AN INTRODUCTION TO DRAGON MMC

by
Phill Harvey-Smith
and
Richard Harding

A Manual for the Dragon MMC SD Card Interface

Acknowledgements

I would like to thank all the people who have helped along the way to making the DragonMMC a reality, these include:

Charlie Robson, who allowed me access to the source for his AtoMMC interface for the Acorn Atom, on which the AVR microcontroller code for the DragonMMC interface is based.

Simon Hardy & the other members of Retro Computer Museum Leicester for doing some initial testing when I first started the project. Simon also did a sterling job demonstrating the later board at the 2017 Dragon meeting.

Bas Gialopsos and Richard Harding for exhaustive testing and feedback. This has led to the eradication of a whole load of major bugs in the interface firmware. Not to mention the implementation of several facilities that had not occurred to me.

Richard Harding, for writing the documentation and taking a major task off my hands.

Chris Poacher for the DMMC logo concept.

Finally, an obligatory shout out for my wife Lesley, for putting up with me talking about retro computers to her for all these years ☺

Phill Harvey-Smith
2017

CONTENTS

INTRODUCTION
WELCOME

DMMC for the Dragon and CoCo in the modern age

When the Dragon and CoCo were in production, it was science fiction that every piece software ever released for them could be stored on a 'solid state' drive with no moving parts.

!!! Science fiction has now become science fact !!!

The Dragon MMC allows a massive software catalogue to be available at your fingertips that can be loaded in a matter of seconds.

Don't' forget to keep up to date with new firmware etc. at the following address:
> http://www.archive.worldofdragon.org/index.php?title=DragonMMC_SDCard

To simplify the use of the DMMC an 'Explorer' can also be found at the address above.

DMMC Hardware Overview

The Dragon MMC is compatible with the following computers:-
- a) Dragon 32
- b) Dragon 64*
- c) Draco 64*
- d) Radio Shack (Tandy) CoCo 1**
- e) Radio Shack (Tandy) CoCo 2**

*As the DMMC relies on the cartridge/expansion memory area, it is only compatible in 32K mode.
** Extended Colour BASIC is required and 32K is recommended.

The DMMC board has the same dimensions as an original Dragon Data DragonDOS board, this means that with minor modification to the DragonDOS case the DMMC can be fitted.

Dragon / CoCo SD/MMC
Ramoth
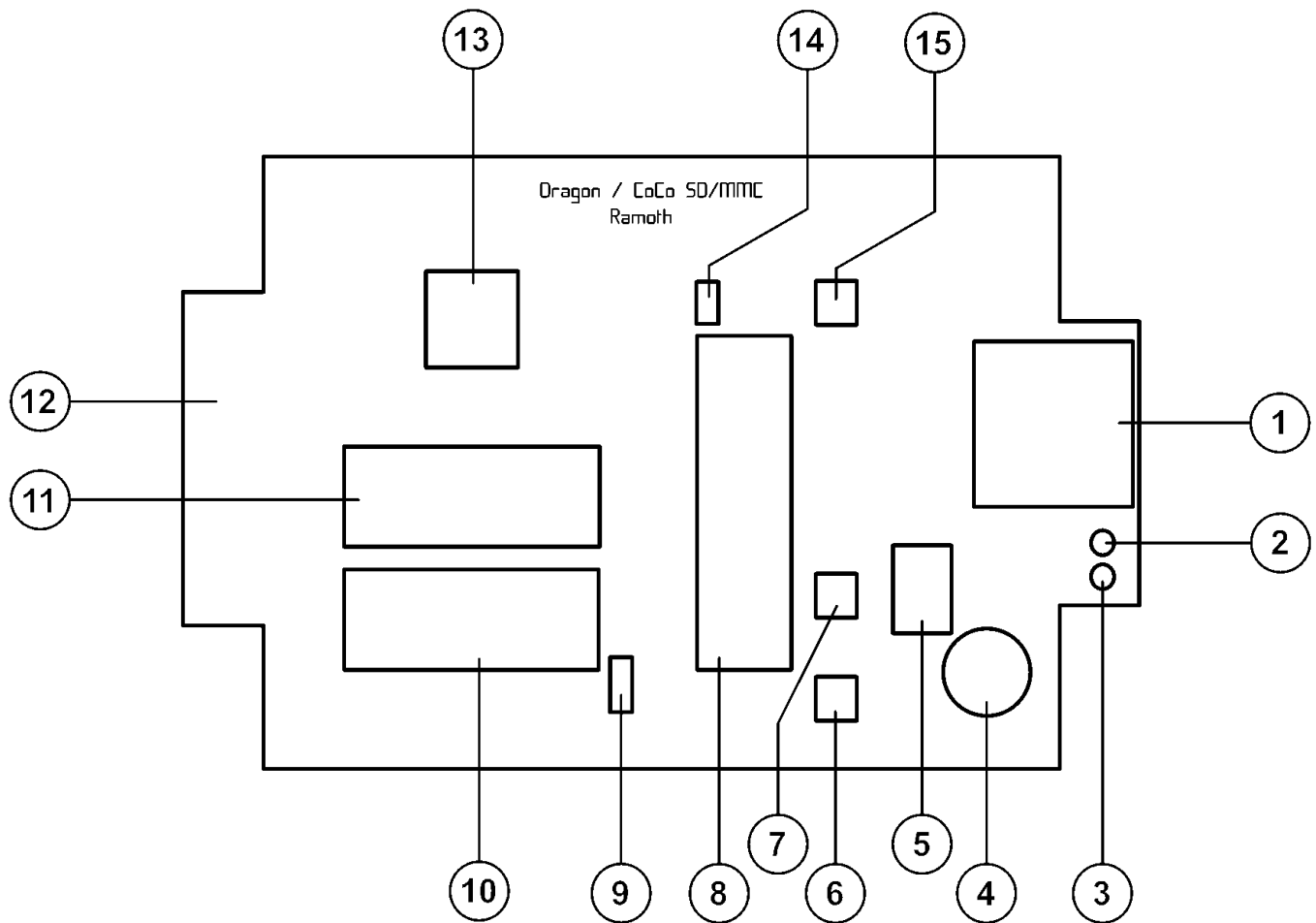
KEY

1. SD/MMC Card Slot
   Insert SD or MMC card here!

2. Red LED
   Lit when DragonMMC board being accessed

3. Green LED
   Lit when MMC/SD card being accessed.

4. Real Time Clock Battery
   Replace with CR1220 or equivalent.

5. Real Time Clock
   Provides timekeeping for the host computer and allows files to be correctly timestamped.

6. Snap Shot Button
   Accesses the NMI menu, allows you make and load snapshots of the active RAM state.

7. AVR Reset Button
   Resets the AVR microcontroller.

8. AVR
   Microcontroller that reads and writes the SD card and clock. Has embedded firmware to read and write a FAT/FAT32 Filesystem and convert it into a format that can be used by the host computer.

9. Jumper 2
   Write protect jumper for EEPROM / FLASH holding 6809 firmware (DEV ONLY)

10. EEPROM
    Holds the 6809 firmware for both the Dragon & CoCo

11. SRAM
    Holds mirrored copy of Dragon & DragonMMC ROMs to allow them to be patched, without any modification to originals, cartridge images are also loaded in here

12. Cartridge Edge Connector
    Plugs into the computers cartridge slot.

13. CPLD
    Handles the interface between the BUS of the 6809 and the AVR. Also does the address decoding for the ROM and RAM, captures writes to the SAM so they can be saved in snapshots.

14. CPLD Program Jumper
    Allows in circuit programming socket of the AVR to update bootloader. (DEV ONLY)

15. Force Cold Reset Button
    Pressing the Force Cold Reset Button whilst pressing the computers reset button will perform a cold reboot when running ROM (cartridge) images.

CHAPTER ONE

GETTING THE SHOW ON THE ROAD

Connecting Your DMMC Interface

Please ensure that *before* inserting or removing anything in the cartridge slot that the computer is switched *off*. Failure to do so may damage both the DMMC and your computer.

Connect your computer to the TV/monitor as normal. Before switching on the computer, insert the DMMC interface into the cartridge slot with the label facing upwards (chips facing up if not in a case). When inserted the DMMC should be flat and level, if using the DMMC without a case ensure it is supported and cannot be knocked or otherwise moved or permanent damage may result.

The computer screen should first show the usual copyrights messages, followed by DMMC messages confirming the DMMC is connected. If it does not, switch off immediately, check the computer connections and that the DMMC is inserted into the cartridge slot correctly. The copyright messages should be followed by OK and the usual cursor.

It's About Time

The DMMC contains a real time clock (RTC) which allows files saved on the computer onto the memory card to be correctly timestamped. The RTC can also be displayed at boot time and used in BASIC.

The RTC can be displayed from BASIC:-

    PRINT MGETDT$
    2017-04-01 11:26:57

To get the most from the DMMC, set the clock to the correct time using the format: MSETDT "YYYY-MM-DD hh:mm:ss"

    MSETDT "2017-04-02 10:27:00"

Note: you must use a leading zero for values less than 10!

The MGETDT$ function can also be used in a program:-

```
10 DT$= MGETDT$
20 PRINT "CURRENT DATE AND TIME IS:"
30 PRINT DT$
```

Setting Up A New Memory Card

The DMMC takes a standard full sized MMC, SD and SDHC cards and has been tested with multiple quality brands (SanDisk etc.), sizes and speeds. Larger capacity SD cards are aimed at fast modern equipment with multimedia in mind, if you are experiencing any issues, with reading or writing to the memory card, try a 4GB or less SD/MMC card (not SDHC or SDXC).

The use of mini and micro cards in adaptors is *not* recommended.

Don't forget: Dragon and CoCo disk images and especially cassette images are *tiny* in comparison to modern storage needs, so 1GB can be overkill.

Format the memory card using FAT or FAT32 and it's ready to go, pop on your favourite disk (DSK / VDK), cassette images (CAS) or cartridge images (ROM) on and get ready to load your favourite software in seconds.

Due to the limitations of the Dragon screen *vs* a modern computer, it's recommended that the amount of files in a folder is limited to maintain usability (ideally < 255). This can be done by dividing games etc. into sub folders, for example by publisher (e.g. DRAGDATA) or alphabetically (e.g. A).

What's On My Memory Card?

Insert your memory card into the card slot with the label facing up. As the DMMC supports hot swapping of memory cards this can be done with the DMMC on or off.
The command CAT will list files and folders in a directory on the memory card.

    CAT

The full title of a file is in the form of a short filename type (8.3):
    filename.filetype
where *filename* is the user-assigned filename (up to 8 characters) and *filetype* is a code (up to 3 characters) used to distinguish types of files.

Whilst the DMMC can load/save to any file type is it recommended that the following file extensions are used to distinguish different files:-

BAS	BASIC programs saved direct to memory Card (MSAVE)
BIN	Machine Code programs/data saved direct to memory Card (MSAVE)
CAS	Cassette (emulator) images
CCO	CoCo BASIC or Machine Code programs saved direct to the memory Card (MSAVE)
CCC	CoCo cartridge (ROM) Image
DGN	Dragon BASIC or Machine Code programs saved direct to the memory Card (MSAVE)
DSK	CoCo Floppy Disk Image.
ROM	Dragon cartridge (ROM) Image
VDK	Dragon Floppy Disk Image.

CAT can also be used with filenames and wild card characters to limit the display of the files.

CAT
MY.CAS
CUTHBERT.CAS
MYDISK.VDK
MYPROG.CAS
RAILRUN.ROM

To look for any .ROM files in the current directory the * wild card can be used:-

CAT "*.ROM"
RAILRUN.ROM

The ? (question mark) can be used to match any single character:-

CAT "?Y.CAS"
MY.CAS

Directories are distinguished from files by '<' and '>' around the name:

CAT
<.>
<..>
<GAMES>
<UTILITY>

Too many files listed to fit on the screen? Try the CATP command which will page the results.

The command CWD can be used to change (the working) directory on the memory card, when entering directory names the '<' and '>' around the name should be omitted:-

CWD "GAMES"

To move into the parent directory use .. as the directory name:-

CWD ".."

You can also move to through multiple directories at the same time:

CWD "GAMES/BLABY"

Finding your current folder is easy with the WD$ function:

PRINT WD$
0:/GAMES/BLABY

The "0:" shows that the current memory card is drive 0, as there is only one card slot the drive is always 0

To go back to the root directory, you can specify "/":-

CWD "/"

Long File Names

Whilst the Dragon (and CoCo) is limited to short filenames (8.3 as above) on the memory card modern computers support long filenames. Any long filenames on the memory card will be shortened to fit the short filename scheme.

When shortened on a modern computer, a '~' (tilde) is used in short version of the name for long filenames (to fit 8.3). However the Dragon character set does not have the "~" character, so instead the inverse "↑" is used. The inverse "↑"cannot be entered on the Dragon or CoCo keyboards, so "↑" (up arrow) is used in replace of the inverse "↑" when entering file and directory names, for example:

Starship Chameleon.CAS	Modern long filename
Starsh~1.CAS	Displayed as short filename on a modern computer
STARSH↑1.CAS	As displayed by the CAT command
STARSH↑1.CAS	As entered from the Dragon keyboard

How this is used on the Dragon:-

    CAT
    STARSH↑1.CAS
    MTAPERUN "STARSH↑1.CAS"

Due to the long filename support and the Dragons (now) limited keyboard, files on the memory card containing "^" (caret symbol) in them are *not* supported.

CHAPTER TWO

LOADING UP

This chapter covers how to load and use emulator cassette, cartridge and disk images on your *real* Dragon or CoCo.

Loading Cassette Images (CAS)

The quickest and easiest method to load cassette images is with the MTAPERUN command:-

MTAPERUN "CUTHBERT.CAS"

This command performs the following actions:
a)   turns on tape emulation
b)   mounts the specified file into the virtual cassette recorder as read only
c)   loads the first program from the emulated tape image (CUTHBERT.CAS in this example).
d)   automatically executes the program, BASIC or machine code.

When using cassette images the physical relay in the computer will still *click* for the cassette recorder motor as if was loading a physical cassette.

Many cassette programs do not expect the disk system to be present, to improve compatibility it's recommend that disk emulation is turned off (MDISK OFF) when not being used.

An alternative method is to use the cassette emulation mode, when enabled all BASIC cassette commands, are directed to the virtual cassette recorder and the mounted cassette image (CAS file).

MCAS

Enables cassette emulation mode, so all tape related commands go to the virtual recorder instead of the physical tape port.

MMOUNTR "BUZZARDB.CAS"

This mounts the tape image BUZZARDB.CAS in the virtual recorder in read only mode.

If you don't know what type of program the cassette image is use the CFTYPE function.
0 = BASIC, 1 = ASCII data/BASIC, 2 = Binary data or Machine Code, 255 = Invalid

```
PRINT CFTYPE
2
CLOADM
```

Load the program using the normal cassette commands.
Putting this all together we have:

```
MCAS
MMOUNTR "BUZZARDB.CAS"
PRINT CFTYPE
2
CLOADM
```

When loading, the computer will behave as if it's loading a real cassette, so the computer will click and F with the program name will appear in the top left hand corner of the screen. The difference is that the program will load in seconds instead of minutes.

If you need to rewind the virtual recorder back to the beginning of the beginning of the tape image use:

```
REWIND
```

When finished with the cassette image it can be unmounted:

```
MMOUNTC
```

Loading Cartridge Images (ROM/CCC)

The DMMC allows use of cartridge image (ROM or CCC) files and will load them into the same memory space as a normal cartridge, to become a virtual cartridge that will behave in the same manner as a real cartridge. The most common cartridges are auto starting and they are loaded using MCARTLOADA :

```
CAT
RAILRUN.ROM
MCARTLOADA "RAILRUN.ROM"
```

This will load the RAILRUN.ROM image into memory and auto start it.

To load a cartridge image into the virtual cartridge but not auto start it, use MCARTLOAD:

CAT "*.ROM"
MYCART.ROM
MCARTLOAD "RAILRUN.ROM"

Note:- Loading a cartridge image will overwrite the MMC memory, so all MMC functions and commands (inc snapshot) will no longer be available, until the next cold boot.

Pressing the Force Cold Reset Button whilst pressing the computers reset button will perform a cold reset when running ROM (cartridge) images.

Using Floppy Disk Images (DSK/VDK)

The DMMC will emulate DragonDOS (Dragon) or RSDOS (CoCo) systems with 4 virtual floppy disk drives, into which 4 disk images can be mounted. At boot time the DMMC determines which computer it's being used on (Dragon or CoCo) and auto enables the appropriate disk system. Enabling the DOS system (the equivalent of connecting the DOS cartridge with disk drives) will cause the computer to do a cold boot, so any programs etc. in memory will be lost. Enable DOS by typing the following,

> MDISK ON

The computer will now perform a cold boot showing the usual copyright and DMMC messages plus the DOS message, you now have 4 disk drives at your disposal that can be used in exactly the same way as a real DOS system but with the benefit of access to the memory card.

The DMMC command to mount (or insert) a disk image into a drive is:
MDISKI *driveno,"filename"*
Where *driveno* is the drive number (1-4 for Dragon / 0-3 for CoCo) and *filename* is the name of the disk image on the memory card. The DMMC uses the file extension on the disk image to determine the disk format, so DSK should be used on CoCo format disks and VDK should be used on Dragon format disks, although the DMMC allows the use of both on either computer.

Let's mount (insert) a disk image into drive 1:-

> CAT
> MYDISK.VDK
> MDISKI 1," MYDISK.VDK"

Now we have a disk in drive 1, it can accessed using the normal DOS commands, such as directory listing and loading a BASIC program:

> DIR 1
> MYPROG.BAS
> LOAD "MYPROG.BAS"

Having 4 disk drives available can make it difficult to keep track of what disk images are mounted, so use the following:-

> MDISKL

This will list the 4 disk drives and the names of the disk images in use in them, in our example it will show MYDISK.VDK in disk drive 1.

Finished with a disk and want to pop it out of the drive?
MDISKO *driveno*
where *driveno* is number of the drive you wish to unmount the image on.

MDISKO 1

Blank disk images can also be created on a Dragon like so:

MDISKC 1,"NEWDRAG.VDK"
DSKINIT 1,2,40

On a Coco:

MDISKC 1,"NEWCOCO.DSK"
DSKINI1

DOS emulation will remain permanently enabled until it is turned off, even if the computer is turned off, DOS emulation can be turned off with:-

MDISK OFF

Disabling the DOS system with MDISK OFF (the equivalent of removing the DOS cartridge with disk drives) will cause the computer to do a cold boot, so any programs etc. in memory will be lost.

The currently mounted virtual disks are saved to the binary SETTINGS file in the root directory of the memory card. This means that the disks will remain mounted even after computer has been turned off/on.

Booting FLEX

As the FLEX operating system needs 64K and uses the same memory space as the DMMC, it requires an additional driver. To allow booting of FLEX please download the latest FLEX unities.
http://www.archive.worldofdragon.org/index.php?title=DragonMMC_SDCard
Copy the download files DISKMMC.DGN and FLEXBOOT.DGN onto your SD/MMC card along with the flex disk image you want to boot e.g. FLEXBOOT.VDK.

Mount the FLEX disk you wish to boot, into the first drive and then run the included DISKMMC.DGN and FLEXBOOT.DGN utilities in order:

```
MDISKI 1,"FLEXBOOT.VDK"
MLOAD "DISKMMC.DGN"
MLOADA "FLEXBOOT.DGN"
```

Note: at present there are no native Flex utilities for manipulating disk images so any images you wish to use, must also be mounted in the other virtual drives before booting.

Don't forget "*FLEX IS BEST*".

CHAPTER THREE

SNAPPY DRAGON MMC

The DMMC snapshot button allows you to save and load 'snapshots'. The snapshot facility saves the current state of the computer and when loaded back in it will return the computer back to exact same point it was pressed. So if you've been adventuring in the forest for hours but don't want to lose all that effort, don't worry snapshot it.

The snapshot facility relies on the NMI interrupt and memory allocated to the cartridge slot, Because of this cartridge images (ROM/CCC) and certain software do not support the snapshot facility.

As default snapshots are saved to the root directory of the memory card, this can be changed using the CWDS command and displayed with the variable SD$.

> MKDIR "0:/DRAG-SSD"
> CWDS "0:/DRAG-SSD"
> PRINT SD$
> 0:/DRAG-SSD
> CWD SD$

This will create the snapshot directory, set the snapshot directory, display its current setting and change directory the snapshot directory.

Pausing, Saving and Loading

Let's try snapshot with the classic Hello World program

> 10 PRINT "HELLO WORLD"
> 20 GOTO 10
> RUN

This will enter a program and set it running forever but now press the Snapshot button on the DMMC. This will have the effect of pausing the normal execution of the computer and bring up the Snapshot menu at the bottom of the screen.

When the Snapshot is activated the computer will be put back into normal text mode. In some cases, the top part of the screen will look corrupted; this is normal and just residual data from the running program.

Press the [X] key to exit the menu, this will un-pause the running program and allow it to continue as if nothing happened.

Press the Snapshot button again to bring up the menu and press the [S] key to save the snapshot. Enter a filename of MYSNAP for the snapshot (up to 8 characters is allowed), there's no need to add the extension it's added automatically (SSD for Dragon snapshots and SSC for CoCo). Press the [X] key to exit the menu and the program will carry on running.

Press the [BREAK] key to stop the program and type NEW to clear the Hello World program from memory.

Let's restore the Hello World program back to its running state, this can be done through the BASIC command MLOADS or through the Snapshot menu. In BASIC enter the following:-

    CWD SD$
    MLOADS "MYSNAP.SSD"

The Hello World program is restored back in an instant to its original running state! Let's clear the program again, press the [BREAK] key to stop the program and type NEW.

Press the Snapshot button to bring up the menu and press the [L] key to load a snapshot. Enter the filename of MYSNAP for the previously saved snapshot, the snapshot will now load and again return the Hello World program to its running state.

You have now successfully paused, saved and loaded your first snapshot.

Booting and Resetting

The snapshot button also has the convenient feature of being able to perform a reset (warm reset) or the equivalent of a power cycle (cold reset).

Press the Snapshot button to bring up the menu and press the **[W]** key to perform a Warm reset, the equivalent of pressing the reset button.

Press the Snapshot button to bring up the menu and press the **[C]** key to perform a Cold reset, equivalent to when the power is turned on.

If an auto starting ROM image is loaded then the computers reset button and DMMC snapshot buttons will not allow you to exit to program, to reset the computer press the force cold reset button at the same time as the computers reset button.

CHAPTER FOUR

GOING NATIVE

In addition to supporting emulator files the DMMC has many BASIC commands and functions that allow you create your own CAS files and incorporate the DMMC into your own programs.

Making Your Own Cassette Image (CAS)

As well as being able to load CAS files generated from original cassettes, with the MCAS tape emulation mode the normal CSAVE and CSAVEM will allow you to save directly to a CAS file.

Let's try saving a BASIC program again using the classic Hello World program

```
10 PRINT "HELLO WORLD"
20 GOTO 10
```

Enable cassette emulation mode, to direct all tape related commands go to the virtual recorder:

```
MCAS
```

Creates and mount a CAS file to save our Hello World program in the current directory:

```
MMOUNTW "HELLOW.CAS"
```

Save the program using the standard CSAVE command:

```
CSAVE "HWORLD"
```

View the CAS file on the card:

```
CAT "HELLOW.CAS"
```

Before we load the program back in, we need to close the virtual tape, mount it for reading and clear the program from memory:

```
MMOUNTC
MMOUNTR "HELLOW.CAS"
NEW
```

Ready to load the program back in from the CAS:

```
CLOAD
```

When loading the computer will behave as if it's loading a real cassette, so the computer will *click* and "F HWORLD" will appear in the top left hand corner whilst loading.

Now we've finished loading we can take the take the virtual cassette out of the virtual tape recorder (MMOUNTC) and run our freshly loaded Hello World program:

```
MMOUNTC
RUN
```

You can now load and save your own BASIC programs, into a CAS file. As the MCAS and MMOUNT*x* commands allow the use of the standard BASIC tape commands further examples on how to use load and save to cassette can be found in the original BASIC manual.

What's In a Directory

The CAT and CATP commands allow you to see what files and directories are in the current directory but there are functions available to return strings with the directory listing.

FINDFIRST$ finds the first file/directory in the current folder

```
PRINT FINDFIRST$
<.>
```

 To get the next file in the directory use the FINDNEXT$ function

```
PRINT FINDNEXT$
<..>
```

Subsequent calls made to FINDNEXT$ returns the next file/directory in the current folder, when no more files are found FINDNEXT$ returns an empty string.
Pattern matching (wildcards) can also be used with FINDFIRST$ in the same manner as CAT

```
PRINT FINDFIRST$ "*.CAS"
MY.CAS
```
and
```
D$ = FINDFIRST$ "*.CAS"
PRINT D$
MY.CAS
```

Now we can put this together to write a program that displays all the CAS files in the current directory:

```
10 PRINT "CURR DIR: " WD$
20 PRINT FINDFIRST$ "*.CAS"
30 N$=FINDNEXT$
40 IF N$="" GOTO 70
50 PRINT N$
60 GOTO 30
70 PRINT "***DONE***"
```

Saving In DMMC Native Format

Let's try saving a BASIC program with the classic Hello World program

```
10 PRINT "HELLO WORLD"
20 GOTO 10
MSAVE "HWORLD.BAS"
```

This will create the file HWORLD.BAS on the memory card, containing the Hello World program.

If auto .BAK creation is enabled when using MSAVEM and HWORLD.BAS already exists it is renamed to HWORLD.BAK and if HWORLD.BAK already exists it is deleted. If auto .BAK creation is not enabled when using MSAVEM and the file HWORLD.BAS already exists the file will not be overwritten and the command will error. See MSETCFG command for details on how to set this.

When saving BASIC programs it is recommend that the file extensions of .BAS, .DGN or .CCO are used.

Now we can clear the program and reload it:

```
NEW
MLOAD "HWORLD.BAS"
LIST
10 PRINT "HELLO WORLD"
20 GOTO 10
```

CHAPTER FIVE

TROUBLESHOOTING

This chapter gives help on loading cassette images and the use of DOS. Also don't forget to keep up to date with new firmware etc. at the following address:

http://www.archive.worldofdragon.org/index.php?title=DragonMMC_SDCard

THE PROBLEM WITH CAS

Over the years CAS files have been produced with various tools in different states of development and this has resulted in CAS files of varying quality. As the DMMC uses the original cassette loading routines, any CAS files with faults won't load. When using emulators like XROAR many of these problems aren't apparent as it can fix some issues 'on the fly'.

Below are some guidelines on how get the CAS files loaded.

First Steps

It may seem obvious but the first thing to try is a different CAS file for the same title. Many titles have different CAS files available and they may have been created at different times with different software and different sources, so there's a good chance that another CAS file will work.

Typically cassette programs do not expect the disk system to be present, so turn disk emulation off with:

MDISK OFF

Along with the disk system not being present some software titles rely on the computer being in a fresh state, where the first command(s) entered after a cold boot are to load the software. If using AUTOEXEC to run a program at boot time, such as the 'Explorer' prevent it from running by pressing the space bar at boot time or to permanently disable it, with (show clock at boot time):

MSETCFG 16

Or with no clock display at boot time:

MSETCFG 0

Perform a cold boot (power off/on) for the change to take effect.

Try both methods of loading CAS files, (MCAS and MTAPERUN) straight after turning on with no autoexec, e.g.:

MCAS
MMOUNTR "BUZZARDB.CAS"
CLOADM

and

MTAPERUN "BUZZARDB.CAS"

New Life for CAS

The XROAR emulator has the ability to 'regenerate' CAS files, fixing a number of common issues along the way. For this you will need a copy of XROAR (www.6809.org.uk) installed with a BASIC ROM for your chosen computer.

Once you have setup XROAR with your chosen ROM, we'll create a new shortcut to set the options to re-write the loaded CAS image into a new file:
- Browse to the folder containing XROAR.
- Right click on the XROAR executable and select 'Create Shortcut'.
- Right click on the new shortcut and select rename.
- Rename the shortcut to 'xroar.exe write CAS'
- Right click on the shortcut and select Properties.
- In the Target: test box add the following: ' -tape-rewrite' to the end of the text. The end of the Target text will now look like:
    xroar.exe -tape-rewrite
- Click OK when done

Now the shortcut has been created we can now regenerate the CAS file:
- Double click on the shortcut 'xroar.exe write CAS' to open XROAR.
- Select the CAS file you wish to regenerate: File -> Cassette -> Input Tape..
- Enter the filename and select the location you wish to save the new CAS file to: File -> Cassette -> Output Tape..
- Use CLOAD or CLOADM to load your software.
- Once loaded quit XROAR.
- You should now have a new CAS file ready to load on your computer with the DMMC.

If you are comfortable using the Windows command line the regeneration of the CAS file can be done one command line:-

xroar -tape-rewrite -timeout-motoroff 1 -run *<input-file>* -tape-write *<output.cas>* -ui null -ao null

    e.g.

xroar -tape-rewrite -timeout-motoroff 1 -run cranky.cas -tape-write NEWcranky.cas -ui null -ao null

THE PROBLEM WITH DOS

DragonDOS has been known to be buggy for years, with Dragon User May 1985 listing over 15 known problems and bugs in DragonDOS 1.0. Whilst DMMC is built on version 1.3, bugs still do remain and are probably more likely to occur if using all the disk drives on the DMMC, given that most users in the day would have only had one or two disk drives.

For example, one bug found during DMMC testing is that when a file is copied (using COPY) from drive 4 to any other drive it will corrupt the destination drive. It is therefore recommended that you regularly backup important disk images to a modern computer in case of disaster when using DragonDOS or RSDSOS.

APPENDIX A

DRAGON MMC COMMANDS

CAT

CAT *"pattern"*

The CAT command lists the files (and folders) on memory card in the DMMC.

CAT	List all files in the current directory.
CAT ".."	List all files in the parent directory.
CAT "./GAMES"	List all files in the GAMES sub directory.
CAT "/"	List all files in the root directory.

The CAT command also accepts the (* and ?) wildcard characters.

*	Matches zero or more characters
?	Matches any single character

CAT "*.CAS"	List all the files in the current folder that end in ".CAS"
CAT "A?"	List 2 character files in the current folder that start with "A" and any second character.
CAT "DISK/*.VDK"	List all files in the subfolder "DISK" that end in".VDK"

Directories are distinguished from files by '<' and '>' around the name:

CAT
<.>
<..>
<GAMES>

If there are less than 256 files and folders returned, folders will appear before files and each will be alphabetically sorted. If more than 256 items are present no sorting will occur due to memory limitations.

See Also:-
CATP
Long File Names (Chapter 1)

CATP

CATP *"pattern"*

The CATP command lists the files (and folders) on the memory card in the DMMC.
If the number of files exceeds what can be displayed on screen the results will be paged.

See CAT command for examples.

CWD

CWD *"dirname"*

Changes the current directory on the memory card to the specified directory.

The command CWD can be used to change (the working) directory on the memory card, when entering directory names the '<' and '>' around the name should be omitted:-
    CWD "GAMES"

To move into the parent directory use .. (dot)(dot) as the directory name:-
    CWD ".."

You can also move to through multiple directories at the same time:-
    CWD "GAMES/BLABY"

To go back to the root directory, you can specify "/":-
    CWD "/"

The CWD command can also be used with strings:-
    A$="GAMES"
    CWD A$

CWDS

CWDS *"dirname"*

Sets the directory where snapshots are saved, by default the root folder on the card is used.

The directory specified must exist and be a full path. If the directory specified does not exist, it will default back to the root folder.

Dragon and CoCos have separate folders and must be set on each computer.

    CWDS "/DRAG-SSD"
    PRINT SD$
    0:/DRAG-SSD

See Also:-
    SD$

HELP
HELPX

The HELP command displays the version text of the DMMC and sub components.

HELP

The HELPX command displays same as HELP plus the compile dates for AVR firmware and AVR bootloader.

HELPX

See Also:-
VER (function)

MCARTLOAD
MCARTLOADA

MCARTLOAD *"filename"*
MCARTLOADA *"filename"*

Load the specified cartridge ROM/CCC image file into the MMC card ram in the $C000-$FFEF area (cartridge slot memory).

MCARTLOADA resets the machine after loading the cartridge image and connects the Q clock to the FIRQ line on the cartridge port. This allows the use of auto-starting cartridge images such as the majority of games.

Note Loading a ROM image will overwrite the MMC ROM, so MMC functions will no longer be available.

MCAS

The MCAS command enables virtual cassette recorder mode, this allows the use of emulator .CAS files as if it was a normal tape.

When enabled all BASIC cassette commands (CLOADM etc.) are directed to the virtual DMMC cassette drive instead of the physical tape drive of the computer.

MCAS
MMOUNTR "GAME.CAS"
CLOAD

Most commercially released software doesn't expect to run on computer with DOS. It's recommend that disk (DOS) emulation is turned off (see MDISK OFF) if not specifically needed when loading cassettes.

The MCAS command copies the upper 32K from the ROM into the 32K of RAM on the MMC board and patches it so that cassette based rom calls are directed to the MMC.

See Also:-
MMOUNTC
MMOUNTR
MMOUNTW
MDISK OFF

MDELETE

MDELETE *"filename"*

Deletes the specified file from the memory card, you will be prompted (Y/N) before deletion.

> MDELETE "MYPROG.DGN"
> ARE YOU SURE (Y/N)
> OK

MDISK ON

Turn on disk emulation for the current platform and do a cold reset.

See Also:-
> MDISKL
> MDISKO
> MDISK OFF
> MSETCFG

MDISK OFF

Turn off disk emulation for the current platform and do a cold reset.

See Also:-
     MDISKL
     MDISKO
     MDISK ON
     MSETCFG

MDISKC

MDISKC *driveno,"filename"*

Creates a blank disk image with the *filename* in drive *driveno* (1-4 for Dragon / 0-3 for CoCo), can be used for VDK and DSK images. Disk will require formatting before use (i.e. DSKINIT or DSKINI).

        MDISKC 1,"NEWDISK.VDK"
        DSKINIT 1,1,40
        CAT
        NEWDISK.VDK
        MDISKL
        1: 0:/VDK/NEWDISK.VDK
        2:
        3:
        4:
        OK

See Also:-
        MDISKI
        MDISKO

MDISKI

MDISKI *driveno,"filename"*

Mount disk image *filename* in drive *driveno* (1-4 for Dragon / 0-3 for CoCo).

> CAT
> MYDISK.VDK
> BACKUP.VDK
> MDISKI 1," MYDISK.VDK"
> MDISKI 3," BACKUP.VDK"

When using disk images the file extension for the images should be DSK for CoCo format and VDK for Dragon format. .

Notes on floppy disk Images:
DSK Floppy disk images (normally CoCo) will be treated as raw sector dumps so the virtual geometry will be guessed from the file size,
VDK Floppy disk images (normally Dragon) have a header that defines the geometry so it that used to determine the disk format.

See Also:-
> MDISKC
> MDISKL
> MDISKO
> MDISK OFF
> MDISK ON

MDISKL

List mounted disk images in the 4 virtual floppy disk drives.

```
CAT
MYDISK.VDK
BACKUP.VDK
MDISKI 1," MYDISK.VDK"
MDISKI 3," BACKUP.VDK"
MDISKL
1: 0:/VDK/MYDISK.VDK
2:
3: 0:/VDK/BACKUP.VDK
4:
OK
```

See Also:-
MDISKI
MDISKO

MDISKO

MDISKO *driveno*

Dismount disk image from *driveno* (1-4 for Dragon / 0-3 for CoCo).

Unmount the disk image in virtual drive 3:-
        MDISKO 3

See Also:-
        MDISKI
        MDISKL

MFILEC

MFILEC *"file1", "file2"*

Copies file1 to file2.

Create a copy of a file:-
        CAT
        MYPROG.BAS
        MFILEC "MYPROG.BAS","COPYPROG.BAS"
        MYPROG.BAS
        COPYPROG.BAS

MFILER

MFILER *"file1", "file2"*

Rename (or move) file1 to file2.

Rename a file:-
    CAT
    MYPROG.BAS
    MFILER "MYPROG.BAS","NEWPROG.BAS"
    CAT
    COPYPROG.BAS

Move a file to the parent folder:
    MFILER "MYPROG.BAS","../MYPROG.BAS"

MKDIR

MKDIR *"dirname"*

Creates the specified directory.

Create a directory in the current folder:-
    MKDIR "GAMES-A"

Absolute and relative paths can also be specified:-
    MKDIR "/GAMES-B"
or
    MKDIR "../GAMES-C"

MLOAD
MLOADA
MLOADC

MLOAD *"filename"*
MLOAD *"filename", location*
MLOADA*,"filename"*
MLOADC *"filename"*

Load the previously saved BASIC or machine code program from the memory card. MLOADA will attempt to automatically run a BASIC program once loaded.

MLOAD "DRAGPROG.BAS"

The *location* is the address to load a machine code program into memory.

MLOAD "MACH.BIN",1000

MLOADC operates as MLOADA but first clears any active program and returns the machine to its power on state.

See Also:-
MSAVE
MSAVEM

MLOADS

MLOADS *"filename"*

Loads a previously saved system snapshot:
MLOADS "GAMESNAP.SSD"

See Also:-
CWDS
SD$

MMIRROR
MMIRORA

MMIRROR *"filename"*
MMIRRORA *"filename"*

Mirrors tape data as it's loaded into the computer to a CAS format file.
MMIRROR "FILENAME.CAS"

After the MMIRROR command is entered, type CLOAD or CLOADM to load the program and let the tape load as normal.

Once the program has loaded correctly, press the snapshot button on the DMMC, this will finalise the mirror/copy process.

MMIRORA works like MMIRROR but just keeps reading the tape, using block in and echoing to the MMC, useful if you have a C90 full of stuff you want to archive all in one go.

MMOUNTC

Unmounts the currently open (.CAS) cassette image file, this will prevent any further reading or writing to the image file until it mounted again.

    MCAS
    MMOUNTW "MYPROG.CAS"
    CSAVE "MYPROG3"
    MMOUNTC

See Also:-
    MCAS
    MMOUNTR
    MMOUNTW
    REWIND

MMOUNTR

MMOUNTR *"filename"*

Mounts an existing cassette image file (.CAS) into the virtual DMMC tape recorder, in read only mode, preventing accidental overwriting.

    MCAS
    MMOUNTR "MYPROG.CAS"
    CLOADM

See Also:-
        MCAS
        MMOUNTC
        MMOUNTW
        REWIND

MMOUNTW

MMOUNTW *"filename"*

Creates and mounts a cassette image file (.CAS) into the virtual DMMC tape recorder in write only mode. This allows the saving of programs and data to a virtual cassette file.

```
MCAS
MMOUNTW "MYPROG.CAS"
CSAVE "APROG"
MMOUNTC
```

See Also:-
MCAS
MMOUNTC
MMOUNTR
REWIND

MSAVE

MSAVE *"filename"*

The MSAVE command may be used to save BASIC programs directly to the memory card.

MSAVE "MYPROG2.BAS"

Will create a file MYPROG2.BAS on the memory card, containing the BASIC program currently in memory.

If auto .BAK creation is enabled when using MSAVEM and MYCODE.BAS already exists it is renamed to MYCODE.BAK and if MYCODE.BAK already exists it is deleted.
If auto .BAK creation is not enabled when using MSAVEM and the file MYCODE.BAS already exists the file will not be overwritten and the command will error.

When saving BASIC programs it is recommend that the file extensions of .BAS, DGN or .CCO are used.

See Also:-
  MLOAD
  MSAVEM
  MSETCFG

MSAVEM

MSAVEM *"filename", start, end, entry*

The MSAVEM command may be used to transfer machine code programs (or other binary data) directly to the memory card.

Note the filename is used as specified, and no extension is added. Files are saved in the same format as files on a DragonDOS disk.

MSAVEM "MYCODE.BIN",2000,5000,4000

Will create a file MYCODE.BIN on the memory card containing the sequence of bytes from memory start-address 2000 to end-address 5000. When this file is loaded, the default EXEC address will be set to 4000.

If auto .BAK creation is enabled when using MSAVEM and MYCODE.BIN already exists it is renamed to MYCODE.BAK and if MYCODE.BAK already exists it is deleted.
If auto .BAK creation is not enabled when using MSAVEM and the file MYCODE.BIN already exists the file will not be overwritten and the command will error.

When saving machine code programs it is recommend that the file extensions of .BIN, .DGN or .CCO are used.

See Also:-
        MLOAD
        MSAVE
        MSETCFG

MSETCFG

MSETCFG *flags*
MSETCFG *flags , <platform>*

Set the configuration flags for the interface, the parameters are bitwise and of the following values.

A valid *platform* is:-

0	System
1	Dragon
2	CoCo

If no platform is specified then the current platform (Dragon or CoCo) is assumed.
Valid *flags* for the System (DMMC) platform:-

Hex	Dec	Purpose
&H80	128	Enable firmware updating from the memory card the next time it is rebooted. **
&H04	4	Enable auto .BAK creation when using MSAVE . When attempting to create a file with MSAVE"TEST.BAS", if TEST.BAS already exists it is renamed to TEST.BAK and if TEST.BAK already exists it is deleted.
&H02	2	Disable the display of files beginning with ' ._' (dot)(underscore) or ' _ '(underscore). This allows hiding of Apple Mac meta data files.
&H01	1	Enable debugging information to be sent to the AVR's TTL serial port. This is really only of use for firmware development, and *WILL* slow down the interface!

** The firmware update flag is reset after a successful firmware update.

Continued…

Valid *flags* for the Dragon and CoCo computers:-

Hex	Dec	Purpose
&H40	64	Enable execution of the "AUTOEXEC" file, this should be a BASIC program saved with MSAVE.
&H20	32	Enable DragonDOS (Dragon) or RSDOS (CoCo) emulation using disk images. Also set by MDISK ON / MDISK OFF
&H10	16	Enable the display of real time clock data and time at boot up.
&H08	8	Enable the display of the firmware compile date at start-up.

Set AUTOEXEC *off*, DOS *on* (32) and RTC display *off*
> MSETCFG 32

Set AUTOEXEC *on* (64), DOS *off* and RTC display *on* (16)
> MSETCFG 80

Or in Hex:
> MSETCFG &H50

Disable the display of files starting with ' ._ '
> MSETCFG 2,0

Enable the display of files starting with ' ._ '
> MSETCFG 0,0

See Also:-
> MGETCFG
> MSETCFGC
> MSETCFGS

MSETCFGC
MSETCFGS

MSETCFGC*flags*
MSETCFGC *flags , <platform>*
MSETCFGC will clear any bit that is set in the config byte, and leave alone any that is clear, i.e
MSETCFGC performs a bitwise OR operation.

MSETCFGS *flags*
MSETCFGS *flags , <platform>*
MSETCFGS will set any bit that is set in the config byte, and leave alone any that is clear, i.e
MSETCFGS performs a bitwise AND NOT operation.

Assume the system configuration byte is currently &H03 (0000 0011)

Command	Current Value	Value After
MSETCFGS &H80,0	&H03 (0000 0011)	&H83 (1000 0011)
MSETCFGC &H01,0	&H03 (0000 0011)	&H02 (0000 0010)

MSETDT

MSETDT *"YYYY-MM-DD hh:mm:ss"*

Set the real time clock, used for date stamping files.

Format **MUST** be:-
YYYY	Year 2000-2099
MM	Month 01-12
DD	Day of month 01-31
hh	Hour in 24 hour format 00-23
mm	Minutes 00-59
ss	Seconds 00-59

Setting the date and time:-
    MSETDT "2017-04-02 10:27:00"

Note: that you must use a leading zero for values less than 10!

See Also:-
    MGETDT$

MTAPERUN

MTAPERUN *"filename"*

Turns on tape emulation, mounts the specified file as read-only and loads the first file from the emulated tape. The first program on the image is automatically executed.
Equivalent to MCAS : MRMOUNT "filename" : CLOAD : RUN for basic programs and MCAS : MRMOUNT "filename" : CLOADM for machine code.

NOTE: MTAPERUN does the equivalent of CLEAR 200,*ramtop* before executing and loading the cassette file (where *ramtop* is the amount of physical RAM e.g. 16K, 32K).
If you do not want this to happen, use MCAS, MMOUNTR and CLOAD(M) instead.

RETOKC

BASIC commands on the Dragon and CoCo are stored in the memory using 'tokens' but the Dragon and Coco use different tokens for the same commands. This means that when a BASIC program is loaded form another platform the resulting program becomes 'garbled'.

This command, re-tokenizes the BASIC program in memory from Dragon to CoCo. If no equivalent command is available the command is converted into a REM statement.

    MLOAD "MYPROG.DGN"
    RETOKC
    MSAVE "MYPROG.CCO"

See Also:-
    RETOKD

RETOKD

Re-tokenize the BASIC program in memory from CoCo to Dragon. If no equivalent command is available the command is converted into a REM statement.

    MLOAD "MYPROG.CCO"
    RETOKD
    MSAVE "MYPROG.DGN"

See Also:-
    RETOKC

REWIND

Rewinds the cassette image file (.CAS) loaded into the virtual DMMC tape recorder.

MCAS
MMOUNTR "MYPROG2.CAS"
CLOADM
?FM ERROR
OK
REWIND
CLOAD
OK

See Also:-
MCAS
MMOUNTC
MMOUNTR
MMOUNTW

APPENDIX B

DRAGON MMC FUNCTIONS

Note: Functions ending in $ return a string, all others return an integer.

CFTYPE

Get the file type of the next file on an emulated cassette.

     0 = Tokenised BASIC,
     1 = ASCII BASIC or ASCII data
     2 = Machine code or Binary data
     255 = Invalid Header

Note: If an invalid CAS or non-cassette image file is mounted, any value 0-255 could be returned.

Use CFTYPE to identify the type of program (BASIC or Machine Code) before loading:-

     MCAS
     MMOUNTR "MYPROG.CAS"
     PRINT CFTYPE
     2
     CLOADM

CFTYPEE

Get the EXEC address of the next file on an emulated cassette.

     MMOUNTR "NERBLE.CAS"
     ?CFTYPEE
     10518

CFTYPEL

Get the load address of the next file on an emulated cassette.

MMOUNTR "NERBLE.CAS"
?CFTYPEL
109

FINDFIRST$

FINDFIRST$ "*pattern*"

Finds the first file/folder based on the pattern, subsequent calls should be made to FINDNEXT$
Returns an empty string when no files are found.

PRINT FINDFIRST$ "*.ROM"

See Also:-
FINDNEXT$
CAT command

FINDNEXT$

FINDNEXT$

Finds the next file in the sequence started by FINDFIRST$

```
PRINT FINDFIRST$ "*.ROM"
N$=FINDNEXT$
PRINT N$
```

Returns an empty string when no more files are found.

If there are less than 256 files and folders present, folders will appear before files and each will be alphabetically sorted. Due to memory limitations if more than 256 items are present no sorting will occur.

MGETCFG

MGETCFG
MGETCFG*(<platform>)*

Get the configuration flags, see MSETCFG for details. If no platform is supplied then the current platform is assumed (Dragon or CoCo).

A valid *platform* is:-

    0   System
    1   Dragon
    2   CoCo
    255   System enquiry

Get the configuration for the current computer (Dragon or CoCo):

    ?MGETCFG
    80

Get the configuration for the CoCo configuration:

    ?MGETCFG(2)
    16

To confirm which computer the current platform is, MGETCFG(255) can be used, returns 1 for Dragon and 2 for CoCo:

    ?MGETCFG(255)
    1

See Also:-

    MSETCFG command

MGETDT$

Returns the current real time clock time in the format as "YYYY-MM-DD hh:mm:ss".

The RTC can be displayed from BASIC:-

```
PRINT MGETDT$
2017-04-01 11:26:57
```

The MGETDT$ function can also be used in a program:-

```
10 DT$= MGETDT$
20 PRINT "CURRENT DATE AND TIME IS:"
30 PRINT DT$
```
Or
```
10 PRINT "CURRENT DATE AND TIME IS:"
20 PRINT MGETDT$
```

See Also:-
    MSETDT

MGETINAME$

MGETINAME$ DriveID

Returns the filename of the mounted disk image on the specified *driveid* (1-4 for Dragon / 0-3 for CoCo), returns an empty string if no disk is mounted on the specified drive.

> PRINT MGETINAME$ 1
> 0:/VDK/MYDISK.VDK

VER

Return the DMMC firmware version

> PRINT VER
> NUMBER TBC

NOT YET IMPLEMETED

SD$

Returns the directory name where snapshot files are saved. The default value is the root folder of the memory card.

    PRINT SD$
    0:/DRAG-SSD

See Also:-
    CWDS

WD$

Returns the current directory name.

    PRINT WD$
    0:/AGAMES
    OK

See Also:-
    CWD

APPENDIX C

DRAGON MMC MISCELLANEOUS
Everything else you need to know

AUTOEXEC

This useful facility allows the automatic execution (running) of a BASIC program when the computer does a cold boot. This allows you to automatically start your favourite game or menu program etc.

The AUTOEXEC file should be a BASIC program in the root directory of the memory card saved with MSAVE . To run the AUTOEXEC program at boot time ensure the option is enabled with MSETCFG

Set AUTOEXEC *on* (64), DOS *off* and RTC display *on* (16)

    MSETCFG 64 + 16
Or:
    MSETCFG 80
Or in Hex:
    MSETCFG &H50

There are separate AUTOEXEC files for Dragon and CoCo, this allows the same card to be used in both computers. Dragon has an AUTOEXEC file extension of .DGN and CoCo .CCO.

This example will create a simple welcome message every time the computer performs a cold boot, just turn the computer off and on again to see your new welcome message.

    10 PRINT "WELCOME TO THE DMMC"
    20 NEW
    CWD "/"
    MSAVE "AUTOEXEC.DGN"

To bypass the running of the AUTOEXEC, press the space bar during the cold boot.

SETTINGS

The DMMC saves the currently mounted virtual disks and location of the snapshot folder to the binary SETTINGS file into the root directory of the memory card. This allows disk images to remain mounted even after computer has been turned off.

There are separate SETTINGS files for Dragon and CoCo, this allows the same card to be used in both computers. Dragon has a SETTINGS file extension of .DGN and CoCo .CCO.

This file user editable and must be set with commands.

See Also:-
    CWDS
    MDISKI

Tape to CAS

The DMMC allows real cassette tape programs to be converted into a CAS file in a simple one step process. The MMIRROR command mirrors/copies tape data into a CAS format file as it's loaded into the computer.

To create your own CAS file connect the tape deck and cassette lead as normal, then enter the MMIRROR command with the filename you wish to call the cassette image on the memory card:

MMIRROR "MYTAPE.CAS"

At the OK prompt type CLOAD or CLOADM to load the program and let the computer complete loading as normal. As the tape is being converted the red LED on the DMMC will blink occasionally.

Once the program has loaded correctly, press the snapshot button on the DMMC, this will finalise the mirror/copy process. You will need to press [X] key to exit the snapshot menu. The newly generated CAS file can now be loaded with the DMMC using the normal tape image commands or used in emulators.

INDEX

THIS PAGE HAS INTENTIONALLY BEEN LEFT BLANK