The Fine Manual

Jump to bottom

charlierobson edited this page on 4 Jun 2020 · 10 revisions

einSDein

SD card interface for Tatung Einstein TC01 computers.

Installation

The EPROM.

Remove the lid of the computer. Extract the two screws at the rear of the machine. Lift the lid from the back, hinging along the front edge, to expose the innards.

Locate the spare EPROM socket to the right of the MOS (EP)ROM, which is fairly central on the main PCB. Insert the supplied EPROM. It may be obscured by disk drive cable, especially if two drives are installed.

Note: While you're there please take a note of the version number - if any - written on the MOS ROM. The OS patches I developed for the utility ROM are compatible with all the MOS versions that I've encountered so far, but there may be other versions that I'm not aware of. If this is the case then I can either supply an EPROM with the latest MOS or (more likely) update the utility code. Currently 1.20 and 1.21 are supported.

Close and secure the lid.

Power up the machine and check that it shows the usual blue screen and DOS messages. There should be nothing different at this time.

The interface.

Connect the interface to the 'Tatung Pipe' extension using the supplied cable. Take care to match the bumps on the plug with the notches on the socket.

Power up the machine again and check that everything looks normal. The blue and white LEDs should light momentarily if a card is inserted. Booting without a card will result in the interface being disabled until a card is inserted and the system restarted.

Usage

SD card.

einSDein requires a FAT formatted SD card. Files should be named according to the older style DOS 8.3 scheme. This is to ensure compatibility with XTALDOS. Long file names will show up in file listings but may not be accessible.

Virtual drives.

Two additional virtual disk drives are made available, 2: and 3:. These drives are usable from any DOS program and behave just as a standard 3″ 40 track SS floppy drive would. Disk images using the de-facto standard 'Extended DSK' format may be mounted and treated as a normal disk. Image files should be 211K, indicating they are of the correct format. They may be placed in the root directory of the card or indeed any other sub-directory of your choice.

Required files.

All required files can be downloaded ready to copy onto an SD card. Go to the following page:

https://github.com/charlierobson/einsdein-vitamins

Click the green 'Clone or download' button to the right of the page and select the 'Download zip' option. Copy the content of the folder 'card-image' to your card.

At reset the interface will check for the file named config.ini in the file system root. This contains configuration information such as the names of the disk images which will be automatically mounted, whether any pre-boot code is executed, where that code is located and the name of a file containing the DOS code which will be booted at reset or ctrl-break.

Standard DOS images hard-wire the boot drive to be drive 0:, and to get around this a modified DOS image is supplied which allows booting from SD card which is not only significantly faster but also obviates the need for a functioning floppy drive or bootable diskette.

Note: at present things go horribly wrong (though nothing fatal) if the DOS file is not present, or badly formatted/corrupt. I'm working on this :D See appendix 0 for further details.

NOTE: *The files here are for later model einSDeins, there are some earlier models that have slightly different firmware and require an alternative DOS. Always keep a backup of working system files before checking out new versions. The system is robust so nothing will break permanently but the ability to revert to a working image is desirable should a software mis-match occur.*

Tricks

Holding the ‘Graph’ key whilst booting will temporarily disable the interface. This is useful if, for example, you wish to boot a floppy.

XTALDOS 2.50

The last version of XTALDOS was 2.05. The source code for this was made available and a number of changes have been made to transparently integrate the file storage of the SD card into normal usage. This is known as XTALDOS250.

If an unknown command is typed at the command prompt then XTALDOS250 (XDOS) will check to see if a command with a matching name exists on the currently selected floppy. If nothing is found then the SD card will be checked and if still no command exists in the current working folder then a final check will be made in a folder called COM which should live in the file system root directory.

Additional command programs are supplied which allow file and folder manipulation on the SD card, as well as disk image creation and the movement of files between DOS volumes and the SD card. The Z80 source for the commands is available.

COMMAND REFERENCE

LS / CAT

LS ([dir]) CAT ([dir])

List the files in the current working folder or in the optionally specified subfolder [dir] of the SD card’s file system. The listing closely follows the format of the in-built DIR command. A star next to a file name indicates a read-only file. There is presently no way to change this flag from the Einstein, it must be done on a desktop computer.

It is easy to get confused with the built-in DIR command, which shows the content of a (virtual) floppy disk. Try not to get confused.

MNT

MNT [n] [([dir]/)file]

Mount a DSK formatted image [file] as drive [n]. If the mount succeeds then config.ini is updated with the new image filename so that it will be automatically mounted at next boot.

Example: To mount BBC BASIC in drive 3 from the disk image BBCBASIC.DSK, which is in the DSK folder, use the following command:

```
MNT 3 DSK/BBCBASIC.DSK
```

CD

CD [dir]

Changes the current working directory of the SD card. einSDein uses the forward slash as a subfolder separator. Commonly understood relative path specifications are available such as:

CD ..
CD ../xx
CD xx/yy
CD /aa

MD

MD [dir]

Creates a sub-folder [dir] in the current working folder of the SD card.

DEL

DEL [file]

Deletes a file from the SD card’s file system.

CP

CP [source] [dest]

Duplicates a file on the SD card.

RN

RN [old] [new]

Renames a file on the SD card.

GET

GET [file]

Copies a file from the SD card to the currently selected floppy drive.

Example: To copy the file 'DBG.COM' from the SD card to drive 0 use the following command:

```
0:
GET DBG.COM
```

PUT

PUT [file]

Copies a file from the currently selected floppy drive to the SD card.

RIP

RIP [file]

Makes a sector copy of the disk in drive 0:.

Note: This command is in early stage development and requires a ‘donor’ image to be present on the SD card in the working folder, currently BLANK.DSK. It is not changed during the operation, it just needs to be there to act as a template for the sector map metadata.

Information for programmers

How einSDein works

The einSDein interface decodes hardware IO ports $70 through $77. The ports and their symbolic names are as follows:

IOP_WRITECMD $70

IOP_WRITEDAT $71

OUTputting to these write-only registers will inform the interface that a command or data byte has been received. It will then either store the data byte (WRITEDAT) or proceed to execute a command according to the supplied command ID (WRITECMD). See appendix 1 for a list of command IDs. Commands may execute for an arbitrary amount of time, and a bit in the status register is set until the command has completed. If a command requires data in order to complete then the data must have already been written to the interface. The interface’s internal buffer pointer must be reset before writing a data stream. The internal buffer can hold a maximum of 512 bytes.

IOP_READ $72

Reading from this read-only register will return a command result or data byte.

IOP_STATUS $74

Various status bits may be read from this read-only register. If the interface has latched a byte ready for the Einstein to read then bit 1 will be set. This bit will be automatically cleared once the byte has been read by INputting from port $72. It is generally unnecessary to use this bit. If a command is currently executing then bit 2 will be set. It will automatically clear once command processing has completed, and the next read from port $72 will be the command’s return code.

IOP_DETECT $76

This special-purpose read-only register returns 42 decimal if the interface is present, otherwise you will receive $FF.

MCALs

Additional MCALs have been implemented which allow the adventurous user to make use of the facilities offered by einSDein. In all cases the A register holds the return code upon completion of the MCAL. As a special bonus the Z flag will be set if no error occurred. Unless otherwise specified no other registers contain useful return information.

SD_COMMAND $E0

Sends a command to the einSDein. The call will not return until the command has finished executing.

In: A = command ID

SD_SEND_DATA $E1

Sends a block of data up to 512 bytes in size to the einSDein. Buffer pointer is not reset so this command may be used to incrementally add data to the einSDein’s buffer.

In: HL = data source address, BC = data length

SD_RECV_DATA $E2

Receives a block of data from the einSDein. Buffer pointer is not reset so this command may be used to incrementally grab data from the einSDein’s buffer.

In: HL = data store address, BC = data length

SD_SEND_DATA_EXEC_CMD $E3

Resets the einSDein’s buffer pointer, sends a block of data up to 512 bytes in size, then executes the supplied command.

In: HL = data source address, BC = data length, A = command ID.

SD_EXEC_CMD_RECV_DATA $E4

Resets the einSDein’s buffer pointer, executes the supplied command then receives a block of data up to 512 bytes in size.

In: HL = data destination address, BC = data length, A = command ID.

Out: A = return code. Z flag will be set if no error occurred.

SD_LOAD_FILE $E8

Loads a file from the SD card into memory.

In: HL = address of file path, DE = destination memory address.

Out: HL = length of file.

SD_SAVE_FILE $E9

Saves a block of memory to a file on the SD card.

In: HL = address of file path, DE = data address, BC = data length.

The best way to see how all this fits together is to look at some of the source files for the transient commands.

Appendices

CONFIG.INI

This file controls how einSDein boots and what disk images are currently loaded. It should contain something like the following:

```
DEFAULT2=DSK/DEFAULT2.DSK
DEFAULT3=DSK/DEFAULT3.DSK
BOOTIMG=BOOT.BIN
DOSIMG=DOS250.BIN
FLAGS=0x80
```

Each line contains exactly one key=value pair. All key names are case insensitive. Every pair must be present. There should be no spaces between key and value.

DEFAULT2 and DEFAULT3 specify which disk images are mounted in drives 2 and 3 respectively. These will be updated by using the **MOUNT** command.

BOOTIMG specifies a filename containing code assembled to run at $8000. This is loaded at system reset and is executed before DOS is loaded. It is used to load the modified DOS and return to the system, skipping the MOS's own boot routine.

DOSIMG is the name of the file containing a DOS image, essentially the first two tracks of a standard XTALDOS floppy. It has to be modified otherwise drive 0 will be selected at boot.

FLAGS only has 2 active bits at the moment and only one of these is of use to end users:

BIT .. FUNCTION
7 controls whether einSDein boots automatically
6 enables serial debugging [likely to change]

Troubleshooting

- einSDein is connected but but machine just shows normal boot screen.
- Screen turns red and hangs.
- einSDein kind-of works but fails with errors after a while.

The main cause of malfunction is an incompatible SD card - try a different one if possible. Make sure that the required system files are present in the root folder of the SD card.

The next is that the interface is not recognised or isn't handling commands correctly. In this case you should try cleaning the pipe connector. Use alcohol & a cotton bud, or insert remove and re-insert the connector a few times. When doing this please be careful, connectors are always a weak link in any electro-mechanical system.

Pages 3

Find a page...

- Home
- DSKTOOL
- The Fine Manual
  - einSDein
    - Installation
      - The EPROM.
      - The interface.
    - Usage
      - SD card.
      - Virtual drives.
      - Required files.
      - Tricks
      - XTALDOS 2.50
    - COMMAND REFERENCE
    - Information for programmers
      - How einSDein works
        - IOP_WRITECMD $70
        - IOP_WRITEDAT $71
        - IOP_READ $72
        - IOP_STATUS $74

    - IOP_DETECT $76
  - MCALs
    - SD_COMMAND $E0
    - SD_SEND_DATA $E1
    - SD_RECV_DATA $E2
    - SD_SEND_DATA_EXEC_CMD $E3
    - SD_EXEC_CMD_RECV_DATA $E4
    - SD_LOAD_FILE $E8
    - SD_SAVE_FILE $E9
- Appendices
  - CONFIG.INI
  - Troubleshooting

Clone this wiki locally

```
https://github.com/charlierobson/einsdein-vitamins.wiki.git
```