Freescale Semiconductor
Application Note

Document Number: AN3986
Rev. 0, 02/2010

Changing the i.MX51 NAND Flash Model for Windows Embedded CE^{TM} 6.0

by Multimedia Applications Division
Freescale Semiconductor, Inc.
Austin, TX

This application note describes how to incorporate a new NAND Flash model in a custom design using the i.MX51 and Windows Embedded CE 6.0.

Most of the custom boards incorporate several components that are not fully compatible with the Freescale Board Support Package (BSP). Frequently, new i.MX hardware designs use different components than those used on Freescale development tool. In general, memory suppliers promote changes on their product portfolios very often. Therefore, memories like DDR, NAND Flash, and NOR Flash are often replaced by models which are not supported by Freescale BSP drivers. This application note is intended to be a guidance to include new NAND Flash models to the Freescale BSP.

Contents

Figure 1 shows the driver architecture of WinCE 6.0 and Eboot on NAND Flash implementation.

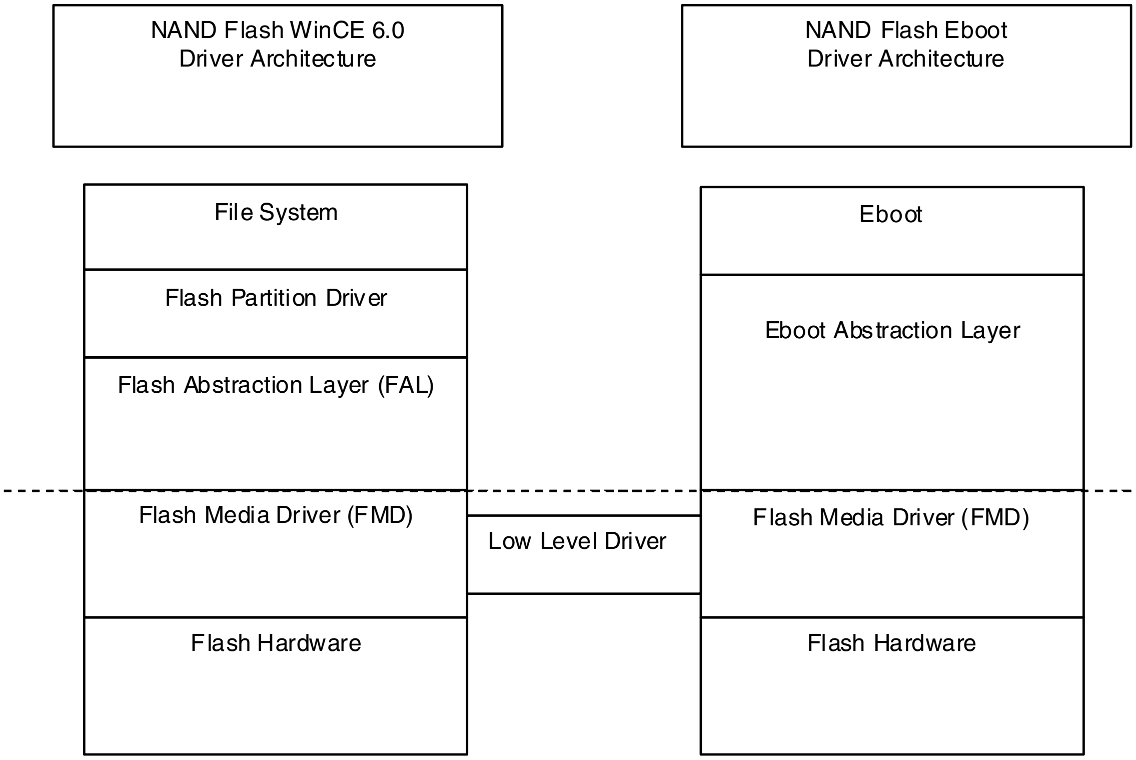


Figure 1. Driver Architecture of WinCE 6.0 and Eboot on NAND Flash Implementation

WinCE and Eboot share the same low level driver on the NAND Flash implementation. When a new memory model is being incorporated, all the changes have to be included under the Flash Media Driver (FMD) layer. After building this layer, both Eboot and NK images support the new Flash memory.

1 Implementation

The following section explains how to implement the NAND Flash.

1.1 NAND Flash Parameter Definition

The following are parameter definitions which have to be configured when a new NAND Flash is added. These parameters can be found on the NAND Flash specification document.

- NAND MARKER—Manufacture's ID code
- NAND DEVICE ID—Device ID code
- NAND BLOCK COUNT—Total number of blocks on the Flash
- NAND PAGE COUNT—Total number of pages on the Flash
- NAND PAGE SIZE—Size of each page
- NAND SPARE SIZE—Total size of spare area in one page
- NAND BUS WIDTH—Size of data bus

- BBI MAIN ADDR—Address page where the BBI is located. See Section A.2, “BBI MAIN ADDR.”
- BBI NUM—Number of pages in a block that contains the BBI. See Section A.3, “BBI NUM.”
- BBIMarkPage[]—Defines the page(s) where the BBI in a block. See Section A.4, “BBIMarkPage.”

1.2 Inclusion of New NAND Flash Models

The following steps show how to include a new NAND Flash model by modifying the FMD files:

NOTE

Ensure that i.MX51 BSP1.6 is installed on the Visual Studio 2005.

1. Extract the following information from the NAND Flash specification. For more information, see Section 1.1, “NAND Flash Parameter Definition.”
   — NAND MARKER
   — NAND DEVICE ID
   — NAND BLOCK COUNT
   — NAND PAGE COUNT
   — NAND PAGE SIZE
   — NAND SPARE SIZE
   — NAND BUS WIDTH
   — BBI MAIN ADDR
   — BBI NUM
   — BBIMarkPage
2. Create a new header file with the following information below, name it with the NAND Flash part number. (For example. `K9LAG08U0M.h`), and store the file in the following location:
   `\WINCE600\PLATFORM\COMMON\SRC\SOC\COMMON_FSL_V2_PDK1_6\NAND\INC\`

   The empty values marked with X and arrows are the ones that should be filled with the information extracted from the NAND Flash model specification.

```
#ifndef __K9LAG08U0M_H__ → __Header File Name_H__
#define __K9LAG08U0M_H__ → __Header File Name_H__

// NAND Flash Chip CMD
#define CMD_READID              (0x90)      // Read ID
#define CMD_READ                (0x00)      // Read data 1st cycle
#define CMD_READ2               (0x30)      // Read data 2nd cycle
#define CMD_RESET               (0xFF)      // Reset
#define CMD_ERASE               (0x60)      // Erase setup
#define CMD_ERASE2              (0xD0)      // Erase
#define CMD_WRITE               (0x80)      // Sequential data input
#define CMD_WRITE2              (0x10)      // Program
#define CMD_STATUS              (0x70)      // Read status

// NAND Flash Chip Size
```

```
#define NAND_BLOCK_CNT          (XXXX) → NAND BLOCK COUNT
#define NAND_PAGE_CNT           (XXX) → NAND PAGE COUNT
#define NAND_PAGE_SIZE          (XXXX) → NAND PAGE SIZE
#define NAND_SPARE_SIZE         (XX) → NAND SPARE SIZE
#define NAND_BUS_WIDTH          (X) → BUS WIDTH

// NAND Flash Chip
#define NAND_NUM_OF_CS          (1)

// NAND Flash Chip ID

#define NAND_MAKER_CODE         (0xXX)→ NAND MARKER
#define NAND_DEVICE_CODE        (0xXX)→ NAND DEVICE ID
#define NAND_ID_CODE            ((NAND_DEVICE_CODE << 8) | NAND_MAKER_CODE)

// NAND Flash Chip Operation Status

#define NAND_STATUS_ERROR_BIT   (0)      // Status Bit0 indicates error
#define NAND_STATUS_BUSY_BIT    (6)      // Status Bit6 indicates busy

// SWAP BBI

#define BBI_MAIN_ADDR           (XXX)→  BBI MAIN ADDR
#define BBI_NUM                 (X)→  BBI NUM
BYTE    BBIMarkPage[1] = {XXX}; → BBIMarkPage

#endif
```

3. Modify the header file `nandbsp.h`, located at: `\WINCE600\PLATFORM\iMX51-EVK-PDK1_6\SRC\COMMON\NANDFMD\nandbsp.h`, and add the `created.h`. The following is the modified example and the changes are marked in bold:

```
//
// Copyright (c) Microsoft Corporation.  All rights reserved.
//
//
// Use of this source code is subject to the terms of the Microsoft end-user
// license agreement (EULA) under which you licensed this SOFTWARE PRODUCT.
// If you did not accept the terms of the EULA, you are not authorized to use
// this source code. For a copy of the EULA, please see the LICENSE.RTF on your
// install media.
//
//-----------------------------------------------------------------------------
//
//  Copyright (C) 2007-2008, Freescale Semiconductor, Inc. All Rights Reserved.
//  THIS SOURCE CODE, AND ITS USE AND DISTRIBUTION, IS SUBJECT TO THE TERMS
//  AND CONDITIONS OF THE APPLICABLE LICENSE AGREEMENT
//
//-----------------------------------------------------------------------------
//
//  File:  nandbsp.h
//
//  Contains definitions for FMD impletation of the SoC NAND flash controller
//  and NAND memory device.
//
//-----------------------------------------------------------------------------
#ifndef __NANDBSP_H__
#define __NANDBSP_H__
```

```
#ifdef BSP_NAND_MT29F32G08
#include "MT29F32G08.h"
#elif BSP_NAND_K9MCG08U5M
#include "K9MCG08U5M.h"
#elif BSP_NAND_K9LBG08U0M
#include "K9LBG08U0M.h"
#elif BSP_NAND_K9LAG08U0M
#include "K9LAG08U0M.h"
#elif BSP_NAND_K9G8G08U0M
#include "K9G8G08U0M.h"
#elif BSP_SI_VER_TO2
#include "MT29F32G08.h"
#else
#include "K9G8G08U0M.h"
#endif
```

4. Edit the sources files located at: `\WINCE600\PLATFORM\iMX51-EVK-PDK1_6\SRC\COMMON\NANDFMD\DRIVER\sources` and `\WINCE600\PLATFORM\iMX51-EVK-PDK1_6\SRC\COMMON\NANDFMD\BOOT \sources`. Both files have the definitions, of which INCLUDE must be selected on the `nandbsp.h` file. Based on the same memory model example, the file must include the following lines of code. The changes are marked in bold.

```
\WINCE600\PLATFORM\iMX51-EVK-PDK1_6\SRC\COMMON\NANDFMD\DRIVER\sources

!if 0
Copyright (c) Microsoft Corporation.  All rights reserved.
!endif
!if 0
Use of this source code is subject to the terms of the Microsoft end-user license agreement
(EULA) under which you licensed this SOFTWARE PRODUCT. If you did not accept the terms
of the EULA, you are not authorized to use this source code. For a copy of the EULA,
please see the LICENSE.RTF on your install media.

Copyright (C) 2007-2008, Freescale Semiconductor, Inc. All Rights Reserved.

THIS SOURCE CODE, AND ITS USE AND DISTRIBUTION, IS SUBJECT TO THE TERMS AND CONDITIONS
OF THE APPLICABLE LICENSE AGREEMENT
!endif
!IF 0

Module Name:

    sources.

Abstract:

This file specifies the target component being built and the list of sources files needed
to build that component.  Also specifies optional compiler switches and libraries that
are unique for the component being built.

!ENDIF

TARGETNAME=nandfmd_driver_lib
TARGETTYPE=LIBRARY
RELEASETYPE=PLATFORM
SYNCHRONIZE_BLOCK=1

WINCEOEM=1
WINCECPU=1
NOMIPS16CODE=1
```

```
INCLUDES=\$
    (INCLUDES); \
    $(_PUBLICROOT)\common\oak\drivers\block\msflashfmd\inc; \
    $(_PLATFORMROOT)\common\src\soc\$(_COMMONSOCDIR)\nand\inc

!IF "$(BSP_NAND_K9G8G08U0M)" == "1"
CDEFINES=$(CDEFINES) -DBSP_NAND_K9G8G08U0M
!ENDIF

!IF "$(BSP_NAND_K9LBG08U0M)" == "1"
CDEFINES=$(CDEFINES) -DBSP_NAND_K9LBG08U0M
!ENDIF
```

!IF "$(BSP_NAND_K9LAG08U0M)" == "1"** → Enviroment variable to select the memory to be used. This has to be named the same as in the **CDEFINES=** **$(CDEFINES) -DBSP_NAND_K9LAG08U0M → Modify the last part with the name model just added on the nandbsp.h file.

!ENDIF → End of IF definition

```
!IF "$(BSP_NAND_K9MCG08U5M)" == "1"
CDEFINES=$(CDEFINES) -DBSP_NAND_K9MCG08U5M
!ENDIF

!IF "$(BSP_NAND_MT29F32G08)" == "1"
CDEFINES=$(CDEFINES) -DBSP_NAND_MT29F32G08
!ENDIF

!IF 1
CDEFINES=$(CDEFINES) -DINTERLEAVE_MODE
!ENDIF
SOURCES=../nandbsp.cpp
```

```
\WINCE600\PLATFORM\iMX51-EVK-PDK1_6\SRC\COMMON\NANDFMD\BOOT\sources

!if 0
Copyright (c) Microsoft Corporation.  All rights reserved.
!endif
!if 0
Use of this source code is subject to the terms of the Microsoft end-user license agreement
(EULA) under which you licensed this SOFTWARE PRODUCT. If you did not accept the terms
of the EULA, you are not authorized to use this source code. For a copy of the EULA,
please see the LICENSE.RTF on your install media.

Copyright (C) 2007-2008, Freescale Semiconductor, Inc. All Rights Reserved.
THIS SOURCE CODE, AND ITS USE AND DISTRIBUTION, IS SUBJECT TO THE TERMS AND CONDITIONS
OF THE APPLICABLE LICENSE AGREEMENT
!endif
!IF 0

Module Name:

    sources.

Abstract:

    This file specifies the target component being built and the list of sources files
needed to build that component.  Also specifies optional compiler switches and libraries
that are unique for the component being built.
!ENDIF
```

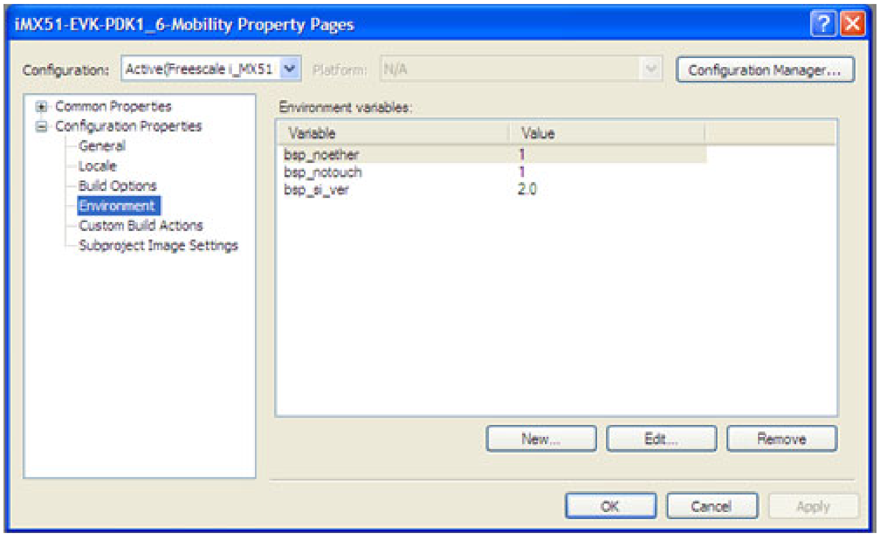
```
TARGETNAME=nandfmd_boot_lib
TARGETTYPE=LIBRARY
RELEASETYPE=PLATFORM
SYNCHRONIZE_BLOCK=1

WINCEOEM=1
WINCECPU=1
NOMIPS16CODE=1

INCLUDES=\
    $(INCLUDES); \
    $(_PUBLICROOT)\common\oak\drivers\block\msflashfmd\inc; \
    $(_PLATFORMROOT)\common\src\soc\$(_COMMONSOCDIR)\nand\inc

!IF "$(BSP_NAND_K9G8G08U0M)" == "1"
CDEFINES=$(CDEFINES) -DBSP_NAND_K9G8G08U0M
!ENDIF

!IF "$(BSP_NAND_K9LBG08U0M)" == "1"
CDEFINES=$(CDEFINES) -DBSP_NAND_K9LBG08U0M
!ENDIF

!IF "$(BSP_NAND_K9LAG08U0M)" == "1" → Enviroment variable to select the memory to be
used. This has to be named the same as in the
CDEFINES=$(CDEFINES) -DBSP_NAND_K9LAG08U0M → Modify the last part with the name model
just added on the nandbsp.h file.

!ENDIF → End of IF definition

!IF "$(BSP_NAND_K9MCG08U5M)" == "1"
CDEFINES=$(CDEFINES) -DBSP_NAND_K9MCG08U5M
!ENDIF

!IF "$(BSP_NAND_MT29F32G08)" == "1"
CDEFINES=$(CDEFINES) -DBSP_NAND_MT29F32G08
!ENDIF

!IF 1
CDEFINES=$(CDEFINES) -DINTERLEAVE_MODE
!ENDIF

SOURCES=../nandbsp.cpp
```

1.3 Configuring the Environment

To configure the environment, perform the following steps:

1. Add to the project the new NAND model environment variable.
    a) In Platform Builder, go to Build > Properties > Configuration Properties > Environment > New.

Figure 2 shows how to add an environment variable to a project.



Configuration: Active(Freescale i_MX51

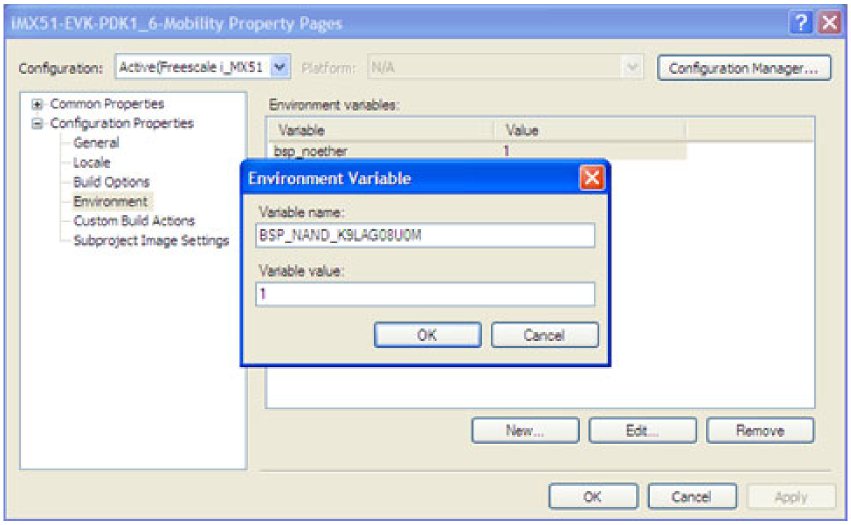
Platform: N/A

Environment variables:

Variable	Value
bsp_noether	1
bsp_notouch	1
bsp_si_ver	2.0

Figure 2. Adding Environment Variable to the Project

2. In the Variable name field, enter the variable defined in the source file which was modified in step 4 in Section 1.2, “Inclusion of New NAND Flash Models.” In the example shown in Figure 3, the Variable name is BSP_NAND_K9LAG08U0M. The variable value must be equal to 1.



Configuration: Active(Freescale i_MX51

Platform: N/A

Environment variables:

Variable	Value
bsp_noether	1

Variable name: BSP_NAND_K9LAG08U0M

Variable value: 1

Figure 3. Entering Variable Name and Value

3. Build the project within the FMD changes.

4. In Platform Builder, go to Build > Advanced Build Commands > Build Current BSP and Subprojects as shown in Figure 4. This process builds the image and the bootloader as well, since both share the same driver.

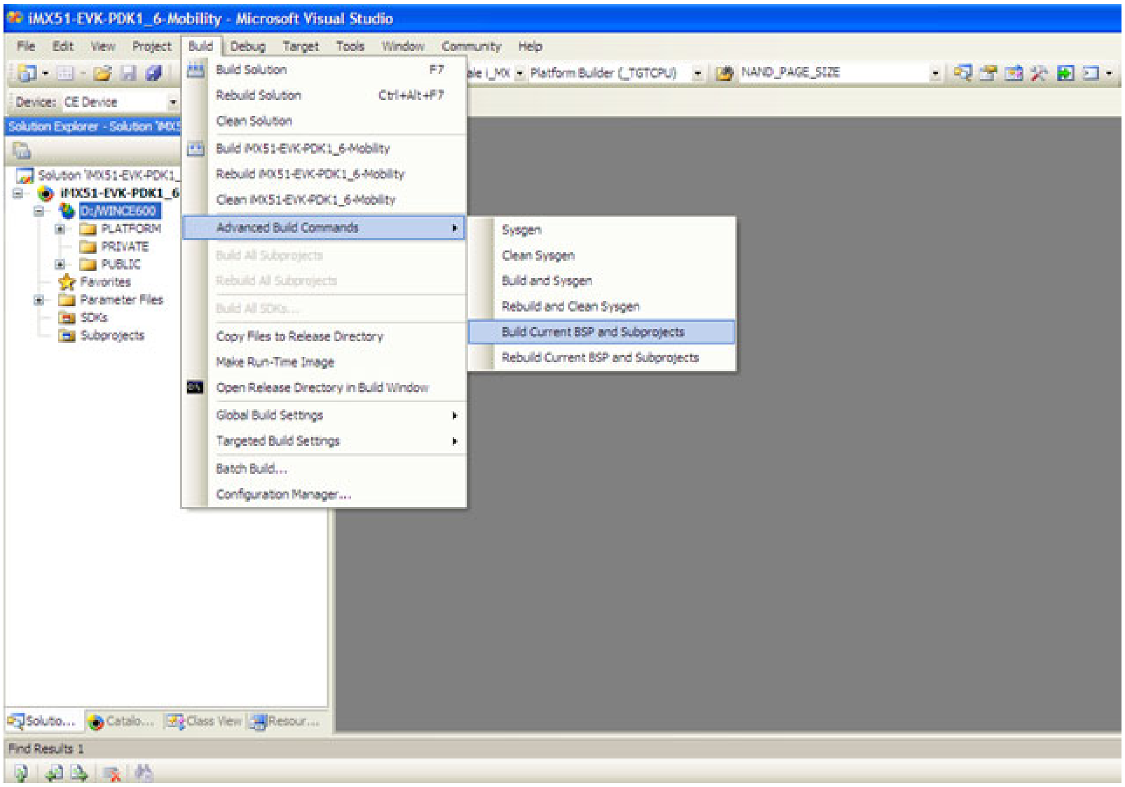


Figure 4. Building Current BSP and Subprojects

NOTE

To download the bootloader and image into a blank NAND Flash, see the Chapter 5 and 6 of the *i.MX51 EVK 1.6 Windows Embedded CE 6.0 User Guide*. This document provides guidelines to program the design through the ATK application.

2 Revision History

Table 1 provides a revision history for this application note.

Table 1. Document Revision History

Rev. Number	Date	Substantive Change(s)
0	02/2010	Initial release.

Appendix A NAND Flash Parameters

The following section explains about the NAND Flash parameters.

A.1 Bad Block Information (BBI)

Every NAND Flash defines a BBI byte(s). This information contains the invalid block(s). When the Near Field Communication (NFC) reads or writes to a specific NAND Flash, it first checks the BBI byte to validate the block that is used. The location of this BBI byte depends on the page size and the spare area of each NAND Flash model.

The following parameters defined by the NAND Flash driver for WinCE 6.0, are used to store the BBI:

- BBI MAIN ADDR
- BBI NUM
- BBIMarkPage

A.2 BBI MAIN ADDR

The BBI MAIN ADDR defines the address location on the page where the BBI is stored. This address depends on the page size and the number of bytes on the spare area. Normally, the BBI is stored on the first byte of the spare area. The following section explains how to get the BBI address:

- Using 2 KB page size with 64 bytes of spare area.

  Consider the first byte of the spare area on this NAND Flash model is 2048. The NFC splits the 2 KB page into 4 pages of $(512 + 16)$ with its corresponding spare area which are 16 bytes. According to the NFC format, the value of the address is $2048 - ((512 + 16) * 3) = 464$.

  2048—First byte of the spare area on the NAND Flash

  512—Page size on the internal NFC

  16—Spare area size of each internal buffer on NFC

  3—Represent the last buffer of NFC minus one, to reach the last buffer of the NAND Flash
- Using 4 KB page size with 218 bytes of spare area.

  Consider the first byte of the spare area on this NAND Flash model is 4096. The NFC splits the 4 KB page into 8 pages of $(512 + 26)$ with its corresponding spare area which are 26 bytes. According to the NFC format, the value of the address should be $4096 - ((512 + 26) * 7) = 330$.

  4096—First byte of the spare area on the NAND Flash

  512—Page size on the internal NFC

  26—Spare area size of each internal buffer on NFC

  7—Represent the last buffer of NFC minus one, to reach the last buffer of the NAND Flash

The following are the most common values of BBI MAIN ADDR:

```
#DEFINE BBI_MAIN_ADDR (464) //Use this for 2k page NAND
#DEFINE BBI_MAIN_ADDR (400) //Use this for 4k page 128 byte spare NAND, 4bit ECC
#DEFINE BBI_MAIN_ADDR (330) //Use this for 4k page 218 byte spare NAND, 8bit ECC
```

A.3 BBI NUM

The BBI NUM handles the number of pages in a block to check for BBI, usually the BBI NUM is equal to 1, however some vendors use 2. For more information, see the NAND Flash manual of each model.

A.4 BBIMarkPage

The BBIMarkPage is an array that contains the address page where the BBI is checked. Usually this page is the first or the last page. For instance, on a 128 page per block NAND Flash model, this value is either 0 or 127 for first or last page. If two or more BBI bytes are used in a page, the BBIMarkPage array must contain all these pages.

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or
+1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku
Tokyo 153-0064
Japan
0120 191014 or
+81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor
Literature Distribution Center
1-800 441-2447 or
+1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor
@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals” must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.