

VOL 4 ISSUE 3

JANUARY 1988

THE
MEMOTECH OWNERS CLUB
MAGAZINE
MEMOTECHNIQUES



PUBLISHED BY: MEMOTECH OWNERS CLUB
13 COPSE ROAD
TOWNHILL PARK
SOUTHAMPTON

CIRCA . . . 335

M.O.C.

VOLUME 4 ISSUE NUMBER 3

CONTENTS

1. Editorial
2. FLITTER PART V
3. "
4. Using RAND
5. " "
6. A Full Software List
7. HARDWARE PROBLEMS
8. FLITTER PART V
9. Your Letters
10. Continued/Advert
11. MTX CP/M
12. MTX CP/M
13. Alans Comments

o o o o o o o o o
--- A Club Facility ---

A program listing facility for those with no printer. Just send in your program on tape (or 5.25" disc) with a pre-paid envelope addressed to yourselves and the club will do the listing for you.

--- Names and Telephone Numbers. ---

i. Paul Wood for 3.5" disc copying, general info and Comms specific info.

Tel 0905 24260

ii. Alan Dobson for help with the following adventures:
Alice, The ZOD and Man From Granny

Tel 061-980-6288

Phil & Hazel Eyres
13 Copse Road
Townhill Park
Southampton

In the next week or so we will have available Program Library and Commercial Software Review booklets, which will hopefully look fairly professional and contain useful information for the budding software buyer.

Ready for the new year, the club has purchased a limited quantity of 1988 Diaries. Because they have the club logo (in gold!) on the front cover they will obviously be the 'in thing' to have. The diaries are 'month page' and have several pages of useful figures at the front. At only £1.75 each (P&P 25p) they are an ideal New Years Gift. Send off soon for them as they will be sent out on a first come first served basis.

I should now be in most evenings, except for the odd game of squash, so I think that it would be best to revert to having the club Hotline between 7 and 8 pm any evening. Please where at all possible refrain from phoning after 9pm. I hope this is ok for everyone. The number to phone now is (0703) 585106, ask for Phil.

If anyone would like back issues they are available for the small remittance of 80p each. At present there are 32 back issues, 10 for volume 1, 10 for volume 2, 10 for volume 3 and 2 for volume 4.

It should be noted that all articles are the copyright of the sender and M.O.C., anyone wishing to have articles published elsewhere should inform us first.

ooo 0-0-0 ooo

INTERFACING PROJECTS

Why not make your new years resolution one which will lead you into the exciting world of micro electronics. Infact what better way to start than with an MOC D.I.V. kit. Everything you need is supplied, except a soldering iron, wire cutters and of course a few hours of your time!! So why not order now.

Interface price list

A full set of components and instructions for the LED kit -->£6.95
A full set of components and instructions for the Speech Synthesiser kit -->£18.00
Connecting cable for the internal port (needed for projects) -->£4.50
All prices are fully inclusive. Please allow 14 days for delivery and make cheques payable to MOC.

COMPUTER WORDSEARCH

The program prints out a 20 by 20 letter square. Hidden are 18 computer terms, can you find them?
(ClueP.190-198)

```

20 VS 4: CLS
30 DIM A(20,20)
40 FOR T=1 TO 20: FOR W=1 TO 20
50 READ A(T,W)
60 CSR W+5,T: PRINT CHR$(A(T,W)+65);
70 NEXT : NEXT
80 GOTO 80
90 DATA 1,0,20,15,0,18,11,1,8,19,2,0,7,20,16,18,14,13
    ,16,14,4,16,12,4,12
100 DATA 14,17,24,6,17,4,18,0,10,14,15,4,4,10,0,12,18
    ,8,9,14,25,6,9,7,18
110 DATA 20,21,1,8,13,0,17,6,7,18,20,10,2,20,17,18,14
    ,18,2,0,14,7,0,15,20
120 DATA 25,24,17,2,8,11,14,17,19,7,4,0,11,14,6,8,13
    ,19,24,8,12,18,14,0,13
130 DATA 0,9,14,1,4,17,4,6,8,5,1,20,23,14,20,23,21,0
    ,17,19,19,4,15,14,2
140 DATA 21,19,24,16,20,19,22,3,3,20,1,4,19,4,4,14,7
    ,17,0,14,23,8,7,12,20
150 DATA 7,22,9,20,14,2,16,11,6,17,20,22,14,21,18,16
    ,23,18,14,7,0,8,0,23,4
160 DATA 7,1,0,8,5,19,10,2,7,8,13,14,24,23,9,19,14
    ,22,17,18,8,20,16,18,0
170 DATA 15,17,4,18,18,0,2,13,0,18,7,8,13,1,4,0,5,23
    ,19,2,20,19,18,0,2
180 DATA 1,24,19,13,24,0,17,17,0,24,20,5,4,18,4,13
    ,16,18,20,20,6,11,0,25,10
190 DATA 18,2,8,14,19,2,4,24,7,2,3,9,14,6,13,22,0,23
    ,14,24,18,4,15,8,6
200 DATA 14,11,8,10,4,4,2,17,18,8,11,11,24,2,7,4,13
    ,4,1,0,11,19,7,14,9
210 DATA 4,24,14,13,1,19,9,4,18,1,12,19,13,1,24,19,0
    ,10,1,15,3,15,14,10,4
220 DATA 13,18,7,8,15,1,0,18,8,18,8,14,12,14,13,3,0
    ,25,24,8,21,7,2,3,0
230 DATA 11,4,5,18,14,14,13,17,8,19,12,24,19,14,9,14
    ,10,4,0,17,4,6,19,7,13
240 DATA 18,19,7,25,10,14,18,7,0,13,4,17,0,22,3,17,0
    ,7,10,14,0,9,14,21,22
    
```

A 1988 MOC diary for the first 5 correct answers!!!

FLITTER PART 5

By

Brian Clarke

This months apology is for the basic lines which were not printed in last months update. I have passed a repeat copy of them to Phil; they are printed below, sorry if this has caused any problems (Phil).

This months update will cover the remainder of the main programme loop, which will enable the game to be run in machine code until such time as one or both ships are destroyed.

The first routine is SCRNUF, together with VALPOS and SETZS, which is BASIC line 520. The reasoning is that, if both ships are in the central area (i.e. not subject to increased or decreased thrust), the 'Z' values should average out to the mean of the Z-value range, whilst maintaining their difference. The Z-range is from 32 to 250, thus the mean is 141. If the Z values are 207 & 247, they should gradually reset to 121 & 161 respectively. This can be done as in the BASIC line 520. (Or should I say could have been done, had the 288 in line 520 been 292. Never mind, I'll stay with 288 for now.)

However, single registers can only store values 0-255, so I am using the combined registers HL and DE to hold values of up to 65535. To check the result, I test H (the MSB of the result). If it is >50 then HL is -ve, if it is 0 then test L. If it also is 0 then the result is zero, otherwise the result is +ve. Thus we add to both LZ and RZ +1 or -1 (=1 less than 0 = 255 for our single register maths), or skip this operation altogether.

SCRNUFC + SCRNUF2. The first sets the sprite colours of sprites 4 & 5 to 0 (transparent) if the Z difference is greater than 40. (Sprites 4 & 5 are the moving sprites either side of the main viewing screen). If however the difference is less than 40, the sprite Y positions and colours are set, the sprite Y position = 2 * Z-value + offset.

SCRNUF3 calculates RZ/32 and LZ/32 for the pattern numbers of sprites 2 and SPLNPAT, where SPLNPAT = 1 if the Right ship is furthest away, or 3 if the Left ship is furthest away (hence keeping the appearance of ships passing 'behind' each other).

DIV32 is the operation of division. It is performed by subtracting 32 from the value, adding one to another register, (the counter), and comparing the result with 32. If the remainder is greater than 32, then repeat the subtraction. If the remainder is less than 32, then 32 cannot be taken from it, so the counter tells you the

result of X/32. Note that this is an INTEGER, thus a decimal result of 2.999 will give a result of 2, as will a decimal result 2.001.

If this accuracy is not good enough for you, I am sure you can improve the programme such as:

```

from CP D          to CP D
    JP NC,DIV32     JP NC,DIV32
    LD A,B          CP 16
    RET            JP NC,ZZZZ
                  INC B
                  ZZZZ: LD A,B
                  RET
    
```

thus adding 1 to the result if the remainder is greater than half of 32.

SCRNUF1 sets various sprite variables to their required values from our list of variables. It also performs the task of calculating the variables for the power bars (ref BASIC lines 460 & 440) and drawing the shot availabilities. Most of this was covered in FLITTER PART 2, however the shot availabilities requires a little more explanation. If we just consider the right player, which begins at label SHOTY.

```

SHOTY: RST 10      sets the m/c into graphics mode.
        DB 101      access VS 5.
        DB 131,3,2,22  csr 2,22; last RST 10 command.
        LD A,(L6)    obvious.
        CP 4         compare L6 to 4.
        JP NC,SHOTZ  jump to label SHOTZ if L6 >=4
        CALL NILSHOT gosub NILSHOT as no shots avail.
        JP SCREEN    continue with programme.
SHOTZ:  CP 8         compare L6 to 8.
        CALL C,ONESHOT gosub ONESHOT if L6<8 (and >=4).
        CALL NC,TWOSHOT or TWOSHOT if L6>=8.
NILSHOT:RST 10     graphics again.
        DB 132," "   draw 4 spaces from current cursor
                    position. This is an extension to
                    DB 131,3,2,22 could have been
                    written DB 135,3,2,22," " - i.e.
                    DB (131+number of characters to be
                    printed),3,csrX,csrY,"(characters)",
                    but we have our CSR pos'n preset,
                    thus is DB (128+no. chrs),"(chrs)".
    
```

And I am sure you can now understand the ONESHOT and TWOSHOT subroutines. Note that I have printed 4 spaces for NILSHOT, this was purely to overcome a problem I had when setting up the programme. Having got over the problem, I did not reset this value.

From the SCREEN up to the ENDDGAME label, this is where the SPRITE (begining DB 170), power bars (LLINE & RLINE beginning DB165 & DB132) plus screen attributes (DB 164) are held. As I stated in FLITTER PART 2, the sprite X and Y positions are held in double registers. (Remember sprite plane is +/-4095 see the VIEW BASIC command in the manual). The single register position we are calculating is inserted into the register which is actually labeled - see below. This is OK, as the X & Y positions are calculated as (LSB/MSB) as is most 2-register data, and the labelled register is the LSB part.

I.E. looking at register 4517 onwards (Sprite SPLN)

4517	DB 170,18	register 4517 holds value 170	451B = 18
4519	SPLNNO:DB 0		4519 = 0
		but will be loaded with sprite number during execution.	
451A	SPLNFAT:DB 0		451A = 0
		will be loaded with pattern no.	
451B	SPLNXPOS:DB 0,0		451B = 0
		will be loaded with X pos'n.	
		This is the LSB for X-pos'n.	
			451C = 0
		Will not change (MSB stays 0)	
451D	SPLNYPOS:DB 0,0,0,0,12		451D = 0
		will be loaded Y pos'n (LSB)	
			451E = 0
		Y pos'n MSB stays at 0	
			451F = 0
		XS stays at 0	
			4520 = 0
		YS stays at 0	
			4521 = 12
		colour is always 12	
4522	DB 170,18	next sprite	

The ENDDGAME routine sends you round the loop again if

- 1/ Left Power > 0
- 2/ Right Power > 0
- 3/ DIFFX,Y & Z greater than predefined limits, otherwise goto QUIT, which returns you to basic. (i.e. one or both ships are destroyed). Use BASIC peeks to read power and proximity levels, and continue as before.

You can now play the game a little faster than you could in basic, but the logic and programme flow are not as efficient as they could be. The next issue will start again from the beginning, showing a flow chart of operations which is aimed at speeding up the programme; this will be achieved in two ways.

A/ performing minimum tasks required to meet our objectives

B/ changing the logic, thus it will not be as the basic programme.

I hope that by now you will have understood some of the principles behind machine code, and that the programme itself holds no mysteries. A reference book that I would have found useful, had I not loaned it out (and forgotten that I had done so) is R. A. & J. W. PENFOLD's 'An Introduction to Z80 Machine Code', published by BERNARD BABANI (BP152) at (the then) price of £2.25. This also has some routines for the NTX machines.

Finally, one thing I should have suggested at the beginning. If there is a routine that you do not understand the operation of, or are curious about, use the PANEL facility. Set the particular routine up as a code line 0 (i.e. ASS.0), enter the code, enter PANEL. As the code is at line 0, the operation starts at £4007, so a) L4007<ret> will list the routine. (L<ret> continues the listing); D4007 will display the HEX code from 4007, but don't forget to press <brk>. Press the full stop (<.> to move register pointer to PC (programme counter), press <R>4007<ret> to set counter up to 4007, then press <S> to step through the programme one command at a time. You can see what all the registers are doing as the programme progresses. A full description of PANEL facilities is too great to go into here, I can only suggest you give it a try. There is a little information in the manual, page 161, but not very much. All the best for now.

```

30 RETURN
100 POKE (16493),LZ
110 POKE (16496),RZ
120 GOSUB 20
200 LET LG=PEEK(16471)
210 LET LP=PEEK(16472)
220 LET RG=PEEK(16474)
230 LET RP=PEEK(16475)
240 LET LX=PEEK(16491)
250 LET LY=PEEK(16492)
260 LET LZ=PEEK(16493)
270 LET RX=PEEK(16494)
280 LET RY=PEEK(16495)
290 LET RZ=PEEK(16496)
300 LET SPSN=PEEK(16497)
310 LET SPLN=PEEK(16498)
410 REM SCRNUP - SCREEN UPDATE
420 SPRITE 5,10,32,(RZ+RZ-LZ-LZ+95),0,0,6:
    SPRITE 4,10,224,(LZ+LZ-RZ-RZ+95),0,0,12
    BASIC Lines which should have been printed last month.

```

This months listing is on page 8.

The use of RAND

By

Mike Kohnstamm and John Green

Following Arthur Wingroves letter on random numbers, two short articles were sent in:-

Mike Kohnstamm;

After some investigation I think I now understand how the MTX random number works.

RAND just sets the seed in RNSEED (see p180 of the manual), ie ROM location $\$FD7E-\$FD7F$ (64895-6 decimal). Try this program to see how this seed is set:

```
10 INPUT "ENTER ANY NUMBER FOR SEED ";A
20 RAND A
30 PRINT "RNSEED SET TO ";PEEK(64896)*256+PEEK(64895)
```

RND takes the number in RNSEED and produces another number from it - the same seed always produces the same number. What is actually produced is an integer between 0 and 65536, which is then divided by 65536 to give RND, so for any seed, $RND*65536$ is always an integer. Add this line to demonstrate this:

```
40 PRINT "RND * 65536 = ";RND * 65536
```

This alone would be of little use as RAND would have to be used to set a new RNSEED every time a random number was needed, so RND also resets the seed every time it is used. The seed is actually set to that number from 0 - 65536 that it first came up with - ie to $RND * 65536$. Now add this line:

```
50 PRINT "NEW RNSEED SET TO ";PEEK(64896)*256+PEEK(64895)
```

For example, if you enter 1000, then $RND * 65536$ will always be 27249 (RND is 0.4157). The new value of RNSEED becomes 27249, ready for RND to generate a new number next time.

You don't actually need to do a RAND at all, since by default, when you switch on the computer, the seed is set to zero and you get the same as a RAND 0, which is as good as any other RAND to start off the sequence.

All this is only true for RAND from 0 to 65536. Numbers higher than this create as seed of 16 (or 17, 18 etc if very much higher). More interesting, as spotted by Arthur, is what happens if you have a negative RAND number. What seems to happen is that instead of the normal process, the machine reads the refresh register of the Z80 chip, and

uses this to set the seed. So if you really do not want to be able to repeat the sequence of random numbers, use a negative RAND to set a random seed. Use the PEEK's above to find out what the seed is actually set to.

On the subject of whether all MTX's come up with the same sequence of random numbers, it is clear that they do, since they are all programmed the same way in the ROM. (My MTX500 and 512 certainly do). The Sinclair Spectrum, although it works on very similar lines, generates a different series of numbers. In case anyone is wondering, the "Z-loader" Spectrum simulator, behaves like a Spectrum, not an MTX.

For those that like exploring the ROM, the RND process is at $\$2048-\$208E$ and RAND is at $\$2C08-\$2C10$, both on page 0.

John Green;

Random numbers or more properly pseudo-random numbers are generated using a method called linear congruence which is basically a simple formula where the next number in a sequence is derived from the previous one. The formula is:

$X(N+1) = \text{the fractional part of } (A*X(N)+B)/C$

where $X(N)$ is the Nth member of the series and A, B and C are the constants. In a basic program this would appear as

```
10 LET X=(MOD(A*X+B),C)/C
```

If the value of C is carefully picked the series of numbers produced will repeat after C iterations.

Let's see if we can find the constants for the MTX. To find the value of C enter the following program.

```
5 LET X = RND
10 FOR N = 1 TO 100000
15 LET Y = RND : IF Y = X THEN GOTO 50
20 NEXT N
50 PRINT N
```

The number 65536 will be printed in just under 5 minutes showing that the MTX will produce that number of pseudo-random numbers before repeating.

Since the initial seed on switch on is 0, we can now find the value of the constant B.

$X(1) = \text{the fractional part of } (A*X(0)+B)/65536$

Continued From Previous Page

But $X(1) = 6.25610352E-04$ so we get from simple algebra since $X(0)=0$ $B = 65536 * 6.25610352E-04 = 41$

Using the second value of RND we can also find the constant A which turns out to have the daunting value of 96272384.

If the following program is now typed in it will produce the same sequence of numbers as the RND function.

```

5 LET A=96272384: LET B=41: LET C=65536
10 LET X=0
15 RAND 0: REM TO START TWO SETS AT SAME POINT
20 FOR N=1 TO 20
25 LET X=(MOD(A*X+B,C))/C
30 PRINT X,RND
35 NEXT N

```

Since $96272384 = 1469 * 65536$ we can make the following changes to the program.

```

5 LET A=1469: LET B=41: LET C=65536
25 LET X=A*X+B/C-INT(A*X+B/C)

```

The constant 1469 provides the different starting points when a positive number is used after RAND, try the following:-

```

5 LET A=1469: LET B=41: LET C=65536
10 INPUT "RAND VARIABLE ";R
15 LET X=R
20 RAND R
25 LET X=A*X+(B+R*A)/C-INT(A*X+(B+R*A)/C)
30 PRINT X,RND
35 FOR N=1 TO 19
40 LET X=A*X+B/C-INT(A*X+B/C)
45 PRINT X,RND
50 NEXT N

```

Note: Since $65536 = 64k$ it would require the entire memory to store the random numbers in a look-up table. The numbers produced by this method should be uniformly distributed about 0.5. A much more complicated formula is required if we require a normal distribution.

:*:~*:~*:~*:

"I just love it when a plan comes together!"

The A-Team...

:*:~*:~*:~*:

MEMOTECH OWNERS CLUB PROGRAM LIBRARY

This is the list of disc's three and four + cassette programs. The costs are: £1.20 for two programs on cassette (we supply the tape) and £2.50 on disc for 20 programs (or £3.50 if you want us to supply the disc). When ordering for discs please state clearly what memory capacity you work on (100K to 1Meg).

Disc 3 starts here.

41. Reversi 2	Assembler version of 24.
42. ISDT	A really good maze game.
43. DBase	Simple database.
44. Diary	Diary & Addresses Program.
45. Terminal	Comm's via RS232 & modem.
46. Skittles	Keeps league tables.
47. Card-Ind	Produces card indexes.
48. 2 * H&W	Large character printing.
49. Hangman	A classic
50. Account	Third accounts package.
51. Mastermind	Another good game.
52. Connect 4	Two player game.
53. Journey Into Danger	NEW adventure game.
54. Connect 4 V2	As for 52.
55. Patience	Card Game.
56. Life	Odd Puzzle.
57. Enigma	Like Mastermind.
58. FKEY	Function key definer.
59. Skydiver	Graphical game.
60. Digger	RockFall or Repton type game.

Disc 4 starts here.

61. MP6	Calculates fuel consumption.
62. Spooler	Dumps Panel & VS 4 to printer.
63. Labels	Label printing program.
64. Ski Version 2	Update of published SKI program.
65. PNT/BJCK	Card game compendium
66. Biorythms	Day forecaster

Cassette ONLY Software

CA01. Renum	Renumbers BASIC programs.
CA02. Merge	Merges BASIC programs.
CA03. Money Manager	Home Accounts package.
CA04. FKEY	Defines Function Keys.
CA05. DBase III	Advanced database program.
CA06. Filetech	Great database program.
CA07. Flight	Brilliant Flight Simulator.
CA08. RamDisc	Better than sliced bread.
CA09. TextEd	Assembler Word Proc.
CA10. Deci_Clock	Brilliant Clock program.

Don't forget the compilation disc at £1!

All cheques made payable to MOC please.

Alan Hamilton, Program Library, Memotech Owners Club,
12 Roebank Road, Beith, Ayrshire, KA15 2DX.

Telephone: 05055 2491

HARDWARE AND SOFTWARE PRICE LIST

JANUARY 1987

MOC

Phil Eyres

13 COPSE ROAD, TOWNHILL PARK, SOUTHAMPTON. Tel 0703 585106

Below is the list of available software for the MTX series, titles held in stock will be dispatched by return, all other titles ordered immediately and sent by return when received. All MOC titles always in stock and sent by return. Please make cheques payable to Memotech Owners Club.

Title	Price	Title	Price	Title	Price
3D TACHYON FIGHTER	7.70	GRAPHICS	6.60	RUTHLESS B.	4.00
AGROVATOR	6.60	HELI-MATHS	8.30	SALES LEDGER	17.50
ALICE	7.70	HIGHWAY ENCOUNTER	8.80	SALTY SAM	6.60
ASTROMILLON	7.70	HUNCHY	6.60	SEPULCRI SCELERATI	7.70
ASTROPAC	7.70	ICEBURG	6.60	SMG	7.70
ATTACK OF KILLER TOMATOES	8.80	JUMPING JACK FLASH	6.60	SNAPPD	7.70
BLOBBO	7.70	KARATE KING	7.70	T. SNOOKER	8.80
BOUNCING BILL	6.60	KILOPEDE	7.70	SON OF PETE	7.70
BRIDGE	7.70	KNUCKLES	8.80	SUPA CODER	8.80
CAVES OF DRB	6.60	LITTLE DEVILS	6.60	SUPER BIKE	6.60
CHAMBEROIDS	7.70	MISSILE COMMAND & ARCADE	6.60	SUPER MINEFIELD	7.70
CHESS	10.00	MATHS 1	10.00	SURFACE SCANNER	7.70
COMBAT	4.40	MAXIMA	7.70	TAPEWORM	7.70
CRYSTAL	7.70	MEMOCHEQUE	7.70	TARGET ZONE	7.70
DISASM	8.80	MEMOSKETCH	8.80	THE WALL	6.60
DOODLEBUG	6.60	MEMOSKETCH SDX	10.00	TOADD	7.70
DOWNSTREAM DANGER	7.70	MISSION ALPHATRON	6.60	TURBO	7.70
DR FRANKIE	6.60	MISSION OMEGA	6.60	USER BASIC	10.00
DRIVE THE CEE-5	7.70	NEMO	7.70	USER BASIC SDX	11.00
EDASH	8.80	OBLITERATION ZONE	7.70	UTILITIES SDX	11.00
EMERALD ISLE	7.70	OBLOIDS	7.70	WORD & PICTURE	10.00
ESCAPE FROM ZARKOS	7.70	PAINTBOX	6.60		
EXTENDED BASIC	8.30	PAYROLL	23.61		
F1 SIMULATOR	5.50	PHAID	7.70		
FATHOMS DEEP	7.70	PONTOON & BLACKJACK	7.70		
FIG FORTH	17.50	POT HOLE PETE	7.70		
FIG FORTH SDX	17.50	PURCHASE LEDGER	14.00		
FIREHOUSE FREDDIE	7.70	QOGO 2	7.70		
FIRST LETTERS 1	10.00	QUANTUM	6.60		
FLUMMOX	7.70	QUAZZIA	7.70		
GHOSTLY CASTLE	3.80	REVERSI	8.80		
GOLDMINE	7.70	ROLLA BEARING	7.70		

MOC TITLES

REVEAL	6.00
SMG2	6.00
PUC-MAN	5.00
MOC FIG-FORTH	7.00
" " " TECH DATA	2.00
" " " TUTORIAL	7.00

We can offer DMX 80 printer ribbons for only £7.00 each, so why not order one today and be prepared for the day your ribbon finally 'bites the dust'!!!

The MTX FIG-FORTH requires an MTX512 or expanded 500, the dictionary associated with Forth is held as part of the Ram-Disc which can be saved separately, fairly quickly. The Ram-Disc allows for 24 'edit' screens to be created and in memory simultaneously. A tutorial will be necessary for the beginner, for this the club has obtained a quantity of the publication Fundamental Forth (This may

vary according to availability).

- Fig-Forth Program £6.00
- Tech Data Sheets £2.00
- Tutorial Book £9.50 (240 pages)

QUOTATIONS

"Come and have a look at this!!, ...the computer is malfunctioning again!"

Hardware Problems

By Geoffrey Gardiner

I had hardly finished my recent article on Imeg drives for the FDx, in which I mentioned their probable reliability, when I started having trouble. Drive C first became temperamental, and then ceased to work at all. I am an inveterate fiddler and could not resist the temptation to try to find the fault which was obviously a loose connection which I might be able to find. But there is a sacred law about loose connections; it says that to find one loose connection you precipitate two more, and that is what happened - and more. By the time I had finished I had to cure loose connections in the 60 way connector between the MTX and FDx (a regular source of troubles), and in the power connection. The latter had two connections which must have been making only intermittent contact for ages, leading to erratic startup.

The fault with the drive prevented it from switching on. The sensor that detects the little hole in the disc was not working and this stopped the solenoid from operating. I had assumed that the solenoid draws the head onto the disc. In fact it withdraws a cam that prevents a spring from pressing the head onto the disc when the head is not reading or writing. As my solenoid was not getting any power the head was kept out of close contact with the disc.

To find possible loose connections required a lot of dismantling and inadvertently I loosened the two screws that hold the head in alignment. A phone call to my son who is an electronics engineer produced a recommendation to seek professional help as I would need an alignment disc and an oscilloscope. More helpfully he told me how I could clean up contacts. A lot of chips plug into sockets and he told me that these are a fruitful source of problems and he prefers not to use them in his designs. Easing the chips loose with a screwdriver and pressing them tight again is usually enough, or the contacts can be cleaned with cleaning fluid though this must not be allowed to get on the chip capsules for fear of contaminating the chips themselves. He advised against use of abrasives such as my favourite 400 grade wet or dry paper. Especially gold plating must not be abraded. I did a general clean up which probably improved the working of the outfit.

But I mulled over that alignment problem and decided to try an old trick. The required tools were a steel rule marked in 64ths of an inch, a "lupe", which is an 8x magnifier, an old disc, a sticky label, and a piece of carbon paper. I put the label on the disc where it is exposed, I placed the disc in the sound drive, pushed the head to its outer location point, and carefully inserted a sliver of carbon paper under the head, carbon side against

the disc, and pressed the head down. While it was down I slid the carbon paper sideways. The result was an impression of the head on the white label. Then I did the same on the faulty drive and compared the distance between the edge of the central hole of each disc and the impression left by the carbon paper. I needed to work to 100th of an inch and when I magnified the 64ths markings on the steel rule it was not difficult to estimate even more closely than a quarter of a division so there was no problem with accuracy. Remarkably quickly I had readjusted the head on the faulty drive to tally with the good drive's head alignment and tightened the retaining screws.

I still had to find the loose connection on the drive. It had to be on the circuit to the sensor. But that was on a PCB hidden under the motor. After much dismantling I had the PCB out and examined every connection with the lupe. And there it was, in the place expected, a hairline crack in between the solder and a pin. I resoldered the connection and a lot of others for good luck. I checked the alignment of the sensor as I had disturbed it. Then I reassembled and switched on. Yes, there is a happy ending for it worked immediately. Who knows it might even continue to do so. So far there has been no problem in reading or writing to disc. Startup is now immediate and everything seems to work much faster. Those notorious "bad sector" messages have been conspicuous by their absence.

ASSEMBLY
LANGUAGE
COURSE

NORMAL R.R.P.
£10.00

SPECIAL CLUB
PRICE

ONLY £7.00

QUOTATIONS

"To detect errors, the programmer must have a conniving mind, one that delights in uncovering flaws where beauty and perfection were once thought to lie."

FLITTER PART IV - CONTINUED

10 60TD 900	442B	LD (SP5YPOS),A	44B2	CP 4		
20 CODE	442B	LD A,95	44B4	JP NC,SHOTY	4550	DB 101
4010 DS 250	442D	SUB B	44B7	CALL NILSHOT	4551	DB 165,2,4,40,4
410A DS 250	442E	LD (SP4YPOS),A	44BA	JP SHOTY	4556	RLINE1: DB 0
4204 DS 250	4431	LD A,6	44BD SHOTX: CP 8	4557	DB 165,2,3,40,3	
42FE DS 184	4433	LD (SP5COL),A	44BF	CALL C,ONESHOT	455C	RLINE2: DB 0
43B6 SCRNUF: LD A,(RY)	4436	LD A,12	44C2	CALL NC,TWOSHOT	455D	DB 164,27,65,2,1
43B9 CP 56	443B	LD (SP4COL),A	44C5 SHOTY: RST 10	4562	DB 165,2,4	
43BB JP C,SCRNUFC	443B SCRNUF3: LD B,0	44C6	DB 101	4565	RLINE3: DB 0,4,191	
43BE CP 136	443D	LD D,32	44C7	DB 131,3,2,22	4568	DB 165,2,3
43C0 JP NC,SCRNUFC	443F	LD A,(RZ)	44C8	LD A,(RB)	456B	RLINE4: DB 0,3,191
43C3 LD A,(LY)	4442	CALL DIV32	44CE	CP 4	456E	DB 132,27,65,2,0
43C6 CP 56	4445	LD (SP2PAT),A	44D0	JP NC,SHOTZ	4573	NOP
43C8 JP C,SCRNUFC	444B	LD B,0	44D3	CALL NILSHOT	4574	ENDGAME:NOP
43CB CP 136	444A	LD D,32	44D6	JP SCREEN	4575	LD A,(LP)
43CD JP NC,SCRNUFC	444C	LD A,(LZ)	44D9 SHOTZ: CP 8	4578	CP 0	
43D0 LD HL,288	444F	CALL DIV32	44DB	CALL C,ONESHOT	457A	JP Z,QUIT
43D3 LD DE,0	4452	LD (SPLNPAT),A	44DE	CALL NC,TWOSHOT	457D	LD A,(RP)
43D6 LD A,(RZ)	4455	JP SCRNUF1	44E1	JP SCREEN	4580	CP 0
43D9 LD E,A	445B DIV32: SUB D	44E4	NILSHOT:RST 10	4582	JP Z,QUIT	
43DA SBC HL,DE	4459	INC B	44E5	DB 132," "	4585	LD A,(DIFFZ)
43DC LD DE,0	445A	CP D	44EA	RET	4588	CP 2
43DF LD A,(LZ)	445B	JP NC,DIV32	44EB	ONESHOT:RST 10	458A	JP NC,KEYS
43E2 LD E,A	445E	LD A,B	44EC	DB 130," "	458D	LD A,(DIFFX)
43E3 SBC HL,DE	445F	RET	44EF	RET	4590	CP 4
43E5 LD A,H	4460 SCRNUF1: LD A,(RX)	44F0	TWOSHOT:RST 10	4592	JP NC,KEYS	
43E6 CP 50	4463	LD (SP2XPOS),A	44F1	DB 130,"**"	4595	LD A,(DIFFY)
43E8 JP C,VALPOS	4466	LD A,(RY)	44F4	RET	4598	CP 2
43EB LD A,255	4469	LD (SP2YPOS),A	44F5	SCREEN: RST 10	459A	JP NC,KEYS
43ED JP SETZS	446C	LD A,(LY)	44F6	DB 170,18,5,10,32,0	459D	QUIT: RET
43FO VALPOS: LD A,L	446F	LD (SPLNIPPOS),A	44FC	SP5YPOS:DB 0,0,0,0		
43F1 CP 0	4472	LD A,(LY)	4500	SP5COL: DB 0	30	RETURN
43F3 JP Z,SCRNUFC	4475	LD (SPLNYPOS),A	4501	DB 170,18,4,10,224,0	100	REM
43F6 LD A,I	4478	LD A,(SPLN)	4507	SP4YPOS:DB 0,0,0,0	120	GOSUB 20
43FB SETZS: LD B,A	447B	LD (SPLNNO),A	450B	SP4COL: DB 0	210	LET LP=PEEK(16472)
43F9 LD A,(LZ)	447E	LD A,(SPSN)	450C	DB 170,18,2	230	LET RP=PEEK(16475)
43FC ADD A,B	4481	LD (SPSNNO),A	450F	SP2PAT: DB 0	240	LET DIFFX=PEEK(16477)
43FD LD (LZ),A	4484	LD A,40	4510	SP2XPOS:DB 0,0	250	LET DIFFY=PEEK(16478)
4400 LD A,(RZ)	4486	LD B,A	4512	SP2YPOS:DB 0,0,0,0,6	260	LET DIFFZ=PEEK(16479)
4403 ADD A,B	4487	LD A,(LP)	4517	DB 170,18	510	IF DIFFZ<2 AND DIFFX<4
4404 LD (RZ),A	448A	ADD A,B	4519	SPLNNO: DB 0		AND DIFFY<2 THEN GOTO 850
4407 SCRNUFC: LD A,(DIFFZ)	448B	LD (LLINE1),A	451A	SPLNPAT:DB 0		
440A CP 48	448E	LD (LLINE2),A	451B	SPLNXPOS:DB 0,0		
440C JP C,SCRNUF2	4491	INC A	451D	SPLNYPOS:DB 0,0,0,0,12		
440F LD A,0	4492	LD (LLINE3),A	4522	DB 170,18		
4411 LD (SP5COL),A	4495	LD (LLINE4),A	4524	SPSNNO: DB 0,10,120,0,1,0,0,0,0		
4414 LD (SP4COL),A	4498	LD A,(RP)	452D	DB 100		
4417 JP SCRNUF3	449B	ADD A,B	452E	DB 165,2,61,40,61		
441A SCRNUF2: LD A,(LZ)	449C	LD (RLINE1),A	4533	LLINE1: DB 0		
441D ADD A,A	449F	LD (RLINE2),A	4534	DB 165,2,60,40,60		
441E LD B,A	44A2	INC A	4539	LLINE2: DB 0		
441F LD A,(RZ)	44A3	LD (RLINE3),A	453A	DB 164,27,65,2,1		
4422 ADD A,A	44A6	LD (RLINE4),A	453F	DB 165,2,60		
4423 SUB B	44A9	RST 10	4542	LLINE3: DB 0,60,191		
4424 LD B,A	44AA	DB 100	4545	DB 165,2,61		
4425 LD A,95	44AB	DB 131,3,5,22	4548	LLINE4: DB 0,61,191		
4427 ADD A,B	44AF	LD A,(LQ)	454B	DB 164,27,65,2,0		

YOUR LETTERS

Help Lines

1. Has anyone got a merge routine for a MTX500 as I would find this very useful.

K.Green 131 Southpark Ave, Norwich, NR4 7AZ

2. Can someone please advise me on how to use the EOF marker when using Disc basic.

Mr G.Price Westpontnewydd, Gwent

Ed->The half a dozen pages in the FDZ manual concerned with the use of disc commands are really useful and easy to understand!! ..That is providing you are a world leader in solving cryptic clues, the syntax for the EOF command is as follows:-

```
DISC EOF f1,(Line Number)
```

e.g.

```
DISC EOF f1,100
```

This will cause program execution to jump to line 100 when EOF is reached.

3. I recently bought two games cassettes from UK Home Computers, when I tried to load them the computer put up on the screen FOUND "" and LOADING "" but when the sounds finished it did not print LOADED. Can you think of any reason that it does not load on my MTX 512.

Colin Cockram.

Ed->Trouble with loading cassettes can be because of one of two things, assuming you have tried all usual volume settings:-

1. The cassette deck, If the playback head on your cassette deck is dirty or it is in a different position to that of the deck used to record the tape, the tape, although good, may not load.

2. The tape is faulty, due to low quality tape copying or some fault incurred through postage.

The tape deck theory can be tried out by using another deck if possible, if the tape still insists on not loading the tape is likely to be faulty.

If the program appears to start to load properly but then just hyperspaces, in my opinion the tape is faulty. Remember though, you should also have the program recorded on the other side of the tape, so try that one.

4. Peter Burns has some information on Ed/Asm but, like Alan Ayre last month is still unable to get Ed/Asm to merge with Basic. This is what Peter has to offer:-

So far I've:-

1. Assembled and loaded code using EDASM option ^M into a LOAD area of memory. The program is ORG'd to 4010 and all code is then relative to a 4010 start.

2. ASSEM 10 then DS'd enough memory to fit the program;

3. Entered PANEL and Moved the code to address 4010;

4. NOP'd the C9 (end of ASSEM 10 I assume) & up to 4009; and

5. then attempted to run it, the first line is there but the listing shows the DS 254's still there and only the first line of EDASM code in place.

Ed->Peter, if it is any help at all, the RAMDISC program from the program library was written in EDASM and when merged with Basic works like this:-

The code looks like this:-

```
LD HL, CODE
```

```
LD DE, £ED00
```

```
LD BC, £0500
```

```
LDIR
```

```
RET
```

```
CODE: DS 250
```

```
DS 250
```

```
DS 250
```

```
DS 250
```

```
DS 250
```

```
DS 30
```

Then the Basic starts here. It would appear that the EDASM code is loaded at CODE, then it is moved to £ED00. The Basic has setup the USER command to access the code at £ED00 (60672 dec), so you could at least call it using RAND USR(60672) which would have the same effect but not allow you to pass variables.

5. Please can you help with some information and advice:-

1. I have a Panasonic KX-P1081, the printer cable connector at the computer end lies very close to the metal casing, initially when setting up I obtained garbage print out, but by inserting a thin strip of plastic card alongside the lower side of the connector plug the problem was cured, wondered if this was a common problem, and if there was a better solution?

2. The silicon disc's look very interesting, can they be battery backed?, also is it possible to make a backup copy to tape?

3. I gather that there is a possibility that there will be a teletext adaptor available shortly, do you have any details?

Ed-> I have heard of printer cable problems before, but I would not say that it is a common problem, it would appear that the plastic is just shielding the rogue interference, so anything that doesn't look to unsightly should do. Just

as a 'by-the-by' for anyone that does not know, the KX-P1081 printer is the DMX 80 printer that Memotech have re badged.

The silicon disc's are very useful, especially if you work extensively under CP/M, so if you work with Newword, DBASEII, Supercalc etc, running these from silicon disc is about the same as working on a PC with programs that are completely memory resident. (Very Fast!!). Their failing is that you lose all when you switch off, I have heard of members successfully adding battery backing although I do not have details.

The tape backup of disc's (silicon or otherwise) seems like a really useful thing to have, what about a little bit of assembler to run under CP/M to do this!??, a really interesting project for our more budding assembler experts, especially after Dave Dulsons recent articles on assembler under CP/M. The club will offer a piece of free software for the best effort.

Sorry, but I do not have any details about a teletext adaptor, I will try and find something out.

For Sale

£32K memory expansion for the MTX500 available for £25, if anyone is interested, please contact:-

Mr K.Joyner 5 Keats Close,High Wycombe, Bucks.HP11 2UJ

Tel 0494 23457.

 Upgraded To DISC ?
 Fed up with waiting for games to LOAD ?
 AFW Software are proud to announce the launch of:-

**MTX Tape to Disc
 Conversion Booklet.**

This major work (37 A4 pages) allows the majority of commercially available game software to be converted to any Memotech disc format. Included are nine worked examples: Sepulcri,Quazzia,Toado, Murder at the Manor etc.

The documentation clearly shows the techniques used in the medium conversion. The booklet also discusses: PANEL (includes utility), Memory, also System variables and the workings of the Basic Interpreter.

Send a cheque or postal order for £5.
 Payable to AFW Software.
 20 Cambridge Road, Whitehaven Cumbria.

ROM CALLS

ADDRESS	REG'S AT ENTRY	REG'S AT EXIT	COMMENTS
£1B	None	None	Prints chr\$(13) & chr\$(10)
£30	DE points to table	BC=result	This is the Basic calculation routine. It calc's a formula stored at DE. The result is put in the BC and the table must end with £FF.
eg.	LD DE,TAB RST 30 CALL £1B50 RET		(See below for codes!!)
TAB:	DB "123",£DF,"75",£D1,"A",£FF		This will print a char to the screen
£BC	A=Char	None	This will set the Z flag if either of the shift keys are pressed
£AD	None	Z flag=true	This will select ROM page 1
£CD	None	None	Basic continue
£250	None	None	Print a string (end with £FF)
£2E7	HL points to table	None	Selects ROM page X
£64A	A = page	None	This will reset the Z flag if the BRK key is pressed
£9F2	None	Z flag=not true	Print a char, then behave like DSI
£1424	A=char	None	This routine will wait for a key to be pressed.
£1B23	None	A=char	Prints BC in Hex
£1B50	BC=Number	None	Prints A in Hex
£1B55	A=Number	None	Prints low nibble in Hex
£1B5C	A=Number	None	This routine will read a line of text from the keyboard and then store it in memory.
£1B87	E=Max length	DE=start HL=end+1 C=Length	

CP/M ON THE MEMOTECH

By

Alan Hamilton

From what I can gather from running the Program Library, there is a fair bit of confusion over CP/M, the availability of software for it and just exactly what it is, how much it is, and where you can get it. Firstly though, let's look at why we should even bother about CP/M.

One of the main facets the MTX can boast is the upgradability of the basic machine into a computer with mammoth power. With CP/M comes the disc drive. You might be very impressed about the reliability and speed of the Memotech when it runs on tapes, but when you connect a disc drive to it, to quote a figure of 9600 baud standard read/write transfer and a screen update of 25,000 baud (on average, generally much faster) probably would not mean much. To say that the disc drives increase the speed of loading and saving three fold and to list a BASIC program without stopping would mean that the listing would move so fast up the screen that it is absolutely impossible to read, probably would though.

That's fine, CP/M is fast, what else? It gives you much more control over the processor than you might otherwise have had. There is purpose written software for running the RS232s. Various programs allowing the examination of the memory, programs to copy programs, to define the function keys and on and on. All these come on your system disc!

When the MTX was brought out, it was decided that CP/M Version 2.2 was the operating system to go for because of the vast amount of software that you could get (well over 1000 titles in all) and the portability of programs between other CP/M Versions. Just as an aside for a min, I have a friend who has a Research Machine 380Z which uses CP/M Version 1.4 (a genuine relic!) and believe it or not, I got all of his software to work on my FDX!

CP/M is really nothing other than a medium to which software can be written and be distributed to many computers without much conversion. Compare this to the many different dialects of BASIC there are nowadays.

CP/M comes on the Memotech in three guises:

- (1) The old FDX system
- (2) The not-quite-as-old SDX system
- (3) The brand-spanking new 3.5" Series 2 version

Taking them in turn, the FDX was (and still is) considered to be the flagship of the MTX range. It can do just about anything you want. It can handle up to four disc drives whether they be 3.5", 5.25" or 8", it makes not a jot of difference. In addition to this, it can handle up to a further four Silicon or RAM Discs which run like lightning, cutting down access time to an instant. The FDX bares the marks of being an old system. It's got a

cooling fan at the back which runs all the time and compared to newer systems is very large. In saying this, it is an absolute joy to use. One snag - you'll be very lucky to get one now as they stopped when Memotech Limited stopped and there never were great numbers of them about.

On to the SDX. The Single Disc system was designed to be used by the upmarket home user or small business user. This is reflected in the amount of expandability it offers. You cannot have any more than two 5.25" or 3.5" drives and there is no provision for Silicon Discs. The edge connectors of the MTX are both used up. One for the SDX Controller and the other for the 80 column card (which is essential for CP/M - there will be a further note on this later on though!) which means that if you've got any little hardware projects or Pascal ROMs, you've got to disconnect the Controller and 80 col. board to use them. One major disadvantage I've had from experience (I'd like someone to prove me wrong) is that the SDX is incapable of copying discs from one memory capacity upwards or downwards to another capacity. The FDX, as far as I can tell is the only one capable of doing that. You can still get the SDX systems. One well known supplier in Swindon has them.

Last but not least is the new system. It has many of the idiosyncrasies of its brother the SDX. The 80 column card still fits onto the internal expansion port and the controller to the external one. However, the new system does have a Silicon Disc and comes complete in one box which houses the 3.5" disc drive, controller and Silicon Disc in one. It's a lot neater than the SDX with no ribbon cables trailing across a table to the disc drive. In addition to this, the new system is very very quiet and pretty fast being newer technology. You can get this system from MCL themselves or through MOC.

When you switch on any of the above systems, or, for that matter, any CP/M system, the computer is absolutely dumb. All it can do is ask you to insert a disc with CP/M on it. When you stick a disc in the disc drive, the CP/M program (or SYSTEM) is read off it and suddenly the computer becomes intelligent. This procedure is commonly known as Booting the System because the computer has to pull itself up by the bootstraps, in a manner of speaking!

CP/M is not very user friendly. All you get when your system is booted is a A) and a cursor flashing beside it. When you type something on the keyboard and press return, the operating system goes through a small procedure to see if what you've typed in is the name of a program, or more correctly, a .COM file (you'll learn more about this in a minute) which is to be loaded. If it finds that your input is not the name of a file on the disc, it prints on

the screen what you typed in followed by a question mark. See what I mean by not very user friendly!

I mentioned .COM files. In CP/M, most files have a filename of 12 characters. The first eight characters being the actual name of the file and the last four being composed of a dot (.) and what is called an EXTENSION. To illustrate this, take this example: if a file you had on the disc was called MEMOTECH.BAS, this would tell the operating system that the file was called MEMOTECH and was of the type BAS. The dot merely separates the name from the type. In general, extensions or types are only there for three purposes:

- (1) To let the operator know what a file is used with,
- (2) To let CP/M know which files can be directly run from the A> prompt (.COM files) and
- (3) To allow programs to generate files of specific kinds under one filename.

The original CP/M systems were developed around the 8080 processor, but since the assembly mnemonics of 8080 and Z80 are so closely related, it didn't take long for the systems to come out onto Z80 based computers, just like the MTX!

The operating system regards all .COM files as programs which have been written in 8080 code so that it can readily deal with it - you wouldn't expect the bare MTX in BASIC to understand a Pascal program would you?

OK, so that's enough about what CP/M does and how it works. What about the price? Well, unfortunately this is the bad bit. CP/M comes expensive. You'll need to lay out about £300 for the older SDX system and even more for the new Series 2 system. You're going to need a monitor too. The 80 column board functions only on a composite or RGB output and doesn't have a modulator to configure it to a television input. However, there is an exception to the rule: If you read Issue 1 Volume 4 of the MOC mag, towards the back, you'll see a review by myself about a system which does not need a monitor! But other than that, you've got to bear in mind that you get what you pay for and I feel that it is worth every penny!

Supa Coder Review
By Alan Hamilton

For those familiar with the speed of assembler and the lack of speed of BASIC, the idea of using the BASIC on the MTX with the speed of Assembler must surely be a tempting choice.

Supa Coder, by JC Hodgson and DC Threlfall is essentially a compiler. That is, it takes the BASIC program currently in memory and converts it in to machine code. To run the compiled version, enter RAND USR and the memory location it was compiled at and hey presto, one super fast program.

Brill! you might think, "I can speed up all the BASIC programs I've got". Well, I suppose that could be said to be true but you better be patient for Supa Coder has quite a few limitations.

The one I found worse was the fact it can only take single dimension arrays. That means that unless your DIM statement merely defines the length of a variable, such as, DIM A\$(1000) then it won't work. Another glaring absentee is the fact that you can't use STEP in a FOR - NEXT loop. Admittedly, the instructions do show you how to get round this.

Speaking of instructions, the ones that come with it are fair enough, mostly just listing the commands it DOES take, taking care not to mention those that it doesn't!

I had a trial run of Supa Coder with one of the sort programs that were listed in issue 3 and I was very impressed by the vast increase in speed and decrease in execution time. For instance, the BASIC version of the program took 21 seconds to run, the compiled version ran in just about 7 secs. One test I ran resulted in the clock still displaying 000000.

Supa Coder isn't much use at maths neither. It can only take a divide sign outside of brackets which means that you can't have the formula to convert degrees to radians ($\text{Angle} \times \pi / 180$) in a program without a lot of very complicated variable swapping. Also, it doesn't seem to be able to handle the function VAL. So you can't convert a string into a number without using the laborious and clumsy ASC function.

So, to sum up then, if you have a pretty simple routine that you are likely to use a lot in a program and need the speed then I think you'll find Supa Coder a bit too limited for much practical use. But, in saying that, it gives an excellent example of how slow BASIC is compared to machine code and I think if there was a Supa Coder 2 brought out which could handle absolutely everything, it would sell like hotcakes!

Price: £5.95

Value for money:	60%	Ease of use:	80%
Practical use:			50%
Total:			55%

PROGRAM LIBRARY
12 Roebank Road,
BEITH
Ayrshire
KA15 2DX
(05055) 2491

Christmas and New Year are out of the way and judging by the amount of mail delivered through my door the Special Offer was a resounding success. It's a pity it's got to stop now, sorry, but the prices are back to £1.20 for two programs on tape and £2.50 for 20 programs on disc. If you want us to supply the disc, the price is £3.50. All cheques payable to MDC please.

More additions to the library, on the CP/M front and to the main library, courtesy of Paul Wood:

CPM5 Small C compiler.

This as far as I can tell is an excellent implementation of the language imaginatively called "C". As it stands it has most of the features of the more expensive compilers. Well worth a look at.

CPM6 Comms Disc.

This is a disc from Paul Wood which has got everything you can imagine to allow you to use your MTX from Electronic Mail to Bread Board Services and so on (except the modem of course!!). Ensure about 300K free on the disc please!

Main Library

65 - PNT/BJCK

This is a marvellous program which shows off the Memotech's graphics to the full. The program is really two programs in one. One is a Pontoon game (21's) and the other BlackJack. One of the best in the library!

66 - BIORHYTHMS

Ever felt down? Physically exhausted? Intellectually stagnant? Well, chances are that you could have predicted it with this program. It comes with a remarkable array of features and is very impressive with nice graphs and explanations.

Other than that, not a lot else to say except that a big thanks must go out to all that sent the library mail to me, I hope you haven't been waiting too long for the programs, the sheer volume of mail coming in has slowed my "system" down quite a bit.

Don't forget that I am always looking for more programs to add to the library. Anything at all really that you think other members might like to see, any format or memory size. Personally, I would like to see more software for disc users and perhaps more that uses the 80 column card. I'll leave it up to yourselves!

One last point - the library is available from me on cassette and on 5.25" discs. I can handle any memory capacity from 100K (type 00) to 1 Megabyte (type 07). Paul Wood is in charge of all the 3.5" copies of the library. If you've got an order for the library on 3.5" disc, please send it to Paul, his address is listed below. Ta!

Paul Wood,
12 Bishops Avenue,
WORCESTER
Worcs
WR3 8XA

All the best for the coming year! - Alan Hamilton

This file was downloaded from

www.primrosebank.net

Abridged Terms & Conditions (Downloads)

Disclaimer

www.primrosebank.net, (*the website*) is provided by Dave Stevenson as a service to the public, is provided "as is" and carries no warranties, expressed or implied, of any kind.

Dave Stevenson is not responsible for, and expressly disclaims all liability for, damages of any kind arising out of use, reference to, or reliance on any information contained within the website or made available for download. Whilst the information contained within the website site is periodically updated, no guarantee is given that the information provided on the website is correct, complete, and up-to-date.

A number of articles on the website contain technical data and practical guidance which may be of use in testing and maintaining various items of vintage computer and electronics hardware. Such articles are not intended to cover all aspects of the tasks involved and may omit essential information, including necessary safety precautions. Performance of the tasks described may risk damage to equipment and/or people. The reader is responsible for ensuring that he/she is capable of performing the tasks described and well as assessing the inherent risks involved and taking appropriate measures to mitigate such risks.

Dave Stevenson expressly disclaims all liability for, damages to equipment or injury of any kind arising out of use of such technical data and guidance.

Unless otherwise noted, all data on the website is deemed to be *Copyright (c) Dave Stevenson, 2009-2014*

You are hereby granted permission to download data and software from the website *for your own personal use. Redistribution of any content from the website without written authorisation from Dave Stevenson is expressly forbidden. You are also expressly forbidden from offering for sale any material obtained from the website.*

As far as possible, information included on the website from other sources has been credited to the respective author and/or publisher. The majority of content on the website is derived from material first published in the 1980s. *This material is likely still under copyright of the original author and/or publishers.* The authors and/or publishers may not have given express permission to copy, transmit or make this information available for download, but I believe that they would have no objection to this archive information being placed into the public domain.

However, should the author and/or publisher of the original material find any content on the website for which they wish to assert their rights, they should notify Dave Stevenson (by e-mail to: webmaster@primrosebank.net) who would be pleased to enter into a dialogue to agree a satisfactory resolution of their concerns.

If you obtained this file as part of a paid-for package, you have been scammed! I suggest that you request a refund from the seller, please also advise Dave Stevenson at the e-mail address above.