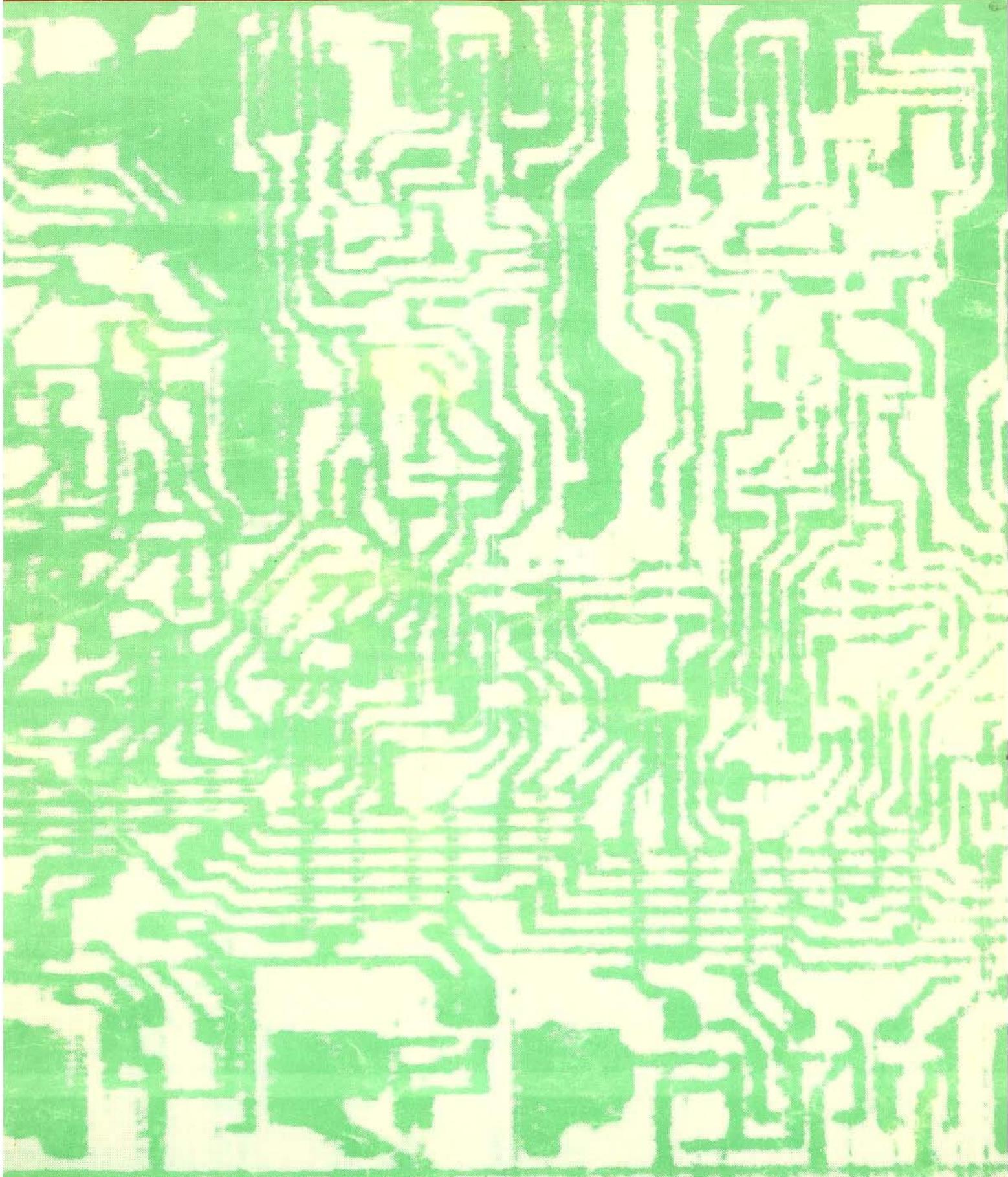


**RS-80 SYSTEM 80 VIDEO GENIE**



**MICRO-80**

PO BOX 213, GOODWOOD, S.A. 5034, AUSTRALIA. TELEPHONE (08) 272 0966. PRICE: AUS. \$2.50, NZ. 3.00, UK. £1.50

**\*\* ABOUT MICRO-80 \*\***

EDITOR: IAN VAGG  
ASSOCIATE EDITORS:  
SOFTWARE LEVEL I : MICHAEL SVENSDOTTER  
SOFTWARE LEVEL II: CHARLIE BARTLETT  
HARDWARE : EDWIN PAAY

MICRO-80 is an international magazine devoted entirely to the Tandy TRS-80 microcomputer and the Dick Smith System 80/Video Genie. It is available at the following prices (all prices shown in Aus.\$ except for U.K. prices which are in pounds Sterling).

12 months subscription	Aus.	\$24.00
	NZ.	\$36.00 (Airmail)
	Hong Kong	\$46.00 (Airmail)
	U.K.	16.00 pounds
Single Copy	Aus.	\$2.50
	N.Z.	\$3.50 (Airmail)
	Hong Kong	\$4.25 (Airmail)
	U.K.	1.50 pounds
Months programs on cassette	Aus.	\$3.50
	N.Z.	\$4.00 (Airmail)
	Hong Kong	\$4.50 (Airmail)
(at present available from Australia only)	U.K.	\$4.75 (Airmail)
12 months subscription to magazine and cassette	Aus.	\$60.00
	N.Z.	\$78.00 (Airmail)
	Hong Kong	\$88.00 (Airmail)
	U.K.	\$41.00 (Airmail)

Special bulk purchase rates are also available to computer shops etc. Please use the form in this issue to order your copy or subscription.

The purpose of MICRO-80 is to publish software and other information to help you get the most from your TRS-80, System 80 or Video Genie and their peripherals. MICRO-80 is in no way connected with either the Tandy or Dick Smith organisations.

**\*\* WE WILL PAY YOU TO PUBLISH YOUR PROGRAMS \*\***

Most of the information we publish is provided by our readers, to whom we pay royalties. An application form containing full details of how you can use your TRS-80 or System 80 to earn some extra income is included in every issue.

**\*\* CONTENT \*\***

Each month we publish at least one applications program in Level I BASIC, one in Level II BASIC and one in DISK BASIC (or disk compatible Level II). We also publish Utility programs in Level II BASIC and Machine Language. At least every second issue has an article on hardware modifications or a constructional article for a useful peripheral. In addition, we run articles on programming techniques both in Assembly Language and BASIC and we print letters to the Editor and new product reviews.

**\*\* COPYRIGHT \*\***

All the material published in this magazine is under copyright. That means that you must not copy it, except for your own use. This applies to photocopying the magazine itself or making copies of programs on tape or disk.

**\*\* LIABILITY \*\***

The programs and other articles in MICRO-80 are published in good faith and we do our utmost to ensure that they function as described. However, no liability can be accepted for the failure of any program or other article to function satisfactorily or for any consequential damages arising from their use for any purpose whatsoever.

## \*\*\*\*\* CONTENTS \*\*\*\*\*

	PAGE
EDITORIAL	2
PRICE CORRECTION	3
MORE ABOUT THE TANDY POCKET COMPUTER	3
MARKET PLACE	3
G.T. BASIC - Part 4	4
DISK DRIVES - QUESTIONS AND ANSWERS	6
READERS REQUESTS	8
INPUT/OUTPUT - Letters to the Editor	9
'80 USERS' GROUPS	11
MICROBUGS	12
KB FIX	12
MICRO-80 PRODUCTS	17
SOFTWARE SECTION	
FIGHTER SQUADRON .....	L1/4K 13
FOOD REQUIREMENTS .....	L1/4K 15
SNAKE .....	L1/4K 21
DRAW II .....	L2/4K 22
DRAW II INSTRUCTIONS .....	L2/4K 23
STRING EDITOR .....	L2/4K 30
MIGHTY MORMAR .....	L2/16K 24
XMAS CARD .....	L2/16K 29
KEYBOARD BLEEPER .....	L2/4K 32
THE CASE OF THE DEAD CURSOR .....	L2/16K m/1 32
CURSOR DRIVER DEMO .....	L2/16K BASIC 34
NEXT MONTH'S ISSUE	35
CASSETTE EDITION INDEX	36
ORDER FORM	36

MICRO-80 is registered for posting in Australia as a publication - CATEGORY B

AUSTRALIAN OFFICE AND EDITOR:-  
MICRO-80 PO BOX 213, GOODWOOD, SA 5034

UK SUBSCRIPTION DEPT.:-  
24 WOODHILL PARK, PEMBURY, TUNBRIDGE WELLS, KENT TN2 4NW.

Printed by:-  
Shovel & Bull printers, 312 A Unley road, HYDE PARK, SA 5061.

Published by MICRO-80, 284 Goodwood Rd., Clarence Park, ADELAIDE.

## \*\*\*\*\* EDITORIAL \*\*\*\*\*

This issue of MICRO-80 is something of a landmark. Being issue No. 12, it marks the completion of MICRO-80's first year of publication. Beset as we are with the constant round of day-to-day activities and being acutely aware of our shortcomings, it is perhaps worth pausing for a moment to look back over the past 12 months and remember some of our achievements.

We started with no previous experience in the publishing business but, as TRS-80 owners ourselves, believed that there was a need for a publication devoted to this new industry. So, we produced Issue 1 - containing programs written by the Editor and a number of friends and acquaintances around Adelaide. We placed a one-third page advertisement in Electronics Australia and sat back and waited. By the end of January, we had 155 subscribers. Our offer to pay a publication fee for readers' programs quickly bore fruit and programs started rolling in. At that time, there was no other magazine in Australia dedicated to micro-computers. The second came along in May, by which time we had launched MICRO-80 PRODUCTS with the objectives of publishing programs and books by Australian Authors and selling imported hardware and software at sensible (i.e. low) prices. One of the purposes of MICRO-80 PRODUCTS quite frankly, was to support the production costs of MICRO-80 which was growing rapidly but still had a long way to go to break even.

May was an eventful month. It saw the unveiling of our first hardware project, a memory expansion unit for the TRS-80 which enabled you to add up to 32K of memory in addition to 4K or 16K in the CPU. Unfortunately, the relatively low volume of demand and the high development costs associated with this product, caused us to shelve the project (but it is not entirely dead yet - look for some interesting announcements in future issues).

May also saw us at our first Home Computer Show in Sydney. That Show spawned one of what we consider our more significant achievements. There we met Barry Lawrence of Conquest Electronics in Katoomba. Barry was and is an authorised Tandy Dealer. He had only recently started selling TRS-80's and sought our advice. As soon as we ascertained that he was free to set his own retail prices, we recommended that he should seriously consider discounting prices for Tandy equipment. This he did and many of our readers have taken advantage of his across-the-board 10% discount offer. Others have found that, if they go to a local authorised Tandy Dealer (not a Tandy Electronics shop) waving their copy of MICRO-80 with Conquest Electronics' advertisement prominently displayed, they will usually get a 10% discount, too - (Hint - Conquest Electronics' prices include freight, why not try for a little more than 10% discount from your local dealer?) We believe that we have been instrumental in saving readers several thousand dollars already and of that, we are proud.

Still in May, we acquired one of the first System 80's delivered in Australia. We quickly discovered a number of incompatibilities between it and the TRS-80 and, when we published the first detailed review in Australia (and the world?) of this computer, it was hard-hitting and spelled out the incompatibilities and deficiencies. It caused something of a furore in areas where sales budgets are king and we were accused of bias and of being uninformed. It is interesting to note that, by early next year, just about everything we criticised will have been remedied, i.e. there will be a volume control and VU meter on the cassette deck plus right-arrow and clear keys. The unusual 32 character mode has not yet been changed but at least the important things will have been done. We can only speculate as to whether MICRO-80 influenced these changes, or not.

About the same time Electronics Today International, in reviewing MICRO-80, described us as being "proudly independent". That we are! It was as a result of some of the things we saw around us in the publishing business that we resolved that the best way to maintain that independence was to discourage advertising in MICRO-80. That policy is one which has been applauded by many of our readers and, despite the financial temptations, will remain. The exception will be for products and services which we believe to be of real value to our readers.

MICRO-80 is a part-time occupation for all the editorial staff and, by mid-year, that was starting to show. Publication dates gradually slipped back until issues were a full month late. Our administration became rather overstrained and a few subscribers seemed to have enormous trouble in getting our administrative systems to recognise their continued existence! It is little compensation but it is interesting to note that this malady is fairly typical of the emerging microcomputing industry. The TRS-80 itself was several months behind schedule, the System 80 was 12 months late and the S 100 expansion for it about 6 months late. Another Australian monthly magazine went from Issue 5 in September to Issue 6 in November. The big budget 80' Microcomputing from the USA seems to have problems in its subscription department if the sporadic delivering of our own copy is anything to go by. All this is caused by rapid growth and limited resources, coupled with technical problems and the fact that we are all still fairly low down on the learning-curve, the big companies included. Still, despite these problems, we continued to notch up a number of notable achievements.

We published Eddy Paay's Level II ROM Reference Manual, a book which has proved of considerable value to many machine language programmers and is undoubtedly of world standard. We started work on an interface for the Olivetti ET-121 typewriter and succeeded where two other bigger Australian organisations had failed. We tackled one of the other incompatibilities between the System 80 and the TRS-80 - its expansion bus. We wanted to run the Exatron Stringy Floppy from a System 80, so we developed the SYSPAND 80 which plugs on to the back of the System 80 and generates a Tandy bus as well as providing a Centronics printer port. Not only can SYSPAND 80 users run a Stringy

Floppy, they can also use the Tandy expansion interface, disk drives etc. etc. In more recent times, we took Tandy to task over the ridiculous price they charge for a lower case modification for the Model 1 and then set about developing our own, cheaper alternative. (This will be released shortly). A System 80 version is also nearing completion, so soon everyone will be able to enjoy proper lower case at sensible prices.

Towards the end of September, Eddy Paay joined us as our first full-time member of staff (yes, all the above had been done at night and on weekends!) In October, we went to the Melbourne Computer Show and we also launched MICRO-80 in the UK. Despite a plethora of microcomputing magazines available on the newstands, the UK had no specialist magazine for TRS-80/Video Genie users. To our surprise, we had an enquiry from Holland and now MICRO-80 is on sale there, too. Don't be surprised if you see a few paragraphs in Dutch in future issues. It is Eddy Paay's native tongue and we would like to show our Dutch readers that they are welcome, too.

We returned from the Melbourne Show to some problems. Publication dates for MICRO-80 were now a good seven weeks late and we were up to three weeks late on orders for MICRO-80 PRODUCTS. Our manual mailing list was on its last legs and our Software Editor resigned because his own business commitments had become too great. Eddy quickly got the MICRO-80 PRODUCTS situation under control then one of the nicer events of the year happened. Several Adelaide readers, observing our problems, offered to help us out. We owe a vote of thanks to Jim Mayfield, who single-handedly got our mailing list onto computer, Charlie Bartlett who took over the task of Software Editing, despite a bout of ill-health now happily resolved, Michael Svensdotter who Edits Level I programs and Jon Edwards who designed the new cover for MICRO-80.

Summing up, our first year has been one of immense activity and progress. We have published over 80 full programs and numerous short routines, etc., 14 hardware articles, a series of six articles on Assembly Language programming, the first four articles in a series on programming in BASIC, answers to readers' letters, lots of hints, tips, news etc. etc. We have paid over \$2000 in publication fees to authors of programs and articles and over \$6000 in royalties for MICRO-80 PRODUCTS.

As much as has been achieved, we are far from satisfied. MICRO-80 is still running late by about seven weeks. We plan to correct this over the next three issues so that you should receive your February MICRO-80 in the first half of that month. Thereafter, we will endeavour to publish MICRO-80 in the first week of each month.

What about the next 12 months? We have acquired new premises in Adelaide at 433 Morphett Street which will become our new headquarters. As well as more space, we will have a shop in which to display and demonstrate our products, and offer the service which our customers deserve. We will

continue to improve MICRO-80. Look for the inclusion of photographs of screen displays etc. in issues in the near future. We will shortly be releasing a considerable number of new programs by Australian authors and, of course, will continue to improve and broaden our range of hardware products. Next year promises to be even more exciting than last. We are now one year older and about 10 years wiser! We know that none of this would be possible without the support of you, our readers. For your support in our first year, we thank you. We guarantee we will work even harder to earn that support next year and in the future.

\*\* SEASONS GREETINGS \*\*

This will be the last issue produced before Christmas so we at MICRO-80 would like to take this opportunity to wish all our readers a very Merry Christmas and a happy New Year. In recognition of the fact that many of you are likely to have a little more time at the keyboard over the next few weeks than you usually do, we have made this a bumper issue for programs. One which we commend in particular, is called "A Xmas Card" by Gordon Hodgson. This program restores our faith in '80 graphics!

- 0000000000 -

\*\* PRICE CORRECTION \*\*

The price for an Airmail Cassette subscription to the UK is incorrectly shown as \$41.00 on the inside front cover. It should be 41.00 pounds Sterling.

\*\*\*\*\* MORE ABOUT THE TRS-80 POCKET COMPUTER \*\*\*\*\*

It has been pointed out to us that the TRS-80 Pocket Computer is actually made by SHARP and is also available through most SHARP Agencies. The main advantage is that these Agencies often discount equipment and selling prices as low as \$168 have been reported.

- 0000000000 -

\*\*\*\*\* MARKET PLACE \*\*\*\*\*

Market place is available to any reader who has hardware to dispose of. An entry costs nothing - you pay MICRO-80 \$5.00 or 5% commission, whichever is the greater - up to a maximum of \$30, after the goods are sold. The commission is calculated on your advertised price.

SYSTEM 80 L2/16K + Dick Smith monitor + Address/Mailing List and Airmail Pilot programs. This equipment is only 3 weeks old and in perfect condition. Total list price of hardware and software \$951.50. Will sell for \*\* \$850.00 \*\*

P. S. Martin, 191 Stuart Street, Blakehurst NSW 2021 Phone: (02) 546 7090

- 0000000000 -

G.T. BASIC - PART FOUR of a continuing series  
by Peter Hartley.

Copyright (C) 1980 Peter Graham Hartley.  
All rights reserved.

Before getting to grips with last month's homework, I must correct a small mistake that I made in the very first part of this series (and I thank the various individuals and users' groups for being quick to point this out). I was a little over-enthusiastic about the number of variables available in Level II, so to put the record straight there are:-

(i)	A to Z	=	26
(ii)	A0 to Z9	=	26 * 10 = 260
(iii)	AA to ZZ	=	26 * 26 = 676

adding up to a grand total of 962 primary variables  
which isn't quite as many as I said, but still more than enough to stop us all from ever running out.

#### HOMEWORK READY?

Last month I set you all the problem of creating a single character Decimal to Hex converter, using only those functions that we have already discussed. The solution that I wanted would have looked very much like this...

```
10 POKE 16553,255
20 DATA 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
30 D = ???
40 FOR K = 0 TO D
50 READ A$
60 NEXT
70 PRINT A$
80 END
```

We will be using this as our working base for the next couple of sessions, so it might be a good idea to type it in and then run it a few times using different values for D in line 30. For any number between 0 and 15, you'll get the right answer. If you wanted to use this for a number larger than 15 though, you'll get a HOW? from Level I and an O.D. ERROR from Level II - there just isn't enough data.

It gets a bit tedious having to keep changing line 30 all the time, so right now seems a good time to examine the INPUT statement. Change line 30 to...

```
30 INPUT D
```

and when you run it you will be greeted with a question mark. Respond with any number within our range of 0 to 15, followed by (ENTER) or (NEW LINE) and it will work as well as before. What happens is that the '80 understands that it needs some outside help now, when it gets to line 30, so it waits there for you to tell it how big D is going to be.

The INPUT statement is quite flexible and can also be used to get the computer to tell the operator what it is expecting. After all, the question mark is fine for us - we've just typed the program in and know what is going on, but when someone else comes to use the program there may be some confusion.

Change line 30 to...

```
30 INPUT "ENTER A NUMBER BETWEEN 0 AND 15 ";D
```

and, while we are about it, change line 70 to...

```
70 PRINT "DECIMAL ";D ;" IS ";A$ ;" IN HEXADECIMAL."
```

and line 80 to...

```
80 RESTORE
```

and a new line 90...

```
90 GOTO30
```

and you'll find that it is a great deal tidier. The INPUT statement is now doubling up as a PRINT statement but you must remember to include the semi-colon between the closing quotation marks and the first variable name.

"Did he say FIRST variable name?"

Just to satisfy your curiosity, remembering that we can have more than one program in ol' '80 at a time, try this...

```
200 INPUT" TYPE IN THREE NUMBERS...";A,B,C
210 PRINT A, B, C
220 GOTO 200
LM=1
```

Now RUN 200 and you will find that:-

(i) if you press (ENTER) or (NEW LINE) after the first number, you'll get TWO QUESTION MARKS - telling you that there is still more to put in, just as you will if you press (ENTER) or (NEW LINE) after the second number.

(ii) if you want, you can type in three numbers with a comma between the first and second and another comma between the second and third and only press (ENTER) or (NEW LINE) once.

(iii) if you put even more numbers in, before pressing (ENTER) or (NEW LINE) the '80 will proceed to demonstrate its superior intellect with a suitably superior message.

(iv) if you press (ENTER) or (NEW LINE) without first typing any number at all, the '80 decides that you are in idiot and gives up. It simply goes away and uses the old values.

(v) any combination of the above!

O.K. Now DELETE 200-220

Let's get to grips with the lack of DATA for coping with numbers greater than 15. The immediately obvious solution is to add more DATA, but I'm not going to allow anything so inefficient. Instead, you are just going to have to sit down and learn about (i) INTEGERS this month, and (ii) ARRAYS next month, which should leave you sufficiently befuddled to deserve to enjoy your Christmas.

We'll do the INTEGERS bit first. As far as the '80 is concerned, an INTEGER is the WHOLE NUMBER NEAREST TO, AND NOT GREATER THAN, THE NUMBER YOU FIRST THOUGHT OF! As this may seem confusing, we'll get the '80 to be the Teacher. It is also a good excuse to add to your workload and teach you about RANDOM NUMBERS and simple CONDITIONAL JUMPS at the same time!

try...

```
200 RANDOM
210 N = RND(100)-50
220 N = N/10
230 PRINT "WHAT IS THE INTEGER OF ";N,
240 INPUT V
250 M = INT(N)
260 IF V = M THEN PRINT "THAT IS CORRECT":GOTO 210
270 PRINT "THE CORRECT ANSWER IS... ";M
280 GOTO 210
and RUN 200 (ENTER) or (NEW LINE) again...
```

#### EXPLANATIONS

Line 200 reinitializes the RANDOM NUMBER GENERATOR. This is useful in games to ensure that the computer doesn't do the same thing every time a program is loaded and run.

Line 210 makes N a whole number picked at random between 1 and 100 take away 50, so the range is effectively from minus 49 to 50.

Line 250 makes M equal to the INTEGER value of N, which was previously divided by 10 in line 220.

Line 260 make the '80 compare your answer in V with its own answer in M. If they match, the rest of line 260 is executed; if they do not match line 270 follows...

Just for the record,  $N = \text{RND}(1)$  would produce either a 0 or a 1 for N, while  $N = \text{RND}(0)$  would produce an up-to-six-decimal-digit-number between 0 and 1.

You'll soon discover that an INTEGER is the next lowest whole number except when the number is already a whole number in which case it is already an integer.

We can use the INTEGER function to get the NEAREST whole number as well, by using a statement like...

```
300 N = INT(A + .5)
```

Next month we'll get stuck into ARRAYS and that will enable us to convert our miserable little DECIMAL/HEX converter into a full-blown piece of useful programming.

**\*\* DISK DRIVES - QUESTIONS AND ANSWERS \*\***

We have had many questions from readers about disk drives and with so many different drives available, the picture certainly is confusing. This article is an attempt to explain the various terms used and provide answers to the most common questions asked.

Q. What is a mini-floppy disk drive?

A. It is a mass storage device which records information onto a diskette of plastic material coated with a magnetic substance, similar in nature to magnetic tape. The diskette spins around its axis like a record on a turntable and the magnetic head is moved into contact with the surface. There is a pressure pad on the reverse side of the diskette which squeezes the diskette up against the head. Information is recorded in concentric circles called tracks (unlike a record where the information is recorded in spiral grooves). The head is free to move radially over the disk and can be positioned very accurately and repeatedly over the same track. The first disk drives used rigid or hard disks. They were large and expensive and stored vast quantities of data (several megabytes or millions of bytes on one disk-pack). Then the so called floppy-disk drive was developed. It used an 8 inch diameter, flexible diskette which meant that mechanical requirements were much less stringent. It cost much less to produce and was much lighter and more portable than hard disk drives. The trade-off for these advantages are that it takes longer to store and retrieve information from floppy disk drives than from hard disk drives, floppies are less reliable and store much less information per disk.

Subsequently, the Shugart Corporation in America developed a mini-floppy disk drive which is similar to the floppy drive, but uses even smaller diskettes - 5-1/4 inch, stores less data per diskette, storage and retrieval is slower than for a floppy but the price is very much less. It is the mini-floppy disk drive which has become the standard unit used with most microcomputers.

Q. How many tracks does a mini-floppy disk drive have?

A. Two years ago, that was an easy question to answer. The original Shugart drives had 35 tracks and so did all the others. One of the major trends in mini-floppies has been towards more and more tracks on the same sized diskette, quaintly phrased as "higher track counts". Micropolis started the trend by producing a 77 track mini-floppy about 2 years ago. At the time, that was revolutionary because most 8 inch floppy drives had 77 tracks so the mini-floppy was rivalling the floppy at least as far as track-count was concerned. (Floppies still stored more information per track). The only problem with the 77 track mini-floppy drive is that it cannot read a diskette made on a 35 track drive and vice versa. Manufacturers found that they could use a bit more space on the diskette, however, and produced 40 track drives that were completely compatible with 35 track drives but gave an extra  $5/35 = 14\%$  data storage per diskette. A standard 40 track mini floppy can store approximately 100K bytes of information on one side of a 5-1/4 inch diskette. Now that they had proved that they could use more of the surface of the diskette for data storage, the next step was to produce an 80 track mini-floppy. This is a very recent development and the best known manufacturer of these drives is MPI who manufactures the 80 track B91 drive. Unfortunately, the 80 track drives are not compatible with 35, 40 or 77 track drives.

Q. Do I need to buy special diskettes for these drives with different track counts?

A. Hand-in-hand with the development of higher track count drives, the diskette manufacturers improved their diskettes. Within the last few months, most manufacturers have dropped 35 track diskettes completely and replaced them with 40 track diskettes. Bear in mind that a new diskette is completely smooth and unformatted and the tracks are positioned by the particular drive. The difference between a 35 track and a 40 track diskette then, is the radial distance over which the information is recorded. That means not only burnishing the surface over a greater area but also having a magnetic material which will perform at higher recording densities because the higher numbered tracks are closer to the centre of the disk. There is the same amount of data recorded on each track so the short, inner tracks, have higher recording densities (i.e. bits per inch) than the longer, outer tracks.

There are also 77 track diskettes available. These have a usable surface area of the same size as a 35 track diskette but, because the tracks are narrower, they too need to use a magnetic medium which will perform at higher recording densities.

Q. Is there a difference in quality between the various diskettes?

A. Decidedly so but, unfortunately, at least as far as the microcomputer user is concerned, the market has not yet had time to sort itself out and poor quality diskettes are often sold at premium prices. Nor is the difference in quality immediately obvious. Quite apart from the ability of the magnetic medium to perform its function satisfactorily, there is the question of abrasivity. Some of the poorer quality diskettes (like cheap cassettes) are highly abrasive. It could well cost \$150 to have a disk drive head replaced, so this could be a real problem in the future for someone using poor quality diskettes.

Q. How do I tell which diskettes are good quality?

A. That is difficult since, as we said above, price is not yet a good guide. At the top end, where price is noticeable, two manufacturers' products are outstanding. Scotch Brand has been accepted as the standard diskette against which all others are judged, in three countries of the world. DYSAN diskettes, which as yet are little known in Australia, were specially developed for service in mini-floppies where operating temperatures are generally higher than in floppy disk drives. By all accounts, DYSAN are very good diskettes and it is probably significant that Microsoft choose DYSAN diskettes for some of their programs which cannot be copied or "backed-up". There are, of course, many other reputable brands of diskettes including BASF, Verbatim, ODP and Nashua to name just a few. As a rule of thumb, be wary of "own-brand" diskettes. A supplier who sells diskettes under his own-brand name is trying to save money and is likely to be tempted by cheap, low-quality diskettes. Incidentally, we omitted to mention earlier, that the actual magnetic diskette is enclosed in a square black cover lined with a low friction material. The diskette cannot be removed from this cover and spins inside it. There is a window through which the head can contact the magnetic diskette and a cutout on one side which engages a microswitch in the disk drive. If this cutout is covered with a piece of tape then the microswitch senses its presence and refuses to allow the disk drive to record (write) on that disk, which is said to be write-protected. There is also a hole right through the cover and one or more holes are punched in the diskette on the same radius. As the diskette spins, the hole(s) in the diskette line up with the holes in the cover and allows a pulse of light from a light emitting diode (called the Index Sector LED) to strike a phototransistor. The resulting pulse is fed to the computer to signal the start of a track or sector.

Q. What is the difference between hard and soft-sectored diskettes?

A. A soft-sectored diskette has only one hole punched in the diskette so it gives one pulse per revolution. It is up to the computer's operating system to split tracks up into sectors (more of that next issue). A hard sectored diskette has a number of holes punched in the diskette, usually 10, 13 or 16 so that there are multiple pulses each revolution. Each pulse represents the start of a data sector on a track so that the computer's operating system has no control over the number of sectors per track. We will go into this whole subject in more detail in the next issue. Suffice to say here that the TRS-80 uses soft sectored diskettes.

Q. What is a flippy-floppy drive?

A. Ugh! A flippy-floppy drive is one in which you can turn the diskette over and use its reverse side. The actual differences between a flippy and a floppy (!?!) is that the flippy-floppy drive has a second microswitch to sense the write-protect notch and a second index sector light-emitting-diode and phototransistor pair to sense the index sector holes when the diskette is reversed. The best known brand of flippy-floppy drive is the Pertec, sold in Australia by Dick Smith Electronics. There are two schools of thought on the reversing of diskettes in a flippy-floppy drive. One says it is OK. The other is concerned that the felt pressure pad is wiping across data. If that pad collects grit, it could scratch the diskette surface and all would be lost, irretrievably. We have personally used diskettes in this way for the best part of a year now, with no problems. However, we tend to use a large number of diskettes a few times. If you use a small number of diskettes often, then you might need to be wary of flippy-floppy drives.

Q. What is a dual-head drive?

A. A dual-head drive has a read-write head on each side of the diskette. The heads move on the same radial carriage but only one is active at a time. A dual-head drive uses both sides of the disk but, because there are no felt pressure pads, only ceramic heads, the objection to the flippy-floppy drive does not apply to dual head drives. These drives are an attractive proposition to anyone on a tight budget because the shared electronics and power supply means that a dual-head drive costs only about one third more than the two-single head drives to which it is almost equivalent. The only real disadvantage of a dual head drive is that, in order to make a copy of valuable data for safekeeping, you need to swap diskettes during the back-up process, since there is not much point in copying from one side to the other if you want to keep the two copies separated. Most people consider this minor inconvenience is much more than offset by the real cost advantages of the dual-head drive. Dual-head drives are now readily available with 40 tracks on each side (e.g. MPI B52) and are available in limited quantities with 80 tracks on each side (e.g. MPI B92).

Q. I have seen single sided/single density diskettes advertised and double sided/double density diskettes but not double sided/single density. Can I use a double sided/double density diskette on a double sided/single density drive?

A. Phew! Yes. We will discuss the difference between single and double density drives next issue but the standard TRS-80 Model I is designed to use single density drives. Double density diskettes will perform at recording densities twice those required by the TRS-80, so they work extremely well on single density systems. In order to limit their product range and keep inventory levels down, diskette manufacturers simply do not bother to produce double sided/single density diskettes.

Q. Is one disk drive sufficient or must I buy two?

A. That depends on you. Two drives are much more convenient to use than one and most people soon want a second drive. The reason is that the disk-operating system and DISK BASIC are stored on a diskette, called the SYSTEM diskette. This diskette must be in position on drive 0 at all times so that these programs can be called up by the computer. They take up a considerable amount of space on the diskette, about 40% of a 35 track diskette so you do not have a great deal of space available for your own programs, data etc. If you have a second drive, you can put a data diskette on the second drive which has about 80% available space for your programs and data. It is also much easier and faster to copy programs from one drive to another than it is from one diskette to another, using the same drive.

That's all this issue. It is a huge subject and we will tackle some of the more technical aspects of disk drives next month.

- 0000000000 -

\*\*\*\*\* READERS' REQUESTS \*\*\*\*\*

This column is a regular feature of MICRO-80. In it, we list all those articles, programs, etc. requested by our readers. We invite contributions from readers to satisfy these requests and will, of course, pay a publication fee for all articles, programs, etc. printed.

\*\* ARTICLES \*\*

- File handling on the '80
- Description of the functions performed by the Expansion -Interface
- Reviews of '80 compatible printers
- Reviews of commercially available software (including that produced by us!)
- Reviews of commercially available hardware
- How to SAVE onto Disk, programs such as Analogue clock and Touchtype
- A master index to the appropriate sections in the Tandy Manuals in Level I, Level II, DOS etc.
- Comparative review of disk drives
- How to convert a Level I program to Level II
- A simple guide to using Level I Arrays
- Review of Dunjonquest program
- An explanation of how to make full use of USR, PEEK & POKE statements.

\*\* SOFTWARE \*\*

- GAME OF LIFE relocated to start at 7000H
- A m/1 program to enable the BREAK key to work like RESET when using an expansion interface
- Stock market program
- Horse racing system
- Sub-routine forum
- Program to "SET" non-graphical symbols
- A new STAR-TREK game
- Conversational programs (like ELIZA)
- 3D programs (such as a maze seen from the inside)

\*\* HARDWARE \*\*

- RS 232 printer interface
- Interfacing the '80 to external hardware
- Review on the performance of line filters
- How to convert a black and white TV to be a monitor
- Review of high resolution graphics mod
- Real time clock
- Radio Teletype/Morse interfacing
- RFI (Radio Frequency Interference) suppression
- Interface for a Teletype printer

Notes:-

- 1) Reader Bill Blunder has pointed out that the December 1976 edition of Australian ETI carried an article on modifying TV sets to use as monitors.
- 2) Electronics Australia for November has an article describing how to construct a simple RS 232 interface for \$15.00.
- 3) We already have an article in hand describing how to save TOUCHTYPE to Disk. A master index to Tandy manuals is also nearing completion.

- 0000000000 -

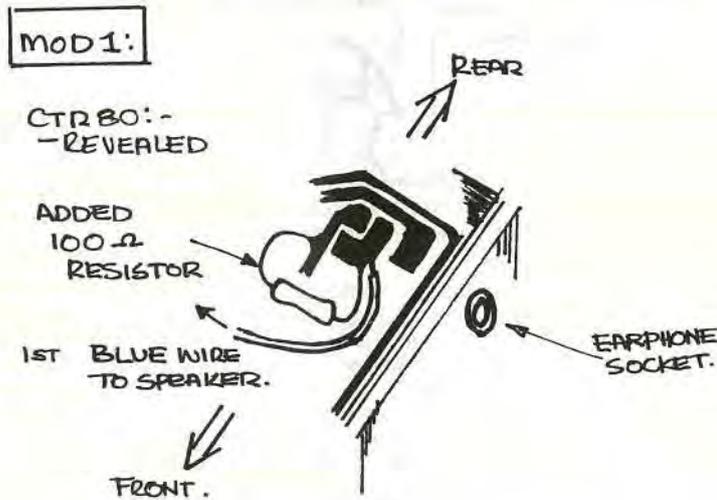
\*\*\*\*\* INPUT/OUTPUT \*\*\*\*\*

FROM: Shirley and Peter Miller, - Stokes Valley, N.Z.  
 1) Modifications for the CTR-80

I have noticed a few friends (with the TRS-80/CTR-80) building "Black Boxes" to let them hear what's going on. All these attempts have (to some degree) affected the performance of this storage system. The modifications outlined below will not degrade the signal nor introduce noise and would suit anyone who intends dedicating the CTR-80 to the computer.

\*\* MOD 1 \*\*

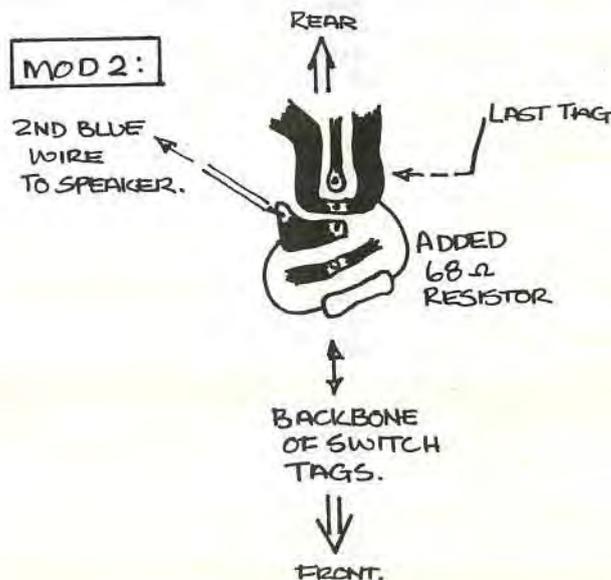
To hear a "load"; a 100 ohm resistor is placed across the two rear solder tags of the earphone socket.



\*\* MOD 2 \*\*

To hear a 'save' - and the sound on some of the latest games; a 68 ohm resistor is soldered across internal switch tags.

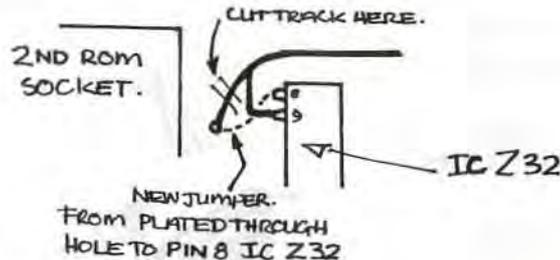
Method: Remove the jacks etc. and free the recorder from the computer. Find a clear spot and place a soft cloth on your work area. Invert the recorder, remove the batteries and using a small Phillips screwdriver - remove the 3 small screws. Swing the back clear to give a full view of the circuit board. Gently remove any insulated tape from the area we intend to work on - it has to be replaced when you have finished. Perform the simple additions. Reassemble after a thorough check of your soldering - ensure you have not joined any adjacent tracks.



## 2) Modifications to the Video.

We all realise that Radio Shack have saved a few cents in this department. My system was worse than most - a 60 Hertz model running on a 50 Hertz main. The modification outlined below also applies to those who have a frequency match but are still dissatisfied with the display. The crystal (available ex-stock from Rakon Industries Ltd. Auckland New Zealand) used is 10.483M Hertz. This replaces the Tandy 10.6445M Hertz.

I do not recommend this modification to anyone who has not dabbled in electronics and still wishes to void their warranty. However, the results have to be seen to be believed. No in-depth instructions will be given - to repress the temptation in those who may try to over-extend themselves.



## MODIFICATION TO THE TRS-80 VIDEO TIMING CHAIN

- (1) REMOVE THE TANDY CRYSTAL <10.6445M HERTZ>
- (2) REPLACE IT WITH A 10.483M HERTZ ONE
- (3) LOCATE IC 32 ; CUT TRACK AS INDICATED BELOW
- (4) ADD JUMPER AS SHOWN

## 3) Lower case Driver (MICRO 80 Issue 5).

Your Issue 9 published a reader's letter giving a solution to the error - with an error trapping routine. My solution is included in the listing below. We all learnt something about integers with this one.

```

20 POKE 16553,255
30 FOR I%=0 TO 29
40 READ D%
50 POKE (32738+I%),D%
60 NEXT
70 POKE 16415,127:POKE 16414,226
80 DATA 221,110,3,221,102,4,218,154,4,221,126,5,183,40,1,119
90 DATA 121,254,32,218,6,5,254,128,210,166,4,195,125,4
100 END

```

(Thank you Shirley and Peter for these useful contributions. We did describe similar modifications to the CTR 41 and CTR 80 cassette recorders in earlier editions of MICRO-80 - Issues 1 and 2 but they certainly bear repeating. We have heard good things about the video mod you describe but have not yet tried it ourselves. Incidentally, there is a suggestion that the Disk Drive Timing program DDT-1 will not work on a CPU modified as you describe, due to the change in clock frequency. Once again, we have no direct experience and would be interested to hear from any reader who has. - Ed)

From: John S. Bone, - Sec'y, Newcastle Personal Computer Society, U.K.  
 I have just finished reading your September MICRO-80, and after showing it to several fellow '80 users, I wish to congratulate you upon your fine magazine. I am the Hon. Secretary for the N.P.C.S., (i.e. UN-paid!!!) and will place a subscription for MICRO-80 with your U.K. distributor as soon as my committee "O.K.s it".  
 We have 40 or so TRS-80 users and 10 plus Video Genie users in our 130 plus membership through the north-east of England and we feel your mag is in a "slot" of its own in the computer mag market, with only 80-U.S. or MICROCOMPUTING - 80 from the U.S.A. to compete with it. They, of course, have a large amount of adverts, we are pleased to see your min. ads. policy, and hope you continue to prosper.

I would like to see articles upon any AUSTRALIAN EDUCATIONAL USE/POLICY OF TRS-80's etc. and any SCHOOL USERS' experiences/applications by your members.  
 The N.P.C.S. caters for both those "just interested" and "The XYZ system USER", by means of monthly open meetings/personal micro demo's, at our local polytechnic college and more frequent sub meetings at user group level, (incl. '80, S100, PET, NASCOM, SINCLAIR ZX80, RSM Z80, ad etc.) (THESE SUB MEETINGS ARE OFTEN OF THE "GOSUB 'PUB type"").

(Thank you for your kind comments, John. We have included the N.P.C.S. in our '80 Users Groups and your request for articles concerning educational applications of TRS-80's in our Readers' Requests column. It is our impression that, to date, there is more experience of the use of microcomputers in education in the U.K. than in Australia. Perhaps some of our U.K. readers would like to prepare an article or two for us. (If that doesn't stir Australian educators into action, nothing will!) - Ed)

From: D.G. Johnston, - Herts. U.K.  
 A publication like MICRO-80 would seem to fill a need, as most magazines here seem to deal mainly with PET and the APPLE.

I have a Level II 16K TRS-80 and feel that to the less knowledgeable among us that an article that "converts" programs from other BASIC's and also from Disk BASIC to avoid L3 errors, would be helpful.

(We have placed your suggestions in the Readers' Requests section, Mr. Johnston. The main problem in converting programs written for one type of microcomputer to run on a different type is not so much in converting the BASIC code but rather in converting the screen formatting and graphics. If you look at most games programs, for example, you will notice that typically only 25% of the code is concerned with the program logic. The remainder is concerned with graphics and screen displays. Unfortunately, the three computers you mention, PET, APPLE and TRS-80 all have completely different graphics capabilities and different ways of formatting the screen, in BASIC. To convert such a program could well be a bigger task than writing it from scratch!

Other programs which do not make much use of graphics, however, are readily converted. We intend to publish useful sub-routines as they are sent in and readers might like to consider sending us routines to enable the TRS-80 to emulate commands and statements which are available in BASIC on other machines but not on the TRS-80.

Similar remarks apply to DISK BASIC commands for Level II machines - Ed)

INPUT/OUTPUT continued bottom of Page 12.

- 00000000000 -

\*\*\*\*\* '80 USERS' GROUPS \*\*\*\*\*

The following is a list of '80 Users' Groups. If you have a group that is not included here, please let us know about it so that we can publish details. Owners of System '80s are welcome at all the groups.

\*\* AUSTRALIA \*\*

BRISBANE: Contact: Mr. Lance Lawes,  
 Tel: Home (07)396 2998  
 Bus.(07)268 1191 Ext.15  
 MEETINGS 1st Sunday of the month at 2 p.m. at 21 Rodney St. Lindum, 4178.

MELBOURNE: EASTERN SUBURBS - 1  
 Contact: Mr. John Fletcher, 89 0677 bet. 9-4

EASTERN SUBURBS - 2  
 MEETINGS: 3rd Wednesday of the month at Kingswood College, 355 Station St. Box Hill.

FRANKSTON: PENINSULAR GROUP  
 (Vic.) MEETINGS: 2nd Tues. of the month (except Jan.)  
 Contact: M.G. Thompson (03)772 2674

GEELONG: \*GEELONG COMPUTER CLUB  
 MEETINGS: 2nd Tues. of the month at TYBAR Engineering, Hampton St. Newtown.  
 Contact: The Geelong Computer Club,  
 P.O. Box 6, Geelong, 3220.

DARWIN: Contact: Tony Domigan, P.O. Box 39086,  
 Winnellie, N.T. 5789.

ADELAIDE: Contact: Rod Stevenson, 51 5241 bet. 9-4

CANBERRA: Contact: Bill Cushing, 10 Urambi Village, Kambah ACT 2902 (Ph. 31 6399)  
 MEETINGS: 3rd Thursday of each month at 7.30 pm. in:-  
 Urambi Village Community Centre, Crozier Circuit, Kambah.

\*\* UNITED KINGDOM \*\*

NEWCASTLE Contact: John Stephen Bone 0632 770036  
 NPCS (Newcastle Personal Computer Society) - see INPUT/OUTPUT

\*\* NEW ZEALAND \*\*

AUCKLAND Contact: Ron Feasy 799 366 (Bus.) 469 455 (Priv.)  
 MEETINGS: 1st Tuesday of each month, 7.30 pm. at:-  
 NZ Solenoid Co. Ltd.,  
 28 Kalmia Street,  
 Ellerslie, Auckland.

- 00000000000 -

## \*\*\*\*\* MICROBUGS \*\*\*\*\*

## \*\* SYSCOPY \*\*

It has been pointed out to us that the SYSCOPY program published last month cannot be entered into the machine using BMON because BMON does not appear to be able to enter the last few bytes. Eddy Paay has looked at the program and determined that SYSCOPY is too close to BMON and, in fact, resides partially in the stack area which BMON uses to store the BASIC registers etc. There are two answers. The first, for those with an Editor/Assembler, is to locate SYSCOPY further away from BMON. Leave at least 100 bytes clear. The second is to change BMON so that the stack area is below SYSCOPY. There was not sufficient time available to work out these changes this month so we will publish them next month.

## \*\* RESISTOR COLOUR - CODE DECODER \*\*

The original program, published last month, did not cater for fractional values where the third band may be gold or silver. Nor did it allow for the two extra tolerance colours for 1% and 2% resistors. We have received an amended version of the program from Mr. Khavounitis adding these features which we will publish next month.

- 0000000000 -

## \*\* KB FIX \*\*

The gremlins continue to dog our efforts to publish a KB FIX program in BASIC. This month we have the program but it has become separated from its original letter so we are unsure of the author. Nevertheless, we will publish it and record our thanks to its now anonymous creator.

This is a short program which self-destructs in the twinkling of an eye, leaving behind it a legacy of non-bouncing keys. It could, of course, be renumbered and included in all your BASIC programs to save reloading it every time you use it.

```
10 REM KEYBOARD DEBOUNCE ROUTINE
20 REM NO NEED TO ENTER MEMORY SIZE
30 REM PROGRAM ERASES ON COMPLETION
40 FOR I=32744 TO 32766:READ J:POKE I,J:NEXT
50 POKE 16526,234:POKE 16527,127:POKE 16561,241
   POKE 16562,127:POKE 16598,241:POKE 16599,127
60 X=USR(N)
70 NEW
```

- 0000000000 -

INPUT/OUTPUT - continued from Page 11

From: J. Pinn, - Wembley Downs, W.A.

One point which may be of interest to you is at I went to my nearest Tandy Store and asked if he could offer similar discounts to CONQUEST ELECTRONICS. He said that he had never heard of a Tandy store offering discounts as they are not permitted to do so. He also said that if I sent to CONQUEST ELECTRONICS that the warranty would not stand. Is this true?

(CONQUEST ELECTRONICS is a Tandy Authorised Dealer not a Tandy Electronics Store. Tandy Authorised dealers are independent organisations who are authorised to sell Tandy goods. A Tandy Electronics Store is owned by Tandy Electronics Australia Pty. Ltd. Store managers and staff are salaried employees of Tandy Electronics. Normally, Authorised Dealers are only appointed in country towns where there are no Tandy stores. Goods sold by Tandy Authorised Dealers carry Tandy's much advertised world-wide warranty just as if they had been bought from a Tandy Electronics store. Fortunately, enlightened consumer legislation in most Australian States makes it illegal for a supplier to control the retail selling price of his goods or to withhold supplies from a retailer who offers discounts. So your Tandy Electronics store manager was incorrect on both counts.

It is interesting to note that the American magazines are full of discount offers by Tandy Authorised Dealers and, reputedly, one of these dealers who concentrates mainly on mail order is the largest single seller of TRS-80's, bar none. - Ed)

From: Mr. Stan Jaques, - West Pennant Hills, NSW

I am having trouble with using the BMON program. The program works very well but when I select the 'C' comand - COPY MEMORY TO TAPE - the screen displays "START, END, ENTRY, NAME. Here you have lost me. The instructions tell me to type in the correct values under each but, from where do I get these values. I am very interested in copying some existing machine language tapes

(START is the starting address in HEX of the machine language program you wish to copy. END is its ending address in HEX whilst ENTRY is the address (in HEX) to which the microprocessor must jump to run the program. NAME is any six letters you choose. If you type in a program from MICRO-80 for example, all these parameters will be specified. If you only have a machine language tape, however, you have a problem. The program we published last month, SYSCOPY, is designed to overcome that problem so we suggest you enter that one first but see this month's and next month's MICROBUGS. - Ed)

- 0000000000 -

## \*\*\*\*\* SOFTWARE SECTION \*\*\*\*\*

## \*\* LEVEL I PROGRAMS \*\*

FIGHTER SQUADRON by Br. P. Van EEKEN ... 4K/L1  
 This is an example of mixing education with a game. You are a squadron leader who has between 2 and 4 planes to destroy GESTAPO H.Q. Before the mission, you have to calculate your arrival time to the target. If, after 5 tries you haven't got the right answer, it will be displayed and the game part of this program starts. If you get the right answer the game starts straight away. GESTAPO H.Q. is drawn on the screen with four guns to demolish. Then your first plane starts its bombing run and continues until it is shot down.

Line 77 tests the number of guesses. If less than 5, the program jumps to line 85 which checks your answer. The subroutine for drawing GESTAPO H.Q. starts at 2000. Line 320 & 330 are not the results of bad typing, 320 IF(P.(N,D)) etc. is the same as IF (POINT(N,D)=1) etc. and 330 - IFP.(N,D)GOS.750:G.400 is the same as IF POINT(N,D)=1 GOSUB750:GOTO400, this is a space saving trick which comes in handy if you only have 4K of memory and you are trying to squeeze the last little bit (in this case byte) out of your machine.

```

1 R=0:G.47
2 F.X=AT.B:S.(X,C):N.X:RET.
3 F.X=CT.D:S.(B,X):N.X:RET.
10 REM BY BR. P. VAN EEKEN
47 IFZ=2G.50
48 D=R.(9)+3:S=R.(50):A=D*S:J=R.(3)+1
50 C.:P."SQUADRON LEADER:":P.:B$=GESTAPO H.Q.
51 P."YOU HAVE";J;"PLANES TO DESTROY ";B$;" WITH.":P.
52 P."YOU MUST HIT ALL THE ANTI-AIRCRAFT GUNS DEAD CENTRE":P.
53 P."TO DESTROY ";B$:P.:Z=2
70 P.A.576;"TO START YOUR MISSION YOU HAVE TO CORRECTLY CALCULATE"
75 P."THE HOURS IT WILL TAKE YOU TO ARRIVE AT YOUR TARGET";A
76 P."KILOMETRES AWAY, TRAVELLING AT";D;:I."KM PER HOUR";P
77 R=R+1:IFR<5G.85
80 P.A.896;"THE CORRECT ANSWER IS";S;:F.R=1T01000:N.R:G.100
85 IFP=5G.100
90 P.A.896;"TRY AGAIN.":A.704:G.70
100 F.X=1T.4:A(X)=0:N.X:GOS.2000
101 R=0:S=0:W=1
105 K=0
110 E=0:GOS.791
115 IFK=0I=R.(12):IFI<4G.115
120 IF(K<9)+(K>100)Z=1
130 IFK>0Z=2
140 IFK>29Z=3
150 IFK>66Z=4
151 IFA(Z)=1Z=Z+1:IFZ=5Z=1:IFE>50G.4000
152 IFA(Z)=1E=E+1:G.151
155 E=0:REST.
160 F.X=1T.Z:REA.A,B:N.X
170 GOS.900
200 F.H=1T.40:P.A.841;
205 IFK>127R.(K-6,I-1):F.X=K-6T.K:R.(X,I):N.X:K=0
220 IFK=0S.(K,I-1):F.K=0T.6:S.(K,I):N.K
230 IFK=7R.(0,I-1):R.(0,I):S.(K,I)
240 R.(K-6,I-1):R.(K-6,I):S.(K-5,I-1):K=K+1:S.(K,I)
245 IFW=0G.300
260 IFP.(61,40)=0T=-1:GOS.2000:G.105
261 IFP.(20,39)=0G.290
270 Z=R.(2):IFZ=2G.240
280 G.400
290 P.A.843;"\\ BOMBS AWAY \\":A.0
291 S.(20,39):W=0:N=K-3:0=I+2:S.(N,I+1)
300 F.D=0T.0+3
310 IFP.(N,D)=0G.350
311 REST.:F.X=1T.4:REA.P,Q
315 IF(N=P+1)*(D=Q+1)GOS.700:G.400
316 N.X
320 IF(P.(N,D))*(N=A)+(N=A+2)GOS.780:G.400
330 IFP.(N,D)GOS.750:G.400
350 R.(N,D-1):S.(N,D):N.D:0=0+4

```

```

400 IFH/2=I.(H/2)G.820
410 IFB-H-4<0R.(A,B-H-1):R.(A+2,B-H-2):G.850
420 IF(B-H-3<>I)*(B-H-4<>I)G.815
430 F.F=K-6T.K:IF(A=F)+(A+2=F)G.440
435 N.F:G.815
440 R.(A,B-H-1):R.(A+2,B-H-2)
441 R.(N,D-1):GOS.790
445 F.C=1T.5
450 R.(K-6,I-1):R.(K-6,I)
460 F.F=K-5T.K+1:S.(F,I):R.(F,I):N.F
465 N.C
469 A$=PLANES
470 R=R+1:P.A.878;R::IFR=1A$=PLANE
475 P.A$;" LOST";
480 H=0:IFR<JG.105
490 P.A.0;"THE GERMAN REICH HAS ANNIHILATED YOUR SQUADRON !"
500 P.:P.:P.T.(20);"HIEL HITLER !":P.
510 I."LIKE TO TRY AGAIN (Y/N) ";S:IFS<>NG.1
520 C.:E.
699 G.105
700 R.(P,Q-1):R.(P,Q):R.(P+2,Q-2):R.(P+2,Q-1):R.(P+2,Q)
701 A(X)=1:R.(N,D-1)
702 P.A.43;"DIRECT HIT !"
709 F.D=1T.3
710 F.X=P-2T.P+4:F.Y=Q+1T.Q+2
715 S.(X,Y):F.E=1T.20:N.E:R.(X,Y):N.Y:N.X:N.D
719 A$=GUNS
720 S=S+1:P.A.940;S::IFS=1A$=GUN
721 R.(P,Q-2)
725 P.A$;" DEMOLISHED*";:G.790
750 P.A.43;"MINOR DAMAGE !";:G.790
780 IFN>11P.A.43;"THAT TICKLES !":G.785
781 F.E=1T.5
782 F.C=D-1T.D+1:F.G=N-2T.N+2:S.(G,C):R.(G,C):N.G:N.C
784 N.E
785 F.U=0T.3:R.(A,B-H-U):R.(A+2,B-H-U):N.U:H=0
790 W=1:R.(N,D-1)
791 S.(20,39):P.A.843;"PRESS <B> TO BOMB*";:P.A.841::RET.
815 S.(A,B-H-3):R.(A,B-H-1)
816 S.(A+2,B-H-4):R.(A+2,B-H-2)
820 N.H
850 P.A.40;":G.110
900 F.G=1T.2
910 F.E=B-2T.B:IFE=BG.920
915 R.(A,E)
920 N.E
930 F.E=BT.B-2S.-1:S.(A,E):N.E
940 F.E=B-2T.B:IFE=BG.950
945 R.(A+2,E)
950 N.E
960 F.E=BT.B-2S.-1:S.(A+2,E):N.E
970 N.G:RET.
2000 C.
2001 IFT=-1GOS.790:T=0
2010 A=0:B=20:C=30:GOS.2
2020 D=40:GOS.3
2030 A=21:B=30:C=35:GOS.2
2040 D=35:C=25:GOS.3
2050 A=31:B=61:GOS.2
2060 D=40:GOS.3
2070 A=62:B=90:C=40:GOS.2
2080 F.X=90T.110:S.(X,C):C=C-1:N.X
2090 A=111:B=127:GOS.2
2092 P.A.913;B$::P.A.841;
2100 C=30:D=47:B=0:GOS.3
2110 B=127:C=20:GOS.3
2120 A=0:C=47:GOS.2
2130 A=37:B=43:C=31:GOS.2
2140 D=38:B=44:GOS.3
2150 A=B:B=51:C=38:GOS.2
2160 A=37:C=35:GOS.2
2170 B=37:GOS.3
2180 C=31:D=35:B=51:GOS.3

```

```

2190 F.X=7T.13:S.(X,Z):N.X
2210 Y=24:F.X=6T.0S.-1
2220 S.(X,Y):Y=Y+1:N.X
2230 Y=24:F.X=14T.20:S.(X,Y):Y=Y+1:N.X
2231 REST.:F.D=1T.4:REA.P,Q:IFA(D)=1G.2234
2232 F.X=P-2T.P+4:F.Y=Q+1T.Q+2:S.(X,Y):N.Y:N.X
2233 A=P:B=Q:GOS.900
2234 N.D
2999 D.9,20,38,22,83,37,114,16
3000 P.A.0;
3001 RET.
4000 F.F=37T.43:R.(F,31):N.F
4001 F.F=31T.38:R.(44,F):N.F
4002 F.F=44T.51:R.(F,38):N.F
4003 F.F=37T.51:R.(F,35):N.F
4005 F.F=31T.35:R.(51,F):N.F
4006 F.F=35T.38:R.(37,F):N.F:A$="
4007 P.A.913;A$;:P.A.843;A$;
4008 P.A.0;B$;" IS NOW INOPERATIVE..... WELL DONE !
4009 P.:G.510

```

- 0000000000 -

#### FOOD REQUIREMENTS

by Br. P. Van Eeken ... 4K/LI

You'll be able to find out how much you are eating each day (in Kilojoules) with this program. You'll also get a breakdown of the percentage of carbohydrates, protein, fats and fibre in your diet. All you have to do is enter the amount you eat in one day, from each of the five basic food groups. If you get ambitious, try adding more items to the various food groups (with their appropriate KJ value), as well as a section for calculating consumption against height, weight and amount of activity you have in a day. Have a go - you may find that you're a better programmer than you thought.

#### DAILY FOOD REQUIREMENTS BY - BR. P. VAN EEKEN

COPYRIGHT - MICRO-80, NOVEMBER 1980  
PO BOX GOODWOOD, SOUTH AUSTRALIA 5034

```

100 T=0:F=0:K=0:N=37:U=0:H=-22:REST.
110 F.X=1T.N:REA.B$:N.X
120 F.X=N+1T.2*N:REA.A(X):N.X:REST.
130 GOS.800:B$=EACH
140 P=5:GOS.960:B$=CUPS
151 GOS.960:GOS.800:P=3:GOS.960
180 GOS.800:P=2:GOS.960:B$=SLICES
191 P=1:GOS.960:B$=EACH
193 P=3:GOS.960:GOS.800:P=3:GOS.960:B$=SLICES
197 P=2:GOS.960:B$=PIECES
202 P=3:GOS.960:B$=TABLESPOONS
210 P=1:GOS.960:GOS.800:P=3:GOS.960:GOS.800
230 P.A.576;"EXTRAS":P."-----"
235 P=3:GOS.960:B$=EACH/SQUARES
241 P=1:GOS.960:B$=GLASSES.
243 GOS.960:B$=SERVES
245 GOS.960:G=T-(A+B+C+D+E)
260 W=100:R=I.((C+G)/T*W+.5):O=I.(A/T*W+.5)
265 Q=I.((B/2+D*.67)/T*W+.5):P=W-(R+O+0):C.:GOS.800
280 P.:P.A.512;"YOUR FOOD INTAKE TOTALS";T;"KILOJOULES.
285 P.T.(24);"-----";T.(50);"NORMAL %
290 P."OF THIS TOTAL...";:B$=CARBOHYDRATES
292 K=50:Z=R:GOS.950:B$=PROTEINS
296 K=20:Z=Q:GOS.950:B$=FATS
301 Z=P:GOS.950:B$=FIBRE NEEDS
306 Z=0:K=10:GOS.950:GOS.995:K=0:J=1:REST.:C.
320 P."YOUR EATING HABITS ":IFO>9T.340
331 U=U+1:H=-22:GOS.970:Z=1:P."FRUIT & VEGE' S
340 M=10:GOS.700:IFZ<>1T.345
345 Z=0:M=3:GOS.700
350 IFR<51T.360
351 P.:Z=1:J=0:H=-22:P."TOO MUCH CARBOHYDRATE FOOD...

```

```

352 M=6:GOS.700
353 Z=0:M=15:GOS.700:Z=1:M=6:GOS.700:REST.
354 Z=0:M=13:GOS.700:U=U+1
360 M=6:GOS.700:IFQ>20T.300
371 P.:GOS.970:J=1:Z=1:U=U+1:H=-22:P.*PROTEIN FOODS.
380 M=9:GOS.700:IFZ<>1T.390
390 Z=0:IFP<26T.400
395 P.:A=1:J=0:H=-22:P.*TOO MUCH FATTY FOOD & MEATS...
396 Z=1:U=U+1
400 M=3:GOS.700
401 IFZ<>1T.410
410 IFT>9000T.450
420 B$=LITTLE.
421 J=3
450 IFT<12000T.460
451 B$=MUCH.
452 J=4
460 IFJ<3T.470
461 P.:P.:H=-22:P.*IN GENERAL, YOU ARE POSSIBLY EATING TOO *:B$
462 REST.:K=0:IFJ<>4T.470
463 Z=0:J=0:M=13:GOS.700:Z=1:M=6:GOS.700:M=9:Z=0:GOS.700
464 M=9:Z=1:GOS.700:G.490
470 IFU<>0T.490
480 C.:P.A.460;*YOU HAVE A WELL BALANCED INTAKE OF FOOD.
490 GOS.995:G.100
600 P.A.720;*FOOD*;T.(35);*AMOUNT*;T.(50);*YOUR ANSWER
610 P.T.(16);*-----*;T.(35);*-----*;T.(50);*-----
615 K=K+1
620 P.T.(16);A$;T.(35);B$;T.(55);:I.A(K)
621 T=T+A(K)*A(K+N):IFK<>10T.630
625 A=T
630 IFK=13B=T-A
640 IFK=19C=T-(A+B)
650 IFK=28D=T-(A+B+C)
660 IFK=31E=T-(A+B+C+D)
699 RET.
700 F.X=1T.M:REA.A$;K=K+1
710 IFZ<>1T.750
711 IFJ=1T.720
712 IFA(K)<2T.750
714 IFK<>16T.730
715 IFA(K)<4T.750
716 G.730
720 IFA(K)>0T.750
730 H=H+22:IFH>44H=0:P.
740 P.T.(H),A$;
750 N.X:RET.
800 C.:P.*DAILY FOOD REQUIREMENTS*:P.*GROUP...
810 P.*1. FRUIT AND VEGETABLES - 4 TO 5 SERVES.
815 P.*2. MILK - 2 TO 3 CUPS.
820 P.*3. BREAD AND CEREALS - 3 OR MORE SERVES.
825 P.*4. PROTEIN (MEAT, EGGS, FISH & CHEESE) - 2 SERVES.
830 P.*5. FATS & OILS - 2 TABLESPOONS.
831 P.:P.*WHAT DO YOU EAT IN ONE DAY:
832 P.:F=F+1:P.A.576;*GROUP*;F:P.*-----
835 RET.
900 D.APPLES,BANANAS,ORANGES,CARROTS,TOMATOES,POTATOES,CABBAGE
901 D.PEAS AND BEANS,OTHER VEGETABLES,OTHER FRUITS,MILK,YOGHURT
902 D.ICECREAM,BREAKFAST CEREAL,SPAGETTI/PASTA,BREAD
903 D.BISCUITS - DRY,BISCUITS - SWEET,CAKES,EGGS,SAUSAGES
904 D.MEAT PIE,BACON,LAMB,BEEF OR STEAK,CHICKEN,FISH,CHEESE
905 D.MARGARINE,BUTTER,CREAM,SUGAR,GRAVY OR SAUCES
906 D.JAMS AND SPREADS,LOLLIES / CHOC.,SOFT DRINK,DESSERTS
920 D.300,364,250,100,100,550,110,200,200,190
921 D.645,600,600
922 D.435,716,250,100,285,500
923 D.450,500,2160,500,600,1800,1100,700,300
924 D.570,578,306
925 D.260,300,325,100,500,1000
950 P.T.(20);B$;* OCCUPY*;Z;*%*;T.(55);K:RET.
960 F.X=1T.P:REA.A$:GOS.600:N.X:RET.
970 P.*NOT ENOUGH...*:RET.
995 P.A.960;*PRESS ENTER *:I.B$:RET.
999 E.

```

\*\*\*\*\* NEW FROM MICRO-80 PRODUCTS \*\*\*\*\*

We have a host of new products and software for TRS-80/SYSTEM 80 microcomputers.

MICROTEK MEMORY EXPANSION/PRINTER MODULES

A number of our readers expressed interest in the memory expansion module we developed earlier in the year. Unfortunately, there was not sufficient interest to make it worthwhile continuing with this rather expensive project, particularly as a completely assembled, tested unit would have had to sell for more than \$200.

Fortunately, MICROTEK in America had the same idea and with the larger population and greater resources, has been able to produce a similar unit which we can sell for only \$149. We are disappointed that we were unable to produce our own design economically but are happy to offer you the MICROTEK unit. It comes fully assembled and tested in a metal cabinet of similar size to the Tandy expansion interface so that you can sit your monitor atop it. The MICROTEK unit intercepts the Tandy power supply cable and has its own internal rectifiers and regulators. As well as providing sockets for 32K of memory, it also has a Centronics parallel printer port which enables you to run a line printer or the Olivetti ET-121. Full instructions for connecting to your system and adding memory chips are included.

MICROTEK MT32-A	OK	\$149
MICROTEK MT32-B	16K	\$208
MICROTEK MT32-C	32K	\$262

Incl.p&p. The memory supplied is our prime, branded 200 nanosecond kit. Fitting is included in the price if memory is ordered with the module.

\*\*\*\*\* SYSPAND-80 FOR SYSTEM 80 COMPUTERS \*\*\*\*\*  
\$119 incl p&p.

The SYSPAND-80 adaptor designed by MICRO-80 PRODUCTS is now available in fully assembled and tested form. Kits and kit prices will be announced next month. The SYSPAND-80 connects to the rear of the SYSTEM-80 and provides a Centronics parallel printer port plus the Tandy bus so that you can run a Stringy Floppy, the MICROTEK memory expansion module or even a Tandy expansion interface/disk drives etc. from your SYSTEM-80. The SYSPAND-80 has its own plug-pack power supply and internal regulator so it imposes no additional load on the SYSTEM 80 power supply.

40 PIN EDGE CONNECTORS \$12.30 +50c. p&p

These difficult to obtain connectors are essential if you are working on a hardware project requiring connection to the TANDY bus. The connector attaches to a 40 way ribbon cable and plugs straight on to the back of your TRS-80.

\*\*\*\*\* GREEN SCREEN SIMULATOR \*\*\*\*\*

\$19.95 incl. p&p

The GREEN SCREEN SIMULATOR is made from a deep green perspex, cut to fit your monitor. It improves contrast and is much more restful to the eyes than the normal grey and white image. All the editorial staff of MICRO-80 (including Scrooge Mc. Hartley) are now using GREEN SCREEN SIMULATORS on their own monitors. Please make sure to specify whether you have an old (squarish) or new (rounded) style monitor when ordering. Not available for Dick Smith monitors.

#####  
OLIVETTI ET-121 DAISY WHEEL TYPEWRITER FOR '80  
only \$1995

#####  
MICRO-80 PRODUCTS has developed an interface to drive the OLIVETTI ET-121 typewriter from a Centronics parallel printer port. The ET-121 is a quiet, high speed (20 cps) auto-correcting, electronic typewriter utilising a Z 80 microprocessor. We have mounted an interface inside the typewriter which does not interfere with manual operation but enables it to be driven as a printer from a Centronics port such as is available in the SYSPAND-80, MICROTEK module or TRS-80 expansion interface.

This makes an ideal combination for the home or office where you need the flexibility of a quiet, reliable, superior typewriter which can double as a correspondence quality printer when driven from your microcomputer.

This issue of MICRO-80 has been typeset using the MICRO-80 OLIVETTI.

Deliveries commence in December. Secure your order by sending a \$50 deposit or write asking for further technical details.

DATA SEPARATOR \$37.00 plus \$1.20 p&p.

When Tandy designed the TRS-80 expansion interface, they did not include a data separator in the disk-controller circuitry, despite the I.C. manufacturer's recommendations to do so. The result is that many disk drive owners suffer a lot of Disk I/O errors. The answer is a data separator. This unit fits inside your expansion interface. It is supplied with full instructions and is a must for the serious disk user.

DISK DRIVE HEAD CLEANING DISKETTES

\$29.00 plus \$1.20 p&p

Disk drives are expensive and so are diskettes. As with any magnetic recording device, a disk drive works better and lasts longer if the head is cleaned regularly. In the past, the problem has been, how do you clean the head without pulling the mechanism apart and running the risk of damaging delicate parts. 3M's have come to our rescue with SCOTCH BRAND, non-abrasive, head cleaning diskettes which thoroughly clean the head in seconds. The cleaning action is less abrasive than an ordinary diskette and no residue is left behind. Each kit contains:

- 2 head cleaning diskettes
- 1 bottle of cleaning fluid
- 1 bottle dispenser cap.

\*\*\*\*\* DISKETTE BREAKTHROUGH \*\*\*\*\*

40 TRACK NASHUA DISKETTES (in library case)  
SINGLE SIDE/SINGLE DENSITY \$45.00 box of 10

40 TRACK VERBATIM DISKETTES  
DOUBLE SIDE/DOUBLE DENSITY \$59.00 box of 10

77 TRACK VERBATIM DISKETTES  
SINGLE SIDE/SINGLE DENSITY \$59.00 box of 10

All prices include p&p.

**NEWDOS 80 IS IN STOCK. (only \$149)**

This long-awaited disk operating system has now arrived and has already been delivered to many eager customers around Australia. It is every bit as good as it was cracked-up to be and MICRO-80 has decided to standardise on it for its own systems. Here are just a few of the many things you can do with NEWDOS 80 which is upward compatible with TRSDOS and NEWDOS + (i.e. TRSDOS and NEWDOS + programs will run on NEWDOS 80 but the reverse is not necessarily so.)

- \* New BASIC commands that support variable record lengths up to 4095 bytes long.
- \* Mix or match disk drives. Supports any track count from 18 to 96. Use 35, 40, 77 or 80 track 5" mini disk drives, 8" disk drives, OR ANY COMBINATION!
- \* An optional security boot-up for BASIC or machine code application programs. User never sees "DOSREADY" or "READY" and is unable to "BREAK", clear screen or issue any direct BASIC statements including "LIST".
- \* New editing commands that allow program lines to be deleted from one location and moved to another or to allow the duplication of a program line with the deletion of the original.
- \* Enhanced and improved RENUMBER that allows relocation of subroutines.
- \* Powerful program chaining.
- \* Device handling for routing to display and printer simultaneously.
- \* DFG function; simultaneous striking of the D, F and G keys will allow user to enter a mini-DOS to perform some DOS commands without disturbing the resident program.
- \* Includes Superzap 3.0 (improved, machine language version of Superzap) and all Apparat 2.1 utilities).

NEWDOS + 35 TRACK VERSION	\$99.00 incl. p&p
40 TRACK VERSION	\$110.00 incl. p&p

NEWDOS by Apparat, is the third generation disk operating system for your TRS-80. NEWDOS corrects over 70 errors and omissions in TRSDOS and Disk BASIC yet the two are completely compatible. Going from TRSDOS to NEWDOS is like going from Level I to Level II, more power, more convenience, greater speed. NEWDOS + includes the following utilities:

- \* Editor-assembler for Disk
  - \* Disassembler (Z80 machine code)
  - \* LM offset - allows transfers of any system tape to a Disk file-automatically relocated.
  - \* BASIC 1 - Lets you convert your computer back to Level I
  - \* LVIDKSL - Saves and loads BASIC 1 programs to disk
  - \* SUPERZAP - display/print/modify any location in memory or on disk.
  - \* RENUMBER BASIC program.
- and much more.

**\*\*\*\*\* MICRO-80 PRODUCTS \*\*\*\*\***

Please use order form on page 36

**\*\*\*\*\* MICROPOLIS 77 TRACK DISK DRIVES \*\*\*\*\***

These fabulous MICROPOLIS disk drives have more than double the storage capacity of the standard 35 track drives.

**\*\* DD-7S only \$775 incl. p&p \*\***  
77 track MICROPOLIS drive complete with cable for four drives, power supply, chassis and includes NEWDOS 80.

**\*\* DD-7 only \$649 incl. p&p \*\***  
Same as above but no cable or NEWDOS 80.

**\*\* DC-4 only \$45 incl. p&p \*\***  
4 drive connector cable - suitable for any disk drives.

**\*\*\*\*\* MPI DISK DRIVES \*\*\*\*\***

MPI is the second largest manufacturer of disk drives in the world. MPI drives use the same form of head control as on 8" drives and consequently, they have the fastest track-to-track access time available: 5msec. All MPIr drives are capable of single or double-density operation. (Double-density operation requires the installation of a special PC board in the expansion interface. This board is not yet available in Australia). As well as the single head 40 track disk drive, MPI also make a dual-head 40 track disk drive. A dual-head drive is almost as versatile as two single-head drives but is much cheaper.

Our MPI drives are supplied bare, set up to operate with TRS-80. They can be used bare or a simple cabinet made up from sheet metal (single-drive cabinets will be available in early December - price \$10.00). They require the use of a separate power supply giving 5 volt at 0.7 amp and 12 volt at 1 amp. All drives are sold with a 90 day warranty and service is available through MICRO-80 PRODUCTS.

MPI SINGLE HEAD DISK DRIVE only \$339 incl. p&p.  
MPI DOUBLE HEAD DISK DRIVE only \$449 incl. p&p.

AVAILABLE SOON:-  
MPI 80 TRACK SINGLE HEAD DRIVE only \$499  
MPI 80 TRACK DOUBLE HEAD DRIVE only \$599

Ring first to check availability.

**FLOPPY DOCTOR AND MEMORY DIAGNOSTIC**

(by MICRO CLINIC) \$29.95 plus 50c. p&p.

Two machine language programs on a diskette together with manual which thoroughly test your disk drives and memory. There are 19 possible error messages in the disk drive test and their likely causes are explained in the manual. Each pass of the memory tests checks every address in RAM 520 times, including the space normally occupied by the diagnostic program itself. When an error occurs the address, expected data, and actual data are printed out together with a detailed error analysis showing the failing bit or bits, the corresponding IC's and their location. This is the most thorough test routine available for TRS-80 disk users.

**\*\* PROGRAMS FROM CREATIVE COMPUTING \*\*****\*\* ADVENTURE PROGRAMS \*\***

**ADVENTURELAND (L2/16K) \$14.95 + 50c. p&p.**  
Try to find and take treasures as you explore a fantasy world. The computer acts as your puppet and carries out your two word commands. Sometimes you will need special objects to do certain things, often a little magic is necessary. Absorbing and challenging.

**THE COUNT ADVENTURE(L2/16K) \$14.95 + 50c. p&p.**  
In this adventure, you awaken in a bed in a castle in Transylvania. You don't know why you are there but you'd better solve the puzzle before it's too late. Just as enthralling as ADVENTURELAND but blood thirstier!

**\*\* GAMES \*\***

**AIR TRAFFIC CONTROLLER(L2/16K) \$9.95 + 50c.p&p**  
One of the hottest selling games in the USA, you are the Air Traffic Controller and the monitor is your radar screen. Bring down the aircraft safely and avoid mid-air collisions.

**Z CHESS (L2/16K) \$19.95 + 50c. p&p.**  
Seven levels of ability, contains all standard moves including castling and En Passant captures. It can play either black or white and its versatile board set-up mode allows specific positions to be played as desired.

**SPACE GAMES (L2/16K) \$13.50 + 50c. p&p.**  
4 Space Games including ULTRA-TREK, ROMULAN, STARWARS and STARLANES. Fast, real-time graphics.

**STRATEGY GAMES (L2/16K) \$9.50 + 50c. p&p.**  
5 Strategy games including TUNNEL VISION (find your way out of a 3-D maze), EVASION - avoid the deadly snake), JIGSAW (put the puzzle together), THE MASTERS (Golf on the '80 for up to 4 players), MOTOR RACING (Compete against the computer at Indy or the Grand Prix).

**GRAPHING PACKAGE (L2/16K) \$9.95 + 50c. p&p**  
A setof 6 utility programs which allow you to draw BAR GRAPHS, GRAPH CARTESIAN COORDINATES, carry out POLAR GRAPHING, PARAMETRIC GRAPHING, LINEAR REGRESSION and PARABOLIC REGRESSION.

**\*\* SCOTCH BRAND COMPUTING CASSETTES \*\***

These super-quality cassettes are now back in stock.

C-10 pack of 10 \$26.00 incl. p&p  
C-30 pack of 10 \$28.00 incl. p&p

**\*\*\*\*\* 16K MEMORY EXPANSION KIT \*\*\*\*\***

**## REDUCED TO ONLY \$59.00 !!!! ##**

These are prime, branded, 200 ns (yes, 200 nanosecond) chips. You will pay much more elsewhere for slow, 350 ns.chips. Ours are guaranteed for 12 months. A pair of DIP shunts is also required to upgrade the CPU memory in the TRS-80 - these cost an additional \$4.00. All kits come complete with full, step-by-step instructions which include labelled photographs. No soldering is required. You do not have to be an experienced electronic technician to instal them.

**\*\* PROGRAMS BY MICROSOFT \*\*****EDITOR ASSEMBLER PLUS (L2/16K)**

**#### REDUCED TO ONLY \$37.50 + \$1.20 p&p. ####**  
A much improved editor-assembler and debug/monitor for L2/16K TRS-80 or SYSTEM 80. Assembles directly into memory, supports macros and conditional assembly, includes new commands-substitute, move, copy and extend.

**LEVEL III BASIC \$59.95 plus \$1.20 p&p.**

Loads on top of Level II BASIC and gives advanced graphics, automatic renumbering, single stroke instructions (shift-key entries) keyboard debounce, suitable for L2/16K and up (Not Disk BASIC)

**ADVENTURE ON DISK \$35.95 plus \$1.20 p&p.**

This is the original ADVENTURE game adapted for the TRS-80. The game fills an entire diskette. Endless variety and challenge as you seek to rise to the level of Grand Master. Until you gain skill, there are whole areas of the cave that you cannot enter. (Requires 32K One Disk)

**DISK EDITOR ASSEMBLER**

**SAVE \$30 ONLY \$107 plus \$2.00 p&p.**  
Supports Macros, linking loader, editor, cross references. See Assembly language programming article in August MICRO-80 for further details. (Requires 32K One Disk)

**BASIC COMPILER**

**\$208 plus \$2.00 p&p.**  
Converts Disk BASIC programs to machine code, automatically. A compiled program runs, on average, 3-10 times faster than the original BASIC program and is much more difficult to pirate. Note: MICROSOFT have temporarily withdrawn this program for reworking to make it less memory-hungry. The revised version should be available about the end of NOVEMBER. All existing owners will receive updated versions at no cost to them. Don't wait though, send in your orders now and be near the front of the queue.

**\*\*\*\*\* BOOKS \*\*\*\*\***

**LEVEL II ROM REFERENCE MANUAL \$24.95+\$1.20 p&p**  
Over 70 pages packed full of useful information and sample prorams. Applies to both TRS-80 and SYSYEM 80.

**TRS-80 DISK AND OTHER MYSTERIES**

**\$24.95+\$1.20 p&p.**

The hottest selling TRS-80 book in the U.S.A. Disk file structures revealed, DOS's compared and explained, how to recover lost files, how to rebuild crashed directories - this is a MUST for the serious Disk user and is a perfect companion to any of the NEWDOS's.

**\*\* OTHER PROGRAMS \*\*****INFINITE BASIC BY RACET (32K/1DISK)**

**\$49.95 + 50c. P&P.**  
Full matrix functions - 30 BASIC commands; 50 more STRING functions as BASIC commands

**GSF/L2/48K \$24.95 + 50c. p&p.**  
18 machine language routines including RACET sorts.

**BUSINESS ADDRESS AND INFORMATION SYSTEM**

**(48K/DISK) \$24.95 + 50c. p&p.**  
Allows you to store addresses and information about businesses, edit them and print them out.

## \*\*\*\*\* SOFTWARE BY AUSTRALIAN AUTHORS \*\*\*\*\*

All our software is suitable for either the SYSTEM 80 or the TRS-80 (with the exception of Me Think it Micro Mazin vol.2).

## \*\* UTILITIES \*\*

S-KEY by Edwin Paay \$15.95 plus 50c. p&p  
S-KEY is a complete keyboard driver routine for the TRS-80 and becomes part of the Level II basic interpreter. With S-KEY loaded the user will have many new features not available with the standard machine.

S-KEY features:

- \* S-KEY provides an auto-repeat for all the keys on the keyboard. If any key is held down longer than about half a second, the key will repeat until it is released.
- \* Graphic symbols can be typed direct from the keyboard, this includes all 64 graphic symbols available from the TRS-80/SYSTEM 80
- \* S-KEY allows text, BASIC commands and/or graphics to be defined to shifted keys. This makes programming much easier as whole commands and statements can be recalled by typing shift and a letter key.
- \* Because S-KEY allows graphics to be typed directly from the keyboard, animation and fast graphics are easily implemented by typing the appropriate graphics symbols directly into PRINT statements.
- \* S-KEY allows the user to LIST a program with PRINT statements containing graphics, properly. S-KEY does this by intercepting the LIST routine when necessary.
- \* S-KEY allows the user to list an updated list of the shift key entries to the video display or line printer.
- \* S-KEY can be disabled and enabled when required. This allows other routines which take control of the keyboard to run with S-KEY as well.

Each cassette has TRS-80, DISK and SYSTEM 80 versions and comes with comprehensive documentation.

BMON by Edwin Paay \$19.95 plus 50c. p&p  
THE ULTIMATE HIGH MEMORY BASIC MONITOR  
L2/16-48K

Our own personnel refuse to write BASIC without first loading this amazing machine language utility program into high memory! BMON Renumbers; Displays BASIC programs on the screen while they are still loading; tells you the memory locations of the program just loaded; lets you stop a load part-way through; merges two programs, with automatic renumbering of the second so as to prevent any clashes of line numbers; recovers your program even though you did type NEW: makes one program invisible while you work on a second (saves hours of cassette time!); lists all the variables used in the program; makes SYSTEM tapes; lets you Edit memory directly...the list goes on and on. Cassette comes with 16K, 32K and 48K versions, ready to load. Can anyone afford NOT to have BMON?

## \*\*\*\*\* LNW RESEARCH EXPANSION INTERFACE BOARDS - \$79.95 + \$2.00 p&amp;p \*\*\*\*\*

These are the Expansion interface boards featured in the constructional article in September's MICRO-80. They are the basis for a high quality, reliable expansion interface for the TRS-80 (or the SYSTEM 80 via out SYSPAND 80 adaptor). Initially we are offering the boards with their comprehensive instruction manuals. Shortly we will offer modular kits of components and fully built up and tested units. Now in stock but hurry, the orders are pouring in.

## \*\* EDUCATIONAL \*\*

RPN CALCULATOR (L2/16K & 32K) \$14.95+50cp&p

Give your computer the power of a \$650 reverse polish notation calculator with 45 functions and selectable accuracy of 8 or 16 digits. The main stack and registers are continuously displayed whilst the menu is always instantly accessible without disturbing any calculations or register values. The cassette comes with both the 16K and 32K versions, the latter giving you the additional power of a programmable calculator. Comes with a very comprehensive 15 page manual, which includes instructions to load and modify the 32K programmable version to run in 16K. Whether for business or pleasure, this package will prove invaluable, and turn your '80 into a very powerful instrument.

TOUCHTYPE (L2/4K) \$19.95 + 50c. p&p.

An interactive, 22 lesson typing course which uses the computer's keyboard and screen to teach you to type rapidly and accurately and, a massive cassette data dump to control your progress. The computer checks for accuracy, and sets timed exercises to check your progress. If you have to look at each key before you press it, or only use two fingers, then this program, plus a little perseverance, will do some amazing things to your typing speed.

## \*\* GAMES \*\*

U BOAT \$7.50 plus 50c. p&p.

Real time simulation at its best! Comes with working sonar-screen and periscope, a full rack of torpedoes, plenty of targets, working fuel and battery meters, helpful Mothership for high-seas reprovisioning and even has emergency radio for that terrible moment when the depth charges put your crew at risk. Requires Level II/16K.

SPACE INVADERS \$7.50 plus 50c. p&p.

Much improved version of this arcade favourite with redesigned laser and cannon blasts, high-speed cannon, 50 roving drone targets, 10 motherships and heaps of fun for all. Level II with 4K and 16K versions on this cassette.

MMM- vol.1 and vol.2. \$7.50 ea. plus 50c. p&p.

Two, three-game cassettes from that master of TRS-80 graphics, Charlie Bartlett. Vol.1 brings you INDY 500, an exciting road race that gets faster and faster the longer you play, SUBHUNT in which your warship blows up unfortunate little submarines all over the place, and KNIEVEL (as in motorcycle, ramp and buses).

Vol.2 (not System 80 compatible) contains TANK, a two player battle of speed and skill, THIEF (steal an old man's hoard of loot then escape from the labyrinth before he gets you), and SHOOTOUT (a Mexican style high-noon for two).

SNAKE by M. Svendsdotter ... 4K/L1

Once you type this game into your '80, make sure you have a whole day to spare. To start with, the board is cleared, this will take about five seconds. Straight away you'll see displayed on the screen (SPACE) CLOCKWISE DIRECTION CHANGE and under that ( ) COUNTER-CLOCKWISE DIRECTION CHANGE, you're probably thinking to yourself, how can I move anywhere on the board with only two controls and stay sane at the same time. You'll probably want to put your foot through the screen to start with but keep trying. you'll have more fun remembering which key makes the snake go in what direction than using your '80 for football practice.

The game starts by drawing the outline of the board on the screen then the snake appears in the form of a string of "S's" and starts moving (left to right). As the game proceeds, the snake gradually gets longer. At first it moves slowly, which gives you a little time to get used to the controls. The pace will gradually increase until you find it impossible to press the right key. This snake's favourite food is frogs' legs (with the frogs still attached). You score 10 points each time the snake eats a frog, done by going over the top of one of the Fs on the board. At line 90 you'll see among other things F.Z=64T0575:A(Z)=0:N.Z this is the part of the program which sets all board positions from 64 to 575 to 0. The arrays from A(64) to A(575) are given the value of 0 (for more information on arrays see Chapter 21 of your Level I Users' Manual). The two lines which draw the board are 100 and 140. In these two lines you'll see .....GOS.400:...., this sub-routine puts the border on the screen and gives the arrays at those locations the value of 9, so your '80 can see where the border is. Line 210 prints the next S, increments to the next position, gradually speeds up the snake, takes away an S from the tail and decides when and where the next frog will appear on the board. SUB 300 is the part of this program which takes away the S, done by PRINTING AT R; (P. AT P; puts the cursor between the two graphics blocks at the bottom of the board, which helps the Level I simulate the inkey function found in Level II machines); the rest of this line sets the array behind the tail back to 0 after the S is taken away and then decrements T (counts backwards) which is the timing loop used to control game speed. Line 330 decides if a frog is to appear and then prints an F on the board. At 220 if POINT(X,Y) and POINT(X+3,Y) are turned on and the next position that the snake is to move to equals 0, it gives the array A(S) a value of M (next position) and jumps back to line 210.

(SNAKE for Level II machines was published in MICRO-80 Issue 1, December 1979)

```

1 REM                               SNAKE FOR LEVEL I MACHINES
2 REM                               BY - MICHAEL SVENDSDOTTER
3 REM
4 REM                               COPYRIGHT - MICRO-80 NOVEMBER 1980
5 REM                               THE INTERNATIONAL MAGAZINE FOR '80 USERS
6 REM                               PO BOX 213 GOODWOOD, SOUTH AUSTRALIA 5034
10 C.
70 P.AT458,"<SPACE> CLOCKWISE DIRECTION CHANGE
80 P.T.(10);*(<1> COUNTER-CLOCKWISE DIRECTION CHANGE
90 P.AT0,"BOARD IS BEING CLEARED":F.Z=64T0575:A(Z)=0:N.Z
100 C.:F.B=0T0577S.577:F.Z=BT0B+62S.Z:GOS.400:A(Z+1)=9:N.Z:N.B
140 F.B=127T064S.-63:F.Z=BT0B+448S.64:GOS.400:N.Z:N.B
150 X=123:Y=30:P=702:S=287:R=S:T=400:E=0:M=2:A(S)=2
200 S.(X,Y):S.(X+3,Y)
210 P.ATS,"S";ATP,:S=S+M:F.Z=0TOT:N.Z:IFR.(2)=1GOS.300
220 IF(P.(X,Y))*(<P.(X+3,Y))*(<A(S)=0)A(S)=M:G.210
230 IF(P.(X,Y)=0)+(P.(X+3,Y)=0)GOS.500:A(S)=M:G.200
240 IFA(S)=8A(S)=M:E=E+10:P.AT0,"SCORE =";E;ATP,:G.200
250 P.AT704,
260 IFA(S)=M*(-1)P."SSLOW DOWN YOUR FINGERS":G.290
270 IFA(S)=9P."YOU'VE HIT A WALL":G.290
280 P."THE SNAKE HAS STOPPED
290 I."PRESS ENTER";A$:G.10
300 P.ATR;ATP,:N=A(R):A(R)=0:R=R+N:IFT>0T=T-3
330 F=RND(287)*2+1:IF(A(F)=0)*(<R.(5)=1)A(F)=8:P.ATF,"F";ATP,
350 RET.
400 A(Z)=9:P.ATZ,"*";:RET.
500 P.ATP;:S=S-M:IFP.(X+3,Y)G.550
510 IFM=2G.600
520 IFM=-2G.610
530 IFM=64G.630
540 G.620
550 IFM=-2G.600
560 IFM=2G.610
570 IFM=64G.620
580 G.630
600 M=64:RET.
610 M=-64:RET.
620 M=2:RET.
630 M=-2:RET.

```

## \*\*\*\*\* LEVEL II PROGRAMS \*\*\*\*\*

DRAW II                    by John Peschar                    LII/4-16K

This is a much better version of the Tandy Draw program, it has all the features of the original plus lots more. The program will run in a 4K machine if the instructions are left out, hence the two separate listings.  
Here is a summary of the instructions.

The ARROW keys set the path taken by the cursor (for as long as the key is held down).  
SHIFT ARROW resets the path taken by the cursor (for as long as the key is held down).  
SHIFT S Saves graphics onto tape.  
SHIFT L Loads graphics from tape.  
SHIFT N Creates negative of graphics.  
SHIFT C Same function as the CLEAR key.  
The 0 to 9 keys are used to enter a number into the increment counter. (If you enter a number and then direct the cursor, it will set or reset for that length. This is the increment function.)  
SHIFT R Resets increment counter to 0.  
SHIFT D Displays increment value and the X and Y coordinates of the cursor. Display is held for as long as the key is pressed.  
The programs have been numbered so that 16K users may merge the two programs together if they wish.

```

1 *   - = * DRAW II * = -
2 *   JOHN PESCHAR
3 *   134 MAIN RD. MARKS POINT 2280 N.S.W.
4 *   TEL. (049) 453616
5 *   COPYRIGHT JUNE 1980
10 CLS: CLEAR500: DEFINTX, Y, B-Z: DEFSTR A: SET(X, Y): V=14400
20 H=H+1: A=INKEY$: IFA=" " AND H>4 THEN SET(X, Y): H=0: GOT020 ELSE IFA=" " THEN RESET(X, Y): GOT020
30 F=2: B=ASC(A): IF B>47 AND B<58 AND LEN(A1)+1<4, A1=A1+A: C=VAL(A1): IF C>127, C=127: GOT020 ELSE GOTO20
40 IF B=91 OR B=100 OR B=90 OR B=8, F=0 ELSE IF B=260 OR B=270 OR B=240 OR B=25, F=1
50 IFF=0 AND C=0 THEN 210 ELSE IFF=0 THEN 100
60 IFF=1 AND C=0 THEN 200 ELSE IFF=1 THEN 90
70 IF B=108 THEN 500 ELSE IF B=115 THEN 400 ELSE IF B=99 THEN CLS ELSE IF B=110 THEN 600 ELSE IF B=100 THEN 300 ELSE IF B=114 THEN A1=" ": C=0
80 GOT020
90 FORD%=1 TO C: SET(X, Y): RESET(X, Y): GOSUB110: SET(X, Y): NEXT: A1=" ": C=0: GOT020
100 FORD%=1 TO C: X1=X: Y1=Y: RESET(X, Y): GOSUB110: SET(X, Y): SET(X1, Y1): NEXT: A1=" ": C=0: GOT020
110 IF (B=270 OR B=91) AND Y-1>=0, Y=Y-1
120 IF (B=260 OR B=10) AND Y+1<=47, Y=Y+1
130 IF (B=240 OR B=8) AND X-1>=0, X=X-1
140 IF (B=250 OR B=9) AND X+1<=127, X=X+1
150 A=INKEY$: IFA<>" " THEN D=C: RETURN ELSE RETURN
200 E=0: SET(X, Y): RESET(X, Y): GOSUB220: SET(X, Y): IF E=0, 200 ELSE 210
210 E=0: X1=X: Y1=Y: RESET(X, Y): GOSUB220: SET(X, Y): SET(X1, Y1): IF E=0 THEN 200 ELSE 210
220 IF (PEEK(V) AND 8)=0 AND Y-1>=0, Y=Y-1: E=1
230 IF (PEEK(V) AND 16)=16 AND Y+1<=47, Y=Y+1: E=1
240 IF (PEEK(V) AND 32)=32 AND X-1>=0, X=X-1: E=1
250 IF (PEEK(V) AND 64)=64 AND X+1<=127, X=X+1: E=1: RETURN ELSE RETURN
300 SET(X, Y): FORT%=15360 TO 15381: A2=A2+CHR$(PEEK(T%)): NEXT
310 IF (PEEK(14337) AND 16)=16, PRINT@0, " INC=" C*X=" X*Y=" Y: GOT0310 ELSE PRINT@0, A2: A2=" ": GOT020
400 SET(X, Y): D=15360: X1=PEEK(15423): X2=PEEK(15422): PRINT#-1, X, Y, X1, X2
410 A3=" ": FOR Z=DT016383: B=PEEK(Z)-119: IF B<10 NEXT: A4=" ": GOT0440
420 A6=STR$(Z-15360): A6=RIGHT$(A6, LEN(A6)-1): A5=STR$(B): A5=RIGHT$(A5, LEN(A5)-1): IFLEN(A6)=4, A6="X"+RIGHT$(A6, 2)
430 IFLEN(A5)<3, A5="0"+A5: GOT0430 ELSE A4=A6+A5
440 A3=A3+A4: IFLEN(A3)=2450 OR A4=" " THEN 450 ELSE NEXT: A4=" ": GOT0440
450 PRINT#-1, A3: D=Z+1: IFA4=" ", PRINT#-1, " END ": GOT020 ELSE 410
500 CLS: INPUT#-1, X, Y, X1, X2
510 INPUT#-1, A3: IFA3=" END " THEN POKE15423, X1: POKE15422, X2: GOT020
520 FOR Z=1 TO LEN(A3) STEP 5: A7=MID$(A3, Z, 3): IF LEFT$(A7, 1)="X", A7="10"+RIGHT$(A7, 2)
530 POKEVAL(A7)+15360, VAL(MID$(A3, Z+3, 2))+119: NEXT: GOT0510
600 FOR I%=15360 TO 16383: J%=PEEK(I%): IF J%<128 POKE I%, 191: NEXT I%: GOT020 ELSE POKE I%, 319-J%: NEXT I%: GOT020

```

INSTRUCTIONS FOLLOW

```
700 CLS:PRINTTAB(18)*- = * DRAW II * = -
```

DRAW II ALLOWS THE OPERATOR TO CREATE GRAPHIC IMAGES ON THE SCREEN, AND TO SAVE THEM ON CASSETTE OR TO LOAD GRAPHICS FROM THE TAPE BACK ONTO THE SCREEN AND CONTINUE 'DRAWING' FROM WHERE IT WAS LEFT OFF.

710 PRINT"DRAWING IS ACCOMPLISHED BY DIRECTING A FLASHING CURSOR IN THE DESIRED DIRECTION WITH THE USE OF THE FOUR ARROW KEYS. THE FOUR ARROW KEYS ON THEIR OWN WILL 'SET' THE CURSORS PATH, WHEREAS <SHIFT ARROW> WILL 'RESET' THE CURSORS PATH.  
720 PRINT"THESE ARE ALSO SIX SPECIAL FUNCTION KEYS WHICH CAN HELP YOU TO SAVE TIME

HIT ANY KEY WHEN READY

```
730 IFINKEY$=""THEN730ELSECLS:PRINT"
```

<SHIFT><L> = LOADS GRAPHICS FROM TAPE.

CURSOR WILL START FLASHING WHEN LOADING IS FINISHED<SHIFT><S> = SAVES GRAPHICS ONTO TAPE.

CURSOR WILL START FLASHING WHEN SAVING IS FINISHED

```
740 PRINT"<SHIFT><C> = SAME AS THE <CLEAR> KEY.
```

<SHIFT><N> = CREATES NEGATIVE OF DISPLAYED GRAPHICS.

THIS COMMAND CHANGES ALL BLACK AREAS TO WHITE AND VISA-VERSA.

```
750 PRINT"
```

IF YOU TYPE IN A NUMBER USING THE 0 TO 9 KEYS AND THEN DIRECT THE CURSOR WITH THE ARROW KEYS, IT WILL 'SET' (OR 'RESET') FOR THAT LENGTH AS INDICATED. THIS IS THE INCREMENT FUNCTION.

HIT ANY KEY WHEN READY

```
760 IFINKEY$=""THEN760ELSECLS:PRINT"
```

THERE ARE TWO KEYS TO TO AID YOU IN OPERATING THE INCREMENT FUNCTION. THESE ARE

<SHIFT><R> = RESETS THE INCREMENT COUNTER TO ZERO

```
770 PRINT"
```

<SHIFT><D> = WHEN THIS KEY IS PRESSED (AND HELD DOWN), THE VALUE CURRENTLY IN THE INCREMENT COUNTER AND THE X AND Y COORDINATES OF THE CURSOR WILL BE

780 PRINT" DISPLAYED IN THE TOP LEFT HAND CORNER OF THE SCREEN. WHEN THE <D> KEY IS RELEASED ANY GRAPHICS THAT WERE DESTROYED WILL BE RESTORED.

HIT ANY KEY WHEN READY

```
790 IFINKEY$=""THEN790ELSECLS:PRINT"* NOTE *
```

IF YOU TRY TO INPUT A VALUE GREATER THAN 127 INTO THE INCREMENT COUNTER, IT WILL BE SET TO 127 AND MUST WITH THE <SHIFT><R> COMMAND.

BE RESET

```
800 PRINT"
```

IF YOU WISH TO STOP THE INCREMENT MOVEMENT OF THE CURSOR THEN HIT ANY KEY WHILE THE CURSOR IS IN MOTION WILL STOP AND THE INCREMENT COUNTER WILL BE RESET TO ZERO.

AND IT

```
810 PRINT"
```

THATS ALL THE INSTRUCTIONS  
H A P P Y D R A W I N G

--:\*:--

```
820 IFINKEY$=""THEN820
```

- 0000000000 -

STRING EDITOR by C.E. Kendall LII/4K

Many users must have wished that the program editor in Level 2 BASIC could be used on text much like the program being used to write this, (ELECTRIC PENCIL). Well here is a demonstration program that gives you some of the functions of a proper text editor. In the demonstration program a string is entered from the keyboard. It is then displayed with a blinking cursor to indicate the position at which the correction will take place.

Editing functions are called by using the arrow keys (ESC, CTRL & BACK SPACE on SYSTEM 80/VIDEO GENIE) so no printing characters are lost. The cursor is moved one space to the left or the right by using the left and right arrow keys (right movement not available on an unmodified SYSTEM 80/VIDEO GENIE).

To change a character, move the cursor to the desired position and type in the correct character (YOU DO NOT NEED TO PRESS THE ENTER/NEWLINE KEY).

To insert new text, position the cursor one character past the point at which the insert is to start, press SHIFT and the up arrow/ESC together, then the new characters and "ENTER/NEWLINE".

For deletion, the cursor is positioned, SHIFT and down arrow/CTRL are pressed together, then the number of characters to be deleted and "ENTER/NEWLINE".

Instructions are included in the program as well.

```

10  *PROGRAM TO DEMONSTRATE AN EDITOR FOR STRING TEXTS
    BY  C.E.KENDALL
        RMB 43, MITCHELL RD., WAGGA WAGGA, NSW 2650
        AUG. 1980
20  CLS: CLEAR 1000: DEFINIT A-P,R-Z: PRINT TAB(15)*"EDITOR DEMONSTRATION": PRINT:
PRINT @ 960, "NOTE - <E> MEANS PRESS ENTER";: PRINT @ 256, "": INPUT "TYPE 1 FOR
INSTRUCTIONS, OTHERWISE <E> ONLY";Q: IF Q=0 THEN 70
30  PRINT @64, "USE "; CHR$(93);" OR "; CHR$(94);" TO MOVE CURSOR TO LEFT OR RIGH
T (SHIFT ARROW GIVES 8 SPACES SHIFT).

TO ALTER A CHARACTER, TYPE IN THE NEW ONE.

TO INTERPOLATE, SET CURSOR TO THE CHAR. AFTER THE POINT OF
INSERTION,";
40  PRINT" TYPE SHIFT [ AND THE NEW MATTER, THEN <e>."
50  PRINT "TO ADD TO THE BEGINNING, SET CURSOR AT 1ST CHARACTER; TO ADD
TO END, SET CURSOR ONE SPACE BEYOND THE LAST & USE SHIFT ";CHR$(91);".
IF INTERPOLATION STARTS WITH A SPACE, USE ";CHR$(94);" BEFORE IT."
60  PRINT:PRINT"TO DELETE, TYPE SHIFT ";CHR$(92);" THEN NO. OF CHAR. TO DELETE &
<E>.

WHEN EDITING FINISHED, TYPE ";CHR$(92);" .":INPUT" <E>";@
70  CLS: INPUT "ENTER STRING TO BE EDITED"; A$: CLS: PRINT A$: PRINT @ 128,A$; ST
RING$(LEN(A$),24);: Q=.5
80  PRINT CHR$(14.5+Q);: Q=-Q: FOR X=0 TO 40: NEXT: Z$=INKEY$: IF Z$="" THEN 80
'BLINKING CURSOR - CHANGE X LIMIT TO ALTER SPEED
90  Z=ASC(Z$): P=POS(0)
100 IF Z=26 THEN 150 ELSE IF Z=27 THEN 140 ELSE IF Z=10 THEN 160
110 IF Z=9 THEN PRINT CHR$(25); ELSE IF Z=25 THEN PRINT STRING$(8,25); ELSE IF Z
=8 THEN PRINT CHR$(24); ELSE IF Z=24 THEN PRINT STRING$(8,24); ELSE R=1: GOTO 13
0 'CURSOR MOVEMENT
120 GOTO 80
130 B$=LEFT$(A$,P)+Z$+RIGHT$(A$,LEN(A$)-P-R): A$=B$: B$="": PRINT @ 128, CHR$(30
); A$: STRING$(LEN(A$)-P,24);: Q=.5: GOTO 80 'REBUILD THE STRING
140 PRINT @ 768, CHR$(30);: INPUT Z$: R=0: GOTO 130 'INTERPOLATE
150 PRINT @ 768, CHR$(30);: INPUT R: Z$="": GOTO 130 'DELETE
160 PRINT: PRINT "EDITING COMPLETED"

```

- 000000000 -

MIGHTY MORMAR by Charlie Bartlett L2/16K

Oh! my Mighty Mormar, you were on your way to our home planet of Hartley with Princess Aleaya on board when the evil General Vagg's Battle Cruiser caught us with a tractor beam and brought us aboard. He then disarmed you, put out your eyes, took the princess and left you for dead in your starship, which he has drained of fuel and left sitting on the flight deck of his battle cruiser. But, my Mighty Mormar he did not see me, your little Robot, stowed away in the corner and he did not count on the courage of you my master, who even though unarmed and blinded will use me as your eyes to rescue the Princess. Being a small robot I only understand a few words so you may need to ask your questions in a different way if I do not understand. We will be rewarded with points for anything we steal along the way and together we will prevent the evil General Vagg from destroying our home planet of Hartley and once again prove that evil does not PAAY.  
(If you really must cheat, look at the program data lines to see which words I understand.)  
(Please note, Messrs Vagg, Hartley and Paay are contemplating legal proceedings citing defamation of characters (ASCII), any reader who uses this program is likely to be regarded as an accomplice - Ed.)

```

1  CLS:PRINT"          MIGHTY MORMAR":CLEAR200:RANDOM:DEFINTA-Z:GOSUB230:
LC=2:SL=64:BL=4:GF=50:RV=16396:GOTO182
6  IF TC<250RRND(GF)>>1THEN16
7  IF TC=300GF=20
8  IF LC<30RLC=90RLC=260RLC=360RLC=37THEN16
9  IF LC>26ANDLC<31THEN16

```

```

10 PRINT"ORMOR. AN ARMED VAGGIAN JUST WALKED IN."
11 GOSUB211:IFVB<>12ORNO<>15THEN104
12 X=13:GOSUB224:IFY<>-1THEN104
13 IFBL=0CLS:PRINT"I'M OUT OF AMMUNITION.":PRINT:GOTO105
14 PRINT"ZZZAP! NO MORE GUARD."
15 BL=BL-1:IFBL=0PRINT"I'M OUT OF AMMUNITION."
16 IFMD<>TCTHEN19
17 X=22:GOSUB224:IFY<>-1THEN19
18 PRINT"YOUR HAMBURGER IS COLD."
19 GOSUB211:TC=TC+1
20 IFVB=0ANDNO=0THEN21ELSE23
21 PRINT"I DON'T KNOW HOW TO DO THAT.":GOTO6
22 PRINT"NOTHING HAPPENED.":GOTO6
23 IFVB>10RNO>7THEN36
24 IFNO=0THEN21
25 IFDS(LC,NO-1)=0PRINT"I CAN'T GO THAT WAY!":GOTO6
26 IFDRANDDS(LC,NO-1)>2ANDDS(LC,NO-1)<6PRINT"I CAN'T GO THAT WAY. FLIGHT DECK ID
OORS ARE OPEN.
NO AIR !!!":GOTO6
27 IFNO=3ANDLC=31ANDNOTDJPRINTM0$:GOTO6
28 IFLC=35ANDDS(LC,NO-1)=36ANDOB(21,1)<>0PRINT"THE ROBOT WON'T LET ME THROUGH.":
GOTO6
29 IFLC=17ANDOB(13,1)=17THEN104
30 IFLC=9ANDOB(5,1)=9THEN104
31 IFLC=9ORLC=17THENHE$(LC)="
32 LC=DS(LC,NO-1)
33 IFLC=34HE$(LC)="
34 IFLC=26THEN204
35 GOTO182
36 IFVB=3GOTO182
37 IFVB<>2THENS3
38 IFNO=0PRINT"I DON'T KNOW WHAT A ";CHR$(34);NO$(0);CHR$(34);" IS.":GOTO6
39 IFCR>5PRINT"I CAN'T CARRY ANY MORE.
HINT : DROP SOMETHING.":GOTO6
40 FORI=1TOLO:IFOB(I,0)=NOXTHEN41ELSENEXTI:GOTO21
41 IFOB(I,1)--1PRINT"I'M ALREADY CARRYING IT.":GOTO6
42 IFOB(I,1)<>LCPRI" I DON'T SEE IT.":GOTO6
43 IFNO<>37THEN46
44 X=13:GOSUB224:IFY<>-1PRINT"I DON'T HAVE A BLASTER TO PUT IT IN.":GOTO6
45 BL=4:OB(I,1)=0:PRINT"MY BLASTER'S RELOADED.":FORI=1TO1:NEXTI:GOTO6
46 IFNO=15ORNO=25ORNO=34PRINT"HE LOOKS PRETTY MEAN TO ME.":GOTO6
47 CR=CR+1:OB(I,1)--1:PRINT"O.K.":FORI=1TO1:NEXTI
48 IFNO=14ANDNOTCMPRI" A VOICE SAYS: ";CHR$(34);"SESAME";CHR$(34);".":CM=-1
49 IFNO=22ANDMD=0THENMD=TC+50
50 IFNO=12HE$(2)="
51 IFNO=13HE$(7)="
52 GOTO6
53 IFVB<>4THENS8
54 PRINT"I'M CARRYING:"
55 K=0:FORI=1TOLO:IFOB(I,1)--1PRINTOB$(I):K=1
56 NEXTI:IFK=0PRINT"NOTHING"
57 PRINT:GOTO6
58 IFVB<>5THENS9ELSEGOSUB220:GOTO6
59 IFVB<>6THENS7
60 IFNO=0THEN38
61 IFLC=2PRINT"THERE'S NO ROOM IN HERE.":GOTO6
62 IC=0:FORI=1TOLO:IFOB(I,1)=LCTHENIC=IC+1
63 NEXTI:IFIC>12PRINT"THERE'S NOT ENOUGH ROOM. GET RID OF SOMETHING.":GOTO6
64 FORI=0TOLO:IFOB(I,0)=NOXTHEN65ELSENEXTI:GOTO21
65 IFOB(I,1)<>-1PRINT"I'M NOT CARRYING IT.":GOTO6
66 CR=CR-1:OB(I,1)=LC:PRINT"O.K.":FORI=1TO1:NEXTI:GOTO6
67 IFVB<>7THENS9
68 IFHE$(LC)=" THENPRINT"HOW AM I SUPPOSED TO KNOW WHAT TO DO ?":GOTO6ELSEPRINTH
E$(LC):GOTO6
69 IFVB<>8THENS1
70 PRINT"PRESS (ENTER) WHEN READY TO : RECORD *":GOSUB226:FORI=0TOLO:PRINT#-1,OB
(I,0),OB(I,1),OB(I,2):NEXT:PRINT#-1,TB,TC,CM:PRINT#-1,DR,BL,MD:PRINT#-1,GF,DJ,CR
:PRINT#-1,LC:PRINT"O.K.":GOTO6
81 IFVB<>9THENS3
82 PRINT"PRESS (ENTER) WHEN READY TO * READ TAPE *":GOSUB226:FORI=0TOLO:INPUT#-1
,OB(I,0),OB(I,1),OB(I,2):NEXT:INPUT#-1,TB,TC,CM:INPUT#-1,DR,BL,MD:INPUT#-1,GF,DJ
,CR:INPUT#-1,LC:GOTO182
93 IFVB<>10THENS94ELSECLS:J=0:GOTO198
94 IFVB<>11THEN108

```

```

95 IFNO<>10THEN21
96 IFLC<>2ANDLC<>11PRINT*WHAT BUTTON,* :GOTO6
97 IFLC=11ANDNOTTBTHENTB=-1:PRINTN2$:GOTO6
98 IFLC=11ANDTBTHENTB=0:PRINTN3$:GOTO6
99 X=12:GOSUB224:IFY<>1THEN22
100 X=24:GOSUB224:IFY<>1THEN22
101 IFNOTTBPRINTN3$:GOTO22
102 IFNOTDRPRINTN4$:GOTO22
103 GOTO197
104 CLS:PRINT*H E L P   ! ! !   *:PRINT
105 PRINT*VAGG SOLDIERS ARE EVERYWHERE. I'VE BEEN CAPTURED.*
106 PRINT*I'M NOW A PRISONER. WOE IS ME...
107 GOTO201
108 IFVB<>12ORN0=0THEN118
109 IFBL=0PRINT* BUT I DON'T HAVE ANY AMMUNITION LEFT.* :GOTO6
110 X=13:GOSUB224:IFY<>-1PRINT* BUT I'M NOT CARRYING A BLASTER.* :GOTO6
111 X=N0:GOSUB224:IFY=-1PRINT* I CAN'T. I'M HOLDING IT.* :GOTO6
112 IFNO=34PRINT*ZZZAP!* :BL=BL-1:GOTO6
113 IFY<>LCPRINT* I DON'T SEE IT.* :GOTO6
114 FORI=1TOLO:IFOB(I,0)=N0THEN115ELSENEXT:GOTO21
115 OB(I,1)=0:FORI=1TO1:NEXT:PRINT*ZZZAP!!! THE *;N0$(N0);* VAPORIZED.*
116 BL=BL-1:IFBL=0PRINT* I'M OUT OF AMMUNITION.*
117 GOTO6
118 IFVB<>13THEN126
119 IFNO=0PRINT* SAY WHAT? * :GOTO6
120 X=14:GOSUB224
121 IFY<>-1ORN0<>19PRINT* O.K.      *;N0$(N0):GOTO6
122 IFDRGOTO22
123 DR=-1:PRINT* A VOICE COMES OVER THE P.A. SYSTEM AND SAYS :
OPENING FLIGHT DECK DOORS
*
124 IFLC>2ANDLC<6PRINT*
YIPS!!! THERE'S NO AIR!!! CROAK...*:END
125 GOTO6
126 IFVB<>14THEN143
127 DCUIO<>20ANDNO<>16ANDNO<>11ANDNO<>33THEN21
128 IFNO=20THEN137
129 IFNO<>16THEN133
130 IFOB(6,1)=-1PRINT* SORRY. I CANT READ MAPS* :GOTO6
131 IFOB(6,1)=LCPRINT* TRY... GET MAP.* :GOTO6
132 PRINT* IT'S NOT HERE.* :GOTO6
133 X=N0:GOSUB224:IFY<>LCANDY<>-1THEN132
134 IFNO=11PRINT* IT SAYS: >> NEEDS TURBO <<*
135 IFNO=33PRINT* IT SAYS: >> OUT OF ORDER <<*
136 GOTO6
137 IFLC<>13PRINT* I DON'T SEE ANY.* :GOTO6
138 PRINT:PRINT* IT SAYS ON THE WALL,*
139 PRINT*>> YOUR MOTHER'S GOT A BIG NOSE <<*
140 PRINT*>> KILROY MADE IT HERE.<<*
141 PRINT*>> SAY SECURITY <<*
142 GOTO6
143 IFVB<>15THEN150
144 IFNO=0PRINT* WHAT'S A *;N0$(0);*?*:GOTO6
145 IFNO<>22PRINT* DON'T BE DAFT.* :GOTO6
146 X=22:GOSUB224:IFY<>-1PRINT* I'M NOT HOLDING IT.* :GOTO6
147 FORI=1TOLO:IFOB(I,0)=22THEN148ELSENEXTI:PRINT* I DON'T KNOW WHERE IT IS.* :GOTO6
148 OB(I,1)=0:PRINT* CHUMP - CHUMP. HUMMM, GOOD.*
149 FORI=1TO1:NEXT:GOTO6
150 IFVB<>16ORN0<>23ORLC<>16THEN154
151 X=23:GOSUB224:IFY<>-1PRINTM1$:GOTO6
152 OB(11,1)=0:OB(14,1)=16:CR=CR-1:PRINTM2$
153 GOTO6
154 IFVB<>18ORN0<>36THEN158
155 IFNO<>36ORLC<>31THEN21
156 X=17:GOSUB224:IFY<>-1PRINTM3$:GOTO6
157 HE$(31)**:DJ=-1:PRINTM4$:GOTO6
158 IFVB<>19ORN0=0THEN169
159 IFNO<>34PRINT* THAT'S STUPID!* :GOTO6
160 IFLC<>35PRINTM5$:GOTO6
161 X=22:GOSUB224:IFY<>-1PRINTM6$:GOTO6
162 IFNO=35PRINTM7$:GOTO6
163 IFNO<>34PRINTN0$(0);MB$:GOTO6
164 IFTC>MDPRINTM9$:GOTO6

```

```

165 FORI=1TOLO:IFOB(I,0)=34THEN166ELSENEXTI:GOTO21
166 K=I:FORI=1TOLO:IFOB(I,0)=22THEN167ELSENEXTI:GOTO21
167 PRINTN0$:HE$(35)="
168 OB(K,1)=0:OB(I,1)=0:FORI=1TO1:NEXTI:GOTO6
169 IFVB<>17THEN171
170 PRINTN0$(0):GOTO6
171 IFVB<>20THEN180
172 IFNO=0THEN21
173 X=NO:GOSUB224:IFY=-1PRINT"I'M CARRYING IT. THAT'S IMPOSSIBLE.":GOTO6
174 IFY=LCTHEN177
175 IFNO<11ORNO=19ORNO=20ORNO=30THEN21
176 PRINT"I CAN'T HIT SOMETHING I CAN'T SEE.":GOTO6
177 IFNO=15ORNO=25ORNO=34PRINT"I'D RATHER NOT. HE MIGHT HIT ME BACK!":GOTO6
178 IFNO=35PRINT"THAT'S NOT NICE!":GOTO6
179 GOTO22
180 IFVB<>21THEN181ELSEPRINT"I'M NOT STRONG ENOUGH TO KILL ANYTHING.":GOTO6
181 GOTO21
182 CLS:PRINTD$(LC):A$=""
183 IFLC=35GF=10
184 IFLC=7THEN190
185 K=0:FORI=1TOLO:IFOB(I,1)<>LCTHEN188
186 IFK=0THENK=1:PRINT:PRINT"AROUND ME I SEE.":A$=OB$(I):GOTO188
187 IFLEN(A$)+5+LEN(OB$(I))>SLTHENPRINTA$:A$=OB$(I)ELSEA$=A$+" "+OB$(I)
188 NEXTI
189 IFAS<>"PRINTA$
190 PRINT:PRINT"OBVIOUS DIRECTIONS ARE ";:K=0
191 FORI=0TO5:IFD$(LC,I)=0THEN194
192 IFK<>0PRINT", ";
193 PRINTN0$(I+1);:K=1
194 NEXTI
195 IFK=0PRINT"UNKNOWN";
196 PRINT", ":GOTO6
197 CLS:GOSUB220
198 IFJ=0PRINT"WE HAVE FAILED OUR MISSION. THE FORCES OF PRINCESS ALEYA WILL
  BE CONQUERED.
  "
199 IFJ=SCPRINT"WE ARE HEROS.
  THE FORCES OF PRINCESS ALEYA WILL CONQUER THE EVIL VAGG          SOLDIERS, AND FR
  EEDOM WILL PREVAIL ON OUR HOME PLANET OF          HARTLEY AND THROUGHOUT THE GALAX
  Y THEY WILL SAY THAT EVIL          DOES NOT PAA."
200 IFJ=0ANDJ<SCPRINT"WE HAVE HELPED THE FORCES OF PRINCESS ALEYA DEFEND THE
  GALAXY. LONG LIVE THE FORCES OF FREEDOM!
  "
201 PRINT
202 INPUT"DO YOU WANT TO PLAY AGAIN (Y OR N)";A$:A$=LEFT$(A$,1)
203 IFAS="Y"THEN2ELSEEND
204 CLS:PRINT"A VOICE BOOMS OUT, ";CHR$(34);"WHO GOES THERE";CHR$(34)
205 GOSUB211
206 IFVB<>13ORNO<>30THEN104
207 PRINTN1$
208 GOSUB211
209 X=31:GOSUB224:IFY<>-1THEN104
210 IFVB<>17ORNO<>31THEN5575ELSE182
211 INPUT"WHAT SHOULD I DO";CMS
212 VB$(0)="":NO$(0)="":VB=0:NO=0:IFLEN(CMS)=0RETURN
213 FORZL=1TOLEN(CMS):IFMID$(CMS,ZL,1)<>" "THENVB$(0)=VB$(0)+MID$(CMS,ZL,1):NEXT
  ZL
214 FORZL=1TOLO:IFVB$(ZL)<>" "ANDLEFT$(VB$(0),LEN(VB$(ZL)))=VB$(ZL)THENVB=ZL:GOTO
  216ELSENEXTZL
215 VB=0:NO$(0)=VB$(0):GOTO217
216 IFLEN(VB$(0))+1>LEN(CMS)THENNO=0:RETURNELSENO$(0)=RIGHT$(CMS,LEN(CMS)-1-LEN
  (VB$(0)))
217 FORZL=1TOLO:IFNO$(ZL)<>" "ANDLEFT$(NO$(0),LEN(NO$(ZL)))=NO$(ZL)THENNO=ZL:GOTO
  219ELSENEXTZL
218 NO=0:RETURN
219 FORZL=1TO1:NEXTZL:RETURN
220 J=0:FORI=1TOLO:IFOB(I,1)=1THENJ=J+OB(I,2)
221 NEXTI:PRINT"OUT OF A MAXIMUM OF";SC;"POINTS, YOU HAVE";J;"POINTS."
222 IFJ=0PRINT"WE'RE NOT DOING TOO GOOD."
223 RETURN
224 FORZL=0TOLO:IFOB(ZL,0)=XTHENY=OB(ZL,1):GOTO225ELSENEXTZL:Y=-99:RETURN
225 FORZL=1TO1:NEXTZL:RETURN
226 A$=INKEY$
227 IFINKEY$=""THEN227ELSEReturn

```

230 LV=21:DIMVB\$(LV):VB\$(1)="GO":VB\$(2)="GET":VB\$(3)="LOOK":VB\$(4)="INVEN":VB\$(5)  
 )="SCORE":VB\$(6)="DROP":VB\$(7)="HELP":VB\$(8)="SAVE":VB\$(9)="LOAD":VB\$(10)="QUIT"  
 :VB\$(11)="PRESS":VB\$(12)="SHOOT":VB\$(13)="SAY":VB\$(14)="READ":VB\$(15)="EAT":VB\$(  
 16)="CSAVE"

231 VB\$(17)="SHOW":VB\$(18)="OPEN":VB\$(19)="FEED":VB\$(20)="HIT":VB\$(21)="KILL"

232 LN=37:DIMN\$(LN):N\$(1)="NORTH":N\$(2)="EAST":N\$(3)="SOUTH":N\$(4)="WEST":N  
 0\$(5)="UP":N\$(6)="DOWN":N\$(10)="BUTTON":N\$(11)="TAG":N\$(12)="FUEL":N\$(13)=""  
 BLASTER":N\$(14)="COMMUNICATOR":N\$(15)="GUARD":N\$(16)="MAP":N\$(17)="KEYS":N\$(  
 18)="NECKLACE"

233 N\$(19)="SESAME":N\$(20)="GRAFFITI":N\$(21)="CAPE":N\$(22)="HAMBURGER":N\$(2  
 3)="TAPE":N\$(24)="TURBO":N\$(25)="SCIENTIST":N\$(26)="PLANS":N\$(27)="SCHEMATIC  
 ":N\$(28)="DEVICE":N\$(29)="GUN":N\$(30)="SECURITY":N\$(31)="I.D.":N\$(32)="CRYSTALS"

234 N\$(33)="SIGN":N\$(34)="ROBOT":N\$(35)="PRINCESS":N\$(36)="DOOR":N\$(37)="AM  
 MUNITION"

235 CL=37:DIMDS\$(CL):DS\$(1)="I'M IN THE PASSENGER & STORAGE COMPARTMENT OF MY SP  
 ACE SHIP.  
 THERE'S AN EXIT HERE TO LEAVE THE SHIP.":DATA2,0,0,0,0,3:DS\$(2)="I'M IN THE COCK  
 PIT OF MY SPACESHIP.  
 A LARGE RED BUTTON SAYS >> PRESS TO BLAST OFF <<":DATA0,0,1,0,0,0  
 236 DS\$(3)="I'M STANDING NEXT TO MY SPACE SHIP WHICH IS LOCATED ON A  
 HUGE FLIGHT DECK.":DATA18,0,4,0,1,0:DS\$(4)="I'M OUT ON THE FLIGHT DECK OF GENERA  
 L VAGG'S BATTLE CRUISER.":DATA3,5,4,4,0,0:DS\$(5)=DS\$(4):DATA4,6,5,4,0,0  
 237 DS\$(6)="I'M IN A HALLWAY. THERE ARE DOORS ON ALL SIDES.  
 THE DOOR TO THE NORTH SAYS: >> CLOSED FOR THE DAY <<":DATA7,0,8,5,0,0:DS\$(7)="I'  
 M IN THE SUPPLY DEPOT.  
 AROUND ME I SEE:  
 ALL KINDS OF THINGS":DATA0,0,6,0,0,0  
 238 DS\$(8)="I'M AT THE END OF ONE OF THE HALLWAYS.  
 I CAN HEAR VOICES NEARBY. SOUNDS LIKE GUARDS.":DATA6,10,0,9,0,12:DS\$(9)="I'M IN  
 THE STRATEGY PLANNING ROOM.":DATA11,8,0,0,0,0:DS\$(10)="I'M IN THE DECONTAMINATIO  
 N AREA.":DATA0,14,0,8,0,0  
 239 DS\$(11)="THIS AREA IS THE TRACTOR BEAM CONTROL ROOM.  
 A LARGE SIGN WARNS: >> DO NOT PRESS ANY BUTTONS <<":DATA0,0,9,0,0,0:DS\$(12)="I'M  
 IN ANOTHER HALLWAY. TO THE EAST IS A RESTROOM.":DATA15,13,0,0,8,0  
 240 DS\$(13)="THIS IS WHAT IS COMMONLY CALLED, THE BATHROOM.  
 THERE'S GRAFFITI WRITTEN ALL OVER THE WALL.  
 PIPES LEAD UP THROUGH THE CEILING.":DATA15,0,0,12,27,0:DS\$(14)="THIS APPEARS TO  
 BE AN INTERROGATION ROOM.":DATA0,0,0,10,0,0:DS\$(15)="I'M IN A LOUNGE."  
 241 DATA0,0,13,12,0,0:DS\$(16)="THIS IS A COMPUTER ROOM. THERE'S A TRS-80 IN HERE

ON THE SCREEN IT SAYS: >>CSAVE TAPE <<":DATA17,0,18,0,0,0:DS\$(17)="I'M IN A TEST  
 ING LABORATORY.":DATA0,0,16,0,0,0  
 242 DS\$(18)="I'M IN A HALLWAY  
 A LARGE ARROW POINTS EAST AND SAYS: >> TO THE VAULT <<":DATA16,25,3,19,0,0:DS\$(1  
 9)="THIS IS THE ENTRANCE TO THE DEVELOPMENT LAB SECTION":DATA20,18,21,20,22,0  
 243 DS\$(20)="I'M IN A LONG CORRIDOR. THERE ARE LABORATORIES ALL AROUND ME.":DATA1  
 9,23,21,20,22,24:DS\$(21)="I'M IN A RESEARCH LAB.":DATA20,0,0,0,0,0:DS\$(22)="I'M  
 LOST":DATA22,22,22,22,22,20:DS\$(23)=DS\$(21):DATA0,0,0,20,0,0:DS\$(24)=DS\$(21):DAT  
 A0,0,0,0,20,0  
 244 DS\$(25)="I'M NEAR THE ENTRANCE TO THE VAULT.  
 A SIGN HERE SAYS: >> AUTHORIZED PERSONEL ONLY <<":DATA0,26,0,18,0,0:DS\$(26)="I'M  
 IN THE VAULT.":DATA0,0,0,25,0,0:DS\$(27)="I'M IN A PIPE TUNNEL WHICH LEADS IN EV  
 ERY DIRECTION.":DATA20,27,27,27,27,13  
 245 DS\$(28)=DS\$(27):DATA29,29,29,29,30,29:DS\$(29)="I'M LOST IN A MAZE OF PIPES."  
 :DATA20,29,29,29,29,27:DS\$(30)="I'M IN THE PIPE MAZE.  
 BELOW ME I THINK I CAN SEE THE JAIL.":DATA29,29,28,29,29,31:DS\$(31)="I'M IN THE  
 JAIL.":DATA32,33,34,35,0,0  
 246 DS\$(32)="I'M IN A JAIL CELL.":DATA0,0,31,0,0,0:DS\$(33)=DS\$(32):DATA0,0,0,31,  
 0,0:DS\$(34)=DS\$(32):DATA31,0,0,0,0,0:DS\$(35)="I'M AT THE SECURITY DESK.  
 TO THE NORTH AN ELEVATOR.":DATA36,31,0,0,0,0:DS\$(36)="I'M IN THE ELEVATOR.":DATA  
 0,0,35,0,37,0  
 247 DS\$(37)=DS\$(36):DATA0,0,14,0,0,36  
 248 LO=23:DIMOB\$(LO):OB\$(1)="A TAG WHICH SAYS: >> NEEDS TURBO <<":DATA11,5,0:OB\$(  
 2)="ANTI-MATTER FUEL":DATA12,5,5:OB\$(3)="BLASTER":DATA13,7,0:OB\$(4)="COMMUNICAT  
 OR":DATA14,9,0:OB\$(5)="A VERY SUPRISED GUARD":DATA15,9,0:OB\$(6)="MAP OF THE SHIP  
 ":DATA16,29,20  
 249 OB\$(7)="SOME KEYS":DATA17,9,0:OB\$(8)="A SHINESTONE NECKLACE":DATA18,10,20:OB  
 \$(9)="PRINCESS ALEYA'S CAPE":DATA21,14,5:OB\$(10)="HAMBURGER":DATA22,15,0:OB\$(11  
 )="A CASSETTE TAPE":DATA23,7,0:OB\$(12)="A TURBOENCABULATOR":DATA24,17,5  
 250 OB\$(13)="AN EVIL LOOKING SCIENTIST":DATA25,17,0:OB\$(14)="SECRET ATTACK PLANS  
 ":DATA26,0,20:OB\$(15)="DEATH RAY SCHEMATIC":DATA27,9,20:OB\$(16)="CLOCKING DEVICE  
 ":DATA28,17,20:OB\$(17)="MICRO LASER GUN":DATA29,24,20:OB\$(18)="I.D. CARD":DATA31  
 ,17,0

```

251 OB$(19)="MALIDIUM CRYSTALS (THE TREASURY!)":DATA32,26,30:OB$(20)="A SIGN WHI
CH SAYS: >> OUT OF ORDER <<":DATA33,3,0:OB$(21)="ATTACK ROBOT":DATA34,35,0:OB$(2
2)="PRINCESS ALEAYA":DATA35,34,50:OB$(23)="AMMUNITION":DATA37,7,0
252 DIMD$(CL,5):RESTORE:FORZL=1TOCL:FORZD=0TO5:READD$(ZL,ZD):NEXTZD:NEXTZL
253 DIMOB(L0,2):FORZL=1TOLO:READOB(ZL,0),OB(ZL,1),OB(ZL,2):NEXTZL
254 DIMHE$(CL):HE$(1)="I THINK WE'RE SUPPOSED TO LEAVE THE STUFF HERE.":HE$(2)="
I WONDER IF WE HAVE ENOUGH FUEL?":HE$(7)="HOW ABOUT A BLASTER.":HE$(9)="TRY SHOO
T GUARD.":HE$(13)="IT MIGHT BE INTERESTING TO READ THE GRAFFITI.":HE$(17)="TRY S
HOOT SCIENTIST."
255 HE$(22)="I'M AS CONFUSED AS YOU ARE.":HE$(29)=HE$(22):HE$(31)="IT MIGHT HELP
IF WE HAD SOME KEYS TO OPEN ANY LOCKED DOORS.":HE$(35)="DID YOU BRING ANYTHING
TO EAT?":M0$="I CAN'T GO THERE. THE DOOR IS LOCKED.":M1$="I'M NOT CARRYING ANY B
LANK TAPE."
256 M2$="THE TRS-80 RECORDED SOMETHING ON THE TAPE, AND THEN IT PRINTED:
>> ATTACK PLANS -- VERY SECRET <<":M3$="I CAN'T. I'M NOT CARRYING ANY KEYS.":M4$
="O.K. THE DOOR TO THE JAIL CELL IS UNLOCKED.":M5$="THERE'S NO ROBOT HERE."
257 M6$="BUT I DON'T HAVE ANY HAMBURGERS.":M7$="MUNCH...MUNCH BURP!
THE PRINCESS THANKS YOU FOR A DELICIOUS MEAL.":M8$="DOES'NT EAT HAMBURGER.":M9$=
"NOTHING HAPPENED. THE HAMBURGER IS COLD YOU KNOW.":N0$="THE ATTACK ROBOT EATS T
HE HAMBURGER AND DISSAPEARS."
258 N1$="I'M AT THE IDENTIFICATION TERMINAL.
ON THE SCREEN IT SAYS: >> SHOW I.D. <<":N2$="THE TRACTOR BEAM IS OFF.":N3$="THE
TRACTOR BEAM IS ON.":N4$="YOU FORGOT TO OPEN THE FLIGHT DECK DOORS."
259 RETURN

```

- 0000000000 -

#### XMAS CARD

by G.C. HODGSON L2/16K

As you type in this program you will wish that you had ordered a cassette subscription! But persevere, it is well worth the effort. When you have finished, you will be treated to the most magnificent graphics that we at MICRO-80 have ever seen on an '80. No more need be said, the program says it all.

```

10 REM *** A XMAS CARD *** BY GORDON HODGSON * COFFS HARBOUR *
20 REM *** INDEX -- 40-70 GRAPHICS CODE 80-270 XMAS TREE
                280-470 RUDOLPH          480-670 PRANCER
                680-860 SLEIGH          870-1140 SANTA
    *** N.B. TURN UP BRIGHT CONTROL FOR BEST EFFECT ***
30 REM <<< TO PERSONALIZE NAMES EDIT LINE 1070 >>>
40 CLS: CLEAR(100): DEFSTR A-X
50 A=CHR$(128): B=CHR$(129): C=CHR$(130): D=CHR$(131): E=CHR$(132): F=CHR$(133): G=CHR
$(135): H=CHR$(136): I=CHR$(138): J=CHR$(139): K=CHR$(140): L=CHR$(141): M=CHR$(142): N
=CHR$(143): O=CHR$(144): P=CHR$(145): Q=CHR$(146): R=CHR$(147): S=CHR$(148): T=CHR$(14
9): U=CHR$(159)
60 V=CHR$(160): W=CHR$(170): X=CHR$(171): A1=CHR$(175): B1=CHR$(176): C1=CHR$(177): D1
=CHR$(178): E1=CHR$(179): F1=CHR$(180): G1=CHR$(184): H1=CHR$(186): I1=CHR$(187): J1=C
HR$(188): K1=CHR$(189): L1=CHR$(190): M1=CHR$(191): N1=CHR$(173): O1=CHR$(172)
70 P1=CHR$(168): Q1=CHR$(163): R1=CHR$(181): S1=CHR$(156): T1=CHR$(151): U1=CHR$(157)
: V1=CHR$(183): W1=CHR$(174): X1=CHR$(161): AA=CHR$(162): BB=CHR$(137): CC=CHR$(167)
80 REM *** XMAS TREE ***
90 PRINT@96,M1;
100 PRINT@159,U;J1;0;
110 PRINT@218,M1;A;A;V;G1;M1;M1;M1;F1;0;A;A;A;M1;
120 PRINT@282,D;A;D1;M1;M1;M1;U;D;R;D;D;B1;B1;R;
130 PRINT@345,K;STRING$(7,188);D1;D1;B1;M1;M1;M1;N;D;
140 PRINT@406,M1;A;K;N;N;M1;U;N;N;N;STRING$(7,191);E1;0;A;A;M1;
150 PRINT@470,B1;B1;B1;J1;J1;S;A;O1;M1;S1;A;A;N;M1;M1;N;D;H;E;B1;J1;J1;B1;B1;
160 PRINT@530,M1;A;A;A;A;D;J;N;N;N;F;A;D;B1;B1;STRING$(5,188);A;N;F;D;E1;B1;
170 PRINT@596,J;M1;M1;M1;J1;J1;J1;J1;M1;M1;B1;A;D;D;D;N;N;N;N;M1;M1;M1;M1;U1;R;B
;A;A;M1;
180 PRINT@656,K;W1;STRING$(5,191);U;N;D;D;E1;D;D;J;M1;M1;M1;J1;J1;J1;J1;STRING$(
7,176);;J1;J1;B1;B1;
190 PRINT@722,B1;E1;I1;M1;T1;A;B1;J1;N;D;N;J1;B1;A;J;N;N;M1;M1;M1;M1;D;E1;D;N;N;
D;D;
200 PRINT@789,D;B;A;A;D;N;J1;N;D;A;A;J1;J1;STRING$(5,128);L1;E1;K1;
210 PRINT@864,M1;M1;
220 PRINT@923,D;A1;STRING$(8,191);U;D;
230 PRINT@989,J;STRING$(6,191);G;
240 PRINT@32,"*";:PRINT@154,"*";:PRINT@167,"*";:PRINT@342,"*";:PRINT@364,"*";:PR
INT@466,"*";:PRINT@560,"*";

```

```

250 FORZ=1T0500:NEXT:PRINT@128,"W H A T ' S";:PRINT@193,"T H I S";:PRINT@256,"I
  S E E ? .....";:FORZ=1T01500:NEXT:PRINT@176,"A X M A S";:PRINT@241,"T R E E ?
  .....";:FORZ=1T01000:NEXT
260 FORY=1T020:NEXT:PRINT@32," ";:PRINT@154,"*";:PRINT@167,"*";:PRINT@342," ";:PRINT@
364,"*";:PRINT@466,"*";:PRINT@560," ";:FORZ=1T025:NEXTZ
270 PRINT@32,"*";:PRINT@154," ";:PRINT@167," ";:PRINT@342,"*";:PRINT@364," ";:PR
INT@466," ";:PRINT@560,"*";:FORZ=1T025:NEXTZ,Y
280 CLS:          REM *** RUDOLPH ***
290 PRINT@35,U;A;G1;F;O;W;S;
300 PRINT@99,M1;P;P1;C;L;G;G1;E;W;F1;
310 PRINT@164,Q1;E1;R;D;J;L;C;N;B;
320 PRINT@229,D;D;L;V;L1;M1;M1;E1;M1;STRING$(5,188);
330 PRINT@295,P1;STRING$(6,191);U;N;D;D;B;
340 PRINT@341,P1;STRING$(17,128);C;A1;STRING$(5,191);R1;
350 PRINT@405,A1;K1;O;STRING$(15,128);V;H1;STRING$(6,191);K1;O;
360 PRINT@470,D;C1;J1;STRING$(9,191);G;P;G;C1;L1;STRING$(10,191);
370 PRINT@534,STRING$(10,188);S1;A;F1;R;B;E1;E1;E1;O1;E1;D;E1;R;D;D;B;
380 PRINT@598,J;STRING$(7,191);T1;P;E;C1;A;K;A;M1;M1;M1;B1;C1;G1;B1;D1;B1;M1;G;
  B;
390 PRINT@659,B1;J1;J1;J1;STRING$(5,191);N;H;E;K;N;N;E;N;E;J;A1;M1;M1;U;N;D;D;E1
;G1;J1;B1;
400 PRINT@723,M1;U;B;U;J1;J1;J1;K;STRING$(11,128);W;M1;U;STRING$(5,128);D;D;N;A1
;K1;
410 PRINT@787,A1;K1;A;I;M1;T1;STRING$(13,128);L1;M1;B;STRING$(8,128);W;M1;
420 PRINT@851,W;M1;A;A;A1;T;STRING$(12,128);U;M1;F;STRING$(9,128);H1;U;
430 PRINT@914,I;G;A;A;A;I;M1;STRING$(11,128);U;L1;T1;STRING$(9,128);D;D;
440 PRINT@984,I;L;STRING$(10,128);C;N;E;
450 FORZ=1T0100:NEXT:PRINT@0,"... A N D";:PRINT@64,"H E R E ' S";:PRINT@128,"R U
  D O L P H ...";:FORZ=1T01500:NEXT:PRINT@192,"... T H E ";:PRINT@256,"R E D - N
  O S E D";:PRINT@320,"R E I N D E E R ...";
460 FORZ=1T030:NEXT:PRINT@243,"*";:PRINT@341,S;:PRINT@404,I;M1;F1;A;:PRINT@340,A;A;:P
RINT@404,J;K1;F1;:PRINT@340,A;:PRINT@403,C;N;J1;B1;:PRINT@339,A;A;:PRINT@403,D;:
FORY=1T020:NEXTY
470 PRINT@339,A;A;:PRINT@403,C;N;J1;B1;:PRINT@340,A;A;:PRINT@403,A;J;K1;F1;:PRIN
T@340,A;S;:PRINT@404,I;M1;F1;A;:PRINT@341,P1;:PRINT@404,A;A1;K1;O;:PRINT@243," "
;:FORY=1T020:NEXTY:NEXTZ
480 REM *** PRANCER ***
490 CLS:PRINT@200,"H E R E C O M E S .....";:FORZ=1T0500:NEXT:PRINT@330,"P R
  A N C E R !!!!";
500 PRINT@48,K;J1;G1;J1;A;P1;S1;
510 PRINT@114,I;M1;A;G1;S;O;G1;E;
520 PRINT@179,J;N1;M1;;R1;J;M1;F1;
530 PRINT@243,B1;A;C;D;L;F1;Q;M1;F;
540 PRINT@307,C;D;N;N;E;D1;C1;P;
550 PRINT@370,B1;B1;J1;J1;STRING$(6,191);J1;B1;B1;
560 PRINT@429,B1;G1;J1;STRING$(6,191);N;N;D;D;D;D;
570 PRINT@471,B1;B1;B1;A;A;A;V;B1;O;A;V;B1;B1;B1;J1;J1;J1;M1;M1;A;H1;STRING$(6,1
91);U;N;D;
580 PRINT@531,D;D;N;N;N;G;D;H;M;N;N;G;V;L1;STRING$(7,191);U;B;H;N;N;N;D;E1;C1;J1
;
590 PRINT@601,W;M1;M1;M1;T;A;A1;STRING$(7,191);G;A;G1;STRING$(10,191);S1;
600 PRINT@656,B1;G1;STRING$(5,188);A;P1;STRING$(5,191);B1;C;D;N;N;D;D;A;P1;J1;M1
;M1;M1;U;G;D;D;D;E1;I1;N;D;
610 PRINT@709,B1;S1;U;B1;B1;J1;S1;M;N;D;D;A;D;Q1;C1;B1;B1;J1;STRING$(5,191);S
TRING$(5,143);G;I;G;H;N;D;Q1;R;B1;B1;B1;B1;S1;CHR$(167);C1;
620 PRINT@772,D;D;A;D;D;B;STRING$(8,128);G1;M1;M1;N;N;N;D;D;D;STRING$(12,128);
I;F;D;B;A;V;G1;G;
630 PRINT@847,B1;J1;N;D;STRING$(23,128);M1;I;N;D;
640 PRINT@903,U;B1;O;A;B1;J1;J1;N;D;
650 PRINT@966,D;D;B;A;D;D;
660 PRINT@444,CHR$(39);
670 FORZ=1T03000:NEXT
680 CLS:          REM *** SLEIGH ***
690 PRINT@3,U;B1;O;
700 PRINT@69,J1;B1;STRING$(20,128);STRING$(7,176);
710 PRINT@132,H1;G;Q1;E1;E1;E1;B1;B1;STRING$(5,128);J1;M1;M1;J1;R;D;D;D;N;A1;M
1;M1;G;E1;G1;J1;J1;G;E1;E1;X1;
720 PRINT@193,B1;O;A;A;J1;M1;N;N;M1;N;Q1;E1;C1;B1;O;A;A;D;D;B;STRING$(7,128);D;X
1;L1;U;D;C1;J1;O;A;V;Q;
730 PRINT@257,D;X1;G1;F1;R;A;B1;J1;J1;B1;M1;M1;M1;M1;U;L;K;STRING$(9,128);P1;K1;
B1;E1;G1;M1;J1;A;K;M;B;A;B;E;
740 PRINT@321,C;N;M1;M1;N;L;E;D;STRING$(5,143);G;A;N;N;STRING$(5,128);J1;M1;M1;K
1;S;X;STRING$(9,191);C;G1;G;A;B1;B1;O;;P1;J1;F1;A;A;B1;B1;A;B1;U;O;B1;U;O;B1;U;O
;B1;

```

```

750 PRINT@384,P1;J1;J1;J1;M1;M1;STRING$(10,143);M1;M1;J1;B1;A;J;F1;C;A1;M1;M1;K1
:F1;Q;E1;M1;M1;M1;M1;U;G;B;M;N;A;A;N;N;K1;Q;A1;M1;T;I;E1;K;M1;K;G;
760 PRINT@451,P1;STRING$(12,188);B1;A;D;N;M1;K1;C;J;O1;D1;J;N;M1;K1;STRING$(7,18
B);U;D;G1;M1;M1;M1;F;B1;B1;B1;A;K;D;
770 PRINT@516,STRING$(5,143);STRING$(10,131);A;C;A1;K1;O;A;Q1;E1;C1;G1;J1;J1;B1;
STRING$(5,131);A;A1;M1;M1;U;B;W1;M1;M1;N;N;B;STRING$(9,128);B1;J1;J1;J1;B1;
780 PRINT@580,A1;STRING$(15,191);K1;A;X;M1;B1;A;D;Q1;E1;B1;J1;J1;J1;M1;M1;M1;M1;
J1;J1;B1;B1;J1;J1;O;A;B1;J1;J1;B1;STRING$(8,128);M1;M1;D;X;M1;M1;
790 PRINT@645,X;STRING$(15,191);K1;O;C;A1;J1;B1;A;J;STRING$(7,191);A1;M1;M1;M1;N
;N;N;N;A;M1;M1;N;O1;C1;J1;E;A;B1;J1;A;W;M1;T;
800 PRINT@710,A1;STRING$(16,191);J1;B1;C;J;A1;J1;B1;E1;D;N;N;N;N;A;Q1;B1;J1;J1;J
1;M1;M1;T;A;O1;L1;U;G;C1;L1;M1;N;B;A;W;M1;T;
810 PRINT@775,C;N;A1;STRING$(16,191);J1;J1;B1;B1;E1;D;N;K;K;K;A;I;N;N;D;E1;E1;E1
;C1;A;A;A;C;D;D;D;B;A;G1;M1;M1;
820 PRINT@843,C;STRING$(5,131);STRING$(20,143);A;J;M1;N;N;E1;;E1;J1;F1;B1;O;A;A;
A;A;B1;J1;M1;M1;D;
830 PRINT@909,K;K;K;STRING$(17,128);K;K;K;A;A;A;A;;I;M1;M1;D;N;N;G;D;E1;B1;J1;M1
;M1;N;D;
840 PRINT@966,STRING$(6,191);STRING$(32,188);L1;M1;N;N;N;D;D;
850 PRINT@228,"*";
860 FORZ=1T0200:NEXT:PRINT@11,"... AND SANTA ON HIS SLEIGH !!...";:FORZ=1T04000:
NEXT:PRINT@109,"THAT'S RIGHT !!!";:FORZ=1T0500:NEXT:PRINT@173,STRING$(16,"*");:P
RINT@237,"IT'S XMAS TIME !!!";:PRINT@301,STRING$(16,"*");:FORZ=1T04000:NEXT
870 CLS:      REM *** SANTA ***
880 PRINT@27,B1;B1;A;B1;B1;A;A;A;A;O1;J1;J1;J1;J1;F1;B1;B1;B1;B1;
890 PRINT@83,B1;B1;A;B1;J1;J1;STRING$(9,191);J1;J1;B1;C;D;N;STRING$(7,191);K1;F1
;
900 PRINT@143,U;J1;J1;STRING$(20,191);J1;O;A;X;STRING$(6,191);T;
910 PRINT@207,L1;STRING$(5,191);N;N;D;D;D;D;E1;E1;D;E1;E1;D;D;N;STRING$(5,191);S
;W;STRING$(5,191);T;
920 PRINT@271,I;A1;M1;U;D;G1;L1;O;I;J1;J1;M1;M1;U;B;C;N;N;N;K1;B1;J;M1;M1;G;V;B1
;D;N;N;A1;M1;T;
930 PRINT@337,D;V;L1;M1;M1;B;A;G1;J1;J1;O;A;A;G1;J1;F1;A;A1;K1;D1;R;A;J;N;B;I;
L;A;J;K1;O;A;A;U;B1;O;
940 PRINT@399,J;K;M;M1;M1;N;F;A;A;A;A;B1;B1;O;A;A;A;C;D;N;M1;F1;E1;D;B;F1
;H;J1;A;A;A;G1;M1;M1;M1;M1;J1;O;
950 PRINT@458,G1;L1;P;A;A;A;K;K;N;D;U;J1;A;A;D;E1;C1;J1;E;B;O1;F1;E1;R;B;U;J1;B1
;A;N;D;D;A;W;F;H1;M1;B;A;C;A1;M1;N;A1;G;
960 PRINT@521,I;M1;M1;M1;J1;J1;J1;J1;M1;M1;B1;J;N;M1;M1;U;D;A;K;K;E;C;J;N;A1;
M1;U;F;G1;M1;M1;M1;F1;Q;I;G;B;
970 PRINT@585,N1;STRING$(11,191);STRING$(5,188);L1;STRING$(5,191);K1;J1;J1;J1;J1
;L1;STRING$(5,191);U;
980 PRINT@650,J;A1;;STRING$(31,191);O;
990 PRINT@716,U;I1;STRING$(30,191);
1000 PRINT@781,C;D;J;N;N;N;N;D;D;A;H1;STRING$(20,191);F;
1010 PRINT@851,B1;J1;J1;STRING$(12,191);A1;STRING$(6,191);N;D;
1020 PRINT@914,D;N;STRING$(11,191);N;N;A;C;D;D;A;D;B;
1030 PRINT@983,STRING$(6,131);
1040 PRINT@29,"*";:PRINT@34,"*";:PRINT@35,"*";:PRINT@113,"*";:PRINT@369,"*";:PRI
NT@409,"*";:PRINT@416,"*";
1050 PRINT@1,STRING$(9,"*");:PRINT@193,STRING$(9,"*");:PRINT@65,"*";:PRINT@129,"
*";:PRINT@73,"*";:PRINT@137,"*";
1060 PRINT@67,"MERRY";:PRINT@132,"XMAS";:PRINT@770,"HAPPY";:PRINT@834,"NEW";:PRI
NT@898,"YEAR";
1070 PRINT@623,"TO * ALL";:PRINT@692,"* OUR";:PRINT@756,"* READERS";:PRINT@820
,"* EVERYWHERE";
1080 PRINT@768,"*";:PRINT@776,"*";:PRINT@832,"*";:PRINT@840,"*";:PRINT@896,"*";:
PRINT@904,"*";
1090 FORZ=1T0100:PRINT@409," ";:PRINT@416," ";:PRINT@29," ";:PRINT@34," ";:PRINT
@35," ";:PRINT@113," ";:PRINT@369," ";:PRINT@1,STRING$(9," ");:PRINT@193,STRING$
(9," ");:PRINT@65," ";:PRINT@129," ";:PRINT@73," ";:PRINT@137," ";
1100 PRINT@768," ";:PRINT@776," ";:PRINT@832," ";:PRINT@840," ";:PRINT@896," ";:
PRINT@904," ";
1110 SET(55,25):SET(61,25):PRINT@409,"*";:PRINT@416,"*";:PRINT@29,"*";:PRINT@34,
"*";:PRINT@35,"*";:PRINT@113,"*";:PRINT@369,"*";:PRINT@1,STRING$(9,"*");:PRINT@1
93,STRING$(9,"*");:PRINT@65,"*";:PRINT@129,"*";:FORZ=1T05
1120 PRINT@73,"*";:PRINT@137,"*";:PRINT@768,"*";:PRINT@776,"*";:PRINT@832,"*";:P
RINT@840,"*";:PRINT@896,"*";:PRINT@904,"*";:RESET(55,25):RESET(61,25)
1130 NEXTY,Z
1140 GOT01140

```

## KEYBOARD BLEEPER, KEY DEBOUNCER AND ALARM by C.E.KENDALL L11/4K

This program can be used to give an audible bleep on each keystroke, it also provides key debounce. To hear the bleep, connect an audio amplifier and speaker to the lead which normally goes to the "AUX" socket on the cassette recorder.

\*EDITORS NOTE: I have discovered that if you remove the "EARPHONE" plug, take the lead which normally goes to the "AUX" socket and plug it in to the socket marked "MIC" (remove the tape if present), press the PLAY key and turn up the volume, the sound will come through the cassette recorder's speaker. This also works for both the previously published sound programs. It works on a CTR-80, I don't know if it will work on other recorders, (try it).

The machine language part of the program is POKEd into an apparently unused part of RAM in Level II BASIC, so you do not need to set any memory size and your keyboard will still bleep even after you have typed "NEW" and got rid of the BASIC part of the program, (you have to run it first of course.)

It gives a high-pitched bleep for the normal character keys and a lower pitched bleep for "ENTER", "SPACE", "BREAK" and the arrows. Two subroutines in the program give examples of other uses for the program.

The keyboard debounce routine is also POKEd into the same area of memory and will also remain in action after "NEW" is typed.

(Note: this program is written specifically for the TRS-80. Perhaps one of our readers who owns a SYSTEM 80/VIDEO GENIE will modify it to use cassette port #-2. We would be happy to publish an amended version. - ED.)

```

10 FOR J=16435 TO 16437: READ I: POKE J,I: NEXT
20 FOR J=16480 TO 16513: READ I: POKE J,I: NEXT
30 POKE 16405,0
40 DATA 195,96,64,205,227,3,183,200,8,14,20,68,62,1,211,255,16,254,68,62,2,211,2
55,16,254,13,32,239,8,201,205,127,10,205,101,64,201
50 ' THE DURATION OF THE NOTE IS SET BY THE 30 IN LINE 40.
    IF IT IS MADE TOO LARGE, THE KEYBOARD ACTION WILL BE
    TOO SLOW.
199 ' SAMPLE LINE TO CALL ALARM SUBROUTINE
200 INPUT X: FOR Q=0 TO 1000: NEXT: GOSUB 2000: GOTO 200
999 ' USE THIS SUBROUTINE TO CALL A SERIES OF BLEEPS FROM
    A PROGRAM. TO VARY THE PITCH, VARY X BETWEEN 32 AND
    127. Z CONTROLS THE INTERVAL BETWEEN BLEEPS.
1000 POKE 16526,123: POKE 16527,64: X=X*256: FOR N=0 TO 10: A=USR(X): FOR Z=0 TO
    100: NEXT Z,N: RETURN
1999 ' SUBROUTINE TO SOUND ALARM UNTIL ANY KEY IS PRESSED.
2000 POKE16526,123:POKE16527,64:X=X*256
2010 FORZ=0TOQ20:NEXT:R=USR(X):A$=INKEY$:IFA$=""THEN2010ELSE200
2020 RETURN

```

- 000000000 -

## THE CASE OF THE DEAD CURSOR.

by E. R. Paay

Looked at your cursor lately? Doesn't look very exiting does it? Very hard to find in certain listings and screen displays. What we need is a flashing cursor like some of the bigger machines have. Many BASIC programmers realize this and create a flashing cursor in their programs using INKEY\$ statements in loops, which then have to be used for all input from the keyboard. BASIC being what it is, this means slow keyboard response and large amounts of BASIC memory are wasted. It was for these reasons that I wrote a flashing cursor program in machine language, some time ago. The program is very simple but we will discuss several features of it for interest's sake.

First of all, there is the initialization sequence. This sets memory size without you having to answer MEMORY SIZE? or READY?. Many readers have asked how this can be done, so I will now explain. The sequence runs from line 290 to 340. The HL register pair should contain the required memory size. Disk BASIC users can of course use location 4049H for TOPMEM instead of 40B1H and drop lines 300 to 340 incl.. Location 40B1H is the top of memory pointer for BASIC and 40A0H labeled SPOINT in the listing is the string space pointer. BASIC uses this to point to space where it can operate on strings, when string type commands are used.

The main routine intercepts the keyboard and maintains a counter. The counter is decremented every time the keyboard is scanned by BASIC, which is done at regular intervals. As soon as this counter reaches zero the count is reset and the cursor is turned off if on and turned on if it is off. If a character comes from the keyboard the counter is reset anyway and a flag is set, causing the normal BASIC cursor to be overwritten immediately by our new cursor when the keyboard is next scanned. The position of the cursor is maintained by the interpreter at two bytes located at 4020H and 4021H, this location I labelled CURLOC (CURsor LOCation).

The source listing provided is for a 16K machine. All that is required for machines with different memory capacities is to change the ORG statement in line 70. Beginner Assembly language programmers might like to have a close look at the source listing which I feel (even if I say so myself) is a good example of what a good professional assembly language program should look like. All addresses have meaningful labels (to me anyway). The use of numeric unlabelled addresses and numbers has been kept to a minimum. A program written this way should not need many remarks as the labels should tell what is being done.

The second program presented is a demo program in BASIC (yes I do occasionally write BASIC programs). It allows the user to change the blink rate and use any character for a cursor. The current values for cursor and delay count used, are displayed and the user is prompted for new values, this allows experimentation with different cursors and flash rates. On a level II machine when a flash rate of 32 is used the cursor will be somewhat like the Electric Pencil cursor and appear to be transparent. Some of the lines in the BASIC listing given appear not to have any line numbers in front of them. In fact, these lines are part of the line(s) above them. I use the down arrow key (CTRL for SYSTEM-80) to start new statements on the next line. This makes it easy to read the program when it is listed on the video display and avoids the cluttered, unreadable type of program listing you get when multistatement lines are typed in to full capacity using no spaces. It only adds one byte to the program so is very economical (except of paper! - Ed.)

The machine language program may be entered via an Editor/Assembler, in which case you should ignore the first two columns which contain the object code. Having typed in the information from column 3 on you then set the ORIGIN to the position you require, Assemble the source code and punch out the object code to cassette or disk. Alternatively, you could enter the object code directly using a monitor. BMON would not be satisfactory for this purpose, however, because the two programs reside at the same memory location. The answer is to attach a block move routine to the program, as described in the September issue or to use Tandy's TBUG to enter the program.

Entering the BASIC program is, of course, quite straightforward but it should be realised that this program is intended to work in conjunction with the machine language program and they must both be in the computer at the same time.

```

00010 ; CURSOR DRIVER By E. R. Paay
00020 ;
00030 ;      25-10-1980.
00040 ;
00050 ;      Copyright (C) MICRO-80
00060 ;
7FA0 00070      ORG      7FA0H
00080 ;
00090 ; Define labels.
00100 ;
06CC 00110 BASRDY EQU 06CCH
03E3 00120 KBSCAN EQU 03E3H
4016 00130 KBVEC  EQU 4016H
4020 00140 CURLOC EQU 4020H
40B1 00150 TOPMEM EQU 40B1H
008C 00160 CURSOR EQU 008CH
0020 00170 CURSOF EQU 0020H
0230 00180 MAXCNT EQU 0230H
40A0 00190 SPOINT EQU 40A0H
1B6E 00200 SETPTR  EQU 1B6EH
00210 ;
00220 ; Initialize.
00230 ;
7FA0 F3 00240 INIT    DI
7FA1 21BB7F 00250      LD      HL,START
7FA4 221640 00260      LD      (KBVEC),HL      ;reset kbvector
7FA7 2B      00270      DEC     HL
7FA8 2B      00280      DEC     HL
7FA9 22B140 00290      LD      (TOPMEM),HL      ;set memory size
7FAC 113200 00300      LD      DE,32H
7FAF B7      00310      OR      A
7FB0 ED52   00320      SBC     HL,DE
7FB2 22A040 00330      LD      (SPOINT),HL      ;set string pointer
7FB5 CD6E1B 00340      CALL   SETPTR      ;set all other pointers
7FB8 C3CC06 00350      JP      BASRDY      ;jump to basic ready mode
00360 ;
00370 ; Main routine.
00380 ;
7FB8 3AFF7F 00390 START  LD      A,(FLAG)
7FBE B7      00400      OR      A
7FBF 2004    00410      JR      NZ,CONT

```

```

7FC1 D9      00420      EXX
7FC2 CDD87F  00430      CALL   SETCUR      ;suppress normal cursor
7FC5 CDE303  00440 CONT    CALL   KBSCAN      ;scan keyboard
7FC8 D9      00450      EXX
7FC9 B7      00460      OR     A
7FCA 2023    00470      JR     NZ,RESCNT
7FCC 2AFD7F  00480      LD     HL,(COUNT)
7FCF 7C      00490      LD     A,H
7FD0 B5      00500      OR     L
7FD1 3E00    00510      LD     A,0
7FD3 2806    00520      JR     Z,SETCUR      ;check counter
7FD5 2B      00530      DEC   HL
7FD6 22FD7F  00540      LD     (COUNT),HL  ;decrement counter
7FD9 D9      00550      EXX
7FDA C9      00560      RET
00570 ;
00580 ; This section flashes cursor.
00590 ;
7FDB 2A2040  00600 SETCUR LD     HL,(CURLOC) ;get cursor location
7FDE 7E      00610      LD     A,(HL)
7FDF FE20    00620      CP     CURSOF      ;cursor on?
7FE1 3E20    00630      LD     A,CURSOF
7FE3 2005    00640      JR     NZ,RESCUR      ;yes! then turn it off
7FE5 3E8C    00650      LD     A,CURSOF      ;no! then turn it on
7FE7 32FF7F  00660      LD     (FLAG),A
7FEA 77      00670 RESCUR LD     (HL),A
7FEB AF      00680      XOR   A
7FEC F5      00690      PUSH  AF
7FED 1805    00700      JR     FIXCNT
00710 ;
00720 ; This section initializes delay counter.
00730 ;
7FEF F5      00740 RESCNT PUSH  AF
7FF0 AF      00750      XOR   A
7FF1 32FF7F  00760      LD     (FLAG),A
7FF4 213002  00770 FIXCNT LD     HL,MAXCNT
7FF7 22FD7F  00780      LD     (COUNT),HL
7FFA F1      00790      POP   AF
7FFB D9      00800      EXX
7FFC C9      00810      RET
7FFD 0000    00820 COUNT DEFW  0      ;delay counter
7FFF 00      00830 FLAG  DEFB  0
7FA0 00      00840      END   INIT
000000 TOTAL ERRORS
10 ' *****
   **          TRS - 80  CURSOR DRIVER DEMO          **
   **          BY : EDWIN R. PRAY                    **
20 ' **          (C) 25 - OCT - 1980                  **
   *****
30 '
40 CLS:
   PRINT*          CURRENT CURSOR CHAR. = ";PEEK (32742);"
   CURRENT DELAY COUNT = ";PEEK (32757) + (PEEK (32758) * 256):
   INPUT*

   ENTER ASCII CODE FOR CURSOR (DEC)
   ";CURSR
45 IF CURSR > 255
   THEN GOTO 40
   ELSE INPUT*

   ENTER FLASH RATE (MAX = 65535)
   ";RATE
50 IF RATE > 65535
   THEN PRINT @ 384,"":
   GOTO 45
   ELSE IF RATE > 32767 THEN RATE = NOT (65535 - RATE)
60 POKE 32742, CURSR:
   POKE 32757, RATE AND 255:
   POKE 32758, (RATE / 256) AND 255
70 GOTO 40
80 END

```

\*\*\*\*\* NEXT MONTH'S ISSUE \*\*\*\*\*

Next month's issue will contain at least the following programs plus the usual features, articles, news, letters, etc.. In addition, there will be a comprehensive index of all material published in the first 12 issues of MICRO-80.

\*\* ALPHABETICAL LISTING - L1 \*\*

Once you have this program, your telephone index will be obsolete - well, you'll have more fun looking for a phone number, at least.

\*\* MUGWUMP - L1 \*\*

This is a "find that animal" game played on a 10 x 10 board. Find the four hidden mugwumps before your 10 guesses are up.

\*\* PINBALL MACHINE - L2 \*\*

This is one pinball machine that you can't tilt! If only you could find a way to charge 20c for 5 shots, this program would make your fortune.

\*\* BOMBING RAID - L2 \*\*

Bloodthirsty lot, aren't we? This is a Level II bombing game . Just so you Level IIers won't feel left out of all this warmongering.

\*\* WORLD TIME - L2 \*\*

Just the shot for the world traveller or trader. Ever wondered when you should ring those friends/relations who live on the other side of the world? They will love hearing from you but not at 4.30 am! This program will solve all that. All you need to know is your own time and whether or not you are on daylight saving time and away it goes. You would be hard put to it to find a country which is not catered for by this program.

\*\* 741 ACTIVE FILTER - L2 \*\*

If you ever have occasion to design an active filter then this is the program for you. All you need do is define whether you want a high-pass or low-pass filter, the corner frequency and the computer does the rest. It designs the filter and then draws the circuit on the screen, complete with component values.

\*\*\*\*\* APPLICATION FOR PUBLICATION OF A PROGRAM IN MICRO-80 \*\*\*\*\*

Tick where appropriate.

To MICRO-80

Please consider the enclosed program for...

- (i) Publication in MICRO-80 .....
- (ii) Publication on disk or cassette only .....
- (iii) Both .....

Name.....

Address .....

.....

.....Post Code.....

\*\*\*\* CHECK LIST \*\*\*\*

Please ensure that the cassette or disk is clearly marked with your name and address, program name(s), Memory size, Level I, II, System 1 or 2, Edtasm, System etc. The use of REM statements with your name and address is suggested, in case the program becomes separated from the accompanying literature.

Ensure that you supply adequate instructions, notes on what the program does and how it does it...etc.

For system tapes, the start, end, and entry points, etc.

The changes or improvements that you think may improve it.

Please package securely - padabags are suggested - and enclose stamps or postage if you want your cassette or disk returned.





# MICRO-80

## LEVEL II ROM REFERENCE MANUAL

by Edwin Paay

Published by MICRO-80 PRODUCTS

Written by Eddy Paay, the LEVEL II ROM REFERENCE MANUAL is the most complete explanation of the Level II BASIC interpreter ever published.

Part 1 lists all the useful and usable ROM routines, describes their functions explains how to use them in your own machine language programs and notes the effect of each on the various Z 80 registers.

Part 1 also details the contents of system RAM and shows you how to intercept BASIC routines as they pass through system RAM. With this knowledge, you can add your own commands to BASIC, for instance, or position BASIC programs in high memory—the only restriction is your own imagination!

Part 2 gives detailed explanations of the processes used for arithmetical calculations, logical operations, data movements, etc. It also describes the various formats used for BASIC, SYSTEM and EDITOR/ASSEMBLER tapes. Each section is illustrated by sample programs which show you how you can use the ROM routines to speed up your machine language programs and reduce the amount of code you need to write.

The LEVEL II ROM REFERENCE MANUAL is intended to be used by machine language programmers. It assumes a basic understanding of the Z 80 instruction set and some experience of Assembly Language programming. But BASIC programmers too will benefit from reading it. They will gain a much better insight into the functioning of the interpreter which should help them to write faster, more concise BASIC programs.

# MICRO-80