SDCard HxC Floppy Emulator Direct Access mode

(Note : All informations in this file are subject to change)

The Direct Access mode can be used by host computer to access directly to the SDCard sectors. All sectors of the SDCard can be read/write by the host computer. By this way up to 32GB of data can be accessed through the floppy disk interface. This feature allows the development of file images selector software or hard disk driver.

To activate this mode, the host computer must move the head to the track 255. Once done, the emulator leave the floppy disk image mode and provide this track layout at the side 0 :

Sector ID=0 Status / Control	Sector ID=1 SDCARD DATA (LBA)	Sector ID=2 SDCARD DATA (LBA+1)	…	Sector ID=7 SDCARD DATA (LBA+6)	Sector ID=8 SDCARD DATA (LBA+7)

Index signal

This track is in MFM DD (250Kbits/s) format.

Status/Control sector :

The status/control sector is the interface to change the LBA address, and get the status of the interface. This sector is updated at each revolutions.

Data sectors :

These sectors are mapped over some SDCard Sector. The SDCard base sector are selected with the LBA parameter.

Some examples :

To READ sector 0x00100004 of the SDCard :

- ➔ Change the LBA to 0x00100000 (see Status/Control sector Write & commands chapter)
- ➔ Read The sector ID 5 (LBA+4)

To Write sector 0x00100002 of the SDCard :

- ➔ Change the LBA to 0x00100000 (see Status/Control sector Write & commands chapter)
- ➔ Write The sector ID 3 (LBA+2)

Status/Control sector Read:

```
typedef struct direct_access_status_sector_
{
        char DAHEADERSIGNATURE[8]; // contain -> HxCFEDA\0
        char FIRMWAREVERSION[12];   // firmware version (null terminated)
        unsigned long lba_base;      // Actual SDCard LBA base address
        unsigned char cmd_cnt;       // Command counter – incremented at each command passed
        unsigned char read_cnt;      // Read counter – incremented at each revolution.
        unsigned char write_cnt;     // Write counter – incremented at each sector write.
        unsigned char last_cmd_status; // Status of the last command passed : 0=no error.
        unsigned char write_locked;  // Write to the sdcard locked.
        unsigned char keys_status;   // Push buttons states.
        unsigned char sd_status;     // SDCard status ->0x00 init ok ! / 0xFF non present/error
        unsigned char SD_WP;         // SDCard write protect state
        unsigned char SD_CD;         // SDCard card detect state
}direct_access_status_sector;
```

The remaining bytes of the sector are unused/set to 0x00.

Status/Control sector Write :

```
typedef struct direct_access_cmd_sector_
{
        char DAHEADERSIGNATURE[8]; // Must be set to “HxCFEDA\0”
        unsigned char cmd_code;       // Command code
        unsigned char parameter_0;    // Parameter 0
        unsigned char parameter_1;    // Parameter 1
        unsigned char parameter_2;    // Parameter 2
        unsigned char parameter_3;     // Parameter 3
        unsigned char parameter_4;     // Parameter 4
        unsigned char parameter_5;     // Parameter 5
        unsigned char parameter_6;     // Parameter 6
        unsigned char parameter_7;     // Parameter 7
}direct_access_cmd_sector;
```

The remaining bytes of the sector must be set to 0x00.

Command set:

CMD_NOP : 0x00

This command does nothing.

Note : Once a Sdcard is removed from the slot, “sd_status” is changed to 0xFF.
To detect the reinsertion of a SDCard the CMD_NOP should be passed until the sd_status is changed to 0x00.

CMD_SET_LBA : 0x01

This command change LBA base address and the number of sector to read/write:

parameter_0 = LBA[7..0]
parameter_1 = LBA[15..8]
parameter_2 = LBA[23..16]
parameter_3 = LBA[31..24]
parameter_4 need to be set to 0xA5 or 0x5A to be able to write on SDCard sector. The value 0x5A disable the SDCard read. This is useful to make write burst into the SDCard.
parameter_5 Number of sector to read or write. If set to 0 the default value 8 is used.

Data sectors follow the Status/Control sector. The number of data sector is set into the parameter_5 register. Each data sector contain an SDCard LBA sector: LBA, LBA+1, LBA+2,…LBA+(parameter_5-1).

CMD_SET_TRACK_POS : 0x02

This command change the track position.

parameter_0 = Disk drive A track position.
parameter_1 = Disk drive B track position.

If all disk drive leave the track 255 position, the emulator come back into the normal mode/floppy disk image mode.