
H9 BBH OKYES User Manual

General Description and Name

Device divided as 3 partitions, I, J, N.
Multiple partitions style BBM.
Block#0 used for system and store bad block information.

Relevant User Options

The following special features on the special features tab apply to this scheme. The default values might work in some cases but please make sure to set the right value according to your system.

Please note only the below special feature items are related to this scheme and ignore any others. If any of below items doesn't exist, please check whether the right version has been installed or contact Data I/O for support by submitting Device Support Request through this address:

<http://www.dataio.com/support/dsr.asp>

Bad Block Handling Type = "H9_BBH_OKYES"

Spare area = "Enabled"

Please refer to "Description of common NAND special features.pdf". Normally set as "Enabled" for this BBM. [Default 'Disabled']

Check BadBlock Marker in Data File : Please refer to "Description of common NAND special features.pdf". **Normally set as "Disabled" for this BBM.**[Default 'Enabled']

Pantech_Skip : Boot and DMSS Size (in blocks) = calculate from
image file. ['I' partition block number + 1]

Pantech_Skip : Data 1 Size (in blocks) = calculate from
image file. ['J' partition block number]

Pantech_Skip : Data 2 Size (in blocks) = calculate from
image file. ['N' partition block number]

Special Notes

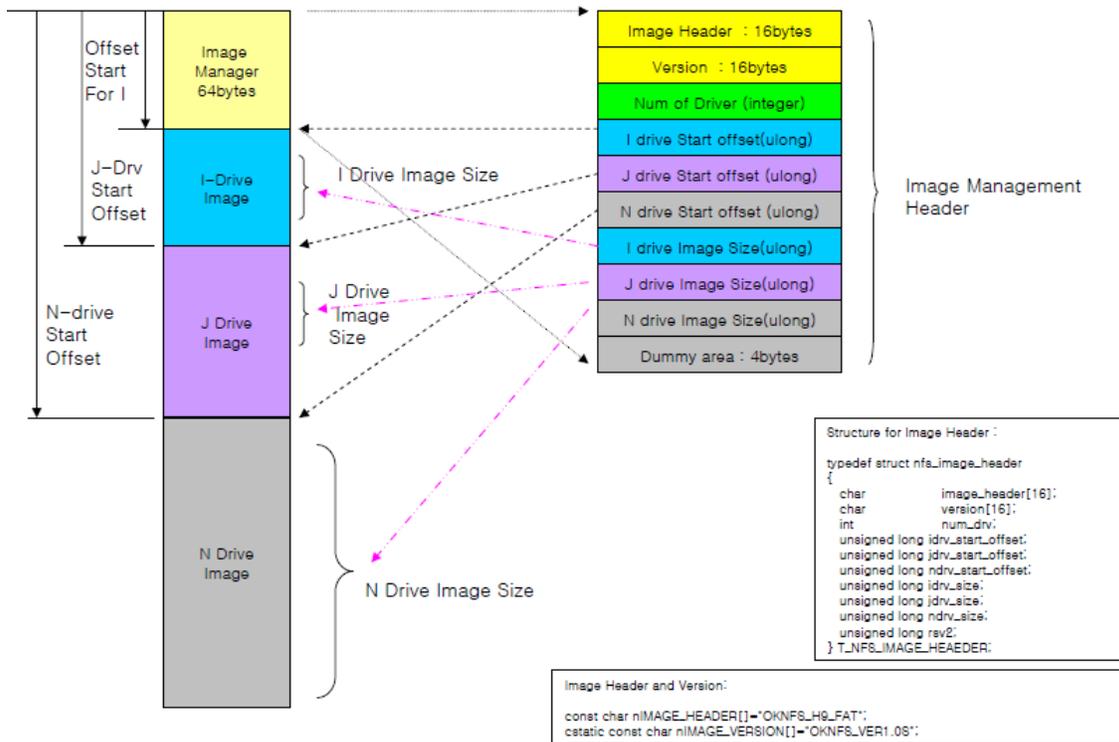
The image file consists like:

First 16KB are used for system management.

This BBM only use maximum 3 logical partitions (drive).

If any driver does not store image data, the bad block information of that area should be update into bad block management table.

Fill the unused values as 0xFF.



OKyes

Don't Distribute

The system management data structure is:

```
typedef struct _nand_system_block {
    char oknfs_f_version[512]; // 0.5K
    T_NFS_LDRVMNG b_lmng[2]; // 1K * 2
    T_BMTS_SECTOR b_bad[MAX_NAND_LOGICAL_DRV]; // 2K * 6
    unsigned char padding[1024]; // 1K
    char oknfs_r_version[512]; // 0.5K
} T_NFS_SYSTEM;
```

NAND bad block management table structure is:

```
// this structure must be 2K bytes
typedef struct _bmt_sector {
    char pre_id[8];
    int nBadSector;
    unsigned long bad_blocno[507];
    char pos_id[8];
} T_BMTS_SECTOR;
```

The File system Driver information structure is:

```
// this structure must be 1K
typedef struct _ldrv_system_mng {
```

```
unsigned long numLdrv;           // number of logical driver
unsigned long lblock_shift;      // logical block shift
unsigned long sector_shift;      // sector shift value
unsigned long dm1;
struct {
    unsigned long flag;          // active flag : bit0:1 active, 0:no active
    unsigned long sbn;          // start block number
    unsigned long nbn;          // number of block
    unsigned long year;
    unsigned long month;
    unsigned long date;
    unsigned long hour;
    unsigned long min;
    unsigned long sec;
    char label[128];             // volume lable
} b_info[MAX_NAND_LOGICAL_DRV];
char padding[24];
} T_NFS_LDRVMNG;
```

Revision History

V1.0 15-Sep-2011
Create this spec.

Appendix

You can get the file "Description of common NAND special features.pdf" from
<http://ftp.dataio.com/FCNotes/BBM/>