

Memotech MTX World



Have your eyes ever been checked?

No, they've always been blue.

Issue NumberTwo

The journal of The Software Source

Published by
The Software Source
12 Roebank Road
Beith
Ayrshire
KA15 2DX

esk...NewsDesk...News Desk...

Editor - Alan Hamilton
 Printer - Stewart Harvey
 PD Software - Brian Houghton
 Commercial Software - Andy Fox
 Repairs - Paul Wood

Issue 2 - August 1989

"Memotech MTX World"

The Journal of The Software Source

(C) 1989 The Software Source

This publication is protected under international copyright law and may not be reproduced in any way, electronic, mechanical or otherwise without the express written consent of the publisher.

Scottish members: The Scottish Alternative Computer show is on the week after The Alternative Micro Show at the Livingston Forum. More details when we have them.

Indexes of the contents of Memotech MTX World are now available at £1 on paper or £2 on disc (excluding the disc).

I got a call from Paul Wood requesting that those of you with the new Series 2 MTX512's try something out. Get to ROM BASIC, and type the following:

PRINT PEEK (64122)

press RET and you should get an answer of 4. If you get anything less than that, and were not in FDXB if you have a CP/M system, then we need to speak to you!!

Being a Scot born and bred I am constantly worried about the amount of money which The Software Source has. I would therefore be grateful if members could pay close attention to the details we publish on how to go about ordering from us: ***IF IN DOUBT, ASK!!*** A couple of stamps for a reply always helps but isn't necessary. It just means your letter won't sit in the pile of post until the next trip to the Post Office.

For those of you unaware of what is happening, we have managed to get a fairly good write-up in Personal Computer World and it has brought some new members in. Our name and address is now etched into the history of micro-computing in the users group directory.

I need **DESPERATELY** programs for printing in the journal. Big ones, small ones, fat ones, thin ones, BASIC, Assembler (especially), Noddy, Pascal, Forth, C, ANYTHING! I would like to keep the Programmer's Notebook going but as of this journal, I have nothing to put in it. I also need your help with some people willing to write regularly for me. **REMEMBER: THE SOFTWARE SOURCE PRIDES ITSELF IN THE SERVICE WE PROVIDE. THAT INCLUDES THE JOURNAL. WE REQUIRE YOUR HELP TO KEEP OUR STANDARDS UP BY PROVIDING US WITH MATERIAL OF ANY SORTS FOR INCLUSION IN THE JOURNAL. PLEASE HELP US!**

Finally, I believe the standard of this journal is up to the last one, if not surpassed it. I hope you agree.

Contacts

General

For general information, membership renewal, journal contents, marketing, advertising, constitution, commercial software, 3.5" to 5.25" to 3.5" conversions contact: **THE SOFTWARE SOURCE, 12 ROEBANK ROAD, BEITH, AYRSHIRE, KA15 2DX. TEL: 0505 52491**

PD Software

For all public domain software orders, latest updates, and general enquiries about the PD Library: **THE SOFTWARE SOURCE PD SOFTWARE LIBRARY, 1 SAXON DRIVE, ROWLEY REGIS, WARLEY, WEST MIDLANDS, B65 9RD, NO PHONE CALLS PLEASE.**

Hardware Problems

For all hardware repairs / reconfigurations, communication difficulties or hardware problems, contact: **PAUL WOOD, 12 BISHOPS AVENUE, WORCESTER, WORCS, WR3 8XA. TEL 0905 24260.**

Hotlines

If you have a programming problem, we are here to help: (Please phone between 7pm and 9pm on week-days and in the afternoon and between 7pm and 9pm at weekends)

BASIC - A Hamilton 0505 52491

Assembler - P Wood 0905 24260

CP/M - B Houghton at PDSL address

Pascal - A Hamilton 0505 52491

All others - B Houghton at PDSL address

No phone?

If you cannot get to a phone to contact us about anything, we are pleased to receive your letters on any computing problems you may have.

Can you help us?

Do you have any ideas on anything that The Software Source could provide for its members? Can you help to provide something which we can't? We want to hear from you if you can help us! Please!!!!

Contents

- 1 - Editor's Page
- 2 - Public Domain Software Library
- 3 - PDSL Charges
- 4 - Competition winner & Non-programmer's competition
- 5 - Hardware Shop & New Software Announcement
- 7 - Programmer's Notebook
- 7 - Programming Competition
- 9 - Local Users Groups
- 9 - Repair Service
- 10 - BASIC Programming - II
- 13 - Mistakes
- 14 - Amstrad DMP2160 Review
- 15 - Classified Advertisements
- 17 - Disc Conversion
- 21 - Dear Ed...
- 23 - Important Announcement
- 27 - Blank Media
- 28 - Basically...
- 30 - Undocumented FDxB
- 31 - Go Forth! (-83)
- 36 - Introducing Pascal
- 40 - CP/M Section
- 42 - Glossary
- 43 - Contacts
- 44 - News Desk

Editor's Page

This journal, text-wise has been ready to roll since the end of July but I wasn't happy with the quality of the print in the last one and having got a couple of new DTPs recently, I started to muck about with them. It was looking as if we would have to stick to the old one for this journal, but TimeWorks decided to behave itself and it is on that that I have published this journal.

Anyway, enough of my rabbiting on, we have a new member of staff to welcome aboard: Take a bow please, Andy Fox. He's going to be in charge of the commercial software which is undergoing a major overhaul. All orders for commercial software should still be sent to me since Andy will be moving soon and we want to avoid losing orders.

And now a plea for help. I am looking for people to write regular columns in the journal. I am particularly interested in someone willing to write about BASIC/Noddy. Other subjects open for inclusion are robotics, Communications, reviews, Forth, Programmer's Notebook, in fact any section of the journal is open for delegation apart from the CP/M section which Brian is running.

Poor Brian has been doing sterling service by running the PD library and supplying a lot of the material for the journal and I would like to take the pressure of him (and me!). Poly term will have started by the next journal so I would be very grateful for your help - the standard of the journal will suffer without it.

The Alternative Micro Show looms closer and closer. Most if not all of "The Team", (Brian, Paul, Me, Andy) will be there for a natter. We've got two stands lined up so you can't miss us!

There are a lot of big things happening in the background with The Software Source, the MTX is back on its feet!

Final few things: Thanks to our resident poet for a very satirical look at The Software Source - I haven't printed it since I don't want to be done for libel! Journal number 3 will be mailed on the 30th of October 1989 and the deadline is 13th October. Those wishing to write for me please let me know very quickly so I can leave room in the journal for you. Come on guys! Give us a hand!

Alan Hamilton

Glossary

ARRAY - A very large variable which is split up into lots of little parts to allow you to slot in information. Is usually likened to a maths matrix.

CONFIG - A CP/M program which is used, normally at startup to tell CP/M the number and capacity of the drives connected to the system.

DOT MATRIX - A type of printer which has an array of (normally) 81 pins arranged 9 x 9 in order to print. As each character comes to be printed, the pins arrange themselves to form the character and the image is formed on the paper.

FDXB - Memotech CP/M BASIC. Most people have differing versions, but most seem to work.

FONT - A font is a different design of character (text). On this page, there are two fonts, the font that makes up these words, and another as the heading.

HIGH-LEVEL LANGUAGE - A programming language which uses words rather than symbols (like assembler). Some high level languages are BASIC, FORTRAN, COBOL, Pascal, COMAL, Pilot, C, Modula-2.

IMBEDDED - A term applied to the insertion of loops one after another

in a program.

INPUT - The BASIC command to receive information from the user.

INTEGER - A whole number like 1 or 2. Integers are NOT decimals like 1.5.

LOOP - In BASIC, a FOR...NEXT sequence of commands. In Pascal, a set of REPEAT...UNTIL, FOR...END, WHILE...END, WITH...DO

OBJECT CODE - Your raw BASIC, Pascal or Forth program is the source code, which is what you type in, the object code is what is left of it when it gets turned into assembler.

PASCAL - A High-level language used a lot in industry and professional programming. Very fast and easy to use.

PRINT - The BASIC command to display something on the screen. Also what printers do.

STEP - Used in a FOR...NEXT loop in BASIC to increment (increase) the controlling variable by an amount more or less than (but sometimes the same as) 1.

are aborted owing to a clash between the Pascal stack and the Guardian RSX. This is usually avoidable by lowering the stack when recompiling the program (adjusting CP/M to give the maximum possible TPA is also a good idea). Alternatively, since EZCPR can be decommissioned by dropping the Guardian and rebooting, exceptionally resistant programs can be run under CP/M. I am writing to CPMSDOSUGUK to get ZCPR2. If my information that it can handle extra memory as a RAM disk is correct, it might be an attractive alternative operating system for MTX 512-2 users. I have (12/07/89 discovered that there is a PD version of ZCPR3. It is a rather large project (5160 kB discs including the MACRO-80 source files) but does have the additional ability to set up named directory files (as in UNIX or MS-DOS), and to require a less specialised installation routine than that needed by its predecessor.

OU 'EST LE MENU?

A Menu Driver (for those readers who were about to ask) is a program which displays the names of some or all of the executable files on a disk and allows you to select one by typing a number or letter or by moving to cursor to an item and typing a control key (usually <RET> or <ESC>). The lack of one is a frequent (and reasonable) criticism of CP/M. FRONT50 (FRONT Version

5.0) is a very good menu driver which, unlike many commercial MS-DOS ones, allows optional entry of command-line parameters when an item is invoked - i.e. you can not only call NewWord, but call it with a named file to edit. The only problems I've found with it are that unlike some commercial menu drivers, files you don't want must be hidden by being given the \$SYS attribute and that despite the very full installation data provided, I cannot persuade it to handle the MTX cursor command. I suspect that it sends nulls in with the command string (the MTX hardware will not accept this). Until someone finds a way to solve this, this means that FRONT must be installed for TVI and that the subject programs must either be installed for a Televideo 920 or Lear-Siegler ADM-3A terminal or must be 'well-behaved' programs.

EZCPR is on CPM #011

FRONT is on CPM #021

Brian Houghton

PD Software Library

We have one addition to the cassette-only library:

CA28 Megaliner.

This is another stage in Nigel and Mike's quest for the largest printed characters they can generate. We also now have copies of the following:

CA03 Money Manager

CA05 Dbase III

CA08 Filetech

CA11 Elements

Incidentally, Money Manager, Dbase-III and Megaliner have resisted all my attempts to archive them to disk. Any suggestions would be most welcome, as tape archives are not exactly convenient. Is it possible that FDX Basic (which has at least one known bug in its handling of source files) can't save a source file that crosses an extent boundary, or are these large ROM Basic programs overwriting something? SDXB7.COM, by the way, is not a practicable alternative host program as it is merely FDXB.COM renamed.

We have two new CP/M disks. CPM029 ZMAC & ZLINK has some very interesting stuff on it. ANYCODE is an assembler patch for WordStar

which allows any possible printer controls to be sent to the LST: device by typing them just as they appear in the printer manual. It works, too - I'm most impressed with it. QK20.COM (QuikKey) is a very powerful keyboard macro definer, which will handle up to 31 keys, save and load its own datafiles and which, being a TSR (terminate and stay resident) program, can be continuously updated while you are working. I am typing this on Wordstar with the accessory keypad redefined to make it feel like NewWord. Most of the time QK can replace the now very old F.COM (an original SM-1 program) which is much less easy to use. ZMAC and ZLINK are a Z-80 relocating assembler. Relocating assemblers let you build very large programs piecemeal. As far as I can tell from a quick look, this early version of ZMAC does not appear to have macro facilities, although a PD alternative to Macro-80 is said to exist.

CPM030 is JRT Pascal. JRT Pascal was a very successful commercial Pascal P-system whose makers eventually put it in the Public Domain. This package is possibly the most important item the library has obtained. NOW IS THE OPPOR-

-TUNITY FOR MEMBERS TO GET AWAY FROM BASIC AND ASSEMBLER! Pascal is an important language, which can provide almost complete portability of source-code between different computers, compilers and operating systems. Unlike BASIC (which looks easy and is difficult), Pascal is extremely easy to learn, and is very fast and very powerful.

Brian Houghton

PDSL Charges

Those of you who may have ordered from the PDSL recently may have been surprised to receive an invoice. Those of you that phoned me up to ask what it was know what I am about to say, but for those that haven't, a little background:

With the sheer size of the journal, just about all of the membership fee which you paid when you joined is used up by the production and mailing. Which is what, technically, you have paid for. However, since Brian and I are both pessimists, me being one simply through being a Scotsman with his hands on the bank balance, we could see The Software Source collapsing before it ever got inflated due to constrictions on the account. We have had to lay out quite a bit of cash through the purchase of decent quality discs to keep backups of the PDSL and the production of the new "Guide To The

Software Source". Anyway, there is no earthly way which the coffers could subsidise the immense amount of mail that travels up and down the country for one reason and another so, to bring a little extra income into the bank account, I have had to impose a small service charge per program to the PD software library.

The charge is 45p per program on cassette or £1 for a main library or CP/M library disc. This service charge is in addition to the rules about sending us your media, envelope and return postage.

Those that I have spoken to about this feel that a small charge is quite acceptable but I realise that some may not think so, in that case, I am open to listen to reasonable case.

In view of the fact that many of you also have ordered cassettes and discs from me, at what are very low prices (with no profit margin at all) for such items, I think that this imposition is fairly reasonable. Your cheques, even though they should be sent to Brian, should still be made payable to The Software Source, postal orders and international money orders are also OK, but please don't send cash.

Anyone wishing to discuss this, I am available on 0505 52491 after 7pm weeknights.

Alan Hamilton

CP/M Section - II

Most people know that there was a near-duel between Digital Research and Microsoft for the IBM operating system. Most people are not aware that, although CP/M was the first real 8-bit operating system, there were several compatible alternatives. I do mean compatible alternatives, and am not talking about TurboDos and similar things. ZCPR (for acronym enthusiasts ZCPR = Z-80 Console Command Processor Replacement) was introduced as a program generating a new CCP in the early '80's. Whereas MSDOS was a CP/M upgrade, ZCPR was based upon UNIX. ZCPR soon evolved into ZCPR2. This is still around as a PD program. Its only real problem is a complex installation procedure involving a partial rewrite of CP/M (and not the thing to start in a spare half-hour!). Within a year or so (1983-'84) Echelon Inc. introduced the commercial program ZCPR3 supported by a complete operating system called ZRDOS (or Z-System). A PD introduction to ZCPR called EZCPR appeared about the same time. We have got EZCPR (CPM011 EZCPR) and I am currently trying to get ZCPR2 from CPMSDOSUGUK. "Oh, ****! Another operating system!" ZCPR is much more powerful than

MS-DOS and might have become the main 8-bit system had not history overtaken it. I have been using EZCPR for nearly a year, and have been unable to provoke it into crashing even with such programs as F.COM and Write-Hand-Man which overwrite parts of the BIOS. Wouldn't you like to type such command lines as: "format b:;syscopy b:=a.;pip b:=nswEEP.*[ov];dir b:" and go away while they were acted upon whatever drives the invoked programs were on? Or to have the ability to move between user areas by typing 'drive[user]:<RET>' and to move data and programs between them with a utility like the MS-DOS 'XCOPY' program? Or to handle user areas (32 of them) like MS-DOS/UNIX directory paths with planned search routes between them? These are the main EZCPR facilities. Others include the ability to read a changed disk without having to log-on to it, password protection for 'dangerous' facilities - even if you are the sole user of your system, the need for a password on such utilities as FORMAT may at least prompt you to think before you engineer a disaster - and an impressive set of support utilities. The only problem I have had with it is that some Pascal programs

converted into machine code so that the computer can use your program. This conversion takes time and it is for this reason that MTX BASIC is comparatively slow. Pascal, on the other hand, is a compiled language. Before you ever run a Pascal program it is compiled into machine code. So, when you come to actually run it, you are actually running machine code which doesn't need to go through an interpreter and hence is far faster than BASIC.

OK, you see the advantages of Pascal over BASIC, but what can you do about it? Well, it rather depends on your computer. If you have a tape based MTX, you'll need the HISOFT Pascal ROM (available from us). If you have a CP/M system, JRT Pascal is a PD compiler available from us or you can get Borland Turbo Pascal, Prospero Pro Pascal, Pascal-80 and various others as commercial packages.

There are lots and lots of books on programming Pascal but I'd like to end by recommending just a few:

Pascal-An introduction to methodical programming.

FINDLAY, William & David A Watt.
Pitman 1985 3rd Edition ISBN 0-273-02188-5

Illustrating Pascal

ALCOCK, David.

Cambridge University Press 1987
ISBN 0-521-33695-3

Programming in Pascal

ZWASS, Vladimir

Barnes & Noble 1985 ISBN 0-06-460201-X

The Pascal Handbook

TIBERGHEN, J.

Sybex 1981

Finally,

Turbo Pascal Program Library

RUGG, Tom & Phil Feldman

Que 1986 ISBN 0-88022-244-1

**INTRODUCE A
FRIEND TO THE
SOFTWARE SOURCE
AND GET HALF PRICE
MEMBERSHIP FOR
YOURSELF!- THE
IDEAL CHRISTMAS
GIFT FOR THE
MTX'ER IN YOUR
LIFE!**

Competition

From the last journal, we have a winner for the programming competition, and a worthy winner it is too.

The grand prize goes to

Mr Paul Trainer

for his excellent graphics demonstration of the MTX.

His program has been added, somewhat at the last minute, to the PDSL as number 105 for you all to have a look at and impress your friends with.

If Paul would like to get in touch with me (Alan) telling me what he'd like up to a maximum of £10, we'll sort it out for him.

Considering that we're not all programmers, we have started a non-programmer's competition (see right), the same prizes stand as for the programming competition on page 7, but please note the different address.

WELL DONE PAUL !!

Non Programmer's Competition

Just answer the following questions and complete the tie breaker:

- 1) What central processing unit (CPU) does the MTX run on?
- 2) At what frequency does it update itself?
- 3) Which company makes, or made, the Video Display Processor which the MTX uses?
- 4) Name two other computers which use the same CPU and VDP as the MTX.

Tie Breaker:

I think The Software Source is worth my membership fee because... in not more than ten words.

All entries to: Competition, The Software Source, 12 Roebank Road, Belth, Ayrshire, KA15 2DX.

Winner decided on the best tie breaker from the correct entries. Closing date is the 6th of October 1989. Editor's decision in final. Prize is £10 of anything we sell or free membership.

Hardware Shop

The Software Source hardware shop sells all you need for the computing DIY enthusiast looking to improve their machine:

32K Memory expansion Boards (with replacement PAL chip) - £19.99

64K Memory expansion Boards (with replacement PAL chip) - £29.99

RS232 boards (both sockets provided) - £19.99

Pascal ROM (including manual and expansion board) - £19.99

Newword ROM (including manual and expansion board) - £19.99

We are also able to provide a unique system for all existing non-CP/M users which means that they will no longer have to shell out vast sums to run the power of CP/M systems. We have 56 column CP/M chips which replace your existing non-CP/M ROM to give you CP/M in 56 columns and **TV OUTPUT - NO MONITOR IS REQUIRED!** Run Newword, the CP/M version on your telly as a colour monitor!!! Please phone for a quote for your system!

All cheques/postal orders/International money orders payable to The Software Source. Send all orders to: The Software Source, 12 Roebank Road, Beith, Ayrshire, KA15 2DX. Allow two weeks for delivery. Prices full inclusive of VAT, Postage and packing.

New Software

An exciting batch of new software is currently undergoing trials and will be ready for release in the next journal. In this batch is the first ever complete LOGO implementation for the MTX, cassette backup for CP/M programs, a complete set of CP/M tools, Invoice management and mail label systems, to name but a few. We hope to be able to supply full reviews of these products in time for the next journal.

Who said The Software Source wasn't doing anything?!

```
COUNTER:=1;
REPEAT
  WRITELN('Element ',COUNTER,' is now ',ELEMENT[COUNTER]);
  COUNTER:=SUCC(COUNTER);
UNTIL (COUNTER=20);
END. {Program}

10 DIM ELEMENT(20)
20 FOR LOOP = 1 TO 20
30 PRINT "Enter element ";LOOP
40 INPUT ELEMENT(LOOP)
50 NEXT LOOP
60 PRINT "Sorting..."
70 FOR LOOP=1 TO 20
80 FOR LOOP2=1 TO 19
90 IF ELEMENT(LOOP2)>ELEMENT(LOOP2+1) THEN TTEMPORARY =
  ELEMENT(LOOP2):ELEMENT(LOOP2)=ELEMENT(LOOP2+1): ELE-
  MENT(LOOP2+1)=TEMPORARY
100 NEXT LOOP2
110 NEXT LOOP
120 COUNTER=1
130 IF COUNTER=20 THEN STOP ELSE PRINT "Element ";CONTER;" is now
  ";ELEMENT(COUNTER)
140 COUNTER=COUNTER+1
150 GOTO 130
```

The BASIC program to me is overly complicated, just look at line 90! It's all crammed into one line which, to someone who knows little about BASIC might find very confusing.

BASIC on the MTX is an interpreted language, i.e. as the program runs, it is

line numbers, we have to tell it where our program starts and stops, hence the BEGIN and END commands. WRITELN probably speaks for itself.

Although the programs look different, you can now see that there is very little variation in them. Let's work in reverse this time. Here is a pascal program and its BASIC counterpart.

Program Bubblesort;

VAR

LOOP, LOOP2, COUNTER, TEMPORARY : INTEGER;

ELEMENT : ARRAY [1..20] OF INTEGER;

BEGIN

FOR LOOP:= 1 TO 20 DO

BEGIN

WRITELN('Enter element ', LOOP);

READLN(ELEMENT[LOOP]);

END; {LOOP}

WRITELN('Sorting...');

FOR LOOP:= 1 TO 20 DO

BEGIN

FOR LOOP2:=1 TO 19 DO

BEGIN

IF (ELEMENT[LOOP2]>ELEMENT[LOOP2+1]) THEN

BEGIN

TEMPORARY:=ELEMENT[LOOP2];

ELEMENT[LOOP2]:=ELEMENT[LOOP2+1];

ELEMENT[LOOP2+1]:=TEMPORARY;

END; {IF}

END; {FOR}

MTX Fontastic

A new Epson compatible printer utility

11 different general purpose icons.

10 different character fonts

10 different underscore patterns

Fully interactive text editor.

Address label functions

Giant poster printing functions which prints down the page instead of across. Print gigantic banners in seconds!

All fonts can be underlined, combined and altered at the press of a few keys.

A snip at **Only £5.00**

Now available on 3.5" & 5.25" non-CP/M disc as well as cassette!

Please make your cheque payable to PAUL TRAINER and send your order to:

1 Montagu Gardens
Oakwood
Leeds
West Yorkshire
LS8 2RN

Further details available on (0532) 498985 evenings only.

Programmer's Notebook

AUTHOR: A HAMILTON

REQUIREMENTS:

LANGUAGE: MTX BASIC

NOTES: This is a very handy little routine which anyone who has ever tried to centre something on the screen will find very useful. Enter your message in A\$, adjust SCREENWIDTH accordingly to suit the mode you're in and make YLINE equal to the line you want the text to appear in.

```
10 LET A$="G'DAY MATE!": LET SCREENWIDTH=40: LET YLINE=10
20 CSR (SCREENWIDTH-2-LEN (A$))/2,YLINE:PRINT A$
```

AUTHOR: A HAMILTON

REQUIREMENTS:

LANGUAGE: MTX BASIC/NODDY

NOTES: This method of displaying Noddy pages removes the necessity to create controlling program pages and then the text page. Type in the BASIC program and create a Noddy page called TEST. Type something on to the screen and on the very last line (press HOME and move up) enter the Noddy commands shown below.

The BASIC program: 10 PLOD "TEST"

The Noddy commands: *D TEST. *E *R

Only one Noddy page is required to run this. You have managed to combine a program and Noddy page in one!

Programming Competition

Following on from the last one, this time I'd like the best sound demo in BASIC in less than 5K. Must be BASIC and must be less than 5K. Prize is £10 of anything The Software Source sells, or free membership when your membership expires. All entries to: Competition, The Software Source, 1 Saxon Drive, Rowley Regis, Warley, West Midlands, B65 9RD. By the 6th of October please.

Introducing Pascal

It is a constant source of amazement to me why more Memotech users have never taken up writing in Pascal rather than BASIC or Assembler since it is both a high level language like BASIC and runs at precisely the same speed as Assembler and, to make life even easier is available not just for the highly-powered CP/M systems.

Pascal is a scientific language - so is BASIC. Its main strong point is its ability, much more so than MTX BASIC of doing very fast calculations and performing operations on complex mathematical formulae. It is a much easier language to write something like a spreadsheet in since there is no need for fancy routines needing to be written to make sure that numbers all line up correctly in a column. There are procedures, WHILE...END loops, REPEAT...UNTIL and on and on, so why is there practically no-one in the Memotech world using it?

I can only assume it's because those people out there who have got to the end of their tether with BASIC and its lack of speed but can't for the life of them suss Assembler just don't know anything about Pascal.

So where do we start? Well, let's get

going with showing you the differences between a sample Pascal and BASIC program.

```
10 FOR T=1 TO 200 20 PRINT "HI
THERE. THIS IS LOOP ";T 30 NEXT
T
```

PROGRAM Loop;

VAR T : INTEGER;

```
BEGIN FOR T:=1 TO 200 DO WRI-
TELN("HI THERE. THIS IS LOOP ",T);
END.
```

You'll notice straight away that the FOR command appears in both programs, and yes, they do the same thing. You'll also notice that strange "VAR T : INTEGER;" thing. I'll explain

In BASIC, you can suddenly start using variables anywhere you like. Just slot it in as required. However, say you come to look at this program in a year's time and wonder just exactly what that variable is there for, what it does etc - you'll be very lucky to figure out what it is. In Pascal you have to define all the variables that you are going to use. The VAR command just tells Pascal that what follows is the declaration of a variable, in this case, the letter T. The : INTEGER part just tells Pascal that T is an integer (whole number).

Because Pascal does not use line

Memotech MTX, SDX & FDX Series

AFW Software Manuals - The Next Step Forward In Programming

Advanced Reference Manual.....£15

The Software Source Journal (v1 i1) say that "its a gigantic book...it is a friendly book, it is clear and concise and where possible avoids the excessive use of computer terminology. The listings are complete modules, which you could slot in easily to any program. Another point so often overlooked is that all such programs have been tested on a CPM system. This book is the sort of technical publication that has been missing off the bookshelves of many a MTX programmer...you should be able to enhance your programming capabilities dramatically."

MOC (v5 i4) say that "the first thing to strike me about this creation was the professional manner in which it was layed out. The bulk of the manual concentrates on MTX graphics. Covering every aspect in detail...assembler routines run riot in this manual. It is of good quality."

Tape To Disc Conversion Booklet.....£5

Not only does this 37 page laser printed manual guide you through the ins and outs of Tape to Disc Conversion, with the help of 9 worked examples - Sepulcri, Agrovator, Quazzia, Toado, Qogo, etc - but as MOC (v4 i4) say in their review of the product "...it makes good use of assembler and the Front Panel. The booklet has plenty of useful routines and explains many of the differences between the different MTX disc systems..."

"AFW Software, aka Alan Wilson, has been responsible for the other really handy reference book, "MTX Tape To Disc Conversion Booklet" which revolutionises the conversion of many commercial games to disc", say The Software Source.

AFW Software

24 Blenheim Place, Larbert,

Stirlingshire, FK5 4PP

Software Source members get a £1 discount on both products! Just quote your membership number

THE THIRD ALTERNATIVE MICRO SHOW AND ELECTRONICS FAIR

*SATURDAY 11 NOVEMBER 1989
BINGLEY HALL, STAFFORDSHIRE SHOW CENTRE
10 AM - 5 PM*

NOW ESTABLISHED AS THE SHOW FOR
THE COMPUTER ENTHUSIAST
THE ALTERNATIVE MICRO SHOW IS MOVING
TO A LARGER SITE AND INTRODUCING
FOR THE FIRST TIME AN ELECTRONICS FAIR

SPECTRUM/BBC/ATARI 8 BIT/QL/EINSTEIN
MEMOTECH/MSX/ADAM/ZX88/LYNX/DRAGON
COMMODORE 8 BIT/JUPITER ACE/TEXAS TI99-4A
ORIC/ENTERPRISE

IN FACT ANY HOME MICRO OTHER THAN ST/AMIGA AND PC's
THE ALTERNATIVE MICRO SHOW AND ELECTRONICS FAIR
WILL BE EXTENSIVELY ADVERTISED IN THE
NATIONAL COMPUTER PRESS, LOCAL RADIO
AND USER GROUP MAGAZINES ETC
AND IS IDEALLY TIMED FOR THE CHRISTMAS TRADE

DON'T DELAY - BOOK TODAY!

Local Users Groups Repairs

One of the problems of running a users group of this size is that we never get to meet the members on a semi regular basis. Given, however, that "The Team" are spread all over the country, the chances are that you have someone within striking distance of you that is involved in the running of the group.

We'd like to see the initiation of local users groups of members of The Software Source to get to know other MTX'ers in their area and to sort out any computing problems you might have, face to face.

You could get together to start a programming project, or something like that, or just for a natter.

If you are interested in getting something like this going in your area, then please let Alan Hamilton know.

The first place I would like to see one is in Scotland since there is quite a population of Memotech users here - more than I thought! John Hodgson seems interested in getting one started in the Bristol area. Any takers?? There are also a lot of London MTX'ers and a surprising amount in Wales.

If you have an item of computer hardware, not just Memotech, that you'd like fixed then The Software Source can help.

In association with Paul Wood, we are able to run the ONLY Memotech repair facility which is guaranteed not to have all the hassle involved in sending something to MCL.

We also guarantee lower charges, much lower, than MCL.

Amongst the many things we have done recently are: non-CP/M to CP/M disc drive conversions, memory upgrades, ROM chip replacements, power supply unit faults, replacement video boards.

If you feel you want rid of a bit of hardware, we are willing to pay a reasonable price for the spare parts, which are in ever-decreasing supply these days.

For further details on how we can help you with your repair, contact Paul Wood on 0905 24260. Or at 12 Bishops Avenue, Worcester, Worcestershire, WR3 8XA

cursor.

KT xxxx (Keep-Till) copy characters between cursor and end of xxxx into insert buffer.

MULTIPLE FILES

F83 has a special vocabulary called 'FILES' which allows a limited amount of multiple file handling, with possibility for considerable expansion. Its most useful word is its local version of LOAD, which takes both a screen number and a filename, and which loads from an external file and then returns to the current file. Since the word OPEN closes the current file and opens another it can, with care, be used for interfile transfers: e.g.

OPEN source n BLOCK where n is the block for transfer,

#blk #buffer 1024 CMOVE,

OPEN dest,

n1 BLOCK,

#buffer #blk1 1024 CMOVE,
UPDATE FLUSH

etc.....

MORE

Files can be expanded by using MORE (n ---) where n is the number of extra screens. This word is not as automatic as in some later versions of F83 (enthusiasts may wish to redefine it). The following syntax is needed to may sure you really do get n more screens: n MORE <RET>

EDITOR <RET>

n N WIPE UPDATE TIMES FORTH
FLUSH <RET>.

WIPE is needed because the rather primitive file system which survives from FORTH's early pre-CP/M days will read garbage from the disk, while each block must be UPDATED so that the system may 'know' that it's there.

SUMMARY

I am not sure how I feel about FORTH, which is great fun to use, but in which published programs have a somewhat incestuous quality of being mostly tools to make it easier to write tools.... in a depressingly recursive manner (incidentally, F83 supports a very simple recursive form in which definitions are simply terminated with 'RECURSE' as opposed to the complex syntax of other FORTHS), but F83 seems simple, trouble free compared to the loading horrors described by Peter Burns in "MEMOTECH MTX WORLD" #1 and is extremely fast.

*The Software
Source
- The Fastest
Growing Memotech
Users Group!*

screen each time you edit a line. When editing 'A' toggles between screen and shadow and 'DONE' behaves as described above. If you get into a real mess then COLD will reset the whole Forth system or QUIT will abandon the edit. When you enter the Editor, Forth asks "Enter your ID....." This is a little misleading; it is asking for a label rather than a password. The usual thing is the date and your initials. This will then appear at the end of line 0 on all the screens edited in that session, and is a very useful way of keeping track of what you've done.

GENERAL EDITOR COMMANDS

WIPE Clear current screen to blanks. WIPE does not take a parameter as in some other versions of FORTH.

A Toggle between current screen and its shadow.

N Load Next screen.

B Go Back one screen.

KEEP Copy current line into insert buffer.

K Exchange contents of insert and find buffers.

W Abbreviation of SAVE-BUFFERS. BUFFS Display find and insert buffers.

DONE Save screen to disk and leave editor.

n EDIT Enter editor at screen n.

ED Re-enter editor after DONE.

n TIMES Repeat commands on the line n times.

MANY Repeat commands until end of screen or until an error occurs.

LINE EDITING COMMANDS

n T Edit line n.

n +T Edit line current + n.

P xxxx Put string xxxx to current line and blank remainder.

U xxxx Insert xxxx under current line.

X Delete current line.

SPLIT Split current line at cursor.

JOIN Copy line beneath current line to the end of the current line.

n NEW Insert multiple lines of text starting at line n.

G (Screen# line ---) Get a line from another screen and insert it at the current line, moving the current and subsequent lines downwards.

BRING (Scr# L1 L2 ---) Get a range of lines and their shadows.

STRING EDITING COMMANDS

F xxxx Find string xxxx from the current cursor position.

R xxxx Replace the string found using F with string xxxx.

E Erase text just found with F.

D xxxx Find xxxx and delete it.

I xxxx Insert xxxx after current cursor.

TILL xxx Delete to end of xxx.

J xxxx Delete up to xxxx (i.e. Justify).

O xxxx Overwrite xxxx from current

BASIC Programming

Part Two

In the last Journal I left you with a program that displays the time and we had started off from arithmetic on the computer and progressed through variables and strings onto LEFT\$, RIGHT\$ and MID\$.

This time I want to explore loops and give some practical applications of what we have learned. Firstly though, let's see how we can get the user to type some input in for us.

You will recall that we used the command PRINT to display information on the screen. You will also recall that we could print numbers, variables and combinations of these. However, what happens if we require some information from the user? Well, we want the user to input information into the MTX and to do this we use the command, INPUT.

Usually INPUT will be followed by either a numeric or string variable. e.g.

10 INPUT A

10 INPUT B\$

However, INPUT has the ability to combine to an extent the functions provided by PRINT. e.g.

10 INPUT "What is the weather like?";A\$

would be the same as:

10 PRINT "What is the weather like?";

20 INPUT A\$

Notice a difference in the use of the semi-colon (;) here. With the INPUT command the semi-colon is used to separate the text which you wish to display from the name of the variable which you wish to store the user's input in. With the PRINT command, the semi-colon tells the computer not to take a fresh line with the next command which displays something on the screen.

What happens with both of these appears to be the same although obviously the first is a lot shorter than the second and consequently uses less memory.

An example follows using some permutations on what we've just learned.

```

10 PRINT "Please tell me your name
";
20 INPUT NAME$
30 PRINT
40 INPUT "What is your age ";AGE
50 PRINT
60 PRINT "Your name is ";NAME$;"
and you are ";AGE;" years old."

```

There is one thing to point out here and it's that semi-colon again. In the PRINT command, the semi-colon tells the computer not to take a new line. It can also be used to separate text from variables inside a PRINT command as shown in line 60,

Looping

When you have typed RUN to execute any of the programs so far, they've only run once and stopped. This is OK for simple programs like we've been used to. Say though, we wanted to do something a set number of times. This is when a FOR...NEXT loop comes into the fray. e.g.

```

10 FOR X=1 TO 20
20 PRINT X
30 NEXT X

```

Let's take each line at a time.

Line 10 - The command FOR tells the MTX that we are setting up a loop. The loop requires a variable to keep track of the number of times it

has gone round. In this case we have used the variable X, but you could use nearly anything. Next comes all the values X will have, i.e. all the whole numbers between 1 and 20.

Line 20 - This line simply prints the present value of X.

Line 30 - This is the important line which tells the MTX that the loop is over and to add one to the value of X and, assuming the upper limit of 20 has not been reached, to go round the loop again.

Here is another example:

```

10 FOR A=1 TO 100
20 PRINT A*A
30 PRINT A
40 NEXT A

```

RUN this and see what happens.

Assuming you have got a grasp of this, we also have the ability to run backwards through a loop. The FOR...NEXT system always adds at least 1 to the controlling variable. But, if we want to go from, say, 300 to 1, if it always adds 1 we'll never get there! So, the extra command STEP is used.

```

10 FOR X=300 TO 1 STEP -1
20 PRINT X
30 NEXT X

```

This will result in the computer dis-

assumes that the entire disk should be accessible as one continuous file, although you are free to divide your disk into smaller blocks if you prefer. After loading it type 1 LOAD, followed by EDITOR MEMOTECH FORTH. This loads the MTX terminal emulator which gives you the basic MTX screen controls and a continuous screen update during editing. To return to the basic Forth editor type EDITOR DUMB FORTH. An abbreviated (50 k) version of MTX.BLK (MTXS.BLK) is included for users with limited disk space. I am personally responsible for the contents of MTX.BLK. As far as I am aware, the defined words work as proposed. I make no claim for their efficient implementation: at least some would run much faster if defined with the very good 8080 Assembler vocabulary provided.

LEAVING To exit F83 type BYE. The operating system will reboot with the display: nn Pages A> To save an updated Forth type SAVE nn F83.COM.

ME AND MY SHADOW

(Sorry!). A Shadow Screen is a screen separated from the current screen by half the total capacity of the file. That is, if the file is 100k (100 screens) the shadow screens start at #50. Shadow screens are normally used for comments and descriptions. When editing a file, the word A toggles between the code screen

and its shadow. You aren't forced to use shadow screens, but, since a Forth file can encompass the whole disk, and removing comments from source screens both lets you make more detailed comments and speeds up the load and compile times which are one of the less acceptable aspects of Forth systems.

FILES

Before you write Forth programs, you must have a file to write them to. To log-on to a disk other than the one holding F83, type n SET-DRIVE, where n is the drive number as in CP/M assembler; A:=0, B:=1, etc. To load a file from outside Forth, you can invoke Forth with the filename, e.g. 'F83 C:MTX.BLK'. To load a file within Forth, type OPEN NAME.EXT. To create a new file, type nn CREATE-FILE NAME.EXT, where nn is the number of pages in the file (one page = 1 kilobyte). If you want to use shadow screens nn should be a whole number. To save a file after an edit session, merely type DONE. F83 contains the words UPDATE, SAVE-BUFFERS and FLUSH, but DONE invokes them in the proper sequence.

THE EDITOR

This is the usual Forth Mickey Mouse job, but is greatly improved by some extra commands and by a set of terminal emulators which give a continuous full-screen update of the

Go Forth! (-83)

The Forth-83 disk contains a document which is of more interest to those curious about the inner workings of the compiler than to those who have first met it. F83 is a serious professional development system and, as such, is a bit daunting even to those already familiar with simpler Forth implementations. This is an attempt to provide a brief introduction to Forth-83 and its editor and file system.

REFERENCES

One of the problems with Forth is that, like Basic, it has too many dialects. F83 is the original and definitive Public Domain version of what is coming to be as near to a standard implementation of Forth as we are likely to see. The reference books which I have found most readily available are: "MASTERING FORTH" Anita Anderson & Martin Tracy. Brady ISBN 0-89303-660-9 Price around £14.00 This is entirely devoted to Forth-83, and is probably the most readily-available complete introduction.

"FORTH, the fourth-generation language" Steve Burnap. TAB Books ISBN 0-8306-2687-5 #14.80

"STARTING FORTH" Leo Brodie, Prentice-Hall 1987, £24.00 The fol-

lowing book (which I have not seen) is said to be the definitive Forth-83 reference: "Inside F83" C.H.Ting, Ofete Enterprises, Inc., (Said to be available from F.I.G.), \$35.00 U.S. In 1989. .pa Please note that earlier editions of Dr. Brodie's book do not cover Forth-83. The book by Dr. Burnap is mostly devoted to Forth-83, but assumes an initial familiarity with Forth systems.

GETTING STARTED

FORTH.DOC gives a complex series of instructions for creating F83 from the Forth Kernel and the meta-compiler. F83.COM is the result of this. You therefore do not have to follow the instructions unless you have problems or require a limited subset of the compiler. Please note that the compacted *.BQK files together expand to about 600 kilobytes of *.BLK files.

LOADING

The simplest way to load Forth files is to type "F83 [drive:]name.ext". Forth doesn't display its "ok" prompt until the RET key is pressed after loading. I suggest that you spend some time preparing a manual by printing out the unsqueezed files on the Forth disk. To do this, after loading a file, type LISTING. This will

-playing the value of X which will start from 300 and decrease to 1. STEP can also be used to add a certain number to the controlling variable. Here are a couple of examples:

```
10 FOR X=100 TO 1 STEP -1
20 PRINT X*X
30 PRINT X
40 NEXT X
```

```
10 FOR Q=0.2567814 TO 5.6987121
20 PRINT Q
30 NEXT Q
```

However! FOR will always add 1 to the value of Q as it goes round so the numbers displayed would be:

```
0.25678174
```

```
1.25678174
```

up to

```
5.25678174
```

Which might not be much use. Again STEP comes in:

```
10 FOR Q=0.2567814 TO 5.6987121
STEP 0.2445138
```

```
20 PRINT Q
```

```
30 NEXT Q
```

Will STEP through the loop adding 0.2445138 each time.

With STEP you can also put in formulae:

```
10 FOR Q=1 TO 360 STEP (PI/4)
20 PRINT SQR(Q)
30 NEXT Q
```

or a more complicated example:

```
10 LET S=1
20 FOR E=0.98674 TO 11.4789
STEP (E/S)
30 LET S=SQR(E)
40 PRINT E;S
```

Loops can be embedded in each other to run through several loops at the same time. This is complicated but follows the same rules as before. Sadly, MTX BASIC does not indent program listings like LISTO on a BBC to make it plain how the imbedded loops are structured so although the listings below are indented, they will not be when you come to type them in on the MTX.

```
10 FOR A=1 TO 10
20 FOR B=10 TO 1 STEP -1
30 PRINT A
40 PRINT B
50 NEXT B
60 NEXT A
```

It is vitally important when using multiple loops like this that the NEXT commands appear in the reverse

order from what they defined. In other words, the inner loop, in the previous example signified by B, must be closed (by using NEXT B) before you can close loop A.

Here is a practical example of looping in a sorting program. Please note that this program employs the phenomenon of ARRAYS which we have not yet covered (coming up soon folks!) so although the actual workings of the program itself may not make a great deal of sense, you will at least be able to see the loops working.

```
10 DIM A(10)
20 FOR X=1 TO 10
30 PRINT "Enter element ";X;
40 INPUT A(X)
50 NEXT X
60 PRINT "Sorting..."
70 FOR LOOP1=1 TO 10
80 FOR LOOP2=1 TO 9
90 IF A(LOOP2) > A(LOOP1)
THEN LET TEMP=A(LOOP2):LET
A(LOOP2)=A(LOOP1+1):LET
A(LOOP1+1)=TEMP:LET TEMP=0
100 NEXT LOOP2
110 NEXT LOOP1
120 PRINT "Sorted!"
130 FOR X=1 TO 10
```

```
140 PRINT A(X)
150 NEXT X
```

Mistaykes

OK, we cocked up the listing of the VS 5 printer dump in the last issue. Mind you, no-one brought it to my attention and it was left to me whilst labouring into the night on a programming project to discover it. The offending line can be found on page 16 in the second column at the bottom right. the line reads:

```
4038 POP AE
```

which, to those with any knowledge of assembler is complete nonsense. the line should say:

```
4038 POP DE
```

Apologies to those who typed this and got "Bad Code" - I did!



```
510 DATA 426, 4, 2, .2, 426, 4, 379,
4, 358, 4, 478, 8, 568, 8, 2, .2
520 DATA 478, 4, 2, .2, 478, 4, 426,
3, 379, 4, 358, 8, 379, 8, 2, .2
530 DATA 426, 4, 2, .2, 426, 4, 379,
4, 358, 4, 379, 8, 2, .4
540 DATA 358, 4, 2, .2, 358, 4, 379,
4, 426, 4, 358, 8, 379, 8, 2, .2
550 DATA 426, 4, 2, .2, 426, 4, 379,
4, 358, 4, 478, 8, 568, 8, 2, .2
560 DATA 478, 4, 2, .2, 478, 4, 426,
4, 379, 4, 358, 8, 379, 8, 2, .2
570 DATA 426, 4, 379, 4, 358, 4,
319, 4, 358, 8, 1, 1
```

%%%%%%%%%%

Apologies for the lack of documentation with the listings in this journal. This is due purely due to lack of space - None of the programs require any special instructions and the sprite editor has its own anyway. If anyone experiences difficulty with the programs, I am most willing to help, just drop me a line or phone Alan at the normal address (can be found on page 43). If you have any programs suitable for listing, I'd love to use them! If you discover a bug in the listing, please tell me so that I can put the amendments in the next journal.

Undocumented FDXB Commands

All these commands are preceded by the command DISC:

REN "newfile.ext" = "oldfile.ext"

TYPE "filename.ext"

QUIT exits back to CP/M, or at least should, but I've never got it to work (or not as the case may be).

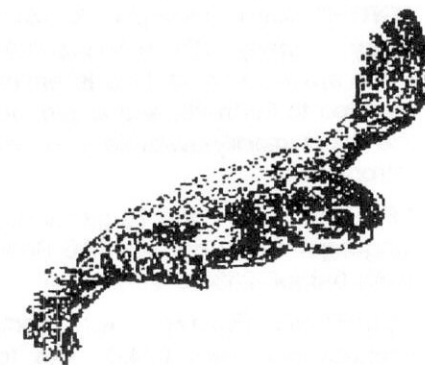
RUN "filename.ext" loads and executes an assembler file saved in the sdpecial .RUN format.

If you utilise the MTX/ROM 5 (non-CP/M disc system) technique, then all the FDXB commands are available except QUIT plus:

COPY

FORMAT

SYSCOPY



Hardware Review

Amstrad DMP2160 Dot Matrix Printer

Having previously had a Brother HR-10 daisywheel printer which performed beautifully I had to reckon with the prospect that I couldn't run something like The Software Source with it. So, I found myself looking for a dot matrix printer.

Having had some bad experiences with mail-order, I decided to trek round the four computer shops in Glasgow for something like a Panasonic KX-P 1081. Sadly, none of them stocked them and so that was the reason for looking at other machines. Particularly the Amstrad.

I can never claim to have been much of a fan of Amstrad stuff but this one seems fine. It's an odd design partly because it is black and partly because the paper sits underneath it and feeds in the front and out the back which I have found a marvellous system because you can insert single sheets of paper very easily and there's none of the usual groping around at the back of the printer, trying to put paper in it.

It has both tractor and friction feed

which was a prerequisite anyway and has a maximum character printing speed of 160 characters per second, which is slightly faster than the Panasonic.

Ribbons are cheap, and the manual is friendly with plenty of examples of programming the printer in BASIC.

The printer comes with NLQ print as well, again another prerequisite which also supports microjustification which in lay-mans terms means that instead of each character on the page taking up the same amount of space, a letter l and letter o take up only the space required to print them which makes the whole thing more natural.

It doesn't, unfortunately, have all the front panel controls like the Panasonic which I would have preferred, but then you can't have everything!

Price: £150-£160. Star Rating ****

Alan Hamilton

```

1040 LINE L,10,L,170
1050 LINE 10,L,170,L
1060 NEXT L
1099 RETURN
2000 REM CLEAR SPRITE
2010 FOR L=1 TO 32
2020 LET SP(L)=0
2030 NEXT L
2040 RETURN
3000 REM CHANGE OF BIT (W) OF
SPRITE (.V.)
3010 LET P=SP(V)
3020 IF P>W*2-1 THEN LET P=P-
W*2: GOTO 3020
3030 IF P>W-1 THEN LET
SP(V)=SP(V)-W: ATTR 2,1 ELSE LET
SP(V)=SP(V)+W: ATTR 2,0
3040 LET X=0:LET Y=V:IF V>16
THEN LET X=80: LET Y=Y-16
3050 LET Y=170-Y*10: LET
S=10:LET T=128
3060 IF W<>T THEN LET T=T/2:
LET S=S+10:GOTO 3060
3070 LET X=X+S: COLOUR 3,9
3080 FOR L=2 TO 8: LINE
X+2,Y+L,X+8,Y+L: NEXT L
3099 RETURN

0 REM LISSAJOU FIGURES
10 VS 4:COLOUR 2,1
12 COLOUR 4,1:COLOUR 3,15
14 PAPER 1:CLS:INK 15
20 LINE 0,0,0,190:LINE 0,0,255,0
30 FOR X=0 TO 1000 STEP 0.5
40 LET AS=COS(7*(X*PI)/360)
50 LET BS=COS(10*(X*PI)/360)
60 LET AS=AS*40+80
70 LET BS=BS*40+100
80 PLOT AS,BS
90 NEXT
999 GOTO 999

%%%%%%%%%%%%%%
0 REM TUNE
90 SBUF 120
95 LET FACT = 1
100 RESTORE 500
110 READ FREQ,DUR
120 IF FREQ<>1 THEN SOUND
0,FREQ*8*FACT,1000, 0, 10, 8*DUR,
1
170 IF FREQ<>1 THEN GOTO 110
180 LET FACT=FACT/20
190 IF FACT>=0.5 THEN GOTO 100
195 LET DUR=DUR/16
200 SOUND 0,0,0,0,0,0,1
500 DATA 358, 4, 2, .2, 358, 4, 379,
4, 426, 4, 358, 8, 379, 8, 2, .2

```

%%%%%%%%%%%%%%

Classified Advertisements

This is a service which we hope will appeal to all members: The ability to advertise anything and everything (provided it's legal!) **FREE OF CHARGE!!** Which can't be bad by any means. As if that wasn't enough, you can keep your advertisement in as long as required, just tell us when you want it removed. Simply either cut out or photocopy this form with all the relevant details and post it off to Alan. Please note that the wording of your advertisement should include your name and address etc. All advertisements are accepted in good faith and selling pirated software is illegal. Anyone found doing so will have their membership cancelled.

+++++

The Software Source Classified Advertisement Form

Name: _____ Membership Number: _____

Address: _____

Please insert my advertisement for (delete as appropriate)

1 2 3 4 5 6 more than 6 issues.

Here is the wording of my advert:

Signed: _____ Date: _____

Basically...

```

10 REM SPRITE DESIGNER
20 GOSUB 1010
30 DIM SP(32): GOSUB 2000
50 LET W=128:LET V=1
60 CSR 3,1: PRINT
"PRESS<SPACE> TO EXIT "
100 LET A$=INKEY$: IF A$<>" "
THEN GOTO 100
102 LET A$=INKEY$:IF A$="" THEN
GOTO 102
105 LET A=ASC(A$): IF A=32 THEN
GOTO 500
110 IF A=26 THEN GOSUB 3000
120 IF A<>10 THEN GOTO 130
125 LET V=V+1:IF V=17 OR V=33
THEN LET V=V-16
130 IF A<>11 THEN GOTO 140
135 LET V=V-1: IF V=0 OR V=16
THEN LET V=V+16
140 IF A<>8 THEN GOTO 150
143 LET W=W*2: IF W<>256 THEN
GOTO 150
146 LET W=1:LET V=V-16: IF V<1
THEN LET V=V+32
150 IF A<>25 THEN GOTO 100
160 LET W=W/2: IF W<>0.5 THEN
GOTO 100
170 LET W=128: LET V=V+16: IF
V>32 THEN LET V=V-32
180 GOTO 100
199 GOTO 100
500 REM PRINT THE LIST OF
NUMBERS
501 REM AS 16*16 SPRITE OR 8*8
510 VS 5: CLS: CSR 0,4: PRINT "
THE EIGHT BY EIGHT DEFINITION:"
PRINT
520 PRINT " GENPAT 3,0";
530 FOR L=1 TO 8:PRINT
";";STR$(SP(L));:NEXT L
540 CSR 0,10:PRINT " THE SIXTEEN
BY SIXTEEN DEFINITION:"
PRINT
550 FOR L=0 TO 3
560 PRINT " GENPAT";STR$(L+4);
0";
570 FOR M=0 TO 7: PRINT
";";STR$(SP(L*8+M+1));:NEXT M
580 PRINT: NEXT L
599 PRINT " TYPE 'RUN' TO DEFINE
A NEW SPRITE": STOP
1000 REM GRAPHICS MODE/DRAW
GRID
1010 VS 4:COLOUR 2,1: COLOUR
0,1: CLS
1020 COLOUR 4,13:COLOUR 3,4
1030 FOR L=10 TO 170 STEP 10

```

Blank Media

Cassettes

C12 - 49p	C20 - 59p
C30 - 64p	C50 - 69p
C62 - 73p	C70 - 78p
C80 - 83p	C92 - 87p
C96 - 93p	C100-98p
C120-120p	

Discs

Unbranded:

- 5.25" for 250K drives 40p ea.
- 5.25" for 500K or 1MB drives 50p ea.
- 3.5" for 1MB drives 115p ea.
- Branded (3M, Verbatim or Dysan) - you choose.
- 5.25" for 500K or 1MB drives 160p ea.
- 3.5" for 1MB drives 190p ea.

Labels

- (Supplied on tractor feed reams)
- 3.5" x 1.5" 1 across per 1000 £7.10
- 3.5" x 1.5" 2 across per 1000 £7.90
- 4.0" x 1.5" 1 across per 1000 £8.60
- 4.0" x 1.5" 2 across per 1000 £9.40
- 2.75" x 1.5" 3 across per 1000 £6.50

Printer ribbons

(only a selection - we can get any one you want)

- Amstrad DMP2000/3000 £3.50 ea
- Panasonic KXP1080/1090 £4.50 ea
- Star LC10 £4.50 ea
- Star NL10 £5.50 ea
- Epson FX/RX80/MX £3.50 ea
- Citizen 120D/LSP10 £4.50 ea

Mannesman Tally MT80/MT80+ £4.50 ea.

Paper

- 11" x 9.5" 60 gsm per 2000 £19.95
- 11" x 9.5" 70 gsm per 2000 £23.45
- 11" x 9.5" 80 gsm per 2000 £25.25
- A4 60 gsm per 2000 £24.30
- A4 70 gsm per 2000 £25.95
- A4 80 gsm per 2000 £29.55
- A4 90 gsm per 1000 £16.75

Misc

- 5.25" disc cleaning kit £5.95
- 3.5" disc cleaning kit £5.95
- Mini-Vac £7.95
- 80 Column printer stand £14.95

3.5 CP/M Disc system manual or SDX CP/M manual wanted. Must contain complete set of instructions for Newword and SuperCalc for purchase or loan. Contact: Mr Keith Wilkinson, 20 East View, Bedlington, Northumberland, NE22 7HD or phone 0670 829157 (evenings)

☆☆☆☆☆☆☆☆☆☆☆☆☆☆

Extended length MTX case top. Encloses motherboard, 80 column card and memory expansion card. Requires base plate. £2 ono + P & P (May be able to deliver depending on locality).

64K internal memory expansion board for MTX500. £30 ono + P & P as required.

Andy Fox, 8 Winston Gardens, Headlingley, Leeds, LS6 3LA. 0532 751053 eves.

☆☆☆☆☆☆☆☆☆☆☆☆☆☆

Sinclair QL Schon Standard Keyboard, casing and anti key bounce chip. All unused, unfitted and brand new with packaging and fitting instructions. £25 + P & P.

Memotech V-ROM for FDX CP/M systems. Gives access to non-CP/M disc operating system. Includes boot disc. £8 + P & P

unused and unopened. £5 + P & P.

Memotech PSU Power Supply. Complete and working. Bargain, £10 + P & P.

Alan Hamilton, 12 Roebank Road, Belth, Ayrshire, KA15 2DX. 0505 52491

☆☆☆☆☆☆☆☆☆☆☆☆☆☆

Wanted: Memotech non-CP/M disc system with single 5.25" disc drive. Willing to pay between £65 and £85. Ad placed for member in Germany, contact Alan Hamilton

#####

Quantity of keyboard overlays for the Memotech keyboard. Quality printed to fit over the function keys they are just the thing for use with Newword, FKEY or F.COM. Clearance price at £2 for 10. Send cheques/postal orders to Keyboard Overlay Offer, The Software Source, 12 Roebank Road, Belth, Ayrshire, KA15 2DX. Price is inclusive of VAT, P & P.

☆☆☆☆☆☆☆☆☆☆☆☆☆☆

Software reviews of commercial and public domain software wanted. Send to The Software Source, 12 Roebank Road, Belth, Ayrshire, KA15 2DX.

☆☆☆☆☆☆☆☆☆☆☆☆☆☆

Amstrad to centronics printer cable

Disc Conversion

Highway Encounter

After receiving my copy of Highway Encounter from MOC, I soon got disillusioned with it due to the time taken to load, and the wonderful way in which it always seemed to take three attempts before it would go in. By that time, the urge to play can easily wear off.

Converting it to run from disc seemed the best idea, but where to start as it doesn't employ the normal methods of loading, but pages out the rom to give itself more room. What's more, it must be on a 512 or bigger to run, and me with a 3.5" CP/M system can only have 32K FDXB with disc or 64K+ with MTX and no disc. Until, that is, the short article by Paul Wood in MOC introducing ROM 5 or MTX and disc functions without CP/M.

A couple of points I have found are:

- 1) Disc needs to be a 59K system, not 54K to utilise ROM 5, otherwise the DISC ERROR message occurs and the functions are not available.
- 2) Entry points in the rom seem to be the same as the SDX ROM.

Before starting on the conversion I'll cover briefly the concept behind the paging out of the roms that the Highway loader uses. In a rom based system, assume 64K, the user RAM is located on page 0 from 4000h to FFFFh and a final page on page 1 8000h to BFFFh.

Whilst in the ram based the final page is moved from page 1 to page 0 overlaying the roms from 0 to 3FFFh. What the loader does is to start in a rom based environment and load in the first block of code, then switch to a RAM environment and move the code to the block of memory overlaying the monitor and Panel roms on page 0. Then switching back to the rom system the code is preserved in the block which is now page 1, and a second block is loaded into "normal" memory. Again the roms are switched out, moving page 1 to page 0 and the new code relocated onto the end of the 1st block and then the game starts, without the use of the roms.

I presume that the program was developed on a CP/M system as the

NEMO	D	TD	0024
OBLITERATION ZONE	T	TD	0022
OBLITERATION ZONE	D	TD	0023
OBLIDS	D	TD	0063
PAINTBOX	T	TD	0085
PAINTBOX	D	TD	0086
PONTOON/BLACKJACK	T	D	0026
PONTOON/BLACKJACK	D	D	0027
POTHOLE PETE	T	TD	0028
POTHOLE PETE	D	TD	0029
PRINTLINER	T	TD	0071
PRINTLINER	D	TD	0072
QOGO	D	TD	0030
REVEAL	D	D	0040
ROLLA BEARING		T	0042
SMG	D	D	0047
SNAPPO	T	TD	0024
SNAPPO	D	TD	0025
SNOWBALL	T	D	0031
SNOWBALL	D	D	0032
SOUL OF A ROBOT	T	TD	0063
SOUL OF A ROBOT	D	TD	0064
STAR COMMAND	D	TD	0033
SUPA CODER		T	0069
SURFACE SCANNER	T	D	0008
SURFACE SCANNER	D	D	0009
TARGET ZONE	D	TD	0035
THE CAVES OF ORB		T	0080
TIME BANDITS	T	D	0051
TIME BANDITS	D	D	0052
TOADO	D	TD	0036
TOUCH TYPIST	T	TD	0059
TOUCH TYPIST	D	TD	0060
TOURNAMENT SNOOKER			0037
TRAMP	T	D	0058
TURBO	D	D	0037
USER EXTEND	T	T	0076
USER UTILITIES	T	T	0081

Software Item

Dest. Sorce Code

26 X 26 SPREADSHEET	T	T	0075
3D TACHYON FIGHTER	D	TD	0034
ADVANCED TOUCH TYPIST	T	TD	0061
ADVANCED TOUCH TYPIST	D	TD	0062
AGROVATOR	T	D	0049
AGROVATOR	D	D	0050
ALICE IN WONDERLAND	D	TD	0001
ARCAZIONS	D	TD	0070
ASTROPAC	D	D	0002
ATTACK OF THE KILLER TOMATOES	T	TD	0040
ATTACK OF THE KILLER TOMATOES	D	TD	0041
BACKGAMMON	D	D	0003
BLOBBO	D	TD	0004
BRIDGE	D	D	0053
CASSETTE INLAY	T	D	0091
CASSETTE INLAY	D	D	0092
CHES	D	D	0005
COBRA	D	D	0006
COBRA	T	D	0007
CONTINENTAL INVADERS	D	TD	0016
CRIBBAGE	T	D	0054
CRIBBAGE	D	D	0055
D			
DOWNSTREAM DANGER	D	D	0048
DRAUGHTS	D	TD	0010
DRIVE THE CEE-5	T	TD	0083
DRIVE THE CEE-5	D	TD	0084
EDASM	T	T	0082
EXTENDED BASIC	T	T	0079
FIG FORTH	T	D	0065
FLUMMOX	T	D	0043
FLUMMOX	D	D	0044
GOLDMINE	T	TD	0011
GOLDMINE	D	TD	0012
HAWKWARS	T	TD	0013
HAWKWARS	D	TD	0014
HIGHWAY ENCOUNTER	T	TD	0038
HIGHWAY ENCOUNTER	D	TD	0039
HUNCHY	D	TD	0015
JUMPING JACK FLASH	T	TD	0067
JUMPING JACK FLASH	D	TD	0067
KILOPEDE	D	TD	0017
KNUCKLES	D	D	0018
MAXIMA	T	D	0019
MAXIMA	D	D	0020
MEMOSKETCH			0088
MINER DICK	D	TD	0045
MTX CARDBOX	D	TD	0074
MTX DISASSEMBLER	T	TD	0077
MTX DISASSEMBLER	D	TD	0078
MTX NODDY TUTOR	T	D	0089
MTX NODDY TUTOR	D	D	0090
MUSICPAD	T	D	0056
MUSICPAD	D	D	0057

Memotech MTX World

Issue 2

bottom address of the code is 100h, just where a CP/M .COM file would originate. This makes the CP/M owner's life easier for the conversion so I'll discuss that first.

Get a 59K system with NO startup parameters, e.g CONFIG etc, containing DDT.COM and MTX.COM

Boot from above disc and load MTX.COM

Load Highway from Tape, and check it works OK.

Reset the machine, re-booting from from disc.

At CP/M prompt enter SAVE 160 HIGH.COM DON'T TRY TO EXECUTE IT YET!!

Select ROM 5 and then re-load HIGHWAY from tape.

When "Please Wait" message appears, press BRK, pause the tape, delete lines after 10 and then save the loader program.

Enter ASS.10 and change the DI at 4058h to RET.

Re-run the program and when the message appears restart the tape to load the 1st block of code into the memory. This will stop with a "No Line" error message 3580. Press ENT/CLS to clear the line.

Enter USER WRITE "HIGH1.DAT",16640,13184

Reset machine and re-boot from disc.

Enter DDT HIGH.COM should result in something like: NEXT A100 PC 0100

Enter IH1.COM to select 1st block of code and then R to load it.

Enter S4000 and the next 8 byte need resetting as below:-

35 FF E0 61 1F FD FF and . to complete.

Exit DDT and then SAVE 160 HIGHWAY.COM

Set the disc to read only and then try the program, just treating it as a CP/M program. If all is well and good after a shortist load period the title screen should appear on the TV screen (not the 80 column one!). If not, check the modifications to the main file were made, and if so try re-building the 1st block again from tape as this is most likely the problem. Setting the disc to r/o is just safety as I lost all the data when a corrupt version ran riot on the disc.

Once a working version exists then it can be put on any disc and only HIGHWAY.COM is needed.

The H1.COM is because the system startup corrupts the area around 100h, with H1.COM holding the data required and this is overlaid on the maincode via DDT.

The 8 bytes from 4000h to 4007 are set 08 - 01 at every reset and thus need replacing with the correct code from the 2nd block.

SDX users.

Unfortunately, your job isn't as easy! Firstly start from typing ROM 3 or ROM 5 depending on your system and follow the instructions down to "Enter USER WRITE". Reload the loader program from disc.

Modify as follows to enable 2nd block of code to be loaded:

```
4055      JR SKIP
4073 SKIP: LD HL,#4480
407C      RET
```

Which will just have the code to load the 2nd block active.

Run and when the message appears, restart the tape at the start of the 2nd block. Once complete, it should drop back to the "Ready" prompt.

Enter USER WRITE
"HIGH2.DAT",17536.27520

Now with the code data saved a short loader routine need writing to replace the tape loader from disc.

Included is the loader written for the disc system. The choice of ROM is left to you to implement, just multiply the rom number by 16 and put its hex value in 40BBh (simply add trailing zero to number.e.g. ROM 5=50h).

80h doesn't select ROM 8 as it may first look, but selects RAM mode, ROM page 0, RAM page 0. This is because of the way the page selection addressing works. The byte is used as below:

Bit 7 6 5 4 3 2 1 0

RAM / | ROM | |--- RAM ---|

ROM

RAM/ROM 0 = ROM based

1 = RAM based.

Listing:-

```
a) Lines 5 & 6 as per tape loader.
7 USER READ "HIGH1.DAT",16640
10 CODE
4054 DI
4055 LD A,(#FAD2)
405B LD (#D800),A
405B LD A,#80
405D OUT (0),A
405F LD HL,#4100
4062 LD DE,#100
4065 LD BC,#3480
4068 LDIR
406A LD A,(#D800)
406D OUT (0),A
406F EI
4070 RET
11 USER LOAD "H4.BAS"
20 USER SAVE "HIGHWAY.BAS"
30 RUN
```

about a fortnight for the software to arrive, slightly longer for overseas members.

If there is going to be a delay in getting the software to you, we will be in touch.

On the next two pages is the print out from the dBase file which contains all the details about the software we have. I will explain how to read the listing.

The name of the software is listed down the left hand side, and the format onto which we can copy it is listed under Dest (for Destination). Sroe is just to tell us what copies we have. The really important bit it is Code. The code number is the hub to the piece of software you want.

Let's go through a few examples of ordering from this list. Say we wanted to order Maxima on 3.5" disc. Look down the list for Maxima and look across to the column with Dest. Look for the D, meaning disc and write down the corresponding code, in this case, 0020. Send us your name, address, membership number, order code and cheque and we'll sort it out.

Let's try ordering Surface Scanner on cassette. Look through the list for Surface Scanner, then for T in Dest, write down the code, 0008.

Finally, lets order MusicPad for disc. Look for Music Pad, check that its for disc, and write down the code, 0057.

The reason for the list being printed in this manner means that the list that actually makes the journal will be right up to date.

Some items don't have any Dest entry. This is because we have the software, but haven't figured out how to copy it yet.

I hope everyone can follow this, if you need any help, drop me a line of give me a call.

We can copy software onto any disc capacity and 3.5" or 5.25" and Non-CP/M as well as CP/M systems.



Important Announcement

The objective of The Software Source, is to be as its name suggests, a source of software. In order to live up to this, and in order to expand what we can do for the members, we have expanded the list of commercial software which we are selling. In the list of software following, there is a wide selection of the old software which used to be available from Orion Software before they went down the tubes.

It is not our intention to break copyright agreements and in order to ensure that the author receives his rightful royalty payments, each item of software you purchase from us will be noted and when and if the author comes to us, we will be able to pay him for the software that we have sold.

Most of the software listed here is available from MOC although some of it is not - those being the software which is officially licensed to The Software Source for which the authors will receive a better royalty rate since their software is not (or shouldn't be) on sale anywhere else.

We feel this is the only way to proceed because it is a ludicrous position to be in that after nearly five years there is a wealth of good

software which is locked up.

Our list is pretty limited at present, this is due to the fact that what is listed is what Andy Fox and I have been able to dig up.

I quite expect you to have items of software which are not listed here and we would be grateful for a loan of it to allow your fellow members to benefit.

As a thanks for loaning us your software, you'll get an item of commercial software back, of your choice. So it's well worth your while.

Although Andy Fox is running the commercial software, it is my experience that too many addresses in the journal lead to confusion. In addition to this, Andy will be moving soon and we don't want your orders going astray. All orders should be sent to me, Alan Hamilton, at the normal address and should include the following: Your Software Source membership number, the format you wish the programs on (3.5", 5.25" disc or cassette), your cheque or postal order and your address.

There will probably be a slight delay since I will send off the orders received in one week in a batch so, as stated in "The Guide", allow

Then save this program and an auto-run version. A second program to load the second block is needed as below:

11 USER READ"HIGH2.DAT",17536

20 CODE

401F LD HL,START

4022 LD DE,#D000

4025 LD BC,#100

4028 LDIR

402A JP #D000

402D START: DI

402E LD A,(#FAD2)

4031 LD (#D800),A

4034 LD A,#80

4036 OUT (0),A

4038 LD HL,#4480

403B LD DE,#3480

403E LD BC,#6B80

4041 LDIR

4043 JP #100

4046 RET

30 USER SAVE "H4.BAS"

40 RUN

Again save this and an auto-run version.

By reloading the auto-run of program 1 should activate the game. If it fails and the programs are OK, then if the screen message is messed up the

likely cause is the 2nd block of datam otherwise its the code block loaded first.

Anyone wishing help with this can contact me and I'll see if I can help. I've converted all the commercial programs I've got for the disc now and am prepared to try any others if anyone is stuck.

A similar technique can be applied to Attack of The Killer Tomatoes, but only 1 file is needed making life simpler.

Also if anyone is interested, I've completed an index of all the articles etc published in MemoPad and will be prepared to pass this on to any interested parties. It's currently in Volume Issue order, but hopefully I'll translate it in the near future. Also if anyone wants copies of articles I'll send copies, cost of copying and postage only. If the owners of the copyright objects, please contact me and then an outstanding order can be sorted out.

Andy Fox,

8 Winston Gardens,

Headlingley,

Leeds, LS6 3LA

0532 751053 eves and weekends

Dear Ed...

Dear Ed

...On the software side, have you any idea where I might be able to get hold of some of the Infocom programs such as *The Hitch Hiker's Guide To The Galaxy* and also which ones were released for the MTX. I'm also interested to get hold of Return To Eden (Level 9) if anyone's got it.

Has anyone ever put any thought into utilising the excess RAM many of us have above 64K? What can realistically be done with it for both the standard and CP/M systems?

After reading past issues of Memo-Pad it prompted me to thinking about the graphics capabilities of the machine. I was wondering if anyone had replaced the VDP chip with a better version, added an improved board to the MTX or harnessed more of the capabilities of the 80 column board. The articles mentioned a new series of VDP chip to replace the existing one doubling the resolution and giving other gains to the screen display. If you're out there - please let someone know!

Andy Fox

Ed-> The Infocom adventures were all released on CP/M and so there isn't any problem about running them on a Memotech. The only problem is actually getting hold of them! After speaking to Graham Bettany, it appears that the Einstein Users have had The Hitch Hikers Guide and other adventures like the Zork series for a while and all it appears we have to do is transfer them onto the CP/M via an RS232 link which could be done in about ten minutes. The biggest problem is that we are technically breaking the law because we are copying the software but leave it with me and I'll see what we can do.

There are some MTX'ers about with different VDP chips than the TMS9938 which is the first of the generation of Texas Instruments VDP chips. I believe the ones they have installed are the ones used in the MSX 2 which gives them a distinct advantage over everyone else because they wouldn't need to have software converted very much from the MSX for it to run. Can anyone help?

It is technically possible to piggy-back more RAM onto the 80 column

board and the graphics resolution expands dramatically. I remember speaking to someone on this aspect a while ago who had actually done it. The figure of 640x640 resolution sticks in my mind which is what the Atari ST can do in hi-res mode only! Whoever said 8-bit machines were dead?

Dear Ed...

...At some stage I want to find someone who can produce a double density A4 graphics dump...Can anyone tell me how to use SPK\$ properly?

Nigel Cooper

Ed-> Any volunteers?

Dear Ed...

I have a query regarding CPM25 as described on page 2 of Issue 1: you say, "This requires MBASIC and TVI". I have Mallard BASIC running on the FDX and this is compatible with MBASIC, but what is TVI?

Paul Jenkins

Ed-> If Mallard BASIC is indeed compatible with MBASIC and can read MBASIC files, then there should be nothing to stop you from running the MBASIC programs under Mallard. TVI is the Televideo Emulator program which you **SHOULD** have on your system disc. It is documented in the CP/M section of that enormous manual. I have heard though of

people not getting it, even though it is documented. If this is the case, let me know and I'll get a copy of it to you. The TVI emulator is to allow us Memotechers to use the programs that have been written using the Televideo Terminal with all the differences in screen controls that there are. Televideo is a popular terminal type and I suppose we are lucky to have the best of both worlds.

Dear Ed...

In your article on BASIC, you talked about LEFT\$, RIGHT\$ and MID\$. Did you know that there is another way to split strings? The format is A\$(s,l) where s is the start position along the string and l is the number of characters to extract. Try this example:

```
10 LET A$="123456789"
20 LET B$=ABCDEFGH
30 LET C$=A$(4,3)+B$(3,5)
40 PRINT C$
50 PRINT C$(2,4)
```

John Hodgson

Ed-> Well, well, there we are! I knew about the ability to extract single characters with A\$(n) but not the A\$(s,l) way. Thanks, John!