

Description of common NAND special features

General Description

This is a conclusion of such special features which have been widely shared within NAND device supports, they might relate to device features or different handling options but not directly BBM related.

Glossary of Terms

BBM – Bad Block Management scheme

ECC – Error Correction Code

SLC - Single Level Cell technology

MLC – Multi-Level Cell technology

OOB Region – Out of Band region in a page. It is also called Spare Area.

semi vendor BB marker – Data located OOB region of a either the first or second page of a block in a device that the manufacturer uses to identify if a block is good or bad.

List of special features and options

Bad Block Handling Type

This item specify which BBM (Bad block management type) will be used.

[Always required, no default value]

Spare area

Disabled - Data file does not contain Spare Area data. TLWin will insert 0xFF into the Spare Area of data file.

Enabled - Data file contains Spare Area data.

ECC - TLWin will compute ECC based on your Bad Blocking Scheme.

Update ECC field - Data file contains Spare Area data but our system will update the ECC related field with proper value.

[Always required, default value not fixed.]

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 '0' if not listed or specified.]

Check BB Marker In DataFile

Whether check the bad block mark position of the customer data file or not. Every NAND chip has its own semi-vendor specified bad block mark located in spare area, if this item enabled, programmer system will check the contents of this marker location and throw error if not all 0xFF.

[Normally not required, default 'Enabled' if not listed or specified.]

Required good block area: Start block

Required good block area: Number of blocks

These two items specify a range of blocks which must all be good. The device will be failed if any bad blocks found within this area.

[Always required, default '0' and '0'.]

Force OTP Blank Check

Disable - Programmer system will do blankcheck on OTP area according to the programming flag of the corresponding sectors.

Enable - Programmer system will always do blankcheck on OTP area.

[Normally not required, default 'Disable' if not listed. Otherwise default value not fixed.]

Skip Free Pages

Disabled - Work as normal, all the data include blank pages will be programmed.

Enabled - The blank pages(all bytes with 0xFF data) of data file will skip programming operation while program function run.

[Normally not required, default 'Disabled'.]

Skip Last Free Space

Disabled - Work as normal, all the data include blank pages will be programmed.

Enabled - The continuous blank pages(all bytes with 0xFF data) located at the end of a block of data file will skip programming operation while program function run.

[Normally not required, default 'Disabled'.]

Only Cal ECC for User Data

YES - Work as normal, if set 'spare area' as 'ECC', static ECC will only be calculated for user data file.

NO - If set 'spare area' as 'ECC', static ECC will be calculated for all the NAND range.

[Normally not required, default 'YES'.]

Handle Empty Page

YES - Work as normal.

NO - Any pages which has all 0xFF within the data file will not be handled in the related DLL of TLwin, which means ECC field or any other tag locations will not be updated, just keep the whole page includes the OOB(might be inserted according to the settings of 'spare area') as all 0xFF.

[Normally not required, default 'YES'.]

Fill00 to Initial BB

Disabled - Not change the original(semi-vendor marked) bad blocks.

Enabled - Program the semi-vendor specified bad block marker existed page(s) to all 0x00 for all original bad blocks. PLEASE NOTE PROGRAMMER SYSTEM WILL ONLY DO THE PROGRAMMING OPERATION BUT WILL NOT CHECK THE RESULT!

[Normally not required, default 'Disabled'.]

bad block detection

This item is useful when customer needs to move the bad block marker location or want to override default bad block markers but use bad block table to record the information. Then we can only use the BBM specified way to detect the bad blocks while reprogramming.

semi vendor BB marker - The normal way, found the bad block according to semi-vendor specified bad block marker. Actually the algorithm will work as earlier source code if this option selected.

BBM specified - This option will give the detection work to BBM. So, it can use any bad block marker location even in main area or bad block table to detect the bad blocks.

BBM then BB marker - Firstly work as “*BBM specified*”, if it failed to find the bad block information, then work as “*semi vendor BB marker*” to find the bad blocks.
[Normally not required, default ‘semi vendor BB marker’.]

BB: max allowed

How many bad blocks allowed for this chip. BBM can override this setting.
[Normally not required, default as semi vendor specified.]

BB: mark position

BB: mark mask

Specify bad block mark position within spare area and its mask. Position is 0 based and LOGIC_1 of the mask means this bit is not care. BBM can override this setting.
[Normally not required, default as semi vendor specified.]

forcely close safety ID check

YES - If ‘Electronic ID Check’ under ‘Process’ tag is also unchecked, the system will not do any ID check operation. This option is not suggested, however, it gives the possibility if it’s really needed in some special & urgent case.

NO - Work as normal.

[Normally not required, default ‘NO’.]

erase error handling

program error handling

These two items only work if we had enabled the related function in the support, they can specify how to handle erase and program error. PLEASE NOTE WE WILL NOT ENABLE THE RELATED FUNCTIONS UNLESS CUSTOMER REQUIRED AND TOTALLY UNDERSTAND THE RISK AND AFFORD RESULT!!

fail the chip - fail the chip if any erase or program error met.

mark as BB and fail - mark the error occurred block of the chip as bad block and then fail the chip.

mark as BB and go on - mark the error occurred of the chip as bad block and continue process.

[Normally not required, default always ‘fail the chip’.]

ERASE AFTER PROGRAM

Actually this item not only related to NAND devices. If enabled, our programmer system will erase the whole chip after programming. This is mainly for flash testing usage.
[Normally not required, default 'Disabled' if not listed or specified.]

Hardware ECC

It is for OneNAND devices which has Hardware ECC calculation capability.

Disabled - Hardware ECC calculation is not allowed

Enabled - Hardware ECC calculation is allowed.

[Normally required for OneNAND devices, default 'Enabled'.]

Special Notes

As devices have different features and customer have different requirement, not all these listed special features have been added for every device.

History

V1.01

Dec 9th, 2010, added Handle Empty Page

V1.02

Jan 6th, 2011, added forcely close safety ID check