

## **Generic MPF NFC ECC IMX35 & BBT Bad Block Scheme**

This scheme is a Generic Multiple Partition format. It uses skip method for bad blocks handling within each partition. Bad blocks within any partition do not affect the location of the other partitions. The binary image is reorganized from large pages (4096/2048 bytes) to small pages (512 + 16 bytes), over-writing the manufacturer Bad Block marker area.

### **Warning –**

After the devices are programmed with this BBM, they can not be reused with any other BBM method since all the original manufacturers' bad block marker area are over-written by this BBM!

### **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 = “Generic MPF NFC\_ECC\_IMX35\_ BBT”

Spare Area = “Enabled”

- This option should be selected when the data file contains spare area information.

Check BB marker in dataFile = “Disabled”

- This setting is required since the image file contains information that will write over the manufacturer Bad Block marker area.

Bad block detection = “BBM specified”

Debug Messages in Eventlog = “On/Off”

- The level of debugging information printed in the eventlog.txt file. “On” is the recommended selection.

All other features are not used for this scheme.

### **Image Preparation**

The image data file should be the size of the device. The value of “FF” should be used to pad between each partition.

### **Exceptions**

- If any other bad Block management or skip is used with the devices that are pre-programmed with this BBM, they will fail because of too many bad blocks.

### **Partition Table Format Partition.mbn:**

- a. A binary file of YourFile.MBN with fixed length of 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.

### **Customers for this scheme**

- General, Freescale IMX processors

### **Revision History**

V1.0 Mar 3th, 2011

Create this spec.

### **Appendix**

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