

THE Magazine for experienced TANDY Color Computer Users

\$4⁵⁰

786 144888EPI7
COLOR & SOFTWARE

Jack RAE
P O BOX 2876
MT ISA

QLD 4825

URGENT!!! OUR NEW PHONE NUMBER IS 075-39-6177

AUSTRALIAN

CoCo

MAGAZINE



This month's special features include:

STOCK LIST — For Inventory Control.

MEMO PAD — A Quality File Control Utility.

MISSION DESTRUCTION — Graphic Adventuring.

BIG TEXT — Creates large text on Hi-res Screen.

plus

GAMES — CoCo3 tackles SHOOTOUT and ONO.

M/L PROGRAMMING — A tutorial by Mal Patrick, Part 4.

VOL3NO10 JUNE87

BLAXLAND COMPUTER SERVICES PTY. LTD.

134 GREAT WESTERN HIGHWAY,
BLAXLAND. N.S.W. 2774

TELEPHONE: (047) 39 3903

SUPER SALE

STOCK PURCHASED AT SPECIAL PRICES

6 only CoCo3's \$400 ea

6 only Tandy Drives \$450 ea

6 only RS232 Packs \$160 ea
(for use with OS-9 level 2)

10 Mb Hard drives with controller

for your CoCo2/3 \$1299

20 Mb Hard drives \$1599

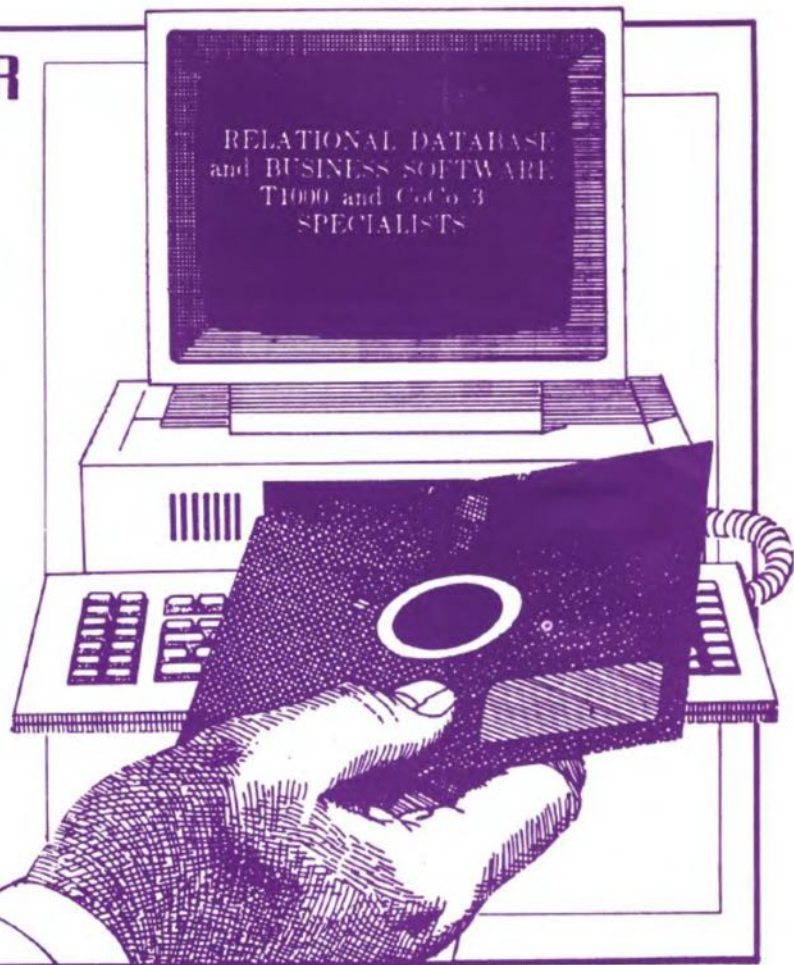
512K upgrades for CoCo3 (won't work
loose—premium 120 n/s 256K chips, gold
edge contacts.)

\$199.00 inc. Disk Public Domain

Software— RAM Disk Text Editor,

Disassembler, Window Utilities,

Coms Software & more!



GOLDLINK

a Goldsoft Service

*642#

for

- Tandy Computer Owners
- Atari Computer Owners
- Gold Coast News and Information
- Rail Enthusiast Information
- Amateur Radio Information
- The Viatel Fast Florist
- Consumer Watch
- Articles and Reviews
- Hardware and Software for
- Your Computer
- Bulletin Boards

PENINSULA MUSIC & ELECTRONICS CENTRE

SHOP 54, KARINGAL HUB,

FRANKSTON 3199

Ph (03) 789 7997. AH (059) 75-4790

.....

Best Prices on AMIGA Monitors
and Data Sheet to suit CoCo 3

.....

Specials on Computer Training
4 lessons for the price of 3 !!

.....

Best prices on Commodore Hardware

Contact: Michael Monck

* FREE

DELIVERY





AUSTRALIA'S NATIONAL VIDEOTEX SERVICE

APPLICATION FORM

DATE OF APPLICATION / /

(BEFORE COMPLETING THIS APPLICATION, PLEASE READ REVERSE SIDE CAREFULLY)

section**1**

PLEASE TICK APPROPRIATE BOX TO INDICATE SERVICE REQUIRED

BUSINESS SERVICE NON-BUSINESS SERVICE

(CHARGES INCURRED ON BUSINESS SERVICES ARE USUALLY TAX DEDUCTIBLE)

SURNAME (OR BUSINESS NAME IF BUSINESS SERVICE)

GIVEN NAMES

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

POSTAL ADDRESS NUMBER/STREET

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

SUBURB/CITY

STATE

POSTCODE

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

--	--	--	--

--	--	--	--	--	--

TELEPHONE NUMBER ON WHICH SERVICE IS REQUIRED (INCLUDING STD CODE)

--	--	--	--	--	--	--	--	--	--	--	--	--

section**2**

CONTACT NAME (IF BUSINESS SERVICE)

GIVEN NAMES

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

POSTAL ADDRESS FOR BILLING IF DIFFERENT FROM SECTION 1 ABOVE
NUMBER/STREET

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

SUBURB/CITY

STATE

POSTCODE

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

--	--	--	--

--	--	--	--	--	--

CONTACT TELEPHONE NUMBER (INCLUDING STD CODE)

--	--	--	--	--	--	--	--	--	--	--	--	--

section**3**

PLEASE DESCRIBE NATURE OF BUSINESS (OR OCCUPATION IF NOT A BUSINESS SERVICE)

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

PLEASE INDICATE TYPE OF EQUIPMENT USED TO ACCESS VIATEL

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

special instructions

THIS FORM SHOULD BE HANDED IN AT ANY TELECOM BUSINESS OFFICE OR MAY BE MAILED WITHOUT A STAMP TO FREEPOST 20, VIATEL BOX 188C, GPO MELBOURNE, VICTORIA 3001

PLEASE ALLOW TEN WORKING DAYS FOR PROCESSING OF APPLICATION AND RETURN MAIL ADVICE.

telecom use only

DTE					
-----	--	--	--	--	--

PP					
----	--	--	--	--	--

VN					
----	--	--	--	--	--

--	--	--	--	--	--

BG			SC		
----	--	--	----	--	--

CI					
----	--	--	--	--	--

GOLDLINK: The place to be on VIATEL

REF					
-----	--	--	--	--	--

REGISTRATION AND SUBSCRIPTIONS

Customers must register as a Business Service if the telephone number nominated for the use of the VIATEL Service is a Business Service and/or VIATEL is to be used wholly or mainly for Business, Commercial, Industrial, Professional or Government purposes. (Charges incurred on Business Services are usually tax deductible.)

Where a Business Telephone Service is nominated for the use of VIATEL, but the use of VIATEL is wholly or mainly for Non-Business purposes, the Customer may be registered as a Non-Business VIATEL subscriber, providing the registration is taken out in the Customer's personal name and address and not a Business name.

Telecom Australia will register the Business or Individual named under Section 1 as a Customer of its VIATEL Service and will provide the Customer with a confidential Customer Identity Number and Personal Password by mail.

Where billing address is indicated, bills and bill related correspondence ONLY will be forwarded to that address. All other correspondence will be forwarded to address under Section 1.

Customers should advise VIATEL of any change of address as soon as possible.

If you lose your Customer Identity Number and/or Personal Password, you must advise VIATEL in writing before new numbers are issued. Our postal address is: Freepost 20, Box 188C, GPO Melbourne, Vic. 3001. FOR SECURITY REASONS REPLACEMENT NUMBERS AND PASSWORDS CANNOT BE PROVIDED OVER THE TELEPHONE.

Customers of VIATEL acknowledge that their name and registered VIATEL Number will appear on the VIATEL Mailbox Directory and that Service Providers and/or other registered VIATEL users may send messages to their VIATEL number.

Telecom Australia undertakes no responsibility in relation to the accuracy of the information or service provided by Service Providers on VIATEL. Telecom Australia will not be responsible for any loss or damage arising out of or in any way connected with the use of this information or service.

Attention is also drawn to the terms and conditions governing the provision of information and services by some Service Providers. These terms and conditions may, in some cases, include a disclaimer absolving the Service Provider from liability regarding information and services supplied on VIATEL. The means of accessing these terms and conditions is set out on the Service Provider's Index Page on VIATEL.

Should you require any changes to your existing telephone equipment (e.g. new exchange line, additional socket), please contact your local District Telecom Office.

In a small number of cases VIATEL reception may be unsatisfactory. Correction may incur an additional charge.

inside COCO

Utilities

8 MINIMON
Son of Mon.

9 DIRECTORY and
MEMOPAD UTILITY
Graphics directory and memopad
facility all in the one
package.

28 MEMORY SHIFTER
Transfers an area of memory.

29 WHERE IS IT?
Changes to Brian Briggs'
original program.

31 AUTOLOAD and PROTECT
Gives password protection to
BASIC programs.

37 BIGTEXT
Prints various sized characters
to the hi-res screen.

42 PRINTER 2
For use with a Siemens
Teleprinter.

50 DISKS and DIRECTORIES
Prints your disks directory.

51 SHORTHAND KEY UTILITY
Makes typing in programs a lot
easier.

55 TEST VDG
Access extra goodies.

Games

16 MISSION DESTRUCTION
A CoCo 3 graphic adventure.

20 ONO
Based on the card game.

24 MAD MILE
Become a racing driver!

34 SHOOTOUT
Shoot the bad guys, before
they shoot you!

44 YAHTZEE
You'll love this one!

47 DRIFTING
An exercise in momentum
conservation.

Music

40 MUSIC PERFORMANCE
For those of you who are
musically inclined.

Graphics

28 POLAR
Draws four flower-like
diagrams.

54 ARTIST
Hi-res graphics demo.

54 DISH
Creates three dimensional
dish shaped objects.

Business

13 STOCK LIST
Takes care of listing your
stock.

22 CoCo WORD
A small word processor.

Applications

41 8 x 8 COLOUR PATTERN
See the 64 colours of the CoCo3

58 TEST YOUR SENSES
The name says it all!

Australian CoCo Magazine is a copyright
publication of GOLDSOFT, P.O. Box 1742,
Southport, Qld. 4215. All articles and
programs are the property of their authors
and may only be copied for the purpose of
providing two backups to the magazine
purchaser.

Founder: Greg Wilson.
Managing Editor: Graham Morphet.
Editor: Alex Hartman.
Accounts: Annette Morphet.
Production: Paul Wynne
Advertising: Graham Morphet.

Subeditors: John Redmond, Fred
Bisseling, Jack Fricker.

Special thanks to: Geoff Fiala,
Martha Gritwhistle, Mike Turk,
Brian Dougan.
Deadlines: 7th of preceeding month.
Phone: 075 51 0577 or contact on
Vitel *64213#
Registered Publication QBG 4009.

IN A NUT SHELL



'What's new

What's new?", I asked when I stepped through the door of the office. (I had recently arrived from a weeks' well-earned holiday.)

"Oh nothing much ...", said a girl that I'd never seen before in my life.

I asked, "Who are you, and why are you sitting at my desk? Where IS my desk? What is this printer? Why is the magazine format smaller?" Why is everything a mess? Why ..."

If I knew that this was happening, I wouldn't have gone on holiday. (Actually, I would have gone on holiday. I just wouldn't have gone to work that day. I'd have stayed in bed!)

Who's New: Karen

We have a new voice on the 'phone. That new voice is Karen and her job is to answer the 'phone and be responsible for all the mail. She also looks after the accounts.

What's New: Magazine Format

Also, as you would have noticed, we have a new magazine format. It is just a few centimeters smaller, and this is due to the system our new printers use.

OS-9 Level II.

I must say that this is a very impressive piece of software! I have now got my little paws on the software and I'm drooling all over it!

Mind you, the manual is VERY big ... I've been forced to take out the BASIC09 section to make room for everything else! (it fits quite comfortably, now ...)

So ... if you're thinking seriously about expanding your system to be more powerful, go get OS-9 level 2.

Grafton User Group

On Thursday (30 April) Graham and I went down to Grafton to meet the new User Group there.

The users who met there that night were people with different computers and interests - there were people with Commodores, IBM's, Tandy's and Apples.

About 40 people came and it was fun. We started at 7.30 and finished at about 11:30 - 12:00.

Programs and Articles Needed (Urgently!)

We have just a very s-m-a-l-l problem here at the office. We are sorta running low on programs and articles for Softgold & CoCo magazine.

So if you've got a piece of software or an article that you've written/you're planning to write, please get it in soon!

Erk! Think of the consequences if we don't get anything!

Conf '87

As the response to the request for people to register for the bus was so poor, we will not be hiring a bus to go to Conf'87 this year.

Competitions

We've been running a number of competitions around here, being namely the ...

- Game competition: We've received a large number of games of excellent standard - as witnessed by the programs in the magazines of late.

The competition is fierce but still there's plenty of time and scope.

?'
.....

CORRECTION

I can see that whoever wins this one will certainly deserve the prize.

- Utilities Competition: Again, there have been a number of entrants to this one, again all of excellent standard.

- Martha's got this graphics competition going ... let's not get her fuming! (She said the other day that she still hasn't seen anything worth giving a prize to!!)

Besides I want to live, so come on, let's see those entries for the Graphics Competition.

We're Moving!!

Yes, that's right. After being in an old house for the past 4.5 years, somebody finally said, "Hey, let's blow this joint!" (In English: "Let's move out of this rickety old house.")

What does this mean to you? Well, for one two new telephone numbers. Check Telecom, Viatel or perhaps the labels on mail cuts from us for the new numbers!

Secondly a new address - it's not going to be 33 North St anymore, it's 26/207 Currumburra Rd, Ashmore, Qld, 4214.

BUT!! - the postal address still stands at PO Box 1742, Southport, Qld, 4215. More on this subject next month ...

Bye for now!

INVERSE SWITCH

"Inverse Switch" (May CoCo, 1987)

There was a small mishap in the diagram and the instructions. The article, as it stands now, doesn't seem to harm the CoCo's; powering them up with the modification installed will get you trash on the screen.

The correct procedure is as follows:

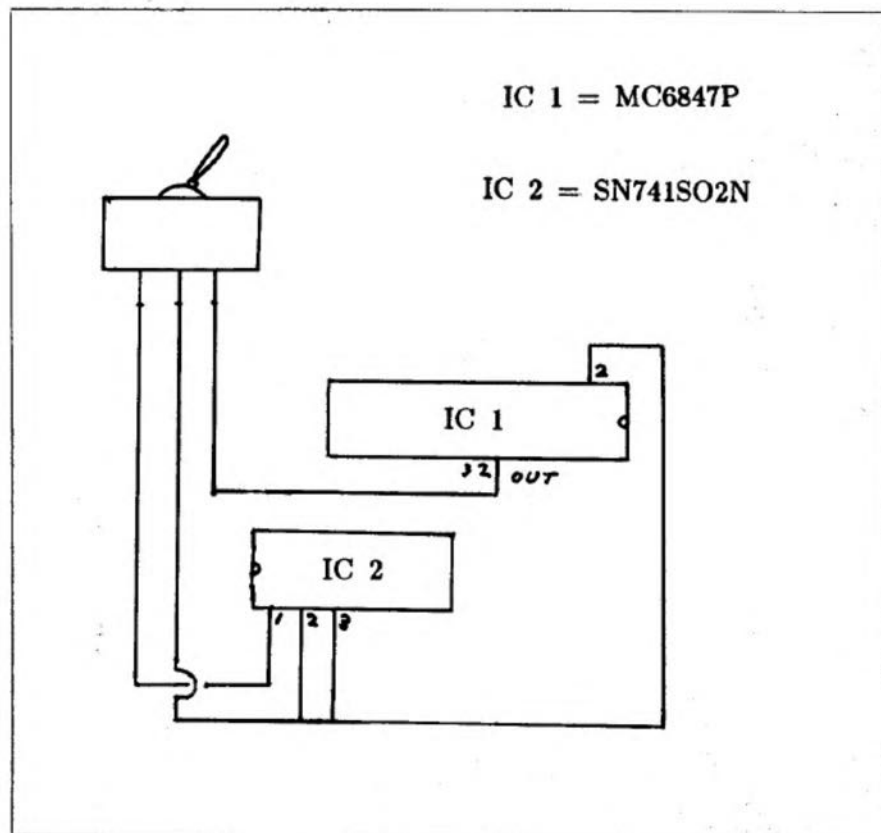
* the left side of the switch will go to pin 1 of the SN74LS02 chip (the 14 pin chip).

* the right side of the switch will go to pin 32 (9th pin on the bottom from the right) of the MC6847P chip (the 40 pin chip).

This pin, by the way, is not touching the socket it once belonged in, ie pin 32 has been bent out of its' socket.

* the middle of the switch attaches itself to pin 2 AND 3 of the SN74LS02 chip (the 14 pin chip) and then continues on to pin 2 of the MC6847P chip.

That's it. VERY sorry for this misleading piece of information.



SUBMITTING YOUR WORK

Ah! So you've finally finished that program? And you say to yourself, "What a great program that would make for CoCo Magazine/Softgold Magazine!"

And so you wonder to yourself, "How am I going to send this program in to the magazine?". Some time goes by and you suddenly realise, "Hey, there's an article in this month's magazine about submitting your work. I'll read through that and maybe that'll help me."

So you rip the magazine out of your stack of other CoCo/Softgold magazines and read the article on how to submit your program.

It reads ...

"... we accept programs stored on both tape and disk ONLY along with a hard copy of the program(s) (optional only; we use it here as a reference to see what the program is/does) and suitable instructions.

Saving to Tape

Each program would be best saved three times with the last save being in ASCII. The tapes we recommend you use are either a C30 or less (the reason for that is that tapes longer than C30 have a tendency to tear).

It'd be even better if you could include some instructions along with the program, either as a separate program or in the wordprocessors listed below.

Saving to Disk

With disk, you'd be best to save it three times with the last save being in ASCII. Also, the extension name for the second and third copy should be different, so to distinguish the three copies. A simulation is given below.

"... I have just saved 3 copies of a program called "HORSE". The directory listing would be:

```
HORSE BAS 0 B 3
HORSE 1 0 B 3
HORSE 2 0 A 3'
```

Any instructions could be saved in the same system using either a program or in the wordprocessors listed below.

Wordprocessors we use.

Here is a list from our most preferable wordprocessors to the drastic measure one could take to tell us how your program works.

1. Telewriter/Telepatch
2. Scripsit
3. PenPal
4. VIP Writer
5. Any form of data file.
6. Instructions written in a separate program."

"Oh wow!", you think to yourself as you read it with awe and astonishment. So you go about your busy little way saving your program and instructions to tape or disk. Then you say to yourself, "Where do I send it?"

You read the article on ...

"... any articles and programs should be sent to this address:

Submissions Editor,
Freeport 5
PO Box 1742,
Southport, Qld, 4215

All mail to this address need not be paid for.

All tapes and disks received will be returned after three months in case we need to refer to something or re-print something."

So place your tape/disk along with your hardcopy of the listing in a postpack (or suitable wrapping) and pop it in the mail.

All done!!

the GOLDSOFT WISHBOOK



The Goldsoft Wishbook
The following products are available
on order from us.

To order, contact us by phone, Viatel
or letter, giving your name, address,
phone number and credit card number, as
well as the Item # shown beside the
product as listed below.

All items include post and packing.

Item #	CoCo Hardware Description	Cost
B 001	512K upgrade kit for CoCo 3 owners	\$199.00
B 002	10 Mb Hard drive inc software	\$1299.00
B 003	Cowling! CoCoNet! Networking extraordinaire!	TBA
G 001	The CoCoConnection - Use your CoCo to control models, alarms - anything electrical	\$206.00
G 002	Video Amplifier with sound - attach your CoCo 1 or 2 to a Video monitor	\$35.00
G 003	The Probe - A temperature sensing unit you plug in to the joy stick port.	\$49.95
G 004	64K Upgrade Kit - upgrade your CoCo2's memory to 64K!	\$55.00

Item #	CoCo Software Description	Cost
B 1001	The Vizi! The ultimate OS-9 L2 Coms package! Multi Windowing, VT52 Emulator, 300-19200 baud, RS 232 protocol	\$159.00
B 1002	IMS - Relational Data Base written in 4GL & VERY fast! OS-9 L2	\$299.00
G 1001	Say the Vordz - two Curriculum based, speller programs for your Tandy Speech / Sound Pack (32K ECB)	\$29.95

Item #	The CoCo 3 Tape/Disk	Cost
G 1002	# 1	\$16.00
G 1003	# 2	\$16.00
G 1004	# 3	\$16.00

Item #	The Best of CoCoZ	Cost
G 1005	# 1 Education	\$16.00
G 1006	# 2 Part 1 16K Games	\$16.00
G 1007	# 2 Part 2 32K Games	\$16.00
G 1008	# 3 Utilities	\$16.00
G 1009	# 4 Business	\$16.00
G 1010	# 5 Adventure Games	\$16.00
G 1011	# 6 Preschool Edu	\$16.00
G 1012	# 7 Graphics	\$16.00
G 1013	# 8 16K Games	\$16.00
G 1014	# 9 32K Games	\$16.00
G 1015	# 10 Education	\$16.00
G 1016	# 11 Education (Disk only)	\$16.00

Item #	Tandy and IBM PC Hardware Description	Cost
Q 001	Colour Monitor (DTX 2001)	\$680.00
Q 002	Mono Monitor	\$190.00
Q 003	Mouse	\$90.00
Q 004	80286 Speed Card	\$550.00

Item #	Tandy and IBM PC Software Description	Cost
Q 1001	dBase II	\$1043.00
Q 1002	dBase III	\$1470.00
Q 1003	Sidekick	\$215.00
Q 1004	Turbo Pascal 8087	\$261.00
Q 1005	Turbo Pascal BCD & 8087	\$244.00
Q 1006	Crosstalk	\$306.00
Q 1007	Lotus 123	\$1054.00
Q 1008	Vordstar 2000+	\$927.00
Z 2001	Webster's New World Writer	\$249.00
Z 2002	Webster's Spelling Checker	\$89.95
Z 2003	Webster's Thesaurus	\$89.95
Z 2004	Windowword	\$269.00
Z 2005	Ready	\$90.00
Z 2006	Thinktank	\$385.00

Item #	CAD Packages Description	Cost
Z 2007	TURBOCAD (V 1.4)	\$399.00

Item #	Trade Business Packages Description	Cost
Z 2008	The Motor Trade Package	\$99.00
Z 2009	The Professional's Pack	\$99.00
Z 2010	The Retailer's Package	\$99.00
Z 2011	The Rental Package	\$99.00

Item #	Accounting Description	Cost
Z 2012	Asset Manager	\$1170.00
Z 2013	Cash Desk/Finance Desk	\$399.00
Z 2014	System 4	\$645.00
Z 2015	C.P.A. Plus	\$395.00

Item #	Databases Description	Cost
Z 2016	Omni3 IBM-Single user (Multi user versions are available)	\$495.00

Item #	Spreadsheets Description	Cost
Z 2017	Logistix	\$399.00

Item #	Fun Description	Cost
Q 1020	Ancient Art of War	\$90.00
Q 1021	Print Shop	\$119.00
Q 1022	Gato	\$68.00
Q 1023	Sargon III	\$95.00
Q 1024	Zork I	\$79.00
Q 1025	Zork II	\$79.00
Q 1026	Zork III	\$79.00
Q 1027	Trinity	\$79.00
Q 1028	Ballyhoo	\$79.00
Q 1029	Hitch Hicker's Guide to the Galaxy	\$79.00
Q 1030	Crossword Magic	\$68.00
Q 1031	The American Challenge	\$68.00
Q 1032	Balance of Power	\$89.00
Q 1033	Racter	\$79.00
Q 1034	Jet	\$114.00
Q 1035	Noonist	\$79.00
Q 1036	Shanghai	\$68.00
Q 1037	Championship Golf	\$89.00
Q 1038	Borrowed Time	\$68.00
Z 2018	The Great International Paper Airplane Construction kit	\$49.95
Z 2019	Star Trek	\$49.95

Z 2020	Championship Boxing	\$69.95
Z 2021	Ultima II	\$69.95
Z 2022	Decision in the Desert	\$69.95
Z 2023	F-15 Strike Eagle	\$69.95
Z 2024	Kings Quest	\$69.95
Z 2025	Mean 16	\$69.95
Z 2026	Boulderdash	\$49.95
Z 2027	Boulderdash II	\$49.95
Z 2028	Conflict in Vietnam	\$69.95
Z 2029	Dambusters	\$69.95
Z 2030	Kings Quest II	\$69.95
Z 2031	PSI-5 Trading Company	\$69.95
Z 2032	Silent Service	\$69.95
Z 2033	Solo Flight	\$69.95
Z 2035	Star Fleet	\$59.95

Item #	Education Description	Cost
Z 2036	Chem Lab	\$69.95
Z 2037	Creature Creator	\$59.95
Z 2038	Crypto Cube	\$59.95
Z 2039	Decimal Dungeon	\$49.95
Z 2040	Donald Duck's Playground	\$59.95
Z 2041	European Nations and Locations	\$59.95
Z 2042	Fraction Action	\$49.95
Z 2043	Math Maze	\$59.95
Z 2044	Kickey's Space Adventure	\$69.95
Z 2045	Mission Algebra	\$59.95
Z 2046	Race Car 'Rithmetic	\$49.95
Z 2047	Remember!	\$89.95
Z 2048	Ships Ahoy	\$59.95
Z 2049	Spellagraph	\$59.95
Z 2050	Spellakazam	\$59.95
Z 2051	Spellicopter	\$59.95
Z 2052	Ten Little Robots	\$49.95
Z 2053	Winnie The Pooh in the 100 Acre Wood	\$69.95

Item #	Specialist Description	Cost
Q 1050	Side Print	\$72.00

Item #	Miscellaneous Items Description	Cost
G 2001	Box of 10 DSDD Disks	\$19.00
G 2002	10 Boxes plus (per box) Blank C30 Cassettes 12 Cassettes	\$16.10 \$2.00 \$18.00
G 2003	Tape cases, 12 for	\$5.00
G 2004	Help - Manual for CoCo	\$9.95

Item #	Modems Description	Cost
D 001	Manual 1200/75 baud plus 300/300 modem	\$240.00
D 002	IBM Half Card 1200/75 & 300/300 (Auto dial - auto answer)	\$370.00
D 003	Desktop 1200/75 & 300 /300 baud modem (Auto dial - auto answer)	\$425.00

Item #	Terminal Programs Description	Cost
G 1017	CoCoTex - Videotex pac (Viatel) for all CoCos	\$79.95
G 1018	Vtex 2 - Videotex pac for IBM Compatibles	\$225.00
G 1019	Supertex 2 for Amiga & Atari 520 ST (specify)	\$99.95

Item #	Modems/Software/Cable Description	Cost
G 005	CoCoTex with cable and manual modem	\$289.00
G 006	CoCoTex with cable and auto dial modem	\$451.00
G 007	Vtex 2 with cable and manual modem	\$385.00
G 008	Vtex 2 with half card auto modem	\$555.00
G 009	Vtex 2 with desktop auto modem and cable	\$720.00

MINIMON

32K ECB
UTILITY

by Charles Bartlett

MINIMON IS FAIRLY self explanatory so I won't go on and on about it except to say that when loaded it will as if it is to be placed in low or high memory.

After you make your selection, the menu is displayed. The numbers at the top left and right of the screen represents the area of memory that is available for a program to be loaded.

All input is to be in HEX. The relative addressing section will display the value to be used in relative addressing.

If the address is out of range it will display "LBSR" and the value to go with it.

The start address should be from the FIRST byte of the instruction.

The Listing:

```

0 GOTO10
3 SAVE"97A:3":END'8
10 ' MINIMON
   (C) C. BARTLETT 10/10/86

20 CLEAR200,&H7FFF:FOR X=&H03B6
TO &H03BD:READMS:POKEK,VAL("&H"+
MS):NEXT:EXEC&H03B6:GOSUB500:CLE
AR200,RM:EXEC&H03B6:DATAACC,0E,01
,1F,02,7E,96,A5:POKE359,57:SCREE
NO,1
30 J=0:CLS3:PRINT@13,"MINIMON";:
PRINT@68,"1 = EDIT MEMORY BLO
CK";:PRINT@100,"2 = RELATIVE AD
DRESSING";:PRINT@132,"3 = SET
A BREAK POINT";:PRINT@164,"4 =
CLEAR A BREAK POINT";:PRINT@196
,"5 = JUMP TO AN ADDRESS";
40 PRINT@228,"6 = SAVE A MEMORY
BLOCK";:PRINT@260,"7 = LOAD A
MEMORY BLOCK";:PRINT@292,"8 = C
ONVERT HEX TO BIN";:PRINT@324,"
9 = CONVERT BIN TO HEX";:PRINT
@356,"0 = DUMP A MEMORY BLOCK";
50 IFPEEK(&H03B7)=&H0E THENL1$="
2000":L2$="7FFF"ELSEL1$="0E01":L
2$="6E00"
60 PRINT@0,L1$;:PRINT@28,L2$;:GO
SUB470:J=-1:I=VAL(N$):ON I+1 GOS
UB390,70,130,210,230,240,250,260
,320,360:GOTO30
70 CLS:PRINT"EDIT ";:GOSUB300:FO
RX=0 TO65535:GOSUB110:FORL=1TO2
80 GOSUB470:IFN$="-"THEN X=X-1:P
RINT:GOSUB110:GOTO80ELSE IFN$="
"THENN$="+":X=X+1:PRINT:GOSUB110
:GOTO80ELSEIFN$="Q"THEN100ELSE A
D$=AD$+N$
90 PRINTN$;:NEXTL:A=VAL("&H"+AD$
):AD$="":POKEK,A:PRINT:NEXTX
100 RETURN
110 H$=RIGHT$("0000"+HEX$(X),4):
HP$=RIGHT$("00"+HEX$(PEEK(X)),2)
:PRINTH$;" ";HP$;" *";:GOSUB12
0:RETURN
120 CP=PEEK(&H88)*256+PEEK(&H89)
:CP=CP-2:C1=INT(CP/256):C2=CP-(C
1*256):POKE&H88,C1:POKE&H89,C2:R
ETURN
130 CLS
140 RL$=" BR ":L=2:INPUT"JUMP FR
OM ";RS$:IF RS$="Q"THEN RETURN
150 INPUT"JUMP TO ";RE$:RS=VAL
("&H"+RS$):RE=VAL("&H"+RE$):IF R
E>RS THEN RD=RE-(RS+2):IF RD>127
THEN 180
160 IF RE<RS THEN RD=(RS+2)-RE:
RD=256-RD:IF RD<0 THEN 190
170 GOTO200
180 L=4:RD=RE-(RS+4):RL$="LBR ":
GOTO200
190 L=4:RD=(RS+4)-RE:RD=65536-RD
:RL$="LBR "
200 PRINTRL$;RIGHT$("0000"+HEX$(
RD),L):PRINT:GOTO140
210 CLS:GOSUB310:PRINT:PRINT"BRE
AKPOINT ";:GOSUB300
220 BO(T)=0:BS(T)=PEEK(BO(T)):PO
KE BO(T),&H39:RETURN
230 CLS:GOSUB310:POKE BO(T),BS(T
):PRINT:PRINT"BREAKPOINT CLEARED
":FOR D=1TO 1000:NEXTD:RETURN
240 CLS:PRINT"JUMP ";:GOSUB300:D
EFUSRO=0:X=USRO(0):GOTO30
250 CLS:GOSUB460:GOSUB450:INPUT
"BLOCK ENTRY ";N$:S=VAL("&H"+11$
):E=VAL("&H"+12$):N=VAL("&H"+N$):
PRINT:PRINT"SAVING ";NMS:SAVEN
M$,S,E,N:GOTO30
260 F=0:CLS:GOSUB460:PRINT"OFFSE
T (Y/N) ";
270 GOSUB470:IFN$="N"THEN290ELSE
IFN$="Y"THEN280ELSE270
280 PRINT:INPUT"OFFSET ";F$:F=VA
L("&H"+F$)
290 LOADM NMS,F:GOTO30
300 INPUT"ORG ";O$:O=VAL("&H"+O$
):RETURN
310 PRINT"BREAKPOINT NO. (0-9) "
;:GOSUB470:T=VAL(N$):RETURN
320 CLS
330 AA$="":DV=128:INPUT"HEX NUMB
ER ";N$:NU=VAL("&H"+N$):IFNU=0
THENRETURN
340 FORW=1TO8:A$=RIGHT$(STR$(ABS
(DV<=NU)),1):IF A$="1" THEN NU=N
U-DV
350 AA$=AA$+A$:DV=INT(DV/2):NEXT
V:PRINTAA$:GOTO330
360 CLS
370 NU=0:DV=128:INPUT"BINARY NUM
BER ";B$:IF B$=""THEN RETURN ELS
E IF LEN(B$)<>8 THEN PRINT"INVAL
ID - MUST HAVE 8 DIGITS":GOTO370
380 FOR W=1TO8:NU=NU+VAL(MID$(B$
,W,1))*DV:DV=DV/2:NEXT:NU$=RIGHT
$("00"+HEX$(NU),2):PRINTNU$:GOTO
370
390 CLS:GOSUB450:PS=VAL("&H"+11$
):PE=VAL("&H"+12$):PRINT"STANDBY
"
400 FOR X=PS TO PE STEP 16:PRINT
#-2,RIGHT$("0000"+HEX$(X),4);"
";:FOR Y=X TO X+15 STEP 2:PRINT
#-2,RIGHT$("0"+HEX$(PEEK(Y)),2);
RIGHT$("0"+HEX$(PEEK(Y+1)),2);"
";
410 C1=PEEK(Y):C2=PEEK(Y+1):IF(C
1>127 OR C1<32)THEN C1=46
420 IF(C2>127 OR C2<32)THEN C2=4
6
430 CL$=CL$+CHR$(C1)+CHR$(C2)
440 NEXTY:PRINT#-2," ";CL$:CL$
="":NEXTX:RETURN
450 INPUT"BLOCK START ";I1$:INPU
T"BLOCK END ";I2$:RETURN
460 INPUT"FILENAME ";NMS:RETURN
470 N$=INKEY$:GOSUB480:IFN$=""TH
EN470ELSEBRETURN
480 IFJ=-1THENRETURNELSEIFJ=0THE
NJ=1:PRINT@429,"select";:GOSUB49
0:RETURNELSEIFJ=1THENJ=0:PRINT@4
29,"SELECT";:GOSUB490:RETURN
490 FOR U=1TO100:NXTU:RETURN
500 J=-1:CLS:PRINT"MINIMON TO LO
OR HI MEM (L/H) ";:GOSUB470:IFN
$="L"THEN RM=&H1FFF:RETURN ELSE
IFN$="H" THEN POKE&H03B7,&H0E:RM
=&H7FFF:RETURN ELSE 500

```

DIRECTORY and MEMOPAD UTILITY

DISK UTILITY

32K ECB

by Robert Seaburn

AFTER RUNNING "DIRECTOR" from disk, you are then presented with a graphics screen as shown in the following screen dump.

First, using the right joystick, select which drive you would like your next program loaded from (0-3).

Move the cursor over the corresponding disk number and press the fire-button.

Next move cursor to any part of the box containing your program name and drawing, by pressing the fire button again.

The program is LOADED and RUN.

Note: For ML programs write a short basic loader or enter exec address after load statement.

From my screen dump you can see what I have on this program. You can add or delete your own titles and drawings.

titles of programs go in line 130 and drawings of these titles go from line 190 onwards.

I have a small memopad utility that can be loaded from the Director, which is explained below.

Memopad running instructions:

1) On start up you are prompted for the date (dd/mm/yy).

2) The main screen appears, again select which drive you

want to load and store you data from or to.

3) select the month you wish to work with by moving cursor over the name of the month and pressing fire button. The month will appear at bottom of screen next to 'current file'.

4) move cursor to the menu box and press fire button. The menu will appear. From the menu you can select any of six routines. They are:

a) return- returns to main screen.

b) load file- will load the current file.

c) save file- will save the current file.

d) kill file- you are asked if you are sure, just press fire button on either yes or no.

e) enter data- you are asked whether you want to view all data or enter new data, press fire button over your selection.

e1) view all data will list all data from the current file if loaded or just move to the input routine if no data has been loaded. at the end of each full page you are asked if you want a new page or not, press fire button over your selection,

MEMO PAD

MENU		DATE 29/04/87	
JANUARY	FEBRUARY	MARCH	APRIL
MAY	JUNE	JULY	AUGUST
SEPTEMBER	OCTOBER	NOVEMBER	DECEMBER

CURRENT FILE-

DIRECTORY

MEMO PAD																				
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> </table>																				

new page will wipe existing page and continue with file data on next page.

e2) new data calculates how many pages there is and displays the last page only and goes to input routine. at input routine, just enter the data ie. appointments, events etc. try to keep them under one line long. when finished entering data press "/" on the next line to return to input screen to view your file data.

f) end program- this will re-RUN "director".

If you try to load, save, kill or enter data with no file selected a message 'no file selected' will appear and you are returned to cursor mode to try again. continued overpage

The Listing:

```
0 GOTO10
1 ***** DIRECTORY *****
   **** ROBERT SEABURN *****
3 SAVE"224:3":END'8
10 D=0:DRIVE DEFAULT
20 DIM C$(1),MC(255),MD(255),C1(
1),C2(1),C3(1),MI(255)
30 C$="NF2H2ND3R3":D$="U27R27D3L
2D2R2D2L2L27BE2BR9R5U8L5ND8BU2F1R
3E2U3H2L3G2D3F1":F$="U31R38D21WG
10L10D10L28BF1R38U31":B$="V31L10
004B"
40 PMODE4,1:SCREEN1,1:COLOR0,1:P
CLS
50 LINE(15,25)-(211,176),PSET,B:
LINE(17,27)-(209,174),PSET,B
60 LINE(81,27)-(145,174),PSET,B
70 LINE(17,76)-(209,125),PSET,B
80 'DRAW DISKS
90 X=220:Y=37
100 DRAW"BM=X; ,=Y;XD$;"
110 Y=Y+48:IFY>182THEN120ELSE100
120 A=214:TX=A:TY=26:T$="0":GOSU
B270:TX=A:TY=74:T$="1":GOSUB270:
TX=A:TY=120:T$="2":GOSUB270:TX=A
:TY=168:T$="3":GOSUB270
130 TX=30:TY=33:T$="MEMO PAD":GO
SUB270:TX=100:T$="":GOSUB2
70:TX=160:T$="":GOSUB270:
TX=25:TY=82:T$="":GOSU
B270:TX=90:T$="":GOSUB
270:TX=148:T$="":GOS
UB270
140 'ENTER PROGRAM NAMES BETWEEN
SPEECH MARKS
150 REM ENTER MORE SCREEN BLOCKS
HERE
160 DRAW"BM56,16S8":TX=71:TY=16:
T$="DIRECTORY":GOSUB180:DRAW"BM0
,0S4"
170 GOTO 190
180 FOR C=1 TO LEN(T$):DRAW"BM=T
X; ,=TY;":A$=MID$(T$,C,1):GOSUB 2
50:TX=TX+10:NEXT C:RETURN
190 DRAW"BM19,60U24R48D36L48U12R
48U12L48BE12D36R12U36R13D36R12BR
3NU6D2R8U8L8BU3NU8R8U8L8BU3NU8R8
U8L8BU3R8U8L8D8"
200 'DRAW SCREEN PICTURES AS IN
190 FROM HERE
210 GOSUB910
220 GOSUB870:GOSUB930:GOSUB940:1
F P THENPLAYB$:GOSUB960:GOTO220
230 GOTO220
240 ' WORDS
250 IF ASC(A$)>62 THEN 260 ELSE
ON ASC(A$)-31 GOTO 280,290,300,3
10,320,330,340,350,360,370,380,3
90,400,410,420,430,440,450,460,4
70,480,490,500,510,520,530,540,5
50,560,570,580
260 ON ASC(A$)-62 GOTO 590,600,6
10,620,630,640,650,660,670,680,6
90,700,710,720,730,740,750,760,7
70,780,790,800,810,820,830,840,8
50,860
270 FOR C=1 TO LEN(T$):DRAW"BM=T
X; ,=TY;":A$=MID$(T$,C,1):GOSUB 2
50:TX=TX+5:NEXT C:RETURN
280 DRAW"BR4":RETURN
290 DRAW"BU2U2LD2":RETURN
300 DRAW"BU2U2BR2D2":RETURN
310 DRAW"BRU4BR2D4URL4RU2LR4":RE
TURN
320 DRAW"RU2L3U2R3LD4":RETURN
330 DRAW"E4BL3DBR3BD2D":RETURN
340 DRAW"BR2NU4R2R3":RETURN
350 DRAW"BRBU4D2":RETURN
360 DRAW"BR2HU2E":RETURN
370 DRAW"BRBU2H":RETURN
380 DRAW"BU2GHF2":RETURN
390 DRAW"BU2R2LUD2":RETURN
400 DRAW"BRU2":RETURN
410 DRAW"BU2R3":RETURN
420 DRAW"BRURD":RETURN
430 DRAW"UE2U":RETURN
440 DRAW"U4R3D4L3":RETURN
450 DRAW"R2LU4G":RETURN
460 DRAW"NR3U2R3U2L3":RETURN
470 DRAW"RU2NL2U2L3":RETURN
480 DRAW"BU4D2R3U2D4":RETURN
490 DRAW"RU2L3U2R3":RETURN
500 DRAW"NU2R3U2L3U2R3":RETURN
510 DRAW"BU4R3DGD2":RETURN
520 DRAW"NU4R3U2NL3U2L3":RETURN
530 DRAW"RU2L3U2R3D2":RETURN
540 DRAW"RULBU2RUL":RETURN
550 DRAW"BU4RDLBD2RD":RETURN
560 DRAW"BU2NE2F2":RETURN
570 DRAW"BUR3BU2L3":RETURN
580 DRAW"E2H2":RETURN
590 DRAW"BU3UR3D2LBDD":RETURN
600 DRAW"NR3U3ER2BD2LDRU":RETURN
610 DRAW"U3ERFDNL3D2":RETURN
620 DRAW"U4R2FGN2FGL2":RETURN
630 DRAW"BRREGLHU2ERF":RETURN
640 DRAW"U4R2FD2GL2":RETURN
650 DRAW"NR3U2NR2U2R3":RETURN
660 DRAW"U2NR2U2R3":RETURN
670 DRAW"BU4BR3L2GD2FREUL":RETUR
N
680 DRAW"U4D2R3U2D4":RETURN
690 DRAW"R2LU4NL":RETURN
700 DRAW"BUFREU3":RETURN
710 DRAW"U2RNF2NE2LU2":RETURN
720 DRAW"NR3U4":RETURN
730 DRAW"U4FREBD4":RETURN
740 DRAW"U4F3DU4":RETURN
750 DRAW"BUU2ERFD2GLH":RETURN
760 DRAW"U4R2FGL2":RETURN
770 DRAW"BUU2ERFD2GNUNRLH":RETURN
780 DRAW"U4R2FGL2RF2":RETURN
790 DRAW"R2EHLHER2":RETURN
800 DRAW"BU4R2LD4":RETURN
810 DRAW"NU4R3U4":RETURN
820 DRAW"BU4D3FREU3":RETURN
830 DRAW"NU4ERFU4":RETURN
840 DRAW"UE2UDGHU2DF2":RETURN
850 DRAW"BU4DFEUDGD2":RETURN
860 DRAW"NR2UE2UL2":RETURN
870 X=JOYSTK(0)*4:Y=JOYSTK(1)*4:
P=ABS((1 AND PEEK(&HFF00))*3-3):
IF X<10 THEN X=10 ELSE IF X>255
THEN X=255
880 IF Y<10 THEN Y=10 ELSE IF Y>
175 THEN Y=175
890 RETURN
900 'DRAW CURSOR
910 GET(0,0)-(6,6),C1,G:LINE(0,0
)-(6,6),PSET,BF:DRAW"BM3,3XC$;
":GET(0,0)-(6,6),C2,G:PUT(0,0)-(
6,6),C1,PSET:RETURN
920 'MOVE CURSOR
930 GET(X-3,Y-3)-(X+3,Y+3),C1,G:
PUT(X-3,Y-3)-(X+3,Y+3),C2,AND:RE
TURN
940 PUT(X-3,Y-3)-(X+3,Y+3),C1,PS
ET:RETURN
950 'SET FUNCTIONS
960 IFX>220ANDX<247ANDY>10ANDY<3
7THENPAINT(222,15),0,0:D=0:LINE(
220,58)-(247,85),PSET,BF:DRAW"
BM220,85;XD$;":LINE(220,106)-(24
7,133),PSET,BF:DRAW"BM220,133;
XD$;":LINE(220,154)-(247,181),PR
ESET,BF:DRAW"BM220,181;XD$;":RET
URN
970 IFX>220ANDX<247ANDY>57ANDY<8
5THENPAINT(222,60),0,0:D=1:LINE(
220,10)-(247,37),PSET,BF:DRAW"
BM220,37;XD$;":LINE(220,106)-(24
7,133),PSET,BF:DRAW"BM220,133;
XD$;":LINE(220,154)-(247,182),PR
ESET,BF:DRAW"BM220,181;XD$;":RET
URN
980 IFX>220ANDX<247ANDY>106ANDY<
133THENPAINT(222,110),0,0:D=2:LI
NE(220,10)-(247,37),PSET,BF:DR
AW"BM220,37;XD$;":LINE(220,57)-(
247,85),PSET,BF:DRAW"BM220,85;
XD$;":LINE(220,154)-(247,182),PR
ESET,BF:DRAW"BM220,181;XD$;":RET
URN
990 IFX>220ANDX<247ANDY>154ANDY<
182THENPAINT(222,160),0,0:D=3:LI
NE(220,10)-(247,37),PSET,BF:DR
AW"BM220,37;XD$;":LINE(220,57)-(
247,85),PSET,BF:DRAW"BM220,85;
XD$;":LINE(220,106)-(247,133),PR
ESET,BF:DRAW"BM220,133;XD$;":RET
URN
1000 IFX>17ANDX<80ANDY>27ANDY<76
THEN T=1:GOTO1100
1010 IFX>82ANDX<144ANDY>27ANDY<7
6THEN T=2:GOTO1100
1020 IFX>146ANDX<208ANDY>27ANDY<
76THEN T=3:GOTO1100
1030 IFX>17ANDX<80ANDY>77ANDY<12
4THEN T=4:GOTO1100
1040 IFX>82ANDX<144ANDY>77ANDY<1
24THEN T=5:GOTO1100
1050 IFX>146ANDX<208ANDY>77ANDY<
124THEN T=6:GOTO1100
1060 REM IFX>17ANDX<80ANDY>126AN
DY<173THEN T=7:GOTO4000
1070 REM IFX>82ANDX<144ANDY>126A
NDY<173THEN T=8:GOTO4000
1080 REM IFX>146ANDX<208ANDY>126
ANDY<173THEN T=9:GOTO4000
1090 RETURN
1100 GET(76,92)-(155,108),MD,G:L
INE(76,92)-(155,108),PSET,BF:L
INE(76,92)-(154,107),PSET,B:LINE
(77,108)-(155,108),PSET:LINE-(15
5,93),PSET:TX=79:TY=98:T$="LOADI
NG PROGRAM":GOSUB270:TX=104:TY=1
05:T$="NOW":GOSUB270
1110 DRIVE D
1120 IFT=1THENRUN"MEMO PAD"
1130 IFT=2THENRUN"
1140 IFT=3THENRUN"
1150 IFT=4THENRUN"
1160 IFT=5THENRUN"
1170 IFT=6THENRUN"
1180 IFT=7THENRUN"
1190 IFT=8THENRUN"
1200 IFT=9THENRUN"
1210 GOTO220
```

The Listing:

```
1 CLEAR500:POKE65495,0:GOTO5
3 SAVE"224A:3":END'8
5 PF=1
10 CLS:PRINT@204,"MEMO PAD":PRIN
T@239,"BY":PRINT@267,"R. SEABURN
":PRINT@300,"(C) 1986"
20 PRINT@355,"":INPUT"DATE (DD/
MM/YY)":DA$
30 D=0:'DRIVE DEFAULT
40 DIM B$(100),C$(1),A(8),AD(8),
C(255),C1(1),C2(1),C3(1),L1(6),L
2(6),L3(6),L4(6),CM(15),CO(15),M
D(255),MO(255),MI(255)
50 C$="NF2H2ND3R3":D$="U27R27D3L
2D2R2D2L27BE2BR9R5U8L5ND8BU2F1R
3E2U3H2L3G2D3F1":F$="U31R38D21G
10L10D10L28BF1R38U31":Z$="V31L10
004B"
60 PMODE4,1:SCREEN1,1:COLOR0,1:P
CLS
70 ' MAIN SCREEN
80 FL$=""
90 LINE(6,26)-(212,190),PRESET,B
F:GOSUB100:GOSUB150:GOSUB190:GOS
UB250:GOSUB1010:GOTO300
100 LINE(15,25)-(211,176),PSET,B
:LINE(17,27)-(209,174),PSET,B
110 LINE(65,27)-(161,174),PSET,B
120 LINE(17,76)-(209,125),PSET,B
130 LINE(113,27)-(113,174),PSET:
RETURN
140 'DRAW FILES
150 X=22:Y=67
160 DRAW"BM=X; ,Y;XF$;"
170 X=X+48:IFX>172 THEN180ELSE16
0
180 X=22:Y=Y+49:IFY>165THENRETUR
NELSE160
190 'DRAW DISKS
200 X=220:Y=37
210 DRAW"BM=X; ,Y;XD$;"
220 Y=Y+48:IFY>182THEN230ELSE210
230 REM
240 TX=214:TY=26:T$=""0":GOSUB370
:TX=214:TY=74:T$="1":GOSUB370:TX
=214:TY=120:T$="2":GOSUB370:TX=2
14:TY=168:T$="3":GOSUB370:RETURN
250 FORZ=1TO13:READA,B,E$:TX=A:T
Y=B:T$=E$:GOSUB370:NEXTZ:RESTORE
260 'MONTHS
270 DRAW"BM90,10S8U4FRED4BR1NR3U
2NR2U2R3BR1ND4FRED4BR1BUU2ERFD2G
LHBD2BR5BU1BR5U4R2FGL2BF2BR1U3ER
FDNL3D2BR1U4R2FD2GL2":DRAW"BM0,0
S4"
280 TX=130:TY=185:T$=FL$:GOSUB37
0
290 TX=34:TY=19:T$="MENU":GOSUB3
70:LINE(15,13)-(74,22),PSET,B:LI
NE(16,23)-(75,23),PSET:LINE-(75,
14),PSET:LINE(120,13)-(211,22),P
SET,B:LINE(121,23)-(212,23),PSET
:LINE-(212,14),PSET:TX=135:TY=19
:T$="DATE":GOSUB370:TX=160:TY=19
:T$=DA$:GOSUB370:RETURN
300 GOSUB970:GOSUB1030:GOSUB1040
:IF P THENPLAYZ$:GOSUB1060:GOTO3
00
310 GOTO300
320 DATA 23,33,JANUARY,71,33,FEB
RUARY,119,33,MARCH,167,33,APRIL,
22,83,MAY,71,83,JUNE
330 DATA 119,83,JULY,167,83,AUGU
ST,20,132,SEPTEMBER,71,132,OCTOB
ER,119,132,NOVEMBER,167,132,DECE
MBER,60,185,CURRENT FILE-
340 'DRAW WORDS
350 IF ASC(A$)>62 THEN 360 ELSE
ON ASC(A$)-31 GOTO 380,390,400,
410,420,430,440,450,460,470,480,
490,500,510,520,530,540,550,560,
570,580,590,600,610,620,630,640,
650,660,670,680
360 ON ASC(A$)-62 GOTO 690,700,
710,720,730,740,750,760,770,780,
790,800,810,820,830,840,850,860,
870,880,890,900,910,920,930,940,
950,960
370 FOR C=1 TO LEN(T$):DRAW"BM-T
X; ,TY;":A$=MID$(T$,C,1):GOSUB 3
50:TX=TX+5:NEXT C:RETURN
380 DRAW"BR4":RETURN
390 DRAW"RB2U2LD2":RETURN
400 DRAW"BU2U2BR2D2":RETURN
410 DRAW"BR4BR2D4URL4RU2LR4":RE
TURN
420 DRAW"R3U2L3U2R3LD4":RETURN
430 DRAW"E4BL3DBR3BD2D":RETURN
440 DRAW"BR2NU4H2R3":RETURN
450 DRAW"BRBU4D2":RETURN
460 DRAW"BR2HU2E":RETURN
470 DRAW"BRU2H":RETURN
480 DRAW"BU2GHF2":RETURN
490 DRAW"BU2R2LUD2":RETURN
500 DRAW"BRU2":RETURN
510 DRAW"BU2R3":RETURN
520 DRAW"BRURD":RETURN
530 DRAW"UE2U":RETURN
540 DRAW"U4R3D4L3":RETURN
550 DRAW"R2LU4G":RETURN
560 DRAW"NR3U2R3U2L3":RETURN
570 DRAW"R3U2NL2U2L3":RETURN
580 DRAW"BU4D2R3U2D4":RETURN
590 DRAW"R3U2L3U2R3":RETURN
600 DRAW"NU2R3U2L3U2R3":RETURN
610 DRAW"BU4R3DG2D":RETURN
620 DRAW"NU4R3U2NL3U2L3":RETURN
630 DRAW"R3U2L3U2R3D2":RETURN
640 DRAW"RULBU2RUL":RETURN
650 DRAW"BU4RDLBD2RD":RETURN
660 DRAW"BU2NE2F2":RETURN
670 DRAW"BR3BU2L3":RETURN
680 DRAW"E2H2":RETURN
690 DRAW"BU3UR3D2LBD":RETURN
700 DRAW"NR3U3ER2BD2LDRU":RETURN
710 DRAW"U3ERFDNL3D2":RETURN
720 DRAW"U4R2FGL2FGL2":RETURN
730 DRAW"BRREGLHU2ERF":RETURN
740 DRAW"U4R2FD2GL2":RETURN
750 DRAW"NR3U2NR2U2R3":RETURN
760 DRAW"U2NR2U2R3":RETURN
770 DRAW"BU4BR3L2GD2FREUL":RETUR
N
780 DRAW"U4D2R3U2D4":RETURN
790 DRAW"R2LU4NLR":RETURN
800 DRAW"BUFREU3":RETURN
810 DRAW"U2RNF2NE2LU2":RETURN
820 DRAW"NR3U4":RETURN
830 DRAW"U4FRED4":RETURN
840 DRAW"U4F3DU4":RETURN
850 DRAW"BU2ERFD2GLH":RETURN
860 DRAW"U4R2FGL2":RETURN
870 DRAW"BU2ERFD2GNUNRLH":RETUR
N
880 DRAW"U4R2FGL2RF2":RETURN
890 DRAW"R2EHLHER2":RETURN
900 DRAW"BU4R2LD4":RETURN
910 DRAW"NU4R3U4":RETURN
920 DRAW"BU4D3FREU3":RETURN
930 DRAW"NU4ERFU4":RETURN
940 DRAW"UE2UDGHUFD2D":RETURN
950 DRAW"BU4DFEUDGD2":RETURN
960 DRAW"NR2UE2UL2":RETURN
970 X=JOYSTK(0)*4:Y=JOYSTK(1)*4:
P=ABS((1 AND PEEK(&HFF00))*3-3):
IF X<10 THEN X=10 ELSE IF X>255
THEN X=255
980 IF Y<10 THEN Y=10 ELSE IF Y>
175 THEN Y=175
990 RETURN
1000 'DRAW CURSOR
1010 GET(0,0)-(6,6),C1,G:LINE(0,
0)-(6,6),PRESET,BF:DRAW"BM3,3XC$
;":GET(0,0)-(6,6),C2,G:PUT(0,0)-
(6,6),C1,PSET:RETURN
1020 'MOVE CURSOR
1030 GET(X-3,Y-3)-(X+3,Y+3),C1,G
:PUT(X-3,Y-3)-(X+3,Y+3),C2,AND:R
ETURN
1040 PUT(X-3,Y-3)-(X+3,Y+3),C1,P
SET:RETURN
1050 'SET FUNCTIONS
1060 IFX>220ANDX<247ANDY>10ANDY<
37THENPAINT(222,15),0,0:D=0:LINE
(220,58)-(247,85),PRESET,BF:DRAW
"BM220,85;XD$;":LINE(220,106)-(2
47,133),PRESET,BF:DRAW"BM220,133
;XD$;":LINE(220,154)-(247,182),P
RESET,BF:DRAW"BM220,181;XD$;":RE
TURN
1070 IFX>220ANDX<247ANDY>57ANDY<
85THENPAINT(222,60),0,0:D=1:LINE
(220,10)-(247,37),PRESET,BF:DRAW
"BM220,37;XD$;":LINE(220,106)-(2
47,133),PRESET,BF:DRAW"BM220,133
;XD$;":LINE(220,154)-(247,182),P
RESET,BF:DRAW"BM220,181;XD$;":RE
TURN
1080 IFX>220ANDX<247ANDY>106ANDY<
133THENPAINT(222,110),0,0:D=2:L
INE(220,10)-(247,37),PRESET,BF:D
RAW"BM220,37;XD$;":LINE(220,57)-
(247,85),PRESET,BF:DRAW"BM220,85
;XD$;":LINE(220,154)-(247,182),P
RESET,BF:DRAW"BM220,181;XD$;":RE
TURN
1090 IFX>220ANDX<247ANDY>154ANDY<
182THENPAINT(222,160),0,0:D=3:L
INE(220,10)-(247,37),PRESET,BF:D
RAW"BM220,37;XD$;":LINE(220,57)-
(247,85),PRESET,BF:DRAW"BM220,85
;XD$;":LINE(220,106)-(247,133),P
RESET,BF:DRAW"BM220,133;XD$;":RE
TURN
1100 IFX>27ANDX<58ANDY>13ANDY<22
THENGOSUB1380:RETURN
1110 IFX>18ANDX<50ANDY>24ANDY<29
THENPUT(15,22)-(75,84),MO,PSET:I
FT=1THENGOTO90ELSERETURN
1120 IFX>18ANDX<63ANDY>33ANDY<39
THENGOSUB1700
1130 IFX>18ANDX<63ANDY>43ANDY<49
THENGOSUB1430
1140 IFX>18ANDX<63ANDY>53ANDY<59
THENGOSUB1520
1150 IFX>18ANDX<68ANDY>63ANDY<69
THENGOSUB1580
1160 IFX>18ANDX<72ANDY>73ANDY<79
```

```

THENPOKE65494,0:RUN"DIRECTOR"
1170 IFX>22ANDX<60ANDY>27ANDY<34
THENFLS="JAN":GOSUB1360:RETURN
1180 IFX>70ANDX<108ANDY>27ANDY<34
THENFLS="FEB":GOSUB1360:RETURN
1190 IFX>118ANDX<156ANDY>27ANDY<34
THENFLS="MAR":GOSUB1360:RETURN
1200 IFX>167ANDX<205ANDY>27ANDY<34
THENFLS="APR":GOSUB1360:RETURN
1210 IFX>22ANDX<60ANDY>77ANDY<86
THENFLS="MAY":GOSUB1360:RETURN
1220 IFX>70ANDX<108ANDY>77ANDY<86
THENFLS="JUN":GOSUB1360:RETURN
1230 IFX>118ANDX<156ANDY>77ANDY<86
THENFLS="JUL":GOSUB1360:RETURN
1240 IFX>167ANDX<205ANDY>77ANDY<86
THENFLS="AUG":GOSUB1360:RETURN
1250 IFX>22ANDX<60ANDY>126ANDY<133
THENFLS="SEP":GOSUB1360:RETURN
1260 IFX>70ANDX<108ANDY>126ANDY<133
THENFLS="OCT":GOSUB1360:RETURN
1270 IFX>118ANDX<156ANDY>126ANDY<133
THENFLS="NOV":GOSUB1360:RETURN
1280 IFX>167ANDX<205ANDY>126ANDY<133
THENFLS="DEC":GOSUB1360:RETURN
1290 IFX>85ANDX<102ANDY>87ANDY<93
THEN1560ELSEIFX>110ANDX<121ANDY>87
ANDY<93THENPUT(80,64)-(126,96),MD,
PSET:RETURN
1300 IFX>71ANDX<87ANDY>149ANDY<157
THENPUT(68,136)-(116,164),MO,PSET:
LINE(9,29)-(210,174),PRESET,BF:TT=1:
TX=12:TY=35:IFRT=1ANDN=<G
THENGOTO1680ELSEIFRT=1ANDN=>G+1
THENPG=PG+17:G=G+17:GOTO1670
1310 IFX>102ANDX<112ANDY>149ANDY<157
THENPUT(68,136)-(116,164),MO,PSET:
GOTO300
1320 IFX>84ANDX<103ANDY>111ANDY<119
THENPG=1:PUT(39,100)-(187,121),MO,
PSET:TX=12:TY=35:GOTO1660
1330 IFX>129ANDX<146ANDY>111ANDY<119
THENGOTO1920
1340 RETURN
1350 'SET FILE NAME
1360 TX=130:TY=185:LINE(125,178)-(150,
189),PRESET,BF:T$=FL$:GOSUB370:TT=0:
RETURN
1370 'DRAW MENU BOX
1380 GET(15,22)-(75,84),MO,G:LINE(15,23)-
(74,83),PRESET,BF:LINE(15,22)-(74,82),
PSET,B:LINE(16,83)-(75,83),PSET:LINE(75,14),
PSET
1390 TX=18:TY=28:T$="RETURN":GOSUB370
1400 TX=18:TY=38:T$="LOAD FILE":GOSUB370
1410 TX=18:TY=48:T$="SAVE FILE":GOSUB370:
TX=18:TY=58:T$="KILL FILE":GOSUB370:TX=18:
TY=68:T$="ENTER DATA":GOSUB370:TX=18:TY=78:
T$="END PROGRAM":GOSUB370
1420 RETURN
1430 'SAVE ROUTINE
1440 IFFLS=" "THEN1860
1450 DRIVE D
1460 OPEN"O",#1,FLS
1470 FORZ=1TO N
1480 WRITE#1,B$(Z)
1490 NEXTZ

```

```

1500 CLOSE#1
1510 RETURN
1520 'KILL FILE ROUTINE
1530 IFFLS=" "THEN1860
1540 GET(80,64)-(126,96),MD,G:LINE(81,65)-
(125,95),PRESET,BF:LINE(81,65)-(124,95),
PSET,B:LINE(82,96)-(125,96),PSET:LINE(125,66),
PSET:TX=86:TY=72:T$="ARE YOU":GOSUB370:
TX=94:TY=82:T$="SURE":GOSUB370:TX=86:TY=92:
T$="YES NO":GOSUB370
1550 RETURN
1560 KILLFLS+"/DAT":PUT(80,64)-(126,96),
MD,PSET
1570 RETURN
1580 'ENTER DATA ROUTINE
1590 PP=1:G=17
1600 IFFLS=" "THEN1860
1610 TX=12:TY=35:IFTT<>1THEN N=0
1620 PUT(15,22)-(75,84),MO,PSET
1630 LINE(6,24)-(212,176),PRESET,BF:LINE(8,27)-
(211,175),PSET,B:T=1
1640 GET(39,100)-(187,121),MO,G:LINE(39,100)-
(187,121),PRESET,BF:LINE(39,100)-(186,120),
PSET,B:LINE(40,121)-(187,121),PSET:LINE(187,101),
PSET
1650 TX=42:TY=106:T$="DO YOU VIEW H TO VIEW ALL
DATA":GOSUB370:TX=70:TY=112:T$="OR ENTER NEW
DATA":GOSUB370:TX=86:TY=118:T$="ALL NEW":
GOSUB370:GOTO300
1660 REM
1670 IFTT=1THENFORZ=PG TO N:T$=B$(Z):GOSUB370:
TX=12:TY=TY+8:IFTY>168THENRT=1:GOTO1680ELSE
NEXTZ
1680 TT=1:CLS:INPUTT$:IFT$="/"THENSCREEN1,1:
RETURNELSEN=N+1:B$(N)=T$:SCREEN1,1:GOSUB370:
TX=12:TY=TY+8:IFTY>168THENGOTO1680:RT=2ELSE
1680
1690 RETURN
1700 'LOAD ROUTINE
1710 IFFLS=" "THEN1860ELSE1720
1720 PUT(15,22)-(75,84),MO,PSET:T=1
1730 GET(81,84)-(145,96),MO,G:LINE(81,84)-
(145,96),PRESET,BF:LINE(81,84)-(144,95),
PSET,B:LINE(82,96)-(145,96),PSET:LINE(145,85),
PSET
1740 TX=86:TY=92:T$="LOADING":GOSUB370:TX=127:
T$=FL$:GOSUB370
1750 POKE65494,0
1760 TX=12:TY=35
1770 N=1
1780 DRIVE D
1790 OPEN"O",#1,FLS
1800 INPUT#1,B$(N)
1810 IF BOF(1)=-1THEN1820ELSEN=N+1:GOTO1800
1820 CLOSE#1:TT=1
1830 PUT(81,84)-(145,96),MO,PSET
1840 RETURN
1850 'CHECK FOR FILE
1860 IFFLS=" "THENGET(65,93)-(161,105),MD,G:
LINE(66,94)-(160,105),PRESET,BF:LINE(66,94)-
(160,104),PSET,B:LINE(67,105)-(161,105),PSET:
LINE(161,95),PSET:TX=73:TY=101:T$="NO FILE
SELECTED":GOSUB

```

```

370:FORZ=1TO500:NEXTZ:PUT(65,93)-(161,105),
MD,PSET:RETURN
1870 RETURN
1880 'NEW PAGE ROUTINE
1890 FORF=1TO500:NEXT
1900 GET(68,136)-(116,164),MD,G:LINE(68,136)-
(116,164),PRESET,BF:LINE(68,136)-(115,163),
PSET,B:LINE(69,164)-(116,164),PSET:LINE(116,137),
PSET
1910 TX=72:TY=142:T$="NEW PAGE":GOSUB370:TX=72:
TY=156:T$="YES NO":GOSUB370:GOTO300:LINE(6,24)-
(212,176),PRESET,BF:IFRT=1THENPP=18:GOTO1670
ELSEIFRT=2THENGOTO1680
1920 IFN<=17 THEN PG=1:TX=12:TY=35:PUT(39,100)-
(187,121),MD,PSET:GOTO1660
1930 IFN<=34 THEN PG=18:TX=12:TY=35:PUT(39,100)-
(187,121),MO,PSET:GOTO1660
1940 IFN<=51 THEN PG=35:TX=12:TY=35:PUT(39,100)-
(187,121),MO,PSET:GOTO1660
1950 IFN<=68 THEN PG=52:TX=12:TY=35:PUT(39,100)-
(187,121),MO,PSET:GOTO1660
1960 GOTO300

```

Memory Tester

```

1 'TEST YOUR MEMORY
2 CLS
3 PRINT"MEMORIZE THESE NUMBERS"
4 PRINT
5 PRINT"YOU HAVE FOUR SECONDS"
6 PRINT
7 FOR X=1 TO 5
8 N(X)=RND(40)
9 PRINTN(X)
10 NEXTX
11 FOR Y=1 TO 3000:NEXTY
12 CLS
13 FORX=1TO5
14 PRINT"WHAT WAS NUMBER ";X
15 INPUTR
16 PRINT
17 IFN(X)=R THENPRINT"CORRECT"ELSEPRINT"WRONG-IT
WAS"R(X)
18 PRINT
19 NEXTX
20 PRINT"*****"
21 FOR X=1 TO1400:NEXTX
22 CLS
23 PRINT"GO AGAIN? Y/N"
24 Z$=INKEY$:IFZ$="" GOTO24
25 IFZ$="Y"THEN GOTO2ELSE26
27 PRINT:PRINT:PRINT
28 PRINT"THANK YOU... HAVE A NICE DAY"

```

Just right for your Xmas Stocking?

STOCK LIST

32K ECB, tape only
BUSINESS

by J. Grech

STOCK LIST IS A business - orientated program designed to take care of listing your stock. Now, this program can be used in any situation anywhere! For example in small supermarkets, Tandy stores etc.

And with a bit of modification to the program, you can do other things as well, like keeping track of your albums & tapes, disks, and so on!

To go through the program (following the main menu) ...

'Inventory Inquiry' is a section to find either the stock number or a VERY short description of the stock (if you for example can't remember the stock number).

'Load Inventory' loads the data from tape. If you come here or anywhere else in the program by accident keep the bottom of the screen in mind. This will show you some of the 'other' options you have.

'Save Inventory' will save your inventory to tape.

'Edit Item no. Menu' will give you a menu. With this menu you can edit your existing stock in memory in two ways. Either by giving you the option of either deleting existing stock or altering the price of the item and/or fixing a minor detail.

'New Item Entry' will allow you to enter a new item into memory.

'Print Inventory Menu' gives you a choice of printing out to the printer a choice of what you want to print.

Mind you, don't forget that this is a TAPE ONLY program and the program itself is very big. Therefore you need to type in...
POKE25,6:POKE3584,0:NEW

The Listing:

```
0 GOTO30
1 '***** STOCK LIST *****
  **** J. GRECH *****
3 SAVE"232:3":END'3
15 'PROGRAM BY J. GRECH 1987
20 'REMEMBER TO POKE25,6:NEW
  BEFORE LOADING
30 CLEAR6000
35 DIMST$(9,200)
40 FORW=0TO200
45 CLS
50 FORY=1024TO1535
55 POKEY,32
60 NEXTY
65 FORY=1057 TO 1086
70 POKEY,35:NEXTY
75 FORY=1089 TO 1185 STEP32
80 POKEY,35:NEXTY
85 FORY=1118 TO 1214 STEP32
90 POKEY,35:NEXTY
95 FORY=1217 TO 1246
100 POKEY,35:NEXTY
105 PRINT@107,"stock list";:POKE
1136,32
110 PRINT@164,"program by j grech";:POKE1195,32:POKE1198,32:POKE
1200,32:POKE1208,49:POKE1209,57:
POKE1210,56:POKE1211,55:FORY=116
3TO1172:POKEY,45:NEXTY
115 PRINT@324,"please enter toda
ys date";:POKE1360,32:POKE1354,3
2:POKE1367,32
120 PRINT@395,"";:LINEINPUT DT$
130 CLS:GOSUB590
135 PRINT@100,"** M A I N M E
N U **"
140 PRINT@195,"1) INVENTORY INQ
UIRY"
145 PRINT@227,"2) LOAD INVENTOR
Y"
150 PRINT@259,"3) SAVE INVENTOR
Y"
155 PRINT@291,"4) EDIT ITEM NO.
MENU"
160 PRINT@323,"5) NEW ITEM ENTR
Y"
165 PRINT@355,"6) PRINT INVENTO
RY MENU"
170 PRINT@449,"PLEASE ENTER MENU
NUMBER.";:INPUT">";A
175 ON A GOTO 185,1565,1650,610,
905,1025
180 GOTO170
185 CLS
190 GOSUB590
195 PRINT@100,"ITEM SELECTION"
200 PRINT@193,"ITEM NO.
```

```
-----
-- OR"
205 PRINT@321,"SEARCH
DESCRP. -----
---"
210 PRINT@449,"AVAILABLE COMMANDS
"
215 PRINT@202,"";:LINEINPUTZ$
220 FORW=1TO200
225 IFZ$=""THEN255ELSEIFZ$=ST$(1
,V)THEN290
230 NEXTW
235 PRINT@295,"item no. not on f
ile";:POKE1323,32:POKE1326,46:PO
KE1327,32:POKE1331,32:POKE1334,3
2
240 PRINT@481,"<3>NEW ITEM <7>E
ND OF INQUIRY";
245 EXEC44539:A$=INKEY$:IFA$="3"
THEN185ELSE250
250 IFA$="7"THEN130ELSE245GOTO24
5
255 PRINT@329,"";:LINEINPUTS$
260 IFS$=""THEN200
265 CLS
270 GOSUB590
275 PRINT@100,"SEARCH";:PRINT@11
3,""S$"";
280 PRINT@160,;:GOTO475
285 GOTO285
290 CLS:C$="ITEM DETAIL":GOSUB29
5:GOTO355
295 PRINT" COLOUR COMPUTER STOCK
CONTROL"
300 PRINT@33,STRING$(30,45)
305 PRINT@66,C$;:PRINT@85,DT$
310 PRINT@97,STRING$(30,45):PRIN
T@417,STRING$(30,45)
315 PRINT@129,"ITEM NO. "ST$(1,W
);
320 PRINT@161,"ITEM. "ST$(2,W);
325 PRINT@193,"RETAIL.....
.$ "ST$(3,W)
330 PRINT@225,"COST.....
.$ "ST$(4,W);
335 PRINT@257,"TRADE DIS.....
.% "ST$(5,W);
340 PRINT@289,"WHOLESALE DIS....
.% "ST$(6,W);
345 PRINT@321,"SALES TAX.....
.% "ST$(7,W);
350 PRINT@353,"SUPPLIER. "ST$(8,
W);:RETURN
355 PRINT@449,"AVAILABLE COMMANDS
"
360 PRINT@481,"<3>NEW ITEM <7>E
ND OF INQUIRY";
365 PRINT@385,"<ENTER> FOR STOCK
DETAIL";
```



```

S"
1010 PRINT@481,"<3>NEW ITEM <7>
END OF INPUT";
1015 EXEC44539;
1020 A$=INKEY$:IFAS="7"THEN130EL
SB915
1025 CLS:GOSUB590
1030 PRINT@105,"PRINTOUT MENU"
1035 PRINT@196,"1) RETAIL PRIN
TOUT"
1040 PRINT@228,"2) COST PRINTO
UT"
1045 PRINT@260,"3) QTY STOCK O
N HAND"
1050 PRINT@292,"4) PRINTOUT BY
SUPPLIER"
1055 PRINT@324,"5) ON HAND BY
SUPPLIER"
1060 PRINT@356,"6) RETURN TO M
AIN MENU"
1065 PRINT@449,"PLEASE ENTER MEN
U NUMBER"

1070 FORV=1TO200
1075 EXEC44539:IS=INKEY$
1080 IFIS=""THEN1075
1085 IFIS="1"THEN1120
1090 IFIS="2"THEN1230
1095 IFIS="3"THEN1295
1100 IFIS="4"THEN1395
1105 IFIS="5"THEN1485
1110 IFIS="6"THEN130
1115 GOTO1075
1120 CLS:GOSUB590
1125 PRINT@105,"RETAIL PRINTOUT"
1130 PRINT@193,"A RETAIL PRINTOU
T, PRINTS THE ITEM NO.; DESCRI
PTION; RETAIL PRICE; TRADE DIS
%; W/SALE DIS%; TAX... IN THAT
ORDER OF ALL ITEMS IN MEMORY"
1135 PRINT@449,"AVAILABLE COMMAND
S"
1140 PRINT@481,"<P>FOR PRINTOUT
<3>FOR MENU";
1145 EXEC44539:IS=INKEY$:IFIS=""
THEN1145
1150 IFIS="3"THEN1025
1155 IFIS="P"THEN PT=PEEK(&HFF22
)AND 1:GOTO1165
1160 GOTO1145
1165 IFPT=1THEN PRINT@386,"PRINTE
R OFFLINE":GOTO1145
1170 PRINT@386,"";:PRINT@481,"
";

1175 GOSUB1180:GOTO1185
1180 PRINT#-2,"ITEM CODE DES
CRPTION";TAB(44);"RETAIL COST
TRADE% W/SALE% TAX%":PRINT#-2
,CHR$(28)CHR$(80)CHR$(45):PRIN
T#-2,"DATE... ";DT$:PRINT#-2,""
:RETURN
1185 PRINT#-2,ST$(1,W);TAB(13);S
T$(2,W);TAB(45);ST$(3,W);TAB(63)
;ST$(5,W);TAB(71);ST$(6,W);TAB(7
6);ST$(7,W):GOSUB1190:GOTO1185
1190 W=W+1:IFW=200THEN1195ELSE12
00
1195 PRINT#-2,CHR$(28)CHR$(31) "
*";" END OF PRINTOUT ";CHR$(28)C
HR$(32) "*" :GOTO1025
1200 IFST$(1,W)=""THEN1190ELSE12
05
1205 IFW=50THEN1225
1210 IFW=100THEN1225
1215 IFW=150THEN1225
1220 RETURN
1225 PRINT#-2,CHR$(10)CHR$(10)CH
R$(10)CHR$(10)CHR$(10)CHR$(10)CH
R$(10)CHR$(10)CHR$(10)CHR$(10)CH
R$(10)CHR$(10)CHR$(10)CHR$(10)CH
R$(10)CHR$(10):GOTO1345
1230 CLS:GOSUB590
1235 PRINT@105,"COST PRINTOUT"
1240 PRINT@193,"A COST PRINTOUT,
PRINTS THE ITEM NO.; DESCRI
PTION; RETAIL PRICE; COST PRIC
E; TRADE DIS%; W/SALE DIS%; TAX
.... IN THAT ORDER OF ALL ITE
MS IN MEMORY"
1245 PRINT@449,"AVAILABLE COMMAND
S":PRINT@481,"<P>FOR PRINTOUT
<3>FOR MENU";
1250 EXEC 44539:IS=INKEY$:IFIS=""
THEN1250
1255 IFIS="3"THEN1025
1260 IFIS="P"THENPT=PEEK(&HFF22)
AND 1:GOTO1270
1265 GOTO1250
1270 IFPT=1THENPRINT@386,"PRINTE
R OFFLINE":GOTO1250
1275 PRINT@386,"":PRINT@480,"
";

1280 GOSUB1180
1285 PRINT#-2,ST$(1,W);TAB(13);S
T$(2,W);TAB(45);ST$(3,W);TAB(53)
;ST$(4,W);TAB(63);ST$(5,W);TAB(7
1);ST$(6,W);TAB(76);ST$(7,W)
1290 GOSUB1190:GOTO1285
1295 CLS:GOSUB590
1300 PRINT@104,"ON HAND PRINTOUT
"
1305 PRINT@193,"A ON HAND PRINTO
UT, PRINTS THE ITEM NO.; DESCRI
PTION; RETAIL PRICE; COST PRIC
E; QTY ON HAND IN THAT ORDER OF
ALL ITEMS IN MEMORY."
1310 PRINT@449,"AVAILABLE COMMAND
S":PRINT@481,"<P>FOR PRINTOUT
<3>FOR MENU";
1315 EXEC44539:IS=INKEY$:IFIS=""
THEN1315
1320 IFIS="3"THEN1025
1325 IFIS="P"THENPT=PEEK(&HFF22)
AND 1:GOTO1335
1330 GOTO1315
1335 IFPT=1THENPRINT@386,"PRINTE
R OFFLINE":GOTO1315
1340 PRINT@386,"":PRINT@480,"
";

1345 PRINT#-2,"ITEM CODE DES
CRPTION";TAB(44);"RETAIL COS
T ON HAND":PRINT#-2,CHR$(28)C
HR$(80)CHR$(45):PRINT#-2,"DATE..
."DT$:PRINT#-2,""
1350 PRINT#-2,ST$(1,W);TAB(13);S
T$(2,W);TAB(45);ST$(3,W);TAB(53)
;ST$(4,W);TAB(65);ST$(9,W)
1355 W=W+1:IFW=200THEN1360ELSE13
65
1360 PRINT#-2,CHR$(28)CHR$(31) "
*";" END OF PRINTOUT ";CHR$(28)C
HR$(32) "*" :GOTO1025
1365 IFST$(1,W)=""THEN1355ELSE13
70
1370 IFW=50THEN1390
1375 IFW=100THEN1390
1380 IFW=150THEN1390
1385 GOTO1350
1390 PRINT#-2,CHR$(10)CHR$(10)CH
R$(10)CHR$(10)CHR$(10)CHR$(10)CH
R$(10)CHR$(10)CHR$(10)CHR$(10)CH
R$(10)CHR$(10):GOTO1345
1395 CLS:GOSUB590:PRINT@104,"SUP
PLIER PRINTOUT"
1400 PRINT@161,"A SUPPLIER PRINT
OUT, PRINTS THE ITEM NO.; DESCRI
PTION; RETAIL; COST; TRADE DIS%
; W/SALE DIS%; TAX;IN THAT ORDE
R OF ALL ITEMS FROM THE SELECTE
D SUPPLIER"
1405 PRINT@449,"AVAILABLE COMMAND
S"
1410 PRINT@481,"<ENTER> FOR MENU
";
1415 PRINT@355,"ENTER SUPPLIER N
AME TO BE PRINTED :";:LIN
EINPUTZ$
1420 IFZ$=""THEN1025
1425 PT=PEEK(&HFF22)AND 1:IFPT=1
THENPRINT@471,"printer":PRINT@50
4,"offline";:GOTO1415
1430 PRINT@471,"
";

1435 P=0:PRINT#-2,"ITEM CODE
DESCRIPTION";TAB(44);"RETAIL C
OST TRADE% W/SALE% TAX%":PRINT
#-2,CHR$(28)CHR$(80)CHR$(45):PR
INT#-2,"DATE... ";DT$:PRINT#-2,CH
R$(10):PRINT#-2," SUPPLIER...
"Z$:PRINT#-2,CHR$(10)
1440 IFZ$=ST$(8,W)THEN1450
1445 W=W+1:IFW=200THEN1195ELSEGO
TO1440
1450 P=P+1:PRINT#-2,ST$(1,W);TAB
(13);ST$(2,W);TAB(45);ST$(3,W);T
AB(53);ST$(4,W);TAB(63);ST$(5,W)
;TAB(71);ST$(6,W);TAB(76);ST$(7,
W)
1455 IF W=200THEN1195
1460 IF P=50 THEN1480
1465 IF P=100 THEN1480
1470 IF P=150 THEN1480
1475 GOTO1445
1480 PRINT#-2,CHR$(10)CHR$(10)CH
R$(10)CHR$(10)CHR$(10)CHR$(10)CH
R$(10)CHR$(10)CHR$(10)CHR$(10)CH
R$(10):GOTO1435
1485 CLS:GOSUB590:PRINT@103,"ON
HAND BY SUPPLIER"
1490 PRINT@161,"A ON HAND BY SUP
PLIER PRINTOUT, PRINTS THE ITEM
NO. DESCRIPTION RETAIL; COST; ON
HAND(STOCK) IN THAT ORDER OF AL
L ITEMS IN MEM- ORY OF THE SELEC
TED SUPPLIER
1495 PRINT@449,"AVAILABLE COMMAND
S":PRINT@481,"<ENTER> FOR MENU";
1500 PRINT@355,"ENTER SUPPLIER N
AME TO BE PRINTED :";:LIN
EINPUTZ$
1505 IF Z$=""THEN1025
1510 PT=PEEK(&HFF22)AND 1:IFPT=1
THENPRINT@471,"printer":PRINT@50
4,"offline";:GOTO1500
1515 PRINT@471,"
";

1520 P=0:PRINT#-2,"ITEM CODE
DESCRIPTION";TAB(44);"RETAIL
COST ON HAND":PRINT#-2,CHR$(2

```

continued on page 19



MISSION DESTRUCTION

GRAPHIC ADVENTURE

by Neil Evans

THIS GAME REQUIRES a 128k CoCo and a cassette player. I am only a beginner at programming.

Actually, this was my very first attempt, so, from one beginner to all those others, you can do it too.

This program was the result of sitting and reading the manual over and over, trying out ideas, then putting those ideas together. So come on get those imaginations going and present your efforts too.

Credit for the music must go to Michael Repasy!, I obtained it from his program "Run For Your Life".

All instructions are presented on the screen. There are three "windows"; one for the commands that can be made, the second one for the situation you are in and the main one is the graphic representation of where you are.

All objects you find are automatically in your possession so don't worry about trying to "get" them. All possible directions are displayed on the situation screen.

There are also three other commands; "jump", "up" and "climb".

"Climb" means climb down. These commands can be used at any time, something may happen or it may not, depending on where you are.

If you press a wrong key or try to go in the wrong direction the screens will disappear then return to the same place.

The Listing:

```

0 GOTO10
1 '***** MISSION: DESTRUCTION *
  **** NIEL EVANS *****
3 SAVE"225:3":END'7
4 GRAPHIC ADVENTURE
5 BY NEIL EVANS
10 WIDTH 40
20 PALETTE RGB
30 CLS2
40 POKE 65497,0
50 LOCATE 9,5:PRINT"MISSION DES

```

```

TRUCTION"
60 LOCATE 9,7:PRINT"A GRAPHIC AD
VENTURE"
70 LOCATE 12,9:PRINT"By Neil Eva
ns"
80 LOCATE 12,11:PRINT"For The Co
co3"
90 GOTO 110
100 POKE65496,0:STOP
110 ON BRK GOTO100
120 LOCATE 9,13:PRINT" <any key
to start>"
130 IF INKEY$=""THEN 130
140 CLS1
160 LOCATE 4,2:ATTR1,4:PRINT"You
are the commander of a ";
170 LOCATE4,3:PRINT"starship sen
t to an alien ";
180 LOCATE 4,4:PRINT"planet to d
estroy a weapon that ";
190 LOCATE4,5:PRINT"has been sho
oting down shuttle ";
200 LOCATE4,6:PRINT"crafts from
Earth that pass ";
210 LOCATE4,7:PRINT"close to the
planet ";
220 LOCATE 4,8:PRINT"Unfortunate
ly your ship was ";
230 LOCATE 4,9:PRINT"fired upon
and crash landed ";
240 LOCATE 4,10:PRINT"on the pla
net, all your supplies";
250 LOCATE 4,11:PRINT"and weapon
s were destroyed ";
260 LOCATE 4,12:PRINT"Can You Co
mplete Your Mission ";
270 LOCATE 4,13:PRINT"And Destro
y The Weapon ";
280 LOCATE4,18:PRINT" G
OOD LUCK! ";
290 LOCATE4,19:PRINT" <any k
ey to begin> ";
300 IF INKEY$=""THEN300
310 GOTO330
320 HDRAW"C4;BM104,4;R110;D92;L1
10;U92":HPAINT(160,45),8,4:RETUR
N
330 WIDTH40
340 PALETTE RGB
350 G=0:K=0:R=0:L=0
360 HSCREEN 2
370 HCLS 5
380 HDRAW"C6;BM100,0;R118;D100;L
118;U100"
390 HDRAW"C6;BM102,2;R114;D96;L1
14;U96"
400 HDRAW"C1;BM99,0;R120;D101;L1
20;U101"
410 HDRAW"C1;BM15,117;R135;D53;L

```

```

135;U53"
420 HDRAW"C1;BM170,117;R135;D53;
L135;U53"
430 HPAINT(12,19),2,1
440 HPAINT(190,169),6,1
450 HPAINT(20,169),6,1
460 HPRINT(6,15),"SITUATION"
470 HPRINT(26,15),"COMMANDS"
480 HPRINT(23,16),"East"
490 HPRINT(23,18),"Jump"
500 HPRINT(29,18),"Climb"
510 HPRINT(29,16),"West"
520 HPRINT(23,17),"North"
530 HPRINT(29,17),"South"
540 HPRINT(23,19),"Up"
550 A$="C8;BM214,28;L110;D22;R11
0;D1;L110;U1;R110;U22;BL10;G20;B
L10;E20;BL10;G20;BL10;E20;BL10;G
20;BL10;E20;BL10;G20;BL10;E20;BL
10;G20;BL10":GOTO600
560 HDRAW A$
570 HPAINT(208,49),15,8
580 HPAINT(109,30),15,8
590 RETURN
600 B$="C8;BM183,98;U1;R1;U1;R1;
U1;R1;U1;R1;U1;R5;U6;R1;U1;R1;E3
;U1;R1;U1;E3;U2;R1;U2;R1;U2;R1;U
1;R2;U1;R3;D2;R2;D2;E2;D1;D21;L1
10;U92;R110;D71":GOTO640
610 HDRAW B$
620 HPAINT(212,90),15,8
630 RETURN
640 C$="C8;BM108,82;R2;D1;R2;D1;
R2;D1;R2;R38;U1;L1;U1;H5;L1;H2;L
1;U1;L1;U3;H3;L1;U1;L1;U1;L2;U1;
L3;U1;L4;U2;L1;U1;L1;U1;L1;U1;L3
;D1;L1;D1;L1;D1;L1;D1;L1;G3;L1;D3;L
1;D3;L2;D3;L2;D2;L1;D1;L1;D3;R2;
D1;R2;D1;R2;U2;R1;U2;R2;U2;R1;U2
;R1;U3;R2;U2;R2;U3;R3;U3"
650 GOTO 720
660 HDRAW C$
670 HPAINT (120,83),15,8
680 PALETTE 14,49
690 HDRAW"C8;BM217,51;L115"
700 HPAINT(160,76),14,8
710 RETURN
720 D$="C8;BM162,77;R13;E13;H13;
G4;D14;BU14;BE4;BF13;BG13;BL13;E
22;BD14;BL8;R4;E7;BH3;G10;BE10;B
H3;E2;U2;F3;L2;BF5;G2;BE2;U2;F3;
L2;G2":GOTO850
730 HCIRCLE(200,12),8,8
740 HPAINT (200,8),3,8
750 RETURN
760 HDRAW D$
770 HCIRCLE (164,75),4,8,1,.375,
.875
780 HPAINT(171,76),0,8

```

```

790 HPAINT(176,60),0,8
800 HPAINT(180,68),0,8
810 HPAINT(181,61),0,8
820 HPAINT(184,64),0,8
830 HPAINT(164,74),11,8
840 RETURN
850 E$="C8;BM156,51;U12;R12;D12"
:GOTO900
860 HDRAW E$
870 HPAINT(160,47),8,8
880 HPAINT(159,42),8,8
890 RETURN
900 F$="C8;BM150,60;D1;L1;D1;L1;
D2;L1;D1;L2;D2;L2;R54;L2;U2;L2;U
1;L1;U2;L1;U1;L1;U1":GOTO960
910 HDRAW F$
920 HCIRCLE(170,60),20,8,.25
930 HPAINT(155,66),15,8
940 HPAINT(170,60),8,8
950 RETURN
960 G$="C8;BM166,52;U12;R26;D12;
BL11;U5;L4;D5":GOTO1550
970 HDRAW G$
980 HCIRCLE(179,40),13,8,1,.5,0
990 HCIRCLE(184,44),3,8
1000 HCIRCLE(173,44),3,8
1010 HPAINT(178,34),4,8
1020 HPAINT(184,48),4,8
1030 HPAINT(178,50),2,8
1040 HPAINT(173,44),5,8
1050 HPAINT(183,44),5,8
1060 RETURN
1070 PALETTE 14,49
1080 HDRAW"C8;BM104,4;R110;D92;L
110;U92"
1090 HLINE(102,21)-(160,3),PSET
1100 HLINE(160,3)-(216,21),PSET
1110 HLINE(160,3)-(160,37),PSET
1120 HLINE(160,37)-(102,55),PSET
1130 HLINE(160,37)-(191,47),PSET
1140 HLINE(191,53)-(191,32),PSET
1150 HLINE(191,32)-(217,40),PSET
1160 HLINE(191,32)-(204,28),PSET
1170 HLINE(204,28)-(217,33),PSET
1180 HLINE(201,35)-(201,56),PSET
1190 HLINE(203,36)-(203,57),PSET
1200 HLINE(212,39)-(212,60),PSET
1210 HLINE(214,40)-(212,61),PSET
1220 HLINE(112,29)-(136,21),PSET
1230 HLINE(136,21)-(144,23),PSET
1240 HLINE(144,23)-(144,24),PSET
1250 HLINE(144,24)-(119,33),PSET
1260 HLINE(119,33)-(119,32),PSET
1270 HLINE(119,32)-(112,29),PSET
1280 HLINE(112,29)-(112,30),PSET
1290 HLINE(112,30)-(119,33),PSET
1300 HLINE(122,32)-(118,37),PSET
1310 HLINE(118,37)-(118,37),PSET
1320 HLINE(118,39)-(125,33),PSET
1330 HLINE(135,28)-(132,32),PSET
1340 HLINE(132,32)-(132,34),PSET
1350 HLINE(132,34)-(137,27),PSET
1360 HLINE(192,53)-(217,61),PSET
1370 HLINE(103,3)-(215,97),PSET,
B
1380 HPAINT(200,5),0,8
1390 HPAINT(200,20),0,8
1400 HPAINT(148,20),0,8
1410 HPAINT(160,68),14,8
1420 HPAINT(132,25),15,8
1430 HPAINT(136,27),12,8
1440 HPAINT(115,31),12,8
1450 HPAINT(120,36),15,8.

1460 HPAINT(136,28),15,8
1470 HPAINT(208,34),15,8
1480 HPAINT(201,40),12,8
1490 HPAINT(214,40),12,8
1500 HPAINT(196,40),15,8
1510 HPAINT(208,40),15,8
1520 HPAINT(215,44),15,8
1530 HPAINT(112,6),0,8
1540 RETURN
1550 H$="C8;BM144,55;R28;E10;L28
;G10;D14;R28;U14;BD14;E10;U14":G
OTO1610
1560 HDRAW H$
1570 HPAINT(160,48),3,8
1580 HPAINT(176,60),3,8
1590 HPAINT(160,60),3,8
1600 RETURN
1610 I$="C8;BM130,16;R2;D8;F3;L8
;E3;R2;BL2;U8":GOTO1660
1620 HDRAW I$
1630 HPAINT(131,26),8,8
1640 HPAINT(131,20),3,8
1650 RETURN
1660 J$="C8;BM108,81;R25;E18;L25
;D17;BU17;G18;BE18;E19;R25;G19":
GOTO1770
1670 HDRAW J$
1680 HPAINT(116,76),4,8
1690 HPAINT(132,72),8,8
1700 HPAINT(136,56),15,8
1710 HLINE(115,79)-(120,74),PSET
1720 HLINE(115,79)-(115,81),PSET
1730 HLINE(115,81)-(120,76),PSET
1740 HLINE(120,74)-(120,81),PSET
1750 HLINE(120,79)-(118,81),PSET
1760 RETURN
1770 K$="C8;BM137,20;R45;D51;L45
;U51;H17;BF17;BR45;E17;BG17;BD51
;F27;BH27;BL45;G27;BE27;BD27;L35
;U96;R114;D96;L80":GOTO1890
1780 HDRAW K$
1790 PALETTE14,49
1800 HPAINT(160,7),14,8
1810 HPAINT(160,96),14,8
1820 HPAINT(116,48),8,8
1830 HPAINT(208,96),8,8
1840 HPAINT(160,48),1,8
1850 RETURN
1860 HCIRCLE(160,84),20,8,.25
1870 HPAINT(160,84),8,8
1880 RETURN
1890 L$="C8;BM131,85;R4;BL4;G3;D
4;F3;R4;E3;R3;D2;R2;U4;R2;D2;R1;
D3;R1;U3;R2;D1;R2;U5;L13;H3;BD4;
L3;D3;R3;U3":GOTO1930
1900 HDRAW L$
1910 HPAINT(133,86),5,8
1920 RETURN
1930 M$="C8;BM173,84;D9;R4;U5;R3
;D2;L3;BR3;BU2;R7;U4;L14;BR14;BD
1;R7;D2;L7;BR2;BD3;U8;BD3;BR2;BU
2;D6;BU2;BR1;BD1;U4":GOTO1970
1940 HDRAW M$
1950 HPAINT(176,86),5,8
1960 RETURN
1970 N$="C8;BM155,25;R14;D11;BR7
;L21;U11;BD11;L9;D18;R4;U12;R2;D
17;R2;D26;R5;U5;L5;BR5;U21;R5;D2
6;R5;U5;L5;BR5;U21;R2;U17;R2;D12
;R4;U18;BL11;BU6;L2;U3;R2;D3;BL6
;U3;L2;D3;R2;BD2;BL1R6;D2;L6;U2"
:GOTO2090
1980 HDRAW N$

1990 HPAINT(156,26),4,8
2000 HPAINT(160,48),0,8
2010 HPAINT(156,82),4,8
2020 HPAINT(166,82),4,8
2030 HPAINT(158,28),3,8
2040 HPAINT(164,28),3,8
2050 HPAINT(160,33),3,8
2060 HPAINT(156,72),8,8
2070 HPAINT(166,72),8,8
2080 RETURN
2090 O$="C1;BM146,88;R34;U15;L34
;D15;BU15;BR4;U6;R26;D6;BU6;BL5;
U5;L16;D5;BU5;BR5;U45;R6;D45;BU5
;R21;BL27;L21;BR21;BU8;L15;BR21;
R15;BL15;BU9;R9;BL15;L9;BR9;BU8;
L5;BR11;R5;BL5;BU8;R3;BL9;L3":GO
TO2550
2100 HDRAW O$
2110 HDRAW"C1;BM104,4;R110;D92;L
110;U92"
2120 HPAINT(164,40),4,1
2130 HPAINT(164,64),0,1
2140 HPAINT(164,69),0,1
2150 HPAINT(164,80),0,1
2160 HPAINT(124,16),8,1
2170 RETURN
2180 GOTO 2190
2190 GOTO2200
2200 HPAINT(20,169),0,2
2210 HPAINT(20,169),6,2
2220 HPRINT(6,15),"SITUATION"
2230 HPAINT(160,40),5,6:RETURN
2240 'ROCKET AT CLIFF
2250 GOSUB560:GOSUB610:GOSUB660:
GOSUB730:GOSUB760:RETURN
2260 'CRATER
2270 GOSUB610:GOSUB660:GOSUB730:
GOSUB910:RETURN
2280 'TOP NOTHING
2290 GOSUB 610:GOSUB660:GOSUB730
:RETURN
2300 'OUTSIDE HUT
2310 GOSUB610:GOSUB660:GOSUB730:
GOSUB970:RETURN
2320 'CLIFF ONLY
2330 GOSUB560:GOSUB610:GOSUB660:
GOSUB730:RETURN
2340 'CLIFF WITH CAVE
2350 GOSUB560:GOSUB610:GOSUB660:
GOSUB730:GOSUB860:RETURN
2360 'INSIDE HUT1 AFTER LOOK
2370 GOSUB1070:GOSUB1620:GOSUB16
70:RETURN
2380 'INSIDE HUT2 AFTER LOOK
2390 GOSUB1070:GOSUB1560:RETURN
2400 'IN TUNNEL SEE NOTHING
2410 GOSUB1780:RETURN
2420 'PIT IN TUNNEL
2430 GOSUB1780:GOSUB1860:RETURN
2440 'ROBOT BELOW
2450 GOSUB1780:GOSUB1980:RETURN
2460 'GUN BELOW
2470 GOSUB1780:GOSUB1940:RETURN
2480 'KEY BELOW
2490 GOSUB 1780:GOSUB1900:RETURN
2500 GOSUB1780:GOSUB1560:RETURN
2510 'WEAPON
2520 GOSUB2100:RETURN
2530 Z$=INKEY$:IF Z$=""THEN2530E
LSERRETURN
2540 GOSUB2190
2550 GOSUB2190
2560 GOSUB2240:HPRINT(3,16),"YOU

```

```

ARE OUTSIDE":HPRINT(3,17),"YOUR
SHIP":HPRINT(3,18),"<exit E onl
y>"
2570 GOSUB2530:IF Z$<"E" THEN GOS
UB2190 ELSE 2590
2580 GOSUB2190:GOTO2560
2590 GOSUB2190
2600 GOSUB2260:HPRINT(3,16),"I S
EE A":HPRINT(3,17),"CRATER":HPR
INT(3,18),"<exits V,E,S >"
2610 GOSUB2530
2620 IF Z$="J" THEN 4710
2630 IF Z$="W" THEN GOTO2540
2640 IF Z$="S" THEN GOTO2720
2650 IF Z$="E" THEN GOTO2810
2660 IF Z$="C" THEN GOTO2680
2670 GOSUB2190:GOTO2600
2680 IF R=0 THEN GOTO2710 ELSE 2690
2690 GOSUB2190:GOSUB2100
2700 HPRINT(3,16),"CONGRATULATIO
NS":HPRINT(3,17),"YOU FOUND IT":
HPRINT(3,18),"YOU WIN":HPRINT(3,
19),"game over":GOTO2700
2710 GOSUB2190:HPRINT(3,16),"YOU
DON'T HAVE":HPRINT(3,17),"ANY R
OPE":GOSUB4640:GOTO2590
2720 GOSUB2190
2730 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits N,S,E>"
2740 GOSUB2530:IF Z$="N" THEN GOTO2770
2750 IF Z$="S" THEN GOTO 3170
2760 IF Z$="E" THEN GOTO 2790
2770 IF Z$="W" THEN GOTO2590
2780 IF Z$="E" THEN GOTO2880
2790 IF Z$="S" THEN GOTO3170
2800 GOSUB2190:GOTO2720
2810 GOSUB2190
2820 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits V,S>"
2830 GOSUB2530:IF Z$="W" THEN GOTO2850
2840 IF Z$="S" THEN GOTO2860
2850 IF Z$="W" THEN GOTO2590
2860 IF Z$="S" THEN GOTO2880
2870 GOSUB2190:GOTO2810
2880 GOSUB2190
2890 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits N,S,E,W>"
2900 GOSUB2530:IF Z$="N" THEN GOTO2810
2910 IF Z$="S" THEN GOTO3050
2920 IF Z$="E" THEN GOTO2950
2930 IF Z$="W" THEN GOTO2720
2940 GOSUB2190:GOTO2880
2950 GOSUB2190
2960 GOSUB2300:L=0:HPRINT(3,16),
"I SEE A HUT":HPRINT(3,17),"Nort

```

```

h TO ENTER":HPRINT(3,18),"<exit
V only>"
2970 GOSUB2530:IF Z$="N" THEN GOTO3000
2980 IF Z$="W" THEN GOTO2880
2990 GOSUB2190:GOTO2950
3000 GOSUB2190
3010 GOSUB2360:L=1:HPRINT(3,16),
"I SEE A LADDER":HPRINT(3,17),"A
ND CANDLE":HPRINT(3,18),"<exit S
>"
3020 GOSUB2530:IF Z$="C" THEN GOTO3870
3030 IF Z$="S" THEN GOTO2950
3040 GOSUB2190:GOTO3000
3050 GOSUB2190
3060 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits N,W>"
3070 GOSUB2530:IF Z$="N" THEN GOTO2880
3080 IF Z$="W" THEN GOTO3100
3090 GOSUB2190:GOTO3050
3100 GOSUB2190
3110 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exit N,S,E,W>"
3120 GOSUB2530:IF Z$="N" THEN GOTO3170
3130 IF Z$="S" THEN GOTO3550
3140 IF Z$="E" THEN GOTO3050
3150 IF Z$="W" THEN GOTO3400
3160 GOSUB2190:GOTO3100
3170 GOSUB2190
3180 GOSUB2300:R=1:HPRINT(3,16),
"YOU ARE OUTSIDE":HPRINT(3,17),"
A HUT":HPRINT(3,18),"East to ent
er":HPRINT(3,19),"<exits N,S,W>"
3190 GOSUB2530:IF Z$="E" THEN GOTO3240
3200 IF Z$="N" THEN GOTO2720
3210 IF Z$="S" THEN GOTO3100
3220 IF Z$="W" THEN GOTO3300
3230 GOSUB2190:GOTO3170
3240 GOSUB2190
3250 IF K=0 THEN GOTO3260 ELSE GOTO
3270
3260 HPRINT(3,16),"YOU DON'T HAV
E":HPRINT(3,17),"THE KEY":GOSUB4
640:GOTO3230
3270 GOSUB2380:R=1:HPRINT(3,16),
"I SEE A BOX ":HPRINT(3,17),"IT
CONTAINS ROPE":HPRINT(3,18),"DYN
AMITE":HPRINT(3,19),"CLUE=VNW,<e
xit W>"
3280 GOSUB2530:IF Z$="W" THEN GOTO3170
3290 GOSUB2190:GOTO3240
3300 GOSUB2190
3310 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits E,W,S>"
3320 GOSUB2530:IF Z$="W" THEN GOTO3360
3330 IF Z$="E" THEN GOTO3170
3340 IF Z$="S" THEN GOTO3400
3350 GOSUB2190:GOTO3300
3360 GOSUB2190
3370 GOSUB2320:HPRINT(3,16),"I S
EE A CLIFF":HPRINT(3,18),"<exit
E only>"
3380 GOSUB2530:IF Z$="E" THEN GOTO3300
3390 GOSUB2190:GOTO3360
3400 GOSUB2190
3410 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exit N,E,W>"
3420 GOSUB2530:IF Z$="N" THEN GOTO3300
3430 IF Z$="E" THEN GOTO3100
3440 IF Z$="W" THEN GOTO3460

```

```

3450 GOSUB2190:GOTO3400
3460 GOSUB2190
3470 GOSUB2320:HPRINT(3,16),"I S
EE A CLIFF":HPRINT(3,18),"<exit
E only>"
3480 GOSUB2530:IF Z$="E" THEN GOTO3400
3490 GOSUB2190:GOTO3460
3500 GOSUB2190
3510 GOSUB2340:L=0:HPRINT(3,16),
"I SEE A":HPRINT(3,17),"CLIFF VI
TH A":HPRINT(3,18),"CAVE West
TO ENT":HPRINT(3,19),"<exit E on
ly>"
3520 GOSUB2530:IF Z$="E" THEN GOTO3
550
3530 IF Z$="W" THEN GOTO3700
3540 GOSUB2190:GOTO3500
3550 GOSUB2190
3560 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exit N,E,W>"
3570 GOSUB2530:IF Z$="N" THEN GOTO3100
3580 IF Z$="E" THEN GOTO3610
3590 IF Z$="W" THEN GOTO3500
3600 GOSUB2190:GOTO3550
3610 GOSUB2190
3620 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits V,S>"
3630 GOSUB2530:IF Z$="W" THEN GOTO3550
3640 IF Z$="S" THEN GOTO3660
3650 GOSUB2190:GOTO3610
3660 GOSUB2190
3670 GOSUB2280:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exit N only>"
3680 GOSUB2530:IF Z$="N" THEN GOTO3610
3690 GOSUB2190:GOTO3660
3700 GOSUB2190:IF L=1 THEN GOTO3710 ELSE
GOSUB320:GOTO3780
3710 GOSUB2400
3720 HPRINT(3,16),"YOU'RE INSIDE
":HPRINT(3,17),"A TUNNEL":HPRINT
(3,18),"<exit V,S,E>":HPRINT(3,1
9),"Vest to leave"
3730 GOSUB2530
3740 IF Z$="E" THEN GOTO3800
3750 IF Z$="S" THEN GOTO3930
3760 IF Z$="W" THEN GOTO3500
3770 GOSUB2190:GOTO3700
3780 HPRINT(3,16),"TOO BAD YOU":
HPRINT(3,17),"CAN'T SEE":GOSUB46
40
3790 GOSUB2190:GOTO3720
3800 GOSUB2190:IF L=1 THEN GOTO3810 ELSE
GOSUB320:GOTO3860
3810 GOSUB2420
3820 HPRINT(3,16),"I SEE A PIT":
HPRINT(3,17),"YOU CAN STILL":HPR
INT(3,18),"PASS":HPRINT(3,19),"<
exits E,W>"
3830 GOSUB2530:IF Z$="W" THEN GOTO3700
3840 IF Z$="E" THEN GOTO3870
3850 GOSUB2190:GOTO3800
3860 GOTO4690
3870 GOSUB2190
3880 GOSUB2400:HPRINT(3,16),"YOU
R AT THE":HPRINT(3,17),"BOTTOM O
F A":HPRINT(3,18),"LADDER":HPRIN
T(3,19),"<exit E,W,S,U>"
3890 GOSUB2530:IF Z$="E" THEN GOTO4210
3900 IF Z$="W" THEN GOTO3800
3910 IF Z$="S" THEN GOTO4110
3911 IF Z$="U" THEN GOTO3000

```

Hint ...

Turn off your drive motors

Here is a short POKE to turn off your disk drive motors the moment you press enter.

POKE 65344,0

3920 GOSUB2190:GOTO3870
3930 GOSUB2190
3940 IFL=1GOTO3950ELSEGOSUB320:G
OTO4000
3950 GOSUB2400
3960 HPRINT(3,16),"I SEE NOTHING
":HPRINT(3,17),"SPECIAL":HPRINT(3
,18),"<exits N,W>"
3970 GOSUB2530:IFZ\$="N"GOTO3700
3980 IFZ\$="W"GOTO4020
3990 GOSUB2190:GOTO3930
4000 HPRINT(3,16),"TOO DARK":GOS
UB4640
4010 GOSUB2190:GOTO3960
4020 GOSUB2190
4030 IF L=1GOTO4040ELSEGOSUB320:
GOTO4090
4040 GOSUB2500
4050 HPRINT(3,16),"I SEE A BOX":
HPRINT(3,17),"West to get box":H
PRINT(3,18),"<exits V,E>"
4060 GOSUB2530:IFZ\$="W"GOTO4720
4070 IFZ\$="E"GOTO3930
4080 GOSUB2190:GOTO4020
4090 HPRINT(3,16),"TOO DARK":GOS
UB4640
4100 GOSUB2190:GOTO4050
4110 GOSUB2190
4120 GOSUB2400:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits N,E>"
4130 GOSUB2530:IFZ\$="N"GOTO3870
4140 IFZ\$="E"GOTO4160
4150 GOSUB2190:GOTO4110
4160 GOSUB2190
4170 GOSUB2400:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits V,S>"
4180 GOSUB2530:IFZ\$="W"GOTO4110
4190 IFZ\$="S"GOTO4500
4200 GOSUB2190:GOTO4160
4210 GOSUB2190
4220 GOSUB2400:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits N,S,E,W>"
"
4230 GOSUB2530:IFZ\$="N"GOTO4360
4240 IFZ\$="S"GOTO4320
4250 IFZ\$="E"GOTO4280
4260 IFZ\$="W"GOTO3870
4270 GOSUB2190:GOTO4210
4280 GOSUB2190
4290 GOSUB2400:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits V only>"
4300 GOSUB2530:IFZ\$="W"GOTO4210
4310 GOSUB2190:GOTO4280
4320 GOSUB2190
4330 GOSUB2400:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exit N only>"
4340 GOSUB2530:IFZ\$="N"GOTO4210
4350 GOSUB2190:GOTO4320
4360 GOSUB2190
4370 GOSUB2400:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA
L":HPRINT(3,18),"<exits S,W>"
4380 GOSUB2530:IFZ\$="S"GOTO4210
4390 IFZ\$="W"GOTO4410
4400 GOSUB2190:GOTO4360
4410 GOSUB2190
4420 GOSUB2400:HPRINT(3,16),"I S
EE NOTHING":HPRINT(3,17),"SPECIA

L":HPRINT(3,18),"<exits S,E>"
4430 GOSUB2530:IFZ\$="S"GOTO4460
4440 IFZ\$="E"GOTO4360
4450 GOSUB2190:GOTO4410
4460 GOSUB2190
4470 GOSUB2460:G=1:HPRINT(3,16),
"YOU HAVE FOUND":HPRINT(3,17),"A
GUN":HPRINT(3,18),"<exit N only
>"
4480 GOSUB2530:IFZ\$="N"GOTO4410
4490 GOSUB2190:GOTO4460
4500 GOSUB2190
4510 GOSUB2440:IF G=0 GOTO4560
4520 HPRINT(3,16),"I SEE A ROBOT
":HPRINT(3,17),"LUCKY YOU FOUND"
:HPRINT(3,18),"THE GUN":HPRINT(3
,19),"<exits N,S>"
4530 GOSUB2530:IFZ\$="N"GOTO4160
4540 IFZ\$="S"GOTO4600
4550 GOSUB2190:GOTO4500
4560 HPRINT(3,16),"I SEE A ROBOT
":HPRINT(3,17),"TAKE A CHANCE":H
PRINT(3,18),"OR TURN BACK":HPRIN
T(3,19),"exit Chance,Back"
4570 GOSUB2530:IFZ\$="B"THEN4160
4580 IFZ\$="C"GOTO4740
4590 GOSUB2190:GOTO4500
4600 GOSUB2190
4610 GOSUB2480:K=1:HPRINT(3,16),
"YOU FOUND A":HPRINT(3,17),"KEY"
:HPRINT(3,18),"<exit N only>"
4620 GOSUB2530:IFZ\$="N"GOTO4500
4630 GOSUB2190:GOTO4600
4640 TIMER=0
4650 FOR=1TO400
4660 TIMER=TIMER+1
4670 NEXT T
4680 IF TIMER>400 THEN RETURN
4690 HCL58:HPRINT(13,10),"BAD LU
CK":HPRINT(13,11),"IN THE DARK":
HPRINT(13,12),"YOU DIDN'T SEE TH
E":HPRINT(13,13),"PIT, AND FELL
TO":HPRINT(13,14),"YOUR DEATH":H
PRINT(13,16),"game over":PLAY"T2
L401CP96CL3FL4P24CFAP64CFAP64CFA
FAO2CO1AFCP64CP96CL3F"
4700 POKE65496,0:STOP
4710 HCL55:HPRINT(13,10),"BAD LU
CK":HPRINT(13,11),"THE HOLE":HPR
INT(13,12),"WAS TO DEEP":HPRINT(
13,13),"YOU BROKE YOUR":HPRINT(1
3,14),"NECK game over":PLAY"T2L4
01CP96CL3FL4P24CFAP64CFAP64CFA
FAO2CO1AFCP64CP96CL3F":POKE65496,0
:STOP
4720 HCL54:HPRINT(13,10),"BAD LU
CK":HPRINT(13,11),"THE BOX WAS F
ULL":HPRINT(13,12),"OF DYNAMITE
WHICH":HPRINT(13,13),"BLEW UP KI
LLING YOU":HPRINT(13,14),"game o
ver"
4730 PLAY"T2L401CP96CL3FL4P24CFA
P64CFAP64CFAP64CFAP64CFAP64CFA
FAO2CO1AFCP64CP96CL3F":POKE65496,0:STOP
4740 HCL51:HPRINT(13,10),"BAD LU
CK":HPRINT(13,11),"HE ZAPPED YOU
":HPRINT(13,12),"DON'T YOU WISH"
:HPRINT(13,13),"YOU HAD A GUN":H
PRINT(13,14),"game over"
4750 PLAY"T2L401CP96CL3FL4P24CFA
P64CFAP64CFAP64CFAP64CFAP64CFAP
64CFAP64CFAP64CFAP64CFAP64CFAP64
F":POKE65496,0:STOP

continued from page 15

8)CHR\$(80)CHR\$(45):PRINT#-2,"DAT
E... "DT\$:PRINT#-2,"":PRINT#-2,"
SUPPLIER... "Z\$:PRINT#-2,CHR\$(
10)
1525 IF Z\$=ST\$(8,W)THEN1535
1530 W=W+1:IFW=200THEN1195ELSEGO
TO1440
1535 P=P+1:PRINT#-2,ST\$(1,W);TAB
(13);ST\$(2,W);TAB(45);ST\$(3,W);T
AB(53);ST\$(4,W);TAB(65);ST\$(9,W)
1540 IFF=50 THEN1560
1545 IFF=100THEN1560
1550 IFF=150THEN1560
1555 GOTO 1530
1560 PRINT#,2,CHR\$(10)CHR\$(10)CH
R\$(10)CHR\$(10)CHR\$(10)CHR\$(10)CH
R\$(10)CHR\$(10)CHR\$(10)CHR\$(10)CH
R\$(10):GOTO1520
1565 CLS
1570 GOSUB590
1575 PRINT@105,"LOAD INVENTORY"
1580 PRINT@196,"1) POSITION TA
PE"
1585 PRINT@228,"2) PRESS PLAY"
1590 PRINT@260,"3) PRESS <L> T
O LOAD"
1595 PRINT@449,"AVAILIBLE COMMAND
S"
1600 PRINT@480,"<3>MAIN MENU. <L
>LOAD INVENTORY";
1605 EXEC44539:IFINKEY\$="L"THEN1
610ELSE130
1610 PRINT@480,"
";
1615 PRINT@358,"FILE INPUT PROC
EDING";
1620 OPEN"1", # -1,"STOCK N."
1625 FORW=1TO200:FORI=1TO9
1630 IF EOF(-1) THEN 1645
1635 INPUT # -1,ST\$(I,W)
1640 NEXT I,W
1645 CLOSE # -1:GOTO130
1650 CLS
1655 GOSUB590
1660 PRINT@105,"SAVE INVENTORY"
1665 PRINT@196,"1) POSITION TA
PE"
1670 PRINT@228,"2) PRESS PLAY
& RECORD"
1675 PRINT@260,"3) PRESS <S> T
O SAVE FILE";
1680 PRINT@449,"AVAILIBLE COMMAND
S"
1685 PRINT@480,"<3>MAIN MENU. <S
>SAVE INVENTORY";
1690 EXEC44539:IFINKEY\$="S"THEN1
695ELSE130
1695 PRINT@480,"
";
1700 PRINT@358,"FILE SAVE PROC
EDING";
1705 OPEN"0", # -1,"STOCK N."
1710 FORW=1TO200:FORI=1TO9
1715 PRINT # -1,ST\$(I,W)
1720 NEXT I,W
1725 CLOSE # -1
1730 PRINT@358,"FILE SAVE COMPLE
TE "
";
1735 PRINT@384,"DO YOU WANT A BA
CK UP TAPE. Y/N";
1740 EXEC44539:IFINKEY\$="Y"THEN1
650ELSE130



You'll go wild over...

ONO

GAME

by Charles Barlett

THIS IS BASED on the card game UNO (r). Variations in the rules will be presented here. The ONO deck has 108 cards, these are made up of:-

- number cards 1 - 9 in Green x2
- number cards 1 - 9 in Yellow x2
- number cards 1 - 9 in Blue x2
- number cards 1 - 9 in Red x2
- number 0 (zero), 1 of each of the above colours
- Skip cards, 4 of each of the above colours
- Draw two cards, 2 of each of the above colours
- Wild cards x4
- Draw four cards x4

The only difference between the UNO and ONO decks is that ONO does not have any reverse cards. However in two player UNO the reverse cards are used as skip cards and as ONO is a two player game, the reverse cards have been replaced with additional skip cards.

At the start of the game, the computer deals itself and you seven cards the remaining cards in the deck are placed face down and the top card of the deck is turned over, this is the face card.

You get to go first, the object of the game is to put all your cards down before your opponent can. The cards left in your opponent's hands are then added to your score.

Additional rounds of the game are played until one of the players reaches 500 - that player is then the winner.

To be able to lay a card, it must be EITHER the same colour OR the same number or type. IE: a red 9 is the face card, you can lay any red card or a 9 card of any colour.

The SKIP card causes your opponent to miss a turn, to lay a skip card the face card must be the same colour as the skip card or the face card must be a skip card, (that situation is possible, if your opponent has laid a skip card to get rid of the points, but has no matching

card to follow, so he picks up from the pack).

The Draw two forces your opponent to pick up to cards from the deck and miss a turn.

The wild card can be played at any time, it allows the player to change to any colour, (including the existing colour).

If the computer lays a Wild card, it will paint it the selected colour, so that you know what to play next. If you lay a Wild card ONO will ask you the colour, just press the key that corresponds to the first letter of the required colour and the card will be painted.

The Draw four card forces your opponent to draw four cards from the deck and to miss a turn. It is a powerful card and thus has restrictions on its use.

YOU MAY NOT LAY A DRAW FOUR CARD IF YOU HOLD ANY CARDS THAT ARE THE SAME COLOR AS THE FACE CARD. If you try, ONO will give you an illegal move message.

If you cannot lay a card, you must draw a card from the deck.

If you can lay a card, you may still draw a card from the deck should you wish to, instead of laying a card for reasons of strategy which you will get to know as you play the game.

Use the joystick to move the card selector around the screen, when it is over the selected card or over the deck, press the fire button to select the card.

If legal, the card will replace the face card. ONO shuffles your cards around on the screen sometimes, to maintain playing room.

ONO RESTRICTION

Either player may hold up to 38 cards at once, if any player is placed in the position of having to draw a 39th card or just draws a 39th card, then that player loses that round and the points on the cards he holds are added to his opponent's score.

In practise this should never happen, unless you are an idiot, in which case you get your reward.

ONO like UNO depends to some degree on the luck of the cards, if you get good cards and play them well you stand a good chance of winning.

However the game still requires more thought than the novice may realise. ONO has programmed your computer to take FULL advantage of all the rules and sneaky techniques known to the game, it has only one skill level... FLAT OUT!!!

If ONO beats you all the time, I suggest you buy a pack of UNO cards and practise with a human companion ... then come back and try me again !!!

The Listing:

```

0 GOTO10
1 '***** ONO *****
  ***** CHARLES BARTLETT *****
3 SAVE"168B:3":END
10 PALETTE RGB:CLBAR2000:PCLBAR1
:ON BRK GOTO 1520
20 ON ERR GOTO 890
30 POKE65497,0:DIM V$(26),BP(40,
1),D(108,1),C(14,3),PD(40,1),CD(
40,1):GOSUB770:PALETTE15,56:PALE
TTE13,29:PALETTE10,28:HSCREEN2
40 GOSUB1490:HCLS15:HCOLOR8:GOSU
B1500:HPRINT(5,18),"(c) 1/1/87
Charles Bartlett":FOR K3=1 TO 5:
FOR K4=1 TO 10
50 MS$="BK"+STR$(K3+90)+"",+STR$(
K4+100)+"S32C5"+V$(15)+"C6"+V$(
14)+"C7"+V$(15):HDRAW MS$:NEXT K
4,K3
60 B=1:FOR CY=5 TO 140 STEP 45:F
OR CX=10 TO 280 STEP 30:BP(B,0)=
CX:BP(B,1)=CY:B=B+1:NEXTCX,CY
70 SH=0:EH=0:VH=1:GOSUB440:SH=1:
EH=12:VH=2:GOSUB440:SH=13:EH=14:
VH=1:GOSUB440
80 GOSUB450:HCLS15:'SHUFFLE
90 FOR D=1 TO 108:IF D(D,0)=11 T
HEN D(D,0)=10:NEXT ELSE NEXT
100 Z=1:FOR Q=1 TO 13 STEP 2:PD(
Z,0)=D(Q,0):PD(Z,1)=D(Q,1):CD(Z,
0)=D(Q+1,0):CD(Z,1)=D(Q+1,1):Z=Z
+1:NEXTQ:TW=7:PC=1:CT=7
110 TC=1:FOR SP=1 TO TH:NC=PD(TC
,0):CC=PD(TC,1):GOSUB580:TC=TC+1
:NEXT
120 SP=39:NC=15:GOSUB580:SP=40:W
C=D(15,0):CC=D(15,1):DH=NC:DC=CC
:GOSUB580:Q=16

```

```

130 IF DN=13 THEN IC=DC:GOSUB380
140 IF DN=14 OR DN=12 OR DN=10 T
HEN SP=40:GOSUB250:MC=D(Q,0):CC=
D(Q,1):DN=MC:DC=CC:GOSUB580:Q=Q+
1:GOTO140
150 IF TN=0 OR CT=0 OR TN>38 OR
CT>38 THEN 1320 ELSE GOSUB500
160 GOSUB310:IF BUTTON(0)=0 THEN
160 ELSE SOUND100,1:SOUND100,1:
IF LC=14 THEN DC=PD(PC,1):LC=0
170 IF PD(PC,0)>12 THEN 210
180 IF PC=39 THEN GOSUB1290:GOTO
900
190 IF PD(PC,1)=DC THEN GOSUB390
:IF DN=10 THEN 150 ELSE IF DN=12
THEN GOSUB430:GOTO150 ELSE 900
200 IF PD(PC,0)=DN THEN GOSUB390
:IF DN=10 THEN 150 ELSE IF DN=12
THEN GOSUB430:GOTO150 ELSE 900
210 IF PD(PC,0)=13 THEN GOSUB390
:MS$="What is color?":GOSUB480:
GOSUB350:IF TN=0 THEN 230 ELSE 9
00
220 IF PD(PC,0)=14 THEN GOSUB330
:IF OK THEN GOSUB390:GOSUB430:GO
SUB430:LC=14:GOTO150 ELSE 240
230 IF TN=0 OR CT=0 OR TN>38 OR
CT>38 THEN 1320
240 MS$="ILLEGAL MOVE":GOSUB480:
SOUND10,3:FOR K=1 TO 1000:NEXT K
:GOTO150
250 HCOLOR15:HLINE(BP(SP,0),BP(S
P,1))-(BP(SP,0)+25,BP(SP,1)+40),
PSET,BF:RETURN
260 JO=JOYSTK(0):J1=JOYSTK(1):IF
JO<5 THEN IF PC=39 THEN PC=TN E
LSE PC=PC-1:IF PC<1 THEN PC=1
270 IF JO>60 THEN PC=PC+1:IF PC>
TN THEN PC=39
280 IF J1<5 THEN IF PC=39 THEN P
C=TN ELSE PC=PC-10:IF PC<1 THEN
PC=PC+10
290 IF J1>60 THEN PC=PC+10:IF PC
>TN THEN PC=39
300 RETURN
310 HCOLOR8:GOSUB320:HCOLOR15:GO
SUB320:GOSUB260:RETURN
320 FOR V=1 TO 2:HLINE(BP(PC,0)-
V,BP(PC,1)-V)-(BP(PC,0)+25+V,BP(
PC,1)+40+V),PSET,B:NEXT:RETURN
330 OK=-1:FOR T=1 TO TN:IF PD(T,
1)=DC AND T<PC AND PD(T,0)<13
AND PD(T,0)<14 THEN OK=0:RETURN
340 NEXT:RETURN
350 IS=INKEYS:IF IS="" THEN 350 ELS
E IF IS="G" THEN IS="GREEN":IC=0
ELSE IF IS="Y" THEN IS="YELLOW
":IC=1 ELSE IF IS="B" THEN IS="B
LUE":IC=2 ELSE IF IS="R" THEN IS
="RED":IC=3 ELSE SOUND10,2:GOTO3
50
360 SOUND200,1
370 MS$="COLOR IS "+IS:GOSUB480
380 DC=IC:HPAINT(BP(40,0)+4,BP(4
0,1)+24),IC,8:HPAINT(BP(40,0)+12
,BP(40,1)+18),IC,8:RETURN
390 SP=40:GOSUB250:SP=PC:GOSUB25
0
400 DC=PD(PC,1):DN=PD(PC,0):NC=D
N:CC=DC:SP=40:GOSUB580:IF PC<>TN
THEN PD(PC,0)=PD(TN,0):PD(PC,1)
=PD(TN,1):SP=PC:NC=PD(PC,0):CC=P
D(PC,1):GOSUB580
410 SP=TN:GOSUB250:TN=TN-1:IF PC
=TN+1 THEN PC=PC-1
420 RETURN
430 CT=CT+1:CD(CT,0)=D(Q,0):CD(C

```

```

T,1)=D(Q,1):Q=Q+1:CT=CT+1:CD(CT,
0)=D(Q,0):CD(CT,1)=D(Q,1):Q=Q+1:
RETURN
440 FOR Z=SN TO EN:FOR SU=0 TO 3
:C(Z,SU)=VH:NEXT SU,Z:RETURN
450 MS$="SHUFFLING":GOSUB480:FOR
D=1 TO 108
460 PALETTE5,RND(63):PALETTE6,RN
D(63):PALETTE7,RND(63):CN=RND(15
)-1:CC=RND(4)-1:IF C(CN,CC)=0 TH
EN 460 ELSE D(D,0)=CN:D(D,1)=CC:
C(CN,CC)=C(CN,CC)-1
470 NEXT D:RETURN
480 HCOLOR13:HLINE(0,183)-(320,1
92),PSET,BF:HCOLOR8:SOUND200,1:H
PRINT(1,23),MS$:FOR U=1TO 1000:N
EXT:SOUND150,1:ID INPUT
490 MS$="INVALID INPUT":GOSUB480
:SOUND10,1:FOR H=1 TO 1000:NEXT:
GOSUB500:RETURN
500 MS$="Play which card?":GOSU
B480:HPRINT(22,23),"I HOLD":HPRI
NT(29,23),CT:IF CT>1 THEN HPRINT
(33,23),"CARDS" ELSE HPRINT(33,2
3),"CARD"
510 RETURN
520 MS$="What is color?":GOSUB
480:INPUT DC:RETURN
530 FOR QQ=1 TO TN:IF PD(QQ,1)=D
C THEN OK=0:RETURN
540 NEXT:RETURN
550 CC=15:GOSUB610:GOSUB560:DN=M
C:DC=CC
560 CX=BP(PC,0):CY=BP(PC,1):NC=P
D(PC,0):CC=PD(PC,1):RETURN
570 CX=BP(TN,0):CY=BP(TN,1):NC=P
D(TN,0):CC=PD(TN,1):GOTO590
580 CX=BP(SP,0):CY=BP(SP,1)
590 IF NC>12 THEN CC=8
600 IF NC>9 AND NC<>15 THEN OS=4
ELSE OS=0
610 HCOLOR CC:HLINE(CX,CY)-(CX+2
5,CY+40),PSET,B:HPRINT(CX+1,CY+1
),4,CC:HCIRCLE(CX+22,CY+22+OS),2
0,CC,1,.50,.75:HCIRCLE(CX+3,CY+(
18-OS)),20,CC,1,0,.25:HLINE(CX+3
,CY+14)-(CX+3,CY+36),PSET:HLINE(
CX+22,CY+2)-(CX+22,CY+20),PSET:H
PAINT(CX+1,CY+1),CC,CC
620 IF NC=14 THEN HLINE(CX+12,CY
)-(CX+12,CY+40),PSET:HLINE(CX,CY
+20)-(CX+25,CY+20),PSET:HPRINT(C
X+14,CY+22),0,8:HPRINT(CX+10,CY+
22),1,8:HPRINT(CX+10,CY+18),3,8:
HPRINT(CX+14,CY+18),2,8
630 IF CC=1 THEN CD=8 ELSE CD=4
640 IF NC>9 THEN 680
650 HNS="BM"+STR$(CX+2)+", "+STR$(
CY+2)+"C"+STR$(CD)+"S4"+NS(NC):
HDRAW HNS
660 LNS="BM"+STR$(CX+19)+", "+STR
$(CY+32)+"C"+STR$(CD)+NS(NC):HDR
AW LNS
670 FOR K2=0 TO 1:FOR K=0 TO 1:M
NS="BM"+STR$(CX+8+K)+", "+STR$(CY
+14+K2)+"C"+STR$(CC)+"S8"+NS(NC)
:HDRAW MNS:NEXT K,K2:RETURN
680 CD=4:HNS="BM"+STR$(CX+2)+", "+
STR$(CY+7)+"C"+STR$(CD)+"S4"
690 LNS="BM"+STR$(CX+2)+", "+STR$(
CY+39)+"C"+STR$(CD)+"S4"
700 IF NC=14 THEN MS$="DRAW":GOSU
B840:HNS=HNS+M1$:HDRAW HNS:MS$="F
OUR":GOSUB840:LVS=LVS+M1$:HDRAW
LVS
710 IF NC=13 THEN MS$="WILD":GOSU
B840:HNS=HNS+M1$:HDRAW HNS:MS$="C

```

```

ARD":GOSUB840:LVS=LVS+M1$:HDRAW
LVS:VN=23:GOSUB760
720 IF NC=12 THEN MS$="DRAW":GOSU
B840:HNS=HNS+M1$:HDRAW HNS:MS$="T
WO":GOSUB840:LVS=LVS+M1$:HDRAW L
VS:VN=4:GOSUB760
730 IF NC=10 THEN MS$="SKIP":GOSU
B840:HNS=HNS+M1$:HDRAW HNS:MS$="T
URN":GOSUB840:LVS=LVS+M1$:HDRAW
LVS:VN=19:GOSUB760
740 IF NC=15 THEN MS$="ONO":GOSUB
840:FOR K2=0 TO 1:FOR K1=0 TO 1:
MNS="BM"+STR$(CX+1+K1)+", "+STR$(
CY+25+K2)+"S6"+M1$:HDRAW MNS:NE
X T K1,K2
750 RETURN
760 FOR K1=0 TO 1:FOR K2=0 TO 1:
MNS="BM"+STR$(CX+7+K1)+", "+STR$(
CY+26+K2)+"C8S8"+VS(VN):HDRAW M
N$:NEXTK2,K1:RETURN
770 NS(0)="BRGD4FR2EU4HNL2BR2":N
$(1)="BR2NGD6NLRBU6BR2":NS(2)="B
DER2FDG4R4BU6BR1":NS(3)="BDR2FD
GNLFDGL2HBR5BU5":NS(4)="BD3NR4E3
ND6BR2":NS(5)="NR4D3R3FDGL2HBR5B
U5":NS(6)="BDNED4FR2EUNL3BU3NL2
RFBR2":NS(7)="BD6UE4UNL4BR"
780 NS(8)="BRGDFNR2GDFFR2EUHUUHL
2BR2":NS(9)="BD5FR2EU4HL2GDFR3BU
3BR1"
790 VS(0)="BR6":VS(1)="U5ER2FD2N
L4D3BR2":VS(2)="U6R3FDGNL3FDGLN3
BR3":VS(3)="BRNR2HU4ER2FBD4GBR3"
:VS(4)="U6R3FD4GNL3BR3":VS(5)="N
R4U3R3U3R4BD6BR2":VS(6)="U3NR3U
3R4BD6BR2":VS(7)="BRNR2HU4ER2FBD
3NL2DGBR3":VS(8)="U3NU3R4NU3D3BR
2"
800 VS(9)="BRRU6NLRBD6LBR4"
810 VS(10)="BU2DFR2ENU5BDBR2":VS
(11)="U3NU3RNE3F3BR2":VS(12)="NU
6R4BR2":VS(13)="U6F2NDE2D6BR2":V
S(14)="U5NUF4NU5DBR2":VS(15)="BR
HU4ER2FD4GNL2BR3":VS(16)="U6R3FD
GNL3BD3BR3":VS(17)="BRHU4ER2FD4N
H2NFGNL2BR3":VS(18)="U6R3FDGLNL2
F2DBR2"
820 VS(19)="BUFR2EUNL2HUER2FBD5B
R2":VS(20)="BR2U6NL2R2BD6BR2":VS
(21)="BUNU5FR2ENU5BDBR2":VS(22)=
"BU2NU4F2E2NU4BD2BR2":VS(23)="BU
NU5FENU2FENU5BDBR2":VS(24)="UE4U
BL4DF4DBR2":VS(25)="BU5NUF2ND3E2
NUBD5BR2":VS(26)="BU6R4DG4DR4BR2
"
830 RETURN
840 M1$="":FOR U=1 TO LEN(MS):M1
=ASC(MID$(MS,U,1))-64:M1$=M1$+VS
(M1):NEXT
850 IF NC=15 THEN M1$="C10"+M1$:
RETURN
860 IF CC=1 THEN M1$="C8"+M1$ EL
SE M1$="C4"+M1$
870 RETURN
880 POK65496,0:END
890 PRINT"ERROR ";ERNO;" IN LINE
";ERLIN:GOTO880
900 IF CT=0 OR TN=0 OR CT>38 OR
TN>38 THEN 1320
910 BI=-1:BN=-1:BC=-1:MS$="MY TU
RN. I HOLD"+STR$(CT)+" CARD":
IF CT>1 THEN MS$=MS$+"S"
920 GOSUB480
930 IF TN<3 THEN GOTO1300

```

continued on page 23



CoCo3 shines again.

CoCo WORD

BUSINESS

by Brett Hooker

COCOWORD IS DESIGNED to be used when you need to do a small one page report of some description.

It makes use of the many features of the new CoCo 3, including the hi-res text screen plus LPEEK and LPOKE commands.

If you look closely at the program you would see that the hi-res screen is accessed using LPEEK and LPOKE. It is a very basic wordprocessor having very small tasks that do not require a lot of editing.

In future editions I may produce updates of this program incorporating colours and more editing area/functions.

For those of you who need a program from which you can just type in something and get a few copies printed out, this is for you.

Functions are:-

- <Clear> - Cepter the current line.
- <F2> - raise the column width.
- <SHFT-F2> - lower column width.
- <CTRL> - print out a copy.
- <SHFT-UP> - save page.
- <SHFT-DVN>- load page.
- <ARROWS> - move cursor.

When you select a load/save command the computer emits a tone. Following that tone you press <R> to abort load/save or any other key to continue.

When you start-up the screen width is set at 32 columns. It is advisable that you alter this to suit your printer straight away, or you could even change the program line to make it easier.

Most of all have fun using this program.

```

1 REM *****
  ** COCOWORD V1.5 **
  ** WRITTEN BRETT HOOKER **
  ** NOVEMBER 1986 **
  *****

2 GOTO 10
3 CSAVE"3WORDV5"
4 END
6 WIDTH 32
10 CLEAR 5000
20 WIDTH 32
70 PCLEAR 8
80 ON BRK GOTO 9999
100 WD=32:LI=0:PO=0
110 LP=#H6C000
120 CLS
130 PRINT @ 69,"COCO WORD V1.5"
140 PRINTTAB(5);"WRITTEN BRETT H
OOKER"
150 PRINTTAB(5);"NOVEMBER 1986"
155 PRINT
160 PRINTTAB(5);"FOR EXCLUSIVE U
SE ON THE"
170 PRINT TAB(5);"*****
*****"
180 PRINT TAB(5);"*COLOR COMPUTE
R THREE*"
185 PRINT TAB(5);"*****
*****"
200 PRINT
210 PRINT TAB(5);"PRESS <ENTER>"
220 AS=INKEY$
230 IF AS<>""THEN 300
240 GOTO 220
300 WIDTH 80
990 REM *****
992 REM *** MAIN PROGRAM ***
994 REM *** CONTROLLED HERE ***
996 REM *****
1000 CLS
1010 LOCATE 0,23
1020 PRINT"WIDTH=";WD;
1090 AS=INKEY$
1100 PL=LP+(PO+2)+(LI#80)
1110 PP=LPEEK(PL)
1120 LPOKE PL,255
1130 LPOKE PL,PP
1140 IFAS=""THEN 1090
1145 IF ASC(AS)=94 THEN 1190
1146 IF ASC(AS)=214 THEN GOSUB35
00:GOTO 1090
1147 IF ASC(AS)=189 THEN GOSUB 4
000:GOTO1090
1148 IF ASC(AS)>90 AND ASC(AS)<9
6 THEN 1155
1150 IF ASC(AS)>31 THEN 1500
1155 TI=HH
1160 IF AS=CHR$(8)THEN GOSUB2000
:GOTO1090
1170 IF AS=CHR$(9)THEN GOSUB2100
:GOTO1090
1180 IF AS=CHR$(10)THENGOSUB2200

```

```

:GOTO1090
1190 IF AS=CHR$(94)THENGOSUB2300
:GOTO1090
1200 IF AS=CHR$(13)THEN PO=0:GOS
UB2200:GOTO1090
1210 IF AS=CHR$(4)THENGOSUB3000:
GOTO1090
1220 IF AS=CHR$(12)THENGOSUB5000
:GOTO1090
1230 IF AS=CHR$(95)THEN GOSUB700
0:GOTO1090
1240 IF AS=CHR$(91)THEN GOSUB800
0:GOTO1090
1499 GOTO 1090
1500 LPOKE PL,ASC(AS)
1510 PO=PO+1
1520 IF PO=WD THEN PO=0:LI=LI+2
1530 IF LI=46 THEN LI=44:SOUND 1
,1
1540 GOTO 1090
1999 REM ***BACKSPACE***
2000 PO=PO-1
2010 IF PO<0 THEN PO=WD-1:LI=LI-
2
2020 IF LI<0 THEN LI=0:SOUND 1,1
2030 RETURN
2099 REM ***FORWARD SPACE***
2100 PO=PO+1
2110 IF PO=WD THEN PO=0:LI=LI+2
2120 IF LI=46 THEN LI=44:SOUND1,
1
2130 RETURN
2190 REM ***DOWN ARROW***
2200 LI=LI+2
2210 IF LI=46 THEN LI=0
2220 RETURN
2290 REM ***UP ARROW***
2300 LI=LI-2
2310 IF LI<0 THEN LI=44
2320 RETURN
3000 WD=WD+1:IF WD>80 THEN WD=1
3010 LOCATE 0,23
3020 PRINT"WIDTH=";WD;
3030 RETURN
3500 WD=WD-1:IF WD<1 THEN WD=80
3510 LOCATE0,23
3520 PRINT"WIDTH=";WD;
3530 RETURN
3999 REM ***PRINT TO PRINTER***
4000 A=#H6C000
4005 FOR C=0 TO 44 STEP 2
4010 FOR B=0 TO WD-1
4020 ZZ=LPEEK(A+(C#80)+(B#2))
4030 PRINT #,-2,CHR$(ZZ);
4040 NEXT B
4050 NEXT C
4080 RETURN
4999 REM ***CENTRE A LINE***

5000 A=#H6C000
5010 VW=A+(LI#80)
5015 LI$=""
5020 FOR B=0 TO WD-1

```


ONO

continued from page 21

```

5030 LIS=LIS+CHRS(LPEBK(VV))
5035 VV=VV+2
5040 NEXT B
5050 IF LEFT$(LIS,1)=" THEN LIS
=RIGHT$(LIS,LEN(LIS)-1):GOTO 505
0
5060 IF RIGHT$(LIS,1)=" THEN LI
S=LEFT$(LIS,LEN(LIS)-1):GOTO 506
0
5070 MI=INT(WD/2)
5080 LE=LEN(LIS)
5085 IF LE>WD THEN RETURN
5090 AA=AH6C000
5100 SS=AA+(LI*80)
5110 FOR B=SS TO SS+(79*2) STEP
2
5120 LPOKE B,32
5130 NEXT B
5140 LL=INT(LE/2)
5150 MI=MI-LL
5160 DD=AA+(LI*80)+(MI*2)
5165 IF LIS="" THEN RETURN
5170 FOR B=1 TO LEN(LIS)
5180 LPOKE DD,ASC(MID$(LIS,B,1))
5190 DD=DD+2
5200 NEXT B
5210 RETURN
6999 REM ***SAVE DATA***
7000 SOUND 100,50
7004 TS=INKEYS
7005 IF TS<>" THEN 7007
7006 GOTO 7004
7007 IF TS="R" THEN RETURN
7010 A=AH6C000
7015 OPEN "O",#-1,"WORDFILE"
7020 FOR C=0 TO 22
7030 FOR B=0 TO 79
7040 PRINT #,-1,CHRS(LPEBK(A+(B*2
)+(C*2*80)))
7050 NEXT B
7060 NEXT C
7070 CLOSE #-1
7080 RETURN
7999 REM ***LOAD DATA***
8000 SOUND 200,50
8004 TS=INKEYS
8005 IF TS<>" THEN 8007
8006 GOTO 8004
8007 IF TS="R" THEN RETURN
8010 A=AH6C000
8015 OPEN "I",#-1,"WORDFILE"
8020 FOR C=0 TO 22
8030 FOR B=0 TO 79
8040 INPUT #,-1,MNS
8042 IF MNS="" THEN MNS=" "
8045 LPOKE A+(B*2)+(C*2*80),ASC(
MNS)
8050 NEXT B
8060 NEXT C
8070 CLOSE #-1
8080 RETURN
9999 GOTO1090

```

```

940 FOR K=1 TO CT
950 IF DN=14 AND CT>1 THEN BI=1:
BN=CD(1,0):BC=CD(1,1):FOR K=2 TO
CT:IF CD(K,0)>BN THEN BN=CD(K,0
):BC=CD(K,1):BI=K:NEXT ELSE NEXT
960 IF DN=14 THEN 1130
970 IF CD(K,1)=DC AND CD(K,0)=12
THEN BI=K:GOTO1130 ELSE IF DN=1
2 AND CD(K,0)=12 THEN BI=K:GOTO1
130
980 IF CD(K,1)=DC AND CD(K,0)=10
THEN BI=K:GOTO1130 ELSE IF DN=1
0 AND CD(K,0)=10 THEN BI=K:GOTO1
130
990 NEXT K
1000 BN=CD(1,0):BC=CD(1,1):OK=0
1010 IF CT=1 THEN 1050
1020 FOR K=2 TO CT:IF CD(K,0)>BN
AND CD(K,1)=DC THEN BN=CD(K,0):
BC=CD(K,1):BI=K:OK=-1
1030 IF CD(K,0)>BN AND CD(K,0)=D
N THEN BN=CD(K,0):BC=CD(K,1):BI=
K:OK=-1
1040 NEXTK:IF OK THEN 1130
1050 FOR K=1 TO CT:IF CD(K,0)=DN
OR CD(K,1)=DC THEN BI=K:GOTO113
0
1060 NEXTK
1070 FOR K=1 TO CT:IF CD(K,0)=13
THEN BI=K:IC=CD(K,1):GOTO1130
1080 NEXTK:OK=-1:FOR K=1 TO CT:IF
CD(K,1)=DC AND CD(K,0)>13 AND
CD(K,0)<>14 THEN OK=0
1090 NEXTK:IF OK THEN FOR K=1 TO
CT:IF CD(K,0)=14 THEN BI=K:GOTO
1130 ELSE NEXT
1100 MNS="I'm taking a card from
the pack":GOSUB480:CT=CT+1:CD(C
T,0)=D(Q,0):CD(CT,1)=D(Q,1):Q=Q+
1
1110 IF Q=109 THEN Q=1
1120 GOTO150
1130 SP=40:GOSUB250:GOSUB1260
1140 IF CD(BI,0)=14 THEN FOR EK=
1 TO 4:GOSUB1290:NEXTEK:GOSUB127
0:GOTO900
1150 IF CD(BI,0)=12 THEN FOR EK=
1 TO 2:GOSUB1290:NEXTEK:GOSUB127
0:GOTO900
1160 IF CD(BI,0)=10 THEN GOSUB12
70:GOTO 900
1170 IF CD(BI,0)=13 THEN GOSUB38
0:GOSUB1270:GOTO150
1180 GOSUB1270
1190 GOTO150
1200 FOR K=1 TO CT:IF CD(K,0)=13
THEN1230
1210 IF CD(K,0)=14 THEN IP=K:GOT
O1240
1220 NEXT:GOTO940
1230 BI=K:IC=CD(K,1):GOTO 1130
1240 OK=-1:FOR K=1 TO CT:IF CD(K
,1)=DC AND K<>IP AND CD(K,0)>13
AND CD(K,0)<>14 THEN 940
1250 NEXT:BI=IP:GOTO1130
1260 DC=CD(BI,1):DN=CD(BI,0):NC=
DN:CC=DC:SP=40:GOSUB580:RETURN
1270 IF BI<>CT THEN CD(BI,0)=CD(
CT,0):CD(BI,1)=CD(CT,1)
1280 CT=CT-1:RETURN

```

```

1290 TN=TN+1:PD(TN,0)=D(Q,0):PD(
TN,1)=D(Q,1):GOSUB570:Q=Q+1
1300 IF Q=109 THEN Q=1:RETURN
1310 RETURN
1320 IF CT=0 OR TN>38 THEN GOSUB
1370
1330 IF XC>500 THEN MNS="I WIN.
PLAY AGAIN (Y/N)":GOSUB480:GOTO
1480
1340 IF TN=0 OR CT>38 THEN GOSUB
1430
1350 IF XP>500 THEN MNS="YOU WI
N. PLAY AGAIN (Y/N)":GOSUB480:GO
TO1480
1360 GOTO40
1370 RS=0:FOR Y1=1 TO TN
1380 IF PD(Y1,0)=14 OR PD(Y1,0)=
13 THEN RS=RS+50:GOTO1410
1390 IF PD(Y1,0)=12 OR PD(Y1,0)=
10 THEN RS=RS+20:GOTO1410
1400 RS=RS+PD(Y1,0)
1410 NEXT:XC=XC+RS:TS$=STR$(XC):
MNS="I WIN. ":GOSUB1420:RETURN
1420 MNS=MNS+"Score"+STR$(RS)+"
Total score "+TS$:GOSUB480:SOUD
D100,1:SOUND200,1:SOUND100,1:FOR
U=1 TO 3000:NEXT:RETURN
1430 RS=0:FOR Y1=1 TO CT
1440 IF CD(Y1,0)=14 OR CD(Y1,0)=
13 THEN RS=RS+50:GOTO1470
1450 IF CD(Y1,0)=12 OR CD(Y1,0)=
10 THEN RS=RS+20:GOTO1470
1460 RS=RS+CD(Y1,0)
1470 NEXT:XP=XP+RS:TS$=STR$(XP):
MNS="YOU WIN. ":GOSUB1420:RETURN
1480 IS=INKEYS:IF IS="" THEN 1480
ELSE IF IS="Y" THEN RUN ELSE IF
IS="N" THEN 880 ELSE 1480
1490 FOR R=0 TO 3:PALETTE 5+R,R:
NEXT:RETURN
1500 ZC=5:FOR R=180 TO 170 STEP-
5:FOR Z=R TO R-5 STEP-1:HCIRCLE(
160,190),Z,ZC,1,.65,.85:HCIRCLE(
160,190),Z+15,ZC,1,.65,.85:NEXTZ
:ZC=ZC+1:NEXT
1510 FOR K1=1 TO 3:FOR K2=1 TO 5
:MNS="BM"+STR$(70+K1)+" "+STR$(3
0+K2)+"$16C0"+V$(18)+"C1"+V$(1)+
"C2"+V$(9)+"C3"+V$(14)+"C0"+V$(2
)+C1"+V$(15)+"C2"+V$(23):HDRAW
MNS:NEXT K2,K1:RETURN
1520 MNS="WANT TO KEEP PLAYING (
Y/N)":GOSUB480:GOTO1480

```

Hint ...

Stop Fast Scrolling

Stop your screen from rolling over the edge - see your listing scroll slowly up the screen ... with this shortie!

```

1 FOR X=1000 TO 1010
2 READ A:POKE X,A:NEXT I
3 POKE383,126:POKE 384,3:POKE 38
5,232:POKE422,126:POKE423,3:POKE
424,232
4 DATA 52,16,142,0,1,189,167,211
,53,16,57
5 NEV

```

Instructions: RUN the program, then POKE 1003,x for different speeds.



It's a mad, mad, mad...

MAD MILE

GAME

by Steve Youngberry

The mad mile is a race around a circuit for up to six players, and up to nine laps.

If you try to take a corner faster than 80 MPH you will crash. There are two roving oil slicks - try to drive faster than 40MPH and you will also crash.

If you crash, you will miss out on your turn.

You have a four speed gearbox. The gear you select will help you to determine your speed. The speed through the gears are...

- 1st gear 20 - 80mph
 - 2nd gear 60 - 120mph
 - 3rd gear 100 - 160mph
 - 4th gear 140 - 200
- Have fun!

The Listing:

```

1 ***** The Mad Mile *****
2 ***** by *****
3 ***** Steve Youngberry *****
4 ***** Tara *****
5 ***** january 1987 *****
8 GOTO10
9 SAVE"233:3":END'1
10 ONBRK GOTO109
11 DIMY,X,T,Z,A:
12 CLS:POKE65497,0:CLER3000:POK
R359,57:POKE65314,52:POKE44014,8
7:POKE44015,66
13 :PALETTE0,0:PALETTE1,0:PALETT
E2,4:PALETTE3,5:PALETTE4,3:PALETT
E5,6:PALETTE6,7:PALETTE12,0:PAL
ETTE13,5:PALETTE14,34:PALETTE15,
14
14 HSCREEN2
15 IFFEEK(31000)=51THENPALETTE1,
2:GOTO36
16 LMS="U120R20M+30,70M+30,-70R2
0D120L20U70M-30,70M-30,-70D70L20
":LIS="U40R10D40L10BR20":LLS="U4
0R10D30R20D10L30BR40":LES="U40R3
0D10L20D5R10D10L10D5R20D10L30":S
PS="BR15
17 LAS="U40R30D40L10U15NL10BU10N
L10U5L10D5BD10D15L10BR40":LDS="U

```

```

40R20F10D20G10L20BE10U20R5F5D10G
5LS
18 HDRAW"C12BM20,160"+LM$:HPAINT
(25,155),12,12:HDRAW"BM130,100"+
LAS+LDS:HPAINT(135,95),12,12:HPA
INT(175,95),12,12
19 HDRAW"BM130,160"+LIS+LLS+LES:
HPAINT(135,155),12,12:HPAINT(155
,155),12,12:HPAINT(195,155),12,1
2
20 HPAINT(0,191),1,12:HPAINT(148
,73),1,12:HPAINT(188,80),1,12:PO
KE&HFFB1,2:HPRINT(16,6),"by WHYB
ILT
21 FORA=0TO15:HLINE(A,0)-(A,191)
,PSET:NEXT:FORA=320TO225STEP-1:H
LINE(A,0)-(A,191),PSET:NEXT:FORA
=0TO35:HLINE(0,A)-(320,A),PSET:N
EXT:FORA=191TO165STEP-1:HLINE(0,
A)-(320,A),PSET:NEXT
22 POKE&HFFB1,0:POKE&HFFB2,2
23 GOSUB41:PLAYT$(1)+T$(2):PLAYT
$(3)+T$(3)+T$(4)
24 HCOLOR0:HPRINT(5,22),"instruc
tions? <y/n>
25 AS=INKEY$:IFAS<>"Y" AND AS<>"
N" THEN25
26 POKE&HFFB1,2
27 IFAS="N" THEN36
28 HCLS:HCOLOR4:HPRINT(14,0),"Th
e Mad Mile":HLINE(113,10)-(207,1
0),PSET
29 HPRINT(5,2),"The Mad Mile
is a race around a":HPRINT(2,3)
,"circuit for up to six players
, and up":HPRINT(2,4),"to nine 1
aps.....
30 HPRINT(2,5),"If you try to ta
ke any corner faster":HPRINT(2
,6),"that 80 mph you will crash.
..... also":HPRINT(2,7),"there
are two roving oil slicks... try
":HPRINT(2,8),"to drive throu
gh one of these faster
31 HPRINT(2,9),"than 40 mph and
you will also crash...":HPRINT(2
,10),"If you crash you will miss
out on your next":HPRINT(2,11),
"turn....":HPRINT(2,12),"You ha
ve a four speed gearbox.... The
32 HPRINT(2,13),"gear you selec
t will help determine":HPRINT(
2,14),"your speed..... The spee
d through the":HPRINT(2,15),"gea
re are....":HPRINT(10,17),"1st g
ear.... 20 to 80":HPRINT(10,1
8),"2nd " ..... 60 to 120
33 HPRINT(10,19),"3rd " .....
100 to 160":HPRINT(10,20),"4th

```

```

" ..... 140 to 200":HPRINT(2,22
),"As drivers finish their tur
n will be":HPRINT(2,23),"cancell
ed... <<INKEY>>
34 EXEC44539
35 ***** SET UP STRING$ ****
36 C$="BH6ND2R4D2L2NL2D2NL2R8U2L
2U2R4D2L2D2R2D4L2D2R2D2L4U2R2U2B
U2L2D2R2BDL8D2R2D2L4U2R2U2L2U4B
D2BR6
37 FORA=1TO6:FORB=1TO4:C$(A,B)="
C"+STR$(A)+"A"+STR$(B)-1)+C$:NE
XT:NEXT
38 O4(1)=1:O4(2)=1:O1(1)=31:O1(2
)=31:O2(1)=21:O2(2)=21:O7=0:LH=2
39 FORA=1TO5:READAS$:EX$(A)=AS:NE
XT:FORA=1TO5:READAS$:FF$(A)=AS:NE
XT
40 GOSUB41:GOTO46
41 T$(1)="T2O2L4CO1L4AL4AL8AO2CD
O1L4GL4L4GGL8GAL4B-O2FL8FFFL8DC
O1L2A
42 T$(2)="O2L4CL8DL4DL8DL8GFL12B
FDL2CO1L8FGL4AO2L8DL4CL8CO1B-L4A
L8EL4EL2F
43 T$(3)="L9CO2FE-DCCO1B-O2L4DL8
FDL4.DL8FO1L4B-O2L8FDDCO1L4B-O2C
L8DCL4CO1B-L8AGL2.F
44 T$(4)="L9FO2CDL8B-DL4E-L8EFL4
GL2.G
45 RETURN
46 HCLS:HDRAW"A0":HCOLOR15:FORA=
30TO290STEP20:HLINE(A,0)-(A,20),
PSET:HLINE(A,160)-(A,180),PSET:N
EXT:FORA=20TO160STEP20:HLINE(10,
A)-(30,A),PSET:NEXT
47 FORA=50TO290STEP20:HLINE(A,40
)-(A,60),PSET:HLINE(A,120)-(A,14
0),PSET:NEXT:HDRAW"BM10,OND180R3
00D20NL280D20NL260D20L20NU40L240
D20NR20D20NR20D40NR260U20R20NU60
R240D60L20NL280U20NU20NR20L260U1
60
48 HPRINT(10,8),"No. of Players?"
49 IS=INKEY$:IFIS<"1"ORIS>"6"THE
N49ELSEI=VAL(IS):HPRINT(24,8),IS
50 HCOLOR15:HPRINT(10,10),"No. of
Laps?":
51 LS=INKEY$:IFLS<"1"ORLS>"9"THE
N51ELSEL=VAL(LS):HPRINT(22,10),L
:FORA=1TOI:L(A)=L:NEXT
52 HLINE(265,69)-(315,111),PSET,
BF:HLINE(265,89)-(315,93),PRESET
,BF:HCOLOR0:HPRINT(34,9),"TOTAL"
:HPRINT(34,10),"LAPS":HPRINT(37,
10),L:I:HPRINT(35,12),"LAPS":HPR
INT(34,13),"TO GO":HPRINT(33,12),
L
53 C=32-(I*2):H=268-I*16:FORA=1T

```

```

01: HDRAW"BM"+STR$(H)+"", 111"+C$(A
,4): HPRINT(C,12), A:C=C+2:H=H+16:
NEXT:GOSUB114
54 HLINE(80,62)-(200,88), PRESET,
BF
55 FORA=1TO1:Y(A)=180:X(A)=130:T
(A)=20:Z(A)=0:A(A)=3:NEXT
56 FORMM=1TO1:HCOLORMM:HLINE(167
,62)-(177,72), PRESET, BF:HLINE(23
7,78)-(247,88), PRESET, BF:HLINE(1
34,95)-(160,105), PRESET, BF:HPRIN
T(10,8),"Player No.":HPRINT(20,8
),MM
57 FORK=1TO2:O3(K)=O3(K)+1:IFO4(K
)=O3(K) THEN O4(K)=RND(5):O3(K)
=0:O7=0:GOSUB102:O7=14:GOSUB93:N
EXTELSENEXT
58 IFY(MM)=.5THENNEXT:GOTO56
59 HCOLORMM:IFO8(MM)=1THENHPRINT
(7,4),"Player":HPRINT(13,4),MM:H
PRINT(16,4),"misses this go":O8(M
M)=0:GOSUB83:GOSUB84:NEXT:GOTO5
6
60 HCOLORMM:HDRAW"BM"+STR$(Y(MM)
)+","+STR$(X(MM))+C$(MM,A(MM)):I
FMG(MM)=1THENMG(MM)=0:HPRINT(7,4
),"Player":HPRINT(13,4),MM:HPRIN
T(16,4),"misses this turn":GOSUB
83:GOSUB84:NEXT:GOTO56
61:HCOLORMM
62 Y=Y(MM):X=X(MM):T=T(MM):Z=Z(M
M):A=A(MM)
63 HCOLOR15:HPRINT(10,10),"Which
gear? (1-4)"
64 AS=INKEYS:IFAS<>"" THENAS=""
65 GS=INKEYS:IFGS<"1"ORG">"4"THE
N65ELSEC=VAL(GS):HPRINT(29,10),G
66 G=RND(4)+(2*(G-1)):IFG>4THENG
C=1

```

```

67 HPRINT(10,12),"Speed":HPRINT(
16,12),G*20
68 FORN=1TOG
69 HLINE(Y-6,X-6)-(Y+6,X+6),PRES
ET,BF
70 IFO1(1)=Y-9 ANDO2(1)=X-9 ANDG
>2THEN103
71 IFO1(2)=Y-9ANDO2(2)=X-9ANDG>2
THEN103
72 IFY=300ANDX=130THENT=Q:Z=20:A
=4:GOTO81
73 IFY=300ANDX=170THENT=-20:Z=0:
A=1:GOTO81
74 IFY=20ANDX=170THENT=0:Z=-20:A
=2:GOTO81
75 IFY=20ANDX=10THENT=20:Z=0:A=3
:GOTO81
76 IFY=300ANDX=10THENT=0:Z=20:A=
4:GOTO81
77 IFY=300ANDX=50THENT=-20:Z=0:A
=1:GOTO81
78 IFY=60ANDX=50THENT=0:Z=20:A=4
:GOTO81
79 IFY=60ANDX=130THENT=20:Z=0:A=
3:GOTO81
80 GOTO85
81 IFGC=1THENHCOLOR15:MG(MM)=1:H
PRINT(7,4),EX$(RND(5)):HPRINT(14
,4),"TO FAST..MISS NEXT GO":GOSU
B82:GOSUB84:N=G:GOTO86:ELSE85
82 PLAY"T255V3101":PLAY"V31":FOR
TT=8TO31:PLAY"V-CDEFG":NEXT:RETU
RN
83 PLAY"T255O2V31":FORTT=8TO31:P
LAY"Y-CD":NEXT:RETURN
84 HLINE(49,30)-(278,39),PRESET,
BF:RETURN
85 Y=Y+T:X=X+Z
86 HDRAW"BM"+STR$(Y)+"", "+STR$(X)

```

```

+C$(MM,A)
87 IFY=180ANDX=130GOSUB104
88 NEXT
89 Y(MM)=Y:X(MM)=X:T(MM)=T:Z(MM)
=Z:A(MM)=A:GC=0
90 FORA=1TO1:IFY(A)=.5THENNEXTEL
SEHDRAW"BM"+STR$(Y(A))+","+STR$(
X(A))+C$(A,A(A)):NEXT
91 NEXT
92 GOTO56
93 OS=RND(8):ONOS GOSUB94,95,96,
97,98,99,100,101:RETURN
94 O1(K)=RND(12)*20+31:O2(K)=41:
GOSUB102:RETURN
95 O1(K)=RND((14)-1)*20+31:O2(K)
=1:GOSUB102:RETURN
96 O1(K)=291:O2(K)=RND((3)-1)*20
+1:GOSUB102:RETURN
97 O1(K)=11:O2(K)=RND((8)-1)*20+
21:GOSUB102:RETURN
98 O1(K)=51:O2(K)=RND(5)*20+21:G
OSUB102:RETURN
99 O1(K)=291:O2(K)=RND(3)*20+101
:GOSUB102:RETURN:***** BOTTOM
RIGHT
100 O1(K)=RND(12)*20+31:O2(K)=12
1:GOSUB102:RETURN
101 O1(K)=RND((14)-1)*20+31:O2(K
)-161:GOSUB102:RETURN:***** BO
TTOM
102 HCOLORO7:HLINE(O1(K),O2(K))-
(O1(K)+18,O2(K)+18),PSET,B:HLINE
(O1(K)+2,O2(K)+2)-(O1(K)+16,O2(K
)+16),PSET,B:HPAINT(O1(K)+1,O2(K
)+1),O7,O7:GOSUB113:RETURN
103 HCOLORMM:N=G:O8(MM)=1:HPRINT
(6,4),EX$(RND(5)):HPRINT(13,4),"
OIL SLICK MISS NEXT GO":GOSUB82:
GOSUB84:GOTO86
104 L(MM)=L(MM)-1:IFL(MM)<1THENL
F=LF+1:LH=LH+5:HPRINT(LH,18),"No
":HPRINT(LH+1,18),MM:HPRINT(LH,1
9),FF$(LF):HLINE(174,124)-(186,1
36),PRESET,BF:N=G:Y=.5:IFLF>1-2
THEN109
105 FORFR=1TO1:IFFR=MM THENNEXTE
LSEIFL(MM)<L(FR) THENLA=LA+1:NEX
T
106 IFLA=I-1 THENL=L-1:HCOLOR15:
HLINE(265,94)-(279,103),PSET,BF:
HCOLORO:HPRINT(33,12),L:HCOLORMM
107 LA=0:RETURN
108 IFL(MM)>OTHERRETURN
109 HLINE(73,61)-(250,90),PRESET
,BF:HPRINT(10,9),"Another game <
y/n>"
110 AS=INKEYS:IFAS="" THEN110ELSE
IFAS="Y" THENPOKE31000,51:RUN
111 IFAS<>"F" THEN110ELSEPALETTER
GB:POKE65496,0:PRINT@500,"by WHY
BILT":END
112 RETURN
113 IFO1(K)=171ANDO2(K)=121THEN1
14ELSERETURN
114 HCOLOR15:FORA=170TO180STEP10
:FORB=120TO130STEP10:HLINE(A,B)-
(A+4,B+4),PSET,BF:HLINE(A+6,B+6)
-(A+9,B+9),PSET,BF:NEXT:NEXT:RET
URN
115 DATA CRUNCH..,WHAM..,POW..,BO
OF..,THUD..,1st,2nd,3rd,4th,5th

```

HINT....

Reset Protect

Whenever one presses the reset button while running a BASIC program, the screen is cleared and the OK prompt is shown. Wouldn't it be nice to press the reset button and automatically have your program RUN? ie, whenever you press the reset button, the computer will reset the variables and re-RUN your program for you.

Well with the following program, this is possible. Just include it in the first few

lines of your program followed by the rest of your program.

NOTE: this program will only work on the CoCo 1.

```

10 CLEAR200,31000
20 FOR X=32742 TO 32767:READ
1:POKE X,1:NEXT
30 EXEC32762
40 DATA 58, 142, 58, 18, 16,
50 DATA 222, 33, 48, 140, 246,
60 DATA 159, 166, 28, 175, 127,
70 DATA 255, 64, 126, 173, 192,
80 DATA 48, 140, 236, 159, 114,
90 DATA 57

```

COME TO at Bunde



Conf '87 this year is to be held at the Uniting Church's campsite in Bundeena NSW.

This is a particularly pretty area of Sydney, situated on the northern tip of the Royal National Park, in Port Hacking.

The water views are fabulous, and the bushwalks are amongst the best in Australia.

Not that you'll have anytime during conference for these things, because as usual, the conference will be jam packed with all sorts of things to see and do!

The big news this year will obviously be the growing use of OS-9 Level 2 on the CoCo 3's; and Conf '87 will be the definitive place to see this excellent system.

By that time initial users will have had time to sort the system out and create some really interesting stuff.

But it is not just OS-9 that is of interest this year.

With the release of the new T1000 EX and SX, interest in these machines has never been higher. We'll have a number of these computers at the conference, as well as their big brothers, the T3000 series, which we'll be putting through their paces.

We've had continuing interest in some of the more diverse subjects covered in the magazine at past conferences, so again this year we'll have tutorials on hardware mods and on Forth.

There'll be Basic Basic and Advanced Basic courses, and an Assembly Language tutorial as well.

Other computers will be discussed, principally the 68000 series of computers, and of course, we'll be showing Goldlink 642 on Viatel - and Videotex in general.

Conference is a place to meet old friends, to meet the people behind the names in the magazine, to learn a lot of new

information, to see the latest Tandy equipment.

We hope you'll come. We're sure you'll be glad you did. But please hurry your booking, because accommodation (which is not obligatory) and places at the conference, are both limited by the size of the centre.

The cost is increased over previous years due entirely to the fact that we are doing it in Sydney which is a good deal more expensive than the Gold Coast!

On the other hand, many of you will save by not having the additional travelling expenses associated with getting to the Gold Coast.

We aim to make the conference a family affair, and the location is a good one for people with families who are less interested in computers, but who would still like to be with dad or mum for the weekend.

The family can take a ferry trip, go for bush walks, or just laze on the beach, whilst you do your thing at the conference.

CONF '87

na N.S.W.

CONF '87

P.O. BOX 1742, SOUTHPORT. QLD. 4215 Phone (075) 510 015

Rates

Accommodated (1) \$87.00
Family of 2, + \$68.00 = \$155.00
Additional family members \$52.00 ea
Includes supper Friday evening, breakfast
lunch and dinner on Saturday and breakfast
and lunch on Sunday plus all accommodation.

Non Accommodated Rates

	One day	Two Days
One person	\$40.00	\$58.00
Sat Evening Meal	\$12.00	\$12.00
	=====	=====
	\$52.00	\$70.00

Additional family \$31.50 \$45.50 /person
Includes morning / afternoon tea and lunch.

\$20.00 deposit required with booking;
final payment to be made by 15th July 1987.

LOCATION:-

Uniting Church's campsite
Bundeena NSW

DATE:- 8th & 9th August, 1987

REGISTER NOW!!

We can only accept a limited number of people this year. DON'T MISS OUT! on a top weekend of FUN, FRIENDSHIP and LEARNING.

Name:

Address:

Phone:

No. People attending:

SPEAK UP!:- Now is your chance to suggest your ideas for any tutorials we may not have mentioned. (participants only).

Tutorials likely to attend:

Please find enclosed:

chq/money order/bankcard/visa/mastercard

Card No.

Signature:



POLAR

16K ECB
by Mal McLaughlan

MEMORY SHIFTER

CoCo 3 + optional EDTASM

by Sean Murdoch

POLAR IS A graphics drawing program of interest to all ages. Four polar diagrams have been chosen that resemble flowers of various types, and can be printed on paper for colouring in.

For senior maths students, the program itself will illustrate how formulae can be used to direct the plotting of the hundreds of co-ordinates that go to make up the lobes (petals) of such a diagram.

The Listing:

```

1 *****POLAR FLOWERS*****
*****BY MAL MCLAUCHLAN*****
*****MARCH, 1987*****
2 GOTO 10
3 SAVE"204:3":END'6
4 **FOUR FLOWER-LIKE DIAGRAMS
  USING POLAR GEOMETRY TO PLOT.
  TO HIGH SCHOOL CHILDREN THEY
  ARE polar graphs -- AND TO
  OUR LITTLES THEY ARE FLOWERS
*****
10 CLS:POKE 65495,0
15 PRINT@72,"polar graphics";:PR
INT@102,"(OR FOUR FLOWERS)";:PRI
NT@320,"FLOWERS FOR THE LITTLE O
NES...";:PRINT@384,"AND MAYBE A
HEADACHE FOR OTHERS.";
17 FOR DL=1 TO 2500:NEXT DL
20 CLS:PRINT@227,"JUST A FEW MOM
ENTS PLEASE..."
30 SX=254:SY=190:RA=0.90
40 HY=95:HX=127
50 R=SIN(8*Z)
60 DEF FNA(Z)=SIN(8*Z)
70 GOSUB 310:GOSUB 330:GOSUB 350
:GOSUB 370
80 POKE65494,0
90 CLS2:PRINT@162,"HOPE YOU LIKE
THE FLOWERS!";:PRINT@322,"OR AR
E YOU OLD ENOUGH TO ";:PRINT@35
4,"CALL THEM POLAR DIAGRAMS? ";
100 GOTO100
110 M=1.0E-30

```

MEMORY SHIFTER CAME ABOUT when I wanted to run through the interpreter code of my CoCo 3. Since this code is located in ROM pack memory, I needed to shift the memory to a lower location.

So I wrote this program. It is short and sweet and may convenience many CoCo users.

The program is self-explanatory. If you don't understand how to use it, then I suspect you won't be needing it.

EDTASM+ listing:

```

00100 *****
00110 *MEMORY SHIFTER*
00120 *****
00130 START LDX >1568
      LOAD X WITH SOURCE START
00140 LDY >1572
      LOAD Y WITH DESTINATION START
00150 MEMO10 LDA ,X+
      LOAD A WITH CONTENTS OF SOURCE
      POINTER AND INCREMENT POINTER
00160 STA ,Y+
      STORE A INTO DESTINATION
      POINTER AND INCREMENT POINTER
00170 CMPX >1570
      COMPARE X WITH SOURCE END
00180 BHI RETURN
      IF HIGHER, BRANCH TO RETURN
00190 BRA MEMO10
00200 RETURN RTS
      RETURN FROM SUBROUTINE
00210 END

```

```

120 FOR Z=0 TO 2*3.1416 STEP 0.1
130 R=ABS(FNA(Z))
140 IF M<R THEN M=R+0.1
150 NEXT Z
160 FOR I=1 TO 1000:NEXT I
170 CLS
180 GOSUB 290
190 FOR Z=0 TO 2*3.1416 STEP 0.0
1
200 R=FNA(Z)
210 U=HX+HY*RA*COS(A*Z)*R/M
220 IF V<0 OR U>SX THEN GOTO 260
230 V=HY+HY*SIN(B*Z)*R/M
240 IFV<0 OR V>SY THEN GOTO 260
250 GOSUB 300

```

The Listing:

```

0 *****
1 * MEMORY SHIFTER *
2 *****
3 *BY SEAN MURDOCH *
4 *****
5 * P.O. BOX 5 *
6 * BRINGELLY. *
7 * NSW. 2171. *
8 *****
9 * (047) 748 291 *
10 *****
11 GOTO13
12 SAVE"229:3":END'8
13 FORT=1536TO1554
14 READA:POKET,A
15 NEXTT
16 DATA190,6,32,16,190,6,36,166,
128,167,160,188,6,34,34,2,32,245
,57
17 CLS
18 INPUT"SOURCE START";S(1)
19 INPUT"SOURCE END";S(2)
20 INPUT"DESTINATION START";S(3)
21 FORT=1TOSSTEP2
22 A$=HEX$(S(INT(T/2)+1))
23 POKE1567+T,VAL("&H"+LEFT$(A$,
LEN(A$)-2))
24 POKE1568+T,VAL("&H"+RIGHT$(A$,
2))
25 NEXTT
26 EXEC1536
27 CLS:PRINT"LOCATIONS"S(1)"TO"S
(2)"HAVE":PRINT"BEEN COPIED AT"S
(3)"TO"S(2)-S(1)+S(3)
28 PRINT:PRINT"THIS IS A PRODUCT
OF":PRINT"merl software."

```

```

260 NEXT Z
270 FOR RPT=1 TO 20:PLAY"T25504A
B":NEXT RPT
280 FOR DL=1 TO 2000:NEXT DL
290 PMODE4,1:PCLS:SCREEN1,1:RETRN
300 PSET(U,V,1):RETURN
310 A=1:B=1:GOTO110
320 GOSUB 330
330 A=3:B=3:GOTO110
340 GOSUB 350
350 A=5:B=3:GOTO110
360 GOSUB 370
370 A=5:B=5:GOTO110

```

If you want to know where it's at, ask...

WHERE IS IT?

16K ECB + Disk
UTILITY

by Ron Simpkin

LISTED BELOW ARE some changes to the program from January 1987 called "Where is it?" by Brian Biggs.

The program originally didn't allow reading direct from the directory.

These are the modifications required to allow reading of the disk directory.

I have still allowed for manual insertion of program names. I've also allowed for more than 24 program names and the program has been rewritten for the DMP-110 printer.

I don't think I need to have cleared so much memory in line 10 but I like to be safe.

I hope this is of some use to other users.

The Listing:

```
0 GOTO10
1 '***** WHERE IS IT *****
  ***** ROM SIMPKIN *****
3 SAVE"222:3":END'8
10 CLEAR4000
20 DIMD$(80),C$(90),N$(80)
30 CLS:FORP=1157TO1179:READA:POKEP,A:NEXT
40 FORP=1189TO1205:READA:POKEP,A:NEXT
50 FORP=1289TO1301:READA:POKEP,A:NEXT
60 FORP=1327TO1330:READA:POKEP,A:NEXT
70 FORP=1413TO1435:READA:POKEP,A:NEXT
80 FORP=1455TO1458:READA:POKEP,A:NEXT
90 DIMG$(40)
100 FORX=1TO1000:NEXT
110 CLS:FORX=1TO10
120 PRINT@225,"SET 1200 BAUD RATE ON PRINTER
130 FORP=1TO50:NEXT
140 NEXT
150 GOSUB830
160 POKE150,41
170 N1$="123":N2$="ABC
180 CLS:PRINTSTRING$(32,"=");:X=0
190 PRINT"status: DISK NUMBER";N1$:PRINT"DISK CODE";N2$:PRINT"CURRENT DRIVE";D:PRINTSTRING$(32,"=");
200 PRINT@224,"1)CHANGE DISK NUMBER":PRINT"2)CHANGE DISK CODE":PRINT"3)CHANGE DRIVE":PRINT"4)ENTER PROGRAMS":PRINT"5)LOAD DISK MENU":PRINT"6)QUIT
210 A$=INKEY$:IF A$=""THEN210
220 IF A$="1"THENGOSUB860:PRINT@223,"";:INPUT"NEW DISK NUMBER";B$:IFLEN(B$)<>3OR B$=""THEN180ELSEN1$=B$:GOTO180
230 IF A$="2"THENGOSUB860:PRINT@223,"";:INPUT"NEW DISKCODE";B$:IFLEN(B$)<>3OR B$=""THEN180ELSEN2$=B$:GOTO180
240 IF A$="3"THENPRINT@423,":INPUT"NEW DRIVE";D:IFD<0ORD>3THEN240ELSE180
250 IF A$="4"THENGOTO330
260 IF A$="5"THEN GOTO880
270 IF A$="6"THENPRINT423,"ARE YOU SURE Y/N?":GOTO290
280 GOTO210
290 A$=INKEY$:IF A$=""THEN290
300 IF A$="N"THEN 180
310 IF A$="Y"THENCLS:PRINT"THANK YOU FOR USING THISPROGRAM":END
320 GOTO290
330 FORN=1TO40
340 CLS:PRINT12,N1$,"-";N2$:
350 PRINT@37,"ENTER THE PROGRAM NAMES":PRINT
360 PRINT@97,"<P>RINT <R>ESTART <D>IRECTORY":PRINT
361 G$(N)=NM$(X)
370 PRINT" PROGRAM #";N;G$(N)
371 IFNM=0 THEN40ELSEPRINTNM$(X)
380 X=X+1
390 IFX=NM+1 THEN40ELSE450
400 INPUTG$(N)
410 IFG$(N)="P"THEN460
420 IFG$(N)="R"THENGOSUB870:GOTO180
430 IFG$(N)="D"THEN760
440 IFLEN(G$(N))>8THENPRINT"TOO LONG (LIMIT:8 CHARACTERS)":GOTO370
450 NEXTN
460 CLS:PRINT@12,N1$,"-";N2$:PRINT:PRINT
470 T=65
480 FORQ=1TO N-1
490 PRINT@T,G$(Q):T=T+11
500 IFQ=3 ORQ=6 ORQ=9 ORQ=12 ORQ=15 ORQ=18 ORQ=21ORQ=24 THENT=T-1
510 NEXT
520 PRINT:PRINT
530 PRINTTAB(5)"IS THIS OK (Y/N) ?";
540 A$=INKEY$:IF A$=""THEN540
550 IF A$="Y"THEN590
560 IF A$="N"THEN190
570 GOTO540
580 REM START PRINTING
590 GOSUB830
600 CLS:PRINT@236,"PRINTING
610 PRINT#-2,CHR$(27);CHR$(31)
620 PRINT#-2,"";N1$,"-";N2$;
630 PRINT#-2,CHR$(27);CHR$(32)
631 PRINT#-2,CHR$(27);CHR$(20)
640 FORQ=1TO N-1
650 PRINT#-2,G$(Q),
660 IFQ/4=INT(Q/4)THENPRINT#-2,CHR$(27);CHR$(26)
670 NEXT
680 PRINT#-2,CHR$(27);CHR$(12);
690 PRINT#-2,CHR$(27);CHR$(19)
700 REM
710 PRINT@256,"PRINT SAME LABEL AGAIN Y/N?
720 A$=INKEY$:IF A$=""THEN720
730 IF A$="Y"THEN590
740 IF A$="N"THENGOSUB870:GOTO180
750 GOTO720
760 CLS:DIRD:PRINTTAB(9)FREE(D);"GRANS FREE";:EXEC44539:GOTO340
770 DATA68,73,83,75,96,76,65,66,69,76,69,82,96,104,80,82,79,71,82,65,77,83,105
780 DATA70,79,82,96,83,71,109,113,112,96,80,82,73,78,84,69,82
790 DATA66,89,96,66,73,76,76,96,83,69,77,80,70
800 DATA113,121,120,117
810 DATA13,15,4,9,6,9,5,4,32,2,25,32,2,18,9,1,14,32,2,9,7,19
820 DATA49,57,56,54
830 PE=PEEK(65314)AND1
840 IFPE=0 THENRETURNELSEPRINT@486,"PRINTER NOT ON LINE!";
850 GOTO830
860 PRINT@483,"MUST BE 3 CHARACTERS LONG";:RETURN
870 FORX=1TO24:G$(X)="" :NEXT:RETURN
880 CLS:PRINT" LOADING DIRECTORY
890 A=0:NM=0
```

continued on page 30

OZZIE OS9

by Fred Bisseling

BACK ONCE AGAIN TO annoy the hell out of you, well... I'll try anyway. This month should have been the operational description of Ian Lobley's program; sorry, you are going to have to wait a little longer for that.

Instead you are getting a little info about OS9 Level II. Today is Tuesday, and having only received OS9 Level II on Saturday afternoon. This is my first impression of my new "toy".

First the documentation is a big improvement over previous versions. It goes into greater detail, which will help the beginner immensely.

My version includes what appears to be a modified and updated version of BASIC09. More about that in my next article.

Although there are some new COMMANDS in Level II, on the debit side, there are several that were in Level I that are not included. These are:- ASM, BINEX, DEBUG, TSMON, SLEEP, PRINTERR, SAVE, SLEEP, TEE and VERIFY.

To most of us, the loss of these COMMANDS is a real nuisance. It means the copying of all these from Level I Version 2 across to the new system disk. Be wary of using Level I VERSION 2 commands of the same name as level II commands... there are differences.

So far I have not encountered any problems using any of the Level I Ver.2 commands (NOT available on level 2) on the new system disk.

The SYS directory now holds the ERROR messages and some files required for the WINDOWS. The DEFs directory doesn't exist.

Level II has provision for double sided 40 track drives and single sided 80 track drives.

When building a system disk using the CONFIGuration programme provided, be sure that

you select a DEFAULT drive as well as D0 and D1 (if you have it). When the system has an ERROR it looks for the DEFAULT drive. Level II also allows for a HARD drive, faster access and greater storage space.

I have a 5meg. hard disk purchased from Blaxland Computers, and had little problem setting it up to run under level II. My biggest problem was trying to figure out why I was getting several COMMANDS in memory immediately on BOOT up.

After about two hours it was discovered that these COMMANDS were actually being loaded with SHELL. This can be overcome by DEleting the SHELL from the commands directory then SAVEing only those files which you wish to keep back into the commands directory under the same name, SHELL.

The end result, if you have two double sided 40 track drives, one hard disk and a 128k CoCo, you can still have 40K of memory free for your programs. A 512K upgrade would certainly make a big difference.

I have heard that STYLograph III with the Tandy Wordpak

Driver will work under Level II. My STYLO doesn't work with the PJB Wordpak or Opak drivers. No doubt someone will figure out a patch to get this great (my opinion, and I'm sticking to it) word processor to work.

I hope so, as I use STYLO on almost a daily basis. In the short time that I have had Level II, I'm most impressed and am looking forward to making more discoveries each time I use it, and I haven't even started on the WINDOWS yet.

Don't forget, you can write to me c/- P.O. Box 770 COOMA, NSW 2630 with any ideas or suggestions (polite ones only).

If you have any items of interest for the magazine, please don't hesitate to send them in.

Finally, I would like to congratulate Robert Kenny and his group in Coffs Harbour on starting a OS9 Users Group. I wish them all the best as they are all new to OS9. Hope to see a large contingent from Coffs at Conf '87 in Sydney in August.

Till next month or the next time on the OS9 Board on Goldlink.

■

WHERE IS IT?

continued from page 29

```
900 FORX=3TO11
910 B=1
920 DSKI$D,17,X,A$,B$
930 C$=A$+LEFT$(B$,127)
940 D$(A)=MID$(C$,B,8)
950 A=A+1:B=B+32
960 IFB>255 THEN980
970 GOTO940
980 IFX=11THEN990ELSENEXTX
990 FORC=0TO A-1
```

```
1000 IFLEFT$(D$(C),1)=CHR$(0)THE
N1040
1010 IFLEFT$(D$(C),1)=CHR$(255)T
HEN1040
1020 NM$(NM)=D$(C)
1030 NM=NM+1
1040 NEXT
1050 GOTO180
```

■

AUTOLOAD & PROTECT

32K ECB, tape only
UTILITY

by Bill Snow

I HAVE WRITTEN a utility program which I call "Protect". The object of the program is to give true password protection to a BASIC program on a CoCo 2 ECB (Mine is 64K), without disclosing the password to unauthorized users.

This has been achieved by utilizing an auto-start routine which prevents LISTING the program prior to being asked for the password, and disabling the print-to-screen function whilst inputting the password.

The program allows the user three attempts at the password after which it assumes that the user is not authorized and the program then self-destructs by way of a "cold-start" poke.

Instructions

First key in "Autoload" and CSAVE it. Then key in "Protect" and CSAVE it as a sub-routine to be used at the beginning of any BASIC program you wish to password protect.

To use, CLOAD "AUTOLOAD" and change the filename in line 15 to the name of the program you wish to protect. (For the purpose of demonstration, I changed it to "Protect".)

Now press <play> and <record> on your cassette and RUN it. The "Autoload" routines will then be saved onto your tape.

Next CLOAD your BASIC program with the "Protect" routine built-in and CSAVE it on the tape immediately after the "Autoload" routine. Rewind the tape and press <play> and type CLOADM:EXEC and the BASIC program will now auto-start and immediately call for the

password before allowing you to continue.

I have built-in the word "password" as the password. To change it merely change line 15 to include your own password when you first key-in "Protect".

Variables Used

'A': Variable for number of password attempts.
'X': Counts password attempts.
'P\$': Password input
'POKE359,255': Disables print-to-screen.
'POKE359,126': Restores print-to-screen.
'S': Counter for sound on successful input.

The lower-case in line 17 is important as this is a warning message and appears on the screen as reverse-video. I would suggest that line 11 be changed to include the title of your program. Lines 0-9 and 21-23 are for cosmetic and demonstration purposes only, and should be deleted when imbedding "Protect" into your own program.

I feel I should acknowledge "500 Peeks, Pokes 'n Exec's for the TRS-80 Colour Computer" by Kishmore M. Santwani for the valuable information obtained. "Autoload" is straight from the book but "Protect" is my own work with the aid of POKES obtained from the book.

The Listing:

```
0 GOTO10
1 '***** AUTOLOAD *****
  '***** BILL SNOW *****
3 SAVE"221:3":END'8
10 FORI=3584 TO 3642:READ A:POKE
  1,A:NEXT I:DATA 12,104,127,1,20
  9,189,166,76,158,25,48,31,111,12
  8,189,173,25,189,167,124,158,25,
  159,126,220,126,76
  15 DATA 189,172,55,189,167,11,16
  ,38,150,211,150,124,16,39,150,20
  5,42,233,159,27,189,167,233,189,
  173,33,189,172,239,126,173,158:C
  SAVEM"AUTOLOAD",3584,3643,3584
```

The Listing:

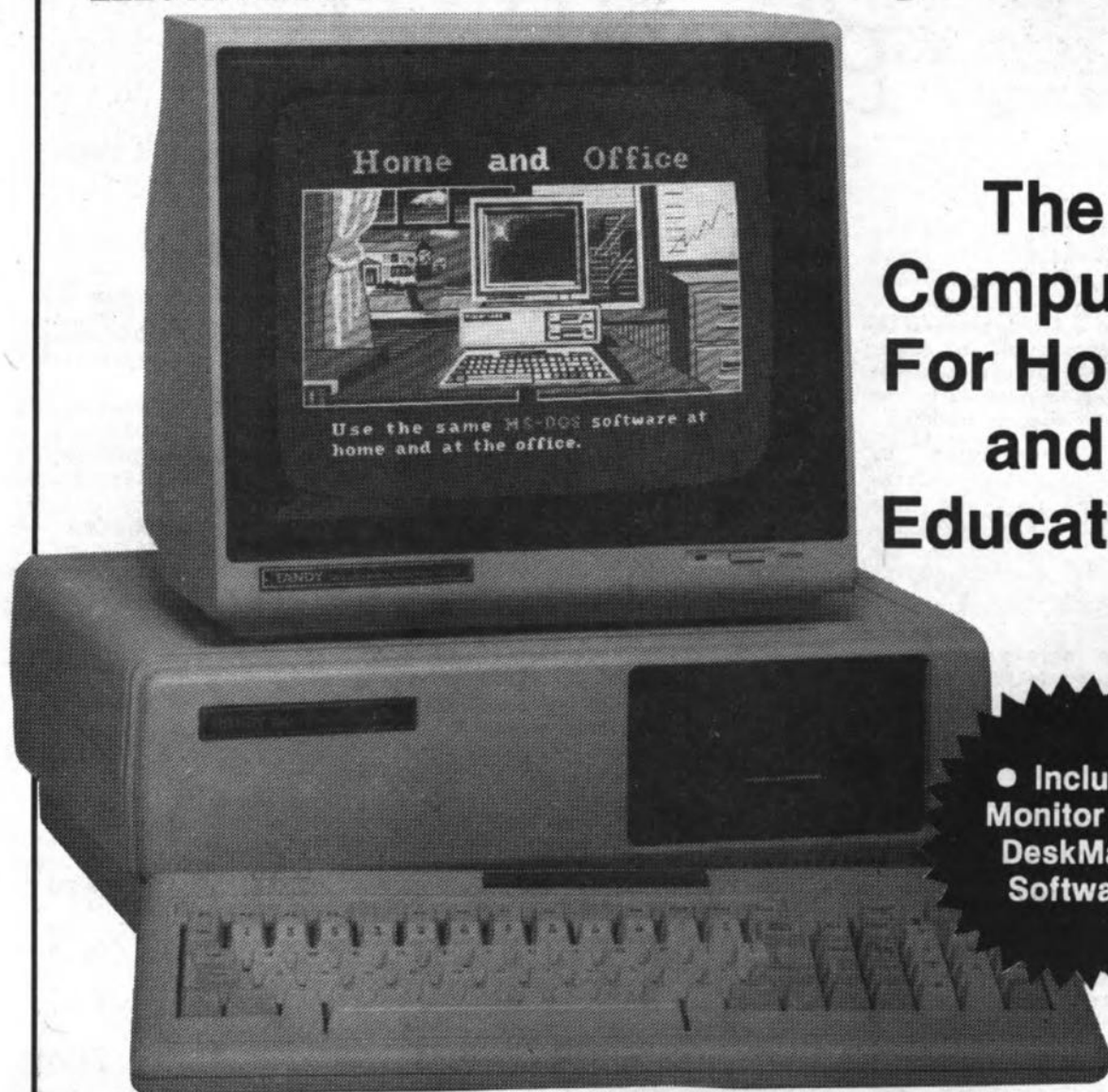
```
0 ' *****
1 ' * PASS-WORD *
2 ' * *
3 ' * PROTECT *
4 ' *****
5 ' APRIL, 1987
6 ' BY
7 ' W.G. SNOW
8 ' 6 BARLOW CLOSE,
9 ' PARKWOOD, 2322.
10 CLS
11 PRINT:PRINT" THIS IS A PROTEC
  TED PROGRAMME
12 PRINT" =====
  ====="
13 A=3:FOR X=1 TO A
14 SOUND 235,1:FOR S=1 TO 50:NEX
  T:SOUND 235,1
15 PRINT@258,"WHAT IS THE PASSWO
  RD PLEASE?":POKE 359,255:INPUT P
  $
16 IFP$="PASSWORD" THEN 20 ELSE
  POKE 359,126
17 IF X=A-1 THEN PRINT@165,"this
  is your last try":SOUND 150,5:F
  OR S=1TO1000:NEXT
18 IF X=A THEN POKE 113,0:EXEC 4
  0999
19 NEXT X
20 POKE 359,126:CLS
21 FOR S=200 TO 255 STEP 10:SOUN
  D S,1:NEXT
22 PRINT@170,"VERY GOOD":PRINT"
  YOU GOT PAST THE PASSWORD."
```



Tandy ELECTRONICS

The Tandy 1000

The Computer For Home and Education



• Includes
Monitor and
DeskMate
Software

Save \$1050

Reg
2348.95

\$1298

Rear View — External Connections



SPECIFICATIONS

Microprocessor: Intel 8088 processor. Clock speed, 4.77 MHz. Operating system: Includes Microsoft's advanced MS/DOS 2.11 with **BASIC Memory:** 128K RAM, expandable to 640K. **Keyboard:** 90-key sculptured, including numeric keypad. Twelve programmable Special Function keys. **Video Display:** Optional high-resolution, non-glare 30.48cm monochrome (green) or color monitor. 80 or 40 characters per line by 25 lines. High-resolution monochrome and color graphics (displaying 8 colors selected from 16). **Disk Drives:** One built-in double-sided, double density, 360K (formatted) thin-line 13-34cm mini-floppy, 48 tracks per inch. **Internal Expansion:** (1) Three user-accessible IBM PC-compatible 25cm card slots, second 360K Floppy Disk Drive. **External Connections:** (2) Standard parallel printer port. (3) Monochrome monitor. (4) Light pen adapter, two joysticks. (5) RGBI Color Monitor. (6) Composite video and audio. **Power:** 240 VAC, 50Hz.

This home and education computer package offers not only the exceptional capabilities of the Tandy 1000 IBM* compatible PC — it also includes DeskMate application software and the VM-Monochrome Monitor! With 128k RAM expandable to 640k and a clock speed of 4.7MHz. Phosphor green VDU displays 80 x 25 text and 640 x 200 graphics. 25-1000/26-3211

DeskMate Multi-Function Disk Software

Six excellent application on one disk: **Calendar** has a diary and reminder alarm. **Text** lets you edit and compose texts with ease. **Filer** compiles and retrieves data. **Worksheet** translates intricate calculations to solutions. **Mail** for intra-computer communications. **Telecom**; using an optional modem, the user can send/receive messages via phone.

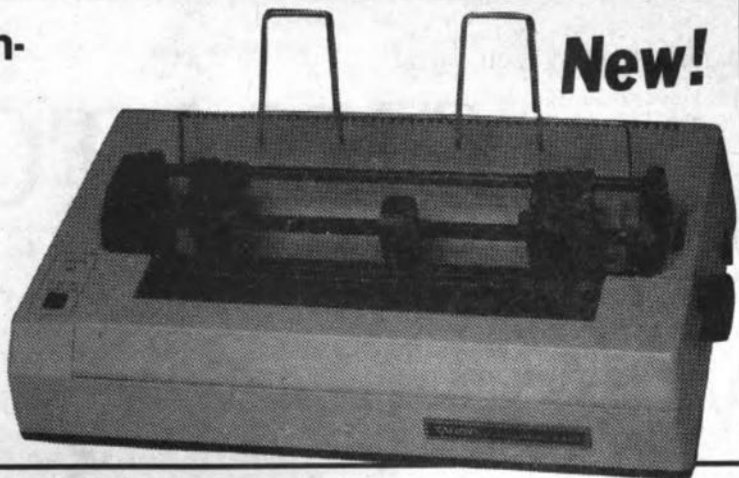
*IBM — Registered Trademark, International Business Machines Corp.

Tandy's Low-Priced High-Performance Printer!

New!

399⁹⁵

DMP 106. IBM Compatible! This Bi-directional printer is ideal for data processing and general-purpose use. Prints 80 upper and lower case characters per line (8 x 9 matrix) at 43 lines per minute at 10 characters per inch. Prints 80 characters per sec. 26-2802



Color Computer Accessories



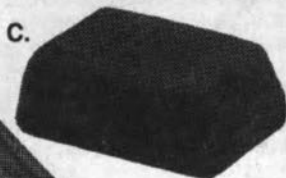
A. Deluxe Joystick
49⁹⁵ EACH

A. Our Best! For accurate, speedy cursor control. 26-3012



B. Printview Ruler
9⁹⁵

B. For enlarging print. With standard measurements. 26-1313



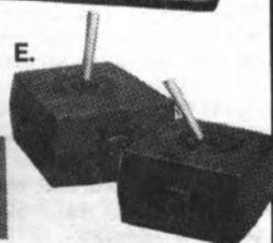
C. Color Mouse
79⁹⁵

C. Adds speed and ease to games and graphics creation. 26-3025



D. Flowchart
7⁹⁵

D. Template makes all your preproduction easier. 26-1312



E. Joysticks
34⁹⁵ PAIR

E. Budget priced joystick for fast 360° movement! 26-3008

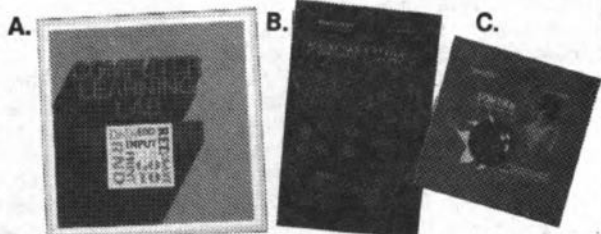
Tandy V2123



399⁹⁵

Enjoy a whole new world in communication! Our V21/23 modem allows access into Telecom's Viatel for a host of convenient services. Also links to bulletin boards and other videotex services. 300 bits per second or 1200 bits per second receive, 75 bits per second send, 240V AC at 5 watts. It's fully Telecom approved and is so easy to install! With excellent value like this you wouldn't want to miss out! 26-9404

Colour Computer Software



A. Donpan. Experience Donpan's return to Donpan Island. Avoid the crazy crows and saw sharks. If you successfully get "home" you'll become "King". Req. Joysticks, CoCo 2/3. 26-3097 **49.95**

B. Flight Simulator 1. Sit back and prepare yourself for some exhilarating high-tech. excitement! Learn how to take-off, land, the basics of manual control and more! Req. 64k disk drive CoCo 2/3 and joystick. 26-3108 **69.95**

C. Color Computer Learning Lab. An excellent self-teaching standard color basic programming course. Includes 8 lessons as well as a 200 page manual! Req. cassette recorder and CoCo 2/3. 26-3153 **99.95**

Tandy ELECTRONICS

A DIVISION OF TANDY AUSTRALIA LIMITED
NO. 11 N.S.W.

Nearly 350 Stores Australia-Wide

WE SERVICE WHAT WE SELL!

**Available From 350 Stores
Australiawide Including
Tandy Computer Centres
or Order On VIATEL *642614#**

Independent Tandy Dealers may not be participating in this ad or have every item advertised. Prices may also vary at individual Dealer Stores


```

;G2H1F2H1G2"
500 CIRCLE(30,40),12,5,2,0,.,5
510 DRAW"BM 31,45;D2F2R3E2U2L7R1
0":PAINT(33,47),,5:LINE(31,45)-(
19,42),PSET
520 DRAW"BM31,56;R3G2L4U1R2":DRA
W"BM17,40;R27U3H2L4G2U4H3L6G3D4H
2L4G2D3":DRAW"BM36,36;E3L1R2L1E1
L2R4L2E1L3R6L3E1L2R4L2E2"
530 FORA=64TO72STEP3:LINE(32,A)-
(34,A+1),PSET,BF:LINE(26,A)-(28,
A+1),PSET,BF:NEXT A:DRAW"BM44,60
;F3BL34E3"
540 GET(3,15)-(58,80),Z1,G:GET(8
3,15)-(138,80),Z2,G:GET(83,85)-(
138,150),Z3,G:GET(163,15)-(218,8
0),Z4,G:GET(3,85)-(58,150),Z5,G:
GET(163,85)-(218,150),Z6,G
550 PCLS:DRAW"BM100,100U4D4BD4D4
U4BU2BL2L4R4BR4R4":GET(92,94)-(1
09,109),Y,G
551 PCLS
555 GOTO5000
556 PRINT@228,"PLEASE WAIT A FEW
MOMENTS";:G$=C$(39)+C$(33)+C$(4
5)+C$(37)+C$(0)+C$(47)+C$(54)+C$(
37)+C$(50)
560 GOSUB2000
570 SCREEN1,1
575 A=48:B=63:C=1
580 DRAW"BM25,155;"+C$(45)+C$(37
)+C$(46)+C$(0):DRAW"BM55,155;"+C$(
16+ME):DRAW"BM130,155;"+C$(51)+C
$(35)+C$(47)+C$(50)+C$(37)
599 K=0:L=0:M=0:I=28:J=65:SP=0:R
N=RN+1:IFINT(RN/4)=RN/4THENCRCR
-2:VV=VV+1:IFVV>4THENDRAW"BM45,1
70;"+C$(38)+C$(41)+C$(50)+C$(51)
+C$(52)+C$(0)+C$(51)+C$(35)+C$(5
0)+C$(37)+C$(37)+C$(46):DRAWCS(0
)+C$(0):GOSUB1000:PRINT @ 295,"PRE
SS FIRE TO BEGIN":GOTO 1101
600 FOR D=1 TO 3
601 A(D)=RND(6):IFD>1THENP=0:IFD
=3ANDA(D-2)=A(D)ORA(D-1)=A(D)THE
N601
602 IF A(D)=1 THEN PUT(I,J)-(I+5
5,J+65),Z1,PSET:K=D
603 IF A(D)=2THENPUT(I,J)-(I+55,
J+65),Z2,PSET:L=D
604 IF A(D)=3THEN PUT(I,J)-(I+55
,J+65),Z3,PSET
605 IF A(D)=4 THEN PUT(I,J)-(I+5
5,J+65),Z4,PSET
606 IF A(D)=5THENPUT(I,J)-(I+55,
J+65),Z5,PSET
607 IF A(D)=6 THENPUT(I,J)-(I+55
,J+65),Z6,PSET:M=D
608 I=I+73
609 NEXT D
699 DRAW"BM 105,40;"+C$(50)+C$(4
7)+C$(53)+C$(46)+C$(36)+C$(0):LI
NE(152,34)-(159,41),PRESET,BF:DR
AW"BM 153,40;"+C$(17+VV)
740 H=JOYSTK(0):V=JOYSTK(1)
745 IFH=0 THEN LINE(A,B)-(A+17,B
+15),PRESET,BF:A=48:B=63:C=1
750 IF H=63 THEN LINE(A,B)-(A+17
,B+15),PRESET,BF:A=192:B=63:C=3
760 IF H>0 AND H<63 THEN LINE(A,
B)-(A+17,B+15),PRESET,BF:A=120:B
=63:C=2
800 IF C>0 THENPUT(A,B)-(A+17,B+

```

```

15),Y,PSET
809 P=PEEK(65280):IFP=126ORP=254
THEN 901
810 SP=SP+1:IF SP=>CR AND L>0 OR
SP=>CR AND M>0 OR SP=>CR ANDK>0
THEN 950
811 IF SP=>CR THEN L=0:M=0:K=0:G
OTO 599
900 GOTO 740
901 IF C=K THEN K=0:GOSUB 910:GO
TO 911
902 IF C=L THEN L=0:GOSUB 910:GO
TO 911
903 IF C=M THEN M=0:GOSUB 910:GO
TO 911
905 GOTO 920
910 U=A:S=B:O=18:P=35:Q=63:FOR R
=1 TO 27:LINE(U-O,S)-(U+P,S+Q),P
RESET,B:O=O-1:S=S+1:P=P-1:Q=Q-2:
PLAY"V31L255T25504G":NEXT R:FORT
=1TO460:NEXT T:SR=SR+10:SP=SP+1:
RETURN
911 LINE(180,149)-(220,160),PRES
ET,BF
912 PP$="":LL$=STR$(SR):AA=LEN(L
L$):FOR II=2 TO AA:BB$=MID$(LL$,
II,1):CC=VAL(BB$):PP$=PP$+C$(CC+
16):NEXT II:DRAW"BM 185,155;"+PP
$
913 IF SP=>CR THEN 810
914 GOTO 740
920 DRAW"BM 95,50;"+C$(33)+C$(33
)+C$(33)+C$(33)+C$(40)+C$(40)+C$(
40)+C$(40)+C$(40)+C$(0):SOUND 2
00,20:LINE(80,42)-(170,55),PRESE
T,BF:GOTO 950
950 PLAY"L4T1201V31BV28AV24GV20F
V16DV10C#V5C"
951 ME=ME-1:FOR Q=1 TO 3:IF K>0
THEN PUT(-45+(73*K),65)-(10+(73*
K),130),Z1,PRESET
952 IF L>0THEN PUT(-45+(73*L),65
)-(10+(73*L),130),Z2,PRESET
953 IF M>0 THEN PUT(-45+(73*M),6
5)-(10+(73*M),130),Z6,PRESET
954 NEXT Q
956 LINE(52,149)-(62,160),PRESET
,BF:DRAW"BM 55,155;"+C$(16+ME)
957 FOR TT=1 TO 460:NEXT TT
958 IF ME=0 THEN DRAW"BM 95,170"
+G$:FOR TT=1 TO 2000:NEXT TT:GOT
O 5000
999 GOTO 599
1000 REM SCREEN 2
1001 PLAY"V31T1001L4CL20004BA#AG
#GF#FBE-DCO3BA#AG#GF#FBE-DC#CO2B
A#AG#GF#FBE-DC#CO1BA"
1002 FOR TT=1 TO 3000:NEXT TT
1003 CLS:SCREEN 0,0:PRINT @ 227,
"PLEASE WAIT FOR SCREEN TWO";
1010 PCLS
1020 FOR A= 0 TO 140 STEP 8:FOR
B=0 TO 196 STEP 8:LINE(A,B)-(A+8
,B+4),PSET,B:LINE(A+4,B+4)-(A+12
,B+8),PSET,B:NEXT B,A:LINE(145,0
)-(156,196),PRESET,BF:LINE(145,0
)-(145,196),PSET
1030 POKE 178,2
1040 LINE(0,0)-(155,8),PSET,BF
1060 LINE(10,22)-(65,87),PRESET,
BF:LINE(81,22)-(136,87),PRESET,B
F:LINE(10,110)-(65,175),PRESET,B
F:LINE(81,110)-(136,175),PRESET,

```

```

BF
1070 POKE 178,3
1080 POKE 178,18:LINE(182,100)-(
256,196),PSET,BF:LINE(191,110)-(
246,175),PRESET,BF:POKE 178,3:LI
NE(219,75)-(165,105),PSET:LINE(2
19,75)-(256,96),PSET:LINE(182,10
0)-(260,196),PSET,B:POKE 178,18:
LINE(256,196)-(256,101),PSET:POK
E 178,3
1090 LINE(165,105)-(182,105),PSE
T:POKE 178,2:PAINT(175,103),,5
1095 POKE 178,3
1100 RETURN
1101 SOUND 100,1
1102 P=PEEK(65280):IF PE=126 OR
P=254 THEN 1103 ELSE 1102
1103 SCREEN 1,1:CR=12:C=0:SP=0:R
N=0:VV=0
1105 DRAW"BM 190,47;"+C$(45)+C$(
37)+C$(46):DRAW"BM 220,47;"+C$(M
E+16)
1106 DRAW"BM 168,32;"+C$(51)+C$(
35)+C$(47)+C$(50)+C$(37):DRAW "B
M 219,32"+PP$
1110 K=0:L=0:M=0:SP=0:RN=RN+1:IF
INT(RN/4)=RN/4 THEN CRCR-2:VV=
VV+1:IF VV=5 THEN DRAW"BM 162,60
;"+C$(48)+C$(50)+C$(37)+C$(48)+C
$(33)+C$(50)+C$(37):DRAW C$(0)+C
$(38)+C$(47)+C$(50):DRAW"BM147,7
2;"+C$(35)+C$(40)+C$(33)+C$(44)+
C$(44)+C$(37):GOTO 1170
1120 FOR D=1 TO 5
1121 A(D)=RND(6):IF D>1 THEN P=0
:IF A(D)=A(D-1) THEN 1121
1122 IF D>2 THEN P=0:IF A(D)=A(D
-2) THEN 1121
1123 IF D>3 THEN P=0:IF A(D)=A(D
-3) THEN 1121
1124 IF D>4 THEN P=0:IF A(D)=A(D
-4) THEN 1121
1125 IF D=5 THEN P=0:IF D=5 AND
RND(3)=2 THEN A=192:B=114:ELSE L
INE(192,113)-(247,179),PRESET,BF
:GOTO 1140
1130 IF D=1 THEN A=10:B=26
1131 IF D=2 THEN A=81:B=26
1132 IF D=3 THEN A=10:B=114
1133 IF D=4 THEN A=81:B=114
1134 IF A(D)=1 THEN PUT(A,B)-(A+
55,B+65),Z1,PSET:K=D
1135 IF A(D)=2 THEN PUT(A,B)-(A+
55,B+65),Z2,PSET:L=D
1136 IF A(D)=3 THEN PUT(A,B)-(A+
55,B+65),Z3,PSET
1137 IF A(D)=4 THEN PUT(A,B)-(A+
55,B+65),Z4,PSET
1138 IF A(D)=5 THEN PUT(A,B)-(A+
55,B+65),Z5,PSET
1139 IF A(D)=6 THEN PUT(A,B)-(A+
55,B+65),Z6,PSET:M=D
1140 NEXT D
1150 DRAW"BM 180,15;"+C$(50)+C$(
47)+C$(53)+C$(46)+C$(36):LINE(22
3,9)-(232,16),PRESET,BF:DRAW"BM
226,15;"+C$(VV+17)
1160 REM
1169 GOTO 1179
1170 DRAW C$(46)+C$(39)+C$(37)+
BR5"+C$(55)+C$(33)+C$(54)+C$(37)
:GOSUB 3000:GOTO 3100
1179 GOTO 1190

```

```

1180 LINE(210,25)-(256,37),PRESE
T,BF:PPS="":LL$=STR$(SR):AA=LEN(
LL$):FORII=2TOAA:BB$=MID$(LL$,II
,1):CC=VAL(BB$):PPS=PPS+C$(CC+16
):NEXTII:DRAW"BM 219,32"+PPS:GO
TO 1200
1190 REM
1199 A=30:B=111:C=3
1200 H=JOYSTK(0):V=JOYSTK(1)
1210 IF H=0 AND V>31 THEN LINE(A
,B)-(A+17,B+15),PRESET,BF:A=28:B
=112:C=3
1220 IF V>31 AND H<63 AND H>0 TH
EN LINE(A,B)-(A+17,B+15),PRESET,
BF:A=100:B=112:C=4
1230 IF H<31 AND V<31 THEN LINE(
A,B)-(A+17,B+15),PRESET,BF:A=28:
B=24:C=1
1240 IF H>31 AND V<31 THEN LINE(
A,B)-(A+17,B+15),PRESET,BF:A=100
:B=24:C=2
1300 IF H=63 AND V>31 THEN LINE(
A,B)-(A+17,B+15),PRESET,BF:A=210
:B=112:C=5
1305 PUT(A,B)-(A+17,B+15),Y,PSET
1310 P=PEEK(65280):IF P=126 OR P
=254 THEN 1902
1320 SP=SP+1:IF SP=>CR AND K>0
OR SP=>CR AND L>0 OR SP=>CR AND
M>0 THEN 1908
1330 IF SP=>CR THEN 1110
1901 GOTO 1200
1902 IF C=K THEN K=0:GOSUB910:GO
TO1180
1903 IF C=L THEN L=0:GOSUB910:GO
TO1180
1904 IF C=M THEN M=0:GOSUB910:GO
TO1180
1907 DRAW"BM172,70;" +C$(33)+C$(3
3)+C$(33)+C$(33)+C$(40)+C$(40)+C
$(40)+C$(40)+C$(40):SOUND 200,20
:LINE(160,62)-(250,74),PRESET,BF
1908 PLAY"L4T1201V31BV28AV24GV20
FV16DV10C#V5C"
1910 FOR D=1 TO 5
1912 IF D=1 THEN A=10:B=26
1914 IF D=2 THEN A=81:B=26
1915 IF D=3 THEN A=10:B=114
1916 IF D=4 THEN A=81:B=114
1918 IF D=5 THEN A=192:B=114
1920 IF K=D THEN PUT(A,B)-(A+55,
B+65),Z1,PRESET:K=0
1922 IF L=D THEN PUT(A,B)-(A+55,
B+65),Z2,PRESET:L=0
1924 IF M=D THEN PUT(A,B)-(A+55,
B+65),Z6,PRESET:M=0
1926 NEXT D
1929 ME=ME-1:LINE(217,41)-(226,5
2),PRESET,BF:DRAW"BM 220,47"+C$(
ME+16)
1930 IF ME=0 THEN DRAW"BM 170,70
;" +G$:FOR TT=1 TO 2000:NEXT TT:G
OTO 5000
1931 GOTO 1110
2000 SCREEN 0,0:PCLS:FOR TT=1 TO
460:NEXT TT
2010 POKE 178,3
2020 LINE(10,10)-(246,182),PSET,
B:LINE(15,15)-(241,177),PSET,B:P
AINT(11,11),,5
2030 LINE(27,61)-(84,131),PSET,B
:LINE(23,57)-(88,135),PSET,B:LIN
E(100,61)-(157,131),PSET,B:LINE

```

```

(96,57)-(161,135),PSET,B:LINE(17
3,61)-(230,131),PSET,B:LINE(169,
57)-(234,135),PSET,B
2035 SOUND 100,1:PRINT @ 295,"PR
ESS FIRE TO BEGIN";
2036 IF PEEK(65280)=254 OR PEEK(
65280)= 126 THEN PRINT @ 295,STR
ING$(32, " ");GOTO 2040: ELSE 20
36
2040 RETURN
3000 REM CHALLENGE LEVEL
3001 PLAY"V31T1001L4CL20004BA#AG
#GF#FEE-DC03BA#AG#GF#FEE-DC#C02B
A#AG#GF#FEE-DC#C01BA"
3002 FOR TT=1TO3000:NEXTTT:CLS:S
CREEN0,0:PRINT @ 225,"PLEASE WAI
T FOR CHALLENGE WAVE";
3010 PCLS
4010 POKE 178,10
4020 LINE(90,25)-(256,196),PSET,
BF:POKE 178,3
4030 LINE(90,100)-(256,120),PRES
ET,BF:FOR A=91 TO 256 STEP 6:LIN
E(A,100)-(A+2,120),PSET,BF:NEXTA
:LINE(87,100)-(256,97),PSET,BF:L
INE(91,120)-(256,118),PSET,BF
4040 LINE(110,29)-(165,95),PRESE
T,BF:LINE(185,29)-(240,95),PRESE
T,BF:LINE(110,122)-(165,188),PRE
SET,BF:LINE(185,122)-(240,188),P
RESET,BF
4050 POKE178,2:LINE(85,25)-(256,
17),PSET,BF:POKE178,3
4060 POKE 178,2:LINE(0,16)-(70,1
96),PSET,BF:POKE 178,3
4070 LINE(9,29)-(64,95),PRESET,B
F:LINE(9,122)-(64,188),PRESET,BF
4080 LINE(0,16)-(74,10),PSET,BF
4081 SOUND 100,1:PRINT @295,"PRE
SS FIRE TO BEGIN";
4082 IF PEEK(65280)=126 OR PEEK(
65280)=254 THEN 4085 ELSE 4082
4085 SCREEN 1,1
4090 DRAW"BM 100,10;C5"+C$(45)+C
$(37)+C$(46):DRAW"BM 130,10"+C$(
16+ME)
4095 DRAW"BM 160,10"+C$(51)+C$(3
5)+C$(47)+C$(50)+C$(37):DRAW"BM
205,10"+PPS
4100 BG=0:SP=0:IF RND(2)=1 THEN
D=124 ELSE D=31
4110 LL=RND(3):IF LL=1 THEN C=9
4120 IF LL=2 THEN C=110
4130 IF LL=3 THEN C=185
4140 LL=RND(3):IF LL=1 THEN PUT(
C,D)-(C+55,D+65),Z1,PSET:BG=1
4150 IF LL=2 THEN PUT(C,D)-(C+55
,D+65),Z2,PSET:BG=2
4160 IF LL=3 THEN PUT(C,D)-(C+55
,D+65),Z6,PSET:BG=3
4200 H=JOYSTK(0):V=JOYSTK(1)
4210 IF H>60 AND V>60 THEN LINE(
A,B)-(A+17,B+15),PRESET,BF:A=204
:B=122
4220 IF H<63 AND H>5 AND V>31 TH
EN LINE(A,B)-(A+17,B+15),PRESET,
BF:A=129:B=122
4230 IF H<5 AND V>31 THEN LINE(A
,B)-(A+17,B+15),PRESET,BF:A=28:B
=122
4240 IF H<5 AND V<31 THEN LINE(A
,B)-(A+17,B+15),PRESET,BF:A=28:B
=29

```

```

4250 IF H>5 AND H<60 AND V<31 TH
EN LINE(A,B)-(A+17,B+15),PRESET,
BF:A=129:B=29
4260 IF H>60 AND V<31 THEN LINE(
A,B)-(A+17,B+15),PRESET,BF:A=204
:B=29
4265 SP=SP+1:IF SP=5 THEN 4310
4267 PUT(A,B)-(A+17,B+15),Y,PSET
4270 P=PEEK(65280)
4280 IF P=126 OR P=254 THEN JJ=0
:IF A=19=C AND D=B+2 THEN GOSUB
910:SR=SR+10:GOTO 4500:ELSE 4310
4300 GOTO 4200
4310 PLAY"L4T1201V31BV28AV24GV20
FV16DV10C#V5C"
4315 IF BG=1 THENPUT(C,D)-(C+55,
D+65),Z1,PRESET
4320 IF BG=2 THEN PUT(C,D)-(C+55
,D+65),Z2,PRESET
4330 IF BG=3 THEN PUT(C,D)-(C+55
,D+65),Z6,PRESET
4340 ME=ME-1:DRAW"BM 100,10;C5"+
C$(45)+C$(37)+C$(46):LINE(127,4)
-(136,11),PRESET,BF:DRAW"BM130,1
0"+C$(16+ME)
4341 IF ME=0 THEN FOR TT=1 TO 30
00:NEXT TT:GOTO 5000
4345 FOR TT=1 TO 1000:NEXT TT
4350 LINE(C,D)-(C+55,D+65),PRESE
T,BF
4400 GOTO 4100
4500 PPS="":LL$=STR$(SR):AA=LEN(
LL$):FOR II=2 TO AA:BB$=MID$(LL$,
II,1):CC=VAL(BB$):PPS=PPS+C$(CC
+16):NEXT II:LINE(200,4)-(254,15
),PRESET,BF:DRAW"BM 205,10"+PPS:
GOTO 4100
5000 CR=12:C=0:SP=0:RN=0:VV=0:ME
=3:C$(0)="COU6RD6RU6RD6RU6RD6BR3
C1":C$(7)="BR2BU4U2RD2BD4BR5":C$(
10)="BR2U6D3NH2NG2NE2F2BDBR4":C
$(11)="BU3R5L3ND2U2RD4BDBR5":C$(
13)="BU3R5BD3BR3":C$(14)="URDBR7
":C$(15)="E5BD5BR3":C$(16)="BUU4
NF4ER3FD4GL3BR7"
5005 IF SR>0 THEN CLS:SR=0
5010 C$(17)="BR3RU6NGD6RBR3":C$(
18)="BU5ER3FDG2L2GDR5BR3":C$(19)
="BU5ER3FDGNLFDGL3HDBR8":C$(20)
="BU6D3R4U3NRD3BR4":C$(21)="BU6
NR5D2R4FD2GL3HDBR8":C$(22)="BUU
4ER3FBD2BLNL3FDGL3BR7":C$(23)="B
U6R5DG4DBR7":C$(24)="BUUHUR3FD
GNL2FDGL3BR7"
5020 C$(25)="BUFR3EU4HL3GDFR4BD3
BR3":C$(29)="BU2R5BU2L5BD4BR8":C
$(33)="U5ER3FD2NL4D3BR3":C$(34)
="U6R5FDGNL3FDGL4BR8":C$(35)="BUU
4ER3FBD4GL3BR7":C$(36)="U6R3F2D2
G2L3BR8":C$(37)="U6NR5D3NR4D3R5B
R3":C$(38)="U3NR4U3R5BD6BR3"
5030 C$(39)="BUU4ER3FBD2NL2D2GL3
BR7":C$(40)="U6BR5D3NL5D3BR3":C$(
41)="BR2R2LU6LR2BD6BR4":C$(42)
="BU2DFR3EU5BD6BR3":C$(43)="U6BR5
G4BF3BR3":C$(44)="NU6R5BR3":C$(4
5)="U6F2RE2D6BR3":C$(46)="U6F5DU
6BD6BR3":C$(47)="BUU4ER3FD4GL3BR
7":C$(48)="U6R4FDGL4D3BR8
5040 C$(49)="BUU4ER3FD3GNHNFGL2B

```

continued on page 39



Seeing is Believing.

BIGTEXT

UTILITY

by Gordon Thurston

BIGTEXT IS A program to print various sized letters to the hi-res screen. It uses the HPRINT statement to print the text, and this is blown up by machine language to make the text.

All the standard text is available. Also, the screen can be cleared to any colour, the background colour can be selected, the foreground colour can be selected, the x and y multiples can be changed, the cursor can be moved at any time with the joystick.

The colour of the cursor is the same as the foreground, and the background colour is at the top left of the screen. The whole screen can be saved to disk, and a disk will hold the program plus up to 4 screens.

Ctrl-M gives the menu. Press Ctrl followed by x, the x multiple, then enter to change the size of letter on the x axis, the same for the y axis.

The rest of the Ctrl commands are similar. Try each one to see the results.

I have included the source code, fully commented for those interested. Basically, it takes each pixel of the HPRINTed character in the right top corner, and prints it the number of times in xmult.

It repeats each line according to the ymult. The screen save uses the memory management unit (mmu).

There are two sets of 8 task registers. The first from &HFFA0 to &HFFA7 determines the normal memory map, and the second from &HFFA8 to &HFFAF the second.

The two maps can be switched by changing the true bit 0 of &HFF91. BASIC normally sets it to 0, but to access the hires screen, it changes the second set of task registers, and by changing the true bit, can access the screens.

The microprocessor can only address 64K at any one time, so by placing a number in a task register, any bank of 8K can be

switched to that position.

Task register &HFFA0 corresponds to processor space &H0000 to &H1FFF (0 to 8191 decimal), &HFFA1 corresponds to &H2000 to &H3FFF (8192 to 16383 decimal) etc.

The number to put into the task register is the memory address divided by &H2000. For instance, &h0000 in the basic addressing space is actually &H70000 (notice the extra zero!), so &H70000 / &H2000=38, which is put into &HFFA0. This inserts 8K of memory into the bottom of the processor space, which is actually &H70000.

If the computer is in 40 or 80 column mode, the mmu should function alright, but in 32 columns, it may not. The reason being, that before the mmu will function, bit 7 of &HF90 must be cleared (not CoCo 1 and 2 compatible), and bit 6 must be to enable the mmu.

If that sounds a bit complicated, you're right. Luckily all that is required is the simple BASIC program included.

When I CLEAR200,16000 or some such low memory, the CoCo 3 will lock up and do strange things, otherwise I would have saved the screen in 2 parts instead of the 4 parts used.

This program was originally designed for use with a slow-scan program to send pictures over the air by amateur radio, but must have many other uses as well. I hope to do a screen dump in the near future.

Getting everything to work

First type in and save the BASIC program. Then, by using EDTASM (or somesuch assembler), type in the source listing and save the assembled version after the BASIC program. Also, save the source code (for future alterations and/or reference, etc).

The Listing:

```
0 GOTO5
1 '***** BIG TEXT *****
   ***** D. THURSTON *****
3 SAVE"220:3":END'8
5 POKE&HE6C6,18:POKE&HE6C7,18
10 CLEAR200,&H5FFF:WIDTH80:HBUFF
1,64:HBUFF2,6000:POKE&HFFD9,0:XM
=4:YM=5:X=8:Y=0:F=1:B=3
11 ONBRKGOTO1000
12 IPPEEK(&H7540)<>26THENLOADM"B
IGLETR"
15 JX=JOYSTK(0):LX=JX:JY=JOYSTK(
1):LY=JY
20 POKE&H7534,10:POKE&H7535,10
30 POKE&H753E,10:POKE&H753F,10
40 HSCREEN2
50 X=10:Y=0
60 HGET(0,0)-(8,8),1
70 POKE&H7534,XM:POKE&H7535,YM:PO
KE&H753E,X/2:POKE&H753F,Y
80 HGET(X,Y)-(X+8*XM,Y+8*YM),2
90 A$=INKEY$
100 IFAS<>""THENHPUT(X,Y)-(X+8*X
M,Y+8*YM),2:GOTO140
110 IFD=0THEND=10:HLINE(X,Y)-(X+
8*XM,Y+8*YM),PSET,B
120 D=D-1:IFD=5THENHPUT(X,Y)-(X+
8*XM,Y+8*YM),2
121 JX=JOYSTK(0):IFJX<>LX THENHP
UT(X,Y)-(X+8*XM,Y+8*YM),2:X=JX*5
:LX=JX:GOTO70
122 JY=JOYSTK(1):IFJY<>LY THENHP
UT(X,Y)-(X+8*XM,Y+8*YM),2:Y=JY*4
:LY=JY:GOTO70
130 GOTO90
140 IFAS=CHR$(8)THENX=X-8*XM:IFX
<0THENX=0:Y=Y-8*YM:X=320-8*XM:PO
KE&H753F,Y:POKE&H753E,X/2 ELSEPO
KE&H753E,X/2
141 IFAS=CHR$(94)THENF=F-1:IFF<0
THENF=0:HCOLORF,B:GOTO70 ELSEHCO
LORF,B:GOTO70
142 IFAS=CHR$(10)THENF=F+1:IFF>1
5THENF=15:HCOLORF,B:GOTO70ELSEHC
OLORF,B:GOTO70
150 IFAS=CHR$(13)THENX=0:Y=Y+8*Y
M:GOTO70
160 IFAS=CHR$(189)THEN250
170 IFAS=CHR$(12)THENHCLSB:GOTO7
0
180 IFY<0THENY=0
190 HPRINT(0,0),A$
200 EXEC&H7540
210 HPUT(0,0)-(8,8),1
220 IFAS=CHR$(8)THEN90
230 X=X+8*XM:IFX>319THENX=0:Y=Y+
8*YM:GOTO70:ELSEGOTO70
```

```

240 AS=INKEYS: IFAS="" THEN240ELSE
PRINTASC(A$): GOTO240
250 CS=INKEYS: IFCs="" THEN250
260 IFCs="X" THENINPUTXM
270 IFCs="Y" THENINPUTYM
280 IFCs="F" THENINPUTF: HCOLORF, B
290 IFCs="B" THENINPUTB: HCOLORB, F
: HLINE(0,0)-(8,8), PSET, BF: HCOLOR
F, B: GOTO60
291 IFCs="S" ORCS="L" THEN400
292 IFCs="M" THEN500
300 GOTO70
310 HPUT(X,Y)-(X+8*XM,Y+8*YM),2
320 IFBUTTON(0) THEN70
330 X=JOYSTK(0)*5: Y=JOYSTK(1)*4
340 HGET(X,Y)-(X+8*XM,Y+8*YM),2
350 HPUT(X,Y)-(X+8*XM,Y+8*YM),2,
NOT
360 GOTO310
400 POKE&HFFD8,0: HSCREEN(0): INPU
T"NAME OF PICTURE"; N$
405 HSCREEN2: POKE&HFFD8,0
410 FORB=0TO3
420 POKE&HFFA3,&H30+B' MOVE IN NE
W BANK OF MEM
430 IFCs="S" THENSAVENs+"/" +STRs
(B),&H6000,&H7FFF,&H6000 ELSELOA
DNs+"/" +STRs(B)
440 NEXT
450 POKE&HFFA3,&H3B: 'REPLACE NOR
MAL BANK
455 POKE&HFFD9,0
460 GOTO70
500 HSCREEN0: CLS6: ATTR3,2
501 LOCATE22,5
505 PRINT"***** BIG TEXT ****
*****";
506 LOCATE22,7: PRINT"*** BY GORD
ON THURSTON ***";
507 LOCATE22,9: PRINT"*****APRIL
1987*****";
508 LOCATE0,11
510 PRINT"CLEAR -CLEAR SCREEN TO
BACKGROUND COLOUR
530 PRINT"L ARO -BACKSPACE
540 PRINT"U ARO -CHANGE FOREGROU
ND COLOUR DOWN ONE
550 PRINT"D ARO -CHANGE FOREGROU
ND COLOUR UP ONE
555 ATTR3,7
560 PRINT"CNTRL -ENTER COMMAND M
ODE FOR FOLLOWING COMMANDS
570 PRINT"L -LOAD PIC FROM D
ISK
580 PRINT"S -SAVE PIC FROM D
ISK
590 PRINT"X -CHANGE X MULTIP
LIER
600 PRINT"Y -CHANGE Y MULTIP
LIER
610 PRINT"B -CHANGE BACKGROU
ND COLOUR (1 TO 15)
620 PRINT"F -CHANGE FOREGROU
ND COLOUR (1 TO 15)
621 PRINT"USE JOYSTICK TO MOVE T
O NEW LOCATION IF DESIRED"
625 ATTR5,3,B
630 PRINT"PRESS ANY KEY"
640 WS=INKEYS: IFWS="" THEN640ELSE
HSCREEN2: GOTO70
1000 ATTR2,2: CLS1: POKE&HFFD8,0: E
ND

```

The Listing:

00090	ORG	30000	
00100	SMHFLG RMB	1	,U HALF FLAG FOR SMALL LETTER
00110	BGHFLG RMB	1	1,U HALF FLAG FOR LARGE LETTER
00120	COLCNT RMB	1	2,U COUNTS SM LETTER COLUMNS
00130	ROWCNT RMB	1	3,U COUNTS SM LETTER ROWS
00140	XMULT FCB	5	4,U HOLDS MULTIPLIER FOR X AXIS
00150	YMULT FCB	5	5,U HOLDS MULTIPLIER FOR Y AXIS
00160	XCNT RMB	1	6,U COUNTS X MULTIPLES
00170	YCNT RMB	1	7,U COUNTS Y MULTIPLES
00180	EOL RMB	2	8,U MARKS END OF LINE
00190	SMPOS RMB	2	10,U HOLDS SMALL POSITION
00200	LGPOS RMB	2	12,U HOLDS LARGE POSITION
00201	SCNCOL FCB	50	14,U PUT SCREEN X POS/2 HERE
00202	SCNROW FCB	50	15,U PUT SCREEN Y POS HERE
00210	START ORCC	#\$50	MASK INTERRUPTS
00220	LDA	#\$3B	MEMORY BAND WITH M/L PROG
00230	STA	\$\$FA5	MOVE TO HIGHER MEM
00240	JMP	HIGHER+\$4000	
00250	HIGHER LEAU	SMHFLG,PCR	INIT POINTER
00260	LDA	#\$30	INIT LOWER MEM TO PROC SPACE
00270	LDX	\$\$FFA0	START OF TASK REGISTERS
00280	MVSCRN STA	,X+	
00290	INCA		
00300	CMPA	#\$35	TEST FOR END OF CHANGES
00310	BNE	MVSCRN	
00311	LDA	15,U	SCNROW GET SCREEN ROW
00312	ADDA	#1	ADD A ROW
00320	LDB	#160	NUMBER OF BYTES PER COL
00330	MUL		CALC ROW OF BIG LETTER
00331	STD	8,U	MARK END OF LINE
00340	SUBD	#160	SUB ONE ROW
00350	ADDB	14,U	ADD SCNCOL TO CALC START BYTE
00360	BCC	NOCARY	SKIP CARRY IF NONE
00370	ADDA	#1	ADD CARRY
00380	NOCARY STD	12,U	LGPOS SAVE POSITION OF FIRST BYTE
00390	LDA	#8	
00400	STA	3,U	INIT ROWCNT
00410	STA	2,U	INIT COLCNT
00420	LDA	4,U	GET XMULT
00430	STA	6,U	INIT XCNT
00440	LDA	5,U	GET YMULT
00450	STA	7,U	INIT YCNT
00455	CLR	10,U	INIT SMPOS
00456	CLR	11,U	SMPOS LSB
00460	CLR	,U	INIT SMHFLG
00470	CLR	1,U	INIT BGHFLG
00480	LDY	12,U	LOAD STARTING POSITION
00490	LDX	#0	POINT TO TOP LEFT OF SCREEN
00500	RUN CMPY	8,U	CHECK FOR END OF LINE
00510	BHS	VARTST	MISS PRINT TO SCREEN IF OFF OF RIG
00520	LDA	,X	GET TWO PIXELS FROM SMALL LETTER
00530	TST	,U	TEST SMALL HALF FLAG
00540	BNE	SNDLG	BRNCH IF R PIX
00550	TST	1,U	TEST LG HALF FLAG
00560	BNE	SHIFTR	BRANCH IF R LG PIX
00570	BRA	MASK1	BOTH LEFT - NO SHIFT
00580	SNDLG TST	1,U	TEST LG HALF FLAG
00590	BEQ	SHIFTL	BRANCH IF LEFT LG PIX
00595	BRA	MASK2	
00600	SHIFTR LSRA		
00610	LSRA		
00620	LSRA		
00630	LSRA		
00635	MASK2 ANDA	#\$0F	SHIFT AND MASK PIX

00640	LDB	,Y	GET 2 PIX OFF SCREEN
00650	ANDB	,\$F0	MASK OUT 2ND PIX
00660	STB	,Y	PUT IT BACK
00670	ORA	,Y	ADD NEW PIX
00680	BRA	STORE	
00690	SHIFTL	LSLA	
00700	LSLA		
00710	LSLA		
00720	LSLA		
00730	MASK1	ANDA	,\$F0
00740	LDB	,Y	GET 2 PIX
00750	ANDB	,\$F0	MASK OUT ONE
00760	STB	,Y	PUT OTHER BACK
00765	ORA	,Y	GET WHOLE BYTE
00770	STORE	STA	,Y
00780	VARTST	LDA	,\$1
00790	EORA	1,U	TOGGLE BIG HALF FLAG
00795	STA	1,U	SAVE FLAG
00800	BNE	NOINC	MISS INCREMENT
00810	LEAY	1,Y	INC Y REG.
00820	NOINC	DEC	6,U
00830	BNE	RUN	GO REPEAT PIXEL
00840	LDA	4,U	GET X MULTIPLIER
00850	STA	6,U	INIT XMULT AGAIN
00860	LDA	#1	
00870	EORA	,U	TOGGLE SMHFLAG
00875	STA	,U	SAVE FLAG
00880	BNE	NOINCS	DON'T INCREMENT
00890	LEAX	1,X	INC X REG
00900	NOINCS	DEC	2,U
00910	BNE	RUN	MORE COLUMNS
00920	LDA	#8	8 COLUMNS
00930	STA	2,U	INIT COL COUNT
00940	LDY	8,U	GET OLD EOL
00950	LEAY	160,Y	ADD A LINE
00960	STY	8,U	MARK NEW EOL
00970	LDY	12,U	GET START OF LAST BIG LINE
00980	LEAY	160,Y	ADDD ONE LINE
00985	STY	12,U	UPDATE LARGE POSITION
00990	LDX	10,U	INIT Y REG.
01000	DEC	7,U	DEC Y COUNT
01010	BNE	RUN	
01020	LDA	5,U	
01030	STA	7,U	INIT Y COUNT
01040	LEAX	160,X	ADD ONE ROW TO X REG
01045	STX	10,U	INC SMALL ROW
01050	DEC	3,U	DEC ROW COUNT
01060	LBNE	RUN	
01070	EXIT	LDA	,\$3B
01080	STA	,\$FA3	MOVE PROGRAM TO ORIGINAL PLACE
01090	JMP	DOWN	
01100	DOWN	LDA	,\$38
01110	LDX	,\$FFA0	
01120	NORM	STA	,X+
01130	INCA		PUT NORMAL MAP BACK
01140	CMPA	,\$3E	
01150	BNE	NORM	
01160	RTS		
01170	END	START	

continued from page 36

R7": C\$ (50) = "U6R4FDGL3RF3BR3": C\$ (51) = "BUFR3EUHL3HUR3FBD5BR3": C\$ (52) = "BR2U6L2R5L2D6BR5": C\$ (53) = "BU5BR5D5GL3BR7": C\$ (54) = "BU6D2BFD BFDUBBU6BR3": C\$ (55) = "U6E2RF2BU6BR3": C\$ (56) = "UE4RUBL5DRF4DBR3"

5050 C\$ (57) = "BU6DF2ND3RND3E2UBD6BR3": C\$ (58) = "BU6R5DG5R5BR3"

5055 PCLS

5060 TT\$ = C\$ (52) + C\$ (40) + C\$ (37) + C\$ (0) + C\$ (35) + C\$ (33) + C\$ (51) + C\$ (52)

5061 DRAW"BM 100,30;" + TT\$: DRAW"BM 101,31;" + TT\$

5070 T\$ = C\$ (51) + C\$ (40) + C\$ (47) + C\$ (47) + C\$ (52) + C\$ (0) + C\$ (47) + C\$ (53) + C\$ (52)

5080 DRAW"BM 65,15;S8;" + T\$: DRAW"BM 66,16;S8;" + T\$: DRAW"BM 67,17;S8;" + T\$

5090 PUT(22,35)-(77,100),Z1,PSET: PUT(99,35)-(154,100),Z2,PSET: PUT(176,35)-(231,100),Z6,PSET

5100 DRAW"BM 85,45;S4;" + C\$ (52) + C\$ (40) + C\$ (37) + C\$ (0) + C\$ (34) + C\$ (33) + C\$ (36) + C\$ (0) + C\$ (39) + C\$ (53) + C\$ (57) + C\$ (51)

5110 DRAW"BM 26,103;" + C\$ (48) + C\$ (41) + C\$ (50) + C\$ (33) + C\$ (52) + C\$ (37)

5120 DRAW"BM 110,103;" + C\$ (52) + C\$ (40) + C\$ (53) + C\$ (39)

5130 DRAW"BM 173,103;" + C\$ (39) + C\$ (33) + C\$ (46) + C\$ (39) + C\$ (51) + C\$ (52) + C\$ (37) + C\$ (50)

5140 PUT(22,110)-(77,175),Z3,PSET: PUT(99,110)-(154,175),Z4,PSET: PUT(176,110)-(231,175),Z5,PSET

5150 DRAW"BM 84,116;" + C\$ (52) + C\$ (40) + C\$ (37) + C\$ (0) + C\$ (39) + C\$ (47) + C\$ (47) + C\$ (36) + C\$ (0): DRAW C\$ (39) + C\$ (53) + C\$ (57) + C\$ (51)

5160 DRAW"BM 38,179;" + C\$ (35) + C\$ (47) + C\$ (48)

5170 DRAW"BM 107,179;" + C\$ (35) + C\$ (44) + C\$ (47) + C\$ (55) + C\$ (46)

5180 DRAW"BM 190,180;" + C\$ (39) + C\$ (41) + C\$ (50) + C\$ (44)

5200 DRAW"BM 50,188;" + C\$ (48) + C\$ (50) + C\$ (37) + C\$ (51) + C\$ (51) + C\$ (0) + C\$ (38) + C\$ (41) + C\$ (50) + C\$ (37): DRAW C\$ (0) + C\$ (52) + C\$ (47) + C\$ (0) + C\$ (34) + C\$ (37) + C\$ (39) + C\$ (41) + C\$ (46)

5209 CP\$ = C\$ (35) + C\$ (47) + C\$ (45) + C\$ (48) + C\$ (44) + C\$ (37) + C\$ (52) + C\$ (37)

5210 SCREEN 1,1

5220 FOR F=1 TO 3: FOR X=1 TO 10: P LAY" T255L25501V28CV29DV20EV10F#V 21G#V16GV31": P=PREEK(65280)

5225 IF P=254 OR P=126 THEN GOTO 556

5226 NEXT X,F

5230 FOR A=1 TO 300

5240 P=PREEK(65280)

5250 IF P=254 OR P=126 THEN GOTO 556

5260 NEXT A: GOTO 5220

MUSIC PERFORMANCE

32K ECB & 'Music +'

by Charlie Toth

CHARLIE IS A member of the Gold Coast Tandy Computer Users' Group.

He recently bought a tape to one of our meetings which was loaded with music he'd made for "Music+".

We like it so much, we asked him to contribute it to the magazine. He gladly accepted and the result is what you're reading here.

In this compilation of songs are included the following: the "Hallelujah Chorus", "The Entertainer", and the title song for the enhanced version of the "Music+" program.

Enjoy!

The Listing:

COL:	LEN	V1	V2	V3	V4
1:	8	D5	A4	F4#	D3
2:	8	D5	G4	0	E3
3:	8	D5	A4	0	F3#
4:	8	A5	F5#	0	D3
5:	8	B5	G5	D5	G3
6:	8	A5	F5#	D5	D3
7:	8	0	0	0	F3#
8:	8	0	0	0	A3
9:	8	D6	A5	F5#	D3
10:	8	D6	A5	F5#	E3
11:	8	D6	A5	F5#	F3#
12:	8	F5#	D5	A4	D3
13:	8	G5	C5#	G4	E3
14:	8	F5#	0	A4	D3
15:	8	0	0	0	0
16:	8	A5	D5	A4	F3#
17:	8	G5	C5#	A4	E3
18:	8	F5#	D5	A4	D3
19:	8	E5	D5	A4	A3
20:	8	E5	C5#	G4	A2
21:	8	D5	A4	F4#	D3
22:	8	A4	0	F4#	D3
23:	8	B4	0	G4	0
24:	8	C5#	0	B4	0
25:	4	D5	A4	F4#	D3
26:	8	A4	A4	D4	F3#
27:	8	B4	G4	D4	G3
28:	8	A4	F4#	D4	D3
29:	8	0	0	0	F3#
30:	8	0	0	0	A3
31:	4	D5	A4	F4#	D3
32:	8	A4	A4	D4	F3#
33:	8	B4	G4	D4	G3
34:	8	A4	F4#	D4	D3
35:	8	0	0	0	0

36:	16	D5	A4	D4	F3#
37:	16	D5	A4	D4	F3#
38:	8	D5	B4	G4	G3
39:	16	D5	A4	F4#	D3
40:	16	D5	A4	F4#	0
41:	8	D5	B4	G4	0
42:	16	D5	A4	D4	F3#
43:	16	D5	A4	D4	F3#
44:	8	D5	B4	G4	G3
45:	16	D5	A4	F4#	D3
46:	16	D5	A4	F4#	0
47:	8	D5	B4	G4	D4
48:	8	D5	A4	D4	F3#
49:	8	C5#	G4	E4	E3
50:	8	D5	F4#	A3	D3
51:	8	D5	E4	A3	A3
52:	8	C5#	E4	A3	A2
53:	8	D5	F4#	A3	D3
54:	8	A5	D5	A4	F3#
55:	8	G5	C5#	A4	E3
56:	8	F5#	D5	A4	D3
57:	4	E5	A4	C4#	A3
58:	8	A4	A4	E4	C4#
59:	8	F5#	A4	D4	D4
60:	8	E5	A4	C4#	A3
61:	8	0	0	0	C4#
62:	8	0	0	0	E4
63:	4	E5	A4	C4#	A3
64:	8	A4	A4	E4	C4#
65:	8	F5#	A4	D4	D4
66:	8	E5	A4	C4#	A3
67:	8	0	0	0	0
68:	16	E5	A4	E4	C4#
69:	16	E5	A4	E4	C4#
70:	8	F5#	A4	D4	D4
71:	16	E5	A4	C4#	A3
72:	16	E5	A4	C4#	A3
73:	8	0	F4#	D4	A3
74:	16	E5	A4	E4	C4#
75:	16	E5	A4	E4	C4#
76:	8	F5#	A4	D4	D4
77:	16	E5	A4	C4#	A3
78:	16	E5	A4	C4#	A3
79:	8	0	F4#	D4	A3
80:	8	E5	A4	E4	C4#
81:	8	F5#	A4	D4	D4
82:	8	E5	A4	E4	C4#
83:	8	D5	A4	F4#	B3
84:	8	D5	G4#	D4	B3
85:	8	C5#	A4	E4	A3
86:	16	0	A4	C4#	E3
87:	16	0	A4	C4#	E3
88:	8	0	A4	D4	F3#
89:	8	0	A4	C4#	E3
90:	4	0	0	0	0
91:	16	D6	D5	0	0
92:	16	E6	E5	0	0
93:	16	C6	C5	0	0

94:	8	A5	A4	0	0
95:	16	B5	B4	0	0
96:	8	G5	G4	0	0
97:	16	D5	D4	0	0
98:	16	E5	E4	0	0
99:	16	C5	C4	0	0
100:	8	A4	A3	0	0
101:	16	B4	B3	0	0
102:	8	G4	G3	0	0
103:	16	D4	D3	0	0
104:	16	E4	E3	0	0
105:	16	C4	C3	0	0
106:	8	A3	A2	0	0
107:	16	B3	B2	0	0
108:	16	A3	A2	0	0
109:	16	G3#	G2#	0	0
110:	8	G3	G2	0	0
111:	8	0	0	0	0
112:	8	G5	D5	B4	G2
113:	16	D4	B3	G3	0
114:	16	D4#	B3	G3	0
115:	16	E4	C3	0	0
116:	16	C5	C3	0	0
117:	16	C5	C4	G3	E3
118:	16	E4	C4	G3	E3
119:	16	C5	G3	G2	0
120:	16	C5	G3	G2	0
121:	16	B4	C4	A3#	G3
122:	16	C5	C4	A3#	G3
123:	8	C5	F3	F2	0
124:	8	C5	C4	A3	0
125:	16	C5	0	0	E3
126:	16	C6	E5	C5	E3
COL:	LEN	V1	V2	V3	V4
127:	16	D6	F5	C4	G3
128:	16	D6#	F5#	C4	G3
129:	16	E6	G5	E5	G2
130:	16	C6	E5	C5	G2
131:	16	D6	F5	C4	G3
132:	16	E6	G5	C4	G3
133:	16	E6	G5	E5	G2
134:	16	B5	D5	B4	G2
135:	8	D6	F5	B3	G3
136:	8	C6	E5	C5	C3
137:	8	C6	E5	C4	G3
138:	8	C6	E5	C4	G3
139:	4	0	0	0	0
140:	4	F4#	D4	A3	D3
141:	8	G4	D4	A3	D3
142:	8	F4#	D4	A3	D3
143:	8	E4	C4#	A3	A2
144:	8	D4	D4	A3	A2
145:	8	C4#	E4	A3	A2
146:	8	E4	C4#	A3	A2
147:	8	D4	D4	D3	0
148:	8	E4	D4	D3	0
149:	8	F4#	A3	D3	0
150:	8	G4	A3	D3	0
151:	4	A4	F3#	0	0

8x8 COLOUR PATTERN

APPLICATION

by Colin North

```

152: 8 ,B4 ,B3 ,G3 ,0
153: 8 ,B4 ,C4# ,G3 ,0
154: 8 ,A4 ,D4 ,F3# ,0
155: 8 ,F4# ,D4 ,F3# ,0
156: 8 ,G4 ,E4 ,E3 ,0
157: 8 ,E4 ,C4# ,E3 ,0
158: 8 ,F4# ,A4 ,D4 ,D3
159: 8 ,D4 ,A4 ,D4 ,D3
160: 8 ,E4 ,A4 ,C4# ,A3
161: 8 ,C4# ,G4 ,C4# ,A3
162: 8 ,D4 ,F4# ,A3 ,D3
163: 16 ,C4# ,F4# ,A3 ,D3
164: 16 ,B3 ,F4# ,A3 ,D3
165: 16 ,A3 ,D4 ,0 ,D3
166: 16 ,G3 ,D4 ,0 ,D3
167: 16 ,F3# ,D4 ,0 ,D3
168: 16 ,E3 ,D4 ,0 ,D3
169: 8 ,D3 ,0 ,A3 ,D3
170: 8 ,A3 ,C4# ,0 ,0
171: 8 ,B3 ,D4 ,0 ,0
172: 8 ,C4# ,E4 ,0 ,0
173: 8 ,F4# ,D4 ,F3# ,0
174: 16 ,E4 ,D4 ,F3# ,0
175: 16 ,F4# ,D4 ,F3# ,0
176: 8 ,G4 ,D4 ,D3 ,0
177: 8 ,F4# ,D4 ,D3 ,0
178: 8 ,E4 ,E4 ,A3 ,C3#
179: 16 ,A4 ,E4 ,A3 ,C3#
180: 16 ,G4 ,E4 ,A3 ,C3#
181: 16 ,F4# ,E4 ,A3 ,A2
182: 16 ,E4 ,E4 ,A3 ,A2
183: 16 ,D4 ,E4 ,A3 ,A2
184: 16 ,C4# ,E4 ,A3 ,A2
185: 8 ,D4 ,E4 ,F3# ,B2
186: 16 ,C4# ,E4 ,F3# ,B2
187: 16 ,D4 ,E4 ,F3# ,B2
188: 16 ,E4 ,B3 ,G2 ,0
189: 16 ,D4 ,B3 ,G2 ,0

```

```

COL: LEN ,V1 ,V2 ,V3 ,V4
190: 16 ,C4# ,B3 ,G2 ,0
191: 16 ,B3 ,B3 ,G2 ,0
192: 8 ,C4# ,C4# ,F3# ,F2#
193: 16 ,B3 ,C4# ,F3# ,F2#
194: 16 ,C4# ,C4# ,F3# ,F2#
195: 8 ,D4 ,D4 ,F3# ,D2
196: 8 ,C4# ,C4# ,F3# ,D2
197: 8 ,B3 ,B3 ,G3# ,E2
198: 16 ,E4 ,B3 ,E3 ,E2
199: 16 ,A4 ,B3 ,E3 ,E2
200: 16 ,G4 ,E4 ,0 ,E2
201: 16 ,F4# ,E4 ,0 ,E2
202: 16 ,E4 ,D4 ,0 ,E2
203: 16 ,D4 ,D4 ,0 ,E2
204: 16 ,C4# ,C4# ,F3# ,F3#
205: 16 ,B3 ,C4# ,F3# ,F3#
206: 16 ,A3# ,B3 ,F3# ,F2#
207: 16 ,G3# ,B3 ,F3# ,F2#
208: 16 ,A3# ,C4# ,F3# ,F2#
209: 16 ,B3 ,C4# ,F3# ,F2#
210: 16 ,C4# ,C4# ,F3# ,F2#
211: 16 ,A3# ,C4# ,F3# ,F2#
212: 96 ,A3# ,C4# ,0 ,0
213: 8 ,B3 ,B3 ,F3# ,E2
214: 8 ,F4# ,B3 ,F3# ,B2
215: 96 ,F4# ,B3 ,F3# ,0
216: 8 ,G4# ,B3 ,F3# ,B2
217: 8 ,A4# ,B3 ,F3# ,B2
218: 96 ,A4# ,B3 ,F3# ,0
219: 2 ,B4 ,0 ,F3# ,B2

```

HERE IS A small application I use to work out what the different 64 colours of the CoCo 3 look like.

The objective is to then fill out the colour table on page 295 of your manual and refer to it whenever you might want to use a certain colour.

120 PRINT" *PRESS ANY KEY TO CONTINUE*";

```

130 EXEC44539
140 HSCREEN2:HCOLOR8,4:HCLS
150 FORX=0TO320STEP80
160 HLINE(X,0)-(X,192),PSET:NEXT
170 HLINE(0,96)-(320,96),PSET
180 FORX=0TO7:READA(X):NEXT
190 X1=2:Y1=2
200 FORX=0TO7
210 HPAINT(X1,Y1),X,8
220 PALETTEX,A(X)
230 IFN>39 THENC=8
240 HCOLORC
250 IF N1>35 THEN N1=2:N2=13
260 IFX=0THEN N1=2:N2=2
270 HPRINT(N1,N2),N
280 N1=N1+10:N=N+1
290 X1=X1+80
300 IFX1>282THENX1=2:Y1=100
310 C=11
320 NEXT
330 FORX=1TO3000:NEXT
340 PALETTERGB
350 IFA(7)=63 THENRESTORE:C=11:N
=0':GOTO70
360 GOTO140
370 DATA 0,1,2,3,4,5,6,7,8,9,10,
11,12,13,14,15
380 DATA 16,17,18,19,20,21,22,23
4,35,36,37,38,39,40,41,42,43,44,
45,46,47,48,49,50
390 DATA 51,52,53,54,55,56,57,58
,59,60,61,62,63
400 RESTORE:CLS5:GOTO70

```

The Listing:

```

0 GOTO10
1 ***** 8*8 COLOUR *****
***** COLIN NORTH *****
3 SAVE"165:3":END'10
4 SAVE"8*8COLOR":END
10 ONBRKGOTO400
20 POKE65497,0
30 '64 COLOR CHECKER
40 'BY COLIN NORTH
50 '17 SEPTEMBER 1986
60 DIMA(7)
70 PALETTERGB
80 WIDTH40:CLS5:ATTR2,4'
90 N1=2:N2=2:N=0:C=11
100 LOCATE3,8:PRINT"THIS PROGRAM
WILL ASSIST YOU TO":PRINT:PRINT
" DETERMINE THE 64 COLORS AVAI
LABLE TO":PRINT:PRINT" YOU,AND
NOTE THEM ON PAGE 295 OF YOUR":
PRINT" MANUAL,"
110 LOCATE6,22:ATTR2,1,B,U

```

APOLOGY

Over the past few months Val Stephen has sent to us a few programs and the such for inclusion in the magazine.

But due to a typing error his name has been written as "Wal Stephenson" and "Val Stephenson".

Many apologies for this oversight.

PRINTER 2

CoCo 32K + Siemens Teleprinter
UTILITY

by Frank Rees

WHILE THE COCO AND MC10 are inexpensive computers, the same cannot be said for suitable printers to use with them. A suitable purchase can be difficult as all have limitations of one form or another and some, in addition to a high price tag, are expensive to run because they need special ribbons or paper.

Teleprinters are a low cost, high quality alternative. Last year I did a disassembled to paper, 12K of the ROM in a VZ300 computer and this year I intend to do the same with the 16K of my CoCo 2 when the disassembler program is finished.

The printer I use is a Siemens 100 teleprinter and for this purpose cannot be beaten. It has clear, non matrix print, inexpensive roll paper and uses the cheapest typewriter ribbons.

These things making the result both good and inexpensive.

In 10 years of computer hobbying, I have used a teleprinter most of the time and found it to be an indispensable tool when programming, to print out

programs or program sections to study away from screen where I can make corrections.

While the teleprinter has its own limits, all capitals and only a short character set, it has not affected its usability to any degree of concern for my purposes.

Thus I continue with each computer to do the work necessary to interface and write a printer program, and have done so for many popular makes. The most recent was "PRINTER 2" for the CoCo. This was, as much as possible, a word for word translation of the MC10 version.

The MC10 "PRINTER 2" was hand assembled using 6803 which, like 6800, lends itself to easy assembly which, with experience, can be written directly into code.

I decided to do a translation that would give a good knowledge of 6803 to 6809 conversion. The 6809, while very sophisticated, lacked many of the straight forward instructions of the 6800-6803 and some in fact are difficult without assembly and a cross reference in a code book.

The long branch and program

counter relative instructions of the 6809, while very nice, had not been needed in my hobbyist programs. An opinion was formed that the 6809 was not really the hobbyist's ideal language.

The translation of 6803 to 6809 was done with no effort to alter the program in any way. Even my method of program counter relative for the 6803 was translated as was except for the initialization program which was so completely different.

Some NOP's used to add any instructions that were left in the CoCo version. The teleprinter can do a separate carriage return or line feed.

The NOP's allow such things to be added as needed.

The comments in the CoCo source code will give a pretty full story of how it will all work.

Letters with any questions shall be answered promptly. Send your letter to ...

Frank Rees,
27 King St
Boort 3537

The Listing:

00100 *****	00340	LEAX	START,PCR
00110 *	00350	STB	\$0167 ; JUMP TO
00120 * PRINTER2 COCO TO SIEMENS 100	00360	STX	\$0168 ; START DIVERT
00130 *	00370	LEAX	INT2,PCR ; INT1 ONCE ONLY
00140 *	00380	STX	EXEC ; TO RE-INT 2
00150 *	00390	INT2 BSR	GETA1
00160 *****	00400	GETA1	PULS
00170 ORG \$7000	00410	FIGS CLR	DIF1,X ; CLEAR FLAG -FIGS
00180	00420	LDA	#68 ; CHAR. PER LINE
00190 SPACE EQU \$00 ; BAUDOT SPACE	00430	STA	MAX ; POKE ORG. + 29H
00200 MARK EQU \$02 ; BAUDOT MARK	00440	LDA	#\$76 ; ITY FIGS
00210 COUNT EQU \$009C ; LINE POSITION X	00450	LBSR	BAUCOUT ; PRINT IT
00220 MAX EQU \$009B ; LINE LENGTH	00460	LDA	#\$0D ; ASC 'CR'
00230 PPORT EQU \$FF20 ; RS232 OUT	00470	BSR	PRINT2
00240 DELAY EQU \$0900 ; 20 MS - 50 BAUD	00480	RIS	;
00250 DIF1 EQU GETA1-PRINT2 ; CLR FLAG INT	00490	START LEAS	+2,S ; A285 FROM STACK
00260 DIF2 EQU FLAG-REF ; TO CHANGE FLAG	00500	DIVERT BSR	STORE1 ; A282 JSR 0167
00270 DIF3 EQU DIF2+1 ; PLUS REF =TABLE	00510	PSHS	B ; A285 SAVE B
00280 EXHC EQU \$009D	00520	LDB	\$6F ; A287 DEVICE FLAG
00290 INT1 LDB \$0167 ; GET VECTOR	00530	INCB	;
00300 LDX \$0168 ; CONTENTS	00540	PULS	B ; A28A RESTORE B
00310 STB STORE1,PCR ; THE NEW	00550	BMI	PRINT1 ; A28C -2+1=-1
00320 STX STORE2,PCR ; O/P VECTOR	00560	BNE	VIDEO ; A28E 0+1=+1
00330 LDB #\$7E ; JUMP OP. CODE	00570	JMP	\$A290 ; A290 -1+1=0 CASS
	00580	VIDEO BRN	PRINT1 ; P AND V VERSION

00590	JMP	\$A30A	; VIDEO	01370	PSHS	X	; SAVE COPY OF X
00600	PRINT1	BRA	PRINT2 ; A2BF FROM PRINTR	01380	LDX	#DELAY	; DELAY COUNT 20MS
00610	STORE1	RMB	1 ; 0167 PREFORM AS	01390	TAKE1	LEAX	-1,X ; -1 FROM COUNT
00620	STORE2	RMB	2 ; 0168-0169	01400		BNE	TAKE1 ; CONT. COUNT DOWN
00630	PRINT2	PSHS	X,B,A ; PRINTER2 START	01410		PULS	X ; RESTORE X VALUE
00640	PRINTR	TFR	A,B ; COPY A TO B	01420		TSTA	; IF ZERO ALL GONE
00650		TFR	CC,A ; COPY CC TO A	01430		BNE	OUTBC ; MORE BITS
00660		PSHS	A ; COPY A TO STACK	01440		RTS	
00670		ORCC	#\$50 ; SE1	01450	PCBQ	BSR REF	; TO GET PC REF
00680		TER	B,A ; COPY B TO A	01460	REF	PULS	X ; X=THIS ADDRESS
00690		NOP		01470		RTS	
00700		NOP		01480	FLAG	FCB	\$00
00710		NOP		01490	*	TABLE	ASC TO BAUDOT
00720		NOP		01500	TABLE	FCB	\$48 ; SPACE
00730	TSTCR	CMPA	#\$0D ; ASC 'CR'	01510		FCB	\$78 ; . EXCLAMATION
00740		BNE	NEXT	01520		FCB	\$4A ; ' DOUBLE QUOTE
00750		BSR	CROUT ; CR,LF OUT	01530		FCB	\$68 ; # ALTER. POUND
00760		BRA	FINISH	01540		FCB	\$74 ; \$
00770	NEXT	CMPA	#\$0C ; ASC FORM FEED	01550		FCB	\$5A ; %
00780		BNE	CONT	01560		FCB	\$4C ; 8 AND SIGN
00790		LDB	#10 ; COUNT 10 'LF'	01570		FCB	\$4A ; '
00800	MORE	PSHS	B ; SAVE ON STACK S	01580		FCB	\$5E ; (
00810		BSR	TTYLF ; OUT LINE FEED	01590		FCB	\$64 ;)
00820		PULS	B ; GET COUNT	01600		FCB	\$FA ; X ASTERICK
00830		DECB	; COUNT DOWN	01610		FCB	\$62 ; +
00840		BNE	MORE ; TILL ALL GONE	01620		FCB	\$58 ; .
00850		BRA	FINISH	01630		FCB	\$46 ; -
00860	CONT	SUBA	#\$20 ; REMOVE CTL CODE	01640		FCB	\$7B ; .
00870		BCC	HERE ; 20 AND OVER	01650		FCB	\$7A ; /
00880		BRA	FINISH ; AS CTL CODE DONE	01660		FCB	\$6C ; 0
00890	HERE	CMPA	#\$40 ; LOWER CASE?	01670		FCB	\$6B ; 1
00900		BCC	CONT ; IF=>40	01680		FCB	\$66 ; 2
00910	*		; A=CHAR=00-3F	01690		FCB	\$42 ; 3
00920	BAUDOT	BSR	PCEQ ; GET REF ADDRESS	01700		FCB	\$54 ; 4
00930		PSHS	X ; SAVE COPY OF X	01710		FCB	\$60 ; 5
00940		TFR	A,B ; COPY A-B A=00-3F	01720		FCB	\$6A ; 6
00950	*		; CHARACTER CODE	01730		FCB	\$4E ; 7
00960		ABX	; X=B+X B=TABLE OS	01740		FCB	\$4C ; 8
00970		LDB	DIF3,X ; X=REF+DIF2+CHAR	01750		FCB	\$70 ; 9
00980	*		; CODE 00-3F=TABLE	01760		FCB	\$5C ; :
00990		PULS	X ; GET ADDRESS REF	01770		FCB	\$5C ; : SEMICOLON
01000	LFTST	PSHS	B ; B=CHAR IN BAUD+	01780		FCB	\$5E ; (LESS THAN
01010		LDA	DIF2,X ; A BIT7=FLAG	01790		FCB	\$7C ; =
01020		ANDB	#\$80 ; B=BIT7 CHAR	01800		FCB	\$64 ;) GREATER THAN
01030		PSHS	B	01810		FCB	\$72 ; ?
01040		CMPA	,S+ ; COMPARE A W/B	01820		FCB	\$56 ; @ ANPERSAND
01050		BEQ	PRCHAR ; IF FLAG SAME	01830		FCB	\$C6 ; A
01060		STB	DIF2,X ; CHANGE FLAG	01840		FCB	\$F2 ; B
01070		LDA	#\$7E ; LETTERS	01850		FCB	\$DC ; C
01080		LDB	DIF2,X ; IF FLAG=LTRS=80	01860		FCB	\$D2 ; D
01090		BNE	LINK ; PRINT LETTERS	01870		FCB	\$C2 ; E
01100		LDA	#\$76 ; ELSE FIGURES	01880		FCB	\$DA ; F
01110	LINK	BSR	BAUOUT ; PRINT IT	01890		FCB	\$F4 ; G
01120	PRCHAR	PULS	A ; GET B TO A	01900		FCB	\$E8 ; H
01130	BPFIN	BSR	BAUOUT ; PRINT IT	01910		FCB	\$0CC ; I
01140	FINISH	PULS	A	01920		FCB	\$D6 ; J
01150		TFR	A,CC ; A TO CCR	01930		FCB	\$DE ; K
01160		PULS	A ; A =CHAR	01940		FCB	\$E4 ; L
01170		CMPA	#\$0D ; ASC WAS IT CR.?	01950		FCB	\$F8 ; M
01180		BEQ	CLRC ; TO CLEAR L/COUNT	01960		FCB	\$D8 ; N
01190		INC	COUNT ; ADD 1 TO L/COUNT	01970		FCB	\$F0 ; O
01200		LDB	COUNT ; GET COUNT	01980		FCB	\$EC ; P
01210		CMPB	MAX ; COUNT=MAX?	01990		FCB	\$EE ; Q
01220		BCS	RET ; NO	02000		FCB	\$D4 ; R
01230		BSR	CROUT ; OUT CR,LF	02010		FCB	\$CA ; S
01240	CLRC	CLR	COUNT ; CLEAR LINE COUNT	02020		FCB	\$E0 ; T
01250	RET	PULS	B,X ; ALL DONE	02030		FCB	\$CE ; U
01260		RTS		02040		FCB	\$FC ; V
				02050		FCB	\$E6 ; W
01270	CROUT	LDA	#\$50 ; CR TTY	02060		FCB	\$FA ; X
01280		BSR	BAUOUT ; PRINT IT	02070		FCB	\$EA ; Y
01290		LDA	#\$44 ; LF TTY	02080		FCB	\$E2 ; Z
01300	BAUOUT	ANDA	#\$7F ; REMOVE LTR FLAG	02090		FCB	\$5E ; (SQUARE BRACKET
01310	*		; OR END BIT 2	02100		FCB	\$7A ; / REVERSE SLASH
01320	OUTBC	LDB	#SPACE ; OFF OR SPACE	02110		FCB	\$64 ;) SQUARE BRACKET
01330		LSRA	; R SHIFT BIT TO C	02120		FCB	\$0CC ; I UP ARROW
01340		BCC	POUT ; IF BIT 0 SPACE	02130		FCB	\$46 ; - LEFT ARROW
01350		LDB	#MARK ; ON OR MARK	02140	*	TABLE	END
01360	POUT	STB	PPOINT ; PRINTER PORT OUT	02150		END	



A CoCo3 version of...

YAHTZEE

GAME

by Kevin Gowan

WHEN I WROTE the original version of Yahtzee back in 1983, it was the first big program I produced, and I felt disappointed that I had to use PMODE 4 in order to obtain the resolution required, which was not exploiting the Colour Computer to its fullest by running a black and white screen. The program itself in its original form has been well received as shown by the comments I've had from Yahtzee devotees who have used it, and I also believe you have seen fit to include it in the 'Best of CoCoOz #2 part 2 - 32K Games'.

When I purchased my CoCo 3 computer a few weeks ago, I had it in mind to enhance the original Yahtzee program to exploit the 64 colours available by simply converting it to run using the new commands available. I thought that it would be a simple matter of changing a few lines here and there, and it would be up there and running in no time at all. It took four days to get it up to the level as presented here in the magazine. Admittedly, the bulk of the original program is still evident, but I've used the PALETTE command extensively to 'colourize' the finished product.

The player's name input routine in the original version has been enhanced in this one to allow error correction / alteration while entering each player's name. Even though the CoCo 3 allows on screen printing in the High-Resolution Graphics mode, the line spacing and character spacing was such that it was incompatible for use while maintaining the original score card format, so I retained the on-screen HDRAW function for the score card layout, and exploited the CoCo 3's HPRINT command (incorporating lower case on screen printing) to give the revised prompts/instructions during the progress of a game.

The HSCREEN 2 mode was selected to give a bountiful

number of simultaneous colour combinations and supply adequate resolution. As the HSCREEN command of the CoCo 3 employs three actions in the one instruction (equivalent to the PMODE, PCLS and SCREEN commands of CoCo 1 & 2), I could not employ the feature of having a blank scorecard in reserve for copying each time players complete a turn, one of two reasons. Firstly, there isn't enough memory to keep a 32K reserve screen with a blank scorecard in it. Secondly, the HSCREEN command causes the screen to be cleared when you use it, thereby losing the current hi-res picture.

I had to arrange to maintain the original scorecard intact at all times, and blank out the relevant scores as each player had their 13 scoring turns. Up to 4 players can be accommodated by the game (same as the original), but now, each player has an individual colour set combination of the same basic scorecard, which is coloured differently again to the scorecard presented at the start of the game (ie 5 colour sets have been employed).

In the original version, the players had to remember their current score for comparison purposes with other players involved, which was a disadvantage. I have remedied this in the CoCo 3 version by utilizing the right hand side of the screen to display the current score status of all players during the progress of the game. (This extra screen space is made available by using the original 256 x 192 pixel area within the 320 x 192 pixel framework using HSCREEN2 of the CoCo 3.) This additional area is used as the introduction phase of the game to indicate a few features of the game.

I hope you find this game of sufficient interest to be included in a future Australian CoCo magazine for the benefit of those CoCo 3 owners who have enjoyed my original version,

and, in general, for any CoCo 3 owner who may want to use a program which demonstrated some of the CoCo 3's capabilities.

The Listing:

```

1 ***** YAHTZEE *****
   ***** K.GOWAN *****
   ***** NOV '86 *****
2 GOTO10
3 SAVE"122:3":END'1
4 'COCO3 VERSION
10 HBUFF 1,250:HBUFF2,250:HBUFF3
   ,250:HBUFF4,250:HBUFF5,250:HBUFF
6,250:HBUFF7,250:HBUFF8,5200:HBU
FF9,80
20 ON BRK GOTO2260
30 WIDTH40
40 POKE65497,0
50 PALETTE RGB
60 FORJ=0TO15:PALETTE J,10:NEXT:
CLS3:HSCREEN2:CLS3
70 HCOLOR2
80 PALETTE2,63:PALETTE1,54:PALET
TE 3,54
90 GOTO2160
100 FOR L=1TO LEN(V$):P$=MID$(V$
,L,1):IF P$=" "THEN130
110 IF ZZ=3 OR ZZ=4 THEN ZZ=ZZ E
LSE ZZ=2
120 HDRAW"BM"+STR$(X)+" "+STR$(Y
)+"":S4C"+STR$(ZZ):HDRAW L$(ASC(P
$)-33)
130 X=X+7:NEXT L:RETURN
140 HPUT(2,150)-(253,190),8:RETU
RN
150 HLINE(XX,YY)-(XX+2,YY+2),PRE
SET,BF:RETURN
160 A$=INKEY$:IF A$=""THEN160ELSE
RETURN
170 IF N(J,Q9)=-1THEN RETURN ELS
E Z=Y0
180 W$=STR$(Z):W$=RIGHT$(W$,LEN(
W$)-1)
190 RHS=RHS+1:IF RHS<1.5THEN HCO
LOR3:HLINE(257,0)-(319,191),PRES
ET,BF:FORC5=1TO Q:HPRINT(33,C5*5
-3),"PLAYER":U$=STR$(C5):HPRINT(
34,C5*5-2),"#"+RIGHT$(U$,LEN(U$
)-1):HPRINT(33,C5*5-1),"SCORES":
HPRINT(33,C5*5+1),".....":NEXT
C5:HCOLOR4
200 IF LEN(W$)<3THEN W$=" "+W$:G
OTO200ELSE GOSUB100:RETURN
210 GOSUB140:Y1=0:Y2=0:FOR J=1TO
6:Y0=N(J,Q9):IF Y0=-1THEN Y0=0
220 Y1=Y1+Y0:X=93:Y=(J*10)+51:GO
SUB170:NEXT J:IF Y1>63THEN Y1=Y1

```

```

+35: X=93: Y=121: W$=" 35": GOSUB100
230 X=93: Y=131: HLINE(93,124)-(11
4,132), PRESET, BF: Z=Y1: GOSUB180
240 FOR J=7 TO 13: Y0=N(J, Q9): IF Y0
=-1 THEN Y0=0
250 Y2=Y2+Y0: X=227: Y=((J-6)*10)+
51: IF J=14 AND N(J, Q9)=0 THEN NEXT
J ELSE GOSUB170: NEXT J
260 Z=N(14, Q9): Y2=Z+Y2: IF Z<0 TH
EN HLINE(227,124)-(247,132), PRES
ET, BF: X=227: Y=131: GOSUB180
270 X=227: Y=141: HLINE(227,134)-(
247,142), PRESET, BF: Z=Y2: GOSUB180
: X=223: Y=23: HLINE(223,16)-(244,2
4), PRESET, BF: Z=Y1+Y2: GOSUB180: TT
(Q9)=Z: GOSUB290: RETURN
280 FORJ=1 TO 7: X=93: Y=J*10+51: HPU
T(X, Y)-(X+20, Y-6), 9: NEXTJ: FORJ=1
TO 8: X=226: Y=J*10+51: HPUT(X, Y)-(X
+20, Y-6), 9: NEXTJ: X=166: Y=23: HPUT
(X, Y)-(X+20, Y-6), 9: RETURN
290 FOR T=1 TO Q: X=275: Y=T*40+10:
HLINE(X, Y)-(X+20, Y-6), PRESET, BF:
ZZ=3: Z=TT(T): GOSUB180: NEXT T: ZZ=
4: HDRAW"C4": RETURN
300 M=(10+(J-1)*28): N=M+20: ON K
GOTO 310,320,330,340,350,360,370
310 HPUT(M,20)-(N,40),1,PSET:RET
URN
320 HPUT(M,20)-(N,40),2,PSET:RET
URN
330 HPUT(M,20)-(N,40),3,PSET:RET
URN
340 HPUT(M,20)-(N,40),4,PSET:RET
URN
350 HPUT(M,20)-(N,40),5,PSET:RET
URN
360 HPUT(M,20)-(N,40),6,PSET:RET
URN
370 HPUT(M,20)-(N,40),7,PSET:RET
URN
380 M=(8+(J-1)*28): N=M+24: HLINE(
M,18)-(N,42),PSET,B: RETURN
390 M=(8+(J-1)*28): N=M+24: HLINE(
M,18)-(N,42),PRESET,B: RETURN
400 GOSUB160: IF A$="Y" THEN 410 ELSE
IF A$="N" THEN GOSUB140: VV=1: RE
TURN ELSE 400
410 GOSUB140: HPRINT(2,21),"Pleas
e choose:[ENTER] to end": SOUND20
0,1
420 GOSUB160: IF A$=CHR$(13) THEN 4
50
430 R=VAL(A$): IF R>50R R<1 THEN 42
0
440 J=R: GOSUB380: D(R+5)=RND(6): G
OTO 420
450 GOSUB140: FOR J=1 TO 5: GOSUB390
: NEXT J: HLINE(166,16)-(187,24),P
RESET, BF: X=166: Y=23: W$=TW$: GOSUB
100: FOR J=1 TO 5: K=7: GOSUB300: D(J)
=D(J+5): K=D(J): GOSUB300: NEXT J: R
ETURN
460 FORJ=0 TO 15: PALETTE J,46: IF J
=2 THEN CLS3: NEXT J ELSE NEXTJ: PA
LETTE1,9: PALETTE2,0: PALETTE3,54:
PALETTE4,46
470 RETURN
480 FORJ=0 TO 15: PALETTEJ,9: IF J=2
THEN CLS3: NEXTJ ELSE NEXTJ: PALET
TE1,54: PALETTE2,63: PALETTE3,18: P
ALETTE4,9
490 RETURN
500 FORJ=0 TO 15: PALETTEJ,45: IF J=
2 THEN CLS3: NEXTJ ELSE NEXTJ: PALE
TTE1,54: PALETTE2,0: PALETTE3,9: PA
LETTE4,45
510 RETURN
520 FORJ=0 TO 15: PALETTE J,24: IF J
=2 THEN CLS3: NEXTJ ELSE NEXTJ: PAL
LETTE1,9: PALETTE2,0: PALETTE3,54: P
ALETTE4,24
530 RETURN
540 '
550 GOSUB1490
560 X=59: Y=162: W$="HOW MANY PLAY
ERS? ": GOSUB100: SOUND200,1
570 GOSUB160: IF VAL(A$)<10R VAL(
A$)>4 THEN 570 ELSE SOUND150,1: Q=VAL
(A$): X=3: Y=142: W$=A$: GOSUB140: GO
SUB100: IF Q=1 THEN W$="PLAYER": G
OSUB100 ELSE W$="PLAYERS": GOSUB1
00
580 HPRINT(2,21),"12 Letters max
:[ENTER] to end"
590 FOR J=1 TO Q: X=24: Y=163: W$="P
LAYER "+CHR$(J+48)+"'S NAME : ":
GOSUB100: SOUND200,1: W1$="": VV=0
600 GOSUB160: IF A$=CHR$(8) THEN 61
0 ELSE IF A$=CHR$(13) THEN 630 ELSE
F=ASC(A$): IF F<32OR F=64 OR F>90
THEN 600
610 IF A$=CHR$(8) THEN X=X-7: IF X
<150 THEN X=150: GOTO 600 ELSE W1$=LE
FT$(W1$,LEN(W1$)-1): HLINE(X,Y)-(
X+7,Y-7),PRESET,BF: SOUND240,1: GO
TO 600
620 VV=VV+1: IF VV>12 THEN 630 ELSE
W1$=W1$+A$: W$=A$: GOSUB100: SOUND2
40,1: GOTO 600
630 IF W1$="" THEN 600 ELSE Q$(J)=V
1$: HLINE(1,156)-(254,164),PRESET
,BF: NEXT J
640 FOR X=1 TO Q: FOR J=1 TO 13: N(J,
X)=-1: NEXT J: N(14,X)=0: NEXT X
650 FOR Q8=1 TO 13: FOR Q9=1 TO Q: GO
SUB280
660 ON Q9 GOSUB 460,480,500,520
670 HLINE(2,143)-(86,135),PRESET
,BF: X=3: Y=142: W$=Q$(Q9): GOSUB100
680 X=166: Y=23: W$="# 1": GOSUB100
: GOSUB210
690 IF Q9=1 THEN PALETTE4,0 ELSE I
F Q9=2 THEN PALETTE4,54 ELSE IF Q9
=3 THEN PALETTE4,63 ELSE PALETTE4,
63
700 HPRINT(3,21),"Press [ENTER]
to throw dice": SOUND200,1
710 GOSUB160: IF A$<CHR$(13) THEN
SOUND150,1: GOTO 710
720 GOSUB140
730 FOR J=1 TO 5: D(J)=RND(6): D(J+5)
=D(J): K=D(J): GOSUB300: NEXT J
740 HPRINT(4,21),"Want a 2nd thr
ow? (Y/N)": SOUND200,1: TW$="# 2":
VV=0: GOSUB400
750 IF VV=1 THEN 980
760 GOSUB140: HPRINT(3,21),"Want
another throw? (Y/N)": SOUND200,1
: TW$="# 3": GOSUB400
770 GOTO 980
780 XX=119: YY=57: X1=XX: Y1=YY: HLI
NE(XX,YY)-(XX+2,YY+2),PSET,BF
790 SOUND200,1: GOSUB160: CK=0
800 IF A$=CHR$(94) THEN IF YY=57T
HEN Y1=57 ELSE Y1=YY-10: GOTO 880
810 IF A$=CHR$(10) THEN IF YY=107
AND XX=119 THEN Y1=107 ELSE IF YY=
127 AND XX=250 THEN Y1=127 ELSE Y1=
YY+10: GOTO 880
820 IF A$=CHR$(8) THEN X1=119: IF X
X=250 AND YY=127 OR YY=117 THEN Y1=
107: GOTO 880 ELSE 880
830 IF A$=CHR$(9) THEN X1=250: GOT
O 880
840 IF A$=CHR$(13) THEN SOUND240,
1: GOTO 870
850 IF A$<>"0" THEN 790
860 SOUND240,1: CK=4
870 K=((YY-57)/10)+1: IF XX=250 TH
EN K=K+6: RETURN ELSE RETURN
880 GOSUB150
890 HLINE(X1,Y1)-(X1+2,Y1+2),PSE
T,BF
900 YY=Y1: XX=X1: GOTO 790
910 GOSUB140: X=17: Y=162: W$="COMB
INATION INVALID FOR CATAGORY": GO
SUB100: FOR J=1 TO 5: NEXT J1: GOSU
B140: GOSUB150: RETURN
920 GOSUB140: HPRINT(2,19),"Your
current turn's finished": HPRINT(
3,22),"Press [ENTER] to continue
"
930 A$=INKEY$: IF A$<CHR$(13) THE
N LL=LL+1: J=J+1: ELSE SOUND240,1
: FOR J=1 TO 5: GOSUB390: K=7: GOSUB30
0: NEXT J: RETURN
940 IF J>5 THEN J=1
950 IF LL=3 THEN SOUND250,1: GOSUB
380 ELSE IF LL=6 THEN SOUND235,1: G
OSUB390: LL=0
960 GOTO 930
970 HPRINT(2,21),"You must forfe
it a catagory": HPRINT(1,22),"ARR
OW keys to select catagory": RETU
RN
980 GOSUB140: HPRINT(1,19),"ARROW
keys to select catagory": HPRINT
(1,21),"[ENTER] when ready, or h
it '0'": HPRINT(1,22),"to score Z
ero in that catagory"
990 GOSUB780
1000 IF CK=4 AND N(K,Q9)=-1 THEN 10
70
1010 IF CK=4 THEN GOSUB910: GOSUB1
50: GOSUB140: GOTO 980
1020 T=0: IF K=14 THEN 1060
1030 T=0: IF N(K,Q9)=-1 THEN GOSU
B1480: ON K GOSUB1400,1410,1420,1
430,1440,1450,1130,1170,1210,126
0,1350,1380,1390
1040 IF CK=0 THEN GOSUB910: GOTO 98
0 ELSE N(K,Q9)=T: GOSUB210: GOSUB15
0: GOSUB920
1050 GOTO 1120
1060 IF N(12,Q9)<1 THEN GOSUB910:
GOSUB150: GOTO 980 ELSE GOSUB1380: I
F CK=1 THEN T=N(14,Q9)+100: GOTO 10
80 ELSE 1030
1070 N(K,Q9)=0: GOSUB210: GOSUB150
: GOSUB140: GOSUB920: GOTO 1120
1080 N(14,Q9)=T: GOSUB210: GOSUB15
0: GOSUB140: X=45: Y=162: W$="YAHTZ
E 100 BONUS AWARDED": GOSUB100: GO
SUB970
1090 GOSUB780
1100 IF N(K,Q9)<>-1 THEN GOSUB910
: GOSUB970: GOTO 1090
1110 N(K,Q9)=0: GOSUB210: GOSUB150
: GOSUB920
1120 NEXT Q9: NEXT Q8: GOTO 2020
1130 T=0: IF D(1)=D(2) AND D(1)=D(
3) THEN 1160 ELSE IF D(1)=D(2) AND D
(1)=D(4) THEN 1160 ELSE IF D(1)=D(2
)&D(1)=D(5) THEN 1160 ELSE IF D(
1)=D(3) AND D(1)=D(4) THEN 1160 ELSE
IF D(1)=D(3) AND D(1)=D(5) THEN 11
60 ELSE IF D(1)=D(4) AND D(1)=D(5)

```

```

THEN1160
1140 IF D(2)=D(3)AND D(2)=D(4)TH
EN1160ELSE IF D(2)=D(3)AND D(2)=
D(5)THEN1160ELSE IF D(2)=D(4)AND
D(2)=D(5)THEN1160ELSE IF D(3)=D
(4)AND D(3)=D(5)THEN1160
1150 CK=0: RETURN
1160 T=D(1)+D(2)+D(3)+D(4)+D(5):
CK=1: RETURN
1170 T=0: IF D(1)=D(2)AND D(1)=D(
3)AND D(1)=D(4)THEN1200ELSE IF D
(1)=D(2)AND D(1)=D(3)AND D(1)=D(
5)THEN1200ELSE IF D(1)=D(2)AND D
(1)=D(4)AND D(1)=D(5)THEN1200
1180 IF D(1)=D(3)AND D(1)=D(4)AN
D D(1)=D(5)THEN1200ELSE IF D(2)=
D(3)AND D(2)=D(4)AND D(2)=D(5)TH
EN1200
1190 CK=0: RETURN
1200 T=D(1)+D(2)+D(3)+D(4)+D(5):
CK=1: RETURN
1210 IF D(1)=D(2)AND D(3)=D(4)AN
D D(3)=D(5)THEN1250ELSE IF D(1)=
D(3)AND D(2)=D(4)AND D(2)=D(5)TH
EN1250ELSE IF D(1)=D(4)AND D(2)=
D(3)AND D(2)=D(5)THEN1250
1220 IF D(1)=D(5)AND D(2)=D(3)AN
D D(2)=D(4)THEN1250ELSE IF D(2)=
D(3)AND D(1)=D(4)AND D(1)=D(5)TH
EN1250ELSE IF D(2)=D(4)AND D(1)=
D(3)AND D(1)=D(5)THEN1250
1230 IF D(2)=D(5)AND D(1)=D(3)AN
D D(1)=D(4)THEN1250ELSE IF D(3)=
D(4)AND D(1)=D(2)AND D(1)=D(5)TH
EN1250ELSE IF D(3)=D(5)AND D(1)=
D(2)AND D(1)=D(4)THEN1250ELSE IF
D(4)=D(5)AND D(1)=D(2)AND D(1)=
D(3)THEN1250
1240 T=0: CK=0: RETURN
1250 T=25: CK=1: RETURN
1260 IF T9=1THEN T=30: CK=1: RETUR
N ELSE T1=0: T=0: CK=0: IF D(1)<D(
2)AND D(1)<D(3)AND D(1)<D(4)AN
D D(2)<D(3)AND D(2)<D(4)AND D(
3)<D(4)THEN T=D(1)+D(2)+D(3)+D(
4): GOSUB1330: IF T1=1THEN1320
1270 T=0: IF D(1)<D(2)AND D(1)<D
(3)AND D(1)<D(4)AND D(2)<D(3)
AND D(2)<D(4)AND D(3)<D(4)THEN
T=D(1)+D(2)+D(3)+D(5): GOSUB1330
: IF T1=1THEN1320
1280 T=0: IF D(1)<D(2)AND D(1)<D
(4)AND D(1)<D(5)AND D(2)<D(4)
AND D(2)<D(5)AND D(4)<D(5)THEN
T=D(1)+D(2)+D(4)+D(5): GOSUB1330
: IF T1=1THEN1320
1290 T=0: IF D(1)<D(3)AND D(1)<D
(4)AND D(1)<D(5)AND D(3)<D(4)
AND D(3)<D(5)AND D(4)<D(5)THEN
T=D(1)+D(3)+D(4)+D(5): GOSUB1330
: IF T1=1THEN1320
1300 T=0: IF D(2)<D(3)AND D(2)<D
(4)AND D(2)<D(5)AND D(3)<D(4)
AND D(3)<D(5)AND D(4)<D(5)THEN
T=D(2)+D(3)+D(4)+D(5): GOSUB1330
: IF T1=1THEN1320
1310 T=0: CK=0: RETURN
1320 T=30: CK=1: RETURN
1330 IF T=100R T=140R T=18THEN T
=1
1340 RETURN
1350 IF T9=1THEN T=40: CK=1: RETUR
N ELSE T=0: CK=0: IF D(1)=D(2)OR D
(1)=D(3)OR D(1)=D(4)OR D(1)=D(5)
OR D(2)=D(3)OR D(2)=D(4)OR D(2)=
D(5)OR D(3)=D(4)OR D(3)=D(5)OR D
(4)=D(5)THEN1370
1360 T=D(1)+D(2)+D(3)+D(4)+D(5):
IF T=150R T=20THEN CK=1: T=40: RET
URN
1370 CK=0: T=0: RETURN
1380 T=0: CK=0: IF D(1)=D(2)AND D(
1)=D(3)AND D(1)=D(4)AND D(1)=D(5)
)THEN T=50: CK=1: RETURN ELSE RETU
RN
1390 T=0: FOR J=1TO5: T=T+D(J): NEX
T J: CK=1: RETURN
1400 T2=1: GOSUB1460: RETURN
1410 T2=2: GOSUB1460: RETURN
1420 T2=3: GOSUB1460: RETURN
1430 T2=4: GOSUB1460: RETURN
1440 T2=5: GOSUB1460: RETURN
1450 T2=6: GOSUB1460: RETURN
1460 T=0: FOR J=1TO5: IF D(J)=T2 T
HEN T=T+T2
1470 NEXT J: IF T=0THEN CK=0: RETU
RN ELSE CK=1: RETURN
1480 GOSUB1380: IF CK=0OR N(12, Q9
)<=-1OR N(D(1), Q9)<=-1THEN RETURN
ELSE T9=1: RETURN
1490 X=159: Y=38: W$="Y: A: H: T: Z: E:
E": GOSUB100
1500 HLINE(0, 0)-(255, 191), PSET, B
: HLINE(0, 45)-(255, 45), PSET: HLINE
(0, 147)-(255, 147), PSET
1510 HDRAW"BM44, 112; S4U8R8D8L8BU
10U8R8D8L8BU10U8R8D8L8BU10U8R8D8
L8BU10U8R8D8L8BU10U8R8D8L8": FOR
J=1TO6: HPAINT(48, 59+(J-1)*10),
0, 0: NEXT J: FOR J=1TO21: READ A, B: H
SET(A, B, 2): NEXT J
1520 HDRAW"BM63, 123; S4NR24U10NR2
4U10NR24U10NR24U10NR24U10NR24U10
NR24U10R24D70": HDRAW"BM91, 134; S4
NR24U11NR24U10NR24U10NR24U10NR24
U10NR24U10NR24U10NR24U10R24D81"
1530 HDRAW"BM196, 133; S4NR24U10NR
24U10NR24U10NR24U10NR24U10NR24U1
0NR24U10NR24U10R24D80": HDRAW"BM2
24, 144; S4NR24U11NR24U10NR24U10NR
24U10NR24U10NR24U10NR24U10NR24U1
0NR24U10R24D91"
1540 HPRINT(33, 1), "YAHTZEE": HPRI
NT(33, 2), " is a": HPRINT(33, 3), "
card": HPRINT(33, 4), " game": HPRI
NT(33, 5), " using": HPRINT(33, 6), "
dice": HPRINT(33, 8), "Up to 4": HP
RINT(33, 9), "players": HPRINT(33, 1
1), " Three": HPRINT(33, 12), " thro
ws": HPRINT(33, 13), " of th
1550 HPRINT(33, 14), "dice is": HPR
INT(33, 15), "allowed": HPRINT(33, 1
7), " 13": HPRINT(33, 18), " moves"
: HPRINT(33, 19), " for a": HPRINT(3
3, 20), " full": HPRINT(33, 21), " sc
ore": HPRINT(33, 22), " card."
1560 X=3: Y=61: W$="ACES ": ZZ=
2: GOSUB100: ZZ=3: W$="ADD": GOSUB10
0: ZZ=2: W$=" 3 OF KIND ": GOS
UB100: ZZ=3: W$="TOT": GOSUB100
1570 X=3: Y=71: W$="TWOS ": ZZ=
2: GOSUB100: ZZ=3: W$="ADD": GOSUB10
0: ZZ=2: W$=" 4 OF KIND ": GOS
UB100: ZZ=3: W$="TOT": GOSUB100
1580 X=3: Y=81: W$="THREE ": ZZ=
2: GOSUB100: ZZ=3: W$="ADD": GOSUB10
0: ZZ=2: W$=" FULLHOUSE ": GOS
UB100: ZZ=3: W$=" 25": GOSUB100
1590 X=3: Y=91: W$="FOURS ": ZZ=
2: GOSUB100: ZZ=3: W$="ADD": GOSUB10
0: ZZ=2: W$=" SMALL STR ": GOS
UB100: ZZ=3: W$=" 30": GOSUB100
1600 X=3: Y=101: W$="FIVES ": ZZ
=2: GOSUB100: ZZ=3: W$="ADD": GOSUB1
00: ZZ=2: W$=" LARGE STR ": GO
SUB100: ZZ=3: W$=" 40": GOSUB100
1610 X=3: Y=111: W$="SIXES ": ZZ
=2: GOSUB100: ZZ=3: W$="ADD": GOSUB1
00: ZZ=2: W$=" YAHTZEE ": GO
SUB100: ZZ=3: W$=" 50": GOSUB100
1620 X=3: Y=121: W$="BONUS">63 ": ZZ
=2: GOSUB100: ZZ=3: W$="+35": GOSUB1
00: ZZ=2: W$=" CHANCE ": GO
SUB100: ZZ=3: W$="TOT": GOSUB100
1630 X=31: Y=131: W$="TOTAL =
": GOSUB100: ZZ=2: W$=" BONUS": GOS
UB100: ZZ=3: W$=" 100": GOSUB10
0
1640 X=157: Y=141: W$="TOTAL =" : G
OSUB100: ZZ=2
1650 X$="R39D10NL39D10L39U20": HD
RAW"BM156, 5; S4"+X$: HDRAW"BM212, 5
; S4"+X$: X=159: Y=13: W$="THROW S
CORE": GOSUB100
1660 K=7: FOR J=1TO5: GOSUB300: NEX
T J
1670 X=19: Y=12: W$="1 2 3 4
5": U9=11: GOSUB100
1680 ZZ=4
1690 RETURN
1700 REM ASSIGN VALUES
1710 FOR J=0TO30: READ R$: L$(J)=R
$: NEXT J
1720 DATA BR2ND1BU2U4, BR1BU4U1BR
2D1, BR1U2L1BU2R1U2BR2D2R1BD2L1D2
, BU1R2ND1R1E1H1L2H1E1R1U1R2, U1E
4U1BL4D1BF4D1, BR4H1U1H3E1F1G2D2F
1R1E2U1, BR2BU4U2, BR2H1U4E1, BR2E1
U4H1
1730 DATA BU2E2NH2NU2NE2NF2D2, BR
2BU1U2NL2NU2R2, BR2NU1G1, BU3R4, BR
2U1, U1E4U1, BU1NE4U4E1R2F1D4G1L2H
1, R2NR2U6L1G1, NR4E4U1H1L2G1, BU1F
1R2E1U1H1NL2E1U1H1L2G1, BR3U6G3R4
1740 DATA BU1F1R2E1U2H1L2G1U3R4,
BU3E1R2F1D2G1L2H1U4E1R3, E4U2L4D1
, BU1U1E1NR2H1U1E1R2F1D1G1F1D1G1L
2H1, BR1R2E1U4H1L2G1D2F1R2E1, BR2U
1BU2U1, BR1BD1E1U1BU2U1, BU3NE3F3
1750 DATA BU1R4BU2L4, BR1E3H3, BU5
E1R2F1D1G1L1D1BD2D1
1760 FOR J=32TO57: READ R$: L$(J)=
R$: NEXT J
1770 DATA U4E2F2D2NL4D2, R3E1U1H1
E1U1H1L3R1D3NR2D3, BE4BU1H1L2G1D4
F1R2E1, R3E1U4H1L2NL1D6, R4U1BU4U1
L3NL1D3NR2D3, R1NR1U3NR2U3NL1R3D1
, BE2R1NR1D1ND1G1L1H1U4E1R2D1, U3N
U3R4NU3D3, BR1R1NR1U6NL1R1, BU1NU1
F1R1E1U5NL1R1
1780 DATA R1U6NL1BD3R1E2U1BD5ND1
H2, R4U1BG1BL2U6NL1R1, U6F2E2D6, U6
F4NU4D2, BU1U4E1R2F1D4G1L2H1, R1NR
1U6NL1R2F1D1GL2, BU1U4E1R2F1D4G1D
1R1BH1L2H1, U6R3F1D1G1L1NL2F2D1, B
U1F1R2E1U1H1L2H1U1E1R2F1, BR2U6NL
2R2
1790 DATA BU1NU5F1R1E1R1ND1U5, BU
3NU3F1D1F1E1U1E1U3, NU6E2F2U6, U1E
4U1BL4D1F4D1, BR1R1NR1U3H2U1BR4D1
G2, BU5U1R4D1G4D1R4U1
1800 DATA 48, 58, 50, 66, 46, 70, 50, 7
6, 48, 78, 46, 80, 46, 86, 50, 86, 46, 90,
50, 90, 46, 96, 50, 96, 48, 98, 46, 100, 5
0, 100, 46, 106, 46, 106, 50, 106, 46, 11
0, 48, 110, 50, 110
1810 RETURN
1820 HCOLOR1

```

continued on page 48

DRIFTING

32K ECB
GAME

by D & T Delbourgo

THIS GAME IS an exercise in momentum conservation. You have five spacemen, drifting outside a mother ship.

Use the right joystick to fire the jetpack (opposite to the motion) so as to direct the spacemen to the mother craft (the circle in the centre of the screen).

If you take too long or use up too much fuel or drift off the screen, then you lose a spaceman.

As the game progresses there is less time allotted and less fuel to spare in making way to the centre.

You may play at two levels. The easy level has no asteroids to impede your path; the difficult level has a sprinkling of boulders which you must avoid -- a crash is fatal.

Have fun.

With practice you will get quite good at judging where and to what extent (elongation of joystick handle) you should fire your jet. The medals you are awarded at the end are proportional to your final score.

The Listing:

```
1 POKE359,57:SCREEN0,1:CLS(RND(9)-1)
2 P=128+RND(127):Q=P-16*(RND(17)-9):IFQ>255ORQ<128THENGOTO2
3 IF P/16=INT(P/16)THEN2
4 PLAY"L5504CEG05CO4GEC":POKE65495,0:FORR=1TO3
5 FORI=1TO36-4*R:POKE957+I+66*R,P:POKE1602-I-66*R,Q:NEXTI:PLAY"C04L95"
6 FORJ=1TO18-4*R:POKE993+32*J+62*R,P:POKE1566-32*J-62*R,Q:NEXT:P:LAY"EO4L95"
7 FORK=1TO34-4*R:POKE1569-K-66*R,P:POKE990+K+66*R,Q:NEXT:PLAY"GO4L95"
8 FORL=1TO16-4*R:POKE1535-32*L-62*R,P:POKE1024+32*L+62*R,Q:NEXT:PLAY"C05L95":NEXTR
9 PRINT@198," D R I F T I N G ..
```

```
..":PRINT@230," by
";:PRINT@262,"D. AND T. DELBO
URGO ";:PRINT@294," (C) 198
5 ";
10 DIMX(6),Y(6):CLEAR1000:FORT=1
TO1000:NEXTT
11 CLSO:X$="D":I=14:GOSUB65:X$="
R":I=12:GOSUB65:X$="I":I=10:GOSU
B65:X$="F":I=8:GOSUB65
12 X$="T":I=6:GOSUB65:X$="I":I=4
:GOSUB65:X$="N":I=2:GOSUB65:X$="
G":I=0:GOSUB65
13 X$="(1)":FORK=1TO11:PRINT@223
+K,X$,:PRINT@222+K,CHR$(128):NE
XTK
14 X$="NOVICE":FORK=1TO13:PRINT@
251-K,X$,:PRINT@257-K,CHR$(128):
:NEXTK
15 X$="(2)":FORK=1TO11:PRINT@287
+K,X$,:PRINT@286+K,CHR$(128):NE
XTK
16 X$="EXPERT":FORK=1TO13:PRINT@
315-K,X$,:PRINT@321-K,CHR$(128):
:NEXTK
17 PRINT@356,"PRESS YOUR CHOICE
NUMBER":S=0:L=0:N=100
18 I$=INKEYS:IFI$=""THEN18
19 IFI$=CHR$(49)THEN21
20 IFI$=CHR$(50)THEN37
21 POKE359,126:GOSUB53:U=RND(4)*
(2*RND(2)-3):V=RND(3)*(2*RND(2)-
3):F=30:N=N-2
22 FORK=1TON:X=X+U:Y=Y+V
23 IFX>254ORX<0THENGOSUB68:GOTO2
1
24 IFY>191ORY<3THENGOSUB68:GOTO2
1
25 IFX>120ANDX<135ANDY>89ANDY<10
4THENGOSUB73:GOTO21
26 IFX>120ANDX<135ANDY>89ANDY<10
4THEN21
27 JO=JOYSTK(0):J1=JOYSTK(1)
28 IFPEEK(65280)=254ORPEEK(65280
)=126THENA=(JO-31):B=(J1-31):ELS
EGOTO35
29 U=U-INT(A/8):V=V-INT(B/8):P=X
+A:Q=Y+B:F=F-INT(SQR(A*A+B*B)/5)
:IFQ<0THENGOSUB82:GOTO21
30 IFP>255THENP=255
31 IFP<0THENP=0
32 IFQ>191THENQ=191
33 IFQ<0THENQ=0
34 LINE(X+1,Y-3)-(P,Q),PSET:PLAY
"L255O2FGFGFG":A=0:B=0:LINE(X+1,
Y-3)-(P,Q),PSET
35 DRAW"BM"+STR$(X)+","+STR$(Y)+
"+S$:DRAW"CO:"+BM"+STR$(X)+,
"+STR$(Y)+","+S$+":C1":A=0:B=0:N
EXTK
36 GOSUB87:GOTO21
```

```
37 POKE359,126:GOSUB53:GOSUB59:U
=RND(6)*(2*RND(2)-3):V=RND(4)*(2
*RND(2)-3):F=30:N=N-2
38 FORK=1TON:X=X+U:Y=Y+V
39 IFX>254ORX<0THENGOSUB68:GOTO3
7
40 IFY>191ORY<3THENGOSUB68:GOTO3
7
41 IFX>120ANDX<135ANDY>89ANDY<10
5THENGOSUB73:GOTO37
42 FORI=1TO6:IFABS(X+1-X(I))<3 A
ND ABS(Y-3-Y(I))<4 THENGOSUB77:G
OTO37
43 NEXTI
44 JO=JOYSTK(0):J1=JOYSTK(1):IF
PEEK(65280)=254ORPEEK(65280)=126
THENA=(JO-31):B=(J1-31):ELSEGOTO
51
45 U=U-INT(A/6):V=V-INT(B/6):P=X
+A:Q=Y+B:F=F-INT(SQR(A*A+B*B)/5)
:IFQ<0THENGOSUB82:GOTO37
46 IFP>255THENP=255
47 IFP<0THENP=0
48 IFQ>191THENQ=191
49 IFQ<0THENQ=0
50 LINE(X+1,Y-3)-(P,Q),PSET:PLAY
"L255O2FGFGFG":A=0:B=0:LINE(X+1,
Y-3)-(P,Q),PSET
51 DRAW"BM"+STR$(X)+","+STR$(Y)+
"+S$:DRAW"CO:"+BM"+STR$(X)+,
"+STR$(Y)+","+S$+":C1":A=0:B=0:N
EXTK
52 GOSUB87:GOTO37
53 PMODE4,1:PCLS:SCREEN1,1
54 FORR=1TO7STEP3:CIRCLE(128,95)
,R:NEXT
55 X=RND(160)+48:Y=RND(124)+39
56 IFX>96ANDX<160THEN53
57 IFY>75ANDY<123THEN53
58 S$="U6RD6RBU3L3UD2":RETURN
59 M$="U2RD2R2L3D2"
60 X(1)=47+RND(160):Y(1)=12+RND(
12):X(2)=47+RND(160):Y(2)=168+RND
(12)
61 X(3)=12+RND(20):Y(3)=10+RND(1
70):X(4)=224+RND(20):Y(4)=10+RND
(170)
62 X(5)=106+RND(34):Y(5)=77:X(6)
=RND(34)+106:Y(6)=113
63 FORI=1TO6:DRAW"BM"+STR$(X(I))
+,"+STR$(Y(I))+","+M$:NEXTI
64 RETURN
65 FORK=1TO10+I:PRINT@160-K,X$,:
PRINT@161-K,CHR$(128):NEXTK:RET
URN
66 R=RND(8):POKE359,57:SCREEN0,1
:CLSR:RETURN
67 FORK=1TO11+I:PRINT@276-I+K,X$
,:PRINT@275-I+K,CHR$(127+16*R):;
```

```

PRINT@276-I+K,CHR$(127+16*R);:NE
XTK:RETURN
68 GOSUB66:PRINT@202,"LOST IN SP
ACE";:PRINT@268,"FAREWELL!";
69 PLAY"T302L10EL3GL10EL3GL10ECD
EFGL3AP50L7BL20AGL10F#G01B02CL5D
P5002L10BL20AGL10DEL7CT2"
70 X$="!":I=0:GOSUB67:X$="L":I=1
:GOSUB67:I=2:GOSUB67:X$="E":I=3:
GOSUB67:X$="W":I=4:GOSUB67
71 X$="E":I=5:GOSUB67:X$="R":I=6
:GOSUB67:X$="A":I=7:GOSUB67:X$="
F":I=8:GOSUB67:FORT=1TO100:NEXTT
:L=L+1:S=S+F:IFL>4THENGOTO92
72 RETURN
73 PLAY"O1L8CO2":CLS0:POKE359,57
:SCREEN0,1:PRINT@200,"YOU'VE MAD
E IT!";
74 X$="B":I=14:GOSUB65:X$="R":I=12
:GOSUB65:X$="A":I=10:GOSUB65:X
$="V":I=8:GOSUB65
75 X$="O":I=6:GOSUB65:X$="!":FOR
I=4TOSTEP-2:GOSUB65:NEXTI
76 PLAY"O3T2L20CDEL10FL6CL20FEFL
10GL6DL20DEFL16AL20GGFFDEFL6FT2"
:S=S+F+100:RETURN
77 GOSUB66:PRINT@202,"METEOR-STR
UCK";:PRINT@268,"HARD LUCK";
78 PLAY"T302L10EBEEL5EFL10GGGGL5
GAL10GGGGL5GFL3ET2"
79 X$="K":I=0:GOSUB67:X$="C":I=1
:GOSUB67:X$="U":I=2:GOSUB67:X$="
L":I=3:GOSUB67
80 X$="D":I=5:GOSUB67:X$="R":I=6
:GOSUB67:X$="A":I=7:GOSUB67:X$="
H":I=8:GOSUB67:L=L+1:S=S+F:IF L>

```

```

4 THEN92
81 FORT=1TO100:NEXT:GOTO37
82 GOSUB66:PRINT@202,"NO MORE FU
EL!";:PRINT@268,"SO SORRY";
83 PLAY"O1T3L10GG02L5CL1001G02CL
5EL10CAL3GL10CFL5EO1L10G02EL5DL1
0O1G02DL5CT2"
84 X$="Y":I=0:GOSUB67:X$="R":I=1
:GOSUB67:I=2:GOSUB67:X$="O":I=3:
GOSUB67
85 X$="S":I=4:GOSUB67:X$="Q":I=7
:GOSUB67:X$="S":I=8:GOSUB67:L=L+
1:S=S+F:IF L>4 THEN92
86 FORT=1TO100:NEXT:RETURN
87 GOSUB66:PRINT@202,"IT'S TOO L
ATE";:PRINT@268,"YOU FROZE";
88 PLAY"1L1001CL15AL25GL10FL5AL
10FL10FT2"
89 X$="E":I=0:GOSUB67:X$="Z":I=1
:GOSUB67:X$="O":I=2:GOSUB67:X$="
R":I=3:GOSUB67:X$="F":I=4:GOSUB6
7
90 X$="U":I=6:GOSUB67:X$="O":I=7
:GOSUB67:X$="Y":I=8:GOSUB67:L=L+
1:S=S+F:IF L>4 THEN92
91 FORT=1TO100:NEXT:RETURN
92 TH=INT(S/512):S3=S-512*TH:HU=
INT(S3/64):S2=S3-64*HU:TE=INT(S2
/8):UN=S2-8*TE
93 CLS0:PRINT"AFTER LOSING FIVE
SPACEMEN":PRINT"YOUR TOTAL SCORE
IS":S:
94 PRINT@64,"
----":PRINT"FOR CONSPICUOUS BRA
VERY IN THE":PRINT"FACE OF ADVER
SITY YOU HAVE BEEN":PRINT"AWARDE
D THESE DECORATIONS"
95 UN$=CHR$(42)+CHR$(42)+CHR$(42)

```

```

)+CHR$(128):TE$=CHR$(223)+CHR$(1
50)+CHR$(223)+CHR$(128):HU$=CHR$
(207)+CHR$(255)+CHR$(207)+CHR$(1
28):TH$=CHR$(227)+CHR$(239)+CHR$
(227)+CHR$(128):HT$=CHR$(236)+CH
R$(239)+CHR$(236)+CHR$(128)
96 IFUN=0 THEN98
97 FORK=1TOUN:PRINT@445+4*K,UN$;
:NEXTK
98 IFTE=0 THEN100
99 FORK=1TOTE:PRINT@381+4*K,TE$;
:NEXTK
100 IFHU=0 THEN102
101 FORK=1TOHU:PRINT@317+4*K,HU$
;:NEXTK
102 IFTH=0 THEN104
103 FORK=1TOTH:PRINT@221+4*K,TH$
;:PRINT@253+4*K,HT$;:NEXTK
104 PLAY"T202L4FL8CFL28GAFLL13GL2
2G03L4CO2AL8FDL28EFDL13EL22EL4AF
L8CFL28GAFLL13GL22G03L4CL28CCDEL1
4FO2L28FGAB-L803CL5CO2L28B-GL14F
P1P1P1T2"
105 CLS0:PRINT@202,"PRESS ANY KE
Y";:PRINT@264,"FOR ANOTHER DRIFT
";
106 POKE65494,0:IS=INKEY$:IFIS="
"THEN106
107 RUN
200 SAVE"205F:3":END'1

```

YAHTZEE

continued from page 46

```

1830 A$="R20D20L20U20"
1840 B$="BR10BD9L1D2R2U2L1BU9BL1
0"
1850 C$="BR16BD3L1D2R2U2L1BU3BL1
2BD15L1D2R2U2L1BU15BL4"
1860 D$="BR16BD15L1D2R2U2L1BU15B
L12BD3L1D2R2U2L1BU3BL4"
1870 E$="BR10BD3L1D2R2U2L1BD12L1
D2R2U2L1BU15BL10"
1880 HDRAW"BM30,160S4"+A$:HGET(3
0,160)-(50,180),7
1890 HDRAW"BM30,160S4"+B$:HGET(3
0,160)-(50,180),1
1900 HDRAW"BM60,160S4"+A$+C$:HGE
T(60,160)-(80,180),2
1910 HDRAW"BM30,160S4"+D$:HGET(3
0,160)-(50,180),3
1920 HDRAW"BM90,160S4"+A$+D$+C$:
HGET(90,160)-(110,180),4
1930 HDRAW"BM120,160S4"+A$+B$:HD
RAW""+C$+D$:HGET(120,160)-(140,1
80),5
1940 HDRAW"BM150,160S4"+A$+C$:HD
RAW""+D$+E$:HGET(150,160)-(170,1
80),6
1950 HLINE(0,0)-(20,20),PRESET,B
F
1960 HPUT(180,160)-(200,180),RND
(6):HPUT(210,160)-(230,180),RND(
6):HPUT(240,160)-(260,180),RND(6)

```

```

):HPUT(270,160)-(290,180),RND(6)
):HPUT(0,160)-(20,180),1:HPUT(299
,160)-(320,180),RND(6)
1970 FORJ=0TO296STEP30:HPUT(0+J,
0)-(20+J,20),RND(6):NEXT:HPUT(29
9,0)-(320,20),RND(6)
1980 FORJ=35TO150STEP30:HPUT(0,J
)-(20,J+20),RND(6):HPUT(299,J)-(
320,J+20),RND(6):NEXT
1990 FOR I=1TO9:PALETTE 1,10:FOR
J=1TO150:NEXTJ:PALETTE1,54:FORJ=
1TO150:NEXTJ,I
2000 FORJ=0TO15:PALETTE J,10:NEX
T:CLS3:HSCREEN2:CLS3:PALETTE1,54
:PALETTE2,63:PALETTE3,0:PALETTE4
,63
2010 RETURN
2020 HSCREEN2:CLS2
2030 X=110:Y=11:W$="GAME FINISHE
D":GOSUB100
2040 FOR J=1TO Q
2050 X=75:Y=(J*40)-18:W$="PLAYER
# "+CHR$(J+48)+" ":GOSUB100
2060 W$=Q$(J):GOSUB100
2070 X=75:Y=Y+12:W$="SCORE TOTAL
=" :GOSUB100
2080 Z=0:FOR K=1TO6:Z=Z+N(K,J):N
EXT K:IF Z>63THEN Z=Z+35
2090 FOR K=7TO14:Z=Z+N(K,J):NEXT
K

```

```

2100 GOSUB180
2110 W$=" POINTS.":GOSUB100
2120 NEXT J
2130 X=50:Y=188:W$="DO YOU WANT
TO PLAY AGAIN? (Y/N)":GOSUB100
2140 GOSUB160:IF A$="N"THEN PALE
TTE RGB:POKE65496,0:WIDTH32:CLS
:PRINT@271,"BYE":END ELSE IF A$="
Y"THEN2150ELSE2140
2150 RUN20
2160 J=RND(-TIMER):DIM Q$(4),D(1
0),N(14,4),L$(57),TI(4)
2170 HGET(2,150)-(253,190),8:HGE
T(0,7)-(20,1),9
2180 GOSUB1700
2190 X=55:Y=60:W$="Y A H T Z E E
"
2200 FOR L=1TO LEN(W$):P$=MID$(W
$,L,1):IF P$=" "THEN2220
2210 HDRAW"BM"+STR$(X)+"",+STR$(
Y)+"",S12:HDRAW L$(ASC(P$)-33)
2220 X=X+17:NEXT L
2230 X=160:Y=103:W$="BY":GOSUB10
0
2240 X=125:Y=134:W$="KEVIN GOVA
N":GOSUB100
2250 GOSUB1820:GOTO540
2260 POKE65496,0:PALETTERGB:PALE
TTE8,63:HSCREEN0
2270 END
2280 GOTO2260
2290 'YAHTZEE (C)1986
2300 'by Kevin Gowan
2310 '3 Inley Court.
2320 'Trott Park 5158
2330 'South Australia

```

FRICKERS FOLLIES

by Jack Fricker

THIS MONTH WE ARE going to talk again about OS-9 level 2. Well, it finally arrived!

"Was it worth it?", I hear you ask. In my opinion, "Yes!" and here's why! I'll start with the things that I like about it first and then look at the things I don't like (well, I didn't say it was perfect ...)

The first thing that I like is that you don't have to buy or modify the device descriptors and drivers to read and write to anything other than 35 tracks single sided.

Yep, there are descriptors there for 35 tracks single sided, 40 tracks double sided, and 80 tracks double sided (yes, 80 tracks double sided!). All in all about 800k per drive.

This is the equivalent of 4 x 35 track drives. There are descriptors for 2 x 80 track (drive 1 & 2) although there aren't any for the first (drive 0). Therefore you are limited to 35 or 40 double sided.

Another thing new about it is they have introduced the default drive ("/DD"). The purpose for this is this:

Some programs look for additional information on the disk and expect this information to be in a specified place.

For instance the "Help" program in version 2 (NOT level 2) used to look for the actual files containing the help data in the root directory /DO/HELP. This became a problem if you had a hard disk or you were using /D1 or /D2 as your commands disk and your "Help", "Sys" and other base directories were located there.

The assembler ISMON and other software also expects to find the information in /DO. What happens now is these programs will now look for that information in "/DD". "/DD" can be any drive (hard or floppy) or even a RAM disk. The operating system comes with descriptors for either a 35 track single sided or 40 track double sided default device. These can be altered so when the hard disk drivers are written, you will also be able to use your hard disk as "/DD".

"Was it worth it?..."

Yes!!"

Strangely enough the hard disk drivers aren't included in level 2 even though they were included in version 1.2 and Tandy have a hard drive for the CoCo 2 & 3 in the U.S. The driver was even mentioned in the manual.

Another oddity is that there is no method included in level 2 for changing the actual stepping rate of the drives.

None of the previous releases of OS-9 have included them either. It shouldn't be that difficult to make the drives run faster and when I find out I will pass that information on to you.

We have covered some of the things that I do like about OS-9 level 2. The first one I consider to be of major importance is memory.

If you moved up from level 1 and purchased a 128K CoCo 3 and OS-9 level 2 and thought you had left the dreaded error number 207 behind, then think again! Somebody in their wisdom decided that you needed a number of utilities in memory instead of loading them off disk as required. They then put them in the boot file so that you couldn't remove them from memory. This is a good idea with some utilities like "Dir", "List" and "Edit" (which isn't in the boot).

Including files in the boot reduces the time taken to run the utilities but takes up valuable memory. The utilities are not actually in the OS-9 boot file on the disk but seem to be loaded during the booting procedure. A number of those files are used once during an OS-9 session and are then useless. Such things as "Settime" and "Tmode" may only be used once and other files such as "Iniz", "Link", "Load", "Procs", and "Mdir" may not even be used at all.

These things should be left up to the user and not arbitrary loaded and then can't be unloaded. Well at this stage I haven't been able to find out how, but when I do, I will pass on the methods on to you. Loading files in this manner is fine when you have 512K, but when you only have 128K this leaves you with only 48K and this is not enough memory to run some programs like "Stylo" which is the wordprocessing program that I use.

Each window that you open up uses more memory. Obviously level 2 was intended for 512K.

Hopefully next month I will have the solutions to these problems and will be able to pass them on to you.

DISKS & DIRECTORIES

16K Disk
UTILITY

by J.D. Cladingboel

I HAVE RECENTLY graduated from cassette to disk on my CoCo, and not only am I blessed this new, faster and more reliable device, but I also have cause to bless the choice I made in selecting my first box of disks. (Multicoloured disks!)

Like everyone else, I guess, it wasn't long before I was faced with a box of disks, some full, some not, with very little idea of exactly what was on which!

Before hunting around for a program I once heard of (somewhere!) which prints a directory on a sheet of paper which is then made into a disk sleeve. I wondered if I could take advantage of the fact that my disks had come in clear plastic sleeves!

All I needed was a program to print a directory on a piece of paper which would fit inside the sleeve, and be visible in front of the disk. For a long directory, I could fold the paper in half, and have both sides visible.

The result is the attached simple program which does this job for me very efficiently. I call it "DIRECT.PRT", and it works by first asking for the disk name (which is all I now write on my disk labels) and the date. As soon as the date is ENTERed the printer commences printing.

When the directory has been printed, finishing with number of granules free, the program asks if you would like to print another. If you reply 'Y', you will not be asked to give the date again, but instead the printer will begin operating as soon as the next disk name is ENTERed.

In this way, a box of disks can be run through in a few minutes, and order restored from chaos!

I claim no copyright on the program, so if you wish to put on more bells and whistles ('is printer ready?', 'an alphabetical directory listing', etc) please feel free to amend it to suit your requirements.

Incidentally, the coloured disks are "Memron" brand and come in a transparent sleeve.

My printer (a DVP 2200) operates at 9600 baud. Line 50 should be amended for other baud rates by changing "POKE150,1" to ...

POKE150,7 for 4800 baud.
POKE150,18 for 2400 baud.
POKE150,40 for 1200 baud.
POKE150,87 for 600 baud.
POKE150,110 for 300 baud.

The Listing:

```
0 GOTO10
3 SAVE"163A:3":END'1
10 '*****
20 ' DIRECTORY PRINT PROGRAM
30 ' SET BAUD RATE IN LINE 50.
40 '*****
50 POKE150,1
60 CLS:LINE INPUT"DISK NAME ";N$
70 INPUT"TODAY'S DATE ";D$
80 PRINT#-2,N$ as at ";D$
90 POKE111,254:DIR
100 PRINT#-2:PRINT#-2, FREE(0);"
    GRANULES FREE."
110 PRINT#-2:PRINT#-2
120 PRINT@32,"DISK DIRECTORY HAS
    BEEN PRINTED."
130 PRINT:PRINT"WOULD YOU LIKE T
    O PRINT ANOTHER?"
140 K$=INKEY$:IF K$=""THEN 140
150 IF K$(">")"Y" THEN END
160 CLS:LINE INPUT"DISK NAME ";N
    $:GOTO80
```

"SUBMISSION" as at 21.1.87

```
CP      BAS  0 B 1
DIRECT  PRT  0 B 1
DIRECT 2  PRT  0 A 1
DIRECT 3  PRT  0 B 1
DIRECT 4  PRT  0 B 1
DIRECT 5  PRT  0 B 1
DIRECT 6  PRT  0 B 1
DIRECT 7  PRT  0 B 1
DIRECT 8  PRT  0 B 1
DIRECT 9  PRT  0 B 1
DIRECT10 PRT  0 B 1
DIRECT11 PRT  0 B 1
DIRECT12 PRT  0 B 1
DIRECT13 PRT  0 B 1
DIRECT14 PRT  0 B 1
DIRECT15 PRT  0 B 1
DIRECT16 PRT  0 B 1
DIRECT17 PRT  0 B 1
DIRECT18 PRT  0 B 1
DIRECT19 PRT  0 B 1
DIRECT20 PRT  0 B 1
DIRECT21 PRT  0 B 1
DIRECT22 PRT  0 B 1
DIRECT23 PRT  0 B 1
DIRECT24 PRT  0 B 1
DIRECT25 PRT  0 B 1
DIRECT26 PRT  0 B 1
DIRECT27 PRT  0 B 1
DIRECT28 PRT  0 B 1
DIRECT29 PRT  0 B 1
DIRECT30 PRT  0 B 1
DIRECT31 PRT  0 B 1
DIRECT32 PRT  0 B 1
DIRECT33 PRT  0 B 1
DIRECT34 PRT  0 B 1
DIRECT35 PRT  0 B 1
DIRECT36 PRT  0 B 1
DIRECT37 PRT  0 B 1
DIRECT38 PRT  0 B 1
DIRECT39 PRT  0 B 1
DIRECT40 PRT  0 B 1
DIRECT41 PRT  0 B 1
DIRECT42 PRT  0 B 1
DIRECT43 PRT  0 B 1
DIRECT44 PRT  0 B 1
DIRECT45 PRT  0 B 1
DIRECT46 PRT  0 B 1
DIRECT47 PRT  0 B 1
DIRECT48 PRT  0 B 1
DIRECT49 PRT  0 B 1
DIRECT50 PRT  0 B 1
DIRECT51 PRT  0 B 1
```

11 GRANULES FREE.

SHORTHAND KEY UTILITY

UTILITY

16K ECB + EDTASM

by Bernard Besasparis

IN COCO OCTOBER 1985 you published a utility by John Carmichael called TIMESAVE.

For keyboard entry of long programs I personally felt the most useful feature was single keying of BASIC words so while retaining the core structure I excised all the other options and expanded the single key repertoire.

By employing all the alphanumeric and numeric keys it was possible in incorporate 72 words which covers most of the BASIC vocabulary.

Pressing either the UP or DOWN arrows activates the routine and prints out on the screen the full word according to the next key press.

It takes a while to memorize all the necessary keys but I have tried to use logical associations to make it easier.

Often the letter represents the initial of the word in and in some cases the next sequential letter gives something closely related.

Thus 'G' is 'GOTO' and 'H' is 'GOSUB'.

Similarly 'P' is 'POKE' and 'Q' is 'PEEK'.

Various other mnemonic tricks were employed but users could probably figure out their own system of minimising the learning pain.

Anyway, I am convinced it's worth the effort. It takes only seconds to load and execute from tape, and save umpteen key-strokes not to mention syntax errors from incorrectly spelled longer commands.

The machine language installs and protects itself as per John's original program regardless of memory size and occupies 628 bytes.

The assembler listing is created with EDTASM+.

To make copies do a '(C)LOADM"KEY1"', then:

CSAVE"KEY1", &H600; &H85F, &H8B6

Thereafter it's a matter of CLOADM and EXEC.

A full chart of what each key does is listed below.

Down Arrow Commands

Press <down-arrow> followed by corresponding letter to get appropriate command.

Let: Command:

A AUDIOON
B MOTORON
C CIRCLE(
D DATA
E ELSE
F FOR
G GOTO
H GOSUB
I INKEYS
J JOYSTK(
K RIGHT\$(
L LEFT\$(
M MID\$(
N NEXT
O CLS
P POKE
Q PEEK(
R RETURN
S STR\$(
T THEN
U IF
V VAL(
W PRINT@
X LINE(
Y PSET(
Z RND(
1 CSAVE"
2 CLOAD"
3 SKIPF"
4 CSAVEM"
5 CLOADM"
6 TRON
7 TROFF
8 RUN
9 LIST
0 EDIT

Up Arrow Commands

Press <up-arrow> followed by corresponding letter to get appropriate command.

Let: Command:

A ASC(
B ABS(
C CHR\$(
D DRAW"

E EXEC
F FIX
G GET(
H PUT(
I INPUT
J INPUT#-1,
K INSTR(
L LINEINPUT
M PMODE
N SOUND
O OPEN"
P PRINT#-2,
Q PRINT#-1,
R READ
S SCREEN
T PLAY"
U USR
V VARPTR(
W PAINT(
X COLOR
Y PRESET(
Z STRINGS(
1 INT(
2 LEN(
3 AND
4 SQR(
5 SGN
6 HEX\$(
7 DEFUSR
8 DIM
9 CLOSE#-1
0 CLEAR

The Listing:

00100	TABPTR	EQU	\$0
00110	ORG		\$600
00120	START	JSR	\$8CF1
00130	TST		\$6F
00140	BEQ		OK
00150	RTS		
00160	OK	LEAS	2,S
00170	CLR		\$70
00180	RUN	PSHS	X,B
00190	AGAIN	JSR	\$A199

00200	JSR	\$A1CB	00930	FCB	5+189	01660	FCC	/RUN/	
00210	BEQ	AGAIN	00940	FCB	4+197	01670	FCB	0	
00220	CMPA	#10	00950	FCB	3+202	01680	FCC	/LIST/	
00230	BNE	UP	00960	FCB	2+208	01690	FCB	0	
00240	LEAU	TABLE, PCR	00970	FCB	1+212	01700	TABLE2	FCB	36
00250	STU	<TABPTR	00980	FCC	/AUDIOON/	01710	FCB	35+5	
00260	BRA	GETKEY	00990	FCB	0	01720	FCB	34+10	
00270	UP	CMPA	#94	01000	FCC	/MOTORON/	01730	FCB	33+16
00280	BNE	OUT	01010	FCB	0	01740	FCB	32+22	
00290	LEAU	TABLE2, PCR	01020	FCC	/CIRCLE(/	01750	FCB	31+27	
00300	STU	<TABPTR	01030	FCB	0	01760	FCB	30+31	
00310	BRA	GETKEY	01040	FCC	/DATA/	01770	FCB	29+36	
00320	OUT	JMP	\$A1B9	01050	FCB	0	FCB	28+41	
00330	GETKEY	JSR	\$A199	01060	FCC	/ELSE/	01790	FCB	27+47
00340	JSR	\$A1CB	01070	FCB	0	01800	FCB	26+57	
00350	BEQ	GETKEY	01080	FCC	/FOR/	01810	FCB	25+64	
00360	CMPA	#48	01090	FCB	0	01820	FCB	24+74	
00370	BLO	AGAIN	01100	FCC	/GOTO/	01830	FCB	23+80	
00380	CMPA	#57	01110	FCB	0	01840	FCB	22+86	
00390	BHI	NEXT	01120	FCC	/GOSUB/	01850	FCB	21+92	
00400	ADDA	#43	01130	FCB	0	01860	FCB	20+102	
00410	BRA	PROC	01140	FCC	/INKEY\$/	01870	FCB	19+112	
00420	NEXT	CMPA	#65	01150	FCB	0	FCB	18+117	
00430	BLO	AGAIN	01160	FCC	/JOYSTK(/	01890	FCB	17+124	
00440	CMPA	#90	01170	FCB	0	01900	FCB	16+130	
00450	BHI	AGAIN	01180	FCC	/RIGHT\$/	01910	FCB	15+134	
00460	PROC	SUBA	#65	01190	FCB	0	FCB	14+142	
00470	LDU	<TABPTR	01200	FCC	/LEFT\$/	01930	FCB	13+149	
00480	LEAU	A, U	01210	FCB	0	01940	FCB	12+155	
00490	LDB	, U	01220	FCC	/MID\$/	01950	FCB	11+163	
00500	CMPB	#128	01230	FCB	0	01960	FCB	10+172	
00510	BLO	NEXT1	01240	FCC	/NEXT/	01970	FCB	9+178	
00520	SUBB	#128	01250	FCB	0	01980	FCB	8+183	
00530	LEAU	128, U	01260	FCC	/CLS/	01990	FCB	7+188	
00540	NEXT1	LEAU	B, U	01270	FCB	0	FCB	6+192	
00550	PULS	X, B	01280	FCC	/POKE/	02010	FCB	5+197	
00560	PRINT	LDA	, U+	01290	FCB	0	FCB	4+202	
00570	BEQ	RUN	01300	FCC	/PEEK(/	02030	FCB	3+208	
00580	STA	, X+	01310	FCB	0	02040	FCB	2+215	
00590	INCB		01320	FCC	/RETURN/	02050	FCB	1+219	
00600	JSR	\$A282	01330	FCB	0	02060	FCC	/ASC(/	
00610	BRA	PRINT	01340	FCC	/STR\$/	02070	FCB	0	
00620	TABLE	FCB	36	01350	FCB	0	FCC	/ABS(/	
00630	FCB	35+8	01360	FCC	/THEN/	02090	FCB	0	
00640	FCB	34+16	01370	FCB	0	02100	FCC	/CHR\$/	
00650	FCB	33+24	01380	FCC	/IF/	02110	FCB	0	
00660	FCB	32+29	01390	FCB	0	02120	FCC	/DRAW"/	
00670	FCB	31+34	01400	FCC	/VAL(/	02130	FCB	0	
00680	FCB	30+38	01410	FCB	0	02140	FCC	/EXEC/	
00690	FCB	29+43	01420	FCC	/PRINT@/	02150	FCB	0	
00700	FCB	28+49	01430	FCB	0	02160	FCC	/FIX/	
00710	FCB	27+56	01440	FCC	/LINE(/	02170	FCB	0	
00720	FCB	26+64	01450	FCB	0	02180	FCC	/GET(/	
00730	FCB	25+72	01460	FCC	/PSET(/	02190	FCB	0	
00740	FCB	24+79	01470	FCB	0	02200	FCC	/PUT(/	
00750	FCB	23+85	01480	FCC	/RND(/	02210	FCB	0	
00760	FCB	22+90	01490	FCB	0	02220	FCC	/INPUT/	
00770	FCB	21+94	01500	FCC	/EDIT/	02230	FCB	0	
00780	FCB	20+99	01510	FCB	0	02240	FCC	/INPUT#-1,	
00790	FCB	19+105	01520	FCC	/CSAVE"/	/			
00800	FCB	18+112	01530	FCB	0	02250	FCB	0	
00810	FCB	17+118	01540	FCC	/CLOAD"/	02260	FCC	/INSTR(/	
00820	FCB	16+123	01550	FCB	0	02270	FCB	0	
00830	FCB	15+126	01560	FCC	/SKIP"/	02280	FCC	/LINEINPUT	
00840	FCB	14+131	01570	FCB	0	/			
00850	FCB	13+138	01580	FCC	/CSAVEM"/	02290	FCB	0	
00860	FCB	12+144	01590	FCB	0	02300	FCC	/PMODE/	
00870	FCB	11+150	01600	FCC	/CLOADM"/	02310	FCB	0	
00880	FCB	10+155	01610	FCB	0	02320	FCC	/SOUND/	
00890	FCB	9+160	01620	FCC	/TRON/	02330	FCB	0	
00900	FCB	8+167	01630	FCB	0	02340	FCC	/OPEN"/	
00910	FCB	7+174	01640	FCC	/TROFF/	02350	FCB	0	
00920	FCB	6+181	01650	FCB	0	02360	FCC	/PRINT#-2,	

```

/
02370 FCB 0
02380 FCC /PRINT#-1,
/
02390 FCB 0
02400 FCC /READ/
02410 FCB 0
02420 FCC /SCREEN/
02430 FCB 0
02440 FCC /PLAY"/
02450 FCB 0
02460 FCC /USR/
02470 FCB 0
02480 FCC /VARPTR(/
02490 FCB 0
02500 FCC /PAINT(/
02510 FCB 0
02520 FCC /COLOR/
02530 FCB 0
02540 FCC /PRESET(/
02550 FCB 0
02560 FCC /STRING$(/
02570 FCB 0
02580 FCC /CLEAR/
02590 FCB 0
02600 FCC /INT(/
02610 FCB 0
02620 FCC /LEN(/
02630 FCB 0
02640 FCC /AND/
02650 FCB 0
02660 FCC /SQR(/
02670 FCB 0
02680 FCC /SGN(/
02690 FCB 0
02700 FCC /HEX$(/
02710 FCB 0
02720 FCC /DEFUSR/
02730 FCB 0
02740 FCC /DIM/
02750 FCB 0
02760 FCC /CLOSE#-1/
02770 FCB 0
02780 MESS FCC /****SINGL
E KEY COMMAND ENTRY*****PROGR
AM IS NOW INSTALLED****/
02790 FDB $0D00
02800 GO LDX $21
02810 LEAX -628,X
02820 STX $21
02830 LDX $27
02840 LEAX -628,X
02850 STX $27
02860 STX $23
02870 LDX ,S
02880 LDS $21
02890 PSHS X
02900 LDX $27
02910 LEAX 2,X
02920 LDU #START
02930 MOVE LDA ,J+
02940 STA ,X+
02950 CPU #MESS
02960 BNE MOVE
02970 LDA #$7E
02980 STA $16A
02990 LDD $27
03000 ADDD #2
03010 STD $16B
03020 JSR $A928
03030 LDX #MESS-1
03040 JMP $B99C
03050 END

```

EXTENDING COLOUR BASIC

GET & PUT

UTILITY 16K CB by George Viera

FOR THOSE few people who still have Colour BASIC and would like to expand their computer's abilities in the form of extra commands and the like, I have the solution.

In the next few months of this magazine will be six programs to increase the standard colour BASIC computer's abilities, plus one demonstration program.

These six extra programs include new functions such as "GET" and "PUT", "LINE", "PMODE", "PLAY", "DRAW" and "CIRCLE".

The demonstration program is a 3D graphics program. Here is a short explanation of the "GET" and "PUT".

This is quite easy to operate and requires that you have your drawing (or whatever) in the text screen graphics supplied by the ROM in the computer, ie, text graphics.

The computer will ask you for the start of the part of screen you want to "get", ie the x1 and y1 of the top left rectangle and the bottom right x2 and y2 location. It is advisable that you put all four locations in the one line separated by a comma.

For this example, type in "0,0,63,11".

The next question asked is where do you want to put this rectangle. It is important that you keep the same distances between the x1 and x2 values and the y1 and y2 values, otherwise you'll get a mutation of whatever you "got".

For this example, type "0,10,63,11".

It will go ahead and do it's thing and reproduce the rectangle.

However, there are a few conditions set down for this utility.

The area of your rectangle must not go over 128. In our example we are using the extreme length possible. To figure out

the area of your rectangle, use this formulae:

$$(x2-x1+1)*(y2-y1+1)$$

If it equals 128, then that's the maximum you can go with your 'GET' rectangle. You can, if you want, expand this, but with modification to the program.

This program has not been 'idiot-proofed'. If when PUTting the rectangle goes over the screen border, then an FC error will result.

Coming next month: "LINE"

The Listing:

```

0 GOTO10
1 '***** GET-PUT EXAMPLE *****
   ***** GEORGE VIERA *****
3 SAVE"212:3":END'9
10 CLEAR1000
15 NM=1
20 CLS(0):FORT=0T031:PRINT@T,CHR
$(RND(127)+128);:NEXT
25 PRINT@448,"GET X1:Y1:X2:Y2";:
INPUTX1,Y1,X2,Y2
30 FORY=Y1 TO Y2
35 FORX=X1 TO X2
40 GP$(NM)=GP$(NM)+CHR$(48+POINT
(X,Y))
45 NEXTX,Y
50 PRINT@448,"":PRINT@448,"PUT X
1:Y1:X2:Y2";:INPUTX1,Y1,X2,Y2
55 N=0:FORY=Y1 TO Y2
60 FORX=X1 TO X2:N=N+1
65 IF MIDS(GP$(NM),N,1)="0" THEN
75
70 SET(X,Y,VAL(MIDS(GP$(NM),N,1)
))
75 NEXTX,Y
80 PRINT@448,"PRESS ANY KEY":EXE
C44539:RUN

```



...and now for something completely different.

ARTIST

GRAPHICS

by Nigel Fredericks

HERE IS MY very first program that I've sent in for the CoCo. This program creates a different way to show the palette colours on the new CoCo 3. Hope you like it!

The Listing:

```

1 ****Hi Res Graphics Demo****
  *****Nigel Fredericks*****
  *****6/1/87*****
2 GOTO 10
3 SAVE"171:3":END'6
10 ON BRK GOTO 890
20 WIDTH 40:PALETTE CMP
30 HSCREEN 2
40 PALETTE 2,56
50 PALETTE 3,12
60 PALETTE 4,15
70 PALETTE 5,36
80 PALETTE 6,40
90 PALETTE 7,60
100 PALETTE 8,15
110 PALETTE 9,0
120 PALETTE 10,64
130 PALETTE 11,53
140 HCLS 2
150 HCOLOR 3,2
160 HCIRCLE(128,106),87
170 HPAINT(128,106),3,3
180 HCOLOR 9,2
190 HLINE(189,192)-(190,147),PSE
T
200 HLINE(190,148)-(230,148),PSE
T
210 HLINE(208,192)-(209,158),PSE
T
220 HLINE(225,192)-(232,176),PSE
T
230 HLINE-(208,156),PSET
240 HLINE(243,192)-(251,175),PSE
T
250 HLINE-(228,147),PSET
260 HPAINT(191,191),11,9
270 HCOLOR 4,2
280 HLINE(188,143)-(230,147),PSE
T,B
290 HPAINT(189,144),4,4
300 HCOLOR 5,2
310 HLINE(229,143)-(230,92),PSET
320 HLINE(229,143)-(188,143),PSE
T
330 HLINE(189,143)-(188,92),PSET
340 HLINE-(165,112),PSET
350 HLINE-(144,83),PSET
360 HLINE-(141,77),PSET
370 HLINE-(138,72),PSET
380 HLINE-(140,71),PSET

```

```

390 HLINE-(143,76),PSET
400 HDRAW"BM 143,76;R1U7R2D7R1U6
R2D6R1U5R2D5"
410 HLINE(152,80)-(142,80),PSET
420 HLINE(152,76)-(152,80),PSET
430 HPAINT(148,78),7,5
440 HLINE-(166,94),PSET
450 HLINE-(184,80),PSET
460 HLINE-(231,80),PSET
470 HCIRCLE(285,85),30,,.5
480 HLINE(231,80)-(256,92),PSET
490 HLINE-(260,92),PSET
500 HLINE(230,92)-(254,105),PSET
510 HLINE-(284,100),PSET
520 HPAINT(210,140),5,5
530 HPAINT(144,76),7,5
540 HCOLOR 7,2
550 HCIRCLE(207,65),16
560 HLINE(192,60)-(187,64),PSET
570 HLINE-(191,64),PSET
580 HLINE(197,56)-(210,60),PSET
590 HLINE-(210,79),PSET
600 HPAINT(208,62),7,7
610 HCOLOR 6,2
620 HLINE(198,58)-(191,61),PSET,
BF
630 HLINE(191,59)-(208,60),PSET
640 HPAINT(212,60),9,7
650 HLINE(194,74)-(198,72),PSET
660 HCOLOR 8,2
670 HPAINT(285,85),1,5
680 HCOLOR 10,2
690 HDRAW"BM 152,76;F10U4H20L4F6
H7;C1;L3U7F5"
700 HCOLOR 9,2
710 HLINE(244,192)-(280,130),PSE
T
720 HLINE-(288,130),PSET
730 HLINE-(320,172),PSET
740 HLINE(252,192)-(284,138),PSE
T
750 HLINE-(320,180),PSET
760 HLINE(265,168)-(309,168),PSE
T
770 HDRAW"C9;BM130,140;H8L10D4L6
U2G14D8F8R4E8R8F8R6E12U8H16D12
780 HDRAW"BM130,168;R6G3H3"
790 HPAINT(125,150),10,9
800 HPRINT(5,1),"The World Is Ju
st The Beginning..."
810 HPRINT(29,14),"With Coco 3"
820 HPRINT(30,15),"Graphics"
830 HPAINT(245,191),10,9
840 FOR I=0 TO 63:PALETTE 1,I
850 PLAY"T803;CCDDAFFGGEBBADCCE
"
860 PLAY"T1002;CBBAGFABP3BEFGG
FEADCE"
870 NEXT I
880 EXEC44539
890 PALETTE CMP:ATTR 2,0:END

```

It's a 3D...

DISH

32K ECB

GRAPHICS by Craig Stewart

DISH IS AN INCREDIBLY simple graphics program that produces three dimensional hidden line plots of folded dish shaped objects. Just follow instructions and it will slowly plot out a dish. It will take quite a while, and even longer if greater detail is wanted.

The Listing:

```

1 GOTO 5
2 ****DISH DRAWER****
3 SAVE"146F:3":END'6
4 END
5 PMODE 4,1
10 CLS:PRINT@10,"DISH DRAWER"
20 PRINT:PRINT"1. BUFF ON BLACK"
:PRINT"2. BLACK ON BUFF"
30 AS=INKEY$:IFAS("<")"1" AND AS(">")
2" THEN 30
40 IF AS="1" THEN PCLS:CO=1 ELSE
PCLS5:CO=0
50 PRINT:INPUT"INPUT AMOUNT OF F
OLDING. 1 VERY LOW - 35 VERY H
IGH ";R2:PRINT:INPUT"INPUT DEGRE
E OF DETAIL. 1 VERY HIGH - >5
0 VERY LOW ";R1
60 SCREEN1,1:XS=1:YS=1:POKE65495
,0
70 A=128:B=A*A:C=96
80 FORX=0TOA STEPXS:S=X*X:P=SQR(C
B-S)
90 FORI=-P TOP STEPRI*YS
100 R=SQR(S+I*I)/A
110 Q=(R-1)*SIN(R2*R)
120 Y=I/3+Q*C
130 IFI=-P THENM=Y:GOTO160
140 IFY>M THENM=Y:GOTO170
150 IFY>N GOTO180
160 N=Y
170 PSET(-X+128,98-Y,CO):PSET(X+
128,98-Y,CO)
180 NEXT:NEXT
190 GOTO190

```

To find the start and end address of a BASIC program, use the following PEEKS:

Start address:
PEEK(&H1B)*256+PEEK(&H1A)

End address:
PEEK(&H1B)*256+PEEK(&H1C)

Access extra goodies.

TEST VDG

16K White case CoCo with 3 slanted stripes in the middle

by Bernard Besasparis

THIS IS FOR THE LATEST 64K CoCo series B and was inspired by the article by Tony DiStefano in Rainbow November 1986.

Not being into hardware I wanted a software fix to access those extra goodies of true lower case, clear border and inverse video in text mode which Tandy had so sneakily deprived me of.

Furthermore, I ruled out the messy business of doing POKES after every print as a practical solution.

A quick visit into ROM led me to the conclusion that the main villain was a 'ANDA #7' instruction in the print routine.

This piece of vandalism deliberately resets the top five bits of \$FF22 to zero each time and disables the aforementioned goodies.

By entering 64K RAM mode I could change the #7 located at \$95C9 to any value and thus preserve the bits that led the VDG chip strut its stuff.

To test this theory I wrote a program which gave me a choice of every possible combination of text screen, including a dummy print routine to prove each setting survived its journey through ROM processing.

It seems to work fine.

In addition, it solves the SCREEN 0,1 problem since bit 4 which controls the colour set can also be salvaged.

There is one slight hitch. Any usage of graphic modes will scramble \$FF22 and I haven't had the courage to sort that lot out yet.

However, this can be overcome by saving the current \$FF22 value and POKING it back immediately prior to returning to the text screen. This will ensure that you get the same

screen configuration you exited from if so required.

The experimental program is a bit messy because I used direct addition and subtraction to alter to alter the bytes rather than OR/AND thus necessitating all those switches.

In normal circumstances all that's needed is to decide what sort of text screen you want, POKE the appropriate values to \$95C9 and \$FF22 then forget about it. To summarize, the text screen is controlled by \$FF22 in the following manner:-

Bit 3 set (binary 8)
= orange screen
Bit 4 set (binary 16)
= lower case
Bit 5 set (binary 32)
= inverse video
Bit 6 set (binary 64)
= clear border

Simply add up the binary components needed and poke what you peek plus this total.

It should clearly understood that POKING to \$95C9 simply allows safe passage of the selected bits while POKING to \$FF22 actually turns on the bits and modifies the VDG accordingly.

Also, if using the direct addition method make sure to set \$95C9 and \$FF22 to 7 at the beginning of the program otherwise if you RUN it again the values will accumulate producing some interesting graphics effects but a totally useless text screen.

The program is self-explanatory presenting two menus. The first turns the options on, the other turns them off.

The bottom left of each screen in the demonstration displays the current value of \$FF22.

The Listing:

```
0 GOTO10
1 '***** TEST VDG *****
   **** BERNARD BESASPARIS ****
3 SAVE"219:3":END*8
10 K=&H95C9:X=7:R=&HFF22
20 REM**ALL 64K RAM MODE**
30 DATA1A,10,8E,80,00,B7,FF,DE,A
   6,80,B7,FF,DF,A7,1F,8C,E0,00,26,
   F1,39
40 FORA=&H3DD TO &H3F1
50 READA$:POKEA,VAL("&H"+A$):NEX
   T
60 EXEC&H3DD
70 REM**MAIN MENU**
80 CLS:PRINT@42,"test options":P
   RINT:PRINT
90 PRINT"1 CLEAR BORDER"
100 PRINT"2 LOWER CASE"
110 PRINT"3 INVERSE"
120 PRINT"4 SCREEN0,1 ENABLE"
130 PRINT"5 GRAPHICS (ANY KEY TO
   EXIT)"
140 PRINT"6 EXECUTE TEST PRINT"
150 PRINT"7 CANCEL OPTIONS"
160 PRINT"8 ORANGE SCREEN"
170 PRINT@480,PEEK(&HFF22);
180 A$=INKEY$:IFA$=""GOTO180
190 ONVAL(A$)GOTO200,210,220,230
   ,250,300,340,480
200 IFS1=1THEN180ELSE$1=1:X=X+64
   :POKEK,X:POKER,PEEK(R)+64:GOTO80
210 IFS2=1THEN180ELSE$2=1:X=X+16
   :POKEK,X:POKER,PEEK(R)+16:GOTO80
220 IFS3=1THEN180ELSE$3=1:X=X+32
   :POKEK,X:POKER,PEEK(R)+32:GOTO80
230 IFS4=1THEN180ELSE$4=1:X=X+8:
   POKER,X:GOTO80
240 REM**DO GRAPHICS**
250 S=PEEK(R)
260 PMODEA:SCREEN1,1:PCLS
270 CIRCLE(128,96),95
280 A$=INKEY$:IFA$=""THEN280ELSE
   POKER,S:GOTO80
290 REM**PRINT ROUTINE**
300 CLS:FORA=1TO10:PRINT"Print 1
   ots of stuff":NEXT:PRINT
310 PRINT"ANY KEY TO EXIT":;PRIN
   T@480,PEEK(&HFF22);
320 A$=INKEY$:IFA$=""THEN320ELSE
   80
330 REM**ALTERNATE MENU**
340 CLS:PRINT@41,"cancel options
   ":PRINT:PRINT
350 PRINT"1 CLEAR BORDER"
360 PRINT"2 LOWER CASE"
```

continued overpage

USING DATA STRUCTURES in FORTH

Languages

by John Redmond

THE LAST TWO Forth articles have had no real working programs in them. This has been a response to criticisms that earlier contributions (with programs) have been too advanced.

I have accepted the criticisms and I am trying to mend my ways, but I challenge you to change yours.

We programmers are a strange lot - forever complaining about our lot in life.

'The CoCo doesn't have enough memory, 0.9 MHz is too slow, the graphics are boring, etc., etc.'

I want to take a lot of you (us) to task on this.

Most of us have double computing standards: those we expect in programs we want to use, and those we use when we program ourselves.

To be blunt, anyone who programs with a BASIC interpreter has no right to complain about the speed of a computer.

His/her program could run up to twenty times faster if an efficient language had been used. And, what's more, the

program could be shorter and easier to understand and maintain.

The speed deficiency of Colour BASIC is something we all know about. Otherwise why all the published programs that consist of pages of data statements for machine language pokes?

To state the obvious, these are unreadable and do almost nothing to educate the readers to anything better than BASIC. To get Rainbow on Tape, or somesuch, is the only way to avoid the tedium of trying to type it all in, but that is just skirting around the problem.

Such practices are just bandaids.

At the other end, assembly programmers sometimes write some

magnificent stuff - tight, fast code. But, really, you need to be a Dougan or a Wright to write any non-trivial program in assembly.

And the debugging and maintenance are horrific.

This is why C has been greeted with such enthusiasm by software developers. The code is much easier to write than assembly, and it is (reputedly) portable, but the code size can be monstrous.

The only thing that is saving C's bacon is the rapid drop in the price of memory.

And then we find that the disks do not have the capacity and take a long time to load programs. Ever watched a Mac load from a floppy? But, of course, you can always get a hard disk.

God bless the C language. End of diatribe (I am not going to say what I would like to say about the way that operating systems have been developing).

If you have persisted with me to this point, I will reward you with some snippets of Forth code which, I hope, will give you some further feel for the special style of the language.

Say you are writing a

full-screen editor and you want to write some messages at particular places on the screen.

You need, therefore, to control the cursor position. Most reasonable Forth packages will have a word to do this. It may be called TAB or AT, and it will expect on the stack two numbers to indicate row and column positions.

The order may vary, so be careful. The Forth-83 standard E*FORTH expects column, then row, so that 0 3 TAB will place the cursor at the start of the fourth line (line 3!).

Perhaps, in the program, you want to type some prompts on line 3. You can insert 0 3 TAB each time you need it. But this takes up space after compilation - and it takes up space in the source program. And it is not clear what its function is. Let's make it clearer, then.

```
: PROMPT 0 3 TAB ;
```

PROMPT points, of course, to the correct part of the screen and it is the first small step to refinement of our cursor control. The program will issue a number of prompts, all at the same screen position, and each will consist of a short text string.

Forget TAB for the moment, and let us consider how text strings might be compiled. This takes us back two months to the action of that remarkable word, WORD. Remember that it expects a delimiter on the stack and that it will continue to extract characters from the Forth input stream until that delimiter is reached - or until the stream is exhausted.

To make the source code readable, we will use the word ASCII, which returns the next character. For instance, 'A' will return 65. A useful new definition might then be:

```
: " ASCII " WORD C@ 1+ ALLOT ;
```

continued from previous page

```
370 PRINT"3 INVERSE"  
380 PRINT"4 SCREEN0,1 ENABLE"  
390 PRINT"5 EXIT"  
400 PRINT@480, PEEK(&HFF22);  
410 AS=INKEYS: IFAS="" THEN410  
420 ONVAL (AS) GOTO430, 440, 450, 460  
,80  
430 IFS1=0 THEN410 ELSE S1=0: X=X-64  
: POKER, X: POKER, PEEK (R) -64: GOTO41  
0  
440 IFS2=0 THEN410 ELSE S2=0: X=X-16  
: POKER, X: POKER, PEEK (R) -16: GOTO41  
0  
450 IFS3=0 THEN410 ELSE S3=0: X=X-32  
: POKER, X: POKER, PEEK (R) -32: GOTO41  
0  
460 IFS4=0 THEN410 ELSE S4=0: X=X-8:  
POKER, X: POKER, PEEK (R) -PEEK (&HC1)  
: GOTO410  
470 PRINT@480, PEEK (&HFF22);  
480 SCREEN0, 1: GOTO80
```



To verbalize this definition: 'Define a new word (called ") which carries out the following actions - 1. Use the ASCII value of " as delimiter for WORD, which returns the address of a dimensioned string; 2. Fetch the byte value at this address, which corresponds to the number of characters in the string; 3.

Add 1 to this length to get the total length (including the length byte); and 4. Advance the dictionary pointer to ALLOT this number of bytes.

Now, if we use " to fetch a string, without having some way of being able to determine its address at some later stage, we are simply wasting our time. How should we do this? Consider STRING:

```
: STRING CREATE " DOES>
COUNT TYPE ;
```

STRING is a defining word (it uses CREATE) which sets up just the right sort of header which will satisfy us. A typical use might be:

```
STRING REDMOND is a
proselytiser."
```

Note that REDMOND is extracted by CREATE to make a header of the same name, and that the rest of the input (up to ") is extracted by ". When REDMOND is executed, the judgment is printed. NOW back to PROMPT.

Remember that we want to output a number of strings on line 3. We might now code:

```
: BUNGLE PROMPT STRING
Don't be stupid" ;
```

If we try this, E*FORTH will respond DON'T UNKNOWN. This is because we have forgotten when what happens. We can recoup the situation by using a standard Forth word, called ."

```
: BUNGLE PROMPT ." Don't
be stupid." ;
```

This word will do what we wanted and print the insult at the correct place on the screen. In truth, ." is defined rather like STRING; so why did it work while STRING did not? This topic is discussed in detail in Brodie's 'Starting Forth', and it would be too much of a digression to discuss it in detail now.

The brief answer is that ." is an immediate word, which acts DURING compilation of BUNGLE,

while STRING would act when BUNGLE is executed.

The problem of temporal indirection again (see last month). Cheer up! This is very difficult for a beginner to grasp (or it was for me). The real lesson here is that, for intelligent programming in Forth, you have to get inside the language and understand, to some extent, how it works. Chapter 9 of 'Starting Forth' ('Under the Hood') will start you in the right direction.

It is important to distinguish defining words from their definitions and this, partly, is where we have been going wrong. To handle our SET of prompts on line 3, we can use:

```
: PROMPT CREATE " DOES> 0 3 TAB
COUNT TYPE ;
```

Remember that the code between CREATE and DOES> is executed at the time that PROMPT is used to define another word, while the code between DOES> and ; is executed when the word, defined using PROMPT, is itself executed. It's easier with examples:

```
PROMPT BUNGLE Don't be stupid"
PROMPT CAJOLE Please try again"
, etc.
```

NOW, when BUNGLE or CAJOLE is executed, the message is printed on line 3. Each is an example of a class of words defined by PROMPT and, as such, is a very considerable data abstraction. And, by using PROMPT, we have generalized a category of screen messages.

We can go further, of course. In Forth, we can always go further.

What about messages at other spots on the screen? We need to generalize even further. Let us finish this month with MESSAGE:

This is used as:

```
3 5 MESSAGE WARNING Insert disk
first"
```

Execution of WARNING will place the string at column 3 of line 5. This generalization of word categories and data structures is part of the special sophistication of Forth, and pushes well past the data typing of Pascal.

Next month, we will start looking at a quite different-looking type of data structure - the decision array, which is quite unique to Forth as a high(?) level language though, curiously, it is quite common for assembly programmers to use computed jumps, which are a simple example of a decision array.

In closing, Forth offers a great deal to all sorts of serious programmers. To the assembly programmer, it offers much the same level of control of memory and devices, with only a small speed penalty, and it offers to the high level programmer the opportunity to soar with the most sophisticated of algorithms and data structures.

Come on!

Come Forth!

```
MESSAGE CREATE , , ( tab coords)
" ( fetch the string and store it)
DOES> DUP ( save base address)
2@ ( fetch tab coords as a double word)
TAB ( use them to control cursor)
4 + ( bump the address past the 2 words)
COUNT TYPE ( get the string to the screen) ;
```

TEST YOUR SENSES

32K ECB
APPLICATION

by Bob Delbourgo

YOUR SENSES TO you are the most important things in the world to have. You can do anything if you didn't have them. In fact, you might as well be a potatoe or a rock if you didn't.

But wouldn't it be shocking if your senses were failing you slowly? Wouldn't you like to know how your senses are going?

This is the sole objective of this program. It tests you on your eyesight (optometric), colour recognizability (colourimetric), ears (audiometric), and your reactions (reactivity).

Mind you, the results of this program aren't to be taken too seriously and should be more for fun than anything else.

After you complete each test, you will be given a report on how you went.

The Listing:

```
1 CLEAR200,15999
2 FORI=1TO64:READD:POKE15999+I,D
:NEXTI:DEFUSR0=16000
3 CLS8:DIML$(26),E$(4):R=RND(-TIMER)
4 GOSUB121:PRINT@72,"TEST YOUR SENSES";:PRINT@104,"BY BOB DELBOURGO";:PRINT@136,"hobart,australia";
5 PRINT@200,"(1) OPTOMETRIC ";:PRINT@232,"(2) COLORIMETRIC";:PRINT@264,"(3) AUDIOMETRIC ";:PRINT@296,"(4) REACTIVITY ";
6 PRINT@360,"INPUT CHOICE # ?";
7 DATA95,189,169,162,189,169,118,189,179,237,237,141,0,47,230,141,0,45,26,16
8 DATA134,2,183,255,32,141,26,134,254,183,255,32,141,19,236,141,0,23,131,0
9 DATA1,39,6,237,141,0,14,32,227,189,169,116,57,230,141,0,6,92,38,253,57,0,0,0
10 N$="L25501GFEDCBAGFEDCBAGFEDCBAAAA"
11 L$(1)="U8E8F8D8U8L16"
```

```
12 L$(2)="U16R12F4G4L12R12F4G4L12"
13 L$(3)="BR16L12H4U8E4R12"
14 L$(4)="U16R12F4D8G4L12"
15 L$(5)="R16L16U8R8L8U8R16"
16 L$(6)="U8R8L8U8R16"
17 L$(7)="BE8R8D4G4L8H4U8E4R12"
18 L$(8)="U16D8R16U8D16"
19 L$(9)="R16L8U16L8R16"
20 L$(10)="BU4F4R4E4U12R4L16"
21 L$(11)="U16D8R8E8G8F8"
22 L$(12)="R16L16U16"
23 L$(13)="U16F8E8D16"
24 L$(14)="U16F16U16"
25 L$(15)="BU4U8E4R8F4D8G4L8H4"
26 L$(16)="U16R12F4G4L12"
27 L$(17)=L$(15)+"BR8F6"
28 L$(18)=L$(16)+"R8F8"
29 L$(19)="R12E4H4L8H4E4R12"
30 L$(20)="BR8U16L8R16"
31 L$(21)="BE16D12G4L8H4U12"
32 L$(22)="BU16D8F8E8U8"
33 L$(23)="BU16D12F4E4U4D4F4E4U12"
34 L$(24)="E16BD16H16"
35 L$(25)="BR8U8H8F8E8"
36 L$(26)="R16L16E16L16"
37 E$(1)="R16L16U8R16L16U8R16"
38 E$(2)="U16D16R8U16D16R8U16"
39 E$(3)="R16U8L16R16U8L16"
40 E$(4)="U16R8D16U16R8D16"
41 A=VAL(INKEY$):IFA<10RA>4THEN41 ELSEONA GOTO42,49,63,73
42 CLS2:GOSUB121:PRINT@72,"TEST YOUR VISION";
43 PRINT@131,"LETTERS IN EVER-DECREASING";:PRINT@163,"SIZES WILL APPEAR IN ORDER";
44 PRINT@227,"YOU HAVE TWO CHANCES (MAX)";:PRINT@259,"IN WHICH TO IDENTIFY EACH ";:PRINT@291,"LETTTER, IF YOU CAN SEE IT!";:PRINT@323,"TEST ONE EYE AT A TIME AND ";
45 PRINT@355,"RUN THROUGH THE WHOLE CARD";:PRINT@387,"FOR OPTOMETRIC ASSESSMENT.";:PRINT@419,"<L> for letters,<E> for eS";:W=0
46 I$=INKEY$:IFI$="E"THEN88
47 IFI$="L"THEN96
48 GOTO46
49 CLS3:GOSUB121:PRINT@71,"COLOUR VISION TEST";:W=0
50 PRINT@131,"TRY TO IDENTIFY THESE COLORED";:PRINT@163,"LETTERS AS THEY APPEAR ON ";:PRINT@195,"DIFFERENT BACKGROUND COLORS";
```

```
51 PRINT@422,"any inkey to continue";:V=0
52 IFINKEY$=""THEN52
53 PMODE3,1:PCLS:SCREEN1,S:COLOR3,1
54 FORP=16TO196STEP60:FORQ=14TO134STEP60:LINE(P,Q)-(P+40,Q+40),PSET,B:NEXTQ,P
55 FORQ=34TO154STEP60:PAINT(96,Q),2,3:PAINT(156,Q),3,3:PAINT(216,Q),4,3:NEXTQ:SC=6:X=1
56 B=1:SP=25:Y=46:A=2:GOSUB108::A=3:Y=106:GOSUB108:A=4:Y=166:GOSUB108
57 B=2:SP=85:Y=46:A=1:GOSUB108:A=3:Y=106:GOSUB108:A=4:Y=166:GOSUB108
58 B=3:SP=145:A=1:Y=46:GOSUB108:A=2:Y=106:GOSUB108:A=4:Y=166:GOSUB108
59 B=4:SP=205:A=1:Y=46:GOSUB108:A=2:Y=106:GOSUB108:A=3:Y=166:GOSUB108
60 S=S+1:IFS<2THEN53ELSE61
61 W=W/6:SCREEN0,0:PRINT@328,"YOU SCORED";24-W;" /24";
62 IFINKEY$=""THEN62ELSERUN
63 CLS6:GOSUB121:PRINT@74,"HEARING TEST";
64 PRINT@131,"ADJUST THE VOLUME ON YOUR ";:PRINT@163,"T.V. SET UNTIL COMFORTABLE";
65 PRINT@227,"after that press any inkey";
66 IFINKEY$=""THENSOUND100,10:GOTO66
67 PRINT@227,"TONES RUN FROM 128 TO 255.";:PRINT@259,"ENTER A TONE IN THAT RANGE";:PRINT@291,"AND SEE IF YOU CAN HEAR IT";:PRINT@423,"PRESS <R> FOR MENU";
68 PRINT@323,"tone =";:INPUT F
69 POKE1373,223:POKE1374,223:POKE1375,RND(26)
70 IFF<128ORF>255THEN68
71 POKE16063,2*F-255:A=USRO((F-10)*50)
72 PRINT@323," ";:F=ORT=1TO500:NEXTT:IFINKEY$="R"THENR$=68
73 CLS7:GOSUB121:PRINT@73,"REACTION TIME";:G=0
74 PRINT@131,"TWO E'S WILL APPEAR ALMOST";:PRINT@163,"SIMULTANEOUSLY IN 2 BOXES.";:PRINT@195,"PRESS <SPACEBAR> IF THEY ";:PRINT@227,"ARE DIFFERENT, <ENTER> IF
```

```

";
75 PRINT@259,"THEY'RE THE SAME.
AFTER 50";:PRINT@291,"GOES COCO
WILL CALCULATE ";:PRINT@323,"YO
UR MEAN REACTION TIME. ";:PRINT
@355,"YOU INCUR A PENALTY OF TWO
";:PRINT@387,"SECONDS FOR PRESSI
NG THE ";
76 PRINT@419,"WRONG KEY, SO BE C
AREFUL!!";:PRINT@459,"any inkey
";
77 IFINKEY$=""THEN77
78 PMODE1,1:PCLS:SCREEN1,1:COLOR
3,1
79 LINE(50,60)-(110,120),PSET,B:
LINE(140,60)-(200,120),PSET,B
80 G=G+1:IFG>50THEN84ELSER=RND(4
):DRAW"S8C6BM65,105;XE$(R);":S=R
ND(4):DRAW"S8C8BM155,105;XE$(S);
"
81 I$=INKEY$:TIMER=0:IFI$=CHR$(3
2)THENIFR=S THENT=T+2:GOTO78 ELS
E T=T+TIMER/60:GOTO78
82 IFI$=CHR$(13)THENIFR=S THENT=
T+TIMER/60:GOTO78 ELSE T=T+2:GOTO
78
83 GOTO81
84 SCREEN0,0:CLS7:GOSUB121:PRINT
@131,"YOUR AVERAGE REACTION TIME
";
85 PRINT@200,I/50;"SECS";
86 PRINT@427,"any inkey";
87 IFINKEY$=""THEN87ELSERUN
88 PMODE4,1:PCLS1:SCREEN1,0:A=0:
B=1
89 FORX=0TO2:SP=96:SC=16:Y=64:GO
SUB113:NEXT

```

```

90 FORX=0TO5:SP=44:SC=8:Y=110:GO
SUB113:NEXTX
91 FORX=0TO10:SP=24:SC=4:Y=135:G
OSUB113:NEXTX
92 FORX=0TO13:SP=18:SC=3:Y=158:G
OSUB113:NEXTX
93 FORX=0TO20:SP=12:SC=2:Y=175:G
OSUB113:NEXTX
94 FORX=0TO36:SP=7:SC=1:Y=185:GO
SUB113:NEXTX
95 GOTO103
96 PMODE4,1:PCLS1:SCREEN1,1:A=0:
B=1
97 FORX=0TO2:SP=96:SC=16:Y=64:GO
SUB108:NEXT
98 FORX=0TO5:SP=44:SC=8:Y=110:GO
SUB108:NEXTX
99 FORX=0TO10:SP=24:SC=4:Y=135:G
OSUB108:NEXTX
100 FORX=0TO13:SP=18:SC=3:Y=158:
GOSUB108:NEXTX
101 FORX=0TO20:SP=12:SC=2:Y=175:
GOSUB108:NEXTX
102 FORX=0TO36:SP=7:SC=1:Y=185:G
OSUB108:NEXTX
103 CLS:GOSUB121
104 PRINT@67," W<80 - EXCELLENT
EYESIGHT";:PRINT@99," 80<W<160 -
GOOD EYESIGHT";:PRINT@131,"160
<W<240 - NEED GLASSES";:PRINT@1
63,"240<W<320 - POOR EYESIGHT";
::PRINT@195,"320<W<400 - YOU NEE
D HELP!";:PRINT@226," 400<W - SE
E AN OPTOMETRIST";
105 PRINT@292,"YOUR RATING IS V
=";:V;:PRINT@388,"hit the <0> to
continue";

```

```

106 IFINKEY$=""0"THENRUNELSE106
107 GOTO107
108 V=0:R=RND(26)
109 I$=INKEY$:DRAW"S"+STR$(SC)+"
BM"+STR$(X*SP)+"","+STR$(Y)+"";C"+
STR$(A)+"XL$(R);":IFI$=""THEN109
110 I=ASC(I$):IFI<65ORI>90THEN10
9
111 IFI=R+64THENSOUND200,1:RETUR
NELSEW=W+SC:V=V+1:PLAYN$:IFV=2TH
ENRETURN
112 DRAW"S"+STR$(SC)+"BM"+STR$(X
*SP)+"","+STR$(Y)+"";C"+STR$(B)+"X
L$(R);":GOTO109
113 V=0:R=RND(4)
114 I$=INKEY$:DRAW"S"+STR$(SC)+"
BM"+STR$(X*SP)+"","+STR$(Y)+"";C"+
STR$(A)+"XE$(R);":IFI$=""THEN114
115 IFI$="E"THENI=1
116 IFI$="N"THENI=2
117 IFI$="V"THENI=3
118 IFI$="S"THENI=4
119 IFI=R THENSOUND200,1:RETURNE
LSEW=W+SC*2:V=V+1:PLAYN$:IFV=2TH
ENRETURN
120 DRAW"S"+STR$(SC)+"BM"+STR$(X
*SP)+"","+STR$(Y)+"";C"+STR$(B)+"X
E$(R);":GOTO114
121 FORI=1TO32:R=RND(26):POKE102
3+I,R:POKE1536-I,R:NEXTI:FORI=1T
O14:R=RND(26):POKE1024+32*I,R:PO
KE1535-32*I,R:NEXTI:RETURN
199 END
200 SAVE"205E:3":END'1

```



MACHINE LANGUAGE

PROGRAMMING

LESSON FOUR
32K CoCo with EDTASM+

by Malcolm Patrick

CPU STANDS FOR Central Processing Unit. In fact when we talk about the CPU we mean the microprocessor (or the heart) of the computer.

It is the registers that hold the information so that they may be PROCESSED.

In this lesson we will explain how the different parts of the CPU are used to form a program, and any time we use the word CPU you will know that it stands for all those little circuits in the computer going about their business.

Another name for the CPU is the MPU (Microprocessor unit).

ALU

In the heart of the computer there is the ALU (Arithmetic Logic Unit) and it's job is to perform simple calculations such as addition, subtraction, and multiplication.

The idea is to take a number from one register and compare, add, take, times, or divide it with another register.

In BASIC we would use a line like this:

```
IF A = B THEN GOTO xxxx
```

This would cause the ALU to compare A to B and hold the results in a particular register then allow the computer to go about its business.

The ALU works on a one BYTE size. That is, it will calculate numbers held in any 8 bits at any one time.

There seems to be a lot of computer jargon being thrown about in this lesson but there is no need to panic.

You must have some idea of their functions because we will be using their terms quite regularly in the future.

REGISTERS

A register is a place where information is temporary placed,

held, or stored for the purpose of adding, taking or comparing.

This lesson we will explain the use of some of the 10 registers used by the 6809 microprocessor (MPU).

The A and B registers are the ACCUMULATOR registers and are one byte in size; that means they hold only 8 BITS or one BITE of information at any given time, and therefore the highest number that could be stored in these registers would be \$FF (notice it only uses hex numbers).

Their purpose is to temporarily hold information so that the ALU can calculate the next move.

The X and Y registers are the INDEX registers and are two BYTES in size. The reason we call them the INDEX registers is because they are used to point to a location in memory just as you would use your index finger to point to some object.

If you look at line 120 of the listing DAYTHREE you will see this:

```
LDX #5400
```

This means "load the X register with 400" while line 170 means ...

```
CMPA ,X
```

... to "compare the A register with the X register".

REGISTERS & ZBUG

It is easy to check on the registers in ZBUG. Load the listing called DAYTHREE.

Assemble it, then go into ZBUG.

Now we can check what is happening by stepping through the program with the comma "," command.

Type 'BUBSRT,' and it will answer with the first command ..

```
BUBSRT, LDX #400
```

You can now look at the registers with the "R" command.

Press the letter "R" and take particular notice of the A,B,X AND Y registers. What we have done so far is loaded DAYTHREE into the EDTASM assembler.

```
1 = A/IM  
2 = Z  
3 = BUBSRT,  
4 = R
```

You will see that there is nothing at this stage in any of the 4 registers even though the command is to Load the X register with #400.

Step through to the next instruction with the comma "," and you will see ...

```
BUB010+3 LDY #0
```

Look at the registers again by pressing "R" ... you will see that the X register is now loaded with the value of 400.

Press the comma twice and look at the "A" register; it is now loaded with a hex number of the first letter that was on your screen at the time.

The next command in the program is:

```
CMPA ,X
```

... which stands for "compare the A register with the X register".

This is where the ALU comes into function. The reason we need to know about the registers is because all information sent to the microprocessor must pass through one or all of these registers.

The next register we will look at is the "D" register.

You will not see this register when you look in zbug because the D register is in actual fact the A and B register joined together to make a two byte register.

We cannot load the A register with the contents from the X

register because the X register is two bytes long and the A register is one byte in length. But we can load the X register into the D register or load the D register into the X register.

"LESS4"

Here is a BASIC program to do the same thing as the machine program we have been running.

Type it in and run it.

Notice how slow it is compared to machine language. The line numbers are similar to the program called DAYTHREE.

Type:

csave"LESS4/BAS"

The listing:

```
120 X=&H400
'point X to start of screen
140 A = PEEK(X)
'look at X and store in A
145 X=X+1
'increase X
146 B=PEEK(X)
'look at X and store in B
160 IF B =>A GOTO 250
'compare A to B go if
equal to or greater than
180 X=X-1:POKE X,B
'put smaller one first
190 X=X+1:POKE X,A
'put larger letter
220 GOTO 120
'go check from start again
250 IF X< 1524 THEN GOTO 140
'is it the end of screen
290 END
```

"PC" - Program Counter Register

As you drive your car down the road the speedometer ticks to keep track of the mileage so too the computer has a register to keep track of where you are at all times.

As you step through the listing you can look at the PC register and see it change as it jumps from line to line.

This number corresponds to the Address in the source code listing.

All the other registers we will leave for a later chapter.

USING ZBUG TO CALCULATE DECIMAL TO HEX

While in zbug there is an easy way to change a decimal number to hexadecimal.

Go into zbug and try this command:

```
110
(rem input base 10: decimal)
```

```
O16
(rem output base 16:
hexadecimal)
```

```
400=
(rem answer to $400 will
be #1024)
```

Notice the (T) will always follow a decimal number.

Now change the input and output:

```
116
(rem input base 16:
hexadecimal)
```

```
O10
(rem Output base 10: decimal)
```

```
1024=
(rem answer of 1024 decimal
will be $400)
```

Take time now to read chapter ONE and chapter TWO of your Radio Shack Assembly Language Programming Book.

TRANSFERRING DATA TO REGISTERS

The CPU registers are like other memory locations in that they can be loaded with data values.

To move and calculate this data is no great feat. First you will have to learn what some of the Mnemonic (source listings) Instructions.

Turn to page 26 of your Radio Shack book and look at figure 3-2.

In line 110 of the mnemonics OPCODE you see "LDA". This means "load the A register."

Next down is "load the B register".

Next down is "load the Y register".

In fact they are all loading different registers upto and including line 170.

Type in the Mnemonics of figure 3-2 and assemble it. Then go to ZBUG and step through the lines with the COMMA command starting at START, looking at the registers as you go with the "R" command.

You will see that the A register is the first loaded with "37". As you will remember that the registers only hold hexadecimal numbers and on line 110 we loaded A with #55 (decimal).

As you go through check the registers with the source code operand. Can you see the corresponding numbers?

Go back and assemble it again and when in zbug set break point

at loop "XLOOP" and go at start "GSTART".

Look again at the registers. The last one - "LDD #1000" - was not executed.

Until your next lesson play around with the loading of registers.

SOME OTHER HINTS WHILST EDITING

If many lines need to be deleted from the mnemonic listing it is much easier to use the "D" command.

"D100:200": delete 100 to 200.

"D#:210": delete from start to 210.

"D500:*": delete from 500 to end of the listing.

Notice the ":" between the numbers to be deleted.

Say you wanted to repeat lines 100 to 200 and starting from line 1000. This could be done with the "C" command.

"C1000,100:200,10"

So we start at line 1000, copy 100 to 200 and increase the line numbers by 10.

Now let's say you wanted to find where the label called START branched from - you could find it with the "F" command.

FSTART would find the word start for you. To find the next one in the mnemonics just enter "F".

QUESTIONNAIRE FOR LESSON FOUR

Q1: Where does the display screen start & end. Answer in both hexadecimal & decimal?

A1:
Q2: Is the decimal system used for the CoCo ALU? Why?

A2:
Q3: What number system is used for computers?

A3:
Q4: How many numbers can be represented by the binary number system?

A4:
Q5: Name another pair of conditions that can be represented by the binary number system.

A5:
Q6: What is a BIT?

A6:
Q7: What holds the data inside

the CPU?

A7:
Q8: What are the TWO Accumulator Registers ?

A8:
Q9: In the 64K CoCo there are 65,536 Bytes. How many bytes are in 1K? How many BITS are in the 64K?

A9:
Q10: What Two registers make up the D register?

A10:
Q11: Name the index registers.

A11:
Q12: What size are the index registers?

A12:
Q13: What is the most common function of the index register?

A13:

Q14: What does CPU stand for?

A14:
Q15: What number system is used for clarity in discussing computer numbers?

A15:
Q16: What is a Register?

A16:
Q17: What does ALU stand for, and what does it do?

A17:
Q18: What is another name for the A and B registers?

A18:
Q19: What keeps track of the instructions in the 6809 CPU?

A19:
Q20: What command allows you to look in the registers whilst in zbug?

A20:
Q21: In zbug what commands are needed to change hexadecimal to decimal?

A21:
Q22: What are MNEMONICS?

A22:

Answers for Questionnaire 4

Q1: Where does the display screen start & end. Answer in both hex. & decimal?
A1: \$400-\$5FF and #1024-#1535

Q2: Is the decimal system used for the CoCo ALU? Why?

A2: No, because the activities the numbers represent would be clumsy or make little sense in decimal.

Q3: What number system is used for computers?

A3: The Binary system.

Q4: How many numbers can be represented by the binary number system?

A4 Two: the 0 and 1.

Q5: Name another pair of conditions that can be represented by the binary number system.

A5: On and Off

Q6: What is a BIT?

A6: One binary digit

Q7: What holds the data inside the CPU?

A7: Registers hold the data inside the Microprocessor.

Q8: What are the TWO Accumulator Registers?

A8: The A and the B.

Q9: In the 64K CoCo there are 65,536 Bytes. How many bytes are in 1K ? How many BITS are in the 64K?

A9: 1024 Bytes and 524,288 bits.

Q10: What Two registers make up the D register?

A10: Both the A & B register make the D register

Q11: Name the index registers?

A11: the X and the Y registers.

Q12: What size are the index registers?

A12: They are 16 bits or two bytes in size.

Q13: What is the most common function of the index register?

A13: To Index an address, that is to hold the number of a memory location for reference.

Q14: What does CPU stand for?
A14: Central Processing Unit.

Q15: What number system is used for clarity in discussing computer numbers?
A15: Hexadecimal.

Q16: What is a Register?
A16: A memory location in the CPU used to hold temporary results.

Q17: What does ALU stand for, and what does it do?

A17: The Arithmetic logic Unit performs simple calculations.

Q18: What is another name for the A and B registers?

A18: The accumulator registers.

Q19: What keeps track of the instructions in the 6809 CPU?
A19: The PC (program counter)

Q20: What command allows you to look in the registers whilst in zbug?

A20: The "R" command.

Q21: In zbug what commands are needed to change hexadecimal to decimal?

A21: I16 and O10

Q22: What are MNEMONICS?
A22: Mnemonics is the source listing (how the machine gets its instructions).

Amendum

"Screen Dump", by Craig Stewart (March CoCo, 1987) tells you everything there is to know about the program, ie the ins and outs, the problems Craig had in developing it and so on, but there was one small thing WE left out - how to use it!

Through trial and error we have found that it will work on all Tandy DMP-130 printers (if you have a printer that ISN'T a

DMP-130, and you find this program works on your printer, please ring us and let us know).

How to use this program:
1. RUN or (C)LOADM your graphics picture.

2. When it is completed, press BREAK to get out of the program, and ...

3. type 'EXEC' and you're in business (ie your printer should be printing away ...)

WHAT'S ON THE BEST OF CoCoOz

Best of CoCoOz #1. EDUCATION

ROADQUIZ ROB WEBB
 SHARE MARKET ALEPH DELTA
 HANGMAN ALEPH DELTA
 AUSTQUIZ P. THOMAS
 ALPHABET RON WEBB
 SPELLING TUTOR IAN LOBLEY
 TANK ADDITION DEAN HODGSON
 FRACTION TUTOR ROBBIE DALZELL
 TABLES BARRIE GERRAND
 ICOSA BOB WALTERS
 KIDSTUFF JOHANNA VAGG
 TAXMAN TONY PARFITT
 FLAGQUIZ ROB WEBB

Best of CoCoOz #2 part 1 16K GAMES

PYTHON V. ARMSTRONG
 COCONIND STEVE COLEMAN
 POKERMCH GRAHAM & MATTHEWS
 OILSLICK JEREMY GANS
 SPEEDMATHS DEAN HODGSON
 CCMETBOR BOB THOMSON
 BATTACK JEREMY GANS
 SKIING JOSHUA GANS
 PROBDICE BOB DELBOURGO
 RALLY TONY PARFITT
 CHECKERS J & J GANS
 FOURDRAW JOHANNA VAGG

Best of CoCoOz #2 part 2 32K GAMES

TREASURE DAVIDSON & GANS
 SHOOTING GALLERY TOM DYKEMA
 MASTERMIND GRAHAM JORDAN
 GARDEN OF EDEN DAVE BLUHORN
 ANESTHESIA MIKE MARTYN
 YAHTZEE KEVIN GOVAN
 OREGON TRAIL DEAN HODGSON
 BATTLESHIP CHRIS SIMPSON
 ADVENTURE + STUART RAYNER
 ANDROMEDIA MAX BETTRIDGE
 LANDATTACK ALDO DEBERNADIS

Best of CoCoOz #3 UTILITIES

SCREEN PRINT TOM DYKEMA
 RANTEST TOM DYKEMA
 PRINT SORT PAUL HUMPHRIES
 BEAUTY BOB THOMPSON
 DATAGEN ROBIN BROWN
 PCOPY BRIAN DOUGAN
 FASTEXT OZ-VIZ
 MONITOR + BRIAN FERGUSON
 COPYDIR THOMAS SZULCHA
 LABELLER FRED BISSELLING
 SPEED CONTROL PAUL HUMPHRIES
 2BC WARREN VARNE
 CREAT-A-TITLE BRIAN FERGUSON
 DISKFILE BRIAN DOUGAN
 BIG REMARKS BOB THOMSON
 LABELLER GORDON BEWITZEN
 DIR MORRIS SINGER
 HI ALEX. HARTMAN

Best of CoCoOz #4 Business

HI ALEX. HARTMAN
 (disk; Disk Directory Manager)
 PERSMAN PAUL HUMPHRIES
 (Personal Finance Management)
 BANKSTAT BARRY HATTAN
 (Annual & Store Statement)
 CC5 GRAHAM MORPHETT
 (tape; Sales invoicing)
 INSURE ROY VANDERSTEBB
 (Analyse Home Contents)
 COCOFILE BRIAN DOUGAN
 (tape; database)
 DPMS PAUL HUMPHRIES
 (disk; Disk Program Management Sys)
 DATABASE PAUL HUMPHRIES
 (tape; THE tape database)
 RESTACC DUNG LY
 (tape; Restaurant Accounts)
 SPDSHEET GRAHAM MORPHETT
 (disk; 22 column spreadsheet)
 PRSPDSHT GRAHAM MORPHETT
 (disk; prints out "SPDSHEET")
 ACS3 GREG WILSON
 (disk; Multi disk database)

Best of CoCoOz #5 ADVENTURES

ADV 32K S. RAYNER
 QUEST TONY PARFITT
 LABYRINTH JAMES REDMOND
 ADV + SEAN LOVE
 CRYSTAL C & K SPRINGETT
 PRISON TIM ALTON
 OPALTON IAN CLARKE
 WIZARD DARRELL BERRY
 TREASURE C. DAVIDSON
 LOST ALEX. HARTMAN

Best of CoCoOz #6 PRESCHOOL

ALPHABET STUART DAWSON
 HATDANCE JOHANNA VAGG
 AUSTSONG McDERMOTT FAMILY
 ADVANCE McDERMOTT FAMILY
 VALTZING McDERMOTT FAMILY
 TIMEKANG McDERMOTT FAMILY
 BAND McDERMOTT FAMILY
 KIDSTUFF JOHANNA VAGG
 MATCHER ?
 LETTERS JACK FINNEN
 BABYSIT JOHANNA VAGG
 SPELLING JOHANNA VAGG
 SPEEDTAB DEAN HODGSON
 10 FACES JOHANNA VAGG

Please Note: Some of the programs on Best of Cocooz # 3 and #4 will not work on the Coco 3.

Best of CoCoOz #7 GRAPHICS

LIL' COCO ANDREW WHITE
 THE ROOM HERMANN FREDRIKSON
 BACK STREET JOY WALLACE
 LOCO MIKE D'ESTERRE
 COCO ART SANDY MCGREGOR
 KANGA JOHANNA VAGG
 THE BOAT SANDY MCGREGOR
 SAD COCO F. BOLLE
 TOWER C. A. SYMS
 VINDY DAY SARAH LAV
 SAILING STEVE YOUNGBERRY
 outhouse STEVE YOUNGBERRY
 SMURF JOHANNA VAGG
 SUNSTATE STEVE YOUNGBERRY
 HELICOPTER ANDREW WHITE
 MARTHA ANDREW WHITE
 BAD MOON STEVE YOUNGBERRY
 MCC JOY WALLACE
 EAGLE ?
 BLASTER PAUL YOULD
 FOGHORN PAUL STEVENSON

Best of CoCoOz #8 16K GAMES

ALIEN STUART SANDERS
 QVERL DARRELL BERRY
 SHOOTOUT CRAIG STEWART
 SHUTTLE CRAIG STEWART
 FROG DARREN OTTERY
 FROGRACE TOM LEHANE
 KIMMAT TOM LEHANE
 GRANDPRI DOUG GREY
 WATER WARS JUSTIN LIPTON
 CATERPILLER JUSTIN LIPTON
 DETECTIVE VAL STEPHEN
 BREAKOUT WHY/BILT

Best of CoCoOz #9 32K GAMES

TRIXINO BOB DELBOURGO
 MATCHEM CHARLES BARTLETT
 GO BOB DELBOURGO
 MARZOD MAX BETTRIDGE
 CHOMPER MAX BETTRIDGE
 POPBALL MAX BETTRIDGE
 LUDO WHY/BILT
 SABRE ANDREW SIMPSON
 MOVEABOUT KEVIN GOVAN
 JIGSAV JAMES REDMOND
 LABYRINTH JAMES REDMOND
 TANK CRAIG STEWART

Best of CoCoOz #10 Education II

METBOR DEAN HODGSON
 DRIVERS TEST ANDREW SIMPSON
 SALE JUSTIN LIPTON
 TABLES PAT KERMODE
 OPALTON IAN CLARKE
 CAPITAL LETTERS BOB HORNE
 TEST MATCH JEFF SHIEN
 SENTENCE ENDINGS BOB HORNE
 ESCAPE DEAN HODGSON
 RAILMATH BOB HORNE
 COUNTDOWN DEAN HODGSON
 WHATZIT BOB HORNE
 HOMOPHONES BOB HORNE
 COMPOUND WORDS BOB HORNE

Best of CoCoOz #11 Education III This is a DISK only issue!!

CHATVIN MANOR BOB HORNE

TAPE \$16 each

DISK \$16 each

GOLDSOFT

P.O. BOX 1742, SOUTHPORT. QLD. 4215 Phone (075) 510 015

Goldsoft Price list as at March, 1987

Please tick your requirements.

Magazines, Tapes & Disks

Australian CoCo (Advanced Programs for your CoCo):

Magazines: Tape () or Disk ()
12 Months \$39.95 () 12 Months \$123.75 ()
6 Months \$24.95 () 6 Months \$ 74.25 ()
1 Month \$ 4.50 () 1 Month \$ 16.50 ()

Softgold (Programs for your CoCo):

Magazines: Tape () or Disk ()
12 Months \$39.95 () 12 Months \$123.75 ()
6 Months \$24.95 () 6 Months \$ 74.25 ()
1 Month \$ 4.50 () 1 Month \$ 16.50 ()

Gold Disk - Available Quarterly:

1 Month: \$16.00 ()

BACK ISSUES

Australian CoCo: Sep 84 - Dec 85: \$2.00 ea ()
Australian CoCo: Jan 86 - Feb 87: \$3.75 ea ()
Australian Mico: Aug 84 - Dec 85: \$2.00 ea ()

ADDITIONAL REQUIREMENTS

.....
.....
.....
.....

New Subscription: () Renewal: ()

Sub No:

Name:

Address:

.....

..... P/Codes.....

Phone: (....)

Please find enclosed:

- a. Cheque ()
- b. Money Order ()
- c. Credit Card ()

Credit Card Type & Number:

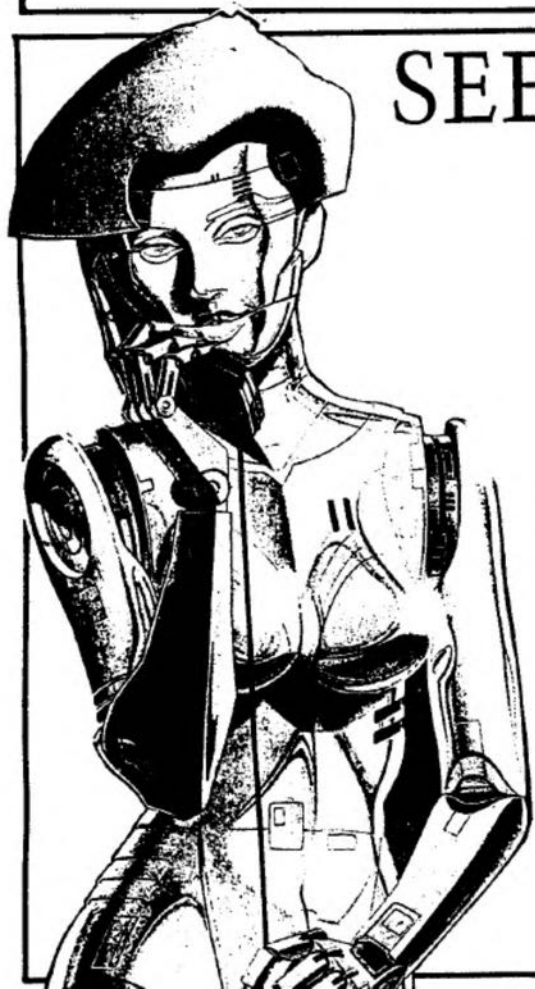
Bankcard () Mastercard () Visa ()

.....

Expiry Date:/.....

Authorised Amount: \$.....

Signed:



SEEN GOLDLINK YET?

You can only access GOLDLINK on VIATEL if you have a videotex terminal program and a modem capable of operating at 1200/75 baud.

Now Goldsoft presents two great specials!

- 1 CoCo Owners: Modem, CoCotex Software and cable\$289.00
- 2 TANDY 1000 or IBM PC Owners: Modem, VTex Software and cable\$399.00

GOLDSOFT
PO BOX 1742
SOUTHPORT QLD 4215
(075) 51-0577

ACT:
 CANBERRA NTH JOHN BURGER 062 58 3924
 CANBERRA STH LES THURBON 062 88 9226

NSV:
 SYDNEY:
 BANKSTOWN PAT DORSETT 02 646 3619
 BLACKTOWN KEITH GALLAGHER 02-627-4627
 CARLINGFORD ROSKO MCKAY 02 624 3353
 CHATSWOOD BILL O'DONNELL 02 419 6081
 COLYTON HERMAN FREDRIKSSON 02 623 6379
 FAIRFIELD ARTH PITTARD 02 72 2881
 GLADESVILLE MARK ROTHVELL 02 817 4627
 HILLS DIST ARTHUR SLADE 02 622 8940
 HORNSBY ATHALIE SMART 02 848 8830
 INGLESBURN STEPHEN RIDGEWAY 02 605 7382
 KENTHURST TOK STUART 02 654 2178
 LEICHHARDT STEVEN CHICOS 02 560 6207
 or GORGE ECHEGARAY 02 560 9664
 LIVERPOOL LEONIE DUGGAN 02-607-3791
 MACQUARIE FIELDS

BARRY DARTON 02 618 1909
 MIMIO GRAHAM POLLOCK 02 603 5028
 SUTHERLAND JAN ANABEL 02 528 3391
 SYDNEY EAST JACKY COCKINGS 02 344 9111
 ALBURY RON DUNCAN 060 43 1031
 ARNDIDALE DOUG BARBER 067 72 7647
 BLAXLAND BRUCE SULLIVAN 047 39 3903
 BROKEN HILL TERRY NOONAN 080 88 2382
 CANDEW KEVIN WINTERS 046.66.8068
 COFFS HARBOUR BOB KENNY 066 51 2205
 COOMA ROSS PRATT 064 52 3065
 COORANBONG GEORGE SAVAGE 049 77 1054
 COOTANUNDRRA CHERYL WILLIS 069 42 2264
 DEWILQUIN WAYNE PATTERSON 058 81 3014
 DUBBO GRAEME CLARKE 068 89 6549
 FORBES JOHANNA VAGG 068 52 2943
 GOSFORD PETER SEIFERT 043 32 7874
 GRAFTON PETER LINDSAY 066 42 2503
 GUYRA MICHAEL J. HARTMANN 067 79 7547
 JUNEE PAUL MALONEY 069 24 1860
 KEMPSEY RICK FULLER 065-62-7222
 LEETON BRETT WALLACE 069-53-2081
 LISMORE ROB HILLARD 066 24 3089
 LITHGOW DAVID BERGER 063 52 2282
 MAITLAND BILL SNOW 049 66 2557
 MUREE ALF BATE 067 52 2465
 MUDGEE BRIAN STONE 063-72-1958
 NARBUNGA HDS WENDY PETERSON 065 68 6723
 NARRONINE GRAEME CLARKE 068 89 6549
 NEWCASTLE LYN DAVSON 049 49 8144
 NUNRA ROY LOPEZ 044 48 7031
 PARKES DAVID SMALL 068 62 2682
 PORT MACQUARIE RON LALOR 065 83 8223
 SPRINGWOOD DAVID SEANONS 047 51 2107

(Stop between numbers - b.h. else
 a.h.; but, hyphen between = both)

USER CONTACTS

TAHMOOR GARY SYLVESTER 046 81 9318
 UPPER HUNTER TERRY GRAVOLIN 065 45 1698
 URALLA FRANK MUDFORD 067 78 4391
 WAGGA WAGGA CES JENKINSON 069 25 2263
 WYONG JOHN WALLACE 043 90 0312

NT:
 DARWIN BRETON PRIOR 089.81.7766

QLD:
 BRISBANE:
 BIRKDALE COLIN NORTH 07 824 2128
 CLAYFIELD JACK FRICKER 07 262 8869
 COLL'WOOD PK ANDREW SIMPSON 07 288 5206
 IPSWICH NICK MURPHY 07 271 1777
 PINE RIVERS BOB CLARKE 07 204 2806
 SOUTH WEST BOB DEYRIES 07 372 7816
 SANDGATE MARK NIGHILL 07 269 3846
 SCARBOROUGH PETER MAY 07 203 6723
 WOODRIDGE ALLAN ALLSOP 07 349 1831
 AIRLIE BEACH GLEN EVANS 079 46 1264
 BIGGENDEN ALAN MENHAM 071 27 1272
 BOWEN TERRY COTTON C/O 077 86 2220
 BUNDABERG RON SIMPKIN 071 71 5301
 CAIRNS JEFF LARSEN 070 54 7127
 DALBY MERRICK TANSKY 074.62.3228
 GLADSTONE CAROL CATHCART 079 78 3594
 GOLD COAST GRAHAM MORPHEIT 075 51 0577
 GYMPIE BERT LLOYD 071 8219100
 HERVEY BAY LESLEY HORWOOD 071 22 4989
 MACKAY LEN MALONEY 079511333x782
 MARYBOROUGH JOHN EFFER 071 21 6638
 MT ISA JACK RAB 077 43 3486
 MURGOON PETER ANGEL 071 68 1628
 ROCKHAMPTON KEIRAN SIMPSON 079 28 6162
 TARA DEBBIE DORFIELD 074 65 3177
 TOOWOOMBA LEN GERSEKOWSKI 076 35 8264
 TOWNSVILLE JOHN O'CALLAGHAN 077 73 2064

SA:
 ADELAIDE JOHN HAINES 08 278 3560
 NORTH STEVEN EISENBERG 08 250 6214
 MORPHETTVALE KEN RICHARDS 08 384 4503
 PORT MOURLUNGA ROB DALZELL 08 386 1647
 SEACOMBE HTS OLENN DAVIS 08 296 7477
 PORT LINCOLN BILL BOARDMAN 086 82 2385
 PORT PIRIE VIC KNAUERHASH 086 32 1230
 WHYALLA MALCOLM PATRICK 086 45 7637

TAS:
 DEVONPORT JEFF BEST 004 24 6759
 HOBART BOB DELBOURGO 002 25 3896
 KINGSTON VIN DE PUIT 002 29 4950
 LAUNCESTON BILL BOVER 003 44 1584
 SMITHTON HARRY CHRISAFIS 004-52-1590
 WYNYARD ANDREW VYLLIE 004 35 1839

VIC:
 MELBOURNE:
 MELBOURNE CCC JOY WALLACE 03 277 5182
 DANDENONG DAVID HORROCKS 03 707 5870
 DONCASTER JUSTIN LIPTON 03 857 5149
 FRANKSTON BOB HAYTER 03.783.9748
 MARRE WARREN LEIGH BAMES 03 704 6680
 NTH EASTERN PETER WOOD 03 435 2018
 MELTON MARIO GERADA 03 743 1323
 RINGWOOD IVOR DAVIES 03 758 4496
 SUNBURY JACK SMIT 03.744.1355
 SUNSHINE IAN BUTTRISS 03 314 3240
 UPR F'TREE GLY RORY DOYLE 03 758 2671
 BAINSDALE COLIN LEHMANN 051 87 1545
 BALLARAT MARK BEVELANDER 053 32 6733
 CHURCHILL GEOFF SPOWART 051 22 1389
 DAYLESFORD DANNY HEDJI 054 24 8329
 GEELONG DAVID COLLEN 052 43 2128
 MAFFRA MAX HUCKERBY 051 45 4315
 MOE JOSEPH HESTER 051 27 7817
 MORRINGTON MICHAEL MONCK 03 789 7997
 MORVELL GEORGE FRANCIS 051 34 5175
 SALE BRYAN MCHUGH 051 44 4792
 SHEPPARTON ROSS FARRAR 058 25 1007
 SMYTHSDALE TONY PATTERSON 053 42 8815
 SWAN HILL BARRIE GERRARD 050.32.2838
 TONGALA TONY HILLIS 058 59 2251
 TRARALGON LEIGH DAVES 051 74 5552
 WONTHAGGI LOIS O'NEARA 050 72 1593
 YARRAWONGA KEN SPONG 057 44 1488

WA:
 PERTH IAIN MACLEOD 09 448 2136
 GIRRAWAHN HANK WILLENSEN 09 342 7639
 KALGOORLIE TERRY BURBETT 090.21.5212

CANADA - CoCo:
 Ontario Richard Hobson 416 293 2346
 Toronto Franz Lichtenberg 416 845 2889

Special interest groups

TEACHERS' INTEREST GROUP
 BRISBANE BOB HORNE 07 281 8151

BUSINESS:
 BRIZBIZ BRIAN BERE-STREETER 07 349 4696

OS9 GROUPS:
 NATIONAL OS9 USERS' GROUP
 GRAEME NICHOLS 02 451 2954

NSV
 SYDNEY
 BANKSTOWN CARL STERN 02 646 3619
 CARLINGFORD ROSKO MCKAY 02 624 3353
 GLADESVILLE MARK ROTHVELL 02 817 4627
 SYDNEY EAST JACKY COCKINGS 02.344.9111
 COOMA FRED BISSELLING 064 52 3263
 QLD
 BRISBANE JACK FRICKER 07 262 8869

VIC
 LATROBE VLY GEORGE FRANCIS 051 34 5175
 VA
 KALGOORLIE TERRY BURBETT 090.21.5212

MC-10 CONTACTS:
 LISMORE BOB HILLARD 066 24 3089
 SYDNEY GRAHAM POLLOCK 02 603 5028
 WARRNAMBOOL GARY FURR 055 62 7440

TANDY 1000 / MS DOS:
 NSV:
 GLADESVILLE MARK ROTHVELL 02 817 4627
 SYDNEY WEST ROGER RUTHERN 047.39.3903
 WYONG JOHN WALLACE 043 90 0312

QLD:
 BRISBANE
 NORTH BRIAN DOUGAN 07 30 2072
 SOUTH BARRY CAVLEY 07 390 7946
 GOLD COAST GRAHAM MORPHEIT 075 51 0577

SA
 PORT LINCOLN BILL BOARDMAN 086 82 2385
 VIC:
 MELBOURNE TONY LLOYD 03 882 4664

FORTH:
 SYDNEY JOHN REDMOND 02 85 3751

ROBOTICS:
 GOLD COAST GRAHAM MORPHEIT 075 51 0577
 SYDNEY GEOFF FIALA 02 84 3172
 WAGGA WAGGA CES JENKINSON 069 25 2263

CHRISTIAN USERS' GROUP:
 COLLIE RAYMOND L. ISAAC 097 34 1578

300 BAUD BULLETIN BOARDS

SYDNEY:
 INFOCENTRE 02 344 9511
 TANDY ACCESS 02 625 8071
 THE COCOCONNECTION 02 618 3591
 HUB 002 49 4405

VIDBOTEX SYSTEMS

VIATEL 01955
 MOUSETEX 059 42 5528
 VIX 4000 03 741 3295

TANDY INFO ON VIATEL

GOLDLINK VIATEL #642#
 BLAXLAND COMPUTER SERVICES VIATEL #64263#
 PARIS RADIO VIATEL #64268#
 POWER CODE VIATEL #64265#
 TANDY VIATEL #64261#

SOME TANDY USERS ON VIATEL

ALLAN BEALE 726353300
 FRED BISSELLING 648232630
 JACK FRICKER 726288690
 JOHN GRIGSBY 945872030
 STUART HALL 939765790
 BOB KENNY 665122050
 JEFF LARSEN 705471270
 IAIN MACLEOD 944821360
 CHRIS WAGLE 689523360
 RICHARD PANKHURST 280717870
 ROSS PRATT 648230650
 RICCATY SCHMAHL 298151500
 ARTHUR SLADE 262289400
 DARCY O'TOOLE 755105770
 RON WRIGHT 352924510

GOLOWINK

COM. STATION

642

ON

Telecom
VIATEL
AUSTRALIA'S NATIONAL VIDEOTEX SERVICE



BULLETIN BOARDS
TANDY COMPUTERS
ATARI COMPUTERS
COMMODORE COMPUTERS
FAST FLORIST
REVIEWS
HARDWARE & SOFTWARE
SHOP

SOFTGOLD MAGAZINE
Registered by Australia Post —
Publication No. QBG 4009
AUSTRALIAN CoCo
Publication No. QBG 4007
P.O. BOX 1742
SOUTHPORT, QLD. Australia. 4215

**POSTAGE
PAID
AUSTRALIA**

YES WE EVEN TALK BACK !!