Last update: april 10th 2006

Level 2 Subprograms

These page contains a description of the various level 2 subprograms as supported by different mass-storage devicecards like floppy disk controllers, harddisk controllers, SCSI controllers and IDE controllers. Level 2 subprograms can be accessed from an assembler program only.

back to documentation index

Subprogram >x0 - Sector read write
Subprogram >11 - Format floppy disk
Subprogram >21 - Format SCSI
Subprogram >x2 - Modify file protection
Subprogram >x3 - File rename
Subprogram >x4 - Access direct input file
Subprogram >x5 - Access direct output file
Subprogram >x7 - Set current path name
Subprogram >x8 - Create subdirectory
Subprogram >x9 - Delete subdirectory
Subprogram >xA - Rename subdirectory
Subprogram >2C - SCSI drive identification
Subprogram >8C - IDE drive identification

Filenames and directory names are always 10 characters long, left justified and padded with blanks on te right. Error codes are returned at >8350 as described at the flag/status byte of the PAB.

The basenumber for the level 2 subprograms for the following devices are:

Floppy disk controllers:	DSK	>10
Myarc harddisk controller:	WDS	>20
Scuzzy controller	SCS	>20
IDE controller:	IDE	>80
Ti to PC serial connection:	HDX	>90

All devices return an error code at address >8350 in the three most significant bits, except SCSI which erroneously returns the error code in the three least siginifant bits:

Error Code	Meaning
>00 (0)	No error
>20 (1)	Device is write protected
>40 (2)	Bad open attribute
>60 (3)	Illegal operation
>80 (4)	Out of buffer space
>A0 (5)	Read past end of file
>C0 (6)	Device error
>E0 (7)	File error

If not described, the entries for the subprograms have the following meaning:

Returned sector number - The sector number actually written or read.
    (DSK and HDX only)

Unit# - Indicates the drive number for wich the operation has to be performed.

The drive numbers starts with 1 and the maximum number of drives depends on the devicecard used.

If the device used is a WDS, SCS or IDE and the most significant bit of unit# is set, the buffer address or filenames or directory names are assumed to be in CPU-ram instead of VDP-ram.

ReadWrite - Indicated the direction of data flow, 0=Write sector, <>0=Read sector

Protect - Protection code: 0 = Unprotect, 1 = Protect

Code - Function to perform:
0 = Read/Write file info
n = Read/Write file sectors of 256 bytes starting at the given sector in the additional file information. After the read/write has been completed this entry contains the actual number of sectors read or written. If all sectors have been read/written this entry will contain >00.

Info - Additional file information adres (+ >8300)

Buffer start address - Points to the start of the buffer for data transfer.
If the device is HDX, the buffer allways starts with a 10 bytes filename for the data to read from or write to followed by the 256 sector data.

Address of filename - Address of a 10 byte filename (see also note at unit#).

Address of pathname - Points to the length of a path name folowed immediatelly by the path name. The path name starts with a device name (DSKx. WDSx etc). The path name must be that of a directory and it must and with a period, and it must be less than or equal to 39 bytes in length (see also note at unit#). All the other subprograms will work on the current directory.

Adress of dir.name - Adres of a 10 byte directory name (see also note at unit#).

Sector number (LSB) - Is the least significant word of the sector number to be written or read. Sectors are addressed as logical sectors starting with zero.

Sector number (MSB) - Is the most significant word of the sector number to be written or read. This is only used for WDS, SCS and IDE devices, but it is allowed to make this entry >0000 for all other devices to form a 32-bit sectornumber.

Additional file information

This information is returned after calling subprogram >X4 (access direct input file) or written when calling subprogram >X5 (access direct output file)

```
        +-------------------------+
>83xx   | Buffer start address    |
        +-------------------------+
>83xx+2 | Number of first sector  |
        +------------+------------+
>83xx+4 | File flags | Recs/Sec   |
        +------------+------------+
>83xx+6 | Eof offset |Recordlength|
        +------------+------------+
>83xx+8 | Number of level 3 recs  |
        +------------+------------+
>83xx+A | MSB 1st sec| MSB l3 rec |
        +------------+------------+
>83xx+C | Extended record length  |
        +-------------------------+
```

Buffer start address - Pointer to te start of a buffer for data transfer. The buffer must be able to store at least n*256 bytes in wich n is the acces code.

Number of first sector - Indicates te sector number at wich read or write should begin. If the acces code=0 (parameter passing), the total numbers of sectors allocated for the file wil be returned or set.

File flags - The flags are at follows with bit 0 being the least significant bit (see also file descriptor record FDR):

```
        Bit:
        0  Program/datafile indicator
            0- Data file
            1- Program file
        1  Binary/Ascii data
            0- Ascii data (Display file)
            1- Binary data (Internal or program file)
        2  Reserved
        3  Protect flag
            0- Not protected
            1- Protected
        4  Reserved
        5  Reserved
        6  Reserved
        7  Fixed/Variable flag
            0- Fixed length records
            1- Variable length records
```

Recs/Sec - Contains the number of Records that wil fit in a sector of of 256 bytes.

Eof offset - The offset in the last sector used of a program or variable data file.

Record length - The logical record length of a data file.

Number of level 3 recs - In case of a fixed length record file, it contains the highest record actually written to. In case of a variable length record file, it contains the highest sector actually written to. The bytes in this field are in reverse order.

(!! Not sure if this is true !!)

MSB 1st sec - If file is bigger than 65535 sectors

MSB l3 rec - If file has more than 65535 level 3 records

Extended record length - Included for completeness and compatability with the Geneve9640. If a record length is bigger than 255 bytes, the logical record length will return as 0 and the extended record length will be used to pass the actual record length.

Subprogram >x0 - Sector read write

(DSK, WDS, SCS, IDE, HDX)

```
        +------------+------------+
>834A   | Returned sector number  |
        +------------+------------+
>834C   | Unit#      | ReadWrite  |
        +------------+------------+
>834E   | Buffer start address    |
        +-------------------------+
>8350   | Sector number (LSB)     |
        +-------------------------+
>8352   | Sector number (MSB)     |
        +-------------------------+
```

Subprogram >11 - Floppy disk formatting

(DSK)

```
        +------------+------------+
>834A   | Returned sectors/disk   |
        +------------+------------+
>834C   | Unit#      | Tracks     |
        +------------+------------+
>834E   | (not used)              |
        +-------------------------+
>8350   | Density    | Sides      |
        +-------------------------+
```

Returned sectors/disk - Number of sectors just formatted by the format routine.

Tracks - Indicates the number of tracks to be formatted. This entry must be either 40 or 80.

Density - The following values can be used:
```
    0,1 : Single density FM, 125Kbits/sec
    2   : Double density MFM, 250Kbits/sec
    3   : Quadruple density MFM. 500Kbit/sec (HFDC only)
```

Also for the HFDC:
The left most 6 bits contain (right justified) interleave information used by the format routine. The interleave can be any number from 1 to the number of sectors per track minus 1. If this field is left blank, the default value of 4 for single density and 5 for double density will be used.

Sides - Indicates the number of sides the floppy is to be formatted:
```
    0,1 : Single sided
    2   : Double sided
```

Subprogram >21 - Disk formatting

(SCS)

```
        +------------+------------+
>834A   | Status     | Result     |
        +------------+------------+
>834C   | Unit#      | Sectorsize |
        +------------+------------+
>834E   | Password                |
        +------------+------------+
>8350   | Error code |
        +------------+
```

Status - SCSI status byte (0=Ok)

Result - SCSI result (0=Ok)

Unit# - Indicates the SCSI-id (=drive number-1) for wich the operation has to be performed.

Sectorsize - Required sectorsize for the device:

```
    >01:    256 bytes
    >02:    512 bytes
    >03:    1024 bytes
    >04:    2048 bytes
    >05:    4096 bytes
    >06:    8192 bytes
    >07:    16384 bytes
```

Password - Address of the 6 byte string "FORMAT" in CPU ram.

Subprogram >x2 - Modify file protection

(DSK, WDS, SCS, IDE, HDX)

```
        +------------+------------+
>834C   | Unit#      | Protect    |
        +------------+------------+
>834E   | Address of filename     |
        +-------------------------+
```

Subprogram >x3 - File rename routine

(DSK, WDS, SCS, IDE, HDX)

```
        +------------+
>834C   | Unit#      |
        +------------+------------+
>834E   | Address of new filename |
        +-------------------------+
>8350   | Address of old filename |
        +-------------------------+
```

Subprogram >x4 - Acces direct input file

(DSK, WDS, SCS, IDE, HDX)

```
        +------------+------------+
>834C   | Unit#      | Code       |
        +------------+------------+
>834E   | Address of filename     |
        +------------+------------+
>8350   | Info       |
        +------------+
```

Subprogram >x5 - Acces direct output file

(DSK, WDS, SCS, IDE, HDX)

```
        +------------+------------+
>834C   | Unit#      | Code       |
        +------------+------------+
>834E   | Address of filename     |
        +------------+------------+
>8350   | Info       |
        +------------+
```

Subprogram >x7 - Set current pathname

(DSK on a HFDC card, WDS, SCS, IDE, HDX)

```
        +------------+
>834C   | Unit#      |
        +------------+------------+
>834E   | Address of pathname     |
        +------------+------------+
>8350   | Error code |
        +------------+
```

Subprogram >x8 - Create subdirectory

(DSK on a HFDC card, WDS, SCS, IDE, HDX)

```
        +------------+
>834C   | Unit#      |
        +------------+------------+
>834E   | Address of dir.name     |
        +------------+------------+
>8350   | Error code |
        +------------+
```

Subprogram >x9 - Delete subdirectory

(DSK on a HFDC card, WDS, SCS, IDE, HDX)

```
        +------------+
>834C   | Unit#      |
        +------------+------------+
>834E   | Pointer to dir.name     |
        +------------+------------+
>8350   | Error code |
        +------------+
```

Subprogram >xA - Directory rename routine

(DSK on a HFDC card, WDS, SCS, IDE, HDX)

```
        +------------+
>834C   | Unit#      |
        +------------+------------+
>834E   | Pointer to new dir.name |
        +-------------------------+
>8350   | Pointer to old dir.name |
        +-------------------------+
```

Subprogram >2C - Identify SCSI drive

(SCS)

```
        +------------+------------+
>834A   | Status     | Result     |
        +------------+------------+
>834C   | Unit#      | >00        |
        +------------+------------+
>834E   | Buffer start address    |
        +-------------------------+
>8350   | Error code |
        +------------+
```

Status - SCSI status byte (0=Ok ?)

Result - SCSI result (0=Ok ?)

Unit# - Indicates the SCSI-id (=drive number-1) for wich the operation has to be performed.

If the most significant bit of unit# is set, the buffer start address is assumed to be in CPU-ram instead of VDP-ram.

Buffer start address - Address for a 44-byte data packet:

Offset	Bytes	Description
0	8	SCSI identification
8	8	Manufacturer identification
16	16	Product identification
32	4	Product version and revision
36	4	Maximum number of sectors
40	4	Sector size

Subprogram >8C - Identify IDE drive

(IDE)

```
        +------------+
>834C   | Unit#      |
        +------------+------------+
>834E   | Buffer start address    |
        +-------------------------+
>8350   | Error code |
        +------------+
```

Unit# - Indicates the drive number for wich the operation has to be performed.

If the most significant bit of unit# is set, the buffer start address is assumed to be in CPU-ram instead of VDP-ram.

Buffer start address - Address for a 136-byte data packet:

```
    Offset  Bytes   Description
    ----------------------------------------------
    0   2   General configuration
            0x8000= "Non magnetic drive"
            0x4000= "Format speed tolerance required"
            0x2000= "Track offset option available"
            0x1000= "Data strobe offset option available"
            0x0800= "Rotational speed tolerance >0.5%"
            0x0400= "Disk transfer rate >10Mbs"
            0x0200= "Disk transfer rate >5Mbs but <10Mbs"
            0x0100= "Disk transfer rate <5Mbs"
            0x0080= "Removable cartridge drive"
            0x0040= "Fixed drive"
            0x0020= "Spindle motor control option"
            0x0010= "Head switch time >15 uSec"
            0x0008= "Not MFM encoded"
            0x0004= "Soft sectored"
            0x0002= "Hard sectored"
            0x0001= "Reserved"
    2   2   Number of cylinders
    4   2   reservered
    6   2   Number of heads
    8   2   Number of unformated bytes per track
    10  2   Number of unformated bytes per sector
    12  2   Sectors per track
    14  6   Vendor unique
    20  20  Serial number (20 ascii characters >0000=not specified)
    40  2   Buffer type
    42  2   Buffer size in 512 bytes increments (>0000=not specified)
    44  2   Number of ECC bytes available on read write long commands (>0000=not specified)
    46  8   Firmware revision (8 ascii characters >0000=not specified)
    54  40  Model number (40 ascii characters >0000=not specified)
    94  2   MSB Vendor unique
            LSB >00=Read/write multiple commands not implemented
                >xx=Maximum number of sectors that can be transferred per interrupt
                on read or write commands
    96  2   0x0000= Cannot not perform doubleword IO
            0x0001= Can perform doubleword IO
    98  2   Capabilities
            0x8000-0x0400 Vendor unique
            0x0200 1=LBA supported
            0x0100 1=DMA supported
            0x0080-0x0001 Vendor unique
    100 2   Reserved
    102 2   MSB PIO data transfer cycle timing mode
            LSB Vendor unique
    104 2   MSB DMA data transfer cycle timing mode
            LSB Vendor unique
    106 2   0x0001 1= fields reported in bytes 108-116 are valid
                   0= fields reported in bytes 108-116 may be valid
    108 2   Number of cylinders
    110 2   Number of heads
    112 2   Number of sectors per track
    114 4   Capacity in sectors
    118     0x8000-0x0200 reserved
            0x0100 1= Multiple sector setting is valid
            LSB    >xx= Current number of sectors that can be
```

```
                    transferred per interrupt on R/W multiple commands
    120 4   Total number of user addressable sectors (LBA mode only)
    124 2   MSB Single word DMA transfer mode active
            LSB Single word DMA transfer mode supported
    126 2   MSB Multiword DMA transer mode active
            LSB Multiword DMA transfer mode supported
    128 4   Partition offset (since IDEDSR V3.0)
    132 4   Partition size   (since IDEDSR V3.0)
```

Ti99-geek