Skip DBBT MPF BCH40 BCH18 iMX6 User Manual

This is a Multiple Partition format scheme. Block0-7 are required good blocks, the rest partitions use "Skip bad block" method within each partition. Any bad block found within any of the partitions will not cause relocation to the starting block of the follow up partitions.

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 does not 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 DBBT MPF BCH40 BCH18 iMX6"

Spare Area : "Disabled"

PartitionTable File : "C:\PartitionTable.mbn"

iMX DBBT Create? : "Disabled"

Error bits allowed in one page: How many error bits allowed within one page while verify, this depends on the features of NAND (SLC, MLC …). [Normally required, default is 0].

Required good block area: Start block : "0"
Required good block area: Number of blocks : "8"
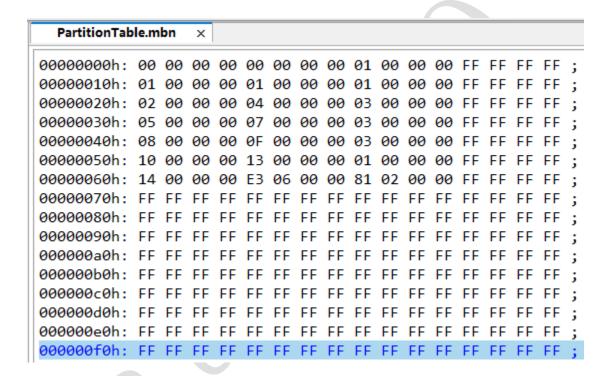Block0-7 are required good blocks

Image Preparation:

FCB and other partitions: raw data should be included in the binary image file at address offset of 0x00, do not conclude ECC data and Meta data.

Partition Table Format Partition.mbn
- A binary file of PartitionTable.mbn with fixed length of 256 bytes.
- Organization: 16 rows x 4 columns. Each table item is 32-bits, little endian byte ordering.
- Each row of the table describes configuration for one partition. Up to 16 partitions can be used.
- Partition configuration:

i. **Start Adr**: address of start of partition in flash block unit. If Start Adr is 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. **Note**: For optimal option, the following example should be used and the 3rd to the last partitions should be adjusted as needed.

```
 PartitionTable.mbn    ×

00000000h: 00 00 00 00 00 00 00 00 01 00 00 00 FF FF FF FF ;
00000010h: 01 00 00 00 01 00 00 00 01 00 00 00 FF FF FF FF ;
00000020h: 02 00 00 00 04 00 00 00 03 00 00 00 FF FF FF FF ;
00000030h: 05 00 00 00 07 00 00 00 03 00 00 00 FF FF FF FF ;
00000040h: 08 00 00 00 0F 00 00 00 03 00 00 00 FF FF FF FF ;
00000050h: 10 00 00 00 13 00 00 00 01 00 00 00 FF FF FF FF ;
00000060h: 14 00 00 00 E3 06 00 00 81 02 00 00 FF FF FF FF ;
00000070h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ;
00000080h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ;
00000090h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ;
000000a0h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ;
000000b0h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ;
000000c0h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ;
000000d0h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ;
000000e0h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ;
000000f0h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ;
```

Appendix

You can get the file "Description of common NAND Special Features.pdf" from
http://ftp.dataio.com/FCNotes/BBM/