



# NAND Flash Memory

**MT29F4G08ABADAH4, MT29F4G08ABADAWP, MT29F4G08ABBDAH4,  
MT29F4G08ABBDAHC, MT29F4G16ABADAH4, MT29F4G16ABADAWP,  
MT29F4G16ABBDAH4, MT29F4G16ABBDAHC, MT29F8G08ADADAH4,  
MT29F8G08ADBDAH4, MT29F8G16ADADAH4, MT29F8G16ADBDAH4**

## Features

- Open NAND Flash Interface (ONFI) 1.0-compliant<sup>1</sup>
- Single-level cell (SLC) technology
- Organization
  - Page size x8: 2112 bytes (2048 + 64 bytes)
  - Page size x16: 1056 words (1024 + 32 words)
  - Block size: 64 pages (128K + 4K bytes)
  - Plane size: 2 planes x 2048 blocks per plane
  - Device size: 4Gb: 4096 blocks; 8Gb: 8192 blocks
- Asynchronous I/O performance
  - <sup>t</sup>RC/<sup>t</sup>WC: 20ns (3.3V), 25ns (1.8V)
- Array performance
  - Read page: 25μs<sup>3</sup>
  - Program page: 200μs (TYP: 1.8V, 3.3V)<sup>3</sup>
  - Erase block: 700μs (TYP)
- Command set: ONFI NAND Flash Protocol
- Advanced command set
  - Program page cache mode<sup>4</sup>
  - Read page cache mode<sup>4</sup>
  - One-time programmable (OTP) mode
  - Two-plane commands<sup>4</sup>
  - Interleaved die (LUN) operations
  - Read unique ID
  - Block lock (1.8V only)
  - Internal data move
- Operation status byte provides software method for detecting
  - Operation completion
  - Pass/fail condition
  - Write-protect status
- Ready/Busy# (R/B#) signal provides a hardware method of detecting operation completion
- WP# signal: Write protect entire device
- First block (block address 00h) is valid when shipped from factory with ECC. For minimum required ECC, see Error Management.
- Block 0 requires 1-bit ECC if PROGRAM/ERASE cycles are less than 1000
- RESET (FFh) required as first command after power-on
- Alternate method of device initialization (Nand\_Init) after power up (contact factory)
- Internal data move operations supported within the plane from which data is read
- Quality and reliability
  - Data retention: 10 years
  - Endurance: 100,000 PROGRAM/ERASE cycles
- Operating voltage range
  - V<sub>CC</sub>: 2.7–3.6V
  - V<sub>CC</sub>: 1.7–1.95V
- Operating temperature:
  - Commercial: 0°C to +70°C
  - Industrial (IT): –40°C to +85°C
- Package
  - 48-pin TSOP type 1, CPL<sup>2</sup>
  - 63-ball VFBGA

- Notes:
1. The ONFI 1.0 specification is available at [www.onfi.org](http://www.onfi.org).
  2. CPL = Center parting line.
  3. See Program and Erase Characteristics for <sup>t</sup>R\_ECC and <sup>t</sup>PROG\_ECC specifications.
  4. These commands supported only with ECC disabled.

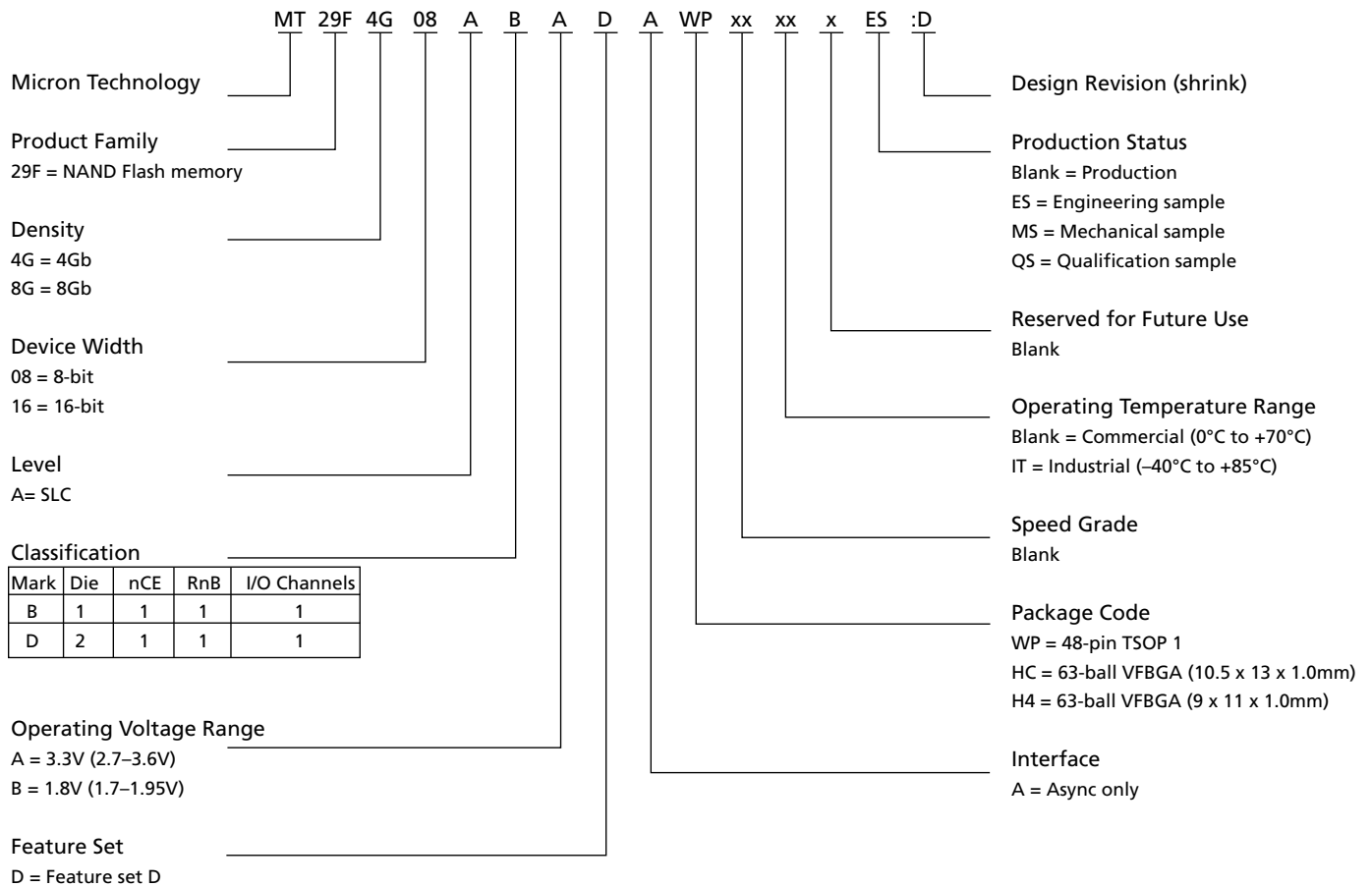


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Features

### Part Numbering Information

Micron NAND Flash devices are available in different configurations and densities. Verify valid part numbers by using Micron's part catalog search at [www.micron.com](http://www.micron.com). To compare features and specifications by device type, visit [www.micron.com/products](http://www.micron.com/products). Contact the factory for devices not found.

**Figure 1: Marketing Part Number Chart**





## Contents

|   |    |
|---|----|
| General Description .....                           | 8  |
| Signal Descriptions .....                           | 8  |
| Signal Assignments .....                            | 9  |
| Package Dimensions .....                            | 12 |
| Architecture .....                                  | 15 |
| Device and Array Organization .....                 | 16 |
| Asynchronous Interface Bus Operation .....          | 20 |
| Asynchronous Enable/Standby .....                   | 20 |
| Asynchronous Commands .....                         | 20 |
| Asynchronous Addresses .....                        | 22 |
| Asynchronous Data Input .....                       | 23 |
| Asynchronous Data Output .....                      | 24 |
| Write Protect# .....                                | 25 |
| Ready/Busy# .....                                   | 25 |
| Device Initialization .....                         | 30 |
| Command Definitions .....                           | 31 |
| Reset Operations .....                              | 34 |
| RESET (FFh) .....                                   | 34 |
| Identification Operations .....                     | 35 |
| READ ID (90h) .....                                 | 35 |
| READ ID Parameter Tables .....                      | 36 |
| READ PARAMETER PAGE (ECh) .....                     | 38 |
| Parameter Page Data Structure Tables .....          | 39 |
| Bare Die Parameter Page Data Structure Tables ..... | 44 |
| READ UNIQUE ID (EDh) .....                          | 47 |
| Feature Operations .....                            | 48 |
| SET FEATURES (EFh) .....                            | 49 |
| GET FEATURES (EEh) .....                            | 50 |
| Status Operations .....                             | 53 |
| READ STATUS (70h) .....                             | 54 |
| READ STATUS ENHANCED (78h) .....                    | 54 |
| Column Address Operations .....                     | 56 |
| RANDOM DATA READ (05h-E0h) .....                    | 56 |
| RANDOM DATA READ TWO-PLANE (06h-E0h) .....          | 57 |
| RANDOM DATA INPUT (85h) .....                       | 58 |
| PROGRAM FOR INTERNAL DATA INPUT (85h) .....         | 59 |
| Read Operations .....                               | 61 |
| READ MODE (00h) .....                               | 63 |
| READ PAGE (00h-30h) .....                           | 63 |
| READ PAGE CACHE SEQUENTIAL (31h) .....              | 64 |
| READ PAGE CACHE RANDOM (00h-31h) .....              | 65 |
| READ PAGE CACHE LAST (3Fh) .....                    | 67 |
| READ PAGE TWO-PLANE 00h-00h-30h .....               | 68 |
| Program Operations .....                            | 70 |
| PROGRAM PAGE (80h-10h) .....                        | 71 |
| PROGRAM PAGE CACHE (80h-15h) .....                  | 71 |
| PROGRAM PAGE TWO-PLANE (80h-11h) .....              | 74 |
| Erase Operations .....                              | 76 |
| ERASE BLOCK (60h-D0h) .....                         | 76 |
| ERASE BLOCK TWO-PLANE (60h-D1h) .....               | 77 |



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Features

|   |     |
|---|-----|
| Internal Data Move Operations .....   | 78  |
| READ FOR INTERNAL DATA MOVE (00h-35h) .....                                   | 79  |
| PROGRAM FOR INTERNAL DATA MOVE (85h-10h) .....                                | 80  |
| PROGRAM FOR INTERNAL DATA MOVE TWO-PLANE (85h-11h) .....                      | 81  |
| Block Lock Feature .....  | 82  |
| WP# and Block Lock .....  | 82  |
| UNLOCK (23h-24h) .....  | 82  |
| LOCK (2Ah) .....  | 85  |
| LOCK TIGHT (2Ch) .....  | 86  |
| BLOCK LOCK READ STATUS (7Ah) .....  | 87  |
| One-Time Programmable (OTP) Operations .....                                  | 89  |
| Legacy OTP Commands .....   | 89  |
| OTP DATA PROGRAM (80h-10h) .....  | 90  |
| RANDOM DATA INPUT (85h) .....   | 91  |
| OTP DATA PROTECT (80h-10) .....   | 92  |
| OTP DATA READ (00h-30h) .....   | 94  |
| Two-Plane Operations .....  | 96  |
| Two-Plane Addressing .....  | 96  |
| Interleaved Die (Multi-LUN) Operations .....                                  | 105 |
| Error Management .....  | 106 |
| Internal ECC and Spare Area Mapping for ECC .....                             | 108 |
| Electrical Specifications .....   | 110 |
| Electrical Specifications – DC Characteristics and Operating Conditions ..... | 112 |
| Electrical Specifications – AC Characteristics and Operating Conditions ..... | 114 |
| Electrical Specifications – Program/Erase Characteristics .....               | 117 |
| Asynchronous Interface Timing Diagrams .....                                  | 118 |
| Revision History .....  | 130 |
| Rev. G, Production – 10/10 .....  | 130 |
| Rev. F, Production – 6/10 .....   | 130 |
| Rev. E, Production – 5/10 .....   | 130 |
| Rev. D, Advance - 3/10 .....  | 130 |
| Rev. C, Advance – 1/10 .....  | 130 |
| Rev. B, Advance – 10/09 .....   | 130 |
| Rev. A, Advance – 7/09 .....  | 131 |



## List of Tables

|   |     |
|---|-----|
| Table 1: Signal Definitions .....   | 8   |
| Table 2: Array Addressing – MT29F4G08 (x8) .....                            | 16  |
| Table 3: Array Addressing – MT29F4G16 (x16) .....                           | 17  |
| Table 4: Array Addressing – MT29F8G08 (x8) .....                            | 18  |
| Table 5: Array Addressing – MT29F8G16 (x16) .....                           | 19  |
| Table 6: Asynchronous Interface Mode Selection .....                        | 20  |
| Table 7: Command Set .....  | 31  |
| Table 8: Two-Plane Command Set .....  | 32  |
| Table 9: READ ID Parameters for Address 00h .....                           | 36  |
| Table 10: READ ID Parameters for Address 20h .....                          | 37  |
| Table 11: Parameter Page Data Structure .....                               | 39  |
| Table 12: Parameter Page Data Structure .....                               | 44  |
| Table 13: Feature Address Definitions .....                                 | 48  |
| Table 14: Feature Address 90h – Array Operation Mode .....                  | 49  |
| Table 15: Feature Addresses 01h: Timing Mode .....                          | 51  |
| Table 16: Feature Addresses 80h: Programmable I/O Drive Strength .....      | 52  |
| Table 17: Feature Addresses 81h: Programmable R/B# Pull-Down Strength ..... | 52  |
| Table 18: Status Register Definition .....                                  | 53  |
| Table 19: Block Lock Address Cycle Assignments .....                        | 84  |
| Table 20: Block Lock Status Register Bit Definitions .....                  | 87  |
| Table 21: Error Management Details .....                                    | 106 |
| Table 22: Absolute Maximum Ratings .....                                    | 110 |
| Table 23: Recommended Operating Conditions .....                            | 110 |
| Table 24: Valid Blocks .....  | 110 |
| Table 25: Capacitance .....   | 111 |
| Table 26: Test Conditions .....   | 111 |
| Table 27: DC Characteristics and Operating Conditions (3.3V) .....          | 112 |
| Table 28: DC Characteristics and Operating Conditions (1.8V) .....          | 113 |
| Table 29: AC Characteristics: Command, Data, and Address Input (3.3V) ..... | 114 |
| Table 30: AC Characteristics: Command, Data, and Address Input (1.8V) ..... | 114 |
| Table 31: AC Characteristics: Normal Operation (3.3V) .....                 | 115 |
| Table 32: AC Characteristics: Normal Operation (1.8V) .....                 | 115 |
| Table 33: Program/Erase Characteristics .....                               | 117 |



## List of Figures

|   |    |
|---|----|
| Figure 1: Marketing Part Number Chart .....   | 2  |
| Figure 2: 48-Pin TSOP – Type 1, CPL (Top View) .....                                  | 9  |
| Figure 3: 63-Ball VFBGA, x8 (Balls Down, Top View) .....                              | 10 |
| Figure 4: 63-Ball VFBGA, x16 (Balls Down, Top View) .....                             | 11 |
| Figure 5: 48-Pin TSOP – Type 1, CPL .....   | 12 |
| Figure 6: 63-Ball VFBGA (10.5mm x 13mm) .....   | 13 |
| Figure 7: 63-Ball VFBGA (9mm x 11mm) .....  | 14 |
| Figure 8: NAND Flash Die (LUN) Functional Block Diagram .....                         | 15 |
| Figure 9: Array Organization – MT29F4G08 (x8) .....                                   | 16 |
| Figure 10: Array Organization – MT29F4G16 (x16) .....                                 | 17 |
| Figure 11: Array Organization – MT29F8G08 (x8) .....                                  | 18 |
| Figure 12: Array Organization – MT29F8G16 (x16) .....                                 | 19 |
| Figure 13: Asynchronous Command Latch Cycle .....                                     | 21 |
| Figure 14: Asynchronous Address Latch Cycle .....                                     | 22 |
| Figure 15: Asynchronous Data Input Cycles .....                                       | 23 |
| Figure 16: Asynchronous Data Output Cycles .....                                      | 24 |
| Figure 17: Asynchronous Data Output Cycles (EDO Mode) .....                           | 25 |
| Figure 18: READ/BUSY# Open Drain .....  | 26 |
| Figure 19: $t_{\text{Fall}}$ and $t_{\text{Rise}}$ (3.3V $V_{\text{CC}}$ ) .....      | 27 |
| Figure 20: $t_{\text{Fall}}$ and $t_{\text{Rise}}$ (1.8V $V_{\text{CC}}$ ) .....      | 27 |
| Figure 21: $I_{\text{OL}}$ vs. $R_p$ ( $V_{\text{CC}} = 3.3V$ $V_{\text{CC}}$ ) ..... | 28 |
| Figure 22: $I_{\text{OL}}$ vs. $R_p$ (1.8V $V_{\text{CC}}$ ) .....                    | 28 |
| Figure 23: $T_C$ vs. $R_p$ .....  | 29 |
| Figure 24: R/B# Power-On Behavior .....   | 30 |
| Figure 25: RESET (FFh) Operation .....  | 34 |
| Figure 26: READ ID (90h) with 00h Address Operation .....                             | 35 |
| Figure 27: READ ID (90h) with 20h Address Operation .....                             | 35 |
| Figure 28: READ PARAMETER (ECh) Operation .....                                       | 38 |
| Figure 29: READ UNIQUE ID (EDh) Operation .....                                       | 47 |
| Figure 30: SET FEATURES (EFh) Operation .....   | 49 |
| Figure 31: GET FEATURES (EEh) Operation .....   | 50 |
| Figure 32: READ STATUS (70h) Operation .....  | 54 |
| Figure 33: READ STATUS ENHANCED (78h) Operation .....                                 | 55 |
| Figure 34: RANDOM DATA READ (05h-E0h) Operation .....                                 | 56 |
| Figure 35: RANDOM DATA READ TWO-PLANE (06h-E0h) Operation .....                       | 57 |
| Figure 36: RANDOM DATA INPUT (85h) Operation .....                                    | 58 |
| Figure 37: PROGRAM FOR INTERNAL DATA INPUT (85h) Operation .....                      | 60 |
| Figure 38: READ PAGE (00h-30h) Operation .....  | 64 |
| Figure 39: READ PAGE (00h-30h) Operation with Internal ECC Enabled .....              | 64 |
| Figure 40: READ PAGE CACHE SEQUENTIAL (31h) Operation .....                           | 65 |
| Figure 41: READ PAGE CACHE RANDOM (00h-31h) Operation .....                           | 66 |
| Figure 42: READ PAGE CACHE LAST (3Fh) Operation .....                                 | 67 |
| Figure 43: READ PAGE TWO-PLANE (00h-00h-30h) Operation .....                          | 69 |
| Figure 44: PROGRAM PAGE (80h-10h) Operation .....                                     | 71 |
| Figure 45: PROGRAM PAGE CACHE (80h-15h) Operation (Start) .....                       | 73 |
| Figure 46: PROGRAM PAGE CACHE (80h-15h) Operation (End) .....                         | 73 |
| Figure 47: PROGRAM PAGE TWO-PLANE (80h-11h) Operation .....                           | 75 |
| Figure 48: ERASE BLOCK (60h-D0h) Operation .....                                      | 76 |
| Figure 49: ERASE BLOCK TWO-PLANE (60h-D1h) Operation .....                            | 77 |
| Figure 50: READ FOR INTERNAL DATA MOVE (00h-35h) Operation .....                      | 79 |



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Features

|  |     |
|--|-----|
| Figure 51: READ FOR INTERNAL DATA MOVE (00h–35h) with RANDOM DATA READ (05h–E0h)                 | 79  |
| Figure 52: INTERNAL DATA MOVE (85h–10h) with Internal ECC Enabled                                | 80  |
| Figure 53: INTERNAL DATA MOVE (85h–10h) with RANDOM DATA INPUT with Internal ECC Enabled         | 80  |
| Figure 54: PROGRAM FOR INTERNAL DATA MOVE (85h–10h) Operation                                    | 80  |
| Figure 55: PROGRAM FOR INTERNAL DATA MOVE (85h–10h) with RANDOM DATA INPUT (85h)                 | 81  |
| Figure 56: PROGRAM FOR INTERNAL DATA MOVE TWO-PLANE (85h–11h) Operation                          | 81  |
| Figure 57: Flash Array Protected: Invert Area Bit = 0  | 83  |
| Figure 58: Flash Array Protected: Invert Area Bit = 1  | 83  |
| Figure 59: UNLOCK Operation  | 84  |
| Figure 60: LOCK Operation  | 85  |
| Figure 61: LOCK TIGHT Operation  | 86  |
| Figure 62: PROGRAM/ERASE Issued to Locked Block  | 87  |
| Figure 63: BLOCK LOCK READ STATUS  | 87  |
| Figure 64: BLOCK LOCK Flowchart  | 88  |
| Figure 65: OTP DATA PROGRAM (After Entering OTP Operation Mode)                                  | 91  |
| Figure 66: OTP DATA PROGRAM Operation with RANDOM DATA INPUT (After Entering OTP Operation Mode) | 92  |
| Figure 67: OTP DATA PROTECT Operation (After Entering OTP Protect Mode)                          | 93  |
| Figure 68: OTP DATA READ   | 94  |
| Figure 69: OTP DATA READ with RANDOM DATA READ Operation   | 95  |
| Figure 70: TWO-PLANE PAGE READ   | 97  |
| Figure 71: TWO-PLANE PAGE READ with RANDOM DATA READ   | 98  |
| Figure 72: TWO-PLANE PROGRAM PAGE  | 98  |
| Figure 73: TWO-PLANE PROGRAM PAGE with RANDOM DATA INPUT   | 99  |
| Figure 74: TWO-PLANE PROGRAM PAGE CACHE MODE   | 100 |
| Figure 75: TWO-PLANE INTERNAL DATA MOVE  | 101 |
| Figure 76: TWO-PLANE INTERNAL DATA MOVE with TWO-PLANE RANDOM DATA READ                          | 102 |
| Figure 77: TWO-PLANE INTERNAL DATA MOVE with RANDOM DATA INPUT                                   | 103 |
| Figure 78: TWO-PLANE BLOCK ERASE   | 104 |
| Figure 79: TWO-PLANE/MULTIPLE-DIE READ STATUS Cycle  | 104 |
| Figure 80: Spare Area Mapping (x8)   | 108 |
| Figure 81: Spare Area Mapping (x16)  | 109 |
| Figure 82: RESET Operation   | 118 |
| Figure 83: READ STATUS Cycle   | 118 |
| Figure 84: READ STATUS ENHANCED Cycle  | 119 |
| Figure 85: READ PARAMETER PAGE   | 119 |
| Figure 86: READ PAGE   | 120 |
| Figure 87: READ PAGE Operation with CE# “Don’t Care”   | 121 |
| Figure 88: RANDOM DATA READ  | 122 |
| Figure 89: READ PAGE CACHE SEQUENTIAL  | 123 |
| Figure 90: READ PAGE CACHE RANDOM  | 124 |
| Figure 91: READ ID Operation   | 125 |
| Figure 92: PROGRAM PAGE Operation  | 125 |
| Figure 93: PROGRAM PAGE Operation with CE# “Don’t Care”  | 126 |
| Figure 94: PROGRAM PAGE Operation with RANDOM DATA INPUT   | 126 |
| Figure 95: PROGRAM PAGE CACHE  | 127 |
| Figure 96: PROGRAM PAGE CACHE Ending on 15h  | 127 |
| Figure 97: INTERNAL DATA MOVE  | 128 |
| Figure 98: INTERNAL DATA MOVE (85h–10h) with Internal ECC Enabled                                | 128 |
| Figure 99: INTERNAL DATA MOVE (85h–10h) with Random Data Input with Internal ECC Enabled         | 129 |
| Figure 100: ERASE BLOCK Operation  | 129 |





## General Description

Micron NAND Flash devices include an asynchronous data interface for high-performance I/O operations. These devices use a highly multiplexed 8-bit bus (I/Ox) to transfer commands, address, and data. There are five control signals used to implement the asynchronous data interface: CE#, CLE, ALE, WE#, and RE#. Additional signals control hardware write protection and monitor device status (R/B#).

This hardware interface creates a low pin-count device with a standard pinout that remains the same from one density to another, enabling future upgrades to higher densities with no board redesign.

A target is the unit of memory accessed by a chip enable signal. A target contains one or more NAND Flash die. A NAND Flash die is the minimum unit that can independently execute commands and report status. A NAND Flash die, in the ONFI specification, is referred to as a logical unit (LUN). There is at least one NAND Flash die per chip enable signal. For further details, see Device and Array Organization.

This device has an internal 4-bit ECC that can be enabled using the GET/SET features. See Internal ECC and Spare Area Mapping for ECC for more information.

## Signal Descriptions

**Table 1: Signal Definitions**

| Signal <sup>1</sup>              | Type   | Description <sup>2</sup>  |
|----------------------------------|--------|---|
| ALE                              | Input  | <b>Address latch enable:</b> Loads an address from I/O[7:0] into the address register.  |
| CE#                              | Input  | <b>Chip enable:</b> Enables or disables one or more die (LUNs) in a target.   |
| CLE                              | Input  | <b>Command latch enable:</b> Loads a command from I/O[7:0] into the command register.   |
| LOCK                             | Input  | When LOCK is HIGH during power-up, the BLOCK LOCK function is enabled. To disable the BLOCK LOCK, connect LOCK to V <sub>SS</sub> during power-up, or leave it disconnected (internal pull-down). |
| RE#                              | Input  | <b>Read enable:</b> Transfers serial data from the NAND Flash to the host system.   |
| WE#                              | Input  | <b>Write enable:</b> Transfers commands, addresses, and serial data from the host system to the NAND Flash.   |
| WP#                              | Input  | <b>Write protect:</b> Enables or disables array PROGRAM and ERASE operations.   |
| I/O[7:0] (x8)<br>I/O[15:0] (x16) | I/O    | <b>Data inputs/outputs:</b> The bidirectional I/Os transfer address, data, and command information.   |
| R/B#                             | Output | <b>Ready/busy:</b> An open-drain, active-low output that requires an external pull-up resistor. This signal indicates target array activity.  |
| V <sub>CC</sub>                  | Supply | <b>V<sub>CC</sub>:</b> Core power supply  |
| V <sub>SS</sub>                  | Supply | <b>V<sub>SS</sub>:</b> Core ground connection   |
| NC                               | –      | <b>No connect:</b> NCs are not internally connected. They can be driven or left unconnected.  |
| DNU                              | –      | <b>Do not use:</b> DNUs must be left unconnected.   |

- Notes:
1. See Device and Array Organization for detailed signal connections.
  2. See Asynchronous Interface Bus Operation for detailed asynchronous interface signal descriptions.

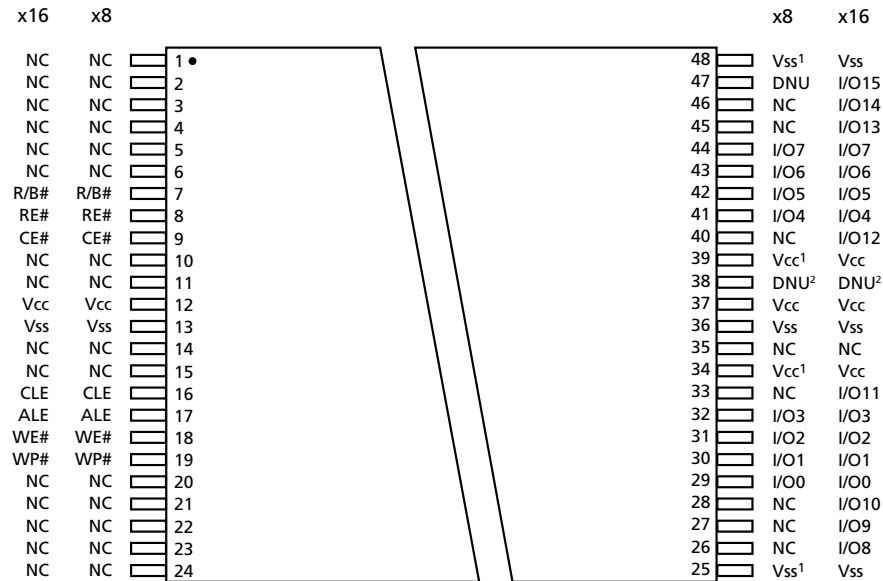




## 4Gb, 8Gb: x8, x16 NAND Flash Memory Signal Assignments

### Signal Assignments

**Figure 2: 48-Pin TSOP – Type 1, CPL (Top View)**

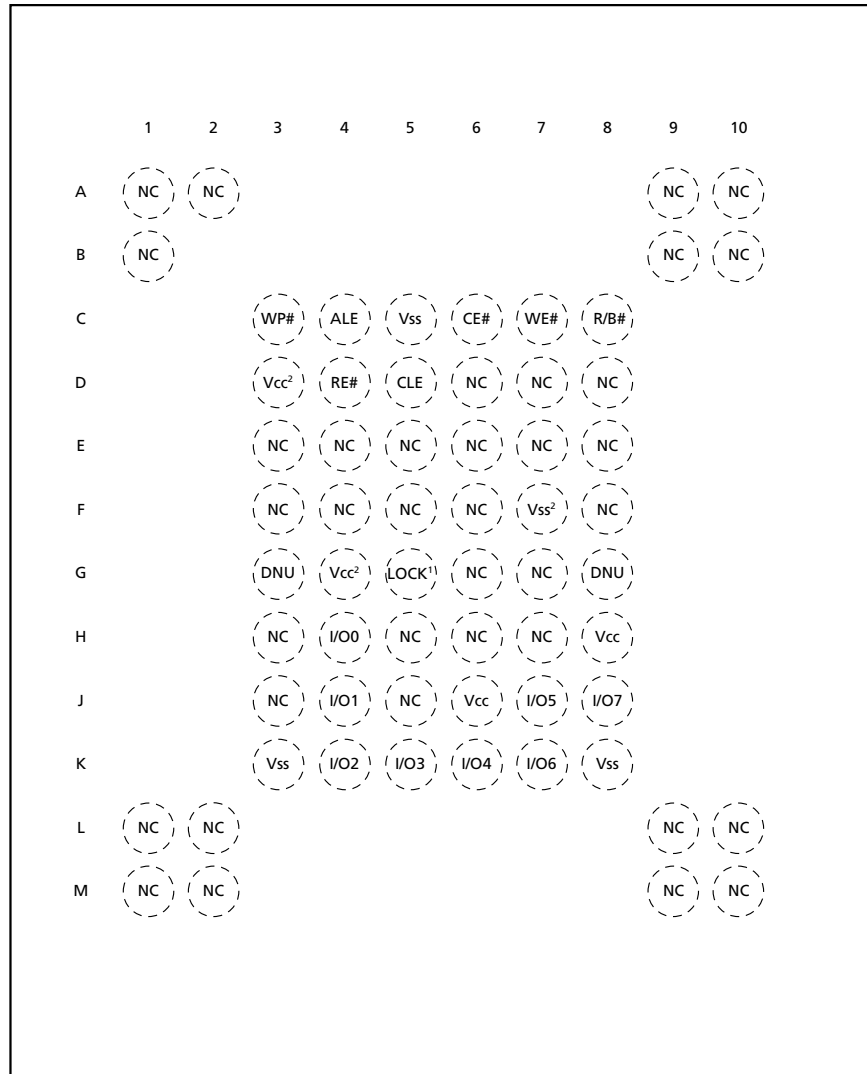


- Notes:
1. These pins might not be bonded in the package; however, Micron recommends that the customer connect these pins to the designated external sources for ONFI compatibility.
  2. For the 3V device, pin 38 is DNU. For the 1.8V device, pin 38 is LOCK.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Signal Assignments

**Figure 3: 63-Ball VFBGA, x8 (Balls Down, Top View)**

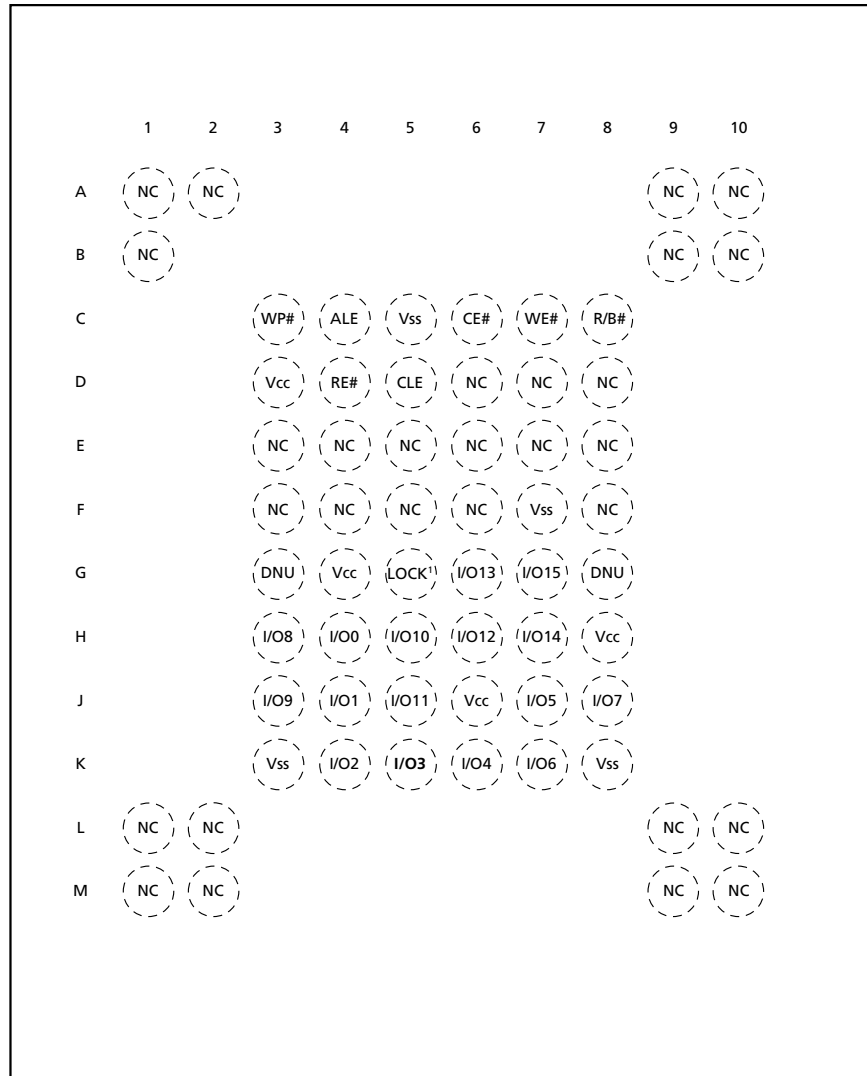


- Notes:
1. For the 3V device, G5 changes to DNU. NO LOCK function is available on the 3.3V device.
  2. These pins might not be bonded in the package; however, Micron recommends that the customer connect these pins to the designated external sources for ONFI compatibility.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Signal Assignments

**Figure 4: 63-Ball VFBGA, x16 (Balls Down, Top View)**



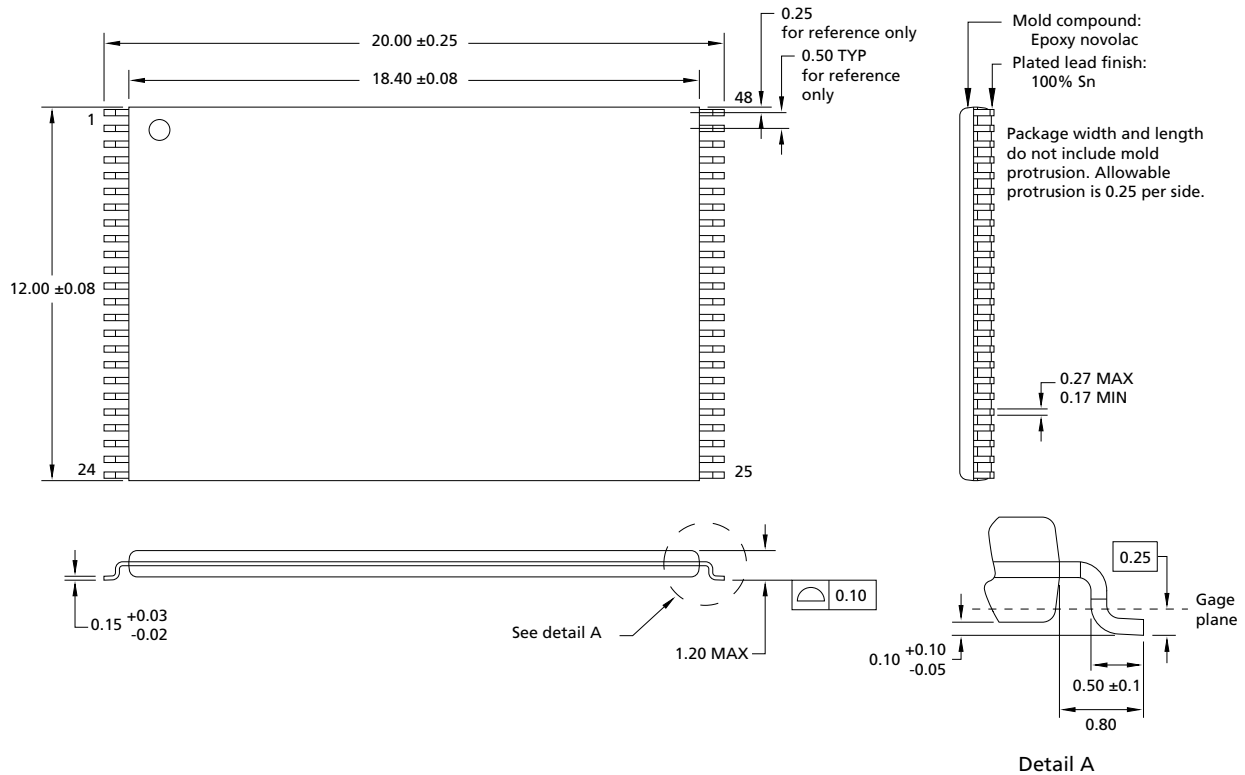
Note: 1. For the 3V device, G5 changes to DNU. NO LOCK function is available on the 3.3V device.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Package Dimensions

### Package Dimensions

**Figure 5: 48-Pin TSOP – Type 1, CPL**

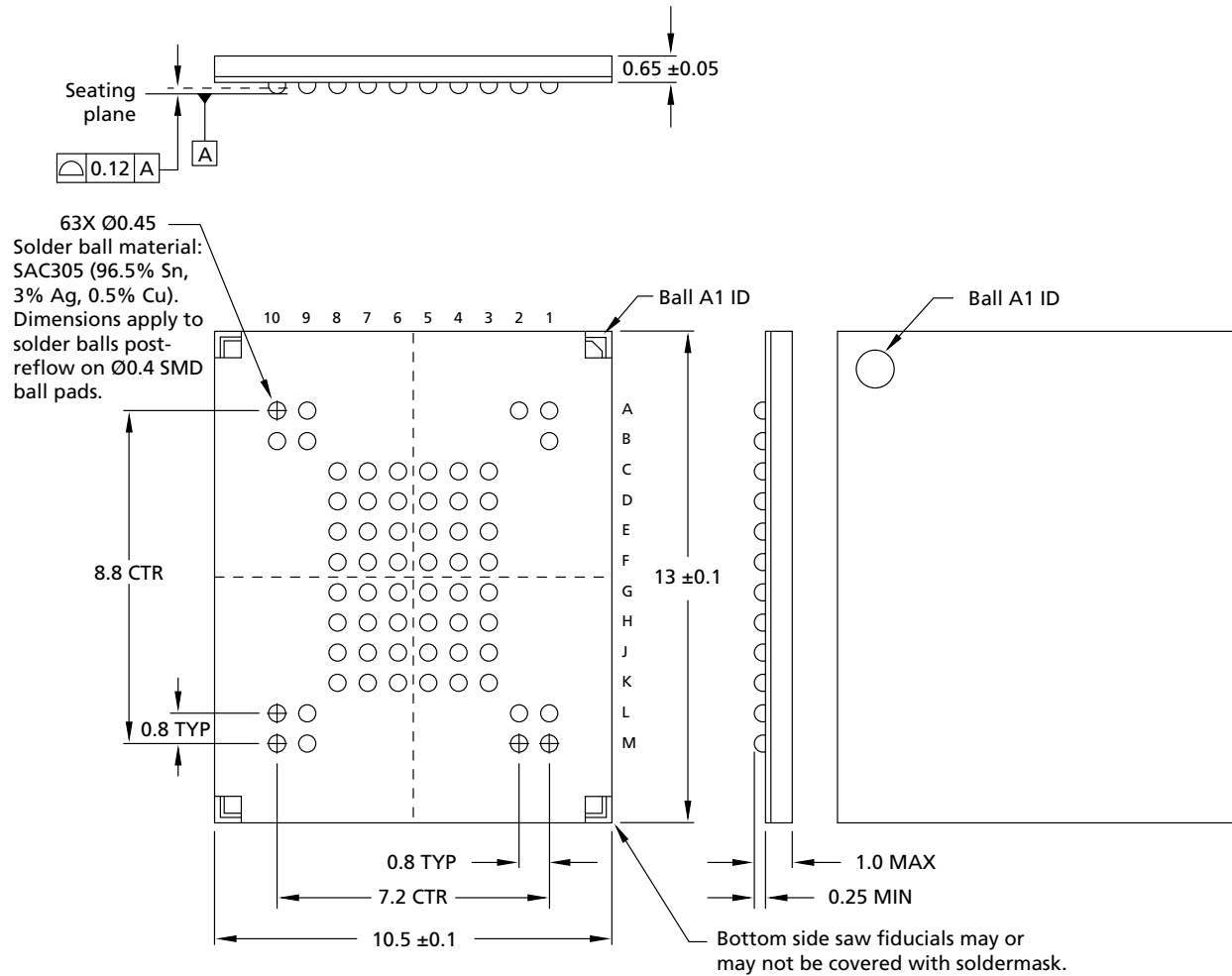


Note: 1. All dimensions are in millimeters.



# 4Gb, 8Gb: x8, x16 NAND Flash Memory Package Dimensions

**Figure 6: 63-Ball VFBGA (10.5mm x 13mm)**

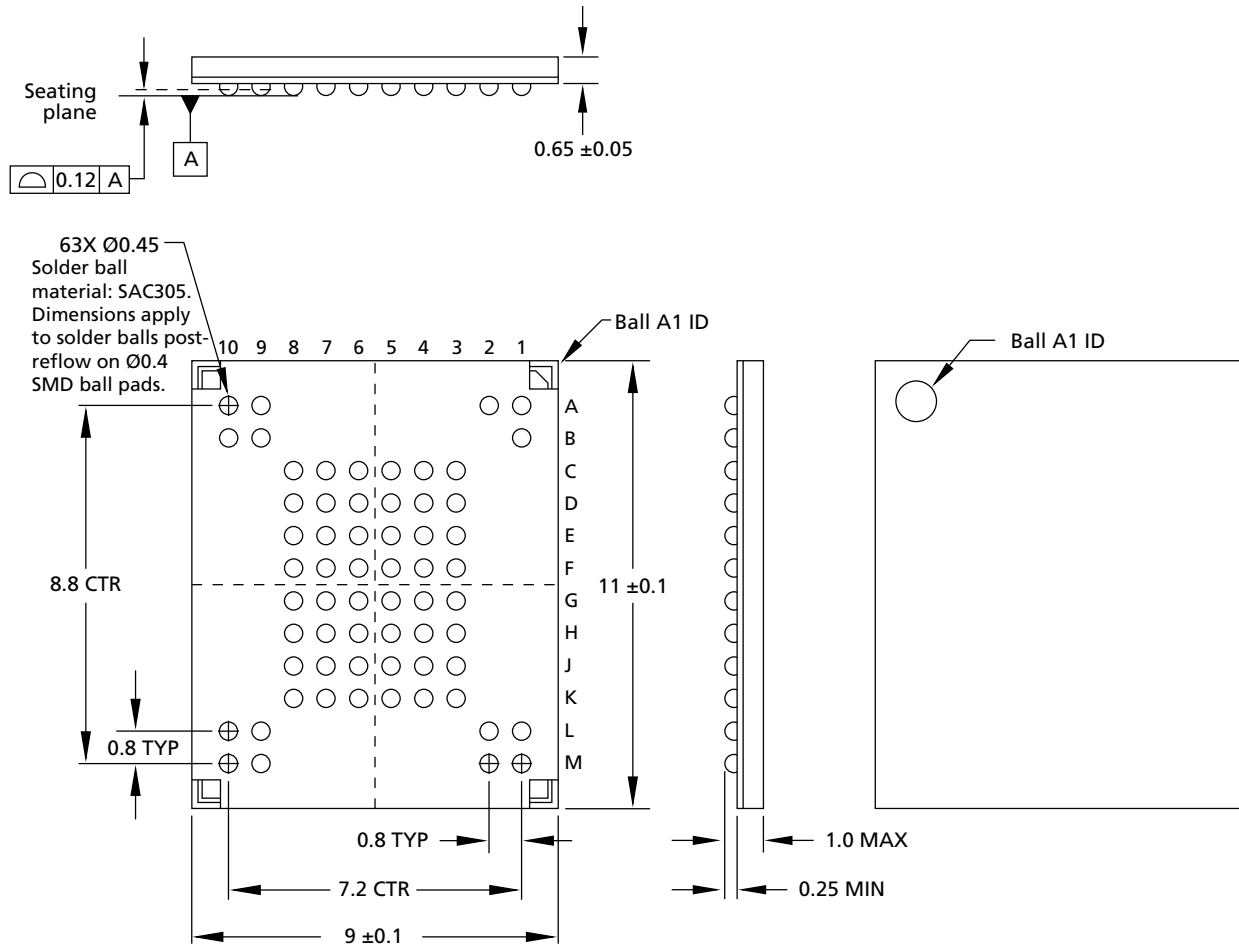


Note: 1. All dimensions are in millimeters.



# 4Gb, 8Gb: x8, x16 NAND Flash Memory Package Dimensions

**Figure 7: 63-Ball VFBGA (9mm x 11mm)**



Note: 1. All dimensions are in millimeters.



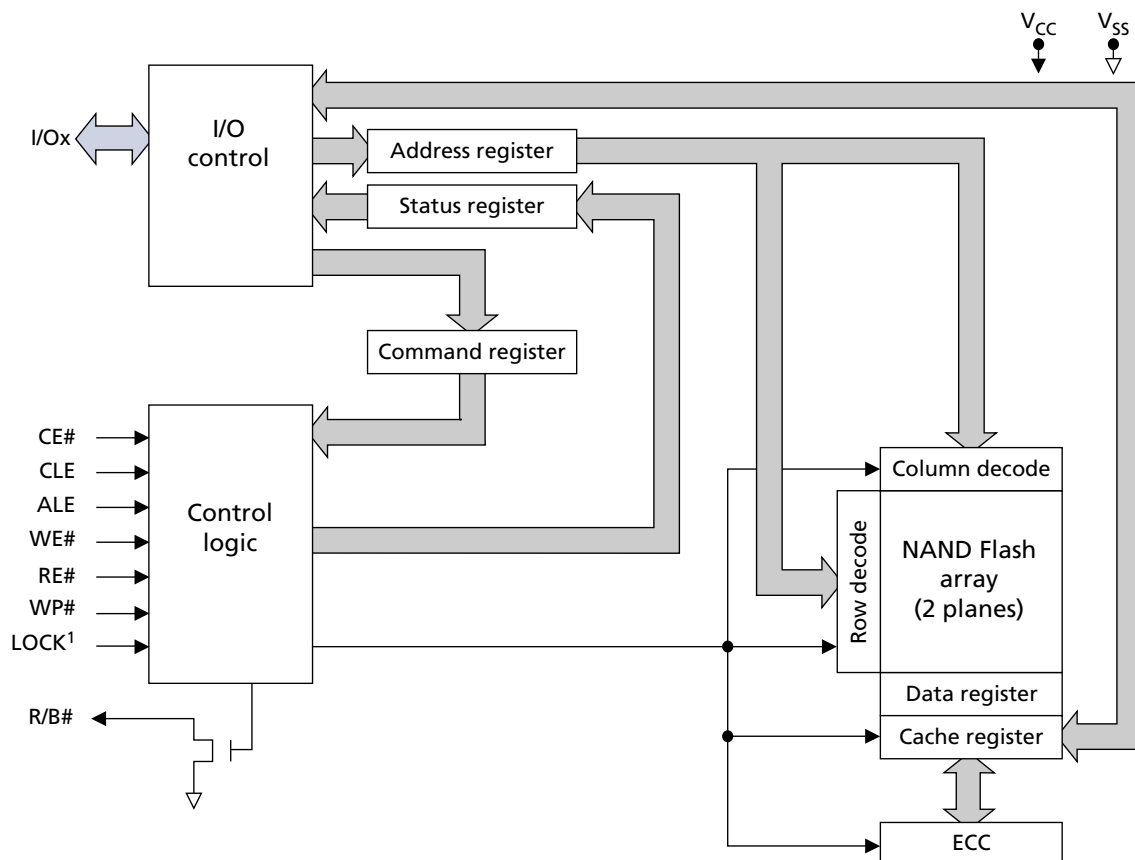
## Architecture

These devices use NAND Flash electrical and command interfaces. Data, commands, and addresses are multiplexed onto the same pins and received by I/O control circuits. The commands received at the I/O control circuits are latched by a command register and are transferred to control logic circuits for generating internal signals to control device operations. The addresses are latched by an address register and sent to a row decoder to select a row address, or to a column decoder to select a column address.

Data is transferred to or from the NAND Flash memory array, byte by byte (x8) or word by word (x16), through a data register and a cache register.

The NAND Flash memory array is programmed and read using page-based operations and is erased using block-based operations. During normal page operations, the data and cache registers act as a single register. During cache operations, the data and cache registers operate independently to increase data throughput. The status register reports the status of die operations.

**Figure 8: NAND Flash Die (LUN) Functional Block Diagram**



Note: 1. The LOCK pin is used on the 1.8V device.

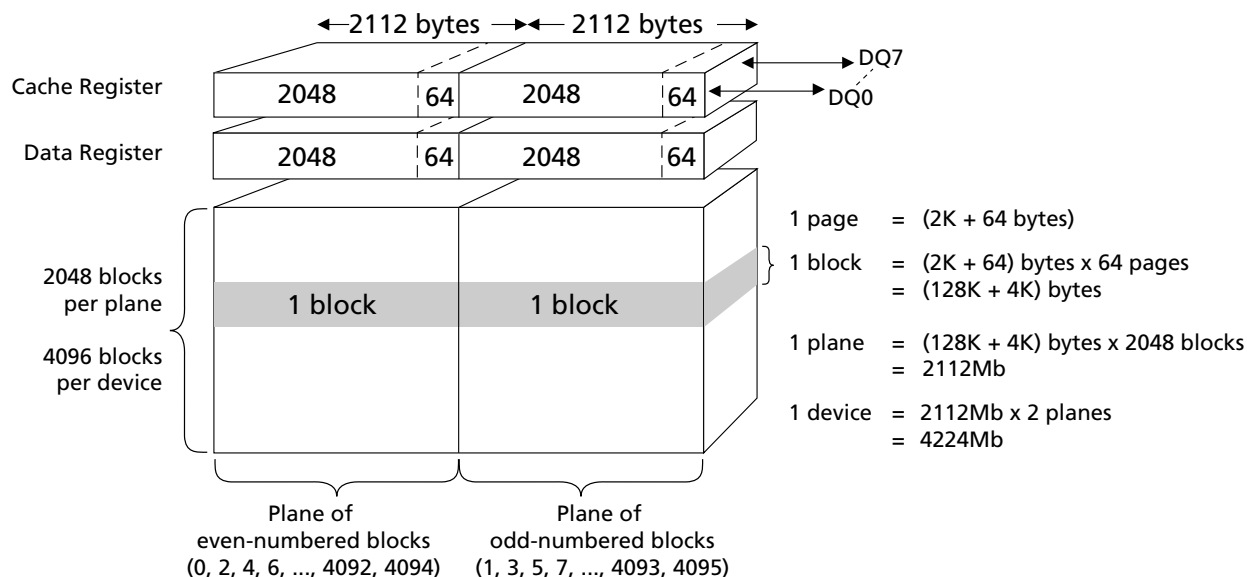




## 4Gb, 8Gb: x8, x16 NAND Flash Memory Device and Array Organization

### Device and Array Organization

**Figure 9: Array Organization – MT29F4G08 (x8)**



**Table 2: Array Addressing – MT29F4G08 (x8)**

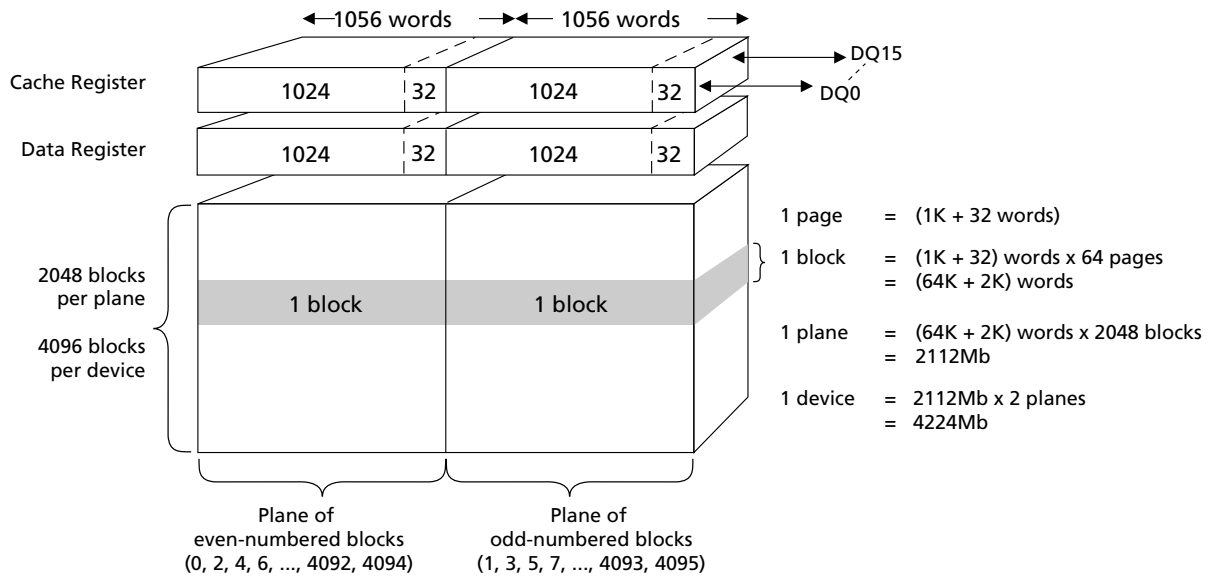
| Cycle  | I/07 | I/06 | I/05 | I/04 | I/03 | I/02 | I/01 | I/00 |
|--------|------|------|------|------|------|------|------|------|
| First  | CA7  | CA6  | CA5  | CA4  | CA3  | CA2  | CA1  | CA0  |
| Second | LOW  | LOW  | LOW  | LOW  | CA11 | CA10 | CA9  | CA8  |
| Third  | BA7  | BA6  | PA5  | PA4  | PA3  | PA2  | PA1  | PA0  |
| Fourth | BA15 | BA14 | BA13 | BA12 | BA11 | BA10 | BA9  | BA8  |
| Fifth  | LOW  | LOW  | LOW  | LOW  | LOW  | LOW  | BA17 | BA16 |

- Notes:
1. Block address concatenated with page address = actual page address. CAx = column address; PAx = page address; BAx = block address.
  2. If CA11 is 1, then CA[10:6] must be 0.
  3. BA6 controls plane selection.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Device and Array Organization

**Figure 10: Array Organization – MT29F4G16 (x16)**



**Table 3: Array Addressing – MT29F4G16 (x16)**

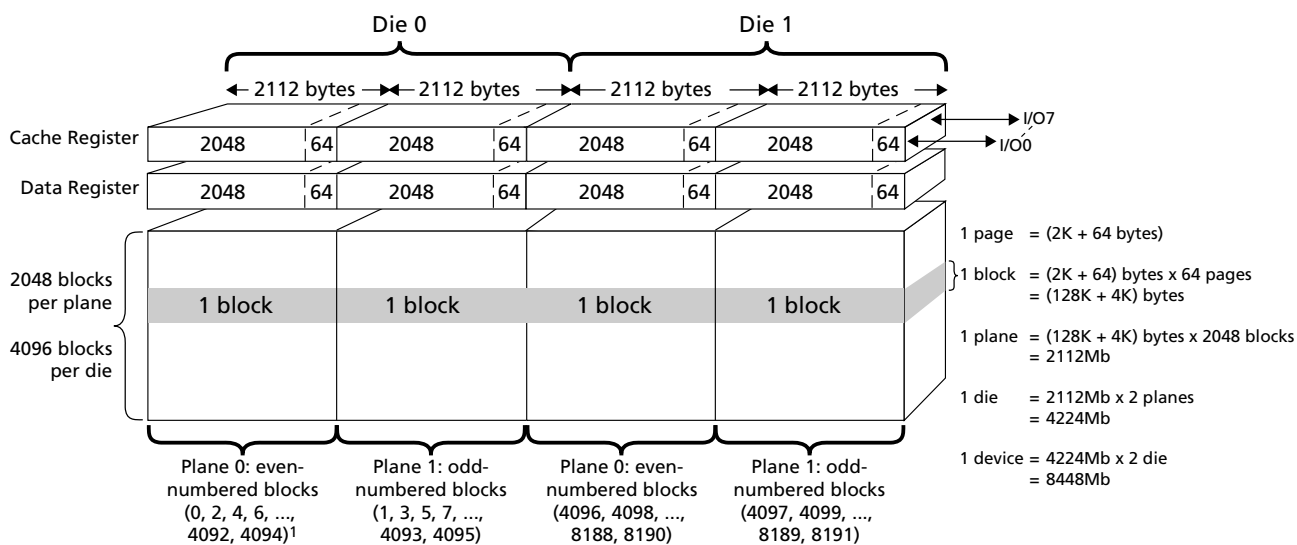
| Cycle  | I/O[15:8] | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0 |
|--------|-----------|------|------|------|------|------|------|------|------|
| First  | LOW       | CA7  | CA6  | CA5  | CA4  | CA3  | CA2  | CA1  | CA0  |
| Second | LOW       | LOW  | LOW  | LOW  | LOW  | LOW  | CA10 | CA9  | CA8  |
| Third  | LOW       | BA7  | BA6  | PA5  | PA4  | PA3  | PA2  | PA1  | PA0  |
| Fourth | LOW       | BA15 | BA14 | BA13 | BA12 | BA11 | BA10 | BA9  | BA8  |
| Fifth  | LOW       | LOW  | LOW  | LOW  | LOW  | LOW  | LOW  | BA17 | BA16 |

- Notes:
1. Block address concatenated with page address = actual page address. CAx = column address; PAx = page address; BAx = block address.
  2. If CA10 = 1, then CA[9:5] must be 0.
  3. BA6 controls plane selection.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Device and Array Organization

**Figure 11: Array Organization – MT29F8G08 (x8)**



Note: 1. Die 0, Plane 0: BA18 = 0; BA6 = 0. Die 0, Plane 1: BA18 = 0; BA6 = 1.  
Die 1, Plane 0: BA18 = 1; BA6 = 0. Die 1, Plane 1: BA18 = 1; BA6 = 1.

**Table 4: Array Addressing – MT29F8G08 (x8)**

| Cycle  | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2              | I/O1 | I/O0 |
|--------|------|------|------|------|------|-------------------|------|------|
| First  | CA7  | CA6  | CA5  | CA4  | CA3  | CA2               | CA1  | CA0  |
| Second | LOW  | LOW  | LOW  | LOW  | CA11 | CA10              | CA9  | CA8  |
| Third  | BA7  | BA6  | PA5  | PA4  | PA3  | PA2               | PA1  | PA0  |
| Fourth | BA15 | BA14 | BA13 | BA12 | BA11 | BA10              | BA9  | BA8  |
| Fifth  | LOW  | LOW  | LOW  | LOW  | LOW  | BA18 <sup>3</sup> | BA17 | BA16 |

Notes: 1. CAx = column address; PAx = page address; BAx = block address.  
2. If CA11 is 1, then CA[10:6] must be 0.  
3. Die address boundary: 0 = 0–4Gb; 1 = 4Gb–8Gb.





## Asynchronous Interface Bus Operation

The bus on the device is multiplexed. Data I/O, addresses, and commands all share the same pins. I/O[15:8] are used only for data in the x16 configuration. Addresses and commands are always supplied on I/O[7:0].

The command sequence typically consists of a COMMAND LATCH cycle, address input cycles, and one or more data cycles, either READ or WRITE.

**Table 6: Asynchronous Interface Mode Selection**

| Mode <sup>1</sup>    | CE# | CLE | ALE | WE# | RE# | I/Ox | WP#                |
|----------------------|-----|-----|-----|-----|-----|------|--------------------|
| Standby <sup>2</sup> | H   | X   | X   | X   | X   | X    | 0V/V <sub>CC</sub> |
| Command input        | L   | H   | L   |     | H   | X    | H                  |
| Address input        | L   | L   | H   |     | H   | X    | H                  |
| Data input           | L   | L   | L   |     | H   | X    | H                  |
| Data output          | L   | L   | L   | H   |     | X    | X                  |
| Write protect        | X   | X   | X   | X   | X   | X    | L                  |

- Notes: 1. Mode selection settings for this table: H = Logic level HIGH; L = Logic level LOW; X = V<sub>IH</sub> or V<sub>IL</sub>.  
2. WP# should be biased to CMOS LOW or HIGH for standby.

## Asynchronous Enable/Standby

When the device is not performing an operation, the CE# pin is typically driven HIGH and the device enters standby mode. The memory will enter standby if CE# goes HIGH while data is being transferred and the device is not busy. This helps reduce power consumption.

The CE# “Don’t Care” operation enables the NAND Flash to reside on the same asynchronous memory bus as other Flash or SRAM devices. Other devices on the memory bus can then be accessed while the NAND Flash is busy with internal operations. This capability is important for designs that require multiple NAND Flash devices on the same bus.

A HIGH CLE signal indicates that a command cycle is taking place. A HIGH ALE signal signifies that an ADDRESS INPUT cycle is occurring.

## Asynchronous Commands

An asynchronous command is written from I/O[7:0] to the command register on the rising edge of WE# when CE# is LOW, ALE is LOW, CLE is HIGH, and RE# is HIGH.

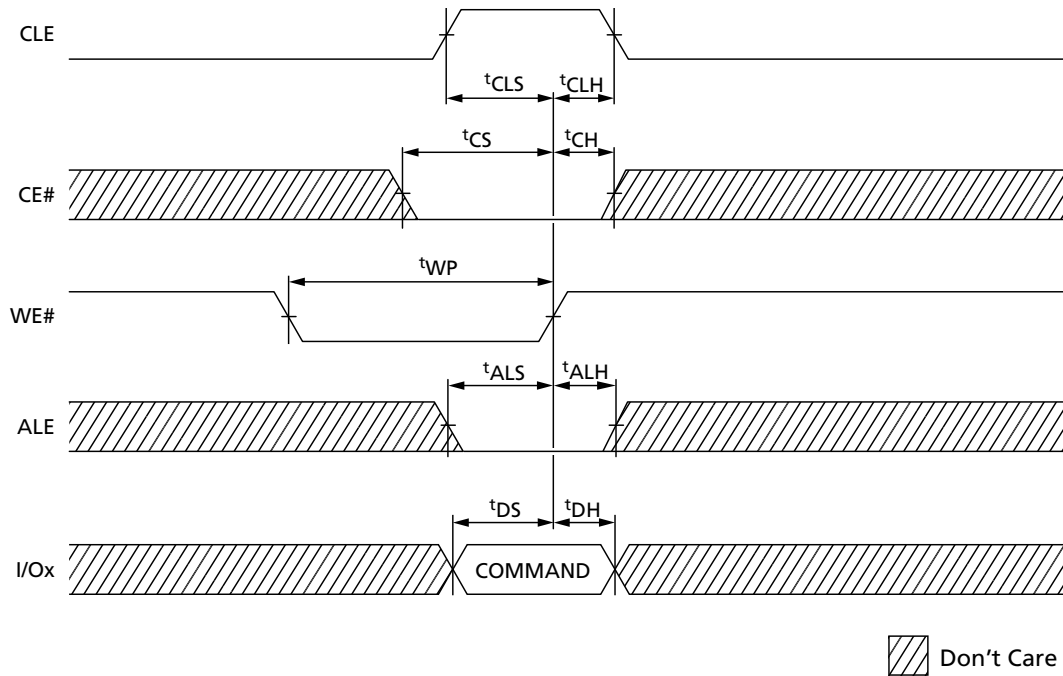
Commands are typically ignored by die (LUNs) that are busy (RDY = 0); however, some commands, including READ STATUS (70h) and READ STATUS ENHANCED (78h), are accepted by die (LUNs) even when they are busy.

For devices with a x16 interface, I/O[15:8] must be written with zeros when a command is issued.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Bus Operation

**Figure 13: Asynchronous Command Latch Cycle**





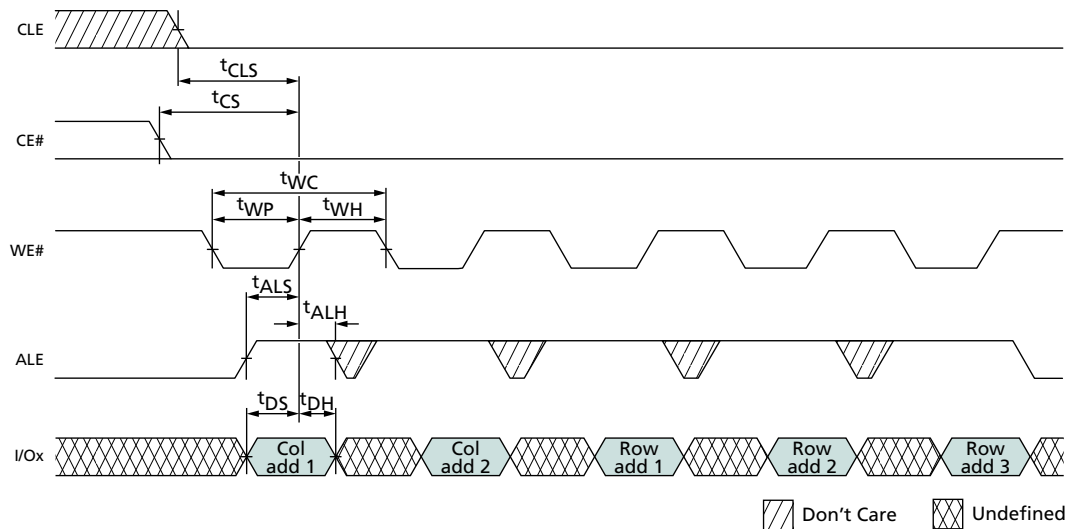
## Asynchronous Addresses

An asynchronous address is written from I/O[7:0] to the address register on the rising edge of WE# when CE# is LOW, ALE is HIGH, CLE is LOW, and RE# is HIGH.

Bits that are not part of the address space must be LOW (see Device and Array Organization). The number of cycles required for each command varies. Refer to the command descriptions to determine addressing requirements.

Addresses are typically ignored by die (LUNs) that are busy (RDY = 0); however, some addresses are accepted by die (LUNs) even when they are busy; for example, like address cycles that follow the READ STATUS ENHANCED (78h) command.

**Figure 14: Asynchronous Address Latch Cycle**







## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Bus Operation

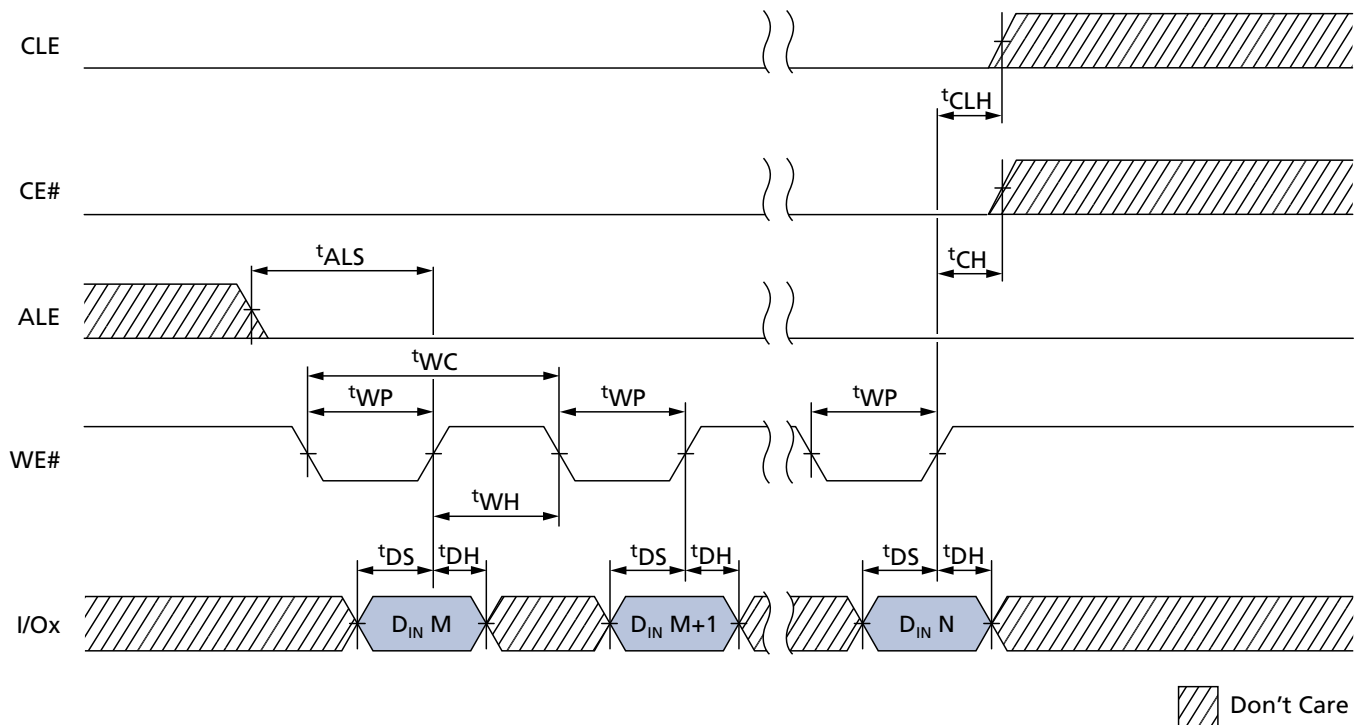
### Asynchronous Data Input

Data is written from I/O[7:0] to the cache register of the selected die (LUN) on the rising edge of WE# when CE# is LOW, ALE is LOW, CLE is LOW, and RE# is HIGH.

Data input is ignored by die (LUNs) that are not selected or are busy (RDY = 0). Data is written to the data register on the rising edge of WE# when CE#, CLE, and ALE are LOW, and the device is not busy.

Data is input on I/O[7:0] on x8 devices and on I/O[15:0] on x16 devices.

**Figure 15: Asynchronous Data Input Cycles**





## Asynchronous Data Output

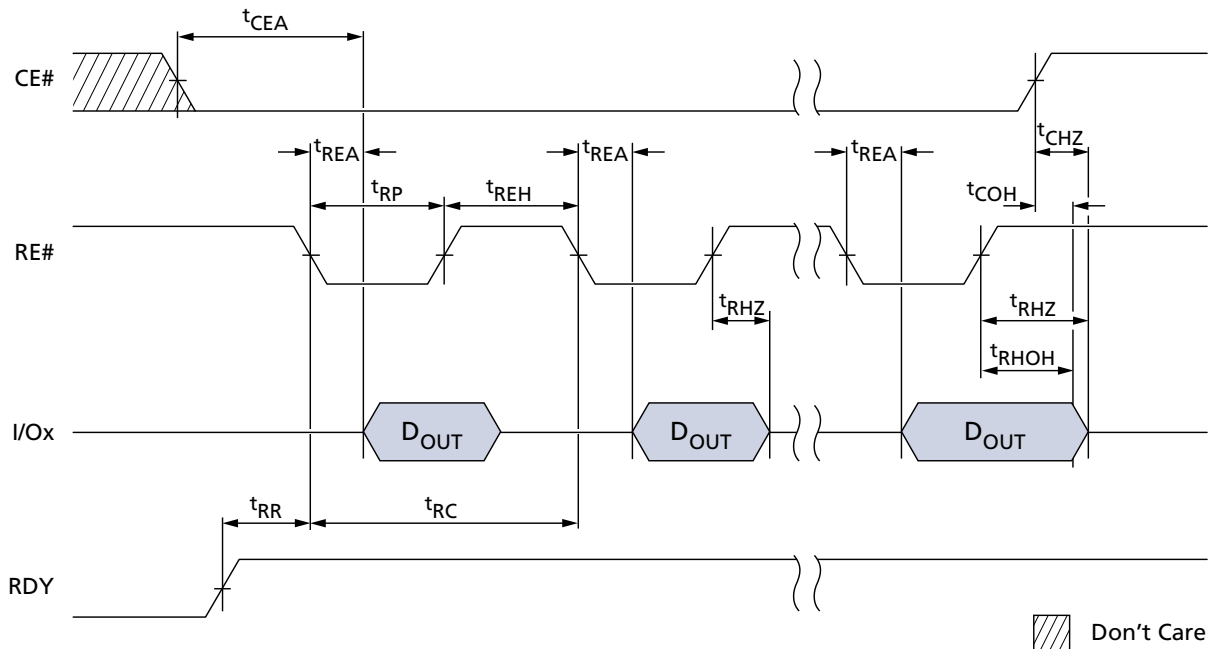
Data can be output from a die (LUN) if it is in a READY state. Data output is supported following a READ operation from the NAND Flash array. Data is output from the cache register of the selected die (LUN) to I/O[7:0] on the falling edge of RE# when CE# is LOW, ALE is LOW, CLE is LOW, and WE# is HIGH.

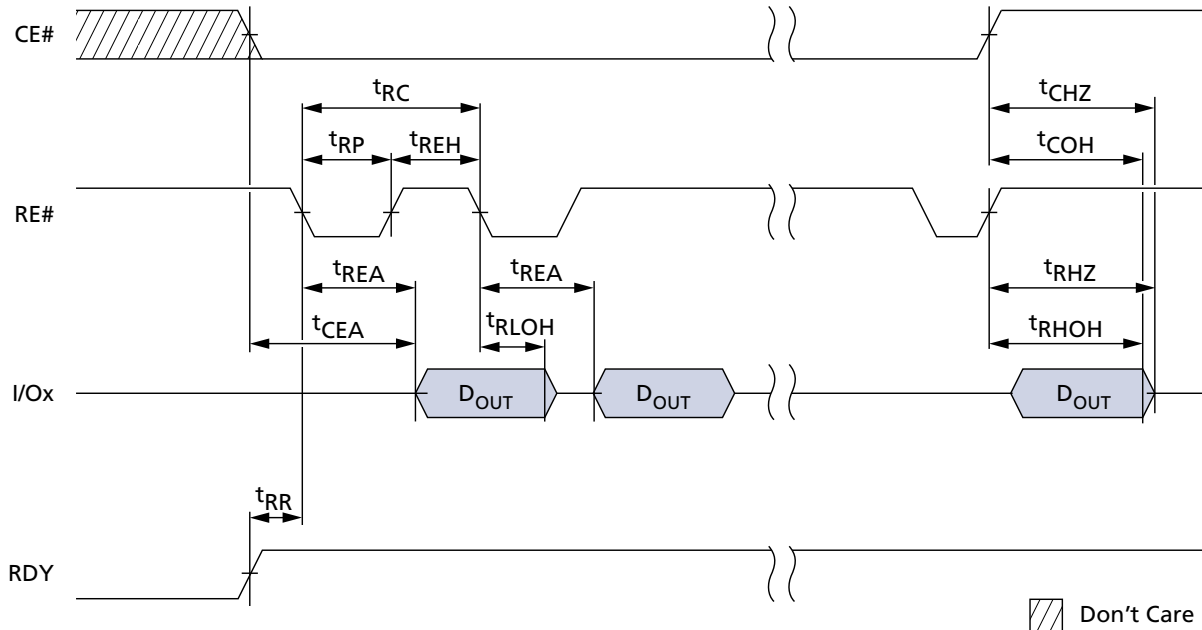
If the host controller is using a  $t_{RC}$  of 30ns or greater, the host can latch the data on the rising edge of RE# (see the figure below for proper timing). If the host controller is using a  $t_{RC}$  of less than 30ns, the host can latch the data on the next falling edge of RE#.

Using the READ STATUS ENHANCED (78h) command prevents data contention following an interleaved die (multi-LUN) operation. After issuing the READ STATUS ENHANCED (78h) command, to enable data output, issue the READ MODE (00h) command.

Data output requests are typically ignored by a die (LUN) that is busy (RDY = 0); however, it is possible to output data from the status register even when a die (LUN) is busy by first issuing the READ STATUS or READ STATUS ENHANCED (78h) command.

**Figure 16: Asynchronous Data Output Cycles**




**Figure 17: Asynchronous Data Output Cycles (EDO Mode)**


## Write Protect#

The write protect# (WP#) signal enables or disables PROGRAM and ERASE operations to a target. When WP# is LOW, PROGRAM and ERASE operations are disabled. When WP# is HIGH, PROGRAM and ERASE operations are enabled.

It is recommended that the host drive WP# LOW during power-on until  $V_{CC}$  is stable to prevent inadvertent PROGRAM and ERASE operations (see Device Initialization for additional details).

WP# must be transitioned only when the target is not busy and prior to beginning a command sequence. After a command sequence is complete and the target is ready, WP# can be transitioned. After WP# is transitioned, the host must wait  $t_{WW}$  before issuing a new command.

The WP# signal is always an active input, even when CE# is HIGH. This signal should not be multiplexed with other signals.

## Ready/Busy#

The ready/busy# (R/B#) signal provides a hardware method of indicating whether a target is ready or busy. A target is busy when one or more of its die (LUNs) are busy ( $RDY = 0$ ). A target is ready when all of its die (LUNs) are ready ( $RDY = 1$ ). Because each die (LUN) contains a status register, it is possible to determine the independent status of each die (LUN) by polling its status register instead of using the R/B# signal (see Status Operations for details regarding die (LUN) status).

This signal requires a pull-up resistor,  $R_p$ , for proper operation. R/B# is HIGH when the target is ready, and transitions LOW when the target is busy. The signal's open-drain



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Bus Operation

driver enables multiple R/B# outputs to be OR-tied. Typically, R/B# is connected to an interrupt pin on the system controller.

The combination of  $R_p$  and capacitive loading of the R/B# circuit determines the rise time of the R/B# signal. The actual value used for  $R_p$  depends on the system timing requirements. Large values of  $R_p$  cause R/B# to be delayed significantly. Between the 10% and 90% points on the R/B# waveform, the rise time is approximately two time constants (TC).

$$TC = R \times C$$

Where  $R = R_p$  (resistance of pull-up resistor), and  $C$  = total capacitive load.

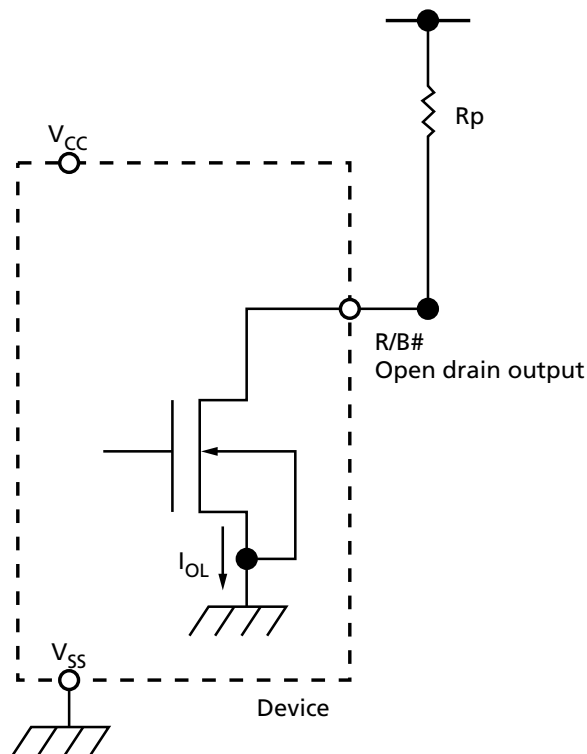
The fall time of the R/B# signal is determined mainly by the output impedance of the R/B# signal and the total load capacitance. Approximate  $R_p$  values using a circuit load of 100pF are provided in Figure 23 (page 29).

The minimum value for  $R_p$  is determined by the output drive capability of the R/B# signal, the output voltage swing, and  $V_{CC}$ .

$$R_p = \frac{V_{CC} (MAX) - V_{OL} (MAX)}{I_{OL} + \Sigma I_L}$$

Where  $\Sigma I_L$  is the sum of the input currents of all devices tied to the R/B# pin.

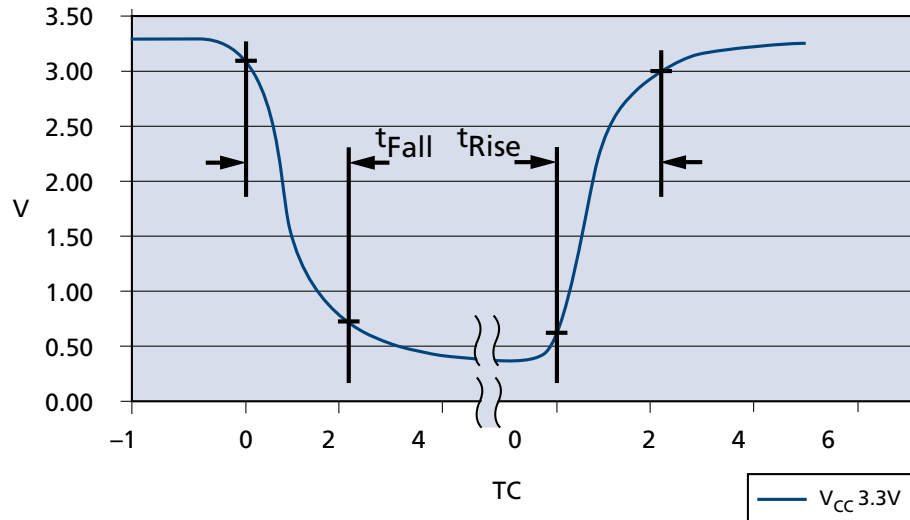
**Figure 18: READ/BUSY# Open Drain**





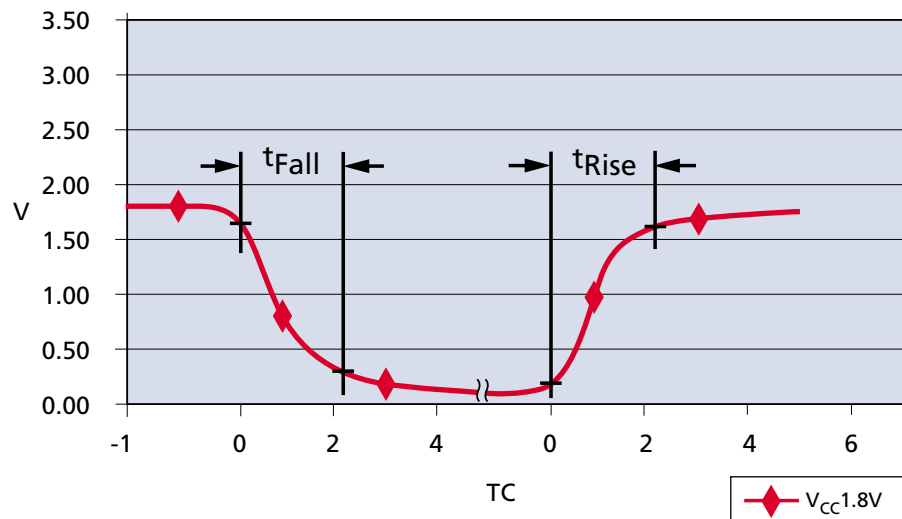
## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Bus Operation

**Figure 19:  $t_{\text{Fall}}$  and  $t_{\text{Rise}}$  (3.3V  $V_{\text{CC}}$ )**



- Notes:
1.  $t_{\text{Fall}}$  and  $t_{\text{Rise}}$  calculated at 10% and 90% points.
  2.  $t_{\text{Rise}}$  dependent on external capacitance and resistive loading and output transistor impedance.
  3.  $t_{\text{Rise}}$  primarily dependent on external pull-up resistor and external capacitive loading.
  4.  $t_{\text{Fall}} = 10\text{ns}$  at 3.3V.
  5. See TC values in Figure 23 (page 29) for approximate  $R_p$  value and TC.

**Figure 20:  $t_{\text{Fall}}$  and  $t_{\text{Rise}}$  (1.8V  $V_{\text{CC}}$ )**

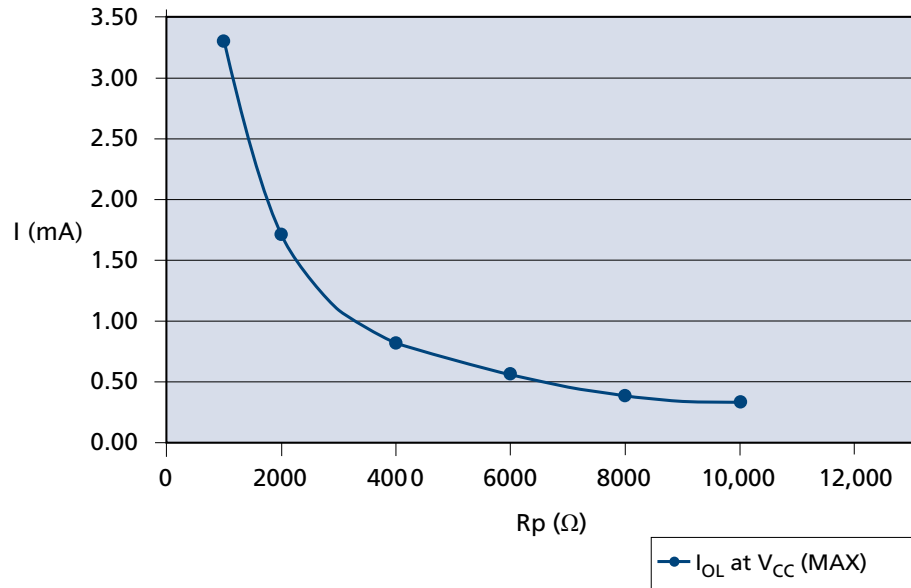


- Notes:
1.  $t_{\text{Fall}}$  and  $t_{\text{Rise}}$  are calculated at 10% and 90% points.
  2.  $t_{\text{Rise}}$  is primarily dependent on external pull-up resistor and external capacitive loading.
  3.  $t_{\text{Fall}} \approx 7\text{ns}$  at 1.8V.
  4. See TC values in Figure 23 (page 29) for TC and approximate  $R_p$  value.

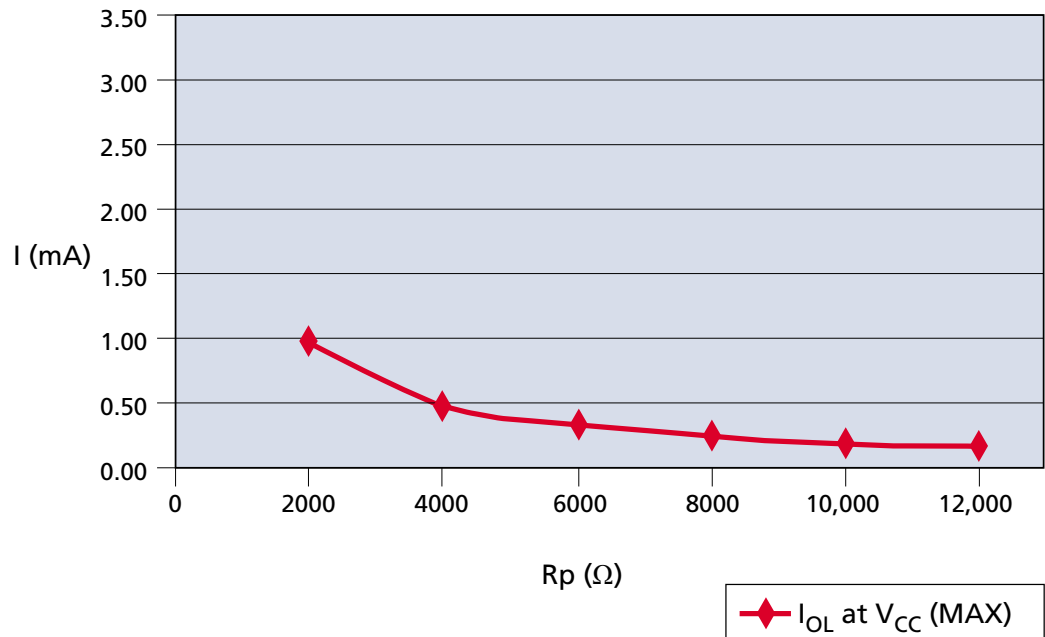


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Bus Operation

**Figure 21:  $I_{OL}$  vs.  $R_p$  ( $V_{CC} = 3.3V$   $V_{CC}$ )**



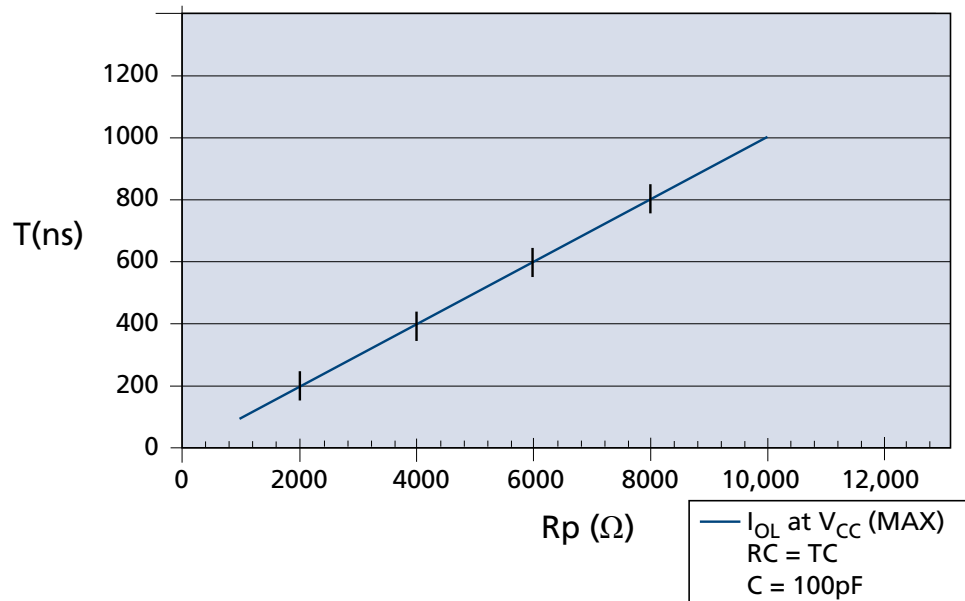
**Figure 22:  $I_{OL}$  vs.  $R_p$  (1.8V  $V_{CC}$ )**





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Bus Operation

Figure 23: TC vs. Rp





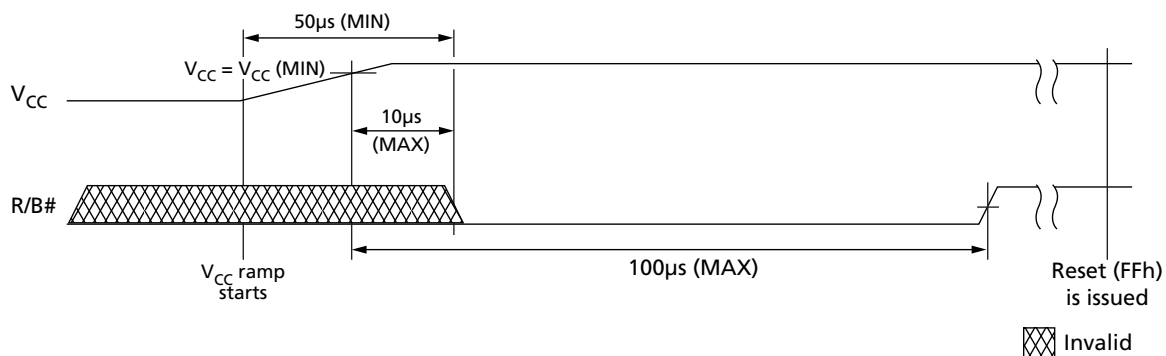


## Device Initialization

Micron NAND Flash devices are designed to prevent data corruption during power transitions.  $V_{CC}$  is internally monitored. (The  $WP\#$  signal supports additional hardware protection during power transitions.) When ramping  $V_{CC}$ , use the following procedure to initialize the device:

1. Ramp  $V_{CC}$ .
2. The host must wait for  $R/B\#$  to be valid and HIGH before issuing RESET (FFh) to any target. The  $R/B\#$  signal becomes valid when  $50\mu s$  has elapsed since the beginning the  $V_{CC}$  ramp, and  $10\mu s$  has elapsed since  $V_{CC}$  reaches  $V_{CC} (MIN)$ .
3. If not monitoring  $R/B\#$ , the host must wait at least  $100\mu s$  after  $V_{CC}$  reaches  $V_{CC} (MIN)$ . If monitoring  $R/B\#$ , the host must wait until  $R/B\#$  is HIGH.
4. The asynchronous interface is active by default for each target. Each LUN draws less than an average of  $10mA$  ( $I_{ST}$ ) measured over intervals of  $1ms$  until the RESET (FFh) command is issued.
5. The RESET (FFh) command must be the first command issued to all targets ( $CE\#s$ ) after the NAND Flash device is powered on. Each target will be busy for  $1ms$  after a RESET command is issued. The RESET busy time can be monitored by polling  $R/B\#$  or issuing the READ STATUS (70h) command to poll the status register.
6. The device is now initialized and ready for normal operation.

**Figure 24: R/B# Power-On Behavior**





## Command Definitions

**Table 7: Command Set**

| Command                              | Command Cycle #1 | Number of Valid Address Cycles | Data Input Cycles | Command Cycle #2 | Valid While Selected LUN is Busy <sup>1</sup> | Valid While Other LUNs are Busy <sup>2</sup> | Notes |
|--------------------------------------|------------------|--------------------------------|-------------------|------------------|---|--|-------|
| <b>Reset Operations</b>              |                  |                                |                   |                  |   |  |       |
| RESET                                | FFh              | 0                              | –                 | –                | Yes   | Yes  |       |
| <b>Identification Operation</b>      |                  |                                |                   |                  |   |  |       |
| READ ID                              | 90h              | 1                              | –                 | –                | No  | No   |       |
| READ PARAMETER PAGE                  | ECh              | 1                              | –                 | –                | No  | No   |       |
| READ UNIQUE ID                       | EDh              | 1                              | –                 | –                | No  | No   |       |
| <b>Feature Operations</b>            |                  |                                |                   |                  |   |  |       |
| GET FEATURES                         | EEh              | 1                              | –                 | –                | No  | No   |       |
| SET FEATURES                         | EFh              | 1                              | 4                 | –                | No  | No   |       |
| <b>Status Operations</b>             |                  |                                |                   |                  |   |  |       |
| READ STATUS                          | 70h              | 0                              | –                 | –                | Yes   |  |       |
| READ STATUS ENHANCED                 | 78h              | 3                              | –                 | –                | Yes   | Yes  |       |
| <b>Column Address Operations</b>     |                  |                                |                   |                  |   |  |       |
| RANDOM DATA READ                     | 05h              | 2                              | –                 | E0h              | No  | Yes  |       |
| RANDOM DATA INPUT                    | 85h              | 2                              | Optional          | –                | No  | Yes  |       |
| PROGRAM FOR INTERNAL DATA MOVE       | 85h              | 5                              | Optional          | –                | No  | Yes  | 3     |
| <b>READ OPERATIONS</b>               |                  |                                |                   |                  |   |  |       |
| READ MODE                            | 00h              | 0                              | –                 | –                | No  | Yes  |       |
| READ PAGE                            | 00h              | 5                              | –                 | 30h              | No  | Yes  |       |
| READ PAGE CACHE SE-SEQUENTIAL        | 31h              | 0                              | –                 | –                | No  | Yes  | 4, 5  |
| READ PAGE CACHE RANDOM               | 00h              | 5                              | –                 | 31h              | No  | Yes  | 4, 5  |
| READ PAGE CACHE LAST                 | 3Fh              | 0                              | –                 | –                | No  | Yes  | 4, 5  |
| <b>Program Operations</b>            |                  |                                |                   |                  |   |  |       |
| PROGRAM PAGE                         | 80h              | 5                              | Yes               | 10h              | No  | Yes  |       |
| PROGRAM PAGE CACHE                   | 80h              | 5                              | Yes               | 15h              | No  | Yes  | 4, 6  |
| <b>Erase Operations</b>              |                  |                                |                   |                  |   |  |       |
| ERASE BLOCK                          | 60h              | 3                              | –                 | D0h              | No  | Yes  |       |
| <b>Internal Data Move Operations</b> |                  |                                |                   |                  |   |  |       |
| READ FOR INTERNAL DATA MOVE          | 00h              | 5                              | –                 | 35h              | No  | Yes  | 3     |
| PROGRAM FOR INTERNAL DATA MOVE       | 85h              | 5                              | Optional          | 10h              | No  | Yes  |       |



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Command Definitions

**Table 7: Command Set (Continued)**

| Command                                       | Command Cycle #1 | Number of Valid Address Cycles | Data Input Cycles | Command Cycle #2 | Valid While Selected LUN is Busy <sup>1</sup> | Valid While Other LUNs are Busy <sup>2</sup> | Notes |
|---|------------------|--------------------------------|-------------------|------------------|---|--|-------|
| <b>Block Lock Operations</b>                  |                  |                                |                   |                  |   |  |       |
| BLOCK UNLOCK LOW                              | 23h              | 3                              | –                 | –                | No  | Yes  |       |
| BLOCK UNLOCK HIGH                             | 24h              | 3                              | –                 | –                | No  | Yes  |       |
| BLOCK LOCK                                    | 2Ah              | –                              | –                 | –                | No  | Yes  |       |
| BLOCK LOCK-TIGHT                              | 2Ch              | –                              | –                 | –                | No  | Yes  |       |
| BLOCK LOCK READ STATUS                        | 7Ah              | 3                              | –                 | –                | No  | Yes  |       |
| <b>One-Time Programmable (OTP) Operations</b> |                  |                                |                   |                  |   |  |       |
| OTP DATA LOCK BY BLOCK (ONFI)                 | 80h              | 5                              | No                | 10h              | No  | No   | 7     |
| OTP DATA PROGRAM (ONFI)                       | 80h              | 5                              | Yes               | 10h              | No  | No   | 7     |
| OTP DATA READ (ONFI)                          | 00h              | 5                              | No                | 30h              | No  | No   | 7     |

- Notes:
1. Busy means RDY = 0.
  2. These commands can be used for interleaved die (multi-LUN) operations (see on page ).
  3. Do not cross plane address boundaries when using READ for INTERNAL DATA MOVE and PROGRAM for INTERNAL DATA MOVE.
  4. These commands supported only with ECC disabled.
  5. Issuing a READ PAGE CACHE series (31h, 00h-31h, 3Fh) command when the array is busy (RDY = 1, ARDY = 0) is supported if the previous command was a READ PAGE (00h-30h) or READ PAGE CACHE series command; otherwise, it is prohibited.
  6. Issuing a PROGRAM PAGE CACHE (80h-15h) command when the array is busy (RDY = 1, ARDY = 0) is supported if the previous command was a PROGRAM PAGE CACHE (80h-15h) command; otherwise, it is prohibited.
  7. OTP commands can be entered only after issuing the SET FEATURES command with the feature address.

**Table 8: Two-Plane Command Set**

Note 4 applies to all parameters and conditions

| Command                               | Command Cycle #1 | Number of Valid Address Cycles | Command Cycle #2 | Number of Valid Address Cycles | Command Cycle #3 | Valid While Selected LUN is Busy | Valid While Other LUNs are Busy | Notes |
|---------------------------------------|------------------|--------------------------------|------------------|--------------------------------|------------------|----------------------------------|---------------------------------|-------|
| READ PAGE TWO-PLANE                   | 00h              | 5                              | 00h              | 5                              | 30h              | No                               | Yes                             |       |
| READ FOR TWO-PLANE INTERNAL DATA MOVE | 00h              | 5                              | 00h              | 5                              | 35h              | No                               | Yes                             | 1     |
| RANDOM DATA READ TWO-PLANE            | 06h              | 5                              | E0h              | –                              | –                | No                               | Yes                             | 2     |



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Command Definitions

**Table 8: Two-Plane Command Set (Continued)**

Note 4 applies to all parameters and conditions

| Command   | Com-<br>mand<br>Cycle #1 | Number of<br>Valid<br>Address<br>Cycles | Com-<br>mand<br>Cycle #2 | Number of<br>Valid<br>Address<br>Cycles | Com-<br>mand<br>Cycle #3 | Valid While<br>Selected<br>LUN is Busy | Valid While<br>Other LUNs<br>are Busy | Notes |
|---|--------------------------|---|--------------------------|---|--------------------------|--|---------------------------------------|-------|
| PROGRAM PAGE<br>TWO-PLANE                         | 80h                      | 5                                       | 11h-80h                  | 5                                       | 10h                      | No                                     | Yes                                   |       |
| PROGRAM PAGE<br>CACHE MODE TWO-<br>PLANE          | 80h                      | 5                                       | 11h-80h                  | 5                                       | 15h                      | No                                     | Yes                                   |       |
| PROGRAM FOR TWO-<br>PLANE INTERNAL DA-<br>TA MOVE | 85h                      | 5                                       | 11h-85h                  | 5                                       | 10h                      | No                                     | Yes                                   | 1     |
| BLOCK ERASE TWO-<br>PLANE                         | 60h                      | 3                                       | D1h-60h                  | 3                                       | D0h                      | No                                     | Yes                                   | 3     |

- Notes:
1. Do not cross plane boundaries when using READ FOR INTERNAL DATA MOVE TWO-PLANE or PROGRAM FOR TWO-PLANE INTERNAL DATA MOVE.
  2. The RANDOM DATA READ TWO-PLANE command is limited to use with the PAGE READ TWO-PLANE command.
  3. D1h command can be omitted.
  4. These commands supported only with ECC disabled.



## Reset Operations

### RESET (FFh)

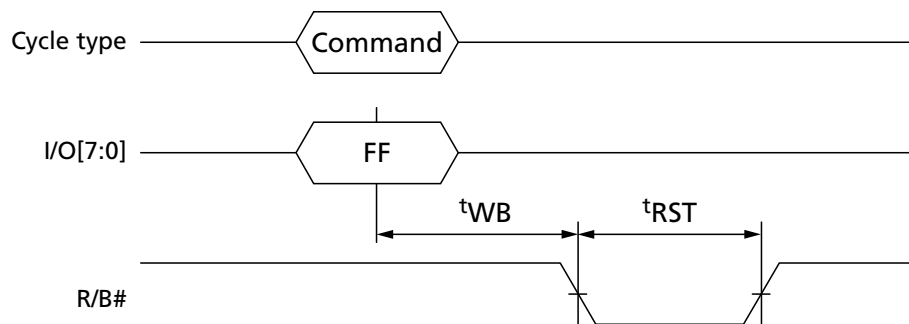
The RESET command is used to put the memory device into a known condition and to abort the command sequence in progress.

READ, PROGRAM, and ERASE commands can be aborted while the device is in the busy state. The contents of the memory location being programmed or the block being erased are no longer valid. The data may be partially erased or programmed, and is invalid. The command register is cleared and is ready for the next command. The data register and cache register contents are marked invalid.

The status register contains the value E0h when WP# is HIGH; otherwise it is written with a 60h value. R/B# goes LOW for  $t_{RST}$  after the RESET command is written to the command register.

The RESET command must be issued to all CE#s as the first command after power-on. The device will be busy for a maximum of 1ms.

**Figure 25: RESET (FFh) Operation**





## Identification Operations

### READ ID (90h)

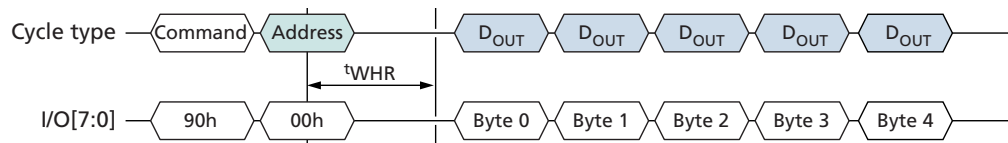
The READ ID (90h) command is used to read identifier codes programmed into the target. This command is accepted by the target only when all die (LUNs) on the target are idle.

Writing 90h to the command register puts the target in read ID mode. The target stays in this mode until another valid command is issued.

When the 90h command is followed by an 00h address cycle, the target returns a 5-byte identifier code that includes the manufacturer ID, device configuration, and part-specific information.

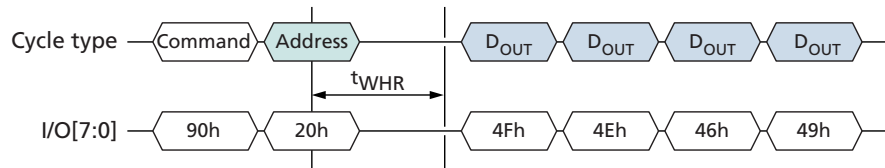
When the 90h command is followed by a 20h address cycle, the target returns the 4-byte ONFI identifier code.

**Figure 26: READ ID (90h) with 00h Address Operation**



Note: 1. See the READ ID Parameter tables for byte definitions.

**Figure 27: READ ID (90h) with 20h Address Operation**



Note: 1. See READ ID Parameter tables for byte definitions.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory READ ID Parameter Tables

### READ ID Parameter Tables

**Table 9: READ ID Parameters for Address 00h**

b = binary; h = hexadecimal

|   |                | Options        | I/07 | I/06 | I/05 | I/04 | I/03 | I/02 | I/01 | I/00 | Value  |
|---|----------------|----------------|------|------|------|------|------|------|------|------|--------|
| <b>Byte 0 – Manufacturer ID</b>             |                |                |      |      |      |      |      |      |      |      |        |
| Manufacturer                                |                | Micron         | 0    | 0    | 1    | 0    | 1    | 1    | 0    | 0    | 2Ch    |
| <b>Byte 1 – Device ID</b>                   |                |                |      |      |      |      |      |      |      |      |        |
| MT29F4G08ABADA                              |                | 4Gb, x8, 3.3V  | 1    | 1    | 0    | 1    | 1    | 1    | 0    | 0    | DCh    |
| MT29F4G16ABADA                              |                | 4Gb, x16, 3.3V | 1    | 1    | 0    | 0    | 1    | 1    | 0    | 0    | CCh    |
| MT29F4G08ABBDA                              |                | 4Gb, x8, 1.8V  | 1    | 0    | 1    | 0    | 1    | 1    | 0    | 0    | ACh    |
| MT29F4G16ABBDA                              |                | 4Gb, x16, 1.8V | 1    | 0    | 1    | 1    | 1    | 1    | 0    | 0    | BCh    |
| MT29F8G08ADBDA                              |                | 8Gb, x8, 1.8V  | 1    | 0    | 1    | 0    | 0    | 0    | 1    | 1    | A3h    |
| MT29F8G16ADBDA                              |                | 8Gb, x16, 1.8V | 1    | 0    | 1    | 1    | 0    | 0    | 1    | 1    | B3h    |
| MT29F8G08ADADA                              |                | 8Gb, x8, 3.3V  | 1    | 1    | 0    | 1    | 0    | 0    | 1    | 1    | D3h    |
| MT29F8G16ADADA                              |                | 8Gb, x16, 3.3V | 1    | 1    | 0    | 0    | 0    | 0    | 1    | 1    | C3h    |
| <b>Byte 2</b>                               |                |                |      |      |      |      |      |      |      |      |        |
| Number of die per CE                        | 1              |                |      |      |      |      |      |      | 0    | 0    | 00b    |
|   | 2              |                |      |      |      |      |      |      | 0    | 1    | 01b    |
| Cell type                                   | SLC            |                |      |      |      |      | 0    | 0    |      |      | 00b    |
| Number of simultaneously programmed pages   | 2              |                |      | 0    | 1    |      |      |      |      |      | 01b    |
| Interleaved operations between multiple die | Not supported  |                | 0    |      |      |      |      |      |      |      | 0b     |
| Cache programming                           | Supported      |                | 1    |      |      |      |      |      |      |      | 1b     |
| Byte value                                  | MT29F4G08ABADA |                | 1    | 0    | 0    | 1    | 0    | 0    | 0    | 0    | 90h    |
|   | MT29F4G16ABADA |                | 1    | 0    | 0    | 1    | 0    | 0    | 0    | 0    | 90h    |
|   | MT29F4G08ABBDA |                | 1    | 0    | 0    | 1    | 0    | 0    | 0    | 0    | 90h    |
|   | MT29F4G16ABBDA |                | 1    | 0    | 0    | 1    | 0    | 0    | 0    | 0    | 90h    |
|   | MT29F8G08ADBDA |                | 1    | 1    | 0    | 1    | 0    | 0    | 0    | 1    | D1h    |
|   | MT29F8G16ADBDA |                | 1    | 1    | 0    | 1    | 0    | 0    | 0    | 1    | D1h    |
|   | MT29F8G08ADADA |                | 1    | 1    | 0    | 1    | 0    | 0    | 0    | 1    | D1h    |
|   | MT29F8G16ADADA |                | 1    | 1    | 0    | 1    | 0    | 0    | 0    | 1    | D1h    |
| <b>Byte 3</b>                               |                |                |      |      |      |      |      |      |      |      |        |
| Page size                                   |                | 2KB            |      |      |      |      |      |      | 0    | 1    | 01b    |
| Spare area size (bytes)                     |                | 64B            |      |      |      |      |      | 1    |      |      | 1b     |
| Block size (without spare)                  |                | 128KB          |      |      | 0    | 1    |      |      |      |      | 01b    |
| Organization                                | x8             |                |      | 0    |      |      |      |      |      |      | 0b     |
|   | x16            |                |      | 1    |      |      |      |      |      |      | 1b     |
| Serial access (MIN)                         | 1.8V           | 25ns           | 0    |      |      |      | 0    |      |      |      | 0xxx0b |
|   | 3.3V           | 20ns           | 1    |      |      |      | 0    |      |      |      | 1xxx0b |





## 4Gb, 8Gb: x8, x16 NAND Flash Memory READ ID Parameter Tables

**Table 9: READ ID Parameters for Address 00h (Continued)**

b = binary; h = hexadecimal

|                    | Options   | I/07 | I/06 | I/05 | I/04 | I/03 | I/02 | I/01 | I/00 | Value |
|--------------------|---|------|------|------|------|------|------|------|------|-------|
| Byte value         | MT29F4G08ABADA                                      | 1    | 0    | 0    | 1    | 0    | 1    | 0    | 1    | 95h   |
|                    | MT29F4G16ABADA                                      | 1    | 1    | 0    | 1    | 0    | 1    | 0    | 1    | D5h   |
|                    | MT29F4G08ABBDA                                      | 0    | 0    | 0    | 1    | 0    | 1    | 0    | 1    | 15h   |
|                    | MT29F4G16ABBDA                                      | 0    | 1    | 0    | 1    | 0    | 1    | 0    | 1    | 55h   |
|                    | MT29F8G08ADBDA                                      | 0    | 0    | 0    | 1    | 0    | 1    | 0    | 1    | 15h   |
|                    | MT29F8G16ADBDA                                      | 0    | 1    | 0    | 1    | 0    | 1    | 0    | 1    | 55h   |
|                    | MT29F8G08ADADA                                      | 1    | 0    | 0    | 1    | 0    | 1    | 0    | 1    | 95h   |
|                    | MT29F8G16ADADA                                      | 1    | 1    | 0    | 1    | 0    | 1    | 0    | 1    | D5h   |
| <b>Byte 4</b>      |   |      |      |      |      |      |      |      |      |       |
| Internal ECC level | 4-bit ECC/512 (main) + 4 (spare) + 8 (parity) bytes |      |      |      |      |      |      | 1    | 0    | 10b   |
| Planes per CE#     | 2   |      |      |      |      | 0    | 1    |      |      | 01b   |
|                    | 4   |      |      |      |      | 1    | 0    |      |      | 10b   |
| Plane size         | 2Gb   |      | 1    | 0    | 1    |      |      |      |      | 101b  |
| Internal ECC       | ECC disabled  | 0    |      |      |      |      |      |      |      | 0b    |
|                    | ECC enabled   | 1    |      |      |      |      |      |      |      | 1b    |
| Byte value         | MT29F4G08ABADA                                      | 0    | 1    | 0    | 1    | 0    | 1    | 1    | 0    | 56h   |
|                    | MT29F4G16ABADA                                      | 0    | 1    | 0    | 1    | 0    | 1    | 1    | 0    | 56h   |
|                    | MT29F4G08ABBDA                                      | 0    | 1    | 0    | 1    | 0    | 1    | 1    | 0    | 56h   |
|                    | MT29F4G16ABBDA                                      | 0    | 1    | 0    | 1    | 0    | 1    | 1    | 0    | 56h   |
|                    | MT29F8G08ADBDA                                      | 0    | 1    | 0    | 1    | 1    | 0    | 1    | 0    | 5Ah   |
|                    | MT29F8G16ADBDA                                      | 0    | 1    | 0    | 1    | 1    | 0    | 1    | 0    | 5Ah   |
|                    | MT29F8G08ADADA                                      | 0    | 1    | 0    | 1    | 1    | 0    | 1    | 0    | 5Ah   |
|                    | MT29F8G16ADADA                                      | 0    | 1    | 0    | 1    | 1    | 0    | 1    | 0    | 5Ah   |

**Table 10: READ ID Parameters for Address 20h**

h = hexadecimal

| Byte | Options   | I/07 | I/06 | I/05 | I/04 | I/03 | I/02 | I/01 | I/00 | Value |
|------|-----------|------|------|------|------|------|------|------|------|-------|
| 0    | "O"       | 0    | 1    | 0    | 0    | 1    | 1    | 1    | 1    | 4Fh   |
| 1    | "N"       | 0    | 1    | 0    | 0    | 1    | 1    | 1    | 0    | 4Eh   |
| 2    | "F"       | 0    | 1    | 0    | 0    | 0    | 1    | 1    | 0    | 46h   |
| 3    | "I"       | 0    | 1    | 0    | 0    | 1    | 0    | 0    | 1    | 49h   |
| 4    | Undefined | X    | X    | X    | X    | X    | X    | X    | X    | XXh   |



## READ PARAMETER PAGE (ECh)

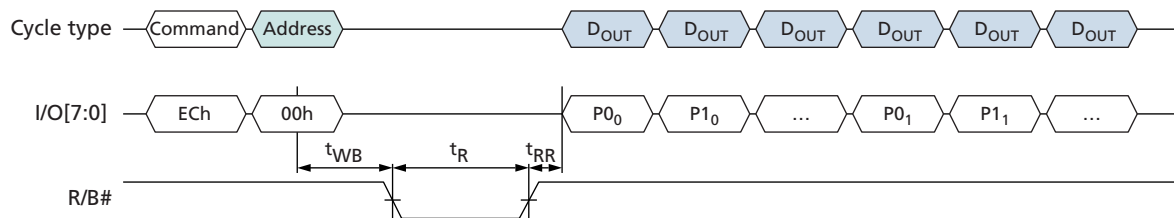
The READ PARAMETER PAGE (ECh) command is used to read the ONFI parameter page programmed into the target. This command is accepted by the target only when all die (LUNs) on the target are idle.

Writing ECh to the command register puts the target in read parameter page mode. The target stays in this mode until another valid command is issued.

When the ECh command is followed by an 00h address cycle, the target goes busy for  $t_R$ . If the READ STATUS (70h) command is used to monitor for command completion, the READ MODE (00h) command must be used to re-enable data output mode. Use of the READ STATUS ENHANCED (78h) command is prohibited while the target is busy and during data output.

A minimum of three copies of the parameter page are stored in the device. Each parameter page is 256 bytes. If desired, the RANDOM DATA READ (05h-E0h) command can be used to change the location of data output.

**Figure 28: READ PARAMETER (ECh) Operation**





## Parameter Page Data Structure Tables

**Table 11: Parameter Page Data Structure**

| Byte  | Description                 | Value  |
|-------|-----------------------------|--|
| 0–3   | Parameter page signature    | 4Fh, 4Eh, 46h, 49h   |
| 4–5   | Revision number             | 02h, 00h   |
| 6–7   | Features supported          | MT29F4G08ABBD4H4   |
|       |                             | MT29F4G08ABBD4HC   |
|       |                             | MT29F4G16ABBD4HC   |
|       |                             | MT29F4G16ABBD4H4   |
|       |                             | MT29F8G08ADBDAH4   |
|       |                             | MT29F8G16ADBDAH4   |
|       |                             | MT29F4G08ABADAWP   |
|       |                             | MT29F4G08ABADAH4   |
|       |                             | MT29F4G16ABADAWP   |
|       |                             | MT29F4G16ABADAH4   |
|       |                             | MT29F8G08ADADAH4   |
|       |                             | MT29F8G16ADADAH4   |
| 8–9   | Optional commands supported | 3Fh, 00h   |
| 10–31 | Reserved                    | 00h, 00h |
| 32–43 | Device manufacturer         | 4Dh, 49h, 43h, 52h, 4Fh, 4Eh, 20h, 20h, 20h, 20h, 20h, 20h   |



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Parameter Page Data Structure Tables

**Table 11: Parameter Page Data Structure (Continued)**

| Byte  | Description                            | Value   |
|-------|--|---|
| 44–63 | Device model                           | MT29F4G08ABBD4H4 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 30h, 38h, 41h, 42h, 42h, 44h, 41h, 48h, 34h, 20h, 20h, 20h, 20h |
|       |  | MT29F4G08ABBD4HC 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 30h, 38h, 41h, 42h, 42h, 44h, 41h, 48h, 43h, 20h, 20h, 20h, 20h |
|       |  | MT29F4G16ABBD4HC 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 31h, 36h, 41h, 42h, 42h, 44h, 41h, 48h, 43h, 20h, 20h, 20h, 20h |
|       |  | MT29F4G16ABBD4H4 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 31h, 36h, 41h, 42h, 42h, 44h, 41h, 48h, 34h, 20h, 20h, 20h, 20h |
|       |  | MT29F8G08ADB4H4 4Dh, 54h, 32h, 39h, 46h, 38h, 47h, 30h, 38h, 41h, 44h, 42h, 44h, 41h, 48h, 34h, 20h, 20h, 20h, 20h  |
|       |  | MT29F8G16ADB4H4 4Dh, 54h, 32h, 39h, 46h, 38h, 47h, 31h, 36h, 41h, 44h, 42h, 44h, 41h, 48h, 34h, 20h, 20h, 20h, 20h  |
|       |  | MT29F4G08ABAD4WP 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 30h, 38h, 41h, 42h, 41h, 44h, 41h, 57h, 50h, 20h, 20h, 20h, 20  |
|       |  | MT29F4G08ABAD4H4 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 30h, 38h, 41h, 42h, 41h, 44h, 41h, 48h, 34h, 20h, 20h, 20h, 20h |
|       |  | MT29F4G16ABAD4WP 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 31h, 36h, 41h, 42h, 41h, 44h, 41h, 57h, 50h, 20h, 20h, 20h, 20h |
|       |  | MT29F4G16ABAD4H4 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 31h, 36h, 41h, 42h, 41h, 44h, 41h, 48h, 34h, 20h, 20h, 20h, 20h |
|       |  | MT29F8G08ADAD4H4 4Dh, 54h, 32h, 39h, 46h, 38h, 47h, 30h, 38h, 41h, 44h, 41h, 44h, 41h, 48h, 34h, 20h, 20h, 20h, 20h |
|       |  | MT29F8G16ADAD4H4 4Dh, 54h, 32h, 39h, 46h, 38h, 47h, 31h, 36h, 41h, 44h, 41h, 44h, 41h, 48h, 34h, 20h, 20h, 20h, 20h |
| 64    | Manufacturer ID                        | 2Ch   |
| 65–66 | Date code                              | 00h, 00h  |
| 67–79 | Reserved                               | 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h  |
| 80–83 | Number of data bytes per page          | 00h, 08h, 00h, 00h  |
| 84–85 | Number of spare bytes per page         | 40h, 00h  |
| 86–89 | Number of data bytes per partial page  | 00h, 02h, 00h, 00h  |
| 90–91 | Number of spare bytes per partial page | 10h, 00h  |
| 92–95 | Number of pages per block              | 40h, 00h, 00h, 00h  |
| 96–99 | Number of blocks per unit              | 00h, 10h, 00h, 00h  |



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Parameter Page Data Structure Tables

**Table 11: Parameter Page Data Structure (Continued)**

| Byte    | Description                                    | Value  |
|---------|--|--|
| 100     | Number of logical units                        | MT29F4G08ABBD4H4   |
|         |  | MT29F4G08ABBD4HC   |
|         |  | MT29F4G16ABBD4HC   |
|         |  | MT29F4G16ABBD4H4   |
|         |  | MT29F8G08ADBDAH4   |
|         |  | MT29F8G16ADBDAH4   |
|         |  | MT29F4G08ABADAWP   |
|         |  | MT29F4G08ABADAH4   |
|         |  | MT29F4G16ABADAWP   |
|         |  | MT29F4G16ABADAH4   |
|         |  | MT29F8G08ADADAH4   |
|         |  | MT29F8G16ADADAH4   |
| 101     | Number of address cycles                       | 23h  |
| 102     | Number of bits per cell                        | 01h  |
| 103–104 | Bad blocks maximum per unit                    | 50h, 00h   |
| 105–106 | –  | –  |
| 107     | Guaranteed valid blocks at beginning of target | 01h  |
| 108–109 | Block endurance for guaranteed valid blocks    | 00h, 00h   |
| 110     | Number of programs per page                    | 04h  |
| 111     | Partial programming attributes                 | 00h  |
| 112     | Number of bits ECC bits                        | 04h  |
| 113     | Number of interleaved address bits             | 01h  |
| 114     | Interleaved operation attributes               | 0Eh  |
| 115–127 | Reserved                                       | 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h |
| 128     | I/O pin capacitance                            | MT29F4G08ABBD4H4   |
|         |  | MT29F4G08ABBD4HC   |
|         |  | MT29F4G16ABBD4HC   |
|         |  | MT29F4G16ABBD4H4   |
|         |  | MT29F8G08ADBDAH4   |
|         |  | MT29F8G16ADBDAH4   |
|         |  | MT29F4G08ABADAWP   |
|         |  | MT29F4G08ABADAH4   |
|         |  | MT29F4G16ABADAWP   |
|         |  | MT29F4G16ABADAH4   |
|         |  | MT29F8G08ADADAH4   |
|         |  | MT29F8G16ADADAH4   |

## 4Gb, 8Gb: x8, x16 NAND Flash Memory Parameter Page Data Structure Tables

### Table 11: Parameter Page Data Structure (Continued)

| Byte    | Description                               |                   | Value  |
|---------|---|-------------------|--|
| 129–130 | Timing mode support                       | MT29F4G08ABBD AH4 | 1Fh, 00h   |
|         |   | MT29F4G08ABBDAHC  | 1Fh, 00h   |
|         |   | MT29F4G16ABBD AHC | 1Fh, 00h   |
|         |   | MT29F4G16ABBD AH4 | 1Fh, 00h   |
|         |   | MT29F8G08ADBDAH4  | 1Fh, 00h   |
|         |   | MT29F8G16ADBDAH4  | 1Fh, 00h   |
|         |   | MT29F4G08ABADAWP  | 3Fh, 00h   |
|         |   | MT29F4G08ABADAH4  | 3Fh, 00h   |
|         |   | MT29F4G16ABADAWP  | 3Fh, 00h   |
|         |   | MT29F4G16ABADAH4  | 3Fh, 00h   |
|         |   | MT29F8G08ADADAH4  | 3Fh, 00h   |
|         |   | MT29F8G16ADADAH4  | 3Fh, 00h   |
| 131–132 | Program cache timing<br>mode support      | MT29F4G08ABBD AH4 | 1Fh, 00h   |
|         |   | MT29F4G08ABBDAHC  | 1Fh, 00h   |
|         |   | MT29F4G16ABBD AHC | 1Fh, 00h   |
|         |   | MT29F4G16ABBD AH4 | 1Fh, 00h   |
|         |   | MT29F8G08ADBDAH4  | 1Fh, 00h   |
|         |   | MT29F8G16ADBDAH4  | 1Fh, 00h   |
|         |   | MT29F4G08ABADAWP  | 3Fh, 00h   |
|         |   | MT29F4G08ABADAH4  | 3Fh, 00h   |
|         |   | MT29F4G16ABADAWP  | 3Fh, 00h   |
|         |   | MT29F4G16ABADAH4  | 3Fh, 00h   |
|         |   | MT29F8G08ADADAH4  | 3Fh, 00h   |
|         |   | MT29F8G16ADADAH4  | 3Fh, 00h   |
| 133–134 | t <sup>PROG</sup> (MAX) page program time |                   | 58h, 02h   |
| 135–136 | t <sup>BERS</sup> (MAX) block erase time  |                   | B8h, 0Bh   |
| 137–138 | t <sup>R</sup> (MAX) page read time       |                   | 19h, 00h   |
| 139–140 | t <sup>CCs</sup> (MIN)                    |                   | 64h, 00h   |
| 141–163 | Reserved                                  |                   | 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,<br>00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h  |
| 164–165 | Vendor-specific revision number           |                   | 01h, 00h   |
| 166–253 | Vendor-specific                           |                   | 01h, 00h, 00h, 02h, 04h, 80h, 01h, 81h, 04h, 01h, 02h, 01h,<br>0Ah, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,<br>00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,<br>00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,<br>00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,<br>00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,<br>00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,<br>00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,<br>00h, 00h, 00h, 00h |
| 254–255 | Integrity CRC                             |                   | Set at test  |

**4Gb, 8Gb: x8, x16 NAND Flash Memory  
Parameter Page Data Structure Tables****Table 11: Parameter Page Data Structure (Continued)**

| Byte    | Description                          | Value |
|---------|--------------------------------------|-------|
| 256–511 | Value of bytes 0–255                 |       |
| 512–767 | Value of bytes 0–255                 |       |
| 768+    | Additional redundant parameter pages |       |



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Bare Die Parameter Page Data Structure Tables

### Bare Die Parameter Page Data Structure Tables

**Table 12: Parameter Page Data Structure**

| Byte  | Description                            | Value  |
|-------|--|--|
| 0–3   | Parameter page signature               | 4Fh, 4Eh, 46h, 49h   |
| 4–5   | Revision number                        | 02h, 00h   |
| 6–7   | Features supported                     | MT29F4G08ABBDA3W 18h, 00h  |
|       |  | MT29F4G16ABBDA3W 19h, 00h  |
|       |  | MT29F8G08ADBDA3W 1Ah, 00h  |
|       |  | MT29F8G16ADBDA3W 1Bh, 00h  |
|       |  | MT29F4G08ABADA3W 18h, 00h  |
|       |  | MT29F4G16ABADA3W 19h, 00h  |
|       |  | MT29F8G08ADADA3W 1Ah, 00h  |
|       |  | MT29F8G16ADADA3W 1Bh, 00h  |
| 8–9   | Optional commands supported            | 3Fh, 00h   |
| 10–31 | Reserved                               | 00h, 00h |
| 32–43 | Device manufacturer                    | 4Dh, 49h, 43h, 52h, 4Fh, 4Eh, 20h, 20h, 20h, 20h, 20h, 20h   |
| 44–63 | Device model                           | MT29F4G08ABBDA3W 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 30h, 38h, 41h, 42h, 42h, 44h, 41h, 33h, 57h, 20h, 20h, 20h, 20h                                  |
|       |  | MT29F4G16ABBDA3W 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 31h, 36h, 41h, 42h, 42h, 44h, 41h, 33h, 57h, 20h, 20h, 20h, 20h                                  |
|       |  | MT29F8G08ADBDA3W 4Dh, 54h, 32h, 39h, 46h, 38h, 47h, 30h, 38h, 41h, 44h, 42h, 44h, 41h, 33h, 57h, 20h, 20h, 20h, 20h                                  |
|       |  | MT29F8G16ADBDA3W 4Dh, 54h, 32h, 39h, 46h, 38h, 47h, 31h, 36h, 41h, 44h, 42h, 44h, 41h, 33h, 57h, 20h, 20h, 20h, 20h                                  |
|       |  | MT29F4G08ABADA3W 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 30h, 38h, 41h, 42h, 41h, 44h, 41h, 33h, 57h, 20h, 20h, 20h, 20h                                  |
|       |  | MT29F4G16ABADA3W 4Dh, 54h, 32h, 39h, 46h, 34h, 47h, 31h, 36h, 41h, 42h, 41h, 44h, 41h, 33h, 57h, 20h, 20h, 20h, 20h                                  |
|       |  | MT29F8G08ADADA3W 4Dh, 54h, 32h, 39h, 46h, 38h, 47h, 30h, 38h, 41h, 44h, 41h, 44h, 41h, 33h, 57h, 20h, 20h, 20h, 20h                                  |
|       |  | MT29F8G16ADADA3W 4Dh, 54h, 32h, 39h, 46h, 38h, 47h, 31h, 36h, 41h, 44h, 41h, 44h, 41h, 33h, 57h, 20h, 20h, 20h, 20h                                  |
| 64    | Manufacturer ID                        | 2Ch  |
| 65–66 | Date code                              | 00h, 00h   |
| 67–79 | Reserved                               | 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h   |
| 80–83 | Number of data bytes per page          | 00h, 08h, 00h, 00h   |
| 84–85 | Number of spare bytes per page         | 40h, 00h   |
| 86–89 | Number of data bytes per partial page  | 00h, 02h, 00h, 00h   |
| 90–91 | Number of spare bytes per partial page | 10h, 00h   |
| 92–95 | Number of pages per block              | 40h, 00h, 00h, 00h   |





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Bare Die Parameter Page Data Structure Tables

**Table 12: Parameter Page Data Structure (Continued)**

| Byte    | Description                                    | Value  |
|---------|--|--|
| 96–99   | Number of blocks per unit                      | 00h, 10h, 00h, 00h   |
| 100     | Number of logical units                        | MT29F4G08ABBD3W 01h  |
|         |  | MT29F4G16ABBD3W 01h  |
|         |  | MT29F8G08ADBDA3W 02h                                       |
|         |  | MT29F8G16ADBDA3W 02h                                       |
|         |  | MT29F4G08ABADA3W 01h                                       |
|         |  | MT29F4G16ABADA3W 01h                                       |
|         |  | MT29F8G08ADADA3W 02h                                       |
|         |  | MT29F8G16ADADA3W 02h                                       |
| 101     | Number of address cycles                       | 23h  |
| 102     | Number of bits per cell                        | 01h  |
| 103–104 | Bad blocks maximum per unit                    | 50h, 00h   |
| 105–106 | Block endurance                                | 01h, 05h   |
| 107     | Guaranteed valid blocks at beginning of target | 01h  |
| 108–109 | Block endurance for guaranteed valid blocks    | 00h, 00h   |
| 110     | Number of programs per page                    | 04h  |
| 111     | Partial programming attributes                 | 00h  |
| 112     | Number of bits ECC bits                        | 04h  |
| 113     | Number of interleaved address bits             | 01h  |
| 114     | Interleaved operation attributes               | 0Eh  |
| 115–127 | Reserved                                       | 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h |
| 128     | I/O pin capacitance                            | MT29F4G08ABBD3W 0Ah  |
|         |  | MT29F4G16ABBD3W 0Ah  |
|         |  | MT29F8G08ADBDA3W 14h                                       |
|         |  | MT29F8G16ADBDA3W 14h                                       |
|         |  | MT29F4G08ABADA3W 0Ah                                       |
|         |  | MT29F4G16ABADA3W 0Ah                                       |
|         |  | MT29F8G08ADADA3W 14h                                       |
|         |  | MT29F8G16ADADA3W 14h                                       |
| 129–130 | Timing mode support                            | MT29F4G08ABBD3W 1Fh, 00h                                   |
|         |  | MT29F4G16ABBD3W 1Fh, 00h                                   |
|         |  | MT29F8G08ADBDA3W 1Fh, 00h                                  |
|         |  | MT29F8G16ADBDA3W 1Fh, 00h                                  |
|         |  | MT29F4G08ABADA3W 3Fh, 00h                                  |
|         |  | MT29F4G16ABADA3W 3Fh, 00h                                  |
|         |  | MT29F8G08ADADA3W 3Fh, 00h                                  |
|         |  | MT29F8G16ADADA3W 3Fh, 00h                                  |





## READ UNIQUE ID (EDh)

The READ UNIQUE ID (EDh) command is used to read a unique identifier programmed into the target. This command is accepted by the target only when all die (LUNs) on the target are idle.

Writing EDh to the command register puts the target in read unique ID mode. The target stays in this mode until another valid command is issued.

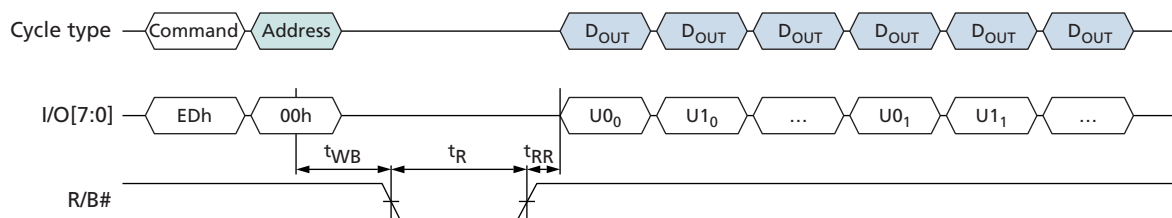
When the EDh command is followed by an 00h address cycle, the target goes busy for  $t_R$ . If the READ STATUS (70h) command is used to monitor for command completion, the READ MODE (00h) command must be used to re-enable data output mode.

After  $t_R$  completes, the host enables data output mode to read the unique ID. When the asynchronous interface is active, one data byte is output per RE# toggle.

Sixteen copies of the unique ID data are stored in the device. Each copy is 32 bytes. The first 16 bytes of a 32-byte copy are unique data, and the second 16 bytes are the complement of the first 16 bytes. The host should XOR the first 16 bytes with the second 16 bytes. If the result is 16 bytes of FFh, then that copy of the unique ID data is correct. In the event that a non-FFh result is returned, the host can repeat the XOR operation on a subsequent copy of the unique ID data. If desired, the RANDOM DATA READ (05h-E0h) command can be used to change the data output location.

The upper eight I/Os on a x16 device are not used and are a “Don’t Care” for x16 devices.

**Figure 29: READ UNIQUE ID (EDh) Operation**





## Feature Operations

The SET FEATURES (EFh) and GET FEATURES (EEh) commands are used to modify the target's default power-on behavior. These commands use a one-byte feature address to determine which subfeature parameters will be read or modified. Each feature address (in the 00h to FFh range) is defined in below. The SET FEATURES (EFh) command writes subfeature parameters (P1–P4) to the specified feature address. The GET FEATURES command reads the subfeature parameters (P1–P4) at the specified feature address.

When a feature is set, by default it remains active until the device is power cycled. It is volatile. Unless otherwise specified in the features table, once a device is set it remains set, even if a RESET (FFh) command is issued. GET/SET FEATURES commands can be used after required RESET to enable features before system BOOT ROM process.

Internal ECC can be enabled/disabled using SET FEATURES (EFh). The SET FEATURES command (EFh), followed by address 90h, followed by four data bytes (only the first data byte is used) will enable/disable internal ECC.

The sequence to enable internal ECC with SET FEATURES is EFh(cmd)-90h(addr)-08h(data)-00h(data)-00h(data)-00h(data)-wait(<sup>t</sup>FEAT).

The sequence to disable internal ECC with SET FEATURES is EFh(cmd)-90h(addr)-00h(data)-00h(data)-00h(data)-00h(data)-wait(<sup>t</sup>FEAT). The GET FEATURES command is EEh.

**Table 13: Feature Address Definitions**

| Feature Address | Definition                          |
|-----------------|-------------------------------------|
| 00h             | Reserved                            |
| 01h             | Timing mode                         |
| 02h–7Fh         | Reserved                            |
| 80h             | Programmable output drive strength  |
| 81h             | Programmable RB# pull-down strength |
| 82h–FFh         | Reserved                            |
| 90h             | Array operation mode                |


**Table 14: Feature Address 90h – Array Operation Mode**

| Subfeature Parameter  | Options        | I/O7         | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0 | Value | Notes |
|-----------------------|----------------|--------------|------|------|------|------|------|------|------|-------|-------|
| P1                    |                |              |      |      |      |      |      |      |      |       |       |
| Operation mode option | Normal         | Reserved (0) |      |      |      |      |      |      | 0    | 00h   | 1     |
|                       | OTP operation  | Reserved (0) |      |      |      |      |      |      | 1    | 01h   |       |
|                       | OTP protection | Reserved (0) |      |      |      |      |      | 1    | 1    | 03h   |       |
|                       | Disable ECC    | Reserved (0) |      |      |      | 0    | 0    | 0    | 0    | 00h   | 1     |
|                       | Enable ECC     | Reserved (0) |      |      |      | 1    | 0    | 0    | 0    | 08h   | 1     |
| P2                    |                |              |      |      |      |      |      |      |      |       |       |
| Reserved              |                | Reserved (0) |      |      |      |      |      |      |      | 00h   |       |
| P3                    |                |              |      |      |      |      |      |      |      |       |       |
| Reserved              |                | Reserved (0) |      |      |      |      |      |      |      | 00h   |       |
| P4                    |                |              |      |      |      |      |      |      |      |       |       |
| Reserved              |                | Reserved (0) |      |      |      |      |      |      |      | 00h   |       |

Note: 1. These bits are reset to 00h on power cycle.

## SET FEATURES (EFh)

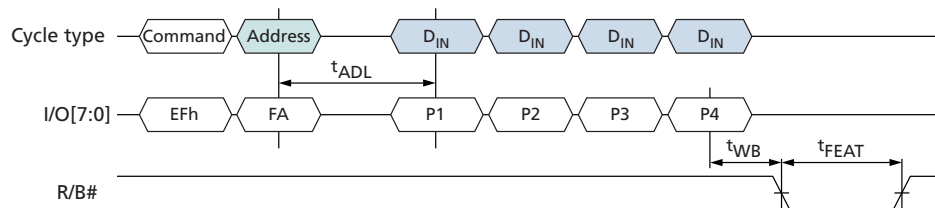
The SET FEATURES (EFh) command writes the subfeature parameters (P1–P4) to the specified feature address to enable or disable target-specific features. This command is accepted by the target only when all die (LUNs) on the target are idle.

Writing EFh to the command register puts the target in the set features mode. The target stays in this mode until another command is issued.

The EFh command is followed by a valid feature address. The host waits for  $t_{ADL}$  before the subfeature parameters are input. When the asynchronous interface is active, one subfeature parameter is latched per rising edge of WE#.

After all four subfeature parameters are input, the target goes busy for  $t_{FEAT}$ . The READ STATUS (70h) command can be used to monitor for command completion.

Feature address 01h (timing mode) operation is unique. If SET FEATURES is used to modify the interface type, the target will be busy for  $t_{ITC}$ .

**Figure 30: SET FEATURES (EFh) Operation**




## GET FEATURES (EEh)

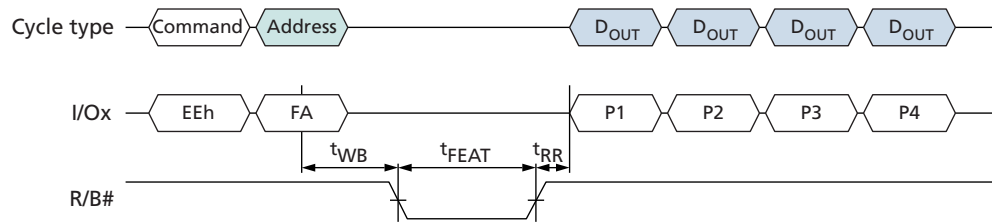
The GET FEATURES (EEh) command reads the subfeature parameters (P1–P4) from the specified feature address. This command is accepted by the target only when all die (LUNs) on the target are idle.

Writing EEh to the command register puts the target in get features mode. The target stays in this mode until another valid command is issued.

When the EEh command is followed by a feature address, the target goes busy for  $t_{FEAT}$ . If the READ STATUS (70h) command is used to monitor for command completion, the READ MODE (00h) command must be used to re-enable data output mode.

After  $t_{FEAT}$  completes, the host enables data output mode to read the subfeature parameters.

**Figure 31: GET FEATURES (EEh) Operation**





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Feature Operations

**Table 15: Feature Addresses 01h: Timing Mode**

| Subfeature Parameter | Options          | I/O7 | I/O6 | I/O5 | I/O4         | I/O3 | I/O2 | I/O1 | I/O0 | Value | Notes |
|----------------------|------------------|------|------|------|--------------|------|------|------|------|-------|-------|
| <b>P1</b>            |                  |      |      |      |              |      |      |      |      |       |       |
| Timing mode          | Mode 0 (default) |      |      |      | Reserved (0) |      | 0    | 0    | 0    | 00h   | 1, 2  |
|                      | Mode 1           |      |      |      | Reserved (0) |      | 0    | 0    | 1    | 01h   | 2     |
|                      | Mode 2           |      |      |      | Reserved (0) |      | 0    | 1    | 0    | 01h   | 2     |
|                      | Mode 3           |      |      |      | Reserved (0) |      | 0    | 1    | 1    | 01h   | 2     |
|                      | Mode 4           |      |      |      | Reserved (0) |      | 1    | 0    | 0    | 01h   | 2     |
|                      | Mode 5           |      |      |      | Reserved (0) |      | 1    | 0    | 1    | 01h   | 3     |
| <b>P2</b>            |                  |      |      |      |              |      |      |      |      |       |       |
|                      |                  |      |      |      | Reserved (0) |      |      |      |      | 00h   |       |
| <b>P3</b>            |                  |      |      |      |              |      |      |      |      |       |       |
|                      |                  |      |      |      | Reserved (0) |      |      |      |      | 00h   |       |
| <b>P4</b>            |                  |      |      |      |              |      |      |      |      |       |       |
|                      |                  |      |      |      | Reserved (0) |      |      |      |      | 00h   |       |

- Notes:
1. The timing mode feature address is used to change the default timing mode. The timing mode should be selected to indicate the maximum speed at which the device will receive commands, addresses, and data cycles. The five supported settings for the timing mode are shown. The default timing mode is mode 0. The device returns to mode 0 when the device is power cycled. Supported timing modes are reported in the parameter page.
  2. Supported for both 1.8V and 3.3V.
  3. Supported for 3.3V only.


**Table 16: Feature Addresses 80h: Programmable I/O Drive Strength**

| Subfeature Parameter | Options        | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0 | Value | Notes |
|----------------------|----------------|------|------|------|------|------|------|------|------|-------|-------|
| <b>P1</b>            |                |      |      |      |      |      |      |      |      |       |       |
| I/O drive strength   | Full (default) |      |      |      |      |      |      | 0    | 0    | 00h   | 1     |
|                      | Three-quarters |      |      |      |      |      |      | 0    | 1    | 01h   |       |
|                      | One-half       |      |      |      |      |      |      | 1    | 0    | 02h   |       |
|                      | One-quarter    |      |      |      |      |      |      | 1    | 1    | 03h   |       |
| <b>P2</b>            |                |      |      |      |      |      |      |      |      |       |       |
|                      |                |      |      |      |      |      |      |      |      | 00h   |       |
| <b>P3</b>            |                |      |      |      |      |      |      |      |      |       |       |
|                      |                |      |      |      |      |      |      |      |      | 00h   |       |
| <b>P4</b>            |                |      |      |      |      |      |      |      |      |       |       |
|                      |                |      |      |      |      |      |      |      |      | 00h   |       |

Note: 1. The programmable drive strength feature address is used to change the default I/O drive strength. Drive strength should be selected based on expected loading of the memory bus. This table shows the four supported output drive strength settings. The default drive strength is full strength. The device returns to the default drive strength mode when the device is power cycled. AC timing parameters may need to be relaxed if I/O drive strength is not set to full.

**Table 17: Feature Addresses 81h: Programmable R/B# Pull-Down Strength**

| Subfeature Parameter    | Options        | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0 | Value | Notes |
|-------------------------|----------------|------|------|------|------|------|------|------|------|-------|-------|
| <b>P1</b>               |                |      |      |      |      |      |      |      |      |       |       |
| R/B# pull-down strength | Full (default) |      |      |      |      |      |      | 0    | 0    | 00h   | 1     |
|                         | Three-quarters |      |      |      |      |      |      | 0    | 1    | 01h   |       |
|                         | One-half       |      |      |      |      |      |      | 1    | 0    | 02h   |       |
|                         | One-quarter    |      |      |      |      |      |      | 1    | 1    | 03h   |       |
| <b>P2</b>               |                |      |      |      |      |      |      |      |      |       |       |
|                         |                |      |      |      |      |      |      |      |      | 00h   |       |
| <b>P3</b>               |                |      |      |      |      |      |      |      |      |       |       |
|                         |                |      |      |      |      |      |      |      |      | 00h   |       |
| <b>P4</b>               |                |      |      |      |      |      |      |      |      |       |       |
|                         |                |      |      |      |      |      |      |      |      | 00h   |       |

Note: 1. This feature address is used to change the default R/B# pull-down strength. Its strength should be selected based on the expected loading of R/B#. Full strength is the default, power-on value.





## Status Operations

Each die (LUN) provides its status independently of other die (LUNs) on the same target through its 8-bit status register.

After the READ STATUS (70h) or READ STATUS ENHANCED (78h) command is issued, status register output is enabled. The contents of the status register are returned on I/O[7:0] for each data output request.

When the asynchronous interface is active and status register output is enabled, changes in the status register are seen on I/O[7:0] as long as CE# and RE# are LOW; it is not necessary to toggle RE# to see the status register update.

While monitoring the status register to determine when a data transfer from the Flash array to the data register (<sup>1</sup>R) is complete, the host must issue the READ MODE (00h) command to disable the status register and enable data output (see Read Operations).

The READ STATUS (70h) command returns the status of the most recently selected die (LUN). To prevent data contention during or following an interleaved die (multi-LUN) operation, the host must enable only one die (LUN) for status output by using the READ STATUS ENHANCED (78h) command (see Interleaved Die (Multi-LUN) Operations).

With internal ECC enabled, a READ STATUS command is required after completion of the data transfer (<sup>1</sup>R\_ECC) to determine whether an uncorrectable read error occurred.

**Table 18: Status Register Definition**

| SR Bit | Program Page  | Program Page Cache Mode | Page Read                        | Page Read Cache Mode   | Block Erase   | Description  |
|--------|---------------|-------------------------|----------------------------------|------------------------|---------------|--|
| 7      | Write protect | Write protect           | Write protect                    | Write protect          | Write protect | 0 = Protected<br>1 = Not protected   |
| 6      | RDY           | RDY <sup>1</sup> cache  | RDY                              | RDY <sup>1</sup> cache | RDY           | 0 = Busy<br>1 = Ready  |
| 5      | ARDY          | ARDY <sup>2</sup>       | ARDY                             | ARDY <sup>2</sup>      | ARDY          | Don't Care   |
| 4      | –             | –                       | –                                | –                      | –             | Don't Care   |
| 3      | –             | –                       | Rewrite recommended <sup>3</sup> | –                      | –             | 0 = Normal or uncorrectable<br>1 = Rewrite recommended                       |
| 2      | –             | –                       | –                                | –                      | –             | Don't Care   |
| 1      | Reserved      | FAILC (N - 1)           | Reserved                         | –                      | –             | Don't Care   |
| 0      | FAIL          | FAIL (N)                | FAIL <sup>4</sup>                | –                      | FAIL          | 0 = Successful PROGRAM/<br>ERASE/READ<br>1 = Error in PROGRAM/<br>ERASE/READ |

- Notes:
1. Status register bit 6 is 1 when the cache is ready to accept new data. R/B# follows bit 6.
  2. Status register bit 5 is 0 during the actual programming operation. If cache mode is used, this bit will be 1 when all internal operations are complete.
  3. A status register bit defined as Rewrite Recommended signifies that the page includes a certain number of READ errors per sector (512B (main) + 4B (spare) + 8B (parity)). A rewrite of this page is recommended. (Up to a 4-bit error has been corrected if internal ECC was enabled.)
  4. A status register bit defined as FAIL signifies that an uncorrectable READ error has occurred.



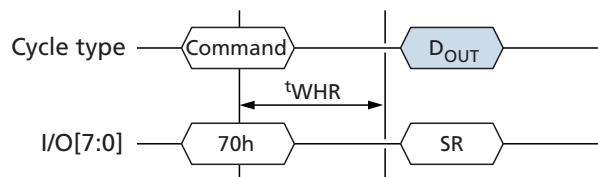
## READ STATUS (70h)

The READ STATUS (70h) command returns the status of the last-selected die (LUN) on a target. This command is accepted by the last-selected die (LUN) even when it is busy (RDY = 0).

If there is only one die (LUN) per target, the READ STATUS (70h) command can be used to return status following any NAND command.

In devices that have more than one die (LUN) per target, during and following interleaved die (multi-LUN) operations, the READ STATUS ENHANCED (78h) command must be used to select the die (LUN) that should report status. In this situation, using the READ STATUS (70h) command will result in bus contention, as two or more die (LUNs) could respond until the next operation is issued. The READ STATUS (70h) command can be used following all single-die (LUN) operations.

**Figure 32: READ STATUS (70h) Operation**



## READ STATUS ENHANCED (78h)

The READ STATUS ENHANCED (78h) command returns the status of the addressed die (LUN) on a target even when it is busy (RDY = 0). This command is accepted by all die (LUNs), even when they are BUSY (RDY = 0).

Writing 78h to the command register, followed by three row address cycles containing the page, block, and LUN addresses, puts the selected die (LUN) into read status mode. The selected die (LUN) stays in this mode until another valid command is issued. Die (LUNs) that are not addressed are deselected to avoid bus contention.

The selected LUN's status is returned when the host requests data output. The RDY and ARDY bits of the status register are shared for all planes on the selected die (LUN). The FAILC and FAIL bits are specific to the plane specified in the row address.

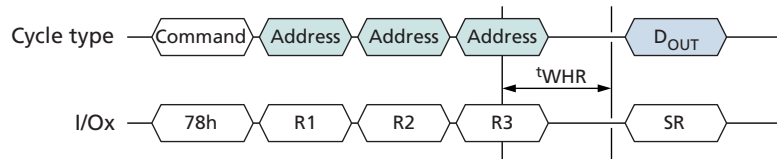
The READ STATUS ENHANCED (78h) command also enables the selected die (LUN) for data output. To begin data output following a READ-series operation after the selected die (LUN) is ready (RDY = 1), issue the READ MODE (00h) command, then begin data output. If the host needs to change the cache register that will output data, use the RANDOM DATA READ TWO-PLANE (06h-E0h) command after the die (LUN) is ready.

Use of the READ STATUS ENHANCED (78h) command is prohibited during the power-on RESET (FFh) command and when OTP mode is enabled. It is also prohibited following some of the other reset, identification, and configuration operations. See individual operations for specific details.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Status Operations

**Figure 33: READ STATUS ENHANCED (78h) Operation**





## Column Address Operations

The column address operations affect how data is input to and output from the cache registers within the selected die (LUNs). These features provide host flexibility for managing data, especially when the host internal buffer is smaller than the number of data bytes or words in the cache register.

When the asynchronous interface is active, column address operations can address any byte in the selected cache register.

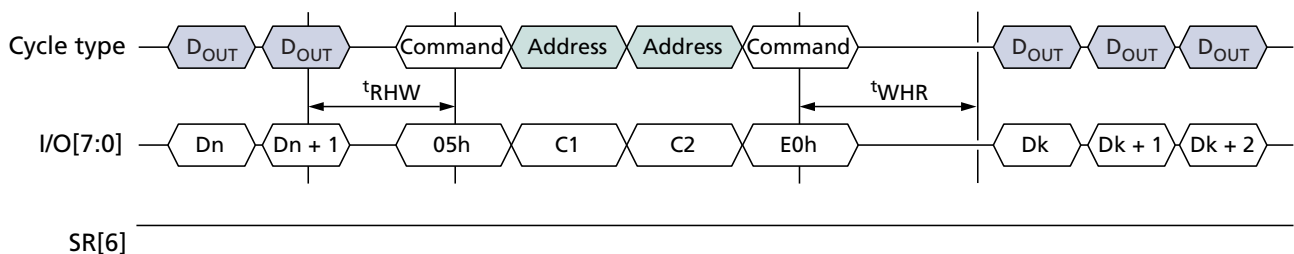
### RANDOM DATA READ (05h-E0h)

The RANDOM DATA READ (05h-E0h) command changes the column address of the selected cache register and enables data output from the last selected die (LUN). This command is accepted by the selected die (LUN) when it is ready (RDY = 1; ARDY = 1). It is also accepted by the selected die (LUN) during CACHE READ operations (RDY = 1; ARDY = 0).

Writing 05h to the command register, followed by two column address cycles containing the column address, followed by the E0h command, puts the selected die (LUN) into data output mode. After the E0h command cycle is issued, the host must wait at least  $t_{WHR}$  before requesting data output. The selected die (LUN) stays in data output mode until another valid command is issued.

In devices with more than one die (LUN) per target, during and following interleaved die (multi-LUN) operations, the READ STATUS ENHANCED (78h) command must be issued prior to issuing the RANDOM DATA READ (05h-E0h). In this situation, using the RANDOM DATA READ (05h-E0h) command without the READ STATUS ENHANCED (78h) command will result in bus contention because two or more die (LUNs) could output data.

**Figure 34: RANDOM DATA READ (05h-E0h) Operation**





## RANDOM DATA READ TWO-PLANE (06h-E0h)

The RANDOM DATA READ TWO-PLANE (06h-E0h) command enables data output on the addressed die's (LUN's) cache register at the specified column address. This command is accepted by a die (LUN) when it is ready (RDY = 1; ARDY = 1).

Writing 06h to the command register, followed by two column address cycles and three row address cycles, followed by E0h, enables data output mode on the address LUN's cache register at the specified column address. After the E0h command cycle is issued, the host must wait at least  $t_{WHR}$  before requesting data output. The selected die (LUN) stays in data output mode until another valid command is issued.

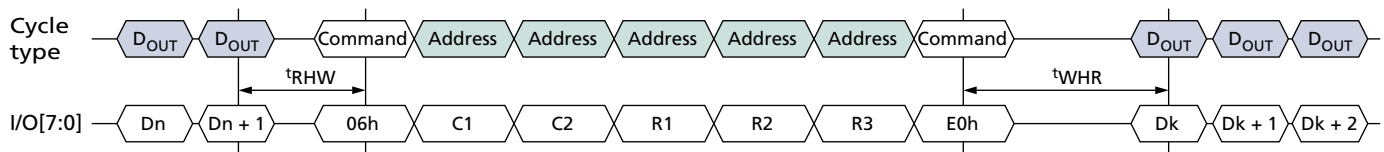
Following a two-plane read page operation, the RANDOM DATA READ TWO-PLANE (06h-E0h) command is used to select the cache register to be enabled for data output. After data output is complete on the selected plane, the command can be issued again to begin data output on another plane.

In devices with more than one die (LUN) per target, after all of the die (LUNs) on the target are ready (RDY = 1), the RANDOM DATA READ TWO-PLANE (06h-E0h) command can be used following an interleaved die (multi-LUN) read operation. Die (LUNs) that are not addressed are deselected to avoid bus contention.

In devices with more than one die (LUN) per target, during interleaved die (multi-LUN) operations where more than one or more die (LUNs) are busy (RDY = 1; ARDY = 0 or RDY = 0; ARDY = 0), the READ STATUS ENHANCED (78h) command must be issued to the die (LUN) to be selected prior to issuing the RANDOM DATA READ TWO-PLANE (06h-E0h). In this situation, using the RANDOM DATA READ TWO-PLANE (06h-E0h) command without the READ STATUS ENHANCED (78h) command will result in bus contention, as two or more die (LUNs) could output data.

If there is a need to update the column address without selecting a new cache register or LUN, the RANDOM DATA READ (05h-E0h) command can be used instead.

**Figure 35: RANDOM DATA READ TWO-PLANE (06h-E0h) Operation**





## RANDOM DATA INPUT (85h)

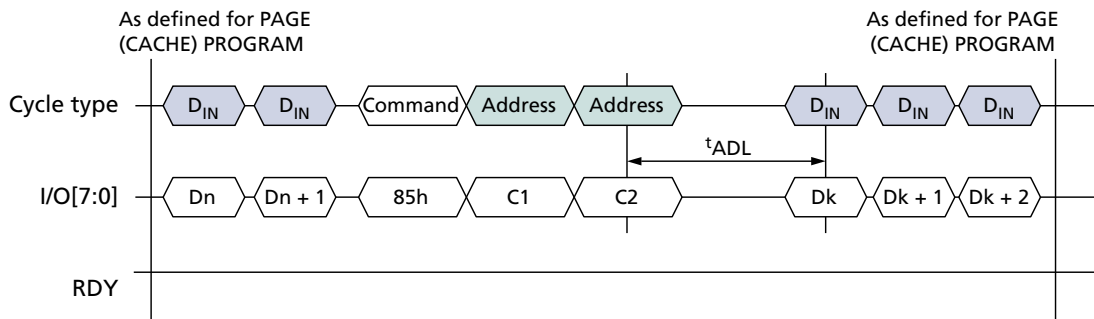
The RANDOM DATA INPUT (85h) command changes the column address of the selected cache register and enables data input on the last-selected die (LUN). This command is accepted by the selected die (LUN) when it is ready (RDY = 1; ARDY = 1). It is also accepted by the selected die (LUN) during cache program operations (RDY = 1; ARDY = 0).

Writing 85h to the command register, followed by two column address cycles containing the column address, puts the selected die (LUN) into data input mode. After the second address cycle is issued, the host must wait at least  $t_{ADL}$  before inputting data. The selected die (LUN) stays in data input mode until another valid command is issued. Though data input mode is enabled, data input from the host is optional. Data input begins at the column address specified.

The RANDOM DATA INPUT (85h) command is allowed after the required address cycles are specified, but prior to the final command cycle (10h, 11h, 15h) of the following commands while data input is permitted: PROGRAM PAGE (80h-10h), PROGRAM PAGE CACHE (80h-15h), PROGRAM FOR INTERNAL DATA MOVE (85h-10h), and PROGRAM FOR TWO-PLANE INTERNAL DATA MOVE (85h-11h).

In devices that have more than one die (LUN) per target, the RANDOM DATA INPUT (85h) command can be used with other commands that support interleaved die (multi-LUN) operations.

**Figure 36: RANDOM DATA INPUT (85h) Operation**





## PROGRAM FOR INTERNAL DATA INPUT (85h)

The PROGRAM FOR INTERNAL DATA INPUT (85h) command changes the row address (block and page) where the cache register contents will be programmed in the NAND Flash array. It also changes the column address of the selected cache register and enables data input on the specified die (LUN). This command is accepted by the selected die (LUN) when it is ready (RDY = 1; ARDY = 1). It is also accepted by the selected die (LUN) during cache programming operations (RDY = 1; ARDY = 0).

Write 85h to the command register. Then write two column address cycles and three row address cycles. This updates the page and block destination of the selected device for the addressed LUN and puts the cache register into data input mode. After the fifth address cycle is issued the host must wait at least <sup>t</sup>ADL before inputting data. The selected LUN stays in data input mode until another valid command is issued. Though data input mode is enabled, data input from the host is optional. Data input begins at the column address specified.

The PROGRAM FOR INTERNAL DATA INPUT (85h) command is allowed after the required address cycles are specified, but prior to the final command cycle (10h, 11h, 15h) of the following commands while data input is permitted: PROGRAM PAGE (80h-10h), PROGRAM PAGE TWO-PLANE (80h-11h), PROGRAM PAGE CACHE (80h-15h), PROGRAM FOR INTERNAL DATA MOVE (85h-10h), and PROGRAM FOR TWO-PLANE INTERNAL DATA MOVE (85h-11h). When used with these commands, the LUN address and plane select bits are required to be identical to the LUN address and plane select bits originally specified.

The PROGRAM FOR INTERNAL DATA INPUT (85h) command enables the host to modify the original page and block address for the data in the cache register to a new page and block address.

In devices that have more than one die (LUN) per target, the PROGRAM FOR INTERNAL DATA INPUT (85h) command can be used with other commands that support interleaved die (multi-LUN) operations.

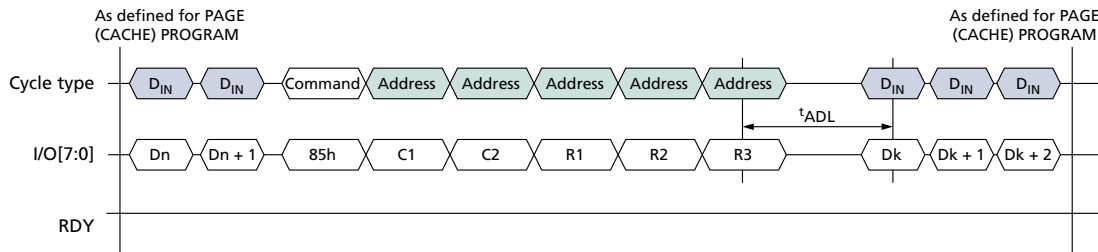
The PROGRAM FOR INTERNAL DATA INPUT (85h) command can be used with the RANDOM DATA READ (05h-E0h) or RANDOM DATA READ TWO-PLANE (06h-E0h) commands to read and modify cache register contents in small sections prior to programming cache register contents to the NAND Flash array. This capability can reduce the amount of buffer memory used in the host controller.

The RANDOM DATA INPUT (85h) command can be used during the PROGRAM FOR INTERNAL DATA MOVE command sequence to modify one or more bytes of the original data. First, data is copied into the cache register using the 00h-35h command sequence, then the RANDOM DATA INPUT (85h) command is written along with the address of the data to be modified next. New data is input on the external data pins. This copies the new data into the cache register.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Column Address Operations

**Figure 37: PROGRAM FOR INTERNAL DATA INPUT (85h) Operation**







## Read Operations

The READ PAGE (00h-30h) command, when issued by itself, reads one page from the NAND Flash array to its cache register and enables data output for that cache register.

During data output the following commands can be used to read and modify the data in the cache registers: RANDOM DATA READ (05h-E0h) and RANDOM DATA INPUT (85h).

### Read Cache Operations

To increase data throughput, the READ PAGE CACHE series (31h, 00h-31h) commands can be used to output data from the cache register while concurrently copying a page from the NAND Flash array to the data register.

To begin a read page cache sequence, begin by reading a page from the NAND Flash array to its corresponding cache register using the READ PAGE (00h-30h) command. R/B# goes LOW during <sup>t</sup>R and the selected die (LUN) is busy (RDY = 0, ARDY = 0). After <sup>t</sup>R (R/B# is HIGH and RDY = 1, ARDY = 1), issue either of these commands:

- READ PAGE CACHE SEQUENTIAL (31h) – copies the next sequential page from the NAND Flash array to the data register
- READ PAGE CACHE RANDOM (00h-31h) – copies the page specified in this command from the NAND Flash array to its corresponding data register

After the READ PAGE CACHE series (31h, 00h-31h) command has been issued, R/B# goes LOW on the target, and RDY = 0 and ARDY = 0 on the die (LUN) for <sup>t</sup>RCBSY while the next page begins copying data from the array to the data register. After <sup>t</sup>RCBSY, R/B# goes HIGH and the die's (LUN's) status register bits indicate the device is busy with a cache operation (RDY = 1, ARDY = 0). The cache register becomes available and the page requested in the READ PAGE CACHE operation is transferred to the data register. At this point, data can be output from the cache register, beginning at column address 0. The RANDOM DATA READ (05h-E0h) command can be used to change the column address of the data output by the die (LUN).

After outputting the desired number of bytes from the cache register, either an additional READ PAGE CACHE series (31h, 00h-31h) operation can be started or the READ PAGE CACHE LAST (3Fh) command can be issued.

If the READ PAGE CACHE LAST (3Fh) command is issued, R/B# goes LOW on the target, and RDY = 0 and ARDY = 0 on the die (LUN) for <sup>t</sup>RCBSY while the data register is copied into the cache register. After <sup>t</sup>RCBSY, R/B# goes HIGH and RDY = 1 and ARDY = 1, indicating that the cache register is available and that the die (LUN) is ready. Data can then be output from the cache register, beginning at column address 0. The RANDOM DATA READ (05h-E0h) command can be used to change the column address of the data being output.

For READ PAGE CACHE series (31h, 00h-31h, 3Fh), during the die (LUN) busy time, <sup>t</sup>RCBSY, when RDY = 0 and ARDY = 0, the only valid commands are status operations (70h, 78h) and RESET (FFh). When RDY = 1 and ARDY = 0, the only valid commands during READ PAGE CACHE series (31h, 00h-31h) operations are status operations (70h, 78h), READ MODE (00h), READ PAGE CACHE series (31h, 00h-31h), RANDOM DATA READ (05h-E0h), and RESET (FFh).



## Two-Plane Read Operations

Two-plane read page operations improve data throughput by copying data from more than one plane simultaneously to the specified cache registers. This is done by prepending one or more READ PAGE TWO-PLANE (00h-00h-30h) commands in front of the READ PAGE (00h-30h) command.

When the die (LUN) is ready, the RANDOM DATA READ TWO-PLANE (06h-E0h) command determines which plane outputs data. During data output, the following commands can be used to read and modify the data in the cache registers: RANDOM DATA READ (05h-E0h) and RANDOM DATA INPUT (85h).

## Two-Plane Read Cache Operations

Two-plane read cache operations can be used to output data from more than one cache register while concurrently copying one or more pages from the NAND Flash array to the data register. This is done by prepending READ PAGE TWO-PLANE (00h-00h-30h) commands in front of the PAGE READ CACHE RANDOM (00h-31h) command.

To begin a two-plane read page cache sequence, begin by issuing a READ PAGE TWO-PLANE operation using the READ PAGE TWO-PLANE (00h-00h-30h) and READ PAGE (00h-30h) commands. R/B# goes LOW during 'R and the selected die (LUN) is busy (RDY = 0, ARDY = 0). After 'R (R/B# is HIGH and RDY = 1, ARDY = 1), issue either of these commands:

- READ PAGE CACHE SEQUENTIAL (31h) – copies the next sequential pages from the previously addressed planes from the NAND Flash array to the data registers.
- READ PAGE TWO-PLANE (00h-00h-30h) [in some cases, followed by READ PAGE CACHE RANDOM (00h-31h)] – copies the pages specified from the NAND Flash array to the corresponding data registers.

After the READ PAGE CACHE series (31h, 00h-31h) command has been issued, R/B# goes LOW on the target, and RDY = 0 and ARDY = 0 on the die (LUN) for 'RCBSY while the next pages begin copying data from the array to the data registers. After 'RCBSY, R/B# goes HIGH and the LUN's status register bits indicate the device is busy with a cache operation (RDY = 1, ARDY = 0). The cache registers become available and the pages requested in the READ PAGE CACHE operation are transferred to the data registers. Issue the RANDOM DATA READ TWO-PLANE (06h-E0h) command to determine which cache register will output data. After data is output, the RANDOM DATA READ TWO-PLANE (06h-E0h) command can be used to output data from other cache registers. After a cache register has been selected, the RANDOM DATA READ (05h-E0h) command can be used to change the column address of the data output.

After outputting data from the cache registers, either an additional TWO-PLANE READ CACHE series (31h, 00h-31h) operation can be started or the READ PAGE CACHE LAST (3Fh) command can be issued.

If the READ PAGE CACHE LAST (3Fh) command is issued, R/B# goes LOW on the target, and RDY = 0 and ARDY = 0 on the die (LUN) for 'RCBSY while the data registers are copied into the cache registers. After 'RCBSY, R/B# goes HIGH and RDY = 1 and ARDY = 1, indicating that the cache registers are available and that the die (LUN) is ready. Issue the RANDOM DATA READ TWO-PLANE (06h-E0h) command to determine which cache register will output data. After data is output, the RANDOM DATA READ TWO-PLANE (06h-E0h) command can be used to output data from other cache registers. After a cache register has been selected, the RANDOM DATA READ (05h-E0h) command can be used to change the column address of the data output.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Read Operations

For READ PAGE CACHE series (31h, 00h-31h, 3Fh), during the die (LUN) busy time, <sup>t</sup>RCBSY, when RDY = 0 and ARDY = 0, the only valid commands are status operations (70h, 78h) and RESET (FFh). When RDY = 1 and ARDY = 0, the only valid commands during READ PAGE CACHE series (31h, 00h-31h) operations are status operations (70h, 78h), READ MODE (00h), two-plane read cache series (31h, 00h-00h-30h, 00h-31h), RANDOM DATA READ (06h-E0h, 05h-E0h), and RESET (FFh).

### READ MODE (00h)

The READ MODE (00h) command disables status output and enables data output for the last-selected die (LUN) and cache register after a READ operation (00h-30h, 00h-3Ah, 00h-35h) has been monitored with a status operation (70h, 78h). This command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1). It is also accepted by the die (LUN) during READ PAGE CACHE (31h, 00h-31h) operations (RDY = 1 and ARDY = 0).

In devices that have more than one die (LUN) per target, during and following interleaved die (multi-LUN) operations, the READ STATUS ENHANCED (78h) command must be used to select only one die (LUN) prior to issuing the READ MODE (00h) command. This prevents bus contention.

### READ PAGE (00h-30h)

The READ PAGE (00h-30h) command copies a page from the NAND Flash array to its respective cache register and enables data output. This command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1).

To read a page from the NAND Flash array, write the 00h command to the command register, then write *n* address cycles to the address registers, and conclude with the 30h command. The selected die (LUN) will go busy (RDY = 0, ARDY = 0) for <sup>t</sup>R as data is transferred.

To determine the progress of the data transfer, the host can monitor the target's R/B# signal or, alternatively, the status operations (70h, 78h) can be used. If the status operations are used to monitor the LUN's status, when the die (LUN) is ready (RDY = 1, ARDY = 1), the host disables status output and enables data output by issuing the READ MODE (00h) command. When the host requests data output, output begins at the column address specified.

During data output the RANDOM DATA READ (05h-E0h) command can be issued.

When internal ECC is enabled, the READ STATUS (70h) command is required after the completion of the data transfer (<sup>t</sup>R\_ECC) to determine whether an uncorrectable read error occurred. (<sup>t</sup>R\_ECC is the data transferred with internal ECC enabled.)

In devices that have more than one die (LUN) per target, during and following interleaved die (multi-LUN) operations the READ STATUS ENHANCED (78h) command must be used to select only one die (LUN) prior to the issue of the READ MODE (00h) command. This prevents bus contention.

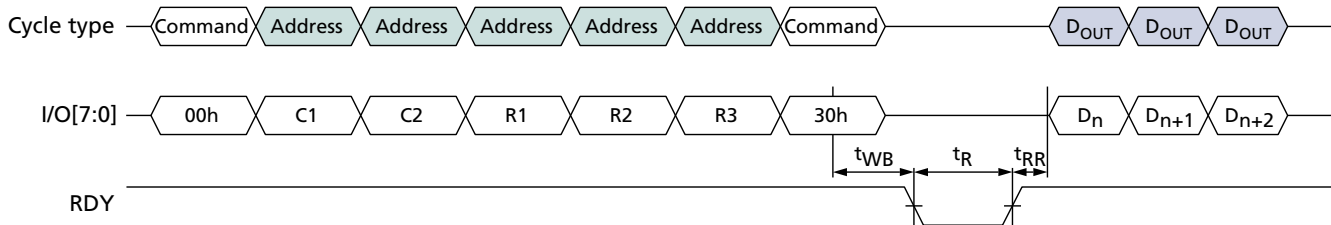
The READ PAGE (00h-30h) command is used as the final command of a two-plane read operation. It is preceded by one or more READ PAGE TWO-PLANE (00h-00h-30h) commands. Data is transferred from the NAND Flash array for all of the addressed planes to their respective cache registers. When the die (LUN) is ready (RDY = 1, ARDY = 1), data output is enabled for the cache register linked to the plane addressed in the READ PAGE (00h-30h) command. When the host requests data output,



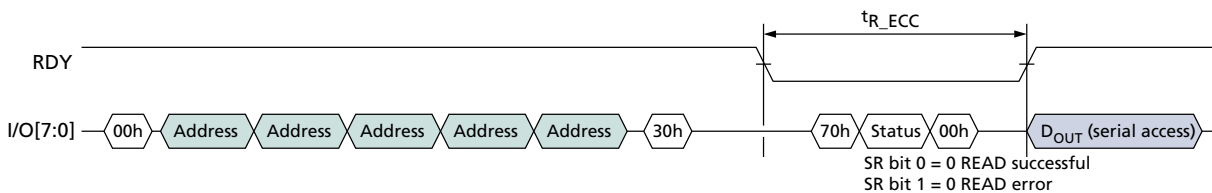
## 4Gb, 8Gb: x8, x16 NAND Flash Memory Read Operations

output begins at the column address last specified in the READ PAGE (00h-30h) command. The RANDOM DATA READ TWO-PLANE (06h-E0h) command is used to enable data output in the other cache registers.

**Figure 38: READ PAGE (00h-30h) Operation**



**Figure 39: READ PAGE (00h-30h) Operation with Internal ECC Enabled**

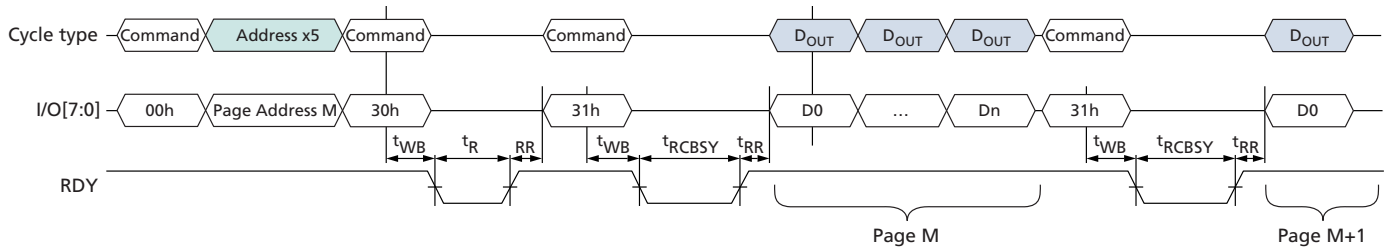


### READ PAGE CACHE SEQUENTIAL (31h)

The READ PAGE CACHE SEQUENTIAL (31h) command reads the next sequential page within a block into the data register while the previous page is output from the cache register. This command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1). It is also accepted by the die (LUN) during READ PAGE CACHE (31h, 00h-31h) operations (RDY = 1 and ARDY = 0).

To issue this command, write 31h to the command register. After this command is issued, R/B# goes LOW and the die (LUN) is busy (RDY = 0, ARDY = 0) for 'RCBSY. After 'RCBSY, R/B# goes HIGH and the die (LUN) is busy with a cache operation (RDY = 1, ARDY = 0), indicating that the cache register is available and that the specified page is copying from the NAND Flash array to the data register. At this point, data can be output from the cache register beginning at column address 0. The RANDOM DATA READ (05h-E0h) command can be used to change the column address of the data being output from the cache register.

The READ PAGE CACHE SEQUENTIAL (31h) command can be used to cross block boundaries. If the READ PAGE CACHE SEQUENTIAL (31h) command is issued after the last page of a block is read into the data register, the next page read will be the next logical block in which the 31h command was issued. Do not issue the READ PAGE CACHE SEQUENTIAL (31h) to cross die (LUN) boundaries. Instead, issue the READ PAGE CACHE LAST (3Fh) command.


**Figure 40: READ PAGE CACHE SEQUENTIAL (31h) Operation**


## READ PAGE CACHE RANDOM (00h-31h)

The READ PAGE CACHE RANDOM (00h-31h) command reads the specified block and page into the data register while the previous page is output from the cache register. This command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1). It is also accepted by the die (LUN) during READ PAGE CACHE (31h, 00h-31h) operations (RDY = 1 and ARDY = 0).

To issue this command, write 00h to the command register, then write  $n$  address cycles to the address register, and conclude by writing 31h to the command register. The column address in the address specified is ignored. The die (LUN) address must match the same die (LUN) address as the previous READ PAGE (00h-30h) command or, if applicable, the previous READ PAGE CACHE RANDOM (00h-31h) command.

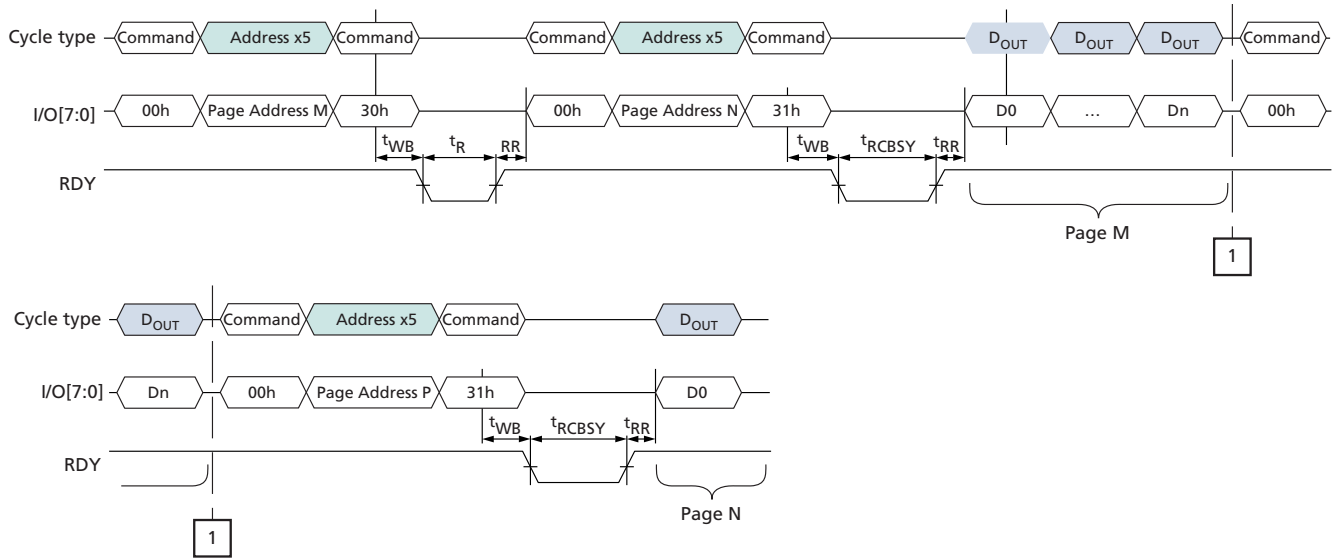
After this command is issued, R/B# goes LOW and the die (LUN) is busy (RDY = 0, ARDY = 0) for  $t_{RCBSY}$ . After  $t_{RCBSY}$ , R/B# goes HIGH and the die (LUN) is busy with a cache operation (RDY = 1, ARDY = 0), indicating that the cache register is available and that the specified page is copying from the NAND Flash array to the data register. At this point, data can be output from the cache register beginning at column address 0. The RANDOM DATA READ (05h-E0h) command can be used to change the column address of the data being output from the cache register.

In devices that have more than one die (LUN) per target, during and following interleaved die (multi-LUN) operations the READ STATUS ENHANCED (78h) command followed by the READ MODE (00h) command must be used to select only one die (LUN) and prevent bus contention.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Read Operations

**Figure 41: READ PAGE CACHE RANDOM (00h-31h) Operation**





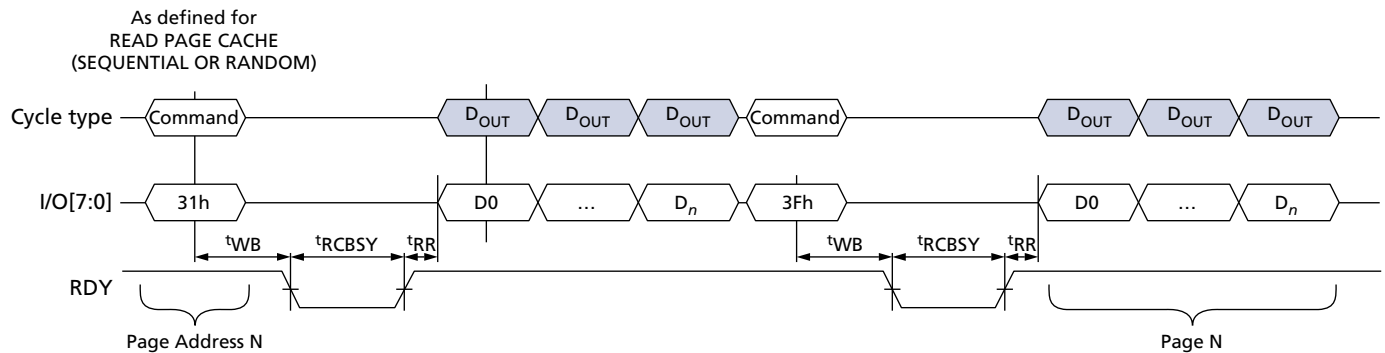
## READ PAGE CACHE LAST (3Fh)

The READ PAGE CACHE LAST (3Fh) command ends the read page cache sequence and copies a page from the data register to the cache register. This command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1). It is also accepted by the die (LUN) during READ PAGE CACHE (31h, 00h-31h) operations (RDY = 1 and ARDY = 0).

To issue the READ PAGE CACHE LAST (3Fh) command, write 3Fh to the command register. After this command is issued, R/B# goes LOW and the die (LUN) is busy (RDY = 0, ARDY = 0) for  $t_{RCBSY}$ . After  $t_{RCBSY}$ , R/B# goes HIGH and the die (LUN) is ready (RDY = 1, ARDY = 1). At this point, data can be output from the cache register, beginning at column address 0. The RANDOM DATA READ (05h-E0h) command can be used to change the column address of the data being output from the cache register.

In devices that have more than one LUN per target, during and following interleaved die (multi-LUN) operations the READ STATUS ENHANCED (78h) command followed by the READ MODE (00h) command must be used to select only one die (LUN) and prevent bus contention.

**Figure 42: READ PAGE CACHE LAST (3Fh) Operation**







## READ PAGE TWO-PLANE 00h-00h-30h

The READ PAGE TWO-PLANE (00h-00h-30h) operation is similar to the PAGE READ (00h-30h) operation. It transfers two pages of data from the NAND Flash array to the data registers. Each page must be from a different plane on the same die.

To enter the READ PAGE TWO-PLANE mode, write the 00h command to the command register, and then write five address cycles for plane 0 (BA6 = 0). Next, write the 00h command to the command register, and five address cycles for plane 1 (BA6 = 1). Finally, issue the 30h command. The first-plane and second-plane addresses must meet the two-plane addressing requirements, and, in addition, they must have identical column addresses.

After the 30h command is written, page data is transferred from both planes to their respective data registers in 'R. During these transfers, R/B# goes LOW. When the transfers are complete, R/B# goes HIGH. To read out the data from the plane 0 data register, pulse RE# repeatedly. After the data cycle from the plane 0 address completes, issue a RANDOM DATA READ TWO-PLANE (06h-E0h) command to select the plane 1 address, then repeatedly pulse RE# to read out the data from the plane 1 data register.

Alternatively, the READ STATUS (70h) command can monitor data transfers. When the transfers are complete, status register bit 6 is set to 1. To read data from the first of the two planes, the user must first issue the RANDOM DATA READ TWO-PLANE (06h-E0h) command and pulse RE# repeatedly.

When the data cycle is complete, issue a RANDOM DATA READ TWO-PLANE (06h-E0h) command to select the other plane. To output the data beginning at the specified column address, pulse RE# repeatedly.

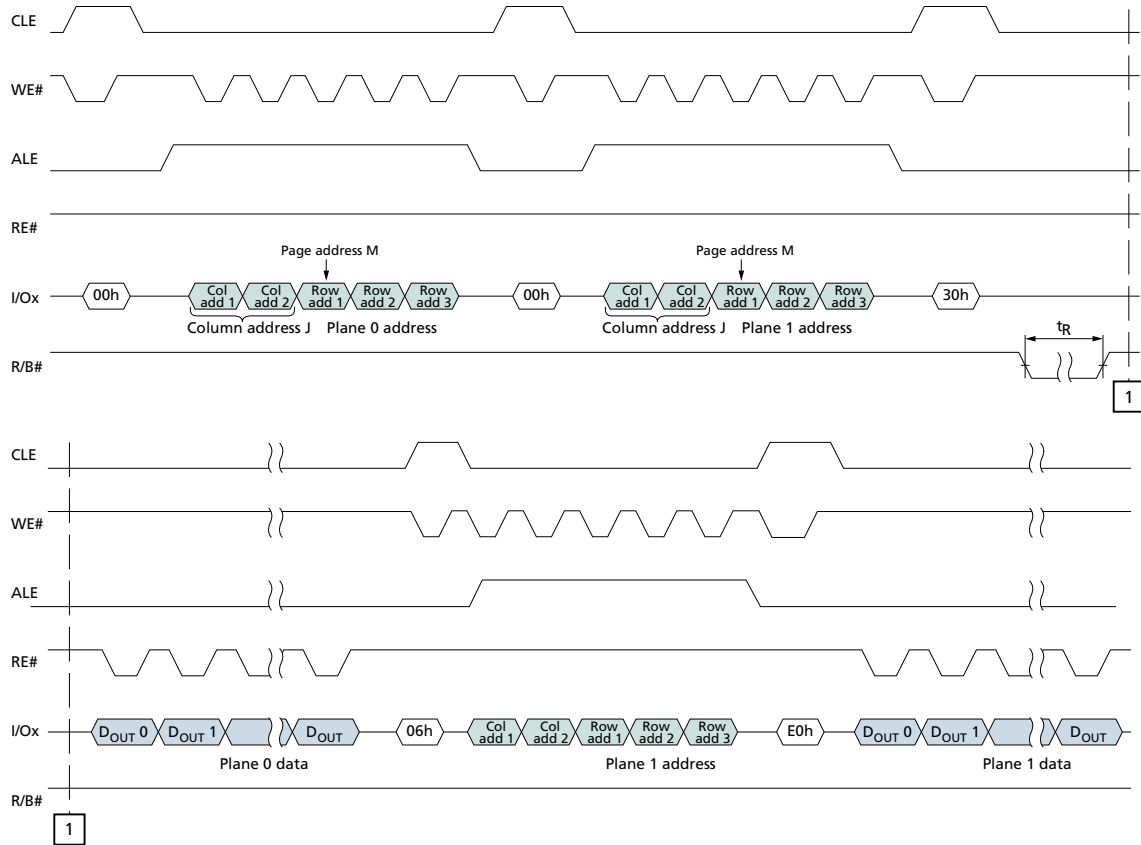
Use of the READ STATUS ENHANCED (78h) command is prohibited during and following a PAGE READ TWO-PLANE operation.





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Read Operations

**Figure 43: READ PAGE TWO-PLANE (00h-00h-30h) Operation**





## Program Operations

Program operations are used to move data from the cache or data registers to the NAND array. During a program operation the contents of the cache and/or data registers are modified by the internal control logic.

Within a block, pages must be programmed sequentially from the least significant page address to the most significant page address (0, 1, 2, ....., 63). During a program operation, the contents of the cache and/or data registers are modified by the internal control logic.

### Program Operations

The PROGRAM PAGE (80h-10h) command, when not preceded by the PROGRAM PAGE TWO-PLANE (80h-11h) command, programs one page from the cache register to the NAND Flash array. When the die (LUN) is ready (RDY = 1, ARDY = 1), the host should check the FAIL bit to verify that the operation has completed successfully.

### Program Cache Operations

The PROGRAM PAGE CACHE (80h-15h) command can be used to improve program operation system performance. When this command is issued, the die (LUN) goes busy (RDY = 0, ARDY = 0) while the cache register contents are copied to the data register, and the die (LUN) is busy with a program cache operation (RDY = 1, ARDY = 0). While the contents of the data register are moved to the NAND Flash array, the cache register is available for an additional PROGRAM PAGE CACHE (80h-15h) or PROGRAM PAGE (80h-10h) command.

For PROGRAM PAGE CACHE series (80h-15h) operations, during the die (LUN) busy times,  $t_{CBSY}$  and  $t_{LPROG}$ , when RDY = 0 and ARDY = 0, the only valid commands are status operations (70h, 78h) and reset (FFh). When RDY = 1 and ARDY = 0, the only valid commands during PROGRAM PAGE CACHE series (80h-15h) operations are status operations (70h, 78h), PROGRAM PAGE CACHE (80h-15h), PROGRAM PAGE (80h-10h), RANDOM DATA INPUT (85h), PROGRAM FOR INTERNAL DATA INPUT (85h), and RESET (FFh).

### Two-Plane Program Operations

The PROGRAM PAGE TWO-PLANE (80h-11h) command can be used to improve program operation system performance by enabling multiple pages to be moved from the cache registers to different planes of the NAND Flash array. This is done by prepending one or more PROGRAM PAGE TWO-PLANE (80h-11h) commands in front of the PROGRAM PAGE (80h-10h) command.

### Two-Plane Program Cache Operations

The PROGRAM PAGE TWO-PLANE (80h-11h) command can be used to improve program cache operation system performance by enabling multiple pages to be moved from the cache registers to the data registers and, while the pages are being transferred from the data registers to different planes of the NAND Flash array, free the cache registers to receive data input from the host. This is done by prepending one or more PROGRAM PAGE TWO-PLANE (80h-11h) commands in front of the PROGRAM PAGE CACHE (80h-15h) command.



## PROGRAM PAGE (80h-10h)

The PROGRAM PAGE (80h-10h) command enables the host to input data to a cache register, and moves the data from the cache register to the specified block and page address in the array of the selected die (LUN). This command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1). It is also accepted by the die (LUN) when it is busy with a PROGRAM PAGE CACHE (80h-15h) operation (RDY = 1, ARDY = 0).

To input a page to the cache register and move it to the NAND array at the block and page address specified, write 80h to the command register. Unless this command has been preceded by a PROGRAM PAGE TWO-PLANE (80h-11h) command, issuing the 80h to the command register clears all of the cache registers' contents on the selected target. Then write  $n$  address cycles containing the column address and row address. Data input cycles follow. Serial data is input beginning at the column address specified. At any time during the data input cycle the RANDOM DATA INPUT (85h) and PROGRAM FOR INTERNAL DATA INPUT (85h) commands may be issued. When data input is complete, write 10h to the command register. The selected LUN will go busy (RDY = 0, ARDY = 0) for  $t_{\text{PROG}}$  as data is transferred.

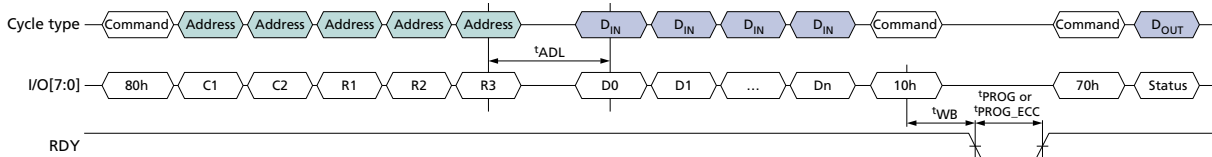
To determine the progress of the data transfer, the host can monitor the target's R/B# signal or, alternatively, the status operations (70h, 78h) may be used. When the die (LUN) is ready (RDY = 1, ARDY = 1), the host should check the status of the FAIL bit.

In devices that have more than one die (LUN) per target, during and following interleaved die (multi-LUN) operations, the READ STATUS ENHANCED (78h) command must be used to select only one die (LUN) for status output. Use of the READ STATUS (70h) command could cause more than one die (LUN) to respond, resulting in bus contention.

The PROGRAM PAGE (80h-10h) command is used as the final command of a two-plane program operation. It is preceded by one or more PROGRAM PAGE TWO-PLANE (80h-11h) commands. Data is transferred from the cache registers for all of the addressed planes to the NAND array. The host should check the status of the operation by using the status operations (70h, 78h).

When internal ECC is enabled, the duration of array programming time is  $t_{\text{PROG\_ECC}}$ . During  $t_{\text{PROG\_ECC}}$ , the internal ECC generates parity bits when error detection is complete.

**Figure 44: PROGRAM PAGE (80h-10h) Operation**



## PROGRAM PAGE CACHE (80h-15h)

The PROGRAM PAGE CACHE (80h-15h) command enables the host to input data to a cache register; copies the data from the cache register to the data register; then moves the data register contents to the specified block and page address in the array of the selected die (LUN). After the data is copied to the data register, the cache register is avail-



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Program Operations

able for additional PROGRAM PAGE CACHE (80h-15h) or PROGRAM PAGE (80h-10h) commands. The PROGRAM PAGE CACHE (80h-15h) command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1). It is also accepted by the die (LUN) when busy with a PROGRAM PAGE CACHE (80h-15h) operation (RDY = 1, ARDY = 0).

To input a page to the cache register to move it to the NAND array at the block and page address specified, write 80h to the command register. Unless this command has been preceded by a PROGRAM PAGE TWO-PLANE (80h-11h) command, issuing the 80h to the command register clears all of the cache registers' contents on the selected target. Then write  $n$  address cycles containing the column address and row address. Data input cycles follow. Serial data is input beginning at the column address specified. At any time during the data input cycle the RANDOM DATA INPUT (85h) and PROGRAM FOR INTERNAL DATA INPUT (85h) commands may be issued. When data input is complete, write 15h to the command register. The selected LUN will go busy (RDY = 0, ARDY = 0) for  $t_{\text{CBSY}}$  to allow the data register to become available from a previous program cache operation, to copy data from the cache register to the data register, and then to begin moving the data register contents to the specified page and block address.

To determine the progress of  $t_{\text{CBSY}}$ , the host can monitor the target's R/B# signal or, alternatively, the status operations (70h, 78h) can be used. When the LUN's status shows that it is busy with a PROGRAM CACHE operation (RDY = 1, ARDY = 0), the host should check the status of the FAILC bit to see if a previous cache operation was successful.

If, after  $t_{\text{CBSY}}$ , the host wants to wait for the program cache operation to complete, without issuing the PROGRAM PAGE (80h-10h) command, the host should monitor ARDY until it is 1. The host should then check the status of the FAIL and FAILC bits.

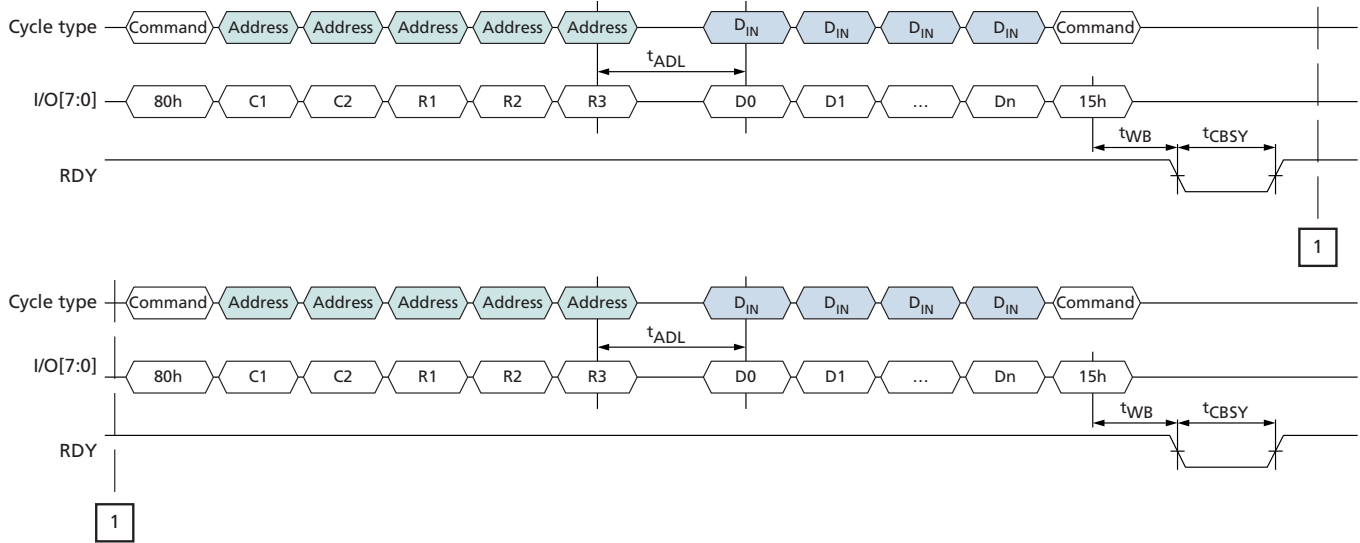
In devices with more than one die (LUN) per target, during and following interleaved die (multi-LUN) operations, the READ STATUS ENHANCED (78h) command must be used to select only one die (LUN) for status output. Use of the READ STATUS (70h) command could cause more than one die (LUN) to respond, resulting in bus contention.

The PROGRAM PAGE CACHE (80h-15h) command is used as the final command of a two-plane program cache operation. It is preceded by one or more PROGRAM PAGE TWO-PLANE (80h-11h) commands. Data for all of the addressed planes is transferred from the cache registers to the corresponding data registers, then moved to the NAND Flash array. The host should check the status of the operation by using the status operations (70h, 78h).

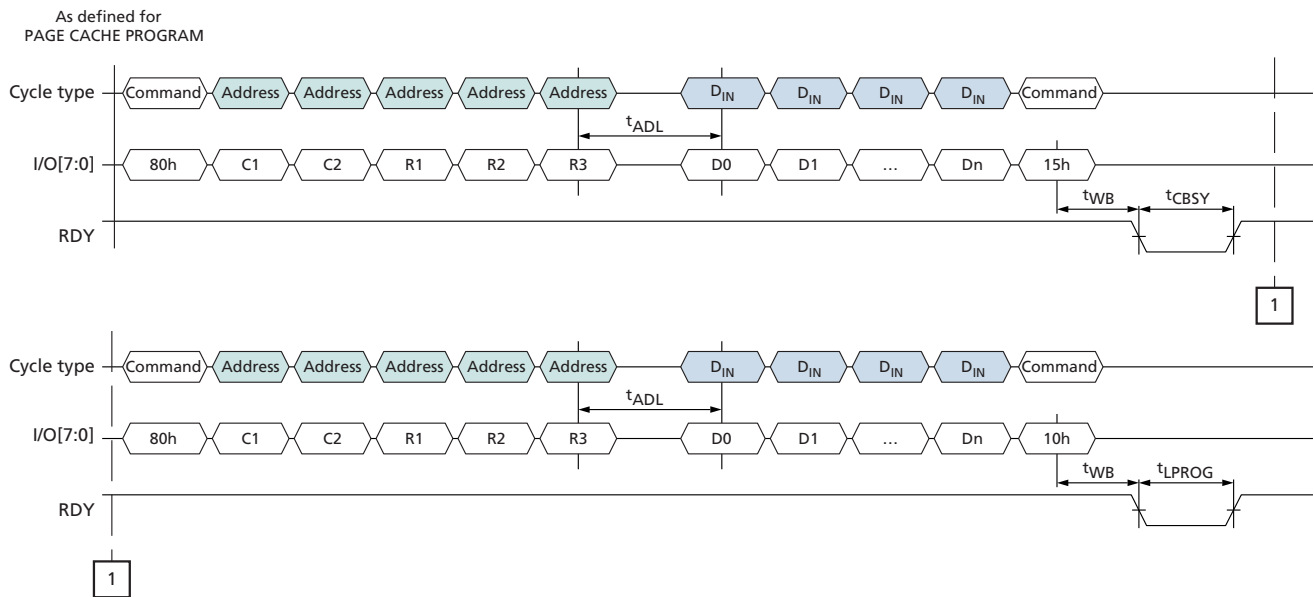


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Program Operations

**Figure 45: PROGRAM PAGE CACHE (80h–15h) Operation (Start)**



**Figure 46: PROGRAM PAGE CACHE (80h–15h) Operation (End)**





## PROGRAM PAGE TWO-PLANE (80h-11h)

The PROGRAM PAGE TWO-PLANE (80h-11h) command enables the host to input data to the addressed plane's cache register and queue the cache register to ultimately be moved to the NAND Flash array. This command can be issued one or more times. Each time a new plane address is specified that plane is also queued for data transfer. To input data for the final plane and to begin the program operation for all previously queued planes, issue either the PROGRAM PAGE (80h-10h) command or the PROGRAM PAGE CACHE (80h-15h) command. All of the queued planes will move the data to the NAND Flash array. This command is accepted by the die (LUN) when it is ready (RDY = 1).

To input a page to the cache register and queue it to be moved to the NAND Flash array at the block and page address specified, write 80h to the command register. Unless this command has been preceded by a PROGRAM PAGE TWO-PLANE (80h-11h) command, issuing the 80h to the command register clears all of the cache registers' contents on the selected target. Write five address cycles containing the column address and row address; data input cycles follow. Serial data is input beginning at the column address specified. At any time during the data input cycle, the RANDOM DATA INPUT (85h) and PROGRAM FOR INTERNAL DATA INPUT (85h) commands can be issued. When data input is complete, write 11h to the command register. The selected die (LUN) will go busy (RDY = 0, ARDY = 0) for <sup>t</sup>DBSY.

To determine the progress of <sup>t</sup>DBSY, the host can monitor the target's R/B# signal or, alternatively, the status operations (70h, 78h) can be used. When the LUN's status shows that it is ready (RDY = 1), additional PROGRAM PAGE TWO-PLANE (80h-11h) commands can be issued to queue additional planes for data transfer. Alternatively, the PROGRAM PAGE (80h-10h) or PROGRAM PAGE CACHE (80h-15h) commands can be issued.

When the PROGRAM PAGE (80h-10h) command is used as the final command of a two-plane program operation, data is transferred from the cache registers to the NAND Flash array for all of the addressed planes during <sup>t</sup>PROG. When the die (LUN) is ready (RDY = 1, ARDY = 1), the host should check the status of the FAIL bit for each of the planes to verify that programming completed successfully.

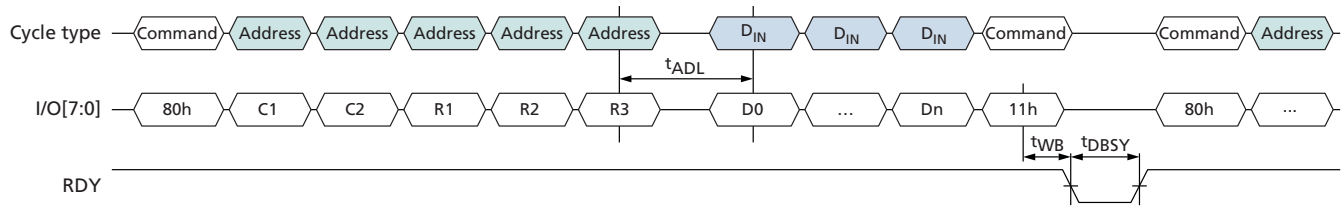
When the PROGRAM PAGE CACHE (80h-15h) command is used as the final command of a program cache two-plane operation, data is transferred from the cache registers to the data registers after the previous array operations finish. The data is then moved from the data registers to the NAND Flash array for all of the addressed planes. This occurs during <sup>t</sup>CBSY. After <sup>t</sup>CBSY, the host should check the status of the FAILC bit for each of the planes from the previous program cache operation, if any, to verify that programming completed successfully.

For the PROGRAM PAGE TWO-PLANE (80h-11h), PROGRAM PAGE (80h-10h), and PROGRAM PAGE CACHE (80h-15h) commands, see Two-Plane Operations for two-plane addressing requirements.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Program Operations

**Figure 47: PROGRAM PAGE TWO-PLANE (80h–11h) Operation**





## Erase Operations

Erase operations are used to clear the contents of a block in the NAND Flash array to prepare its pages for program operations.

### Erase Operations

The ERASE BLOCK (60h-D0h) command, when not preceded by the ERASE BLOCK TWO-PLANE (60h-D1h) command, erases one block in the NAND Flash array. When the die (LUN) is ready (RDY = 1, ARDY = 1), the host should check the FAIL bit to verify that this operation completed successfully.

### TWO-PLANE ERASE Operations

The ERASE BLOCK TWO-PLANE (60h-D1h) command can be used to further system performance of erase operations by allowing more than one block to be erased in the NAND array. This is done by prepending one or more ERASE BLOCK TWO-PLANE (60h-D1h) commands in front of the ERASE BLOCK (60h-D0h) command. See Two-Plane Operations for details.

## ERASE BLOCK (60h-D0h)

The ERASE BLOCK (60h-D0h) command erases the specified block in the NAND Flash array. This command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1).

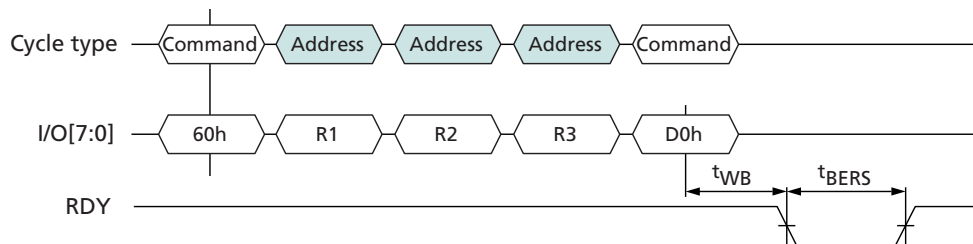
To erase a block, write 60h to the command register. Then write three address cycles containing the row address; the page address is ignored. Conclude by writing D0h to the command register. The selected die (LUN) will go busy (RDY = 0, ARDY = 0) for  $t_{BERS}$  while the block is erased.

To determine the progress of an ERASE operation, the host can monitor the target's R/B# signal, or alternatively, the status operations (70h, 78h) can be used. When the die (LUN) is ready (RDY = 1, ARDY = 1) the host should check the status of the FAIL bit.

In devices that have more than one die (LUN) per target, during and following interleaved die (multi-LUN) operations, the READ STATUS ENHANCED (78h) command must be used to select only one die (LUN) for status output. Use of the READ STATUS (70h) command could cause more than one die (LUN) to respond, resulting in bus contention.

The ERASE BLOCK (60h-D0h) command is used as the final command of an erase two-plane operation. It is preceded by one or more ERASE BLOCK TWO-PLANE (60h-D1h) commands. All blocks in the addressed planes are erased. The host should check the status of the operation by using the status operations (70h, 78h). See Two-Plane Operations for two-plane addressing requirements.

**Figure 48: ERASE BLOCK (60h-D0h) Operation**







## ERASE BLOCK TWO-PLANE (60h-D1h)

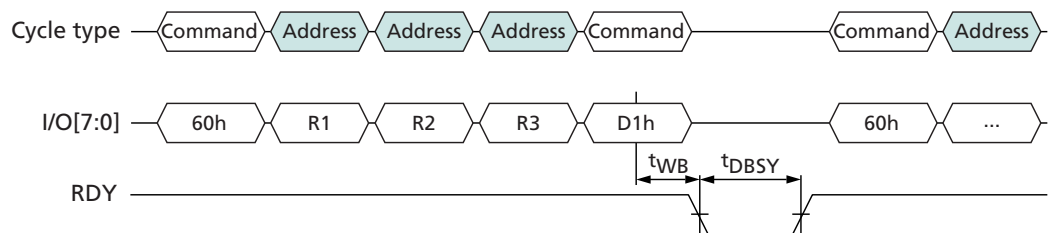
The ERASE BLOCK TWO-PLANE (60h-D1h) command queues a block in the specified plane to be erased in the NAND Flash array. This command can be issued one or more times. Each time a new plane address is specified, that plane is also queued for a block to be erased. To specify the final block to be erased and to begin the ERASE operation for all previously queued planes, issue the ERASE BLOCK (60h-D0h) command. This command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1).

To queue a block to be erased, write 60h to the command register, then write three address cycles containing the row address; the page address is ignored. Conclude by writing D1h to the command register. The selected die (LUN) will go busy (RDY = 0, ARDY = 0) for  $t_{DBSY}$ .

To determine the progress of  $t_{DBSY}$ , the host can monitor the target's R/B# signal, or alternatively, the status operations (70h, 78h) can be used. When the LUN's status shows that it is ready (RDY = 1, ARDY = 1), additional ERASE BLOCK TWO-PLANE (60h-D1h) commands can be issued to queue additional planes for erase. Alternatively, the ERASE BLOCK (60h-D0h) command can be issued to erase all of the queued blocks.

For two-plane addressing requirements for the ERASE BLOCK TWO-PLANE (60h-D1h) and ERASE BLOCK (60h-D0h) commands, see Two-Plane Operations.

**Figure 49: ERASE BLOCK TWO-PLANE (60h-D1h) Operation**





## Internal Data Move Operations

Internal data move operations make it possible to transfer data within a device from one page to another using the cache register. This is particularly useful for block management and wear leveling.

The INTERNAL DATA MOVE operation is a two-step process consisting of a READ FOR INTERNAL DATA MOVE (00h-35h) and a PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command. To move data from one page to another on the same plane, first issue the READ FOR INTERNAL DATA MOVE (00h-35h) command. When the die (LUN) is ready (RDY = 1, ARDY = 1), the host can transfer the data to a new page by issuing the PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command. When the die (LUN) is again ready (RDY = 1, ARDY = 1), the host should check the FAIL bit to verify that this operation completed successfully.

To prevent bit errors from accumulating over multiple INTERNAL DATA MOVE operations, it is recommended that the host read the data out of the cache register after the READ FOR INTERNAL DATA MOVE (00h-35h) completes and prior to issuing the PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command. The RANDOM DATA READ (05h-E0h) command can be used to change the column address. The host should check the data for ECC errors and correct them. When the PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command is issued, any corrected data can be input. The PROGRAM FOR INTERNAL DATA INPUT (85h) command can be used to change the column address.

It is not possible to use the READ FOR INTERNAL DATA MOVE operation to move data from one plane to another or from one die (LUN) to another. Instead, use a READ PAGE (00h-30h) or READ FOR INTERNAL DATA MOVE (00h-35h) command to read the data out of the NAND, and then use a PROGRAM PAGE (80h-10h) command with data input to program the data to a new plane or die (LUN).

Between the READ FOR INTERNAL DATA MOVE (00h-35h) and PROGRAM FOR INTERNAL DATA MOVE (85h-10h) commands, the following commands are supported: status operations (70h, 78h) and column address operations (05h-E0h, 06h-E0h, 85h). The RESET operation (FFh) can be issued after READ FOR INTERNAL DATA MOVE (00h-35h), but the contents of the cache registers on the target are not valid.

In devices that have more than one die (LUN) per target, once the READ FOR INTERNAL DATA MOVE (00h-35h) is issued, interleaved die (multi-LUN) operations are prohibited until after the PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command is issued.

### Two-Plane Read for Internal Data Move Operations

Two-plane internal data move read operations improve read data throughput by copying data simultaneously from more than one plane to the specified cache registers. This is done by issuing the READ PAGE TWO-PLANE (00h-00h-30h) command or the READ FOR INTERNAL DATA MOVE (00h-00h-35h) command.

The INTERNAL DATA MOVE PROGRAM TWO-PLANE (85h-11h) command can be used to further system performance of PROGRAM FOR INTERNAL DATA MOVE operations by enabling movement of multiple pages from the cache registers to different planes of the NAND Flash array. This is done by prepending one or more PROGRAM FOR INTERNAL DATA MOVE (85h-11h) commands in front of the PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command. See Two-Plane Operations for details.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Internal Data Move Operations

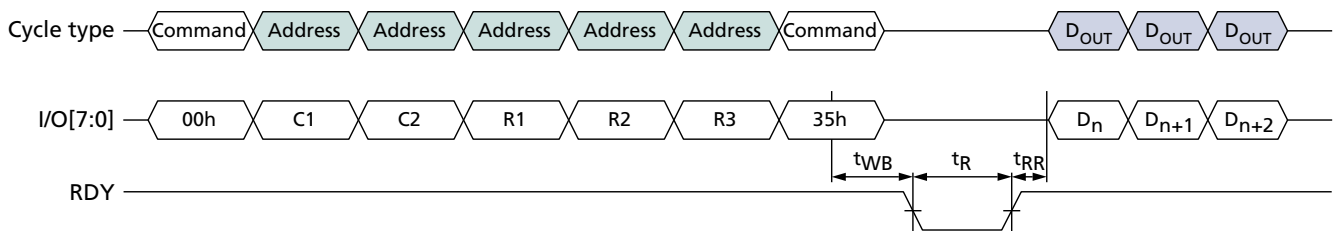
### READ FOR INTERNAL DATA MOVE (00h-35h)

The READ FOR INTERNAL DATA MOVE (00h-35h) command is functionally identical to the READ PAGE (00h-30h) command, except that 35h is written to the command register instead of 30h.

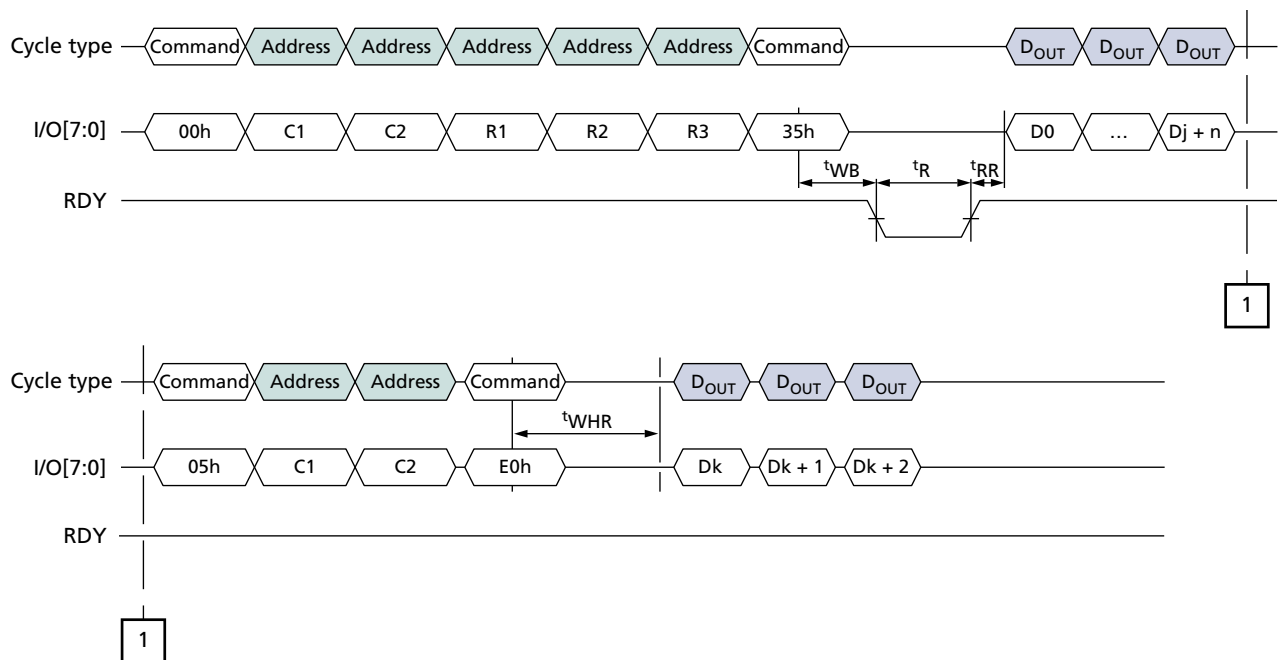
Though it is not required, it is recommended that the host read the data out of the device to verify the data prior to issuing the PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command to prevent the propagation of data errors.

If internal ECC is enabled, the data does not need to be toggled out by the host to be corrected and moving data can then be written to a new page without data reloading, which improves system performance.

**Figure 50: READ FOR INTERNAL DATA MOVE (00h-35h) Operation**



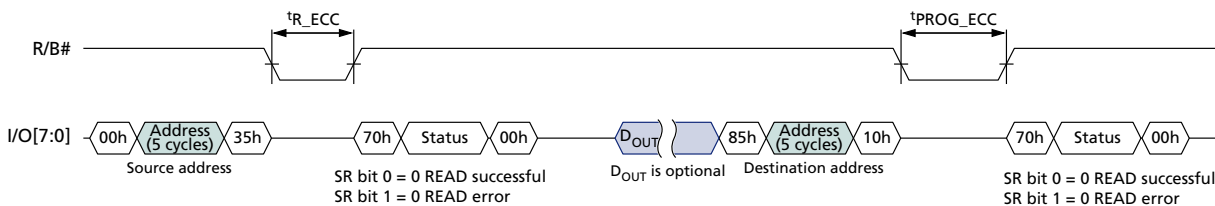
**Figure 51: READ FOR INTERNAL DATA MOVE (00h-35h) with RANDOM DATA READ (05h-E0h)**



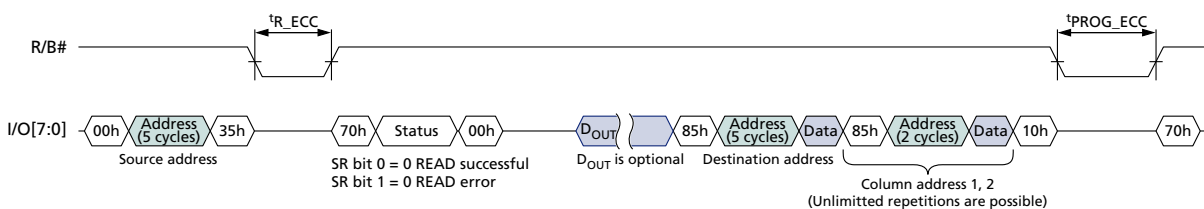


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Internal Data Move Operations

**Figure 52: INTERNAL DATA MOVE (85h-10h) with Internal ECC Enabled**



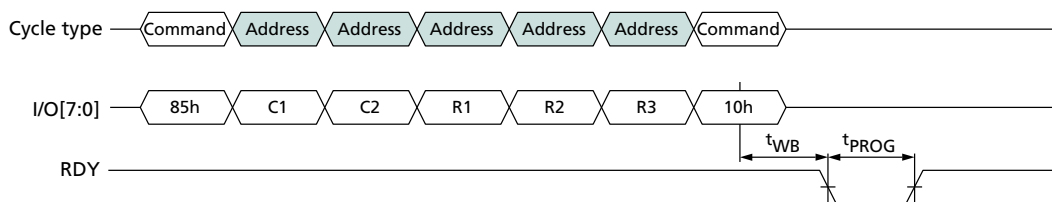
**Figure 53: INTERNAL DATA MOVE (85h-10h) with RANDOM DATA INPUT with Internal ECC Enabled**



### PROGRAM FOR INTERNAL DATA MOVE (85h-10h)

The PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command is functionally identical to the PROGRAM PAGE (80h-10h) command, except that when 85h is written to the command register, cache register contents are not cleared.

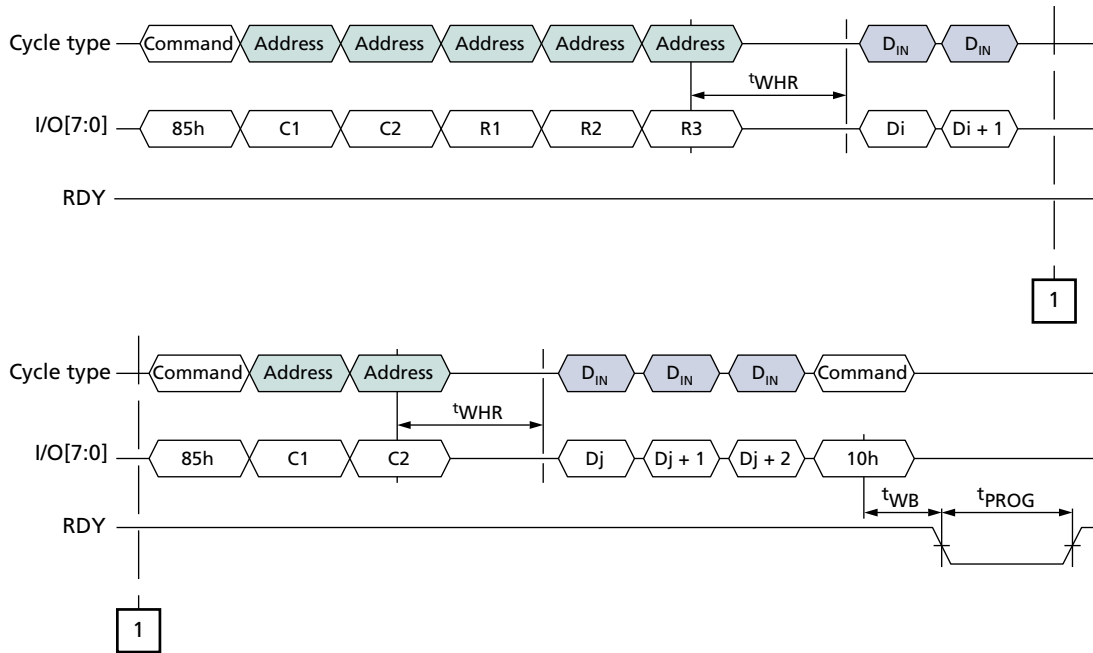
**Figure 54: PROGRAM FOR INTERNAL DATA MOVE (85h-10h) Operation**





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Internal Data Move Operations

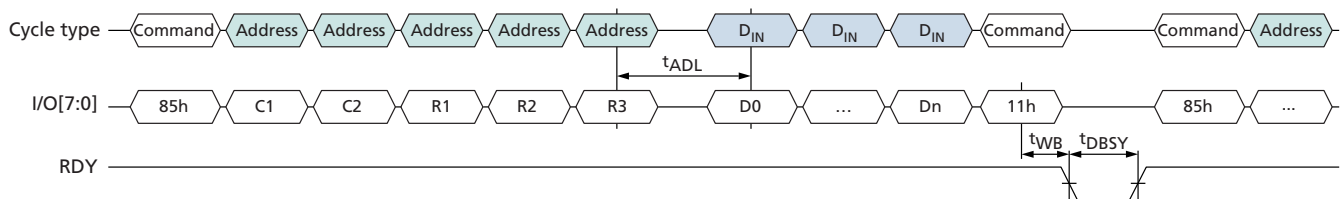
**Figure 55: PROGRAM FOR INTERNAL DATA MOVE (85h-10h) with RANDOM DATA INPUT (85h)**



### PROGRAM FOR INTERNAL DATA MOVE TWO-PLANE (85h-11h)

The PROGRAM FOR INTERNAL DATA MOVE TWO-PLANE (85h-11h) command is functionally identical to the PROGRAM PAGE TWO-PLANE (85h-11h) command, except that when 85h is written to the command register, cache register contents are not cleared. See Program Operations for further details.

**Figure 56: PROGRAM FOR INTERNAL DATA MOVE TWO-PLANE (85h-11h) Operation**





## Block Lock Feature

The block lock feature protects either the entire device or ranges of blocks from being programmed and erased. Using the block lock feature is preferable to using WP# to prevent PROGRAM and ERASE operations.

Block lock is enabled and disabled at power-on through the LOCK pin. At power-on, if LOCK is LOW, all BLOCK LOCK commands are disabled. However if LOCK is HIGH at power-on, the BLOCK LOCK commands are enabled and, by default, all the blocks on the device are protected, or locked, from PROGRAM and ERASE operations, even if WP# is HIGH.

Before the contents of the device can be modified, the device must first be unlocked. Either a range of blocks or the entire device may be unlocked. PROGRAM and ERASE operations complete successfully only in the block ranges that have been unlocked. Blocks, once unlocked, can be locked again to protect them from further PROGRAM and ERASE operations.

Blocks that are locked can be protected further, or locked tight. When locked tight, the device's blocks can no longer be locked or unlocked until the device is power cycled.

## WP# and Block Lock

The following is true when the block lock feature is enabled:

- Holding WP# LOW locks all blocks, provided the blocks are not locked tight.
- If WP# is held LOW to lock blocks, then returned to HIGH, a new UNLOCK command must be issued to unlock blocks.

## UNLOCK (23h-24h)

By default at power-on, if LOCK is HIGH, all the blocks are locked and protected from PROGRAM and ERASE operations. The UNLOCK (23h) command is used to unlock a range of blocks. Unlocked blocks have no protection and can be programmed or erased.

The UNLOCK command uses two registers, a lower boundary block address register and an upper boundary block address register, and the invert area bit to determine what range of blocks are unlocked. When the invert area bit = 0, the range of blocks within the lower and upper boundary address registers are unlocked. When the invert area bit = 1, the range of blocks outside the boundaries of the lower and upper boundary address registers are unlocked. The lower boundary block address must be less than the upper boundary block address. The figures below show examples of how the lower and upper boundary address registers work with the invert area bit.

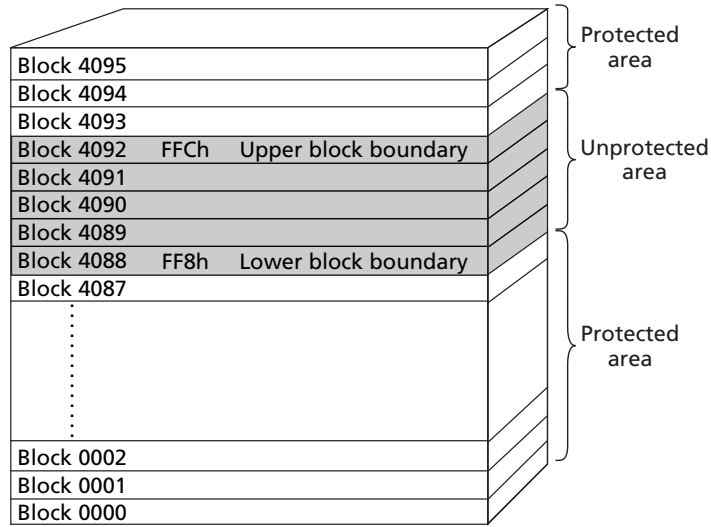
To unlock a range of blocks, issue the UNLOCK (23h) command followed by the appropriate address cycles that indicate the lower boundary block address. Then issue the 24h command followed by the appropriate address cycles that indicate the upper boundary block address. The least significant page address bit, PA0, should be set to 1 if setting the invert area bit; otherwise, it should be 0. The other page address bits should be 0.

Only one range of blocks can be specified in the lower and upper boundary block address registers. If after unlocking a range of blocks the UNLOCK command is again issued, the new block address range determines which blocks are unlocked. The previous unlocked block address range is not retained.

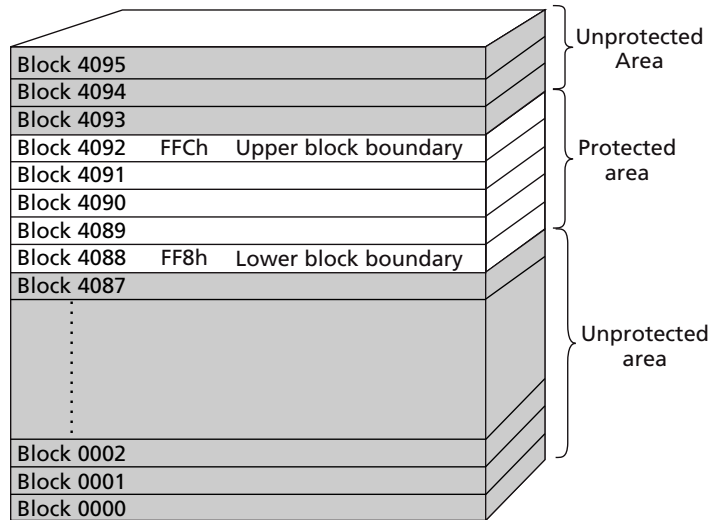


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Block Lock Feature

**Figure 57: Flash Array Protected: Invert Area Bit = 0**



**Figure 58: Flash Array Protected: Invert Area Bit = 1**





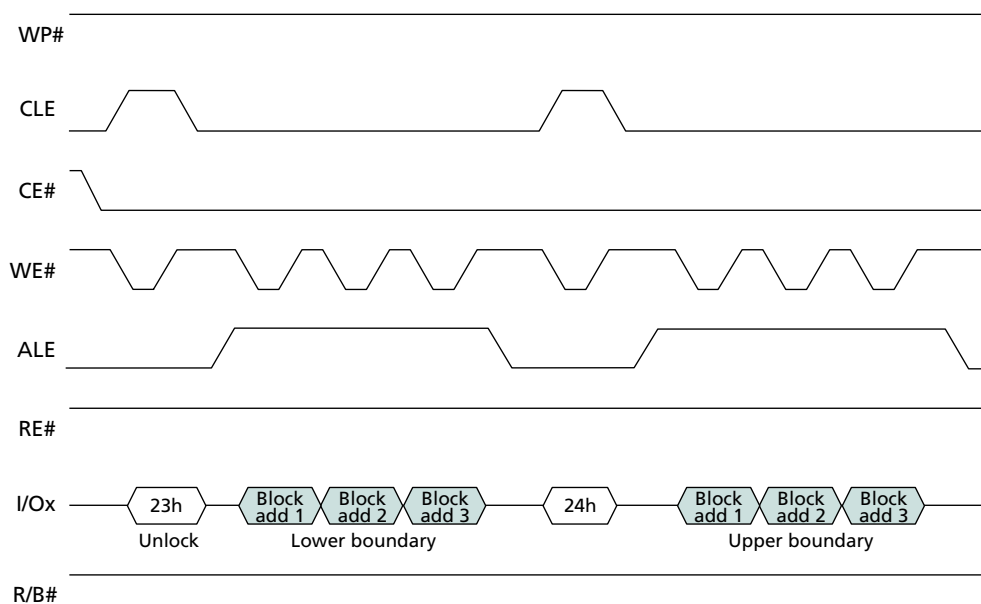
## 4Gb, 8Gb: x8, x16 NAND Flash Memory Block Lock Feature

**Table 19: Block Lock Address Cycle Assignments**

| ALE Cycle | I/O[15:8] <sup>1</sup> | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0                         |
|-----------|------------------------|------|------|------|------|------|------|------|------------------------------|
| First     | LOW                    | BA7  | BA6  | LOW  | LOW  | LOW  | LOW  | LOW  | Invert area bit <sup>2</sup> |
| Second    | LOW                    | BA15 | BA14 | BA13 | BA12 | BA11 | BA10 | BA9  | BA8                          |
| Third     | LOW                    | LOW  | LOW  | LOW  | LOW  | LOW  | LOW  | BA17 | BA16                         |

- Notes: 1. I/O[15:8] is applicable only for x16 devices.  
2. Invert area bit is applicable for 24h command; it may be LOW or HIGH for 23h command.

**Figure 59: UNLOCK Operation**







## LOCK (2Ah)

By default at power-on, if LOCK is HIGH, all the blocks are locked and protected from PROGRAM and ERASE operations. If portions of the device are unlocked using the UNLOCK (23h) command, they can be locked again using the LOCK (2Ah) command. The LOCK command locks all of the blocks in the device. Locked blocks are write-protected from PROGRAM and ERASE operations.

To lock all of the blocks in the device, issue the LOCK (2Ah) command.

When a PROGRAM or ERASE operation is issued to a locked block, R/B# goes LOW for t<sub>LBSY</sub>. The PROGRAM or ERASE operation does not complete. Any READ STATUS command reports bit 7 as 0, indicating that the block is protected.

The LOCK (2Ah) command is disabled if LOCK is LOW at power-on or if the device is locked tight.

**Figure 60: LOCK Operation**





## LOCK TIGHT (2Ch)

The LOCK TIGHT (2Ch) command prevents locked blocks from being unlocked and also prevents unlocked blocks from being locked. When this command is issued, the UNLOCK (23h) and LOCK (2Ah) commands are disabled. This provides an additional level of protection against inadvertent PROGRAM and ERASE operations to locked blocks.

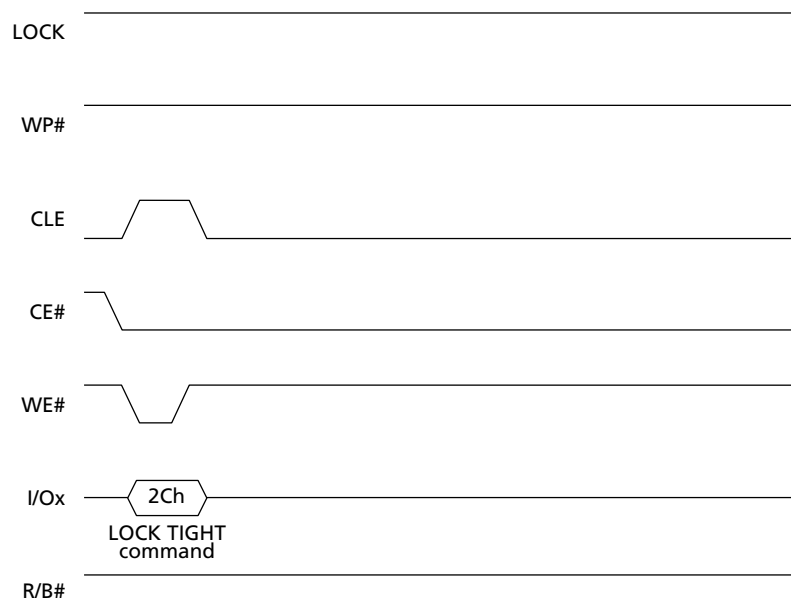
To implement LOCK TIGHT in all of the locked blocks in the device, verify that WP# is HIGH and then issue the LOCK TIGHT (2Ch) command.

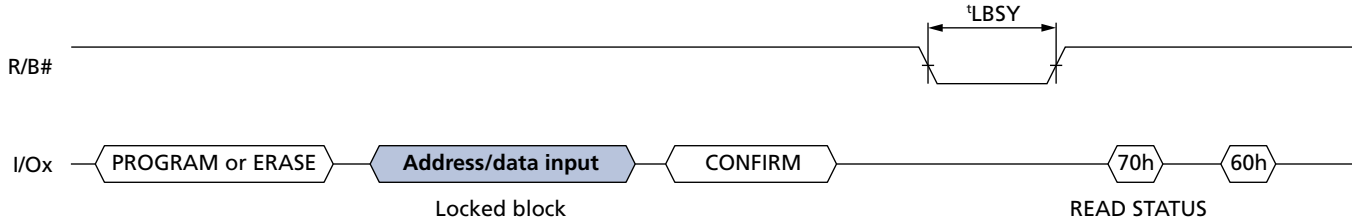
When a PROGRAM or ERASE operation is issued to a locked block that has also been locked tight, R/B# goes LOW for <sup>1</sup>LBSY. The PROGRAM or ERASE operation does not complete. The READ STATUS (70h) command reports bit 7 as 0, indicating that the block is protected. PROGRAM and ERASE operations complete successfully to blocks that were not locked at the time the LOCK TIGHT command was issued.

After the LOCK TIGHT command is issued, the command cannot be disabled via a software command. The only ways to disable the lock tight status is to power cycle the device. When the lock tight status is disabled, all of the blocks become locked, the same as if the LOCK (2Ah) command had been issued.

The LOCK TIGHT (2Ch) command is disabled if LOCK is LOW at power-on.

**Figure 61: LOCK TIGHT Operation**



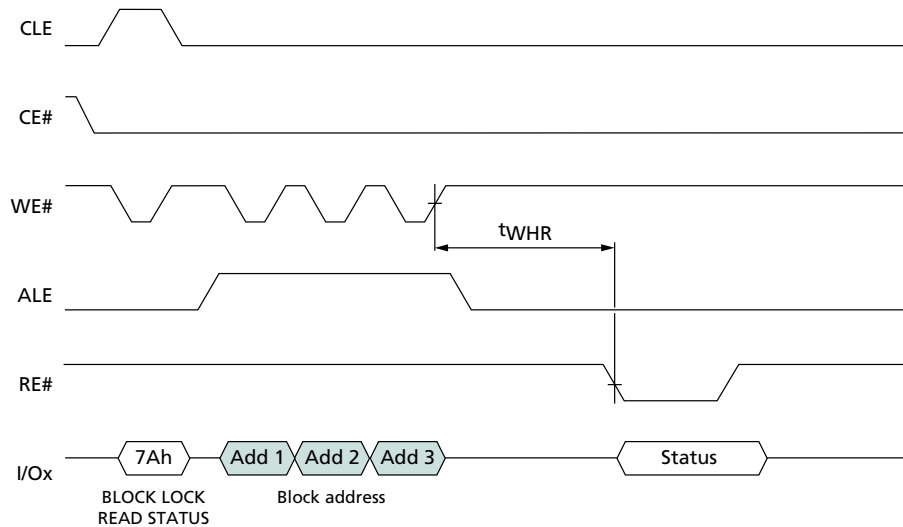

**Figure 62: PROGRAM/ERASE Issued to Locked Block**


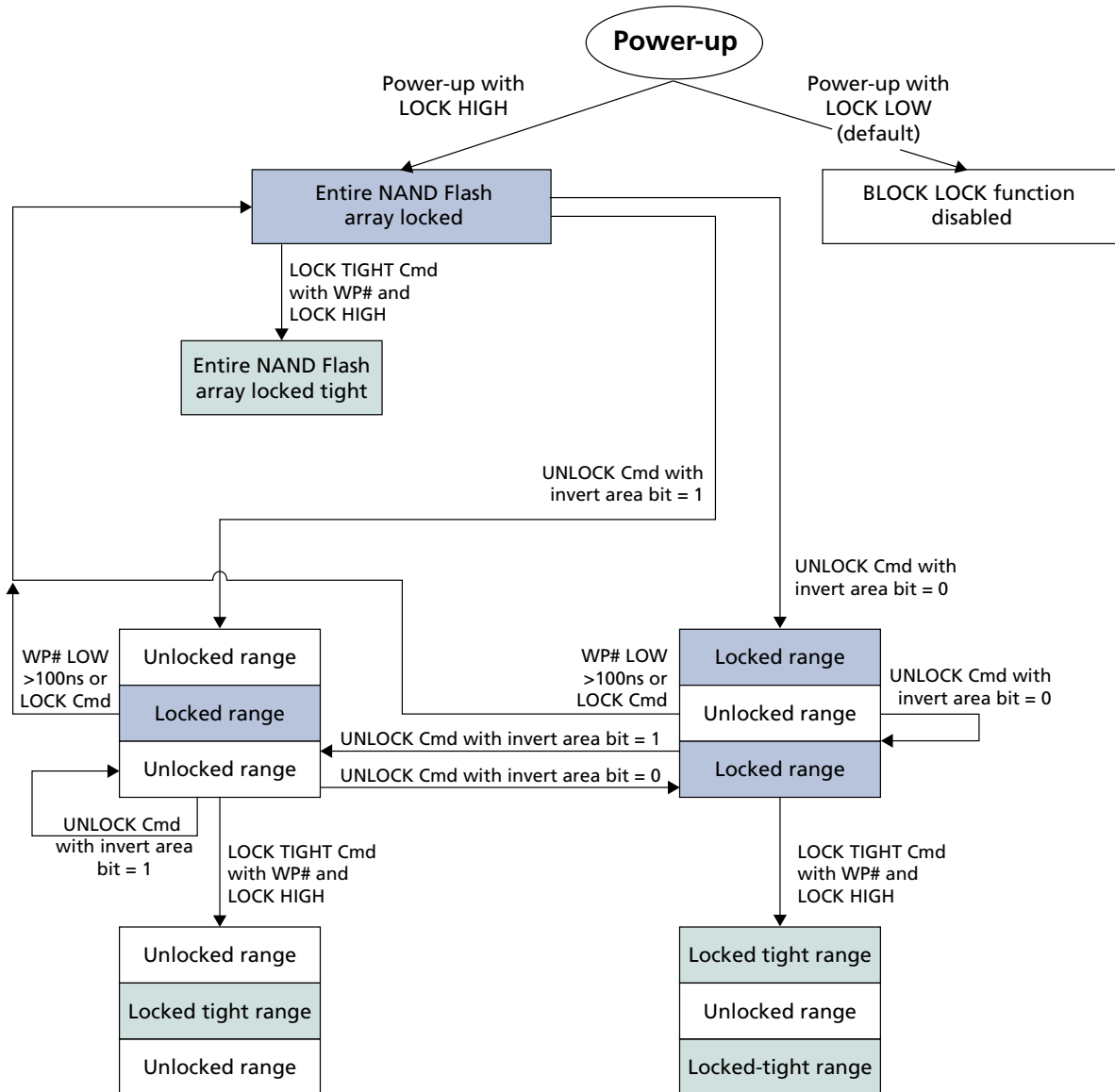
## BLOCK LOCK READ STATUS (7Ah)

The BLOCK LOCK READ STATUS (7Ah) command is used to determine the protection status of individual blocks. The address cycles have the same format, as shown below, and the invert area bit should be set LOW. On the falling edge of RE# the I/O pins output the block lock status register, which contains the information on the protection status of the block.

**Table 20: Block Lock Status Register Bit Definitions**

| Block Lock Status Register Definitions            | I/O[7:3] | I/O2 (Lock#) | I/O1 (LT#) | I/O0 (LT) |
|---|----------|--------------|------------|-----------|
| Block is locked tight                             | X        | 0            | 0          | 1         |
| Block is locked                                   | X        | 0            | 1          | 0         |
| Block is unlocked, and device is locked tight     | X        | 1            | 0          | 1         |
| Block is unlocked, and device is not locked tight | X        | 1            | 1          | 0         |

**Figure 63: BLOCK LOCK READ STATUS**



**Figure 64: BLOCK LOCK Flowchart**




## One-Time Programmable (OTP) Operations

This Micron NAND Flash device offers a protected, one-time programmable NAND Flash memory area. Thirty full pages (2112 bytes per page) of OTP data are available on the device, and the entire range is guaranteed to be good. The OTP area is accessible only through the OTP commands. Customers can use the OTP area any way they choose; typical uses include programming serial numbers or other data for permanent storage.

The OTP area leaves the factory in an unwritten state (all bits are 1s). Programming or partial-page programming enables the user to program only 0 bits in the OTP area. The OTP area cannot be erased, whether it is protected or not. Protecting the OTP area prevents further programming of that area.

Micron provides a unique way to program and verify data before permanently protecting it and preventing future changes. The OTP area is only accessible while in OTP operation mode. To set the device to OTP operation mode, issue the SET FEATURE (EFh) command to feature address 90h and write 01h to P1, followed by three cycles of 00h to P2-P4. For parameters to enter OTP mode, see Features Operations.

When the device is in OTP operation mode, all subsequent PAGE READ (00h-30h) and PROGRAM PAGE (80h-10h) commands are applied to the OTP area. The OTP area is assigned to page addresses 02h-1Fh. To program an OTP page, issue the PROGRAM PAGE (80h-10h) command. The pages must be programmed in the ascending order. Similarly, to read an OTP page, issue the PAGE READ (00h-30h) command.

Protecting the OTP is done by entering OTP protect mode. To set the device to OTP protect mode, issue the SET FEATURE (EFh) command to feature address 90h and write 03h to P1, followed by three cycles of 00h to P2-P4.

To determine whether the device is busy during an OTP operation, either monitor R/B# or use the READ STATUS (70h) command.

To exit OTP operation or protect mode, write 00h to P1 at feature address 90h.

## Legacy OTP Commands

For legacy OTP commands, OTP DATA PROGRAM (A0h-10h), OTP DATA PROTECT (A5h-10h), and OTP DATA READ (AFh-30h), refer to the MT29F4GxxAxC data sheet.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory One-Time Programmable (OTP) Operations

### OTP DATA PROGRAM (80h-10h)

The OTP DATA PROGRAM (80h-10h) command is used to write data to the pages within the OTP area. An entire page can be programmed at one time, or a page can be partially programmed up to eight times. Only the OTP area allows up to eight partial-page programs. The rest of the blocks support only four partial-page programs. There is no ERASE operation for OTP pages.

PROGRAM PAGE enables programming into an offset of an OTP page using two bytes of the column address (CA[12:0]). The command is compatible with the RANDOM DATA INPUT (85h) command. The PROGRAM PAGE command will not execute if the OTP area has been protected.

To use the PROGRAM PAGE command, issue the 80h command. Issue  $n$  address cycles. The first two address cycles are the column address. For the remaining cycles, select a page in the range of 02h-00h through 1Fh-00h. Next, write from 1–2112 bytes of data. After data input is complete, issue the 10h command. The internal control logic automatically executes the proper programming algorithm and controls the necessary timing for programming and verification.

R/B# goes LOW for the duration of the array programming time ( $t_{\text{PROG}}$ ). The READ STATUS (70h) command is the only valid command for reading status in OTP operation mode. Bit 5 of the status register reflects the state of R/B#. When the device is ready, read bit 0 of the status register to determine whether the operation passed or failed (see Status Operations). Each OTP page can be programmed to 8 partial-page programming.

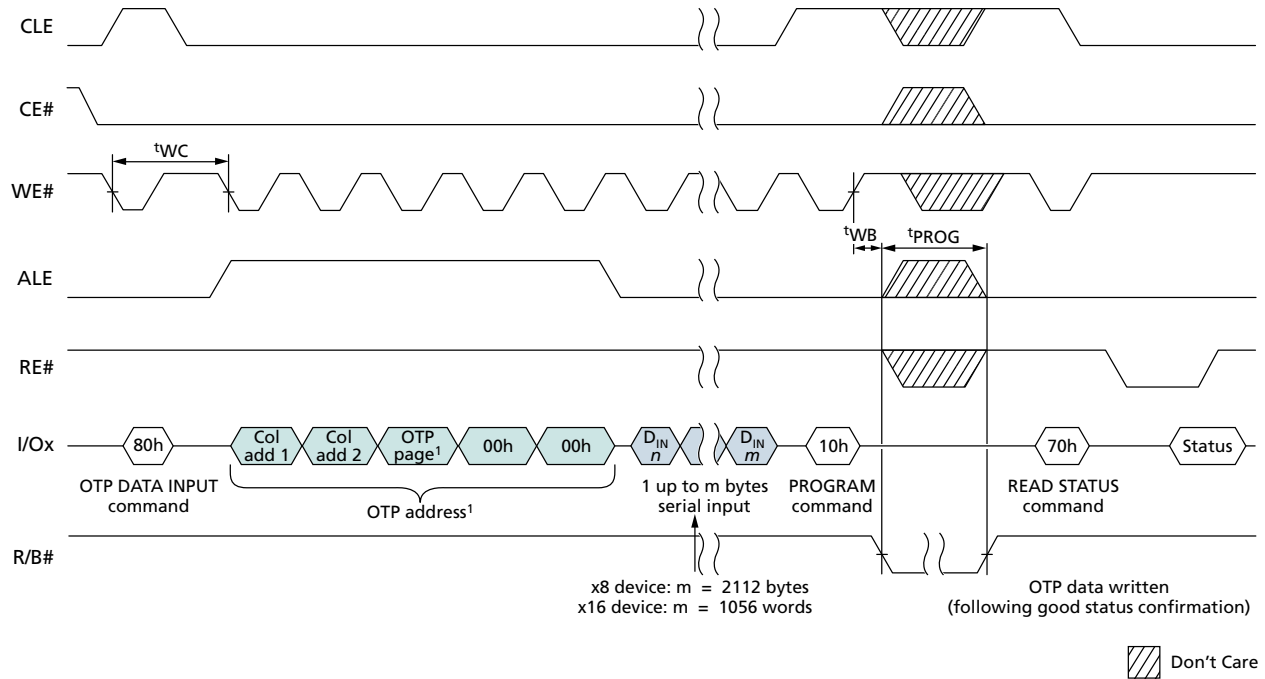


## 4Gb, 8Gb: x8, x16 NAND Flash Memory One-Time Programmable (OTP) Operations

### RANDOM DATA INPUT (85h)

After the initial OTP data set is input, additional data can be written to a new column address with the RANDOM DATA INPUT (85h) command. The RANDOM DATA INPUT command can be used any number of times in the same page prior to the OTP PAGE WRITE (10h) command being issued.

**Figure 65: OTP DATA PROGRAM (After Entering OTP Operation Mode)**

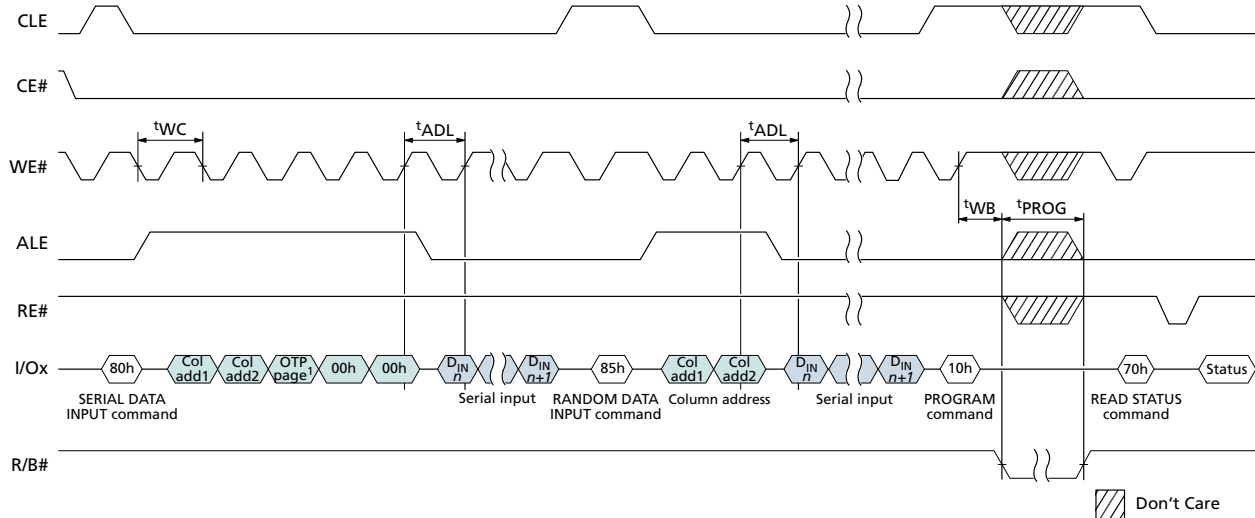


Note: 1. The OTP page must be within the 02h–1Fh range.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory One-Time Programmable (OTP) Operations

**Figure 66: OTP DATA PROGRAM Operation with RANDOM DATA INPUT (After Entering OTP Operation Mode)**



### OTP DATA PROTECT (80h-10)

The OTP area is protected on a block basis. To protect a block, set the device to OTP protect mode, then issue the PROGRAM PAGE (80h-10h) command and write OTP address 00h, 00h, 00h, 00h. To set the device to OTP protect mode, issue the SET FEATURE (EFh) command to 90h (feature address) and write 03h to P1, followed by three cycles of 00h to P2-P4.

After the data is protected, it cannot be programmed further. When the OTP area is protected, the pages within the area are no longer programmable and cannot be unprotected.

To use the PROGRAM PAGE command to protect the OTP area, issue the 80h command, followed by  $n$  address cycles, write 00h data, data cycle of 00h, followed by the 10h command. (An example of the address sequence is shown in the following figure.) If an OTP DATA PROGRAM command is issued after the OTP area has been protected, R/B# will go LOW for  $t_{OBSY}$ .

The READ STATUS (70h) command is the only valid command for reading status in OTP operation mode. Bit 5 of the status register reflects the state of R/B#.

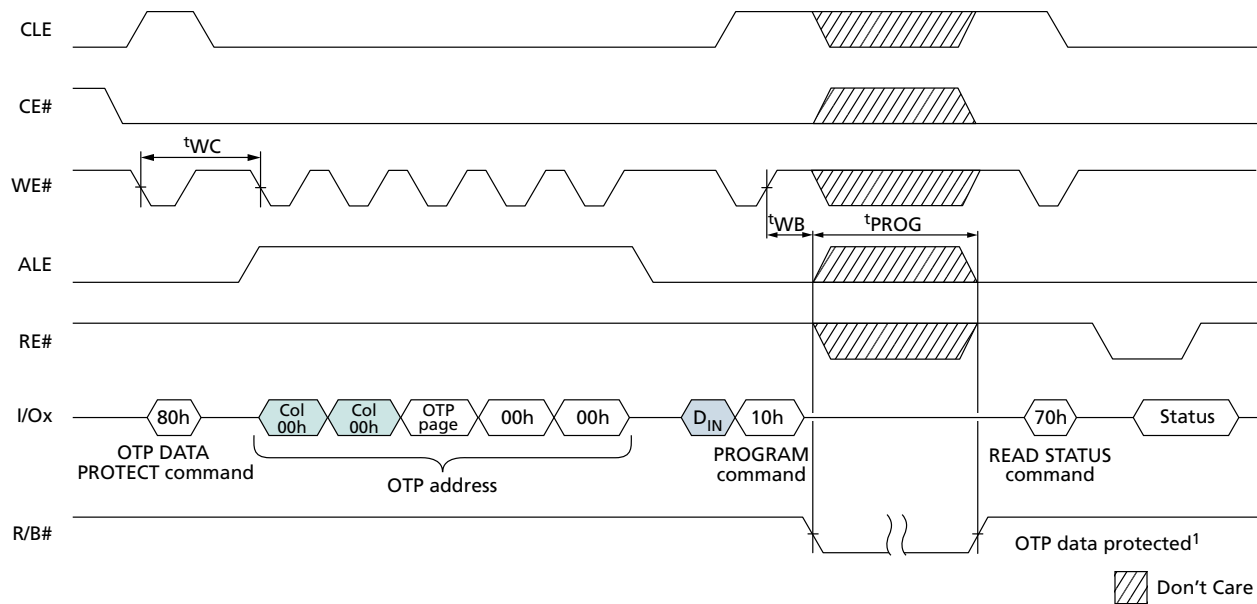
When the device is ready, read bit 0 of the status register to determine whether the operation passed or failed (see Status Operations).





## 4Gb, 8Gb: x8, x16 NAND Flash Memory One-Time Programmable (OTP) Operations

**Figure 67: OTP DATA PROTECT Operation (After Entering OTP Protect Mode)**



Note: 1. OTP data is protected following a good status confirmation.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory One-Time Programmable (OTP) Operations

### OTP DATA READ (00h-30h)

To read data from the OTP area, set the device to OTP operation mode, then issue the PAGE READ (00h-30h) command. Data can be read from OTP pages within the OTP area whether the area is protected or not.

To use the PAGE READ command for reading data from the OTP area, issue the 00h command, and then issue five address cycles: for the first two cycles, the column address; and for the remaining address cycles, select a page in the range of 02h-00h-00h through 1Fh-00h-00h. Lastly, issue the 30h command. The PAGE READ CACHE MODE command is not supported on OTP pages.

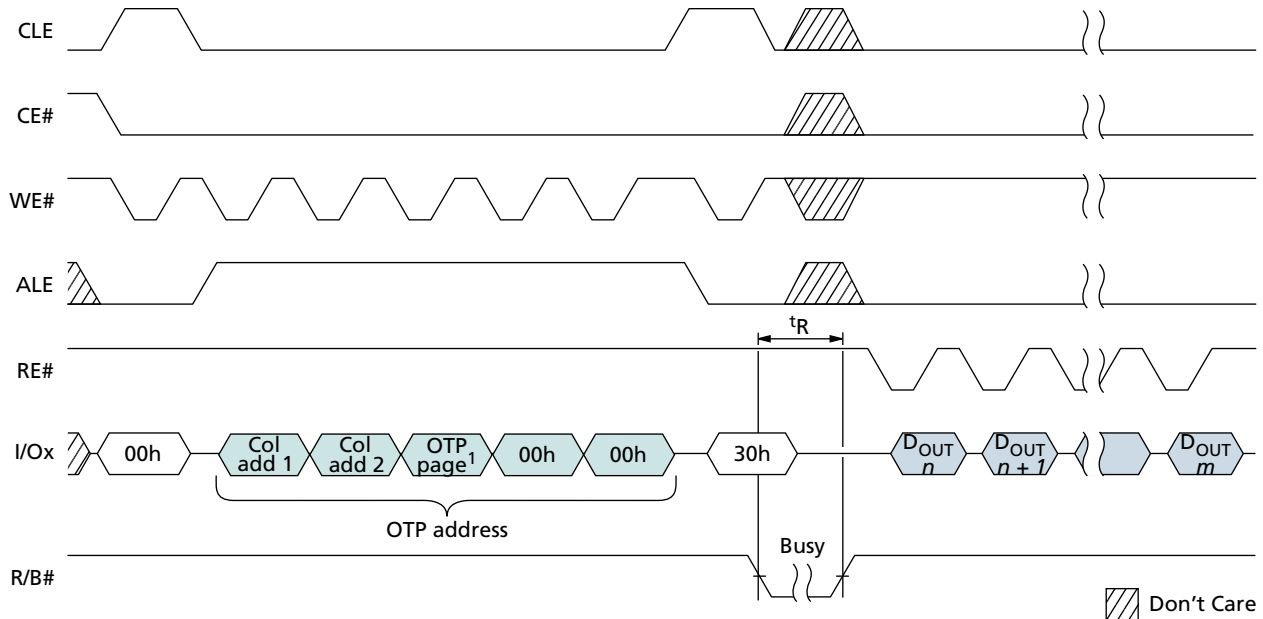
R/B# goes LOW (<sup>t</sup>R) while the data is moved from the OTP page to the data register. The READ STATUS (70h) command is the only valid command for reading status in OTP operation mode. Bit 5 of the status register reflects the state of R/B# (see Status Operations).

Normal READ operation timings apply to OTP read accesses. Additional pages within the OTP area can be selected by repeating the OTP DATA READ command.

The PAGE READ command is compatible with the RANDOM DATA OUTPUT (05h-E0h) command.

Only data on the current page can be read. Pulsing RE# outputs data sequentially.

**Figure 68: OTP DATA READ**

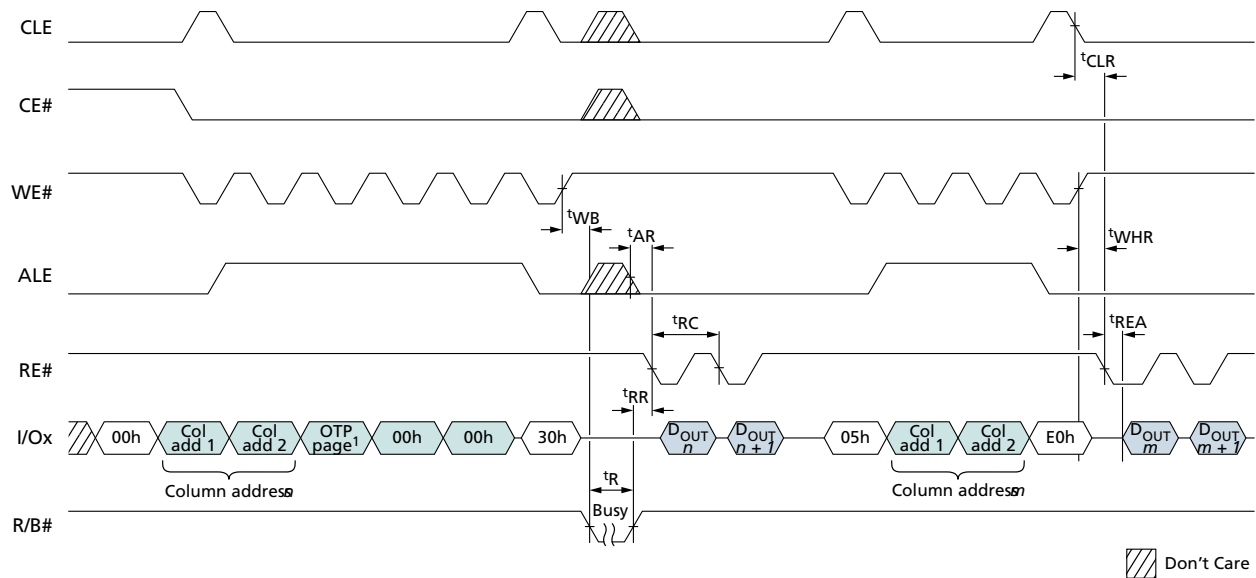


Note: 1. The OTP page must be within the 02h–1Fh range.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory One-Time Programmable (OTP) Operations

**Figure 69: OTP DATA READ with RANDOM DATA READ Operation**



Note: 1. The OTP page must be within the range 02h–1Fh.



## Two-Plane Operations

Each NAND Flash logical unit (LUN) is divided into multiple physical planes. Each plane contains a cache register and a data register independent of the other planes. The planes are addressed via the low-order block address bits. Specific details are provided in Device and Array Organization.

Two-plane operations make better use of the NAND Flash arrays on these physical planes by performing concurrent READ, PROGRAM, or ERASE operations on multiple planes, significantly improving system performance. Two-plane operations must be of the same type across the planes; for example, it is not possible to perform a PROGRAM operation on one plane with an ERASE operation on another.

When issuing two-plane program or erase operations, use the READ STATUS (70h) command and check whether the previous operation(s) failed. If the READ STATUS (70h) command indicates that an error occurred (FAIL = 1 and/or FAILC = 1), use the READ STATUS ENHANCED (78h) command to determine which plane operation failed.

## Two-Plane Addressing

Two-plane commands require multiple, five-cycle addresses, one address per operational plane. For a given two-plane operation, these addresses are subject to the following requirements:

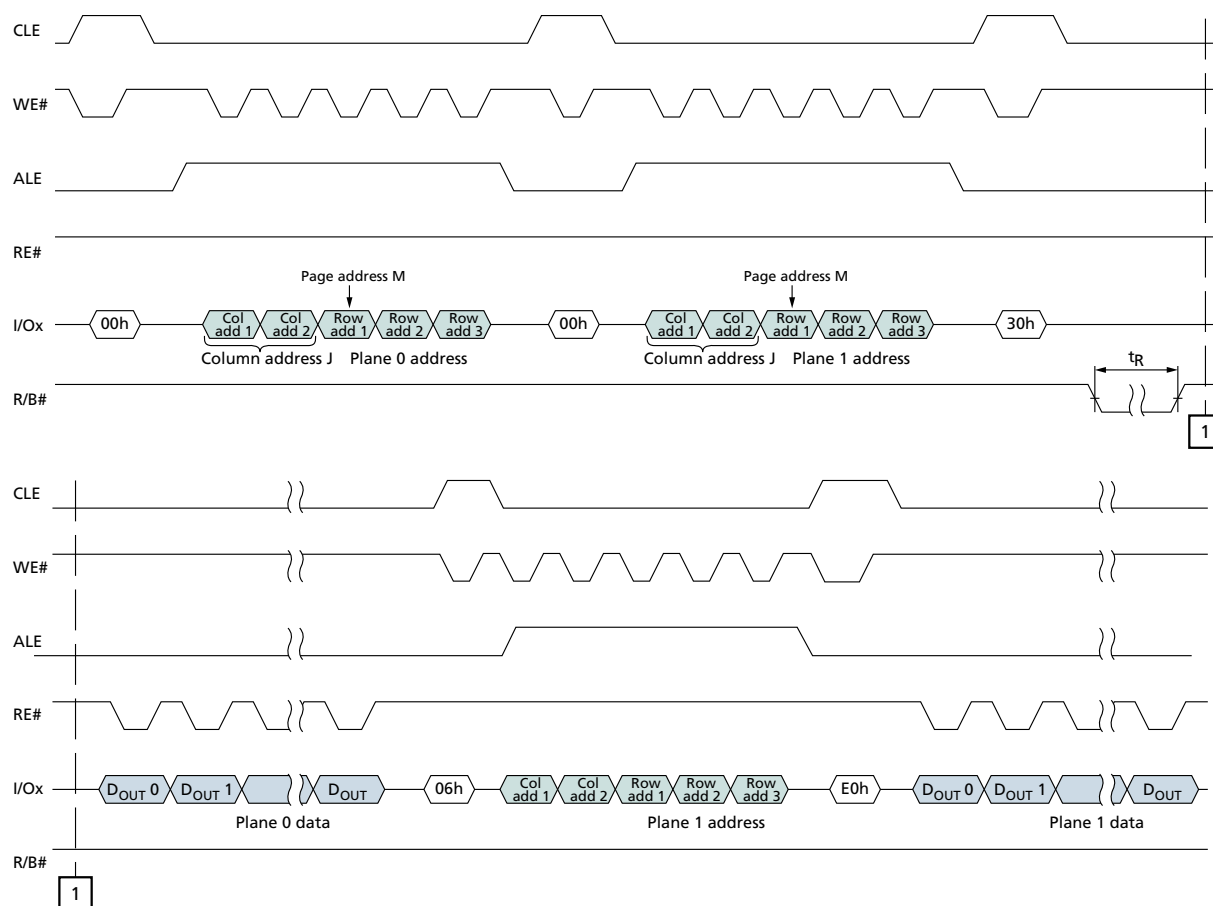
- The LUN address bit(s) must be identical for all of the issued addresses.
- The plane select bit, BA[6], must be different for each issued address.
- The page address bits, PA[5:0], must be identical for each issued address.

The READ STATUS (70h) command should be used following two-plane program page and erase block operations on a single die (LUN).



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Two-Plane Operations

**Figure 70: TWO-PLANE PAGE READ**

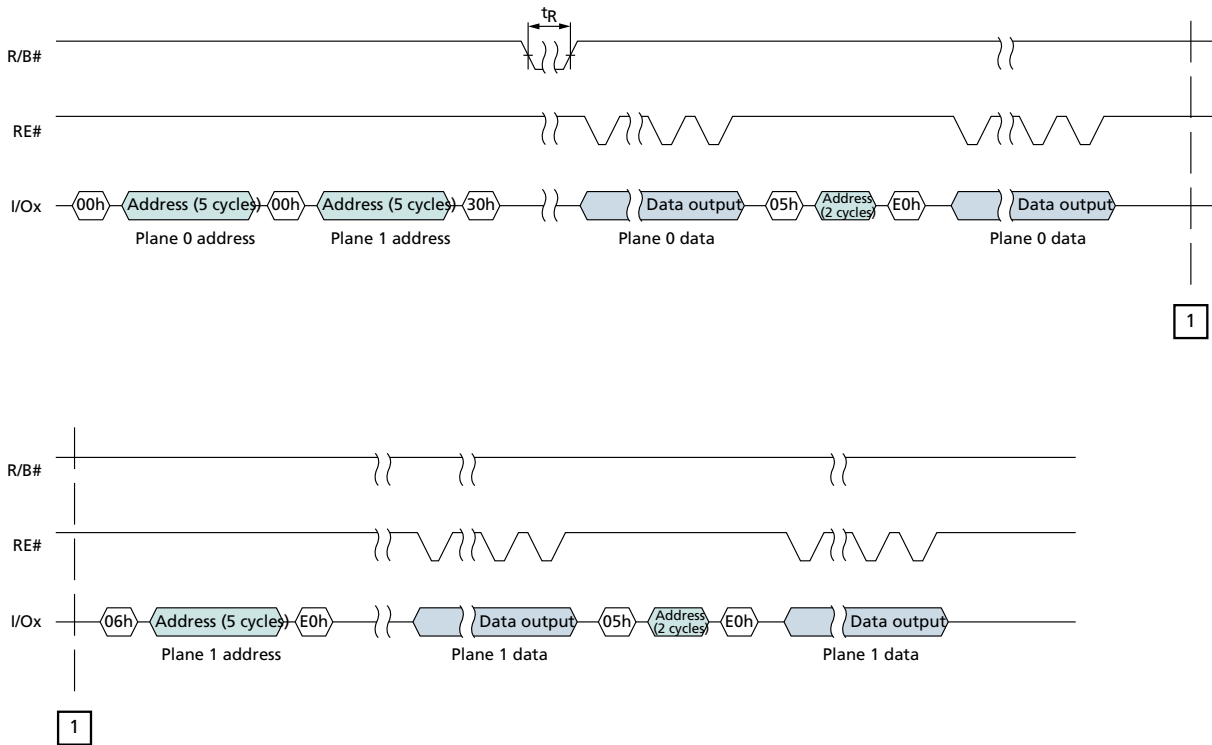


- Notes:
1. Column and page addresses must be the same.
  2. The least significant block address bit, BA6, must be different for the first- and second-plane addresses.

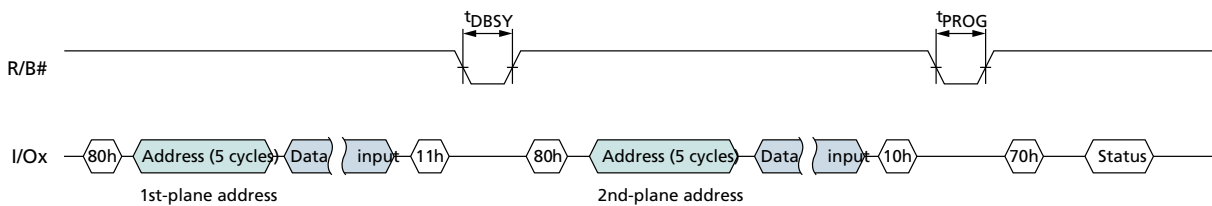


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Two-Plane Operations

**Figure 71: TWO-PLANE PAGE READ with RANDOM DATA READ**



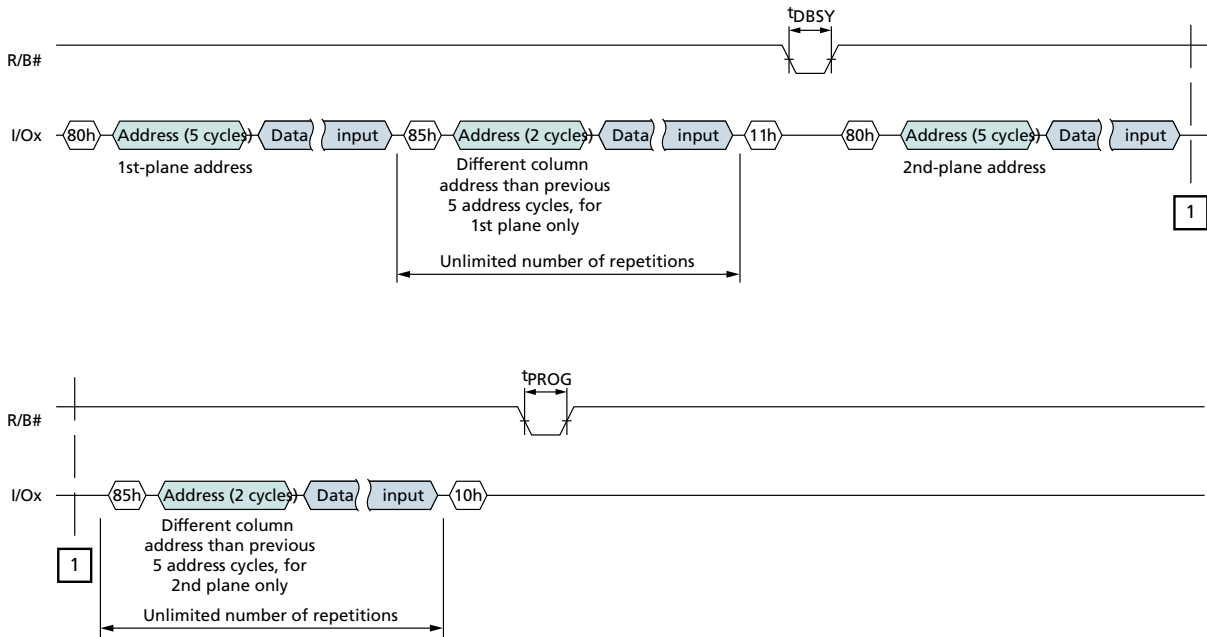
**Figure 72: TWO-PLANE PROGRAM PAGE**





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Two-Plane Operations

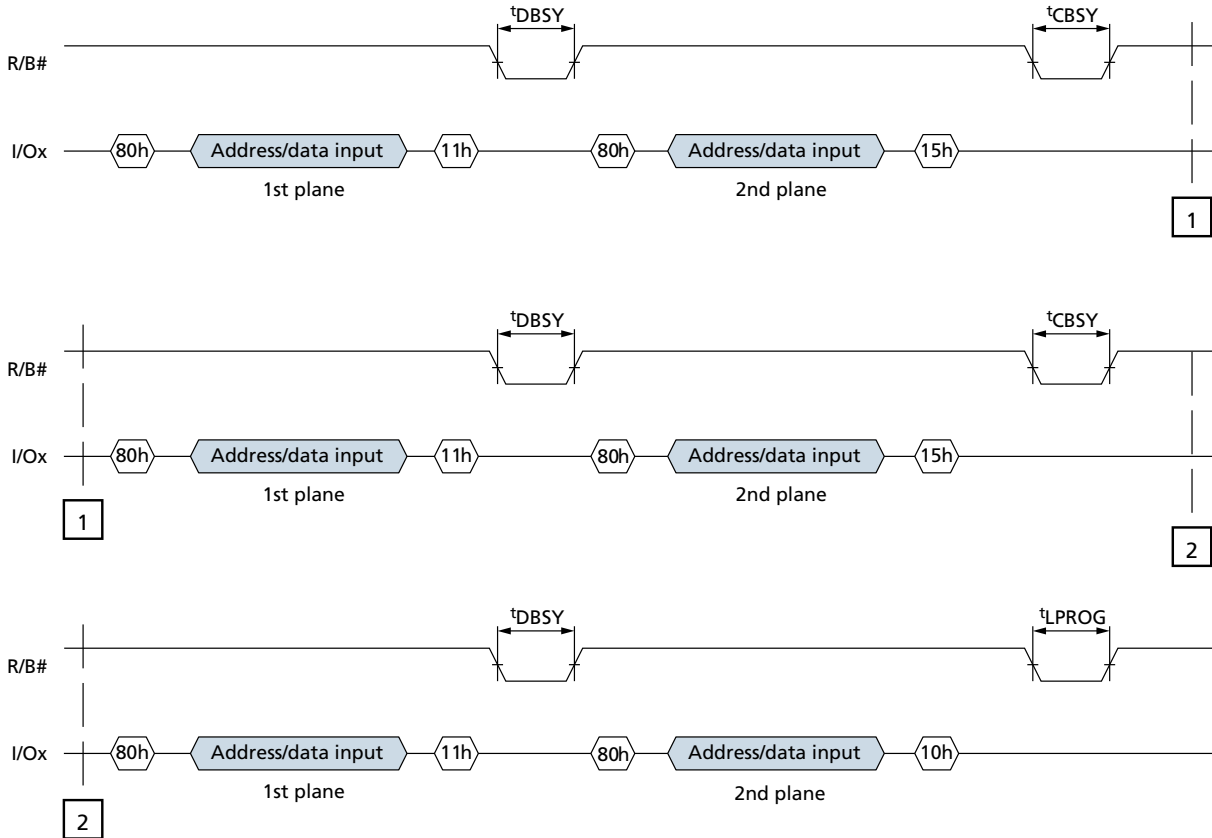
**Figure 73: TWO-PLANE PROGRAM PAGE with RANDOM DATA INPUT**





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Two-Plane Operations

**Figure 74: TWO-PLANE PROGRAM PAGE CACHE MODE**

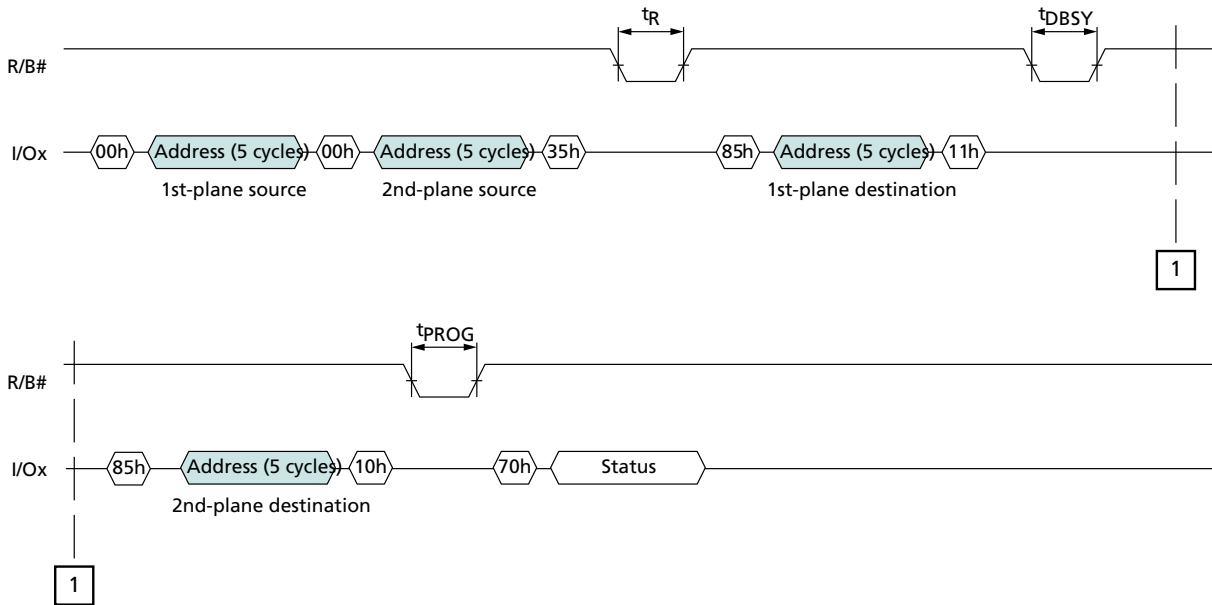






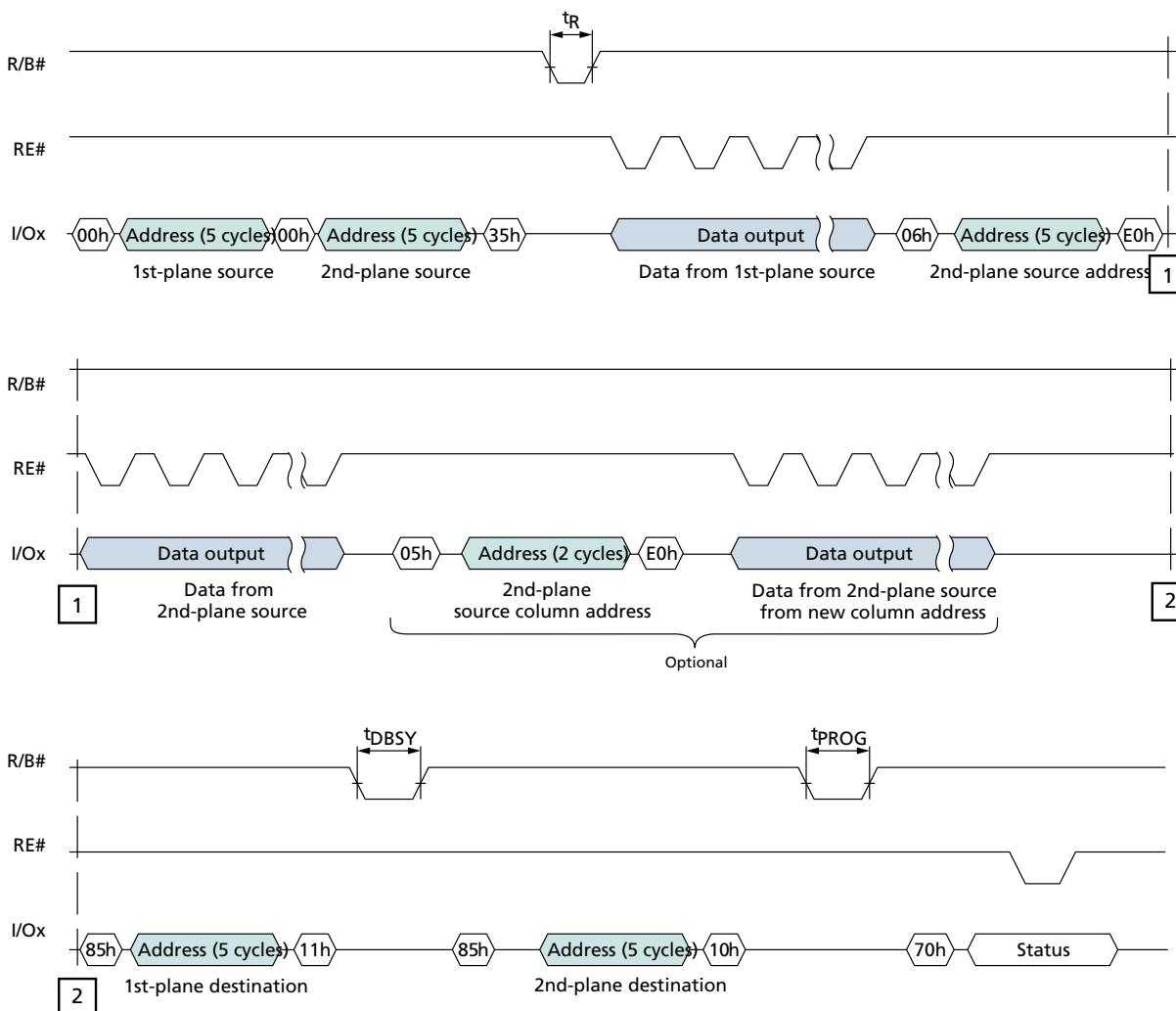
## 4Gb, 8Gb: x8, x16 NAND Flash Memory Two-Plane Operations

**Figure 75: TWO-PLANE INTERNAL DATA MOVE**



## 4Gb, 8Gb: x8, x16 NAND Flash Memory

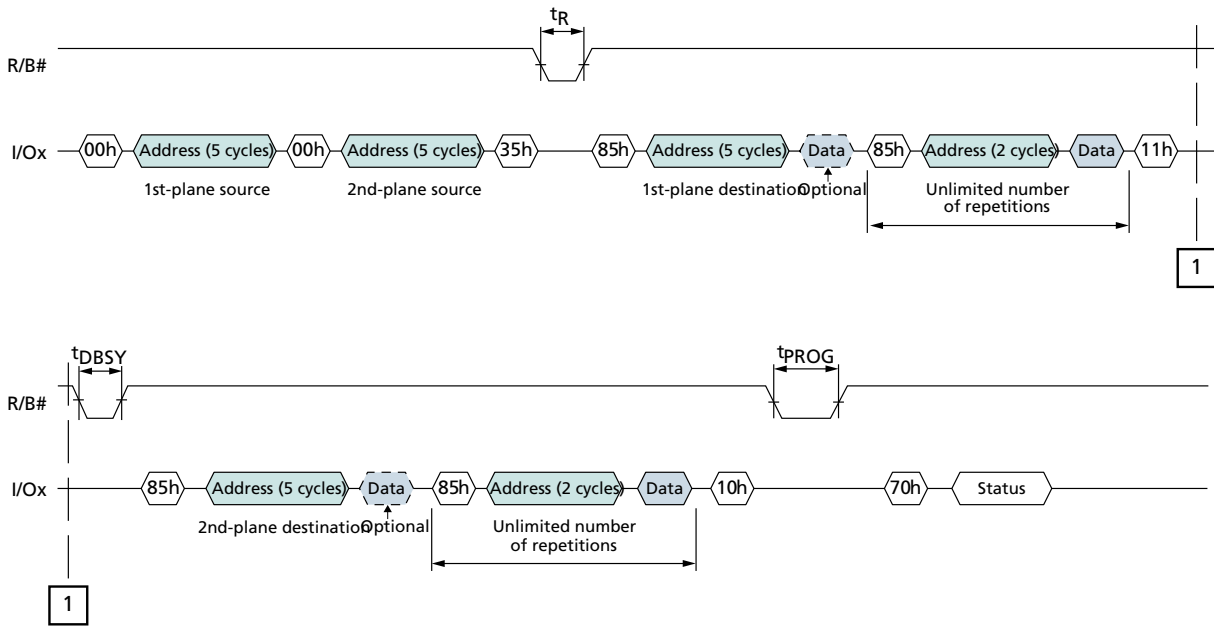
**Figure 76: TWO-PLANE INTERNAL DATA MOVE with TWO-PLANE RANDOM DATA READ**





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Two-Plane Operations

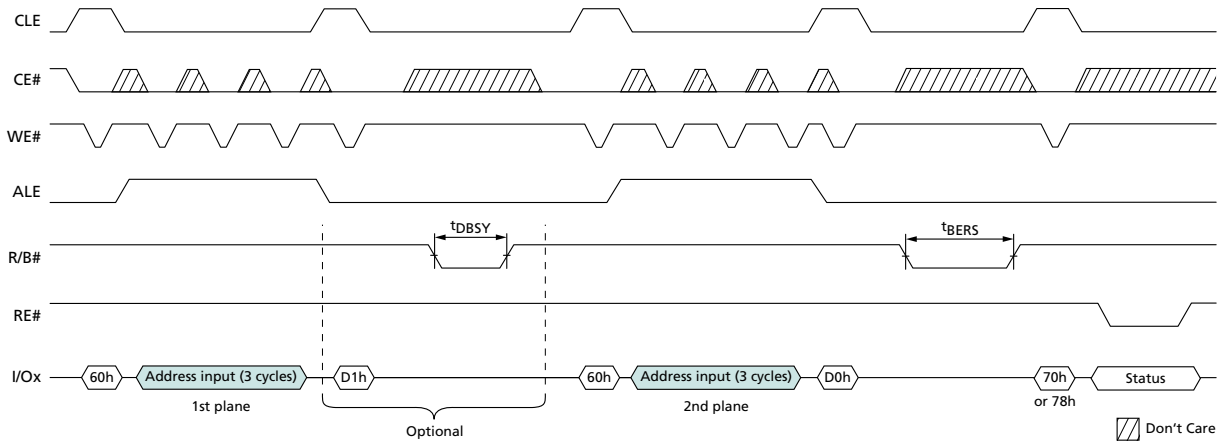
**Figure 77: TWO-PLANE INTERNAL DATA MOVE with RANDOM DATA INPUT**



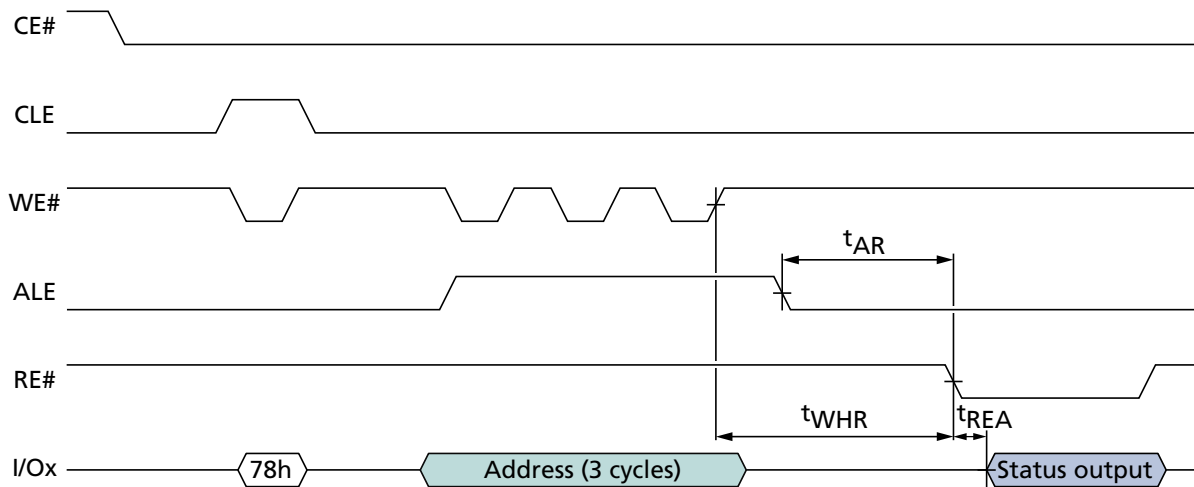


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Two-Plane Operations

**Figure 78: TWO-PLANE BLOCK ERASE**



**Figure 79: TWO-PLANE/MULTIPLE-DIE READ STATUS Cycle**





## Interleaved Die (Multi-LUN) Operations

In devices that have more than one die (LUN) per target, it is possible to improve performance by interleaving operations between the die (LUNs). An interleaved die (multi-LUN) operation is one that is issued to an idle die (LUN) (RDY = 1) while another die (LUN) is busy (RDY = 0).

Interleaved die (multi-LUN) operations are prohibited following RESET (FFh), identification (90h, ECh, EDh), and configuration (EEh, EFh) operations until ARDY = 1 for all of the die (LUNs) on the target.

During an interleaved die (multi-LUN) operation, there are two methods to determine operation completion. The R/B# signal indicates when all of the die (LUNs) have finished their operations. R/B# remains LOW while any die (LUN) is busy. When R/B# goes HIGH, all of the die (LUNs) are idle and the operations are complete. Alternatively, the READ STATUS ENHANCED (78h) command can report the status of each die (LUN) individually.

If a die (LUN) is performing a cache operation, like PROGRAM PAGE CACHE (80h-15h), then the die (LUN) is able to accept the data for another cache operation when status register bit 6 is 1. All operations, including cache operations, are complete on a die when status register bit 5 is 1.

During and following interleaved die (multi-LUN) operations, the READ STATUS (70h) command is prohibited. Instead, use the READ STATUS ENHANCED (78h) command to monitor status. This command selects which die (LUN) will report status. When two-plane commands are used with interleaved die (multi-LUN) operations, the two-plane commands must also meet the requirements in Two-Plane Operations.

See Command Definitions for the list of commands that can be issued while other die (LUNs) are busy.

During an interleaved die (multi-LUN) operation that involves a PROGRAM series (80h-10h, 80h-15h) operation and a READ operation, the PROGRAM series operation must be issued before the READ series operation. The data from the READ series operation must be output to the host before the next PROGRAM series operation is issued. This is because the 80h command clears the cache register contents of all cache registers on all planes.



## Error Management

Each NAND Flash die (LUN) is specified to have a minimum number of valid blocks (NVB) of the total available blocks. This means the die (LUNs) could have blocks that are invalid when shipped from the factory. An invalid block is one that contains at least one page that has more bad bits than can be corrected by the minimum required ECC. Additional blocks can develop with use. However, the total number of available blocks per die (LUN) will not fall below NVB during the endurance life of the product.

Although NAND Flash memory devices could contain bad blocks, they can be used quite reliably in systems that provide bad block management and error-correction algorithms. This type of software environment ensures data integrity.

Internal circuitry isolates each block from other blocks, so the presence of a bad block does not affect the operation of the rest of the NAND Flash array.

NAND Flash devices are shipped from the factory erased. The factory identifies invalid blocks before shipping by attempting to program the bad block mark into every location in the first page of each invalid block. It may not be possible to program every location with the bad block mark. However, the first spare area location in each bad block is guaranteed to contain the bad block mark. This method is compliant with ONFI Factory Defect Mapping requirements. See the following table for the first spare area location and the bad block mark.

System software should check the first spare area location on the first page of each block prior to performing any PROGRAM or ERASE operations on the NAND Flash device. A bad block table can then be created, enabling system software to map around these areas. Factory testing is performed under worst-case conditions. Because invalid blocks could be marginal, it may not be possible to recover this information if the block is erased.

Over time, some memory locations may fail to program or erase properly. In order to ensure that data is stored properly over the life of the NAND Flash device, the following precautions are required:

- Always check status after a PROGRAM or ERASE operation
- Under typical conditions, use the minimum required ECC (see table below)
- Use bad block management and wear-leveling algorithms

The first block (physical block address 00h) for each CE# is guaranteed to be valid with ECC when shipped from the factory.

**Table 21: Error Management Details**

| Description                                  | Requirement                     |
|--|---------------------------------|
| Minimum number of valid blocks (NVB) per LUN | 4016                            |
| Total available blocks per LUN               | 4096                            |
| First spare area location                    | x8: byte 2048<br>x16: word 1024 |
| Bad-block mark                               | x8: 00h<br>x16: 0000h           |
| Minimum required ECC                         | 4-bit ECC per 528 bytes         |

**4Gb, 8Gb: x8, x16 NAND Flash Memory  
Error Management****Table 21: Error Management Details (Continued)**

| Description   | Requirement   |
|---|---|
| Minimum ECC with internal ECC enabled                                       | 4-bit ECC per 516 bytes (user data) + 8 bytes (parity data) |
| Minimum required ECC for block 0 if PROGRAM/ERASE cycles are less than 1000 | 1-bit ECC per 528 bytes                                     |



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Internal ECC and Spare Area Mapping for ECC

### Internal ECC and Spare Area Mapping for ECC

Internal ECC enables 5-bit detection and 4-bit error correction in 512 bytes (x8) or 256 words (x16) of the main area and 4 bytes (x8) or 2 words (x16) of metadata I in the spare area. The metadata II area, which consists of two bytes (x8) and one word (x16), is not ECC protected. During the busy time for PROGRAM operations, internal ECC generates parity bits when error detection is complete.

During READ operations the device executes the internal ECC engine (5-bit detection and 4-bit error correction). When the READ operation is complete, read status bit 0 must be checked to determine whether errors larger than four bits have occurred.

Following the READ STATUS command, the device must be returned to read mode by issuing the 00h command.

Limitations of internal ECC include the spare area, defined in the figures below, and ECC parity areas that cannot be written to. Each ECC user area (referred to as main and spare) must be written within one partial-page program so that the NAND device can calculate the proper ECC parity. The number of partial-page programs within a page cannot exceed four.

**Figure 80: Spare Area Mapping (x8)**

| Max Byte Address | Min Byte Address | ECC Protected | Area    | Description          |
|------------------|------------------|---------------|---------|----------------------|
| 1FFh             | 000h             | Yes           | Main 0  | User data            |
| 3FFh             | 200h             | Yes           | Main 1  | User data            |
| 5FFh             | 400h             | Yes           | Main 2  | User data            |
| 7FFh             | 600h             | Yes           | Main 3  | User data            |
| 801h             | 800h             | No            |         | Reserved             |
| 803h             | 802h             | No            |         | User metadata II     |
| 807h             | 804h             | Yes           | Spare 0 | User metadata I      |
| 80Fh             | 808h             | Yes           | Spare 0 | ECC for main/spare 0 |
| 811h             | 810h             | No            |         | Reserved             |
| 813h             | 812h             | No            |         | User metadata II     |
| 817h             | 814h             | Yes           | Spare 1 | User metadata I      |
| 81Fh             | 818h             | Yes           | Spare 1 | ECC for main/spare 1 |
| 821h             | 820h             | No            |         | Reserved             |
| 823h             | 822h             | No            |         | User metadata II     |
| 827h             | 824h             | Yes           | Spare 2 | User metadata I      |
| 82Fh             | 828h             | Yes           | Spare 2 | ECC for main/spare 2 |
| 831h             | 830h             | No            |         | User data            |
| 833h             | 832h             | No            |         | User metadata II     |
| 837h             | 834h             | Yes           | Spare 3 | User metadata I      |
| 83Fh             | 838h             | Yes           | Spare 3 | ECC for main/spare 3 |

| Bad Block Information | ECC Parity | User Data (Metadata) |
|-----------------------|------------|----------------------|
| 2 bytes               | 8 bytes    | 6 bytes              |





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Internal ECC and Spare Area Mapping for ECC

**Figure 81: Spare Area Mapping (x16)**

| Max word Address | Min word Address | ECC Protected | Area    | Description          |
|------------------|------------------|---------------|---------|----------------------|
| 0FFh             | 000h             | Yes           | Main 0  | User data            |
| 1FFh             | 100h             | Yes           | Main 1  | User data            |
| 2FFh             | 200h             | Yes           | Main 2  | User data            |
| 3FFh             | 300h             | Yes           | Main 3  | User data            |
| 400h             | 400h             | No            |         | Reserved             |
| 401h             | 401h             | No            |         | User metadata II     |
| 403h             | 402h             | Yes           | Spare 0 | User metadata I      |
| 407h             | 404h             | Yes           | Spare 0 | ECC for main/spare 0 |
| 408h             | 408h             | No            |         | Reserved             |
| 409h             | 409h             | No            |         | User metadata II     |
| 40Bh             | 40Ah             | Yes           | Spare 1 | User metadata I      |
| 40Fh             | 40Ch             | Yes           | Spare 1 | ECC for main/spare 1 |
| 410h             | 410h             | No            |         | Reserved             |
| 411h             | 411h             | No            |         | User metadata II     |
| 413h             | 412h             | Yes           | Spare 2 | User metadata I      |
| 417h             | 414h             | Yes           | Spare 2 | ECC for main/spare 2 |
| 418h             | 418h             | No            |         | User data            |
| 419h             | 419h             | No            |         | User metadata II     |
| 41Bh             | 41Ah             | Yes           | Spare 3 | User metadata I      |
| 41Fh             | 41Ch             | Yes           | Spare 3 | ECC for main/spare 3 |

| Bad Block Information | ECC Parity | User Data (Metadata) |
|-----------------------|------------|----------------------|
| 1 word                | 4 words    | 3 words              |



## Electrical Specifications

Stresses greater than those listed can cause permanent damage to the device. This is stress rating only, and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not guaranteed. Exposure to absolute maximum rating conditions for extended periods can affect reliability.

**Table 22: Absolute Maximum Ratings**

Voltage on any pin relative to V<sub>SS</sub>

| Parameter/Condition                |      | Symbol           | Min  | Max | Unit |
|------------------------------------|------|------------------|------|-----|------|
| Voltage input                      | 1.8V | V <sub>IN</sub>  | −0.6 | 2.4 | V    |
|                                    | 3.3V |                  | −0.6 | 4.6 | V    |
| V <sub>CC</sub> supply voltage     | 1.8V | V <sub>CC</sub>  | −0.6 | 2.4 | V    |
|                                    | 3.3V |                  | −0.6 | 4.6 | V    |
| Storage temperature                |      | T <sub>STG</sub> | −65  | 150 | °C   |
| Short circuit output current, I/Os |      | –                | –    | 5   | mA   |

**Table 23: Recommended Operating Conditions**

| Parameter/Condition            |            | Symbol          | Min | Typ | Max  | Unit |
|--------------------------------|------------|-----------------|-----|-----|------|------|
| Operating temperature          | Commercial | T <sub>A</sub>  | 0   | –   | 70   | °C   |
|                                | Industrial |                 | –40 | –   | 85   | °C   |
| V <sub>CC</sub> supply voltage | 1.8V       | V <sub>CC</sub> | 1.7 | 1.8 | 1.95 | V    |
|                                | 3.3V       |                 | 2.7 | 3.3 | 3.6  | V    |
| Ground supply voltage          |            | V <sub>SS</sub> | 0   | 0   | 0    | V    |

**Table 24: Valid Blocks**

| Parameter          | Symbol | Device  | Min  | Max  | Unit   | Notes   |
|--------------------|--------|---------|------|------|--------|---------|
| Valid block number | NVB    | MT29F4G | 4016 | 4096 | Blocks | 1, 2    |
|                    |        | MT29F8G | 8032 | 8192 | Blocks | 1, 2, 3 |

- Notes:
- Invalid blocks are blocks that contain one or more bad bits. The device may contain bad blocks upon shipment. Additional bad blocks may develop over time; however, the total number of available blocks will not drop below NVB during the endurance life of the device. Do not erase or program blocks marked invalid by the factory.
  - Block 00h (the first block) is guaranteed to be valid with ECC when shipped from the factory.
  - Each 4Gb section has a maximum of 80 invalid blocks.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Electrical Specifications

**Table 25: Capacitance**

Notes 1–3 apply to all parameters and conditions

| Description                    | Symbol   | Max | Unit |
|--------------------------------|----------|-----|------|
| Input capacitance              | $C_{IN}$ | 10  | pF   |
| Input/output capacitance (I/O) | $C_{IO}$ | 10  | pF   |

- Notes:
1. These parameters are verified in device characterization and are not 100% tested.
  2. Test conditions:  $T_C = 25^\circ\text{C}$ ;  $f = 1\text{ MHz}$ ;  $V_{IN} = 0\text{V}$ .
  3. Capacitance ( $C_{IN} = C_{IO} = 20\text{pF}$ ) for MT29F8G.

**Table 26: Test Conditions**

| Parameter                      |      | Value                                    | Notes |
|--------------------------------|------|--|-------|
| Input pulse levels             |      | 0.0V to $V_{CC}$                         |       |
| Input rise and fall times      | 1.8V | 2.5ns                                    |       |
|                                | 3.3V | 5.0ns                                    |       |
| Input and output timing levels |      | $V_{CC}/2$                               |       |
| Output load                    |      | 1 TTL GATE and $CL = 30\text{pF}$ (1.8V) | 1     |
|                                |      | 1 TTL GATE and $CL = 50\text{pF}$ (3.3V) |       |
| Output load                    |      | 1 TTL GATE and $CL = 30\text{pF}$ (1.8V) | 1     |
|                                |      | 1 TTL GATE and $CL = 50\text{pF}$ (3.3V) |       |

- Note:
1. Verified in device characterization, not 100% tested.



## Electrical Specifications – DC Characteristics and Operating Conditions

**Table 27: DC Characteristics and Operating Conditions (3.3V)**

| Parameter                     | Conditions   | Symbol                   | Min                  | Typ | Max                 | Unit          | Notes |
|-------------------------------|--|--------------------------|----------------------|-----|---------------------|---------------|-------|
| Sequential READ current       | $t_{RC} = t_{RC} \text{ (MIN)}; CE\# = V_{IL}; I_{OUT} = 0\text{mA}$ | $I_{CC1}$                | –                    | 25  | 35                  | mA            |       |
| PROGRAM current               | –  | $I_{CC2}$                | –                    | 25  | 35                  | mA            |       |
| ERASE current                 | –  | $I_{CC3}$                | –                    | 25  | 35                  | mA            |       |
| Standby current (TTL)         | $CE\# = V_{IH}; WP\# = 0V/V_{CC}$                                    | $I_{SB1}$                | –                    | –   | 1                   | mA            |       |
| Standby current (CMOS)        | $CE\# = V_{CC} - 0.2V; WP\# = 0V/V_{CC}$                             | $I_{SB2}$                | –                    | 10  | 50                  | $\mu\text{A}$ |       |
| Staggered power-up current    | Rise time = 1ms<br>Line capacitance = $0.1\mu\text{F}$               | $I_{ST}$                 | –                    | –   | 10 per die          | mA            | 1     |
| Input leakage current         | $V_{IN} = 0V \text{ to } V_{CC}$                                     | $I_{LI}$                 | –                    | –   | $\pm 10$            | $\mu\text{A}$ |       |
| Output leakage current        | $V_{OUT} = 0V \text{ to } V_{CC}$                                    | $I_{LO}$                 | –                    | –   | $\pm 10$            | $\mu\text{A}$ |       |
| Input high voltage            | I/O[7:0], I/O[15:0],<br>CE#, CLE, ALE, WE#, RE#,<br>WP#              | $V_{IH}$                 | $0.8 \times V_{CC}$  | –   | $V_{CC} + 0.3$      | V             |       |
| Input low voltage, all inputs | –  | $V_{IL}$                 | –0.3                 | –   | $0.2 \times V_{CC}$ | V             |       |
| Output high voltage           | $I_{OH} = -400\mu\text{A}$   | $V_{OH}$                 | $0.67 \times V_{CC}$ | –   | –                   | V             | 3     |
| Output low voltage            | $I_{OL} = 2.1\text{mA}$  | $V_{OL}$                 | –                    | –   | 0.4                 | V             | 3     |
| Output low current            | $V_{OL} = 0.4V$  | $I_{OL} \text{ (R/B\#)}$ | 8                    | 10  | –                   | mA            | 2     |

- Notes:
1. Measurement is taken with 1ms averaging intervals and begins after  $V_{CC}$  reaches  $V_{CC}(\text{MIN})$ .
  2.  $I_{OL} \text{ (R/B\#)}$  may need to be relaxed if R/B pull-down strength is not set to full.
  3.  $V_{OH}$  and  $V_{OL}$  may need to be relaxed if I/O drive strength is not set to full.



# 4Gb, 8Gb: x8, x16 NAND Flash Memory

## Electrical Specifications – DC Characteristics and Operating Conditions

**Table 28: DC Characteristics and Operating Conditions (1.8V)**

| Parameter                     | Conditions   | Symbol                 | Min                 | Typ | Max                 | Unit          | Notes |
|-------------------------------|--|------------------------|---------------------|-----|---------------------|---------------|-------|
| Sequential READ current       | $t_{RC} = t_{RC}(\text{MIN}); CE\# = V_{IL}; I_{OUT} = 0\text{mA}$ | $I_{CC1}$              | –                   | 13  | 20                  | mA            | 1, 2  |
| PROGRAM current               | –  | $I_{CC2}$              | –                   | 10  | 20                  | mA            | 1, 2  |
| ERASE current                 | –  | $I_{CC3}$              | –                   | 10  | 20                  | mA            | 1, 2  |
| Standby current (TTL)         | $CE\# = V_{IH}; WP\# = 0V/V_{CC}$                                  | $I_{SB1}$              | –                   | –   | 1                   | mA            |       |
| Standby current (CMOS)        | $CE\# = V_{CC} - 0.2V; WP\# = 0V/V_{CC}$                           | $I_{SB2}$              | –                   | 10  | 50                  | $\mu\text{A}$ |       |
| Staggered power-up current    | Rise time = 1ms<br>Line capacitance = 0.1 $\mu\text{F}$            | $I_{ST}$               | –                   | –   | 10 per die          | mA            | 3     |
| Input leakage current         | $V_{IN} = 0V \text{ to } V_{CC}$                                   | $I_{LI}$               | –                   | –   | $\pm 10$            | $\mu\text{A}$ |       |
| Output leakage current        | $V_{OUT} = 0V \text{ to } V_{CC}$                                  | $I_{LO}$               | –                   | –   | $\pm 10$            | $\mu\text{A}$ |       |
| Input high voltage            | I/O[7:0], I/O[15:0],<br>CE#, CLE, ALE, WE#, RE#,<br>WP#            | $V_{IH}$               | $0.8 \times V_{CC}$ | –   | $V_{CC} + 0.3$      | V             |       |
| Input low voltage, all inputs | –  | $V_{IL}$               | –0.3                | –   | $0.2 \times V_{CC}$ | V             |       |
| Output high voltage           | $I_{OH} = -100\mu\text{A}$   | $V_{OH}$               | $V_{CC} - 0.1$      | –   | –                   | V             | 4     |
| Output low voltage            | $I_{OL} = +100\mu\text{A}$   | $V_{OL}$               | –                   | –   | 0.1                 | V             | 4     |
| Output low current (R/B#)     | $V_{OL} = 0.2V$  | $I_{OL}(\text{R/B}\#)$ | 3                   | 4   | –                   | mA            | 5     |

- Notes:
1. Typical and maximum values are for single-plane operation only. If device supports dual-plane operation, values are 20mA (TYP) and 40mA (MAX).
  2. Values are for single-die operations. Values could be higher for interleaved-die operations.
  3. Measurement is taken with 1ms averaging intervals and begins after  $V_{CC}$  reaches  $V_{CC}(\text{MIN})$ .
  4. Test conditions for  $V_{OH}$  and  $V_{OL}$ .
  5. DC characteristics may need to be relaxed if R/B# pull-down strength is not set to full.



## Electrical Specifications – AC Characteristics and Operating Conditions

**Table 29: AC Characteristics: Command, Data, and Address Input (3.3V)**

Note 1 applies to all

| Parameter                 | Symbol    | Min | Max | Unit | Notes |
|---------------------------|-----------|-----|-----|------|-------|
| ALE to data start         | $t_{ADL}$ | 70  | –   | ns   | 2     |
| ALE hold time             | $t_{ALH}$ | 5   | –   | ns   |       |
| ALE setup time            | $t_{ALS}$ | 10  | –   | ns   |       |
| CE# hold time             | $t_{CH}$  | 5   | –   | ns   |       |
| CLE hold time             | $t_{CLH}$ | 5   | –   | ns   |       |
| CLE setup time            | $t_{CLS}$ | 10  | –   | ns   |       |
| CE# setup time            | $t_{CS}$  | 15  | –   | ns   |       |
| Data hold time            | $t_{DH}$  | 5   | –   | ns   |       |
| Data setup time           | $t_{DS}$  | 7   | –   | ns   |       |
| WRITE cycle time          | $t_{WC}$  | 20  | –   | ns   | 2     |
| WE# pulse width HIGH      | $t_{WH}$  | 7   | –   | ns   | 2     |
| WE# pulse width           | $t_{WP}$  | 10  | –   | ns   | 2     |
| WP# transition to WE# LOW | $t_{WW}$  | 100 | –   | ns   |       |

- Notes:
1. Operating mode timings meet ONFI timing mode 5 parameters.
  2. Timing for  $t_{ADL}$  begins in the address cycle, on the final rising edge of WE#, and ends with the first rising edge of WE# for data input.

**Table 30: AC Characteristics: Command, Data, and Address Input (1.8V)**

Note 1 applies to all

| Parameter                 | Symbol    | Min | Max | Unit | Notes |
|---------------------------|-----------|-----|-----|------|-------|
| ALE to data start         | $t_{ADL}$ | 70  | –   | ns   | 2     |
| ALE hold time             | $t_{ALH}$ | 5   | –   | ns   |       |
| ALE setup time            | $t_{ALS}$ | 10  | –   | ns   |       |
| CE# hold time             | $t_{CH}$  | 5   | –   | ns   |       |
| CLE hold time             | $t_{CLH}$ | 5   | –   | ns   |       |
| CLE setup time            | $t_{CLS}$ | 10  | –   | ns   |       |
| CE# setup time            | $t_{CS}$  | 20  | –   | ns   |       |
| Data hold time            | $t_{DH}$  | 5   | –   | ns   |       |
| Data setup time           | $t_{DS}$  | 10  | –   | ns   |       |
| WRITE cycle time          | $t_{WC}$  | 25  | –   | ns   | 2     |
| WE# pulse width HIGH      | $t_{WH}$  | 10  | –   | ns   | 2     |
| WE# pulse width           | $t_{WP}$  | 12  | –   | ns   | 2     |
| WP# transition to WE# LOW | $t_{WW}$  | 100 | –   | ns   |       |

- Notes:
1. Operating mode timings meet ONFI timing mode 4 parameters.
  2. Timing for  $t_{ADL}$  begins in the address cycle on the final rising edge of WE#, and ends with the first rising edge of WE# for data input.



## 4Gb, 8Gb: x8, x16 NAND Flash Memory

### Electrical Specifications – AC Characteristics and Operating Conditions

**Table 31: AC Characteristics: Normal Operation (3.3V)**

Note 1 applies to all

| Parameter                       | Symbol     | Min | Max      | Unit    | Notes |
|---------------------------------|------------|-----|----------|---------|-------|
| ALE to RE# delay                | $t_{AR}$   | 10  | –        | ns      |       |
| CE# access time                 | $t_{CEA}$  | –   | 25       | ns      |       |
| CE# HIGH to output High-Z       | $t_{CHZ}$  | –   | 50       | ns      | 2     |
| CLE to RE# delay                | $t_{CLR}$  | 10  | –        | ns      |       |
| CE# HIGH to output hold         | $t_{COH}$  | 15  | –        | ns      |       |
| Output High-Z to RE# LOW        | $t_{IR}$   | 0   | –        | ns      |       |
| READ cycle time                 | $t_{RC}$   | 20  | –        | ns      |       |
| RE# access time                 | $t_{REA}$  | –   | 16       | ns      |       |
| RE# HIGH hold time              | $t_{REH}$  | 7   | –        | ns      |       |
| RE# HIGH to output hold         | $t_{RHOH}$ | 15  | –        | ns      |       |
| RE# HIGH to WE# LOW             | $t_{RHW}$  | 100 | –        | ns      |       |
| RE# HIGH to output High-Z       | $t_{RHZ}$  | –   | 100      | ns      | 2     |
| RE# LOW to output hold          | $t_{RLOH}$ | 5   | –        | ns      |       |
| RE# pulse width                 | $t_{RP}$   | 10  | –        | ns      |       |
| Ready to RE# LOW                | $t_{RR}$   | 20  | –        | ns      |       |
| Reset time (READ/PROGRAM/ERASE) | $t_{RST}$  | –   | 5/10/500 | $\mu$ s | 3     |
| WE# HIGH to busy                | $t_{WB}$   | –   | 100      | ns      |       |
| WE# HIGH to RE# LOW             | $t_{WHR}$  | 60  | –        | ns      |       |

- Notes:
1. AC characteristics may need to be relaxed if I/O drive strength is not set to full.
  2. Transition is measured  $\pm 200$ mV from steady-state voltage with load. This parameter is sampled and not 100% tested.
  3. The first time the RESET (FFh) command is issued while the device is idle, the device will go busy for a maximum of 1ms. Thereafter, the device goes busy for a maximum of 5 $\mu$ s.

**Table 32: AC Characteristics: Normal Operation (1.8V)**

Note 1 applies to all

| Parameter                 | Symbol     | Min | Max | Unit | Notes |
|---------------------------|------------|-----|-----|------|-------|
| ALE to RE# delay          | $t_{AR}$   | 10  | –   | ns   |       |
| CE# access time           | $t_{CEA}$  | –   | 25  | ns   |       |
| CE# HIGH to output High-Z | $t_{CHZ}$  | –   | 50  | ns   | 2     |
| CLE to RE# delay          | $t_{CLR}$  | 10  | –   | ns   |       |
| CE# HIGH to output hold   | $t_{COH}$  | 15  | –   | ns   |       |
| Output High-Z to RE# LOW  | $t_{IR}$   | 0   | –   | ns   |       |
| READ cycle time           | $t_{RC}$   | 25  | –   | ns   |       |
| RE# access time           | $t_{REA}$  | –   | 22  | ns   |       |
| RE# HIGH hold time        | $t_{REH}$  | 10  | –   | ns   |       |
| RE# HIGH to output hold   | $t_{RHOH}$ | 15  | –   | ns   |       |
| RE# HIGH to WE# LOW       | $t_{RHW}$  | 100 | –   | ns   |       |



# 4Gb, 8Gb: x8, x16 NAND Flash Memory

## Electrical Specifications – AC Characteristics and Operating Conditions

**Table 32: AC Characteristics: Normal Operation (1.8V) (Continued)**

Note 1 applies to all

| Parameter                       | Symbol     | Min | Max      | Unit    | Notes |
|---------------------------------|------------|-----|----------|---------|-------|
| RE# HIGH to output High-Z       | $t_{RHZ}$  | –   | 65       | ns      | 2     |
| RE# LOW to output hold          | $t_{RLOH}$ | 3   | –        | ns      |       |
| RE# pulse width                 | $t_{RP}$   | 12  | –        | ns      |       |
| Ready to RE# LOW                | $t_{RR}$   | 20  | –        | ns      |       |
| Reset time (READ/PROGRAM/ERASE) | $t_{RST}$  | –   | 5/10/500 | $\mu$ s | 3     |
| WE# HIGH to busy                | $t_{WB}$   | –   | 100      | ns      |       |
| WE# HIGH to RE# LOW             | $t_{WHR}$  | 80  | –        | ns      |       |

- Notes:
1. AC characteristics may need to be relaxed if I/O drive strength is not set to full.
  2. Transition is measured  $\pm 200$ mV from steady-state voltage with load. This parameter is sampled and not 100% tested.
  3. The first time the RESET (FFh) command is issued while the device is idle, the device will be busy for a maximum of 1ms. Thereafter, the device is busy for a maximum of 5 $\mu$ s.





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Electrical Specifications – Program/Erase Characteristics

### Electrical Specifications – Program/Erase Characteristics

**Table 33: Program/Erase Characteristics**

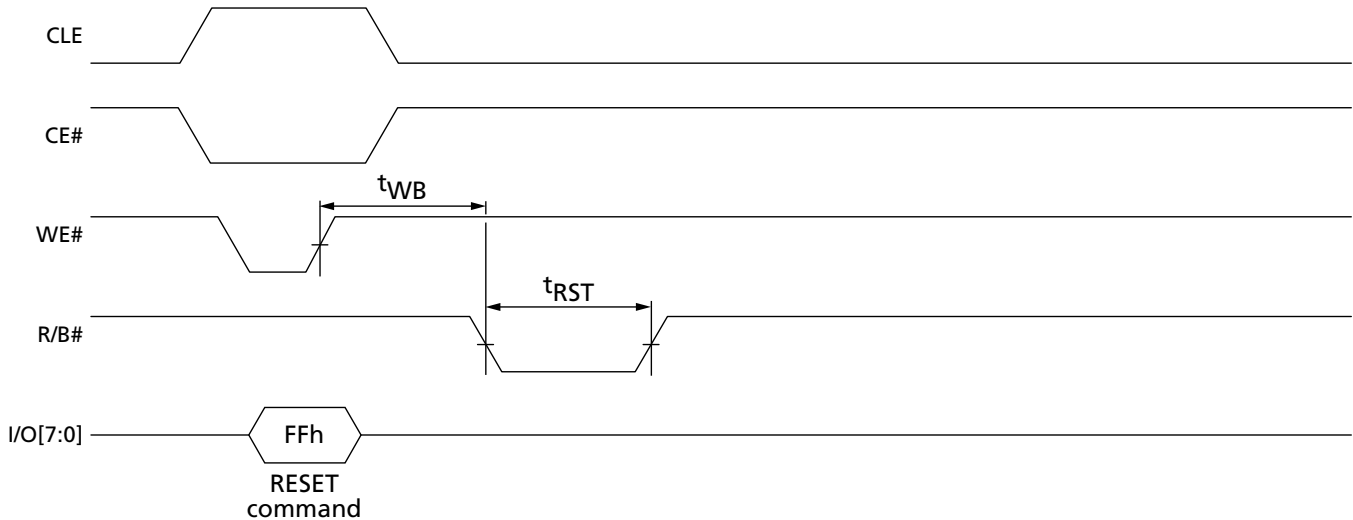
| Parameter  | Symbol          | Typ | Max | Unit    | Notes |
|--|-----------------|-----|-----|---------|-------|
| Number of partial-page programs  | NOP             | –   | 4   | cycles  | 1     |
| BLOCK ERASE operation time   | $t_{BERS}$      | 0.7 | 3   | ms      |       |
| Busy time for PROGRAM CACHE operation  | $t_{CBSY}$      | 3   | 600 | $\mu s$ | 2     |
| Cache read busy time   | $t_{RCBSY}$     | 3   | 25  | $\mu s$ |       |
| Busy time for SET FEATURES and GET FEATURES operations                             | $t_{FEAT}$      | –   | 1   | $\mu s$ |       |
| Busy time for OTP DATA PROGRAM operation if OTP is protected                       | $t_{OBSY}$      | –   | 30  | $\mu s$ |       |
| Busy time for PROGRAM/ERASE on locked blocks                                       | $t_{LBSY}$      | –   | 3   | $\mu s$ |       |
| PROGRAM PAGE operation time, internal ECC disabled                                 | $t_{PROG}$      | 200 | 600 | $\mu s$ | 8     |
| PROGRAM PAGE operation time, internal ECC enabled                                  | $t_{PROG\_ECC}$ | 220 | 600 | $\mu s$ | 3, 8  |
| Data transfer from Flash array to data register, internal ECC disabled             | $t_R$           | –   | 25  | $\mu s$ | 6, 7  |
| Data transfer from Flash array to data register, internal ECC enabled              | $t_{R\_ECC}$    | 45  | 70  | $\mu s$ | 3, 5  |
| Busy time for OTP DATA PROGRAM operation if OTP is protected, internal ECC enabled | $t_{OBSY\_ECC}$ | –   | 50  | $\mu s$ |       |
| Busy time for TWO-PLANE PROGRAM PAGE or TWO-PLANE BLOCK ERASE operation            | $t_{DBSY}$      | 0.5 | 1   | $\mu s$ |       |

- Notes:
1. Four total partial-page programs to the same page. If ECC is enabled, then the device is limited to one partial-page program per ECC user area, not exceeding four partial-page programs per page.
  2.  $t_{CBSY}$  MAX time depends on timing between internal program completion and data-in.
  3. Parameters are with internal ECC enabled.
  4. Typical is nominal voltage and room temperature.
  5. Typical  $t_{R\_ECC}$  is under typical process corner, nominal voltage, and at room temperature.
  6. Data transfer from Flash array to data register with internal ECC disabled.
  7. AC characteristics may need to be relaxed if I/O drive strength is not set to full.
  8. Typical program time is defined as the time within which more than 50% of the pages are programmed at nominal voltage and room temperature.

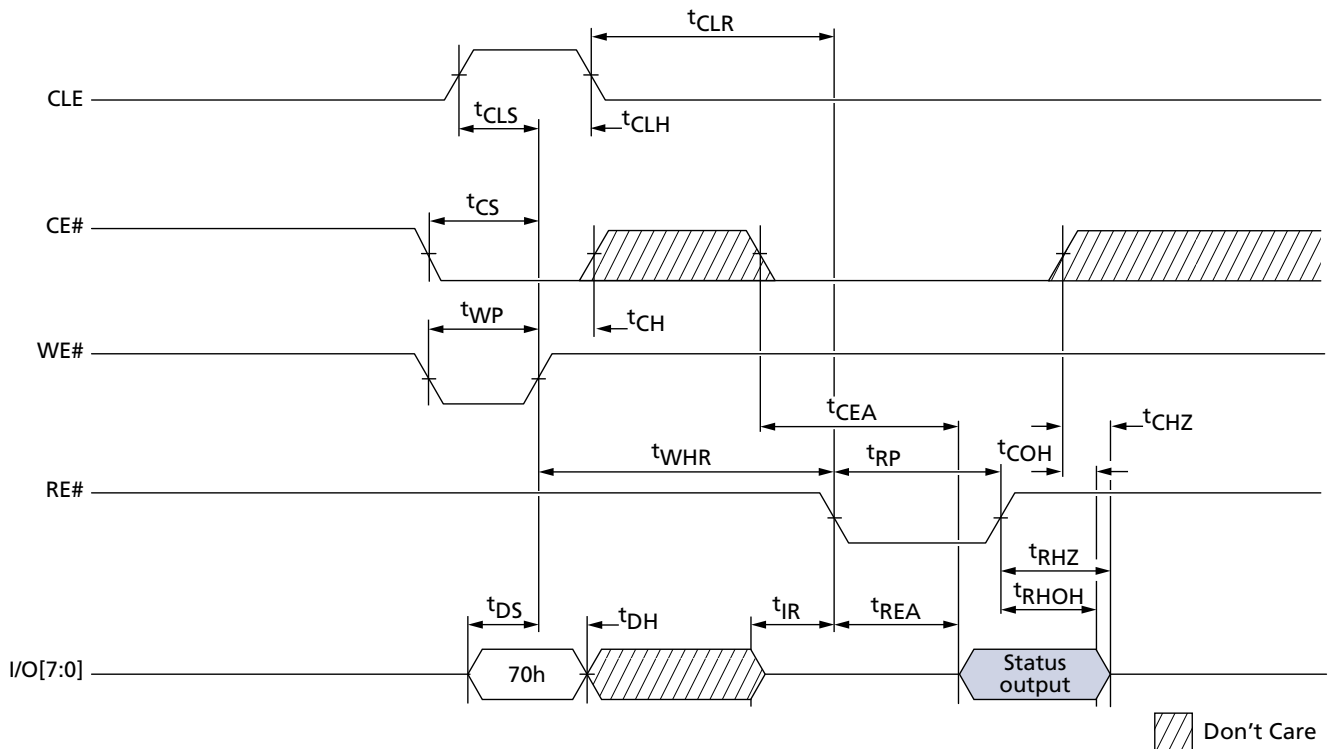


## Asynchronous Interface Timing Diagrams

**Figure 82: RESET Operation**



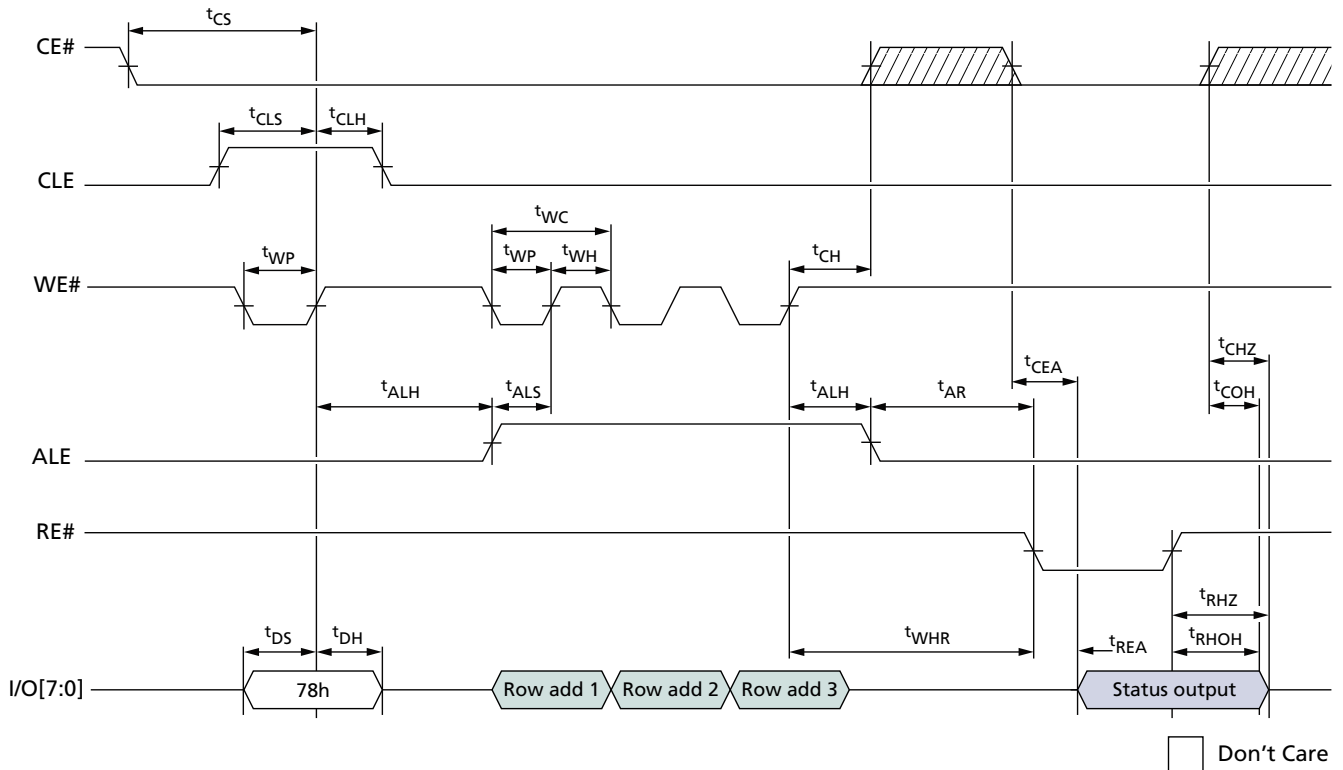
**Figure 83: READ STATUS Cycle**



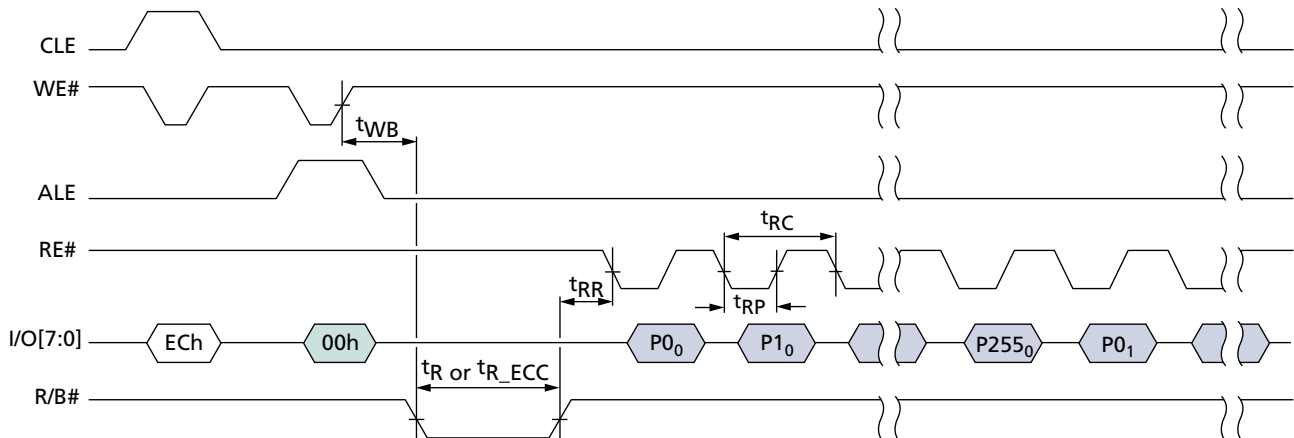


# 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

**Figure 84: READ STATUS ENHANCED Cycle**



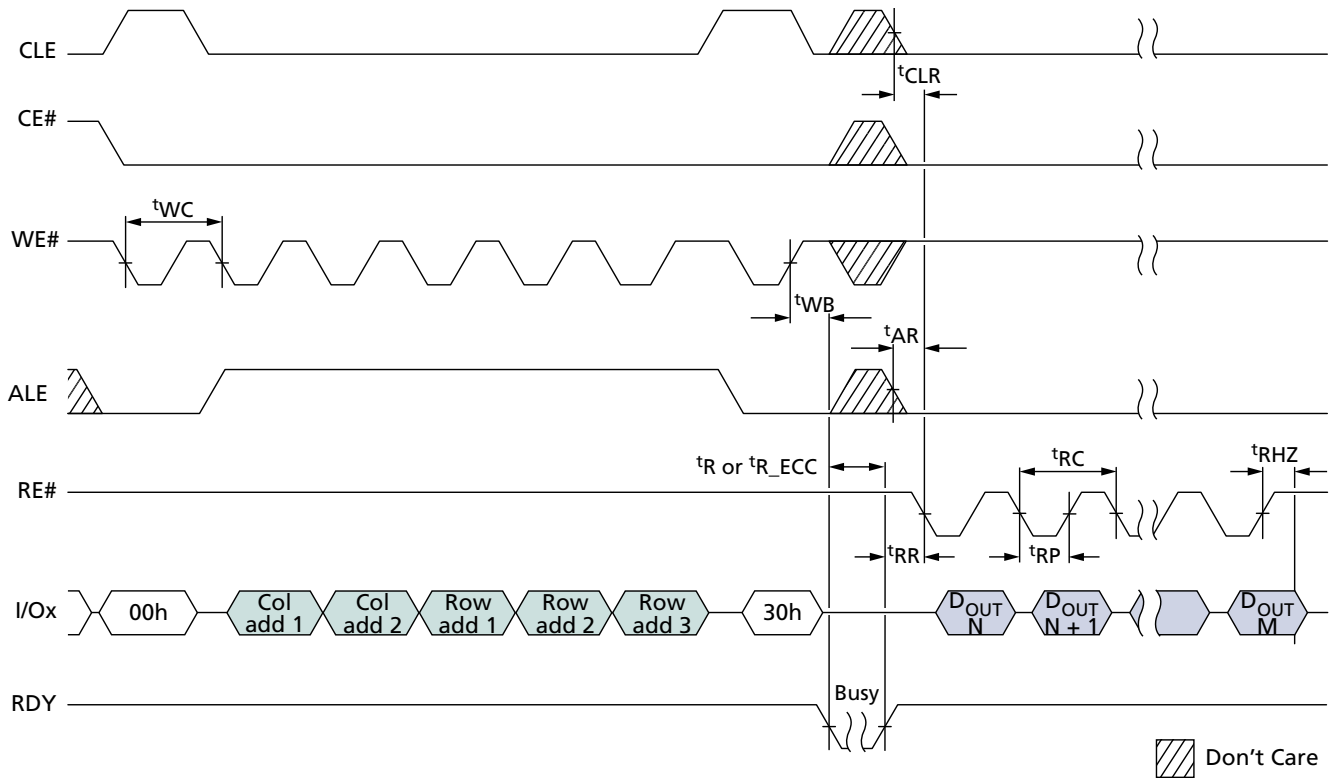
**Figure 85: READ PARAMETER PAGE**





# 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

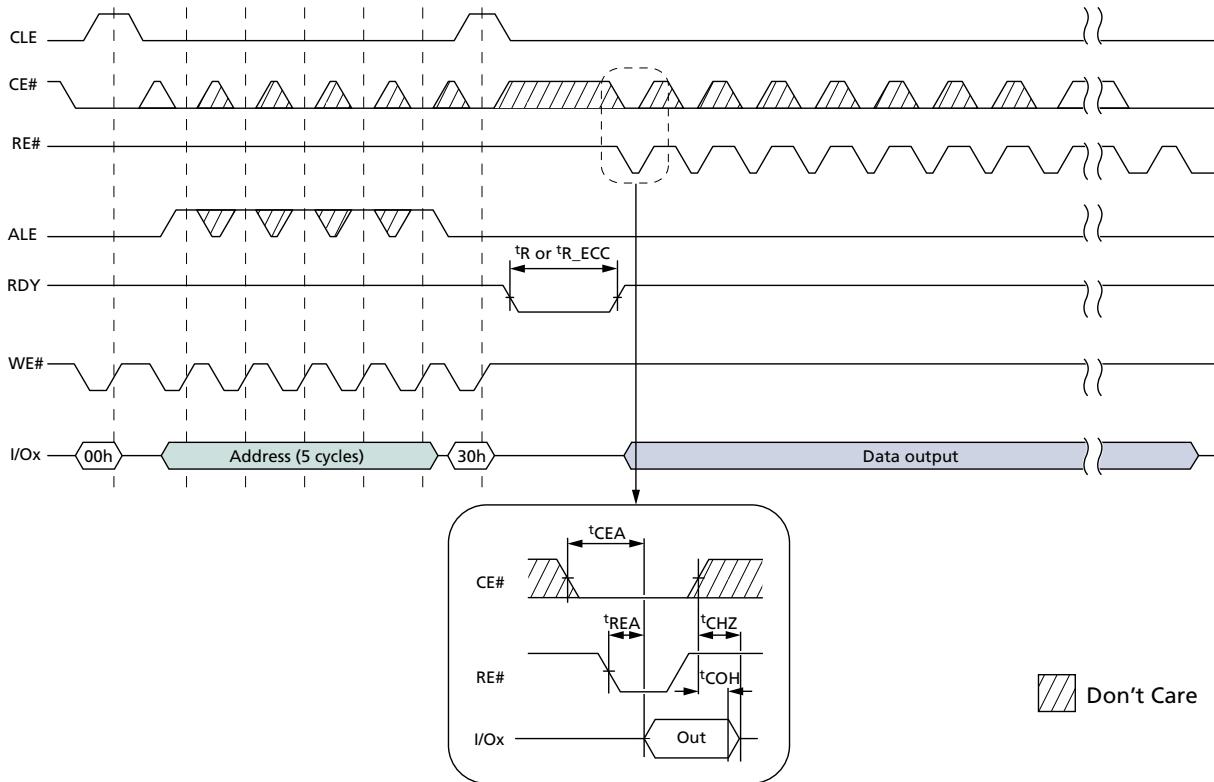
**Figure 86: READ PAGE**





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

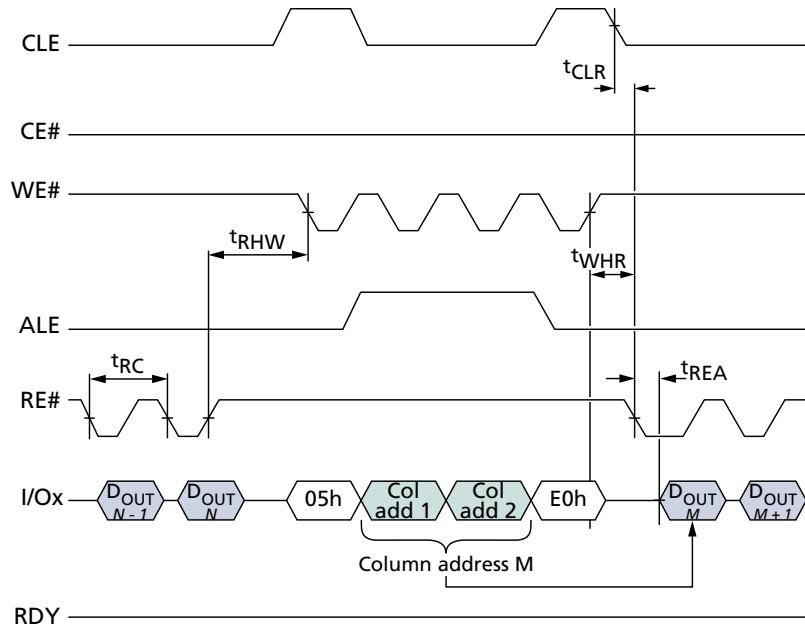
**Figure 87: READ PAGE Operation with CE# "Don't Care"**





## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

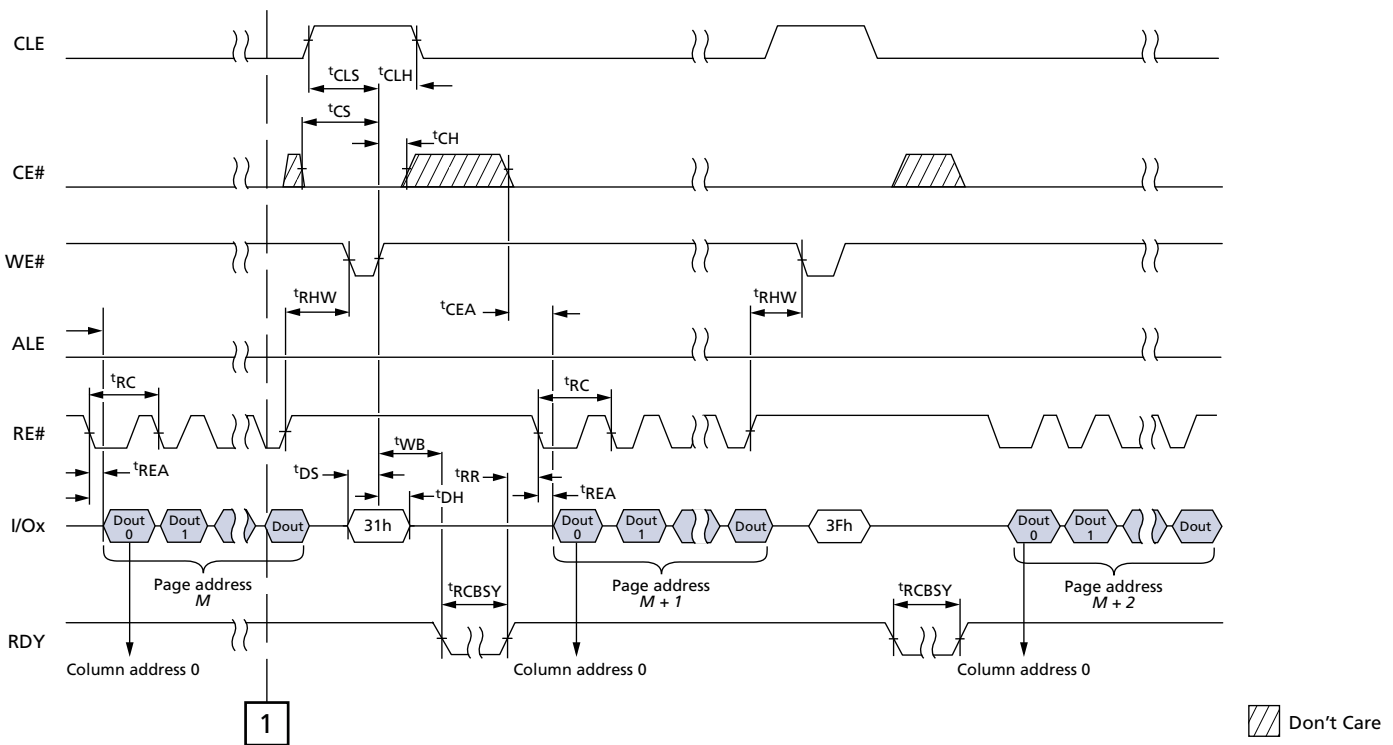
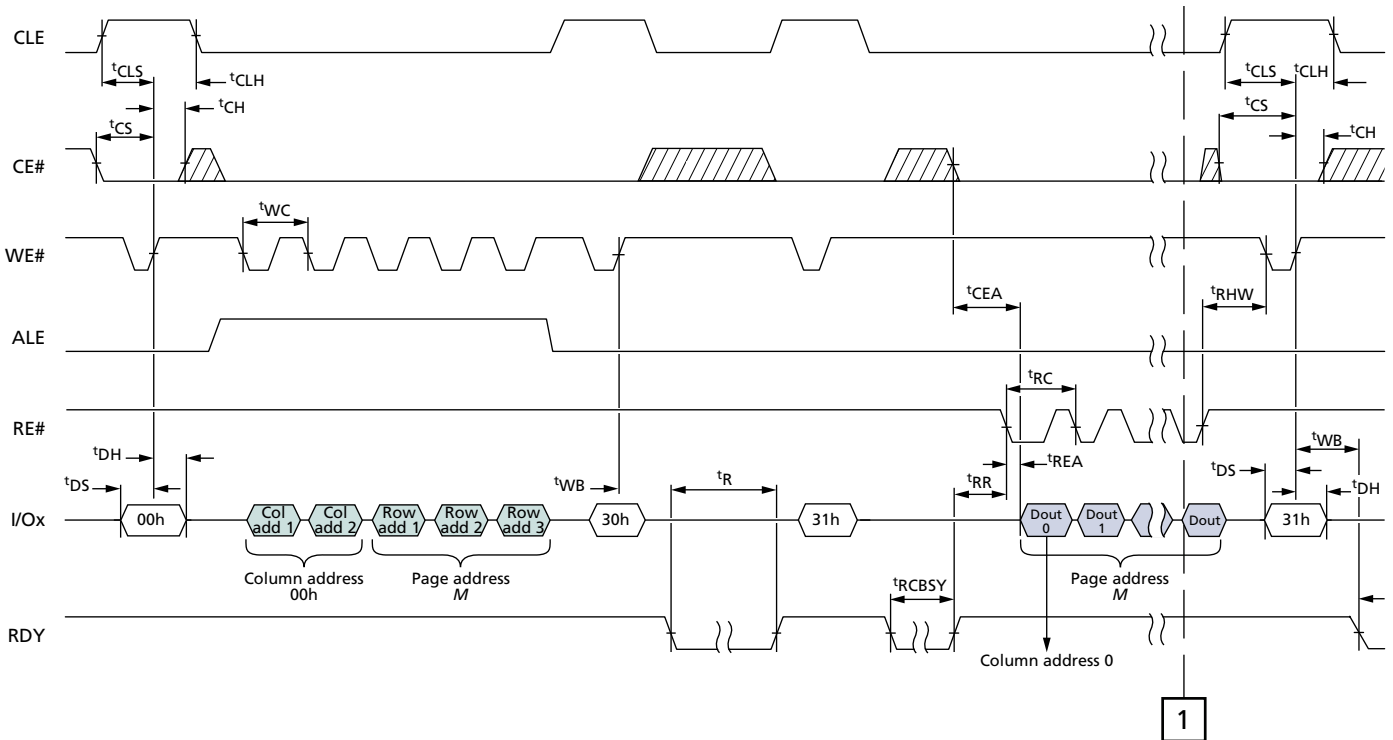
**Figure 88: RANDOM DATA READ**





# 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

**Figure 89: READ PAGE CACHE SEQUENTIAL**

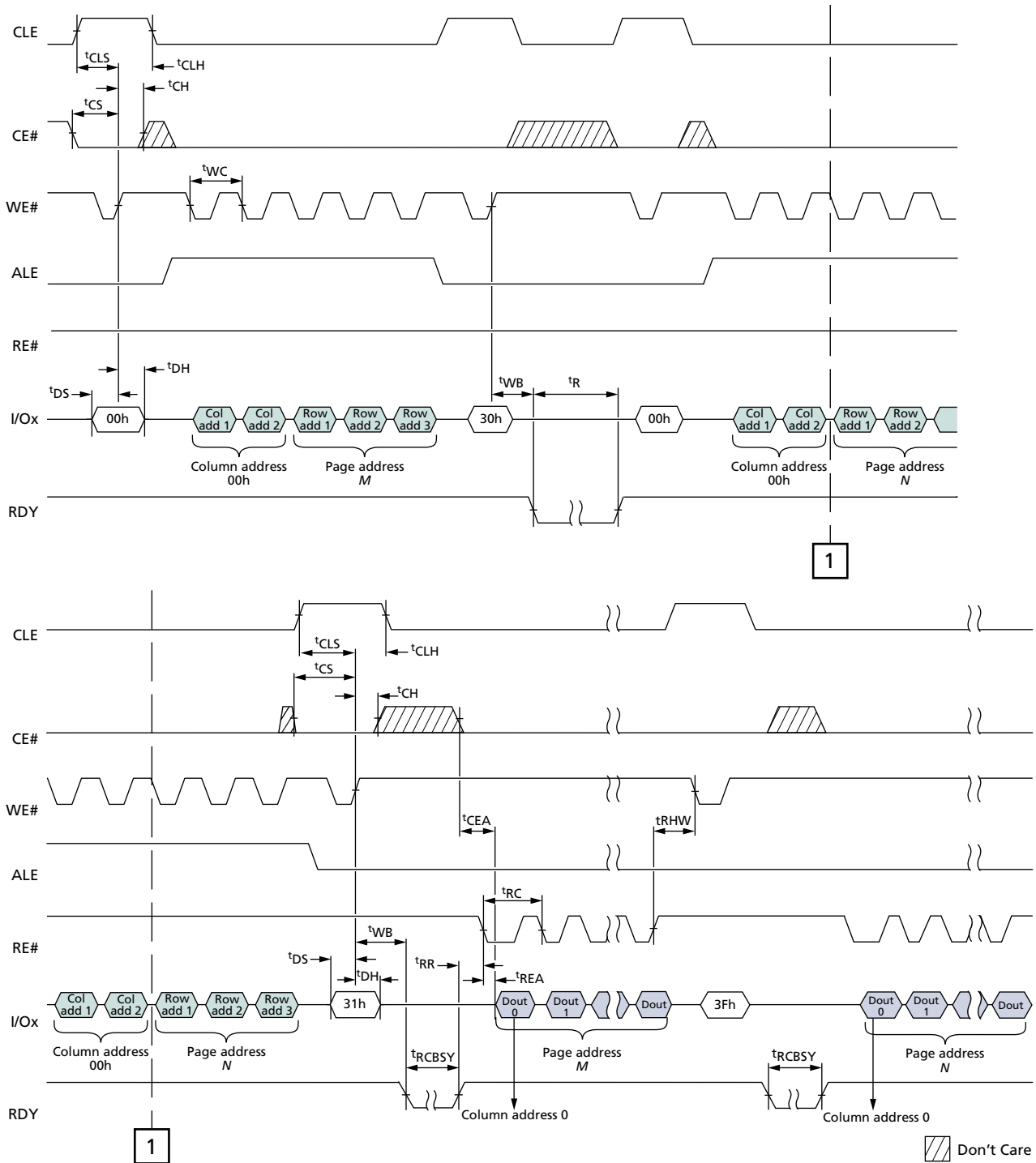


Don't Care



# 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

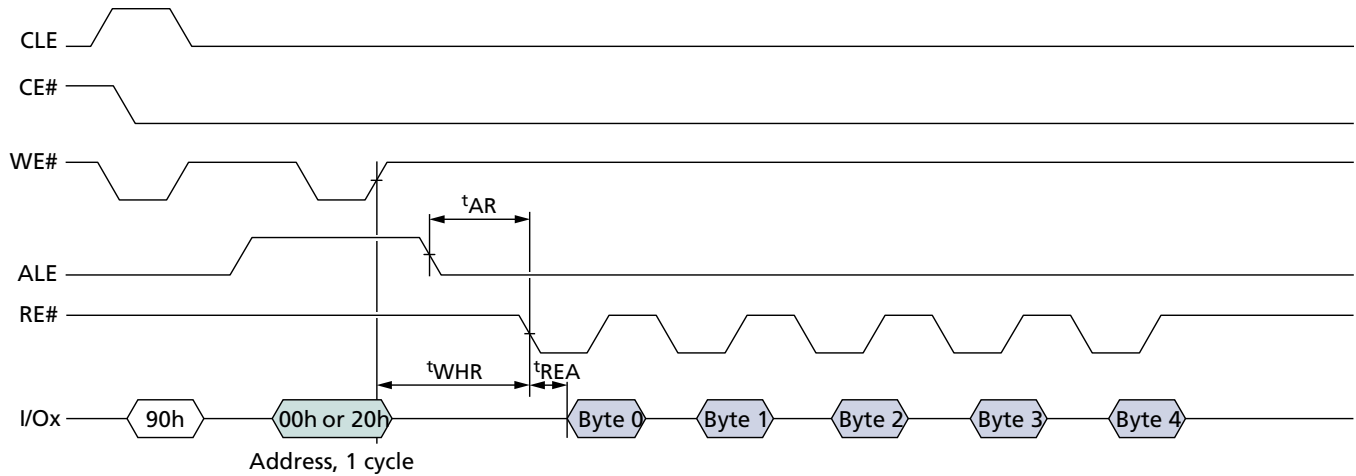
**Figure 90: READ PAGE CACHE RANDOM**



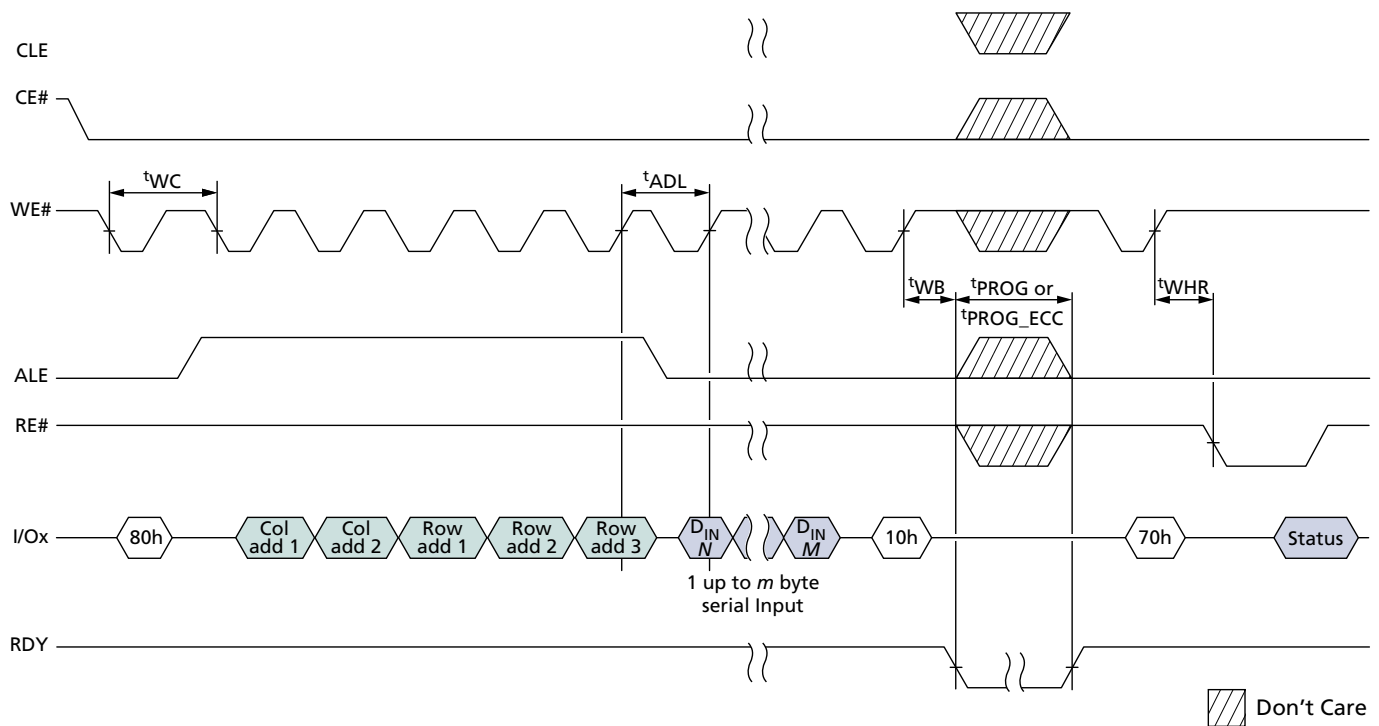


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

### Figure 91: READ ID Operation



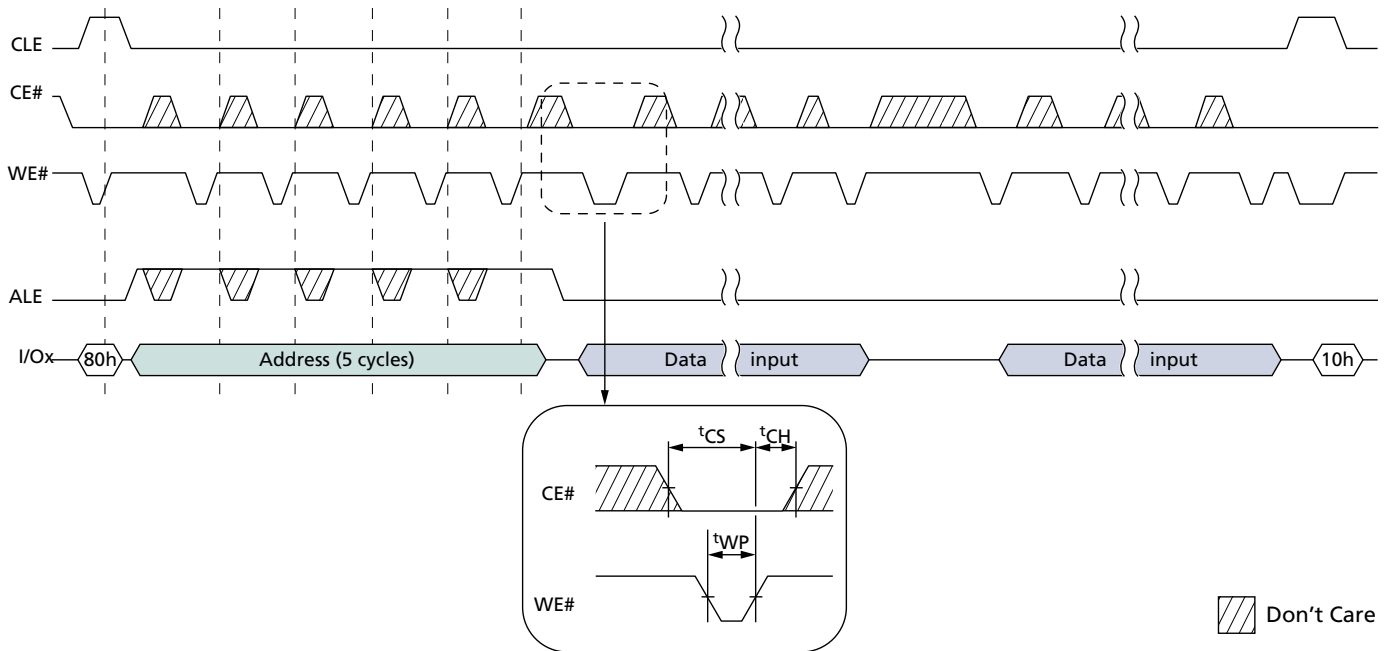
### Figure 92: PROGRAM PAGE Operation



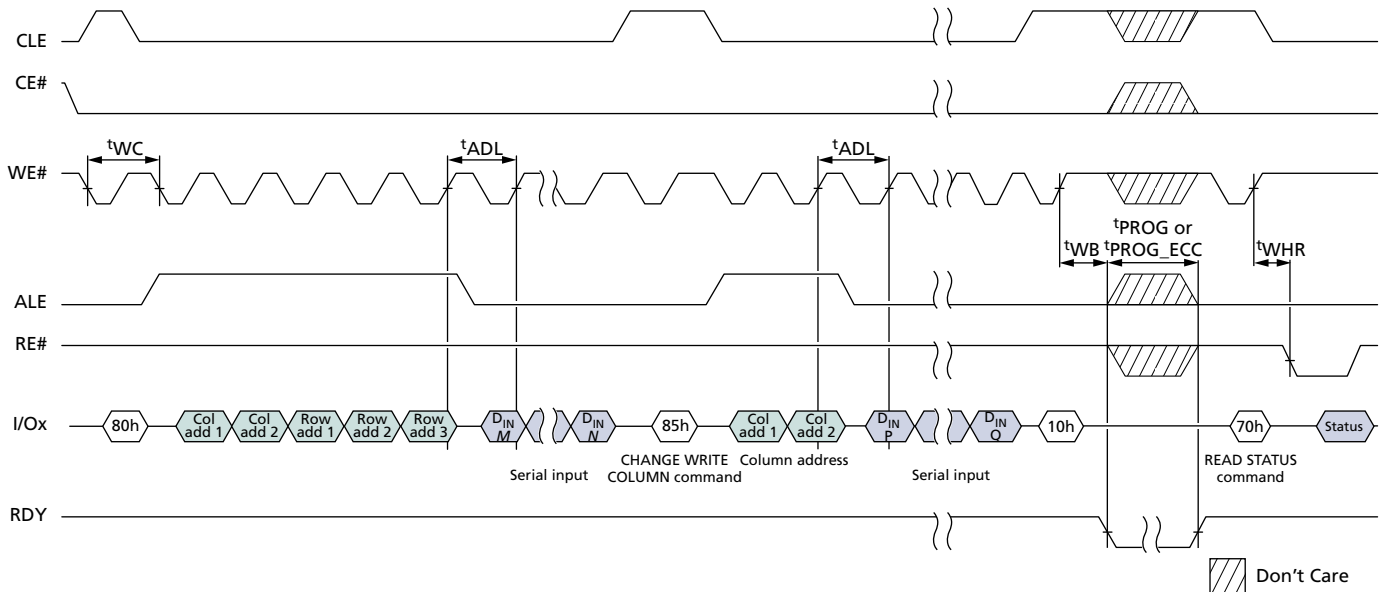


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

**Figure 93: PROGRAM PAGE Operation with CE# "Don't Care"**



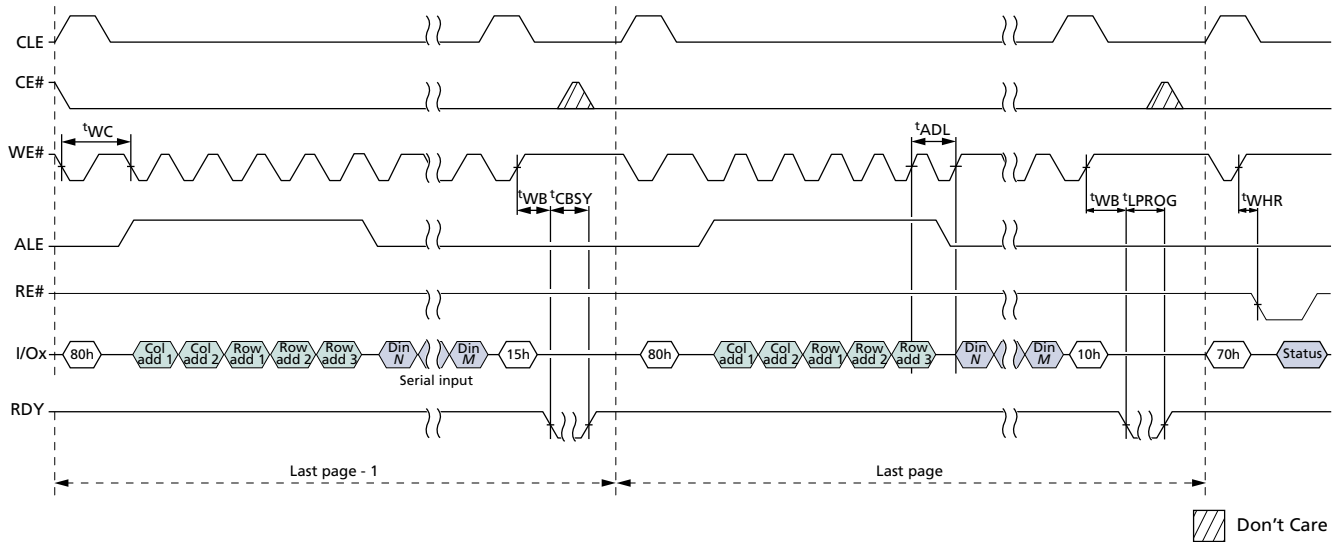
**Figure 94: PROGRAM PAGE Operation with RANDOM DATA INPUT**



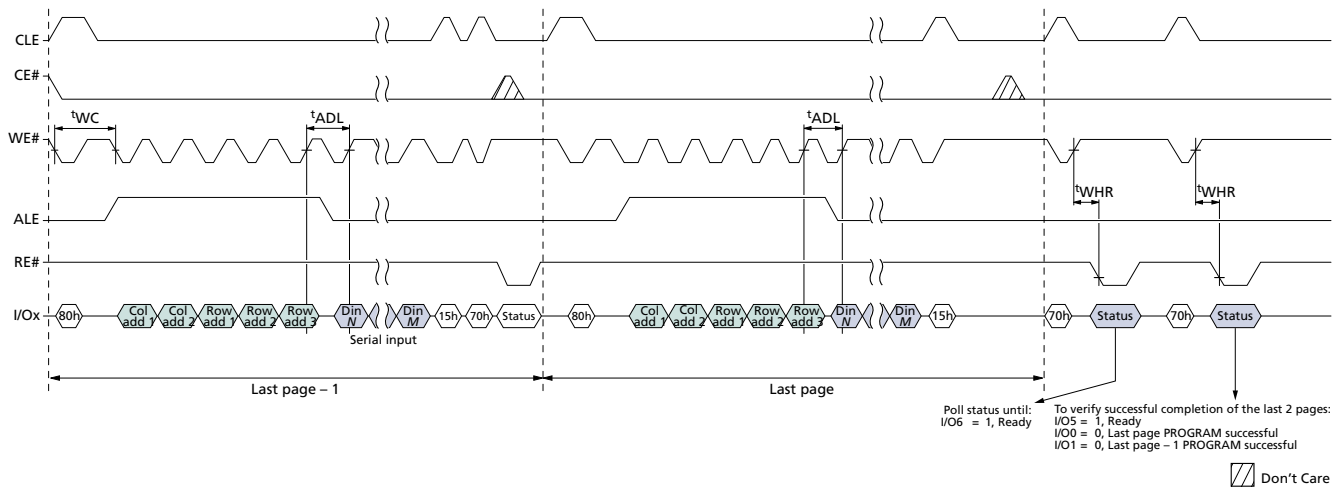


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

**Figure 95: PROGRAM PAGE CACHE**



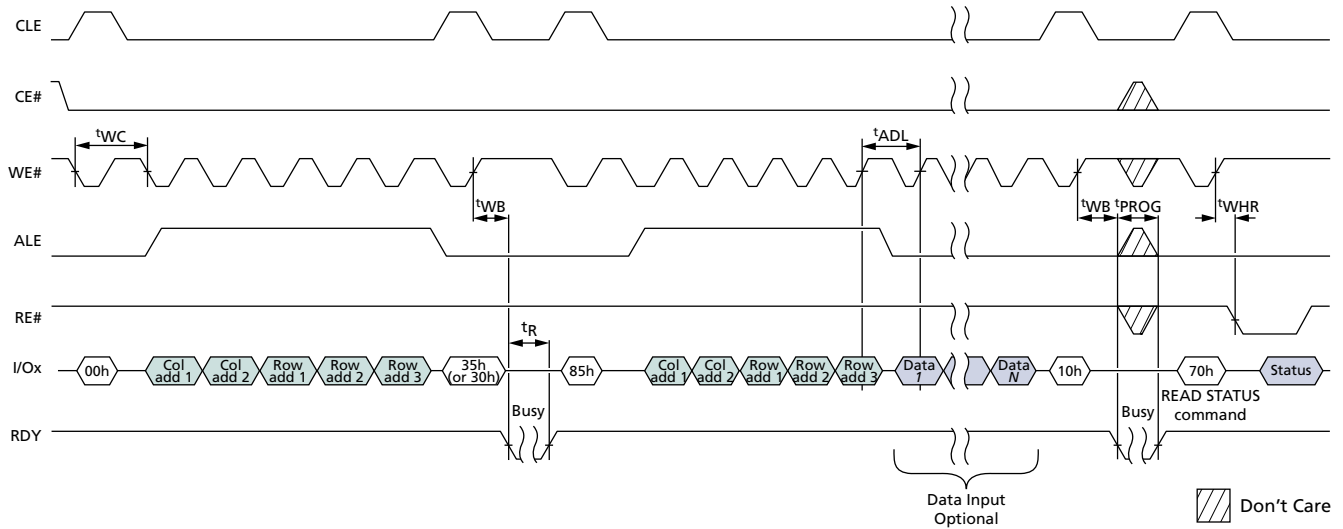
**Figure 96: PROGRAM PAGE CACHE Ending on 15h**



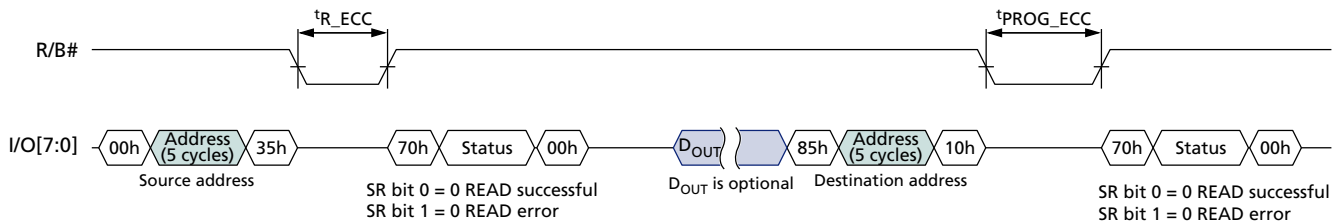


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

**Figure 97: INTERNAL DATA MOVE**



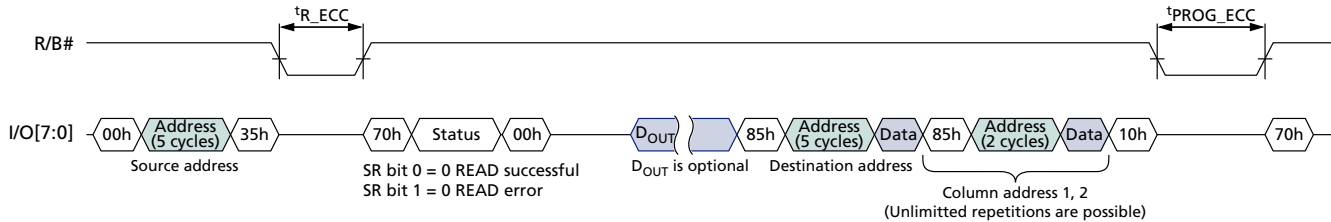
**Figure 98: INTERNAL DATA MOVE (85h-10h) with Internal ECC Enabled**



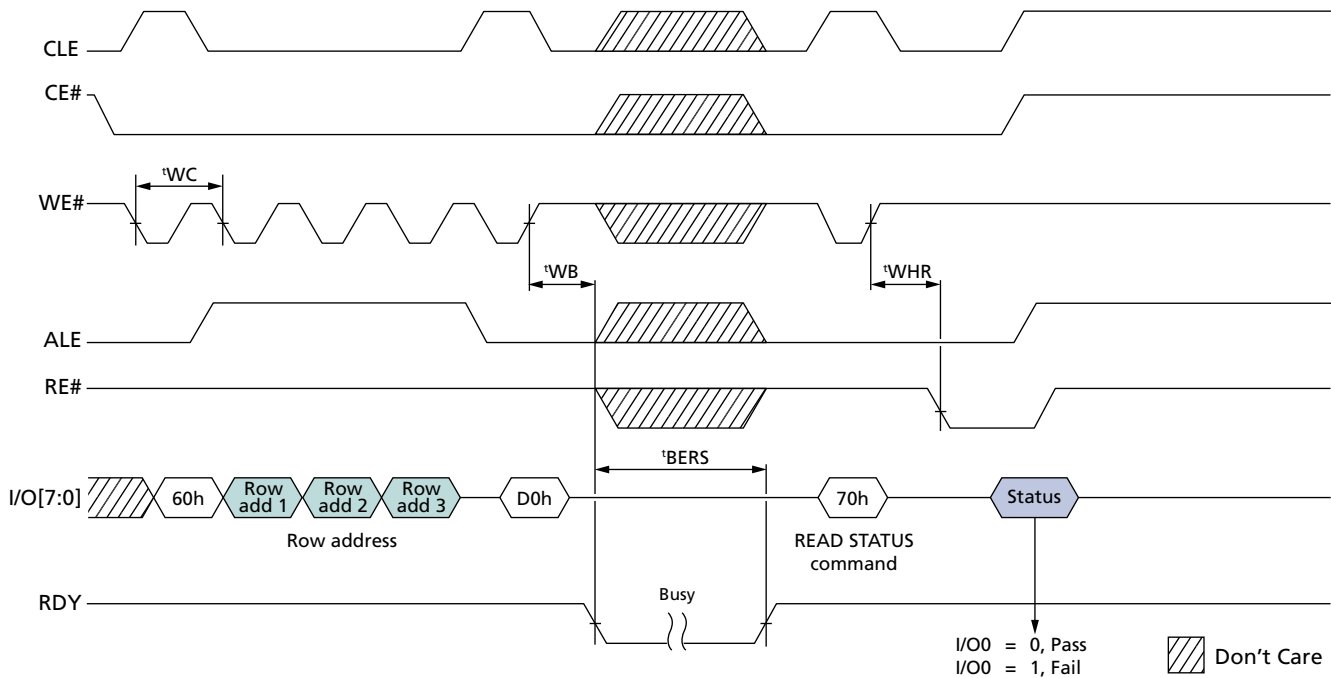


## 4Gb, 8Gb: x8, x16 NAND Flash Memory Asynchronous Interface Timing Diagrams

**Figure 99: INTERNAL DATA MOVE (85h-10h) with Random Data Input with Internal ECC Enabled**



**Figure 100: ERASE BLOCK Operation**





## Revision History

### Rev. G, Production – 10/10

- Removed the words "or by factory (always enabled)" from the General Description

### Rev. F, Production – 6/10

- Replaced blank with 3 for number of valid address cycles on Block Erase Two-Plane in the Two-Plane Command Set Table

### Rev. E, Production – 5/10

- Added part numbers to document
- Removed Endurance spec from Features and Parameter Page Data Structure Table
- Filled in missing values to READ ID Table
- Changed <sup>t</sup>BERS from .5 to .7 in Electrical Specifications – Program/Erase Characteristics
- Replaced Status Register Definition table with the correct one for ECC

### Rev. D, Advance - 3/10

- Updated value for byte 113 to 01h; value for byte 114 to 0Eh in Parameter Page Data Structure Tables
- Updated note 6 in Electrical Specifications - Program/Erase Characteristics to say "disabled"
- Fixed note typo in Features
- Updated OTP Protect - changed to protect by block; removed protect by page
- Updated 1.8V Active Current specs for single die and fixed typos in DC tables

### Rev. C, Advance – 1/10

- Updated READ ID Tables to include the following value changes: Byte 1 – MT29F4G08ABBDA (4Gb, x8, 1.8V) Value: ACh, MT29F4G16ABBDA (4Gb, x16, 1.8V) Value: BCh; Byte 2 – MT29F4G08ABBDA Value: 90h, MT29F4G16ABBDA Value: 90h; Byte 3 – MT29F4G08ABBDA Value: 15h, MT29F4G16ABBDA Value: 55h; Byte 4 – MT29F4G08ABBDA Value: 56h, MT29F4G16ABBDA Value: 56h; Removed H4 from part numbers
- Added Bare Die Parameter Page Data Structure Table
- Removed Boot Block

### Rev. B, Advance – 10/09

- Removed part numbers: MT29F4G08ABBDAWP and MT29F4G16ABBDAWP
- Updated "Internal Data Move with Internal ECC Enabled" graphic spec from <sup>t</sup>R to <sup>t</sup>R\_ECC
- Updated "Internal Data Move with Random Data Input with Internal ECC Enabled" graphic spec from <sup>t</sup>R to <sup>t</sup>R\_ECC
- Updated Boot Block Operation to include dual-plane restrictions
- Added <sup>t</sup>RCBSY spec to Electrical Specifications - Program/Erase Characteristics
- Added note for <sup>t</sup>PROG and <sup>t</sup>PROG\_ECC specifications to Electrical Specifications - Program/Erase Characteristics



## 4Gb, 8Gb: x8, x16 NAND Flash Memory Revision History

- Moved note from <sup>t</sup>RHW to <sup>t</sup>RHZ in AC Characteristics and Operating Conditions

### Rev. A, Advance – 7/09

- Initial release

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900  
www.micron.com/productsupport Customer Comment Line: 800-932-4992  
Micron and the Micron logo are trademarks of Micron Technology, Inc.

All other trademarks are the property of their respective owners.

This data sheet contains minimum and maximum limits specified over the power supply and temperature range set forth herein.  
Although considered final, these specifications are subject to change, as further product development and data characterization sometimes occur.