
BBM Visteon User Manual

General Description and Name

BBM_Visteon

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 = "BBM_Visteon"

Spare Area = "**Enabled**" is the only valid option. This means the programmer expects the spare area data to be included in the image, and that it will program that data into the device.

PartitionTable File = C:\PartitionTable.mbn(default)

The path of the partition table file on your PC.

This BBM support maximum 256 partitions.

Required good block area: Start block = "0" This will require the entered block to be a valid block

Required good block area: Number of blocks = "0" This will be the total number of blocks required to be valid after the start block.

Several special features are needed if the customer needs to select different bad block mark.

They are as below:

Special BB Mark Valid = "**Enabled**" This is the switch allowing to set different bad block mark which is not the same as the device spec. Only when it is "Enabled", the different bad block mark position can be set from 0~63(2048Byte-Nand device) of the spare area. If it is "Disabled", the special bad block mark position setting is forbidden.

Bad Block Marker Offset = "0" This is decided by the customer where the bad block mark is located according to the customer data file. For example, if the customer wants to program data into position 0 of the spare area and regard position 6 as bad block mark(2048Byte-Nand device), then he can set this special feature as "6".

Bad Block Mark Masks = “0xFFFF” This is applied to a 16-bit NAND device. It is decided by the customer if the bad block mark is masked high 8 bits or low 8 bits. For example, if it is set as 0xFFFF, the all 16-bits of the bad block mark is checked. If “0x00FF”, only low 8-bit of the bad block mark is checked. If “0xFF00”, only high 8-bit of the bad block mark is checked.

The partition table example:

Start Block (4 bytes)	End Block (4 bytes)	Actual Data Block (4 bytes)
0x000	0x009	0x004
0x00A	0x00B	0x001
0x00C	0x00E	0x002
0x00F	0x011	0x000
0x012	0x09B	0x074
0x09C	0x0A0	0x004
0x0A1	0x0A2	0x000
0x0A3	0x0CF	0x000
0x0D0	0x0D1	0x000
0x0D2	0x7FF	0x16D
0xFFFFFFFF	0xFFFFFFFF	0xFFFFFFFF

Revision History

V1.0 11/06/2013

Appendix

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