Memotech's MTX Computer Reviewed

An impressive new British Z80-based microcomputer.

Dick Leslie

When Memotech's MTX computer was launched at the Earl's Court Computer Fair in June, it was clear that if it lived up to its advance specification it would be one of the most substantial and potentially powerful computers available in the sub-£300 price range. Now that it is available in production quantities, just eight weeks after the proposed release date in August, the time has come to look inside this new British Micro to see what makes it tick.

Memotech seem to have provided everything bar the kitchen sink in the MTX package. Inside the polystyrene box lies the computer, all 2.6 kilograms of it and with an array of 79 keys spread along its 49cm width, a separate power supply (complete with illuminated mains switch), five cassettes tapes, an assortment of leads and documents, and the manual.

One of the leaflets explains that the manual supplied is provisional, and that the proper version is expected from the printers and will be sent on to the MTX purchaser. In view of this, it would be unfair to comment on the quality of the provisional handbook, though at over 250 A4 pages it will certainly keep most users happy until the proper version arrives.

The MTX computer is sold as an all-purpose computer, and from its starting price of £275 inclusive, it can be expanded in stages until it holds 512K of on-board RAM and quite a few megabytes of memory on Floppy Discs, Silicon Discs or Winchester Discs. Since the MTX is using the Z80 CPU as its processor, and that can only address 64K of ROM/RAM, Memotech are using software to page in and out the various options of on-board memory. This process is transparent to the user, who merely enjoys the vast amount of memory at his disposal, but to the software writers who write the operating code for paged systems it represents an extra challenge — paged software is that such more difficult to de-bug!

An interesting point about the MTX memory map is that the ROM is normally in low memory, so that the Z80 restarts may be utilised, but the CP/M operating system requires RAM at low memory so when an upgrade to CP/M is made, part of the memory map alters. **Figure 1** shows the memory map of the basic MTX 500, and shows how the extra memory pages are filled in as extra RAM upgrades are purchased.

The 32K MTX 500 and the 64K MTX 512 have their 24K of ROM represented by SYS-A, SYS-B and SYS-C, and have 32K of RAM from 8000H to FFFFH. The MTX 512 uses 16K of its extra RAM to fill the gap at 4000H-7FFFH, (area (a) in **Figure 1**) and the other 16K is banked up on page 1 between 8000H-BFFFH (area (b)).

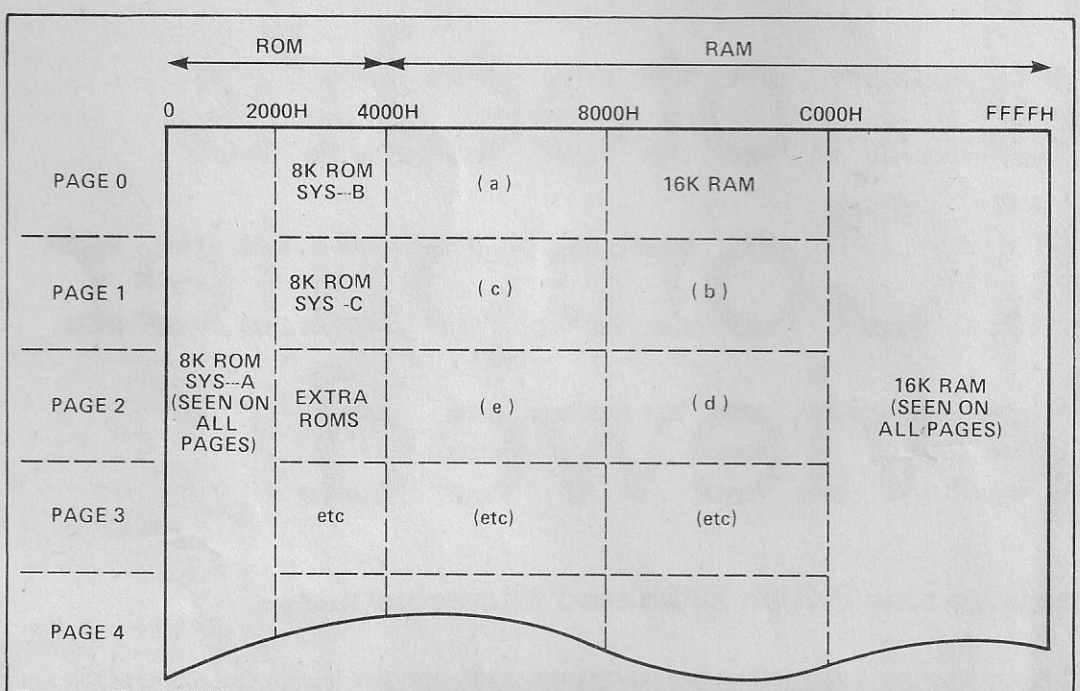

Figure 1. The MTX memory map, showing the paged memory system for both extra ROM and RAM

The additional memory boards begin to fill up the pages as appropriate in the order (a),(b),(c),(d) etc., while extra ROM is added in a similar manner on top of SYS-B and SYS-C. Eight pages of ROM are allowed, and sixteen of RAM, but the diagram shows just four pages.

The MTX case consists of two aluminium extrusions hinged together at the front. A single pcb slides into the lower part, and the voltage regulators are bolted to the metal so that the whole case acts as an enormous heatsink, which is very effective. The keyboard is housed in the upper portion and aluminium plates at each end of the computer hold to two halves together. The whole assembly is extremely rigid.

To take a look inside you must first unscrew the screws holding on the end plates. These screws are really quite childproof, since you need a small allen key to turn them, but once the end plates fall away the keyboard hinges up and the insides are revealed.

There are 48 chips on the main pcb, and a few more on a little "piggy back" pcb near the UHF modulator. The little board isn't a correction — in fact there are no signs of any last minute alterations on the boards — but contains the British TV circuitry; export models of the MTX will have different piggy-back pcbs to suit the requirements of the recipient country.

There are edge connectors on both ends of the main board: they are identical and contain all the important CPU signals (see **Figure 2**). One connector is intended for external ROM packs, such as a ROM FORTH, while the other connects to any expansion board which may be fitted in the large space beneath the auxillary keypads. The expansions may be RAM boards or the communications board. The latter board carries two completely independent RS232 channels and a disc drive bus, but not the floppy disc controller itself since that and certain other boards fit inside the disc-drive case.

An Abundance of Chips, Sockets and Ports

All the chips are standard off-the-shelf types, apart from a small TTL ROM which does all the fancy decoding for the paged memory. There isn't a ULA in sight, which perhaps explains why there have been no serious delays in getting this micro into the market place.

The TTL ROM and the larger ICs are socketed, and all components are laid out neatly and without any space being wasted. Memotech are using the Z80A processor running at 4MHz. At this speed it runs a BASIC which is faster than the Commodore 64's, but not quite as fast as Acorn's BBC computer. However, there are rumblings at Memotech about fitting a 6MHz version of the Z80 chip, and if that is successful then the MTX would be one of the fastest micros around.

It is often said that devotees of the Z80 and 6502 CPUs are not capable of seeing each other's point of view, and I must declare that I sit firmly in the Z80 camp. The Z80 chip allows compatability with CP/M, but that is of little concern to users outside the business field. However there are more immediate advantages associated with the Z80: all ports have their own address allocation and do not intrude into the memory map and it is on the whole much cheaper to add memory to a Z80 computer; the fact that a 32K memory card for the MTX costs

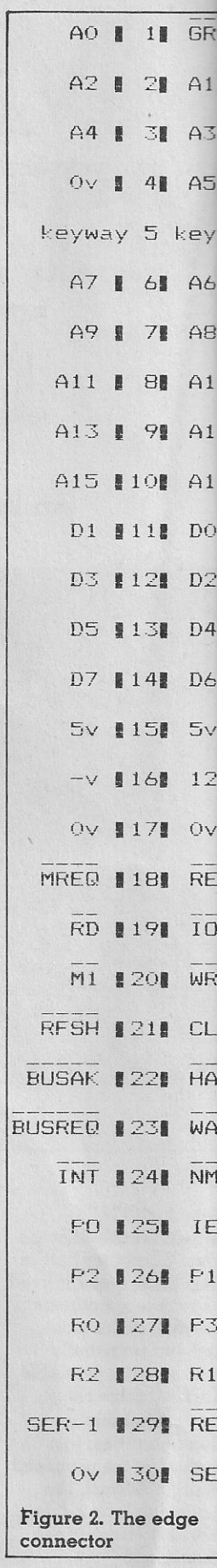

Figure 2. The edge connector

ABS	ASC	ATN	LN	CHR$	CLEAR
CLS	CONT	COS	DATA	DIM	EDIT
EXP	AND	GOSUB	GOTO	INKEY$	IF-THEN-ELSE
INPUT	INT	LEFT$	LEN	LET	LIST
LLIST	LOAD	LOG	MID$	NEW	FOR-STEP-NEXT
OUT	PAUSE	PEEK	POKE	PRINT	ON/GOSUB
RAND	READ	REM	RESTORE	RETURN	ON/GOTO
RIGHT$	RND	RUN	SAVE	SGN	SIN
SQR	STOP	STR$	PI	TAN	USR
VAL	VERIFY	OR	NOT	MOD	LPRINT
BAUD	CLOCK	INK	PAPER	PAUSE	NODDY
PLOD	CSR	PANEL	SOUND	ASSEM	ATTR
COLOUR	EDITOR	DSI	SDBUF	TIME$	GR$
SPK$	ROM	CRVS	VS	VIEW	GENPAT
CIRCLE	DRAW	PLOT	ARC	LINE	ANGLE
PHI	SPRITE	MVSPR	ADJSPR	CTLSPR	AUTO

Figure 3. BASIC commands and functions available on the MTX.

£50 and a 20K memory card for the BBC micro costs £114 bears witness to this!

Most of the timing in the circuit is carried out by the Z80 Counter-Timer Circuit (CTC) chip. This IC has four independent timing channels, and two important tasks it has to do for the MTX are to control the video circuitry and operate a real-time clock.

This clock is interesting, in that it is easily set up from BASIC. Unfortunately the clock has no battery back-up, since the CTC isn't a CMOS device, and draws rather a lot of current, but it does count in hours, minutes and seconds while the computer is running, up to a total of 99 hours, when it cycles round again. Time-out signals from two other CTC timers are taken to the edge-connector bus, which might prove convenient to the builders of MTX add-ons.

I would guess that one add-on which won't be needed is a sound board. The MTX uses the Texas Instrument's SN76489 as its sound processor, and very good it is too, especially if the output is taken to a hi-fi. There are three tone generators and one noise generator in this chip, and they can be manipulated quite simply from BASIC using the command SOUND followed by 3 parameters, or more complex sounds including synthesised music can be created using SOUND followed by 7 parameters.

Although there are no ULAs present in the MTX, the chip count has certainly been reduced by the decision to use the TMS9929A video-processor to handle the screen display. This is the European version of the Texas chip, which simplifies the entire video circuitry, but it still has a nasty habit of putting oval-shaped circles onto our TV screens. Memotech are working on a modification to improve matters, but I am told that the problem is not nearly so bad on a video monitor.
The video socket is a high-quality BNC socket and it shares the back edge of the computer with eight other sockets of various types and two cut-outs which are ready for the RS232 interface.

Apart from the usual power, cassette and TV sockets, there is a Centronics-standard printer port, two joystick ports, and a sound output for connection to an external amplifier. Not content with that, Memotech has also provided an uncommitted parallel I/O port on the pcb itself, and a suitable slot at the back to allow ribbon cable to pass out.

The Software

The MTX has BASIC and powerful low-level languages all resident in ROM. **Figure 3** shows the BASIC commands and functions available, which include a fair number of powerful graphic and sprite handling commands.

A closer look reveals that some rather more standard words are missing from the BASIC, presumably due to lack of space. TAB and DEF/FN are missing, and there seems to be no IN to accompany the OUT. Other more advanced commands that you might miss having are MERGE, and the structuring facilities of REPEAT/UNTIL/DEFPROC which the BBC micro has made popular.

The MTX saves its variables with the BASIC program, so you don't need to worry about separate save commands for those, but there may not be a direct way of saving and loading machine code — this is an area where the provisional manual is vague. However, most of the time any machine code that you write will be incorporated in the BASIC listing and will therefore load and save automatically.

This is possible because the MTX has a built in assembler which can be called at anytime from within BASIC. If you type ASS. 10, the line 10 CODE will appear in the BASIC listing, and the computer will pass into assembly mode. Using the assembler is easy, and the editing and listing facilities are more than adequate. The syntax of each line is checked at the time of entry (as it is with the BASIC) so the system is a pleasure to use.

The object code created is placed in memory starting at the line called CODE, along with a few other labels and essential pieces of information, stored as well. The source code isn't saved, and when you see it listed you are really looking at a disassembly of the machine code itself; and you may have a print out of the listing to scrutinise, if you so wish. The whole procedure is designed to optimise the use of memory space and to encourage programmers to use machine code at appropriate times in their BASIC program. In fact one of the first uses you are likely to make of the assembler will be to replace that missing IN command!

The Front Panel

The "front panel" is a software implementation of the flashing lights and switches in which the original mainframe computers used to specialise. It represents a view into machine-language of the MTX, but as **Figure 4** shows, the modern day panel consists of alphanumeric information on the VDU, rather than the fireworks display so beloved by the directors of science fiction films.

Behind the "front panel" lies a machine code monitor which enables the user to test and debug programs which have been written on an assembler. The RML 380Z/480Z, which is the micro widely available in schools and colleges under the DoI scheme, has a similar panel command, but the MTX version is definitely superior since it has the ability to disassemble blocks of code.

Figure 4 shows that there are four display areas; the hexadecimal tabulation at the bottom of the screen shows the contents of memory around a specified byte (A193 in the example) and you can scroll up and down the memory ad-infinitum using the cursor control keys. It is also possible to change the contents of the byte pointed to (F1 in the example) and so make adjustments to the program under review.

At the top right there is a display of the contents of the

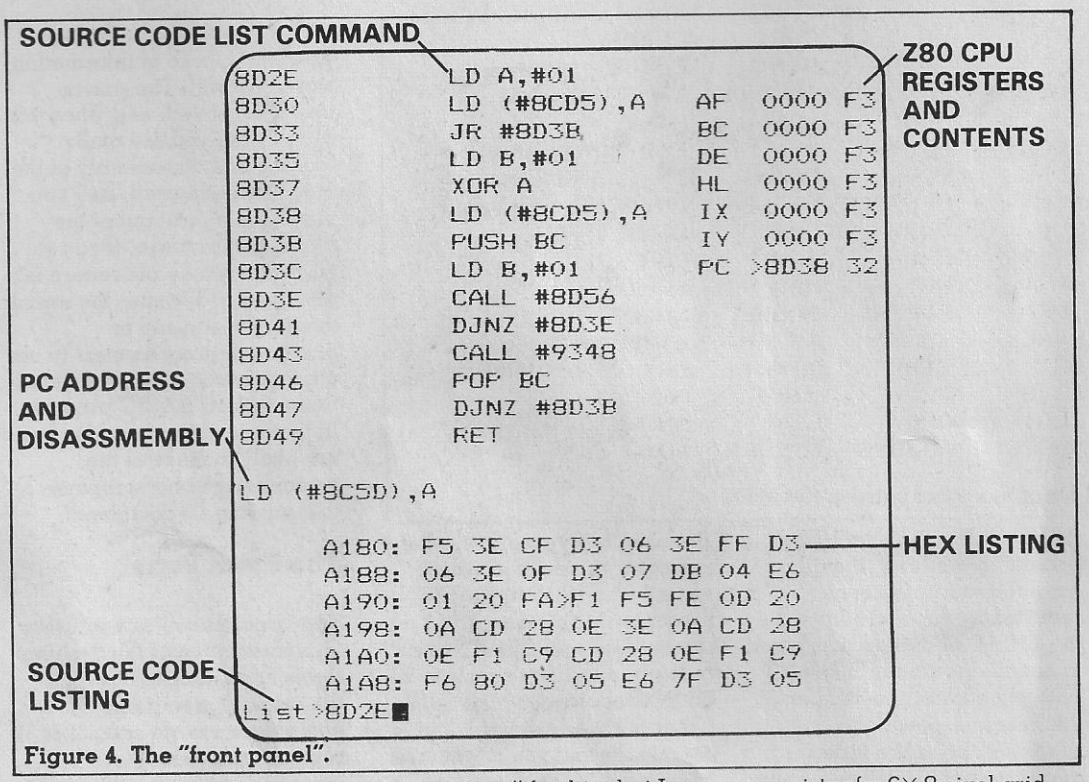

Figure 4. The "front panel".

Z80 registers. They show zero until they are changed by the programmer, or until a program is single-stepped, when they will show their own content and the content of the address at which they are pointing. Z80 flags are shown, as and when they come into operation but the Interrupt Vector register is not show: perhaps because it would be unwise to attempt to alter its contents. In the example given, the Program Counter (PC) has been set to 8D38 and that address holds byte 32H. The instruction that begins at the PC address is automatically disassembled and displayed at lower left as source code.

Entire blocks of memory may also be disassembled as a list (top left) which can have a starting address quite different from that set for the Hex tabulation. Remember, though, that machine code DATA disassembles into invalid source code, and there is never any way around that particular problem. However, Memotech have made sure that when source code is listed under command of the MTX ASSEMBLER, data bytes are correctly identified as such. One command missing from the front panel is a "search for byte pattern" facility, but I think that this is more than compensated for by the facility of listing in source code format, and Memotech are to be congratulated for providing this most excellent of monitors.

Using the MTX Screens

The MTX screen can operate in four modes:

1) Graphics M1:256× 192 pixels

2) Graphics M2:256× 192 pixels

3) Text Mode:40× 24 character positions

4) Multicolour mode:64× 48 positions

Text may be used within graphic modes 1 and 2, in which case there will be 32 character positions across the screen, and a pattern position will be made up of 8× 8 pixels. In Graphics Mode One 256 possible patterns may be defined for the 768 pattern positions, with two unique colours allowed for each pattern definition. Graphics Mode Two provides 768 pattern definitions for the 768 pattern positions, and all 16 colours may be used in a single pattern position.
In Text Mode a pattern position consists of a 6× 8 pixel grid, and just two colours can be defined for the whole screen. Finally, in Multicolour Mode the screen is broken into a grid of 64× 48 positions, each of which is a 4× 4 pixel pattern, and each position can be given its own unique colour from the palette of 16.

All the patterns created by illuminating the pixels exist on what is known as the 'pattern plane' of the video screen. Behind the pattern plane there is the backdrop plane, a somewhat larger area which also happens to create the border around the pattern plane.

Colour for these two planes can be easily selected by the commands INK and PAPER, but there are many more complex commands available to set up the pattern plane, and to create any one of 32 sprite planes which overlay the pattern plane.

Sprites can be created and manipulated using the powerful commands of Memotech's extended BASIC. All the planes have been assigned an order of priority, with sprite 0 taking top priority and sprite 31 taking least priority. The pattern plane may be regarded as 32 and the backdrop 33 in the priority hierarchy. Whenever a pixel on a higher priority plane is transparent, the colour of the next plane can be seen through that plane.

There is much more that could be said of sprites: they make the writing of animated programs relatively simple, so that would-be programmers at home or in school can concentrate on ideas and style without being too far sidetracked by the time-consuming task of writing code. The MTX has sprites, and you don't have to buy any extras to use them. Twenty fou pages of the provisional manual are devoted to them and other advanced graphics features, and the appendices are helpful in the detail they provide for machine-code programmers.

If you never have the time to experiment with high resolution graphics, there are many more features of the MT which should please you. The separate editing keypad and the function keys (for typing i user-defined characters) allow for efficiency in editing programs, accepting INPUT from the screen, and in composing entire screen-fulls of text.

The Memotech computer has NODDY, a text-oriented language which can be used its own right to create interactive programs, or can be used from BASIC to call u predefined screens. A direct screen input (DSI) command allows the user to roam about freely within the screen, ending when carriage return pressed. Useful keys such as "tab" step the cursor forward in tabulation steps; control keys have certain functions such as switching the cursor and off, and various "escape key sequences" do other use things — Escape, K duplicat a line for example.

Clearly, a great deal of thought has gone into the design of the MTX range.

MTX 500

These days it is expected of a new computer that it should have a large memory, high-resolution colour graphics, multichannel sound facility and a joystick port. The MTX has all these things. It seems to have remarkably few problems and shortcomings and the track record from Memotech suggests that bugs do not stay uncorrected for long.

The keyboard is superb — goodness knows how much it is costing Memotech to fit it — and it gives the machine a very definite professional style. A closer inspection of the hardware and software reveals that the professionalism runs deep, and that although the machine may be used by a beginner, it is equally suitable for the expert and likely to keep any user satisfied for a good many years. If Memotech can get their MTX 500 and good software support into the shops and the schools in quantity, then I think they are set fair to take a large slice of the market away from Acorn and Commodore.