PERSONAL COMPUTER MAGAZINE

BITS & BYTES

11-9826

Issue No. 5, February 1983: $1.00

The Epson HX-20 and Act Sirius 1 reviewed

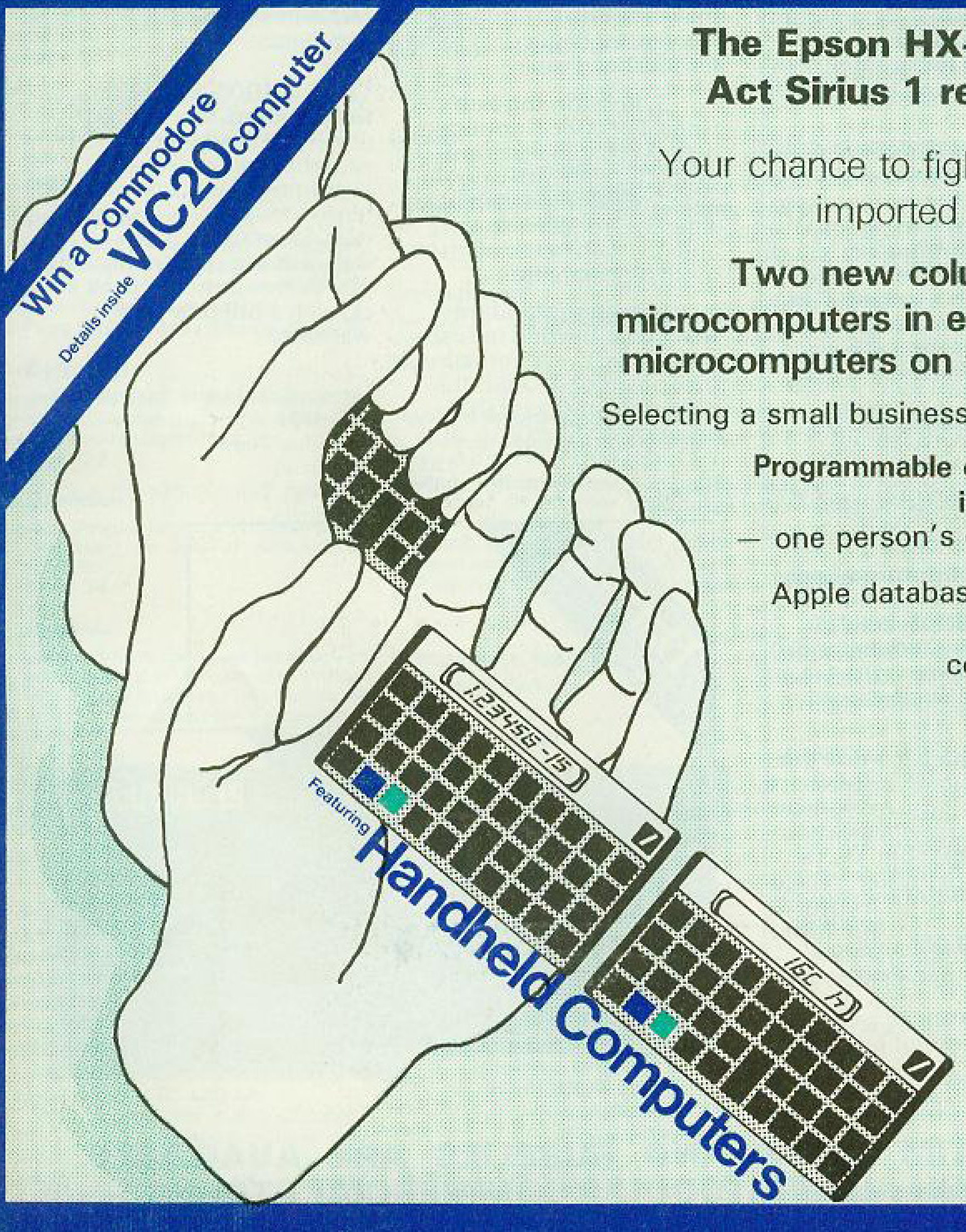
Your chance to fight tax on imported software

Two new columns — microcomputers in education microcomputers on the farm

Selecting a small business computer

Programmable calculators in business
— one person's experience

Apple database program
plus
columns for
VIC
Sinclair
80 Users



And now for the performance of a lifetime!

Introducing 7 data-shielding improvements from Verbatim for greater disk durability and longer data life.

Improvements to protect your data from head-to-disk abrasion. Improvements to shield your data against loss due to environmental conditions. Improvements that will deliver a longer lifetime of trouble-free data recording, storage and retrieval than ever before possible.

1. A longer-lasting lubricant. Our new lubricant is more resistant to diffusion, to protect against data-destroying head-to-disk contact.

2. An improved liner. Our new liner cleans and removes debris better. It also enables more lubricant to reach the recording head, protecting against head wear.

3. A thicker, more durable coating. Our disks have a more uniform oxide coating for more adhesive and cohesive strength. We've also made it thicker; providing 10% more protective lubricant and an optimized signal resolution for the new recording heads.

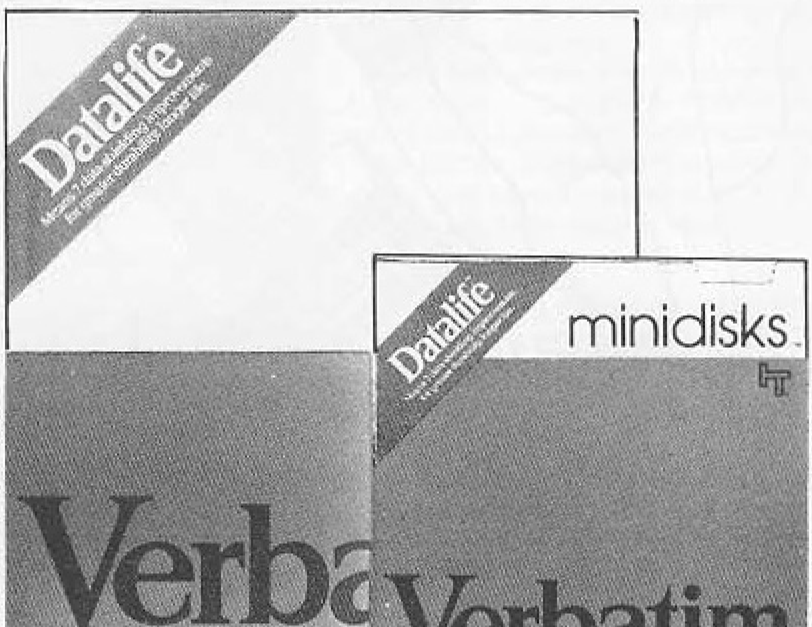
4. Advanced polishing techniques. Our burnishing makes our disks uniformly smooth, for better data transfer, less head wear.

5. Reinforcing hub rings. For critical applications, Verbatim disks are available with hub rings to aid in registration, eliminate slippage, reduce errors, and give you better alignment repeatability.

6. Testing standards that go far beyond industry standards. Every Verbatim disk meets or exceeds the most demanding of IBM, Shugart, ANSI, ECMA, and ISO standards – because we insist on Verbatim being the industry standard of excellence.

7. A 100% Error-Free Certification. Our certification is not based on random sampling, or statistical averaging, it is based on extensive testing of every single disk. Therefore we can state that our disks really are 100% error-free, and back it up with a **full one-year warranty.**

Compudata Media Systems Ltd.
Box 3273
48 Ellice Road
Auckland
4446-085 Telex 60044



VERBATIM DISKS AND SUPPLIES ARE AVAILABLE THROUGH ALL WHITCOULLS COMMERCIAL STORES

inside BITS & BYTES

LETTERS

VIC GRAPHICS

Sirs,

I thank Grant McLean for his defence of the VIC. (December issue). This machine certainly is no toy and byte for buck is excellent value. I would rate it a good, small machine with slightly limited expansion capabilities. Its forthcoming big sister, the Commodore 64, appears a lot more worth while in the long run.

As to inaccuracy in my October article I think I should outline the historical background to the 8-colour controversy. My remarks were based on the British experience with VIC in 1981. There was a great rush at that time to get the VIC on sale before a BBC prototype was made public and Britain was treated to three months of high-volume publicity on the VIC... but no machines.

My remarks were based on a review in "Personal Computing", by Gavin Sanders, who, to quote, "had exclusive use of one of the first three to arrive in Britain". On page 29 of this review he remarks on

Continued on page 12

EDITORIAL

Time for us all to speak out

The time has come for personal-computer clubs and other groups to speak up on sales tax.

The Customs Department is about to review the method of assessing duty on software; more important, according to one official the review may also look at the tax on hardware.

New Zealand's 40 per cent sales tax on hardware is indefensible. It is Luddite and anti-progressive. It is a drag on new technology. We believe that it was introduced (by the last Labour Government) without sufficient consideration of its effects.

The software duty review is associated with New Zealand's adopting the valuation code of the General Agreement on Tariffs and Trade (G.A.T.T.).

The case of personal-computer software duty is confused (see article in this issue), and a senior official in the Customs Department indicated that the sales tax would also be considered in the review.

This is of considerable importance, as a report of the Industries Development Commission just released recommends a reduction of the 40 per cent sales tax to 10 per cent, in line with the tax on plant and machinery.

Clubs and societies can now strike a blow for their members by making submissions to the Department on the tax. Don't forget the duty. And don't forget to put in a plug for fair treatment for the commercial computing world.

One way for a club to act would be to discuss the matter at a club night, someone noting points that arise during a general brainstorming. A sub-committee might then mould the thought into a coherent argument and send off the submission.

When you have your submission, strong in evidence and premises and logically presented as a convincing argument, send it off to:

The Controller of Customs
Private Bag
Wellington
(Attention: Trade Division)

Coming up in BITS & BYTES

Do you use cassettes?
Next month we have a feature on cassettes including tips on their care and even a way of making your own.

Hardware reviews
- Apple III
- Microbee
- Hitachi Peach [sorry no room this month]

Education
Nick Smythe continues his appraisal of microcomputer networks for schools.

Farming
More farm software, this time written by a Wairarapa company for several different microcomputer brands.

Business
John Vargo continues his series on selecting a small business computer.

BITS & BYTES is published monthly, except January, by Neill Birss, Dion Crooks and Paul Crooks. Editorial and subscriber inquiries to Post Office Box 827, Christchurch.

ADVERTISING:
Auckland — Wendy Whitehead. Phone 794-807
Wellington — Annie Carrad: Phone 723-431
South Island — Graham Beecroft: Phone 554-265 Chch or telephone Christchurch 66-566

EDITORIAL:
Auckland — Cathy and Selwyn Arrow: 30A Bracken Ave, Takapuna
Christchurch — P.O. Box 827 or Phone 66-566

Production: Roger Browning, Graeme Patterson, Lee Tecksui, Janine Morrell
Typesetting: Gail Rankin
Cover and Graphics: Sally Williams

Technical Editor: Chris O'Donoghue

Subscription rate: $8 a year (11 issues) adults
$6 a year for school pupils

Subscriptions begin from the next issue of BITS & BYTES after the subscription is received.

Back copies are available (except issue 1) at $1 per copy plus 50c per copy postage and packaging.

Subscription addresses:
When sending in subscriptions please include postal zones for the cities. If your label is incorrectly addressed please send it to us with the correction marked.

Distribution inquiries: Bookshops — Gordon and Gotch, Ltd. Computer stores — direct to the publishers.

Printed in Christchurch by D.N. Adams, Ltd.

Micro News... Micro News... Micro News... Micro News... Micro News... Micro News...

Interest in personal computers in New Zealand seems certain to receive a further boost with the screening of the 10 part B.B.C. computer programme beginning on March 2.

Television New Zealand will be heavily promoting the programme and the B.B.C. computer itself as TVNZ will receive a commission on every B.B.C. sold.

The computer programme aroused a lot of interest when screened in the United Kingdom and with a 6 pm timeslot the 25 minute programmes should attract many viewers here as well.

* * * *

A lot of effort is being put into technical support for the B.B.C. computer.

New Zealand wholesalers of the B.B.C., Access Data together with Barson Computers, will be providing a technical support hot-line available to all purchasers of the B.B.C., whether bought in New Zealand or not.

A qualified person will be onsite for every major installation in schools.

* * * *

IBM is expected to announce in February the release in new Zealand of its much awaited personal computer.

More details in our March issue.

* * * *

The Omnibyte Corporation of the U.S.A. have appointed E. C. Gough Ltd as their sole New Zealand agents.

Omnibyte manufacture a range of single board computers based around Motorola's MC6800, MC6809 and MC68000 microprocessors. Other products include a full range of support boards, micro-processor training systems, through to software development systems.

The latest addition to Omnibyte's range of single board computers is the OB68K1 which is based on the MC68000 16 bit microprocessor.

The OB68K1 has been designed to meet a variety of processing applications from small single board dedicated instruments to extremely large multiprocessing systems utilising several processor boards with shared memory and Input/Output.

The OB68K1 complies with IEEE P796 bus specifications (Multibus) and can have either 32K or 128K of RAM fitted on board. Sockets for up to 64K of ROM are also provided.

The OB68K1 also features a triple 16 bit timer/counter, two RS-232C serial ports, and two fully programmable 16 bit parallel ports.

* * * *

ICL has published the first edition of a catalogue of software for use on ICL computers and is currently circulating it to more than 50 countries.

Over 600 items of system and application software are listed for current ICL equipment ranging from the ICL Personal Computer to mainframe computers operating under VME 2900. More than two-thirds of the entries are from independent suppliers.

ICL(NZ) is now collecting entries from local software houses for inclusion in a New Zealand edition of the catalogue. The entries will be incorporated in the next International edition which will give free worldwide publicity for the New Zealand suppliers.

* * * *

The latest shipments of OSBORNE 1 Portable Business Computers have double density disk drives. This increases storage from 90K to 180K (formatted) per drive.

New Zealand distributors, Sirius Systems, advise that there will not be any increase in price. The OSBORNE 1 will still be $3,650.00.

Double density upgrade kits will be available some time in the new year but exact availability and pricing is not yet known.

* * * *

The Cromemco personal computer (the C-10) will be released in New Zealand early this year according to local agents McLean Information Technology.

Using a Z-80A microprocessor, the standard unit comes with 64K RAM, 24K ROM, a single 5¼ inch floppy disk drive (with 390K capacity) a detached CRT and four software packages for $4100.

The software packages are a CP/M compatible operating system, 32K structured BASIC, a word processor and a financial spread sheet calculator.

Three interface ports are also standard — an RS232 serial port and two printer ports (one serial and one parallel).

The C-10 is also expandable to a multi-user configuration (a number of units can be linked together for use by several people).

* * * *

Sirius Systems advise that they now have a representative in Wellington. Dr Jim Baltaxe, formerly of Massey University and one of the original OSBORNE 1 owners, has just been appointed corporate representative based in Wellington. Jim is an OSBORNE enthusiast and will make his presence felt to those interested in, or who own an OSBORNE 1.

He can be contacted through COMPUSALES SOFTWARE & HARDWARE LIMITED, 75 Ghuznee Street, Wellington.

Chance to fight duty on imported software

Imported computer software should be assessed for duty under the rules of last years GATT valuation code agreement. But as this would add yet another 70% to the original cost of a program, the same formula as was used up to 1982 is still being applied by the customs department as an interim measure.

This formula bases the value for duty (VFD) on:

Value of bare cassette or disk + cost of putting information on the medium + 100%.

As each cassette is worth about $2.00 to $3.00, duty should only amount to around $5.00.

The GATT agreement which came into force in New Zealand on July 1, 1982, requires the assessment of VFD to be based on the purchase price of the goods.

Under this agreement the rate of duty for computer software is 25 per cent under section 92.12 of the NZ tariff schedule.

The duty payable is then;

Wholesale FOB price for goods x 0.25.

FOB stands for freight on board — the cost of manufacture etc, plus freight costs to the vessel at the port of origin. If the "FOB" price paid by the purchaser is clearly a retail price then this is reduced by 20 per cent to establish a wholesale FOB.

Sales Tax

The level of sales tax payable is 40% of the estimated landed retail value.

The landed retail value, R, is:

(Wholesale FOB x 1.25 + Freight and insurance) x 1.25.

Sales tax payable is (R x 0.4), less 1% for prompt payment.

For the private individual importing software, this results in duty and sales tax at least 70 per cent of the purchase price. The customs dept appears to acknowledge that this is an "excessive" amount and that such a windfall of dollars was not intended by government. It is at this moment undertaking a review of the relevant parts of section 92.12 of the tariff schedule.

In addition the GATT valuation committee in Geneva is at present studying the whole question of tariffs in this area following "prompting" from several countries, including the USA. Customs is therefore being flexible (in some centres) in allowing the importer the option of having duty and sales tax determined under the GATT method, or under the interim formula which allows for the VFD of computer software to be based on the material value of the goods as described above.

The interim formula clearly results in a much lower VFD as the value of the information (i.e. program) itself is not being included in the formula.

Under Study

This interim measure is being used only while the matter is under study. It applies to tape and disks only and makes no distinction between types of programs such as games or utilities. Cartridges containing ROM's are classified as hardware-related programs and are automatically assessed for VFD on the basis of the FOB price.

The question remains then of what formula will apply when a business or an individual brings software into the country? The answer for this hinges on the matter of documentation. A business house is generally well organised in this area and so will probably have the necessary papers showing the material value etc of the goods, enabling the interim option to be applied.

The private, individual, however may not be able to persuade a software supplier in another country to change his standard documentation to accommodate our special needs, although it would not seem too much trouble for a statement such as ''value of tape $3.00'' to be included.

Documentation

The customs section of the local Post Office should then apply the VFD option according to the documentation available.

Therefore, if the supplier sends a copy of the invoice with the goods then the full purchase price will be the basis of assessment of VFD.

However, if a packaging slip only is sent with the goods or invoice stating ''no commercial value'' (NCV) is sent, provided the quantity of goods is not large, then Post Office Customs appear to be prepared to be flexible and assess VFD on the basis of an assumed material cost.

If the supplier cannot or will not provide material-value information it is clearly to the individual's advantage to request that no price information be sent with the goods, but that formal invoices, credit card slips etc be sent by separate mail (at extra cost if need be).

This is not circumventing the system, it is simply ensuring that customs has the opportunity to apply the VFD option of most advantage to the individual software importer.

Experience in Wellington has shown that on the basis of an assumed material cost, the VFD for a cassette or disk is taken to be $5.00.

In Christchurch, the feeling is that customs have been overcharging right from the start. Prior to July 1982 they valued software at its list price plus 25 per cent . . . not on media cost. Since that date they have combined the ''best'' features of both assessments, taking the full landed value of software from the GATT method and the 100 per cent markup from the other!

In Auckland, the situation is different again. The software is assessed on the full purchase, (sometimes this is reduced at the discretion of the officer concerned), the 25 per cent duty is then calculated and this appears to be added to the assessed price before calculating the 40 per cent sales tax. In one case recently, sales tax amounted to just over 60 per cent of the assessed value, so making a grand total of 86 per cent to be paid. The assessment on assumed material cost has not been applied in Auckland to our knowledge since July 1982, even though it has been reintroduced until a definitive policy is resolved.

Act Now

Just for once YOU have a chance to do something about this state of affairs.

The customs review mentioned above is accepting submissions on ''media for data processing equipment'', reference C 952.060/27. But only until 15 February 1983. So don't lose any time. Why don't we try to flood them with non-emotional submissions, for example tax on information but no tax if the same program is printed, impossibility of policing, damage to economy, etc. Submissions should be sent to Customs Head Office, Wellington. Attention D.G. Walker.

In the long term if the GATT formula is adhered to this will continue to retard the progress of computing in New Zealand.

Auckland • Wellington • and Christchurch correspondents. Compiled by Selwyn Arrow.

Low cost database

By BRIAN STRONG

Let's stop right here and establish what a database — one of computing's magic words — actually is.

Data consist of separate pieces of information; the entries on a bank statement, recipes in a book, a cardboard box full of cuttings or pages from magazines. When the word "database" is used it means hauling all those unrelated bits of information together and organising them in such a way that we can find things when we want them. The simplest form of a database is the notebook you carry with you, or its poor relation, the back of a cigarette packet on which you note important items and then throw away.

If you have just exhausted the possibilities of your new computer's manual and are starting to look around for software to set up your own database, some of you may still be a little pale from reading the three-figure price tag on most of it. Congratulations, you have just learned one of the facts of life you were never told. Database software, if you are looking for the tried, tested and reliable kind, is expensive, but you only get what you pay for.

Using that sort of logic you may now assume that the program that follows is sub-standard because it has only cost you the price of this issue (unless you're so mean you've flogged someone else's copy). It certainly isn't in the same league as a package such as PFS, but it sure beats nothing if your wallet is suffering from post-Christmas hernia and all your credit cards are full.

This program has been written for the Apple II, but can easily be adapted to other systems.

That's probably the most frustrating statement computer magazines ever print, always when the listing is for another system. Don't give up as there'll be explanations of some of the functions after the listing and you should find equivalents in your manual.

This "database" is called Notebook. If works for disk drives and cassettes because all you do is SAVE the entire program each time. All input is in the form of DATA entries and instructions are included in the listing. The only limitation will be in the amount of RAM available in your machine.

Notebook is a scratchpad for storing odds and ends. It will keep you going until you are financial enough to get something more eleborate.

```
5  NORMAL : TEXT
10  CALL  - 936: VTAB 3: HTAB 14: PRINT
     "NOTEBOOK": HTAB 14: PRINT "-
     --------": VTAB 10: HTAB 2: INPUT
     "DO YOU WANT THE INSTRUCTIONS
      (Y/N) : ";F$: HOME : IF F$ =
     "N" THEN 140
20  HOME : VTAB 3: HTAB 14: PRINT
     "NOTEBOOK": HTAB 14: PRINT "-
     --------": PRINT : PRINT :
30  PRINT "THIS PROGRAM IS DESIGNE
     D FOR STORING": PRINT : PRINT
     "AND RETRIEVING REFERENCES,RE
     MINDERS": PRINT : PRINT "AND
     OTHER INFORMATION THAT YOU NE
     ED": PRINT : PRINT "TO RECALL
     ."
40  VTAB 22: HTAB 5: PRINT "<<PRES
     S ANY KEY TO CONTINUE>>": GET
     A$: HOME : PRINT
50  VTAB 3: PRINT "INFORMATION IS
     STORED AS DATA ENTRIES": PRINT
     : PRINT "COMMENCING AT LINE N
     UMBER 1000. THE": PRINT : PRINT
     "FORMAT IS :- ": PRINT
60  PRINT "L/N DATA HEADING (COMMA
     ) DETAILS.": PRINT : PRINT "E
     XAMPLE :-": PRINT : PRINT "10
     00 DATA MEETING, CLUBROOMS-JA
     N 3RD"
70  VTAB 22: HTAB 5: PRINT "<<PRES
     S ANY KEY TO CONTINUE>>": GET
     A$: HOME : PRINT : PRINT : PRINT

80  PRINT "!!! IMPORTANT !!!": PRINT
     : PRINT "DON'T FORGET THE COM
     MA AFTER A HEADING": PRINT : PRINT
     "DATA ENTRIES CAN BE VARYING
     LENGTHS.": PRINT : PRINT "FIN
     AL DATA ENTRY MUST BE :-": PRINT
     : PRINT "L/N DATA END"
90  VTAB 22: HTAB 5: PRINT "<<PRES
     S ANY KEY TO CONTINUE>>": GET
     A$: HOME : PRINT : PRINT : PRINT

100  PRINT "A 'SEARCH' USING THE H
     EADING (MEETING)": PRINT : PRINT
     "WILL EXTRACT ALL ENTRIES WIT
     H THAT": PRINT : PRINT "HEADI
     NG.": PRINT : PRINT
110  VTAB 22: HTAB 5: PRINT "<<PRE
     SS ANY KEY TO CONTINUE>>": GET
     A$: HOME : PRINT
120  PRINT "BOTH 'SEARCH' AND 'ALL
     DATA' HAVE": PRINT : PRINT "O
     PTIONAL PRINTER OUTPUT IF REQ
     UIRED.": PRINT : PRINT " 'ALL

     DATA' DISPLAYS COMPLETE DATA"
     : PRINT : PRINT "ENTRIES IN O
     RDER OF LINE NUMBER.": PRINT

125  PRINT " 'ENTER NEW DATA' SCRO
     LLS THE PROGRAM": PRINT : PRINT
     "TO THE CURRENT INPUT POINT."

130  VTAB 22: HTAB 3: PRINT "**PRE
     SS ANY KEY TO COMMENCE PROGRA
     M**": GET A$: HOME
140  VTAB 3: HTAB 15: PRINT "MENU"
     : HTAB 15: PRINT "----": PRINT
     : PRINT
142  HTAB 5: PRINT "(1) ALLDATA": PRINT
     : HTAB 5: PRINT "(2) SEARCH":
      PRINT : HTAB 5: PRINT "(3) E
     NTER NEW DATA": PRINT : HTAB
     5: PRINT "(4) QUIT PROGRAM": PRINT

150  PRINT : PRINT : PRINT "WHICH
     ONE ... (1-4) :";: POKE  - 16
     368,0: GET A$: PRINT A$:CH =
      VAL (A$): IF CH < 1 OR CH >
     4 THEN 140
155  HOME : ON CH GOTO 160,200,500
     ,600
156  END
160  HOME
165  VTAB 3: INPUT "YOU WANT ALL T
     HE ENTRIES (Y/N) : ";A$: HOME
     : PRINT
170  IF A$ = "Y" THEN 205: GOTO 14
     0
200  VTAB 5: INPUT "ENTER HEADING
     FOR SEARCH : ";X$: PRINT : PRINT

205  PRINT "ENTER :": PRINT : INPUT
     "(S) SCREEN OR (P) PRINTER OU
     TPUT :";B$: HOME
210  IF B$ = "P" THEN  GOSUB 400
215  HOME
245  PRINT "HEADING" SPC( 12)"DETA
     ILS"
250  PRINT "-------" SPC( 12)"----
     ---"
255  POKE 34,2
300  REM
304 ML = 1000
305  FOR I = 1 TO ML
307 K =  PEEK (37): IF K >  = 20 THEN
      GOSUB 700
308  REM
310  READ E$: IF E$ = "END" THEN 3
     35
315  READ T$
320  IF A$ <  > "Y" THEN 335
325  PRINT E$;" " SPC( 2)T$;" ": PRINT

330  GOTO 350
335  IF E$ <  > X$ THEN 350
340 E = E + 1: IF E = 1 THEN 345:E
     $ = ""
345  PRINT E$;" " SPC( 2)T$;" ": PRINT

350  NEXT I: PRINT : PRINT
355  PR# 0: RESTORE : POKE 34,0
358  PRINT "====== THAT'S ALL ====
     ==": PRINT
360  VTAB 23: PRINT "(1) RETURN TO
      MENU  (2) END PROGRAM .. ? "
     ;: POKE  - 16368,0: GET Z$: PRINT
     Z$:CH =  VAL (Z$): IF CH < 1 OR
     CH > 2 THEN 360: HOME
365  HOME : ON CH GOTO 140,600
366  END
367  REM
368  REM .PRINTER SWITCH ON
400  PR# 1: RETURN
410  END
500  REM .NEWDATA
505  HOME
510  LIST 1000,
520  END
530  REM
600  HOME : VTAB 10: HTAB 10: PRINT
     "<<<<PROGRAM ENDED>>>>"
610  END
620  REM
700  POKE  - 16368,0: VTAB 23: INVERSE
     : PRINT "MORE TO COME - PRESS
      ANY KEY";: GET Z$: PRINT Z$:
      NORMAL : HOME : RETURN
705  END
710  REM
800  REM .^^^^^^^^^^^^^^^^^^^^^
805  REM . BY BRIAN STRONG
810  REM . FOR BITS & BYTES
815  REM .^^^^^^^^^^^^^^^^^^^^^
820  REM .
825  REM .DATA STARTS HERE
830  REM .
1000  DATA  MEETING,FEB 5-8PM
1001  DATA  SHEEP,LAMB LOSS 35
1002  DATA  COMPUTER, RENEW SUB TO
      BITS & BYTES
1003  DATA  GARDEN,BEANS SOWN OCT
      5TH
1004  DATA  BOOK,NEW MANUAL ORDERE
      D JAN 10
```

Continued on page 33

MicroBee

After several years of development, the MICROBEE and education have come together. Many educators are already aware of the MICROBEE educational and home computer system.

★ POWERFUL 16K MICROWORLD BASIC
★ UNIQUE CONTINUOUS MEMORY ★BUILT IN SOUND
★HIGH RESOLUTION, HIGH VISIBILITY GRAPHICS
★SERIAL AND PARALLEL PORTS

MICROBEE

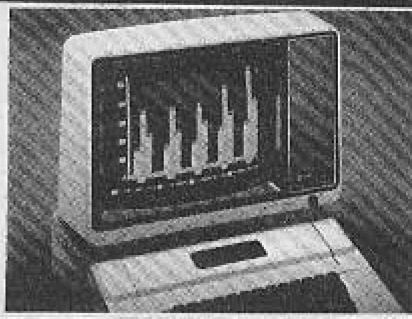
Tax Free Educ.	**Price**
16K	**$525**
32K	**$660**
64K	**$927**

EXPANSION POWER
S100 expansion unit
Disc Drives and CP/M 2.2

MICROBEE a cost effective answer to changing technology

ZENITH GREEN SCREEN MONITOR
ZVM—121 VIDEO MONITOR

Tax Free Educ. Price
$310



MICROBEE DOES MORE

	Normal Retail
16K Microbee	**$829**
32K Microbee	**$990**
64K Microbee	**$1390**
Zenith Monitor	**$380**

Farm management budgeting programs by Agrisoft Ltd.

Word Bee word processing in ROM

OTHER EQUIPMENT AVAILABLE	Retail	Tax Free Educ. Price
TRAK DISC DRIVE FOR APPLE II - 5.25" (looks like an Apple drive - fits on to existing controller)	**$890**	**$590**
TRAK DISC CONTROLLER FOR APPLE II	**$249**	**$185**

FLOPPY DISCS — NEVER TO BE REPEATED SPECIAL

SHUGART 8"; soft sectored; SSSD, SSDD, DSSD, @ $8.95 each (boxes of 10)
8"; hard sectored; SSSD, DSSD, @ $8.95 each (boxes of 10)
5¼"; soft sectored; SSSD, @ $8.75 each (boxes of 10)
5¼"; hard sectored; DSSD, @ $8.75 each (boxes of 10)
CDC 8"; soft sectored; SSSD, @ $7.50 each (singles)

CHECK-POINT COMPUTERS LIMITED
Private Bag, Tawa, Wellington Tel (04) 326-988
Showroom: — 36 Main Road, Tawa, Wellington, N.Z.

Two ways to solve a problem

By MIKE THOMSON

There are two ways to solve your problems with a computer: bring the problem to the machine, or take the machine to the problem. Most business applications require the processing of great amounts of data and despite machines such as the Osborne or HP85 which can be carried around to a degree, most problems are still taken to the computer in the main office.

The calculator, on the other hand, is truly portable, but until two or three years ago all it did was crunch numbers, and then only under what the RAF called a "mark one eyeball". It couldn't do more than display the results of a single calculation.

About two years ago a new type fo calculator appeared, "Alpha numeric programmable". With this instrument you could give it a list of things to do with your numbers. It could retain data, and be programmed to make decisions as well as prompt for data in straight English. This ability places it very close to a true computer in potential. Indeed, it can be more convenient to use than a bigger and sophisticated unit.

I am a supervisor of a photo-process department in a printing plant. We produce printing plates for the production of cartons, labels and books. Each day I have two computational jobs to do. The first is estimating how much it will cost to do a job. The second, how much it cost to do it.

Estimating requires a figure composed of a total of all the items and labour that I imagine I am going to use. Costing requires a

How it runs

Pressing a special assigned key initialises the program. This clears all the intermediate storage registers of any previous data. It counts up all the priced data and compares the total with a master figure. If this is true then the display shows "Data Checked" and proceeds to the text prompt if the printer is attached, or display "item number" if it is not. If the check is false, a warning, "Date Wrong", is displayed intermittently with a beep warning as well as continuously printing the same message on the printer. It is also not possible to continue with the program. This check is made because accidentally pressing say, the statistics key, in normal calculator mode can disturb the data. The text prompt only appears if the printer is attached.

By pressing the alpha keys you can type in a description of the job. Pressing "Zero" exits the mode and prints the message on the printer. The calculator then prompts with the message, "Item Number".

In response I choose from a menu on a card on a table stuck on the back the item I want. (See illustration of menu/scratch pad. Figures outside are register address; inside are my estimates or quotes.) For example, item 30 x 40 film. I input and press run. The display shows "30 x 40 FILM" then asks, "HOW MANY". I input, say 8 pieces, and press run. If the printer is attached it prints out "30 x 40 FILM. 8" and then the calculator display repeats by asking "item number" again.

If the printer is not on, then we go to the same prompt. I input item number and quantity at will and in any order until all my data is completed.

The basis of the program is this: The input of the item number gives the calculator the address of which register to fetch a price.

Say we are calling the example above item 7, 30 x40 film. On the input of 7 the calculator goes to subroutine LBL7, where it is instructed to display "30 x 40 film". This it does by putting the message into the alpha register. It then recalls to the X Reg. the number in register 7, in this case 250.0075. Then it takes the interger of this number, ie 250, and divides by 100 which converts it to 2.50. This is then stored in the stacks. It recalls the number again and takes the fractional part, .0075, which is multiplied by 100 to give .75 (this is storing two prices in one value).

The prompt "How Many" requires an input which is say 8. The two values, i.e. 2.50 and .75, are then multiplied in turn by 8, the sum of each is then stored in the appropriate accumulating register (in this case, material and overhead).

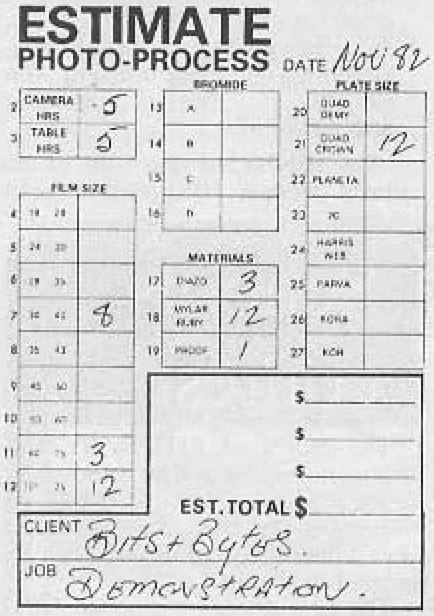
If the printer is attached it will take the alpha message and add to it the input figure for "how many" and print it out in list form (see illustration).

Now comes the single most mind-saving event. I press one assigned key and the HP41 fetches the data for the first column and adds it up, then the second, then across to the final labour material and overhead totals. It then adds these to arrive at the final overall total.

But that's not all! It's remembered all the sub-totals along the way.

If the printer is attached, the HP41 will print the sum of each total out with an appropriate label beside it, all this in 30 seconds.

As I sometimes find myself standing in the middle of a room, the HP41 displays all in sequence with sufficient time for each line to be read easily. The final total remains in the display, and just in case you have missed anything, the press of another assigned key will repeat the display as many times as you would wish.



breakdown of two areas: preparation, which is all handwork, and platemaking which is an automatic process.

These require a separate breakdown into material, labour, and overhead totals. The job total is the sum of all these totals added together. Thus 12 totals are required for each job.

Estimating required in most cases is an over-all total. I was doing this 10 to 20 times a day and came to think of it as "The Head Man's Burden". However, life changed the day I purchased a Hewlett Packard 41 calculator. Up to then anything with COS, TAN or SIN scared me (I still don't know what they are for). But one night with the HP41 and its excellent manual not only caused a domestic incident (I wouldn't come to bed) but also sowed the seeds of an idea that this handful could do me a power of good. Two years, and many programs later I have a

```
*DATA CHECK*
PHOTOPROCESS
   ESTIMATE

THE BITS AND BYTES DEMO-
NSTRATION ESTIMATE:
24 PAGES 4 COL:


ITEMS
------------------------
CAMERA/HRS   0.5
TABLE/HRS    5.0
30/40/FILM:  8.0
60*75/FILM:  3.0
1M*75/FILM:  12.0
CROM/PROOF   1.0
DIAZO PRINT  3.0
MYLAR/RUBY   12.0
QUAD CROWN   12.0
------------------------
                      $
PREPARATION
MATERIAL:          175.55
LABOUR  :           14.30
OVERHEAD:          124.00
------------------------
PR/TOTAL:          313.85
------------------------
PLATES
MATERIAL:          229.68
LABOUR  :           24.60
OVERHEAD:           43.20
------------------------
PL/TOTAL:          297.48
------------------------
TOTAL
MATERIAL:          405.23
LABOUR  :           38.90
OVERHEAD:          167.20
------------------------
ORIG TOTAL:        611.33
------------------------
OUT WORK
SCANNING:          250.00
STEPPING:            0.00
------------------------
EST TOTAL:         861.33
------------------------
```

Illustration of a final print-out

compact portable system that computes all 12 totals and prints or displays them at the touch of one key.

The HP 41 has a memory which can be divided into data - storage registers and program registers. If you allocate say 30 registers to data they will be allocated and addressed from No. 0 to 29.

I have 26 material items and two hourly rates making a total of 28. Each item has two charges to its total, a material, e.g. $2.50 and an

```
PRICE REGISTERS :

R00=  0.0000     R20=  2015.0361
R01=  0.0000     R21=  1914.0360
R02=  1050.0260  R22=  18.3004
R03=  875.0260   R23=  1500.0452
R04=  86.0049    R24=  1230.3625
R05=  137.0078   R25=  1400.0326
R06=  188.0107   R26=  550.0300
R07=  250.0075   R27=  250.0250
R08=  304.0166   R28=  2.0500
R09=  308.0110   R29=  1.7500
R10=  400.0250   R30=  0.0000
R11=  650.0305   R31=  0.0000
R12=  820.0405   R32=  0.0000
R13=  115.0025   R33=  0.0000
R14=  236.0050   R34=  0.0000
R15=  345.0075   R35=  0.0000
R16=  461.0100   R36=  0.0000
R17=  250.0050   R37=  0.0000
R18=  250.0050   R38=  0.0000
R19=  15.0375    R39=  0.0000
```

Print-out of data registers. The prices are entered as one number as described in the text. The registers for zero are for intermediate storage of sums.

(Note that dummy prices are used in these tables)

overhead , e.g. $0.75. These two figures are combined with a decimal point between them. Thus item No. 7 might be 250.0075. This figure is stored in register No. 7. All items are entred into the data registers this way from 2 to 29 (see price registered printout). A total fo 39 registers are used for my program: 28 for item price data and the rest as storage for intermediate results during computation. The rest of memory is program.

The card reader is a unit which fits neatly on the top of the HP41. This unit is almost as clever as the calculator. Among some of its abilities are recording programs, playing back programs, and recording data. All data if you want, but also selectively any particular registers you want.

So with a card or so in your pocket one keystroke can instruct the 41 to down load data on the card which can be removed and the calculator cleared for another computation.

When you're back in the office you can "Play it again, Sam" and re-enter the data into perhaps another program that can further analyse the figures.

When you switch off, the HP 41 remembers all. You don't lose your program or data. It's available the moment you switch on again.

So, what is the HP41, a calculator or a computer? I think the latter, because I have run a program along exactly the same principle on a HP85 and found that while everything moved faster I came away with the same answer, only I had to take my problem to the computer that was plugged into the wall.

While output is only a strip and I only record data on magnetic cards, Hewlett Packard has announced the IL (interface loop) for the HP41 which will allow it to use 80 column printers, a data tape of 130K capacity and talk to other computers, by modem or direct as well as display on a monitor.

But aside from all this, I find the one great thing is I can take my computer to the problem - in my pocket. And use it on the spot.

BBC user group

A BBC micro-computer user group has been formed in Wellington.

Membership is available throughout New Zealand. At present regular meetings are held only in Wellington but this will be extended to any centre where membership rises to a sufficient level.

First of its kind

A new portable computer from Hewlett-Packard — the first of its kind from the firm — features capabilities of larger, more expensive computers, including BASIC language programming power, inferfacing and software.

The HP-75C portable computer measures only 25.40cm (10in) by 12.70cm (5in) by 3.18cm (1¼in), weighs 737, (26 ounces), runs on batteries and retains programs and data when turned off. This portability is designed to let the HP-75C provide desktop-computer solutions anywhere and anytime.

Key features of the new portable computer include:

- 169 instructions, including 147 BASIC commands, statements and functions, and other functions including time and appointment in a 48K-byte ROM-based operating system.
- Built-in Hewlett-Packard interface loop (HP-IL) for communicating with instruments, peripherals, and other computers.
- Software for specific applications such as engineering, maths, and statistics, and general solutions such as electronic spreadsheets and graphics presentations.
- A completely redefinable keyboard for customising the HP-75C for specific applications.

The HP-75C is the first of the firm's Series 70 portable computers.

Schools contest

The results of BITS & BYTES' first competition will be announced in the March edition.

Judging has been completed and the winning programme will be printed in the next issue.

The first prize is a Sinclair ZX81 home computer provided by David Reid Electronics.

Pocket computer power

A pocket computer is how Sharp describes its PC-1500, and the description emphasises the portability of this advanced hand-held. It would seem ideal for a sales rep or for the businessman who likes to take his work home and perhaps would like to tackle business problems while riding in the commuter train or bus, or on a plane.

The PC-1500 measures 195mm by 86mm by 25mm, and has a typewriter-style keyboard and a numeric keypad. With 19K of ROM and 3.5K bytes of RAM (2.6K in the user area), and capability of expansion to 11.5K bytes, the PC-1500 lives up to its classification as a computer. The 26-digit display features upper and lower case letters.

The colour printer attachment gives output at up to 11 characters per second in four colours and in a range of nine sizes. The unit is also a cassette interface which allows the machine to be connected with up to two cassette tape recorders.

A clock function indicates month, date, hour, minute, and second, and a beep function allows it to give an audible reminder to the user at the same time an appropriate message is flashed on the screen.

A memory safeguard prevents the erasing of programs or data, even when the PC-1500 is off.

The CE159RAM/ROM module can be set in three phases to give varying arrangements of ROM and RAM. A program can in effect be locked into this module and a screw taken out, meaning to all intents and purposes that no-one can get at it.

The Sharp PC-1500 with the printer/cassette interface attached.

A random-number function and function keys are useful for games, the random-function being able to generate "playing cards" for blackjack.

Prices in New Zealand: PC-1500, $560. The printer/cassette interface, $436. 4K RAM module, $165. The 8K RAM module, $200. Matrix software board, $260. Interface for larger computers, $459. CE 159 RAM/ROM module, $280.

Calculators

Sharp markets a wide range of its calculators in New Zealand. Some of the more interesting to student readers will be:

- The EL512, with hexadecimal conversion, direct formula entry, a maximum capacity of 128 steps, 15 levels of parantheses, and eight levels of pending operations, expanded memory and 61 factory-programmed functions. It sells for $85.
- The EL-506H, also with a hexadecimal conversion key. It has 48 pre-programmed scientific functions, three levels of parantheses and four levels of pending operations. The price is $49.95.
- The EL-550, retailing at $149, with a thermal printer. It has 46 functions, three levels of pending operations, and 15 levels of parantheses.
- The EL-515 is a 50 function solar-powered scientific calculator. Artificial light will power it, too. The price is $82.

BITS & BYTES will be running short reviews of hand-held computers and calculators from time to time, and some of the above Sharp machines will be looked at in more detail.

Working with a Casio

A review of the Casio Fx-702P hand-held computer by Jamie Walton of Oamaru

I have had my Casio Fx-702P for about a year now and have discovered many of its advantages and pitfalls.

To begin with is the display. As with all other calculators of its type it is rather limited with only a 20 character display. But Casio have done quite well in providing display controls, giving a 62 character auto scroll and some handy program output commands.

The keyboard looks rather complicated and bewildering to begin with but after a while the purpose and application of the commands above and below each key become clear, though they are not the advertised one-key commands.

The lay-out is good considering there are 65 keys to fit onto the calculator. Alphabet, program and mathematical (excluding *,/,+,-, exponent) commands are on the 35 white keys, to the left, and the numbers, mode and program selection keys on the 30 black keys to the right.

Programming the Casio involves moving between 'RUN' and 'WRT' modes to write the program and then check to see that it runs OK. The 'TRACE ON' mode option is handy for debugging ailing programs.

The commands are the same as standard BASIC, but for those of you, like me, who already know some BASIC DON'T assume that the syntax is the same, as Casio takes some liberties with various commands causing some interesting errors in strange places. Command sizes are also shortened down to a minimal 2 to 4 letters.

Finally some people will be interested in the speed of the Casio in relation to some other computers, so here are some approximate results for various computers: Casio Fx702P approx. 10 sec, Sharp PC-1211 approx. 1 min. 55 sec., Atari 400, approx. 1 sec., Apple II + less than 1 sec.

So you can see that the Casio holds its own with machines of its own size but has a long way to go to catch up with the bigger computers.

Overall I believe that the Casio is a reasonably good buy and that anyone looking for a hand held micro of this price range ($300) would do well with the Fx-702P.

The Quadratics Solution program below is an example of how the Casio can be put to good use for students. If an EER-3 is displayed, unfortunately the equation given is impossible to solve.

Quadratics Solution Program

```
10  PRT "EQN: AX2 + BX + C =
    0"
20  INP "INPUT (A), (B), (C)",
    A,B,C
30  D = (-B+ SQR (B2-4*A*C))
    /(2*A)
40  PRT "X1 = "; D
50  E = (-B-SQR (B2-4*A*C))/
    (2*A)
60  PRT "X2="; E
70  END
133 STEPS
```

PC-1251

• This is a wallet size Sharp computer, about to be introduced in New Zealand. It has 24K bytes of ROM and 4.2K of RAM, INCLUDING 3.7K of usable area. Eighteen reservable keys for fast programming and 18 definable keys for quick program recall help with the computer's extended BASIC.

Optional features include a printer/microcassette recorder designed to fit the machine and a 24-digit thermal printer.

The printer/cassette recorder is designed to hold the computer in a unit 205mm by 149mm by 23mm.

The computer will retail at $295, and the printer/microcassette recorder will probably retail for about $370.

From Page 1

the 8 colour constraint... and its solution through POKEs.

The advertising in the UK in spring, 1981, was, in my view, overstating what could be done on a VIC with a PAL television at that time. This is not to seriously question the reputability of Commodore. As my article suggested every manufacturer prefers to avoid portraying their graphics "warts and all". As consumers let us be wary.

Pip Forer

Alien Destroyer

Dear Sirs,

I hate to admit it, but there is something wrong with my program, Alien Destroyer, published in the November issue of "BITS & BYTES". First of all the instruction is CALL -151, not CALL 151, (to enter monitor).

Second, part of the shape table is incorrect. Essentially it is O.K., but make these changes so that the program will run correctly.

When in monitor (CALL -151) type the following:

```
IDFC:
    03 00 08 00 14 00 24 00
    24 25 25 2D 2E 2E 36 3F
    3C 37 3F 00 4D 6D 31 3F
    DF 3F 2E 4D 09 F5 IF FF
    0E 4D 05 00 49 6D 3A DF
    BF 6D 49 35 FF FF 37 0D
    4D 0D 05 00
```

Since the program was published I have made some good changes to it. The game is now controlled by paddles which makes movement much faster and allows you to move whilst firing. If anyone would like a copy of this I would be glad to send a listing. Or, if you send me a DOS 3.3 disk I will save the version onto it and return it, ready to run. All I ask is that you include return postage for the disk or listing (i.e. stamps). This new version has been tried and tested by the school's computer club and they say it is much more interesting and fun to play now.

By the way, it's Otumoetai not Otumoeta.

Richard Hobbis.

Richard is a pupil at Otumoetai College, Tauranga.

'I am the Sirius 1, the No. 1 choice in personal computers'

By KERRY MARSHALL

This computer came to my attention early in 1982 when the first advertisements appeared in various UK magazines. The features offered seemed revolutionary but surprisingly there was no mention of the Sirius in the USA press. The Sirius is manufactured and marketed in the US under the name Victor.

The UK distributor, ACT, reports that in the first few months of marketing, the Sirius has become number three in terms of dollar value of sales. There are also some 70 software houses writing systems specifically for the Sirius.

If this trend is followed in NZ, the machine promises to be a big seller.

The machine arrived in one large box, into which it amazingly was able to be replaced when I had finished. This is a handy idea if you need to move the computer frequently.

Unpacking and assembling is straightforward using the manual. The keyboard and screen connect

Microcomputer Summary

Name:	Act Sirius 1.
Processor:	Intel 8088 (16-bit).
Ram:	128k expandable 1MByte.
Input/Output:	2 Serial, 1 Parallel (Programmable)
Display:	25 lines x 80 characters (9 x 12 in a 10 x 16 cell) or 50 lines x 132 characters; programmable character font.
Graphics	800 x 400 pixels.
Sound:	digital voice coder/decoder.
Cost:	approx $9000.
Languages:	BASIC 86 (compiled and interpreted); MS COBOL and CIS COBOL; MS PASCAL, FORTRAN, PL/1.

to the processor/disc unit with one cable each, and only one power cord is needed, saving hassles with plugs and leads.

Screen

The Screen is designed so that it can be positioned to the operator's requirements — it can be tilted and swivelled as needed. Coupled with the detached keyboard, this allows flexibility for setting up.

The immediate and memorable impression one has of the screen is the graphics. With a resolution of 800 by 400 pixels, the Sirius can display remarkable pictures, diagrams and graphic material. The demonstration included a 3D simulation of the Sirius which was rotated through 360 degrees.

With the large memory and speed of the processor, there are opportunities for design and engineering systems to be implemented on this microcomputer.

Several character fonts are available and, along with other features, these are programmable — they are read from disc at boot time so that a user can select diferent fonts depending upon individual applications.

Scientific symbols, Greek letters and other alphabets can be used. Standard text gives 80 characters on 25 lines, but using a compressed character set allows a full 132 characters to be displayed.

Keyboard

One of the featured aspects of the Sirius design is the "Ergonomic" keyboard. When I first saw the keyboard I wondered about a licence to drive it. But it soon became clear.

The keyboard has five groups of keys. The main group is a standard Qwerty layout, but each key can have three functions — upper and lower case and also special symbols. These are controlled by the shift and ALT keys. A group of keys to the left of the main group provide escape, intensity, reverse-video, under-lining and auto-repeat modes. The third group of keys controls functions normally allocated to buttons, switches, or control keys or codes.

The brightness and contrast of the screen are simply set, as is the volume of the loudspeaker, using the keyboard. Clearing the screen and returning the cursor to the home position are also key functions.

A set of numeric keys is useful for entry of numeric data and can also be used in the calculator mode, although this requires the use of the Calc key.

The final group of keys is the function keys, above the main keys. The seven keys can be used under program control and in the demonstration word processor, they were used for functions such as saving the text, or quitting.

Functions can be set with the character font and loaded at boot time, providing great facilities for customising the system for **specific applications.**

The Sirius is, of course, one of the first 16-bit microcomputers and uses the Intel 8088 chip — the same as the 8086 except that the 8088 can interface with the eight-bit chips.

The Sirius can use the IBM personal computer operating system but the system supplied had CP/M-86 and MS-DOS. The use of CP/M-86 allows access to a wider range of software

The RAM is standard 128k expandable to 1MB (512k internally) and should provide ample expansion capability for most commercial and specialised users.

In designing the Sirius, Chuck Peddle has allowed for the common languages to be implemented on the machine without the need for expansion extras. Standard are Microsoft BASIC, compiled BASIC, COBOL, FORTRAN, PL/1 and PASCAL. For software developers, and those organisations with a variety of needs (records, scientific) and personnel, the Sirius can be used for a range of applications.

Sound

Incredibly, the Sirius can speak — it has a sample program that actually greets you with the words, "I am the Sirius 1, the number one choice in personal computers." Using the CODEX system, speech can be generated and stored. This is a feature which will become more in demand in the next few years as users come to recognise its usefulness.

Discs

The Sirius has two 5¼ floppy discs as an integral part of the processor. Using a variable speed approach, the standard capacity is 600k per drive, with an optional capacity of 1.2MB per drive. For a microcomputer this is a big advance in disc storage, and with the Winchester drives available, 10MB, expansion of capacity is no problem.

The discs under review operated reliably but I was not impressed by the noise level during operation of the drive.

Networks are all the rage at the moment and the Sirius has been designed with networking capabilities in mind. Whether networks become widely established or not, the Sirius enables a user to use the computer as an intelligent terminal.

Development in the UK is aimed at providing communications software for the Sirius to talk to ICL and IBM mainframes.

Documentation

Operator's manual and the programming manual (COBOL, BASIC, CP/M-86) were provided with the review system. The operator's manual was clear and easy to follow, while the programming documentation was well set out, indexed, and had good examples.

The review system was supplied by Barr Bros Computers Ltd, P.O. Box 177, Papakura.

Small business computing

By JOHN J. VARGO

Last month's article discussed a practical approach to sizing a computer system to meet your business needs. Also covered were the costs involved in acquiring a computer system, including many of the hidden costs.

Having done most of the groundwork, it is now time to prepare your Request for Proposal (RFP), evaluate the replies from vendors and compare the results to the financial benefits of the proposed system.

THE REQUEST FOR PROPOSAL

The next stage in your selection process should be to put together a formal document called a Request for Proposal (RFP). The RFP serves the purpose of communicating clearly to all vendors the same information: your needs that must be fulfilled, and the information you require to make your decision.

The result of a clean and concise RFP will be reasonably uniform proposals which will allow you to do a more objective evaluation.

The document should contain the following:

- Background information of your company
- A listing of each business application you would like to implement together with the specific information gathered during the feasibility study i.e.:
  - number of records
  - size of records
  - numer of transactions and size
  - reports required and frequency
  - other special requirements decided on by the study team.
- Ground rules including due date for proposals, projected delivery date, desired date of installation, and contractual requirements.
- Personnel and training requirements
- Reporting and information needs
- Mandatory requirements, which if a proposal does not include them, the proposal will not be considered for selection (e.g. must be an on-line system, must support the inventory control and auto order function, require colour graphics, etc).

Once this Request for Proposal is completed, send it to all the potential vendors and give them an opportunity to ask any questions they might have. This will increase the quality of the proposals you receive. The RFP should be concise but complete. Also, bear in mind that many small computer dealers/vendors are not used to having their customers so well organised, and it may require a bit of prodding on your part to get a complete proposal from some fo them. Do not let this put you off, be diligent and remember that the information system of your company for the next five years is at stake.

BENEFITS OF USING A SMALL BUSINESS COMPUTER

As discussed in a previous article, a small business computer will not end all the headaches of running your own business, nor will it make a balky accounting system run smoothly.

There are, however, many financial benefits that can accrue from computerisation. If you are currently running your accounting system and other information needs manually, your efforts will be greatly reduced.

You can also expect less errors, and elimination of many of the repetitive steps of copying and posting. Even more importantly, you will be able to obtain information that is currently unavailable because of the cumbersome nature of the manual approach.

A computerized system may also be able to help you:

- shorten the billing cycle
- reduce amount of stock carried
- increase sales
- control costs
- manage your cash better
- plan and control your business strategy
- reduce labour expense.

How much benefit you will derive from each of the above will be determined by which applications are computerized, and the particular circumstances that prevail in your company. The data that was gathered when doing the feasibility study for each application area will be invaluable when determining the potential financial benefits.

A closer look at the specific application areas at this point should prove enlightening.

General Ledger application may benefit from saving time consuming posting and searching for errors. Together with reducing the month end or year end closing cycle, you will be able to have timely balance sheet and profit and loss statements at the end of every month.

In addition, you could benefit from regular cash flow statements and comparison of budgeted figures to actual on a monthly and year-to-date basis. This will put you in a position to plan better and take advantage of business opportunities that arise.

If your business is a retail, wholesale, or manufacturing business then inventory will be a primary asset and a candidate for improved control. This system is usually triggered by an order from a customer and then follows through many steps from typing shipping labels and recording back-orders, to computing salesmen's commission, discounts and invoicing.

Many of these steps can happen automatically once the initial order is entered into the computer. This step alleviates the time-consuming procedures of manual operation with all the associated opportunities for error. Improved inventory control may increase sales by avoiding out-of-stock conditions, yet still reduce the overall stock carried through more timely purchasing and reduction of slow moving items.

By speeding up the billing cycle you will gain better control over accounts receivable and improve cash flow. With the computer you will also be in a position to more precisely control whom you sell to on credit as well as keep a more accurate record of slow payments. This will allow you to follow up on them before they become a bad debt.

Accounts payable is an often overlooked area when it comes to improvements. The discounts lost because of a long payment cycle, together with tighter control of cash disbursements, can make this a very profitable area to computerize.

If the business is tight on cash, the additional control through creditors agings and cash flow reports may allow a reduction of your overdraft and other borrowings. The resulting savings in interest and the improved relations with the bank manager will be beneficial in future business dealings.

The benefits of word processing are widely known in terms of generation of form letters, standard contracts, and revision of often used documents. Because of the speed with which documents can be generated, the use of a word processor may increase your sales by allowing you to quote on more jobs than you are presently able to, and to do so on a more timely basis.

If used in conjunction with a financial modelling package and a data base management package, it is also possible to model the bidding and estimating process. Previous quotations can then be called upon for comparison. All of this will allow you to make more timely, accurate, and competitive quotes.

Business modelling can benefit the company immediately, as in the above example. More often, the modelling process is used in a somewhat longer term environment. Setting up budgets for comparison to current actual figures is a good example. Creating planning models for the next 6 months, year, or five years, — including assumptions about the economy, inflation and foreign exchange rates — can be very useful as well as enlightening.

With many of the financial modelling packages, once your model is set up, it is just a matter of changing a few numbers to see the effects of the changed assumptions immediately. No laborious recalculations are required since the computer does it for you.

This type of application may not show up immediately in your net profit or reduced overdraft, but it will give you the long term edge often needed to survive in a highly competitive environment of an economic downturn.

These are some of the applications in which a small business computer could benefit a business. The list is not exhaustive, and the benefits derived by any particular business are dependent on the care with which the original analysis and implementation is carried out.

Alternative graphics languages

By PIP FORER

A couple of issues ago we reviewed the importance of the BASIC with which your machine came endowed. This defined your ability to easily cope with graphics programming. There were BASICs and basics. Supposed now you want to try a new approach to generating graphics beyond BASIC and beyond utilities. Or suppose you have found that for other reasons, say speed, BASIC was no longer suitable for you but you still wanted graphics. What is there on offer?

The obvious answer is to move to another language and these are now numerous. In the last two years the languages released for cheap microcomputers have multiplied tremendously. Cobol, Ada, and even Dynamo have now joined the list. There are even some machines, such as the Z-GRASS-32 and the Jupiter which come without BASIC as their standard language. The first uses Z-GRASS, while the latter is a $250 UK machine using Forth. This short conclusion to my series roams over some of the languages available with microcomputer graphics capabilities. In particular I will mention PILOT, PASCAL, FORTH (GRAFORTH), LOGO, C-MAC and Z-GRASS as a range of different languages that support graphics applications in different ways. These languages have two distinct characteristics that differentiate them.

Firstly, some compile and others interpret. Pascal is a compiled language that converts your source (written) code into object (machine) code before you can run the program. It is also the basis of the PILOT implementation on the Apple. Most of the other languages are interpreter based. There is no compiling stage and they are very interactive. You may be aware that compiling BASICs are available. These give you an element of the best of both worlds... interactive development and faster execution of a developed product. The Poly system comes with a BASIC compiler as part of its operating system and there are several BASIC compilers available on other machines. BASIC compilers only go so far in changing how you work. They still leave you with BASICs imperfections and they are still slow at the end, albeit not so slow.

Secondly, some languages are structured. A structured language allows the user to define *procedures.* An operation, such as drawing a square, can be defined as a procedure and this new procedure given a name, say SQUARE. SQUARE now effectively becomes a new verb in the language. This makes for civilised and orderly programming. Most of the languages we are looking at are structured, and procedure based. Having a structured language is a very worth-while move.

So, if you move beyond BASIC for graphics where do you go?

One possibility is simply to throw away the high-level languages and get down to machine code or assembler. This will give you great speed and make you independent of different compilers or interpreters, although probably far more machine dependent. There is an excellent book on fast arcade graphics in assembler for the Apple II by Jeremy Stanton for those who are toying with this idea. The main hurdle is that getting in to assembler from a simple BASIC is a rather drastic step.

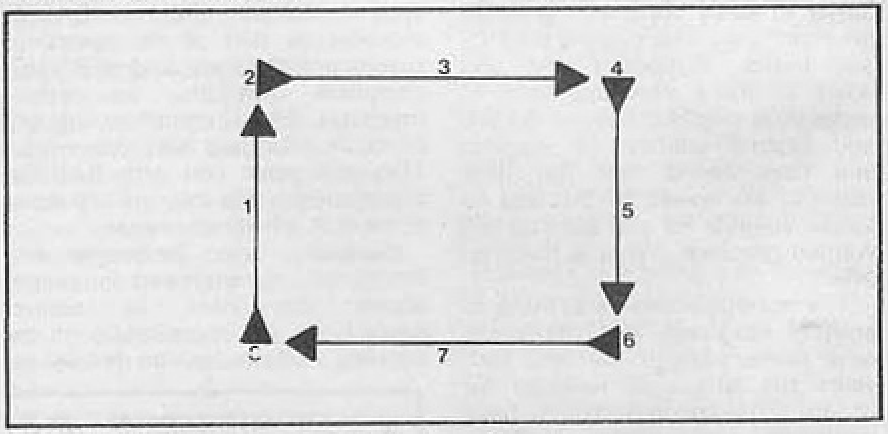
You may, therefore, look around for something which is a little less traumatic as a step up from BASIC.

Where you look will depend on what aspect of BASIC you wish to improve on and just what your applications of graphics are.

For a start if you are in education you may look at better languages for achieving certain educational aims. Of these PILOT and LOGO are best known and both have graphics-oriented versions available on microcomputers. Perversely these two languages represent entirely different ways of teaching. PILOT was designed for delivering tutorial C.A. I. (Computer Aided Instruction) in which a prerequisite is a fairly formal, albeit flexible, learning path prespecified by the author for the student. LOGO was designed with a philosophy that discovery was the best route to acquiring thinking skills, particularly in maths, and it is consequently designed as a medium for experimentation.

For PILOT, graphics varies on different machines. On the Apple it consists of two sorts that can be used in PILOT programs. There is a set of editing and presentation routines that allows the user to create screen pictures in "canvas" mode, that is simply specifying a background diagram. The PILOT available in New Zealand is soon to be superceded and has a pretty unexceptional screen creator for this, which further suffers from incompatibility with any other way of creating backgrounds. However, it has a very good character editor for creating standard-sized block shapes, shapes and letters. This can be used for fast animation, but with the usual requirement for cunning and endeavour that character graphics impose. Graphics on this version of PILOT certainly enliven the usual tedium of a PILOT session, but they are slow and rather limiting.

LOGO is also not a fast language but it is more interesting. Partly this is because it is more interactive. Mainly it is because of the use of turtle graphics. Now turtle graphics are not unique to LOGO (see my comments on Pascal) and LOGO is far more than a wandering member of the Chelonia: it is quite a sophisticated language behind its graphics. But turtle graphics are what have captured the imagination. Turtle graphics allow a visual approach to problem solving, program writing, and debugging. Their essence is a little turtle that moves around the screen under the user's control. The turtle can be set down at any point and told to go in a given direction for a certain distance, turn left or right, and so on. This works well for first-time graphics users (such as those in primary schools) since one can identify well with the turtle and its actions. You need far less knowledge of co-ordinates and such than to operate a normal "plot" based program in BASIC.



Drawing the square. In LOGO the triangle represents the turtle and the numbers the steps given in the text.

The friendliness of the turtle is complemented by the availability of named procedures in LOGO. This is exampled by the need to draw a square as in figure 1. You can do this by telling the turtle

```
FORWARD 10
RIGHT 90
FORWARD 10
RIGHT 90
FORWARD 10
RIGHT 90
FORWARD 10
RIGHT 90
```

in LOGO. This will draw a square with sides of 10 units length. You can make this into an easily repeated operation by defining it as a procedures as in

```
TO SQUARE
REPEAT 4 (FORWARD 10
RIGHT 90)
END
```

The second line of this is just the FOR/NEXT loop equivalent of the preceding eight statements. The first and last line define this all as a procedure. In future any statement SQUARE will access this procedure and draw a square. With a set of procedures for different shapes the user gets to draw various scenes with the aid of a vocabulary of procedures. My favourite example of this is in Papert's "Mindstorms", where two procedures, PETAL and STALK, are set up. Then a new procedure TO FLOWER involves combining these procedures to draw a flower. In turn TO FLOWER becomes a component in TO GARDEN in which a variety of flower shapes are drawn on the screen as a flowerbed.

LOGO has become a cult in many American schools and a glance at our own primary school maths perscriptions shows why. It is worth finding more out about. "Bits and Bytes" will be reviewing one version shortly and both "Byte" (8/82) and "Creative Computing" (10/82) have recently carried reviews and in-depth commentary.

However, LOGO (in the confusing number of versions now available) was designed as an aid to learning, not as a graphics language as such. Where do you go if what you want is spectacular graphics capabilities from a language? The answer is varied. There are one or two arcane languages just designed for creating artistic effects. The disk Fire Organ is a trailer for a language C-MAC which I have never actually seen advertised but which creates some very pretty and fast effects. Occasional interviews in the microcomputer press reveal new work in this area. By far the best known approach however has been DeFanti's Z-GRASS language designed specifically for good graphics. This was initially designed for larger machines but has now been released by Astrovision on its own machine, the Z-GRASS-32. Since it is designed especially for graphics 2-GRASS allows you to handle parts of a picture better, draw things with more speed and so on. It is however not widely available on other small systems.

More common and growing in popularity is FORTH, of which the Apple based version GRAFORTH is the most graphic example. FORTH is a structured language like LOGO but its syntax is far less friendly. At times it is pithy in the extreme. The number of FORTH users is growing and those encountering it for the first time either like it or hate it. It is fast, in the Apple case 10 times Applesoft speed for drawing. The structured design allows you to create a library of typical graphics procedures which can speed programming. For instance you frequently need to scale data values into an available space on a screen. Write the general procedure to that once and you have a utility available in any other program. FORTH is not simply a graphics language but its characteristics make it very sueful if you want to produce fast graphics for particular reasons. It is not fast enough for good arcade games though.

Another language, the final one, and not specifically a graphics language, is Pascal. However, we should mention it here as the second most widely used language on the most numerous educational system currently (December, 1982) in New Zealand. That is the Apple II. Graphics on the Apple Pascal implementation are rather tacked on to the central idea of producing a better and more powerful language than BASIC. Commentators seem almost unanimous in viewing Apple Pascal as unfriendly. It does have some useful (single screen) graphics procedures though, including both co-ordinate and turtle graphics, mixed text, and graphics, the ability to define an active "viewport" on the screen and a powerful if not super-fast form of block graphics. You would not stumble on Pascal as a graphics language, I think, but if you were using Pascal graphics can be used to effectively augment more powerful programming.

That just about wraps up this brief series. As you can see, a lot of options are open to graphics users. These options have only just started to expand. During the course of writing these notes it has become increasingly easy to add video on to computer graphics through videotape and videodisks. Chips to allow various machines access to sprite graphcis (player-missile graphics on the Atari) have appeared. Most significantly the falling cost of memory and processing power has opened up various new peripheral options. Cheap, 128K, add-on memory boards are becoming more and more common, giving fast access to more and more graphics. Boards are appearing tailored to do fast crunching for 3-D graphics or to allow humble machines to create 512 by 512 high quality pictures. Among the claims of such plug-on boards is the ability to draw 1,500,000 pixels a second. More and more is possible at less and less cost. For those interested in cheap computer graphics there are exciting times ahead.

Pip Forer is extending his series on graphics to cover hardware extensions in the next issue of BITS & BYTES.

Time for computer studies in School Cert?

With increasing numbers of students at high schools taking courses using computers, and a greatly increased spending on school equipment it is time that computer studies became a School Certificate subject.

Suitable qualified teachers are available and have in fact, offered to write both suitable syllabuses and examination papers, not only for School Certificate but for a continuing course through each level at high school.

At present, some schools offering computer studies do so at third and fourth form levels, have a year's gap while pupils take an alternative School Certificate subject, and then continue with sixth and seventh form studies. In this situation, the lack of a School Certificate examination is little short of ridiculous.

The examination would give a uniform level of attainment, recognisable by employers as opposed to the widely varying courses offered in different schools.

The appropriateness of developing new skills is very evident with the increasing unemployment amongst our school leaver population. The Commission for the Future indicated strongly that one area which would have an increasing number of job opportunities is in the area of computer based industries.

I would like to know:

- What are the criteria for the introduction of a new School Certificate subject?
- What, if anything, has been done to implement Computer Studies as a subject for School Certificate?
- What steps need to be taken to ensure that Computer Studies becomes a School Certificate subject within the next academic year if possible, or within the near future?
- How do other people involved in the computing field feel about this matter?
- Comment would be appreciated.

We need to be looking towards the future to develop an appropriate education for our children now.

P. J. PARK,
Dunedin

Pick up your entry form from these participating VIC 20 Dealers.

NORTHLAND
Garnet Keene Ltd
36–40 Rathbone Street
WHANGAREI
Phone 784-999

Mid-Northern Services Ltd
No 1 State Highway
OHAEAWAI
Phone 7835

AUCKLAND
Adair Home Appliances Ltd
3 Sunnyhaven Ave, Beachhaven
AUCKLAND
Phone 439-089

Andersons Electrical Ltd
965 Dominion Rd, Mt Roskill
AUCKLAND
Phone 697-283

Armed Forces Canteen Council
Military Camp, Papakura
AUCKLAND
Phone 298-5487

Ashford Television Ltd
168 Kapa Rd, Onkei
AUCKLAND
Phone 583-293

The Byte Shop
1 Fort Street
AUCKLAND
Phone 32-680

Computer World
Cnr Lorne & Victoria Streets
AUCKLAND
Phone 31-394

Farmers Trading Company Ltd
Hobson St
AUCKLAND
Phone 799-220

K Road Video & Computer Co
66 Pitt Street, Newton
AUCKLAND
Phone 399-655

Micromax
P.O. Box 33954
TAKAPUNA
Phone 479-8024

Organ World of Music Ltd
Como St Entrance — Shore City
AUCKLAND 9
Phone 494-472

Public Service Investment Store
Parkway Arc, 54 Hurstmere Rd
AUCKLAND
Phone 493-157

Regers Home Appliances Ltd
238 Broadway, Newmarket
AUCKLAND
Phone 543-804

Salcom Electronics
2a Basin View Lane, Panmure
AUCKLAND
Phone 577-199

Shore & Video TV Services
144 Glamorgan Drive, Torbay
AUCKLAND
Phone 489-979

Sound Centre Ltd
Civic Centre Arc, Browns Bay
& Huron St, Takapuna
AUCKLAND
Phone 479-6056

Southmark Electronics Ltd
100 Anzac Avenue
AUCKLAND
Phone 798-078

Supatech Electronics
430 Mt Eden Road
AUCKLAND
Phone 605-216

Telerad Ltd
68 Galway Street, Onehunga
AUCKLAND
Phone 600-505

Mr R. Thornton
6 Portage Road, Papatoetoe
AUCKLAND
Phone 278-3185

John Walker Music Ltd
60 O'Shannessey St, Papakura
AUCKLAND
Phone 299-8827

WAIKATO/KING COUNTRY
Busy Bee Computing Ltd
Cnr Mannering & Dreghorn Pl
TOKOROA
Phone 65-953

Computer Link
Hikurangi House
TAUMARUNUI
Phone 6490

James Electronics Ltd
328 Pollen Street
THAMES
Phone 86-893

Dollar Save Furniture
Tristram Street
FRANKTON
Phone 82-987

Doug Morris Ltd
Great South Road
HUNTLY
Phone 59-113

Muir Radio
Arawa Street
MATAMATA
Phone 7458

Technology Resources Ltd
8 Thackeray Street
HAMILTON
Phone 393-601

Waikato Computer Centre
Princes Street
HAMILTON
Phone 393-416

BAY OF PLENTY
Compucation Systems
50 First Avenue
TAURANGA
Phone 88-433

Hamilton Sanyo
57 Eruera Street
ROTORUA
Phone 87-146

Jackson & Booth Ltd
Islington Street
KAWERAU
Phone 7030

HAWKES BAY
Progressive Computer Systems
Dalton Street
NAPIER
Phone 54-444

Timms Business Equipment Ltd
302 West Eastbourne Street
HASTINGS
Phone 69-906

Timms Business Equipment Ltd
Tennyson Street
NAPIER
Phone 54-250

GISBORNE
Triad Electrical Ltd
348 Palmerston Street
GISBORNE
Phone 5258

TARANAKI/WANGANUI
Lamberts Ltd
49 Liardet Street
NEW PLYMOUTH
Phone 83-667

Stewart Appliances
134A Victoria Avenue
WANGANUI
Phone 52-700

Taranaki Micro Electronics
Centre Crt Bldg, Devon St East
NEW PLYMOUTH
Phone 84-067

MANAWATU/WAIRARAPA
David Brice Electronics
47 Kimbolton Road
FEILDING
Phone 37-141

Collertons Computer Centre
170 Queen Street
MASTERTON
Phone 83-348

Leader & Watt Ltd
153–163 Rangitikei Street
PALMERSTON NORTH
Phone 84-934

Viscount Electronics Ltd
306–308 Church Street
PALMERSTON NORTH
Phone 88-696

Wairarapa Camera Service
42 Queen Street
MASTERTON
Phone 84-697

WELLINGTON
Einstein Scientific
177 Willis Street
WELLINGTON
Phone 851-055

Ice Clear TV Services
187 Main Road
UPPER HUTT
Phone 282-475

Jenness Appl. & Music Centre
173 High Street
LOWER HUTT
Phone 691-145

Office Requisites Ltd
Cnr W/Taylor & Featherston Sts
WELLINGTON
Phone 721-902

Public Service Investment Store
Whitmore/Featherston Sts
WELLINGTON
Phone 722-123

NELSON/MARLBOROUGH
Harkt Home Appliances
42 Charles Street
BLENHEIM
Phone 88-999

Personal Computer Systems
37 Torlesse Street
NELSON
Phone 79-382

Ross & Staig TV Services Ltd
58 Collingwood Street
NELSON
Phone 80-387

CANTERBURY
The Computer Centre
593a Colombo Street
CHRISTCHURCH
Phone 793-428

Crew Video Services Ltd
81 Ivory Street
RANGIORA
Phone 6200

Good Time Enterprises
200 Manchester Street
CHRISTCHURCH
Phone 797-811

SICOM Ltd
66 Oxford Terrace
CHRISTCHURCH
Phone 796-259

Smith & Church Electrical Ltd
Cass Street
ASHBURTON
Phone 89-019

David Wells Ltd
High Street Arcade
Hereford St
CHRISTCHURCH
Phone 797-228

Cliff's Business Equipment
40 Stafford Street
TIMARU
Phone 44-241

OTAGO/SOUTHLAND
Andersons Electrical Ltd
39 Helwick St, Wanaka
CENTRAL OTAGO
Phone 7833

Eclipse Radio & Hobbies
134–136 Stuart Street
DUNEDIN
Phone 778-102

Eastern Southland Computers
C/- Robin Isaac, Medway St
GORE
Phone 5710

Ken France Electronics
40 Tarbet Street
OTAGO
Phone 8021

DES Business Systems Ltd
135 Don Street
INVERCARGILL
Phone 84-448

Otago Business Equipment Ltd
Cnr Liverpool & Bond Streets
DUNEDIN
Phone 776-227

Shand Vic House
24 Bremner Road, Fairfield
DUNEDIN
Phone 881-432

Commodore COMPUTER

and

BITS . . & . . . . . BYTES . .

NEW ZEALAND'S PERSONAL COMPUTER MAGAZINE

INVITE YOU TO WIN A . . . .

COMMODORE VIC 20 COMPUTER–FOR THE WHOLE FAMILY!

FIRST PRIZE

World renowned VIC 20 — Commodore's brilliant home computer is at your dealer's today! It's a family machine — with something for everyone ... play the best video games available ... or teach yourself a language ... write your own programmes ... even use it to budget for that new car! (And that's just a tiny sample of what it can do!)

VIC 20 plugs straight into your television set and comes complete with

- colour and sound
- full size typewriter keyboard
- graphics capability
- easy to learn BASIC language
- easy to follow instruction manual
- full service and support

Take a look for yourself! The affordable Commodore VIC 20 is at your dealer's today.

HOW TO ENTER

Simply tell us in 300 words or less

"WHAT I COULD DO WITH A VIC 20 HOME COMPUTER".

Entry is open to all ages. Neatness and presentation will count.

PRIZES

The winner of the contest will receive:

*** a Commodore VIC 20 Home Computer**
*** a Commodore Datasette**
*** and software**
to a total value of over $1000

Six runners up will each receive a computer book courtesy of Bits & Bytes.

CONDITIONS OF ENTRY

1. Each entry must be accompanied by an official entry form properly completed. These forms are available only at your authorised Commodore VIC 20 dealer (a full list of participating dealers is printed on the opposite page).
2. Post in an envelope (one entry per envelope) to "VIC 20" competition, Bits & Bytes Magazine, P.O. Box 827, Christchurch.
3. Judging will be by Commodore Computer (NZ) Ltd. The judges decision will be final and no correspondence will be entered into.
4. All entries become the property of Commodore Computer (NZ) Ltd and the right to use entries for promotional purposes is reserved by them.
5. Employees and their immediate families of Commodore Computer (NZ) Ltd, their dealers and Bits & Bytes magazine are not eligible to enter.
6. Prizes are not transferable or redeemable for cash.
7. Closing date for entries is 31 March 1983 and all entries must be received or postmarked by that date.
8. All prizewinners will be notified by mail and the names of all winners will be published in the May issue of Bits & Bytes.
9. Only one entry per person will be allowed.

The sound of music

By B. M. BULLEN

This article deals with making music on your VIC-20, ranging from simple tunes through to a 3 part round.

The basics of simple tunes are covered quite well in the personal guide that comes with your VIC. If you haven't spent much time on music yet you may like to re-read pages 67 to 79 (beware page 75 and 76 should come after pages 77 to 79) before trying the ideas in this article.

If you are interested in experimenting with music start with a simple tune. If like me you're not an expert on music and its "little black dots" you may need the help of a friend to translate it for you. I got the music teacher at my school to find and translate the tune I will use for my demonstrations in this article. For the timing I asked him to write the shortest note as 1 and then all the others as multiples of that note. Here's part of line 3 as an example.

c, 2, c, 1, d, 1, e, 1, f, 1, g, 2, g, 2, g, 2, a, 2

This I then turned into the right POKE numbers as given by the table on Pg 73 of your guide. You can see the end result as the data at the end of listing 1. Listing 1 shows my first program. Enter it on your VIC and run it. Didn't take long did it? Obviously it needs modification. Try adding

```
20 INPUT"ENTER TEMPO";T
```

and changing line 130 to read

```
130 FOR C=0 TO D*T: NEXT
```

This allows us to alter the tempo of our tune at will. Try it for different tempos.

Now for a few "refinements". First we can add a bit of depth to our sound by using all three voices. Add these lines to your program

```
100 POKE S1, N
120 POKE S3, N
140 POKE S1, 0
160 POKE S3,0
```

Run this and note the difference.

You can make it sound more like a piano by making the sound die away rather than just stop. Add

```
135 FOR VL=15 TO 0 STEP -1:
POKE V, VL: NEXT
175 POKE V, 15
```

Try this version. We cna make the tune repeat by adding

```
40 INPUT"NO OF TIMES
"JNR ;NR
200 RESTORE: NEXT R
210 POKEV,O:END
```

We could even play the tune backwards. To do this requires some major modifications to our program. First we will read all the notes and their duration into a 2-D array N(I,J). For those unfamiliar with arrays we are basically setting up a table in the VIC's memory that looks like this.

Column		0	1
Row	0	225	2
	1		
	2		

and so on. The first note is stored at N(0,0) and its duration at N(0,1). The third note is stored at N(2,0) and its duration at N(2,1). Listing 2 shows the program to do this. Run this - it should sound the same as

LISTING 1

```
READY.

10 V=36878:S1=V-4:S2=V-3:S3=V-2
30 POKE V,15
80 READ N,D
90 IF N = -1 THEN 200
110 POKE S2,N
130 FOR C=0 TO D  :NEXT C
150 POKES2,0
170 FOR P=0TO 20:NEXT
180 GOTO 80
200 POKEV,0
210 END
1000 DATA 225,2,225,2,225,2,223,2,223,2,223,2,219,2,219,2,219,2,215,4,215,2
1010 DATA 209,2,209,2,209,2,207,2,207,2,207,2,201,2,201,2,201,2,195,6
1020 DATA 231,2,231,2,231,2,228,2,228,2,228,2,225,2,225,2,225,2,223,4,223,2
1030 DATA 219,2,219,2,219,2,215,2,215,2,215,2,209,2,209,2,209,2,207,6
1040 DATA 195,2,195,1,201,1,207,1,209,1,215,2,215,2,215,2,219,2,219,2,219,2,207,
4,207,2
1050 DATA 209,2,209,1,215,1,219,1,223,1,225,2,195,2,195,2,209,2,201,2,215,2,195,
6,-1,-1

READY.
```

LISTING 2

```
READY.

5 DIM N(66,1)
10 V=36878:S1=V-4:S2=V-3:S3=V-2
20 INPUT"TEMPO";T
30 POKEV,15
40 FORI=0TO66:FORJ=0TO1:READ N(I,J):NEXTJ,I
50 INPUT"NO OF TIMES";NR
60 FOR R= 1 TO NR
80 FOR I=0 TO 66
90 POKE S2,N(I,0):D=N(I,1)
100 FOR C = 0 TO T*D:NEXT
110 POKES2,0
120 FORP=0 TO 20:NEXT
130 NEXT I
140 NEXTR
150 POKEV,0
160 END
1000 DATA 225,2,225,2,225,2,223,2,223,2,223,2,219,2,219,2,219,2,215,4,215,2
1010 DATA 209,2,209,2,209,2,207,2,207,2,207,2,201,2,201,2,201,2,195,6
1020 DATA 231,2,231,2,231,2,228,2,228,2,228,2,225,2,225,2,225,2,223,4,223,2
1030 DATA 219,2,219,2,219,2,215,2,215,2,215,2,209,2,209,2,209,2,207,6
1040 DATA 195,2,195,1,201,1,207,1,209,1,215,2,215,2,215,2,219,2,219,2,219,2,207,
4,207,2
1050 DATA 209,2,209,1,215,1,219,1,223,1,225,2,195,2,195,2,209,2,201,2,215,2,195,
6,-1,-1

READY.
```

before. Now let's try it backwards, change line 80 to read
80 FOR I = 66 to 0 STEP -1

Using array storage gives us greater flexibility. Now let's try a short bit of harmony. This involves having all three voices playing different notes at the same time. Our first example will be a simple one where the voices hold each note for the same length of time. For a tune I'll use the first part of the three-part round used later on. We'll use only two voices. Enter listing 3. You will notice that it's identical in form to the early example except that we read in two notes not one. Run it and see what it sounds like. Again you could use array storage to give greater flexibility. Try it, I suggest a 3-D array would be best. Storing the data like this N(1st note, 2nd note, duration).

Most harmony does not have each voice holding each note for the same duration, so let's look at how you can program for this.

I've used a three-part round as an example of this. Voice 1 plays line 1 (1st 21 notes) while Voice 2 plays line 2 (notes 22-42) and Voice 3 plays line 3 (notes 43 to end). Then voice 1 goes to line 2, voice 2 to line 3 and voice 3 to line 1, and so on.

Enter listing 4 and run it. A quick explanation of the trickier parts of the program: Lines 100 to1 20 turn on each voice to the right note and store how long it will be held. Line 130 holds the note for 1 interval. Lines 200 to 240 keep a track of how long voice 1 has held a note. If it's time for a new note they set P1 to the right position in the music and turn voice 1 off. Similarly 300 to 340 and 400 to 440 for voices 2 and 3.

Try your own modification to this program. Can you get voice 1 to play line 1 then voice 1 to play line 2 while voice 2 plays line 1 then voice 3 to play line 1, voice 2 line 2 and voice 1 line 3 and then repeat. Like all things experimentation is the only way to learn. May your VICs be alive with the sound of music.

Any questions, suggestions, problems you can contact me at 32 Eastburn St, Papakura, or C/- "Bits & Bytes". If you want a personal reply, please send a stamped addressed envelope.

LISTING 3

```
.READY.

10 V=36878:S1=V-4:S2=V-3:S3=V-2
20 INPUT"TEMPO";T
30 POKEV,15
40 INPUT"NO OF TIMES";NR
50 FOR R=1 TO NR
60 READ N1,N2,D
70 IF N1 =-1 THEN 160
80 POKE S1,N1
90 POKE S2,N2
100 FOR C= 0 TO D*T:NEXT
110 POKES1,0
120 POKES2,0
130 FOR P=0 TO20 :NEXT
140 GOTO 60
150 RESTORE:NEXTR
160 POKE V,0:END
500 DATA225,231,2,225,231,2,225,231,2,223,228,2,223,228,2,223,228,2
510 DATA 219,225,2,219,225,2,219,225,2,215,223,4,215,223,2
520 DATA 209,219,2,209,219,2,209,219,2,207,215,2,207,215,2,207,215,2
530 DATA 201,209,2,201,209,2,201,209,2,195,207,6,-1,0,0

READY.
```

LISTING 4

```
READY.

10 PRINT" A ROUND IN THREE PARTS"
20 DIM N(66,1)
30 FOR I = 0 TO 66
40 FOR J = 0 TO 1:READ N(I,J):NEXTJ,I
50 V=36878:S1=V-4:S2=V-3:S3=V-2
60 POKE V,15:P1=0:P2=21:P3=42
70 INPUT"ENTER TEMPO";T
80 INPUT"NO OF TIMES";NR
90 FOR R=1 TO NR
95 FOR L =0 TO 143
100 POKE S1,N(P1,0):D1 = N(P1,1)
110 POKE S2,N(P2,0):D2 = N(P2,1)
120 POKE S3,N(P3,0):D3 = N(P3,1)
130 FOR C=0 TO T:NEXT
200 T1=T1+1
210 IF T1 < D1 THEN 300
220 P1 = P1 +1
230 IF P1> 66 THEN P1=0
240 POKE S1,0:T1 = 0
300 T2=T2+1
310 IF T2 < D2 THEN 400
320 P2 = P2 +1
330 IF P2> 66 THEN P2=0
340 POKE S2,0:T2 = 0
400 T3=T3+1
410 IF T3 < D3 THEN 500
420 P3 = P3 +1
430 IF P3> 66 THEN P3=0
440 POKE S3,0:T3 = 0
500 NEXT
510 NEXT
520 POKEV,0:POKES1,0:POKES2,0:POKES3,0
550 END
1000 DATA 225,2,225,2,225,2,223,2,223,2,223,2,219,2,219,2,219,2,215,4,215,2
1010 DATA 209,2,209,2,209,2,207,2,207,2,207,2,201,2,201,2,201,2,195,6
1020 DATA 231,2,231,2,231,2,228,2,228,2,228,2,225,2,225,2,225,2,223,4,223,2
1030 DATA 219,2,219,2,219,2,215,2,215,2,215,2,209,2,209,2,209,2,207,6
1040 DATA 195,2,195,1,201,1,207,1,209,1,215,2,215,2,215,2,219,2,219,2,219,2,207,
4,207,2
1050 DATA 209,2,209,1,215,1,219,1,223,1,225,2,195,2,195,2,209,2,201,2,215,2,195,
6,-1,-1

READY.
```

PRINT — and what it means

Basic BASIC No. 4

By GORDON FINDLAY

Continuing a series on BASIC for complete beginners.

This month I want to look carefully at the PRINT statement and its variations, and at formatting output.

Those of use who were around in the bad old days - before personal computers - learned programming in FORTRAN, ALGOL, COBOL, or some other equally bizarre language. In these languages we had to learn how to produce printed output by giving detailed instructions about where on the page each piece of information was to be printed, how many figures to print, and where to put the decimal point. This was called FORMATTING, and a horrible job it was, too! Formatting is still the major concern of commercial programming. With the advent of BASIC people have tended to stay clear of output formatting, relying on the computer to decide what to print and where to put it.

This is understandable, but in my view shortsighted. Without thoughtful presentation of output programs are only useful to their authors, and only while the author can remember what the various bits of information mean. But it is easy to get presentable output in BASIC!

What do I mean by presentable output? First, the screen (or print out - but I'll refer just to the screen) should not be overcrowded. Just because a screen will display 25 lines of 80 characters each is no reason to use them all. Usually information is easier to assimilate in small bits. In fact, in many cases just a few lines should be displayed, then the screen cleared.

Very often you can avoid having the screen scroll - roll up from the bottom. Very often too, you can let the user set the rate of information displayed to suit himself.

Machine dependence

Machines vary considerably in their PRINT statements. There is no consensus about the number of lines on the screen, or the number of characters per line. Every machine has a command to clear the screen. The ones that I know of are these:

CLS - for TRS80 and similar
HOME - for the Apple
PRINT CHR$(147) - for the PET

RIGHT NOW go and look up what it is for your machine if it isn't listed here. In the examples I will use CLS - watch out for it!

If we want to print a number of items at one time, say a few numbers, some text, some strings and so on, we can just list them in a PRINT statement. There are, generally, two possible punctuation marks we can use to separate things in the list - commas, and semicolons. The usual thing is that if commas are used the things in the PRINT statement are printed spaced out, while semicolons are used to print them closer together. Here is a program to run to demonstrate, and to let you see exactly what happens in the case of your machine.

```
10 REM DEMONSTRATION OF PRINT
   SEPARATORS
20 REM
30 A = 1
40 B = 3
50 C = -5
60 D = 3.1415926
70 X$ = "BITS"
80 Y$ = "AND"
90 Z$ = "BYTES"
100REM CHANGE THE NEXT STATEMENT
   TO SUIT YOUR MACHINE:
110CLS
120PRINT "1.NUMBERS, WITH
   COMMAS:"
130PRINT
140PRINT A,B,C,D
150PRINT
160PRINT "2. NUMBERS, WITH
   SEMICOLONS:"
170PRINT
180PRINT A;B;C;D
190PRINT
200PRINT "3. STRINGS, WITH COMMAS:"
210PRINT
220PRINT X$, Y$, Z$
230PRINT
240PRINT "4. STRINGS, WITH
   SEMICOLONS:"
250PRINT
260PRINT X$;Y$;Z$
270END
```

These are the things to look for:

1. How many numbers are printed on a line when commasare used?
2. How many spaces are between numbers when semicolons are used?
3. Are strings spaced at all when semicolons are used?

The lines with just PRINT produce blank lines - a very important thing to be able to do! Blank lines make output readable.

Most likely you will find that no spaces are inserted at all when strings are printed with semicolons. To get spaces, you must print them yourself.

```
10 PRINT X$;" ";Y$;" ";Z$
```

The space in each set of quotes is printed.

TABs

A very useful facility is to be able to say exactly whereabouts a number or string is to be printed. This is done by using a TAB - just like a typewriter tabulator. To print in, say column 12, include TAB(12) in the print statement just before the number or string. Semicolons must be used to separate items if TAB is used. So to print a number in

In the belly of the beast III

Every computer has at its heart a Central Processing Unit (CPU), which controls the activities of every other part of the machine. The other parts of the computer are called simply peripherals because they 'hang off' the CPU. We could easily devote the rest of this article to the variety of peripherals available, but instead we will just look at a few.

Two important classes of peripherals are input and output devices, such as keyboards and printers. Another is storage devices and once again there is a vast range. The most obvious for the home-computer user is the cassette tape or the disk drive. In one sense both of these are also input/output devices and this confusion just goes to show why we use the general word, peripherals.

Incidently, when we talk about "on-line" and "off-line" peripheral devices we are talking about the computer's ability to

BY GERRIT BAHLMAN

use peripheral devices without humans having to be involved. With a cassette tape you have to either put a new tape in and switch on the tape recorder or you might have to rewind or some such thing before the computer can use the tape for storage or input of information. But, if you have a disk drive with the right disk, the computer can control it without any help from you. A disk drive is said to be "on-line" and tape drive is "off-line".

It is clear from this brief discussion of two peripheral secondary storage devices (the tape and the disk drives) that the computer has the ability to control. The controller is inside the central processing unit with two other components, the arithmetic unit and primary storage or memory.

It is important to point out that the physical layout can vary from machine to machine. Primary storage is often separated into

Continued on next page

From previous page

column 8 and its square in column 20, we can use the statement:

```
PRINT TAB (8);X;TAB(20);X*X
```

Put it in a loop, and you get a nicely set out table:

```
10 REM TABULATION DEMONSTRATION
20 REM   CHANGE   NEXT   LINE   IF
   NECESSARY:
30 CLS
40 X = 1
50 PRINT TAB (8);X;TAB(20);X*X
60 X = X + 1
70 IF X <= 10 THEN GOTO 50
80 END
```

Finishing lines

One other use for punctuation marks. Suppose that you wish to print again on the same line. This can be accomplished by leaving a trailing semicolon on the first lot of printing. It's easier to demonstrate than to describe:

```
10 X$ = "FIRST"
20 Y$ = "SECOND"
30 Z$ = "THIRD"
40 PRINT X$;" ";
50 PRINT Y$
60 PRINT Z$
70 END
```

The semicolon on the end of line 40 means that when line 50 is done, the string Y$ is printed on the same line as X$. The output is:

```
FIRST SECOND
THIRD
```

Other variations

There are a lot of other variations of the PRINT statement, but they depend on the particular machine you are using. Get to know all the variations you have available. TRS-80 users have PRINT@, PRINT USING, and combinations of them. Apple users have SPC(), HTAB, VTAB, INVERSE, FLASH. PRINT USING is also available for the Apple, but in the form of a software routine, which is less convenient. Of all the variations, PRINT USING is the most flexible. Unfortunately, it is not easy to follow in most microcomputer implementations - well worth the effort though.

The use of colour, varying sizes and styles of lettering, inverse video and so on are too machine-specific for us here.

Making the display pause

When you are presenting a lot of information, you need to let the person reading it have some control over the speed of presentation. One easy way is to INPUT something when you want a pause:

```
INPUT ZZ$
```

This forces the user to type something before the program carries on. Here is an example - it prints ten lines, pauses, and then another ten lines, clearing the screen first.

```
10   CLS
20   X = 1
30   PRINT X,X*X
40   X = X + 1
50   IF X <= 10 THEN GOTO 30
60   INPUT ZZ$
70   CLS
80   PRINT X,X*X
90   X = X + 1
100  IF X <= 20 THEN GOTO 80
110  END
```

Notice that we input a variable - in fact a string, called ZZ$ - which we don't do anything with! It is a dummy variable, just forcing the user to do something before the program starts throwing information at him again. It is usual to use a string because that enables the user to type anything (almost) rather than having to make sure he types a number.

Many machines allow you to type nothing at all - just pressing the 'NEWLINE', 'ENTER', or 'RETURN' key (or whatever your micro calls it). If so, a very nice form is this:

```
INPUT "PRESS 'RETURN' TO
CONTINUE"; ZZ$
```

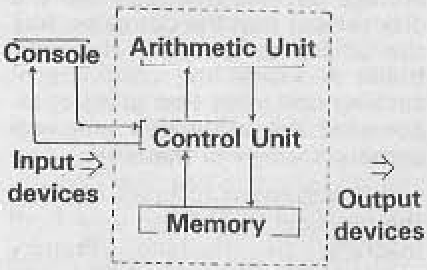
where once again the combination of a message and an input is used.

The user of your program can only communicate with it through the PRINT and INPUT statements that you provide. Even if the user is only ever going to be you, take the trouble to make this communication as easy, and as pleasant, as possible.

In The Belly of the Beast III

From previous page

two parts. Registers that hold one or two words of memory and the main memory which may be quite separate from the control unit. Nevertheless, we will describe the whole shebang as the CPU. Figure 1 will help to explain what is meant.



Central Processing Unit

Figure 1

In the good old days when computers could just barely do sums and used up enough electricity to light a small town the main memory of the computer was called the core store. You might like to know why it was called this.

Originally, small cores of ferromagnetic material were laced together on a grid. The little core magnets (that's what they were) were of two types.

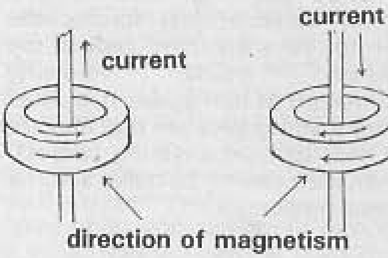


Figure 2

When an electric current is passed along a wire, a magnetic field is formed around the wire. This can be easily demonstrated with a wire carrying a current through a sheet of paper or cardboard on which iron filings have been sprinkled. The filings arrange themselves in concentric rings around the wire. If we pass a current along a wire on to which an iron ring has been threaded, the ring becomes magnetised in one direction, and if we reverse the current the direction of magnetisation also reverses, but in both cases when the current is switched off the ring retains some magnetisation and will attract small pieces of metal such as iron filings.

So, we have a little magnet in the shape of a ring which can be changed from one state to another given the right pulse of electricity. Two states, that sounds like binary. And, that sounds like a bit memory. So, vast numbers of these little cores were wired together. Each core had three wires passing through it. One was used to sense which way round the magnetism was and two were used to switch the state of the core. Figure 3 shows what it looked like.

To switch one particular core and only that core, half the current

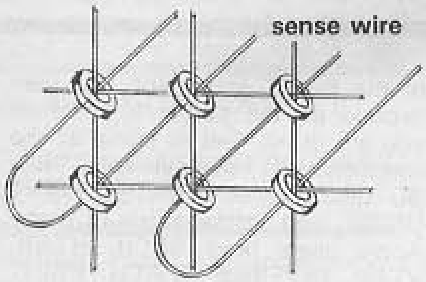


Figure 3

needed to do the switching was sent down each of the two wires that passed through that core. The other cores would only ever have half the current needed for a switch over of their magnetism and so they remained as they were. Setting the cores to particular states was the equivalent of writing into memory. The important point to note is that each core has a special place in the lattice of cores. You can find a core by numbering the wires like a graph frame and giving the coordinates of the particular core. Each core has a particular address. This idea of an address is important because we will use the same idea in describing machine-code programming.

To read from this memory a half pulse is sent down the wires in the core and if it flips then it generates a wee pulse in the sensing wire. If you were trying to write an "O" when it flipped it must have been an "I" otherwise it would not have flipped! Of course, if it flipped you would have to flip it back but you would know what it has "stored" in the first place.

You might think that this all sounds complicated but compared to some of the more recent developments it was very simple and bulky. At any rate, you now know why primary memory in computers is sometimes still called "core store". It is an old fashioned term and I doubt that you will ever use a computer that actually has "core store".

Now, let's get back to the CPU. The arithmetic unit does all the addition, subtraction, and division of quantities and is like a calculator. The control unit contains the necessary circuits to perform the logical-switching jobs needed to run through a program.

The computer processes information which can be obtained from memory and the results are returned to memory. There is usually a minimum amount of information that can be processed in any one operation and moved to or from memory. This is called a "word". A word may represent a number to be used in numerical calculations or one or more characters of non-numeric information.

A word consisting of a number of bits, may be used to represent anything that the user cares to represent. Each bit of an N-bit word can be an "O" or an "I" so that a word can have 2^N different states.

If a computer word consists of N bits then it is drawn as in Figure 4.

Home computers are typically 8-bit machines

The bits are numbered from left to right. Bit O is often referred to as the "most significant bit" and bit N-1 is referred to as "the least significant bit".

bit 0	bit 1	bit 2	bit 3	bit N-1

N bit computer word

Figure 4

FARMERS PLUG IN TO COMPUTING

By Dr PETER NUTHALL

Farmers around the world are starting to discover the benefits of the electronic revolution not only in the form of microcomputers but also in the form of large computer based information and marketing systems. The research effort that has produced the microcomputer has also meant the cost of large machines has markedly decreased. Similarly, methods of communicating with these machines from afar have improved to the point where it is possible for farmers to have telephone based direct connections to mainframe computers. The net effect is a whole new range of farming aids have become a possibility at a cost that may well make them an economically sound proposition.

In North America the power of the large computer is being harnessed in a number of experimental remote marketing systems. A large proportion of the cotton production in Texas changes hands through a computer based market. The direct comparison in New Zealand is the wool market. Because cotton can be described without the need for visual inspection (fibre length, diameter, strength...), details of lots on offer can be held on a central computer and accessed by potential buyers through terminals without the need for bringing buyers and cotton to central selling points.

Not so easily described are store cattle. However, this hasn't deterred a group at a Texas University from setting up a computer based market for these animals. Calves are reared at one end of the state but the feedlotters tend to be situated at the other end. Using traditional marketing methods involves considerable time and travelling including the movement of stock to yards for unloading and reloading. The computer based system does away with a lot of this expense through having lots for sale listed on the computer so that buyers only have to travel to local terminals hooked into the central computer. The organiser also maintains that the prices received have been greater than conventional market prices due to increased competition.

When a farmer wants to sell a mob he calls in a third party grader who visits the farm and inspects the cattle. These graders attend grading schools to ensure uniformity and use standar classifications for scoring the animals. Numbers, weights, and breed are also recorded. Details are then entered into the computer, again using a local terminal connected to the central machine, and at the appointed hour each day bidding starts. Prior to this potential buyers can turn on their terminal and list out details of all the lots on offer. Once bidding starts each lot is displayed on the terminal screen for a fixed time. The buyers key in their offer which appears on their screen and, if it is the highest bid so far, it also appears on all other terminals hooked into the central computer. If your bid is not the highest you can increase your offer at any time. At the end of the alloted period the lot is knocked down to the highest bidder provided the reserve has been reached.

Meanwhile, the cattle have not left the farm. If a sale is not made nothing is lost. If a sale is made the cattle change hands on an agreed date at a local weighing station. The contract price is based on this weighing together with a re-count of the numbers involved. Funds flow through a central clearing bank and the system automatically prints sale notices and other paper work including the incorporation of any penalties for not supplying what was originally promised.

Besides the auction system cattle can also be offered on contract. Farmers set the price and have their mob listed on the computer for a given number of days. If a sale is not made within the given period the animals can be withdrawn or re-offered, perhaps at a different price.

One of the advantages of the system is that statistics on the cattle market can be readily obtained. As sales are made the statistics are updated and at any time buyers or sellers can call up on their terminal information on the numbers sold over the recent past and the prices being paid. The result is better informed marketing for both small and large traders.

The Texas scheme has only been in operation for two years so it is too early to tell whether most cattle will eventually be sold through the scheme. Certainly at this stage the numbers are growing as people gain more confidence in the system. Time will also tell whether the not inconsiderable cost of a computer based scheme will in fact be less than traditional marketing costs.

Cattle are not the only animals being traded through electronic means. Lamb and pigs are also coming under the electronic hammer in North America. Not all of these systems rely on total computerisation as this is not always appropriate in the initial stages. One lamb market in Oklahoma, for example, relies on using video shots of the animals shown at a central location to which the buyers come. Closer to home an Australian cattle selling experiment relies on using telex communication for bringing the buyers together.

Marketing is one way of using contemporary computer power. Another is the provision of "on the farm" general farming information. In Britain, for example, owners of a slightly modified TV set can get up on their screen several pages of constantly changing agricultural information without the need for a direct connection. In Canada, using telephone connections, farmers with an appropriate terminal can extract a wide range of constantly updated data whenever the need

Introducing the first of our monthly columns on microcomputers and farming, compiled by Chris McLeod, an agricultural computer consultant.

Contributions and inquiries are welcomed and should be directed to Chris McLeod, R.D. 5, Ashburton or c/o BITS & BYTES, Box 827, Christchurch.

arises. An assessment of just how valuable all these schemes are going to be must again wait for more years of experience.

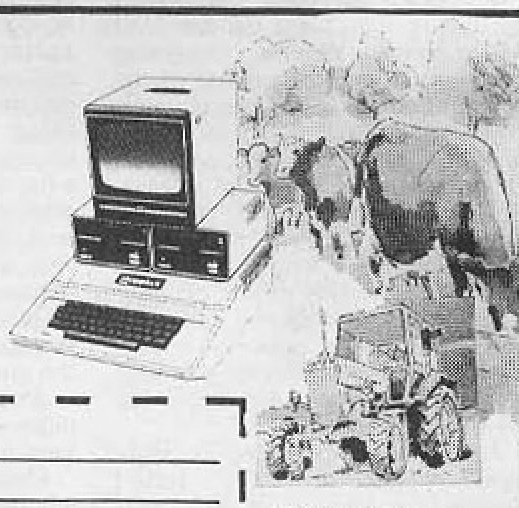
In Kentucky, a trial system was set up some two years ago with the intention of exploring the feasibility of an electronic information system for use by farmers in their farm kitchen. The researchers decided the system could be made 98% reliable and could be of considerable benefit to farmers provided the right information was put up on the system and was constantly updated. If it wasn't, it was a waste of the benefits of the electronic system. If the information doesn't change it is probably better suited to the printed page.

Two hundred farmers in the Kentucky trial were given small black boxes with a numeric key pad. These were hooked on to their TV set to act as a display screen. Inside the black box was a small microcomputer capable of storing several pages of information. When the farmer wanted to access information on the central university computer he dialled the computer's number using his telephone. When the beeping was heard he flicked a switch on his telephone which connected his telephone line directly to the black box. At this stage messages, in colour, appeared on his TV so he could then key in the numbers of the pages of information required. The "black box" then automatically transmitted these numbers back to the central computer which responded by sending back down the telephone line the screenful of information requested. Once the 'black box' had received and stored all this information a message on the TV told the farmer to disconnect his telephone (thus minimising any toll charge). With the information stored in his black box the farmer could then, through pushing the "next page" button, review the information stored, page by page, on his TV. An hour later, for example, he could repeat the whole procedure to get an update on, perhaps, local weather or, say, the local store stock aution prices.

The 200 trial farmers used the system mainly for the market and weather information stored on the system. General farming information was seldom accessed though this may have been due in part to the lack of adequate updating systems.

None of these systems has yet reached New Zealand though there is no technical reason why they shouldn't be started. However, one of the advantages of not being first is that we can learn from other people's mistakes. But, the time must be fast approaching where experimental systems should be set up. Lincoln College is starting the ball rolling through research on information systems. Logically this should be associated with a marketing system to make better use of available resources. Maybe it is time for funds to be set aside for this work. Such work should be linked to microcomputer investigations as they can also be used for terminals and possibly as recipients for downloaded programs. The potential benefits from carefully designed and co-ordinated systems are too great for us to put off starting research and development for another year.

Dr Nuthall is head of the Kellogg Farm Management unit at Lincoln College which is developing farm software.



A format of many uses

Many uses can be found for a large format on the ZX81 for example display - advertising or result read-out. Their implementation in ZX81 BASIC is painfully slow. This program in machine code is extremely fast.

The machine code is listed separately as many machine code loading programs have been published.

The BASIC lines may be modified to give a mix of large or small characters.

Owners of greater than 3¼K RAM need not call 16514 as this simply allots a full screen.

This program by Dominic Toon. recently won the Auckland Sinclair Users Group programming competition.

```
ZX81

THIS PROGRAM PRINTS THIS SIZE
LETTERS IN THE BLINK OF N EYE
ON A 1K ZX81
-HIT "ENTER" FOR A SPACE

BIG CHR$
1982
BY
DOMINIC
TOON

SCREEN FULL

     REM LN 77 -  NOT ! CLEAR Y
?UNOT $4 NEXT 7?USRND?-   ACS 7
ACS )( IF $ 2;;#7777 FAST   ACS
 63ACS 63ACS 73ACS 73E:RND? GOSU
B 7 NEW #775:RND( THEN )) ;6:RND
 LPRINT $4ATN TAN
  10 RAND USR 16514
  50 PRINT AT 21,0;"TYPE LETTER"
  60 FOR N=0 TO 16 STEP 4
  70 FOR I=0 TO 28 STEP 4
  80 PRINT AT N,I;"$"
  90 IF INKEY$<>"" THEN GOTO 90
 100 IF INKEY$="" THEN GOTO 100
 110 LET A=CODE INKEY$
 120 IF A>63 THEN LET A=0
 130 POKE 16417,A
 140 PRINT AT N,I;
 150 RAND USR 16535
 160 NEXT I
  43 NEXT N
 180 PRINT AT 21,0;"SCREEN FULL"

 MACHINE CODE-

16514 CD 23 0F 0E 16 06 20 AF D7
16523 10 FD 0E 75 3C D7 0D 20 F3
16532 C3 2B 0F 3A 21 40 5F 16 00
16541 06 03 CB 23 CB 12 10 FA 21
16550 00 1E 19 0E 04 56 23 5E 23
16559 E5 06 04 AF CB 23 1F CB 22
16568 1F CB 23 1F CB 23 1F 2A 0E
16577 40 77 ED 67 7E FE 06 38 03
16586 2F E6 87 77 23 22 0E 40 10
16595 DE 11 10 00 19 22 0E 40 E1
16604 0D 20 CC C9
```

Hex code lists for the ZX81

By P.R. WALKER

When working in machine code or assembler, hexadecimal (base 16 i.e. 0, 1 . . . D, E, F) makes a lot more sense than ordinary base 10 decimal. I am chewing into Ian Logan's "Understanding Your ZX81 ROM" and found I had no convenient way of looking at the RAM and ROM.

This little utility takes a start address (decimal, I work this out on a calculator) and prints 16 lines of 8 bytes plus a few extra lines for overlap. It's best to keep to neat fractions of a K, 16 x 8 is 80 in hex and twice that is 100.

For some reason this goes more quickly than the original version printing in decimal. Can anyone explain?

```
  5 REM DECIMAL BYTES TO HEX
  6 REM
  7 SLOW
 10 PRINT "ENTER START ADDRESS,
      DECIMAL"
 15 INPUT START
 20 CLS
 30 FOR A = 0 to 20
 40 FOR B = 0 TO 7
 50 LET N = PEEK (START + 8 * A + B)
 60 LET H = INT (N/16)
 70 LET L = N - 16 * H
 80 PRINT TAB 4 * B; CHR$ (H + 28);
      CHRS$ (L + 28);
 90 NEXT B
100 NEXT A
```

Fixing a ZX81 printer problem

By BRUCE STEVENSON

A common problem with the ZX-81 printer is that it has a tendency to print lines of characters which are squat (compressed in the vertical direction). This fault is caused by paper slippage. The rubber drive roller, visible at the back of the printer with paper carrier removed, is not gripping the paper firmly enough to draw it through from the paper carrier. In bad cases, the paper may not feed at all, even when you push the paper feed button.

Here's how to fix it

First, increase the spring tension on the metal roller in the paper carrier as follows: Remove the paper carrier from the printer, then the roll of paper from the carrier. You will see a small spring at each side of the carrier. Remove one very carefully, noting its orientation. Now unwind the spring slightly, no more than 5 degrees. This will have the effect of increasing the spring tension on the metal roller.

Re-insert the spring, taking care that it doesn't spring away from you. If you've unwound it too much, it won't sit properly and will jump out on you. Repeat for the spring at the other side of the paper carrier.

At this point, it's worth putting just a little drop of light oil at the points where the springs bear against the metal roller; also where the paper roll hubs run in the carrier.

Now turn your attention to the rubber drive roller at the rear of the main body of the printer. This should be clean: keep your greasy mitts off it! Clean it with a cotton bud dipped in warm soapy water, and rinse thoroughly with clean water. To get at all sides of the roller you will need to apply power to the printer and press the paper feed button. I have used steel wool to roughen the surface of the roller slightly; anything that increases friction is worth while.

This done, and the roller quite dry, reassemble and marvel at the new-found height of your characters.

Note that there is a disadvantage to this procedure — you'll use more paper!

Southland software

A number of farm programs for use with Visicalc and the Apple III microcomputer have been developed at Dunrobin Station in Southland.

The programs or templates use the calculation and spread sheet capabilities of Visicalc III as a base and have been written by Mr Bernard Pinney.

He has written templates for numerous purposes such as:

- Wages calculations, reconciliations and bonus systems.
- Paddock lists which are integrated with fertilizer calculations and seed requirements.
- Land development costings
- Sheep gross margins
- Cash flow analysis
- Deer records including stock on hand, reconciliations, gross margins, velvetting analysis and indexing in order of merit.

Templates for many of these functions are available on a 13cm (5¼ inch) disk for $200.

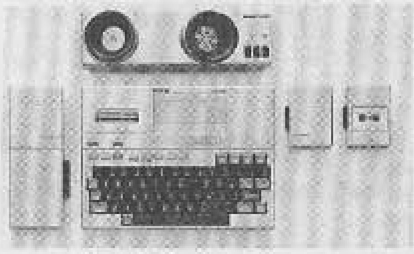
Mr Pinney says the cost of the Apple III, one disk drive and the Visicalc (about $8000) has been repaid several times in the 10 months the system has been operating at Dunrobin one of New Zealand's largest farm operations.

He cites as one example wages, which previously took half a day to prepare, now take half an hour.

Mr Pinney, who is also president of the New Zealand Deerfarmer's Association, feels the templates will be particularly useful to diversified farms but will also aid accountants and stock firm managers anxious to try out "What if" possibilities for future farm programmes.

Enquiries to Bernard Pinney, Dunrobin Station, RD 2 Lumsden, telephone 46R Mossburn.

HX-20 "A TRULY PORTABLE MICRO"



By SELWYN ARROW

Can you envisage someday having a battery powered, go-anywhere, microcomputer complete with printer and microcassette that will easily fit inside a briefcase, and still leave room for all those important papers as well as your lunch? No, I am not looking into a crystal ball, I am looking at the latest item of microcomputer miniaturisation from Japan.

When Epson say portable they really mean it! The complete device is exactly the size of an A4 sheet of paper and stands only 45mm high. If you can imagine a stack of 18 BITS & BYTES magazines, that would be the size of it.

Unfortunately it costs a lot more than 18 BITS & BYTES. But don't let the price of this machine put you off, it has a lot more going for it than microcomputers three times its size.

At first glance the inbuilt microprinter catches the eye.

Another special feature is its liquid crystal display. This is a 20 character by 4 line window on a large 'virtual' screen that can have up to 255 characters per line or up to 255 lines (limited only by the available memory). The display also includes a graphic screen which can be displayed independently or combined with the text screen.

Adjustable Viewing

Have you ever found the problem with LCD diplays, where you have to move the display or yourself to get a good contrast? Well no more. Epson, who produced the first LCD calculator display panels, have now perfected an electronically adjustable viewing angle which is varied at the touch of a knob.

The keyboard is full typewriter size, and includes a numeric keypad mode selected by the num key. The keys have a professional action with auto repeat on most, and no key bounce at all. Many keys are multi function in conjunction with the shift, control and graphic keys. The graphic key provides 32 graphic characters (see printout) plus the 5 programmable function keys also provide many control and shorthand functions for use in basic and to control the inbuilt peripherals.

When the machine is first turned on it displays a menu, the first item of which enables you to initialise the internal parameters including the date and time which are then available on request. This is only necessary when first purchased or if the batteries have been removed for some reason. I would not recommend doing this unless necessary as it also cold

Continued on page 38

Microcomputer Summary

Name:	HX-20 Portable Computer.
Manufacturer:	Epson Corporation, Japan.
Microprocessors:	CMOS 8 Bit 6301 for both main and slave.
Clock Speed:	614kHz (both).
RAM	16K Standard. Battery backed-up.
ROM:	32K Standard. Expandable internally to 40K.
Input/Output:	Audio cassette interface, RS232C — 8 pin din connector; 110-4800bps (bits per second), Serial — 5 pin din. 38.4Kbps, Bar Code Reader — special connector.
Keyboard:	Full size 56 key typewriter style, includes a ten keypad plus 12 extra keys.
Display:	Liquid crystal screen, 20 characters by 4 lines.
Languages:	Epson Extended Microsoft BASIC.
Graphics:	LCD 120 x 32 dot matrix, dot addressable.
Sound:	Programmable square wave. Four octave with half tones, using built in piezo-electric speaker.
Cost:	$1600.
Options:	Expansion unit 8K RAM plus 24K ROM or 16K RAM plus 16K ROM, fits to the left of the keyboard. Microcassette — uses standard microcassette tapes. Plugs into the right of the display. ROM cartridge — uses 8, 16 or 32K ROM. Plugs into the right of the display instead of the microcassette.
Peripherals	Standard audio cassette, display controller — this serial interface provides 32 characters by 16 lines in 4 colours on a monitor or TV set. Monochrome graphics are 128 x 96 dots, 4 colour graphics are 128 x 64 dots, floppy disks — TF-20 drive holds two 13cm (5¼ inch) disks for 640KB capacity. Acoustic coupler. Full size printer, bar code reader.
Special Features:	Built in microprinter — 24 column, 5 x 7 dot matrix impact type, 42 lines per minute; rechargeable power supply — 40 to 100 hours between charges, automatic 'low power' display. Eight hour recharge.
Review Unit From:	Microprocessor Developments Ltd, 24 Manukau Rd, Epsom, Auckland.

The art of machine code programming

By GORDON FINDLAY

Since the last column that I wrote dealt almost exclusively with BASIC, I thought that this month I should deal with machine code programing, specifically related to the '80. I shall take up from where Gerry Bahlman leaves off, and deal with some of the products available for the '80.

To start in machine code programming you need at least three pieces of software. Although I have heard of hardy souls who can do without an assembler, I wouldn't recommend it!

As Gerry has explained, machine code is written in the form of mnemonic instructions, which are then 'translated' into hexadecimal, or binary, form. In our case, the mnemonics are in Z-80 assembly code.

The mnemonics must be typed in, and stored somewhere, and no doubt altered (edited). This requires an editor, which may be tape or disk (or both) based.

Then the mnemonics must be translated - by an assembler. This produces an actual machine-code program - again on disk or tape.

This program may have errors in it! You need to be able to debug it, and for this you need a monitor.

Sometimes you may have a machine code program in object form (i.e. in hex), and want to go back to source, or mnemonic form - for this you need a disassembler.

Let's look at some of the available software. This isn't a full review - that would require a lot more space - just an overview of material which is readily available.

The editor you use will depend on the assembler you use. Often these two come together in one package. The first of these to be produced was Tandy's Editor-assembler: Edtasm.

This is a simple, but reasonably powerful piece of software. It is on tape, and only outputs to tape. The editor is reasonably good, being similar to the BASIC level II editor, with simple search functions.

Microsoft released an upgrade, called EDTASM-PLUS. This gives a more powerful editor, with many more powerful search and replace functions, and a much more powerful assembler.

This assembler allows you to use macros - redefined sequences of instructions - and to assemble into memory directly, instead of first on to tape, and having to load the object code from tape before running it. This saves time in a tape system - especially when you remember that you will probably have to reload the EDTASM-PLUS (three minutes) and the source code!

EDTASM-PLUS is now available for disk systems as well, but I haven't had a chance to try it.

For disk systems there are a lot of alternatives. Those using NEWDOS80 will have APPARAT's modifications of the original Tandy EDTASM. This is adequate, but there are better alternatives.

At the other end of the power spectrum there is MACRO-80, which is another Microsoft product, sold as the Radio Shack Disk Assembler, or as the Microsoft ALDS (Assembly Language Development System).

This is a very powerful assembler indeed! It allows you to have macros - even nested macros - conditional assembly, with some very powerful pseudo-ops.

It is a little difficult to use at first because of its power, and its relatively obscure documentation.

Using MACRO-80, you can write a program in several distinct blocks, assemble them quite separately, and when finished tie them all together with a linking-loader which is also supplied in the package, called (you guessed it!) LINK-80.

This makes programming much easier once you master the linking processes, and makes it feasible to keep a lot of routines in a library. They can then be used almost instantly (you do need to include some declarations) in other programs.

Of course there is a trade-off: compared with the other assemblers, which do not need link-loading, the process seems a little slower - but don't forget, it was easier to develop the program in the first place!

The source code for MACRO-80 can be written using almost any editor or word processor.

The editor which comes with ALDS is EDIT-80. It is a powerful editor but a clumsy one. It is the only editor I have come across which allows you to use files which are on disk rather than in RAM - this means that much bigger files can be manipulated. However, this leads to other complications, and most people use this editor only when they have to.

Fortunately, MACRO-80 doesn't insist on line numbers, so any word processor can be used - I routinely use SCRIPSIT or ELECTRIC PENCIL, no doubt others would do as well.

Between MACRO-80 and Tandy's assembler are a host of others, each with their good and bad points. The choice is restricted in this country - if you especially want something different, you may need to get it from the States.

Disassemblers also come in a wide range. The standard is set by Apparat's disassembler, which comes on the NEWDOS disk. This is a fast program, with a wide range of options - such as storing and/or printing the disassembled instructions, building cross-reference tables, etc.

SYSTEM 80/TRS 80

For tape operation, there are two marketed by Instant Software - TLDIS, and THE DISASSEMBLER. Of the two TLDIS is much the superior - it automatically assigns labels to instructions as needed, and replaces absolute addresses with these labels. This is a great help if you wish to modify the final code.

Unfortunately, there is a minor bug in TLDIS, which causes difficulty in disassembles around 7D90-7DB0H. I used to have a fix for this - if anybody needs it, let me know and I will try to recreate it, or find it in my files (i.e. the large heap of paper in the dining room that my wife keeps tripping over.)

The essential thing about a disassembler is that it should be relocatable - i.e. able to be moved to an area of RAM which doesn't conflict with the program being disassembled. This rules out disassemblers in BASIC for many purposes.

A MONITOR is a piece of software which enables you to interact with a machine code program as it is running.

Typically a monitor will permit single stepping through a program, inspecting and modifying the microprocessor registers and memory, outputting the contents of memory to a screen or printer, executing a program, setting 'breakpoints', that is, points at which the program will halt and return control to the user, and various other facilities.

The original EDTASM had a companion monitor, T-BUG. This was all right in its day, but in my view anyway, is a bit too simple.

EDTASM-PLUS was accompanied by Z-BUG, which is considerably more powerful, and has the advantage that it can be co-resident with the editor, and the assembler, making switching from one to the other very quick and easy.

In fact, for speedy development of short or medium length programs, up to 3 or 4 Kbytes or so, I find EDTASM-PLUS with Z-BUG the most convenient - although you are limited to tape.

Other monitors abound - there are literally hundreds! TASMON is a very good one - easy to use, plenty of commands, with useful utilities. Unfortunately, TASMON is not relocatable, so your program cannot occupy the same space that it does.

There is another program, locally available, called ULTRA-MON which is also very good, and has the advantage of being relocatable. There are numerous others, and perhaps some of you can pass on your opinions of some of them?

There are many other useful tools for the machine language programmer. Of course you need a book detailing the Z-80 instruction set. You also need to understand the structure of your machine, especially the memory map. And don't forget, there are all those routines in ROM and in DOS which you can use! There are many many books about these things.

The one I refer to most is MICROSOFT BASIC DECODED, by James Favour - another of the TRS-80 MYSTERIES series.

Next month I will list some of the routines in ROM which are of most use. Some are well known, others will be new to you I think.

From page 6

```
1005  DATA  JOHN,PHONE NUMBER NOW
      123456
1006  DATA  MORTGAGE,NEXT PAYMENT
      JAN 20TH
1007  DATA  DOG,VET ON MARCH 7
1008  DATA  CAR, WARRANT DUE JUN 2
      0
1009  DATA  CAR,ENGINE REPAIRS $25
      0
1010  DATA  SPRAY, DILUTE 1:7
1020  DATA  END
```

For those of you wishing to convert this program from Applesoft BASIC the following should have equivalents in your computer manuals.

VTAB and HTAB are vertical and horizontal screen tab positions.

POKE 34, 2 lowers the screen two lines leaving the headings fixed and POKE 34,0 on line 355 restores top of screen. ML=1000 sets input on line 305 to a maximum limit of 1000th items.

PEEK(37) is a machine routine to check for a maximum of 20 lines on information on screen.

READ E$ reads the first DATA entry and READ T$ the second.

SPC(2) puts two spaces between the items on screen.

Line 360 branches the program and also provides an error trap.

HOME clears the screen. GET A$ is the same as INKEY$ for TRS80/System 80.

LINE 700 prevents information scrolling off the screen once it is filled. When a key is pressed fresh information fills the screen.

And this is Notebook, a low-cost database program you can use for storing anything you like. Remember if you want to specialise you can copy the program for each specific category and save them on separate cassettes or disks.

— Copyright.
Brian Strong

The next issue in school hardware — *the multi-micro dilemma*

By NICK SMYTHE

The year 1982 saw a major wave of hardware purchasing phase pass over New Zealand schools. Almost every high school in New Zealand now has at least one computer. Many have two and a fortunate few have half a dozen or more. Already several microcomputer firms are setting up special offers to tempt primary schools to buy *their* product before anyone else's. In the secondary-sector parents and schools have acted in the absence of Government support to the tune of at least $1,500,000 to get their school computer equipment.

In most schools the one or two machines so far purchased have whetted appetites rather than fulfulled needs. Their presence has provided a facility for getting to grips with some issues of using computers in schools. It has also stimulated such interest that most machines are swamped. The result is that many schools are now looking to purchase more machines. Many are looking at a quantum jump from one or two to a computing laboratory: they plan their next purchase as a major project involving six or more microcomputers.

To move in this direction there are two strategies, both expensive. One is to add independent machines piecemeal as you go along. These machines each need a disk drive and monitor and keyboard. If you want to print results on paper means either a lot of printers or a lot of shuffling around of people and disks to the few computers with printers.

The other strategy is to get machines together to share expensive peripherals in some way. This is usually by networking them: connecting them through a common cable along which messages can pass to a common disk drive and printer(s), and possibly between users. The first reason for networking is that you can use fewer disk drives and printers. Since screens and microcomputer components are now getting to be the cheapest component of the whole operation a network lets you add new stations (keyboards) relatively cheaply while sharing the expensive and intermittently used disks and printers. There are other educational reasons for networking, too. The most common one is the ability to set a whole class off on a given unit together and monitor individual progress from a central machine. There are some more creative ways that potential user interaction can be used as well.

A new issue is emerging in school computing: what is a good network system?

Introducing a new column for teachers and others interested in computer education compiled by Mike Wall, lecturer in computer education at Christchurch Teachers College.

Contributions and inquiries are welcomed and should be directed to Mike Wall, 56 Wayside Avenue, Christchurch 5 or c/o BITS & BYTES, Box 827, Christchurch.

The machine you buy as a single machine may not be the one you want to build a network around. You may want a network that will support more than one kind of machine so that in future you can add new makes of equipment without disrupting the whole working system. Will a particular make let you use such a network?

At its simplest a "network" may just be a multiplexing arrangement where access to the printer or disk by any user is shared. If someone-else is using the peripheral that you want then you either have to come back later or your request gets queued until the peripheral becomes free. In either case queueing causes you delay, in the formal instance requiring you to keep calling until you get through.

However, grander networks can offer a wide range of features. They can for instance allow individual machines to talk directly to each other and do tricks such as view each other's current screens. This has use in certain teaching applications. The most common use to date is where one "master" computer will download a similar program to the other "slaves" on the network or will poll the other users to see what their progress is. Networks also come with software that handles security and shares disk space around between users. In a classroom that can be useful in keeping certain program or data files safe from harm or regulating how much storage on disk users get. You can ensure only the right people get access to PacMan or Galaxion, too.

A typical network layout is shown in Figure 1. Usually the microcomputers are linked in to a cable that also has on it at least two "servers". These are stations that control the operation of the printers and disks. They should hold requests for access to the disk until the disk is free and spool any printing. Spooling usually entails storing printing instructions to disk until the printer is free, so freeing each user's own station for more computing rather than have users waiting for their turn to print. Sometimes the printer and disk servers are independent pieces of equipment, as is the case with the Poly disk/printer drive. The central Poly drive spools printing and handles disk requests. More normally you have to dedicate a microcomputer to the task. This may also control inter-machine communication but often this is independent of the servers.In some networks you can use their server computer independently still in a limited way but normally it is taken right out of consideration for other work.

It is worth noting that the cable used for linking machines in different networks varies greatly in nature and price. Some networks use cheap twisted pair cable, others use expensive multi-wire links. If

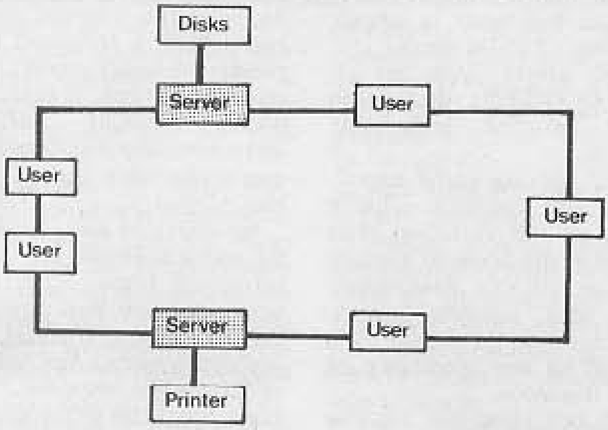


Figure 1: A typical micro-network layout, if such a thing exists.

you have machines in different places the cost of cable may be significant, and the dearest cable may not mean the best network.

Networks are expensive. The basic trade-off one is getting into is trading a high initial cost against a low cost for each new computer you add on to the ystem. One of the problems with the Poly computer in its early days was that it cost a lot to buy one, simply because the disk system with it was a large 8in disk designed to serve several Polys. To schools looking for their first computer on minimal funds and no clear idea how many they might seek in the future this was a big marketing disadvantage.

This disadvantage could be turned on its head now if schools know they are looking for a networked system, partly because the rival Apple was not originally designed for such a configuration. On the other hand it is true that there are several networking systems suitable for the Apple II, one of which is particularly advanced. The cost of these systems is high, however, and support of the systems has yet to be proved.

The black horse in this field is the BBC machine which uses a system called Econet, used in Britain and Australia but again not at the time of writing widely assessed here. Networks need some in-depth evaluation. Building on the machine you originally bought may, or may not, be the best path to a network if you think networking is for you in the first place (and you may not be now: in some cases networking may simply be inappropriate or uneconomic).

Next month Nick Smythe answers some of the questions surrounding school microcomputer networks and outlines the major networks available in New Zealand.

A load of old COBOL

"COBOL", by George Jackson. Published by TAB Books Inc., 1982
Paperback, 290 pages. $19.95. Reviewed by Warren Marett.

"Complete, up-to-the-minute reference on THE national computer business language" brags the front cover of this 1982 first edition. But the section on history and background ends in 1966 with a mention of a "proposed USASI standard COBOL". There is no word on the 1968 standard or the all-important 1974 standard.

An appendix gives assignment codes for six series of computers, all of which came out of the ark, literally speaking (whatever did happen to the last few IBM 1130s in New Zealand?). The book frequently refers to 80-column punched cards with scant regard for more modern methods of input. The glossary contains Electrical Accounting Machine, Magnetic Core Storage, and Film Optical Sensing Device for Input to Computers but makes no mention of microprocessor, MOS memory or floppy disk.

Could it be this book is largely adapted from a 1960s work?

The book starts with an introduction to COBOL, describing its history, benefits, and basic components.

Chapter 2 defines some important terms and finishes with a description of the notation that will be used in the book to specify the language. In this description there are four printing errors, which is enough to make one uncertain as to the credibility of the rest of the book.

The next ten chapters are the heart of the book, giving the elements of the language in the same style as presented in a COBOL reference manual. Examples are given throughout these chapters to illustrate each element. However, the major example, a standard master/transaction update exercise, is badly designed and poorly described.

Few pointers on programming style are given and there is no mention of modern programming techniques.

Nearly 130 pages are then devoted to a glossary on COBOL and a glossary on data processing, the most notable feature of which is an annoying habit of listing unnecessary cross-references, such as "language, common business oriented — See Common Business Oriented Language". There is no list of COBOL reserved words to aid programmers.

An appendix which attempts to describe the internal operation of computers is followed by the appendix on assignment codes and a two-page index. A good reference book should either be alphabetically-organised or have a comprehensive index. This book has neither.

Needless to say, the book can't be recommended. If you want a reference book on COBOL then you would be well advised to use the COBOL manual from a reputable computer manufacturer or software supplier. Then this book has little more to offer.

One for words people

"Introduction to Word Processing," by Hal Glatzer. Sybex. 210 pages. N.Z. price $22.75. Reviewed by John Neill.

A question frequently thrown at home-computer enthusiasts is, "But what use are these machines?" People sitting in gigs or astride hacks probably asked the same question about motorcars 75 years ago. Home computers don't have to be useful to justify their purpose. At least not any more than does a set of golf clubs, a ham radio, or photographic equipment. But there is no question that microcomputers do offer many potential uses, and one of these is word processing.

Glatzer's book is an ideal introduction to word processing for the home-computer initiate. What word processing is, what it does, how to get more information and some tips on how to buy the right package are spelled out simply, and with helpful illustrations.

The author shows how microcomputers compare extremely favourably with stand-alone units and mainframes for word processing.

Writing, editing, the types of files, formatting, printers, output to microfilm, and telecommunications are covered. Inevitably, some of the tips for the book's mainly American audience will not apply in New Zealand, but with a little commonsense all of the points can be adapted for this country.

In his review of the BBC microcomputer in the November issue of BITS & BYTES, Pip Forer spoke of one of the great divides in civilisation being between those who follow the Z80 based microcomputers and those of owe their allegiance to 6502-based machines.

Similarly there is a great gulf in most English-speaking countries between those whom the education system drafted into the pen for those "good at maths", and the rest who went into the other pen, for those "good at words". If you are one who has been left with a lifelong complex about maths because of the drafting, you really should get into word processing to obtain the best value from your microcomputer. All who write letters, reports, articles, even, perhaps, poetry, will derive great benefit from word processing. Reading Glatzer's book would be a good first step.

Pascal Games

"Apple Pascal Games" by Douglas Hergert & Joseph T. Kalash, Sybex Inc. 371 pages. Price in New Zealand $25.95. Reviewed by Gerrit Bahlman.

With the advent of a large number of Apple computers in secondary schools. it would be assumed by many that BASIC would now totally dominate the interest of secondary students. BASIC games would be generated from all sorts of sources, in particular from books such as Mateosan's "Inside Basic Games". But, with the advent of these Apple computers there came also language boards and Pascal packages. However, apart from the few enlightened souls, that partciular package may well rest in the depths of some back-of-the-classroom cupboard.

Games are an inevitable drawcard for the new computer fan, availability being the primary criterion in terms of listings that can be typed in and then run. A significant library of material is available in BASIC, and now thanks to "Apple Pascal Games" there are at least some in Pascal.

The book provides the newcomer to Pascal with 27 graded games, starting with beginners' favourites such as Guessit, Keno, Nim, and Blackjack and ranging to some moderate examples of the use of turtlegraphics in Gunshot Picture and Cribbage.

The book leads the beginner through the main features of the Pascal language and makes an important attempt to explain diagramatic structuring of a program. Of those programs which are commonly found in BASIC books on games it is particularly interesting to compare the listings of the programs. The clarity of Pascal will become evident to the reader as will the relative brevity of the listings.

Each game is explained prior to a listing being given and where practical a sample run is provided. The result is an overview of the game prior to approaching the listing, which is further simplified by careful attention to programming "tricks" used to achieve particular effects.

The level of complexity achieved in the games is well graded and there is plenty of scope for individual enhancements. Listings are well commented with indications as to where alterations may be made.

The graphics employed is not overly complex but the major facilities of the Turtlegraphics module are clearly demonstrated.

The level of originality is deliberately not high. It is clear from the comments of the author that this book was written as a parallel to "Inside BASIC Games", reviewed in this magazine last month, and some of the BASIC games presented there are duplicated here. That is not to suggest that there are no original games; merely that this book is a tool to achieve a level of confidence in Pascal. Familiar games at the simple level are one way to introduce readers to the versatility and elegance of Pascal.

From page 31

starts (clears) all the memory contents.

The second item on the menu enables you to enter the monitor to access the internal 6301 CPU machine language, to change operating parameters and to control registers etc.

The third item on the menu is BASIC. This is an enhanced Microsoft BASIC that has many added words to control the features specific to the HX-20.

To give some idea of the total number of commands and statements the appendix lists 140 reserved words, while there are 78 pages of commands and 28 pages of functions in the BASIC preference manual.

Powerful BASIC

It is a very powerful dialect of BASIC that bears no relation to the size of the computer.

The BASIC is flexible in other ways as well. Five RAM areas are available for completely separate programs.

Program area No. 1 is always selected when BASIC is first executed (Menu item No. 2). A program can be named and will then appear in the menu by use of the title " . . . ." command. This program is now protected against deletion and the only way it can be new'd is be retitling it as ". . . .".

To access any unused program area the login . . . command is used. Even if the power is turned off before a program is named it is retained in memory.

The only disadvantage I found was the time spent in searching through the two manuals to find out how to get to an unused program area.

What are my impressions of this deceptively small computer?

While it is not readily suited for use as a fully fledged portable word processor due to its inbuilt screen size, never the less it will create its own niche in the business market due to its special abilities. These include: true portability; mobile communications via telephone modem or by direct connection to another computer; and memory saving which enables it to be used as an appointments schedule, memo pad, data collector and data reference source all packaged together.

CLASSIFIEDS

FREE. Free classified advertisements (up to 20 words) are offered to members of micro clubs, students, and hobbyists generally. Over 20 words, and a small charge is made.

Wanted ZX80 users to swap software and information. Carl McNeil, 19 Royal Terrace, Alexandra, Central Otago.

Garry Fejoa, formerly 56 Morris Rd, Auckland 6. Please advise subscriptions, BITS & BYTES, of your new address.

For Sale, ZX80, 40 different pieces of software, manuals, lots of articles, $100, Carl McNeil, Royal Terrace, Alexandra. Ph. 8426.

Andrew Were. Please advise BITS & BYTES subscriptions of your new address.

Money from your printer (if you live in Christchurch). Wanted to rent, evenings, or preferably week-ends, a printer, letter quality if possible. Ring Neill Birss 66-566 or 588-750 (home).

For Sale: Sinclair ZX81, full keyboard, 16K, video or modulated RF output, inverse video switch, much software (chess), $400. Tel. Wellington 283-280.

For Sale: ZX80 computer, 2K RAM power pack and lots of software inc, $125 o.n.o., 133 Reeves Rd, Pakuranga, Auckland. Ph: 565-493.

For Sale: Wizzard software. Software list and club details available from Brian Mayo, Church Street, Katikati. Phone 490-326.

2650: I have a 2650 and would like to hear from anyone with one. Write to David Murray, P.O. Box 13-675, Christchurch.

I have a Heath/Zenith *Z89* Micro and wish to correspond with other owners of *Z89s* with the view of starting a N.Z. Heath User's Group. I can be contacted at the following address: 94 Dowse Drive, Maungaraki, Lower Hutt.

For Sale: Commodore 4022 Printer. One year old. Contact Kerry Marshall, ph 437-615, Napier or P.O. Box 3036, Napier.

For Sale: Commodore Pet 32K, 4040 dual disk drive, cassette unit, sound unit, over 500 programs, toolkit, disk-o-pro chips etc. Phone Nelson 85-494 evenings.

Computer: Colour $4,200.00 o.n.o. Colour screen, deluxe keyboard 2 x 5¼ in. disk drives, Microline printer, Floppy disks and software. Ideal hobbiest and/or small business. Phone Tauranga 84-453 or 84-175 after hours.

VIC-20's for sale. For the best all-round deal ring Brian Bullen. Ph 298-8676 Papakura.

For Sale: VIC-20 Computers, peripherals and software. Discount prices and free software with every computer and peripheral. PERSONAL COMPUTER SYSTEMS, P.O. Box 860, Nelson. Telephone NN 79-362.

VIC-20 Users: Would you like access to a large library of public domain software? Then join N.Z.'s first VIC club. For details write to Nelson VIC Users Group, P.O. Box 860, Nelson.

For Sale: Heath/Zenith H-89A, all-in-one computer. 48K RAM, 3 Serial I/O ports, disk drive. HDOS operating system with Extended B.H. BASIC, plus CP/M. Comprehensive manuals. $5750. Christchurch 852-881 after 5 p.m.

Sell: MICRO (U.S.) magazine, issues 19, 25-26, 28-29, 31-34, 36. $10 for the lot. Telephone Steven Darnold, Alexandra 6833.

Wanted: A System 80, Blue Label, with or without software, price negotiable. Contact Rod, 639 Worcester St, Christchurch. Ph 895-431.

COMMERCIAL CLASSIFIEDS

Commercial Classifieds are available at 50c per word (minimum of 15 words). BITS & BYTES reserves the right to determine if any advertisement is a commercial classified. All commercial classifieds must be paid for before publication unless by prior arrangement.

CLUB CONTACTS

WHANGAREI COMPUTER GROUP: Tom Allan,
3 Maunu Rd, Whangarei. Phone 83-063 (w). Meets every second Wednesday of the month at Northland Community College.

NZ MICROCOMPUTER CLUB INC., P.O. Box 6210, Auckland. The Monthly Meeting is held the first Wednesday of each month at the VHF Clubrooms, Hazel Ave., Mt Roskill, from 7.30 p.m. Visitors are also welcome to the Computer workshop in the Clubrooms, 10am – 5pm, on the Saturday following the above meeting.

The following user groups are part of the club. All meetings shown start 7.30pm at the VHF Clubroom.

APPLE USERS' GROUP: Bruce Given, 12 Irirangi Rd., One Tree Hill, Phone 667-720 (h).

ATARI MICROCOMPUTER USERS GROUP: Brian or Dean Yakas, Phone 8363 060 (h). Meetings: 1st Monday.

BIG BOARD USER GROUP: Steve Van Veen, Flat 5, 111 Melrose Rd, Mt Roskill, Auckland 4. Phone (09) 659-991 (h).

COMMODORE USERS' GROUP: Doug Miller, 18 Weldene Ave., Glenfield. Phone 444-9617 (h), 497-081 (w). Meetings: Third Wednesday.

CP/M USERS' GROUP: Kerry Koppert, 2/870 Dominion Rd., Balmoral. Phone 69-5355 (h).

DREAM 6800 USERS: Peter Whelan, 22 Kelston St, New Lynn, Auckland, Phone (09) 875-110 (h).

KIM USERS: John Hirst, 1A Northboro Rd, Takapuna, phone (09) 497-852 (h).

LNW USERS: Ray James, phone (09) 30-839 (w), 585-587 (h).

SINCLAIR USERS' GROUP: Doug Farmer, Phone 567-589 (h). Meetings: Fourth Wednesday.

SORCERER USERS' GROUP (NZ): Selwyn Arrow, phone 491-012 (h).

1802 USERS' GROUP: Brian Conquer, phone 655-984 (h).

2650 USERS' GROUP: Trevor Sheffield, phone 676-591 (h).

The above contacts can usually be found at NZ Microcomputer Club Meetings, or via P.O. Box 6210, Auckland.

Regular Meetings are:

MICRO CLUB, First Wednesday, plus an all day Computer Workshop the Saturday following, (10am – 5pm), all welcome.

ATARI MICROCOMPUTER USERS' GROUP: First Monday.

COMMODORE USERS' GROUP: Third Wednesday.

SINCLAIR USERS' GROUP: Fourth Wednesday. All meetings start at 7.30 pm at the VHF Clubrooms, at the end of Hazel Ave. (off Dominion Rd), Mt Roskill, Auckland.

Other active User Groups within the Club are:

APPLE, CP/M, DREAM 6800, SMALL BUSINESS, KIM, LNW, SORCERER, 1802 and 2650. They can all be contacted at club meetings or via NZ Microcomputer Club, P.O. Box 6210, Auckland.

Other Auckland-based groups:

ACES (Auckland Computer Education Society): Ray Clarke, 1 Dundas Pl., Henderson, Phone 836-9737 (h).

CMUG (Combined Microcomputer Users' Group): This is an association of Microcomputer Clubs, Groups, etc, formed to co-ordinate activities and to give a combined voice on topics concerning all micro users. Representation from all Clubs and Groups is welcomed to: CMUG C/- PO Box 6210, Auckland.

HP41C USERS' GROUP (Auckland): C/- Calculator Centre, P.O. Box 6044, Auckland: Grant Buchanan, 790-328 (w). Meets third Wednesday, 7pm, at Centre Computers, Great South Rd., Epsom.

NZ TRS-80 MICROCOMPUTER CLUB: Olaf Skarsholt, 203A Godley Rd., Titirangi. 817-8698 (h). Meets first Tuesday, VHF Clubrooms, Hazel Ave., Mt Roskill, Auckland.

OSI USERS' GROUP (Ak): Vince Martin-Smith,
44 Murdoch Rd., Grey Lynn, Auckland. Meets
third Tuesday, VHF Clubrooms, Hazel Ave.,
Mt Roskill.

SYMPOOL (NZ SYM USER GROUP): J. Robertson, P.O. Box 580, Manurewa, Phone 266-2188 (h).

NOTE: If your club or group is not listed, drop a line with the details to: Club Contacts, BITS & BYTES, Box 827, Christchurch. The deadline for additions and alterations is the second weekend of the month before the next issue.

BITS & BYTES also welcomes news of club occurrences, lists of offices, etc., Let's hear from you!

A.Z.T.E.C.: Brian Mayo, Church Street, Katikati. Phone 490-326. Members use all micros and the club has just bought a Wizzard.

TAURANGA SINCLAIR COMPUTER CLUB: C. Ward, Secretary, P.O. Box 6037, Brookfield, Tauranga. Phone 82962 or 89234.

ATARI 400/800 USER CLUB: Dave Brown, P.O. Box 6053, Hamilton, Phone (071) 54-692 (h).

GISBORNE MICROPROCESSOR USERS' GROUP: Stuart Mullett-Merrick, P.O. Box 486, Gisborne, Phone 88-828.

ELECTRIC APPLE USERS' GROUP: Noel Bridgeman, P.O. Box 3105, Fitzroy, New Plymouth, Phone 80-216.

TARANAKI MICROCOMPUTER SOCIETY: P.O. Box 7003, Bell Block, New Plymouth: Francis Slater, Phone 84-514.

HAWKE'S BAY MICROCOMPUTER USERS' GROUP: Bob Brady, Pirimai Pharmacy, Pirimai Plaza, Napier, Phone 439-016.

MOTOROLA USER GROUP: Harry Wiggins, (ZL2BFR), P.O. Box, 1718, Palmerston North, Phone (063) 82-527 (h).

OSBORNE USER GROUP: Dr Jim Baltaxe, 18 Matipo St, Palmerston North, Phone (063) 64-411.

MICRO AND PEOPLE IN SOCIETY (MAPS): Levin. Meets on second and fourth Thursday of each month. Contacts: D. Cole, 28 Edinburgh St, Levin. Ph 83-904, or W. Withell, P.O. Box 405, Levin.

CENTRAL DISTRICTS COMPUTERS IN EDUCATION SOCIETY: Contact: Rory Butler, 4 John Street, Levin. (069) 84-466 or Margaret Morgan, 18 Standen Street,Karori, Wellington. (04) 767-167.

UPPER HUTT COMPUTER CLUB: Shane Doyle, 18 Holdworth Avenue, Upper Hutt. Phone 278-545. an all-machine club.

NZ SUPER 80 USERS GROUP: C/- Peanut Computers, 5 Dundee Pl., Chartwell, Wellington 4. Phone 791-172.

WELLINGTON MICROCOMPUTING SOCIETY Inc.: P.O. Box 1581, Wellington, or Bill Parkin (h) 725-086. Meetings are held in Wang's Building, 203-209 Willis Street, on the 2nd Tuesday each month at 7.30 p.m.

BBC USER GROUP. Users of other machines welcome too. Write P.O. Box 1581, Wellington, or Phone 861-213, Wellington.

NELSON MICROCOMPUTER CLUB: Dr Chris Feltham, Marsden Valley Rd, Nelson. Phone (054) 73300 (h)

NELSON VIC USERS GROUP: Peter Archer, P.O. Box 860, Nelson. Phone (054) 79-362 (h).

BLENHEIM COMPUTER CLUB: Club night second Wednesday of month. Ivan Meynell, Secretary, P.O. Box 668, Phone (h) 85-207 or (w) 87-834.

CHRISTCHURCH '80 USERS' GROUP: David Smith, P.O. Box 4118, Christchurch, Phone 63-111 (h).

CHRISTCHURCH PEGASUS USERS' GROUP: Don Smith, 53 Farquhars Rd, Redwood, Christchurch, Phone (03) 526-994 (h), 64-544 (w), ZL3AFP.

CHRISTCHURCH APPLE USERS' GROUP: Paul Neiderer, C/- P.O. Box 1472, Christchurch, Phone 796-100 (w).

OSI USERS GROUP (CH): Barry Long, 377 Barrington St., Spreydon, Christchurch. Phone 384-560 (h).

CHRISTCHURCH SINCLAIR USERS' GROUP: Contact, Mr J. Mitchell (385-141), P.O. Box 33-098.

CHRISTCHURCH COMMODORE USERS' GROUP: Contacts: John Kramer, 885-533 and John Sparrow, 896-099.

SOUTH CANTERBURY COMPUTERS' GROUP: Caters for all machines for ZX81 to IBM34. Geoff McCaughan, Phone Timaru 84-200 or P.O. Box 73.

LEADING EDGE HOME COMPUTER CLUB: Elaine Orr, Leading Edge Computers, P.O. Box 2260, Dunedin. Phone 55-268 (w).

GLOSSARY

Algorithm: A list of instructions for carrying out some process step by step.

Applications program: A program written to carry out a specific job, for example an accounting or word processing program.

BASIC: Beginners' All-purpose Symbolic Instruction Code. The most widely used, and easiest to learn, high level programming language (a language with English-like instructions) for microcomputers.

Binary: The system of counting in 1's and 0's used by all digital computers. The 1's and 0's are represented in the computer by electrical pulses, either on or off.

Bit: Binary digit. Each bit represents a character in a binary number, that is either a 1 or 0. The number 2 equals 10 in binary and is two bits.

Boot: To load the operating system into the computer from a disk or tape. Usually one of the first steps in preparing the computer for use.

Buffer: An area of memory used for temporary storage while transferring data to or from a peripheral such as a printer or a disk drive.

Bug: An error in a program.

Byte: Eight bits. A letter or number is usually represented in a computer by a series of eight bits called a byte and the computer handles these as one unit or "word".

Character: Letters, numbers, symbols and punctuation marks each of which has a specific meaning in programming languages.

Chip: An integrated circuit etched on a tiny piece of silicon. A number of integrated circuits are used in computers.

Computer language: Any group of letters, numbers, symbols and punctuation marks that enable a user to instruct or communicate with a computer. See also Programming languages and Machine language.

Courseware: Name for computer programs used in teaching applications.

CP/M: A disk operating system available for microcomputers using a particular microprocessor (that is the 8080 and Z80 based microcomputers such as the TRS 80 and System 80). See also Disk Operating Systems.

Cursor: A mark on a video that indicates where the next character will be shown, or where a change can next be made.

Disk: A flat, circular magnetic surface on which the computer can store and retrieve data and programs. A flexible or floppy disk is a single 8 inch or 5¼ inch disk of flexible plastic enclosed in an envelope. A hard disk is actually an assembly of several discs of hard plastic material, mounted one above another on the same spindle. The hard disk holds much more information – up to hundreds of millions of bytes – while floppy disks typically hold between 140,000 and three million bytes.

Disk drive: The mechanical device which rotates the disk and positions the read/write head so information can be retrieved or sent to the disk by the computer.

Diskette: Another name for a 5¼ inch floppy disk.

Disk operating system: A set of programs that operate and control one or more disk drives. See CP/M for one example. Other examples are TRSDOS (on TRS 80) and DOS 3.3 (for Apples).

DOS: See Disk Operating System.

Dump: Popular term for sending data from a computer to a mass storage device such as disks or tape.

Execute: A command that tells a computer to carry out a user's instructions or program.

File: A continuous collection of characters (or bytes) that the user considers a unit (for example on accounts receivable file), stored on a tape or disk for later use.

Firmware: Programs fixed in a computer's ROM (Read Only Memory); as compared to software, programs held outside the computer.

Floppy disks: See Disks.

Hard disks: See Disks.

Hardware: The computer itself and peripheral machines for storing, reading in and printing out information.

High-level language: Any Englishlike language, such as BASIC, that provides easier use for untrained programmers. There are now many such languages and dialects of the same language (for example MicroBASIC, PolyBASIC etc).

Input: Any kind of information that one enters into a computer.

Input device: Any machine that enters information into a computer. Usually done through a typewriter like keyboard.

Interactive: Refers to the "conversation" or communication between a computer and the operator.

Interface: Any hardware/software system that links a microcomputer and any other device.

I/O Acronym for "input/output".

KILOBYTE (or K): Represents 1024 bytes. For example 5K is 5120 bytes (5 x 1024).

Machine language: The binary code language that a computer can directly "understand".

Mass storage: A place in which large amounts of information are stored, such as a cassette tape or floppy disk.

Megabyte (or Mb): Represents a million bytes.

Memory: The part of the microcomputer that stores information and instructions. Each piece of information or instruction has a unique location assigned to it within a memory. There is internal memory inside the microcomputer itself, and external memory stored on a peripheral device such as disks or tape.

Memory capacity: Amount of available storage space, in Kbytes.

Menu: List of options within a program that allows the operator to choose which part to interact with (see Interactive). The options are displayed on a screen and the operator chooses one. Menus allow user to easily and quickly set into programs without knowing any technical methods.

Microcomputer: A small computer based on a microprocessor.

Microprocessor: The central processing unit or "intelligent" part of a microcomputer. It is contained on a single chip of silicon and controls all the functions and calculations.

Modem: Modulator-demodulator. An instrument that connects a microcomputer to a telephone and allows it to communicate with another computer over the telephone lines.

Program: A set or collection of instructions written in a particular programming language that causes a computer to carry out or execute a given operation.

RAM: Random access memory. Any memory into which you "read" or call up data, or "write" or enter information and instructions.

REM statement: A remark statement in BASIC. It serves as a memo to programmers, and plays no part in the running program.

Resolution: A measure of the number of points (pixels) on a computer screen.

ROM: Read only memory. Any memory in which information or instructions have been permanently fixed.

Simulation: Creation of a mathematical model on computers that reflects a realistic system.

Software: Any programs used to operate a computer.

Storage: See Mass storage.

System: A collection of hardware and software where the whole is greater that the sum of the parts.

Tape: Cassette tape used for the storage of information and instructions (not music).

VDU: Visual display unit. A device that shows computer output on a television screen.

Word: A group of bits that are processed together by the computer. Most microcomputers use eight or 16 bit words.

It's less hassle regularly

Having difficulty getting your copy of "Bits & Bytes" regularly? We suggest that you become a subscriber. If you prefer to buy from your bookstore or computer shop place a regular order. This will help ensure you always get a copy.

BACK COPIES AND SUBSCRIPTIONS

All subscriptions run from the next issue of BITS & BYTES after the subscription is received.

Back copies are available for all issues except number 1 (although numbers are limited).

The price for back copies is $1 PER ISSUE plus 50 cents postage and packaging PER ISSUE.

Please send your cheque or postal note with your order for back copies.

BITS . . & . . .
. . BYTES . .

NEW ZEALAND'S
PERSONAL COMPUTER MAGAZINE

If you want the latest micro news on
- * education ✓
- * business ✓
- * clubs ✓
- * hardware ✓
- * software ✓

If you want micro
- * games ✓
- * hints ✓
- * help ✓
- * competitions ✓

.... then this is your magazine. There is a small subscription charge (to cover postage costs). Just fill out the card enclosed in the magazine or the coupon below.

No stamp is required if you
write this address on your envelope

Freepost No. 125
Bits & Bytes
P.O. Box 827
Christchurch

POST YOUR SUBSCRIPTION TODAY!!

Please tick appropriate boxes

NAME ....

ADDRESS ....

....

....

Please mark which field interests you

- Education ☐
- Competitions ☐
- Business use ☐
- Hobby use ☐
- Farming ☐
- General news ☐
- Programs ☐
- Hints ☐

School ☐ Adult ☐

If school name it ....

Form ....

I enclose payment for one year's subscription to Bits & Bytes Adult $8 ☐ School $6 ☐

Payment
Visacard ☐
Bankcard ☐

Card holder signature ....

Date card expires ....

Cheque ☐ Cheque No. ....
Postal Order ☐ Postal note ☐

Here is the news. . . .

The B.B.C. has arrived

"Firstly the machine has been laid out with a clear growth path in mind: you can see how the design has room for expansion.
Secondly the machine as it stands is highly flexible. It possesses the 'openness' of the Apple plug-it-in peripheral card philosophy, the screen software flexibility of the Atari 400 and 800 machines and the teletext compatibility of Poly.
Finally it seems to have taken all the little edges that most current machines possess, rounded them off and then French polished them."

Dr. P. FORER in Bits and Bytes
November, 1982

B.B.C. Micro

The face of things to come!

FACTS THAT MAKE THE BBC BETTER

- Price... is competitive... $1995
- Good Colour... essential for entertainment and in presenting information, you can highlight different areas at the same time.
- The resolution of the screen allows 640 dots x 256 dots which makes the BBC the best system available at this price. When making Graphs and curves, the better the resolution the clearer the curve.
- Sound... four sound channels, three note generators, one noise generator. Full sound envelope control on sound channels. Is adaptable to have a voice synthesizer fitted. The production of the sound does not stop or delay the running of the central process.
- Basic... allows the use of PROCEDURES and REPEAT UNTIL loops, which give it the advantages of a structured language.
- The BBC model B is the computer of the future... It is able to be hooked up to a second processor when necessary, the Z80 and 6502 already released. The BBC will never go out of date! A.16 Bit Processor to be released soon.

☆ **EDUCATION**
☆ **TRAINING COURSES AVAILABLE**
☆ **ON SITE DEMONSTRATIONS**
☆ **HOME DEMONSTRATIONS**
☆ **FULL TECHNICAL ADVISORY SERVICE**
☆ **GAMES/ENTERTAINMENT**

Distributed by:

access Access Data Ltd Auckland Phone 686 578

Branches in CHRISTCHURCH and WELLINGTON — and authorised dealers.