

---

# REMEMORIZER

User's Guide







By Andrew Key

# User's Guide

REMEMOrizer Version 1.0.0

(16/01/2013)

Document Revision 1.2

## Credits

Relating to the construction of REMEMOrizer, in no particular order...

Geoff Boyd, for designing the Memotech hardware, and making it available to me. Also, for providing feedback on my earlier design.

Brian Pritchard, author of the MTX Series Operators Manual. This is really quite a comprehensive technical document, especially towards the end. Not only does it reproduce tech sheets from the chip vendors, it also includes the full MTX circuit diagrams. Invaluable when implementing hardware.

The author of the FDX User Manual and Peter Ketzschmar for making a scan of it available from his website. Especially useful for the exact pixel layout of the 80 column card. See the MEMOTECH MTX Computersystem website: <http://mtxinfo.de/>

William Brendling for supplying PDF scans of the SDX ROM listing. The SDX support in REMEMOTECH, and thus REMEMOrizer, wouldn't have been possible without this.

Lez Anderson for sending electronics stuff (tools and components), some of which I was able to use on this project.

The Zilog Z80 Technical Manual and the Zilog Z80 CTC Technical Manual.

And me of course, for designing REMEMOrizer, using the information above.

## Document Revision History

1.2	16/01/2013	Updated for REMEMOrizer RS232 fix
1.1	21/10/2012	First issue

# TABLE OF CONTENTS

1. Introduction.....	7
2. Design Overview .....	8
Hardware .....	8
Memory .....	9
Cassette Tape .....	14
Floppy Disc Controllers .....	14
80 Column Card .....	15
Sound .....	15
Software .....	16
OS ROM .....	16
BASIC ROM .....	16
CP/M Boot ROM .....	17
SDX BASIC ROM .....	18
TAPE SUPPORT .....	18
3. Installation and Setup.....	19
REMEMOrizer contents .....	19
Schematic Diagram of REMEMOrizer Board.....	19
Preparation .....	20
Deciding on whether to use the replacement ROMs.....	20
MTX500 Jumper .....	21
MTX512 S2 .....	21
RS232 .....	22
Plug it in .....	22
4. Operation .....	23
Power-on .....	23
Reset .....	23
RAM Disc .....	23

SD Card.....	23
Sample SD Card.....	23
Accessing SD Cards from CP/M.....	24
Accessing SD Cards from MTX BASIC.....	24
Accessing SD Cards on Linux.....	24
Accessing SD Cards on Windows.....	26
Tape Operation.....	28
Tape Mode Selection.....	28
Tape Breaker.....	28
80 Column Screen Mode Selection.....	28
5. CP/M Utilities.....	29
RECONFIG.COM.....	29
REFORMAT.COM.....	29
REZTAPE.COM.....	30
REZTAPE.....	30
REZTAPE <i>slot</i> .....	30
REZTAPE INPUT.MTX <i>slot</i> .....	30
REZTAPE <i>slot</i> OUTPUT.MTX.....	30
REZTAPE ?.....	30
REZPATCH.COM.....	31
MTX.COM, MTXL.COM and RUN.COM.....	31
6. Incompatibilities.....	32
Index.....	33

# 1. INTRODUCTION

REMEMOrizer is an add-on for the Memotech MTX series of home computers which were produced in the UK between 1983 and 1985. The range included the MTX500, MTX512, RS128 and MTX512 Series 2, these were all Z80 based machines, the major difference between them being the amount of installed Random Access Memory (RAM).

As was typical of similar home computers of the time, the base models were limited to a display resolution of 40x24 characters or 256x192 pixels and relied on the use of Compact Cassette tapes as their storage medium.

A number of expansion options were available at the time, including disk based storage systems (FDX and SDX) which allowed the user to run the industry standard CP/M operating system with an 80 Column card installed.

Although some cheaper disk options were introduced late in the machine's life, these were not common and most users had to make do with video output to TV or low resolution monitor and tape based storage.

REMEMOrizer is a 21<sup>st</sup> Century enhancement for the MTX that provides random access storage capabilities, using Secure Digital (SD) cards emulating floppy disks, and 80 Column mode output to Video Graphics Array (VGA) compatible monitors. In short, it's an SDX, only smaller, faster, cheaper and better. This is the upgrade you wished you'd had in 1984.

REMEMOrizer provides the following enhancements :-

- 80 column card, with 80x24 and 80x48 modes, outputting to 640x480 @ 60Hz VGA
- SD Card as a large and fast alternative to floppy disk
- 512KB of extra memory
  - 320KB is directly usable, and is typically used as a RAM Disc
  - 128KB is hidden, and is used as virtual cassette
  - 32KB can be used to upgrade MTX500s to 64KB, with a PAL replacement
  - 32KB can be used if your MTX has no memory at all (ie: a failed upgrade)
- CP/M boot ROM and SDX BASIC ROM providing access to SD Card and RAM Disc
- Replacements for the fixed OS and BASIC ROMs, providing Virtual cassette support

REMEMOrizer was designed by Andrew Key (email: andy.z.key@googlemail.com)

The latest copy of the REMEMOrizer documentation and technical data can be found on-line at <http://www.nyangau.org/REMEMOrizer/REMEMOrizer.htm>.

## 2. DESIGN OVERVIEW

This section provides an overview of the REMEMOrizer design and functionality. As a User Guide, this document is not intended to give the reader detailed instruction on the assembly and programming of the REMEMORIZER hardware. It is assumed that the reader already has a working REMEMOrizer; if this is not the case, full instruction for building one can be found on the web site.

The heart of REMEMOrizer is a field-programmable gate array (FPGA), the Xilinx Spartan-3E, built into an OHO-Elektronik GODIL (or good old DIL, where DIL is dual in-line) electronic programmable logic module. Full details of the GODIL are available on the *trenz electronic* website, <http://www.trenz-electronic.de/products/fpga-boards/ofo-elektronik.html>.



The GODIL executes the complex logic functions and is interfaced with an SD card reader module which acts as an emulated floppy disk, a 512KB static RAM module and supplies the VGA output from the emulated 80 column card.

### HARDWARE

REMEMOrizer includes emulation of the following hardware :-

- Memory
- Cassette tape
- Floppy Disc Controllers
- 80 column card
- Sound

The remainder of this section discusses the hardware emulation in some detail, however, an in depth knowledge of the inner workings of REMEMOrizer are not required before the user can install and use it. You may choose to skip this section and head straight to Chapter 3 which discusses REMEMOrizer installation.

***Please ensure you have read and understood Chapter 3, Installation, before attempting to install and use REMEMOrizer, serious damage could occur to your MTX and/or REMEMOrizer if you configure the jumper settings or ROMs incorrectly!***



## MEMORY

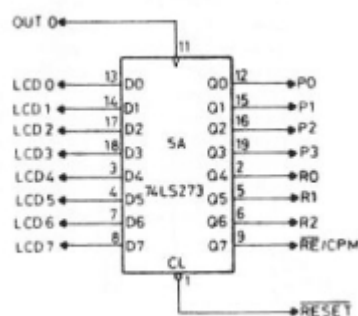
REMEMOrizer provides :-

- replacement for the 8KB OS ROM
- replacement for the 8KB BASIC ROM
- 8KB CP/M boot ROM
- 8KB SDX BASIC ROM
- 320KB of extra RAM
- 128KB virtual cassette tape space
- 32KB memory upgrade if attaching REMEMOrizer to an MTX500
- 64KB memory upgrade if attaching to MTX with no memory at all

The final bullet point is relevant if the user has tried to upgrade an MTX500, has removed the RAM and now has a functional MTX with no memory installed.

## MTX MEMORY MAPPING

The Z80 is an 8 bit processor with a 16 bit address bus, this allows the processor to directly access a total of 64K of memory ( $2^{16}$  bits). The processor also has a separate 64K of addresses for input (I) and output (O) devices. The addresses and data lines for memory and I/O instructions use the same physical wiring connections, the Z80 uses different control signals to define when a memory or I/O operation is being performed. The 64K memory address limit can be extended using a technique called bank or page switching. The operating system can select which page(s) are addressed at any given time. On the MTX computer, Output port 0 is the page port which uses the 74LS373 octal latch in position 5A on the system board :-



Bits D0 to D3 (P0 to P3) define the RAM page address

Bits D4 to D6 (R0 to R2) define the ROM page address

Bit D7 (RE/CPM) defines a ROM based system (D7 = 0) or a RAM based system (D7 = 1)

The paged memory map structure of the MTX computers was designed to operate in two modes :-

1. ROM Based (RELCPM = 0)

ROMs are mapped from 0 to 3FFFh. The 8K (2000h bytes) monitor ROM is always available in area 0 to 1FFFh and the paged ROMs of 8K (2000h bytes) each are mapped from 2000h to 3fffh as eight pages 0 to 7 set by R2,R1,R0 in the page port write only register. Up to 512K of RAM is mapped on 16 pages (0 to F) set up by P3,P2,P1 and P0 in the page port write only register. The area C000h to FFFFh is a 16K (4000h bytes) block common to all RAM pages. The 32K (8000h bytes) block from 4000h to BFFFh is mapped as 16 pages. The 32K bytes of RAM for an MTX500 is mapped from 8000h to FFFFh (page 0). The 64K bytes of RAM for an MTX512 is mapped from 4000h to FFFFh (page 0). The additional 16K is mapped from 8000h to C000h on page 1.

2. RAM Based (RELCPM = 1)

All ROMs are switched out in this mode and up to 16 pages of 48K (C000h bytes) are mapped from 0 to BFFFh. These pages are set by P3, P2, P1 and P0 in the page port write only register. In the area C000h to FFFFh is a 16K block (4000h bytes) of RAM common to all pages.

MEMORY MAP PICTURES

Default MTX ROM Based Memory Map, RELCPM = 0

R2,R1,R0	0x0000..0x1fff	0x2000..0x3fff	0x4000..0x7fff	0x8000..0xbfff	0xc000..0xffff
0	MONITOR  A  (OS)	SYS B (BASIC)	512	500 / 512	500/512  4000h  Bytes  Common  Block
1		SYS C (ASSEM)	(128a)	512	
2			(128c)	(128b)	
3			(128e)	(128d)	
4		CP/M boot ROM	(128g)	(128f)	
5		SDX ROM	(128h)		
6					
7		CARTRIDGE			
R2, R1, R0					
(128K Add-on to MTX512 shown in brackets (a-h))					

Default MTX RAM Based Memory Map, RELCPM = 1

0x0000..0x3fff	0x4000..0x7fff	0x8000..0xbfff	0xc000..0xffff	P3,P2,P1,P0
512	512	500 / 512	500 / 512	0
(128a)	(128b)	(128c)		1
(128d)	(128e)	(128f)	4000h	2
(128g)	(128h)		Bytes	3
			Common	4
			Block	5
				6
				7
				8
				9
				A
				B
				C
				D
				E
				F
(128K Add-on to	MTX512 shown	in brackets (a-h))		

## REMEMORIZER MEMORY MAP MODIFICATIONS

In these pictures, ROMs are shown with their names and are 8KB in size and RAM pages are assigned letters and are 16KB in size. RAM pages  $\alpha$  to  $\delta$  are the normal 64KB present in an MTX512. RAM pages a to t are extra pages, which are used as 320KB of RAM Disc.

MTX+REMEMOrizer logical memory map, as seen in RELCPMH=0 mode :-

R2,R1,R0	0x0000..0x1fff	0x2000..0x3fff	0x4000..0x7fff	0x8000..0xbfff	0xc000..0xffff	P3,P2,P1,P0
0	OS *	BASIC *	$\gamma$ **	$\beta$ ***	$\alpha$ ***	0
1		ASSEM	a	$\delta$ **		1
2			c	b		2
3			e	d		3
4		CP/M boot ROM	g	f		4
5		SDX ROM ****	i	h		5
6			k	j		6
7			m	l		7
			o	n		8
			q	p		9
			s	r		A
				t		B
						C
						D
						E
				any SRAM		any SRAM

\* These ROMs are the REMEMOrizer modified versions if the ROMs jumper is set "high", or if set "low" the MTX motherboard is expected to provide them.

\*\* These RAM pages are provided by REMEMOrizer if the MTX500 jumper is set "high", or if set "low" the MTX motherboard is expected to provide 64KB RAM.

\*\*\* These RAM pages are provided by REMEMOrizer if the MTX500 jumper is set "high", and the ROMs jumper is set "high" and REMEMOrizer notices that your MTX doesn't have any memory at all!

\*\*\*\* The first 6KB of this ROM contains regular SDX ROM code. The next 1KB has been stolen, and contains virtual tape code. The last 1KB is in fact writable, and contains virtual tape variables!

MTX+REMEMOrizer logical memory map, as seen in RELCPMH=1 mode :-

0x0000..0x3fff	0x4000..0x7fff	0x8000..0xbfff	0xc000..0xffff	P3,P2,P1,P0
δ **	γ **	β ***	α ***	0
a	b	c		1
d	e	f		2
g	h	i		3
j	k	l		4
m	n	o		5
p	q	r		6
s	t			7
				8
				9
				A
				B
				C
				D
				E
	any SRAM	any SRAM		F

In RAM page 15, it is possible to address any 16KB page of SRAM. Which page is visible at 0x4000..0x7fff is controlled by page register 1 (port 0xd0), and which page is visible at 0x8000..0xbfff is controlled by page register 2 (port 0xd1).

REMEMOrizer SRAM physical memory map :-

Address	Content
0x00000..0x03fff	RAM page α ***
0x04000..0x07fff	RAM page β ***
0x08000..0x0bfff	RAM page γ **
0x0c000..0x0ffff	RAM page δ **
0x10000..0x1ffff	virtual cassette area, read/write tape
0x20000..0x2ffff	virtual cassette area, read only tape
0x30000..0x7ffff	RAM pages a to t, RAM Disc area

This physical map is shown as it is what you see through the "any SRAM" windows in the normal memory map, when you set the page registers appropriately.

### PATCHABLE ROMS

Note that although REMEMOrizer provides images of ROMs, and although ROM stands for Read-Only Memory, there is in fact a backdoor allowing programs such as REZPATCH.COM to unprotect ROMs, write to them, and then protect them again.

## CASSETTE TAPE

By default REMEMOrizer does not load from or save to cassette tape. Almost all of the Memotech library on cassette has been converted into .MTX file format. Instead, REMEMOrizer supports "virtual cassette tapes".

"Virtual cassette tapes" are implemented as hidden areas of SRAM, per the diagram above. There are two virtual tapes, 64KB each.

Looking at the known library of Memotech cassettes, almost all of them will fit within 64KB.

The REMEMOrizer supplied OS ROM is patched to jump to virtual tape code at the end of SDX ROM 5. The last portion of this ROM is in fact writeable, and is where the virtual tape logic keeps its variables.

Virtual cassette tapes are accessed from CP/M using the REZTAPE command covered in Chapter 5.

(SW2 on the GODIL allows the user to switch between "Virtual" tape support and physical tape support. Refer to Chapter 3, Installation, to identify the position of switch SW2.)

## FLOPPY DISC CONTROLLERS

REMEMOrizer includes an SD Card. SD Cards between 64MB and 1GB are supported. Only 64MB of data may be stored on them. REMEMOrizer considers them to contain 8 8MB partitions. This is somewhat generous, as the entire Memotech software library will fit comfortably within one 8MB partition.

It accesses this using a Serial Peripheral Interface (SPI) interface. It has hardware support for driving the SPI interface so that byte transfer speed is effectively limited by the Z80. It has a novel feature in that reading data from SPI on one port triggers the sending of an 0xff byte to trigger the next transfer. This means that reading of data from SD Card needn't be twice as slow as writing it (as it would otherwise be).

Compared to REMEMOTECH, REMEMOrizers SD Card is slow. REMEMOTECH can transfer one byte in 21T cycles (at upto 25MHz), because the SPI logic is driven at 50MHz. REMEMOrizer can transfer one byte in 28T cycles (at 4MHz), because the SPI logic has to be driven from the CPU clock. Having said this, in a simple speed trial REMEMOrizer is still over 4x faster than SDX floppy, and massively faster than cassette tape.

Unfortunately the fact that CP/M sectors are 128 bytes and SD Card blocks are 512 bytes makes the whole thing somewhat inefficient. To read a 128 byte sector, we must read the enclosing 512 byte block. And to write a 128 byte sector, we must read the enclosing 512 byte block, modify a part of it, then write it back. Even with this handicap, it is still usable. Clever driver software helps improve things.

***A green LED flashes when SD Card is being accessed and for a couple of seconds afterwards, and the intent is that the user doesn't remove the SD Card until the LED goes off.*** This simple feature allows the SD Card driver code to go faster.

## 80 COLUMN CARD

REMEMOrizer implements a video card which is largely compatible with the original FDX 80 column card.

It outputs in 8 colours to VGA, 640x480 at 60Hz but does not RGB or Composite video, like the FDX did.

In addition to the normal 80x24 mode, it also supports 80x48 mode. To do this it has 8KB of memory, rather than 4KB.

It supports accesses to ports 0x30, 0x31, 0x32, 0x33, 0x38 and 0x39. Inputting from port 0x30 does not cause the bell to ring.

It emulates a subset of the 6845 CRTC registers (as per datasheet), specifically registers 10, 12, 13, 14 and 15. In addition, it has REMEMOrizer special register 31, in which bit 0 controls whether it is 80x24 or 80x48.

The normal Memotech alphanumeric font is present in on-chip ROM in the FPGA. The graphics characters are programmatically generated from the graphic character number, saving 2.5KB of scarce on-chip memory.

## SOUND

The Memotech sound chip is programmed by writing to port 6 and strobing the output by reading from port 3.

On a normal MTX, when you input from port 3, the result is 3. There is nothing in the circuit to explicitly ensure this is the case. Its just happens that way - the 3 is still on the bus from the opcode fetch, so thats the data which is returned. Wait a short while, and the bus will float high.

With REMEMOrizer the bus floats high much quicker, probably as a consequence of the FPGA pulling it high, so without any further intervention, inputting from port 3 returns 0xff.

This is a problem, because Pothole Pete and Son of Pete both input from port 3 and keep doing so until the result is 3. It is as if the programmers believed that getting 3 back indicates the operation was complete.

Accordingly, REMEMOrizer will explicitly drive the bus to the value 3 when you input from port 3.

This same hack appears in MEMU and REMEMOTECH.

## SOFTWARE

REMEMOrizer includes following software :-

- OS ROM
- BASIC ROM
- CP/M Boot ROM
- SDX BASIC ROM
- Tape Support
- CP/M Utilities
- Sample SD card

### OS ROM

The OS ROM sits in the fixed ROM page.

If the ROMs jumper is set "high", then REMEMOrizer supplies a patched version of the OS ROM which does not corrupt the RAM Disc and which has its cassette load/save routine patched to jump to the REMEMOrizer tape support.

If the ROMs jumper is set "low", then the motherboard ROM is used.

(Refer to Chapter 3, Installation, to identify the position of the ROMs jumper block.)

### BASIC ROM

The BASIC ROM sits in ROM page 0.

If the ROMs jumper is set "high", then REMEMOrizer supplies a copy of the BASIC ROM. The only reason it is included as in most MTXs, both the OS ROM and BASIC ROM are provided from a single 16KB ROM chip. So if you've removed this chip from the motherboard, and set the ROMs jumper "high", REMEMOrizer had better supply both the OS and BASIC ROMs.

If the ROMs jumper is set "low", then the motherboard ROM is used.

(Refer to Chapter 3, Installation, to identify the position of the ROMs jumper block.)



## CP/M BOOT ROM

The CP/M boot ROM sits in ROM page 4.

REMEMOrizer provides this ROM unless you press SW1. So to bypass booting to CP/M, press this switch whilst resetting the MTX.

(Refer to Chapter 3, Installation, to identify the position of switch SW1.)

It is different to the version found in an FDX or found in an SDX with CP/M support, and is pretty much the same as the REMEMOTECH version.

The CP/M Boot ROM is broken into a number of modules.

The **AZMON** module is the initial monitor. It is modified to boot F:52, B:18, B:19, ..., B:1F.

Drives B to 1F are the 8MB SD card partitions and have the same disc configuration

Drive F is configured as Type 52 – 320KB RAM disc

It always loads the RAM Disc support, whether you boot from it or not.

The **ASDXKBD** module is the keyboard driver. This is the same driver code used in REMEMOTECH, which queries the CPU clock divider by reading a port. On REMEMOrizer, that port always returns the value for 4MHz.

**ACRT** is the 80 column card driver. This has been modified so that it knows that the REMEMOrizer screen buffer has 4096 character cells (rather than the 2048 in a normal card). It also supports a new 80x48 mode as well as the 80x24 mode. It provides new escape codes ^[Y and ^[Z to switch into 80x24 or 80x48 mode.

**ASDCARD** is the replacement for the usual floppy disc driver code found in normal Memotech FDX and SDX systems. It supports a single 8MB size of disk, but through the use of 8 drive type codes (18..1F) allows you to access 8 8MB partitions of the SD Card.

The SD Card hardware turns on a green LED when the SD Card is accessed and flashes it for a couple of seconds afterwards, and the user should not remove or change the SD Card until it goes off. The driver reads a port which tells it that the card was accessed in the last second, and if so, it can assume that whatever sector it read last is still what it would get if it read that sector again. In this way, sequential reads of N 128 byte sectors only result in N/4 512 byte block reads (instead of N 512 byte block reads). Also, sequential writes of N 128 byte sectors only result in N/4 512 byte block reads and N block writes (instead of N 512 byte block reads and N 512 byte writes). This feature produces a measurable speed increase, perhaps as much as 2x.

The **ARAM** module provides RAM Disc support. It is a subset of the SIDISC module in the original Memotech computers, which supported Silicon Disc and RAM Disc. 256KB type 50 RAM Discs are supported. 512KB type 51 RAM Discs are not supported - there isn't enough memory. A special 320KB type 52 has been defined so as to make full use of the memory available on REMEMOrizer. Conveniently this exactly matches the type 03 floppy in size and geometry.

ARAM is present in memory, whether you boot from RAM Disc or not. There is no need for a SIDISC.COM program.

## SDX BASIC ROM

The SDX BASIC ROM sits in ROM page 5.

The SDX ROM in a real SDX provides USER commands to MTX BASIC, allowing the user to read and write files on a floppy disk. The SDX ROM in REMEMOrizer is pretty much the same as the REMEMOTECH version.

Instead of floppy disc, it supports SD Card partition 0. Unlike the original SDX ROM, this version also supports 320KB RAM Disc. In other words, the drive configuration supported is B:18,F:52.

To support these disk types, it was necessary to steal areas of high memory that would otherwise be used by the NODE RING ROM. This support is therefore incompatible with the NODE RING ROM.

Remember, to use the SDX support, you must first use ROM 5. The first drive (SD Card partition 0) must contain a 59K CP/M system. If this is not the case, you'll get a DISC ERROR.

If you try to use USER FORMAT, you will get a Mistake error. Track and sector based formatting doesn't make sense for SD Card or RAM Disc media. Instead use the REFORMAT.COM CP/M executable.

## TAPE SUPPORT

The tape support is physically present in the SDX ROM, and the OS ROM is patched to jump there.

By default, REMEMOrizer loads and saves to virtual cassette, as administered using the REZTAPE.COM command.

However, if you press SW2, then a little red LED lights up to indicate that virtual tape support is disabled. In this mode, real cassette load and saving occurs. This can happen because the code at the end of SDX ROM 5 also includes a copy of the code normally present in the OS ROM.

(Refer to Chapter 3, Installation, to identify the position of switch SW2.)

(Refer to Chapter 4, Operation, for a description of the use of SW2 to select the tape option.)

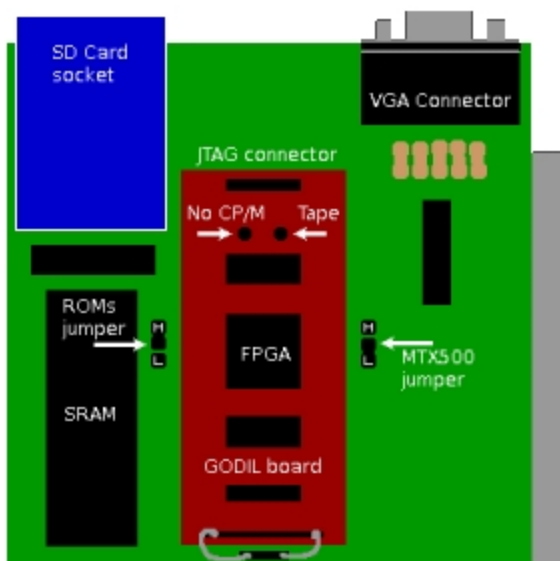
### 3. INSTALLATION AND SETUP

#### REMEMORIZER CONTENTS

If your REMEMOrizer was supplied by the designer, your REMEMOrizer should have been supplied with :-

- The REMEMOrizer add-on card for your MTX
- Replacement OS and BASIC ROM chips (usage is optional)
- Replacement for PAL chip in MTX board position 6A
  - GAL marked "2\_32" for an MTX500 with 2 ROM chips (MTX500 only)
  - GAL marked "3\_32" for an MTX500 with 3 ROM chips (MTX500 only)
  - GAL marked "2\_64\_S2" for an MTX512S2 only
- If you have an RS232 board, a replacement GAL marked "serial" for the RS232 PAL, (IC8)
- SD Card containing REMEMOrizer utilities and other MTX & CP/M software

#### SCHEMATIC DIAGRAM OF REMEMORIZER BOARD



#### ROMs Jumper

H = REMEMOrizer will provide replacement ROMs.

(The existing ROMs *must* be removed from the MTX main board)

L = Use existing MTX ROMs

#### MTX500 Jumper

H = Set if you have an MTX500 (32K RAM)

L = Set if you have 64K RAM

***Please read through to the end of this chapter and understand it before connecting REMEMOrizer to your MTX. Serious damage could occur to your MTX and/or REMEMOrizer if you configure the jumper settings or ROMs incorrectly!***

If you have any questions, **STOP**, and contact the designer for help before you begin.

Andy Key can be contacted by e-mail at [andy.z.key@gmail.com](mailto:andy.z.key@gmail.com)

## PREPARATION

### DECIDING ON WHETHER TO USE THE REPLACEMENT ROMS

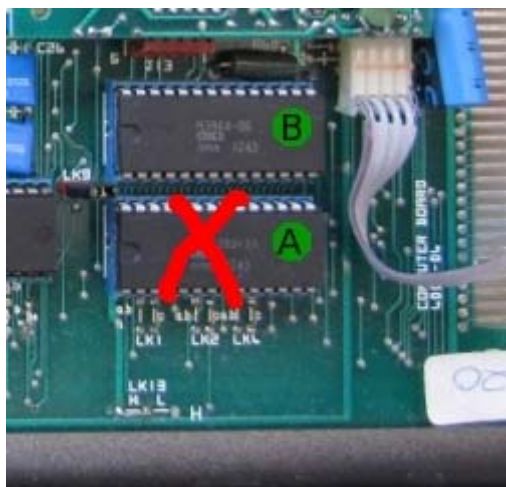
Replacement of the existing ROMs is optional, however, the memory test routine in the OS ROM corrupts the RAM disc on start-up and the cassette tape routines access the real tape I/O. If you don't remove the existing ROMs containing OS and BASIC and use the replacements supplied with REMEMOrizer, your RAM Disc can be corrupted at boot time, and you won't have the ability to use virtual cassette tapes.

If you remove the ROMs containing OS and BASIC, set the ROMs jumper "high". REMEMOrizer will provide replacements. These don't corrupt the RAM Disc and are modified so that the cassette load/save routines access virtual cassette tapes.

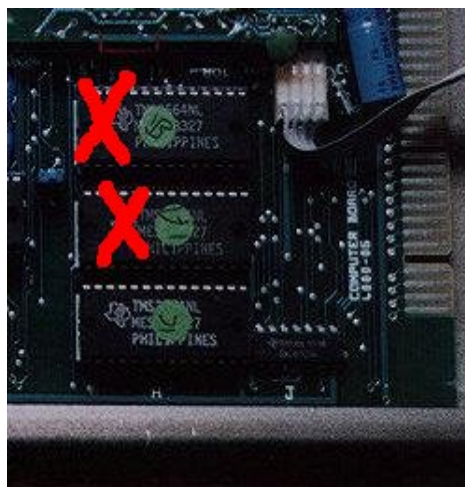
**Warning: Do not set the jumper "high" if your MTX still has the ROMs containing OS or BASIC. Both the motherboard and REMEMOrizer will attempt to drive the bus when these ROMs are accessed, which could result in permanent damage to the MTX and/or REMEMOrizer.**

If you do not wish to use the replacement ROMs, ensure that the ROMs jumper is in the "L" position and leave the existing ROMs in place.

If you do wish to use the replacement ROMs, you need to identify which version of the MTX computer board you have :-



4004 Computer Board



4005 Computer Board

ROM Name	Location	Size	Contents	ROM Name	Location	Size	Contents
ROM B	8H	8 KB	ASSEM	<b>ROM B</b>	8H	8 KB	BASIC
<b>ROM A</b>	9H	16 KB	OS&BASIC	<b>ROM A</b>	9H	8 KB	OS
	10H		empty	ROM C	10H	8KB	ASSEM

If you remove the ROM(s) containing OS and BASIC, set the ROMs jumper "high".

If you don't remove ROMs containing OS and BASIC, set the ROMs jumper "low".

## MTX500 JUMPER

Earlier MTX500s had a memory decoder PAL which made pages of the 32KB of memory appear in more than one place in the memory map. You can replace the PAL with a GAL which ensures the pages appear only once. Later MTX500s had a PAL which did this, and its entirely safe to make this change whether or not you plan to use REMEMOrizer. But REMEMOrizer can only work with MTX500s that only present the pages once. Therefore, if you have a MTX500, replace the PAL with the 2\_32 or 3\_32 GAL supplied with REMEMOrizer. The 2\_ or 3\_ prefix refers to how many ROMs you have. Even if you don't always use REMEMOrizer, you can leave this new GAL installed.

Technically, if you know you have the newer variety of MTX500, you needn't replace the PAL, but rather than describe the process of telling the difference, and expecting the user to follow it, its simpler to change the chip and be sure.

If your MTX only has 32KB of RAM, set the MTX500 jumper "high". This causes REMEMOrizer to provide a RAM upgrade, using 32KB of its SRAM.

If your MTX has 64KB of RAM, set the MTX500 jumper "low".

***Warning: Do not set the jumper "high" if your MTX has 64KB of RAM, or if you have not replaced the PAL in your MTX500. Both the motherboard and REMEMOrizer will attempt to drive the bus when the top 32KB of RAM is accessed, which could result in permanent damage to the MTX and/or REMEMOrizer.***

Note: If, have a partially upgraded MTX500, you may have an MTX000, i.e.: a MTX with no memory at all! If the MTX500 jumper is "high" and the ROMs jumper is also "high", then the modified OS ROM will notice the lack of memory and tell REMEMOrizer to also provide the lower 32KB RAM.

## MTX512 S2

If you have an MTX512 S2, then your motherboard will have 4256 DRAMs and a different memory decoder PAL. This PAL makes 64KB available in RELCPMH=0 mode and 208KB available in RELCPMH=1 mode. 144KB of the memory in RELCPMH=1 mode will clash with memory provided by REMEMOrizer.

Therefore, if you have a an MTX512 S2, replace the PAL with the 2\_64\_s2 GAL supplied with REMEMOrizer. As all MTX512s had 2 ROMs, there is no 3\_64\_s2 GAL. This GAL only makes 64KB memory available in RELCPMH=1 mode.

***Warning: Do not attach REMEMOrizer to an MTX512 S2 if you have not replaced the PAL. Both the motherboard and REMEMOrizer will attempt to drive the bus when high memory is accessed, which could result in permanent damage to the MTX and/or REMEMOrizer.***

## RS232

The RS232 card provides two functions. It provides RS232 ports, and it acts as a bridge to an FDX box (even if you don't have one connected). As a consequence, if the MTX tries to read ROM 4 or ROM 5, or input from ports 20H or higher, or if it detects DMA from the FDX, it will drive the MTX data bus. This is clearly incompatible with an SDX, and its incompatible with REMEMOrizer too. An SDX and REMEMOrizer have their own ROMs and ports at these locations.

Therefore, if you have an RS232 card, remove its PAL (IC8 on the RS232 board), and replace it with the serial GAL supplied with REMEMOrizer. This prevents the RS232 card from acting as a proxy for an FDX which doesn't exist.

***Warning: Do not attach REMEMOrizer to an MTX with an RS232 card if you have not replaced the PAL. Both the RS232 card and REMEMOrizer will attempt to drive the bus at the same time, which could result in permanent damage to the RS232 card and/or REMEMOrizer.***

Note: Should you want to disconnect REMEMOrizer and reconnect to an FDX, the replacement PAL mustbe removed and replaced with the old RS232 PAL.

## PLUG IT IN

Double check the ROM and MTX500 jumper positions as well as the ROM and GALs fitted.

Connect the REMEMOrizer board to the connector on the left hand side of the MTX.

Connect a VGA monitor to the 15 pin D-sub connector at the top right of the REMEMOrizer.

## 4. OPERATION

### POWER-ON

The default behaviour of REMEMOrizer is to boot up in CP/M mode, however, upon power on, the FPGA on the GODIL on the REMEMOrizer takes two seconds to initialise, and in that time, the MTX will not see any of the ROM images it provides. As a result, the MTX will almost certainly not boot CP/M. Two red LEDs are lit during initialisation, and when complete, then they turn off and a green LED lights instead. Just press the Reset keys after this has happened, and the MTX should see the ROMs and boot fine.

### RESET

Pressing SW1 makes the CP/M boot ROM vanish from the memory map. So if you do this whilst resetting, the MTX will boot directly to MTX BASIC, not to CP/M. SW1 is the button on the GODIL marked "No CP/M" in Chapter 3.

The same effect can be achieved with the latest REMEMOrizer by pressing both Shift keys during resetting.

### RAM DISC

The sample SD Card contains a CP/M script which sets up the RAM Disc the first time. To use :-

```
A>SUB RAMDISC
```

After running it, reset to boot from RAM Disc.

### SD CARD

Each SD Card can have up to 8 8MB partitions. Each of these would contain a CP/M 2.2 filesystem.

A green LED lights up when REMEMOrizer is reading or writing the SD Card. Then it flickers and turns off. **Wait for it to turn off before removing or changing the SD Card.**

### SAMPLE SD CARD

The package includes a couple of sample SD Card images, in the file sddisc.bin and sddisc2.bin. They are 8MB SD Card partitions with CP/M 2.2 filesystems. The first has a 59K CP/M system on it, the second has 54K. If you copy them to the first and second 8MB of a SD Card, you can access them by configuring type 18 and type 19 drives.

These images have some handy utility files.

The first image has various games in .COM format. There are also a few games in .RUN file format, most of which can be run using the RUN.COM program. There are also some SDX BASIC .BAS files. It also has lots of .MTX files.

The second image has Icicle Works, which needs a 54K CP/M system to work. I've never got to the bottom of why this is, and it is tricky to debug.

- Interrupt the CP/M bootstrap by holding down <Enter>
- Then type BC19 to boot from the second partition
  - Then run DT.COM followed by TLOAD.COM followed by TV.COM
  - Or type SUB ICICLE.

### ACCESSING SD CARDS FROM CP/M

You'll need to use RECONFIG.COM (see Chapter 5 for details) to configure a drive (or drives) to access partitions on the SD Card. Type codes 18..1F correspond to 8MB partitions 0 to 7 on the card. eg: To access partitions 0, 1 and 2 :-

```
A>RECONFIG B:18,C:19,D:1A
```

### ACCESSING SD CARDS FROM MTX BASIC

You can switch to MTX BASIC, and use ROM 5 to gain access to disk via USER commands.

```
A>MTXL
```

```
ROM 5
```

```
USER DIR
```

```
USER RUN "TOADO.RUN"
```

### ACCESSING SD CARDS ON LINUX

To access SD Card partitions from Linux, use the cpmttools package, and ensure /usr/share/cpmttools/diskdefs contains the Memotech CP/M filesystem disk definitions from the MFLOPPY package.

Although we RECONFIG drives as types 18..1F (depending on what partition we want the drive to map to), all of these types are 8MB in size and have the same disk configuration (ie: the same CP/M DPB). Also cpmttools has no concept of "offset from the disk start". The diskdefs file therefore only has a single type 18 definition, and in the examples that follow, we always refer to -f memotech-type18.

To read partitions, I use commands like this :-

```
$ su
# dd if=/dev/sde of=partition0.dat bs=8M skip=0 count=1
```



```

# dd if=/dev/sde of=partition1.dat bs=8M skip=1 count=1
# dd if=/dev/sde of=partition2.dat bs=8M skip=2 count=1
# ^D

$ cpmls -f memotech-type18 partition0.dat
$ cpmls -f memotech-type18 partition1.dat
$ cpmls -f memotech-type18 partition2.dat

```

To write them, I use commands like this :-

```

$ mkfs.cpm -f memotech-type18 -b boot59.dat partition0.dat
$ mkfs.cpm -f memotech-type18 -b boot54.dat partition1.dat
$ mkfs.cpm -f memotech-type18 -b boot54.dat partition2.dat
$ cpmcp -f memotech-type18 partition0.dat SOMEFILE.COM 0:
$ cpmcp -f memotech-type18 partition1.dat SOMEMORE.COM 0:
$ cpmcp -f memotech-type18 partition2.dat EVENMORE.COM 0:
$ su
# dd if=partition0.dat of=/dev/sde bs=8M seek=0
# dd if=partition1.dat of=/dev/sde bs=8M seek=1
# dd if=partition2.dat of=/dev/sde bs=8M seek=2
# ^D

```

Although the examples show multiple partitions, the whole Memotech software library probably fits within just one.

## ACCESSING SD CARDS ON WINDOWS

If you must use Windows, then these ports of dd and cpmttools are useful.

I put dd.exe in a directory on the PATH, and I unzip cpmttools so that it ends up in the C:\cpmttools directory.

The first thing to do is to determine which block device is your SD Card. You don't want to accidentally write to your hard disk. Insert the SD Card and type :-

```
C:\ak\tmp>dd --filter=removable --list
rawwrite dd for windows version 0.5.
Written by John Newbigin <jn@it.swin.edu.au>
This program is covered by the GPL. See copying.txt for details
```

### *NT Block Device Objects*

```
\\?\Device\Harddisk2\Partition0
  link to \\?\Device\Harddisk2\DR7
\\?\Device\Harddisk3\Partition0
  link to \\?\Device\Harddisk3\DR8
\\?\Device\Harddisk4\Partition0
  link to \\?\Device\Harddisk4\DR13
  Removable media other than floppy. Block size = 512
  size is 1015808000 bytes
```

The `--filter=removable` ensures the list returned only includes removable devices. You're looking for the one around the right size. In this case, as I'm using a 1GB SD Card, its clearly `\\?\Device\Harddisk4\Partition0`.

To read one of the partitions (0-7), use the following command. The partition number is the skip= value, and partitions 0 to 7 correspond to CONFIG types 18 to 1F :-

```
C:\ak\tmp>dd if=\\?\Device\Harddisk4\Partition0 of=partition0.dat
bs=8M skip=0 count=1
rawwrite dd for windows version 0.5.
Written by John Newbigin <jn@it.swin.edu.au>
This program is covered by the GPL. See copying.txt for details
1+0 records in
1+0 records out
```

The following command adds cpmttools to your PATH :-

```
C:\ak\tmp>\cpmttools\cpmttools
```

The diskdefs supplied with it doesn't understand Memotech disk formats. So save it away, and replace with the file from my MFLOPPY package :-

```
C:\ak\tmp>copy \cpmttools\diskdefs \cpmttools\diskdefs.saved
1 file(s) copied.
```

```
C:\ak\tmp>copy \ak\mtx\mfloppy\diskdefs \cpmttools\diskdefs
```

```
Overwrite \cpmtools\diskdefs? (Yes/No/All): Y
1 file(s) copied.
```

Warning: Don't be tempted to edit the diskdefs file. The cpmtools package expects each line to end with just a line-feed character (as in UNIX), and most Windows text editors will terminate each line with carriage-return and line-feed (as in DOS and Windows). As a result, cpmtools will complain it doesn't recognise disk formats, even though they are in the diskdefs file. I view this as a bug in cpmtools.

To list the partition content, copy in a new file, and list it again :-

```
C:\ak\tmp>cpmls -fmemotech-type18 partition0.dat 0:a*.run
0:
alpha.run
astropac.run
```

```
C:\ak\tmp>cpmcp -f memotech-type18 partition0.dat AGROVATA.RUN 0:
```

```
C:\ak\tmp>cpmls -f memotech-type18 partition0.dat 0:a*.run
0:
agrovata.run
alpha.run
astropac.run
```

To write the partition back onto the SD Card, use :-

```
C:\ak\tmp>dd --filter=removable if=partition0.dat
of=\\?\Device\Harddisk4\Partition0 bs=8M seek=7
rawwrite dd for windows version 0.5.
Written by John Newbigin <jn@it.swin.edu.au>
This program is covered by the GPL. See copying.txt for details
1+0 records in
1+0 records out
```

Pay careful attention to the fact I use -filter=removable as a safety measure to protect against writing to my harddisk, and the right of= value.

Also notice that the example has seek=7 meaning we actually wrote to partition 7, and so would have to RECONFIG E:1F and DIR E: to see the files.

## TAPE OPERATION

### TAPE MODE SELECTION

REMEMOrizer allows the user to switch between the Virtual Tape mode and Physical Tape mode using SW2 on the GODIL board.

If you press SW2, then a little red LED lights up to indicate that virtual tape support is disabled. In this mode, real cassette load and saving occurs. This can happen because the code at the end of SDX ROM 5 also includes a copy of the code normally present in the OS ROM.

### TAPE BREAKER

Tape support has a feature called *tape-breaker*. During LOADING, hold down SW1. After each block is loaded, the border will go yellow. Press <INS> to proceed as normal, in which case the border flashes green. Press <ENT> <CLS> to break in at that point, in which case the border flashes red, and the break entry point in the BASIC ROM is entered.

One way to use this feature is to LOAD the program twice. The first time, simply press INS each time and count how many blocks are loaded before the program starts. The second time, press INS one fewer time, and then ENT CLS for the last time. Hopefully there is enough of the BASIC program there to inspect.

If you are loading from .mtx file, you can also use MEMU with its -diag-console and -diag-tape options to discover how many blocks there are in your program.

Because of these features, it may now be easy to LOAD from tape and then SAVE to virtual tape, and vice-versa. Of course, games that employ auto-run and other copy-protection schemes may still prove difficult.

## 80 COLUMN SCREEN MODE SELECTION

The REMEMOrizer screen buffer has 4096 character cells (rather than the 2048 in a normal card). It supports a new 80x48 mode as well as the 80x24 mode. The user can switch between the 24 line and 48 line screen modes by entering escape codes ^[Y and ^[Z.

This is particularly useful when running NewWord in 48 line mode using N48.COM.

## 5. CP/M UTILITIES

### RECONFIG.COM

RECONFIG.COM is a version of CONFIG.COM, but enhanced for REMEMOTECH and REMEMOrizer. It gives give better error messages and has been enhanced to support extra drive types. It supports type 18..1F drives, which are 8MB SD Card partitions. Do not configure two drives with the same 1x type code, as this can cause data corruption. It also supports type 52 drives, which are 320KB RAM Discs.

Every time you configure or change a drive type, high memory is allocated for the allocation and check vectors. High memory is in short supply, and after configuring and reconfiguring lots of drives, you can run out. This issue can eased by using a 54K system disk, rather than 59K. This is made worse in REMEMOTECH and REMEMOrizer, as in order to access more of the SD Card, you may find yourself configuring and reconfiguring a drive to access different partitions on it. The included RECONFIG.COM is enhanced to spot reconfiguring a drive to a new type code with the same DPB (which implies the same geometry), and if so, it reuses the existing allocations and check vectors.

The new RECONFIG.COM is a superset of the original shipped by Memotech, and can actually be used on earlier Memotech systems.

### REFORMAT.COM

Back in the day, there were 3 different formatting programs :-

FORMAT.COM - for FDX floppy disks, hard disks and Silicon Discs

FRMSDX.COM - for SDX floppy disks

SIDFMT.COM - for Silicon Disc or RAM Disc on either FDX or SDX

REMEMOrizer includes REFORMAT.COM which doesn't issue hardware level formatting commands, and instead it overwrites the reserved tracks and directory area with sectors containing 0xe5 bytes. This can therefore be used to "format" the RAM Disc and SD Card found on REMEMOrizer. None of the original formatting programs are therefore needed, none of them work, and so none of them are included.

REFORMAT.COM can be used on an original Memotech system to wipe a disk, providing the disk is already formatted to the right type code.

## REZTAPE.COM

REZTAPE.COM is a program for managing the virtual cassette tapes feature in REMEMOrizer. It is much like the RETAPE.COM program used to do the same thing on REMEMOTECH.

There is one read/write 64KB SRAM slot (slot number 0) and a read-only 64KB SRAM slot (slot number 1).

The following commands can be used :-

### REZTAPE

List bytes 1-15 of each virtual cassette tape in each slot. ie: the name to LOAD from MTX BASIC.

### REZTAPE *SLOT*

Delete the contents of the indicated slot.

### REZTAPE INPUT.MTX *SLOT*

Read file INPUT.MTX and write it into the indicated slot. Note: The file cannot be larger than the size of the slot.

### REZTAPE *SLOT* OUTPUT.MTX

Read the indicated slot and write to a file called OUTPUT.MTX. Note: The resulting file will be the size of the slot, regardless of how many bytes have been saved to that virtual cassette tape.

### REZTAPE ?

Show help text, summarising the above.

The INOUT routine in the MTX BASIC ROM has been patched so that :-

LOAD "" Loads from slot 0.

LOAD "NAME"

Looks for NAME in slots 0 to 1 and loads it. If no slot has this name, then a "No data" error results (a real Memotech MTX just keeps reading hoping that it will eventually find a saved file with this name, but REMEMOrizer knows it is not present and so gives up).

VERIFY "" Verifies against the data in slot 0.

VERIFY "NAME"

Looks for NAME in slots 0 to 1 and verifies against it. If no slot has this name, then a "No data" error results.

SAVE "NAME" Saves to slot 0.

The recommended approach to use is to use REZTAPE.COM to write the game you wish load to a tape slot, switch to MTX BASIC, and the LOAD it. If you save anything from MTX BASIC, boot to CP/M afterwards and use REZTAPE.COM to read slot 0 and write to a .MTX file.

For example, prior to switching to MTX BASIC :-

```
A>REZTAPE QUAZZIA.MTX 0
A>REZTAPE SNAPPO.MTX 1
A>MTXL
```

Then the usual MTX BASIC LOAD command can be used to load it :-

```
LOAD "QUAZZIA"
```

## REZPATCH.COM

REZPATCH.COM is a program which can patch the ROM images that REMEMOrizer provides. ie: The fixed OS ROM, BASIC ROM 0, CP/M boot ROM 4 and SDX ROM 5.

Its usage is like this :-

```
A>REZPATCH iobyte address {byte}
```

The change lasts until power-off, or next change.

As an example, to make MTX BASIC say "Randy", rather than "Ready", use :-

```
A>REZPATCH 00 026D 52 61 6E 64 79
```

**Caution: Use with care.**

## MTX.COM, MTXL.COM AND RUN.COM

REMEMOrizer also has the MTX.COM and MTXL.COM programs, which can be used to switch into MTX BASIC. MTXL.COM is preferred, as it does not corrupt RAM Disc in the transition.

RUN.COM is a CP/M program which can be used to run some .RUN files, without having to switch into MTX BASIC, and use the SDX ROM to USER RUN "FILE.RUN" them. This program puts the file content directly into memory, switches into the right memory mode, and jumps to the start of the program. This means that MTX BASIC has never been running, and has not set up any of its system variables. As a result, some .RUN files may not run properly this way.

## 6. INCOMPATIBILITIES

There are a small number of incompatibilities between REMEMOrizer and real Memotech hardware.

- CP/M programs which directly do low level disk configuration or disk I/O, directly to the hardware itself, such as FORMAT.COM, FRMSDX.COM, SIDFMT.COM, RCHECK.COM and FDXB.COM.
- Programs which do direct access to the cassette tape ports, will never be able to use virtual cassette tape, such as certain games loaders. They will of course attempt to use real cassette tape.
- REMEMOrizer is incompatible with the NODE ROM. This is because the SDX support uses memory normally used by the NODE ROM for SD Card and RAM Disc variables and buffers
- REMEMOrizer is incompatible with the FDX, HDX or HRX, in exactly the same way that an SDX is incompatible with them.



## INDEX

24 line.....	28	OS ROM .....	9, 14, 16, 18, 20, 21, 31
48 line.....	28	page port.....	9, 10
80 column card.....	7, 8, 15, 17	Power-on.....	23
Accessing SD Cards from CP/M .....	24	RAM.....	8, 9
Accessing SD Cards from MTX BASIC .....	24	RAM Disc 7, 12, 13, 16, 17, 18, 20, 23, 29, 31, 32	
Accessing SD Cards on Linux .....	24	RECONFIG.COM .....	24, 29
Accessing SD Cards on Windows .....	26	REFORMAT.COM.....	18, 29
ACRT .....	17	RELCPM .....	9, 10, 11
ARAM.....	17	Reset.....	23
ASDCARD .....	17	REZPATCH.COM .....	13, 31
ASDXKBD.....	17	REZTAPE.COM .....	18, 30
AZMON .....	17	ROMs jumper .....	20
BASIC ROM .....	7, 9, 16, 18, 19, 30, 31	ROMs Jumper .....	19
Cassette tape .....	8	RS232 .....	19, 22
CP/M Boot ROM.....	16, 17	RUN.COM.....	31
CP/M Utilities .....	29	SD card .....	8, 16
cpmtools.....	26	SD Card 7, 14, 17, 18, 19, 23, 24, 26, 27, 29, 32	
dd .....	26	Software .....	16
Floppy Disc.....	8, 14	Sound.....	8, 15
GODIL .....	8, 28	SW1 .....	17, 23
Incompatibilities .....	32	SW2 .....	14, 18, 28
MTX.COM .....	31	Tape .....	14, 16
MTX500 Jumper .....	19, 21	TAPE SUPPORT .....	18
MTX512 S2 .....	21		
MTXL.COM .....	31		