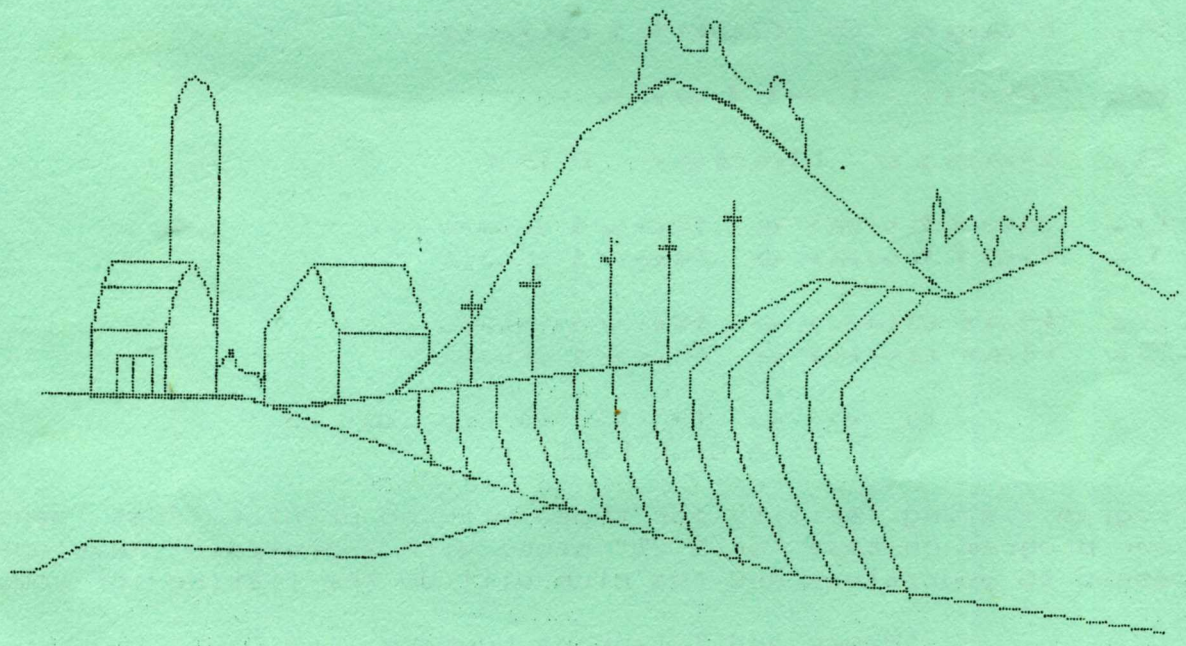
VOL 3 ISSUE 10 SEPTEMBER 1987

THE MEMOTECH OWNERS CLUB MAGAZINE

MEMOTECHNIQUES



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

CIRCA ...304

M.O.C.

VOLUME 3 ISSUE NUMBER 10

CONTENTS

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 ZOO and Man From Granny
Tel 061-980-6288

If anyone has any good graphics designs for a front cover then we would love to see them!!!

E D I T O R I A L (September 1987)

Phil Eyres
13 Copse Road
Townhill Park
Southampton

This magazine see's the end of the clubs third year, an initial look at the accounts shows that we are still viable, but that the percentage of funds left over after all normal costs is shrinking due to inflation. Roughly working it out, the £7.00 membership should be £8.10 taking into account three years of inflation. Although the club is not 'broke', I think to ensure that we are ok at the end of next year, membership should be increased to £8.00 to cover costs. I hope everyone thinks this to be fair. I will publish the year end accounts in the next issue.

Secondly, I think it fair to update everyone on the state of affairs concerning the new pro-activity I mentioned in the last magazine.

Alan Hamilton started a flood of enquires by getting our name in Popular Computing Weekly, this is boosting membership a bit at present, which is very good news.

We are in the throws of getting Alan started as Program Library administrator, he will hopefully be up and running next month. ...Look out for some special offers a bit nearer Christmas!!.

This month we have a lot more space taken up with Hardware and Software reviews, I hope to continue this to some degree, ensuring that we get one review every month as a minimum. We are also about to put together a booklet of reviews of software published so far.

I have had quite a few contibrutions for the magazine, in one form or another. A hearty thanks to everyone who has made the effort - Thanks Very Much!!. Please keep this going, it's nice to have to much to put in a magazine.

The winner of this months competition is Arthur Hills, see the Program Library next month, for his program, it is too long for the magazine, but well worth sending in for. Arthur will be receiving a copy of The Wall for his efforts, very shortly.

I can read/write only disc's in 5.25" format and up to 500K, if anyone with 3.5" systems would like something from the club or has something to offer on 3.5" format please send to Paul Wood, his address is listed opposite.

As I am more often than not out playing squash on Mondays between 6 and 7 pm, I feel it would be best to move the Hotline to after 7.40pm. I hope this is ok for everyone. The number to phone now is (0703) 585106, ask for Phil. However, feel free to phone any evening after 6pm.

If anyone would like back issues they are available for the small remittance of 80p each. At present there are 27 back issues, 10 for volume 1, 10 for volume 2 and 7 for volume 3.

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

Softwar e

Software prices for the best and most popular software:-

Zarkos	£7.00	Chamberoids	£7.00
Qogo2	£7.00	26*26 SpreadSht	£8.50
Karate King	£7.00	Son Of Pete	£7.00
S.M.G	£7.00	T.Snooker	£8.00512Only
Doodlebugs	£6.00	Super Bike	£6.00
J.J.Flash	£6.00	Ed/Asm	£8.50
Cee-5	£7.00	MTX Asm Lang Cse	£10.00
Highway Encounter	£8.50		

Some late news, Ron Gladwin of Uk Home Computers has reported to me that, he has heard that some people are selling MTX500's boxed and badged as 512's. This is obviously a serious problem, so be careful to check properly when you buy.

Ron has some 32K upgrade boards available at good prices, to anyone wishing to extend their machine a bit.

Ron's number is 0793 695034.

FLITTER

PART II

By Brian Clarke

Those of you who have been patient enough to type in and play the game will, by now, have found out that the weakest part of he programme (apart from the lack of speed) is the joystick input, lines 100-190

Try typing in the following :-

```
1 OUT (5),223:LET AA=INP(5):CSR 10,10:PRINT AA:GOTO 1
```

If you RUN this programme you will see 255 in the middle of your screen. If you press the <HOME> key, this will change to 127. It sometimes flashes back to 255. This is due to the System Interrupts, and need not concern us at the moment. If you now press the <A> key, the value changes to 254. Pressing both <A> and <HOME> gives the vaalue 126.

Working in binary, for an OUT (5) of 223, the INP(5) can be represented as :-

BIT	7	6	5	4	3	2	1	0
VAL	128	64	32	16	8	4	2	1
KEY	HOME	RET	:	L	J	G	D	A

In the MEMOTECH system, the status of a bit for keyboard input is a 1 if the key is NOT pressed, and a 0 if the it is. In MEMOTECH basic, a BOOLEAN comparison gives a 0 for false, or a -1 for true.

From the above table, if no key is pressed, all the bit's have the value 1, and the byte value is 255. Pressing <HOME> makes bit 7=0, the byte is 01111111 = 127 (255 - 'key val', i.e. 255 - 128 = 127). Similarly, pressing <A> = 255 - 1 = 254. Both <HOME> and <A> = 255 - 128 - 1 = 126.

Thus, where line 110 of he BASIC programme states

```
LET RF=-(AA=127)
```

If AA = 127 (i.e. right fire key pressed)

(AA=127) results in -1 (boolean true)

-(AA=127) gives -(-1), i.e. 1

and if AA is not 127, RF = -(0) = 0.

Thus RF is used as a toggle flag, to tell us if the 'Right Fire' key has been pressed.

To convert the keyboard input section of the programme into machine code, we will need to reserve 10 bytes for these keboard input flags. The sequence of the machine code will be:

1/ Jump the flags.
2/ Set all flags to 0, by first loading register a with 0 then load each flag in turn with register a.
3/ Dissable interrupts, send a value out of port 5, and read in values from ports 5 & 6. Enable interrupts.
4/ Check input for required value. If it matches, set respective flag to 1.
5/ Create an interface for the BASIC programme to read the reserved 10 keyboard input flags.

If you now reload the BASIC programme, if it is not already in, and type in the following programme lines, you should find that the programme is a little more reliable. The speed will not have increased yet, as we have not taken out much basic. Note that the programme is assembled using an MTX 500 lookalike: for MTX512's, the code lines will be assembled to 4010 to 40D3 instead of 8010 to 80D3, and the PEEK command need to be changed from 32788 - 32797 to 16404 - 16413 respectively.

The principle behind the BASIC part of this ammendment is that RF is set to the value of the keyboard input flag, as is LF. The X value is calculated as X + (move right) - (move left).

==========

Next month, I shall be looking at the graphics in machine code, using the RST 10 commands. The following is from an early copy of the MOC mag., and is the 'BIT FORMAT' for RST 10 COMMAND byte:-

```
7 6 5 4 3 2 1 0 = BIT
1 0 c <---n---> = STATUS
```

Explanation :-

Bit 7 must be set to 1, bit 6 must be set to 0

Bit 5 (c) is the continue flag, i.e. are there to be further graphics commands following the present one. If so, this Bit needs to be set to 1, if not, set this bit to 0.

Bits 4 to 0 contain the number of bytes in the command string.

Continued Overleaf

Continued From Page 2

The byte above is the COMMAND byte, and is the first data byte after the RST 10 command. The following data byte is the ASCII OPERATION byte, followed by the INFORMATION data byte(s).

AN EXAMPLE (or two).

BASIC command PLOT x,y.
ASCII code for PLOT is 1.
Lets make x=25, y=108.
COMMAND string: Assume this is the final graphic command, hence bit 5=0. Bits 4-0 depend on the number of ASCII OPERATION bytes (in this case 1 byte long) and the number of INFORMATION bytes (in this case 2, x and y). Hence n = 1 + 2 =3.

Thus the COMMAND byte is 1000011 =128 (bit 7 & 6) + bit 5 = 0 + 3 (string length) = 131
The machine code for this operation is then

```
RST 10
DB 131,1,25,108  (COMMAND, OPERATION and INFORMATION)
RET
```

A slightly more complex example, CRVS n,t,x,y,w,h,s.
Use n=4,t=1,x=5,y=0,w=22,h=24,s=32.

ASCII OPERATION for CRVS = 27,89.
OPERATION string = 2 bytes,
INFORMATION string = 7 bytes, total = 9 bytes.

Assume this is not the last RST 10 command, hence Bit 5 of command byte = 1. The value of the command byte is 128 + 32 + 9 = 169. Placing this in our previous example (PLOT x,y), we now have

```
RST 10
DB 169,27,89,4,1,5,0,22,24,32
DB 131,1,25,108
RET
```

RST 10 GRAPHIC COMMANDS

BASIC	COMMAND (final)	COMMAND (continue)	ASCII	INFORMATION
PLOT	131	163	1	x,y
LINE	133	165	2	x1,y1,x2,y2
CSR	131	163	3	x,y
CLS/HOME	129	161	12	----
CTLSPR	131	163	14	p,x
GENPAT	139	171	15	p,n,d0,d1,d2,d3, d4,d5,d6,d7
COLOUR	131	163	16	p,n
ADJSPR	132	164	17	p,n,v
SPRITE	138	170	18	n,p,xp(2 bytes), yp(2 bytes), xs,ys,col
MOVSPR	132	164	19	p,n,d
VIEW	131	163	20	dir,dis
ATTR	132	164	27,65	p,state
GR$	133	165	27,67	x,y,b
CRVS	136	168	27,89	n,t,x,w,h,s
VS	131	163	27,90	n

An alternative for accessing and clearing VS's is based on the following BIT format

```
7 6 5 4 3 2 1 0
0 1 c * X <-n->
```

Where c = continue, X clears the screen and n is screen number. * does not matter.

Thus the instruction is 64 + (32 to continue) + (8 to cls) + (screen no.).

That's all for now. Have fun.

Listing for FLITTER on Page 4

FLITTER - The Listing

```
10 GOTO 900
20 CODE
4010          JP KEYS
4013 TEMP:    DB 0
4014 LL:      DB 0
4015 LR:      DB 0
4016 LU:      DB 0
4017 LD:      DB 0
4018 LF:      DB 0
4019 RL:      DB 0
401A RR:      DB 0
401B RU:      DB 0
401C RD:      DB 0
401D RF:      DB 0
401E KEYS:    LD A,0
4020          LD (LL),A
4023          LD (LR),A
4026          LD (LU),A
4029          LD (LD),A
402C          LD (LF),A
402F          LD (RL),A
4032          LD (RR),A
4035          LD (RU),A
4038          LD (RD),A
403B          LD (RF),A
403E          LD A,127
4040          DI
4041          OUT (5),A
4043          IN A,(5)
4045          LD (TEMP),A
4048          IN A,(6)
404A          EI
404B          BIT 0,A
404D          JP NZ,TRYLL
4050          LD A,1
4052          LD (LF),A
4055 TRYLL:   LD A,(TEMP)
4058          BIT 0,A
405A          JP NZ,TRYLR
405D          LD A,1
405F          LD (LL),A
4062          LD A,(TEMP)
4065 TRYLR:   BIT 1,A
4067          JP NZ,TRYLU
406A          LD A,1
406C          LD (LR),A
406F          LD A,(TEMP)
4072 TRYLU:   BIT 2,A
4074          JP NZ,TRYLD
4077          LD A,1
              4079          LD (LU),A
407C          LD A,(TEMP)
407F TRYLD:   BIT 3,A
4081          JP NZ,RIGHT
4084          LD A,1
4086          LD (LD),A
4089 RIGHT:   LD A,223
408B          CALL SCAN
408E          JP NZ,TRYRL
4091          LD A,1
4093          LD (RF),A
4096 TRYRL:   LD A,247
4098          CALL SCAN
409B          JP NZ,TRYRR
409E          LD A,1
40A0          LD (RL),A
40A3 TRYRR:   LD A,239
40A5          CALL SCAN
40A8          JP NZ,TRYRU
40AB          LD A,1
40AD          LD (RR),A
40B0 TRYRU:   LD A,251
40B2          CALL SCAN
40B5          JP NZ,TRYRD
40B8          LD A,1
40BA          LD (RU),A
40BD TRYRD:   LD A,191
40BF          CALL SCAN
40C2          JP NZ,ENDKEY
40C5          LD A,1
40C7          LD (RD),A
40CA ENDKEY:  RET
40CB SCAN:    DI
40CC          OUT (5),A
40CE          IN A,(5)
40D0          EI
40D1          CP 127
40D3          RET

30 RETURN
100 REM KEYS
110 GOSUB 20
120 LET RF=PEEK(32797): REM MTX512 LET RF=PEEK(16413)
130 LET RX=RX+PEEK(32794)-PEEK(32793): REM MTX512 LET
RX=RX+PEEK(16410)-PEEK(16409)
140 LET RY=RY+PEEK(32795)-PEEK(32796): REM MTX512 LET
RY=RY+PEEK(16411)-PEEK(16412)
150 LET LF=PEEK(32792): REM MTX512 LET LF=PEEK(16408)
160 LET LX=LX+PEEK(32789)-PEEK(32788): REM MTX512 LET
LX=LX+PEEK(16405)-PEEK(16404)
170 LET LY=LY+PEEK(32790)-PEEK(32791): REM MTX512 LET
LY=LY+PEEK(16406)-PEEK(16407)
180 REM
```

ASSEMBLY LANGUAGE PROGRAMMINGWITH CP/M
PART II
By Dave Dulson

Before we can start writing a program we first have to know a little about CP/M.

The CP/M system has four functional modules, they are called CCP console command processor. Primarily a command interpreter, the intelligent part of the operating system.

Next is the BIOS basic input output system, this is in charge of communicating with various devices connected to the system.

Next is the BDOS basic disc operating system,this is in charge of managing the disc files.

Finally their is the TPA the transient program area, this holds programs that are loaded from the disc under the command of the CCP.

The available memory is also divided into zones, the top of memory is reserved for the CCP,BDOS and BIOS, the first 256 bytes of memory are reserved for the bootstrap loader and the system and between memory location 100 hex and the bottom of the CCP is where the TPA can be found. All executable programs must start at address 100 hex, also another useful address when program writing is 0005 hex this is an entry point address to the BDOS, and so is a gateway to the operating system.

The call to the operating system must be accompanied by a parameter held as a function number in the C register of the Z80.

There are 36 codes for CP/M 2.2 to access various I/O devices and the disc files and these are listed below.

FUNCTION NUMBER	HEX NUMBER	FUNCTION NAME
1	01	Console Input
2	02	Console output
3	03	Reader Input
4	04	Punch Output
5	05	List Output
6	06	Direct Console I/O
7	07	Get I/O Status
8	08	Set I/O Status
9	09	Print Buffer
10	0A	Read Console Buffer
11	0B	Interrogate Console Status
12	0C	Lift Disc Head
13	0D	Reset Disc System
14	0E	Select Disc
15	0F	Open File
16	10	Close File
17	11	Search for File
18	12	Search for next Occurrence
19	13	Delete File
20	14	Read Sequentially
21	15	Write Sequentially
22	16	Create New File
23	17	Rename File
24	18	Return log-in Vector
25	19	Return current Disc
26	1A	Set DMA address
27	1B	Get allocation Vector address
28	1C	Write protect Disc
29	1D	Get read only Vector
30	1E	Set File Attributes
31	1F	Get Disc Param-block address
32	20	Get or Set User Code
33	21	Read Randomly
34	22	Write randomly
35	23	Compute File Size
36	24	Set Random Record Position

HARDWARE AND SOFTWARE PRICE LIST

This month we have a special new offer, your favourite tape games, ...available on disc, compatible with all 500K CPM systems.

Disc One contains:-
- OBLOIDS
- QUASAR
- TOADO
- KNUCKLES
- TURBO

Disc Two contains:-
- MISSION ALPHATRON
- ASTROPAC
- STAR COMMAND
- 3D TACH FIGHTER
- NEMO

Disc Three contains:-
- STAR COMMANDER
- PACMAN
- MINEFIELD
- POTHOLE PETE
- PHAID
- CHAMBEROIDS

Disc Four contains:-
- 3D TACH FIGHTER
- MISSILE COMMAND
- TIME BANDITS
- BACKGAMMON
- ARCAZIONS

Disc's are available in 5.25" 500K format for only £8.00, or 3.5" 1Meg format for only £13.00.

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	£7.50 (240 pages)

Cheques payable to MOC please, orders from stock normally despatched by return, else, please allow 2 working weeks.

Ron Gladwin of UK Home Computers, (Tel 0793 695034) has on offer a Spectrum Loader that will convert your 512 into a 40K Spectrum, this will allow you to type in Spectrum programs, it is also said to load some Spectrum software. At £2.95 you may find it worth a try.

Ron also has some FDX Silicon disc's on offer, he has both 256K and 1Meg boards. The 1 Meg boards are only £100 and are available from us. Just think, you could be running your Supercalc or Newword at 3 to 4 times the speed you are used to. Two other goodies on offer from Uk Home Computers are MTX 512's for £43 and RS128's for £68. Please contact Ron directly.

All 'SUPER CHEAPIES' will be despatched by return of post.

!!!SUPER CHEAPIES!!!
(ONLY FROM STOCK)

DESC	QTY	PRICE (Each)	DESC	QTY	PRICE (Each)
			THE ZOO	3	£4.50
RETURN TO EDEN	1	£7.00	COBRA	1	£4.50
EMERALD ISLE	1	£7.00			
BLOBBO	5	£4.50	MINER DICK	1	£4.50
KILOPEDE	3	£4.50			
REVERSI	1	£4.50	HELI-MATHS	2	£4.00
MINEFIELD	3	£4.50			
FIRE HOUSE FREDD	2	£4.00			
TOADO	4	£3.50			
NEMO	1	£4.50	PONTOON &B'JACK	3	£4.50
SNAPPO	3	£4.50			
PAYROLL	1	£10.00			
PURCHASE LEDGER	1	£7.00			
PHYSICS 1	3	£5.50			
MATHS 1	1	£5.50	TAPEWORM	1	£4.50

Software prices for the best and most popular software:-

Zarkos	£7.00	Chamberoids	£7.00	
Qogo2	£7.00	26*26 SpreadSht	£8.50	
Karate King	£7.00	Son Of Pete	£7.00	
S.M.G	£7.00	T.Snooker	£8.00	512 Only
Doodlebugs	£6.00	Super Bike	£6.00	
J.J.Flash	£6.00	Ed/Asm	£8.50	
Cee-5	£7.00	MTX Asm Lang Cse	£7.00	
Highway Encounter	£8.50			

Order a copy today!!!!!!

Remember PRINTER RIBBONS are only £7.00
REVEAL is only £6.00
SMG II is only £6.00
PUC-MAN is only £5.00

The club has a special purchase offer, on The MTX ASSEMBLER LANGUAGE COURSE by John Grayson. We can supply this program at the NEW incredibly low price of Only £7.00.

Don't forget to order your copy of Fruit Machine from Graysoft today!!!!

Continued From Page 5

The best way to demonstrate the writing of assembly programs with CP/M, is to write a program as a text file using Newword and then assemble it, as every one has a 8080 assembler I shall use 8080 mnemonics in the demonstration program. The program is a simple delay program that prints on the screen when it start and again when it ends.

As it is a 8080 file it will require a file extension of ASM, if we were to use a Z80 assembler we would require a file extension of ZSM. It is important as the assembler can recognize the file type. When using Newword to produce the assembler program text file use the N option from the main menu this is the None Document type. If the D option is used or a Word Processor other than Newword then you will require the PIP command to reset the Parity flag set by the Word Processor. The program is called DELAY.ASM, and by using Newword in the None document mode create the file listed below.

```
;************************************
;*                                  *
;*      10 Sec DELAY ROUTINE        *
;*                                  *
;************************************

LF      EQU 0AH
CR      EQU 0DH
BDOS    EQU 5

        ORG 100H

        LXI D,TEXT1     ;print text routine
        MVI C,9         ;uses CP/M BDOS call
        PUSH B          ;at address 0005
        CALL BDOS
        POP B           ;preserves function reg

DELAY   LXI H,3000H     ;10 second delay
        MVI B,0
DLY     DCR B
        JNZ DLY
        DCX H
        MOV A,H
        ORA L
        JNZ DLY

        LXI D,TEXT2     ;print text for
        CALL BDOS       ;time out message
        JMP 0           ;return to CP/M

TEXT1   DB 1AH,'START OF DELAY ROUTINE $'
TEXT2   DB CR,LF,LF,'TIMED OUT $'
```

This process will produce two files.

DELAY.ASM and DELAY.BAK (8080)

The start address of the program must be 100 hex and must be specified at the beginning of the program with an ORG statement. Due to the nature of Newword and other Word Processors other than in None Document mode will set a parity flag when compiling their files and so this parity flag has to be set to zero before the program can be assembled.

The best way to do this is to use PIP.COM as PIP wont write a file of the same name to the same drive, you can either write to a different drive or do has I do and add a 1 to the end of the filename. To get PIP to reset the Parity flag you must use a letter Z in brackets at the end.

PIP A:DELAY1.ASM=A:DELAY.ASM[Z]

The next step is to assemble the program this will check for syntax errors only and does not mean necessarily that the program will run. The process of assembly will produce two more files these will be

DELAY.HEX and DELAY.PRN .

The HEX file when combined with the Loader program LOAD.COM once all the errors have been erased will produce a COM file. The PRN file is a combination of the files DELAY.ASM and DELAY.HEX, there by showing the mnemonic's and there relative addresses with the relevant object codes, as can be seen below.

```
                ;************************************
                ;*                                  *
                ;*      10 Sec DELAY ROUTINE        *
                ;*                                  *
                ;************************************
000A =          LF      EQU 0AH
000D =          CR      EQU 0DH
0005 =          BDOS    EQU 5
0100                    ORG 100H
0100 112201             LXI D,TEXT1     ;print text routine
0103 0E09               MVI C,9         ;uses CP/M BDOS call
0105 C5                 PUSH B          ;at address 0005
0106 CD0500             CALL BDOS
0109 C1                 POP B       ;preserves function reg
010A 210030     DELAY   LXI H,3000H     ;10 second delay
010D 0600               MVI B,0
010F 05         DLY     DCR B
0110 C20F01             JNZ DLY
0113 2B                 DCX H
0114 7C                 MOV A,H
0115 B5                 ORA L
0116 C20F01             JNZ DLY
0119 113B01             LXI D,TEXT2     ;print text for
011C CD0500             CALL BDOS       ;time out message
011F C30000             JMP 0           ;return to CP/M
0122 1A53544152 TEXT1   DB 1AH,'START OF DELAY ROUTINE $'
013B 0D0A0A5449 TEXT2   DB CR,LF,LF,'TIMED OUT $'
```

CONCLUDING ARTICLE NEXT MONTH

YOUR LETTERS

Questions

1.

i) Alan Crawford, Laceby, Nr Grimsby

I've just bought an MTX 512 from UK Home Computers. When I'm in 32 column mode (VS 4) I loose a column off the edge of the TV screen. Also, when I display large blocks of colour in the centre of the screen, the entire screen seems to 'kink' in the middle. Sound effects, such as the page key 'ping' cause sound bars all over the screen. Are these common problems or should I complain to UK Home Computers and try to get a replacement?

Ed-> The problem with loosing a character off the left hand edge of a graphics screen is common to all MTX's, it's just one of those inherent bugs. However you should not loose a character when you are in text mode. If you do, then you will need to alter the horizontal hold on the TV. Your problem with blocks of colour 'kinking', could be related to the Video Processor Chip. This chip is only able to display colours in blocks 1 pixel deep by 8 wide (When using Basic), within this box you can have one background colour and one foreground colour. This all looks ok until you try and use two or more foreground colours, then things become unpredictable. You would also appear to have some interference between your main computer board and the board on which the VDP is mounted. It is a good idea to shield this board from both the main PCB and the keyboard. You can do this by carefully opening the case (take care not to strain the cable that connects the keyboard to the main board) and inserting paper or insulating tape between the boards.

ii) I'm trying to convert a program that I wrote on my Atari 130XE to the MTX. I have a problem in that the Atari accessed the data directly each time it was required by using something like RESTORE BASE+10*INDEX. This doesn't appear possible on the MTX, can anyone offer a way round this?.

iii) Does anyone know where I can obtain a cartridge port cover? I didn't get one when I bought my MTX.

iv) Does anybody have any ideas about how I can get my MTX talking to my Atari 130XE? The Atari has a non-standard serial port, an expansion bus and two joystick ports. Any suggestions?

Ed-> I see no reason why you can not connect the two machines, probably the best method would be to join the serial ports, wiring things up initially can be a problem, electronics magazines or a paperback book on serial interfacing is probably the best place to start. If you do not have the serial ports connected to the MTX then you may be able to do it using the internal port connector. A lead to bring this port outside of the MTX is available from us and perhaps to get some ideas as to how to program it, it would be worth having a go at the LED interfacing kit!.

Hints & Tips

1.Arthur Wingrove Gosport, Hampshire.

In issue 8 and 9 I was interested in your articles on sorting. I have enclosed a simple program on Tandem sorting which may be of further interest. (Listing printed below). However - in running the first program (issue 8) I noticed a repetition of the test numbers and also that you had used a positive seed RAND 1000. This led to a doodling with random numbers. I had realised these were only pseudo random numbers but I had never previously observed that the use of a positive RAND value gave an exact number repetition each time the program was run. I wrote a short demo program:-

```
10 CSR 5,2
20 DIM A(50)
30 LET N=10
40 RAND 1000
50 PRINT
60 FOR I=1 TO N
70 LET A(I)=INT((ABS(RND*50): PRINT A(I);
80 NEXT I
90 CSR 5,7
100 INPUT "RUN AGAIN -- NOTE NUMBERS --- PRESS Y
THEN RET ";C$
110 IF C$="Y" THEN RUN
```

Every time the program is run it produces a screen display of :-

20 39 24 10 34 11 8 48 39 24

I would be interested to hear if other machines produce the same numbers. Using a negative value E.G. -1000 does produce a different set of numbers each time. The obvious lesson is to use negative values.

This led to a doodling with random numbers. I reset by using the pair of Black reset keys, the same table of values were produced starting :-

1) 6.25610352 E-04
2) 0.919647217
3) 0.962387085
4) 0.747253418

I went on to list a table of 50 values. If a newly switched on machine used a program having say 10 random numbers the first ten in the table would be used. If a new program was then run having say 15 random numbers the values from eleven through to 25 would be used. It appears as if the RND values are in the form of a Look-up table. Does anyone know the inner workings of the RND system?. As a party piece I wrote:-

```
10 LET N=0
20 PRINT RND
30 CLS
40 LET N=N+1
50 IF N=39 THEN GOTO 60 ELSE GOTO 20
60 PRINT RND
70 GOTO 70
```

As long as it is a newly switched on machine, the number 0.379638672 is always printed.

DISC BASIC ON THE FDX

by Geoff Gardiner

The FDX has DISC BASIC and as discs are so much easier to use than cassettes FDX owners will want to use it. It is a bit of a disappointment to find therefore that some of the glories of the MTX BASIC are missing from my FDX version. The particular characteristic that is missing is the ability the MTX has to make use of up to 512K of memory. I have the RS128 which has 128K of memory fitted.

Curiosity about the differences between MTX BASIC and DISC BASIC led me to acquire a copy of the MTX ROM listing and compare it with sections of DISC BASIC which I looked at with the Panel disassembler. When one is using MTX BASIC the first thing that happens at startup is a jump to 0194H of memory where there is a routine to count the number of 32K pages of memory that have been installed at location 4000h and keep a record of the result, plus one, because the program assumes that there will be a half page commencing at 8000h on the memory page after the last page commencing at 4000h.

I presume that DISC BASIC for the FDX does not have this page counting routine because it can use only 64K of memory. A glance at the memory map leads one to guess why. When the system is ROM based the bottom 2000h of memory is common to the first 8 pages of memory and the routines located in it are available therefore to 8 pages of memory. But when the system is not ROM but RAM based the bottom 2000h of each page is separate. I suppose this could be overcome by having the bottom 2000h of basic repeated on each page, or by putting the routines in high memory, above 0C000h, which is common to all 16 possible pages of memory. A further problem with RAM based memory may be that it is probably not possible to split a page at 4000h, as happens with ROM based memory, although I was curious at seeing that the memory configuration when using PANEL is "90", not "80" which gives the impression (probably wrongly) that page one of low memory is being combined with page 0 of memory above 4000h.

PANEL in ROM is on page 1 between 2000h and 4000h for MTX basic uses 2 pages of ROM between 2000h and 4000h. As DISC BASIC uses only one 64K page of RAM memory it has to have the Panel and Assembler routines on page 0. In addition there are the disc handling routines to be provided for, and the 80 column screen driver. These routines occupy locations 4000h to 8000h of RAM and that explains why an FDX system using DISC BASIC uses from location 8000h as temporary programming area (TPA) whereas the MTX 512 starts from 4000h. FDX BASIC has no more TPA than a an MTX 500.

The illustration in the manual of the ROM based memory map shows DISC on pages 4 and 5 of the memory, CART, whatever that may be, on page 7. I assume that the plan was to have a ROM based system providing for disc and 512K of memory as standard, all the routines being in ROM. Is that what the SDX system provides? I have not seen the SDX ROM listing but perhaps someone else can answer this question. If the answer is "yes", it would be worth while for FDX users who are basic enthusiasts to have an SDX as well. These are tantalising possibilities. But is there any reason why FDX owners who have extra memory should not have a disc based BASIC for loading into RAM which would make use of all the memory available? Is it just a question of finding someone to write it or does it exist?

Another enhancement that would be nice for FDX owners now that memory chips are cheap is an improvemnet of the 80 column display. The 80 column board uses the Motorola 6845 CRTC but makes only limited use of that chip's graphics potential. The chip can handle 16K of memory for the visual display but the FDX is equipped with only 4K. The FDX's display capability is limited to what the two character generators can do as there seems to be no way of bypassing them. A browse through the data sheet for the 6845 chip makes clear that it can do far more, as any BBC, Amstrad, or IBM PC owner can confirm. Obviously as the MTX has excellent graphics capability, but uses a different chip, it was doubless originally regarded as an unnecessary and expensive duplication to provide full graphics on the 80 column board.

Title: SUPER BIKE By Jim Thoms
Machine: MTX 500 or 512
Publisher: Orion
Price: Around £6.00 Review By James Townley

Super Bike is an arcade type game which has been about for quite some time. It takes nearly 3 minutes to load, but much of this time is taken up with the loading screen and the game screen, both of which are Memosketch designs. Anybody with Memosketch will immediately recognise the superb loading screen as one of the sample drawings.

The game is played on an everlasting road, with a few mountains on the horizon. The screen itself is quite good, but only a static display; the only signs of movement are the white lines in the middle of the road. The motorbike can be controlled either by joystick or cursor keys - I found a combination of both most effective as it allows better speed control as well as steering.

There are three hazards to contend with; road signs, motorcyclists and helicopters. The road signs are easy enough to avoid. The maniacs who come at you are a little more difficult as they aren't as random as they first seem! The helicopters drop what looks like a pot of custard from the skies. This can be quite annoying as the custard stays on the road much longer than the other obstacles and causes many an accident. The only other factor is the countdown. This seems to be roughly linked to fuel consumption and stops you going at 2Km/h! The idea is to go as far as possible down the road.

I found the game rather boring after a while, so I would only recommend it to occasional games players or people who like seeing how long they can keep doing the same thing! I can't see any end to the road, but if anyone out there has found one, please write in and tell us all about it!!

Graphics: 8
Sound: 6
Instructions: 4
Addictive Qualities: 5

- -

Note...Note...Note...Note...Note...Note...Note...Note...

Remember to send in your high scores for the High-Scores table. It has been some months since it has been printed, so why not send them in and give others a new score to beat for next month!!!

Title: The Wall By Fredrik Lundkuist
Machine: MTX 512
Publisher: Orion
Price: Around £6.00 Review By James Townley

The Wall is very much an arcade game and comes from a Swedish MTX owner. It takes about 2.5 minutes to load. If you like the little tune it plays (or you can't find how to stop it!) and wait to the end you will discover a guide to the various baddies you are about to encounter.

After selecting game or practice mode you can begin. The Wall is basically 'Breakout', with lots of added features. The wall is made up of six rows of bricks, grouped into three colour bands. The idea is to break through, then send your ball through the exit. Once this is done your bat will fly up and must be steered through the exit, allowing you to go on to the next level. If this is not done you have to start the wall again. Needless to say this can be very frustrating!

There are three main enemies: Zipper, Gulpers and Spiders. Zipper causes havoc by sending your ball in random directions whenever you hit it. Gulpers are extremely nasty and climb up and down the side walls, eating your ball whenever you bounce it on them. Spiders are the most interesting of all the baddies; first a large spider scuttles across making a horizontal web, then hordes of baby spiders descend from the main web to make vertical webs. These hinder you as you try to keep the ball in play, but you can always kill them and get a few points.

Another wall resident is the yellow bird. You can get a bonus from killing the poor fellow (if you're that sort of person) but I tended to lose control of the ball if I purposely went for it. The final point of interest is the 'soft' blue ball which becomes semi-hard if released and can be used to cause a bit more damage to the wall. Unfortunately it can't be used to break out.

On the whole I found the game very enjoyable to play. The graphics are colourful and the ball can be made to do some amazing things. There is a fair amount of skill needed to keep the ball under control as different parts of the bat make different angles. I would recommend it to anyone, especially those who are tempted to write it off as a breakout clone.

Graphics: 8
Sound: 8
Instructions: 4
Addictive Qualities: 9

Title: TOURNAMENT SNOOKER AUTHOR:
Machine: MTX 512
From: MAGNIFICENT 7 SOFTWARE
Price: £8.00

Review by James Townley

Tournament Snooker is a two person simulation using a bird's eye view of a snooker table. As well as the table there is a large cue ball and a superb scrolling message box on the screen. After first choosing from 3 languages the message box tells you that the program was written by Godwin Graham - who for some reason is known as the 'Geezer' - and is published by Magnificent 7 software.

The basic way of potting a ball is to revolve the cue around the white ball, using 'Q' and fine 'Q', until it points in the desired direction. The appropriate key is then pressed twice to decide the power of the shot. The balls then rebound around the table and hopefully the chosen ball disappears into the chosen pocket!

One of the program's clever, and extremely useful features is the large cue ball. This shows how the white will strike the target ball if the current cue direction is followed. It also indicates where the cue will strike the cue ball. By moving this it is possible to impart various types of spin.

I found the program extremely realistic in all respects. The balls moved with correct speed and the use of spin was accurately taken into account. The sound affects are minimal, but quite realistic. Try as I might I couldn't find any sloppy programming or niggling faults. The game can be saved to tape in seconds and the wide range of features such as practice mode, instant replay and setting up the table are all explained in the excellent instructions.

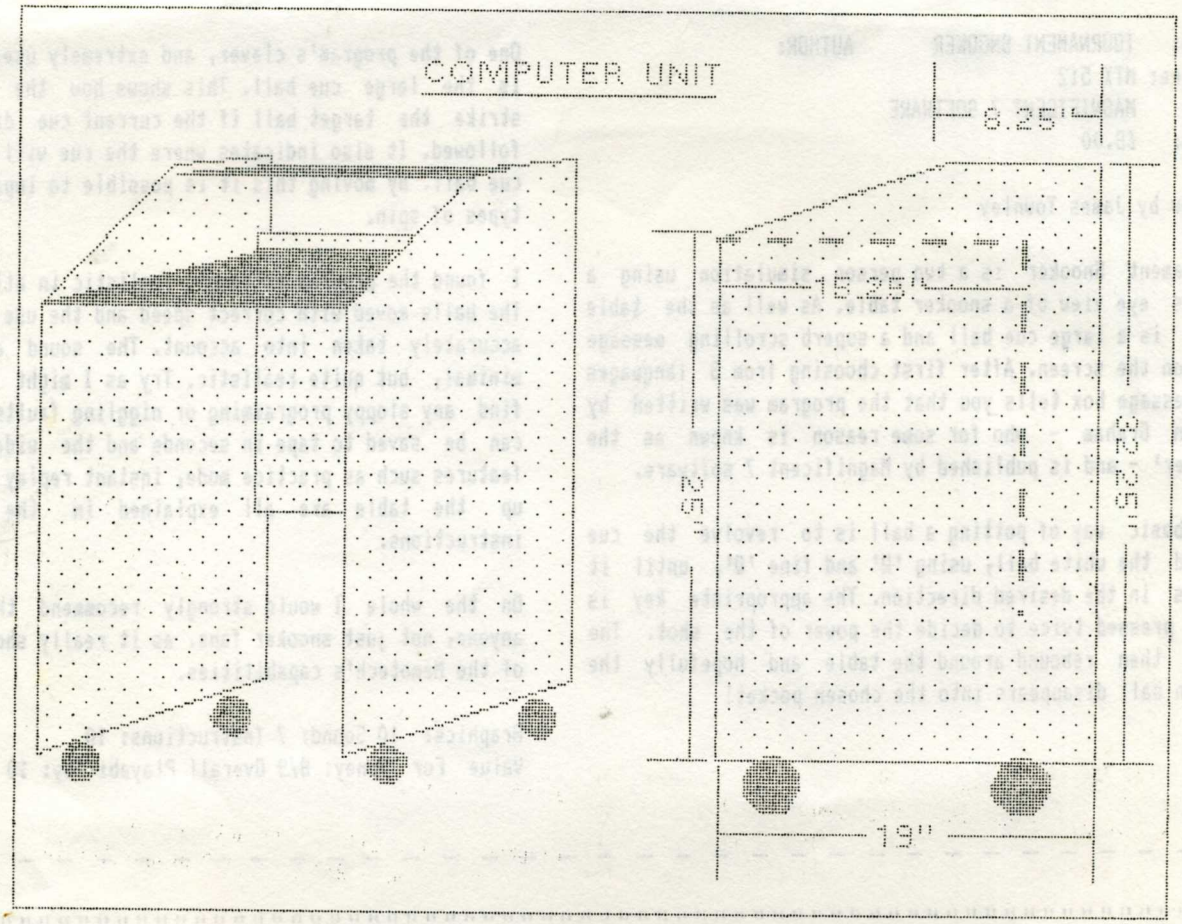
On the whole I would strongly recommend the game to anyone, not just snooker fans, as it really shows off some of the Memotech's capabilities.

Graphics: 10 Sound: 7 Instructions: 10
Value For Money: 8/9 Overall Playability: 10

Program Library

For anyone that does not have a spare disc or two, or does not have the facility to easily send the club a disc, The club will supply 5.25" 500K disc's for library use at a standard fee of £1 per disc. This may well also prove to be the best method of obtaining program library information for those with 1Meg 5.25" disc drives, as I have great difficulty in reading 500K disc's formatted on a 1Meg machine. (I think the machine formats the wrong half of the disc!!).

Alan Hamilton is in the process of setting up the library at his end, hopefully taking it over in the near future. He intends to boost it up a bit, by supplying more in the way of CP/M disc's, and better quality documentation.



Supplier: Texas Price: £18.00 - £20.00
Dimensions: 31.75" * 19" * 32.25"
Construction: .75" Contiboard look-alike
Finish: Teak effect or white with red trim

The unit is sold as a flat pack for home assembly but, since all screw holes are pre-drilled, this is a simple task. In view of the price I was pleasantly surprised at the quality: it was adequately packed, all screw holes lined up, exposed screw heads were capped and all 'show wood' was neatly finished with edging tape.

When assembled, there are two working surfaces, the main one being 30.25" * 15.25" and the top shelf 30.25" * 8.25". The latter easily takes my Brother HR5 printer and Philips 7502 monitor... I had some doubts about this but an inspection (before I bought the kit!) revealed four rubber feet underneath with a footprint of 10" * 7".

The main surface is more than adequate for the computer and 3.5" drive with plenty of room behind for both power packs. It would also take the printer transformer but, unfortunately this would mean that the computer to printer cable would be routed over a power pack which didn't seem ideal for good data transmission, so it had to go elsewhere.

The desk is half-backed, presumably to give structural rigidity, but it does offer two further advantages:

1. It gives a raised edge at the rear of the main shelf which prevents papers (and hardware!) being pushed onto the floor.

2. It is recessed enough for one or more multi-socket strips to be permanently fixed and there is room for all surplus cabling to be neatly clipped to give a tidy layout.

CONTINUED FROM PAGE 12

I have a few minor criticisms:

1. The height of the main working surface is just adequate but 2" more would have been much better.

2. The castors, presumably intended for moving the unit when cleaning, are simply screwed into the bottom edge of each side. So those who intend to trundle the assembly from bedroom to sitting room will have to take care to put no sideways load on it and it would be wise to screw strengthening strips along the bottom edges.

3. The top shelf is unsupported and, over a period of time, will probably bow from the weight of the monitor.

So I intend to make a few modifications using two pieces of Contiboard, 6" and 9" wide I bought with the kit. Firstly I shall put a 6" wide shelf along the back underneath the main surface to take the printer transformer and other odds and ends. The second was originally to put a 6" wide central support under the top surface (useful stuff this Contiboard!) but then I decided also to add a 9" shelf about 1.5" below the top one to house the computer and CP/M manuals. Neither will tax the ability of even an incompetent DIYer like me since only butt joints need be used.

So, for a couple of hours work, I shall have a compact work station and can say good-bye to the fag of assembling all the bits and pieces every time I want to play. However, like most of us, I cannot resist the temptation to 'improve'... so when I saw some A4 size office trays on special offer in our local stationers, I thought how useful they'd be for printer paper and now I'm trying to think of how I can quickly clip them on the side when needed. Also I fancied a pullout flap at each end of the main work surface for note books etc. The flaps are no problem..good old 9" Contiboard again! But the runners are another matter since the right hand one will have to bear a bit of pressure when I'm writing. I think the answer will be to make a sort of open-ended box from some scrap wood in the garage and screw and glue these to shelf and sides. This would have the added advantage of providing extra support for the main surface which can't be bad... Fortunately the last two mods. can be done with the desk in use otherwise I can see me still pondering the matter in six months time...and the desk not in use!

Jack Penley-Martin.

GAP FILLERS...GAP FILLERS...GAP FILLERS... GAP FILLERS...

Hint
When writing a program it is sometimes nice to know how much memory you've used, at the end of your program add a CODE line (Using ASSEM instruction), there is no need to type in any code just enter the assembler then exit again, when you list your program the last line will read like this:-

```
LINE NUMBER CODE ----- 100 CODE
HEX ADDRESS RET  ----- 440F RET
```

Then by subtracting 4000H (8000H for MTX500) from the above address you obtain the program length in HEX.

Basic Screen Dump Routine

```
5999 REM DUNP ROUTINE
6000 VS 4: LPRINT CHR$(27);"3";CHR$(12);
6001 FOR Y=191 TO 0 STEP -8:LPRINT CHR$(27);"K";
     CHR$(255);CHR$(0);
6002 FOR X=0 TO 255 : LET A$=GR$(X,Y,8)
6003 LPRINT A$;:NEXT:LPRINT:NEXT
```

This works for the DMX 80 and other Epson compatible printers.

A simple way to convert your MTX 521 to a 500 is:-
1) POKE 0 into location £FA7A (Dec 46122)
2) Type NEW and press the return key
You now have a 500. I have found that it was safer to switch the machine off and on again before setting the machine to a MTX 500 as this cleared out the memory. I was caught out on several occasions when I thought my program was working on a 500 but in fact it was calling routines that were still in memory between £4000 and £7FFF.

Next Month...

Sorting it Out - Part III
We have three more sort routines for you to try out!!!