
Skip Avaya User Manual

General Description and Name

This scheme Implements the skip block method. Update the Partition table/ ECC in the first page of the device. .

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 = "Skip Avaya"

Spare area = "Enable"

PartitionTable File = The path of the partition table file on your PC.

Special Notes

- This BBM PC file should contain the OOB(spare areas).
- This BBM should set total partition number fix to 9. Other will fail.
- The last partition end should be not exceed the device blocks – reserved max bad blocks.(e.g. total 1024 block, 20 block reserved for bad blocks, then partition end should not exceed 1004).
- This BBM PC file should contain the correct ECC, except the first page can be ignore. Because this page ECC will be update when programing.
- Format of PartitionTable.mbn:
 - a. Binary file fixed length 256 bytes.
 - b. Organization:16 rows x 4 columns. Each table item is 32-bits, little endian byte ordering.
 - c. Each row of the table describes configuration for one partition. Up to 16 partitions can be used.
 - d. Partition configuration:
 - i. **Start Adr**: address of start of partition in flash blocks. The programmer will set the file read pointer and the programmer write pointer to Start Adr. If Start Adr=0xFFFFFFFF, skip to the next partition.
 - ii. **End Adr**: last valid block in the current partition. The last data block programmed must be equal to or less than End Adr, otherwise the programmer will reject the flash device.
 - iii. **Actual Data Length**: number of blocks of data to read from the input file and write to the flash in the current partition
 - iv. **Attribute**: ignore
 - v. **Please note to keep: Actual Data Length = End Adr - Start Adr + 1 for this BBM**
 - v. **Please note to keep: Data file should not exceed the last partition end.**

vi.Example PartitionTable.mbn

| NAND Flash Block | | | Attribute |
|------------------|------------|--------------------|------------|
| Start Adr | End Adr | Actual Data Length | |
| 0x0 | 0x7FF | 0x360 | 0xFFFFFFFF |
| 0x800 | 0xFFF | 0x30 | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |
| 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF | 0xFFFFFFFF |

file:

Revision History

- V1.0 Date: 2012-06-29
Create this spec.
- V2.0 Date: 2012-07-09
Add a special feature.
- V3.0 Date: 2012-07-16
Remove a special feature.
- V4.0 Date: 2012-07-16
Add partition file description.
- V5.0 Date: 2012-07-25
Add special feature.
- V6.0 Date: 2012-08-06
Add special notes.

Appendix

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