# Practical Computing

50p

An ECC Publication. Volume 1 Issue 2

October 1978

## Taking the chore out of VAT

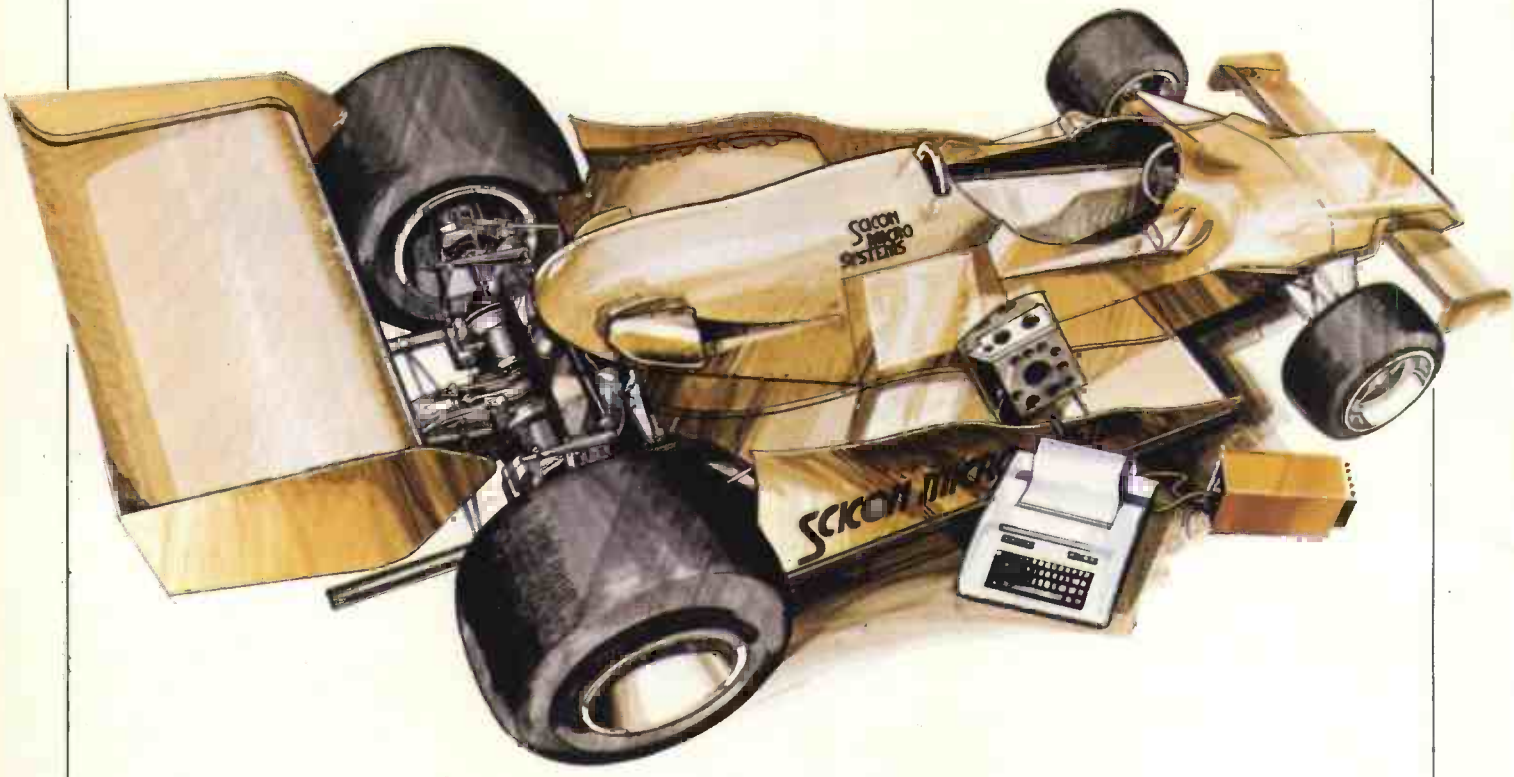
## Two bargain terminals

## Wiring for sound

## We review Pet —the £700 computer


WIN A FREE COMPUTER

# 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, Scientific Control Systems Limited, Sanderson House, 49-57 Berners Street, London W1P 4AQ.

SCICON MICRO SYSTEMS

# Contents

# MK14-the only low-cost keyboard -addressable microcomputer!

## The new Science of Cambridge MK14 Microcomputer kit


**Just £39.95 (+ £3.20 VAT, and p&p)**

**The MK14 National Semiconductor Scamp based Microcomputer Kit gives you the power and performance of a professional keyboard-addressable unit – for less than half the normal price. It has a specification that makes it perfect for the engineer who needs to keep up to date with digital systems, or for use in school science departments. It's ideal for hobbyists and amateur electronics enthusiasts, too.**

But the MK14 isn't just a training aid. It's been designed for practical performance, so you can use it as a working component of, even the heart of, larger electronic systems and equipment.

**MK14 Specification**

* Hexadecimal keyboard
* 8-digit, 7-segment LED display
* 512 x 8 Prom, containing monitor program and interface instructions
* 256 bytes of RAM
* 4MHz crystal
* 5V stabiliser
* Single 6V power supply
* Space available for extra 256 byte RAM and 16 port 1/0
* Edge connector access to all data lines and 1/0 ports

**Free Manual**

Every MK14 Microcomputer kit includes a free Training Manual. It contains operational instructions and examples for training applications, and numerous programs including math routines (square root, etc) digital alarm clock, single-step, music box, mastermind and moon landing games, self-replication, general purpose sequencing, etc.


**Designed for fast, easy assembly**

Each 31-piece kit includes everything you need to make a full-scale working microprocessor, from 14 chips, a 4-part keyboard, display interface components, to PCB, switch and fixings. Further software packages, including serial interface to TTY and cassette, are available, and are regularly supplemented.

The MK14 can be assembled by anyone with a fine-tip soldering iron and a few hours' spare time, using the illustrated step-by-step instructions provided.

**Tomorrow's technology – today!**

*"It is not unreasonable to assume that within the next five years . . . there will be hardly any companies engaged in electronics that are not using micro-processors in one area or another."*

Phil Pittman, Wireless World, Nov. 1977.

The low-cost computing power of the microprocessor is already being used to replace other forms of digital, analogue, electro-mechanical, even purely mechanical forms of control systems.

The Science of Cambridge MK14 Standard Microcomputer Kit allows you to learn more about this exciting and rapidly advancing area of technology. It allows you to use your own microcomputer in practical applications of your own design. And it allows you to do it at a fraction of the price you'd have to pay elsewhere.

Getting your MK14 Kit is easy. Just fill in the coupon below, and post it to us today, with a cheque or PO made payable to Science of Cambridge. And, of course, it comes to you with a comprehensive guarantee. If for any reason, you're not completely satisfied with your MK14, return it to us within 14 days for a full cash refund.

Science of Cambridge Ltd,
6 Kings Parade,
Cambridge,
Cambs., CB2 1SN.
Telephone: Cambridge (0223) 311488

To: Science of Cambridge Ltd,
6 Kings Parade, Cambridge,
Cambs., CB2 1SN.

Please send me an MK14 Standard Microcomputer Kit. I enclose cheque/ Money order/PO for £43.55 (£39.95 + 8% VAT and 40p p&p).

Name _____

Address (please print) _____

_____

_____

_____

Allow 21 days for delivery.

**Science of Cambridge**

PC/9

Practical Computing

Managing Editor
**Dennis Jarrett**

Computabits Editor
**Nick Hampshire**

Production Editor
**Harold Mayes**

Advertisement Manager
**Erica Gibson**

Subscription Manager
**Annabel Hunt**

Publisher
**Wim Hoeksma**

Publishing Assistant
**Carole Fancourt**

Managing Director
**Richard Hease**

Editorial, Advertising and subscriptions: 01-278 9517.

Practical Computing is published by EEC 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).



**We will pay £5 for the best letter published each month. Here is this month's winning letter:**

# Home-brew club for London

ON WEDNESDAY, October 5 at 6.30 pm, we are holding the inaugural meeting of the North London Hobby Computer Club in Room 47 in the Old Polytechnic Building at Holloway Road, just opposite Holloway Road underground station on the Piccadilly Line.

The Department of Electronic and Communications Engineering and the Polytechnic of North London have made available many resources for this venture. Within the department there are two Pets, with a third coming, four SWIP 6800 computer systems, with floppy discs, printers and VDUs and some Kim and Motorola microcomputer systems. Most will be available for use, as will some Pets and SWTP systems in other departments.

As we envisage the club at the moment little "home-brew" activities are anticipated before Christmas, with any meetings centring on talks by manufacturers and discussions on programming.

From the New Year, however, we anticipate three sets of activities running concurrently, or sequentially—it all depends on how many people turn up. They are short courses on programming, Basic and machine level; a home-brew section using the facilities of the department—up to 35 people can solder and test at the same time—and introductory talks and discussions for those anticipating their own systems.

As you can see, we are preparing a varied programme which should be of interest to a wide variety of people. Obviously, students from the Poly will be coming, but we want to emphasise that this is a club open to all interested. The Poly will be providing some back-up, especially with expert staff and other facilities. This is all part of the Community Development Programme instituted recently.

I hope *Practical Computing* will be able to help us get this off the ground. Those organising it are members of the Amateur Computer Club, as well as lecturers in digital electronics.

**Robin Bradbeer,**
**Senior Lecturer,**
**Acting Club Secretary**

# MPU system prices

THERE has been some controversy recently concerning the price of American MPU systems in the U.K. There are companies who simply change the $ sign for a £ sign and call that the U.K. price; most are more reasonable. My own company does not, as yet, sell any American systems—we build our own—but recently I have been working out some figures.

Let us take as an example a reasonably-sized American microcomputer system which sells for a basic $2,000. A U.K. agent receives 25 percent discount but will usually have to order 10 units at a time; he therefore puts up $15,000. This is probably done by letter of credit; it could be three months before he gets delivery.

Adding interest charges at 10 percent on $15,000 for three months produces a total outlay of $15,500. Add freight and duty to the imported product and you finish with a total outlay of $18,000 for 10 units. A quick conversion reveals this to be about £1,000 per unit. You can count in another 15 percent for technical back-up and physical stocks; then there will be a 50 percent mark-up. You end up at about £1,600-£1,700.

Why should the U.K. dealer receive the equivalent of 30-40 percent discount when his American cousin gets only 25 percent? There are several possible reasons:
Low-volume sales compared to the U.S. market which is 12-18 months ahead of ours;
Need for more technical back-up due to the low level of customer education in U.K.

Part of the profits may be disappearing to a central European distribution office in France, Germany or Switzerland—this may also include currency conversions which work to the benefit of the Europeans, so the U.K. dealer may be getting only about 25 percent discount.

Because of hobby magazines, local computer stores and the next-door neighbours, the average American buyer has a good idea of the product and price before he shops. His U.K. counterpart will want several demonstrations and a long talk with an expert before he even considers taking away a data sheet to study. So the actual cost of each sale is higher in the U.K. than in the U.S.

Sales volumes are low because of the relatively high prices of the equipment in the U.K. Even if the U.K. dealer made no profit and sold the example system for £1,000, this probably represents a quarter of the average annual salary for a programmer or engineer, the most likely first customers.

In the U.S. the typical engineer or programmer can expect to get about $20,000 pa; our $2,000 example is equivalent to only one-tenth of his salary.

This difference is magnified by the higher salary rate paid to foreign engineers and the costs of importing, including

(*continued on page* 11)

# LET YOUR SYSTEMS GROW WITH S.E.E.D.

*Please send S.A.E. for details to:-*

## STRUMECH ENG. ELECTRONICS DIV.

### PORTLAND HOUSE, COPPICE SIDE, BROWNHILLS, WALSALL

### "Sole U.K. Distributors"

● Circle No. 107

---

# THE NEWBEAR COMPUTING STORE

The Bear announces its new store at:-
2 Gatley Road,
Cheadle,
Cheshire
Tel: 061-491 0134
callers welcome, mail order to Newbury.

**PETITEVID VDU KIT £85**

110 to 1,200 baud V24, 64 ch × 16 line scrolling, all on 8in. × 4in. PCB. Needs TV set, UHF Modulator and ASCII keyboard.

**BOOKS**

| | | P&P |
|---|---|---|
| Computer designs. 77-68 a 6800 Microcomputer | £7·50 | 50 |
| Spare diagram set for 77-68 | £1·50 | 50 |
| WB-1 a TTL Microcomputer | £6·50 | 50 |
| Spare diagram set for WB-1 | £1·00 | 50 |
| Zilog Z-80 Technical Manual | £4·00 | 50 |
| Z-80 PIO Manual | £2·50 | 50 |
| Z-80 Programming Manual | £4·00 | 75 |
| Motorola. Understanding Microprocessors | £2·75 | 30 |
| M8800 Microprocessor Programming Manual | £4·50 | 50 |
| M6800 Applications Manual | £9·50 | 1·00 |
| M.O.S. Technology KIM 1 User Manual | £5·00 | 75 |
| 6500 Programming Manual | £5·00 | 75 |
| 6500 Hardware Manual | £5·00 | 75 |
| Adam Osbourne Introduction to Microcomputers | | |
| Vol. 0 Beginners Book | £5·25 | 50 |
| Vol. 1 Basic Concepts | £5·95 | 50 |
| Vol. 2 Some Real Products | £11·95 | 1·00 |
| 8080 Programming for Logic Design | £5·95 | 50 |
| 6800 Programming for Logic Design | £5·95 | 50 |
| Some Common Basic Programs | £8·95 | 50 |
| Payroll with Cost Accounting in Basic | £9·95 | 1·00 |
| Sybex Microprocessors | £8·00 | 50 |
| Microprocessor Interfacing Techniques C20 | £8·00 | 50 |
| Scelbi 8080 Software Gourmet Guide Cookbook | £7·95 | 50 |
| 6800 Software Gourmet Guide Cookbook | £7·95 | 50 |
| The Scelbi Byte Primer | £9·95 | 1·00 |
| What to Do After you Hit Return | £7·00 | 75 |
| PCC Reference Book on Home Computers | £4·95 | 50 |
| Instant Basic | £4·95 | 75 |
| My Computer Likes Me | £1·65 | 30 |
| Games with a Pocket Calculator | £1·75 | 30 |
| Games Tricks & Puzzles for a Hand Calculator | £2·45 | 30 |
| Best of Creative Computing Vol. 1 | £6·95 | 75 |
| Best of Creative Computing Vol. 2 | £6·95 | 75 |
| Hobby Computers are here | £3·99 | 50 |
| Best of Byte | £8·95 | 75 |
| 101 Basic Computer Games | £5·50 | 75 |
| The Home Computer Revolution | £3·45 | 50 |
| Computer Lib | £5·95 | 50 |
| First Book of Kim | £7.00 | 50 |

## Hardware Components Section

**MEMORIES**

| | | | |
|---|---|---|---|
| 2102-1 | £1·25 | 4027 1250ns | £4·50 |
| 2102L-1 | £1·35 | 2114 (450ns) | £10·00 |
| 2112 | £3·04 | 4116 (250ns) | £19·00 |
| 2513 | £8·50 | 2708 (450ns) | £10·50 |
| | | 2716 (1I) | £22·50 |

**SPECIAL MOTOROLA MICROCOMPUTING ICs**

| | | | |
|---|---|---|---|
| MC6800P | £9·26 | SWATBUG | £16·00 |
| MC6820P | £4·63 | MC6830L7 | £11·33 |
| MC6850P | £8·00 | MC8602P | £2·88 |
| MC6810AP | £3·61 | MC14538P | £3·69 |
| | | MC3459 | £2·00 |

**ZILOG MICROPROCESSORS**

| | | | |
|---|---|---|---|
| Z-80 CPU 2·5MHz | £15·50 | Z-80A CPU | £20·50 |
| Z-80 PIO PS | £10·00 | Z-80A PIO | £11·55 |
| Z-80 CTC PS | £10·00 | Z-80A CTC | £11·55 |

**MICROPROCESSORS**

SC/MP Mk II £10·30 6502 £14.93 8080 £6·00

**INTERFACING ICs**

**BUFFERS**

| | | | | | |
|---|---|---|---|---|---|
| SFF96364 | £16·20 | VDU | | 81LS95 | £1·30 |
| MC1488P | £1·40 | | | 81LS96 | £1·30 |
| MC1489P | £1·40 | | | 81LS97 | £1·30 |
| 75150P | £1·30 | For V24 | | 81LS98 | £1·30 |
| 75150N | £1·20 | RS232C | | 8T26 | £1·84 |
| 75154 | £2·50 | | | 8T85 | £1·60 |
| 4N33 | £1·95 | | | 8T97 | £1·60 |
| AY-5-1013 | £5·40 | | | 74367 | £1·30 |
| 8402 | £4·50 | | | | |

**LOW POWER SCHOTTKY TTL PRICE LIST**

| | | | | | |
|---|---|---|---|---|---|
| 74LS00 | ·18 | 74LS30 | ·16 | 74LS109 | ·38 |
| 74LS01 | ·21 | 74LS32 | ·34 | 74LS112 | ·38 |
| 74LS02 | ·21 | 74LS33 | ·40 | 74LS113 | ·38 |
| 74LS03 | ·21 | 74LS37 | ·30 | 74LS114 | ·38 |
| 74LS04 | ·26 | 74LS38 | ·30 | 74LS125 | ·50 |
| 74LS05 | ·26 | 74LS40 | ·27 | 74LS126 | ·50 |
| 74LS08 | ·21 | 74LS42 | ·88 | 74LS132 | ·95 |
| 74LS09 | ·21 | 74LS47 | ·96 | 74LS136 | ·38 |
| 74LS10 | ·21 | 74LS48 | ·96 | 74LS138 | ·65 |
| 74LS11 | ·26 | 74LS51 | ·21 | 74LS139 | ·65 |
| 74LS12 | ·21 | 74LS54 | ·21 | 74LS151 | ·96 |
| 74LS13 | ·55 | 74LS55 | ·21 | 74LS153 | ·58 |
| 74LS14 | £1·26 | 74LS73 | ·34 | 74LS154 | £1·45 |
| 74LS15 | ·21 | 74LS74 | ·38 | 74LS155 | £1·05 |
| 74LS20 | ·21 | 74LS75 | ·55 | 74LS156 | £1·05 |
| 74LS21 | ·26 | 74LS76 | ·34 | 74LS157 | ·60 |
| 74LS22 | ·21 | 74LS78 | ·34 | 74LS158 | ·65 |
| 74LS26 | ·31 | 74LS83 | £1·05 | 74LS160 | £1·22 |
| 74LS27 | ·21 | 74LS86 | ·38 | etc— | |
| 74LS28 | ·40 | 74LS107 | ·38 | | |

**NEW RANGE OF LOW COST S100 8K RAM BOARDS, PCBs, ETC . . . PLEASE SEND FOR LATEST LIST.**

**BEAR BAGS (Kits)**

| | |
|---|---|
| 1 177-68 components and edge connectors | £49·50 |
| 2 77-68 Led's and switches | £14·95 |
| 3 77-68 Power Supply | £17·95 |
| 4 77-68 19" 5a Rack and Backplate | £26·70 |
| 5 77 68 4K Ram PCB and components | £75·00 |
| 6 77 68 Mon 1 PCB and components | £50·00 |
| 7 4K Ram Exorciser PCB and components | £71·50 |
| 8 8K Ram Exorciser PCB and components | £160·00 |
| 9 Petitevid VDU Kit | £85·00 |
| 10 Kansas City Cassette Interface | £18·95 |
| 11 UHF modulator | £4·50 |
| 12 77 68 VDU | |

Available direct from us or from your local Bear Bag Stockist (dealer enquiries welcome).

**VIM 1**

The new 6502 based micro from Synertak. Fully assembled and tested £199.00 plus 8% VAT. Send for data Sheet. Carriage £1.00.

**PCB and COMPONENTS 77·66 PRINTED CIRCUIT BOARDS**

| | |
|---|---|
| CPU | £10·00 |
| 4K RAM | £10·00 |
| Soft Monitor | £10·00 |
| VDU Keyboard Interface | £10·00 |
| ROM Monitor | T.B.A. |
| Veroboard 8 × 8 | £4·10 |
| Veroboard backplate 8 × 17 | £4·30 |
| Hexadecimal Keyboards | £61·25 |
| FND 500 7 Segment display | £1·05 |
| Decoder chip | £3·45 |
| ASCII Keyboard (unencoded) | £27·50 |
| Keyboard assembled and tested | £86·00 |
| Keyboard case | £20·33 |

Goods are normally shipped within 24 hours, subject to availability. Barclaycard & Access. VAT at 8% for Hardware Components. 30p postage and packing unless otherwise stated. Cheques to be made out to The Newbear Computing Store. Send for an up-to-date catalogue to The Newbear Computing Store, 7 Bone Lane, Newbury 0635-49223. Callers are welcome Monday to Saturday 9.00a.m.-5.30p.m., but please ring to let us know when you are calling.

● Circle No. 108

(*continued from page 9*)
the profits made by the foreign designers and manufacturers.

At present, the U.K. boasts something like six companies making home and small business computers based on microprocessors, my own company being one of them. Recently we had 95 percent of our kits waiting on the shelves for three months, due to the 'memory famine'. During that time we have had to watch the customers dwindle and the bank overdraft charges rise. We hope soon to be able to deliver from stock, but the majority of the advertising and selling done three and four months ago has been wasted. Other U.K. companies are in similar situations for various reasons.

The Government is considering investing £50 million in a product which has not yet been designed (the 64K RAM) to be made in a factory which has not yet been built. Foreign manufacturers are already beginning to produce prototypes to be made by experienced personnel in existing factories.

Why not invest some of that £50 million in an attempt to stimulate a home market by reducing the prices of the end products? There are several U.K. companies capable of designing better microcomputers than the Americans and with the world-famous U.K. software in them.

As an example, we considered buying an American microprocessor development system for £365; in fact, we designed our own with improved hardware, better interface facilities, and much-improved software. It sells for £155, less than half of the American equivalent. A fall in component prices of about 25 percent helped but, even so, we could have done it for about half the cost of any equivalent American design.

Apply this philosophy to our $2,000 product and it would end up at about £700; and a simple home computer, such as those currently selling at £500-£700, could be built for £350-£400.

Let others cut their own throats in the component business but let us take advantage of the low costs which result, capitalising our own hardware and software design capabilities to make systems.

We have managed to sell the idea of Teletext and Viewdata to most of Europe; we should now follow with compatible low-cost microcomputer systems. When we have a large home market and a good share of the world market for micro systems, it may then be time to look at the idea of IC manufacturing in the U.K.

In the meantime, spare another thought for the 'ridiculous' profits made by importers of American systems. Some of them are genuinely interested in making a market for home computers in the U.K. If the Government is not going to help, the money has to be raised somewhere.

**John H. Miller-Kirkpatrick,**
**Technical Director,**
**Bywood Electronics.**

## Question of logistics

I READ with great interest the article which appeared in the July/August issue of *Practical Computing*.

Concerning the brief history of the personal computing industry, I must say that I think the NASCOM 1 success story tends to conflict with the conclusions you have drawn in your article and I would welcome the opportunity to expound this company's philosophy as far as the amateur market is concerned.

I am sure you are familiar with NASCOM 1 and I am also of the opinion that you will understand the original philosophy behind its design to some degree discounted and offset the factors you outlined in the early part of your article concerning relatively high prices on a cash-in-advance basis for something which proves to be of dubious engineering standards.

While I would be the first to concede that we have not been able to attain the delivery schedules we had anticipated, this was not for the reasons you have outlined but for the logistics of dealing with an indigenous manufacturer in the U.K.

I am pleased to say that our company enjoyed a great deal of interest at the Do-it-Yourself computer show and this has been followed by a reasonable number of confirmed orders. It is our intention to stay in the forefront of microcomputer design and development and we will always aim a portion of our marketing effort and expertise towards the bottom end of the market you say is developing, namely the amateur enthusiast.

I would like to wish your magazine every success in the future as I feel that dedicated publications to this new industry are of benefit to manufacturers and customers alike.

**J. A. Marshall,**
**Managing Director,**
**Nasco Sales Ltd,**
**Chesham, Bucks.**

## For the record

MY COLLEAGUES and I would like to point out some errors in biographical details at the end of your reprint of our On-line paper, *Microcomputers in the Construction Industry*.

John Paterson is a lecturer at Reading University. Ted Cogswell is a lecturer at Southampton College of Technology. My surname is Frith, not Firth.

**J. Frith**
**University of Reading.**

## Feedback for you

**WHETHER you are using or contemplating the use of computers in the home, in business or in education, the *Practical Computing* Feedback columns may be of help to you.**

**Starting next month, Feedback will deal with your problems. If you want to know how to put a system together, or to find out what is available from where and at what price, tell us about it and we will do what we can to assist.**

**If we cannot answer your query, we will publish your request and pass on the replies we receive.**

**Feedback will also be a forum for any interesting developments you may have made. Please let us know about them, so that we can tell others.**

**Write to Feedback, *Practical Computing*, 2 Duncan Terrace, London, N1 8BJ. We look forward to hearing from you.**

# Pets in the kitchen

EAST meets West at the Orient Restaurant off Tottenham Court Road in London. East, represented by Chandru Idnani, who owns the restaurant, is supplying initiative and expertise, while West—or to be more accurate South-West Technology—adds hardware, the result being a computerised restaurant.

Chandru claims that when his system becomes fully operational it will be the first of its kind, helping waiters and kitchen staff to keep track of orders, preparing bills, controlling stock levels and perhaps eventually moving into the more mundane area of payroll.

Nor does he intend to keep it to himself. "We've tried to make it as universal as possible, so that it can be used in other types of restaurant," he says. To this end he is preparing to open a computer store on the floor above his restaurant, there to sell packages not only to caterers but also to other businesses whose functions he thinks can be computerised in a similar way.

## Baking hot

As you may imagine, he is a man of considerable energy and expertise. He also has a background in mathematics and a degree in physics at University College, London, where work with a research group introduced him to computer hardware. That was in the late 1960s but it was not until the end of last year that he decided that micros could benefit his business.

He left the academic life for catering in 1969 and has looked at various ways of mechanising the business since then. A scheme to automate the preparation of some dishes—to enable him to maintain the standard of food when the chef was ill or on holiday—proved impossible to put into practice but he succeeded in simplifying the baking of bread, a task which had formerly required a man with an asbestos forearm.

Chandru has been working now on the computer system for about five months and expects to have it running in another month. It uses an SWTP 6800 with 40K bytes of RAM, a cassette interface, keyboard and printers.

He has added "bits and pieces to the micro" but it is the software which has delayed him. "I thought it would be very easy but in the end I had to get experienced help." The experienced help came from friends but even so he has written much of it himself, having learned Basic from scratch. when it's finished the program should run in about 16K.

## Coded orders

The program has two major loops—one for overall control, the other for the processing of customers' orders. When the restaurant opens in the morning there are the usual initialisation procedures—entering the date, marking any items from the 180 on the menu which are not available, and setting totals to zero if required

So business begins. A waiter —identified by a code—takes orders—broken into coded items from a table, which is again given a number. There is nothing unusual so far but the waiter then takes his notebook to the bar and details of the order are typed into the system from a numeric keypad.

The 'order loop' has three options; to initiate an order, to add to it, or to produce a bill. In the first and second cases the order will be transmitted to a PR40 printer in the kitchen; a second printer will be kept behind the bar to print bills. Chandru is using rolls of paper 4½ in. wide for this purpose.

When the customer pays, the system records the method and size of payment, handling credit cards, luncheon vouchers and complimentary meals as well as cash. At the end of a day's trading a reconciliation can be produced and a list showing the totals of each item ordered during the day. By comparing this list to stock input, Chandru has an immediate balance of stock and a guide to future buying.

There are other incidental benefits—menus printed daily, infallibly accurate and legible bills, and possibly improved service. The cost has hardly been prohibitive; Chandru estimates £2,000 for the hardware and plenty of goodwill for the software.

The customer will probably notice very little difference—the only immediate sign will be a line reading Bill Produced By Computer at the foot. Grafton Way's only oriental dancers may well cause them to overlook even that.

Chandru also has a Commodore Pet which he hopes to use in the restaurant when he can attach a printer to it. His ideas on the future development of the system include both machines. He thinks it could be particularly useful to English restaurants which, unlike his own, tend to change their menus from day to day.

He also sees a market for a grocery package, with the emphasis on stock control with, say, 500 items. Putting his money where his mouth is, his computer store will be open in four to six weeks.

When Chandru spoke to *Practical Computing* he was joined by the district's former health inspector, who had only praise for the restaurant and showed no sign of alarm at the thought of Pets and suchlike in the kitchen. That is progress indeed. ■

# Checkmate for Ian

IF you are used to cramming your programs into systems with only 2K of memory, you will have little sympathy with Ian Toyn, a 16-year old schoolboy from Lincoln. He has just written a chess-playing program but cannot test it as it is too big even for the mainframe at the local College of Technology to handle.

Ian is at Yarborough High School, Lincoln, where the maths department has been running courses since 1976.

A Teletype terminal at the school sends programs and data to the College of Technology for processing. Ian's program, however, is so big that until the college expands its system, he will not be able to run and debug it, so it looks like checkmate. ■

# TALKING TO YOUR APPLE

"I TALK to the trees", ran the song. A silly thing to do, perhaps, but there is an inanimate object to which you can talk which will understand you, the Apple II micro system.

Keen Computers of Nottingham, which markets Apple products, is selling a voice input system which allows you to give spoken commands to your machine.

The system is called the Speech Lab Model 20A and costs £165. It comprises a microphone, the voice recognition board, manual, and six demonstration programs.

The programs are either tape or listings and allow the beginner to do voice prints on the screen; play Mastermind and Blackjack; teach the cursor to go up and down, left and right; recognise people by their voices; and play Shooting Stars.

As Keen says, the advantages of using voice input are enormous, as "speech is a human's highest capacity output channel".

## Two models

Speech Lab is in two models. Model 20 has a vocabulary of 32 words. It is available as the 20S for S-100 computers and the 20A for the Apple II unit. Features include:

ROM-based software with a re-locatable program so that you can load it anywhere you have memory; after loading, the 2K ROM can be disabled under computer control.

The speech recognition program, callable from any Basic, requires 4K bytes of RAM supplied by the user, which can be located anywhere in the address space.

Hardware includes two band-pass filters with two bits of amplitude, two zero crossing detectors and a linear amplifier.

A combined hardware/software manual includes 10 experiments.

Model 50 is an S-100 bus compatible system (8080 or Z-80) capable of handling a 64-word vocabulary, using 64 bytes of storage per spoken word. Features of Model 50 include CMOS design for reliability and low power consumption. Response is real-time.

Software for Speech Lab includes seven complete programs, three of which are offered in source and on paper tape and four in source alone. The three paper tape and source are: speech basic programming language; assembly language recognition program; and hardware self-test program.

The source programs are provided in Speech Basic to plot and correlate speech data. Two recognition programs offered will clarify speech concepts.

The assembly language program uses less than 4K bytes for a 32-word vocabulary and it loads at 100H. A high-core version is available. ■


**A voice response unit working on the Apple II.**


# Home keyboard

SIRTON PRODUCTS has produced a very cheap ASCII keyboard for use with any home micro system. It costs £88.50 and is complete with case, power supply and UHF modulator, so that it can be plugged straight into a domestic TV set.

Sirton also offers a self-contained VDU system which gives a display of 16 lines of 64 characters. In addition to normal cursor controls, it can present reverse video—black on white—and the facility to make the whole screen or selected characters flash to attract the attention of the operator.

Further details: Sirton Products, 13 Warwick Road, Coulsdon, Surrey CR3 2EF. Telephone: 01-660 5617. ■

# Addition to family

CHIP manufacturer Zilog has expanded its MCZ-1 family of microcomputer systems with the introduction of a new business system, the MCZ-1/60. Like other systems in the range, it is based on the Z80 microprocessor.

It is equipped with a display and keyboard and two floppy disc drives. To make it acceptable in the business environment, it can be programmed in standard Cobol, as well as PLZ and two versions of Basic.

The minimum configuration costs £5,300, and has 32K bytes of main memory, expandable to 64K.

The MCZ-1 range starts at around £3,500 for the model 01 and is marketed in the U.K. by Memec Systems, of Thame, Oxfordshire. ■

# Design courses

A SERIES of three-day courses on the design of micro processor-based systems has been set up.

Day 1 of the course is entitled Microcomputer Systems-Fundamental concepts, day 2, Microcomputer Interfacing and Programmable Devices; and day 3, Microcomputer Software and Program Design.

Details from Prodex (Seminars) Ltd., 79 High Street, Tunbridge Wells, Kent TN1 1XZ; telephone 0892 39664.

# Catalogue update

PETSOFT, the microsoftware house specialising in programs for the Commodore Pet is now updating its catalogue of more than 60 titles monthly.

New programs include stock control, sales analysis, reformat, Life, and a backgammon program which displays the board, 'shakes' the dice, and plays either the Pet user or itself.

Petsoft says the introduction of business and application software has generated sales to universities, banks and large corporations, including ICI, Rank and the Post Office.

Special high-energy, low-noise cassettes chosen for their low drop-off characteristics, in the short C-12 length, cost £4·75 for 10 plus 50p postage, or £45 for 100, carriage paid in the U.K.

Catalogues can be obtained, from Petsoft, PO Box 9, Newbury, Berkshire, RG13 1PB. Telephone 01-353 1100, and 0635-201131. ■

# Bet with Ecstasy

"ECSTASY" may not be the first thing to register when calculators are mentioned and it does not occur often in the presence of bookmakers, either—unless you win, of course.

Efficient Computing Systems of Douglas, Isle of Man, however, has put together a system based on a Texas Instruments TI-58 programmable calculator, called it Ecstasy, and made it essential equipment for many local bookmakers.

The key to the betting shop application of the TI-58 is a device known as the Custom CROM—which stands for Constant Read-Only Memory. It is a small module which contains special programs developed by Dr Les Waller, a consultant to ECS.

The programs, which Dr Waller designed to be economical in terms of memory usage and number of steps, can calculate complex bets from double to Yankee in one operation.

Further information: John Gibbons, Texas Instruments. Telephone: Bedford 67466; or Dr Les Waller, Middlesborough 85399. ■

# Mersey beat micros

LIVERPOOL has acquired a micro shop, Micro Digital, spawned by Datapool Services, a computerised book-keeping bureau based in the city.

Bruce Everiss, the managing director, is casting the net very wide and the bureau parentage means a strong bias towards the small business market. The shop caters for customers wanting dedicated micro-packs and general-purpose micro systems and kits.

Initially, it is acting as a showroom for Apple, Nascom, Casu and Cambridge Mark 14.

Micro Digital is geared towards the hobbyist and the small business. Everiss reckons to be able to supply tailored systems to most requirements and with that in mind is developing a system around the Z80 and a British-made S100 bus.

Micro Digital is at 25 Brunswick Street, Liverpool 3; you can call on 051 – 708 8624. Open from 9 am to 5.30 pm. ■

# Expertise the key

PETS and Apples are now being sold in Tottenham Court Road, London, W1 by Euro Calc. The shop, which has been specialising in calculators, has now added these computers to its range of equipment.

Tony Manton, who runs the shop, explains that Euro Calc is concerned mainly with selling ready-built systems, for which it will supply software development effort and expertise rather than kits.

The shop opens from 9.30 am to 6pm (7pm on Thursdays) at 244 Tottenham Court Road, London, W1. Tel: 01-636 8161.

Close to Euro Calc is Heathkit, which is worth a visit. ■

# Calculator packages

TEXAS INSTRUMENTS has set up a library of cheap software packages for use on its programmable calculators. The library is called the Professional Program Exchange and includes hundreds of ready-written programs which users of TI-58/59 calculators can buy for $3 each —that's about £1.60.

It costs $15 (£8) to join the exchange and membership includes a catalogue listing all the programs available. Twenty separate categories are covered, including games, engineering and finance.

As well as buying packages, users are being encouraged to contribute their own programs to the library. Prizes are offered to those whose software is approved and added. ■

# Back-up storage for Zilog Z80

ANYONE with a microprocessor system based on the Intel 8080 or Zilog Z80 and in need of more back-up storage should be interested in a new product being marketed at the Sintrom Microshop.

The Micropolis Meta Floppy 1054 it is a four-drive minidiskette unit. It takes four 5¼in. high capacity floppy discs and provides 1·26 million bytes of on-line storage.

That exceeds the capacity of many larger floppy disc drives and Sintrom claims the 1054 has a number of exceptional technical features which make it possible. Not only is capacity high but performance is very respectable, too. Track-to-track access time is 30 milliseconds and data transfer can be as high as 250,000 bytes per second.

One other handy feature is a disc insertion interlock which prevents accidental damage to the disc by ensuring that the user cannot close the drive door until the disc is positioned properly. An illuminated display always shows the logical address of each drive to prevent operating errors.

The 1054 is complete with controller, power supply, chassis, enclosure, cabling and a new Basic software package.

The Micropolis Disc Extended Basic includes a new chain command which allows the user to split large programs into segment. Each segment resides on disc and is called into main memory when required—the technique referred to as virtual storage in the world of mainframe computers. The 1054 costs £1,999.

Further details from Sintrom Electronics Ltd, Arkwright Road, Reading, Berkshire. Telephone, Reading (0734) 85464. ■

# Tune in to Crystal

TORQUAY now has a micro shop, Crystal Electronics, run by Trevor Brownen and his wife. They have the agency for Apple, Nascom and Newbear —whose Bearbags must be one of the neatest pieces of marketing in the field.

As well as supplying kit—and Crystal is the only agency south of Bristol, Brownen believes—there is an in-built consultancy service, since he likes to spend time discussing customers' requirements and advising them on their systems.

Brownen's background is in electronics and computers and the consultancy is free. The shop is at 40, Magdalene Road, and is open from 9am to 5pm. Although you might have to wait a week or so for some items, Brownen is confident he will be able to satisfy most requirements on a cash-and-carry basis shortly. Call Crystal on 0803 – 22699. ■

# Extension for Kim

THE COMMODORE Kim 1 micro has been extended by GR Electronics of Newport with the addition of a video board, additional memory, a pocket terminal and a range of software products.

With the video board a television becomes a VDU with 16 lines of 64 characters. At £150 this item costs £1 more than the Kim 1 itself. The Memory Plus board contains 8K bytes of random access memory and provides for up to 8K bytes erasable programmmable read-only memory. It costs £199.

The GR Electronics pocket terminal, costing £240, allows input of the full ASCII character set. ■

# Home computer coupling

My love and I have decided to buy
a microcomputer personal kit
and a semidetached software house
in which to program it

with bit-sliced ducks
flying up the wall
and Chinese Girl printouts
papering the entrance hall.

Through logic gates of love each night
we enter our world of Basic delights
riding along our I/O fun bus
holding hands with our bugs

and drinking ROM with iced coke
while you solder-iron my best dreams
and I edit your schemes
to upgrade some day
to a much bigger machine
and a man with a DP budget and a
much bigger development team.

And our database is grown pregnant
with shopping lists and babies names,
with shattered hopes and video games
even this poem's now being writ
by the (LOGIN)
micro computer personal kit

(AND NOW HE HAS GONE
HIS LOVE WITH I
IS NO LONGER SHY.
LOGOUT)

**by MALCOLM PELTU, Editor,**
*Computer Weekly*. **Version 2.3**

# Micros or Calculators?

**'Programmable calculators can take over many of the functions of a computer in a small business system and can represent a very cost-effective alternative to the use of time-sharing services'.**

by JOHN GIBBONS
Texas Instruments Ltd.

SO MUCH has been written about the applications of microprocessors that it is often forgotten that many tasks for which they are being proposed are within the scope of a good programmable calculator. In fact, programmable calculators can take over many of the functions of a computer in, for example, a small business system, and can represent a very cost-effective alternative to the use of time-sharing services.

Obviously, there will always be some functions requiring, say, mass storage, where the computer approach will be the only viable one, but the programmable calculator is expanding its range of applications all the time.

## Comparison

The term 'programmable calculator' covers a wide range of products, ranging in price from below the £30 mark to about £1,700. When one is considering calculators which can be considered as alternatives to microcomputer systems, however, one has to look to the more sophisticated machines with program steps running into hundreds or thousands, either on magnetic cards or in pre-programmed modules.

Some idea of the computing power obtainable from a modern hand-held programmable calculator can be gained from the table, which compares a mainframe computer *circa* 1955 with a modern hand-held programmable calculator, the TI-59. It can be seen that, in terms of what the modern machine can do, 'personal computing' is now very much with us.

Professional users, small businesses and home users all can benefit from the flexible programming, computing power, data-storage capability and high reliability of the modern calculator.

It is worth looking in more detail at the various modes of operation available to the user with a programmable calculator such as the TI-59.

First, the magnetic card facility can be used for the storage of user programs and data. Users can build their own personal program libraries, for example. Next, the user can benefit from the availability of what TI calls 'solid state software,' plug-in modules containing a host of pre-written programs stored in solid-state ROM. Among those available are a Master Library containing a selection of commonly-used mathematical, statistical, and financial routines and conversions, as well as games and diagnostic programs. There are also program libraries aimed specifically at marine navigation, surveying, aviation and applied statistics.

This concept has been extended recently with the introduction of custom CROMs (Constant Read-Only Memories) which take the same plug-in form but which are developed specifically where a user has a volume requirement for a particular application.

One enterprising company, Efficient Computer Systems, is taking a customised calculator plus a PC100B printer, adding a CROM developed to its own specification, and re-packaging the complete machine as a device known as ECSTASY. This is designed specifically to help bookmakers in calculations associated with combinations of bets, which can be very time-consuming using conventional methods.

It is important to remember that all the functions can be incorporated in what is a hand-held machine. Even with the addition of a thermal printer—which also permits simple user prompting, plotting and editing routines—the result is still a very compact machine.

## Applicability

Some idea of the applicability of programmable calculators to tasks which normally would be considered the province of a full-scale computing system can be gained from our own experience, as users, within the world-wide Texas Instruments organisation. Almost 10,000 TI-59

(*continued on next page*)

**Table I.**

| | IBM 650 computer | TI-59 calculator |
|---|---|---|
| Components | 2,000 tubes | 166,500 transistor equivalents |
| Power, KVA | 17·7 | 0·00018 |
| Volume, cu.ft. | 270 | 0·017 |
| Weight, lbs. | 5650 | 0·67 |
| Air conditioning | 5 to 10 tons | none |
| Memory capacity<br>Primary<br>Secondary | <br>3,000 bits<br>100,000 bits | <br>7,680 bits<br>40,000 bits |
| Execution time, milliseconds<br>Add<br>Multiply | <br>0·75<br>20·0 | <br>0·070<br>4·0 |
| Price | $200,000 (1955 dollars) | $299.95 (1978 dollars) |

(*continued from previous page*)

programmable calculators are in use within the TI organisation alone and it is estimated that, as a result, company time-sharing costs have been cut by around 40 percent.

In addition, of course, there are considerable savings on executive time, since they have the calculators immediately available. Accuracy is improved, repetitive operations are speeded, and programmability offers the additional benefits of 'what if?' operations and better, quicker decisions.

## Versatility

Typical application areas for this type of programmable calculator include sales desks—for instant pricing and order totalling; buyers—prices, discounts and profit comparisons; estimators—least-cost analysis and estimations; production control—yield calculations and production costs; stock control—usage forecasts, service factors; accountants—costing, planning, depreciation, cash flow and taxation; actuaries—investments, yields and prices; estate agents—investment comparisons and return on assets; general management—planning, decision-making and 'what if?' decisions; insurance—cash analyses, plan evaluations and

> '. . . it fills the gap between the conventional programmable calculator and the personal computer . . .'

presentation; and architects—building design, heating requirements and insulation.

In addition to normal business-type use, applications include TI-58 or TI-59 calculators being employed by hot-air balloonists for navigational and endurance computations in pursuit of a world-record bid, sailors in *The Observer* Round-Britain yacht race for navigational/speed calculations, and as part of a training/evaluation North Polar flight by the RAF.

The next step from this type of calculator is represented by the Texas Instruments SR60A, termed a 'personal-computer/calculator' because it fills the gap between the conventional programmable calculator and the presently-available personal computers, which tend to be tied into a TV-display type of presentation and require the use of a formal programming language.

Costing between £1,200 and £1,700, depending on memory configuration, the SR60A offers a combination of a highly-intelligent calculator, a 'question/answer' prompting display with normal English-language instructions and a quiet built-in thermal printer.

As a result, the machine is ideally suited to small businesses and other applications where it is likely to be used by operators without special training in the use of computers.

One business equipment firm, Betos Systems of Nottingham, has developed its own payroll program for the machine and reckons it can save considerable sums for any firm with more than 50 employees in this role alone—apart from all the other facilities managers can use.

## At the bar

Another company, Abacus of Dublin, has developed a program for calculating bar stocks, while the organisers of the 1978 Milk Race, the Tour of Britain cycle race, found the SR60A invaluable in providing instant printouts of the complex points system for results at each stage of the race.

This brief review of some aspects of the current programmable-calculator scene has highlighted certain aspects. Application areas are still growing and there are signs that the gap between programmable calculators and microprocessor-based personal computers will be closed still further in the coming months. ■

# COMPUTING'S FUN WITH YOUR PET


THERE'S a school in Dorset which saved 40 tons of waste paper to buy its Pet computer.

Last month 500 Pet systems were sold in the U.K., bringing the total to 1,500 here and 3,000 in Europe.

There is no doubt Pet is the fastest-selling "home" computer of its kind in this country and we like it.

We would not recommend you to go to the limit of hunting around for 40 tons of waste paper, like Queen Elizabeth's School, but it is certainly worth hunting for your local Pet dealer to have a look at the system.

The Pet was introduced to be sold to the home computer market, so it would be unfair, we feel, to review it from the view of a computer professional.

We know, however, a number of large organisations who are buying, or considering buying, Pets and, presumably, they are not going to use them as playthings.

According to Commodore which manufacturers the Pet, education and super-calculator applications account for more than half its sales. Business administration and program development accounts for about 20 percent, while the hobbyist is about five percent.

Our Pet (price £695 inc VAT) which we had on loan for a short time, arrived without documentation. We received a booklet entitled *An introduction to your new Pet* which shows how to write simple programs and how to load, but not to save, programs.

In an appendix there is a command and statement summary and notes on cleaning the tape heads. There are, in fact, two versions of the booklet; the first tells how to investigate hardware problems and what the interface specifications are, and the second lists and explains the software error messages.

## Programs without problems

Working from the booklet we wrote and ran a number of simple programs without problems. The display is very clear and steady, although it is a little cluttered with text on adjacent lines. The keyboard is a calculator rather than a typewriter keyboard, although letters are held in the normal QWERTY sequence.

Using a two-finger "hunt and peck" technique the keyboard was not too unpleasant to use but we asked one of our typists to try it. "Oh, my God", she said, "I will have to cut my nails". It is certainly not possible to touch-type if you are used to using a normal typewriter and our typist soon adopted a two-finger technique. One advantage of the keyboard is that program entry normally does not require the use of the shift key.

Using the cassette unit to save and load programs was very easy. Tapes can hold multiple named files and the LOAD command will either load the next file or search for a named file.

We had no instructions on using the SAVE command to dump a program to tape but we had no problem in doing so.

Type SAVE or SAVE "name" and the message "PRESS PLAY AND RECORD" is displayed; press these and the program is written to tape.

Pet responds with READY when the program has been written. The tape we used for saving and loading programs was a W. H. Smith C90, which is reasonably cheap. There is a note in the users' manual which says that you should use "good, low-noise, high-energy tape" and not "three-for-£1 type tapes".

Pet BASIC is a good extended BASIC. Features include—
Integer, floating point and string variables;
A full set of scientific functions;
Logical operators;
Multistatement lines;
String functions, LEFT$, RIGHT$, MID$, CHR$, VAL, STR$;
PEEK, POKE, USR, SYS to interface to memory and machine language subroutines;
Logical operators;
Time-of-day variable.

Variable names are a letter or a letter followed by a letter or a digit. Integer

(*continued on next page*)

(*continued from previous page*)
variables have a % following the name and strings have a $. Floating point variables occupy seven bytes and maintain nine significant digits.

String variables can be up to 255 characters long. The documentation is not very clear about integer variables; they occupy seven bytes (the same as floating point) but are restricted to +32767 to 32768.

For reasons which are not explained in the manual, integers cannot be used in some statements. For example, FOR 1% =1TO20 gives an error message. In addition to simple variables, one- and two-dimensional arrays can be used. Each element of a string array can contain 0 to 255 characters.

## Simplified keying

As well as the normal INPUT and PRINT statements there is a GET statement which gets a single character from the keyboard or tape. Data on tapes can be read or written and named files can be specified so that a tape will be searched for a specific file.

The PEEK and POKE statements allow a specific memory location to be examined or changed. The SYS function allows control to be transferred from BASIC to a machine routine at a specified address.

Program keying is simplified by the fact that the shift key is not required. There is a delete key which deletes the last character input and an insert key which allows characters to be entered in the middle of a line. There is no RENUMBER command, but a re-number program is available.

Running programs can be interrupted with the STOP key and variable values can be entered or displayed using immediate statements. The program can then be continued by typing CONT.

We received the following documentation:

*An Introduction TO YOUR NEW PET.* There are two versions of this, both of which have much the same introductory and statement summary sections. They have different information in the appendices; the first has hardware notes and the second has software error information. We recommend you have both versions although a re-print containing the best of both is said to be coming shortly.

*PET Computer Users' Handbook.* This is badly-produced and difficult to use but essential reading if you are to use the system seriously. Try the section on using the cassette unit for data files if you want to get confused. CBM says the U.S. version is worse.

*PET Users' Club Newsletter.* There were two of these, both containing useful hints, details of errors found by other users, and information on new hardware and software releases.

There are also hardware and software manuals on the MCS 6500 for people who want to delve deeper into the machine. There is also a tutorial tape available from Petsoft, a company selling PET software.

A golden rule for small computer users is "don't buy unless the system can be expanded". Pet has the following hardware ports-

IEEE—488 interface
8-bit user interface
2nd cassette interface—drive available now
Memory expansion interface.

You can buy an adapter (but not from CBM) to enable an RS 232 device (a printer) to be interfaced via the IEEE port. If you plan to store data files on the cassettes you will need a second drive to enable files to be updated (reading on one drive, and writing with the updates to the second drive).

It is a safe bet the PET hardware and software items will be available from a large number of alternative sources. In the U.K., alternative hardware, memory expansion and RS 232 adaptors is available from R. Baily Associates.

Under development at CBM are floppy disc printer, memory expansion and modem.

## Wide variety of software

One of the main attractions of buying a Pet is the software available. Worldwide, the number of independent dealers, users and "publishers" of software for the Pet is astonishing, until one realises that the company expects to sell 35,000 Pets this year. If it can get anywhere near its production targets, it could well be higher.

Commodore publishes software both developed by itself and by its users. Software for the Pet can be obtained from two sources—one is obviously Commodore and its dealers and companies selling Pet, the other is the community of Pet users. Examples of software available from Commodore—by no means a comprehensive list:

Basic—Interactive. Written by two college professors. Teaches you BASIC and how to program. Fifteen chapters, six sample programs and homework assignments. Price £9.

OSERO—A game of skill offering two levels of play against the computer. Price £8.

Pontoon—Board game with a true 52-card pack.

Wrap Trap—Dynamic graphics game in which the player has to trap the computer. Good arcade-quality graphics. Price £8.

Noughts & Crosses—Exactly that. Price £3.

Lunar Lander—Try to put your spaceship on the moon. £8. We had great fun with that.

Rotate—Difficult for non-experts. Similar to little plastic trays with movable letters and letter missing. Price £5.

Biorhythms—Find out when you are up or down. Price £8. We seem to be permanently down.

Commodore also has a number of packages for business applications like management, stock control and inventory programs.

Typical of the software produced by outside firms for the Pet is that produced by PETSoft at 318 Fulham Road, London, SW10. Among the software available from that company is:

Fighter pilot—Rates your skill against shooting down enemy pilots as they streak past the cross hairs of your gunsight. Addictive. Price £7·50.

Alien attack—Dynamic graphics as you

(*continued on next page*)

(*continued from previous page*)

captain your own spaceship through the galaxy thick with meteorites, space pirates and aliens. Price £7·40.

Personality test—Fifty questions fired at you after which the computer tells you what you are really like. £9.

Life expectancy—Do you really want to know? Price £5.

Chicken recipes—First of a series of recipes. £4.

Payroll—For up to 100 named employees, handling bonuses, overtime and deductions. Price £12.

Mortgage—Five programs covering all aspects of mortgages. Calculates mortgage outstanding, terms, and tax relief. Frightening. Price £7.

Assembler—One-pass and two-pass assemblers for Pet, full text editor and documentation. £49.

Basic Renumber—This routine renumbers your Basic program from a given number in given increments. Alters GOSUB and GOTO to match. Auto delete if required. £35.

Binary re-number—Line only remember, re-numbers programs from 10 in increments of 10. These parameters may be changed easily. Program is a binary routine and is transparent to the user and machine. Price £25.

PEEK & POKE—How to use two of the most useful Pet statements. £5.

Plus permutations and combinations, factorials, cubic and quadratic equations, simultateous equations, complex arithmetic, prime factors.

Pet has the most impressive list of dealers of any personal computer available. It includes 35 names distributed around the country, many of them highly-respected businesses in electronics and computer systems. They include Memec, Lasky and Sumlock.

Apart from what this means in terms of locally-available sales and support, most of the outlets see themselves eventually as software publishers. Accordingly they have a vested interest in making packages good—and cheap. It would be hard to over-stress the significance of this dealer network. It means that the machine is receiving a powerful boost towards the kind of volume sales which make future price cuts almost certain.

## CONCLUSIONS

- **Disappointing keyboard.**
- **Poor documentation, except for the simple introduction.**
- **Much more than a simple home computer.**
- **Very good BASIC.**
- **Good display and graphics capability but lower-case restricts the graphics.**
- **Reasonable expansion capability.**
- **Wide range of programs available and under development.**

## TECHNICAL SPECIFICATIONS

Dimensions: 16½″ wide by 18½″ deep. 14″ overall height.
**Weight**: 44lbs

### MEMORY

**Random Access Memory** (user memory): 8K included Expandable to 32K bytes externally.
**Read only Memory** (operating system resident in the computer): 13K bytes
8K BASIC interpreter
4K—Operating system
1K—Diagnostic routine

### VIDEO DISPLAY UNIT

9″ enclosed, black and white, high-resolution CRT
1,000 character display, arranged 40 columns by 25 lines
8 × 8 dot matrix for characters and continuous graphics
Automatic scrolling from bottom of screen
Winking cursor with full motion control
Reverse field on all characters (white on black or black on white)
64 standard ASCII characters; 64 graphic characters

### KEYBOARD

9½″ wide × 3″ deep; 73 keys
All 64 ASCII characters available without shift. Calculator style numeric key pad
All 64 graphics and reverse field characters accessible from keyboard (with shift)
**Screen Control**: Clear and erase
**Editing**: Character insertion and deletion

### CASSETTE STORAGE

Fast Commodore-designed redundant-recording scheme, assuring reliable data recovery
Cassette drive modified by Commodore for much higher reliability of recording and record retention
High noise immunity, error detection, and correction
Uses standard audio cassette tapes
Tape files, named

### OPERATING SYSTEM

Machine language accessibility
File management in operating system
Cursor control, reverse field and graphics under simple BASIC control
Cassette file management from BASIC
Pseudo random number generator

### INPUT/OUTPUT

All other I/O supported through IEEE-488 instrument interface which allows for multiple intelligent peripherals
All I/O automatically managed by operating system software
Single character I/O with GET command
Easy screen line-edit capability
Flexible I/O structure allows for BASIC expansion with intelligent peripherals

### BASIC INTERPRETER

Expanded 8K BASIC; 20% faster than most other 8K BASICS
Upward expansion from current popular BASIC language
Strings, integers and multiple dimension arrays
10 significant digits; floating point numbers
Direct memory access through PEEK and POKE commands

### PRICES

Pet 2001 Personal Computer: £695 inc VAT
Pet cassette deck: £59 inc VAT
Pet 2020 printer: £495 inc VAT
Typical software prices in text.

Pet users' handbook: £5
Pet introductory booklet: £1
Pet users' club manual: £10.

## AUTHORISED COMMODORE DEALERS

### LONDON & HOME COUNTIES

C.S.S. Systems Ltd.,
502 Kingsland Road,
London, E84AE
Te . 01-254-9293

Sumlock Bondain Ltd.,
Sumlock Anita House,
15 Clerkenwell Close,
London, EC1
Tel. 01-253-2447

Petalect Ltd.,
33/35 Portugal Road,
Woking,
Surrey.
04862-69032

Central Southern Equipment,
12 Wokingham Road,
Reading, RG6 1JG
Tel. 0734-61492

### NORTH WEST & NORTH WALES

Cytek (U.K.) Ltd.,
17 Exchange Hall,
Corn Exchange Building,
Manchester M4 3EY.
Tel. 061-832-7604

Rockcliff Brothers Ltd.,
Long Lane,
Aintree,
Liverpool 19
Tel. 051-521-5830

Cortex Computers Ltd.,
25-35 Edge Lane,
Liverpool 7.
Tel. 051-263-5783

D.A.M.S. (Office Eqpt.) Ltd,
30-36 Dale Street,
Liverpool 2.
Tel. 051-227-3301

Sumlock Electronic Services
(Manchester) Ltd.,
196 Deansgate,
Manchester,
M3 3WE
Tel. 061-228-3507

Automated Business Eqpt. Ltd.,
Mersey House,
Heaton Mersey Industrial Estate,
Battersea Road,
Heaton Mersey,
Stockport,
Cheshire SK4 3EA
Tel. 061-432-4299

### LONDON RETAIL

The Byte Shop,
426/428 Cranbrook Road,
Gants Hill,
Ilford,
Essex.
Tel. 01-554-2177

Laskys
42 Tottenham Court Road,
London, W1.
Tel. 01-637-2232

(*continued on next page*)

*(continued from previous page)*

Euro-Calc Ltd.,
224 Tottenham Court Road,
London, W1.
Tel. 01-636-8161

### NORTH EAST

Currie & Maughan Calculator Serv.
5 Bewick Garth,
Mickley,
Stocksfield,
Northumberland,
NE43 7AV.
Tel. 06615-2194

Intex Datalogic Ltd.,
Eaglescliffe Industrial Estate,
Eaglescliffe,
Cleveland.
TS16 OPN
Tel. 0642-781193

### YORKSHIRE & HUMBERSIDE

A.T.A. Management Consultants Ltd.,
12 Welholme Avenue,
Grimsby,
South Humberside. Tel. 58281

Microprocessor Services Ltd.,
1 Nunburnholme Avenue,
North Ferreby,
North Humberside.
Tel. 58281

Kingston Computers,
84 High Street,
Bridlington,
E. Yorks.
Tel. 0262-72837

Holdene Ltd.,
10 Blenheim Terrace,
Woodhouse Lane,
Leeds 2.
Tel. 0532-459459

### MIDLANDS

Memec Systems Ltd,
Thame Park Industrial Estate,
Thame,
Oxon OX9 3RS.
Tel. 384421-3149

Taylor-Wilson Systems Ltd,
Oakfield House,
Station Road,
Dorridge,
Solihull B93 8HQ
Tel. 05645-6192

Becos Systems Ltd,
Bennett House,
155 Mansfield Road,
Nottingham NG1 3FR
Tel. 0602-48108

Davidson-Richards Ltd,
14 Duffield Road,
Derby.
Tel. 0332-266803

Arden Data Processing,
Municipal Buildings,
Charles Street,
Leicester.
Tel. 0533-22255

### SCOTLAND

Robox Office Eqpt.,
84 Townhead,
Kirkintillock,
Glasgow.
041-776-4388

### CHANNEL ISLANDS

Commercial Business Systems Ltd.,
Les Forgettes,
Clos au Conte,
Castel,
Guernsey.
Tel. 0481-55574

### WEST COUNTRY & SOUTH WALES

Computabits Ltd.,
41 Vincent Street,
Yeovil,
Somerset.
Tel. 0935-26522

C.S.S. Systems Ltd.,
351 Fishponds Road,
Bristol BSS 6RB
Tel. 651449

G.R. Electronics Ltd.,
80 Church Road,
Newport,
Gwent. NPT 7EH
Tel. 0633-67426

Sumlock Tabdown Ltd.,
Stirling House,
Fairfax Street,
Bristol BS1 8HX
Tel. 0272-26683

### NORTHERN IRELAND

Medical & Scientific Computer Serv.,
Altona Road,
Lisburn,
Co. Antrim,
N. Ireland.
Tel. 02382-77533

### SOUTHERN IRELAND

Software Development Serv. Ltd.,
84 Northumberland Road,
Ballsbridge,
Dublin 4.
Ireland.
Tel. 683171

### Specialised Technical & Control Dealers

Fairhurst Instruments,
Dean Court,
Woodford Road,
Wilmslow SK9 2LT
Cheshire.
Tel. 09964-25694

Datron Sales Ltd.,
Penmark House,
Woodbridge Meadows,
Guildford,
Surrey GU1 1BA
Tel. 0483-37337

More dealers may have been added to the network since we went to press. If you are one, please let us know and we will print your name in the next edition. ■

# WIN A COMPUTER COMPETITION

To mark the launch of Practical Computing we are giving away an Apple II computer.

## THE MAIN PRIZE

The winner of the competition will receive a complete boxed system including microprocessor, keyboard, power supply, high level language interpreter (in firmware), plus 4K bytes of read and write memory.

You will also get game paddles and a tape cassette machine. All you will then have to do is connect the system into a standard colour television and it is ready to use. It comes in a smart moulded plastic case which you simply plug into your mains supply.

## PLUS

In addition to the computer we shall be giving away TEN prizes of £25 to runners up in the competition.

## HOW TO ENTER

We are looking for ideas on how you would use the Apple II computer which was reviewed in the July/August issue of Practical Computing.

It could be a game you have invented. Or it could be a business application which you have developed. It could even be a project which you are working on in school or in college.

To enter all you have to do is write a description of not more than 3,000 words on your application. We require a description of the application or project plus your solution on how you would develop the application.

---

No more than one entry may be submitted per person. Each entry must be accompanied by an official entry voucher.

We shall, however, accept a project entry from a single education establishment.

Each entry must not exceed 3,000 words and must be typed, double-spaced. Handwritten manuscript will be accepted provided it is legible.

Closing date for the competition is October 15 1978 and entries received after that date will not be considered.

The competition will be judged by the editorial staff of Practical Computing. The Judges' decision will be final. No correspondence will be entered into. All entries become the copyright of *Practical Computing* and entries will only be returned if a stamped addressed envelope is provided.

Employees and relations of ECC, Practical Computing and WHICH COMPUTER? are, of course, excluded from entering. Winners will be notified by post, and winning entries will be published in future editions of *Practical Computing*.

---

**Please detach and enclose with your entry**

I accept the entry rules stated and agree to abide by the judges' decision.

Name _____

Company (if applicable) _____

Address _____

_____

Tel No: _____ Date _____

Signed _____

**Now post please to:**
**Practical Computing**
**2 Duncan Terrace**
**London N.1.**

# HOW COMPUTER LOOKS TO GROW

**IT'S NEARER to Mayfair than Soho; so Computer Workshop is, as the estate agents would say, nicely situated. So nicely situated that the initial impression is really quite distressing by comparison with other computer shops —has one arrived at the wrong address? You step into the cream-painted and mirror-panelled entrance lobby, casting around for some sign to confirm that Computer Workshop is indeed here. All one knows is that this is 38 Dover Street, London, W1, and that is where Computer Workshop is said to be found.**

One wades across a few acres of carpeting and seizes a passing pin-striped stranger with one's steely gaze. Said stranger pleasantly admits that he's never heard of it but observes that something called South West Technical Products is on the first floor.

On the first floor one finds it—South West Technical Products, that is. Then, in smaller letters, Computer Workshop. The keen youth there reluctantly leaves the VDU and asks if he can help. One considers that there isn't really much chance of that but asks for John Burnett, the managing director. At last one finds what one was looking for: and in some respects it resembles entropy.

In a small office to one end of the main showroom he is on the telephone. Around him are two other people similarly engaged. Around and beneath him are cardboard cartons of things without names and printed circuit boards with the name M6800 stamped on them. Minifloppy discs holding data and coffee cups abound. See what one means?

"I used to be a computer man selling bureau services," proclaims Burnett. "But I'd always wanted to start my own business and for years I'd been trying to sort out how to do it. To be a consulatant or maybe to sell bureau time, but to be independent while doing it. Looking for a low-cost data-collection device I came across a low-cost VDU kit. It sold for £215 without a case. You had to provide your own TV or monitor but there was nothing in the country like it."

It was made by a company called South West Technical Products and as a direct result of the encounter in May, 1976, Computer Workshop was formed. It was the first computer firm of its kind in Europe.

"I used to go around showing the VDU kit to people, walking into expensive computer showrooms and hooking it into the cables there. It never failed to work, though. The idea evolved from selling a service to emulating the burgeoning microcomputer shops in the States—although, in the end, we didn't exactly do that. Instead we became a specialist supplying South West Technical Products goods."

In May, 1976 Computer Workshop turned over £2,000-worth of the VDU kits and was based in Fulham. The address isn't all that has changed.

## Beating barriers

"The partner with whom I started is always starting new businesses—and then selling them. His next success was selling camping holidays in the Mediterranean. He likes to put his money into new ventures and then finds something else to do.

"Early this year we had become the biggest single SWTP customer in this country so started talking about setting up a factory here to beat freight and duty barriers.

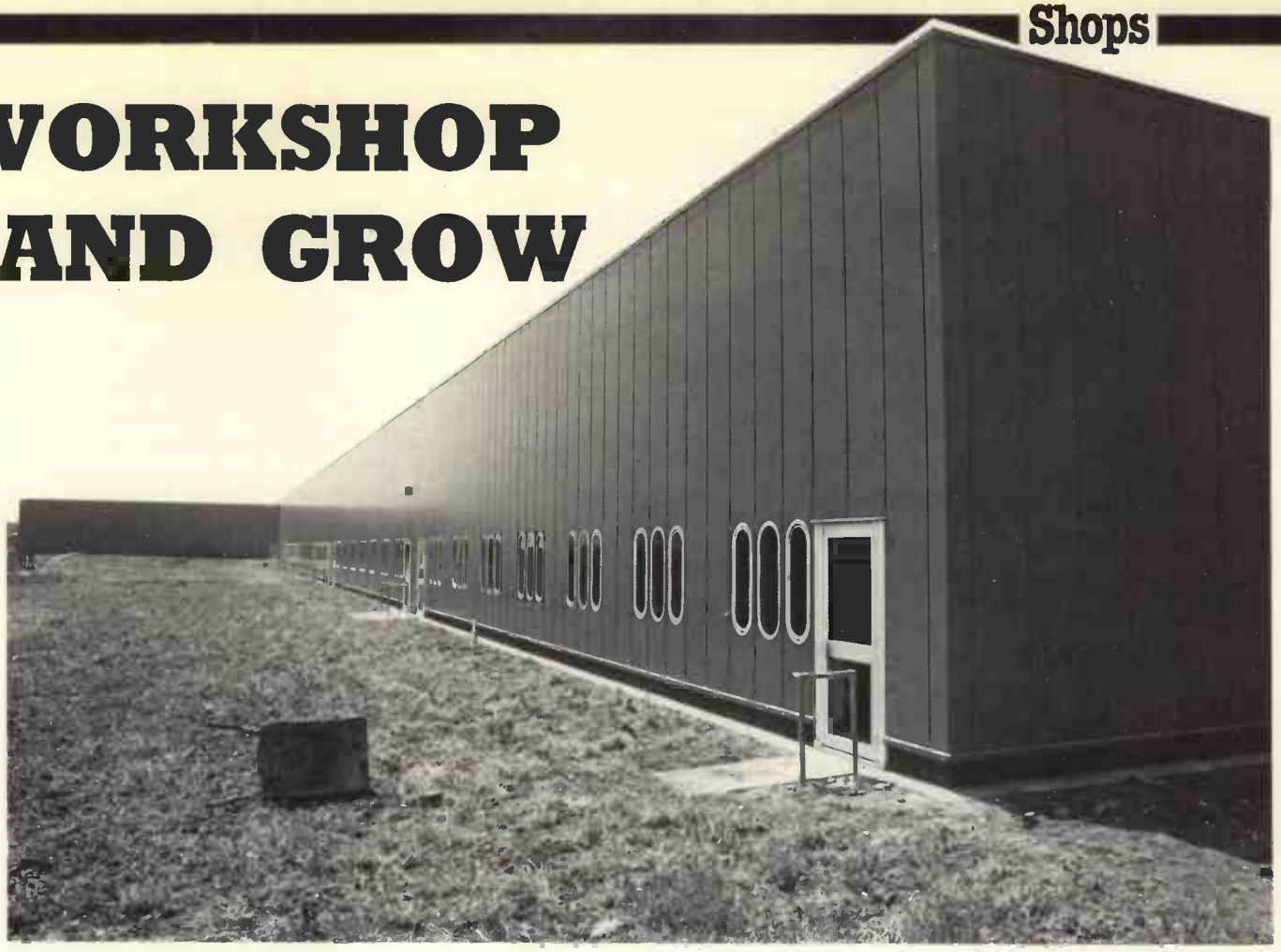
"During the negotiations SWTP in the U.S. financed 50 percent of the U.K. operations and 50 percent was financed by Computer Workshop. In effect, Computer Workshop was then taken over by South West Technical Products Computer Ltd, the holding company for the new factory in Peterborough and the shop."

At that point Burnett's original partner sold out, leaving him with a sizeable chunk of the action for himself.

Today, Computer Workshop is the retailing end of South West Technical Products; as you would expect, it sells only SWTP kit. "We decided that to support a large number of different systems would weaken our support capability and we

(*continued on next page*)

# WORKSHOP —AND GROW



(*continued from previous page*)

would never get to grips with every system. We felt it was the right decision to concentrate on SWTP because it was the lowest-cost manufacturer in the equipment area and one of the largest".

Judging by the new premises and the current turnover of around £75,000 per month, he may be correct, in business terms. But is it a shop you would want to visit? To whom, for instance, does it sell all its kit?

"We're still selling right across the field," says Burnett, with the exception of, say, the hobbyist. Hobbyists in this country tend to build their own systems from chips and other components which they buy from large distributors. This is not what we sell: we offer the complete thing. Hobbyists *per se* represent such a small part of our market that one can almost discount them. Someone comes in and says they're going to buy them for the home—but then they come back later and say they're using it in the office."

## Not a toy

If you want to know what he is selling which makes him so unsuitable for hobbyists you can send for a catalogue or read any of his advertisements. Neither of those two pieces of reading matter is very lengthy. That is because it is all essentially about one system which is expandable.

The SWTP processor is of course an M6800-based device which, with twin minifloppies, VDU and 16K of memory will cost around £1,925. That is the price fully assembled with a disc-based operating system and BASIC compiler. It is fun to play with but it is definitely no toy.

As an example, Burnett attaches a bi-directional daisywheel printer (made by Rioch and about £1,800 more) and runs off a few impressively-personalised letters on it. One of the packages is word processing—"but we don't really sell it on that," he adds casually.

There is no way of looking at a cheque for £2,000 without thinking it a lot of money even if you get a lot for it. So how about something cheaper? Remove the minifloppies and replace them with an ordinary audio cassette recorder—that saves you about £760. Buy it in kit form and assemble it yourself—that knocks off about 20 percent. Use less than 16KB RAM and you are definitely well below £1,000.

"If you compare this to something like the Pet or the Tandy TRS-80 which are completely integrated units, we can't really compete. To build Pet-like systems from our equipment would cost more. But ours is more flexible and you can expand it as much as you like. There is no difficulty at all; you can plug in any kind of terminal and add any cards you want."

It is, in short, a system you cannot easily outgrow. Which must be fine for customers, included among whom are ICI, Rolls Royce, Dunlop, Hawker Siddeley, GEC, Marconi and ITT.

Computer workshop also has systems installed in "many Ministry of Defence establishments, but they never tell us what they use it for." Possibly they play Star Wars on it; maybe something more prosaic, like Real Wars.

"We're also getting a new type of customer—first-time users and people who are replacing their visual record computers. And we are suddenly beginning to attract smaller companies."

This is something Burnett is pleased about. Whereas most firms selling computers want a few more customers like ICI, Burnett thinks small.

## Plenty of ideas

"I get people coming in who are interested in writing expensive software. I want to stamp on that sort of thing—I'm writing ledger programs I want to sell for £50. Fundamentally I believe that software is overpriced but it depends on the number of copies you expect to sell. If you have an integrated ledger at £50 you will have a hundred sales for it almost as a matter of course.

"When it comes to business it is clear that Burnett is running a lot of new ideas and not running a charity, but in no way is he trying to keep all of the gravy to himself.

"We just cannot support more than a tiny fraction of the customers out there. So we try to attract programmers—not just coders but people who know business —to install systems and in some cases write applications software.

"As packages become available they will probably make their money by selling

(*continued on next page*)

(*continued from previous page*)
hardware we will let them have at a discount. We foresee one-man turnkey systems houses.

"I'm beginning to develop a network of people on this activity—the one thing that I'm trying to instill into them all is that very shortly they will be making their margins from installing systems and not from writing software,"

Some people, it seems, engulf markets slowly and insidiously: Burnett appears to be doing it quickly and insidiously. Consider the established computer industry. It has a 'bible'—the *Computer Users' Year Book*—which lists every computer installation in the country; or, at least, it used to do. It is already missing a few hundred installations, for Burnett's activities seem to have gone unnoticed by the CUYB.

## Changing face

Such a standard reference source is the CUYB that salesmen traditionally look in it to see who has what and by implication who has no computer. Burnett and his ilk are changing that. Once the mainframe computer manufacturers knew the exact state of the market. Now they do not and the situation will worsen from their point of view as one-man systems outfits start installing £2,000 systems for smaller and smaller outfits.

"The big companies are stuck with marketing strategies and structures which cannot handle unit sales at such low cost. They have dug their own graves and one way they have done it is by convincing people that the user cannot go near the computer. Operators are trained on a particular machine; programmers knowing only one language. With BASIC there is no loyalty to any machine; it is easy to learn; the users will no longer have to rely on specialist programmers. The ignorance in the established computer industry of these things is mind-blowing".

At this point one quickly begins to wonder whether the sky might not be too low a limit to set on Computer Workshop aims. What is coming next? Will the machines start talking? ?

"Yes. This machine will have a plug-in card for £350 to give a full vocabulary of voice output using phonetic input—it is easier than voice input, but you never know what is coming next.

"It is all a bit like doing a jigsaw where you're told where every bit goes. The only problem is if they don't work immediately, you might need help. I have built all of these kits myself and I know nothing about electronics."

The kind of problem which occurs if you decide to do it yourself would be a perfectly-assembled kit but with a solder bridge—a piece of solder where it shouldn't be causing current to flow where it shouldn't flow. If you are just a little used to assembling circuits on printed circuit boards you will probably be able to spot it. Failing that you can always take it back to Computer Workshop.

Burnett confirms our view that the VDU kit is "an absolute sod" to build, though. In fact, if help and service is on your mind you can take out a maintenance contract on your system. Computer Field Maintenance, a well-known maintenance firm in the established industry, will give you a choice of maintenance contracts, right to guaranteeing an engineer on-site within eight hours of a malfunction.

## Training courses

Perhaps your worries do not concern the hardware so much as the concepts or the programming. Well, Computer Workshop is using part of its London showroom to run training courses. They cover BASIC for business users; word processing; programs for microprocessor control systems (for logic designers); and the use of micros in education.

It sounds fine and surely all that is needed is for the price to come down a little so everyone can have one in at home? It seems not.

"Between £2,000 and £5,000 we are in a price range where businesses can justify the use of systems for a much lower volume of data or even a much more

(*continued on next page*)

(continued from previous page)
trivial application. Prices are failing but as we introduce new and more sophisticated items we will stay in that price range and the cost of existing systems will fall accordingly."

Burnett is after business and he knows how much business has to spend. He also knows exactly from where his business is coming.

"How many people," he asks rhetorically, "are using computers at the moment compared to those who aren't using computers? That's where the market is."

If you do not have a computer, Computer Workshop is thinking of you.

## Only a start

Meanwhile, just off the Peterborough by-pass, a low and featureless building in a sickly shade of green sits in the raw landscape of Peterborough's new industrial estate. That is the new factory of Southwest Technical Products (Computing) Co. Ltd., the first manufacturing operation to be set up in this country by a U.S. micro manufacturer.

In fact, the Americans own half the company, the balance being held by Burnett and his partner, Hugh Woodsend.

In July, the 10,000 sq. ft. unit was sparsely occupied. It had a staff of 26, of whom 14 were employed on production. The resulting output is around 150 units a month—processors, terminals and disc drives—equivalent to about 50 complete systems.

This, however, is only the start. Burnett is prepared to meet a rapid increase in demand and intends to train up to 10 production staff a week—the forecast is for 40 production staff and 12 engineers by March, a year after the factory opening.

The plant next door is also earmarked for takeover and with other production tweaking this will move its production capacity up to 400 complete systems a month.

The ability to respond rapidly to increased demand is, of course, one of the main reasons for manufacturing in this country: "Where micros are concerned, if you do not have it on the shelf, you can kiss the customer good-bye."

Delays in shipments have been a big Computer Workshop problem in the past but Burnett reports that back orders are down to around £40,000, compared to £150,000 last year.

"We're beginning to get a high-speed turnround on repairs, too," he adds. "It was chronic at one time." There are now engineers working full-time on repairs in London and Peterborough.

The interesting and ever-growing library of software available includes a text editor and a system for use by barristers. The invoicing system and production scheduling program written for SWTPC use are likely to be developed as packages and a freight-handling package is also in the pipeline.

## Cosmetic aid

Given this development of software skills, SWTPC should be able to offer the business user an attractive package. Even the cosmetic deficiencies of the casings have been noted and we are assured that their spartan functionalism will soon be a thing of the past.

All in all, SWTPC optimism looks like being well-founded. The product is already well-liked in many quarters and has an established customer base. Product development seems to be proceeding at a reasonable pace.

The presence of a manufacturing base in the U.K. will certainly be an asset as competition strengthens—at least until other U.S. manufacturers follow the lead. ■


# The Micro Master

**In deepest Cornwall, a 53-year-old schoolmaster is waging a campaign to spread the word of the microprocessor revolution to schools in the county. The man, a self-confessed "bit of a nut-case", is Desmond Old, head of the department of electronic engineering and computing at Cornwall Technical College and he is in the process of building his sixth microprocessor system for educational purposes.**

Much of his research is done in his attic at home where he finds it "much more convenient. I can leave things when they're half done and I don't have to clear up afterwards".

He first became interested in computers in a "serious" way in the early 60s when Sir Walter Puckey, a Cornishman on the board of ICL, decided he would like to do something for Cornish children.

He tried partially to finance some form of computing facility for use at this college and schools", says Old, "The immediate result was that a number of us on the computing and technical side attended an ICL course in London to learn how to write programs in Fortran".

The college came involved in computer education in a big way in 1965, using time on the county treasurer's ICL 1902 computer. It was then decided to branch out into the schools and teach computing in a general way. College lecturers visited the local grammar schools and gave lectures in computer appreciation and, in particular, Fortran programming.

'We took away the programs from the scholars, on coding sheets, and took them back to college", explained Old. "We had a dp clerk who punched them and they were sent to the county treasurer's installation over a very early and crude form of data link with an old ICT 70 003."

## Blocked by the inspectorate

The situation was clearly not satisfactory and in the early 1970s the college tried to obtain its own mainframe but was blocked by the inspectorate, which "regarded us as not really big enough users in the educational world to warrant having a machine of our own, whether it be stand-alone or some terminal. As a result we were encouraged to use a London bureau in no way associated with the local government machine."

That, too, was not an ideal situation and when Old "got wind of the micro revolution in the U.S., it opened dramatically what was to us a very closed door in financial terms."

The first system the college built was the Motorola D1 kit, "which got our feet wet in the business", says Old. "We then built the cheapest one which could run Basic in kit form, the MITS 680b, which we purchased almost two years ago. That worked extremely well, and still does, but it is a design which MITS has decided not to proceed with and, as a result, much of the extension of that system for which we were hoping didn't materialise.

"We then built the MITS 8800b which, complete with a pair of floppy disc units and one or two other miscellaneous facilities, is now the main system in the college."

A fair amount of software is written in the college both in machine language and in high-level languages such as Fortran and principally in Basic. They also run an in-between language used in education, called Cesil.

That started with a form of CES. "IBM did a version of it for us to use on the 4360 system which was Cecil, to get over any copyright business there might have been," Old said. "We have now implemented it ourself on micro and call it Secil." That is written in Basic for the

*(continued on next page)*

(*continued from previous page*)
college microprocessors, the 6800 and the 8080.

The main system applications are running programs written in either Basic or Secil. "We simply try to get people to gain confidence in program writing", says Old.

"Where possible, we get them to enter and run the programs themselves. Where physically impossible, we deal with them on a batch basis. This is not 100 percent satisfactory but providing the teaching in the classroom situation is well thought out and well done, it's not that unsatisfactory.

## Servicing noise measurement

Some help is also given to other groups in the college where computerisation is applicable, such as the noise laboratory. There thought is being given to a system for a library for maintaining a file which can be accessed readily for servicing noise measurement and abatement. They also help people with power systems and access to data and they help to set up their systems.

Old estimates that his work in the college has saved a considerable sum. In the last full year of using the bureau service, they spent £6,500 on time-sharing and the rental of terminal equipment, and about £2,000 in telephone charges. The bulk of that will be saved in future years and the college is now able to use the extra money to enter new and more advanced fields.

He estimates that he has saved £1,000 to £2,000 by using the three working systems in the college—the MITS 680b, 8800b and the SWTP 68, which they do not use much for "serious" data processing.

Old believes the widespread use of micros for small businesses is not far away. "The micro system we have is very readily capable of performing in any dp situation in which a small-to medium-size business finds itself, whether it be manufacturing, medical practice or an office, assuming that the work could have been done by a bureau", he says.

There are cost advantages and security would play a large part in deciding to buy a micro. "If a doctor can lock away his disc files at night, then he will feel far happier that he has 100 percent security." "In the States many medical people with a streak of do-it yourself have written a tremendous amount of software to keep patient records. If it is happened there, it is bound to happen here".

Another area where he thinks micros can be used is by small-to medium-sized firms for payroll. "There is not much software about at the moment", says Old, "but I think if people are prepared to write their own it will be a very attractive proposition".

## Coin analysis program

The college is writing a modular program for payroll based on the British fiscal method and one based on PAYE is already running.

The approach taken by Old and his colleagues is that of partial computerisation. If the computer breaks down, then people can go on doing the work manually.

"We feel that many of the small business people will gain much more confidence in adopting an increased payroll activity if they see partial computerisation agreeing with what they can check."

They also have a program which will deal with graduated pension reductions. For demonstration purposes there is a coin analysis program with which they can work out the various denominations of currency as required for each pay packet. They are now constructing the payroll file, structuring the employee information and the correct number of hours he works in the day or the week.

Most of the projects originated in the attic of Old's home in Camborne. His interest in electronics began with radio and he still works in that sphere.

"I've always been very keen on building

(*continued on next page*)

(*continued from previous page*)
anything just for the sheer hell of it", he says. "It was through the amateur radio journals coming from the U.S. in the 1973–74 period that my attention was drawn to firms like MITS and the people bringing micros to the end-user in a very economic way.

## Attic place for innovation

The Old attic has been the scene of most of the innovations in use in the college. Old is wholeheartedly backed by his wife—affectionately known as "Miss"—and his two sons, graduate electronic engineers.

"We are continually exchanging ideas and I am gradually getting a band of my own staff in the college enthusiastic about this; but it's a question of how many hours there are in the day and most of my staff find there are only 24. With the co-operation of my wife, I have managed to find 36 and those extra 12 hours are taken up with the microprocessing business".

Old has a MITS 8800b for his personal use and has built a hybrid from what he considers to be the most cost-effective components available in the business. He imported them from San Francisco and they consist of a Morrow's front panel and another board, "plus a load of memory from a firm called Thinker Toys. For something like £500 I have a very powerful machine with an 8080 processor, 32K memory and S100 bus."

He is all for bringing computers to the people, especially in education. "For many years computers have been in the hands of the wrong people", he says. "They are now coming back to the engineers who conceived them originally. "There was a time when they were so large and so expensive that only management types could persuade the boards of companies to buy them. As a result, there was an elite class looking after computers who tried, rather successfully, to maintain a cloak of mystique about what was going on. Computers are now coming down to the engineer. This is where the micro will really take off. They will build a bridge between the real world and the computer world.

## Fantastic future for micros

"I think in education the microprocessor has a fantastic future. One can make learning such fun that one can now bring about what was tried in the early '60s with the old electromechanical machines—program arrangement. It might be mundane things or conventional education.

"If someone only had the courage to employ 10, 20 or 100 unemployed graduate teachers and set up a software house which is really well-conceived and well-organised, there is plenty of money to be made. If one is looking for steady employment, there is a fantastic field for producing packages for physics or any of the O and A level subjects where a person can go to a computer centre and get involved with what they want to talk about.

## No plans to commercialise

Old has no plans to market his own system commercially. "Money doesn't interest me as long as I have enough", he says. "I don't like working to the deadlines of the commercial world. I'm a bit of an academic, I suppose. I do what interests me. My two sons may wish, when the microcomputer business settles a little, to go out into the wide world of business and they might tempt me to act as some form of partner."

He sees himself retiring within the next year or so, even though he is still only 53. "It's not because I have nothing to do," he says. "It simply means that I can get in contact with the other 24 hours in the day, which I can't at present." ■

# MCS-86™ is here today. era in microcomputers.

For easy expansion, add our 8288 Bus Controller, which generates Multibus™ timing signals. And, using our 8286/8287 Octal Transceivers, MCS-86 interfaces with larger, buffered systems.

**Get started with software now using our Intellec® development system and PL/M 86,** an expanded version of our popular PL/M high level language. And, with the ASM 86 assembler, you can write assembly language programs for your MCS-86 designs and translate your existing 8080 and 8085 software to run on 8086. Both PL/M 86 and ASM 86 operate under ISIS-II, Intellec's advanced operating system.

MCS-86 components are available today. Order your MCS-86 Prototype Kit from your local Intel distributor. The kit includes all the essential components to begin your evaluation of this new era in microcomputers. Or, for more information, write:

Intel Corporation (UK) Ltd
Broadfield House
4 Between Towns Road
Oxford OX4 3NB

Telephone
Oxford (0865) 771431

Telex 837203


**intel® delivers.**

37

# Bargain VDUs

## by Martin Collins

**Until recently there were few computer terminals, apart from the ubiquitous Teletype, which cost less than £1,000; and most VDUs even today cost considerably more than that. While one can still be charged more than £1,500 by some manufacturers, however, it is possible to buy a VDU for less than £500. This article compares the Computer Workshop CT-64 and the ACT-1 from Strumech Engineering.**


The CT-64

The ACT-1

THE CT-64 was designed by South West Technical Products and is being manufacturerd by a subsidiary in Peterborough. The terminal is sold either in kit form or fully assembled and can be used with a monitor—Computer Workshop supplies a matching monitor—or via a VHF modulator and a standard television.

Computer Workshop, like most small system suppliers, is trying to sell assembled systems rather than kits. The people there say that the CT-64 kit is difficult to assemble; it takes three times as long as the SWTP processor kit based on the M6800, so you should probably buy it in kit form only if you have had a lot of experience in assembling systems.

The display consists of 16 lines of 64 (or 32) characters. The full 128-code ASCII character set can be generated and as characters are formed by a 9 × 5 dot matric, lower-case letters have true descenders. The control characters for cursor movement, home, page or scroll and the like, can be defined by the user. Inverse video is supplied for all or part of the screen. As an option the terminal can have two pages of memory, each holding 16 lines of 64 characters, which can be selected on an either/or basis.

The terminal can be operated in scroll or page mode. In scroll mode, as a new line is displayed or entered the display rolls up by one line and the top line is lost; in page mode, information is entered or displayed from the top to the bottom of the screen. These two modes of operation, together with the fact that all normal cursor movements are allowed, mean that any required screen formatting can be performed.

### Normal layout

The keyboard follows the normal QWERTY layout with additional keys for escape, backspace, echo, and on-line/off-line. Additionally there is one user-definable key available to anyone assembling the kit. If a key is held down the character repeats. The keyboard has a rather imprecise feel to it compared to more expensive terminals.

The terminal has a standard RS 232 interface and can be configured to any required combination of parity and stop bits. The assembled units have an external baud rate switch which enables the terminal to be operated at 110, 150, 300, 600, and 1,200 baud. Computer Workshop is also working on a modification to enable the terminal to operate at 9,600 baud.

Kits carry a 90-day warranty, subject to their being sensibly assembled, while the factory-built units have a full 12-month warranty. Maintenance can be arranged through an independent company (CFM) or faulty units can be returned to the factory for repair.

At present there is a six-week lead time on delivery of assembled units but once full production is under way in Peter-

*(continued on next page)*

(*continued from previous page*)
borough, Computer Workshop is planning to be able to supply from stock.

The ACT-1 offers rather fewer features than the CT-64. The terminal is imported from the U.S. and is available only as an assembled unit.

The display consists of 16 lines of 64 characters and upper- and lower-case characters can be displayed. As with the CT-64, output can be to a standard video monitor or, with an optional VHF modulator, to a standard TV set. Characters are formed by a 9 × 7 dot matrix and lower-case characters have full descenders. There is no page mode but the screen can be scrolled up or down. Left, right and home are the only cursor movements allowed; the screen formatting capability, therefore, is more limited than on the CT-64.

The keyboard is a normal QWERTY unit with additional keys for escape, tab, backspace, break, rub-out and clear. The unit appears to be more robust than the CT-64, with a metal rather than a plastic case and a more positive action on the keyboard. There is an internal switch for the baud rate and the terminal can run at 110, 300, 600, 1,200, 2,400, 9,600 and 19,200 baud.

The terminal has a standard RS232 or 20mA current loop interface and can be configured to any required combination of parity and stop bits.

The terminal has a 90-day warranty and Strumech will arrange maintenance for terminals sold as part of a complete system. Faulty units can be returned to Strumech for repair. The company has only just started to supply the ACT-1 and at present is quoting three weeks for delivery, but intends in time to meet orders from stock. Strumech will also supply a monitor if required. ■

## Conclusions

- **Compared to the costs of two or three years ago, both terminals offer good value for money.**
- **If you are not experienced at assembling kits and you do not want the extra features by the CT-64, the ACT-1 scores because it is cheaper. It certainly feels more robust than the CT-64 as well. On the other hand, the CT-64 offers plenty of features for a low price.**
- **If you plan to spend a lot of time using the VDU, it is definitely worthwhile investing in a monitor; and the monitor for the CT-64 is very clear. An alternative to buying a monitor from either Computer Workshop or Strumech would be to shop around for a second-hand or re-conditioned unit, which should cost about £60-£80,**

| SUMMARY | CT-64 | ACT-1 |
|---|---|---|
| Lines per display | 16 | 16 |
| Characters per line | 64/32 | 62 |
| Character set | 128 ASCII | 96 ASCII |
| Character generation | 9 × 7 | 9 × 7 |
| Cursor control | Home, up, down, left, right | Home, left, right |
| Scroll mode | Yes | Yes (up and down) |
| Page mode | Yes | No |
| Two pages | Yes | No |
| Reverse screen | Yes | No |
| Auto repeat | Yes | No |
| Data rates | 110–1200 | 110–19,200 |
| Interface | RS 232 | RS232/current loop |
| **PRICES** | | |
| Kit | £230 | — |
| Assembled | £315–£347 | £260 |
| VHF-Modulator | £4·50 | £10 |
| Monitor | £140 | £125 |

# Practical Computing


## The new magazine for the computer enthusiast.

Each month we shall be looking at how computers can be used in the home, school and as small business systems.

**In The Home**

You don't have to be a programmer or computer professional to read Practical Computing. It's the magazine that tells you how to get started, who to buy from, and what you can do with computers in the home.

There are hundreds of applications that you can run on your own computer – and it doesn't cost the earth.

Each month we give a practical review of a home computer, detailing what it can do and how much it costs.

We look at peripherals. You can build your own or buy them off-the-shelf. We also look at what software is available and give you a step-by-step guide to programming your own system.

There's games to get you started. We show you how to develop TV games, space war games, puzzles, sports simulation, music and speech synthesis and how to put your dream games into reality.

Then there's a host of other applications you can develop like automating your home, sorting out your finances, mathematics, and even how to speak to your computer.

**In Business**

Whether you are running a small business or you want to develop business applications, Practical Computing shows you how to get started and what you can do.

Word processing, accountancy, mailing lists, small business systems, stockbroking, stock control, typesetting, simulations, cash flow, payroll, surveys, bookkeeping, graphics, and real-time control are just some of the applications you could develop.

We show you how you can make money out of developing and selling your computer applications. And if you are a small business, we tell you about the packages that have been developed and the costs of buying your own system.

Each month there are articles describing business applications in detail. We review the equipment you need and how much it costs.

You can save money too! Our evaluations cover microcomputers, terminals, peripherals, software packages, periodicals, booklets and suppliers of equipment. We help you choose the computer that's best for you – before you spend money on one that isn't.

**And in Education**

We examine how computers can be used in education. Develop timetables, learn about programming, classroom computer projects and games, mathematics and business studies are some of the applications described.

We also look at how other schools, colleges and universities are using computers along with actual project descriptions and case studies.

Practical Computing is fun. We've got style, colour and we're packed with lots of ideas to help you get the most out of your computer. Try us now.

## Available from all leading newsagents

**U.K. DISTRIBUTORS:**

Airamco, Ltd
30 Witches Linn
Androssan, Ayrshire, Scotland
0 294-65540

Newbear Computing Store
Bone Lane
Newbury, Berks. U.K.
0 635-46898

### MICROPROCESSORS

| | |
|---|---|
| F8 | 16.95 |
| Z80 | 20.00 |
| Z80A | 25.00 |
| I802 | 19.95 |
| 2650 | 24.95 |
| AM2901 | 22.95 |
| 6502 | 10.95 |
| 6800 | 17.95 |
| 6802 | 24.95 |
| 8008-1 | 12.00 |
| 8035 | 22.00 |
| 8080A | 9.95 |
| 8085 | 27.00 |
| 8748 | 60.00 |
| TMS9900 | 67.00 |
| 8080A | |

### SUPPORT DEVICES

| | |
|---|---|
| 8212 | 3.00 |
| 8214 | 8.50 |
| 8216 | 3.75 |
| 8224 | 3.50 |
| 8224-4 | 9.95 |
| 8226 | 3.95 |
| 8228 | 7.95 |
| 8238 | 7.50 |
| 8251 | 8.75 |
| 8253 | 20.95 |
| 8255 | 11.00 |
| 8257 | 19.95 |
| 8259 | 19.95 |
| 8275 | 75.00 |
| 8279 | 20.00 |

### 6800 SUPPORT

| | |
|---|---|
| 6810P | 4.95 |
| 68B10P | 6.00 |
| 6820P | 7.50 |
| 6821P | 7.50 |
| 6828P | 11.25 |
| 6834P | 16.95 |
| 6850P | 9.75 |
| 6852P | 11.75 |
| 6860P | 10.00 |
| 6862P | 14.50 |
| 6871P | 28.00 |
| 6875P | 8.75 |
| 6880 | 2.00 |

### CHARACTER GEN.

| | | |
|---|---|---|
| 2513 U/L | | 6.75 |
| 2513 (5v) | U/C | 9.75 |
| 2513 (5v) | L/C | 10.95 |
| 6571 | | 10.95 |
| 6571A | | 10.95 |
| 6574 | | 13.25 |

### DYNAMIC RAMS

| | |
|---|---|
| 416D (200ns) | 20.00 |
| 4116 (200ns) | 20.00 |
| 2104/4096 | 4.00 |
| 2107B-4 | 3.95 |
| TMS4027 | 4.00 |
| TMS4050 | 4.00 |
| TMS4060 | 4.50 |
| 4096 | 4.00 |
| MM5270 | 4.50 |

### PROMS

| | |
|---|---|
| 1702A | 5.00 |
| 2516 (5v) | 50.00 |
| 2708 | 9.00 |
| 2716 (TI) | 30.00 |
| 2716 (INTEL) | 50.00 |
| 2758 | 26.60 |

### STATIC RAMS

| | 1-63 | 64 up |
|---|---|---|
| 21L02 (450ns) | 1.50 | 1.18 |
| 21L02 (250ns) | 1.75 | 1.50 |
| 410D | 10.00 | 8.50 |
| 2101-1 | 2.95 | 2.50 |
| 2102 | 1.25 | .90 |
| 2111-1 | 3.25 | 2.65 |
| 2112-1 | 2.75 | 2.35 |
| 2114 (300ns) | 10.00 | 8.25 |
| 2114 (450ns) | 9.00 | 7.69 |
| 2125L | 11.00 | 8.30 |
| TMS4044 (250ns) | 8.95 | 8.00 |
| TMS4044 (450ns) | 8.20 | 7.40 |
| 4200A | 10.00 | 8.60 |
| TMS4045 (250ns) | 8.95 | 8.00 |
| TMS4045 (450ns) | 8.20 | 7.40 |

### FLOPPY DISC CHIPS

| | |
|---|---|
| 1771B-01 | 39.95 |

### KEYBOARD ENKODERS

| | |
|---|---|
| AY-5-2376 | 12.75 |
| AY-5-3600 | 13.75 |

| | | | | | |
|---|---|---|---|---|---|
| 21L02 (350ns) Static Rams 120 @ $1.00 ea. | 1702A E-PROM $4.75 ea. | 6502 Microprocessor 5 @ $11.00 ea. | 2708 (450ns) E-PROM 8 @ $7.50 ea. | 21L02 (250ns) Static Rams 100 @ $1.25 ea. | Z-80A Microprocessor 5 @ $25.00 ea. |
| Z-80 Microprocessor 5 @ $20.00 ea. | MM5257 lo pw repl. TMS4044 8 for $8.00 ea. | 2114L (250ns) 8 for $8.25 ea. | TMS 4044 (250ns) 16 @ $8.00 ea. | 4200A (200ns) Static Rams 25 @ $10.00 ea. | 4116 (200ns) 16K Dyn. Ram 8 @ $20.00 ea. |

## SHUGART 801R
## 8" FLOPPY DISC DRIVE.

$550.00

## KIM – 1
Assembled and Tested
**$245.00**

## MEMORY PLUS
for KIM–1
8K RAM (21L02)
8K EPROM
ASSEMBLED & TESTED
$245.00

## THE APPLE II COMPUTER

One of the best "Total Package" home and business computers on the market. "Basic" in ROM, Color Graphics, Floating Point Basic Package, etc.

16K version only **$1,095.00**

## 416D 16K x 1

Dynamic Ram Chip can be used for expanding Apple II Memory or the TRS-80 (200ns)

8 for $20.00ea.
16 for $18.00ea.

Call for quote on larger quantities

## full ASCII
## PROFESSIONAL KEYBOARDS

* Full 128 Character ASCII
* Tri-Mode MOS Encoding
* MOS/DTL/TTL Compatable Output
* Two-key Rollover
* Level and Pulse Strobe
* Shift and Alpha Lock
* Selectable parity
* Positive or Negative Logic.

MODEL 756 (56 keys)

PRICING INFORMATION

| | |
|---|---|
| Model 756 (assembled) | $59.95 |
| Model 756K (kit) | $49.95 |
| Model 702 enclosure | $29.95 |
| Model 710 numeric pad | $9.95 |
| Model 756MF Mtg.Frame | $8.95 |

## MOTHER BOARDS - S100 STYLE

| | |
|---|---|
| 9 slot "Little Mother" Assembled and Tested | $35.00 $75.00 |
| 13 slot with front panel slot | |
| Bare board | $35.00 |
| Kit | $70.00 |
| Assembled & Tested | $110.00 |
| 22 slot Assembled & Tested | $149.95 |

## CONNECTORS

DB-25P $2.25 DB-25S $3.25
COVER $1.50

| | |
|---|---|
| 44 Pin - PC & EYE | $1.95 |
| 44 Pin - WW | $2.50 |
| 86 Pin - (6800) PC | $5.00 |
| 86 Pin - (COSMAC ELF) PC | $5.00 |
| 100 Pin - (Altair) PC | $4.50 |
| 100 Pin - (Imsai) WW | $4.25 |
| 100 Pin - (IMSAI) PC | $3.25 |

## MODEL 801R Shugart Disc with Cabinet

Includes Cabinet, Disc Drive, Power Supply, Cable, Fan & Data Cable.
Has AC line filter.
Cabinet size 10"H x 10"W x 16"D
MODEL DM 2700-S **$750.00**

## FLOPPY DISC INTERFACE

JADE Floppy Disc (Tarbell Board) KIT **$175.00 ea.**

S.D. Computer Products
Versa Floppy Kit **$149.00 ea.**
Assembled & Tested **$189.00 ea.**

## STATIC RAM BOARDS
ASSEMBLED & TESTED

8K

| | |
|---|---|
| Ram 8 (250ns) | $169.95 |
| Ram 8B (450ns) | $139.95 |
| 250ns KIT Mem-1 | $169.95 |
| 450ns KIT Mem-1 | $125.00 |
| BARE BOARD | $25.00 |

16K Uses 2114 (lo pwr.)

| | |
|---|---|
| Ram 16 (250ns) | $375.00 |
| Ram 16B (450ns) | $325.00 |
| MEM-2 Kit (250ns) | $285.00 |
| 32K Assembled & Tested by SEALS ELECTRONICS | |
| JG-32 (250ns) | $795.00 |
| JG-32B (450ns) | $725.00 |
| 250ns KIT | $575.00 |

**6800 Adapter - adapts Mem-1 8K board to Motorola MEK 6800D2 evaluation kit.**

## 16K STATIC BOARD

with memory management can be used with Alpha Micro or Cromenco Systems. ASSEMBLED & TESTED

| | |
|---|---|
| RAM 65(250ns) | $390.00 |
| RAM 65B (450ns) | $350.00 |

## E-PROM BOARDS

| | | |
|---|---|---|
| MR-8 (8K uses 2708) with 1K RAM | KIT | $99.50 |
| MR-16T (16K uses 2716) with 1K RAM | KIT | $99.50 |
| EPM-1 (uses up to 4K of 1702) | | $59.95 |
| RAM/N/ROM (16K uses any E-PROM) | KIT | $117.00 |
| JG-8/16 (uses 2708 or 2716) | KIT | $59.95 |
| BARE BOARD | | $30.00 |

**EXPANDABLE E-PROM** – S.D. Computer Products
16K or 32K EPROM $49.95 without EPROM

Allows you to use either 2708's for 16K of Eprom or 2716's for 32K of EPROM.

## COMPUTER MAINFRAME

Includes: **$295.00**
Power Supply +8v at 18amps ±16v at 2 amps
Mother Board - 12 slots with connectors Assembled & Tested
Has Whisper Quiet Fan & AC Line Filter
Cabinet size 7"H x 19"W x 22" D

## DYNAMIC RAM BOARD
by S. D. Computer Products

On board refresh is provided with no wait states or cycle stealing required. +8 VDC 400MA DC, +18 VDC 400MA and -18 VDC 30MA DC.

**EXPANDABLE 32K** uses 4115 (200ns)
8K Kit $151.00 24K Kit $325.00
16K Kit $240.00 32K Kit $400.00

**EXPANDABLE 64K** uses 4116 (200ns)
16K Kit $250.00 48K Kit $675.00
32K Kit $475.00 64K Kit $875.00

**JADE 16K DYNAMIC KIT** uses 4096 (300ns) **$200.00**

## JADE Z80 KIT

with PROVISIONS for ONBOARD 2708 and POWER ON JUMP

| | |
|---|---|
| (2MHZ) | $135.00ea. |
| Assembled & Tested | $170.00ea. |
| (4MHZ) | $149.95ea. |
| Assembled & Tested | $184.95ea. |
| Bare Board | $35.00ea. |

## JADE VIDEO INTERFACE KIT

KIT **$99.95**
Assembled & Tested **$139.95**

S-100 Bus compatible
32 or 64 Characters per line - 16 lines
Graphics (128 x 48 matrix)
Parallel & compositive video
On board low-power memory

Powerful software included for cursor, home, EOL, Scroll Graphics/Character Upper case, lower case and Greek. Black-on-white & White-on-black.

## JADE PARALLEL/SERIAL INTERFACE KIT

KIT **$124.95**
Assembled & Tested **$154.95**

* S-100
* 2 Serial interfaces with RS232 interfaces or 1 Kansas City cassette interface.
* Serial interfaces are crystal controlled
* Selectable baud rates.
* Cassette works up to 1200 baud.
* 1 parallel port.

## TV-1

Convert T.V set to Video Monitor
KIT . . . $8.95

## JADE 8080A KIT

$100.00 KIT
BARE BOARD $30.00

# JADE
## Computer Products

4901 W. Rosecrans
Hawthorne, Calif. 90250 U.S.A.

*Phone: (213) 679-3313*
*Telex: 18-1823 JADECP HWTH*

Discounts available at OEM quantities. All prices above are quoted in U.S. $ F.O.B. Hawthorne, Calif., U.S.A. To expedite shipments please include international money order or bankers check payable against any U.S. bank in U.S. $, or use your Barclay, Access, or American Express credit card (include card number, expiration date, and signature). Add 20% of total order for postage.

● Circle No. 125

**Four off-beat entrepreneurs in Surbiton are enjoying themselves picking up old computers and making them work. And, what is more, they're making money at it. We visit Galdor Computing to find the secret of their success.**


# ELECTRONIC GRAVEYARD

THE ENTRANCE to Galdor Computing Ltd, at 52 Brighton Road, Surbiton, is a narrow doorway sandwiched between a men's boutique and an insurance broker. Through it, you enter a dark passage where there is just room to squeeze past a tired-looking disc drive, two huge air-conditioning units and several dusty boxes of components. The passage leads to what was once the back garden of No. 52, now the Galdor computer room.

It is no ordinary computer room; at first sight it is more like an electronic elephant's graveyard. Some 40 ft. square by 9 ft. high, it is packed full of machinery —so full that there is barely room to move between the grey cabinets.

Almost every known peripheral seems to be represented. An old ICT card interpreter stands in a corner; a multiplexer nudges a graph plotter. Some cabinets stand open to reveal half-embowelled innards, and every flat surface is stacked with boxes of cards, discs and tapes, chunks of abandoned core store, manuals, and PCBs by the hundred.

Somewhere in the midst of all this you should find one or more directors of Galdor, perhaps mending a disc drive, testing a highly-modified version of an ICL operating system, playing Star Trek, or running a job for a customer. For while Galdor is a properly-constituted limited company offering conventional bureau services, it is not run in the same way or with the same aims as an ordinary computer bureau.

## Natural

Perhaps the easiest way to understand Galdor is to go back to its beginnings. Around 1968, half-a-dozen students of electrical engineering at Kingston Polytechnic decided they needed more computing facilities than the Poly could offer them. It seemed the natural thing to buy their own, so they did. London University was dispensing with an ICT 1301, and they obtained it for £300, buyer to collect.

Finding somewhere to put it was no problem; there was plenty of room in the back garden. It was a matter of getting planning permission, knocking down the existing shed, building a computer room from the foundations, dismantling the computer, transporting it to Surbiton, and putting it together again. Nothing to it.

Flippancy aside, the nonchalance with which Galdor "staff" tackle the most daunting projects is one of the things which most impresses the visitor. Another is their equal familiarity with hardware and software. Commercial computing tends to divide people into hardware or software specialists, so that real all-round knowledge has until recently been a rarity. At Galdor it is taken for granted.

## Then there were two

Of the original half-dozen, only Andrew Keen and Stuart Fife remain. They have been joined by Pete Singleton, who paid a visit two years ago, decided to stay the night, and has been there ever since.

The three form the full-time staff of Galdor. Another director, John Sheane, has a job with ICL, and there are part-time helpers who drop in whenever possible.

After serving long and well, the ICT 1301 was replaced a little over a year ago and is now being re-commissioned by another enthusiast. Since then, hardware development has been extremely rapid. The first replacement was an ICL 1901, which quickly showed itself to be very short of processing power. It was soon replaced by an ICL 1903.

That is the machine which Galdor is now running, equipped with 32K words of core store, six EDS8 disc drives, eight 7-track tape drives, paper tape reader and punch, card reader and punch, and two line printers of 600 and 1,250 lines per minute.

That, however, is by no means the end of the story. One main reason for the congestion in the computer room is the vast bulk of a 1905E which stretches almost the length of the room. It has a 128K

*(continued on next page)*

Andrew Keen at the console.

Stuart Fife assists young visitors to play Lunar Lander.

(*continued from previous page*)

store and will be equipped with the same eight tape drives, but no fewer than 14 disc drives, three printers, a graph plotter, a local video controller initially with three VDUs, and a multiplexer to handle up to 10 telephone channels.

All the hardware is already there and the process of getting it running is already well under way. "We seem to have bitten off a bit more than we can cope with this time," says Keen, peering over the clutter. All the same, he is already planning the acquisition of a second 1905 to provide back-up, as soon as room can be found for it.

Acquiring all that equipment has made the members of Galdor experts at the Steptoe game. First, they look through the *Computer Users' Year Book* to see which firms are still running the kind of machines in which they are interested. Those firms are then contacted to see if they have anything they wish to dispose of.

## High hit rate

The method produces quite a high hit rate. "The trick then," says Keen, "is to offer fractionally more than the scrap merchant," though often they find that firms are sympathetic and generous once they learn about Galdor.

The result is that much of the equipment goes to them at near-giveaway prices. One of the line printers, for example, cost £30, and disc packs are in plentiful supply at around 50p each.

The 1905E cost £2,000, a massive sum by Galdor standards. Keen points out that for around the same sum they could have bought a modern micro with about the same processing power, but that of course, would have nothing like the same capacity for handling peripherals.

Naturally enough, not all the equipment acquired in this way is in perfect working order. One or two of the tape decks, for example, have a nasty habit of splitting tapes, and disc drives are often a source of trouble. That is a problem which Galdor members take in their stride and there is very little with which they are not capable of dealing, though there are times when it proves simpler to replace the faulty unit rather than track down and cure the fault.

Financially, Galdor has always been self-supporting. Almost from the start, it sold machine time on the 1301. The first customer was a friendly society which took one and a half hours a day and the volume has grown steadily ever since.

Today, the work includes such projects as mailing lists for clubs and societies and a back-up service for firms running 1900 series machines of their own.

Predictably, Galdor rates are among the lowest in the land—at £12 an hour. Even that may be modified and payment in kind accepted from particularly hard-up customers. It is characteristic that Fife sees this as an advantage, allowing Galdor to offer a service to organisations which would otherwise not be able to afford it.

The unconventional approach to computing seems to affect some customers. Users perhaps unacustomed to the idea that computing can be fun are liable to offer thanks for a pleasant evening by "accidentally" leaving behind a couple of boxes of stationery. Singleton is doing some programming for a firm which is test-marketing beefburgers. The "spin-off" from this project has considerable effect on the diet of the resident members.

All this adds up to a method of running a computer bureau which is unlikely to make anyone a fortune. Turnover in 1977 was regarded as healthy at £180 a week showing a trading profit of £40-60 a week; not the kind of figures to set a bank manager's heart aglow.

Making a fortune, however, is clearly not one of the Galdor aims. What those aims are is less easy to define and seems to depend largely on which of the directors or helpers you choose to ask.

In general, the sheer pleasure of building and running a large computer system without the normal pressures and restrictions of commercial life seems paramount. The profit motive is conspicuous by its absence.

## Playing hard

Galdor finances, while they must be a taxman's nightmare, are extremely simple by normal standards. Since all workers are either directors or paid voluntary helpers, the firm has no employees, a fact which eliminates effectively nearly all the bureaucracy involved in running a business. While Galdor customers expect and receive professional service, its members are relatively free to decide whether or not to accept a particular job.

It was perhaps Singleton, who admits to being unemployed more times than he cares to tell, who put it best. "I decided long ago," he said, "that work was a dead loss. So nowadays I don't work—I just play hard."

Galdor positively welcomes visitors, whether just to look round, to make use of the machine, or to help. Most professional users, and possibly even more equipment, are also welcome.

*Galdor is at 52 Brighton Road, Surbiton, Surrey. Tel: 01-399 1300.* ■

# ILLUSTRATING BASIC

## ( A SIMPLE PROGRAMMING LANGUAGE )

**THIS MONTH we begin a series of articles on how to program in Basic, probably the most widely-used programming language for small computers.**

**For this series we have secured the serialisation rights for one of the best books on the subject, *Illustrating BASIC*, by Donald Alcock.**

**Each month we shall publish a chapter from the book which was specially written for the newcomer. Even if you are not in that category, it can still teach you a great deal.**

**It is written with 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.**

★

**Alcock Illustrating BASIC Chapter 1. © Cambridge University Press. Reprinted by permission.**

★

**Copies of *Illustrating BASIC* can be obtained from *Practical Computing*. See the enquiry card in this issue for details.**

## CONTENTS

# PREFACE

TO MAKE A COMPUTER DO A CALCULATION — HOWEVER SIMPLE — YOU MUST FIRST DESCRIBE EVERY STEP OF THAT CALCULATION IN A *LANGUAGE* THE COMPUTER CAN UNDERSTAND; THIS DESCRIPTION IS CALLED A *PROGRAM*. THIS BOOK PRESENTS A POPULAR AND WIDELY AVAILABLE LANGUAGE CALLED *BASIC* AND EXPLAINS HOW TO WRITE SIMPLE PROGRAMS IN IT.

ALTHOUGH MORE ELEGANT AND POWERFUL LANGUAGES ARE FAVOURED BY PROFESSIONAL PROGRAMMERS *BASIC* IS ADEQUATE FOR MOST NON-PROFESSIONALS AND EXCELLENT FOR BEGINNERS.

*BASIC* WAS BORN IN AMERICA AT DARTMOUTH COLLEGE, NEW HAMPSHIRE, IN 1964 AS A SIMPLE COMPUTER LANGUAGE FOR BEGINNERS. IT PROVED POPULAR AND HAS BEEN COPIED AND EXTENDED BY MANY COMPUTER MAKERS, COLLEGES, UNIVERSITIES AND "TIME-SHARING" SERVICES. NOW, LIKE ENGLISH, *BASIC* HAS *DIALECTS*: A PROGRAM WRITTEN IN ONE IS UNLIKELY TO WORK ON A COMPUTER COMMITTED TO ANOTHER.

THERE IS SOON TO BE A STANDARD FOR "MINIMAL BASIC" BY THE AMERICAN NATIONAL STANDARDS INSTITUTE † (A.N.S.I. X3J2), AND "SPECIFICATION FOR STANDARD BASIC" BY BULL, FREEMAN & GARLAND HAS BEEN PUBLISHED BY THE NATIONAL COMPUTING CENTRE, U.K.* (1973). THESE HAVE NOT YET HAD TIME TO ENCOURAGE EVERYONE TO FALL INTO LINE SO THE *BASIC* YOU MEET WILL PROBABLY NOT BE STANDARD. I HAVE ACCEPTED THIS AS A FACT OF LIFE, AND, IN WRITING THIS BOOK, KEPT AT MY SIDE ELEVEN MANUALS — EACH DEFINING A DIFFERENT *BASIC*. FOUR OF THESE VERSIONS ARE AVAILABLE ON BIG COMPUTERS OPERATED BY INTERNATIONAL "TIME-SHARING" SERVICES; THE OTHER SEVEN ON COMPUTERS RANGING FROM BIG TO "DESK-TOP". FROM THESE ELEVEN MANUALS I HAVE TRIED TO DISCOVER AND POINT OUT WHERE *BASICS* COMMONLY DIFFER FROM ONE ANOTHER AND RECOMMEND WAYS OF AVOIDING DEPENDENCE ON ANY ONE PARTICULAR VERSION. I HAVE USED THE WORD "PORTABLE" TO DESCRIBE A *BASIC* PROGRAM WRITTEN WITH INDEPENDENCE IN MIND — AND TREAT THE NEED FOR PORTABILITY AS AN AXIOM.

† A.N.S.I. X3J2/76-01 ISSUED FOR PUBLIC COMMENT IN JANUARY 1976

* REFERRED TO IN THE TEXT AS "N.C.C. STANDARD BASIC"

ILLUSTRATING BASIC PAGE 1

YOU DON'T HAVE TO BE A COMPUTER SCIENTIST TO READ THIS BOOK; IT IS FOR STUDENTS MEETING COMPUTERS FOR THE FIRST TIME; FOR THOSE IN INDUSTRY (PARTICULARLY ENGINEERS) WHO NEVER FORMALLY STUDIED COMPUTING BUT WOULD LIKE TO WRITE SIMPLE COMPUTER PROGRAMS; FOR MANAGERS WHO DO NOT WANT TO WRITE PROGRAMS BUT WOULD LIKE TO KNOW MORE ABOUT A FIELD IN WHICH THEY OFTEN HAVE TO TAKE DECISIONS; AND FOR THOSE WHO CAN ALREADY WRITE IN *BASIC* BUT SEEK A BROADER VIEW OF "PORTABLE" PROGRAMMING AND AN INTRODUCTION TO A FEW PROGRAMMERS' TECHNIQUES LIKE "STATE TABLES" AND "LIST PROCESSING".

THE TEXT OF THE BOOK IS ARRANGED FOR THE MOST PART IN DOUBLE-PAGE SPREADS, EACH DEALING WITH A SINGLE "STATEMENT" OF THE *BASIC* LANGUAGE. BECAUSE SO MANY STATEMENTS ARE INTERDEPENDENT THIS ARRANGEMENT DEMANDS FORWARD REFERENCES NOW AND AGAIN, BUT NOVICES TO COMPUTING MAY IGNORE FORWARD REFERENCES FIRST TIME THROUGH THE BOOK WITHOUT FEAR OF MISSING SOMETHING ESSENTIAL TO UNDERSTANDING THE SUBJECT MATTER.

I RECORD MY DEEP GRATITUDE TO THREE PEOPLE WHO MADE THIS BOOK POSSIBLE; MY WIFE, FAY, WHO SUFFERED GRASS-WIDOWHOOD BUT NEVER CEASED HER WARM ENCOURAGEMENT; MY PARTNER, BRIAN SHEARING, WHO HAS TAUGHT ME SO MUCH ABOUT COMPUTING AND ALLOWED ME TIME OFF WORK TO WRITE THE BOOK; AND CHARLES LANG WHO BELIEVED IN MY IDEAS AND PERSUADED ME TO GIVE THEM FORM.

MY BOOK IS INFORMAL IN LANGUAGE AND UNUSUAL IN PRESENTATION. RATHER THAN WRITE A JUSTIFICATION I WOULD ONLY REMARK THAT A CAREFUL READER MIGHT DIAGNOSE A SEVERE ASTIGMATISM IN MY EYE AND A PERSISTENT SHAKE IN MY HAND.

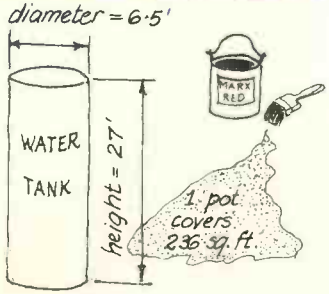
REIGATE,
SURREY, U.K.

*Donald Alcock*
JANUARY 1977

# 1 COMPONENTS OF THE LANGUAGE

## PROBLEM

HOW MANY POTS OF PAINT DO YOU NEED TO PAINT THE ROOF AND WALL OF THIS WATER TANK?




WE COULD GO STRAIGHT AT IT LIKE THIS:

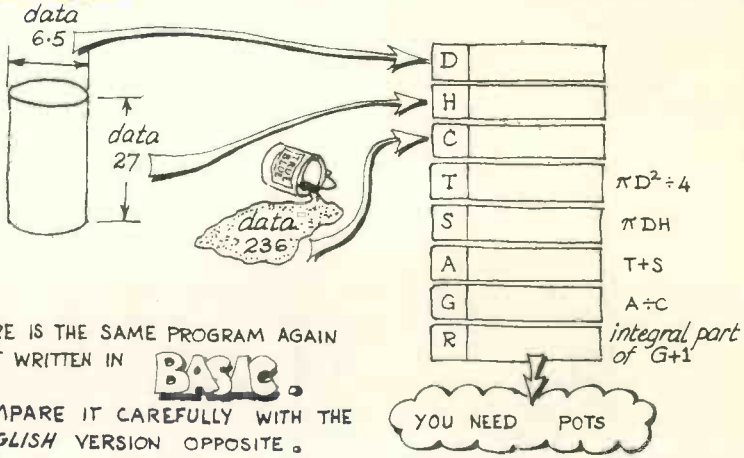
roof area, $T = \pi \times 6{\cdot}5^2 \div 4 = 33{\cdot}2$
wall area, $S = \pi \times 6{\cdot}5 \times 27 = 551$
total area, $A = T + S = 584{\cdot}2$
number of pots, $G = A \div 236 = 2{\cdot}48$
rounding up, $R = 3$
∴ you need 3 pots of paint

OR WE COULD WRITE A *PROGRAM* (IN ENGLISH) TO SOLVE THE PROBLEM.

1. **REM**ARK: A PROGRAM IN ENGLISH
2. THE FOLLOWING NUMBERS ARE THE **DATA** 6·5, 27, 236
3. **READ** THE DATA, CALLING THEM **D, H & C** RESPECTIVELY
   *(think of this as putting the data into little boxes labelled D, H & C respectively – see opposite page)*
4. WORK OUT $3{\cdot}14 \times D^2 \div 4$ AND LET THE RESULT BE CALLED **T**
   *(i.e. put the result in a little box labelled T)*
5. WORK OUT $3{\cdot}14 \times D \times H$ AND LET THE RESULT BE CALLED **S**
6. ADD **T** TO **S** AND LET THE RESULT BE CALLED **A**
7. WORK OUT **A ÷ C** AND LET THE RESULT BE CALLED **G**
8. ROUND **G** TO THE NEXT WHOLE NUMBER AND LET THE RESULT BE CALLED **R**
   *(i.e. add 1 to G and take the integral part of the result)*
9. **PRINT** "YOU NEED"; **R**; "POTS"
   *(i.e. print whatever whole number R turns out to be)*
10. THE **END**

THIS HAS THE ADVANTAGE OF BEING GOOD FOR ANY SIZE OF TANK AND PAINT POT ⇄ YOU NEED ONLY REPLACE THE DATA ON LINE 2.


TRY *OBEYING* THE ENGLISH PROGRAM OPPOSITE ⇄ FEEL WHAT IT WOULD BE LIKE TO *BE* A COMPUTER ⇄ *DEFILE* THIS PAGE BY WRITING NUMBERS IN THE LITTLE BOXES BELOW.




HERE IS THE SAME PROGRAM AGAIN BUT WRITTEN IN BASIC.

COMPARE IT CAREFULLY WITH THE *ENGLISH* VERSION OPPOSITE.

```
1   REM  A PROGRAM IN BASIC
2   DATA 6.5, 27, 236
3   READ  D , H , C
4   LET  T = 3.14 * D ↑ 2 / 4
5   LET  S = 3.14 * D * H
6   LET  A = T + S
7   LET  G = A / C
8   LET  R = INT (G + 1)
9   PRINT  "YOU NEED"; R ; "POTS"
10  END
```

*notice*
* meaning multiply
↑ meaning raise to a power
/ meaning divide

AND THIS, WHEN OBEYED, WOULD PRODUCE:

```
YOU NEED 3 POTS
```

**FIRST** PREPARE YOUR PROGRAM BY TYPING INSTRUCTIONS AT THE KEYBOARD ⇝ THE COMPUTER SIMPLY *STORES* THE PROGRAM AT THIS STAGE :

**IT DOESN'T OBEY ANY INSTRUCTIONS**

```
1 REM  A PROGRAM IN BASIC
2 DATA 6.5, 27, 236
3 READ  D, H , C
4 LET  T=3.14*D↑2/4
5 LET  S=3.14*D*H
6 LET  A=T + S
7 LET  G=A/C
8 LET  R=INT(G+1)
9 PRINT "YOU NEED";R;"POTS"
10 END
```

**THEN** TYPE RUN

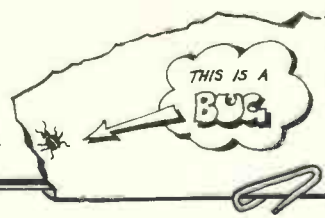
WHICH SETS THE COMPUTER TO WORK *OBEYING* THE STORED INSTRUCTIONS ONE AFTER THE OTHER ⇝ IN NUMBERED SEQUENCE ⇝ WHILST YOU RELAX .


EVENTUALLY THE COMPUTER WILL OBEY THE INSTRUCTION END . THAT MAKES IT STOP .

```
RUN
YOU NEED 3 POTS
```

**BUT** BEFORE YOU CAN TAKE THE FIRST STEP AND START TYPING THE PROGRAM YOU HAVE TO GO THROUGH THE RITUAL OF *SIGNING ON* AND TELLING THE COMPUTER YOU WANT TO USE *BASIC*.

DIFFERENT COMPUTERS (( EVEN IDENTICAL COMPUTERS RUN BY DIFFERENT ORGANISATIONS )) OFTEN HAVE DIFFERENT WAYS OF DOING THESE THINGS , SO IF YOU WANT TO TRY THE PROGRAM NOW GET SOMEONE WHO "KNOWS THE SYSTEM" TO SIGN ON FOR YOU AND CALL UP *BASIC*.



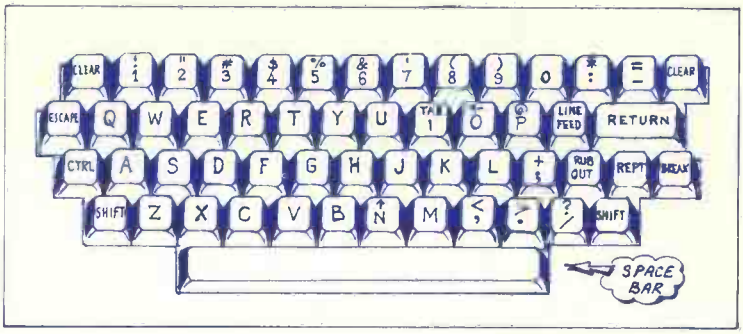

"*BUG*" IS COMPUTER FOR A MISTAKE . TH SHOWS THIS LITTLE B AGAINST ILLUSTRATIO MISTAKES IN PROGR

ILLUSTRATING BASIC PAGE 4

# KEYBOARD

EVERY PROGRAM IN *BASIC* HAS TO BE TYPED ON A KEYBOARD

*PROBABLY SOMETHING LIKE THIS*




ALTHOUGH POSITIONS OF LETTERS & DIGITS ARE THE SAME ON MOST KEYBOARDS, KEYS LIKE RUB OUT & BREAK IN THE PICTURE ABOVE VARY IN NAME, POSITION AND FUNCTION FROM ONE INSTALLATION TO ANOTHER.

NOTICE ALL LETTERS ARE CAPITAL LETTERS. NOTICE ALSO THERE IS A KEY FOR 1 AND A KEY FOR ZERO (BOTH IN THE TOP ROW). NEVER PRESS THE LETTERS I AND O IN THEIR PLACE.

AS ON AN ORDINARY TYPEWRITER, PRESSING SHIFT AT THE SAME TIME AS ANOTHER KEY GIVES THE CHARACTER SHOWN ON THE UPPER HALF OF THAT KEY: THUS # 3 TOGETHER WITH SHIFT GIVES # WHEREAS # 3 ALONE, 3.

THE "BACK ARROW" ← SERVES TO DELETE THE CHARACTER ON ITS LEFT FROM THE COMPUTER'S MEMORY; TWO OF THEM DELETE THE PREVIOUS TWO CHARACTERS, AND SO ON. THUS IF YOU TYPE PRIMP←←NT THEN *BASIC* RECEIVES THE WORD PRINT. (REMEMBER THIS BY EXCLAIMING *OH SHIFT!* WHEN YOU HIT THE WRONG KEY.) SOME *BASICS*, HOWEVER, USE AN UNDERSCORE CHARACTER FOR THIS PURPOSE: PRIMP__NT.

MOST *BASICS* USE A KEY (PERHAPS "RUBOUT") WHICH, WHEN PRESSED, DELETES THE WHOLE OF THE LINE YOU ARE TYPING FROM THE COMPUTER'S MEMORY; ANOTHER (PERHAPS "BREAK") STOPS A PROGRAM RUNNING.

FOR A *NEW LINE* IN *BASIC* PRESS RETURN

WHEN TYPING "OFF LINE" PRESS RETURN then LINEFEED

JARGON
E BOOK
UG
ONS OF
AMS.

CHAPTER 1

# TYPING

IF YOU INTEND TO USE *BASIC* A LOT, LEARN TOUCH TYPING. TEN FINGERS ARE FASTER AND LESS FRUSTRATING THAN TWO.

THERE IS A LIMIT TO THE LENGTH OF A TYPED LINE — MOST *BASICS* ALLOW LINES UP TO 72 CHARACTERS LONG. SOME ALLOW LONGER LINES BUT IT IS BEST TO ACCEPT A LIMIT OF 72.

SOME *BASICS* ALLOW GREAT FREEDOM WITH THE SPACE BAR; SOME DISREGARD SPACES EXCEPT THOSE BETWEEN QUOTATION MARKS. THUS IT WOULD BE ALLOWABLE TO TYPE:

```
8FORD=STOP
```

INSTEAD OF:

```
8 FOR D = S TO P
```

BUT IT IS OBVIOUSLY SILLY TO OBSCURE THE MEANING OF THE PROGRAM IN ORDER TO SAVE A FEW TAPS ON THE SPACE BAR.

SOME *BASICS* REFUSE TO ALLOW SPACES WITHIN THE CONTROLLING WORDS OF THE LANGUAGE. THUS THE FOLLOWING WOULD BE WRONG:

```
23 L E T  A = B+C
```

SOME *BASICS* DEMAND AT LEAST ONE SPACE BEFORE EACH CONTROLLING WORD, OR AFTER IT, OR BOTH:

```
20DATA 6.5, 27, 236
80 PRINT"YOU NEED";R;"POTS"
```

SOME *BASICS* REFUSE TO ACCEPT SPACES WITHIN LINE NUMBERS BUT DO NOT OBJECT TO THEM INSIDE OTHER NUMBERS:

```
1 000 LET A=1 000 . 0
1000 LET A=1000.0
```

SOME BASICS OBJECT TO THESE TOO

SOME *BASICS* DO NOT ALLOW SPACES IN FRONT OF LINE NUMBERS:

```
 95 LET A=B
100 LET C= D*F + G
```

SPACES OPTIONAL HERE

GENERALLY WHEN *ONE* SPACE IS ALLOWED (OR DEMANDED) THEN *SEVERAL* ARE ALLOWED. AND GENERALLY A SPACE IS OPTIONAL ON EITHER SIDE OF THESE ⇨ ( , ; * + / − = ↑ > < )
BUT NOT IN 1.5E2 (SEE PAGE 9) NOR BETWEEN > AND = (SEE PAGE 41).

A PROGRAM WHICH ACCEPTS ALL THESE RESTRICTIONS SHOULD BE ACCEPTABLE TO ANY VERSION OF *BASIC*.

# LINE NUMBERS

LEAVE GAPS IN THEIR SEQUENCE THUS :

USE 10s OR 5s

```
10  REM    A PROGRAM IN BASIC
20  DATA   6.5,  27,  286
30  READ   D, H, C
40  LET    T = 3.14 * D↑2/4
50  LET    S = 3.14 * D * H
60  LET    A = T + S
70  LET    G = A/C
80  PRINT "YOU NEED"; R; "POTS"
90  END
```

THIS IS 236. SEE PAGE 5

THERE IS A *MISTAKE* IN THIS PROGRAM : THE LAST LET WAS FORGOTTEN. INSERTING IT IS SIMPLE ; JUST TYPE :

```
75 LET R=INT(G+1)
```

AND THE COMPUTER PUTS LINE 75 BETWEEN LINE 70 & LINE 80. IT MAKES NO DIFFERENCE IN WHAT ORDER YOU *TYPE* THE LINES ; THE COMPUTER SORTS THEM INTO ASCENDING ORDER OF LINE NUMBER.

IF YOU TYPE SEVERAL LINES WITH THE *SAME* LINE NUMBER THE COMPUTER OBLITERATES EACH PREVIOUS VERSION THUS ACCEPTING THE LINE TYPED LAST. IF THE LINE TYPED LAST IS *JUST* A LINE NUMBER WITH NOTHING AFTER IT THEN THE *WHOLE LINE VANISHES* FROM THE COMPUTER'S MEMORY INCLUDING THE LINE NUMBER. THAT IS HOW TO DELETE UNWANTED LINES. THUS :

```
120 LET A = B + C
125 LET E = F
120 LET A = B + G
125
120 LET A=B
```

RESULTS IN THE COMPUTER REMEMBERING ONLY :

```
120 LET A = B
```

THE FIRST LINE NUMBER IN A PROGRAM MUST BE GREATER THAN 0. THERE IS ALWAYS A LIMIT TO THE HIGHEST LINE NUMBER ; SOME *BASICS* STOP AT 9999, SO IT IS BEST TO ACCEPT THIS AS THE LIMIT.

THE *LAST* STATEMENT OF EVERY PROGRAM MUST BE : (NO OTHER STATEMENT BUT THE LAST MAY SAY END). END

CHAPTER 1

# STATEMENTS

A *BASIC* PROGRAM IS A SEQUENCE OF NUMBERED LINES CALLED *STATEMENTS*.

A STATEMENT MAY SIMPLY *STATE* SOMETHING

```
110 DATA 1, 2, 4
120 END
```

```
30 READ A,B,C
40 LET G=A*B↑2+C
50 PRINT "ANSWER IS";G
```

OR IT MAY *INSTRUCT* THE COMPUTER TO *DO* SOMETHING. A COMMON SYNONYM FOR *STATEMENT* IS *INSTRUCTION*; THE STATEMENTS THAT *DO* THINGS ARE *EXECUTABLE* INSTRUCTIONS.

THE COMPUTER FINDS OUT WHAT IS STATED OR WHAT TO DO BY LOOKING AT THE FIRST WORD: DATA, END, READ, LET *etc.*

OR SOMETIMES AT THE FIRST *TWO* WORDS: MAT READ, MAT PRINT *etc.* (( WE MEET MAT ON PAGE 76 )).

BUT THERE IS AN IMPORTANT EXCEPTION:

THE WORD LET MAY BE *OMITTED* IN MOST VERSIONS OF *BASIC*.

```
40  G = A*B↑2 + C
```

## REM

REM STANDS FOR *REMARK*.
REM STATEMENTS CAUSE NO ACTION BY THE COMPUTER; YOU INCLUDE THEM TO CLARIFY YOUR PROGRAM.

```
10 REM          *** WATER TANKS ***
20 REM
30 REM   A PROGRAM TO ILLUSTRATE BASIC
40 REM   -*-*-*-*-*-*-*-*-*-*-*-*-*-*-
50 DATA    6.5,    27,     236
60 REM     DIAM, HEIGHT, COVERAGE
```

REM FOR BLANK LINES

REM FOR EMBELLISHMENT

REM FOR CLARITY

THE EXAMPLES IN THIS BOOK DO NOT HAVE MANY "REM" STATEMENTS BECAUSE I HAVE ANNOTATED PROGRAMS WITH LITTLE ARROWS AND CLOUDS SO AS TO SAVE SPACE.

# NUMBERS

YOU CAN TYPE NUMBERS THREE WAYS ⇝ AS *INTEGERS*, AS *REALS* OR IN *E-FORM*.

*INTEGER* FORM
(WHOLE NUMBERS)

*REAL* FORM
(DECIMAL NUMBERS)

```
160 DATA   0,  2, +4, 1000, -30

170 DATA +0.70, 4., .6, -1.3, 987.65
```

4. MEANS 4.0

*E*-FORM
(EXPONENT FORM)
WHERE *E* SAYS:
"TIMES TEN TO THE..."

```
190 DATA 1E3, 13.6E-4, -13.6E6, -.0136E9
```

$1.0\times10^{3}$ = 1,000 ; $13.6\times10^{-4}$ = 0.00136 ; $-13.6\times10^{6}$ = −13,600,000 ; $-.0136\times10^{9}$ = −13,600,000

*E* INTRODUCES AN *INTEGER* SAYING HOW MANY PLACES TO SHIFT THE DECIMAL POINT. SHIFT TO THE RIGHT IF THE INTEGER IS POSITIVE; OTHERWISE LEFT.

```
190 DATA  E3, 13.6E1.2, 13.6 E 2.
```

IN THE *E* FORM THERE MUST ALWAYS BE A NUMBER IN FRONT OF THE *E* AND AN *INTEGER* AFTER IT. SOME *BASIC*S ALLOW SPACES WITHIN AN *E* FORM BUT IT IS BEST NOT TO HAVE THEM.

## ±10^38

IN SOME *BASIC*S THE BIGGEST NUMBER THAT CAN BE STORED IS APPROXIMATELY $\pm10^{38}$ (*BIG* MEANS FAR FROM ZERO ON EITHER SIDE; *SMALL* MEANS CLOSE TO ZERO ON EITHER SIDE). OTHER *BASIC*S CAN HANDLE MUCH BIGGER NUMBERS THAN $\pm10^{38}$; IT DEPENDS ON THE COMPUTER'S "WORD LENGTH" AND WHETHER THE "WORDS" ARE USED SINGLY, IN PAIRS, OR IN MULTIPLES. BUT ***NO*** *BASIC* SHOULD REFUSE TO HANDLE A NUMBER AS BIG AS ±100,000,000,000,000,000,000,000,000,000,000,000,000,000.

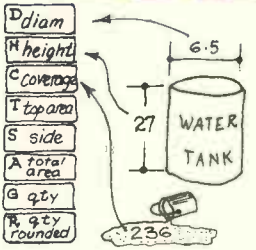
## 6~7 SIG.FIG.

IN SOME *BASIC*S THE PRECISION OF STORAGE AND ARITHMETIC IS BETWEEN 6 AND 7 SIGNIFICANT DECIMAL DIGITS ⇝ 987,654,321 WOULD BE STORED AS APPROXIMATELY 987,654,000. OTHER *BASIC*S OFFER MUCH HIGHER PRECISION, 15 SIGNIFICANT FIGURES BEING TYPICAL. AGAIN IT DEPENDS ON THE COMPUTER'S "WORD LENGTH" AND HOW THE "WORDS" ARE USED. BUT ***NO*** *BASIC* SHOULD WORK TO LESS PRECISION THAN 6 TO 7 SIG. FIGS.

(THE VAGUENESS OF "6 TO 7" IS BECAUSE MOST COMPUTERS USE BINARY ARITHMETIC, NOT DECIMAL. A MORE PRECISE RENDERING WOULD BE "24 BINARY DIGITS FOR POSITIVE NUMBERS; 23 FOR NEGATIVE; OR VICE VERSA" BUT THESE IMPLICATIONS NEED NOT BOTHER THE NOVICE TO *BASIC*.)

# VARIABLES

THERE ARE 286 *SIMPLE* NUMERICAL VARIABLES IN *BASIC*.




WE SAW SOME OF THESE ON PAGE 3. THEY ARE THE LITTLE BOXES USED TO HOLD NUMBERS.

THE FULL 286 ARE SHOWN BELOW. IT IS USEFUL TO KEEP A LARGE-SCALE CHART LIKE THIS AND MAKE A PHOTOCOPY FOR EACH NEW PROGRAM. AS YOU USE EACH VARIABLE WRITE A NOTE IN ITS BOX SAYING WHAT YOU USE IT FOR. THIS STOPS YOU USING VARIABLES ALREADY USED FOR SOMETHING ELSE — A COMMON SOURCE OF TROUBLE.

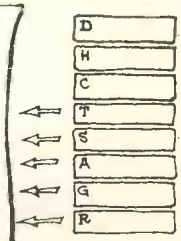
| A | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 |
|---|---|---|---|---|---|---|---|---|---|---|
| B | B0 | B1 | B2 | B3 | B4 | B5 | B6 | B7 | B8 | B9 |
| C | C0 | C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 |
| D | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 |
| E | E0 | E1 | E2 | E3 | E4 | E5 | E6 | E7 | E8 | E9 |
| F | F0 | F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 | F9 |
| G | G0 | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 | G9 |
| H | H0 | H1 | H2 | H3 | H4 | H5 | H6 | H7 | H8 | H9 |
| I | I0 | I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | I9 |
| J | J0 | J1 | J2 | J3 | J4 | J5 | J6 | J7 | J8 | J9 |
| K | K0 | K1 | K2 | K3 | K4 | K5 | K6 | K7 | K8 | K9 |
| L | L0 | L1 | L2 | L3 | L4 | L5 | L6 | L7 | L8 | L9 |
| M | M0 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | M9 |
| N | N0 | N1 | N2 | N3 | N4 | N5 | N6 | N7 | N8 | N9 |
| O | O0 | O1 | O2 | O3 | O4 | O5 | O6 | O7 | O8 | O9 |
| P | P0 | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 |
| Q | Q0 | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 | Q9 |
| R | R0 | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 | R9 |
| S | S0 | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 |
| T | T0 | T1 | T2 | T3 | T4 | T5 | T6 | T7 | T8 | T9 |
| U | U0 | U1 | U2 | U3 | U4 | U5 | U6 | U7 | U8 | U9 |
| V | V0 | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 | V9 |
| W | W0 | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 |
| X | X0 | X1 | X2 | X3 | X4 | X5 | X6 | X7 | X8 | X9 |
| Y | Y0 | Y1 | Y2 | Y3 | Y4 | Y5 | Y6 | Y7 | Y8 | Y9 |
| Z | Z0 | Z1 | Z2 | Z3 | Z4 | Z5 | Z6 | Z7 | Z8 | Z9 |

# REPLACEMENT

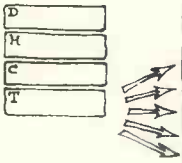
YOU MAY USE EACH VARIABLE MANY TIMES.

PUTTING A NUMBER INTO A VARIABLE SIMPLY REPLACES THE ONE ALREADY THERE. COMPARE THE FOLLOWING TWO PROGRAMS; THE FIRST IS THE ONE ON PAGE 3 WITH NEW LINE NUMBERS:

```
10 REM  A PROGRAM IN BASIC
20 DATA  6.5, 27, 236
30 READ  D, H, C
40 LET  T = 3.14 * D↑2 / 4
50 LET  S = 3.14 * D * H
60 LET  A = T + S
70 LET  G = A/C
80 LET  R = INT(G+1)
90 PRINT "YOU NEED"; R; "POTS"
100 END
```




THE SECOND DOES THE SAME JOB BUT USES VARIABLE T OVER AND OVER AGAIN:




```
10 REM   TO ILLUSTRATE REPLACEMENT
20 DATA   6.5,    27,    236
30 READ   D,      H,     C
40 LET   T = 3.14 * D↑2/4
50 LET   T = T + 3.14 * D * H
60 LET   T = T/C
70 LET   T = INT (T+1)
80 PRINT "YOU NEED"; T; "POTS"
90 END
```

WHEN A PROGRAM STARTS RUNNING EVERY VARIABLE HAS SOME INITIAL VALUE; IT DEPENDS ON THE VERSION OF *BASIC* WHAT THAT VALUE IS. IN SOME *BASICS* IT IS **ZERO**; IN SOME IT IS WHATEVER THE PREVIOUS PROGRAM LEFT BEHIND; IN OTHERS IT IS A SPECIAL SIGNAL TO SAY **VARIABLE NOT SET.** SO IN THE FOLLOWING COMPLETE PROGRAM:

```
10 PRINT  V
20 END
```

YOU MIGHT GET **0** OR RUBBISH LIKE −123.456 OR A MESSAGE FROM THE COMPUTER TO SAY IT CANNOT GO ON BECAUSE VARIABLE V IS NOT SET WHEN FIRST USED.

**DON'T** RELY ON ANY VARIABLE BEING ZERO WHEN THE PROGRAM STARTS; SET IT FIRST:

```
10 LET  V=0
```


*BASIC* CAN HANDLE *WORDS* AS WELL AS NUMBERS. WE SAW THIS ON PAGE 4.

```
9 PRINT "YOU NEED"; R; "POTS"
10 END
RUN
YOU NEED 3 POTS
```

THE "YOU NEED" AND THE "POTS" ARE CALLED *TEXTS* IN THIS BOOK. OTHER TERMS IN THE JARGON ARE: *STRING*; *LITERAL STRING*; *ALPHAMERIC STRING*; *ALPHAMERIC LITERAL*; AND THERE MAY BE MORE.

TEXTS ARE WORDS OR SENTENCES OR ARRANGEMENTS OF CHARACTERS ENCLOSED IN QUOTATION MARKS. BY THIS DEFINITION YOU CAN'T HAVE A TEXT CONTAINING QUOTATION MARKS BECAUSE THE COMPUTER WOULD THINK THEY MARKED THE END OF IT; BUT YOU CAN HAVE APOSTROPHES IN TEXTS:

```
20 PRINT "IT'S EASY"
```

HOWEVER, SOME *BASIC*S ALLOW TEXTS TO BE ENCLOSED BETWEEN APOSTROPHES AS AN OPTION; SUCH TEXTS MAY HAVE QUOTATION MARKS IN THEM BUT NOT APOSTROPHES. OTHER *BASICS* GET ROUND THE PROBLEM BY TREATING A *PAIR* OF QUOTATION MARKS INSIDE A TEXT AS SIGNIFYING A SINGLE QUOTATION MARK:

```
30 PRINT "SHE SAID ""OOH!"""
```

PRODUCES:

```
SHE SAID "OOH!"
```

BUT IT IS BEST TO AVOID HAVING QUOTATION MARKS IN TEXTS.

SEMICOLONS IN THE "PRINT" STATEMENT MAKE THE COMPUTER ABUT THE THINGS TO BE PRINTED ONE AGAINST THE OTHER; COMMAS WOULD MAKE THE COMPUTER SPREAD THEM OUT ACROSS THE PAGE. ALL THIS IS EXPLAINED FROM PAGE 28 ON, WHERE THE "PRINT" STATEMENT IS EXPLAINED IN DETAIL.

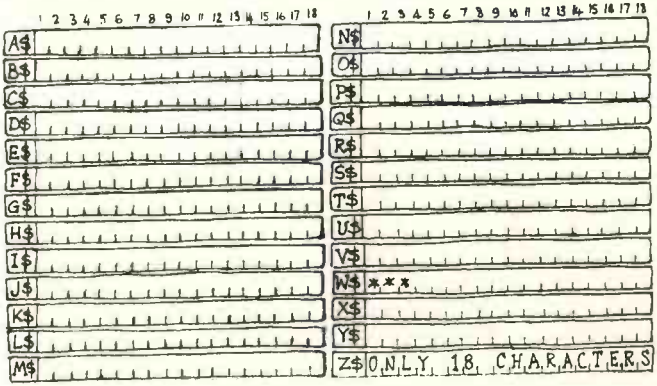
TEXTS IN THE "PRINT" STATEMENT MAY BE OF ANY LENGTH THAT WILL FIT THE LINE BEING TYPED. IF YOU WANT SOMETHING PRINTED RIGHT ACROSS THE OUTPUT PAGE YOU MUST PRINT TWO OR MORE TEXTS; ALL BUT THE LAST HAVING A SEMICOLON AFTER THEM.

```
120 PRINT "----------";
130 PRINT "----------";
140 PRINT "----------"
```

# TEXTUAL VARIABLES

IN ADDITION TO THE 286 LITTLE BOXES FOR STORING NUMBERS THERE ARE AT LEAST 26 FOR STORING TEXTS; THESE ARE CALLED *TEXTUAL VARIABLES*. WE HAVE COINED THIS TERM TO BALANCE *NUMERICAL VARIABLE* BUT OTHER TERMS IN THE JARGON ARE: *TEXT VARIABLE, STRING VARIABLE & LITERAL VARIABLE*.

EVERY *BASIC* HAS A LIMIT TO THE LENGTH OF TEXT THAT CAN BE STORED IN A TEXTUAL VARIABLE: SOME ALLOW AS MANY AS 4095 CHARACTERS; OTHERS AS FEW AS 18 ≈ A MAXIMUM FOR "PORTABILITY".




SOME *BASICS* HAVE 286 TEXTUAL VARIABLES: A$, A0$, A1$, A2$. *etc.* to Z8$, Z9$.

WHEN THE PROGRAM STARTS, TEXTUAL VARIABLES MAY CONTAIN BLANKS; OR TEXTS LEFT OVER FROM A PREVIOUS PROGRAM; OR SPECIAL SIGNALS TO SAY *VARIABLE UNSET*. AS WITH NUMERICAL VARIABLES (PAGE 10) IT DEPENDS ON THE *BASIC* BEING USED. SET VARIABLES BEFORE READING FROM THEM IF YOU WANT TO WRITE A "PORTABLE" PROGRAM.

TEXTUAL VARIABLES MAY BE SET BY "LET" STATEMENTS:

```
10  LET  W$ = "***"
20  LET  Z$ = "ONLY 18 CHARACTERS STORED"
```

AND PRINTED USING "PRINT" STATEMENTS:

```
30  PRINT  W$; Z$; W$
40  END
RUN
***ONLY 18 CHARACTERS***
```

LIMIT TO FIT INTO TEXTUAL VARIABLE

TEXTUAL VARIABLES MAY ALSO BE SET BY "READ" (PAGE 16) AND BY "INPUT" (PAGE 18). THEY MAY BE COMPARED BY "IF" (PAGE 41).

```
50  IF  I$ = "YES"  THEN  80
60  IF  I$ = W$     THEN  9000
```

# BOOK REVIEWS

**The BASIC book business is hotly contested. Preparing for this review, we found more than a dozen widely-available texts from a variety of publishers, ranging from hobby computer firms to conventional hard-back houses. Among computer languages, Basic is relatively easy to learn; it is also widely available. It will almost certainly be one of the languages, probably the only one, offered with any small system you consider.**

**We used two reviewers; one was an experienced computer user, consultant and teacher of dp concepts, the other a newcomer, with a very recent low-level introduction to the subject of computing. The books we chose illustrate the several different styles available to the Basic book writer; there are many other titles, of course, and we would be glad to hear of any personal preferences from readers.**

**ELEMENTS OF BASIC**
**by R. Lewis and B. H. Blakeley**
**published by NCC Publications Ltd, 1972**
**cloth cover: A4**
**price £3.**

IN SUCH a competitive market, there are a number of ways of selling a book. You might commission a well-known author from another field. You might be a well-known publisher yourself. Or you can rely on gimmicks.

The NCC *Elements of BASIC* falls into the second category. One could reasonably expect it to sell a large number of copies simply because it is published by the National Computing Centre. The book is fairly conventional, an A4 paperback, with the slight difference that it is laid out sideways, giving two columns to a page.

## Simple in approach

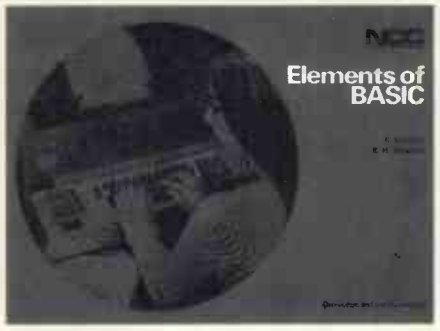
In some places, the format is used to good effect—for example, by listing a program in the right-hand column with the accompanying text in the left. It might have been advantageous further to exploit this feature.

Its introduction proclaims that "a great deal of attention has been given to its design so that it can be used as part of a course in computing at secondary school level". The book is essentially simple in approach and could be of use for self-teaching; this is helped by the extensive use of flowcharts for sample programs.

Similarly, the authors try to adopt a friendly stance, involving much use of the personal pronoun. "Some BASIC systems require us to reserve space . . . we have seen that . . . you may have tackled problem 1c. 3 earlier . . .".

As a style of English, this is frequently condescending and occasionally irritating. Neither characteristic improves the appeal or the lucidity of the book.

How about this? "If it is desired to have the numbers printed in descending order, you may change statement 140 . . .". What the writer means by this confusion of the personal and impersonal




is something like "if you want the numbers printed in descending order . . .".

The book is also misleading at times. "Statements are stored in the computer at specific addresses (their statement number) . . .". Well, the first part of this is self-evident; and statement numbers do not relate to specific addresses.

Elsewhere there is the instruction LET A=A+2 with the comment "notice that '2' is not contained in a storage location but is generated by the computer when required". By this the authors mean that '2' is not a variable and it is not an address; their phrasing, however, leaves plenty of room for confusion.

Similarly, there are points which pass unexplained. On string-handling, the book states correctly that BASIC imposes a limit on the number of characters allowed in a string; it then says that "we shall restrict ourselves to 40 characters" with no explanation of this, apparently, arbitrary limitation.

## Likely to confuse

In fact, the maximum number of characters in a string differs from one implementation of the BASIC language to another. Some allow as many as 4,096 characters, others as few as 18. Alcock's book points out that if you adopt 18 as the upper limit, you will be learning a use of character string which will apply to any BASIC.

The 40-character restriction would be acceptable, if explained, provided the authors concentrated on the same system throughout the book. Unfortunately, that is not the case. One chapter states specifically that all the programs in it were run on different computers; at best, that is likely to lead to confusion.

Also likely to confuse are statements like "on the system used to run the program, a # is printed for a %". In a book which seems to have had a fairly expensive production, it would have needed little effort to edit the program so that the student was not required to remember the alteration.

The book uses program listing taken directly from the computer printout. This

*(continued on next page)*

(*continued from previous page*)
has two real advantages. First, it is clear precisely what that unspecified computer does when running the BASIC program; and, second, it breaks up the text in an interesting way.

To fit the page format, however, the listings are reduced in size—and the quality of the reproduction is not always of the best, which again makes them difficult to read.

There are several relatively minor points on which the book may be criticised but there are also some very helpful features which should be commended. For example, indenting is often used in the program listing examples to emphasise the program structure—but sadly not in all the examples.

The tricky question of file handling, which can vary dramatically from one BASIC to another, is dealt with by a chapter covering "the main ideas, problems and flowcharts", while some specific BASIC implementations are treated in an appendix. The authors picked six simple file handling programs and show how they are coded in nine different BASICS.

## Could be updated

Only two of those can legitimately be described as minicomputers—the HP 2000 and CTL Modular One—and both machines have now been superceded by newer models. The other implementations are on two time-sharing services (IBM and Honeywell), two ICL mainframes (1900 and System 4), one deleted system (Xerox Sigma), the Burroughs B700 (now virtually displaced by the newer B800), and the DECsystem-10.

As illustrations of alternative BASICs, this multiple-machine appendix is interesting, particularly as the authors' annotations are useful and to the point, but the computer business moves quickly, and this book dates from 1972. Some more up-to-date examples might be more valid.

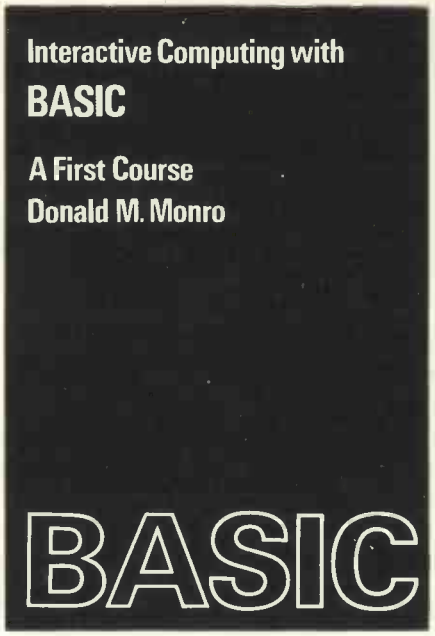
## CONCLUSION

● **Overall we found the book competent but boring, generally worthy but often patronising, concerned to make the instruction comprehensive rather than effective.**

**INTERACTIVE COMPUTING WITH BASIC – A FIRST COURSE**
**by Donald M. Munro**
**published by Edward Arnold Ltd, 1974**
**paperback: A4**
**price £3·25**

ANOTHER fairly conventional book. Designed originally for use by electrical engineering students at Imperial College, London, it has a very strong mathematical bias to its examples.

They show the power of the language




but it will deter many people. The sections on matrix operations are particularly impenetrable, and some readers would not wish to be able to solve the Laplace equation in BASIC. A lack of interest in the examples might well be reflected in the reader's attitude to BASIC itself.

That is a pity, since the book, in general, shows much evidence of thought and care in its preparation—for a start, there is a really handy list of BASIC statements inside the front covers and the language is summarised in a useful appendix.

## Style is terse

It was never written for business users and for them its inappropriateness is made more certain by the absence of a description of some language features which would be of particular use to such people.

The most important omission is some coverage of file handling—although the book deals with the use of a READ statement from DATA within the program. Similarly, little attention is paid to features of the BASIC system as opposed to those of the language. The general style is terse and the approach of the book definitely displays its origins. We cannot comment on its suitability as a textbook for scientifically-orientated undergraduates, which is undoubtedly its chief aim but we could not recommend it as an introduction to BASIC or a self-instruction text for business users.

## CONCLUSION

● **Conventional, competent, probably a good textbook for anyone who requires to learn the language for solving problems with a mathematical bias, but not for the small business system buyer.**

**INSTANT BASIC**
**by Jerald R. Brown**
**published by Dymax Inc, 1977**
**paperback; quarto**
**price £4·95 (from LP Enterprises Ltd)**

FROM the dustjacket blurb: "For the microcomputer enthusiast or the user of the Digital BASIC-Plus language, there is finally a book to teach you BASIC. No longer will you have to struggle through the reference manual provided with your computer.

"The author has a quiet writing style which slowly introduces new ideas in a pleasant, non-mathematical context. To offset this style of writing, he has used the zaniest, wildest graphics available, making the book a barrel of fun to read. INSTANT BASIC is a "active participation" workbook. The book is designed to use with your home computer readily available so you can do it!

"You'll like this book. It's designed for beginners, covers most of the language and is fun to use".

Well, that is a matter of opinion. This book derives directly from the home of the hobby microcomputer, California; and its publishers have links with one of the pioneering propagandist organisations there, the People's Computer Company—incidentially, the PCC monthly magazine *People's Computing* is required reading for small-computer enthusiasts.

## Welcome change

Dymax publishes a number of titles in the general area—like *Games with the pocket calculator*—excellent for school kids—and *Your Home Computer*—one of the calmest and sanest introductions to minicomputers we have seen, and highly recommended for the complete novice.

Dymax also publishes *My Computer Likes Me* (*when I speak in BASIC*). This is a conversational introduction to the language, a book of only 64 pages which is effective propaganda for BASIC and has no aspirations to be a textbook. *Instant BASIC* has aspiration to spare.

As the blurb indicates, this is a book written specifically for two really popular BASICs—the Digital RSTS/E implementation for the PDP-11, BASIC -Plus, and the very similar MITS/Altair BASIC offered on the 8080A-based microcomputers sold here by Compelec.

This is a welcome change in orientation from other textbooks, most of which seem to have been written by authors with experience only of large-machine, time-sharing BASICs.

The book definitely avoids a mathematical bias, which is also good, but there our enthusiasm ends.

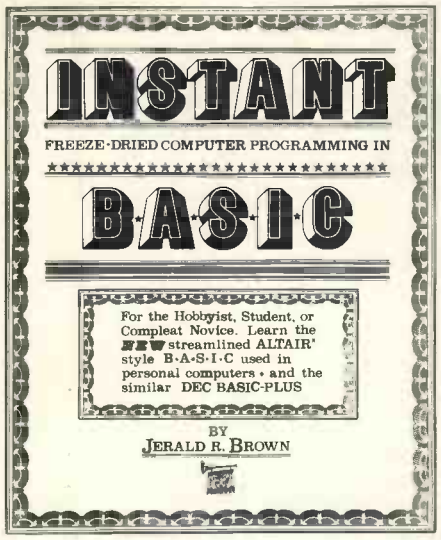
*Instant BASIC* is produced in that irritating, late-1960s style which encourages the designer to go wild with the Letraset catalogue. The text is sober enough but it is surrounded by a real

(*continued on next page*)

(*continued from previous page*)

welter of graphics—fancy borders, illustrations, line drawings and a variety of aggressive typefaces.

That approach does not work. It does not serve to break up the text in an inter-




esting and attractive manner, it does not make the book "fun to read" and it certainly does not make it easy to use.

In fact, *Instant BASIC* is not as bad as it sounds. It covers most of BASIC and it does so quite accurately—we found no obvious flaws in the text. It is aimed squarely at the novice—it even tells you in quite banal detail how to use a keyboard—and it applies to small computers; both attributes separate this book from many others.

In our view, though, the design and presentation of *Instant BASIC* are a serious misjudgment, and that title should not be taken too literally—the book has 158 pages, each of which must be read.

## CONCLUSION

- **Good try in terms of ambitions, an explicit and sometimes simply monotonous rendition of the Altair and Digital versions of BASIC, an irritating and ultimately unreadable presentation if you do not share its ideas on humour and style.**

**ILLUSTRATING BASIC**
**by Donald Alcock**
**published by Cambridge University Press, 1977**
**paperback, A5**
**price £1·50 (hardback at £4·95)**

WE LEFT the best until last. This book also adopts a somewhat gimmicky style—it is handwritten throughout, including the printout examples. On the other hand, this imparts an appealing quirkiness rather than the offensive self-congratulation which might have ensued.

Alcock's introduction does not seek to explain the absence of type-setting but it indicates a dry wit we liked: "My book is informal in language and unusual in presentation. Rather than write a justification I would remark only that a careful reader might diagnose a severe astigmatism in my eye and a persistent shake in my hand".

*Practical Computing* readers may already have deduced that the reviewers liked this book. It should be said, however, that handwriting can be tiring if you read much of it at a time—that might not be very likely, of course; and at least one of us took issue with Alcock's idiosyncratic hyphens—larger than the average printed dash, resembling a somewhat tipsy slug, and casting a fat shadow.

The pen, however, also gives the ability to use more graphic means of representing concepts than is usually the case. For example, a bug appears in text like a cross between a ladybird and an untidy spider—very effective.

The book is written at a fairly introductory level. As such it is clearly appropriate for the novice but it seems unlikely to satisfy one category at which it is apparently aimed, namely those that are already able to program reasonably well in Basic and who need a reference work. Nevertheless, even such people would gain some useful information about the language from this book, though an alternative like the NCC one might make a better reference textbook.

## Dialect problem

From the start the author makes the point that there are many versions of the BASIC language and that they are noticeably different. He says "I have accepted this as a fact of life and, in writing this book, kept at my side manuals, each defining a different BASIC".

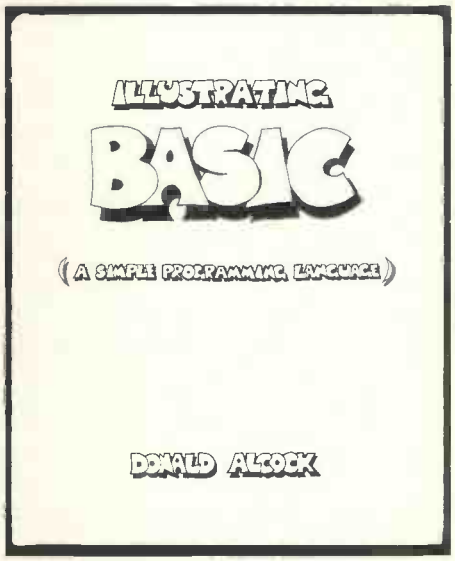
Discussing the language function, he has attempted to show the differences between dialects; this might result in the reader knowing all the differences between various dialects, while being unable to program in any one of them. An alternative approach might have been to concentrate on one version, perhaps pointing out the various differences in an appendix.

What Alcock does, in fact, is to present a truly portable BASIC as far as he can. In particular, where there are differences in scope he will pick the interpretation which covers most of the options. For instance, as we noted earlier there is considerable variety in just how many characters you can have in a string. *Illustrating BASIC* points out that if you assume 18 is the maximum, you will be writing BASIC programs with text strings which should run on most machines.

There are few technical errors in the text and we had to search to find any inadequate statements. Here is one, though: "In some BASICs, the biggest number which can be stored is approximately $\pm 10^{38}$". This is meaningless unless related to the word size of the computer.

More generally, we are slightly suspicious of the scientific/engineering bias in some of the application examples. True, the deflection of a diving board with a diver on it is followed immediately by a mortgage loan calculation; and GOSUB is illustrated both by finding "the highest common factor of two numbers by Euclid's method" and by a neat little Mastermind-type game called 'Moo'—we know it as 'Cows and Bulls'.

And, after all, the majority of BASIC users are probably outside the commercial field in any case. Alcock should not be




criticised for attempting to satisfy everyone if he succeeds—and, by and large, he does.

Arrays and matrices are probably the most difficult aspect of BASIC for the novice business system user to grasp. Alcock gives plenty of description on arrays without saying what difference there is between an element in an array and a 'simple' variable. That is not too bad, in fact—the relationship becomes clearer with practice.

Matrix algebra can, however, become tricky. As Alcock says, "In BASIC a matrix is simply a rectangular array of subscripted variables . . . Don't run away—you don't have to know matrix algebra to find MAT statements useful". In principle he is right, and his treatment of this area is as lucid as any we have seen.

On the other hand, MAT arithmetic and knowledge of matrices in general are by no means essential to writing business-orientated programs.

It is refreshing to find a BASIC text which takes this much care over the needs of readers, rather than stopping short at the nature of the language.

## CONCLUSION

- **Humane, interesting, comprehensive, and—in paperback form, at least—excellent value for money. Congratulations to the author. Very highly recommended.** ■

# Hop on a Nasbus to 32K of memory now

The Nascom – 1 is designed with expansion in mind. This is made possible by using the best products available. The Z80 microprocessor incorporated in the basic system is so powerful it can support 64K bytes of memory and 256 ports. To utilize this capability, we have designed the buffered 77 – way Nasbus.

With this arrangement, the way is clear for considerable expansion, starting with our new memory expansion board. It has 16 memory sockets and two EPROM sockets. Therefore, you can fill it with 4K dynamic RAM up to a maximum 8K or with 16K dynamic RAM up to a maximum of 32K. A 2K Tiny BASIC in EPROM has been developed for the board.

To go with the board, we have produced a very flexible I.O board with three PIOs each giving two, 8bit ports, plus a UART for serial interface.

As you start building up your Nascom system you will need a convenient means of storing boards. Our new, custom-designed, Vero frame will allow for a Nascom – 1 to link through a buffer board to a 77-way Motherboard. There is then the option of eight or more expansion boards. To power this capability there is a new 8.5 amp power supply especially designed for the frame.

No other system offers so much at such a low cost. And it all starts with the basic Nascom – 1 kit which for just £197.50 offers an intelligently usable system with video and cassette interface, a full alpha-numeric keyboard and a mighty CPU chip. So if you want the best – make it a Nascom system.

## Nascom-1 Kit still only £197.50 +VAT

## Stop press...

**Microcomputer Seminar**

Nascom Microcomputer's highly successful seminar is coming to Bristol. The programme will be similar to London and Manchester, both of which were sold out. The day includes five lectures, demonstrations and an open forum. Venue is the Dragonara Hotel, Bristol, Saturday, October 14th, 09.50 to 17.30.

**Admission:** £4.50 (inc. VAT). Lunch will be available at £4.00 (inc. VAT) per head if there is sufficient demand.

**Nascom Distributors**

**Camera Centre,** Barrow-in-Furness, Cumbria
**Crystal Electronics,** Torquay, Devon
**Electrovalue,** Egham & Manchester
**Eley Electronics,** Glenfield, Leicester
**Henry's Radio,** London W2
**Lock Distribution,** Oldham, Lancs
**Lynx Electronics,** Chesham, Bucks
**Microdigital,** Liverpool L2
**Teleplay,** New Barnet, Herts

Nascom Microcomputers

**Please send me:**

| | |
|---|---|
| .....Buffer Board Kit/s at | £25.00 each |
| .....4K RAM Board/s at | £70.00 each |
| .....16K RAM Board/s at | £179.00 each |
| .....4K RAM (4027) at | £35.00 each |
| .....16K RAM (4116) at | £106.00 each |
| .....Tiny BASIC/s at | £25.00 each |
| .....Nascom–1 Kit/s at | £197.50 each |
| .....2.2A power supply/ies at | £24.50 each |

*Add VAT at 8% to all above prices. (P+P not inc.)*

.....Bristol Seminar Tickets at £4.50 each

I am/am not interested in a frame for my Nascom–1.

I enclose a cheque/P.O. for £ ..... made payable to Nascom Microcomputers, 92 Broad Street, Chesham, Bucks. Tel: (02405) 75151

Name _____

Address _____

_____

Tel. No _____

PC 978

● Circle No. 132

# TAKING THE CHORE OUT OF VAT

**Probably one of the most disliked and time-consuming of the statutory chores imposed on businessmen is the calculation of Value Added Tax returns. Accounting for VAT was thus a natural application for one of the low-cost, microprocessor-based computers. A computer will not abolish the amount of work but it will reduce it substantially, with the bonus of increased accuracy and clarity of records.**

**This is the first of two articles which will contain a complete suite of programs written to handle VAT accounting. They are written in TDL Basic but can, with minor modifications, be run with most versions of Basic available on small computers.**

BEFORE discussing the design and use of these programs it is necessary to point out that they have certain limitations which all potential users must bear in mind. The most important is that they have not been cleared by Customs and Excise as a computerised VAT system.

By virtue of the audit trails and transaction listings produced on hard copy, however, they do not have to be authorised; they represent one method among many for accounting for VAT. The second limitation is that although the system can handle credit notes, imports and exports, as well as normal sales and purchases, it has not been designed to cope with any of the retailers' VAT schemes.

## Time-consuming

The author's business deals with retail sales of microcomputer equipment, mainly complete systems. Therefore some method of recording the appropriate details was needed, namely tax point date; some method of identifying the transaction back to the invoice; an indicator to signify whether to add, delete or change the record in question; the type of transaction—sale, purchase, import; the VAT rate applicable; the values involved—VAT, gross & nett value; some indicator as to the VAT return classification.

As I do not participate in one of the retailers' VAT schemes, I needed only to account for VAT on a transaction-by-transaction basis, with some method of reading all the records for a chosen VAT period and deriving the figures required for my VAT return—probably the most time-consuming part of VAT accounting.

by
**GEOFF LYNCH**

Before you can claim VAT relief on a purchase you need to be registered; the purchase must be wholly as a result of your business needs; and the supplier's invoice must show the minimum of information as required by law.

The system does not hold all this information on the computer, as the key to the system is a number held on the VAT transaction record and written on the original invoice. Thus an audit trail exists, both forwards from the original, into the system, and backwards from the system to the original documents.

The programs listed in this article are written in a pre-release version of TDL Disc Basic for use under the Digital Research operating system CP/M on Z80 systems. The minimum requirements are the provision of an operating system and Basic which allows sequential disc file accessing of up to three files at once, console device, a list device and memory sufficient to allow an internal memory sort of the largest transaction file you will ever create in one run.

## Minimum of change

To use these listings with the minimum of change a good extended Basic will be required—North Star, Microsoft, Cromemco. I use a Lear Siegler ADM-3A, PerSCI dual discs running under CP/M, 48K of memory, and an old Teletype model 33 as the list device.

The current system has no automatic control of file versions, although this would be relatively easy to add. You are required to keep a log of which files are what. Also, due to my business being relatively small in terms of individual transaction value, no single transaction with a gross value larger then £9,999·99

*(continued on next page)*

## Welcome to Computabits

***Practical Computing* is pleased to announce that it has acquired the publication *Computabits*. It will be run each month in a special section and will continue to be edited by Nick Hampshire.**

**Existing subscribers to *Computabits* will now receive *Practical Computing* each month at no extra charge.**

**The style of *Computabits* will continue. "It is a forum to deal with the need for exchanging information in the rapidly-expanding field of microcomputers", says Hampshire. "I am confident that *Practical Computing* will meet this requirement and I am enthusiastic and optimistic about the new opportunities available to *Computabits* as a result of its incorporation".**

*(continued from previous page)*

can be handled, nor can a negative—credit note on a sale—value of less then £999·99 be handled. This is due primarily to the line width of the TTY 33. Invoices with more than one rate of VAT are required to be entered as separate transactions for each different rate. A look at the program listings will reveal that changing the value field sizes is straightforward, although one wants to be careful about the 'print using' statements.

## Four programs

The system consists of four programs in the old-fashioned classic system architecture—a data entry and format validate program, a sort, an update master file program and finally the report program. The data entry program collects data entered by the user from invoices, performs validation on fields, outputs to both the list device and to a disc file. Under CP/M many transaction files can be created and concatenated into one prior to the sort using PIP— Peripheral Interchange Program, a CP/M utility.

Thus as long as you keep a record, data entry can be performed as often as required. The sort uses the file created by the data entry program as input, reads all the records into memory, and sorts them into ascending order based on the 13-byte key (more of which later).

Once the sort is completed, the records are output to a file with the same name as the input, but with an extension of 'new'. A previously-created master file is input to the update program, along with the sorted transaction file. Records are added, deleted or replaced as appropriate. As this is being done, a listing is produced of the actions being taken and this provides the audit trail so necessary in accounting. As in all the programs, control totals are produced for clerical reconciliation.

## Cassette changes

The listing of the update should be perused by the user to spot if any errors have occurrred and, if so, they should be corrected by creating the appropriate transactions using the data entry program and performing the sort, update cycle again. Once satisfied that all data which should be on the master file is present—i.e. all that period's invoices—and it is all correct, the report program may be run.

The report program asks for the file name, as do the other programs, and then asks for the dates in which the period falls. The dates must be correct for the Vatman to be happy. The master file is read and the data accumulated to produce a report in VAT return format. Once again, the controls should be reconciled and, once done, the report transcribed on to the return, signed and posted.

For those with only cassette drives, programs one and four can be amended by you to run on cassettes, and this would give some assistance to VAT record-keeping. You would lose the ability to sort and update, unless you had three cassette drives but you could ensure that all one period's transactions went on the transaction file, which the report program will work from with little amendment.

TDL Basic supports print and input statements to the console—device #0 or the default device;—a list device— & device #2; a reader—device #3; a punch—device #4; and disc files—device #S 5 to 255. Device #1 is reserved for the program load/save device—normally a disc file. The disc files require to be explicitly opened.

Open #<Unit>, <Mode>, <File Name> where unit is number, mode is a string with the value of "I" for input, "O" for output, "R" for random mode and "U" for update in place mode. <File Name> is any CP/M acceptable filename string. The file is then accessed by reference to its unit number in print or input statements, which are ASCII mode commands.

## Ease of testing

Internal format storage is also supported—via write & read statements—but I have used ASCII for ease of testing and cross-machine support. Once a file has been opened, if input, an EOF statement is required to direct action at end of file; e.g., 1200 on EOF #5 GOTO 1340.

In this system all disc data is handled as string variable records of 63 bytes length (not including the CR, LF). The record size is not required by the system as it uses the CR, LF or the 'quote marks' to delimit the records. Thus to read a record:

```
1210 INPUT ##5, TL$
```

This would input the next record into string variable TL$, where I can then dissect it, using the MID$ and VAL (QV) functions.

An output file must be closed to ensure that the end of file marker is written and CP/M marks its own EOF on the file.

This is accomplished by

```
1640 CLOSE #5, #6
```

This would close files 5 and 6. An all-embracing close is available, which is close without any reference to a unit. The clear statement has an I/O function not normally seen. A clear statement under TDL Disc Basic has a second argument, e.g:

```
CLEAR 3000, <N>
```

The <N> is the total number of disc files which will be open at any one time. This is required to reserve space for the file control blocks needed under CP/M. TDL Disc Basic has numerous other I/O options but as none of them is invoked I will desist from a tedious explanation.

TDL Basic has an extensive repertoire

*(continued on next page)*

(*continued from previous page*)

of 'print using' features, but the only one which must be explained is the 'LLLLLL feature. Where a 'L - - - - - L appears, a string variable will be left justified to the start at the ' . If the string is too long to fit, it will be truncated on the right; if too short, it will be filled on the right with spaces. The # # # # # . # # is as in other Basics.

Within the Basic program a file can be erased or re-named and the presence of a file determined. The formats of these statements are as follow:

```
ERASE <FILENAME>
RENAME <OLD FILENAME> <NEW FILENAME>
X=LOOKUP (<FILENAME>)
X=0 IF FILE PRESENT, -1 IF NOT PRESENT.
```

**PROGRAM ONE, TRANSACTION CREATE.**

| | |
|---|---|
| 100 – 660 | PROGRAM INITIALISATION AND FILE SET UP. |
| 670 – 1120 | MAIN LOOP OF PROGRAM FLOW. |
| 670 – 1040 | DATA ENTRY AND VALIDATION, VAT CALCULATION. |
| 1050 – 1090 | USER CONSOLE CHECK OF DATA ENTERED. |
| 1095 – 1120 | & |
| 1430 – 1710 | DISC RECORD CREATE, LIST TO AUDIT TRAIL OF RECORDS, TOTALS. |
| 1130 – 1330 | CONTROL TOTAL PRINT AT END OF JOB, END OF PROGRAM. |
| 1340 – 1410 | CONSOLE PRINT OF COMPLETED RECORD. |
| 1730 – 1800 | LIST DEVICE HEADINGS ROUTINE. |

In the two example outputs accompanying the program listing, user input is underlined. Example 1 is what would be the console print if a TTY 33 was used as console. Example 2 is the corresponding list device output for the same run. The effect of using a TTY as console is to negate the advantages of using a high-speed VDU such as a Lear Siegler ADM3-A.

## Invoice code

The first two entries on Example 1 are the date of the run—which appears on the list device headings—and the entry of the filename to be created. This must have an extension, such as 'UNS', if the file is to be used in the CP/M environment, as this is needed by program two.

The entries starting at point 1 consist of the tax point date, the 'tran code' and a single digit number within the 'tran code'. The date must be as shown, namely 'YYMMDD'. the 'tran code' is the number you must associate with every invoice. In my system, my invoices are numbered with a five-digit number, so this is used as the 'tran code' for sales—the code must be five digits long.

For purchases, normally use the purchase order number, as suppliers normally quote this on invoices, and again this is a five-digit number. It is this number which must be on each document you are submitting a VAT return in respect of, even if you have to write it on the document yourself. The serial number (single digit) is to allow multiple entries per invoice for different VAT rates. It must be a number but the program could be amended to allow any character.

A 'null' entry for the tax-point date will initiate the end-of-job process. In most Basics typing a comma followed by zero, comma, zero and return will be sufficient.

The next set of data entered consists of a single character to indicate if the transaction is to add, delete or replace a record on the master file. This is followed by up to 23 characters of 'comment'—type data—normally supplier's name and invoice number, or customer's name. Not all this data is printed, but all 23 characters are held on the record. If this system is to be interfaced to an accounts-payable and/or accounts-receivable system, this field can be used for account coding information.

The last entry indicates whether the transaction is a normal sale(s), a normal purchase(p), or one of the 'specials'. A look at the corresponding entry for transaction 5 will show the codes available.

If a 'D' was entered as the transaction type, no more data will be requested from the user, as the rest of the fields will be filled-in for him, as a delete does not require values. Transaction 4 is an example of such a transaction.

## Calculates values

The third set of data requested is the VAT rate—S for standard, A for higher rate, and Z for zero rate. These rates are held in the program against the code indicated. A change in VAT rates will require a simple amendment at statements 970–990. Secondly, the user must indicate if the gross value or the nett value is about to be input. This is done by entering 'G' or 'N', The appropriate value is then entered. The program then calculates the VAT and the value not entered.

The complete transaction is then displayed on the console device for the user to check and, if necessary, re-input to correct. A carriage return entered in reply to the 'Return if OK? will complete all action on the transaction by printing it on the list device and outputting it to disc. If not OK, enter anything else and you will be led through the data entry sequence again from the beginning.

Example 2 is the corresponding list device output for the same run. Note the control totals printed at the end of the job. Both the transaction listing and the controls should be kept in a secure place for use at the next stage, the sort.

For those unfamiliar with VAT terms, be very careful with transactions which attract no tax. They may be zero-rated or they may be exempt—there is a difference. Refer to the appropriate VAT Guide (HM Customs & Excise Notice No 700).

**PROGRAM TWO, THE SORT.**

| | |
|---|---|
| 10 – 380 | PROGRAM INITIALISATION AND FILE SET UP. |
| 390 – 490 | INPUT RECORDS AND STORE IN STRING ARRAY. |
| 500 – 580 | END OF INPUT FILE. |
| 590 – 870 | SORT LOGIC. |
| 880 – 1000 | OUTPUT RECORDS FROM SORTED ARRAY. |
| 1010 – 1100 | CLOSE OUTPUT FILE AND END OF JOB CONTROLS. |

(*continued on next page*)



## VAT TRANSACTION & MASTER FILE LAYOUT

| START POS | NO. BYTES | VAR NAME | DESCRIPTION |
|---|---|---|---|
| 1 | 6 | TD$ | TRANSACTION TAX DATE (YYMMDD) |
| 7 | 5 | TC$ | DOCUMENT AUDIT CODE NUMBER |
| 12 | 1 | S$ | SERIAL DIGIT WITHIN DOCUMENT |
| 13 | 1 | TT$ | TRANSACTION TYPE (A=ADD, D=DELETE, R=REPL) |
| 14 | 23 | CM$ | COMMENT FIELD, USER DEFINED |
| 37 | 1 | I$ | INDICATOR AS TO INPUT OR OUTPUT VAT |
| 38 | 1 | ST$ | INDICATOR AS TO VAT RETURN CLASS, SEE * 1 |
| 40 | 4 | V1 | VAT RATE, 2 IMPLIED DEC PL (NNVNN) |
| 44 | 7 | G | GROSS TRANS VALUE, 2 IMPLIED DEC PL |
| 51 | 7 | N | NET TRANS VALUE, 2 IMPLIED DEC PL |
| 58 | 6 | V2 | VAT TAX VALUE, 2 IMPLIED DEC PL |

* 1
VALUES FOR ST$ ARE AS FOLLOWS:-
0 = EXPORT SALE CLASS
2 = DEFERRED ACCOUNTING VAT IMPORT
3 = VAT DUE ON UNDERDECLARATIONS, NOTIFIED BY CUSTOMS
4 = VAT DUE ON OTHER UNDERDECLARATIONS
6 = VAT REFUNDABLE DUE TO ASSET PURCHASE
7 = VAT DEDUCTIBLE ON OVERDECLARATIONS, NOTIFIED BY CUSTOMS
8 = VAT DEDUCTIBLE ON OTHER OVERDECLARATIONS
S = NORMAL TAXABLE SUPPLY (SALE)
P = NORMAL TAXABLE PURCHASE (BOTH THESE MAY BE ZERO RATED)
E = EXEMPT TRANSACTION (SALE)

*(continued from previous page)*

1130 – 1240 INPUT FILE CONTROL TOTALS ROUTINE.
1260 – 1380 OUTPUT FILE CONTROL TOTALS ROUTINE.
1400 – 1480 PRINT ROUTINE FOR INPUT FILE CONTROLS.
1500 – 1580 PRINT ROUTINE FOR OUTPUT FILE CONTROLS.

This program is relatively straightforward and inspection of the program listing will show the logic flow. Only two things need be noted, the use of 'INSTR' and the re-naming of the input file.

'INSTR' is a TDL Basic feature which is an instring function. The first string argument is searched for the occurrence of the second string argument and its character position, if found, is returned.

If the second string is not located a zero is returned. Therefore, as used in this program, the filename is searched for A '.' (line 270). This is expected as the program is being run under CP/M and a file extension is expected. If not found, the program cannot be run.

This is being used in conjuncation with the 're-name' feature—see explantion in program one. To change the name of the input file in

LINE 890. THE NAME CHANGE LOGIC IS AT LINES 290–310.

The logic for the sort was copied from *The Best of Creative Computing, Vol. 1.*

*Next month's article will provide the listings of programs three and four, the update and report programs, as well as the flowcharts and running instructions.*

### EXAMPLE 1, SIMULATED CONSOLE OUTPUT FROM VAT TRANSACTION CREATE

```
RUN
VAT TRANSACTION CREATE PROGRAM
PLEASE ENTER TODAYS DATE YY/MM/DD? 78/07/10
ENTER FILENAME OF FILE TO BE CREATED?
  VATTES01.UNS
TRANS DATE(YYMMDD), TRAN CODE, SERIAL #?
  780510, 78021, 1
A,D, OR R; USER INFO, S P OR X? A,CUST 1243, S
VAT RATE(S A B OR Z), G OR N, VALUE? S, N, 195

TRANSACT   TRANS    S  T        USER
DATE       CODE     #  C        COMMENTS
78/05/10   78021    1  A        CUST 1243

I  X      VAT      GROSS     NETT      VAT
O  C      RATE     VALUE     VALUE
O  S      8·00     210·60    195·00    15·60
'RETURN' IF OK?__
TRANS DATE(YYMMDD), TRAN CODE, SERIAL # ?
  780512, 78034, 1
A, D, OR R; USER INFO, S  P  OR X?
  A, PAY WILLIS-INV 15647, P
VAT RATE(S A B OR Z), G OR N, VALUE?
  S, G, 34

TRANSACT   TRANS    S  T        USER
DATE       CODE     #  C        COMMENTS
78/05/12   78034    1  A        PAY WILLIS

I  X      VAT      GROSS     NETT      VAT
O  C      RATE     VALUE     VALUE
1  P      8·00     34·00     31·48     2·52
'RETURN' IF OK?__
TRANS DATE(YYMMDD), TRAN CODE, SERIAL # ?
  780602, 00001, 1
A, D, OR R; USER INFO, S P OR X?  D, DEL ORIG, S

TRANSACT   TRANS    S  T        USER
DATE       CODE     #  C        COMMENTS
78/06/02   00001    1  D        DEL ORIG

I  X      VAT      GROSS     NETT      VAT
O  C      RATE     VALUE     VALUE
X  X      0·00     0·00      0·00      0·00
'RETURN' IF OK?__
TRANS DATE(YYMMDD), TRAN CODE, SERIAL # ?
  780603, 78045, 1
A,D, OR R; USER INFO, S P OR X?
  R,AMEND ORIG, S
VATE RATE(S A B OR Z), G OR N, VALUE?
  Z, G, 100·15

TRANSACT   TRANS    S  T        USER
DATE       CODE     #  C        COMMENTS
78/06/03   78045    1  R        AMEND
                                 ORIG

I  X      VAT      GROSS     NETT      VAT
O  C      RATE     VALUE     VALUE
O  S      0·00     100·15    100·15    0·00
'RETURN' IF OK?__
TRANS DATE(YYMMDD), TRAN CODE, SERIAL # ?
  780612, 00120, 1
A,D, OR R; USER INFO, S P OR X? A, IMPORT, X
ENTER  2 FOR IMPORT, 6 FOR ASSET PURCHASE
       3/4 FOR UNDER DECLARATIONS,
       7/8 FOR OVER DECLARATIONS
       0  FOR EXPORT SALE
       E  FOR EXEMPT SALE
? 2
VAT RATE(S A B OR Z), G OR N, VALUE?
  S, 1245·68·U  S, G, 1245·65

TRANSACT   TRANS    S  T        USER
DATE       CODE     #  C        COMMENTS
78/06/12   00120    1  A        IMPORT

I  X      VAT      GROSS     NETT      VAT
O  C      RATE     VALUE     VALUE
O  2      8·00     1245·65   1153·37   92·28
'RETURN' IF OK?__
TRANS DATE(YYMMDD), TRAN CODE, SERIAL # ?
  ,0,0
END OF INPUT (Y/N)
? Y
END OF JOB
TOTAL RECS CREATED   =  5
TOTAL GROSS VALUE    =  1590·40
TOTAL NETT VALUE     =  1480·00
HASH TOTAL OF VAT    =  110·40
```

### EXAMPLE 2, LIST DEVICE OUTPUT FROM VAT TRANSACTION CREATE

```
78/07/10  VAT TRANSACTION CREATE PROGRAM
PAGE 1
78/05/10  78021   1  A  CUST 1234
78/05/12  78034   1  A  PAY WILLIS-INV 15
78/06/02  00001   1  D  DEL ORIG
78/06/03  78045   1  R  AMEND ORIG
78/06/12  00120   1  A  IMPORT

O  S   8·00   210·60    195·00    15·60
1  P   8·00    34·00     31·48     2·52
```

*(continued on next page)*

*(continued from previous page)*

```
X  X   0·00      0·00      0·00      0·00
O  S   0·00    100·15    100·15      0·00
O  2   8·00   1245·65   1153·37     92·28
  END OF JOB
FILE CREATED NAMED VATTES01·UNS
  TOTAL RECS CREATED  =  5
ADDITIONS  =  3    DELETIONS  =  1
REPLACEMENTS  =  1
  TOTAL GROSS VALUE   =   1590·40
  TOTAL NETT VALUE    =   1480·00
  HASH TOTAL OF VAT   =    110·40
```

## LISTING OF PROGRAM ONE, VAT SUITE VAT DATA ENTRY & TRANSACTION CREATE

```
100  REM VAT TRANSACTION CREATE PROGRAM
110  REM COPYRIGHT 1978 (C) XITAN SYSTEMS
120  REM WRITTEN BY G. C. LYNCH MARCH 1978
130  CLEAR 1000, I
140  S=0: TN=0
150  REM F$= FILENAME USED BY USER FOR
       OUTPUT FILE
160  REM D$= DATE FOR HEADINGS ETC
170  REM LC= LINE COUNT
180  REM PC= PAGE COUNT
190  REM TD$= TRANS DATE IN YYMMDD
       FORMAT
200  REM TC$= TRANS CODE IN YYNNN
       FORMAT
210  REM S$= SERIAL NO WITHIN TRANS CODE
       (1 NUMERIC DIGIT)
220  REM TT$= TRANS TYPE (A ADD, D DELETE,
       R REPLACE)
230  REM CM$= COMMENT OR ANY OTHER
       STRING (23 DIGITS MAX)
240  REM ST$= SALE, PURCHASE OR X FOR
       SPECIAL (SEE VAT RETURN)
250  REM R$= VATE RATE INDICATOR (S,A,B,
       OR Z FOR EXEMPT)
260  REM G$= GROSS OR NETT INDICATOR
       (G OR N)
270  REM G= GROSS VALUE
280  REM N= NETT VALUE
290  REM V= VAT VALUE
300  REM S= STANDARD VAT RATE (8·00%)
310  REM A= HIGHER VAT RATE (12·50%)
320  REM B=RATE NOT USED
330  REM Z= EXEMPT RATE OF VAT (0·00%)
340  REM TL$= STRING PORTION OF RECORD
350  REM I$= 1 FOR INPUT VAT, 0 FOR OUTPUT
       VAT
360  REM
370  REM
380  S0$="'"'"+STRING$("L", 37)
390  S1$=" #### ##### ##### ##### ####"
400  S2$="'"L/'L/'L ' CCCC ' ' 'LLLLLLLL ' '
       '##.## #####.## #####.## ###.##"
410  S3$="'"L/L/LL 'CCCC ' ' 'LLLLLLLLLLLLLLL
       ' ' ##.## #####.## #####.##
420  S4$=" TOTAL GROSS VALUE  =
       #####.##"
430  S5$=" TOTAL NETT VALUE   =
       #####.##"
440  S6$=" HASH TOTAL OF VAT  =
       #####.##"
450  S7$=S0$+S1$
460  PRINT: PRINT" VAT TRANSACTION CREATE
       PROGRAM "
470  PRINT: INPUT" PLEASE ENTER TODAYS
       DATE YY/MM/DD"; D$
480  PRINT: INPUT "ENTER FILENAME OF FILE
       TO BE CREATED"; F$
490  REM
500  REM
510  REM NOW OPEN FILE
520  OPEN #5, "O", F$
530  REM
540  REM
550  REM
560  REM INIT OTHER VALUES HERE
570  LC=80 : PC=0
580  TL$=SPACE$(38)
585  REM TOTAL ACCUMS INIT TO ZERO
590  TG=0 : TN=0 : TR=0 : TV=0
600  TA=0 : TD=0 : TC=0
610  REM
620  REM
630  REM
640  REM PRINT HEADINGS
650  GOSUB 1730
660  REM
670  REM MAIN LOOP
680  REM
683  REM DELETE NEXT STATEMENT IF YOU
       DON'T HAVE CURSOR CONTROL
       (ADM3A)
685  PRINT CHR$(30); CHR$(26);
690  PRINT: INPUT "TRANS DATE (YYMMDD).
       TRAN CODE, SERIAL # "; TD$, TC$, S$
700  REM IF NULL STRING INPUT AS TD$ THEN
       CHECK FOR END OF JOB
710  IF LEN(TD$)=0 THEN PRINT "END OF
       INPUT (Y/N)" : INPUT Z$ : IF LEFT$ (Z$, 1)
       ="Y" THEN 1140 ELSE 685
720  T=VAL(TD$) : X=VAL(S$)
730  IF S$ <"0" THEN PRINT "ERROR IN SERIAL
       NO" : GOTO 690
740  IF S$ >"9" THEN PRINT "ERROR IN SERIAL
       NO" : GOTO 690
750  M=VAL(MID$(TD$, 3, 2))
760  D=VAL(MID$(TD$, 5, 2))
770  IF M <1 OR M> 12 THEN PRINT "ERROR IN
       MONTH" : GOTO 690
780  IF D <1 OR D> 31 THEN PRINT "ERROR IN
       DAY" : GOTO 690
790  IF LEN(TC$) < > 5 THEN PRINT "ERROR IN
       TRANS CODE" : GOTO 690
800  REM
810  REM
820  REM
830  PRINT: INPUT "A, D, OR R; USER INFO,
       S P OR X"; TT$, CM$, ST$
840  CM$=LEFT$(CM$, 23)
850  IF TT$ < > "A" THEN IF TT$ < > "D"
       THEN IF TT$ < > " R" THEN PRINT
       "ERROR IN TRAN TYPE" : GOTO 830
860  REM IF DELETE (TT$=D) THEN FILL FIELDS
       AND DON'T BOTHER WITH REST OF
       INPUT
870  IF TT$ < > "D" THEN 890
880  ST$="X" : I$="X" : R$="Z" : R=0.0 :
       V=0.0 : N=0.0 : G=0.0 : GOTO 1040
890  IF ST$ < > "S" THEN IF ST$ < > "P"
       THEN IF ST$ < > "X" THEN PRINT
       "ERROR IN VAT TYPE" : GOTO 830
900  IF ST$="X"
       THEN  PRINT: PRINT "ENTER 2 FOR
       IMPORT, 6 FOR ASSET PURCHASE":
                PRINT"          3/4 FOR UNDER
                  DECLARATIONS,":
                PRINT"          /8 FOR OVER
                  DECLARATIONS":
                PRINT"          0  FOR EXPORT
                  SALE":
                PRINT"          E  FOR EXEMPT
                  SALE": INPUT
                  ST$
910  IF ST$="S" OR ST$="2" OR ST$="3" OR
       ST$="4" OR ST$="E" OR ST$="0" THEN
       I$="O"
920  IF ST$="P" OR ST$="6" OR ST$="7" OR
       ST$="8" THEN I$="I"
930  PRINT : INPUT" VAT RATE(S A B OR Z),
       G OR N, VALUE"; R$, G$, X
940  IF R$ < > "S" THEN IF R$ < > "A" THEN IF
       R$ < > "Z" THEN PRINT "ERROR IN VAT
       RATE" : GOTO 930
950  IF G$ < > "G" THEN IF G$ < > "N" THEN
       PRINT '.ERROR IN GROSS/NETT IND" :
       GOTO 930
955  IF ST$="0" OR ST$="E" THEN R=0.00 :
       GOTO 1000
960  IF X<0 THEN PRINT "VALUE NEGATIVE,
       IS IT CREDIT NOTE"; : INPUT Z$:IF
       LEFT$ Z$, 1)="Y" THEN 970 ELSE 930
965  IF X=0 THEN PRINT "VALUE ZERO -
       ERROR": GOTO 930
970  IF R$="S" THEN R=8.00
980  IF R$="A" THEN R=12.50
990  IF R$="Z" THEN R=0.00
1000 IF G$="G" THEN
       N=(100/(100+R))*X:N=(INT(N*100))/
       100: G=X: V=G-N
1010 IF G$="N" THEN
       N=X: V=(INT(R*N))/100:G=N+V
1020 REM
1030 REM
1040 REM ALL THE DATA IS COLLECTED NOW
1050 REM GET THE USER TO CHECK IT
1060 PRINT
1065 REM PRINT RECORD ON CONSOLE FOR
       CHECKING
1070 GOSUB 1340
1080 INPUT "'RETURN' IF OK";A$
1085 REM IS USER SATISFIED ? IF NOT DO IT ALL
       AGAIN
1090 IF A$ < > "" THEN 680
1095 REM CREATE RECORD, OUTPUT IT, &
       PRINT IT ON LIST DEVICE
1100 GOSUB 1430
1110 REM RECORD NOW DEALT WITH, START
       AGAIN !
1120 GOTO 680
1130 REM
1140 REM EOJ AND CLOSE FILE ROUTINES
1150 REM
1160 PRINT #0 : PRINT #0," END OF JOB "
1170 PRINT #0," TOTAL RECS CREATED = "; TR
1180 PRINT #0, USING S4$; TG
1190 PRINT #0, USING S5$; TN
1200 PRINT #0, USING S6$; TV
1210 PRINT #0 : PRINT #0
1220 PRINT #2 : PRINT #2," END OF JOB "
1230 PRINT #2, "FILE CREATED NAMED"; F$
1240 PRINT #2, "TOTAL RECS CREATED = "; TR
1250 PRINT #2, "ADDITIONS = "; TA;"
       DELETIONS = "; TD;" REPLACEMENTS
       = ";TC
1260 PRINT #2, USING S4$; TG
1270 PRINT #2, USING S5$; TN
1280 PRINT #2, USING S6$; TV
1290 PRINT #2 : PRINT #2
1300 REM NOW CLOSE FILE
1310 CLOSE #5
1320 REM
1330 END
1340 REM CONSOLE RECORD PRINT ROUTINE
1350 REM
1360 PRINT: PRINT
1370 PRINT" TRANSACT TRANS S   T   USER
       I  X  VAT    GROSS     NETT     VAT"
1380 PRINT" DATE CODE #   C COMMENTS
       O   C RATE   VALUE      VALUE '
1390 PRINT USING S2$; LEFT$(TD$, 2), MID$
       (TD$, 3, 2), RIGHT$(TD$, 2), TC$, S$, TT$,
       MID$(CM$, 1, 10), I$, ST$, R, G, N, V
1400 PRINT
1410 RETURN
1420 REM
1430 REM CREATE RECORD, TOTAL IF UP,
```

*(continued on next page)*

*(continued from previous page)*

```
      WRITE IT OUT, LIST IT
1440  REM
1450  IF LC>66 THEN GOSUB 1730
1460  LC=LC+1
1470  PRINT #2, USING S3$; LEFT$(TD$, 2),
        MID$(TC$, 3, 2), RIGHT$ (TD$, 2), TC$,
        S$, TT$, MID$(CM$, 1, 17), I$, ST$, R, G, N, V
1480  REM NOW CREATE TL$ PORTION OF
        RECORD
1490  MID$(TL$, 1, 6)=TD$
1500  MID$(TL$, 7, 5)=TC$
1510  MID$(TL$, 12,1)=S$
1520  MID$(TL$, 13, 1) =TT$
1530  MID$(TL$, 14, 23)=CM$
1540  MID$(TL$, 37, 1)=I$
1550  MID$(TL$, 38, 1)=ST
1555  REM NOW TOTAL UP HASH TOTALS FOR
        CONTROLS
1560  TG=TG+G : TN=TN+N : TV=TV+V
1570  TR=TR+1
1580  IF TT$="A" THEN TA=TA+1
1590  IF TT$="D" THEN TD=TD+1
1600  IF TT$="R" THEN TC=TC+1
1610  REM
1620  REM NOW GET RID OF DECIMAL POINTS
        IN VALUES
1630  REM
1640  R=R*100 : G=G*100 : N=N*100 : V=V*100
1650  REM
1660  REM NOW WE'RE READY TO OUTPUT
        RECORD
1670  REM
1680  PRINT #5, USING S7$; TL$, R, G, N, V
1690  REM FINISHED
1700  REM
1710  RETURN
1720  REM
1730  REM LINE PRINTER HEADINGS ROUTINE
1740  REM
1750  PC=PC+1
1760  PRINT #2 : PRINT #2
1770  PRINT #2," ";D$;" VAT TRANSACTION
        CREATE PROGRAM"; TAB(60);" PAGE
        ";PC
1780  PRINT #2 : LC=4
1790  RETURN
1800  REM
```

## LISTING OF PROGRAM TWO, VAT SUITE TRANSACTION FILE MEMORY SORT

```
  10  REM VAT TRANSACTION SORT PROGRAM
  20  REM COPYRIGHT (1978) XITAN SYSTEMS
  30  REM WRITTEN BY G. C. LYNCH
  40  REM
  50  REM
  60  REM USES SHELL-METZNER METHOD
  70  REM
  80  REM READS VAT TRANSACTIONS INTO
        STRING ARRAY I $
  90  REM SORTS THEM INTERNALLY
 100  CHANGES NAME OF INPUT FILE TO
        EXTENSION OF OLD
 110  REM OUTPUTS TO FILE WITH EXT OF NEW
 120  REM
 130  REM
 140  REM
 150  REM INITIALISATION
 160  REM THE NEXT TWO STATEMENTS MUST BE
        AMENDED FOR YOUR MEMORY SIZE
 170  CLEAR 20000, 2
 180  DIM I $(300)
 190  REM
 200  REM
 210  S0$="'LLLLLLLLLLLLLLLLLLL
        #####.##"
 220  N7=0:C7=0
 230  PRINT" VAT TRANSACTION SORT PROGRAM'
 240  INPUT "ENTER DATE OF RUN (DD/MM/YY)
        ";D$
 250  PRINT #2," VAT TRANSACTION SORT RUN
        ON "; D$
 260  INPUT "ENTER FULL NAME OF FILE TO BE
        SORTED"; F$
 270  X=INSTR(F$, ".")
 280  IF X=0 THEN 260
 290  FI$=LEFT$(F$,X-1)
 300  FS$=FI$: FS$=FS$+". OLD":FO$=FS$
 310  FS$=FI$: FS$=FS$+". NEW":FQ$=FS$
 320  REM
 330  REM
 340  REM OPEN INPUT FILE AS UNIT 5
 350  OPEN #5, "I", F$
 360  ON EOF #5 GOTO 520
 370  REM
 380  REM
 390  REM READ RECORD FROM INPUT FILE
 400  INPUT #5, TL$
 410  REM INCREMENT RECORD COUNT
 420  J=J+1
 430  REM INPUT FILE CONTROL TOTALS
        ROUTINE
 440  GOSUB 1130
 450  REM MOVE RECORD INTO INTERNAL ARRAY
 460  I$(J)=TL$
 470  REM NOW GET NEXT RECORD
 480  GOTO 400
 490  REM
 500  REM
 510  REM
 520  REM END OF FILE REACHED
 530  REM PRINT CONTROL TOTALS FOR INPUT
        FILE
 540  GOSUB 1400
 550  REM CLOSE FILE
 560  CLOSE #5
 570  REM
 580  REM
 590  REM SORT ROUTINE
 600  REM
 610  Y=J
 620  M6=Y
 630  M6=INT(M6/2)
 640  REM IF M6=0 THEN ITS END OF SORT
 650  IF M6=0 THEN 870
 660  K6=Y-M6
 670  J6=1
 680  I6=J6
 690  L6=I6+M6
 700  C7=C7+1
 710  IF MID$(I$(I6), 1, 13) <=MIDS(I$(L6), 1, 13)
        THEN 820
 720  N7=N7+1
 730  REM PRINT ON CONSOLE TO SHOW SORT
        STILL WORKING
 740  PRINT"S";
 750  EXCHANGE I$(I6), I$(L6)
 760  REM EXCHANGE IS SPECIAL FEATURE IN
        TDL BASIC
 770  REM USE FOLLOWING REM CODE IF YOU
        HAVEN'T GOT EQUIVALENT
 780  REM VV$=I$(L6) : I$(L6)=I$(I6) : I$(I6)=VV$
 790  REM
 800  I6=I6-M6
 810  IF I6>=1 THEN 690
 820  J6=J6+1
 830  IF J6>K6 THEN 630
 840  GOTO 680
 850  REM END OF SORT CODE
 860  REM
 870  REM SORT ROUTINE COMPLETED
 880  REM RENAME INEUT FILE PRIOR TO OUTPUT
 890  RENAME F$,FO$
 900  REM
 910  REM OPEN OUTPUT FILE
 920  OPEN #6, "O",FQ$
 930  REM
 940  REM NOW OUTPUT RECORDS HELD IN
        ARRAY I$
 950  REM
 960  FOR K=1 TO J
 970  PRINT #6, I$(K)
 980  REM OUTPUT FILE CONTROL TOTALS
        ROUTINE
 990  GOSUB 1260
1000  NEXT K
1010  REM
1020  REM FINISHED WITH OUTPUT FILE
1030  CLOSE #6
1040  REM
1050  REM PRINT OUTPUT FILE CONTROLS
1060  GOSUB 1500
1070  REM
1080  PRINT" END OF JOB"
1090  PRINT #2" END OF JOB"
1100  END
1110  REM END OF PROGRAM FLOW
1120  REM
1130  REM I/P CONTROL TOTALS ROUTINE
1140  REM USES TL$ TO DERIVE DATA
1150  REM IR
1160  REM IG  =  GROSS  VALUE  (PENCE)
1170  REM IN  =  NETT   VALUE  (PENCE)
1180  REM IV  =  VAT    VALUE  (PENCE)
1190  IR=IR+1
1200  IG=IG+   VAL(MID$(TL$, 44, 7))
1210  IN=IN+   VAL(MID$(TL$, 51, 7))
1220  IV=IV+   VAL(MID$(TL$, 58, 6))
1230  REM HASH TOTALS FINISHED
1240  RETURN
1250  REM
1260  REM OUTPUT CONTROL TOTALS ROUTINE
1270  REM USES I$(K) TO DERIVE DATA
1280  REM
1290  REM QR  =  TOTAL RECS
1300  REM QG  =  GROSS  VALUE  (PENCE)
1310  REM QN  =  NETT   VALUE  (PENCE)
1320  REM QV  =  VAT    VALUE  (PENCE)
1330  QR=QR+1
1340  QG=QG+   VAL(MID$(I$(K), 44, 7))
1350  QN=QN+   VAL(MID$(I$(K), 51, 7))
1360  QV=QV+   VAL(MID$(I$(K), 58, 6))
1370  REM HASH TOTALS FINISHED
1380  RETURN
1390  REM
1400  REM PRINT INPUT CONTROL TOTALS AT
        EOF
1410  PRINT #2, USING S0$; "TOTAL RECORDS
        INPUT", IR
1420  PRINT #2, USING S0$; "HASH TOTAL INP
        GROSS"; IG 100
1430  PRINT #2, USING S0$; "HASH TOTAL INP
        NET"; IN/100
1440  PRINT #2, USING S0$; "HASH TOTAL INP
        VAT"; IV/100
1450  PRINT #2: PRINT #2, "FILE INPUT AS";
        F$ "NOW RENAMED TO"; F0$
1460  PRINT #2
1470  REM TOTALS NOW OUTPUT
1480  RETURN
1490  REM
1500  REM PRINT OUTPUT CONTROL TOTALS
        AT EOF
1510  PRINT #2, USING S0$; "TOTAL RECORDS
        OUTPUT", QR
1520  PRINT #2, USING S0$; "HASH TOTAL OUT
        GROSS"; QG 100
1530  PRINT #2, USING S0$; "HASH TOTAL OUT
        NET"; QN/100
1540  PRINT #2, USING S0$; "HASH TOTAL OUT
        VAT"; QV/100
1550  PRINT #2: PRINT #2, "FILE NOW SORTED
        & OUTPUT AS"; FQ$
1560  PRINT #2
1570  RETURN
```

# WIRING FOR SOUND

A SIMPLE and amusing application for single board computers like the KIM is to use it as a mini electronic organ. Each key of its integral keyboard can be used to produce a different note.

In this version we use 21 of the keys (0 to F, AD,DA, +, GO, PC) to give notes in ascending order from the G below Middle C (196 Hz) to the C sharp in the octave above Middle C (555Hz).

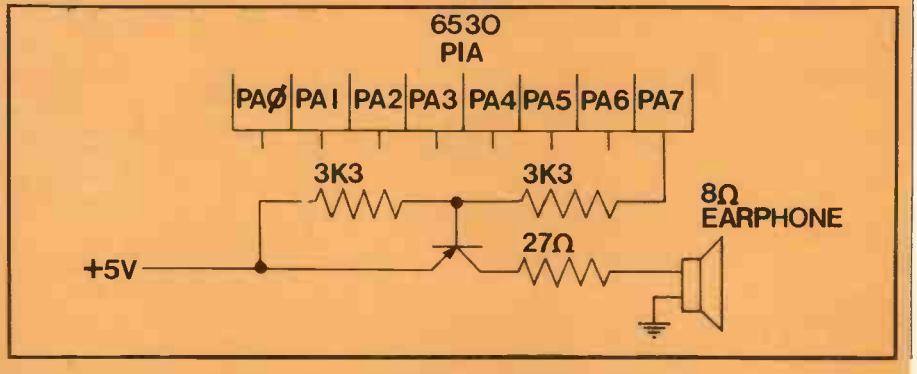
The waveform is not the ideal output of a music generator, since it is an asymetric square wave; the result, however, is acceptable. The waveform is generated by the software and is output on any of the PIA,PA0-7 pins; the output from any one of the pins can then be fed with suitable attenuation into an audio amplifier.

The program starts at Hex 200 with an initialisation routine and carries on to a loop at Hex 210 which looks for key depressions.

The GETKEY subroutine of the KIM monitor places a number, between 0 and 21, depending on which key is being pressed, in the accumulator. That number is placed in the index register from which it can be used as a pointer by the interrupt routines to the correct value in the data tables.

This data value is loaded into the counter/timer of the 6530, which generates an interrupt to the processor after a period determined by the data value. Each alternate interrupt gives rise to an output from the PIA of either 00 or FF Hex.

```
                        ; PROGRAM STARTS AT
                        $200
                        ;
                        GETKEY  =$1F6A
                        INT VEC =$17FE
                        PIA     =$1700
                        PIADD   =$1701
                        CT8     =$170D
                        CT1024  =$170F
0000                    XSTORE  ★=★+1
0001        FF          TEST    .BYTE $FF
0002                        ★=2
                        ; THE INTERRUPT
                        HANDLER
0002  48                INT 1   PHA
0003  8A                        TXA
0004  48                        PHA
0005  A6  00                    LDX XSTORE
0007  BD  32  02                LDA TABLE1,X
000A  8D  0F  17                STA CT1024
000D  A9  00                    LDA #$00
000F  8D  00  17                STA PIA
0012  A9  1B                    LDA #INT2
0014  8D  FE  17                STA INTVEC
0017  68                        PLA
0018  AA                        TAX
0019  68                        PLA
001A  40                        RTI
001B  48                INT2    PHA
001C  8A                        TXA
001D  48                        PHA
001E  A6  00                    LDX XSTORE
0020  BD  47  02                LDA TABLE2,X
0023  8D  0D  17                STA CT8
0026  A9  FF                    LDA #$FF
0028  8D  00  17                STA PIA
002B  A9  02                    LDA #INT1
002D  8D  FE  17                STA INTVEC
0030  68                        PLA
0031  AA                        TAX
0032  68                        PLA
0033  40                        RTI
                        ; THE MAIN PROGRAM,
                        START AT $0200
0034                        ★=$0200
0200  78                      SEI
0201  A9  02                  LDA #INT1
0203  8D  FE  17              STA INT VEC
0206  A9  00                  LDA #0
0208  8D  FF  17              STA INTVEC+1
020B  A9  FF                  LDA #$FF
020D  8D  01  17              STA PIADD
0210  20  6A  1F    BACK      JSR GETKEY
0213  C9  15                  CMP #$15
0215  B0  13                  BCS ELSE
0217  AA                     TAX
0218  86  00                  STX XSTORE
021A  A5  01                  LDA TEST
021C  C9  00                  CMP #0
021E  F0  F0                  BEQ BACK
0220  A9  00                  LDA #0
0222  85  01                  STA TEST
0224  58                      CLI
0225  00                     BRK
0226  EA                     NOP
0227  4C  10  02              JMP BACK
022A  78              ELSE    SEI
022B  A9  FF                  LDA #$FF
022D  85  01                  STA TEST
022F  4C  10  02              JMP BACK
0232        04    TABLE 1   .BYTE 4,4,3,3,3
0233        04
0234        03
0235        03
0236        03
0237        03              .BYTE 3,2,2,2,2
0238        02
0239        02
023A        02
023B        02
023C        02              .BYTE 2,2,1,1,1
023D        02
023E        01
023F        01
0240        01
0241        01              .BYTE 1,1,1,1,1,1,1
0242        01
0243        01
0244        01
0245        01
0246        01
0247        7E    TABLE 2   .BYTE 126,90,184,
                            152,122
0248        5A
0249        B8
024A        98
024B        7A
024C        5E              .BYTE 94,195,170,
                            146,123
024D        C3
024E        AA
024F        92
0250        7B
0251        66              .BYTE 102,82,191,
                            173,156
0252        52
0253        BF
0254        AD
0255        9C
0256        8C              .BYTE 140,125,111,
                            85,73,97
0257        7D
0258        6F
0259        55
025A        49
025B        61
025C
```



# CONTROLLING AC POWER

THE ABILITY to use a low-cost computer like the Pet to control external AC power devices such as electric lamps and motors opens a wide range of applications in a great many fields.

The problem may appear simple; there is an output port from the processor and one connects one of these lines to a TRIAC or SCR and connects the load across it. But this solution will almost certainly prove to be dangerous and because of noise problems make your processor unreliable. To overcome these problems we must separate the AC power device ground from the computer ground and electrically isolate the outputs of our processor from the control circuitry.

The circuit which I shall outline is that required for one single control output. If you wish to utilise all eight outputs of a PIA you repeat the circuit eight times. The output from a PIA, like the 6820 or 6520, is a latched TTL level output, each of the eight lines being under independent programme control. The first stage in the control circuit is to feed this output into an opto-isolator via a current limiting resistor. This, as its name suggests, isolates the rest of the control circuitry from the processor, thus preventing noise and damaging voltages from reaching your computer.

To boost the power output from the opto-isolator, we use an emitter follower NPN transistor. This transistor is supplied by the control circuit's own +12v power supply, the control circuit is shown in fig 1. The output of the circuit A comes from the transistor via a 680 ohm resistor, whose function is to limit the current in case of a short circuit; the output is about 5v at 10ma. For low power DC applications, this output could be fed into a reed relay, which would allow control of DC devices drawing up to ·5 amps at up to 50 volts.

We want to be able to control high power AC devices and the ideal device for doing this is the solid state relay. The SSR is similar in construction to an opto-isolator, in that its input portion is an LED. Instead of a phototransistor, however, a photosensitive resistor is used; it provides the turn-on current of the TRIAC portion of the SSR.

Thus we are able to control AC mains power at currents of up to 25 amps with a 5v 10ma input.

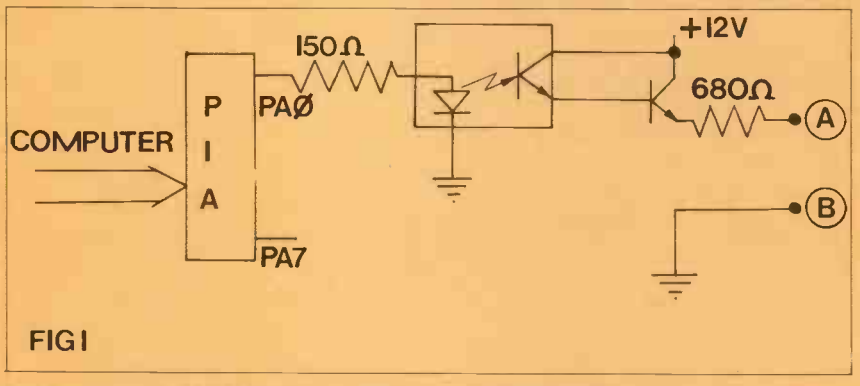


FIG 1

Fig 2 shows an SSR-based power control circuit. The resistor capacitor filter is to suppress transients produced by inductive loads which might cause erratic operation of the SSR. The LED provides an indication of the ON/OFF status and the fuse provides overload protection.

These two simple circuits should give some idea of how to construct systems requiring the control by computer of large AC and DC electrical devices. ■

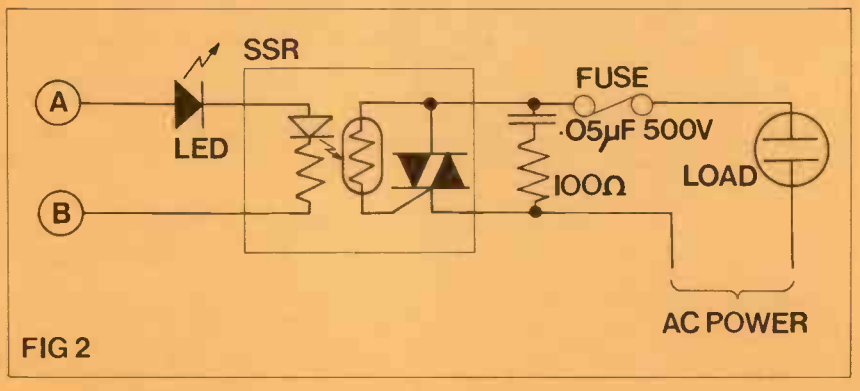


FIG 2

# Too little effort put into software choice

ONE of the best and most sophisticated operating systems available for microcomputers is known as CP/M.

It is surprising how many people tend to give a great deal of time and thought to choosing hardware but put very little effort into the selection of software. Often it is the quality of the software rather than the hardware which makes a system perform well.

A very important part of the software, especially on disc-based computers, is the operating system software, which usually goes under names like DOS, MINIDOS, RDOS, CDOS or CP/M.

The operating system is the software which integrates and controls all the individual components of a computer system. It can be regarded as a software interface between the machine code environment and the high-level language environment.

## Widespread use

It is the operating system which controls the speed and efficiency of disc access by the high-level language. Some operating systems also provide user functions like disc copying and initialisation, as well as offering the user the ability to look at and change sections of RAM in disc.

The choice of operating system also determines the range of software you can run on your system. Thus, some operating systems will allow you only to run software from one manufacturer, while others will allow you to use a wide range.

In its various forms CP/M has been in widespread use for more than three years and with a price tag of around £50 represents amazing value in a software package.

## Easy to alter

CP/M is marketed by an American company, Digital Research. The package is defined as a control program/monitor for a microcomputer system employing Intel 8080 or Zilog Z80 CPU and IBM-compatible flexible disc for back-up storage.

Its importance lies not only in the success and universal use of the 8080 and its extended family of chips, but also in the ease with which CP/M can be altered to work in different configurations.

The secret of its easy re-configuration is its highly-structured and modular design. Although it was developed originally for the Intel MDS 800, its structure and excellent documentation has attracted the attention of other manufacturers. Consequently, CP/M can be obtained for a variety of 8080 micro systems employing hard- or soft-sectored, single- or dual-density, mini- or full-size floppy discs.

## Sophisticated

CP/M is essentially a monitor system. The term 'monitor' may not be obvious to those who have not developed programs in interactive environments but, as it suggests, it simply monitors, in particular, the operator's keyboard. A monitor program would, for example, reflect, in a duplex system, the keyboard input to a video monitor or terminal printer and at the same time respond to a set of commands which execute programs or routines.

This is what CP/M is about. Time-sharing users and computer operators will be very familiar with the advantages of an interactive monitor and those with such experience will appreciate the sophistication of commands available under CP/M.

Starting-up CP/M is much like any other disc operating system with a two-stage cold and warm 'boot'. As the system springs to life, it outputs a sign-on message to the console, followed by a prompt to indicate the monitor is ready for a command.

## Similarity

Those used to developing programs on sharing systems could probably guess what to do next. Type in DIRECTORY, perhaps? Right. In fact, the command is abbreviated to DIR and, as you would expect, a list of the files on the directory—in this case a floppy disc—is produced at the console device.

The size of files or directories can be determined by the statistics or STAT command. Another command, TYPE, followed by a filename, will list ASCII files at the terminal. Other simple commands also exist for re-naming files (REN), erasing files (ERA) and copying files (PIP).

Digital users may feel that some of this looks familiar and the similarity of CP/M to Digital systems continues. For example, where have you seen the file name extension BAS before? This extension and others allow the user and system to identify file types.

One file extension, COM, is particularly important. If you feel you would like an

*(continued on next page)*

*(continued from previous page)*

additional command in your monitor set, program and test it, call it whatever COM and you have a new command.

Another close similarity to Digital software is the CP/M text editor. ED filename initiates a very powerful character and line-editing system which compares favourably to the best editors available on large time-sharing systems.

## It's a delight

The editor allows paging in chunks of a file, string searching, string substitutions, moving a character pointer and inserting or deleting characters or lines. A casual user might find it a little difficult to cope with but those who are modifying code frequently will delight in using this super program development aid.

Along with CP/M is a standard Intel 8080 assembler. In general, one assembler looks much like another and the only features of the CP/M assembler worth mentioning are directives such as ORG, EQU, SET, IF and ENDIF—the last three provide good facilities for conditional assemblies. Diagnostics are as helpful as most assemblers and the product of the assembly is a print and a hex file written to disc.

## With bonus

LOAD, predictably, loads the HEX files into memory. More exicting are the debugging aids in the program DDT. Facilities are fairly extensive and allow direct input of code, display of code in HEX,ASCII or mnemonics, movement of segments of memory and substitution of memory valves or CPU register content or state. Finally, debugged programs can be saved back on disc by the SAVE command.

CP/M is flexible to different hardware configurations and to this end the input/output drivers are available as source files. The size of the CP/M system can be modified from 16K upwards and after the BIOS (Basic input output system), which contains the drivers, has been 'patched in' using the aids mentioned, the new system can be saved back on disc with SYSGEN.

Finally, CP/M usually has one bonus piece of software free—a BASIC-E compiler, Although BASIC-E would be considered by most a fairly indifferent BASIC, it has an extended Dartmouth BASIC set and at the price must be considered excellent value.

The documentation to CP/M is excellent. The five manuals are available for around £15. Compared to most system manuals, they are extremely well-written and, even more surprisingly, accurate.

Outside CP/M are several other features which indirectly make the product look more interesting. For example, there is an extensive CP/M users' library. It embodies 20 full diskette volumes of 'public domain' software, and, at about £10 per volume, they represent exceptional value.

## Free entry

The content varies from esoteric utilities to general business packages and games. The CP/M Users' Club costs nothing to join and details are available from The Editor, *Practical Computing*, 2 Duncan Terrace, London, N1. (Please send a stamped-addressed envelope).

Yet one other exciting aspect of CP/M has to be mentioned—Microsoft products for CP/M. Microsoft, an American organisation, has developed undoubtably the best set of micro software which runs under CP/M. It includes the Microsoft BASIC interpreter—this is very similar to Altair BASIC—an ANSI FORTRAN compiler, an ANSI Cobol compiler and most recently, APL and PASCAL. The price range of these interpreters/compilers is as low as £150-£400.

Even the pundits who criticise the 8080 series hardware for particular applications have to admit that software systems such as CP/M and an 8080 disc system outperform in facilities, though not necessarily in speed, many of the minis on the market. ■

# Library club for CP/M users

A USERS' club library for CP/M has been established. We have received a listing of the first 14 volumes of discs in the library and look forward to seeing and using some of the programs when our copies of the discs arrive.

All the programs are free of copyright, so anyone is free to use them or modify them, we are told. The following is a brief list of extracts from the contents:

Vol 1: 34 various CP/M utility programs; driver routines for both Diablo and Qume printers; diskette-to-cassette and cassette-to-disc transfer; disassemblers; CP/M 1/0 subroutines.

Vol 3: a collection of 39 games programs written in BASIC-E, including old favourites like Amaze, Black Friday, Lunar Lander, Wumpus and for those with enough memory, a 27K Startrek.

Vol 4: contains some interesting-looking programs, including ML80, which is described as a Macroprocessor; and

*(continued on next page)*

(*continued from previous page*)

Actor, which is described as an interpreter.

Vol 5: contains both a compiler and an interpreter version of BASIC-E and a small list of programs written in Microsoft BASIC, like Othello and Startrek.

Vol 6: programmes from the Chicago area Computer Hobbyist Software Exchange, the most interesting of which is probably the cache mail list programes.

Vol 7: another disc for people interested in unusaul languages, this is Pilot, an educational language with a linguistic rather than a numeric base. An improved version of Pilot for the Z-80 is on Vol 12.

### Disc formatter

Vol 8: a further set of 28 CP/M utility programmes including a disc formatter for systems using the Tarbell controller or any other controller using the WD1771 chip. Those with a Bytesaver should find the PROM blowing programme useful.

Vol 9: the complete set of 12 programes in the General Ledger Package, by Bud Shamburger, first published in *Interface Age*, September, 1977.

Vol 10: the replacement for Vol. 2 and contains the Lawrence Livermoor version of Tiny BASIC.

Vol 11: A Tiny Basic with disc handling which runs in 5K, written by Processor Technology.

Vol 13: a mixed collection of 21 programs written in Microsoft Basic and Basic-E; mostly games.

Vol 14: another set of CP/M utilities, including a disc viewer program which will dump on to a standard console file, CP/M groups, or sectors in ASC11 or Hex format simultaneously.

### Expansion hope

As a result of a meeting at the DIY Computer Show in June, a Cromemco and North Star Users' Group has been set up in the U.K. It has only 15 members but since it is known that there are more than 200 machines by the two manufacturers in the U.K., there are great hopes it will expand.

The aims of the group are to promote the free interchange of information and experience in both hardware and software between users and anyone else who is interested. Computabits is prepared to provide space in each issue for members to publicise their activities.

The next meeting of the group will be in September. If you are interested in joining please contact the editor of Computabits. ■

# Pilot new language for micros

A NEW language for micros called Pilot is available from Computer Workshop. Pilot (Program Inquiry Learning or Teaching) can be used for controlling interactive conversation with a computer. It is capable of being used successfully by very inexpert programmers, but is also attractive to the expert.

It is built around four instructions—Type, Accept Input, Match Input, Conditional Jump. 'Accept input' will literally take anything.

Match input will search the input for a variety of things such as an embedded numerical answer, an embedded keyword, a choice or combination of keywords, or gross mis-spelling of keywords. The Jump may be made dependent on the previous Match or use a condition, as in Basic.

### All-round aid

The importance of such a language must not be under-estimated. It is useful for an accountant to be able to type "print-out time sheet for Jones Ltd" and then the computer sorts out each file as required and takes appropriate action.

It is useful for a teacher to have a program which accepts a French phrase, points out spelling mistakes, wrong tense, and the like. In the past this has not been possible, except on large systems in complicated languages. It is now possible on a cheap micro and with minimal programming skills.

### Passing it on

To help get things going, a Pilot Users' Group has been set up. "We hope to hear from people who have produced something useful and we shall organise distribution of the information," says Charles Sweeten, director of computing at Oundle School, who is also Secretary of MUSE (Minicomputer Users in Secondary Education).

If you are interested in the user group contact Philip Couzens, Oundle School, Oundle, Peterborough. ■

---

# A PRACTICAL GLOSSARY

## Running the terminological gamut from B to C

### Buzzwords
The term buzzword is a piece of jargon used to describe the jargon the computer industry generates with such whole-hearted enthusiasm, and which this glossary is hopefully an aid to penetrating.

### Byte
If a *bit* is the smallest unit of information stored in a computer, then a byte is the smallest unit with any real meaning. It is a collection, normally, eight bits, which together represent a number or a character. Sharp-eyed readers will note that ASCII is eight-bit code; so one byte usually corresponds to one character, and the terms character and byte are frequently used interchangeably. For example, a *floppy disc* holding 71KB ( =71,000 bytes) wiil store 71,000 characters. Well, it will not quite do that, since the usable capacity is always less than the manufacturer says it is because of internal formatting considerations.

### Cards
One of the oldest forms of *input* medium and now somewhat-outmoded. They may be long and slim and capable of holding representations of 80 characters, or short and fat and capable of holding up to 96 characters. IBM, and not many other companies, uses the 96-character card.)

Those characters are represented by combinations of holes punched in the cards. If your computer uses an eight-level character code, and if you have an 80-column card, the card will be organised as an 80 × 8 matrix—80 columns vertically, eight rows horizontally. The character code determines which of the eight possible punch positions denotes which particular character. Unlike most other input media, you can rest coffee cups on cards, write addresses on them, and lose or damage single records easily.

Incidentally, 80-column cards were so widespread in the early 1960s that this became a *de facto* standard for record lengths, which is why so many terminals and other things assume you require lines 80 characters long. The irritating point about cards is that a record is always at least one card long—even if you want only one or two characters on it, you still have to use a whole card. You can forget about cards, anyway.

### CCD
Charge coupled device is a new memory technology of bubble memory. Expensive and not yet readily available.

### Cartridge
There are two kinds of cartridge. Tape cartridges are almost invariably made by 3M; they are like a cassette but they hold more, are more expensive, and more robust.

Disc cartridges hold from 2·5 to 10 Megabytes. They are the next stage from floppy discs and cost more.

### Cassette
Philips has set the standard for both audio and data cassettes. Micros will work happily on audio cassettes, although if you need to keep data very clean and tidy, you will probably have to pay more and buy a data cassette

### Character
A letter or numeric digit. Included here for completeness, although you know exactly what it is anyway.

### Chassis
The computer chassis is the box which contains the processor and main memory. It incorporates various elements, like the backplane, so it is an integral part of the computer system rather than a receptacle.

### Checkpoint
See breakpoint.

### Chip
A chip is a piece of silicon, normally about a quarter of an inch square and thick, holding the components which make up all or part of a microcomputer—one micro may be split across several chips. It is effectively the medium which holds the message. Anyone who uses the phrases "chips with everything" or "when the chips are down" automatically gets the *Practical Computing* Dumb Award, which consists of an on-line boxing glove loaded with a horseshoe for ritual self-effacement.

### Clock
The rate at which a computer performs operations is controlled internally by a clock. That is an electronic circuit or group of electronic components which generate a set of control signals. Each set of control signals will initiate an action on the part of the central processing unit (CPU).

### COBOL
The Common Business Orientated Language is one of the best-known high-level programming languages. It was designed for commercial applications, so its mathematical abilities are limited—some of us can sympathise with this. Because it has been designed to make program-writing easy in a commercial context, it can take up a good deal of room in the computer; this makes it more popular on larger systems rather than micros. Having said that, there are at least two Cobols for micros and they are British.

### CODASYL
The organisation responsible for the design of Cobol and an attempt to produce a database manager which is universally accepted. It lives in the United States, consists of representatives from government bodies and suppliers of computer systems and services, and promotes standardisation. Its outpourings are detailed and esoteric, which means that Cobol- and Codasyl-compatible database systems are complex and verbose.

It stands for Committee on Data Systems Languages.

### Code
A code in computer terms means the same as in other contexts; it is a means of representing one thing by something else, James White, in *Your Home Computer* (recommended) says. Sometimes a code is used for secrecy: in a computer a code is used for efficiency. Some programmers are confused by this.

The most common codes in computing, used to represent numbers and letters, are ASCII and EBCDIC. qv. (Which Is code for 'which see').

### Communications
This one is a real jungle of techniques and terminology. As a blanket term, it can refer to computer systems where one part is linked to another to allow transmission of information over a communications line which is normally an ordinary telephone line. Things become very complicated when there are numbers of system components sending data to each other, and a whole subculture has developed in the computer industry which concerns itself with exactly how you organise those components and the information flowing between them.

So let us remain with the simple end; communications are what happens when two system components communicate, and they communicate by sending data to each other.

### Compiler
The language in which instructions for the computer are written—in the form of a program, normally—is not one which the computer can use directly. It needs to be changed into a form which the computer can recognise. One way of doing this is to use a compilér.

A compiler is a specialised program which translates the source program into code the computer can execute. It does so much faster than the other method of translating the instructions, which involves using an interpreter. The interpreter has the great virtue, however, of enabling the user to change bits of a program and test the change immediately, which makes it very useful for program development.

You can also interrupt an interpreter in full spate to get some intermediate results, say, and then let it carry on. A compiler would not like you to do that.

What goes into a compiler is source code; what comes out is object code.

### Computer
A computer is a clever collection of components which enables you to put information in, store it, modify it, and get it out again. That is a very arbitrary definition and one which would fit a programmable calculator, too. A programmable calculator can be distinguished from a computer by its name—the distinction is a marketing one.

### Console
What you do to mournful micros. Also the control point of a computer system from the human operator's point of view. It is probably a keyboard and printer or VDU, for inputting messages to the system and getting back a response.

Sometimes the term is used for the front panel of a mini or a micro, the switches and/or pushbuttons which initiate system operations. Some minis have a programmer's panel or programmer's console. which include switches to set the contents of particular memory locations.

### Content Addressing
A method of obtaining information in the main computer memory by scanning it to find that information specifically. Conventionally, you have to ask the computer to tell you what is in a particular address.

You need the very special, very expensive, and very unproven content, addressable memories, to do it. The idea sounds great in principle.

### CORAL
CORAL, usually CORAL 66, is a language developed for real-time applications by the Royal Radar Establishment at Malvern. It is a trifle esoteric for the micro user. Still, the hills there are pleasant.

# New from Texas Instruments. The world's most powerful pocket calculators. For the easiest problem solving ever.

The new Texas Instruments Programmable TI-58 and TI-59 make your problem solving simpler and easier by making the electronics do more work. Now, commonly encountered programmes in maths, science, finance and statistics are set up and accessible at the touch of a key. You need add only the variables.

*Solid State Software*† is the name of this technological achievement from Texas Instruments. Even the programming is now included in the solid-state electronics. You get complete, pre-written problem solving libraries in convenient plug-in modules. Yet, no prior programming knowledge is necessary.

**The programmable TI-58 at £99.95***

Includes a Master Library *Solid State Software* module packed with 25 useful programmes, all at your command. Or you can key-in your own programmes and store the data — using up to 480 programme steps or up to 60 memories — and employ the Master Library programmes as subroutines.

Optional plug-in library modules are available to convert your calculator into a specialised problem solver in the fields of applied statistics, surveying, aviation, navigation — with many more 5,000-step libraries to come.

**The programmable TI-59 at £249.95***

Includes all the features of the TI-58 — plus more programme steps, more memories, and a magnetic-card capability. Record your own programmes on convenient magnetic cards and store them permanently in your personal problem-solving library.

The Texas Instruments Programmable TI-58 and TI-59 and the compatible PC-100A alphanumeric printer/plotter include a 1-year warranty. See the world's most powerful pocket calculators now.
Or use the coupon to obtain full product information.

Please send me full details on the TI-58, TI-59 and PC-100A.

Name _____

Address _____

_____

_____

_____

WC·1·78


Simulated Calculator Display.

**Giant technology. From the people who made micro-electronic calculators and watches possible.**

## TEXAS INSTRUMENTS LIMITED

†Trademark of Texas Instruments
*Suggested retail price, including VAT.

Texas Instruments Ltd, Supply Division, Manton Lane, Bedford. Tel: Bedford (0234) 67466.
Branch offices at: Slough – 0753 33411. Edinburgh – 031 229 5573. Stockport – 061 442 7000. Southampton – 0703 27267

● Circle No. 309

# COMPUTER WORKSHOP

A COMPLETE SERVICE TO
THE MICRO-COMPUTER BUYER

## Courses

You don't need to know a CPU from a VDU to benefit from our Courses.

| PEOPLE | PRICES | | SUBJECTS |
|---|---|---|---|
| Courses for Businessmen | 3-day Courses | £100 | BASIC |
| Courses for Teachers | 1-day Courses | £ 40 | Word Processing |
| Courses for Engineers | 5-evening Courses | £ 25 | ASSEMBLER |
| Courses for Hobbyists | | | Applications |
| Courses for Doctors | | | Med. Interviewing |

Courses arranged for Groups on request.

All Courses guaranteed to be in ENGLISH. Do NOT bring your dictionary of Computer Jargon. Courses held at our Training School conveniently situated in the West End.

## Application Packages

Text Editing and Word Processing £50.

Allows you to create and edit text files on disk or tape. Then with sophisticated control of spacing, margins, justification, and many other features, a perfectly formatted manual, letter, book, or whatever can be produced on the printer. Users of this type of program on large mainframe Computers find themselves immediately "at home" with this package.

Information Retrieval/Selective Mailing/Subscription Accounting £25.

This very versatile package allows you to create and maintain files of information (Databases to some) with as complicated (or simple) a structure of analysis codes as you need. Allows for entry of Cash Received and automatic generation of Subscriptions Due if required. The power of the selection routines allows you to enter as many parameters as you want for comparison with the records on file. For instance an Estate Agent could select all the houses with more than 2 bedrooms and a garage less than 5 miles from a station in the 20 to 25 thousand pound price bracket.

Watch This Space For More Applications Descriptions In Following Months Issues.

## Equipment. SWTPC, Centronics & Ricoh

Computers £275 - £975 (256-40K Memory). Twin Minifloppy £860.
1.2 megabyte Twin Floppy £1,525. VDU £455. Printers £250 - £1,800.
Printers from 40-col. dot matrix to double daisy wheel typewriters.

## Maintenance, Leasing & Software Development

Arranged through specialist companies.

## User Group Meetings

Held on the last Monday of every month at Dover Street.

COMPUTER WORKSHOP. 38, DOVER STREET, LONDON. W1X 3RB. 01-491 7507
COMPUTER WORKSHOP. 29, HANGING DITCH, MANCHESTER M4 3ES 061-832 2269