50p

Practical Computing

An ECC Publication. Volume 1 Issue 4 December [illegible]

A micro solution to a jumbo problem

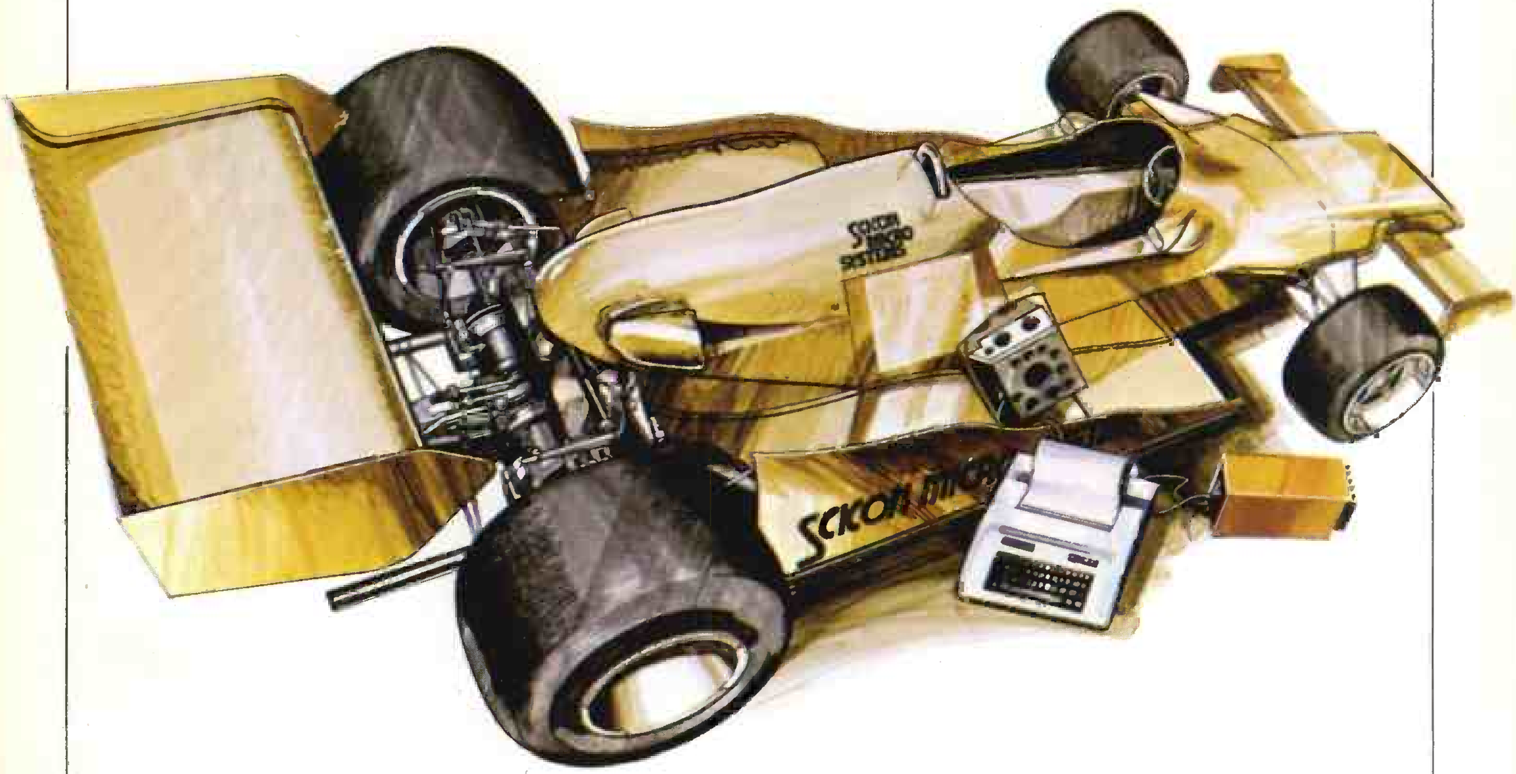
New competition: three computers must be won

Research Machines review

Choosing your first computer

Battleships, racing cars and monster games

The 200 mph micro



Imagine installing a microprocessor on a Formula 1 racing car. This was just one of the varied tasks given to Scicon's Micro Systems team. Speed, suspension movement, g-forces and chassis roll were the parameters that had to be measured. Vibration, interference, weight and temperature were just some of the problems to contend with.

Our Micro Systems team took it in their stride and came up with an innovative solution. They are also at home with commercial, industrial, scientific and military applications.

For further information about Scicon's capability in micros circle No. 101 on the free reader enquiry service. We can't offer you a drive around Brands Hatch but we can offer you fast, effective solutions to your special microcomputer requirements. Or if you like the sound of working for a team involved with advanced technology telephone or write to–

John Howes, 01-580 5599. Scicon Micro Systems, Scicon Consultancy International Limited, Sanderson House, 49-57 Berners Street, London W1P 4AQ.

● Circle No. 101

SCICON MICRO SYSTEMS

Practical Computing

Managing Editor
Dennis Jarrett

Computabits Editor
Nick Hampshire

Production Editor
Harold Mayes

Advertisement Manager
Erica Gibson

Advertisement Representative
Tom Moloney

Subscription Manager
Annabel Hunt

Publisher
Wim Hoeksma

Company Secretary
Carole Fancourt

Managing Director
Richard Hease

Editorial, Advertising and subscriptions: 01-278 9517.

Practical Computing is published by ECC as a subsidiary of WHICH COMPUTER? Ltd at its registered office, 2 Duncan Terrace, London, N1, and printed by Bournehall Press Ltd, Welwyn Garden City. Distributed by Moore Harness Ltd, 31 Corsica Street, London, N5.

ISSN 0141-5433.

Subscription rates: Single copy: 50p. Subscriptions: U.K., £6 per annum (including postage); overseas, £12 (including airmail postage).

Contents

Our Feedback columns offer readers the opportunity of bringing their computing experience and problems to the attention of others, as well as to seek our advice or to make suggestions, which we are always happy to receive. Make sure you use Feedback – it is your chance to keep in touch.

Required reading

AS AN absolute novice, I would like to ask advice. The first thing I need to discover is exactly what all the terms you use mean. I am sure your glossary will help but I would prefer not to have the information spread over a period of months. Can you suggest some publication(s) which would explain what you are writing about?

Is there also some way of obtaining information about the 'best' set-up for a particular requirement—I suppose this will come with knowledge—but where do I start? I want a computer mainly for my own personal use, for the playing of games—yes, an expensive toy—and the 'umpiring' of more complicated multi-player games. I would also like to use it for small business purposes. Could I use it also to store an index of 600-700 books and authors?

Your review of the Pet made it appear attractive but I don't know as yet to what many of the specifications refer and I have nothing with which to compare.

Ian Waugh
Wardley
Tyne & Wear

● **Our favourites among introductory books are: *An Introduction to Personal and Business Computing*, by Rodney Zaks (Sybex); *Your Home Computer*, by James White (Dynex); *Computers and Commonsense*, by Roger Hunt and John Shelley (Prentice Hall Inc.).**

The first two are American but are available widely in specialist bookshops. The third is British.

There is no easy way for us to tell you exactly what you need. We like Pet, to be sure, but it is by no means the only good, cheap personal computer. The Tandy TRS-80 reviewed in November is an obvious alternative, and it costs less.

***Understanding Basic* will give you a good introduction to the subject.**

How to buy

THE LETTER by John Miller-Kirkpatrick in *Practical Computing* in October raised some interesting points concerning the finances of U.K. importers of American equipment. For some time I have been contemplating buying some computer parts from the States, simply because the price is so low.

I have tried, without success, to discover the procedure for importing such equipment as a private individual. Several banks were unable to tell me if import duty would be payable, although one 'thought' that U.K. VAT would be due. Yet another said that I would need an import licence. Is it really cheaper to buy from U.K. importers, or to do the job yourself?

I wonder if you could find out the details of the import regulations, and print a statement letting us know the position. I for one, would be very grateful.

L. M. Newell
University of Essex
Colchester

● **Obviously a good idea; we shall look into it.**

Synopsis

MAY I say welcome to a very informative magazine for computer 'freaks'? Perhaps in the next issue, or otherwise, could you please inform me how to put together a small processing system? I should like, if possible, to construct my own rather than lay out what seems to me an extremely high price for a manufactured American system.

What we buffs need is a schematic diagram of the basic unit to which we could add, as finances permit. Failing this, a good British kit, if there is one on the market, as described by the technical director of Bywood Electronics.

But who are Bywood Electronics? Where do they operate? Is the system worth buying? Can a compatible system be made from a kit? Who supplies both calculator and microprocessor chips? How about keyboards? Will the completed unit interface with a video input on a UHF TV? Will it interface with a portable cassette tape recorder? What about power supplies?

I could go on but let us have your views and ideas on the above, by letter or by reply in the magazine. In the meantime may I wish you every success with your publication and may your circulation ever increase.

J. C. Ayres
Lowestoft
Suffolk

● **You have just written a synopsis of what we plan to cover in the forthcoming issues. We will deal with the issues you raise as quickly as possible.**

Pet for Spain?

I AM hoping to buy a computer in England next year and to bring it to Spain with me in the autumn. My present indications are that I shall go for a Pet.

I am most concerned about the reliability and servicing of the computer I buy, as there are no service facilities near me and so I am trying to find out as much as I can about the comparative merits of the various machines on the market in this respect and users' experience with them.

If you have any advice to offer I would be grateful. In particular, if you know of any company which has some link with a company in Spain, I would be most interested.

Peter Shafe
Alicante
Spain

● **The Pet has a good reputation for reliability. Most microcomputers are fairly modular in design, which means that replacement by post is feasible, so maintainability probably will not be too much of a problem if you stay with the famous-name makes. Would any manufacturers care to comment?**

War-gaming

I WORK in electronics and my hobby is war-gaming. A recent article (July/August) in *Strategy and Tactics*, War-games and the (micro) computer, made me eager to try something similar. I need to know what systems are available, and the prices, as well as which ones can be used/programmed to play war games.

I would like also to find out how to program for war-gaming and contact anyone who has already done something of this kind.

R. Meattie
MESF/EEC, RAF,
Bury St Edmunds,
Suffolk

● **Any offers?**

Which system?

I WAS delighted to read your Feedback column in *Practical Computing* as it is exactly what I, and no doubt, many others in my situation are seeking.

I wish to build my own micro system to understand the various aspects of how they work, their limitations and the like. Having built a simple system, I would like to be able to expand it as my understanding and experience allows, to finish with a useful system which could be used as a development tool for smaller dedicated systems.

Although my experience of micros so far is limited to an introductory course of evening classes at the local polytechnic, I have a background of sequential logic and am involved with a large sequential logic system for a newly-commissioned process plant.

Being virtually a complete novice to micros and clearly falling into the home computerist or hobbyist category of

(*continued on page 17*)

58-60 Northfield Road
Kings Norton
Birmingham
B30 1JH
Phone: 021-459 5959

DATA DESIGN TECHNIQUES LTD

We have the answer to your small system peripheral requirements

PRINTERS — TERMINALS		
DECwriter II	LA36	KSR
Teletype 43	KSR	RO
Tally 1612	KSR	RO
Tally 1202		Parallel

PLUS

A wide choice of reconditioned printers available at rock-bottom prices.

MAINTENANCE SUPPORT

A full range of service, including on site and base workshops. For all Micros and peripherals, our engineers will integrate, build and debug your systems.

CALL YOUR NEAREST OFFICE and DISCUSS your REQUIREMENTS

DDT MAINTENANCE LTD

12 Leeming Road
Borehamwood
Herts
WD6 4DU
Phone: 01-207 1717

5-6 Lower Church Street
Chepstow
Gwent
NP6 5HJ
Phone: 02912 2193

SEE US AT COMPEC

● Circle No. 118

PETALECT

ELECTRONIC SERVICING LTD.

WANTED

Good Homes for Intelligent Pets

THE

PET 2001 Computer

£643·52 plus VAT

This unbelievably versatile, compact, portable and self-contained unit has many varied applications and offers tremendous benefits in the worlds of

- **BUSINESS and COMMERCE:** Can be used efficiently for Trend Analysis · Stock Control · Payroll · Invoicing · Inventory Control, etc.
- **SCIENCE and INDUSTRY:** The 'PET' has a comprehensive set of scientific functions useful to scientists, engineers and industry.
- **EDUCATION:** An ideal tool for teaching and it can be used to keep records, exam results, attendance figures, etc.
- **ENTERTAINMENT:** Games including Backgammon, Noughts and Crosses, Pontoon, Black Jack and Moon Landing

Possesses all usual alphanumerics PLUS 64 graphic characters for plots, artwork, etc., a printer, 2nd cassette deck and software available AND IN THE NEAR FUTURE 'Floppy Disc' data and programme storage system.

We have six years' experience in servicing electronic calculators, mini-computers in S.E. England. 24-hour service contract available at £69·50 per annum. Credit and leasing terms available.

For full details and demonstration contact Peter Watts . . . Now!

PETALECT

ELECTRONIC SERVICING LTD

(Authorised Commodore Pet Dealer) Specialists in Electronic Servicing, Programming, Electronic Design and Prototype Manufacture

33 PORTUGAL ROAD, WOKING, SURREY GU21 5JE.

Tel: Woking (04862) 69032/68497

● Circle No. 119

(*continued from page 15*)
(potential) users, my initial requirement is information on Which type of system to acquire:

I have been advised to start with a kit, to appreciate the hardware aspect, and also for the technical back-up I may require during the initial stages;

The minimum practical amount of test gear/tools which would be required;

Cost and availability.

Any recommendations you may care to make would be gratefully received.

P. Walton
Yarm
Cleveland

Club call

WHILE reading the first edition of *Practical Computing*, I read an article mentioning the Amateur Computer Club. Could you tell me how I may contact this club, with a view to joining or corresponding with members having interests similar to my own?

Ingrid Prince
Edgbaston
Birmingham

● **Our contact at the Amateur Computer Club is Mike Lord and he is at 0268 411125. May we also suggest the Midlands Amateur Computer Club, based at 27 Loweswater Road, Coventry? Our contact there is Roy Diamond (Coventry 454061).**

In last month's issue, by the way, we published a list of user groups.

Mailing programs

I READ with great interest the letter from G. Myles, London SW6 as I, too, run a chartered surveyors and estate agents and would be pleased if you could send me a selection of names of firms who run mailing programmes and the machines to which they are appropriate. As with Mr Miles, my budget is about £1,000.

C. E. P. Darley

● **We have sent some to Mr Darley. Anyone with mailing programs, please write to us.**

Mersey Z-men

WITH reference to your user group listings in the November issue, we have formed a sub-group of the STEM minimicro club for users of 380Z/280Z systems in Liverpool.

Could you include us in your listings to help people with systems in the Merseyside area to be able to get in touch?

It would be nice to see some Basic programs in your magazine, especially in a subset of Basic which could be used by many types of systems which run various types of Basics.

Alan Pope
PAAL Enterprise
Crosby
Liverpool

Hire service

DO YOU know of any firm which hires microcomputers like the Pet, Research Machines 380Z or Tandy on a short-term basis? Our school would eventually like to purchase but cannot do so until next year.

Mrs. J. K. Cameron
Northgate High School
Ipswich, Suffolk

● **CCS Microhire (046 26 73301) specialises in very short-term rentals. For longer-term periods you could try Hamilton Rentals (01-739 3444) or a new company, MBS Rentals (093 23 49511).**

THE following letters are typical of a mixed bag of queries about buying systems:

Manx appeal

VERY SHORTLY I hope that my school will be allocated some money to buy a computer. It is difficult for me, from my isolated position in the Isle of Man, to decide what should be ordered. So I would appreciate any advice that you can give me.

The school is a comprehensive, with pupils aged 11 to 18, in a mainly rural area; it has a largely traditional background.

At present, we do a little computing with the sixth form, using punched cards, which we send to Imperial College, London. The language used is Fortran 4. I should like to expand computing and run a course perhaps to O level standard or further, and I can see that some or our university-bound sixth formers will want to make considerable use of the computer.

I should also like to establish links with the science department and I hope the rest of the school might notice the benefit of having a computer.

The recognised schools' language seems to be Basic and microprocessors or microcomputers have been recommended to me by the schools inspector.

I should like some practical advice from people who have some knowledge of microcomputers, or better still, have used them.

It is suggested that up to £1,000 may be available to buy a microcomputer, with possibly more money available in the near future.

I hope that you will give me the benefit of your knowledge, as I would very much like to know which microcomputer would be most suitable for our use.

P. Kinnish
Ramsey
Isle of Man

A-level choice

I AM soon to make my choice for A levels and I hope later to take a university degree. The region of computing I am hoping to head for is hardware design. I can't seem to find much information on this subject, I don't know what subject I have to take at A level, and I can't find anything about university courses in this field.

Please could you help me to get to know more about what I have to do to go into this area?

I am also thinking of buying a mini-computer kit which can be expanded into a comprehensive system. I have in mind at the moment the Kim 1 system. Please could you give some advice about this, too, and whether it is the right system? It will have to be on a budget of about £200.

I know this is a lot to find out but I would be very grateful for all the advice you can give me.

David Griffin
London NW10

No jargon man

AS A small business man I am interested in computers as I feel that this would reduce my workload on wages, stock control, accounting and VAT.

As I have no knowledge whatsoever I am confused with the computer jargon. Perhaps you could suggest a booklet explaining in plain English what on earth is 32K of a 64K memory, or the difference between a ROM and a RAM, byte, I/O port, and the like.

As indicated the computer would in the first place be required for business purposes and, of course, the cost is of importance. I thought perhaps a Pet, Kim, Nascom 1 or Sym 1 would be in my range, with the opportunity of expanding into print-out at a later stage.

I should be pleased to receive your advice and guidance. I should like to mention that I have a portable TV and tape recorder, should this help to reduce costs.

J. H. Gibb
Barford
Warwickshire

Small budget

I SHOULD be grateful if you would please tell me of any build-it-yourself computers which could be built over a period of about 2–3 years from component pads. This time limit will have to be so, since I do not have much of a budget for computers at the age of 16. I hope you can oblige. If so, you've started a computer fanatic on his way.

Andrew White
Co. Armagh
Northern Ireland

● ***Practical Computing*** **will be examinng how you can choose your first computer in forthcoming issues. In this issue we detail what to look for from the point of view of a complete novice. There is a Buyers' Guide which we shall be printing in each issue, detailing equipment available, the price and typical applications.**

A computerised reservations system for less than £5,000 sounds like an impossibility. Yet it is something the British Universities North America Club has achieved.

Micro system tackles Jumbo job

BUNAC, the British Universities North America Club, is a non profit-making organisation which arranges charter flights and working holidays to North America for British students. It is run largely by voluntary helpers, most of whom are involved in running the 52 BUNAC clubs on university campuses, but has a permanent staff of seven at its headquarters just off Tottenham Court Road, London.

One of the main tasks of the permanent staff is to process the 2,500 flight reservations they receive each year—a trivial number by British Airways standards, of course, but a different matter for a small staff doing everything by hand.

Until recently, the process involved 11 operations for each passenger, starting with the printing of a metal addressing plate with all relevant details, including that required by the Civil Aviation Authority. In addition, each passenger is contacted in writing five times in the course of dealing with his booking.

Overwhelmed by paperwork

The result was that, at peak periods, the level of paperwork overwhelmed the office. The system worked reasonably well up to about 1,500 flights a year. Above that level, the number of misfilings of records increased and began to create something of a chain reaction.

There were also occasional disasters, such as the time the complete filing system was spilt on the floor.

It was a clear case for computerisation but even so, the BUNAC U.K. general manager, Jim Buck, says: "Conceptually, we wouldn't have come within light years of computers ourselves. Shortage of funds and total ignorance of computing made the idea virtually unthinkable".

Illumination spread first from a television feature on Tomorrow's World and an article in a Sunday newspaper, detailing computers for as little as £2,000. Impressed, Buck telephoned the BBC to ask for the names of the manufacturer, and it proved to be SWTPC. Then he began searching to see what else was available.

Tracking-down micros in October, 1977 proved to be tricky, especially for someone with little idea of what he was seeking. Buck never discovered who were the other manufacturers. The Business Efficiency Exhibition was full of equipment which was fast, glamorous and very expensive, but nothing was within £1,000 of the cost of the SWTPC hardware.

The process of shopping around, however, provided BUNAC staff with a better insight into the potential of computers. From thinking in terms of little more than a glorified addressing machine, they appreciated other advantages.

Most important of them was security of information, a factor which Buck stresses repeatedly in discussing the system. The idea that information, once entered correctly into the computer, could be guaranteed accurate, and that it could be guaranteed not to be lost, was a revelation. The first time the computer produced a passenger list without a single mistake was a milestone. "Nobody ever had passenger lists as accurate as that", says Buck.

The BUNAC system; 32K processor, twin FD8 discs, printer and VDU.

Mental anguish factor

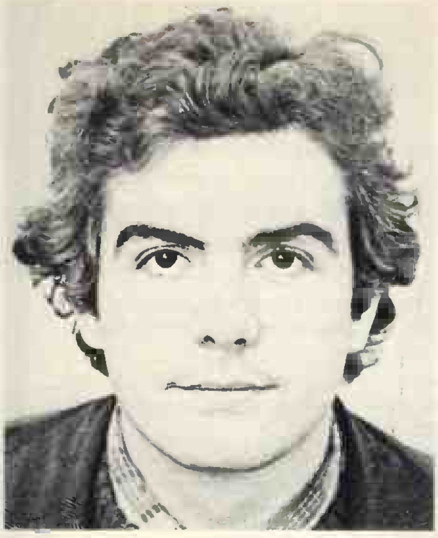
Financial and staff savings, however, were a secondary consideration. BUNAC was able to hire one fewer temporary staff during the three peak months and handled more flights than in the previous year. More important, Buck says, is the saving in "mental anguish" and the long hours of overtime for BUNAC permanent staff.

The decision to buy the computer was taken in November, 1977 and the machine was delivered in February. Systems design and programming was carried out by Tim Beyts of Beyts Logic, to whom BUNAC was introduced by SWTPC. After evaluating BUNAC requirements, Beyts specified a 32K system, with a VDU, twin FD8 disc drives and a Centronics 701 printer. The cost, including software, was about £4,750.

Meanwhile, Buck rushed to buy a Basic manual. He settled on Jerald Brown's *Instant Basic* and recommends it to others who are baffled by the mathematical aspects, which he feels are overemphasised in most manuals.

As a result, he was able soon to provide

(continued on next page)



Tim Beyts, who wrote the software.

(continued from previous page)

a rough specification of his requirements, in terms of input and output, amendment, enquiry and reporting facilities. The system was operational by March 15.

Data, as before, is taken from passenger reservations and output in the form of booking details in CAA format, passenger lists, address labels and analyses of total bookings. Accounting is reasonably simple, since most charges are fixed and common to all passengers, and was omitted from the original design, though it will probably be added next year.

Data errors are negligible

The passenger file builds up over the season to create a bulky database, occupying three discs. So even a short enquiry or amendment run can take 40 minutes but this is not regarded as a great handicap—the staff proceed with another task.

At present, three of the seven staff have learned to use the computer, though an operator manual is being prepared which will enable anyone, including temporary staff, to carry-out data entry. Data errors on input so far have been negligible. Apart from the built-in validation routines, Buck attributes that to a new-found pride in accuracy which the computer seems to have generated. He speaks of "a small sense of achievement" as each entry is input correctly.

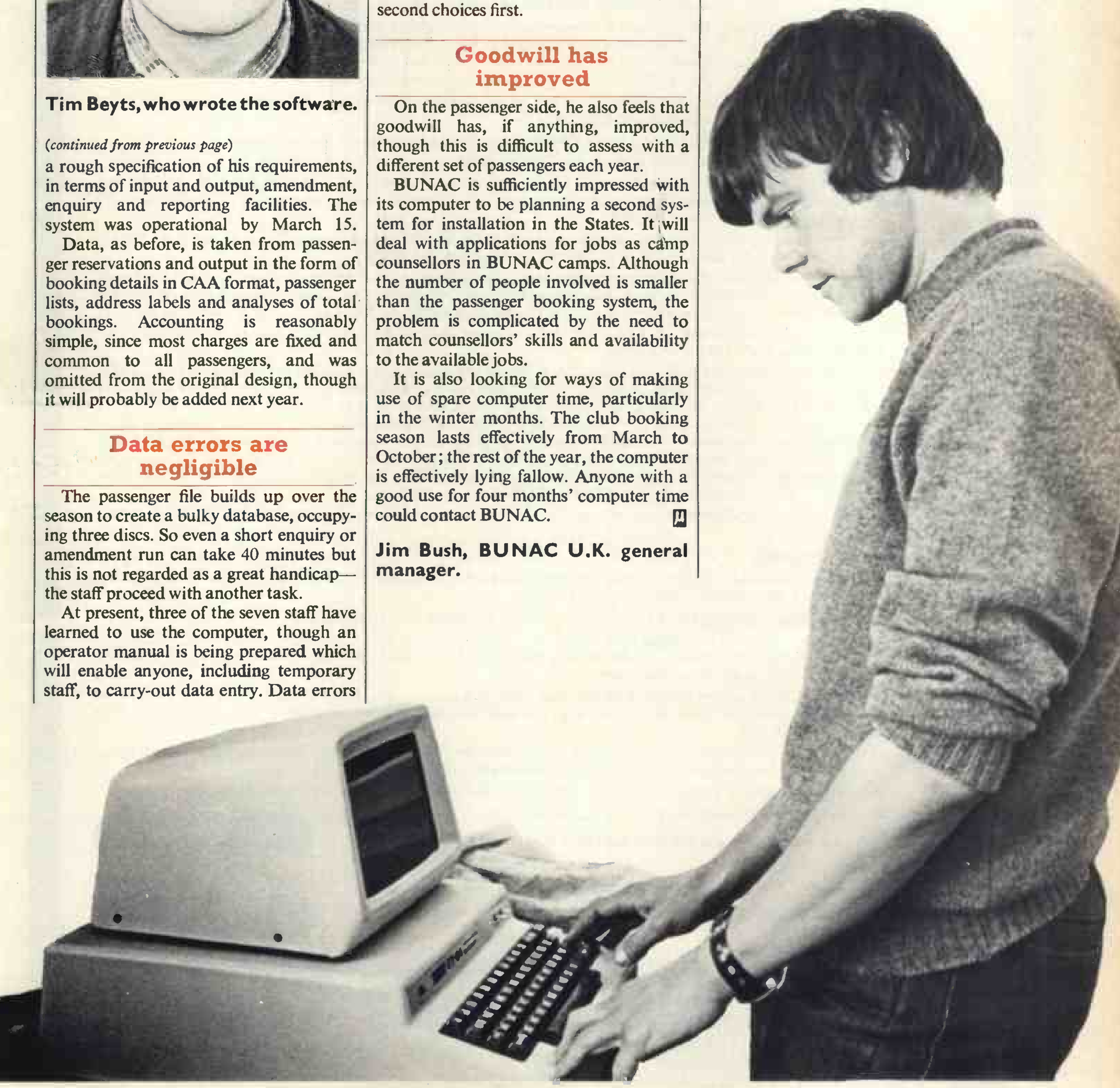
He is also relieved to find that the computer has not deprived him of his "feel" for the way the business is working. Partly because he still deals with much of the data entry himself, he finds he is, for example, still able to decide which would be the best alternative date for a flight which has to be altered, without having to run through the list of passengers' second choices first.

Goodwill has improved

On the passenger side, he also feels that goodwill has, if anything, improved, though this is difficult to assess with a different set of passengers each year.

BUNAC is sufficiently impressed with its computer to be planning a second system for installation in the States. It will deal with applications for jobs as camp counsellors in BUNAC camps. Although the number of people involved is smaller than the passenger booking system, the problem is complicated by the need to match counsellors' skills and availability to the available jobs.

It is also looking for ways of making use of spare computer time, particularly in the winter months. The club booking season lasts effectively from March to October; the rest of the year, the computer is effectively lying fallow. Anyone with a good use for four months' computer time could contact BUNAC.



Jim Bush, BUNAC U.K. general manager.

Pascal is rivalling Basic

PASCAL seems to be rivalling Basic as the favourite programming language of the computer aficionado. At any rate, the U.S. hobbyist magazines are all full of it and so was our last issue. The argument is advanced that PASCAL is more flexible, more powerful across a good range of applications, and generally easier to use.

Not too many systems available here have the language, but Equinox is making it available with its 300. This is the top end of the Equinox line, differing from the rest of the family of S-100 bus systems by incorporating a 16-bit processor as the Western Digital IDP-16 chip set, functionally similar to the LST-11.

Equinox offers a 64K system with 600KB floppy disc for about £5,000—upgrade goodies include Calcomp hard discs from 10 to 300MB. The price includes all system software, which means the languages are bundled. As well as the new PASCAL, there is a compiler for extended Basic and the mathematical language LISP. Fortran is promised next year.

Equinox will also be offering FORTH on the 300 in 1979. We have commented before on this operating system and embedded programming language for microprocessor development systems, but it is now available, and apparently fully-supported, in the U.K.

MicroFORTH is, predictably enough, the micro version; there is also a MiniFORTH. The micro implementation gives all the system software you need, including a text editor, in less than 6KB. It has a floppy disc plus documentation and already there are versions for most of the more popular prototyping hardware packages. Prices range between \$2,500 and \$3,500

Is it worth it? Well, Forth Inc quotes impressive savings on program development time. Using an Assembler is reckoned to take nearly 10 times as long; PL/M weighs in with 6·5 times; and Basic systems take about 4·5 times as long. The memory overhead is significantly lower with FORTH, too.

One thing more—part of the purchase is "access to the MicroFORTH telephone Hotline" which provides "immediate" help in answering queries. You have to wait until after 4 pm, of course, to give the Californians time to put away their wheaties but this sounds like a really useful standby.

My Practical Computing week

I BEGIN on Monday
my own very first
practical home computing week.

Washing away my bleary-eyed blues
with solders and patches and soft kind of glues,
I fixed me the bodywork in which I could drive
data down buses like bees to a hive.

Tuesday dawned slowly roundabout noon
and by the end of the day I was over the moon,
because I'd practically knitted and crotchetted and sewed
a cosily-structured basic utility softwear workload.

Wednesday was friends' day, the communications trends day
where after surprise presents switching packets play
we settled to sherry drinking chat, both bitty and baudy.

Multimillipede bugs, with bells on each ankle, swarmed the next morning
and joy began passing, the problems were dawning
as with a thunderous crash fell my very own personal computing world
and into dumping, debugging, depression, despair I was head-longingly hurled.

Black coffee-hazy towards Friday I sped,
turned tail on the bugs and to my allotment I fled
for some practical gardening while munching wafers and chips,
I tended my loopins and with my nerves came to grips.

Calmed and relaxed (Chelsea had won)
I darned and I patched and it all became fun,
playing Star Wars and micro-based railways,
I'd found my Nirvana for all my unemployed days.

Next morning in bed I read the Sunday viewpapers
like the Prestel Express and the News of the Telecapers
then turned to my automatic, all-purpose-built woman
knowing now why I'm a practical computing leisure society fan.

Malcolm Peltu
Editor, Computer Weekly.

Bookshop

RESIDENTS in the Birmingham area not already aware of its existence should hasten to the Computer Bookshop, where books on all aspects of computing and microprocessors await perusal.

They range from books for the raw newcomer, to games, to the finer points of microprocessor programming.

The shop is at Temple House, 43–48 New Street, Birmingham B2 4LA.

Graffiti competition

WINNER of this month's Graffiti competition is R. W. Stranks of Cheltenham, Glos., who will be receiving a calculator. The fact that he sent six entries had nothing to do with his success. Some of his offerings are:

There is a young fellow called Snow
Whose micro we can't get to go
The contents of the Stack
We never get back
Just FF, FA and FO.

* * *

"*Abandon hope all ye that enter here*" *Rem. statement above the 'you're lost' routine, from OTHELLO*

* * *

A gallant programmer from Tooting
Took a fancy to Practical Computing
His program for VAT
Was far better than that
Anyone else had been mooting.

A Z80 phreak name of Smiler
Invented the Basic compiler
When he turned the thing on
It burst into song
And its printout was never worthwhiler.

* * *

Programmers delighting in chess
Created one hell of a mess
They set up a grapple
Twixt a PET and an APPLE
With a result that was anyone's guess.

Second best was from R. J. Fiddik of West Loo, Cornwall: "*It is well worth putting in your memory that all Apples do not have cores*".

Third, from M. Collins of Letchworth, Herts: "*All loads come from ROM*".

Entries for next month's Great Graffiti should be received by December 31, 1978. Address them to Great Graffiti Competition, *Practical Computing*, 2 Duncan Terrace, London, N1.

Evaluation aids

THE growing range of RCA evaluations aids for understanding microcomputer systems has been expanded by the addition of a new assembler/editor design system. Based on the RCA CDP1800 family of microprocessors, the kit enables the user to develop assembler-language programs for his micro system.

The resident editor program allows standard text editing, including the addition or deletion of characters, words or complete lines. The resident assembler translates mnemonic commands into machine code and generates any necessary error messages.

A 4K read–write memory is supplied with the kit, enough to hold the editor program and a working buffer of 1K bytes, or to hold the assembler with enough storage for the assembler and 100 labels.

Byte stake for Charterhouse

THE £100,000 Charterhouse investment in The Byte Shop probably will not signal a general rush on the part of City institutions to put money into the glorious and profitable future of microcomputers, but at least it demonstrates that someone under the shadow of St Paul's is taking an interest in the computer business.

Charterhouse is no stranger to the idea, of course; it has money in CAP. CAP, though, is a fairly conservative home for your cash by computing standards, and The Byte Shop is a very different kettle of chips.

The Byte Shop Ltd is one shop in Gants Hill selling home and hobby micros. It has no connection with the U.S. Byte Shop chain, incidentally, nor with *Byte* magazine. It began as a spin-off from Computer Aided Systems, a South London turnkey system supplier which still packages minis and delivers them with working applications software.

The Byte Shop has ambitions, though, with "national coverage" promised by the end of 1980 and five more shops due to open soon as a start.

The Charterhouse investment is in the form of a 25 percent stake by one of the group's venture capital subsidiaries. Also taking 25 percent is United Electronic Holdings, a new electronic distribution company in which Charterhouse has a substantial holding. The package involves a total of £100,000.

That leaves Bill Cannings, founder of CAS and The Byte Shop, with half the company, but since he risked his existing operation to get the micro business under way, and since a bank manager proved to be distinctly frosty when approached, Cannings is bound to be pleased about the outside commitment.

Musical effects

ARE YOU disappointed with the mute responses of your VDU, or irritated by the chattering of your output printer? We know how you feel. So it's nice to know General Instruments Microelectrics has taken to heart your plight and produced two devices to enable you to obtain more interesting sounds from your micro system.

They are two new peripheral microcircuits which, under microprocessor control, can be made to emit a wide range of complex sounds.

Program Sound Generators have applications in entertainment, education and security. They can produce sounds like a musical instrument, sound effects for electronic games, or warning sounds for security systems.

Anyone interested should telephone 01-439 7052 so that they can hear an automatic demonstration.

Hard-copy for your Pet

IF YOU criticise the Commodore Pet, you will probably be complaining about the lack of a hard-copy output device. Now you will not have that excuse—GR Electronics is supplying Pet systems complete with printers and they are high-quality units with an excellent pedigree. They are IBM 3982s, which are like a golf-ball typewriter without the keyboard.

GR has had second-hand units renovated by a specialist who will also look after maintenance of the printers. According to GR's Doug Watkins, they should cost less than £500 each. Which means that you'll be able to obtain a Pet system with a letter-quality printer for less than £1,100, and that's a system which will produce business system with word processor capabilities.

Not wanting to rush anybody, but there are only so many used IBM 3982s around.

It is about time those refurbished IBM typewriters started reaching the U.K. market. American hobbyists have been able for some time to attach them to microcomputers.

To be fair, the excellent print quality and the good pricing of the GR/IBM printer is achieved with a fairly slow performance by comparison with matrix printers. Commodore's 80cps matrix printer is due to reach the U.S. market now, though in the U.K. deliveries probably will not begin in earnest until the New Year.

American dealers have the printer at around $700. An alternative already available here is the Teletype 43, with 30cps and many fans, and now a Pet interface by courtesy of Peripheral Hardware Ltd.

If you're tired of waiting for the Commodore floppy disc system to add to your Pet, there's one appearing from a new company, Midland Micronics Ltd, in Solihull. It drives one or two minifloppies and it is complete with a PROM which you plug into the Pet board.

The complete bundle, including a twin disc drive costs £1,300; for a single disc drive, it costs £870. Deliveries should be starting now, and you'll be able to obtain the kit through the Pet dealer network, we're told.

Managing with micros

A ONE-DAY conference to explore the potential impact of microprocessors on the process of management is to be held on January 10 at St. Albans, Herts. It is sponsored jointly by *Practical Computing* and Eastern Counties Operational Research Society. The types of questions the conference will be attempting to answer are:

- How can OR use microprocessors in solving complex management decision problems?
- What new management tasks will microprocessors perform and what tasks will they take over from office workers, secretaries, mini computers and mainframes?
- What effects will there be on management communications? How will microprocessors link to communication networks?
- How will management interact with the shop-floor microprocessor?
- How far will computerisation go? Will every small business be using a microprocessor and every executive have one at home?

The conference will include a session on the "office of the future" by Logica Ltd. A display of microprocessor and word processing equipment will be included. Numbers are limited to 120, so apply early. Tickets cost £5 each. Write for tickets and further information to:

Ian Roderick, 237 Lonsdale Road, Stevenage, Herts., enclosing payment. Cheques should be made payable to Eastern Counties Operational Research Society.

Cornish link-up

A GROUP of Cornish computer enthusiasts has set up a company in Penzance to sell computer equipment and services.

Factor One's first product is a micro-based system for a firm of local accountants. It is based on the S-100 bus and a 280A drive and can handle accounts, payroll and administration.

The firm had made contact with some U.S. suppliers and is offering the Sorcerer computer from Exidy Inc in the U.S., reviewed in this issue.

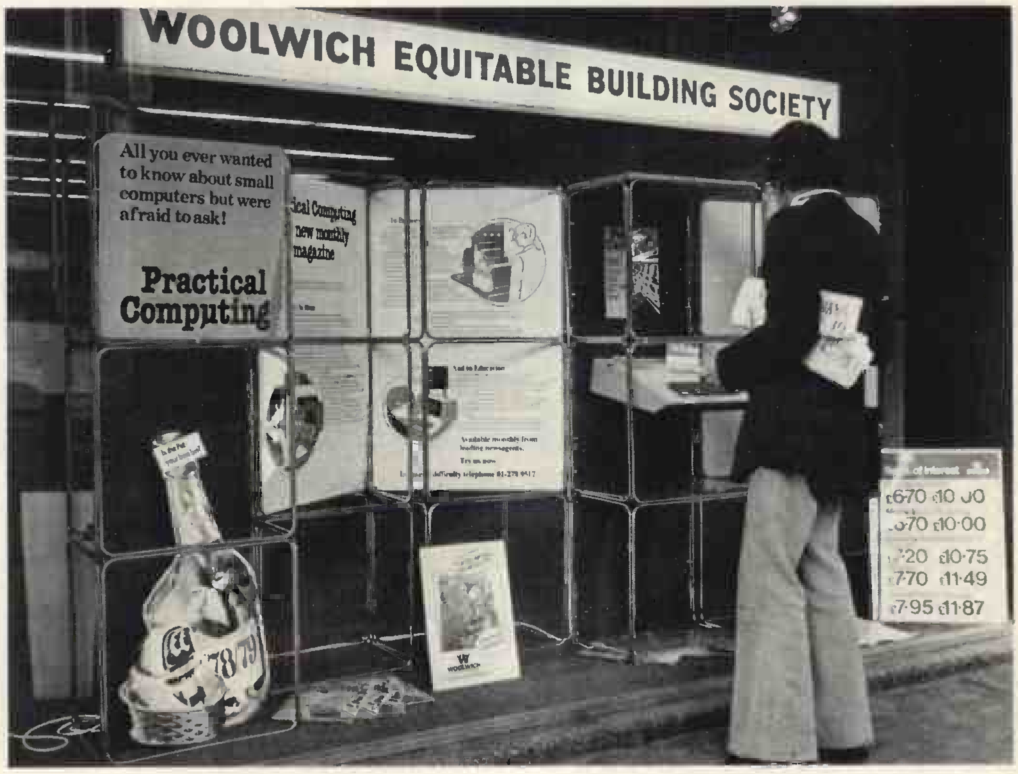
In addition, Factor One is building a business of supplying parts and accessories for micros. It is representing the Jade Corporation and has an English version of the Jade catalogue which details components, complete computers and peripheral equipment.

Copies can be obtained from Factor One at 17 Market Place, Penzance, Cornwall.

We're with the Woolwich

Practical Computing is starting a tour of the top Woolwich Building Society windows, detailing everything you want to know about micros but have been afraid to ask.

The first window featuring _Practical Computing_ is at the Woolwich Building Soeiety in Victoria Street, London SW1, opposite Westminster Cathedral.



SORCERER—IT'S MAGIC

A HOBBY computer just coming to the market in the U.K. is the Sorcerer. It is made by the American firm, Exidy Inc, a video games manufacturer. First shipments are being delivered to two firms, Comp Computer Components of New Barnet and Factor One Computers of Penzance.

It is a machine we intend to review in more detail later. Comp Computer Components, however, loaned us its demonstration system briefly to try its facilities and it is impressive.

The main selling point of the Sorcerer is price. You can buy a 16K version for £760 or a 32K version for £950 and, considering its facilities, this represents good value for money.

The system loaned to us was the 32K version. Standard configuration includes a 61-key typewriter-type keyboard and 16-key pad. It looks like the Tandy keyboard without the numeric pad. To that you add your own power supply. TV monitor and cassette tape recorder(s).

Striking

For expansion purposes it takes the S-100 bus, which gives you the ability to interconnect large memories, disc drives, speech and communications facilities.

The striking feature of the Sorcerer, though, is the way you load the Basic. It's a standard Basic which is loaded by way of a cartridge into the side of the keyboard. It looks rather like an 8-track stereo cartridge but inside is a read-only memory device which stores the language, which does not have to be Basic.

In addition to standard Basic is Assembly, APL, Pilot, DOS (Fortran and Cobol); even a word processing package is under development.

Unfortunately, only the Basic was available with our system and it would be unfair to review the system in great detail without being able to use the other languages or, indeed, any of the packaged software which has yet to find its way to the U.K.

The Basic, in our opinion, is somewhat limited, although the graphics and keyboard facilities, as well as the potential to add other languages, compensates for this. (See table, next page, for Basic commands and statements).

Upper- and lower-case alphanumerics are included in the 128 pre-defined and fixed character set, so the keyboard is truly typewriter style and can be used easily for letter writing or text editing. Thirty lines of text, amounting to 1,920 characters, are visible at any time before automatic scrolling to the next page. There are 64 defined graphics characters and 64 user-defined characters; alternatively, all 128 graphic characters may be user-defined. (See technical specifications, next page).

Two manuals

There are two manuals with the systems—*A Guided Tour of Personal Computing* and *A Short Tour of Basic*.

The first, which takes you through system set-up and introduces you to all of the machine facilities, is written in a typically American style, wasting no time on explanations but bringing you right into the operation of the computer. It's breezy and, in a way, easy to read but not quite as easy to understand as the Tandy TRS 80 manual.

The second manual welcomes you to Basic and is excellent. It is very easy to understand and within a couple of weeks of evening learning, the novice should be writing Basic programs and spending hours debugging them.

As an aside, it is a pity most of the manuals we have read do not concentrate more on telling you how to develop a system. It is as if to know Basic is enough. What is needed is a simple systems analysis book to accompany these manuals, instructing the user how to develop a system and then to program it. The flowchart in a manual is not good enough.

It will be interesting to undertake a detailed review of the capabilities of Sorcerer

(continued on next page)

(continued from previous page)
when all the software and language facilities of the system are available.

Comp Computer Components and Factor One Computers say that there is a great interest being shown in this machine, so it should not be long before we will be able to test it in greater detail.

Standard Basic Commands and Statements

Statements

Let	Rem	Fn	Wait
Then	Stop	Data	On..GoTo
GoTo	Def	Dim	If..GoTo
Restore	Print	GoSub	On..GOSub
Return	Next	Input	If...gosub
Read	Get	Out	
For	Step	End	

Commands

Run	Null
List	CSave
New	CLoad
Clear	Peek
Cont	Poke

Mathematical Functions

Abs	Int	Sgn	Tan
Atn	Inp	Sin	Usr
Cos	Log	Sqr	And
Exp	Pos	Spc	Or
Fre	Rnd	Tab	Not

String Functions

Asc	Str$
Len	Mid$
Val	Left$
Chr$	Rights$

Operators

= — / / < = >
\+ * ..<w< = >

Editing

.. Control C ?
.. Control 3 : Return

Specifications

Processor: Z80
Processor clock: 2·1 MHz
Serial I/O: RS232, 300 or 1,200 baud, 25-pin "D" type connector.
Parallel I/O: 8-bit input and output, latched and buffered port with hand-shaking, 25-pin "D" connector.
Memory: Read-only memory (ROM) 4K byte. Power-on monitor program connection for 16K ROM PAC program cartridge. Random access memory (RAM) for 8K bytes expandable to 32K byte.
Expansion: Edge-card connection to S-100 bus expansion unit.
Cassette I/O: Dual recorders, 300 or 1,200 baud data transfer rate, remote control of motor on/off.
Video I/O: 30 lines of 64 characters or 1,920 characters full screen, 128 full ASCII character set. 64 defined graphic characters and 64 user-defined characters; alternatively all 128 graphic characters may be user-defined. 512 (hor.) × 240 (vert.) graphic resolution.
Automatic scroll, erase end of line and end of screen, delete character, erase screen.
Cursor Home, Up, Down, Left, Right.
Cabinet: Dimensions 19·25in. × 13in. × 4in., weight 13 lb.
ROM PAC Cartridges: (Standard Basic included)
Media: 8-track cartridge enclosure, read-only memory on PC board.
Programs: Standard Basic Assembly, Pilot, APL, DOS (Fortran, Cobol).
Cassette Tape
Media: Standard Phillips cassettes
Programs: Casino, Personal Physician, Personal Data Management, Management Aids, Computer Aided Instruction, Advance Engineering.

Keenness at Poly

IF ANYONE doubted the quality of interest in home and hobby computers in North London, the inaugural meeting of the North London Hobby Computer Club would have set them straight. The organisers say 400 people attended, which seemed a little on the high side to us, but it was certainly more than 300, and that astonished us.

There was a strike at the Poly of North London, which meant a cold theatre and no amplification for the hardy. Pets and SWTP systems were on show but most were dead. The introductory speeches were inevitably a trifle turgid, though the two "respected industry figures" invited to share the platform did well—they were from *Practical Computing* and the different but equally excellent magazine *ETI*.

Those present seemed keen, filling-in their questionnaires without a murmur and clustering around the half-dozen 'interest groups' the club proposes to run.

Membership will cost them £10, for which they have some use of the Poly facilities—Poly electronics and business departments are the moving spirits of the venture.

For you . . .

IN next month's packed issue of *Pratical Computing*:

How to convert an IBM typewriter into an output terminal.

Installing a word processing system on a Nascom.

Why Panther is putting a Pet in its car.

We review Nascom.

More software business packages.

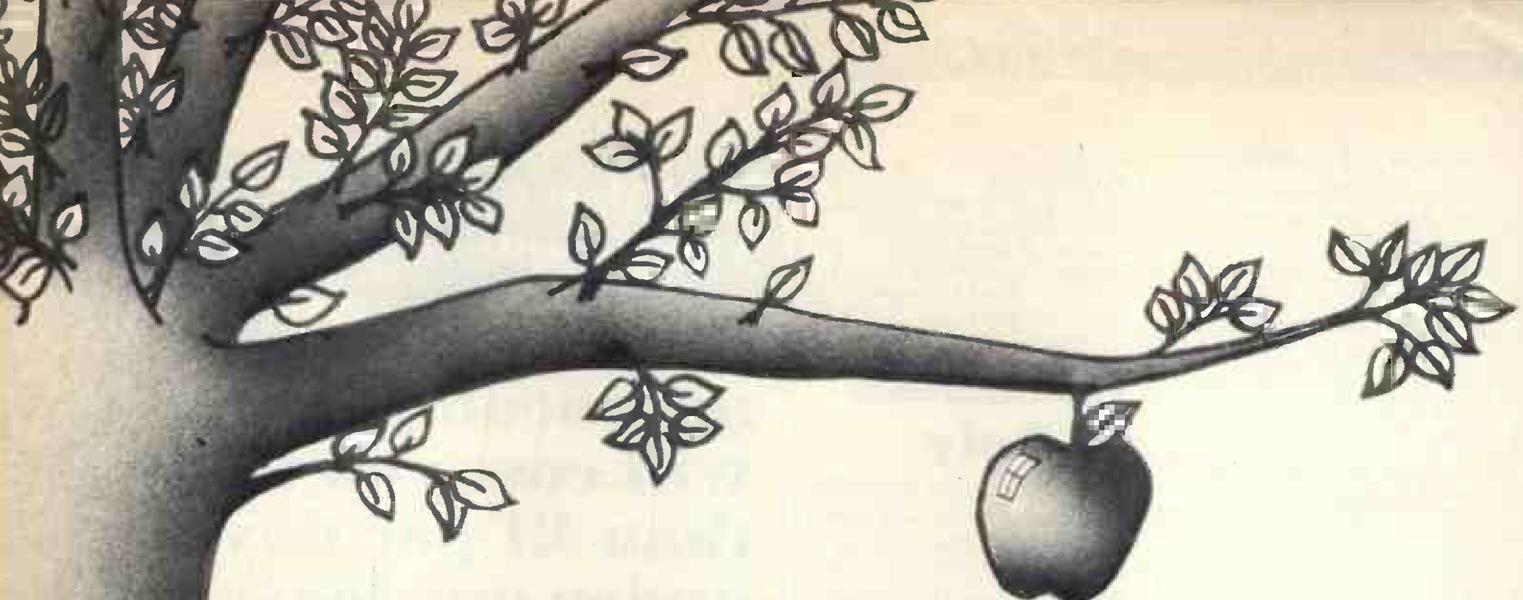
Low-cost printers.

A function generator for the Kim.

Plus Games, Glossary, Printout, Letters and much, much more.

At all good newsagents from December 15, 1978.

Subscription details 01-278 9617.



Research Machines 380Z micro

by Martin Collins

MOST manufacturers tell us that their computer can be used for anything. "You want a system for the home, small business, education? Look no further. Ours is the one you want".

It was most refreshing, therefore, when Research Machines approached us and said: "Our system is sold for the most part to the education and scientific market".

Knowing a system is aimed at a specific market also makes it easier to review, as we can evaluate it in the light of the requirements of the potential users in that market.

That said, it is a pity Research Machines' system is not a little more commercialised. A few games packages and some business applications would provide a powerful system, and we would be back on the roundabout.

Research Machines Ltd is characterised best by its lack of packages. The company, formed in 1973 and operating in Oxford, is a British company.

Very tough

The system supplied for review was a 32K 380Z with keyboard, monitor and, cassette recorder. We were supplied also with a set of documentation and a number of program tapes.

I asked RML if I could take home the system for the weekend if I promised to keep the children from it. "Don't worry about the children", they said, "it's very tough, they will do no harm".

It certainly seems, physically, to be the most robust system we have yet reviewed. One disappointment for the children as well was that RML supplies no games. If you form the impression we like playing games, you are correct.

The system was supplied with a 13A extension box in addition to all the plugs and cables and we had no problems in setting up. As it is in four separate units—system chassis, keyboard, display and cassette recorder—there are many wires but this means that the failure of a single unit does not necessarily render the system inoperative.

For example, we output the display to a Hitachi monitor but, as all 380Zs are fitted with an alternative output for an unmodified television set, an alternative display can be used easily if the monitor fails.

The cassette recorder was unmodified except for the fact that RML checks the head alignment before supplying it to a user. This, it claims, largely overcomes any problems of tape incompatibility between different recorders. We certainly had no problems in reading any tapes.

Applications

When RML began designing the system it was involved in many discussions with some London Education Authority microcomputer working parties then looking at the requirements of a system suited for use in schools and colleges.

Some computer educationalists may argue that it is better to purchase 20 Pets than one system on to which you would hang a number of terminals. RML does not subscribe to that view, which I am sure was prevalent around 1974, but which is less common today.

For the most part, the 380Z is being used by schools to teach the principles and programming of computers. It is ideal for teaching Basic, Cesil, machine and assembly languages. It can cope with up to eight simultaneous users.

Other applications include its use in data logging or experimental control, or it can be linked to other minis for data capture.

Hardware

Opening the system cabinet revealed that RML had left plenty of room for expansion. The processor (280A), 32K of memory, cassette, keyboard input and display interfaces are contained on two boards, and there is plenty of room for expansion, as the cabinet can hold eight more boards.

The bus uses flexible flat cables rather than the more usual motherboard. This means that the system is "non-standard" but RML is prevaricating about an S-100 interface to enable S-100 cards to be used with the 380Z.

RML claims that the cable bus is both cheaper and more reliable than an S100 mother board. A further possible approach is to offer boards like parallel and analogue interfaces, high-resolution graphics and voice input. The system is not available in kit form. The boards, and indeed the whole system, were built to a very high standard.

The only controls on the system cabinet are power on/off and re-set. Turning-on power causes the system to enter the cassette operating system (COS) which

(continued on page 29)

The world's best-selling personal computer

APPLE II

∗New powerful basic and new graphics on Rom. £75 card.
∗Talk to Apple with voice recognition, speaker trained, 32 word vocabulary. £165 card.
∗Colour Graphics. High resolution 280h x 192v, 6 colours, easy-to-use. 16 colours, very powerful. Low resolution 40h x 48v,
∗Apple's disks. Powerful DOS. 116K bytes capacity, multiple drives, fast access. £395.
∗Use Apple as a computer terminal 110 or 300 BAUD. Full or half duplex or use with a Dec-writer. £95 card.
∗Use any 8 bit parallel printer with Apple 11. Print up to 3,700 lines per minute. 255 character lines, upper and lower case. £100 card.

Personal Computers Limited

THE BIG APPLE 48K RAM-18K ROM £1850 incl.VAT

194-200 Bishopsgate, London E.C.2. 01-283 3391

● Circle No. 125

CCS Microhire

MICROCOMPUTER RENTAL SPECIALISTS

Before you buy a micro, why not hire it for a day or a weekend?

From £2 a day we hire out a range of micros for evaluation/experience or program development.

* Apple II
* Commodore Pet
* Nascom I or Micros
* Research Machines 380Z
* Rockwell AIM65
* SWTC 6800 or MSI 6800
* Tandy TRS 80

Protect your future investment.
Try out a system now.

For details write to
CCS Microhire, Freepost, Letchworth, Herts SG6 4YA

● Circle No. 126

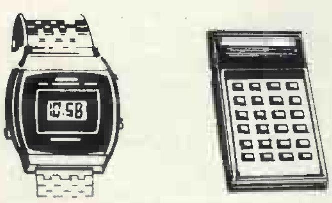
Euro-Calc Ltd

224 TOTTENHAM COURT ROAD, LONDON W1. TEL: 01-636 8161

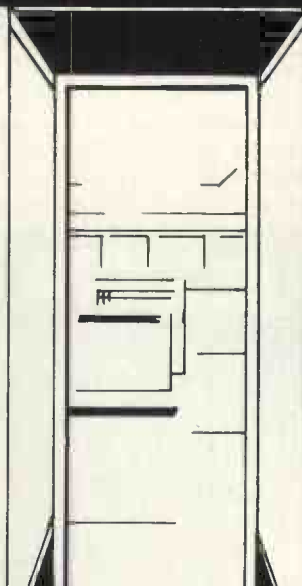
For electronic calculators and watches.

We have the largest range in London at the most competitive prices.

Export and Personal Export Tax, free facilities arranged.

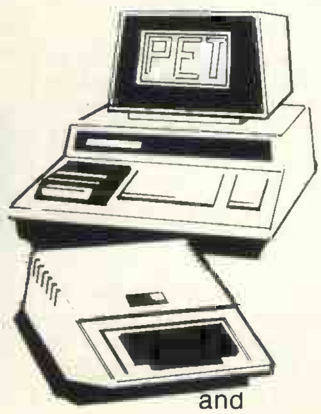


HEWLETT PACKARD ✳ CASIO ✳ SHARP ✳ TEXAS INSTRUMENTS ✳ OLIVETTI ✳ SEIKO ✳ SINCLAIR ✳ COMMODORE ✳ SANYO ✳ NATIONAL PANASONIC, ETC.



●JUST OPEN●
our new
MICROCOMPUTER CENTRE AT: 55-56 HIGH HOLBORN LONDON WC1

We stock the
COMMODORE PET 2001-8



and
APPLE II
microcomputers with a full range of hardware and software options.

● Circle No. 127

(*continued from page 27*)

occupies 3K of ROM on the processor board. The re-set button causes the system to return to COS. The monitor supplied was a standard Hitachi unit and the display was very clear and stable. The keyboard, like the rest of the system, was very well built.

The cassette interface can run at 300 or 1,200 baud. Tapes distributed by RML are recorded normally at 300 baud as this reduces the possibility of a tape not loading correctly. RML says that users normally have no problems in using the interface at 1,200 baud.

Software

The system is controlled by the COS monitor, which is held in ROM on the processor board. There is a text editor, assembler, three versions of Basic and a number of diagnostic programs. The documentation supplied was in a single-ring binder and although we managed to find out everything we wanted to know, it was difficult to find any particular piece of information.

The COS monitor, developed by RML, provides the usual functions of program loading and dumping, starting programs, and debugging aids. There is a front-panel mode which displays the contents of the registers and a selected portion of memory on the screen.

To the first-time user this will be very confusing but in a teaching environment, or for an experienced user, it is a very powerful aid. COS documentation includes a full source listing and a description of the subroutines which are available to the assembler programmer.

The text editor and assembler were developed by RML. We did not have time to use them but from the documentation they appear more than adequate.

The three versions of Basic are:

Tiny Basic, with integer arithmetic;

9K Basic with single precision floating point arithmetic (six significant figures), strong variables, and graphics statements;

12K Basic; as 9K Basic but, without graphics, with extended precision arithmetic (12 significant figures), PRINT USING, and a number of other extensions.

Documentation on the Basics is very sketchy. On the whole it assumes that the reader is already an experienced Basic programmer. It is also confusing in that the description of 12K Basic refers to 8K Basic (what is that?).

Both the 9K and 12K Basics are based on TDL Basic. We had no problems using them and they appear to be reasonably fast. One disadvantage is that there is no syntax checking as statements are entered and the run-time error messages are not very explicit.

System expansion

RML announced recently two floppy disc systems which will be available shortly. They are the MDS, with mini floppies giving 80K bytes per drive, and FDS double-sided, full-size discs giving 500K bytes per drive. The FDS will be upgradable to double-density some time during 1979.

RML will offer the CP/M operating system, so a wide range of software, including Fortran and Cobol compilers, will be available on the system. Additionally RML XDB, the company's extended disc Basic, will be available.

Conclusions

- **The 380Z is one of the best micro systems we have examined and it is British.**
- **It is more expensive than some competitive systems but the extra cost is justified by the high standard of manufacture.**
- **RML is not selling to first-time users and the documentation reflects this fact.**
- **For the enthusiast a 280Z is available comprising the CPU board and VDU board assembled and tested, to which must be added cabinet, power supply, cassette recorder and television set.**

Technical specifications

HARDWARE

CPU: The 380Z uses the Zilog Z80A microprocessor with a 4MHz clock.

RAM (Random Access Memory): The 380Z has sockets and circuitry for two blocks of dynamic RAM. Each block can use either 4K or 16K bit devices. 4K and 16K blocks can be mixed. Thus the basic 380Z can be supplied with up to 32K socket on one of the circuit boards.

Printer interfaces: The SIO-1 interface card is available to provide RS232 or 20mA current loop standard serial input output to serial peripherals.

Input/output support: cassette I/O can be selected to be at 300 or 1,200 bits/second. Output to the VDU is at a speed equivalent to about 5,000 baud with automatic paging, which can be suppressed. Output routines are included for the SIO-1, 2 and 3 interfaces and for the Centronics Line Printers.

Hard-copy: Three printers are available. The Centronics 701 is a 132-column dot matrix printer which uses 'intelligent' bi-directional printing to achieve a throughput of about 60cps. The Centronics 779 is an 80-column dot matrix printer with a throughput of about 30cps. The Trend 800 is a quiet, lightweight (11kg) printer with a 30cps throughput. It uses standard Teletype paper.

Disc system: Floppy disc systems are being developed along with the requisite software.

SOFTWARE

RML 9K Basic: Size approximately 9K bytes; suggested minimum size of memory, 16K bytes; precision; 7 digits; floating point.

Commands, keywords and functions

ABS	FN	LNULL	PLOT	SPC
AND	FOR	LOAD	POKE	SQR
ASC	FRE	LOG	POS	STEP
ATN	GOSUB	LPOS	PRINT	STOP
CHR$	GOTO	LPRINT	RANDOMIZE	STR$
CLEAR	GRAPH	LVAR	READ	TAB
COS	IF	LWIDTH	REM	TAN
DATA	INP	MID$	RENUMBER	THEN
DEF	INPUT	NEW	RESTORE	TO
DELETE	INT	NEXT	RETURN	TRACE
DIM	LEFT$	NOT	RIGHT$	USR
EDIT	LEN	NULL	RND	VAL
ELSE	LET	ON	RUN	WIDTH
END	LIST	OR	SAVE	
EXP	LLIST	OUT	SGN	
FILES	LLVAR	PEEK	SIN	

General: This is a fast, general-purpose Basic interpreter with floating point numbers, string-handling functions, graphics, re-numbering and line editing. Extensions support the 380Z graphics and allow the plotting of symbols, characters and numbers. Two versions are available, the larger including also the ability to read and write data files on cassette, utilising the RML cassette file system.

RML 12K Basic: Size, approximately 12K bytes; suggested minimum size of memory, 20K bytes; precision; 12 digits; floating point.

General: This includes all the features of RML 9K Basic with additions. Lines may be formatted. AUTO line numbering. A REM can follow every statement, MID$ can be used to insert a substring into an existing string. PRINT USING statement for formatted output—options, right-justified, numeric fields, decimal point alignment, floating £ sign, commas every three digits, string field specification. Hexadecimal constants. Substring search. Enhancement of user-defined functions (multi-line, recursive). Program loading under program control. COPY. EXCHANGE. CALL.

The RML 12K Basic will be of use in business and scientific applications requiring features not found in the RML 9K Basic. Due to its 12 digit precision, however, RML 12K Basic is not as fast as the RML 9K Basic. Graphics commands will be added to 12K Basic in late 1978.

RML 2K Tiny Basic: Suggested minimum size of memory, 4K bytes; precision; INTEGERS only. Range +32767 to —32767; commands, RUN, LIST, NEW, OLD, SAVE. Keywords: REMARK, LET, INPUT, PRINT, PLOT, IF, GOTO, GOSUB, RETURN, FOR, NEXT, STEP, GRAPH, TEST.

Functions: ABS, RND, SIZE.

General: This is a small, powerful, integer Basic adapted from Li-Chen Wang's Palo-Alto Tiny Basic for the 8080. The original 8080 code has been streamlined to use Z80 codes and graph plotting with 380Z VDU graphics has been added. This is a Tiny Basic for graphics, games and learning Basic programming.

RML Interactive Text Editor: Suggested minimum size of memory, 8K: This is a character-orientated text editor comparable to text editors available on large mainframes. The editor responds to a command string which can include commands to locate, delete, type and insert characters and lines, and to move an internal pointer. A command string can be saved and executed as a Macro.

In the interactive mode, normal editor operation is continued and uses four scrolling lines at the bottom of the console screen. Whenever a command string is terminated, text either side of the pointer is displayed immediately on the upper 20 lines of the screen. Portions of the command string can be repeated n times using brackets—brackets can be nested up to eight levels. The editor also includes an immediate mode.

General: A powerful text editor.

RML Z80 Assembler: This is an absolute assembler using Zilog mnemonics and producing object code in either the industry-standard Intel format or the RML binary format. The assembler incorporates a number of special features to speed program development on cassette-based systems.

Source text is copied into memory during the first pass; the second pass can be made using this stored copy, by-passing the relatively slow process of re-reading the source from cassette. Errors can be corrected using a co-resident editor which implements a subset of the commands available in the RML text editor. Larger programs can be assembled in stages in a similar manner, if desired, or in the normal line by line fashion.

Utilities: Supplied free with all 380Z/280Z systems. Diagnostics, Memory, cassette.

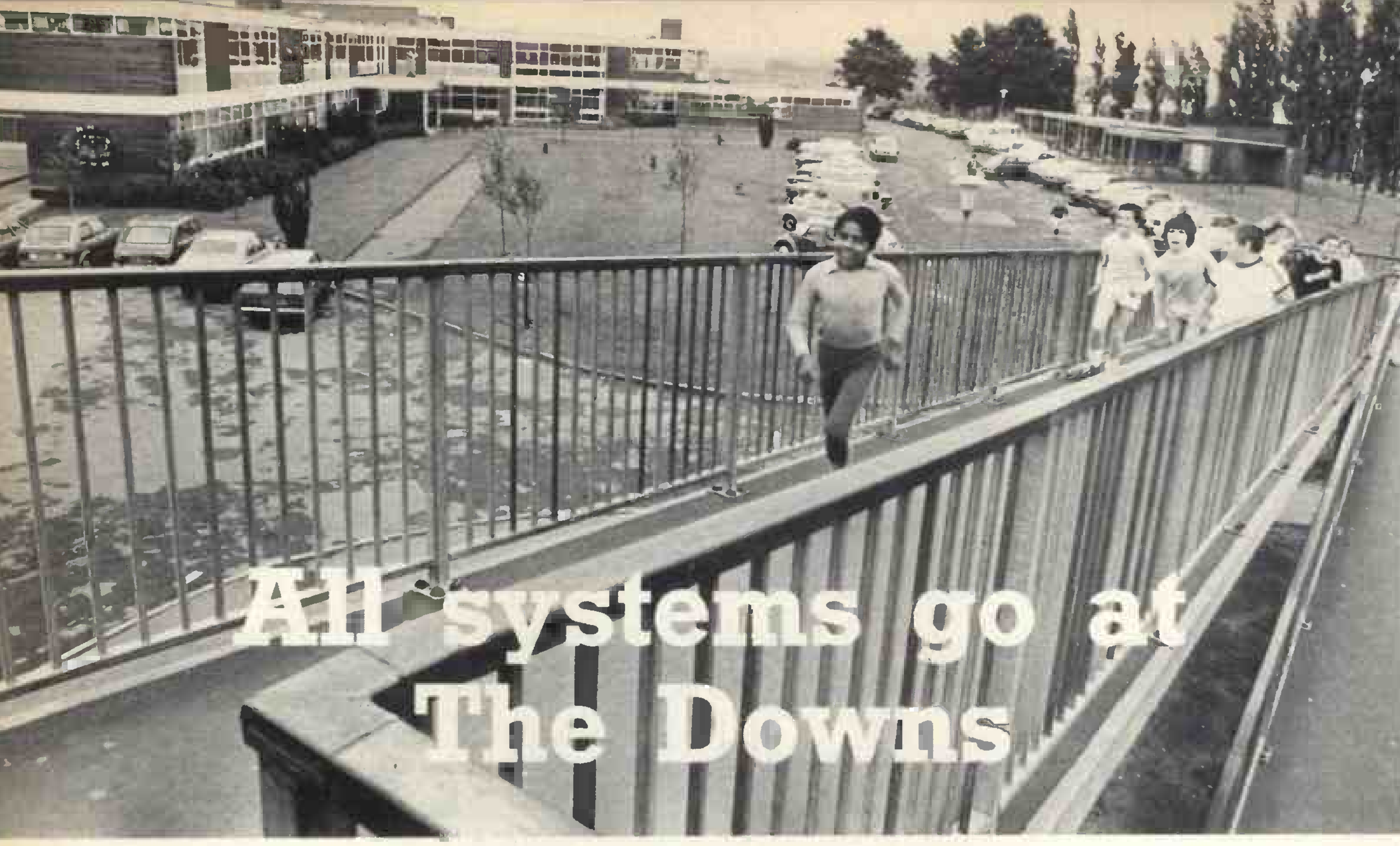
Prices

4K Bytes	£834
8K Bytes	£884
16K Bytes	£965
20K Bytes	£1,015
32K Bytes	£1,158
48K Bytes	£1,421

A 3K TOM Monitor is supplied as standard. The 380Z may be supplied without a keyboard. For cost of the 380Z without a keyboard, subtract £159 from the above prices.

280Z: Consists of the two PCBs used in the 380Z, assembled and tested. To complete a system the user will require a TV set, power supply, cassette recorder, keyboard and a case, if required. Specifications are otherwise as for the 380Z.

Other hardware	
4K Bytes	£398
8K Bytes	£448
16K Bytes	£529
20K Bytes	£579
32K Bytes	£722
FDS-2 dual full floppy disc system	£1,695
MDS-2 dual minifloppy disc system	£895
MDS-1 single minifloppy disc system	£629
Printers	
Centronics 701	£1,389
Centronics 779	£847
Centronics connecting cables	£5
Accessories	
380Z-compatible cassette recorder with record/playback head accurately aligned. Includes connecting cable	£40
Video monitors from	£123
Interfaces	
SIO-1 Serial Interface Card	£74
SIO-2 Serial Interfaced Card	£19
SIO-3 Serial Interface Card	£18
Software	
ZPL 8K Basic Interpreter	£24·50
ZPL 12K Basic Interpreter	£75
RML Z80 Assembler	£25



All systems go at The Downs

IN PERIOD 3 at 11.30 a.m. in room 213, a lower six form takes O-level Physics. They are studying the effects of superimposing one waveform on another, using the school's computer to plot the results of varying parameters. The experiment takes about five minutes to set up, 10 minutes to explain, and 20 minutes for every member of the class to run.

Colin Wells deputises for another teacher in Period 4 at 1.30 p.m. with a lower-stream, second-year class. They are playing five-a-side football on a TV monitor connected to the computer.

"Of course, it's not the sort of thing we do normally during lessons, but as I'm only covering, and you did express an interest . . . ". The class has had little experience of computers. It takes all of 10 minutes for them to grasp the principles of the game and how to use the keyboard to move the players. Boys against girls, they take it in turns to move, acquiring speed and enthusiasm all the time. One-nil to the girls—loud cheers.

More serious

Period 5 at 2.30 p.m. is much more serious. A fourth year CSE class is writing its own programs as part of the computer studies course. As they finish their programs, they go to the computer, type them in, and run them. They know exactly what to do, and Wells is able to let them proceed while he discusses the effects of computing on the school.

Those three lessons give a reasonable cross-section of computing at The Downs School, at Dartford, Kent. On other days the computer, an Altair 680b, might be playing music, controlling a mechanical 'turtle', or producing poetry. The breadth of computer usage at the school is unusual, partly because of the nature of the school, but more because of Wells, who has been developing computing there since 1970.

The Downs School has some 1,900 pupils and is that rare bird of modern education, a secondary modern, though on the verge of becoming comprehensive. Possibly the absence of a strong academic stream has helped avoid the bias of computing towards examination courses and allowed greater scope for more general application. Certainly the availability of microprocessors has allowed Wells to extend the uses of computing to a far greater extent than was possible previously.

Colin Wells

Since 1970 the school has passed through most of the familiar phases of educational computing—batch processing at a local college; remote batch processing on an off-line terminal; an on-line terminal to a computer bureau; and in the summer of 1976, the loan of an Arcturus minicomputer for three months.

That last development which for the first time made a computer directly accessible to the users, generated so much enthusiasm among staff and pupils that it was handed back with great regret. On the other hand, its departure coincided with the arrival of the first low-cost microcomputers in this country, and success with the minicomputer helped Wells to convince the governors that micro would be a worthwhile investment.

Assembly time

The decision to buy the Altair was taken in the autumn of 1976. Wells notes that only three months before he had suggested to an incredulous headmaster that the school might have a computer of its own before 1980.

The computer arrived in Kit form in December, 1976, its 16K memory board a month later, and was augmented progressively by a SWTPC video terminal, a cassette drive and a Trend 30cps printer. Assembly of this heterogeneous collection

(continued on next page)

(*continued from previous page*)

took more than three months and provided a catalogue of mishaps and frustrations which, two years later, it would be unfair to the manufacturers to relate.

Wells' emphasis on the need for sound technical knowledge, supplied by the head of modern languages, Chris Dyos, continues to apply to anyone still contemplating building a computer from a kit.

The Altair has been in heavy demand, averaging at least 30 hours' usage per week. A keenly-supported computing club keeps it occupied after school hours and it has been put to use by the physics, English, modern languages and music departments, as well as to extensive use for computer studies.

Sources for ideas

The list of available programs is seemingly endless, for Wells is a prolific programmer. Many ideas are borrowed from existing programs but generally are modified or enhanced in the process of editing them for the Altair.

For sources of ideas, he scours conference papers, *Computer Education*, the journal of the BCS Computer Education Group, and MUSE newsletters. He also recommends the Scottish Centres for Computer Education and the Advisory Unit for Computer Based Education, among other sources.

Some of the more off-beat ideas are worth mentioning as examples. For the English department, there is a readability program which enables the computer to suggest a suitable reading age for any passage typed-in, based on established standards of word and sentence length.

Insight to control

The mechanical 'turtle' derives from the idea that, if a computer can control the two motors in a cassette drive, it can control any other two similar motors. The turtle consists of two motor-driven wheels in a simple chassis, controlled by keyboard commands or program. As well as giving insight into machine control by computer, it can be used, given sufficient accuracy, in a variety of mathematical studies.

Wiring a loudspeaker to some of the address lines has provided a musical facility. Besides providing considerable scope for CSE projects, the precise control over pitch and speed is useful to the music department for studying sound and musical components.

All this activity stems from Wells' belief that, while computer science examination courses may be used to justify the purchase of a computer, its main function should be as a teaching aid in all subjects. All pupils take a computer studies course in the third year, so that they are aware of what the computer can do for them in other subjects by the time they start their examination courses.

Science teacher Martin Lawrence with a lower sixth physics class.

He also feels that, particularly at secondary modern level, the entertaining aspects of computing are important for pupils initially to be involved. As a result, many of the programs he has written have a strong games content.

A simple arithmetic test is turned into a fascinating challenge by adding a timing routine. For the first of a series of nine questions, the pupil is given nine seconds to answer but, if he is correct, only eight seconds for the next and so on, until the last question has to be answered in one second. The effect, according to Wells, is riveting.

Impressive results

Some results produced by pupils are also impressive, including the simulation of a banking operation, the analysis of election results with output in the form of histograms, and, of course, the famous five-a-side football game—a joint effort by the computing club.

At the end of the year, Wells is leaving to join a research group in Plymouth, working on the uses of micros in the classroom. Not surprisingly, he hopes to find time to continue writing programs, particularly for use in lower forms.

THE NEW HORIZON

The microcomputer for those who need more than the minimum. The right processor for business, scientific and educational use. Proven applications include Games • Educational • Word Processing • Invoicing • Stock Control • Sales Ledger • Purchase Ledger • Mailing • Scientific.

Languages
Powerful Basic including sequential and random access disc files • formatted output • strings • line editor • machine language CALL • many other facilities. Optional additional software (under CP/M operating system) includes BASIC compiler, FORTRAN and COBOL.

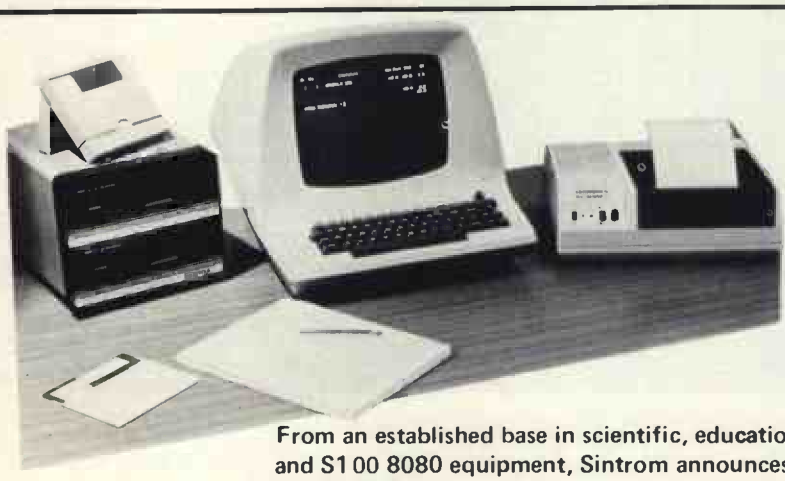
The Horizon computer includes:-

Specification
Zilog Z80A MPU • S-100 bus (12 slots) • Solid well-built case • Up to three Shugart mini-floppy disc drives, 90KB each • Serial port for CRT or Teletype • Real-time clock on motherboard • Optional additional serial port and parallel port • Powerful operating system and monitor • Access to wide range of S-100 special application boards.

Horizon Z80A computer with 24K RAM and 1 disc drive £1483.
2 Disc drives £1823 (exclusive of VAT and carriage).

Equinox Computer Systems Ltd,
32-35 Featherstone Street,
London EC1Y 8QX.
Tel: 01-253 3781/9837.

● Circle No. 128



SINTROM means business

From an established base in scientific, educational and personal computers using SWTPC 6800 and S1 00 8080 equipment, Sintrom announces a range of small systems for the business user. Easy availability of Micropolis-compatible applications software plus CP/M, COBOL, APL, FORTRAN and BASIC allows a speedy implementation of the total system.

● Ledger ● Payroll ● Inventory ● Word Processing ● High Resolution Graphics ● Industrial Control

Single Disk 143K £453
Dual Disk 630K £1159
Include PSU, S100 controller, Basic/MDOS.
Add-on 143K £399
Add-on 630K £859

Host S100 computer with 32K and 1/0 card
From £1200

Plus integral dual Disk
From £2200

ADM 3A VDU £620
Centronics μPrinter £350-£400
Centronics 701 £1400

Office Hours:
Monday - Saturday
Access/Barclaycard
Prices exclude VAT/CARR

Write for free catalogue

SINTROM GROUP

Sintrom Microshop
Arkwright Road,
Reading, Berks. RG2 OLS.
Tel: Reading (0734) 84322

● Circle No. 129

Something in the City

BOWLER-HATTED, pin-striped and twin-setted, the grey hordes of commuters pour *en masse* each weekday morning from London's Cannon Street station, bound for their working days in the City.

Since April, there has been a friendly face to welcome them as they go through the ticket barriers. A Commodore Pet gleams through the window of a shop in the complex outside the station and spells-out continuously its inviting message.

The shop is run by Sumlock-Bondain, a company with its roots in the Sumlock-Anita calculator business. It was formed by people made redundant by the take-over of Sumlock-Anita by the U.S. giant Rockwell, who did not want to see their talents going to waste and decided to continue in the same line of business.

Originally they sold the Sumlock calculator range but when that was wound up by Rockwell, they found favour with Texas Instruments, for whom they are wholesalers, and with Hewlett-Packard, for whom they act as a main agent.

As the company was getting on its feet, disaster struck. A fire at its London headquarters destroyed a great deal of property and many valuable files.

Managing director John White recalls: "We were the first of the Green Goddess fires—during the firemen's strike—and we lost the records of all our clients in the City and the West End."

Promoting systems

Records of that kind are the life-blood of any company, so special measures had to be taken. "We needed somewhere fast in the City. The situation inside Cannon Street station seemed ideal, with people passing through on their way to and from work".

The area contains the greatest concentration of mainframe computer systems in the U.K. and so the shop tends to attract many computer people, as well as hard-headed businessmen wanting to know what a personal computer can do for them.

"It has worked very well", says White, who used his first Pet system to do some initial advertising for the shop. Although it was not open until May, from April the shop had a Pet system in the window running a continuous program informing curious passers-by of what was to come.

Being in the centre of the financial district of London, the policy of the shop is to promote systems for business use. "We've no real intention of going towards the hobby market", says White.

With his experience in the calculator field, and still with a strong investment in that side of the business, the same philosophy is applied to microcomputers. He wants to supply systems which the unskilled user can operate without resorting first to the soldering iron.

The only computer system on show at the shop at the moment is the Pet; the remainder of the space is occupied by calculators. The company also supplies the Adler TA20 system, and the Compucorp 625 series.

As well as the casual shopper, big companies are catered for. Among recent orders have been Pet systems delivered to ICI, and a £25,000 Compucorp system for London Transport.

That is all somewhat up-market and Sumlock-Bondain realises the potential of some of the cheaper microcomputer systems as well as anyone. It has been looking at other systems to fill its range and has signed an agreement to stock the Horizon machine from North Star Computers, which starts at just less than £1,000.

The shop now also has the right to market the Equinox system—price range £1,800–£2,500. According to shop manager Mike Kick: "After the Christmas rush, in which we expect to sell plenty of calculators, more space will be given to computer systems".

The emphasis will be still on ready-to-run systems. As well as carrying most of the Commodore and Petsoft program cassettes for the Pet, Sumlock-Bondain has commissioned a software house, Micro Software Systems of Grays Thurrock, Essex, to write software for specific applications.

"We shall be running full demonstrations in the shop with different packages", says Kick. "One week it will be a solicitor's systems. The next week it will be packages for accountants".

The plan is to have a qualified adviser on hand at all times to deal with customer queries.

What about back-up? "People with any problem are always at liberty to come to us with their problems", says Kick. An added advantage is that Sumlock-Bondain has a sister organisation—Anita Electronic Services (London)—which specialises in servicing and maintenance. The links with software houses should also provide the necessary help with any programming difficulties.

Looking for growth

So what kind of people does Kick find attracted to his shop? "Anyone and everyone, really".

Casually he mentions, too, that Lord Rothschild goes in to browse round from time to time. "He has bought two calculators but has so far resisted the attractions of the Pet. He's shown an interest in it", says Kick, "but he's not bought one yet".

As to the future, White and Kick see only growth and expansion. "We shall be opening a new shop in Norwich very shortly", says White, "and there will be more in London, too".

Big Apples from little apples grow

IT HAD to happen sooner or later—a multinational company taking an interest in microcomputers. ITT needs little introduction, if only because of its size. It is to manufacture the Apple II personal computer in Europe.

The prime reason for the interest of *Practical Computing* in the ITT consumer products division is that in May, the company informed the world at large of the wonders of Prestel and of its plans for the viewdata market.

The ITT microcomputer was one of a range of products for viewdata demonstrated, but the eagle-eyed were quick to note that the microcomputer was, in fact, an Apple II with an ITT label on it.

Many things have happened since, not least the launch of *Practical Computing* when it reviewed the Apple II and referred to ITT intentions.

Curiosity aroused, however, we decided to visit ITT at Basildon to interview the man in charge of the company's viewdata strategy for the U.K., Graham Pypus, who proved to be most helpful.

Colour matching

It seems ITT will begin manufacturing the Apple in the U.K. early next year and will sell it as the ITT 2020 personal computer. The company has entered into an agreement with Apple in California to buy the basic Apple printed circuit board and to market the micro in its finished form in the U.K. and Europe.

"We'll be making our own casings in the first quarter of next year", says Pypus. "They will have certain cosmetic alterations—essentially colour matching to our product range in silver with a black keyboard. We're hoping to launch this as a product shortly".

ITT is not aiming at the hobbyist market essentially but is looking to assemble complete systems configurations for business users—comprising the microcomputer, a minifloppy disc drive and a printer.

"We have had firm indications from Andre Sousan of Eurapple—Apple's exporting arm—that whereas in the past hobbyists have been the main market for Apple, this trend is now decreasing.

"Spending patterns differ between the U.S. and the U.K., anyway. We feel that the U.K. market for the ITT 2020 will come from the small businessman. There is bound to be an over-spill to hobbyists but the message we want to put across is that it is a small business system".

The agreement with Apple is non-exclusive. ITT would not need one anyway; it would take a sizeable organisation to gear production to a competitive capacity. Also the ITT image lately does not accord with anything which might suggest any hint of a monopoly, particularly in Europe.

Made in Basildon

"Obviously a lot depends on the spirit of the agreement", Pypus continues. "We would not be too happy if Apple supplied its boards to others in competition with us. It is in Apple's interest, too, that we should spearhead the introduction of the product to the European market".

European operations will be directed from ITT plants in the U.K., France and Germany. The U.K. headquarters for manufacturing will be Basildon, where the keyboard and casing will be made. The power supply will be manufactured in Germany and the UHF modulator in France.

"Our intention in the U.K. is to build a network of distributors over the next few weeks. We are looking essentially for software houses, because ITT will not be involved directly in the provision of software.

"We will award distributorships to some 15 companies which will undertake full responsibility for servicing systems and software back-up. One of the prime requirements is that they have a direct sales force.

Keen interest

"The distributors, in turn will allocate dealership franchises within their areas. We're not looking to proliferate on the retail market; we do not necessarily believe that the High Street shop is necessarily geared to provide the product.

"So far we have received direct applications for virtually every area for distributorships and dealerships, despite having made no formal announcement of our intentions.

"Initially we will have a sales liaison team of two people who will deal with end-user problems and enquiries and refer to the appropriate distribution".

Prestel, though, has been pushed back somewhat. Pypus says: "We had hoped for the first public trial in June but there have been slippages on the date, although we understand that there will be some 40 or 50 connections in the near future. Our viewdata products will be launched in March or April".

ITT forecasts that some 25,000 viewdata sets will be installed in 1979, increasing to around one million in 1983. That would mean nearly two-and-a-half million sets in the home, or one in every eight households. Total sales value for the microcomputer is expected to reach £160 million by 1982.

A printer to which the contents of a television screen may be dumped is offered at the low-end configuration and called a rotary printer, manufactured by SCI in the U.S.

Likely prices are: ITT 2020 4K bytes RAM, £827; 16K bytes RAM, £950; 32K bytes RAM £1,114; 48K bytes RAM, £1,278. (All prices plus 8 percent VAT).

An Apple in disguise.

Off-the-shelf packages for Pet

THIS MONTH we start a regular software column which will look at off-the-shelf packages. We begin with three business applications which have been developed to run on the Commodore Pet.

The Pet, reviewed in our October issue, costs £695 and is complete with built-in screen, tape recorder and keyboard and is suitable for limited business applications.

Until recently, there has been little off-the-shelf-software business, available for the Pet. The programs reviewed this month are two for stock control and one for sales analysis.

Both the stock control packages cost between £15 and £20. They consist of one program each and are complete with documentation, at present half-a-dozen photocopies of type-written A4 sheets. They are distributed by Commodore at 360 Euston Road, London, NW1.

The programs were written by Pet dealers, Rockstock by Rockliff Brothers of Liverpool and Ardenstock by Arden Data Processing of Leicester.

Rockstock was written for a Pet with two cassette decks and is capable of handling an unlimited number of products. We found the documentation of average quality and even a complete newcomer to computing would find it fairly easy to follow.

Three modes

There are three main modes of operation. The creation of a new master file, which will contain the data on all the product lines is the first thing the user will do having purchased the program after, of course, taking a copy of his program tape in case of accidents.

The second mode is used to examine the stock file. In this mode the data on each product is displayed sequentially; it is obtained as required from the data tape on drive 2.

The data is displayed in scrolling or page mode; data on each item is displayed for about 10 seconds and then data for the next item is displayed, the user having the option to freeze the display on the screen at any time.

The third is the update mode which allows the user to change the data on any particular product or to add new products to the file.

The user can thus sell stock, order stock, or add stock, or alter the buying and selling prices, the re-order level, or the product description.

To try to avoid unauthorised access and changing of data, the program employs password encoding of the data files and uses three levels of full or restricted access.

Ardenstock uses one cassette deck on the basic Pet and the user is limited to a

maximum of 90 products, a total which could be increased to 255 by adding more memory. Whereas the data in Rockstock is accessed sequentially from the data tape, Ardenstock stores all the data as data arrays within the Pet memory, thus allowing the user fast random access to any data.

The examination mode in Ardenstock is very simple and fast. Because only one cassette deck is used, the creation and in particular the update mode are rather lengthy and involved processes.

This is a factor imposed by the limitations of the Pet as a computer system and should be relieved when the manufacturers produce a floppy disc for the machine. Like Rockstock, there is good and ample documentation with this program.

Well written

Both programs are well written and the authors have tried with reasonable success to overcome the limitations of the machine and produce a viable business program. It is arguable, however, whether either of the programs is actually a stock control program; they seem more like stock recording programs.

Some attempt has been made in Ardenstock to show and record stock movements; no attempt has been made, however, to utilise this data to produce, for example, automatic re-ordering of stock. The time taken to access data in both programs is rather long; this is more the fault of the system rather than the program. Users of Rockstock can, however, reduce the information access time by limiting the number of products on a tape to between 25 and 50.

One or other of the two programs should prove a useful purchase to anyone wanting to use a Pet in a stock control type of application.

The sales analysis package is available from Petsoft, a company specialising in writing and selling software for the Pet.

It is for companies wishing to maintain up-to-date sales figures. Typically it can be used by firms such as car dealers, wholesalers of electrical components and the food industry, who are at present the biggest customers for the package.

It displays and compares in easy, readable visual graphic form, the achievements of, for example, salesmen within their respective divisions and their total sales over a period of time.

The program holds a capacity of four divisions, six salesmen and 12 months. It converts this information into a graphic display (bar charts) so avoiding the confusion of interpreting many over-lapping numeric figures.

By loading information once a month, the program analyses that information and keeps a company up-to-date on its sales figures. The program shows the sales trend or how seasonal fluctuations in the market have affected sales over the previous months. After this data is analysed, it is then compared and displayed on bar charts.

Time-saving

Small companies do not always have the time to look at sales figures until audit time; then it is often too late. This can be avoided by the sales analysis program which is time-saving, easy to follow and simply efficient. It can correct things when they are wrong, before it is too late, and does not require a qualified person to operate it.

A similar program on a larger computer would obviously be far more expensive, and at £10, the program is certainly worthwhile for what it will do.

If it has a limitation it is because it is only for companies whose turnover does not exceed £512,000 per month.

● *Petsoft has also announced the release of a new catalogue featuring more than 100 business, educational and applications programs for use with the Pet. As well as programs, there are new software tools available, among them an Automatic Program Linker (£10) and some clever new games, including the long-awaited American Micro-chess program (£14). Catalogues can be obtained from Petsoft, PO Box 9, Newbury, Berkshire. 0635 201131.*

LABYRINTH has been written in Basic by Sean McHugh to run on a Pet but can obviously be converted to run on most micros. Some program notes at the end of the program take into account a few peculiarities of the printer we used.

You are placed in the centre of an invisible maze in three dimensions and you have to get out.

There are monsters in the maze. Poor, starving victims of a rather primitive ecology, or blood-crazed carnivores, depending on your point of view.

The monsters can hear movement and will devour you if they can reach you before becoming confused by the repeated blows to the head which they suffer by wandering around in the dark.

You can defend yourself by cueing all of the known anit-monster spells in button presses, the alphabet serving in this case for cues.

You can grope your way to the various doors of the maze and find behind them gold, stairs, or yet more monsters.

Fast and clever

You have to be fast, cautious and clever to survive the Labyrinth.

The –* at the end of each program line signifies end of line and doe not have to be typed in the program.

With Pet a ? means a PRINT statement.

The particular printer used could not print Pet's specific command characters nor the graphics characters. There are five command characters to watch for, as well as the home symbol and the clear screen symbol.

Figure 3 stands for CLEAR SCREEN on the following lines: 10, 1010, 2130, 2140, 4150, 8000, 8100.

The printer has reproduced the graphics characters in lines 8100, 8200 and 8300 as lower-case letters. The relevant symbols are slanting lines at the four corners and straight lines at the four sides. The S in the middle is a "heart" symbol to indicate the man. In line 8070 the "Q" stands for a small dot—about the size of a match-stick head" which is the monster. Have fun. We did.

Take your pick as y chase the monster, glory or sink the fle

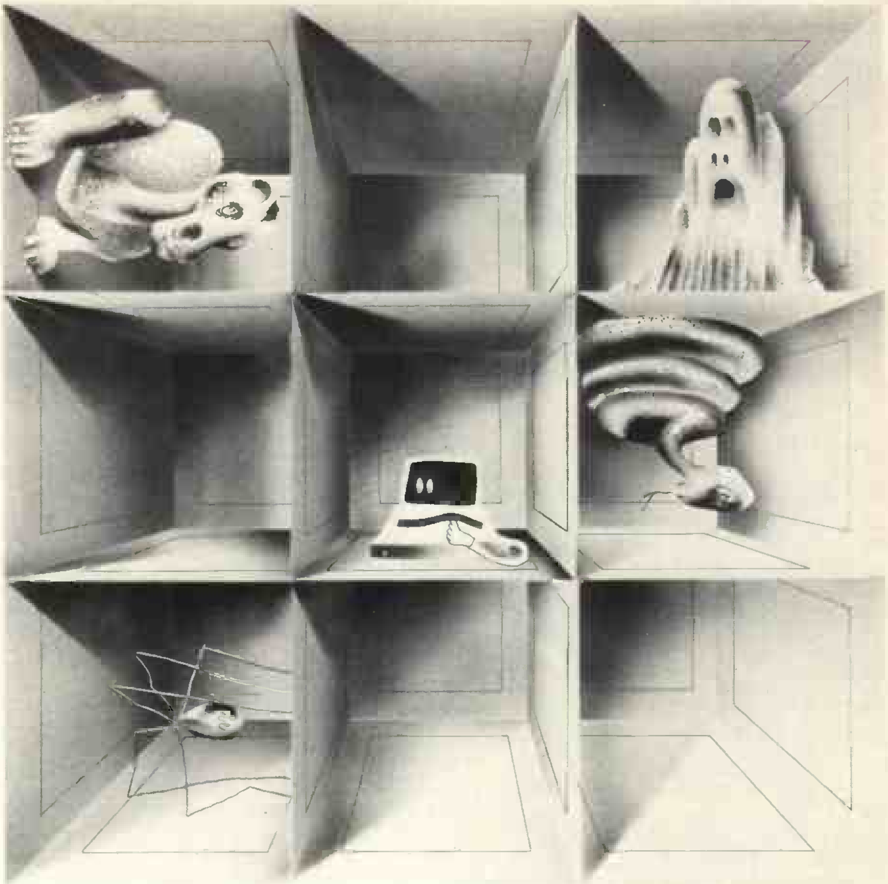
This month we show you how to program three exciting games into your system. All are written in Basic. *Labyrinth* is a monster-versus-man game. *Cars* puts you at the controls of a Formula One racing car and *Battleships* is the popular find-the-enemy game.

```
10 PRINT"3"–*
20 A1 = 20:A2 = 36: S9 = 33008–*
40 DIMV (9.2)–*
50 FORK = 1 TO 9 :READY (K.1), V(K.2): NEXT–*
70 DATA 1,-1,1,0,1,1–*
80 DATA 0,-1,0,0,0,1–*
90 DATA -1,-1,-1,0,-1,1–*
120 DIMC (8.2)–*
130 FORK = 1 TO 8;READ C (K.1), C(K.2) NEXT–*
140 DATA-1.0-1.1.0.1.1.1.1.0-1.-1.0-1,-1,-1.–*
200 DIMS SV (8)–*
210 FORK = 0 TO 8 (READS V$(K): NEXT–*
220 DATA "HEAVING MOUNTAIN BLOAT",
    "PURPLE BELCH THING", RANDY
    NEWPUNK"–*
230 DATA "OVERWEIGHT WRAITH",
    "GALLOPING TWITCHER"–*
240 DATA "GROUP O VAMPIRE", DANDRUFFED
    WEREWOLF", "RABID GROPE", "BLOODY
    'ORRIBLE"–*
400 PRINT "DO YOU KNOW HOW TO PLAY"–*
410 PRINT "1 PLEASE PRESS Y FOR YES N FOR NO"–*
420 GET A$ : IFA$ = "N" THEN 8000–*
430 IFA$ <> "Y" THEN 420–*
1010 PRINT "3 YOU ARE ON LEVEL "LV" WITH
    "GL" GOLD PIECES"–*
1012 PRINT "YOU HAVE SPELL-BLASTED "CD" OF
    THE LOCAL NASTIES"–*
1013 PRINT : PRINT "PLEASE PRESS S TO START"–*
1014 GETS $ : IFS$"="THEN 1014–*
1015 PRINT "3": H1% = 10: H2% = INT (10* RND
    (T1)+10); FR = FR = 1–*
1020 IF PEEK (S9 + H1% *40 + H2%<>32 THEN
    1010–*
1030 POKES 9 + H1%*40 + H2%, 83–*
1040 M1% = INT (10*RND(T1) + 10): M2% = INT (10*
    RND (T1) + 10)–*
1045 IF PEEK (S9 + M1% 40 + M2%) = 102TH
    EN1100–*
1050 IF PEEK (S9 + M1%*40+M2%) <>32 THEN 1040–*
1100 REM MOVE SEQUENCE–*
1130 GOSUB 2800 : PRINT "32 PRESS A NUMBER
    TO MOVE" –*
1135 GET D$ : N = VAL (D$) IF N<1 THEN 1135–*
1140 IF N = 5 THEN GO SUB 2800: GO SUB 2900:
    GO TO 9000–*
1150 F1 = H1%:F2 = H2%:T1 = F1+V (N.1):T2=
    F2+V(N.2)–*
1155 IF T1>1 ORT 1 > A1 ORT 2 <1 ORT 2> A2 THEN
    9100–*
1160 H = 59 + F1 *40 + F2 :G = S9+T1* 40+T2–*
1162 IF PEEK (H) = 160 THEN PRINT "3 YOU GOT
    CRUSHED BY A FALLING WALL": END–*
1165 IF PEEK (G) = 160 THEN CH = 25 :GO TO
    1270–*
1168 IF PEEK (G) = 102 THEN 1300–*
1170 GO SUB 2900–*
1180 IF CH <15 THEN POKEG,132: GO TO 2000–*
1270 IF CH THEN POKE G,160 :POKEH,214 : GO
    SUB 2700:POKE H, 83: GO TO 1130–*
1300 POKE H, 102: POKE G, 83–*
1305 H1% = T1: H2% = T2"–*
1310 GO SUB 2800–*
1350 GO SUB 2950 : NM = CH–*
1390 F3 = M1% ;F4 = M2%–*
1400 FORMO = 1 TONM–*
1410 H = S9 + F3 *40 + F4 : IF PEEK (H) = 83
    THEN 1430–*
1420 POKE H, 102–*
1430 D1 = T1 F3:D2 = T2=F4–*
1450 R = (D1*D1 + D2*D2) 5:IFR<1.2 THEN 4000–*
1510 J1 = INT (D1/R):J2 = INT (D2/R)–*
1530 G = S9+(F3+ J1) *40 FT4+ J2–*
1544 IF PEEK (G) = 83 THEN 4000–*
1545 IF PEEK (G) = 160 THEN 1610–*
1548 IF PEEK (G) = 102 THEN 1570–*
1550 GO SUB 2900–*
1560 IF CH 50 THEN POKE G,160:POKE H, 214:
    GO SUB 2700:POKE H, 81:GO SUB 3000,
    GO TO 1610–
1570 POKE H,102:POKE G,81:F3 = F3*+J1 = F4
    F4+J2 = *
1610 NEXT–*
1630 I F M 1% = F3 AND M2% = F4 THEN POKE
    S9+F3 *40+F4; 102: GO TO 1040–*
1650 M1% = F3 :M2% = F4 –*
1700 GO TO 1100–*
2000 REM D S G M SEQ–*
2010 PRINT "3 WOULD'NT YOU JUST LOVE TO
    OPEN THAT DOOR?–*
2020 PRINT: PRINT "PRESS Y FOR YES OR N FOR
    NO"–*
2030 GET A$ :IF A$ "THEN 2030–*
2040 IF A $ = "N"THEN T1 = F1 :T2 = F2: GO TO 2150–*
2050 POKE H, 102:POKE G, 83 : GO SUB 2900–*
2055 GO SUB 2800–*
2060 IF CH <30 THEN PRINT "WHOOPS! " GO SUB
    2700: GO TO 4000–*
2070 IF CH<60 THEN GL = G1 + CH: FOR I = 1 TO
    7, PRINT "GOLD!!, NEXT "PRINT CH;
    PRINT" PIECES!"–*
2080 IF CH<60 THEN GO SUB 2750: GO TO 1305–*
2090 POKEH, 83: POKE G, 147,PRINT "DO COME
    DOWN THE STAIRS"–*
2110 PRINT "PRESS U FOR UP, D FOR DOWN, N FOR
    NO"–*
2120 "GET A$ :IF A$ = "" THEN 2120–*
2130 IF A$ = "U" THEN IV = IV+ 1:PRINT "3"
    GO TO 1010–*
2140 IF A$ = L "D" THEN LV = LV—1: PRINT "3" GO
    TO 1010–*
2150 GO SUB 2800: PRINT "VERY WELL – BE
    UNADVENTUROUS":PRINT"STAY ON
    LEVEL"–*
2155 PRINT LV: PRINT "WITH",: PRINT GL; PRINT
    "GOLD PIECES"; GO SUB 2750–*
2160 T1 = F1:T2 = F2: GO TO 1305–*
2700 FOR DL = 1 TO 100:NEXT: RETURN–*
2750 FOR DL = 1 TO 2000 NEXT: RETURN–*
2800 REM BLANKS. SCREEN TOP–*
2802 PRINT "3 = " –*
2805 FORT = 1 TO 5–*
2810 PRINT '
2815 NEXT–*
2817 PRINT "3 = "–*
2820 RETURN–*
2900 CH = INT ((100-1)* RND (1) + 1):RETURN–*
2950 CH = INT ((9-1)*RND (1) + 1): RETURN–*
3000 FOR I7 = 1 TO 8–*
3010 T3 = F3+ C (17,1):T4 = F4 + C (17,2)CL = S9+
    T3*40+T4–*
3012 IF PEEK (C1) = 83 THEN 4000–*
3015 IF PEEK (C1) = 102 THEN POKE C1,81 :POKES
    9+F3+40 + F4:102:F3 = T3:F4 = T4; GO TO
    3110–*
3020 GO SUB 2900: IF CH<25 THEN POKE CL,160–*
3030 IF T3<1 or T3>A 1 or T4<1 or T4>A2 THEN
    3090–*
3050 IF PEEK (CL) = 160 THEN 3090–*
3070 POKES 9 + F3 *40+ F4,102:POKE CL, 81.:
    F3 = T3, F4 = T4: RETURN–*
3090 NEXT–*
3110 RETURN–*
4000 REM.WEAPON–*
4005 GO SUB 2950:PRINT "IT'S THE " SY$ (CH-1);
    GO SUB 2750–*
4010 GO SUB 2800: PRINT "YOU GOT THREE
    SECONDS TO HIT THE RIGHT SPELL LETTER–"*
4025 GO SUB 2750: GO SUB 2800–*
4030 N1 = T1 :SP = INT (C90–65) *RND (1) + 65)–*
4035 PRINT "FROM NOW!"–*
4040 GET A$–*
4050 IF A$ = CHR $ (SP) THEN H1% = T1 :H2% =
    T2:CD = CD+1 GO TO 4085–*
4060 M1 = T1 :IF M1>N1 + 1080 THEN 4100–*
4070 PRINT " 3 N OPE, YOU GOT":PRINTINT
    ((1080 –(M1–N1))/360; PRINT"SECONDS
    NOW!–*
4080 GO TO 4040–*
4085 GO SUB 2800; IF INT ((RND (1) + FR.)) THEN
    4100–*
4090 FOR I = 0 TO 10:PRINT "0.K!, NEXT:PRINT
    "YOU GOT IT"?–*
4094 POKE H.102:GO TO 1040–*
4100 FOR I = 1 TO 20:PRINT "CURSES, CURSES,
    GERROFF YOU BRUTE"–*
4110 PRINT " 2 YAMM CHOMPA CHOMPA ASHRRUP
    CHOMP"–*
4130 PRINT "DOWN ,BLAST YOU"–*
4140 NEXT: GO SUB 2750–*
4150 FOR I: 1 TO 100: NEXT : PRINT "3
    AAAAAAAGH"–*
4160 PRINT:PRINT: PRINT :PRINT "2 CHOMP!–*
4170 END–*
8000 PRINT "3 HAVING OFFENDED 311 THE SEER
    SPEKTAKLES"–*
8005 PRINT "1 BY ASKING IF HE HAD CRYSTAL
    BALLS"–*
8010 PRINT "1 YOU HAVE BEEN CAST"–*
8015 PRINT "1 INTO THE CORRIDORS OF
    REWOP"–*
8020 PRINT "1 YOUR PET DEMON-ME-WILL HELP
    YOU":PRINT "1 BY THIS SCREEN"–*
8030 PRINT " 1 YOU WILL SEE':PRINT"12 2 WALLS
    LIKE THIS"–*
8040 PRINT "& CORRIDORS"–*
8050 PRINT "2D2 DOORS"–*
8060 PRINT "2A2 STAIRS"–*
8070 PRINT " Q MONSTERS"–*
8080 PRINT "S AND YOURSELF: PRINT "1 PLEASE
    PRESS A BUTTON"–*
8090 GETA$ :IF A$ = " THEN 8090–*
8100 PRINT "31111111" PRINT TAB.(9) "M B N"
    PRINT TAB (9) 789"–*
8200 PRINT TAB (9) " 4 S 6 "–*
8300 PRINT TAB (9) "123":PRINT TAB (9) N B M"–*
8400 PRINT "11 PRESS THE NUMBERS SHOWN TO
    MOVE IN THE 1 DIRECTIONS INDICATED"–*
8410 PRINT "1 IF YOU ARE UNLUCKY ENOUGH
    TO MEET A 1 MONSTER"–*
8420 PRINT " 1 YOU HAVE ONLY ONE CHANCE"–*
8430 PRINT " 1 TO PRESS THE CORRECT LETTER
    OF THE ALPHABET"–*
8500 PRINT " 1 PLEASE PRESS A BUTTON"–*
8600 GET A$ : IF A$ = "" THEN 8600–*
8700 GO TO 1010–*
9000 REM HELP–*
9010 PRINT "1 ASSUME YOU NEED HELP I CAN
    CAN TRY":PRINT"1 IF YOU PRESS Y FOR
    YES"–*
9020 PRINT "IN FOR NO WILL LET YOU MOVE"–*
9030 GET X$:IF X$ = "Y" THEN 9060–*
9040 IF X$ = "N" THEN 1100–*
```

ou drive to et

```
9050 IF X$=" " OR X$= "5" THEN 9030-*
9060 IF CH>= 50 THEN LV=LV—1:GO TO 1010-*
9070 GO TO 4100-*
9100 REM TEST EXIT-*
9110 IF LV THEN POKE S9 + T1*40 + T2,160: GO
     TO 1130-*
9120 PRINT "3 YOU GOT OUT !! 311 YOU TOOK
     "GL" GOLD PIECES, 31111 AND KILLED
     " CD" MONSTERS"-*
9140 PRINT:PRINT "WELL DONE INDEED"-*
9150 FOR D1= 1 TO 5000:NEXT-*
9160 FOR I= 1 TO 10 PRINT:"WATCH OUT !!":
     NEXT:M=M + 1-*
9170 IF M<2 THEN GO TO 4000-*
9180 PRINT "NO IT'S O.K. YOU ARE OUT ! -*
     READY -*-* READY-*
```

Note: in line 10 "3" means clear screen.
-* signifies carriage return and does not need to be programmed in.
In any line where number 32 occurs this signifies a reverse field.



On the grid

In this program you are at the controls of a Formula One racing car attempting to complete a lap round the Brands Hatch circuit. At each of 22 points round the circuit you are given a description of the next section of the track, and have to type three numbers, separated by commas, to indicate the gear, the brake setting and the accelerator settings respectively. The clutch is automatic and former world champion racing driver Jackie Stewart will steer for you.

The data input for gear, brake and accelerator settings are checked to ensure that the values are integers, that the gear is in the range 1–5, and the brake and accelerator settings are in the range 0–9. The new speed is calculated based on the speed for the previous section, plus a factor depending on the gear and accelerator settings and minus a factor depending on the brake setting and frictional losses. The engine revs are worked out from the speed and the gear. A number of checks are performed:

(i) Have you stopped?
(ii) Have you stalled?
(iii) Have you slightly over-revved the engine? You will be warned, but should you do this three times you will blow-up the engine.
(iv) Have you seriously over-revved and blown the engine?
(v) Have you exceeded the maximum safe speed for the section by 10 mph? If so, you have crashed.
(vi) Have you exceeded the maximum safe speed by up to 10 mph? If so, you skid and lose 25 percent of your speed.

Barring fatal mishaps, you are told the speed you have attained and the engine revs, and you are given the information for the next section.

When you have either crashed or completed a lap, you are invited to attempt another lap. You may choose to drive yourself, or be shown how it should be done by an expert.

```
10 DIM M$(10), QS(18), J$(9), X(22), Y(22), Z(22),
   G(5), J(22), L(10)
20 PRINT TAB (8); "CARS"
30 PRINT TAB (8); "=====))
40 LET J$ = "NORMAL"
50 PRINT "WOULD YOU LIKE INSTRUCTIONS?
   TYPE YES OR NO & PRESS RETURN."
60 INPUT Q$
70 IF Q$ = "YES" THEN 110
80 IF Q$ = "NO" THEN 230
90 PRINT "REPLY ' "; Q$; " '  NOT
   UNDERSTOOD. PLEASE ANSWER YES OR NO!"
100 GOTO 60.
110 PRINT "YOU ARE ABOUT TO DRIVE THE
    LATEST FORMULA 1 LOTUS"
120 PRINT "AROUND BRANDS HATCH. JACKIE
    STEWART WILL STEER FOR"
130 PRINT "YOU. THE CLUTCH IS AUTOMATIC.
    ALL YOU ARE REQUIRED"
140 PRINT "TO DO IS TO CHANGE THE GEAR,
    BRAKE AND ACCELERATOR"
150 PRINT 'SETTINGS AT EACH OF 22 POINTS ON
    THE CIRCUIT."
160 PRINT "THE CAR HAS FIVE FORWARD GEARS"
170 PRINT "MAX. SAFE ENGINE SPELD IS 10,000
    RPM. DON'T OVER-REV"
180 PRINT "AT EACH POINT ON THE CIRCUIT
    ENTER G, B, A."
190 PRINT TAB(6); "WHERE     G = GEAR",
    "(1-5)"
200 PRINT TAB(15); "B = BRAKE", "(0-9)"
210 PRINT TAB(6); "AND   A = ACCELERATOR",
    "(0-9)"
220 PRINT "*****PLEASE FASTEN YOUR SAFETY
    BELT*****"
230 MAT READ G
240 DATA 2.4, 1.9, 1.5, 1.2, 1
250 MAT READ L
260 DATA 0, 6, 15, 20, 25, 30, 40, 59, 80, 95
270 MAT READ J
280 DATA 180, 180, 90, 180, 180, 64, 75, 180, 125, 44,
    180, 180, 94, 180
290 DATA 73, 180, 62, 31, 180, 121, 86, 180
300 MAT READ X
310 DATA 1, 3, 5, 4, 5, 1, 3, 3, 4, 5, 3, 4, 3, 4, 5, 3, 2,
    4, 2, 4, 5, 4
320 MAT READ Y
330 DATA 1, 0, 4, 0, 0, 9, 0, 1, 1, 8, 0, 0, 5, 0, 6, 0, 9
    7, 0, 0, 6, 0
340 MAT READ Z
350 DATA 9, 9, 5, 9, 9, 3, 3, 9, 7, 2, 9, 9, 3, 9, 1, 8, 9,
    8, 9, 8, 9
360 READ K, S, W
370 DATA 0, 1, 0
380 PRINT
390 PRINT "FLAG DROPPED - YOUR'E OFF"
400 FOR I = 1 TO 22
410 PRINT "ENTER G, B, A"
420 IF J$ = "EXPERT" THEN 650
430 INPUT G1, B1, A1
440 IF A1 + B1 + G1 = INT(A1) + INT(B1) +
    INT(G1) THEN 470
450 PRINT "VALUES MUST BE WHOLE
    NUMBERS - ";
460 GOTO 550
470 IF G1 < 1 THEN 490
480 IF G1 <= 5 THEN 520
490 PRINT "PLEASE TELL THIS WOMAN DRIVER
    THAT THERE ARE GEARS"
500 PRINT "1,2,3,4, & 5, AND ASK HER TO ";
510 GOTO   550
520 IF B1 < 0 THEN 540
530 IF B1 <= 9 THEN 570
540 PRINT "DON'T CHEAT - BRAKE SETTINGS MAY
    BE 0-9 INCLUSIVE"
550 PRINT "RETYPE CORRECTLY"
560 GOTO   410
570 IF A1 >= 0 THEN 600
580 PRINT "TRY PRESSING THE ACCELERATOR
    PEDAL DOWN - ";
590 GOTO   550
600 LET B1 = B1 + 1
610 IF A1 <= 9 THEN 690
620 PRINT "YOU HAVE PRESSED THE
    ACCELERATOR THROUGH THE
    FLOORBOARDS!"
630 PRINT "TAKE YOUR BOOTS OFF AND TRY
    AGAIN"
640 GOTO   410
650 LET G1 = X(I)
660 LET B1 = Y(I) + 1
670 LET A1 = Z(I)
680 PRINT X(I); Y(I); Z(I)
690 LET S = S + A1 * 3.2 * G(G1) — L(B1) — S / 20
700 IF S > 2 THEN 730
710 PRINT "CHICKEN! YOU HAVE BRAKED TO A
    STANDSTILL."
720 GOTO, 410
730 LET R = S * G(G1) * 61
740 IF R > 2000 THEN 790
750 PRINT "ENGINE STALLED"
760 PRINT "SELECT FIRST GEAR & START HER UP
    AGAIN!"
770 LET S = 0
780 GOTO   410
790 IF R <= 10000 THEN 920
800 IF R > 11900 THEN 900
810 PRINT "*** WARNING - YOU HAVE SLIGHTLY
    OVER REVVED ONCE";
```

(continued on next page)

(continued from previous page)

```
820 IF W = 2 THEN 890
830 IF W = 1 THEN 860
840 PRINT "& GOT AWAY WITH IT"
850 GOTO  870
860 PRINT "AGAIN"
870 LET W = W + 1
880 GOTO  920
890 PRINT "TOO OFTEN!"
900 PRINT "ENGINE SEIZED WHEN YOU
     REACHED"; R; " REVS!"
910 GOTO  1090
920 IF S <* J(I THEN 1000
930 IF S < (J(I) + 10) THEN 980
940 PRINT "***** YOU HAVE CRASHED *****"
950 PRINT "YOU WERE TRYING TO CORNER AT ";
     S; " M.P.H."
960 PRINT "THE MAXIMUM SAFE SPEED FOR THIS
     SECTION IS"; J(I); " MPH."
970 GOTO  1090
980 PRINT "YOU HAVE SKIDDED AND LOST 25%
     OF YOUR SPEED"
990 LET S = S - S / 4
1000 PRINT "SPEED = ";INT(10*S)/10;" M.P.H. ,
      TACHO = ";INT(R);" R.P.M."
1010 LET K = K + S
1020 IF I = 22 THEN 1060
1030 READ M$
1040 PRINT
1050 PRINT "POINT"; I; M$
1060 NEXT I
1070 LET S1 = K / 22
1080 PRINT "LAP COMPLETED - AVERAGE SPEED
      WAS "; S1; " M'P.H."
1090 PRINT
1100 PRINT "WOULD YOU LIKE ANOTHER GO?
      (YES/NO)"
1110 INPUT Q$
1120 IF Q$ = "YES" THEN 1160
1130 IF Q$ = "NO" THEN 1510
1140 PRINT "REPLY ' "; Q$; " ' NOT
      UNDERSTOOD. PLEASE ANSWER YES OR NO"
1150 GOTO  1110
1160 RESTORE
1170 IF J$ <> "EXPERT" THEN 1200
1180 LET J$ = "NORMAL"
1190 GOTO  230
1200 PRINT "AFTER YOUR MISERABLE ATTEMPT
      AT DRIVING"
1210 PRINT "WOULD YOU LIKE A
      DEMONSTRATION BY AN EXPERT DRIVER
      (YES/NO)"
1220 INPUT J$
1230 IF J$ = "YES" THEN 1270
1240 IF J$ = "NO" THEN 230
1250 PRINT "REPLY ' "; J$; " ' NOT
      UNDERSTOOD. PLEASE ANSWER YES OR NO"
1260 GOTO  1220
1270 PRINT "NOW WATCH A LAP BY J. D. LEE !"
1280 LET J$ = "EXPERT"
1290 GOTO  230
1300 DATA "ON THE STRAIGHT.          "
1310 DATA "SLOW FOR RH BEND.         "
1320 DATA "DOWNHILL STRAIGHT.        "
1330 DATA "KEEP GOING NOW            "
1340 DATA "SHARP CORNER.             "
1350 DATA "CORNER—EASING UP.         "
1360 DATA "STRAIGHT AHEAD.           "
1370 DATA "FAST LH BEND.             "
1380 DATA "ANCHORS ON NOW.           "
1390 DATA "ALL CLEAR.                "
1400 DATA "TRY FOR 163 M.P.H.        "
1410 DATA "FAST R/H BEND.            "
1420 DATA "SHORT STRAIGHT.           "
1430 DATA "TAKE CARE.                "
1440 DATA "DOWNHILL STRAIGHT.        "
1450 DATA "TRICKY R/H BEND.          "
1460 DATA "HAIRPIN.                  "
1470 DATA "GO MAN GO.                "
1480 DATA "BANKED CURVE.             "
1490 DATA "BEND TIGHTENING UP.       "
1500 DATA "LAST CHANCE TO WIN."
1510 PRINT "JUST YOU DRIVE HOME
      CAREFULLY!"
1520 END
```

```
TRIAL RUN
      CARS
WOULD YOU LIKE INSTRUCTIONS?  TYPE YES OR
 OR NO L PRESS RETURN.
? NO

FLAG DROPPED - YOU'RE OFF
ENTER G, B, A
? 1,0,9
*** WARNING - YOU HAVE SLIGHTLY OVER
 REVVED ONCE & GOT AWAY WITH IT
SPEED = 70 M.P.H. , TACHO = 10258 R.P.M.

POINT 1 ON THE STRAIGHT.
ENTER G, B, A
? 4,0,9
SPEED = 101.1 M.P.B. , TACHO = 7402 R.P.M.

POINT 2 SLOW FOR RH BEND.
ENTER G, B, A
? 2,4,4
*** WARNING - YOUR HAVE SLIGHTLY OVER
 REVVED ONCE AGAIN
YOU HAVE SKIDDED AND LOST 25% OF YOUR -
 SPEED
SPEED = 71.5 M.P.H. , TACHO = 11055 R.P.M.

POINT 3 DOWNHILL STRAIGHT.
ENTER G, B, A
? 5,0,9
SPEED = 96.7 M.P.H. , TACHO = 5902 R.P.M.

POINT 4 KEEP GOING NOW.
ENTER G, B, A
? 5,0,9
SPEED = 120.7 M.P.H. , TACHO = 7364 R.P.M.

POINT 5 SHARP CORNER.
ENTER G,B,A
? 2,4,3
ENGINE SEIZED WHEN YOU REACHED 12509.2
 REVS!
WOULD YOU LIKE ANOTHER GO? (YES/NO)
? NO
JUST YOU DRIVE HOME CAREFULLY!
```

Sink the fleet

This program is based on the popular game in which the users fire shells attempting to sink the enemy fleet. Eight ships are hidden on an 8 × 8 grid or matrix:

(i) two aircraft carriers (each four squares long)
(ii) two battleships (each three squares long)
(iii) two cruisers (each two squares long) and
(iv) two minesweepers (each one square long).

You type the rectilinear co-ordinates (x, y) of a square you think is occupied by a ship, and are told which type of ship you have hit, or alternatively if you have missed. To sink a ship you must hit each of the squares the ship occupies.

Our intelligence organisation has intercepted a map of the enemy fleet position, which may be a true map, or in one of three codes. In the first code the true positions are reflected across a vertical mirror through the centre of the map, hence a ship actually in square 1, 6 would be shown in code on the map as 8, 6.

The second code involves reflection of the co-ordinates across a horizontal mirror through the centre of the map, hence position 2, 1 would be shown as 2, 8. The third code involves reflection across a vertical mirror followed by reflection across a horizontal mirror, and corresponds to a rotation of 180 degrees about the centre point on the board.

The game provides a good exercise in identifying coordinates in a two dimensional plane, and also in the reflection and rotation of coordinates.

The eight ships occupy 20 or the 64 squares, hence the chance of hitting a ship by firing at random is 20/64. Since you have only 25 rounds of ammunition, winning the battle depends on decoding the map very rapidly.

The computer chooses the fleet position randomly, with the ships arranged along horizontal or vertical lines, avoiding touching or intersection of ships. The code used for the first map is chosen randomly by asking the player to type in the time.

At the end of a game, the player is asked if he would like another game using a different map, and if so, whether he would like the same or a different code. At any time the user can abandon the game by firing a shot at the point 0, 0. The computer checks that:

(i) shots are aimed only at squares on the board
(ii) whole numbers are used for co-ordinates
(iii) answers to questions are either YES or NO.

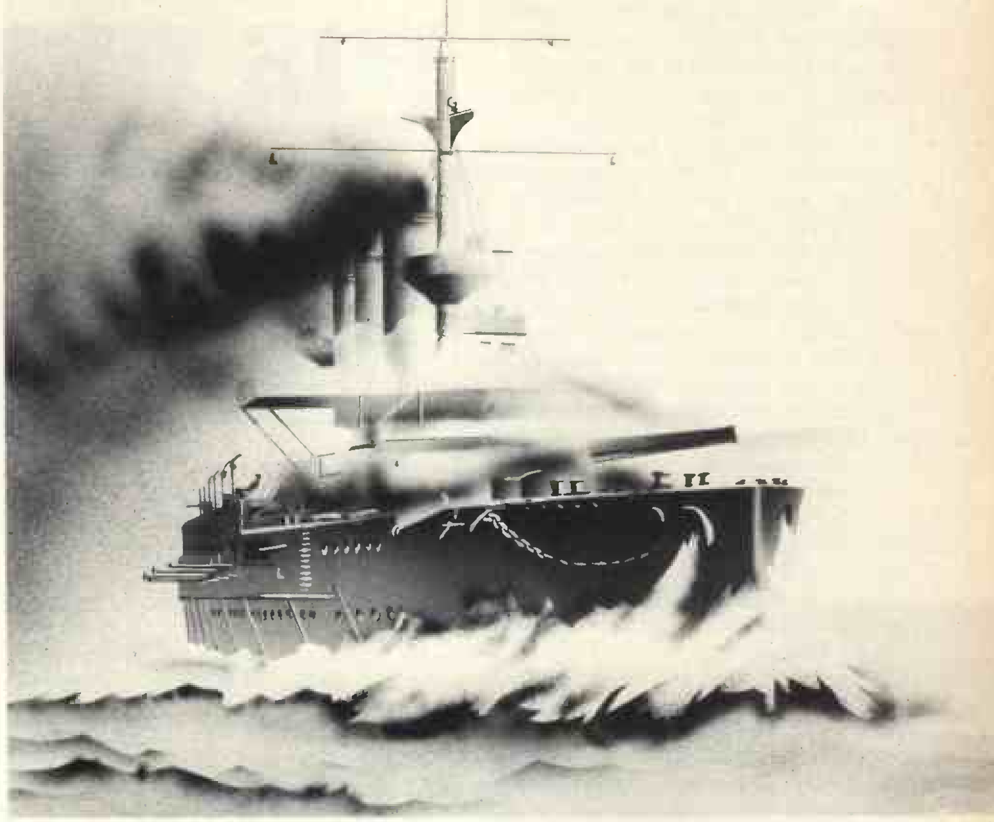
If the fleet has not been sunk in 25 shots, you have lost the battle and the computer terminates the game.

```
10 DIM A(8), B(8, 8), C(8), F(8, 8), Q$(9)
20 PRINT TAB(8); "BATTLESHIPS"
30 PRINT TAB(8); "==========="
40 LET G = 1
50 PRINT
60 REM ***RANDOMIZE
70 PRINT "TIME CHECK"
80 PRINT "PLEASE TYPE THE NUMBER OF
    MINUTES PAST THE HOUR"
90 PRINT "THEN PRESS RETURN."
100 INPUT T
110 FOR I = 1 TO ABS(T)
120 LET C1 = RND(0)
130 NEXT I
140 PRINT "THIS MAP OF THE ENEMY'S FLEET
     POSITIONS HAS BEEN CAPTURED."
150 PRINT "TRY TO DECODE THE MAP, AND USE
     IT IF YOU CAN."
160 PRINT
170 REM ***SET CODE
180 LET C1 = INT(RND(0) * 2)
190 LET C1 = 1 - C1
200 LET C2 = INT(RND(0) * 2)
210 MAT  F = ZER
220 MAT READ A
230 DATA 1. 0. 0. 1. -1, 0, 0, -1
240 REM ***SET UP THE BOARD
```

(continued on next page)

(continued from previous page)

```
250 FOR I = 8 TO 1 STEP -1
260 LET S = INT (I + 1) / 2)
270 LET X = INT(RND(0) * 8) + 1
280 LET Y = INT(RND(0) * 8) + 1
290 LET D = INT(RND(0) * 4) * 2 + 1
300 FOR J = 0 TO S - 1
310 IF X + J * A(D) = 9 THEN 270
320 IF X + J * A(D) = 0 THEN 270
330 IF Y + J * A(D + 1) = 9 THEN 270
340 IF Y + J * A(D + 1) = 0 THEN 270
350 IF F((X + J * A(D)), (Y + J * A(D + 1))) <> 0
    THEN 270
360 NEXT J
370 FOR J = 0 TO S - 1
380 LET X1 = X + J * A(D)
390 LET Y1 = Y + J * A(D + 1)
400 FOR K = 1 TO 7 STEP 2
410 IF X1 + A(K) = 9 THEN 460
420 IF X1 + A(K) = 0 THEN 460
430 IF Y1 + A (K + 1) = 9 THEN 460
440 IF Y1 + A (K + 1) = 0 THEN 460
450 LET F((X1 + A(K)), (Y1 + A (K+1))) = -1
460 NEXT K
470 NEXT J
480 FOR J = 0 TO S - 1
490 LET F((X + J * A(D)), (Y + J* A(D + 1))) = 1
500 NEXT J
510 NEXT I
520 REM ***PRINT MAP
538 FOR J = 1 TO 8
540 PRINT " ("; (9 - J); ") ";
550 LET Y1 = (J * C1 + C1) * (9 - J))
560 FOR K = 1 TO 8
570 LET X1 = (K * C2 + (1 - C2) * (9 - K))
580 IF F(X1, Y1) > 0 THEN 620
590 LET F(X1, Y1) = 0
600 PRINT " . ";
610 GOTO  630
620 PRINT P(X1, Y1);
630 NEXT K
640 PRINT
650 NEXT J
660 PRINT
670 PRINT TAB(7); "(1) (2) (3) (4) (5) (6) (7) (8)"
680 MAT B = ZER
690 IF G > 1 THEN 980
700 PRINT
710 PRINT "WOULD YOU LIKE INSTRUCTIONS?
    TYPE YES OR NO & PRESS RETURN"
720 INPUT Q$
730 IF Q$ = "YES" THEN 770
740 IF Q$ = "NO" THEN 990
750 PRINT "REPLY ' "; Q$; " ' NOT
    UNDERSTOOD. PLEASE ANSWER YES OR NO"
760 GOTO  720
770 PRINT "THE MAP GIVEN MAY BE CORRECT, OR
    IN ONE OF THREE DIFFERENT"
790 PRINT "CODES. THE POSSIBLE CODES ARE:"
790 PRINT "(A) THE POINTS ARE REFLECTED
    ABOUT A VERTICAL LINE"
800 PRINT "(B) THE POINTS ARE REFLECTED
    ABOUT A HORIZONTAL LINE"
810 PRINT "(C) BOTH (A) & (B) ARE CARRIED OUT
    SIMULTANEOUSLY"
820 PRINT "THE GRID SQUARES ARE
    NUMBERED CONVENTIONALLY, WITH
    SQUARE"
830 PRINT "1, 1 IN THE BOTTOM LEFT HAND
    CORNER. OTHER SQUARES ARE"
840 PRINT "REFERENCED BY TYPING TWO
    NUMBERS SEPARATED BY A COMMA, TO"
850 PRINT "INDICATE THE X & Y RESPECTIVELY"
860 PRINT "THE BOARD COMPRISES OF TWO
    MINE SWEEPERS      (1 SQUARE EACH)"
870 PRINT TAB(23); "TWO CRUISERS
    (2 SQUARES EACH)"
880 PRINT TAB(23); "TWO BATTLESHIPS
    (3 SQUARES EACH),"
890 PRINT TAB(19); "AND TWO AIRCRAFT
    CARRIERS (4 SQUARES EACH)."
900 PRINT "WHEN INVITED TO SHOOT A SHELL,
    YOU TYPE THE CO-ORDINATES"
910 PRINT "OF THE SQUARE YOU WANT TO HIT,
    & THE COMPUTER WILL TELL"
920  PRINT "YOU WHETHER YOU HAVE HIT,
    SUNK, OR MISSED A SHIP."
930 PRINT "TO SINK A MINE SWEEPER YOU MUST
    HIT IT ONCE, A CRUISER"
940 PRINT "MUST BE HIT TWICE, A
    BATTLESHIP THREE TIMES, & AN"
950 PRINT "AIRCRAFT CARRIER FOUR TIMES."
960 PRINT "YOU ONLY HAVE 25 SHOTS TO SINK
    THE ENEMY FLEET"
970 PRINT "TO ABANDON THE GAME TYPE IN 0,0
    WHEN INVITED TO SHOOT."
980 PRINT
996 LET L = 0
1000 MAT READ C
1010 DATA 3, 3, 2  2, 1, 1, 0, 0
1020 LET S = 0
1030 PRINT "TYPE THE CO-ORDINATES FOR FIRST
     SHOT X,Y"
1040 INPUT X, Y
1050 IF X <> 0 THEN 1090
1060 IF Y <> 0 THEN 1160
1070 PRINT "GAME ABANDONED"
1080 GOTO  1490
1090 IF X + Y = INT(X) + INT(Y) THEN 1120
1100 PRINT "DON'T CHEAT - YOU MUST USE
     WHOLE NUMBERS. TRY AGAIN."
1110 GOTO  1040
1120 IF X > 8 THEN 1160
1130 IF X < 1 THEN 1160
1140 IF Y > 8 THEN 1160
1150 IF Y >= 1 THEN 1180
1160 PRINT "SHOT OFF THE BOARD. TRY AGAIN."
1170 GOTO  1040
1180 LET S = S + 1
```



```
1190 IF F(X, Y) = 0 THEN 1350
1200 IF C(F(X, Y)) < 4 THEN 1230
1210 PRINT "WAKEY-WAKEY! YOU HAVE
     ALREADY SUNK A";
1220 GOTO  1330
1230 IF B(X, Y) > 0 THEN 1320
1240 LET B(X, Y) = F(X, Y)
1250 PRINT "DIEPCT HIT ON A";
1260 GOSUB  1670
1270 LET C(F(X, Y)) = C(F(X, Y)) + 1
1280 IF C(F(X, Y)) >= 4 THEN 1370
1290 IF S = 25 THEN 1470
1300 PRINT " - NEXT SHOT"
1310 GOTO  1040
1320 PRINT "YOU HAVE ALREADY HIT A";
1330 GOSUB  1670
1340 PRINT " AT THIS POINT."
1350 PRINT "MISS!";
1360 GOTO  1290
1370 PRINT " - AND YOU SUNK IT";
1380 LET L = L + 1
1390 IF L < 8 THEN 1290
1400 PRINT
1410 PRINT
1420 PRINT "YOU HAVE WIPED OUT THE
     ENEMY'S FLEET";
1430 PRINT " USING "; S; "SHOTS"
1440 IF S > 20 THEN 1490
1450 PRINT "YOU MUST KNOW THE CODE - A
     DIRECT HIT EVERY TIME."
1460 GOTO  1490
1470 PRINT "YOU HAVE USED UP YOUR 25
     ROUNDS OF AMMUNITION !"
1480 PRINT "WAVE THE WHITE FLAG & BEG FOR
     MERCY."
1490 PRINT
1500 PRINT "WOULD YOU LIKE ANOTHER GAME"
1510 INPUT Q$
1520 IF Q$ = "YES" THEN 1560
1530 IF Q$ = "NO" THEN 1640
1540 PRINT "REPLY ' "; Q$ " ' NOT
     UNDERSTOOD. PLEASE ANSWER YES OR NO."
1550 GOTO  1510
1560 RESTORE
1570 LET G = G + 1
1580 PRINT "WOULD YOU LIKE THE SAME CODE
     AGAIN?"
1590 INPUT Q$
1600 IF Q$ = "YES" THEN 210
1610 IF Q$ = "NO" THEN 190
1620 PRINT "REPLY ' "; Q$; " ' NOT
     UNDERSTOOD. PLEASE ANSWER YES OR NO"
1630 GOTO  1590
1640 PRINT "OK - GOODBYE"
1650 STOP
1660 REM ***SUBROUTINE TO IDENTIFY TYPE OF
     SHIP
1670 IF F(X, Y) <= 6 THEN 1700
1680 PRINT "N AIRCRAFT CARRIER";
1690 RETURN
1700 IF F(X, Y) <= 4 THEN 1730
1710 PRINT " BATTLESHIP";
1720 RETURN
1730 IF F(X, Y) <= 2 THEN 1760
1740 PRINT " CRUISER";
1750 RETURN
1760 PRINT " MINESWEEPER";
1770 RETURN
1780 END
```

TRIAL RUN

```
        BATTLESHIPS

TIME CHECK
PLEASE TYPE THE NUMBER OF MINUTES PAST THE
  HOUR
THEN PRESS RETURN.
? 13
THIS MAP OF THE ENEMY'S FLEET POSITIONS HAS
  BEEN CAPTURED.
TRY TO DECODE THE MAP, AND USE IT IF YOU
  CAN.
( 8 )  .  .  7  .  .  .  4  4
( 7 )  .  .  7     2  .  .  .
( 6 )  .  .  7  .  .  .  5  .
( 5 )  1  .  7  .  3  .  5  .
( 4 )  .  .  .  .  3  .  5  .
( 3 )  .  6  .  .  .  .  .  .
( 2 )  .  6  .  8  8  8  8  .
( 1 )  .  6  .  .  .  .  .  .
      (1) (2) (3) (4) (5) (6) (7) (8)
WOULD YOU LIKE INSTRUCTIONS? TYPE YES OR
  NO & PRESS RETURN
? NO
TYPE THE CO-ORDINATES FOR FIRST SHOT X,Y
?, 8,8
MISS! - NEXT SHOT
? 8,1
MISS! - NEXT SHOT
? 1,8
DIRECT HIT ON A CRUISER - NEXT SHOT
? 2,8
DIRECT HIT ON A CRUISER - AND YOU SUNK IT -
  NEXT SHOT
? 4,7
DIRECT HIT ON A MINESWEEPER - AND YOU
  SUNK IT - NEXT SHOT
? 8,5
DIRECT HIT ON A MINESWEEPER - AND YOU
  SUNK IT - NEXT SHOT
```

Cars and Battleships are from a new book, Computer Programs that Work. The book, which runs to 100 pages, contains 24 programs in Basic.

Computer Programs that Work is available from Sigma Technical Press, 23 Dippons Mill Close, Tettenhall, Wolverhampton, West Midlands. Price £2·40.

GADGET FREAK HIS PET COLLEC

JOHN THEODORSON knew virtually nothing about computers last February. Since then, he has taught himself to program in Basic; bought a CBM Pet and installed extensive modifications; and programmed it to deal with much of the paperwork involved in running his firm, Tametrend landscape contractors.

Tametrend tidies the landscape after the roadbuilders, town planners, gravel diggers, and so forth have dishevelled it. Theodorson runs the firm from his home in Long Buckby, Northamptonshire.

Downstairs, the house is full of animals —five dogs, four cats and a cage of white budgerigars. A number of consumer durables suggest another foible—a video camera, an elaborate TV games unit, a radiotelephone in the car. Theodorson is, he admits cheerfully, a gadget freak and a man who loves pets.

Upstairs, in the study, is perhaps the ultimate in both categories. Apart from the extra cassette drive on top, the Pet looks fairly conventional at first sight, until you notice the DECwriter printer alongside it connected by an interface box on the window-cill.

Open the casing and the other major modification is revealed—a 24K memory board on top of the original CBM board. Between them, these add-ons convert the Pet from a useful beginner's computer into a powerful business system.

In addition, Theodorson has added two minor enhancements. One is a software-controlled bleeper unit, which can be used to signal to the operator, allowing him, for example, to get on with something else during a long print run. The other is a joystick, at present used mainly for "zapping klingons", but intended for more productive graphic uses in the future. All these devices have been developed with help from Mick Hambly of HB Computers of Kettering, who, we understand, will be offering some of them for sale.

by **HUGH BUSBY**

The system has developed to its present state by a classic "suck-it-and-see" process. Impressed by what he had read about microcomputers, Theodorson visited an exhibition in February to discover what they could do to relieve some of the drudgery of his own business.

He looked at several machines, but was impressed by what he calls the "professional presentation" of the CBM machine. In other words, it is a neat, self-contained unit, rather than an assortment of boxes connected by a jumble of cables; it springs to life when you plug it in; and it can be understood and explained without the need for a barrage of buzz-words.

At the time, Theodorson knew very little of what computers could do for him or what he wanted from them. Since then, he has learned a great deal both about computing and about the limitations of the Pet. He still feels the Pet is the ideal beginners' computer for those who want to advance in programming rather than delve into the mysteries of electronics.

Beginner's choice

"If you put me back to square one, knowing what I know now", he says, "then, no, I probably wouldn't buy a Pet. As an absolute beginner, though, I still don't think I could have made a better choice".

The Pet, ordered from the stand at the exhibition, was delivered in April and was, he thinks, the second to be installed in the U.K. The interval permitted time for learning Basic and the first project, a

(continued on next page)

Pets of all kinds—and they're not all computers. John Theodorson has five dogs, four cats and four budgerigars.

ADDS TO TION

(continued from previous page)
program to deal with VAT returns, was under way rapidly.

Programming seemed to take "weeks and weeks and weeks" but was probably more like two weeks. Programming the hard core of calculations was straightforward enough and took about a day. Most of the time was spent dealing with unfamiliar problems such as data formatting and in 'polishing' the program, improving its efficiency and adding extra facilities.

At this point the limitations of the Pet began to show. The VAT program worked well but required all results to be copied from the visual display. Theodorson soon became disenchanted with that and started looking for a printer. The DECwriter, a 30 characters-per-second LA36, arrived from Data Design Techniques of North London in May.

Transformation

The transformation was dramatic. The chore which used to take two-and-a-half-days per quarter now takes some two-and-a-half hours and is virtually guaranteed to be error-free.

At the same time, each new stage of development seemed to reveal further limitations in the system. One small problem concerned the printer. While the Pet software recognises the printer, it allows only half the full range of ASCII commands to be used in addressing it. The result is that lower-case characters cannot be used, although they are available on the DECwriter. CBM is circumventing the problem by using an intelligent printer.

Another problem is the keyboard, which is not really suitable for the input of large amounts of data, particularly by a trained typist. The keyboard of the DECwriter is much more suitable and the bi-directional interface, from Robin Bailey Associates, allows it to be used as a replacement.

Expansion

On the software side, the next problem to be tackled was payroll. Despite a relatively small staff—eight permanent plus half-a-dozen subcontractors—Theodorson found that doing it manually occupied most of Friday, allowing for interruptions.

Running it on the computer, of course, eliminates nearly all the manual work but even so is a fairly lengthy business, due to the speed of the tapes, the need to update and copy, and the fact that a separate file has to be set up for each employee.

That, together with Theodorson's plans for other applications, led him to think of further expansion, such as extra memory and disc drives. The memory, a 24K RAM board from Robin Bailey Associates, was no problem, except that he had to wait until August for delivery.

With no prospect of disc drives from CBM in the foreseeable future and nothing else available in the U.K., Theodorson shopped around in the U.S. He settled finally on a Horizon system, using Shugart discs and an S-100 bus. Interfacing it to the Pet required an IEEE to S-100 adapter, available from Huh Electronics in the U.S., which also supplied the bleeper unit.

Games, too

Despite a background as an officer in the Royal Corps of Signals, Theodorson claims that he is not particularly hardware-orientated. "I spent all my time in the army charging around in canoes", he says, "so I wound up as an administrator".

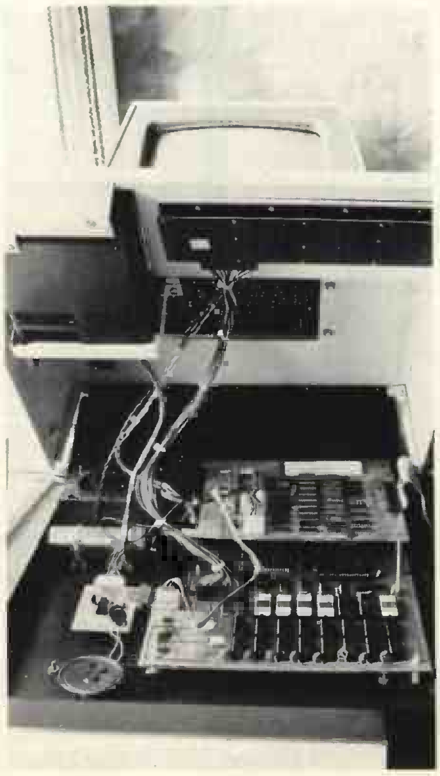
All the installation work he has done so far, he points out, requires nothing more than the most basic skill with a soldering iron, and very little of that. Building the adapter board for the disc drives will be the most elaborate job he has done.

The next software projects are sales and purchase ledgers, using a full double-entry system, and invoicing. In addition to its business uses, Theodorson has found time to program a good deal of fun into the Pet, and a sizeable chunk of his program library consists of games programs. Some are bought from the States, of which his favourites are an eight-level chess program and a bridge program, both from Personal Software.

Theodorson's system, showing the printer interface. Pet with extra cassette drive, joystick and printer.

★

Inside the Pet. The 24K add-on memory is purpose-built. At left is the bleeper unit.



WE CAN HELP SAVE YOU BOTH WITH A MICROCOMPUTER

Let us advise you as to the best system whether it be for business, industry, medicine or education.
We can install a complete system ready to go and easy to use.

WE ARE AGENTS & SPECIALISTS FOR THE A-P-P-L-E II

MICROSOLVE

Microsolve Computer Services Limited
252 Hale Lane, Edgware, Middx HA8 8NT
Telephone: 01-958 4347

● **Circle No. 134**

MICROCOMPUTERS

CASH & CARRY

(PET and KIM off the shelf)

OR

TOTAL TURNKEY

(and everything in between)

Full sales and technical support from our team of Systems Engineers, Programmers, Business Analysts,

Software Advisors, Hardware Demonstrators, plus Telephone Emergency Service and Fast Field Support.

The "BIG TEAM" for the "LITTLE COMPUTER"
Visit our showrooms and meet the Team

DATA EFFICIENCY LIMITED
Microsense Computers Division
Maxted Road Maylands Avenue
Hemel Hempstead Herts HP2 7LE
Telephone: Hemel Hempstead (0442) 41191 (3 lines)
Telex: 825554 DATEFF G

● **Circle No. 135**

INTERESTED IN HOME COMPUTING?

nm
Nascom Microcomputers

Start now and don't get left behind
THE NASCOM 1 is here
Ex-stock with full technical services

BLANK C12
Racal Quality
CASSETTES
£4.90 for 10

Plus the opportunity to join the fastest moving club of personal computer users enabling you to get the most our of your computer. You can OBTAIN and EXCHANGE programs and other software — many now available.

The Powerful Z80 Microprocessor
Professional Keyboard
1 Kbyte Monitor in EPROM
2 Kbyte RAM (expandable)
Audio Cassette interface
Plugs into your domestic TV
Easy construction from straightforward instructions
— no drilling or special tools
— Just neat soldering required.

Only £197.50 + 8% VAT (includes p & p + insurance)

Manuals seperately	3.50	Monitor quality improved	
Z80 programming Manual	6.90	TV Modulator	2.50
Z80 Technical Manual	3.40	Power supply suitable for	
PIO Technical Manual	3.40	NASCOM	19.90

(All prices add 8% VAT)

COMPUTER COMPONENTS

A selection of our Computer components

CPU AND SUPPORT DEVICES

Z80	cpu	14.90
Z80A	cpu	20.50
Z80	PIO	9.90
Z80	CTC	9.90
8080A	cpu	7.95
8212	8 Bit I/O Port	3.50
8214	Priority Interrupt Control	9.00
8216	Bi-Directional Bus Driver	3.75
8224	Clock Generator & Driver ($2MH_z$)	3.50
8224-4	Clock Generator & Driver ($4MH_z$)	9.95
8226	Inverting Bi-Directional Bus Driver	3.95
8228	System Controller & Bus Driver	7.95
8238	System Controller & Bus Driver	7.50
8251	Programmable Communication Int.	9.95
8253	Programmable Interval Timer	18.95
8255	Programmable Peripheral Int.	12.00
8257	Prog. Direct Mem. Access Cont.	17.95
8259	Programmable Interrupt Cont.	17.95
6800	cpu	11.90

MEMORIES

2102	1K	1.20
21L02	1K	1.40
2101	1K	2.95
2111	1K	3.95
2112	1K	2.95
2114	4K	9.95
4116	16K	19.90
1702A	256B	5.00
2708	1K	10.90
2708	(NASBUG)	13.90
2716	(5V, INTEL)	29.00

OTHER USEFUL DEVICES

6576	char gen	10.90
6402	UART	9.90
1771	Floppy disk controller	49.00

Full selection of US and British Magazines and books for the Computer Hobbyist.

AT LAST A COMPUTER JOY STICK
Complete with hardware which neatly packs into a stylish plastic joystick case, plugging directly into your **NASCOM P10** port with no extras. Full documentation is supplied. Each joy stock is supplied with a test routine that allows you to draw pictures on your VDU and step routines to write your home games. Plus one free game on cassette.
I joystick £14·90 I pair of joy sticks £28·90 includes VAT, post and packing
The **NASCOM SOFTWARE USERS BOOK** is being compiled; if you would like your program included send it to us with an s.a.e., in return you will get a personal copy of the book.
B BUG monitor for **NASCOM** users. Well worth the money. Send s.a.e. for details. £23·40 including VAT, post and packing.

BARCLAYCARD

All prices include VAT except where shown. Orders over £5 post and packing free otherwise add 20p. Please make cheques and postal orders payable to COMP. or phone your order quoting BARCLAYCARD or ACCESS number.

Buy it with Access

COMP COMPUTER COMPONENTS

14 STATION ROAD · NEW BARNET · HERTFORDSHIRE · TEL: 01-441 2922
01-449 6596
CLOSE TO NEW BARNET BR STATION — MOORGATE LINE
OPEN – 10am to 7pm – Monday to Saturday

● **Circle No. 136**

ILLUSTRATING BASIC

(A SIMPLE PROGRAMMING LANGUAGE)

WE CONTINUE our series of articles on how to program in Basic, probably the most widely-used programming language for small computers. For the series, we have obtained the serialisation rights for one of the best books on the subject, *Illustrating Basic* by Donald Alcock.

★

Each month, we are publishing a part of the book, so by the end of the series you will have the complete book. It is written with a distinct informality and has a rather unusual presentation; but it is this style, we believe, which makes it one of the most easy to read tutorials.

★

★

Copies of *Illustrating Basic* can be obtained from *Practical Computing*.

CONTENTS

PRINT

THIS IS THE MOST VERSATILE STATEMENT IN *BASIC* AND ALSO ONE WHICH DIFFERS MUCH IN INTERPRETATION FROM ONE *BASIC* TO ANOTHER.

THE FOLLOWING PROGRAM PRINTS THE SQUARE OF ANY NUMBER:

```
10 LET  E$ = "EQUALS"
20 PRINT "TYPE A NUMBER"
30 INPUT  A
40 PRINT "THE SQUARE OF"; A ; E$ ; A*A
50 END
RUN
TYPE  A NUMBER
?   4
THE  SQUARE  OF  4  EQUALS  16
```

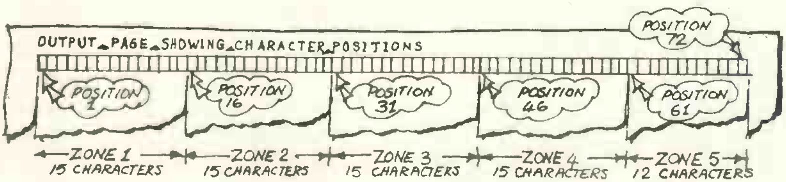
NUMERICAL VARIABLE — EXPRESSION — TEXT OF ANY LENGTH — TEXTUAL VARIABLE

LINE 40 SHOWS EXAMPLES OF FOUR OF THE POSSIBLE THINGS THAT CAN BE PUT IN THE LIST FOLLOWING THE WORD PRINT:

- ★ "THE SQUARE OF" : A TEXT (ANY LENGTH THAT WILL FIT THE LINE) WHICH GETS PRINTED EXACTLY AS TYPED BUT WITHOUT THE QUOTATION MARKS.
- ★ A : A NUMERICAL VARIABLE WHOSE *VALUE* GETS PRINTED — NOT ITS NAME.
- ★ E$: A TEXTUAL VARIABLE WHOSE *TEXT* GETS PRINTED.
- ★ A*A : AN EXPRESSION WHOSE *VALUE* GETS PRINTED.

A PRINT STATEMENT MAY HAVE ANY MIXTURE OF TEXTS, VARIABLES AND EXPRESSIONS AS LONG AS THE LIST WILL FIT THE LINE. THEY ARE SEPARATED FROM ONE ANOTHER BY *SEMICOLONS* (AS ABOVE) OR BY *COMMAS* WHICH CAUSE *BASIC* TO PRINT ITEMS IN *ZONES* AS DESCRIBED BELOW.

THE PICTURE BELOW SHOWS A PAGE WITH 72 POSITIONS ACROSS IT. *BASIC* DIVIDES THE PAGE INTO 4 ZONES, EACH OF 15 CHARACTER POSITIONS, AND ONE ZONE OF 12. (*BASICS* DIFFER GREATLY HERE BOTH IN WIDTH OF PAGE AND IN WIDTH OF ZONE, BUT THE CONCEPT IS UNIVERSAL.) THE CHARACTER POSITIONS ARE NUMBERED FROM 1, AS DONE BY SOME *BASICS*, BUT MANY *BASICS* NUMBER FROM ZERO (*i.e.* 0 TO 71 INSTEAD OF 1 TO 72).



```
1 PRINT  A; B, C
```

A *SEMICOLON* TELLS THE PRINTING HEAD *NOT TO MOVE* BEFORE PRINTING THE NUMBER OR TEXT SIGNIFIED (IN THIS CASE THE VALUE OF B). IF THERE ISN'T ENOUGH ROOM ON THE LINE THEN THE PRINTING HEAD RETURNS TO START A NEW LINE.

```
2 PRINT  A; B , C
```

A *COMMA* TELLS THE PRINTING HEAD TO MOVE TO THE FIRST POSITION IN THE NEXT AVAILABLE ZONE BEFORE PRINTING THE NUMBER OR TEXT SIGNIFIED (IN THIS CASE THE VALUE OF C). IF THE PRINTING HEAD IS ALREADY SOMEWHERE IN ZONE 5 THEN THE NEXT AVAILABLE ZONE IS ZONE 1 ON THE FOLLOWING LINE.

```
3 PRINT  A; B,
4 PRINT  C, D
```

THEN WHAT ABOUT THE *FIRST* ITEM IN A LIST (SUCH AS C) ? NO EXCEPTION : *BASIC* ACTS ON THE COMMA OR SEMICOLON AT THE END OF THE *PREVIOUS* LIST, EXACTLY AS THOUGH YOU HAD TYPED (IN THIS CASE) :

```
3 PRINT  A; B ,C, D
```

```
5  PRINT  E; F
6  PRINT  G, H
```

WHAT IF THERE IS *NO* COMMA OR SEMICOLON AT THE END OF THE PREVIOUS LIST — OR NO PREVIOUS LIST AT ALL ?

ANSWER: AFTER OBEYING A PRINT STATEMENT HAVING NO COMMA OR SEMICOLON AT THE END OF ITS LIST THE PRINTING HEAD RETURNS TO START A NEW LINE. YOU MAY ALSO ASSUME THIS HAPPENS BEFORE *BASIC* OBEYS THE VERY FIRST PRINT STATEMENT.

```
7  PRINT
```

WHAT IF THERE IS AN *EMPTY* LIST ? THEN *BASIC* PRINTS NOTHING ON THE LINE, AND BECAUSE THERE IS NO COMMA OR SEMICOLON THE PRINTING HEAD RETURNS TO START A NEW LINE AS DESCRIBED ABOVE. IN SHORT : *THIS IS THE WAY TO MAKE BASIC PRINT A BLANK LINE.*

```
8 PRINT  I "SQD=" I*I
```

SOME *BASICS* ALLOW YOU TO LEAVE OUT THE PUNCTUATION ON EITHER SIDE OF A TEXT — ASSUMING EITHER A COMMA OR SEMICOLON DEPENDING ON THE PARTICULAR VERSION. DON'T DO IT.

HERE IS AN EXAMPLE SHOWING THE USE OF COMMAS & SEMICOLONS.

```
10  DATA    "DAYS", 28, 30, 31
20  READ     T$,    A , B, C
30  PRINT  "JAN", "FEB", "MAR", "APR"
40  PRINT  , "(29 IN LEAP YEARS)"
50  PRINT  C; T$, A; T$, C; T$, B; T$
60  END
RUN
JAN             FEB             MAR             APR
                (29 IN LEAP YEARS)
 31 DAYS         28 DAYS         31 DAYS         30 DAYS
```

NOTE THE COMMA AT THE START OF THE LIST, CAUSING A "SKIP" TO THE SECOND ZONE.

PRINT (CONTINUED)

WHY DOES *BASIC* PRINT

THE SQUARE OF 4 EQUALS 16

AND NOT "THE SQUARE OF 4.00000 EQUALS 16.0000" (REMEMBERING *BASIC* DOES ARITHMETIC TO AT LEAST 6 FIGURES) ?

THE ANSWER IS THAT *BASIC* ASSUMES YOU ARE NOT INTERESTED IN "TRAILING ZEROS". *BASIC* TAKES OTHER DECISIONS ABOUT THE WIDTH AND STYLE OF PRINTED NUMBERS AS EXPLAINED BELOW.

IF A NUMBER CAN BE ACCURATELY EXPRESSED AS AN INTEGER OF SIX DIGITS OR LESS THEN *BASIC* PRINTS IT AS AN *INTEGER*.

```
10 LET   A = 654321
20 PRINT   A ; -A
30 PRINT   0.00; 20.0; 20.00002
```

SEE THE KEY BELOW FOR ▼ & ▲

```
▼654321▲-654321
▼0▲▼20▲▼20
```

IF A NUMBER IS SMALLER (NEARER TO ZERO) THAN 0.1 *BASIC* PRINTS IT IN *E-FORM*. MOST *BASICS* PRINT ONE DIGIT BEFORE THE DECIMAL POINT AND FIVE AFTER, BUT SOME PRINT ALL SIX DIGITS AFTER THE POINT. THEN MOST *BASICS* PRINT AN E FOLLOWED BY A PLUS OR MINUS SIGN FOLLOWED BY TWO DIGITS FOR THE EXPONENT.

```
40 PRINT   A/10↑7 ;   -A / 10↑9
50 PRINT   0.060 ;   -0.0006
```

```
▼6.54321E-03▲-6.54321E-04▲
▼6.00000E+02▲-6.00000E-04▲
```

BASIC ALSO PRINTS BIG NUMBERS (10^6 AND BIGGER) IN *E-FORM*.

```
60 PRINT   100*A ;
70 PRINT   -1000000000*A
```

```
▼6.54321E+07▲-6.54321E+13▲
```

BASIC PRINTS NUMBERS BETWEEN 0.1 AND 10^6 IN *DECIMAL FORM*.

```
80 PRINT   A/10 ;   -A/100
90 PRINT   -A/1000   ;  A/1000000
```

```
▼65432.1▲-6543.21▲
-654.321▲▼.654321▲
```

THIS IS WHAT *BASIC* DOES WHEN PRINTING A NUMBER:

- ▼ PRINTS A MINUS SIGN IF THE NUMBER IS NEGATIVE, OTHERWISE A SPACE;
- # THEN PRINTS AN *INTEGER* OR AN *E-FORM* OR A *DECIMAL NUMBER* DEPENDING ON SIZE AND PRECISION AS EXPLAINED ABOVE;
- ▲ THEN PRINTS ONE TRAILING SPACE.

SOME *BASICS* THEN ADD *TWO* OR *ONE* OR *NO* FURTHER TRAILING SPACES TO MAKE THE TOTAL NUMBER OF PRINTED CHARACTERS A MULTIPLE OF *THREE*: OTHERS ALWAYS PRINT TWO TRAILING SPACES.

MOST *BASICS* PRINT *TEXTS* AND *TEXTUAL VARIABLES* WITHOUT ADDING TRAILING SPACES AND WITHOUT STRIPPING TRAILING SPACES FROM THE TEXTS THEMSELVES.

```
100 DATA  "CATCH" , "DALMATIANS" , "STEPS"
110 READ     C$   ,     D$       ,   S$
120 DATA   22.00  ,    1.01E2    ,   39
130 READ     C    ,     D        ,   S
140 PRINT C$; C   ,    D; D$     ,   S; S$
150 PRINT
160 PRINT C$; -C  ,    S; S$     ,   -D; D$
170 END
RUN
```

OUTPUT FROM LINE 140 ON

```
CATCH▼22▲      ▼101▲DALMATIANS▼39▲STEPS
                                              ← BLANK LINE
CATCH-22▲      ▼39▲STEPS      -101▲DALMATIANS
|← ZONE 1 →|← ZONE 2 →|← ZONE 3 →|← ZONE 4 →|← ZONE 5 →|
```

TYPE AND RUN THE PROGRAM ABOVE (FROM LINE 10 TO LINE 170) TO SEE IF YOUR VERSION OF *BASIC* BEHAVES DIFFERENTLY. FROM VERSIONS THAT MAKE THE LENGTH OF EVERY NUMBER A MULTIPLE OF THREE CHARACTER POSITIONS THE OUTPUT WOULD BE:

```
CATCH▼22▲▲▲    ▼101▲▲DALMATIANS                ▼39▲▲▲STEPS

CATCH-22▲▲▲    ▼39▲▲▲STEPS    -101▲▲DALMATIANS
```

TAB( );

MOST *BASICS* HAVE THE TAB FUNCTION WHICH MAY BE USED WITH **PRINT**.

TO VARY THE LAYOUT OF RESULTS *UNDER CONTROL OF A PROGRAM* YOU MAY PUT TAB FUNCTIONS IN THE LIST THAT FOLLOWS THE WORD PRINT.

```
10  PRINT  "GIVE ME A VALUE FOR A"
20  INPUT  A
30  PRINT TAB(A); "MAN"; TAB(2*A); "EATING"; TAB(3*A+5);"FISH"
40  END
RUN
GIVE ME A VALUE FOR A
? 3
  MANEATING  FISH
RUN
GIVE ME A VALUE FOR A
? 5
    MAN  EATING    FISH
```

POSITION 5 — POSITION 2×5=10 — POSITION 3×5+5=20

A TAB FUNCTION ONLY MAKES SENSE AS AN ITEM IN A LIST AFTER THE WORD PRINT. YOU MAY PUT A VARIABLE OR EXPRESSION INSIDE THE BRACKETS; *BASIC* WORKS OUT ITS VALUE AND MOVES THE PRINTING HEAD TO THE CHARACTER POSITION GIVEN BY THE RESULT. THERE ARE SOME TRICKY POINTS TO WATCH WHEN USING TAB:

★ SOME *BASICS* NUMBER CHARACTER POSITIONS FROM 1 (PAGE 28) BUT MANY *BASICS* NUMBER THEM FROM ZERO. YOU CAN MAKE PROGRAMS "PORTABLE" BY ALWAYS NUMBERING FROM 1 AND NEVER USING POSITIONS GREATER THAN 71.

★ SOME *BASICS* USE THE *NEAREST* INTEGER TO THE RESULT OF THE EXPRESSION WHEREAS MANY *BASICS* TAKE THE *INTEGRAL PART*. IF YOUR FUNCTION COULD YIELD A NON-INTEGRAL RESULT USE INT TO *ENSURE* THE RESULT YOU INTEND. *e.g.*

```
;TAB(INT(P/3+.5));
```

★ SOME *BASICS* DISREGARD THE PUNCTUATION MARK FOLLOWING A TAB( ) e.g. TAB(6); IS TREATED THE SAME WAY AS TAB(6), BUT OTHERS TREAT A COMMA AFTER TAB AS AN ERROR; YET OTHERS ACT ON SUCH A COMMA BY MOVING THE PRINTING HEAD TO THE NEXT ZONE. SO ALWAYS USE SEMICOLONS AFTER TAB( );.

★ IF THE PRINTING HEAD HAS ALREADY PASSED THE POSITION EVALUATED BY THE APPROPRIATE TAB FUNCTION ⇆ OR IF THE EVALUATED POSITION IS OFF THE PAGE ⇆ DIFFERENT *BASICS* TAKE DIFFERENT ACTIONS USUALLY RESULTING IN MESSY OUTPUT. DON'T RELY ON SPECIFIC INTERPRETATIONS; GET THE EXPRESSIONS RIGHT.

★ TAB( ) MUST STAND ALONE AS AN ITEM IN THE LIST; IT MAY NOT BE COMBINED IN AN EXPRESSION HOWEVER SIMPLE THE EXPRESSION:

```
100 PRINT   4*TAB(A*6); X
```

TAB() IS USEFUL FOR PLOTTING CRUDE GRAPHS ON THE OUTPUT PAGE. FOR AN EXAMPLE OF THIS WE HAVE TO ANTICIPATE PAGE 48 WHICH EXPLAINS WHY STATEMENT 30 BELOW LETS X TAKE SUCCESSIVE VALUES OF 0, 15, 30 *etc. to* 180.

```
10 PRINT  "THE GRAPH OF COSINE(X) FOR X = 0 TO 180";
15 PRINT  " IN STEPS OF 15 DEGREES"
20 PRINT  " X"; TAB(62);"COS(X)"
30 FOR  X = 0 TO 180 STEP 15
40 LET  C = COS(X * 3.141592 / 180)
50 LET  P = INT(30 * ABS(C)+.5) * SGN(C)
60 PRINT  X; TAB(P + 36); "*"
70 NEXT X
80 END

RUN

THE GRAPH OF COSINE(X) FOR X = 0 TO 180 IN STEPS OF 15 DEGREES
 X
 0                                                            COS(X)
                                                                  *
 15                                                              *
 30                                                           *
 45                                                      *
 60                                                *
 75                                         *
 90                                 *
 105                        *
 120                 *
 135           *
 150      *
 165  *
 180 *
```

CONVERT TO RADIANS & GET COSINE IN C

COS(0) IS SCALED TO 30 PRINT POSITIONS & ROUNDED TO THE NEAREST POSITION

VERTICAL AXIS IS OFFSET 36 POSITIONS TO MIDDLE OF PAGE

PRINT USING

NOT EVERY *BASIC* HAS THIS STATEMENT ⇔ DETAILED RULES VARY AMONG THOSE THAT DO.

IT IS A DEVICE FOR BUILDING *IMAGES* OF THE DESIRED OUTPUT.

```
10 LET G=1000.00
20 PRINT USING 30 ,G,G/20
30:####GUILDERS! COME, TAKE ##!
40 END

RUN

1000 GUILDERS! COME, TAKE 50!
```

COLON : FOLLOWED BY AN "IMAGE"

ONE BASIC USES % NOT :

LINE NUMBER OF "IMAGE" TO BE USED

LIST STARTING WITH A COMMA

"MOULD" FOR G

"MOULD" FOR G/20

THE PRINT USING STATEMENT POINTS TO ANOTHER LINE OF THE PROGRAM HOLDING AN *IMAGE* OF WHAT IS TO BE PRINTED. THIS IMAGE MAY CONTAIN *MOULDS* (USUALLY CALLED *FORMATS*). EACH MOULD DETERMINES THE PLACE AND SHAPE OF ITS CORRESPONDING ITEM IN THE PRINT LIST WHEN PRINTED.

AFTER THE WORDS "PRINT USING" TYPE THE NUMBER OF THE LINE OF PROGRAM WHERE THE IMAGE IS TO BE FOUND. (THIS MAY BE ANYWHERE IN THE PROGRAM AND SEVERAL PRINT STATEMENTS MAY USE IT.) THEN TYPE A *COMMA* † WHICH STARTS A LIST OF VARIABLES OR EXPRESSIONS SEPARATED BY *COMMAS*. DON'T TYPE A COMMA AT THE END OF THE LIST; THE PRINTING HEAD RETURNS AUTOMATICALLY TO START A NEW LINE WHEN A PRINT USING STATEMENT HAS BEEN OBEYED. (CONVERSELY THE PRINTING HEAD *DOESN'T* START A NEW LINE *BEFORE* SUCH A STATEMENT IS OBEYED; IT STAYS WHERE IT WAS LEFT BY THE PREVIOUS "PRINT" OR "PRINT USING".)

FOR THE *IMAGE* LINE: TYPE A *COLON* AFTER THE LINE NUMBER. THEN TYPE EXACTLY WHAT YOU WANT THE COMPUTER TO PRINT ⇔ BUT REPRESENT EACH *DIGIT* BY # IN ORDER TO FIX A *MOULD* FOR THE SHAPE OF THE PRINTED NUMBER. IF YOU WANT A DECIMAL POINT PRINTED THEN PUT A DECIMAL POINT INSIDE THE MOULD.

```
10:FL####.##! COME, TAKE FL###.##!
20 PRINT USING 10 , 1000,50
30 END

RUN

FL 1000.00! COME, TAKE FL 50.00!
```

SEE FOOTNOTE

FL IS THE CURRENCY SIGN FOR GUILDERS

† N.C.C. "STANDARD BASIC" SAYS A COLON: IT WOULD BE MUCH NICER THAN A COMMA BUT I HAVE STILL TO FIND A *BASIC* THAT ALLOWS A COLON.

IF THE PROGRAM COMPUTES NUMBERS TOO BIG TO FIT THEIR MOULDS THEN *BASIC* OBJECTS: SOME *BASICS* BY FILLING THE MOULD WITH ASTERISKS, SOME BY CHANGING OR EXTENDING THE MOULD, SOME BY STOPPING EXECUTION ALTOGETHER.

```
###.##
999.99
-99.99
```

THIS MOULD CAN COPE WITH POSITIVE NUMBERS AS BIG AS 999.99 AND NEGATIVE NUMBERS AS BIG AS -99.99

NO HARM IS DONE BY MAKING THE MOULDS LONGER THAN STRICTLY NECESSARY.

```
5:INTEGRAL ##   DECIMAL ###.##
10 LET A= 9.999
15 PRINT USING 5     ,A,A
20 END
RUN

INTEGRAL 9  DECIMAL  10.00
```

INTEGRAL PART OF 9.999

9.999 "ROUNDED" TO TWO DECIMALS

BASIC FILLS INTEGER MOULDS FROM RIGHT TO LEFT: ####

BASIC FILLS DECIMAL MOULDS FROM THE POINT OUTWARDS: ###.##

BASIC PUTS THE *INTEGRAL PART* OF A NUMBER INTO AN INTEGER MOULD. *BASIC* *"ROUNDS"* A NUMBER DESTINED FOR A DECIMAL MOULD TO THE NUMBER OF PLACES SIGNIFIED.

IF THE PRINT LIST OFFERS MORE ITEMS THAN THE IMAGE CAN DIGEST THEN *BASIC* USES THE IMAGE AGAIN AND AGAIN UNTIL THE LIST IS CONSUMED.

```
10  :## BARLEY LOAVES & ## SMALL FISHES
20  PRINT USING 10    ,5,2,10,4,20
30  END

RUN

 5 BARLEY LOAVES &  2 SMALL FISHES
10 BARLEY LOAVES &  4 SMALL FISHES
20 BARLEY LOAVES &
```

THIS EXAMPLE ALSO SHOWS WHAT HAPPENS WHEN THE LIST OFFERS *TOO FEW* ITEMS FOR THE IMAGE. PRINTING STOPS AT THE FIRST FRUSTRATED MOULD.

SO FAR

THE RULES ARE THE SAME FOR NEARLY ALL VERSIONS OF *BASIC* THAT OFFER PRINT USING.

(AT LEAST ONE, HOWEVER, DEMANDS THE FORMS SHOWN HERE: THESE ARE *OPTIONAL* FACILITIES IN SEVERAL *BASICS*).

```
100 LET A$="ANS=##.#"
110 PRINT USING A$ ,X
```

```
100 PRINT USING "ANS=##.#",X
```

THE FACILITIES & RULES OVERLEAF ARE MORE VARIABLE FROM ONE *BASIC* TO ANOTHER THAN THOSE EXPLAINED SO FAR: FOR THE SAKE OF "PORTABILITY" IT WOULD BE BEST TO AVOID THOSE OVERLEAF.

PRINT USING (CONTINUED)

YOU CAN MAKE MOULDS FOR PRINTING NUMBERS IN *E-FORM* BY ADDING UP-ARROWS TO A DECIMAL MOULD. FOR ANY ONE VERSION OF *BASIC* THE NUMBER OF ARROWS IS FIXED: IN SOME FOUR; IN OTHERS FIVE (AND SOME *BASICS* USE AN EXCLAMATION MARK IN PLACE OF AN ARROW).

-34.5600 ⇒ #.###↑↑↑↑ ⇒ -.346E+02

THERE MUST BE AT LEAST ONE # BEFORE THE DECIMAL POINT TO RESERVE A POSITION FOR THE SIGN.

```
10:***A TITLE***
20:
30 PRINT USING 10
40 PRINT USING 20
50 END

RUN

***A TITLE***
```

BLANK LINE BY USING 20

IN SOME *BASICS* YOU MAY HAVE AN EMPTY PRINT LIST — EVEN AN EMPTY IMAGE.

(THESE THINGS ARE EASIER DONE BY PRINT WITHOUT THE USING.)

SOME *BASICS* ALLOW CURRENCY SIGNS IN THE MOULD.

1.234		$1.23
23.456	⇒ $$$.## ⇒	$23.46
345.678		$345.68

THE CURRENCY SIGN (POSSIBLY £ IN THE U.K.) "FLOATS" TO THE LEFT. SOME *BASICS* PERFORM THE SAME TRICK WITH ASTERISKS.

SOME *BASICS* ALLOW† A PLUS OR MINUS SIGN IN FRONT OF THE MOULD *e.g.* +##.# AND -#.##

\+ SAYS "PRINT A PLUS SIGN IN FRONT OF THE NUMBER IF IT IS POSITIVE; A MINUS SIGN IF NEGATIVE"

\- SAYS "PRINT A *SPACE* IN FRONT OF THE NUMBER IF IT IS POSITIVE; A MINUS SIGN IF NEGATIVE"

IT IS NOT ALWAYS CLEAR FROM THE MANUALS WHETHER THIS FACILITY CAN BE USED IN CONJUNCTION WITH THE CURRENCY SIGN.

† *N.C.C. "STANDARD BASIC" REQUIRES A SIGN IN FRONT OF THE MOULD.*

IN MOST *BASICS* OFFERING PRINT USING IT IS POSSIBLE TO MAKE MOULDS FOR *TEXTUAL* VARIABLES. THESE ARE EXTREMELY USEFUL BUT UNFORTUNATELY THEIR DETAILS DIFFER.

```
10  REM     COMMON  START
20  LET  T$= "TO"
30  LET  B$= "BE"
40  PRINT USING 50  ,T$,B$,T$,B$
```

SEE FOUR DIFFERENT LINES 50 BELOW

```
60  REM     COMMON  ENDING
70  END

RUN
```

DIFFERENT LINES 50 FOR DIFFERENT VERSIONS OF *BASIC* ARE ILLUSTRATED BELOW. THE RESULT PRODUCED BY USING EACH IMAGE IS SHOWN IMMEDIATELY BELOW THAT IMAGE.

```
50: ##   ### OR NOT #.#   #
   TO  BE    OR NOT TO    B
```

FILLED FROM LEFT TO RIGHT

DECIMAL POINT (AND #) ACT THE SAME WAY AS #

TRUNCATION TO THE RIGHT IF TEXT IS TOO LONG FOR MOULD

```
50: 'LL  'RRR OR NOT  'CCC     '
   TO      BE OR NOT   TO     B
```

APOSTROPHE DENOTES TEXTUAL MOULD

L'S SAY JUSTIFY LEFT

R'S SAY JUSTIFY RIGHT

C'S SAY CENTRALIZE

```
50:\\    \\   OR NOT   \  \  !
   TO    BE   OR NOT   TO    B
```

! FOR SINGLE CHARACTER

SPACES BETWEEN \ AND \ ARE SIGNIFICANT & THE MOULD IS FILLED FROM LEFT TO RIGHT

```
50: <#     <#  OR NOT  >####     >
    TO      BE  OR NOT     TO     B
```

THIS ONE IS N.C.C. "STANDARD BASIC".

< SAYS JUSTIFY LEFT & TRUNCATE TO THE RIGHT

> SAYS JUSTIFY RIGHT & TRUNCATE TO THE LEFT

SINGLE CHARACTER MOULD — ALSO <

THERE MAY BE MORE WAYS YET. WHY CAN'T WE STANDARDIZE? EVEN BAD STANDARDS WOULD BE BETTER THAN NONE AT ALL.

Winners plan to assist physically handicapped

An application to employ the analytical power of the microcomputer to the rehabilitation of severely-handicapped people into the mainstream of society has won for J R and G Seagrave, of Ruislip, Middlesex the Apple II computer in the *Practical Computing* competition. The winning entry will be published next month.

Judges' report

By the closing date we were positively overwhelmed by the number of entries, most of them arriving in the last three days. Several others failed to arrive by the closing date. Competitions are obviously popular or perhaps it is the appeal of a class microcomputer like Apple II.

It is usual for the judges' report in competitions of this kind to include some sycophantic reference to 'the surprising quality of the entries' or something of that kind. In this case, we mean it sincerely. The winnowing-out which produced the final short-list was done initially by the editorial team, and they nominated more than 70 of the entries as being good enough to publish. Of those, 32 reached the judging table. We could not produce a shorter short-list—the quality was so good.

Wide scope

We kept the competition deliberately wide open. The rules asked for uses for an Apple II. We did this to give everyone a chance and, incidentally, that will be a feature of most of our competitions. We do not wish to restrict entries to particular applications; we do not want to exclude any age or interest groups; and we definitely do not want to pre-suppose a knowledge of computers or programming.

So for this competition we looked for entries which would make effective use of an Apple II, which was kindly donated to us by Mike Sterland of Personal Computers Ltd. Some of those short-listed were hand-written attempts from youthful amateurs; some were carefully-typed presentations from businessmen. In the final group we had games, project control, manufacturing, house design, business (forecasting as well as accounting), teaching (for the handicapped and for others), medical applications, and several home systems.

We also had a number of interesting 'radical technology' ideas. We considered a number of 'think' pieces, on the nature and role of microcomputers in society, especially, of course, in relation to employment.

We found a winner and we did so by applying some fairly simple rules. We wanted an entry which utilised the small computer appropriately, and some of you wrote about limited systems which made somewhat inadequate use of its programmability.

The winner, we feel, demonstrated an appreciation of the potential and the limitations; it would make full use of the system described.

Meanwhile, all other contestants are welcome to allow their competition entries to carry forward to the next competition—the *Computer for Christmas* announced in this issue. We will allow this to happen twice, so one competition entry will do for two consecutive competitions providing you send two appropriate entry forms—not necessarily at the same time, of course—and tell us you want your entry to carry forward. Since each competition will have different criteria for winners, and since each competition will have a different winner, everyone is guaranteed the chance to win.

In any case we shall be printing the best of the entries in future issues; many of them are well worth sharing with you. We will also look at the suggestion that we actually print a booklet of the games entered for competitions, probably with our own people turning them first into Basic code (guaranteed to work) if that is needed.

The winner

The entry from the Seagraves met all the criteria the judges used and as we progressed through successive short-lists and successive ratings of the remaining candidates, it became increasingly obvious that this application was favoured by all of us.

As Mr Seagrave notes, this application represents one of the greatest relative

(continued on next page)

JUDGING the competition, left to right, Richard Hease, Dennis Jarrett and Wim Hoeksma of *Practical Computing* with Mike Sterland of Personal Computers Ltd, donor of the Apple II computer.

(continued from previous page)

benefits of the computer. We felt it also made good use of the capabilities of the microcomputer system under offer.

The runners-up

Ten prizes of £25 are on their way to the best of the rest, and deciding on only 10 was not easy.

- *Ian Clark—project for the classification of stick and leaf insects.*

At first we thought this would be too esoteric an application for Apple but, reading on, it emerged as a classic of its kind.

- *Malcolm Cox — microprocessor-controlled lighting system.*

This entry would produce a pyrotechnic light show for discos. It looks feasible, though perhaps Apple is a trifle overpowered.

- *Barry Denyer—customer information, wholesale warehouse.*

Or indeed in several other similar applications areas with a fairly restricted number of customers. This system would handle stock control and customer records, and one interesting development using the Apple's special attributes would be to colour-code those records.

- *M R Hodder—an Apple today keeps the doctor away.*

The success of this entry proves that we did not disqualify automatically all awful puns. Apple would be used in a hospital pharmacy to file data on drugs, particularly on drugs which cannot or should not be used in combination.

- *T J Radford—Warlock Warren.*

This includes a neat introduction to the 'dungeons and dragons' school of fantasy games before Radford gets under way with his contribution.

- *J S Raynor—a program to demonstrate the distribution of daylight inside buildings.*

Architects have to make many decisions when designing buildings, which is one reason why many people dislike their designs. This program details one area of building design which in the author's view receives too little attention.

- *A G Roberts—teaching typing.*

Thought out very carefully, defining the problem and also defining the constraints under which a solution had to be implemented.

- *John Roddy — elementary reading assistance system.*

A "practical and entertaining use of a small processor/display system" is a word recognition exercise for children just reaching reading age.

- *C T Stevenson—cardiac monitoring system.*

Chris Stevenson is keen to develop a compact diagnostic tool for GPs, a system which would be implemented as firmware on a portable cardiac recording device to be carried "at all times" by people at risk but whose condition does not merit admission to hospital.

- *R M Wellings—a tool for company representatives.*

This is a good, solid use for a small computer. The author is a representative for a major international company and has to keep detailed records on 1,200 customers.

Honourable Mentions

M Adams	Biofeedback systems
R Adams	Gardener's database
A Arblaster	(Several)
I Baron	Ley-line investigation
M Barrett	A preparation tool for performance artists
C Baughan	Turning brain patterns into music
K Bean	Travellers' database
G Brennand	Data capture and processing for archeologists
M Brook	Quarry engineering maintenance
R Burdon	Mailing list and record-keeping
I Chapman	Anagram game
P Charlton	Voice-controlled microprocessor
J Crombie	Car race game
M Dixon	Perception game
T Dixon	Controlling mass data storage
H Dobbs	A system for managing proportional representation in the electoral process
N Dransfield	MLA—a system for de-centralising organisations
S Edwards	Research for a new type of generator
C Faulkner-King	Developments of the 'pathfinder' game
I Feaver	Equine database
J Finnie	Garment production and stock control system
R Ford	Predicting air traffic control conflicts
G Foster	Teaching aid
G Foy	Analysis of the alternative culture
D Francis	Carpet design
A Gray	Mapping the night sky
K Hall	Wobbly Jelly game
D Hand	Wordlab—a teaching system
G Hayes	Domestically-orientated computer
W Henry	Puzzle techniques
R Higginson	Computers in schools
C Homewood	King for a Year, a game which involves ruling an imaginary country
P Hough	Learning to drive
J Howden	Submarine Commander, a war game
L Howe	Towards tomorrow, a training requirement programme
M Hughes	Simulation techniques
P Jamieson	Library system
J Jaworski	Animated graphics
A Jones	Chess game recording system
I Jones	Articulation training
R Jones	The 'keyword' game
C Kah	Printed circuit board design
C Kelly	Text editing
G Kitt	Testing for colour blindness
M Knight	Genetic engineering game
P Landless	Diary
A Langley	Small business accounting
J McConnell	Astrophotography
J McIntosh	3D graphics
S McMillan	Motor vehicle diagnostic system for the layman
S Mann	Micros in the home
F Marden	Invasion game
S Marsh	Filing system for abstracts
B Martin	Health service program
N Martin	Simple robot
S Massey	Teaching aid
I Matthews	Calendar program
N Matty	Car fault diagnosis
A Mayne	Microviewdata
T Moore	Vehicle simulator for driver performance
A Mothew	Flight planning for light aircraft
F Murphy	War-gaming
W Nicholson	Currency Corner, a money management game
G Offord	Comic stocks
D Oldfield	Teaching aid in the home
M Parry	Flight simulator
D Paul	Project and data management
J Pearson	Making car number plates
W Ramsbottom	Fossil features database
K Randall	Analysis of musical style
G Relf	Timing and scoring in orienteering
P Reynolds	Forecaster
A Rigby	Computer Scrabble
D Robinson	Control of the home
P Rodwell	Taking computers to the people
P Ruffhead	Syntax checker for programming languages
D Ryan	Micro-controlled central heating
M Schorah	Cyphering
K Seville	Operating schedules for distributed processing networks
K Shore	Teaching programs
P Smallwood	Bullet game
A Smith	Teaching the handicapped
M Spencer	Explosion game
M Spicer	Micro in engineering design
R Stewart	Using an Apple
E Sunderland	Program for home teaching
J Sylvester	Computers in education
S Taylor	Letter writer
D Telfer	Analysis of space satellite pictures
E Thornton	Medical records
R Toogood	Digital compass
S Towsend	Medical tests
J Uff	House design
B Waddell	Charity finance office
P Wall	Business game
E Ware	Central heating specification
A White	School computer
P White	Battle game
R Wood	Process controller system
Anon	Archiving information system

Kim projects

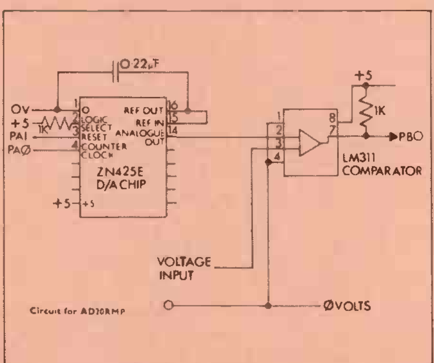
THIS MONTH we continue our article on adding analogue input/output to the Kim and other 6502-based machines. Last month we looked at the circuits used and outlined some applications. For convenience, last month's diagrams are repeated.

In the first example of A/D conversion, using A2DRMP, the internal binary counter of the ZN425E is driven directly from software in the Kim-1. Fig. 1 is the circuit layout and fig. 2 shows how the program and electronics interact to form the simplest of A/D converters.

By storing zero in DA both the ZN425E counter re-set pin and the clock pin are taken to logical zero. By clearing the counter in this way the analogue output of the D/A chip is guaranteed to be at zero volts. At the same time zero is stored in the variable SUM, which will be used as a tally of how many steps it took for the test voltage from the chip to rise above the unknown target voltage.

INC DA sets PAO, the clock line, to high. This has no direct effect; it is PAO falling to zero which advances the counter. To check to see if the test voltage is still less than the target voltage, the DB resistor is loaded into the A-resistor—PBO is connected to the comparator output. By rotating A to the right, this bit is transferred from the low order bit of the resistor into the carry bit. Assuming this is still clear DEC, DA clears PAO. The internal counter in the D/A chip increments by one and hence the analogue output rises by 8·52mV.

Figure 1



Keeping count

Incrementing SUM keeps count of the number of steps so far. In fig. 2 the test voltage rises above the target after eight clock pulses, showing that the unknown voltage was just above 59·6mV. When this occurs the comparator output goes high, which in turn sets the carry bit and the program counter jumps to DONE.

Two ways

It is possible that the unknown target voltage is greater than the highest voltage produced by the ZN425E. This being so, the internal counter, and SUM, would reach 255 and promptly re-start at zero and the program is potentially in a most undesirable infinite loop. The subroutine should try only once to find a value; unpredictable infinite loops are generally frowned upon in computing. BEQ OVFLOW detects when SUM falls to zero and control jumps to OVFLOW.

There are two ways in which this subroutine can return to its calling code. If the code branches to DONE, the data stored in SUM is valid and the A-resistor contains 0. If the counter has overflowed, and the branch was to OVFLOW, then the A-resistor will contain $FF (—1) and the contents of SUM should be ignored. SUM will contain zero, because that caused the subroutine to return.

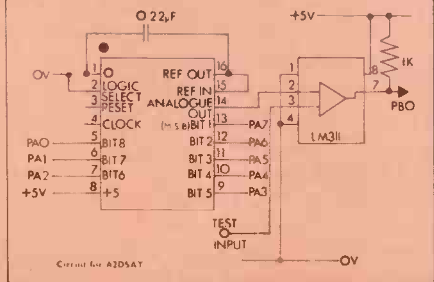
In fact, it is common programming practice to make a subroutine return the actual result of its computation in one resistor or location and to set the content of another resistor according to the exit conditions which caused the return, error, failure or success.

Direction resistor

Program DVM1 uses the A2DRMP routine to read in a voltage value and then display either the number in SUM or an overflow message on the Kim-1 display. First SETUP is called. This loads $FF into DDA, setting the data direction resistor PAO-PA7 all to outputs. It then loads $00 into DDB, setting the data

(continued on next page)

Figure 2



(continued from previous page)

direction resistors for PB0–PB7 setting all these to inputs.

If the A-resistor does not contain zero, CMP #0 is false. The program, DVM1, will display an out-of-range error message. It might display the word "ERROR". Unfortunately, the Kim-1 SCANS routine works only for the hexidecimal digits 0–9 and A–F and "EAAOA1" didn't look so good. In the end, "ABCDEF" was settled on—not very original but effective enough.

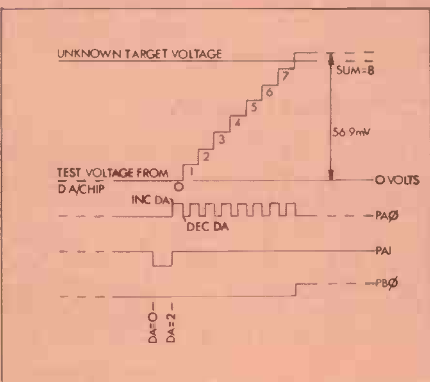
SCANS displays the 24 bits in DISP+2, DISP+1 and DISP as six groups of four-bit hex digits. It is these locations which are loaded with the message. This done, the code returns to the conversion and starts all over again (JMP DVM1A—the setup is not required again).

Clearing locations

If the result of A2DRMP was valid, we wish to display the contents of SUM, not as hex digits $00 to $FF, but in decimal 0–255. The contents of SUM are loaded into the A-resistor and a call to DECMAL is made.

On entering DECMAL, the A-resistor has to be used to clear the three DISP locations used by SCANS. So the contents of this resistor are pushed on to the stack (PHA) and the resistor is used before the original contents are restored from the stack (PLA). The binary-to-decimal conversion works by subtracting successively 100 from the A-resistor until it is less than 100. Each time one is added

Figure 3: A2 DRMP algorithm



to DISP+1, the fourth seven-segment display position on the Kim-1. From the remainder, which must be less than 100, 10 is subtracted repeatedly until the A-resistor contains less than 10. As before, each subtraction is accompanied by an increment of DISP.

Redundant

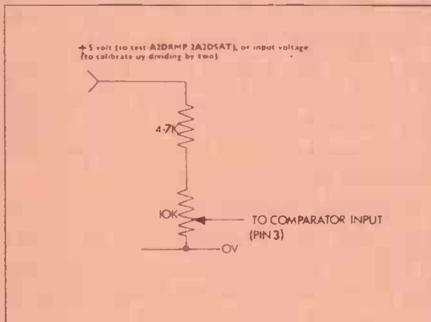
You may notice a totally redundant instruction in this code. For the twos compliment subtraction (SBD #10) to function correctly the carry bit must be set (SEC), indicating no borrow. This is done dutifully, according to the programming manual.

Because the previous instruction was branch, however, if carry clear (BCC) the carry bit must be set at this point. Carry on writing code like that and the bogey man will get me.

Continuous display

The 10s are now in the bottom four bits of DISP, corresponding to the sixth display. By shifting DISP to the left four times—multiplying by 16—they are then in the top four-bit locations, the fifth display. The units remainder is still in the A-resistor and this is finally added into the

Figure 4: Test circuit.



bottom four bits of DISP. As before, SCANS is called and the program jumps back to the conversion routine again, providing a continuous display. Fig, 4 shows a potential divider set-up to test DVM1.

There are clearly disadvantages with this A/D conversion technique. The conversion is by no means the fastest possible. Worse than that, it does not take a fixed time. If the target voltage is low, it is fast. If the target voltage is high, or above the range altogether, it becomes progressively slower, to such an extent that the display dims noticeably with higher voltages, solely due to the increased conversion time.

Fewer wires

On the advantage side there are only three parallel port locations used, with correspondingly fewer wires to be misplaced. The second conversion algorithm, variously called the Successive Approximation Technique or Binary Chop, uses nine ports. On the other hand, it is much faster and always takes eight attempts.

The successive approximation technique is the optimum strategy for finding an unknown number between fixed limits, with the only clue given being "too high" or "too low". The next attempt is always halfway between the limits. The limits are adjusted after each response. The higher limit is dropped to the last guess if the

(continued on next page)

(*continued from previous page*)

response was "too high", the lower limit raised if "too low".

The unknown voltage is assumed to be in the range 0 to 2·18v, the D/A chip output corresponding to the numbers 0 to 255. The output of the comparator indicates whether the converter voltage is greater ('1') or less ('0') than the unknown value.

Electronic layout

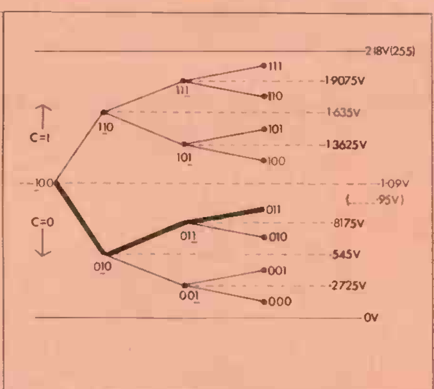
Fig. 3 shows the electronic layout in this case. The internal counter of the ZN425E has been disabled (by connecting the LOGIC SELECT (pin 2) to zero volts). The chip now converts the eight-bit number from the parallel port (PA0–PA7) directly into a voltage. Fig. 5 shows a tree diagram of the program over the first three of the eight stages of the conversion. A2DSAT is the Kim-1 code implementation of the algorithm.

Work through A2DSAT, referring to fig. 5. Assume a target voltage of 0·95v. First load the A-resistor with 1,000,000 ($80) and store it in MASK and the output port DA. The ZN425E assumes an output voltage of 1·09v, $80 is 128, halfway between 0 and 255. The NOP instruction allows a time for the D/A chip to settle to its correct value. Next the state of the comparator is loaded into the A-resistor and PBO is rotated right into the carry bit.

Control transfer

If 1·09v is greater than the target voltage, the carry bit will be set and control will be transferred to the label ONE. If C is clear, as it will be in our example, the code must clear the testing bit. To this end

Figure 5: First three stages of the successive approximation algorithms (underline (010) indicates position of mask bit)

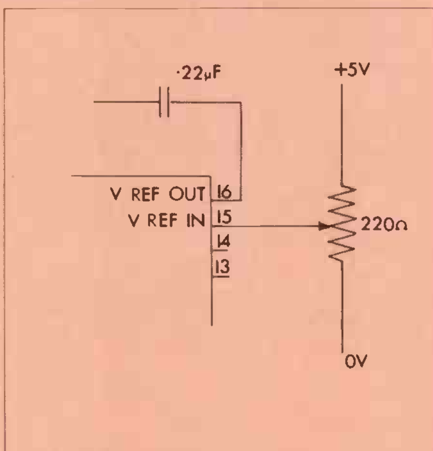


MASK (10,000,000) is loaded into the A-resistor and Exclusively ORed (EDR) with 11111111. If the corresponding bit in MASK is the same as the A-resistor (0 and 0 or 1 and 1) then that bit in the A-resistor is cleared. If they are different (0 and 1 or 1 and 0) then that bit is set, giving 01111111. By ANDing (AND) this with the current contents of DA the mask bit, and only the mask bit, is cleared. Control carries on from the label ONE.

Others unchanged

MASK is then shifted to the left, giving 01000000. This is then ORed into DA setting PA6 but leaving all the other ports unchanged. If our target voltage was greater than 1·09v we have 11000000 stored in DA, and if less (0·95v) we have

Figure 6: Altering the reference voltage on 2N425E



01000000. In the former case the unknown voltage is now known to be between 1·09 and 2·18v. Then the new comparison will be between above 1·635v and below it—in the latter case, above and below 0·545v.

The comparator output is read again and the current test bit cleared or left. In the example the test voltage is less than the target, so it remains at 01000000. The next bit is set (01100000) and the test determines whether the unknown voltage is above or below 0·8175v; 0·95v is above, so 01100000 remains. After only three iterations around the loop, the target of 0·95 is known to be between 0·8175 and 1·09v. After a further five iterations, the target voltage will have been measured to with-in 8·516mV.

Embedded

MASK only ever contains one bit set. This starts in the high-order bit position and is shifted successively left towards the low-order bit for each branch decision in the tree. After all eight bits have been tested, the testing bit will appear in the carry bit. Instead of branching to NEXT the final value in DA will be loaced into the A-resistor and the subroutine will return.

A2DSAT is embedded in DVM2. Like DVM1, it uses SETUP to initialise the data direction ports. Then it jumps to the conversion routine, transfers the answer

(*continued on next page*)

SHOPWINDOW

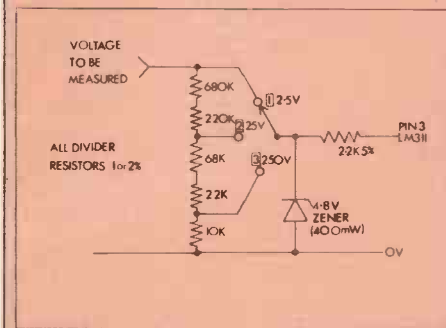
(continued from previous page)
in the A-resistor directly to DECMAL, and then displays the value using SCANS. This lot is repeated *ad infinitum* by jumping directly to the conversion routine (DVM2A). The re-set button is provided to terminate this loop; it also tidies up the stack.

Need for caution

Remember that any negative voltage displays as zero, and any over-range voltage as 255. So treat those values with caution.

While DVM1 and DVM2 are supposed to stand for Digital VoltMeter, they have a very unfortunate anomoly. When 255 is displayed it corresponds only to 2·18v. There are at least four solutions to this disparity. One, it could be ignored. Second, it could be multiplied by 1·1697248, either with a pocket calculator or by some very tiresome software. Third, a resistive potential divider could be arranged so that a voltage of 2·55 gives a reading of only 128 instead of the over-range indication (see fig. 4).

Figure 7: Input attenuation for three-range digital voltmeter using KIM-1



If the result of the conversion is shifted one place to the right—multiplied by two—before jumping to DECMAL, the voltmeter skill displays 0–255, except that these now correspond to an input between 0 and 2·55v. These are now in steps of two.

Best method

Fourth and best, the ZN425E generates an internal voltage reference for the conversion of about 2·5v; this is fed normally directly into the reference input pin (15). If those two pins are disconnected and a slightly higher voltage derived from a potential divider, fig. 6, across the 0 and +5 supply rails, is fed into pin 15 a full calibration may be achieved in this way.

A voltmeter will usually be calibrated by comparison with a device known to be accurate. Feed one volt into the comparator and adjust the 220 ohm potentiometer until the display reads 100.

Fig. 7 shows an input potential divider for measuring higher voltages. With the switch in the 0–2·55v postion, the input voltage goes directly into the comparator. With the switch in the 0–25·5v range position, only one-tenth of the voltage appears across the 1000K ohm resistor made up from 68+22+10K resistors and nine-tenths is 'lost' across the 900K resistor (680+220K).

In the 0–255 volt range, 99/100ths of the input voltage is across the 990K resistor (680+220+68+22K) and only one-hundredth across the 10K resistor and the comparator input. I would be very dubious about putting 250v anywhere near a C-MOS microprocessor. The 2·2K resistor and 4·7 volt zener diode should provide adequate protection from all but the most ardent high-voltage wire danglers.

Transformation

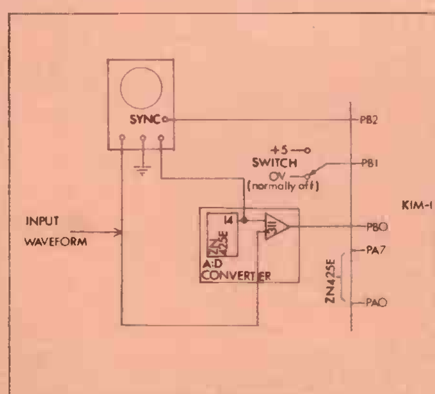
SCOPE is a further demonstration of the capabilities of the A/D conversion routine. It transforms an ordinary oscilloscope into a digital storage oscilloscope. Fig. 8 shows a dual-beam oscilloscope attached to the A/D converter and the Kim-1. In record mode, the signal is displayed on both channels of the oscilloscope, on one directly. On the other, it appears as part of the A/D conversion process, monitoring the ZN425E analogue output.

This second signal shows the successive approximation technique, trial voltages 'homing'-in on the waveform during each sample. Each time the signal waveform is digitised by A2DSAT, the result is stored away in a buffer, 512 bytes long. Whenever the end of the buffer is reached the pointer is set to the beginning again. It is, in effect, circular and always contains the last 512 samples. The current value overwrites the sample read in 513 bytes before.

Logic input

Further, a logic input to PB1 is monitored with each sample. While it remains low, the recording continues. As soon as it goes high, the recording stops and the

Figure 8: Layout for digital storage scope



(continued on next page)

(*continued from previous page*)

last 512 bytes are frozen. Just before returning, the recording routine saves the current value of the buffer pointer in SYNCL and SYNCH, showing where the stored sample begins and ends. Fig. 8 shows a microswitch wired into PB1; any logic signal would do.

Because the buffer is more than 256 bytes long, the convenient absolute indexed mode cannot be used. Instead, the (indirect), Y mode is employed. For RECORD, the address of the beginning of the buffer is loaded into POINTR and POINTR+1, the Y resistor is zeroed. The sample is digitised and stored in the address contained in POINTR and POINTR+1. The state of PB1 is monitored; if it is set control jumps to ENDREC, where the sync value is saved and control returns.

Next sample

The low-order pointer byte is incremented and if it is less than 256 the next sample is obtained. If it was not—i.e., it was 255 and has become 0—the high-order POINTR+1 byte is incremented and loaded into the A-resistor. If A is equal to #PHIGH+2, then both 256 byte halves of the buffer are full, and the pointer is set to its initial value at RESET. If not, A is stored back in POINTR+1 and samples will continue to be stored in the second 256 buffer page.

PLAY is the playback routine. It starts with FB2 being configured as an output; the signal to syncronise the oscilloscope will appear here. POINTR is set to the start of the buffer and Y is cleared; the sample value is taken from the buffer and stored in DA. The value appears on pin 14 of the D/A chip and is displayed on the second channel of the oscilloscope. The comparator output is ignored.

If POINTR and POINTR+1 are equal to SYNCL and SYNCH respectively, PB2 is briefly pulsed high, thereby synchronising the timebase scan to the buffer. After this POINTR and POINTR+1 are manipulated to keep the pointer cycling round the buffer as in RECORD. This code continues to display the contents of the buffer until the re-set button is used.

Numeric display

You don't need an oscilloscope to use this code. If, instead of putting the buffer value out to the D/A converter the code jumped to DECMAL and SCANS, it would display the sample on the seven-segment displays. As it stands, it would race through the values too quickly to be any use.

If the switch connected to PB1 was used to step through the buffer, however, it would make a useful numeric display. The null subroutine YOURS returns immediately. Code could be written here to use the display. Remember, if you use the switch, to test for both on and off before you return from YOURS, otherwise the program will cycle through values as if it wasn't there, each time the switch is pressed.

Not instantaneous

Even with A2DSAT, the conversion is by no means instantaneous, requiring some 350 microseconds. Further, while the conversion time is more consistent than A2DRMP, it still varies according to the input voltage. For each trial in which the testing bit must be unset before continuing at ONE, 13 extra machine cycles occur which would not have done so if the testing bit was left set. They could be padded out with NOPs, but at the expense of further increasing the conversion time. In any case the effect is not apparent in most applications.

It is worthwhile to optimise the code in A2DSAT as much as possible. Each instruction consumes valuable microseconds; any instruction omitted is a contribution to efficiency. This is especially true if the instruction is removed from the inside loop, between NEXT and BCC NEXT. The LDA DA instruction might be omitted if DB was rotated directly, thereby saving four machine cycles eight times.

The A/D converter is a real boon to the computer games writer. Without doubt, it is the ability to enter your move or action into the computer quickly and easily and to have the machine respond instantly, which adds greatly to the excitement of any game.

Joystick controls make superior input devices. Recently they have become easy to obtain cheaply, mainly due to the boom in television game chips and the need to input to them.

Slick control

At first sight it would seem impossible to read more than one voltage into the micro with a single A/D converter, and joystick controls for two players would generate at least four voltages. Fortunately, C-MOS integrated circuits are available to multiplex two, four or even 16 channels into one converter, under software control.

Using a joystick could make for rather slick cursor control in a memory-mapped VDU—just the job for a fast text editor or the like.

The photographer and Kim-1 owner could design an intelligent light meter, with a digitised photocell value. The program could calculate the optimal shutter and aperture settings, given the desired depth of focus and film characteristics, the speed of the subject, and so on. Unfor-

(*continued on next page*)

(continued from previous page)

tunately, by the time the programmer understands the problem well enough to code it, he can usually take one look at the scene—f8 at 100th; click; before you even have time to press the GO button. The Kim would fare better during the more sedate and deliberate darkroom phase, as an enlarger timer/exposure meter and process controller.

For the garden

The gardener could attach a thermistor temperature probe to the converter and then run RECORD with a 169-second (169×512—about 24 hours) delay between data samples. On playback each day, the gardener sets the most expensive MAX-MIN thermometer this side of Surbiton.

As a final thought while on the subject of A/D converters and SCOPE, it is worth trying to record a sample of speech with this program. You would need some kind of amplifier and level shifter to bring the microphone signal up to 2·5v. You might also be well advised to put some kind of attentuation on the analogue output before feeding that signal into your stereo amplifier instead of the oscilloscope for playback, thereby preventing the loudspeaker coils jumping out of the cabinet at you. The speech sample is very short with the memory available in the basic Kim-1—only about 0·2 seconds.

With a 4K memory expansion block the recording will last about one and a half seconds. "One, two, three, fo . . .". The replayed speech is even then only just recognisable. It is improved by using as much treble cut as possible; a synthesiser low-pass filter is ideal to reduce the distortion caused by the digitisation steps. The main problem is the low sampling rate. It should be at least 6,000 times a second, a complete conversion and storage cycle of 166 microseconds.

There are various ways in which the conversion code could be speeded. The conversion would no longer be a separate subroutine; the call and return take 12 cycles. The playback switch would be connected to the interrupt line (10 cycles). The digitisation could be reduced to seven, or even six, bits—by loading MASK with $40 or $20 instead of $80.

Speeding code

As an alternative it is possible to construct the successive approximation algorithm from electronic logic rather than software. A 15-microsecond A/D converter is possible with the ZN425E, which is about as fast as a microprocessor could store the results anyway, without some kind of direct memory access (DMA).

With this equipment experiments into digital sound processing—real-time digital filters—speaking computers (a talking calculator for instance) and speech recognition by software all become a reality.

```
                ;DEFINE SYSTEM LOCATIONS AND
                ROUTINES
                DA     =$1700
                DDA    =$1701
                DB     =$1702
                DDB    =$1703
                SCANS  =$1F1F
                DISP   =$F9
                PLOW   =$00
                PHIGH  =$02
0000            TEST   *=*+1
0001            MASK   *=*+1
0002            SUM    *=*+1
0003            POINTR *=*+2
0005            SYNCL  *=*+1
0006            SYNCH  *=*+1
                ;
                ;TEST DECMAL - DISPLAYS
                ;NUMBER IN LOCATION 0000
                ;USING SCANS.
                ;
0007  A5 00     TESTD  LDA TEST
0009  20 5C 00         JSR  DECMAL
000C  20 1F 1F  PDLY   JSR  SCANS
000F  4C 0C 00         JMP  PDLY
                ;
                ;DIGITAL VOLTMETER USING
                ;RAMP A:D
                ;
0012  20 51 00  DVM1   JSR SETUP
                       CONFIGURE PORTS
0015  20 8C 00  DVM1A  JSR A2DRMP
                       GET VOLTAGE
0018  C9 00            CMP #0
                       IS IT VALID?
001A  F0 12            BEQ VALID
                       YES
001C  A9 AB            LDA #$AB
001E  85 FB            STA DISP+2
0020  A9 CD            LDA #$CD
0022  85 FA            STA DISP+1
0024  A9 EF            LDA #$EF
0026  85 F9            STA DISP
0028  20 1F 1F         JSR SCANS
                       DISPLAY "ABCDEF"
002B  4C 15 00         JMP DVM1A
002E  A5 02     VALID  LDA SUM
0030  20 5C 00         JSR DECMAL
0033  20 1F 1F         JSR SCANS
                       DISPLAY DATA
0036  4C 15 00         JMP DVM1A
                ;
                ;DIGITAL VOLTMETER USING
                ;SUCCESSIVE APPROXIMATION
                TECHNIQUE.
                ;
0039  20 51 00  DVM2   JSR SETUP
003C  20 B1 00  DVM2A  JSR A2DSAT
003F  20 5C 00         JSR DECMAL
0042  20 1F 1F         JSR SCANS
0045  4C 3C 00         JMP DVM2A
                ;
                ;DIGITAL STORAGE SCOPE
                ;
0048  20 00 01  SCOPE   JSR RECORD
004B  20 33 01  SCOPEA  JST PLAY
004E  4C 4B 00          JMP SCOPEA
                ;
                ;SETUP DATA DIRECTION
                ;REGISTERS
                ;
0051  A9 FF     SETUP  LDA #$FF    OUTPUT
0053  8D 01 17         STA DDA
0056  A9 00            LDA #0      INPUT
0058  8D 03 17         STA DDB
005B  60               RTS
                ;
                ;CONVERTS NUMBER IN A REG
                ;TO DECIAMAL IN DISP, READY
                ;FOR A CALL TO SCANS.
                ;
005C  48        DECMAL PHA
005D  48 00            LDA =0
                       CLEAR DISP
005F  85 F9            STA DISP
0061  85 FA            STA DISP+1
0063  85 FB            STA DISP+2
0065  68               PLA
0066  C9 64     N100   CMP #100
                       BRANCH IF<=99
0068  90 08            BCC N10
006A  38        SEC
                       NO BORROW
006B  E9 64            SBC #100
                       A LESS 100
006D  E6 FA            INC DISP+1
                       HUNDREDS + 1
006F  4C 66 00         JMP N100
                       AND AGAIN
0072  C9 0A     N10    CMP #10
                       BRANCH IF <=9
0074  90 08            BCC N1
0076  38               SEC
0077  E9 0A            SBC #10
```

(continued on next page)

(continued from previous page)

```
0079  E6 F9           IMC DISP
                      TENS + 1
007B  4C 72 00        JMP N10
007E  06 F9     N1    ASL DISP
                      SHIFT TENS TO
0080  06 F9           ASL DISP
                      UPPER DIGIT
0082  06 F9           ASL DISP
                      POSITION IN
0084  06 F9           ASL DISP
                      DISP.
0086  18              CLC
0087  65 F9           ADC DISP
                      ADD UNITS
0089  85 F9           STA DISP
008B  60              RTS
                ;
                ;ANALOGUE TO DIGITAL
                ;CONVERSION - 1 CLOCKS
                ;ZN425E D TO A CHIP UP TO
                ;256 TIMES. RAMP ALGORITHM
                ;
                ;IF VOLTAGE IN RANGE:
                ;  RETURNS IN SUM, SETS A REG
                ;  TO 0
                ;IF VOLTAGE OUT OF RANGE:
                ;  SETS SUM TO 0 AND A REG
                ;  TO $FF
                ;
008C  A9 00     A2DRMP LDA #0
008E  8D 00 17        STA DA
                      RESET ZN245
                      COUNTER
0091  85 02           STA SUM
                      CLEAR SUM
0093  A9 02           LDA #2
0095  8D 00 17        STA DA
                      COUNTER READY
0098  EE 00 17  COUNT INC DA
009B  AD 02 17        LDA DB
                      COMPARATOR STATE
009E  6A              ROR A
                      INTO C BIT
009F  B0 0A           BCS
                      DONE IS HIGH
00A1  CE 00 17        DEC DA
                      CLOCK ZN425
00A4  E6 02           INC SUM
                      KEEP COUNT
00A6  F0 06           BEQ
                      OVFLOW VOLT
                      OVERRANGE
00A8  4C 98 00        JMP COUNT
00AB  A9 00     DONE  LDA #0 IF A=0
                      SUM IS VALID
00AD  60              RTS
00AE  A9 FF     OVFLOW LDA #$FF
                      IF A=-1 SUM IS
                      INVALID
00B0  60              RTS
                ;
                ;ANALOGUE TO DIGITAL
                ;CONVERSION - 2 SUCCESSIVE
                ;APPROXIMATION TECHNIQUE
                ;RETURNS IN A REG
                ;
00B1  A9 80     A2DSAT LDA #$80
                      TOP BIT
00B3  85 01           STA MASK
00B5  8D 00 17        STA DA
00B8  EA              NOP
00B9  AD 02 17  NEXT  LDA DB
                      GET RESULT.
00BC  6A              R OR A
                      INTO CARRY
00BD  90 0A           BCC ONE
                      LEAVE SET
00BF  A5 01           LDA MASK
                      ELSE CLEAR
00C1  49 FF           EOR #$FF
00C3  2D 00 17        AND DA
                      COMPLEMENT DA
00C6  8D 00 17        SRA DA
00C9  46 01     ONE   LSR MASK
00CB  A5 01           LDA MASK
00CD  0D 00 17        ORA DA
                      SET NEXT BIT IN DA
00D0  8D 00 17        STA DA
00D3  90 E4           BCC NEXT
                      MASK INTO CARRY?
00D5  AD 00 17        LDA DA
                      YES!
00D8  60              RTS
                ;
00D9                  *=$100
                ;
                ;CONTINUALLY SAMPLES AN
                ;INPUT WAVEFORM AND STORES
                ;IN A CIRCULAR BUFFER OF 512
                ;BYTES' IF PB1 GOES HIGH
                ;RECORDING STOPS, BUFFER
                ;POINTER IS STORED IN SYNCL &
                ;SYNCH.
                ;
0100  20 51 00  RECORD JSR SETUP
0103  A9 00     RESET LDA #PLOW
0105  85 03           STA POINTR
                      POINTER TO
0107  A9 02           LDA #PHIGH
                      BEGINNING OF
0109  85 04           STA POINTR+1
                      CIRCULAR
010B  A0 00           LDY #0
                      BUFFER
010D  20 B1 00  AQUIRE JSR A2DSAT
                      CONVERSION
0110  91 03           STA (POINTR),Y
                      SAVE
0112  AD 02 17        LDA DB
0115  6A              ROR A
0116  6A              ROR A
0117  B0 11           BCS ENDREC
0119  E6 03           INC POINTR
011B  D0 F0           BNE ACQUIRE
                      END OF PAGE
011D  E6 04           INC POINTR+1
011F  A5 04           LDA POINTR+1
0121  C9 04           CMP #PHIGH+2
0123  F0 DE           BEQ RESET
0125  85 04           STA POINTR+1
0127  4C 0D 01        JMP AQUIRE
012A  A4 03     ENDREC LDA POINTR
                      SAVE SYNC
012C  85 05           STA SYNCL
012E  A5 04           LDA POINTR+1
                      POINTR
0130  85 16           STA SYNCH
0132  60              RTS
                ;
                ;REPLAYS CIRCULAR BUFFER
                ;FILLED BY RECORD. PB1
                ;SYNCHRONISES SCOPE TO START
                ;OF BUFFER
                ;
0133  A9 04     PLAY  LDA #4
                      PB2 AS SYNC
0135  8D 03 17        STA DDB
                      OUTPUT
0138  A9 00     RESTRT LDA #PLOW
013A  85 03           STA POINTR
013C  A9 02           LDA #PHIGH
013E  85 04           STA POINTR+1
0140  A0 00           LDY #0
0142  B1 03     CONT  LDA (POINTR),Y
0144  20 72 01        JSR YOURS
0147  8D 00 17        STA DA
                      OUTPUT SAMPLE
014A  A5 05           LDA SYNCL
014C  C5 03           CMP POINTR
014E  D0 10           BNE NOSYNC
0150  A5 06           LDA SYNCH
0152  C5 04           CMP POINTR+1
0154  D0 0A           BNE NOSYNC
0156  A9 04           LDA #4
                      SYNC PULSE TO
0158  8D 02 17        STA DB
011D  E6 04           INC POINTR+1
011F  A5 04           LDA POINTR+1
0121  C9 04           CMP #PHIGH+2
0123  F0 DE           BEQ RESET
0125  85 04           STA POINTR+1
0127  4C 0D 01        JMP AQUIRE
012A  A5 03     ENDREC LDA POINTR
                      SAVE SYNC
012C  85 05           STA SYNCL
012E  A5 04           LDA POINTR+1
                      POINTR
0130  85 06           STA SYNCH
0132  60              RTS
                ;
                ;REPLAYS CIRCULAR BUFFER
                ;FILLED BY RECORD. BP1
                ;SYNCHRONISES SCOPE TO START
                ;OF BUFFER
                ;
0133  A9 04     PLAY  LDA #4
                      PB2 AS SYNC
0135  8D 03 17        STA DDB
                      OUTPUT
0138  A9 00     RESTRT LDA #PLOW
013A  85 03           STA POINTR
013C  A9 02           LDA #PHIGH
013E  85 04           STA POINTR+1
0140  A0 00           LDY #0
0142  B1 03     CONT  LDA (POINTR),Y
0144  20 72 01        JSR YOURS
0147  8D 00 17        STA DA
                      OUTPUT SAMPLE
014A  A5 05           LDA SYNCL
014C  C5 03           CMP POINTR
014E  D0 10           BNE NOSYNC
0150  A5 06           LDA SYNCH
0152  C5 04           CMP POINTR+1
0154  D0 0A           BNE NOSYNC
0156  A9 04           LDA #4
                      SYNC PULSE TO
0158  8D 02 17        STA DB
0158  A9 00           LDA #0
015D  8D 02 17        STA DB
                      SCOPE
0160  E6 03     NOSYNC INC POINTR
                      BUFFER
0162  D0 DE           BNE CONT
                      POINTR
0164  E6 04           INC POINTR+1
                      RESET
0166  A5 04           LDA POINTR+1
0168  C9 04           CMP #PHIGH+2
016A  F0 CC           BEQ RESTRT
016C  85 04           STA POINTR+1
016E  4C 42 01        JMP CONT
0171  60              RTS
                ;
                ;
                ;USER DEFINED ROUTINE
                ;
0172  60        YOURS RTS
0173
```

Basic aid to small business

This article is the first of two parts describing a program in Basic to aid a small business in the preparation of quotations, invoices and order forms.

by Nick Hampshire

ANY businessman considering purchase of a computer, however big or small the machine, must be certain the machine will not only pay for itself but actually contribute towards the profitability of his company. For many, this is a difficult decision.

In most businesses the main financial return from buying a computer will result from a reduction of costs through greater efficiency and improved productivity. Obviously, the number and scope of the applications depend a great deal on the size and power of the machine and also, though perhaps to a lesser extent, the nature of the business.

The average small business system would be a disc-based computer with a medium-speed printer, programmable in Basic and costing about £3,500. It would not be unreasonable to expect such a system to pay for itself in two years. This would require the computer to contribute more than £30 per week in increased productivity and efficiency.

To justify the purchase of a computer by a business, we must look for the applications where the greatest improvements in efficiency and productivity can be achieved. Preferably this should be done with the minimum programming effort and changes in organisation of the business.

Intricate

There are many well-known applications for computers in business, stock control, general ledger and payroll. An application such as payroll is very intricate and requires that the person writing the program has a very good understanding of taxation. Such applications can, if not properly written, be a potential source of serious financial losses in the business.

A stock control system may result in the holding of excessive or inadequate stock, due to a badly-written and designed reordering algorithm. These are factors which often discourage a businessman from embarking on a course of computerisation.

It would, however, be unfortunate if these factors dissuaded him, since they are far from the only applications for a computer in business.

We are familiar with these applications simply because they have been available for many years on time-sharing bureaux and large mainframes, but we are wrong to think of them as the only applications; the average business abounds with candidates for computerisation, applications which will result in increased efficiency and reduced effort both for the businessman and his employees.

Examples of such applications are the production of quotations, orders and invoices, all three of which consume a large amount of time and are prone to a considerable degree of error.

Three ideal jobs

It is not unusual for a businessman to have to produce a dozen quotations a week, each accurate and fully-itemised. Each involves a long and complex process, which usually has to be done by the businessman himself and, depending on the length of the quotation, each can take him well over an hour to produce.

Production of orders and invoices involves a similar process; a large number of calculations are involved where, if a mistake is made, considerable inconvenience and perhaps financial loss may result.

Again, it is usually advisable that the businessman supervises and checks the writing of orders, time which he can often ill afford to spare.

We thus have these three jobs as ideal candidates for computerisation, since they are both labour-intensive and prone to potential costly error. I have written a fairly simple set of programs, included in this article, to perform these functions. It will, I hope prove a useful guide to those considering implementing a similar system.

Quotations

The writing of an order, invoice or a quotation can be divided into two processes. The first is the selection of the items, such as charges, to be included; in the case of an invoice, this will be derived from the customer's order and your own delivery notes. The second process is the writing of the document, with a short

(continued on next page)

(continued from previous page)
description of each item, the quantity, its unit price, VAT, and total price. It is this second process with its requirements of neat formatted typing and a considerable number of calculations which we shall computerise.

With such a computer program all that is required is that we enter the code number of an item obtained from a stock list of prices and charges and the number of units of that item, a process repeated for all the items to be included.

Given that information, the computer extracts from memory all the relvant data, performs the correct calculations, and prints-out our order, invoice or quotation. The product is neat, accurate, and, above all, produced very quickly. The computer thus is paying for itself by increasing productivity, with the added bonus of improved efficiency.

Software

The software is of very general design and can be used by any business; what makes it specific to a particular business are the items placed on the data file. The data file is, of course, our stock list of component prices, and labour and ancillary charges. The system could be expanded by modifying the programmes to use other files, e.g., customer and supplier, each file containing a list of names and addresses of our customers and suppliers, an addition which would save us entering this data each time.

Having gone so far we could expand the system still further to store records of all transactions which would enable the database to be used for a general ledger system. Similarly, component quantities could be stored in the stock file, which would enable the data to be used in a stock control program. We shall, however in this article confine ourselves to the use of a single simple stock file.

Hardware

All the programs in this article were written on a Cromemco Z2-D system but could be modified without a great deal of effort to run on any other version of extended disc Basic. The hardware used was the Z2-D with two 5in. disc drives and 32K of memory, in which was running the standard Cromemco disc Basic. This system was interfaced to a Teletype 43, which performed the dual function of console and printer.

The software could be altered easily to accept a standard VDU console with separate printer; in the case of a Cromemco system this would require the addition of a serial I/O board (TU-ART) with the printer connected to port 50.

In Cromemco Basic we can output to a printer at this port by making the following insertions and modifications, in this case to print A$.

```
OUT $50$, 08
OPEN/2/ "$T5"
PRINT/2/A$
CLOSE/2/
```

In a system running Altair or Microsoft Basic, the LPRINT statement can be used. The Teletype 43 was chosen as a console/printer, firstly because it offered the lowest-cost peripheral option for the Cromemco system. Secondly, the 43 is a medium-speed printer with a high-quality dot matrix type-face; this printer is available also in a friction-feed version which would allow one to use existing stationery.

File structure

The most important part of our proposed system is the data or stick file; this is the common element around which all our programs will be constructed. Since it will be necessary to access quickly a set of data which may not necessarily be in the same order as it appears on the file, we will use random access files.

A random access file is divided into an indefinite number of records, each containing a complete set of data. In our application, a record will contain a description of an item of stock, its retail price, VAT rate, trade price supplier code and any other required information on that item. The records are numbered 1, 2, 3 and so on; each number represents a different item of stock—the highest record number is thus equal to the number of stock lines.

A constraint on the maximum number of records is imposed by the capacity of the discs in our system. Thus, if each record is 128 bytes long, we can store a maximum of 562 records on a 5in. disc and 1,953 on an 8in. disc, assuming both are single-density and single-sided. If this proves to be a constraint we can add more disc drives or insert and remove discs manually, though this is a solution which is not recommended.

Identical

If our programs are to be able to read and write meaningful data to and from the data file, each record must have an identical construction. This means that we must decide exactly what data is required and which bytes within the record they occupy. Each record in the stock file in 128 bytes long, far more than is actually required. It does allow, however, for the inclusion of further data at a later date. The data in each record is arranged as follows:

```
Byte  0-30 Description
     31-40 Retail Price
     41-50 VAT Rate
     51-60 Trade Price
     61-70 Supplier Code.
```

As you can see, the 128 bytes is not all used; neither is our data as compact as it could be. The user is free, however, to alter this to suit his own needs. The data file is given a name and is called "STOCK" in the programs. One other data file is required in this system, known as "POINTER". The primary function of this file is to contain the highest record number so far entered.

This is required when entering a new

(continued on next page)

(continued from previous page)

record or listing existing records, since it allows the computer to recognise the end of the current "STOCK" file. Other records within "POINTER" show the current order, invoice and quotation numbers. Each record within this file contains a single item of numerical data and each record is 10 bytes long. Note that programs are stored on disc A and data on disc-drivers B, C and D.

Programs

To perform the functions of printing and calculating invoices, orders and quotations, a set of seven programs are used. Of these, three perform the actual printing and calculations, three perform various database update and examination functions, and the last program ties together all the previous six.

This program is known as a menu-picking program, since it allows the person using the computer to select a particular function and run the program to perform it. Thus, after finishing a function, the menu-picking program is always loaded automatically and the user asked which function he wishes to do next.

The menu-picking program thus could be regarded as a form of automatic index. The reason we need this program lies in the fact that all our programs are stored on one disc drive and only one of them is loaded at any one time into the computer memory.

The menu program thus saves us from having to load manually programs from disc into the computer. The following are listings of the first four programs—the remaining three will be published next month—together with a brief explanation of each. We start with PICK our menu-picking program. Note that in the program listing the PRINT statement is abbreviated to @.

```
20 @"QUOTATION, INVOICE AND ORDER
PROGRAM. FUNCTION SELECT."
30 @ : @ : @
40 @"1 PRINT QUOTATION."
45 @"2 PRINT ORDER."
50 @"3 PRINT INVOICE."
60 @"4 PRINT TOTAL STOCK LIST."
70 @"5 INPUT DATA ON NEW STOCK ITEM."
80 @"6 EXAMINE AND UPDATE STOCK DATA."
100 @ : @ : @
110 INPUT "OPERATION?        ",A
120 IF A>6 THEN 100
130 ON A GOTO 140,150,160,170,180,190
140 RUN "QUOTATION"
150 RUN "ORDER"
160 RUN "INVOICE"
170 RUN "LIST"
180 RUN "NEW"
190 RUN "UPDATE"
200 END
>>
```

NEW is probably the most important of the seven programs, since it is used to create the database used by the other programs.

```
10 DIM A$)128)
20 DIM B$(128)
110 OPEN/1,50/"B:POINTER"
120 GET/1,2,1/A
130 PUT/1,2,1/A+1
140 CLOSE/1/
150 INPUT "DESCRIPTION        ",A$
160 L=LEN(A$)
170 IF L<=29 THEN 190
180 @"        **** DESCRIPTION TOO LONG ****"
185 GOTO 150
190 INPUT "PRICE RETAIL (POUNDS)        ",B
195 INPUT "TRADE PRICE        ",C
200 INPUT "VAT RATE %        ",V
201 INPUT "SUPPLIER CODE        ",S
204 INPUT "IS THIS CORRECT? Y OR N ",G$
206 IF G$="N"THEN 150
210 LET B$=STR$(B)
215 LET S$=STR$(S)
220 LET V$=STR$ (V)
225 LET C$=STR$(C)
230 A$(31,40)=B$
235 A$(51,60)=C$
240 A$(41,50)=V$
245 A$(61,70)=S$
310 OPEN/1,128/"B:STOCK"
320 FOR X=1 TO A-1
330 GET/1,X/B$(-1)
340 IF B$(0,4)<=A$(0,4)THEN 370
350 PUT/1,X/A$(-1)
360 A$=B$
370 NEXT X
380 PUT/1,A/A$(-1)
390 CLOSE/1/
400 INPUT "MORE? Y OR N        ",G$
410 IF G$="Y"THEN 10
420 RUN "PICK"
430 END
>>
```

Line 110–140 reads the highest record number from the file. POINTER stores it as variable A and increments the contents of the file by 1.

Lines 150–201 prompt the user to input the relevant data on the new item of stock. The function of lines 210–245 is to convert the numeric data into string format and insert it in the correct positions in the 128-byte string A$.

Two important functions are performed by lines 310–390. It stores A$, which is the data on the new item, on the disc and also sorts it, so that the description of the item is in alphabetical order with respect to the other records on the file.

The remaining lines of the program ask the user if he wishes to insert another record on to the file; if so, we branch to the beginning of the program; otherwise PICK is loaded from disc.

```
10 DIM A$(128)
20 DIM X$(30)
25 @ : @ : @
30 @ "ITEM NO        DESCRIPTION
   PRICE R     PRICE P     VAT %     SUPPLIER"
150 OPEN/1,10/"B:POINTER"
160 GET/1,2,1/A
170 CLOSE/1/
190 OPEN/1,128/"B:STOCK"
200 FOR Q=1 TO, A-1
210 GET/1,Q/A$(-1)
230 P$=A$(31,40)
235 C$=A$(51,60)
240 V$=A$(41,50)
245 S$=A$(61,70)
250 P=VAL (P$)
255 C=VAL(C$)
260 V=VAL(V$)
265 S=VAL(S$)
270 X$=A$(0,30)
280 @Q;TAB(10);X$;TAB(41);P;TAB(52);C;TAB(63);
    V;TAB(74);S
290 @
300 NEXT Q
310 CLOSE/1/
350 @ : @ : @ : @
360 RUN "PICK"
370 END
>>
```

List produces a complete printed list of all the data on the stock file under the headings laid out in the print statement on line 30. Lines 150–170 read the highest record number from POINTER and store it as variable A. A is, in fact, the number of the next record to be entered by NEW; thus the highest existing record number is A-1.

Line 190 opens the file STOCK on drive B and line 200 sets up a FOR-NEXT loop, to read all the records on that file from record 1 to record A-1.

Lines 200–280 get record number Q

(continued on next page)

(continued from previous page)

from the file as A. This string is then split into subrecords, to obtain the stored data which is printed-out in column form by line 280. This loop continues until all records have been printed. The program then loads PICK.

```
10 DIM A$(135)
12 DIM B$(128)
15 DIM X$(100)
110 INPUT "NAME"                ",A$(1,25)
130 INPUT "ADDRESS"             ",A$(26,50)
140 INPUT "                    -",A$(51,75)
150 INPUT "                    -",A$(76,100)
160 INPUT "                    -",A$(101,125)
240 INPUT "DATE? (EG 12.8.78)  ",A$(126,135)
252 INPUT "IS THIS CORRECT? Y OR N ",G$
254 IF G$="N"THEN 110
274 INPUT "NUMBER OF DIFFERENT ITEMS IN
    QUOTATION?       ",T
280 @ "INPUT THE ITEM NUMBER FOLLOWED BY
    THE QUANTITY IN THE FORM -1,2"
300 FOR A=1 TO T
310 @A; : INPUT B(A),C(A)
320 IF B(A)=0 THEN 340
330 NEXT A
340 INPUT "IS THIS CORRECT? Y OR N ",G$
350 IF G$="N" THEN 260
360 INPUT "MOVE PAPER TO TOP OF FORM THEN
    TYPE /G/",G$
370 IF G$="G" THEN 380
375 GOTO 360
380 @ : @ : @ : @ : @ : @ : @
390 @"        PRACTICAL COMPUTING, 2 DUNCAN
    TERRACE, LONDON N1. 01-278 9517."
395 @ : @ : @"   TO:   ";
400 @A$(1,25)
410 @TAB(9);A$(26,   50)
420 @TAB(9);A$(51,75)
430 @TAB(9);A$(76,100)
440 @TAB(9;A$(101,125)
500 @TAB(9);A$(126,135)
510 OPEN/1,10/"B:POINTER"
520 GET/1,1,1/W
530 PUT/1,1,1/W+1
540 CLOSE/1/
550 @ : @ : @          QUOTATION NUMBER   ";W
555 @ : @ : @
556@"———————————————————————————————————"
560 @"QUANTITY      DESCRIPTION
    UNIT PRICE      VAT      TOTAL PRICE"
565 @ : @
568 U=0 : Y=0
570 OPEN/1,128/"B:STOCK"
580 FOR A=1 TO T
582 IF A=14 THEN GOSUB 1500
584 IF A=28 THEN GOSUB 1600
586 IF A=42 THEN GOSUB 1700
590 GET/1,B(A)/B$(-1)
600 P$=(31,40)
610 V$=B$(41,50)
620 P=VAL(P$)
630 V=VAL(V$)
632 S=INT((P★C(A))★100)/100
634 0=INT(((S/100)★V)★100)/100
636 S=S+0
638 X$=B$(0,29)
640 @TAB(3);C(A);TAB(10);X$;TAB(40);P;TAB(50);
    O;TAB(60);S
642 Y=Y+0 : U=U+S
650 NEXT A
660 CLOSE/1/
661 @ : @
662 @TAB(40);"———————————————————"
664 @TAB(40);"TOTAL VAT ";Y
665 @
666 @TAB(50);"TOTAL        ";U
670 @ : @ : @ : @ : @
680 @ : @ : @ : @ : @
682 INPUT "          ",G$
684 IF G$="G"THEN 690
686 GOTO 682
690 INPUT"MORE? Y/N   ",X$
700 IF X$="Y"THEN 110
740 RUN "PICK"
750 END
1500 N=2
1510 GOTO 2000
1600 N=3
1610 GOTO 2000
1700 N=4
1710 GOTO 2000
2000 @ : @ : @ : @ : @ : @ : @ : @ : @ : @ : @
     @ : @ : @ : @
2010 @"PAGE   ";N
2020 RETURN
>>
```

Input Section

This is the first of the output programs. Lines 110 to 350 comprise the data input section. The program thus requests us to input the recipient's name, address and the date. We are also asked to input the number of items in the quotation for each different item; we then input the item number and the number of units of that item.

The item number is the same as the record number and is obtained from the complete list of stock lines generated by LIST.

The program then asks the user to make sure that there is a new sheet of paper in the printer before it prints the quotation. The section of the program which prints the actual quotation lies between lines 380 and 686. Lines 380, 390 print a heading for the quotation eight lines down from the top of the sheet.

Lines 395–500 then print underneath the heading on the left-hand side the name and address of the person to whom it is being sent, and the data. The current quotation number is obtained in lines 510–540 from a record in POINTER and this is printed on the quotation in line 505.

The column headings of the quotation are printed in lines 556–560. The purpose of the instructions in lines 582–586 and the subroutine called by them in lines 1500–2020 allows long costings with up to 56 items to be laid in a neat page format with 14 items per page. The first item number is stored in B(1) and the quantity of that item in C(1); the next item is stored in B(2) and so on.

Loop for values

In line 580 we set up a loop for values of A between 1 and T where T is the number of items in the quotation, this number having been entered in line 274. We are thus able to read the records whose numbers are stored in B(A) in line 590; the string representation of the numeric variables is then converted back to numeric form and used in various calculations.

Each calculated entry is then printed in the correct format by line 640; having printed all T entries the total value of the costing is printed-out in lines 661–670. The purpose of lines 682–686 is to allow the user to advance the paper in the printer to the end of the current sheet before doing another quotation, or exiting to PICK.

Next month I will conclude this article with the listings of the remaining three programs, together with some sample runs from all seven programs. I will also show how this suite of programs is used, and give some further ideas on how it can be expanded and modified.

Handy device

FOR the keen type who likes to build from scratch there is a handy device from Comart of St Neots. The company has designed and is manufacturing a chassis and power supply unit, the microBox, which can accommodate up to six S100 printed circuit cards. The box costs £70 and £195 with the power supply.

It is available by direct order through the U.K.-wide Comart network.

For information: Comart, PO Box 2, St Neots, Huntingdon, Cambs PE19 4NY. Tel: 0480 215005

A PRACTICAL GLOSSARY

Continuing the terminological gamut from D to E

Disc

A disc is a disc usually of metal, coated with a material on which data can be recorded on tracks. The tracks are concentric rather than the spiral of an LP. Read/write heads can position themselves quickly over the required data without having to get through all the preceding storage area; that's why disc storage is described as *direct-access* (qv) or random access, though it is not really random. Discs are fixed or removable. Fixed discs have permanently-inserted read/write heads, so they are very fast at reading or writing data. Obviously the disc can't be used to back-up the system via a *dump* (qv), or for file storage but it can be used as an extension of main memory. Fixed discs are expensive and rare on smaller systems, except when used in combination with a cartridge disc. Removable or exchangeable discs can be used for back-up and you can keep files on them for use as and when required. Access is slower than on fixed disc but much faster than tape. In the removable category you get:

Floppy discs. Everybody's favourite because you get cheap and reasonable fast direct access. Small and large size, single-and-double density; plenty of variety there.

Cartridge discs. The next step up. Usually 5 to 20 megabytes stored in what looks like a large plastic plate, two inches thick and inserted into the front of the drive —some go into the top. Faster than floppies, of course.

disc packs. For the big boys only; usually 20MB up (as fast as 300M). A stack of discs, normally six or 11 for protection in a plastic hood is the cheapest way to store plenty of data.

Data modules. The latest IBM goodie; very expensive but very clever way to store about 317·5 MB in a removable sealed unit with its own built-in read/write heads.

Drive

Device which transports some recording medium, usually discs.

Dump

Transfer the contents of main memory on to backing storage, typically to give you a security copy.

Duplex

System which permits transmission in both directions simultaneously.

EAN

An uncharacteristically short acronym, EAN stands for European Article Number. Designed to simplify classification of consumer items, it appears typically as the bar code—the row of little black lines on your tins of rhubarb (or whatever).

EAROM

Electrically-Alterable Read-Only Memory. It is effectively a synonym for EPROM and you can probably forget it—RAM, ROM and PROM are the significant types of memory and we will be reaching them later.

EBCDIC

Extended Binary Coded Decimal Interchange Code. One of the two principal character codes, the other being ASCII—as you will know if you read the first instalment of the glossary. EBCDIC is another helpful IBM contribution to standardisation.

The EBCDIC code allows for many more control characters and special graphics symbols including, the highly-useful 'hook', 'fork' and 'chair', than does ASCII.

Most micros usually stay with ASCII. We won't be dogmatic about it because some systems might have to communicate with an IBM mainframe, though you might still be able to use ASCII code.

These character codes become important when you're attaching terminals. That is why ASCII is so popular—plugging-in terminals is simplified when you can be certain that the list and the peripheral will assign the same meanings to the same bit patterns. Connecting an IBM terminal which generates EBCDIC code is almost impossible.

ECL

Emitter-Coupled Logic, a fast logic circuit used in the fastest computers. Also known as MECL—Monolithic ECL. Now forget it.

ECMA

It sounds like a skin complaint but it stands for the European Computer Manufacturers' Association. Although it specifies Europe, included in the membership are branches of the large American corporations. The clans gather and establish standards which subsequently are rendered obsolete. The best such joke was ECMA agonising over a standard for 80-column punched cards; when the in-fighting had produced a result, IBM announced the totally different 96-column card and very quickly followed with the floppy disc, which effectively outmoded cards altogether. Well, perhaps that's a matter of opinion; let us say the floppy disc contributed to the use of a new type of computing which has meant the decline of the 80-column card.

Edit

An instruction or group of instructions (or a button) causing data to be inserted, deleted, or relocated. Editing is, of course, a vital feature of word processing equipment—it is about the only feature on some.

An *editor* in this case, is not an individual with ever-poised red pen but a program for editing data.

EDP

Electronic Data Processing is what the Americans call dp or plain old data processing. They're correct, of course. *Electronic* data processing is not the only way to process data. EDP is what computers are for. They process—sort, list, modify, re-arrange—data electronically. The result is, hopefully, information.

EDS

Exchangeable Disc Store. This is basically ICL terminology but it is a neat abbreviation to cover removable discs—cartridge discs and disc packs.

The usage derived from ICL normally puts a figure after the abbreviation, denoting the capacity of the disc drive. So an EDS 5 is a 5 megabyte unit, probably a cartridge disc at that capacity; EDS 30 is a 30MB drive and probably a pack.

EDSAC

The Electronic Delay Storage Automatic Computer was the first of the computers as we know and love them. It was constructed in the mathematics department of Cambridge University and its first real task was to calculate a table of prime numbers in 1949. EDSAC's other main claim to fame is that it shows you what happens when you don't try to force your acronyms to make words like BUCKET, BITWIDDLE, and FAME. Here endeth the history lesson.

Elapsed time

This usually means the time taken to carry-out a particular job—it might be called 'clock time' by some people. Processing time ('run time' or 'mill time') is very different. You can be on the computer for several hours but the processor will have been active for a few seconds.

Elapsed time is defined formally as the time between entering the data and initial instructions and extracting the required final information.

Electronic composer

A style typewriter machine which simplifies type-setting and upsets trade unionists. Alternatively, a micro which can turn out *I wanna hold your hand* in the style of Stockhausen.

Electro-sensitive printers

Low-cost, non-impact printers tend to use electro-sensitive mechanisms and can be considered as a speedy alternative to thermal printers. Thermal printers build-up a character by scorching a dot on to specially-coated paper; electro-sensitive printers do the same but with electricity rather than heat. Since singeing takes some time, the electrostatic alternative is usually faster—the clever SCI rotary printer is rated at 2,200 characters per second. The fastest thermal printer we've seen is Olivetti's at 80cps—Dataproducts is now selling this mechanism.

The Centronics Micro-I is another of the new breed of electro-sensitive printers, a development by Sharp in Japan which prints at 240cps.

The problem with these super-fast printers—end-user prices are creeping below £800 now—is that you don't get a very white paper from them.

Electrostatic printer

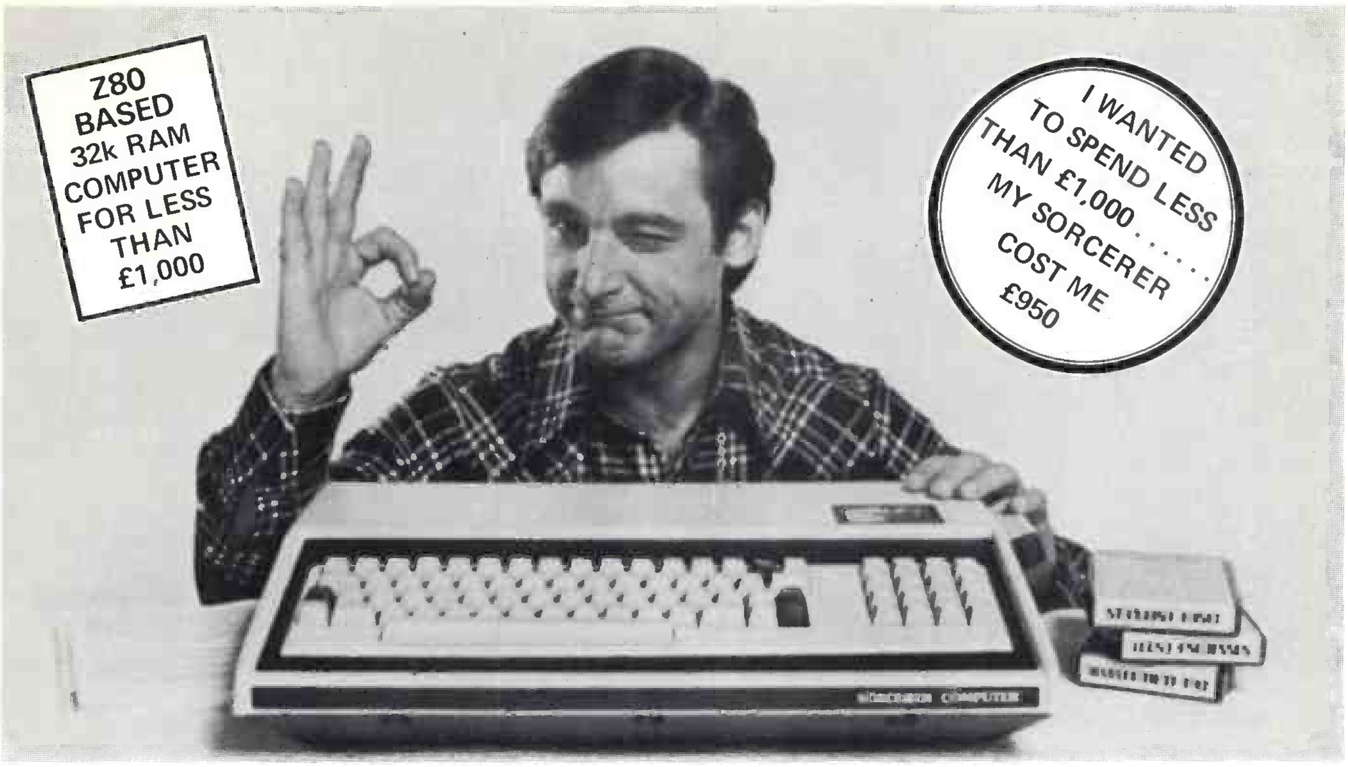
Electrostatic printers are non-impact, so you don't get copies but they are also quiet and fast because they don't involve many moving parts. You need special paper with a di-electric coating, though. To cut short a longish story, the technique involves charging the paper in a dot pattern—characters and shapes are built up in a matrix, as with impact matrix printers—and then passing the paper through a toner solution, which causes black particles to adhere to the charged dots. Hey presto—black dots on white paper.

Electrostatic devices are usually fairly expensive and are promoted as printer/plotters, with resolutions of 400 or more dots per inch; output speeds for text can be more than 2,000 lines per minute.

Empty medium

Hold on to you're hats, it's joke time again. Empty medium contrasts with virgin medium and neither has anything to do with redundant nubile spiritualists. Virgin medium is completely untouched, like a coil of paper tape. Empty medium is ready-to-record data—paper tape punched with feed holes, perhaps. You'll never hear either term but the first person to work them both into the same sentence during a normal conversation receives the *Practical Computer* award for effluence beyond the call of duty.

THE SORCERER HAS ARRIVED



Introducing the personal computer you've waited for. The Exidy Sorcerer.

I didn't buy my personal computer until I found the one that had all the features I was looking for.
The Exidy Sorcerer does everything I wanted to do and a few things I never dreamed of.
It isn't magic. Exidy started with the best features of other computers, added some tricks of their own, and put it all together with more flexibility than ever before available. Presto! My reasons for waiting just disappeared.

I wanted pre-packed programs. Software on inexpensive cassette tapes for the Sorcerer is available from Exidy and many other software makers.

I wanted user programmability The Sorcerer's unique plug-in ROM PAC™ Cartridges contain programming languages such as Standard (Altair 8k*) BASIC, Assembler and Editor (so I can develop system software), operating systems such as DOS (so I can also use FORTRAN and COBOL) and applications packages such as Word Processor.

*Altair is a trademark of Pertec Computer Corp.

I wanted Graphics, and the Sorcerer is super. Its 256 character set — more than any other personal computer — includes 128 graphic symbols that I can define.

I wanted high resolution video. With 122,880 points in a 512 x 240 format, I get the most detailed illustrations.

I wanted to display more information. The Sorcerer displays 1920 characters in 30 lines of 64 characters — equal to a double-spaced typed page.

I wanted a full, professional keyboard. The Sorcerer's 79-key data processing keyboard provides designated graphics, the complete ASCII character set in upper and lower case, and a 16-key numeric pad.

I wanted memory. The 12k of ROM holds a Power-On Monitor and Standard BASIC; 32k of RAM is supplied on board.

I wanted expandability. Serial and parallel I/Os are built in, and the optional 6-slot S-100 expansion unit lets my system grow.

I wanted a computer that's easy enough for children to use. I just connect my Sorcerer to a video display and a cassette tape recorder, and if I have any questions the easy-to-understand Operation and BASIC Programming manuals have the answers.

I wanted to buy from an experienced Manufacturer. In five years Exidy has become the third largest producer of microprocessor-based video arcade games.

I wanted to spend less than £1,000. (This is where COMP. does a little magic). My Sorcerer cost me £950!.

Now, what are you waiting for?

Call COMP. on 01-441 2922 or write to

COMP COMPUTER COMPONENTS

14 Station Road,
New Barnet, Herts. EN5 1QW.

(Price shown ex. VAT)

DEALER ENQUIRIES INVITED, A LIMITED NUMBER OF DEALERSHIPS STILL AVAILABLE.

We're getting bigger to give you a better service

We've opened a Sales and Customer Service Centre in London. And our new factory is fully operational. Southwest Technical Products provide a range of superb computer systems with technical backing second to none.

Systems
To suit all types of user – OEM, process control, data handling, small business systems, and all accounting functions.

Software
Low cost packages for word processing, selective mailing, progress control and invoicing. Our Software Development Unit available to prepare programmes to customer specification.

Training
Inexpensive courses (at Dover Street): BASIC – programming for the businessman; microcomputers in EDUCATION; WORD PROCESSING made easy; SOFTWARE DEVELOPMENT – make your microprocessor work.

Maintenance
Comprehensive national service by Computer Field Maintenance Ltd.

Sales Office: 38 Dover Street, London W1.
Tel: 01-491 7507 Telex: 268913.
Factory: 12 Tresham Road, Orton Southgate, Peterborough.
Tel: 0733-234433 Telex: 32600.

SWTPc

the symbol of reliability

● Circle No. 310