Freescale™ DragonBall™ MX1 Adaptation for NAND Flash Memory

MT29F2G08/MT29F2G16

Overview

NAND Flash devices have quickly become the preferred solution for high-density non-volatile memory storage, particularly in mobile product applications where size and power are important considerations.

While various processors may not include a NAND Flash controller, it is possible to interface NAND Flash devices to almost any processor. This is the case with the Freescale™ DragonBall™ MC9328MX1 application processor ("MX1"). By utilizing low-level software drivers and the MX1 address/data/control pins, the MX1 can communicate with the NAND Flash device and use its full range of capabilities. This technical note introduces hardware and software methods for implementing this approach and assumes a general understanding of the Micron 2Gb NAND Flash device on the part of the reader. Figure 1 on page 3 shows a typical hardware interface for connecting non-native processors to NAND Flash. Implementation results described in this technical note are derived from the setup and configuration shown in Figure 2 on page 4.

Detailed information on Micron NAND Flash devices can be found on the Micron Web site at www.micron.com/products/nand. The methods discussed in this document apply to the MX1 applications processor as well as to numerous other platforms.

It is important to note that this discussion of hardware and software to produce a viable NAND Flash interface does not reflect the use of error correction code (ECC), block management, or wear-leveling methods—all essential elements for a robust NAND Flash implementation, which are discussed in TN-29-17 at www.micron.com/products/nand/technotes.

Products and specifications discussed herein are for evaluation and reference purposes only and are subject to change by Micron without notice. Products are only warranted by Micron to meet Micron's production data sheet specifications. All information discussed herein is provided on an "as is" basis, without warranties of any kind.

MX1 Applications Processor

The MX1 applications processor targets a wide variety of mobile applications, including MP3 players, “smart” phones, and handheld devices. The processor is powered by an ARM9™ microprocessor core, with numerous added features that make the MX1 a key component of many mobile applications (see “Reference Materials” on page 15).

MX1 Memory and Storage Requirements

NAND Flash is ideal for MX1 designs, as it meets the high-volume memory/storage requirements of MX1 target applications; however, as noted previously, the MX1 applications processor does not include an embedded NAND Flash interface to communicate with NAND Flash devices.

MX1/NAND Flash Interface

Available MX1 resources can be used to create a glueless NAND Flash interface, with address/data pins serving as the I/O and control lines for the NAND Flash device. With the hardware connections in place, a software interface is required for communication between the NAND Flash device and the MX1. Low-level software and drivers configure the MX1 I/O pins to be used for the various NAND Flash functions. The software simulates a NAND Flash interface by executing PROGRAM, READ, and ERASE functions.

In the Micron NAND Flash implementation with the MX1, the chip select (CS#) control register is set to select 5 wait states, yielding a 62ns NAND Flash cycle time. A dedicated NAND Flash controller could be expected to adjust the NAND Flash interface signal routing such that a device timing closer to 50ns would be achieved. The fastest NAND Flash timing established for tested devices is 50ns.

NAND Flash/Processor Communications

Selecting a processor with built-in NAND Flash support gives designers and users the best performance and the greatest ease of implementation. When built-in NAND Flash support is not an option, interfacing directly to NAND Flash is the next-best alternative. As mentioned previously, in such implementations, software, hardware, ECC, and block management are necessary to ensure a reliable storage solution.

Hardware Interface

The hardware interface to NAND Flash is straightforward, as shown in Figure 1 on page 3. As all commands, addresses, and data share the NAND Flash I/O pins, it is easy to connect the pins directly to the processor’s data bus and devise a convenient, memory-mapped interface for the processor.

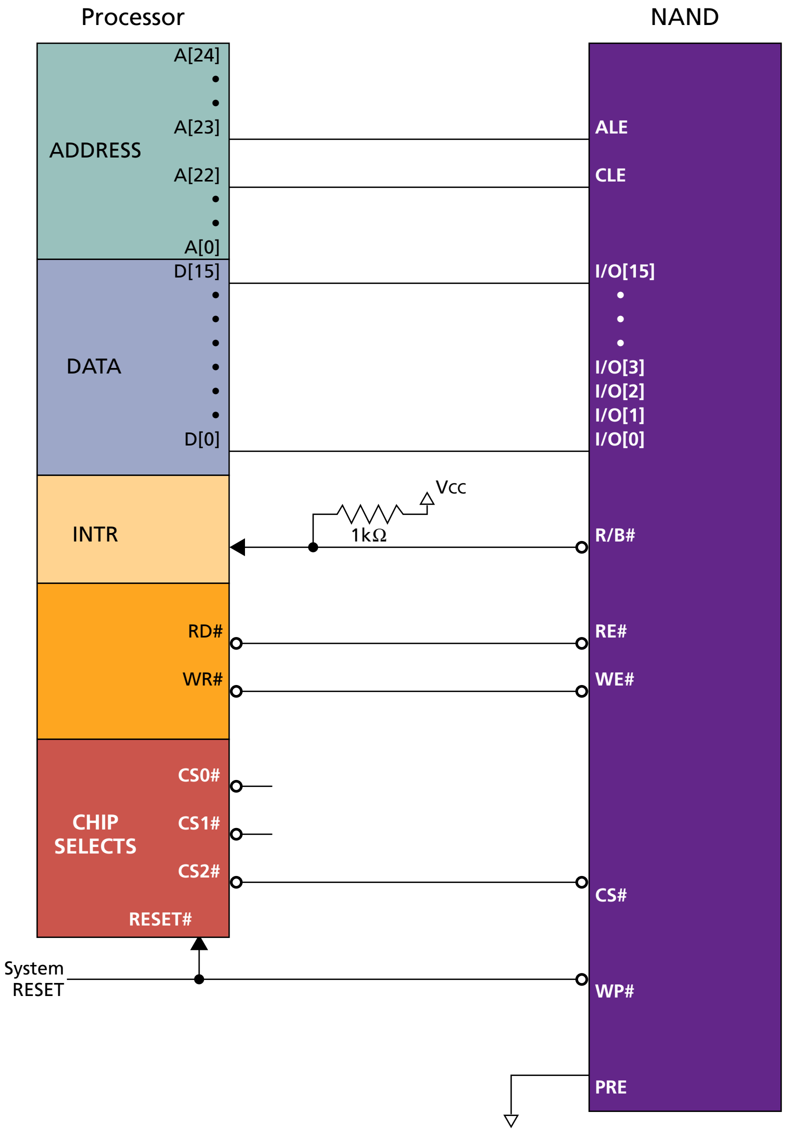
The NAND Flash control signals (ALE and CLE) are connected to address lines on the processor. Address line selection is determined by the overall architecture of the processor and the system software.

In the example provided, ALE is connected to A23; CLE is connected to A22. The NAND Flash RE# and WE# are generally connected to the appropriate READ and WRITE control lines of the processor. The NAND Flash CE# signal is connected to a memory-mapped processor chip select. It is important to ensure that CE# is selected for all NAND Flash transactions.

R/B# from the NAND Flash device can be connected to an interrupt input on the processor. As this is an open-drain signal, a pull-up resistor (usually 1kΩ) is required. WP# can be connected to the processor RESET (L) signal. The PRE signal from the NAND Flash device is not used in this example and is tied to ground (GND).

Low-level software is required to complete the various command, address, and data cycles. Low-level driver functions are described in “Software Communication” on page 9.

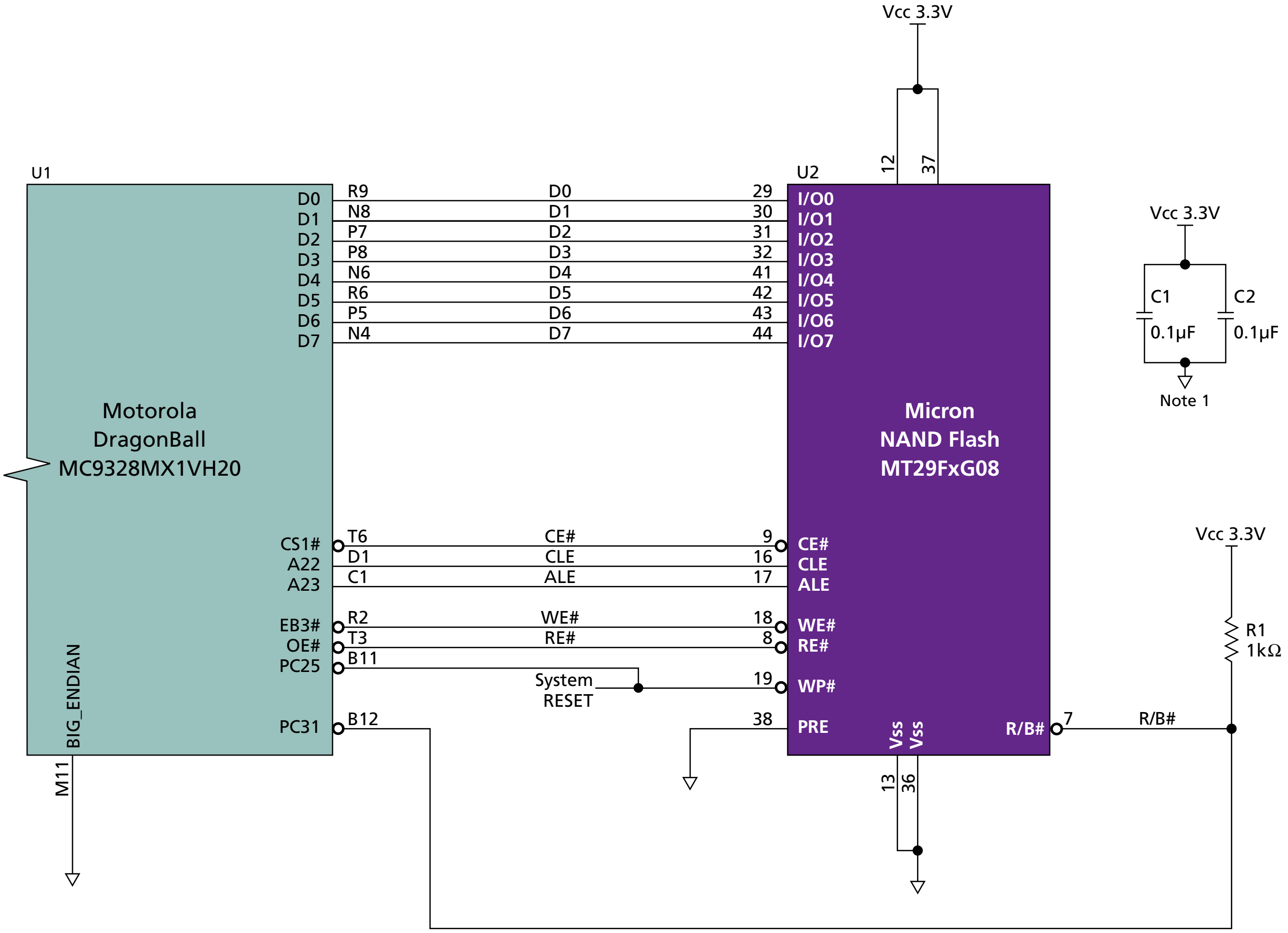
Figure 1: General Block Diagram Connecting Non-Native Processors to NAND Flash Device



Hardware Communication

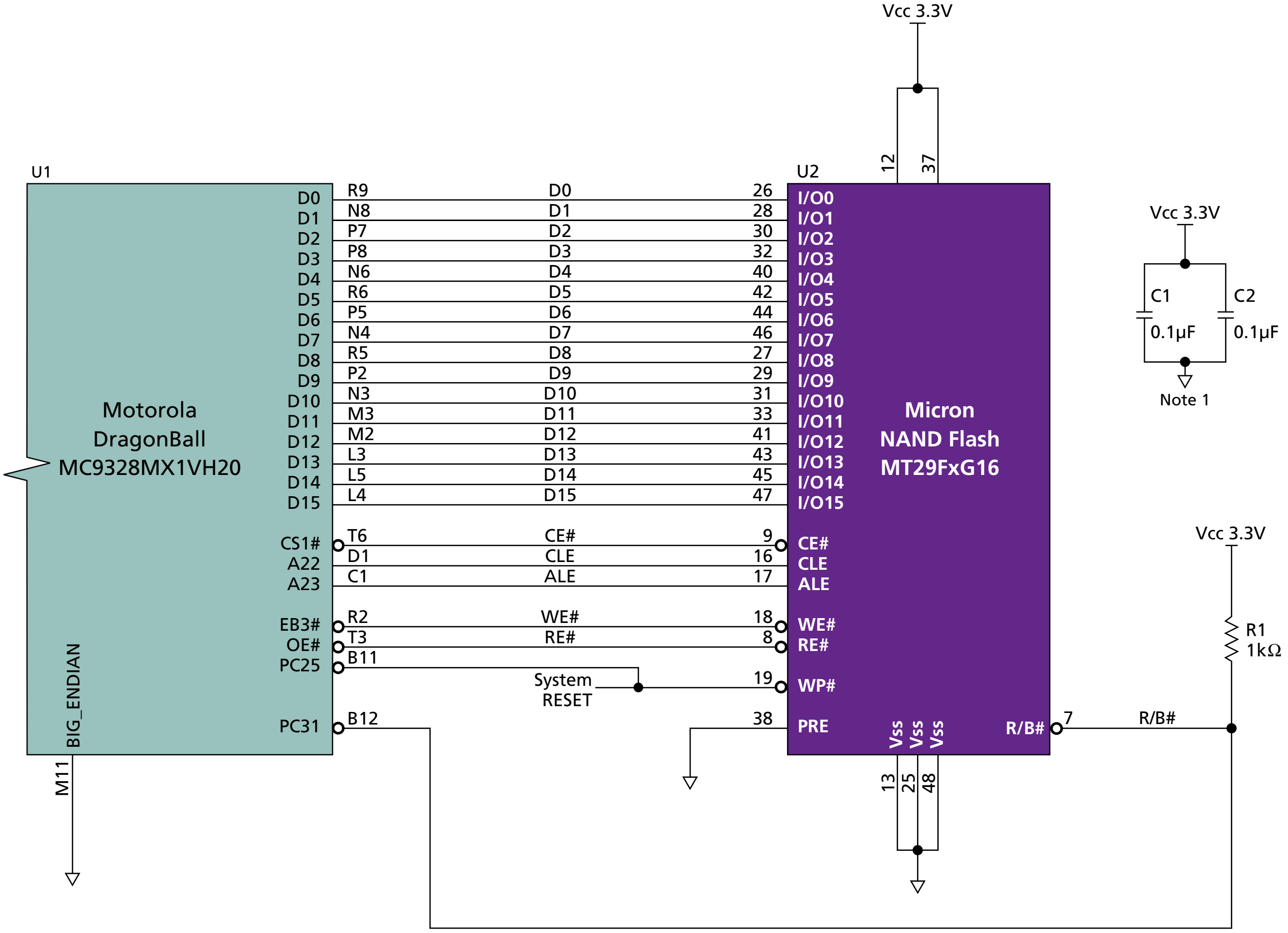
Figure 2 shows the hardware connections Micron used for communication between the MX1 and a x8 Micron NAND Flash device; Figure 3 on page 5 shows the connections for a x16 device. These connections also apply for a 4Gb NAND Flash device; however, the x8 pinout differs from the x16 pinout, which must be taken into account.

Figure 2: Micron NAND Flash and MX1 Hardware Connections (x8)



Notes: 1. Place C1 and C2 near U2 pins 12 and 37, respectively.

Figure 3: Micron NAND Flash and MX1 Hardware Connections (x16)



Notes: 1. Place C1 and C2 near U2 pins 12 and 37, respectively.

MX1 Applications Processor Signals

The MX1 application processor external interface module (EIM) provides up to six chip selects (CS#[5:0]), which support a variety of memory devices. All of the chip selects can be configured to support standard SRAM-style interfaces, and two of the chip selects can be programmed for an SDRAM-type interface.

Each of the EIM chip selects can also be programmed to support 8-bit, 16-bit, or 32-bit memory data paths. CS0# can provide the system boot memory interface.

The MC9328MX1 EIM interface signals used to connect to the Micron NAND Flash device in this Micron application are shown in Table 1.

Table 1: MX1 Signal Connections to Micron NAND Flash

MX1 Signal	MX1 Signal Description	Micron NAND Flash Signals
A[23:22]	Address bus: Outputs access external memory or I/O devices.	ALE, CLE

D[7:0]
or
D[15:0]
	Data bus: Bidirectional data bus transfers information between the processor and external memory or I/O devices.	I/O[7:0]
		for MT29F2G08
		I/O[15:0]
		for MT29F2G16

CS1#	Chip select: LOW true chip select outputs select a specific memory or I/O device. These chip selects are generated through an internal decode in processor address bits A[31:24].	CE#
OE#	Output enable: LOW true output indicates that the current bus cycle is a READ cycle. Used by the external memory or I/O device to place READ data on the data bus.	RE#
EB[3]#[1]	Byte enable: LOW true byte enable outputs indicate which data bytes are active for the current bus cycle.	WE#
PC25, PC31[2]	General purpose I/O: General use input or output pins that can be programmed for use as various memory or I/O interface signals.	WP#, R/B#

Notes: 1. One of the byte enables is used in place of the LOW true data WE# signal from the processor. This is due to a limitation in software control of the WE# output. The MX1 allows leading- and trailing-edge timing adjustments of the byte enables, but not of WE#.
2. The GPIO signals are not dedicated signals from the MC9328MX1 processor. These signals can be provided by available multifunction pins on the processor or by user-provided glue logic.

MX1 Register Setup

For the MC9328MX1 system developed at Micron, the NAND Flash interface was placed on CS1#, with the option of supporting 8-bit or 16-bit NAND Flash devices.

Table 2 on page 7 lists the CS1# upper control register bit definitions, together with the MX1 bit names and the bit settings used in the Micron application. For an 8-bit or a 16-bit NAND Flash interface, CS1#_U = 0x00000501 in this example. Figure 4 on page 7 provides a graphic representation of the upper control register configuration.

Table 3 on page 8 lists the CS1# lower control register bit definitions, together with the MX1 bit names and the bit settings used in the Micron application. For an 8-bit NAND Flash interface, CS1#_L = 0x31240B01 in this example. For a 16-bit NAND Flash interface, CS1#_L = 0x31240D01. Figure 5 on page 8 shows the lower control register configuration.

Table 2: MC9328MX1 CS1# Upper Control Register Configuration

MX1 Bit Name	Bit Definition	Micron NAND Flash Bit Setting Used
DTACK_SEL Bit 63	DTACK select: Set to 0 for generic DTACK function. This is actually a "Don't Care," as DTACK is not used in this application.	0
Reserved Bit 62	Reserved: Set to "0."	0
BCD Bits 61:60	Burst clock divisor: Set to "00" for a burst clock divisor of 1. These bits are "Don't Care," as the burst clock is not used for this application.	00
BCS Bits 59:56	Burst clock start: Set to "0000" for a burst clock start one-half cycle before BCLK. This is a "Don't Care," as burst clock is not used in this application.	0000
PSZ Bits 55:54	Page size: Set to "00" for a 4-word page size. This is a "Don't Care," as page mode is not used in this application.	00
PME Bit 53	Page mode emulation: Set to "0" to disable page mode emulation.	0
SYNC Bit 52	Synchronous burst mode enable: Set to "0" to disable synchronous burst mode.	0
DOL Bits 51:48	Data output length: Set to "0000" for 1 system clock delay. This is a "Don't Care," as burst mode is not used in this application.	0000
CNC Bits 47:46	Chip select negation clock cycles: Set to "00" to specify a minimum of 0 clock cycles for which the chip select must remain negated after it has been negated.	00
WSC Bits 45:40	Wait state controls: Set to "000101" for 5 wait states (6 clock transfers) for access to the NAND Flash device connected to the chip select.	000101
Reserved Bit 39	Reserved: Set to "0."	0
WWS Bits 38:36	Write wait state: Set to "000." No additional wait states are required for WRITE cycles.	000
EDC Bits 35:32	Extra dead cycle: Set to "0001" for 1 idle cycle, inserted after a READ cycle, for back-to-back external transfers.	0001

Notes: 1. In this example, 0x00000501 was used for MX1 register CS1#U with both 8-bit and 16-bit NAND Flash devices.

Figure 4: Upper Control Register Configuration

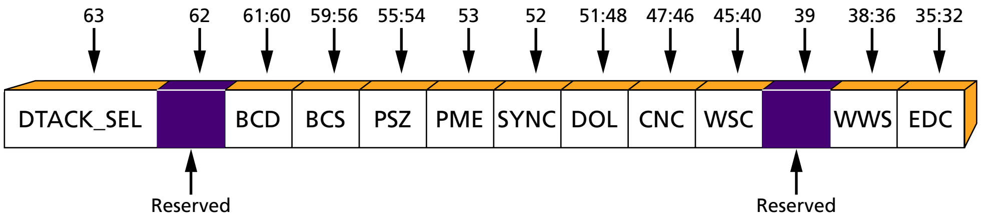
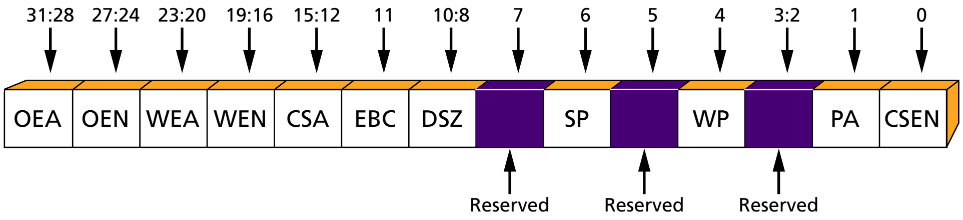


Table 3: MC9328MX1 Chip Select 1 Lower Control Register Configuration

MX1 Bit Name	Description Bit Definition	Micron NAND Flash Bit Setting Used
OEA Bits 31:28	OE# assert: Set to "0011" to allow 3 half clocks before OE# asserts during a READ cycle.	0011
OEN Bits 27:24	OE# negate: Set to "0001" to allow 1 half clock before end-of-access for OE# to negate.	0001
WEA Bits 23:20	EB [3:0] assert: Set to "0010" to allow 2 half clocks before WE# asserts during WRITE cycles.	0010
WEN Bits 19:16	EB [3:0] negate during WRITE: Set to "0100" to allow 4 half clocks before end-of-access for WE# to negate.	0100
CSA Bits 15:12	Chip select assert: Set to "0000" to allow 0 half clocks before chip select assertion and negation during external WRITE cycles.	0000
EBC Bit 11	Enable byte control: Set to "1" so that EB[3:0] only assert during WRITE cycles.	1
DSZ Bits 10:8	Data port size: Set to "011" for an 8-bit data port width located on D[7:0]. For a 16-bit NAND Flash device residing on D[15:0], this would be set to "101."	011
Reserved Bit 7	Reserved: Set to "0."	0
SP Bit 6	Supervisor protect: Set to "0" to support user-mode access in this chip select range.	0
Reserved Bit 5	Reserved: Set to "0."	0
WP Bit 4	Write protect: Set to "0" to support WRITEs occurring in this chip select range.	0
Reserved Bits 3:2	Reserved: Set to "0."	0
PA Bit 1	Pin assert: Set to "0." This bit is a "Don't Care," as the chip select is not operating as a programmable output pin.	0
CSEN Bit 0	Chip select enable: Set to "1" to enable the chip select.	1

Notes: 1. In this example, 0x31240B01 was used for MX1 register CS1#L with 8-bit NAND Flash devices, and 0x31240D01 was used with 16-bit NAND Flash devices.

Figure 5: Lower Control Register Configuration



Software Communication

After hardware communication paths have been established, software commands are necessary to execute the various NAND Flash device commands. These commands are grouped as follows:

- DragonBall MX1 I/O initialization operations
- READ operations
- PROGRAM operations
- INTERNAL DATA MOVE operations
- ERASE operations
- RESET operations

Command-use instructions are discussed in "Software Sample Files" on page 15. Initialization code details for use in an MX1/Micron NAND Flash interface are provided in two files on the Micron Web site. The files are:

- nandIO.h
- nandIO.c

To view or download these or other software files described in this document, the user must agree to licensing terms as specified by Micron. Licensing and software access are available at www.micron.com/products/nand/software/index.

Note: Micron sample software is configured for use with x8 NAND Flash devices.

DragonBall MX1 I/O Initialization Functions

NAND_Init

The NAND_Init function is used to initialize the MX1 I/Os and CS1# for operation with a NAND Flash device. The function writes Micron configuration settings to the I/O configuration registers located on the MX1.

For additional information, download the complete DragonBall MX1 applications processor data sheet and reference manual (see "Reference Materials" on page 15).

Table 4: DragonBall MX1 I/O Initialization Function

Function Commands	Parameters	Returns
NAND_Init	void	None

READ Command Functions

All READ command function parameters are provided in Table 5 on page 11.

PAGE READ Command Sequence

NAND_ReadPage: Reads a page of data from the NAND Flash array. This function issues the 00h command to begin the PAGE READ operation, inputs the NAND Flash address where the PAGE READ is to begin, issues the 30h command to confirm the PAGE READ, waits for device ready, then reads up to 2,112 bytes of data from the data register.

RANDOM DATA READ Command Sequence

NAND_ReadPageRandomStart: Begins the RANDOM DATA READ operation by issuing the 00h command, 5 address cycles, and a 30h command.

NAND_ReadPageRandom: Sends commands 05h and E0h, along with the column address, to the NAND Flash device. Only the column address is necessary, as the page number does not change in a RANDOM READ operation. Data bytes are then read out for the specified number of bytes.

PAGE READ CACHE MODE Command Sequence

NAND_ReadPageCacheStart: Begins a PAGE READ CACHE MODE operation. This function issues the 00h command, followed by 5 address cycles, then issues the 30h command.

NAND_ReadPageCache: Continues the command sequence by sending the 31h command to the device. After the device returns to the ready state, data can be clocked out.

NAND_ReadPageCacheLast: Completes this command sequence by issuing the 3Fh command. When the device returns to the ready state the second time, the specified data bytes are read out.

READ ID Command Sequence

NAND_ReadID: Issues the 90h command to read out the 4 bytes of device-specific ID and configuration code (manufacturer ID and device ID of the NAND Flash device) programmed at the factory.

READ STATUS Command Sequence

NAND_ReadStatus: Issues the 70h command to read the status register and determine the current status of the NAND Flash device.

Table 5: READ Command Functions

Functions	Parameters	Returns
NAND_ReadPage	a_uiPageNum, a_usColNum, a_usReadSizeByte, a_pucReadBuf	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ReadPageRandomStart	a_uiPageNum, a_usColNum, a_usReadSizeByte, a_pucReadBuf	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ReadPageRandom	a_usColNum, a_usReadSizeBytes, a_pucReadBuf	None
NAND_ReadPageCacheStart	a_uiPageNum, a_usColNum	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ReadPageCache	a_usReadSizeByte, a_pucReadBuf	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ReadPageCacheLast	a_usReadSizeByte, a_pucReadBuf	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ReadID	a_ReadID	None
NAND_ReadStatus	void	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT

PROGRAM Functions

All PROGRAM command function parameters are provided in Table 6.

PROGRAM PAGE Command Sequence

NAND_ProgramPage: Issues the 80h command to program a page of data to the NAND Flash device. This is followed by 5 address cycles and the data to be programmed, then the 10h command to confirm the PROGRAM PAGE operation.

Random PROGRAM PAGE Command Sequence

NAND_ProgramPageRandomStart: Issues the 80h command to begin random programming, followed by 5 address cycles, then the data to be programmed. Data is stored in the read buffer.

NAND_ProgramPageRandom: Issues the 85h command to initiate a RANDOM PAGE PROGRAM operation, followed by 2 address cycles and the data to be stored in the read buffer. Only the column address is required, as the page number does not change in RANDOM DATA INPUT operations.

NAND_ProgramPageRandomLast: Completes the random programming sequence by issuing the 10h command.

PROGRAM PAGE CACHE MODE Command Sequence

NAND_ProgramPageCache: Starts the sequence by issuing the 80h command, followed by 5 address cycles, the data to be programmed, and the 15h command.

NAND_ProgramPageCacheLast: Completes the sequence by issuing the 80h command, followed by 5 address cycles, the data to be programmed, and the 10h command.

Table 6: PROGRAM Command Functions

Function Commands	Parameters	Returns
NAND_ProgramPage	a_uiPageNum, a_usColNum, a_usReadSizeByte, a_pucReadBuf	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ProgramPageRandomStart	a_uiPageNum, a_usColNum, a_usReadSizeByte, a_pucReadBuf	None
NAND_ProgramPageRandom	a_usColNum, a_usReadSizeByte, a_pucReadBuf	None
NAND_ProgramPageRandomLast	void	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ProgramPageCache	a_uiPageNum, a_usColNum, a_usReadSizeByte, a_pucReadBuf	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ProgramPageCacheLast	a_uiPageNum, a_usColNum, a_usReadSizeByte, a_pucReadBuf	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT

READ for INTERNAL DATA MOVE Functions

All READ for INTERNAL DATA MOVE command function parameters are provided in Table 7.

READ for INTERNAL DATA MOVE Command Sequence

NAND_ReadInternalDataMove: Starts by issuing a 00h command, followed by 5 address cycles and the 35h command. This reads the page specified by the address and stores the data in the NAND Flash on-chip cache register when the READ is complete.

NAND_ProgramInternalDataMove: Completes an INTERNAL DATA MOVE by issuing an 85h command, followed by 5 address cycles to the location where the data in the on-chip cache register will be moved, then a 35h command.

READ for INTERNAL DATA MOVE with RANDOM DATA INPUT Command Sequence

NAND_ProgramInternalDataMoveRandomStart: Issues the 00h command, followed by 5 address cycles and the 30h command. This reads the data from one page of memory.

NAND_ProgramInternalDataMoveRandom: Issues an 85h command to initiate a RANDOM PAGE operation, followed by 2 address cycles that correspond to a new column address within the selected page. This then makes changes to the data read.

NAND_ProgramInternalDataMoveRandomLast: Completes the sequence by issuing a 10h command. This programs the modified data to another page in the NAND Flash device within the selected block.

Table 7: INTERNAL DATA MOVE Command Functions

Function Commands	Parameters	Returns
NAND_ReadInternalDataMove	a_uiPageNum, a_usColNum	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ProgramInternalDataMove	a_uiPageNum, a_usColNum	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT
NAND_ProgramInternalDataMoveRandomStart	a_uiPageNum, a_usColNum, a_usReadSizeByte, a_pucReadBuf	None
NAND_ProgramInternalDataMoveRandom	a_usColNum, a_usReadSizeByte, a_pucReadBuf	None
NAND_ProgramInternalDataMoveRandomLast	void	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT

ERASE Functions

All ERASE command function parameters are provided in Table 8.

BLOCK ERASE Command Sequence

NAND_EraseBlock: Issues the 60h command and erases a block of data in the NAND Flash device, returning all bytes in the block to FFh.

Table 8: BLOCK ERASE Command Functions

Function Commands	Parameters	Returns
NAND_EraseBlock	a_uiPageNum	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT

RESET Command Functions

All RESET command function parameters are provided in Table 9.

RESET Command Sequence

NAND_Reset: Issues the FFh command to reset the NAND Flash device.

Table 9: RESET Command Functions

Function Commands	Parameters	Returns
NAND_Reset	void	NAND_IO_RC_PASS, NAND_IO_RC_FAIL, or NAND_IO_RC_TIMEOUT

Software Sample Files

A sample software implementation file provides functional descriptions and detailed discussion of the commands noted in this document, as well as additional information for implementing communications between a Micron NAND Flash device and the MX1 applications processor. This information is augmented by specific examples using x8 NAND Flash devices.

Sample implementation software is in this file:

- nandIOTest.cpp

To view or download this or other software files referenced in this document, the user must agree to licensing terms as specified by Micron. Licensing and software access are available at:

www.micron.com/products/nand/software/index

Reference Materials

MX1 Applications Processor

See the Freescale Web site for detailed information on the DragonBall MX1 applications processor, at:

www.freescale.com

Micron NAND Flash Devices

Micron NAND Flash devices are described in detail at:

www.micron.com/products/nand/

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900
prodmktg@micron.com www.micron.com Customer Comment Line: 800-932-4992

Revision History

Rev. B 05/07

- Added one Web link and updated others.
- Figure 2 on page 4: Fixed part number typographical error.
- "MX1 Register Setup" on page 6: Clarified description.

Rev. A 06/05

- Initial release.