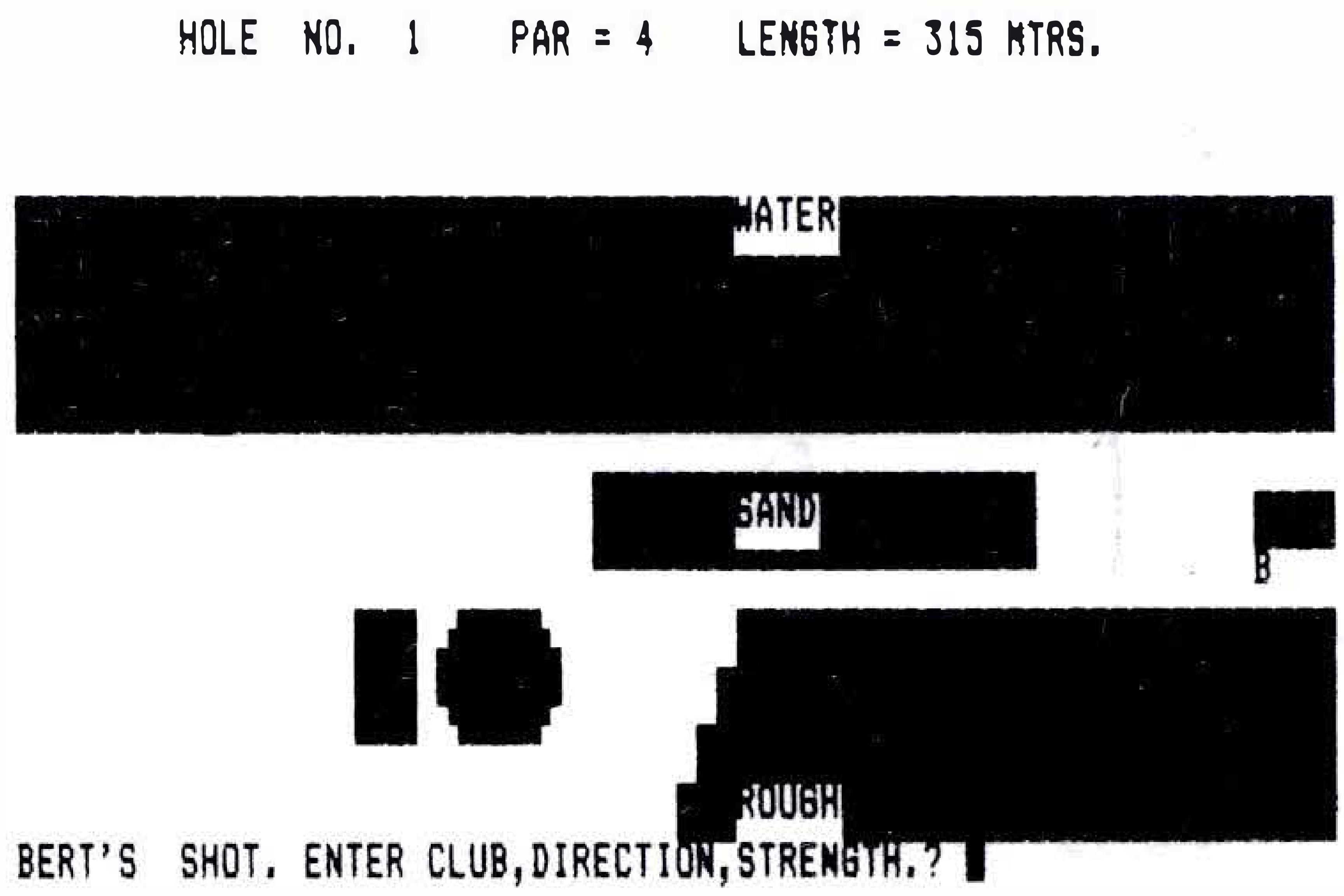
MICRO-80

P.O. BOX 213, GOODWOOD, S.A. 5034. AUSTRALIA. TELEPHONE (08) 211 7244. PRICE: AUS. $2.50, N.Z. $4.00, U.K. £1.50
Registered by Australia Post — Publication No. SBQ2207

Vol. 3, Issue 12, July 1983

Level II
GOLF



```
HOLE  NO.  1    PAR = 4    LENGTH = 315 MTRS.

BERT'S  SHOT. ENTER CLUB,DIRECTION,STRENGTH.? 
```

Also in this issue:

PROGRAMMING:
Theory and Techniques of Sorting — Part 7

REVIEWS:
Astrology

HARDWARE:
Double your System-80 RAM for $15

SOFTWARE:

- Anagrams — Level II
- Cleanup — Level I
- E = MC² — Level II
- S.A. Horse Performance Guide — Level II
- Mastermind — Colour
- Sink the Enemy Navy — Colour

• TRS-80 • SYSTEM 80 • VIDEO GENIE
• PMC-80 • HITACHI PEACH
• TRS-80 COLOUR COMPUTER

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

EDITOR:	RYSZARD WIWATOWSKI
ASSOCIATE EDITORS:	
SOFTWARE :	CHARLIE BARTLETT
HARDWARE :	EDWIN PAAY

MICRO-80 is an international magazine devoted to the Tandy TRS-80 Model I, Model III and Colour microcomputers, the Dick Smith System 80/Video Genie and the Hitachi Peach. It is available at the following prices:

	12 MONTH SUB.	SINGLE COPY
MAGAZINE ONLY	$ 26-00	$ 2-50
CASSETTE PLUS MAGAZINE	$ 65-00	$ 4-00 (cass. only)
DISK PLUS MAGAZINE	$125-00	$10-00 (disk only)

MICRO-80 is available in the United Kingdom from:

U.K. SUBSCRIPTION DEPT. 24 Woodhill Park, Pembury, Tunbridge Wells, KENT TN2 4NW

MAGAZINE ONLY	£ 16-00	£ 1-50
CASSETTE PLUS MAGAZINE	£ 43-60	£ N/A
DISK PLUS MAGAZINE	£ 75-00	£ N/A

MICRO-80 is available in New Zealand from:

MICRO PROCESSOR SERVICES, 940A Columbo Street, CHRISTCHURCH 1 N.Z. Ph 62894

MAGAZINE ONLY	NZ$ 43-00	NZ$ 4-00
CASSETTE PLUS MAGAZINE	NZ$ 89-00	NZ$ 5-00
DISK PLUS MAGAZINE	NZ$175-00	NZ$15-00

MICRO-80 is despatched from Australia by airmail to other countries at the following rates:

(12 MONTH SUB)	MAGAZINE	CASS + MAG	DISK + MAG
PAPUA NEW GUINEA	Aus$40-00	Aus$ 83-00	Aus$ 143-00
HONG KONG/SINGAPORE	Aus$44-00	Aus$ 88-00	Aus$ 148-00
INDIA/JAPAN	Aus$49-00	Aus$ 95-00	Aus$ 155-00
USA/MIDDLE EAST/CANADA	Aus$55-00	Aus$102-00	Aus$ 162-00

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/Video Genie or Peach and its peripherals. MICRO-80 is in no way connected with any of the Tandy, Dick Smith or Hitachi 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 microcomputer to earn some extra income is included in every issue.

** CONTENT **

Each month we publish at least one applications program in BASIC for each of the microcomputers we support. We also publish Utility programs in BASIC and Machine Language. We publish articles on hardware modifications, constructional articles for useful peripherals, articles on programming techniques both in Assembly Language and BASIC, new product reviews for both hardware and software and we print letters to the Editor.

** COPYRIGHT **

All the material published in this magazine is under copyright. This 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 *****

MICRO-80 is Registered by Australia Post - Publication No. SBQ2207

AUSTRALIAN OFFICE AND EDITOR:
MICRO-80
P.O. BOX 213
GOODWOOD, SOUTH AUSTRALIA, 5034.
TEL. (08) 211 7244

U.K. SUBSCRIPTION DEPT:
24 WOOHILL PARK, PEMBURY,
TUNBRIDGE WELLS, KENT TN2 4NW

Printed by:
Shovel & Bull Printers,
379 South Road,
MILE END, SA., 5031

Published in Australia by:
MICRO-80
433 Morphett Street,
Adelaide, SA., 5000

***** EDITORIAL *****

For some time now, Tandy has been offering discounts and monthly specials on may of its computer products. The Models 2 and 3, in particular, are substantially discounted from their regular price. It is interesting to note that the new Models 12 and 4, intended to replace the Models 2 and 3 respectively, have recently been released in the U.S. Evidently, Tandy in Australia is preparing to upgrade its whole computer range in the near future and the customer can take advantage of lower prices on current stock.

The new Model 4 (released in the last week of April) comes in a white case like the Models 12 and 16. It features 64K RAM (with 128K available as an option), two single-sided, double density 40 track drives, an 80 x 24 display and an improved keyboard. The basic hardware is apparently very similar to that of its predecessor and all of the available Model 3 software should run in a 'Model 3 mode'. The base Model 4 is priced at $US1,999 and comes with TRSDOS 6.0 which is actually the first implementation of the RAM based version of LDOS.

There are rumours that CP/M 3.0 (or CP/M Plus) will soon be available for the Model 4. Not only does this make the Model 4 more attractive from the buyer's point of view, but it will avoid the unfortunate situation in which Model 16 owners have found themselves - advanced and powerful hardware but with no software available to make use of it. The ability to run the latest banked version of CP/M and hence, to access the large established base of CP/M software will be a strong selling point for the Model 4. Also, Model 3 owners can convert their machines into Model 4's with an upgrade kit that sells for about $US800.

The Dick Smith organisation has already expanded its computer line to include the new compact VZ-200 which was announced via a rather glowing review in the April issue of APC magazine. With a low $199 price tag, this machine is aimed at the large domestic market currently contested by the Commodore VIC-20, the Sinclair Spectrum, the Colour Genie and the Tandy Colour Computer. But at such a low price, it will undoubtedly take a big share of this market. I imagine it could even be used by electrical retailers to increase the sales of colour television sets.

Like Tandy, Dick Smith's has also been offering special price reductions on the System-80 and associated peripherals in the past few months. A company spokesman informed us that the Hong Kong manufacturer has ceased producing the System-80 and that when currentstocks are sold, it will be no longer available. However, Dick Smith's will continue to provide support for the System 80 in the technical, software and service areas.

Reports from some of our readers indicate that the new version of DOSPLUS (3.5) is being shipped to owners of 3.4 who have taken advantage of the upgrade offer made by Micro-Systems Software. I have had a glimpse of this new version and must say that I am impressed. DOSPLUS 3.5 bears a strong resemblance in overall design and structure to LDOS, but this is not surprising when you consider that Tandy has adopted LDOS 5.1 as one of its product line.

- 0000000000 -

***** PEEKing (UK) - by Tony Edwards *****

The latest problem that has been troubling the U.K. software industry is the program library. For a long time programs, mostly games programs, have been swapped between friends and in local computer clubs, but recently this has become more organized. Clubs have been setting up formal libraries of tapes for short term loan, and now national tape library services have been set up. These are run just like normal libraries in that tapes containing programs can be borrowed for short periods for a small fee. Unlike book libraries, there is not usually any royalty payment to the owner of the copyright of the program. On the face of it, under English law, this is quite legal if the tapes are obtained by normal commercial means.

The problem is what the borrowers do with the tapes once they have them on loan. The program libraries claim that the borrowers test the programs against their needs and, if they are satisfactory, they buy copies for themselves when the borrowed tapes are due for return. It is difficult to believe that this is what actually happens in this imperfect world. We all know how easy it is to copy a program once it is available on tape. Many programs exist which will copy a tape by reproducing the signal, thus making an identical copy, anti-copy tricks and all. In the final case, it is not difficult to make a tape to tape copy using two recorders. The software writers claim that borrowers simply copy the programs they like and never need to buy an original. With the difficult legal position with regard to the copyright of programs in this country, it is difficult to stop such goings on. One company of software producers started legal moves against a program library, but the case never came to court as the library agreed that all its members should sign an undertaking not to copy library tapes. Another large software house now sells its programs with the written warning that they must not be used in a library service. No doubt we have not heard the end of this story, but until the law is clarified by a test case, problems like this one will continue to bug the software world.

- 0000000000 -

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

From: Robert Pinna - Lathlain, W.A.

Having just recently purchased a TRS-80 colour computer, I would like to know if there is any provision for the future to provide us (unlucky owners of TRS-80 C/C) with a similar service as you have for TRS-80 LI & III.

I suggest you have a cassette for C/C even if the charges would have to be a bit higher. As you well know, typing those programs from the magazine is not everybody's cup of tea.

(We are currently looking into this possibility and would very much like to offer a similar service to our Colour Computer readers. -Ed.)

From: J.C. Haseldine - Mulgrave, Vic.

I have a TRS-80 Model I Level II, CTR-80 cassette recorder, a Tandy Model VII printer, a LNW 48K interface and two MPI B51 single density disk drives. I am especially interested in using this equipment in association with my amateur radio equipment for CW, RTTY ASCII purposes and would appreciate any information re software and interfacing which you may have. Perhaps a series of articles in MICRO-80 would be interesting to other people.

With best wishes for the continued success of MICRO-80.

(Thank you for the kind words. In the October, 1980 issue we published an article entitled:

"Computerised RTTY-Mactronics and the System 80" by Ron Collins.

We have some other material on file that will be published in a future issue and would welcome more as a number of our readers are interested in this subject. -Ed.)

From John Tinney - Mulgrave, Vic.

I must say how much I have enjoyed previous tapes that I have had from you - particularly the Dr. Who Adventure, despite the couple of minor bugs in it.

The OM error can be easily overcome by increasing the CLEAR statement in line 64Ø. We fixed it by making the line read:

64Ø CLEAR 85Ø: etc.

The other error is in line 84Ø where the null input statement takes you to line 14Ø instead of returning you for another try to INPUT A$.

However, as I said, these bugs are minor. The program itself is intriguing, infuriating and everything that an adventure program should be. Please - may we have more of them!!?

(I'm glad to hear you enjoyed this adventure, but I'm puzzled by your fix for the Out of Memory error. Normally one would clear more string space after encountering an Out of String space error during program execution. The problem some readers were experiencing was that the main program could not even be loaded into memory, despite the fact that the program should fit into 16K of memory, and I can't see how changing line 64Ø would cure it.

The second bug that you mention is a definite error and I have included it in this month's Microbugs column. It should, of course, be 84Ø instead of 14Ø. -Ed.)

From: Grant Barnes - Moe, Vic.

I recently purchased the adventure game of Asylum and I'm going out of my mind trying to find the professor's office. If you or any of your readers could help me I would much appreciate it.

(There are those among us who can help but, sadly, I am not one of them. -Ed.)

From: Richard Siggs - Fulham, S.A.

I have a copy of the "TRS-80 Adventure" from Microsoft. It has slowly been driving me mad. Does anybody know of a way to get to the "shadowy figure which seems to be attracting your attention" after going west from "Y2"? Can anybody help me?

(At last, something familiar! Many months ago I spent many hours on this adventure and regarded myself as a seasoned adventurer. But, alas, one thing or another cropped up and I've not revisited it since. However, I do remember that I began to feel much more confident when a friend and I began to map the entire cave and, in particular, the two mazes - a procedure I strongly recommend. Unfortunately, these maps are no more, but I can tell you to regard such statements with caution because some things are not necessarily what they seem. - Ed.)

From: Mr. I. Vanco - Herston, Qld.

Thank you for your recent letter and advice regarding my problem of loading EDTASM + produced SYSTEM tapes. Your publication of my letter resulted in a Tandy Technican who resides in my street coming to my aid. The problem was solved by a critical Head Alignment on the recorder.

(You're welcome! Perhaps other readers, who have had difficulties may find this will solve their problems too. - Ed.)

- 0000000000 -

DEPARTMENTS

***** KALEIDOSCOPE *****

For Colour Computer readers this month we have two programs - Sink the Enemy Navy and Mastermind. As well, we have some technical information that will be of interest.

In low memory, there are four locations that contain the pointers to the start and end of the BASIC program currently resident in memory. The pointer to the start of the BASIC program is kept in locations 25 and 26 and the pointer to the end, in locations 27 and 28. Actually, two must be subtracted from the end of program pointer value to get the true end of the program. To print these values to the screen, type in the following:

```
Start of Program  -   PRINT 256*PEEK(25)+PEEK(26) (enter)
End of Program    -   PRINT 256*PEEK(27)+PEEK(28) - 2 (enter).
```

When you wish to make a back-up copy of a machine language program, you need to know the start, end and execution addresses of the program. After you CLOADM the program, these addresses are contained in the following locations on cassette systems:

Start Address at 487,488
End Address at 126,127 (must subtract 1 from this value).
Execute Address at 157,158.

If the program is not auto-starting, then you can print these values in the same way as for the BASIC pointers. Those of you who cut your teeth on the Model I will notice that the 6809 stores two-byte values with the most significant byte first, followed by the least significant byte second (whereas the Z80 stores them the other way around).

You can speed-up your BASIC programs by using POKE 65495,0. This can be used in the program itself, but it may upset the timing of any input or output operations. In these cases, the POKE 65494,0 will restore normal operation. A combination of the two can be used quite effectively, for example, to speed up screen displays and long calculations.

- 0000000000 -

***** PEACH BOWL *****

During the preparation of this month's Peach programs (Sink the Enemy Navy & Mastermind), we discovered some interesting features of the pseudo-random number generator. The RANDOMIZE statement, which allows you to choose the random number seed, was very useful in the testing of the programs because you can set it to the same value for each test and the same sequence of random numbers will be generated. This makes it very easy to retrace your steps after you've encountered a bug related to a 'random' value.

On the other hand, the RND function is more limited than on other machines. For example, RND(18) doesn't return an integer in the range 0-18, but instead gives value between 0 and 1. This can be a source of mysterious problems, especially if you are used to a machine that does the former. The simple solution of INT (19*RND(18)) doesn't seem quite so elegant in comparison.

In last month's Input/Output column, one of our Peach readers, Mr. John Wardley, reported problems in using the serial interface at 4800 baud for his printer. He has since informed us that by using the 300 baud rate, he has managed to overcome the loss of data. As well, he sent us this interesting little program (of which we were not aware):

```
10 SCREEN 0,1
20 FOR X = 4 TO 9
30 READ Y: POKE & HFFC6,X : POKE &HFFC7,Y
40 NEXT X
50 POKE &HA5,&H14 : POKE &H23A,50 : POKE &H23C,49
60 DATA 63, 6, 50, 55, &H83, 6
```

With a high resolution monitor and NEW ON7, this program produces a 50 line display - a format that is useful, as John points out, for long listings and de-bugging programs.

We have tried it and found that it works nicely on our disk based Peach from a cold start but doesn't always produce the desired result if other programs have been run. Also, it doesn't seem to work on a cassette - based Peach, even from a cold start. Perhaps our readers can look into this one and enlighten us all.

- 0000000000 -

***** GROUP ONE *****

T his month we have four Level II programs and one Level I program (which can be used on a Level II machine with the aid of Level I in Level 2 from the Free Software Library). The S.A. Horse Performance Guide is only suitable for 16K Level II cassette machines. The Golf program can be used on a disk system provided the following changes are made.

```
10 DATA205,127,10,125,217,1,0,4,254,1,40,8,17,0,184,33,0,60
20 DATA24,6,17,0,60,33,0,184,237,176,217,201
60 DEFUSR=LM
```

These changes have already been made to the file GOLF/BAS on the distribution disk.

Here are some hints sent in by Mr. D. Brenton, one of our readers:

The keyboard can be scanned by using PEEKs. The table below tells you which 'address to PEEK', while the 'value returned' tells you which key was being pressed.

Value			Address to PEEK					
RETURNED	14337	14338	14340	14344	14352	14368	14400	14464
	3801	3802	3804	3808	3810	3820	3840	3880
1	@	H	P	X	0	(or 8	(ENTER)	(SHIFT)
2	A	I	Q	Y	1	) or 9	(CLEAR)	
4	B	J	R	Z	2	* or :	(BREAK)	
8	C	K	S		3	+ or ;	[↑	
16	D	L	T		4	< or ,	\ ↓	
32	E	M	U		5	= or -	] ←	
64	F	N	V		6	> or ,	^ →	
128	G	0	W		7	? or /	(SPACE BAR)	

Continuous PEEKing can be used to provide auto-repeats.

To control your BREAK key in LEVEL II BASIC, POKE these values into these addresses (ADDRESS/VALUE)

16396/201			-	enable
16396/175			-	disable
16396/62	16397/ASC	16398/0	-	return a character with an ASCII value of ASC (ASC/128 returns keyword even if not displayed).
16396/165	16397/198	16398/ASC	-	same as above, but when SHIFT is also held down, character returned in ASC+1.
16396/195	16397/LSB	16398/MSB	-	jump to address (MSB*256+LSB) eg. to reset the system, LSB=0, MSB=0

D id you know that there are 5 extra symbols that can be accessed by your keyboard other than the normal characters? These can be accessed by holding down 'Z' & '2' (press ← to get the rid of the excess characters, then press a number from 3 to 7. the following symbols can be produced: ↑ ↓ ← → _

- 0000000000 -

***** FORM THREE *****

The information regarding programs and hints given above is also relevant to the Model III with the exception of the last tip. The method for entering extra characters from the keyboard does not work because of a different decoding algorithm.

Some of our Model III disk subscribers have reported difficulties reading the distribution disks. These cannot be read directly by the Model III because the distribution DOS is NEWDOS 2.1 and the disk format is 35 track single density Model I. However, the files can be quite easily moved to a Model III format disk by one of the following methods.

TRSDOS and DOSPLUS users can CONVERT the files across to a Model III format disk with the utilities provided on the system disk. Newdos 80 V2.0 users will have to do a little more work. Firstly, place the distribution disk in drive 1 and do the following;

(1) PDRIVE 0 6,TI=A, TD=A, TC=35, SPT=10, TSR=3, GPL=2, DDSL=17, DDGA=2 (the PDRIVE specification for a Model I disk - only needs to be done once).

(2) PDRIVE 0 1=6,A
(Model I disks can now be read in drive 1 but not the directory).

(3) WRDIRP 1
(to read and rewrite the directory sectors with the correct address mark for Model III Newdos).

The distribution disk can now be used in the normal way. The files can either be copied to a Model III format disk or simply run from the distribution disk.

A good idea is to save the original drive 1 PDRIVE specification in an unused slot before doing this, e.g.:

```
PDRIVE 0 7 = 1
```

The system can then be quickly restored to its original state by:

```
PDRIVE 0 1 = 7,A
```

A word of warning - NEWDOS can transfer all types of files from Model I disks including system files and DOS utilities. These are of no use on the Model III and may produce disastrous effects if you attempt to run them.

- 00000000000 -

PROGRAMMING

***** THEORY AND TECHNIQUES OF SORTING - PART 7 *****

by Bernie Simson

All the previous articles in this series concentrated on algorithms for purely sorting a list of values, whether numeric or alphanumeric, while the list resided in the main memory of the computer. It may now be obvious to those who have been following the articles that to merely sort a list of values in main memory based on some predetermined collating sequence (Ordering Rule) by whatever algorithm, is of no practical use to the user who wants to produce a report of, say, debtors in descending order of Amount Owed within ascending order of Postcode, from a Debtors Masterfile that contained, among other vitals, the Amount Owed and Postcode of the debtor. Such a report would be useful to an organization having regional collection officers. So, assuming the Debtors File is in Debtor code sequence, an internal memory sort of a variety presented in the preceding article will have to be augmented with some other mechanisms to produce a report in the desired sequence. This leads to the discussion on Record Sorting and External Sorting Techniques.

It is necessary to consider Record Sorting Techniques because in most applications, data to be sorted is organized as records.

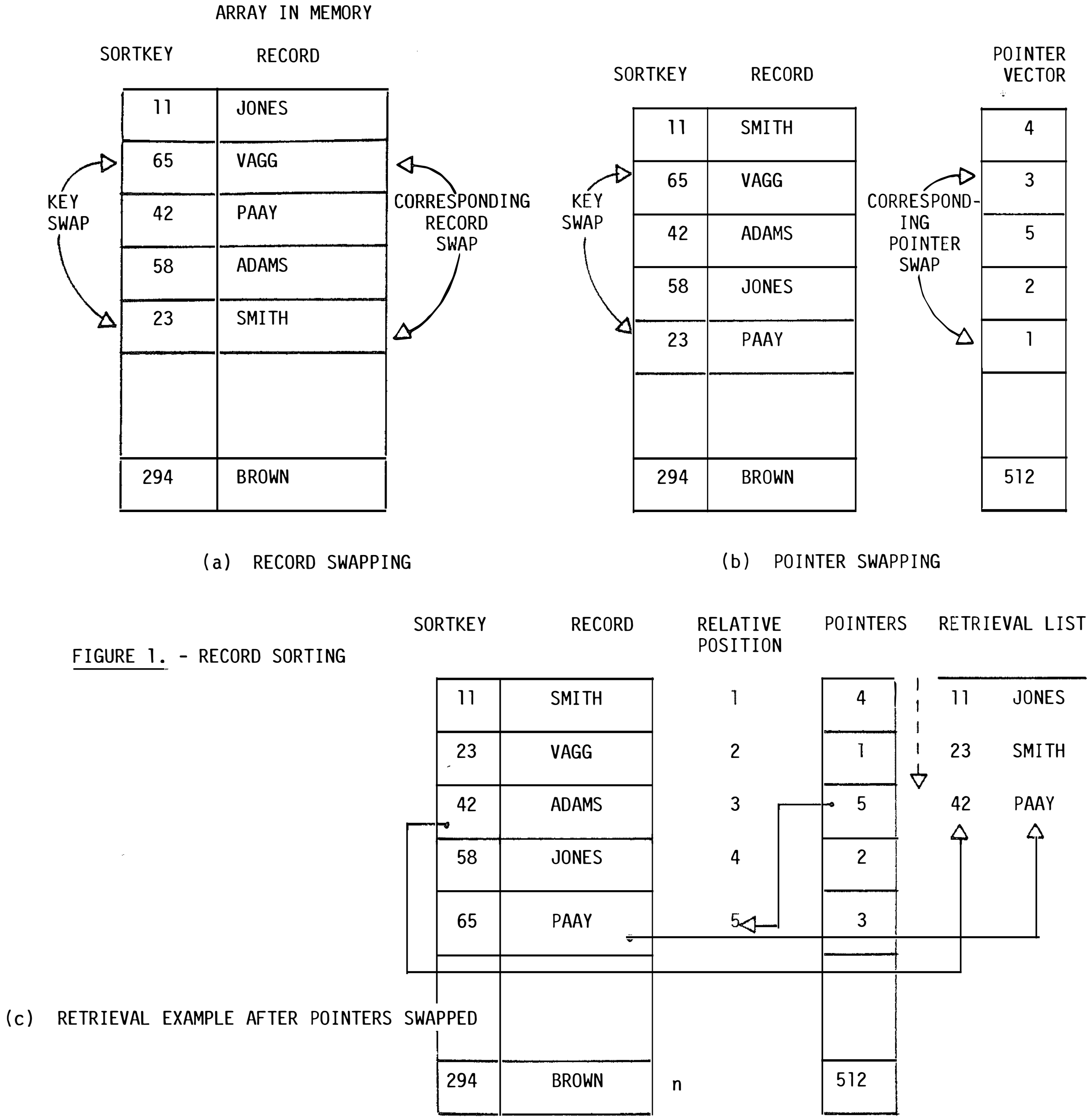
External Sorting Techniques are necessary when the complete file, or its complete list of keys, is too big to fit in main memory all at once, which, in most applications, is usually the case.

RECORD SORTING

It is assumed that a Disk system is available. Sorting records that will not all fit in main memory in a Tape-based system is...well, er, shall we say a little cumbersome!

Also assume at this point that all the records of the input file will fit in main memory. Figure 1 shows two techniques for sorting records. The records are read from Disk into an array, comprising a column containing the keys upon which ordering is to occur, and a column for the rest of the record. If the key and the rest of the record are of different Types (e.g. numeric & alpha), then two arrays are used. It does not matter as far as the algorithm is concerned.

In the first technique, the key column is considered as a list of items, and is sorted using one of the many Internal Sorting Algorithms in Articles 1 to 6 of this series, except that when two Keys are swapped in the sort, their corresponding records are also swapped, so that the



(a) RECORD SWAPPING

(b) POINTER SWAPPING

(c) RETRIEVAL EXAMPLE AFTER POINTERS SWAPPED

FIGURE 1. - RECORD SORTING

relationship between the key and the record is not lost. This can become quite a processing overhead, especially if the rest of the record is quite long.

In the second technique, a pointer vector is also introduced, initialized to start at some predetermined point (1 in this example) so that each element value is one greater than the previous element value. Then the key column is sorted as before, except that when two keys are swapped in the sort, instead of the corresponding records being swapped, the associated pointer vector items are swapped. The pointer vector thereby becomes an index into the record, so when an ordered list of the records is required, the key vector is read sequentially, using the pointer vector index to access the rest of the record, as demonstrated in Figure 1.

This latter technique of record sorting is more efficient, but at a cost of additional storage in main memory of the pointer vector. However, this is a small cost compared to the overhead of swapping the rest of the record on key swap.

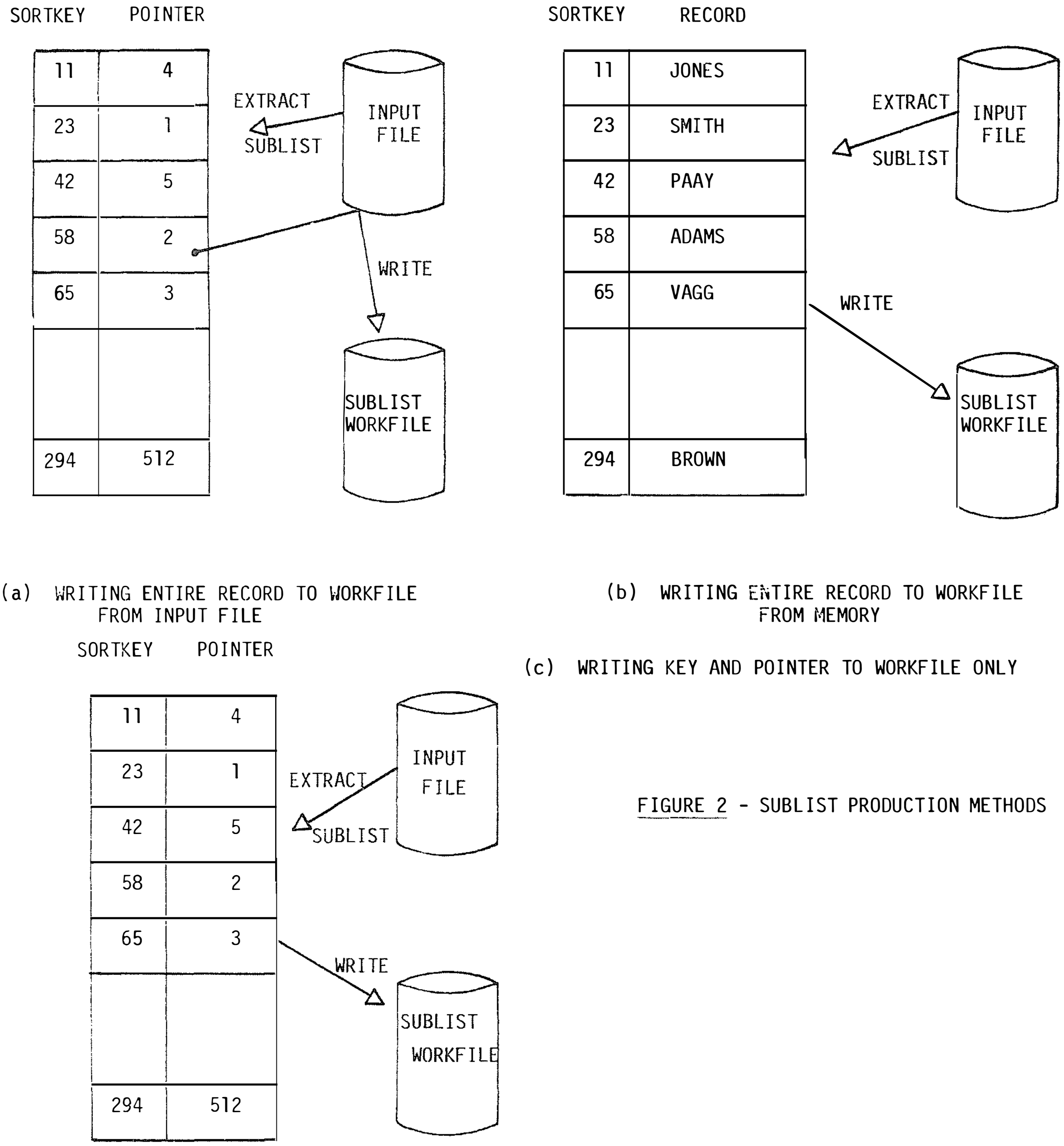
EXTERNAL SORTING

As mentioned before, external sorting is necessary when all the records requiring sorting will not all fit in main memory. External sorting involves merging lists of records created in the internal sort phase of the sort program.

Let's first consider the techniques of producing sorted sublists. A chunk of the input file is read into main memory and sorted as described under record sorting. When complete, the chunk has to be stored away so that another chunk can be read in and sorted. The chunk is referred to as a sublist. Sublists are stored by writing them to one or more temporary disk files called Workfiles. These Workfiles vary in nature, depending on which merging technique is used on the sublists. Three methods of sublist production will be shown, with particular regard to external sorting in a microcomputer environment.

METHOD 1

Refer to Figure 2(A). Only the sorting key is read into an array, and a pointer vector is built while reading in. The pointer vector items are swapped when a key swap occurs. When the internal sort is complete, the input file records are accessed using the pointer vector as an index, and the entire records so retrieved are written to a Workfile.



(a) WRITING ENTIRE RECORD TO WORKFILE FROM INPUT FILE

(b) WRITING ENTIRE RECORD TO WORKFILE FROM MEMORY

(c) WRITING KEY AND POINTER TO WORKFILE ONLY

FIGURE 2 - SUBLIST PRODUCTION METHODS

METHOD 2

Refer to Figure 2(B). The entire record is read into one or several arrays, and a pointer vector is built while reading. The key column is sorted as in Figure 1(B), again producing an index in the pointer vector, and this time the records are written from memory to the Workfile, without having to access the input file again.

METHOD 3

Refer to Figure 2(C). The input file is read and only the key is placed in an array, with a pointer vector, and sorted in the same way as in Method 1. This time, instead of the entire record being retrieved from the input file and written to the Workfile, only the array is written (being key and pointer) to the Workfile. The output file, when properly sequenced after merging, is known as an "ADDROUT" file.

Each of the three methods has its own advantages and disadvantages.

METHOD 1: Because only the key is read in, a large internal sort list is possible, (i.e. more records per given memory space available), meaning large parts of the file are sorted internally, where operations are fastest, and not as many sublists are produced as in Method 2, meaning fewer merge operations. Its disadvantage lies in the fact that an additional random read of the input file is required to create the output sublist in the Workfile. Disk access operations are expensive in terms of time.

METHOD 2: Because the entire record is read into memory at the beginning of the internal sort phase, no additional random read as in Method 1 is required to create the sublist in the Workfile. The disadvantage is that because each item in memory involves the entire record, fewer items per given memory space can be read in during each internal sort phase, thereby meaning more sublists in the Workfile for merging. This will result in more Disk operations during merging, something that you should keep to a minimum in time-critical operations.

METHOD 3: Same advantages as for Method 1. Also, since sublist records are smaller (key and pointer only) than in Method 1, merging operations will be faster. The disadvantage is that the output file after merging will only be an ADDROUT type. If a final sorted file is required, the ADDROUT pointers must be used as an index into the input file to retrieve the records in the desired sequence. However, this need not be a disadvantage if the program requiring records to be in a particular sequence recognizes that fact (e.g. the Debtors Report mentioned above) and reads the ADDROUT file sequentially instead of the final output file.

Note that, at this point, I have not shown how the sublists are organized in the Workfile(s). This will be discussed in the next article.

So far, I have shown how records could be sorted and sublists produced ready for merging. There is still the Debtors Report problem of sorting on Amount Owed within Postcode, with a mixture of ordering - descending on Amount Owed, ascending on Postcode. These features, and more, are provided by most computer manufacturer-supplied sort packages.

SORT PACKAGES

This is a general-purpose software package that is driven by user-supplied parameters, and can be run standalone, or under "Call" by a host program, whatever the case may be. Some of the features provided by these packages are:

* Input file/output file specification, with optional input file replacement.
* Multi-key sorting (Debtor Report example)
* Multi-sequence sorting (ascending/descending mix)
* Selective input record selection
* Multiple input files
* Merge option activation only
* ADDROUT output file only

Let's look at how some of these features could be implemented.

MULTI-KEY

The various subkeys of the input file which will determine the final sequence are usually specified by their position and length in the record. So, in our example, Amount Owed may be specified as starting at byte 35, as length 5, as Type Packed Decimal, as Sort Level 2, and Postcode as starting at byte 22, of length 4, as Type Character, as Sort Level 1.

The various parts of the record are extracted and used to form a sortkey, with Postcode being in the most significant part (left) of the sortkey. (The sortkey is built by concatenating the selected subkey values when reading the input file, and is used as the key for sequencing in the internal sort phase, and external merge phase).

MULTI-SEQUENCE

When specifying the subkeys, the sequence of each is also specified, whether ascending or descending. In our example, Postcode is specified as ascending, and Amount Owed as descending, so that the most serious Debtors appear at the top of the list for each region. In order for records to be sequenced according to this specification, the subkey Amount Owed is complemented before it is built into the sortkey. This means that for a particular record, if the amount owed is 12,448.30, its complement is 9,987,551.70, because the maximum value for a Packed Decimal field of length 5 is 9,999,999.99 and 12,448.30 subtracted from 9,999,999.99 is 9,987,551.70.

If the Postcode of an input record is 5090, then the sortkey will be an alpha with a value of "5090998755170" for that input record.

Using this complement technique for descending subkeys, the internal sorting algorithm is simplified because it need not determine whether an ascending or descending comparison should be made on two keys, because the sortkey will have already taken this into account, even if only one key of descending sequence is specified. The sortkey will simply be the complement of the specified key in this case.

SELECTIVE RECORD SELECTION

This is simply achieved by requesting parameters for record selection, such that if the specified condition was true, the input record is selected, otherwise it is ignored, and will therefore not appear in the final sorted file.

ANSI COBOL SORT FACILITY

The American National Standards Institute has documented a Sort Facility for COBOL compilers (most commonly accepted high-level language for business applications).

In essence, this provides the COBOL programmer with sorting facilities by writing the sort command in the program. It provides most of the features found in manufacturer-supplied sort packages, with an added option of performing user-specified operations on the record before it is passed for sorting, and before it is written to the Workfile.

However, as with most "standard" languages, various enhancements are made by manufacturers to the compiler, and other features as defined by ANSI are dropped, when implemented on their own hardware. Consequently, not all COBOL compilers support the Standard Sort Facility. I imagine that this would tend to make the compiler unnecessarily complex, in view of their own supplied sort packages. Also, the sort facility must, by definition, be of a very general nature, and therefore usually not the most efficient implementation on a particular machine. What the manufacturers can do is supply a Sort Package that is more efficient because it is designed to run on their hardware. So much for "Standards" - they sometimes tend to restrict technological progress.

TO SUMMARIZE...

There is more to sorting in a practical sense than sorting a single list of items in memory. Practical sorting requires sorting of records, with regard to the fact that not all the records may fit in main memory.

There are various phases to sorting records in a file:

1. Parameter specification
2. Input file sublist transfer to memory
3. Internal sort
4. Output sublist production
5. Sublist merging.

The various techniques of sublist merging will be considered in the next article.

- 0000000000 -

HARDWARE

A number of readers have made requests for an article explaining how to increase the amount of memory in the System 80 CPU. The following article by Mr. Brian Hill describes how such a modification can be made. However, before attempting this modification please take careful note of the following warnings.

WARNING

Installing this modification WILL VOID YOUR WARRANTY. Readers with little or no experience of electronics and soldering are advised NOT to perform this modification. The use of a low voltage, well-earthed soldering iron is MANDATORY to avoid the risk of causing damage to the components in your computer. MICRO-80 advises readers that we have not tested this modification and can accept no responsibility for its performance. Eddy, our Hardware Editor, has reservations about the System 80 power transformer. On early models in particular, the transformer is barely adequate and there is the possibility that the extra current required by this modification will cause this transformer to run hotter and fail.

***** HOW TO DOUBLE THE SYSTEM 80 RAM FOR $15 *****

by Brian L. Hill

Faced with the need for more memory, a $425 price tag on the expansion interface and a low bank account, I took a hard look at the insides of my System 80. I found that I could double my memory capacity to 32K for $15!

My converted System 80 has been running with this modification since January and I have converted another machine owned by a friend with no problems.

The System 80 divides the 64K memory addressing capability of the Z80 microprocessor into four blocks of 16K:

BLOCK 1: system ROM, video RAM and keyboard
2: internal RAM
3: external RAM - total of 32K
4: external RAM - total of 48K

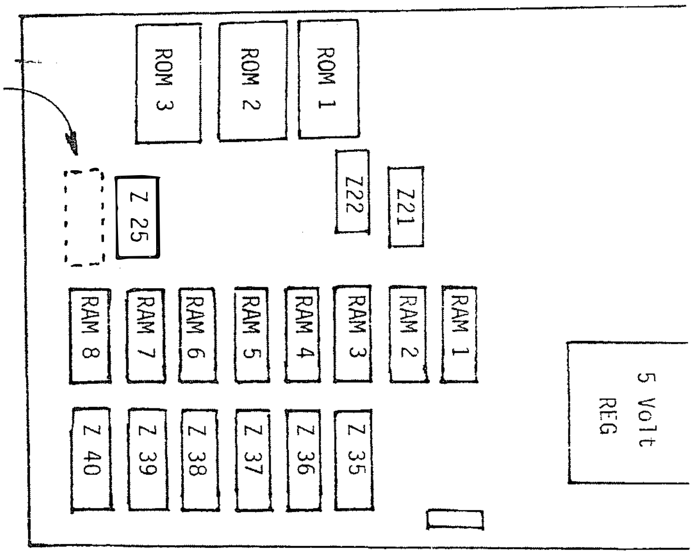
To access any memory in the system, the block has to be selected, the address within the block decoded, the block of memory enabled and then the data buffers for that block turned on. In the System 80, IC Z25 (74LS139) is used to decode address bits 14 & 15 to select the four groups of 16K blocks. The outputs for Blocks 1 & 2 are used with additional circuitry to enable ROM, etc. and to turn on the data buffers for ROM & RAM. The output pins for Blocks 3 & 4 are not connected or used since additional RAM is expected to be in the expansion interface complete with its own decoders and data buffers.

To double the memory, all I had to do was to "piggy-back" the new RAM on top of the original RAM. The new RAM would then share all the address lines and data lines of the original RAM. The only thing left to do was to enable the new RAM when addressed and to turn on the data buffers. To select the new RAM only required a few logic gates in one chip and I could use the unused output pin for Block 3 from Z25 to enable the data buffers. And when I looked inside the machine, I found that the manufacturer had provided a spare 16 pin DIL location right next to Z25!

PROCEDURE:

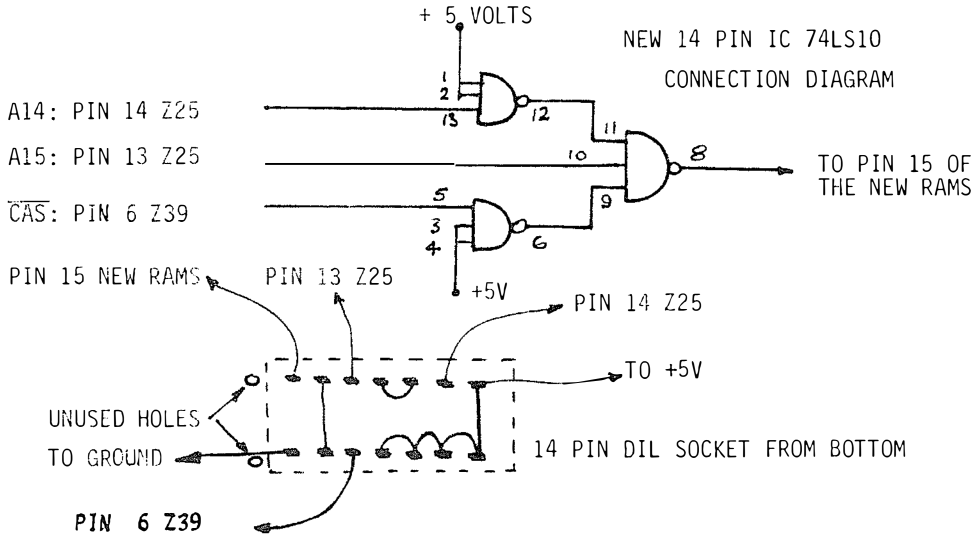
1. Remove the top cover of the machine and then the keyboard.
2. Separate the two main boards and remove the CPU board (left hand one that connects to the keyboard).
3. At the bottom of the board, next to Z25 and the last RAM (#8 on my diagram) there is the spare 16 pin DIL Location. Note that on the top of the board there is a 5 volt supply rail going to pin 16.

UNUSED 16 PIN DIL LOCATION
INSTALL SOCKET TO HOLD 74LS10



4. Fit a 14 pin DIL socket into the spare 16 pin DIL location, keeping the socket to the left so that pin 14 fits into the 5 volt supply.
5. Wire up the socket jumpers as per the diagram. I used KYNAR wire-wrap wire because of its fine size and tough insulation. The circuit was chosen to keep all jumpers as straight as possible to eliminate the possibility of short circuits.
6. Locate Z39 and connect a jumper from pin 6 Z39 ($\overline{CAS}$) to pin 5 of the socket. This will be the RAM enable clock pulse.
7. Jumper from Z25 to the 14 pin socket as per the diagram to bring the address lines 14 & 15. These are decoded to pass the enable clock pulse to the RAM enable pins when the 3rd Block of memory is addressed.
8. Finally connect the supply ground to pin 7 of the 14 pin socket. The ground rail is on

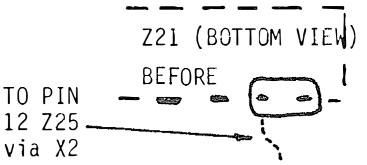
the underside of the board running along the bottom edge of the board from the side to the new socket. (You would think this was meant to be!)

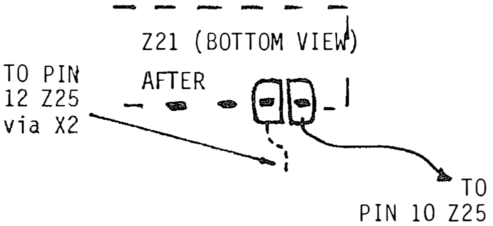


9. Locate Z21 from the diagram and remove the excess solder from pins 1 & 2 with a solder sucker or wick. These two are bridged together and a connection taken from pin 2 on the top of the board goes to Z22. So we'll borrow pin 1. (Z21 = 74LS20)

10. Carefully cut a groove between pins 1 & 2 with a sharp knife removing about 1 mm of copper as shown in the diagram.

11. Connect pin 1 of Z21 to pin 10 of Z25. This will enable the data buffers when addressing Block 3.

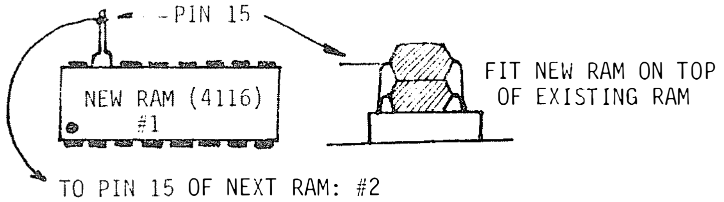


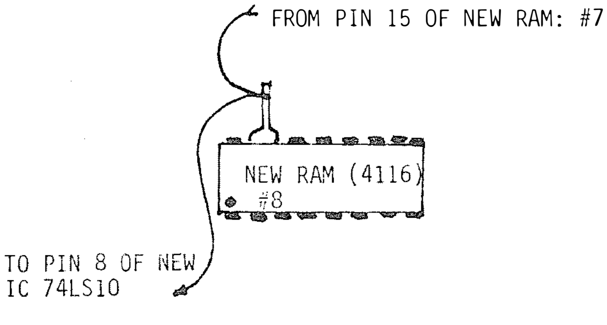


12. Take the 8 new RAM chips and carefully bend pin 15 outwards as shown in the diagram. Bend the rest inwards so they are parallel.

13. Carefully slide the new RAM's over the original RAM's, until they slide into the chip sockets and sit on top of the originals. Although only a little of the legs enter the sockets, the additional pressure of double thickness in the socket hole does hold them quite firmly.

14. With more fine wire, carefully connect all the pin 15's together, looping from RAM #1 to RAM #8. Use a clean tip on the iron to solder the wire to the legs as quickly as possible.





PARTS LIST

8 off 4116 RAM CHIPS
1 off 74LS10 CHIP
1 off 14 PIN DIL SOCKET

* * * * * * * *

15. Finally jumper the looped pin 15's to pin 8 of the new socket. If you use KYNAR wire, then you can pass the wire jumper to the other side of the board through the plated-through hole between Z25 and the new socket.
16. Fit a 74LS10 chip to the new socket and reassemble the computer (provided you have double checked it all!).
17. Power up your System 80 and answer "READY ?" with 'NEW LINE' as normal. The first thing you will notice is that the computer will take a little longer to come back with the READY prompt. This is because you now have a little more memory to check out.
18. When READY comes back, type 'PRINT MEM' and 'NEW LINE' and the computer will respond with 31956.

As I have said, the modification worked just fine, but I did experience a problem which had not occurred before. The System 80 suffered from unexpected crashes, usuall y around 5pm to 7pm. The extra drain on the 5 volt power supply made my machine less resistant to brown-outs, a drop in the supply voltage due to extra demand on the supply system. The answer was simple: swap the 5 volt 1 amp regulator with a LM323 5 volt 3 amp regulator (pin compatible) and no more problems. The other machine I modified lives in Newcastle. Its owner has had no problems at all and it is operating on the original 1 amp regulator. Maybe it is because I live in a country town, or perhaps because I live in N.S.W. with its notorious power generating system. The transformer, rectifiers and capacitors are capable of supplying the modest demands the additional memory requires. So if you experience those annoying crashes, try changing the regulator.

Finally, the BLOCK 4 decode (pin 9 of Z25) is still there. I brought it out to an unused pin on the expansion edge connector. The reason I did not use more memory internally, is because I plan to build a 16K RAM Board to plug onto the expansion connector and allow me to add RAM/EPROM mix to 16K. I can then have all my favourite routines (FASTER, RENUM, KEWORD, MERGE, PACK, etc.) in the machine all the time.

- 000000000 -

REVIEWS

***** TRS-80 ASTROLOGY *****

A Review by David Nourse.

This program, written in machine language for 16K Level I and Level II systems, will compute a Natal Horoscope for anyone born in the present century. The Horoscope can be printed on an 80 column printer, producing an impressive wheel-shaped chart as well as tables of Astrological data.

The tables of data, which would enable the user to draw up his own chart, can also be output as screen displays.

A Natal Horoscope shows the position of the sun, moon and planets in relation to the earth at the time of a person's birth. The program offers a much quicker and easier method of preparing a Natal Chart than the traditional methods, which involve intricate time calculations and tedious reference to tables of planetary movements. It is well documented: a 33 page booklet provides detailed instructions for running the program and a summary of the effects of the planets, signs and houses. An attractive wall poster provides additional summary information and some very stylish artwork.

The data generated by the program include the precise placement of the planets in their signs and houses, the positions of the cusps of each house, and the elements and modes of the signs. A table giving most of the important aspects of the planets, that is, their apparent angular relationships (which may be important factors in a chart), is also produced. I have verified the accuracy of the computations by comparisons with published data. The accuracy is generally much better than the one degree margin of error mentioned in the documentation.

The program requires the precise time, date and place of birth to compute a chart. If the birth-time is unknown, incomplete data can still be generated. The place of birth is entered as co-ordinates of latitude and longitude, which can usually be obtained from a school atlas. A "progressed" chart, indicating trends for a particular year, can also be computed using a subject's birth data, simply by adding to the birth date one day for each year of the subject's life (an established astrological technique).

Interpretation of the results is left to the user. Astrologers believe that the positions of the planets at the moment of birth have a profound and lasting influence on individual behaviour. Proper interpretation requires a knowledge of the many ways in which planetary influences may interact. The program documentation does not give sufficient detail for this purpose. However, some useful books are readily available (for example, Jeff Mayo's "Teach Yourself Astrology", published by Hodder and Stoughton).

Tandy classifies this program as a game. Many people do, however, take Astrology seriously. A careful interpretation of a Natal Horoscope can produce surprising results which are difficult to explain as chance outcomes. My own initial scepticism has been somewhat eroded by investigation!

My only reservations about the program relate to the price and to a technicality. While this is a sophisticated program, incorporating complex corrections for time and lunar position, I feel that its $59.95 price tag is excessive. On the technical side, the program uses the Placidean System of House Division, which is not universally accepted by Astrologers. I can, however, recommend the program to practising Astrologers and beginners alike as a means of saving time and avoiding inaccuracies in casting horoscopes.

- 0000000000 -

***** MICROBUGS *****

Although we make every effort to ensure accuracy in the material we publish, inevitably errors and omissions will occur. In this section, we print corrections to those bugs that have been reported.

FREE SOFTWARE LIBRARY - COMPOSER

Due to an oversight, some of the changes to the 80 composer program in the Free Software Library have been omitted from the instructions. These changes are ONLY required where the program is being typed in from the book and the user is trying to convert the cassette version of the program to run on a disk system. The program as supplied on the Free Software disk is correct and will run on a disk system. This Microbug ONLY affects users trying to convert the cassette version. The following lines are those that need to be changed:-

```
10 DATA62,16,211,254,221,33,54,128,221,78,0,121,183,200,221,70,1
,62,5,211,255,16,254,221,70,1,62,6,211,255,16,254,13,194,14,128,
221,35,221,35,1,255,255,33,48,0,9,218,46,128,195,8,128
20 CLEAR1000:DEFUSR0=&H8000:FORK=&H8000TO&H8034:READ Y:POKEK,Y:N
EXT
25 POKE&H8036,0
180 P=&H8034:GOSUB420
380 W=USR0(0):GOTO190
410 PRINT@832,"DUMP PARAMETERS ARE: START =-32768  END="P+N+2" E
NTRY =-32768";:GOTO190
The following lines should be DELETED :-
405,600,610,620,630,640
```

DR. WHO ADVENTURE - Vol. 3, No. 8 July, 1982

There is an error in line 84Ø of the Main program. It should read:

```
84Ø  A$="":INPUTA$:IFA$=""84Ø
```

If you forget to CLEAR 5Ø after running the Initialiser then an OM error will occur when you try to CLOAD the main program. To avoid this possibility, add the following line to the Initialiser:

```
19Ø CLEAR 5Ø
```

- 0000000000 -

SOFTWARE

***** SINK THE ENEMY NAVY (Colour) - by J.C. Bennett *****

A 2-dimension array is set up to record the status of each position on the grid, where:

0 = No ship here; no moves (shots fired) here.

1 = A move (shot fired) has been registered here.

2-9 correspond to the 8 different ships.

10 = A square adjacent to a diagonal ship.

There is a one dimension array set up for each ship to record the 'POKE' position (Lines 2900-3000). These numbers are used when blocking in the areas indicating that a ship has been sunk (Lines 2020-2890).

While the player is reading the instructions the computer is positioning the ships.

A random position is selected (L380)
A random direction is selected (L390)
The length of ship is calculated (L400)

All the positions needed for that length of ship in that direction from that position are checked. If any are already being used or are outside the grid, then there is a return to Line 380.

If the required number of squares are available, then those positions in the 2-dimension array are set to the appropriate number.

If the ship is set diagonally then the adjacent positions to the left and right are set to 10 to guard against having ships set diagonally at right angles and crossing (L3010-3090).

Moves are entered using INKEY$. The ASCII value is used and numbers calculated for A and B. A is used to calculate horizontal position and B for vertical when making a display on the screen. The same A and B are used to access the information stored in the 2-dimension array (L1780).

The score is increased by 10 for every hit on a target. A sunk ship scores 100 times its length minus twice the number of moves taken before that ship was sunk (e.g. L2090).

TO PLAY

Select your co-ordinates and type them in. If you miss you will see "#" displayed at those co-ordinates and a 'miss' tone will be heard. If a hit is scored a '*' will be displayed and a 'hit' tone will be heard. When all the possible hits on a ship have been made, the '*' signs will be replaced by coloured blocks that indicate the type of ship sunk.

- 0000000000 -

***** MASTERMIND (Colour) by D. Zwart *****

This game will hide a specified number of characters of a specified nature. For example, the computer asks how many items to hide. This is only limited by the amount of string space CLEARed (I played the game with 40 letters hidden without CLEARIng string space). The next thing the computer prompts is "FROM". This is where you specify the lowest letter, number or punctuation you want hidden. (Your ASCII character code will show you how the punctuation is arranged). Then it prompts 'TO'. This is where you specify the highest letter, number or punctuation you want hidden. At this stage, the computer is waiting for you to type in your guess. If you make a mistake the back space will rub out your current line. If you give up the ESC key will give you the answer. A tone will be heard each time a key is pressed. Good Luck!

Line 90	Will dimension to the number you specify to hide.
100-120	Are input requests
120	Will change the A and B inputs around if they were entered high/low instead of low/high.
130	Randomly selects your specified parameters.
140	Counts the inputs per guess.
170	Checks for back space and rubs out current line returning to 140 to reset the counters.
180	Checks for 'ESC' and prints the answer.
190	Checks for keys outside specified parameters and rejects them.
210	Counts numbers in right place.
220	Counts numbers in wrong place.
240	Prints your current guess
250	Prints how many 'rights' and counts number of guesses.
260	Gives how many moves you took and resets the game.

- 0000000000 -

***** GOLF - LII/16K *****

by Robert Glucz

This program is designed to run on a TRS-80/System 80 Level II - 16K machine and uses approximately 14K of available memory.

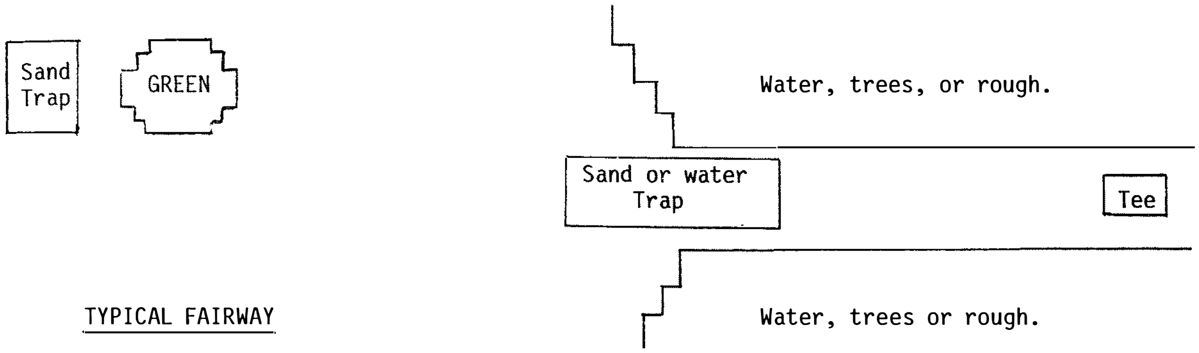
The majority of instructions are provided within the program itself, and after a few games these may be by-passed. However, this should serve as an overview, as well as clarifying some minor points.

MAIN FEATURES

- Play from 1 to 18 holes.
- Graphic display of fairways and greens.
- Sand and water traps, trees, rough, etc.
- Par 3, 4, 5 of varying length.
- Select clubs, direction, strength.
- Hook or slice on strokes
- Duff shots, penalties, and rebound off trees.
- Cut and slope on greens.
- Handicaps
- Replay stroke facility.
- Practice green.

GAMES FORMAT

The game begins by asking players' names and proceeds to explain the rules (if desired). A fairway is then drawn on the screen and each player is asked, in turn, to play a stroke.



TYPICAL FAIRWAY

After each shot, the possibility exists to replay the shot, if the desired result wasn't achieved, or merely for practice (as long as nobody is watching).

When both players have landed on the green, the fairway display disappears and is replaced by the green and hole.

Placement of the balls on the green is semi-random.

The closer you land to the centre of the green (while on the fairway) the closer you are placed to the hole.

On the green itself, two parameters will affect your putt; slope and cut.

The slope of the green is a random value for each green that is multiplied by the strength factor of your putt, and is either added or subtracted from the angle (direction) you chose to putt. (This corresponds to slope up or down on the screen).

Slope governs the shot until near the end of its movement, at which point the cut of the green will take over and deflect the ball either left or right by a random angle. Cut and slope values are displayed as "?" until after the first person has putted and they are then shown on the screen.

This gives the second player an advantage; well-deserved since he landed closest to the hole, i.e. furthest away putts first.

When both players have putted-out, the scores are displayed and the next hole is presented, with the lowest scorer on the previous hole playing first from the tee.

Important points to note are:

- Club, direction, strength, etc. should always be entered separated by comma's, otherwise the screen will scroll up and the display will be affected.
- If this happens, you can either replaythe shot (which redraws the screen) or do nothing, in which case the screen will be redrawn for the next player's shot.
- Entering "9,0,0" as your shot will allow you to access information that you may have forgotten, such as clubs, directions, hints, etc. You may then return to your shot.
- "9,0,0" cannot be selected whilst on the green. You are left to ponder your troubles.
- If you wish to restart the program, use RUN 100 rather than RUN, since the machine-language code embedded at the start tends to cause "funnies". It is all right to use just RUN for the first execution of the program after loading (in fact, it's necessary, or you won't load the machine language program at all).

- 0000000000 -

***** CLEANUP - LI/4K by D.S. Brenton *****

This is a game of skill. The town's mayor has sent you on a mission to coll ect all of the rubbish piles left in the town area. This may seem easy, but there are two problems -

1. You cannot hit any white spot as these are sites where rubbish has already been collected.
2. You cannot hit any white line, as these represent routes that have already been cleaned before. There is one exception to this rule: when you have collected all of the rubbish, you may touch the white areas and your score will be calculated.

If you destroy the top white area by pushing the ENTER key too many times, you will be accused of cheating and lose the game. Choose your level of difficulty, and good luck!

- 0000000000 -

***** E = MC2 - LII/4K *****

In this short program, you can see what happens to the mass of an object as its speed increases. Einstein's theory of relativity proposes that the speed of light (about 300,000km. per second or 186,000 miles per second) is the limiting velocity for any object in our universe. As a measure of its consistency, it also predicts that the mass of an object increases with speed approaching infinity as the speed approachesthat of light.

First of all, you enter the mass of the body at rest (in kilograms) and then its speed (in miles per hour). The program will then display its relativistic mass at that speed.

- 000000000 -

***** ANAGRAMS - LII/16K by Michael Merrylees *****

Anagrams are words in which all the letters are jumbled up. It is an old game often played in schools, in crossword puzzles and on quiz shows.

The purpose of the game is to try and unjumble the anagrams in a limited time. If the player is successful he is rewarded with a point added to his score.

When you run the program, it will ask if you want instructions - if so, type "I" and away you go.

All during the program the only time you press "ENTER" is when you are asked which level of difficulty you want and whether Professional or Amateur timing is wanted.
Anagrams unfortunately requires a computer with 16K to run. The program itself takes up about 8K and the strings and variables bring it up to about 10.5K. Even using the memory savers following, the program still wouldn't fit in a 4K Level II TRS-80.

For any people with not much memory I suggest you remove the remarks, and remove lines 2Ø - 12Ø (the instructions).

You can save some memory by removing data. MAKE SURE that the total amount of words in line numbers 53Ø-65Ø, 66Ø-67Ø, and 68Ø-72Ø all are divisible by 5. Then put the totals of the three groups in a data statement as I have at line 62Ø.

- 0000000000 -

***** SOUTH AUSTRALIAN RACE HORSE PERFORMANCE GUIDE - LII/16K By Geoff Egel *****

The main aim of these programs is to provide an up-to-date performance guide record for any horse that has raced in South Australia, given that relevant data has been entered on a Masterfile and, when used in conjunction with the Inquiry file, to show the following data for the last fifteen weeks: weeks since run, distance, weight carried, finishing rating and today's rating. All horses entered on the Masterfile are sorted into alphabetic order.

Requirements

(a) One TRS-80 Level 2 16K equipped with a cassette recorder.

(b) Access to the racing section of the Sunday Mail published in South Australia.

(c) Optionally, all firsts, seconds and thirds from all mid-week races raced in South Australia.

(d) Two C-15 cassettes (one being for the Inquiry file, the other for the Inquiry file update).

(e) Three C-120 cassettes for the Master file.

SPECIAL FEATURES

(a) All data is automatically sorted into alphabetic order and deleted after fifteen weeks.

(b) Provision has been made for error correction at all stages without the need to retype the correct data.

(c) The Inquiry file is able to hold information for up to one hundred and fifty runners.

(d) The Masterfile can be updated by additional information insertion or by using the Inquiry file update.

(e) Additional information is used to enter mid-week results.

(f) Weekly Masterfile updates can be checked for accuracy and data can be corrected before being merged with the Masterfile.

(g) A Masterfile routine has been included to enable the operator to correct the Masterfile data.

OPERATING INSTRUCTIONS

Inquiry File Request

Individual horses can be compared using the single comparison.

The race horses for a full meeting are entered by using the Inquiry File.

Data common to all horses does not need to be retyped.

When completed, the Inquiry file is compared to the Masterfile and data displayed should be written down for further reference. The horse having the best ratings is usually the one to follow for a place (although they sometimes win, more often they place).

Masterfile Update

This consists of two programs, Part 2 and 3.

(a) Masterfile Inquiry file update.

This program will allow the updating of information collected on the Inquiry file request collected the previous day (e.g. horse name, weight, distance) from results published in the Adelaide Sunday Mail or in the case of mid-week results, using the Time Variants set out below.

(b) Additional Information Insertion.

This option should only be used when the Inquiry file has been updated or when the Inquiry file would not be of much use as with mid-week results where the first three places can only be approximate. When the Inquiry file with Additional Information has been completed, it should then be saved on a cassette no smaller than C-15.

To begin a Masterfile all it is necessary to do is enter the information using Additional Information Insertion and save it to tape. This will become the new Masterfile.

The number of horse names and data should not exceed one hundred and eighty at one time.

(c) Masterfile Merge and Update (Part 3).

To start this program the computer should be reset and memory size set to 32512. This program will allow an array of two hundred and fifty items which will be deleted and added to in alphabetical sequence via the output and input routines.

This program will also ask for a time period of 1-15 weeks; if a longer period is required, then the relevant program line would need to be changed.

If updating is to be conducted mid-weekly as well as at weekends, Line 26Ø should be changed from DC=DC+1 to DC=DC+.5. There is also an option that will allow you to check and correct data in the array before sorting and output. All data is packed in groups of five for input and output. To save data for a period of fifteen weeks on one side of a cassette will require a C120 cassette and the three generation Tape Saving System should be used (Grandfather - Father - Son).

HINTS

The cassette recorder should be demagnetised after each completed Masterfile update.

MICRO-80 PRODUCTS CATALOGUE

This catalogue contains a selection from the wide range of peripherals, interfaces, computers and software carried by MICRO-80 for your computer. If you don't see the item you want, contact us, we probably have it anyway!

MICRO-80 has been supplying customers throughout Australia and the Pacific region by mail-order for 2½ years. Our customers find this a simple and efficient way to do business. You may place your order by telephone or by mailing the order form from any issue of MICRO-80 magazine. Generally, it takes about one week from receipt of order until despatch. You should allow 2-3 days for your letter to reach us and 7-10 days for the parcel to reach you, making a total turnaround time of 2½-3 weeks.

WARRANTY AND SERVICE

All hardware products carry a 90 day parts and labour warranty either from the manufacturer/distributor or from MICRO-80 Pty Ltd. In many cases, warranty servicing can be arranged in your own city, otherwise goods will be repaired by our own team of technicians in our Adelaide workshops.

TRADE-INS AND TERMS

MICRO-80 can accept your existing equipment as a trade-in on new equipment. We can also arrange consumer mortgage financing or leasing on larger hardware purchases. Contact us for details.

LNW EXPANSION INTERFACE for the Model I

- Fully assembled in attractive case with documentation and power supply.

  Complete with 32K RAM, Floppy Disk and RS-232-C interfaces.

  PRICE $550 plus $10 freight

- BARE BOARD, with documentation for $110 plus $2 p.&p.

SOFTWARE — THE LNW 80 COMPUTER:

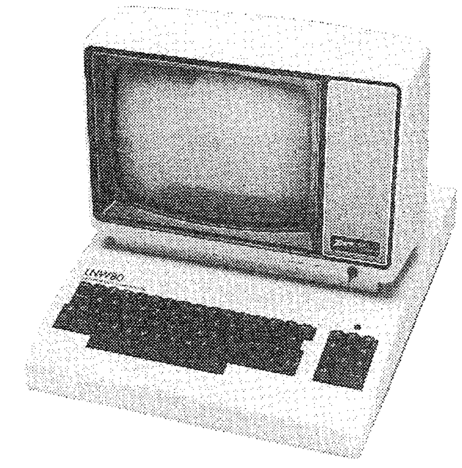
CHARM **$55.00 plus $2.00 p.&p.**

A programmable character generator for designing character sets, symbols and graphic characters with a maximum of ease and flexibility.

AUTOPLOT **$125.00 plus $2.00 p.&p.**

Autoplot enables you to make use of the high resolution capability of the LNW 80 more easily, with the ability to produce hard copies on a suitable printer with bit mapped graphics.

THE LNW80 MkII MICROCOMPUTER



Manufactured in America by LNW Research Corporation, the LNW80 II has the following outstanding features:

- Completely software and hardware COMPATIBLE with the TRS-80 Model 1.
- HIGH RESOLUTION COLOUR GRAPHICS — 4 MODES:
  - B/W LO-RES 128 x 48
  - B/W HI-RES 480 x 192
  - COLOUR LO-RES 128 x 192 in 8 COLOURS
  - COLOUR HI-RES 480 x 192 in 8 COLOURS
- CP/M Disk Operating System.
- Single and Double Density Disk Operation.
- Supports 5¼ inch or 8 inch Floppy Disk Drives.
- 48K RAM in TRS-80 mode plus 16K High Resolution graphics RAM.
- 64K RAM in CP/M mode plus 32K Banked in, usable in BASIC, plus the 16K High Resolution Graphics RAM.
- 4 MHz Z80A microprocessor — over twice the operating speed of the Model 1.
- HI-RES COLOUR (R-G-B) and B&W video outputs.
- 3 screen display modes:
  - 64 characters x 16 lines
  - 80 characters x 16 lines
  - 80 characters x 24 lines
- SOFTWARE SUPPORT

Apart from being able to run all TRS-80 Model 1 software and all CP/M software, there is also an extended BASIC interpreter available for the LNW80 II using most of the same commands as the TRS-80 Colour Computer but with full LNW Graphics Resolution, SET, RESET, POINT, LINE and CIRCLE as well as special commands to generate sound effects and tones. TRS-80 Colour Computer BASIC programs can be transferred to the LNW with only minor changes.

Prices include Sales Tax and are subject to change without notice. Prices are FOB Adelaide. Add $20 road freight anywhere in Australia.
All equipment carries MICRO-80's Australia-wide 90-day warranty covering parts and labour.

The LNW80 II is the ideal computer for the serious hobbyist or businessman who is seeking a higher performance, more reliable computer to replace his TRS-80 Model 1 without sacrificing his investment in software or his programming experience. The LNW80 II uses standard Tandy or Tandy compatible disk drives. If you already have a disk TRS-80 system you may continue to use your existing disk drives on the LNW80 II.

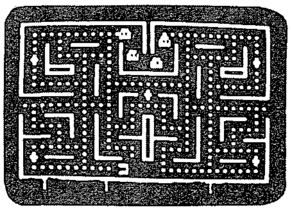
LNW80 II Computer — complete except for disk drives and monitor Includes:

— CP/M Disk Operating System Dosplus 3.4 Double Density Disk Operating System
— LNW Extended Colour Basic Interpreter . . . . **$2850** INC.S.T.

HI-RES Green Phosphor Monitor . . . . **$265** INC.S.T.

Super HI-RES Hitachi RGB Colour Monitor . . . . **$1250** INC.S.T.

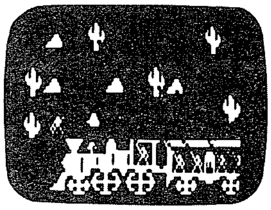
Two Singlesided 40 Track Double Density Disk
in cabinet with power supply and cable . . . . **$825** INC.S.T.



SCARFMAN

This incredibly popular game craze now runs on your TRS-80! It's eat or be eaten You run Scrarfman around the maze, gobbling up everything in your path Try to eat it all before nasty monsters devour you. Excellent high speed machine language action game from the Cornsoft Group. With sound

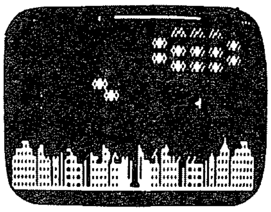
Price: $17.95



THE WILD WEST

It's up to you to keep the West beautiful with Outlaws and renegade Indians on all sides. Even the train has been captured by Outlaws with all the payroll on board. Can you clean up the Wild West?

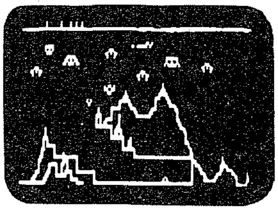
Price: $26.50



SPACE ATTACK

Steady your nerves, keep a sharp lookout, and prepare for battle to save your city. Fiendish aliens are all around, and if they destroy the city you lose.

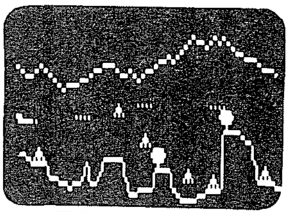
Price: $26.50



STRIKE FORCE

As the primary defender of a world of cities under deadly alien attack, your weaponry is the latest rapid fire missiles, long range radar, and incendiary "star shells." Your force field can absorb only a limited number of impacts. A complex game of strategy, skill and reflexes from Melbourne House

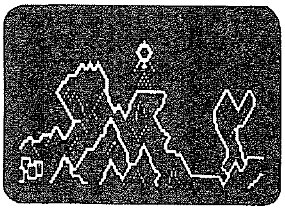
Price: $26.50



PENETRATOR

Soar swiftly over jagged landscape, swooping high and low to avoid obstacles and enemy missiles attacks. With miles of wild terrain and tunnels to penetrate, you're well armed with bombs and multiple forward missile capability From Melbourne House. Features sound, trainer mode and customizing program.

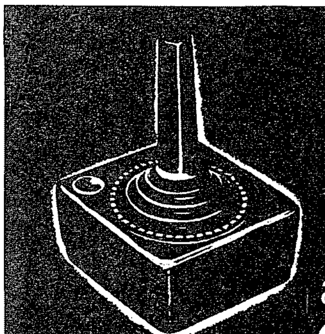
Price: $36.50



LUNAR LANDER

As a vast panoramic moonscape scrolls by, select one of many landing sights The more perilous the spot, the more points scored -- if you land safely. You control LEM main engines and side thrusters. One of the best uses of TRS-80 graphics we have ever seen. From Adventure International. With sound.

Price: $26.50



STICKEROO JOYSTICK INTERFACE

for the TRS-80 MODEL I and SYSTEM 80

FROM **$32.00**

ADD $2.00 p. & p.

CONVERT YOUR COMPUTER INTO AN ARCADE GAMES MACHINE

Micro-80's Stickeroo Interface Features:

•Compatible with Joysticks for Atari, Vic-20 and most video games •Saves your keyboard from abuse •Compatible with programs from leading US software houses: Big-Five, Cornsoft, Melbourne House, Adventure International •Adds a whole new dimension of pleasure and fun to your favourite games •Will be supported in MICRO-80 •Can be used with your own basic or ML Programs •Comes complete, ready to plug in and use •Absolutely no modifications required to your computer.

Due to popular demand, Stickeroo Interface is now available separately so you can use the Joystick of your choice.

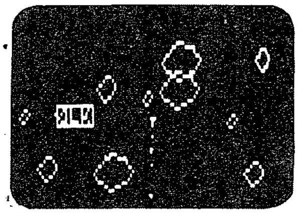
PRICE INCLUDES ... STICKEROO + INSTRUCTIONS + DEMO PROGRAM LISTING
PLEASE SPECIFY TRS-80 MODEL I OR SYSTEM 80 WHEN ORDERING

The Stickeroo Interface plugs in to the expansion edge connector and may not be suitable for expanded systems.

PISTOL GRIP JOYSTICK WITH FIRE BUTTON

$25 + $2 p & p (No p & p required if ordered with Stickeroo Interface)

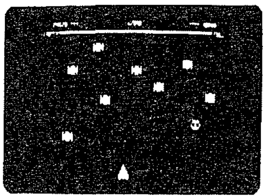
ALL GAMES ADVERTISED ON THIS PAGE ARE STICKEROO COMPATIBLE



SUPER NOVA

Asteroids float ominously around the screen. You must destroy the asteroids before they destroy you! (Big asteroids break into little ones) Your ship will respond to thrust, rotate, hyperspace and fire. Watch out for that saucer with the laser! As reviewed in May 1981 Byte Magazine.

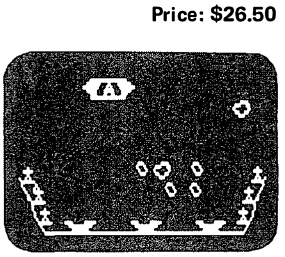
Price: $26.50



COSMIC FIGHTER®

Your ship comes out of hyperspace under a convoy of aliens You destroy every one. But another set appears These seem more intelligent You eliminate them too Your fuel supply is diminishing. You must destroy two more sets before you can dock The space station is now on your scanner . With sound!

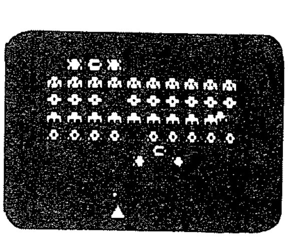
Price: $26.50



METEOR MISSION II

As you look down on your view, astronauts cry out for rescue. You must maneuver through the asteroids and meteors (Can you get back to the space station?) Fire lasers to destroy the asteroids, but watch out, there could be an alien Flagship lurking Includes sound effects!

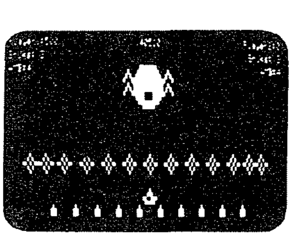
Price: $26.50



GALAXY INVASION

The sound of the klaxon is calling you! Invaders have been spotted warping toward Earth. You shift right and left as you fire your lasers. A few break formation and fly straight at you! You place your finger on the fire button knowing that this shot must connect! With sound effects!

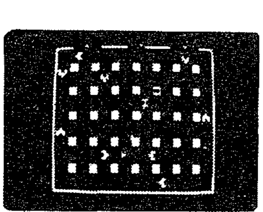
Price: $26.50



DEFENSE COMMAND

The invaders are back! Alone, you defend the all important nuclear fuel canisters from the repeated attacks of thieving aliens, repeatedly An alien passes your guard, snatches a canister and flys straight off. Quick! You have one last chance to blast him from the sky! With sound and voice.

Price: $26.50



ATTACK FORCE®

As your ship appears on the bottom of the maze, eight alien ships appear on the top, all traveling directly at you! You move toward them and fire missiles But the more aliens you destroy the faster the remaining ones become If you get too good you must endure the "Flag-ship" . With sound effects!

Price: $26.50

FOR YOUR ENTERTAINMENT

MICRO-80 now offers you the widest range possible in entertainment software. These programs are supplied on cassette for the Level II/16K TRS-80 Model I/III (except as noted). They are also suitable for the System 80 but sound may not be available unless a hardware modification has been fitted to reverse the roles of recorders #1 and #2. *Order yours now while stocks last!*

DEFENCE PENETRATOR $20.95

DEFENCE PENETRATOR is based on one of the most popular arcade favourites of all time with smooth graphics and sound effects. With realistic scrolling planetscape it's the best game yet.

DEVIL'S TOWER $25.95

Aliens move in waves of 5 attackers with their robot scouts attacking you from the mountain, their war machines and their protector ships putting up force fields to protect them. Only your skill and fast reflexes can save the plant.

BATLE STATION $21.50

The aim of the game is to defend your space station against the attack of four alien space ships.

MORGOTH $20.95

Morgoth is a unique action packed adventure allowing you to wander through the enchanted dominion of Morgoth and collect the lost treasures of KAZARD KALLAHAN. But Beware! You must escape before the satanic Morgoth is aroused and seeks yea!

KILLER BEETLES $21.50

The aim of the game is to dig traps. When a beetle falls in you must fill it in to bury them, before they can catch you.

STAR CRESTA $20.95

Star Cresta takes you beyond the limits of your computer and into the Cosmic void itself! Beware! Iron clad concentration and lightning relfexes are required to destroy the evil empress.

JUNGLE RAIDERS $21.50

The aim of the game is to defend your four bases from the marauding Jungle Raiders. Your skill all the Jungle Raiders and they try to hit you with their spears or drag off all four of your bases.

ALIEN TAXI $28.50

Your goal is to pick up and deliver passengers to an underground resort hotel. There is a fare at each of the 12 taxi stands on the first level and 12 more on the second level.

KILLER GORILLA $21.50

Four completely different frames. Each one offering a different challenge, makes this one of the most complex and stimulating games ever written for a TRS-80. The game keeps track of the top ten scores along with a six character name for each score.

JUNGLE BOY $21.50

The ultimate challenge! Are your reflexes fast enough to swing Jungle Boy from vine to vine? Can you swing through the jungle? Can you swim by the alligators? These are just some of the things you will find very challenging in Jungle Boy.

STELLAR WARP $20.95

Animation with superior fighter craft brings you an even greater challenge. As your computer advances your level, the aliens become more dangerous and the harder it is to stay alive!

HOPPY $21.50

The aim of the game is to get your frogs across the busy highway without being squashed and then across the river by means of floating logs and turtles.

PANIK $28.50

Your mission is to rid the galaxy of the Mzors forever. Mzors are half animal and half machine. Their leaders are very difficult to destroy and are capable of creating more warriors at will. Your weapons are your energy pistol, short range transporter pack and your courage.

INSECT FRENZY $21.50

The aim is to stop the centipede from getting you, all the time keeping an eye out for the giant spider.

ALIEN CRESTA $21.50

The aim is to defend your ship from numerous attacks from an assortment of aliens. If you get hit three times, it's all over.

DESERT PERIL $28.50

The Zagons have mined the desert and have put killer satellites, drone bomber balloons, and flying dragons along the whole trail. The future of your planet's race depends on your skill and daring.

RALLY RACER $20.95

Drive through an action packed maze and try to hit all the flags before Morgan the Mad motorist or Crazy Harry and his killer hoodlums catch you!

NOTE:

As the prices of imported software may vary, these prices are valid for current stock only and prices are subject to change without notice.

BUY YOUR MODEL 3 FROM MICRO-80 AND SAVE $00's

MICRO-80 fits reliable MPI disk drives to the TRS-80 Model 3 to give system capacities and capabilities far in excess of those available elsewhere. All our conversions utilise low dissipation, switching-mode supplies to avoid screen jitter and overheating. The disk controller boards used incorporate special compensation circuitary for 80 track disk drives and may also be used to run 8 inch disk drives with an appropriate cable and DOS.

MODEL 340 **$2595**

2 40 TRACK SINGLE-HEAD DISK DRIVES GIVING
350K FORMATTED STORAGE, 48K RAM

MODEL 340+ **$2975**

2 40 TRACK DUAL-HEAD DRIVES GIVING
700K FORMATTED STORAGE, 48K RAM

MODEL 500 — 5+ MEGABYTE MODEL 3 **$6250**

1 40 TRACK DUAL-HEAD DRIVE GIVING 350K
OF FLOPPY DISK STORAGE FOR TRANSFERRING
PROGRAMS AND BACKUP, 48K RAM, EXTERNAL
5 MEGABYTE WINCHESTER SUB-SYSTEM,
DOSPLUS 4.0 DISK OPERATING SYSTEM

The MODEL 500 offers the high speed, mass storage capacity and reliability of a Winchester drive for thousands of dollars less than you would pay for any comparable system. Model 500 is a serious business computer able to tackle the most demanding tasks.

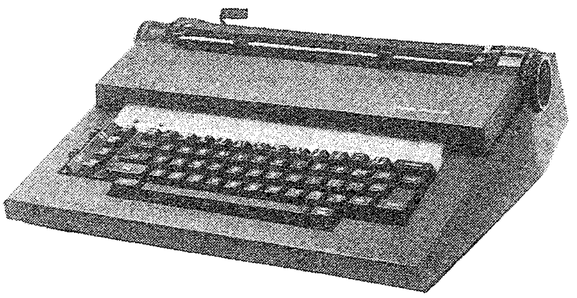
WINCHESTER DISK DRIVE SUB-SYSTEM **5MByte $3495**
10MByte $3995

This Winchester Disk Drive sub-system provides either 5 or 10 Megabyte of reliable, high speed storage. It connects to any standard Model 3 equipped with one or more floppy disk drives and does not void the Tandy warranty. Complete with DOSPLUS 4.0 Disk Operating system.

Prices include Sales Tax and are subject to change without notice. Prices are FOB Adelaide. Add $20 road freight anywhere in Australia. All computers and peripherals carry MICRO-80's Australia-wide. 90-day warranty covering parts and labour.

Daisy Wheel Printers/Typewriters

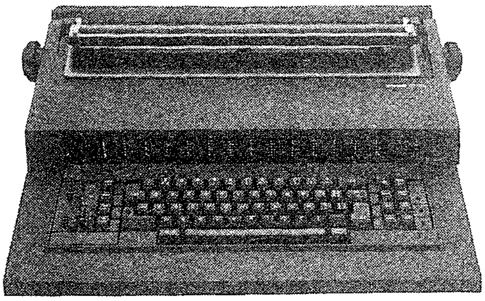
OLIVETTI PRAXIS 35



$895

plus $10 road freight anywhere in Australia

OLIVETTI ET-121



$1500

plus $20 road freight anywhere in Australia

MICRO-80 has converted these OLIVETTI typewriters to work with the TRS-80, SYSTEM 80 or any other microcomputer with a Centronics parallel port. Now you can have the best of both worlds — an attractive, modern, correcting electronic typewriter which doubles as a correspondence quality Daisy Wheel printer when used with your microcomputer.

The **PRAXIS** is a portable typewriter, designed for private and light commercial use with an average print speed of 6.5 c.p.s.

The **ET-121** is a large typewriter intended for heavier duty and features a print speed of up to 11.5 c.p.s.

Centronics printer cable to suit TRS-80 or SYSTEM 80 . . . . $39

MICRO-80 is an A-Grade Olivetti distributor and has been producing printer conversions for Olivetti daisy wheel typewriters for several years. Write or call for full details.

16K Memory Upgrade Kit

$30

plus $2.00 p. & p.

Large volume means we can buy better and can pass the savings on to you. There are our proven, prime, branded 200 nanosecond chips, guaranteed for 12 months. A pair of DIP shunts is also required to upgrade 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 install them.

Lower Case Modification

$49

plus $2.00 p. & p.

The MICRO-80 modification features true below-the-line descenders, a block cursor and symbols for the 4 playing-card suits. Each kit comes with comprehensive fitting instructions and two universal lower-case driver routines on cassette to enable you to display lower case. These routines are self-relocating, self-protecting and will co-reside with other machine language programs (the second includes keyboard-debounce and flashing cursor). Fitting requires soldering inside the computer and should only be carried out by an experienced hobbyist or technician. A fitting service is available in capital cities for only $20.00 and a list of installers is included with each kit. (Specify TRS-80 Model I or System 80 when ordering.)

All prices include Sales Tax and are correct at time of publication but are subject to changes without notice. All equipment carries MICRO-80's Australia-wide 90-day warranty covering parts and labour.

DISK OPERATING SYSTEMS & DEVELOPMENT SOFTWARE

You can increase your programming productivity, the execution speed and 'user friendliness' of your programs by using an enhanced Disk Operating System (DOS). Together with the other utility software, you can get the most from your disk drives.

DOSPLUS 3.4 **$149.95**

(Specify Model I single/double density or Model III)

A powerful DOS that provides many features and comes with a stand alone manual. With a high-degree of compatibility with TRSDOS, DOSPLUS 3.4 is suitable for the first-time or experienced user.

DOSPLUS 3.5 **$160.00**

(Specify Model I or Model III)

DOSPLUS 3.5 is a powerful, sophisticated DOS intended for the experienced user. The system can be configured to suit your requirements, provides greatly enhanced features over 3.4 and new features like single-key entry, date-stamping of files, a Help file and more. More user friendly than 3.4, DOSPLUS 3.5 comes with a very extensive stand-alone manual.

ENHBAS **$52.95**

ENHBAS adds over 30 new commands and functions to your BASIC interpreter including high speed SORT, labels in BASIC, RESTORE to any line number, WHILE-WEND for structured programming, SCROLL, LEFT, INVERT, DRAW and PLOT to give you ease of control over graphics, SOUND and PLAY to add realistic sound effects and many more. Makes programming a breeze! Available for Model I or III, disk or cassette — specify which when ordering.

NEWDOS 80 VERSION 2.0 **$185.00**

(Specify Model I or Model III)

Newdos 80 suits the experienced user who has already used TRSDOS, understands the manual and is prepared to learn the somewhat complicated syntax of one of the most powerful DOS's available. With the correct hardware, Newdos 80 supports any mix of single- or double-sided, single or double density, 5" or 8" disk drives with track counts up to 96. It provides powerful, flexible file handling in BASIC including variable length records up to 4096 bytes. Definitely not for the beginner.

MASTER DISK DIRECTORY **$20.95**

FIND THE PROGRAM FAST!! PAYS FOR ITSELF BY RELEASING REDUDANT DISK SPACE!! MASTER DIRECTORY records the directories of all your individual disks onto one directory disk. Then it allows you examine them, find an individual file quickly, list files alphabetically, weed out redundant files, identify disks with free space, list files by extension, etc., etc. This program is invaluable for the serious disk user and will pay for itself many times over. Not fully compatible with NEWDOS 80.

THE FLOPPY DOCTOR/MEMORY DIAGNOSTIC

Model III Disk **$43.50**

THE MICRO CLINIC offers two programs designed to thoroughly check out the two most trouble-prone sections of the TRS-80 — the disk system (controller and drives) and the memory arrays. Both programs are written in Z80 machine code and are supplied together on diskette for a minimum 32K, one disk system.

Note: For DOSes, include $2.00 for freight.

MORE ENTERTAINMENT SOFTWARE

BOSKONE ALERT **$25.50**

You have total control of every aspect of your fighter and must use your laser to destroy 9 Deathstars before the Earth comes into range.

OUTLAND **$25.95**

You must use your skills, reflexes and an array of weapons to defend your colony against the attacks of Xenos Star Raiders and prevent its destruction.

STELLAR WARP **$20.95**

Use your fighter craft to destroy the aliens who become more dangerous as your level advances. Beware of the space mines. In an emergency, activate Stellar Warp.

DOOMSDAY MISSION **$25.50**

You must disarm a number of nuclear missiles left by saboteurs on one of our space stations. Any direct assault on the station could launch those missiles.

DT-80 DOT MATRIX PRINTER

Features:

- 80 cps bi-directional, logic seeking
- 40, 71, 80 or 142 characters per line
- Normal and italic alphanumeric, symbol and semi-graphic characters
- Unidirectional bit image graphics (8 x 640 or 8 x 1280 dot/line)
- Tractor and friction feed

SPECIAL PRICE FOR THIS MONTH ONLY —

$599

★ ★ ★ NEW PRODUCT ★ ★ ★

CASE DP 515 DOT MATRIX PRINTER

Features:

100 cps bidirection, logic seeking
136, 164, or 233 characters per line
9 x 9 in character mode (6 x 6 for block graphics)
ASCII, italics, block graphics, special and proportional characters
Unidirectional bit image graphics (8 x 816 or 8 x 1632 dots/line)
Superscript, subscript, underlining
Epson compatible control codes
Tractor, friction and single sheet

PRICED AT ONLY $1,099!

Optional serial interface available for $113 plus fitting

OTHER PRINTERS AVAILABLE:

EPSON RX-80 **$995**
Features: 100 cps, 6 character sizes, bit image and graphic modes.

ITOH PROWRITER 8510 **$1150**
Features: 120 cps, bit graphics and proportional printing.

EPSON FX-80 **$1399**
Features: 160 cps, 6 character sizes, proportional printing, bit graphics.

All prices include Sales Tax and are correct at time of publication but are subject to change without notice.
All equipment carries MICRO-80's Australia-wide 90 day warranty covering parts and labour.
Add $10 road freight anywhere in Australia.

Whilst these programs are helpful in picking the main chances in a race, no responsibility can be accepted for losses incurred in the operation of these programs.

MID-WEEK FINISHING POSITION TIMES VARIANTS

The paper shows the time taken for the winning horse to complete the journey; the times for second and third can be approximated by adding the following to that time:

For each length	.15 seconds
For each half-length	.07 seconds
Head	.04 seconds
Short half-head	.02 seconds
Long neck	.08 seconds
Short neck	.06 seconds
3/4 length	.11 seconds
1/4 length	.04 seconds

THEORY OF OPERATION

The Inquiry file is used to access the information contained on the Masterfile. The names of horses and relevant data is stored in the Inquiry file array (A$) and sorted into alphabetic order. When all the data has been entered, it is then compared to the data on the Masterfile. When a match occurs a rating is calculated and stored in the array (Y$). The 'time' saved on the Masterfile is compared to the National Record stored in the data lines for the particular distance (which will need changing when broken). Track condition, distance and weight are also taken into account. The individual comparison is used where a rating not contained on the Masterfile is desired (e.g. an interstate horse finishing with a place). For the mid-week Inquiry File the instructions regarding tape number one can be ignored.

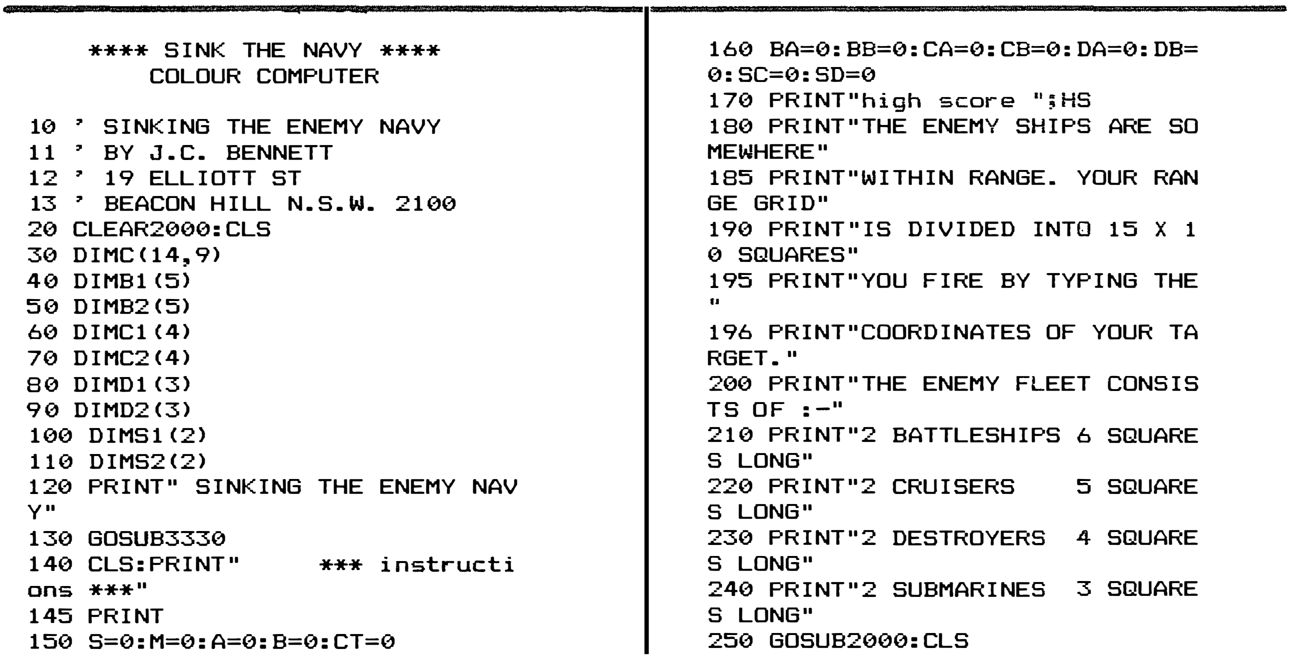
MASTERFILE UPDATING

The weekend updating is begun by using the Inquiry file tape and entering the information requested. If Inquiry file tape is not available (as with the mid-week results) then information can be entered via 'additional information insertion'. The updated Inquiry file can be checked for errors and corrections made. As a backup measure, there is incorporated a 'save on tape' routine - this allows the re-loading of the updated Inquiry file (via 'week tape load') if problems arise. If this occurs, the "Ends" must be deleted using the Inquiry file check. The Inquiry file can now be merged with the Masterfile.

The updating process can take up to two and half hours but can be left unattended. A list of options will be displayed when done.

NOTE - This program is only suitable for 16K Level II cassette-based machines.

- 0000000000 -



```
     **** SINK THE NAVY ****
         COLOUR COMPUTER

10 ' SINKING THE ENEMY NAVY
11 ' BY J.C. BENNETT
12 ' 19 ELLIOTT ST
13 ' BEACON HILL N.S.W. 2100
20 CLEAR2000:CLS
30 DIMC(14,9)
40 DIMB1(5)
50 DIMB2(5)
60 DIMC1(4)
70 DIMC2(4)
80 DIMD1(3)
90 DIMD2(3)
100 DIMS1(2)
110 DIMS2(2)
120 PRINT" SINKING THE ENEMY NAV
Y"
130 GOSUB3330
140 CLS:PRINT"     *** instructi
ons ***"
145 PRINT
150 S=0:M=0:A=0:B=0:CT=0
160 BA=0:BB=0:CA=0:CB=0:DA=0:DB=
0:SC=0:SD=0
170 PRINT"high score ";HS
180 PRINT"THE ENEMY SHIPS ARE SO
MEWHERE"
185 PRINT"WITHIN RANGE. YOUR RAN
GE GRID"
190 PRINT"IS DIVIDED INTO 15 X 1
0 SQUARES"
195 PRINT"YOU FIRE BY TYPING THE
"
196 PRINT"COORDINATES OF YOUR TA
RGET."
200 PRINT"THE ENEMY FLEET CONSIS
TS OF :-"
210 PRINT"2 BATTLESHIPS 6 SQUARE
S LONG"
220 PRINT"2 CRUISERS    5 SQUARE
S LONG"
230 PRINT"2 DESTROYERS  4 SQUARE
S LONG"
240 PRINT"2 SUBMARINES  3 SQUARE
S LONG"
250 GOSUB2000:CLS
```

```
260 PRINT"TO THE RIGHT OF THE SC
REEN WILL"
270 PRINT"BE THE FOLLOWING :-"
280 PRINT"1.high  score"
285 PRINT"2.moves taken"
290 PRINT"3.your  score"
295 PRINT"THE SOONER YOU SINK A
SHIP THE"
300 PRINT"HIGHER YOUR SCORE."
310 PRINT"YOU HAVE A MAXIMUM OF
100 MOVES"
320 C(A,B)=0
330 A=A+1:IFA<15THEN320
340 A=0:B=B+1:IFB<10THEN320
350 'PLACING SHIPS
360 T=2
370 '
380 A=RND(14):B=RND(9)
390 D=RND(8):IFD=0THEN390
400 ST=7-INT(T/2)
410 ON D GOSUB850,950,1050,1150,
1250,1380,1510,1640
411 PRINT@480,"SETTING UP GAME "
;
420 T=T+1:IFT<10THEN370
470 CLS:POKE1024,128:E=1024:F=0
480 E=E+1:F=F+1
490 POKEE,F
500 IFE<1039THEN480
505 POKE1040,128
510 E=1024:F=47
520 E=E+32:F=F+1
530 POKEE,F
540 IFE<1344THEN520
550 E=1376:POKEE,128:F=0
560 E=E+1:F=F+1
570 POKEE,F
580 IFE<1391THEN560
585 POKE1392,128
590 E=1040:F=47
600 E=E+32:F=F+1
610 POKEE,F
620 IFE<1360THEN600
630 B1=6:B2=6:C1=5:C2=5:D1=4:D2=
4:S1=3:S2=3
660 PRINT@384,CHR$(175);"BATTLES
HIPS = ";B1;":6  ";B2;" :6";
670 PRINT@416,CHR$(255);"CRUISER
S    = ";C1;":5  ";C2;" :5";
680 PRINT@448,CHR$(159);"DESTROY
ERS  = ";D1;":4  ";D2;" :4";
690 PRINT@480,CHR$(239);"SUBMARI
NES  = ";S1;":3  ";S2;" :3";
695 PRINT@273,"HI SCORE";HS;
700 GOSUB2005
720 M1=ASC(IN$)
730 IFM1<58ANDM1>47THENB=M1-48:G
OTO750
740 IFM1<80ANDM1>64THENA=M1-65EL
SE700
750 GOSUB2005
770 M2=ASC(IN$)
780 IFM2<58ANDM1>47THENB=M2-48:G
OTO800
790 IFM2<80ANDM2>64THENA=M2-65EL
SE750
800 M=M+1:PRINT@305,"MOVE NO.";M
;
810 GOSUB1770
820 IFB1=0ANDB2=0ANDC1=0ANDC2=0A
NDD1=0ANDD2=0ANDS1=0ANDS2=0THEN3
050
830 IFM=100THEN3090
840 GOTO660
850 '** horizontal left **
860 SA=A
870 IFA>14-ST THENT=T-1:GOTO940
880 IFC(A,B)>0THENT=T-1:GOTO940
890 A=A+1:CT=CT+1:IFCT<ST THEN88
0
900 A=SA:CT=0
910 C(A,B)=T
920 GOSUB2900
930 A=A+1:CT=CT+1:IFCT<ST THEN91
0
940 CT=0:RETURN
950 '** horizontal right **
960 SA=A
970 IFA<ST THENT=T-1:GOTO1040
980 IFC(A,B)>0THENT=T-1:GOTO1040
990 A=A-1:CT=CT+1:IFCT<ST THEN98
0
1000 A=SA:CT=0
1010 C(A,B)=T
1020 GOSUB2900
1030 A=A-1:CT=CT+1:IFCT<ST THEN1
010
1040 CT=0:RETURN
1050 '** down **
1060 SB=B
1070 IFB>9-ST THENT=T-1:GOTO1140
1080 IFC(A,B)>0THENT=T-1:GOTO114
0
1090 B=B+1:CT=CT+1:IFCT<ST THEN1
080
1100 B=SB:CT=0
1110 C(A,B)=T
1120 GOSUB2900
1130 B=B+1:CT=CT+1:IFCT<ST THEN1
110
1140 CT=0:RETURN
1150 '** up **
1160 SB=B
1170 IFB<ST THENT=T-1:GOTO1240
1180 IFC(A,B)>0 THENT=T-1:GOTO12
40
1190 B=B-1:CT=CT+1:IFCT<ST THEN1
180
1200 B=SB:CT=0
1210 C(A,B)=T
1220 GOSUB2900
1230 B=B-1:CT=CT+1:IFCT<ST THEN1
210
1240 CT=0:RETURN
1250 '** diagonal up left **
1260 SA=A:SB=B
1270 IFA<ST OR B<ST THENT=T-1:GO
TO1370
1280 IFC(A,B)>0THENT=T-1:GOTO137
0
1290 A=A-1:B=B-1:CT=CT+1
1300 IFCT<ST THEN1280
1310 A=SA:B=SB:CT=0
1320 C(A,B)=T
1330 GOSUB2900
1340 GOSUB3010
1350 A=A-1:B=B-1:CT=CT+1
1360 IFCT<ST THEN1320
1370 CT=0:RETURN
1380 '** diagonal up right **
1390 SA=A:SB=B
1400 IFA>14-ST OR B<ST THENT=T-1
:GOTO1500
1410 IFC(A,B)>0THENT=T-1:GOTO150
0
1420 A=A+1:B=B-1:CT=CT+1
1430 IFCT<ST THEN1410
1440 A=SA:B=SB:CT=0
1450 C(A,B)=T
1460 GOSUB2900
1470 GOSUB3010
1480 A=A+1:B=B-1:CT=CT+1
1490 IFCT<ST THEN1450
1500 CT=0:RETURN
1510 '** diagonal down left **
1520 SA=A:SB=B
1530 IFB<ST OR A<ST THENT=T-1:GO
TO1630
1540 IFC(A,B)>0THENT=T-1:GOTO163
0
1550 A=A-1:B=B-1:CT=CT+1
1560 IFCT<ST THEN1540
1570 A=SA:B=SB:CT=0
1580 C(A,B)=T
1590 GOSUB2900
1600 GOSUB3010
1610 A=A-1:B=B-1:CT=CT+1
1620 IFCT<ST THEN1580
```

```
1630 CT=0:RETURN
1640 '** diagonal down right **
1650 SA=A:SB=B
1660 IFB>9-ST OR A>14-ST THENT=T
-1:GOTO1760
1670 IFC(A,B)>0THENT=T-1:GOTO176
0
1680 A=A+1:B=B+1:CT=CT+1
1690 IFCT<ST THEN1670
1700 A=SA:B=SB:CT=0
1710 C(A,B)=T
1720 GOSUB2900
1730 GOSUB3010
1740 A=A+1:B=B+1:CT=CT+1
1750 IFCT<ST THEN1710
1760 CT=0:RETURN
1770 '** players move **
1780 P=C(A,B)
1790 C(A,B)=1
1800 PP=1057+A+B*32
1810 IFP=0 OR P=10 THENPOKEPP,35
:SOUND25,2:GOTO2010
1820 IFP=1THEN2010
1830 IFP>1 AND P<10 THENPOKEPP,4
2
1840 S=S+10:SOUND200,3
1850 IFB1>0 AND P=2 THENB1=B1-1
1860 IFB2>0 AND P=3 THENB2=B2-1
1870 IFC1>0 AND P=4 THENC1=C1-1
1880 IFC2>0 AND P=5 THENC2=C2-1
1890 IFD1>0 AND P=6 THEND1=D1-1
1900 IFD2>0 AND P=7 THEND2=D2-1
1910 IFS1>0 AND P=8 THENS1=S1-1
1920 IFS2>0 AND P=9 THENS2=S2-1
1930 IFB1=0 AND BA=0 THENGOSUB20
20
1940 IFB2=0 AND BB=0 THENGOSUB21
30
1950 IFC1=0 AND CA=0 THENGOSUB22
40
1960 IFC2=0 AND CB=0 THENGOSUB23
50
1970 IFD1=0 AND DA=0 THENGOSUB24
60
1980 IFD2=0 AND DB=0 THENGOSUB25
70
1990 IFS1=0 AND SC=0 THENGOSUB26
80
1991 IFS2=0 AND SD=0 THENGOSUB27
90
1992 RETURN
2000 PRINT@480,"PRESS ANY KEY TO
 TURN THE PAGE";
2005 IN$=INKEY$:IFIN$=""THEN2005
ELSERETURN
2010 RETURN
```

```
2020 '** battleship sunk **
2030 H=0
2040 P=B1(H)
2060 POKEP,175
2080 H=H+1:IFH<6THEN2040
2090 S=S+600-M*2
2100 GOSUB3300
2110 BA=1
2120 RETURN
2130 '** battleship 2 sunk **
2140 H=0
2150 P=B2(H)
2160 POKEP,175
2190 H=H+1:IFH<6THEN2150
2200 S=S+600-M*2
2210 GOSUB3300
2220 BB=1
2230 RETURN
2240 '** cruiser 1 sunk **
2250 H=0
2260 P=C1(H)
2280 POKEP,255
2300 H=H+1:IFH<5THEN2260
2310 S=S+500-M*2
2320 GOSUB3300
2330 CA=1
2340 RETURN
2350 '** cruiser 2 sunk **
2360 H=0
2370 P=C2(H)
2390 POKEP,255
2410 H=H+1:IFH<5THEN2370
2420 S=S+500-M*2
2430 GOSUB3300
2440 CB=1
2450 RETURN
2460 '** destroyer 1 sunk **
2470 H=0
2480 P=D1(H)
2500 POKEP,159
2520 H=H+1:IFH<4THEN2480
2530 S=S+400-M*2
2540 GOSUB3300
2550 DA=1
2560 RETURN
2570 '** destroyer 2 sunk **
2580 H=0
2590 P=D2(H)
2610 POKEP,159
2630 H=H+1:IFH<4THEN2590
2640 S=S+400-M*2
2650 GOSUB3300
2660 DB=1
2670 RETURN
2680 '** submarine 1 sunk **
2690 H=0
```

```
2700 P=S1(H)
2720 POKEP,239
2740 H=H+1:IFH<3THEN2700
2750 S=S+300-M*2
2760 GOSUB3300
2770 SC=1
2780 RETURN
2790 '** submarine 2 sunk **
2800 H=0
2810 P=S2(H)
2830 POKEP,239
2850 H=H+1:IFH<3THEN2810
2860 S=S+300-M*2
2870 GOSUB3300
2880 SD=1
2890 RETURN
2900 '** storing ship positions
2910 P=1057+A+B*32
2920 IFT=2THENB1(CT)=P
2930 IFT=3THENB2(CT)=P
2940 IFT=4THENC1(CT)=P
2950 IFT=5THENC2(CT)=P
2960 IFT=6THEND1(CT)=P
2970 IFT=7THEND2(CT)=P
2980 IFT=8THENS1(CT)=P
2990 IFT=9THENS2(CT)=P
3000 RETURN
3010 ' setting squares adjacent
3015 ' to diagonals
3016 IFA=0THEN3030
3020 IFC(A-1,B)=0THENC(A-1,B)=10
ELSE3025
3025 IFA=14THEN3040
3030 IFC(A+1,B)=0THENC(A+1,B)=10
ELSE3040
3040 RETURN
3050 '** all ships sunk **
3060 CLS:PRINT@161,"c o n g r a
t u l a t i o n s";
3070 PRINT:PRINT:PRINT"   you sa
nk the entire fleet!"
3080 GOTO3100
3090 CLS:PRINT@165,"t i m e  i s
  u p";
3100 PRINT:PRINT"YOU TOOK";M;"MO
VES"
3110 PRINT"YOUR SCORE WAS";S;
3120 PRINT:PRINT"THE HIGH SCORE
IS :";HS;
3130 PRINT:PRINT"ANOTHER GAME (Y
/N)";
3140 GOSUB2005
3150 IFIN$="Y"THEN3190
3160 IFIN$="N"THEN3250
3170 GOTO3140
3190 '** playing again **
```

```
3200 IFS>HS THEN HS=S
3210 PRINT@480,"the high score n
ow is :";HS;
3230 FORX=0TO500:NEXTX
3240 GOTO140
3250 '** the end **
3260 CLS:PRINT@163,"i hope you e
njoyed playing";
3270 FORX=0TO50:NEXTX
3280 END
3300 PRINT@337,"SCORE:";S;:GOSUB
3320:RETURN
3320 X=255
3325 SOUNDX,1:X=X-5:IFX=190THENR
ETURNELSE3325
3330 '** initial display **
3331 CLS(0)
3340 B$=STRING$(8,128)+CHR$(138)
+CHR$(129)+CHR$(128)+CHR$(138)+S
TRING$(5,128)
3350 C$=CHR$(144)+STRING$(5,147)
+STRING$(7,159)+STRING$(3,147)+C
HR$(144)
3360 E$=STRING$(14,144)
3370 D$=CHR$(144)+STRING$(5,156)
+STRING$(4,159)+CHR$(144)+STRING
$(2,159)+STRING$(3,156)+CHR$(144
)
3380 Q=334
3390 PRINT@Q,B$;
3400 PRINT@Q+32,C$;
3405 GOSUB3500
3410 Q=Q-1:IFQ>325THEN3390
3420 P=1519
3430 POKEP,154
3435 GOSUB3520
3440 P=P-32:IFP>1359THEN3430
3445 X=100
3450 PRINT@Q,E$;
3460 PRINT@Q+32,B$;
3470 PRINT@Q+64,D$;
3475 GOSUB3530
3480 Q=Q+32:IFQ<426THEN3450
3490 RETURN
3500 SOUND150,1:SOUND50,1:SOUND1
00,1
3510 RETURN
3520 X=X+5:SOUNDX,1:RETURN
3530 X=X-10:SOUNDX,1:RETURN
```

**** MASTERMIND ****
COLOUR COMPUTER

```
10 '*** (C) 17/03/81   D. ZWART
20 '
30 'modified for the tandy color
    computer by
                MICRO-80
40 '
50 CLS:PRINT@7,"** mastermind **
"
60 PRINT:PRINT"BACKSPACE WILL RU
B OUT THE LINE":PRINT"YOU ARE WO
RKING ON. PRESS ^ TO":PRINT"GET
THE ANSWER"
70 PRINT:PRINT"HOW MANY CHARACTE
RS DO YOU WANT":INPUT"ME TO HIDE
 ";X:CLS
80 FORZ=0TO63:SET(Z,0,3):SET(Z,3
1,3):NEXTZ:FORY=0TO31:SET(0,Y,3)
:SET(63,Y,3):NEXTY:PRINT@10,"mas
termind";
90 DIMA$(X),B$(X)
100 PRINT@65," FROM ";:GOSUB300:
A$=IN$:PRINTA$;" TO ";:GOSUB300:
B$=IN$:PRINTB$;
110 A=ASC(A$):B=ASC(B$)
120 IFB<A THEN D=A:A=B:B=D:PRINT
@65," FROM ";B$;" TO ";A$;
130 FORN=1TOX:D=RND(B+1-A)+A-1:A
$(N)=CHR$(D):NEXT
140 PRINT@129,"?";:M=0:O=0
150 FORT=1TOX
160 GOSUB300:D=ASC(IN$):B$(T)=IN
$:B$=IN$
170 IFD=8THENFORS=2TO T:PRINTIN$
;:NEXTS:GOTO140
180 IFD=94THENPRINT@417,"THE ANS
WER WAS ";:PRINT@449,"";:FORS=1T
OX:PRINTA$(S);:NEXT:O=X:GOTO260
190 IFD<A OR D>B THEN160
200 FORN=1TOX
210 IFB$=A$(T)THENO=O+1:GOTO240
220 IFB$=A$(N)THENM=M+1:GOTO240
230 NEXTN
240 PRINTB$;:NEXT T
250 PRINT@225,O;"RIGHT";M;"WRONG
 PLACE";:K=K+1
260 IFO=X THEN PRINT@385,"YOU TO
OK";K;"TURNS";:GOTO280
270 GOTO140
280 FORS=1TO2000:NEXTS:GOSUB290:
RUN
290 CLS(1):SOUND50,1:CLS(2):SOUN
D150,2:CLS(3):SOUND50,2:CLS(4):S
OUND150,5:CLS(5):RETURN
300 IN$=INKEY$:IFIN$=""THEN300
310 IFT=1THENPRINT@225,STRING$(2
9," ");:PRINT@129,STRING$(29," "
);:PRINT@129,"?";
320 SOUND100,1:RETURN
```

**** SINK THE NAVY ****
HITACHI PEACH

```
10 REM ** SINKING THE ENEMY NAVY **  BY
J.C.BENNETT
15 REM MODIFIED FOR THE PEACH BY MICRO-8
0
20 CLEAR 2000:WIDTH80:SCREEN0,1
30 DIM C(14,9)
40 DIM B1(5,1)
50 DIM B2(5,1)
60 DIM C1(4,1)
70 DIM C2(4,1)
80 DIM D1(3,1)
90 DIM D2(3,1)
100 DIM S1(2,1)
110 DIM S2(2,1)
120 LOCATE15,11:PRINT"SINKING THE ENEMY
NAVY"
130 GOSUB3290
140 CLS:LOCATE32,3:PRINT"***  INSTRUCTIO
NS  ***"
150 S=0:M=0:A=0:B=0:CT=0
160 BA=0:BB=0:CA=0:CB=0:DA=0:DB=0:SC=0:S
D=0
170 LOCATE7,4:PRINT"HIGH SCORE :";HS;
180 LOCATE7,7:PRINT"THE ENEMY SHIPS ARE
SOMEWHERE WITHIN RANGE"
190 LOCATE7,8:PRINT"YOUR RANGE IS A GRID
 15 BY 10 SQUARES"
200 LOCATE7,10:PRINT"YOU FIRE BY TYPING
THE COORDINATES OF YOUR TARGET"
210 LOCATE7,12:PRINT"THE ENEMY FLEET CON
SISTS OF :-
220 LOCATE7,13:PRINT"2 BATTLESHIPS EACH
6 SQUARES LONG";
230 LOCATE7,14:PRINT"2 CRUISERS    EACH
5 SQUARES LONG";
240 LOCATE7,15:PRINT"2 DESTROYERS  EACH
4 SQUARES LONG";
250 LOCATE7,16:PRINT"2 SUBMARINES  EACH
3 SQUARES LONG";
260 LOCATE47,12:PRINT"OVER THIS SIDE WIL
L BE..";
270 LOCATE47,13:PRINT"1.HIGH SCORE";
280 LOCATE47,14:PRINT"2.MOVES TAKEN";
290 LOCATE47,15:PRINT"3.YOUR SCORE";
300 LOCATE7,17:PRINT"THE SOONER YOU SINK
 A SHIP THE HIGHER YOUR SCORE"
310 LOCATE7,11:PRINT"YOU HAVE A MAXIMUM
OF 100 MOVES";
320 C(A,B)=0
330 A=A+1:IF A<15 THEN320
340 A=0:B=B+1:IF B<10 THEN320
350 REM ** PLACING SHIPS **
```

```
360 T=2
370 LOCATE3,23:RANDOMIZE:LOCATE0,23:PRIN
TCHR$(26);
380 A=INT(15*RND(14)):B=INT(10*RND(9))
390 D=INT(9*RND(8)):IF D=0THEN390
400 ST=7-INT(T/2)
410 ON D GOSUB850,950,1050,1150,1250,138
0,1510,1640
420 T=T+1:IF T<10 THEN380
430 LOCATE7,18:PRINT"PRESS <P> WHEN YOU
ARE READY TO PLAY";
440 P$=INKEY$
450 IF P$="" THEN440
460 IF P$="P"THEN470 ELSE440
470 CLS:EH=7:EV=3:F=64
480 EH=EH+3:F=F+1
490 LOCATEEH,EV:PRINTCHR$(F);
500 IF F<79 THEN480
510 EH=7:EV=3:F=47
520 EV=EV+1:F=F+1
530 LOCATEEH,EV:PRINTCHR$(F);
540 IF F<57 THEN520
550 EH=7:EV=14:F=64
560 EH=EH+3:F=F+1
570 LOCATEEH,EV:PRINTCHR$(F);
580 IF F<79 THEN560
590 EH=55:EV=3:F=47
600 EV=EV+1:F=F+1
610 LOCATEEH,EV:PRINTCHR$(F);
620 IF F<57 THEN600
630 B1=6:B2=6:C1=5:C2=5:D1=4:D2=4:S1=3:S
2=3
640 LOCATE47,16:PRINT"HIGH SCORE:"HS;
650 REM  ** PLAY **
660 LOCATE7,15:PRINT"BATTLESHIPS = ";B1;
": 6     ";B2;": 6";
670 LOCATE7,16:PRINT"CRUISERS    = ";C1;
": 5     ";C2;": 5";
680 LOCATE7,17:PRINT"DESTROYERS  = ";D1;
": 4     ";D2;": 4";
690 LOCATE7,18:PRINT"SUBMARINES  = ";S1;
": 3     ";S2;": 3";
700 P$=INKEY$
710 IF P$=""THEN700
720 M1=ASC(P$)
730 IF M1<58 AND M1>47 THENB=M1-48 :GOTO
750
740 IF M1<80 AND M1>64 THENA=M1-65 ELSE7
00
750 P$=INKEY$
760 IF P$=""THEN750
770 M2=ASC(P$)
780 IF M2<58 AND M1>47 THENB=M2-48:GOTO8
00
790 IF M2<80 AND M2>64 THENA=M2-65 ELSE7
50
```

```
800 M=M+1:LOCATE47,17:PRINT"MOVE NO.";M;
810 GOSUB1770
820 IF B1=0 AND B2=0 AND C1=0 AND C2=0 A
ND D1=0 AND D2=0 AND S1=0 AND S2=0 THEN3
050
830 IF M=100 THEN3090
840 GOTO660
850 REM ** HORIZONTAL LEFT **
860 SA=A
870 IF A>14-ST THENT=T-1:GOTO940
880 IF C(A,B)>0 THENT=T-1:GOTO940
890 A=A+1:CT=CT+1:IF CT<ST THEN880
900 A=SA:CT=0
910 C(A,B)=T
920 GOSUB2900
930 A=A+1:CT=CT+1:IF CT<ST THEN910
940 CT=0:RETURN
950 REM ** HORIZONTAL RIGHT **
960 SA=A
970 IF A<ST THENT=T-1:GOTO1040
980 IF C(A,B)>0 THENT=T-1:GOTO1040
990 A=A-1:CT=CT+1:IF CT<ST THEN980
1000 A=SA:CT=0
1010 C(A,B)=T
1020 GOSUB2900
1030 A=A-1:CT=CT+1:IF CT<ST THEN1010
1040 CT=0:RETURN
1050 REM ** DOWN **
1060 SB=B
1070 IF B>9-ST THENT=T-1:GOTO1140
1080 IF C(A,B)>0 THENT=T-1:GOTO1140
1090 B=B+1:CT=CT+1:IF CT<ST THEN1080
1100 B=SB:CT=0
1110 C(A,B)=T
1120 GOSUB2900
1130 B=B+1:CT=CT+1:IF CT<ST THEN1110
1140 CT=0:RETURN
1150 REM ** UP **
1160 SB=B
1170 IF B<ST THENT=T-1:GOTO1240
1180 IF C(A,B)>0 THENT=T-1:GOTO1240
1190 B=B-1:CT=CT+1:IF CT<ST THEN1180
1200 B=SB:CT=0
1210 C(A,B)=T
1220 GOSUB2900
1230 B=B-1:CT=CT+1:IF CT<ST THEN1210
1240 CT=0:RETURN
1250 REM ** DIAGONAL UP LEFT **
1260 SA=A:SB=B
1270 IF A<ST OR B<ST THENT=T-1:GOTO1370
1280 IFC(A,B)>0THENT=T-1:GOTO1370
1290 A=A-1:B=B-1:CT=CT+1
1300 IF CT<ST THEN1280
1310 A=SA:B=SB:CT=0
1320 C(A,B)=T
```

```
1330 GOSUB2900
1340 GOSUB3010
1350 A=A-1:B=B-1:CT=CT+1
1360 IF CT<ST THEN1320
1370 CT=0:RETURN
1380 REM ** DIAGONAL UP RIGHT **
1390 SA=A:SB=B
1400 IF A>14-ST OR B<ST THENT=T-1:GOTO15
00
1410 IF C(A,B)>0 THENT=T-1:GOTO1500
1420 A=A+1:B=B-1:CT=CT+1
1430 IF CT<ST THEN1410
1440 A=SA:B=SB:CT=0
1450 C(A,B)=T
1460 GOSUB2900
1470 GOSUB3010
1480 A=A+1:B=B-1:CT=CT+1
1490 IF CT<ST THEN1450
1500 CT=0:RETURN
1510 REM ** DIAGONAL DOWN LEFT **
1520 SA=A:SB=B
1530 IF B<ST OR A<ST THENT=T-1:GOTO1630
1540 IF C(A,B)>0 THENT=T-1:GOTO1630
1550 A=A-1:B=B-1:CT=CT+1
1560 IF CT<ST THEN1540
1570 A=SA:B=SB:CT=0
1580 C(A,B)=T
1590 GOSUB2900
1600 GOSUB3010
1610 A=A-1:B=B-1:CT=CT+1
1620 IF CT<ST THEN1580
1630 CT=0:RETURN
1640 REM ** DIAGONAL DOWN RIGHT **
1650 SA=A:SB=B
1660 IF B>9-ST OR A>14-ST THENT=T-1:GOTO
1760
1670 IF C(A,B)>0 THENT=T-1:GOTO1760
1680 A=A+1:B=B+1:CT=CT+1
1690 IF CT<ST THEN1670
1700 A=SA:B=SB:CT=0
1710 C(A,B)=T
1720 GOSUB2900
1730 GOSUB3010
1740 A=A+1:B=B+1:CT=CT+1
1750 IF CT<ST THEN1710
1760 CT=0:RETURN
1770 REM ** PLAYER'S MOVE **
1780 P=C(A,B)
1790 C(A,B)=1
1800 PH=10+A*3:PV=4+B
1810 IF P=0 OR P=10 THENLOCATEPH,PV:PRIN
T"X";:BEEP:GOTO2010
1820 IF P=1 THEN2010
1830 IF P>1 ANDP<10 THENLOCATEPH,PV:PRIN
T"*";:BEEP:BEEP:BEEP
1840 S=S+10:LOCATE47,18:PRINT"SCORE:";S;
```

```
1850 IF B1>0 AND P=2 THENB1=B1-1
1860 IF B2>0 AND P=3 THENB2=B2-1
1870 IF C1>0 AND P=4 THENC1=C1-1
1880 IF C2>0 AND P=5 THENC2=C2-1
1890 IF D1>0 AND P=6 THEND1=D1-1
1900 IF D2>0 AND P=7 THEND2=D2-1
1910 IF S1>0 AND P=8 THENS1=S1-1
1920 IF S2>0 AND P=9 THENS2=S2-1
1930 IF B1=0 AND BA=0THENGOSUB2020
1940 IF B2=0 AND BB=0 THENGOSUB2130
1950 IF C1=0 AND CA=0THENGOSUB2240
1960 IF C2=0 AND CB=0THENGOSUB2350
1970 IF D1=0 AND DA=0THENGOSUB2460
1980 IF D2=0 AND DB=0THENGOSUB2570
1990 IF S1=0 AND SC=0THENGOSUB2680
2000 IF S2=0 AND SD=0THENGOSUB2790
2010 RETURN
2020 REM ** BATTLESHIP SUNK **
2030 H=0
2040 PH=B1(H,0):PV=B1(H,1)
2050 GOSUB3500
2080 H=H+1:IF H<6 THEN2040
2090 S=S+600-M*2
2100 LOCATE47,18:PRINT"SCORE:";S;
2110 BA=1
2120 RETURN
2130 REM ** BATTLESHIP 2 SUNK **
2140 H=0
2150 PH=B2(H,0):PV=B2(H,1)
2160 GOSUB3500
2190 H=H+1:IF H<6 THEN2150
2200 S=S+600-M*2
2210 LOCATE47,18:PRINT"SCORE:";S;
2220 BB=1
2230 RETURN
2240 REM ** CRUISER 1 SUNK **
2250 H=0
2260 PH=C1(H,0):PV=C1(H,1)
2270 GOSUB3500
2300 H=H+1:IF H<5 THEN2260
2310 S=S+500-M*2
2320 LOCATE47,18:PRINT"SCORE:";S;
2330 CA=1
2340 RETURN
2350 REM ** CRUISER 2 SUNK **
2360 H=0
2370 PH=C2(H,0):PV=C2(H,1)
2380 GOSUB3500
2410 H=H+1:IF H<5 THEN2370
2420 S=S+500-M*2
2430 LOCATE47,18:PRINT"SCORE:";S;
2440 CB=1
2450 RETURN
2460 REM ** DESTROYER 1 SUNK **
2470 H=0
2480 PH=D1(H,0):PV=D1(H,1)
2490 GOSUB3500
2520 H=H+1:IF H<4 THEN2480
2530 S=S+400-M*2
2540 LOCATE47,18:PRINT"SCORE:";S;
2550 DA=1
2560 RETURN
2570 REM ** DESTROYER 2 SUNK **
2580 H=0
2590 PH=D2(H,0):PV=D2(H,1)
2600 GOSUB3500
2630 H=H+1:IF H<4 THEN2590
2640 S=S+400-M*2
2650 LOCATE47,18:PRINT"SCORE:";S;
2660 DB=1
2670 RETURN
2680 REM ** SUBMARINE 1 SUNK **
2690 H=0
2700 PH=S1(H,0):PV=S1(H,1)
2710 GOSUB3500
2740 H=H+1:IF H<3 THEN2700
2750 S=S+300-M*2
2760 LOCATE47,18:PRINT"SCORE:";S;
2770 SC=1
2780 RETURN
2790 REM ** SUBMARINE 2 SUNK **
2800 H=0
2810 PH=S2(H,0):PV=S2(H,1)
2820 GOSUB3500
2850 H=H+1:IF H<3 THEN2810
2860 S=S+300-M*2
2870 LOCATE47,18:PRINT"SCORE:";S;
2880 SD=1
2890 RETURN
2900 REM ** STORING SHIP POSITIONS **
2910 PH=10+A*3:PV=4+B
2920 IF T=2 THENB1(CT,0)=PH:B1(CT,1)=PV
2930 IF T=3 THENB2(CT,0)=PH:B2(CT,1)=PV
2940 IF T=4 THENC1(CT,0)=PH:C1(CT,1)=PV
2950 IF T=5 THENC2(CT,0)=PH:C2(CT,1)=PV
2960 IF T=6 THEND1(CT,0)=PH:D1(CT,1)=PV
2970 IF T=7 THEND2(CT,0)=PH:D2(CT,1)=PV
2980 IF T=8 THENS1(CT,0)=PH:S1(CT,1)=PV
2990 IF T=9 THENS2(CT,0)=PH:S2(CT,1)=PV
3000 RETURN
3010 REM ** SETTING SQUARES ADJACENT TO
DIAGONALS **
3015 IF A=0 THEN 3030
3020 IFC(A-1,B)=0 THENC(A-1,B)=10 ELSE30
25
3025 IF A=14 THEN 3040
3030 IFC(A+1,B)=0 THENC(A+1,B)=10 ELSE30
40
3040 RETURN
3050 REM ** ALL SHIPS SUNK **
3060 CLS: LOCATE29,5:PRINT"C O N G R A T
 U L A T I O N S !"
3070 LOCATE35,7:PRINT"YOU SANK THE ENTIR
E FLEET!"
3080 GOTO3100
3090 CLS:LOCATE29,5:PRINT"T I M E    I S
   U P "
3100 LOCATE7,10:PRINT"YOU TOOK";M;"MOVES
"
3110 LOCATE7,11:PRINT"YOUR SCORE WAS";S;
3120 LOCATE7,13:PRINT"THE HIGH SCORE IS
:";HS;
3130 LOCATE7,16:PRINT"PRESS <P> IF YOU W
OULD LIKE ANOTHER GAME"
3140 LOCATE7,18:PRINT"PRESS <F> IF YOU W
OULD LIKE TO FINISH";
3150 P$=INKEY$
3160 IF P$=""THEN3150
3170 IF P$="P"THEN3190
3180 IF P$="F"THEN3250 ELSE3150
3190 REM ** PLAYING AGAIN **
3200 IF S>HS THENHS=S
3210 LOCATE7,18:PRINTSTRING$(40,32);
3220 LOCATE7,18:PRINT"THE HIGH SCORE NOW
 IS :";HS;
3230 FOR X=0TO250:NEXT X
3240 GOTO140
3250 REM ** END **
3260 CLS:LOCATE15,11:PRINT"I HOPE YOU EN
JOYED PLAYING"
3270 FOR X=0TO50:NEXT X
3280 END
3290 REM ** INITIAL DISPLAY **
3300 B$=STRING$(8,32)+CHR$(133)+CHR$(138
)+CHR$(32)+CHR$(133)+STRING$(5,32)
3310 C$=CHR$(32)+STRING$(5,130)+STRING$(
7,134)+STRING$(3,130)+CHR$(32)
3320 E$=STRING$(14,32)
3330 D$=CHR$(32)+STRING$(5,130)+STRING$(
4,134)+CHR$(32)+STRING$(2,134)+STRING$(3
,130)+CHR$(32)
3340 Q=60
3350 LOCATEQ,15:PRINTB$;
3360 LOCATEQ,16:PRINTC$;
3370 Q=Q-1:IF Q>30 THEN3350
3380 P=24
3390 LOCATE40,P:PRINTCHR$(132);
3400 P=P-1:IF P>16 THEN3390
3405 BEEP:BEEP:BEEP
3410 LOCATEQ,P-2:PRINTE$;
3420 LOCATEQ,P-1:PRINTB$;
3430 LOCATEQ,P:PRINTD$;
3440 P=P+1:IF P<24 THEN3410
3450 RETURN
3500 LOCATEPH-1,PV:PRINTCHR$(135);
3510 LOCATEPH,PV:PRINTCHR$(135);
3520 LOCATEPH+1,PV:PRINTCHR$(135);
3530 RETURN
```

**** MASTERMIND ****
HITACHI PEACH

```
2 REM (C) 17/03/81 D. ZWART
3 REM MODIFIED FOR THE PEACH BY MICRO-80
5 CLS:PRINT:PRINT"      # MASTERMIND #":P
RINT:PRINT"BACKSPACE WILL RUB OUT THE LI
NE YOU ARE WORKING ON"
10 PRINT:PRINT"'ESC' WILL GIVE YOU THE A
NSWER":PRINT
15 RANDOMIZE:PRINT
20 INPUT"HOW MANY CHARACTERS TO HIDE";X
30 DIMA$(X),B$(X)
40 PRINT:PRINT"FROM ";
50 A$=INKEY$:IFA$=""THEN50
60 PRINTA$" TO ";
70 B$=INKEY$:IFB$=""THEN70
80 PRINTB$
90 A=ASC(A$):B=ASC(B$)
100 IFB<ATHEND=A:A=B:B=D
110 FORN=1TOX:D=INT((B-A+1)*RND(X))+A:A$
(N)=CHR$(D):NEXT
120 M=0:O=0
130 FORT=1TOX
140 B$=INKEY$:IFB$=""THEN140
150 D=ASC(B$):B$(T)=B$
160 IFD=8THENFORS=2TOT:PRINTB$;:NEXTS:GO
TO120
170 IFD=27THENPRINT:PRINT"THE ANSWER WAS
":FORS=1TOX:PRINTA$(S);:NEXTS:O=X:GOTO25
0
180 IFD<AORD>BTHEN140
190 FORN=1TOX
200 IFB$=A$(T)THENO=O+1:GOTO230
210 IFB$=A$(N)THENM=M+1:GOTO230
220 NEXTN
230 PRINTB$;:NEXTT
240 PRINTO;"RIGHT";M;"WRONG PLACE":K=K+1
250 IF X<>O THEN120
260 PRINT:PRINT"YOU TOOK";K;"TURNS":FORS
=1TO4000:NEXTS:RUN
```

**** LII/16K   GOLF ****

TRS-80/SYSTEM-80

```
10 DATA205,127,10,125,217,1,0,4,254,1,40,8,17,0,120,33,0,60
20 DATA24,6,17,0,60,33,0,120,237,176,217,201
30 LL$="THIS IS A DUMMY STRING FOR USE"
40 LL=VARPTR(LL$):LM=PEEK(LL+1)+PEEK(LL+2)*256
50 FORLN=LMTOLM+29:READLO:POKELN,LO:NEXTLN
60 POKE16526,(PEEK(LL+1)):POKE16527,(PEEK(LL+2))
70 REM GOLF
80 REM COPYRIGHT - ROBERT GLUCZ
90 REM 1/1707 DANDENONG RD. OAKLEIGH, VIC. 3166
100 CLS:PRINT"WELCOME TO THE GOLF COURSE OF THE 80'S! "
110 FORXX=1TO1500:NEXT:CLEAR500
120 PRINT:PRINT:PRINT"TWO PLAYERS ARE REQUIRED."
130 PRINT:INPUT"PLEASE INTRODUCE YOURSELVES";P1$,P2$
140 CLS:PRINT"HELLO ";P1$;" AND ";P2$
150 PRINT:INPUT"WOULD YOU LIKE ME TO EXPLAIN THE LOCAL COURSE RU
LES";Q$
160 IFLEFT$(Q$,1)="N"THEN510
170 GOSUB2800
180 CLS:PRINT@21,"80 COUNTRY CLUB RULES."
190 PRINT@85,"----------------------"
200 PRINT:PRINT"FOR EACH FAIRWAY SHOT YOU MUST SELECT:"
210 PRINT:PRINT"-- THE APPROPRIATE CLUB"
220 PRINT"-- DIRECTION IN WHICH TO HIT"
230 PRINT"-- STRENGTH TO BE USED"
240 GOSUB360
250 GOSUB260:GOSUB370:GOSUB440:GOTO510
260 CLS:PRINT"CLUBS AVAILABLE ARE:"
270 PRINT:PRINT"1  WOOD   = 140 METRES"
280 PRINT"2  WOOD   = 120"
290 PRINT"3  IRON   = 105"
300 PRINT"4  IRON   = 90"
310 PRINT"5  IRON   = 70"
320 PRINT"6  IRON   = 50"
330 PRINT"7  IRON   = 35"
340 PRINT"8 =WEDGE  = 15"
350 PRINT:PRINT"ALL CLUBS ARE SELECTED BY THEIR NUMBERS.":PRINT"
IF YOU TRY TO USE AN INCORRECT CLUB;":PRINT"-EG. WOOD FROM SAND
TRAP":PRINT"YOU WILL BE ADVISED TO USE ANOTHER CLUB.":GOSUB360:R
ETURN
360 PRINT@960,"PRESS ANY KEY TO CONTINUE";:IFINKEY$=""THEN360ELS
E RETURN
370 CLS:PRINT"DIRECTION IS SPECIFIED BY A NUMBER BETWEEN 0 AND 3
60 DEGREES."
380 PRINT:PRINT"DIRECTIONS ARE AS SHOWN BELOW:"
390 X=46:Y=24:FORA=1TO34:SET(X,Y):X=X+1:NEXTA
400 X1=62:X2=63:Y=16:FORA=1TO17:SET(X1,Y):SET(X2,Y):Y=Y+1:NEXTA
410 X=46:Y1=16:Y2=32:FORA=1TO17:SET(X,Y1):SET(X+1,Y1):SET(X,Y2):
SET(X+1,Y2):X=X+2:Y1=Y1+1:Y2=Y2-1:NEXTA
420 PRINT@276,"135";TAB(31)"90";TAB(40)"45";:PRINT@531,"180";:PR
INT@553,"0";:PRINT@724,"225";TAB(30)"270";TAB(39)"315";
430 GOSUB360:RETURN
440 CLS:PRINT"H I N T S."
450 PRINT"--AS STRENGTH INCREASES, SO DOES THE CHANCE OF HOOKING
 OR":PRINT"SLICING. DON'T USE STRENGTH UNNECESSARILY."
460 PRINT:PRINT"--TRY TO LAND IN CENTRE OF GREEN. THIS PUTS YOU
CLOSER TO THE":PRINT"HOLE, AND ALSO LETS YOUR OPPONENT GO FIRST,
 ALLOWING YOU":PRINT"TO SEE THE SLOPE AND CUT VALUES BEFORE PUTT
ING."
470 PRINT:PRINT"--WATCH YOUR OPPONENT'S SHOT; IT MAY GIVE YOU CL
UES."
480 PRINT:PRINT"--TRY NOT TO LAND IN WATER OR PAST THE BOUNDARY
- THIS ADDS":PRINT"PENALTY STROKES."
490 PRINT:PRINT"IF YOU ARE ON YOUR OWN, YOU CAN CHEAT BY REPLAYI
NG BAD SHOTS":PRINT"THIS, OF COURSE, IS NOT RECOMMENDED!!"
```

```
500 GOSUB360:RETURN
510 CLS:PRINT"WHAT IS YOUR HANDICAP ";P1$;" (0-30)";:INPUTH1:IFH
1<0ORH1>30THEN510
520 PRINT"AND YOURS ";P2$;:INPUTH2:IFH2<0ORH2>30THEN520
530 PRINT:PRINT:IFH1<10ANDH2<10PRINT"A SEASONED PAIR OF GOLFERS,
 I SEE.":PRINT"YOU WON'T NEED PUTTING PRACTICE THEN!":FORXX=1TO1
000:NEXTXX:GOTO570
540 INPUT"WOULD YOU LIKE SOME PUTTING PRACTICE";Q2$
550 IFLEFT$(Q2$,1)="Y"THEN560ELSE570
560 PG=1:PRINT"OFF TO THE PRACTICE GREEN THEN!":GOTO1650
570 CLS:PRINT"READY TO TEE OFF THEN ARE WE?":FORXX=1TO10:FORXY=1
TO20:NEXTXY,XX
580 PRINT:PRINT:INPUT"HOW MANY HOLES WOULD YOU LIKE TO PLAY";HO
590 PRINT:PRINT"OFF YOU GO THEN, AND GOOD LUCK TO BOTH OF YOU."
600 PRINT"I'LL SEE YOU AT THE NINETEENTH HOLE!!":GOSUB360:RANDOM
:H5=1:QP$="YOU'VE DUFFED IT!! REPLACE YOUR DIVOT!"
610 TPAR=0:SC=0:SD=0:SE=1
620 FORHO=H5TOHO:CLS:PR=0:PS=0:O1=0:O2=0:PI=0:PZ=0:PA=0:PB=0:PC=
0:PD=0:PE=0:PF=0:U=120:V=25:U1=U:V1=V:U2=U:V2=V:A0=RND(0):RDOG=0
:LDOG=0:LT=0:RT=0:LW=0:RW=0:RR=0:LR=0:WT=0:ST=0:NT=0:IFA0<.2THEN
PAR=3:DIST=RND(20)+40:GOTO650
630 IFA0>.8THENPAR=5:DIST=RND(30)+80:GOTO650
640 PAR=4:DIST=RND(20)+70
650 TPAR=TPAR+PAR:IFPAR=3THENANG=0:LDOG=1:RDOG=1:GOTO680
660 ANG=RND(10):DOG=RND(0):IFDOG<.35THENRDOG=1:ANG=-ANG
670 IFDOG>.65THENLDOG=1:GOTO680
680 L=RND(0):IFL<.4THENLT=1:GOTO700
690 IFL>.7THENLW=1ELSELR=1
700 R=RND(0):IFR<.4THENRT=1:GOTO720
710 IFR>.7THENRW=1ELSERR=1
720 TRAP=RND(0):IFTRAP<.4THENST=1:GOTO750
730 IFTRAP>.8THENWT=1:GOTO750
740 NT=1:AP=0:AQ=0:GOTO760
750 AP=RND(3)*10+20:AQ=RND(3)*10+50
760 HP=DIST+ABS(ANG)+3
770 PRINT@8,"HOLE  NO. ";HO;"    PAR =";PAR;"    LENGTH =";INT(HP*
3.5);"MTRS."
780 IFLDOG=1THENX6=15552:X7=15807:GOSUB2580:GOTO800
790 X6=15584:X7=15615:FORY=1TO4:FORX=X6TOX7:POKEX,191:NEXTX:X6=X
6+65:X7=X7+64:NEXTY
800 IFRDOG=1THENX6=16000:X7=16255:GOSUB2580:GOTO820
810 X6=16224:X7=16255:FORY=1TO4:FORX=X6TOX7:POKEX,191:NEXTX:X6=X
6-63:X7=X7-64:NEXTY
820 IFLT=1THENPB$="TREES":GOTO840
830 IFLW=1THENPB$="WATER"ELSEPB$="ROUGH"
840 IFRT=1THENPT$="TREES":GOTO860
850 IFRW=1THENPT$="WATER"ELSEPT$="ROUGH"
860 PRINT@227,PT$;:PRINT@867,PB$;
870 FORY=24TO26:FORX=120TO127:SET(X,Y):NEXTX,Y
880 IFANG=0THENY1=22:Y2=27:GOTO900
890 Y1=25+ANG
900 X1=128-(DIST+5):X2=128-(DIST-6):X3=128-(DIST+8):X4=128-(DIST
+13)
910 IFANG<0THENY2=25+(ANG-6)
920 IFANG>0THENY2=25+(ANG+6)
930 IFY1>Y2THENY8=Y2:Y2=Y1:Y1=Y8
940 GOSUB950:GOTO970
950 F=2:FORY=Y1TOY1+1:FORX=X1+FTOX2-F:SET(X,Y):NEXTX:F=1:NEXTY:F
ORY=Y1+2TOY2-2:FORX=X1TOX2:SET(X,Y):NEXTX,Y:FORY=Y2-1TOY2:FORX=X
1+FTOX2-F:SET(X,Y):NEXTX:F=2:NEXTY
960 FORY=Y1TOY2:FORX=X4TOX3:SET(X,Y):NEXTX,Y:RETURN
970 IFNT=1THEN1030
980 IFDIST<=AQ+10THENAQ=DIST-10
990 IFPAR=3THENAP=15
1000 FORY=23TO27:FORX=128-AQTO128-AP:SET(X,Y):NEXTX,Y
1010 IFST=1ANDPAR<>3PRINT@547,"SAND";ELSEIFST=1ANDPAR=3PRINT@560
,"SAND";
1020 IFWT=1ANDPAR<>3PRINT@547,"WATER";ELSEIFWT=1ANDPAR=3PRINT@56
0,"WATER";
1030 SA=S1:SB=S2:IFSC<SDTHENP1=1:SE=1ELSEIFSD<SCTHENP1=0:SE=0ELS
EIFSC=SDANDSE=1THENP1=1ELSEP1=0
1040 F=USR(0)
1050 IFP1=1THENGOSUB2510ELSEGOSUB2520
1060 GOSUB2570:PRINT@896,PX$;"'S  SHOT. ENTER CLUB,DIRECTION,STR
ENGTH.";:INPUTCL,DR,SR
1070 IFCL=9THEN2590
1080 IFCL<1ORCL>8ORDR<0ORDR>360ORSR<0ORSR>9THEN1060
1090 IFP1=1ANDPE=1ANDCL<>8ORP1=0ANDPF=1ANDCL<>8ORP1=1ANDPC=1ANDC
L<>8ORP1=0ANDPD=1ANDCL<>8THENPRINT@968,"TRY USING A WEDGE ";PX$;
"                                ";:FORXX=1TO1000:NEXTXX:GOTO1060
1100 IFP1=1ANDPK=1ANDCL<3ORP1=0ANDPB=1ANDCL<3PRINT@968,"TRY USIN
G AN IRON  ";PX$;"                                     ";:FORXX=1TO1000
:NEXTXX:GOTO1060
1110 IF(P1=1ANDPC=1)OR(P1=0ANDPD=1)THEN1120ELSE1130
1120 TH=RND(0):IFTH>.9THEN1130ELSECD=5:SR=((9-CL)*5-2+SR)/2:GOTO
1140
1130 CD=(9-CL)*5
1140 IFP1=1THENPK=0:PC=0:PE=0ELSEPB=0:PD=0:PF=0
1150 QL=0:QP=RND(0):IFQP>.05THEN1160ELSESR=0:CD=5:QL=1
1160 IFSR=0THENFADE=0ELSEFADE=SR*5
1170 SR=2*SR:AB=RND(0)
1180 IFAB<.3THENFL=1:GOTO1200
1190 IFAB>.7THENFR=1ELSEFADE=0
1200 IFP1=1THENU=U1:V=V1:GOSUB1220:GOSUB1230:GOTO1250
1210 U=U2:V=V2:GOSUB1220:GOSUB1230:GOTO1250
1220 X=COS(DR*0.0174533):Y=SIN(DR*0.0174533)*0.375:RETURN
1230 FORW=0TOCD:RESET(U,V):SET(U+X,V-Y):U=U+X:V=V-Y:IFU>120THENU
=120:V=V+YELSEIFU<0THENU=0:V=V+YELSEIFV<9THENV=9:U=U-XELSEIFV>41
THENV=41:U=U-X
1240 NEXTW:RETURN
1250 IFQL=1THENGOSUB2470:PRINT@72,QP$;
1260 IFFR=1THENDR=DR-FADE:GOTO1280
1270 IFFL=1THENDR=DR+FADE
1280 IFSR=0THEN1290ELSECD=SR:GOSUB1220:GOSUB1230:FR=0:FL=0
1290 GOSUB1300:GOTO1510
1300 IFU=0ORV=9ORV=41ORU=128THENPN=1:IFP1=1THENPP=1ELSEPQ=1
1310 GOSUB950
1320 A=POINT(U-1,V):B=POINT(U+1,V):C=POINT(U,V-1):D=POINT(U,V+1)
:IFA=0ANDB=0ANDC=0ORA=0ANDB=0ANDD=0ORA=0ANDC=0ANDD=0ORB=0ANDC=0A
NDD=0THEN1490
1330 IF(LDOG=1ORU>64)ANDV<=20THEN1350ELSEIF(RDOG=1ORU>64)ANDV>=3
0THEN1360
```

```
1340 IFX1<=UANDX2>=UANDY1<=VANDY2>=VTHEN1470ELSE1370
1350 IFRT=1THEN1410ELSEIFRW=1THENV=21:GOTO1450ELSE1390
1360 IFLT=1THEN1410ELSEIFLW=1THENV=29:GOTO1450ELSE1390
1370 IFWT=1THENU=127-AQ:GOTO1450
1380 GOTO1430
1390 GOSUB2470:PRINT@72,PX$;" IS IN THE ROUGH.";:IFP1=1THENPK=1E
LSEPB=1
1400 GOTO1500
1410 GOSUB2470:PRINT@72,PX$;" HAS LANDED AMONGST TREES.";:IFP1=1
THENPC=1ELSEPD=1
1420 GOTO1500
1430 GOSUB2470:PRINT@72,PX$;" IS IN A SAND TRAP.";:IFP1=1THENPE=
1ELSEPF=1
1440 GOTO1500
1450 GOSUB2470:PRINT@72,PX$;" HAS LANDED IN WATER. PENALTY ADDED
.";:IFP1=1THENS1=S1+1:PP=1ELSES2=S2+1:PQ=1
1460 GOTO1500
1470 GOSUB2470:PRINT@72,PX$;" IS ON THE GREEN.  GOOD SHOT!!";:IF
P1=1THENPI=1ELSEPZ=1
1480 IFU>X1+3ANDU<X2-3ANDV>Y1+1ANDV<Y2-1THENIFP1=1THENPR=1ELSEPS
=1:GOTO1500
1490 IFPN=1THENPN=0:GOSUB2470:PRINT@72,PX$;" HAS HIT PAST BOUNDA
RY. PENALTY ADDED.";:IFP1=1THENS1=S1+1:PQ=0ELSES2=S2+1:PP=0
1500 RETURN
1510 GOSUB2570:PRINT@904,"REPLAY SHOT?  (Y/N)";:INPUTF1$
1520 IFLEFT$(F1$,1)="Y"THEN1530ELSE1560
1530 CLS:F=USR(1)
1540 GOSUB2720:GOSUB2770:GOSUB2740:GOTO2760
1550 GOSUB1300:GOSUB2720:GOSUB2880:GOTO1050
1560 F=USR(1):IFP1=1THENU1=U:V1=V:S1=S1+1:GOSUB2880:GOTO1580
1570 U2=U:V2=V:S2=S2+1:GOSUB2880
1580 IFPI=1ANDPZ=1THENPEN=0:GOTO1640
1590 IFP1=1THENP1=0ELSEP1=1
1600 PEN=0
1610 IFP1=1ANDPI=1THENP1=0
1620 IFP1=0ANDPZ=1THENP1=1
1630 GOTO1050
1640 GOSUB2470:PRINT@72,"CONGRATULATIONS, YOU'RE BOTH ON THE GRE
EN!"
1650 FORXX=1TO2000:NEXTXX:CLS:Y=6
1660 GOSUB2500:Y=39:GOSUB2500
1670 AC=RND(40)+40:AD=RND(15)+14:PM=0
1680 AG=RND(10)*2.5:AH=RND(0):AJ=RND(10)/10:AK=RND(0):PRINT@971,
"CUT = ?           SLOPE = ?";:GOSUB1690:GOTO1700
1690 F=2:FORY=ADTOAD+1:FORX=AC+FTOAC+9-F:SET(X,Y):NEXTX:F=1:NEXT
Y:FORY=AD+2TOAD+3:FORX=ACTOAC+9:SET(X,Y):NEXTX,Y:FORY=AD+4TOAD+5
:FORX=AC+FTOAC+9-F:SET(X,Y):NEXTX:F=2:NEXTY:RETURN
1700 IFPR=1THENGOSUB1720ELSEGOSUB1730
1710 U1=AE:V1=AF:PR=0:GOTO1760
1720 AE=RND(40)+40:AF=RND(15)+14:RETURN
1730 AR=RND(0):AS=RND(0):IFAR<.5THENAE=RND(40)+1ELSEAE=RND(47)+7
7
1740 IFAS<.5THENAF=RND(9)+7ELSEAF=RND(9)+27
1750 RETURN
1760 IFPS=1THENGOSUB1720ELSEGOSUB1730:U2=AE:V2=AF:PS=0
1770 IFU1-AC>0THENR1=1ELSER1=0:IFU2-AC>0THENR2=1ELSER2=0
1780 IFR1=1ANDR2=1ANDU1>U2ORR1=0ANDR2=0ANDU2>U1ORR1=1ANDR2=0ANDA
C-U2<U1-ACORR1=0ANDR2=1ANDU2-AC<AC-U1THENP1=1ELSEP1=0
1790 SET(U1,V1):SET(U1+1,V1):SET(U2,V2):SET(U2+1,V2)
1800 FORBB=1TO2:GOSUB1910:GOSUB2060:IFP1=1THENP1=0ELSEP1=1
1810 NEXTBB:IFO1=1ANDO2=1THEN2120
1820 IFO2=1THENO1=1:O2=0
1830 IFO1=1THENO2=1:O1=0
1840 PRINT@6,"GET OUT YOUR PUTTERS!!";:FORXX=1TO1000:NEXTXX
1850 GOSUB2480:GOSUB2490
1860 IFPM=1THENPRINT@960,"CUT =";AG;"DEG'S ";AH$;"    SLOPE =";A
J/2;"* STRENGTH, ";SL$;" (DEG'S)";
1870 GOSUB2530
1880 GOSUB2560:PRINT@0,PX$;"'S SHOT. ENTER DIR (0-360), STRENGTH
 (1-100)";:INPUTDR,ST1
1890 IFST1<1ORST1>100ORDR<0ORDR>360THEN1880
1900 GOSUB1910:GOTO1940
1910 IFP1=1THENU=U1:V=V1:GOTO1930
1920 U=U2:V=V2
1930 RETURN
1940 IFST1<15THEN1980ELSEST1=ST1-15:SLE=(AJ/2)*ST1:IFAK<=.5THENS
L$="UP":IFU<AC+2THENDR=DR+SLEELSEDR=DR-SLE
1950 IFAK>.5THENSL$="DOWN":IFU<AC+2THENDR=DR-SLEELSEDR=DR+SLE
1960 GOSUB1220:GOSUB1990:IFAH<.5THENAH$="LEFT":DR=DR+AGELSEAH$="
RIGHT":DR=DR-AG
1970 ST1=15
1980 GOSUB1220:GOSUB1990:GOTO2010
1990 FORQ=1TOST1:RESET(U,V):RESET(U+1,V):SET(U+X,V-Y):SET(U+X+1,
V-Y):U=U+X:V=V-Y:IFU<1THENU=1ELSEIFU>126THENU=126ELSEIFV<7THENV=
7ELSEIFV>37THENV=37ELSENEXTQ
2000 RETURN
2010  GOSUB1690
2020 IFP1=1THENU1=U:V1=V:S1=S1+1:GOTO2040
2030 U2=U:V2=V:S2=S2+1
2040 IFP1=0THENP1=1ELSEP1=0
2050 PM=1:GOSUB2060:IFO1=1ANDO2=1THEN2120ELSE1850
2060 A=POINT(U+2,V):B=POINT(U-1,V):C=POINT(U,V-1):D=POINT(U,V+1)
:IFV<37ANDV>7AND((A=-1ANDC=-1)OR(A=-1ANDD=-1)OR(B=-1ANDC=-1)OR(B
=-1ANDD=-1))THENOO=1ELSEOO=0
2070 IFOO=1ANDP1=1THENO2=1
2080 IFOO=1ANDP1=0THENO1=1
2090 IFO1=1THENP1=0
2100 IFO2=1THENP1=1
2110 RETURN
2120 IFPG=1THENPG=0:S1=0:S2=0:O1=0:O2=0:CLS:INPUT"HAVE YOU HAD E
NOUGH PRACTICE";Q3$:IFLEFT$(Q3$,1)="N"THENPG=1:GOTO1650ELSE570
2130 FORXX=1TO2000:NEXTXX:CLS
2140 SC=S1-SA:SD=S2-SB
2150 PRINT"SCORES AFTER HOLE NO.";HO:GOSUB2160:GOTO2210
2160 IFS1>TPARTHENS1$="OVER":SX=S1-TPAR:GOTO2180
2170 IFS1<TPARTHENS1T="UNDER":SX=TPAR-S1ELSES1$=" =PAR":SX=S1
2180 IFS2>TPARTHENS2$="OVER":SY=S2-TPAR:GOTO2200
2190 IFS2<TPARTHENS2$="UNDER":SY=TPAR-S2ELSES2$=" =PAR":SY=S2
2200 RETURN
2210 PRINT:PRINTP1$;" =";S1;"   (";SX;S1$;")"
2220 PRINT:PRINTP2$;" =";S2;"   (";SY;S2$;")"
2230 IFSC=PARTHENPRINT@512,"GOOD PAR ";P1$
```

```
2240 IFSC+1=PARTHENPRINT@512,"GREAT PLAY ";P1$;". A BIRDIE."
2250 IFSC+2=PARANDPAR<>3THENPRINT@512,"FANTASTIC, ";P1$;". AN EA
GLE!!"
2260 IFSC+2=PARANDPAR=3THENPRINT@512,"A HOLE-IN-ONE BY ";P1$;"!!
!!"
2270 IFSD=PARTHENPRINT@640,P2$;" PARRED THAT HOLE."
2280 IFSD+1=PARTHENPRINT@640,"A BIRDIE FOR ";P2$
2290 IFSD+2=PARANDPAR<>3THENPRINT@640,"AMAZING, ";P2$;". AN EAGL
E."
2300 IFSD+2=PARANDPAR=3THENPRINT@640,"WHAT A SHOT. ";P2$;" GOT A
 HOLE-IN-ONE!!!!!"
2310 GOSUB360:IFHO+1>HOTHEN2340
2320 PRINT:PRINT"OFF WE GO TO HOLE NO.";HO+1
2330 H5=HO+1:ONERRORGOTO2790:FORXX=1TO1500:NEXTXX
2340 NEXTHO
2350 CLS:PRINT@96,"B A R"
2360 PRINT:PRINT"WELL HERE WE ARE AT THE NINETEENTH HOLE!"
2370 PRINT:PRINT"CAN I SEE YOUR SCORE CARDS?"
2380 S1=S1-INT(HO/18*H1):S2=S2-INT(HO/18*H2)
2390 GOSUB2160
2400 PRINT:PRINTP1$;" =";S1;"   (HANDICAP ADJUSTED)   ";SX;S1$
2410 PRINTP2$;" =";S2;"   (HANDICAP ADJUSTED)   ";SY;S2$
2420 IFS1<S2THENSP$=P1$ELSEIFS2<S1THENSP$=P2$ELSESP$="BOTH OF YO
U."
2430 PRINT:PRINT"CONGRATULATIONS, ";SP$
2440 PRINT:INPUT"ANOTHER ROUND";Q4$:IFQ4$="Y"THEN100
2450 PRINT:PRINT"ANOTHER ALE, THEN?"
2460 END
2470 PRINT@72,STRING$(64,32);:RETURN
2480 PRINT@INT(1+V2/3)*64+INT(U2/2),LEFT$(P2$,1);:RETURN
2490 PRINT@INT(1+V1/3)*64+INT(U1/2),LEFT$(P1$,1);:RETURN
2500 FORX=0TO127:SET(X,Y)::NEXTX:RETURN
2510 PX$=P1$:GOSUB2490:FORX=1TO20:RESET(U1,V1):FORXX=1TO20:NEXTX
X:SET(U1,V1):FORXX=1TO50:NEXTXX,X:RETURN
2520 PX$=P2$:GOSUB2480:FORX=1TO20:RESET(U2,V2):FORXX=1TO20:NEXTX
X:SET(U2,V2):FORXX=1TO50:NEXTXX,X:RETURN
2530 IFP1=1THENPX$=P1$:FORX=1TO20:RESET(U1,V1):RESET(U1+1,V1):FO
RXX=1TO20:NEXTXX:SET(U1,V1):SET(U1+1,V1):FORXX=1TO50:NEXTXX,X
2540 IFP1=0THENPX$=P2$:FORX=1TO20:RESET(U2,V2):RESET(U2+1,V2):FO
RXX=1TO20:NEXTXX:SET(U2,V2):SET(U2+1,V2):FORXX=1TO50:NEXTXX,X
2550 RETURN
2560 PRINT@0,STRING$(64,32);:RETURN
2570 PRINT@896,STRING$(64,32);:RETURN
2580 FORX=X6TOX7:POKEX,191:NEXTX:RETURN
2590 CLS:PRINT"P R O  S H O P"
2600 PRINT:PRINT"WHAT DO YOU WANT TO ASK ABOUT?"
2610 PRINT:PRINT"1-- CLUBS."
2620 PRINT:PRINT"2-- DIRECTIONS."
2630 PRINT:PRINT"3-- HINTS."
2640 PRINT:PRINT"4--RETURN TO GAME."
2650 PRINT:PRINT:INPUT"SELECT CATEGORY (BY NUMBER)";CA
2660 IFCA>=5THEN2590
2670 IFCA=4THEN2700
2680 ONCAGOSUB260,370,440
2690 GOTO2590
2700 CLS:F=USR(1)
2710 GOTO1050
2720 IFP1=1ANDPP=1THENS1=S1-1:PP=0ELSEIFP1=0ANDPQ=1THENS2=S2-1:P
Q=0
2730 RETURN
2740 IFP1=1THENU=U1:V=V1ELSEU=U2:V=V2
2750 RETURN
2760 IFU>=120THENGOSUB2770:GOTO1050ELSE1550
2770 IFP1=1THENPK=0:PP=0:PC=0:PE=0:PI=0ELSEPB=0:PQ=0:PD=0:PF=0:P
Z=0
2780 RETURN
2790 CLS:RESUME620
2800 CLS:PRINT"HOW TO PLAY GOLF."
2810 PRINT:PRINT"A FAIRWAY WILL BE DRAWN ON THE SCREEN, AND EACH
 PLAYER WILL BE":PRINT"ASKED, IN TURN, TO PLAY A STROKE.":PRINT:
PRINT"RESPOND WITH:  CLUB (NUMBERS 1-8) , DIRECTION (0-360 DEGRE
ES) ,":PRINT"STRENGTH (NUMBERS 0 MIN.--9 MAX.)"
2820 PRINT:PRINT"EACH CLUB HITS A SET DISTANCE, BUT STRENGTH MAY
 BE SELECTED":PRINT"TO INCREASE THAT DISTANCE. (HOWEVER, THIS IN
CREASES CHANCE":PRINT"OF SLICING OR HOOKING A SHOT.)"
2830 PRINT:PRINT"THE FIRST LETTER OF EACH PLAYER'S NAME IS PRINT
ED UNDER HIS":PRINT"BALL, AND THE BALL FLASHES TO SHOW ITS LOCAT
ION AND WHO'S":PRINT"SHOT IT IS.":GOSUB360
2840 CLS:PRINT"WHEN BOTH PLAYERS ARE ON THE GREEN, THE SCREEN WI
LL THEN":PRINT"DISPLAY ONLY THE GREEN AND HOLE.":PRINT:PRINT"ON
THE GREEN A STROKE IS PLAYED BY SELECTING:":PRINT"DIRECTION (0-3
60 DEGREES) , STRENGTH (1-99)"
2850 PRINT"A STRENGTH OF 99 WILL ALMOST TRAVEL THE WIDTH OF THE
SCREEN.":PRINT:PRINT"CUT AND SLOPE ARE PRESENT ON THE GREEN, AND
 VALUES ARE":PRINT"SHOWN AFTER THE FIRST PLAYER'S SHOT. THIS ASS
UMES PLAYER 2"
2860 PRINT"WAS WATCHING, AND WOULD THEREBY KNOW THE VALUES.":PRI
NT:PRINT"VALUES MUST BE ENTERED USING COMMA'S FOR SEPARATION":PR
INT"OTHERWISE, THE DISPLAY WILL BE DESTROYED.":PRINT:PRINT"TO RE
COVER THE SCREEN, ENTER 9,0,0 AS YOUR SHOT."
2870 GOSUB360:RETURN
2880 SET(U1,V1):SET(U2,V2):GOSUB2480:GOSUB2490:RETURN
```

**** LI/4K   CLEANUP ****

TRS-80/SYSTEM-80

```
5 GOS.900
10 C.
20 F.I=1T.30:P.A.0:F.J=1T.10:N.J
21 P.A.0;"CLEANUP FOR LEVEL I - A VERY MESSY GAME - BY D.S.BRENT
ON"
22 F.J=1T.10:N.J:N.I
30 P.:P."  THIS IS A GAME OF SKILL FOR ONE PLAYER.  THE OBJECT"
40 P."IS TO COLLECT ALL OF THE @'S DISPLAYED ON THE SCREEN,"
50 P."WITHOUT HITTING ANY WHITE AREA.  WHEN YOU THINK YOU HAVE"
60 P."CLEAR THE SCREEN OF THE @ CHARACTERS, RUN INTO A WHITE"
70 P."AREA.  YOUR SCORE WILL THEN BE CALCULATED.  BE CAREFUL"
```

```
80 P."NOT TO PRESS THE CONTROL KEYS TOO OFTEN, AS THIS WILL"
90 P."WIPE THE BOARD, AND THE GAME WILL NOT FUNCTION PROPERLY."
100 P.:I."PRESS  <ENTER>  TO CONTINUE";A$
105 GOS.900
110 C.:P."YOUR CONTROL KEYS ARE :"
115 P.:P."   @     - TO MOVE UP"
120 P."<ENTER> - TO MOVE DOWN"
130 P."    ]     - TO MOVE LEFT"
140 P."    ^     - TO MOVE RIGHT"
150 P.A.512;:L=0:I."WHAT LEVEL OF PLAY DO YOU WANT (12-100)";L
151 IF(L<12)+(L>100)T.150
155 GOS.900
160 C.:F.I=0T.127:S.(I,6):S.(I,47):N.I
170 F.I=7T.46:S.(0,I):S.(127,I):N.I
180 F.I=1T.L*3:X=R.(126)+1:Y=R.(40)+6:S.(X,Y):N.I
200 C=0:F.I=1T.12:F.J=1T.L/12:C=C+1:A(C)=R.(30)*2+I*64+129
201 P.A.A(C);"@";:N.J:N.I
204 X=R.(126)+1:Y=R.(40)+6
205 P.A.0;"  PRESS <S> WHEN YOU SEE THE FLASHING DOT";
206 S.(2,0):P.A.0;:S.(X,Y)
207 IFP.(2,0)T.R.(X,Y):G.206
210 P.A.0:R=R.(3)-2:IFR=0T.S=R.(2)*2-3:G.220
215 S=0
220 P.A.1;:S.(0,0):S.(2,0):S.(4,0):S.(0,3)
225 IFP.(0,6)=0T.1000
230 IFP.(0,3)=0T.R=0:S=1:G.220
231 IFP.(0,0)=0T.R=-1:S=0:G.220
232 IF(P.(2,0)=0)*(P.(4,0)=1)*(P.(0,0)=1)T.R=1:S=0:G.220
233 IF(P.(4,0)=0)*(P.(0,3)=1)T.R=0:S=-1:G.220
234 X=X+R:Y=Y+S:IFP.(X,Y)T.P.A.0;"CALCULATING SCORE":G.500
235 S.(X,Y):G.225
500 Z=0:F.I=1T.C:X=(A(I)-I.(A(I)/64)*64)*2:Y=I.(A(I)/64)*3
501 F.J=XT.X+1:F.K=YT.Y+2
502 IFF=-1T.N.K:N.J:F=0:N.I:G.750
503 IFP.(J,K)=1T.Z=Z+1:F=-1:G.502
504 N.K:N.J:N.I
750 GOS.900:C.:P."END OF GAME ";:F.I=1T.30:P."!";:N.I:P.
755 P.:P."YOU SCORED";(Z/C)*100;"PERCENT."
760 P.:P.:P.
765 P."DO YOU WANT TO PLAY AGAIN (Y/N)";:Y=1:N=0
766 I.X:GOS.900:IFX=1T.110
770 C.:E.
900 C.:F.I=1T.75:P.A.R.(1000);"@";:N.I
901 F.I=1T.500:N.I:RET.
1000 C.:F.I=1T.200:P."YOU CHEATED !!! ";:F.J=1T.3:N.J:N.I
1001 G.760
```

**** LII/4K   E=MC2 ****

TRS-80/SYSTEM-80

```
10 REM.                    E = MC[2

B.J.C. 1978                                    (C) MICRO-80 1980
20 CLS: PRINT: PRINT"THIS PROGRAM COMPUTES THE RELATIVE MASS OF
A BODY IN MOTION
IN ACCORDANCE WITH  EINSTEIN'S RELATIVITY THEORY.  THIS STATES
THAT A BODY WILL HAVE INFINITE MASS AT THE VELOCITY OF LIGHT."
30 PRINT: INPUT"PRESS ENTER TO CONTINUE";X
40 DEFDBL M,P,V,C,R
50 CLS: PRINT: INPUT"ENTER MASS IN KGS. OF BODY AT REST";M
60 PRINT: INPUT"ENTER VELOCITY  -  MILES PER HOUR";P
70 V=1.61*P/3600
80 C=299792.8
90 R=M*(SQR(1/(1-((V*V)/(C*C)))))
100 CLS: PRINT: PRINT"
A BODY WITH A MASS AT REST OF";M;"KG. WILL HAVE
A RELATIVE MASS OF";R;" KG. AT";P;"M.P.H."
110 PRINT: INPUT"PRESS ENTER TO CONTINUE";X: GOTO40
120 REM. ENTER .001 KG AT 670344171.0625153 M.P.H. JUST TO
        SEE WHAT HAPPENS !
130 END
```

**** LII/16K   ANAGRAMS ****

TRS-80/SYSTEM-80

```
10 ' ********************* A N A G R A M S ********************
20 ' *                                                          *
30 ' *               BY  MICHAEL MERRYLEES                      *
40 ' *                           AGED 12-13                     *
50 ' *                                                          *
60 ' *                                                          *
70 ' *       I ENJOYED MAKING  " ANAGRAMS "  VERY MUCH          *
80 ' *     AND I HOPE YOU WILL ENJOY MY WORK IN THE SAME WAY    *
90 ' *       MY FAMILY AND I HAVE .                             *
100 '*          P.S.  THE JUMBLED UP WORDS IN DATA ARE NOT      *
110 '*       AS THE FINISHED ANAGRAMS COME OUT ...              *
120 '*          I HAVE JUST CODED THE WORDS SO YOU DON'T KNOW   *
130 '*       WHAT ALL THE WORDS ARE IN MY PROGRAM WHEN YOU      *
140 '*            TYPE  IT  IN .
150 '*                   THANKYOU.                              *
160 '*

                            *
170 '************************************************************
180 '**************** INSTRUCTIONS ******************************
190 CLS:CLEAR2000:N9=1
200 PRINTTAB(24)"A N A G R A M S"
210 PRINT:PRINT
220 PRINTTAB(15)"BY";CHR$(10);"                        MICHAEL MERRYL
EES"
230 PRINT:PRINT:PRINT"IF INSTRUCTIONS ARE NEEDED TYPE ";CHR$(34)
;"I";CHR$(34);" ELSE TYPE ";CHR$(34);"N";CHR$(34);
240 A$=INKEY$:IFA$=""THEN240
250 IFLEFT$(A$,1)="I"THEN270
```

```
260 IFLEFT$(A$,1)="N"THEN370ELSEGOTO240
270 PRINT@0,CHR$(30):PRINT@24,"A N A G R A M S":PRINT
280 PRINT"   ANAGRAMS ARE WORDS IN WHICH ALL THE LETTERS ARE JUM
BLED UP    AND YOU ARE GIVEN THE TASK OF UNJUMBLING THE LETTERS I
N A        LIMITED TIME. I HAVE TAKEN THIS OLD GAME AND WITH THE
HELP OF    DATA AND RANDOM NUMBER STATEMENTS I HAVE CONVERTED IT"
;
290 PRINT" INTO A    COMPUTER PROGRAM.";CHR$(10);"   THERE ARE F
OUR LEVELS IN THE PROGRAM :";CHR$(10);"        1  =   4-6 LETTER
 WORDS";CHR$(10);"          2  =   7-8 LETTER WORDS"
300 PRINT"        3  =   9-13 LETTER WORDS";CHR$(10);"   AND  4
 =   WHICH HAS WORDS RANGING FROM";CHR$(10);"
4-13 LETTERS IN THEM";
310 PRINT:PRINT:PRINT:PRINT"PRESS ANY KEY TO CONTINUE";
320 A$=INKEY$:IFA$=""THEN320
330 PRINT@640,CHR$(30):PRINT@704,CHR$(30):PRINT@768,CHR$(30):PRI
NT@960,CHR$(30):PRINT@24,"A N A G R A M S":PRINT:PRINT:PRINT"
IN EACH ONE OF THESE LEVELS THE COMPUTER WILL SHOW YOU ONE OF TH
E MANY";
340 PRINT" WORDS POSSIBLE AND YOU WILL HAVE TO WORK IT OUT AND
 TYPE IT IN DURING A LIMITED TIME. IF YOU CANNOT WORK THE ANAGRA
MOUT AND YOU WOULD LIKE EITHER TO STOP OR TRY ANOTHER ANAGRAM
 THEN TYPE IN '/'. ";CHR$(30)
350 PRINT"    THERE ARE ALSO TWO LEVELS OF DIFFICULTY IN THE PRO
GRAM";CHR$(10);"
          (P)  --   PROFESSIONAL";CHR$(10);"          (A)  --
AMATEUR"
360 PRINT"     N.B.     PLEASE DO NOT PRESS ENTER AFTER";CHR$(10
);"                YOU ENTER YOUR ANSWER !!"
370 PRINT:INPUT"ENTER THE LEVEL YOU WANT (1 - 4)";U:IFU>4ORU<1TH
EN370
380 PRINT@896,CHR$(30);:PRINT@896,;:INPUT"(P)ROFESSIONAL  OR  (A
)MATEUR ";TT$:IFTT$="P"THENTT=50ELSETT=150
390 '**************** DATA INPUT **********************************
400 CLS:PRINT:PRINT:PRINT:PRINT"PRESS ENTER FOR COMPUTER TO STAR
T DATA INPUT";
410 A$=INKEY$:IFA$=""THEN410
420 PRINT:PRINT"   ENTERING COUNTERS ..."
430 READV,W,X
440 Y=V+W+X
450 IFU=1THENZ=V:M=6:DIMD$(V)
460 IFU=2THENZ=W+V:M=8:DIMD$(W)
470 IFU=3THENZ=Y:M=13:DIMD$(Y)
480 IFU=4THENZ=Y:M=13:DIMD$(Y)
490 T=1
500 PRINT"     ENTERING WORDS :-"CHR$(10);"      THIS MAY TAKE A
 WHILE ACCORDING TO";CHR$(10);"      WHICH LEVEL YOU PICKED SO W
ATCH THE";CHR$(10);"    SNOW WHILE YOUR COMPUTER LISTENS TO";CHR
$(10);"         SOME   MUSIC   ....."
510 FORN2=1TO2000:NEXT
520 FORB=1TOZSTEP5:READD$(T),D$(T+1),D$(T+2),D$(T+3),D$(T+4)
530 IFU=1THENT=T+5
540 IFU=2ANDB>VTHENT=T+5
550 IFU=3ANDB>WTHENT=T+5
560 IFU=4THENT=T+5
570 FORWW=1TO5:SET(RND(127),RND(47)):NEXT
580 NEXTB:FORB=1TOT-1
590 IFLEN(D$(B))<8THENN3=INT(LEN(D$(B))/2):D$(B)=RIGHT$(D$(B),N3
)+LEFT$(D$(B),LEN(D$(B))-N3)ELSEN3=INT(LEN(D$(B))/3):D$(B)=RIGHT
$(D$(B),N3)+MID$(D$(B),LEN(D$(B))-2*N3+1,N3)+LEFT$(D$(B),LEN(D$(
B))-2*N3)
600 SET(RND(127),RND(47)):NEXTB
610 RANDOM
620 PRINT"   FINISHED !!!"
630 PRINT"I AM NOW WORKING OUT A FEW    A N A G R A M S   FOR YO
U
640 '************** THE MAKINGS OF AN ANAGRAM ******************
650 S=RND(T-1):IFU=2ANDS<VTHEN650
660 IFU=3ANDS<WTHEN650
670 A$=D$(S)
680 A=LEN(A$)
690 IFL=1THEN720
700 IFN5=1THEN720
710 DIMA$(M),A(M),B(M)
720 FORB=1TOA:A$(B)=MID$(A$,B,1):NEXT
730 N=1:NN=1
740 FORB=1TOA
750 A(B)=RND(A)
760 FORC=1TOB
770 IFA(B)=B(C)THENNN=NN+1:IFNN>20THENN5=1:GOTO650ELSEGOTO750
780 NEXTC
790 B(B)=A(B)
800 NEXTB
810 FORB=1TOA:B$=B$+A$(A(B)):NEXT
820 N=N+1:IFN>5THENRANDOM:GOTO650
830 IFB$=A$THEN720
840 '*************** MAIN SCREEN LAYOUT **************************
850 IFN9<>1THENPRINT@524,"----------------";:GOTO900
860 CLS:PRINTCHR$(23):FORN8=0TO113STEP4:SET(N8,0):SET(N8,46):NEX
T:FORN8=2TO44STEP2:SET(0,N8):SET(112,N8):NEXT:FORN8=36TO72STEP4:
SET(N8,1):SET(N8,6):SET(N8+1,1):SET(N8+1,6):NEXT:FORN8=2TO5:SET(
36,N8):SET(73,N8):NEXT:PRINT@84,"ANAGRAMS";
870 PRINT@198,".......................";:PRINT@336,"YOUR ANAGRAM"
;:PRINT@408,STRING$(4,92);:PRINT@516,STRING$(4,94);"------------
-----";STRING$(4,93);:PRINT@664,"[[[[";:PRINT@710,"SCORE:";:PRIN
T@776,"***";:PRINT@740,"TIME:";:PRINT@804,"**** MINS";
880 PRINT@908,"*****************";:PRINT@776,"000";
890 FORN8=1TO1000:NEXT
900 N7=INT(28-A+512):IFN7/2<>INT(N7/2)THENN7=N7-1
910 PRINT@N7,B$;:PRINT@802,USING"###.#";A*TT/500;:PRINT@214,"STA
RT";
920 '*************** LISTENING TO YOUR ANSWERS *****************
930 C=1:N7=N7+384
940 FORB=1TOTT*A:G=TT*A-B:GOSUB1110:C$=C$+INKEY$
950 IFRIGHT$(C$,1)=CHR$(8)ANDC$<>CHR$(8)THENC$=LEFT$(C$,LEN(C$)-
2):PRINT@N7,C$;"**";
960 IFC$=CHR$(8)THENC$=""
970 IFRIGHT$(C$,1)=CHR$(10)ORRIGHT$(C$,1)=CHR$(13)THENC$=LEFT$(C
$,LEN(C$)-1)
980 PRINT@N7,C$;
990 IFRIGHT$(C$,1)="/"THENPRINT@200,"THE SOLVED ANAGRAM:";:PRINT
@N7-384,A$;:FORN8=1TO200:NEXT:GOTO1050
```

```
1000 IFLEN(C$)+1>ATHEN1030
1010 NEXTB
1020 PRINT@198,"OUT OF TIME - IT WAS..";:PRINT@N7-384,A$;:GOTO10
50
1030 IFC$=A$THENPRINT@198,"CORRECT ! MY ANAGRAM -";:PRINT@N7-384
,A$;:N6=N6+1:N6$=STR$(N6):PRINT@780-LEN(N6$),N6$;:GOTO1050
1040 PRINT@198,"WRONG  BAD LUCK IT WAS";:PRINT@N7-384,A$;:GOTO10
50
1050 FORN8=1TO2000:NEXT:PRINT@198," 1 FOR MORE, 2 TO END ";
1060 Z$=INKEY$:IFZ$=""THEN1060
1070 IFZ$="1"THENPRINT@N7,STRING$(LEN(B$),"*");:A$="":B$="":C$="
":L=1:N9=2:PRINT@528,"I'M THINKING";:PRINT@198,"................
......";:PRINT@804,"****";:RANDOM:GOTO650
1080 CLEAR50
1090 PRINT@198,"THANKS FOR PLAYING BYE";:FORA=1TO1000:NEXT:FORA=
1TO15:PRINT:NEXT:END
1100 '*************** TIME  SUB-ROUTINE ************************
1110 PRINT@802,USING"###.#";G/500;:RETURN
1120 '********** DATA STATEMENTS WITH ENCODED ANAGRAMS *********
1130 DATA40,60,70
1140 DATAHEAC,REHI,GEUR,TESI,BTDE,ALOV,INVA,WNLA,OMWH,ITKN,IPTR
1150 DATACLECY,ULTFA,ROLEN,ORDSW,UGEGA,IZESE,OKYSM,DTHWI,CHTYA,E
UEQU,RIPST
1160 DATAIONMOT,EUMMUS,RCEFIE,MITCOM,IREDES,RORMIR,CLEMUS,OURLAB
,AGEGAR,IALSER,EALORD,NEDBUR,PITPUL,ISHPUN,PETPUP,IFYPUR,ESSSTR,
IPESTR
1170 DATAYCLEBIC,MNEYCHI,DEMNCOM,CENDDES,MACHSTO,SAGESAU,ACCOTOB
,ICLEVEH,NESSWIT,EIPTREC,ENSEIMM,UGHTDRO,REMEEXT,IENTANC,UIREINQ
,TURYCEN,IZENCIT,CERTCON,LINGCEI,TIFYCER,PARECOM,EOUSHID,OINEHER
,LTHYHEA,ATRETHE,RIERTER,RSTYTHI,THERWHE,STLEWHI,PPEDWRA,STLEWRE
1180 DATAINGSSLGO
1190 DATAAPSELLCO,ROUSMOHU,UNCENOAN,BBLEUASQ,NDAHRAVE,TIONXATA,L
LELRAPA,ICALYSPH,GIONLIRE,ERVEESPR,NITEFIDE,SENEROKE,CENTNOIN,ER
SEIVUN,OUGHORTH,ATORDIRA,NTICMARO,STERGIRE,ARCHSERE,LYSERAPA,YPU
SATPL,HLETMPPA,LIARCUPE,RAITRTPO,EIVERCPE,ACRESSMA,IFULRCME,CIAN
SIMU
1200 DATANCEULAAMB,GUEALOCAT,NZALUEINF,URENATSIG,MMEGRAPRO,ENTMA
NPER,IZEOGNREC,ENDOMMREC,SALEARREH,OUSIGIREL,NIAUMOPNE,EGEVILPRI
,URECEDPRO,LERPELPRO,REDFERPRE,BLEERAMIS
1210 DATASIONMISCOM,ABLEOURFAV,ENCESCICON,RATEUSTILL,TIONOLUREV,
NERYTIOSTA,ITEREWRTYP,IOUSTORVIC,TIONUPAOCC,ULARTICPAR,MENTLIAPA
R,URRAKABKOO,DUALIVIIND,ANCEORTIMP
1220 DATAAGANTRAVEXT,ATIONOCIASS,NDENTEPEIND,UNITYORTOPP,ATIONPA
RPRE,SIBLEPONRES,RHAGEMORHAE,OITREONNREC,CABLECTIPRA,ORATEROBCOR
,ATIONORMINF,PMENTELODEV,RFEITNTECOU,CTIONTINDIS,ANCESITTREM,CIO
USONSUNC
1230 DATANATECTIOAFFE,TIONERSACONV,HIEFKERCHAND,ALLYSIONOCCA,IBL
EEMPTCONT,ONERECTICONF,CITYNTRIECCE,ELEDRALLUNPA,TINGUCIAEXCR,AT
EDEMORCOMM,ATESUNICCOMM,IOUSNTATOSTE,TONECE-SPUMI
1240 DATAATIONMMODACCO,ATIONRMINDETE,ATELYRTUNUNFO,EMENTRTISADVE
,TIOUSCIENCONS,ATIONUNCIPRON,ANCESUMSTCIRC,ATIONSSINASSA,ATIONID
ERCONS,SMENTRRASEMBA,IENCENVENINCO
1250 RETURN
1260 '**********************************************************
1270 '*************** T H A N K Y O U **************************
```

**** LII/16K   HORSE PERFORMANCE GUIDE PART 1. ****

TRS-80/SYSTEM-80

```
10 CLEAR 8000:DIM A$(151),Y$(20):T$="/":A$(151)="END"
20 DEFINT N:POKE 16553,255
30 T=0:CLS: PRINT"RACE HORSE PERFORMANCE GUIDE  PART (1)   INQUI
RY FILE REQUEST ":PRINT STRING$(64,140):PRINT"(1) SINGLE COMPARI
SON":PRINT "(2) INQUIRY FILE"
40 PRINT:  PRINT STRING$(64,183): PRINT :PRINT "PROGRAMMER GEOFF
 EGEL 18 STURT ST LOXTON 5333 PH 847972": PRINT: PRINT STRING$(6
4,183)
50 V$=INKEY$: IF V$="" THEN 50ELSE IF V$="1" THEN 910ELSE IF V$=
"2" THEN  CLS:INPUT"SET#-1 RECORDER TO REC MODE  WITH BLANK TAPE
 FOR INQUIRY          FILE PRESS ENTER TO CONTINUE";V$ ELSE GOSU
B 990: GOTO 30
60 CLS:PRINT"INQUIRY NAMES":PRINT STRING$(63,140):INPUT"ENTER NU
MBER OF RACES THIS MEETING";N :  IF (N=0)OR(N>9) THEN 60
70 T=T+1:PRINTT;:INPUT"ENTER NUMBER OF RUNNERS IN THIS RACE";A:Z
=Z+A:IF N=T THEN 80ELSE 70
80  I=0:R=0:FOR A=1 TO Z
90 CLS:PRINT"*** OPTIONS ***":PRINT STRING$(63,140):PRINT"(1) RA
CE NUMBER":PRINT"(2) DISTANCE":PRINT"(3) HORSE NAME ":PRINT"(4)
WEIGHT ":PRINT:PRINT"LAST HORSE ENTERED WAS ";B$
100 V$=INKEY$:IF V$=""THEN 100ELSE N=VAL(V$):IF N=0 THEN 100ELSE
 IF N>4 THEN 100ELSE CLS:ON N GOTO110,120,130,140
110 R=0:P$="0":INPUT"RACE NUMBER";P$:IF P$="0"THEN 110ELSE IF X=
1 THEN 160
120 D$="0":INPUT"DISTANCE";D$:IF D$="0" THEN 120ELSE IF X=1 THEN
 160
130 RN$="0":INPUT"HORSE NAME";RN$:IF RN$="0" THEN 130ELSE IF X=1
 THEN 160
140 W=0:INPUT"WEIGHT";W:IF W=0 THEN 140ELSE F=30:INPUT"LESS WEIG
HT ALLOWANCE";F:IF F>9  THEN 140ELSE W=W-F:IF X=1 THEN 160
150 R=R+1
160 B$=RN$+T$+P$+T$+STR$(R)+T$+STR$(W)+T$+D$+T$:CLS:PRINT B$:PRI
NT STRING$(63,140):PRINT "HAS AN ERROR BEEN MADE (1) YES (2) NO"
170 V$=INKEY$:IF V$="" THEN 170ELSE N=VAL(V$) : IF N=1 THEN X=1:
GOTO 90
180 X=0:CLS:A$(1)=B$: GOSUB 1060
190 B=1:FOR N=1TO 149:B=B+1
200 IF A$(N)>A$(B) THEN K$=A$(N):A$(N)=A$(B):A$(B)=K$
210 NEXT N:  K$="": NEXT A:GOSUB 1080:CLS: PRINT "INQUIRY FILE C
OMPLETED"
220  PRINT " MASTERFILE SEARCH COMMENCES":P=0:PRINT STRING$(64,1
40):B=1: INPUT" SET MASTERFILE TAPE TO PLAY  WITH RECORDER #-1 P
RESS ENTER TO CONTINUE";V$
230 CLS:PRINT"RUNNING":PRINT STRING$(64,140)
240 A$(151)="END": A=0: W=5:B=1
250 GOSUB 380
260 GOSUB 330
270 IF A$(A)="END" THEN 820
280 IF T$=M$ THEN 320
290 IF T$>M$ THEN GOSUB 380:IF C$(1)="END" THEN 810 ELSE 280
300 IF T$<M$ THEN GOSUB 1000:PRINT "SEARCH FOR THIS HORSE COMPLE
TED":PRINT "(1) REPEAT DISPLAY":PRINT"(2) CONTINUE"
```

```
310 V$=INKEY$:IF V$="" THEN 310 ELSE IF V$="1" THEN CLS:GOTO 300
:ELSE IF V$="2" THEN A=A+1:GOSUB 1050:GOSUB 330:GOTO 270:ELSE GO
SUB 990: GOTO 300
320  GOSUB 440: GOSUB 380: GOTO 280
330 IF A$(A)="" THEN A=A+1:GOTO 330
340 IF A$(A)="END" THEN 360
350 GOSUB 700:T$=D$:GOSUB 730:GOSUB 730:GOSUB 730:TW=VAL(D$)
360 RETURN
370 W=0:INPUT#-1,N$(1),N$(2),N$(3),N$(4),N$(5):POKE 16553,255
380 IF W=5 THEN 370ELSE W=W+1:C$(1)=N$(W):PRINT C$(1):IF C$(1)="
END" THEN 420ELSE GOSUB 710:M$=D$:GOSUB 730:IF D$="0" THEN DC=0
ELSE DC=VAL(D$)
390 GOSUB 730:LW=VAL(D$):GOSUB 730:H=VAL(D$):GOSUB 730:TC$=D$:IF
(D$="FAST")OR(D$="SLOPPY") THEN Z=0
400 IF D$="GOOD"THEN Z=4 ELSE IF D$="SLOW" THEN Z=8 ELSE IF D$="
MUDDY" THEN Z=12 ELSE IF D$="HEAVY" THEN Z=16 ELSE Z=0
410  GOSUB 730:M=VAL(D$):GOSUB 730:S=VAL(D$)
420 RETURN
430 REM    #### RACE HORSE SPEED   BREAKDOWN        ####
440 P=0:AA=100
450 S=M*60+S:L=S*100: Q=L
460 READ TD,C: IF (TD=0)OR(C=0)THEN RESTORE: GOTO 680
470 IF(TD<>H) THEN  460
480 RESTORE:IF(C=L) GOTO 510
490 IF(C<L) THEN L=L-C:L=L/20:AA=AA-L:GOTO 510
500 IF (C>L)THEN L=C-L:L=L/20:AA=AA+L
510 F=AA
520 R=1 : HH=0:U=1
530 IF(TW=LW) THEN 620
540 IF(D>1600) THEN 560
550 R=R/2:U=U/2
560 IF(TW>60)AND(LW>60)THEN R=R*2:GOTO 590
570 IF (TW>60)AND(LW<60)THEN U=TW-60
580 IF(LW>60)AND(TW<60)THEN U=LW-60
590 IF TW>LW THEN P=TW-LW ELSE P=LW-TW
600 R=R*P:U=HH*U:R=R+U
610 IF(TW>LW) THEN AA=AA-R ELSE AA=AA+R
620 AA=AA+ZZ: RESTORE: IF Q<C THEN PRINT " NEW RECORD "TD,Q:INPU
T" PRESS ENTER TO CONTINUE ";V$
630 BD$="DIS ":BW$= " WIE ": BC$=" ACT RAT  ": BS$=" TOD$ RAT "
640 MM$= " ":Y$(20)=STR$(DC)+MM$+BD$+STR$(H)+BW$+STR$(LW)+MM$+TC
$+BC$+BR$+STR$(F)+BS$+STR$(AA)
650 FOR Y=1 TO 19
660 IF Y$(Y)="" THEN Y$(Y)=Y$(20): Y$(20)=""
670 NEXT Y: RETURN
680 PRINT " THIS DISTANCE OF ";H;" METERS AND TIME ";Q;"  NOT RE
CORDED":INPUT"PRESS ENTER TO CONTINUE";V$: RETURN
690 REM ##### STRING BREAKDOWN ROUTINE    ######
700 TT$=A$(A):GOTO 720
710 TT$=C$(B)
720 P=0: V=0
730 V=P+1: T=0
740 P=P+1: T=T+1
750 D$=MID$(TT$,P,1) :IF ASC(D$)=47 THEN K=T-1: GOTO 770
760 GOTO 740
770 D$=MID$(TT$,V,K): RETURN
```

```
780 DATA 800,4610,900,5210,1000,5608,1010,5840,1100,6280,1110,63
60,1200,6780,1250,7310,1280,7510,1290,7580,1300,7600,1380,8390,1
400,8118,1450,8597,1500,8704,1550,9290,1600,9380,1750,10580,1800
,10718,1850,11180,1900,11550,2000,11980
790 DATA 2020,12280,2050,42430,2100,12860,2200,13350,2300,14600,
2380,14800,2400,14621,2450,15111,2500,15470,2600,16100,2800,1739
0,3200,14580,3400,21880
800 DATA 1050 ,6150,1460,8650,1650,10550,1700,10600,2250,12244,1
310,7720,2830,19040,3840,36887,3250,2190,3350,24530,1350,7980,28
50,18821,2900,19700,0,0
810 CLS:PRINT "ALL MASTERFILE NOW HAS BEEN ASSESSED": GOTO 870
820 CLS:PRINT" INQUIRY FILE COMPLETED ":PRINT STRING$(64,140):PR
INT"(1) REPEAT  SEARCH":PRINT"(2) RETURN TO OPTIONS"
830 V$=INKEY$:IF V$="" THEN 830:IF V$="1" THEN 270 ELSE IF V$="2
" THEN 30 ELSE GOSUB 990: GOTO 820
840 GOSUB 1000
850 PRINT"SEARCH FOR  THIS HORSE COMPLETED PRESS (1) TO CONTINUE
 (2) REPEAT DISPLAY"
860 V$=INKEY$:IF V$="" THEN 860ELSE IF V$="1" GOTO 870ELSE IF V$
="2" THEN PRINT A$(A):PRINT STRING$(64,140): GOTO 840ELSE GOSUB
990: GOTO 850
870 CLS:PRINT" INQUIRY FILE HAS COME TO THE END ":PRINT STRING$(
64,140)
880 PRINT "(1) RETURN TO OPTIONS":PRINT"(2) END":PRINT"(3) REPEA
T RUN"
890 V$=INKEY$:IF V$="" THEN 890ELSE IF V$="3" THEN 220 ELSE  IF
V$="1" THEN 30ELSE IF V$="2" THEN  PRINT " BYE FROM HORSE PERFOR
MANCE GUIDE HOPE YOU PICKED A GOOD PRICED  WINNER": END ELSE  GO
SUB990: GOTO 880
900 GOSUB 1050
910 DC=0:CLS:PRINT "INDIVIDUAL HORSE COMPARISON":PRINT STRING$(6
4,140):PRINT "I WILL NEED THE FOLLOWING DATA":H=0:INPUT"DISTANCE
 OF PARTICULAR RACE ";H:IF H=0 THEN 910
920 LW=0:TW=0:INPUT"WEIGHT THEN (,)WEIGHT NOW";LW,TW:IF (TW=0)OR
(LW=0) THEN 920
930 TC$="NULL":Z=50:INPUT"TRACK CONDITION  (IN A NUMBER) (0)(FAS
T OR SLOPPY)  (4)(GOOD)   (8) (SLOW) (12)(MUDDY) (16)(HEAVY)";Z:
 IF Z=50 THEN 930
940 M=0:S=0:INPUT"TIME IN MINUTES(,)SECONDS";M,S:IF(M=0)AND(S=0)
 THEN 940
950 GOSUB 430
960 CLS:GOSUB 1000:PRINT STRING$(64,140)
970 PRINT " (1) NEW COMPARISION(2) RETURN TO OPTIONS": PRINT"(3)
 REPEAT DISPLAY (4) ADDITIONAL DATA"
980 V$=INKEY$:IF V$="" THEN 980ELSE IF V$="1" THEN 900ELSE IF V$
="2" THEN 30ELSE IF V$="3" THEN 960ELSE IF V$="4" THEN 910ELSE G
OSUB 990: GOTO970
990 CLS:PRINT@384,STRING$(64,140),"INCORRECT RESPONSE":PRINT STR
ING$(64,140):FOR K=1 TO 1500:NEXT K:RETURN
1000 Y=20: CLS :PRINT A$(A):PRINT STRING$(64,140)
1010 IF Y=0 THEN 1040ELSE Y=Y-1
1020 IF Y$(Y)="" THEN 1010ELSE PRINT Y$(Y)
1030 GOTO 1010
1040 RETURN
1050 FOR Y=1 TO 20: Y$(Y)="": NEXT Y : RETURN
1060 I=I+1:F$(I)=B$
```

```
1070 IF I=5 THEN GOTO 1140  ELSE RETURN
1080 IF I=0 THEN 1090 ELSE ON I GOTO 1100,1110,1120,1130
1090 F$(1)="END"
1100 F$(2)="END"
1110 F$(3)="END"
1120 F$(4)="END"
1130 F$(5)="END"
1140  CLS:PRINT"SAVING ON TAPE":PRINT F$(1):PRINT F$(2):PRINT F$
(3):PRINT F$(4):PRINT F$(5):PRINT#-1,F$(1),F$(2),F$(3),F$(4),F$(
5):FOR I=1 TO 5:F$(I)="": NEXT I:I=0:RETURN
```

**** LII/16K   HORSE PERFORMANCE GUIDE PART 2. ****

TRS-80/SYSTEM-80

```
10 CLEAR 9000:DIMA$(200): T$="/":A$(200)="END"
20 DEFINT N,H
30 PRINT" HORSE PERFORMANCE GUIDE";: I=0
40 CLS:PRINT " PART (2)  MASTERFILE UPDATE":PRINT STRING$(64,140
):PRINT"(1) INQUIRY FILE UPDATE":PRINT"(2) ADDITIONAL INFORMATIO
N INSERTION":PRINT "(3) INSTRUCTIONS"
50 PRINT STRING$(64,140):PRINT"PROGRAMMER GEOFF EGEL 18 STURT ST
 LOXTON";:PRINT " 5333 PH 847972":PRINT :PRINT STRING$(64,191)
60 V$=INKEY$:IF V$="" THEN 60ELSE IF V$="1" THEN 70ELSE IF V$="2
" THEN 220ELSE IF V$="3" THEN 650ELSE GOSUB 610: CLS: GOTO 40
70 PRINT"INQUIRY FILE UPDATE":PRINT STRING$(64,140)
80 CLS:INPUT"SET#-1 RECORDER TO PLAY MODE WITH INQUIRY FILE PRES
S ENTER TO   CONTINUE";V$
90 F=0:INPUT#-1,F$(1),F$(2),F$(3),F$(4),F$(5)
100 IF F=5 THEN 90ELSE F=F+1:IF F$(F)="END" THEN F=0:GOTO 220
110 C$(1)=F$(F)
120 CLS:PRINT C$(1):PRINT STRING$(64,140):PRINT" *** OPTIONS ***
":PRINT"(1) TRACK CONDITION ":C=0:PRINT"(2) TIME":PRINT"(3) SCRA
TCHED"
130  V$=INKEY$:IF V$="" THEN 130ELSE N=VAL(V$):IF(N=0)OR(N>3) TH
EN GOSUB 610: GOTO 120ELSE ON N GOTO 140,150,210
140 TC$="":INPUT"TRACK CONDITION(FAST)(SLOPPY)(MUDDY)(SLOW)(HEAV
Y)(GOOD)";TC$:IF TC$="" THEN 140
150 M$="":INPUT"TIME IN MINUTES(/)SECONDS";M$:IF M$="" THEN 150
160 GOSUB 420:G$=D$:GOSUB 440:GOSUB 440:GOSUB 440:W$=D$: GOSUB 4
40
170 B$="": B$=G$+T$+STR$(C)+T$+W$+T$+D$+T$+TC$+T$+M$+T$:A$(1)=B$
180 B=1:FOR H=1 TO 198:B=B+1
190 IF A$(H)>A$(B) THEN K$=A$(H):A$(H)=A$(B):A$(B)=K$
200 NEXT H
210 GOTO 100
220 CLS:PRINT"ADDITIONAL INFORMATION":PRINT STRING$(64,140):PRIN
T"(1) DATE":PRINT"(2) DISTANCE":PRINT"(3) NAME":PRINT"(4) WIEGHT
":PRINT"(5) TIME":PRINT "(6) SAVE ON TAPE":PRINT "(7) INQUIRY FI
LE UPDATE CHECK"
230 PRINT STRING$(64,140):PRINT"LAST HORSE ENTERED  ";B$
240 V$=INKEY$:IF V$="" THEN 240ELSE N=VAL(V$):IF(N=0)OR(N>7) THE
N GOSUB 610:GOTO 220ELSE CLS:ON N GOTO 250,260,270,280,290,350,6
20
250 D=16: INPUT"NUMBER OF WEEKS SINCE RUN IF IT INCLUDES A HALF
WEEK ADD .5 TO TOTAL MAX 15 WEEKS ";D :IF D>=16  THEN 250ELSE IF
 X=1 THEN 300
260 E$="" : INPUT " DISTANCE /RACE CONDITION (FAST)(SLOPPY)(GOOD
)(SLOW)(MUDDY)(HEAVY) SEP BY(/)";E$:IF E$="" THEN 260ELSE IF X=1
 THEN 300
270 W$="":INPUT" NAME OF HORSE ";W$:IF W$="" THEN 270ELSE IF X=1
 THEN 300
280 K$="" :INPUT"WIEGHT CARRIED ";K$:IF K$="" THEN 280ELSE IF X=
1 THEN 300
290 M$="" :INPUT" TIME TAKEN IN MINUTES AND SECONDS SEP BY(/)";M
$:IF M$="" THEN 290
300 B$=W$+T$+STR$(D)+T$+K$+T$+E$+T$+M$+T$:CLS:PRINT B$:PRINT "HA
S AN ERROR BEEN MADE Y/N"
310 V$=INKEY$:IF V$="Y" THEN  X=1: GOTO 220ELSE IF V$="N" THEN X
=0 ELSE 310
320 PRINT" *** SORTING  ***":B=1: A$(1)=B$: FOR N=1 TO 198: B=B+
1
330 IF A$(N)>A$(B) THEN K$=A$(N): A$(N)=A$(B): A$(B)=K$
340 NEXT N: GOTO 220
350 CLS:PRINT"UPDATED INFORMATION  SAVE":PRINT STRING$(63,140):I
NPUT"SET TAPE DECK TO RECORD PRESS ENTER TO CONTINUE";V$
360 FOR N=1 TO 199
370 IF A$(N)="" THEN 390
380 N$=A$(N):GOSUB 490
390 NEXT N: IF A$(199)="" THEN CLS: PRINT"NO DATA IN ARRAY":FOR
QQ=1 TO 1500:NEXT QQ:CLS:GOTO 220ELSE GOTO 520
400 REM ##### STRING BREAKDOWN ROUTINE    ######
410 TT$=A$(A):GOTO 430
420 TT$=C$(1)
430 P=0
440 V=P+1: T=0
450 P=P+1: T=T+1
460 D$=MID$(TT$,P,1) :IF ASC(D$)=47 THEN K=T-1: GOTO 480
470 GOTO 450
480 D$=MID$(TT$,V,K): RETURN
490 IF F>5 THEN F=0 : GOTO 490ELSE F=F+1:F$(F)=N$
500 IF F=5 THEN PRINT#-1,F$(1),F$(2),F$(3),F$(4),F$(5):F=0:FOR F
=1 TO 5:PRINT F$(F):NEXT F: F=0
510 RETURN
520 IF F=0 THEN 540
530 ON F GOTO 550,560,570,580
540 F$(1)="END"
550 F$(2)="END"
560 F$(3)="END"
570 F$(4)="END"
580 F$(5)="END"
590 F=0:PRINT #-1,F$(1),F$(2),F$(3),F$(4),F$(5)
600 CLS:PRINT"LOADING UPDATED INQUIRY FILE COMPLETED":PRINT" PLE
ASE  CLOAD PART TWO OF MASTER FILE UPDATE": FOR QQ=1 TO 5000:NEX
T QQ:GOTO 220
```

```
610 CLS:PRINT@384,STRING$(64,140),"INCORRECT RESPONSE":PRINT STR
ING$(64,140):FOR M=1 TO 1500: NEXT M: RETURN
620 FOR N=1 TO 199:IF A$(N)="" THEN 640ELSE PRINT A$(N),N
630 FOR O=1 TO 250:V$=INKEY$:IF V$="" THEN NEXT O:GOTO 640ELSE I
NPUT"ARRAY NUMBER";V:CLS:PRINT A$(V):INPUT"CORRECTION";A$(V):O=2
50
640 NEXT N: GOTO 220
650 CLS:PRINT" MASTERFILE UPDATE INSTRUCTIONS":PRINT STRING$(63,
140)
660 PRINT" MASTERFILE UPDATE CONSISTS OF TWO SEPERATE PROGRAMS T
HIS ONE   WILL ALLOW YOU TO UPDATE THE INQUIRY FILE WHICH CONTAI
NS THE     RUNNERS DETAILS   WITH  THE RESULTS AS PUBLISHED IN TH
E          ADELIADE SUNDAY MAIL
670 PRINT "YOU SHOULD FIRST USE THE INQUIRY FILE UPDATE IN THIS
PROGRAM AND THEN USE ADDITIONAL INFORMATION TO INPUT RESULTS FRO
M MID WEEK RACES.
680 PRINT"WHEN YOU HAVE FINISHED ENTERING ALL DATA MAKE A COPY B
Y USING   THE TAPE SAVE ROUTINE WHEN THIS IS COMPLETED CLOAD THE
 FOLLOWING  PROGRAM OF THE MASTERFILE UPDATE PROGRAM TAPE ((M)."
690 PRINT"PLEASE NOTE PRESS RESET AND SET MEMORY SIZE TO 32512 T
O USE THE SECOND PROGRAM"
700 PRINT STRING$(63,140):INPUT"PRESS ENTER TO CONTINUE";V$:GOTO
 30
```

**** LII/16K   HORSE PERFORMANCE GUIDE PART 3. ****

TRS-80/SYSTEM-80

```
10 CLS:CLEAR 11000:DIM A$(250),X(2):PRINT"MASTERFILE MERGE AND U
PDATE":DEFINT A-Z:TT=5
20 INPUT"ERROR CHECKING ROUTINE Y/N";Z$:IF Z$="" THEN 20
30 W=16: INPUT"NUMBER OF WEEKS TO HOLD DATA MAX (15) ";W:IF W>15
 THEN 30ELSE W=W+1
40 N=0:POKE 16526,0:POKE 16527,127:FOR I=1 TO 203:READ A:N=N+A:P
OKE I+32511,A:NEXT
50 INPUT"SET THIS WEEKS UPDATED INQUIRY FILE TO PLAY PRESS ENTER
 TO CONTINUE";B$: PRINT "RUNNING"
60 FOR N=1 TO 250 STEP 5:INPUT#-1,A$(N),A$(N+1),A$(N+2),A$(N+3),
A$(N+4):IF A$(N+4)="END" THEN N=300 :GOTO 70 ELSE NEXT N
70 FOR N=1 TO 250:IF A$(N)="" OR A$(N)="END" THEN A$(N)="":GOTO
90
80 K=K+1
90 NEXT N
100 N=K:N=N/5:N=INT(N):N=N*5:IF N<>K THEN K=K+1:GOTO 100
110 IF I$(TT)="END" THEN C=0: GOTO 380
120 INPUT"SET OLD MASTERFILE TO PLAY PRESS ENTER TO CONTINUE";B$
130 PRINT" RUNNING":S=K+1:FOR N=S TO 250
140 IF TT=5 THEN INPUT#-1,I$(1),I$(2),I$(3),I$(4),I$(5): TT=1  E
LSE  TT=TT+1
150 IF I$(TT)="END" THEN A$(N)="ZZZZ": N=300:GOTO 300
160 P=0
170 P=P+1
180 K$=MID$(I$(TT),P,1)
190 IF K$="/" THEN 210
200 GOTO 170
210 K$=MID$(I$(TT),1,P)
220 R=P+1:B=0
230 P=P+1
240 B=B+1
250 D$=MID$(I$(TT),P,1)
260 IF D$="/" THEN B=B-1:D$=MID$(I$(TT),R,B):DC=VAL(D$):DC=DC+1:
IF DC>=W THEN I$(TT)="":GOTO 140ELSE 280
270 GOTO 230
280 P$=MID$(I$(TT),P):T$=STR$(DC):A$(N)=K$+T$+P$
290 NEXT N
300 IF Z$<>"Y" THEN 360
310 P=0:FOR M=1 TO 250 STEP 10:CLS
320 FOR SS=1 TO 10:P=P+1
330 PRINT P;A$(P):NEXT SS:PRINT"(A)LTER  (C)ONTINUE"
340  V$=INKEY$:IF V$="" THEN 340ELSE  IF V$="A" THEN INPUT"LINE
NUMBER";G:CLS:PRINT A$(G):INPUT"CORRECT DATA";A$(G):P=P-10:GOTO
320
350 NEXT M
360 Z=0:X(0)=250:X(1)=VARPTR (A$(1)):Z=USR(VARPTR(X(0)))
370 CLS:C=0:INPUT"PLEASE SET TO RECORD WITH NEW MASTERFILE PRESS
 ENTER TO CONTINUE";B$: PRINT "RUNNING"
380 C=C+1:IF C=250 THEN 530 ELSE IF A$(C)="" THEN 380 ELSE C=C-1
390 J=250-K: FOR S=C TO J STEP 5
400 F$(1)=A$(S):A$(S)="":IF F$(1)="ZZZZ" THEN 460
410 F$(2)=A$(S+1):A$(S+1)="" :IF F$(2)="ZZZZ" THEN 470
420 F$(3)=A$(S+2):A$(S+2)="":IF F$(3)="ZZZZ" THEN 480
430 F$(4)=A$(S+3):A$(S+3)="":IF F$(4)="ZZZZ" THEN 490
440 F$(5)=A$(S+4):A$(S+4)="":IF F$(5)="ZZZZ" THEN 500
450 GOTO 510
460 F$(1)="END"
470 F$(2)="END"
480 F$(3)="END"
490 F$(4)="END"
500 F$(5)="END"
510 PRINT F$(5):PRINT#-1,F$(1),F$(2),F$(3),F$(4),F$(5):IF F$(5)=
"END" THEN PRINT"MASTERFILE COMPLETED":END ELSE NEXT S
520 T=0:P=250-K: FOR NN=1 TO K: T=T+1:P=P+1: A$(T)=A$(P):A$(P)="
":NEXT NN:GOTO 110
530 PRINT"NO DATA LEFT IN ARRAY":END
540 DATA 205,127,10,94,35,86,237,83,19,127,35,94,35,86,237,83,21
3,127,33,0,0,34,211,127,237,91,211,127,203,59,127,203,58,48,2,20
3,251,237,83,211,127,122,179,200,42,19,127,237,82,34,207,127,33,
0,0,34,205,127,42,205,127,34,203,127,42,203,127
550 DATA  237,91,211,127,25,34,209,127,235,33,0,0,25,25,25,229,2
37,91,203,127,33,0,0,25,25,25,237,75,213,127,9,235,225,9,229,213
,14,0,126,71,26,184,48,3,14,1,71,175,176,40,25,197,19,35,78,35,7
0,197,225,235,78,35,70,197,225,193,26,150,56,10,32,39
560 DATA 19,35,16,246,203,65,32,31,209,225,6,3,78,235,126,113,23
5,119,35,19,16,246,42,211,127,235,42,203,127,175,237,82,34,203,1
27,48,144,24,2,209,225,42,205,127,17,1,0,175,25,34,205,127,237,9
1,207,127,237,82,218,58,127,195,24,127
570 INPUT#-1,A$(1),A$(2),A$(3),A$(4),A$(5):PRINT A$(5): GOTO 570
```

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

Next month's issue will contain at least the following programs plus the usual features and articles. An (80) after a program title indicates that the program will be for TRS-80 Model 1/3 or System 80/Video Genie computers. (Colour) indicates that the program will be for the TRS-80 Colour Computer and the Hitachi Peach.

** STARSHOOT LI/4K (80) **

This program is an LI/4K version of the game published in Micro-80 July 1981. The object of the game is to get a pattern of stars from the initial position, by shooting stars, to the end position. When a star is shot the pattern will change, depending on which star is shot.

** URANIUM CORE LII/16K (80) **

Uranium Core is set sometime in the future. Earth is rapidly running out of natural resources. Your mission is to retrieve the uranium cores found in the second universe and return them to our universe via the universe interface.

** DEFUSR FUNCTION LII/16K (80) **

This program will enable Level 2 users to use the Disk Basic command DEFUSR in their programs, which is much simpler to use than the pokes that normally have to be used. Instead of entering - POKE 16526,0 : POKE 16527,125 now you can just enter - DEFUSR=32000. Best of all, when you upgrade your system to Disk Basic you will find that the programs you have written for use with this DEFUSR function will be compatible with the format of the Disk Basic Defusr function.

** SINGLE KEY MENU Model III (80) **

This is a program just for Model III users who get callouses on their fingertips from typing in all those DOS commands so necessary to get anything out of their machines. When correctly set up, this machine language program displays a list of up to 16 of your most commonly used programs and/or commands which can be called up with a press of your finger.

** ARISTOCRAT (Colour) **

The Aristocrat is a pretty flashy pinball machine, with coloured wheels and sound. Hear the wheels spin and listen to the coins dropping into the tray when you win. The best thing of all, though, is that you don't have to put in any money to play.

** STAR TREK (Colour) **

You are in command of the Enterprise; you must destroy the Klingons before they destroy the Earth. Your ship is equipped with short and long range scanners, phasers, photon torpedoes and shields. When your ship receives damages, you can locate and dock with a Starbase for repairs and re-fuelling.

APPLICATION FOR PUBLICATION OF A PROGRAM IN MICRO-80

Date ....

To MICRO-80
SOFTWARE DEPT.
P.O. BOX 145
MORPHETT VALE, SA., 5162

Please consider the enclosed program for publication in MICRO-80.

Name ....

Address ....

....

.... Postcode ....

* * * 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.

***** CASSETTE/DISK EDITION INDEX *****

The cassette edition of MICRO-80 contains all the software listed each month, on cassette. The cassette also contains the source code for machine language programs which may not have been printed due to space restrictions. All programs are recorded twice. Level I programs can only be loaded into a Level I TRS-80 if the Level I in Level 2 program from the MICRO-80 Software Library - Vol. 1 is first loaded into your Level 2 TRS-80 or System 80/Video Genie. Note: System 80/Video Genie computers have had different tape-counters fitted at different times. The approximate start positions shown are correct for the very early System 80 without the volume control or level meter. They are probably incorrect for later machines. The rates for a cassette subscription are printed on the inside front cover of each issue of the magazine.

The disk edition contains all those programs which can be executed from disk, including Level I programs. Level I disk programs are saved into the NEWDOS format. Users require the Level I/CMD utility supplied with NEWDOS+ or NEWDOS 80 version 1.0 to run them.

				APPROX. START POSITION		
SIDE 1	TYPE	I.D.	DISK FILESPEC	CTR-41	CTR-80	SYSTEM 80
E=MC2	LII/4K	E	EEQMC2/BAS	18	10	5
"	"	"	"	33	18	8
ANAGRAMS	LII/16K	A	ANAGRAMS/BAS	49	27	12
"	"	"	"	131	73	37
GOLF	LII/16K	G	GOLF/BAS	205	114	66
"	"	"	"	310	173	104
SIDE 2						
H.R.G. PART 1	LII/16K	B	PART1/BAS	18	10	5
"	"	"	"	86	48	23
H.R.G. PART 2	LII/16K	C	PART2/BAS	148	83	41
"	"	"	"	189	106	61
H.R.G. PART 3	LII/16K	D	PART3/BAS	227	127	73
"	"	"	"	254	142	82
CLEANUP	LI/4K	-	CLEANUP/LV1	281	157	90
"	"	-	"	314	176	106

TO:
MICRO-80, P.O. BOX 213, GOODWOOD,
SOUTH AUSTRALIA. 5034.

Please RUSH to me the items shown below:

$ .... enclosed Date ....

.... 12 month subscription to MICRO-80

.... 12 mth. subs. to MICRO-80, plus the cassette edition

.... 12 mth. subs. to MICRO-80, plus the disc edition

.... The latest issue of MICRO-80 (see inside front cover for prices)

The MICRO-80 PRODUCTS listed below:

DESCRIPTION	PRICE

TOTAL ENCLOSED WITH ORDER

□ Cheque □ Bankcard □ Money Order

Bankcard Account Number

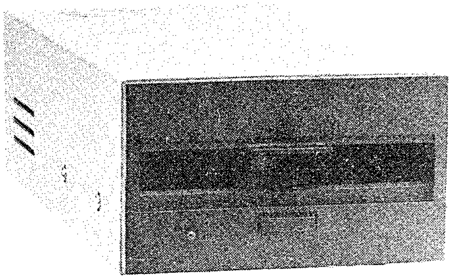
Signature .... Exp. End ....

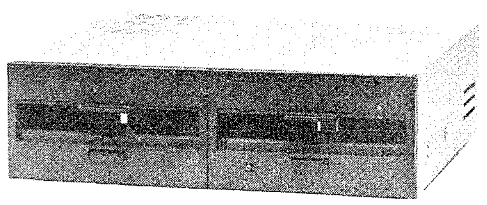
NAME ....

ADDRESS ....

.... Postcode ....

SAVE A PACKET ON MICRO-80's DISK DRIVE PACKAGES FOR TRS-80 MODEL 1 AND SYSTEM 80 MICROCOMPUTERS





SINGLE DRIVE PACKAGE *from* ... $499

DUAL DRIVE PACKAGE *from* ... $874

Bigger volume means lower cost price, which we are passing on to you. Avoid the annoying bundle of cables, wires and separate boxes. MICRO-80 is now offering our well-proven MPI disk drives in attractive, self-contained single or dual-drive cabinets complete with internal power supply. Our drive 0 and dual-drive packages also include the appropriate version of DOSPLUS and dual-drive cable.

The best news of all is the specially reduced package prices ...

SAVE $23 — $107 over our already low prices!

Choose the appropriate system from the table below:

DRIVE TYPE	No. of Tracks	No. of Heads	Capacity	Dosplus Version	Price	* Saving
DRIVE 0						
1 x MPI B51	40	1	100K	3.3	$499	$77.95
1 x MPI B52	40	2	200K	3.4	$639	$97.95
1 x MPI B92	80	2	400K	3.4	$799	$107.95
DRIVE 1						
1 x MPI B51	40	1	100K	—	$415	$23.00
1 x MPI B52	40	2	200K	—	$525	$23.00
1 x MPI B92	80	2	400K	—	$695	$23.00

*Represents the saving compared with buying all the items included in the package separately

•Drive 0 package includes one bare disk drive, self-contained single-drive cabinet/power supply as illustrated, two drive cable and the version of DOSPLUS indicated.

•Drive 1 package includes one bare disk drive and self-contained single-drive cabinet/power supply as illustrated.

If it's a dual-drive system you need, then take advantage of our dual-drive package and

SAVE a further $40 on the price of two single-drive packages ...

DRIVE TYPE	No. of Tracks	No. of Heads	Capacity	Dosplus Version	Price
2 x MPI B51	40 ea	1 ea	2 x 100K	3.3	$874
2 x MPI B52	40 ea	2 ea	2 x 200K	3.4	$1125
2 x MPI B92	80 ea	2 ea	2 x 400K	3.4	$1454

Dual-drive package includes two bare disk drives, self-contained dual-drive cabinet/power supply as illustrated, two drive cables and the version of Dosplus indicated.

NOTE: All 40 track drives are completely compatible with 35 track operating systems such as TRSDOS. DOSPLUS allows you to realise an additional 14% capacity compared with TRSDOS. Under DOSPLUS 3.4, 80 track drives can read 35/40 track diskettes.

All disk drive components are still available separately:

BARE DRIVES — MPI drives offer the fastest track-to-track access time (5 milliseconds) available. All drives are capable of operating in double density for 80% greater storage capacity.

	Price	Freight		Price	Freight
MPI B51 40 track, single-head, 100K	$349	$5.00	Self-contained, single drive cabinet/power supply	$99	$5.00
MPI B52 40 track, dual-head, 200K	$449	$5.00	Self-contained, dual-drive cabinet/power supply	$135	$5.00
MPI B92 80 track, dual-head, 400K	$619	$5.00	Two drive cable	$39	$2.00
			Four drive cable	$49	$2.00
Separate, dual-drive power supply	$85	$8.00	DOSPLUS 3.3	$99.95	$2.00
			DOSPLUS 3.4	$149.95	$2.00

Prices are FOB Adelaide. Add $5.00 freight for single drive package, $10.00 for dual-drive package. Prices are in Australian dollars. Freight is road freight anywhere in Australia.

All items carry a 90-day parts and labour warranty. Repairs to be carried out in our Adelaide workshops.

MICRO-80

LEVEL 2 ROM

ASSEMBLY LANGUAGE TOOLKIT

by Edwin Paay

FOR TRS-80 MODEL 1, MODEL 3 AND SYSTEM 80/VIDEO GENIE

This is a new package consisting of two invaluable components:

- **A ROM REFERENCE** Manual which catalogues, describes and cross-references the useful and usable ROM routines which you can incorporate into your own machine language or BASIC programs.
- **DBUG**, a machine language disassembling debugging program to speed up the development of your own machine language programs. DBUG is distributed on a cassette and may used from disk or cassette.

Part 1 of the ROM REFERENCE manual 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/Assembly tapes. There is a special section devoted to those additional routines in the TRS-80 Model 3 ROM. This is the first time this information has been made available, anywhere. Differences between the System 80/Video Genie are also described. Part 1 is organised into subject specific tables so that you can quickly locate all the routines to carry out a given function and then choose the one which meets your requirements.

Part 2 gives detailed information about each of the routines in the order in which they appear in the ROM. It describes their functions, explains how to use them in your own machine language programs and notes the effect of each on the various Z80 registers.

Part 2 also details the contents of system RAM and shows you how to intercept BASIC routines. 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!

The Appendices contain sample programmes 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.

DBUG: Eddy Paay was not satisfied with any of the commercially available debugging programs, so he developed his own. DBUG: allows you to single-step through your program; has a disassembler which disassembles the next instruction before executing it or allows you to bypass execution and pass on through the program, disassembling as you go; displays/edits memory in Hex or ASCII; allows Register editing; has the ability to read and write System tapes and all this on the bottom 3 lines of your screen, thus freeing the rest of the screen for program displays. Four versions of DBUG are included in the package to cope with different memory sizes.

The best news of all is the price. The complete Level 2 ROM ASSEMBLY LANGUAGE TOOLKIT is only:

— Aus. $29.95 + $2.00 p&p
— UK £18.00 + £1.00 p&p

SPECIAL OFFER TO OWNERS OF THE LEVEL II ROM REFERENCE MANUAL ...

UPGRADE TO THIS ASSEMBLY LANGUAGE TOOKIT FOR ONLY $19.95!

Send back your original Level II ROM Reference Manual plus a cheque, money order or Bankcard authorisation for $19.95 plus $2.00 p&p and we will send you the new ASSEMBLY LANGUAGE TOOLKIT

MICRO-80