUENCY TABLE

A - 36	B - 8	C - 14
D - 20	E - 50	F - 12
G - 8	H - 27	I - 35
J - 3	K - 4	L - 18
M - 14	N - 34	O - 34
P - 8	Q - 3	R - 26
S - 34	T - 39	U - 15
V - 6	W - 10	X - 2
Y - 9	Z - 1	? - 1
- 5	, - 7	SPACE - 91

BINARY LETTER CODES

A - 0001	B - 110110	C - 10110
D - 01110	E - 0110	F - 011110
G - 101111	H - 0110	I - 0000
J - 01111110	K - 1101110	L - 00110
M - 10010	N - 1100	O - 1110
P - 101110	Q - 01111111	R - 1000
S - 1111	T - 0010	U - 11010
V - 100111	W - 001110	X - 11011110
Y - 001111	Z - 110111110	, - 100110
- 0111110	? - 110111111	SPACE - 010

TEXT COMPACTION
TEXT COMPACTION
TEXT COMPACTION
TEXT COMPACTION
TEXT COMPACTION

You will see that some characters are represented by 3 or 4 bits while some take up 8 and even 9 bits. These character codes are used to locate the characters they represent in the decode tree that is implemented in the computer. In the computer there are no actual right or left turns. The decode tree has been implemented in the computer as a series of consecutive memory locations. Starting at the first memory loaction (or base address) the program checks the first bit of a particular code and if it is a "0" it adds 1 to the base address, if it is a "1" and the contents of the current memory location is less than 32 (20 hex) then the contents of the memory location is added to the base address and the new address thus calculated is examined. If the new location contains a value greater than 31 (19 hex) a valid character has been found and this is output to the screen and the program starts at the base address again and checks the next bit of the code. This process keeps going until all the message has been decoded. The decode tree has been set up so that each code in the binary letter code table will lead to the correct letter. The binary codes are stored in a continuous stream of bits, so that for the purposes of decoding, the end of a particular piece of code is reached when a valid character is found in the decode tree. Thus every bit, for each byte read, is used to store the decode information.

The program which does the text compaction will also allow you to edit the messages in a text compacted file, print the messages in a file to the screen or the printer and save the text to disc as a file which can be loaded and used by your own BASIC programs. The program consists of 2 parts, a BASIC driver program which handles the screen and input, and a machine code portion which does the text compaction, decode, deleting and inserting of messages. Listing 1 is the BASIC driver and Listing 2 is a BASIC program which will generate a file on disc containing the machine code. This file will be called "TEXT.ASM". After typing in Listing 1 and saving it. Type in Listing 2 save it and then run it. After it has finished load the first program and then load the machine code portion by typing LOAD "TEXT.ASM",8,1. You should then test the program by running it. A Title screen will appear and then a menu which will allow you to select the various options. The top line of the screen is a status line which displays the number of messages currently stored and the amount of memory remaining. If the memory remaining is not 24392 and the number of mess -ages stored is not 0 then you must have made a mistake typing in one of the programs. Test all the options on the menu to make sure everything is functioning properly and then you can save the program and give it any name you like. The next time you want to use it just load the program, the machine code and BASIC are now one unit.

A few instructions on using this system with your own programs. The maximum number of characters you can store in a message is 250. This is no real limitation as if you want to display more you can just save all the text in a series of messages and then display them one after another. If you make a mistake while entering a message you can use the delete key to delete the unwanted characters. To use the text compacted files in your own BASIC programs you must follow some simple rules:-

a. Check Listing 3. The first line of your program must correspond to the first line of this program. Listing 3 is loading a text compacted file called "TEST FILE" (you will use the name of the file that contains your own compacted text). The variable MN% is used to inform the decode routines which message you want to display. Because the decode code and messages use bank switching to store data under the KERNAL and BASIC ROMs a small amount of memory is set aside for the bank switching code, that is the reason for the two pokes.

b. When you want to display a message all you have to do is to issue a PRINT command (if you don't want this message to be appended to the last one), set MN% to the message number you want to display, and then issue a SYS40848 command.

A few warnings. If you try to display a message that is not on file you could at worst end up with a screen full of garbage or a single "?". If you keep getting stray "?" then either you haven't declared the variable MN% first or the text compacted file hasn't loaded correctly. Be careful if you are using RAM between C000 and D000 (hex) as you could overwrite this area of RAM if your text compacted file exceeds 8K. I haven't been able to fool the casette into saving and loading RAM under the KERNAL and BASIC ROMs yet so the system currently only works for the disc drive. If any of you have any clues on saving RAM to casette from this area I would be most grateful to hear from you. Then I could incorporate a cassette option into the system.

Well I hope you find this utility useful. We should see some really interesting adventure games in BASIC now that you have more room for actual BASIC code.

Listing 1: TEXT.BAS

```
10 IF A$<>"D" THEN POKE 55,144: POKE 56,159: CLR :
   MN%=0:MS$="":NA$="": GOSUB 60'LHIL
25 GOSUB 900: PRINT "[DOWN2,RIGHT4,RVS]1[OFF,SPACE]
   INITIALIZE MESSAGE BUFFER"'CEQN
30 PRINT "[RIGHT4,RVS]2[OFF,SPACE]ENTER MESSAGE
   TEXT": PRINT "[RIGHT4,RVS]3[OFF,SPACE]SAVE
   MESSAGES TO DISC"'CBCO
35 PRINT "[RIGHT4,RVS]4[OFF,SPACE]LIST MESSAGES":
   PRINT "[RIGHT4,RVS]5[OFF,SPACE]LOAD MESSAGE FILE
   FROM DISC"'CBJT
40 PRINT "[RIGHT4,RVS]6[OFF,SPACE]FINISH":
   PRINT "[DOWN2,RIGHT4]ENTER OPTION PLEASE"'CBEK
45 GET A$: IF VAL (A$)<1 OR  VAL (A$)>6 OR A$=""
   THEN 45'KRTP
50 ON VAL (A$) GOSUB 100,200,300,400,500,600: GOTO 25'EFUJ
60 PRINT "[CLR,WHT,RIGHT3]TEXT COMPACTION
   PROGRAM": PRINT "[RIGHT6]BY PAUL MARKOWSKI"
   'CBYQ
65 FOR I=1 TO 1000: NEXT : SYS 5632: RETURN 'GNLN
100 PRINT "[CLR,RIGHT3]INITIALIZE MESSAGE BUFFER":
    PRINT "[DOWN2,RIGHT4]ARE YOU SURE Y/N?"'CBOJ
105 GET A$: IF A$<>"Y" AND A$<>"N" THEN 105'IKQH
110 IF A$="N" THEN RETURN 'E◌MY
115 SYS 5632: PRINT "[DOWN2,RIGHT4]BUFFER INITIALIZED":
    FOR I=1 TO 600: NEXT : RETURN 'HNGN
200 GOSUB 900: PRINT "[DOWN2,RIGHT4,RVS]A[OFF]
    DD A MESSAGE": PRINT "[RIGHT4,
    RVS]I[OFF]NSERT A MESSAGE"'DFBI
205 PRINT "[RIGHT4,RVS]D[OFF]ELETE A MESSAGE":
    PRINT "[RIGHT4,RVS]Q[OFF]UIT TO MAIN MENU":
    MS$=""'DFQP
210 PRINT "[DOWN,RIGHT4]ENTER OPTION PLEASE"'BACD
215 GET A$: IF A$<>"A" AND A$<>"I" AND A$<>"D"
    AND A$<>"Q" THEN 215'OOSP
220 IF A$="D" THEN 240'DFKB
225 IF A$="I" THEN 260'DFRG
230 IF A$="Q" THEN RETURN 'ECPC
232 GOSUB 800: IF LEN (MS$)>251 THEN 200'FPRH
235 SYS 6036: PRINT : PRINT "[DOWN]MESSAGE ADDED":
    FOR I=0 TO 600: NEXT : GOTO 200'IRPP
240 GOSUB 900: INPUT "[DOWN]NUMBER OF MESSAGE TO
    DELETE";MN%: IF MN%>C THEN 240'FQVN
242 PRINT : SYS 40848: PRINT : PRINT "[DOWN]IS THIS
    THE MESSAGE Y/N?"'EIVM
245 GET A$: IF A$<>"Y" AND A$<>"N" THEN 245'IKVM
250 IF A$="Y" THEN SYS 6261: PRINT :
    PRINT "[DOWN]MESSAGE DELETED": FOR I=0 TO 600:
    NEXT 'KPEO
255 GOTO 200'BDBH
260 GOSUB 900: INPUT "[DOWN]MESSAGE NUMBER TO
    INSERT BEFORE";MN%: IF MN%>C THEN 260'FQKR
262 PRINT : SYS 40848: PRINT : PRINT "[DOWN]BEFORE
    THIS ONE Y/N?"'EIFN
265 GET A$: IF A$<>"Y" AND A$<>"N" THEN 265'IKXO
270 IF A$="N" THEN 200'DFQG
275 GOSUB 800: IF LEN (MS$)>251 THEN 200'FPRO
280 SYS 6790: PRINT : PRINT "MESSAGE INSERTED":
    FOR I=0 TO 600: NEXT 'HNKP
285 GOTO 200'BDBK
300 GOSUB 900: PRINT "[DOWN,RIGHT4]SAVE MESSAGES
    TO DISC"'CEKF
310 INPUT "[DOWN,RIGHT2]FILENAME PLEASE";NA$'BEFD
320 OPEN 15,8,15: PRINT#15,"S:"+NA$: CLOSE 15:
    OPEN 1,8,5,NA$+",P,W"'GCEJ
325 PRINT#1, CHR$(144) CHR$(159);: SYS 6362: CLOSE 1'FUAL
340 SYS 65484: PRINT "[DOWN,RIGHT4]MESSAGES SAVED":
    FOR I=0 TO 800: NEXT :RETURN 'HOIM
400 GOSUB 900:IF C=0 THEN PRINT"[DOWN]NO MESSAGES":
    PRINT "[DOWN]PRESS ANY KEY TO CONTINUE"'GHTN
402 IF C=0 THEN  GET A$: IF A$="" THEN 402'HKRG
403 IF C=0 THEN  RETURN 'ECDD
404 PRINT "[DOWN,RIGHT4]LIST TO[SPACE,
    RVS]S[OFF]CREEN OR[SPACE,RVS]P[OFF]RINTER"'BAVL
405 GET A$: IF A$<>"S" AND A$<>"P" THEN 405'IKPK
410 IF A$="P" THEN 465'DFGC
```

CONTINUED OVERLEAF

TEXT COMPACTION CONTINUED

```
415 PRINT "[DOWN,RIGHT3]ALL MESSAGES Y/N?""BAYJ
420 GET A$: IF A$<>"Y" AND A$<>"N" THEN 420'IKQH
425 IF A$="Y" THEN 435'DFMI
430 INPUT "[DOWN,RIGHT4]MESSAGE NUMBER PLEASE";
    MN%: IF MN%>C THEN 430'EMAM
432 PRINT "[DOWN]"MN%;: SYS 40848: PRINT : GOTO 461
    'EPGI
435 J=0: GOSUB 900: PRINT "[DOWN]": FOR I=1 TO C:
    MN%=I: PRINT MN%;: SYS 40848: PRINT :J=J+1'MHPV
440 IF J<20 THEN GOTO 460'EGEF
445 PRINT "[RIGHT4]HIT ANY KEY FOR MORE":J=0'CDKO
450 GET A$: IF A$="" THEN 450'EIIH
455 GOSUB 900: PRINT "[DOWN]"'CEKK
460 NEXT I'BBCE
461 PRINT "[DOWN,RIGHT4]PRESS ANY KEY TO CONTINUE"
    'BAZN
462 GET A$: IF A$="" THEN 462'EILK
463 RETURN 'BAQH
465 OPEN 1,4: PRINT#1, CHR$(14)"MESSAGES" CHR$(15):
    PRINT#1'FQNR
470 FOR I=1 TO C:MN%=I: PRINT#1,MN%;: SYS 6447'GUUN
480 SYS 65484: PRINT#1: NEXT I: PRINT#1: CLOSE 1:
    RETURN 'GOJM
500 GOSUB 900: PRINT "[DOWN,RIGHT4]LOAD MESSAGES
    FROM DISC"'CERH
510 INPUT "[DOWN,RIGHT2]FILENAME PLEASE";NA$'BEFF
515 A$="D": LOAD NA$,8,1'CKGI
600 PRINT "[DOWN,RIGHT4]HAVE YOU SAVED YOUR
    MESSAGES Y/N?""BAQK
605 GET A$: IF A$<>"Y" AND A$<>"N" THEN 605'IKVM
610 IF A$="N" THEN  RETURN 'ECME
615 PRINT "[CLR]": END 'CBFH
800 GOSUB 900: PRINT "[DOWN,RIGHT4]ENTER MESSAGE
    PLEASE":MS$="": POKE 204,0'EOCN
805 GET A$: IF A$< CHR$(13) OR A$> CHR$(90) THEN 805
    'ISIP
810 IF A$> CHR$(64) AND A$< CHR$(91) THEN 820'HPWK
815 IF A$<>"?" AND A$<>"," AND A$<>"." AND A$<>
    CHR$(13) AND A$<> CHR$(20) AND A$<> CHR$(32)
    THEN 805'WCED
820 IF A$=CHR$(13) THEN A$="*": GOTO 828'GMHK
825 PRINT A$;'BDGK
827 IF A$=CHR$(20) THEN MS$=LEFT$(MS$,( LEN (MS$)-1)):
    GOTO 805'JCQW
828 MS$=MS$+A$'CIWP
830 IF LEN (MS$)>251 THEN PRINT "[DOWN,RIGHT2]
    MESSAGE TOO LONG": FORI=0 TO 600: NEXT :
    GOTO 840'KTFT
835 IF A$<>"*" THEN 805'EFTN
840 B=FRE (0): POKE 204,255: RETURN 'ENKL
900 C=PEEK (40959)*256+ PEEK (40958):
    D=65535-( PEEK (40957)*256+ PEEK (40956))'LTDT
905 PRINT"[CLR]MESSAGES =";C;"[SPACE2]BYTES FREE = ";D:
    RETURN 'CGWR
```

Listing 2: TEXT.ASM/GEN

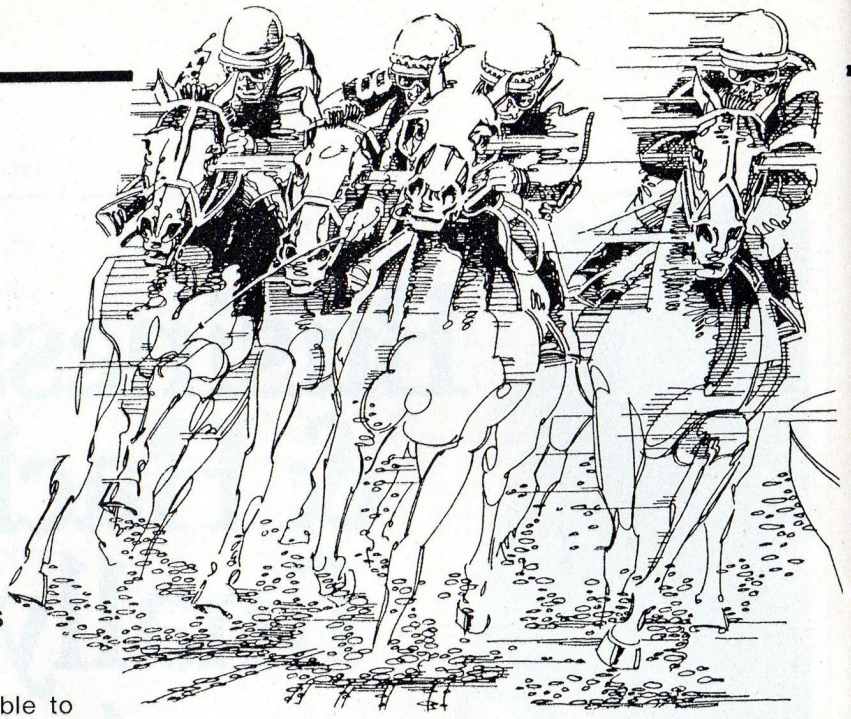
```
5 PRINT "[CLR]SAVING TEXT.ASM":ADD=5632:
  HB=INT (ADD/256):LB=ADD-HB*256'IFAU
10 OPEN 15,8,15: PRINT#15,"S:TEXT.ASM": OPEN 1,8,1,
   "TEXT.ASM,P,W"'DSJI
20 PRINT#1, CHR$(LB) CHR$(HB);: READ M'ENID
30 IF M=-1 THEN CLOSE 1: CLOSE 15: PRINT "FINISHED":
   END 'IICH
40 PRINT#1, CHR$(M);'CGXC
50 READ M: GOTO 30'CEYC
1000 DATA   72,138, 72,152, 72,169,  0,141,  0, 21,169,144,133, 34,169,159'BETD
1001 DATA  133, 35,169, 95,133, 36,169, 25,133, 37,169,133,133, 38,169, 26'BGIF
1002 DATA  133, 39, 32, 43, 22,104,168,104,170,104, 96,160,  0,165, 39,197'BFFG
1003 DATA   37,240, 17,177, 36,145, 34,200,208,243,230, 35,240,  4,230, 37'BFTH
1004 DATA  208,235, 56, 96,177, 36,145, 34,200,196, 38,144,247,240,245, 24'BHBI
1005 DATA   96,234,234,234,234,234,234,234,234,234,165, 39,197, 37,240'BKPK
1006 DATA   22,160,  1,136,177, 38,145, 34,152,208,248,165, 39,197, 37,240'BGVK
1007 DATA    6,198, 39,198, 35,208,236, 56,165, 34,229, 38,133, 34,165, 35'BESL
1008 DATA  233,  0,133, 35,164, 38,169,  0,133, 38,200,136,177, 38,145, 34'BELL
1009 DATA  196, 36,208,247, 96,160,  9,177, 45,141,255, 20,200,177, 45,133'BGXN
1010 DATA   34,200,177, 45,133, 35,160,  0,169,  8,141,254, 20,169,  1,141'BDAE
1011 DATA  253, 20,177, 34,140,252, 20, 32,223, 22,172,252, 20,200,204,255'BHWG
1012 DATA   20,144,239,234,234,173,254, 20,201,  8,240,  3, 32,247, 23,173'BFQH
1013 DATA  253, 20,160,  0,153,  0, 21,152,172,253, 20,153,  0, 21, 96,201'BCWH
1014 DATA   65,176, 22,201, 63,176, 24,201, 46,176, 24,201, 44,176, 24,201'BFXJ
1015 DATA   42,176, 24,201, 32,176, 24,144,  6,216,233, 60,170,208, 18,162'BFXK
1016 DATA    4,208, 14,162,  3,208, 10,162,  2,208,  6,162,  1,208,  2,162'BYSJ
1017 DATA    0,189,117, 23,141,249, 20, 41, 15,141,251, 20,160,  0,189, 86'BDAL
1018 DATA   23, 42, 46,250, 20,206,254, 20,208,  3, 32, 61, 23,200,204,251'BDBM
1019 DATA   20,144,  1, 96,192,  8,144,233,173,249, 20,208,228, 72,152, 72'BERN
1020 DATA  173,250, 20,172,253, 20,153,  0, 21,200,140,253, 20,169,  8,141'BFDF
1021 DATA  254, 20,104,168,104, 96, 64,240,152,124,223, 16,216,176,112, 96'BIDH
1022 DATA  120,188,160,  0,126,220, 48,144,192,224,184,127,128,248, 32,208'BJFI
1023 DATA  156, 56,222, 60,223,  3,  5,  6,  7,137,  4,  6,  5,  5,  4,  6'BQTG
1024 DATA    6,  4,  4,  8,  7,  5,  5,  4,  4,  6,  8,  4,  5,  4,  5,  6'BGHF
1025 DATA    6,  8,  6,  9, 72,138, 72,152, 72, 32,149, 22,173,252,159,133'BABK
1026 DATA   34,173,253,159,133, 35,169,  0,133, 36,169, 21,133, 37,173,  0'BEOL
1027 DATA   21,133, 38,169, 21,133, 39, 32, 43, 22,176, 29,136,152,216,109'BFGN
1028 DATA  252,159,141,252,159,173,253,159,105,  0,141,253,159,176, 10,238'BKIP
1029 DATA  254,159,208, 23,238,255,159,208, 18,173,252,159,133,247,173,253'BLCQ
1030 DATA  159,133,248,169,  0,168,145,247, 32, 66, 25,104,168,104,170,104'BIPH
1031 DATA   96,234,234,234,234,234,234, 14,250, 20,206,254, 20,208,248, 32'BIVI
1032 DATA   61, 23, 96, 72,138, 72,152, 72,173,  0, 21, 72,216, 24,109,252'BCMI
1033 DATA  159,133, 34,173,253,159,105,  0,176, 81,133, 35,173,252,159,133'BIOK
1034 DATA   38,173,253,159,133, 39,165, 34,141,252,159,165, 35,141,253,159'BJCL
1035 DATA  165,247,133, 36,165,248,133, 37,120,169, 52,133,  1, 32, 91, 22'BFGM
1036 DATA  169, 55,133,  1, 88,169,  0,133, 36,169, 21,133, 37,104,133, 38'BDUM
1037 DATA  198, 38,169, 21,133, 39,165,247,133, 34,165,248,133, 35, 32, 43'BGBO
1038 DATA   22,238,254,159,208,  9,238,255,159,208,  4,104, 32, 66, 25,104'BFTP
1039 DATA  168,104,170,104, 96, 72,138, 72,152, 72, 24,165,247,133, 34,165'BHRQ
1040 DATA  248,133, 35,120,169, 52,133,  1,160,  0,177,247,170,101,247,133'BHSI
1041 DATA   36,165,248,105,  0,133, 37,173,252,159,133, 38,173,253,159,133'BIRJ
1042 DATA   39, 32, 43, 22,169, 55,133,  1, 88,176, 38,134, 35, 56,173,252'BCGJ
1043 DATA  159,229, 35,141,252,159,173,253,159,233,  0,141,253,159, 56,173'BJYL
1044 DATA  254,159,233,  1,141,254,159,173,255,159,233,  0,141,255,159,176'BJVM
1045 DATA    3, 32, 66, 25,104,168,104,170,104, 96, 72,138, 72,152, 72,162'BENM
1046 DATA    1, 32,201,255,160,  0,169,144,133, 36,169,159,133, 37,205,253'BGBO
1047 DATA  159,240, 29,120,169, 52,133,  1,177, 36, 72,169, 55,133,  1, 88'BDDO
1048 DATA  104, 32,210,255,200,208,236,230, 37,165, 37,205,253,159,144,227'BKQR
1049 DATA  120,169, 52,133,  1,177, 36, 72,169, 55,133,  1, 88,104, 32,210'BDJQ
1050 DATA  255,200,204,252,159,144,233,240,231,104,168,104,170,104, 96, 72'BLLK
1051 DATA  138, 72,152, 72,162,  1, 32,201,255, 32,161,159,104,168,104,170'BHTK
1052 DATA  104, 96, 32,204,255,162, 14,189, 80, 25, 32,210,255,202,208,247'BHCL
1053 DATA   96, 89, 82, 79, 77, 69, 77, 32, 69, 82, 79, 77, 32, 79, 78, 72'BWDK
1054 DATA  138, 72,152, 72, 32,204,255, 32,161,159,104,168,104,170,104, 96'BIFN
1055 DATA  160,  0,177, 45,201,205,208, 34,200,177, 45,201,206,208, 27,200'BHGO
1056 DATA  177, 45,141,251,159,200,177, 45,141,250,159,120,169, 52,133,  1'BIEP
1057 DATA   32,  0,160,169, 55,133,  1, 88,144,  8, 32,204,255,169, 63, 32'BBPP
1058 DATA  210,255, 96, 72,169, 55,133,  1, 88,104, 32,210,255,120,169, 52'BFJR
1059 DATA  133,  1, 96,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0'BJON
1060 DATA    0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,183,160,  0,  0,169'BMOG
1061 DATA  183,133,247,169,160,133,248,169,  1,141,248,159,169,  0,141,249'BJBL
1062 DATA  159,173,249,159,205,251,159,208, 13,173,248,159,205,250,159,208'BMDN
1063 DATA    5, 32, 70,160, 24, 96,160,  0,177,247,240, 19, 24,101,247,133'BDDM
1064 DATA  247,144,  2,230,248,238,248,159,208,215,238,249,159,208,210, 56'BKXP
1065 DATA   96,234,234,234,234,160,  1,162,  0,177,247,141,247,159,169,128'BIUP
1066 DATA   72, 45,247,159,208,  3,232,208,  6,138, 24,125,122,160,170,189'BGNQ
1067 DATA  122,160,201, 32,144, 11,201, 42,208,  2,104, 96, 32,212,159,162'BGXR
1068 DATA    0,104, 74,208,219,200,208,209, 96, 26, 12,  4,  2, 73, 65,  2'BYJQ
1069 DATA   84,  2, 76,  2, 87, 89,  2, 32,  2, 69,  2, 68,  2, 70,  2, 46'BPOQ
1070 DATA    2, 74, 81, 16,  8,  2, 82,  2, 77,  2, 44, 86,  2, 72,  2, 67'BPHI
1071 DATA    2, 80, 71, 14,  2, 78,  2, 85,  2, 66,  2, 75,  2, 88,  2, 90'BPEJ
1072 DATA   63,  2, 79,  2, 42, 83, 72,138, 72,152, 72, 32,149, 22, 32,  3'BWVL
1073 DATA   24,104,168,104,170,104, 96, 44,-1'BGPJ
```

Listing 3: TEST

```
10 MN%=0: IF A<>1 THEN POKE 55,144: POKE 56,159:A=1:
   LOAD "TEST FILE",8,1'JDLL
15 INPUT "MESSAGE NUMBER PLEASE";MN%'BEPJ
20 IF MN%=0 THEN END 'EEIB
30 PRINT : SYS 40848: PRINT : GOTO 15'EKQD
```

RACING-CAL DESIGNER'S NOTES

by Rob Harwood



The notion of having a computer that is able to predict the outcome of a horse race must have crossed most people's mind at some stage.

The Race-Cal programme unfortunately is not able to transform your computer into such a machine.

Now that I have shattered all of your illusions let me explain exactly what this powerful piece of software is capable of doing. The number of factors governing the outcome of a horse race are innumerable.

Some of those factors are incalculably diverse that it would be impossible to include them in a mathematical equation. There is however a large amount of information available to the public which we are able to convert into figures, enter into the computer and come up with an assessment of how a particular horse should perform, all things being equal. All the information you need to operate the programme is available in sport's sections of newspapers and of course racing publications. When you assess a horse without the aid of Race-Cal you must mentally weigh up all of the given information and formulate in your own mind just how this horse compares with other horses in the same race. When you consider the number of horses in a race this becomes quite a formidable task. In Race-Cal this task is greatly simplified and because of this you are able to think in greater depth about some of the governing factors.

On using the programme you will be requested to make a personal assessment on some apects of the horse you are rating, the remaining information can be entered directly from the newspaper.

Once all of the information has been entered the screen will display a rating figure for the current horse that you are rating. This figure is your guide and you will instantly be able to compare one horse against another.

As you can imagine writing a programme of this complexity is not something that can be achieved without research and also gathering information from people more knowledgeable than I in the world of horse racing. Following the initial research a comparative, relative mathematical structure began to take shape.

This structure was relatively small in comparison to the structure that is currently in Race-Cal.

Initial testing over a period of weeks showed promise and therefore I became motivated to find ways of improving the validity of the ratings.

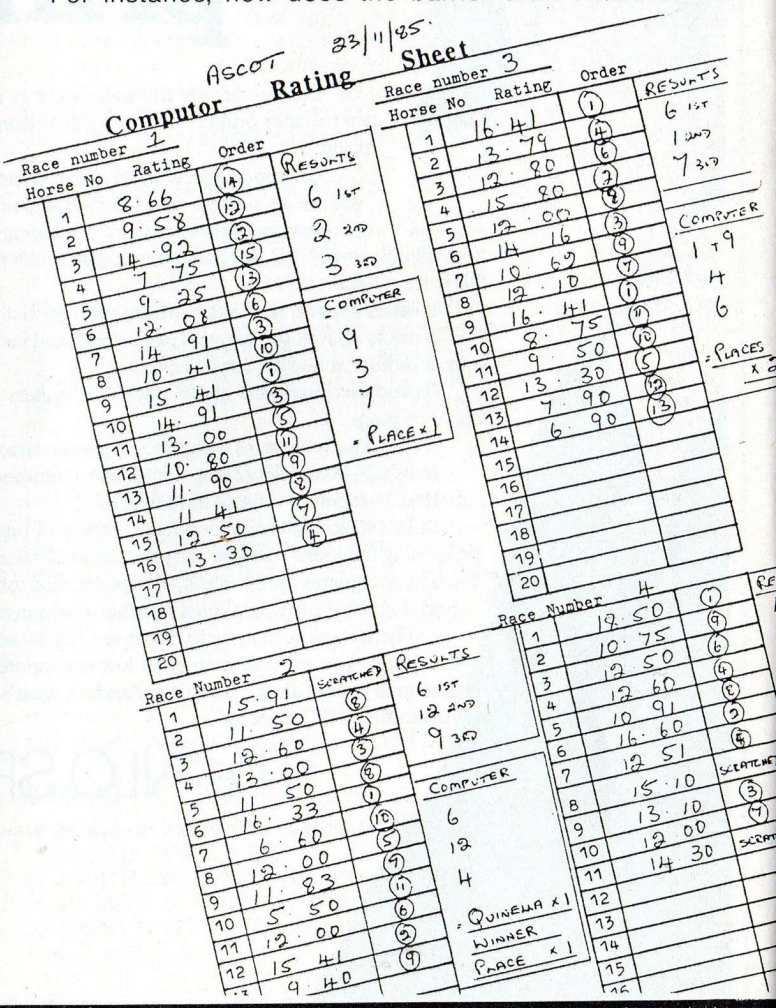
Since all of the information needed to operate the programme is derived from newspapers I contacted a sport's writer who also runs a tipping segment in the newspaper he writes for.

On the other side of the coin I discussed the programme with a racing personality involved in the Australian racing scene.

Advice and information given by these people proved invaluable as their contributions increased the number of variables that the computer could take into account and therefore increased its reliablity factor.

Entering the information into the computer is one thing, working out how they related to each other was another.

For instance, how does the barrier draw relate to the



horses weight? What is the relationship between the odds, past form, jockey and trainer?

The relationship of these factors is of primary importance as of course this is the heart of Race-Cal.

So it was back to the testing stage, over the next few months many subtle changes were made to the comparative structure, each step enhancing the reliability factor.

Throughout the final stages of development I decided to introduce a qualitive control element during testing.

The computer ratings were made available to a number of people prior to the race having been run.

These people were not told which horses to back but were given the computer rating figures when making their own assessment of a particular race.

Results from these tests were more than satisfactory and so that structure became the basis for Race-Cal.

After speaking to people who have used Race-Cal over a period of time something interesting became evident. People who previously had a superficial knowledge about the qualities of jockeys, trainers and horses began to develop an in-depth knowledge about these and other aspects of horse racing.

This educational element was not an intentional inclusion on my part, however this bonus element can only improve the quality of the user's assessments.

Race-Cal is simple yet effective to operate. It encourages you to think about all of the factors in a race and lets you evaluate them in a clear logical way. The factors that we will never be able to include in an evaluation are things such as racing tactics of other riders, how good a start a horse gets and so forth.

Remember that in all gambling sports there is an element of LUCK and that is something you can't calculate but I hope you have some.

ASCOT 23/11/85.

RACE - CAL. BARRIER RATING.

BARRIER NUMBER

STANCE	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
000																		
100	9	9	6	8	10	5	7	8	8	8	7	8	6	4	4	2	2	2
200	8	7	7	8	6	10	8	7	8	6	5	4	4	3	4	2	2	2
400	9	8	4	7	9	8	7	10	4	6	8	6	4	2	3	2	2	2
450																		
500																		
600	10	6	7	7	8	6	5	6	3	6	3	6	4	2	2	2	2	2
00	8	7	7	6	7	8	4	10	9	10	3	6	4	3	2	1	1	1
0																		
)																		
	6																	

3 CLEANAWAY GRAD

TAB TRIFECTA

$8500 1800 metres.

1 3 3 5c 2 EASTERN FLIGHT tm/4
2 7c 0 4c 2c GALWAY UNION 12 ..... Stephen Miller 5
3 8c 5 6 4 MAN OF STEEL 14 ..... P. Dyson 5
4 6 3c 3 1c NORTHWARD BOUND 13 ..... L. Millington 5
5 0 8c 8c 3c PARDITO 3 ..... C. Wake 52.
6 7c 0 5 3 PORTIQUE 2 ..... F. Harris A3.0 52.
7 0 0 0 2c PRINCE REVE 8 ..... R. Franklin 52.0
8 0 x 5 0 SANDALAY 5 ..... D. Miller 52.0
9 4 2 8 8 SHANNONDOI b 10 ..... R. Kemp 52.0
10 0 8c 5c 0 THE PERSIAN JAGUAR s 7 ..... M. Sestich 52.0
11 6 4 4 0 TREETALK a 11 ..... B. Trinder A1.5 52.0
12 9 8c 4c 3c VALE IMP 1 ..... V. Sigley A8.0 52.0
13 0 5c 8c 0 BROOKLET 6 ..... G. Donnelly 52.0
14 0 x 0c 7c GOLDWIN BLACKBIRD 9 ..... J. Hustwitt 52.0
A. Hughes 50.5

5/2 Eastern Flight, 7/2 Northward Bound, 5 Shannondoi, 6 Galway Union, 10 Portique, Man Of Steel, 16 Tree Talk, 14 Sandalay, 25 Pardito, Prince Reve, 100 Vale Imp, Brooklet, 200 The Persian Jaguar, Goldwin Blackbird.

4 City Of Belmont - Quality

$20,000. 3-yr-olds. 1600 metres.

9/2 DANCE ACT — Evans
Custodians Pty Ltd's ch f
Denton (USA)-Amber
Oro 3 51.5 J. Miller 1
7 EXECUTIVE CLASS 4 2
51.0 L. Millington
7/4 *WALENSKA 2 57.0 R. 3
Kemp
Valen..., Valen...; hd. 1:39.28. Winner trained by K.C. Leach. Scr.: Fair Estimation. Lahore. TAB Nos: 4, 6, 1. TOTE: Abt 5/1. *Denotes favourite.

100 Adventure World 9 50.5 4
S. Sheehy
11/2 Bold Torque 5 51.0 5
Stephen Miller
12 Beau Champagne 6 52.0 6
M. Sestich
7 Lady Saraa 1 53.5 P. 7
Dyson
12 Hide 8 50.0 A. Hughes 8
20 Haremedal 7 car. 50.5 D. 9
Gundry
S.O.: $2.90; pl. 90c. $1.10. 65c.
QUINELLA: $9.35. DOUBLE: $97.
TRIFECTA: $122.10. TIERCE: $9.50.
TAB FORECAST: $21.70.
FLUCTUATIONS: Dance Act 9/2, 5. 2. 7/4; Executive Class 6, 8, 7; Walenska 9/2; Executive World 50, 100; Bold Torque 7, 11/2; Beau Champagne 10. 14. 12; Lady Saraa 6, 8, 7; Hide 14. 15. 13; Haremedal 10, 20.

PLACES x 3

LETTERS TO THE EDITOR

Address letters to: The Editor, Commodore Magazine, Kim Books 82 Alexander Street Crows Nest 2065

ILLAWARRA BBS

Dear Sir,
The information in the Vol 5 No 3 issue of Commodore Magazine regarding the Illawarra BBS was only partly correct. I don't know who gave you the information so I can't blame anyone.

The Illawarra BBS started operations in July 1985 and went to 24 hour operation in late August 1985 using 2 X 1541 drives and an Avtek automatic answer modem. September '85 saw another upgrade to 1 only SFD 1001 one Megabyte Drive.

The system not only caters for Commodore 64 users, but any and all types of computers, with seven different makes with software downloads for each type available.

The next update will be online games on Saturday/Sunday as these days, at this time are the quietest.

The telephone number to this Wonderful and Glorious System is: (042) 84 4354

Would you also ask your readers to listen to the voice and the information that is being given as when I realised what was happening I put a message on the answering machine that is on my voice phone regarding the correct telephone number for the IBBS but to no avail. Either I was blasted with a carrier straight away or when the voice answered the person(s) simply hung up the phone. To my mind the height of rudeness.

Enough of my complaints, I like the magazine and I wish you all the best for the coming Festive Season.

Yours Faithfully
John Simon
IBBS

CLASSIFIED ADVERTS

To the Editor: (Mr. Beamish)
May I please suggest an idea to you that I have seen in some of Commodore's English magazines that I think could be successful in your magazine.

My idea is that you could start up a classified section on the last few pages. In England they have been a big success. People could write in what they would like to buy, sell or even write addresses for Commodore User Pen-Friends.

As I said before this is only an idea but I would like to see you go through with it very much.

My name is Damien Beeby and I am 12 years of age. Could you please reply.

Yours Faithfully
Damien Beeby
Healesville VIC.

ED – Good Idea Damien!. We will publish subscribers Non-Commercial Classified adverts free for the time being and see how it goes.
Thanks for your interest.

FAST-DISK/BASAD/DOS 5.1

Dear Mr Beamish,
Paul Blair put me in a quandary with his excellent programs Fast-Disk and Basad. They clash in memory and Basad, of course, occupies the whole block.

However, DOS 5.1 supplied with the 1541 is quite a versatile system. As it starts at 52224 it would also clash with Fast-Disk starting at 52480. I have lowered Fast-Disk by 12 pages so that it now starts at 49408 which fits in nicely with DOS 5.1 with space each end for its operation. As DOS 5.1 is a wedge its symbols work quite satisfactorily through the Fast-Disk routine.

I have also coupled Fast-Disk and DOS 5.1 in the Basic loading program so for a couple of extra seconds have them both constantly available. On the few occasions I need to quit DOS, for example, to use the symbols elsewhere, Fast-Disk is still available.

Incidentally, from talks with some local users, I feel there may be many who do not understand the full facilities of DOS 5.1. Particularly ↑ for load and run and % to load m/l programs I find very useful. I have a Datassette permanently attached to the C64 along with the 1541 as I use tape for backup of new utilities and most other new programs. As you know if you specify the device number with @#1, DOS 5.1 works with tape. Obviously no change in speed is available, but otherwise your Fast-Disk seems quite transparent to the tape operation. I am happily enjoying a quite reasonable DOS together with really highspeed disk loading.

On the question of DOS 5.1, do you think it may be worth a small box in a future edition setting out the full range of commands and facilities of DOS 5.1?

Finally, my Fast-Disk does not operate when the printer is switched on. Its an 801 on the same serial line. The disk runs but never stops at the normal end of the program.

Why, I don't know, but the solution is easy - keep the printer off except when needed. In any case I always include an "Is printer on" prompt in my programs, simply in an attempt to avoid error stoppage which is likely to be a problem to get going again.

Best wishes to you and the team.

Tony Atkinson
Croydon VIC.

ED – Thanks for the information in your letter. I'm sure the readers would be very interested in your relocation listings.

Can we have a copy for publication please?

ADVENTURE CLUB

Dear Merv,
I am writing to advise you that my club, "Adventure News" is finally up and running. The club is Australia wide, and for Commodore 64, SX64 and C-128 users only, and really of use only to adventurers.

A copy of the club newsletter is enclosed. If you could give the club a mention in the Commodore Magazine, I would be extremely grateful.

Way back in Vol 4 No 4, you mentioned that any club that registers with you would receive 2 copies of Commodore Magazine, in exchange for the club newsletter or details of their activities. Does this offer still stand?

Yours from the adventurers pit
Stuart Elflett
MSF 550,
Toogoolawah, Qld 4313

ED – Adventure News is very well written and I think it will be of interest to a number of our readers.

Yes the Free Club Subscription is still current (only one copy now though) and you are on our list. Good luck!

Dear Mervyn,
I get the Commodore Magazine because I have a Vic 20 and was wondering if you could help me.

We were told Commodore will not be making computers anymore. Is this true?

If it's not true we will probably be buying a Commodore 128, Commodore 64 & Disk Drive or an Apple IIC.

Which one should we get?
Confused - Craig Chapman
Lorn N.S.W.

ED – Did I read correctly – an APPLE IIC!! Oh Dear Dear No! That's not civilised. The 128 is a good machine and I suspect a lot Cheaper. Unfortunately the Vic-20 is no more for this world but a good contact is:
ACT VIC-20 USERS ASSOCIATION
25 KERFERD ST.
WATSON A.C.T. 2602

LISTINGS ON DISK - TOUCH TYPE/DECIMALS

Dear Mervyn,
If a key other than a numeral is pressed when entering a numeric variable the "redo from start" error signal appears after "RETURN". This can be avoided by entering a string variable e.g. x$ and getting the value by x = VAL(x$). A check for errors in the string is fairly easy and the attached program 1 does this.

However, if you are learning touch typing, as I am I you are entering decimal numbers it is easy to hit a comma instead of a full stop. This is disaster as the "extra ignored" message appears and only those figures before the comma are accepted as the input. I have tried to stop this by redirecting the error vectors 768, 769 but this message does not appear to got this way. Do your machine language experts have a method to

CONTINUED ON PAGE 18

A HOLIDAY PROGRAM

Paul Blair

This one is for the younger fry, to help pass the time away. It uses sprites in a quite cunning way. No more clues than that.

The listing uses NICE LISTER conventions to show Commodore characters, and HELPOUT for typing accuracy. If you don't need HELPOUT, omit the last 5 characters ('XXXX) on the end of each line.

3x3 E.T. PUZZLE

```
1000 REM: 3X3 E.T. PUZZLE'BOBW
1010 REM: AUTHOR UNKNOWN'BODX
1020 REM: PUBLIC DOMAIN SOFTWARE'BVUB
1030 REM: AMENDED FOR COMMODORE MAGAZINE'BDRE
1040 :'ABHW
1050 POKE53280,12:POKE53281,9:DIMA$(11):C=11:VC=9'FGKH
1060 PRINT "[CLS]":V=53248'CHQC
1070 POKEV+21,255:POKEV+23,255:POKEV+29,255'GXUI
1080 3FORE=200TO207:POKE2040+E-200,E:FORD=64*ETO64*E+62:
     READS'NFVP
1090 POKED,S:NEXT:POKEV+39+E-200,C:NEXT'HPTJ ✓
1100PRINT"[<BLK>,SPACE5,<O><O><O><O><O><O><O><O>
   <O><O><O><O><O><O><O><O><O><O><O><O>]""BAGI
1110 FORD=1TO16:PRINT"[SPACE5,RVS,SPACE18]":NEXT'FGBC
1120 PRINT "[HOME,DOWN10,RIGHT14,<RED>,RVS,SPACE2]"
     'BAMY
1130 POKEV+8,112:POKEV+9,106'ENQC ✓
1140 FORD=105TO64STEP-1'FHJC
1150 POKEV,D:POKEV+1,D'DIGC
1160 POKEV+2,112:POKEV+3,D'ELCE
1170 POKEV+4,160+64-D:POKEV+5,D'GOHI
1180 POKEV+6,D:POKEV+7,106'ELIH
1190 POKEV+10,160+64-D:POKEV+11,106'GSEK
1200 POKEV+12,D:POKEV+13,106+106-D'GRUC
1210 POKEV+14,112:POKEV+15,106+106-D:NEXT'HULE
1220 :'ABHW
1230 PRINT"[HOME]";:FORD=1TO18:READA$'FJBC
1240 PRINT"[RIGHT24]";'BBOC
1250 FORL=1TOLEN(A$):PRINTMID$(A$,L,1);:FORH=1TO100:
     NEXT:NEXT'LYTL
1260 PRINT:NEXT:POKE198,0'DHKE
1270 GETA$:IFA$<>CHR$(13)THEN1270'GNDI
1280 PRINT"[HOME]";'BBFD
1290 FORD=1TO23:PRINTSPC(24)"[SPACE16]";:FORB=1TO30:
     NEXT:NEXT'KQHP
1300 PRINT"[HOME]";:FORD=1TO8:READA$'FICA
1310 PRINT"[RIGHT24]";'BBOA
1320 FORL=1TOLEN(A$):PRINTMID$(A$,L,1);:FORH=1TO100:
     NEXT:NEXT'LYTJ
1330 PRINT:NEXT'CBJA
1340 FORL=1TO2E3:NEXT:FORD=PEEK(V+14)TOPEEK(V+14)
     +48STEP.25: POKEV+14,D:NEXT'QFFQ
1350 PRINT "[DOWN2,RIGHT24,<RED>]PRESS RETURN"
     ;POKE198,0'CGMK
1360 GETA$:IFA$<>CHR$(13)THEN1360'GNDI
1370 PRINT"[HOME]";'BBFD
1380 FORD=1TO23:PRINTSPC(24)"[SPACE16]";:FORB=1TO30:
     NEXT:NEXT'KQHP
1390 FORD=1TO11:READA$(D):NEXT:A=INT(RND(1)*11)+1
     'KVAP
1400 FORD=0TO7:A(D)=(VAL(MID$(A$(A),D+1,1))-1):NEXT
     'JBCH
1410 FORD=1TO9:READB(D):NEXT:POKEV+21,0'HPEF
1420 PRINT "[HOME,DOWN21,BLK,SPACE2]O.K HERE COMES
     THE SCRAMBLED PICTURE""BAQL
1430 POKEV+A(0)*2,64:POKEV+A(0)*2+1,64'HUYI
1440 POKEV+A(1)*2,112:POKEV+A(1)*2+1,64'HVSJ
1450 POKEV+A(2)*2,160:POKEV+A(2)*2+1,64'HVXK
1460 POKEV+A(3)*2,64:POKEV+A(3)*2+1,106'HVAL
1470 POKEV+A(4)*2,112:POKEV+A(4)*2+1,106'HWON
1480 POKEV+A(5)*2,160:POKEV+A(5)*2+1,106'HWTO
1490 POKEV+A(6)*2,64:POKEV+A(6)*2+1,148'HVMQ
1500 POKEV+A(7)*2,112:POKEV+A(7)*2+1,148'HWAH
1510 C=0:FORD=0TO7:C=C+21A(D):POKEV+21,C:
     FORH=1TO200:NEXT:NEXT'ODIN
1520 PRINT "[HOME,DOWN20,WHT]'BAFC
1530 PRINT "[HOME,DOWN4]" SPC(30)"1[SPACE2]2[SPACE2]3
     [DOWN2,LEFT7]4[SPACE2]5[SPACE2]6[DOWN2,LEFT7]7
     [SPACE2]8[SPACE2]9""CDLP
1540 PRINT "[HOME,DOWN19,RIGHT5]MOVE""BALF
1550 GETM$:M=VAL(M$):IFM$=""THEN1550'GPSK
1560 IFM<1THEN1550'DGUH
1570 M1=VAL(LEFT$(STR$(B(VC)),2))'EPDK
1580 M2=VAL(MID$(STR$(B(VC)),3,1))'ERTM
1590 M3=VAL(MID$(STR$(B(VC)),4,1))'ERVN
1600 M4=VAL(RIGHT$(STR$(B(VC)),1))'EPBF
1610 IF M<>M1 AND M<>M2 AND M<>M3 AND M<>M4
     THEN 1550'NQDM
1620 PRINT "[HOME,DOWN19,RIGHT5,RVS,SPACE4,OFF]:
     IFM    <4THENY=64'FGKI
1630 IFM>3ANDM<7THENY=106'GIJH
1640 IFM>6THENY=148'EGOH
1650 IFM=1ORM=4ORM=7THENX=64'IJIL
1660 IFM=2ORM=5ORM=8THENX=112'IKDM
1670 IFM/3=INT(M/3)THENX=160'HKDM
1680 IFM=VC+3THENGOSUB1790'FIFM
1690 IFM+3=VCTHENGOSUB1810'FIDM
1700 IFM=VC+1THENGOSUB1830'FIDE
1710 IFM+1=VCTHENGOSUB1850'FIAG
1720 A(VC-1)=A(M-1):A(M-1)=99:C=0'GWKK
1730 FORD=1TO8:IFA(D)=DTHENC=C+1'IMGK
1740 NEXT:IFC=7THEN2440'EHCH
1750 VC=M:M5=M5+1'DJXJ
1760 PRINT "[HOME,RIGHT18,DOWN14,RIGHT7,RVS,BLK]
     MOVES[OFF,WHT]";M5'BDUN
1770 GOTO1540'BEHI
1780 REM: UP'BDJJ
1790 FORL=YTOY-42STEP-1:POKEV+A(M-1)*2+1,L:NEXT:
     RETURN'NTVV
1800 REM: DOWN'BFBD
1810 FORL=YTOY+42:POKEV+A(M-1)*2+1,L:NEXT:RETURN
     'LSMN
1820 REM: LEFT'BFSE
1830 FORL=XTOX-48STEP-1:POKEV+A(M-1)*2,L:NEXT:
     RETURN'MSMP
1840 REM: RIGHT'BGUH
```

CONTINUED OVERLEAF

CONTINUED FROM PREVIOUS PAGE

```
1850 FORL=XTOX+48:POKEV+A(M-1)*2,L:NEXT:RETURN
     'KRDQ
1860 :'ABHH
1870 REM: DATA FOR 8 SPRITES'BQSN
1880 :'ABHJ
1890 DATA  0, 0, 0, 0, 0, 31, 0, 1,224, 0, 6, 0'BBXP
1900 DATA  0, 24, 15, 0, 96,127, 0, 65,255, 0,135,255'BKFJ
1910 DATA  1, 31,255, 2, 63,252, 4, 63,252, 4,127,252'BLEK
1920 DATA  4,255,128, 8,254, 15, 8,254,127, 16,254,187'BOAM
1930 DATA 19,252,159, 35,248,143, 39,248,112, 47,240, 15'BRVN
1940 DATA 35,224, 0, 0, 0, 0,224, 0, 0, 30, 0,127'BGAM
1950 DATA  1,255,128, 92, 0, 16,252,198, 17,255,241, 10'BOIP
1960 DATA 255,248,146,255,252,212, 63,254, 32, 31,255,143'BTHR
1970 DATA 31,255,255,231,255,255,211,255,255,208, 16, 68'BTMS
1980 DATA 216,122, 68,144,154,137, 17,113, 2,226,134, 18'BRES
1990 DATA 21, 27, 41, 42, 36,196, 0, 0, 0, 0, 0, 0'BFNR
2000 DATA 128, 0, 0,103, 0, 0, 24,240, 0, 0, 28, 0'BGAA
2010 DATA  0, 3, 0, 31,128,192,255,248, 32,255,252, 32'BNFC
2020 DATA 255,252, 16,255,254, 16,192,126, 8,128, 63, 8'BPXE
2030 DATA 63, 31, 4,220,158, 4, 46, 94, 4,102,126, 8'BKKE
2040 DATA 62, 78, 8,156,192, 8,127, 0,136, 32,230, 0'BLPF
2050 DATA 35, 1,192, 34, 0, 63, 33,128, 0, 16,120, 28'BKXG
2060 DATA 16, 7,255, 10, 0, 1, 9,147, 14, 4,100, 8'BHDG
2070 DATA  2, 24, 16, 1, 32, 16, 0,192, 32, 0, 54,192'BIAH
2080 DATA  0, 9, 0, 0, 4, 3, 0, 2,132, 0, 1,104'BCHH
2090 DATA  0, 0,145, 0, 0, 65, 0, 0,126, 0, 0, 72'BEMI
2100 DATA  4, 68, 2,132,137,154, 73, 40,180,137, 36, 9'BMTC
2110 DATA 145, 65, 0, 18, 65, 32, 34,130, 32, 36,132, 24'BMDD
2120 DATA 73, 0, 4, 74, 0, 0, 74, 3,142,144, 7,255'BHKD
2130 DATA 145, 30,254,144,248, 62, 16, 16, 62, 75, 11,158'BPNG
2140 DATA 160,134,124, 32, 64, 24, 16, 56, 0, 8, 7, 0'BJDG
2150 DATA  2, 0, 14, 0, 0,136, 0, 65,136, 14, 50, 48'BHVG
2160 DATA  1,140, 80,192, 64, 96, 48, 48,160, 8, 9, 64'BKII
2170 DATA  4, 6, 64, 2, 0,128, 67, 33, 0, 34,198, 0'BGPI
2180 DATA 18, 24, 0, 20, 96, 0, 36,128, 0, 43, 0, 0'BGWJ
2190 DATA 68, 0, 0, 4, 0, 0, 24, 0, 0, 32, 0, 0'BBIJ
2200 DATA 64, 0, 0,128, 0, 0, 0, 0, 38, 0, 0, 33'BDNB
2210 DATA  0, 0, 32, 0, 0,144, 0, 3,156, 0, 31, 18'BFHD
2220 DATA  3,190, 9, 31,166, 8, 30, 3, 4, 24, 3, 4'BFKE
2230 DATA 24, 3, 7, 27,131, 4, 31,129,132, 14, 1,162'BKUG
2240 DATA 12, 97,130, 12,232, 3, 15,192, 2, 15, 0, 1'BJUG
2250 DATA  4, 0, 1, 0, 0, 1, 0, 0, 1, 1,128, 17'BBGG
2260 DATA  0,120,225,248, 7, 2, 6, 0, 12, 1,255,240'BJUI
2270 DATA  0, 64, 0,128, 32, 0, 65, 32, 0, 62, 32, 0'BGSJ
2280 DATA  0, 64, 0, 0, 64, 0,192, 64, 0, 35, 64, 0'BFAK
2290 DATA 28, 64, 0, 0, 32, 0,128, 32, 0,112, 32, 0'BHLL
2300 DATA  8,160, 0, 7, 32, 0, 0, 16, 0, 0, 16, 0'BDQC
2310 :'ABHX
2320 DATA "[YEL]YOUR OBJECTIVE",IS TO RECREATE,THIS
     PICTURE,"OF E.T."'BBYN
2330 DATA ,"[BLK]HE IS MADE UP",OF A GRID OF,3X3 SPRITES,
     YOU CAN MOVE'BIUN
2340 DATA EACH CELL INTO,THE VACANT ONE,,"[GRN]YOU DO
     THIS BY","TYPING THE NO."'BDUQ
2350 DATA OF THE CELL YOU,WANT TO MOVE'BXKJ
2360 DATA ,"[<RED>]NOW PRESS RETURN"'BBTI
2370 DATA "THE GRID IS SET","UP LIKE THIS:[*]"'BBSN
2380 DATA "[WHT,SPACE2]1[SPACE2]2[SPACE2]3","[SPACE2]4
     [SPACE2]5[SPACE2]6","[SPACE2]7[SPACE2]8[SPACE29"'BCGK
2390 DATA ,"[BLK]SO TYPING 8","WILL DO THIS"'BCHN
2400 DATA 82167534,17654238,85476213,32856471'BKAF
2410 DATA 38764152,41257360,64132875,61873425BKRG
2420 DATA 57638241,84657312,28654137'BBXF
2430 DATA 0024,0135,0260,0157,2468,0359,0480,0579,0680'BTBK
2440 POKE54296,15:A1=54291:R1=54292:A2=54277:R2=54278'FPLN
2450 H1=54287:L1=54286:W1=54290:H2=54273:L2=54272:
     W2=54276'GWQR
2460 POKEA1,102:POKER1,0:POKEA2,96:POKER2,0'EWIL
2470 READH,L,D:IFH=0THEN2690'EMEK
2480 POKEW1,0:POKEW2,0'CJXJ
2490 POKEW1,17:POKEW2,33'CLIK
2500 POKEH1,0:POKEH1,H:POKEH2,0:POKEH2,H'ETSF
2510 POKEL1,0:POKEL1,L:POKEL2,0:POKEL2,L:FORE=0TO7*D:
     POKER2,136'JGLM
2520 POKER1,136:NEXT:GOTO2470'DMSF
2530 DATA 34, 75, 6, 51, 97, 6, 45,198, 1, 43, 52, 'BHHI
2540 DATA 38,126, 1, 43, 52, 1, 34, 75, 6, 25,177, 10'BJLK
2550 DATA 28,214, 6, 57,172, 6, 51, 97, 1, 48,127, 1'BJYL
2560 DATA 43, 52, 1, 48,127, 1, 38,126, 6, 64,188, 10'BKQM
2570 DATA 38,126, 6, 68,149, 6, 64,188, 1, 57,172, 1'BKKN
2580 DATA 51, 97, 1, 45,198, 1, 40,200, 6, 34, 75, 10'BJKO
2590 DATA 40,200, 6, 34, 75, 6, 68,149, 4, 76,252, 2'BJUP
2600 DATA 64,188, 12, 68,150, 6,102,196, 6, 90, 99, 1'BLGH
2610 DATA 86, 52, 1, 76,126, 1, 86, 52, 1, 68, 75, 6'BHKH
2620 DATA 50,177, 10, 56,214, 6,114,172, 6,102, 97, 1'BMHJ
2630 DATA 96,127, 1, 86, 52, 1, 96,127, 1, 76,126, 6'BJDK
2640 DATA 128,188, 10, 76,126, 6,136,149, 6,128,188, 1'BOBM
2650 DATA 114,172, 1,102, 97, 1, 90,198, 1, 80,200, 6'BLDM
2660 DATA 68, 75, 10, 80,200, 6, 68, 75, 6,136,149, 4'BKCN
2670 DATA 152,252, 2,128,188, 12,136,149, 6,204,194, 6'BPOP
2680 DATA 192,254, 6,152,252, 6,136,149, 12, 0, 0, 0'BLBP
2690 POKER1,14:POKE54296,0:POKEH1,0:POKEL1,0:POKEH2,0:
     POKEL2,0:END'HJLU
```

Presentation (C) Paul Blair 1985

SAVE "HOWIDAY",1

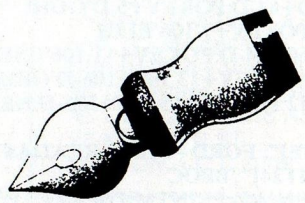
LETTERS - CONTINUED FROM PAGE 16

stop this?

The only answer I have found is to use GET. Program 2 attached is an example of this with pretty complete protection against errors. It maybe of use to your readers.

Yours Sincerely
David Belson
Buderim, Qld.

ED – Can anyone help David with this one?



Listing 1

```
10 PRINT "[CLR,DOWN]"
20 INPUT "A NUMBER";A$:D=0
30 FOR X=1 TO LEN (A$)
40 B$=MID$(A$,X,1)
50 IF X=1 AND B$="0" THEN PRINT "[UP2]": GOTO 20
60 IF X=1 AND (B$="-" OR B$="+") THEN 100
70 IF D>0 AND B$="." THEN PRINT "[UP2]": GOTO 20
80 IF B$="." THEN D=1: GOTO 100
90 IF B$<"0" OR B$>"9" THEN PRINT "[UP2]": GOTO 20
100 NEXT X
110 PRINT "[DOWN]"A$
```

Listing 2

```
10 PRINT "[CLR,DOWN]"
20 C=0:D=0:A$="":B$=""
30 FOR X=1 TO 39: PRINT " ";: NEXT X:X=0: PRINT : PRINT "[UP2]"
40 C$="ENTER A NUMBER ' ": PRINT C$
50 GET A$: IF A$="" THEN PRINT "[RVS]<[OFF,LEFT]";: GOTO 50
60 PRINT A$;
70 C=C+1: IF C=1 AND A$=CHR$(13) THEN PRINT "[UP2]":C=0:
   GOTO 40
80 IF A$=CHR$(13) THEN PRINT "[UP]" SPC(LEN (C$)+ LEN
   (B$))" ": GOTO 130
90 B$=B$+A$
100 A$="": GOTO 50
130 FOR X=1 TO LEN (B$):B1$=MID$(B$,X,1)
140 IF X=1 AND B1$="0" THEN PRINT "[UP2]": GOTO 20
150 IF X=1 AND B1$="-" THEN 190
160 IF D>0 AND B1$="." THEN PRINT "[UP2]": GOTO 20
170 IF B1$="." THEN D=1: GOTO 190
180 IF B1$<"0" OR B1$>"9" THEN PRINT "[UP2]": GOTO 20
190 NEXT X
200 PRINT "[DOWN]"B$
```

THE VIC MAGICIANS APPRENTICE

Michael Spiteri

THE VIC MAGICIAN'S APPRENTICES HALF YEAR EXAM

Our apprentice has to face his half-year exams. The copy below has been rewritten since the originals were stolen from the 'Government Printers'. For his work books our apprentice has used the Commodore Magazine and The VIC Users Guide that comes with your machine.

Incentive

You can complete the exam just for fun or Superior Software have offered to give a copy of their VIC FAMILY PACK (15 programs) to the first correct entry **they** receive. Send to:

SUPERIOR SOFTWARE
20 Larool Cres.,
Seaford
Vic 3198

NOTE: Send direct to Superior Software not the magazine.

INSTRUCTIONS

Each examinee will refrain from using spells and other mysteries and non computing skills he/she has acquired. Answer the questions in the space provided and in accordance with any special instructions that may be given within this paper.

The examiner will frown upon scribbled comments such as "Kiss me I'm a frog" and the use of invisible ink is forbidden.

Signed
Vic Amiga Wiz. (Syd)
Frog Pond
Princesses may apply

NAME: .....

ADDRESS: .....

.....

.....

PHONE: .....

Question 1 - Multiple Choice

(Circle the correct solution)

a) What is the micro processor used by the Vic?

i) 6502
ii) Z80
iii) 68000

b) What BASIC command followed by a number executes a machine-language program?

i) RUN
ii) SYS
iii) GOTO
iv) USR

c) Which one of the following commands is NOT Vic BASIC?

i) PEEK
ii) CMD
iii) TO
iv) DRAW

d) Which of the following will not execute a program?

i) RUN
ii) GOTO
iii) GOSUB
iv) None of the last three.

e) Which Vic-20 command clears the screen?

i) CLS
ii) CLR
iii) CLEAR
iv) None of the above.

Question 2 - Written answers

a) How much RAM does the standard Vic contain which is useable?

b) Write a ONE-LINE program that PRINTs "HELLO" down the screen non-stop.

c) What combination of keys makes the Vic auto-load-run a program?

d) How many screen locations are there?

e) What BASIC command changes the screen colour to BLUE and the border to RED?

f) What BASIC command sets the volume to 14?

g) How many pins does the memory expansion cartridge slot have?

Question 3 - Practical

On a separate sheet of paper, write programs that:

a) Will create a siren sound.
b) Will fill the screen the random characters.
c) Work out the multiplication of 5 numbers.

Sound Effects for the Commodore 64

WHEEE

by David Bergmeier

I mentioned in the last article I would discuss randomness in sound effects. Well here goes. Type in the six initialization lines :-

```
10 S=54272
20 FORI=0TO24:POKES+I,0:NEXT
30 POKES+24,15
40 POKES+5,0
50 POKES+6,240
60 POKES+4,17
```

Now for the randomness. For a real random sound effect add the following :-

```
70 FORI=1TO100
80 POKES+1,INT(RND(1)*256)
90 NEXT
100 POKES+1,0
```

All this sound effect does is POKE a random number into the pitch. (Great for making computers sound as though they can think.) To add a bit more regularity to random sounds it is best to return to the FOR-NEXT loop. Enter the initialization (lines 10-60), then add the following :-

```
70 FORI=1TO10
80 R=INT(RND(1)*90)+10
90 FORJ=1TOR
100 POKES+1,J
110 NEXT
120 NEXT
130 POKES+1,0
```

How do you work out the value of 'R' (line 80) ? I'm glad you asked. I used the following formula for any two (positive or negative) integers provided the first one is greater than the second (ie. A>B). The formula is :-

```
80 R=INT(RND(1)*(B-A))+A
```

So line 80 generates a random number between 10 and 100. The FOR-NEXT loop simply counts from 10 to the random number and changes the pitch. Every time you run it , this sound effect will be different from the last. This is easily modified. Change line 90 to count from 100 to R STEP-1 . You could also calculate two random numbers (R1 and R2) . Then (provided the ranges are correct) alter line 90 to count from R1 to R2 with the correct STEP (according to whether R1 is greater than R2 or not). Make sure when selecting ranges , that R1 is ALWAYS greater than R2 (STEP1) or R1 is ALWAYS less than R2 (STEP-1).

Up till now we have been using the ADSR but not understanding how it works. Firstly the SUSTAIN. This is a volume (like S+24) that ranges from zero to fifteen. The volume is what the note will sound at until at is turned off. It comes into effect only when the attack and decay have been completed. The ATTACK is the rate at which the tone rises from zero volume to maximum volume. The DECAY is the time for the volume to fade from maximum volume to the SUSTAIN volume. (If the SUSTAIN volume is the maximum 15, then the DECAY is not heard.) The RELEASE is the time for the SUSTAIN volume to fade away to zero once the SUSTAIN has been stopped. The ATTACK DECAY and RELEASE rates range from super fast to super slow , according to the value POKEd into the ADSR memory locations (S+5 and S+6). The values are as follows :-

VALUE	ATTACK RATE	DECAY & RELEASE RATE
0	0.002 secs	0.006 secs
2	0.008 secs	0.024 secs
2	0.016 secs	0.048 secs
3	0.024 secs	0.072 secs
4	0.038 secs	0.114 secs
5	0.056 secs	0.168 secs
6	0.068 secs	0.204 secs
7	0.080 secs	0.240 secs
8	0.1 secs	0.3 secs
9	0.25 secs	0.75 secs
10	0.5 secs	1.5 secs
11	0.8 secs	2.4 secs
12	1 sec	3 secs
13	3 secs	9 secs
14	5 secs	15 secs
15	8 secs	24 secs

To control the ADSR , BIT 0 of the waveform must be understood and used. This particular BIT is a controller for the ADSR. It determines when the ATTACK will start and when the SUSTAIN will finish. Have a quick glance at the six initialization lines. The waveform (POKE S+4) is always 17,33,65 or 129. However , to select the waveform you only have to select 16,32,64 or 128 (BITs 4,5,6 and 7 of S+4). The sound will not work with only these settings though. You MUST firstly set BIT 0 in S+4 to a one. (Hence the POKE of S+4 with 16+1,32+1, etc.) This enables the sound to be heard instantly the pitch is specified. When BIT 0 in S+4 is changed to a zero, the RELEASE part of the sound begins. And since there was no release in the ADSR settings, the sound went super-fast to zero. (It sounds almost instantaneous.) To use the ADSR change the initialization lines to read the following :-

```
10 S=54272:AT=12:DK=14:SU=5:RE=15
20 FORI=0TO24:POKES+I,0:NEXT
30 POKES+24,15
40 POKES+5,AT*16+DK
50 POKES+6,SU*16+RE
60 POKES+4,16
```

To hear exactly what the ADSR sounds like , enter the next few lines :-

```
70 POKES+1,20
80 PRINT" PRESS A KEY TO START ADSR "
90 GETA$:IFA$=""THEN90
100 POKES+4,17
110 PRINT" PRESS A KEY TO END ADSR "
120 GETA$:IFA$=""THEN120
130 POKES+4,16
140 PRINT" PRESS A KEY TO END "
150 GETA$:IFA$=""THEN150
160 POKES+1,0
```

Don't press a key to end the ADSR until you have fully heard the ATTACK , DECAY and SUSTAIN. Only when you think the volume of the note is constant can you press a key to end the ADSR. Then when the sound has fully died away, press a key to end the sound effect. This is only a simple program but it fully demonstrates the ADSR. Try changing the values of the ADSR in line 10 but remember, some of the times (see table) are so short that there is practically no difference between one value and another. (eg. values 0 to 5)

Now comes the adding of ADSR to other sound effects , and this can be quite tricky. Firstly let's add an ADSR to the first sound effect in this article. Enter the six initialization lines (with the ADSR POKEs) and then add the following lines :-

```
70 POKES+4,17
80 FORI=1TO500
90 POKES+1,INT(RND(1)*255)+1
100 NEXT
110 POKES+4,16
120 POKES+1,0
```

Then change the ADSR in line 10 to :-

```
10 S=54272:AT=0:DK=13:SU=5:RE=15
```

Now we have a totally random sound effect that decays away slowly. The changes by now should be fairly obvious so go ahead and create some really interesting sound effects.

Now we know everything about ADSR, what about an explosion ? Well, type in the first six initialization lines, using a noise waveform (POKES+4,128).

Now add the following:-

```
70 POKES+1,10
80 POKES+4,129
90 FORI=1TO1000:NEXT
100 POKES+4,128
```

The ADSR should look something like any of the following :-

```
10 S=54272:AT=0:DK=0:SU=15:RE=15
10 S=54272:AT=0:DK=0:SU=15:RE=15
10 S=54272:AT=12:DK=0:SU=15:RE=15
10 S=54272:AT=10:DK=9:SU=10:RE=13
```

Of course there are many other possible ADSR settings. Try a few and see what happens.

By now you may well have realized that so far we have only been using one voice. Didn't it mention somewhere on the box there were three ? Well the box was correct. There are three voices each of them able to sound independently. The locations of all the voices are as follows :-

LOCATION	DESCRIPTION
S+0	Low byte of pitch (voice 1)
S+1	Hi byte of pitch (voice 1)
S+2	Low byte of pulse width (voice 1)
S+3	Hi byte of pulse width (voice 1)
S+4	Waveform (voice 1)
S+5	Attack*16 + decay (voice 1)
S+6	Sustain*16 + release (voice 1)
S+7	Low byte of pitch (voice 2)
S+8	Hi byte of pitch (voice 2)
S+9	Low byte of pulse width (voice 2)
S+10	Hi byte of pulse width (voice 2)
S+11	Waveform (voice 2)
S+12	Attack*16 + decay (voice 2)
S+13	Sustain*16 + release (voice 2)
S+14	Low byte of pitch (voice 3)
S+15	Hi byte of pitch (voice 3)
S+16	Low byte of pulse width (voice 3)
S+17	Hi byte of pulse width (voice 3)
S+18	Waveform (voice 3)
S+19	Attack*16 + decay (voice 3)
S+20	Sustain*16 + release (voice 3)

To use any voice you must first initialize it. This will be the same six lines as previously used except that instead of doing it only for voice 1 , it will have to be done for each voice that is to be used. Examples of sound effects using multiple voices can be found in the attached program which also contains some advanced one voice sound effects. I won't bother to explain these , the changes that can be made should be quite obvious.

Good luck and with a bit of rearranging of my program , you should come up with some astounding noises. I wouldn't turn up the siren too much (I think there's a law against loud Police sirens not coming from Police cars.)

The next article delves into the murky depths of ring modulation , filters and synchronization.

SFX GENERATOR 3

```
10 REM "################################"
   'BAPC
20 REM "#[SPACE2]SOUND EFFECTS GENERATOR V-3 #""BAXG
30 REM "# BY DAVID BERGMEIER JUNE 1985 #""BABH
40 REM "################################"
   'BAPF
100 PRINT "[CLR,WHT]";: POKE 53280,6: POKE 53281,0'DRJA
110 PRINT " < SOUND EFFECTS GENERATOR VERSION 3 ""BAEG
120 PRINT "[DOWN,RED,RIGHT3]PLEASE MAKE YOUR
S    ELECTION[GRN,DOWN]""BASF
130 PRINT "[RVS,SPACE12]1 RANDOM SOUNDS #1[SPACE10]"
   'BABF
140 PRINT "[RVS,SPACE12]2 RANDOM SOUNDS #2[SPACE10]"
   'BADG
150 PRINT "[RVS,SPACE12]3 RANDOM SOUNDS #3[SPACE10]"
   'BAAI
160 PRINT "[RVS,SPACE12]4 RANDOM SOUNDS #4[SPACE10]"
   'BACJ
170 PRINT "[RVS,SPACE12]5 ADSR TEST[SPACE17]""BAAJ
180 PRINT "[RVS,SPACE12]6 EXPLOSION #1[SPACE14]""BAEK
190 PRINT "[RVS,SPACE12]7 EXPLOSION #2[SPACE14]""BAGL
200 PRINT "[RVS,SPACE12]8 RANDOM WITH ADSR[SPACE10]"
   'BAGE
210 PRINT "[RVS,SPACE12]9 ADSR GONG[SPACE17]""BAIE
220 PRINT "[RVS,SPACE12]0 - NEXT PAGE -SPACE13,HOME]"
   'BADF
230 GET A$: IF A$="" THEN 230'EIED
240 IF A$="0" THEN 280'DFUD
250 IF VAL (A$)<1 THEN 230'EIMF
260 ON VAL (A$) GOSUB 1000,2000,3000,4000,5000,6000,7000,8000,
   9000'DXEM
270 GOTO 100'BDAE
280 PRINT "[CLR,WHT]";: POKE 53280,6: POKE 53281,0'DRJJ
290 PRINT " < SOUND EFFECTS GENERATOR VERSION 3 ""BAEP
300 PRINT "[DOWN,RED,RIGHT3]PLEASE MAKE YOUR
   SELECTION[CYN,DOWN]""BASG
310 PRINT "[RVS,SPACE12]1 HI-LO SIREN[SPACE15]""BAVF
320 PRINT "[RVS,SPACE12]2 WARNING[SPACE19]""BAAF
330 PRINT "[RVS,SPACE12]3 STEAM WHISTLE[SPACE13]""BADH
340 PRINT "[RVS,SPACE12]4 SIREN #1 (2 VOICES)[SPACE5]""BAPI
350 PRINT "[RVS,SPACE12]5 SIREN #2 (3 VOICES)[SPACE5]""BASJ
360 PRINT "[RVS,SPACE12]6 SIREN #3 (2 VOICES)[SPACE5]""BATK
370 PRINT "[RVS,SPACE12]7 FUNNY SIREN ( 2 VOICES )[SPACE2]"
   'BAKM
380 PRINT "[RVS,SPACE12]8 WEIRD SIREN ( 2 VOICES )[SPACE2]"
   'BAPN
390 PRINT "[RVS,SPACE12]9 HOSPITAL ( 2 VOICES )[SPACE5]"
   'BABO
400 PRINT "[RVS,SPACE12]0 - NEXT PAGE -SPACE13]""BAJF
410 GET A$: IF A$="" THEN 410'EIED
420 IF A$="0" THEN 100'DFLD
430 IF VAL (A$)<1 THEN 410'EIMF
440 ON VAL (A$) GOSUB 10000,11000,12000,13000,14000,15000,
   16000,17000,18000'DHYO450 GOTO 100'BDAE
1000 REM "### RANDOM SOUNDS ONE ###""BAQY
1010 S=54272'BGJV
1020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDA
1030 POKE S+24,15'C GO X
```

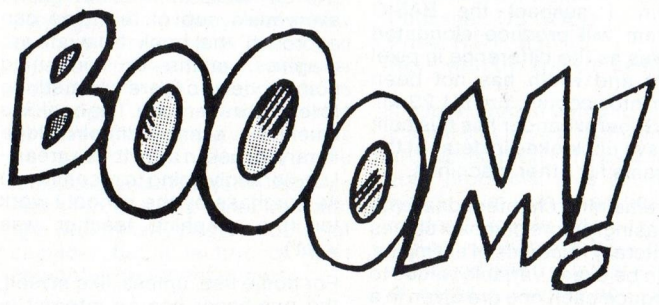
CONTINUED OVERLEAF

ZZZAP

```
1040 POKE S+5,0'CELY
1050 POKE S+6,240'C GO A
1060 POKE S+4,17'CFQB
1070 FOR I=1 TO 200'DFBC
1080 POKE S+1, INT ( RND (1)*256)'FLYG
1090 NEXT 'BAEC
1100 POKE S+1,0'CEHV
1110 RETURN 'BAQU
2000 REM "### RANDOM SOUNDS TWO ###"'BAPA
2010 S=54272'BGJW
2020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDB
2030 POKE S+24,15'CGOY
2040 POKE S+5,0'CELA
2050 POKE S+6,240'CGOB
2060 POKE S+4,17'CFQC
2070 FOR I=1 TO 20'DEDD
2080 R=INT ( RND (1)*90)+10'FKCH
2090 FOR J=1 TO R'DDNF
2100 POKE S+1,J'CEIW
2110 NEXT 'BAEV
2120 NEXT 'BAEW
2130 POKE S+1,0'CEHA
2140 RETURN 'BAQY
3000 REM "### RANDOM SOUNDS THREE ###"'BANC
3010 S=54272'BGJX
3020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDC
3030 POKE S+24,15'CGOA
3040 POKE S+5,0'CELB
3050 POKE S+6,240'CGOC
3060 POKE S+4,17'CFQD
3070 FOR I=1 TO 10'DECE
3080 R1=INT ( RND (1)*90)+1'FKDI
3090 R2=INT ( RND (1)*155)+100'FNAJ
3100 FOR J=R1 TO R2'DFOY
3110 POKE S+1,J'CEIY
3120 NEXT 'BAEX
3130 NEXT 'BAEY
3140 POKE S+1,0'CEHC
3150 RETURN 'BAQB
4000 REM "### RANDOM SOUNDS FOUR ###"'BAHC
4010 S=54272'BGJY
4020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDD
4030 POKE S+24,15'CGOB
4040 POKE S+5,0'CELC
4050 POKE S+6,240'CGOD
4060 POKE S+4,17'CFQE
4070 FOR I=1 TO 20'DEDF
4080 R=INT ( RND (1)*90)+10'FKCJ
4090 FOR J=100 TO R STEP -1'FGLJ
4100 POKE S+1,J'CEIY
4110 NEXT 'BAEX
4120 NEXT 'BAEY
4130 POKE S+1,0'CEHC
4140 RETURN 'BAQB
5000 REM "### ADSR TEST ###"'BAGB
5010 S=54272:AT=13:DK=14:SU=3:RE=15'FAAH
5020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDE
5030 POKE S+24,15'CGOC
5040 POKE S+5,AT*16+DK'EJNF
5050 POKE S+6,SU*16+RE'EJQG
5060 POKE S+4,16'CFPF
5070 PRINT "[CLR,WHT]START ATTACK"'BAMI
5080 POKE S+1,20'CFHH
5090 POKE S+4,17'CFQI
5100 FOR I=1 TO 3000: NEXT 'EHHB
5110 PRINT "[DOWN,CYN]START DECAY"'BADD
5120 FOR I=1 TO 5000: NEXT 'EHJD
5130 PRINT "[DOWN,PUR]HOLD SUSTAIN"'BAQF
5140 FOR I=1 TO 5000: NEXT 'EHJF
5150 PRINT "[DOWN,GRN]START RELEASE"'BAFH
5160 POKE S+4,16'CFPG
5170 FOR I=1 TO 14000: NEXT 'EICJ
5180 PRINT "[DOWN,<RED>]END ADSR"'BAEJ
5190 POKE S+1,0'CEHJ
5200 FOR I=1 TO 1000: NEXT 'EHFC
5210 RETURN 'BAQA
6000 REM "### EXPLOSION #1 ###"'BAFC
6010 S=54272:AT=0:DK=0:SU=15:RE=15'FYVI
6020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDF
6030 POKE S+24,15'CGOD
6040 POKE S+5,AT*16+DK'EJNG
6050 POKE S+6,SU*16+RE'EJQH
6060 POKE S+4,128'CGRG
6070 POKE S+1,10'CFGH
6080 POKE S+4,129'CGSI
6090 FOR I=1 TO 1000: NEXT 'EHFK
6100 POKE S+4,128'CGRB
6110 RETURN 'BAQA
6120 REM 'BARB
6130 REM "- IN THIS EFFECT THE PITCH IS"'BAEK
6140 REM "[SPACE2]NOT TURNED OFF SO THE SOUND"'BAVL
6150 REM "[SPACE2]CAN STILL BE HEARD FADING"'BAGL
6160 REM "[SPACE2]AWAY EVEN THOUGH THE CODE"'BARM
6170 REM "[SPACE2]FOR THE EFFECT HAS BEEN"'BAEM
6180 REM "[SPACE2]COMPLETED."'BANK
7000 REM "### EXPLOSION #1 ###"'BAFD
7010 S=54272:AT=15:DK=15:SU=0:RE=0'FYVJ
7020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDG
7030 POKE S+24,15'CGOE
7040 POKE S+5,AT*16+DK'EJNH
7050 POKE S+6,SU*16+RE'EJQI
7060 POKE S+4,128'CGRH
7070 POKE S+1,10'CFGI
7080 POKE S+4,129'CGSJ
7090 FOR I=1 TO 25000: NEXT 'EIEM
7100 POKE S+4,128
7110 RETURN 'BAQB
8000 REM "### RANDOM SOUND WITH ADSR ###"'BABH
8010 S=54272:AT=0:DK=13:SU=0:RE=0'FXQK
8020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDH
8030 POKE S+24,15'CGOF
8040 POKE S+5,AT*16+DK'EJNI
8050 POKE S+6,SU*16+RE'EJQJ
8060 POKE S+4,17'CFQI
8070 FOR I=1 TO 750'DFLJ
8080 POKE S+1, INT ( RND (1)*256)'FLYN
8090 NEXT 'BAEJ
8100 POKE S+1,0'CEHD
8110 RETURN 'BAQC
9000 REM "### RANDOM SOUND WITH ADSR ###"'BABI
9010 S=54272:AT=0:DK=10:SU=0:RE=0'FXNL
9020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDI
9030 POKE S+24,15'CGOG
9040 POKE S+5,AT*16+DK'EJNJ
9050 POKE S+6,SU*16+RE'EJQK
9060 POKE S+4,16'CFPJ
9070 FOR I=100 TO 10 STEP -15'FIDM
9080 POKE S+1,I'CEHL
9090 POKE S+4,16'CFPM
9100 POKE S+4,17'CFQE
9110 FOR J=1 TO 1000: NEXT 'EHGG
9120 NEXT 'BAEE
9130 POKE S+1,0'CEHH
9140 RETURN 'BAQG
10000 REM "### HI-LO SIREN ###"'BAUU
10010 S=54272:AT=0:DK=0:SU=15:RE=0'FXSB
10020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDX
10030 POKE S+24,15'CGOV
10040 POKE S+5,AT*16+DK'EJNY
10050 POKE S+6,SU*16+RE'EJQA
10060 POKE S+4,17'CFQY
10070 FOR I=1 TO 5'DDIA
10080 POKE S+1,20'CFHB
10090 FOR J=1 TO 500: NEXT 'EGMD
10100 POKE S+1,10'CFGT
10110 FOR J=1 TO 500: NEXT 'EGMV
10120 NEXT 'BAET
10130 POKE S+1,0'CEHW
10140 RETURN 'BAQV
11000 REM "### ALIEN WARNING ###"'BAGW
11010 S=54272:AT=0:DK=0:SU=15:RE=0'FXSC
11020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDY
11030 POKE S+24,15'CGOW
```

```
11040 POKE S+5,AT*16+DK'EJNA
11050 POKE S+6,SU*16+RE'EJQB
11060 POKE S+4,17'CFQA
11070 FOR I=1 TO 30'DEEB
11080 FOR J=60 TO 100 STEP 4'EHTD
11090 POKE S+1,J'CEID
11100 NEXT 'BAES
11110 FOR J=100 TO 60 STEP -4'FHJX
11120 POKE S+1,J'CEIW
11130 NEXT 'BAEV
11140 NEXT 'BAEW
11150 POKE S+1,0'CEHA
11160 RETURN 'BAQY
12000 REM "### STEAM WHISTLE ###"'BAIX
12010 S=54272:AT=0:DK=0:SU=15:RE=11'FYRD
12020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDA
12030 POKE S+24,15'CGOX
12040 POKE S+5,AT*16+DK: POKE S+12,AT*16+DK'IUUB
12050 POKE S+6,SU*16+RE: POKE S+13,SU*16+RE'IUAC
12060 POKE S+4,33: POKE S+11,17'EMEE
12070 POKE S+1,34: POKE S+8,43'ELPE
12080 POKE S+4,32: POKE S+11,16'EMCA
12090 FOR I=1 TO 1000: NEXT 'EHFF
12100 POKE S+4,33: POKE S+11,17'EMEY
12110 FOR I=1 TO 500: NEXT 'EGLX
12120 POKE S+4,32: POKE S+11,16'EMCB
12130 FOR I=1 TO 1000: NEXT 'EHFA
12140 POKE S+1,0: POKE S+8,0'EJFC
12150 RETURN 'BAQY
13000 REM "### SIREN ONE ###"'BAYX
13010 S=54272:AT=0:DK=0:SU=15:RE=0'FXSE
13020 FOR I=0 TO 24: POKE S+I,0: NEXT 'GKDB
13030 POKE S+24,15'CGOY
13040 POKE S+5,AT*16+DK: POKE S+12,AT*16+DK'IUUC
13050 POKE S+6,SU*16+RE: POKE S+13,SU*16+RE'IUAD
13060 POKE S+4,17: POKE S+11,17'EMGF
13070 FOR I=1 TO 3'DDGD
13080 FOR J=10 TO 160'DGZF
13090 POKE S+1,J'CEIF
13100 POKE S+8,J+3'DFFX
13110 NEXT 'BAEV
13120 FOR J=160 TO 10 STEP -1'FHHB
13130 POKE S+1,J'CEIA
13140 POKE S+8,J+3'DFFC
13150 NEXT 'BAEA
13160 NEXT 'BAEB
13170 POKE S+1,0: POKE S+8,0'EJFA
13180 RETURN 'BAQD
14000 REM "### SIREN TWO ###"'BAXY
14010 S1=54272:S2=54279:S3=54286:AT=0:DK=0:SU=15:RE=0
      'HPHE
14020 FOR I=0 TO 24: POKE S1+I,0: NEXT 'GLWD
14030 POKE S1+24,15'CHNA
14040 POKE S1+5,AT*16+DK: POKE S2+5,AT*16+DK:
      POKE S3+5,AT*16+DK'MHOI
14050 POKE S1+6,SU*16+RE: POKE S2+6,SU*16+RE:
      POKE S3+6,SU*16+RE'MHXJ
14060 POKE S1+4,33: POKE S2+4,17: POKE S3+4,17'GUCD
14070 FOR I=1 TO 3'DDGE
14080 FOR J=10 TO 160'DGZA
14090 POKE S1+1,J: POKE S2+1,J+3: POKE S3+1,J-3'ITRH
14100 NEXT 'BAEV
14110 FOR J=160 TO 10 STEP -1'FHHB
14120 POKE S1+1,J: POKE S2+1,J+3: POKE S3+1,J-3'ITRB
14130 NEXT 'BAEY
14140 NEXT 'BAEA
14150 POKE S1+1,0: POKE S2+1,0: POKE S3+1,0'GRIC
14160 RETURN 'BAQC
15000 REM "### SIREN THREE ###"'BABA
15010 S1=54272:S2=54279:AT=0:DK=0:SU=15:RE=0'GHPC
15020 FOR I=0 TO 24: POKE S1+I,0: NEXT 'GLWE
15030 POKE S1+24,15'CHNB
15040 POKE S1+5,AT*16+DK: POKE S2+5,AT*16+DK'IVWE
15050 POKE S1+6,SU*16+RE: POKE S2+6,SU*16+RE'IVDF
15060 POKE S1+4,33: POKE S2+4,33'ENFB
15070 FOR I=1 TO 255'DFLF
15080 POKE S1+1,I'CFGA
15090 POKE S2+1,255-I'DIDC
15100 NEXT 'BAEW
15110 POKE S1+1,0: POKE S2+1,0'ELWC
```


```
15120 RETURN 'BAQY
16000 REM "### FUNNY SIREN ###"'BAAB
16010 S1=54272:S2=54279:AT=0:DK=0:SU=15:RE=0'GHPD
16020 FOR I=0 TO 24: POKE S1+I,0: NEXT 'GLWF
16030 POKE S1+24,15'CHNC
16040 POKE S1+5,AT*16+DK: POKE S2+5,AT*16+DK'IVWF
16050 POKE S1+6,SU*16+RE: POKE S2+6,SU*16+RE'IVDG
16060 POKE S1+4,17: POKE S2+4,17'ENJC
16070 J=30: FOR I=1 TO 255 STEP 5'FKYD
16080 POKE S1+1,I: POKE S2+1,J'ELSE
16090 FOR K=1 TO 20: NEXT :J=J-.5'GKEF
16100 POKE S1+1,0: POKE S2+1,0'ELWC
16110 FOR K=1 TO I/3: NEXT 'FFFD
16120 NEXT 'BAEA
16130 RETURN 'BAQB
17000 REM "### WIERDO SIREN ###"'BADD
17010 S1=54272:S2=54279:AT=0:DK=0:SU=15:RE=0'GHPE
17020 FOR I=0 TO 24: POKE S1+I,0: NEXT 'GLWA
17030 POKE S1+24,15'CHND
17040 POKE S1+5,AT*16+DK: POKE S2+5,AT*16+DK'IVWG
17050 POKE S1+6,SU*16+RE: POKE S2+6,SU*16+RE'IVDH
17060 POKE S1+4,17: POKE S2+4,17'ENJD
17070 FOR I=1 TO 1'DDEB
17080 FOR J=1 TO 255 STEP 2'EGAD
17090 POKE S1+1,J'CFHD
17100 POKE S2+1,( INT ( SIN (J)*20)+50)'GPPF
17110 NEXT 'BAEA
17120 FOR J=255 TO 1 STEP -2'FGPF
17130 POKE S1+1,J'CFHE
17140 POKE S2+1,( INT ( SIN (J)*20)+50)'GPPD
17150 NEXT 'BAEE
17160 NEXT 'BAEF
17170 POKE S1+1,0: POKE S2+1,0'ELWE
17180 RETURN 'BAQB
18000 REM "###|SPACE14|###"'BANC
18010 S1=54272:S2=54279:AT=0:DK=0:SU=15:RE=10'GIOF
18020 FOR I=0 TO 24: POKE S1+I,0: NEXT 'GLWB
18030 POKE S1+24,15'CHNE
18040 POKE S1+5,AT*16+DK: POKE S2+5,AT*16+DK'IVWH
18050 POKE S1+6,SU*16+RE: POKE S2+6,SU*16+RE'IVDI
18060 POKE S1+4,129: POKE S2+4,33'EOJE
18070 POKE S1+1,20: POKE S1+4,128'EOAF
18080 FOR I=1 TO 1000: NEXT 'EHFE
18090 POKE S1+4,33'CGNE
18100 FOR I=1 TO 5'DDIC
18110 POKE S1+1,30: POKE S2+1,32'ENUA
18120 FOR J=1 TO 500: NEXT 'EGMF
18130 POKE S1+1,20: POKE S2+1,23'ENTC
18140 FOR J=1 TO 500: NEXT 'EGMB
18150 NEXT 'BAEF
18160 POKE S1+1,0: POKE S2+1,0'ELWE
18170 RETURN 'BAQB
```



□

BOOKS & THINGS

Reviews of the latest Software & Publications available.

MICROCOMPUTER ART

Title: Microcomputer Art
Author: Ross Edwards
Publisher: Prentice-Hall Australia
Price: R.R.P. $19.95
Reviewed by: Peter Davies

Fantastic, beautiful.... were my first reactions to flicking through the pages of this book. Patterns generated by mathematical formulae hold a fascination for me and I was tempted to begin typing in programs immediately. I resisted on the grounds that it might be better to read the instructions first.

The book constructively and methodically develops programs to emulate engraving machines which were used from 1780 to 1914 to satisfy the then trend of engraved decorative objects. To provide added interest, a short account of these mechanical devices is included along with their relationships with a computer. Because of this mechanical aspect I query the title 'art' – the contents of the book are not quite what one would expect from the title.

Chapter 3 introduces the reader to the framework program required for the rest of the book and begins by plotting a circle on the screen using polar co-ordinates (see my article in 'Commodore Magazine', Vol. 5 No 1 !). It should be stressed here that an understanding of the mathematics involved is not required to be able to use any of the programs in the book. In some sections I would have liked to have seen more mathematical detail rather than 'it may be deduced that' but I am certain the average reader would not !

Different versions of the framework programs are given for eight different microcomputers. For the Commodore 64, two versions are offered. One is 'straight' BASIC as per the user's manual and we all know how slow that is. The other is for a Superexpander cartridge and can easily be adapted to Simon's Basic and the like as well as other Commodore computers.

The machine language assistance of Superexpander really is essential for some of the programs where BASIC would take over 30 minutes to run. I suspect the BASIC program will produce elongated pictures as the difference in pixel height and width has not been taken into account. Y = Y/1.23 will fix it. Superexpander has this built in. I have not looked in detail at the programs for other machines.

The remaining chapters deal with increasingly more complex curves with literally hundreds of examples, 643 to be exact. Variable values to reproduce each one are given in a large table at the end of the book. The examples take up around 130 pages of the 227 page book – apparently leaving little experimentation for the reader. How untrue – within minutes of typing in any program you can have distinctly different patterns from those in the book.

I ran a few programs and they worked perfectly. The workhorse section of each is as small as four lines and could be crunched into less with a small saving in running time; the longest programs can be typed in just a few minutes.

I suspect a further time saving in running time could be made by using DEF FN and calling the function from within the loops instead of the BASIC interpreter having to deal with a complex formula for each increment of a loop.

The main reason for the author's inclusion of so many diagrams is because he is trying to develop patterns in the values and relationships between the variables so as to predict the characteristics of a pattern before running it. In chapter 8 a summary and guide lines are provided as a conclusion. This aspect gives the impression that the book was originally some sort of thesis.

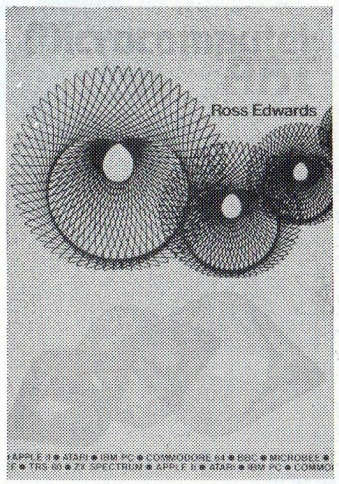
The advanced patterns in the last section of the book are extremely intricate, perhaps too much so for adequate resolution on a normal TV set (and maybe also on a monitor for some of them). The detail will only be shown on a plotter (such as the 1520). An omission from the programs is a routine for a plotter though it is easy to write one for oneself.

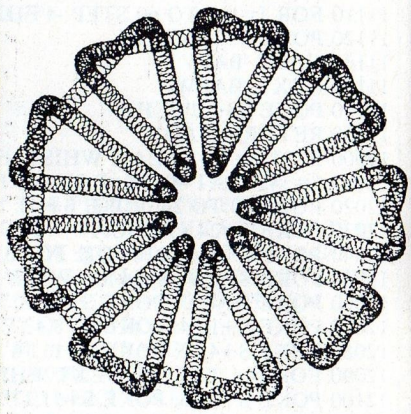
Dotted around the text are numerous quotations which I found of interest. Also little programming gems are found here and there to speed up execution. These techniques use the symmetry of the patterns to plot as many as four points from a single step in a loop. It is fascinating to watch the pattern grow from four directions simultaneously.

So what use is the book and for whom ?

It is fun – well, I think so but it is not everyone's cup of tea. One can approach the book from an art, graphics, maths or computing point of view so there's something to learn for everyone. The book is a must for every school/college library for use in all of these areas – I am certainly going to recommend its purchase by the school I work for (the graphics teacher was 'rapt').

For home use, unless, like myself, the purchaser has an interest in





such things then it is not a book the average amateur computerist would buy for him/herself but s/he would be pleased to receive it as a present.

Value for money ? Well, with 130 pages of examples there are only about 90 pages of explanations and programs – even then most of the programs are the same but different. Again it depends on your interest in the topic.

COMMODORE 64 GRAPHICS

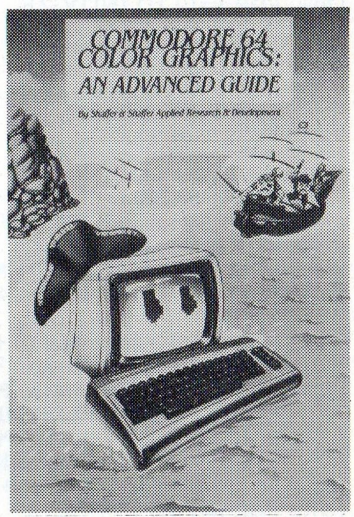
Title: Commodore 64 Graphics
An Advanced Guide
Author: Shaffer & Shaffer Applied Research & Development
Publisher: Prentice-Hall Australia
Price: R.R.P. $23.95
Reviewed by: Peter Davies

This book is a sequel to Commodore 64 Color Graphics: A Beginner's Guide (which I have not seen). While there are references to the first book, it is not essential to have it to be able to use the second one.

The authors point out that to use the Advanced Guide you do need to have some programming knowledge but that knowledge does not have to be extensive. By following the chapters and entering the 'Tools' as you come across them, by the end of the book you have a Simon's Basic type of program with some machine language assistance. Commands are not programmable as such but are called by various GOSUBS.

The book can be used in either of two ways. You can simply type in the program and follow the instructions on how to use it. However, each tool has an explanation as to how it works so you can also improve your programming skills by understanding what is going on. Some of the explanations are, of necessity, mathematical but don't let this put you off – you don't have to read them if you don't want to. All the explanations are given in sections with a blue border after the instructions for the use of a particular subroutine.

The writing style is very easy to follow and understand, and examples using each subroutine are given so you can see what it does. Also there are a few exercises to ensure that you have got the idea.



After an introductory chapter a macine language program has to be entered for the HiRes routines, PLOT, PAINT and SPRITE commands. A double check is made of the entered machine code and I found the code to be error free in use. In fact I have not found any errors in any of the programs though there is a minor error in the matrix notation on page 123 which I would think is typographical.

I found the book most interesting in that you can quickly learn to draw your own shapes and paint them. A completed picture can be used later in the book as a background for a game involving sprites. In particular I was delighted with the explanations for translation, rotation and dilation (moving,

spinning and enlarging) of a shape you have designed especially as I was teaching the topic to Year 11 students at the same time as I was working my way through the book.

If your interest is in designing your own games you should gain a great deal from this book. The great advantage is that the programs you develop are self contained and so do not rely on a cartridge to help. An added advantage is that a picure can be saved on disc and read straight into memory.

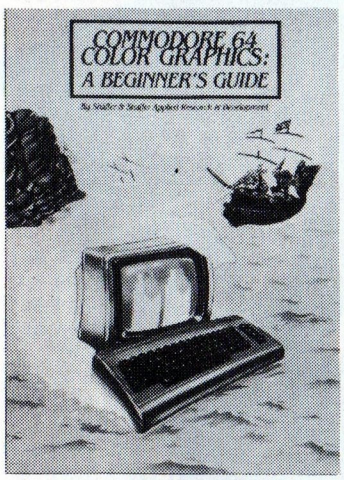
My main interest at the moment is in developing educational software and there are inumerable uses of this book for this purpose. An offshoot that never entered my head was spotted by my daughter – the program can be used to design knitting patterns and compare colour combinations.

Compared with many 'computer' books this one represents exce llent value for money.

MORE GRAPHICS!!

Title:COLOUR GRAPHICS:
A Beginners Guide
Author: Shaffer and Shaffer Applied Research & Development
Publisher: Prentice-Hall Australia
Price: R.R.P. $23.95
Reviewed by: Wayne B. Hodges

This is one of two well set-out manuals which examines and instructs you in the art of Creating Colour Graphics for the Commodore 64. There are two books in the series for Beginners and Advanced users. In the beginners guide you will expect to learn about creating



Graphics for your C-64, providing you do have an understanding of a few simple Commands, such as Program, line number, goto and run. Now if you can handle this, away we go.

As soon as you open this book you are immediately confronted by a two masted Sailing Ship from another era, which is colourfully designed by your Commodore 64. This book promises this as a reward for your efforts. The introductions aside we get into the first Chapter.

Each Chapter displays a portion of the program needed to draw your ship, along with a list of instructions about your task. The first Chapter will establish the organisation of the final program, and to familiarize you with "subroutines". This Sector is quite elementary and easy going for most users.

The next Chapter involves itself with the recreating of the picture of your reward, "The Ship", examining High Resolution Graphics, and the implementation in your programme. An interesting thing, each section of the programme is quite thoroughly examined and provides exactly what it does, and an excellent line-by-line description.

In the next Chapter the plot thickens indeed, you will learn how to paint the blue Background Sky, and how to draw in the water. I am enjoying this trip immensely, each step is explained in "Plain Everyday English".

After a few Chapters your efforts are reviewed, and it is apparent by this stage (at Chapter Four) you have learned to:

★ Turn on High res Graphics
★ Change the colour codes
★ Turn the screens pixels to background colour
★ Plot specific and random points
★ Change colours of a block and
★ return to text mode
★ ZAP the main routine to draw a new picture.

Phew! You say and pat yourself on the back, impossible I hear you mutter, but no it is not with this most exciting advance in learning about Graphics. This book is so well explained you will amaze yourself.

In Chapter five things begin to take shape, more to the point, in the shape of a Ship, this part of your tour into Graphics you actually are ready to add the Ship to your picture. The final Chapter teache you about sprite graphics, which is most useful for you amateur programmers who wish to know more or simply as a next step in your learning process.

At the end of this manual you will find a number of appendices for reference, such as design charts, colour charts and a further reading list.

Conclusion
After experiencing this Book myself I honestly found it most instructive and thoroughly interesting. The main thrust of this "Beginners Guide to Colour Graphics" is to show you simply and surely how to create your own works of art. I found it a must for anyone interested in furthering their knowledge here, I can recommend this book for a start and then progressing to more advanced Graphic design.

I SPY!!

Title: SPY VS SPY
Tape-64
Author:Mike Riedel
Publishers: Beyond Software
Price: R.R.P. $24.95
Reviewed by: William Leader

Spy vs Spy is based on the MAD magazine's cartoon of the same name. Two spys, one black, one white, must battle against one another and against the clock to get the secret plans and everything else needed for a speedy getaway. The main problem is that there are five items needed to escape and only one of each in the maze like complex.

Each spy has a club which he uses to fight his opponent and also a selection of devious traps, ranging from the old time bomb to string triggered guns and buckets of water. The screen is split into two parts, each spy moves indepenndently of the other on his half of the screen and they only meet when they enter the same room together. When they meet both spys appear in the top screen and must fight or leave the room. Death is not final it merely means you lose any items you were carrying and also some time. Joysticks are required to play and the game can be played against the computer or an opponent.

The game is very well done and the graphics are very good, especially the animation. The game retains much of the humour of the cartoon, as traps go off and the spys giggle quietly to themselves. The game is very playable, it is simple enough to make it playable, yet complex enough to make it interesting.

Though hard to fault there is one thing I find frustrating. After playing for some time you invariably become very good at it and you must improve the computer 4opponent's IQ to provide a challenge. When you do this however the computer becomes so good that it doesn't make mistakes and the game isn't as much fun. This is a minor fault and one which many people would not consider but it is true of most computer opponent games.

Spy vs Spy is an excellent game, it is well presented and well written. Anybody who likes something slightly different will love this game. It is well worth the money and is one of the better programs on the market today.

STELLAR - 7

Title: STELLAR - 7
Disk/Tape-64
Author: Damon Slye
Publisher: US GOLD
Price: R.R.P. $29.95
Reviewed by: William Leader

Three dimensional vector graphics at their best. STELLAR-7 is an excellent rendition of the arcade game BATTLEZONE. In a new anti-gravehical you must rid a series of planets of hostile forces. There are many different enemies, each with different firepower and armour.

These are represented by hollow 3-D images which move and rotate realistically. You have the latest weaponry and defence devices with which you must fight your way to the last planet and the evil mind behind it all, the nasty Gir Draxon. Joystick is optional but desirable.

The graphics are excellent. They are smooth and well designed. The game-play is very good because it requires strategy, cunning and a good eye to be successful. The sound, what there is of it, is reasonable.

The main fault of the game is speed. The graphics are smooth but not exceptionally fast. Everything seems much faster than yourself which is frustrating when you are pursuing an enemy. The level system is also annoying. In the first level you can only get to the fourth planet before you must start again on the next level.

Despite these problems, the game is very good and good value. If you like strategy games with lots of action, then this is for you, but if you want blindingly fast action then maybe you had better have a good look first.

F-15 STRIKE EAGLE

Tape/Disk – C64
Author: Grant Irani
Publisher: US GOLD
Price: R.R.P. $39.95
Reviewed by: William Leader

If you're tired of trying to do tricky manouvers in your flight simulator and failing then F-15 can solve all your problems. It is a simulator which simulates sorties over modern trouble spots, such as Libya and Vietnam. Your plane is equipped with the latest selection of missiles, bombs and defence

CONTINUED OVERLEAF

BOOKS & THINGS

CONTINUED FROM PREVIOUS PAGE

devices. The main difference with this simulator is the power of the F-15, making rolls, loops and other fancy stunts not only easy but essential. This makes the game very playable.

The aim is to bomb a primary target and as many secondary targets as possible before returning to base, avoiding SAM'S and other planes as you do. A joystick is desirable but not necessary.

The manouverability of the plane makes the game extremely playable. The diversity of the weapons and targets adds to the interest of the game. It is very well presented and well thought out.

My main criticism of F-15 is the graphics. While the game is very fast, the graphics suffer slightly. The ground targets are nothing more than flat triangles. If they were made into solid objects the game would be more enjoyable. Also the game does not cater for taking off and landing. You begin in the air and merely have to fly over the base to return to it. If these minor faults were dealt with the game would be superb.

As it is, the game is very good and good value if you have the money. It caters not only for simulation enthusiasts but also for the arcade buffs. F-15 should suit anybody who likes a good game.

RAID OVER MOSCOW

Title: RAID OVER MOSCOW
Tape/Disk – C64
Author: B. Carver
Publisher: Access Software
Price: R.R.P. $29.95
Reviewed by: William Leader

Russia has fired missiles at one of America's cities! You are America's only hope. You must manouver your planes out of the hanger (not too easy) and fly down to the Soviet missile base. You must penetrate its defences and blow up the silo. Do this to three launching sites and Moscow is in your sights. Fighting your way into the headquarters (with your bazooka) brings you face to face with the robot controller. Destroying this robot with your bouncing disks brings destruction to the Moscow defences and victory to America. A joystick is required to play.

The game is a follow-up of BEACHHEAD, though the graphics are better. The different stages are in full perspective, including the approach to the silos and Moscow which features diagonal scrolling. The game-play is superb, with so much variety you can't possibly get bored with it. The graphics are good with shadows and 3D effects.

It is hard to fault and the only problem I found was the abrupt end after you have completed the mission. The game could continue after this with a new scenario on the next level.

This exceptional game is better than BEACHHEAD and more interesting. Anyone who likes arcade action will like RAID OVER MOSCOW because of the diversity of the stages. It is well worth buying and a must for BEACHHEAD fans.

SUB - COOKING!!

Title: Commodore 64 Subroutine Cookbook
Author: David D. Busch
Publisher: Prentice-Hall Australia
Price: R.R.P. $12.95
Reviewed by: Wayne B. Hodges

This is for all those programmers out there, who, when in the midst of writing a masterpiece are left scratching their head. Now this process is not because of any ailment of the scalp region, but they are left wondering what it is they need to complete their task. This book can help in this area, it provides many helpful routines to add to your programme.

Now if you have read this far, then you are interested, now with that in mind let me lead you through the benefits of this Computer-side companion. Each Chapter has a number of features which assist us in understanding exactly what a particular sub-routine does; the end result, it presents line by line details describing each function.

After the initial formalities we are confronted with the first chapter, which explains the simple approach to merge these subbies into your programs.

After one we meet Chapter two (amazing). This one examines Joysticks in some detail, this one provides a numberj of routines for your joysticks, where else would you get subbies like these. Chapter three is well timed, it shows you how to use the clock in your programs. About time, eh!!

The next chapter sounds good too, it has selected a number of well thought out subbies that can do wonders to your masterpiece.

Chapter five is a bit tricky, it is a mine of information, simply an amazing encyclopaedia of PEEKs and POKEs for all functions. Getting down to basics is happening in Chapter Six, and in Seven provides a number of very handy routines, such as, dealing cards, flipping coins and many others.

Chapter Eight looks at data files, and Nine gives you the business by showing us a number of handy routines that would complement any programme for extra interest.

Basic Functions, Bits, Bytes and Communications are examined in the final chapters.

SUMMARY

Now simply this novel publication from our friends at Prentice-Hall is essential reference for any of you delving into the world of programming. In my opinion this book would be a most invaluable guide and friend in need during those quiet and lonely hours at the Commodore 64 keyboard.

THE WIZZ

Title: The Wizz
Disk/Tape for the Commodore 64
Authors: P. Spiess & M. Haines
Publishers: Edu-Kit Productions

"Let the Wizz show you the FUN-WAY to help you with MATHS".

This is a chess-like game for one or two players which teaches the addition and subtraction of negative and positive numbers in an entertianing manner.

Players can either use a Keyboard of Joystick.

The aim of the game is to gain the highest score and at the same time keep their opponent's score as low as possible.

* A more detailed review of THE WIZZ will appear in a future issue.

TABLES GRID

Title: Tables Grid
Publisher: Edu-Kit Productions

Tables Grid is designed to help you learn your mulitplication tables off by heart in an entertaining manner.

Tables Grid is unique in that at the end of each game you can see at a glance your total tables knowledge in graphic form.

Tables appear on the left hand side of the screen and as you type in the answers one of four types of grid patterns is created on the right hand side of the screen.

At lower levels of difficulty, these include a hearts grid, electric grid or a maze grid. If you choose the most difficult level, then you are rewarded by any one of several fascinating picture jig saws being created on the right hand side of the screen as you get each table correct.

If you get a table wrong, the correct answer will appear on the screen. For each game, you can choose a time limit from 3 to 39 seconds in which to type in your answer.

An added feature of Tables Grid is that you can get a "print out" of your results which shows a precise graphic record of your progress and performance.

* A more detailed review of Tables Grid will appear in a future issue.

CONTINUED ON PAGE 50

ADDRESS BOOK64

A. Wiggins

Address book will hold names and addresses, that can be sorted into postcode order, or alphabetical order.

The sort can be made on the cassette player if needed.

```
0 REM I CAN HOLD 100 ENTRIES TO CHANGE THIS ADJUST
  (D1)IN LINE NO.22'BBXM
1 REM A WIGGINS 1 CHESTER MELBOURNE VIC.'BEXH
2 PRINT "[CLR,DOWN2]ENTER PRINTER DEVICE NUMBER.":
  PRINT " 2 TO 7 ONLY.""CBHM
4 PRINT "IF YOU DO NOT HAVE A PRINTER ENTER (2).""BACM
6 PRINT "IF YOU HAVE AN (ASR 33) ENTER (3).": POKE 198,0'CGKO
8 Z$="": GET Z$: IF Z$="" THEN 8'FJRL
10 Z=VAL (Z$): IF Z<2 OR Z>7 THEN 8'HLQE
12 IF Z=3 THEN Z=128'EGYD
14 IF A=>1 AND Z>3 AND Z<128 THEN CLOSE Z'JJSJ
16 POKE 50000,Q: POKE 50001,Z: CLR 'DQSI
18 IF PEEK (50001)=128 THEN OPEN 128,2,3,
   CHR$(163)+ CHR$(160)'IDGP
20 Z=PEEK (50001): IF Z<128 AND Z>3 THEN OPEN Z,Z'ISXH
22 D1=101: DIM A$(D1,11),B$(11),J$(D1,11):P$="PHONE NO.""DKYL
24 AN$="SORRY FOR THE DELAY IT'S MY DAY OFF""BDWM
26 A=1:Q=PEEK (50000): IF Q=0 THEN 500'GRGL
28 IF Q=5 THEN 112'DFDI
30 POKE 198,0: REM EMPTY KEYBOARD BUFFER'CAWG
32 K$="": PRINT : PRINT "[UP,SPACE3]PRESS RETURN WHEN
   READY""DEOL
34 GET K$: IF K$<> CHR$(13) THEN 34'GLYJ
36 PRINT "[UP,SPACE30,UP]": RETURN 'CBSL
60 A$(A,0)="STOP": OPEN 1,1,1,N$'CQBH
62 FOR X=1 TO A: FOR W=0 TO 11'GIEJ
64 PRINT#1,A$(X,W): NEXT W'CLCJ
66 NEXT : CLOSE 1'CCPJ
68 GOTO 500'BDEL
70 OPEN 1,1,0,N$'BINF
72 IF A=>D1 THEN A$(D1,0)="STOP": CLOSE 1: GOTO 130'HRWO
74 INPUT#1,A$(A,0): IF A$(A,0)="STOP" THEN CLOSE 1:
   GOTO 500'GWYQ
76 FOR W=1 TO 11: INPUT#1,A$(A,W): PRINT A$(A,W):
   NEXT 'GXTR
78 A=A+1'CDPN
80 GOTO 72'BCQE
100 PRINT "[CLR,DOWN]""BALV
102 PRINT : PRINT "DO YOU WISH TO ADD NEW LIST TO
    CURRENT[SPACE2]LIST Y/N ?.""CBFL
104 POKE 198,0'BFXA
106 Q$="": GET Q$: IF Q$<>"Y" AND Q$<>"N" THEN 106'JNGK
108 IF Q$="N" THEN Q=5: GOTO 14'FHII
110 IF Q$="Y" THEN 112'DFUY
112 PRINT "[CLR,DOWN]""BALY
114 PRINT "ENTER NAME OF LIST YOU WISH TO LOAD""BAAK
116 N$="": INPUT N$: GOTO 70'DIBF
118 PRINT "[CLR,DOWN]": IF A<=1 THEN PRINT "[RVS]SORRY
    NOTHING TO SAVE YET[OFF]": GOSUB 30: GOTO 500'IKET
120 PRINT "ENTER NAME OF LIST YOU WISH TO SAVE""BAPH
122 N$="": INPUT N$: GOTO 60'DIAC
130 IF A=>D1 THEN A$(D1,0)="STOP": PRINT : PRINT
    "[RVS]ADDRESS BOOK FULL[OFF]": GOSUB 30:
    GOTO 500'JUCO
132 PRINT "[CLR,DOWN]":AD$="NO'S 1&2 FOR NAME.NO'S 3/10
    FOR ADDRESS.NO.11 FOR PHONE NO.""CEWR
134 PRINT : PRINT "TO END ENTER (STOP) FOR FIRST NAME":
    PRINT AD$: PRINT 'EGGO
136 PRINT "JUST PRESS RETURN IF ADDS COMPLETE":
    PRINT 'CBXO
138 PRINT "THIS IS ADDRESS NO."A'BBNL
140 INPUT "(1) FIRST NAME";A$(A,1): IF A$(A,1)="STOP"
    THEN 500'ETTJ
142 IF A$(A,1)="" THEN PRINT "[UP2]": GOTO 140'FLBH
144 INPUT "(2) LAST NAME";A$(A,2): IF A$(A,2)="" THEN
    PRINT "[UP2]": GOTO 144'GUUO
146 B=3: PRINT "[DOWN]ADDRESS 3 TO 10": PRINT "ADD POST
    CODE TO THE LAST ENTRY.""DEDU
147 PRINT B;:A$(A,B)="": INPUT A$(A,B)'DSNL
148 IF A$(A,B)="" THEN GOSUB 5000'ELEL
149 IF A$(A,B)="" THEN FOR X=B TO 10:A$(A,X)="N/A": NEXT :
    GOTO 152'JYOT
150 B=B+1: IF B<=10 THEN GOTO 147'HKTG
151 IF A$(A,10)<>"N/A" THEN GOSUB 5010'FMUH
152 INPUT "(11) PHONE NO.";A$(A,11)'BJLH
154 A$(A,11)=P$+A$(A,11)'CSRI
156 A=A+1'CDPI
158 GOTO 130'BDDJ
160 PRINT "[CLR,DOWN2]""BADC
162 IF A<=1 THEN PRINT : PRINT "SORRY NOTHING HERE TO
    SEE.": GOSUB 30: GOTO 500'IKVR
163 PRINT "[CLR,DOWN2]DO YOU WISH TO PRINT NAMES
    ONLY. Y/N""BABP
164 L$="": GET L$: IF L$<>"Y" AND L$<>"N" THEN 164'JNPO
165 PRINT "[DOWN]DO YOU WISH TO PRINT ALL NAMES
    A TO Z.[SPACE16,RVS,SPACE3]Y/N[SPACE3,OFF]""BAQU
166 M$="": GET M$: IF M$<>"Y" AND M$<>"N" THEN 166'JNVQ
167 IF M$="N" THEN PRINT "ENTER INDEX YOU WISH TO SEE
    (A TO Z) ?.""ECLV
168 IF M$="N" THEN M1$="": GET M1$: IF M1$<"A" OR M1$>"Z"
    THEN 168'KTQU
169 PRINT "[CLR,DOWN2]""BADL
170 X1=0: FOR X=1 TO A-1: IF M$="N" AND
    LEFT$(A$(X,2),1)<>M1$ THEN 187'MDEQ
171 IF L$<>"Y" THEN PRINT A$(X,0)'FJXI
172 PRINT A$(X,1)" "A$(X,2):X1=1'CSDJ
174 IF L$="Y" THEN 182'DFWJ
176 FOR W=3 TO 11: IF A$(X,W)<>"N/A"
    THEN PRINT A$(X,W)'ITPS
178 NEXT 'BAEK
180 PRINT 'BACD
182 PRINT "PRESS RETURN TO CONTINUE""BAFM
184 GET Z$: IF Z$<> CHR$(13) THEN 184'GMIN
186 PRINT "[UP,SPACE25,UP]""BAUN
187 NEXT : IF X1=0 THEN PRINT "SORRY NO "M1$" 'S LISTED.":
    PRINT 'GIJU
188 PRINT : PRINT TAB(10)"[RVS]END OF LIST.[OFF]": FOR X=1
    TO 500: NEXT : GOSUB 30: GOTO 500'JSCX
190 SP=0: FOR X=1 TO A-1:W=LEN (A$(X,1)): IF W>SP THEN
    SP=W'LBRR
191 NEXT :SP=SP+2: PRINT "[CLR,DOWN2]YOU CAN OMIT (2)
    NAMES FROM GOING[SPACE7]TO PRINT#.""EHLW
192 PRINT "DO YOU WISH TO OMIT ANY Y/N ?.":B1$="":
    B2$=""'DIER
193 B$="": GET B$: IF B$<>"Y" AND B$<>"N" THEN 193'JNCQ
194 IF B$="N" THEN 200'DFRL
195 PRINT "JUST HIT RETURN FOR FIRST NAME IF
    YOU[SPACE3]CHANGED YOU MIND.""BATY
196 B$="": INPUT "(1) FIRST NAME";B$:B1$=B$: IF B$=""
    THEN 200'GSMU
197 B$="": INPUT "(1) LAST NAME";B$:B1$=B1$+B$'EPUT
198 B$="": INPUT "(2) FIRST NAME";B$:B2$=B$: IF B$=""
    THEN 200'GSOW
199 B$="": INPUT "(2) LAST NAME";B$:B2$=B2$+B$'EPXV
200 IF Z=2 THEN PRINT : PRINT "[RVS]SORRY NO PRINTER
    ASKED FOR AT START OF[SPACE2]PROGRAM.[OFF]":
    PRINT: GOSUB 30: GOTO 500'ILBQ
201 IF A<=1 THEN PRINT "[CLR,DOWN2]SORRY NOTHING HERE
    TO PRINT#.": GOSUB 30: GOTO 500'HJFL
202 PRINT "[CLR,DOWN2]DO YOU WISH TO PRINT# NAMES
    ONLY. Y/N""BALJ
203 L$="": GET L$: IF L$<>"Y" AND L$<>"N" THEN 203'JNJI
204 IF L$="Y" THEN 207'DFUD
205 PRINT "[DOWN]DO YOU WISH TO PRINT# PHONE NO.
    Y/N""BAWK
206 Q$="": GET Q$: IF Q$<>"Y" AND Q$<>"N" THEN 206'JNHL
207 PRINT "[DOWN]DO YOU WISH TO PRINT ALL NAMES A TO
    Z.[SPACE16,RVS,SPACE3]Y/N[SPACE3,OFF]""BAQR
208 M$="": GET M$: IF M$<>"Y" AND M$<>"N" THEN 208'JNSN
209 IF M$="N" THEN PRINT "ENTER INDEX YOU WISH TO SEE
    (A TO Z) ?.""ECLS
210 IF M$="N" THEN M1$="": GET M1$: IF M1$<"A" OR M1$>"Z"
    THEN 210'KTEI
211 PRINT "[CLR,DOWN2]GONE TO PRINT#.": GOSUB 30:C=A:
    X1=0: FOR A=1 TO C-1:B$=A$(A,1)+A$(A,2)'KHDP
212 IF B$=B1$ OR B$=B2$ THEN 230' FN NE
216 IF M$="N" AND LEFT$(A$(A,2),1)<>M1$ THEN 230'HTJL
218 IF L$="Y" THEN PS=LEN (A$(A,1)): PRINT#Z,A$(A,1)
    SPC(SP-PS)A$(A,2):X1=1: GOSUB 234: GOTO 230'LWMW
220 PRINT#Z,A$(A,1)" "A$(A,2):X1=1: GOSUB 234'DYLE
222 FOR W=3 TO 10: IF A$(A,W)<>"N/A" THEN
    PRINT#Z,A$(A,W): GOSUB 234'JAVL
223 NEXT 'BAEB
224 IF Q$="Y" THEN PRINT#Z,A$(A,11): GOSUB 234'FQGI
226 PRINT#Z: GOSUB 234: GOSUB 30'DIXH
230 NEXT 'BAEY
231 IF X1=0 THEN PRINT "SORRY NO "M1$" 'S LISTED.":PRINT :
    GOSUB 30'GKOL
```

CONTINUED OVERLEAF

ADDRESS BOOK - CONTINUED FROM PREVIOUS PAGE

```
232 GOTO 500'BDEC
234 PRINT#Z, CHR$(0)+ CHR$(0)+ CHR$(0);'GMHJ
236 SR=ST: IF ( PEEK (673) AND 1) THEN 236'FQHL
238 RETURN 'BAQH
240 Q1=A-1: PRINT 'DFKD
242 O1$=A$(Q1,2)+A$(Q1,1)'CTRH
244 L=0: FOR I=1 TO Q1-1:A$=A$(I,2)+A$(I,1)'HACN
246 PRINT "[UP,SPACE37,UP]"'BARM
248 PRINT Q1;: PRINT A$(I,1)" "A$(I,2)'CSCN
250 IF A$<=O1$ THEN 254'EI IF
252 L=1:O1$=A$: FOR B=0 TO 11:B$(B)=A$(Q1,B):
    A$(Q1,B)=A$(I,B):A$(I,B)=B$(B): NEXT 'JHKU
254 NEXT : IF L<>0 THEN 261'FGCJ
256 FOR Z=1 TO A-2:Y=Z+1'GIYM
257 C$=A$(Z,2)+A$(Z,1)'CQHM
258 D$=A$(Y,2)+A$(Y,1)'CQGN
259 IF C$>D$ THEN L=1'EGYN
260 NEXT : IF L=0 THEN 6000'EHAF
261 Q1=Q1-1: IF Q1>1 THEN 242'FMYI
262 L=0: GOTO 6000'CHHG
500 PRINT "[CLR,DOWN2]"'BADA
502 PRINT "WHAT WOULD YOU LIKE TO DO ?."'BARI
504 PRINT "MAKE OR ADD TO A LIST (1)."'BAVJ
506 PRINT "SEE LIST[SPACE14](2)."'BACK
508 PRINT "SEND LIST TO PRINTER# (3)."'BAXO
510 PRINT "SAVE LIST TO TAPE#[SPACE4](4)."'BADG
512 PRINT "GET LIST FROM TAPE#[SPACE3](5)."'BAOI
514 PRINT "MAKE CHANGES[SPACE10](6)."'BALJ
516 PRINT "SORT[SPACE18](7)."'BAWK
519 PRINT "FIND ADDRESS BY NAME[SPACE2](8)."'BAGP
520 PRINT "END[SPACE19](9)."'BARF
530 POKE 198,0'BFXD
540 Q$="": GET Q$: IF Q$="" THEN 540'FLMI
541 Q=VAL (Q$)'CFQG
542 IF Q<1 OR Q>9 THEN 540'FHJJ
544 ON Q GOTO 130,160,190,118,100,600,700,1000,2000'CNGP
600 IF A<=1 THEN PRINT "[CLR,DOWN2]SORRY NOTHING HERE
    TO CHANGE.": GOSUB 30: GOTO 500'HJUO
601 PRINT "[CLR,DOWN2]ENTER NAME OF CHANGE."'BAAH
602 PRINT "ENTER (STOP) FOR FIRST NAME TO[SPACE15]
    RETURN TO MENU."'BAWQ
620 PRINT "ENTER AT LEAST FIRST (4) CHARACTERS FOR FIRST
    AND LAST NAME.": PRINT 'CBCT
621 C1$="": INPUT "FIRST NAME";C1$: IF C1$="STOP"
    THEN 500'FPQN
622 IF C1$="" THEN PRINT "[UP2]": GOTO 621'FHXJ
624 PRINT :W=LEN (C1$)'DHKJ
630 C2$="": INPUT "LAST NAME";C2$: IF C2$="" THEN PRINT
    "[UP2]": GOTO630'HQRO
632 V=LEN (C2$)'CGFH
640 CC=0: FOR X=1 TO A'EHQI
650 A$=A$(X,1):B$=A$(X,2): IF C1$=MID$(A$,1,W) AND
    C2$=MID$(B$,1,V) THEN CC=X:X=A'LXVW
660 NEXT 'BAEG
662 IF CC=0 THEN PRINT : PRINT "SORRY THAT NAME NOT
    LISTED.": GOSUB 30: GOTO 600'HLTV
663 PRINT "[CLR,DOWN]": PRINT A$(CC,0)'CJSM
664 PRINT A$(CC,1)" "A$(CC,2)'BQKN
665 FOR W=3 TO 11: PRINT A$(CC,W): NEXT ' FORR
666 PRINT "WHAT DO YOU WISH TO CHANGE ?."'BALU
667 PRINT : PRINT "FIRST NAME[SPACE5](1)."'CBUS
668 PRINT "LAST NAME[SPACE6](2)."'BANS
669 PRINT "PHONE NO.[SPACE6](3)."'BAXT
670 PRINT "ADDRESS[SPACE8](4)."'BARL
671 PRINT "[RVS]REMOVE PERSON FROM FILE[OFF](5)"'BABQ
672 PRINT "TO RETURN TO[SPACE7]"'BAHN
673 PRINT "MENU[SPACE11](6)."'BAJN
674 POKE 198,0'BFXM
675 Q$="": GET Q$: IF Q$="" THEN 675'FLVR
676 Q=VAL (Q$): IF Q<1 OR Q>6 THEN 675'HNJU
678 PRINT : ON Q GOTO 680,681,682,684,6100,500'DBTV
680 INPUT "(1) FIRST NAME";A$(CC,1): GOTO 663'CNYP
681 INPUT "(2) LAST NAME";A$(CC,2): GOTO 663'CNWP
682 A$="": INPUT "(11) PHONE NO.";A$: IF A$="" THEN 663'FMTS
683 A$(CC,11)=P$+A$: GOTO 663'DRVQ
684 D1$="[DOWN]TO WIPE OUT LINE BELOW AND ANY
    FURTHER[SPACE2]LINES ENTER N/A"'BDJC
685 D2$="IF LINE BELOW INCORRECT ENTER CORRECT
    [SPACE3]DATA. IF LINE OK PRESS RETURN.[DOWN]"'BDKH
688 PRINT "[CLR,DOWN2]": PRINT A$(CC,0): PRINT A$(CC,1)
    " "A$(CC,2): PRINT : PRINT D1$: PRINT D2$'GKOC
692 FOR W=3 TO 10: PRINT A$(CC,W): PRINT : PRINT W;:
    INPUT A$(CC,W)'HBKV
693 IF A$(CC,W)="N/A" THEN GOSUB 5020'EMIR
694 IF A$(CC,W)="N/A" THEN FOR X=W TO 10:A$(CC,X)="N/A":
    NEXT :W=10'JBYB
695 PRINT "[CLR,DOWN2]": PRINT A$(CC,0): PRINT A$(CC,1)
    " "A$(CC,2)'DBXV
696 FOR X=3 TO W: PRINT A$(CC,X): NEXT : PRINT D1$:
    PRINT D2$'HVYX
697 NEXT 'BAEQ
698 IF A$(CC,10)<>"N/A" THEN GOSUB 5030'FNQX
699 GOTO 663'BDOT
700 IF A<=1 THEN PRINT "SORRY NOTHING HERE TO SORT.":
    GOSUB 30: GOTO 500'HJXO
701 IF A=2 THEN PRINT "[CLR,DOWN2]SORRY YOU ONLY HAVE
    ONE ITEM.": GOSUB 30: GOTO 500'GJHQ
702 PRINT "[CLR,DOWN2]IF YOU HAVE A LOT OF ENTRIES IT
    WOULD[SPACE3]SORT BETTER USING YOUR ";'BBBV
703 PRINT "CASSETTE PLAYER[SPACE2]AS AN AID.YOU WILL
    NEED A BLANK TAPE."'BAIS
704 PRINT "[DOWN2]DO YOU WISH TO DO A CASSETTE SORT
    Y/N ?"'BASP
705 S$="": GET S$: IF S$<>"Y" AND S$<>"N" THEN 705'JNTP
706 PRINT "[CLR,DOWN2]WHAT DO YOU WISH TO SORT ?."'BAWP
707 POKE 198,0'BFXJ
708 PRINT "POST CODE[SPACE6](1)."'BAYN
709 PRINT "FULL SORT[SPACE6](2)."'BABO
710 PRINT "RETURN TO MENU (3)."'BAOH
711 POKE 198,0'BFXE
712 Q$="": GET Q$: IF Q$="" THEN 712'FLNJ
713 Q=VAL (Q$): IF Q<1 OR Q>3 THEN 712'HNXM
714 IF S$="Y" THEN Q=Q+3'FFEK
715 ON Q GOTO 716,240,500,8000,9000,500'CBDN
716 Q1=A-1: PRINT 'DFKL
718 O1$=A$(Q1,0)+A$(Q1,2)+A$(Q1,1)'DCER
720 L=0: FOR I=1 TO Q1-1:A$=A$(I,0)+A$(I,2)+A$(I,1)'IHHO
722 PRINT "[UP,SPACE37,UP]"'BARL
724 PRINT Q1;: PRINT A$(I,1)" "A$(I,2)'CSCM
726 IF A$<=O1$ THEN 730'EIHN
728 L=1:O1$=A$: FOR B=0 TO 11:B$(B)=A$(Q1,B):
    A$(Q1,B)=A$(I,B):A$(I,B)=B$(B): NEXT 'JHKD
730 NEXT : IF L<>0 THEN 739'FGMI
732 FOR Z=1 TO A-2:Y=Z+1'GIYL
734 C$=A$(Z,0)+A$(Z,2)+A$(Z,1)'DXEO
736 D$=A$(Y,0)+A$(Y,2)+A$(Y,1)'DXCQ
737 IF C$>D$ THEN L=1'EGYO
738 NEXT : IF L=0 THEN 6000'EHAP
739 Q1=Q1-1: IF Q1>1 THEN 718'FMCT
740 L=0: GOTO 6000'CHHH
1000 IF A<=1 THEN PRINT "[CLR,DOWN2]SORRY NOTHING
     HERE YET.": GOSUB 30: GOTO 500'HJVG
1001 PRINT "[CLR,DOWN2]TO FIND ADDRESS BY NAME."'BAAB
1002 PRINT "ENTER (STOP) FOR FIRST NAME TO RETURN
     [SPACE3]TO MENU."'BATI
1003 PRINT "ENTER AT LEAST FIRST (4) CHARACTERS FOR
     FIRST AND LAST NAME.": PRINT 'CBCN
1004 C1$="": INPUT "FIRST NAME";C1$: IF C1$="STOP"
     THEN 500'FPQH
1006 IF C1$="" THEN PRINT "[UP2]": GOTO 1004'FIRE
1008 PRINT :W=LEN (C1$)'DHKE
1010 C2$="": INPUT "LAST NAME";C2$: IF C2$="" THEN PRINT
     "[UP2]": GOTO1010'HRIF
1011 V=LEN (C2$)'CGFW
1012 PRINT :CC=0: FOR X=1 TO A'FIWB
1014 A$=A$(X,1):B$=A$(X,2): IF C1$=MID$(A$,1,W) AND
     C2$=MID$(B$,1,V) THEN CC=X:X=A'LXVP
1016 NEXT 'BAEA
1018 IF CC=0 THEN PRINT : PRINT "SORRY THAT
     NAME NOT LISTED.": GOSUB 30: GOTO 1000
1020 PRINT A$(CC,1)" "A$(CC,2)'BQKX
1022 FOR W=3 TO 11: IF A$(CC,W)<>"N/A" THEN
     PRINT A$(CC,W)'IVGH
1024 NEXT : PRINT 'CBJA
1026 PRINT "DO YOU WISH TO LIST THIS ADDRESS":
     PRINT "ON THE PRINTER# Y/N ?."'CBYQ
1028 POKE 198,0'BFXE
1030 Q$="": GET Q$: IF Q$<>"Y" AND Q$<>"N" THEN 1030'JOBF
1032 IF Q$="N" THEN 1000'DGEB
1034 PRINT "[CLR,DOWN2]DO YOU WISH TO PRINT# PHONE NO.
     Y/N"'BAGK
1036 POKE 198,0'BFXD
```

```
1038 Q$="": GET Q$: IF Q$<>"Y" AND Q$<>"N" THEN 1038'JOJN
1040 PRINT "[DOWN]GONE TO PRINT#.": GOSUB 30'CDCC
1042 PRINT#Z,A$(CC,1)" "A$(CC,2): GOSUB 1052'CXAF
1044 FOR X=3 TO 10: IF A$(CC,X)<>"N/A" THEN
     PRINT#Z,A$(CC,X): GOSUB 1052'JDDN
1046 NEXT 'BAED
1048 IF Q$="Y" THEN PRINT#Z,A$(CC,11): GOSUB 1052'FSQM
1050 GOTO 1000'BEXY
1052 PRINT#Z, CHR$(0)+ CHR$(0)+ CHR$(0);'GMHG
1054 SR=ST: IF ( PEEK (673) AND 1) THEN 1054'FRYJ
1056 RETURN 'BAQE
2000 END 'BACT
5000 R=4: IF RIGHT$(A$(A,B-1),1)=CHR$(46) THEN R=5'IVAG
5001 A$(A,0)=RIGHT$(A$(A,B-1),R)'DTVD
5002 A$=LEFT$(A$(A,0),1): IF A$< CHR$(49) OR A$> CHR$(57)
     THEN A$(A,0)="1""KIFM
5004 RETURN 'BAQB
5010 R=4: IF RIGHT$(A$(A,10),1)=CHR$(46) THEN R=5'HVMH
5011 A$(A,0)=RIGHT$(A$(A,10),R)'CTHE
5012 A$=LEFT$(A$(A,0),1): IF A$< CHR$(49) OR A$> CHR$(57)
     THEN A$(A,0)="1""KIFN
5014 RETURN 'BAQC
5020 R=4: IF RIGHT$(A$(CC,W-1),1)=CHR$(46) THEN R=5'IWJJ
5021 A$(CC,0)=RIGHT$(A$(CC,W-1),R)'DVYG
5022 A$=LEFT$(A$(CC,0),1): IF A$< CHR$(49) OR A$> CHR$(57)
     THEN A$(CC,0)="1""KKSO
5024 RETURN 'BAQD
5030 R=4: IF RIGHT$(A$(CC,10),1)=CHR$(46) THEN R=5'HWFJ
5031 A$(CC,0)=RIGHT$(A$(CC,10),R)'CVUG
5032 A$=LEFT$(A$(CC,0),1): IF A$< CHR$(49) OR A$> CHR$(57)
     THEN A$(CC,0)="1""KKSP
5034 RETURN 'BAQE
6000 REM ** SOUND SUBROUTINE **'BTHD
6020 S=54272'BGJC
6030 POKE S+24,15'C GO D
6040 POKE S+5,50'CFOE
6045 FOR X=1 TO 50'DEVJ
6050 POKE S+1, RND (X)*32+50'FKQI
6060 POKE S+4,17'CFQG
6070 FOR I=1 TO 10: NEXT 'EFJI
6080 POKE S+4,16'CFPI
6090 NEXT X'BBRH
6091 FOR X=1 TO 24: POKE 54272+X,0: NEXT 'GOFO
6092 IF L=1 THEN RETURN 'ECNL
6094 GOTO 500'BDEM
6100 PRINT : PRINT "THIS MAY TAKE A LITTLE TIME."'CBKH
6101 L=1: FOR Z=CC TO A-1:Y=Z+1'HMHH
6102 FOR B=0 TO 11:A$(Z,B)=A$(Y,B): NEXT : NEXT :A=A-1:
     GOSUB 6000:L=0: IF A>1 THEN 600'NOJR
6104 GOTO 500'BDEE
8000 T$="9000": PRINT "[CLR,DOWN2]""CDCD
8001 PRINT "PUT BLANK TAPE INTO CASSETTE PLAYER.":
     GOSUB 30: PRINT "[CLR,DOWN2]""DENO
8002 ED=0: OPEN 1,1,1,"SORTED": FOR X=1 TO A-1:Q$=A$(X,0):
     IF Q$=>T$ THEN 8004'LJJS
8003 T$=Q$'BEIE
8004 NEXT :T=VAL (T$): PRINT "[CLR,DOWN2,SPACE5,RVS]
     SORTING.[OFF]":AA=0'FLYO
8005 FOR X=1 TO AA: FOR B=0 TO 11:J$(X,B)="": NEXT :
     NEXT 'JTAO
8006 AA=1:L=0:T$=STR$(T):W=LEN (T$):
     T$=RIGHT$(T$,W-1)'JDOS
8007 IF T=9001 THEN T=0: PRINT#1,"STOP": CLOSE 1:A=1:
     GOTO 6000'IUMR
8008 PRINT " "T$: FOR X=1 TO A-1:W=LEN (A$(X,0)):
     IF RIGHT$(A$(X,0),1)=CHR$(46) THEN W=W-1'OMBA
8009 Q$=LEFT$(A$(X,0),W)'CNTN
8010 IF Q$<>T$ THEN 8012'EICF
8011 FOR B=0 TO 11:J$(AA,B)=A$(X,B):A$(X,B)="": NEXT :Q$="":
     AA=AA+1:L=1'KQWR
8012 NEXT :T=T+1: IF L=0 THEN PRINT "[UP2]": GOTO 8006'IMQL
8014 Q1=AA: IF AA=2 THEN PRINT "[UP,SPACE4,UP]": PRINT Q1;:
     PRINT J$(1,0)" "J$(1,1)" "J$(1,2): GOTO 8050'IOWT
8018 LL=0: FOR I=1 TO Q1-1'FJXO
8020 PRINT "[UP,SPACE37,UP]""BARI
8022 PRINT Q1;: PRINT J$(I,0)" "J$(I,1)" "J$(I,2)'CABK
8024 IF J$(I,0)+J$(I,2)+J$(I,1)<=J$(Q1,0)+J$(Q1,2)+J$(Q1,1)
     THEN 8030'IYTV
8026 LL=1: FOR B=0 TO 11:B$(B)=J$(Q1,B):J$(Q1,B)=J$(I,B):
     J$(I,B)=B$(B): NEXT'ICCY
8030 NEXT : IF LL<>0 THEN 8036'FIJH
8032 FOR X=1 TO AA-1:Y=X+1: IF J$(X,0)+J$(X,2)+J$(X,1)>
     J$(Y,0)+J$(Y,2)+J$(Y,1) THEN LL=1'OFXB
8034 NEXT : IF LL=0 THEN 8050'EICL
8036 Q1=Q1-1: IF Q1>2 THEN 8018' FN BP
8050 ED=ED+AA-1'DHDI
8051 FOR X=1 TO AA: FOR B=0 TO 11'GJXL
8054 PRINT#1,J$(X,B): NEXT : NEXT 'DLQM
8055 IF ED=>A-1 THEN T=9000'GJKQ
8056 GOTO 8005'BEKM
9000 T=90: PRINT "[CLR,DOWN2]""CEWD
9001 PRINT "PUT BLANK TAPE INTO CASSETTE PLAYER.":
     GOSUB 30: PRINT "[CLR,DOWN2]""DENP
9002 OPEN 1,1,1,"SORTED": FOR X=1 TO A-1:Q$=A$(X,2):
     IF LEFT$(Q$,1)=> CHR$(T) THEN 9004'MKEU
9003 T=ASC (Q$)'CFUG
9004 NEXT :T=T-1: PRINT "[CLR,DOWN2,SPACE5,RVS]SORTING.
     [OFF]":AA=0'FJKO
9005 FOR X=1 TO AA: FOR B=0 TO 11:J$(X,B)="": NEXT :
     NEXT 'JTAP
9006 AA=1:L=0:T=T+1: IF T=58 THEN T=65'IROQ
9007 B$=CHR$(T): IF T=91 THEN T=0: PRINT#1,"STOP": CLOSE 1:
     A=1: GOTO 6000'KYBV
9008 PRINT " "B$'BCFK
9009 FOR X=1 TO A-1:Q$=A$(X,2)'FOXQ
9010 IF LEFT$(Q$,1)<>B$ THEN 9012'FMDH
9011 FOR B=0 TO 11:J$(AA,B)=A$(X,B):A$(X,B)="": NEXT :Q$="":
     AA=AA+1:L=1'KQWS
9012 NEXT : IF L=0 THEN PRINT "[UP2]": GOTO 9006'GIGK
9014 Q1=AA: IF AA=2 THEN PRINT "[UP,SPACE4,UP]": PRINT Q1;:
     PRINT J$(1,1)" "J$(1,2): GOTO 9050'IHPT
9018 LL=0: FOR I=1 TO Q1-1'FJXP
9020 PRINT "[UP,SPACE37,UP]""BARJ
9022 PRINT Q1;: PRINT J$(I,1)" "J$(I,2)'CSUK
9024 IF J$(I,2)+J$(I,1)<=J$(Q1,2)+J$(Q1,1) THEN 9030'GJPR
9026 LL=1: FOR B=0 TO 11:B$(B)=J$(Q1,B):J$(Q1,B)=J$(I,B):
     J$(I,B)=B$(B): NEXT'ICCA
9030 NEXT : IF LL<>0 THEN 9036'FIKI
9032 FOR X=1 TO AA-1:Y=X+1: IF J$(X,2)+J$(X,1)>J$(Y,2)+J$(Y,1)
     THEN LL=1'MQUW
9034 NEXT : IF LL=0 THEN 9050'EIDM
9036 Q1=Q1-1: IF Q1>2 THEN 9018' FN CQ
9050 FOR X=1 TO AA: FOR B=0 TO 11'GJXL
9054 PRINT#1,J$(X,B): NEXT : NEXT : GOTO 9005'EQFP
```

□

SWAP ROUTINE

Tony Atkinson

Here is a tiny but powerful machine language routine which implements a SWAP function found in some larger Basics.

The routine has not been implemented as a wedge. All wedges involve some slowing of the machine and was not considered worthwhile in this case. If you put S equal to the starting address of the routine SYSS calls the function and there is no saving by using SWAP.

The syntax is as follows:

SYSS,A$,B$ SYSS,A%,B% SYSS,A,B
SYSS,X(2,3),X(4,5)

The commas are critical.

Probably SWAP finds most common use in various sort routines but has many uses, for example in maths and statistics. An interesting case, if you store screens in two dimensional arrays, Swap can be very effective. As listed below, the routine will Swap all variables; floating point, integer and string. It will also handle elements of floating point arrays of any dimension. With a very slight modification, indeed one value, it will also swap string and integer arrays of any dimension.

Before discussing the working of the routine let's have a look at how variables and arrays are held in memory.

All variables are held in seven consecutive bytes in the section of RAM reserved for variables immediately above the Basic program.

In each case the first 2 bytes hold the variable's name. You only need to use one character for a name but provision is made for two.

The remaining 5 bytes hold the details of the particular variable.

Floating point variables use 5 bytes to hold floating point values. The last 5 bytes are taken up with this 5 byte configuration of the floating point value; so all 7 bytes are used for F.P. variables.

Integer values are held in two bytes in high byte/low byte order. Thus only 4 bytes are used by integer variables; two for name, two for value. But they are still stored in 7 bytes; the last three always set to 0.

Strings are quite different. The actual characters which make up the string may be held in a number of other places in memory. If,for example, they are contained in a program line or data line, they remain there unless changed. Any new or changed strings, for example if entered as a direct command or through the operation of your program, are stored in the section for strings reserved at the top of RAM.

Thus in the case of string variables, after the first two name bytes, byte 3 holds the number of actual characters in the string. Bytes 4 and 5 in usual low byte/high byte order, hold the address where the actual characters of the string are held. In other words by looking at these bytes the machine knows how many characters it needs to pick up and the address of the first of these characters. All 7 bytes are still used for each string variable with the last 2 always set to 0.

If we can find the address of the third byte of any named variable we can swap the last 5 bytes with the last 5 bytes of another variable of the SAME type.

This can be done quickly and easily using a Basic routine from the ROM.

In the case of strings no new or changed strings are created by Swap. These can clutter up the string section of RAM. There will be no call for garbage collection. This can be very important in a large program or sort.

In the routine as listed below all 5 bytes after the two name bytes are swapped to make it generally applicable to all three types of variables. Although you can see we only need change 2 bytes for integer variables and 3 for strings.

Now let's look at arrays. All arrays are held in that section of memory set aside for arrays between variables and strings areas.

Each array starts off with a header. The first two bytes hold the array name, then such things as number of bytes, number of dimensions, etc., in other words a discription of the total array. Our interest is in the details of a particular element in the array. In the case of numeric arrays the values of each element are in a row of bytes immediately after the header information. In a similar way to variables, floating point values need 5 bytes while 2 bytes are used for integer values.

In the case of string arrays the actual characters of the string are held elsewhere in memory just as they are with string variables. So after the header we have a collection of 3 bytes for each element of the array. The first holds the number of characters while the next two hold the address where those characters start in memory. Again just like string variables.

A little problem arises in that there are no blank bytes in arrays as in variable headers. For floating point arrays, we need to swap 5 bytes for an element and the program below therefore works with these as well as with the three types of variables.

However, for integer array elements we only need to swap 2 bytes and for string arrays 3 bytes need to be swapped. If we move more bytes, unlike variables, we will be moving parts of another element; can be a bit messy!

This simply needs a change in the counter for the loop; the value in the Y Register. One change, maybe a Poke, will provide this powerful little function which can then work with any sort of variable or array. It would be possible to have the program itself check the type of variable or array and alter the counter accordingly. I do not consider the added complexity and length are worthwhile when used as a Basic function in your own Basic program. Further-more as it stands, the routine fits comfortably into a first line REM. Personally I believe this is the best place to hold these routines if they fit and do not introduce any bugs with control characters.

BASIC LOADER

Listing 1 is a Basic loader program which puts the Swap routine in the cassette buffer. The routine can be relocated anywhere you wish without any change.

LISTING 1

```
10 N=828'BELX
20 READ D:IF D=-1THEN END'GELC
30 POKE N,D:N=N+1: GOTO 20'EKUD
40 DATA 32,253,174,32,139,176,132,252,133,
   251,32,253,174,32'BBMK
50 DATA 139,176,132,254,133,253,160,4,177,
   251,72,177,253,145'BCDL
60 DATA 251,104,145,253,136,16,243,96,-1
   'BHUI
```

ML ROUTINE

Listing 2 is a commented listing of the machine language routine itself. This program depends on the Basic ROM routine located at address 45195. The machine uses this routine to find these parameters and it returns in the accumulator and Y register, in low byte/high byte fashion, the address discussed above for each variable or array element.

This is done twice, after checking commas, for each variable or element and the addresses stored in 251/254. These are the free bytes on Zero Page which allow us to use the very valuable post-index, indirect address mode for the swapping of the information in the little loop.

You know when swapping two variables it is necessary to do something like C=A, A=B, B=C. In other words, we use a temporary variable You know when swapping two variables it is necessary to do something like C=A, A=B, B=C. In other words, we use a temporary variable like C or a temporary address to store A while you change A to B, then B to the first A value. In this program the stack is used for the temporary storage.

Summarising the change in the loop counter value, ie. Y in address 849 in the above program:

Y=4 swaps – floating point, integer and string variables and floating point array elements.

Y=2 swaps – integer and string variables and string arrays. This is a very useful combinations.

Y=1 swaps – integer array elements.

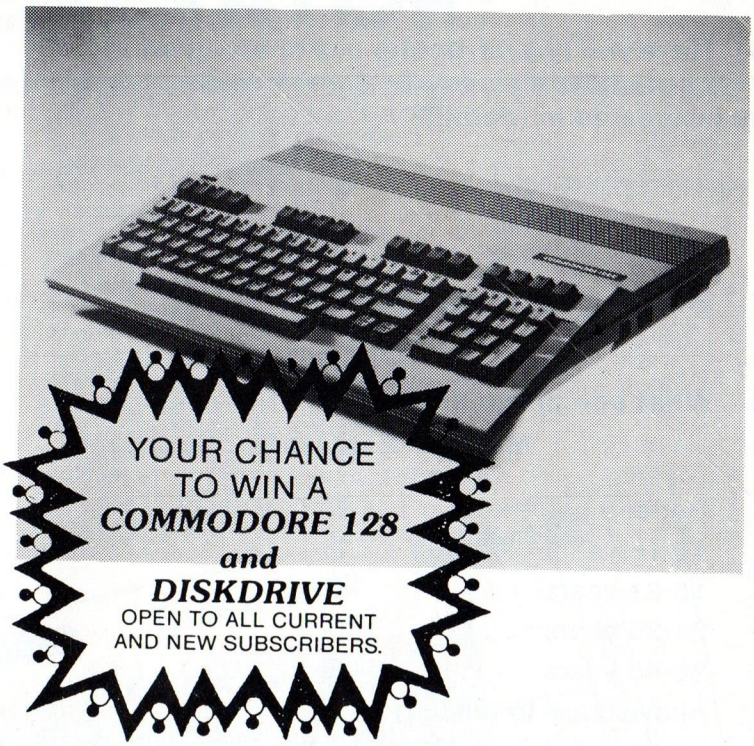
FOR SWAP LISTING SEE PAGE 33

SUBSCRIBER SURVEY

Since KIM BOOKS took over the publication of the COMMODORE MAGAZINE in June 1984, the magazine has grown and gained respect as a quality publication. To make sure that we continue to develop in accordance to your wishes we have decided to carry out a survey of subscribers.

The survey will span the next three issues. Those subscribers (with Australian Postal Address) who send in completed survey forms which include their name and postcode will be eligible for to enter our competition the prize for which will be a Commodore 128 computer and a diskdrive.

A maximum of three entries per subscriber is permitted. One for each individual survey form completed. (Entrants must be current subscribers).



1. What computer equipment do you own/use (including non/Commodore)

Computer (model):.....
Disk Drive (model):.....
Cassette:Monitor (type/model):.....
Printer (model/type):.....
Plotter (model/type):.....
Interface (model/type):.....
Modem (model/baud):.....
Wordprocessor Software:.....
Others items (e.g. Simon's BASIC, Digitizer, Joystick etc.):
.....
.....
.....

2. What are your main interests in computing? (Please grade from 1 major to 5 passing)

	1	2	3	4	5
Education					
Arcade style games					
Adventure games					
Graphics					
Sound					
Spreadsheets					
Wordprocessing					
Business					

Other (specify).....
.....
.....

3. What Computer Languages do you use? (Please grade 1 - 5 as above)

	1	2	3	4	5
BASIC					
BASIC advanced					
LOGO					
FORTH					
COMAL					
PASCAL					
COBOL					
FORTRAN					
PILOT					
MACRO ASSEMBLER					

Other (specify).....
.....
.....

4. Are you a member of a Commodore or Microcomputer User Group. (Please Specify.)

.....
.....

5. Are you a subscriber or member of any telephone database? (Please Specify).

VIATEL:.....
TELEDATA:.....
MINERVA:.....
SOURCE:.....
BBS (specify):.....
.....
.....

CONTINUED OVERLEAF

CONTINUED FROM PREVIOUS PAGE

6. Have you any particular use or application for your computer that you feel we'd readers would be interested in (detail)?

.....

.....

.....

7. What is your occupation?

.....

8. What age group are you in?

Male ☐, Female ☐ please tick

Under 9 years	☐
10-15 years	☐
16-21 years	☐
21-35 years	☐
36-45 years	☐

Above (care to tell us?) .....

9. What other interests are you involved in?

(i.e. Sailing, theatre, electronics travel, war games etc.)

.....

.....

.....

10. How long have you been a reader of The Commodore Magazine and how did you first come in contact with it?

.....

11. What other publications do you read? (Computer and non-computer orientated):

	Regularly	Occasionaly	Rarely
Australian Personal Computer:	.....		
Your Computer:	.....		
Australian Commodore Review:	.....		
PC Games:	.....		
AHOY:	.....		
Zapp:	.....		
Power Play:	.....		
Compute:	.....		
The Transactor:	.....		
TV News:	.....		
TIME:	.....		
New Idea:	.....		
Others (specify)	.....		

.....

.....

12. Do you hire Videos and how frequently?

(e.g. once a week, once a fortnight etc.):

.....

Next issue we will take a look at your likes, dislikes and suggestions for the magazine.

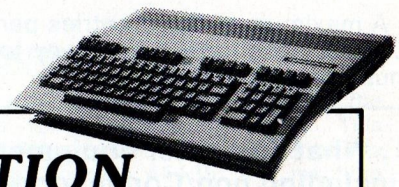
Subscriber code: .....

or

Name: .....

Postcode: .....

(Make sure the postcode and name is the same as that on your subscriber address)



COMPETITION

In a maximum of 200 words tell the Editor what program you would like to write for any Commodore Home Computer. Give sufficient information to determine what is entered into the computer, where it is obtained, what is the computer output and the benefits the output will achieve. The program can be in any area: entertainment, educational, business, utility, social etc.

This competition is for both beginners and experienced computerists. You do not have to write the program or actually know how to. Each entry will be judged on its own merits based on general feasability, originality and presentation.

The Editors' decision is final and no correspondence will be entered into.

CLOSING DATE:

Final Closing date for the competition will be announced in Issue 35 of The Commodore Magazine and the winner will be announced in Issue 36.

NOTE: To be eligible for the competition this survey sheet, accompanied by your entry, must reach us by February 14th, 1986. All survey questions are optional.

Send to:

READER SURVEY
KIM BOOKS
82 Alexander Street
Crows Nest
NSW 2065

Information obtained from this survey will be utilized for statistical purposes only and will not be linked or retained in association with personal subscriber files.

LOTTO GENERATOR

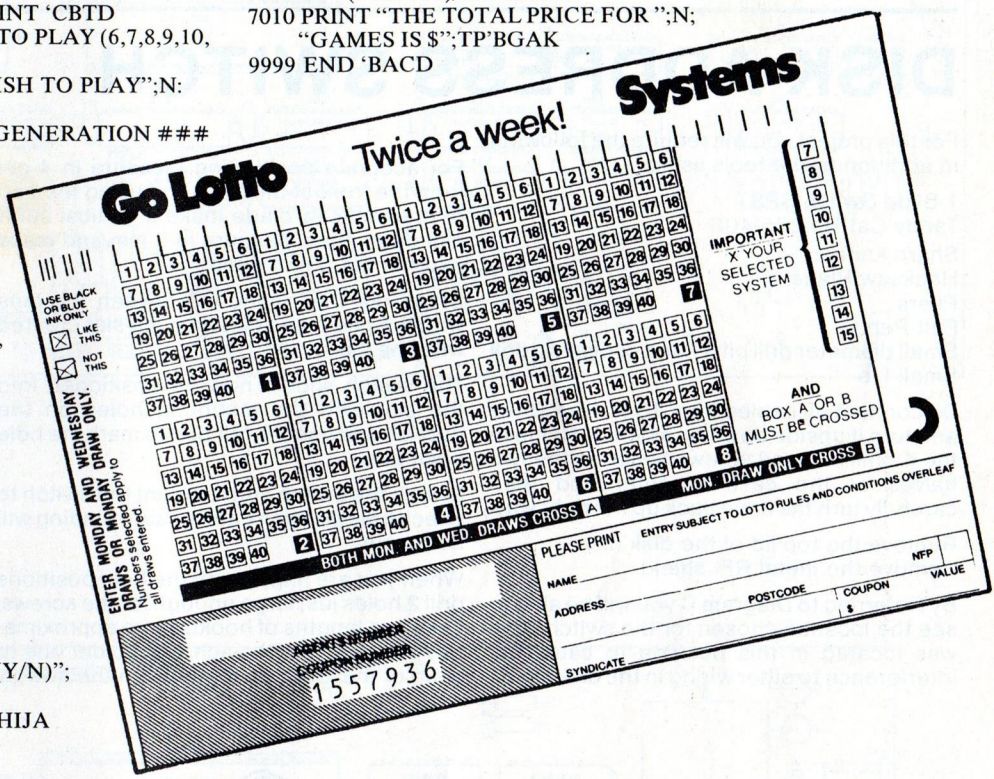
by Bill Shields

Written for the Commodore 64 this program can be easily adapted to other machines. Amend line 1000 which sets colour and clears the screen.

The LOTTO GENERATOR lets you choose system and number of games, Monday or Wednesday draw and calculates the entry cost (NSW LOTTO).

LOTTO GENERATOR

```
10 REM ### INITILIZE ###'BPSB
100 POKE 53280,0: POKE 53281,0: PRINT "[CLR]"'DQOA
110 PRINT "[<BLK>,SPACE7]LOTTO NUMBER GENERATOR"
'BAFE
120 PRINT "[SPACE10]BY BILL SHIELDS": PRINT 'CBTD
122 PRINT "WHAT SYSTEMS DO YOU WANT TO PLAY (6,7,8,9,10,
    11,12,13,14,15)": INPUT N1'CDLP
125 INPUT "HOW MANY GAMES DO YOU WISH TO PLAY";N:
    IF N<6 OR N>20 THEN 125'GLLR
127 REM ### SET UP ARRAY FOR NUMBER GENERATION ###
    'BKNO
128 N1=INT (N1):N=INT (N)'ELRK
130 DIM A(16,21)'BIOA
140 FOR X=1 TO N'DDXB
150 FOR Y=1 TO N1'DEXC
160 A(Y,X)=INT ( RND (1)*39)+1'FOLH
170 NEXT Y'BBSC
180 NEXT X'BBRD
182 PRINT "[DOWN5,RIGHT12]THINKING....."
    'BACK
190 FOR X=1 TO N'DDXG
200 FOR Y=1 TO N1'DEXX
210 FOR Y1=1 TO N1'DFRA
220 IF Y=Y1 THEN 260'DGTB
230 IF A(Y,X)=A(Y1,X) THEN 400'DQJE
260 NEXT Y1'B CMD
270 NEXT Y'BBSD
280 NEXT X'BBRE
399 GOTO 1000'BEXQ
400 A(Y,X)=INT ( RND (1)°39)+1'FOLE
410 GOTO 200'BDBA
1000 INPUT "DO YOU WANT A HARD COPY (Y/N)";
     H$'BDRB
1010 IF H$<>"Y" AND H$<>"N" THEN 1000'HIJA
1020 IF H$="Y" THEN 2000'DGHX
1030 FOR X=1 TO N'DDXX
1035 PRINT "THE NUMBERS FOR GAME NO. ";X;"ARE"'BDPJ
1040 FOR Y=1 TO N1'DEXY
1050 PRINT A(Y,X);'BHQA
1060 NEXT 'BAEY
1070 PRINT 'BACA
1080 NEXT 'BAEB
1299 GOTO 3000'BEAO
2000 OPEN 4,4'BDAU
2002 FOR X=1 TO N'DDXX
2010 PRINT#4, CHR$(14)"THE NUMBERS FOR GAME NO. ";X;
     "ARE:"'CJNF
2020 FOR Y=1 TO N1'DEXX
2030 PRINT#4,A(Y,X);'BJLY
2035 NEXT 'BAEC
2040 PRINT#4'BBDX
2050 NEXT 'BAEY
3000 PRINT "DO YOU WANT TO PLAY[SPACE,
     RVS]M[OFF]ONDAY OR MON/[RVS]W[OFF]EN'S DRAW":
     INPUT M$'CDMJ
3003 IF M$<>"M" AND M$<>"W" THEN 3000'HISE
3005 IF M$="W" THEN 5000'DGND
3010 ON N1-5 GOTO 3100,3200,3300,3400,3500,3600,3700,3800,3900,
     4000'DCIH
3100 P=P+25: GOTO 7000'DJLY
3200 P=P+180: GOTO 7000'DKFB
3300 P=P+720: GOTO 7000'DKFC
3400 P=P+2130: GOTO 7000'DLAD
3500 P=P+5290: GOTO 7000'DLKE
3600 P=P+11640: GOTO 7000'DMEF
3700 P=P+23290: GOTO 7000'DMIG
3800 P=P+46390: GOTO 7000'DMOH
3900 P=P+75465: GOTO 7000'DMTI
4000 P=P+250740: GOTO 7000'DNIA
5000 ON N1-5 GOTO 5100,5200,5300,5400,5500,5600,5700,5800,5900,
     5000'DCCI
5100 P=P+50: GOTO 7000'DJJB
5200 P=P+355: GOTO 7000'DKJD
5300 P=P+1420: GOTO 7000'DLBE
5400 P=P+4230: GOTO 7000'DLDF
5500 P=P+10540: GOTO 7000'DMCG
5600 P=P+23180: GOTO 7000'DMGH
5700 P=P+46390: GOTO 7000'DMOI
5800 P=P+86090: GOTO 7000'DMPJ
5900 P=P+150540: GOTO 7000'DNFK
7000 TP=(P*N+15)/100'ELME
7010 PRINT "THE TOTAL PRICE FOR ";N;
     "GAMES IS $";TP'BGAK
9999 END 'BACD
```



SWAP CONTINUED FROM PAGE 30

```
828  32 253 174   JSR 44797
831  32 139 176   JSR 45195
834 132 252       STY 252
836 133 251       STA 251
838  32 253 174   JSR 44797
841  32 139 176   JSR 45195
844 132 254       STY 254
846 133 253       STA 253
848 160   4       LDY 4
850 177 251       LDA (251),Y
852  72           PHA
853 177 253       LDA (253),Y
855 145 251       STA (251),Y
857 104           PLA
858 145 253       STA (253),Y
860 136           DEY
861  16 243       BPL 850
863  96           RTS
```

828 BASIC routine checks for comma
831 BASIC routine to find variable address
834 Store address in this and next line
836 low byte in 251
838 Check comma and do it again for next
841 variable
848 ###This is the Y to be changed### Held
850 in address 849
852 Stack used as temporary storage
861 Branch until Y becomes negative.

TWO SWITCH PROJECTS

Vince Morton

I run the following configuration: C64, 1541 and SKAI 64 Disk Drives, with an Xectec interface running into a BMC BX 1000 Printer.

I have found that the Skai 64 with its fast format can cause problems with disks being read on the 1541. For some reason the 1541 does not always respond to disks formatted on the Skai 64. This led to a problem for me, as I often use the disks in different configurations.

The Skai 64 was easy to alter the address on, as it has external address switches, however with the 1541 I was left with either changing the address by software, which can be a pain when you are in the middle of a program, or continually opening the lid of the 1541 to cut and solder links. The end result of this was that I fitted a slide switch into the side of the 1541 which connects to the two ends of the link and allows me to change the address of the 1541 from the outside to device 8 or 9.

TOOLS LIST

Apart from the parts mentioned for each project you will need to obtain the following items.

- Soldering Iron with a fine tip
- Resin cored solder
- Roll of insulation tape
- Cutters & Phillips head screwdriver
- Hook up wire

DISK ADDRESS SWITCH

For this project you will require the following in addition to the tools listed.

1 Slide Switch SPST
Tandy Cat No 275-406
Sharp knife
Hacksaw Blade
Pliers
Felt Pen
Small diameter drill bit & hand or electric drill
Small file

Disconnect all cables from 1541 Disk Drive and turn it upside down on a bench. Remove the 4 phillips head screws securing the two halves of the case together and then carefully turn the drive back up the right way.

Remove the top lid of the disk drive.
Remove the metal RFI shield.

By Referring to Diagram C you will be able to see the location chosen for the switch, (this was located in this position to cause no interference to other wiring in the disk drive.)

For accurate positioning measure in 4 cm from the front of the case and using the end of the hacksaw blade make a vertical cut 8 mm deep, then measure in 8 mm and make another cut 8 mm deep.

Break out the plastic in between the cuts with the pliers and clean up the slot created with the file.

The switch should now be positioned into the slot with the mounting holes on the outside and use the felt pen to mark the hole positions.

Switch the switch and remount the switch to check hole positions and ensure binding will not occur.

When you are happy with the hole positions drill 2 holes just large enough for the screws. Take two lengths of hookup wire approximately 25 cm in length each, and solder one to each of the 2 terminal points on the switch.

Insulate the terminals with insulation tape, and then mount the switch into the case with the two screws provided, ensuring the wires are connected to the end of the switch nearest the front of the drive.

Locate link E1 from either Diagram D or E depending on your drive model. You will notice that E1 is shaped as 2 semi-circles joined by a thin bridge of track.

Using a sharp knife carefully cut out the thin bridge of track. Now solder one wire to each of the semi-circles of E1, ensuring that the wires touch only those semi-circles.

During the installation of the switch some debris will have inevitably found its way into the drive casing and must be removed. This can be achieved by placing the disk on its side (select switch down) and using a 10 mm paint brush, brush out all the debris.

The drive may now be reassembled and the switch checked for binding. Take care not to damage your switch wires during assembly.

If you have followed the instructions correctly then changing your drive address is as simple as the flick of a switch.

With the switch in the forward position the drive is Device 8 and with the switch back the drive is Device 9.

A couple of Dymo labels will finish off the job nicely.

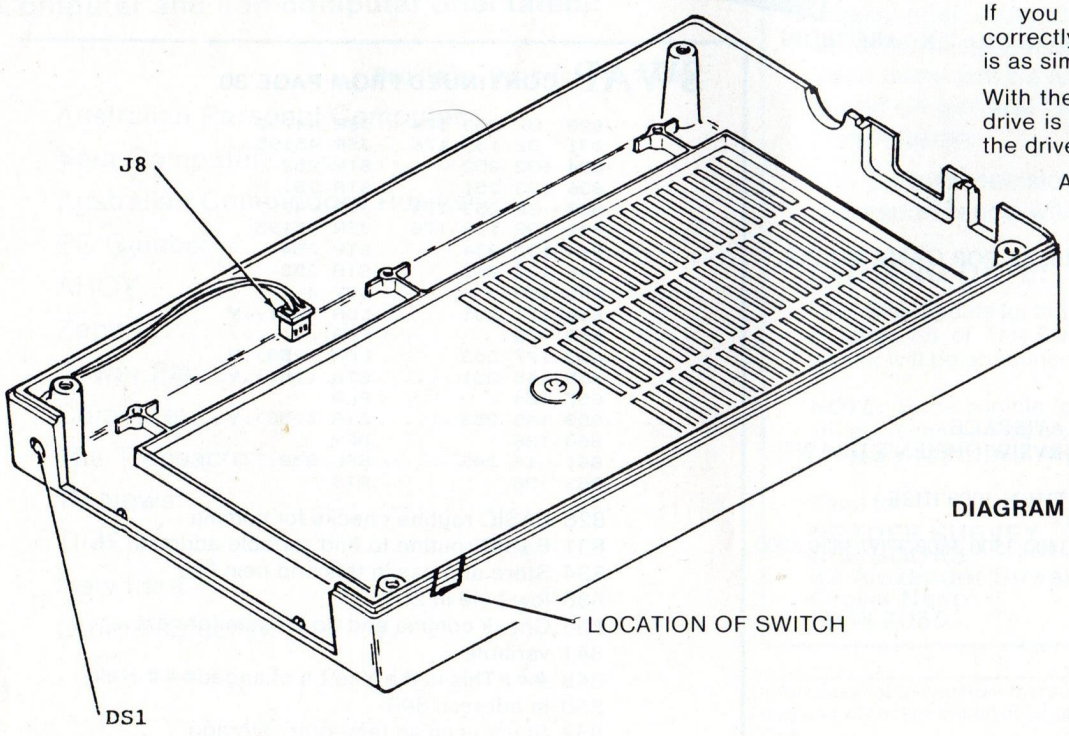


DIAGRAM C

DIAGRAM D

OLD STYLE 1540 or 1541

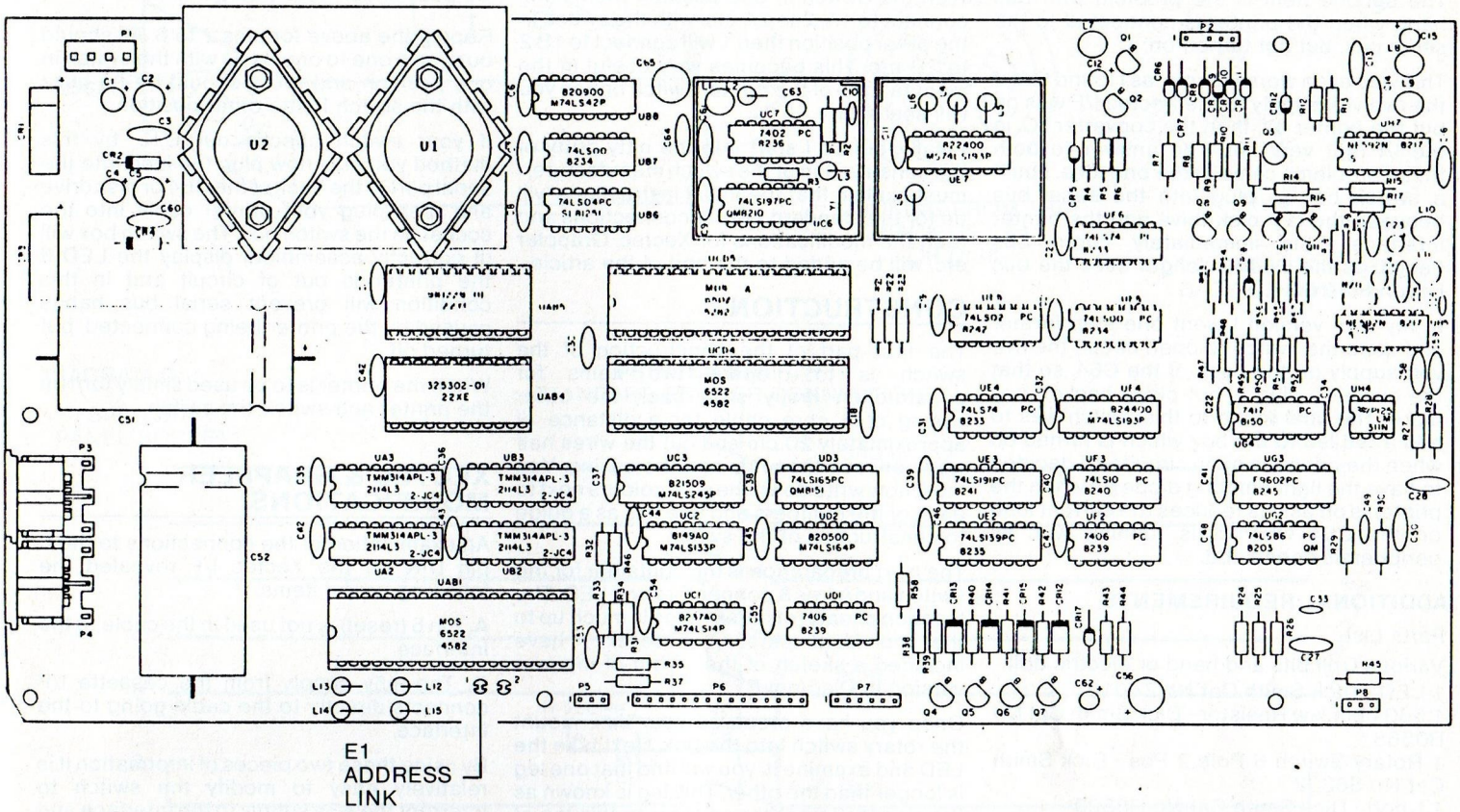
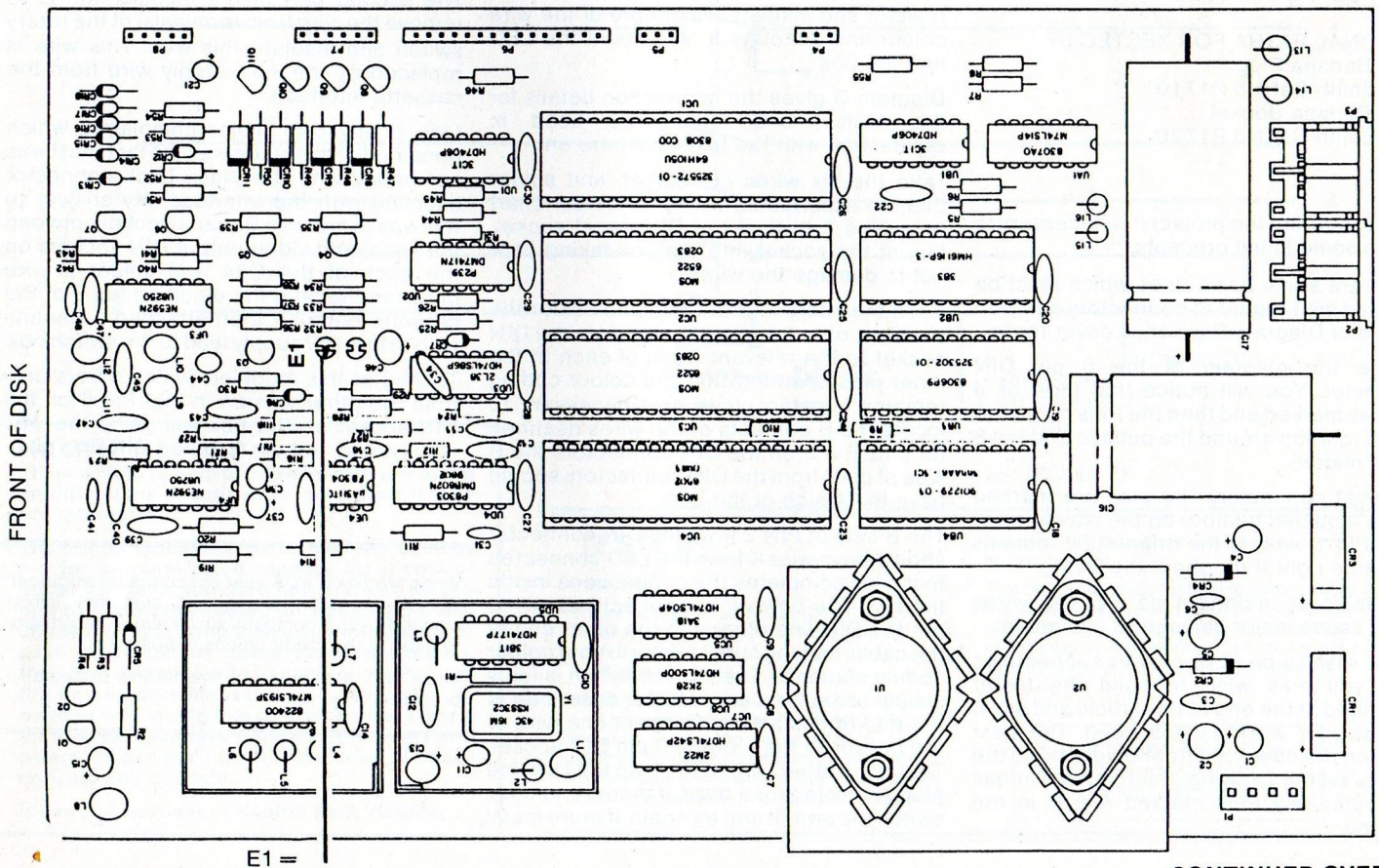


DIAGRAM E

NEW STYLE 1541



CONTINUED OVERLEAF

PRINTER SWITCH

The second item is the problem with bus hang when the printer is connected to the serial bus, but not turned on.

This was even worse for me, as I found that if the five volt supply to the Xectec I/F was on but the printer off, then the converter I.C. in the I/F got very hot. The answer to both these problems came in the one idea. I built a switch box to plug into the serial bus between the last disk drive and the printer interface. This immediately cured bus hangs, as the C64 no longer sees the bus beyond the drive.

In my own version I went one step further and used the switch to open circuit the five volt supply on the back of the C64, so that the cassette plug is not piggy backed any more. The final touch to this switch was to add a red led to the box which is turned on when the printer is open circuited, I decided to have the light emitting diode on when the printer is off as this reduces the current load on the C64 when it is running with all peripherals connected.

ADDITIONAL REQUIREMENTS

Parts List

Various Drill bits and hand or electric drill
1 LED - Dick Smith Cat No Z4010
1 560 ohm ¼w Resistor - Dick Smith Cat No R0568
1 Rotary Switch 6 Pole 2 Pos - Dick Smith Cat No S6302
1 Knob - Dick Smith Cat No H3800
1 Box - Dick Smith Cat No H2753
1 6 Pin DIN Line Plug - Tandy Cat No 274-9608
1 6 Pin DIN Panel Socket
Tandy Cat No 274-9618
1 Meter 3 Pair Cable
Tandy Cat No 278-9528

OPTIONAL EXTRA FOR XECTEC I/F
4mm Banana Plug
Dick Smith Cat No P1710
4mm Banana Socket
Dick Smith Cat No P1720

Prior to starting the project it is necessary to set up some initial precepts.

There are some basic rules which must be followed with regard to connectors and with the aid of Diagram F we shall cover these.

Locate the diagram of the 6 pin DIN connector. You will notice that there is a Keyway marked and then the Pins number 1 to 5 in rotation around the outside and Pin 6 in the middle.

On most connectors the pins are marked with a moulded number on the plastic on at least Pin 1, so that the orientation remains the same right throughout the wiring.

This is most important as crossed wires could cause major damage to equipment.

If you wish to be sure of your connections then you may wish to build the tester described at the end of this article and use it to test your work as described. The next thing on the check list from Diagram F is the Rotary switch Drawing. Each Inner Pin has two outer row Pins marked A & B in the drawing.

With the Switch in one position then 1 will connect to 1A, 2 to 2A etc. with the switch in the other position then 1 will connect to 1B 2 to 2B etc. This becomes very useful in the manufacture of the printer switch box as you will see later.

Finally, before I start into the nitty gritty of the construction of the switch manufacture, I must explain that the initial instructions will be for the standard printer connections and then the modifications for Xectec, Grappler etc. will be added to the end of the article.

CONSTRUCTION

The first part of the construction of the switch is to prepare two items for construction firstly strip back the outer casing of 6 core cable for a distance of approximately 20 cm and cut the wires half way between the ends and the casing. You must now write down the wire colours next to each of the numbers and use this as a guide to construction of the switch.

The next preparation is the container for the switch and wires & connectors and LED. The location of each of these items is much up to the individual person, however I have included a sketch of the layout of my own version in Diagram F.

Once you have prepared your box, mount the rotary switch into the box. Next take the LED and examine it, you will find that one leg is longer than the other. This leg is known as the anode. Solder the resistor to this leg of the LED and insulate with tape. Take two different coloured pieces of hookup wire, and solder one to the other leg of the LED and insulate taking note of the wire colour and annotate it "K".

Solder the other wire to the bare end of the resistor and insulate. Take note of the wire colour and annotate it "A". Mount the LED into the box.

Diagram G gives the connection details for the switch, and should be used in connection with the text from here on.

Take the six wires cut earlier, and solder them in accordance with your colour chart onto pins 1 to 6 of the DIN panel socket. Mount the socket into the box taking care not to damage the wires.

Feed the cable into the box and solder the six wires onto the inner 6 pins from the DIN socket to the relevant A pin of each of the inner pins ensuring that the colour coding remains constant. It is now necessary to locate the B contacts of the wires destined for pins 6 & 2 on the DIN connectors the B side of pin 6 from the DIN connectors should have the A side of the LED.

The B side of PIN 2 from the DIN connector should have wire K from the LED connected to it. This completes the connections inside the box. The only other connections are to put the DIN line plug onto the other end of the cable, making sure to keep to your colour coding standard. Testing the switch is fairly simple using the buzzer tester described at the end of this article. Connect one wire of the buzzer to Pin 1 of the 6 pin DIN socket. Touch the other wire to Pin 1 on the DIN line plug and listen for a buzz, if there is no buzz switch the switch and try again. If there is still no buzz check your wiring.

Repeat the above for pins 2 to 6. All should buzz on a one to one basis with the switch in one position and there should be no buzz with the switch in the other position.

If your switch checks out O.K., by this method you may now plug the lead into the serial port at the back of the C64 or disk drive and then plug your printer cable into the socket on the switch box. The switch box will (if correctly assembled) display the LED if the printer is out of circuit and in this condition will prevent serial bus hangs caused by the printer being connected, but turned off.

When the printer is to be used simply turn on the printer and switch the switch.

XECTEC & GRAPPLER MODIFICATIONS

An examination of the connections to the 6 pin DIN on the Xectec I/F revealed the following useful items.

A. Pin 6 (reset) is not used in the cable to the interface.

B. The +5v supply from the cassette I/F connects directly to the cable going to the interface.

By using these two pieces of information it is relatively easy to modify the switch to interrupt the +5v supply to the interface and thus protect it from overheating.

The first part of the modification is to separate the 2 x +5v wires in the interface DIN plug and connect the wire from the interface to Pin 6 of the DIN plug.

The second part of the modification is to remove the wire from terminal 6 of the rotary switch and insulate this wire. This wire is replaced by the +5v supply wire from the cassette interface.

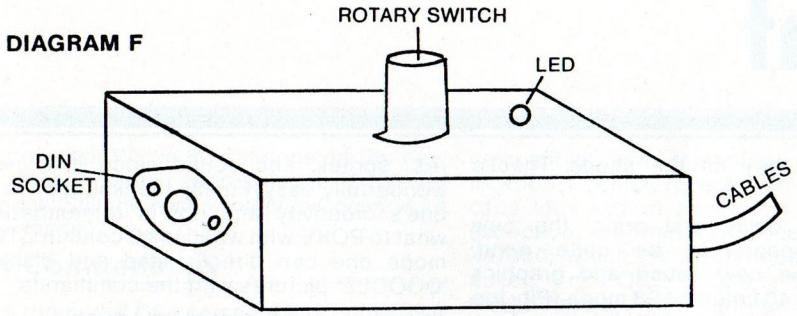
One of my own major complaints which remained at this stage was the fact that I was not happy with the piggy back connector supplied with the interface. My answer to this was to mount a Banana Socket between the serial and video output DIN sockets on the back of the C64 and solder a wire between that and the diode on leg 2 of the cassette output. I then attached a Banana plug to the +5v supply lead to the switch box.

Testing of this modified switch differs only from the other version in that Pin 6 on the DIN socket on the box will go to the +5v supply lead, not pin 6 on the DIN line plug.

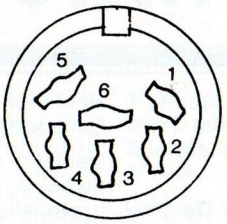
Vince Morton is a 35 year old computer engineer. He is currently training and customer liason manager with a company which supplies software to most of the credit unions in Australia.

Vince has a number of interesting projects in preparation for us.

DIAGRAM F

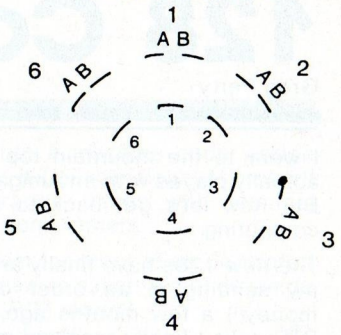


6 PIN DIN SOCKET



LOOKING FROM THE BACK OF C64 OR DISK

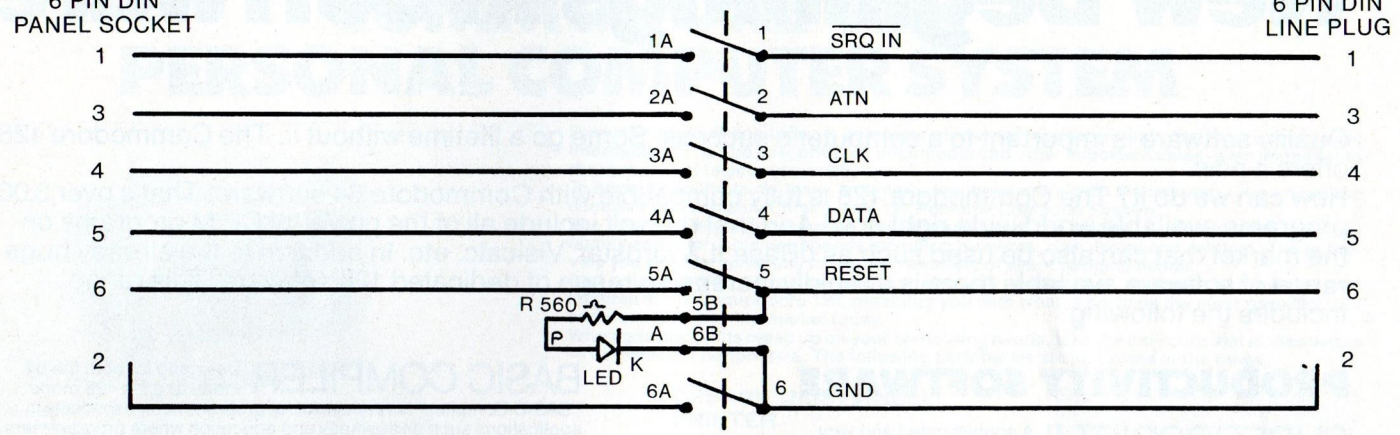
ROTARY SWITCH



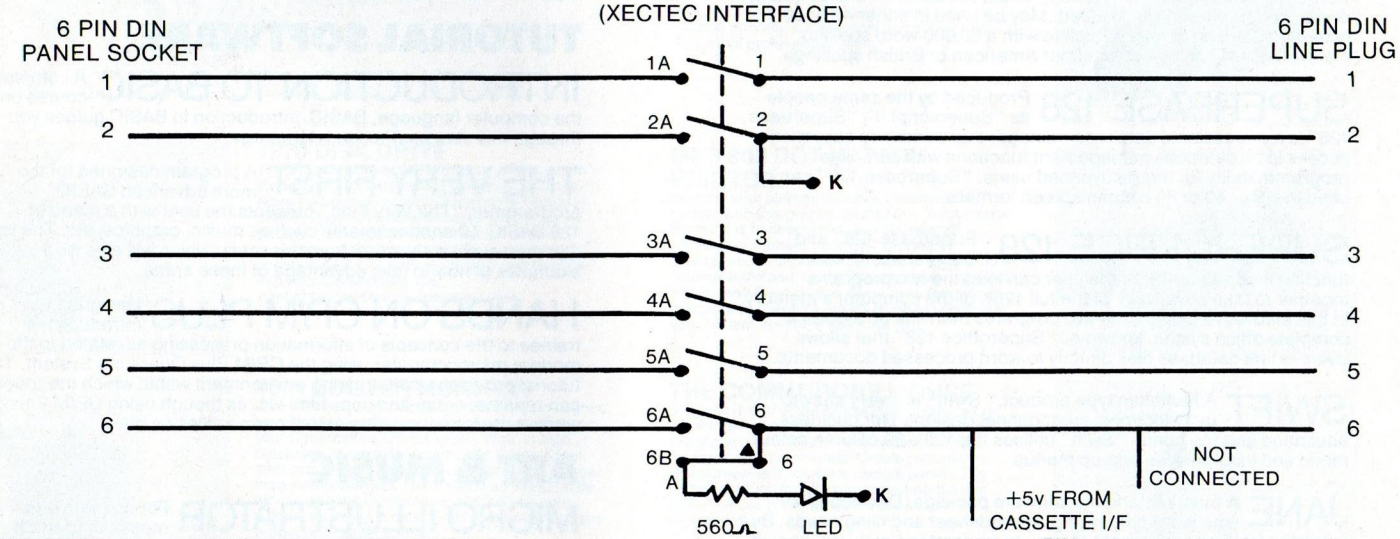
LOOKING FROM THE BACK OF SWITCH

DIAGRAM G

PRINTER SWITCH
(STANDARD PRINTER CONNECTION)



PRINTER SWITCH
(XECTEC INTERFACE)



HELP IF NEEDED

"I am willing to help any reader with any of the modifications described in my article if they are interested.

The printer switchbox can be manufactured by me for any interested readers for a cost of about $30 for a standard unit and $32 for the one to run on the Xectec plus postage. If any one is willing to bring their equipment to me to have modifications noted above performed, then they are quite free to contact me through the Commodore magazine.
Anyone who would like to attack their system themselves can have details of the mods and where to get the bits for the cost of photo copying and postage."

Vince is a resident of Kealba Park Victoria.

TESTER

Parts List

1 mini buzzer – Tandy Cat No 273-053
1 battery clip – Tandy Cat No 270-401
1 AA battery
2 lengths of wire

Solder together the red wire from Buzzer and the red wire from the battery clip and insulate. Solder the two lengths of wire to each of the black wires from the buzzer and battery clip and insulate the connections. Insert the battery into the clip. Touching together the two wires will now cause the buzzer to buzz.

This buzzer is handy for testing cables, (as is the switch box) fuses (a good fuse will cause a buzz) and many other items of low resistance. Never attempt to test an item with the buzzer, with power connected.

Commodore Magazine
82 Alexander Street
Crows Nest N.S.W.2065

128 Comment

Greg Perry

I went to the mountain top last month, I actually played with an Amiga for two days! But now let's get back to some serious computing.

The new 128s have finally arrived! Despite my sending in an order complete with money(!) a few months ago, after several STD calls I finally received my system two weeks after they hit the shops. Thanks Commodore!

Despite the delay and price, the new machine appears to be quite good, especially the new sound and graphics commands in 40 column 128 mode. (Pity the PLAY command is out of key!) I expect we will actually see many more users using high res., sprites, and sound since it is now wonderfully easy. Finally the limitation is in one's creativity and not in remembering what to POKE with what! In 40 column C128 mode one can directly load and display 'DOODLE' pictures with the commands

BLOAD "name",B0,P7168:GRAPHIC 1

Another pleasure is in being able to use

Superscript WP in 80 columns on the new monitor. (I haven't abandoned my 8032 yet though. Still the best workhorse Commodore has made.)

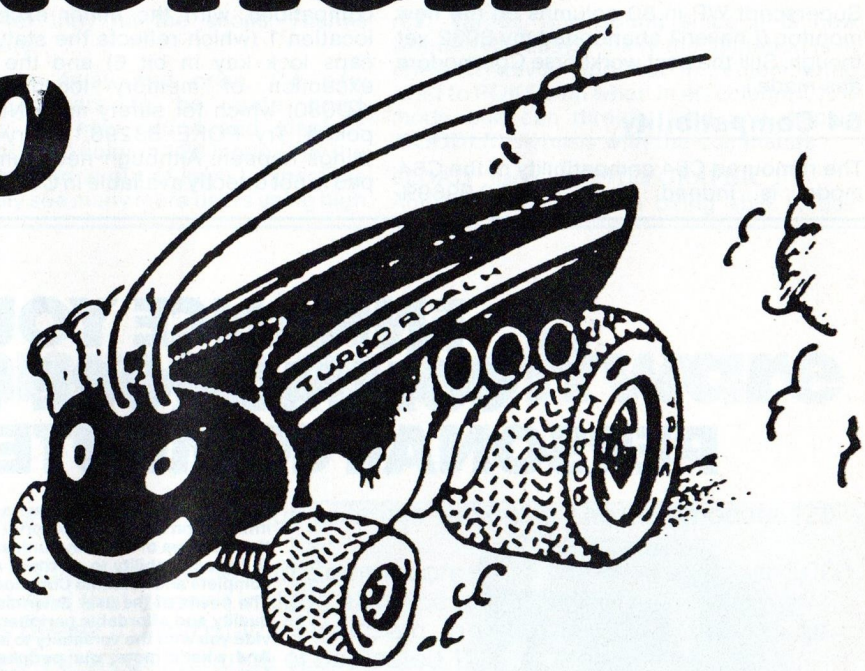
64 Compatibility

The rumoured C64 compatibility of the C64 mode is indeed true (it's 99.9969% compatible), with the minor exception of location 1 (which reflects the status of the caps lock key in bit 6) and the MAJOR exception of memory location 53296 ($D030) which for safety must NEVER be poked. Try POKE 53296,1. Very strange things happen. Although new number key pad is not directly available in C64 mode it is possible to write a machine code wedge which activates it. Leave it alone! At present it may do strange things with terminal programs and some others.

One big problem for most of us dedicated Commodore users is exactly what do we do with the new CP/M mode?
['A> do something now!'] □

A sample of the promotional material Commodore is producing to support the release of 128.

COCKROACH TURBO ROM



A Review

Greg Perry

While investigating the variations of the C64 Kernal ROMs for an article in the Commodore Magazine (Vol 5 No 4), one of our editors, Paul Blair, made a number of suggestions that some of the less used Kernal routines could usefully be replaced with improved disk access commands.

Unlike many other good ideas, this did not end up in the rubbish bin but was picked up by Ralph Down and Stu Burrows from Cockroach Software on the Gold Coast. (Probably from an old magazine lying on the beach!) With the experience gained from their 'Turbo-Roach' products, and by replacing all the cassette and RS232 routines with several pieces of tightly written machine code, they have managed to produce a very worthwile product for the Commodore 64 disk user which takes a dramatic step in bringing the 1541 into the computer age.

In short, Cockroach Software have produced a plug-in replacement for the standard Kernal ROM of the C64 which increases the speed of the SAVE, LOAD, and VERIFY commands by a factor of roughly five. They have also found space to include a number of extra useful commands. The ROM can even be custom made to display your name and choice of screen and border colours on power-up.

As a bonus, the new 8K Turbo-ROM is actually supplied in a 16K package with an attached switch, providing the user with the best of both worlds - at the flick of the switch the user can select either the Turbo-ROM or the normal Kernal ROM. (Unfortuntely, it is often necessary to turn the C64 off before switching ROMs.) The package appears to perform very well with most software except, of course, communications programs, such as TERM64, because the RS232 routines have been removed. Such programs must be loaded and run with the standard Kernal or loaded in Turbo mode then the ROMs switched before resetting the C64.

The new chip is mounted in an adaptor socket which replaces the socket normally occupied by the 8K Kernal ROM. In most cases it is simply a matter of carefully removing the old ROM from its socket and inserting the Turbo-ROM. Unfortunately, about 20% of C64's have soldered-in ROMs. In these cases it is necessary to unsolder the existing Kernal chip and the solder in the 24 pin replacement socket supplied with the kit. Since it is quite easy to damage the printed circuit board of C64, this task should be entrusted to a competent technician. (Or see your local user group.)

The Turbo-ROM is compatible with the C64 and 1541 drive and a version is available for the SX 64. (This must be specified when ordering.) The authors say that the new ROM is NOT compatible with drives other than the 1541. I have not had the chance to try the ROM with the SKAI SUPER drive but there would appear to be a good chance of success.

The main features of the Turbo-ROM are:

- All SAVE, LOAD, and VERIFY commands now default to disk. (There is no need to add the ',8' after these commands.)
- Five times faster LOAD.
- Five times faster VERIFY.
- Five times faster SAVE.
- Optional fast FORMAT (30 seconds).
- Improved 'SAVE @' command. (Old program is actually scratched first.)
- Screen dump to printer – even during program execution – C= F7.
- Built-in enhanced DOS wedge using the '@' command.
- Three extra commands have also been added. These are:
  - 'ZAP' <RETUR> – same as SYS 64738 (cold start).
  - 'OLD' <RETURN> – restores BASIC program after NEW or RESET.
  - 'MON' <RETURN> – enter a machine code monitor

The built-in DOS wedge is very welcome, removing one of the main problems encountered with C64 disk access. For example, entering '@' on its own displays the disk status, while entering '@$' displays the disk directory on the screen. (Does not affect program memory.) The '@' command can also be used to send any other direct disk access commands such as SCRATCH, RENAME, VALIDATE, NEW etc.

Even though the disk access defaults to drive 8, the Turbo-ROM will support 2 drives as devices 8 and 9. The command '@9' has been incorporated to allow drive 8 to be changed to drive 9.

As well as a single key low res screen dump via C= f7 (even works during program execution), the '@P' command provides the equivalent of OPEN4,4:CMD4. That is, it directs all printed output to the printer for program LISTings and the like. ('@O' returns output to the screen.)

A number of other interesting features make for easier program loading.

First, entering '*' <RETURN> loads first program on disk (similar to LOAD"*",8,1). Second, pressing SHIFT RUN/STOP prints 'LOAD' [RIGHT18] ',8,1' and leaves the cursor over the second comma which allows the user to change the comma to a colon ':' before pressing RETURN. This last command greatly simplifies program loading. For example, display a directory with the '@$'

command, move the cursor up to the line displaying the required program, press SHIFT/RUN/STOP, then press RETURN to load the program. Couldn't be simpler.

Unlike some fast loading and saving routines, the Turbo-ROM works with other devices such as printers on the serial bus. It is also compatible with the VIC SWITCH network used in many schools. With such networks, not all eight computers need be fitted with the ROM but only those which are will load faster. (Turbo-ROM will not work with the MULTI-LINK network.)

Most importantly, the Turbo-ROM does not affect sequential or relative file access and is safe to use with data base programs. (Unlike some other packages which have been known to corrupt disks with relative files.) The Turbo-ROM does, however, affect the C64's jiffy clock and hence the TI and TI$ commands.

The 'MON' command is very useful for machine language programmers. This command looks for MONAD (Paul Blair's machine language monitor) or DRVMON64 (a machine language monitor by Mike Henry) at $8000 then $C000 and, if present, enters the monitor. The fast disk access works with most ML monitors, although fast SAVE does not work with MONAD.

Using Turbo-ROM

Because the Turbo-ROM is simply a modified Kernal and therefore requires very few 'hooks' or patches into the C64, just about all the C64's memory is still freely available for programs. The cartridge port is also still available for monitor cartridges, such as HESMON, and the like.

The overall success of this approach is shown by the fact that Cockroach Software appear to have been successful in making their ROM load most commercial programs. There are, however, a small number of commercial programs which still refuse to load in Turbo mode. Of course, turn the computer off, switch in the standard Kernal ROM, turn the computer back on, and such programs can then be loaded normally.

In most cases, loading times of commercial software are significantly improved with the Turbo-ROM. However, fast loading will not work with some commercial software which use their own 'slow loading' routines using direct disk access commands.

Even many of the more recent commercial software which have fast loading built-in, for example, SUMMER GAMES II, TURBO-64 disks, ISEPIC, and DISECTOR, will load normally using the Turbo-ROM.

Just how much speed improvement is there with using the Turbo-ROM? The figures below give a good indication. Figures for the popular, but considerably more expensive, 1541 FLASH are also included by way of comparison. I have not included times for the EPYX FAST LOAD Cartridge since it does not support a fast SAVE routine. (The Turbo-ROM is slighlty faster on loading than EPYX.)

PROGRAM	Loading Time in Seconds		
	NORMAL	1541 FLASH	Turbo–ROM
PITSTOP II	144.99	FAILED	29.78 (4.87)
COLOSSUS CHESS	126.24	43.91 (2.87)	29.29 (4.31)
DOODLE	87.92	33.44 (2.63)	24.85 (3.54)
GHOST BUSTERS	185.67	93.94 (1.98)	76.97 (2.41)
PFS FILE	129.50	47.57 (2.72)	29.28 (4.42)
EASY SCRIPT	63.49	31.30 (2.03)	54.91 (1.16)

(The figures in parentheses show the speed improvement factor with respect to standard times.)

The following table shows the typical speed enhancement on SAVE and LOAD operations a user may expect for everyday programs.

Disk Blocks	Time for operation in seconds. Normal	1541 FLASH	Turbo-ROM
25 (SAVE)	19.64	14.20 (1.38)	6.90 (2.85)
25 (LOAD)	17.40	5.74 (3.03)	3.82 (4.55)
50 (SAVE)	36.83	26.03 (1.41)	9.10 (4.05)
50 (LOAD)	33.15	10.30 (3.22)	6.30 (5.26)
100 (SAVE)	70.40	49.99 (1.41)	13.60 (5.18)
100 (LOAD)	65.44	19.35 (3.24)	11.13 (5.88)
200 (SAVE)	137.16	97.94 (1.40)	22.57 (6.08)
200 (LOAD)	127.32	38.77 (3.28)	21.20 (6.01)

Notice that the speed improvement becomes greater as the program length increases. One other interesting point; because the Turbo-ROM's fast SAVE routine chains the disk sectors in a different manner (although completely compatible with normal operation), programs SAVEd with the Turbo-ROM will also LOAD approximately 8-10% faster with that ROM.

A Minor Problem: Disk Errors

After using the Turbo-ROM for a month I have found it to be a very useful addition to my C64, with no major problems.

However, a minor problem arises if a disk error is encountered when using the fast LOAD or SAVE routines. With the fast LOAD, the drive may go into limbo and the C64 goes to sleep out of boredom. Actually, when an error is encountered, the drive attempts to re-read the offending sector, and retries, and retries forever. (But with no horrible noises!). During this time the C64 keeps waiting for the rest of the program which never comes. Unfortunately, since the screen blanks when both LOADing or SAVEing, and since the disk just keep spinning, there is no direct indication that a problem has occurred. (One piece of irrelevant information – while loading a program, it is actually possible to open the disk door, take out the disk, replace it and close the door, and the program will continue to load correctly! However, this is not recommended.)

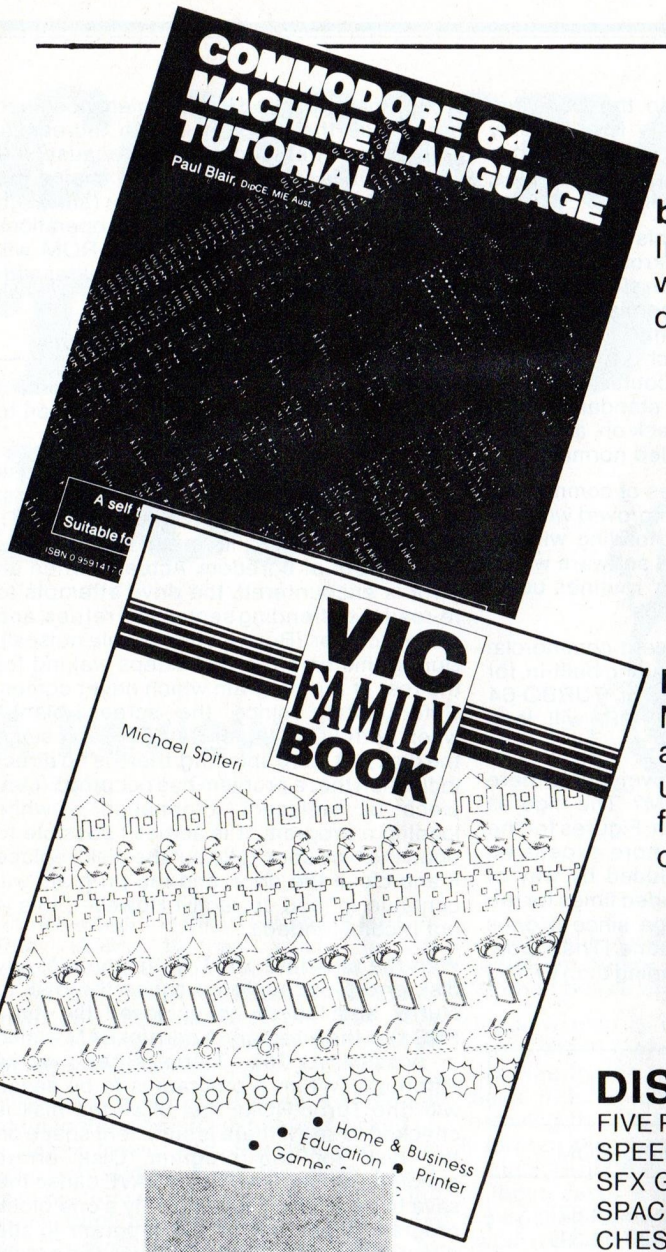
With the standard 1541 SAVE routine, any disk errors encountered cause the disk to stutter and leaves an unclosed file (type PRG*) in the directory. A corrupt BAM is also a possibility. The Turbo SAVE works differently. When a program is to be saved with the Turbo-ROM, first of all the disk is checked to see if there is sufficient space on the disk for the program. Disk errors encountered with the Turbo SAVE cause the save to be aborted, leaving only a one block program allocated for the program in the directory and the BAM. However, the actual program length may be longer.

If this occurs, under no circumstances should the further programs be saved to the offending disk before the errors are rectified or the disk reformatted. (Which only takes 30 seconds using the fast format). Until this is done, it will generally be impossible to save other programs successfully to this disk because the same error will be found again.

Overall

In summary, I commend Cockroach Software on a fine product. One of the features that particularly appeals to me is tha fact that it performs its job unobtrusively and the extra commands are both easy to understand and use. The price, by the way is $40.00 per ROM (including 24 pin adaptor socket) or $45.00 if a customised message and colours are required. Suggested fitting cost is an extra $5.00, either from Cockroach directly or via your local user group.

NEXT ISSUE:
Epyx Fastload vs Turbo Rom
– A Comparison Test/Review



COMMODORE 64 MACHINE LANGUAGE TUTORIAL

by Paul Blair

Introduction to Machine Language complete with listing that illustrates and will quickly demonstrate many of the points made in the text.

CLEARANCE SALE

$10.00

Accompanying Disk $10.00

INCLUDES MONAD

THE VIC FAMILY BOOK

by Michael Spiteri

Michael Spiteri's first book is a collection of short and effective programs for the VIC. Programs to use in the home, work and play. A special section for adventure gamers and super expander cartridge owners.

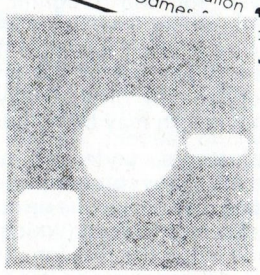
CLEARANCE SALE

$9.50

DISK OFFER # 5

FIVE PROGRAMS FOR THE 802 (1526) PRINTER - Issue 32
SPEECH PROGRAM (64 & VIC) - Issue 32
SFX GENERATORS 1, 2 & 3 - Issue 32/33
SPACEBATTLE - Issue 32
CHESSPLAYER - Issue 33
64/GEMINI/CARDCO - Issue 32
LOTTO GENERATOR - Issue 33
TEXT COMPACTION - Issue 33
ADDRESS BOOK - Issue 33
Plus Others

Released Late January 1986



$10.00

***P&P included**

ORDER FORM: I/We wish to order the following items:

COMMODORE 64 ML TUTORIAL	$10.00		
VIC FAMILY BOOK	$9.50		
DISK OFFER # 5	$10.00		
P & P $1.50 Certified $2.50			

KiM BOOKS
(Division of Mervyn Beamish Graphics Pty Ltd)
82 Alexander Street,
Crows Nest, N.S.W. 2065
(02) 439 1827

Name ..... Address .....

..... Phone .....

☐ I enclose cheque for .....

☐ Please charge my BANKCARD/ VISA/ MASTERCARD .....

Expiry date ..... Signature .....

Adventure Help

ADVENTURE HELP SPECIAL

Michael Spiteri

Welcome to an extra-special Adventure Help.

This article was due to be printed months ago, but it seems the it was lost on a corrupted disk in some hidden cave! I managed to dig up an extra copy, after surviving many traps and getting lost several times. So here is the Treasure!

Some of the problems are a bit old, but they may cometo the aid of some desperate adventurers!

To solve your adventure problems, write to:

Adventure Help
Commodore Magazine
82 Alexander St
Crows Nest NSW 2065

OR contact me on VIATEL number 378697780

Ben Wright of Brisbane had trouble with cannibals in **Interceptor's Jewels of Babylon**. If you are stuck in the same spot then try giving the watch to the cannibals. Where is the watch you may ask. Well, that is another story.

A. Lister is one of those brave adventurers who is tackling Infocom's **ZORK I**. This chap sent me 5 problems which were bugging him. Here are the problems, three of them with solutions.

1) CANNOT GET INTO HADES: You'll need the bell, 2 candles, matches and prayer book. Ring bell, light candles and read a prayer. You should be able to enter Hades now.

2) CANNOT INFLATE BOAT: You need the pump to inflate the boat.

3) CANNOT OPEN GRATING: Grating can be opened from the underside.

4) CANNOT GET THROUGH WOODEN DOOR IN LOUNGE: Can anyone help?

5) CANNOT GET PLATINUM BAR: Echo Echo Echo.

Another **ZORK**er is Simon Dobner, he has difficulty getting into Hades. He also has difficulty getting into the coal mine. Can anyone help Simon?

Poor ol Marion Zacher. After getting 60% in Dotsoft's **ALIEN**, finding the UFO and fitting the disk, she doesn't know what to do next! Come one everyone, give her some help!

Scott Adams is up to his tricks again. David Cutling is stuck in **THE COUNT**. He cannot kill the stupid vampire! A popular problem. I was told you have to murder the dude during the day when he's asleep.

Scott Adams again! Andrew Gill cannot light the lamp in ADVENTURELAND, which is preventing him from entering the 'black hole'. It's been years since I played this game, but I think you need (matches. Use the command LIGHT ON.

And again!!! Kevin Ferguson is stuck in the dome in **THE HULK**. He cant get out without getting killed!! Try pushing the button, Kevin, then hitting your head. You should be able to leave safely now! Kevin is also stuck in **WARGAMES** – what do you do with the podule? He asks. I don't know! Does anyone?? And in **KING SOLOMON'S MINE** (Dotsoft), what is he supposed to do with the box & amulet! He has problems in **AXTEC TOMB** – how do you get up into the tree house!

Kevin did'nt just ask for tips, he gave an awful lot of them. I will give you a selection. Here are some of his tips:

BASTOW MANOR: To enter the house, make your way to the shed and move the case to reveal a trap door. If you become tired, drop the branch. Then light the torch, pick up the branch and go through the trap door. Use the branch to cross the pit.

WARGAMES: Remove grill from airduct with the screwdriver. Tie cable to tree and climb cable. Break the lock on the machinery room door.

AZTEC TOMB: Catch and examine the fish in the jar. Water the plant and climb the beanstalk. Frighten the elephant with the dead mouse.

Kevin has also got cheat sheets for Mountain Valley adventurers. Write to me for details. Thanks for an interesting letter, Kev!

BASTOW MANOR is troubling lots of people these days. Mr. M. Davis of NSW is having trouble getting the Brass Key from the closet without getting hit from behind. He's wearing the armour, but that doesn't seem to help.

An adventure that has been keeping me busy is Infocom's **HITCH HIKERS GUIDE TO THE GALAXY**. C. Bell of Bondi, NSW offers are solution for the Babel Fish problem. Here are the instructions to catch a fish:

1) Hang gown on hook.
2) Put satchel in front of the Robot Panel.
3) Cover the drain with the towel.
4) Put the mail on the satchel.
5) Push the button on the dispensing machine!

He has problems in this game. Does anyone know how to get the Ultrasonic Vacuum Aid back to the heart of gold? How do you get through the screen door?

I'd like to give a special thanks to Keith Campbell, the adventure expert from England's Computer and Video Games magazine. Many of the hints were supplied by Keith. Thanks Keith!

To finish off, here are some tips for some well know adventures:

DALLAS QUEST You'll need the ring, bugle and pouch before you board the plane. The chief is really JR in disguise. Don't give him the ring!!!

TAREK The Fungoid has two important items one you'll need to get into the central computer and the other to get the internal shuttle security code. Also if things start falling apart and you've become disoriented try to get back to your own quarters.

HEROES OF KARN A bird will get rid of a serpent as long as you let the frog hero do the work.

SPIDERMAN Have a good feel of the locations around Mysterio's location.

EUREKA! A lift will take you down, but make sure you can go up as well!

THE HOBBIT Throw barrel out of trap door and jump onto it!

That's all for now. I'll need more letters if you want another bumper help page!

□

THE OTHERS

David Roth

White Lightning: A professional development system

White Lightning is a FORTH-based arcade game development system from OASIS Software. In addition to an (almost) standard FIG FORTH, it has over 300 graphics and sound commands. Since such commands take up most of the execution time in a typical arcade game, their implementation in FORTH makes White Lightning comparable in speed to machine code. White Lightning FORTH is more readable than machine code and easier to debug since it has reasonable crash protection. I was able to warm start White Lightning most of the time after a system crash with the use of a 'reset' switch. White Lightning is not for beginners – it is aimed at people who want to write high quality games. It is also well suited for some scientific and mathematical applications, as I will demonstrate later. It could be a useful tool for undergraduate mathematicians or scientists seeking more speed than is available from BASIC or PASCAL.

A sprite generator is provided, together with a predefined arcade set. Up to 255 software sprites may be defined, depending on their size and the available memory. These sprites are in addition to the standard hardware sprites available on the CBM-64. A software sprite may be up to 6 screens wide (e.g. for use as a scrolling background), but such large sprites are costly in memory. FORTH words are provided which allow sprites to be rapidly inverted, rotated, reflected, etc. One of the best features of the package is the multi-tasking capability – running different programs in foreground and background at the same time. For example a background program (interrupt driven) can handle a scrolling backdrop, while a foreground program handles the objects which move against the backdrop (as with "Attack of the Mutant Camels"). This eliminates the need for complex calculations to get a smooth scroll.

Since new FORTH words may be defined in terms of existing ones, the White Lightning language is readily extendible. For instance, a diagonal scroll command may be set up using a combination of existing horizontal and vertical scrolls. There is also a BASIC interface which alows the use within FORTH of a limited number of BASIC commands. This interface is useful for text, but there is no easy way to access BASIC variables from FORTH. A complete BASIC package, BASIC Lightning is provided at no extra charge on the White Lightning disk. It has a similar range of commands to those available in White Lightning, in addition to a useful set of structured programming constructs (e.g. CASE, PROCEDURES, etc). This BASIC package may be used to quickly test out sprite designs and game ideas for later implementation in White Lightning. I understand that a compiler will shortly become available for BASIC Lightning.

An utility is provided which converts completed White Lightning programs into programs which can run independently of the White Lightning environment. Such programs may be marketed without any constraints or costs (although OASIS would like a plug !). The package is well documented and clear examples are given of the graphics and sound capabilities. A BASIC demo program is included on the distribution disk, but unfortunately no White Lightning demo program was provided. It seems a pity that no array handling words or string/numeric I/O routines were provided, since these functions are frequently required in many games (see previous issues of this magazine for examples of such routines). Mathematical functions such as sine or square root would also be useful (with appropriate scaling and use of double precision) for maths/science work.

The speed of White Lightning allows simulations of physical events involving many calculations to be displayed reasonably quickly – the movements of molecules, atomic decay, prey/predator interaction, etc. The following program provides a simple illustration of Brownian motion. This would be hopelessly slow in BASIC or even PASCAL, but the speed of FORTH can provide a crude 'motion picture' of moving molecules. The program demonstrates the use of White Lightning graphics commands, the structured programming construct CASE ... ENDCASE, and the generation of random numbers. The effects of compression on gases is also shown – when the run/stop key is pressed, the 'graphics area' of the screen is halved and the 'molecules' appear to move faster in the smaller space. When the program has compiled, 'GO' starts execution.

□

```
( simulation of brownian motion )
( redefine VARIABLE and CREATE to standard FORTH )
: variable here variable ; : create variable -2 allot ;
( mn = no molecules, an = array size )
50 constant mn mn 2 * constant an
( define arrays to hold x and y co-ordinates )
create x an allot create y an allot
( variables )
variable x1 variable y1 variable x1 variable y1 319 x1 ! 199 y1 !

( allocate random co-ordinates - within screen limits to each
molecule )
: setup mn 0 do x1 @ rnd x i 2 * + ! y1 @ rnd y i 2 * + ! loop ;
setup ( execute it )

( adjust co-ordinates according to the toss of a 4 sided die )
                                        : tossup 4 rnd
                case 1 of x1 @ 1 + x1 ! endof
       2 of x1 @ 1 - x1 ! endof
       3 of y1 @ 1 + y1 ! endof
       4 of y1 @ 1 - y1 ! endof
 endcase
 ( check if x or y are beyond screen limits - held in x1, y1 - if
so, reset them to screen limits )
x1 @ x1 @ > if x1 @ x1 ! endif x1 @ 0 < if 0 x1 ! endif
y1 @ y1 @ > if y1 @ y1 ! endif y1 @ 0 < if 0 y1 ! endif ;

( plot points on screen )
: points mn 0 do
  x i 2 * + @ col !
  y i 2 * + @ row !
  plot loop ;

( move each point according to TOSSUP )
: brownian                              mn 0 do
                        x i 2 * + @ x1 !
         y i 2 * + @ y1 !
         tossup
         x1 @ x i 2 * + !
         y1 @ y i 2 * + !
        loop ;

( mainline )
: go
  ( set up graphics screen - 'sprite 0' )
  ( white background, black background )
  white black 0 setatr
  ( hires graphics )
   s2col mono hires
  ( sprite 0 is_the_screen )
  0 spn ! sclr
  ( screen is white foreground, black border )
   white hpaper black hborder
  ( set 'plot' mode)
   3 mode
   begin  ( do forever )
      points ( plot points )
      sclr   ( clear them )
      brownian ( move them )
      ?terminal ( run/stop pressed ? )
      ( if pressed, halve x co-ordinate.          If x co-ordinate <
2, then reset it back to 319 )
      if x1 @ 2 / x1 ! x1 @ 10 < if 319           x1 ! endif setup
endif
   0 until ;
```

MORE SNIPPETS

Paul Blair

SUPERBASE and EASY SCRIPT have become (Commodore) standards. Having been around for a while now, it was logical that improved versions should appear – to include a wider range of features, and correct one or two minor irritations (ho hum) that users have noticed.

There is news of upgrades. We have not actually seen either of these upgrades, but overseas magazines are carrying details of what is happening, so we will bring the news to you.

EASY SCRIPT has been overtaken by SUPERSCRIPT (getting confusing, all these .....SCRIPT names!!). The advertising blurb indicates some new features...

Menu selection of commands or
Select command by its first letter
5 function calculator built in
Cut and paste blocks of numbers
Select printer type from a menu
Selective Mailmerge feature
Store commands on a key
Word wrap on/off

There are more variants, it seems. But the biggest change is that EASY SPELL has been built in to obviate the need to perform an individual/special load to check your spelling. That sounds nice.

Haven't seen it in the shops yet. Precision Software sell it for about 70 quid in the UK, with an upgrade (send your old disk back etc) for 40 quid. Given that there have been about a quarter of a million copies sold (and another 2 million shifted in mysterious ways) Precision could get rather busy.

One writer has given a list of "would be nice" items still missing. Here's the list:

True proportional spacing
Windows
Undo command
Spelling correction
Non-destructive block move
Sort column facility
Built in indexer
Telecommunicate features

Make up your own mind about these. But for what it is at what it costs, SUPERSCRIPT appears to be an incredibly powerful package.

If anyone has a copy, or any dealer is selling/trading up to SUPERSCRIPT, we'd be pleased to hear.

A tip for EASY SCRIPT users. We have previously noted a problem with dropped letters from the first line of new paragraphs when editing with EASY SPELL. One reader has suggested a simple fix.

Most of us leave a blank line between paragraphs. To do this, we press RETURN on a blank line and get a reversed left wedge ("<"). The fix is - don't. Press SHIFT and RETURN together, then go on typing.

This creates the blank (very!!) line, and it is reported that files then edit without leaving out characters.

After using EASY SPELL and keeping your text all intact, you can THEN go through and add the standard RETURNs to subdivide your text into paragraphs.

SUPERBASE V2 (is the package marked V3??) is on sale in Australia. The first price I saw was $199, which is now $149.

The changes that we know about are:

A new labels program
– can now use all field types
– can now use up to 4 items/line
– can now use forced spaces for formatting
– store label formats for later use
– full menu and help screen support
A new delete routine
A new command
– you can now export from a list
Better disk space management

A new utility program is provided. This may be an admission of past problems (as my mail tells me you too have the odd problem with SB).

The utility programs loads from a BASIC environment. Its purpose is to permit splitting a database and copy its uncorrupted information and file definitions onto a formatted disk. It converts V1 files to V2 at the same time.

It will also permit recovery of corrupted databases (hooray!!) files and records. It will not recover deleted records.

A copy feature is included for sequential files up to 112 blocks long.

We would welcome any further news/comments/praise/gripes/questions/Irish jokes about either packages.

 □

Beginners Corner: Poke Among This Lot

Peter Davies

The other day I was searching for a particular POKE address that I couldn't remember and while trying to find it among my collection of magazines I recorded all those that I found. They are listed below – if the list is useful to me it is probably of use to others too.

Warning – do not POKE around while your newly completed, unsaved masterpice is in memory as you may lose it completely. Also reset all adresses to their default values before loading a program or it may crash.

There are over 64,000 addresses that can be POKEd with values from 0 to 255. Many achieve nothing at all; those in this article represent a very small proportion of the remainder.

SYS 64738 will give a cold start (the just switched on state) but SYS 64767 will give a cold start without changing the screen colour. (While thinking about SYS's, SYS 65511 will close all open files.)

Everyone is probably familiar with POKE 53280,C for changing the border colour and POKE 53281,C for screen colour and the colours corresponding to the values of C (from 0 to 15). A less frequently used one is POKE 646,C for setting text colour.

Placing the cursor with cursor down's and right's can be replaced with the much more easily readable:

POKE 214,Y-1:PRINT:POKE 211,X or
POKE 211,X:POKE 214,Y:SYS 58732

Disabling one or more keys is often asked about in readers' letters; below are a few which are most commomly requested.

POKE 649,0 disables the keyboard (10 re-enables) but don't disable the keyboard in direct mode as you will not be able to type in the re- enabling poke !

POKE 657,128 disables the shifted Commodore key.

POKE 808,251 disables the RUN/STOP key; POKE 808,232 disables the warm start RUN/STOP and RESTORE combination.

POKE 775,168 will, among others, disable the LIST command while POKE 775,171 will cause a crash on executing LIST.

POKE 819,246 and POKE 818,32 both disable the SAVE command.

Has your HiRes masterpiece crashed? If so

POKE 53272,PEEK(53272) AND 247:POKE 53265,PEEK(53265) AND 223

will enable the error message to be read. You have to type this in blind so type slowly and carefully.

An invisible cursor can be perplexing; try POKE646,PEEK(53281).

This sets the text colour to the screen colour.

For programming, a number of useful pokes exist:

POKE 19,64 before an INPUT and the question mark is not displayed.

0 returns things to normal

POKE 199,1 is the same as CTRL RVS ON and POKE 199,0 CTRL RVS OFF, POKE 649,X sets the number of characters the keyboard buffer will hold; X must be =<10; put X=0 and see what happens when you try to use the keyboard

While on the keyboard, POKE 198,0 will empty the keyboard buffer prior to using a GET and POKE 631,13 will put RETURN into the buffer for dynamic programming techniques.

Other POKES of use are:

POKE 650,255 makes all keys repeat but POKE 650,127 and no keys repeat (not even the cursor keys)

POKE 56325,X where X can take values from 0 to 255 controls the cursor speed

POKE 56324,28:POKE 56325,0 slows the LIST speed as does POKE 56325,X where values of X less than 58 speed up the listing and those greater progressively slow it.

Finally, ever got a DEVICE NOT PRESENT error when you know it is ? The value at address 144 may have been changed.

POKE 144,0 before taking any drastic steps.

□

THE SUPERBASE PAGE

Paul Blair

Well, here goes again. There have been a few new developments since the last column, so let's look at the new news first.

By sending an original diskette of SUPERBASE V1 back to Precision Software (6 Park Terrace, Worcester Park, Surrey KT4 7JZ, England), I have been able to update to SUPERBASE V2.02. The changeover cost me ten pounds (about twenty dollars) plus some mailing charges.

Worthy of note was the speed of delivery. I posted off my request one Friday lunchtime, and had the return disk the following Monday week. From the UK!! Full marks to Precision, Speed Mailers of the Year. And a note of appreciation to the postal people, too, who usually get only complaints. The service was great.

The diskette is double sided- the front is your prime copy, the reverse is your backup copy. On the same diskette is SUPERBASE for the Plus 4 (remember them?), so its a bit of a bonus if you happen to own one of each computer, which probably won't be many of you.

I'll spend a little time with V2.02 before telling you more. At first sight, there do not seem to many differences (the manual additions supplied with the upgrade run to 6 typed pages). The principal claimed differences between versions 1 and 2 are these:

- a new labels program that permits multiple choices for field selection, use of forced spaces, multiple copies, and a method of storing label formats for later use.
- a delete routine for disk housekeeping. You get to this by using EXECUTE then typing DELETE. If you use this for a total purge, you must be careful to delete not only the records, but the file definition from the directory. If you don't, and decide to re-use the old file name, SUPERBASE will pick up the old FORMAT definition from the disk, and apply it to your new file.
- export from a list is now possible.
- disk space management has been improved. This is critical when your disk is nearly full. V2.02 takes steps to protect you from the trauma of a part-written record.
- a new utility program. This is accessed from Basic, and gives you the power (!) to split a database and copy its parameters onto a formatted disk. In fact, copy seems to be the main function of the utility. Oh, and you may (not always 'can') recover a corrupted database. That last joy should never be your lot, and, as I have noted before in these pages, I wonder why Precision should worry? It should never happen, should it.

That's a quick summary. In use, there seems to be no speed increase, but I do like the fact that the change is relatively low cost, and does offer better security for the SUPERBASE system. It's easy to be critical, but after using some of the more complex (and a multiple times more expensive) packages on a Different Brand, SUPERBASE is a very flexible system.

If you don't want to upgrade to V2.02, might I suggest you check which revision of Version 1 you are using. There were at least two versions sold in Australia – V1.0E and V1.0S. The version comes up on the prompt line at the top of each main menu. As you might expect, Version S is the more up-to-date and reliable of the two. If you have Version E, I suggest you try to get a copy of Version S (legally, of course).

Some snippets of news from Precision. They tell me there is a book about SB due out around now. That's all I know, but I will try to get hold of a copy, and pass on to you some information about price and availability. Maybe someone has seen it already and knows a bit more about it – and could write so we can all share the news.

Next. With the C128 not far away (this is late October, and still no sign...) Precision have not been idle. As they have done for the larger Commodore computers (the 8296 business computer springs to mind) they intend to package SUPERBASE and SUPERSCRIPT (the successor to Easy Script) into one co-resident package. The 8296 version is named SUPEROFFICE, and permits simple transfer of information between word processor and database. With the ability to define macros and transfer them between functions, plus record access in about one quarter of a second with the 1571 disk drive, the old bug-bear of slowly grinding through a FIND could be a thing of the past.

Cricket is a pleasant game, and I can 'Norm' it for hours. But it was cricket that led me to a problem with SUPERBASE that I had not come across before. If you're not ahead of me by now, I am referring to the Sloppy Sort routine built into SB.

Out in the northern part of Canberra, the cricket lovers are pretty dynamic - so dynamic that one bright young lad took it on himself to computerise (dreadful word, that!) the area membership records. Between overs, he set up a nice little file to records names, addresses and age groups for the junior club members. He did it very nicely, too. Each member was assigned a registration number, and the 'Under' something age group was included with each record.

The key field was assigned to 'Surname', so the system logically stored member details in alphabetical order. Because several members of some families were involved, duplicate keys were permitted. Remember that - it's important.

If you have paddled through the SB Manual, you will have come across page T-48, and its reference to sorting. The syntax is easy enough:

MENU 2/SORT (F4)/all on ( ) to "xxxx"

where the round brackets are square ones (it's our typesetter, not me) and enclose the field name that you want as the target of the sorting routine, and you want the resulting output written to "xxxx" for later use. So we set this all up, with "rego-no" as the item in square round brackets, and went off for a cup of coffee.

Eventually SB finished its job, the coffee pot was dry, so I set up a little program to show me how it had gone, using rego-no and surname to show me how nice it all was. It had gone, no mistake about that. But the result was not useable.

This is what I got:

```
001 Haslam
002 Misfeld
```

and so on to....

```
005 Caynoto
287 Mitchell
007 Schlesinger
048 Gee
009 Chamberlain....
```

That's only a short bit of it, but where did 287 and 048 come from? And where were 006 and 008?

Some little while later I found them. Guess what? Rego-no 006 had the surname "Mitchell", and 008 was "Gee". They were genuine records, as were 287 and 048. But what went wrong?

The clue lies on page R-55 of the manual, where it says "Do not specify the key as a field to sort on, as it is automatically included at the end of the list". Nice touch, you think? Maybe. But I don't think so.

The explanation is rather long, so I'll precis it down a bit. It has to do with permission for duplicate keys, and the order in which SB stores records that have duplicate keys. It has to do with the order of entry. If you enter a record for JOE BLOGGS tonight, and one for WILF BLOGGS tomorrow night, Wilf will be stored AHEAD of Joe. FIND will work OK, and you may locate both records. But SORT is different, and will retain the details of Wilf as the current record during the sort, and ignore Joe till later.

Well, I think this is something of a drawback. But unless I rewrite SB, I have to learn to live with this fact. What can I do? Refuse to accept duplicate keys? Good idea – not always practical, but the most simple way of getting integrity in the sorting process. Any other ways?

Hugh de Glanville, who edits the SB page in the English ICPUG magazine, has some suggestions, and I bring them to you with his blessing.

The first technique is not to permit duplicate keys, but give the impression that you are. Instead of a field for "rego-no" and "surname", append the rego-no to the surname. If Wilf was 123 and Joe was 456, the key field would be BLOGGS123 for Wilf, and BLOGGS456 for Joe. Then you could write a program like this:

```
10 select from "xxxx":eol menu
20 k$=(surname):k=len(k$)-3
30 n$=left$(k$,k)
40 display n$
50 goto 10
```

TWO Programs from David Balean

tape auto run

This program is a tape auto-run. It is designed for BASIC programs, but it could easily be modified for machine code if so desired (if anyone does, please tell us and we'll pass it on).

Instructions are at the start of the program.

```
100 REM: TAPE AUTO-RUN'BNCY
110 REM: FOR C64 BASIC PROGRAMS'BUQC
120 REM: D M BALEAN 5/85'BNFA
130 REM:'BBAX
140 PRINTCHR$(147):PRINT:PRINT" *****
    C-64 TAPE AUTO-RUN *****"'EHWK
150 PRINT"   FOR BASIC PROGRAMS ONLY"'BADI
160 CLR:FORA=49152TO49318:READB:POKEA,B:T=T+B:NEXT
    'JXLL
170 IFT<>20358THENPRINT"DATA ERROR":END'GHMK
180 PRINT:PRINT"TO USE :-":PRINT'DCYH
190 PRINT" WHEN YOU HAVE RUN THIS PROGRAM,"'BAEN
200 PRINT" LOAD OR TYPE IN THE PROGRAM TO BE MADE"
    'BAUG
210 PRINT" SELF-RUNNING WHEN LOADED FROM TAPE."
    'BAEH
220 PRINT:PRINT" IN THE DIRECT MODE TYPE :-"'CBPG
230 PRINT"   SYS49152"CHR$(34)"FILENAME"CHR$(34)"
    [RVS]RETURN[OFF]"'DIEK
240 PRINT" (THIS SAVES THE PROGRAM TO TAPE)"'BASJ
250 PRINT:PRINT" ON RELOADING, THE SCREEN IS CLEARED,"
    'CBCM
260 PRINT" AND RUN/STOP/RESTORE IS DISABLED."'BAAL
270 PRINT" THE JIFFY CLOCK FUNCTIONS NORMALLY"'BAQN
280 PRINT" AND THE PROGRAM AUTO-STARTS":'BBAM
290 :'ABHF
300 DATA 165, 43,133,251,165, 44,133,252'BEDD
310 DATA 165, 45,133,253,165, 46,133,254'BELE
320 DATA 169,189,133, 43,141, 2, 3,169'BBUE
330 DATA  2,133, 44,141, 3, 3,169, 4'BWIE
340 DATA 133, 45,169, 3,133, 46,160, 66'BBQG
350 DATA 185,100,192,153,189, 2,136, 16'BDOH
360 DATA 247, 32,212,225,169, 3,133,185'BDNI
370 DATA 32, 89,225,165,251,133, 43,165'BDSJ
380 DATA 252,133, 44,165,253,133, 45,165'BEHL
390 DATA 254,133, 46,169,131,141, 2, 3'BBFL
400 DATA 169,164,141, 3, 3,169, 1,170'BALD
410 DATA 168, 32,186,255,169, 0, 32,189'BCEE
420 DATA 255, 76, 89,225,169,131,141, 2'BCXF
430 DATA  3,169,164,141, 3, 3,169,147'BARG
440 DATA 32,210,255,169,230,141, 41, 3'BCXH
450 DATA 169,142,141, 40, 3,169, 0,133'BBEI
460 DATA 157, 32,213,255,169, 1,170,168'BDRJ
470 DATA 32,186,255,169, 0,170,168, 32'BCTK
480 DATA 189,255, 32,213,255,134, 45,132'BEJM
490 DATA 46,134, 47,132, 48,134, 49,132'BCRM
500 DATA 50, 32, 89,166, 76,174,167'BXKE
```

SUPERBASE - CONTINUED

Don't forget the round square (or was it square round?) brackets around surname" are special.

A bit awkward, but it works. This sort of string handling routine can be used in many and varied forms (rather like the example we used last year when discussing relative files and ISAM).

Maybe you have struck this problem too, and solved it in a way that we all could share. If you have, drop me a line at 35 Calder Crescent, Holder ACT 2611, or leave a message on VIATEL 628835840. Or maybe you have found/solved something else that would be of interest? Same deal - we're always glad of feedback.

Next issue - Version 2.02 and the hidden commands the manual doesn't mention.

delete/old 64

This routine provides a shorthand method for recovering a program that has been NEWed, and a means of deleting unwanted lines in a block. This saves the tedium of entering a blank line number, and pressing RETURN for each line.

The keyboard routines are presented at the head of the program. "D" uses the same syntax as LIST, which will make it easy to remember.

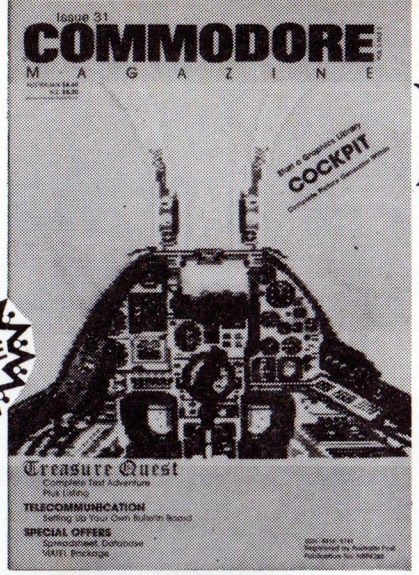
If you are not using HELPOUT, omit the last 5 characters ('XXXX) on each line.

```
100 REM: DELETE/OLD ROUTINE'BSAA
110 REM: FOR C64 DIRECT MODE'BRKB
120 REM: D M BALEAN 5/85'BNFA
130 REM:'BBAX
140 PRINTCHR$(147):PRINT"     DELETE/OLD":PRINT'EHLH
150 CLR:FORA=49152TO49331:READB:POKEA,B:C=C+B:NEXT
    'JXWK
160 IFC<>19761THENPRINT"DATA ERROR":END'GHBJ
170 PRINT"DELETE:-"'BALE
180 PRINT" D    = DELETE ALL"'BARI
190 PRINT" D L1  = DELETE L1 ONLY"'BANK
200 PRINT" D L1-L2 = DELETE L1 TO L2"'BAPC
210 PRINT" D L1-  = DELETE L1 TO END"'BAQD
220 PRINT" D -L2  = DELETE UP TO L2"'BARE
230 PRINT:PRINT"OLD:-"'CBHB
240 PRINT" O = RECOVER NEWED OR DELETED PROGRAM"
    'BAXK
250 PRINT"   IF STILL IN MEMORY"'BADH
260 SYS49152:NEW'CGBE
270 DATA 169, 11,141, 2, 3,169,192,141, 3, 3, 96, 32'BLLK
280 DATA 96,165,134,122,134,251,132,123,132,252, 32,115'BUMN
290 DATA  0,240,240,176, 3, 76,144,164,201, 79,208, 12'BPCN
300 DATA 32,115, 0,240,115,166,251,164,252, 76,134,164'BSAF
310 DATA 201, 68,208,245, 32,115, 0,208, 6, 32, 68,166'BOJG
320 DATA 76, 11,192,166, 43,164, 44,176, 12, 32,107,169'BQSH
330 DATA 32, 19,166,144,220,166, 95,164, 96,138, 72,152'BRCI
340 DATA 72, 32,121, 0,240, 36,201, 45,208, 27, 32,115'BORI
350 DATA  0, 56,165, 45,233, 2,170,164, 46,176, 1,136'BNOJ
360 DATA 32,121, 0,240, 22, 32,107,169, 32, 19,166,176'BPXL
370 DATA  5,104,104, 76, 41,192,160, 0,177, 95,170,200'BPYM
380 DATA 177, 95,168,104,133, 96,104,133, 95,152,160, 1'BRRN
390 DATA 145, 95,136,138,145, 95,140, 0, 2, 76,169,164'BPAO
400 DATA 169,255,160, 1,145, 43, 32, 51,165, 24,165, 34'BPJG
410 DATA 105, 2,133, 45,165, 35, 32, 85,166, 76,116,164'BPPH
```

NOTE:
This subscription offer supercedes and invalidates all subscription and re-subscription offers prior to the date shown on card.

A CHANCE TO WIN A COMMODORE 128

Refer Page 31



PLEASE ALLOW 4-5 WEEKS FOR FIRST ISSUE DELIVERY

VALID FROM OCTOBER 1 1985

$3.75 per copy

SAVE ON NEWSTAND PRICES BY SUBSCRIBING

☐ 6 ISSUES $22.50

START AT ISSUE 34

Enclosed please find Cheque/Money Order for $....

CREDIT CARDS:
Bankcard/Visa/Mastercard/AMEX No.....

NAME .....

ADDRESS .....

CITY..... STATE/P.CODE.....

Signature ..... Expiry Date .....

NEW ZEALAND $NZ25.00 (Personal cheques accepted) + $A8 Airmail

OVERSEAS $A27.00 + $A8 Airmail (optional)

A CHANCE TO WIN A COMMODORE 128

Refer Page 31

PLEASE ALLOW 4-5 WEEKS FOR FIRST ISSUE DELIVERY

VALID FROM OCTOBER 1 1985

$3.75 per copy

SAVE ON NEWSTAND PRICES BY SUBSCRIBING

☐ 6 ISSUES $22.50

START AT ISSUE 34

Enclosed please find Cheque/Money Order for $....

CREDIT CARDS:
Bankcard/Visa/Mastercard/AMEX No.....

NAME .....

ADDRESS .....

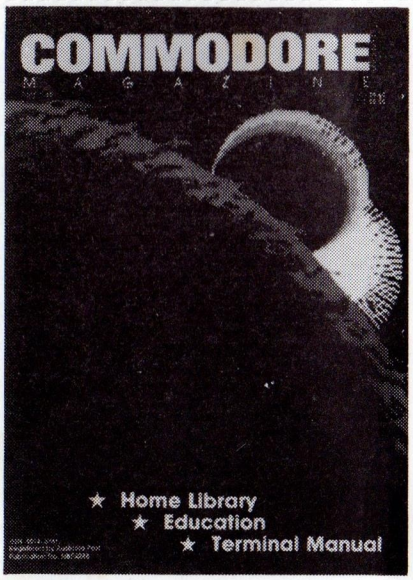
CITY..... STATE/P.CODE.....

Signature ..... Expiry Date .....

NEW ZEALAND $NZ25.00 (Personal cheques accepted) + $A8 Airmail

OVERSEAS $A27.00 + $A8 Airmail (optional)

We're Cleaning out the STOREROOM Get Your Missing Copies while stocks last.



Delivery
Some Issues are in limited supply. Sales will be made on a "first come first serve basis."

$3.60 per copy including P & P

BACK ISSUE ORDER FORM

PLEASE FORWARD ME THE FOLLOWING BACK ISSUES OF COMMODORE MAGAZINE

VOL 4	1 ☐	VOL 4	6 ☐	
	2 ☐	VOL 5	1 ☐	
	3 ☐		2 ☐	
	4 ☐		3 ■	SOLD OUT
	5 ☐		4 ■	

Enclosed is Cheque/Money order for $.....

CREDIT CARD:
Bankcard/Visa/Mastercard/AMEX No.....

NAME .....

ADDRESS .....

CITY..... P/CODE.....

Signature..... Expiry Date.....

N.Z. & Overseas – Please pay in Australian Currency
N.Z. personal cheques accepted at current rate of exchange.

POSTAGE
PAID
WITHIN
AUSTRALIA

FREEPOST 24
KIM BOOKS
82 ALEXANDER ST.,
CROWS NEST 2065

POSTAGE
PAID
WITHIN
AUSTRALIA

FREEPOST 24
KIM BOOKS
82 ALEXANDER ST.,
CROWS NEST 2065

POSTAGE
PAID
WITHIN
AUSTRALIA

FREEPOST 24
KIM BOOKS
82 ALEXANDER ST.,
CROWS NEST 2065

Random * Tables64

A. Wiggins

Random * Tables will ask answers to tables from 1*1 to 12*12.

```
0 REM A WIGGINS MELBOURNE VIC.'BVKE
1 PRINT "[CLR,DOWN]": PRINT TAB(10)"RANDOM TIMES
  TABLES": PRINT 'EFUI
2 PRINT TAB(7)"PLEASE TURN UP THE SOUND."'CCMI
3 PRINT : PRINT TAB(7)"PRESS RETURN WHEN READY"'DDVJ
4 Q$="": GET Q$: IF Q$<> CHR$(13) THEN 4'HNHJ
5 GOSUB 100: POKE 198,0'CJLF
6 PRINT "[CLR,DOWN2]HOW MANY QUESTIONS":
  PRINT "WOULD YOU LIKE TO ANSWER ?."'CBMS
7 Z=0: INPUT Z: IF Z<=0 THEN PRINT "[UP2]": GOTO 7'IJCN
20 A=INT (12* RND (1))+1'FJGD
30 B=INT (12* RND (1))+1: PRINT : PRINT 'HLXF
31 PRINT "TYPE IN THE ANSWER TO QUESTION ";AN+1'CEMK
32 IF AN=>1 THEN PRINT "IF YOU HAVE HAD ENOUGH TYPE IN"
   'FDAM
33 IF AN=>1 THEN PRINT "-1 AS YOUR ANSWER TO
   QUESTION";AN+1'GHSP
34 PRINT 'BACD
35 PRINT 'BACE
38 PRINT TAB(7)"QUESTION ("AN+1")"" OF ("Z")": PRINT :
   PRINT 'FIJQ
39 PRINT TAB(10)A"*"B"=[SPACE2]";:C$="": INPUT C$:
   C=VAL( C$): IF C=0 THEN PRINT "[UP2]": GOTO 39'LYGV
41 IF C=-1 AND AN<1 THEN PRINT "[UP2]": GOTO 39'IIRJ
42 IF C=-1 AND AN=>1 THEN GOTO 60'IHUI
43 PRINT "[CLR]"'BATE
44 PRINT 'BACE
45 PRINT 'BACF
46 AN=AN+1'CFUI
50 IF C=A*B THEN PRINT "YOU GOT IT RIGHT ":R=R+1:
   PRINT A"*"B"=";A*B: GOSUB 2000'KSEQ
55 IF C<>A*B THEN PRINT "WRONG":W=W+1:
   PRINT A"*"B"=";A*B;",NOT "C: GOSUB 1000'LUYV
56 IF AN=>Z THEN 60'EFCK
57 GOTO 20'BCJI
60 PRINT "[DOWN2]QUESTIONS ANSWERED ";AN'BDYI
65 PRINT "[DOWN]QUESTIONS RIGHT ";R'BCFM
66 PRINT "[DOWN]QUESTIONS WRONG ";W'BCAN
67 R$=STR$(R/AN*100): REM RIGHT'FQDQ
68 W$=STR$(W/AN*100): REM WRONG'FQDR
70 PRINT "[DOWN]" MID$(R$,1,7)" % RIGHT"'CICH
71 PRINT "[DOWN]" MID$(W$,1,7)" % WRONG"'CIWI
72 PRINT : PRINT 'CBHG
73 IF R/AN*100<=50 THEN PRINT " *********TRY A LITTLE
   HARDER*********"'HIKU
74 IF R/AN*100>50 AND R/AN*100<=75 THEN
   PRINT "[SPACE7]*********NOT BAD*********"'LQLX
75 IF R/AN*100>75 AND R/AN*100<=85 THEN PRINT
   "[SPACE6]*********VERY GOOD*********"'LQQA
76 IF R/AN*100>85 AND R/AN*100<100 THEN PRINT
   "[SPACE6]*********EXCELLENT*********"'KRDA
77 IF R/AN*100=100 THEN PRINT "[SPACE2]*********YOU
   DON'T NEED ME*********"'GJWX
95 PRINT : PRINT TAB(3)"DO YOU WISH TO DO SOME MORE Y/N
   [SPACE2]?."'DDEV
96 Z$="": GET Z$: IF Z$<>"Y" AND Z$<>"N" THEN 96'JMCU
97 IF Z$="Y" THEN CLR : GOSUB 100: GOTO 6'GISR
98 END 'BACN
100 PRINT "[CLR,DOWN]": PRINT TAB(17)"MUSIC":
    PRINT TAB(10)"(BY JIM BUTTERFIELD)"'FIJH
110 L1=54272:L2=54279:L3=54286'DXCD
120 H1=L1+1:H2=L2+1:H3=L3+1'GRDF
130 V1=L1+4:V2=L2+4:V3=L3+4'GRDG
140 POKE 54296,15'BIAB
150 POKE V1+1,9: POKE V2+2,0'ELNE
160 POKE V2+1,36: POKE V2+2,36'ENNG
170 POKE V3+1,18: POKE V3+2,170'EOMH
180 T=TI'BDME
200 POKE V1,16: POKE V2,32: POKE V3,16'DRKB
210 READ S: IF S=0 GOTO 290'EHWA
220 READ X1,Y1,X2,Y2,X3,Y3'BRLC
230 IF X1 THEN POKE H1,X1: POKE L1,Y1: POKE V1,17'FTLG
240 IF X2 THEN POKE H2,X2: POKE L2,Y2: POKE V2,33'FTPH
250 IF X3 THEN POKE H3,X3: POKE L3,Y3: POKE V3,17'FTXI
260 T=T+S'CDLE
270 IF T>TI GOTO 270'DGDG
280 GOTO 200'BDBF
290 FOR J=L1 TO 54296: POKE J,0: NEXT J'FOLL
295 RETURN 'BAQK
300 DATA 20,34,75,21,154,8,147'BVTB
310 DATA 20,34,75,25,177,0,0'BTMC
320 DATA 20,38,126,28,214,6,108'BWRD
330 DATA 20,43,52,25,177,0,0'BTHE
340 DATA 20,34,75,21,154,8,147'BVTF
350 DATA 20,43,180,25,177,0,0'BUHG
360 DATA 20,38,126,22,227,8,23'BVPH
370 DATA 20,0,0,25,177,0,0'BRDH
400 DATA 20,34,75,21,154,8,147'BVTC
410 DATA 20,34,75,25,177,0,0'BTMD
420 DATA 20,38,126,28,214,6,108'BWRE
430 DATA 20,43,52,25,177,0,0'BTHF
440 DATA 20,34,75,21,154,8,147'BVTG
450 DATA 20,0,0,25,177,0,0'BRDG
460 DATA 20,32,94,22,227,8,23'BUPI
470 DATA 20,0,0,19,63,6,108
500 DATA 20,34,75,21,154,8,147'BVTD
510 DATA 20,34,75,25,177,0,0'BTME
520 DATA 20,38,126,21,154,7,163'BWPF
530 DATA 20,43,52,17,37,0,0'BSLF
540 DATA 20,45,198,28,214,7,53'BVBH
550 DATA 20,43,52,34,75,0,0'BSMH
560 DATA 20,38,116,28,214,0,206'BWJJ
570 DATA 20,34,75,22,227,0,0'BTFK
600 DATA 20,32,94,25,177,6,108'BVWE
610 DATA 20,25,177,22,227,0,0'BUGF
620 DATA 20,28,214,21,154,7,53'BVMG
630 DATA 20,32,94,19,63,8,23'BTVH
640 DATA 20,34,75,21,154,8,147'BVTI
650 DATA 20,0,0,25,177,6,108'BTIJ
660 DATA 20,34,75,21,154,4,73'BUPK
670 DATA 20,0,0,0,0,0,0'BONK
700 DATA 0'BBDB
1000 REM ** SOUND SUBROUTINE **'BTHX
1020 S=54272'BGJW
1030 POKE S+24,15'C GO X
1040 POKE S+5,50'CFOY
1045 FOR X=1 TO 50'DEVE
1050 POKE S+1, RND (X)*32+50'FKQD
1060 POKE S+4,17'CFQB
1070 FOR I=1 TO 10: NEXT 'EFJD
1080 POKE S+4,16'CFPD
1090 NEXT X'BBRC
1091 FOR X=1 TO 24: POKE 54272+X,0: NEXT 'GOFJ
1092 RETURN 'BAQE
2000 REM ** SOUND SUBROUTINE **'BTHY
2010 FOR Q=1 TO 10'DEKW
2020 S=54272'BGJX
2030 POKE S+1,20'CFHY
2040 POKE S+5,9'CEUA
2050 POKE S+6,9'CEVB
2060 POKE S+24,15'C GO C
2070 POKE S+4,17'CFQD
2075 FOR X=1 TO 20: NEXT X'EGNJ
2080 FOR X=1 TO 20: NEXT X'EGNF
2090 POKE S+4,16'CFPF
2094 FOR X=0 TO 10: NEXT : NEXT Q'FHLL
2095 FOR X=1 TO 24: POKE 54272+X,0: NEXT 'GOFO
2100 RETURN 'BAQU
```

BOOKS & THINGS

CONTINUED FROM PAGE 27

TRAIN DISPATCHER

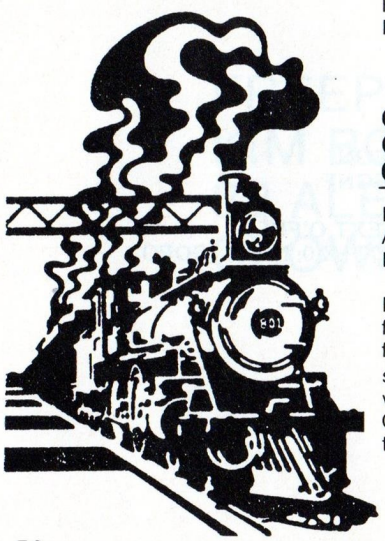
Title: Train Dispatcher
Tape/Disk: C-64, Vic20
Authors: R.W. Brew & T.B. Levine
Publisher: Signal Computer Consultants - P.O. Box 18222 Pitsburg PA 15236, U.S.A.
Price: $29.95 American (inc.P&P)
Availability: Direct from publishers
Reviewed by: T. Steadman

GENERAL DESCRIPTION
A game that simulates control of railroad operations for a 8 hour shift, in control of 150 miles of territory. You are the dispatcher in the central control traffic office and have to move up to 12 trains by positioning points clearing signals ahead of the trains. There are 5 levels from visitor to trainmaster. You have the following colour displays to help you: Overview, 20 zones, Train sheets, block permits.

GOOD POINTS
The colour display are easy to read and well set out, keyboard overlay makes operation easy, sound used to tell you when train moves to new section and when commands worked. The Manual is set out well and easy to read with a coloured reference section in the middle. Additional information on mileages, playing level differences and scoring is at the rear of the book. Operation is very realistic.

BAD POINTS
Music has clicks at the end of notes. While on the zone display if the train moves to a new section the train display is updated but the train number above the old section is not deleted. I personally do not like the way the function keys are used and there is no high score feature. Also the keyboard overlay will not fit SX64.

SUMMARY
This game has been well thought out except for the minor flaws noted above, for railroaders, railroad modellers, budding railroad people and people who want to test there mind and reflexes at the same time this is for you. Yes, I do think it is good value for money.



CAULDRON

For the Commodore 64

Reviewed by: Michael Spiteri

Cassette: R.R.P. $20.00
Publisher: Melbourne House

Cauldron is a new arcade/adventure game from Melbourne House.

Life as a witch isn't easy, you know, flying around the planet collecting keys. Those darn spooks, whales, plants(!), and flying things sure drive me up the broomstick!!

That is what the player must do in the first level of this game, here we have a scramble type game with very impressive graphics. Trees, oceans, doors, spooks($#&%) all look very realistic. Melbourne House have used realistic 3D graphics to make the first level a very exciting game.

I had problems first controlling the witch, but soon picked it up. The witch must collect 4 coloured keys from around the world that open the exit to the underworld where more nasties and more levels exist. In the underworld lie ingredients that are required to make a spell to get rid of a pumpkin that is pestering the country – Goody, the witch is a good witch.

The graphics in the next levels are just as good as the first, and the game is just as hard. You have 9 hags to play with (pardon the expression), when they all run out – game over. Underground we have a very difficult platform game and a unique game that defies description. I was very impressed with the graphics, sound and animation. The program is well packaged with interesting documentation. The game also loads in under 3 minutes using the PAVLODA loader. This is definitely the best arcade game Melbourne House has to offer. I believe they have a winner with Cauldron. At only $20 it has to be excellent value for money. Ask for a demonstration at your local computer store – I'm sure you'll be impressed.

From Melbourne House
Price: $20 on turbo cassette.

GRAPHICS AND SOUND ON THE COMMODORE 64

Publisher: Prentice-Hall Australia
Author: Greg Perry
Reviewed by: Peter Davies

I've always been somewhat confused about the procedures to follow when using the SID sound synthesizer chip and the VIC II video controller chip used in the Commodore 64. If you have too then there is help at hand at a reasonable price in the form of Greg Perry's excellent book 'Graphics and Sound on the Commodore 64'.

The book follows the well proven adage that practice makes perfect and you will find a continuous stream of examples and exercises to assist you in coming to terms with the 64's powerful graphics and sound capabilities. You soon get the feeling that you've seen this sort of thing before (high school maybe?) but it proves to be an excellent way to learn.

The book is also crammed full of technical information for those so minded and yet I feel that even beginners should have little difficulty in understanding most of it.

The first chapter includes a brief explanation on how to use the book, a guide to the conventions used in entering programs listed in the book and an explanation of the binary number system so vital when PEEKing and POKEing about the 64.

The second chapter deals with the generation of simple colour and graphics using the 64's block characters available from the keyboard. The techniques of handling colour and screen memory are explained and this section of the book culminates in the design and implimentation of a fairly complex maze program.

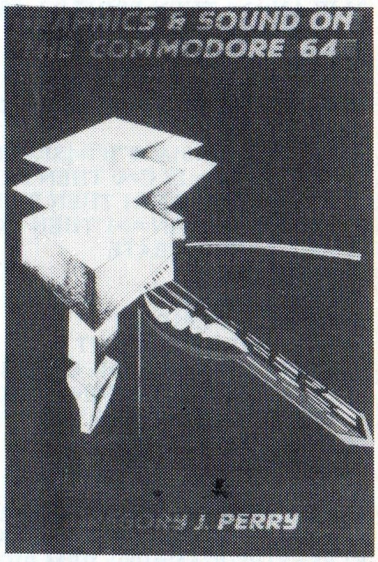
Chapter 3 is where the complicated stuff begins and this is where you come to realise the value of Greg's unique approach. There are some good examples of how to generate programmable characters and there is also an excellent utility for creating data statements from any section of RAM so that they can be used in your own program. The chapter finally finishes up with an explanation of how to program multicolour characters.

Chapter 4 launches into the fascinating world of high resolution graphics and again there are plenty of BASIC examples to show you how its all done. There is even a BASIC program to generate some machine code routines to assist in speeding up the graphics on the 64.

Chapter 5 explains the procedure required to use the 8 sprite available on the 64 and there are some good examples of how to generate sprite shapes easily from within a BASIC program. There is also an explanation of how to move sprites around the screen and how to detect collisions between sprites, the background and other sprites.

Chapter 6 onwards deals with the use of the SID chip. The nature of sound is explained and a description of the basic operation of the SID is given. Then the book describes the individual SID registers and how to program them with plenty of examples and exercizes to keep you interested. I liked the explanation of programming multiple voice music and there are 2 music utilities provided to assist you in programming your own music.

The appendices consist of a section describing the operation of banked memory in the 64, a section on programming the 64's joysticks, a description of a smooth scrolling routine, a table of even-tempered notes and last but not least a description of programs that can be found on the disc that you may purchase with the book (I didn't have this).



Overall the book is well set out and easy to follow, however I do have two criticisms. Most of the program examples in the book are reasonably short and can be easily understood, however some of the larger examples primarily the Maze program on page 44 could do with some program structure to make them more readable and easier to understand. A central control section with GOSUBs calling subroutines which perform specific tasks makes a program much easier to understand (especially for a novice) as well as making it easier to debug and modify. The other criticism centres around the lack of memory diagrams to assist in explaining the bank selection features for the VIC II chip. You know the old cliche, a pictures worth a thousand words.

At $15.95 for the book and an extra $10.00 for the disc this has to be one of the best buys for someone wishing to gain an insight into the graphics and sound capabilities of the Commodore 64. □

COMMODORE DOCTOR

by Dr. Greg Perry

The aim of this column is to help our readers with any problems they have with CBM/PETs, VICs, C64s, Plus 4/C16 and associated Commodore equipment. Send us your queries and we will do our best to provide an intelligent answer.

Alternatively, if you don't have any immediate problems but have discovered some smart tricks in BASIC or machine code, or even better ways to program some of our answers/articles we would be interested to hear from you. You never know the routine may even win you a prize for the best item published each month. Also drop us a line if you would like a specific topic covered in the magazine.

Write to

Commodore Doctor
The Commodore Magazine
82 Alexander Street
Crows Nest, NSW 2065.

OR MAIL them to me on VIATEL 738329500

Please ensure that any program listings are in NICE LISTER format and include a REM statement with your name and address. (By the time it passes through several hands and reaches me sometimes bits of the letter can have been mislayed. If not, I'm also likely to lose it!) Machine code programs should be in assembler format and not directly in hex. I apologise for the fact that, in general, letters can not be answered personally. Also, because of printing schedules and other factors, some questions may not appear until two months after they are received.

COMMENT

I would like wish all readers the best for the festive season and to thank all those who have supported me over the past year. This column has been quite interesting to write, and I have enjoyed the many favourable comments and amusing feedback received.

Questions and Answers.

Q. Our school has a network of C64s, linked by a VICSwitch together with an EPSON RX-80 printer and an Epson CO64 Interface card for the C64. Everything works perfectly except that we have not been successful in getting a LOGO graphics dump from the C64. Though offering advice, neither Commodore nor their their agents have been able to help us with a software program to enable us to do LOGO screen dumps.

Are you familiar with any procedure to dump LOGO graphics (including expansion options) onto the EPSOM RX-80? Or, do you now of anyone who would be in a position to supply the needed information.

S McKinney
Campbellfield Primary School
Augusta Avenue
Campbellfield Vic 3061

A. As far as I can ascertain you should be having no problems provided you set up the LOGO program for use with a 1525/801 type printer. The procedure is discussed in the manual as far as I remember, however at this point in time I cannot lay my hands on my manual. I have included your full address in case others may have any further comment on this problem. If this doesn't work please drop me another line and we will try to look into the matter in more depth.

Q. Is there a way you can store programs (games) from cassette to disk? I have a C64 and a 1541 drive and what I would like to do is store the game ELITE from cassette to disk. How would I go about it? (The reason for storing onto disk is for backup only.)

Also I have a VZ200 printer plotter (4 colour) which uses a centronics interface input. Is there a circuit that I can interface the C64 with the VZ200?

Joseph Tiziano
Fairy Meadow, NSW

A. Unfortunately, there is no simple, if even possible, way to transfer commercial cassette based games from tape to disk. Most of the latest turbo protection systems involve the loading of several parts of the program into different memory locations as well as an auto run. Without breaking the protection system (which is illegal in any case), it is not possible to obtain a 'clear' copy of the program so as to save it onto a disk.

For you VZ200 printer plotter, the easiest way would be to purchase one of the standard Centronics interfaces widely available. However this will be somewhat expensive. The alternative is to make yourself up a centronics cable for the user port of the C64. Since the user port has all the lines for a parallel centronics connection it involves only two plugs, cable, and a bit of soldering. As far software to drive the user port/centronics output, there is no need to reinvent the wheel since a nice set of 'wedges' are available from the ACT Commodore User Group. Drop them a line c/o PO Box 599, BELCONNEN, ACT 2616. They can also supply a wiring diagram for the cable!

Competitions.

Competition 2

I have had quite a range of entries for this one. Still(!) nobody has solved it.

The Problem: Two computer experts, who live on country properties, are having a quiet drink in a country pub. Expert 'A' owns a rectangular property which is totally enclosed within a 23 by 23 kilometre square.

Expert 'B' knows the area of the property and that the sides are whole numbers (integers), but does not know the dimensions. He ('B') asks if the breadth of the property is greater than half the length. Expert 'A' answers. (We are not privilege to rural conversation, but we know the answer was either yes or no.) On hearing the answer, expert 'B' can now calculate the dimensions of the property. A farmer has been quietly listening to the conversation, and, although he did not previously know the area of the property, on hearing both the question and answer, thinks for a while, and then, to their astonishment, tells the computer people what the area and dimensions of the property are. (What happens to him after that we won't discuss.)

That's the problem. What is the area and dimensions of the property?

Some of this months' entries were quite amusing. Many answers were expressed in supremely confident terms, but most were way off the mark. One reader even suggested that the computer experts had been at the pub for too long (never!) and were in fact quite beyond reason! Another reported that he came up with a program which ".. ran for 15 minutes and didn't help me a bit!" Yet another commented "Do you actually know just how many different sized properties could fit into a 23 km squared boundary! I seem to understand it but lord knows how long it would take to calculate."

I assure you that there is a specific logical answer to the problem. I actually sat down for an hour or so and double checked the solution and there really is a valid answer.

Most people are totally missing the point. Somehow you must find a rectangle which can be PROVED to be 'unique' by the L>B* relationship with respect to all other possible rectangles which may be enclosed within the 23x23 square. There is no guesswork.

You must, therefore, work out ALL the possible rectangles (including squares) which can be contained within the area 23x23. Or, more correctly, a list of all the different areas and the possible dimensions. Any given area usually has more that one possible set of factors. For example, the area of 30 has factors 3x10, 30x1 and 5x6. (Remember that it is possible to have a length greater than 23 by laying the rectangle along the diagonal.)

Then, compare the way in which the dimensions relate to L>2*B. For example, the area 30 has two sets of dimensions where L>2*B and one where L<2*B. In this case, the answer would have to be 30x1 since there is no way to chose between the other two. (That's not the real answer!) If one finds another area with factors related similarly (two greater than and one less than) then neither can be the 'unique' one.

Now comes the easy part! Find out which rectangle is the unique one! You must also write a BASIC program to prove your answer. It is not easy and requires a considerable amount of calculation.

The winners jackpot stands at 8 disks or 8 C10 Cassettes. *Greg has also agreed to throw into the pot a copy of his recently published book on Sound and Graphics for the C64.*

COMPETITION 3/2

The problem was: using PRINT statements containing only one '*', write a BASIC program to draw the following pattern on the screen.

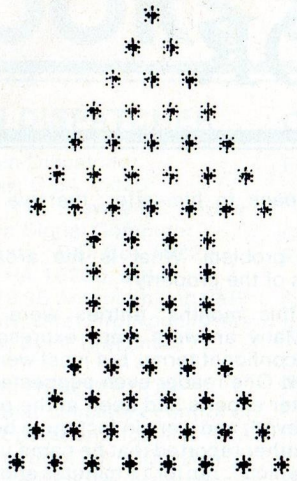
With over thirty answers the selection of the winner was not an easy task. Most entries were quite competent. It was encouraging to find so many entries from our younger readers. (By the way Robert, maybe it could be a peaceful Xmas Tree (?) instead of a rocket.)

The following winner was chosen, mainly because of their deviousness in obtaining a one liner!

```
1 FOR I=1 TO 5:X=20-I+(I>7)*(4-I)-(I>11)
*(11-I):PRINT:PRINT SPC(X):
FOR J=1 TO 20-X: PRINT"* ";:NEXT J,I
```

CONTINUED OVERLEAF

CONTINUED FROM PREVIOUS PAGE



(Remember to abbreviate all keywords. SPC(is sP. There is also no need for a semicolon after the SPC(X) command.) - D. & S.G Pfennig, Hackham West 5163

Competition 4/1. Machine Code.

The problem was: Write a program to convert a three digit ASCII number to a single byte hex number. Assume that decimal 159 as its ASCII equivalent is stored in memory starting at location $1000. That is

Loc	Hex	ASCII
$1000 =	$31	= 1
$1001 =	$35	= 5
$1002 =	$39	= 9

The program must convert these values to a single byte hexadecimal value ($9F) which must appear in location $1004. (It must also work forany decimal number 0-255!)

Mail for this problem was most interesting with many people using lovely routines. First of all, everyone knows that 159 decimal is $9F but did anyone but Mike Gregory from Winston Hills realize that the sum of $31+$35+$39 = $9F? I didn't!

One interesting point was made by Deiter Pfennig. How do you multiply by 100 in machine code? The usual way is to multiply by ten twice, but if you give it some thought it can be done like this.

```
LDA #....    ;N must be <3
ASL          ; *2 = 2*N
ASL          ; *2 = 2*N = T1
STA $FE      ; save partial product 4*N
ASL          ; *2 = 8*N
ASL          ; *2 = 16*N
ASL          ; *2 = 32*N = T2
STA $FD      ; save partial product 32*N
ASL          ; *2 = 64*N
CLC
ADC $FD      ; = 64*N + 32*N = 96*N
CLC
ADC $FE      ; 96*N + 2*N = 100*N
BRK
```

Unfortunately this didn't help him.

For most MC programs it is sufficient to do the job properly, irrespective of the speed since it's generally fast enough. But in order to write the fastest program one must optimise the code and find the correct balance. Generally, fast code itilizes some of the following:-

A. Uses Zero page addressing as much as possible
Note: STA $FE / LDA $FE is 5 cycles
PHA/PLA is 7 cycles !

B. Limits branching and JMPs and JSRs

C.Limits index addressing and uses Z page and limits crossing page boundaries (adds 1 extra clock cycle).

D. Limits calculation and uses lookup tables.

For this exercise, the fastest method involves using a lookup table (D) and overcomes the limitations of (A) and (C). Two people sent in almost identical entries. These winning entry is shown below by John O'Donnell from Whyalla Stuart, SA. Also Nick Hawthorn from Auckland NZ was extremely close but his entry had a minor problem.

```
$1005 START LDA $1002    ; get decimal unit
            EOR #$30     ; remove ascii **
            CLC          ; prepare to add
            LDY $1001    ; get tens in Y
            ADC $0FEE,Y  ; lookup and add decimal tens
                         ; with full assembler use
                         ; ADC TENS-48,Y
            LDY $1000    ; get decimal value for 100s
            ADC $0FEB,Y  ; lookup and add decimal hundreds
                         ; ADC HDRS-48,Y
            STA $1004
            BRK

$101B HDRS  .BYT $00,$64,$C8                          ;0,100,200
$101E TENS  .BYT $00,$0A,$14,$1E,$28,$3C,$46,$50 ;0,10,20...etc
            .BYT $5A                                  ;... up to 90
```

Execution time is 30 clock cycles. This could be dropped to 28 if the tables were moved so that the ADC instruction did not have to cross a page boundary.

I think that an AND #$0F in line 2 would be better but the EOR works.

In future I must remember to make any future machine code competitions easier to judge since I specified the winner would be judged on speed (clock cycles) alone, I have had to calculate the speed of each entry one by one!

By way of an example, a typical method of solving the problem but not using lookup tables was entered by C.A.P Bundy from Applecross, WA. (Slightly edited version.)

```
LDA $1002 ;get units
AND #$0F  ;chop ascii
STA $42   ;save U
LDA $1001 ;get tens
AND #$0F  ;chop ascii
STA $41   ;save T
LDA $1000 ;get hundreds
AND #$0F  ;chop ascii
STA $40   ;save H
CLC
ASL A     ;H*2
ASL A     ;*2 = H*4
ADC $40   ;+H = H*5
ASL A     :*2 = H*10
ADC $41   ;add T
STA $41   ; save total tens
          (H*10+T)
ASL A     ;*2
ASL A     ;*2
ADC $41   ;+ total tens
          = (H*10+T)*5
ASL       ;*2 =(H*10+T)*10
ADC $42   ;add units
STA $1004 ;done at last
BRK
```

Quite a nice program in 64 clock cycles.

Keep up the programming. Hopefully there will be more competitions in the new year.

Merry Xmas to all,
Greg Perry

Definitions

BASIC - Bloody Awful Stuff In Computers.
LIST - What you begin to do to one side when you've been programming too long in one go!
DEF FN - A FN that can't hear very well.
DIM - Not clever enough
END - The bit you bang when it doesn't work
GET - What you say to your little brother or sister as he/she approaches your disk drive.

Definitions courtesy of Stuart Elflett Alias The Happy Hacker - Qld.

High Score

SPRITEMAN 64
92,290 Brendon Madden VIC
TRASHMAN
407,705 Stephen Norman N.S.W.
VIC FROGGER
225,000 J.H.Fry ACT
FALCON PATROL
5,525 Jeremy Bone SA
BEACH HEAD
95.000 Jeremy Bone SA
SAMMY LIGHTFOOT
86,830 Felix Ryan QLD
THRESHOLD
425,800 S. Puckeridge TAS.
MUNCH-MANIA
49,000 Shirley Puckeridge TAS.
C64 FROGGER
74,840 Shirley Puckeridge TAS.
ZAXXON
60,200 Shirley Puckeridge TAS.
GHOSTBUSTERS
$26,800 Shirley Puckeridge TAS.
(AC No 50454700)
WIZARD OF WOR
104,390 Shirley Puckeridge TAS.
(Dungeon 19-Worlord Supreme)
LUNAR LEEPER
127,040 "Happy Hacker - Qld"

PRIZE WINNERS

Our "Piling" System is such that a couple of you have missed out on your Prizes. If so write to the Editor stating Vol. & Issue No. of the magazine where your name appears as a Prize Winner.
Mervyn Beamish

Masochist Corner.....

Graphics Library No. 4: Chessplayer

Kiwisoft Programs Pty. Ltd.

The GRAPHICS LIBRARY series is very popular with our readers. Based on the enquiries we have received, there are a number of you out there that text entered each picture while others wait for the relevent disk offer to arrive.

The DISPLAY.BAS program that is published below is yet another way of displaying your picture on the screen through you C64. This program, adapted from one published in the PAINTPIC™ manual, allows you to select pictures and change between text and graphic screens. The problem with DISPLAY.BAS is that it is very slow. While loading the picture from disk your drive will stop and start, and generally act up – DON'T PANIC!

CHESSPLAYERGEN

This program generates CHESSPLAYER picture (refer front cover). We have dropped the usual checksum facility this month but HELPOUT has been utilized to assist you in loading the program (refer page 4).

First load and run CHESSPLAYERGEN which will generate the CHESSPLAYER file, then save it to tape or disk.

Secondly NEW, LOAD "DISPLAY.BAS" and RUN. The picture will come up on your screen.

NOTE There should be 13 items in each of the datalines except the last two. Check that you put in all commas shown.

Good text entry!

DISPLAY.BAS PROGRAM

```
1 REM DISPLAY PICTURE'BORD
2 REM'BARA
3 REM BY KIWISOFT PROGRAMS LTD'BVHH
4 REM'BARC
5 REM YOU ARE FREE TO USE THIS PROGRAM FOR YOUR OWN WORK'BPKP
10 PRINT:PRINT"THIS UNLOCKED BASIC PROGRAM IS THE DISK"'CBBI
20 PRINT"VERSION OF THE PROGRAM IN THE PAINTPIC MANUAL"'BAYK
70 REM MAINLINE BEGINS'BOWH
80 DIMU1%(199)'BILG
90 INPUT"NAME OF PICTURE";U$:REM NEED U$ FOR THE LOAD ROUTINE'CCUR
100 PRINT"{HOME}";:REM CLEAR AND HOME'COQA
110 GOSUB32000:REM LOAD THE PICTURE'CUDC
120 GOSUB31000:REM SAVE CM AREA TOP'CTVD
130 GOSUB1200:REM CLEAR TOP OF TEXT SCREEN'CANG
140 PRINT"HIT Q TO QUIT OR ANY CHAR TO SWITCH SCREENS"'BAML
150 GETA$:IFA$=""GOTO150'EIBE
160 IFA$="Q"THEN STOP'ECRE
170 GOSUB31100:GOSUB32100:REM RESTORE CM AND TURN ON PICTURE'DMTN
180 GETA$:IFA$=""GOTO180'EIEH
190 GOSUB32200:IFA$="Q"THENSTOP:REM TURN OFF PICTURE AND ON TEXT, STOP IF Q
    'GPXS
200 GOTO120:REM CYCLE AGAIN
210 REM END OF MAIN PROGRAM BEGIN SUBROUTINES'BHAH
1199 REM SUBROUTINE TO CLEAR TOP 200 LOCS IN SCREEN AREA AND POSITION TO HOME
    'BGBD
1200 PRINT"{HOME}";:FORI=1TO200:PRINT" ";:NEXT:PRINT"{HOME}";:RETURN'INOC
30999 REM SAVE THE FIRST 200 CM TABLE VALUES'BDDW
31000 US=55296:FORU=USTOUS+199:U1%(U-US)=PEEK(U):NEXT:RETURN'KFCB
31099 REM RESTORE THE FIRST 200 CM TABLE VALUES'BGRP
31100 US=55296:FORU=USTOUS+199:POKEU,U1%(U-US):NEXT:RETURN'JEAB
31999 REMLOAD PICTURE U$ FROM TAPE'BVLV
32000 OPEN1,8,2,U$:US=55296:GET#1,U$:IFU$<>"P"THENSTOP'IALF
32001 GOSUB32090:UB=UW:UJ=US:REMSET TO LOAD CM'ECTF
32002 GOSUB32060:IFUJ<=US+999GOTO32002'GSFE
32003 UC=23552:UJ=UC:U=FRE(0):REM SET TO LOAD TS AND BS, GARBAGE COLLECT
    'FYMI
32004 GOSUB32060:IFUJ<=UC+999GOTO32004'GSQA
32005 UA=24576:UJ=UA:U=FRE(0):REM SET TO LOAD DOTS, GARBAGE COLLECT'FVIJ
32006 GOSUB32060:U=FRE(0):IFUJ<=UA+7999GOTO32006'IYMF
32007 CLOSE1:RETURN'CCCD
32060 GOSUB32090:UH=UW:GOSUB32090:UE=UW:GOSUB32090:UE=256*UW+UE:
    REM VAL,COUNT'JWFM
32061 FORUT=0TOUE-1:POKEUJ+UT,UH:NEXT:UJ=UJ+UE:RETURN'KXNI
32090 GET#1,U$:UW=ASC(U$+CHR$(0)):RETURN'GQXF
32099 REMTURN ON PICTURE SCREEN (SEE P101 IN PROGRAMMER'S REFERENCE GUIDE)
    'BGGX
32100 POKE53272,120:POKE53265,PEEK(53265)OR32:REMPICTURE ADDR AND BITMAP SET
    'FYEF
32101 POKE53270,PEEK(53270)OR16:REM MULTICOLOR MODE'EFFA
32102 POKE53281,UB:REM SET BACKGROUND COLOR'CCMA
32103 POKE56578,PEEK(56578)OR3:POKE56576,(PEEK(56576)AND252)OR2:REM BANK 1
    'IPUG
32104 RETURN'BAQA
32199 REMTURN OFF PICTURE SCREEN AND BACK ON TEXT SCREEN'BOVU
32200 POKE53272,21:POKE53265,PEEK(53265)AND223:REMPICTURE ADDR AND BITMAP
    RESET'FBFG
32201 POKE53270,PEEK(53270)AND239:REM MULTICOLOR MODE OFF'EJWC
32202 POKE53281,6:REM RESET BACKGROUND COLOR'CDQB
32203 POKE56578,PEEK(56578)OR3:POKE56576,(PEEK(56576)AND252)OR3:REM BANK 0
    'IPUH
32204 RETURN'BAQB
```

CHESSPLAYERGEN

```
1 REM PROGRAM TO MAKE PICTURE IN CADPIC FORM'BHRI
2 REM SUPPLIED BY KIWISOFT PROGRAMS LTD'BEDJ
110 PRINT:PRINT"BEGIN STORING PICTURE":PRINT'DCGE
120 PRINT"ENTER T FOR TAPE OR D FOR DISK STORAGE":INPUT" (T/D)";A$'CEWK
130 A$=LEFT$(A$,1):IFA$="T"THENDEV=1:SA=1:B$="":GOTO150'JBGJ
140 DEV=8:SA=2:B$=",S,W":IFA$<>"D"GOTO120'HRSI
150 OPEN1,DEV,SA,"CHESSPLAYER"+B$'CLLH
160 RESTORE:READK:READK:A=2:PRINT#1,"P";CHR$(K);:REMEMBER 2 SEMICOLONS'HHON
170 FORI=1TO864'DFRE
175 FORJ=AT06'DDBJ
180 D=0:READC:READB:IFC=0THEND=1'HLDJ
185 PRINT#1,CHR$(B);CHR$(C);CHR$(D);:REMEMBER 3 SEMICOLONS'FGAT
190 NEXTJ:READK:REM THROW OUT CHECKSUM'DUCL
195 A=1:PRINT"%";:NEXTI'DGRM
200 CLOSE1'BBIV
205 PRINT:PRINT"PICTURE SAVED":END'DCYG
1000 DATA 68,9,17,15,9,8,14, ,17,15,9,8,5189'BISY
1010 DATA 14, ,3,15,1, ,1,15,2,6,10,15,5082'BGLA
1020 DATA 9,8,14, ,5,15,2,6,6,15,3,5,5088'BFSB
1030 DATA 1,15,9,8,16, ,3,15,2,6,6,15,5096'BGOC
1040 DATA 3,5,1,15,9,8,16, ,3,15,2,6,5083'BFID
1050 DATA 10,15,9,8,23, ,5,15,1, ,2,15,5103'BGRE
1060 DATA 9,8,31, ,9,8,32, ,8,8,32, ,5145'BDAE
1070 DATA 8,8,32, ,6,8,1,10,1,8,25, ,5107'BEFG
1080 DATA 5,2,1,15,2,8,2,12,3,8,1,11,5070'BGOH
1090 DATA 1,8,25, ,6,2,7,8,4,15,23, ,5099'BEQI
1100 DATA 6,2,7,8,4,15,23, ,6,2,7,8,5088'BEVA
1110 DATA 2,15,1, ,1,15,23, ,6,2,7,8,5080'BEBA
1120 DATA 1,10,26, ,7,2,7,8,15, ,6,8,5090'BEHC
1130 DATA 5, ,2,2,1,11,1,12,2,2,1,11,5050'BFUD
1140 DATA 6,8,16, ,6,8,5, ,1,11,1,2,5064'BDHD
1150 DATA 2,11,3,12,2,8,20, ,6,12,5, ,5081'BFLF
1160 DATA 6,12,1,2,15, ,7,2,1, ,5,12,5063'BETF
1170 DATA 5, ,7,2,5,8,3,12,7, ,7,2,5058'BCPG
1180 DATA 11, ,7,2,4,8,1,11,3,12,1, ,5060'BENH
1190 DATA 3,12,3, ,7,2,11, ,7,2,4,8,5059'BDGI
1200 DATA 1,11,7,12,3,11,7,2,11, ,7,2,5074'BGNB
1210 DATA 4,8,1,11,7,12,3,11,7,2,11, ,5077'BGTC
1220 DATA 11,2,1,11,7,12,3,11,6,2,1,5,5072'BHHD
1230 DATA 17,25,3,201,3,169,3,201,1,249,1,25,5898'BOHG
1240 DATA 4,201,17,25,1,28,7,25,3,249,3,167,5730'BNLG
1250 DATA 3,201,1,249,1,25,4,201,13,25,2,30,5755'BNOH
1260 DATA 10,25,3,249,3,199,3,201,1,249,1,25,5969'BOOJ
1270 DATA 4,201,9,25,4,16,2,30,2,29,1,28,5351'BKWJ
1280 DATA 4,25,1,18,2,25,3,249,1,192,2,199,5721'BMNK
1290 DATA 3,201,1,249,1,25,4,201,8,25,1,137,5856'BNCL
1300 DATA 2,128,2,16,2,30,2,29,5,25,1,22,5264'BKTD
1310 DATA 2,25,1,201,1,255,1,249,6,201,1,249,6192'BOWE
1320 DATA 1,25,4,201,7,9,1,25,5,16,1,30,5325'BJXE
1330 DATA 1,174,2,29,5,25,2,22,1,25,1,47,5334'BKYG
1340 DATA 1,41,4,201,1,192,2,201,1,249,5,201,6099'BOQH
1350 DATA 7,9,1,25,3,16,4,25,2,29,4,25,5150'BIKH
1360 DATA 1,16,1,27,1,22,1,25,2,41,3,201,5341'BKCJ
1370 DATA 2,192,2,201,1,249,5,201,7,9,1,25,5895'BMKK
1380 DATA 8,242,6,25,2,27,1,25,1,32,2,41,5412'BKSL
1390 DATA 1,32,5,201,1,249,5,25,7,9,1,25,5561'BKDM
1400 DATA 8,242,2,25,9,16,2,32,1,240,5,249,5831'BMIE
1410 DATA 5,25,7,9,1,25,8,242,2,25,9,16,5374'BJTF
1420 DATA 3,240,5,249,5,25,7,9,1,25,8,242,5819'BLWG
1430 DATA 2,25,9,16,2,240,2,242,1,162,2,169,5872'BNHH
1440 DATA 1,249,1,25,8,137,3,9,1,25,8,242,5709'BLSI
```

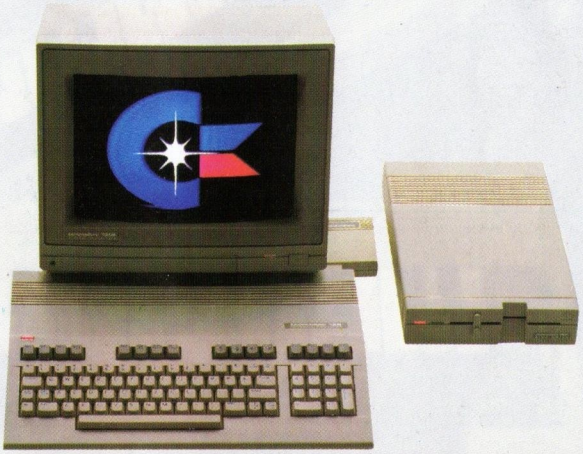
```
1450 DATA 2,137,4,128,3,16,4,240,3,242,1,2,5782'BMAJ
1460 DATA 1,18,1,25,1, ,8,137,3,9,1,25,5229'BHIJ
1470 DATA 6,249,4,137,4,128,3,16,6,240,1,2,5796'BMTL
1480 DATA 1, ,2,241,1,1,8,137,3,9,1,25,5429'BHDL
1490 DATA 6,249,4,137,4,128,3,16,3,240,1, ,5791'BLLN
1500 DATA 1,240,1,251,2,250,1,10,1,252,1,28,6038'BNEF
1510 DATA 5,139,3,137,3,9,1,25,1,252,2,242,5819'BMLG
1520 DATA 3,252,1,140,3,137,4,128,3,16,3,240,5930'BOUH
1530 DATA 1, ,1,240,1,251,4,250,1,32,4,27,5812'BKII
1540 DATA 1,11,3,137,3,9,1,25,1,252,1,162,5606'BLRJ
1550 DATA 4,252,1,188,1,185,2,25,7,16,3,240,5924'BNMK
1560 DATA 1, ,1,252,1,251,2,250,2,42,1,32,5835'BKJL
1570 DATA 3,27,2,139,3,137,4,25,1,188,1,194,5724'BNRM
1580 DATA 5,188,1,185,2,25,10,16,1, ,1,140,5574'BLAN
1590 DATA 3,139,3,43,2,27,2,139,1,27,3,137,5526'BMKO
1600 DATA 4,25,7,184,3,25,1,16,7,240,2,16,5530'BLCG
1610 DATA 1, ,1,139,1,128,3,139,2,27,4,139,5584'BLLH
1620 DATA 1,27,3,137,4,25,7,185,3,25,1,16,5434'BLII
1630 DATA 8,240,1,160,1,168,7,139,3,171,2,27,5927'BOLK
1640 DATA 3,128,4,16,6,185,1,201,2,98,1,242,5887'BNTK
1650 DATA 1,16,8,240,1,160,1,165,1,27,4,11,5635'BMTL
1660 DATA 1,128,1,139,3,171,2,27,3,128,4,16,5623'BNEM
1670 DATA 6,176,1,192,1,194,2,98,10,16,1, ,5697'BLVN
1680 DATA 1,16,6,176,3,171,2,187,1,176,2,128,5869'BOVP
1690 DATA 4,16,2,176,1,16,3,176,1,192,1,194,5782'BNQP
1700 DATA 2,98,10,16,1, ,1,16,6,176,2,187,5515
1710 DATA 4,176,2,128,4,16,3,176,19,32,1,128,5689'BOVJ
1720 DATA 11,176,6,32,3,176,19,32,12,176,6,32,5681'BPLK
1730 DATA 4,16,18,32,12,176,6,32,1,168,1,170,5636'BOFL
1740 DATA 2,168,3,160,1,168,1,21,1,31,1,61,5618'BMVL
1750 DATA 2,63,3,31,2,85,1,215,1,87,1,247,5738'BLNM
1760 DATA 1,95,1,85,1,255,2,95,1,253,1,245,6035'BMLN
1770 DATA 2,85,1,213,1,247,1,215,1,95,2,127,5990'BNJO
1780 DATA 1,119,2,87,1,247,1,117,1,253,1,255,6085'BODQ
1790 DATA 1,223,1,215,2,223,1,221,3,245,1,253,6389'BPMR
1800 DATA 1,93,1,213,1,117,1,253,1,85,2,87,5855'BMLI
1810 DATA 5,85,1,93,1,247,1,127,1,117,1,93,5772'BMQJ
1820 DATA 1,119,2,85,1,231,1,245,1,237,1,247,6171'BOAK
1830 DATA 1,237,1,231,1,229,1,231,1,245,1,119,6298'BPTM
1840 DATA 1,93,1,127,1,245,1,85,1,213,1,85,5854'BMJM
1850 DATA 1,255,1,117,1,87,1,117,4,85,2,213,5884'BNNN
1860 DATA 8,85,1,87,5,85,1,124,1,223,1,117,5738'BMUO
1870 DATA 1,87,1,117,3,85,1, ,1,244,1,93,5634'BJIP
1880 DATA 1,125,1,213,1,117,1,93,1,117,1, ,5671'BLMQ
1890 DATA 1,64,1,16,1,124,1,252,3,124,4,15,5606'BMPR
1900 DATA 1,3,3, ,6,255,1,127,1,63,1,253,5714'BJCI
1910 DATA 1,247,1,213,4,223,3,255,1,223,1,255,6427'BPQL
1920 DATA 3,223,1,222,1, ,2,192,2,240,2,252,6140'BMDL
1930 DATA 1,101,6, ,1,192,9, ,5,40,2,10,5367'BHVL
1940 DATA 1,2,8, ,1,51,1,204,1,51,1,204,5525'BILM
1950 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDO
1960 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDP
1970 DATA 1, ,1,204,1,51,1,204,1,51,1,204,5720'BKVQ
1980 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDR
1990 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDS
2000 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDB
2010 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDC
2020 DATA 1,51,1,204,1,3,1,204,1,3,1,204,5675'BKED
2030 DATA 1,3,1,204,1, ,1,192,1,32,1,204,5641'BJHD
2040 DATA 1,51,1,204,1,51,1,252,1,32,1, ,5596'BJQE
2050 DATA 1,3,1,204,1,51,1,207,1,255,3, ,5728'BJTF
2060 DATA 1,51,1,204,1,51,1,255,4, ,1,51,5621'BJMG
2070 DATA 1,204,1,255,1,192,4, ,1,255,1,207,6122'BMMI
2080 DATA 1,48,4, ,1,63,2,252,4, ,1,15,5391'BGBI
2090 DATA 1,252,1,3,5, ,1,255,1, ,1,168,5688'BHIJ
2100 DATA 2,40,1,42,1,40,1,42,2,168,1,31,5371'BKIC
2110 DATA 1,23,1,31,3,30,1,23,1,29,1,255,5399'BKTD
2120 DATA 1,253,1,181,2,165,1,233,1,218,1,234,6291'BPOF
2130 DATA 2,93,1,95,1,87,1,119,1,247,1,187,5835'BMAF
2830 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDM
2840 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDN
2850 DATA 1,204,1,127,1,76,1,127,1,76,1,127,5743'BNBO
2860 DATA 1,76,1,127,1,79,1,255,1,207,1,51,5801'BMCP
2870 DATA 1,207,1,51,1,204,1,63,1,207,1,245,5983'BNUQ
2880 DATA 1,253,1,255,5,253,1,51,1,204,1,51,6077'BNTR
2890 DATA 1,204,1,51,1,204,1,51,1,204,1,3,5723'BLYS
2900 DATA 1,192,1,3,1,192,1,3,1,192,1,3,5591'BJYJ
2910 DATA 1,192,1,47,1,239,1,47,1,239,1,47,5817'BMQL
2920 DATA 1,239,1,47,1,239,48,255,1,128,1,1,5962'BNQM
2930 DATA 1,7,1,16,1,64,1,85,2,255,1,26,5460'BJCN
2940 DATA 1,85,1,253,1,6,1,26,1,74,1,5,5455'BIIN
2950 DATA 1,51,1,128,1,96,1,128,1,160,1,136,5705'BNWP
2960 DATA 1,34,1,72,1,63,2,3,5,1,1,255,5439'BIDP
2970 DATA 8,85,2,195,1,227,1,195,4,231,7,85,6041'BNVR
2980 DATA 1,80,1,105,1,101,6,105,2,89,2,85,5578'BMGS
2990 DATA 4,89,4,229,1,237,1,253,1,237,1,239,6296'BOUU
3000 DATA 24,85,1,125,7,85,1,204,1,52,1,245,5831'BNGC
3010 DATA 1,223,1,82,1,94,1,86,1,84,1,112,5687'BLKD
3020 DATA 1,207,1,21,1,245,1,165,1,181,1,149,5974'BOCE
3030 DATA 1,53,1,124,1,252,6,124,2,15,1,13,5593'BMQF
3040 DATA 5,15,16,255,1,112,1,220,1,124,5,255,6010'BPBH
3050 DATA 1,63,2,15,2,3,2,192,1,240,7,255,5783'BLEH
3060 DATA 1,63,3,255,5,247,8,234,1, ,2,192,6011'BLBI
3070 DATA 2,240,2,244,1,223,1,51,1,204,1,51,6021'BNWJ
3080 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDK
3090 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDL
3100 DATA 1,204,2,127,1,124,2,127,1,124,1,127,5841'BPFE
3110 DATA 1,124,1,255,1,243,1,204,1,51,1,60,5943'BNME
3120 DATA 1,207,1,51,1,204,1,61,7,253,1,51,5839'BMWF
3130 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDG
3140 DATA 1,204,1,3,1,192,1,3,1,192,1,3,5603'BJMG
3150 DATA 1,192,1,3,1,192,1,47,1,239,1,47,5726'BLHI
3160 DATA 1,239,1,47,1,239,1,47,1,239,3,255,6074'BNPJ
3170 DATA 5,247,19,255,5,85,3,255,5,223,11,255,6368'BQYL
3180 DATA 1, ,1,34,1,8,1,34,1,8,1,60,5150'BFOK
3190 DATA 1,51,1,63,1, ,1,34,1,136,1,34,5324'BIOL
3200 DATA 1,136,1, ,2,42,2, ,1,128,1, ,5314'BFJD
3210 DATA 1,128,1, ,1,171,1,169,5, ,3,85,5565'BIME
3220 DATA 1,23,4, ,4,231,1,63,3, ,1,68,5399'BGIF
3230 DATA 1,16,1,68,1,16,1,68,1,26,1,2,5202'BIWG
3240 DATA 1, ,4,41,1,42,1,41,1,42,1, ,5175'BFKH
3250 DATA 1,155,1,91,1,155,1,153,1,91,1,123,5774'BNAJ
3260 DATA 1,169,1,171,1,237,1,239,1,253,1,247,6322'BPWL
3270 DATA 2,229,1,247,1,237,2,117,1,85,1,117,6040'BOYL
3280 DATA 2,245,1,125,1,215,7,85,1,87,16,85,5870'BNVM
3290 DATA 1,92,1,112,3,74,1,72,1,74,1,72,5504'BKWN
3300 DATA 1,61,1,205,2,137,1,9,1,137,1,33,5589'BLDF
3310 DATA 1,137,8,124,7,15,1,42,7,255,1,79,5677'BMYG
3320 DATA 6,223,1,95,1,223,8,255,1,240,2,252,6307'BOCH
3330 DATA 5,255,1,63,2,15,2,3,2,192,1,242,5783'BLEI
3340 DATA 5,247,2,255,1,63,8,234,4,255,1,247,6322'BOJK
3350 DATA 1,223,1,247,1,255,1,51,1,204,1,51,6037'BNLK
3360 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDL
3370 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDM
3380 DATA 1,204,1,127,1,124,1,127,1,124,1,127,5839'BPKO
3390 DATA 1,124,2,127,1,51,1,252,1,63,1,207,5831'BNN0
3400 DATA 1,51,1,204,1,51,1,252,1,61,1,253,5878'BMTG
3410 DATA 1,61,1,253,1,61,1,253,1,61,1,253,5948'BMYH
3420 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDI
3430 DATA 1,51,1,204,1,3,1,192,1,3,1,192,5651'BKKJ
3440 DATA 1,3,1,192,1,3,1,192,1,47,1,239,5682'BKCK
3450 DATA 1,47,1,239,1,47,1,239,1,47,1,239,5864'BMUL
3460 DATA 8,247,16,255,8,85,8,223,8,255,1,34,6148'BOEN
3470 DATA 1,8,6,255,1,34,1,191,6,255,1, ,5759'BJPN
3480 DATA 1,252,1,239,5,255,2, ,1,250,5,255,6266'BMFO
3490 DATA 2, ,1,168,1,250,4,255,3, ,1,168,5853'BJFP
3500 DATA 1,250,3,255,5, ,1,250,2,255,6, ,6028'BJAG
3510 DATA 1,192,1,255,7, ,1,192,1,85,6, ,5741'BINH
4210 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDG
4220 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDH
4230 DATA 1,204,1,51,1,204,1,3,1,192,1,3,5663'BKHI
4240 DATA 1,192,1,3,1,192,1,3,1,192,1,15,5603'BKNJ
4250 DATA 1,207,1,15,1,207,1,15,1,207,1,15,5672'BMMK
4260 DATA 1,207,8,247,40,255,2,207,1, ,1,207,6176'BNDL
4270 DATA 1, ,1,207,1,204,1,207,2,237,1,45,5907'BLQM
4280 DATA 1,221,1,29,3,221,1, ,7,60,6,46,5596'BJEN
4290 DATA 2,42,1,243,1,3,1,243,1,3,1,243,5784'BKPO
4300 DATA 1,240,1,255,1,240,1,254,1,250,1,248,6493'BPNH
4310 DATA 2,240,1, ,1,240,1, ,1,255,5,63,5809'BIVG
4320 DATA 2,15,8,255,1,252,1,255,1,252,1,255,6298'BOGJ
4330 DATA 1,252,1,255,1,252,1,255,1,206,1,48,6274'BOYJ
4340 DATA 1,206,1,50,1,206,1,50,1,206,1,50,5774'BMJK
4350 DATA 43,170,1,174,4,175,4,170,4,250,1,162,6158'BQAM
4360 DATA 1,136,1,162,1,136,1,162,1,136,1,162,5900'BPGN
4370 DATA 1,136,1,34,1,136,1,32,1,136,1,32,5512'BMHN
4380 DATA 1,128,1,32,1,128,8, ,4,63,2,15,5383'BJBO
4390 DATA 2,3,8,255,2,240,3,252,2,195,1,243,6206'BNAP
4400 DATA 1,15,2,3,1, ,1,3,1,15,2,63,5107'BFJG
4410 DATA 8,234,1,240,2,252,5,255,1,51,1,204,6254'BOUI
4420 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDJ
4430 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDK
4440 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDL
4450 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDM
4460 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDN
4470 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDO
4480 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDP
4490 DATA 1,51,1,204,1,51,1,204,1,3,1,192,5711'BLCQ
4500 DATA 1,3,1,192,1,3,1,192,1,3,1,192,5591'BJYH
4510 DATA 1,15,1,207,1,15,1,207,1,15,1,207,5672'BMMJ
4520 DATA 1,15,1,207,7,247,41,255,2,207,1, ,5984'BMIK
4530 DATA 1,207,1, ,2,207,1,255,2,221,1,29,5927'BLUL
4540 DATA 1,221,1,29,1,221,1,253,1,221,3,60,6013'BNYM
4550 DATA 1, ,1,60,1, ,2,60,8,42,8,240,5423'BGSM
4560 DATA 1,14,4,12,4,15,2,143,3,131,1,143,5473'BMKO
4570 DATA 1,191,8,255,1,252,1,255,1,252,1,255,6473'BPCQ
4580 DATA 1,252,1,255,1,252,1,255,1,206,1,50,6276'BOTQ
4590 DATA 1,206,1,50,1,206,1,50,1,206,1,50,5774'BMJR
4600 DATA 40,170,4,175,3,174,1,170,6,250,2,170,6165'BQVK
4610 DATA 1,162,1,136,1,162,1,136,1,162,1,136,5900'BPGL
4620 DATA 1,162,1,136,1, ,1,128,11, ,1,85,5527'BJWK
4630 DATA 1,64,1,127,5, ,1,85,1,1,1,85,5372'BHGL
4640 DATA 1,255,1,252,2,63,2,15,2,2,1,3,5599'BJBN
4650 DATA 1, ,1,240,1,244,1,234,1,164,1,144,6032'BMDO
4660 DATA 1,145,1,12,1, ,1,4,1,124,1,154,5445'BJLO
4670 DATA 1,166,1,105,1,106,3,234,1,42,1,194,5855'BOXQ
4680 DATA 1,170,1,90,1,117,8,255,1,51,1,204,5900'BNQR
4690 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDS
4700 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDK
4710 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDL
4720 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDM
4730 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDN
4740 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDO
4750 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDP
4760 DATA 1,51,1,204,1,51,1,204,1,3,1,192,5711'BLCQ
4770 DATA 1,3,1,192,1,3,1,192,1,3,1,204,5603'BJMQ
4780 DATA 1,15,1,207,1,15,1,207,1,15,1,207,5672'BMMS
4790 DATA 1,15,1,207,48,255,1, ,1,192,2, ,5723'BJCS
4800 DATA 1,63,1,207,1,63,1,207,1,1,3, ,5549'BIXK
4810 DATA 1,192,3,255,1,60,4, ,1,252,2,255,6026'BLUM
4820 DATA 2,42,4, ,3,255,1,240,4, ,1,192,5744'BIBM
4830 DATA 2,255,2,15,4, ,1,255,1,240,1,192,5968'BLFO
4840 DATA 1,203,1,15,4, ,4,255,1,15,1,13,5513'BJMO
4850 DATA 1,7,1,28,1,252,1,255,1,253,1,211,6012'BMIQ
4860 DATA 1,127,1,255,1,252,1,255,1,14,1,21,5930'BNNR
4870 DATA 2,255,1,204,1,255,1,204,1,48,1,170,6143'BOOS
4880 DATA 1,106,1,214,1,249,1,58,1,130,1,160,5923'BOPT
4890 DATA 1,234,3,170,1,42,1,74,1,150,1,165,5843'BNTU
5590 DATA 1,240,1,44,1,11,7, ,1,15,1, ,5322'BGFR
5600 DATA 1,3,4, ,1,63,1,207,1,192,1,48,5522'BIXJ
5610 DATA 1,192,3, ,1,240,1,255,1,128,7, ,5829'BJCL
5620 DATA 1,60,5,12,1,3,1,12,13,255,3,85,5451'BKPM
5630 DATA 2,255,6,85,2,255,6,95,8,255,1,51,6021'BMTN
5640 DATA 1,204,1,51,1,92,1,87,1,165,2,169,5775'BMNO
5650 DATA 1,251,2,239,2,78,1,126,1,50,1,130,5882'BNAP
5660 DATA 1,191,4,175,1,165,1,5,1,185,6,255,5990'BNSQ
5670 DATA 1,127,1,159,16,255,1,240,1,243,1,252,6297'BQYS
5680 DATA 1,243,1,252,1,243,2,252,1,192,1, ,6189'BMUS
5690 DATA 1,195,1,15,3,207,1,15,1,12,1,255,5707'BMTT
5700 DATA 1,252,1,255,1,252,1,255,1,252,1,255,6527'BPSM
5710 DATA 1,204,1,48,1,204,1,48,1,204,1,48,5762'BMVM
5720 DATA 1,204,1,48,1,255,1,234,1,254,1,255,6256'BOYN
5730 DATA 1,254,1,235,1,254,1,175,1, ,1,160,6084'BMRO
5740 DATA 1,168,4,170,1,168,1,34,1,42,1,10,5601'BMOP
5750 DATA 1,34,1,10,1,42,1,138,1,10,32,170,5441'BMAQ
5760 DATA 8,128,1,51,1,192,1,51,1,192,1,63,5690'BMFR
5770 DATA 1,207,1,48,1,192,1,51,1,204,1,51,5759'BMYS
5780 DATA 1, ,2,255,1, ,1,204,1,55,1,204,5725'BIPS
5790 DATA 1,51,1, ,2,255,1, ,1,204,1,51,5568'BHST
5800 DATA 1,204,1,51,1, ,2,255,1, ,1,204,5721'BIHL
5810 DATA 1,51,1,192,1,51,1, ,1,255,1,252,5807'BKQN
5820 DATA 1,3,1,192,1, ,1,192,1, ,1,192,5585'BHAN
5830 DATA 1, ,1,192,1, ,1,192,1, ,2,32,5423'BFLO
5840 DATA 1,8,1,5,1,6,1,8,1,2,1,160,5195'BFXP
5850 DATA 1,128,1,160,1,168,1,32,1,136,1,34,5664'BNVR
5860 DATA 1,160,1,2,2,8,2, ,2,3,1,15,5197'BFSR
5870 DATA 1,214,1,232,1,160,1,128,1,32,1,128,5900'BOFT
5880 DATA 2,32,1,192,1,8,5, ,1,136,1,34,5413'BIVT
5890 DATA 1,8,1,2,1,8,1,34,1, ,1,32,5090'BETT
5900 DATA 1,136,1,51,1,204,1,51,1,140,1,35,5623'BMEN
5910 DATA 1,8,1,3,1,8,1,51,1,204,1,51,5331'BHMN
5920 DATA 1,204,1,51,1,204,1,51,1,204,1,3,5723'BLYP
5930 DATA 1,192,1,3,1,192,1,3,1,192,1,3,5591'BJYP
5940 DATA 1,192,1, ,1,240,1,12,1,207,1,15,5672'BKHR
5950 DATA 1,207,1,15,1,207,1,3,2,12,1,204,5655'BLHS
5960 DATA 1,76,1, ,1,76,1,60,8,255,16,85,5580'BJRT
5970 DATA 5,95,2,255,1,95,8,255,1,169,1,90,5977'BMOU
5980 DATA 1,106,1,166,1,150,2,90,1,152,1,82,5753'BNTV
5990 DATA 2,216,1,88,4,168,1,121,1,125,1,173,5901'BOAW
6000 DATA 1,169,1,165,1,153,1,106,1,169,1,167,5935'BPGG
6010 DATA 1,171,1,105,1,154,1,102,1,89,1,86,5713'BNUG
6020 DATA 1,106,4,255,1,191,1,93,1,165,1,86,5905'BNKH
6030 DATA 4,255,2,95,1,159,1,171,3,252,1,255,6199'BOPJ
6040 DATA 1,252,3,255,1,207,1,63,1,207,1,63,6055'BNUJ
6050 DATA 1,195,1,51,1,115,1,51,1,252,1,255,5925'BNVK
6060 DATA 1,252,2,255,1,240,1,192,1, ,1,204,6150'BMFL
6070 DATA 1,48,1,204,1,255,4, ,1,170,1,174,5860'BLVM
6080 DATA 1,250,1,254,1,63,3, ,4,168,2,170,5917'BLCN
6090 DATA 1,42,1,8,1,138,1,10,2,138,1,170,5513'BLLO
6100 DATA 1,10,1,170,1,42,32,170,8,128,1,51,5615'BNNG
6110 DATA 1,192,1,51,1,192,1,51,1,192,1,51,5735'BMVH
6120 DATA 1,192,1,51,1,204,1,51,1,205,1,51,5760'BMII
6130 DATA 1,204,1,51,1, ,1,51,1,204,1,87,5603'BJMI
6140 DATA 1,85,1,55,1,204,1,51,1,12,1,51,5464'BKOK
6150 DATA 1,204,1,51,1,76,1,51,1,204,1,51,5643'BLKL
6160 DATA 1,204,1,51,1,192,1,51,1,192,1,51,5747'BMSM
6170 DATA 1,192,1,51,1,192,1, ,1,192,1, ,5633'BIUM
6180 DATA 1,192,1, ,1,192,1, ,1,192,8, ,5589'BGKN
6190 DATA 1,136,1,162,1,160,1, ,2,15,1,13,5493'BKJP
6200 DATA 1,61,1,47,1,32,1,144,1,20,1,212,5522'BLCH
6210 DATA 3,84,1,96,7,112,1,34,1,136,1,34,5510'BLXI
6220 DATA 1,136,1,34,1,136,1,34,1,136,1,32,5514'BMLJ
6230 DATA 1,136,1,32,1,136,1,34,1,136,1,34,5514'BMLK
6240 DATA 1,136,1,2,1,8,1,2,1,8,1,32,5194'BGTK
6250 DATA 1,136,1,34,1,136,1,51,1,140,1,21,5524'BMEM
6260 DATA 2,48,1,12,1,131,1, ,1,3,1,252,5453'BILM
6270 DATA 1,87,4, ,1,192,1,15,1,207,1,255,5765'BKGO
```

```
2140 DATA 1,165,1,255,1,95,1,117,2,255,1,223,6117'BOAG
2150 DATA 1,247,1,213,1,117,1,213,2,85,2,213,6096'BOTH
2160 DATA 1,245,1,85,1,255,1,127,1,223,3,245,6188'BOCJ
2170 DATA 1,117,4,85,1,213,12,85,6,229,2,245,6000'BOAJ
2180 DATA 40,85,2,93,1,87,5,85,2,252,2,124,5778'BMDK
2190 DATA 1,252,3,124,2, ,2,12,4,15,1,63,5479'BJAK
2200 DATA 2,15,2,3,1, ,2,192,7,255,1,63,5543'BIEC
2210 DATA 2,221,1,217,1,222,1,223,1,221,1,214,6325'BPSF
2220 DATA 1,223,1,253,1,108,1,121,1,100,1,93,5904'BOEF
2230 DATA 1,205,1,41,1,129,1,176,1,192,1,124,5873'BOXG
2240 DATA 1,224,1,144,1,108,1,84,1,103,8, ,5676'BLWH
2250 DATA 1,34,2,32,1,40,4,42,8, ,1,51,5216'BHQH
2260 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDJ
2270 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDK
2280 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDL
2290 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDM
2300 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDE
2310 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDF
2320 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDG
2330 DATA 1,204,1,51,1,204,1,51,1,204,1, ,5720'BKVH
2340 DATA 1,192,1, ,1,192,1,3,1,192,1,3,5588'BIEH
2350 DATA 1,192,2, ,1,255,1,192,1,47,1,239,5932'BLFJ
2360 DATA 1,47,1,239,1, ,1,15,1,252,1, ,5559'BHCJ
2370 DATA 4,255,1,3,1,255,1, ,6,255,1, ,5782'BHHK
2380 DATA 1,63,5,255,1,240,1,3,6,255,1, ,5831'BJCL
2390 DATA 15,255,1,168,3,160,2,168,2,40,1,16,5831'BOEN
2400 DATA 1,32,1,56,1,28,1,29,1,20,1,21,5192'BJSE
2410 DATA 1,16,1,90,1,65,1,2,1,70,1,2,5251'BHOF
2420 DATA 1,3,1,4,1,1,1,181,1,198,1,190,5583'BJBG
2430 DATA 1,243,1,255,1,140,1,60,1,3,1,229,5936'BMSI
2440 DATA 2,181,1,95,1,111,1,95,1,23,1,21,5533'BLPJ
2450 DATA 1,105,1,125,2,105,4,109,14,85,2,86,5639'BOCL
2460 DATA 6,85,2,149,8,245,30,85,1,86,1,90,5788'BMOL
2470 DATA 4,85,1,117,1,237,1,185,1,169,8,85,5894'BNIM
2480 DATA 8,124,8,15,1,240,2,252,5,255,1,63,5974'BNNN
2490 DATA 2,15,2,3,2,192,1,240,7,223,1,29,5717'BLXO
2500 DATA 1,4,1,128,1,161,1,239,1,231,1,237,6006'BNMG
2510 DATA 1,103,1,239,1,63,1,79,1,191,1,79,5760'BMKH
2520 DATA 1,63,1,15,1,63,1,255,1, ,2,192,5595'BJCI
2530 DATA 2,240,2,244,1,247,1,42,2,10,1, ,5792'BKNJ
2540 DATA 1,42,2,10,1,2,8, ,1,51,1,204,5323'BHFJ
2550 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDL
2560 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDM
2570 DATA 1,51,1,204,1,51,1,204,1,255,1,84,5855'BMSN
2580 DATA 1,117,1,76,1,127,1,79,1,51,1,204,5660'BMYO
2590 DATA 1,51,1,204,1,115,1,84,1,53,1,252,5765'BMSP
2600 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDH
2610 DATA 1,115,1,84,1,51,1,204,1,51,1,204,5715'BMII
2620 DATA 1,51,1,204,1,51,1,204,1,3,1,192,5711'BLCJ
2630 DATA 1,3,1,192,1,3,1,192,1,3,1,192,5591'BJYJ
2640 DATA 1,47,1,239,1,47,1,239,1,47,1,239,5864'BMUL
2650 DATA 1,47,1,239,48,255,1,168,1,162,1,168,6092'BPON
2660 DATA 1,170,1,40,1,42,1,40,1,34,1, ,5332'BIDM
26 '0 DATA 1,6,4,26,1,24,1,18,1,65,1, ,5148'BGBN
2680 DATA 1,136,1,32,1,136,1,128,1, ,1,170,5608'BLMP
2690 DATA 1,16,1,1,5, ,1,168,1,85,1,37,5317'BHEP
2700 DATA 1,101,1,169,1,105,3,85,3,109,1,101,5680'BOPI
2710 DATA 1,171,1,234,2,230,8,85,3,86,1,90,5912'BMIJ
2720 DATA 1,106,1,101,1,105,1,102,3,85,1,101,5608'BOWK
2730 DATA 1,153,3,89,5,245,3,229,24,85,1,106,5944'BOSM
27- 0 DATA 1,86,2,90,1,85,1,87,1,85,1,95,5535'BJAM
2750 DATA 1,170,1,162,1,168,1,97,1,216,1,49,5868'BNRN
2760 DATA 1,209,1,61,1,149,3,85,1,213,1,21,5746'BMAO
2770 DATA 1,7,1,253,8,124,8,15,8,255,1,240,5921'BMIP
2780 DATA 2,252,5,255,1,29,2,15,2,3,2,192,5760'BLDQ
2790 DATA 1,240,2,238,1,230,1,251,1,243,1,255,6464'BPJS
2800 DATA 1,85,2,63,3,255,1,127,1,223,1,87,5849'BMRJ
2810 DATA 1,255,5,247,3,255,1,2,2,192,2,224,6189'BNLK
2820 DATA 2,232,1,234,8, ,1,51,1,204,1,51,5786'BKTL

3520 DATA 1,160,1,95,1,90,1,127,5, ,1,221,5703'BKPJ
3530 DATA 1,170,2,255,4, ,1,85,1,138,3,254,5914'BLFK
3540 DATA 3, ,1,85,1,149,1,165,1,169,1,138,5714'BLKL
3550 DATA 1,170,1,42,1, ,2,72,1,74,1,72,5437'BIYL
3560 DATA 1,106,1,232,2,250,1,33,1,161,1,33,5822'BNEN
3570 DATA 1,161,1,33,1,161,1,35,1,139,1,124,5659'BNUO
3580 DATA 1,252,3,124,1,252,1,92,1,252,1,74,6054'BNSP
3590 DATA 1,22,1,21,4,61,1,5,1,63,1,159,5340'BJRP
3600 DATA 1,111,1,79,1,15,1,79,1,15,1,79,5384'BKJI
3610 DATA 1,215,1,223,2,255,1,63,2,15,1,3,5782'BLSJ
3620 DATA 9,255,7,223,1,240,2,252,5,255,1,63,6313'BOAL
3630 DATA 2,15,2,3,2,192,1,240,7,234,1,42,5741'BLRL
3640 DATA 8,255,1,51,1,204,1,51,1,204,1,51,5829'BMUM
3650 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDN
3660 DATA 1,204,1,51,1,204,1,51,1,204,1,127,5847'BNJO
3670 DATA 1,85,1,49,1,204,1,51,1,204,1,51,5650'BLPP
3680 DATA 1,204,2,255,1,87,1,197,1,51,1,204,6005'BNTQ
3690 DATA 1,51,1,204,3,253,1,93,1,21,1,204,5834'BMMR
3700 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDJ
3710 DATA 1,51,1,204,1,51,1,204,1,3,1,192,5711'BLCK
3720 DATA 1,3,1,192,1,3,1,192,1,3,1,192,5591'BJYK
3730 DATA 1,47,1,239,1,47,1,239,1,47,1,239,5864'BMUM
3740 DATA 1,47,1,239,8,247,16,255,8,85,8,223,6138'BOCO
3750 DATA 8,255,1,51,1,204,1,51,1,204,1,51,5829'BMUO
3760 DATA 1,204,1,48,6,255,3,253,6,255,2,252,6286'BOJQ
3770 DATA 6,255,2,63,40,255,1,192,7,255,2, ,6078'BMQQ
3780 DATA 1,252,5,255,3, ,1,240,4,255,4, ,6020'BJSQ
3790 DATA 1,240,3,255,5, ,1,192,2,255,1,10,5965'BLYS
3800 DATA 5, ,1,128,1,252,1,185,1,8,6, ,5588'BHPJ
3810 DATA 1,92,2,30,5, ,1,29,1,52,1,28,5242'BHXK
3820 DATA 1,52,1,29,1,170,2, ,1,15,1,79,5352'BIAL
3830 DATA 1,15,1,79,1,31,1,175,1,171,1,43,5520'BLQN
3840 DATA 1,27,1,26,1,22,1,4,3, ,1,4,5091'BFNN
3850 DATA 3,255,5,63,3,223,1,221,1,125,1,221,6122'BOIP
3860 DATA 1,245,1,253,6,255,2,127,1,240,2,252,6385'BPWR
3870 DATA 5,255,1, ,4,42,3,234,7,255,1,63,5870'BKJR
3880 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDS
3890 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDT
3900 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDL
3910 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDM
3920 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDN
3930 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDO
3940 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDP
3950 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDQ
3960 DATA 1,3,1,192,1,3,1,192,1,3,1,192,5591'BJYQ
3970 DATA 1,3,1,192,1,47,1,239,1,47,1,239,5773'BLLS
3980 DATA 1,47,1,207,1,15,1,207,8,247,16,255,6006'BODT
3990 DATA 4,85,4,255,4,223,12,255,2,207,1, ,6052'BMAU
4000 DATA 1,207,1, ,1,207,1,204,1,207,1,253,6084'BMID
4010 DATA 1,221,1,61,1,221,1,29,2,221,1,237,5997'BNVE
4020 DATA 2,252,1, ,1,60,2,252,1, ,1,252,5824'BIPE
4030 DATA 2,63,1,58,3,42,1,46,1,38,1,255,5511'BKCG
4040 DATA 2,240,1,255,1, ,1,240,2,243,1,255,6241'BMHH
4050 DATA 2,15,1,255,1,15,1,14,1,250,1,251,5807'BMMI
4060 DATA 16,255,1,252,1,255,1,252,1,255,1,252,6542'BQTK
4070 DATA 1,255,1,252,1,255,1,15,1,51,1,206,6040'BNIK
4080 DATA 1,50,1,206,1,50,1,206,1,50,32,170,5769'BNPL
4090 DATA 1,175,7,170,1,252,1,255,6,170,1, ,6039'BMBM
4100 DATA 1,240,1,170,1,171,4,170,2, ,1,128,5889'BMXE
4110 DATA 1,191,2,170,1,162,1,136,3, ,1,32,5700'BLHF
4120 DATA 1,170,1,168,1,34,1,136,2,16,1,20,5551'BMNG
4130 DATA 2,4,3, ,8,63,1,253,13,255,1,240,5843'BKAH
4140 DATA 1,192,5,255,3,63,8,234,1,63,2,15,5842'BMMI
4150 DATA 2,3,2,192,1,240,1,51,1,204,1,51,5749'BLPJ
4160 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDK
4170 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDL
4180 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDM
4190 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDN
4200 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDF

4900 DATA 1,40,40,170,1,160,1,128,1,160,1,128,5831'BPDN
4910 DATA 1,160,3,128,8, ,7,115,1,67,4,85,5579'BKVN
4920 DATA 2,53,2,61,1,2,6,10,1, ,1,81,5220'BGIN
4930 DATA 1,17,1,65,1,17,2,85,1,149,1,17,5357'BKJP
4940 DATA 2,90,1,74,3,90,2,106,1,26,1,134,5530'BLSQ
4950 DATA 1,146,1,134,1,146,1,134,1,146,1,166,5878'BPAS
4960 DATA 5,255,1,253,1,246,1,221,1,51,1,204,6240'BOKS
4970 DATA 1,51,1,204,1,51,1,220,1,119,1,93,5744'BMMT
4980 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDU
4990 DATA 1,19,1,68,1,51,1,204,1,51,1,204,5603'BLLV
5000 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDE
5010 DATA 1,51,1,204,1,49,1,196,1,55,1,223,5784'BMEF
5020 DATA 1,51,1,204,1, ,1,79,1,63,1,252,5655'BJBF
5030 DATA 1,243,1,204,1,16,1,63,2,255,1,243,6031'BNIH
5040 DATA 1,204,1,51,1,204,1,20,1,253,2,255,5994'BNPI
5050 DATA 1,63,1,192,1,3,1,192,1,63,1,79,5598'BKOJ
5060 DATA 1,31,1,195,1,244,1,252,1,60,1,207,5995'BNBK
5070 DATA 5,255,2,63,1,15,40,255,1,51,1,204,5893'BNEL
5080 DATA 1,51,1,204,1,51,1,204,1,51,1,204,5771'BMDM
5090 DATA 25,255,1,192,6,204,1,12,1,63,9,255,6024'BODN
5100 DATA 2,252,1,255,2,243,1,255,1,115,1,243,6371'BPMG
5110 DATA 2,207,2,63,1,60,1,252,1,255,1,252,6097'BNXG
5120 DATA 2,255,1,245,1,64,1, ,1,204,1,48,5823'BKVH
5130 DATA 1,204,1,62,1,84,1,5,2, ,3,234,5598'BIGH
5140 DATA 2,170,1,80,1,5,1, ,2,138,1,162,5563'BJWI
5150 DATA 3,168,1,8,1,82,1,42,1,170,1,154,5632'BLYK
5160 DATA 1,138,2,130,1,162,1,34,32,170,8,128,5807'BPSM
5170 DATA 2, ,1,5,2,1,1,255,1, ,1,255,5524'BFPL
5180 DATA 2,106,1,42,1,89,1,90,1,255,1, ,5589'BJIN
5190 DATA 1,255,1,58,1,63,1,252,1,80,1,90,5804'BLBO
5200 DATA 1,255,1, ,1,255,1,170,1,255,2, ,5942'BJYF
5210 DATA 1,170,1,255,1, ,1,255,1,162,1,255,6103'BMKH
5220 DATA 2, ,1,170,1,255,1, ,1,255,1,170,5857'BJAH
5230 DATA 1,149,1,213,1,21,1,140,1,240,1, ,5769'BLKJ
5240 DATA 1,192,1,34,1,86,1,87,1,95,1,171,5671'BLOK
5250 DATA 1,15,1,32,1, ,1,238,1,190,1,250,5731'BKIL
5260 DATA 1,255,1,206,1,63,2,255,1,185,1,167,6138'BOBN
5270 DATA 1,237,1,171,1,250,1,255,1,239,1,187,6345'BPXO
5280 DATA 1,17,1,132,1,241,1,244,1,189,1,236,6065'BOUO
5290 DATA 1,189,1,239,1,51,1,204,1,51,1,204,5944'BNXP
5300 DATA 1,51,1,204,1,51,1,204,1,31,1,223,5770'BMBH
5310 DATA 1,15,1,124,1,51,1,79,1,49,1,112,5436'BLTI
5320 DATA 1,51,1,204,1,51,1,207,1,241,1,84,5844'BMOJ
5330 DATA 2, ,1,51,1,204,1,60,1,213,1,80,5615'BJJJ
5340 DATA 3, ,1,3,1,192,1,63,1,87,1,1,5354'BGYK
5350 DATA 3, ,1,15,1,207,1,15,1,207,1,255,5707'BKOM
5360 DATA 1,15,1,3,1, ,2,195,1,243,1,240,5703'BJOM
5370 DATA 1,243,1,252,1,243,1,252,40,255,1,51,6341'BPGP
5380 DATA 1,204,1,51,1,204,1,51,1,204,1,51,5771'BMDP
5390 DATA 1,204,7,255,1,254,7,255,1,191,8,255,6439'BPPR
5400 DATA 1,204,2,192,1,204,1,243,2,204,1,243,6298'BPKJ
5410 DATA 8,255,4,243,1,255,3,243,1,48,1,192,6254'BQIK
5420 DATA 14, ,1,12,1,51,1,12,4, ,1,51,5148'BGIJ
5430 DATA 8, ,1,4,7, ,8,34,32,170,8,128,5400'BHIK
5440 DATA 1,51,1,204,1,63,1,192,1,51,1,192,5759'BMYM
5450 DATA 1,51,1,192,1,51,1,204,1,255,1, ,5759'BKTN
5460 DATA 1,51,1,204,1,51,1,205,1,51,1,204,5772'BMFO
5470 DATA 1,255,1, ,1,51,1,204,1,23,1,85,5624'BJSO
5480 DATA 1,51,1,204,1,255,1, ,1,51,1,204,5771'BKHQ
5490 DATA 1,51,1,76,1,51,1,204,1,255,1, ,5643'BJUQ
5500 DATA 1,51,1,192,1,51,1,192,1, ,1,192,5684'BKWJ
5510 DATA 1, ,1,192,1, ,1,192,1, ,1,192,5582'BGVJ
5520 DATA 1,12,1,8,1,10,1,8,1,9,2, ,5054'BETJ
5530 DATA 1,8,1,190,1,170,1,2,1, ,1,128,5504'BIOL
5540 DATA 1,64,1,80,1,160,1,170,1,234,1,2,5716'BLMN
5550 DATA 1,40,1,34,3,2,1,127,1,223,1,255,5689'BLVO
5560 DATA 1,250,1,251,1,234,1,170,1,168,1,51,6130'BOGP
5570 DATA 1,204,1,51,1,206,1,251,1,239,1,162,6119'BOLQ
5580 DATA 1,136,1,112,1,48,1, ,1,48,1,32,5382'BJUQ

6280 DATA 1,63,1,3,3, ,1,207,1,51,1,63,5395'BHXO
6290 DATA 1,67,1,207,1,243,1,51,1,3,40,255,5871'BMXQ
6300 DATA 2,168,1,104,1,88,1,60,1,63,1, ,5490'BJEH
6310 DATA 2,170,2,171,1,175,1,189,1,250,1,21,5984'BOEJ
6320 DATA 1,160,1,106,1,86,3,85,1,106,1,96,5647'BMJK
6330 DATA 1, ,1,166,1,153,1,150,1,85,1,86,5646'BKCL
6340 DATA 1,101,1,64,1,85,1,170,1,169,2,85,5681'BMFM
6350 DATA 1,170,1,85,2,84,2,90,2,85,1,170,5693'BLNN
6360 DATA 3,64,1,255,1,191,2,87,1,171,3,3,5782'BLLO
6370 DATA 2,48,1, ,1,48,1,12,1,48,1,204,5367'BIDO
6380 DATA 1,252,8, ,1,12,1,51,1,12,4, ,5343'BGLP
6390 DATA 1,204,7, ,1,128,2,2,1, ,2,2,5350'BFJQ
6400 DATA 1, ,2,2,1,170,7,42,32,170,8,128,5563'BKXJ
6410 DATA 1,63,1,207,1,63,1,192,1,51,1,192,5774'BMCK
6420 DATA 1,51,1,192,3,255,1, ,1,51,1,204,5761'BKOL
6430 DATA 1,51,1,204,1,240,2,255,1,15,1,48,5820'BMLM
6440 DATA 1,204,1,51,1,20,1, ,3,255,1, ,5538'BHMM
6450 DATA 1,204,1,51,1,204,1,3,1,252,1,255,5975'BMPO
6460 DATA 1,252,1,15,1,192,1,51,1,192,1, ,5708'BKQP
6470 DATA 1,192,1, ,1,192,1, ,1,192,1, ,5582'BGVP
6480 DATA 1,192,1,2,1,1,2,8,1,12,1,32,5254'BHLQ
6490 DATA 1,16,1, ,1,207,1,79,1,67,1,168,5543'BJGS
6500 DATA 1,162,1,152,1,98,1,136,2,84,1,212,5851'BNBK
6510 DATA 1,244,1,60,1,15,1,11,1,128,1,114,5578'BMNL
6520 DATA 1,120,1,122,1,120,1,122,1,120,1,122,5732'BPCN
6530 DATA 1,184,1,34,1,139,1,34,1,139,1,34,5570'BMYN
6540 DATA 1,136,1,34,1,168,1,34,1,136,1,226,5740'BNUO
6550 DATA 1,136,1,242,1,252,1,254,1,143,1,34,6067'BOPP
6560 DATA 1,136,1,34,1,136,1,34,1,136,1, ,5482'BKOQ
6570 DATA 1,192,1,128,1,160,1,128,1,160,1,128,5902'BPKS
6580 DATA 1,8,1,128,1, ,1,48,1, ,3,12,5204'BFOR
6590 DATA 1,3,1, ,1,131,8, ,1,51,1,3,5201'BEHR
6600 DATA 1,51,1,243,2,51,1,207,1,243,40,255,6096'BOTL
6610 DATA 4, ,1,16,12, ,1,1,7, ,1,85,5128'BDAK
6620 DATA 2, ,4,1,1,16,4,85,3,4,1,64,5185'BFFM
6630 DATA 2,80,2,64,3, ,8,3,3,204,1,192,5562'BIGN
6640 DATA 1,207,1,195,1,204,1,195,1,3,1,12,5822'BMPP
6650 DATA 1,63,1,252,1,255,1,252,1,255,1,252,6335'BOUQ
6660 DATA 1,48,1,204,1,48,1,204,1,48,1,204,5762'BMVR
6670 DATA 1,48,1,204,1,188,4,170,3,234,1,2,5857'BMGS
6680 DATA 1,130,1,162,1,160,1,162,1,170,1,160,5950'BPYU
6690 DATA 1,136,2,170,1,42,1,0,170,1,171,4,170,5878'BPQV
6700 DATA 2,171,1,237,1,77,1,170,1,105,4,170,5940'BOWM
6710 DATA 1,234,1,238,2,166,8,170,8,128,1,51,6008'BOCO
6720 DATA 1,192,1,51,1,192,1,48,1,207,1,63,5759'BMIO
6730 DATA 1,207,1,49,1,204,1,51,1,204,1, ,5721'BKGP
6740 DATA 1,255,2,251,1,85,1,205,1,51,1,204,6058'BNRQ
6750 DATA 1,51,1, ,2,255,1,51,1,76,1,51,5491'BIYQ
6760 DATA 1,204,1,51,1,12,1,233,1,214,1,51,5771'BMDS
6770 DATA 1,192,1,35,1,224,1,171,1,168,1,171,5967'BOBU
6780 DATA 1,41,1, ,1,193,1, ,1,196,1,16,5452'BHUT
6790 DATA 1,72,1,160,1,168,1,72,1,130,1,32,5640'BMLV
6800 DATA 1,8,1,168,1,184,1,186,1,252,1,34,5838'BMNN
6810 DATA 1,136,1,34,1,140,1,35,1,136,1,35,5522'BMJO
6820 DATA 1,140,1,32,1,136,1,34,1,136,1,34,5518'BMKP
6830 DATA 1,200,1,242,1,248,1,42,1,136,1,34,5908'BNOQ
6840 DATA 1,140,1,46,1,136,1,34,1,136,1,42,5540'BMJR
6850 DATA 1,138,1,34,1,136,1,34,1,136,1,34,5518'BMTS
6860 DATA 1,136,1,47,1,139,1,163,1,169,1,36,5696'BNPT
6870 DATA 1,144,1,85,1,72,1,243,1,252,1,79,5881'BMJU
6880 DATA 1,81,1,38,1,136,1,34,1,64,1,145,5504'BLUV
6890 DATA 1,96,2,160,1,106,1,40,1,10,2, ,5420'BJGV
6900 DATA 1,131,1,3,1,128,1,32,1,131,1,35,5466'BLHO
6910 DATA 1,128,1, ,1,240,1,12,1,205,1,12,5603'BKVP
6920 DATA 1,205,1,13,1,204,1,143,2,191,1,143,5906'BOIQ
6930 DATA 1,207,1,143,1,207,1,143,40,255,10, ,6009'BOHR
6940 DATA 1,128,11, ,1,15,1,255,1,1,5, ,5419'BHRR
6950 DATA 2,255,1,84,1,4,1,20,2,16,1,80,5467'BJDS
6960 DATA 1,208,1,252,8, ,6,3,1,7,1,3,5491'BGDT
```

```
6970 DATA 1,204,1,243,1,255,1,243,1,240,1,243,6434'BPDW
6980 DATA 1,240,1,243,1,255,1,60,1,63,1,60,5927'BMSW
6990 DATA 1,63,1,60,1,255,1,60,1,48,1,204,5696'BLBX
7000 DATA 1,48,1,204,1,48,1,204,1,48,1,204,5762'BMVG
7010 DATA 5,234,1,255,1,170,1,239,1,168,1,179,6255'BPHI
7020 DATA 1,132,1,136,1,134,1,246,1,134,1,166,5954'BPPJ
7030 DATA 4,170,1,169,1,104,1,41,1,105,4,170,5771'BORJ
7040 DATA 1,166,1,162,2,166,1,40,6,105,1,77,5728'BNHK
7050 DATA 1,162,5,174,2,166,1,174,2,182,1,170,6040'BPQM
7060 DATA 1,166,1,134,2,182,1,16,1,32,1,144,5681'BNTM
7070 DATA 1,145,1,144,1,16,2,220,1,48,1,64,5644'BMTN
7080 DATA 1,115,1,64,1,179,1,64,1,113,1,69,5610'BMXO
7090 DATA 1,8,1,204,1,56,1,9,1,24,2,89,5397'BISO
7100 DATA 1,24,1, ,1,136,1, ,1,89,1,3,5258'BFXG
7110 DATA 1,67,1,84,1,83,1,6,1,69,1,162,5477'BJSI
7120 DATA 1,40,1,82,1,12,1,79,1,63,1,165,5447'BKAJ
7130 DATA 1,184,1,114,1,120,1,114,1,120,1,51,5709'BOBK
7140 DATA 1,191,1,42,1,8,1,50,1,184,1,50,5531'BKNL
7150 DATA 1,136,1,50,1,186,1,254,1,136,1,50,5818'BNXM
7160 DATA 1,138,1,50,1,184,1,178,1,184,1,51,5791'BNHN
7170 DATA 1,143,1,179,1,191,1,51,1,191,1,60,5821'BNUO
7180 DATA 1,252,1,242,1,248,2,250,1,204,1,62,6265'BOPP
7190 DATA 1,207,1,51,1,34,1,136,1,34,1,136,5604'BMLQ
7200 DATA 1,131,1,139,1,2,1,8,1,34,1,136,5456'BKQI
7210 DATA 1,35,1,139,1,35,1,139,1,34,1,136,5524'BMWJ
7220 DATA 1, ,1,248,1,254,1,255,1,239,1,188,6190'BMGK
7230 DATA 1,83,1,67,1,2,1,168,1,40,1,162,5528'BKAL
7240 DATA 1,149,1,127,1,255,1,200,1,168,1,170,6075'BPON
7250 DATA 1,42,1,8,1,191,1,111,1,251,1,254,5863'BMQN
7260 DATA 1,35,1,140,1,179,2,240,1,188,1,255,6044'BOXO
7270 DATA 1,204,1,12,1,204,1,8,1,193,1,4,5631'BKGP
7280 DATA 1,194,1,2,1,136,3,143,1,207,1,14,5704'BMNQ
7290 DATA 1,62,1,250,1,58,1,255,3,239,4,170,6045'BNCR
7300 DATA 4,255,4,170,4,255,4,170,4,255,4,170,6299'BPOK
7310 DATA 4,255,4,170,8, ,1,3,1,15,6,63,5530'BIFJ
7320 DATA 2,255,1,171,1,187,1,175,2,187,3,255,6240'BPEM
7330 DATA 1,254,1,238,1,174,1,239,3,255,1,191,6359'BPIN
7340 DATA 1,239,2,255,1,175,1,252,1,240,8, ,6175'BMDN
7350 DATA 8,3,2,243,1,240,2,255,1,243,2,252,6252'BNSO
7360 DATA 1,255,1,60,1,255,1,12,1,15,1,204,5807'BMMP
7370 DATA 1,207,1,204,1,48,1,204,1,48,1,192,5909'BNAQ
7380 DATA 1,51,1,195,1,51,1,195,1,235,1,250,5983'BNER
7390 DATA 1,255,1, ,1,234,3,170,1,134,1,4,5805'BKMS
7400 DATA 1,55,1,63,1,162,1,168,2,170,2,105,5731'BNUK
7410 DATA 1,99,1,207,1,160,1,10,2,170,1,164,5817'BNVL
7420 DATA 1,167,1,5,1,255,1,42,1,128,2,170,5774'BMCM
7430 DATA 1,77,1,125,1,112,1,42,1,128,1,42,5532'BMLN
7440 DATA 2,170,1,207,1,255,1,10,1,2,1,168,5819'BMSO
7450 DATA 1,2,1,168,1,170,1,191,1,62,1, ,5599'BJBP
7460 DATA 1,168,1,10,1,160,1,10,1,170,1,12,5536'BMBQ
7470 DATA 1, ,1,170,1,11,1,129,1,42,1,128,5486'BKNR
7480 DATA 1,170,1,114,1,122,1,74,1,96,1,95,5677'BMGS
7490 DATA 1,192,1,255,1, ,2,24,2,25,1,201,5705'BKKT
7500 DATA 1,255,1, ,1,255,2,67,1,15,1,255,5854'BKEL
7510 DATA 1,63,1,207,1,255,1, ,1,63,1,3,5597'BIEL
7520 DATA 1,207,3,255,1,63,1,255,1,63,1,184,6035'BNAN
7530 DATA 1,63,4,255,1,21,1,50,1,184,1,59,5641'BLYO
7540 DATA 2,255,1,252,1,5,1,85,1,50,4,255,5912'BLBP
7550 DATA 1,1,2,85,2,240,2,255,1, ,3,85,5677'BILP
7560 DATA 1,207,1,63,1,255,1,51,1,255,1,171,6008'BNQR
7570 DATA 2,168,1,2,1,8,1,2,1,8,1,34,5229'BGBR
7580 DATA 1,8,1,35,1,11,1,34,1,137,1,33,5264'BJSS
7590 DATA 1,137,1,33,1,137,1,33,1,137,2,67,5551'BMWU
7600 DATA 1,79,1,76,1,84,1,53,1,3,1,52,5353'BILL
7610 DATA 1,34,1,136,1,2,1,8,1,34,1,8,5228'BHWM
7620 DATA 1,200,1,42,1,51,1,136,1,32,1,130,5597'BMGO
7630 DATA 1,42,2,40,1,160,1,191,1,232,1,179,5851'BNRP
7640 DATA 1,203,1,50,1,8,1,34,1,136,1,2,5439'BJRP
7650 DATA 1,132,1,2,1,136,1,2,1,136,1,10,5424'BKXR
7660 DATA 1,138,1,186,1,170,6,42,40,170,5, ,5760'BMBS
7670 DATA 3,16,1,15,1,3,5, ,1,8,2,255,5310'BGRS
7680 DATA 1,63,4, ,1,32,3,255,1,240,2, ,5602'BHOT
7690 DATA 1,10,1,8,1,240,1,192,3, ,1,32,5490'BINU
7700 DATA 2,136,5, ,1,160,1,32,1,128,1,3,5470'BJQM
7710 DATA 7,2,5,252,1,192,1,255,1,224,1,207,6148'BNBO
7720 DATA 1,204,1,207,1,204,1,192,1,15,1,255,6083'BOLP
7730 DATA 1, ,1,48,1,204,1,48,1,204,1, ,5510'BHKP
7740 DATA 2,255,2, ,1,248,1,232,1,255,1, ,5998'BJHR
7750 DATA 1,254,1,255,1,2,1,42,1,48,1,136,5743'BLVS
7760 DATA 1,34,1,136,1,34,1,136,1,34,1,170,5550'BMLT
7770 DATA 1,2,1,136,1,34,1,136,1,34,1,136,5484'BLQU
7780 DATA 1,34,1,170,1,10,1,128,1,34,1,136,5518'BMKV
7790 DATA 1,34,1,136,1,34,2,170,1,3,1,32,5416'BKIW
7800 DATA 1,136,1,34,1,136,1,34,3,170,1, ,5518'BKOO
7810 DATA 1,136,1,34,1,136,1,34,3,170,1, ,5518'BKOP
7820 DATA 1,136,1,34,1,136,1,34,4,170,1, ,5519'BKQQ
7830 DATA 1,34,1,136,1,34,4,170,1, ,1,34,5417'BJSQ
7840 DATA 1,136,1,34,5,170,1, ,1,136,1,34,5520'BKJS
7850 DATA 2,168,1,164,2,168,1, ,1,170,1,42,5720'BLUT
7860 DATA 4,170,1, ,6,170,1,128,1,10,5,170,5666'BLUU
7870 DATA 1,160,1,2,5,170,1,168,1,2,4,170,5685'BLWV
7880 DATA 1,175,1,170,1,2,2,170,1,169,1,191,5884'BNEW
7890 DATA 1,213,1,85,1,136,1,170,1,173,2,213,5997'BODX
7900 DATA 3,85,2, ,1,192,1,116,1,95,1,87,5584'BJQP
7910 DATA 1,85,1,89,1,32,1,8,1,32,1,136,5388'BJLQ
7920 DATA 1,34,1,234,1,238,1,171,1,29,1,7,5719'BLCR
7930 DATA 1,113,1,65,1,76,1,156,1,7,1,244,5667'BLGS
7940 DATA 1,34,1,32,1,226,1,32,1,34,1,136,5500'BLDT
7950 DATA 1,34,1,136,1,162,1,136,1,34,1,136,5644'BNRU
7960 DATA 1,34,1,136,1,34,1,137,1,34,1,136,5517'BMRV
7970 DATA 1,34,1,136,1,42,1,143,1,127,1,253,5741'BNNW
7980 DATA 1,10,1,136,1,8,1,170,1,40,1,42,5412'BKAX
7990 DATA 1,32,1,34,2,42,2,170,1,42,43,170,5540'BMJY
8000 DATA 5,20,3,21,1,40,3,8,2, ,2,80,5185'BGTG
8010 DATA 3,136,1,32,4, ,2,8,1,10,5, ,5202'BFKH
8020 DATA 2,136,1,32,5, ,1,128,1,160,6, ,5472'BIWI
8030 DATA 8,2,2,192,1,224,3,255,1,243,1,240,6172'BNUK
8040 DATA 1, ,1,4,1, ,1,240,1,255,1,252,5757'BHSK
8050 DATA 1,255,1,252,4, ,1,48,1,207,1,51,5822'BKTM
8060 DATA 1,207,1, ,1,2,1, ,1,2,1, ,5217'BCHL
8070 DATA 1,232,1,234,1,232,1,8,1,34,1,136,5882'BMSO
8080 DATA 1,34,1,136,1,34,2,170,1,136,1,34,5551'BMNP
8090 DATA 1,136,1,34,1,136,1,34,1,136,1,170,5652'BNPQ
8100 DATA 1,136,1,34,1,136,1,34,1,136,1,34,5516'BMPI
8110 DATA 1,136,1,170,1,136,1,34,1,136,1,34,5652'BNPJ
8120 DATA 1,136,1,34,1,136,1,162,1,136,1,34,5644'BNRK
8130 DATA 1,136,1,34,1,136,1,34,1,136,1,34,5516'BMPL
8140 DATA 1,136,1,34,1,136,1,34,1,136,1,34,5516'BMPM
8150 DATA 1,136,1,34,1,136,1,34,1,136,1,34,5516'BMPN
8160 DATA 1,136,1,34,1,136,1,34,1,136,1,34,5516'BMPO
8170 DATA 1,136,1,34,1,136,1,34,1,136,1,34,5516'BMPP
8180 DATA 1,136,1,34,1,136,1,34,1,136,1,34,5516'BMPQ
8190 DATA 1,136,1,34,1,170,1,42,1,170,1,42,5600'BMDR
8200 DATA 1,170,1,42,1,170,1,42,18,170,1,171,5788'BOYJ
8210 DATA 1,165,2,149,2,85,1,181,1,85,1,213,5886'BNRK
8220 DATA 10,85,1,101,1,105,1,102,2,85,1,90,5584'BNOL
8230 DATA 1,106,1,153,1,105,1,169,1,165,1,89,5793'BOEN
8240 DATA 1,169,1,149,1,117,1,213,3,85,1,89,5830'BNNN
8250 DATA 1,213,1,69,2,85,1,101,2,85,1,119,5680'BMDO
8260 DATA 1,117,1,53,2,117,3,229,1,255,1,119,5899'BOJQ
8270 DATA 1,221,3,85,1,213,1,89,1,255,1,87,5958'BMTQ
8280 DATA 1,221,1,87,1,85,1,118,1,89,1,166,5772'BMQR
8290 DATA 1,254,1,221,1,182,1,222,1,182,1,106,6173'BPDT
8300 DATA 1,154,1,233,1,40,1, ,6,40,6,170,5653'BKOK
8310 DATA 1,162,1,32,6,170,1,10,1,34,7,170,5595'BMUL
8320 DATA 1,234,7,170,1,138,16,170,8,21,1,80,5847'BOEN
8330 DATA 7,85,3, ,3,80,2,85,14, ,1,11,5291'BGIM
8340 DATA 1,8,5, ,1,3,1,192,1,42,1,2,5257'BFVN
8350 DATA 1, ,1,2,2,3,1,204,1,3,1,79,5298'BFWO
8360 DATA 1,195,1,192,1,195,1,192,1,195,1,192,6167'BPLR
8370 DATA 1,195,1,12,1,63,1,252,1,255,1,254,6037'BNUR
8380 DATA 1,253,1,254,1,234,1,170,1,63,2,240,6221'BOIS
8390 DATA 1,2,1,1,1,106,2,170,1,255,1,11,5552'BKBT
8400 DATA 1,144,1,96,1,136,3,170,1,136,1,34,5724'BNYL
8410 DATA 1,136,1,34,1,136,1,34,1,136,1,34,5516'BMPM
8420 DATA 1,168,1,170,1,168,1,170,1,168,3,170,6022'BPQO
8430 DATA 1,4,1,34,1,74,1,34,1,136,1,16,5304'BJRN
8440 DATA 2,170,1,42,1,10,1,42,1,26,1,138,5435'BLIP
8450 DATA 1,26,1,154,5,170,1,162,1,42,1,130,5694'BNSQ
8460 DATA 1,162,1,136,1,42,1,170,1,10,1,106,5632'BNCR
8470 DATA 1,10,1,162,1,72,1,136,1,162,1,170,5718'BNMS
8480 DATA 3,160,1,170,1,168,1,136,1,34,1,136,5812'BOST
8490 DATA 1,42,1, ,1,130,1, ,1,106,1,136,5420'BIYT
8500 DATA 1,34,1,136,2,170,1,34,1,136,1,34,5551'BMNM
8510 DATA 1,170,1,42,2,170,1,171,2,168,4,170,5902'BOSN
8520 DATA 1,128,1,51,1,204,1,179,1,34,2,170,5773'BNXO
8530 DATA 1,132,1,1,1,116,1,93,1,20,1,173,5541'BLEP
8540 DATA 11,85,1,101,1,89,1,86,1,85,1,150,5612'BMCQ
8550 DATA 1,89,1,86,2,90,1,170,1,106,1,170,5718'BMHR
8560 DATA 1,168,7,165,1,21,2,85,1,89,1,86,5627'BLAS
8570 DATA 4,85,3,89,1,86,1,153,1,149,1,105,5678'BMBT
8580 DATA 1,89,1,214,1,217,1,218,2,234,2,218,6198'BOEV
8590 DATA 1,234,1,229,1,166,1,153,1,106,1,170,6064'BPKW
8600 DATA 1,102,2,170,1,91,1,102,1,169,1,102,5743'BOGN
8610 DATA 1,170,1,106,2,170,1,154,1,100,2,166,5874'BPIP
8620 DATA 1,165,1,170,1,168,1,34,1,40,1,148,5731'BNUP
8630 DATA 2,170,1,168,1,34,1,136,1,34,1,137,5686'BNEQ
8640 DATA 1,170,1,138,1,34,1,138,1,34,1,138,5658'BNCR
8650 DATA 1,168,1,162,1,8,1,130,1,10,1,130,5614'BMES
8660 DATA 1,160,1,128,1,136,1,32,1,138,1,32,5632'BNLT
8670 DATA 1,42,1,2,1, ,1,42,1,136,1,42,5270'BHHT
8680 DATA 1,34,1,136,1,34,1, ,1,128,1, ,5338'BHRU
8690 DATA 1,130,1,170,1,42,1,2,2,128,1,138,5617'BMKW
8700 DATA 1, ,1,2,3,170,1,138,1,32,1,128,5478'BJXN
8710 DATA 1,34,1,32,3,21,1,20,1, ,1,252,5367'BIKO
8720 DATA 1,194,1,34,1,85,1,87,1,64,1,62,5532'BKIQ
8730 DATA 1, ,1,1,1,32,1,8,1,64,1,243,5354'BGPQ
8740 DATA 1,192,1,42,1,1,1,86,1,128,1, ,5455'BIYR
8750 DATA 1,252,1,242,1, ,1,106,1, ,1,168,5774'BJVS
8760 DATA 1,2,1,136,1,192,1,32,1, ,1,168,5536'BJWT
8770 DATA 1, ,1,72,1,34,1,130,1,1,1,166,5409'BINU
8780 DATA 1, ,1,64,1,42,1,32,1,8,1, ,5152'BERU
8790 DATA 1,59,1,197,1,183,1,15,1,53,1,215,5728'BMJX
8800 DATA 1,55,1,79,1,3,1,207,1,3,1,15,5368'BIFO
8810 DATA 1,51,1,15,1,63,1,15,1,170,1,180,5500'BLDQ
8820 DATA 1,250,1, ,1,252,1,235,1,202,1,208,6153'BMCR
8830 DATA 1,160,1,68,1, ,1,234,1, ,1,160,5628'BITR
8840 DATA 1,170,1,234,1,2,1,102,1,24,1,32,5570'BLYT
8850 DATA 1,170,1, ,1,128,1,170,1,136,1,34,5644'BLMU
8860 DATA 1,136,1,34,1,136,1,162,1,8,1,34,5516'BLRV
8870 DATA 1,168,1,170,1,168,2,170,2,168,2,170,6023'BPSX
8880 DATA 1,8,1,66,1,42,1,128,1, ,1,42,5292'BHYW
8890 DATA 3,170,1,42,1,128,1,10,2,170,1,18,5547'BMOY
8900 DATA 1,129,1,34,1,130,1,161,1,129,1,161,5750'BOLQ
8910 DATA 1,128,1,160,1,34,1,8,1,34,1,136,5506'BLPR
8920 DATA 1,162,1,136,1,162,1,136,1,162,1,164,5928'BPRT
8930 DATA 1,162,1,168,1,162,1,160,1,162,1,168,5988'BPEU
8940 DATA 1,24,1,2,1,32,1,66,1, ,1,2,5132'BFFT
8950 DATA 1,32,1,130,1,136,1,34,1,136,1,34,5508'BMIV
8960 DATA 1,136,1,34,1,136,1,34,5,168,1,170,5688'BNIW
8970 DATA 1,168,1,170,1,34,1,136,1,34,1,64,5612'BMSX
8980 DATA 1,18,1,32,1,8,1,34,2,42,1,34,5175'BIWX
8990 DATA 1,4,1,32,1,64,1,32,1, ,1,89,5227'BGWY
9000 DATA 1,86,6,85,1,105,1,150,1,89,1,86,5612'BLNI
9010 DATA 1,90,1,86,1,90,1,86,3,128,1,162,5650'BLIJ
9020 DATA 1,138,1,34,1,138,1,34,3,21,5,165,5542'BMWK
9030 DATA 2,85,1,102,1,86,4,85,1,91,1,103,5562'BLDL
9040 DATA 1,171,2,107,1,155,1,107,1,103,2,192,5843'BPJN
9050 DATA 1,64,2, ,1,42,1,138,1,104,5, ,5359'BHXM
9060 DATA 1,128,1,170,1,42,1,136,1,34,1,136,5652'BNPO
9070 DATA 1,34,1,136,1,38,2,170,1,136,1,34,5555'BMVP
9080 DATA 1,138,1,34,1,170,1,168,2,128,1,138,5783'BOCR
9090 DATA 1,42,3,170,2,138,1,170,1,129,1,146,5804'BOUR
9100 DATA 1,160,1,170,1,162,1,168,1,162,1,168,5996'BPCK
9110 DATA 1,136,1,137,1,34,1,168,1,64,1,170,5715'BNYK
9120 DATA 1, ,1,170,2,136,1,34,1,152,1,42,5541'BKGL
9130 DATA 1, ,1,136,1,32,1,16,1,98,1, ,5288'BGCL
9140 DATA 1,134,1,2,1,8,1,130,1,8,1,2,5290'BHLM
9150 DATA 1,34,1,136,1,98,1,32,1,132,1,2,5440'BKLO
9160 DATA 1,128,1,2,1,32,1,130,1,96,1,138,5532'BLNP
9170 DATA 1,32,1,138,1,32,1,8,1,37,1,4,5257'BIYP
9180 DATA 1,38,1,4,1,37,1,8,1,34,1,130,5257'BIYQ
9190 DATA 1,8,1,66,1,72,1,70,1,1,1,130,5353'BISR
9200 DATA 1,8,1,42,1,69,1,68,1,69,1,68,5330'BITJ
9210 DATA 1,6,3, ,1,66,1,72,1,2,1,72,5226'BFWK
9220 DATA 1,68,1,34,1,136,1,1,1,98,1,152,5495'BKGM
9230 DATA 1,34,1,136,1, ,2,32,1,137,1, ,5346'BHPM
9240 DATA 1,136,1, ,1,128,1,132,1, ,1,136,5538'BJRN
9250 DATA 1,5,1,111,1,183,1,95,1,191,1,21,5612'BLLP
9260 DATA 1,7,1,111,1,51,1,15,1,3,1,15,5208'BIHP
9270 DATA 1,3,1,15,1,3,1,15,1,119,1,85,5246'BIUQ
9280 DATA 1,120,1,200,1,98,1,72,1,34,1,32,5562'BLKS
9290 DATA 1,2,1,64,1,129,1,40,1, ,1,68,5309'BHWS
9300 DATA 1,8,1,72,1,170,1,42,1,2,1, ,5300'BGGK
9310 DATA 1,144,1,4,1, ,1,98,1,136,1,34,5422'BIVL
9320 DATA 1,136,1,34,1,136,1,34,1,136,3,170,5654'BNTN
9330 DATA 1,168,1,169,1,170,3,168,1,164,1,34,5881'BOJP
9340 DATA 1,132,1, ,1,32,1,152,1,2,1,152,5476'BJKO
9350 DATA 1, ,1,136,1,66,1,24,1,132,1,32,5396'BJVP
9360 DATA 1,8,1,32,1,128,1,160,1,128,1,160,5622'BMLR
9370 DATA 1,128,1,160,1,168,2,162,1,136,1,162,5923'BPQT
9380 DATA 1,136,1,162,1,136,1,162,1,136,1,162,5900'BPGU
9390 DATA 1,168,1,160,1,162,1,168,1,160,1,169,5993'BPFV
9400 DATA 1,160,1,24,1,138,1, ,1,34,1,128,5490'BKMM
9410 DATA 1,74,1,8,1,74,1,136,1,34,1,136,5468'BKGN
9420 DATA 1,34,1,136,1,34,1,136,1,32,8,168,5553'BMBO
9430 DATA 1,152,1,98,1,136,1,34,1,136,1,34,5596'BMGP
9440 DATA 1,136,1, ,1,34,1,72,1,34,1,164,5446'BJWP
9450 DATA 1,2,1,106,2, ,8,85,1,90,1,102,5399'BIEQ
9460 DATA 1,154,1,106,1,102,1,154,1,106,1,170,5798'BPJT
9470 DATA 1,138,1,34,1,138,1,34,1,138,1,34,5522'BMST
9480 DATA 1,138,1,34,4,165,4,169,1,85,1,101,5704'BNEU
9490 DATA 1,89,5,85,1,155,1,103,1,91,1,87,5620'BLLV
9500 DATA 1,91,1,87,1,91,1,87,1,16,1,68,5446'BJTN
9510 DATA 1,16,1,34,1,32,1,34,1,136,1,170,5428'BLMO
9520 DATA 1,34,1,40,1,34,1,40,1,137,1,40,5331'BKAP
9530 DATA 1,152,1,168,2,170,1,42,1,138,1,34,5711'BNQQ
9540 DATA 1,138,1,34,4,170,1, ,1,34,1,8,5393'BIBQ
9550 DATA 1,32,1,170,1,136,1,162,1,168,1,170,5844'BOTS
9560 DATA 1,136,1,10,1,160,2,162,1,34,1,161,5670'BNGT
9570 DATA 1,42,1,160,1,34,1,160,1,32,1,64,5498'BLRU
9580 DATA 1,162,3, ,1,162,1,8,1, ,1,128,5468'BHAU
9590 DATA 1,34,1,8,1,34,1,104,1,2,1,160,5348'BJPV
9600 DATA 1,2,1,66,1,18,1, ,1,32,1,128,5252'BHQN
9610 DATA 1, ,1,128,1,32,1,9,1,32,1,128,5335'BISO
9620 DATA 1, ,1,72,1,136,2, ,1,10,1,32,5257'BGKP
9630 DATA 1,10,1,96,1,10,1,38,1,8,1,64,5232'BITQ
9640 DATA 1,38, , , , , , , , , ,5039'BTAP
9650 DATA 1682,101928,1624,98724,1618,100245'BJJT
```

Congratulations !!

If you own an Apple IIc, you'd have to add all this



to match the versatility, expandability and higher intelligence of the new Commodore 128

(and it costs less too).

The new Commodore 128™ personal computer is breakthrough technology at a breakthrough price. It outshines the Apple® IIc in performance capability, performance quality and price. It is expandable to 512K RAM while the IIc isn't expandable at all.

And the new Commodore 128 has a numeric keypad built right into its keyboard that makes crunching numbers a lot easier. And the Commodore 128 has graphic and sound capabilities that far exceed those of the Apple IIc. But the most important news is that

the new Commodore 128 jumps you into a whole new world of business, productivity, education and word processing programs while still running over 3,000 programs designed for the Commodore 64.™ That's what we call a higher intelligence.

COMMODORE 128 A Higher Intelligence

commodore COMPUTER

Keeping up with you.

BEEMAN MAYRHOFER STOTT/CC422

The Commodore 64 Music Maker.

The Keyboard For The Keyboard

Well, it's not exactly 'clip on Chopin', but the Commodore 64 Music Maker certainly adds a new dimension to home computers. It's the 'keyboard' for the 'keyboard'. Just clip on and play.
Music Maker comes with instructions and easy-step melody book to enable you to explore the wonderful world of music at your own pace and in your own time. In fact, it's so much fun that music lessons may never be the same again. Music Maker encourages musical adventure and experiment,and that, in our opinion, is a good place to start experiencing the magic of making your own music.
Talk to your nearest Commodore dealer today about the simple, easy way to turn your Commodore 64 computer into a Music Maker.

commodore COMPUTER

Keeping up with you.

Contact your nearest Commodore Dealer or Commodore Business Machines:
Sydney: (02) 427 4888. Melbourne: (03) 429 9855. Brisbane: (07) 393 0300. Perth: (09) 389 1266. BMS/CC373