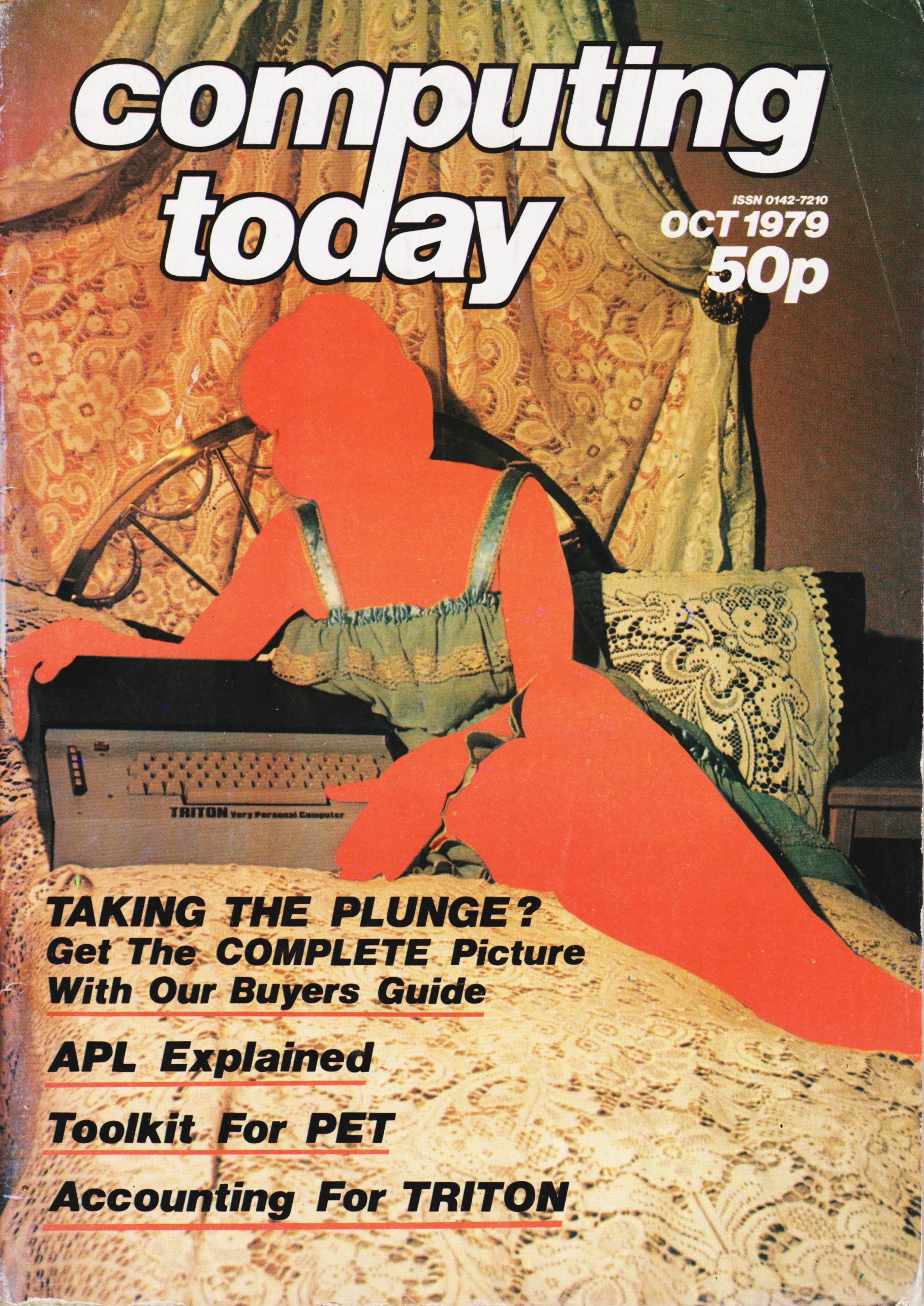
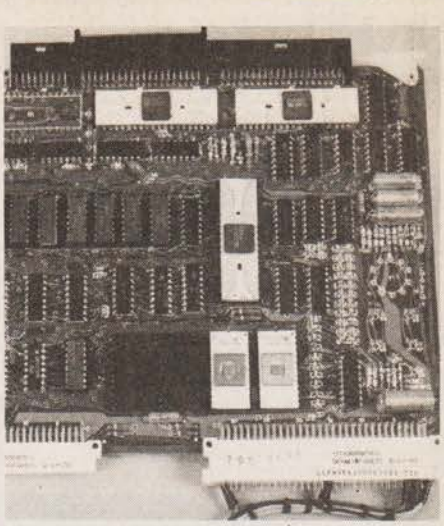

computing today
ISSN 0142-7210
OCT 1979
50p
TRITON Very Personal Computer
TAKING THE PLUNGE?
Get The COMPLETE Picture
With Our Buyers Guide
APL Explained
Toolkit For PET
Accounting For TRITON

computing today

VOL.1,NO. 8
OCT 1979



Editor: Ron Harris B.Sc
Ed. Assistant: Henry Budgett
Art Director: Diego Rincon
Production: Dee Camilleri, Loraine Radmore, Paul Edwards, Tony Strakes, Joanne Barseghian.
Ad. Manager: Chris Surgenor
Ad. Representative: David Sinfield
Editorial Director: Halvor Moorshead

Computing Today International is normally published on the third Friday of the month prior to the cover date.

Distributed by Argus Distribution Ltd. Printed by LSG. Limited, Lincoln.

EDITORIAL AND ADVERTISEMENT OFFICE
145 Charing Cross Road, London WC2H 0EE. Telephone 01-437 1002/3/4/5

L.P. ENTERPRISES

Room CT10
313 Kingston Road, Ilford,
Essex, IG1 1PJ, England
01-553 1001

EUROPE'S LARGEST SELECTION OF MICROCOMPUTER BOOKS, MAGAZINES AND SOFTWARE FOR THE HOBBYIST, EDUCATIONALIST, PROFESSIONAL AND RETAILER.

BOOKS

SUMMER HOLIDAY BONUS: for the purchase of 3 books or more, and paying by cheque, P.O. or cash, give yourself a 10% DISCOUNT!

Introduction to Microcomputers: by Osborne	
Vol 0: Beginners Book	£5.95
Vol 1: Basic Concepts	£6.30
Vol 2: Some Real Microprocessors (without binder)	£18.95
Vol 2: Some Real Microprocessors (with binder)	£24.70
Vol 3: Some Real Support Devices (without binder)	£11.95
Vol 3: Some Real Support Devices (with binder)	£17.70
Updating subscription (6 issues) for Vol 2	£18.95
Updating subscription (6 issues(for Vol 3	£18.95
Updating subscriptions for Vol 2 & 3	£30.00
1 Updating issue (specify for Vol 2 or 3)	£4.00
1 Binder (Specify for Vol 2 or 3)	£5.75

6800 Programming for Logic Design	£6.30
8080 Programming for Logic Design	£6.30
Z80 Programming for Logic Design	£6.30

More BASIC Computer Games	£5.50
BASIC Computer Games (also see software section)	£5.00
What To Do After You Hit Return	£8.95
8080 Galaxy Game	£6.95
SUPER-WUMPUS — A game in 6800 Assembler code & BASIC	£4.25
Computer Music	£6.75
Computer Rage (A Board Game)	£6.95
Artist and Computer	£3.95
Games with a Pocket Calculator	£1.75
Games, Tricks & Puzzles for a Hand Calculator	£2.49
Introduction to TRS-80 graphics	£5.75
Take My Computer Please... (light hearted fiction)	£3.25

Z80 Instruction Handbook	£2.95
8080 Programmers Pocket Guide	£1.95
8080 Hex Code Card	£1.95
8080 Octal Code Card	£1.95

Best of BYTE	£8.95
Scelbi BYTE Primer	£8.95
Best of Creative Computing Vol 1	£6.95
Best of Creative Computing Vol 2	£6.95
Best of MICRO (Issues 1-6 of Micro Magazine)	£5.50

Z80 Assembly Language Programming	£6.45
6502 Assembly Language Programming (coming soon)	£6.45
Microcomputer Programming 6502	£7.95
6502 Applications Book	£7.95
8080A/8085 Assembly Language Programming	£6.45
6800 Assembly Language Programming	£6.45
8080 Software Gourmet Guide and Cookbook	£6.95
6800 Software Gourmet Guide and Cookbook	£6.95
8080/8085 Software Design	£6.75
6800 Tracer — An aid to 6800 Programme Debugging	£3.95
Program Design	£4.25
Programming Techniques: Simulation	£4.25

PIMS — A Database Management System	£5.95
Scelbal High Level Language + Supplements	£15.00
Basex — A Simple Language + Compiler for the 8080	£5.50

Microprocessors from Chips to Systems	£7.00
Microprocessor Interfacing Techniques	£8.75
Z80 Microcomputer Handbook	£7.50
TV Typewriter Cookbook	£7.50
Cheap Video Cookbook	£4.30
CMOS Cookbook	£7.50
IC OP-AMP Cookbook	£8.95
RTL Cookbook	£4.25
TTL Cookbook	£7.50
IC Timer Cookbook	£7.50
Ciarcias Circuit Cellar	£5.50
First Book of KIM	£7.00

Introduction to Personal and Business Computing	£4.95
Getting Involved with your Own Computer	£4.75
Buyer's Guide to Microsoftware	£2.40
How to Profit from Your Personal Computer	£5.50
Microcomputer Potpourri	£1.75
Hobby Computers are Here	£3.95
New Hobby Computers	£3.95
Understanding Microcomputers and Small Computer Systems	£6.95

Instant BASIC	£6.95
Basic BASIC	£6.50
Advanced BASIC	£6.00
My Computer Likes Me . . . When I Speak in BASIC	£2.75
Calculating with BASIC	£4.95
Users Guide to North Star BASIC	£10.00
Introduction to PASCAL	£3.95

Accounts Payable and Accounts Receivable	£10.95
Payroll with Cost Accounting	£10.95
General Ledger	£10.95

Basic Software Library:	
Vol 1: Business and Games Programs	£17.50
Vol 2: Maths, Engineering and Statistical Programs	£17.50
Vol 3: Advanced Business Programs	£26.95
Vol 4: General Purpose Programs	£7.95
Vol 5: Experimenters Programs	£7.95
Vol 6: Miniature Business System	£32.50
Vol 7: Chess/Medbil/Wdproc Programs	£26.95

Some Common BASIC Programs	£6.30
Computer Programs that Work (in BASIC)	£2.55
32 BASIC Programs for the PET	£10.10

8080 Standard Monitor	£9.95
8080 Standard Editor	£9.95
8080 Standard Assembler	£9.95
Special Package: 8080 Assembler, Editor, Monitor	£20.00
Bar Code Loader for 6800, 8080, Z80 and 6502	£1.75
Tiny Assembler for 6800 Systems	£5.75
RA 6800 ML — An M600 Relocatable Macro Assembler	£15.95
LINK 68 — An M6800 Linking Loader	£5.50
MONDEB — An advanced M6800 Monitor Debugger	£3.50

MAGAZINES

SUMMER HOLIDAY BONUS: For the purchase of 3 Magazines or more, and paying by cheque, P.O. or cash, give yourself a 10% DISCOUNT!

Magazine Subscriptions:

Subscriptions start within 3 weeks

	UK Price	Overseas Price
MICRO-6502 Journal (12 issues)	£12.50	£12.50
Personal Computing (12 issues)	£17.00	£17.00
Interface Age (12 issues)	£25.00	£25.00
Dr Dobbs Journal (10 issues)	£13.50	£13.50
Computer Music Journal (4 issues)	£11.00	£11.00
People's Computers (6 issues)	£8.50	£8.50
BYTE (12 issues)	£24.50	£24.50
Creative Computing (12 issues)	£16.50	£16.50
Kilobaud (12 issues)	£21.00	£21.00

Magazine Back Issues:

Micro-6502 Journal	£1.50
Personal Computing	£1.95
Interface Age	£2.95
ROM	£1.95
Dr Dobbs Journal	£1.95
Computer Music Journal	£3.75
People's Computers	£1.95
BYTE	£2.95
Creative Computing	£1.95
Calculators and Computers	£1.95
Kilobaud (reprints only)	P.O.A.
73	£2.25
Magazine Storage Box (Holds 12)	£1.25

HOW TO ORDER

Please note our book magazine prices include postage and packing, but not insurance, if wanted add 12p for every £10. of **books** ordered. Make cheques, PO's etc. payable to:- **L.P.Enterprises.**

CREDIT CARDS accepted
BARCLAYCARD VISA/ACCESS/DINERS CLUB/
AMERICAN EXPRESS
Phone: 01-553 1001 for Credit Card orders (24 hr answering service)

GREEN IS BEAUTIFUL

Available now from Petsoft is an instant green screen device. Using a special optical perspex it will attach to the front of any standard PET and relieve eyestrain as well as cutting down on ambient light reflection. For a mere £8.50 plus VAT you can get one from your local dealer or direct from Petsoft at 5 Vicarage Road, Edgbaston, Birmingham B15 3ES.

COLOUR NASCOM

Adding colour graphics to a Nascom is now easy with a Micrographics kit from William Stuart. Using a true dot generator rather than preformed characters the system gives control over the whole screen by using supplied subroutines. All program code is relocateable and compatible with any Nascom monitor. As well as allowing eight colours to be used in the graphics a choice of eight background colours are available and text can be mixed with diagrams. Text colours can be changed in two instructions. The kit is supplied complete with a PAL modulator and documentation. A demo program is supplied as well to show what can be done. Cost is £45 for the kit or £59 assembled and it can be obtained from William Stuart Systems Ltd., Dower House, Herongate, Brentwood, Essex. CM13 3SD.

NEWBEAR EXPAND

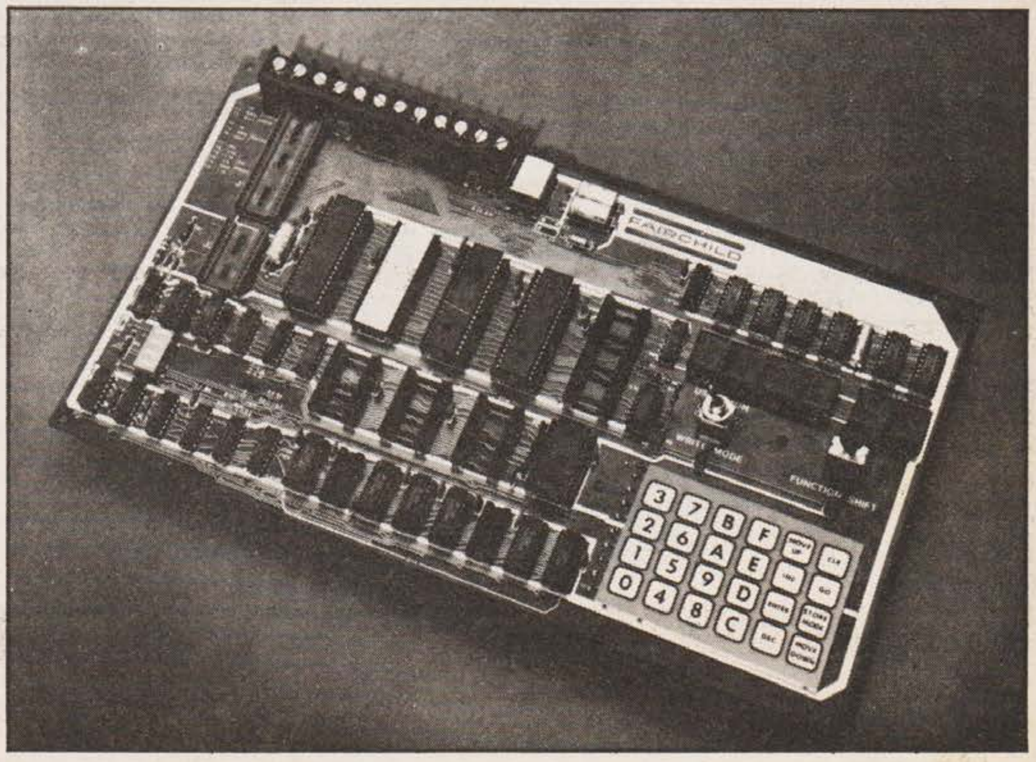
The Newbear computing store has recently added dealerships for the Apple and Ithaca International to its collection. The decision to opt for Apple rather than the ITT2020 was taken after discussions held at the NCC and they can now supply the 16K Eurapple at £830 plus VAT. From Ithaca comes the DPS-1 system, an S100 based system with front panel, backplane and power supply. Up to 20 user boards can be held and a full range is available. More details on both new dealerships can be had from Newbear on 0635-30505.

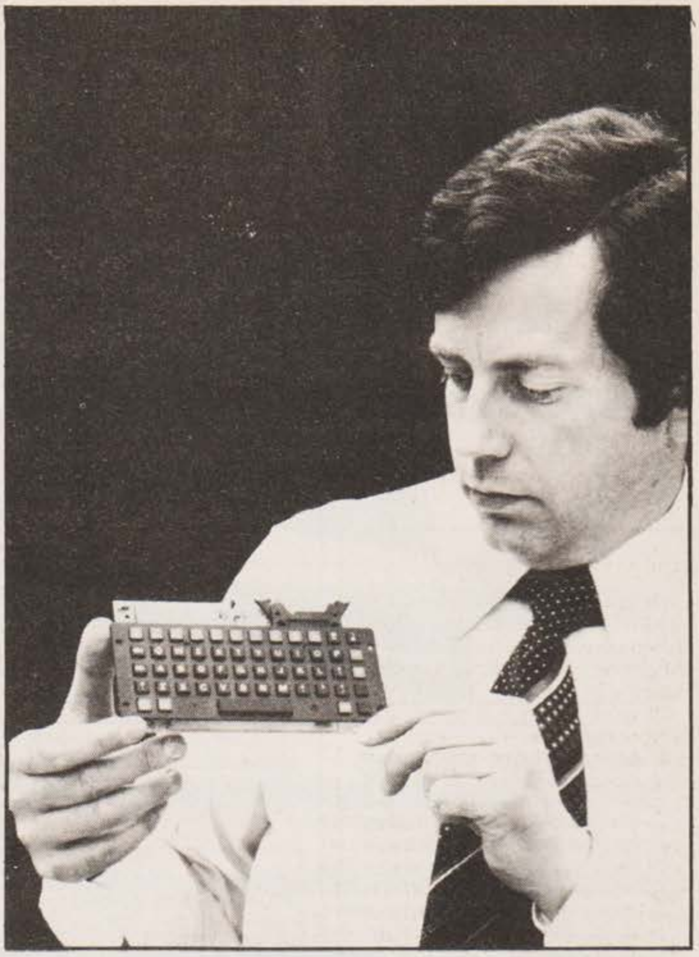
BOOK SELL OUT

The National Computing Centre's book called Introducing Microprocessors has gone into reprint already — it was only launched in June. This is one of the 18 new titles being published and the free catalogue is available from the NCC at Oxford Road, Manchester, M1 7ED. Over 90 books are available covering a wide range of subjects from standards to applications.

PEP SYSTEM FROM FAIRCHILD

A single board evaluator has been recently introduced by Fairchild for the F387X family of 8 bit micros. The PEP system allows a user to write and debug programs and hardware for the chip series and also allows in circuit emulation via an umbilical. Programs can be called down from a cross assembler on a development system such as the Intel MDS or the Formulator. On board facilities include a keypad and a six digit LED display. Also available is an RS232 Port at 110 to 1200 Baud for connection to a full terminal. 2K of 2114 static RAM is available, expandable to 4K as well as a separate 128 byte workspace. You can also blow 2716 EPROMS from the board and there is space for up to 6K of these devices on board. The monitor is a 2K firmware type and other features include crystal controlled clocks, four programmable timers and four general purpose interrupt controls. Contact Fairchild at 230 High Street, Potters Bar, Herts EN6 5BU.





LITTLE FINGERS?

A new range of miniature keyboards is being imported by Walmore Electronics. The Apex range of microkeyboards are full 8 bit ASCII and about 1/3rd the size of a conventional keyboard. Designed mainly for portable and instrumentation equipment and are fully TTL/MOS compatible. All switches have a tactile "feel" and there are selectable parity and strobe/data inversion facilities. For those of you with little fingers please contact Walmore Electronics Ltd., Displays Division, 11-15 Betterton Street, Drury Lane, London WC2H 9BS.

CLUB FORUM

Some more connections have reached me on the Club Survey this month. Apparently the Eltham ACC and the South East London Micro-computer Club are one and the same and the latter name is now the correct one, or SELMIC for short. They now have around 150 members and a new committee. Primary contacts are Treasurer, Mr Hugh Gillespie on 01-303 4968; Membership secretary, Mr Malcolm Beresford on 01-698 1422 and the Chairman, Mr John Williamson on 01-850 4195.

The latest schedule has reached me from the East London Amateur Computer Club. They have demos scheduled for Apple on the 16th of October and the AGM on the 20th of November, among others. They meet on the third Tuesday of each month at Harrow Green Library in Cathall Road, Leytonstone from 7pm to 10pm. Bus routes 10 and 262 and Leyton tube are most convenient transport. An informal meet is held on the first Tuesday of each month in The Bell pub, Leytonstone High Road. A book loan scheme is being set up within the club, a newsletter is published and membership is currently £2.50 (half price for students). For more details contact the Chairman, Jim Turner at 63 Millais Road, London E11.

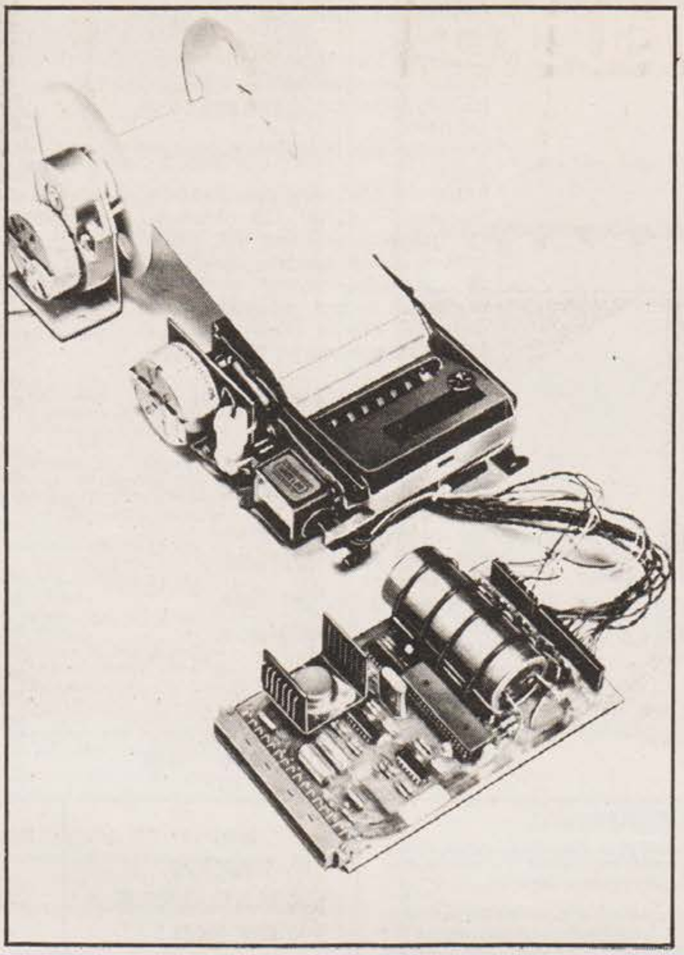
Next on the list is the Brunel Technical College Computing Club. They now have an elected committee of 7 and 42 mem-

PRINTA PRINTER

A range of OEM printers is available from Landis and Gyr of North London. All of the Sodeco range in either 15 or 20 column widths can be supplied in text or data formats and a two control cards are available for connection to either an ASCII data source of a BCD source. Both interface cards are on a Eurocard format and require +24V unstabilized. A buffer store can be included to hold one line whilst one is printing. Also available is a winder unit to take up the roll print. Further details from Landis and Gyr, Industrial Components Division, FREEPOST, London W3 6BR or ring 01-992 5311.

STARBASE BASIC

Starbase, have produced an 8K BASIC for the Nascom. The language is a modification of the Microsoft standard package and locates at 1000 to 2FFFF and will work in any system with an additional 16K. All variables are copied into a workspace area so the language may be PROM blown with no mods. All current monitors are supported and line editing will be available with Nas-Sys. Extra features of the language are a WIDTH command and a facility to use a standard terminal via the serial interface. The LIST command will scroll a specified number of lines and the Nascom graphics option is also supported. Program Load and Dump uses the T4 routine with checking and file names. The language costs £30 from your local dealer, Nascom or Starbase at Waxhouse Gate, 15 High Street, St. Albans, Hertfordshire



WINNERS LOSE OUT

It has been brought to our attention that a competition currently being run by Comp Computer Components, see this issue, although totally legal leaves them the winners. How so you may ask? Well the Rules state that all the submitted software becomes the property of Compukit and they may publish it as they see fit. Considering the high prices paid for software by companies and magazines it seems to us that the entrants are paying for their own prizes and leaving Compukit with a veritable pile of valuable 'rejects'.

HIGH RES FOR APPLE

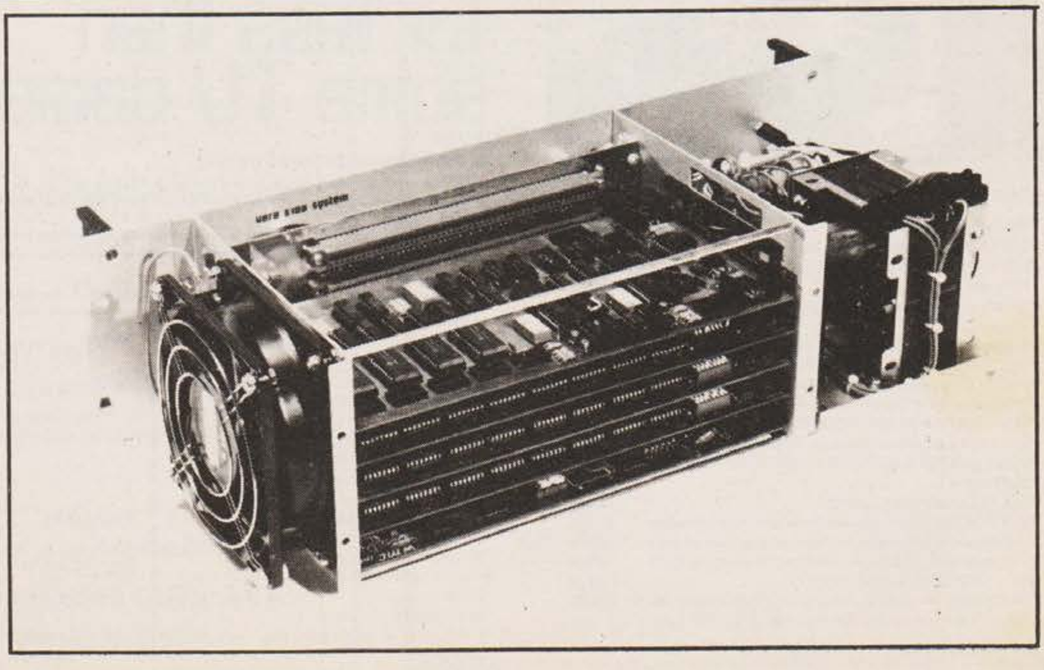
Wanna play with high resolution graphics? Can't afford the terminal! Well you can now simulate the Tektronix 4010 family with your Apple. For about £300 Personal Computers will let you have a TEKSIM ROM chip and adaptor which allows you to display graphics from a host computer, send graphics to the host computer and produce colour graphics. The TEKSIM requires the Apple communication card for connection to the host and then uses software to simulate the Tektronix style graphics on 32 lines of 68 characters. Local commands allow you to save displays on tape or disk. For use as an intelligent graphics terminal the Apple offers a low cost alternative to the Tektronix range with the addition of TEKSIM. For more details contact Personal Computers on 01-623 7970.

HOLD YOUR S100

Vero have just released an S100 racking system complete with integral PSU. Capable of handling up to six cards at 4MHz it supplies +8, –8 and +18V and has a fan for cool running. Mains input is filtered and there is a front panel reset control. The unit can either be rack mounted or cased and the sample we saw was of very high quality indeed. A full range of S100 items is available such as prototype boards and card extenders. For further information contact Vero at the Industrial Estate, Chandlers Ford, Eastleigh, Hants.

bers, the oldest was born in 1902! They meet every third Wednesday and membership costs £2.00. They will be re-starting meetings at the beginning of term and there will be two groups:— skilled and not skilled. Primary contact is Mr S.W. Rabone, 18, Castle Road, Worle, Weston Super Mare, Avon BS22 9JW.

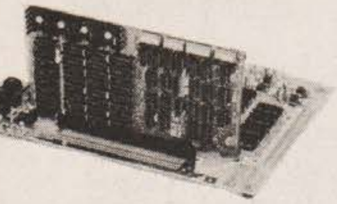
And last but not least this month we have a new club to join our ranks called the Swansea and South Wales Amateur Computer Club the acting committee would like to hear from anyone in the area. Primary contact is Peter Skan, the acting chairman, and you can contact him at 6D7, Vivian House, Roman Bridge Close, Blackpill, Swansea SA3 5BG.



NEW from NEWTRONICS 'EXPLORER 85'

microcomputer kit
Low Cost with On Board
S-100 expansion

at **£295** + VAT



The 'EXPLORER 85' is inexpensive with all the advantages of a powerful board plus potential for 'infinite' expansion.

* Uses New Fast INTEL 8085 cpu, 100% compatible with 8080A software but 50% faster than 8080A cpu.
* Powerful 2K monitor
* 4K user RAM expandable to 64K
* Provision for 8K PROM or EPROM
* Buffered and decoded S100 expansion on board (up to 6 S100 boards)
* Cassette interface (with motor control and cassette file structure) RS232: 20ma loop: 4 8 bit and 1 6 bit I/O ports
* Programmable 14 bit binary counter/timer
* Separate ASC11/Video Terminal features:— a full 128 character set upper/lower case, full cursor control, Greek symbols for Maths, 75 ohm video output convertible to baudot output, selectable baud rate, RS 232 or 20ma loop, I/O 64 or 32 characters by 16 line (monitor or TV)
* And lots of other great features

	Price Ex Vat	P & P
LEVEL 'C' S-100 main frame expander kit. Increases the number of S-100 slots to 6. Includes all sheet metal, 5 slot extender board, board-guides and brackets. Fits into EXPLORER cabinet (less S-100 pin connectors).	32.80	2.00
LEVEL 'E' Add 8K sockets, power supply regulator and decoupling components for popular 2716 or 2516 EPROMS (EPROMS not included)	5.00	Free
DE LUXE STEEL CABINET FOR EXPLORER 85	33.50	2.00
DE LUXE STEEL CABINET FOR VIDEO KEYBOARD TERMINAL	15.00	2.00
POWER SUPPLY IN STEEL CABINET	25.00	2.00
GOLD PLATED S-100 BUS CONNECTORS	4.00	Free
RF MODULATOR 8kHz required when using TV set aerial socket	6.00	Free
S-100 16K RAM EXPANDABLE Expandable to 60K on one board	165.00	2.00
INTEL 80805 Users Manual	6.00	Free
8K 'MICROSOFT' BASIC ON CASSETTE TAPE The most versatile and popular basic ever written. Complete with documentation.	53.00	Free

NEWTRONICS KEYBOARD TERMINAL

The Newtronics Keyboard Terminal is a low cost stand alone Video Terminal that operates quietly and maintenance free. It will allow you to display on a monitor 16 lines of 64 characters or 16 lines of 32 characters on a modified TV (RF Modulator required).

The characters can be any of the 96 ASC 11 alphanumerics and any of the 32 special characters, in addition to upper lower case capability it has scroll-up features and full X-Y cursor control. All that is required from your microcomputer is 300 baud RS232-C or 20ma loop serial data plus a power source of 8v DC and 6 3v AC. The steel cabinet is finished in IBM Blue-Black. And if that is not enough the price is only **£135.85 + VAT as a Kit, or £175 + VAT assembled and tested.** Plus £2 P&P (Monitor not included).

NOW AVAILABLE 8K FULL BASIC

NEW NOW 'NEWSOFT' GAMES for ELF II 4 GAMES for £5.00

send for list

BUY AN ELF II microcomputer for less than some TV games

Only £79.95 +VAT

ELF 11 BOARD WITH VIDEO OUTPUT

STOP reading about computers and get your "hands on" an, ELF 11 and Tom Pitman's short course. ELF 11 demonstrates all the 91 commands which an RCA 1802 can execute, and the short course speedily instructs you how to use them.

ELF 11's VIDEO OUTPUT makes it unique among computers selling at such a modest price. ELF 11 is perfect for engineers, business, industry, scientific and educational purposes.

*Power Supply (6.3v AC) for ELF 11	£5.00
*ELF 11 Deluxe Steel Cabinet (IBM Blue)	£23.01
*Giant Board Kit System/Moniter, Interface to/cassette, RS232, TTY, etc.	£35.00
*4K Static RAM board kits (requires expansion power supply)	£69.44
*Expansion power supply (required when adding 4K Rams)	£19.00
*ASC11 Keyboard Kits 96 printable characters, etc.	£50.58
*ASC11 D/lux steel cab (IBM Blue)	£15.02
*Kluge prototype board (build your own circuits)	£12.83
*86 pin Gold plated connectors each	£4.00
*ELF Light pen writes/draws on TV screens	£6.50
*Video graphics board 32/64 characters by 16 lines on TV/monitor screens	£69.95
*ELF 11 Tiny basic on cassette	£13.50
*ELF 11 Bug/monitor powerful systems monitor/editor	£13.50
*T. Pitmans short course in programming manual (nil VAT)	£4.00
*T. Pitman short course on tiny basic manual (nil VAT)	£4.00
*RCA 1802 users manual (nil VAT)	£4.00
*On cassette Text Editor: Assembler, Disassembler (each)	£16.95

Save 10% and buy all three together.

All units can be supplied wired and tested

SPECIFICATION

★RCA 1802 8 bit microprocessor with 256 byte RAM expandable to 64K bytes
★RCA 1861 video IC to display program on TV screen via the RF Modulator
Single Board with professional hex keyboard fully decoded to eliminate the waste of memory for keyboard decoding circuits
Load, run and memory project switches
16 registers
Interrup, DMA and ALU
Stable crystal clock
Built in power regulator
5 slot plug in expansion bus (less connectors)

Send S.A.E. for comprehensive brochure

Name ....

Address ....

....

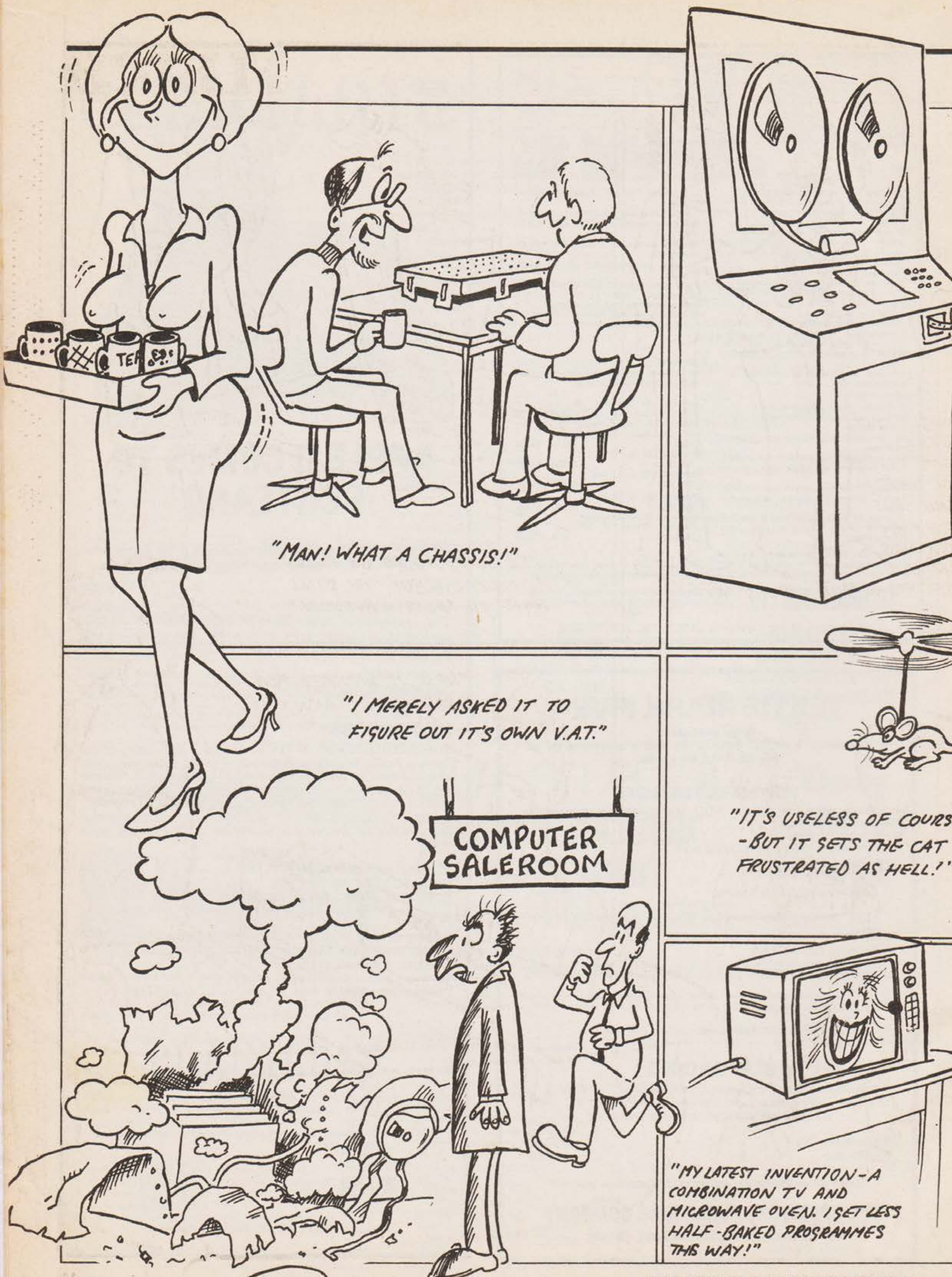
....

Postcode ....

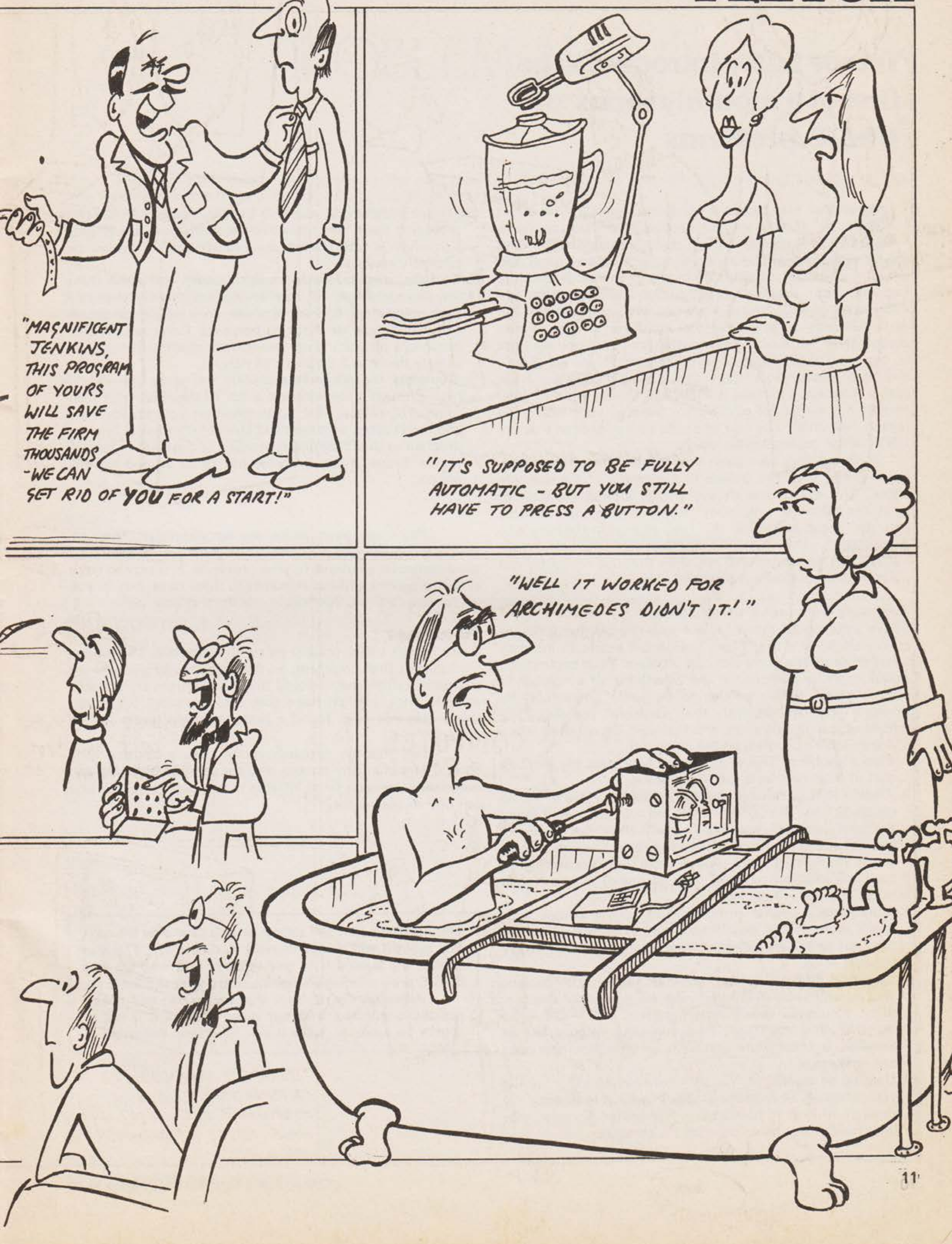
Barclaycard/Access

....

To Newtronics 138 Kingsland Road London E2 8BY Dept CT Tel: 01-739 1582. Sole UK Agents


TEA
"MAN! WHAT A CHASSIS!"
"I MERELY ASKED IT TO FIGURE OUT IT'S OWN V.A.T."
COMPUTER SALEROOM
"IT'S USELESS OF COURS
-BUT IT GETS THE CAT
FRUSTRATED AS HELL!"
"MY LATEST INVENTION-A COMBINATION TV AND MICROWAVE OVEN. I GET LESS HALF-BAKED PROGRAMMES THIS WAY!"

FLETCH


"MAGNIFICENT JENKINS, THIS PROGRAM OF YOURS WILL SAVE THE FIRM THOUSANDS -WE CAN GET RID OF YOU FOR A START!"
"IT'S SUPPOSED TO BE FULLY AUTOMATIC - BUT YOU STILL HAVE TO PRESS A BUTTON."
"WELL IT WORKED FOR ARCHIMEDES DIDN'T IT!"

Mr. T.L. Lusty.

Trevor Lusty introduces the first of a monthly series of BASIC problems

Suppose that someone gives you 1p. It has no effect at all. If you are given a thousand times as much, £10, it is nice but it will do little more than fill your car up with petrol. If you are given a thousand times this amount, £10,000, the situation begins to change. It certainly improves your financial standing, but it does not entirely relieve you of your financial worries. The whole amount can easily be spent on a new car, the deposit on a house, a new computer or just a nice long vacation, and then you are back to worrying about money again. However, if you are given a thousand times more than this, £10,000,000, the basic nature of your situation is changed! Unless you are foolhardy, it is no longer possible to spend all your money on personal expenses. The cost of a suit, a meal, or even a tanker full of petrol becomes meaningless.

Computers have given us a greater change in computation speed than the change in magnitude of the example above. This change has already greatly altered the view of what can be done, and many people believe that we have only just begun to exploit this basic change in the nature of computation. Our goal is to effectively and efficiently harness this computational speed so that our computer can quickly solve some of our problems.

Organizing Problems for the Computer

I have little doubt that you have experienced the difficulty of explaining to a friend how you would tackle an intricate situation or solve a fairly difficult problem. Your explanation would normally comprise of the description of a number of distinct steps, simple enough to be easily understood by someone not familiar with that particular problem. The following is a list of steps which should be checked when trying to solve a problem on the computer.

1. **Find a problem.** This is usually an easy step. My life, like that of most people, is full of problems.
2. **Decide if it is suitable for the computer.** This is by no means as easy. Not all problems are suitable – eg. I fancy the bird in the Computing Today T shirt, but I bet my computer can't help me! Some problems are too trivial and it would take more time to write the program than to solve the problem by hand. A good rule of thumb though is, 'If in doubt – have a go'.
3. **Formulate a precise problem.** This is not as easy as it might seem. Many problems, when examined in detail, turn out to be very different from what they were initially thought to be.
4. **Identify the important variables.** Make idealizations and assumptions about the variables and their relationships. Introduce quantitative measures. (36 – 24 – 36 perhaps) You may well find that your original list of variables is incomplete and have to introduce new ones at a later stage.
5. **Develop an algorithm.** You will need a method for solving your problem. A recipe is a cook's method for solving the problem of how to bake a cake. Remember, however, that your recipe must be suitable for the computer.
6. **Write a program.** If you have been thorough with the two preceding steps, this should not be a difficult task. If your algorithm is formulated as a flowchart, it should be relatively easy.
7. **Test the program.** Compute some results and check these for reasonableness. If they are unreasonable you should re-examine steps 4, 5 and 6 above. You should use simple and varied data for checking purposes. Don't assume that because a program gives the correct results with one set of data that it will do so for all data.
8. **Complete the documentation.** By this stage you should have a rough flowchart and a list of variables with their functions. Tidy up the documentation you already have and insert some comment (REMARK) statements into the program so that a minimum amount of time will be wasted in the future if the program has to be altered for any reason.

The steps given above are not definitive. They are meant as a guide and a 'check-list' to help you to formulate good computer solutions to your problems. It is easy to write simple programs without reference to these steps, but, as you get more ambitious, you will ignore them at your peril.

The Problems

Every month I shall present you with a problem. The following month I shall comment on the problem and give you my solutions. I shall never pretend that my solutions are the only possible ones, I might hope that they are good, but if you think that you have found a better solution please let me know.

This month's problem comes with a solution attached. There are many reasons why the solution given is poor, how many can you find? What is the longest road that you can find an answer for?

The problems that will appear on this page over the next few months will slowly increase in difficulty. The idea behind the feature is to give those of you who are into BASIC programming something to try your skills out on.

Although the problem that we present will not be actually connected with our series on BASIC it will of course be easier to solve if you have been following it.

PROBLEM PAGE

PROBLEM No. 1

There are eight houses on my side of the road and they have even numbers. I noticed that the sum of the house numbers on one side of my house equalled the sum of the house numbers on the other side.

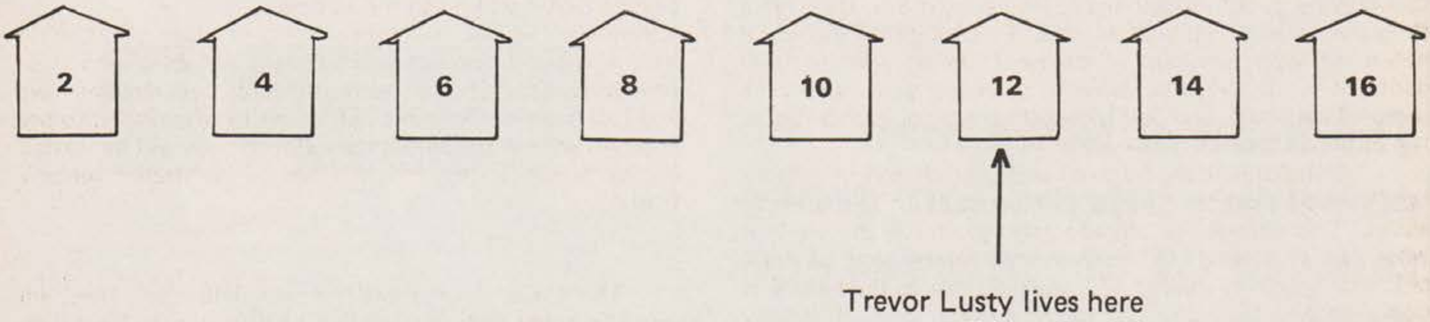
i.e. 2 + 4 + 6 + 8 + 10 = 30
14 + 16 = 30

My uncle lives on the odd side of a much longer road than mine. (more than 100 houses). Yet his house has the same property as mine, that the sum of the house numbers on one side quals the sum of the house numbers on the other.

Find possible values for :—
(1) His house number.
(2) The number of houses in the road.

Here is a solution, but there is a much better one — — can you find it?

Why is this program poor?
Can you find answers for longer roads?



SOLUTION NO. 1

```
100 REM MY UNCLE LIVES
IN HOUSE - N
120 LET N=1
130 LET N=N+2
140 REM SUM HOUSE NUMBERS
    ON LEFT - L
150 LET L=0
160 FOR I=1 TO N-2 STEP 2
170 LET L=L+I
180 NEXT I
190 REM SUM HOUSE NUMBERS
    ON RIGHT IN - R
200 REM STARTING AT HOUSE
    NUMBER - H
210 LET H=N
220 LET R=0
230 LET H=H+2
240 LET R=R+H
250 REM DOES THE LEFT SUM
    EQUAL THE RIGHT SUM?
260 IF L=R THEN 300
270 IF L>R THEN 230
280 GOTO 130
290 REM HOW MANY HOUSES?
300 LET T=(H+1)/2
310 IF T>100 THEN 330
320 GOTO 130
330 PRINT "THERE ARE ";T;
    "HOUSES IN THE ROAD"
340 PRINT "UNCLE LIVES AT
    HOUSE NUMBER ";N
350 END
```

The head of NPLs man-computer interaction group talks to CT about machine development and its impact

Dr. Chris Evans is, head of the man-computer interaction group at the National Physical Laboratory (NPL), was trained as a psychologist but for the past 15 years has been intimately involved in computer science. His particular areas of interest at present are the applications of computers in medical and education environments. Chris Evans talks to CT about the development of computers and some of the implications of the staggering drop in the cost of "Computing Power" over the past decade.

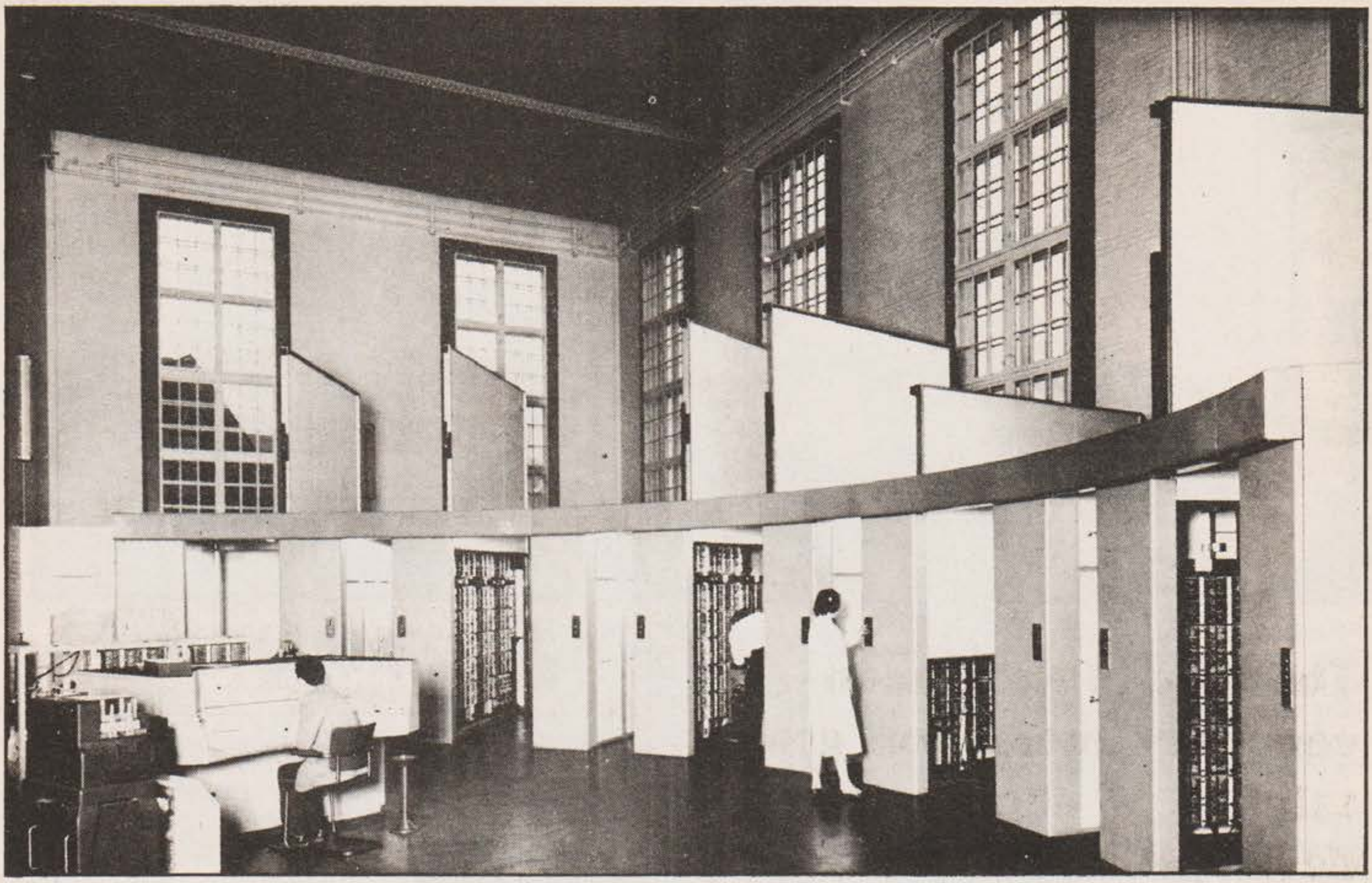
The first point to make clear is that computers are not a technological innovation but are an inevitable part of any complex monetary orientated society such as ours. Just as society could not function without modern transport facilities to ship people and cargo around the world so without some form of computing machines, man could not cope with the volume of data processing necessary to support our way of life.

This fact was recognised as long ago as the 17th century when the increasing amount of world trade created a demand for more and more clerks to record the details of cargo movements and to produce the necessary accounting information. This demanded for clerical skills was soon to outstrip the vailable supply of trained personnel. Pascal, a 17th century mathematician and engineer, recognised that if some means of replacing the slow, pen and paper, calculations that made up the bulk of a typical clerks job, then not only would the clerks work be speeded but unskilled labour could be used to operate the machined releasing skilled employees for more demanding work.

Cog In A Wheel

With this in mind Pascal produced a machine based on a system of cogs and wheels that would perform the basics of addition and subtraction mechanically. The machine was however bulky and very expensive, costing some £50 at 17th century prices. The machine could never provide a cost effective method of processing accounting information.

Pascal's machine was dedicated to performing one or two special tasks and it wasn't until the early 19th century Babbage proposed a general purpose machine that could be instructed to perform one of a given set of operations at any particular time, and that a series of such operations could be



Above: The original ACE computer developed at the NPL. It was used to solve the complex mathematical problems created in the various sections of the establishment. (Crown Copyright)
Left: The Commodore PET,smaller but with about the same power courtesy of the silicon chip !

assembled to sequentially process information in any manner desired. By having one central processing unit, performing many tasks, in sequence, instead of a series of dedicated assemblies, the cost of a computer could be considerably reduced.

The level of technology at which Babbage worked, however, still demanded that the machine be based on a series of cogs and wheels and once again no cost effective argument for the use of such a system could be made.

The next major development came about in the mid 19th century with the first census of the American population. The census was to take place every 10 years but some 7 years after the gathering of the data for the first such census the information had still not been processed into a useable form and by that time it, at any rate, was out of date. The method of handling the information, still pen and paper, was clearly never going to be able to cope with the volume of material generated.

What A Card

The administration decided to hold a competition designed to stimulate inventions that would make the task of producing census information a possibility. The winner was a machine designed by Hollerith.

Hollerith's device was based on a punched card system. The information about each individual was to be encoded. on a punched card. Then by a system of switches and levers these cards could rapidly be sorted into various categories and counted. The machine sped up the handling of the task by as much as 500 times.

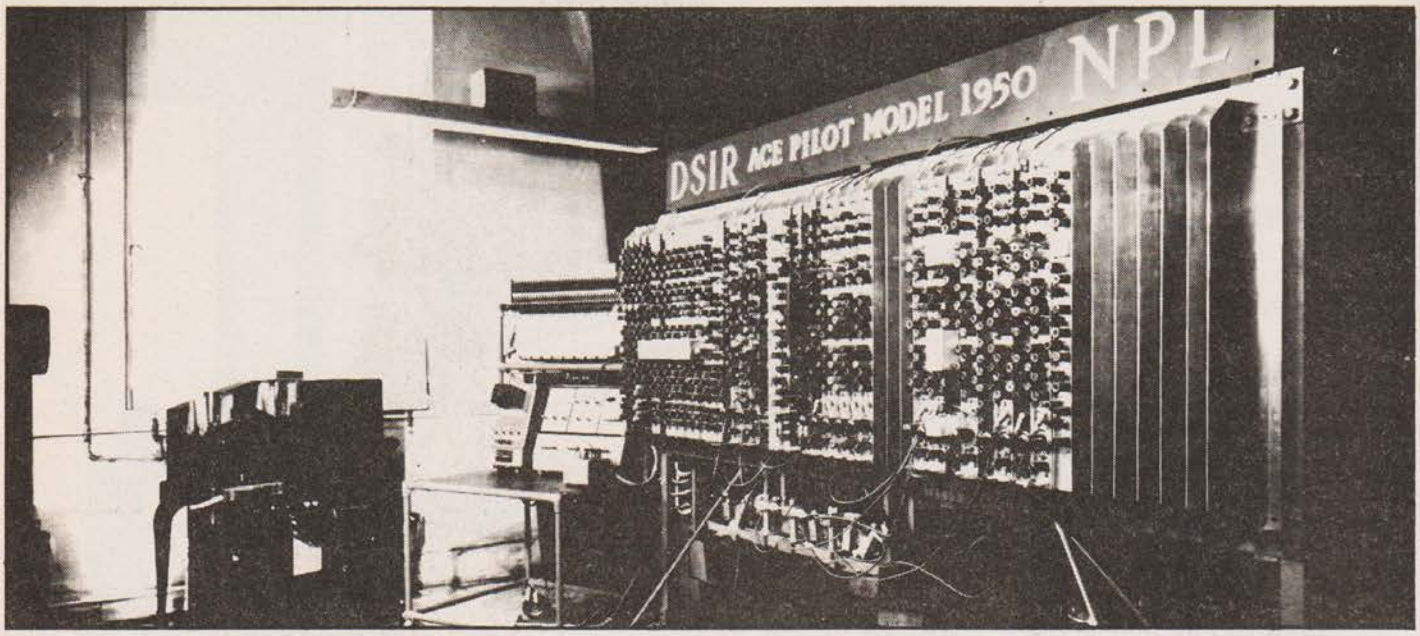
Machines based on Hollerith's principles were developed and refined to speed up many areas of information processing over the next few decades, but were essentially sorting devices as opposed to computing machines.

Puzzle Over Hitler

The first computer in a form that would probably be recognisable to us today was the famous Enigma machine used by the British to crack German coding systems during the second world war. The introduction of electronic switches, in the form of valves, greatly increased the speed at which arithmetic operations could be performed and resulted in a device far more powerful than a mechanically based system.

The Enigma machine was however still based on the decimal (base ten) system of arithmetic, which while tailored to human style computations is not very suited to a machines requirements. The binary (base two) arithmetic concept had been known for many years and its rules were well defined. The binary system represents numbers as a series of 1's and 0's. All the operations of addition, subtraction and multiplication and division can readily be performed using binary logic and the major advantage gained is the ease with which a computer can generate and manipulate the 1's and 0's of binary language using electronic switches that are either on (1) or off (0).

Binary machines based on valves were developed, one such being ACE which was for some time used at the NPL. Valves, though, were relatively bulky devices that consumed a lot of power and due to hand assembly techniques demanded by their intricate internal structure, expensive. In terms of size and power requirements, it was



The Pilot ACE machine. It was developed to test the ideas behind the full scale model. A section of the machine is now at the Science Museum. (Crown Copyright)

thought that a valve computer equal in power to the human brain would be the size of the Isle of Wight and require Niagara Falls to power it. Some thirty years ago the development of the transistor, with its small physical size, low power requirements and cost, removed the last barriers preventing the production of powerful, mass market computers.

Transistor Advantage

The commercial boom that followed took many by surprise but the advantages in speed and accuracy of processing by computer meant that the computer became an essential part of many business organisations. The cost of computers was still high, however, and only larger companies could justify the installation of their own machine, smaller operations had to resort to buying "spare time" on other peoples machines or making use of computer bureau facilities. A major part of the cost of a computer was still attributable to the amount of labour involved in assembly, for although transistors were cheap, assembling them into the circuits that made up a computer was time consuming. Developments in the field of semiconductor technology were soon to change this.

The transistor had been developed at the Bell Laboratories by a design team lead by William Schockley. *[Shockley left Bell to set up his own company together with 8 other scientists. After 2 years the 8 left after personal disagreements with Schockley and briefly joined Fairchild, soon moving on to set up their own companies in the bay area of San Francisco. The area was subsequently to be known as Silicon Valley. These companies headed by "The Fairchildren" (from Fairchild), each in keen competition were responsible for the latest revolution in the computer industry, the Large Scale Integrated circuit.]* This was a small chip of germanium, treated to produce a single active element. Germanium was not an ideal material however, suffering from a number of undesirable characteristics. Silicon on the other hand produced an almost perfect device and the refinement of manufacturing techniques that allowed more than one transistor to be integrated onto a single "chip" meant that the large logic blocks of a computer could be mass produced — resulting in large savings when assembling a complete computer. From the single transistor per chip in 1957, by 1963, 8 devices were being designed onto the chip and in 1977 a massive 250,000 transistors could be accommodated on a tiny chip of silicon.

Going back to the previous analogy with this level of sophistication it is possible to reduce the bulk of a computer with the same computing power as a human brain to a size not much bigger than the average car.

Vauxhall Brain?

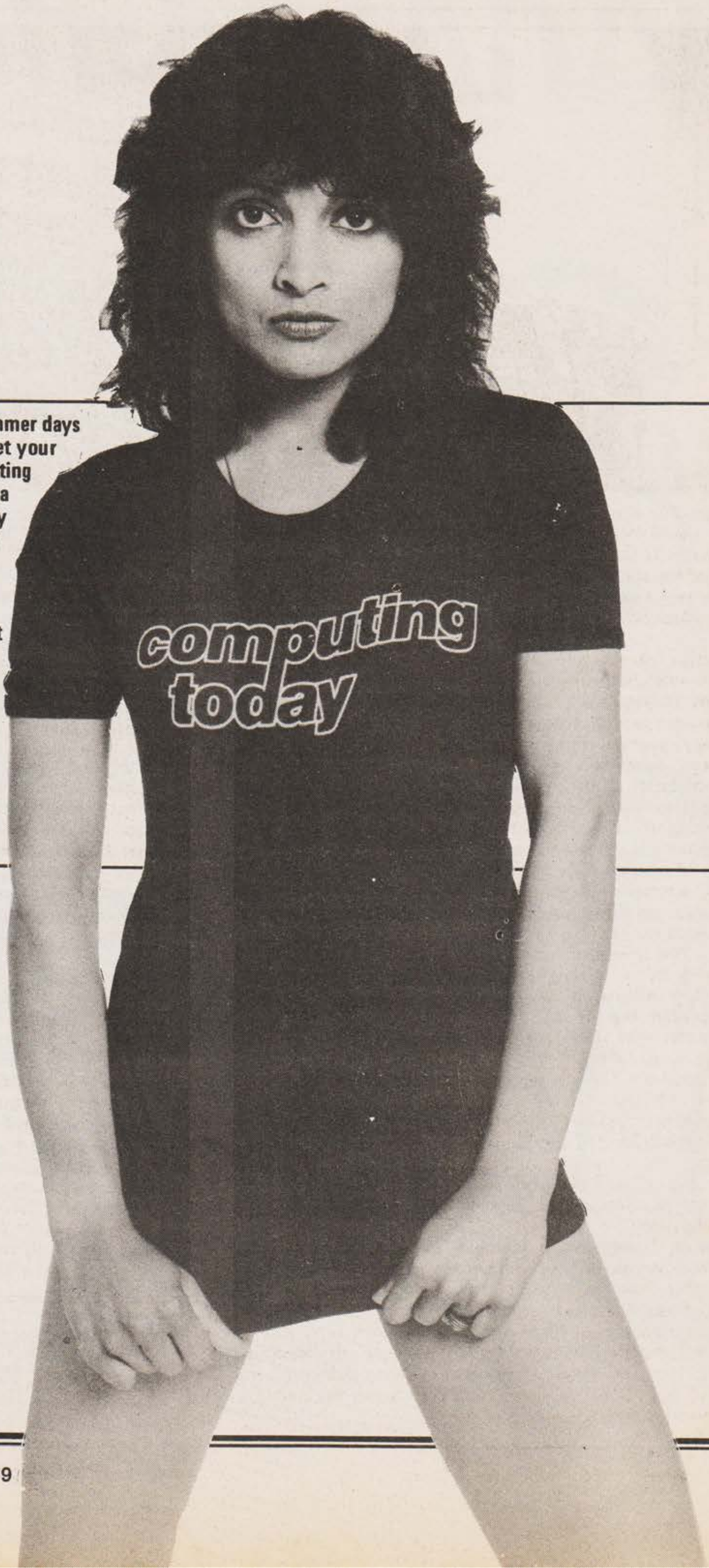
This massive processing power combined with mass manufacturing techniques led manufacturers to look for applications that needed this sort of performance coupled with a large possible market. The pocket calculator was one of the first products to satisfy these requirements, the digital watch being another.

These applications, like Pascal's machine of the 17th century, are deducated and it is the LSI circuit that echos Babbages idea of a general purpose computing machine, that is one of the most important products of LSI circuit development. This is the microprocessor.

MPU On The Moon

The Microprocessor is the Central Processing Unit (CPU) of a computer encapsulated into a single multi pin package. The MPU must be externally programmed by the user for it to be able to perform the specific operations demanded of it. Microprocessors have found applications ranging from controlling sewing machines and microwave ovens (where they replace traditional mechanical control systems) to producing chess playing machines, not forgetting making the Apollo moon landing possible. It's also the microprocessor that is responsible for the fact that today computers equal in power to the room sized machines costing 10's of thousands of pounds of 10 years ago can today be bought for less than £500 and will fit quite comfortably on a coffee table. And that really is where we came in . . . .

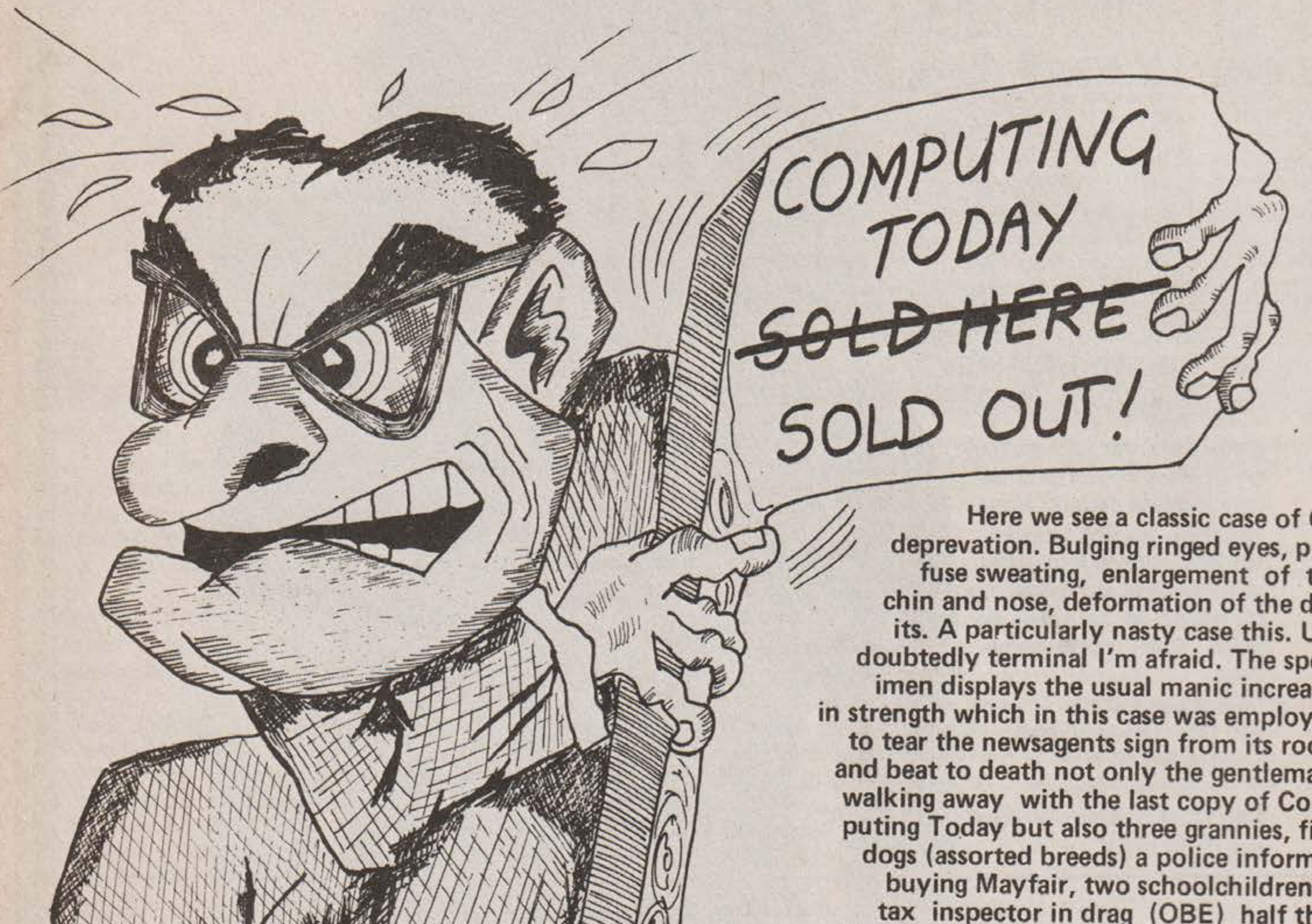
Dr. Chris Evans has recorded a series of interviews with some of the pioneers of computing in the early 50's and 60's, for the Science Museum. Copies of the tapes can be obtained from the Museum at Exhibition Road, London S.W.7. Dr. Evans has also published many books, among them the recently produced Dictionary of Psychology from Arrow.



With the promise of hot summer days ahead you will all want to get your hands on one of our Computing Today T shirts. Available at a bargain price of £2 each they are in black cotton with our logo in white on both front and back. We don't guarantee that you will look quite as stunning as our model but it's bound to be a talking point. Order yours today from:
Computing Today,
145 Charing Cross Road,
London WC2H 0EE

Don't forget to give us your size (S, M or L)!

ARRRGGGHHH.....



Here we see a classic case of CT deprevation. Bulging ringed eyes, profuse sweating, enlargement of the chin and nose, deformation of the digits. A particularly nasty case this. Undoubtedly terminal I'm afraid. The specimen displays the usual manic increase in strength which in this case was employed to tear the newsagents sign from its roots and beat to death not only the gentleman walking away with the last copy of Computing Today but also three grannies, five dogs (assorted breeds) a police informer buying Mayfair, two schoolchildren, a tax inspector in drag (OBE) half the County Cricket Club and a passing coach load of nuns, before being brought down by the entire workforce of the local iron foundary. Of course all this could have been avoided oh so easily had the poor unfortunate taken out a subscription to CT and had a years supply delivered to his door. As it is the nurse is gonna have a hell of a time trying to get the magazines through the letterbox of the padded cell. Why not make the world a safer place and take out a sub yourself now — before the hands begin to tremble and the eyes begin to spin. It only costs £8.00 a year (£9.00 overseas surface mail) — a small price to pay.

Computing Today
Subscriptions Dept.,

M.A.P. PO Box 35,
Bridge Street,
Hemel Hempstead,
Hertfordshire.

Gentlemen, the Petdisk has landed £499

(single disk)

The U.K.-designed and manufactured Novapak disk system for Commodore's PET*, first seen at Compec '78, is (after extensive industrial evaluation), now available to the domestic user. Its unique saddle configuration continues the integrated design concept of your PET, with no trailing wires or bulky desk-top modules.

- ***Novapac** may be used with any available RAM plane.
- *May be used with latest versions of PET.
- ***Data transfer** takes place at 15,000 char/sec — effectively 1,000 times faster than cassette!
- ***Storage capacity** is 125 K/bytes (unformatted) on 40 tracks per diskette side.
- ***Dual index sensors** permit dual-side recording for 250 K/bytes per diskette.
- ***Easy operation** full-width doors prevent media damage.
- ***System expandable** to ½ Mbyte on-line storage (4 drives).
- ***Dual head and 2D** versions provide 2 Mbytes on-line.
- ***Industry Standard IBM 3740** recording format for industry-wide media compatibility offered only by NOVAPAK.
- ***Dedicated Intel 8048** microprocessor and 1771 FDC minimise PET software overhead.
- ***Local hardware and software** support available, including applications, packages for small business use.

The sophisticated Disk Operating System is disk-resident, which allows for future DOS-enhancements without hardware alterations. PDOS supports multiple file handling, allocating disk space dynamically to each as and when necessary. Any file may occupy from 1 to 600 sectors as required, at up to 16 non-contiguous locations on the disk, PDSO may be used alone, or within a BASIC program and offers user-specified password security for any file. Multiple access-modes simplify BASIC program construction, and the user may generate tailored DOS modules.

Novapac dual-disk system complete with PDOS and BASIC demonstration programs on disc **£899** + VAT.

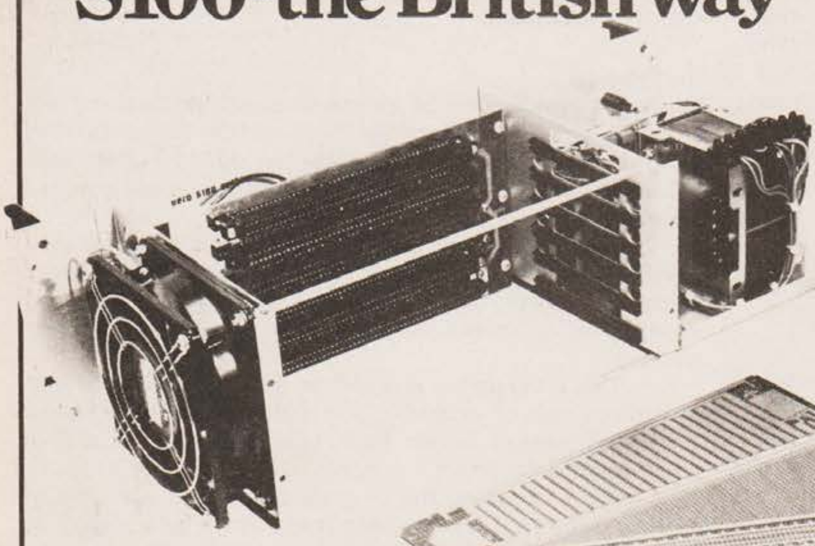
Available from the manufacturer or selected dealers.
Terms: 50% with order, balance on delivery.
Full cash with order is subject to 5% discount.
VAT-FREE Export arranged (Must be shipped by us).

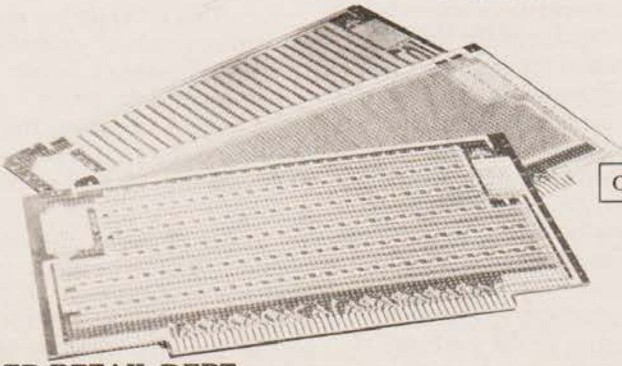
analog electronics

47 Ridgeway Ave,
Coventry
Tel: (0203) 417761

S100-the British way

The Vero S100 Sub Rack is a 19" rack mountable development kit, complete with its own power supply and backplane motherboard, for the construction and evaluation of microprocessor based systems to the S100 format. The power supply provides three voltage levels — +8V, +18V and -18V. The Sub Rack has its own cooling fan providing airflow across the boards and the power supply. A full range of allied items to enable a complete system to be constructed are available.



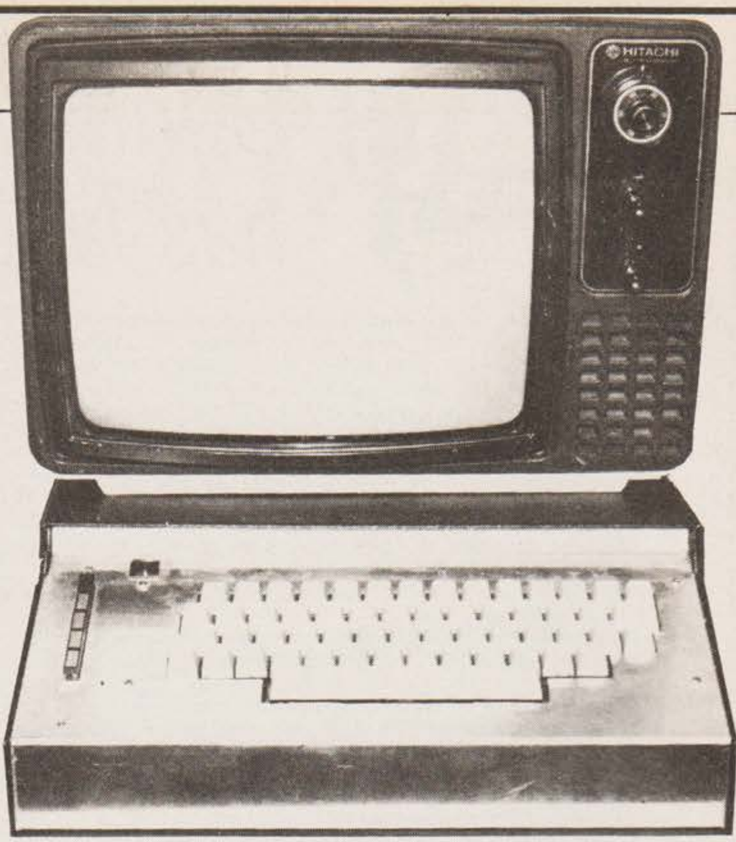


ORDER CODE	ITEM DESCRIPTION
188-2341H	S100 Sub Rack
06-0095L	S100 Dip Board
06-2337L	S100 High Density Board
06-2338F	S100 Square Pad Board
15-1630K	Compatible Connector (Solderlug)
15-1632L	Compatible Connector (Miniwrap)
09-2340H	S100 Extender Board
48-8345K	Mk. II D Series Case
75-2867G	Keyboard Console
79-1729L	Verowire Wiring Kit

vero

VERO ELECTRONICS LTD RETAIL DEPT.
Industrial Estate, Chandler's Ford,
Hampshire SO5 3ZR
Tel: (04215) 62829

John Hiscott



Keep track of your books with Triton

This personal accounting program is written for Triton using 8K RAM card, 7K floating point BASIC and teletype. Simple double-entry book-keeping is employed in order to ensure arithmetical accuracy and the program provides for analysis of all income or expenditure according to the requirements of the user in the form of a list of balances which can be printed either before or after data is entered. Procedural instructions appear on the VDU. Provision has also been made for selected account balances to be expressed as a percentage of income. The information printed for each individual transaction gives the date (DDMM), cheque or reference number, serial number for the entry (computed automatically providing a continuity check for each printout) and the name of, and amount entered into, the two or more accounts involved.

Double-entry necessitates two or more entries for each transaction – for example:–

1 – Electricity bill paid by cheque. A negative amount would be shown for both the electricity a/c and the bank account.

2 – Salary credited to bank account. A positive amount would be shown for both the salary account and the bank account.

3 – Gas bill wrongly charged to electricity account. A negative amount would be shown for the gas account and a positive amount for the electricity account thus leaving overall totals unaltered whilst correcting the two accounts involved.

4 – A transfer of funds between bank, building society or post office accounts would entail two separate transactions. Firstly negative entries in both the account giving the funds and the 'transfers' account. Secondly positive entries in both the account receiving the funds and the 'transfers' account. A zero balance would then result in the 'transfer' account whilst the bank etc. balances would be adjusted to reflect the movement of funds.

The program provides for the retention of text and balances (as variables) on cassette and the final screen display gives the necessary start and end addresses. Line number 0 limits the amount of memory used to avoid needless recording of zero bytes but there is nevertheless a useful amount of free space for direct additions to the program. This can easily be verified by executing a hexdump with the program loaded. To return to BASIC from the monitor you must enter 'G' with a start address E017 in order not to clear the program.

It is most important to remember that the RUN command initialises all variables and must not, repeat not be used after loading from cassette. Instead use 'GOTO 0' as a 'PSEUDO' command.

The Program

The listing provides for 34 income/expenditure accounts and 4 bank, building society or post office accounts. The accounts have been named to cover the usual headings with the remainder labelled as 'spare' – these can be easily edited to suit the needs of the user.

The number of accounts which can be accommodated is dependant on the number of variables and arrays which can be fitted into the available memory and could well be sufficient to enable the program to be used for business purposes.

This program has enabled the author to dispense with the tedious task of writing-up and casting manually prepared accounting records which have been replaced by a ring-binder to hold the prints.

When initiating the program do not request a list of balances until an entry has been made in the salary account otherwise the percentage calculations will produce an error message.

To transfer the balances from an existing manual accounting system make one composite input entry as for example in Figure 1. If starting with a bank balance for which no analysis is available, enter the amount (positive) in the opening balances account (no.32) and in the bank account (no. 91). See example Figure 2. Figure 3 shows speciment input enyries and Figure 4 shows the resulting listing of balances.

After the final list of balances has been printed for

TRITON ACCOUNTS

```
DATE OF LAST PRINT...... 0
DATE OF LAST TRANSACTION 0
NUMBER OF LAST ENTRY.... 0
*************************
DATE OF THIS PRINT 1807
-----------------------
PERSONAL ACCOUNTS - 1979
***********************
TRANSACTION DATE..... 1807
ENTRY NUMBER......... 1
CHEQUE/REFERENCE NO.. 0
COMPUTER EQUIPMENT..>-39.00
WIFE'S ALLOWANCE....>-175.00
ELECTRICITY.........>-55.85
GARAGE RENT.........>-38.75
GAS.................>-109.50
HOUSEKEEPING........>-478.00
MILK................>-28.70
GENERAL RATES.......>-130.00
ROAD FUND LICENCE...>-50.00
MORTGAGE REPAYMENT..>-475.78
TELEPHONE CHARGES...>-17.74
TV LICENCE FEE......>-25.00
WATER RATES.........>-18.42
SALARY - NET........>1822.49
BANK ACCOUNT........>130.75
*************************
TOTAL INCOME/EXPENDITURE.......... 130.75
TOTAL BANK AND B/S BALANCES....... 130.75
BALANCES AGREE
```

Fig.1. A composite entry to start off the Account package.

```
DATE OF LAST PRINT...... 0
DATE OF LAST TRANSACTION 0
NUMBER OF LAST ENTRY.... 0
*************************
DATE OF THIS PRINT 1807
-----------------------
PERSONAL ACCOUNTS - 1979
***********************
TRANSACTION DATE..... 1807
ENTRY NUMBER......... 1
CHEQUE/REFERENCE NO.. 0
OPENING BALANCES....>130.75
BANK ACCOUNT........>130.75
*************************
TOTAL INCOME/EXPENDITURE.......... 130.75
TOTAL BANK AND B/S BALANCES....... 130.75
BALANCES AGREE
```

Fig.2. Starting the bank balance.

```
DATE OF LAST PRINT...... 0
DATE OF LAST TRANSACTION 0
NUMBER OF LAST ENTRY.... 0
**************************
DATE OF THIS PRINT 1807
-----------------------
PERSONAL ACCOUNTS - 1979
***********************
TRANSACTION DATE..... 107
ENTRY NUMBER......... 1
CHEQUE/REFERENCE NO.. 0
SALARY - NET........>320.70
BANK ACCOUNT........>320.70
*************************
TRANSACTION DATE..... 507
ENTRY NUMBER......... 2
CHEQUE/REFERENCE NO.. 1111
GAS.................>-58.87
BANK ACCOUNT........>-58.87
*************************
TRANSACTION DATE..... 1107
ENTRY NUMBER......... 3
CHEQUE/REFERENCE NO.. 1112
HOUSEKEEPING........>-110.00
BANK ACCOUNT........>-110.00
*************************
TRANSACTION DATE..... 1807
ENTRY NUMBER......... 4
CHEQUE/REFERENCE NO.. 1113
TELEPHONE CHARGES...>-19.88
BANK ACCOUNT........>-19.88
*************************
TRANSACTION DATE..... 1807
ENTRY NUMBER......... 5
CHEQUE/REFERENCE NO.. 1114
HOUSEKEEPING........>-25.00
BANK ACCOUNT........>-25.00
*************************
TOTAL INCOME/EXPENDITURE.......... 106.95
TOTAL BANK AND B/S BALANCES....... 106.95
BALANCES AGREE
```

Fig.3. Specimen entries to the bank balance.

```
BALANCES C/F - PRINT DATE 1807
-----------------------

1  SPARE.................. 0.00
2  CAR COSTS & PETROL..... 0.00 (-0.00 % SALARY)
3  SPARE.................. 0.00
4  COMPUTER EQUIPMENT..... 0.00
5  WIFE'S ALLOWANCE....... 0.00 (-0.00 % SALARY)
6  LOANS TO WIFE.......... 0.00
7  POCKET MONEY - CHILDREN 0.00
8  DECORATING............. 0.00 (-0.00 % SALARY)
9  ELECTRICITY............ 0.00 (-0.00 % SALARY)
10 FURNITURE ETC.......... 0.00 (-0.00 % SALARY)
11 GARAGE RENT............ 0.00
12 GAS...................-58.87 ( 18.36 % SALARY)
13 HOUSEKEEPING..........-135.00 ( 42.10 % SALARY)
14 INTEREST RECEIVED/PAID. 0.00
15 MISCELLANEOUS.......... 0.00
16 MILK................... 0.00
17 SPARE.................. 0.00
18 SPARE.................. 0.00
19 RATES.................. 0.00 (-0.00 % SALARY)
20 REDIFFUSION............ 0.00
21 ROAD FUND LICENCE...... 0.00
22 SALARY................. 320.70
23 MORTGAGE REPAYMENT..... 0.00 (-0.00 % SALARY)
24 HEATING MAINTENANCE.... 0.00 (-0.00 % SALARY)
25 TELEPHONE.............-19.88
26 TV LICENCE............. 0.00
27 WASHING MACHINE REPAIR. 0.00
28 WATER RATES............ 0.00 (-0.00 % SALARY)
29 TRAIN FARES............ 0.00
30 CAR PARTS/SERVICING.... 0.00
31 BANK CHARGES........... 0.00
32 OPENING BALANCES....... 0.00
33 TRANSFERS.............. 0.00
34 INVESTMENTS............ 0.00
                          --------
TOTAL INCOME/EXPENDITURE IS... 106.95
                          --------
91 BANK BALANCE........... 106.95
92 BLDG. SOC. BALANCE..... 0.00
93 POST OFFICE SAVINGS A/C 0.00
94 SPARE (BANK) BALANCE... 0.00
                          --------
TOTAL BANK AND B/S BALANCES... 106.95
                          --------
```

Fig.4. Output listing for the Account.

the year (or other period of account) the heading can be suitably altered and all variables cleared to start the new account. The previous total income/expenditure should then be entered in the 'opening balances' account and the bank etc. balances reinstated as they were. The analysis of income/expenditure then starts afresh. The program should run with a minimum of modification on other (floating point) BASIC's. The statements which might need alteration are:—

> <> = not equal to.
> PRINT #0# = print whole numbers without decimals.
> PRINT #2Z# =print 2 decimal places with zeros.
> subscripted variables i.e. C1, C2, C3 etc.
> POKE 5121, 0 = switch to VDU.
> POKE 5121,85 = switch to printer.
> POKE statements in line 0:— These limit the amount of memory available to the BASIC interpreter and are related to the start and finish addresses given at the end of the program for the transfer to cassette.

The requirement in this connection is that adequate memory space should be made available for text and the variables to be saved. The program as written provides 7.5K bytes. The 7K floating point BASIC establishes Triton as a most useful instrument in the field of home management, club and society accounting and this program could well be expanded to embrace income tax, VAT and budgetary control.

```
PERSONAL ACCOUNTS - PROGRAM LISTING
***********************************

>LIST
0 POKE5970,00;POKE5971,52
1 J1=0;01=0
2 POKE5121,85
4 PRINT#0#
5 PRINT"DATE OF LAST PRINT......",K1
6 PRINT"DATE OF LAST TRANSACTION",T6
7 PRINT"NUMBER OF LAST ENTRY....",S
10 PRINT"***************************"
15 POKE5121,0
25 PRINT"ENTER DATE OF THIS PRINT"
30 INPUT,K1
33 POKE5121,85
35 PRINT#0#,;PRINT"DATE OF THIS PRINT",K1
40 PRINT"------------------------"
50 PRINT"PERSONAL ACCOUNTS - 1979"
60 PRINT"************************"
61 PRINT#2Z#,
62 POKE5121,00
65 PRINT"TO PRINT OPENING BALANCES ENTER 70"
66 PRINT"TO RECORD TRANSACTION ENTER 820"
67 INPUT,01
68 GOTO01
70 POKE5121,85;GOTO80
75 POKE5121,85;PRINT#0#;PRINT"BALANCES C/F - PRINT DATE",K1;GOTO90
80 PRINT"BALANCES BROUGHT FORWARD"
90 PRINT"------------------------"
95 PRINT#2Z#
100 PRINT"1  SPARE....................",B1
110 PRINT"2  CAR COSTS + PETROL.....",B2,"(",B2/D4*-100,"% SALARY)"
120 PRINT"3  SPARE....................",B3
130 PRINT"4  COMPUTER EQUIPMENT.....",B4
140 PRINT"5  WIFE'S ALLOWANCE.......",B5,"(",B5/D4*-100,"% SALARY)"
150 PRINT"6  LOANS TO WIFE..........",B6
160 PRINT"7  POCKET MONEY - CHILDREN",B7
170 PRINT"8  DECORATING.............",B8,"(",B8/D4*-100,"% SALARY)"
180 PRINT"9  ELECTRICITY............",B9,"(",B9/D4*-100,"% SALARY)"
190 PRINT"10 FURNITURE ETC..........",C1,"(",C1/D4*-100,"% SALARY)"
200 PRINT"11 GARAGE RENT............",C2
210 PRINT"12 GAS....................",C3,"(",C3/D4*-100,"% SALARY)"
220 PRINT"13 HOUSEKEEPING...........",C4,"(",C4/D4*-100,"% SALARY)"
230 PRINT"14 INTEREST RECEIVED/PAID.",C5
240 PRINT"15 MISCELLANEOUS..........",C6
250 PRINT"16 MILK...................",C7
260 PRINT"17 SPARE..................",C8
270 PRINT"18 SPARE..................",C9
280 PRINT"19 RATES..................",D1,"(",D1/D4*-100,"% SALARY)"
290 PRINT"20 REDIFFUSION............",D2
300 PRINT"21 ROAD FUND LICENCE......",D3
310 PRINT"22 SALARY.................",D4
320 PRINT"23 MORTGAGE REPAYMENT.....",D5,"(",D5/D4*-100,"% SALARY)"
330 PRINT"24 HEATING MAINTENANCE....",D6,"(",D6/D4*-100,"% SALARY)"
340 PRINT"25 TELEPHONE..............",D7
350 PRINT"26 TV LICENCE.............",D8
360 PRINT"27 WASHING MACHINE REPAIR.",D9
370 PRINT"28 WATER RATES............",E1,"(",E1/D4*-100,"% SALARY)"
380 PRINT"29 TRAIN FARES............",E2
390 PRINT"30 CAR PARTS/SERVICING....",E3
400 PRINT"31 BANK CHARGES...........",E4
410 PRINT"32 OPENING BALANCES.......",E5
420 PRINT"33 TRANSFERS..............",E6
430 PRINT"34 INVESTMENTS............",E7
500 T1=B1+B2+B3+B4+B5+B6+B7+B8+B9
510 T2=C1+C2+C3+C4+C5+C6+C7+C8+C9
520 T3=D1+D2+D3+D4+D5+D6+D7+D8+D9
530 T4=T1+T2+T3+E1+E2+E3+E4+E5+E6+E7
540 PRINT"                                  ---------"
550 PRINT"TOTAL INCOME/EXPENDITURE IS...",T4
560 PRINT"                                  ---------"
600 PRINT"91 BANK BALANCE...........",F1
610 PRINT"92 BLDG. SOC. BALANCE.....",F2
620 PRINT"93 POST OFFICE SAVINGS A/C",F3
630 PRINT"94 SPARE (BANK) BALANCE...",F4
650 T5=F1+F2+F3+F4
700 PRINT"                                  ---------"
710 PRINT"TOTAL BANK AND B/S BALANCES...",T5
720 PRINT"                                  ---------"
750 IFJ1=75GOTO9000
800 PRINT
810 POKE5121,00
820 PRINT
822 PRINT##,
823 PRINT"ENTER TRANSACTION DATE DDMM",;INPUT,T6
824 PRINT"ENTER CHEQUE OR REFERENCE NUMBER - LAST FOUR ONLY",
825 INPUT,Q1
826 POKE5121,85;PRINT#0#,;PRINT"TRANSACTION DATE.....",T6
827 S=S+1
828 PRINT"ENTRY NUMBER.........",S
829 PRINT"CHEQUE/REFERENCE NO..",Q1
830 POKE5121,00;PRINT"ENTER A/C NO. OR 0 TO COMPLETE ENTRY",
832 INPUT,T7
840 PRINT#2Z#,
850 IFT7=0GOTO2000
855 POKE5121,85
860 GOTOT7+870
871 PRINT"SPARE...............",;INPUT,V1;B1=B1+V1;GOTO830
872 PRINT"CAR/PETROL..........",;INPUT,V2;B2=B2+V2;GOTO830
873 PRINT"SPARE...............",;INPUT,V3;B3=B3+V3;GOTO830
874 PRINT"COMPUTER EQUIPMENT..",;INPUT,V4;B4=B4+V4;GOTO830
875 PRINT"WIFE'S ALLOWANCE....",;INPUT,V5;B5=B5+V5;GOTO830
876 PRINT"LOANS TO WIFE.......",;INPUT,V6;B6=B6+V6;GOTO830
877 PRINT"POCKET MONEY........",;INPUT,V7;B7=B7+V7;GOTO830
878 PRINT"DECORATING..........",;INPUT,V8;B8=B8+V8;GOTO830
879 PRINT"ELECTRICITY.........",;INPUT,V9;B9=B9+V9;GOTO830
880 PRINT"FURNITURE ETC.......",;INPUT,W1;C1=C1+W1;GOTO830
881 PRINT"GARAGE RENT.........",;INPUT,W2;C2=C2+W2;GOTO830
882 PRINT"GAS.................",;INPUT,W3;C3=C3+W3;GOTO830
883 PRINT"HOUSEKEEPING........",;INPUT,W4;C4=C4+W4;GOTO830
884 PRINT"INTEREST REC'D/PAID.",;INPUT,W5;C5=C5+W5;GOTO830
885 PRINT"MISCELLANEOUS.......",;INPUT,W6;C6=C6+W6;GOTO830
886 PRINT"MILK................",;INPUT,W7;C7=C7+W7;GOTO830
887 PRINT"SPARE...............",;INPUT,W8;C8=C8+W8;GOTO830
888 PRINT"SPARE...............",;INPUT,W9;C9=C9+W9;GOTO830
889 PRINT"GENERAL RATES.......",;INPUT,X1;D1=D1+X1;GOTO830
890 PRINT"REDIFFUSION RENTAL..",;INPUT,X2;D2=D2+X2;GOTO830
891 PRINT"ROAD FUND LICENCE...",;INPUT,X3;D3=D3+X3;GOTO830
892 PRINT"SALARY - NET........",;INPUT,X4;D4=D4+X4;GOTO830
893 PRINT"MORTGAGE REPAYMENT..",;INPUT,X5;D5=D5+X5;GOTO830
894 PRINT"HEATING MAINTENANCE.",;INPUT,X6;D6=D6+X6;GOTO830
895 PRINT"TELEPHONE CHARGES...",;INPUT,X7;D7=D7+X7;GOTO830
896 PRINT"TV LICENCE FEE......",;INPUT,X8;D8=D8+X8;GOTO830
897 PRINT"WASHING M/C REPAIRS.",;INPUT,X9;D9=D9+X9;GOTO830
898 PRINT"WATER RATES.........",;INPUT,Y1;E1=E1+Y1;GOTO830
899 PRINT"TRAIN FARES.........",;INPUT,Y2;E2=E2+Y2;GOTO830
900 PRINT"CAR PARTS/SERVICING.",;INPUT,Y3;E3=E3+Y3;GOTO830
901 PRINT"BANK CHARGES........",;INPUT,Y4;E4=E4+Y4;GOTO830
902 PRINT"OPENING BALANCES....",;INPUT,Y5;E5=E5+Y5;GOTO830
903 PRINT"TRANSFERS...........",;INPUT,Y6;E6=E6+Y6;GOTO830
904 PRINT"INVESTMENTS.........",;INPUT,Y7;E7=E7+Y7;GOTO830
2000 PRINT"ENTER BANK OR B/S CODE"
2002 PRINT"BANK=91 B/S=92 POSB=93 SPARE=94"
2004 PRINT"ADJUSTING ENTRY=95",;INPUT,S1;POKE5121,85
2010 GOTOS1+2000
2091 PRINT"BANK ACCOUNT........",;INPUT,S2;F1=F1+S2;GOTO3000
2092 PRINT"BUILDING SOCIETY A/C",;INPUT,S3;F2=F2+S3;GOTO3000
2093 PRINT"POST OFFICE ACCOUNT.",;INPUT,S4;F3=F3+S4;GOTO3000
2094 PRINT"SPARE (BANK) ACCOUNT",;INPUT,S5;F4=F4+S5;GOTO3000
2095 PRINT"ADJUSTING ENTRY ONLY";GOTO3000
2100 POKE5121,00
3000 PRINT"****************************"
3005 POKE5121,00;PRINT"TO MAKE FURTHER ENTRY ENTER 1"
3010 PRINT"TO PRINT TOTALS AND CLOSE ENTER 0",
3020 INPUT,S6
3030 IFS6=1GOTO823
3040 IFS6=0GOTO4000
4000 M1=B1+B2+B3+B4+B5+B6+B7+B8+B9
4010 M2=C1+C2+C3+C4+C5+C6+C7+C8+C9
4020 M3=D1+D2+D3+D4+D5+D6+D7+D8+D9
4030 M4=M1+M2+M3+E1+E2+E3+E4+E5+E6+E7
4035 POKE5121,85
4050 PRINT"TOTAL INCOME/EXPENDITURE..........",M4
5000 M5=F1+F2+F3+F4
5010 PRINT"TOTAL BANK AND B/S BALANCES.......",M5
5020 IFM4=M5PRINT"BALANCES AGREE"
5025 IFM4=M5GOTO8000
5030 IFM4<>M5PRINT"BALANCES DISAGREE - PLEASE EXAMINE"
5040 PRINT"THE ENTRIES AND ADJUST";GOTO830
8000 POKE5121,00;PRINT"TO PRINT CLOSING BALANCES ENTER 75"
8010 PRINT"TO FINISH ENTER 9000"
8020 INPUT,J1;GOTOJ1
9000 POKE5121,00;PRINT"END OF RUN"
9010 PRINT"**********TRANSFER TO TAPE************"
9020 PRINT"(START ADDRESS 5800 - END ADDRESS 13500)"
9030 END
```

P.E.T NEWS

PET BOOM IN BRITAIN!

AUTHORISED PET COMMODORE DEALERS

Birmingham
Camden Electronics
021-773-8240

CPS (Data Systems)
021-707-3866

Taylor Wilson Systems
Knowle (056-45) 6192

Bolton
B & B Consultants
0204-26644

Bournemouth
Stage One Computers
0202-23570

Bradford
Ackroyd T/W &
Add M/C
0274-31835

Brentwood Essex
Direct Data Marketing
0277-229379

Bristol
Bristol Computer Centre
0272-23430

Sumlock Tabdown
0272-26685

Cambridge
Cambridge Computer
Store 0223-68155

Cardiff
Sigma Systems
0222-21515

Colchester
Dataview Ltd
0206-78811

Derby
Davidson Richards
0332-366803

Durham
Dyson Instruments
0385-66937

Edinburgh
Micro Centre
031-225-2022

Exeter
A.C. Systems
0392-71718

Grimsby
Allen Computers
0472-40568

Hemel Hempstead
Data Efficiency
0442-57137

Hove
Amplicon Micro Systems
0273-720716

Leeds
Holdene 0532-459459

Liverpool
Dams Office Equipment
051-227-3301

Rockliff Bros
051-521-5830

London E2
Ragnorok Electronic
Systems 01-981-2748

THE PET 4K COMPUTER at £450 (+VAT) it's got to be Britains best buy!

Why the PET is the number one Personal Computer

The Self Contained PETS
The self contained PET models 2001-4 and 8 come complete with TV screen, keyboard and built-in cassette deck as well as the computer circuitry. They are simply plugged into any 13 amp mains and no special knowledge is needed for running standard programs — over 200 of which are available on cassettes.

The Big Memory PETS
The BIG MEMORY PETS contain the same main features as for the 2001-4 and 8 models except that they incorporate a full typewriter size keyboard and have larger internal memory of 16K and 32K bytes RAM respectively.

Professional Printers

High specification printers can print onto paper all the PET characters — letters, numbers and graphics. A tractor feed model has the advantage of accepting mailing labels, using standard pre-printed forms. The only connections required are an A/C lead and PET connecting leads. The PET is programmable, allowing the printer to format print and it accepts 8½" paper giving up to four copies.

Dual Drive Floppy Disc

The Dual Drive Floppy Disk is the latest in Disk technology with extremely large storage capability and excellent file management.

The Floppy Disk operating system used with the PET computer enables a program to read or write data in the background while simultaneously transferring data to the PET. The Floppy Disk is a reliable low cost unit, and is convenient for high speed data transfer. Due to the latest technological advances incorporated in this disk, a total of 360K bytes are available in the two standard 5¼" disks.

Only two connections are necessary — an A/C lead and PET interface lead.

Software and application areas for PET

There are a large number of programs that can readily be used with the PET. Personalised programming is available from many Authorised Dealers. Over 200 programs are now available from Commodore and other software suppliers for the PET. Popular program titles include Stock Control, Statistics, Payroll, Strathclyde Basic Course, Chess, Lunar Landing and Education Packs.

FOR THE COMMERCIAL USER
The Commodore PET offers for the first time a really cost effective business computer for use in Accountancy, Statistics, Stock Control, Payroll, Invoicing etc.

FOR THE SCIENTIST AND THE LABORATORY
PET has a comprehensive set of scientific functions making it a far superior tool to the best programmable calculators. PET interfaces directly with hundreds of laboratory instruments. PET is an ideal industrial and commercial controller.

FOR THE EDUCATIONAL WORLD
The extensive basic language makes PET an ideal tool for teaching computer programming. Programs can be written to "tutor" the user (pupil) in almost any discipline, including BASIC itself. And, of course, the PET can be used to take care of school records, exam results, attendance figures etc.

IN THE HOME
The PET is an extremely creative and instructive learning medium of the future for young and old alike. There are also large numbers of entertainment programmes available including chess and space games.

☎ Phone about PET now!

CALL YOUR LOCAL AUTHORISED DEALER or in case of difficulty contact:
COMMODORE SYSTEMS DIVISION
360 Euston Road, London. Tel: 01-388-5702

London EC1
Sumlock Bondain
01-253-2447

London N14
Micro Computation
01-882 5104

London NW4
Da Vinci Computers
01-202-9630

London SW14
Micro Computer Centre
01-876-6609

London WS
Adda Computers
01-579-5845

London WC1
Euro-Calc 01-405-3113

London WC2
TLC World Trading
01-839-3893

Manchester
Cytek 061-832-7604

Executive Reprographic
061-228-1637

Sumlock Electronic
Services 061-834-4233

Matlock
Lowe Electronics
0629-2817

Morley, W. Yorks
Yorkshire Electronic
Services 0532-522181

Norwich
Sumlock Bondain
0603-26259

Nottingham
Betos Systems 0602-48108

Oxford
Orchard Electronics Ltd.
Wallingford (0491)-35529

Plymouth
J.A.D. Integrated Services
0752-62616

Preston
Preston Computer Centre
0772-57684

Reading
CSE Computers
0734-61492

Southampton
Business Electronics
0703-738248

Symtec Systems
0703-37731

Xitan Systems Ltd.
0703-38740

Sunderland Tripont
Associated Systems
0783-73310

Woking
P.P.M.
Brookwood (04867)-80111

Petalect 04862-20727

Yeovil
Computerbits
0935-26522

North Scotland
Thistle Computers
Kirkwall (0856)-3140

Northern Ireland
Medical & Scientific
Lisburn (08462)-77533

PETSOFT 1st AGAIN!

CURSOR The Cassette Magazine for Pet owners

Each issue contains at least five brand new programs. Plus, editorial coverage, new product announcements, reviews and programmers hints. £36 for ten issues mailed direct to you.

76 PROGRAMS

for only **£15**

Specially adapted for the PET from Adam Osborne and Associates best-selling book "Some Common Basic Programs". 76 essential mathematical and Financial programs on one cassette.

PET PROGRAMMERS TOOLKIT ... AVAILABLE OCT.

10 powerful routines that greatly enhance PET BASIC, all on a single plug-in ROM chip. New commands include Re-number, Trace, Help, Find, Unlist, Delete, Append, Auto and Dump.

£55+V.A.T. for 16 and 32k PETS

£75+V.A.T. for 8k PETS with old ROMS

150 Pet Programs in the new Catalogue

including: Word Processor	£25
Assembler	£25
Critical Path	£15
Micro Chess	£14
French Vocab	£10
Music	£10
Debtors Letters	£15
Backgammon	£8
Restaurant Finder	£5
Payroll	£25
Program Development Aid System	£15
Stock Control	£15
Mailing List	£15
PET BASIC Tutorial	£15
Games Pack	£7

PLUS V.A.T.

PETACT

POWERFUL

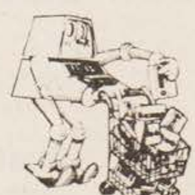
Business Packages ...

PURCHASE ACCOUNTING

Handles Purchase Ledger, prepares a list of outstanding balances and prints remittance advices

SALES ACCOUNTING

All the facilities for maintenance of Sales Ledger, prepares list of outstanding balances and prints statements. Cassette systems from **£225**+V.A.T. Disk versions from **£350** includes training course.



Petsoft programs are available on cassette or disk for all models of the COMMODORE PET
See them at over 200 computer stores, or contact us direct.
CREDIT CARD orders are accepted by telephone.
PET is the trademark of commodore who recommend PETACT Business Systems

Petsoft *Part of the* **ACT** *Computer Group*
Radclyffe House, 66-68 Hagley Road, Edgbaston, Birmingham B16 8PF
Telephone 021-455 8585 *Telex* **339396**

My name is ....

I live at ....

.... Postcode ...

Please rush me a copy of your new catalogue of software for the Pet

POWER SUPPLY INTERLOCK

Mr. P.L. Newland.

A power supply interlock to prevent heartache!

With the reduction in price of dynamic memories, their use is becoming attractive for home computer use (see, for example, Computing Today, April — 16K expansion). The low cost dynamic RAMs require three power supplies; —5 volts and +12 volts. It seems to be a little known fact just how critical these supplies are to each other. To quote the Texas Instruments MOS Memory Data Book:—

> "VBB (—5 volt) must be applied to the device before or at the same time as the other supplies and removed last. Failure to observe this precaution will cause dissipation in excess of the absolute maximum ratings due to internal forward bias conditions. This also applies to system use where failure of the VBB supply must immediately shut down the other supplies."

Most home computers use integrated circuit regulators of the LM309K or 7800 series. These have internal foldback current limiting and this may be used to advantage to add the very simple interlock circuit of Fig. 1.

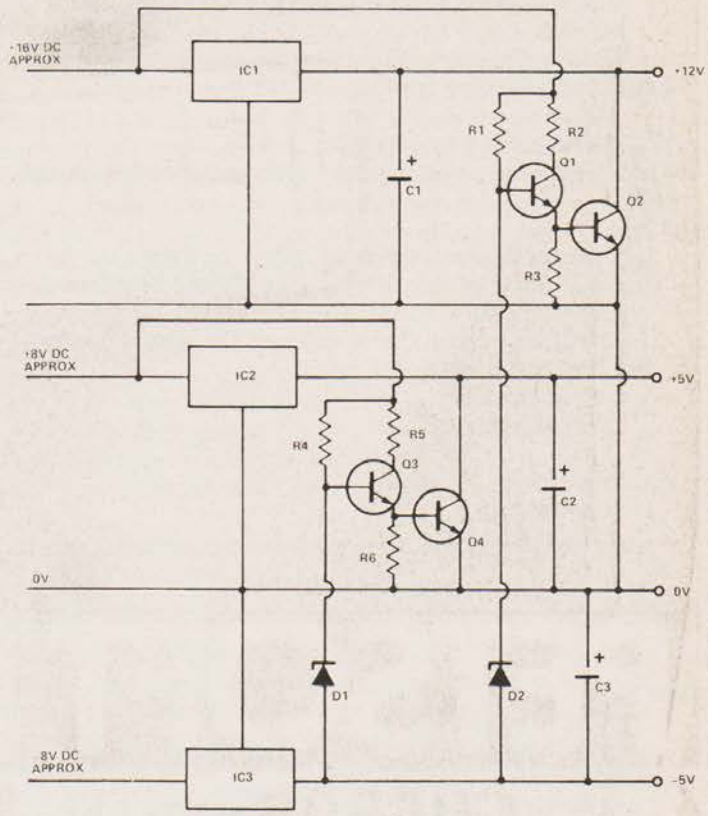


Fig.1 The circuit diagram for the power supply interlock.

HOW IT WORKS

If the negative supply is more than 3V7 negative, the currents in resistors R1 and R4 are diverted through zener diodes D1 and D2. Transistors Q1 — Q4 are therefore normally cut-off. If the negative supply drops below 3.7 volts, the transistors become forward biassed and the currents from resistors R1 and R4 are amplified by Q1 and Q3 which turns Q2 and Q4 hard on. The positive regulators IC1 and IC2 are therefore clamped into their foldback mode and the dynamic RAMs are protected. On switch-on, the negative supply must reach 3V7 before the clamp transistors are released. Also, any fault in the negative supply, or a short across it, will immediately clamp both positive supplies close to zero volts.

PARTS LIST

Resistors (All ¼ W unless specified)

R1	4k7	
R2	270R	(1 Watt)
R3,6	2k2	
R4	1K	
R5	47R	(2 Watt)

Capacitors

C1,2,3	4700 uF (electrolytic)

Semiconductors

IC1	7812
IC2	7805
IC3	7905
Q1,3	BFY50
Q2,4	TIP3055
D1, D2	400mW Zener, 5V1

Miscellaneous

Heatsinks for Q2, Q4.

Dear Sir,

TRS-80 MINI LEDGER PROGRAM : AUGUST ISSUE.

With reference to the above the program, as printed, contains the following errors:—

(a) Line 420 Insert a comma after '128,68'
(b) Line 740 Semi-colon missing between end of quotes and Z1.
(c) Line 770
 (i) Changes the minus signs in quotes to signs.
 (ii) Semi-colon missing after quotes, should be inserted between quotes and P÷Z3–Z2

I will be grateful if you will pass the attached letter to your accounts branch.

Yours sincerely,
W.H.Davies.

98, Henley Road,
Cheltenham,
Glos. GL51 0LD

Dear Sir,

You've goofed again!

The 'Mini Ledger' article in your August issue is chock full of misleading statements.

The 'Hexadecimal' listing is no such thing. It is a 'hardware' print of the TRS80 BASIC source program. It is not even a listing of the program printed in the article (check the line numbers if you don't believe me).

So anyone without BASIC had better forget this program.

You've wasted three pages of your magazine printing a useless listing!

Maybe someone should check articles before you publish them!

Yours
R.J.Hughes.

11, Inman Road,
Earlsfield,
London,
SW18 3BB.

Dear Sir,

As an avid reader of your excellent magazine, may I say how much I enjoy the wide variety of articles you present, particularly the "Softspot" which has provided some exciting and well-documented programs.

I was surpprised therefore, to see in the "MINI LEDGER" article, an irrelevant and superfluous hexadecimal listing of the previously listed BASIC program, wasting both time and valuable space.

Is it not obvious that any Z80-based micro capable of running the hexa. program as listed, will be equipped with a BASIC interpreter and, therefore, the means to input a program in the form of BASIC statements, thus making such a listing redundant.

An hexadecimal program listing is only of value where:—

a) a BASIC interpreter is available for the program run, but no monitor is provided to accept the input of BASIC program statements . . . (unheard of in my experience).
b) the program is at machine-code-level and can be run directly, (ie. not under control of a BASIC interpreter). Such a program is invariably quicker, shorter and more efficient than an equivalent BASIC program.

Yours sincerely,
J.R.Lawton

2, Gwenys Crescent,
Stoke-On-Trent,
ST3 2DB

Dear Sirs,

In most technical publications, to which I subscribe, space is at an absolute premium.

However, this does not appear to be the case with CT. Apparently Mr. WH Davies (Aug '79) is under a misconception regarding the difference between a compiler, generating machine code, and an interpreter, producing an encoded source, intended for subsequent execution by the same.

It seems this impression extends to the editorial staff, who have approved the inclusion of three pages of redundant hexadecimal code, following the program listing.

This space could have been better employed in a number of different ways, such as explaining the differences between compilers and interpreters (assuming you are aware of them).

I do not wish to criticise Mr Davies, who was acting in good faith when he submitted his article. The blame must be laid squarely on the shoulders of the editorial staff for failing in their job.

I hope you will find the space to print this letter.

Yours Temporarily Annoyed,
JD Wheatley,
Programmer LBI.

Dear Sir,

I must comment on Mr.T.M.Spence's letter in the July issue, with regard to Mr. Ingleson's letter about his Nascom.

Mr. Spence's fault was of his own making (easily done considering the no. of joints to be made). Mr. Ingleson's faults were not, nor were mine. For those who may be having problems, perhaps my experiences may be of help.

1. To check for etching faults on Nascom, the best place to look is where the print runs closest to and alongside the I.C. pins. Particularly on the component side of the board. The print may link to a pin it should not go to, and once the I.C. bases are fitted you will not see it. If you have already assembled your kit, play safe and remove all I.C's before making resistance tests and put them safely away.
Note. The faulty print looks normal, use your print diagram. In my case pins 5 and 7 were linked on I.C.55 and pins 2 and 5 were linked on I.C.5.
2. Some kits may need a mod to I.C.18 if the space bar deletes the last character. Link pins 5 and 12 and disconnect pin 5 from the I.C. base. (Bend I.C. leg).
3. If problems are with cass interface, even after Nascoms recommendations are carried out, check transistors for low or no gain. In my case TR5. I have used a BC148 in its place.
4. If memory locations are corrupted after executing and then resetting, try a different series of memory chips. Borrow from a friend if you can. Swopping VDU and USER RAM may not be a valid test as they are connected differently as you will see on the diagram. My Nascom will not tolerate 21L02B chips in USER RAM. RAM for VDU does not seem critical.

As anyone can see these faults were not caused by the constructor.

The working kit is very good and I am pleased with it, so too I believe is Mr. Ingleson. More attention to quality control and it will be better still.

Please keep up the Nascom Packages, and how about an article on Pitfalls and practice using Z80 Machine Code. Keep up the good work.

Yours sincerely,
D.J.Child.

6, Bank,
Eccleshill,
Bradford
BD10 8BH.

Dear Sir,

Has there been a printing error in the list of connections in Table 2 of MPU's by experiment as C20 shows a second 560R? The capacitor C* would then be in the C21 connections.

Yours sincerely,
W.H.Edwards.

91, Grove Road,
Millhouses,
Sheffield S72GY
Yorks.

Dear Sir,

I am writing an elementary book on "MICRO-PROCESSORS", to be published by the Babani Press.

I would like to include a list of "Amateur Computer Societies" at the end of the book. Could you print a small request in 'Computing Today' for addresses of Computer Societies in Britain, Canada, Australia, N.Z. I want to hear from societies who welcome 'raw beginners' as members

I would also like to know if any university or college in Britain offers or is planning to introduce a degree course in "MICROPROCESSOR STUDIES" or "MICRO-ELECTRONICS TECHNOLOGY".

Thanks
R.N.Soar.

c/o Babani Press,
The Grampians,
Shepherds Bush Road,
London W6 7NF.

Dear Sir,

In the August (august?) issue of C.T., E.B.Simmons brought up a few points on computer vocabulary, most of which were quite valid. However, language is developed by usage, and within the computer industry (where I have languished for seven years!), the word describing one OR more pieces of information has always been DATA, not DATUM.

In fact, neither are strictly correct, since a datum is a measurement reference point, whereas an item of computer data is an attribute of the problem being solved. Rather a subtle difference, I'm afraid.

Incidentally, I wonder if he/she has ever tried to analyse post codes on a computer? It's not as simple as it might seem.

Yours,
Pete Newman
Programming Consultant.

Atkins On-Line Limited,
Fourmost House 12-22 West Street,
Epsom, Surrey KT18 7RH

Dear Sir,

Since I am myself the owner of a disk-based TRS-80, I was intrigued by the article 'Mini Ledger' by Mr.W.H.Davies in your August issue. Lest any of your readers be misled, could I point out that the TRS-80 BASIC is an interpreter, not a compiler. The hexadecimal listings which you print are in fact simply the encoded representation of the BASIC source code. They are not Z80 machine code, and are therefore of no value to Z80 users without the TRS-80 BASIC.

Yours faithfully,
T.J. Bourne.

16, Crossways,
Laverstock Green,
Hemel Hempstead,
Herts HP3 8PU.

computing today

technical book service

What Is A Microprocessor ?
2 Cassette tapes plus a 72 page book deal with many aspects of microprocessors including Binary and Hexadecimal counting, Pro–gramming etc. £10.75.

NEW:
Adams, C. BEGINNERS GUIDE TO COMPUTERS AND MICROPROCESSORS WITH PROJECTS £5.60
Understanding building programming and operating your own microcomputer.

NEW:
Albrecht, B. BASIC FOR HOME COMPUTERS. A self teaching guide £4.75
Shows you how to read, write and understand basic programming language used in the new personal size microcomputers.

Albrecht B. BASIC. A self teaching guide (2nd edition) £4.50
Teach yourself the programming language BASIC. You will learn how to use the computer as a tool in home or office and you will need no special maths or science background.

Alcock, D. ILLUSTRATING BASIC £2.60
This book presents a popular and widely available language called BASIC, and explains how to write simple programs.

Altman, I. MICROPROCESSORS £10.65
Gives a general overview of the technology design ideas and explains practical applications.

Altman, L. APPLYING MICROPROCESSORS £12.00
Follow volume which takes you into the second and third generation devices.

Aspinall, D. INTRO TO MICROPROCESSORS £6.40
Explains the characteristics of the component.

NEW:
Barden, W. Z-80 MICROCOMPUTER HANDBOOK £7.65

Barden, W. HOW TO BUY AND USE MINICOMPUTERS AND MICROCOMPUTERS £7.75
Discusses these smaller computers and shows how they can be used in a variety of practical and recreational tasks in the home or business.

Barden, W. HOW TO PROGRAM MICROCOMPUTERS £7.00
This book explains assembly language programming of microcomputers based on the Intel 8080, Motorola MC6800 and MOS Technology MCS6502 microprocessor.

Barna, A. INTRODUCTION TO MICROCOMPUTERS AND MICROPROCESSORS £8.15
Provides the basic knowledge required to understand microprocessor systems. Presents a fundamental discussion of many topics in both hardware and software.

Bibbero, R. J. MICROPROCESSORS IN INSTRUMENTS AND CONTROL £12.45
Introduces the background elements, paying particular regard to the dynamics and computational instrumentation required to accomplish real-time data processing tasks.

Lancaster, D. TV TYPEWRITER COOKBOOK £7.75
An in-depth coverage of tv typewriters (tv's) the only truly low cost microcomputer and small display interface.

Lancaster, D. CHEAP VIDEO COOKBOOK £6.50

Lesea, A. MICROPROCESSOR INTERFACING TECHNIQUES £8.50

NEW:
Leventhal. INTRO TO MICROPROCESSORS £16.70

NEW:
Lewis, T. G. MIND APPLIANCE HOME COMPUTER APPLICATIONS £4.75

NEW:
Libes, S. SMALL COMPUTER SYSTEMS HANDBOOK £5.75
The Primer written for those new to the field of personal home computers.

NEW:
Lippiatt. ARCHITECTURE OF SMALL COMPUTER SYSTEMS £4.35

NEW:
Moody, R. FIRST BOOK OF MICROCOMPUTERS £3.85
(the home computer owners best friend).

McGlynn, D. R. MICROPROCESSORS — Technology, Architecture & Applications £9.00
This introduction to the 'computer-on-a-chip' provides a clear explanation of the important new device.

McMurran, PROGRAMMING MICROPROCESSORS £5.50
A practical programming guide that includes architecture, arithmetic/logic operations, fixed and floating point computations, data exchange with peripheral devices computers and other programming aids.

NEW:
Nagin, P. BASIC WITH STYLE £4.00
Programming Proverbs. Principles of good programming with numerous examples to improve programming style and producing.

NEW:
Ogdin SOFTWARE DESIGN FOR MICROCOMPUTERS £7.00

NEW:
Ogdin. MICROCOMPUTER DESIGN £7.05

Peatman, J. B. MICROCOMPUTER BASED DESIGN £20.40

This book is intended for undergraduate courses on microprocessors.

NEW:

Bursky, D. MICROCOMPUTER BOARD DATA MANUAL £5.40

Bursky, D. MICROPROCESSOR DATA £5.40

Includes complete description of the processor. Support circuits, Architecture, Software, etc.

Coan, J. S. BASIC BASIC £7.50

An introduction to computer programming in BASIC language.

Coan, J. S. ADVANCED BASIC £7.30

Applications and problems.

NEW:

Duncan. MICROPROCESSOR SOFTWARE ENGINEERING £13.50

NEW:

Freiberger, S. CONSUMERS GUIDE TO PERSONAL COMPUTING AND MICROCOMPUTERS £5.50

NEW:

Frenzel, L. GETTING ACQUAINTED WITH MICROPROCESSORS £7.10

This is an invaluable book for those who want to know more about hobby and personal computing.

Gilmore, C. M. BEGINNERS GUIDE TO MICROPROCESSORS £4.75

Gosling, R. E. BEGINNING BASIC £3.25

Introduces BASIC to first time users.

Graham, N. MICROPROCESSOR PROGRAMMING FOR COMPUTER HOBBYISTS £7.00

Haviland, N. P. THE COMPULATOR BOOK £6.20

Building super calculators and minicomputer hardware with calculator chips.

Heiserman, D. L. MINIPROCESSORS FROM CALCULATORS TO COMPUTERS £4.85

Hilburn, J. L. MICROCOMPUTERS, MICROPROCESSORS, HARDWARE, SOFTWARE AND APPLICATIONS £16.95

Complete and practical introduction to the design, programming operation, uses and maintenance of modern microprocessors, their integrated circuits and other components.

Klingman, E. MICROPROCESSOR SYSTEMS DESIGN £16.95

Outstanding for its information on real microprocessors, this text is both an introduction and a detailed information source treating over a dozen processors, including new third generation devices. No prior knowledge of microprocessors or microelectronics is required for the reader.

Kemeny, J. G. BASIC PROGRAMMING £6.55

A basic text.

Korn, G. A. MICROPROCESSOR AND SMALL DIGITAL COMPUTER SYSTEMS FOR ENGINEERS AND SCIENTISTS £19.00

This book covers the types, languages, design software and applications of microprocessors.

Rao, G. U. MICROPROCESSOR AND MICROPROCESSOR SYSTEMS £20.50

A completely up-to-date report on the state-of-the-art of microprocessors and microcomputers written by one of the leading experts.

Rony, P. H. THE 8080A BUGBOOK: Microcomputer Interfacing & Programming £8.15

The principles, concepts and applications of an 8-bit microcomputer based on the 8080 microprocessor IU chip. The emphasis is on a computer as a controller.

Scelbi. 6800 SOFTWARE GOURMET GUIDE AND COOKBOOK £8.80

Scelbi. 8080 SOFTWARE GOURMET GUIDE AND COOKBOOK £8,80

Scelbi. UNDERSTANDING MICROCOMPUTERS £8.60

Gives the fundamental concepts of virtually all microcomputers.

NEW:

Schoman, K. THE BASIC WORKBOOK £3.70

Creative techniques for beginning programmers.

NEW:

Sirion, D. BASIC FROM THE GROUND UP £6.00

Soucek, B. MICROPROCESSORS AND MICROCOMPUTERS £19.00

Here is a description of the applications programming and interfacing techniques common to all microprocessors.

NEW:

Spracklen, D. SARGON £9.75

A computer chess program in Z-80 assembly language.

NEW:

Tracton. 57 PRACTICAL PROGRAMS & GAMES IN BASIC £6.40

Programs for everything from Space war games tc Blackjack.

Waite. M. MICROCOMPUTER PRIMER £6.25

Introduces the beginner to the basic principles of the microcomputer.

Ward. MICROPROCESSOR / MICROPROGRAMMING HANDBOOK £6.00

Authoritative practical guide to microprocessor construction programming and applications.

NEW:

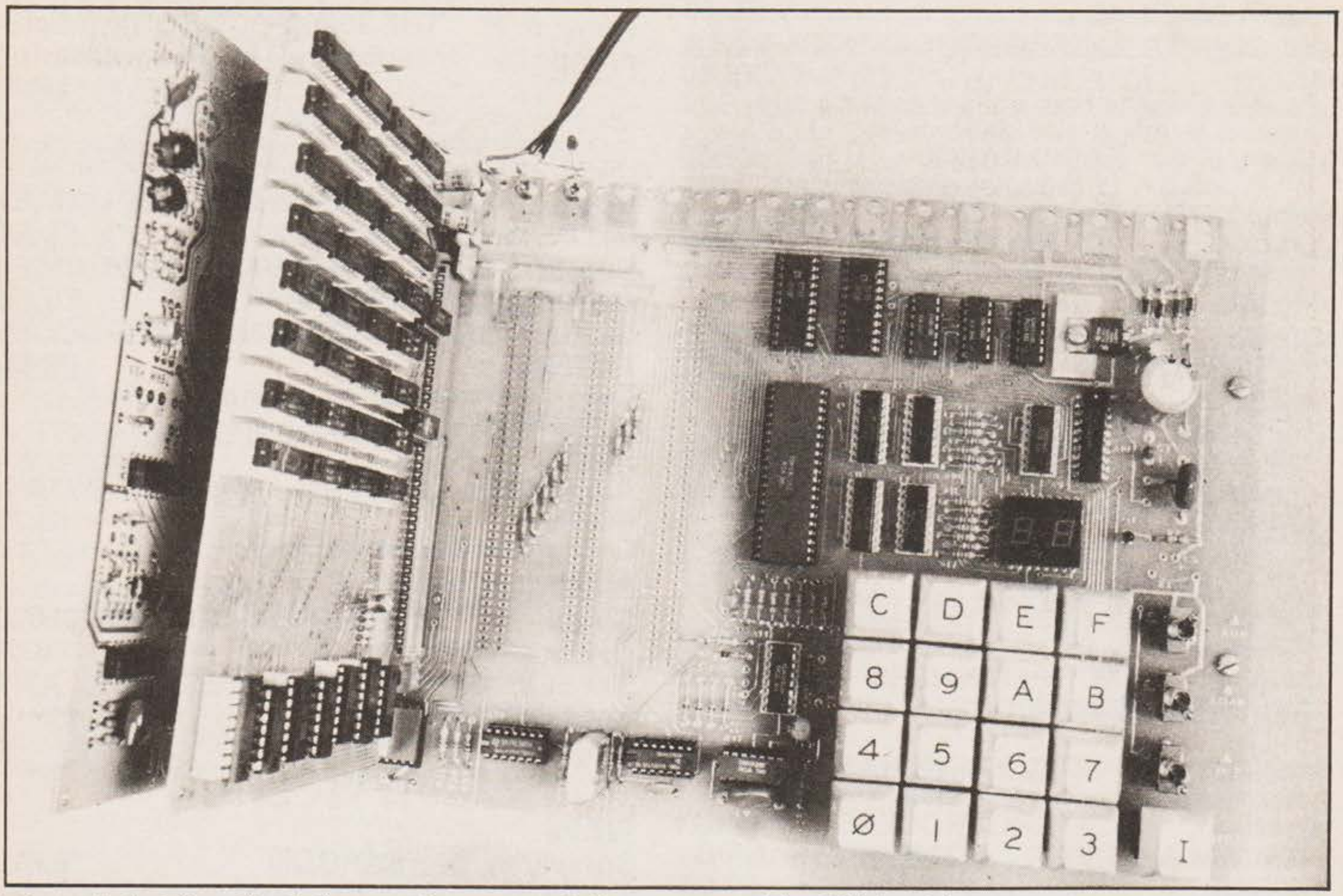
Veronis. MICROPROCESSOR £12.85

Zaks, R. INTRODUCTION TO PERSONAL AND BUSINESS COMPUTING £8.50

Zaks, R. MICROPROCESSORS FROM CHIPS TO SYSTEMS £8.50

Note that all prices include postage and packing. Please make cheques, etc, payable to Computing Today Book Service (Payment in U.K. currency only please) and send to:
Computing Today Book Service,
P.O. Box 79,
Maidenhead, Berks.

D.Sinfield.



David Sinfield reviews this Tiny Basic gnome computer

The minimal Netronics ELF II system consists of an RCA 1802 CPU, 256 bytes of RAM, an RCA 1852 VDU controller, a 2 digit seven seg display and a hex keyboard as well as all the vital bits that hang all these together. It is available as a kit but for anyone lacking the requisite confidence in their constructional abilities ready-built ELFs (or ELVES?) can also be supplied. I didn't build the machine reviewed but can see no practical problems in locating and soldering the components at their marked positions on the double sided, through plated PCB if the step-by-step instructions are followed systematically and the normal precautions taken to avoid blowing the CMOS chips by discharging static through them.

Having sorted out the hardware and laid on the 8V unregulated supply it's time to consider programming. At this stage machine code is all there is and a little work is required to get the hang of this at the first encounter. On the minimal ELF II there is no monitor (that comes a little later) and the three switches to the right of the hex keypad are used in various combinations to enter, examine and run programs and zero the stack pointer. Tom Pittman's short course in programming, specially written for RCA1802 based systems and, I suspect, for the ELF II in particular initiates the user in the meaning and use of all the instructions in the 1802's vocabulary with example programs at each stage. Later programs use a VHF modulator and TV set.

The CPU board has no monitor and as programs get longer and more sophisticated this becomes more of a disadvantage. Also the only way of entering programs on the CPU board is from the keypad. These two problems can be solved at a stroke using the "Giant" board. This provides I/O and a monitor. (The Giant board is also available built or as a kit with full instructions).

The connection to the CPU board is via an edge connector mounted on the left of the board. Although I have some reservations about this connection as no mechanical support is given to the vertically mounted "Giant" board it gave no trouble.

The monitor resides in ROM at memory location F000 and is entered by using an unconditional long jump to this location. Once the monitor is entered six operating modes are available and are selected using the codes 00 to 05.

00 runs a program from a specified two byte location.

01 allows the contents of any specified location to be displayed on the dual seven segment display. While the input key is held depressed the low order byte of the address is displayed and as the key is released the contents of the location are displayed. By pressing and releasing the input key the memory can be stepped through and a check kept on the address of the memory being displayed.

02 is similar to 01 but the contents of a specified location can be altered as well as inspected. The same method of displaying the address is used.

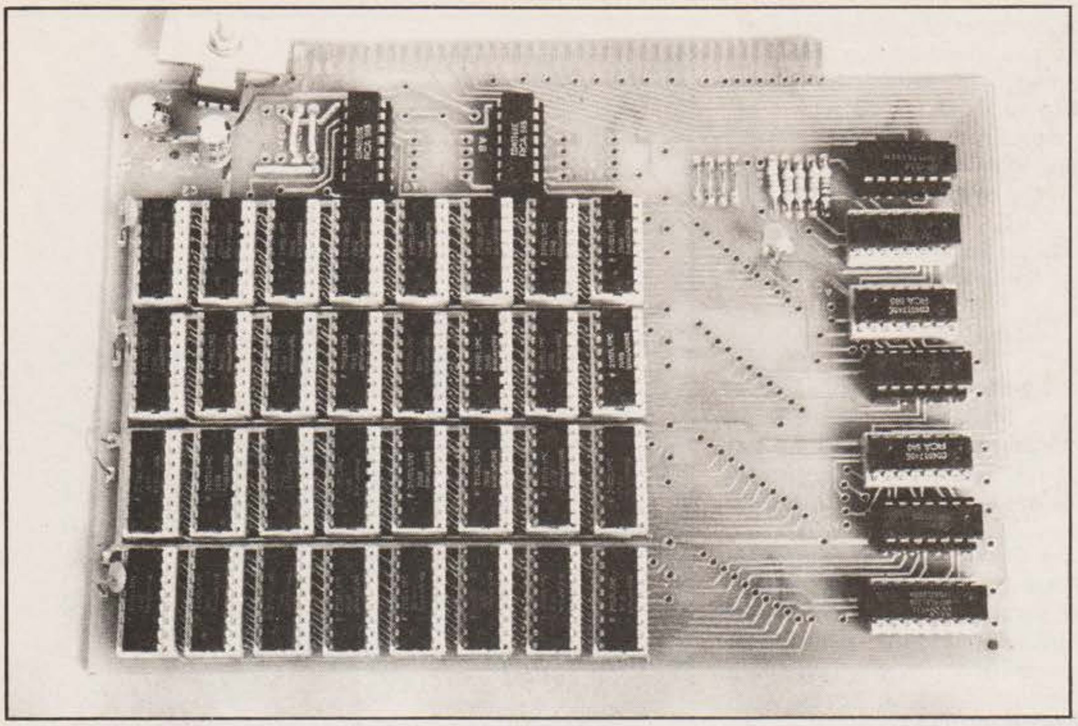


Fig.1. (far left) The Elf showing "Giant" board and 4K RAM expansion in sockets.

Fig.2. (left) 4K RAM board. The voltage regulator at top left gets hot enough to fry eggs on — definitely one to miss with the fingers.

INSTRUCTION SUMMARY

MNEM	NAME	OPCODE	PAGE
ADC —	ADd with Carry	74	42
ADCI b	ADd with Carry Immediate	7C bb	42
ADD —	Add	F4	41
ADI b	ADd Immediate	FC bb	42
AND —	Logical AND	F2	39
ANI b	ANd Immediate	FA bb	40
B1 a	Branch on External Flag 1	34 aa	16
B2 a	Branch on External Flag 2	35 aa	16
B3 a	Branch on External Flag 3	36 aa	16
B4 a	Branch on External Flag 4	37 aa	16
BDF a	Branch if DF is 1	33 aa	41
BN1 a	Branch on Not External Flag 1	3C a7	17
BN2 a	Branch on Not External Flag 2	3D aa	17
BN3 a	Branch on Not External Flag 3	3E aa	17
BN4 a	Branch on Not External Flag 4	3F aa	17
BNF a	Branch if DF is 0	3B aa	42
BNQ a	Branch if Q is off	39 aa	25
BNZ a	Branch on Not Zero	3A aa	23
BQ a	Branch if Q is on	31 aa	25
BR a	Branch unconditionally	30 aa	18
BZ a	Branch on Zero	32 aa	23
DEC r	Decrement register	2r	27
DIS —	Return & DISable interrupts	71	53
GHI r	Get HIgh byte of register	9r	29
GLO r	Get LOw byte of register	8r	29
IDL —	Idle	00	14
INC r	Increment register	1r	27
INP p	Input to memory & D; 9≤p≤F	6p	20
IRX —	Increment R(X)	60	35
LBDF aa	Long Branch if DF is 1	C3 aaaa	42
LBNF aa	Long Branch if DF is 0	CB aaaa	42
LBNQ aa	Long Branch if Q is off	C9 aaaa	24
LBNZ aa	Long Branch if Not Zero	CA aaaa	23
LBQ aa	Long Branch if Q is on	C1 aaaa	24
LBR aa	Long Branch unconditionally	C0 aaaa	18
LBZ aa	Long Branch if Zero	C2 aaaa	23
LDA r	Load D & Advance	4r	34
LDI b	Load D Immediate	F8 bb	31

LDN r	Load D via N 1≤r≤F	0r	35
LDX —	Load D via R(X)	F0	35
LDXA —	Load D via R(X)Advance	72	35
LSDF —	Long Skip if DF is 1	CF	42
LSIE —	Long Skip if Interrupts are Enabled	CC	53
LSKP —	Long Skip	C8	22
LSNF —	Long Skip if DF is 0	C7	42
LSNQ —	Long Skip if Q is off	C5	25
LSNZ —	Long Skip if Not Zero	C6	24
LSQ —	Long Skip if Q is on	CD	25
LSZ —	Long Skip if Zero	CE	24
MARK —	Save X & P	79	55
NOP —	No Operation	C4	22
OR —	Logical OR	F1	38
ORI b	OR Immediate	F9 bb	39
OUT p	Output from memory; 1≤p≤7	6p	20
PHI r	Put D into HIgh byte of register	Br	29
PLO r	Put into LOw byte of register	Ar	29
REQ —	Reset Q	7A	15
RET —	RETurn	70	53
SAV —	SAVe T	78	56
SD —	Subtract D from memory	F5	43
SDB —	Subtract D from memory with Borrow	75	46
SDBI b	Subtract D from Immediate byte with Borrow	7D bb	46
SDI b	Subtract D from Immediate byte	FD bb	44
SEP r	SEt P	Dr	52
SEQ —	Set Q	7B	14
SHL —	SHift D Left	FE	47
SHLC —	SHift D Left with Carry	7E	47
SHR —	SHift D Right	F6	48
SHRC —	SHift D Right with Carry	76	48
SKP —	Skip one byte	38	22
SM —	Subtract Memory byte from D	F7	46
SMB —	Subtract Memory byte from d with Borrow	77	46
SMBI b	Subtract Immediate Memory from D with Borrow	7F bb	46
SMI b	Subtract Immediate Memory from D	FF bb	46
STR r	STore D into memory	5r	32
STXD —	STore D via R(X) & Decrement	73	33
XOR —	EXclusive OR	F3	39
XRI b	EXclusive OR Immediate	FB bb	40

03 writes to cassette the contents of any section of memory between any two specified two byte locations. When the machine is writing the Q led to the right of the display is lit.
04 reads from cassette, storing the program between two specified two byte locations. The rate at which information is read in is about 100 bytes per second.
05 scans the memory looking for a specified byte and when found gives the low order byte of its address.

Most cassette players will be found suitable as long as there is some means of getting a signal into a microphone input and out of an extension speaker output. The ELF I was loaned had a LED indicator and a decoupling arrangement on the tape input port. Software is available on cassette and I'll mention that I tried later.

Another function of the Giant board is to provide various other I/O formats. These are jumper selected and include 8 bit parallel in, 8 bit parallel out, TTY/2mA and R5232 all of which can be implemented without additional buffering. Full details are provided in the assembly instructions.

Before mentioning any more add-ons it will be worthwhile considering the 4K RAM cards up to 4 of which can be accommodated on edge correctors on the main board as is the Giant board. The only problem I found was with the voltage regulator which got very hot but even after about 12 hours of continuous use showed no sign of failing.

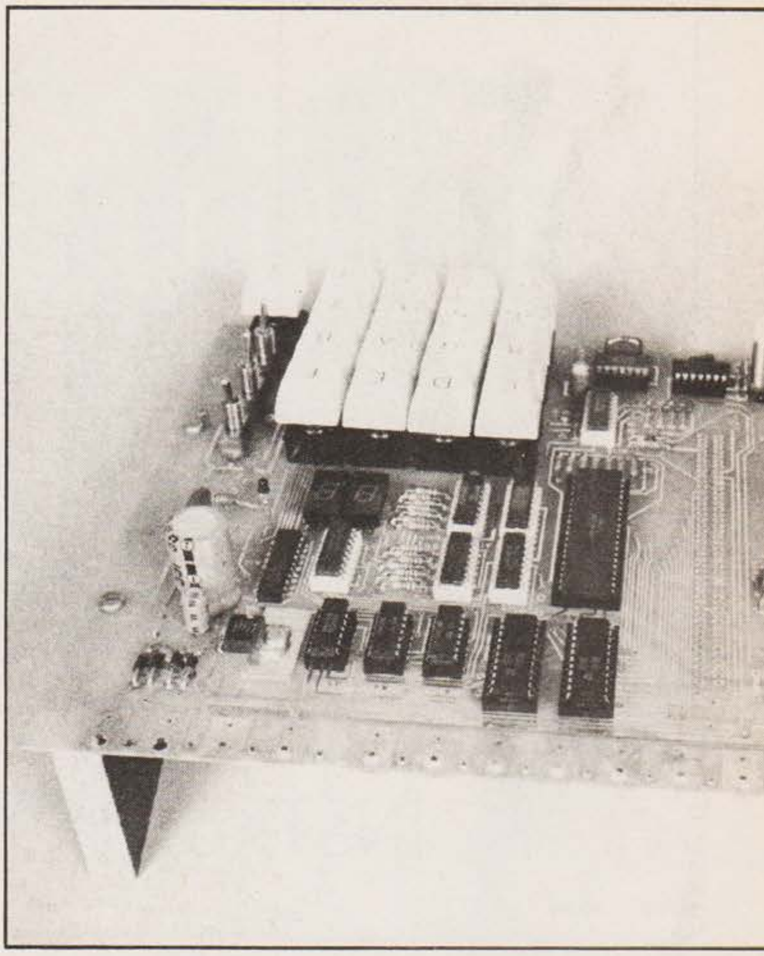
Sooner or later some means of examining more than one location at a time is an advantage and Elf-bug monitor on tape enables 24 locations to be displayed on a TV using a UHF modulator. The commands used by the Elf-bug are parallel to those used by the monitor and although the cassette read/write commands cannot be dealt with under Elf-bug they are passed on to the giant board monitor for execution. Also displayed using Elf-bug are the values in the registers. These appear each time Elf-bug is restarted.

This just about covers using the Elf II for machine code programming but it's capabilities don't end here. Although I didn't have a chance to try it I understand that there is an Assembler available on tape so I'll move on to the Elf II and BASIC.

Tiny BASIC on tape is available for the ELF and to use it an ASCII keyboard is required. Newtronics supply one for the Elf which has an excellent feel. Alternatively the new Newtronics terminal can be used and as this is what I had I'll restrict the description to this.

The VDU board used has all the usual alphanumerics but the graphics are sacrificed in favour of Greek letters. The output to a monitor can be set to 64 x 16 and to a TV 32 x 16. The BASIC is a little limited but I understand an extended version is being worked on. The only real complaint I had was that I found it impossible to POKE characters to the screen — a useful facility when writing games programs particularly.

The Elf II then is not the machine for anyone who wants to take it home, plug it in and run BASIC programs — there are much cheaper machines with better BASIC. The real appeal of the Elf is that building it up in easy stages gives a good insight into how both the hardware and software works, and so far control applications, development work and education and training purposes the Elf II is well worth considering.



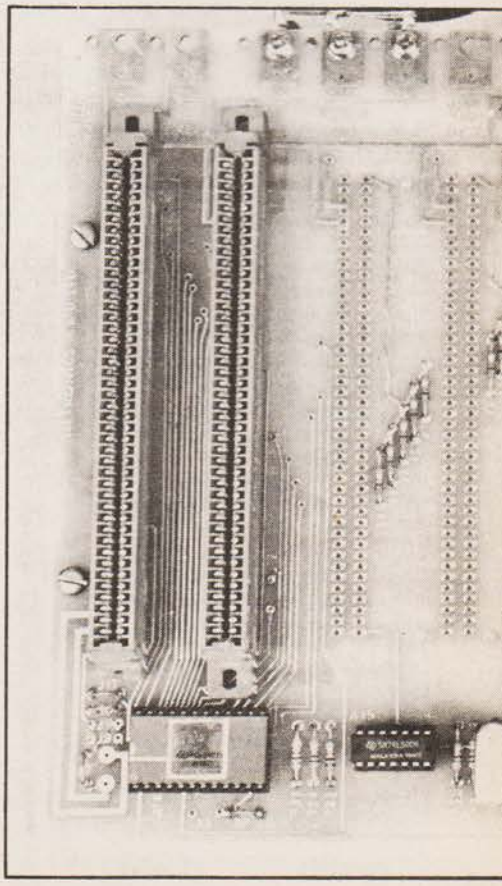


Fig.4 (right). The minimal system — a processor board, hex keypad and 2 digit 7 segment display. Only two of the five possible edge connectors are fitted here.

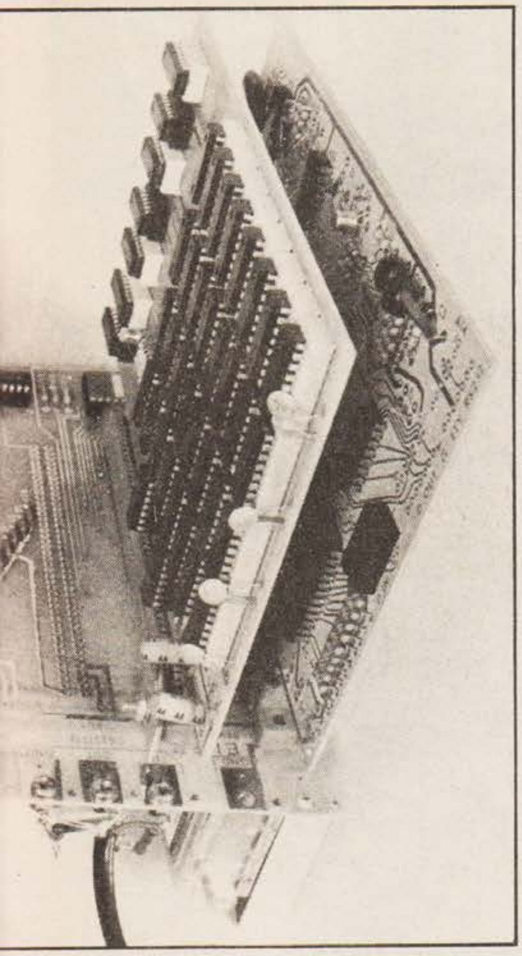
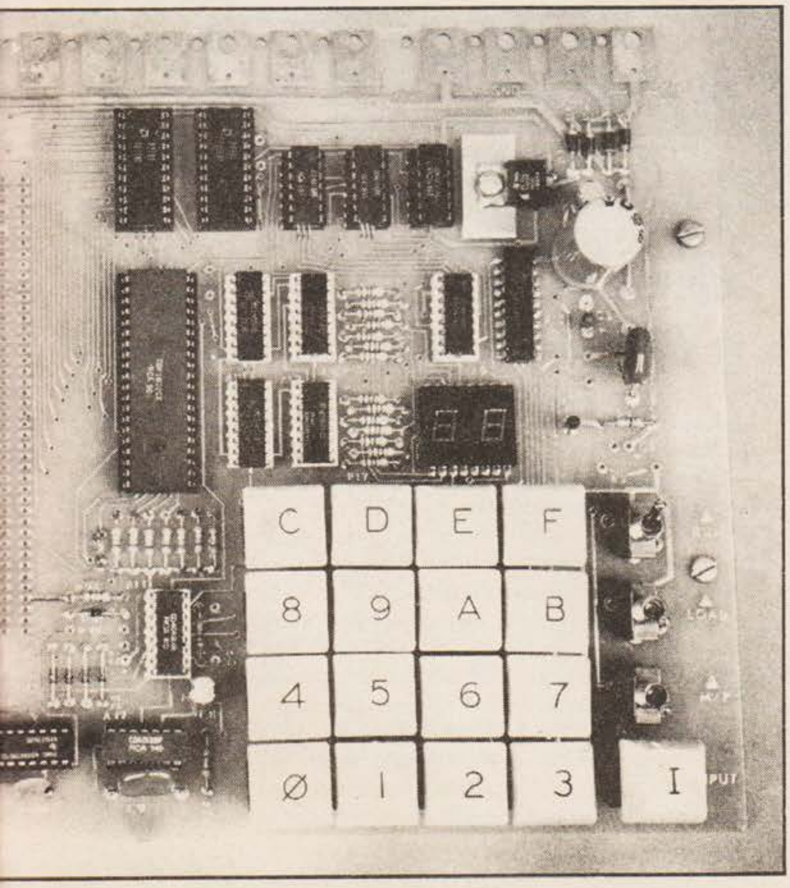
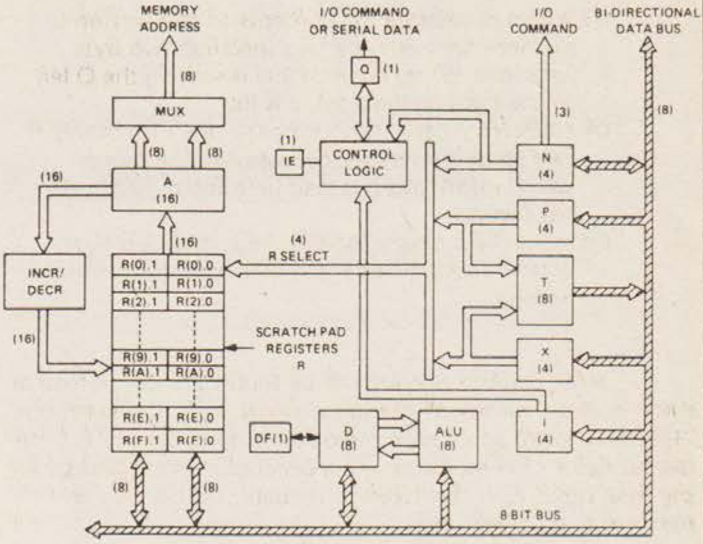


Fig.3 (left) The back view. Note cassette decoupling circuit and cassette input LED.






```
..  PROGRAM 2.1  --  MEMORY CLEAR
..
0000 90  CLEAR GHI 0  .. REGISTER 0 HAS 0001
0001 AE        PLO 14 .. MAKE RE=0000
0002 BE        PHI 14
0003 EE  LOOP  SEX 14 .. EACH TIME, R14 IS -1
0004 73        STXD   .. D STILL HAS 00
0005 30        BR  LOOP .. GO BACK FOR ANOTHER
0006 03
```

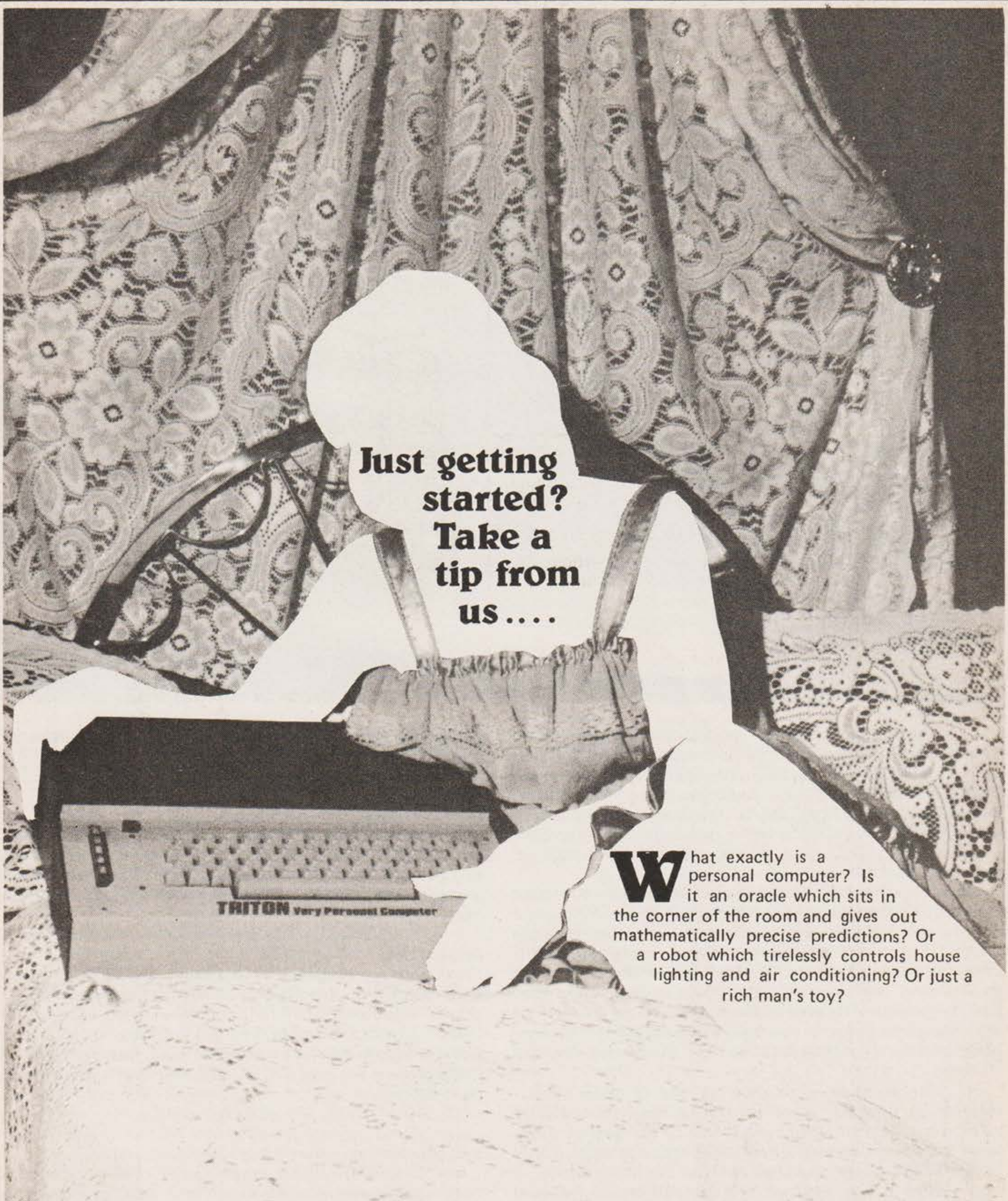
```
..  PROGRAM 2.2  --  MEMORY SEQUENCER
..
0000 90  SEQ   GHI 0  .. THIS PART IS
0001 AE        PLO 14 ..   JUST LIKE CLEAR
0002 BE        PHI 14
0003 EE  LOOP  SEX 14
0004 8E        GLO 14 .. THIS IS ADDRESS VALUE
0005 73        STXD          .. SO DATA=ADDRESS
0006 30        BR  LOOP .. REPEAT UNTIL DONE
0007 03
```

```
..  PROGRAM 2.3  --  SLOW BLINK
..
0000 91  BLINK GHI 1  .. LOOK AT TIMER IN 1
0001 CE        LSZ    .. IS ZERO ONLY 1/256
0002 7A        REQ    .. IF NOT 00, Q OFF
0003 38        SKP
0004 7B        SEQ    .. WHILE ZERO, Q ON
0005 11        INC 1  .. BUMP COUNTER
0006 30        BR  BLINK ..   THEN REPEAT.
0007 00
```

Above some sample programs and a schematic from the documentation provided.

Thanks to NEWTRONICS of 138 Kingsland Road, London N1, the distributor of the Elf II in this country, for the loan of the machine.

BUYERS GUIDE



Just getting started? Take a tip from us....

What exactly is a personal computer? Is it an oracle which sits in the corner of the room and gives out mathematically precise predictions? Or a robot which tirelessly controls house lighting and air conditioning? Or just a rich man's toy?

The Compucolor home computer with its integral disk and separate keyboard. Although brilliantly documented the hardware design is slightly suspect. The colour graphics are very easy to use.

The hobby computer market has split into two clearly-defined areas: The first is exemplified by products such as the Commodore PET, the Exidy Sorcerer and the Tandy TRS-80. This is the "easy-to-use, plug-in-and-go" high end of the market. Applications. Teaching the buyer how to use a computer, working as a calculator, figuring bank balances and astrological charts, playing 'sophisticated' computer games.

The other half of the market is the get-you-started systems (usually sold as kits) which don't do anything useful as they stand but are very much more instructive. Products which fall into this area are too diverse to mention, as the capital needed to begin production of these devices is much smaller than that required to produce a larger unit. They can't immediately be used as calculators and to do anything with them except learn, you have to have a fairly high level of knowledge – which you can get by buying one and studying.

The eventual application of one of these smaller systems is, however, more interesting than the complex calculator type mentioned earlier. Due to the level of knowledge the smaller systems offer, the possibilities depend only on the ability of the user. You can turn one into a calculator type system if you want to. You can also use it to control air conditioning or vacuum the floor or even tell you jokes.

The high price systems can, of course, be used for 'control applications' (house lighting, etc.) but they are more difficult to modify for this purpose and the user would presumably be happier modifying something he paid £200 for and built himself than something for which he paid £1000 and has never seen the inside of.

Up-market Machines

What do we mean by an 'up-market' personal computer? Essentially, it's one that can work out 2 + 2 = 4 as soon as you plug it in. It's a super-calculator. It's (usually) programmed in BASIC (Beginner's All-purpose Symbolic Instruction Code).

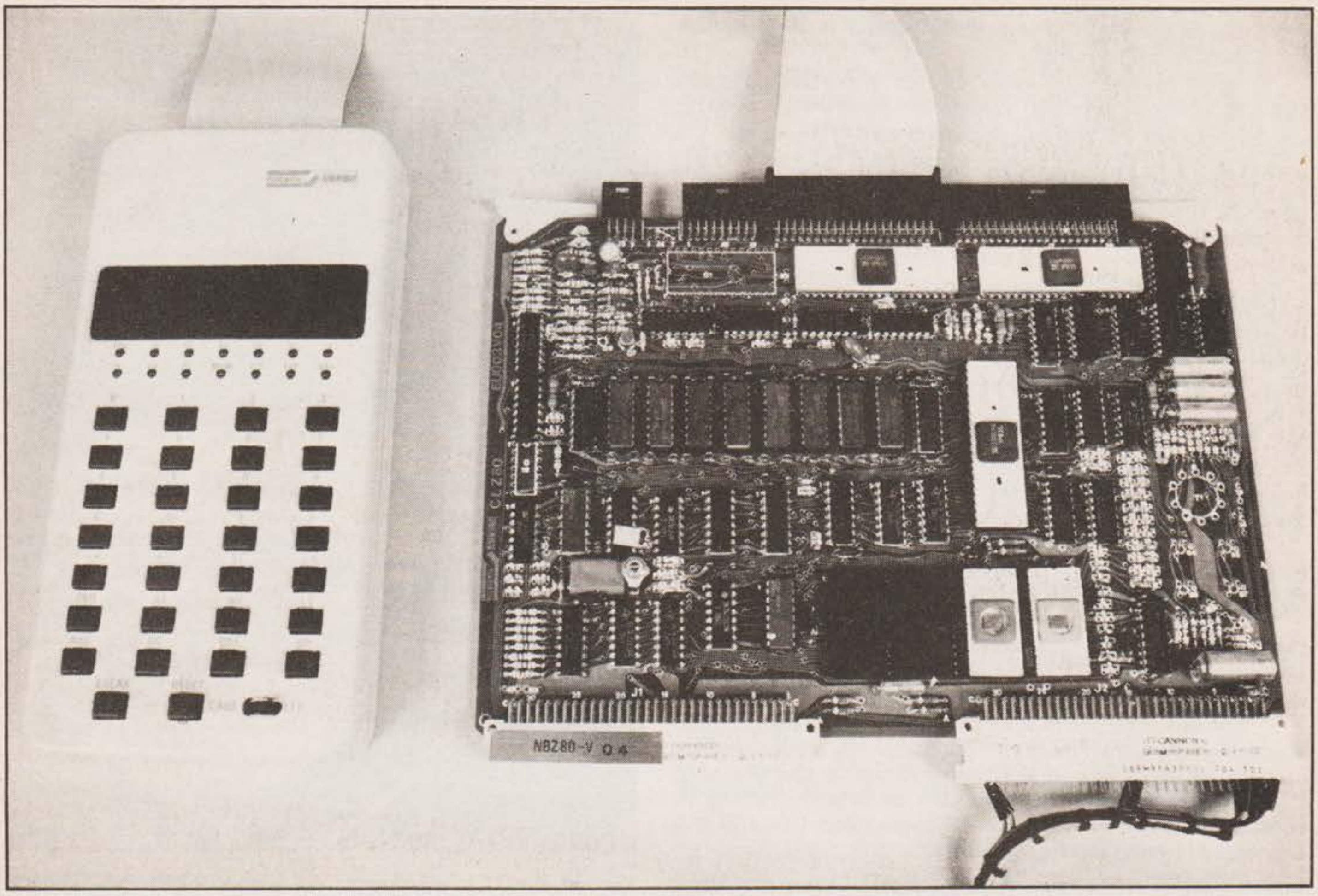
Fine. What do you do once you've got it out of the packing case?

First, you plug it in to a wall socket. This is usually all the wiring up you have to do. Let's take the example of the Commodore PET. The PET has a Video Display Unit (VDU) (which looks just like a TV screen, except that it displays words instead of pictures) built into the top of it. You switch it on, type in "? 2 + 2", and the PET will display on its screen:

```
? 2+2
4
```

showing that the answer is 4.

What if you want to program it? There is a comprehensive guide with the machine which tells you how to go about this. A program is a sequence of instructions which you want the machine to carry out. Once you have entered them, the computer will do them in sequence on request – just like a programmable calculator. If, however, you turn the machine off, it will forget the program that is is holding at present. This is pretty useless if it's a program which you want to use again and again, and so the PET has the facility



A basic Nanocomputer system. Superb design allows the educational system to be expanded to a full Z80 based computer with all the standard facilities.

of recording any given program onto a cassette using an inbuilt cassette recorder. You simply insert a perfectly normal audio cassette and type in the word "SAVE". The PET will record the program onto the tape. You take the tape out, turn the machine off and the next time you want to use that particular program, you put the tape back in and type "LOAD". The PET will play the tape and remind itself of the program which you had it store earlier.

Now let's look at the Exidy Sorcerer. Although the Sorcerer is a more powerful machine (i.e. it can do more things in a shorter amount of time than the PET can) it doesn't have a built-in VDU and cassette deck. Is this a problem? Not at all. You simply connect the aerial socket of your TV set to the back of the Sorcerer and behold! it becomes the VDU. Similarly, when you want to save a program, you connect it to your cassette deck (the one you've just recorded "ABBA" on) and away you go. The Sorcerer is also programmed in BASIC (although it has the facility to change languages when Exidy, the manufacturers, get around to producing the firmware) and also comes complete with a full instruction manual on how to go about writing a program.

A machine very similar to the Sorcerer is the Sord M-100. The major difference is that it has an S-100 output built in (see later) and two analogue inputs for connecting it to . . . well, anything you can think of!

The Apple offers much the same facilities as the Sorcerer, except that if you happen to have a colour TV set to plug into the back, it will allow you to have letters and words in different colours – for example, if you have written a program to tell you when your biorhythms are at their worst, you can arrange to have the words "take care!" flashed on the screen in bright red whenever you are about to go through a bad patch.

The Tandy TRS-80 is also much the same in its basic structure as the Sorcerer, except that Tandy supply a specially-designed cassette recorder and cut-down TV set to go with it as part of the package.

What's The Difference

Right about now, you should be asking yourself the question, "What's the difference between all these systems, apart from the prices?". In order to answer that one, we'll split the facilities offered into several areas:

1) VDU
2) Tape recording
3) Keyboard
4) BASIC
5) Inputs and Outputs

First, the VDU. The only special feature in a VDU is whether it's colour or black and white.

Tape recording is very simply dealt with as well – they either have it built in, supplied or external (ie: you use your own). In terms of reliability of recording, there's not much difference, although in the past PET has come in for a good deal of criticism Commodore now seem to have solved most of the problems. The main reason is that audio and data recording are not the same, data is much more critical, you can't lose bits without upsetting the computer – the human ear is much more tolerant.

The keyboard is really only an important consideration if you can type. If you can't, there's very little to choose between the makes mentioned above. If you can, the PET is going to be a bit slower to use than the rest as it has very small keys. However Commodore are now producing the larger PETs with a proper typewriter keyboard, once again customer feedback has produced a design change in the original machine.

BASIC, the language in which all of the above machines are programmed comes in many different shapes and sizes. Some are easy to use and very powerful (ie: fast) and some are not so easy, others not so fast. The particular brand of BASIC used in the PET is written by a Californian firm called Microsoft and is very easy to use and reasonably fast. Some machines such as the Apple and Sorcerer are in the process of being equipped with other languages such as Pascal but it should be noted that all these machines have the capability of being programmed in machine code.

The Sorcerer's BASIC is essentially the same as the PET's but with a few little add-ons. The Apple and TRS-80 BASIC are also about the same as the PET's.

What it all boils down to is that Apple users will swear blind that the Apple version is streets ahead of the rest and similarly any PET user will go blue shouting that the PET's version is the best and so on . . .

If at some time in the future, after you have completely mastered your up-market machine (which takes about six months on average) you may want to add a few things to it, such as a printer, a floppy disc or even something to control your air conditioning. All these need I/O. I/O is a lazy way of saying Inputs and Outputs.

The PET has a particularly obscure I/O scheme, using what is called the IEEE 488 bus. No other personal computer on the market (that we know of) uses this scheme, some desktop computers, notably Hewlett Packard, use the system however so you can use the PET for instrumentation work. Luckily the PET has become so popular in the States and Europe that it's worthwhile for people other than Commodore to produce add-on bits which will convert the PET to S-100.

S-100 is an almost universal I/O scheme. Printer, floppy discs and even voice recognition units are readily available which will plug straight into anything which uses

MICROPROCESSORS

WHAT'S THE DIFFERENCE BETWEEN THEM?

The most commonly used microprocessor chips in the hobby market are the 8080, Z-80, 2650, SC/MP and 6502. Other variants can be easily spotted – the 8085 is very similar to the 8080 but with certain changes. The Z-8000, newly arrived on our shores is a 16 bit machine, that means it's more accurate, but as yet it has not been implemented on any system.

This is a difficult question – it's like high-level languages (of which there are many different types and variants), people who are used to a particular one will prefer it to any other. Long arguments develop between programmers over the good and bad points of each language. It's the same way with processors.

The 8080 is probably the processor with the most 'software support' – it has the most programs written for it. The Z-80 can run any program written for the 8080, as well as some which the 8080 cannot.

The SC/MP has the advantage that it needs practically no 'support chips' – it will more or less stand alone and is thus ideal for many 'dedicated' applications, such as doorbells, alarms, etc.

The major differences between the processors in terms of programming are the instruction sets and the number of registers.

The instruction set of a processor is a list of all of the different arithmetic and logical operations it can perform – like the number of keys on a calculator. The registers in the processors are the same as calculator memories – the more, the better.

The 8080 instruction set is about the same level of complexity as the 2650 and the 6502. This is adequate for most applications.

The SC/MP has a rather limited instruction set and relies on it's ease of application for its appeal.

The Z-80 instruction set includes the 8080 set – and then some! It also has twice the number of registers. In general, though, it is usually felt that the 8080 level of complexity is sufficient for the beginner.

Above: A cased version of TECS, the BASIC/ Teletext and Prestel system. It could turn your living room into a data base as it is a fully intelligent terminal, based on a 6800.

Left: The familiar Apple II colour graphics computer with an add-on disk. Much software and hardware support is available.

the S-100 bus system.

The Sorcerer has a system called the RS-232. This is a rather old fashioned I/O system which was used with teletype machines. It also has a more sophisticated system for which S-100 adaptors are available.

Both the Apple and the TRS-80 have their own systems, with S-100 adaptors being available for either.

Commodore have only recently produced add-ons for their machine, as have Exidy but because of the delay in production many companies have started to produce hardware for the popular machines to fill these gaps. With S-100 type systems however a vast range of second source hardware exists to satisfy user needs. Manufacturers like Tandy and Apple have produced add-ons for their machines, printers and disks mainly.

Cheaper Alternatives

The 'Up-Market' systems mentioned above will cost you hundreds of pounds each. What about something cheaper?

For some time there have been units on the market which are described as 'evaluation kits', intended for use by industrial concerns who are wondering whether to use a particular microprocessor or an alternative one. These kits have evolved into the 'single-board' devices available to amateurs at very low prices today.

They usually consist of a single printed circuit board, carrying all of the devices necessary to teach the user just what is going on inside a computer.

They carry a set of keys which enable you to input Hex digits and a number of 'control keys', such as 'RUN', 'DELETE' and the like.

Most of these units come with their own power supplies, so that all you have to do is to plug them in. Most of them also come with fairly good instruction manuals, although there are one or two which are not so good. Try to get a look at the instruction manual before you buy — in some cases it's a more important consideration than the unit itself.

A good instruction booklet will usually begin with something you can understand and will continue with something you can understand after you've read the first bit.

A poor instruction booklet will begin with something which you can understand and then continue with something which is totally foreign to you.

BUS SYSTEMS

S-100 AND ALL THAT

What exactly is the S-100 bus? Or any bus for that matter? No, they're nothing to do with public transport. The word 'bus' is short for 'omnibus' (literally: 'for all'). Basically, it's a method of interconnecting parts of a computer system so that they can communicate with each other.

It takes the form of a 'backplane' or 'mother board' which holds several edge connectors. Printed circuit boards can be plugged into these, one edge of the board being covered in gold-plated strips right up to its edge. Contacts on the edge connector make electrical contact with these strips. The S-100 bus system uses double-sided boards with 50 strips per side (thus the 100 in S-100!).

Each board — one of which will be the microprocessor board, holding the micro chip itself plus all the other 'support' chips necessary to get the thing to work, such as oscillators and buffers etc — has some outputs and some inputs which are connected to the bus in a standard configuration. There are sixteen lines of the bus which carry information on 'addresses'. This is how a position in memory is defined — by a sixteen digit binary number. When the microprocessor wants to find out what's at a particular address on the board which carries the memory, it puts that address on the sixteen address lines, puts out a request on some of the other lines of the bus and the memory board looks up the required information and puts it onto the 'data' section of the bus. The microprocessor board then reads the data from the bus.

Other buses have differing numbers of lines and the positions of the data and address lines are also different but they work in essentially the same manner. Unfortunately, it is difficult to connect a board intended for one bus system to a board intended for another. For this reason, each manufacturer either uses his own bus structure, or sees the light and uses the S-100, which is about as close to a standard as the hobby computer field has.

What Can You Do?

These evaluation kits are basically for education only. In order to do anything useful with them (such as add 2 + 2), you have to know how to use them and programming the problem almost always takes longer than working out the answer on paper. However for some reason the UK hobbyist seems to have made these evaluation systems do far more than you would expect. The Nascom is very popular and also expandable, Acorn seems to be headed the same way and the faithful MK14 has had more added to it than most people realize.

The output will usually not be via a VDU although there are some exceptions in which there is a TV output similar to the one on the Sorcerer or Apple.

A more usual form of output though is a number of calculator-like digits (called seven-segment displays). These can indicate the numbers and letters 0 to 9 and A to F necessary to display Hex digits.

The sort of thing you can do with the unit as it stands is to look at a specific memory location (or pigeon-hole, of those of you who learned about computers at school) to see what's in it. This is exactly the same as displaying the contents of one of the calculator's memories, except that each location will only hold (usually) two hex digits. So you enter (via the keyboard) A56B, which is the number of the location you want to look at. You then press the 'MEM' button, or whatever it's called on the particular model you have bought, and the display shows A56B 6F, which indicates that the value stored in 'pigeon hole' A56B is 6F. This won't mean much to you until you realise that A56B is the location which stores, say, the result of an addition which the machine has just carried out and that 6F is equivalent to 111 in decimal that is! Then the information becomes a bit more interesting.

In fact, the whole thing becomes fascinating once you get into it.

Once you have completely mastered this sort of unit (and that takes a lot more time than it takes to master the more up-market all-singing, all-dancing units) then you reach a bit of a problem. The problem is, do you expand your system with extra memory and facilities which inevitably means a rather untidy system or do you scrap the lot and buy a single unit machine such as PET.

Although you may have spent over a hundred pounds on the unit, you have gained well over that in education. You can now decide whether to graduate to a more flexible system, and sell your system to another beginner or carry on.

S-100 Systems

Remember S-100? Earlier in the article we mentioned it as a standard for interconnection of peripherals (add-ons such as printers, disc drives and the like) to large hobby computers.

Suppose you buy a get-you-started unit with S-100 connections on the bottom of it. These take the form of 100 'edge connections' which allow you to plug the board directly into a dirty great socket.

This socket is connected, pin for pin, with several other such sockets, all mounted on a printed circuit board called a backplane or motherboard.

Other S-100 boards, containing extra memory, or the electronics necessary to connect to a printer, or a voice synthesis board, or a TV display, (or whatever you want) can be plugged into the motherboard and are connected automatically with the main board, which carries the microprocessor.

Why not start with an S-100 system if you'll want one eventually anyway? The answer is the price (isn't it always?).

A complete S-100 system – complete enough to start computing on – will cost you just less than a large hobby computer. It's also much more difficult to get going. Some people have started on S-100 and got off the ground, but it's difficult (at the moment).

Summary

Computing is just like hi-fi. What you buy depends on how much time and money you want to spend and how enthusiastic you are (which is not always the same thing). What ever you do buy, you can rest assured that if you stick at it you will not only gain a very valuable tool, you'll have a lot of fun doing it.

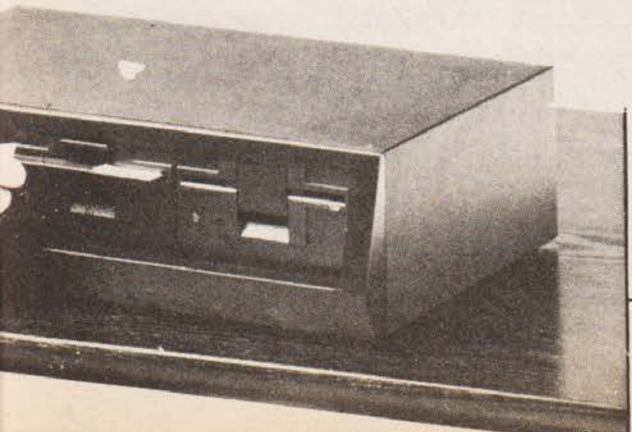
Above: The ubiquitous PET with Commodores own printer and dual disks. Right: Rair's "Black Box" system, it costs more but does a lot more including having a wide variety of languages available.

COMPLETE SYSTEMS

Name PET
Supplier: Commodore Systems, 360 Euston Road, London.
Telephone: 01-388 5702
CPU: 6502
Standard system: RAM 4-32K ROM 14K
I/O IEEE-488
Tape Internal
Keys: Full ASCII
Expansion: RAM Yes ROM Yes
Etc. Printers, disks, extra cassette, I/O.
Languages: BASIC
Price: from £460
Reviewed: April '78 ETI
Notes: The original single unit home computer, expansion from a wide variety of sources.

Name TRS 80
Supplier: Tandy Corporation, Bilston Road, Wednesbury, West Midlands.
Telephone: 021-556 6101
CPU: Z80
Standard system: RAM 4K ROM 4K
I/O RS232 Tape Yes
Keys: Full ASCII
Expansion: RAM Yes ROM Yes
Etc. Printer, disk
Languages: BASIC in various versions
Price: from £500
Reviewed: ETI August '78, April '79
Notes: An expandable complete system available in two versions for home and office use.

Name Apple II
Supplier: Keen Computers, 5 The Poultry, Nottingham + others.
Telephone: 0602-583254
CPU: 6502
Standard system: RAM 16K ROM 8K
I/O Various options
Tape Yes
Keys: Full ASCII
Expansion: RAM Yes ROM Yes
Etc. Disks, printer, speech synthesiser.
Languages: BASIC, Pascal
Price: £830
Reviewed: April '79
Notes: A cased, expandable colour graphics system with wide support.



Name: Sorcerer
Supplier: Factor One, 11-17, Market Place, Penzance, Cornwall.
Telephone: 0736-66326
CPU: Z80
Standard system:— RAM 8K ROM 4K
I/O Parallel Tape Yes
Keys: Full ASCII
Expansion:— RAM Yes ROM
Etc. S100 expansion, printer, VDU, plug-in software, I/O
Languages: BASIC, (APL, Fortran and Pascal soon)
Price: from £700
Reviewed: Supplement 4
Notes: Self contained home computer with good graphics.

Name TECS
Supplier: Technalogics, 8 Egerton Street, Liverpool.
Telephone: 051-724 2695
CPU: 6800
Standard system: RAM 4K ROM 4K
I/O Parallel + RS232
Tape KANSAS
Keys: Full ASCII + TV control
Expansion: RAM Yes ROM Yes
Etc. Disk, printer, Prestel, I/O, case.
Languages: BASIC, Assembler, Pascal
Price: from £390
Reviewed: May '79
Notes: Can be used as a full intelligent Prestel terminal, Teletext is standard.

Name Compucolor
Supplier: Abacus Computers Ltd., 62 New Cavendish Street, London W1M 7LD
Telephone: 01 580 8841
CPU: 8080
Standard system: RAM 8K ROM 16K
I/O RS232 Tape DISK
Keys: Full ASCII + graphics control
Expansion: RAM Yes ROM
Etc.
Languages: BASIC
Price: £1390, includes monitor and disk
Reviewed: June '79
Notes: Colour graphics system with inbuilt monitor and disk.

Name 380Z
Supplier: Research Machines, PO Box 75, 209 Cowley Road, Oxford.
Telephone: 0865-49792
CPU: Z80A
Standard system: RAM 4K ROM 4K
I/O RS232 + parallel
Tape CUTS
Keys: Full ASCII
Expansion: RAM Yes ROM Yes
Etc. Disks, printer, I/O, various TV monitors.
Languages: BASIC, Assembler, Fortran, Algol, Cobol.
Price: £1025 incl cassette and monitor.
Reviewed: April '79
Notes: Expandable cased system used in education and scientific fields.

KITS AND SINGLE BOARD COMPUTERS

Name NASCOM 1
Supplier: Nascom Microcomputers, 92 Broad Street, Chesham, Bucks.
Telephone: 02405-75151
CPU: Z80
Standard system: **RAM** 2K **ROM** 1-2K
I/O Parallel + RS232
Tape Yes
Keys: Full ASCII
Expansion: **RAM** Yes **ROM** Yes
Etc. Moter
Etc. Motherboard, frame PROM blower, graphics, I/O boards
Languages: BASIC, Assembler, M5
Price: £178
Reviewed: Supplement 1, April '79
Notes: OEM expandable system.

Name Superboard II
Supplier: Lotus Sound, 4 Morgan Street, London E3 5AB + others.
Telephone: 01-981 3993
CPU: 6502
Standard system: **RAM** 4K **ROM** 10K
I/O Parallel
Tape KANSAS
Keys: Full ASCII
Expansion: **RAM** Yes **ROM**
Etc. Case, Extended monitor
Languages: BASIC
Price: £188
Reviewed: July '79
Notes: Single board system with BASIC

Name AIM 65
Supplier: Pelco Electronics, Enterprise House, 83-85 Western Road, Hove, Sussex.
Telephone: 0273-722155
CPU: 6500
Standard system: **RAM** 2-4K **ROM** 8K-20K
I/O RS232 + parallel
Tape 2
Keys: Full ASCII + printer
Expansion: **RAM** Yes **ROM**
Etc. Motherboard, case bubble memory.
Languages: Optional BASIC
Price: £250
Reviewed:
Notes: Single board system with integral printer.

Name ELF
Supplier: Newtronics, 138 Kingsland Road, London N1
Telephone: 01-739 1582
CPU: 1802
Standard system: **RAM** 256 byte **ROM** None
I/O **Tape** No
Keys: Hex pad
Expansion: **RAM** Yes **ROM** Yes
Etc. Motherboard, tape I/O, ASCII keyboard, VDU.
Languages: BASIC ontape
Price: from £79
Reviewed: October CT
Notes: 1802 development kit but expandable.

Name TRITON
Supplier: TransAm, 12 Chapel Street, London NW1 5DH
Telephone: 01-402 8137
CPU: 8080
Standard system: **RAM** 4K **ROM** 4K
I/O Parallel + RS232
Tape Yes
Keys: Full ASCII
Expansion: **RAM** Yes **ROM** Yes
Etc. Motherboard, printer, PROM blower, various monitors.
Languages: Various BASIC's, Assembler.
Price: from £286
Reviewed: ETI Nov '78
Notes: Single board cased system in kit form but expandable in both hardware and firmware.

Name MK14
Supplier: Science of Cambridge, 6 Kings Parade, Cambridge.
Telephone: 0223-311488
CPU: SC/MP
Standard system: **RAM** 2566 byte **ROM** 512 byte
I/O Parallel **Tape**
Keys: Hex pad
Expansion: **RAM** Yes **ROM**
Etc. Tape, VDU, PROM blower
Languages: Machine code
Price: £47
Reviewed: ETI Sept '78
Notes: SC/MP evaluation system, a good starting point.

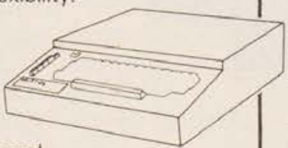
Name Nanocomputer
Supplier: Midwich Computer Company, Hillsborough House, Churchgate Street, Old Harlow, Essex.
Telephone: 0279-25756
CPU: Z80
Standard system: **RAM** 4-4K ROM 4K16K
Standard system: **RAM** 4-16K **ROM** 2-4K
I/O Parallel + RS232
Tape Yes
Keys: Hex terminal
Expansion: **RAM** Yes **ROM** Yes
Etc. Disk, frame, Experiment kit.
Languages: BASIC, Assembler
Price: from £260
Reviewed: Aug '79
Notes: OEM system for education but fully expandable.

Name: Acorn
Supplier: Acorn Computers Ltd., 4A, Market Hill, Cambridge.
Telephone: 0223-312772
CPU: 6502
Standard system:– **RAM** 1K **ROM** ½K
I/O Parallel **Tape** CUTS
Keys: Hex pad
Expansion:– **RAM** Yes **ROM** Yes
Etc. VDU, disk, I/O
Languages: BASIC soon
Price: from £75
Reviewed: August '79
Notes: A 6502 evaluation kit of good design and suitable for wide expansion.

TRANSAM COMPUTER PRODUCTS

TRITON SINGLE BOARD PERSONAL COMPUTER

Three new exciting expandable systems designed for ease of construction and flexibility. Kits come complete with case, power supply, full keyboard, PCB. All components available separately, see catalogue. Full hardware. Programming manual available. The system is easy to expand and is well supported. Features – 2, 2.5 or 7K Basic in Eprom (see catalogue).



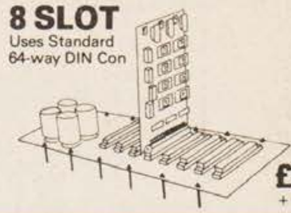
- •Single Board Holds up to 8K Memory
- •Cassette Interface
- •Three Firmware options
- •Basic in Eprom
- •Plug-in Expansion Boards

Personal Computer £286 + VAT From

EXPANSION MOTHERBOARD

8 SLOT Uses Standard 64-way DIN Con

TRITON. Expand your Triton simply and easily with our new 8-slot motherboard complete with its own P.S.U. takes 8 plug-in Euro cards. Plug-in 8k RAM card and Eprom cards now available. Kit complete with PSU + 1 set connectors.

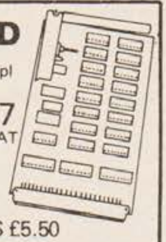


£50 + VAT

8K RAM CARD

Triton 8k Static RAM card kit uses 2114L low power 4k static RAMS. On board regulation. Mem jump select.

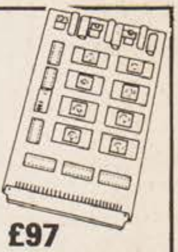
Compl Kit £97 + VAT



PCB only £15 – RAMS £5.50
Kit less RAMS £31 incl 5SKTS & components

8K EPROM CARD

Triton 8k Eprom card kit designed to take up to 8 x 2708 Eproms (1k x8) as RAM card



PCB only £15
Kit less Eproms
Eproms (blank) £9

£97 Complete kit + VAT

BI DIRECTIONAL MATRIX PRINTER £595 + VAT

The BD80 is a low cost, 80 column line printer with microprocessor control to provide excellent reliability and performance

- •5 x 7 Matrix
- •10 Char per inch
- •6 lines/inch
- •400 Char Buffer
- •Full ASC II Char Set
- •10 lines/sec Paper Advance
- •112 Char/sec
- •84 lines per minute
- •Self Test
- •Fully Cased

A UNIQUE PRINTER FAST AND RELIABLE

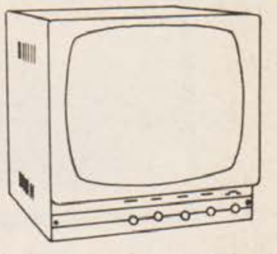
Switch selectable baud rate from 110 to 9600 on a standard V24 and RS232 Interface – Send SAE for further details. Ideal printer for Triton or any system requiring high speed, reliable, hard copy. We can supply consumables.

VIDEO MONITOR

NEW

A brand new fully cased (metal) high resolution 10" video monitor with PSU for only £69 + VAT Ideal for Triton or anyy home computer system. Carriage by Securior can be arranged. Send SAE for details or see our new catalogue

ONLY **£69!** + VAT



PCB CONNECTORS

Edge connectors gold contact double sided PCB connectors

.1"	Price	.156"	Price
22/44	£3.20	6/12	£1.62
25/50	£3.60	10/20	£1.90
28/56	£3.90	15/30	£3.00
30/60	£4.15	18/36	£3.00
85/70	£4.60	22/44	£3.00
36/72	£4.75	28/56	£3.40
40/80	£5.00	36/72	£3.90
43/86	£5.50	43/82	£4.60
50/100	£5.80	(S100 Bus)	

+ VAT

VIDEO DISPLAY INTERFACE MODULE

Completely built and tested the SFXE x 68364 card uses the industry standard SPF 96364 CRT control chip and allows ASC II parallel input of data to be output to video monitor 64 character by 16 line display full cursor control. Single 5v supply. Full details available on request (Send SAE). UHF modulator available, £2.50 extra + VAT.

£69 + VAT

COMPONENTS 74LXX

								SN74LS324N	1.80
SN74LS00N	.18	SN74LS54N	.21	SN74LS136N	.40	SN74LS194AN	1.89	SN74LS325N	2.55
SN74LS01N	.18	SN74LS55N	.21	SN74LS138N	.75	SN74LS195AN	.85	SN74LS326N	2.55
SN74LS02N	.20	SN74LS56N	.21	SN74LS139N	.75	SN74LS196N	1.20	SN74LS327N	2.56
SN74LS03N	.18	SN74LS63N	1.50	SN74LS145N	1.20	SN74LS197N	1.20	SN74LS352N	1.35
SN74LS04N	.20	SN74LS73N	.35	SN74LS148N	1.75	SN74LS221N	1.25	SN74LS353N	1.50
SN74LS05N	.26	SN74LS73N	.40	SN74LS151N	.85	SN74LS240N	2.20	SN74LS365N	.65
SN74LS08N	.20	SN74LS75N	.46	SN74LS153N	.60	SN74LS241N	1.90	SN74LS366N	.65
SN74LS09N	.22	SN74LS76N	.35	SN74LS154N	1.60	SN74LS242N	1.90	SN74LS367N	.65
SN74LS10N	.18	SN74LS78N	.35	SN74LS155N	1.25	SN74LS243N	1.95	SN74LS368N	.65
SN74LS11N	.26	SN74LS83AN	1.15	SN74LS156N	1.25	SN74LS244N	2.10	SN74LS373N	1.75
SN74LS12N	.25	SN74LS85N	1.10	SN74LS157N	.60	SN74LS245N	2.60	SN74LS374N	1.70
SN74LS13N	.55	SN74LS86N	.40	SN74LS158N	.99	SN74LS247N	1.25	SN74LS375N	.72
SN74LS14N	.89	SN74LS90N	.65	SN74LS160N	1.15	SN74LS248N	1.95	SN74LS377N	1.75
SN74LS15N	.25	SN74LS91N	.99	SN74LS161N	1.15	SN74LS249N	1.30	SN74LS378N	1.32
SN74LS20N	.20	SN74LS92N	.90	SN74LS162N	1.15	SN74LS251N	1.45	SN74LS379N	1.40
SN74LS21N	.26	SN74LS93BN	.65	SN74LS163N	.90	SN74LS253N	1.25	SN74LS381N	3.65
SN74LS22N	.26	SN74LS95AN	1.20	SN74LS164N	1.50	SN74LS257N	1.40	SN74LS386N	.57
SN74LS26N	.29	SN74LS96N	1.75	SN74LS165N	1.70	SN74LS258N	.95	SN74LS390N	1.98
SN74LS27N	.35	SN74LS107N	.39	SN74LS166N	1.75	SN74LS259N	1.45	SN74LS393N	1.50
SN74LS28N	.35	SN74LS109N	.39	SN74LS168N	1.95	SN74LS260N	.39	SN74LS395N	1.80
SN74LS30N	.25	SN74LS112N	.39	SN74LS169N	1.95	SN74LS261N	3.50	SN74LS396N	1.70
SN74LS32N	.27	SN74LS113N	.44	SN74LS170N	2.50	SN74LS266N	.39	SN74LS398N	2.75
SN74LS33N	.39	SN74LS114N	.44	SN74LS173N	2.20	SN74LS273N	1.85	SN74LS399N	1.60
SN74LS37N	.29	SN74LS122N	.79	SN74LS174N	1.15	SN74LS279N	.79	SN74LS424N	4.50
SN74LS38N	.29	SN74LS123N	.90	SN74LS175N	1.05	SN74LS280N	1.75	SN74LS445N	1.25
SN74LS40N	.25	SN74LS124N	1.50	SN74LS181N	2.75	SN74LS283N	1.80	SN74LS447N	1.25
SN74LS42N	.79	SN74LS125N	.65	SN74LS190N	1.75	SN74LS290N	1.80	SN74LS490N	1.95
SN74LS47N	.95	SN74LS126N	.65	SN74LS191N	1.75	SN74LS293N	1.80	SN74LS668N	.95
SN74LS48N	.95	SN74LS132N	.75	SN74LS192N	1.45	SN74LS295AN	2.20	SN74LS669N	.95
SN74LS49N	1.09	SN74LS133N	.39	SN74LS193N	1.75	SN74LS298N	2.20	SN74LS670N	2.70

MEMORY AND SUPPORT CHIPS

(prices exclude VAT)

SUPPORT		RAMS		EPROMS		LM555N	.30	VOLT REGS		CRYSTALS		MISCLE	
8212	2.20	2101	2.32	1702	6.00	LM556N	.75	7805	.90	100K	3.00	2513	7.50
8216	2.20	2102L-4	1.20	5204	8.00	LM709CN	.37	7812	.90	200K	3.70	TMS6011	5.00
8224	2.80	2111	2.32	2708	9.00	LM723CH	.58	7815	.90	1MHZ	3.60	MC14411	12.00
8226	2.20	2112	2.46	2516	28.00	LM723CN	.43	7824	.90	1008K	3.50	MC14412	12.90
8228	4.20	6810	4.08	2716	22.00	LM733CN	1.30	7805K	1.50	1843K	3.00	96364	10.95
8238	4.20	8154	8.18	**ROMS**		LM739CN	1.30	7812K	1.50	2MHZ	1.50	**CPU's**	
8245	11.00	2114	5.50	74S287	3.70	LM741CN-14	.33	7815K	1.50	2457K	3.05	8080	6.33
8246	11.00	2102L-3	1.60	74S472	12.00	LM741CN-8	.25	7824K	1.50	3276K	2.70	6800	10.00
8251	5.00	74C920	11.00	74S70	8.00	LM747CN-14	.79	7905	1.10	3MHZ	3.05	Z80	8.00
8253	11.00	74C921	11.00	74S473	12.48	LM747CH	1.19	7912	1.10	4MHZ	2.10	Z80A	15.00
8255	5.00	74C929	11.00	74S474	12.48	LM748CN-8	.45	7915	1.10	4.43M	1.00	8085	12.95
8257	11.00	4027	11.00	**I/O**		LM748CH	.46	7924	1.10	5MHZ	2.70	6502	8.00
8259	12.50	4044	14.70	2513	7.50	LM1458H	.72	7905K	1.80	6MHZ	2.70	SCMP II	10.00
8292	18.00	4045	9.15	96364	10.95	LM1458N-8	.48	7912K	1.80	7MHZ	2.70	6802	13.95
6820P	4.50	4060	7.00	14412	12.90	LM1488D	855	7915K	1.50	7.168M	2.50	9900	30.00
6821P	4.50	2107	7.80	**LINEARS**		LM1489D	.85	7924K	1.80	8MHZ	2.70	**W/WRAP SKTS**	
6850P	4.60	4116	8.00	LM301AH	.39	LM1489AD	1.25	**DIL SKTS**		10MHZ	2.70	8DIL	.20
6852P	5.50	4118	20.00	LM301AN-8		LM1495N-14	.99	8DIL	.14	10.7M	2.70		
AY-5-2376	11.50	80P20	10.00	LM (Mini Dip)	.30	LM3302N	.65	14DIL	.15	18M	2.90	14DIL	.35
MC14411	12.00	Z80CTC	10.00	LM308N	.99	LM3401N	.65	16DIL	.17	48M	2.90	16DIL	.42
M57109	12.43	Z80AP10	14.00	LM309K		LM3403N	1.20	18DIL	.24	**CMOS**		18DIL	.60
M57160	10.00	Z80ACTC	14.00	LMT03	1.45	LM3900N	.54	20DIL	.27	CD4011	.15	24DIL	.52
M57161	10.00			LM311H	1.29	TL080CP	1.49	24DIL	.30	CD4040	.79	28DIL	.74
TMS6011	5.00			LM318H	2.25	TL081CP	.69	28DIL	.36	+ full range		40DIL	.95
81LS95	1.80			LM323K	6.00	TL082CP	1.29	48DIL	.50				
81LS96	1.80			LM324N	.79	TL083CN	1.65						
81LS97	1.80			LM339N	.54	TL085CN	1.68						
81LS98	1.80												

TRITON DOCUMENTATION

available separately as follows, prices include p & p

Triton manual – detailed circuit description and constructional details plus user documentation on level 4.1 monitor and basic	£5.70
L4.1 listing – listing of 1K monitor and 2K tiny basic	£4.20
L5.1 user documentation on level 5.1 firmware	£1.20
L5.1 listing – listing of 1.5K monitor and 2.5K basic	£5.20
L6.1 user documentation on 7K basic interpreter	
Motherboard, 8K RAM and 8K EPROM constructional details	SAE
User group newsletter subscription £4 per annum Triton software – SAE for list of programs available for Triton.	Send

HOME COMPUTING CATALOGUE

If you're in town, visit our showroom in Chapel Street, next to Edware Road tube station. We have Tritons on display plus a comprehensive range of components and accessories, specifically for personal computer users. Books, mags, tapes, data, cables plus much more. Showroom open 6 days a week. (Half day Thurs. from 1.30).

NEW

A4 size catalogue filled with our latest products

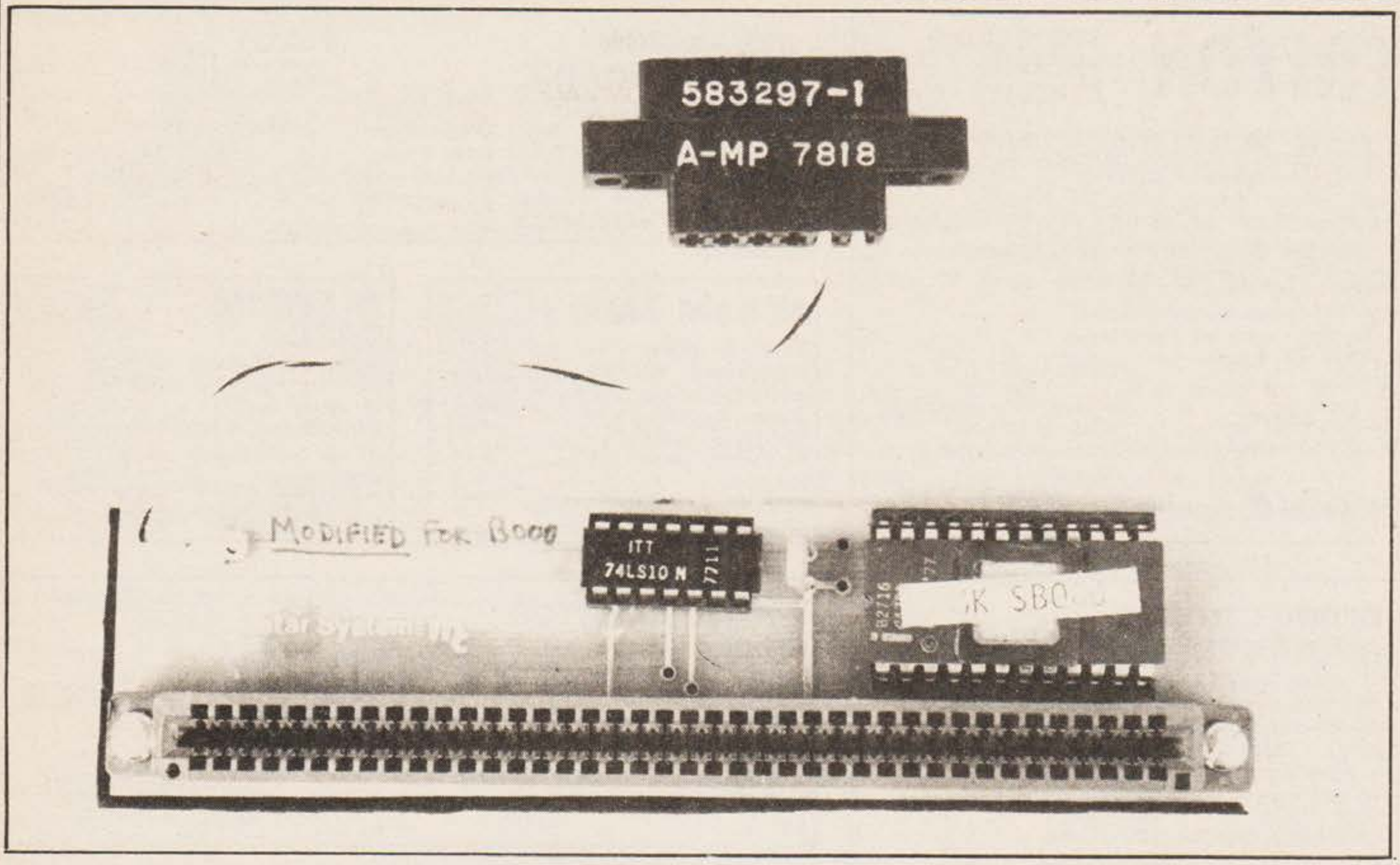
40p + SAE

ALL PRICES EXCLUDE VAT

TRANSAM

VAT ALL PRICES EXCLUDE VAT

TRANSAM COMPONENTS LTD.
12 CHAPEL STREET
LONDON, NW1 TEL: 402 8137



Firmware magic for the PET

Once upon a time a home computer was designed. The company called it the PET and it became one of the best selling home computers in the world. However because of the lack of skilled programmers using PET mistakes were made, errors crept into programs and users got frustrated because they couldn't do simple things. If this sounds like a fairy tale to you and you are waiting for the happy ending — wait no longer.

Firmware Goes Faster

The new product which has changed the PET from a "dumb" system into a reasonably "intelligent" one is the Programmers Toolkit. It contains a set of machine code routines, their functions are listed in Table 1, which provide for the user a number of utility type commands.

The pre-production version that we were lent was designed for use on an old-ROM machine. As can be seen in the photograph it simply consists of the firmware ROM, an IC and an edge connector socket that plugs on to the memory expansion port. Power is supplied to the unit by the second plug that goes into the spare cassette interface. Fitting the unit took about 30 seconds and provided you remember to turn your machine off first no problems were encountered.

A second version of the Toolkit is available for new-ROM machines, this plugs directly into the spare left hand empty socket inside, again fitting should not be a problem.

Power On . . . . . Power Up

Once fitted the new system is initialised by typing a single command SYS 45056, and the device echoes back the fact that it is loaded. That's all there is to it really, you can use your PET as normal without even knowing you have something extra under the bonnet as it were.

What exactly do you get? Well to start with you can type your programs in without using line numbers. By use of the command AUTO X, Y (where X is the starting line and Y the interval between lines) your line numbers are automatically put on the screen for you each time you press RETURN. To end program input you have to type RETURN twice. Having written your program you will now wish to run it and this is where the real power of the Toolkit appears.

Debugging Software

The range of debugging aids is extensive, seven in all, and they allow you to correct virtually any mistake in your BASIC program. If we assume you have created a syntax error in the program when you try to run it all you get back is ?SYNTAX ERROR. Very helpful! Type HELP and the Toolkit returns the first line it finds with a syntax error and displays the offending character in reverse field so you can see it. Edit the line and carry on.

If you find, as we always do, that you have had to add a line or two in at a later date and you want to tidy up the program just type RENUMBER X, Y (where X and Y are as explained before) and the Toolkit does the rest, and very quickly too.

One of the really boring things about BASIC is you have to delete each line separately, with the Toolkit you can DELETE 100, 350 or any given numbers in one command.

To see just how or why your program is not working quite as you thought there are two other commands, TRACE and STEP. The TRACE command followed by RUN executes your program at about a tenth of the normal speed and always shows the last six line numbers of execution in the top right hand corner of the screen. This allows you to check loops and subroutine jumps. You can stop and start the program at any point by use of the RUN/STOP key. The STEP command causes the PET to execute one command at a time by use of the SHIFT key, holding it down runs the program continuously at about one tenth normal speed. Both these commands can be turned off by the command OFF.

Two other commands are available for debugging purposes, DUMP which lists all variables and strings used in the program and FIND which allows you to list all occurrences of a specified character string, eg GOTO or A$ or FRED.

Last But Not Least

One final command is available on the system, APPEND. This is a command that allows you to add previously written bits of program to the one you are currently working on. All you do is type APPEND "filename" and then press PLAY on the cassette. When it finds the program on tape it appends it to the one in memory. All you have to do then is to tidy it up and carry on. Start your subroutine library now!

Conclusions

The device we have described above allows you to perform a number of useful functions on any BASIC program. Because it becomes part of your PET you don't lose any memory space, merely gain power. It will be marketed in this country by Petsoft from September at a cost of £75 for the plug on version or £55 for the single ROM. The device has Commodores approval for use with PET and we think that it is an exceptionally good piece of equipment. Try one at your local dealer and see if you are as impressed.

Table 1

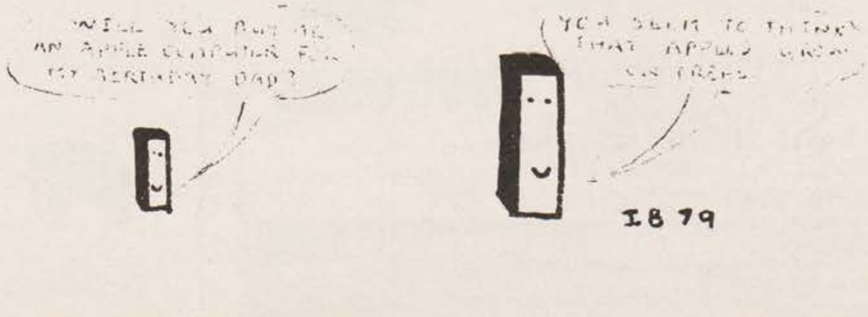
COMMAND	Function
AUTO	Automatic line numbering, any start, any step
DELETE	Bulk deletion, any number to any number
RENUMBER	Automatic re-numbering, any start, any step
HELP	Returns error in reverse field
TRACE	Displays last six executed line numbers in top right of screen
STEP	Single step with SHIFT key, displays line numbers as TRACE
OFF	Turns off TRACE and STEP
APPEND	Builds programs from library tape
DUMP	Displays all variables and strings used
FIND	Displays all occurrences of a specified string or command etc.

Acknowledgement: We are most grateful to Julian Allason of Petsoft for letting us have one of the only two 'Toolkits' available at the time of writing this review.

Mr.L.E.Gilham

With respect and acknowledgement to Mr I Davidson, this is his Etch-A-Sketch program (Softspot May'79) modified to run on a standard Nascom 1 with a T2 monitor.

```
0C50   3E   CLEAR
0C51   1E   SCREEN
0C52   CD
0C53   3B
0C54   01
0C55   21   POINTER TO
0C56   E1   CENTER OF
0C57   09   SCREEN
0C58   CD   CALL CHIN
0C59   3E
0C5A   00
0C5B   47   LD B WITH A
0C5C   70   LD (HL) WITH B
0C5D   CD   CALL CHIN
0C5E   3E
0C5E   00
0C60   FE   CP
0C61   52   'R'
0C62   CA   IF 'R' THEN
0C63   77   JP 0C77
0C64   0C
0C65   FE   CP
0C66   4C   'L'
0C67   CA   IF 'L' THEN
0C68   7B   JP 0C78
0C69   0C
0C6A   FE   CP
0C6B   55   'U'
0C6C   CA   IF 'U' THEN
0C6D   7F   JP 0C7F
0C6E   0C
0C6F   FE   CP
0C70   44   'D'
0C71   CA   IF 'D' THEN
0C72   8A   JP 0C8A
0C73   0C
0C74   C3   IF NONE OF
0C75   5D   ABOVE BACK
0C76   0C   TO 0C5D
MOVE RIGHT
0C77   23   INC HL
0C78   C3   JP TO
0C79   5C   0C5C
0C7A   0C
MOVE LEFT
0C7B   2B   DEC HL
0C7C   C3   JP TO
0C7D   5C   0C5C
0C7E   0C
MOVE UP
0C7F   97   CLEAR ACC
0C80   2B   DEC HL
0C81   3C   INC ACC
0C82   FE   CP
0C83   40   40H
0C84   C2   LOOP
0C85   80   FOR
0C86   0C   MATCH
0C87   C3   JP TO
0C88   5C   0C5C
0C89   0C
MOVE DOWN
0C8A   97   CLEAR ACC
0C8B   23   INC HL
0C8C   3C   INC ACC
0C8D   FE   CP
0C8E   40   40H
0C8F   C2   LOOP
0C90   8B   FOR
0C91   0C   MATCH
0C92   C3   JP TO
0C93   5C   0C5C
0C94   0C
```



Well - do you know how it works?

It has been observed that not only does a language (English, French etc!) provide a means of expressing thoughts, it also tends to limit thoughts, to those expressible in the language whether to others, or to one's self. Just as a road provides a path for transportation, if you get used to travelling by car you kind of have to stick to the road, and the surrounding terrain remains undiscovered.

So it is with computer languages. Iverson Notation was developed for humans to use, to advance their ability to think and solve problems. APL grew from this, and is probably the only language to pick up an already developed notation, as opposed to acting as a compromise between English and machine code.

Thus while other languages tend frequently to constrict one in trying to solve a problem, learning and using APL tends to help one to handle the problem. In fact with experience one seems to be able to write an APL program while thinking up the method of solution. Other languages usually require an intermediate block diagram or flow chart stage for a similar problem. The proof of this lies in the absolutely fanatical following for APL, and the well attended and highly regarded APL conferences which take place to discuss applications and possible new features for the language.

We think APL is great for the mind and body, and that it's about to arrive in a big way. This article deals with what it is, where it's been, and a look at when it should be used, and where it's going.

What It's Like

In as much as APL is quite different from other computer languages, has vastly different qualities, and has fanatical supporters, it is very useful to know something about the language itself.

APL having originally evolved from Iverson Notation, it is most helpful to start with those features which are actually part of the notation.

Iverson Notation

The first step to getting a grasp on Iverson Notation is to be familiar with a few 'buzz-concepts' essential to the tropic.

Essentially Iverson Notation is a mathematical notation which permits expressions to be written in very compact and condensed forms. The idea is that by giving a lot of the most used functions single symbol names (instead of just + — x and ÷) the user can not only more easily write his ideas down, but even learn to think on a higher level. A good example of this is a sorting operation, for example putting a list of names in alphabetical order. Iverson Notation gives you the concepts of how to think of the list of names, and then allows you to write down, using just a couple of symbols, the method used to do it. Example later.

Data Type

In a mathematical expression, one may have a variety of different types of data. Most people are familiar with integers and real numbers, but also letters, words, collections of characters can be data.

Do not confuse, however, character data with names for variables. For example, in the ordinary algebraic expression A=2xB, A and B are variables while 2 is a constant. On the other hand, in a computer context it would be quite permissible for C='ABCD', ie 'ABCD' is the 'value' of variable C.

Rank

In ordinary algebra we mostly use variables of a single 'element', known as scalars (eg a=2). Sometimes vectors are used, really a list of numbers known by one variable name (eg a=3, 7, 2, -1, or a(1)=3, a(2)=7, a(3)=2 and a(4)=-1). 'Matrices' are especially popular for such applications as solution of complicated equations, and have two 'subscripts'. Example:

```
a = 1  3  0
    7  5  4
    8  7  2
```

or a(1,1) = 1, a(2,1) = 7, a(3,3) = 2, etc.

The concept of using subscripts can be extended on and on to any number of 'dimensions'. The general term for all such variables is 'array'. The number of dimensions or subscripts, it has is known as its 'rank'. Hence a (b,c,d) has rank of 3, a 2 dimensional matrix has a rank of 2, a vector rank 1, and a scalar has rank 0.

APL Applied

APL's big strength stems from the ability of a person familiar with the language to write a program as fast as thinking of the method. In fact one even tends to think of the solution in APL form. The other influential feature is that APL is an interactive language.

These two combine to produce an ideal way to use a computer for applications where the programs need to be ready fast, and where the applications area may be new (and hence the program is experimental and may require much revision). This is not to say that in other applications APL is not suitable, but until recently APL has been though of as expensive in terms of execution time and memory use, or even simply to obtain access. In addition compared to other languages there are relatively few experienced APL programmers, the number familiar with the language will of course grow.

One estimate holds that an APL programmer can produce a program on the order of 25 times the speed of a COBOL programmer, (including debugging) and that the result is of course a far more compact listing.

One point to watch, however, is that while almost anyone can plough through a BASIC, FORTRAN or COBOL program (with enough patience), it is possible to program in APL in such a dense and tricky manner as to make the program virtually impossible to decipher by other persons. Programmers who do this also enjoy reducing 100 line FORTRAN programs to an APL one-liner. (There seems to be some special attraction to the 'one-line' challenge!).

However, if a program is written with reading also in mind, with a bit of descriptive documentation, there is no problem.

APL EXPLAINED

What APL Looks Like

SCALAR DYADIC FUNCTIONS

X+Y		X plus Y
X-Y		X minus Y
X×Y		X times Y
X÷Y		X divided by Y
X*Y		X to the Y-th power
X⌈Y		Maximum of X and Y
X⌊Y		Minimum of X and Y
X∣Y		X-residue of Y
X⍟Y		Base-X logarithm of Y
X!Y		Binomial coefficient: Y items taken X at a time
X○Y		See trigonometric functions
X<Y	•	X less than Y
X≤Y	•	X less than or equal to Y
X=Y	•	X equal to Y
X≥Y	•	X greater than or equal to Y
X>Y	•	X greater than Y
X≠Y	•	X not equal to Y
X∧Y		X and Y
X∨Y		X or Y
X⍲Y		Not both X and Y
X⍱Y		Neither X nor Y

TRIGONOMETRIC FUNCTIONS

R←X○Y

(Y in radians)		(R in radians)	
X	R	X	R
0	(1-Y*2)*.5		
1	Sine Y	¯1	Arcsin Y
2	Cosine Y	¯2	Arccos Y
3	Tangent Y	¯3	Arctan Y
4	(1+Y*2)*.5	¯4	(¯1+Y*2)*.5
5	Sinh Y	¯5	Arcsinh Y
6	Cosh Y	¯6	Arccosh Y
7	Tanh Y	¯7	Arctanh Y

SCALAR MONADIC FUNCTIONS

+Y		Y
-Y		0-Y
×Y		Signum Y
÷Y		Reciprocal of Y
*Y		e to the Y-th power
⌈Y	•	Ceiling of Y
⌊Y	•	Floor of Y
∣Y		Absolute value of Y
⍟Y		Natural logarithm of Y
!Y		Factorial Y; Gamma Y+1
○Y		Pi times Y
?Y	#	A random number from ⍳Y
~Y		Not Y

• ⎕CT dependent
⎕IO dependent

SYMBOLS

( )	Parentheses for nesting
[ ]	Brackets for indexing
⎕	Quad for input-output
⍞	Quote-quad for character input
'	Quote delimits character literals
⍝	Lamp indicates comment
¯	Negative sign
E	Exponential notation
∆	Delta Trace (T∆) and stop (S∆) control
⌶	I-beam for system functions
.	Decimal point
∧	Caret locates errors
;	Semicolon separates list elements
◊	Diamond used as statement separator
→	Branch
:	Colon delimits labels and locks
⌷	Squish-quad display of non-printable characters

The following characters are available as graphics:

⍎ α { ⊢ ⊂ ∩ ¨
⍕ ⍵ } ⊣ ⊃ ∪ _

FUNCTION DEFINITION

Function definition is opened or closed by ∇

Use of ⍫ at the close will lock the function.

The following are valid with open definition

[⎕]	Display entire function
[n⎕]	Display line n
[⎕n]	Display from line n to the end
[n⎕p]	Display line n and position at p prior to editing
[n⎕0]	Display line n and position at right-hand end of line
[n]	(Re)define line n

MIXED FUNCTIONS

X⍴Y		Reshape Y to have dimensions of X
⍴Y		Dimensions of Y
X[Y]	#	Y-th elements of X
X⍳Y	• #	First locations of Y within vector X
⍳Y	#	Y consecutive integers from origin (0 or 1)
X∊Y	•	Membership of X in Y
X⊤Y		Representation of Y in number system X
X⊥Y		Value of the representation Y in number system X
X?Y	#	X integers selected randomly without repetition from ⍳Y
X⌽[Z]Y		Rotation by X along the Z-th dimension of Y
⌽[Z]Y		Reversal along the Z-th dimension of Y
X⍉Y	#	Transpose of Y according to X
⍉Y		Transpose of Y (⌽⍳⍴⍴Y)⍉Y
X,[Z]Y		Catenate or laminate Y to X along Z-th dimension
,Y		Ravel of Y (makes Y a vector)
X↑Y		Take first or last X elements of Y as X is + or -
X↓Y		Drop first or last X elements of Y as X is + or -
X←Y		X is assigned the value of Y
⍋Y	#	Index vector such that Y[⍋Y] is in ascending order
⍒Y	#	Index vector such that Y[⍒Y] is in descending order
X/[Z]Y		X (logical) compressing along the Z-th dimension of Y
X\[Z]Y		X (logical) expanding along the Z-th dimension of Y
R←X⌹Y		Matrix divide. X approximates R+.×Y
⌹Y		Generalized matrix inverse of Y
⍎Y		Executes the character string Y
⍕Y		Character representation of Y
X⍕Y		Numeric expression Y is converted to character according to numeric format control vector X

• below denotes any scalar dyadic function

•\[Z]Y	• applied cumulatively along the Z-th dimension of Y
•/[Z]Y	The • reduction along the Z-th dimension of Y
X∘.•Y	Generalized outer product of X and Y
X•.•Y	Generalized inner product of X and Y e.g. X+.×Y is ordinary matrix product of X and Y
⊖	⌽ applied to first dimension (reverse or rotate)
⌿	/ applied to first dimension (compress or reduce)
⍀	\ applied to first dimension (expand or scan)

If the expression [Z] is omitted, the operation applies to the last dimension of the argument array Y. The expression [Z] is ⎕IO dependent.

• ⎕CT dependent
⎕IO dependent

SYSTEM COMMANDS

Note: Bracketed words are optional.

)[n]	Allow editing of immediate execution line at position n
)nnnnnnn [:lock]	Sign-on - invalid sign-on acts as)BLOT
)BLOT	Print a mask for typing secure information
)CLEAR	Clear the active workspace
)CONTINUE [HOLD] [:lock]	Terminate a session and store the active workspace in CONTINUE
)COPY name [:lock] [list]	Copy objects from a workspace
)DROP name [:lock]	Delete a stored workspace
)ERASE list	Erase objects
)FNS [letters]	List names of functions
)GROUP name list	Define a group
)GROUP name	Disperse a group
)GRP name	List members of a group
)GRPS [letters]	List names of groups
)KEYB LOCK	Lock keyboard (to receive messages)
)KEYB NOMSG	Suppress incoming messages
)KEYB	Return keyboard to normal state
)LIB [nnn]	List names of workspaces in library
)LOAD name [:lock]	Activate a copy of a stored workspace
)MSG taskid text	Send message to designated terminal
)MSGN taskid text	Send message to designated terminal without locking sender's keyboard to receive reply
)OFF [HOLD] [:lock]	Terminate a work session
)OPR text	Send message to SHARP APL operator
)OPRN text	Send message to SHARP APL operator without locking sender's keyboard to receive reply
)PCOPY name [:lock] [list]	As)COPY, but protect contents of active WS from being overwritten
)PORTS aaa	List port number and task-ID of user aaa
)PORTS nnnnnnn	List port number and task-ID of user nnnnnnn
)RESET	Clear the)SI stack
)SAVE [name] [:lock]	Store a copy of the active workspace
)SEAL	Lock all functions in the workspace, prevent dispersal of contents
)SI	Display the state indicator
)SIV	Display state indicator and names of local variables
)SYMBOLS [n]	Display/set symbol table size
)TERM [name]	Display/set terminal type. See WS 5 TERM for current list of names
)VARS [letters]	List names of variables
)WSID [name]	Display/set workspace name

Fig. 1. I.P. Sharp Associates provide a handy reference card for programmers using their systems. It is just about all you would ever need to remember or take with you to a desert island equipped with only an APL computer, on a scant 224 square inches of card. No more heavy manuals to lug around! Most of the information in this exerpt is general, we have included the section on system commands to give an idea of what is typical. The special characters shown here are those produced on an impact kind of typewriter terminal.

Functions

APL has a large number of characters which denote functions (examples are the familiar + − x ÷ but there are many more exciting ones! A couple of useful words to know here are 'monadic' and 'dyadic' which mean 'having one argument' and 'having two arguments' respectively. Note that each argument may be an array. Thus A+B may yield a scalar if A and B are scalar, but if they are arrays of the same size and shape, each element of A is added to the corresponding element of B to make the corresponding element of the resulting array.

Consistency And Execution Order

One of the great virtues of Iverson Notation is its consistency. All monadic functions are written with the one function symbol to the left of the argument, such as !F which means the factorial of F, ÷X which is the reciprocal of X, or ?Y, a random selection from the first Y integers, while A means absolute value of A. And remember F,X,Y and A could be arrays! In fact, the APL user tends to think that the basic data form is the array, which occasionally 'happens' to be a scalar or vector.

Similarly, all dyadic functions are written with the symbol between the two arguments.

Finally, execution order is right to left, regardless of functions (ie x does not take precedence over +). The order may be changed using brackets. This order is consist-

```
        ACCUM[0]
    ∇ Z ACCUM A;B;L
[1]   A IMPROVED
[2]     A←A[↑A[;1];]
[3]     B←A[;1]
[4]     L←(1↓B) ≠ 1 B
[5]     L←L,1
[6]     B←L/B
[7]     A←L/+  A[;2]
[8]     A←A–0, 1  A
[9]     Z←B,[1.5]  A
    ∇
      ∇ CHARS[0]∇
    ∇ Z CHARS  X;A
[1]   A RETURNS A LIST OF CHARACTERS IN (X),
[2]   A SORTED ACCORDING TO THEIR POSITION IN
        AV
[3]     A←'  ', CHAR
[4]     Z←A∊X
[5]     Z←Z/A
[6]     X←( (X  X)=  fX)/X  (  X∊A)/X
[7]     X←AV  X,Z
[8]     Z←AV[X[  X]]
    ∇
      ∇ DEB[0]∇
    ∇ Z DEB  X
[1]   A DELETE LEADING, TRAILING AN EXTRA
        1  0fX IN (X)
[2]     Z←1  0fX
[3]     →(1=ffX)fL1
[4]     X←,X
[5]     L1:Z  (–  Z←  1↑Z)↓X  (Zv0,  1↓Z  X≠Z)/X
    ∇
      ∇ DLBDTB[0]∇
    ∇ Z←DLBDTB X
[1]   A REMOVES LEADING AND TRAILING BLANKS
        FROM TEXT INPUT (X)
[2]     Z←( (v  Z)  Φ
    ∇
      ∇ COMPRESSNAME[0]∇
    ∇ Z←COMPRESSNAME NAME;L
[1]   A AN ALGORITHM FOR NAME COMPRESSION
[2]   A [1] DELETE SECOND ELEMENT OF REPEATED
        CONSONANT PAIR
[3]   A [2] DELETE 'AEIOUY' EXCEPT WHEN FIRST
        LETTER IN NAME
[4]   A 1ST STATEMENT IS NOT ABSOLUTELY
        REQUIRED
[5]   A
[6]     →(fNAME←DEB NAME)↓fZ  ''
[7]     Z←nL  NAME  'AEIOUY'
[8]     L←(0,1  L)  NAME=1↓NAME,'  '
[9]     Z←L/NAME
```

Fig. 2. Another example of what APL looks like, this printout from a DEC dot matrix terminal. The resolution is not quite as good, nor the resulting characters so exotic looking, but the terminal is faster and quieter.

ent with the normal use of functions such as sin, log, d/dx etc.

The APL Character Set

In order to write all functions compactly, a large collection of new symbols was designed, which is now fossilized in the type elements of various impact type terminals. These usually have all the upper case letters, plus numbers and punctuation and so forth, and of course quite a selection of not normally found symbols. Even more symbols are made by backspacing and over striking a second symbol on the first. (eg A, and ÷ to make ÷ etc).

Before non-APL fans get turned off by all this new symbology, it must be said that learning these symbols is extraordinarily easy, as many of them remind you of their meaning by their shape.

Two important notes to make: the use of the proper multiply sign (rather than an asterisk for multiply as in many languages) and instead of =. This is a logical choice since A A+1 really means 'A is assigned the value A+1', rather than 0=1. (By the way, APLers often say 'gets' rather than 'is assigned the value').

In Fig. 1 is a listing of APL symbols and their meanings, while Fig. 2 shows what a typical output looks like and there are more annotated examples in Fig. 3.

Environment

The implementation of Iverson Notation as a computer language brought real muscle to the elegant notation. As a computer language it is used interactively, that is to say you sit at a terminal (or at your personal APL machine!) and communicate directly and immediately with the computer.

You have the option of typing a statement and obtaining its result on the spot (immediate execution mode), writing a function or program (function definition mode), and executing such a function or program (again – immediate execution mode). Various editing and line numbering facilities are provided for function writing, and diagnostics and error messages help to debug functions.

The Workspace

On 'big systems', each APL user has a 'workspace', a fenced-off piece of memory, as it were, of some arbitrary size, say 50k bytes, where all his work is stored, all variables functions and programs, and in which the programs are executed. On completion of a session the user can store his workspace, even including parially executed programs, as complete state information is contained in the workspace. This storing process is usually onto disk, and is done in a relatively fool proof way as the user signs off. The workspace then sits in suspended animation, awaiting the user's continued efforts next time.

A user may also copy functions from another workspace into his own, (if so permitted) which of course gives access to a lot of useful software which others on the same system have developed.

Other Big System APL Features

Just as a user may copy information from another workspace, so can he access system 'libraries' where large volumes of software are typically found, such as routines for plotting, various engineering, statistical, and work processing packages, and of course games.

Additionally, extensive file systems (usually disk) are generally available for storing large amounts of program code or data.

APL Characters On Existing Machines

How do you put APL on a machine which has no APL character set? While new machines will be able to display the characters, and new character generator ROMs may satisfy a few other ways around the problem are possible.

One is to use three to five letter keywords instead of symbols, such that a single key could signify the function, and the whole keyword pops onto the screen. This doesn't provide an elegant looking display, but it does give you APL with only a small sacrifice.

Implementation On Micros

The workspace, library and file concepts are more system dependent than the Iverson Notation part of APL. They are also likely to be seen only in limited form at first on small systems. For instance, one workspace is likely to be almost the entire memory of a micro-based machine, so there will be only one user at a time. A disk unit will be essential, for storing workspace(s) and files, and one will presumably be able to create one's own libraries, or buy library disks with various useful packages on them.

'Virtual Workspace' schemes are already being worked on, wherein the user thinks he has a large amount of memory available, when in fact the machine has only a small amount, but swaps portions of the workspace on and off a disk as they are needed. This requires some fancy memory management to accomplish effectively.

```
      ∇ ENTRY
[1]     B 10 '
[2]     A←A,B
[3]     →1↑B×'*'
      ∇
```

ENTRY allows the user to enter a list of names, one at a time, alloting ten characters to each name, and filling any unused positions with blanks. ENTRY stops if the user types a single "*". The resulting vector is A.

```
      ∇ RESHAPE
[1]     C←(fA)÷10),10
[2]     D←CfA
      ∇
```

RESHAPE changes the list of names from a vector to an array with one name per row.

```
      ∇ ORDER
[1]     E←ALPHA D
[2]     L←1↑C
[3]     F←E, L
[4]     N←10
[5]     F←F[↑F[;N];]
[6]     N←N−1
[7]     →5+3XN=0
[8]     D←D[F[;11];]
      ∇
```

ORDER takes the array D, makes a numerical array E whose entries represent the letters in D, according to ALPHA. E is sorted by row, and the new order is then imposed on D. Job finished!

```
      ALPHA
 ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

ALPHA is the vector variable to which we assigned the desired organizing sequence. It's value is the character "blank" followed by the alphabet. Other orders or characters could have been chosen instead.

```
      ENTRY

JOHN
FRED
ANDY
*
```

User types "ENTRY", the function waits for the users entries, until it encounters the "*".

```
      A
JOHN          FRED          ANDY
      B
*
```

The resulting A and B values after ENTRY.

```
      RESHAPE
      C                        D
   10                     JOHN
                          FRED
                          ANDY
                          *
```

RESHAPE, and the resulting C and D values.

```
      ORDER

      E                                    F
11 16  9 15  1  1  1  1  1  1      2 15  5 26  1  1  1  1  1  1  3
 7 19  6  5  1  1  1  1  1  1      7 19  6  5  1  1  1  1  1  1  2
 2 15  5 26  1  1  1  1  1  1     11 16  9 15  1  1  1  1  1  1  1
28  1  1  1  1  1  1  1  1  1     28  1  1  1  1  1  1  1  1  1  4

      L

      D
ANDY
FRED
JOHN
*
```

User executes the function ORDER, and the resulting values of E. L. F, and finally what we were waiting for . . . D, now in order.

Fig. 3. An example of how to sort a list of names. The 3 functions are listed at the top. Below is a series showing what the user did (indented), and what the machine's response was (Yes, we know it could be done in one line or less).

```
      ∇ EXAMPLE
[1]     A←' '
[2]     ENTRY
[3]     RESHAPE
[4]     ORDER
[5]     ⎕←D
      ∇
```

EXAMPLE is a function which ties these three functions together into one, and then prints the result out.

Where has APL Been?

Recent Micro Moves

Until recently there hasn't been much stirring in the way of APL for the microcomputer system, in the way BASIC has been available for every microcomputer around.

Now, suddently there appears to be a terrific battle brewing, as the race is on to feed APL to what is expected to be an open-mouthed market. Judging by the almost astounding enthusiasm of current APL users, each new user will be spreading the word far more quickly than was true with BASIC, that is if he can tear himself away from the console!

Videobrain

First out with APL appears to be umtech with their F8 based Video Brain. Fitting into 13k – worth of ROM-in-a-cartridge, APL/S is a subset of 'full APL', although there is no standard APL. While a standard for APL is being worked on, big machine versions provide a reference which differs only slightly from one to another, and then only in housekeeping facilities rather than in the basic notation.

Z80 And Vanguard

Vanguard Systems Corp. have announced an APL interpreter on floppy disk for Z80, and of course all companies with Z80 based machines are eagerly waiting for it. It is reported to be 27K.

Microsoft

Meanwhile, over at Microsoft, much brain work is going into the development of APL interpreters for all kinds of processors, past present and future, ie 8080, Z80 and the 16 bit micros 8086, Z8000 and 68000. Bill Gates, president and APL Product Manager at Microsoft figures fall '79 will be the time his 8080 and Z80 interpreters will be ready with the 16 bit versions in early 1980. Vanguard are hoping to manage the same feat.

It's Coming!

Well, it's almost here. It appears that the development of APL on a 16 bit microprocessor will make the first really comfortable implementation of APL for personal, home, business or educational use. Because of this general purpose nature most new APL machines will be easy to use and hence either contain APL on ROM (fool proof) or on disk (most machines will have a disk or two anyhow).

Near Future

What is about to happen in the way of micro APL?

'APL will see an incredible increase in popularity because it will be exposed to so many new people. To date, it has been an expensive language to use and is almost never introduced to first time computer users. APL's strengths assure that a significant percentage of personal computer users will adopt it as "the language". However it is not the ideal first time language and has some limitations of its own. BASIC will continue to dominate in this role, although specialized languages will be supported by personal computer manufacturers. Microsoft will introduce APL on the TRS-80, Exidy Sorcerer, Interact One, Nascom and NEC TK-80 in 1979.

Also the old time APL users will enjoy the low cost and ease of access to the low priced machines. These old time users will be key in generating new converts. New tutorial material making APL seem less mathematical and not forcing the user to see the utility of all the operators will be required. The statistical, accounting and model type will be key ones for the new small business computers. With APL's following, IBM's support of it and its strength, our OEM's have decided they can't ignore it. Our response on the product to date has been incredible.

First Version

Details of the 8080/Z80 version: 'Our 8080/Z80 version (long awaited) will be out in April running under CP/M. Other versions such as the virtual workspace one will follow . . . Equivalence with the 5100 was my goal – and I met that with 24K of code.

We will license our APL to most of our existing OEM's.'

Performance

What kind of performance is expected from micro APL?

'Microsoft APL has all the features of IBM APL/SV (360/370) APL except that the I/O and shared variables are handled differently and the math will not be totally the same since we use binary exponents where IBM uses hex. Our

APL is twice as fast as the 5100 which is very adequate and often better than remote APL timesharing systems. Our 8086 and Z-8000 APL's will be at least five times faster and will be available in early 1980. This APL will outperform APL on most minis.

The string search and block transfer are an advantage that our Z-80 version will use. In specific cases this could cause a 2:1 speed difference, but overall the Z-80 to 8080 difference will be less than 15% at equal clock speeds. More important are the design techniques which allow a 32K byte work space to be equivalent to almost twice that much workspace on the 5100 through more efficient management and special representations.'

Far Future

Looking forward about a year Gates sees some of the following features being offered:

Workspaces up to 40K with virtual workspace to follow.
Libraries on disk.
Unlimited number of dimensions for arrays.
16 digit computation (floating point)
Availability of all primitives normally found.
Sophisticated system commands, error messages.
Initially only SAVE and LOAD workspace 'file' operations, but later random access I/O on arbitrary objects.

Later on Gates thinks 16 bit APL will have up to 10 meg workspaces, same library facilities as I.P. Sharp. Significantly, he comments that many language extensions may be persued, something he feels IBM has stifled. 'APL should not be allowed to stagnate. It is very weak in control structures. General lists should be added.'

I.P. Sharp

An interesting story came to our attention recently in connection with micro APL. I.P. Sharp Associates, known worldwide for their APL communications network and software expertise, were apparently working on a micro based APL consumer product! Sharp's Steve Kohalmi described the project. A major Canadian manufacturer of entertainment products went to Sharp with the idea, and developed the hardware prototype while Sharp worked on the software. The produce was based on a GI 1600 microprocessor, and was to have floppy disk(s) and some colour graphics capability. The APL interpreter, which was developed on a PDP11/34 simulating GI code, has dynamic workspace management, floating point numbers, but only one dimensional arrays. 16K of 16 bit words was required. Unfortunately the whole combination was never tested as interest in further development has appeared to lapse. Anybody want to take over an almost debugged GI 1600 APL? Call Sharp.

Acknowledgements

Our thanks to the following people who were very helpful in the preparation of this article:

Bill Gates, Microsoft;
Chuck Williams, John Koiste, MCM Computers;
Steve Kohalmi, I.P. Sharp Associates;
I.P. Sharp Newsletter

Suggested Books

Kenneth E. Iverson: "Elementary Analysis"; APL Press, 1976.
The APL Press has available a comprehensive collection of publications on APL, the interested reader should write for a complete list.
Leonard Gilman and Allen J. Rose: "APL An Interactive Approach"; John Wiley and Sons Inc, 1974.
A.D. Falkoff and K.E. Iverson, "APL Language", IBM Corp, 1975 (Form No. GC26-3847)

Ian L. Powell.

A Tiny BASIC Star Trek program that will have you zapping Klingons till Stardate 29·35

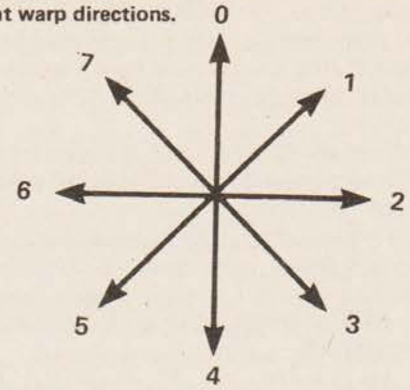
The main object of this Startrek game is to destroy all the Klingon battle cruisers in the Galaxy, The Klingons, Stars and Starbases are randomly positioned within a simulated Galaxy which is divided into a matrix of 8 by 8 sectors, making 64 in all. The Federation starship Enterprise, of which you are in command, has been assigned the task of seeking out and destroying all the Klingon invaders. You have at your disposal an impressive arsenal of weapons, including phasers and photon torpedos which are under your direct control. The Enterprise may be replenished with energy and weapons by docking at a Starbase.

Commands

1) The HELM – This allows you to move the Enterprise through space to any point in your present sector, or to any other sector in the Galaxy. To do this you have to give a warp factor and a course which can be in any one of eight directions. At the end of each manoeuvre you are automatically given a short range scan.
   (a) COURSE (0–7), see Figure. 1
   (b) WARP (1–63), gives you control over the number of units travelled, each unit being 1/8 of a sector.

NOTES (i) To move the Enterprise into another sector you merely give the Helm sufficient warp factor to get there, (each sector is 8 units wide).
(ii) You may not travel through Stars, Klingons or Starbases.
(iii) To dock at a Starbase you manoeuvre into an adjacent position to the left or right of the Starbase.

Fig.1 The eight warp directions.



2) LONG RANGE SCAN – This gives you a summary of the objects in your present sector and adjacent ones. The information is displayed as a three figured number with the hundreds indicating the number of Klingons, the tens the number of Starbases and the units the number of Stars, as shown in Fig. 2.

H T U

KLINGONS STARBASES STARS

Fig.2 The long range scan display.

3) PHASERS – This is an energy weapon, the affect of which diminishes with distance. You are informed of the amount of energy left in the main banks and asked to enter the amount of energy to be diverted to the phaser weapon. To destroy a Klingon you have to deplete his energy to zero. On entering a sector, each Klingon has a full energy bank of 200 units. To calculate the effect of the phasers, the energy is divided equally between the Klingons in the current sector and is further reduced by dividing by the distance between the Klingon and the Enterprise. The result of this calculation is subtracted from the Klingon's energy bank.

Any Klingons left in the sector, after the operation of the weapon, shoot back. The hits on the Enterprise are calculated from the amount left in each Klingon's energy bank, divided by the distance between the Enterprise and the Klingon. However, if you are docked at a Starbase you are protected by its shields.

4) PHOTON TORPEDOS – These are a torpedo like weapon where a single bolt of intense energy is propagated in a single direction. Due to the high energy content of the bolt a Klingon or Starbase is destroyed by a single shot. Stars, however, will absorb the energy. The bolts have an unlimited range within the current sector and are controlled only by direction. The direction command is the same format as that used in the HELM control: COURSE (0–7).

NOTES (i) You cannot shoot through Stars.
(ii) If you destroy a Starbase you are relieved of duty.
(iii) Any Klingons left after your attack will shoot back, using the same procedure as with the phaser weapon.

5) SHIELDS – The shields protect the Enterprise from enemy weaponry. Every hit on the Enterprise depletes the shields by the amount equal to the hit. To protect the Enterprise there must be enough energy in the shields to neutralise any attack or it will be destroyed.

After requesting the command you are given the total energy available, (the sum of the main banks plus that in the shields). You are then asked to enter the amount of energy to be diverted to the shields.

NOTES (i) If the command reduces the energy in the shields, any surplus is diverted to the main banks.
(ii) When docked at a Starbase you are protected by the Starbase shields.
(iii) As well as being attacked when you shoot at the Klingons, you may also be attacked on a random basis, if the condition is red and you are executing the HELM, SHORT RANGE SCAN or LONG RANGE SCAN commands.

6) SHORT RANGE SCAN – This gives you the positions of the Stars, Klingons, Starbases and the Enterprise in your present sector. You are also given the stardate, the status of the main energy banks, your condition, (GREEN, RED, DOCKED), the number of photon torpedos left, the shield energy and the total number of Klingons left in the Galaxy.

7) RESIGNATION . . .

Video Display Characters

The objects in the Galaxy are displayed as follows:–

Object	Displayed Character
Enterprise	–0–
Klingon	>!<
Star	*
Starbase	<0>

Comments on Program Conversion for other Computers

1) The IF statement operates on the rest of the line and can be used to control multiple statements.

2) The N in the PRINT statements control the number of digits printed.

3) The RND(N) function generates a random number between 1 and N.

4) The TRITON variables are two byte integers which is the minimum length for the program's operation.

Program Notes

Line Nos.	
20 – 90	Initialisation of variables.
100 – 150	Initialisation of sectors. Each sector is represented as a single number – the hundreds representing the number of Klingons, the tens the number of Starbases and the units the number of Stars.
160	Choose the initial current sector.
170	Choose the initial position of the Enterprise in the current sector.
180 – 190	Joins up the ends of the Galaxy.
200 – 330	Sets up the positions of the Stars, Starbases and Klingons in the current sector.
200 – 220	Decodes the sector number.
230	Zeros sector and loads Enterprise position.
245 – 260	Sets up the Stars.
270 – 290	Sets up the Starbases.
300 – 320	Sets up the Klingons.
360 – 370	Tests for Enterprise condition – GREEN, RED, DOCKED.
380	Reloads weapons and energy banks when condition DOCKED.
400 – 540	Short Range scan outputs the position of the Stars, Starbases, Klingons and the Enterprise; also gives information on the stardate, energy banks, shields, torpedos, total number of Klingons in the Galaxy, etc.
550	Random test for Klingon to fire on Enterprise after a command.
600	Test for all Klingons destroyed.
605 – 620	Select command (1–7)
1000 – 1140	HELM control (1)
1000 – 1035	Read warp factor.
1045	Blank Enterprise position in the current sector matrix: decrement energy and increment date.
1090 – 1130	Move Enterprise W units.
1135	Calculate new sector if Enterprise leaves the current sector.
1140	Put new Enterprise position into the current sector matrix.
2000 – 2040	LONG RANGE SCANE (2) Prints out the sector numbers of the current sector and the surrounding ones.
3000 – 3120	PHASER control (3)
3000 – 3010	Prints energy reserve and reads phaser energy.
3020	Decrement energy banks, and test for no Klingons in current sector.
3030	Divide phaser energy by the number of Klingons in the current sector and calculate the Enterprise position vector.
3040 – 3050	Search current sector matrix for Klingons.
3060 – 3070	Calculate distance between Enterprise and Klingons: Decrement Klingon energy bank.
3080	Test for Klingon destroyed and take appropriate action.
3090	Klingon not destroyed so store new Klingon energy level: Calculate hit on Enterprise and subtract from shields.
3110	Test for destruction of the Enterprise.
3120	Updates the number of Klingons in the current sector.
4000 – 4080	PHOTON TORPEDOS control (4)
4000	Test for no torpedos
4010	Read course: decrement number of torpedos.
4010 – 4050	Step along course until object encountered.
4060	Test for Starbase hit.
4065	Test for object not a Klingon.
4070	Klingon destroyed.
4080	Set phaser energy to zero and jump to phaser routine to calculate the hits on the Enterprise shields from the remaining Klingons.
5000 – 5030	SHIELD control (5)
5000	Add shield power to that in the main energy banks: Prints total energy.
5010	Read shield energy and subtract it from the main energy banks.
5020	Jump for next command.
6000	SHORT RANGE SCAN (6) Jump to short range scan routine.
7000	RESIGNATION (7)
8000 – 8030	Subroutine to calculate distance between the Enterprise and Klingons.
8200	Klingon destroyed subroutine.

8300 Print energy subroutine.
8500 – 8570 Subroutine to read course and calculate the direction increment vector.
8600 Game lost: Stop.

NOTE

The subscripted characters in the PRINT statements used to display the Enterprise, etc., (line nos. 440 – 460) indicate the character is to be typed in conjunction with the control or shift key.

Fig.3 The program listing.

```
10 REM STARTREK SIMULATION PROGRAM
20 REM A TINY BASIC VERSION
30 REM I.L.POWELL 10/3/1979
40 K=0
50 T=RND(200)+200
60 E=3000
70 O=15
80 S=0
90 L=T
95 REM SET UP SECTORS
100 FOR I=0 TO 63
103 X=0
105 Y=0
110 IF RND(10)<8 X=RND(3)
120 IF RND(100)>88 Y=1
130 Z=RND(5)
140 @(I)=X*100+Y*10+Z
145 K=K+X
150 NEXT I
160 Q=RND(64)-1
170 A=RND(64)-1
180 IF Q<0 Q=Q+64
190 IF Q>63 Q=Q-64
199 REM SET UP CURRENT SECTOR
200 Z=@(Q)
210 X=Z/100
215 Z=Z-X*100
220 Y=Z/10
225 Z=Z-Y*10
230 FOR I=64 TO 127
233 @(I)=0
235 NEXT I
237 @(A+64)=4
240 FOR I=1 TO 5
245 IF I>Z GOTO 270
250 J=RND(64)+63
255 IF @(J)#0 GOTO 250
260 @(J)=3
270 IF I>Y GOTO 300
280 J=RND(64)+63
285 IF @(J)#0 GOTO 280
290 @(J)=2
300 IF I>X GOTO 330
310 J=RND(64)+63
315 IF @(J)#0 GOTO 310
320 @(J)=-200
330 NEXT I
340 REM SHORT RANGE SCAN
350 C=1
355 IF X#0 C=2
370 IF (@(A+63)=2)+(@(A+65)=2) C=3
380 IF C=3 E=3000; S=0; O=15
400 FOR I=0 TO 56 STEP 8
410 FOR J=64 TO 71
420 U=@(I+J)
430 IF U=0 PRINT '   ',
440 IF U<0 PRINT '>!<',
450 IF U=2 PRINT '<O>',
455 IF U=3 PRINT ' * ',
460 IF U=4 PRINT '-O-',
470 NEXT J
480 IF I=8 PRINT 'STARDATE',T
485 IF (I=0)+(I=56) PRINT
490 IF I#16 GOTO 500
492 PRINT 'CONDITION',
493 IF C=1 PRINT 'GREEN'
494 IF C=2 PRINT 'RED'
496 IF C=3 PRINT 'DOCKED'
500 IF I=24 PRINT 'ENERGY',E
510 IF I=32 PRINT 'TORPEDOS',O
520 IF I=40 PRINT 'SHIELDS',S
530 IF I=48 PRINT 'KLINGONS',K
540 NEXT I
550 IF L#T IF RND(9)<6 GOTO 4080
590 REM COMMAND CONTROL
600 IF K=0 PRINT 'THE FEDERATION
    HAS BEEN SAVED' ; STOP
605 INPUT 'COMMAND' B
610 IF (B>7)+(B<1) GOTO 600
620 GOTO B*1000
999 REM HELM CONTROL
1000 GOSUB 8500
1020 INPUT 'WARP(1-63)'W
1030 IF (W<1)+(W 69) GOTO 1020
1035 IF E W GOSUB 8300 ; GOTO 600
1040 @(A+64)=0
1045 E=E-W
1050 T=T+W
1090 FOR I=1 TO W
1110 IF (V<0)+(V>7)+(H<0)+(H>7)
     GOTO 1120
1113 U=0
1115 IF @(V+8*H+64)#0 V=V-N ;
     H=H-M ; GOTO 1140
1120 NEXT I
1125 IF V<0 V=V-8
1130 IF H<0 H=H-8
1135 IF U Q=Q+8*(H/8)+V/8 ;
     GOTO 170
1140 A=V+8*H
1150 @(A+64)=4
```

```
1160 GOTO 350
1999 REM LONG RANGE SCAN
2000 FOR I=-8 TO 8 STEP 8
2010 FOR J=-1 TO 1
2015 U=Q+J+I
2020 IF U<0 U=U+64
2025 IF U>63 U=U-64
2030 PRINT#4,@(U),
2033 NEXT J
2035 PRINT
2037 NEXT I
2040 GOTO 550
2999 REM PHASER CONTROL
3000 GOSUB 8300
3005 INPUT 'PHASERS' P
3010 IF E<P GOTO 3000
3020 E=E-P
3025 IF X=0 GOTO 3120
3030 P=P/X
3033 H=A/8
3035 V=A-8*H
3040 FOR I=0 TO 63
3050 J=I+64
3055 IF @(J)>=0 GOTO 3110
3060 F=I/8
3065 U=I-F*8
3070 GOSUB 8000
3073 G=@(J)
3075 G=G+P/D
3080 IF G>=0 GOSUB 8200
3085 GOTO 3110
3090 @(J)=G
3100 IF C#3 G=-G/D ; S=S-G ;
     PRINT#3,G,'HIT ON SHIELDS'
3110 NEXT I
3115 IF S<0 GOTO 8100
3120 @(Q)=(Q)-(@(Q)/100-X)*100
3130 GOTO 600
3999 REM PHOTON TORPEDO CONTROL
4000 IF O<1 PRINT 'NO TORPEDOS' ;
     GOTO 600
4010 GOSUB 8500
4013 O=O-1
4015 FOR I=1 TO 16
4020 V=V+N
4025 H=H+M
4030 IF (V<0)+(V>7)+(H<0)+(H>7)
     GOTO 4080
4040 J=V+8*H+64
4045 IF @(J)#0 GOTO 4060
4050 NEXT I
4060 IF @(J)=2 PRINT 'STAR BASE
     DESTROYED' ; GOTO 8600
4065 IF @(J)>0 GOTO 4080
4070 GOSUB 8200
4080 P=0
4090 GOTO 3020
5000 E=E+S
5005 GOSUB 8300
5010 INPUT 'SHIELDS' S
5015 E=E-S
5020 IF E<1 PRINT 'S>E' ;
     GOTO 5000
5030 GOTO 600
5999 REM JUMP TO SHORT
     RANGE SCAN
6000 GOTO 350
6999 REM RESIGNATION
7000 PRINT 'ACCEPTED'
7010 GOTO 8600
7900 REM SUBROUTINES
7910 REM CALCULATE KLINGON
     ENTERPRISE DISTANCE
8000 Z=H-F
8005 Y=V-U
8010 FOR D=1 TO 8
8015 G=D+1
8020 IF G*G (Z*Z+Y*Y) RETURN
8030 NEXT D
8040 RETURN
8090 REM
8100 PRINT 'ENTERPRISE DEAD
     IN SPACE'
8110 GOTO 8600
8190 REM
8200 PRINT 'KLINGON DESTROYED'
8210 X=X-1
8220 K=K-1
8230 @(J)=0
8240 RETURN
8290 REM
8300 PRINT 'ENERGY=',#5,E
8310 RETURN
8390 REM READ COURSE AND
     CALCULATE MOVEMENT VECTOR
8500 INPUT (COURSE(0-7)' B
8510 IF (B<0)+(B>7) GOTO 8500
8520 M=0
8523 N=0
8525 H=A/8
8527 V=A-H*8
8530 IF (B<2)+(B>6) M=-1
8540 IF B>2 IF B<6 M=1
8550 IF B<4 IF B>0 N=1
8560 IF B>4 N=-1
8570 RETURN
8590 REM
8600 PRINT 'YOU ARE RELIEVED
     OF DUTY'
8610 STOP
```

Ian Sinclair.

MPU's BY EXPERIMENT

In part four of our series Ian Sinclair takes a look at some of the SC/MP's logical instructions

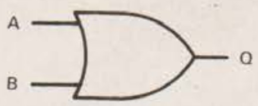
Your breadboard microprocessor is now in just about its final form, and we can spend some time looking at some more of the instructions which the 8060 can carry out. In Part 3, we ended up trying out the AND IMMEDIATE instruction, using the STOre sequence of 11001000, followed by 00000001 to display the final number in the accumulator. In this part, we'll kick off by looking at a few more of these IMMEDIATE instructions, starting with OR IMMEDIATE.

Immediately Logical

Like the AND IMMEDIATE, the OR instruction takes the byte which is stored in the acummulator and compares it bit by bit with an IMMEDIATE-loaded number, which is the byte following the OR instruction. Each bit in the accumulator is OR'd with the corresponding bit of the new number, obeying the rules of the two-input OR-gate as shown in Fig. 1. To demonstrate this action, go through the following sequence.

First of all, RESET. This is particularly important if you have just switched on, becuase the 8060, like all microprocessors, will start up with its register flip-flops in random order. All sorts of things can go wrong if you try to use it without clearing the decks first. Now CANCEL RESET, and go through the sequence of 11000100 PUSH 10101010 PUSH. This is LOAD IMMEDIATE followed by the data byte 10101010, an easily recognisable pattern, so that the accumulator is now loaded with this byte. Follow this with 11011100 PUSH 01010101 PUSH. The first byte of this pair is the OR-IMMEDIATE instruction, and the second is the data byte which is to be OR'd. Fig. 2 shows what the result should be – a complete byte of 1's.

This result should now be sorted in the accumulator, and to display the bits on the LEDs we use the sequence (which is going to become rather familiar) of 11001000 PUSH 00000001 PUSH. The address number will now be 0111 (we're coming up to the 7th push) and the DLEDs will show the result, 11111111 as it ought to be.

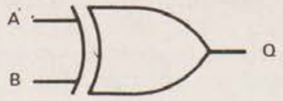


A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

Fig.1 The 2-bit OR-gate and its truth table.

```
    1 0 1 0 1 0 1 0
OR  0 1 0 1 0 1 0 1
  = 1 1 1 1 1 1 1 1
```

Fig.2 OR'ing two bytes together.



A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

Fig.3 The 2-bit exclusive-OR (XOR) gate and its truth table.

```
     1 0 1 0 1 0 1 0
XOR  1 1 1 1 1 1 1 1
   = 0 1 0 1 0 1 0 1
```

Fig.4 XOR'ing two bytes together.

Exclusive Instructions

The EXCLUSIVE-OR IMMEDIATE instruction uses the same sort of sequence, but the action is rather different, as Fig. 3 shows. The truth table for the 2-input X-OR gate gives a 1 output only for a single 1 input, and a zero when both inputs are 0 or both inputs are 1. As usual, the 8060 carries out the X-OR on eight bits at a time, using the sequence starting with RESET, CANCEL RESET, 11000100 PUSH 10101010 PUSH. This loads the accumulator with the data byte 10101010 as before. The next steps are 11100100 PUSH 11111111 PUSH. The first byte of this pair is the XRI (X-OR IMMEDIATE), and the second is the data byte. The X-OR of these two should give the byte 01010101, and this can be viewed by using the sequence 11001000 PUSH 00000001 PUSH as before. See Fig. 4. This XOR instruction is very often used to check that a byte which is stored in the accumulator is identical to some other byte fetched from memory – if the two are identical, the accumulator contents will end up as zero, and a JUMP-IF-ZERO instruction can then guide the microprocessor to a new set of instructions.

Adding A Bit More

That disposes of the main logic instructions for the moment. Let's look now at the 8060 in action on some arithmetic – simple binary addition. From now on, also, we'll start to use the shortened forms of the instructions, or memonics, when we refer to the instructions in words. There aren't any short forms for the binary codes, though, until we take the step to hexadecimal programming later.

Binary addition follows simple laws which are summarised in the truth table of Fig. 5. As in decimal arithmetic, a 'carry' is added in to the next most significant pair of digits, that is the next to the left when we write the number in the usual way. The simplest of these ADD instructions in the 8060 set is the ADD BINARY IMMEDIATE, ADI, and we can now try out an elementary binary addition, starting in the usual way with RESET, CANCEL RESET. Now use

11000100 PUSH 00001111 PUSH so as to load the accumulator with the data byte 00001111. This is followed by 11110100 PUSH 00010101 PUSH, which is the ADI code followed by the byte which is to be added. As Fig. 6 shows, this should give the result 00100100, and this can be checked by the usual display sequence 11001000 PUSH 00000001 PUSH.

Now this instruction obviously carries out the binary addition in this example, but what happens if there is a carry out of the most significant bit? We can try it right now — prepare the chip in the usual way, and this time load in a large number by setting up the sequence: 11000100 PUSH 11111111 PUSH. Now follow with ADI, which is 11110100 PUSH, and another data byte 01111111 PUSH. Fig. 7 shows the result of this addition. To display, use 11001000 PUSH 00000001 PUSH. Since there are only eight DLEDs, we can see only eight bits of answer — so what has happened to the final carry bit?

The Question Answered

The answer appears if we follow this up with another piece of binary arithmetic without resetting. We shall add 1 and 1, which ought to give 00000010, binary 2, by using the sequence 11000100 (LDI) PUSH 00000001 (data) PUSH 11110100 (ADI) PUSH 00000001 (data) PUSH 11001000 PUSH 00000001 PUSH. The display might be expected to show 00000010, but it actually shows 00000011 — which is binary 3. The extra 1 which has been added in has been the 1 carried over from the previous addition. The question now is — where was this carry bit stored between times? The answer is that a flip-flop, one of a group of eight called, slightly misleadingly, the status register, was used to hold the carry bit, and to add it into the next sum. On the 8060, this bit is called the CARRY/LINK (CY/L) and is bit 7 (counting from 0, remember) of the status register. I've said that it's a trifle misleading to call this set of 'status' flip-flops a register; that statement needs an explanation. Usually a register contains a byte which represents a binary number, part of a binary number or a coded character (a letter or graphic symbol). The status set of flip-flops doesn't represent any byte; each flip-flop of the set is quite independant of the rest, so that each bit is set by actions which need not affect any of the others. On the other hand, the whole set of bits can be exchanged with the byte in the accumulator, so that the bits of the status register can all be changed or tested simultaneously if we want to do so. Because of this action, the name of register is justified.

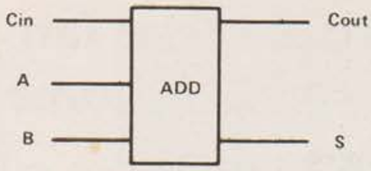
The CY/L bit is particularly useful when large binary numbers have to be added. This is done by breaking the numbers down into groups of eight bits, and then adding corresponding bytes, as illustrated in Fig. 8. If we start, as would be normal, with the lowest order byte, then the CARRY/LINK ensures that a carry bit is correctly added to each successive byte as required, but without needing any special program instructions.

Reveal Your Status

While we're on the subject of the status register, we might as well take a look at this CY/L bit by teasing it out of its lair. RESET, CANCEL RESET, and do the addition of 11111111 to 01111111 again, but without the final two display steps. Now set up 00000110 PUSH. This is the COPY STATUS TO ACCUMULATOR (CSA) instruction, which, as the name suggests, loads the accumulator with the same pattern of bits as are contained in the flip-flops of the status register, without changing the status register in any way (it's a *copy* instruction, not an *exchange* instruction). We can now display this byte by the usual sequence of 11001000 PUSH 00000001 PUSH. This causes DLED 7 to glow, indicating that bit 7 (the eighth, remember, because we start at 0) of the status register was set by the addition operation.

This CY/L bit can be set or reset by program operations which do not involve addition. The instruction 00000010 clears the CY/L bit — set up this code on the data switches, PUSH, and then go through the sequence of 00000110 (CSA) PUSH 11001000 PUSH 00000001 PUSH to display the status register. Similarly, the instruction 00000011 PUSH sets the CY/L bit to 1, you can check this one out for yourself using the method shown above, but substituting the set instruction for the clear instruction. The whole of the status register can also be changed by copying a byte from the accumulator (instruction CAS) to the status register.

Sometimes the automatic operation of the CY/L bit is a nuisance; the example we used showed how the second



Cin -- CARRY IN FROM PREVIOUS ADDITION
Cout – CARRY OUT TO NEXT ADDITION
S – SUM

Cin	A	B	S	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Fig.5 The ADD function and its truth table.

```
      0 0 0 0 1 1 1 1
ADI   0 0 0 1 0 1 0 1
    = 0 0 1 0 0 1 0 0
```

Fig.6 ADD IMMEDIATE operations on two bytes.

```
       1 1 1 1 1 1 1 1
ADI    0 1 1 1 1 1 1 1
1 <--  0 1 1 1 1 1 1 0
carry
```

Fig.7 A carry out of the byte.

```
      1 0 0 1 0 1 1 0 1 1 1 0 0 1 1 0
ADI   0 0 1 1 1 0 1 1 1 0 0 1 0 1 1 1
    = 1 1 0 1 0 0 1 0 0 1 1 1 1 1 0 1

HIGHER BYTE                     LOWER BYTE
1 0 0 1 0 1 1 0                 1 1 1 0 0 1 1 0
0 0 1 1 1 0 1 1   <-- 1         1 0 0 1 0 1 1 1
1 1 0 1 0 0 1 0      carry      0 .1 1 1 1 1 0 1
```

Fig.8 Double-byte arithmetic. Using the carry/link bit lets us carry out sixteen-bit arithmetic in two steps, adding the lower order bytes first.

```
                 0 1 1 1 1 1 1 1
                 0 1 1 1 1 1 1 1
NOT CARRY        1 1 1 1 1 1 1 0
FROM BYTE        ---------------
```

Fig.9 An addition with no carry out of the byte – but the overflow flag is set.

```
                                     INTERPRETATION
        AS CARRIED OUT               UNSIGNED   SIGNED
        8th 7th
        1 1 0 0 0 0 0 1                193        -63
        1 1 0 0 0 0 1 0              + 194        -62
   1 <- 1 0 0 0 0 0 1 1                387       -125
 carry                               (remember carry
        Carry from 7th and 8th        = 256)

        8th 7th
        1 0 0 0 0 0 0 1                129       -127
        1 0 0 0 0 0 1 0              + 130       -126
   1 <- 0 0 0 0 0 0 1 1                259       -253
 carry  ---------------                ---       ----
        Carry from 8th only          (remember   but answer does
                                      = 256)     not have 1 in 8th
                                                 place, so OV is set

        8th 7th
        0 1 0 0 0 0 0 1                 65        + 65
        0 1 0 0 0 0 1 0               + 66        + 66
        1 0 0 0 0 0 1 1                131       + 131
        Carry from 7th only                      but there is a 1 in
                                                 the 8th place, so
                                                 OV is set.
```

Fig.10 Using the OV bit to indicate when an answer may be wrongly read, when signed arithmetic is used. The binary addition is as carried out be the microprocessor, the decimal arithmetic is as we would interpret it.

of two additions could be affected by a carry from the first, which is why the CY/L clear instruction is available. This program instruction would always be used between separate additions to avoid unwanted carries, and is often used at the start of a piece of program in case the CY/L has not been previously cleared. At the start of a program or subroutine, the most convenient way of clearing the CY/L bit is to load the accumulator with zeros, and then copy the accumulator byte to the status register, using the CAS instruction, so that all the bits of the status register are set to 0.

Let's look at another status bit (or flag) which is affected by addition. The addition sum this time is as shown in Fig. 9, there's no carry out of the most significant bit from this one. RESET and CANCEL, then load and add by using the sequence 11000100 (LDI) PUSH 01111111 (data) PUSH 11110100 (ADI) PUSH 01111111 (data) PUSH. This should not have set the CY/L bit, and we can show this by copying the status register to the accumulator, using 00000110 (CSA) PUSH and displaying by the sequence 11001000 PUSH 00000001 PUSH.

Some Light On The Problem

There's certainly no light on DLED7, but DLED6 is lit! This indicates that another status flag has been set, and this one's called overflow (OV). Why should this be set? The reasons are a bit (sorry) complicated and will take some time to explain, but the mechanism is straightforward – the OV bit is set by the result of X-ORing the carries from bits 6 and 7 in an addition. If there's a carry from one of these places only, the OV bit is set. If there's a carry from both places, or no carry from either, the OV bit remains reset at 0. Unlike the CY/L bit, the OV bit is not added into subsequent additions, it's purely a warning.

A warning of what? Well, the OV bit comes into its own when the most significant bit of an 8-bit number is used to represent the sign of the number. The convention is that a 1 in the most significant place represents a negative number in 2's complement form, and 0 represents a positive number. The microprocessor itself treats all binary numbers in the same way, simply as numbers, but the OV bit is set to warn if a mistake could take place when the numbers are being interpreted as signed numbers, that is taking the most significant bit as a sign bit. It's not the microprocessor which is likely to make the mistake, but the user, which is why the OV bit is simply a warning.

Have a look at the examples in Fig. 10. Now if we regard these simply as eight bit binary numbers, with the most significant bit used as a digit, then all the binary additions are straightforward and valid. This is how the microprocessor will treat these additions. If we take the MSB as representing sign, however, things look rather different. Addition (a) is of –63 to –62, giving –125. A negative answer should have a MSB set to 1, which is the case in this example, and the carry out of the most significant bit can be ignored. Because this addition is valid *however we interpret it,* the OV bit is not set, though the CY/L is.

Addition (b) is of –126 to –127, and the result should be –253 if we're interpreting the sign bits correctly. It certainly works out correctly if we convert back, but the MSB is NOT 1, so that we would not convert this number in the normal run of events. For such an operation, the OV bit is set to warn us that the number is a negative number even though the MSB is zero. Addition (c) is of two positive numbers, but the answer has a 1 in the MSB position, indicating a negative sign. In this case also, the OV bit is set to 1.

Signing Off

The OV bit, then, warns that the sign of the MSB is wrong when we are interpreting the numbers as signed numbers, and the bit is set only when there is a carry out of D6 OR D7 but not both. It's then up to the programmer to make use of this OV bit to correct the sign if needed. In these additions, the microprocessor goes about its additions in the same way regardless of how we think of the numbers. As we'll see later, there are some instructions in which the microprocessor will take the MSB as indicating a sign.

There are no special instructions in the 8060 set for setting or resetting the OV bit apart from the others in the status register, because it's purely an advisory bit which does not affect any subsequent operation. It can be detected, set or reset only by the CSA and CAS instructions, along with all the other bits in the status register. All in all, it gives you some respect for the programs which are built into these pocket calculators which play with negative numbers as easily as with positive ones!

Paul M. Glover.

1 Dimensional Life for MK14

Many readers will be familiar with the Game of Life which is played by a computer on a 2 dimensional grid of cells. This has been adapted for the MK14 by arranging all the cells in a line. The field is 16 wide, an occupied cell is represented by a vertical bar and each digit of the display contains 2 cells.

The rules are as follows:–

i) If a cell is vacant, one is born if it has 2 or 3 neighbours.
ii) If a cell is occupied it dies if it has 1 or 3 neighbours.
iii) In 1.D. life each cell has 4 "adjacent" cells, i.e. 2 either side.

Mode Of Operation

The 16 cells have locations 0E80-0E8F. These are filled with the starting configuration, X'01 for occupied and X'00 for vacant.

GOing from 0FA0 causes this and subsequent generations to be displayed.

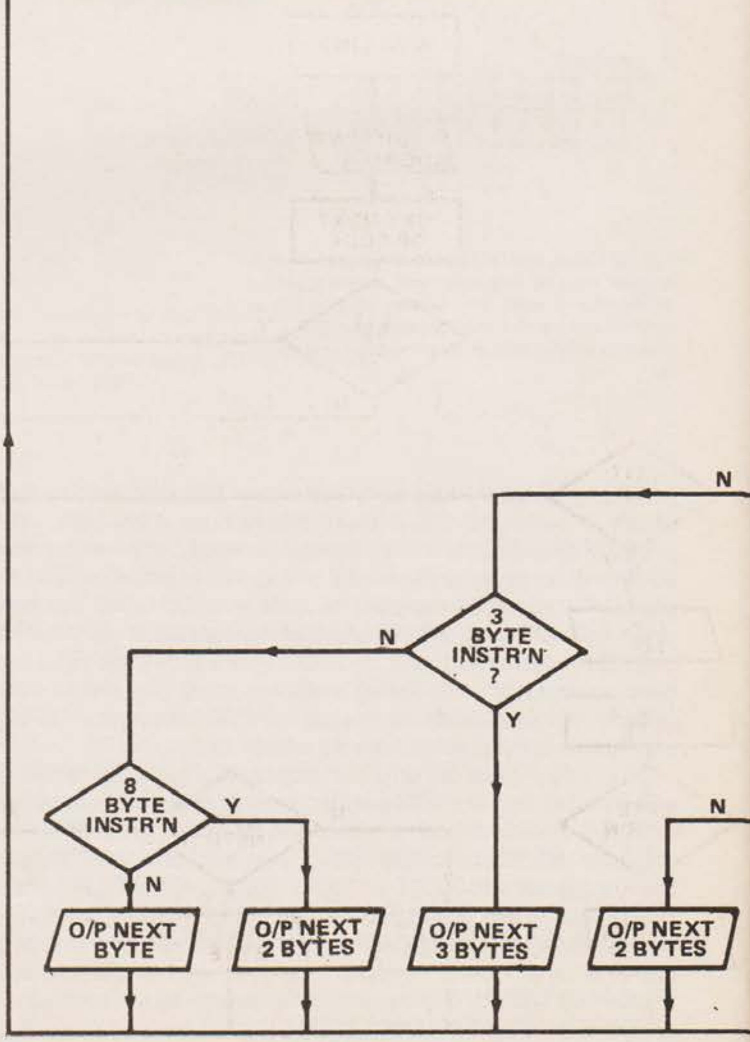
The display port of the program could be altered to give the status of the cells on 16 LED's connected to the output ports.

To improve the display; If address 0FC0 is changed to C4 63 and 0FCB to DC 5C then the cells become small squares.

Program listing for 1 Dimensional Life

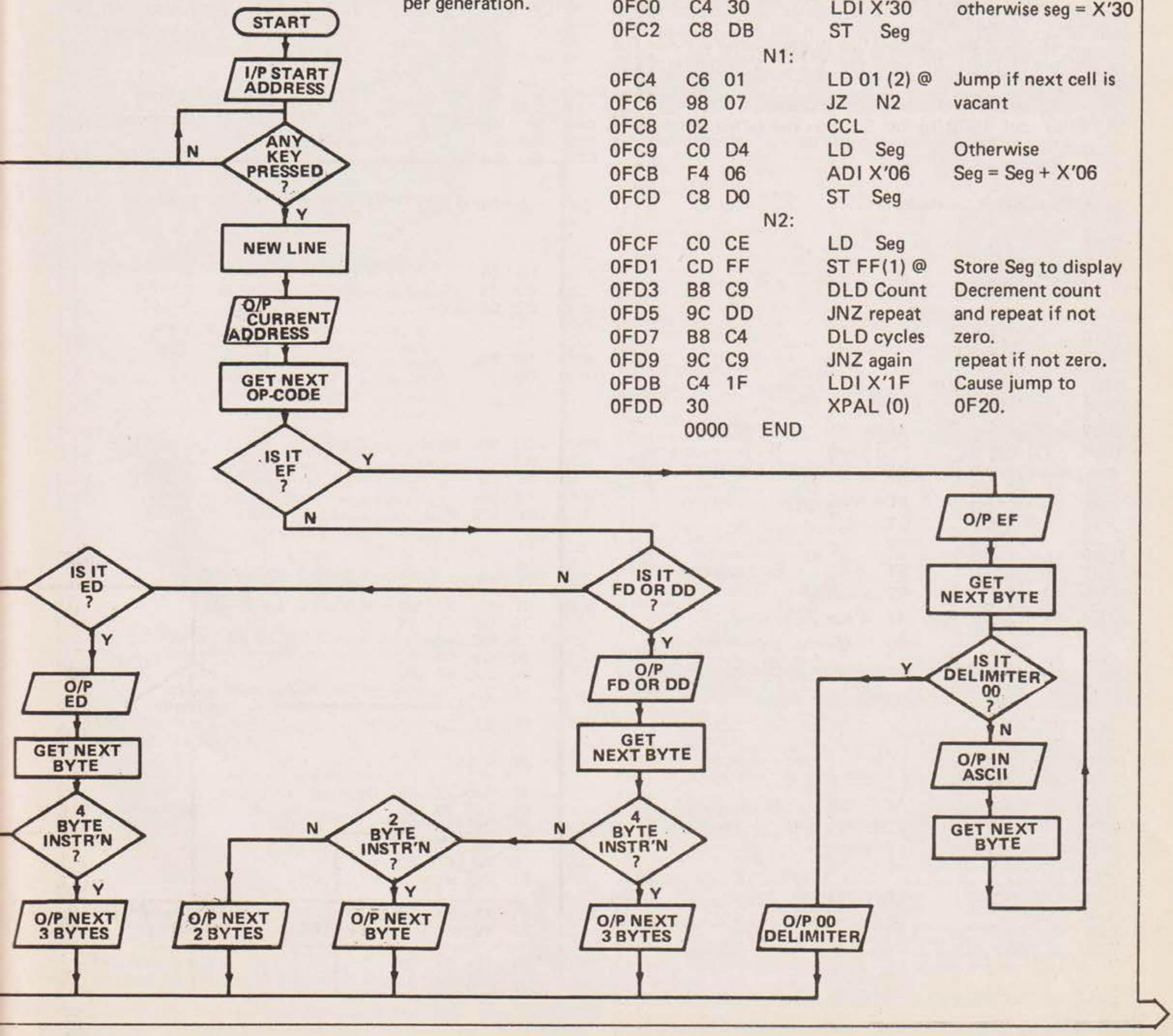
```
0F12        T0  =  +1
0F13        T1  =  +1
0F14        T2  =  +1
0F15        N   =  +1
0F16        Sum =  +1

            Start:
0F20  C4 0E       LDI X'0E
0F22  36          XPAH (2)       P2 = 0E80
0F23  C4 80       LDI X'80       (or start of cells).
0F25  32          XPAL (2)
0F26  C4 00       LDI X'00
0F28  C8 EC       ST   N
0F2A  C8 E8       ST   T1
0F2C  C8 E7       ST   T2        clear stores
0F2E  CA 11       ST  X'11 (2)
0F30  CA 10       ST  X'10 (2)
                  4 x NOP        gaps filled with
                                 NOPs
            Next:
0F36  C6 01       LD  01 (2) @
0F38  C8 D9       ST   T0        T0 = state of cells.
0F3A  02          CCL
0F3B  C0 D7       LD   T1
0F3D  F0 D6       ADD  T2        Sum = no of
0F3F  F6 01       ADD 01 (2) @        neighbours
0F41  F2 00       ADD 00 (2)
0F43  C8 D2       ST   Sum
0F45  C0 CC       LD   T0
0F47  9C 11       JNZ  Space     Jump if cell present
0F49  03          SCL
0F4A  C0 CB       LD   Sum
0F4C  FC 02       CAI X'02
0F4E  98 20       JZ   Born      If sum = 2 or 3 then
0F50  03          SCL              jump to born.
0F51  FC 01       CAI X'01
0F53  98 1B       JZ   Born
0F55  90 21       JMP  Same      Otherwise as before.
                  3 x NOP        Gaps filled with
                                   NOPs
            Space:
0F5A  03          SCL
0F5B  C0 BA       LD   Sum
0F5D  FC 02       CAI X'02
0F5F  98 17       JZ   Same      If sum = 2 or 4 cell
0F61  03          SCL            remains the same.
0F62  FC 02       CAI X'02
0F64  98 12       JZ   Same
                  4 x NOP
            Dies:
0F6A  C4 00       LDI X'00       Otherwise dies.
0F6C  CA FE       ST   FE (2)    (clear location)
0F6E  90 08       JMP  Same
            Born:
0F70  C4 01       LDI X'01       Cell is born
0F72  CA FE       ST   FE (2)    (set location)
                  4 x NOP
```

Flowchart for the INTAB program (see next page).



```
                Same:
0F78   C6 FF          LD FF(2) @     Decrement pointer.
0F7A   A8 9A          ILD N          Increment counter
0F7C   E4 10          XRI X'10       and if all cells
0F7E   98 20          JZ Display     examined jump to
0F80   C0 92          LD    T1       display.
0F82   C8 91          ST    T2       T2 = T1
0F84   C0 8D          LD    T0
0F86   C8 8C          ST    T1       T1 = T0
0F88   90 AC          JMP  Next      go and examine next
                                     cell.
0F9C            Cycles = +1          Display Port
0F9D            Count  = +1
0F9E            Seg    = +1
                Display:             Entry point
0FA0   C4 FF          LDI X'FF       Set cycles counter,
0FA2   C8 F9          ST Cycles      X'FF gives 5 seconds
                                     per generation.
```

```
0FA4   C4 0D   Again: LDI X'0D
0FA6   35             XPAH (1)       P1 = 0D08
0FA7   C4 08          LDI X'08
0FA9   31             XPAL (1)
0FAA   C4 0E          LDI X'0E
0FAC   36             XPAH (2)       P2 = 0E80
0FAD   C4 80          LDI X'80       (or start of cells)
0FAF   32             XPAL (2)
0FB0   C4 10          LDI X'10       count = X'10
0FB2   C8 EA          ST Count
               Repeat:
0FB4   C4 FF          LDI X'FF       Delay
0FB6   8F 01          DLY X'01
0FB8   C4 00          LDI X'00
0FBA   C8 E3          ST    Seg      Clear seg.
0FBC   C6 01          LD  01 (2) @
0FBE   98 04          JZ    N1       Jump if cell vacant
0FC0   C4 30          LDI X'30       otherwise seg = X'30
0FC2   C8 DB          ST    Seg
               N1:
0FC4   C6 01          LD 01 (2) @    Jump if next cell is
0FC6   98 07          JZ    N2       vacant
0FC8   02             CCL
0FC9   C0 D4          LD    Seg      Otherwise
0FCB   F4 06          ADI X'06       Seg = Seg + X'06
0FCD   C8 D0          ST    Seg
               N2:
0FCF   C0 CE          LD    Seg
0FD1   CD FF          ST FF(1) @     Store Seg to display
0FD3   B8 C9          DLD Count      Decrement count
0FD5   9C DD          JNZ repeat     and repeat if not
0FD7   B8 C4          DLD cycles     zero.
0FD9   9C C9          JNZ again      repeat if not zero.
0FDB   C4 1F          LDI X'1F       Cause jump to
0FDD   30             XPAL (0)       0F20.
       0000    END
```



R.M. Lucas

INTAB (Intelligent Tabulate) for NASCOM 1

Too often programs are supplied in a tabulated form, possibly with errors, which prove hard to track down.

INTAB is a no frills "disassembler" which generates an address for each op-code an prints that op-code and its associated operand(s), if any on the same line. (like a properly written coding sheet).

Though INTAB does not generate any mnemonics anyone who has spent any time programming in hex will be familiar with most of the more common instructions.

USE

Set locations 0C51 & 0C52 to the address at which INTAB is to start (lower address first) and execute from 0C50. Pressing any key will step through the tabulate one instruction at a time. Exit is via RESET'

DATA

The NASCOM STRING output routine (EF ..data.. 00) is dealt with specially. The characters in the string are printed in ASCII, including the control characters 1D,1E,1F.

It is impossible however to protect against data tables, these do not tend to look like executable programs anyway.

Intab is 200D bytes long and runs under both Nasbug and B -- Bug.

ADDRESS	MACHINE CODE	LABEL	OP	TO	FROM	REMARK
0C50	11 n n	START	LD	DE	n n	
C53	CD 3E 00	WAIT	CALL	CHIN		
C56	01 05 00		LD	BC	0005	
C59	CD 40 02		CALL	CRLF		
C5C	7A		LD	A	D	
C5D	CD 44 02		CALL	B2HEX		
C60	7B		LD	A	E	
C61	CD 44 02		CALL	B2HEX		
C64	EF 20 20 00		RST	40		
C68	1A		LD	A	(DE)	
C69	FE EF		CP	EFH		
C6B	28 58		JRZ	STRING		
C6D	FE DD		CP	DDH		
C6F	28 23		JRZ	INDEX		
C71	FE FD		CP	FDH		
C73	28 1F		JRZ	INDEX		
C75	FE ED		CP	EDH		
C77	28 32		JRZ	Z80 EDs		
C79	0E 1A		LD	C	1AH	
C7B	21 D4 0C		LD	HL	0CD4	
C7E	ED B1		CPIR			
C80	06 03	3 BYTES	LD	B	03H	
C82	28 0B		JRZ	PRINT		
C84	01 19 00		LD	BC	0019	
C87	ED B1		CPIR			
C89	06 02	2 BYTES	LD	B	02H	
C8B	28 02		JRZ	PRINT		
C8D	06 01	1 BYTE	LD	B	01H	
C8F	CD BA 0C	PRINT	CALL	CODE 2		
C92	18 BF		JR	WAIT		
0C94	CD B8 0C	INDEX	CALL	CODE 1		
C97	21 EB 0C		LD	HL	0CEB	
C9A	ED B1		CPIR			
C9C	28 E2		JRZ	3 BYTES		
C9E	21 0D 0D		LD	HL	0D0D	
CA1	0E 0B		LD	C	0BH	
CA3	ED B1		CPIR			
CA5	28 E6		JRZ	1 BYTE		
CA7	06 02		LD	B	02H	
CA9	18 E4		JR	PRINT		
0CAB	CD B8 0C	Z80 EDs	CALL	CODE 1		
CAE	0C		INC	C		
CAF	21 07 0D		LD	HL	0D07	
CB2	ED B1		CPIR			
CB4	28 CA		JRZ	3 BYTES		
CB6	18 D5		JR	1 BYTE		
0CB8	06 01	CODE 1	LD	B	01H	
CBA	CD 44 02	CODE 2	CALL	B2HEX		
CBD	CD 3C 02		CALL	SPACE		
CC0	13		INC	DE		
CC1	1A		LD	A	(DE)	
CC2	10 F6		DJNZ	CODE 2		
CC4	C9		RET			
0CC5	CD B8 0C	STRING	CALL	CODE1		
CC8	A7	S1	AND	A		
CC9	28 C2		JRZ	1 BYTE		
CCB	F6 80		OR	80H		
CCD	CD 3B 01		CALL	CRT		
CD0	13		INC	DE		
CD1	1A		LD	A	(DE)	
CD2	18 F4		JR	S1		
0CD4	01 11 31 32					8080 3 byters
CD8	3A C2 C3 C4					
CDC	CA CC CD D2					
CE0	D4 DA DC E2					
CE4	E4 EA EC F2					
CE8	F4 FA FC					
0CEB	21 22 2A					2nd byte of Z80 INDEX 4 byters
CEE	36 CB					
CF0	06 0E 10 16					8080 2 byters
CF4	18 1E 20 26					
CF8	28 2E 30 38					
CFC	3E C6 CE D3					
D00	D6 DB DE E6					
D04	EE F6 FE					
0D07	43 4B 53 5B					2nd byte of Z80 EDs 4 byters
D0B	73 7B					
0D0D	09 19 23 29					2nd byte of Z80 INDEX 2 byters
D11	2B 39 E1 E3					
D15	E5 E9 F9					

CT MARKETPLACE

ALARM-CHRONO LCD

This new addition to our unbeatable selection of bargains is no ordinary LCD watch. It's a slim, multi-function, dual time chronograph alarm watch, no less.

This model will show hours, minutes, seconds, date, day of the week, stop watch, split time, alarm and alternate dual time zone — not all at once, of course. There is also a night light.

Hours, minutes, seconds and day of the week are displayed continuously, while the date will appear at the touch of a button. The day of the week is indicated by a flag. When used as a stopwatch, the maximum count is 0.1secs. short of thirteen hours.

£19.95

(Inclusive of 15% VAT and Postage).

To:
ALARM/CHRONO LCD WATCH Offer,
CT Magazine,
145 Charing Cross Road,
London WC2H 0EE

Please find enclosed my cheque/PO for £19.95 (payable to CT Magazine) for my Alarm/Chrono LCD watch.

Name . . . .

Address . . . .

. . . .

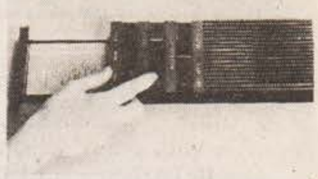
. . . .

Please allow up to 28 days for delivery.

24 TUNE DOOR CHIMES

DOOR TUNES £17.13 + VAT.

Waddington's Videomaster announce a doorbell that doesn't go Brrringgg, Ding-Dong or Bzzzzz. Instead it plays 24 different classical and popular tunes. It will play the tune you select for your mood, the season or the visitor you are expecting to call. Door tunes is not only great fun and a wonderful ice breaker, but is also very functionally and beautifully designed to enhance your home. There is something for Christmas, something for your continental visitors or your relations from the states, and even something for the Queen. Door tunes is easy to install and has seperate controls for volume, tone and tempo.



T.V. GAMES

PROGRAMMABLE £29.50 + VAT.
COLOUR CARTRIDGE T.V. GAME.

The TV game can be compared to an audio cassette deck and is programmed to play a multitude of different games in COLOUR, using various plug-in cartridges. At long last a TV game is available which will keep pace with improving technology by allowing you to extend your library of games with the purchase of additional cartridges as new games are developed. Each cartridge contains up to ten different action games and the first cartridge containing ten sports games is included free with the console. Other cartridges are currently available to enable you to play such games as Grand Prix Motor Racing, Super Wipeout and Stunt Rider. Further cartridges are to be released later this year, including Tank Battle, Hunt the Sub and Target. The console comes complete with two removable joystick player controls to enable you to move in all four directions (up/down/right/left) and built into these joystick controls are ball serve and target fire buttons. Other features include several difficulty option switches, automatic on screen digital scoring and colour coding on scores and balls. Lifelike sounds are transmitted through the TV's speaker, simulating the actual game being played.

Manufactured by Waddington's Videomaster and guaranteed for one year.

EXTRA CARTRIDGES:

ROAD RACE – £8.87 + VAT.
Grand Prix motor racing with gear changes, crash noises

SUPER WIPEOUT – £9.17 + VAT.
10 different games of blasting obstacles off the screen.

STUNT RIDER – £12.16 + VAT.
Motorcycle speed trials, jumping obstacles, leaping various rows of up to 24 buses etc.

NON-PROGRAMMABLE TV GAMES

6 Game – COLOURSCORE II – £13.50 + VAT.

10 Game COLOUR SPORTSWORLD £22.50 + VAT.

CHESS COMPUTERS

STAR CHESS – £55.09 + VAT,
PLAY CHESS AGAINST YOUR PARTNER.

using your own TV to display the board and pieces. Star Chess is a new absorbing game for two players, which will interest and excite all ages. The unit plugs into the aerial socket of your TV set and displays the board and pieces in full colour (or black and white) on your TV screen. Based on the moves of chess. It adds even more excitement and interest to the game. For those who have never played, Star Chess is a novel introduction to the classic game of chess. For the experienced chess player, there are whole new dimensions of unpredictability and chance added to the strategy of the game. Not only can pieces be taken in conventional chess type moves, but each piece can also exchange rocket fire with its opponents. The unit comes complete with a free 18V mains adaptor, full instructions and twelve months guarantee.

CHESS CHALLENGER 7 – £85.65 + VAT.
PLAY CHESS AGAINST THE COMPUTER.

The stylish, compact, portable console can be set to play at seven different levels of ability from beginner to expert including "Mate in two" and "Chess by mail". The computer will only make responses which obey international chess rules. Castling, on passant, and promoting a pawn are all included as part of the computer's programme. It is possible to enter any given problem from magazines or newspapers or alternatively establish your own board position and watch the computer react. The positions of all pieces can be verified by using the computor memory recall button.

Price includes unit with wood grained housing, and Staunton design chess pieces. Computer plays black or white and against itself and comes complete with a mains adaptor and 12 months guarantee.

OTHER CHESS COMPUTERS IN OUR RANGE INCLUDE

CHESS CHAMPION – 6 LEVELS £43.95 + VAT.
CHESS CHALLENGER – 10 LEVELS – £138.70 + VAT.
BORIS – MULTI-LEVEL TALKING DISPLAY £163.04 + VAT.

ELECTRONIC CHESS BOARD TUTOR £17.17 + VAT.

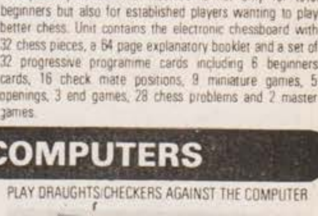
A special bulk purchase of these amazing chess teaching machines enables us to offer them at only £19.75 less than half recommended retail price. The electronic chess tutor is a simple battery operated machine that can actually teach anyone to play chess and improve their game right up to championship level. This machine is not only for total beginners but also for established players wanting to play better chess. Unit contains the electronic chessboard with 32 chess pieces, a 64 page explanatory booklet and a set of 32 progressive programme cards including 6 beginners cards, 16 check mate positions, 9 miniature games, 5 openings, 3 end games, 28 chess problems and 2 master games.

DRAUGHTS COMPUTERS

CHECKER CHALLENGER 2 LEVELS £43.98 + VAT.
4 LEVELS £80.09 + VAT.

The draughts computer enables you to sharpen your skills, improve your game, and play whenever you want. The computer incorporates a sophisticated, reliable, decision-making microprocessor as its brain. Its high level of thinking ability enables it to respond with its best counter moves like a skilled human opponent. You can select offence or defence and change playing difficulty levels at any time. Positions can be verified by computer memory recall. Machine does not permit illegal moves and can solve set problems. Computer comes complete with instructions, mains adapator and twelve months guarantee.

PLAY DRAUGHTS/CHECKERS AGAINST THE COMPUTER



FOR FREE BROCHURES – SEND S.A.E

For FREE illustrated brochures and reviews on TV and chess games please send a stamped addressed envelope, and state which particular games you require information on.

Callers welcome at our shop in Welling – demonstrations daily – open from 9am-5.30pm Mon-Sat (9am-1pm Wed).

To order by telephone please quote your name, address and Access/Barclaycard number

Postage and Packing FREE

AJD DIRECT SUPPLIES LIMITED, Dept **CT 10** 102 Bellegrove Road, Welling, Kent DA16 3QD. Tel: 01-303 9145 (Day) 01-850 8652 (Evenings)

Mr. J.R. Keneally

Get your priorities right!

This program is a good illustration of the flexibility of the basic NASCOM kit. The on-board PIO has been used to provide regular program interrupts which activate a priority scheduler. Up to five independent programs can be run under control of the Scheduler, with each program able to activate or de-activate other programs, or communicate via common data areas. Normally, however, each program will be run at regular intervals preset by the user.

Priority Rules OK!

Since the Scheduler is based on a priority scheme, a high priority program will suspend a lower priority program if the former becomes eligible to run before the lower priority program has finished. Consequently, higher priority programs will normally be of short execution time. There are many applications for a system of this type, for example:—

(1) Polling of a number of devices e.g. heating control systems, burglar alarms etc., while allowing 'simultaneous' execution of applications programs.

(2) Real-time simulation programs.

(3) Non time-critical communication between systems.

Software Details

In the system described, the NASCOM monitor is automatically run as the lowest priority program. However, some of the monitor commands can only be used with extreme care, in particular those which either reset the stack pointer, or cause re-entry to the monitor. These include Execute, Breakpoint and Single Shot commands. It is possible to overcome these restrictions, but since it is risky to do program testing due to the possibility of a system crash, use of the S and B commands is not recommended. The E command is not needed, since any program can be run under control of the Scheduler as described below.

Activation Of The Scheduler

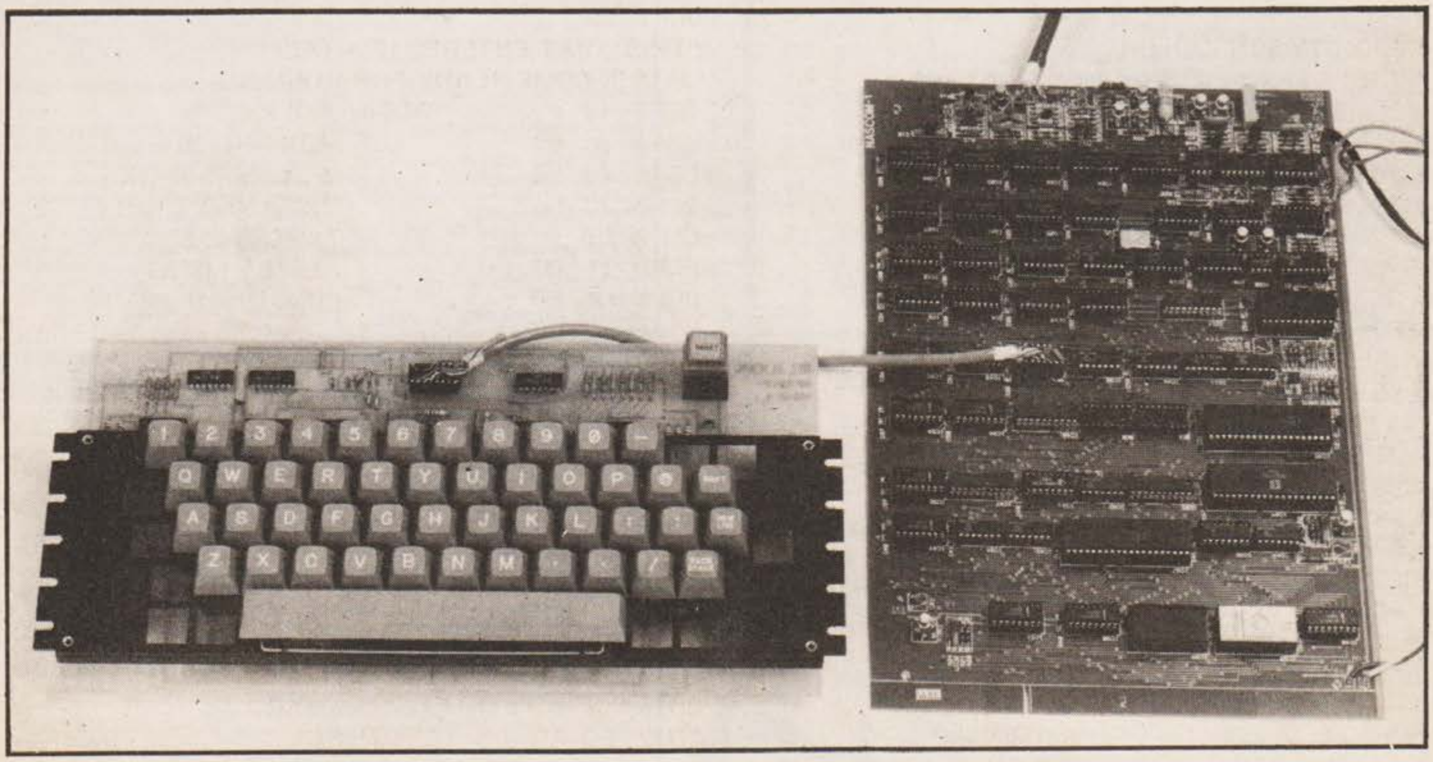
The Scheduler program should be loaded from cassette or keyboard as usual. Then, the monitor command EC90 is used to start the system. The monitor prompt should reappear immediately, showing that the monitor has been re-entered. To test the system, try the following program:—

```
     >ME00
     0E00>32 98 0A 3C 32 98 0A C9
     >MC72
     0C72>08 08 04 00 0E
     >
```

This should cycle through all possible characters on the TV. If there is no response, the PIO may be in a non-reset state. This can usually be cleared by single-shotting a RETI instruction, and/or clearing the PIO internal interrupt enable (see PIO manual). When in doubt, check pin 22 of the PIO. If this is at zero volts, the PIO is hung-up waiting for a CPU response to an interrupt request. In all cases, as a last resort, the power can be switched off to clear the PIO.

Installing Programs

To run a program under the Scheduler, five control bytes must be input using the monitor M command. Byte 1 is the frequency with which the program is run, byte 2 the delay before the first activation, byte 3 is set to 4 to indicate that the program is available to run (i.e. loaded), and bytes 4 and



The Basic Nascom 1 which runs the Scheduler program.

NASCOM SCHEDULER

5 specify the entry address. Note that bytes 4 and 5 must be set up before byte 3, otherwise the program will execute from an arbitrary address. However, no problems have been experienced if all five bytes are input in one M command line.

The highest priority control bytes begin at address 0C72. Lower priorities occupy higher addresses, e.g. the next lower priority occupies location 0C77 onwards.

Hardware Details
The Scheduler expects bit 0 of PIO port A to be toggled with a TTL level clock to pin 1 of SKA. The remaining bits of port A are set up as outputs, and can be used by any program. Typical clock frequency would be in the range 30 to 60 Hz. This clock can be conveniently obtained from the on board oscillator intended to drive a printer, if the frequency is reduced by using an electrolytic across C12. The output from this oscillator is available at LINK 4 on the main board.

General Restrictions
To minimise stack space requirements, the alternate register set of the Z80 is not saved when a program is suspended. This means that only one program can use these at present. However, this is easily changed by saving and restoring these in the PIO interrupt routine.

Writing Programs For The Scheduler
The only restrictions on programs written to run under the Scheduler are that the stack pointer should not be re-assigned, and the program should terminate with a RET instruction. Care must also be taken with use of monitor routines, since these are not re-entrant.

Fig. 1. The program listing for the Scheduler.

```
"PRIORITY SCHEDULER
"FIRST DEFINE INTERRUPT.   VECTORS
0C60  B2 0C                  AD PIO
0C62  XX..XX                 RS 14
0C70  05 00                  DT 5,0
0C72  00 00 00 00 00         DT 0,0,0,0,0
0C77  00 00 00 00 00         DT 0,0,0,0,0
0C7C  00 00 00 00 00         DT 0,0,0,0,0
0C81  00 00 00 00 00         DT 0,0,0,0,0
0C86  00 00 05 00 00         DT 0,0,5,0,0
"NOW SET UP PIO CONTROL WORDS
0C8B  CF 01 60 B7 FE         DT CFH, 1, 60H, B7H, FEH
"ENTRY POINT FOR SCHEDULER
0C90  21 71 0C 36 88         MOV !HL, C71H & (HL),88H
0C95  11 04 00               MOV DE,4
0C98  2B 46                  DEC HL; MOV B, (HL)
0C9A  19               LP1:  ADD HL,DE
0C9B  CB 8E 23               RSETB1 (HL); INC HL
0C9E  10 FA                  JR, DNZ LP1
"ABOVE CODE INITS. TASK CONTROL LIST
"NOW ACTIVATE PIO
0CA0  3E 0C ED 47            MOV !A,0CH & I,A
0CA4  ED 5E                  INTMODE 2
0CA6  FB                     INT ON
0CA7  0E 06 2E 8B            MOV !C,6 & L,8BH
0CAB  06 05                  MOV B,5
0CAD  ED B3                  OUT, INCRPT
0CAF  C3 86 02               JP 286H
                             "JUMPS BACK TO...
                             "NASCOM MONITOR
0CB2  C5               PIO:  PUSH BC
                             "PIO INTR. S/R
0CB3  D5 E5 DD E5            PUSH !DE&HL&IX
0CB7  FD E5                  PUSH IY
0CB9  F5                     PUSH AF
0CBA  21 71 0C               MOV HL,C71H
0CBD  6E 45                  MOV !L, (HL) & B,L
0CBF  CB CE                  SETB1 (HL)
0CC1  CD 17 0D               CALL RETIN
0CC4  FB 2E 87               INTON; MOV L,87H
0CC7  35              LOOP:  DEC (HL)
0CC8  20 0D 2D               JR,NZ SKIP; DEC L
0CCB  7E 2C                  MOV A, (HL); INC L
0CCD  77 2C                  MOV (HL),A; INC L
0CCF  CB 56                  TSTB2 (HL)
0CD1  28 03                  JR,Z SKIA
0CD3  CB C6 45               SETB0 (HL); MOV B,L
0CD6  2D              SKIA:  DEC L
0CD7  7D D6 05        SKIP:  MOV A,L; SUB 5
0CDA  FE 6E 6F               CP 6EH; MOV L,A
0CDD  20 E8                  JR,NZ LOOP
0CDF  2E 71 7E               MOV!L,71 & A, (HL)
0CE2  B8 20 0D               CP B; JR,NZ ACTA
0CE5  70             REACT:  MOV (HL),B
0CE6  6E             RACTA:  MOV L, (HL)
0CE7  CB 8E                  RSTB1 (HL)
0CE9  F1 FD E1               POP !AF & IY
0CEC  DD E1 E1               POP !IX & HL
0CEF  D1 C1                  POP !DE & BC
0CF1  C9                     RET
"THIS POINT ENTERED IF A TASK''
"HAS BECOME READY FOR RUNNING
0CF2  38 F2           ACTA:  JR,C RACTA
0CF4  70 68            ACT:  MOV !(HL), B & L,B
0CF6  2C 5E                  INC L; MOV E, (HL)
0CF8  2C 56                  INC L; MOV D, (HL)
0CFA  EB                     SWP DE,HL
0CFB  11 00 0D               MOV DE,RSTRT
0CFE  D5 E9                  PUSH DE; JP (HL)
                             "JUMP TO NEW PROGRAM
0D00  21 71 0C       RSTRT:  MOV HL,C71H
                             "RENTRY AFTER TASK
                             "COMPLETION
0D03  7E 6F                  MOV !A, (HL) & L,A
0D05  CB 86                  RSETB0 (HL)
0D07  C6 05 6F       SKIPB:  ADD 5; MOV L,A
0D0A  CB 46                  TSTB0 (HL)
0D0C  28 F9                  JR,Z SKIPB
0D0E  CB 4E                  TSTB1 (HL)
0D10  45 2E 71               MOV !B,L & L,71H
0D13  20 D0                  JR,NZ REACT
0D15  18 DD                  JR ACT
0D17  ED 4D          RETIN:  RETI

                             END
```

computing today

WHAT TO LOOK FOR
IN THE NOVEMBER ISSUE.
ON SALE 12TH OCTOBER

TURING OVER A NEW LEAF

The famous English mathematician Alan Turing, apart from developing the early ACE computer at the National Physical Laboratories had previously produced a theoretical machine. Called, not surprisingly, a Turing Machine it is a mathematical model of a computer. In this article by Giles Gummer the theory of these machines is explained and a simulation program is presented for the Nascom.

TRS–80 FOR YOU?

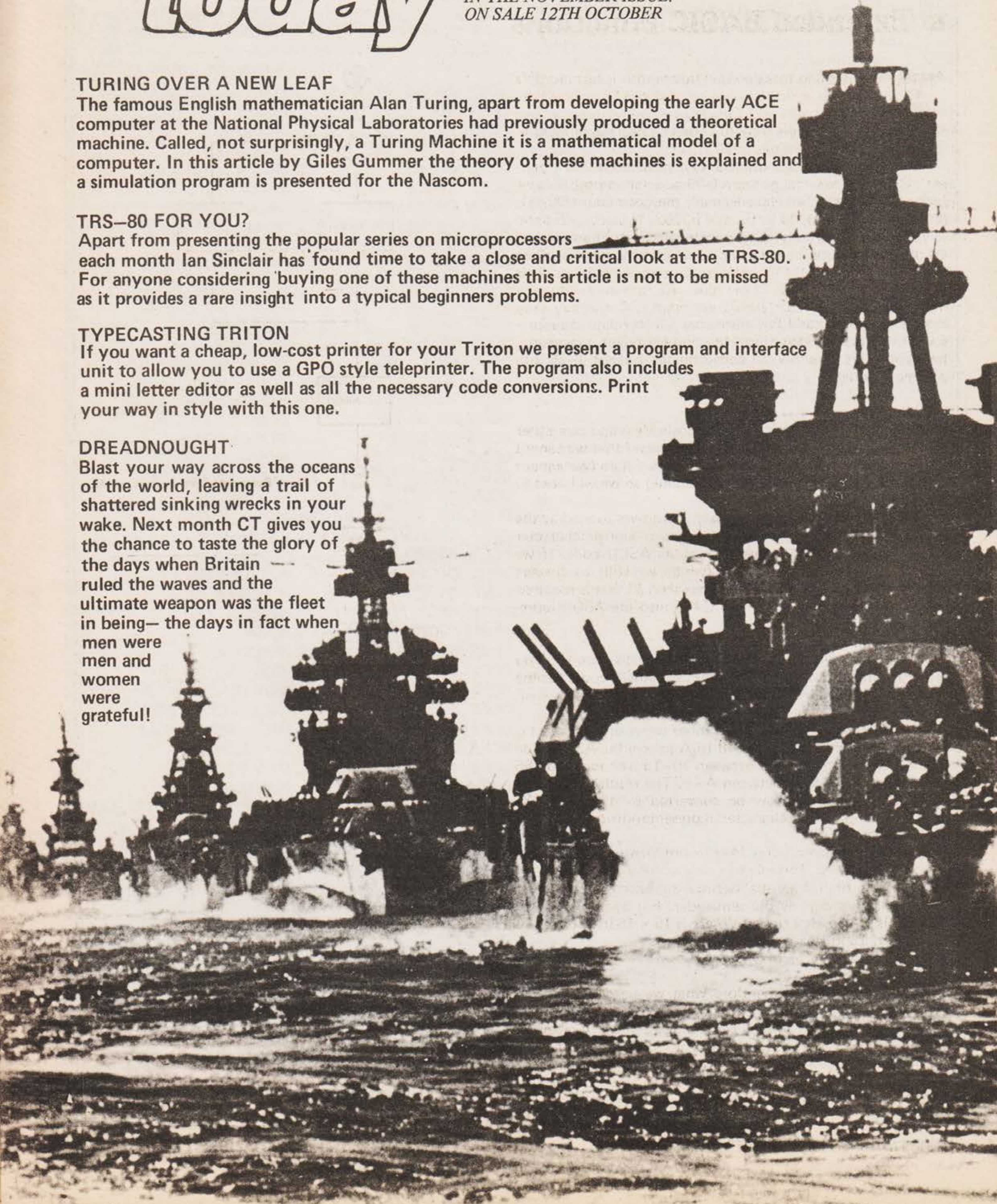
Apart from presenting the popular series on microprocessors each month Ian Sinclair has found time to take a close and critical look at the TRS-80. For anyone considering buying one of these machines this article is not to be missed as it provides a rare insight into a typical beginners problems.

TYPECASTING TRITON

If you want a cheap, low-cost printer for your Triton we present a program and interface unit to allow you to use a GPO style teleprinter. The program also includes a mini letter editor as well as all the necessary code conversions. Print your way in style with this one.

DREADNOUGHT

Blast your way across the oceans of the world, leaving a trail of shattered sinking wrecks in your wake. Next month CT gives you the chance to taste the glory of the days when Britain — ruled the waves and the ultimate weapon was the fleet in being— the days in fact when men were men and women were grateful!



Phil Cornes.

Phil Cornes continues his look at Extended BASIC Functions

The first thing to deal with this month is last month's homework, to make sure you get the hang of all the string functions and Extended BASIC generally. The first problem we set you was to convert a decimal number in the range 0–255 into 2 digits HEX.

Hexadecimal is a number system based on 16 different "digits". These range from 0–9 and for the other 6 we use letters A to F. In Hexadecimal, then, we count 00, 01, 02 . . . . . . .0F, 10, 11 . . . . 1F, 20, 21 . . . . 2F, and so on up to FF. In each digit position we can put any one of 16 different symbols so that in a 2 digit Hex number there are 16 x 16 (ie 256) different permutations of symbols. This means that if we start from zero we can represent any integer from 0 to 255 with 2 hex digits. Conversely, any integer between 0 and 255 needs only 2 Hex digits to represent it. Before we delve into the two Hex digit conversion, let us look at a method of converting an integer 0–15 into a single Hex digit.

Handling Hex

The first point to note is that our single Hex digit can either be a number 0–9 or a letter A–F. This means that we cannot use a numeric variable to store the Hex digit (we cannot store character A–F in a numeric variable) so we will have to use a string (say A$).

If you remember from last month we looked at the CHR$(X) function. This function returns a single character string which has X (in decimal) as its ASCII code. If we decide to utilize this function (which we will) to convert from ASCII code to a Hex character then all that is required is to convert the input number 0–15 into the ASCII representation of its HEX equivalent.

Consider Fig. 1:–

Let A be the number 0–15 that we wish to convert. You may remember from last month that the ASCII codes for the digits 0–9 were 48–57 and the ASCII codes for letters A–F were 65–70.

This means that if the number we wish to convert is between 0–9 we should add 48 to A to give the ASCII code of the Hex digit and if A is between 10–15 we should add 55 to give an ASCII code between A–F. The resulting value of C (ASCII code) can now be converted by the CHR$(X) function to the Hex character representation of the input value of A.

We can extend this idea to produce two Hex digits. The most significant Hex digit in a two Hex digit number is the number of 16s in the decimal number and the least significant Hex digit is the remainder. For example 250 in decimal is FA in Hex because there is 15 x 16 in 250 and 10 remainder. 15 in Hex is F and 10 in Hex is A.

An Easy Solution

To solve our problem therefore what we need to do is to divide the input number up into two decimal numbers between 0–15 each of which can then be converted separately into Hex digits by the subroutine of Fig. 1. This procedure is laid out in Fig. 2 (to save re-drawing it, Fig. 1 is called as a subroutine of Fig. 2). Both of these flow charts can be coded into BASIC as follows.

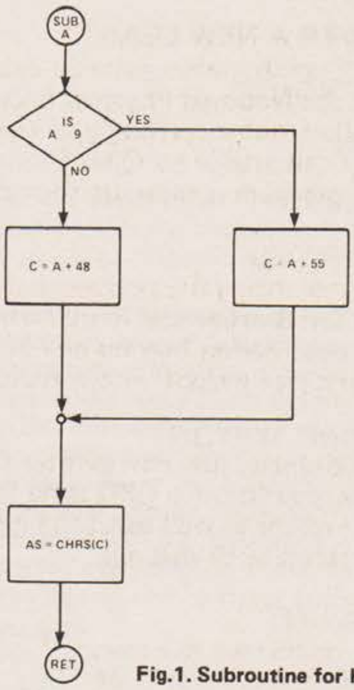


Fig.1. Subroutine for Hex conversion.

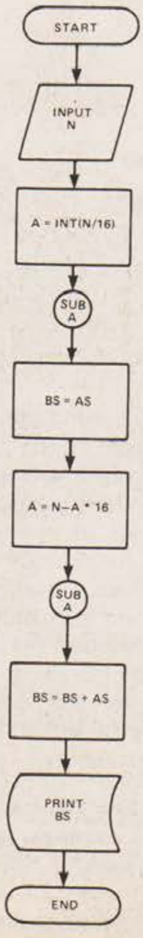


Fig.2. Complete BASIC flowchart for conversion program.

```
 10 INPUT N
 20 A=INT(N/16)
 30 GOSUB 500
 40 B$=A$
 50 A=N–A*16
 60 GOSUB 500
 70 B$=B$+A$
 80 PRINT B$
 90 END
500 IF A > 9 THEN 530
510 C=A+48
520 GOTO 540
530 C=A+55
540 A$=CHR$(C)
550 RETURN
600 END
```

NOTE:– The END statement of line 90 is needed otherwise the computer will "crash" into the subroutine at line 500 straight after the printout of B$ by line 80 even though there is no subroutine call. Thus when the computer reaches line 550 it will execute a return statement and find that there is no return address stored and "bomb out".

Similar principles to those given above using the ASC(STRING) function can be applied to the second question of the homework. The flow charts of Figs. 3 and 4 and the program below are given as a possible solution without explanation. It is left to you to sort out how they work (good practice).

```
 10 INPUT A$
 20 B$=LEFT$(A$,1)
 30 GOSUB 500
 40 N=A*16
 50 B$=RIGHT$(A$,1)
 60 GOSUB 500
 70 T=T+A
 80 PRINT T
 90 END
500 C=ASC(B$)
510 IF C > 57 THEN 540
520 A=C–48
530 GOTO 550
540 A=C–55
550 RETURN
600 END
```

Right! Having demonstrated some of Extended BASIC's facilities we will now go on to look at the third thing we mentioned last month for homework – sorting. Approximately 95% of all computer time used anywhere is used by commercial concerns in the process of running a business. Very little computer time is used for scientific applications. Of this vast amount of computer time used commercially, about one third is spent sorting and searching through lists of data items so that useful information may be extracted. It is to these two areas, sorting and searching that our attention now turns and this month we will look at a simple search and at how to merge two sorted lists together.

Simple Search

Imagine a list of names, addresses and telephone numbers. If we are given a name we can look through the list for this name and when we find it we can look up the corresponding address and telephone number. This process is called searching and the item we are using to find the required information is called the key. Here we are using the name to find the address and telephone number so the name is the key. If the items in the list are in random order, then there is really

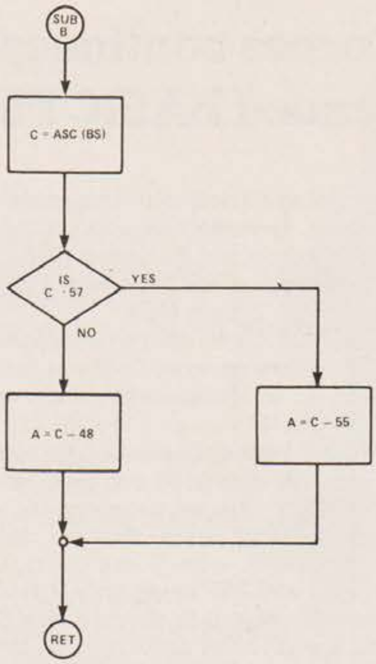


Fig.3. Subroutine for Hex to Decimal conversion.

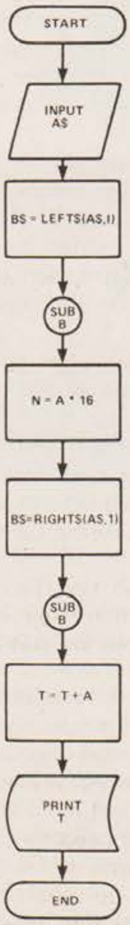


Fig.4. Complete BASIC flowchart for reverse conversion.

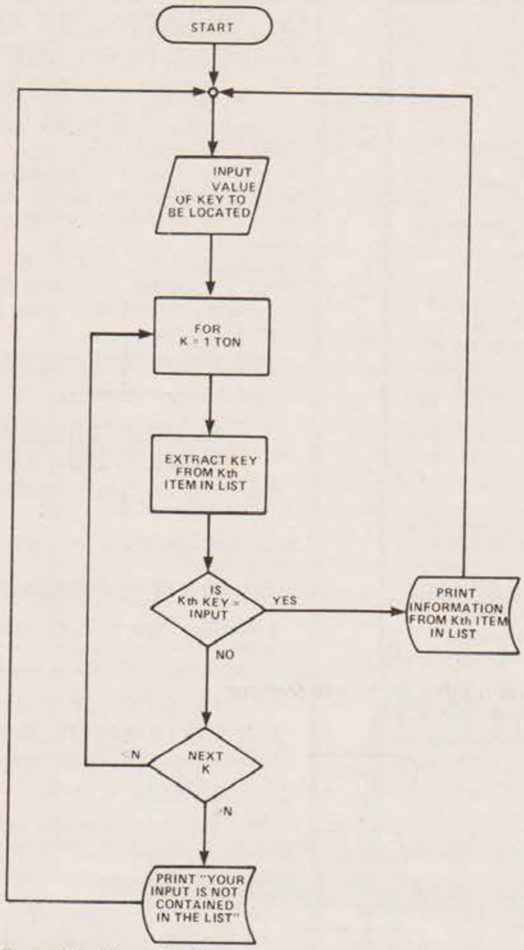


Fig.5. Search routine flowchart.

only one way of finding any given item; that is to look through the list from the beginning, one item at a time, until the required item is located.

So the flowchart of an algorithm to perform this simple searching technique might be as Fig. 5. The following notes might help you to understand Fig. 5 better. N is the number of items in the list and K is the number of the item that we are currently looking at. The list itself might well be contained in a string array, eg A$(0) to A$(9) is a list of ten elements each of which would be capable of storing 256 characters.

Working through Fig. 5 the first box asks that we input the key that we are to search for (eg input the name whose address and telephone number we seek). We then set up a FOR NEXT loop from 1 to the number of items in the list. The third box extracts the key from the Kth item of the list. This is necessary because it is quite possible that the search key might not be at the beginning of each item. For example, if our list were in telephone number order with telephone numbers specified first, then address specified second, and lastly name, we would need to extract the last so many characters to compare them with our input. This comparison is carried out in the 4th flowchart box. If it turns out that the extracted key from the Kth item on the list is the same as the input then the information (address and telephone number) is printed out. If the two items are not the same, K is incremented by one and the key is extracted from the next item in the list to be checked. This process is continued until the extracted key matches the input or until the computer has checked through all the items on the list and not found a match in which case a message to this effect is printed.

Two List Merging

Imagine two lists of numbers that have already been sorted out into numerical order and imagine further that these two lists have been assigned to two array variables, eg

X	A(X)	B(X)
1	1	–1
2	3	2
3	12	13
4	24	15

It is required that the two lists A(X) and B(X) be merged together to produce a single list containing 8 items and that the merged list say C(X) should also be in ascending numerical order.

The method of solution for this problem is to compare the first item in array A with the first item in array B. The least of these should be made the first item of array C. In this case B(1) is less than A(1) so C(1) will be set equal to B(1) (ie –1). We will now compare A(1) with B(2), the least of these will become C(2). Here A(1) < B(2) so C(2)= A(1)=1 and we then compare A(2) with B(2) and put the least of these into C(3) etc. A program to perform this algorithm is given below without any further explanation so that you can dry run it yourselves.

```
 10 DIM A(4),B(4),C(8)
 15 FOR X=1TO4
 20 INPUT A(X),B(X)
 25 NEX X
 30 A=1
 35 B=1
 40 C=1
 50 IF A(A) > B(B) THEN 90
 60 C(C)=A(A)
 70 A=A+1
 80 GOTO 110
 90 C(C)=B(B)
100 B=B+1
110 C=C+1
120 IF A=5 THEN 150
130 IF B=5 THEN 190
140 GOTO 50
150 FOR D=BTO4
160 C(D+4)=B(D)
170 NEXT D
180 GOTO 220
190 FOR D=A TO 4
200 C(D+4)=A(D)
210 NEXT D
220 FOR C=1TO8
230 PRINT C(C)
240 NEXT C
250 END
```

Try It Yourself

The homework for this month is to work out what lines 120–210 of the two list merge program above are included for and when you have worked it out try to think of another way of doing this which although not as general will reduce the length of the program listing considerably.

Next month we will go on to look at a more involved search algorithm and a simple sort routine.

MINI-ADS & CLASSIFIED

Complete

TRS 80 and SORCERER

systems with lower case and word processing.

MICROPOLIS disc drives £350

SOFTWARE proper business systems

For information please write:

EMG Box 118,
30 Heathfield Road,
Croydon, Surrey.

GAMES FOR NASCOM

Five new Nascom games

SNARK HUNT, CHASE AND TRAIL, BULLS AND COWS, FOUR IN A ROW AND LIFE ON CASSETTE.

All run on minimum Nascom system. All five recorded both sides. Fantastic value complete with printed instruction booklet. £11.50 inc. VAT and postage and packaging.

(Instruction Booklet only £1.00, refunded if games cassette purchased)

STRATHAND

44 St. Andrews Square,
Glasgow G1 5PL
041-552 6731

callers welcome

Tel. order welcome with Access and Barclaycard. Now on Telex 777268, 24 hours service

JADE SSM S100 INTER SYSTEMS WMC

S100 Motherboards — 9 slot and 12 slot
CPU Boards Z80 and Z80A kit or A & T
I/O Boards with tape interface kit or A & T
8K 450ns Static Ram Kit or A & T
16K 450ns Static Ram Kit or A & T
Video Boards with fine graphics
Floppy Disc controller D/S D/D
Main frame cabinets with Motherboard and PSU
5.25" and 8" Floppy Disc's   Eprom Boards

S.A.E. for price list and special offers

MBE (Electronic Specialists)
58 Croft Road,
Birmingham B26 1SQ
Phone 021-784-6632

INTENSIVE WEEKEND COURSES IN

BASIC

including hands-on mini computer operation.

This short intensive course is intended to instruct from minimal knowledge to an operational capability of computer programming in BASIC high level language. The course is fully residential from Friday evening to Sunday afternoon.
Option of non-residential weekend, weekday evening and weekday courses available if required.

For further details of dates available, fees, etc. **Phone (0401) 43139, or write to:**
Dept CT
CLEVELAND BUSINESS SERVICES
Cleveland House, ROUTH
Beverley, North Humberside

COLOUR MODULATOR

Kit **£9.95** inc UHF Modulator

FOR ALL TV GRAPHICS!

Red, Green, Blue inputs (can be mixed)
Coming soon: Nascom Colour Kit. **£39.95** inc. Graphics Software

WILLIAM STUART SYSTEMS
Dower House, Billericay Road, Herongate, Brentwood, Essex CM13 3SD Tel (0277) 810244
Barclaycard/Access welcome

STACKS OF MEMORY! 4K Bit Static MOS Ram N.E.C. UPD 410 D-2 (100ns). £4.00 each plus 30p P & P & Data. P. Town Products, 30 Wolsey Drive, Walton-on-Thames, Surrey. KT12 3AZ. Tel: Walton 21078.

PET CASSETTE PROGRAMS, 6502 Mnemonic loader (forget about Hex codes and work in decimal), £5, Computer Art, £3, Reaction Tester, £4, P. J. Slack, 5 Heywood Close, Alderley Edge, Cheshire.

TRITON, professionally built, in case, perfect working order, complete with manual, £265. Telephone Basingstoke 29623.

STEPPING MOTOR DRIVES: 0.5 amps/phase, £17. 4 amps/phase, £21. Motors 16 oz. in., £15. Send for details, R. Harden, 24 Belvoir Avenue, Trentham, Stoke-on-Trent.

TRS 80, Level II, 16K Microcomputer keyboard, video, cassette, power supply new and boxed, used twice only, £500 cash. Mr. Bailey, 32 Newells, Jackmans Estate, Letchworth, Hertfordshire.

ELF II, ex-demo system, under trade, £495 (list £697). Triton (full board Ram) £295. Olivetti TE300, offers? Most types bought/sold also software/hardware. Rayleigh 774089.

THE NEW uHEX 480 EPROM PROGRAMMER 2704 and 2708

Control programs for Z80, 8080, 6800, 6500. Please state machine.
Programs permit programming any length block into the eprom, so even unexpanded machines can now program eproms.
Needs only +5V, +12V, −5V.
Host computers require a PIO (PIA) complete kit only £35 or ready built and tested £5 extra.

PIO, PIA INTERFACE PANELS

Available for Z80/8080, 6800/6500

THE uHEX 416 DELUXE EPROM PROGRAMMER

Push Button Selection for 2704, 2708, 2716. Still only £65.

All prices inclusive. SAE for further information about these products.

MICRO HEX COMPUTERS
2 Studley Rise, Trowbridge, Wilts.

ATTENTION MICRO COMPUTER USERS

CASSETTES

A new range of top quality tapes Specifically designed for micro computers

FLOPPY DISCS

A complete range of professional products for all makes of machine and specifications.

FREE

Storage case for your cassettes or floppies on certain orders during September/October

Send for details to:—

N. SINE
105 Sanderstead Road, Croydon CR2 0PJ or Tel 01 681 1279

EXTENDED NASCOM with floating point, basic on Eproms and 8K memory, professionaly built and fully working with games tapes, Startrek, £300. Phone Lindfield (0203) 462697.

LIFE FOR NASCOM-1. Complete machine code program listing, with comments. 12 x 12 matrix, runs very fast. Bargain at £3.50. Chris Blackmore, 31 Herne Rise, Ilminster, Somerset, TA19 0HH.

TINY BASIC, Floating Point subroutines. Operations + − x ÷ √ Range ±9.999 number and ±32767 exponent. Plus SINE of integers. Listing £1.70, Triton cassette + listing £3.30. Andover Software Kits, 15 Winchester Road, Andover, Hants. SP10 2EG.

THIS SECTION IS A PRE-PAYMENT SERVICE ONLY

MINI-ADS: 31/4" x 21/8", 1-3 £38, 4-11 £36, 12 or more £34 per insertion. CLASSIFIED DISPLAY: 19p per word. Minimum 25 words. Boxed classifieds are £6.33 per col. centimetre. No P.O. Box Numbers can be accepted without full address.

Enquiries to: Advertising Department, 01-437 5982. 25-27 Oxford Street, London W1R 1RF

AD INDEX

SEND LARGE S.A.E. FOR OUR FREE '79 CATALOGUE

AS SEEN IN
P.E. AUGUST, SEPTEMBER
OCTOBER 1979

EUROPES FASTEST SELLING ONE BOARD COMPUTER — JUST CHECK THE SPEC'S.

COMPUKIT UK101

SAMPLE TAPE
WITH EXTENDED
MACHINE CODE MONITOR
AND DISSASSEMBLER
INCLUDED FREE

LOW COST SUPERBOARD IN KIT FORM

The Compukit UK101 has everything a one board 'superboard' should have.

- ★ Uses ultra-powerful 6502 microprocessor.
- ★ 50Hz Frame refresh for steady clear picture (U.S.A. products with 60Hz frame refresh always results in jittery displays)
- ★ 48 chars by 16 lines — 1K memory mapped video system providing high speed access to screen display enabling animated games and graphs.
- ★ Extensive 256 character set which includes full upper and lower case alphanumerics, Greek symbols for mathematical constants and numerous graphic characters enabling you to form almost any shape you desire anywhere on the screen.
- ★ Video output and UHF Highgrade modulator (8Mz Bandwidth) which connects direct to the aerial socket of your T.V. Channel 36 UHF.
- ★ Fully stabilised 5V power supply including transformer on board.
- ★ Standard KANSAS city tape interface providing high reliability program storage — use on any standard domestic tape or cassette recorder.
- ★ 4K user RAM expandable to 8K on board £49 extra.
- ★ 40 line expansion interface socket on board for attachment of extender card containing 24K RAM and disk controller. (Ohio Scientific compatible).
- ★ 6502 machine code accessible through powerful 2K machine code monitor on board.
- ★ High quality thru plated P.C.B. with all I.C.'s mounted on sockets.
- ★ Professional 52 Key keyboard in 3 colours — software polled meaning that all debouncing and key decoding done in software.

COMMANDS

CONT LIST NEW NULL RUN

STATEMENTS

CLEAR	DATA	DEF	DIM	END	FOR
GOTO	GOSUB	IF..GOTO	IF..THEN	INPUT	LET
NEXT	ON..GOTO	ON..GOSUB	POKE	PRINT	READ
REM	RESTORE	RETURN	STOP		

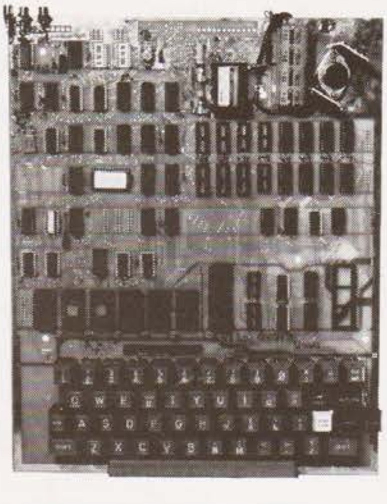
EXPRESSIONS

OPERATORS

-, +, *, /, ↑ NOT, AND, OR, >, <, <>, >=, <= RANGE 10^{-32} to 10^{+32}

VARIABLES

A, B, C, ... Z and two letter variables.
The above can all be subscripted when used in an array. String variables use above names plus $ e.g. A$.



*8K Microsoft Basic means conversion to and from Pet, Apple and Sorcerer easy. Many compatible programs already in print.

SPECIAL CHARACTERS

@ Erases line being typed, then provides carriage return, line feed.
← Erases last character typed.
CR Carriage Return — must be at the end of each line.
: Separates statements on a line.
CONTROL/C Execution or printing of a list is interrupted at the end of a line. "BREAK IN LINE XXXX" is printed, indicating line number of next statement to be executed or printed.
CONTROL/O No outputs occur until return made to command mode. If an Input statement is encountered, either another CONTROL/O is typed, or an error occurs.
? Equivalent to PRINT

Simple Soldering due to clear and consise instructions compiled by **Dr. A.A. Berk**, BSc.PhD

NO EXTRAS NEEDED JUST HIT 'RETURN' AND GO.

Build, understand, and program your own computer for only a small outlay.

KIT ONLY **£219** + VAT
including RF Modulator & Power supply.
Absolutely no extras.

Available ready assembled and tested, ready to go for
£269 + VAT

FUNCTIONS

ABS(X)	ATN(X)	COS(X)	EXP(X)
LOG(X)	PEEK(I)	POS(I)	RND(X)
SPC(I)	SQR(X)	TAB(I)	TAN(X)
FRE(X)	INT(X)		
SGN(X)	SIN(X)		
USR(I)			

STRING FUNCTIONS

ASC(X$)	CHR$(I)	FRE(X$)	LEFT$(X$,I)
RIGHT$(X$,I)		STR$(X)	
LEN(X$)	MID$(X$,I,J)		
VAL(X$)			

EXTRAS AVAILABLE SOON

COLOUR ADD-ON enables you to choose your foreground and background colour anywhere on the screen. Flash any character on the screen at will. Full documentation and parts in kit form.

AD-A-RAM EXTENDER CARD provides up to 32K Dynamic RAM Expansion, 8 Eprom sockets for 2708's or 2716's. Parallel Port (centronics compatible) and an RS232C serial port.

WIN YOURSELF AN ANADEX DP8000 LINE PRINTER

There's never enough good software around. That's why **COMPUKIT LTD.** are sponsoring a software contest. There are 2 categories:
1) **Business and Education**
2) **Fun and Games**
One lineprinter will be awarded to the winner of each category.
Send or bring along to the address shown below the following:
1) The program on cassette in the format used by the **COMPUKIT UK101**
2) Any documentation that you have for the program (source listing not necessary)
3) This coupon signed by you accepting the rules and conditions of the competition.

RULES:
1) Entries, including documentation, must be printed by computer or typed double spaced, with your name on every page.
2) Send or bring your entries to the address shown below.
3) Entries must be received by midnight on 29/2/80, any received after this time are void.
Winners will be notified by post before 31/3/80.
4) You warrant by your signature that all programs and documentation material included is entirely your own creation, and that no rights to it have been given or sold to any other party, and you agree to allow **COMPUKIT LTD.** to use, publish, distribute, modify, and edit it as it sees fit.
5) All entries become the property of **COMPUKIT LTD.** No entries will be returned nor any questions answered regarding individual entries.
6) Judging will be by a selected panel chosen by, and including representatives of COMPUKIT LTD. Judges may assign programs to any of the categories as they see fit. Decision of the judges is final.
7) Employees of **COMPUKIT LTD**, its dealers, distributors, advertising agencies and media are not eligible to enter.

Name ______________________________

Address ______________________________

____

____

I agree to abide by the above mentioned rules.

Signature ______________________________

COMP COMPUTER COMPONENTS

14 STATION ROAD, NEW BARNET, HERTFORDSHIRE — **TEL: 01-441 2922 (Sales)**
CLOSE TO NEW BARNET BR STATION — MOORGATE LINE — **01-449 6596**
OPEN — 10am to 7pm — Monday to Saturday — **TELEX: 298755**

All Products Ex-Stock Please check availability

(Part of the Compshop Ltd. Group).