

# NAND Flash and Mobile LPDDR 168-Ball Package-on-Package (PoP) MCP Combination Memory (TI OMAP™)

**MT29C4G48MAYAPAKQ-5 IT, MT29C4G48MAZAPAKQ-5 IT,**

**MT29C4G48MAZAPAKQ-6 IT, MT29C4G96MAZAPCJG-5 IT,**

**MT29C4G96MAZAPCJG-6 IT, MT29C8G96MAZAPDJV-5 IT,**

**MT29C8G96MAZAPDJV-6 IT**

## Features

- Micron® NAND Flash and LPDDR components
- RoHS-compliant, “green” package
- Separate NAND Flash and LPDDR interfaces
- Space-saving multichip package/package-on-package combination
- Low-voltage operation (1.70–1.95V)
- Industrial temperature range: –40°C to +85°C

## NAND Flash-Specific Features

### Organization

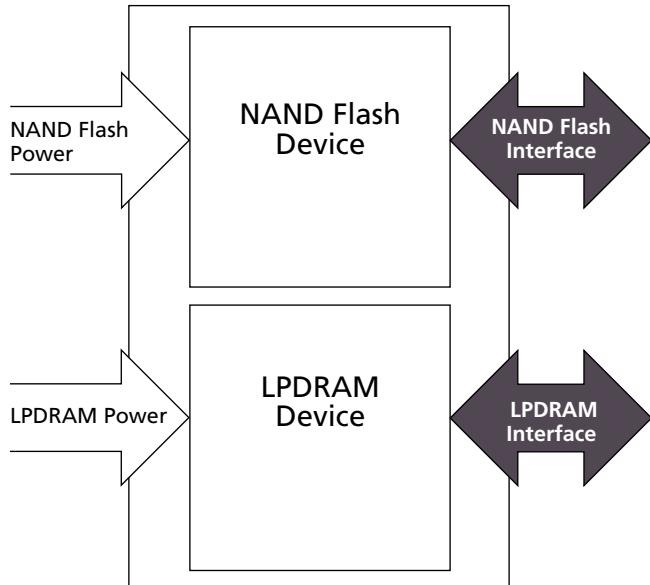
- Page size
  - x8: 2112 bytes (2048 + 64 bytes)
  - x16: 1056 words (1024 + 32 words)
- Block size: 64 pages (128K + 4K bytes)

## Mobile LPDDR-Specific Features

- No external voltage reference required
- No minimum clock rate requirement
- 1.8V LVCMS-compatible inputs
- Programmable burst lengths
- Partial-array self refresh (PASR)
- Deep power-down (DPD) mode
- Selectable output drive strength
- STATUS REGISTER READ (SRR) supported<sup>1</sup>

- Notes:
1. Contact factory for remapped SRR output.
  2. For physical part markings, see Part Numbering Information (page 2).

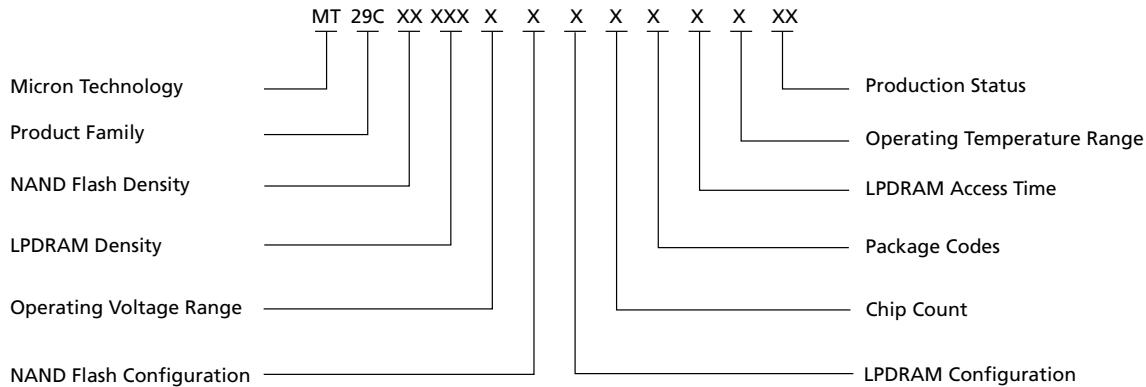
Figure 1: PoP Block Diagram



## Part Numbering Information

Micron NAND Flash and LPDRAM devices are available in different configurations and densities. The MCP/PoP part numbering guide is available at [www.micron.com/numbering](http://www.micron.com/numbering).

**Figure 2: Part Number Chart**



## Device Marking

Due to the size of the package, the Micron-standard part number is not printed on the top of the device. Instead, an abbreviated device mark consisting of a 5-digit alphanumeric code is used. The abbreviated device marks are cross-referenced to the Micron part numbers at the FBGA Part Marking Decoder site: [www.micron.com/decoder](http://www.micron.com/decoder). To view the location of the abbreviated mark on the device, refer to customer service note CSN-11, "Product Mark/Label," at [www.micron.com/csn](http://www.micron.com/csn).



## Contents

MCP General Description .....	11
Ball Assignments and Descriptions .....	12
Electrical Specifications .....	16
Device Diagrams .....	17
Package Dimensions .....	19
4Gb, 8Gb: x8, x16 NAND Flash Memory .....	22
Features .....	22
General Description .....	23
Architecture .....	24
Device and Array Organization .....	25
Asynchronous Interface Bus Operation .....	29
Asynchronous Enable/Standby .....	29
Asynchronous Commands .....	29
Asynchronous Addresses .....	31
Asynchronous Data Input .....	32
Asynchronous Data Output .....	33
Write Protect# .....	34
Ready/Busy# .....	34
Device Initialization .....	39
Command Definitions .....	40
Reset Operations .....	43
RESET (FFh) .....	43
Identification Operations .....	44
READ ID (90h) .....	44
READ ID Parameter Tables .....	45
READ PARAMETER PAGE (ECh) .....	47
Bare Die Parameter Page Data Structure Tables .....	48
READ UNIQUE ID (EDh) .....	51
Feature Operations .....	52
SET FEATURES (EFh) .....	53
GET FEATURES (EEh) .....	54
Status Operations .....	57
READ STATUS (70h) .....	58
READ STATUS ENHANCED (78h) .....	58
Column Address Operations .....	60
RANDOM DATA READ (05h-E0h) .....	60
RANDOM DATA READ TWO-PLANE (06h-E0h) .....	61
RANDOM DATA INPUT (85h) .....	62
PROGRAM FOR INTERNAL DATA INPUT (85h) .....	63
Read Operations .....	65
READ MODE (00h) .....	67
READ PAGE (00h-30h) .....	67
READ PAGE CACHE SEQUENTIAL (31h) .....	68
READ PAGE CACHE RANDOM (00h-31h) .....	69
READ PAGE CACHE LAST (3Fh) .....	71
READ PAGE TWO-PLANE 00h-00h-30h .....	72
Program Operations .....	74
PROGRAM PAGE (80h-10h) .....	75
PROGRAM PAGE CACHE (80h-15h) .....	75
PROGRAM PAGE TWO-PLANE (80h-11h) .....	78



Erase Operations .....	80
ERASE BLOCK (60h-D0h) .....	80
ERASE BLOCK TWO-PLANE (60h-D1h) .....	81
Internal Data Move Operations .....	82
READ FOR INTERNAL DATA MOVE (00h-35h) .....	83
PROGRAM FOR INTERNAL DATA MOVE (85h-10h) .....	84
PROGRAM FOR INTERNAL DATA MOVE TWO-PLANE (85h-11h) .....	85
Block Lock Feature .....	86
WP# and Block Lock .....	86
UNLOCK (23h-24h) .....	86
LOCK (2Ah) .....	89
LOCK TIGHT (2Ch) .....	90
BLOCK LOCK READ STATUS (7Ah) .....	91
One-Time Programmable (OTP) Operations .....	93
Legacy OTP Commands .....	93
OTP DATA PROGRAM (80h-10h) .....	94
RANDOM DATA INPUT (85h) .....	95
OTP DATA PROTECT (80h-10) .....	96
OTP DATA READ (00h-30h) .....	98
Two-Plane Operations .....	100
Two-Plane Addressing .....	100
Interleaved Die (Multi-LUN) Operations .....	109
Error Management .....	110
Internal ECC and Spare Area Mapping for ECC .....	112
Electrical Specifications .....	114
Electrical Specifications – DC Characteristics and Operating Conditions .....	116
Electrical Specifications – AC Characteristics and Operating Conditions .....	118
Electrical Specifications – Program/Erase Characteristics .....	121
Asynchronous Interface Timing Diagrams .....	122
2Gb: x16, x32 Mobile LPDDR SDRAM .....	134
Features .....	134
General Description .....	135
Functional Block Diagrams .....	136
Electrical Specifications .....	138
Electrical Specifications – IDD Parameters .....	141
Electrical Specifications – AC Operating Conditions .....	147
Output Drive Characteristics .....	152
Functional Description .....	155
Commands .....	156
DESELECT .....	157
NO OPERATION .....	157
LOAD MODE REGISTER .....	157
ACTIVE .....	157
READ .....	158
WRITE .....	159
PRECHARGE .....	160
BURST TERMINATE .....	161
AUTO REFRESH .....	161
SELF REFRESH .....	162
DEEP POWER-DOWN .....	162
Truth Tables .....	163
State Diagram .....	168



Initialization .....	169
Standard Mode Register .....	172
Burst Length .....	173
Burst Type .....	173
CAS Latency .....	174
Operating Mode .....	175
Extended Mode Register .....	176
Temperature-Compensated Self Refresh .....	176
Partial-Array Self Refresh .....	177
Output Drive Strength .....	177
Status Read Register .....	178
Bank/Row Activation .....	180
READ Operation .....	181
WRITE Operation .....	192
PRECHARGE Operation .....	204
Auto Precharge .....	204
Concurrent Auto Precharge .....	205
AUTO REFRESH Operation .....	210
SELF REFRESH Operation .....	211
Power-Down .....	212
Deep Power-Down .....	214
Clock Change Frequency .....	216
Revision History .....	217
Rev. H, Production – 3/11 .....	217
Rev. G, Preliminary – 6/10 .....	217
Rev. F, Preliminary – 3/10 .....	217
Rev. E, Preliminary – 2/10 .....	217
Rev. D, Preliminary – 1/10 .....	217
Rev. C, Preliminary – 12/09 .....	217
Rev. B, Preliminary – 10/09 .....	218
Rev. A, Preliminary – 7/09 .....	218



## List of Tables

Table 1: x8, x16 NAND Ball Descriptions .....	13
Table 2: x32 LPDDR Ball Descriptions .....	14
Table 3: Non-Device-Specific Descriptions .....	15
Table 4: Absolute Maximum Ratings .....	16
Table 5: Recommended Operating Conditions .....	16
Table 6: Array Addressing – MT29F4G08 (x8) .....	25
Table 7: Array Addressing – MT29F4G16 (x16) .....	26
Table 8: Array Addressing – MT29F8G08 (x8) .....	27
Table 9: Array Addressing – MT29F8G16 (x16) .....	28
Table 10: Asynchronous Interface Mode Selection .....	29
Table 11: Command Set .....	40
Table 12: Two-Plane Command Set .....	41
Table 13: READ ID Parameters for Address 00h .....	45
Table 14: READ ID Parameters for Address 20h .....	46
Table 15: Parameter Page Data Structure .....	48
Table 16: Feature Address Definitions .....	52
Table 17: Feature Address 90h – Array Operation Mode .....	53
Table 18: Feature Addresses 01h: Timing Mode .....	55
Table 19: Feature Addresses 80h: Programmable I/O Drive Strength .....	56
Table 20: Feature Addresses 81h: Programmable R/B# Pull-Down Strength .....	56
Table 21: Status Register Definition .....	57
Table 22: Block Lock Address Cycle Assignments .....	88
Table 23: Block Lock Status Register Bit Definitions .....	91
Table 24: Error Management Details .....	110
Table 25: Absolute Maximum Ratings .....	114
Table 26: Recommended Operating Conditions .....	114
Table 27: Valid Blocks .....	114
Table 28: Capacitance .....	115
Table 29: Test Conditions .....	115
Table 30: DC Characteristics and Operating Conditions (3.3V) .....	116
Table 31: DC Characteristics and Operating Conditions (1.8V) .....	117
Table 32: AC Characteristics: Command, Data, and Address Input (3.3V) .....	118
Table 33: AC Characteristics: Command, Data, and Address Input (1.8V) .....	118
Table 34: AC Characteristics: Normal Operation (3.3V) .....	119
Table 35: AC Characteristics: Normal Operation (1.8V) .....	119
Table 36: Program/Erase Characteristics .....	121
Table 37: Configuration Addressing – 2Gb .....	134
Table 38: Absolute Maximum Ratings .....	138
Table 39: AC/DC Electrical Characteristics and Operating Conditions .....	138
Table 40: Capacitance (x16, x32) .....	140
Table 41: $I_{DD}$ Specifications and Conditions, -40°C to +85°C (x16) .....	141
Table 42: $I_{DD}$ Specifications and Conditions, -40°C to +85°C (x32) .....	142
Table 43: $I_{DD}$ Specifications and Conditions, -40°C to +105°C (x16) .....	143
Table 44: $I_{DD}$ Specifications and Conditions, -40°C to +105°C (x32) .....	144
Table 45: $I_{DD6}$ Specifications and Conditions .....	145
Table 46: Electrical Characteristics and Recommended AC Operating Conditions .....	147
Table 47: Target Output Drive Characteristics (Full Strength) .....	152
Table 48: Target Output Drive Characteristics (Three-Quarter Strength) .....	153
Table 49: Target Output Drive Characteristics (One-Half Strength) .....	154
Table 50: Truth Table – Commands .....	156



Table 51: DM Operation Truth Table .....	157
Table 52: Truth Table – Current State Bank $n$ – Command to Bank $n$ .....	163
Table 53: Truth Table – Current State Bank $n$ – Command to Bank $m$ .....	165
Table 54: Truth Table – CKE .....	167
Table 55: Burst Definition Table .....	173



## List of Figures

Figure 1: PoP Block Diagram .....	1
Figure 2: Part Number Chart .....	2
Figure 3: 168-Ball VFBGA (NAND x8, x16; LPDDR x32) Ball Assignments .....	12
Figure 4: 168-Ball (Single LPDDR) Functional Block Diagram .....	17
Figure 5: 168-Ball (Dual LPDDR) Functional Block Diagram .....	18
Figure 6: 168-Ball VFBGA (Package Code: JG) .....	19
Figure 7: 168-Ball VFBGA (Package Code: JV) .....	20
Figure 8: 168-Ball WFBGA (Package Code: KQ) .....	21
Figure 9: NAND Flash Die (LUN) Functional Block Diagram .....	24
Figure 10: Array Organization – MT29F4G08 (x8) .....	25
Figure 11: Array Organization – MT29F4G16 (x16) .....	26
Figure 12: Array Organization – MT29F8G08 (x8) .....	27
Figure 13: Array Organization – MT29F8G16 (x16) .....	28
Figure 14: Asynchronous Command Latch Cycle .....	30
Figure 15: Asynchronous Address Latch Cycle .....	31
Figure 16: Asynchronous Data Input Cycles .....	32
Figure 17: Asynchronous Data Output Cycles .....	33
Figure 18: Asynchronous Data Output Cycles (EDO Mode) .....	34
Figure 19: READ/BUSY# Open Drain .....	35
Figure 20: $t_{Fall}$ and $t_{Rise}$ (3.3V $V_{CC}$ ) .....	36
Figure 21: $t_{Fall}$ and $t_{Rise}$ (1.8V $V_{CC}$ ) .....	36
Figure 22: $I_{OL}$ vs. $R_p$ ( $V_{CC} = 3.3V V_{CC}$ ) .....	37
Figure 23: $I_{OL}$ vs. $R_p$ (1.8V $V_{CC}$ ) .....	37
Figure 24: $T_C$ vs. $R_p$ .....	38
Figure 25: R/B# Power-On Behavior .....	39
Figure 26: RESET (FFh) Operation .....	43
Figure 27: READ ID (90h) with 00h Address Operation .....	44
Figure 28: READ ID (90h) with 20h Address Operation .....	44
Figure 29: READ PARAMETER (ECh) Operation .....	47
Figure 30: READ UNIQUE ID (EDh) Operation .....	51
Figure 31: SET FEATURES (EFh) Operation .....	53
Figure 32: GET FEATURES (EEh) Operation .....	54
Figure 33: READ STATUS (70h) Operation .....	58
Figure 34: READ STATUS ENHANCED (78h) Operation .....	59
Figure 35: RANDOM DATA READ (05h-E0h) Operation .....	60
Figure 36: RANDOM DATA READ TWO-PLANE (06h-E0h) Operation .....	61
Figure 37: RANDOM DATA INPUT (85h) Operation .....	62
Figure 38: PROGRAM FOR INTERNAL DATA INPUT (85h) Operation .....	64
Figure 39: READ PAGE (00h-30h) Operation .....	68
Figure 40: READ PAGE (00h-30h) Operation with Internal ECC Enabled .....	68
Figure 41: READ PAGE CACHE SEQUENTIAL (31h) Operation .....	69
Figure 42: READ PAGE CACHE RANDOM (00h-31h) Operation .....	70
Figure 43: READ PAGE CACHE LAST (3Fh) Operation .....	71
Figure 44: READ PAGE TWO-PLANE (00h-00h-30h) Operation .....	73
Figure 45: PROGRAM PAGE (80h-10h) Operation .....	75
Figure 46: PROGRAM PAGE CACHE (80h-15h) Operation (Start) .....	77
Figure 47: PROGRAM PAGE CACHE (80h-15h) Operation (End) .....	77
Figure 48: PROGRAM PAGE TWO-PLANE (80h-11h) Operation .....	79
Figure 49: ERASE BLOCK (60h-D0h) Operation .....	80
Figure 50: ERASE BLOCK TWO-PLANE (60h-D1h) Operation .....	81



Figure 51: READ FOR INTERNAL DATA MOVE (00h-35h) Operation .....	83
Figure 52: READ FOR INTERNAL DATA MOVE (00h-35h) with RANDOM DATA READ (05h-E0h) .....	83
Figure 53: INTERNAL DATA MOVE (85h-10h) with Internal ECC Enabled .....	84
Figure 54: INTERNAL DATA MOVE (85h-10h) with RANDOM DATA INPUT with Internal ECC Enabled .....	84
Figure 55: PROGRAM FOR INTERNAL DATA MOVE (85h-10h) Operation .....	84
Figure 56: PROGRAM FOR INTERNAL DATA MOVE (85h-10h) with RANDOM DATA INPUT (85h) .....	85
Figure 57: PROGRAM FOR INTERNAL DATA MOVE TWO-PLANE (85h-11h) Operation .....	85
Figure 58: Flash Array Protected: Invert Area Bit = 0 .....	87
Figure 59: Flash Array Protected: Invert Area Bit = 1 .....	87
Figure 60: UNLOCK Operation .....	88
Figure 61: LOCK Operation .....	89
Figure 62: LOCK TIGHT Operation .....	90
Figure 63: PROGRAM/ERASE Issued to Locked Block .....	91
Figure 64: BLOCK LOCK READ STATUS .....	91
Figure 65: BLOCK LOCK Flowchart .....	92
Figure 66: OTP DATA PROGRAM (After Entering OTP Operation Mode) .....	95
Figure 67: OTP DATA PROGRAM Operation with RANDOM DATA INPUT (After Entering OTP Operation Mode) .....	96
Figure 68: OTP DATA PROTECT Operation (After Entering OTP Protect Mode) .....	97
Figure 69: OTP DATA READ .....	98
Figure 70: OTP DATA READ with RANDOM DATA READ Operation .....	99
Figure 71: TWO-PLANE PAGE READ .....	101
Figure 72: TWO-PLANE PAGE READ with RANDOM DATA READ .....	102
Figure 73: TWO-PLANE PROGRAM PAGE .....	102
Figure 74: TWO-PLANE PROGRAM PAGE with RANDOM DATA INPUT .....	103
Figure 75: TWO-PLANE PROGRAM PAGE CACHE MODE .....	104
Figure 76: TWO-PLANE INTERNAL DATA MOVE .....	105
Figure 77: TWO-PLANE INTERNAL DATA MOVE with TWO-PLANE RANDOM DATA READ .....	106
Figure 78: TWO-PLANE INTERNAL DATA MOVE with RANDOM DATA INPUT .....	107
Figure 79: TWO-PLANE BLOCK ERASE .....	108
Figure 80: TWO-PLANE/MULTIPLE-DIE READ STATUS Cycle .....	108
Figure 81: Spare Area Mapping (x8) .....	112
Figure 82: Spare Area Mapping (x16) .....	113
Figure 83: RESET Operation .....	122
Figure 84: READ STATUS Cycle .....	122
Figure 85: READ STATUS ENHANCED Cycle .....	123
Figure 86: READ PARAMETER PAGE .....	123
Figure 87: READ PAGE .....	124
Figure 88: READ PAGE Operation with CE# “Don’t Care” .....	125
Figure 89: RANDOM DATA READ .....	126
Figure 90: READ PAGE CACHE SEQUENTIAL .....	127
Figure 91: READ PAGE CACHE RANDOM .....	128
Figure 92: READ ID Operation .....	129
Figure 93: PROGRAM PAGE Operation .....	129
Figure 94: PROGRAM PAGE Operation with CE# “Don’t Care” .....	130
Figure 95: PROGRAM PAGE Operation with RANDOM DATA INPUT .....	130
Figure 96: PROGRAM PAGE CACHE .....	131
Figure 97: PROGRAM PAGE CACHE Ending on 15h .....	131
Figure 98: INTERNAL DATA MOVE .....	132
Figure 99: INTERNAL DATA MOVE (85h-10h) with Internal ECC Enabled .....	132
Figure 100: INTERNAL DATA MOVE (85h-10h) with Random Data Input with Internal ECC Enabled .....	133
Figure 101: ERASE BLOCK Operation .....	133



Figure 102: Functional Block Diagram (x16) .....	136
Figure 103: Functional Block Diagram (x32) .....	137
Figure 104: Typical Self Refresh Current vs. Temperature .....	146
Figure 105: ACTIVE Command .....	158
Figure 106: READ Command .....	159
Figure 107: WRITE Command .....	160
Figure 108: PRECHARGE Command .....	161
Figure 109: DEEP POWER-DOWN Command .....	162
Figure 110: Simplified State Diagram .....	168
Figure 111: Initialize and Load Mode Registers .....	170
Figure 112: Alternate Initialization with CKE LOW .....	171
Figure 113: Standard Mode Register Definition .....	172
Figure 114: CAS Latency .....	175
Figure 115: Extended Mode Register .....	176
Figure 116: Status Read Register Timing .....	178
Figure 117: Status Register Definition .....	179
Figure 118: READ Burst .....	182
Figure 119: Consecutive READ Bursts .....	183
Figure 120: Nonconsecutive READ Bursts .....	184
Figure 121: Random Read Accesses .....	185
Figure 122: Terminating a READ Burst .....	186
Figure 123: READ-to-WRITE .....	187
Figure 124: READ-to-PRECHARGE .....	188
Figure 125: Data Output Timing – $t_{DQSQ}$ , $t_{QH}$ , and Data Valid Window (x16) .....	189
Figure 126: Data Output Timing – $t_{DQSQ}$ , $t_{QH}$ , and Data Valid Window (x32) .....	190
Figure 127: Data Output Timing – $t_{AC}$ and $t_{DQSCK}$ .....	191
Figure 128: Data Input Timing .....	193
Figure 129: Write – DM Operation .....	194
Figure 130: WRITE Burst .....	195
Figure 131: Consecutive WRITE-to-WRITE .....	196
Figure 132: Nonconsecutive WRITE-to-WRITE .....	196
Figure 133: Random WRITE Cycles .....	197
Figure 134: WRITE-to-READ – Uninterrupting .....	198
Figure 135: WRITE-to-READ – Interrupting .....	199
Figure 136: WRITE-to-READ – Odd Number of Data, Interrupting .....	200
Figure 137: WRITE-to-PRECHARGE – Uninterrupting .....	201
Figure 138: WRITE-to-PRECHARGE – Interrupting .....	202
Figure 139: WRITE-to-PRECHARGE – Odd Number of Data, Interrupting .....	203
Figure 140: Bank Read – With Auto Precharge .....	206
Figure 141: Bank Read – Without Auto Precharge .....	207
Figure 142: Bank Write – With Auto Precharge .....	208
Figure 143: Bank Write – Without Auto Precharge .....	209
Figure 144: Auto Refresh Mode .....	210
Figure 145: Self Refresh Mode .....	212
Figure 146: Power-Down Entry (in Active or Precharge Mode) .....	213
Figure 147: Power-Down Mode (Active or Precharge) .....	214
Figure 148: Deep Power-Down Mode .....	215
Figure 149: Clock Stop Mode .....	216

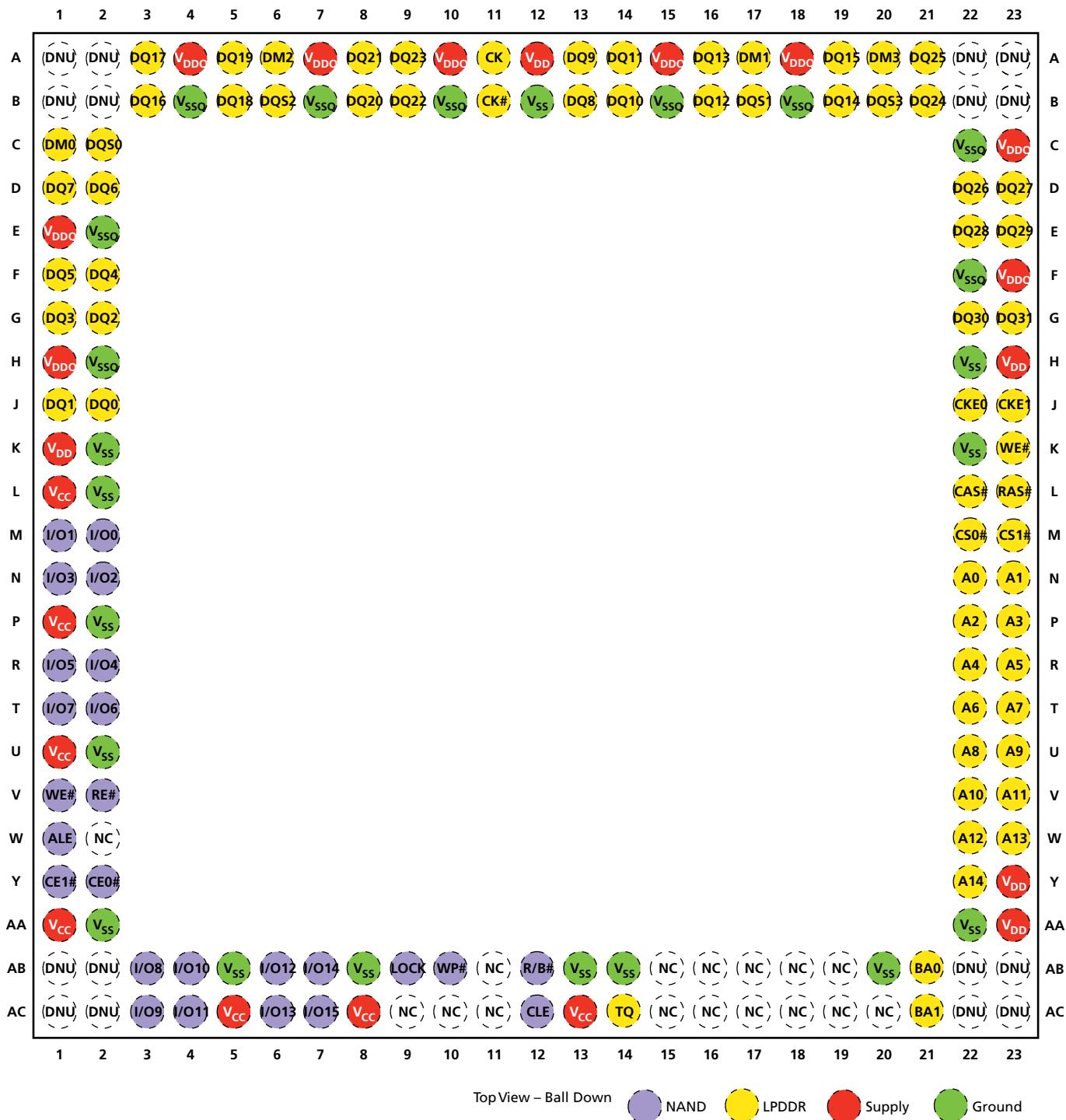


## MCP General Description

Micron package-on-package (PoP) MCP products combine NAND Flash and Mobile LPDRAM devices in a single MCP. These products target mobile applications with low-power, high-performance, and minimal package-footprint design requirements. The NAND Flash and Mobile LPDRAM devices are also members of the Micron discrete memory products portfolio.

The NAND Flash and Mobile LPDRAM devices are packaged with separate interfaces (no shared address, control, data, or power balls). This bus architecture supports an optimized interface to processors with separate NAND Flash and Mobile LPDRAM buses. The NAND Flash and Mobile LPDRAM devices have separate core power connections and share a common ground (that is, V<sub>SS</sub> is tied together on the two devices).

The bus architecture of this device also supports separate NAND Flash and Mobile LPDRAM functionality without concern for device interaction.

**Ball Assignments and Descriptions****Figure 3: 168-Ball VFBGA (NAND x8, x16; LPDDR x32) Ball Assignments**

**Table 1: x8, x16 NAND Ball Descriptions**

<b>Symbol</b>	<b>Type</b>	<b>Description</b>
ALE	Input	Address latch enable: When ALE is HIGH, addresses can be transferred to the on-chip address register.
CE0#, CE1#	Input	Chip enable: Gates transfers between the host system and the NAND device. CE1# is used when a second CE# is required and is RFU <sup>1</sup> in all other configurations.
CLE	Input	Command latch enable: When CLE is HIGH, commands can be transferred to the on-chip command register.
LOCK	Input	When LOCK is HIGH during power-up, the BLOCK LOCK function is enabled. To disable BLOCK LOCK, connect LOCK to V <sub>SS</sub> during power-up, or leave it unconnected (internal pull-down).
RE#	Input	Read enable: Gates information from the NAND device to the host system.
WE#	Input	Write enable: Gates information from the host system to the NAND device.
WP#	Input	Write protect: Driving WP# LOW blocks ERASE and PROGRAM operations.
I/O[15:0]	Input/output	Data inputs/outputs: The bidirectional I/Os transfer address, data, and instruction information. Data is output only during READ operations; at other times the I/Os are inputs. I/O[15:8] are RFU for x8 NAND devices.
R/B#	Output	Ready/busy: Open-drain, active-LOW output that indicates when an internal operation is in progress.
V <sub>CC</sub>	Supply	V <sub>CC</sub> : NAND power supply.

Note: 1. Balls marked RFU may or may not be connected internally. These balls should not be used. Contact factory for details.

**Table 2: x32 LPDDR Ball Descriptions**

<b>Symbol</b>	<b>Type</b>	<b>Description</b>
A[14:0]	Input	Address inputs: Specifies the row or column address. Also used to load the mode registers. The maximum LPDDR address is determined by density and configuration. Consult the LPDDR product data sheet for the maximum address for a given density and configuration. Unused address balls become RFU. <sup>1</sup>
BA0, BA1	Input	Bank address inputs: Specifies one of the 4 banks.
CAS#	Input	Column select: Specifies which command to execute.
CK, CK#	Input	CK is the system clock. CK and CK# are differential clock inputs. All address and control signals are sampled and referenced on the crossing of the rising edge of CK with the falling edge of CK#.
CKE0, CKE1	Input	Clock enable. CKE0 is used for a single LPDDR product. CKE1 is used for dual LPDDR products and is considered RFU for single LPDDR MCPs.
CS0#, CS1#	Input	Chip select: CS0# is used for a single LPDDR product. CS1# is used for dual LPDDR products and is considered RFU for single LPDDR MCPs.
DM[3:0]	Input	Data mask: Determines which bytes are written during WRITE operations.
RAS#	Input	Row select: Specifies the command to execute.
WE#	Input	Write enable: Specifies the command to execute.
DQ[31:0]	Input/ output	Data bus: Data inputs/outputs.
DQS[3:0]	Input/ output	Data strobe: Coordinates READ/WRITE transfers of data; one DQS per DQ byte.
TQ	Output	Temperature sensor output: TQ HIGH when LPDDR T <sub>j</sub> exceeds 85°C.
V <sub>DD</sub>	Supply	V <sub>DD</sub> : LPDDR power supply.
V <sub>DDQ</sub>	Supply	V <sub>DDQ</sub> : LPDDR I/O power supply.
V <sub>SSQ</sub>	Supply	V <sub>SSQ</sub> : LPDDR I/O ground.

Note: 1. Balls marked RFU may or may not be connected internally. These balls should not be used. Contact factory for details.

**Table 3: Non-Device-Specific Descriptions**

<b>Symbol</b>	<b>Type</b>	<b>Description</b>
V <sub>SS</sub>	Supply	V <sub>SS</sub> : Shared ground.
<b>Symbol</b>		
DNU	–	Do not use: Must be grounded or left floating.
NC	–	No connect: Not internally connected.
RFU <sup>1</sup>	–	Reserved for future use.

Note: 1. Balls marked RFU may or may not be connected internally. These balls should not be used. Contact factory for details.



## Electrical Specifications

**Table 4: Absolute Maximum Ratings**

Parameters/Conditions	Symbol	Min	Max	Unit
$V_{CC}$ , $V_{DD}$ , $V_{DDQ}$ supply voltage relative to $V_{SS}$	$V_{CC}$ , $V_{DD}$ , $V_{DDQ}$	-1.0	2.4	V
Voltage on any pin relative to $V_{SS}$	$V_{IN}$	-0.5	2.4 or (supply voltage <sup>1</sup> + 0.3V), whichever is less	V
Storage temperature range	-	-55	+150	°C

Note: 1. Supply voltage references  $V_{CC}$ ,  $V_{DD}$ , or  $V_{DDQ}$ .

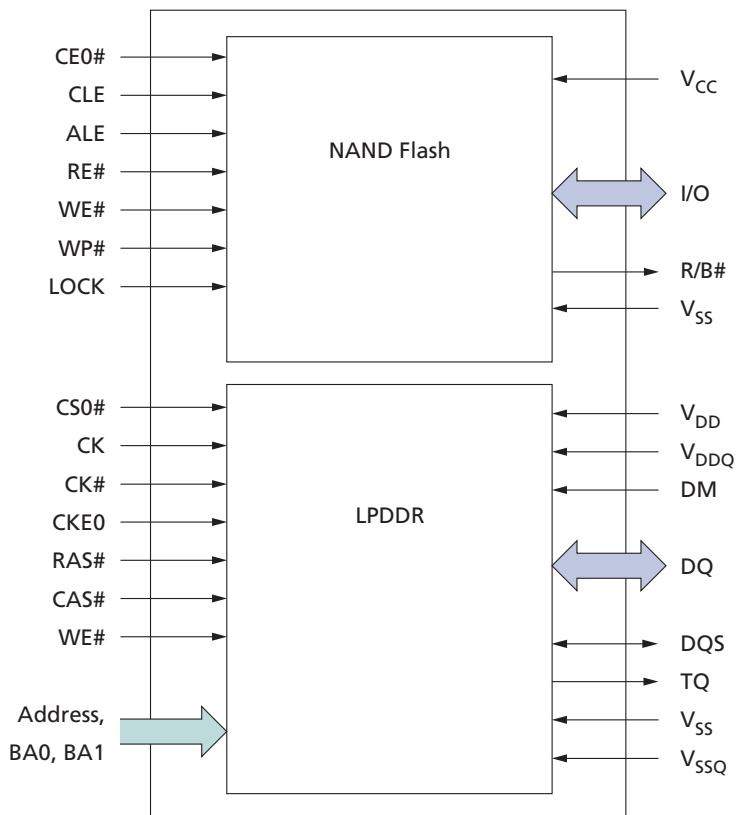
Stresses greater than those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure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

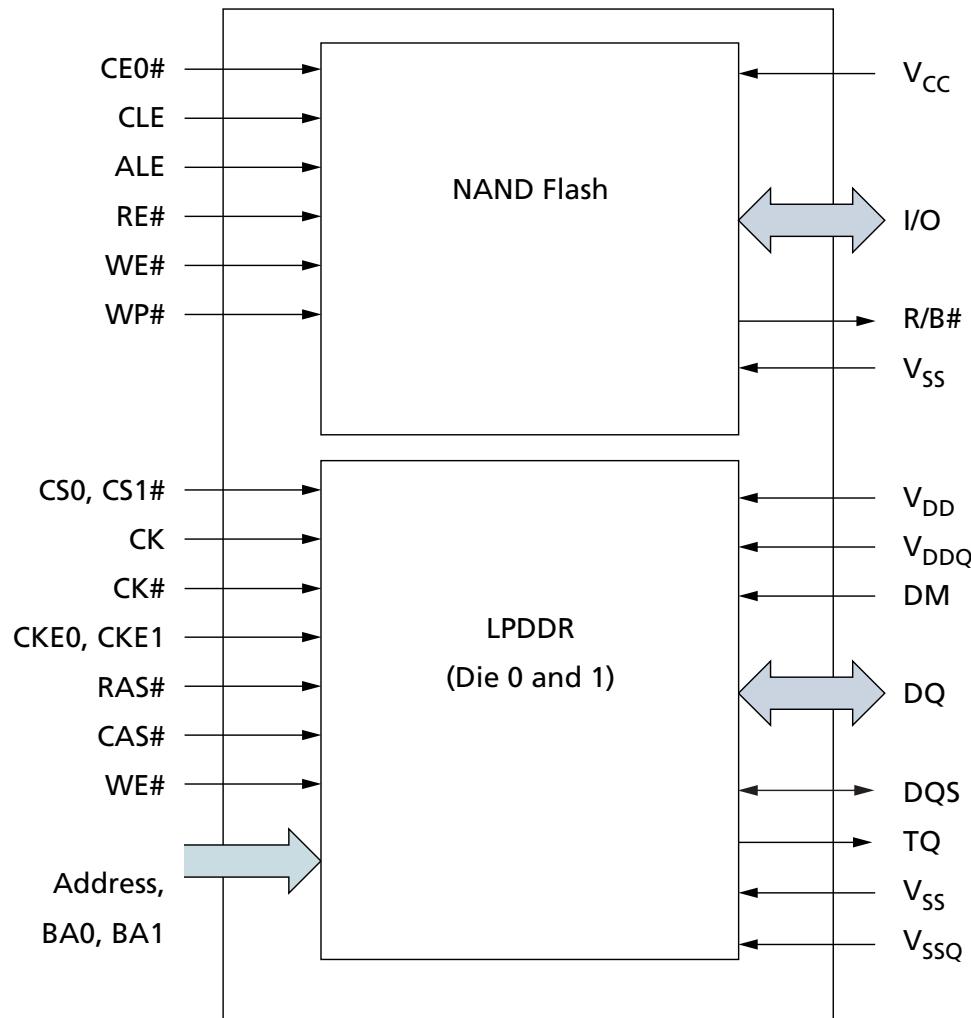
**Table 5: Recommended Operating Conditions**

Parameters	Symbol	Min	Typ	Max	Unit
Supply voltage	$V_{CC}$ , $V_{DD}$	1.70	1.80	1.95	V
I/O supply voltage	$V_{DDQ}$	1.70	1.80	1.95	V
Operating temperature range	-	-40	-	+85	°C

## Device Diagrams

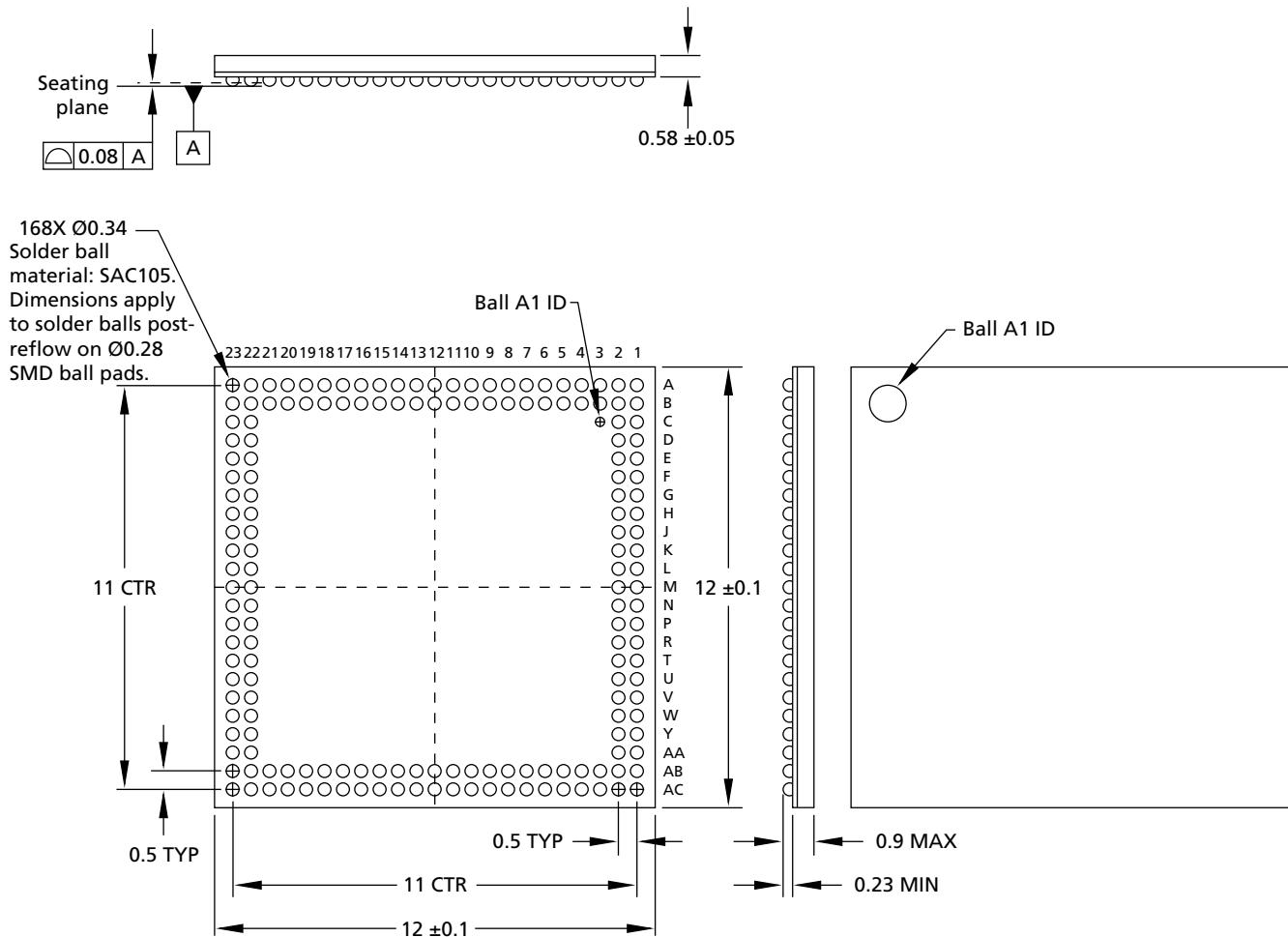
**Figure 4: 168-Ball (Single LPDDR) Functional Block Diagram**



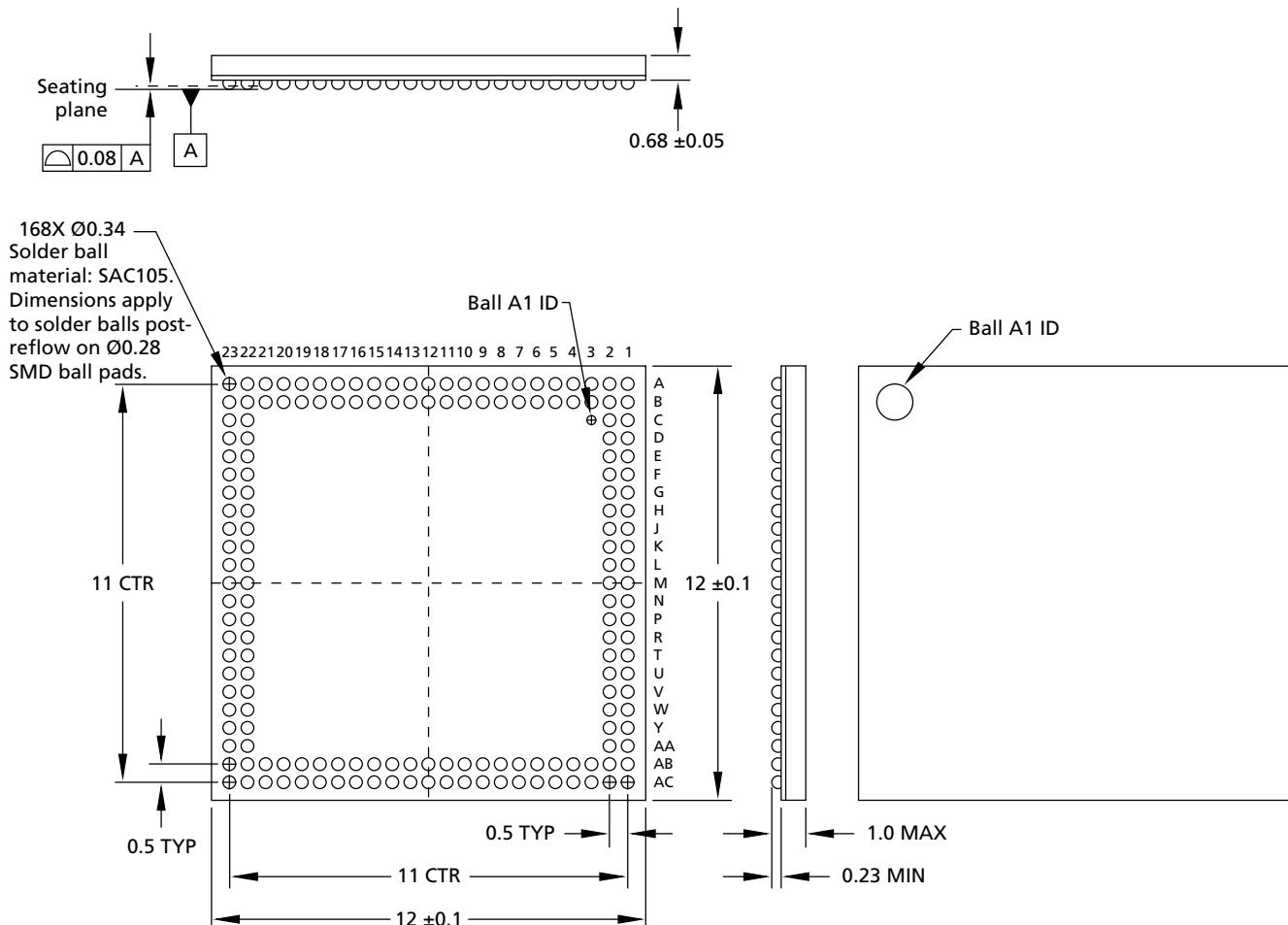
**Figure 5: 168-Ball (Dual LPDDR) Functional Block Diagram**

## Package Dimensions

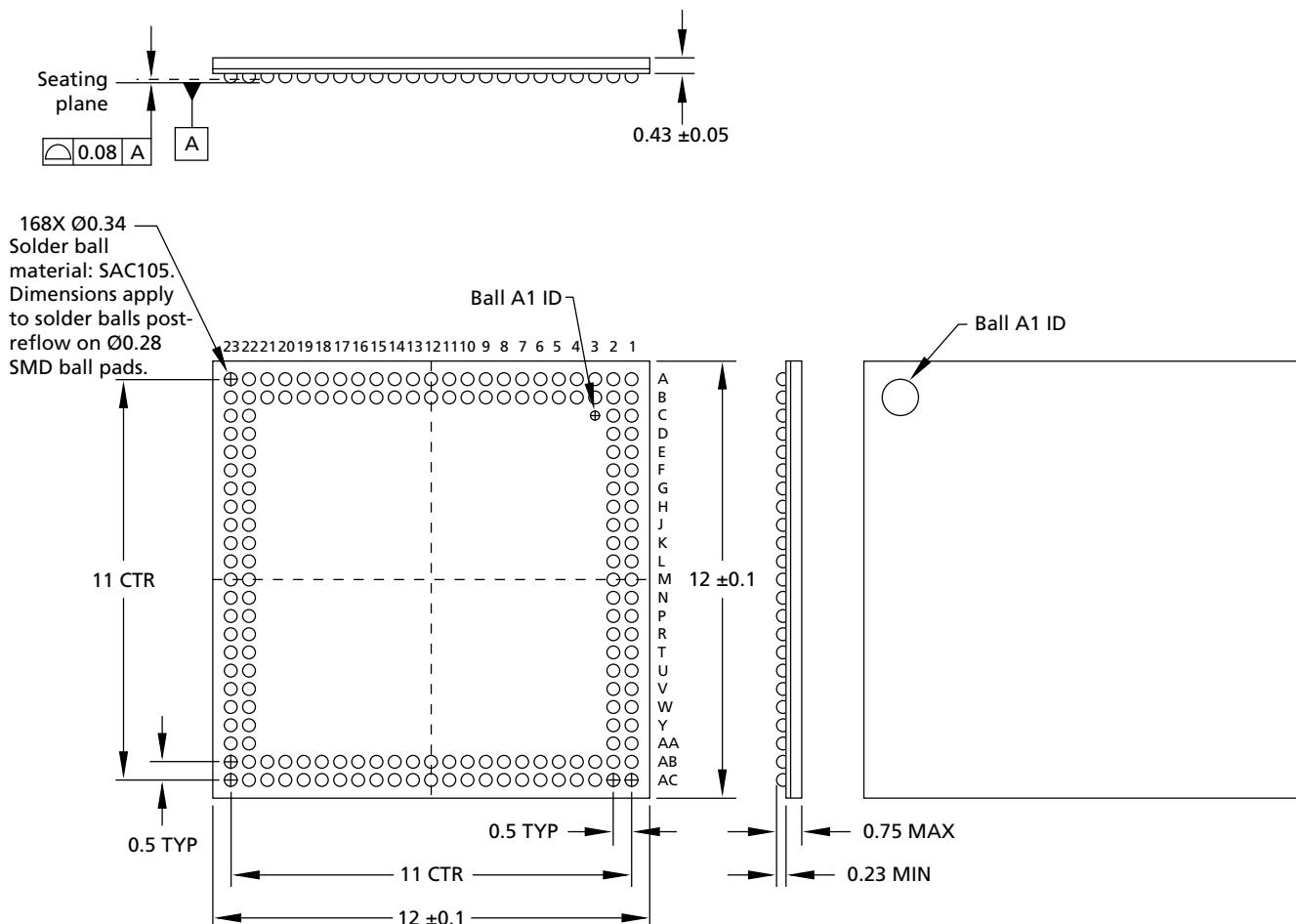
**Figure 6: 168-Ball VFBGA (Package Code: JG)**



Note: 1. All dimensions are in millimeters.

**Figure 7: 168-Ball VFBGA (Package Code: JV)**

Note: 1. All dimensions are in millimeters.

**Figure 8: 168-Ball WFBGA (Package Code: KQ)**

Note: 1. All dimensions are in millimeters.



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

### **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
  - $t_{RC}/t_{WC}$ : 20ns (3.3V), 25ns (1.8V)
- Array performance
  - Read page: 25 $\mu$ s<sup>2</sup>
  - Program page: 200 $\mu$ s (TYP: 1.8V, 3.3V)<sup>2</sup>
  - Erase block: 700 $\mu$ s (TYP)
- Command set: ONFI NAND Flash Protocol
- Advanced command set
  - Program page cache mode<sup>3</sup>
  - Read page cache mode<sup>3</sup>
  - One-time programmable (OTP) mode
  - Two-plane commands<sup>3</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
- Blocks 0–15 (block addresses 00h–0Fh) are 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
- 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

Notes: 1. The ONFI 1.0 specification is available at [www.onfi.org](http://www.onfi.org).



2. See Electrical Specifications – Program/Erase Characteristics (page 121) for  $tR_ECC$  and  $tPROG_ECC$  specifications.
3. These commands supported only with ECC disabled.

## 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.

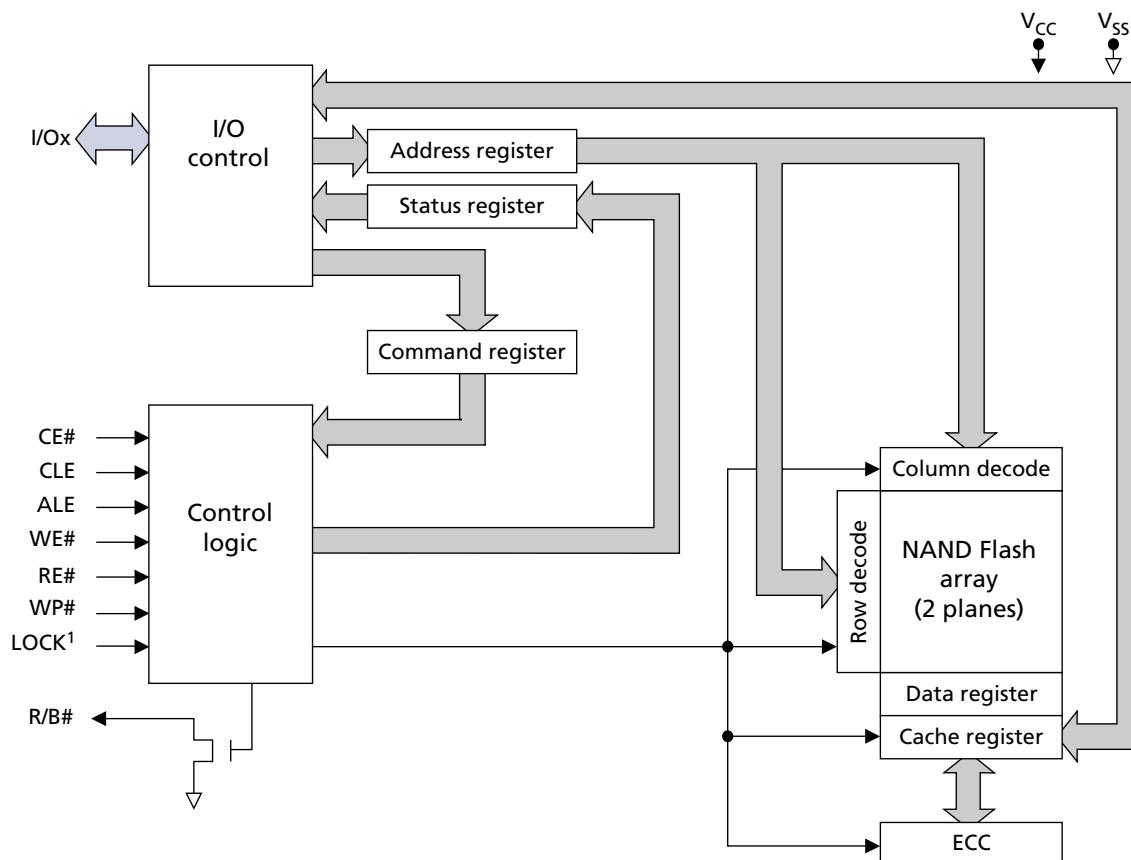
## Architecture

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.

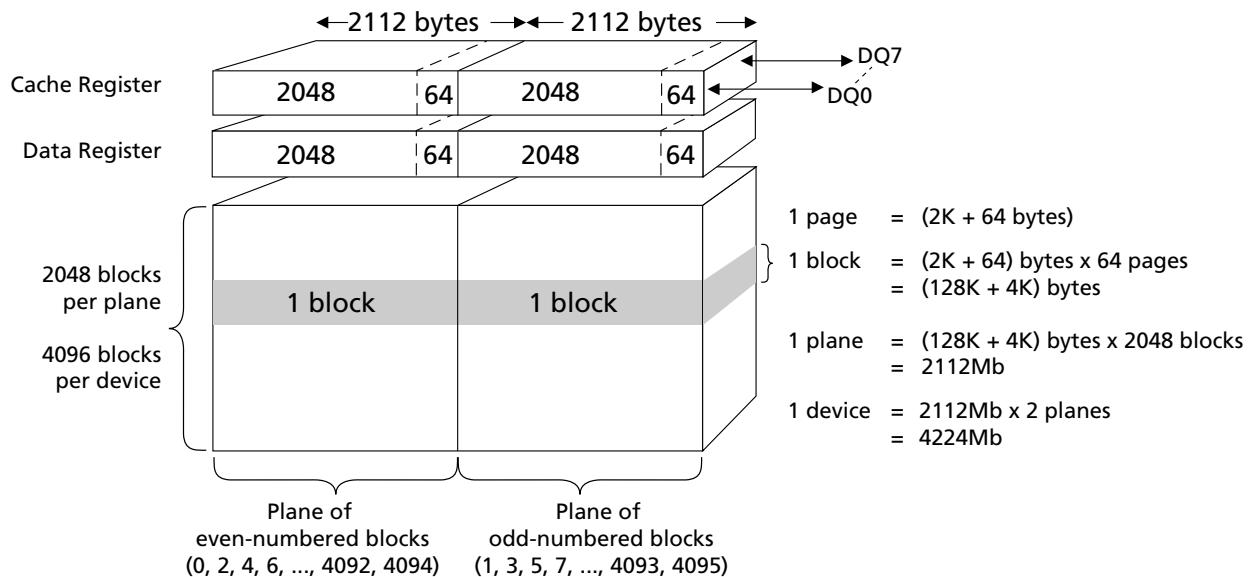
Data is transferred to or from the NAND Flash memory array, byte by byte (x8) or word by word (x16), through a data register and a cache register.

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 9: NAND Flash Die (LUN) Functional Block Diagram**



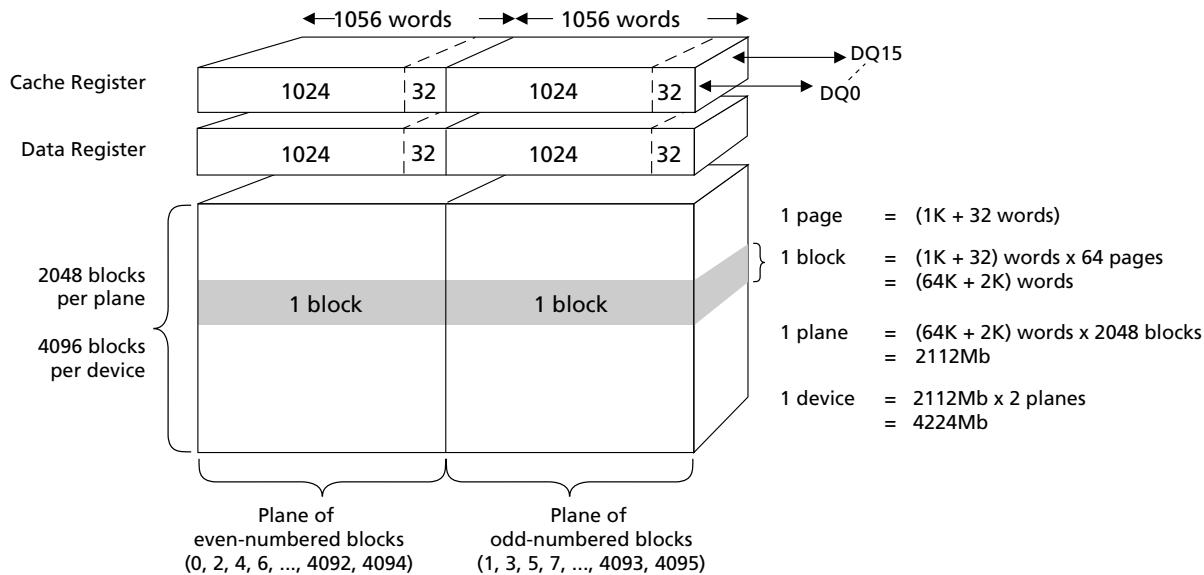
Note: 1. The **LOCK** pin is used on the 1.8V device.

**Device and Array Organization****Figure 10: Array Organization – MT29F4G08 (x8)****Table 6: 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:

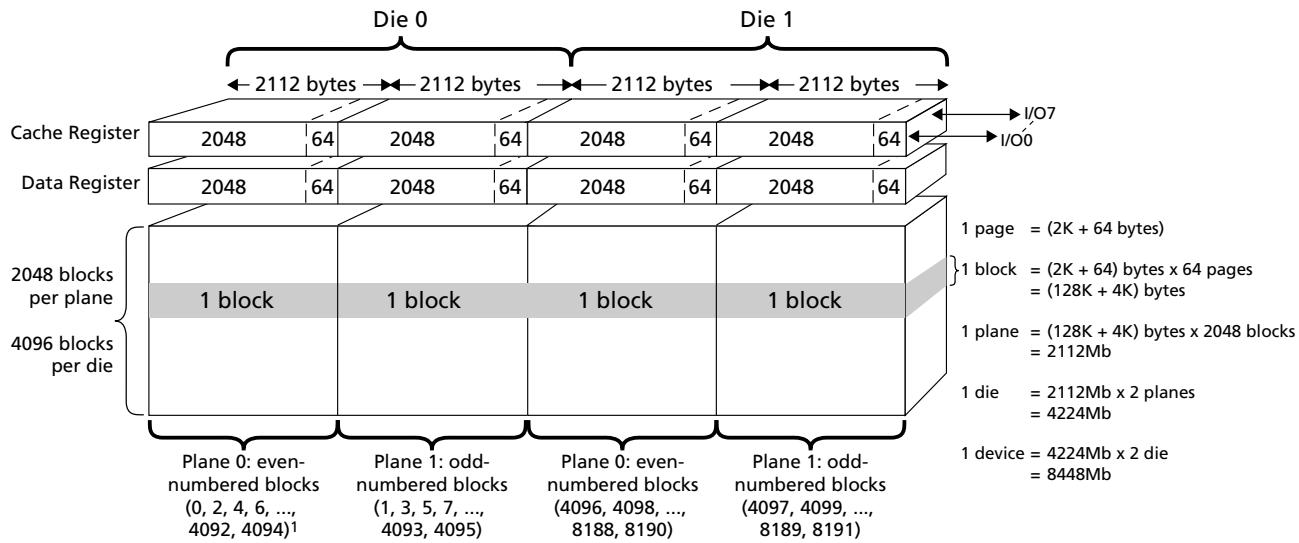
1. Block address concatenated with page address = actual page address. CAx = column address; PAx = page address; BAx = block address.
2. If CA11 is 1, then CA[10:6] must be 0.
3. BA6 controls plane selection.

**Figure 11: Array Organization – MT29F4G16 (x16)****Table 7: Array Addressing – MT29F4G16 (x16)**

Cycle	I/O[15:8]	I/07	I/06	I/05	I/04	I/03	I/02	I/01	I/00
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.

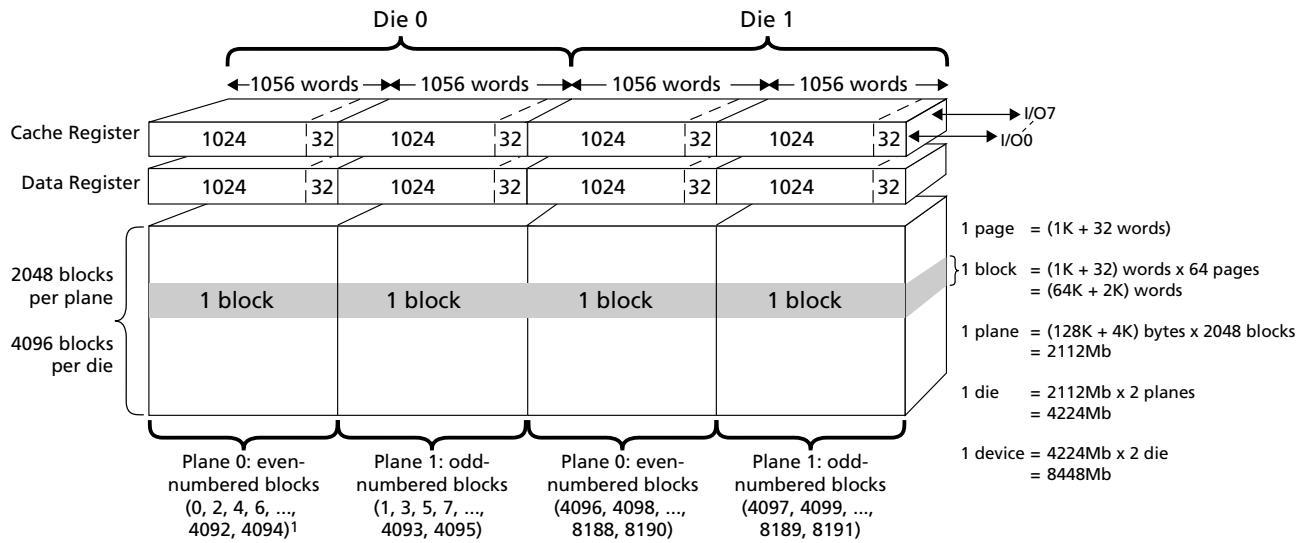
**Figure 12: Array Organization – MT29F8G08 (x8)**

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 8: Array Addressing – MT29F8G08 (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	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.

**Figure 13: Array Organization – MT29F8G16 (x16)**

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 9: Array Addressing – MT29F8G16 ( x16)**

Cycle	I/O[15:8]	I/07	I/06	I/05	I/04	I/03	I/02	I/01	I/00
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	PA8
Fifth	LOW	LOW	LOW	LOW	LOW	LOW	BA18 <sup>3</sup>	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. 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 10: 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/Standyby

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.

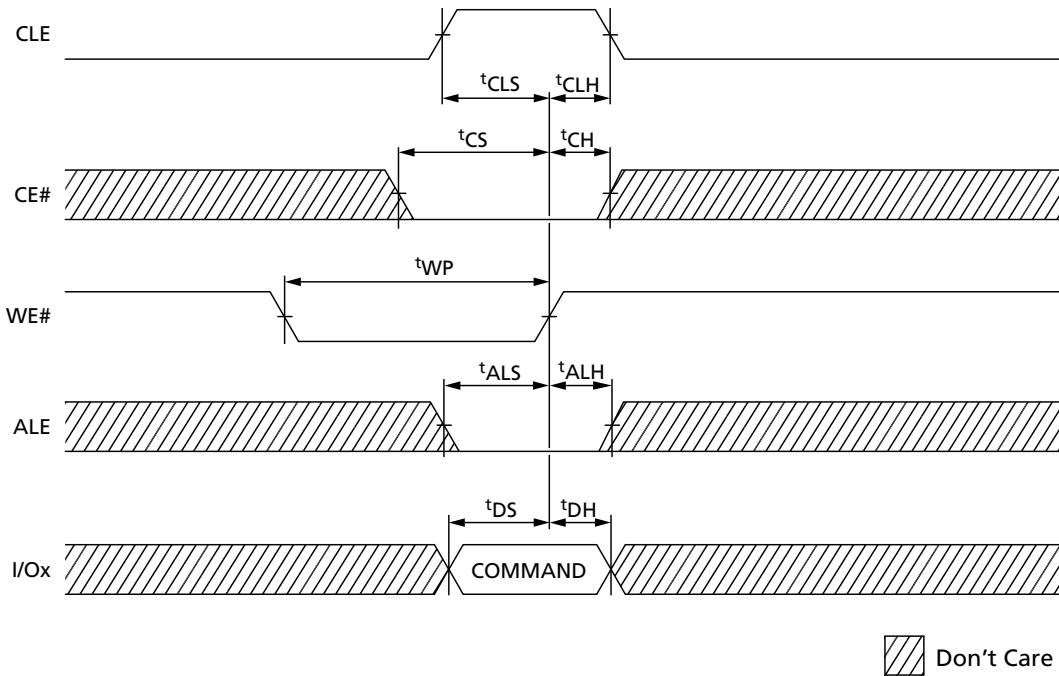
A HIGH CLE signal indicates that a command cycle is taking place. A HIGH ALE signal signifies that an ADDRESS INPUT cycle is occurring.

## 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.

**Figure 14: Asynchronous Command Latch Cycle**

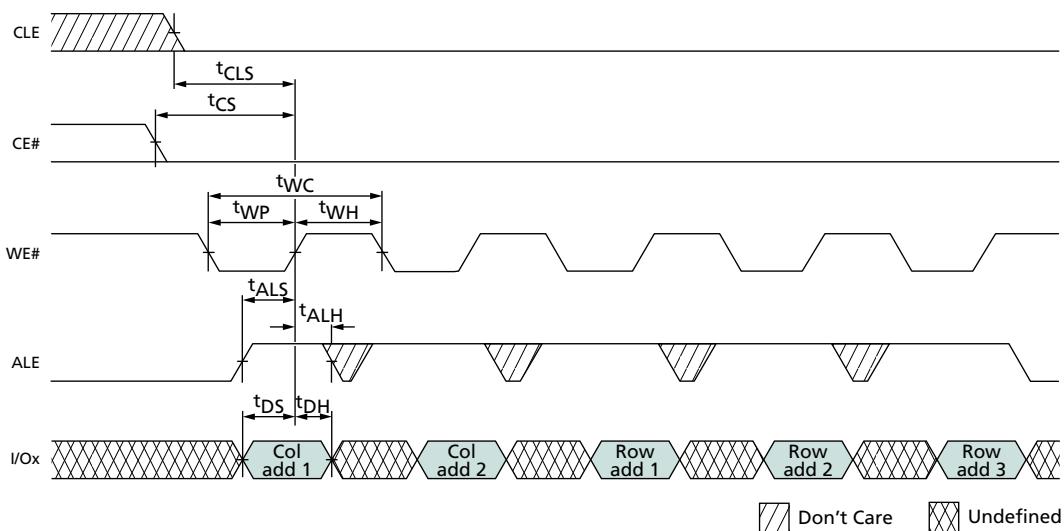
## Asynchronous Addresses

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 15: Asynchronous Address Latch Cycle**



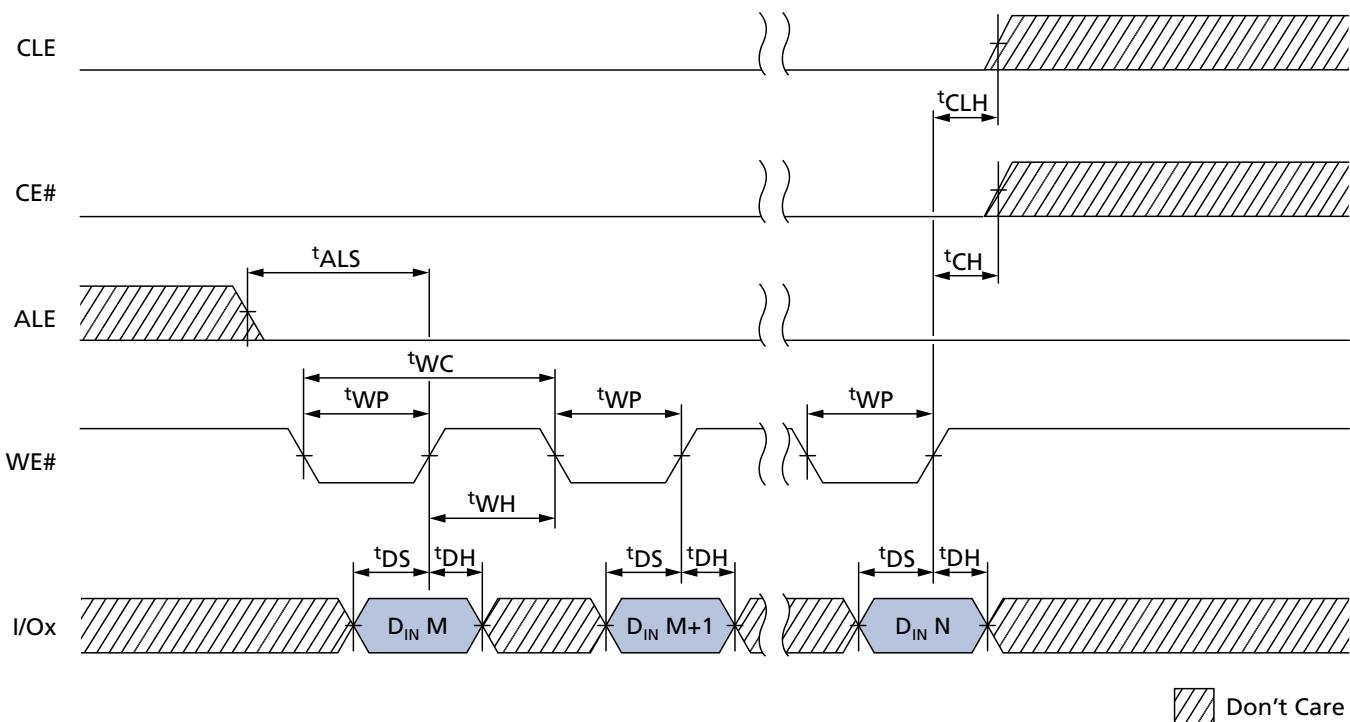
## Asynchronous Data Input

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 16: Asynchronous Data Input Cycles**



## Asynchronous Data Output

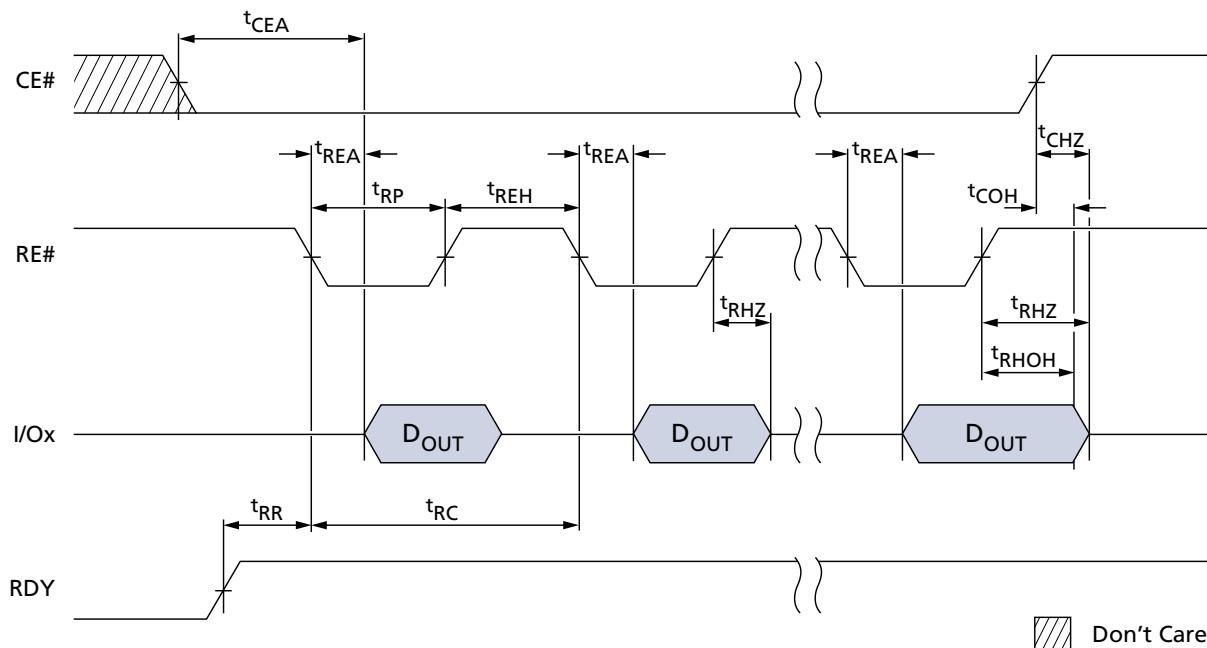
Data can be output from a die (LUN) if it is in a READY state. Data output is supported following a READ operation from the NAND Flash array. Data is output from the cache register of the selected die (LUN) to I/O[7:0] on the falling edge of RE# when CE# is LOW, ALE is LOW, CLE is LOW, and WE# is HIGH.

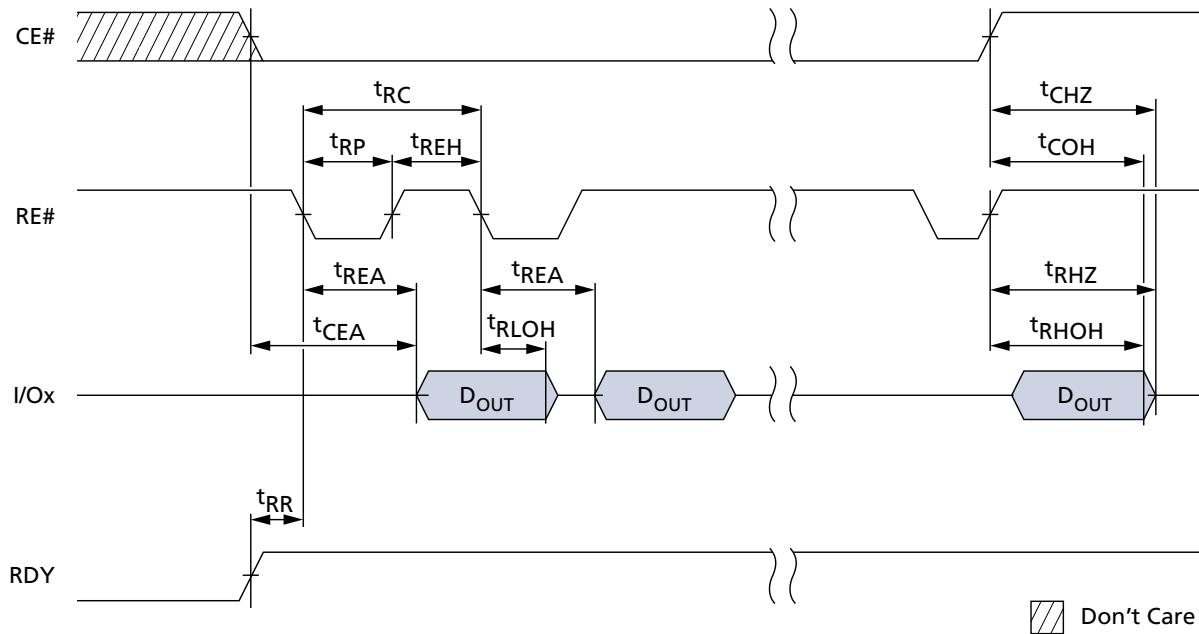
If the host controller is using a  $t_{RC}$  of 30ns or greater, the host can latch the data on the rising edge of RE# (see the figure below for proper timing). If the host controller is using a  $t_{RC}$  of less than 30ns, the host can latch the data on the next falling edge of RE#.

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 17: Asynchronous Data Output Cycles**



**Figure 18: 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<sub>CC</sub> 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<sub>WW</sub> 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, Rp, 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

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 Rp and capacitive loading of the R/B# circuit determines the rise time of the R/B# signal. The actual value used for Rp depends on the system timing requirements. Large values of Rp 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 = Rp (resistance of pull-up resistor), and C = total capacitive load.

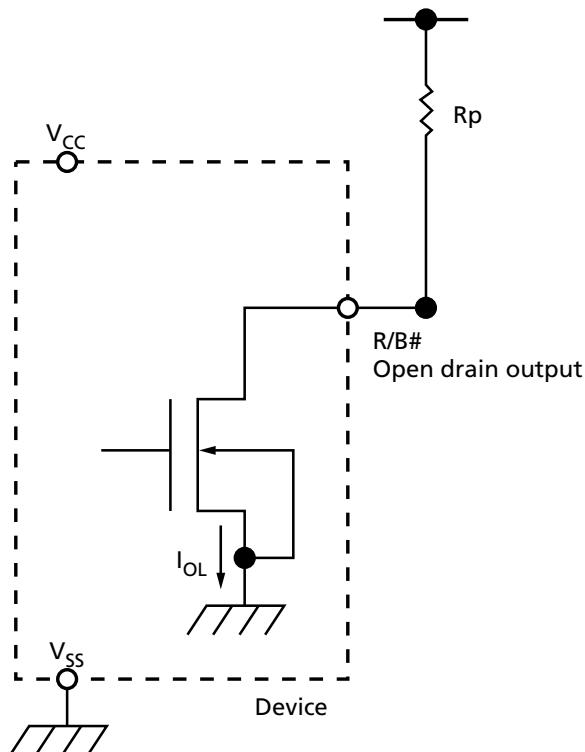
The fall time of the R/B# signal is determined mainly by the output impedance of the R/B# signal and the total load capacitance. Approximate Rp values using a circuit load of 100pF are provided in Figure 24 (page 38).

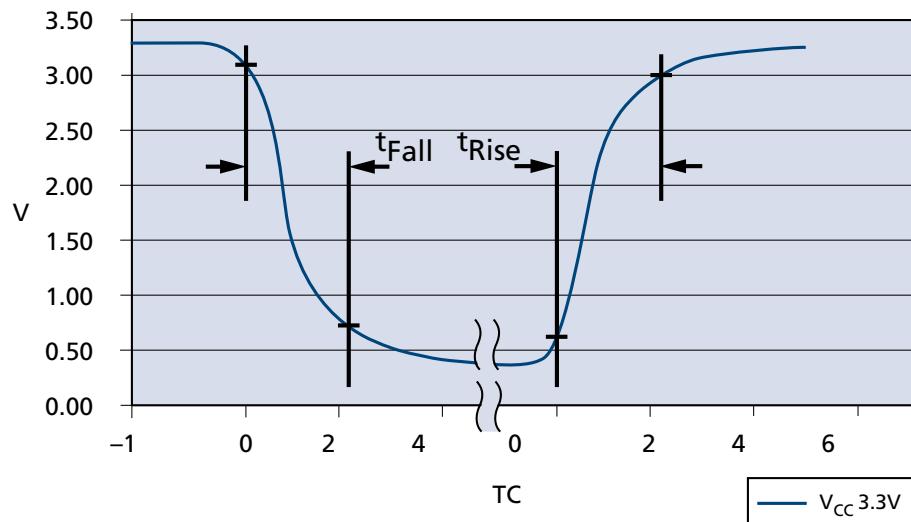
The minimum value for Rp is determined by the output drive capability of the R/B# signal, the output voltage swing, and V<sub>CC</sub>.

$$Rp = \frac{V_{CC}(\text{MAX}) - V_{OL}(\text{MAX})}{I_{OL} + \Sigma_{IL}}$$

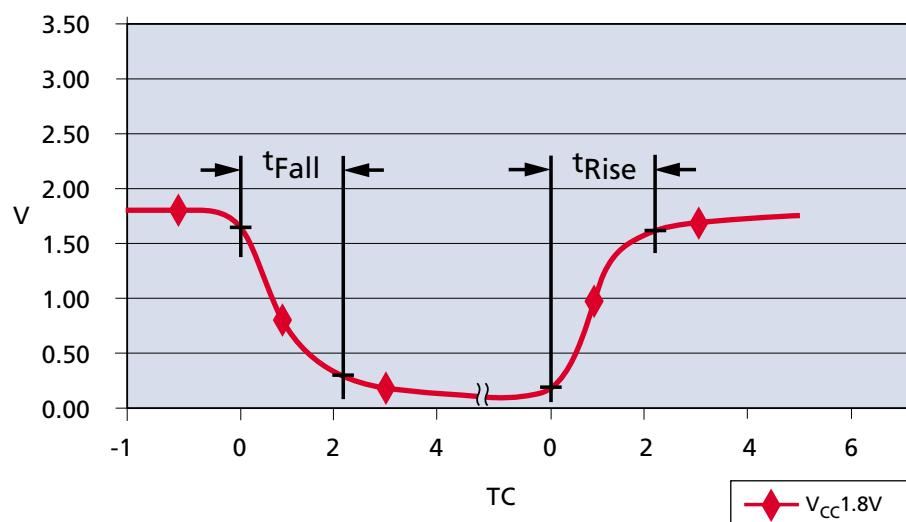
Where  $\Sigma_{IL}$  is the sum of the input currents of all devices tied to the R/B# pin.

**Figure 19: READ/BUSY# Open Drain**

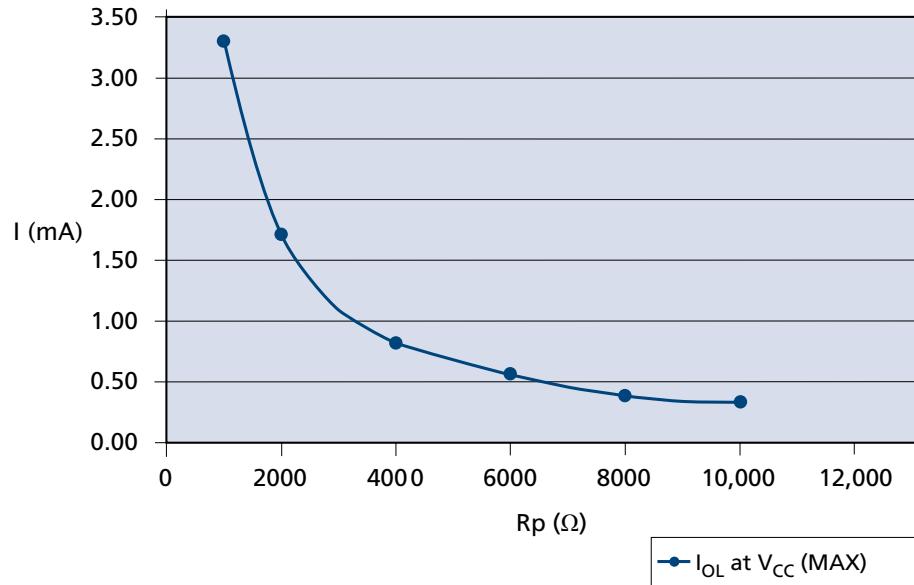
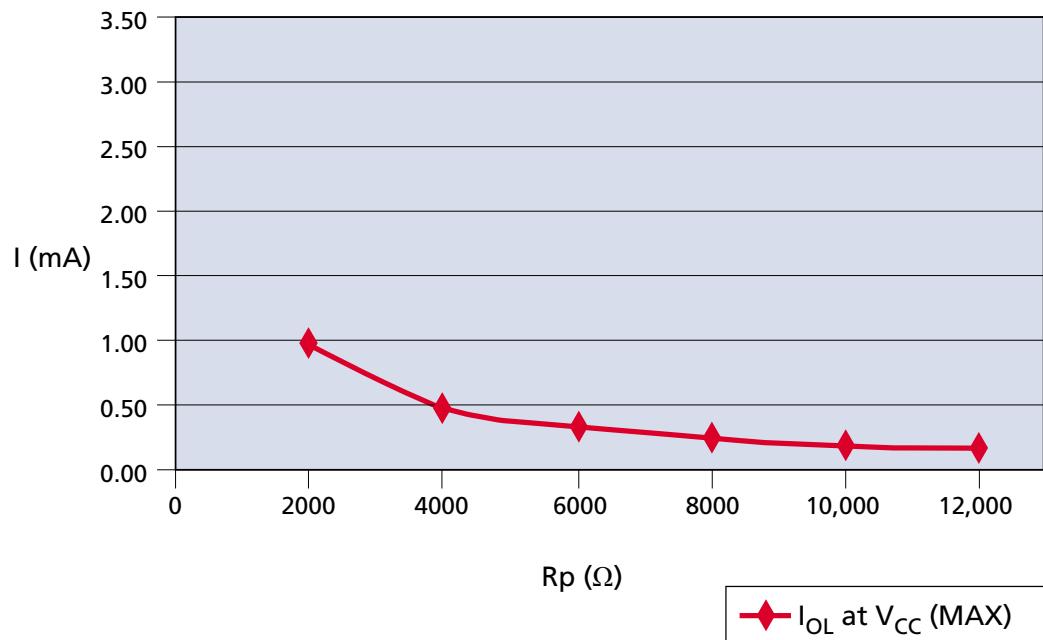


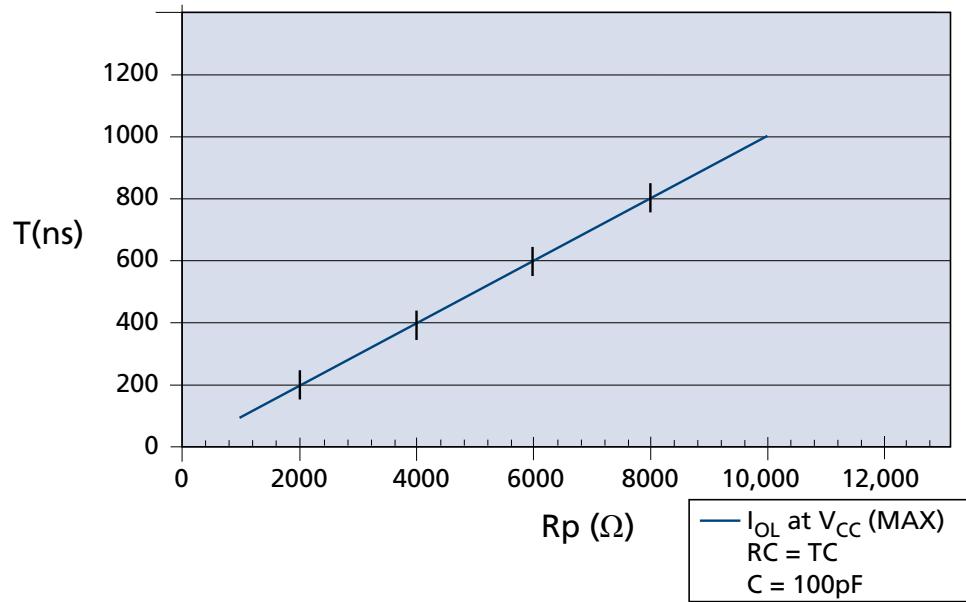
**Figure 20: tFall and tRise (3.3V V<sub>CC</sub>)**

- Notes:
1. tFall and tRise calculated at 10% and 90% points.
  2. tRise dependent on external capacitance and resistive loading and output transistor impedance.
  3. tRise primarily dependent on external pull-up resistor and external capacitive loading.
  4. tFall = 10ns at 3.3V.
  5. See TC values in Figure 24 (page 38) for approximate Rp value and TC.

**Figure 21: tFall and tRise (1.8V V<sub>CC</sub>)**

- Notes:
1. tFall and tRise are calculated at 10% and 90% points.
  2. tRise is primarily dependent on external pull-up resistor and external capacitive loading.
  3. tFall ≈ 7ns at 1.8V.
  4. See TC values in Figure 24 (page 38) for TC and approximate Rp value.

**Figure 22:  $I_{OL}$  vs.  $R_p$  ( $V_{CC} = 3.3V$   $V_{CC}$ )****Figure 23:  $I_{OL}$  vs.  $R_p$  (1.8V  $V_{CC}$ )**

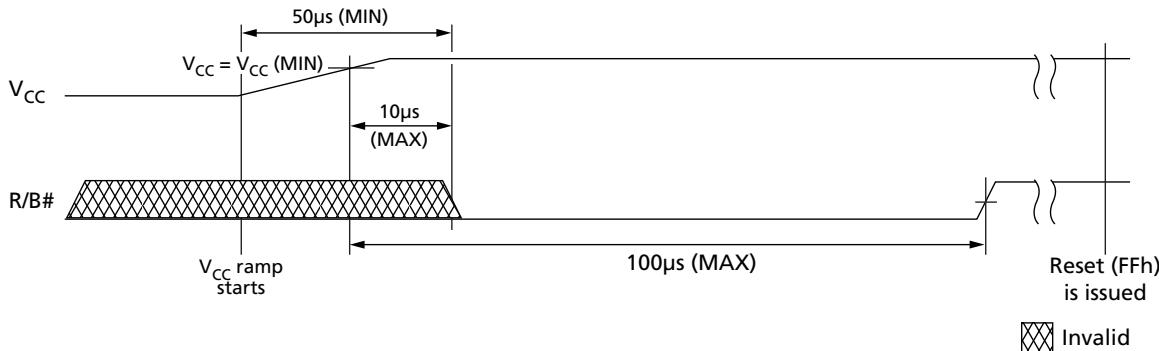
**Figure 24: TC vs. Rp**

## 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 25: R/B# Power-On Behavior**





## Command Definitions

**Table 11: 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 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	

**Table 11: Command Set (Continued)**

<b>Command</b>	<b>Command Cycle #1</b>	<b>Number of Valid Address Cycles</b>	<b>Data Input Cycles</b>	<b>Command Cycle #2</b>	<b>Valid While Selected LUN is Busy<sup>1</sup></b>	<b>Valid While Other LUNs are Busy<sup>2</sup></b>	<b>Notes</b>
<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 12: Two-Plane Command Set**

Note 4 applies to all parameters and conditions

<b>Command</b>	<b>Com-mand Cycle #1</b>	<b>Number of Valid Address Cycles</b>	<b>Com-mand Cycle #2</b>	<b>Number of Valid Address Cycles</b>	<b>Com-mand Cycle #3</b>	<b>Valid While Selected LUN is Busy</b>	<b>Valid While Other LUNs are Busy</b>	<b>Notes</b>
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

**Table 12: Two-Plane Command Set (Continued)**

Note 4 applies to all parameters and conditions

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

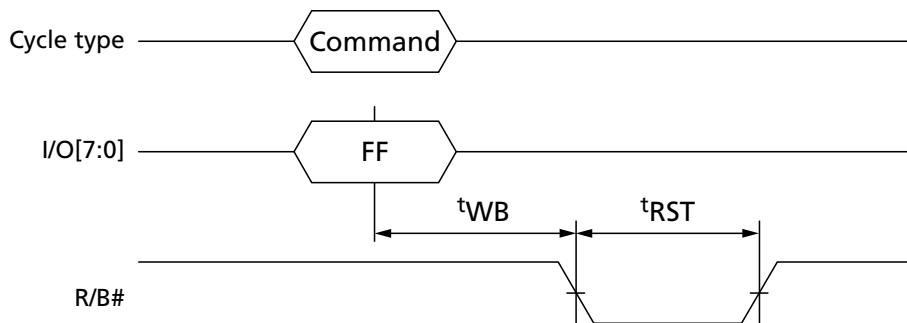
The RESET command is used to put the memory device into a known condition and to abort the command sequence in progress.

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 26: RESET (FFh) Operation**



## Identification Operations

### READ ID (90h)

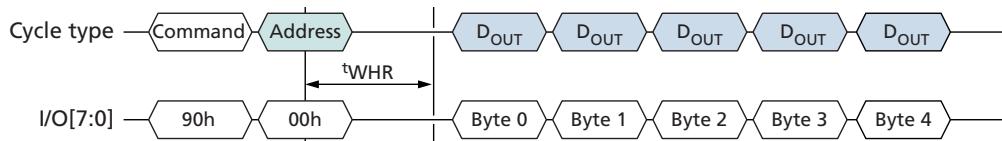
The READ ID (90h) command is used to read identifier codes programmed into the target. This command is accepted by the target only when all die (LUNs) on the target are idle.

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.

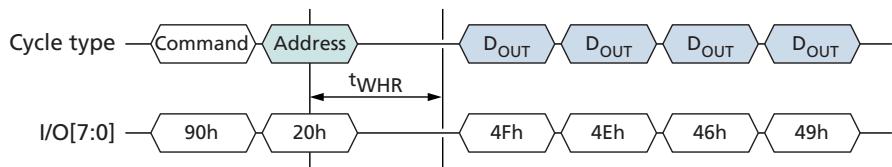
When the 90h command is followed by a 20h address cycle, the target returns the 4-byte ONFI identifier code.

**Figure 27: READ ID (90h) with 00h Address Operation**



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

**Figure 28: READ ID (90h) with 20h Address Operation**



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



## READ ID Parameter Tables

**Table 13: 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

**Table 13: READ ID Parameters for Address 00h (Continued)**

b = binary; h = hexadecimal

	<b>Options</b>	I/07	I/06	I/05	I/04	I/03	I/02	I/01	I/00	<b>Value</b>
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 14: READ ID Parameters for Address 20h**

h = hexadecimal

<b>Byte</b>	<b>Options</b>	I/07	I/06	I/05	I/04	I/03	I/02	I/01	I/00	<b>Value</b>
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)

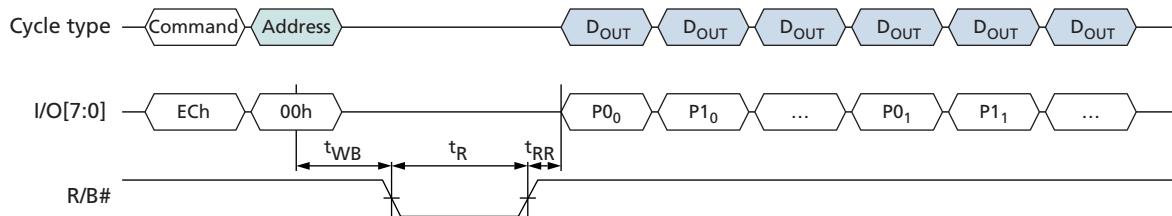
The READ PARAMETER PAGE (ECh) command is used to read the ONFI parameter page programmed into the target. This command is accepted by the target only when all die (LUNs) on the target are idle.

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 29: READ PARAMETER (ECh) Operation**





## Bare Die Parameter Page Data Structure Tables

**Table 15: Parameter Page Data Structure**

<b>Byte</b>	<b>Description</b>	<b>Value</b>
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
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, 34h, 47h, 31h, 36h, 41h, 44h, 42h, 44h, 41h, 33h, 57h, 20h, 20h, 20h, 20h
64	Manufacturer ID	2Ch
65–66	Date code	00h, 00h
67–79	Reserved	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

**Table 15: Parameter Page Data Structure (Continued)**

<b>Byte</b>	<b>Description</b>	<b>Value</b>
96–99	Number of blocks per unit	00h, 10h, 00h, 00h
100	Number of logical units	MT29F4G08ABBDA3W 01h
		MT29F4G16ABBDA3W 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
128	I/O pin capacitance	MT29F4G08ABBDA3W 0Ah MT29F4G16ABBDA3W 0Ah MT29F8G08ADBDA3W 14h MT29F8G16ADBDA3W 14h MT29F4G08ABADA3W 0Ah MT29F4G16ABADA3W 0Ah MT29F8G08ADADA3W 14h MT29F8G16ADADA3W 14h
129–130	Timing mode support	MT29F4G08ABBDA3W 1Fh, 00h MT29F4G16ABBDA3W 1Fh, 00h MT29F8G08ADBDA3W 1Fh, 00h MT29F8G16ADBDA3W 1Fh, 00h MT29F4G08ABADA3W 3Fh, 00h MT29F4G16ABADA3W 3Fh, 00h MT29F8G08ADADA3W 3Fh, 00h MT29F8G16ADADA3W 3Fh, 00h



**Table 15: Parameter Page Data Structure (Continued)**

## 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.

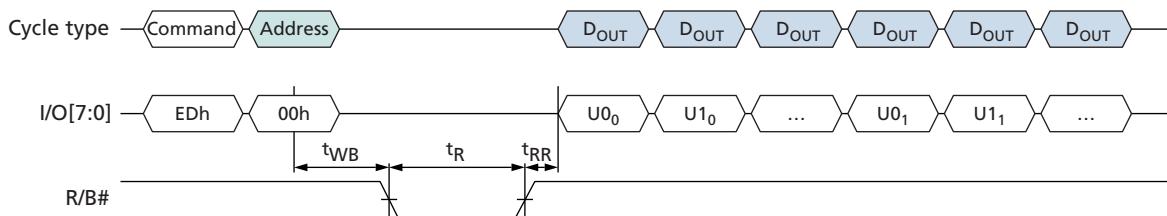
When the EDh command is followed by an 00h address cycle, the target goes busy for  $t_R$ . If the READ STATUS (70h) command is used to monitor for command completion, the READ MODE (00h) command must be used to re-enable data output mode.

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 30: 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 16: 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 17: Feature Address 90h – Array Operation Mode**

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

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

## SET FEATURES (EFh)

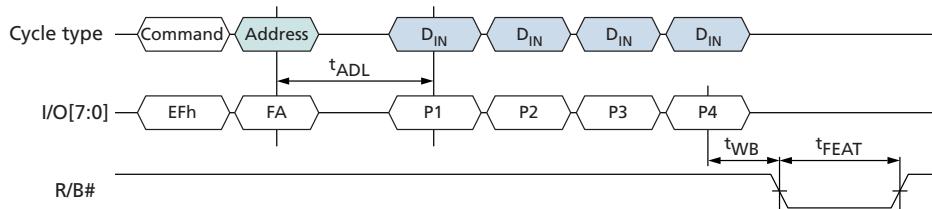
The SET FEATURES (EFh) command writes the subfeature parameters (P1–P4) to the specified feature address to enable or disable target-specific features. This command is accepted by the target only when all die (LUNs) on the target are idle.

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 31: SET FEATURES (EFh) Operation**

## GET FEATURES (EEh)

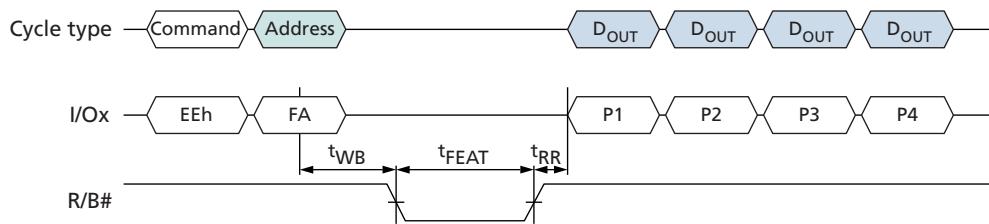
The GET FEATURES (EEh) command reads the subfeature parameters (P1–P4) from the specified feature address. This command is accepted by the target only when all die (LUNs) on the target are idle.

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 32: GET FEATURES (EEh) Operation**



**Table 18: Feature Addresses 01h: Timing Mode**

<b>Subfeature Parameter</b>	<b>Options</b>	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 19: Feature Addresses 80h: Programmable I/O Drive Strength**

<b>Subfeature Parameter</b>	<b>Options</b>	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)			Reserved (0)			0	0	00h	1	
	Three-quarters			Reserved (0)			0	1	01h		
	One-half			Reserved (0)			1	0	02h		
	One-quarter			Reserved (0)			1	1	03h		
<b>P2</b>											
			Reserved (0)						00h		
<b>P3</b>											
			Reserved (0)						00h		
<b>P4</b>											
			Reserved (0)						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 20: Feature Addresses 81h: Programmable R/B# Pull-Down Strength**

<b>Subfeature Parameter</b>	<b>Options</b>	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>											
			Reserved (0)						00h		
<b>P3</b>											
			Reserved (0)						00h		
<b>P4</b>											
			Reserved (0)						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>t</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>t</sup>R\_ECC) to determine whether an uncorrectable read error occurred.

**Table 21: 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 1 = Not protected
6	RDY	RDY <sup>1</sup> cache	RDY	RDY <sup>1</sup> cache	RDY	0 = Busy 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 1 = Rewrite recommended
2	–	–	–	–	–	Don't Care
1	FAILC (N - 1)	FAILC (N - 1)	Reserved	–	–	Don't Care
0	FAIL	FAIL (N)	FAIL <sup>4</sup>	–	FAIL	0 = Successful PROGRAM/ERASE/READ 1 = Error in PROGRAM/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.

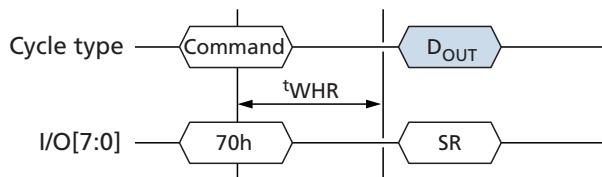
## READ STATUS (70h)

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 33: READ STATUS (70h) Operation**



## READ STATUS ENHANCED (78h)

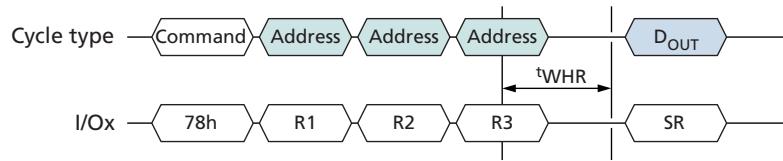
The READ STATUS ENHANCED (78h) command returns the status of the addressed die (LUN) on a target even when it is busy (RDY = 0). This command is accepted by all die (LUNs), even when they are BUSY (RDY = 0).

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.

**Figure 34: 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.

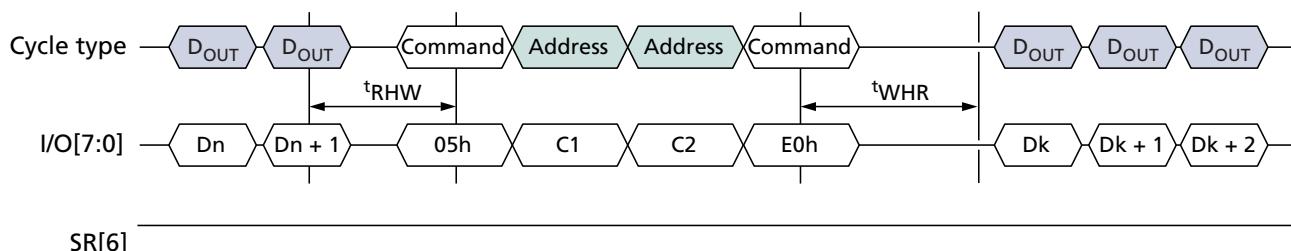
### RANDOM DATA READ (05h-E0h)

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 35: 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.

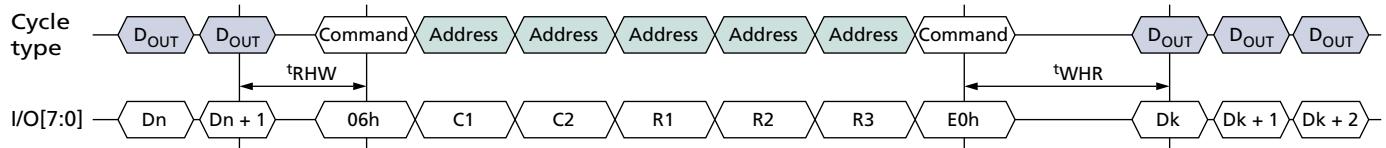
Following a two-plane read page operation, the RANDOM DATA READ TWO-PLANE (06h-E0h) command is used to select the cache register to be enabled for data output. After data output is complete on the selected plane, the command can be issued again to begin data output on another plane.

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



## RANDOM DATA INPUT (85h)

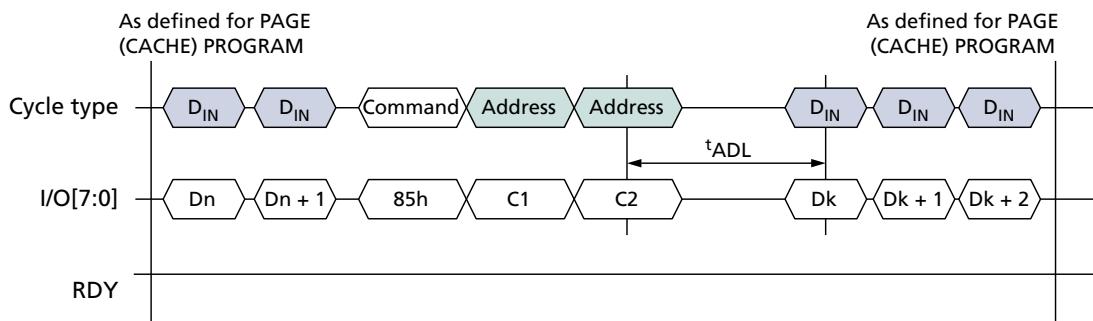
The RANDOM DATA INPUT (85h) command changes the column address of the selected cache register and enables data input on the last-selected die (LUN). This command is accepted by the selected die (LUN) when it is ready (RDY = 1; ARDY = 1). It is also accepted by the selected die (LUN) during cache program 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 37: 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  $t_{ADL}$  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.

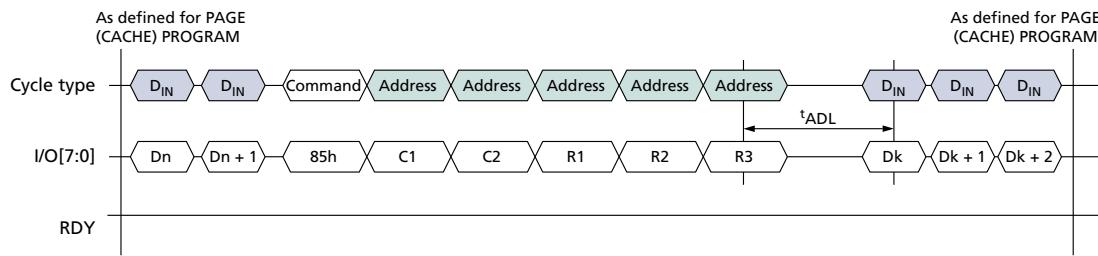
The PROGRAM FOR INTERNAL DATA INPUT (85h) command is allowed after the required address cycles are specified, but prior to the final command cycle (10h, 11h, 15h) of the following commands while data input is permitted: PROGRAM PAGE (80h-10h), PROGRAM PAGE TWO-PLANE (80h-11h), PROGRAM PAGE CACHE (80h-15h),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

**Figure 38: 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  $t^R$  and the selected die (LUN) is busy (RDY = 0, ARDY = 0). After  $t^R$  (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  $t^{RCBSY}$  while the next page begins copying data from the array to the data register. After  $t^{RCBSY}$ , 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  $t^{RCBSY}$  while the data register is copied into the cache register. After  $t^{RCBSY}$ , 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,  $t^{RCBSY}$ , 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  $t_R$  and the selected die (LUN) is busy (RDY = 0, ARDY = 0). After  $t_R$  (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  $t_{RCBSY}$  while the next pages begin copying data from the array to the data registers. After  $t_{RCBSY}$ , 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  $t_{RCBSY}$  while the data registers are copied into the cache registers. After  $t_{RCBSY}$ , 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.



For READ PAGE CACHE series (31h, 00h-31h, 3Fh), during the die (LUN) busy time,  $t_{RCBSY}$ , 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  $t_R$  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.

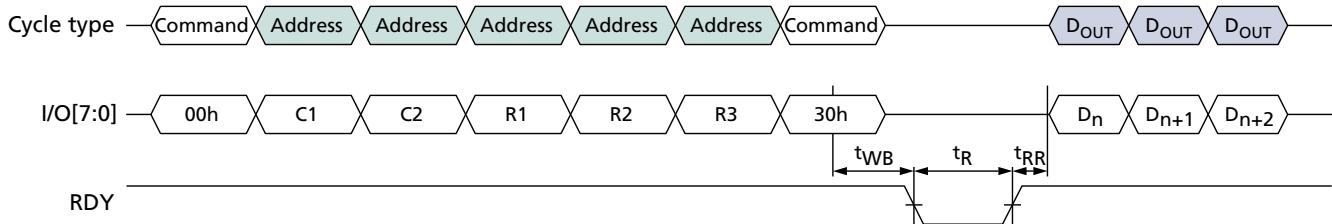
When internal ECC is enabled, the READ STATUS (70h) command is required after the completion of the data transfer ( $t_{R\_ECC}$ ) to determine whether an uncorrectable read error occurred. ( $t_{R\_ECC}$  is the data transferred with internal ECC enabled.)

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.

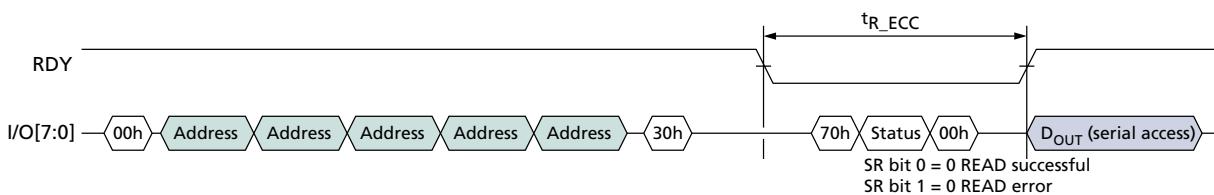
The READ PAGE (00h-30h) command is used as the final command of a two-plane read operation. It is preceded by one or more READ PAGE TWO-PLANE (00h-00h-30h) commands. Data is transferred from the NAND Flash array for all of the addressed planes to their respective cache registers. When the die (LUN) is 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,

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 39: READ PAGE (00h-30h) Operation**



**Figure 40: READ PAGE (00h-30h) Operation with Internal ECC Enabled**

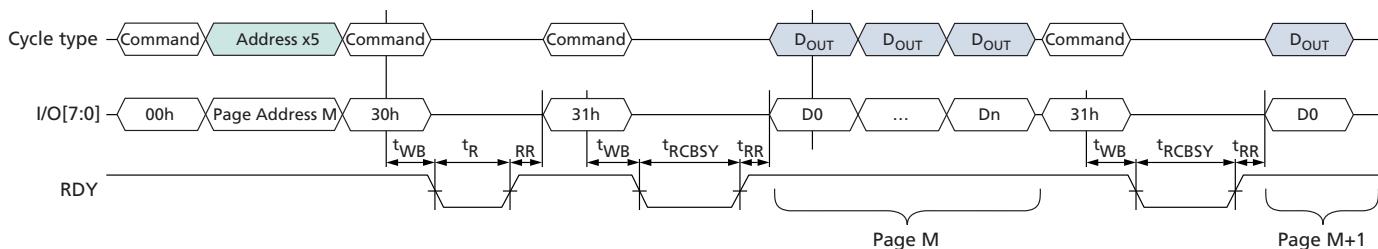


## READ PAGE CACHE SEQUENTIAL (31h)

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 41: READ PAGE CACHE SEQUENTIAL (31h) Operation**


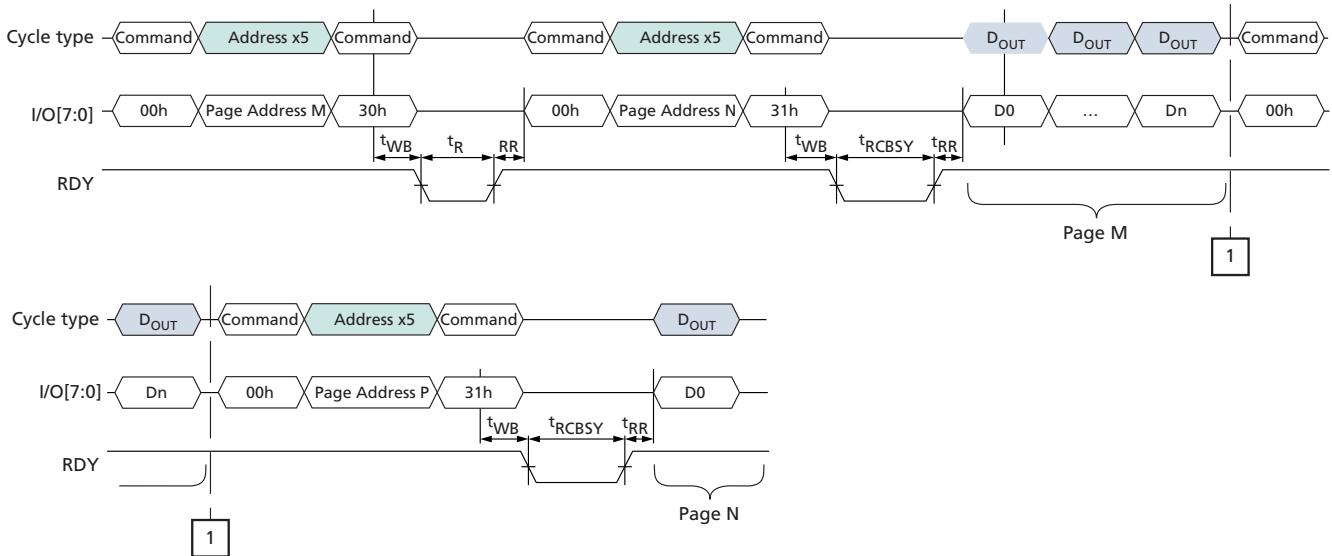
### READ PAGE CACHE RANDOM (00h-31h)

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.

**Figure 42: READ PAGE CACHE RANDOM (00h-31h) Operation**


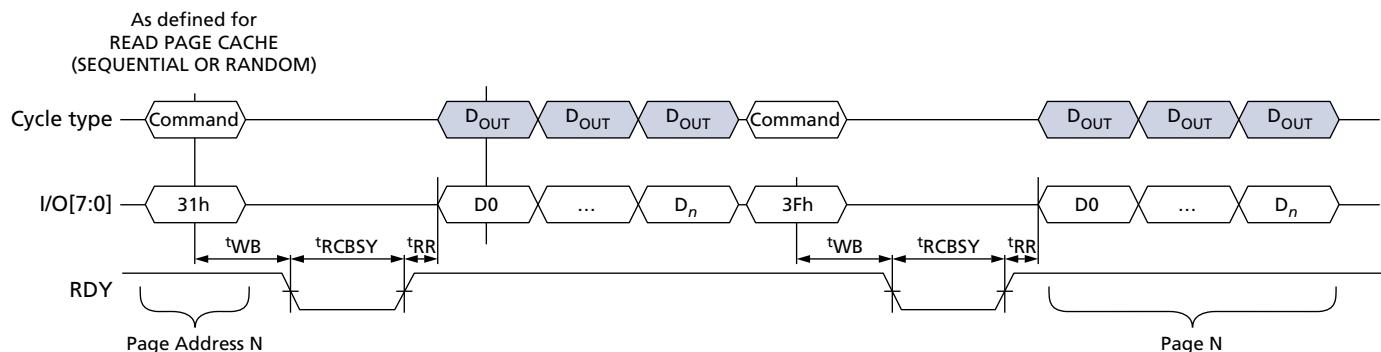
## READ PAGE CACHE LAST (3Fh)

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 43: 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.

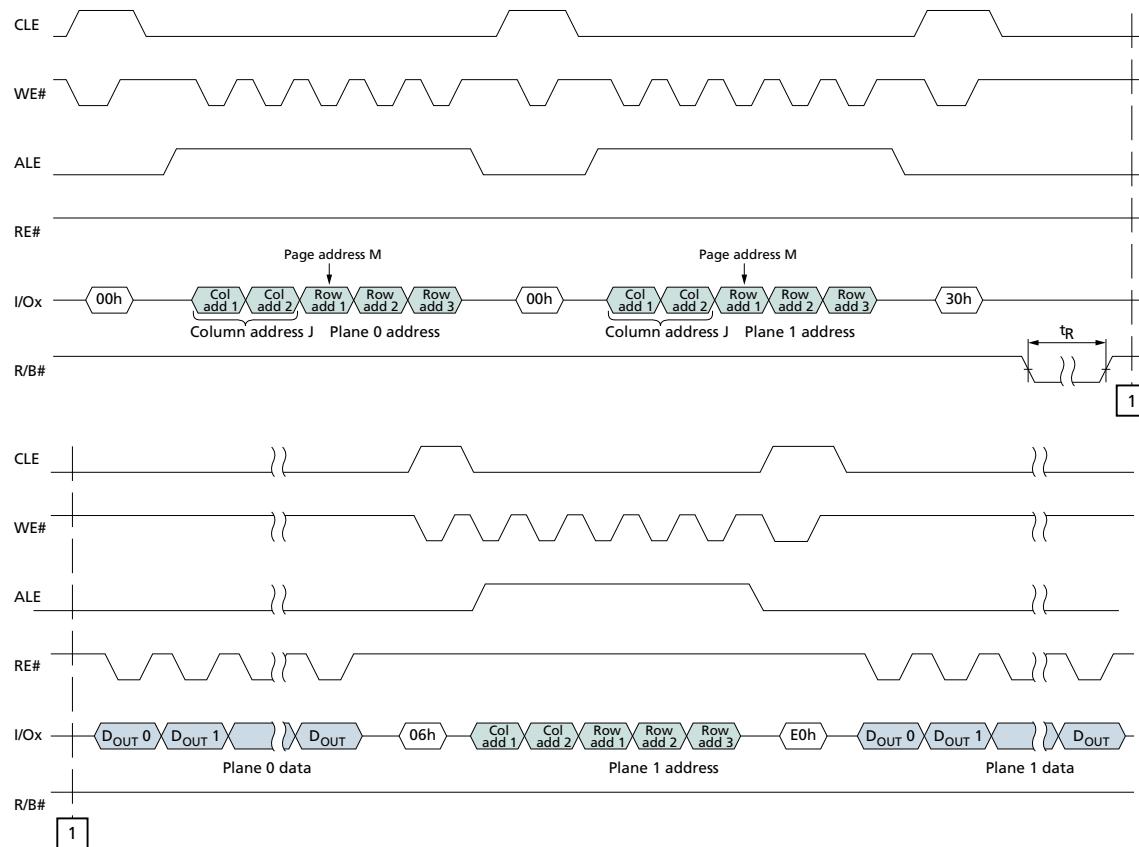
To enter the READ PAGE TWO-PLANE mode, write the 00h command to the command register, and then write five address cycles for plane 0 (BA6 = 0). Next, write the 00h command to the command register, and five address cycles for plane 1 (BA6 = 1). Finally, issue the 30h command. The first-plane and second-plane 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.

**Figure 44: 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, <sup>t</sup>CBSY and <sup>t</sup>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  $t_{PROG}$  as data is transferred.

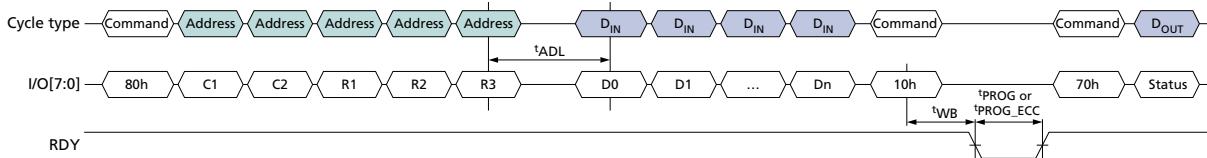
To determine the progress of the data transfer, the host can monitor the target's R/B# signal or, alternatively, the status operations (70h, 78h) may be used. When the die (LUN) is ready (RDY = 1, ARDY = 1), the host should check the status of the FAIL bit.

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  $t_{PROG\_ECC}$ . During  $t_{PROG\_ECC}$ , the internal ECC generates parity bits when error detection is complete.

**Figure 45: 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-



## 168-Ball NAND Flash and LPDDR PoP (TI OMAP) MCP 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).

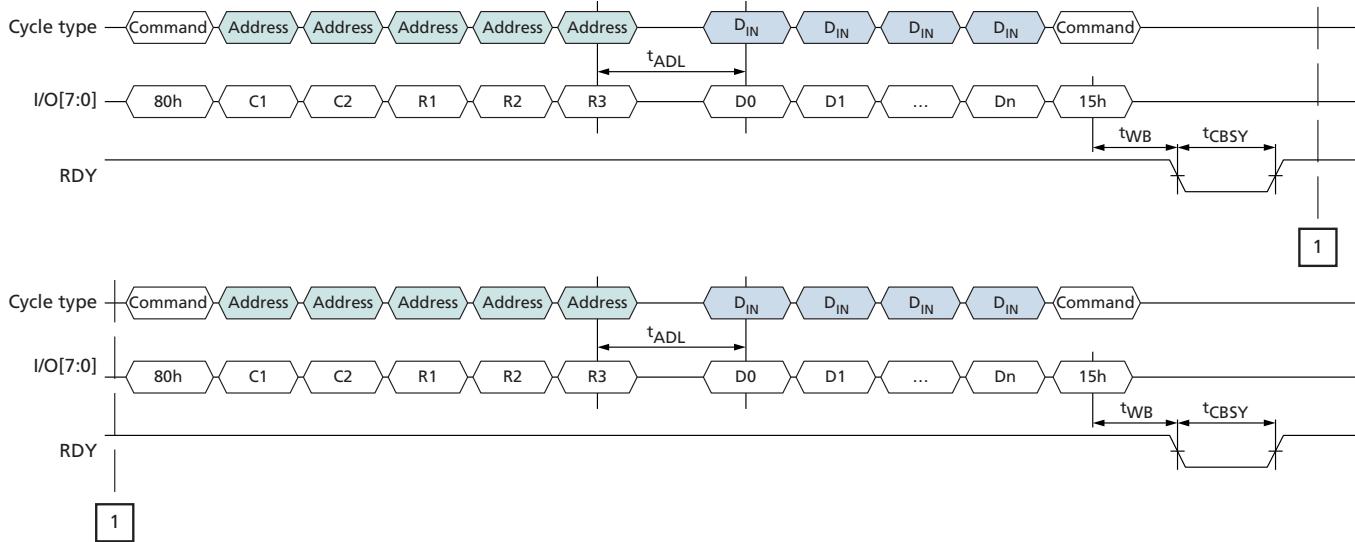
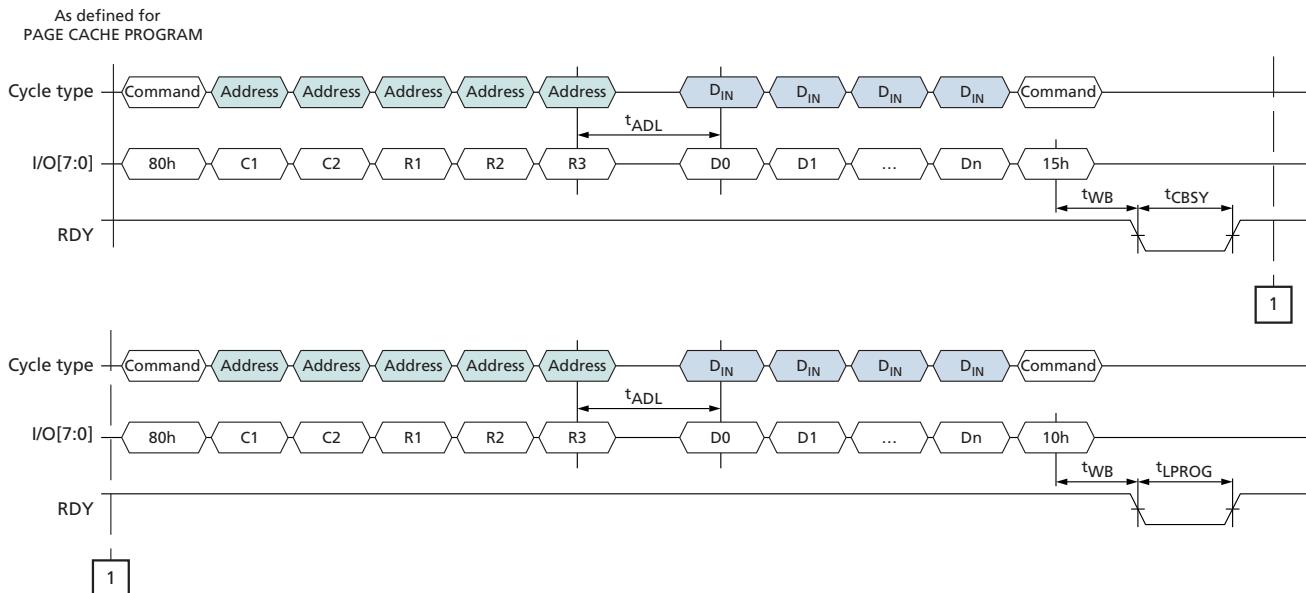
To input a page to the cache register to move it to the NAND array at the block and page address specified, write 80h to the command register. Unless this command has been preceded by a PROGRAM PAGE TWO-PLANE (80h-11h) command, issuing the 80h to the command register clears all of the cache registers' 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  $t_{CBSY}$  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  $t_{CBSY}$ , 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  $t_{CBSY}$ , 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).

**Figure 46: PROGRAM PAGE CACHE (80h-15h) Operation (Start)****Figure 47: 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).

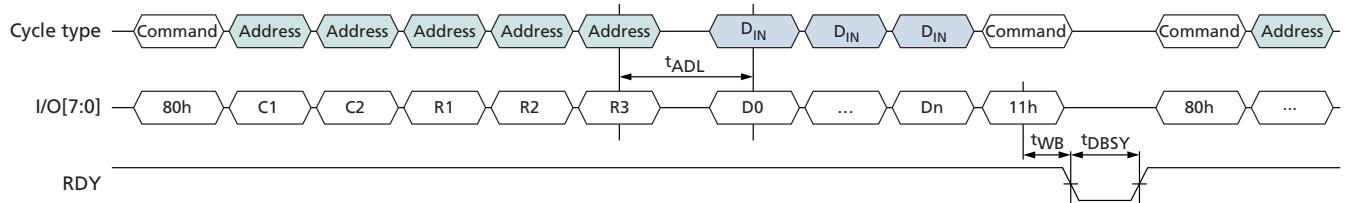
To input a page to the cache register and queue it to be moved to the NAND Flash array at the block and page address specified, write 80h to the command register. Unless this command has been preceded by a PROGRAM PAGE TWO-PLANE (80h-11h) command, issuing the 80h to the command register clears all of 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  $t_{DBSY}$ .

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), 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  $t_{PROG}$ . 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  $t_{CBSY}$ . After  $t_{CBSY}$ , 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.

**Figure 48: 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).

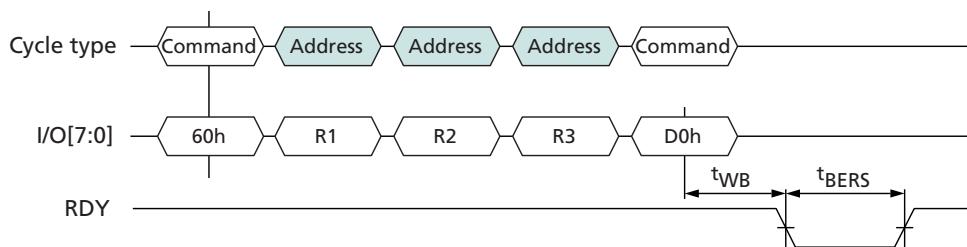
To erase a block, write 60h to the command register. Then write three address cycles containing the row address; the page address is ignored. Conclude by writing D0h to the command register. The selected die (LUN) will go busy (RDY = 0, ARDY = 0) for  $t_{BERS}$  while the block is erased.

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 49: ERASE BLOCK (60h-D0h) Operation**



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

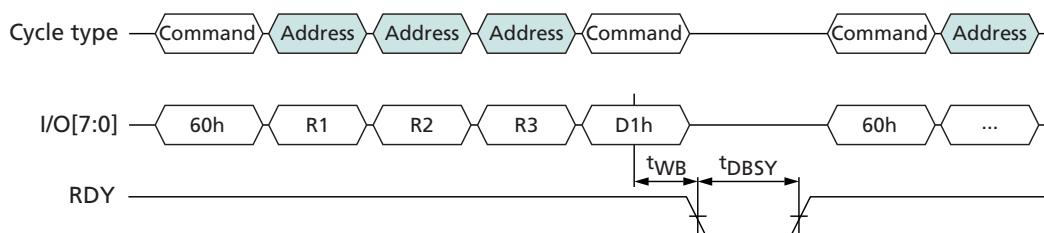
The ERASE BLOCK TWO-PLANE (60h-D1h) command queues a block in the specified plane to be erased in the NAND Flash array. This command can be issued one or more times. Each time a new plane address is specified, that plane is also queued for a block to be erased. To specify the final block to be erased 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 'DBSY.

To determine the progress of '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 50: 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.

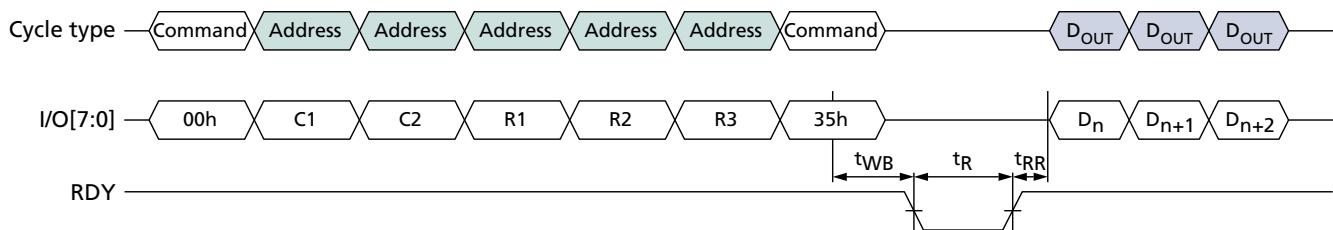
## READ FOR INTERNAL DATA MOVE (00h-35h)

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.

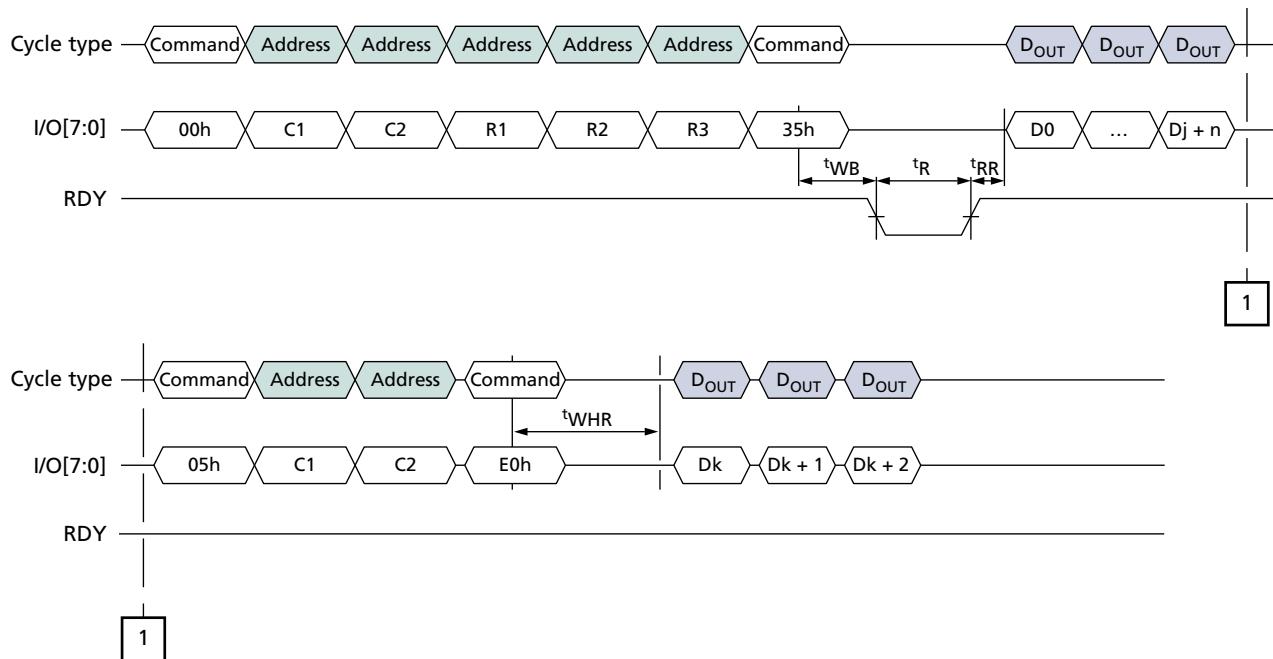
Though it is not required, it is recommended that the host read the data out of the device to verify the data prior to issuing the PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command to prevent the propagation of data errors.

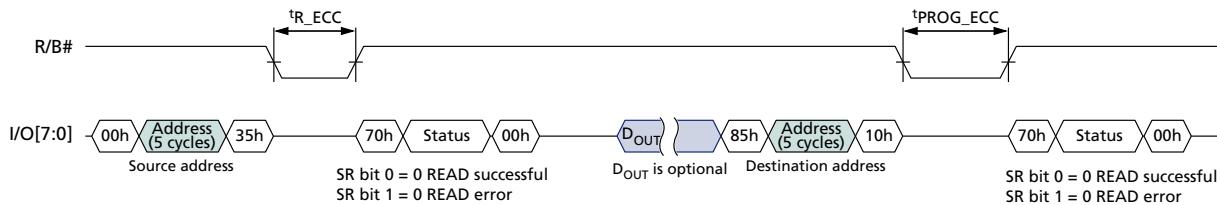
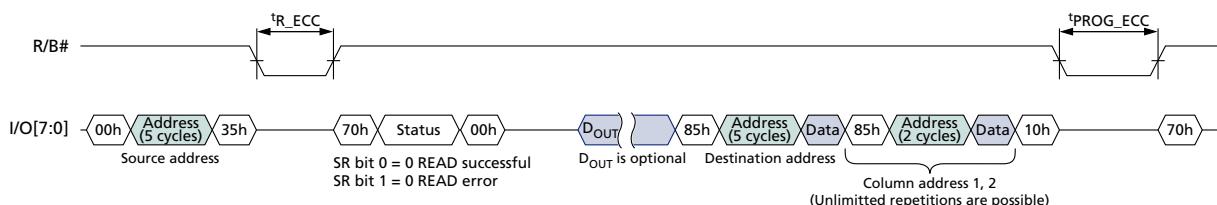
If internal ECC is enabled, the data does not need to be toggled out by the host to be corrected and moving data can then be written to a new page without data reloading, which improves system performance.

**Figure 51: READ FOR INTERNAL DATA MOVE (00h-35h) Operation**



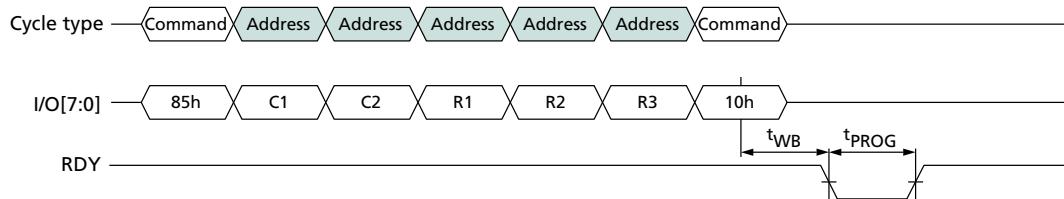
**Figure 52: READ FOR INTERNAL DATA MOVE (00h-35h) with RANDOM DATA READ (05h-E0h)**

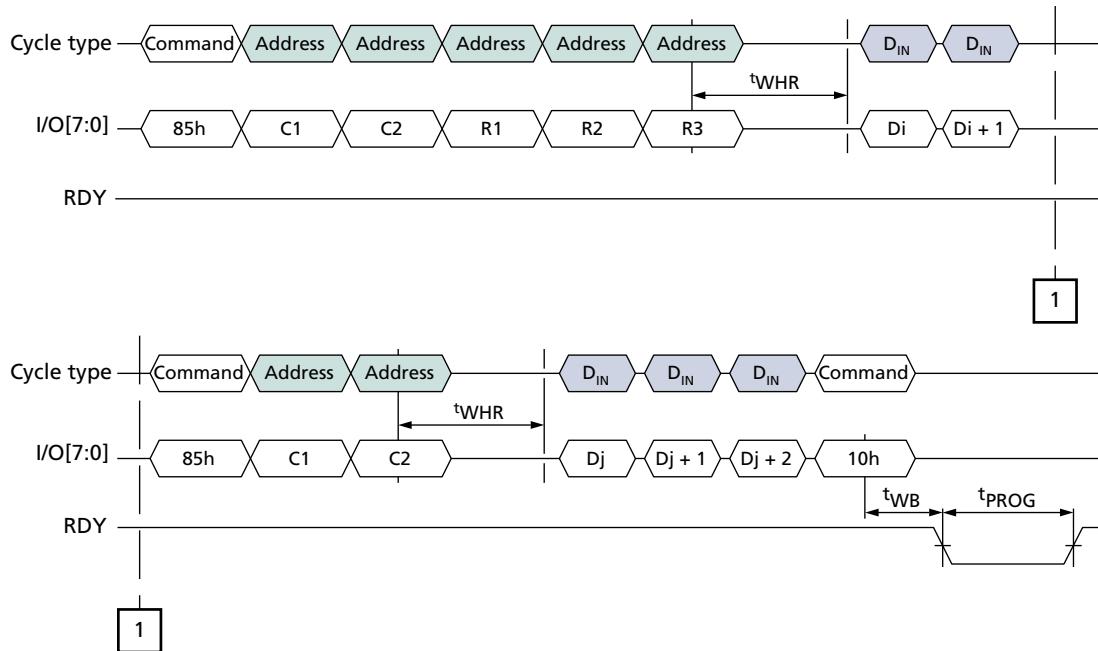


**Figure 53: INTERNAL DATA MOVE (85h-10h) with Internal ECC Enabled**

**Figure 54: INTERNAL DATA MOVE (85h-10h) with RANDOM DATA INPUT with Internal ECC Enabled**


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

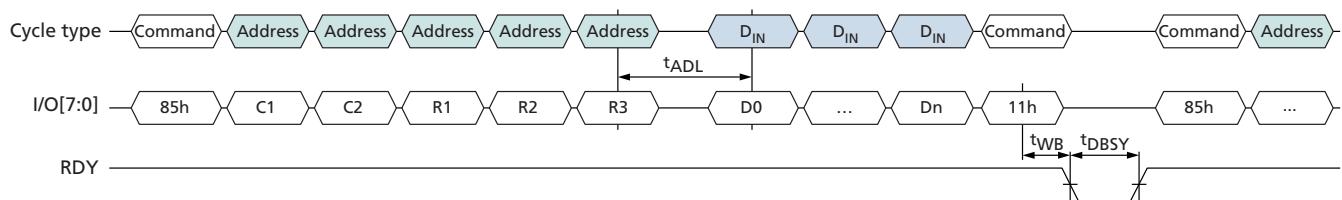
The PROGRAM FOR INTERNAL DATA MOVE (85h-10h) command is functionally identical to the PROGRAM PAGE (80h-10h) command, except that when 85h is written to the command register, cache register contents are not cleared.

**Figure 55: PROGRAM FOR INTERNAL DATA MOVE (85h-10h) Operation**


**Figure 56: 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 57: 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.

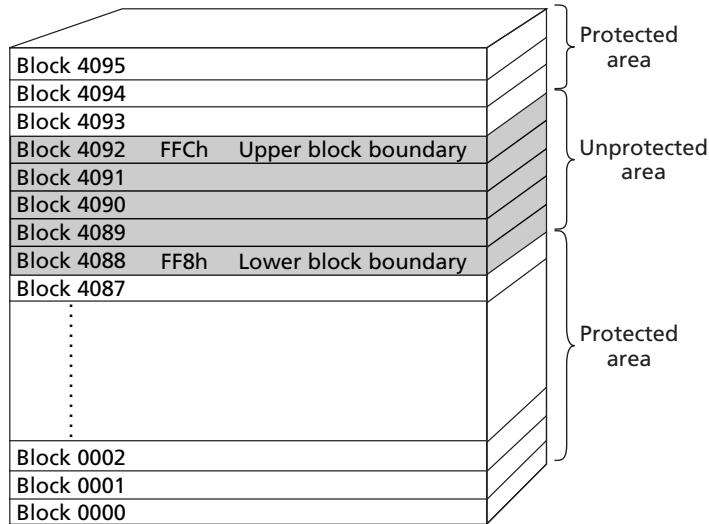
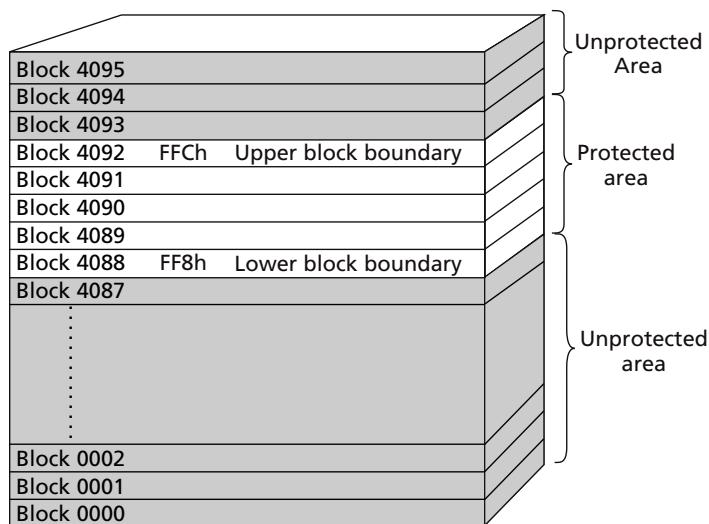
## UNLOCK (23h-24h)

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.

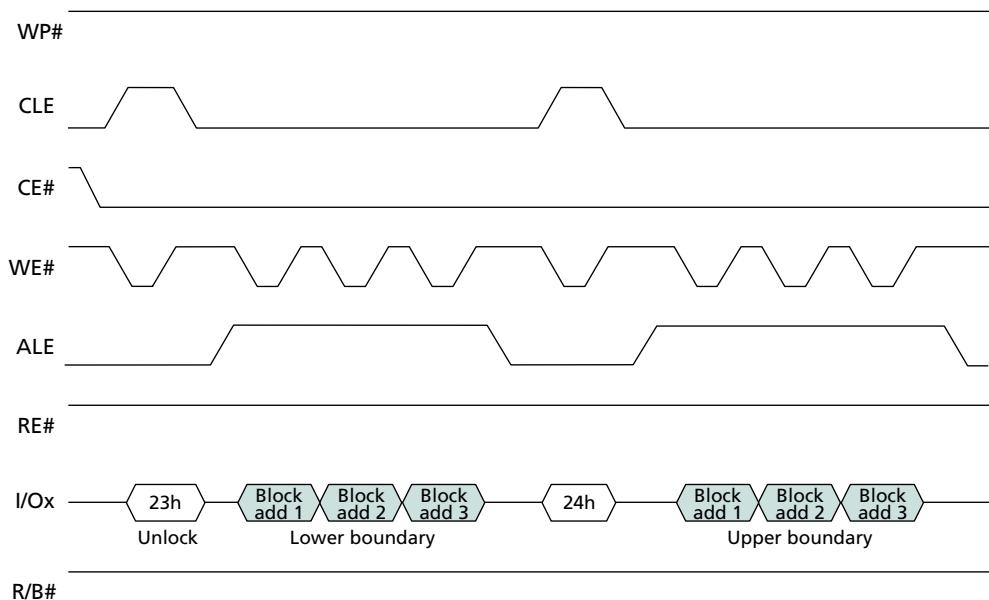
**Figure 58: Flash Array Protected: Invert Area Bit = 0****Figure 59: Flash Array Protected: Invert Area Bit = 1**

**Table 22: 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 60: UNLOCK Operation**

## LOCK (2Ah)

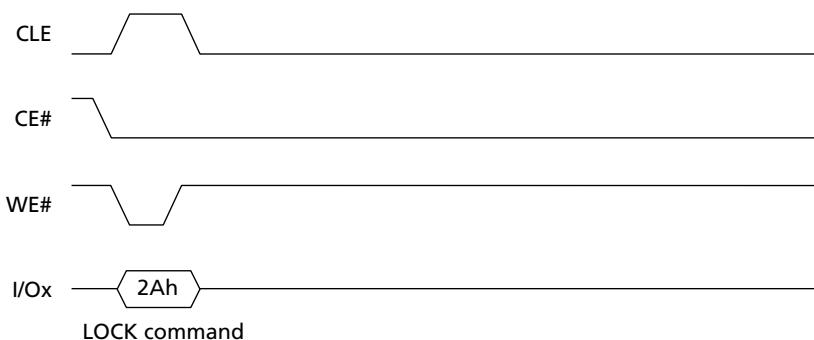
By default at power-on, if LOCK is HIGH, all the blocks are locked and protected from PROGRAM and ERASE operations. If portions of the device are unlocked using the UNLOCK (23h) command, they can be locked again using the LOCK (2Ah) command. The LOCK command locks all of the blocks in the device. Locked 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 61: 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.

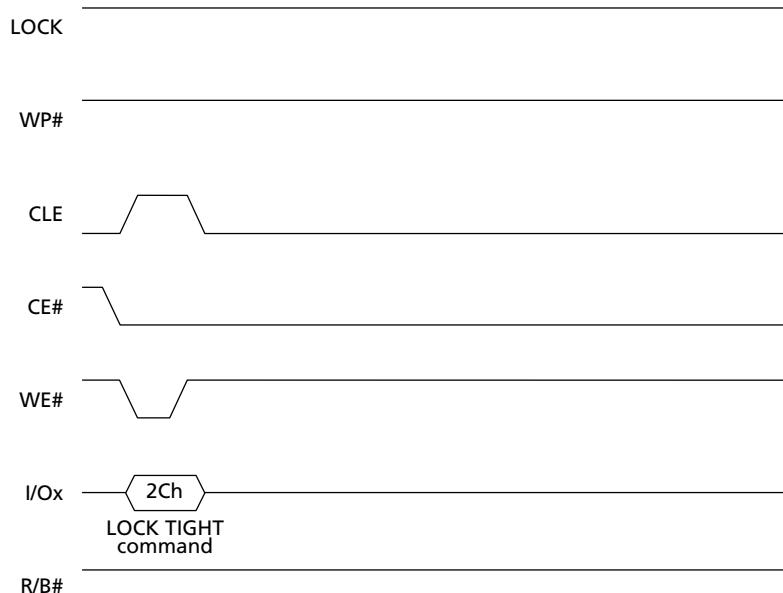
To implement LOCK TIGHT in all of the locked blocks in the device, verify that WP# is HIGH and then issue the LOCK TIGHT (2Ch) command.

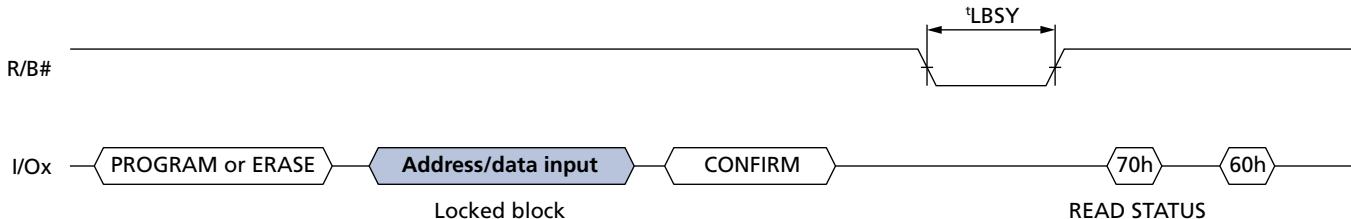
When a PROGRAM or ERASE operation is issued to a locked block that has also been locked tight, R/B# goes LOW for  $t_{LSY}$ . The PROGRAM or ERASE operation does not complete. The READ STATUS (70h) command reports bit 7 as 0, indicating that the block is protected. PROGRAM and ERASE operations complete 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 62: LOCK TIGHT Operation**



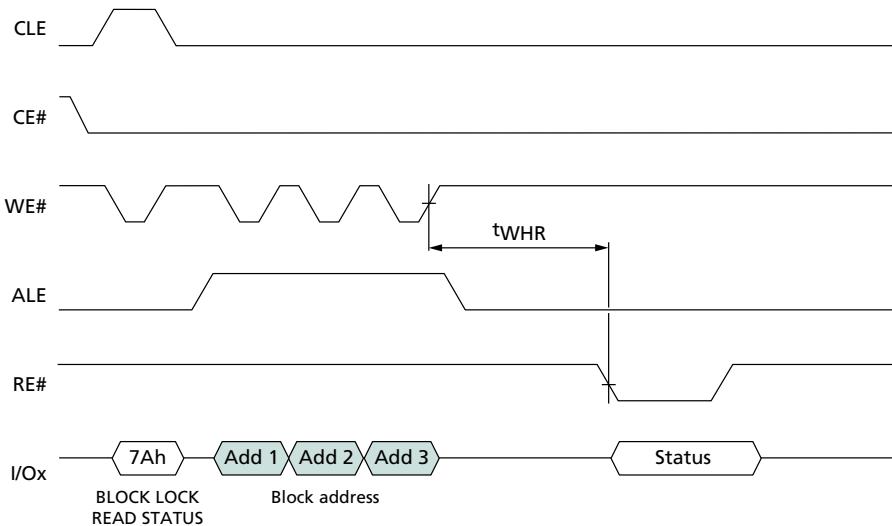
**Figure 63: PROGRAM/ERASE Issued to Locked Block**

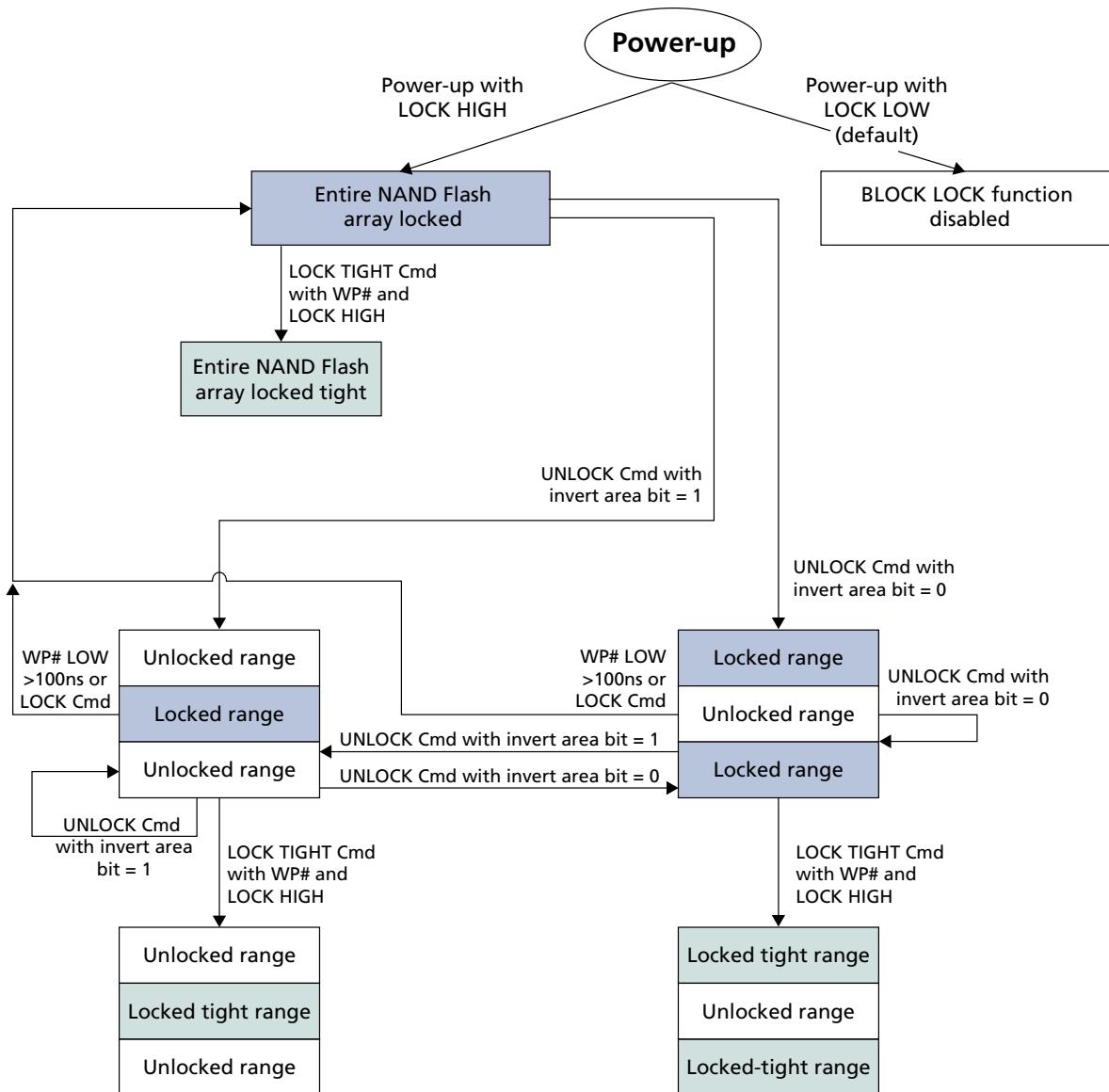
### BLOCK LOCK READ STATUS (7Ah)

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 23: 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 64: BLOCK LOCK READ STATUS**

**Figure 65: 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.

**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.

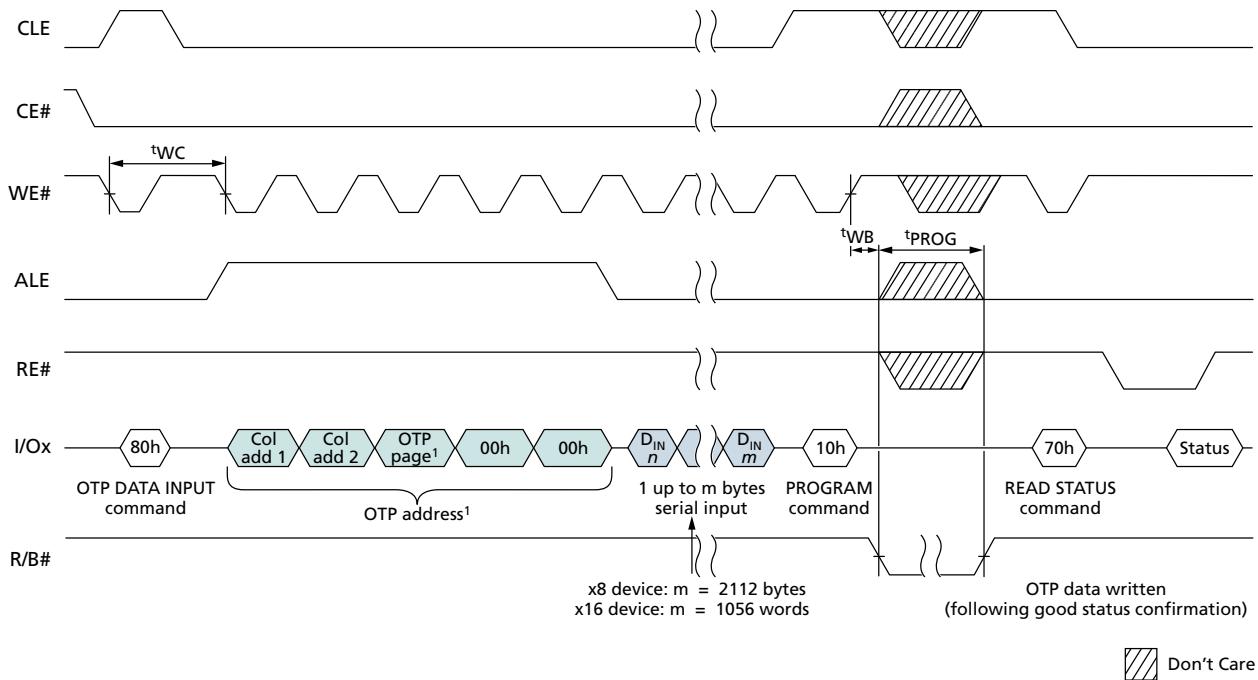
To use the PROGRAM PAGE command, issue the 80h command. Issue  $n$  address cycles. The first two address cycles are the column address. For the remaining cycles, select a page in the range of 02h-00h through 1Fh-00h. Next, write from 1-2112 bytes of data. After data input is complete, issue the 10h 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 ( $t_{PROG}$ ). 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.

## RANDOM DATA INPUT (85h)

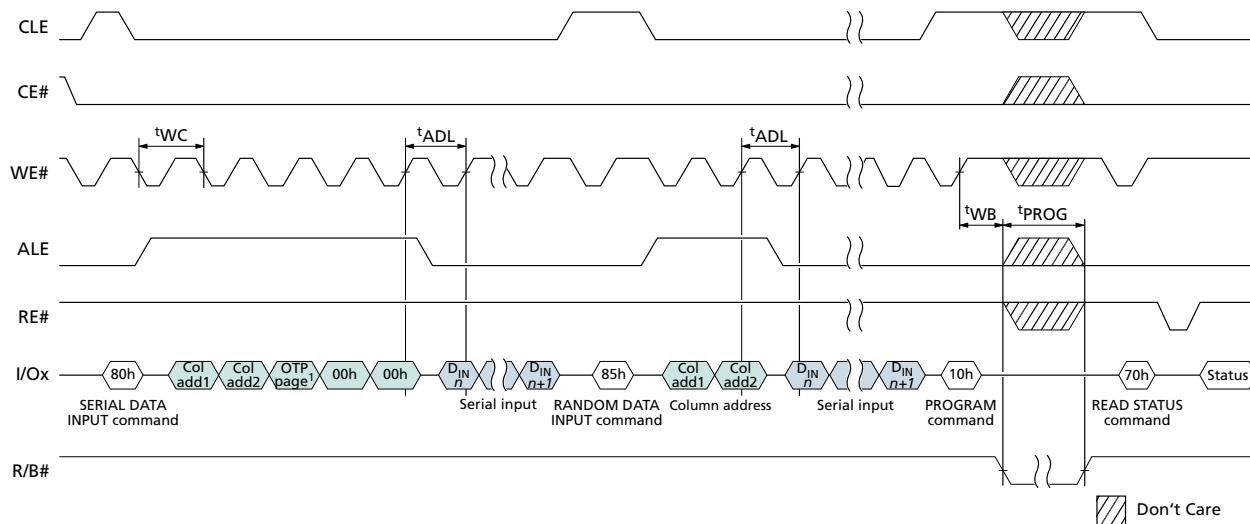
After the initial OTP data set is input, additional data can be written to a new column address with the RANDOM DATA INPUT (85h) command. The RANDOM DATA INPUT command can be used any number of times in the same page prior to the OTP PAGE WRITE (10h) command being issued.

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



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

**Figure 67: OTP DATA PROGRAM Operation with RANDOM DATA INPUT (After Entering OTP Operation Mode)**



## OTP DATA PROTECT (80h-10)

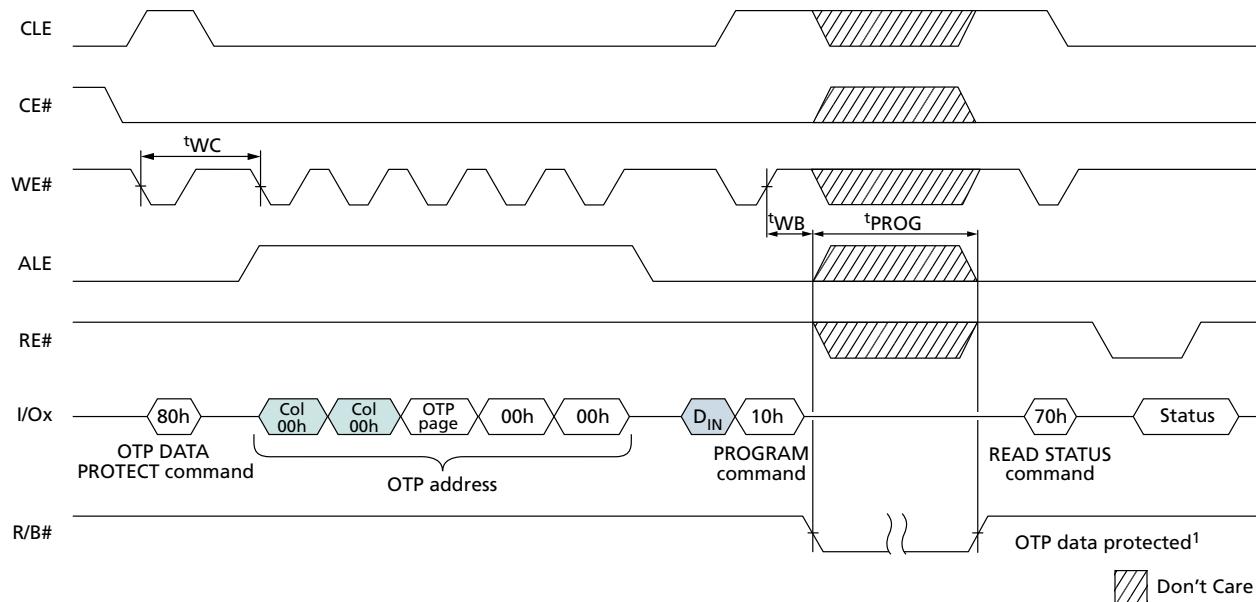
The OTP area is protected on a block basis. To protect a block, set the device to OTP protect mode, then issue the PROGRAM PAGE (80h-10h) command and write OTP address 00h, 00h, 00h, 00h. To set the device to OTP protect mode, issue the SET FEATURE (EFh) command to 90h (feature address) and write 03h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

**Figure 68: OTP DATA PROTECT Operation (After Entering OTP Protect Mode)**


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

## 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.

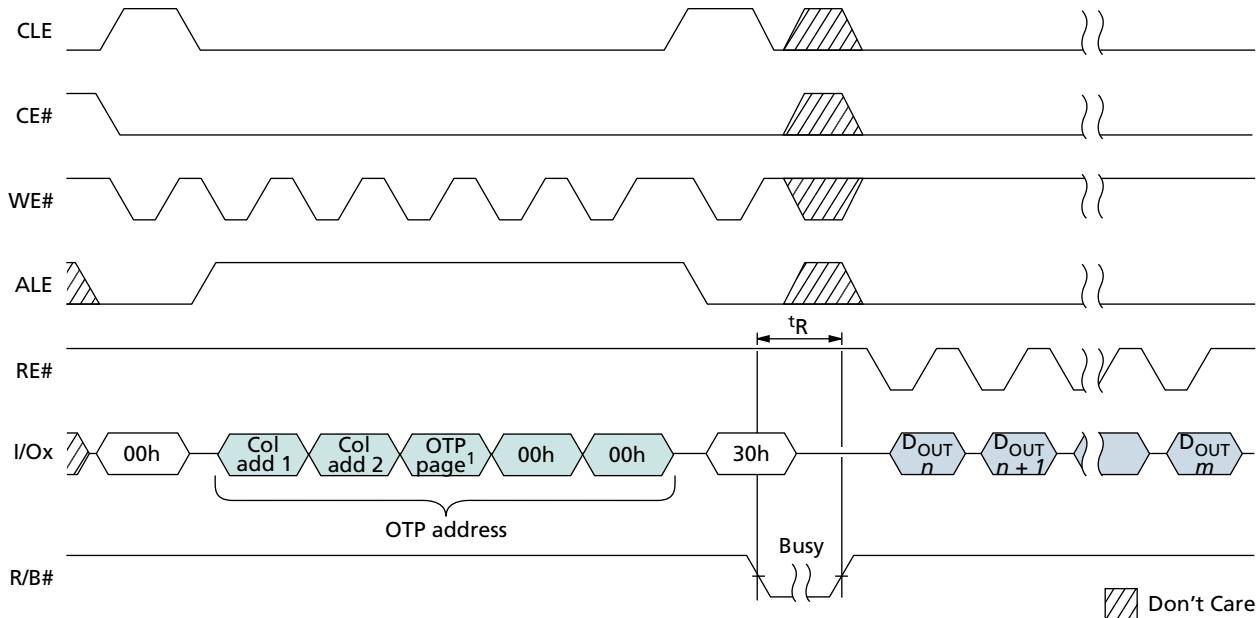
R/B# goes LOW ( $t_R$ ) while the data is moved from the OTP page to the data register. The READ STATUS (70h) command is the only valid command for reading status in OTP operation mode. Bit 5 of the status register reflects the state of R/B# (see Status Operations).

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.

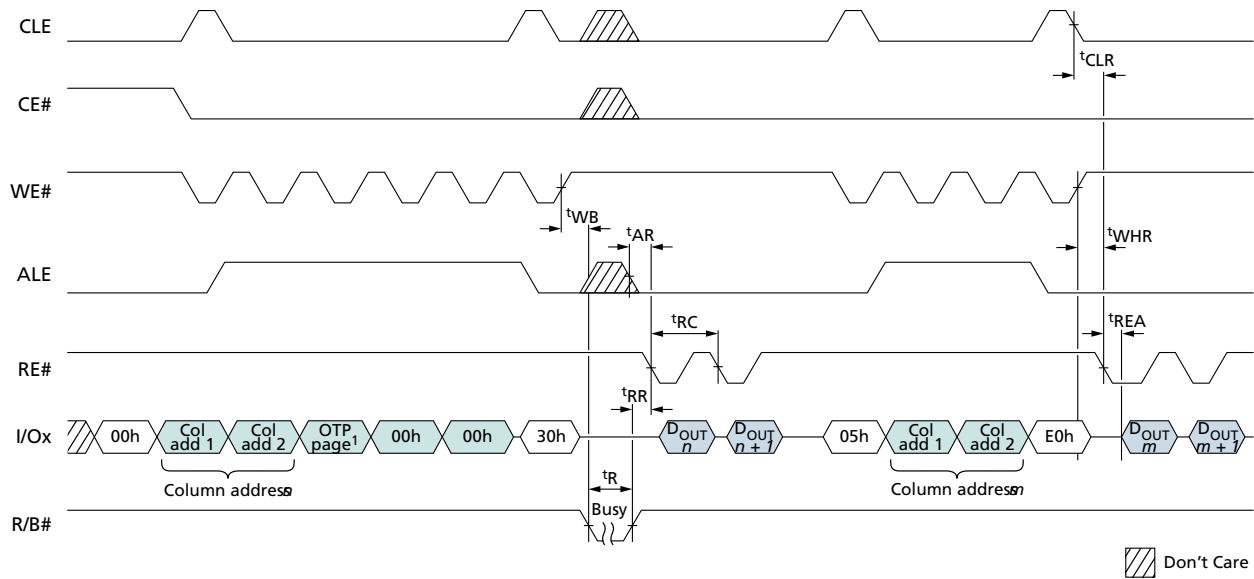
The PAGE READ command is compatible with the RANDOM DATA OUTPUT (05h-E0h) command.

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

**Figure 69: OTP DATA READ**



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

**Figure 70: 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.

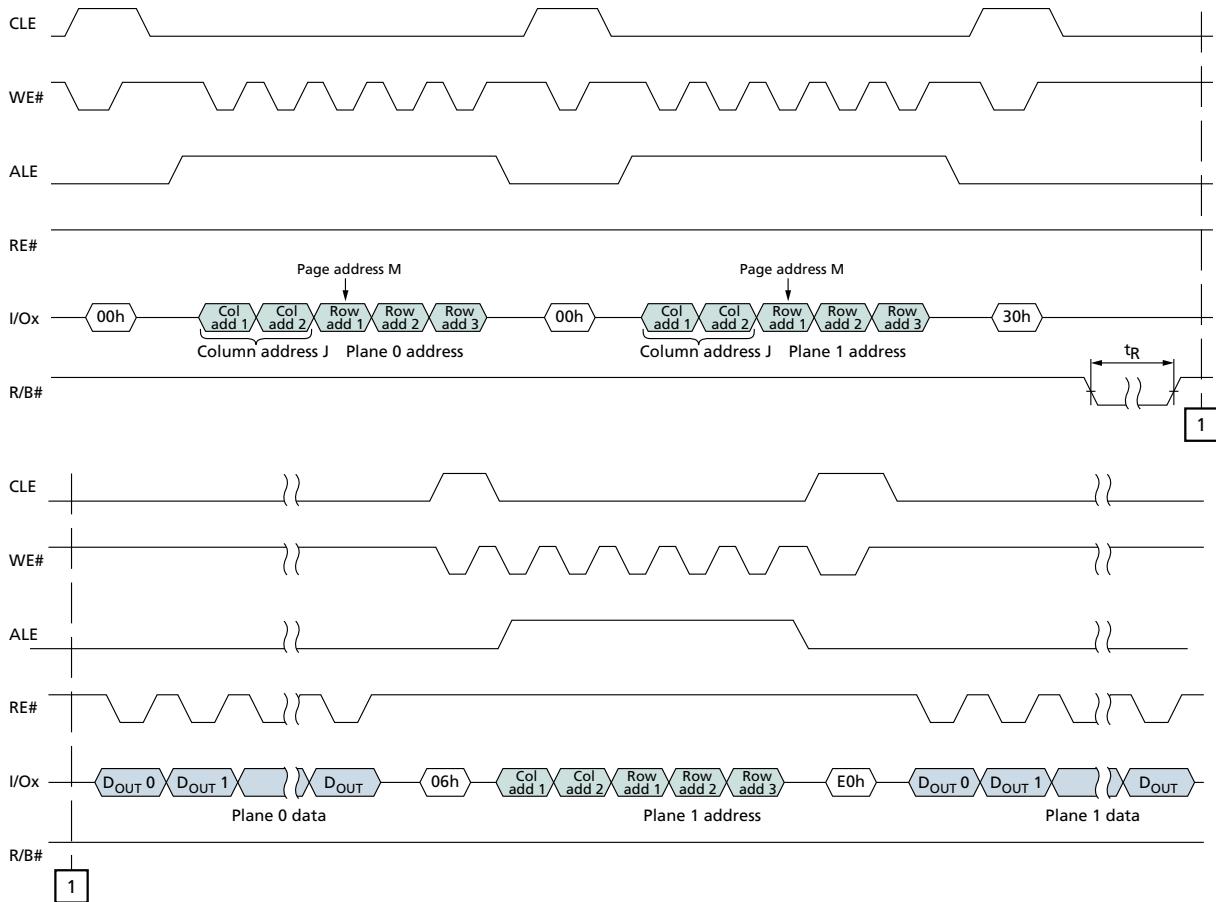
When issuing two-plane program or erase operations, use the READ STATUS (70h) command and check whether the previous operation(s) failed. If the READ STATUS (70h) command indicates that an error occurred (FAIL = 1 and/or FAILC = 1), use the READ STATUS ENHANCED (78h) command to determine which plane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

## Two-Plane Addressing

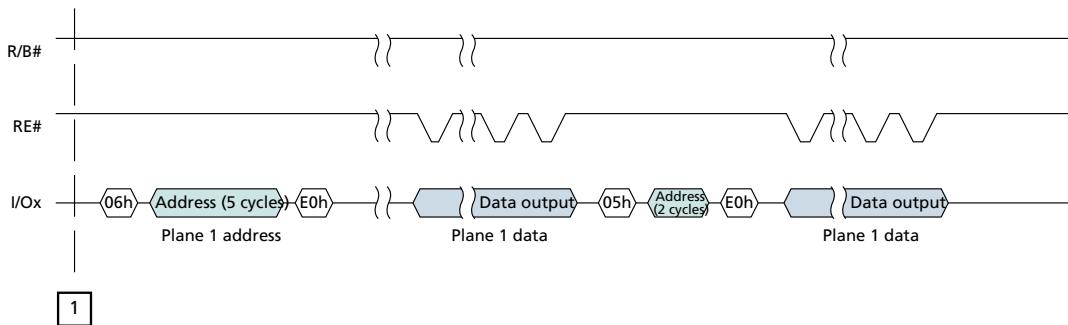
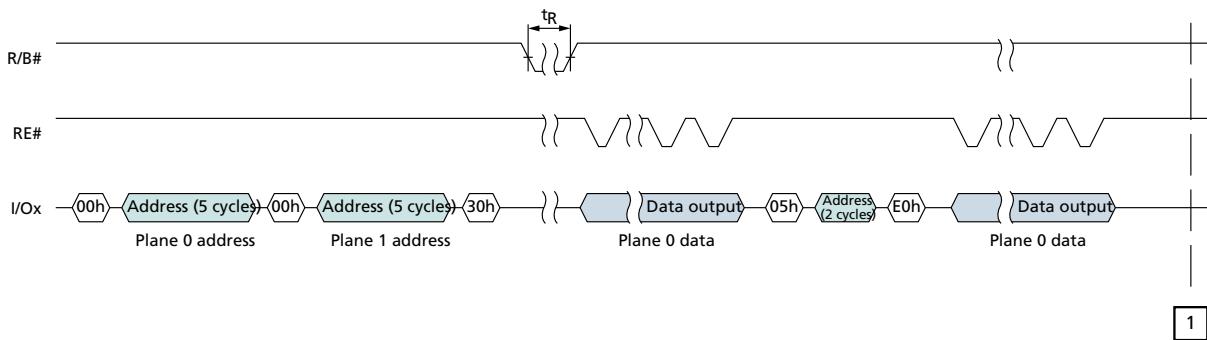
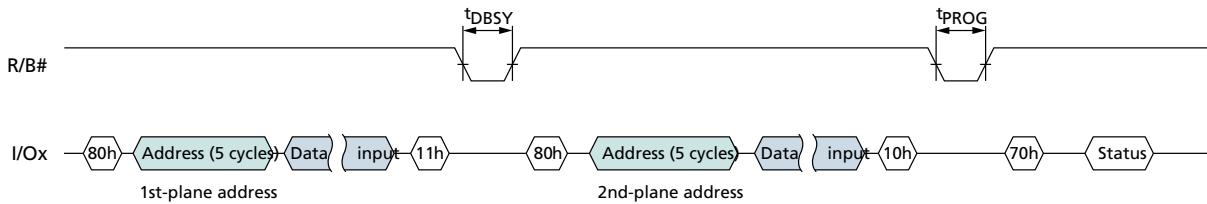
Two-plane commands require multiple, five-cycle addresses, one address per operational plane. For a given two-plane operation, these addresses are subject to the following requirements:

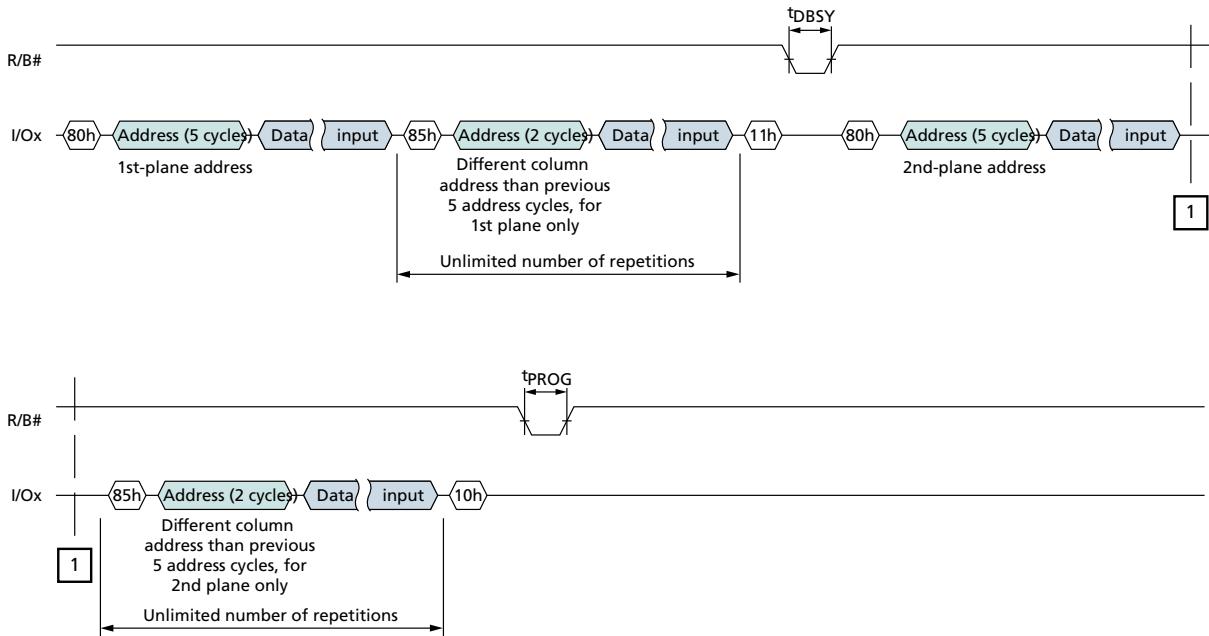
- The LUN address bit(s) must be identical for all of the issued addresses.
- The plane select bit, BA[6], must be different for each issued address.
- The page address bits, PA[5:0], must be identical for each issued address.

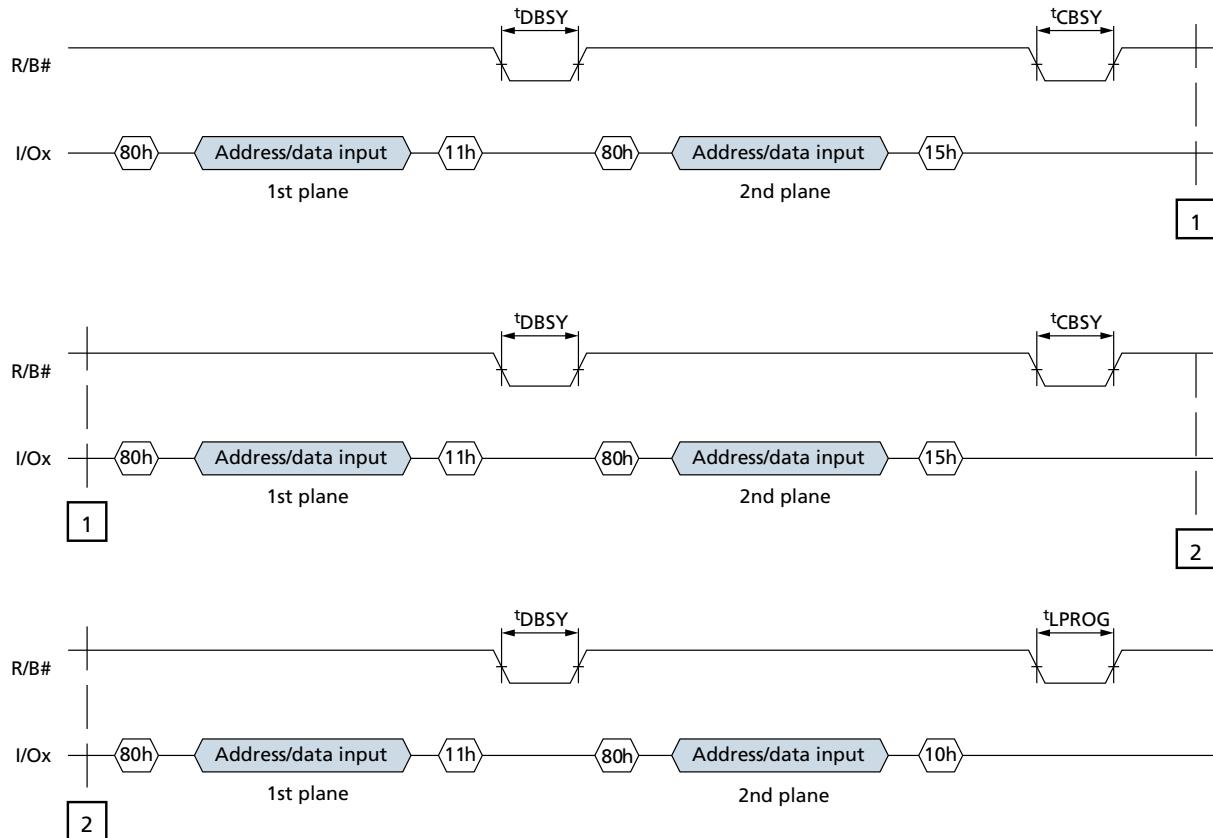
The READ STATUS (70h) command should be used following two-plane program page and erase block operations on a single die (LUN).

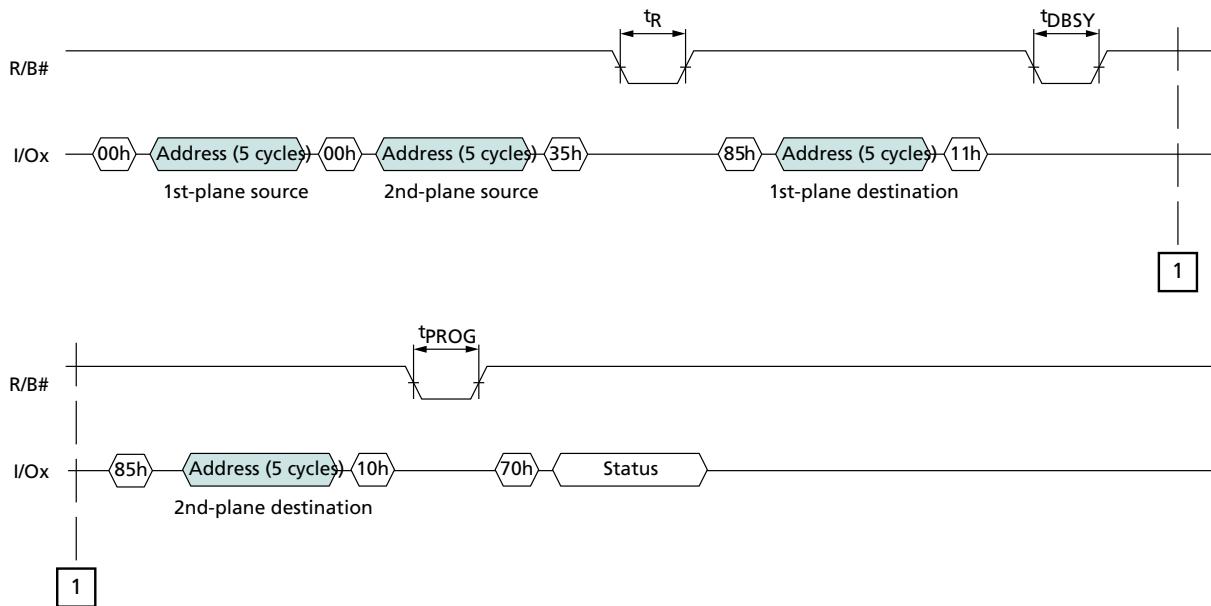
**Figure 71: TWO-PLANE PAGE READ**

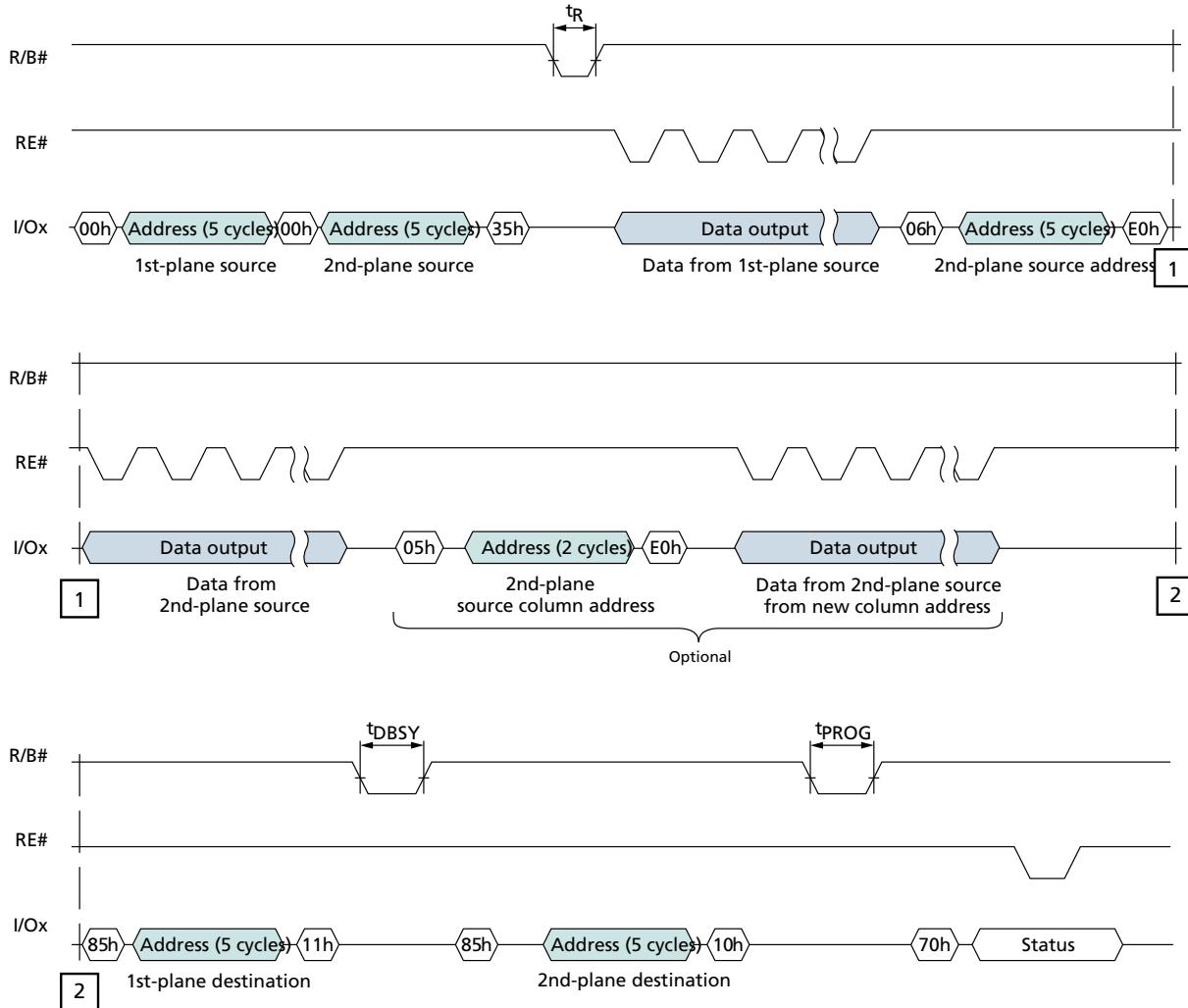
- Notes:
1. Column and page addresses must be the same.
  2. The least significant block address bit, BA6, must be different for the first- and second-plane addresses.

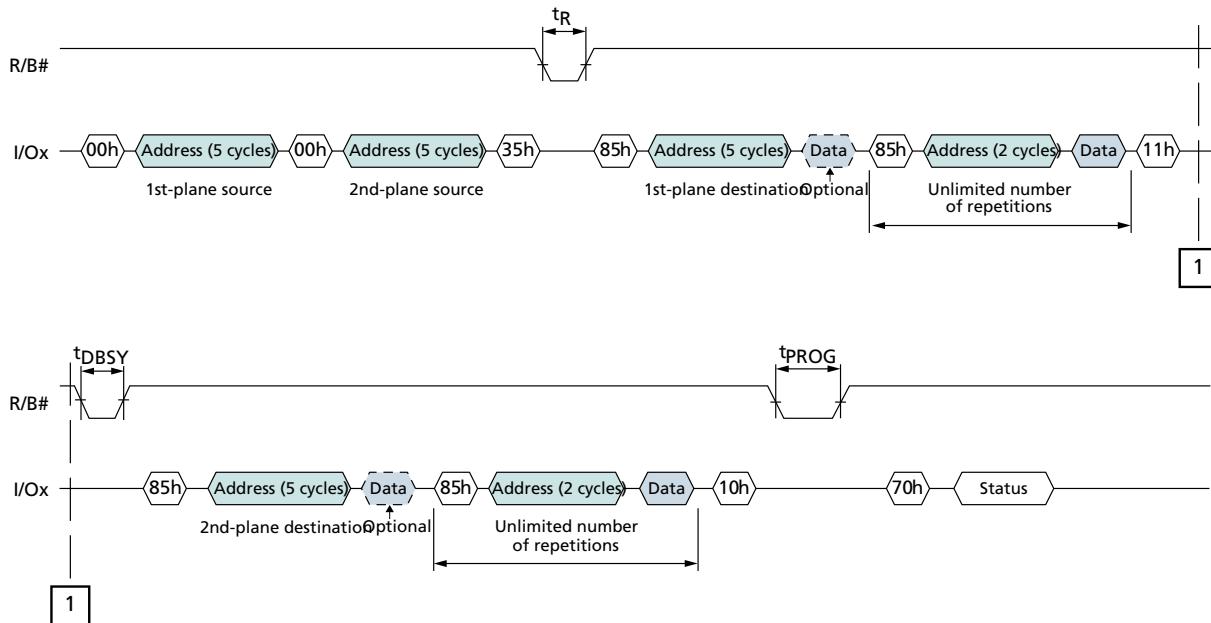
**Figure 72: TWO-PLANE PAGE READ with RANDOM DATA READ****Figure 73: TWO-PLANE PROGRAM PAGE**

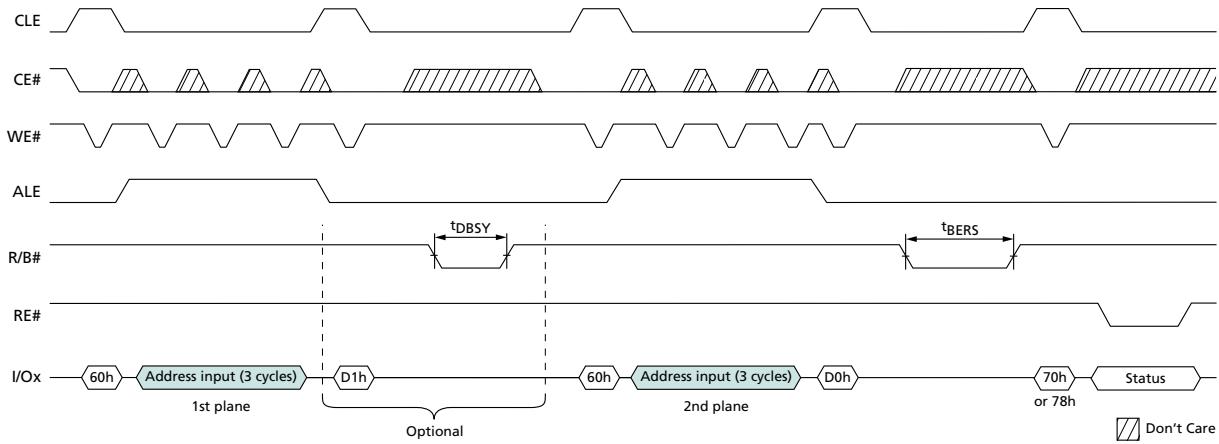
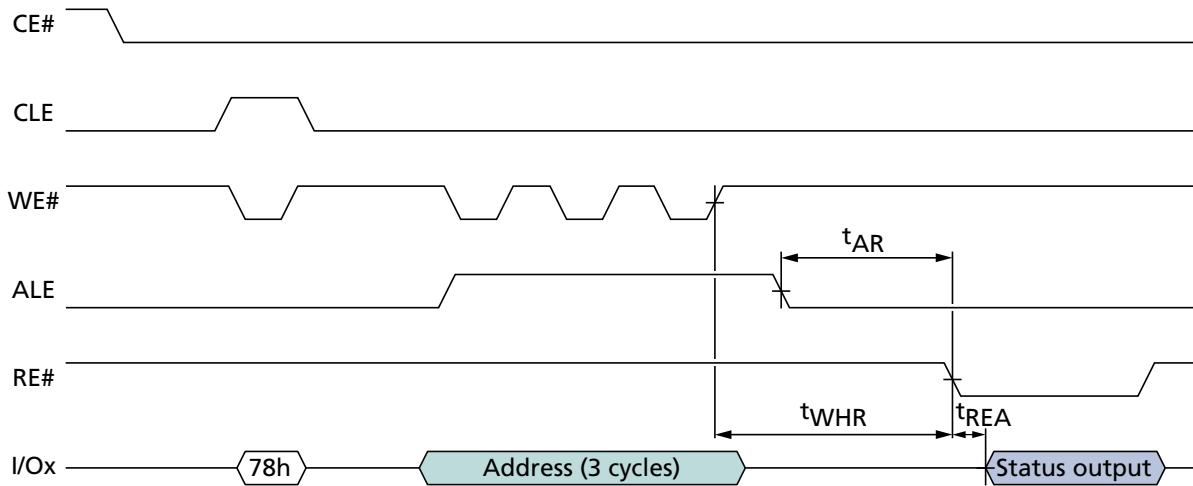
**Figure 74: TWO-PLANE PROGRAM PAGE with RANDOM DATA INPUT**

**Figure 75: TWO-PLANE PROGRAM PAGE CACHE MODE**

**Figure 76: TWO-PLANE INTERNAL DATA MOVE**

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


**Figure 78: TWO-PLANE INTERNAL DATA MOVE with RANDOM DATA INPUT**

**Figure 79: TWO-PLANE BLOCK ERASE****Figure 80: 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 24: 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 x16: word 1024
Bad-block mark	x8: 00h x16: 0000h
Minimum required ECC	4-bit ECC per 528 bytes

**Table 24: 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



## 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 81: 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

**Figure 82: 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 25: Absolute Maximum Ratings**

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

<b>Parameter/Condition</b>		<b>Symbol</b>	<b>Min</b>	<b>Max</b>	<b>Unit</b>
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 26: Recommended Operating Conditions**

<b>Parameter/Condition</b>		<b>Symbol</b>	<b>Min</b>	<b>Typ</b>	<b>Max</b>	<b>Unit</b>
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 27: Valid Blocks**

<b>Parameter</b>	<b>Symbol</b>	<b>Device</b>	<b>Min</b>	<b>Max</b>	<b>Unit</b>	<b>Notes</b>
Valid block number	NVB	MT29F4G	4016	4096	Blocks	1, 2
		MT29F8G	8032	8192	Blocks	1, 2, 3

- Notes:
1. 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.
  2. Block 00h (the first block) is guaranteed to be valid with ECC when shipped from the factory.
  3. Each 4Gb section has a maximum of 80 invalid blocks.

**Table 28: 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:
- These parameters are verified in device characterization and are not 100% tested.
  - Test conditions:  $T_C = 25^\circ\text{C}$ ;  $f = 1 \text{ MHz}$ ;  $V_{IN} = 0\text{V}$ .
  - Capacitance ( $C_{IN} = C_{IO} = 20\text{pF}$ ) for MT29F8G.

**Table 29: 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:
- Verified in device characterization, not 100% tested.



## Electrical Specifications – DC Characteristics and Operating Conditions

**Table 30: 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 Line capacitance = $0.1\mu\text{F}$	$I_{ST}$	–	–	10 per die	mA	1
Input leakage current	$V_{IN} = 0V$ to $V_{CC}$	$I_{LI}$	–	–	$\pm 10$	$\mu\text{A}$	
Output leakage current	$V_{OUT} = 0V$ to $V_{CC}$	$I_{LO}$	–	–	$\pm 10$	$\mu\text{A}$	
Input high voltage	$I/O[7:0]$ , $I/O[15:0]$ , $CE\#$ , CLE, ALE, WE#, RE#, $WP\#$	$V_{IH}$	$0.8 \times V_{CC}$	–	$V_{CC} + 0.3$	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	$V_{OL} = 0.4V$	$I_{OL}$ (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.  $I_{OL}$  (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.

**Table 31: DC Characteristics and Operating Conditions (1.8V)**

<b>Parameter</b>	<b>Conditions</b>	<b>Symbol</b>	<b>Min</b>	<b>Typ</b>	<b>Max</b>	<b>Unit</b>	<b>Notes</b>
Sequential READ current	$t_{RC} = t_{RC}$ (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# = $V_{CC} - 0.2\text{V}$ ; WP# = 0V/ $V_{CC}$	$I_{SB2}$	–	10	50	$\mu\text{A}$	
Staggered power-up current	Rise time = 1ms Line capacitance = 0.1 $\mu\text{F}$	$I_{ST}$	–	–	10 per die	mA	3
Input leakage current	$V_{IN} = 0\text{V}$ to $V_{CC}$	$I_{LI}$	–	–	$\pm 10$	$\mu\text{A}$	
Output leakage current	$V_{OUT} = 0\text{V}$ to $V_{CC}$	$I_{LO}$	–	–	$\pm 10$	$\mu\text{A}$	
Input high voltage	I/O[7:0], I/O[15:0], CE#, CLE, ALE, WE#, RE#, 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#)	$V_{OL} = 0.2\text{V}$	$I_{OL}$ (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  $V_{CC}$ (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 32: 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 33: 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.

**Table 34: 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	μs	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  $\pm 200\text{mV}$  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 5μs.

**Table 35: 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	

**Table 35: 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  $\pm 200\text{mV}$  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 $\mu\text{s}$ .



## Electrical Specifications – Program/Erase Characteristics

**Table 36: Program/Erase Characteristics**

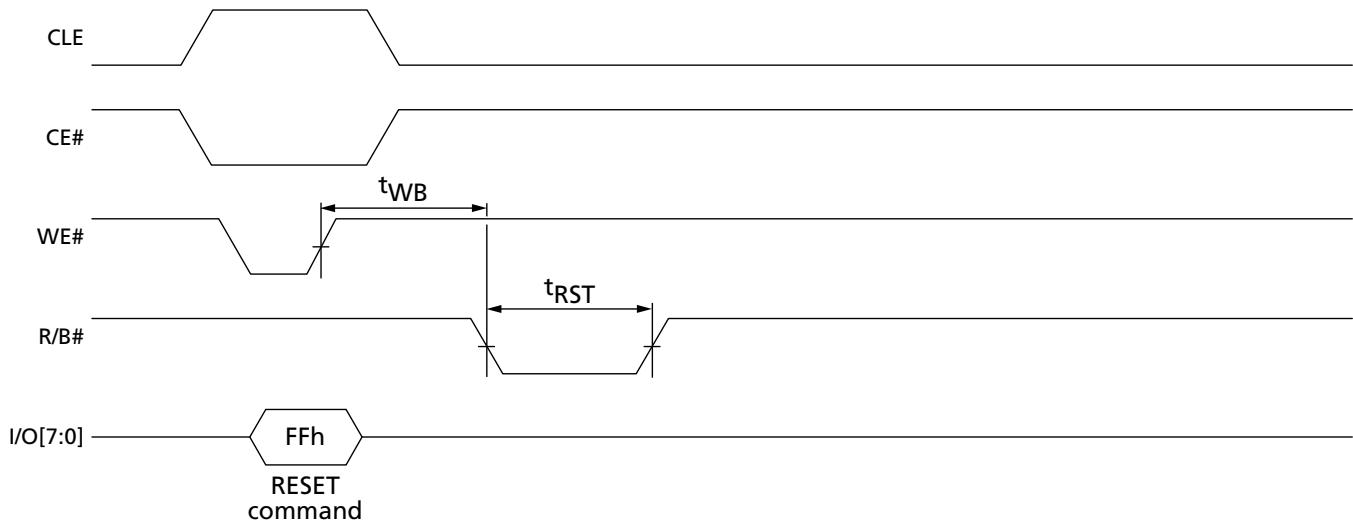
Parameter	Symbol	Typ	Max	Unit	Notes
Number of partial-page programs	NOP	–	4	cycles	1
BLOCK ERASE operation time	$t_{BERS}$	0.7	3	ms	
Busy time for PROGRAM CACHE operation	$t_{CBSY}$	3	600	μs	2
Cache read busy time	$t_{RCBSY}$	3	25	μs	
Busy time for SET FEATURES and GET FEATURES operations	$t_{FEAT}$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

Notes:

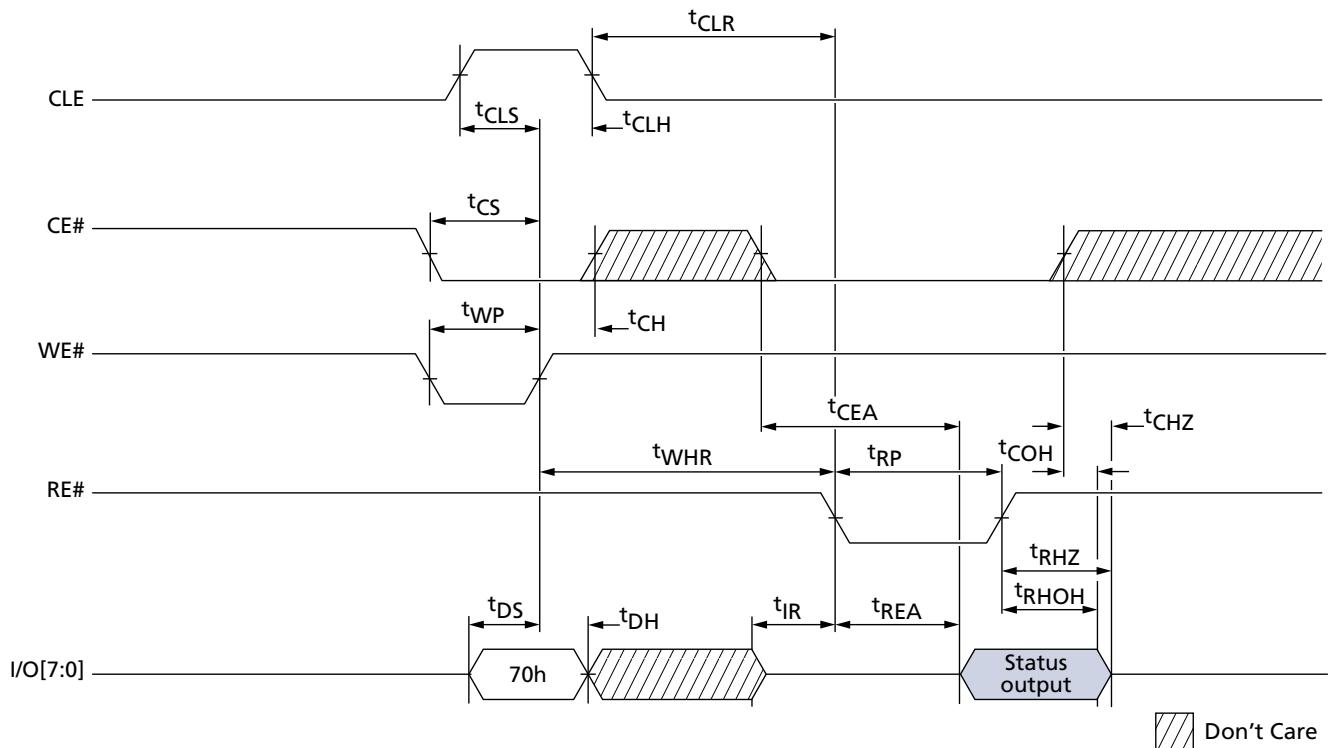
- Four total partial-page programs to the same page. If ECC is enabled, then the device is limited to one partial-page program per ECC user area, not exceeding four partial-page programs per page.
- $t_{CBSY}$  MAX time depends on timing between internal program completion and data-in.
- Parameters are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

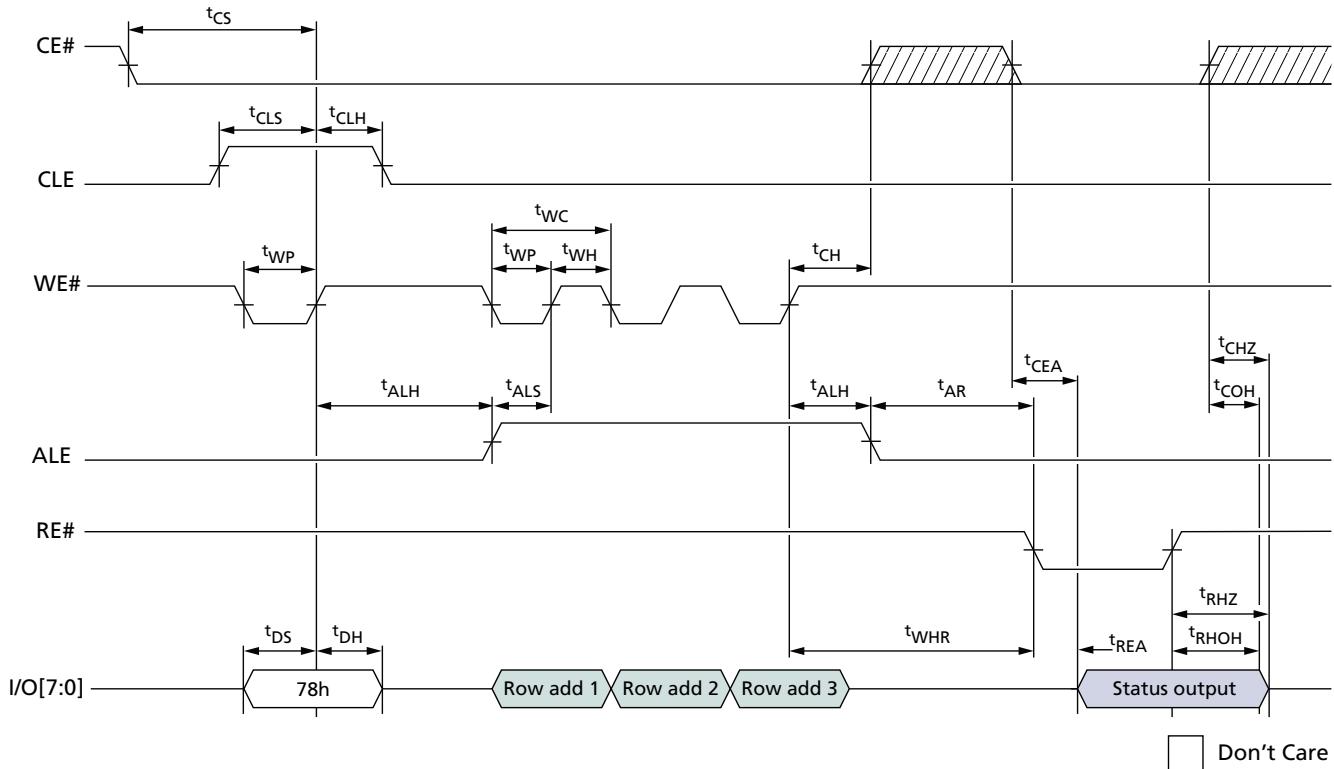
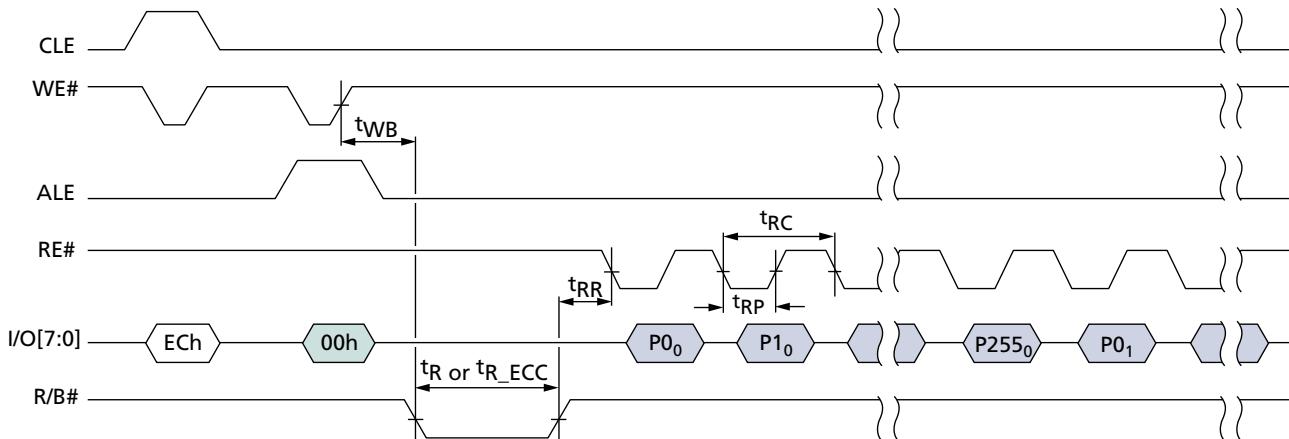
## Asynchronous Interface Timing Diagrams

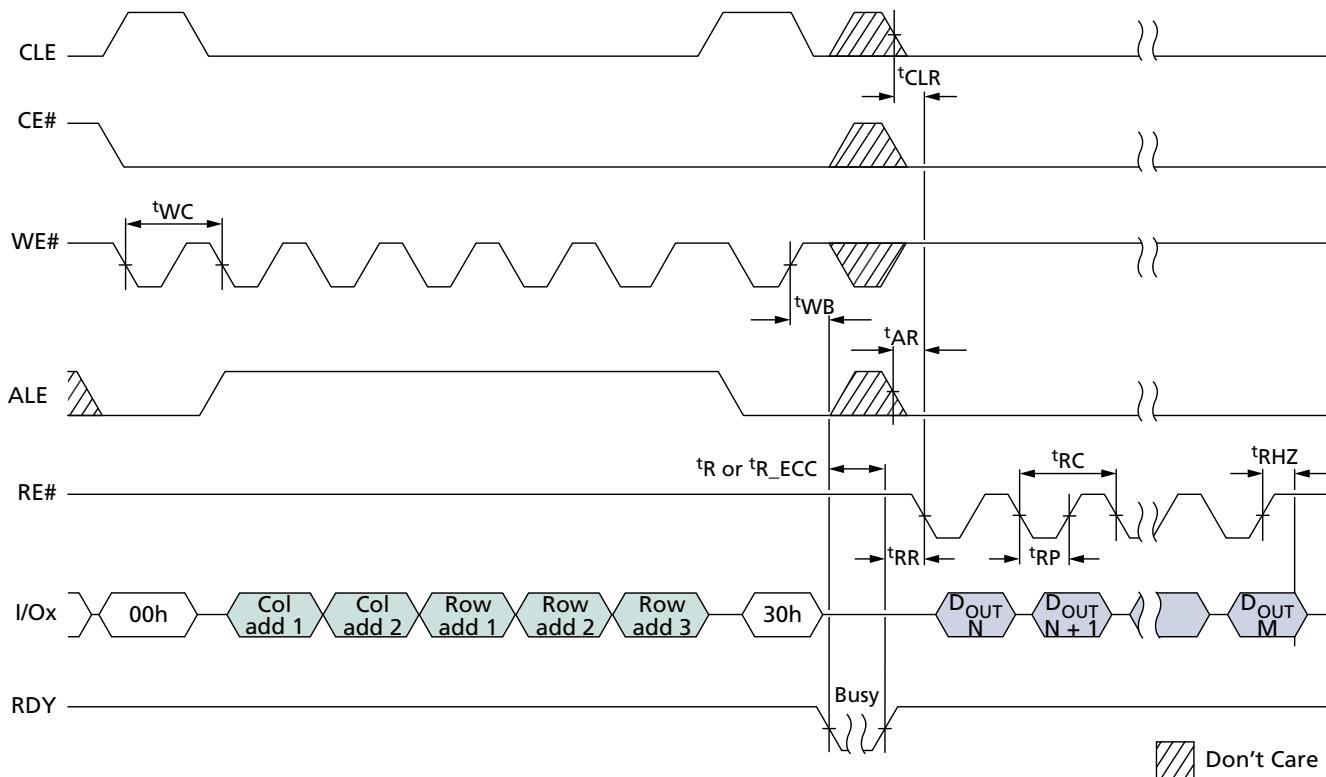
**Figure 83: RESET Operation**

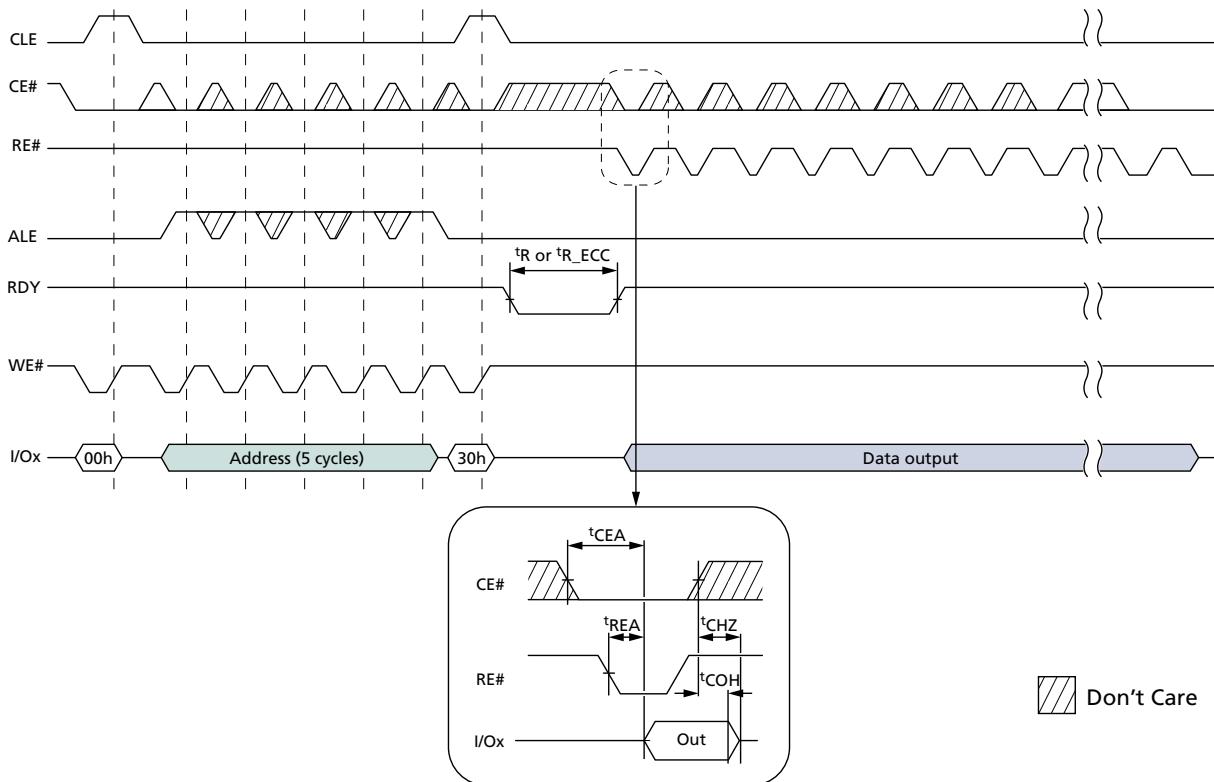


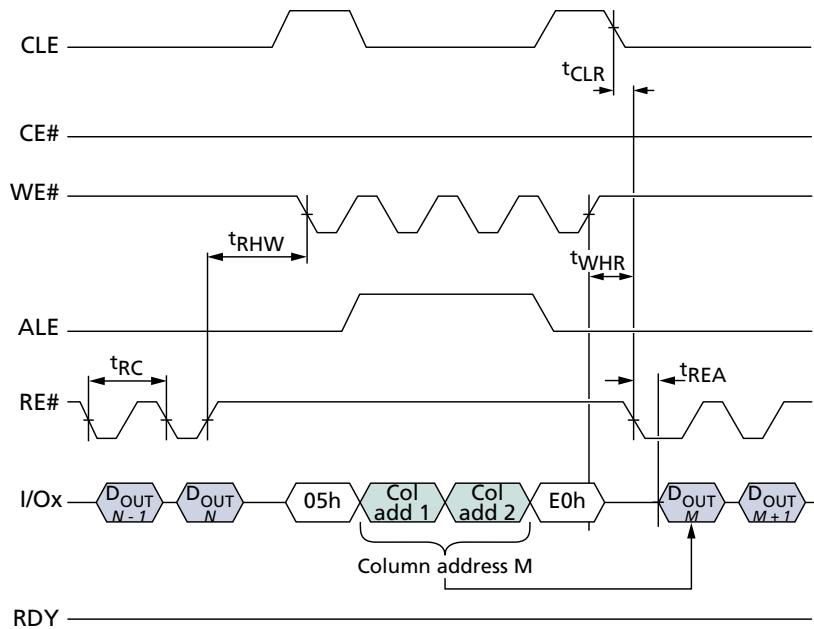
**Figure 84: READ STATUS Cycle**

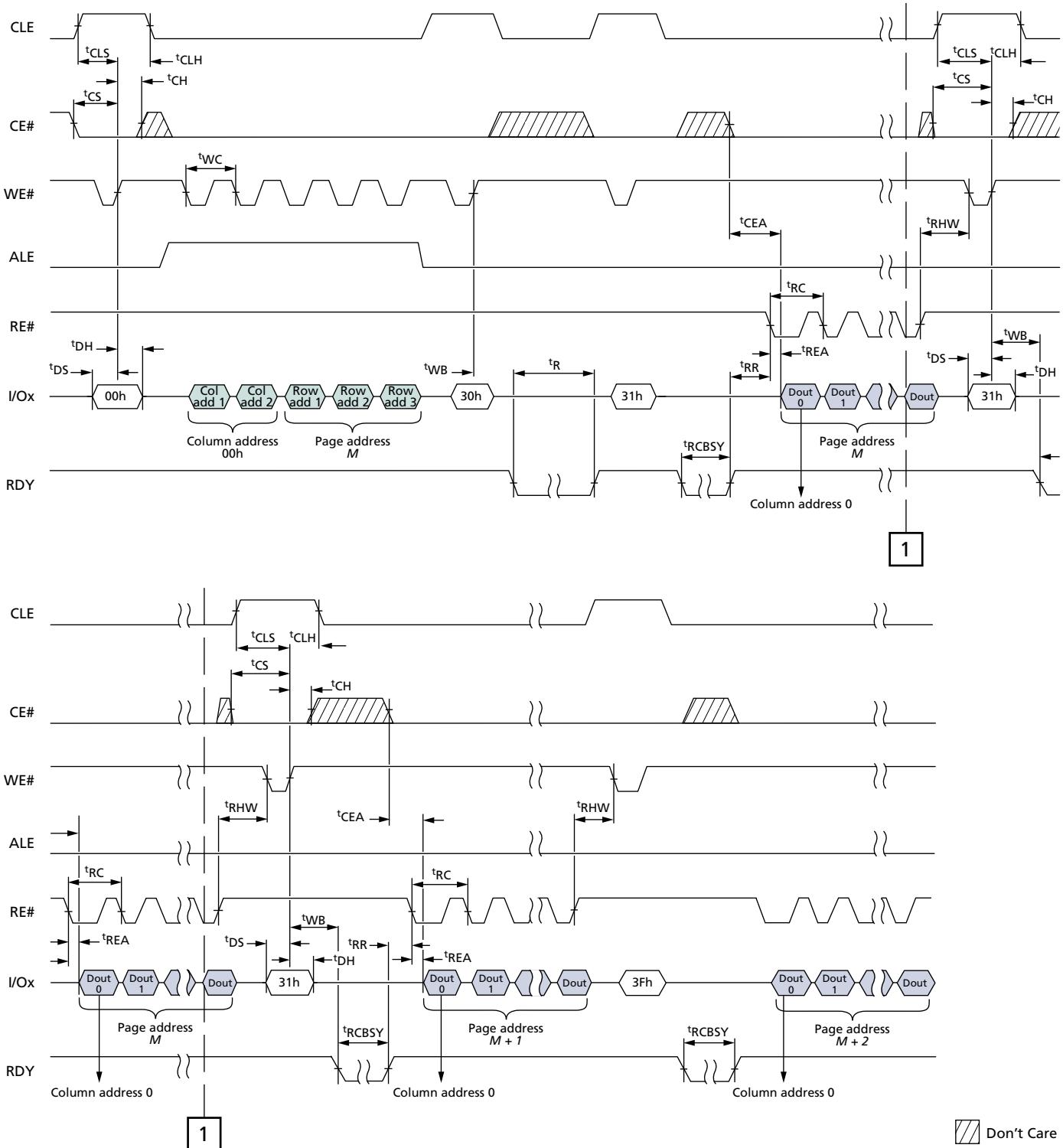


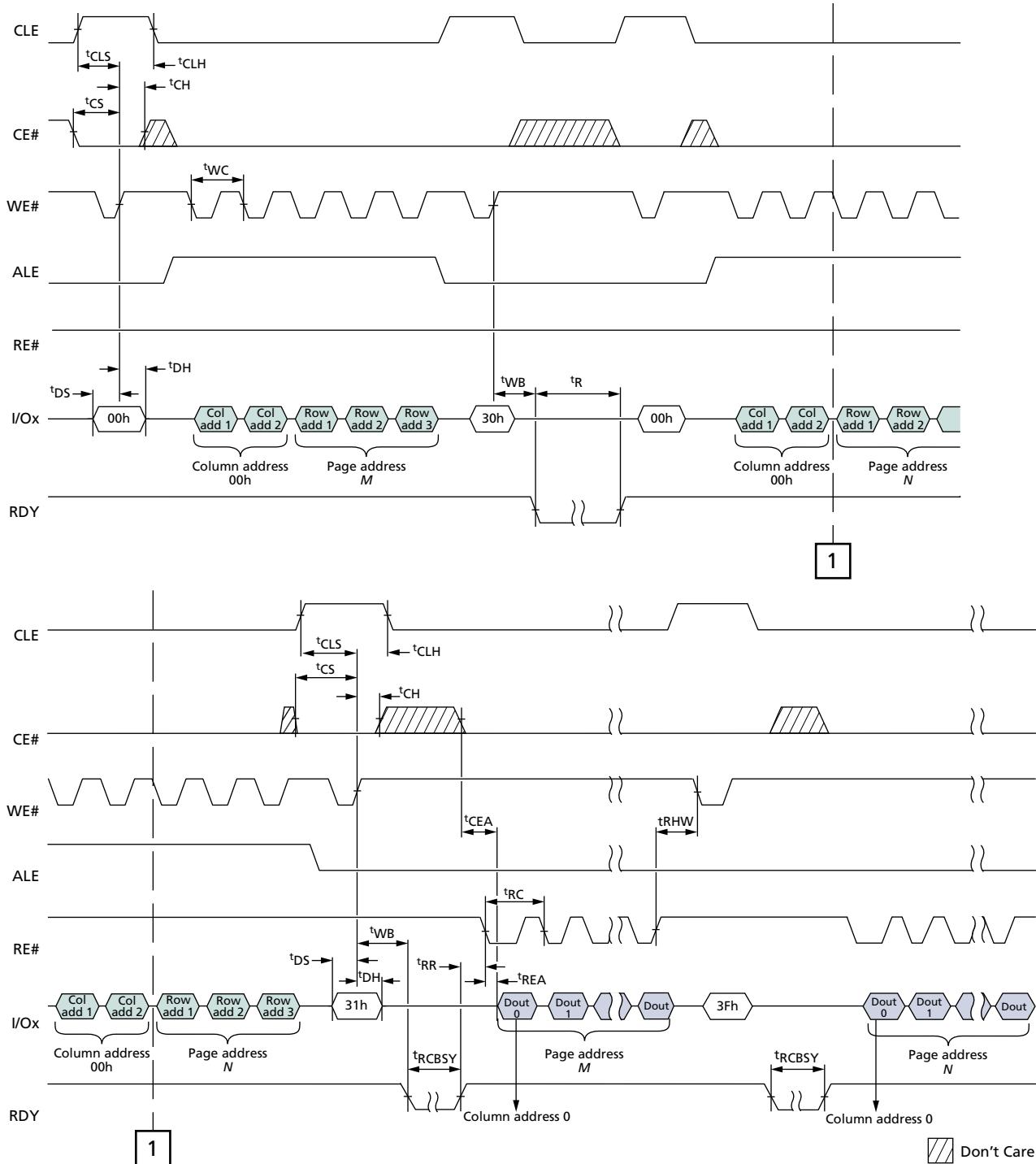
**Figure 85: READ STATUS ENHANCED Cycle****Figure 86: READ PARAMETER PAGE**

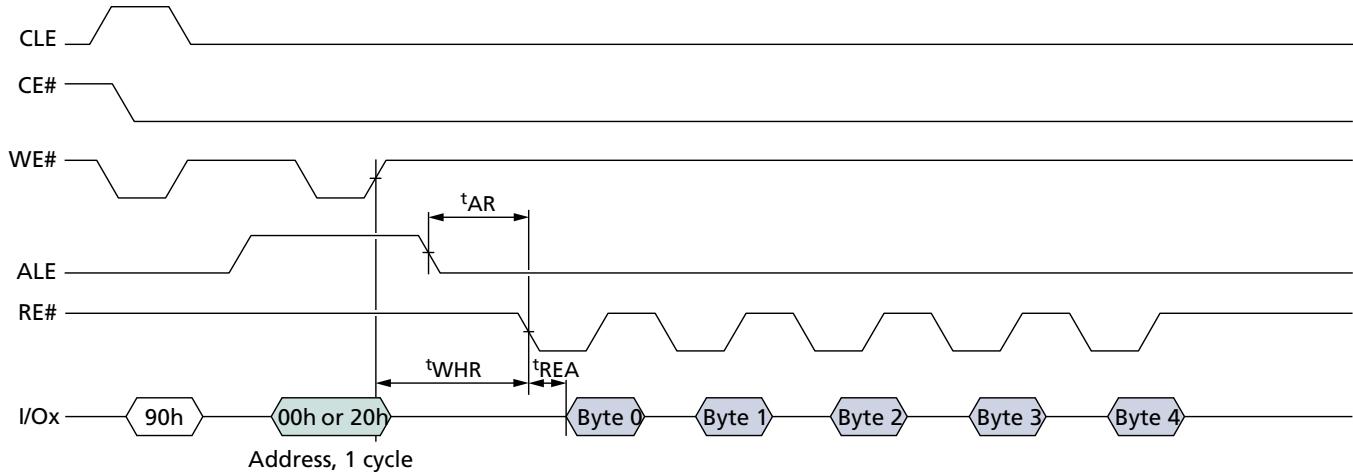
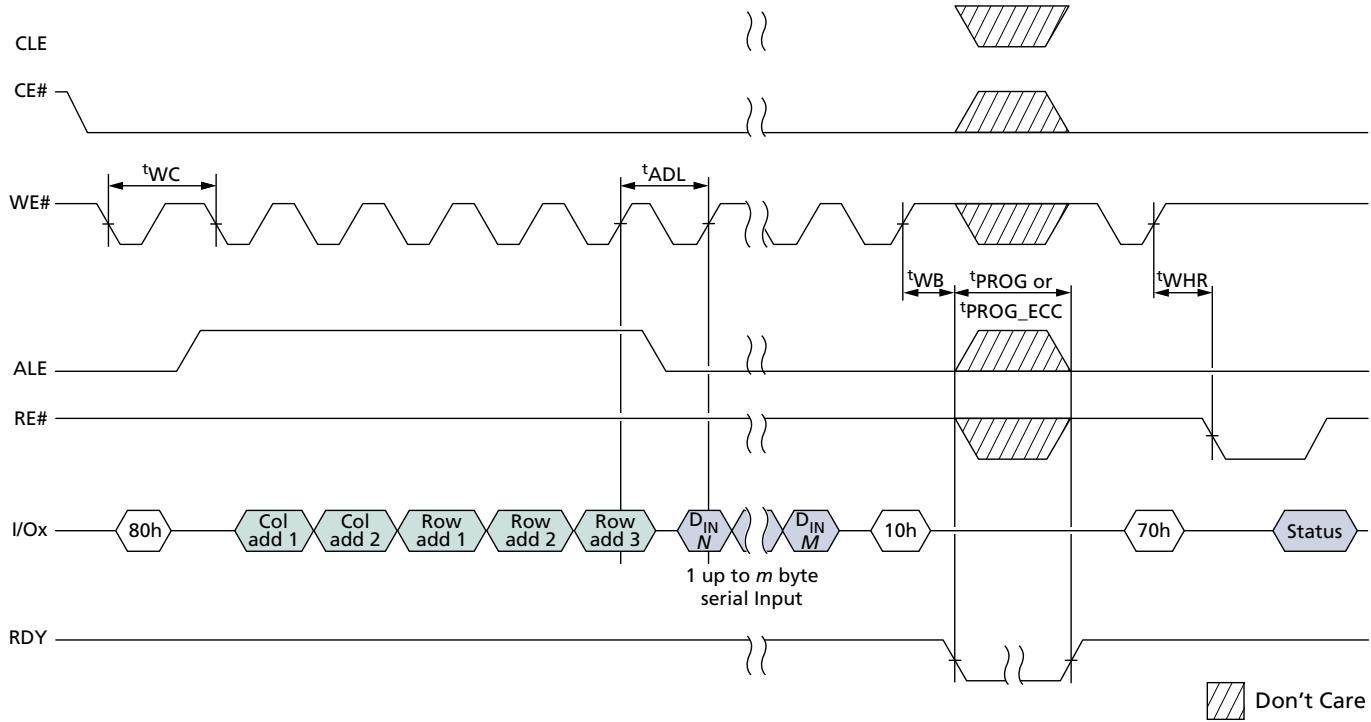
**Figure 87: READ PAGE**

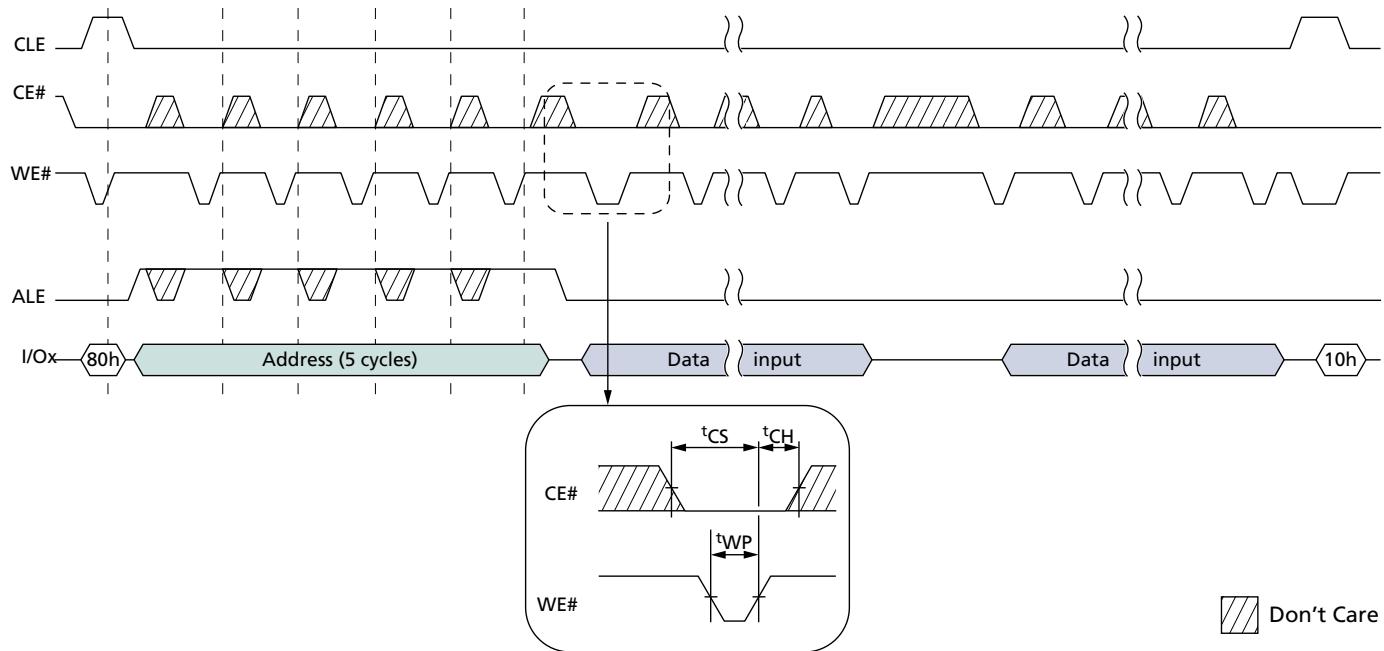
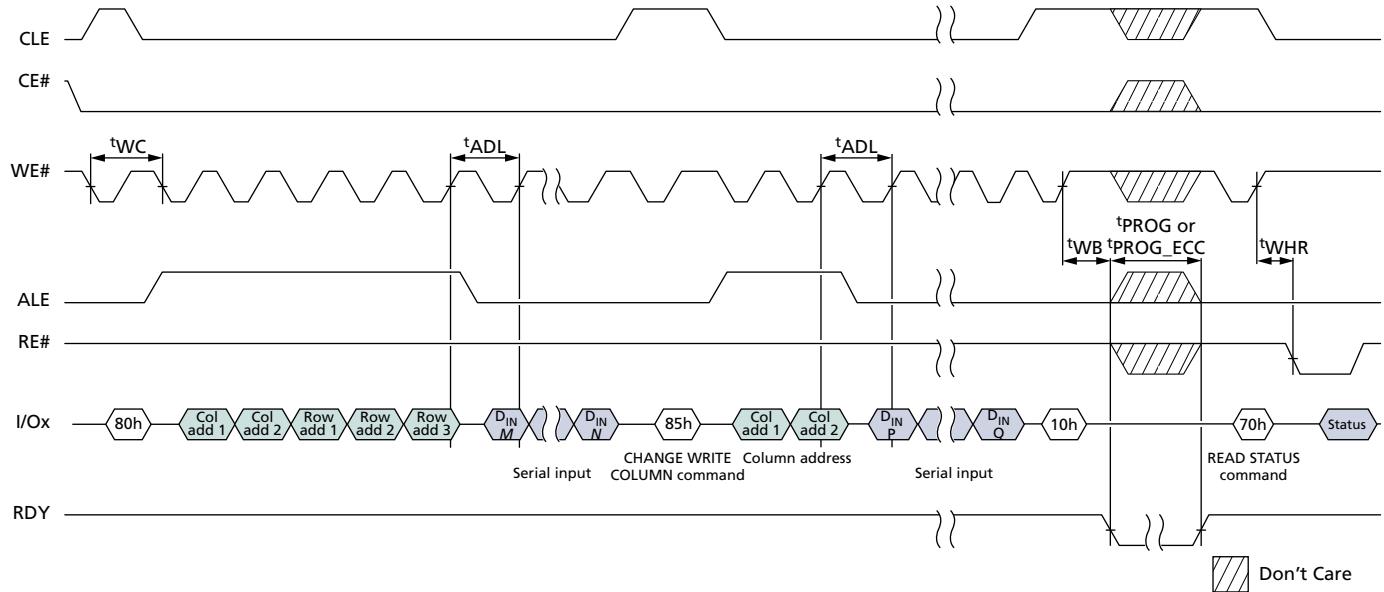
**Figure 88: READ PAGE Operation with CE# "Don't Care"**

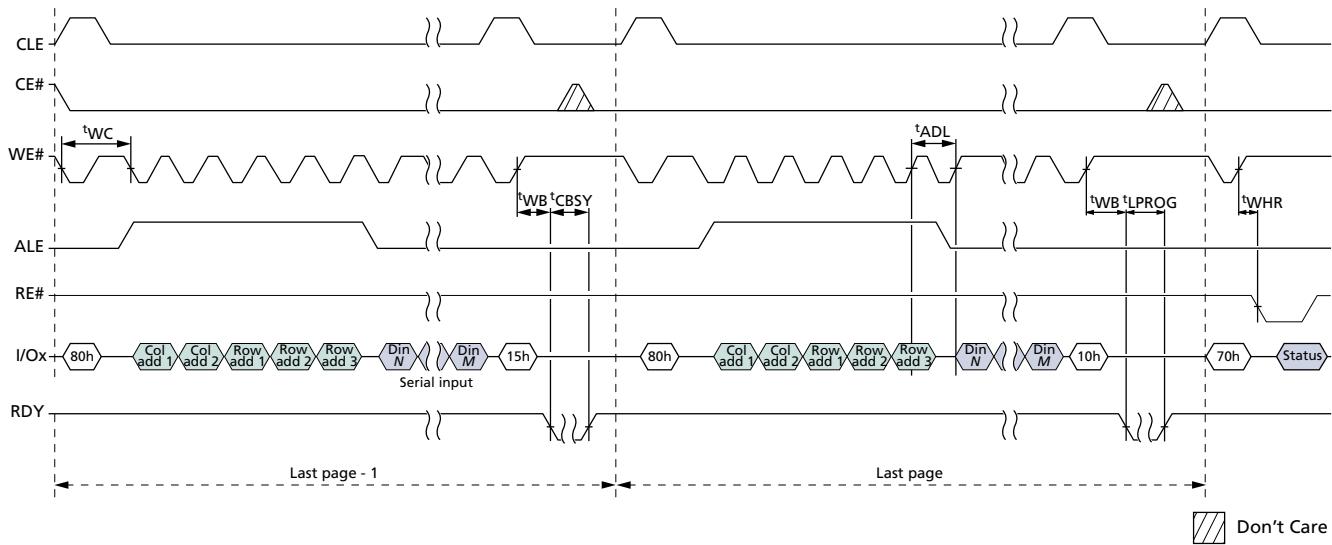
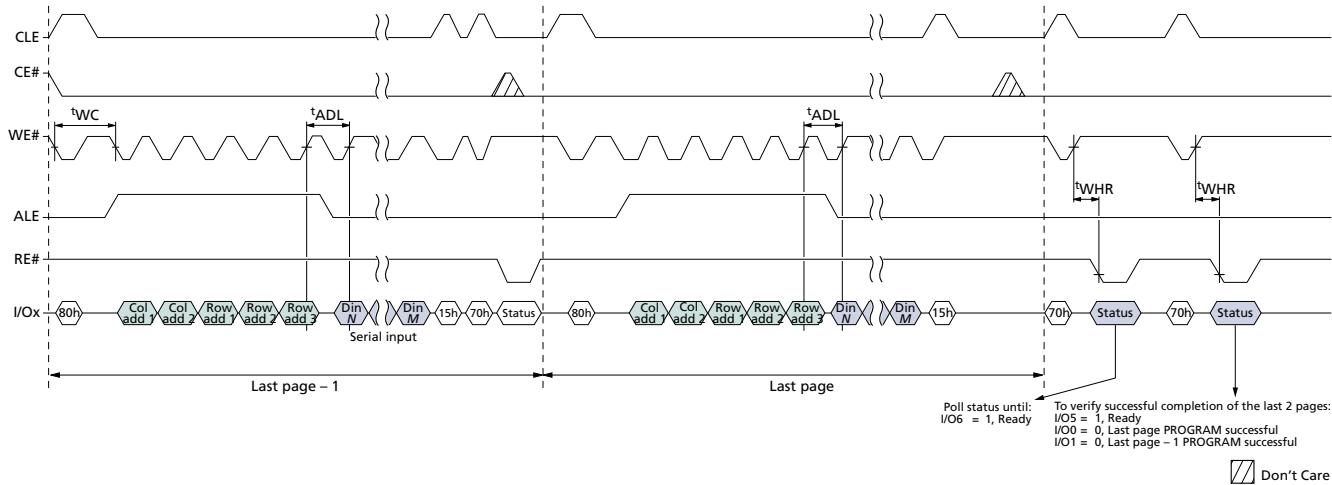
**Figure 89: RANDOM DATA READ**

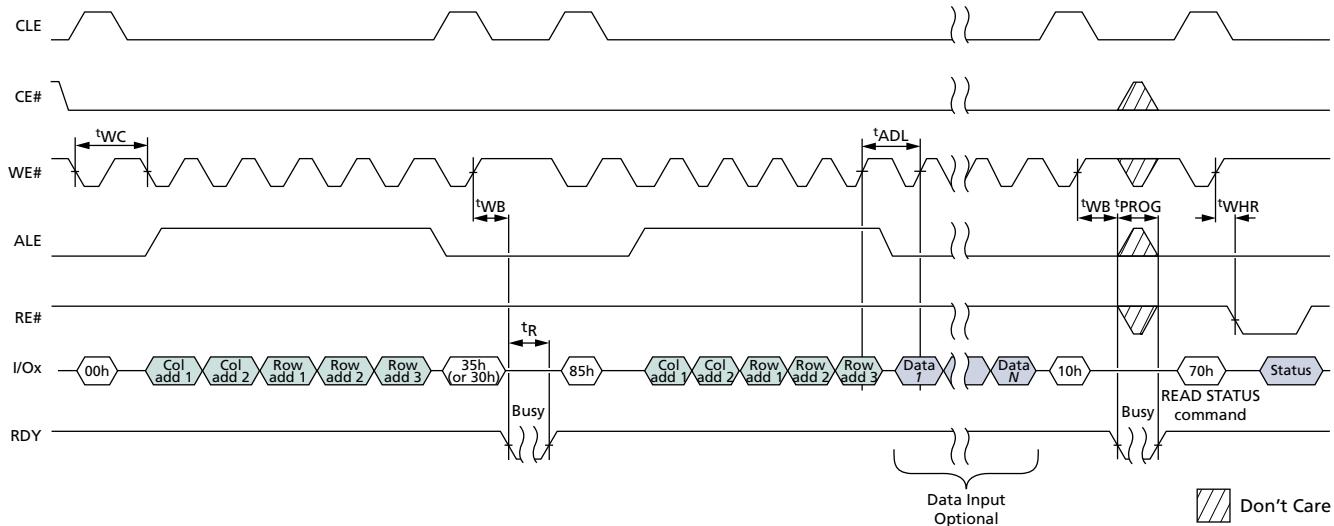
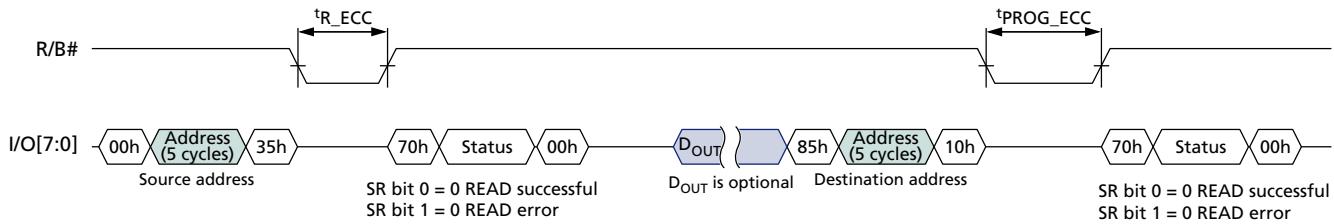
**Figure 90: READ PAGE CACHE SEQUENTIAL**

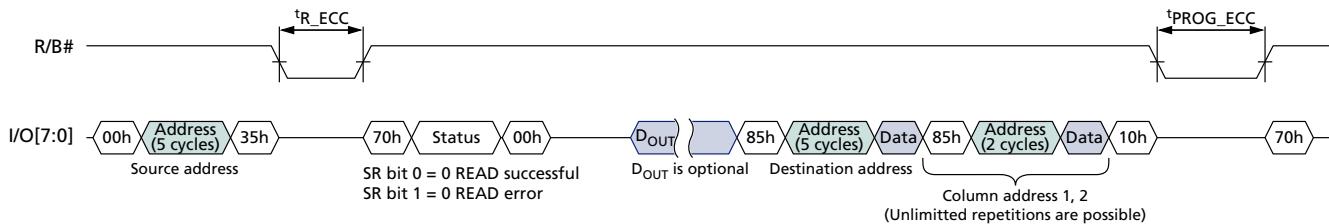
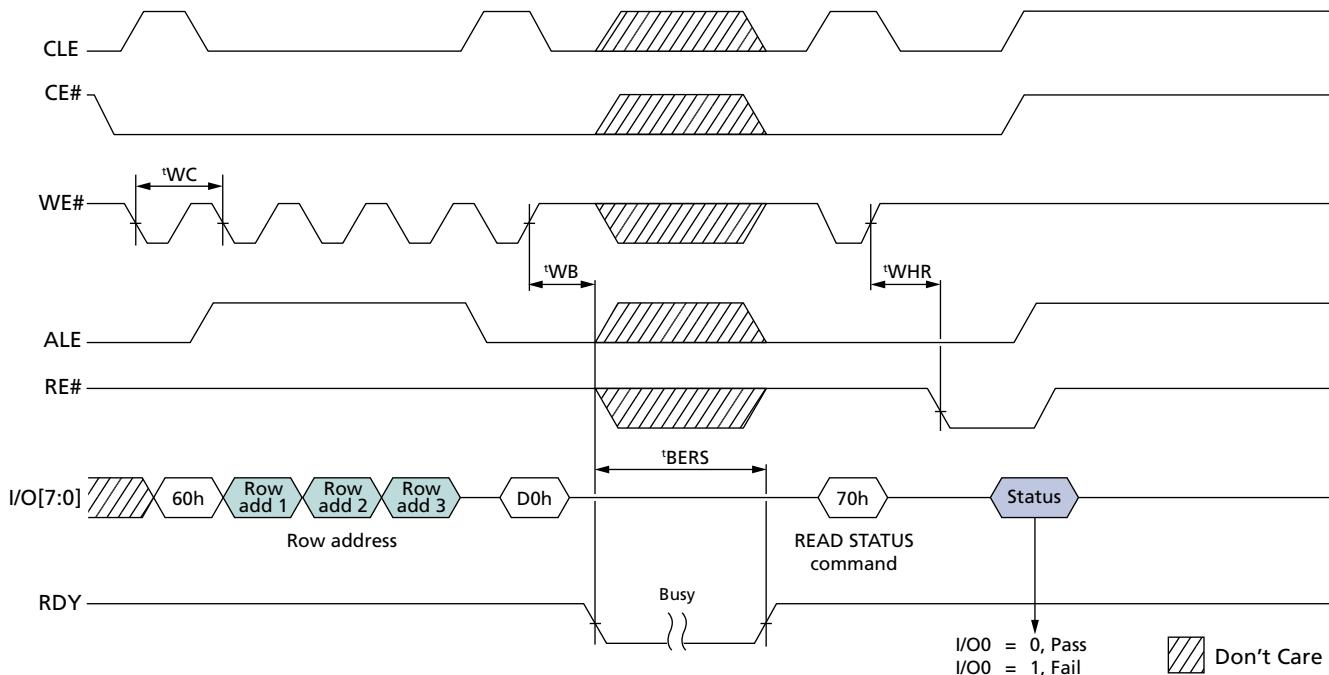
**Figure 91: READ PAGE CACHE RANDOM**

**Figure 92: READ ID Operation****Figure 93: PROGRAM PAGE Operation**

**Figure 94: PROGRAM PAGE Operation with CE# "Don't Care"****Figure 95: PROGRAM PAGE Operation with RANDOM DATA INPUT**

**Figure 96: PROGRAM PAGE CACHE****Figure 97: PROGRAM PAGE CACHE Ending on 15h**

**Figure 98: INTERNAL DATA MOVE****Figure 99: INTERNAL DATA MOVE (85h-10h) with Internal ECC Enabled**

**Figure 100: INTERNAL DATA MOVE (85h-10h) with Random Data Input with Internal ECC Enabled**

**Figure 101: ERASE BLOCK Operation**




## 2Gb: x16, x32 Mobile LPDDR SDRAM

### Features

- $V_{DD}/V_{DDQ} = 1.70\text{--}1.95V$
- Bidirectional data strobe per byte of data (DQS)
- Internal, pipelined double data rate (DDR) architecture; two data accesses per clock cycle
- Differential clock inputs (CK and CK#)
- Commands entered on each positive CK edge
- DQS edge-aligned with data for READs; center-aligned with data for WRITEs
- 4 internal banks for concurrent operation
- Data masks (DM) for masking write data—one mask per byte
- Programmable burst lengths (BL): 2, 4, 8, or 16
- Concurrent auto precharge option is supported
- Auto refresh and self refresh modes
- 1.8V LVCMSO-compatible inputs
- Temperature-compensated self refresh (TCSR)
- Partial-array self refresh (PASR)
- Deep power-down (DPD)
- Status read register (SRR)
- Selectable output drive strength (DS)
- Clock stop capability
- 64ms refresh

**Table 37: Configuration Addressing – 2Gb**

Architecture	128 Meg x 16	64 Meg x 32	Reduced Page-Size Option 128 Meg x 16	Reduced Page-Size Option 64 Meg x 32
Configuration	32 Meg x 16 x 4 banks	16 Meg x 32 x 4 banks	32 Meg x 16 x 4 banks	16 Meg x 32 x 4 banks
Refresh count	8K	8K	8K	8K
Row addressing	16K A[13:0]	16K A[13:0]	32K A[14:0]	32K A[14:0]
Column addressing	2K A11, A[9:0]	1K A[9:0]	1K A[9:0]	512K A[8:0]



## General Description

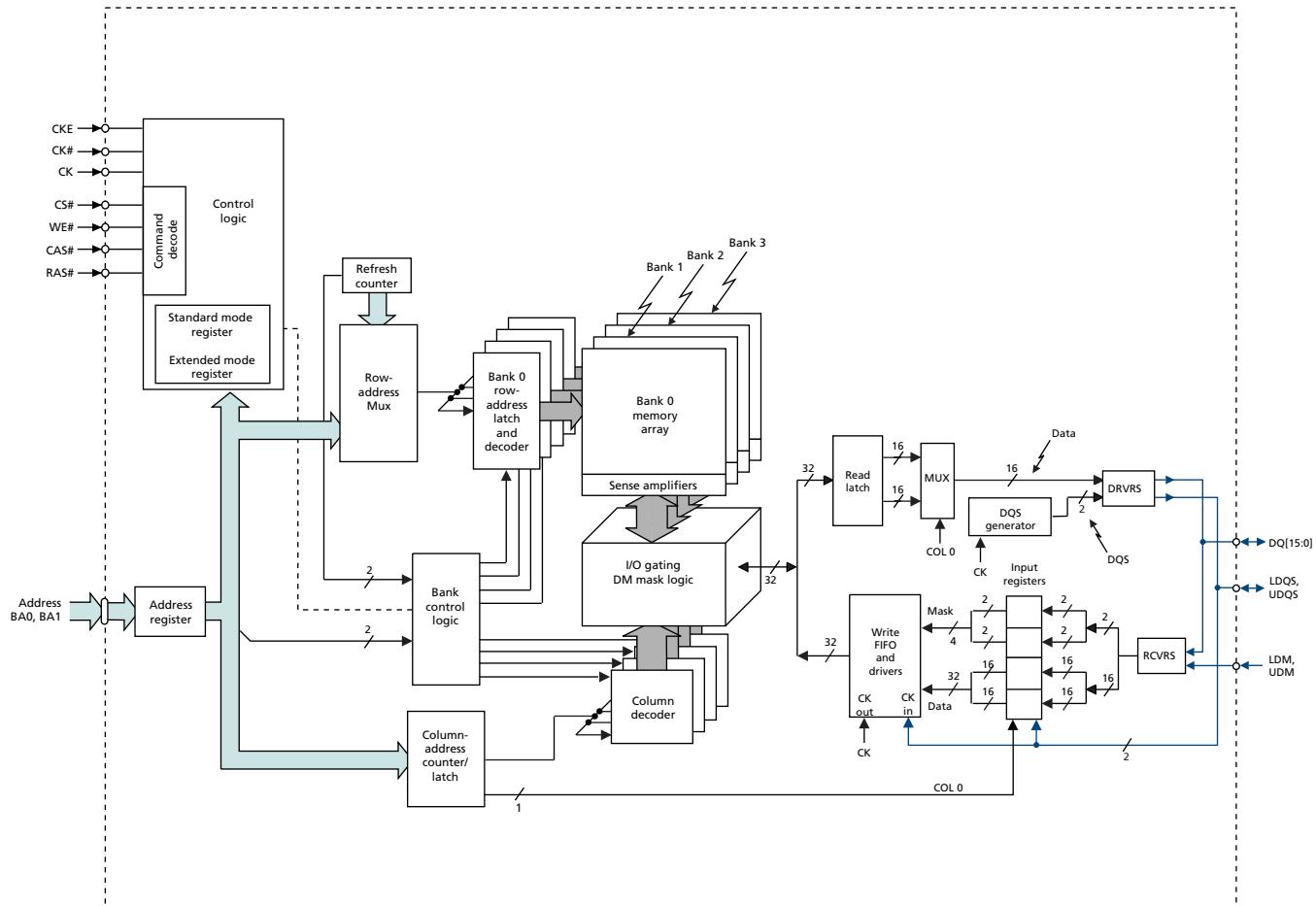
The 2Gb Mobile low-power DDR SDRAM is a high-speed CMOS, dynamic random-access memory containing 2,147,483,648 bits. It is internally configured as a quad-bank DRAM. Each of the x16's 536,870,912-bit banks is organized as 16,384 rows by 2048 columns by 16 bits. Each of the x32's 536,870,912-bit banks is organized as 16,384 rows by 1024 columns by 32 bits. In the reduced page-size (LG) option, each of the x32's 536,870,912-bit banks is organized as 32,768 rows by 512 columns by 32 bits. In the reduced page-size (R4) option, each of the x16's 536,870,912-bit banks is organized as 32,768 rows by 1024 columns x 16 bits.

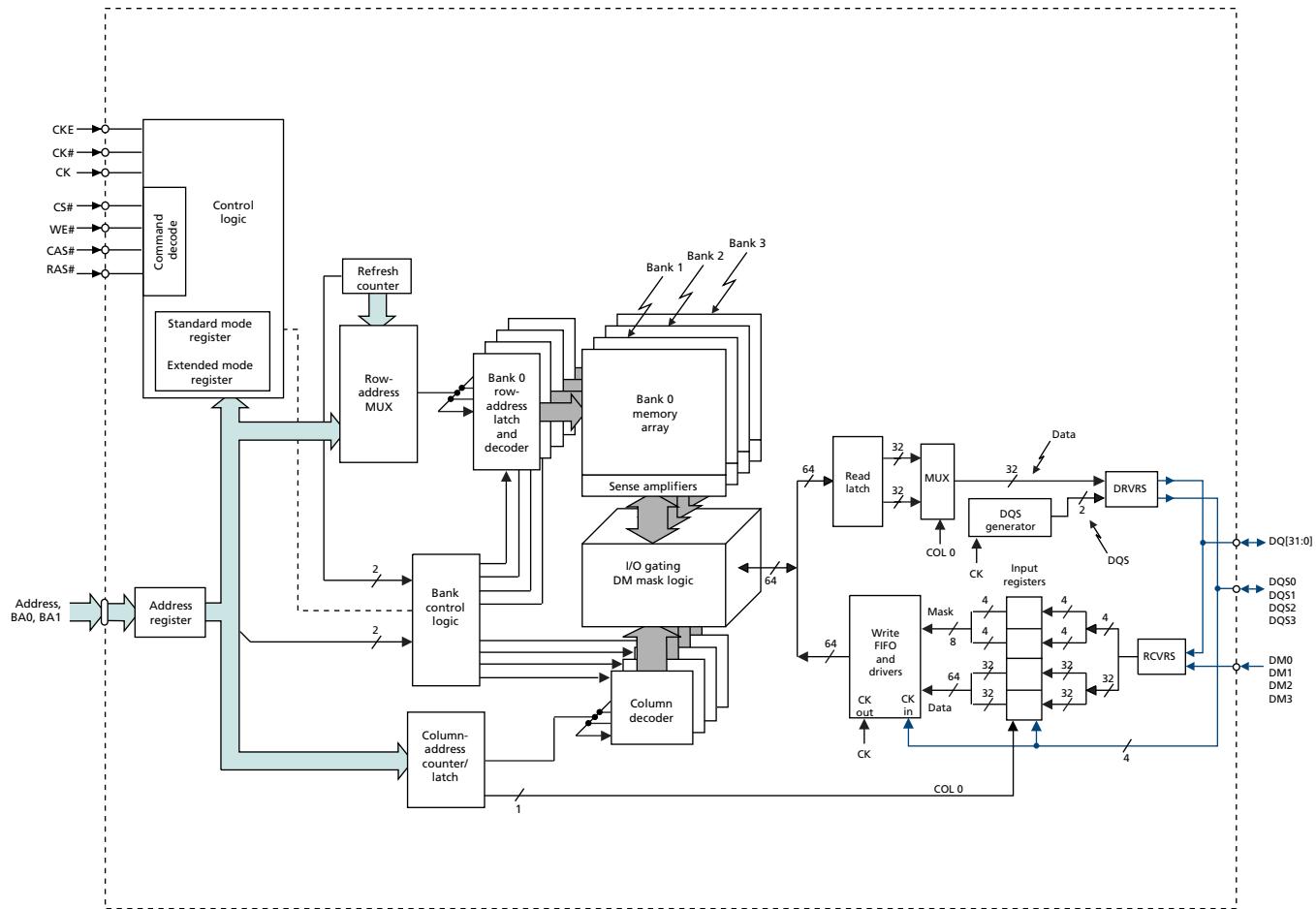
**Note:**

1. Throughout this data sheet, various figures and text refer to DQs as "DQ." DQ should be interpreted as any and all DQ collectively, unless specifically stated otherwise. Additionally, the x16 is divided into 2 bytes: the lower byte and the upper byte. For the lower byte (DQ[7:0]), DM refers to LDM and DQS refers to LDQS. For the upper byte (DQ[15:8]), DM refers to UDM and DQS refers to UDQS. The x32 is divided into 4 bytes. For DQ[7:0], DM refers to DM0 and DQS refers to DQS0. For DQ[15:8], DM refers to DM1 and DQS refers to DQS1. For DQ[23:16], DM refers to DM2 and DQS refers to DQS2. For DQ[31:24], DM refers to DM3 and DQS refers to DQS3.
2. Complete functionality is described throughout the document; any page or diagram may have been simplified to convey a topic and may not be inclusive of all requirements.
3. Any specific requirement takes precedence over a general statement.

## Functional Block Diagrams

**Figure 102: Functional Block Diagram (x16)**



**Figure 103: Functional Block Diagram (x32)**



## Electrical Specifications

Stresses greater than those listed may cause permanent damage to the device. This is a 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 implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

**Table 38: Absolute Maximum Ratings**

Note 1 applies to all parameters in this table

Parameter	Symbol	Min	Max	Unit
$V_{DD}/V_{DDQ}$ supply voltage relative to $V_{SS}$	$V_{DD}/V_{DDQ}$	-1.0	2.4	V
Voltage on any pin relative to $V_{SS}$	$V_{IN}$	-0.5	2.4 or ( $V_{DDQ} + 0.3V$ ), whichever is less	V
Storage temperature (plastic)	$T_{STG}$	-55	150	°C

Note: 1.  $V_{DD}$  and  $V_{DDQ}$  must be within 300mV of each other at all times.  $V_{DDQ}$  must not exceed  $V_{DD}$ .

**Table 39: AC/DC Electrical Characteristics and Operating Conditions**

Notes 1–5 apply to all parameters/conditions in this table;  $V_{DD}/V_{DDQ} = 1.70\text{--}1.95V$

Parameter/Condition	Symbol	Min	Max	Unit	Notes
Supply voltage	$V_{DD}$	1.70	1.95	V	6, 7
I/O supply voltage	$V_{DDQ}$	1.70	1.95	V	6, 7
<b>Address and command inputs</b>					
Input voltage high	$V_{IH}$	$0.8 \times V_{DDQ}$	$V_{DDQ} + 0.3$	V	8, 9
Input voltage low	$V_{IL}$	-0.3	$0.2 \times V_{DDQ}$	V	8, 9
<b>Clock inputs (CK, CK#)</b>					
DC input voltage	$V_{IN}$	-0.3	$V_{DDQ} + 0.3$	V	10
DC input differential voltage	$V_{ID(DC)}$	$0.4 \times V_{DDQ}$	$V_{DDQ} + 0.6$	V	10, 11
AC input differential voltage	$V_{ID(AC)}$	$0.6 \times V_{DDQ}$	$V_{DDQ} + 0.6$	V	10, 11
AC differential crossing voltage	$V_{IX}$	$0.4 \times V_{DDQ}$	$0.6 \times V_{DDQ}$	V	10, 12
<b>Data inputs</b>					
DC input high voltage	$V_{IH(DC)}$	$0.7 \times V_{DDQ}$	$V_{DDQ} + 0.3$	V	8, 9, 13
DC input low voltage	$V_{IL(DC)}$	-0.3	$0.3 \times V_{DDQ}$	V	8, 9, 13
AC input high voltage	$V_{IH(AC)}$	$0.8 \times V_{DDQ}$	$V_{DDQ} + 0.3$	V	8, 9, 13
AC input low voltage	$V_{IL(AC)}$	-0.3	$0.2 \times V_{DDQ}$	V	8, 9, 13
<b>Data outputs</b>					
DC output high voltage: Logic 1 ( $I_{OH} = -0.1\text{mA}$ )	$V_{OH}$	$0.9 \times V_{DDQ}$	-	V	
DC output low voltage: Logic 0 ( $I_{OL} = 0.1\text{mA}$ )	$V_{OL}$	-	$0.1 \times V_{DDQ}$	V	
<b>Leakage current</b>					
Input leakage current	$I_I$	-1	1	$\mu\text{A}$	
Any input $0V \leq V_{IN} \leq V_{DD}$ (All other pins not under test = 0V)					

**Table 39: AC/DC Electrical Characteristics and Operating Conditions (Continued)**Notes 1–5 apply to all parameters/conditions in this table;  $V_{DD}/V_{DDQ} = 1.70\text{--}1.95\text{V}$ 

Parameter/Condition	Symbol	Min	Max	Unit	Notes
Output leakage current (DQ are disabled; $0\text{V} \leq V_{OUT} \leq V_{DDQ}$ )	$I_{OZ}$	-1.5	1.5	$\mu\text{A}$	
<b>Operating temperature</b>					
Commercial	$T_A$	0	70	$^{\circ}\text{C}$	
Industrial	$T_A$	-40	85	$^{\circ}\text{C}$	
Automotive	$T_A$	-40	105	$^{\circ}\text{C}$	

- Notes:
1. All voltages referenced to  $V_{SS}$ .
  2. All parameters assume proper device initialization.
  3. Tests for AC timing,  $I_{DD}$ , and electrical AC and DC characteristics may be conducted at nominal supply voltage levels, but the related specifications and device operation are guaranteed for the full voltage range specified.
  4. Outputs measured with equivalent load; transmission line delay is assumed to be very small:
- 
5. Timing and  $I_{DD}$  tests may use a  $V_{IL}$ -to- $V_{IH}$  swing of up to 1.5V in the test environment, but input timing is still referenced to  $V_{DDQ}/2$  (or to the crossing point for CK/CK#). The output timing reference voltage level is  $V_{DDQ}/2$ .
6. Any positive glitch must be less than one-third of the clock cycle and not more than +200mV or 2.0V, whichever is less. Any negative glitch must be less than one-third of the clock cycle and not exceed either -150mV or +1.6V, whichever is more positive.
7.  $V_{DD}$  and  $V_{DDQ}$  must track each other and  $V_{DDQ}$  must be less than or equal to  $V_{DD}$ .
8. To maintain a valid level, the transitioning edge of the input must:
  - Sustain a constant slew rate from the current AC level through to the target AC level,  $V_{IL(AC)}$  Or  $V_{IH(AC)}$ .
  - Reach at least the target AC level.
  - After the AC target level is reached, continue to maintain at least the target DC level,  $V_{IL(DC)}$  or  $V_{IH(DC)}$ .
9.  $V_{IH}$  overshoot:  $V_{IHmax} = V_{DDQ} + 1.0\text{V}$  for a pulse width  $\leq 3\text{ns}$  and the pulse width cannot be greater than one-third of the cycle rate.  $V_{IL}$  undershoot:  $V_{ILmin} = -1.0\text{V}$  for a pulse width  $\leq 3\text{ns}$  and the pulse width cannot be greater than one-third of the cycle rate.
10. CK and CK# input slew rate must be  $\geq 1\text{ V/ns}$  (2 V/ns if measured differentially).
11.  $V_{ID}$  is the magnitude of the difference between the input level on CK and the input level on CK#.
12. The value of  $V_{IX}$  is expected to equal  $V_{DDQ}/2$  of the transmitting device and must track variations in the DC level of the same.
13. DQ and DM input slew rates must not deviate from DQS by more than 10%. 50ps must be added to  $t_{DS}$  and  $t_{DH}$  for each 100 mV/ns reduction in slew rate. If slew rate exceeds 4 V/ns, functionality is uncertain.

**Table 40: Capacitance (x16, x32)**

Note 1 applies to all the parameters in this table

Parameter	Symbol	Min	Max	Unit	Notes
Input capacitance: CK, CK#	$C_{CK}$	1.5	3.5	pF	
Delta input capacitance: CK, CK#	$C_{DCK}$	–	0.25	pF	2
Input capacitance: command and address	$C_I$	1.5	3.5	pF	
Delta input capacitance: command and address	$C_{DI}$	–	0.5	pF	2
Input/output capacitance: DQ, DQS, DM	$C_{IO}$	2.0	4.5	pF	
Delta input/output capacitance: DQ, DQS, DM	$C_{DIO}$	–	0.5	pF	3

- Notes:
1. This parameter is sampled.  $V_{DD}/V_{DDQ} = 1.70\text{--}1.95V$ ,  $f = 100\text{ MHz}$ ,  $T_A = 25^\circ\text{C}$ ,  $V_{OUT(DC)} = V_{DDQ}/2$ ,  $V_{OUT}$  (peak-to-peak) = 0.2V. DM input is grouped with I/O pins, reflecting the fact that they are matched in loading.
  2. The input capacitance per pin group will not differ by more than this maximum amount for any given device.
  3. The I/O capacitance per DQS and DQ byte/group will not differ by more than this maximum amount for any given device.



## Electrical Specifications – I<sub>DD</sub> Parameters

**Table 41: I<sub>DD</sub> Specifications and Conditions, -40°C to +85°C (x16)**

Notes 1–5 apply to all the parameters/conditions in this table; V<sub>DD</sub>/V<sub>DDQ</sub> = 1.70–1.95V

Parameter/Condition	Symbol	Max				Unit	Notes	
		-5	-54	-6	-75			
Operating 1 bank active precharge current: t <sub>RC</sub> = t <sub>RC</sub> (MIN); t <sub>CK</sub> = t <sub>CK</sub> (MIN); CKE is HIGH; CS is HIGH between valid commands; Address inputs are switching every 2 clock cycles; Data bus inputs are stable	I <sub>DD0</sub>	100	100	80	70	mA	6	
Precharge power-down standby current: All banks idle; CKE is LOW; CS is HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2P</sub>	900	900	900	900	µA	7, 8	
Precharge power-down standby current: Clock stopped; All banks idle; CKE is LOW; CS is HIGH; CK = LOW, CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2PS</sub>	900	900	900	900	µA	7	
Precharge nonpower-down standby current: All banks idle; CKE = HIGH; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2N</sub>	15	15	15	12	mA	9	
Precharge nonpower-down standby current: Clock stopped; All banks idle; CKE = HIGH; CS = HIGH; CK = LOW, CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2NS</sub>	9	9	8	8	mA	9	
Active power-down standby current: 1 bank active; CKE = LOW; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3P</sub>	5	5	5	5	mA	8	
Active power-down standby current: Clock stopped; 1 bank active; CKE = LOW; CS = HIGH; CK = LOW; CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3PS</sub>	5	5	5	5	mA		
Active nonpower-down standby: 1 bank active; CKE = HIGH; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3N</sub>	17	17	16	15	mA	6	
Active nonpower-down standby: Clock stopped; 1 bank active; CKE = HIGH; CS = HIGH; CK = LOW; CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3NS</sub>	14	14	13	12	mA	6	
Operating burst read: 1 bank active; BL = 4; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Continuous READ bursts; lout = 0mA; Address inputs are switching every 2 clock cycles; 50% data changing each burst	I <sub>DD4R</sub>	130	125	115	105	mA	6	
Operating burst write: 1 bank active; BL = 4; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Continuous WRITE bursts; Address inputs are switching; 50% data changing each burst	I <sub>DD4W</sub>	130	125	115	105	mA	6	
Auto refresh: Burst refresh; CKE = HIGH; Address and control inputs are switching; Data bus inputs are stable	t <sub>RFC</sub> = 138ns	I <sub>DD5</sub>	170	170	170	170	mA	10
	t <sub>RFC</sub> = t <sub>REFI</sub>	I <sub>DD5A</sub>	12	12	12	12	mA	10, 11
Deep power-down current: Address and control balls are stable; Data bus inputs are stable	I <sub>DD8</sub>	10	10	10	10	µA	7, 13	

**Table 42: I<sub>DD</sub> Specifications and Conditions, -40°C to +85°C (x32)**Notes 1–5 apply to all the parameters/conditions in this table; V<sub>DD</sub>/V<sub>DDQ</sub> = 1.70–1.95V

Parameter/Condition	Symbol	Max				Unit	Notes	
		-5	-54	-6	-75			
Operating 1 bank active precharge current: t <sub>RC</sub> = t <sub>RC</sub> (MIN); t <sub>CK</sub> = t <sub>CK</sub> (MIN); CKE is HIGH; CS is HIGH between valid commands; Address inputs are switching every 2 clock cycles; Data bus inputs are stable	I <sub>DD0</sub>	100	100	80	70	mA	6	
Precharge power-down standby current: All banks idle; CKE is LOW; CS is HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2P</sub>	900	900	900	900	µA	7, 8	
Precharge power-down standby current: Clock stopped; All banks idle; CKE is LOW; CS is HIGH, CK = LOW, CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2PS</sub>	900	900	900	900	µA	7	
Precharge nonpower-down standby current: All banks idle; CKE = HIGH; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2N</sub>	15	15	15	12	mA	9	
Precharge nonpower-down standby current: Clock stopped; All banks idle; CKE = HIGH; CS = HIGH; CK = LOW, CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2NS</sub>	9	9	8	8	mA	9	
Active power-down standby current: 1 bank active; CKE = LOW; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3P</sub>	5	5	5	5	mA	8	
Active power-down standby current: Clock stopped; 1 bank active; CKE = LOW; CS = HIGH; CK = LOW; CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3PS</sub>	5	5	5	5	mA		
Active nonpower-down standby: 1 bank active; CKE = HIGH; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3N</sub>	17	17	16	15	mA	6	
Active nonpower-down standby: Clock stopped; 1 bank active; CKE = HIGH; CS = HIGH; CK = LOW; CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3NS</sub>	14	14	13	12	mA	6	
Operating burst read: 1 bank active; BL = 4; CL = 3; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Continuous READ bursts; I <sub>out</sub> = 0mA; Address inputs are switching every 2 clock cycles; 50% data changing each burst	I <sub>DD4R</sub>	150	145	140	120	mA	6	
Operating burst write: One bank active; BL = 4; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Continuous WRITE bursts; Address inputs are switching; 50% data changing each burst	I <sub>DD4W</sub>	150	145	140	120	mA	6	
Auto refresh: Burst refresh; CKE = HIGH; Address and control inputs are switching; Data bus inputs are stable	t <sub>RFC</sub> = 138ns	I <sub>DD5</sub>	170	170	170	170	mA	10
	t <sub>RFC</sub> = t <sub>REFI</sub>	I <sub>DD5A</sub>	12	12	12	12	mA	10, 11
Deep power-down current: Address and control pins are stable; Data bus inputs are stable	I <sub>DD8</sub>	10	10	10	10	µA	7, 13	

**Table 43: I<sub>DD</sub> Specifications and Conditions, -40°C to +105°C (x16)**Notes 1–5 apply to all the parameters/conditions in this table; V<sub>DD</sub>/V<sub>DDQ</sub> = 1.70–1.95V

Parameter/Condition	Symbol	Max				Unit	Notes	
		-5	-54	-6	-75			
Operating 1 bank active precharge current: t <sub>RC</sub> = t <sub>RC</sub> (MIN); t <sub>CK</sub> = t <sub>CK</sub> (MIN); CKE is HIGH; CS is HIGH between valid commands; Address inputs are switching every 2 clock cycles; Data bus inputs are stable	I <sub>DD0</sub>	100	100	80	70	mA	6	
Precharge power-down standby current: All banks idle; CKE is LOW; CS is HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2P</sub>	1500	1500	1500	1500	µA	7, 8	
Precharge power-down standby current: Clock stopped; All banks idle; CKE is LOW; CS is HIGH; CK = LOW, CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2PS</sub>	1500	1500	1500	1500	µA	7	
Precharge nonpower-down standby current: All banks idle; CKE = HIGH; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2N</sub>	19	19	19	16	mA	9	
Precharge nonpower-down standby current: Clock stopped; All banks idle; CKE = HIGH; CS = HIGH; CK = LOW, CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2NS</sub>	13	13	12	12	mA	9	
Active power-down standby current: 1 bank active; CKE = LOW; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3P</sub>	9	9	9	9	mA	8	
Active power-down standby current: Clock stopped; 1 bank active; CKE = LOW; CS = HIGH; CK = LOW; CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3PS</sub>	9	9	9	9	mA		
Active nonpower-down standby: 1 bank active; CKE = HIGH; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3N</sub>	21	21	20	19	mA	6	
Active nonpower-down standby: Clock stopped; 1 bank active; CKE = HIGH; CS = HIGH; CK = LOW; CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3NS</sub>	18	18	17	15	mA	6	
Operating burst read: 1 bank active; BL = 4; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Continuous READ bursts; I <sub>out</sub> = 0mA; Address inputs are switching every 2 clock cycles; 50% data changing each burst	I <sub>DD4R</sub>	130	125	115	105	mA	6	
Operating burst write: 1 bank active; BL = 4; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Continuous WRITE bursts; Address inputs are switching; 50% data changing each burst	I <sub>DD4W</sub>	130	125	115	105	mA	6	
Auto refresh: Burst refresh; CKE = HIGH; Address and control inputs are switching; Data bus inputs are stable	t <sub>RFC</sub> = 138ns	I <sub>DD5</sub>	170	170	170	170	mA	10
	t <sub>RFC</sub> = t <sub>REFI</sub>	I <sub>DD5A</sub>	13	13	13	13	mA	10, 11
Deep power-down current: Address and control balls are stable; Data bus inputs are stable	I <sub>DD8</sub>	15	15	15	15	µA	7, 13	

**Table 44: I<sub>DD</sub> Specifications and Conditions, -40°C to +105°C (x32)**Notes 1–5 apply to all the parameters/conditions in this table; V<sub>DD</sub>/V<sub>DDQ</sub> = 1.70–1.95V

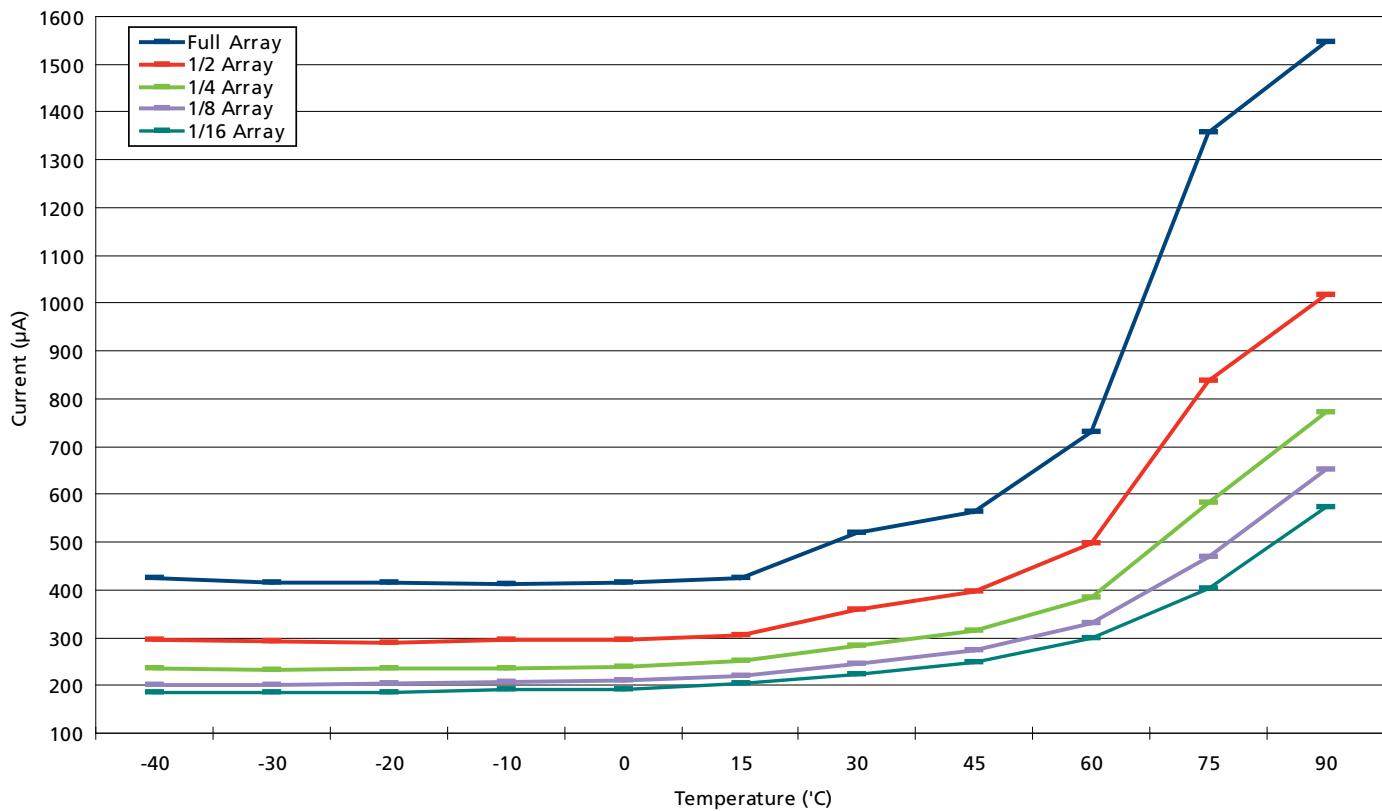
Parameter/Condition	Symbol	Max				Unit	Notes	
		-5	-54	-6	-75			
Operating 1 bank active precharge current: t <sub>RC</sub> = t <sub>RC</sub> (MIN); t <sub>CK</sub> = t <sub>CK</sub> (MIN); CKE is HIGH; CS is HIGH between valid commands; Address inputs are switching every 2 clock cycles; Data bus inputs are stable	I <sub>DD0</sub>	100	100	80	70	mA	6	
Precharge power-down standby current: All banks idle; CKE is LOW; CS is HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2P</sub>	1500	1500	1500	1500	µA	7, 8	
Precharge power-down standby current: Clock stopped; All banks idle; CKE is LOW; CS is HIGH, CK = LOW, CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2PS</sub>	1500	1500	1500	1500	µA	7	
Precharge nonpower-down standby current: All banks idle; CKE = HIGH; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2N</sub>	19	19	19	16	mA	9	
Precharge nonpower-down standby current: Clock stopped; All banks idle; CKE = HIGH; CS = HIGH; CK = LOW, CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD2NS</sub>	13	13	12	12	mA	9	
Active power-down standby current: 1 bank active; CKE = LOW; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3P</sub>	9	9	9	9	mA	8	
Active power-down standby current: Clock stopped; 1 bank active; CKE = LOW; CS = HIGH; CK = LOW; CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3PS</sub>	9	9	9	9	mA		
Active nonpower-down standby: 1 bank active; CKE = HIGH; CS = HIGH; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3N</sub>	21	21	20	19	mA	6	
Active nonpower-down standby: Clock stopped; 1 bank active; CKE = HIGH; CS = HIGH; CK = LOW; CK# = HIGH; Address and control inputs are switching; Data bus inputs are stable	I <sub>DD3NS</sub>	18	18	17	15	mA	6	
Operating burst read: 1 bank active; BL = 4; CL = 3; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Continuous READ bursts; I <sub>out</sub> = 0mA; Address inputs are switching every 2 clock cycles; 50% data changing each burst	I <sub>DD4R</sub>	150	145	140	120	mA	6	
Operating burst write: One bank active; BL = 4; t <sub>CK</sub> = t <sub>CK</sub> (MIN); Continuous WRITE bursts; Address inputs are switching; 50% data changing each burst	I <sub>DD4W</sub>	150	145	140	120	mA	6	
Auto refresh: Burst refresh; CKE = HIGH; Address and control inputs are switching; Data bus inputs are stable	t <sub>RFC</sub> = 138ns	I <sub>DD5</sub>	170	170	170	170	mA	10
	t <sub>RFC</sub> = t <sub>REFI</sub>	I <sub>DD5A</sub>	13	13	13	13	mA	10, 11
Deep power-down current: Address and control pins are stable; Data bus inputs are stable	I <sub>DD8</sub>	15	15	15	15	µA	7, 13	

**Table 45:  $I_{DD6}$  Specifications and Conditions**Notes 1–5, 7, and 12 apply to all the parameters/conditions in this table;  $V_{DD}/V_{DDQ} = 1.70\text{--}1.95\text{V}$ 

<b>Parameter/Condition</b>		<b>Symbol</b>	<b>Value</b>	<b>Units</b>
Self refresh:  CKE = LOW; $t^{CK} = t^{CK}$ (MIN); Address and control inputs are stable; Data bus inputs are stable	Full array, 105°C	$I_{DD6}$	n/a <sup>14</sup>	μA
	Full array, 85°C		2000	μA
	Full array, 45°C		900	μA
	1/2 array, 85°C		1450	μA
	1/2 array, 45°C		700	μA
	1/4 array, 85°C		1230	μA
	1/4 array, 45°C		600	μA
	1/8 array, 85°C		1090	μA
	1/8 array, 45°C		575	μA
	1/16 array, 85°C		1020	μA
	1/16 array, 45°C		550	μA

- Notes:
1. All voltages referenced to  $V_{SS}$ .
  2. Tests for  $I_{DD}$  characteristics may be conducted at nominal supply voltage levels, but the related specifications and device operation are guaranteed for the full voltage range specified.
  3. Timing and  $I_{DD}$  tests may use a  $V_{IL}$ -to- $V_{IH}$  swing of up to 1.5V in the test environment, but input timing is still referenced to  $V_{DDQ/2}$  (or to the crossing point for CK/CK#). The output timing reference voltage level is  $V_{DDQ/2}$ .
  4.  $I_{DD}$  is dependent on output loading and cycle rates. Specified values are obtained with minimum cycle time with the outputs open.
  5.  $I_{DD}$  specifications are tested after the device is properly initialized and values are averaged at the defined cycle rate.
  6. MIN ( $t^{RC}$  or  $t^{RFC}$ ) for  $I_{DD}$  measurements is the smallest multiple of  $t^{CK}$  that meets the minimum absolute value for the respective parameter.  $t^{RASmax}$  for  $I_{DD}$  measurements is the largest multiple of  $t^{CK}$  that meets the maximum absolute value for  $t^{RAS}$ .
  7. Measurement is taken 500ms after entering into this operating mode to provide settling time for the tester.
  8.  $V_{DD}$  must not vary more than 4% if CKE is not active while any bank is active.
  9.  $I_{DD2N}$  specifies DQ, DQS, and DM to be driven to a valid high or low logic level.
  10. CKE must be active (HIGH) during the entire time a REFRESH command is executed. From the time the AUTO REFRESH command is registered, CKE must be active at each rising clock edge until  $t^{RFC}$  later.
  11. This limit is a nominal value and does not result in a fail. CKE is HIGH during REFRESH command period ( $t^{RFC}$  (MIN)) else CKE is LOW (for example, during standby).
  12. Values for  $I_{DD6}$  85°C are guaranteed for the entire temperature range. All other  $I_{DD6}$  values are estimated.
  13. Typical values at 25°C, not a maximum value.
  14. Self refresh is not supported for AT (85°C to 105°C) operation.

Figure 104: Typical Self Refresh Current vs. Temperature





## Electrical Specifications – AC Operating Conditions

**Table 46: Electrical Characteristics and Recommended AC Operating Conditions**

Notes 1–9 apply to all the parameters in this table;  $V_{DD}/V_{DDO} = 1.70\text{--}1.95V$

Parameter	Symbol	-5		-54		-6		-75		Unit	Notes
		Min	Max	Min	Max	Min	Max	Min	Max		
Access window of DQ from CK/CK#	t <sub>AC</sub>	2.0	5.0	2.0	5.0	2.0	5.0	2.0	6.0	ns	
		2.0	6.5	2.0	6.5	2.0	6.5	2.0	6.5		
Clock cycle time	t <sub>CK</sub>	5.0	–	5.4	–	6	–	7.5	–	ns	10
		12	–	12	–	12	–	12	–		
CK high-level width	t <sub>CH</sub>	0.45	0.55	0.45	0.55	0.45	0.55	0.45	0.55	t <sub>CK</sub>	
CK low-level width	t <sub>CL</sub>	0.45	0.55	0.45	0.55	0.45	0.55	0.45	0.55	t <sub>CK</sub>	
CKE minimum pulse width (high and low)	t <sub>CKE</sub>	1	–	1	–	1	–	1	–	t <sub>CK</sub>	11
Auto precharge write recovery + precharge time	t <sub>DAL</sub>	–	–	–	–	–	–	–	–	–	12
DQ and DM input hold time relative to DQS (fast slew rate)	t <sub>DH<sub>f</sub></sub>	0.48	–	0.54	–	0.6	–	0.8	–	ns	13, 14, 15
DQ and DM input hold time relative to DQS (slow slew rate)	t <sub>DH<sub>s</sub></sub>	0.58	–	0.64	–	0.7	–	0.9	–	ns	
DQ and DM input setup time relative to DQS (fast slew rate)	t <sub>DS<sub>f</sub></sub>	0.48	–	0.54	–	0.6	–	0.8	–	ns	13, 14, 15
DQ and DM input setup time relative to DQS (slow slew rate)	t <sub>DS<sub>s</sub></sub>	0.58	–	0.64	–	0.7	–	0.9	–	ns	
DQ and DM input pulse width (for each input)	t <sub>DIPW</sub>	1.8	–	1.9	–	2.1	–	1.8	–	ns	16
Access window of DQS from CK/CK#	t <sub>DQSK</sub>	2.0	5.0	2.0	5.0	2.0	5.0	2.0	6.0	ns	
		2.0	6.5	2.0	6.5	2.0	6.5	2.0	6.5	ns	
DQS input high pulse width	t <sub>DQSH</sub>	0.4	0.6	0.4	0.6	0.4	0.6	0.4	0.6	t <sub>CK</sub>	
DQS input low pulse width	t <sub>DQSL</sub>	0.4	0.6	0.4	0.6	0.4	0.6	0.4	0.6	t <sub>CK</sub>	
DQS–DQ skew, DQS to last DQ valid, per group, per access	t <sub>DQSQ</sub>	–	0.4	–	0.45	–	0.45	–	0.6	ns	13, 17
WRITE command to first DQS latching transition	t <sub>DQSS</sub>	0.75	1.25	0.75	1.25	0.75	1.25	0.75	1.25	t <sub>CK</sub>	
DQS falling edge from CK rising – hold time	t <sub>DSH</sub>	0.2	–	0.2	–	0.2	–	0.2	–	t <sub>CK</sub>	
DQS falling edge to CK rising – setup time	t <sub>DSS</sub>	0.2	–	0.2	–	0.2	–	0.2	–	t <sub>CK</sub>	

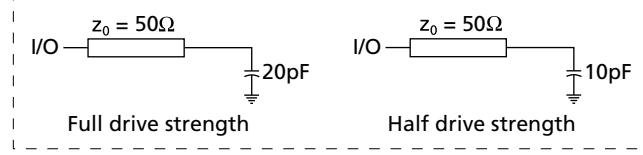
**Table 46: Electrical Characteristics and Recommended AC Operating Conditions (Continued)**Notes 1–9 apply to all the parameters in this table;  $V_{DD}/V_{DDQ} = 1.70\text{--}1.95V$ 

Parameter	Symbol	-5		-54		-6		-75		Unit	Notes
		Min	Max	Min	Max	Min	Max	Min	Max		
Data valid output window (DVW)	n/a	$t_{QH}$ - $t_{DQSQ}$		ns	17						
Half-clock period	$t_{HP}$	$t_{CH}$ , $t_{CL}$	-	ns	18						
Data-out High-Z window from CK/ CK#	$t_{HZ}$	-	5.0	-	5.0	-	5.0	-	6.0	ns	19, 20
		-	6.5	-	6.5	-	6.5	-	6.5	ns	
Data-out Low-Z window from CK/CK#	$t_{LZ}$	1.0	-	1.0	-	1.0	-	1.0	-	ns	19
Address and control input hold time (fast slew rate)	$t_{IH_F}$	0.9	-	1.0	-	1.1	-	1.3	-	ns	15, 21
Address and control input hold time (slow slew rate)	$t_{IH_S}$	1.1	-	1.2	-	1.3	-	1.5	-	ns	
Address and control input setup time (fast slew rate)	$t_{IS_F}$	0.9	-	1.0	-	1.1	-	1.3	-	ns	15, 21
Address and control input setup time (slow slew rate)	$t_{IS_S}$	1.1	-	1.2	-	1.3	-	1.5	-	ns	
Address and control input pulse width	$t_{IPW}$	2.3	-	2.5	-	2.6	-	$t_{IS} + t_{IH}$	-	ns	16
LOAD MODE REGISTER command cycle time	$t_{MRD}$	2	-	2	-	2	-	2	-	$t_{CK}$	
DQ-DQS hold, DQS to first DQ to go nonvalid, per access	$t_{QH}$	$t_{HP} - t_{QHS}$	-	ns	13, 17						
Data hold skew factor	$t_{QHS}$	-	0.5	-	0.5	-	0.65	-	0.75	ns	
ACTIVE-to-PRECHARGE command	$t_{RAS}$	40	70,000	42	70,000	42	70,000	45	70,000	ns	22
ACTIVE to ACTIVE/ACTIVE to AUTO REFRESH command period	$t_{RC}$	55	-	58.2	-	60	-	67.5	-	ns	
Active to read or write delay	$t_{RCD}$	15	-	16.2	-	18	-	22.5	-	ns	
Refresh period	$t_{REF}$	-	64	-	64	-	64	-	64	ms	28
Average periodic refresh interval: 64Mb, 128Mb, and 256Mb (x32)	$t_{REFI}$	-	15.6	-	15.6	-	15.6	-	15.6	μs	28
Average periodic refresh interval: 256Mb, 512Mb, 1Gb, 2Gb	$t_{REFI}$	-	7.8	-	7.8	-	7.8	-	7.8	μs	28
AUTO REFRESH command period	$t_{RFC}$	72	-	72	-	72	-	72	-	ns	

**Table 46: Electrical Characteristics and Recommended AC Operating Conditions (Continued)**Notes 1–9 apply to all the parameters in this table;  $V_{DD}/V_{DDQ} = 1.70\text{--}1.95V$ 

Parameter	Symbol	-5		-54		-6		-75		Unit	Notes
		Min	Max	Min	Max	Min	Max	Min	Max		
PRECHARGE command period	$t_{RP}$	15	–	16.2	–	18	–	22.5	–	ns	
DQS read preamble	CL = 3	$t_{RPRE}$	0.9	1.1	0.9	1.1	0.9	1.1	0.9	1.1	$t_{CK}$
	CL = 2	$t_{RPRE}$	0.5	1.1	0.5	1.1	0.5	1.1	0.5	1.1	$t_{CK}$
DQS read postamble	$t_{RPST}$	0.4	0.6	0.4	0.6	0.4	0.6	0.4	0.6	$t_{CK}$	
Active bank <i>a</i> to active bank <i>b</i> command	$t_{RRD}$	10	–	10.8	–	12	–	15	–	ns	
Read of SRR to next valid command	$t_{SRC}$	CL + 1	–	CL + 1	–	CL + 1	–	CL + 1	–	$t_{CK}$	
SRR to read	$t_{SRR}$	2	–	2	–	2	–	2	–	$t_{CK}$	
DQS write preamble	$t_{WPRE}$	0.25	–	0.25	–	0.25	–	0.25	–	$t_{CK}$	
DQS write preamble setup time	$t_{WPRES}$	0	–	0	–	0	–	0	–	ns	23, 24
DQS write postamble	$t_{WPST}$	0.4	0.6	0.4	0.6	0.4	0.6	0.4	0.6	$t_{CK}$	25
Write recovery time	$t_{WR}$	15	–	15	–	15	–	15	–	ns	26
Internal WRITE-to-READ command delay	$t_{WTR}$	2	–	2	–	1	–	1	–	$t_{CK}$	
Exit power-down mode to first valid command	$t_{XP}$	2	–	2	–	1	–	1	–	$t_{CK}$	
Exit self refresh to first valid command	$t_{XSR}$	112.5	–	112.5	–	112.5	–	112.5	–	ns	27

- Notes:
1. All voltages referenced to  $V_{SS}$ .
  2. All parameters assume proper device initialization.
  3. Tests for AC timing and electrical AC and DC characteristics may be conducted at nominal supply voltage levels, but the related specifications and device operation are guaranteed for the full voltage ranges specified.
  4. The circuit shown below represents the timing reference load used in defining the relevant timing parameters of the device. It is not intended to be either a precise representation of the typical system environment or a depiction of the actual load presented by a production tester. System designers will use IBIS or other simulation tools to correlate the timing reference load to system environment. Specifications are correlated to production test conditions (generally a coaxial transmission line terminated at the tester electronics). For the half-strength driver with a nominal 10pF load, parameters  $t_{AC}$  and  $t_{QH}$  are expected to be in the same range. However, these parameters are not subject to production test but are estimated by design/characterization. Use of IBIS or other simulation tools for system design validation is suggested.





## 168-Ball NAND Flash and LPDDR PoP (TI OMAP) MCP Electrical Specifications – AC Operating Conditions

5. The CK/CK# input reference voltage level (for timing referenced to CK/CK#) is the point at which CK and CK# cross; the input reference voltage level for signals other than CK/CK# is  $V_{DDQ}/2$ .
6. A CK and CK# input slew rate  $\geq 1$  V/ns (2 V/ns if measured differentially) is assumed for all parameters.
7. All AC timings assume an input slew rate of 1 V/ns.
8. CAS latency definition: with CL = 2, the first data element is valid at ( $t^{CK} + t^{AC}$ ) after the clock at which the READ command was registered; for CL = 3, the first data element is valid at ( $2 \times t^{CK} + t^{AC}$ ) after the first clock at which the READ command was registered.
9. Timing tests may use a  $V_{IL}$ -to- $V_{IH}$  swing of up to 1.5V in the test environment, but input timing is still referenced to  $V_{DDQ}/2$  or to the crossing point for CK/CK#. The output timing reference voltage level is  $V_{DDQ}/2$ .
10. Clock frequency change is only permitted during clock stop, power-down, or self refresh mode.
11. In cases where the device is in self refresh mode for  $t^{CKE}$ ,  $t^{CKE}$  starts at the rising edge of the clock and ends when CKE transitions HIGH.
12.  $t^{DAL} = (t^{WR}/t^{CK}) + (t^{RP}/t^{CK})$ : for each term, if not already an integer, round up to the next highest integer.
13. Referenced to each output group: for x16, LDQS with DQ[7:0]; and UDQS with DQ[15:8]. For x32, DQS0 with DQ[7:0]; DQS1 with DQ[15:8]; DQS2 with DQ[23:16]; and DQS3 with DQ[31:24].
14. DQ and DM input slew rates must not deviate from DQS by more than 10%. If the DQ/DM DQS slew rate is less than 1.0 V/ns, timing must be derated: 50ps must be added to  $t^{DS}$  and  $t^{DH}$  for each 100 mV/ns reduction in slew rate. If the slew rate exceeds 4 V/ns, functionality is uncertain.
15. The transition time for input signals (CAS#, CKE, CS#, DM, DQ, DQS, RAS#, WE#, and addresses) are measured between  $V_{IL(DC)}$  to  $V_{IH(AC)}$  for rising input signals and  $V_{IH(DC)}$  to  $V_{IL(AC)}$  for falling input signals.
16. These parameters guarantee device timing but are not tested on each device.
17. The valid data window is derived by achieving other specifications:  $t^{HP}$  ( $t^{CK}/2$ ),  $t^{DQSQ}$ , and  $t^{QH}$  ( $t^{HP} - t^{QHS}$ ). The data valid window derates directly proportional with the clock duty cycle and a practical data valid window can be derived. The clock is provided a maximum duty cycle variation of 45/55. Functionality is uncertain when operating beyond a 45/55 ratio.
18.  $t^{HP}$  (MIN) is the lesser of  $t^{CL}$  (MIN) and  $t^{CH}$  (MIN) actually applied to the device CK and CK# inputs, collectively.
19.  $t^{HZ}$  and  $t^{LZ}$  transitions occur in the same access time windows as valid data transitions. These parameters are not referenced to a specific voltage level, but specify when the device output is no longer driving ( $t^{HZ}$ ) or begins driving ( $t^{LZ}$ ).
20.  $t^{HZ}$  (MAX) will prevail over  $t^{DQSCK}$  (MAX) +  $t^{RPST}$  (MAX) condition.
21. Fast command/address input slew rate  $\geq 1$  V/ns. Slow command/address input slew rate  $\geq 0.5$  V/ns. If the slew rate is less than 0.5 V/ns, timing must be derated:  $t^{IS}$  has an additional 50ps per each 100 mV/ns reduction in slew rate from the 0.5 V/ns.  $t^{IH}$  has 0ps added, therefore, it remains constant. If the slew rate exceeds 4.5 V/ns, functionality is uncertain.
22. READs and WRITEs with auto precharge must not be issued until  $t^{RAS}$  (MIN) can be satisfied prior to the internal PRECHARGE command being issued.
23. This is not a device limit. The device will operate with a negative value, but system performance could be degraded due to bus turnaround.
24. It is recommended that DQS be valid (HIGH or LOW) on or before the WRITE command. The case shown (DQS going from High-Z to logic low) applies when no WRITEs were previously in progress on the bus. If a previous WRITE was in progress, DQS could be HIGH during this time, depending on  $t^{DQSS}$ .



## 168-Ball NAND Flash and LPDDR PoP (TI OMAP) MCP Electrical Specifications – AC Operating Conditions

25. The maximum limit for this parameter is not a device limit. The device will operate with a greater value for this parameter, but system performance (bus turnaround) will degrade accordingly.
26. At least 1 clock cycle is required during  $t_{WR}$  time when in auto precharge mode.
27. Clock must be toggled a minimum of two times during the  $t_{XSR}$  period.
28. For the Automotive Temperature parts,  $t_{REF} = t_{REF}/2$  and  $t_{REF\ I} = t_{REF\ I}/2$ .



## Output Drive Characteristics

**Table 47: Target Output Drive Characteristics (Full Strength)**

Notes 1–2 apply to all values; characteristics are specified under best and worst process variations/conditions

<b>Voltage (V)</b>	<b>Pull-Down Current (mA)</b>		<b>Pull-Up Current (mA)</b>	
	<b>Min</b>	<b>Max</b>	<b>Min</b>	<b>Max</b>
0.00	0.00	0.00	0.00	0.00
0.10	2.80	18.53	-2.80	-18.53
0.20	5.60	26.80	-5.60	-26.80
0.30	8.40	32.80	-8.40	-32.80
0.40	11.20	37.05	-11.20	-37.05
0.50	14.00	40.00	-14.00	-40.00
0.60	16.80	42.50	-16.80	-42.50
0.70	19.60	44.57	-19.60	-44.57
0.80	22.40	46.50	-22.40	-46.50
0.85	23.80	47.48	-23.80	-47.48
0.90	23.80	48.50	-23.80	-48.50
0.95	23.80	49.40	-23.80	-49.40
1.00	23.80	50.05	-23.80	-50.05
1.10	23.80	51.35	-23.80	-51.35
1.20	23.80	52.65	-23.80	-52.65
1.30	23.80	53.95	-23.80	-53.95
1.40	23.80	55.25	-23.80	-55.25
1.50	23.80	56.55	-23.80	-56.55
1.60	23.80	57.85	-23.80	-57.85
1.70	23.80	59.15	-23.80	-59.15
1.80	—	60.45	—	-60.45
1.90	—	61.75	—	-61.75

- Notes:
1. Based on nominal impedance of  $25\Omega$  (full strength) at  $V_{DDQ}/2$ .
  2. The full variation in driver current from minimum to maximum, due to process, voltage, and temperature, will lie within the outer bounding lines of the I-V curves.

**Table 48: Target Output Drive Characteristics (Three-Quarter Strength)**

Notes 1–3 apply to all values; characteristics are specified under best and worst process variations/conditions

Voltage (V)	Pull-Down Current (mA)		Pull-Up Current (mA)	
	Min	Max	Min	Max
0.00	0.00	0.00	0.00	0.00
0.10	1.96	12.97	-1.96	-12.97
0.20	3.92	18.76	-3.92	-18.76
0.30	5.88	22.96	-5.88	-22.96
0.40	7.84	25.94	-7.84	-25.94
0.50	9.80	28.00	-9.80	-28.00
0.60	11.76	29.75	-11.76	-29.75
0.70	13.72	31.20	-13.72	-31.20
0.80	15.68	32.55	-15.68	-32.55
0.85	16.66	33.24	-16.66	-33.24
0.90	16.66	33.95	-16.66	-33.95
0.95	16.66	34.58	-16.66	-34.58
1.00	16.66	35.04	-16.66	-35.04
1.10	16.66	35.95	-16.66	-35.95
1.20	16.66	36.86	-16.66	-36.86
1.30	16.66	37.77	-16.66	-37.77
1.40	16.66	38.68	-16.66	-38.68
1.50	16.66	39.59	-16.66	-39.59
1.60	16.66	40.50	-16.66	-40.50
1.70	16.66	41.41	-16.66	-41.41
1.80	—	42.32	—	-42.32
1.90	—	43.23	—	-43.23

- Notes:
1. Based on nominal impedance of  $37\Omega$  (three-quarter drive strength) at  $V_{DDQ}/2$ .
  2. The full variation in driver current from minimum to maximum, due to process, voltage, and temperature, will lie within the outer bounding lines of the I-V curves.
  3. Contact factory for availability of three-quarter drive strength.

**Table 49: Target Output Drive Characteristics (One-Half Strength)**

Notes 1–3 apply to all values; characteristics are specified under best and worst process variations/conditions

Voltage (V)	Pull-Down Current (mA)		Pull-Up Current (mA)	
	Min	Max	Min	Max
0.00	0.00	0.00	0.00	0.00
0.10	1.27	8.42	-1.27	-8.42
0.20	2.55	12.30	-2.55	-12.30
0.30	3.82	14.95	-3.82	-14.95
0.40	5.09	16.84	-5.09	-16.84
0.50	6.36	18.20	-6.36	-18.20
0.60	7.64	19.30	-7.64	-19.30
0.70	8.91	20.30	-8.91	-20.30
0.80	10.16	21.20	-10.16	-21.20
0.85	10.80	21.60	-10.80	-21.60
0.90	10.80	22.00	-10.80	-22.00
0.95	10.80	22.45	-10.80	-22.45
1.00	10.80	22.73	-10.80	-22.73
1.10	10.80	23.21	-10.80	-23.21
1.20	10.80	23.67	-10.80	-23.67
1.30	10.80	24.14	-10.80	-24.14
1.40	10.80	24.61	-10.80	-24.61
1.50	10.80	25.08	-10.80	-25.08
1.60	10.80	25.54	-10.80	-25.54
1.70	10.80	26.01	-10.80	-26.01
1.80	—	26.48	—	-26.48
1.90	—	26.95	—	-26.95

- Notes:
1. Based on nominal impedance of  $55\Omega$  (one-half drive strength) at  $V_{DDQ}/2$ .
  2. The full variation in driver current from minimum to maximum, due to process, voltage, and temperature, will lie within the outer bounding lines of the I-V curves.
  3. The I-V curve for one-quarter drive strength is approximately 50% of one-half drive strength.



## Functional Description

The Mobile LPDDR SDRAM uses a double data rate architecture to achieve high-speed operation. The double data rate architecture is essentially a  $2n$ -prefetch architecture, with an interface designed to transfer two data words per clock cycle at the I/O. Single read or write access for the device consists of a single  $2n$ -bit-wide, one-clock-cycle data transfer at the internal DRAM core and two corresponding  $n$ -bit-wide, one-half-clock-cycle data transfers at the I/O.

A bidirectional data strobe (DQS) is transmitted externally, along with data, for use in data capture at the receiver. DQS is a strobe transmitted by the device during READs and by the memory controller during WRITEs. DQS is edge-aligned with data for READs and center-aligned with data for WRITEs. The x16 device has two data strobes, one for the lower byte and one for the upper byte; the x32 device has four data strobes, one per byte.

The LPDDR device operates from a differential clock (CK and CK#); the crossing of CK going HIGH and CK# going LOW will be referred to as the positive edge of CK. Commands (address and control signals) are registered at every positive edge of CK. Input data is registered on both edges of DQS, and output data is referenced to both edges of DQS, as well as to both edges of CK.

Read and write accesses to the device are burst-oriented; accesses start at a selected location and continue for a programmed number of locations in a programmed sequence. Accesses begin with the registration of an ACTIVE command, followed by a READ or WRITE command. The address bits registered coincident with the ACTIVE command are used to select the bank and row to be accessed. The address bits registered coincident with the READ or WRITE command are used to select the starting column location for the burst access.

The device provides for programmable READ or WRITE burst lengths of 2, 4, 8, or 16. An auto precharge function can be enabled to provide a self-timed row precharge that is initiated at the end of the burst access.

As with standard DDR SDRAM, the pipelined, multibank architecture of LPDDR supports concurrent operation, thereby providing high effective bandwidth by hiding row precharge and activation time.

An auto refresh mode is provided, along with a power-saving power-down mode. Deep power-down mode is offered to achieve maximum power reduction by eliminating the power of the memory array. Data will not be retained after the device enters deep power-down mode.

Two self refresh features, temperature-compensated self refresh (TCSR) and partial-array self refresh (PASR), offer additional power savings. TCSR is controlled by the automatic on-chip temperature sensor. PASR can be customized using the extended mode register settings. The two features can be combined to achieve even greater power savings.

The DLL that is typically used on standard DDR devices is not necessary on LPDDR devices. It has been omitted to save power.



## Commands

A quick reference for available commands is provided in Table 50 and Table 51 (page 157), followed by a written description of each command. Three additional truth tables (Table 52 (page 163), Table 53 (page 165), and Table 54 (page 167)) provide CKE commands and current/next state information.

**Table 50: Truth Table – Commands**

CKE is HIGH for all commands shown except SELF REFRESH and DEEP POWER-DOWN; all states and sequences not shown are reserved and/or illegal

Name (Function)	CS#	RAS#	CAS#	WE#	Address	Notes
DESELECT (NOP)	H	X	X	X	X	1
NO OPERATION (NOP)	L	H	H	H	X	1
ACTIVE (select bank and activate row)	L	L	H	H	Bank/row	2
READ (select bank and column, and start READ burst)	L	H	L	H	Bank/column	3
WRITE (select bank and column, and start WRITE burst)	L	H	L	L	Bank/column	3
BURST TERMINATE or DEEP POWER-DOWN (enter deep power-down mode)	L	H	H	L	X	4, 5
PRECHARGE (deactivate row in bank or banks)	L	L	H	L	Code	6
AUTO REFRESH (refresh all or single bank) or SELF REFRESH (enter self refresh mode)	L	L	L	H	X	7, 8
LOAD MODE REGISTER	L	L	L	L	Op-code	9

- Notes:
1. DESELECT and NOP are functionally interchangeable.
  2. BA0–BA1 provide bank address and A[0:I] provide row address (where I = the most significant address bit for each configuration).
  3. BA0–BA1 provide bank address; A[0:I] provide column address (where I = the most significant address bit for each configuration); A10 HIGH enables the auto precharge feature (nonpersistent); A10 LOW disables the auto precharge feature.
  4. Applies only to READ bursts with auto precharge disabled; this command is undefined and should not be used for READ bursts with auto precharge enabled and for WRITE bursts.
  5. This command is a BURST TERMINATE if CKE is HIGH and DEEP POWER-DOWN if CKE is LOW.
  6. A10 LOW: BA0–BA1 determine which bank is precharged.  
A10 HIGH: all banks are precharged and BA0–BA1 are “Don’t Care.”
  7. This command is AUTO REFRESH if CKE is HIGH, SELF REFRESH if CKE is LOW.
  8. Internal refresh counter controls row addressing; in self refresh mode all inputs and I/Os are “Don’t Care” except for CKE.
  9. BA0–BA1 select the standard mode register, extended mode register, or status register.

**Table 51: DM Operation Truth Table**

Name (Function)	DM	DQ	Notes
Write enable	L	Valid	1, 2
Write inhibit	H	X	1, 2

Notes: 1. Used to mask write data; provided coincident with the corresponding data.  
 2. All states and sequences not shown are reserved and/or illegal.

## DESELECT

The DESELECT function (CS# HIGH) prevents new commands from being executed by the device. Operations already in progress are not affected.

## NO OPERATION

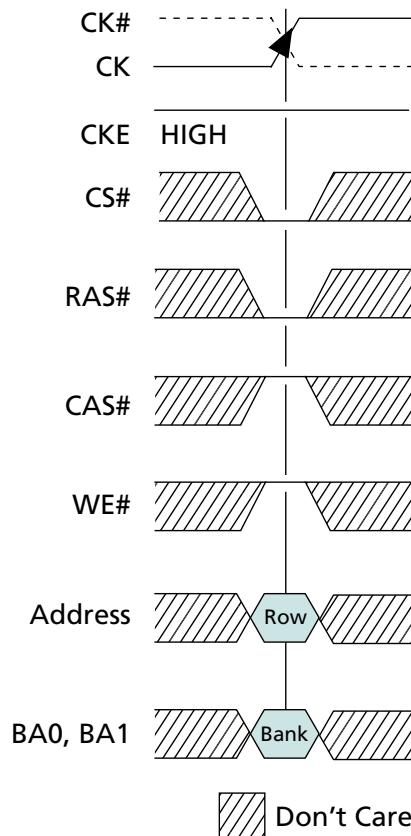
The NO OPERATION (NOP) command is used to instruct the selected device to perform a NOP. This prevents unwanted commands from being registered during idle or wait states. Operations already in progress are not affected.

## LOAD MODE REGISTER

The mode registers are loaded via inputs A[0:n]. See mode register descriptions in Standard Mode Register (page 172) and Extended Mode Register (page 176). The LOAD MODE REGISTER command can only be issued when all banks are idle, and a subsequent executable command cannot be issued until  $t_{MRD}$  is met.

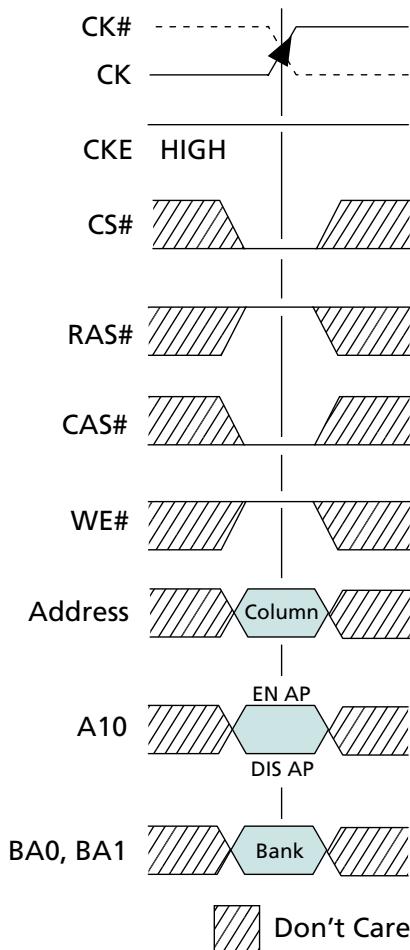
## ACTIVE

The ACTIVE command is used to activate a row in a particular bank for a subsequent access. The values on the BA0 and BA1 inputs select the bank, and the address provided on inputs A[0:n] selects the row. This row remains active for accesses until a PRECHARGE command is issued to that bank. A PRECHARGE command must be issued before opening a different row in the same bank.

**Figure 105: ACTIVE Command**

## READ

The READ command is used to initiate a burst read access to an active row. The values on the BA0 and BA1 inputs select the bank; the address provided on inputs A[ $I:0$ ] (where  $I$  = the most significant column address bit for each configuration) selects the starting column location. The value on input A10 determines whether auto precharge is used. If auto precharge is selected, the row being accessed will be precharged at the end of the READ burst; if auto precharge is not selected, the row will remain open for subsequent accesses.

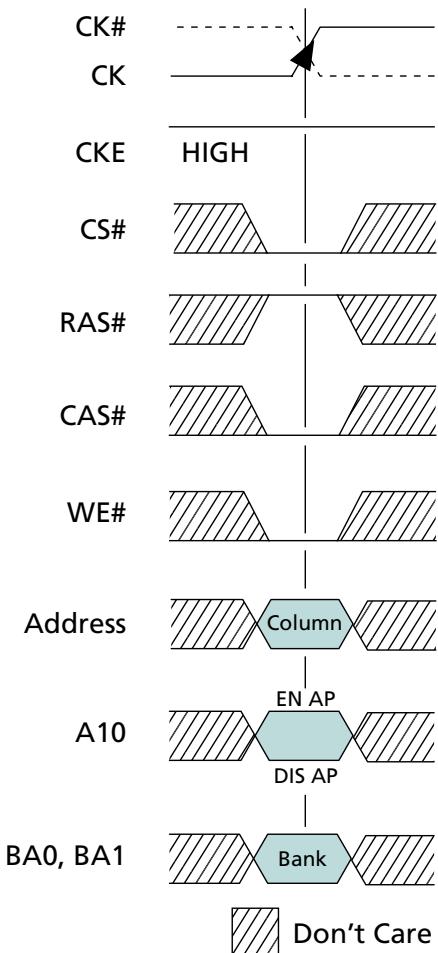
**Figure 106: READ Command**

Note: 1. EN AP = enable auto precharge; DIS AP = disable auto precharge.

## WRITE

The WRITE command is used to initiate a burst write access to an active row. The values on the BA0 and BA1 inputs select the bank; the address provided on inputs A<sub>I:0</sub> (where I = the most significant column address bit for each configuration) selects the starting column location. The value on input A10 determines whether auto precharge is used. If auto precharge is selected, the row being accessed will be precharged at the end of the WRITE burst; if auto precharge is not selected, the row will remain open for subsequent accesses. Input data appearing on the DQ is written to the memory array, subject to the DM input logic level appearing coincident with the data. If a given DM signal is registered LOW, the corresponding data will be written to memory; if the DM signal is registered HIGH, the corresponding data inputs will be ignored, and a WRITE will not be executed to that byte/column location.

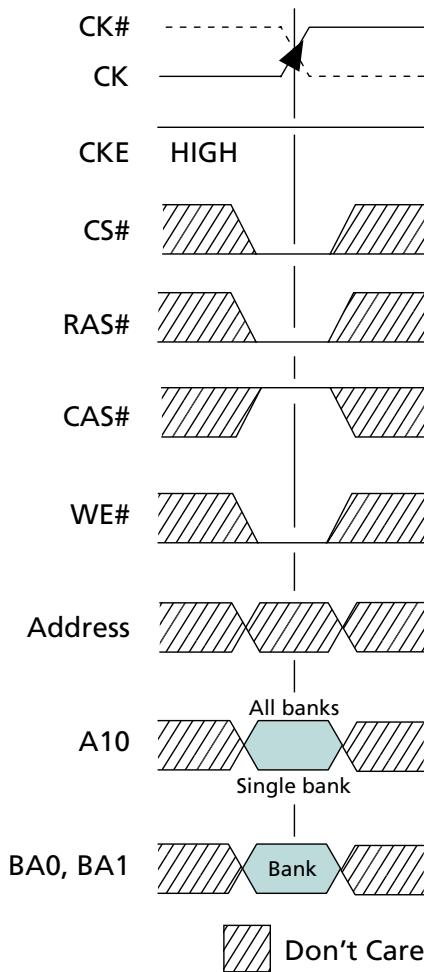
If a WRITE or a READ is in progress, the entire data burst must be complete prior to stopping the clock (see Clock Change Frequency (page 216)). A burst completion for WRITEs is defined when the write postamble and <sup>t</sup>WR or <sup>t</sup>WTR are satisfied.

**Figure 107: WRITE Command**

Note: 1. EN AP = enable auto precharge; DIS AP = disable auto precharge.

## PRECHARGE

The PRECHARGE command is used to deactivate the open row in a particular bank or the open row in all banks. The bank(s) will be available for a subsequent row access a specified time ( $t_{RP}$ ) after the PRECHARGE command is issued. Input A10 determines whether one or all banks will be precharged, and in the case where only one bank is precharged, inputs BA0 and BA1 select the bank. Otherwise, BA0 and BA1 are treated as "Don't Care." After a bank has been precharged, it is in the idle state and must be activated prior to any READ or WRITE commands being issued to that bank.

**Figure 108: PRECHARGE Command**

Note: 1. If A10 is HIGH, bank address becomes "Don't Care."

## BURST TERMINATE

The BURST TERMINATE command is used to truncate READ bursts with auto pre-charge disabled. The most recently registered READ command prior to the BURST TERMINATE command will be truncated, as described in READ Operation (page 181). The open page from which the READ was terminated remains open.

## AUTO REFRESH

AUTO REFRESH is used during normal operation of the device and is analogous to CAS#-BEFORE-RAS# (CBR) REFRESH in FPM/EDO DRAM. The AUTO REFRESH command is nonpersistent and must be issued each time a refresh is required.

Addressing is generated by the internal refresh controller. This makes the address bits a "Don't Care" during an AUTO REFRESH command.

For improved efficiency in scheduling and switching between tasks, some flexibility in the absolute refresh interval is provided. The auto refresh period begins when the AUTO REFRESH command is registered and ends 'RFC later.

## SELF REFRESH

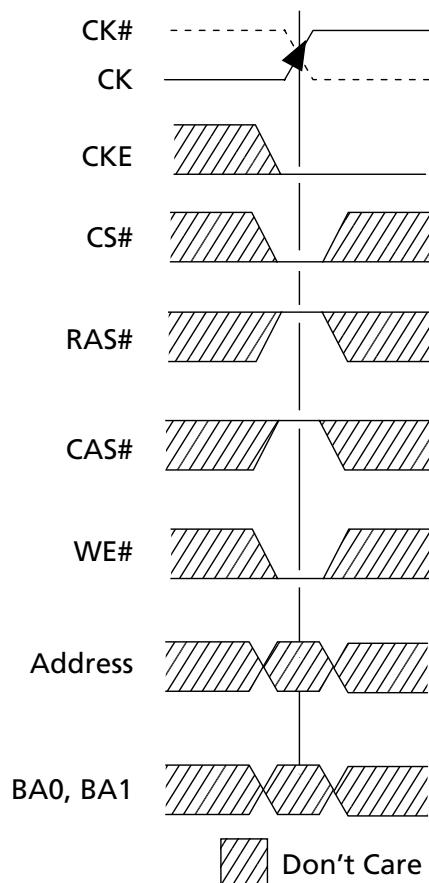
The SELF REFRESH command is used to place the device in self refresh mode; self refresh mode is used to retain data in the memory device while the rest of the system is powered down. When in self refresh mode, the device retains data without external clocking. The SELF REFRESH command is initiated like an AUTO REFRESH command, except that CKE is disabled (LOW). After the SELF REFRESH command is registered, all inputs to the device become “Don’t Care” with the exception of CKE, which must remain LOW.

Micron recommends that, prior to self refresh entry and immediately upon self refresh exit, the user perform a burst auto refresh cycle for the number of refresh rows. Alternatively, if a distributed refresh pattern is used, this pattern should be immediately resumed upon self refresh exit.

## DEEP POWER-DOWN

The DEEP POWER-DOWN (DPD) command is used to enter DPD mode, which achieves maximum power reduction by eliminating the power to the memory array. Data will not be retained when the device enters DPD mode. The DPD command is the same as a BURST TERMINATE command with CKE LOW.

**Figure 109: DEEP POWER-DOWN Command**





## Truth Tables

**Table 52: Truth Table – Current State Bank  $n$  – Command to Bank  $n$**

Notes 1–6 apply to all parameters in this table

Current State	CS#	RAS#	CAS#	WE#	Command/Action	Notes
Any	H	X	X	X	DESELECT (NOP/continue previous operation)	
	L	H	H	H	NO OPERATION (NOP/continue previous operation)	
Idle	L	L	H	H	ACTIVE (select and activate row)	
	L	L	L	H	AUTO REFRESH	7
	L	L	L	L	LOAD MODE REGISTER	7
Row active	L	H	L	H	READ (select column and start READ burst)	10
	L	H	L	L	WRITE (select column and start WRITE burst)	10
	L	L	H	L	PRECHARGE (deactivate row in bank or banks)	8
Read (auto pre-charge disabled)	L	H	L	H	READ (select column and start new READ burst)	10
	L	H	L	L	WRITE (select column and start WRITE burst)	10, 12
	L	L	H	L	PRECHARGE (truncate READ burst, start PRECHARGE)	8
	L	H	H	L	BURST TERMINATE	9
Write (auto pre-charge disabled)	L	H	L	H	READ (select column and start READ burst)	10, 11
	L	H	L	L	WRITE (select column and start new WRITE burst)	10
	L	L	H	L	PRECHARGE (truncate WRITE burst, start PRECHARGE)	8, 11

Notes: 1. This table applies when  $CKE_{n-1}$  was HIGH,  $CKE_n$  is HIGH and after  $tXSR$  has been met (if the previous state was self refresh), after  $tXP$  has been met (if the previous state was power-down), or after a full initialization (if the previous state was deep power-down).

2. This table is bank-specific, except where noted (for example, the current state is for a specific bank and the commands shown are supported for that bank when in that state). Exceptions are covered in the notes below.

3. Current state definitions:

Idle: The bank has been precharged, and  $tRP$  has been met.

Row active: A row in the bank has been activated, and  $tRCD$  has been met. No data bursts/ accesses and no register accesses are in progress.

Read: A READ burst has been initiated with auto precharge disabled and has not yet terminated or been terminated.

Write: A WRITE burst has been initiated with auto precharge disabled and has not yet terminated or been terminated.

4. The states listed below must not be interrupted by a command issued to the same bank. COMMAND INHIBIT or NOP commands, or supported commands to the other bank, must be issued on any clock edge occurring during these states. Supported commands to any other bank are determined by that bank's current state.

Precharging: Starts with registration of a PRECHARGE command and ends when  $tRP$  is met. After  $tRP$  is met, the bank will be in the idle state.

Row activating: Starts with registration of an ACTIVE command and ends when  $tRCD$  is met. After  $tRCD$  is met, the bank will be in the row active state.



Read with auto-precharge enabled: Starts with registration of a READ command with auto precharge enabled and ends when  $t_{RP}$  has been met. After  $t_{RP}$  is met, the bank will be in the idle state.

Write with auto-precharge enabled: Starts with registration of a WRITE command with auto precharge enabled and ends when  $t_{RP}$  has been met. After  $t_{RP}$  is met, the bank will be in the idle state.

5. The states listed below must not be interrupted by any executable command; DESELECT or NOP commands must be applied on each positive clock edge during these states.

Refreshing: Starts with registration of an AUTO REFRESH command and ends when  $t_{RFC}$  is met. After  $t_{RFC}$  is met, the device will be in the all banks idle state.

Accessing mode register: Starts with registration of a LOAD MODE REGISTER command and ends when  $t_{MRD}$  has been met. After  $t_{MRD}$  is met, the device will be in the all banks idle state.

Precharging all: Starts with registration of a PRECHARGE ALL command and ends when  $t_{RP}$  is met. After  $t_{RP}$  is met, all banks will be in the idle state.

6. All states and sequences not shown are illegal or reserved.
7. Not bank-specific; requires that all banks are idle, and bursts are not in progress.
8. May or may not be bank-specific; if multiple banks need to be precharged, each must be in a valid state for precharging.
9. Not bank-specific; BURST TERMINATE affects the most recent READ burst, regardless of bank.
10. READs or WRITEs listed in the Command/Action column include READs or WRITEs with auto precharge enabled and READs or WRITEs with auto precharge disabled.
11. Requires appropriate DM masking.
12. A WRITE command can be applied after the completion of the READ burst; otherwise, a BURST TERMINATE must be used to end the READ burst prior to asserting a WRITE command.

**Table 53: Truth Table – Current State Bank *n* – Command to Bank *m***

Notes 1–6 apply to all parameters in this table

<b>Current State</b>	<b>CS#</b>	<b>RAS#</b>	<b>CAS#</b>	<b>WE#</b>	<b>Command/Action</b>	<b>Notes</b>
Any	H	X	X	X	DESELECT (NOP/continue previous operation)	
	L	H	H	H	NO OPERATION (NOP/continue previous operation)	
Idle	X	X	X	X	Any command supported to bank <i>m</i>	
Row activating, active, or pre-charging	L	L	H	H	ACTIVE (select and activate row)	
	L	H	L	H	READ (select column and start READ burst)	
	L	H	L	L	WRITE (select column and start WRITE burst)	
	L	L	H	L	PRECHARGE	
Read (auto pre-charge disabled)	L	L	H	H	ACTIVE (select and activate row)	
	L	H	L	H	READ (select column and start new READ burst)	
	L	H	L	L	WRITE (select column and start WRITE burst)	7
	L	L	H	L	PRECHARGE	
Write (auto pre-charge disabled)	L	L	H	H	ACTIVE (select and activate row)	
	L	H	L	H	READ (select column and start READ burst)	
	L	H	L	L	WRITE (select column and start new WRITE burst)	
	L	L	H	L	PRECHARGE	
Read (with auto precharge)	L	L	H	H	ACTIVE (select and activate row)	
	L	H	L	H	READ (select column and start new READ burst)	
	L	H	L	L	WRITE (select column and start WRITE burst)	7
	L	L	H	L	PRECHARGE	
Write (with auto precharge)	L	L	H	H	ACTIVE (select and activate row)	
	L	H	L	H	READ (select column and start READ burst)	
	L	H	L	L	WRITE (select column and start new WRITE burst)	
	L	L	H	L	PRECHARGE	

- Notes:
1. This table applies when  $\text{CKE}_{n-1}$  was HIGH,  $\text{CKE}_n$  is HIGH and after  $t_{XSR}$  has been met (if the previous state was self refresh), after  $t_{XP}$  has been met (if the previous state was power-down) or after a full initialization (if the previous state was deep power-down).
  2. This table describes alternate bank operation, except where noted (for example, the current state is for bank *n* and the commands shown are those supported for issue to bank *m*, assuming that bank *m* is in such a state that the given command is supported). Exceptions are covered in the notes below.
  3. Current state definitions:

Idle: The bank has been precharged, and  $t_{RP}$  has been met.

Row active: A row in the bank has been activated, and  $t_{RCD}$  has been met. No data bursts/ accesses and no register accesses are in progress.

Read: A READ burst has been initiated and has not yet terminated or been terminated.

Write: A WRITE burst has been initiated and has not yet terminated or been terminated.

3a. Both the read with auto precharge enabled state or the write with auto precharge enabled state can be broken into two parts: the access period and the precharge period. For read with auto precharge, the precharge period is defined as if the same burst was



## 168-Ball NAND Flash and LPDDR PoP (TI OMAP) MCP Truth Tables

executed with auto precharge disabled and then followed with the earliest possible PRECHARGE command that still accesses all of the data in the burst. For write with auto precharge, the precharge period begins when  $t_{WR}$  ends, with  $t_{WR}$  measured as if auto precharge was disabled. The access period starts with registration of the command and ends when the precharge period (or  $t_{RP}$ ) begins. This device supports concurrent auto precharge such that when a read with auto precharge is enabled or a write with auto precharge is enabled, any command to other banks is supported, as long as that command does not interrupt the read or write data transfer already in process. In either case, all other related limitations apply (i.e., contention between read data and write data must be avoided).

3b. The minimum delay from a READ or WRITE command (with auto precharge enabled) to a command to a different bank is summarized below.

<b>From Command</b>	<b>To Command</b>	<b>Minimum Delay (with Concurrent Auto Precharge)</b>
WRITE with Auto Precharge	READ or READ with auto precharge WRITE or WRITE with auto precharge PRECHARGE ACTIVE	$[1 + (BL/2)] t_{CK} + t_{WTR}$ $(BL/2) t_{CK}$ $1 t_{CK}$ $1 t_{CK}$
READ with Auto Precharge	READ or READ with auto precharge WRITE or WRITE with auto precharge PRECHARGE ACTIVE	$(BL/2) \times t_{CK}$ $[CL + (BL/2)] t_{CK}$ $1 t_{CK}$ $1 t_{CK}$

4. AUTO REFRESH and LOAD MODE REGISTER commands can only be issued when all banks are idle.
5. All states and sequences not shown are illegal or reserved.
6. Requires appropriate DM masking.
7. A WRITE command can be applied after the completion of the READ burst; otherwise, a BURST TERMINATE must be used to end the READ burst prior to asserting a WRITE command.

**Table 54: Truth Table – CKE**

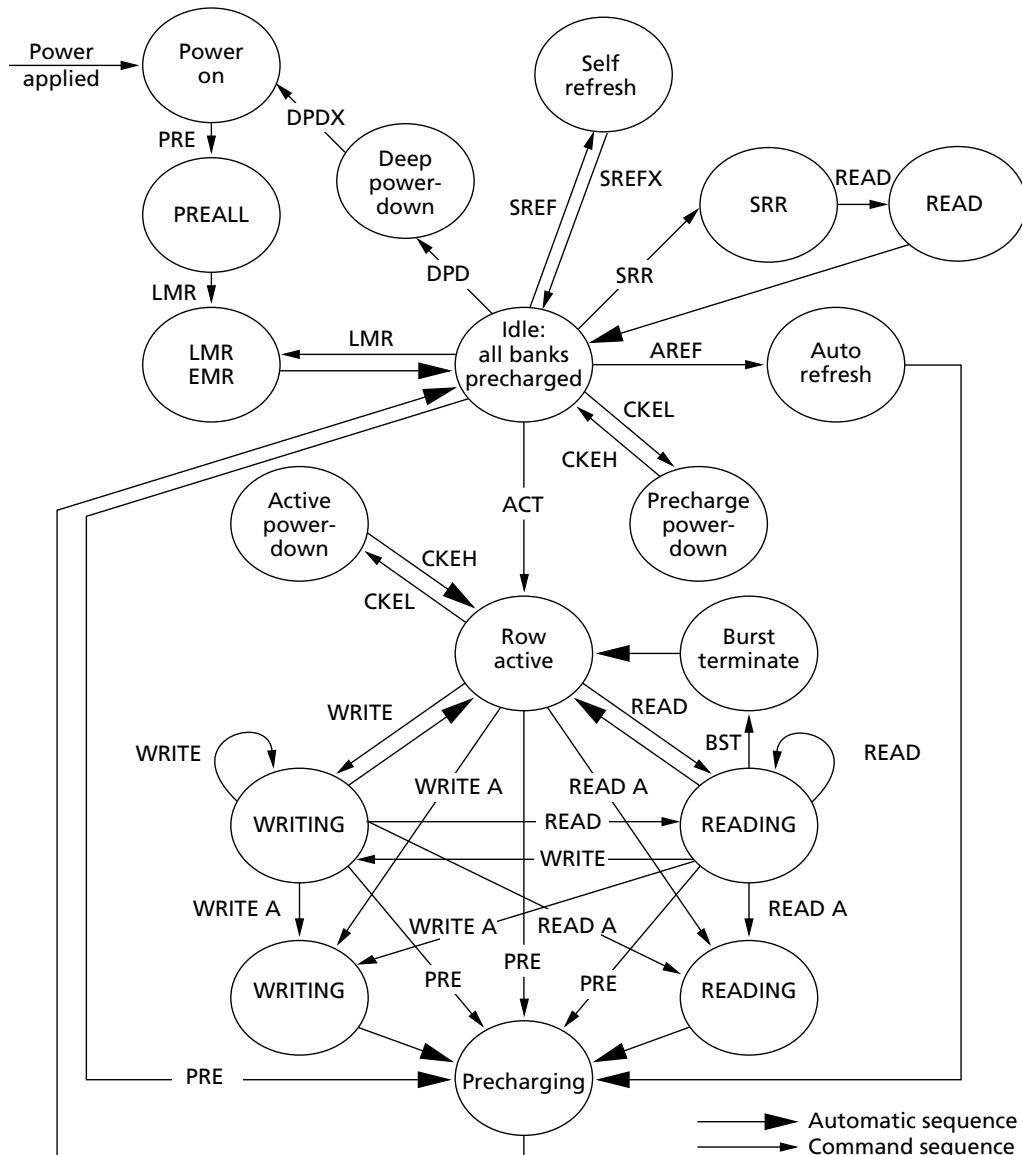
Notes 1–4 apply to all parameters in this table

<b>Current State</b>	<b>CKE<sub>n - 1</sub></b>	<b>CKE<sub>n</sub></b>	<b>COMMAND<sub>n</sub></b>	<b>ACTION<sub>n</sub></b>	<b>Notes</b>
Active power-down	L	L	X	Maintain active power-down	
Deep power-down	L	L	X	Maintain deep power-down	
Precharge power-down	L	L	X	Maintain precharge power-down	
Self refresh	L	L	X	Maintain self refresh	
Active power-down	L	H	DESELECT or NOP	Exit active power-down	5
Deep power-down	L	H	DESELECT or NOP	Exit deep power-down	6
Precharge power-down	L	H	DESELECT or NOP	Exit precharge power-down	
Self refresh	L	H	DESELECT or NOP	Exit self refresh	5, 7
Bank(s) active	H	L	DESELECT or NOP	Active power-down entry	
All banks idle	H	L	BURST TERMINATE	Deep power-down entry	
All banks idle	H	L	DESELECT or NOP	Precharge power-down entry	
All banks idle	H	L	AUTO REFRESH	Self refresh entry	
	H	H	See Table 53 (page 165)		
	H	H	See Table 53 (page 165)		

- Notes:
1. CKE<sub>n</sub> is the logic state of CKE at clock edge *n*; CKE<sub>n - 1</sub> was the state of CKE at the previous clock edge.
  2. Current state is the state of the DDR SDRAM immediately prior to clock edge *n*.
  3. COMMAND<sub>n</sub> is the command registered at clock edge *n*, and ACTION<sub>n</sub> is a result of COMMAND<sub>n</sub>.
  4. All states and sequences not shown are illegal or reserved.
  5. DESELECT or NOP commands should be issued on each clock edge occurring during the t<sub>XP</sub> or t<sub>XSR</sub> period.
  6. After exiting deep power-down mode, a full DRAM initialization sequence is required.
  7. The clock must toggle at least two times during the t<sub>XSR</sub> period.

## State Diagram

Figure 110: Simplified State Diagram



ACT = ACTIVE

AREF = AUTO REFRESH

BST = BURST TERMINATE

CKEH = Exit power-down

CKEL = Enter power-down

DPD = Enter deep power-down

DPDX = Exit deep power-down

EMR = LOAD EXTENDED MODE REGISTER

LMR = LOAD MODE REGISTER

PRE = PRECHARGE

PREALL = PRECHARGE all banks

READ = READ w/o auto precharge

READ A = READ w/ auto precharge

SREF = Enter self refresh

SREFX = Exit self refresh

SRR = STATUS REGISTER READ

WRITE = WRITE w/o auto precharge

WRITE A = WRITE w/ auto precharge



## Initialization

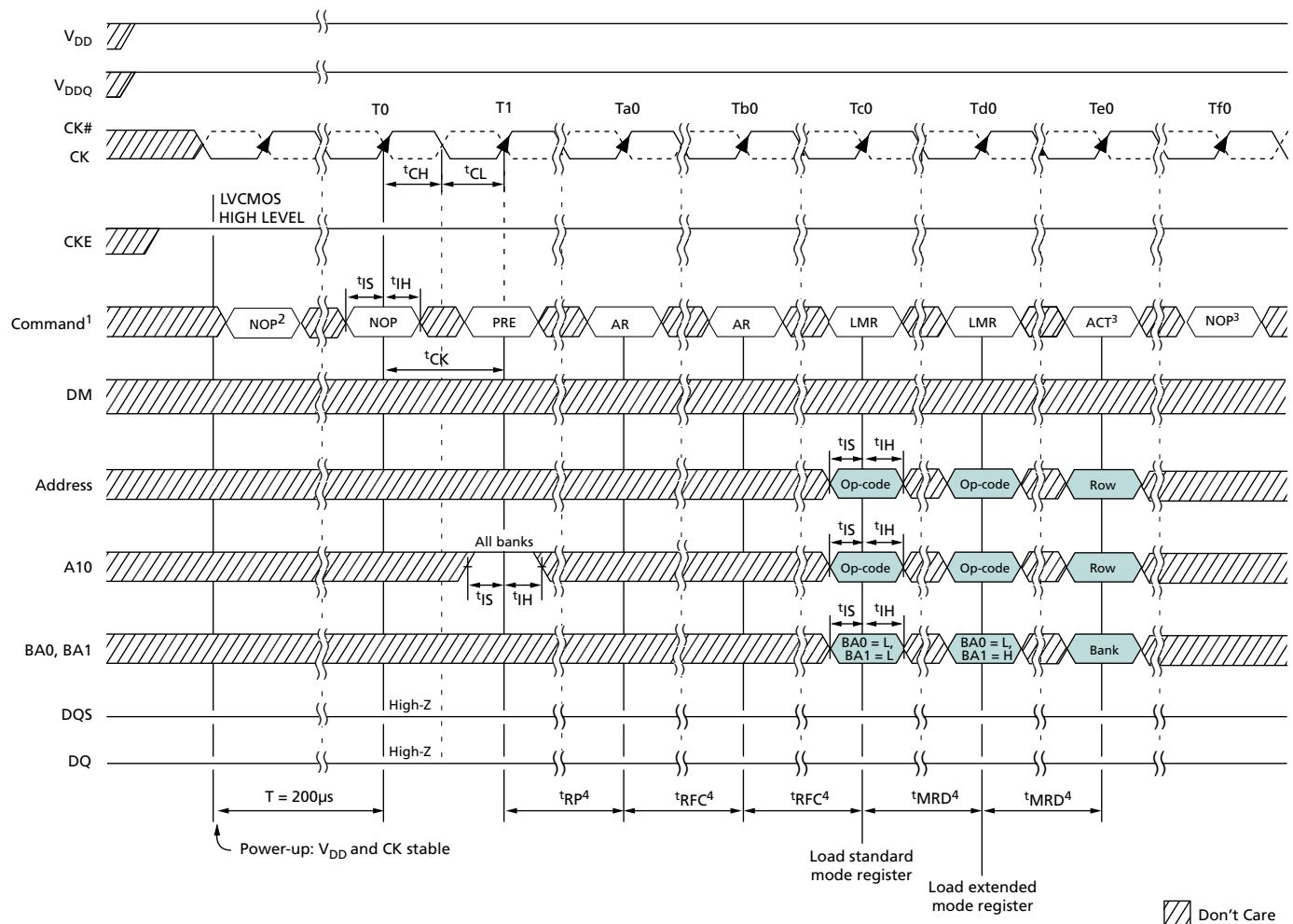
Prior to normal operation, the device must be powered up and initialized in a pre-defined manner. Using initialization procedures other than those specified will result in undefined operation.

If there is an interruption to the device power, the device must be re-initialized using the initialization sequence described below to ensure proper functionality of the device.

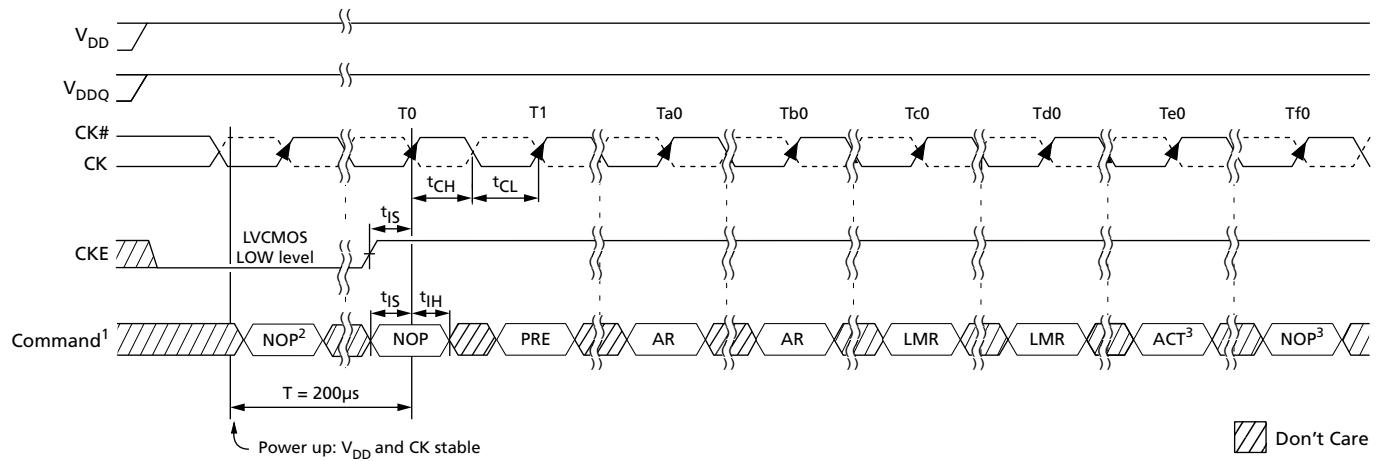
To properly initialize the device, this sequence must be followed:

1. The core power ( $V_{DD}$ ) and I/O power ( $V_{DDQ}$ ) must be brought up simultaneously. It is recommended that  $V_{DD}$  and  $V_{DDQ}$  be from the same power source, or  $V_{DDQ}$  must never exceed  $V_{DD}$ . Standard initialization requires that CKE be asserted HIGH (see Figure 111 (page 170)). Alternatively, initialization can be completed with CKE LOW provided that CKE transitions HIGH 'IS prior to T0 (see Figure 112 (page 171)).
2. When power supply voltages are stable and the CKE has been driven HIGH, it is safe to apply the clock.
3. When the clock is stable, a  $200\mu s$  minimum delay is required by the Mobile LPDDR prior to applying an executable command. During this time, NOP or DESELECT commands must be issued on the command bus.
4. Issue a PRECHARGE ALL command.
5. Issue NOP or DESELECT commands for at least  $t_{RP}$  time.
6. Issue an AUTO REFRESH command followed by NOP or DESELECT commands for at least  $t_{RFC}$  time. Issue a second AUTO REFRESH command followed by NOP or DESELECT commands for at least  $t_{RFC}$  time. Two AUTO REFRESH commands must be issued. Typically, both of these commands are issued at this stage as described above.
7. Using the LOAD MODE REGISTER command, load the standard mode register as desired.
8. Issue NOP or DESELECT commands for at least  $t_{MRD}$  time.
9. Using the LOAD MODE REGISTER command, load the extended mode register to the desired operating modes. Note that the sequence in which the standard and extended mode registers are programmed is not critical.
10. Issue NOP or DESELECT commands for at least  $t_{MRD}$  time.

After steps 1–10 are completed, the device has been properly initialized and is ready to receive any valid command.

**Figure 111: Initialize and Load Mode Registers**

- Notes:
1. PRE = PRECHARGE command; LMR = LOAD MODE REGISTER command; AR = AUTO REFRESH command; ACT = ACTIVE command.
  2. NOP or DESELECT commands are required for at least 200µs.
  3. Other valid commands are possible.
  4. NOPs or DESELECTs are required during this time.

**Figure 112: Alternate Initialization with CKE LOW**

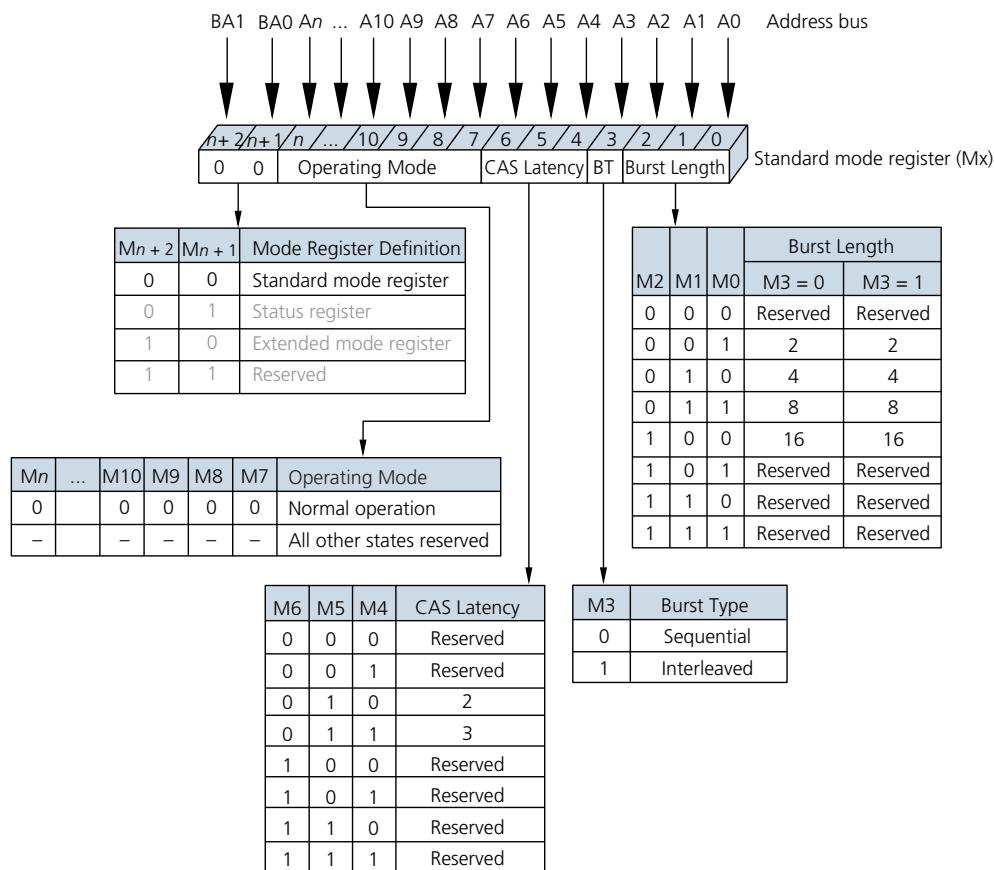
- Notes:
1. PRE = PRECHARGE command; LMR = LOAD MODE REGISTER command; AR = AUTO REFRESH command; ACT = ACTIVE command.
  2. NOP or DESELECT commands are required for at least 200μs.
  3. Other valid commands are possible.

## Standard Mode Register

The standard mode register bit definition enables the selection of burst length, burst type, CAS latency (CL), and operating mode, as shown in Figure 113. Reserved states should not be used as this may result in setting the device into an unknown state or cause incompatibility with future versions of LPDDR devices. The standard mode register is programmed via the LOAD MODE REGISTER command (with BA0 = 0 and BA1 = 0) and will retain the stored information until it is programmed again, until the device goes into deep power-down mode, or until the device loses power.

Reprogramming the mode register will not alter the contents of the memory, provided it is performed correctly. The mode register must be loaded when all banks are idle and no bursts are in progress, and the controller must wait  $t_{MRD}$  before initiating the subsequent operation. Violating any of these requirements will result in unspecified operation.

**Figure 113: Standard Mode Register Definition**



Note: 1. The integer n is equal to the most significant address bit.



## Burst Length

Read and write accesses to the device are burst-oriented, and the burst length (BL) is programmable. The burst length determines the maximum number of column locations that can be accessed for a given READ or WRITE command. Burst lengths of 2, 4, 8, or 16 locations are available for both sequential and interleaved burst types.

When a READ or WRITE command is issued, a block of columns equal to the burst length is effectively selected. All accesses for that burst take place within this block, meaning that the burst will wrap when a boundary is reached. The block is uniquely selected by  $A[i:1]$  when  $BL = 2$ , by  $A[i:2]$  when  $BL = 4$ , by  $A[i:3]$  when  $BL = 8$ , and by  $A[i:4]$  when  $BL = 16$ , where  $Ai$  is the most significant column address bit for a given configuration. The remaining (least significant) address bits are used to specify the starting location within the block. The programmed burst length applies to both READ and WRITE bursts.

## Burst Type

Accesses within a given burst can be programmed to be either sequential or interleaved via the standard mode register.

The ordering of accesses within a burst is determined by the burst length, the burst type, and the starting column address.

**Table 55: Burst Definition Table**

Burst Length	Starting Column Address			Order of Accesses Within a Burst	
				Type = Sequential	Type = Interleaved
<b>2</b>			<b>A0</b>		
			0	0-1	0-1
			1	1-0	1-0
<b>4</b>		<b>A1</b>	<b>A0</b>		
		0	0	0-1-2-3	0-1-2-3
		0	1	1-2-3-0	1-0-3-2
		1	0	2-3-0-1	2-3-0-1
		1	1	3-0-1-2	3-2-1-0
<b>8</b>	<b>A2</b>	<b>A1</b>	<b>A0</b>		
		0	0	0-1-2-3-4-5-6-7	0-1-2-3-4-5-6-7
		0	0	1-2-3-4-5-6-7-0	1-0-3-2-5-4-7-6
		0	1	2-3-4-5-6-7-0-1	2-3-0-1-6-7-4-5
		0	1	3-4-5-6-7-0-1-2	3-2-1-0-7-6-5-4
		1	0	4-5-6-7-0-1-2-3	4-5-6-7-0-1-2-3
		1	0	5-6-7-0-1-2-3-4	5-4-7-6-1-0-3-2
		1	1	6-7-0-1-2-3-4-5	6-7-4-5-2-3-0-1
		1	1	7-0-1-2-3-4-5-6	7-6-5-4-3-2-1-0
<b>16</b>	<b>A3</b>	<b>A2</b>	<b>A1</b>	<b>A0</b>	

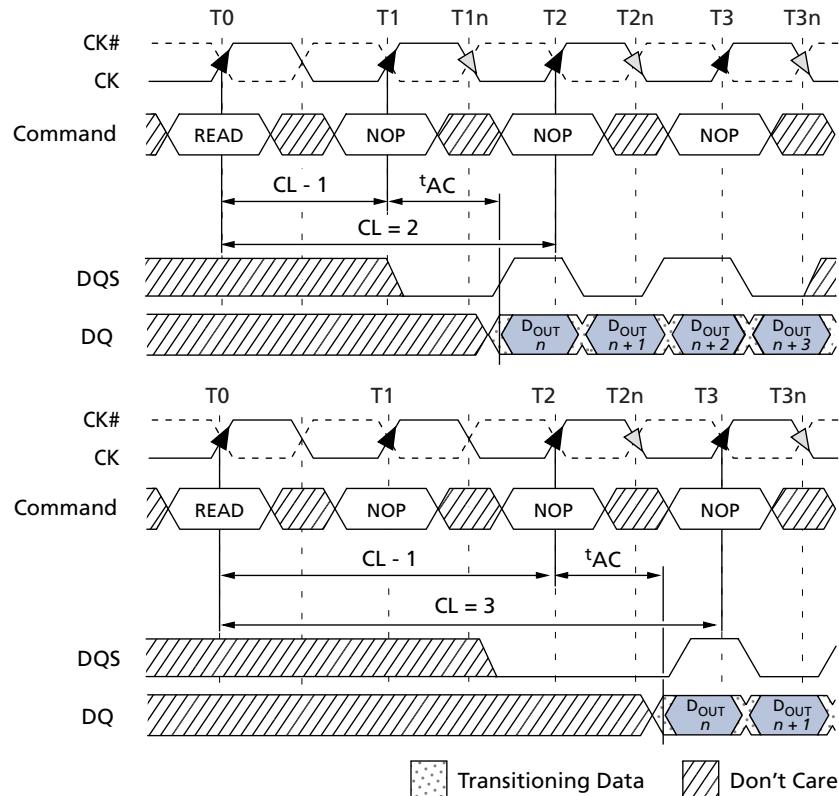
**Table 55: Burst Definition Table (Continued)**

Burst Length	Starting Column Address				Order of Accesses Within a Burst	
					Type = Sequential	
0	0	0	0	0	0-1-2-3-4-5-6-7-8-9-A-B-C-D-E-F	0-1-2-3-4-5-6-7-8-9-A-B-C-D-E-F
	0	0	0	1	1-2-3-4-5-6-7-8-9-A-B-C-D-E-F-0	1-0-3-2-5-4-7-6-9-8-B-A-D-C-F-E
	0	0	1	0	2-3-4-5-6-7-8-9-A-B-C-D-E-F-0-1	2-3-0-1-6-7-4-5-A-B-8-9-E-F-C-D
	0	0	1	1	3-4-5-6-7-8-9-A-B-C-D-E-F-0-1-2	3-2-1-0-7-6-5-4-B-A-9-8-F-E-D-C
	0	1	0	0	4-5-6-7-8-9-A-B-C-D-E-F-0-1-2-3	4-5-6-7-0-1-2-3-C-D-E-F-8-9-A-B
	0	1	0	1	5-6-7-8-9-A-B-C-D-E-F-0-1-2-3-4	5-4-7-6-1-0-3-2-D-C-F-E-9-8-B-A
	0	1	1	0	6-7-8-9-A-B-C-D-E-F-0-1-2-3-4-5	6-7-4-5-2-3-0-1-E-F-C-D-A-B-8-9
	0	1	1	1	7-8-9-A-B-C-D-E-F-0-1-2-3-4-5-6	7-6-5-4-3-2-1-0-F-E-D-C-B-A-9-8
	1	0	0	0	8-9-A-B-C-D-E-F-0-1-2-3-4-5-6-7	8-9-A-B-C-D-E-F-0-1-2-3-4-5-6-7
	1	0	0	1	9-A-B-C-D-E-F-0-1-2-3-4-5-6-7-8	9-8-B-A-D-C-F-E-1-0-3-2-5-4-7-6
	1	0	1	0	A-B-C-D-E-F-0-1-2-3-4-5-6-7-8-9	A-B-8-9-E-F-C-D-2-3-0-1-6-7-4-5
	1	0	1	1	B-C-D-E-F-0-1-2-3-4-5-6-7-8-9-A	B-A-9-8-F-E-D-C-3-2-1-0-7-6-5-4
	1	1	0	0	C-D-E-F-0-1-2-3-4-5-6-7-8-9-A-B	C-D-E-F-8-9-A-B-4-5-6-7-0-1-2-3
	1	1	0	1	D-E-F-0-1-2-3-4-5-6-7-8-9-A-B-C	D-C-F-E-9-8-B-A-5-4-7-6-1-0-3-2
	1	1	1	0	E-F-0-1-2-3-4-5-6-7-8-9-A-B-C-D	E-F-C-D-A-B-8-9-6-7-4-5-2-3-0-1
	1	1	1	1	F-0-1-2-3-4-5-6-7-8-9-A-B-C-D-E	F-E-D-C-B-A-9-8-7-6-5-4-3-2-1-0

## CAS Latency

The CAS latency (CL) is the delay, in clock cycles, between the registration of a READ command and the availability of the first output data. The latency can be set to 2 or 3 clocks, as shown in Figure 114 (page 175).

For CL = 3, if the READ command is registered at clock edge  $n$ , then the data will be nominally available at  $(n + 2 \text{ clocks} + t_{AC})$ . For CL = 2, if the READ command is registered at clock edge  $n$ , then the data will be nominally available at  $(n + 1 \text{ clock} + t_{AC})$ .

**Figure 114: CAS Latency**

## Operating Mode

The normal operating mode is selected by issuing a LOAD MODE REGISTER command with bits A[ $n:7$ ] each set to zero, and bits A[6:0] set to the desired values.

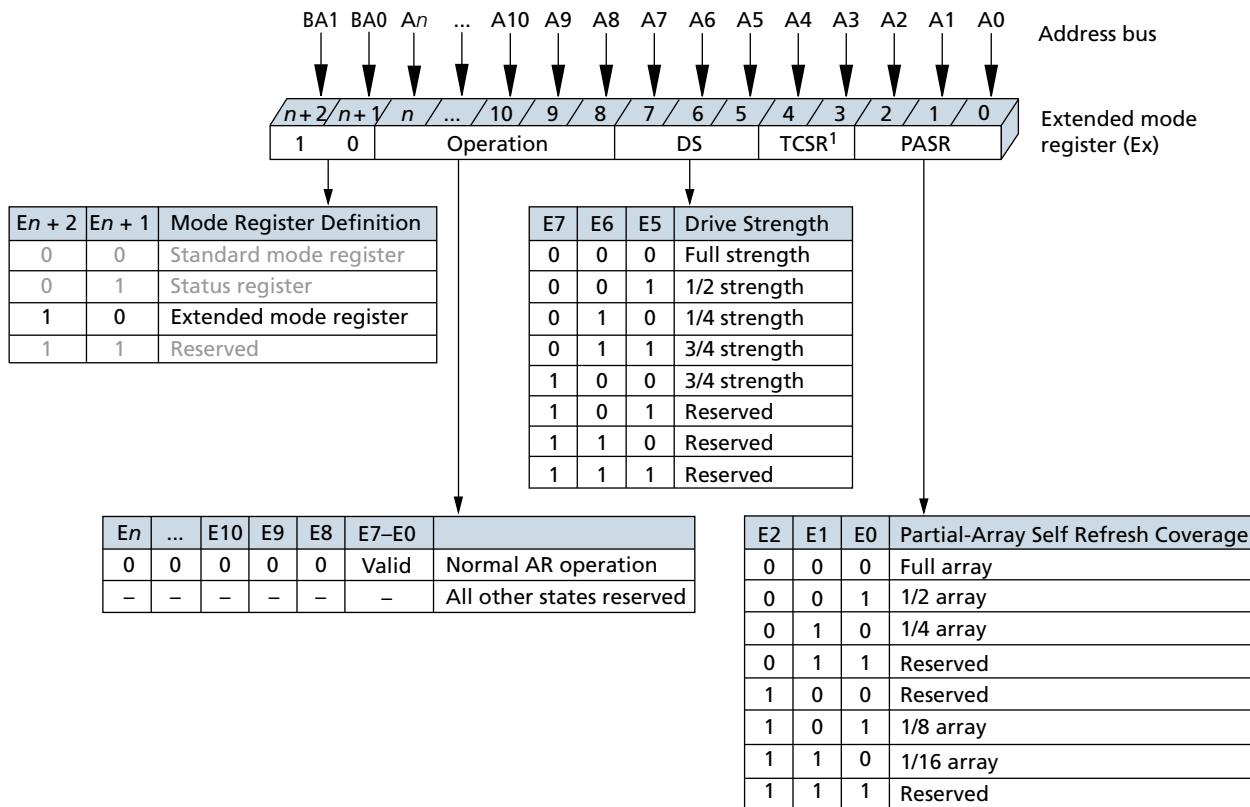
All other combinations of values for A[ $n:7$ ] are reserved for future use. Reserved states should not be used because unknown operation or incompatibility with future versions may result.

## Extended Mode Register

The EMR controls additional functions beyond those set by the mode registers. These additional functions include drive strength, TCSR, and PASR.

The EMR is programmed via the LOAD MODE REGISTER command with BA0 = 0 and BA1 = 1. Information in the EMR will be retained until it is programmed again, the device goes into deep power-down mode, or the device loses power.

**Figure 115: Extended Mode Register**



Notes:

- On-die temperature sensor is used in place of TCSR. Setting these bits will have no effect.
- The integer n is equal to the most significant address bit.

## Temperature-Compensated Self Refresh

This device includes a temperature sensor that is implemented for automatic control of the self refresh oscillator. Programming the temperature-compensated self refresh (TCSR) bits will have no effect on the device. The self refresh oscillator will continue to refresh at the optimal factory-programmed rate for the device temperature.



## Partial-Array Self Refresh

For further power savings during self refresh, the partial-array self refresh (PASR) feature enables the controller to select the amount of memory to be refreshed during self refresh. The refresh options include:

- Full array: banks 0, 1, 2, and 3
- One-half array: banks 0 and 1
- One-quarter array: bank 0
- One-eighth array: bank 0 with row address most significant bit (MSB) = 0
- One-sixteenth array: bank 0 with row address MSB = 0 and row address MSB - 1 = 0

READ and WRITE commands can still be issued to the full array during standard operation, but only the selected regions of the array will be refreshed during self refresh. Data in regions that are not selected will be lost.

## Output Drive Strength

Because the device is designed for use in smaller systems that are typically point-to-point connections, an option to control the drive strength of the output buffers is provided. Drive strength should be selected based on the expected loading of the memory bus. The output driver settings are  $25\Omega$ ,  $37\Omega$ , and  $55\Omega$  internal impedance for full, three-quarter, and one-half drive strengths, respectively.

## Status Read Register

The status read register (SRR) is used to read the manufacturer ID, revision ID, refresh multiplier, width type, and density of the device, as shown in Figure 117 (page 179).

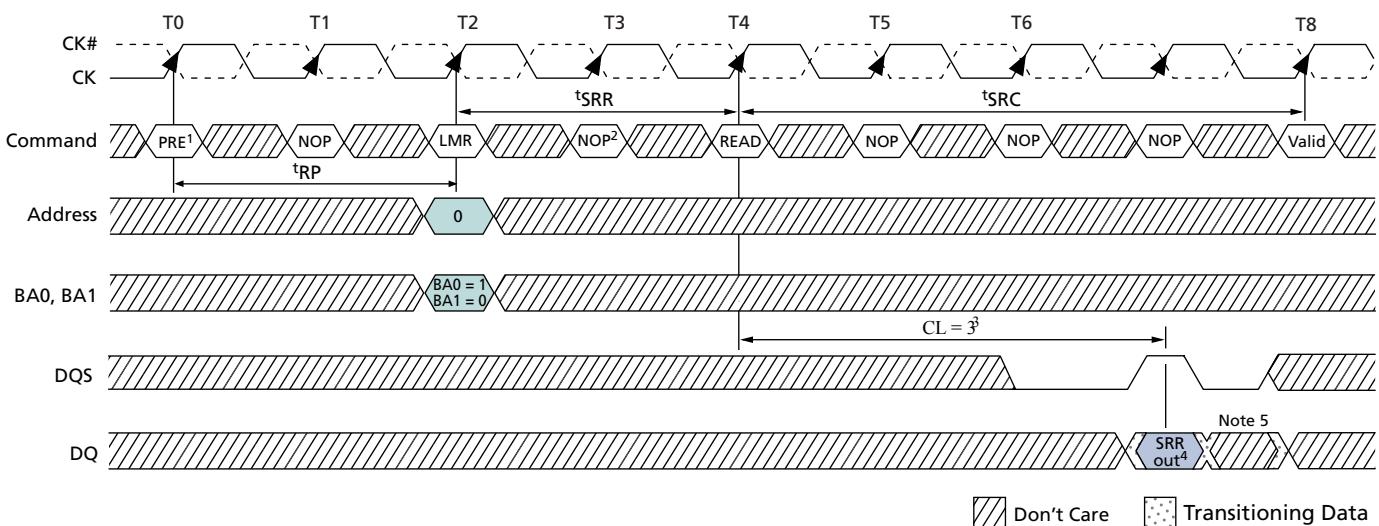
The SRR is read via the LOAD MODE REGISTER command with BA0 = 1 and BA1 = 0.

The sequence to perform an SRR command is as follows:

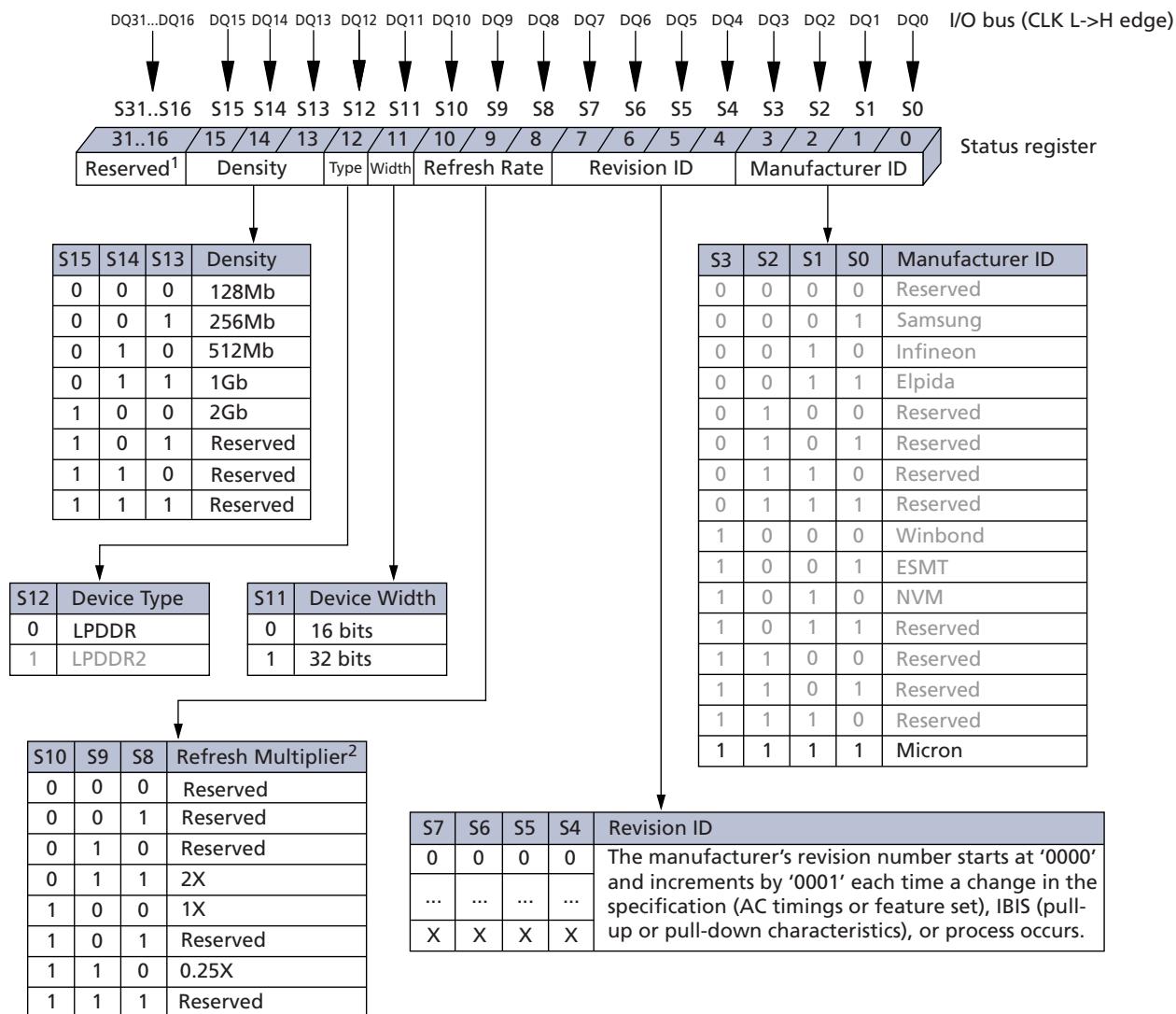
1. The device must be properly initialized and in the idle or all banks precharged state.
2. Issue a LOAD MODE REGISTER command with BA[1:0] = 01 and all address pins set to 0.
3. Wait  $t_{SRR}$ ; only NOP or DESELECT commands are supported during the  $t_{SRR}$  time.
4. Issue a READ command.
5. Subsequent commands to the device must be issued  $t_{SRC}$  after the SRR READ command is issued; only NOP or DESELECT commands are supported during  $t_{SRC}$ .

SRR output is read with a burst length of 2. SRR data is driven to the outputs on the first bit of the burst, with the output being "Don't Care" on the second bit of the burst.

**Figure 116: Status Read Register Timing**



- Notes:
1. All banks must be idle prior to status register read.
  2. NOP or DESELECT commands are required between the LMR and READ commands ( $t_{SRR}$ ), and between the READ and the next VALID command ( $t_{SRC}$ ).
  3. CAS latency is predetermined by the programming of the mode register. CL = 3 is shown as an example only.
  4. Burst length is fixed to 2 for SRR regardless of the value programmed by the mode register.
  5. The second bit of the data-out burst is a "Don't Care."

**Figure 117: Status Register Definition**

- Notes:
1. Reserved bits should be set to 0 for future compatibility.
  2. Refresh multiplier is based on the memory device on-board temperature sensor. Required average periodic refresh interval =  $t_{REFI} \times$  multiplier.



## Bank/Row Activation

Before any READ or WRITE commands can be issued to a bank within the device, a row in that bank must be opened. This is accomplished via the ACTIVE command, which selects both the bank and the row to be activated (see Figure 105 (page 158)). After a row is opened with the ACTIVE command, a READ or WRITE command can be issued to that row, subject to the  $t_{RCD}$  specification.

A subsequent ACTIVE command to a different row in the same bank can only be issued after the previous active row has been precharged. The minimum time interval between successive ACTIVE commands to the same bank is defined by  $t_{RC}$ .

A subsequent ACTIVE command to another bank can be issued while the first bank is being accessed, which results in a reduction of total row access overhead. The minimum time interval between successive ACTIVE commands to different banks is defined by  $t_{RRD}$ .



## **READ Operation**

READ burst operations are initiated with a READ command, as shown in Figure 106 (page 159). The starting column and bank addresses are provided with the READ command, and auto precharge is either enabled or disabled for that burst access. If auto precharge is enabled, the row being accessed is precharged at the completion of the burst. For the READ commands used in the following illustrations, auto precharge is disabled.

During READ bursts, the valid data-out element from the starting column address will be available following the CL after the READ command. Each subsequent data-out element will be valid nominally at the next positive or negative clock edge. Figure 118 (page 182) shows general timing for each possible CL setting.

DQS is driven by the device along with output data. The initial LOW state on DQS is known as the read preamble; the LOW state coincident with the last data-out element is known as the read postamble. The READ burst is considered complete when the read postamble is satisfied.

Upon completion of a burst, assuming no other commands have been initiated, the DQ will go to High-Z. A detailed explanation of  $t_{DQSQ}$  (valid data-out skew),  $t_{QH}$  (data-out window hold), and the valid data window is depicted in Figure 125 (page 189) and Figure 126 (page 190). A detailed explanation of  $t_{DQSCK}$  (DQS transition skew to CK) and  $t_{AC}$  (data-out transition skew to CK) is depicted in Figure 127 (page 191).

Data from any READ burst can be truncated by a READ or WRITE command to the same or alternate bank, by a BURST TERMINATE command, or by a PRECHARGE command to the same bank, provided that the auto precharge mode was not activated.

Data from any READ burst can be concatenated with or truncated with data from a subsequent READ command. In either case, a continuous flow of data can be maintained. The first data element from the new burst either follows the last element of a completed burst or the last desired data element of a longer burst that is being truncated. The new READ command should be issued  $x$  cycles after the first READ command, where  $x$  equals the number of desired data element pairs (pairs are required by the  $2n$ -prefetch architecture). This is shown in Figure 119 (page 183).

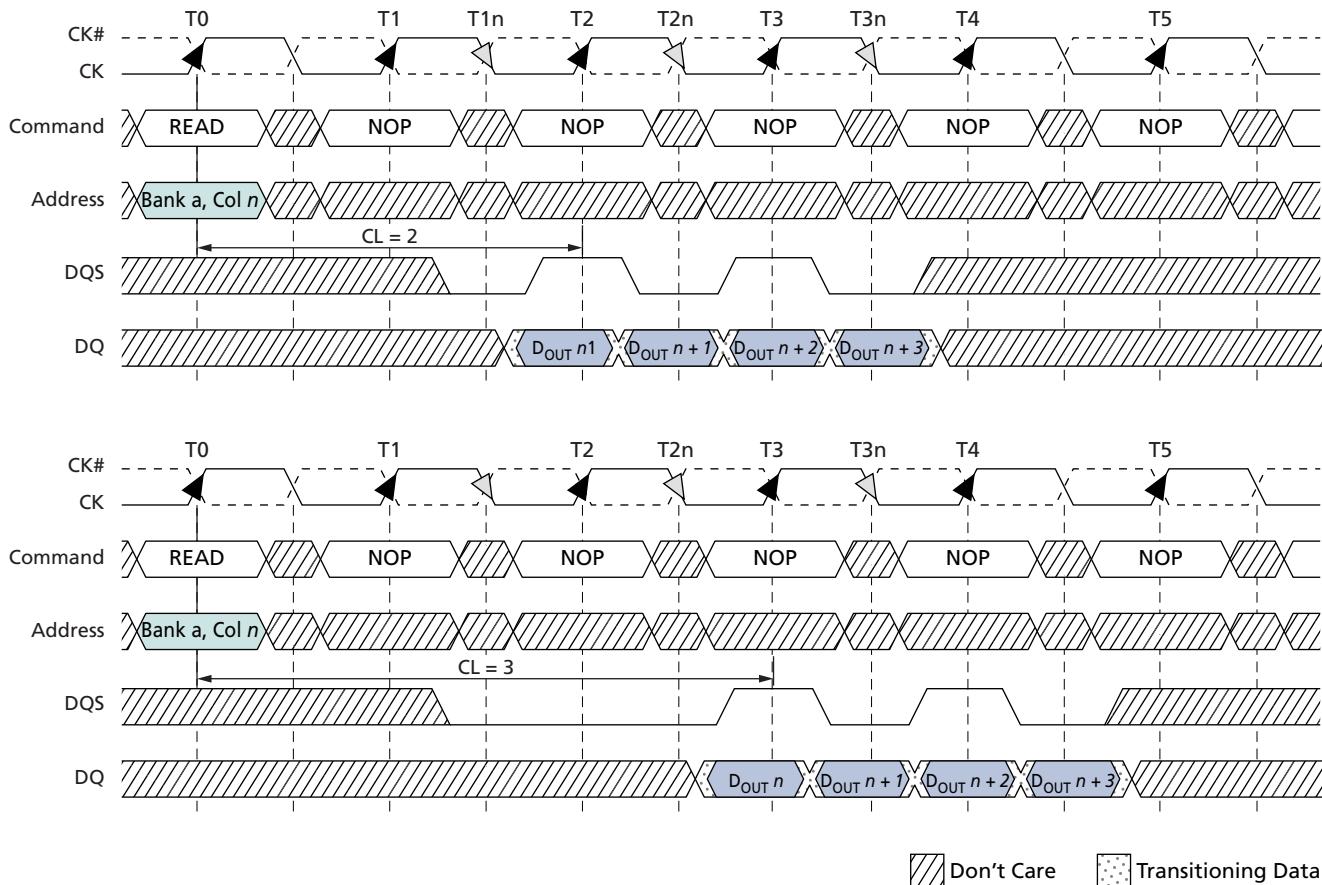
A READ command can be initiated on any clock cycle following a previous READ command. Nonconsecutive read data is shown in Figure 120 (page 184). Full-speed random read accesses within a page (or pages) can be performed as shown in Figure 121 (page 185).

Data from any READ burst can be truncated with a BURST TERMINATE command, as shown in Figure 122 (page 186). The BURST TERMINATE latency is equal to the READ (CAS) latency; for example, the BURST TERMINATE command should be issued  $x$  cycles after the READ command, where  $x$  equals the number of desired data element pairs (pairs are required by the  $2n$ -prefetch architecture).

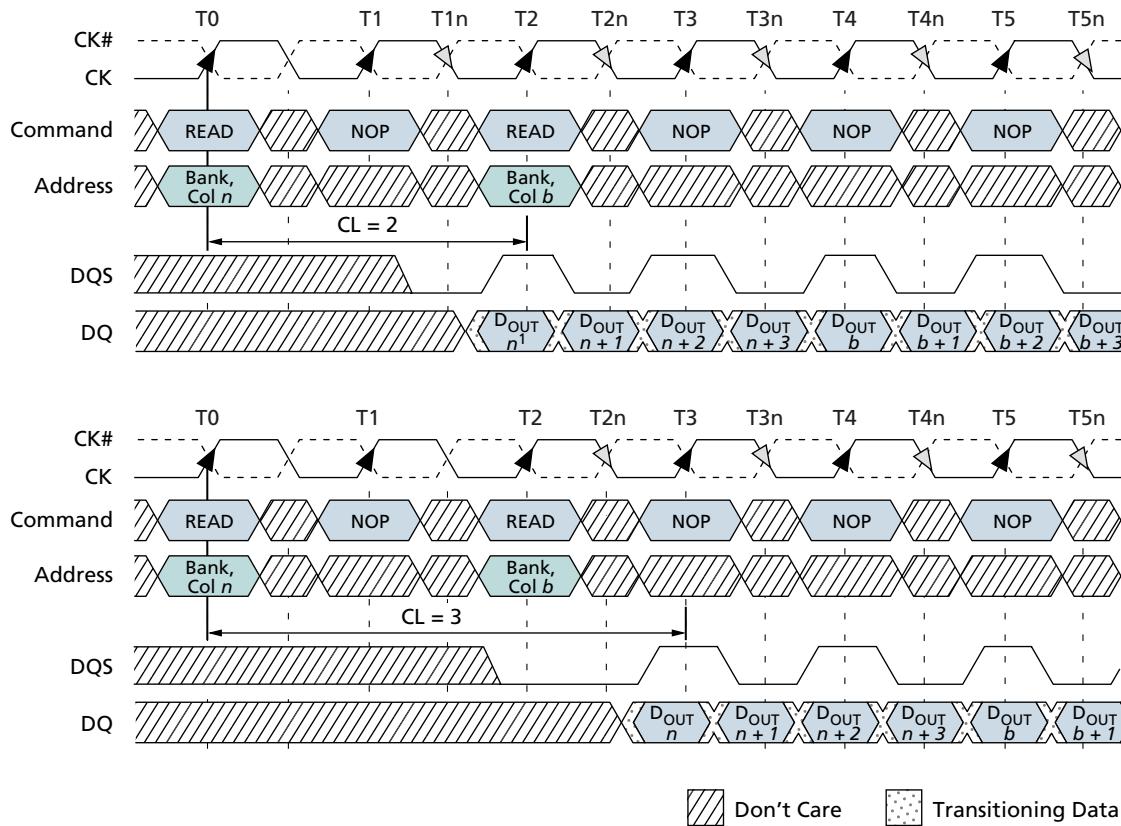
Data from any READ burst must be completed or truncated before a subsequent WRITE command can be issued. If truncation is necessary, the BURST TERMINATE command must be used, as shown in Figure 123 (page 187). A READ burst can be followed by, or truncated with, a PRECHARGE command to the same bank, provided that auto precharge was not activated. The PRECHARGE command should be issued  $x$  cycles after the READ command, where  $x$  equals the number of desired data element pairs. This is shown in Figure 124 (page 188). Following the PRECHARGE command, a subsequent

command to the same bank cannot be issued until  $t_{RP}$  is met. Part of the row precharge time is hidden during the access of the last data elements.

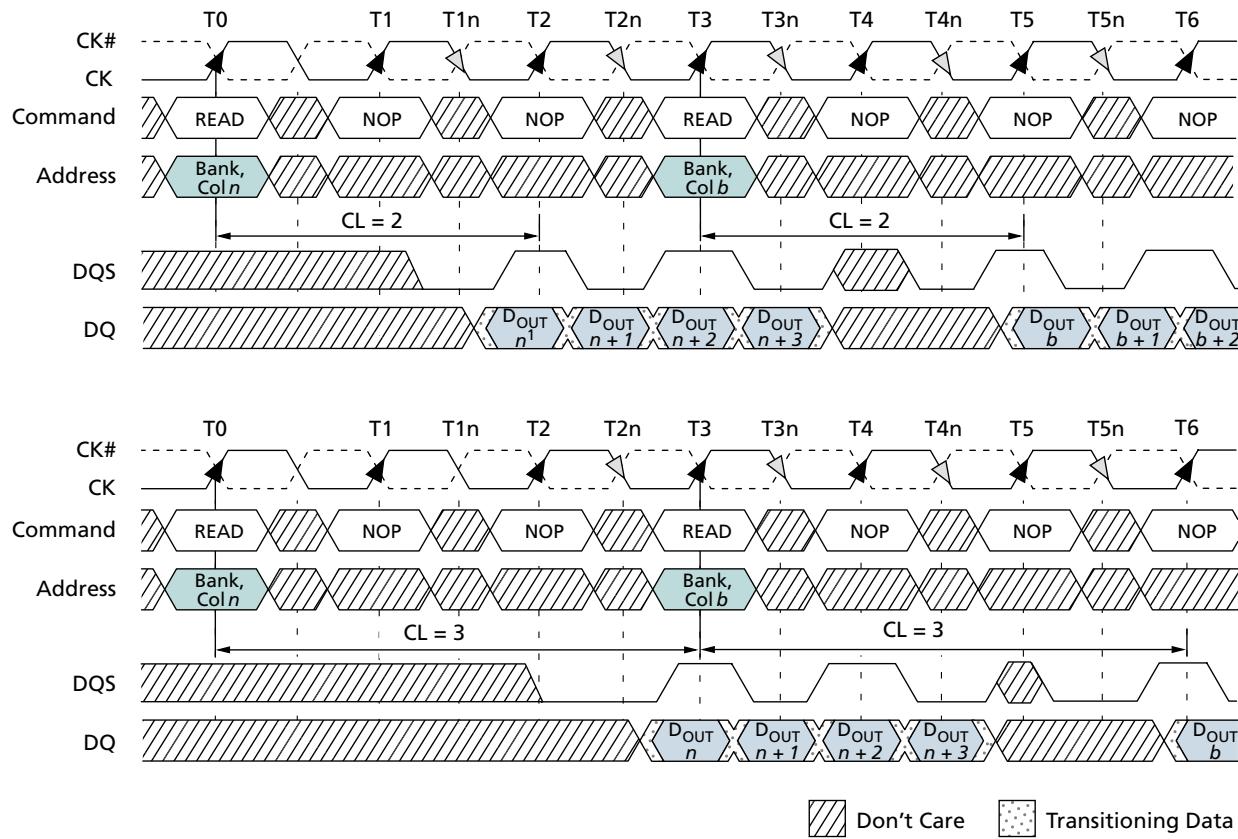
**Figure 118: READ Burst**



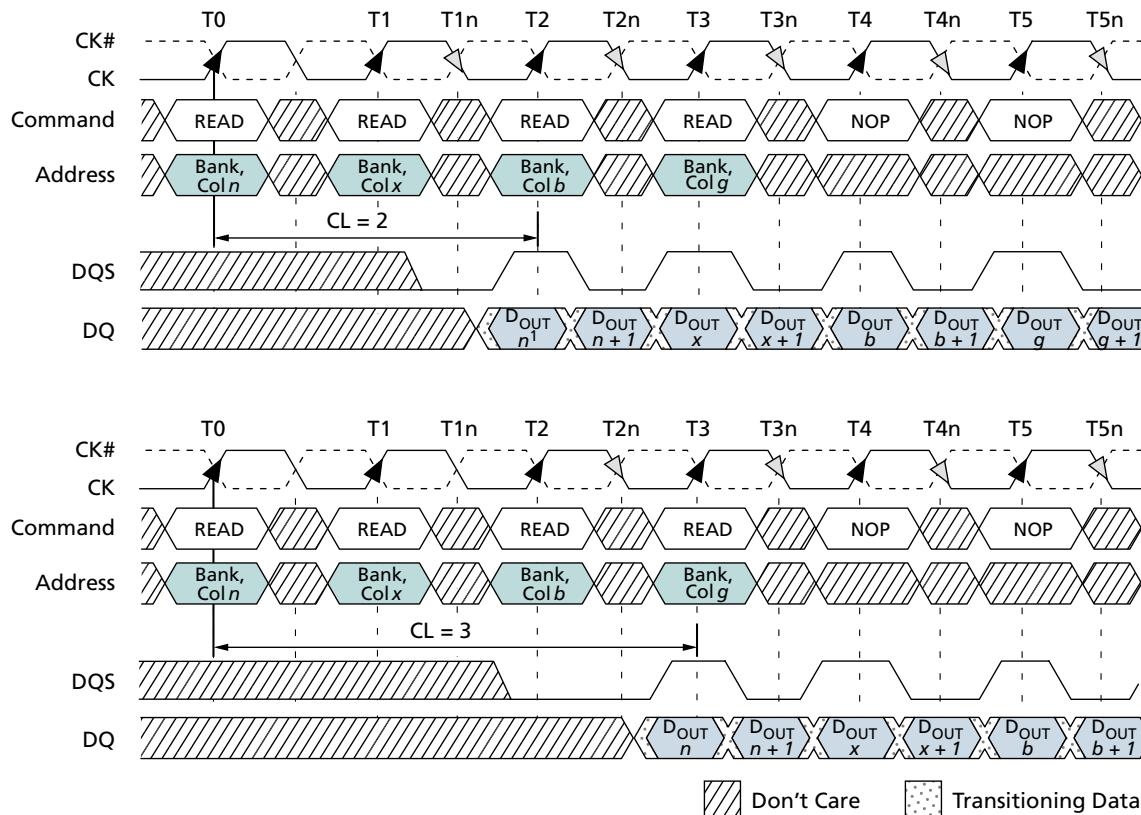
- Notes:
1.  $D_{OUT\ n}$  = data-out from column  $n$ .
  2.  $BL = 4$ .
  3. Shown with nominal  $t_{AC}$ ,  $t_{DQSCK}$ , and  $t_{DQSQ}$ .

**Figure 119: Consecutive READ Bursts**

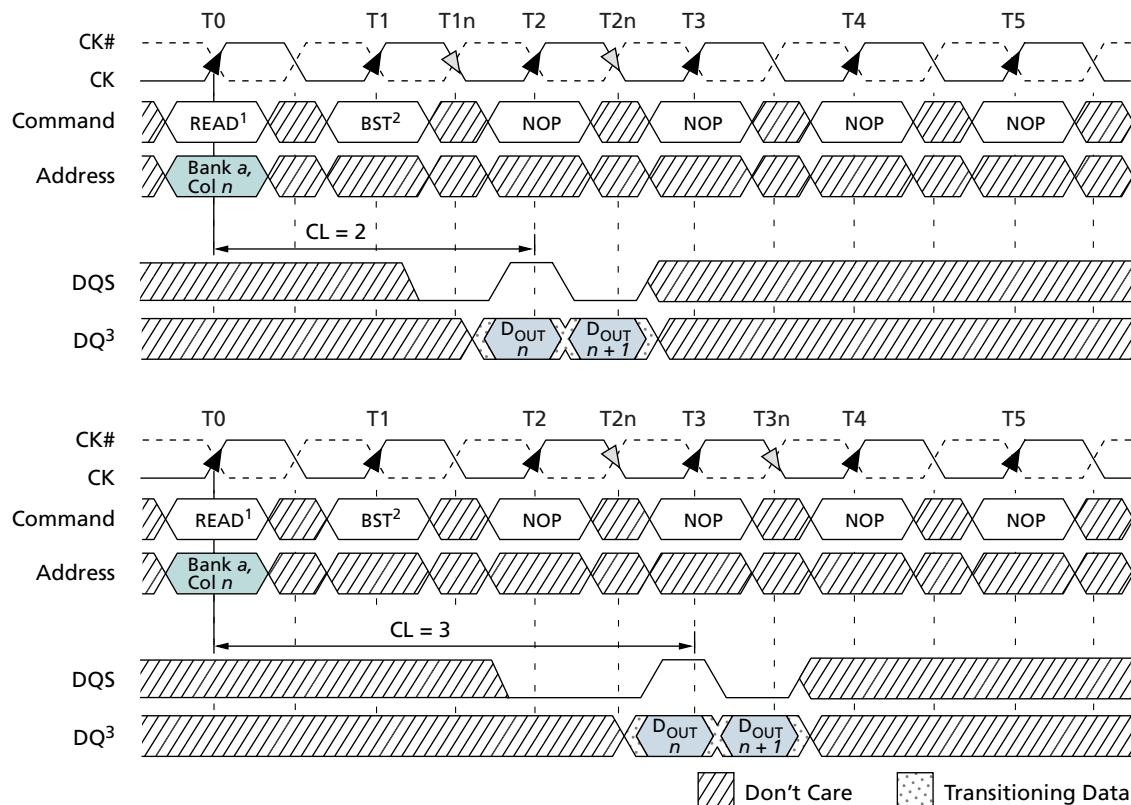
- Notes:
1.  $D_{OUTn}$  (or  $b$ ) = data-out from column  $n$  (or column  $b$ ).
  2. BL = 4, 8, or 16 (if 4, the bursts are concatenated; if 8 or 16, the second burst interrupts the first).
  3. Shown with nominal  $t_{AC}$ ,  $t_{DQSCK}$ , and  $t_{DQSQ}$ .
  4. Example applies only when READ commands are issued to same device.

**Figure 120: Nonconsecutive READ Bursts**

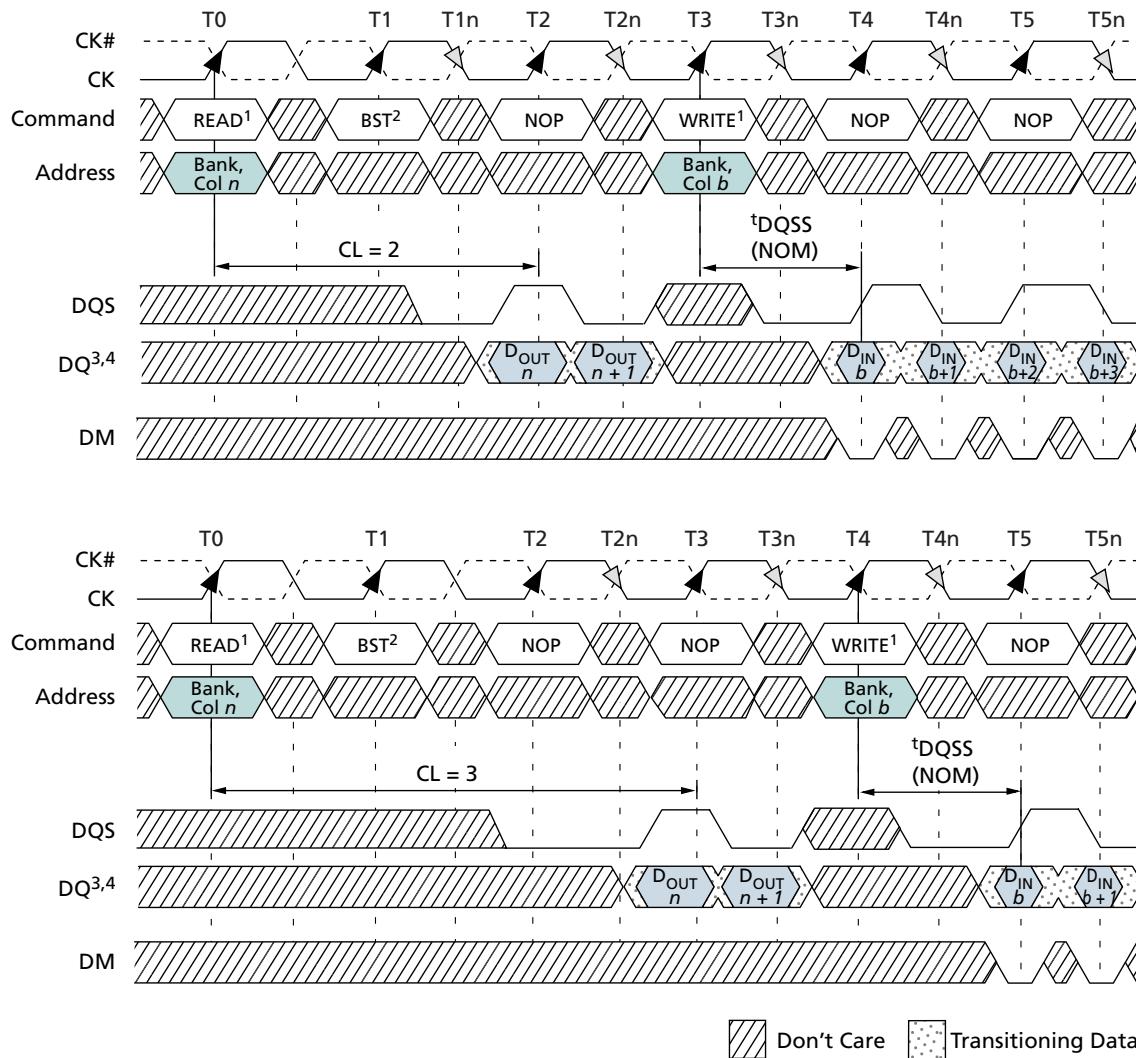
- Notes:
1.  $D_{OUTn}$  (or  $b$ ) = data-out from column  $n$  (or column  $b$ ).
  2. BL = 4, 8, or 16 (if burst is 8 or 16, the second burst interrupts the first).
  3. Shown with nominal  $t_{AC}$ ,  $t_{DQSCK}$ , and  $t_{DQSQ}$ .
  4. Example applies when READ commands are issued to different devices or nonconsecutive READs.

**Figure 121: Random Read Accesses**

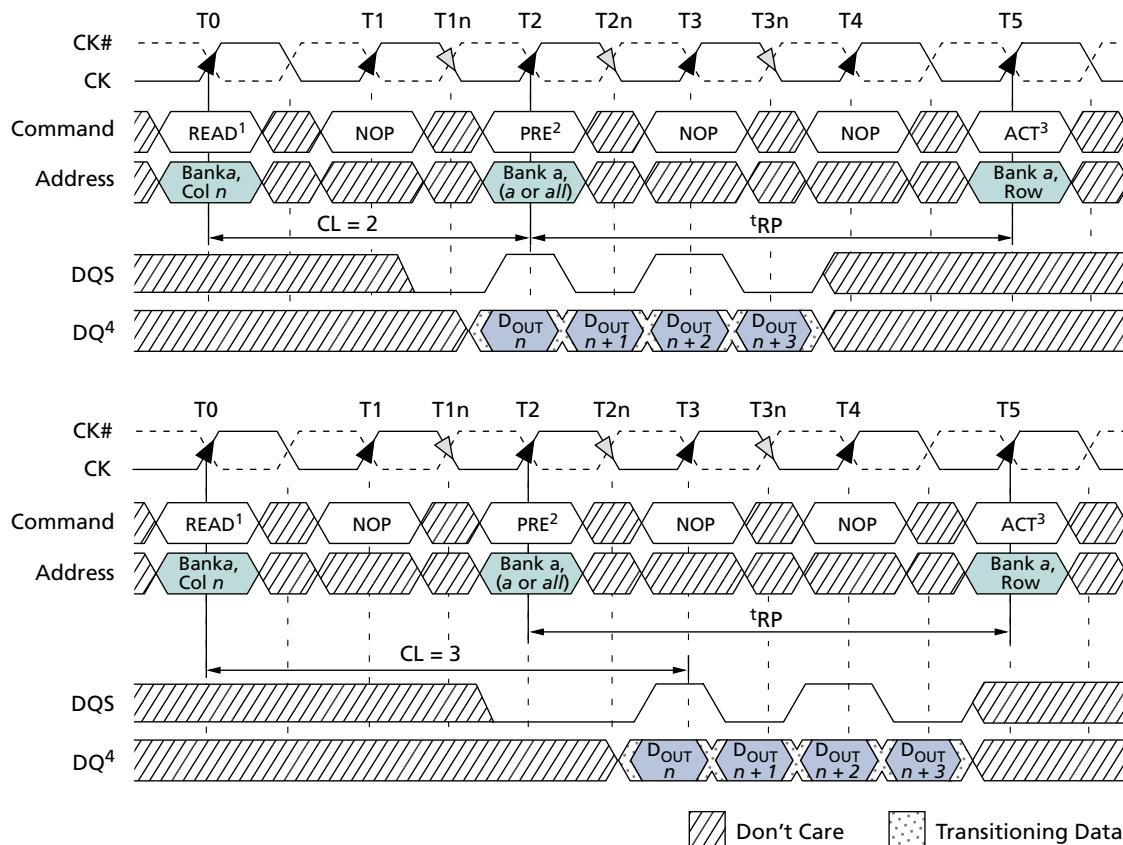
- Notes:
1.  $D_{OUT\ n}$  (or  $x$ ,  $b$ ,  $g$ ) = data-out from column  $n$  (or column  $x$ , column  $b$ , column  $g$ ).
  2. BL = 2, 4, 8, or 16 (if 4, 8, or 16, the following burst interrupts the previous).
  3. READs are to an active row in any bank.
  4. Shown with nominal  $t_{AC}$ ,  $t_{DQSCK}$ , and  $t_{DQSQ}$ .

**Figure 122: Terminating a READ Burst**

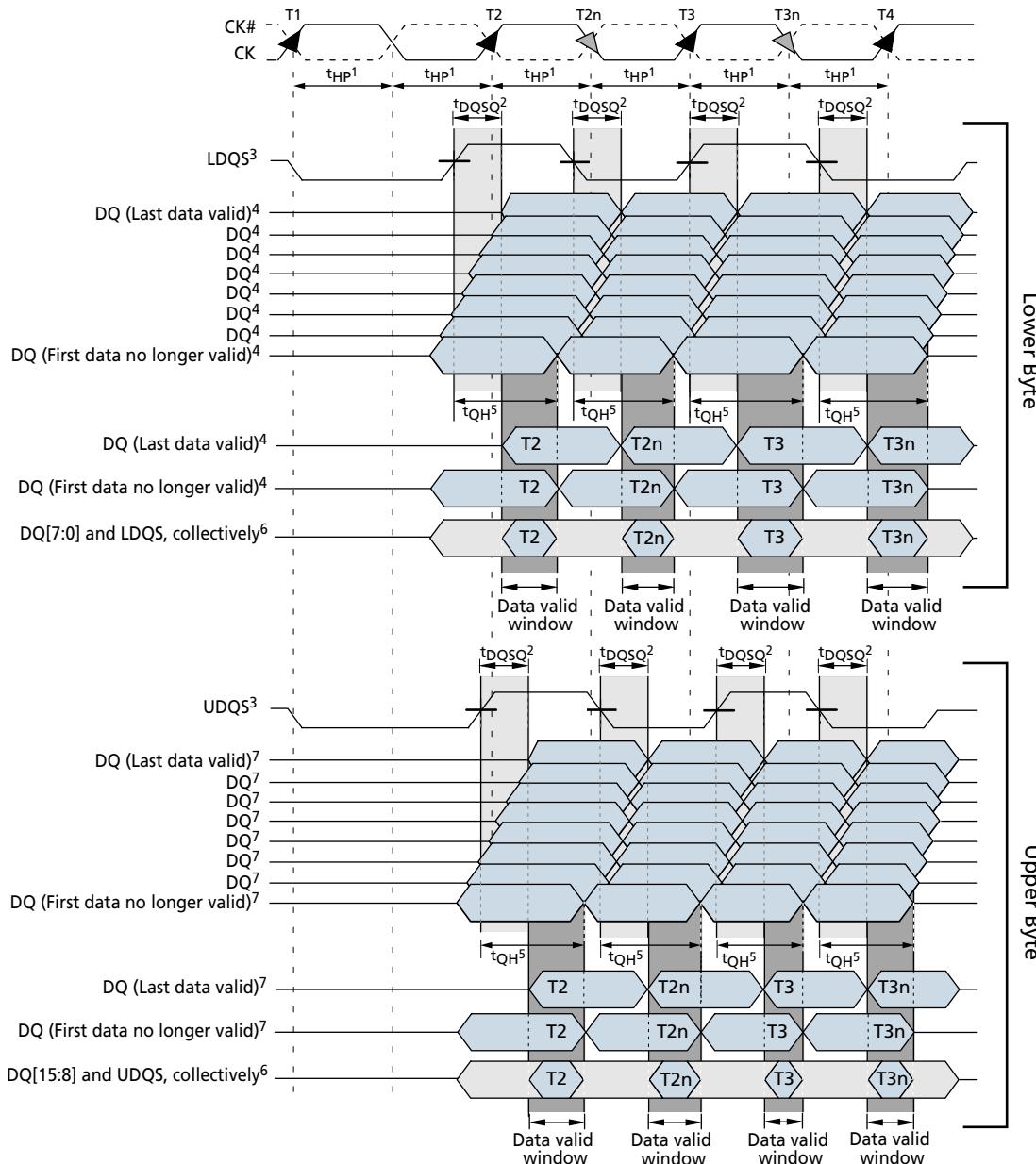
- Notes:
1. BL = 4, 8, or 16.
  2. BST = BURST TERMINATE command; page remains open.
  3. D<sub>OUTn</sub> = data-out from column n.
  4. Shown with nominal <sup>t</sup>AC, <sup>t</sup>DQSCK, and <sup>t</sup>DQSQ.
  5. CKE = HIGH.

**Figure 123: READ-to-WRITE**

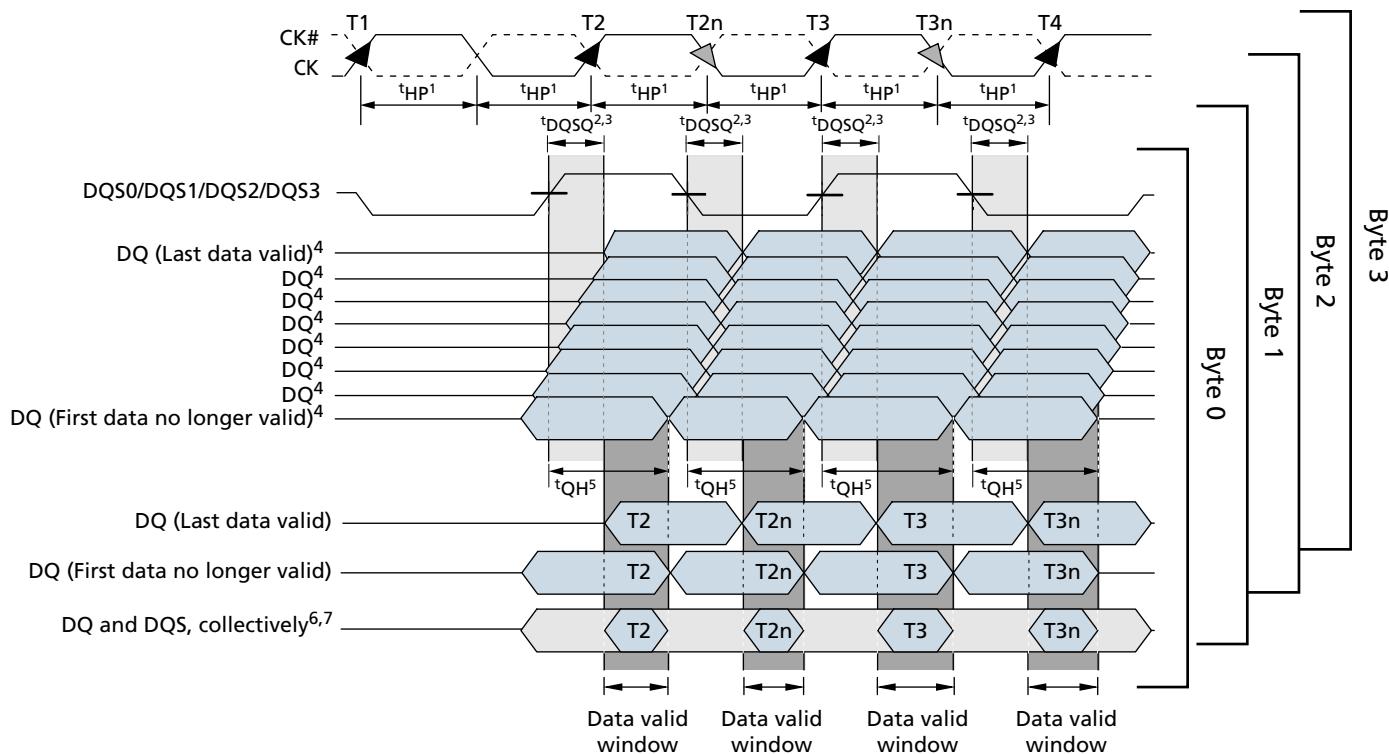
- Notes:
1. BL = 4 in the cases shown (applies for bursts of 8 and 16 as well; if BL = 2, the BST command shown can be NOP).
  2. BST = BURST TERMINATE command; page remains open.
  3. D<sub>OUT</sub>n = data-out from column n.
  4. D<sub>IN</sub>b = data-in from column b.
  5. Shown with nominal tAC, tDQSCK, and tDQSQ.
  6. CKE = HIGH.

**Figure 124: READ-to-PRECHARGE**

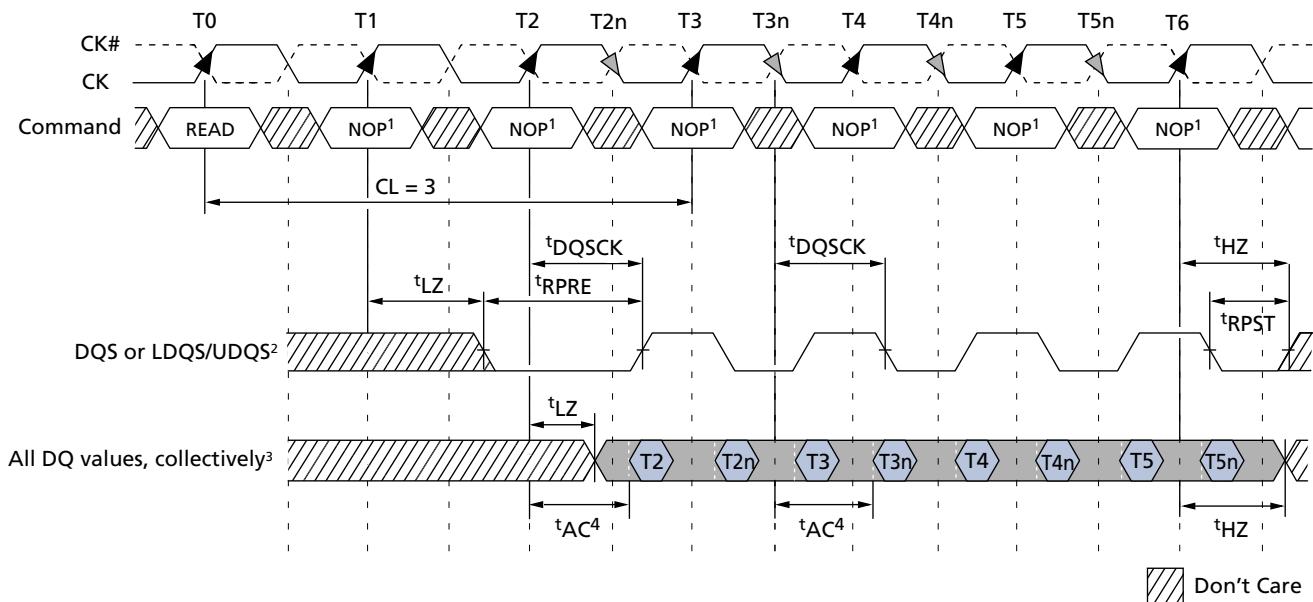
- Notes:
1. BL = 4, or an interrupted burst of 8 or 16.
  2. PRE = PRECHARGE command.
  3. ACT = ACTIVE command.
  4. DOUT<sub>n</sub> = data-out from column n.
  5. Shown with nominal tAC, tDQSK, and tDQSQ.
  6. READ-to-PRECHARGE equals 2 clocks, which enables 2 data pairs of data-out.
  7. A READ command with auto precharge enabled, provided tRAS (MIN) is met, would cause a precharge to be performed at x number of clock cycles after the READ command, where x = BL/2.

**Figure 125: Data Output Timing –  $t_{DQSQ}$ ,  $t_{QH}$ , and Data Valid Window (x16)**

- Notes:
1.  $t_{HP}$  is the lesser of  $t_{CL}$  or  $t_{CH}$  clock transition collectively when a bank is active.
  2.  $t_{DQSQ}$  is derived at each DQS clock edge and is not cumulative over time and begins with DQS transition and ends with the last valid DQ transition.
  3. DQ transitioning after DQS transitions define the  $t_{DQSQ}$  window. LDQS defines the lower byte and UDQS defines the upper byte.
  4. DQ0, DQ1, DQ2, DQ3, DQ4, DQ5, DQ6, or DQ7.
  5.  $t_{QH}$  is derived from  $t_{HP}$ :  $t_{QH} = t_{HP} - t_{QHS}$ .
  6. The data valid window is derived for each DQS transitions and is defined as  $t_{QH} - t_{DQSQ}$ .
  7. DQ8, DQ9, DQ10, DQ11, DQ12, DQ13, DQ14, or DQ15.

**Figure 126: Data Output Timing –  $t_{DQSQ}$ ,  $t_{QH}$ , and Data Valid Window (x32)**


- Notes:
1.  $t_{HP}$  is the lesser of  $t_{CL}$  or  $t_{CH}$  clock transition collectively when a bank is active.
  2. DQ transitioning after DQS transitions define the  $t_{DQSQ}$  window.
  3.  $t_{DQSQ}$  is derived at each DQS clock edge and is not cumulative over time; it begins with DQS transition and ends with the last valid DQ transition.
  4. Byte 0 is DQ[7:0], byte 1 is DQ[15:8], byte 2 is DQ[23:16], byte 3 is DQ[31:24].
  5.  $t_{QH}$  is derived from  $t_{HP}$ :  $t_{QH} = t_{HP} - t_{QHS}$ .
  6. The data valid window is derived for each DQS transition and is  $t_{QH} - t_{DQSQ}$ .
  7. DQ[7:0] and DQS0 for byte 0; DQ[15:8] and DQS1 for byte 1; DQ[23:16] and DQS2 for byte 2; DQ[31:23] and DQS3 for byte 3.

**Figure 127: Data Output Timing –  $t_{AC}$  and  $t_{DQSCK}$** 

- Notes:
1. Commands other than NOP can be valid during this cycle.
  2. DQ transitioning after DQS transitions define  $t_{DQSQ}$  window.
  3. All DQ must transition by  $t_{DQSQ}$  after DQS transitions, regardless of  $t_{AC}$ .
  4.  $t_{AC}$  is the DQ output window relative to CK and is the long-term component of DQ skew.



## WRITE Operation

WRITE bursts are initiated with a WRITE command, as shown in Figure 107 (page 160). The starting column and bank addresses are provided with the WRITE command, and auto precharge is either enabled or disabled for that access. If auto precharge is enabled, the row being accessed is precharged at the completion of the burst. For the WRITE commands used in the following illustrations, auto precharge is disabled. Basic data input timing is shown in Figure 128 (page 193) (this timing applies to all WRITE operations).

Input data appearing on the data bus is written to the memory array subject to the state of data mask (DM) inputs coincident with the data. If DM is registered LOW, the corresponding data will be written; if DM is registered HIGH, the corresponding data will be ignored, and the write will not be executed to that byte/column location. DM operation is illustrated in Figure 129 (page 194).

During WRITE bursts, the first valid data-in element will be registered on the first rising edge of DQS following the WRITE command, and subsequent data elements will be registered on successive edges of DQS. The LOW state of DQS between the WRITE command and the first rising edge is known as the write preamble; the LOW state of DQS following the last data-in element is known as the write postamble. The WRITE burst is complete when the write postamble and  $t_{WR}$  or  $t_{WTR}$  are satisfied.

The time between the WRITE command and the first corresponding rising edge of DQS ( $t_{DQSS}$ ) is specified with a relatively wide range (75%–125% of one clock cycle). All WRITE diagrams show the nominal case. Where the two extreme cases (that is,  $t_{DQSS}$  [MIN] and  $t_{DQSS}$  [MAX]) might not be obvious, they have also been included. Figure 130 (page 195) shows the nominal case and the extremes of  $t_{DQSS}$  for a burst of 4. Upon completion of a burst, assuming no other commands have been initiated, the DQ will remain High-Z and any additional input data will be ignored.

Data for any WRITE burst can be concatenated with or truncated by a subsequent WRITE command. In either case, a continuous flow of input data can be maintained. The new WRITE command can be issued on any positive edge of clock following the previous WRITE command. The first data element from the new burst is applied after either the last element of a completed burst or the last desired data element of a longer burst that is being truncated. The new WRITE command should be issued  $x$  cycles after the first WRITE command, where  $x$  equals the number of desired data element pairs (pairs are required by the  $2n$ -prefetch architecture).

Figure 131 (page 196) shows concatenated bursts of 4. An example of nonconsecutive WRITEs is shown in Figure 132 (page 196). Full-speed random write accesses within a page or pages can be performed, as shown in Figure 133 (page 197).

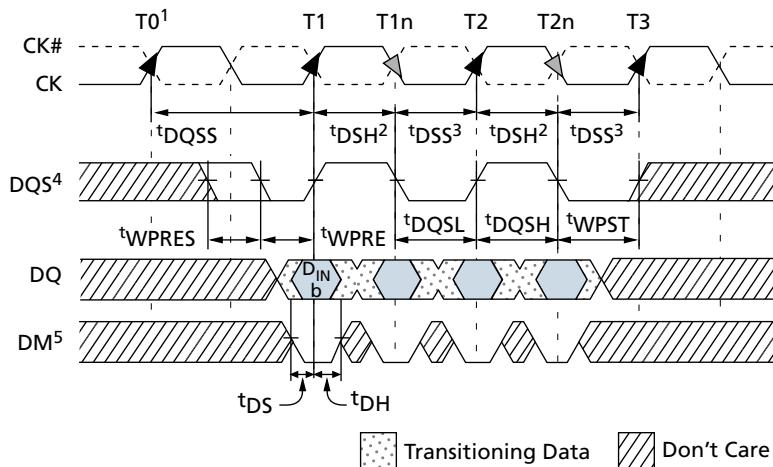
Data for any WRITE burst can be followed by a subsequent READ command. To follow a WRITE without truncating the WRITE burst,  $t_{WTR}$  should be met, as shown in Figure 134 (page 198).

Data for any WRITE burst can be truncated by a subsequent READ command, as shown in Figure 135 (page 199). Note that only the data-in pairs that are registered prior to the  $t_{WTR}$  period are written to the internal array, and any subsequent data-in should be masked with DM, as shown in Figure 136 (page 200).

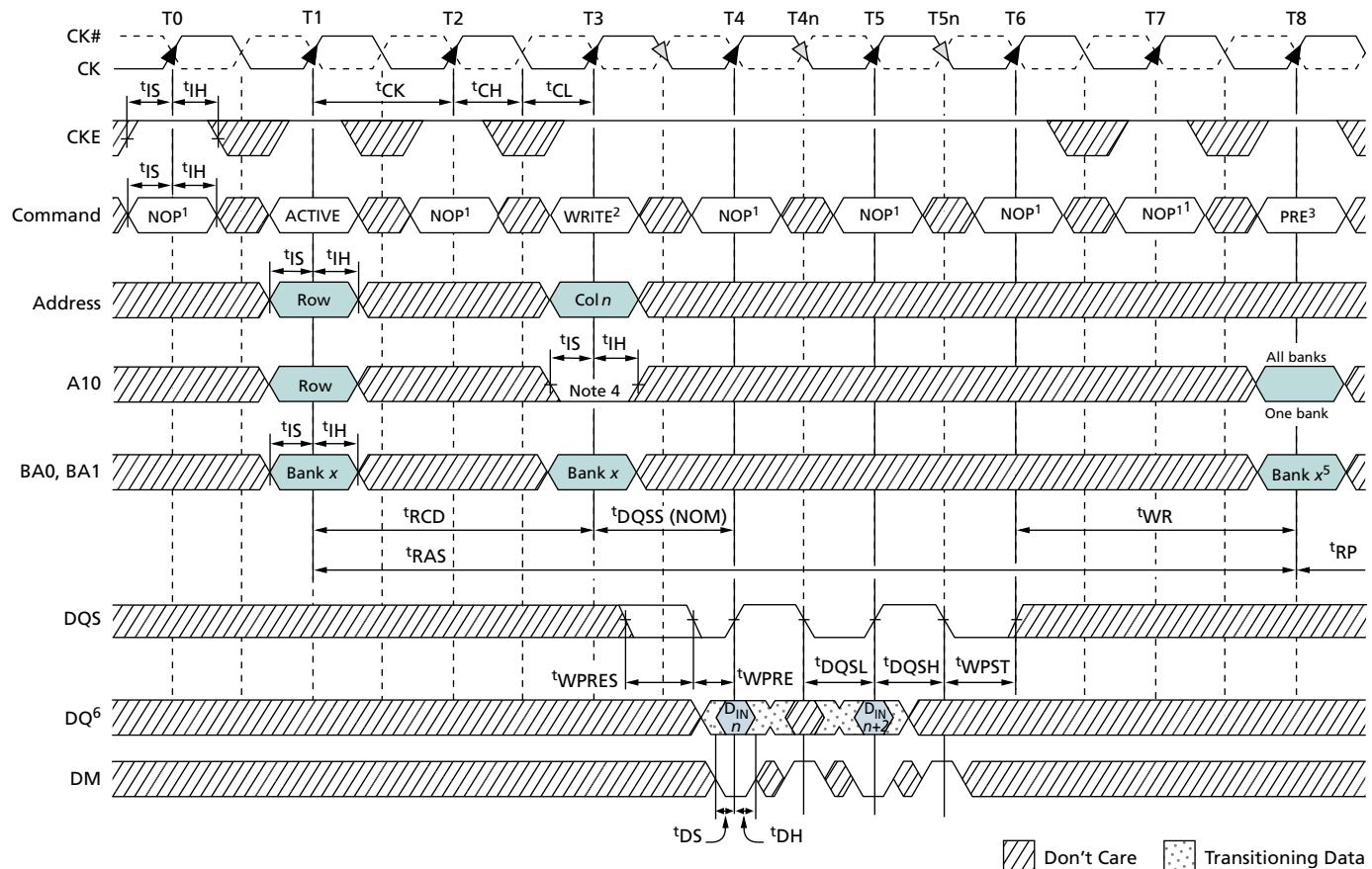
Data for any WRITE burst can be followed by a subsequent PRECHARGE command. To follow a WRITE without truncating the WRITE burst,  $t_{WR}$  should be met, as shown in Figure 137 (page 201).

Data for any WRITE burst can be truncated by a subsequent PRECHARGE command, as shown in Figure 138 (page 202) and Figure 139 (page 203). Note that only the data-in pairs that are registered prior to the  $t^{WR}$  period are written to the internal array, and any subsequent data-in should be masked with DM, as shown in Figure 138 (page 202) and Figure 139 (page 203). After the PRECHARGE command, a subsequent command to the same bank cannot be issued until  $t^{RP}$  is met.

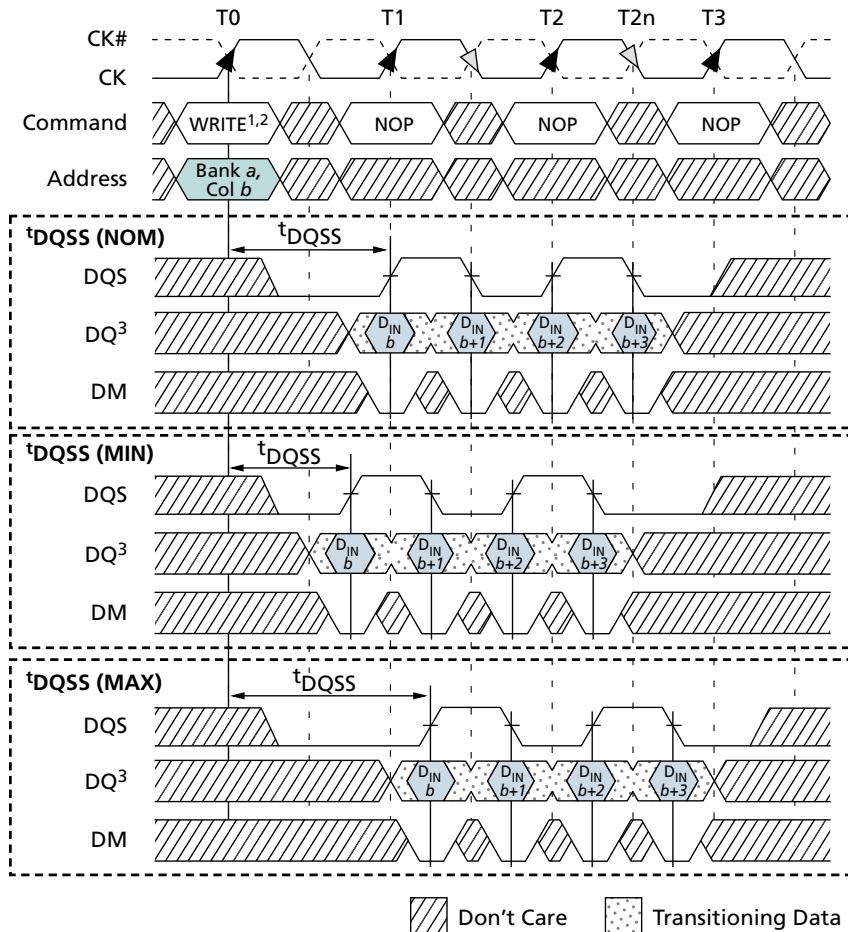
**Figure 128: Data Input Timing**



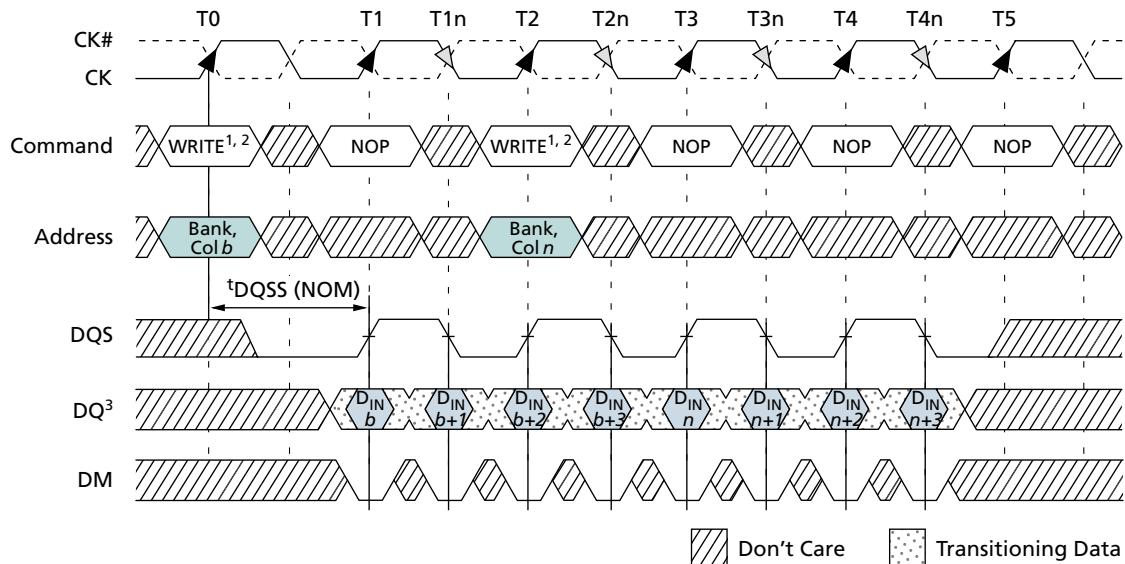
- Notes:
1. WRITE command issued at  $T_0^1$ .
  2.  $t^{DSH}$  (MIN) generally occurs during  $t^{DQSS}$  (MIN).
  3.  $t^{DSS}$  (MIN) generally occurs during  $t^{DQSS}$  (MAX).
  4. For x16, LDQS controls the lower byte; UDQS controls the upper byte. For x32, DQS0 controls DQ[7:0], DQS1 controls DQ[15:8], DQS2 controls DQ[23:16], and DQS3 controls DQ[31:24].
  5. For x16, LDM controls the lower byte; UDM controls the upper byte. For x32, DM0 controls DQ[7:0], DM1 controls DQ[15:8], DM2 controls DQ[23:16], and DM3 controls DQ[31:24].

**Figure 129: Write – DM Operation**

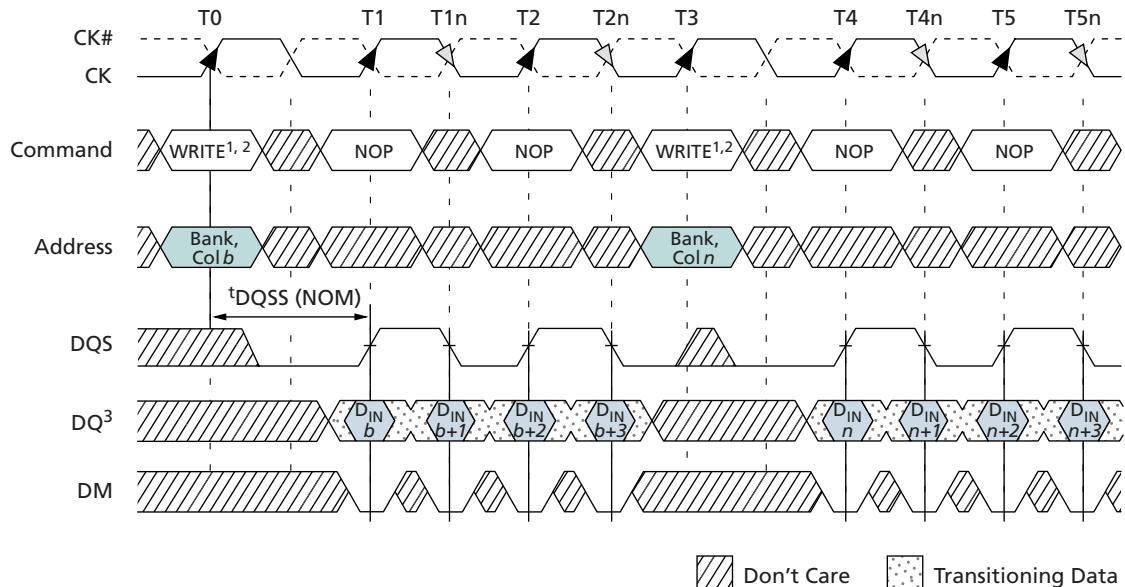
- Notes:
1. NOP commands are shown for ease of illustration; other commands may be valid at these times.
  2. BL = 4 in the case shown.
  3. PRE = PRECHARGE.
  4. Disable auto precharge.
  5. Bank x at T8 is “Don’t Care” if A10 is HIGH at T8.
  6. D<sub>IN</sub>n = data-in from column n.

**Figure 130: WRITE Burst**

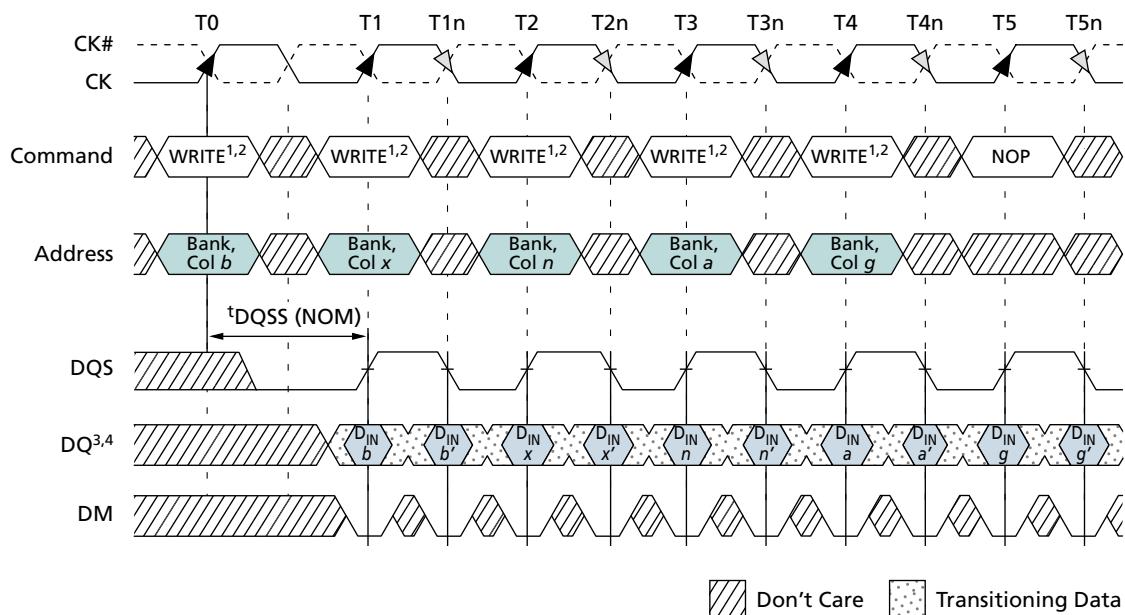
- Notes:
1. An uninterrupted burst of 4 is shown.
  2. A10 is LOW with the WRITE command (auto precharge is disabled).
  3. D<sub>IN</sub><sub>b</sub> = data-in for column b.

**Figure 131: Consecutive WRITE-to-WRITE**

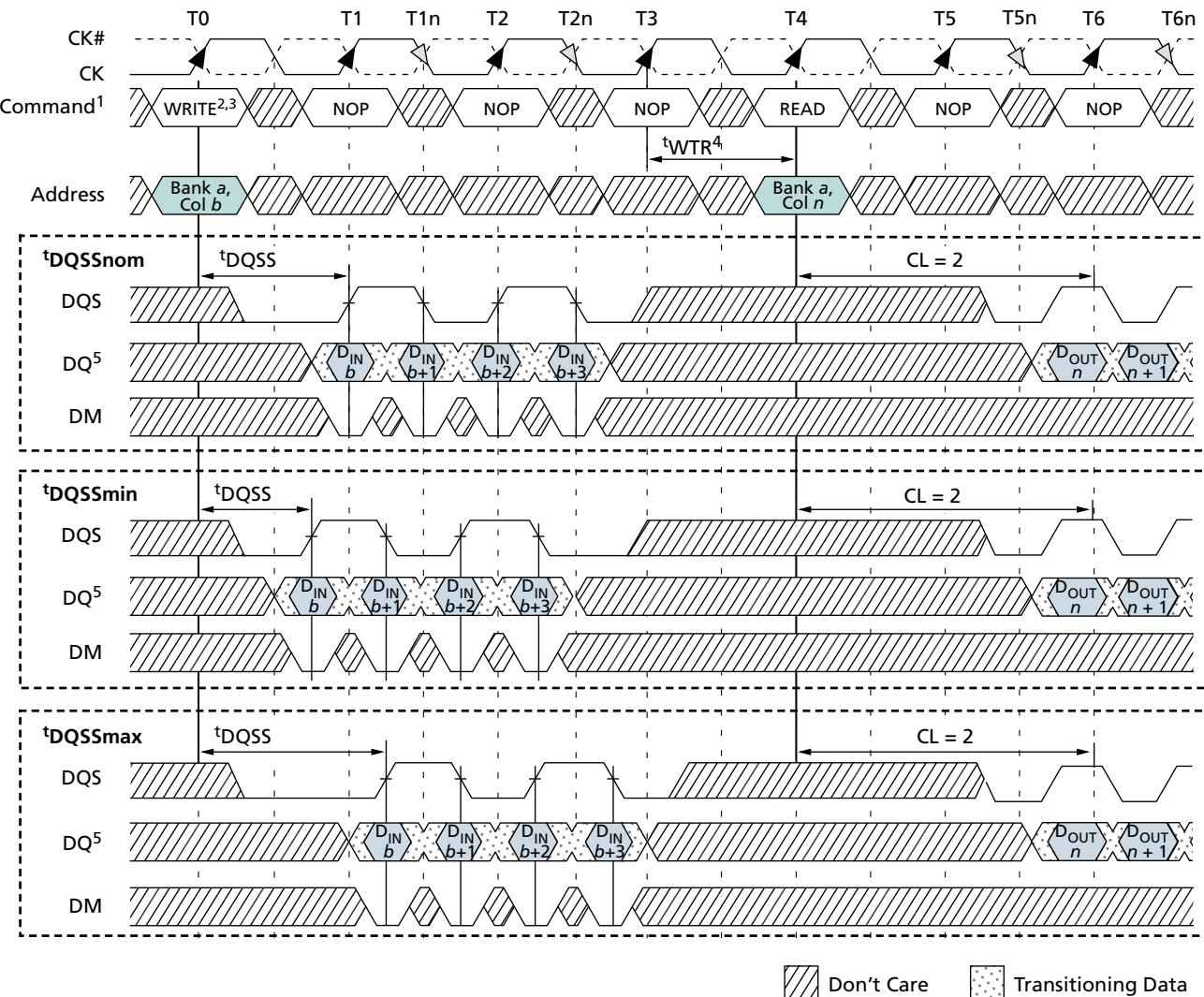
- Notes:
1. Each WRITE command can be to any bank.
  2. An uninterrupted burst of 4 is shown.
  3. D<sub>IN</sub><sub>b</sub> (n) = data-in for column b (n).

**Figure 132: Nonconsecutive WRITE-to-WRITE**

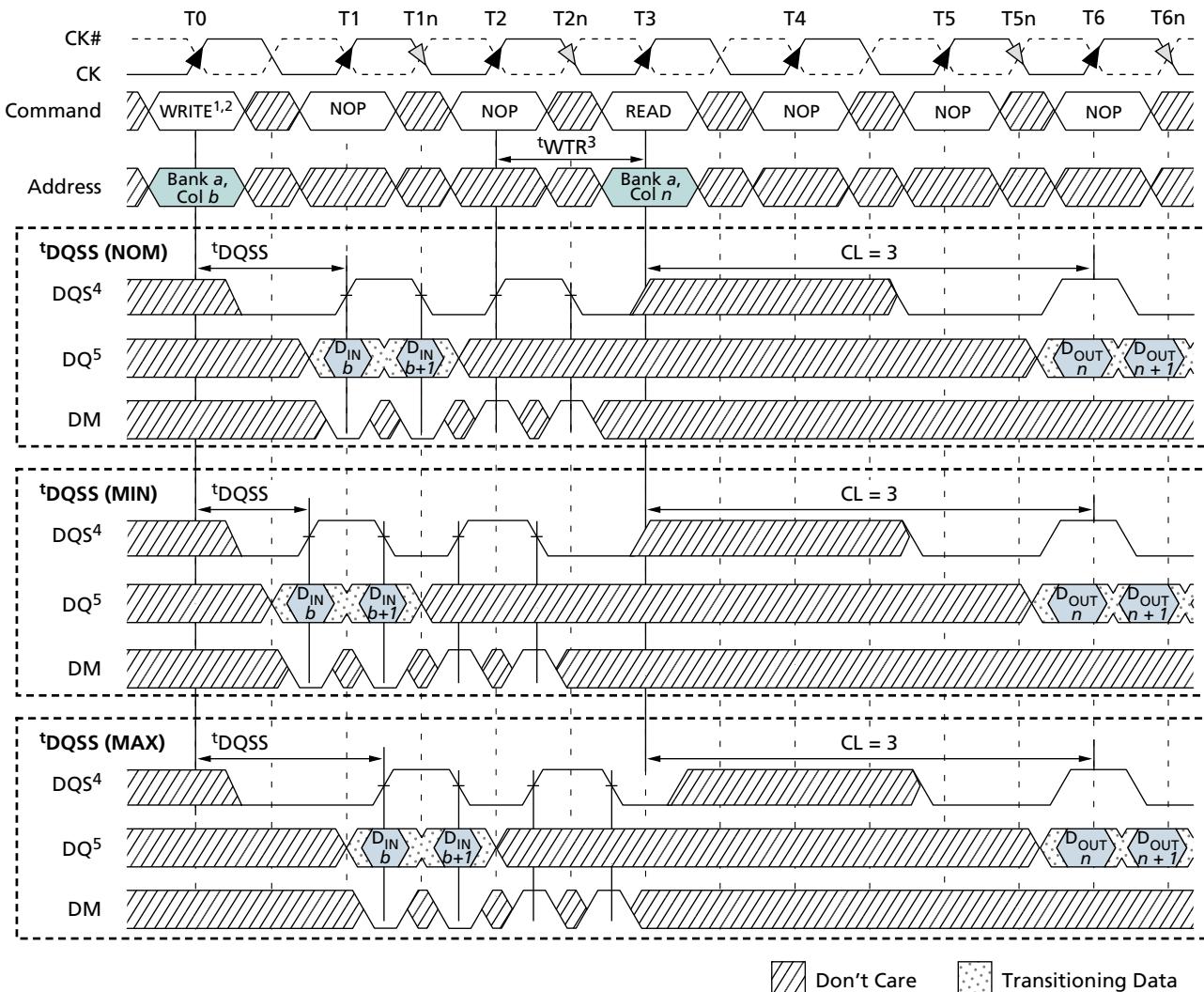
- Notes:
1. Each WRITE command can be to any bank.
  2. An uninterrupted burst of 4 is shown.
  3. D<sub>IN</sub><sub>b</sub> (n) = data-in for column b (n).

**Figure 133: Random WRITE Cycles**

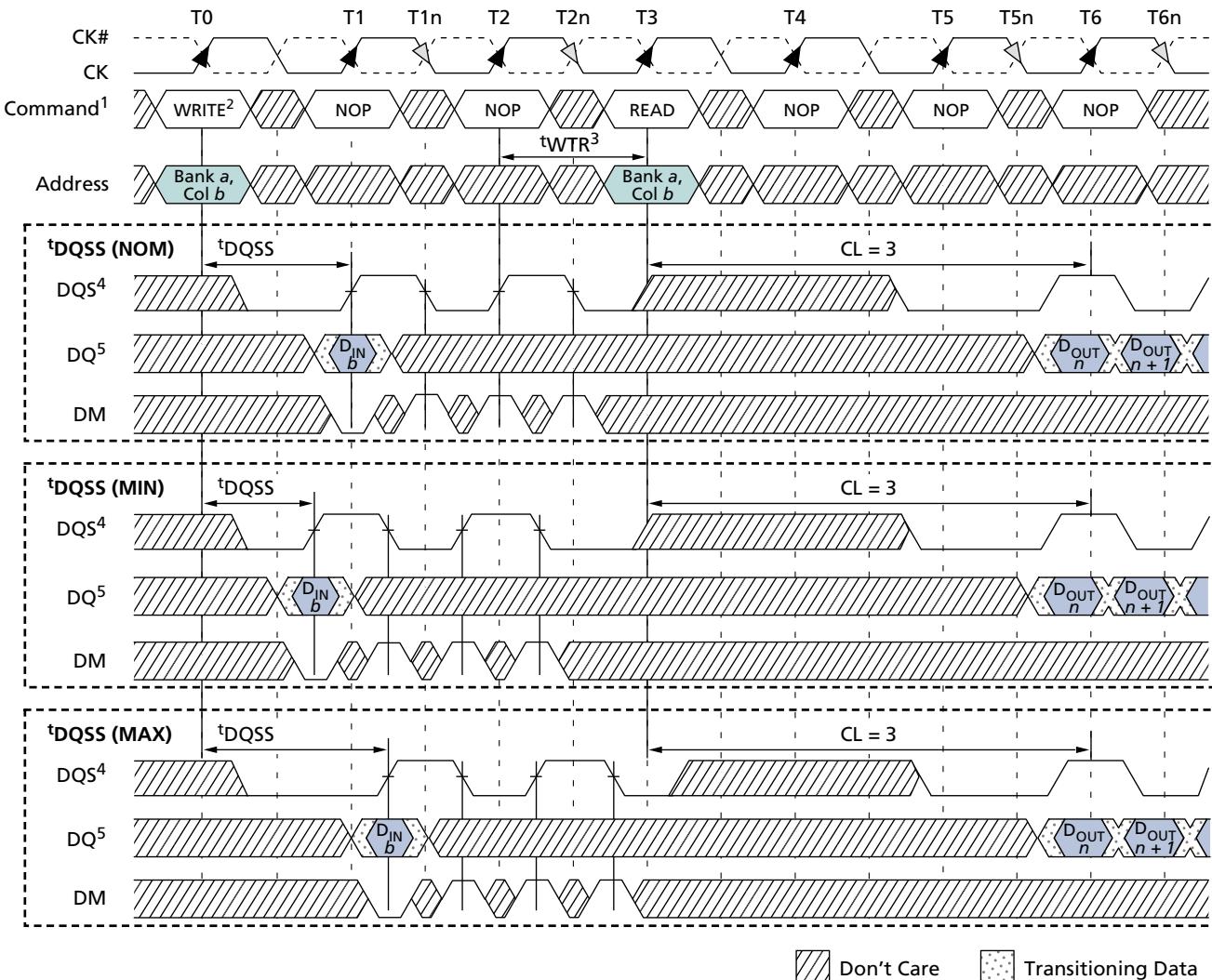
- Notes:
1. Each WRITE command can be to any bank.
  2. Programmed BL = 2, 4, 8, or 16 in cases shown.
  3.  $D_{IN}b$  (or  $x, n, a, g$ ) = data-in for column  $b$  (or  $x, n, a, g$ ).
  4.  $b'$  (or  $x, n, a, g$ ) = the next data-in following  $D_{IN}b$  ( $x, n, a, g$ ) according to the programmed burst order.

**Figure 134: WRITE-to-READ – Uninterrupting**

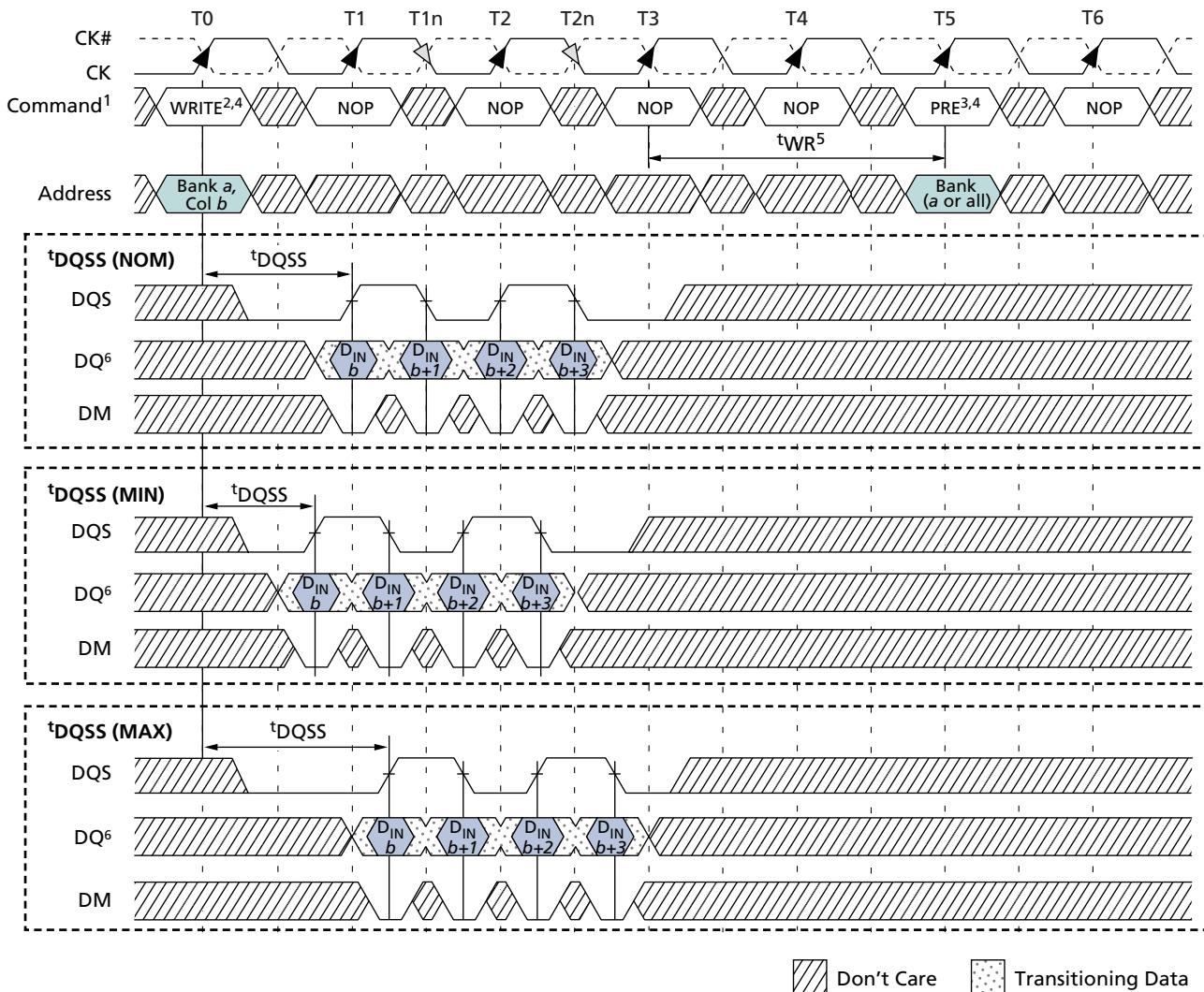
- Notes:
1. The READ and WRITE commands are to the same device. However, the READ and WRITE commands may be to different devices, in which case  $t^{WTR}$  is not required and the READ command could be applied earlier.
  2. A10 is LOW with the WRITE command (auto precharge is disabled).
  3. An uninterrupted burst of 4 is shown.
  4.  $t^{WTR}$  is referenced from the first positive CK edge after the last data-in pair.
  5.  $D_{IN}b$  = data-in for column b;  $D_{OUT}n$  = data-out for column n.

**Figure 135: WRITE-to-READ – Interrupting**

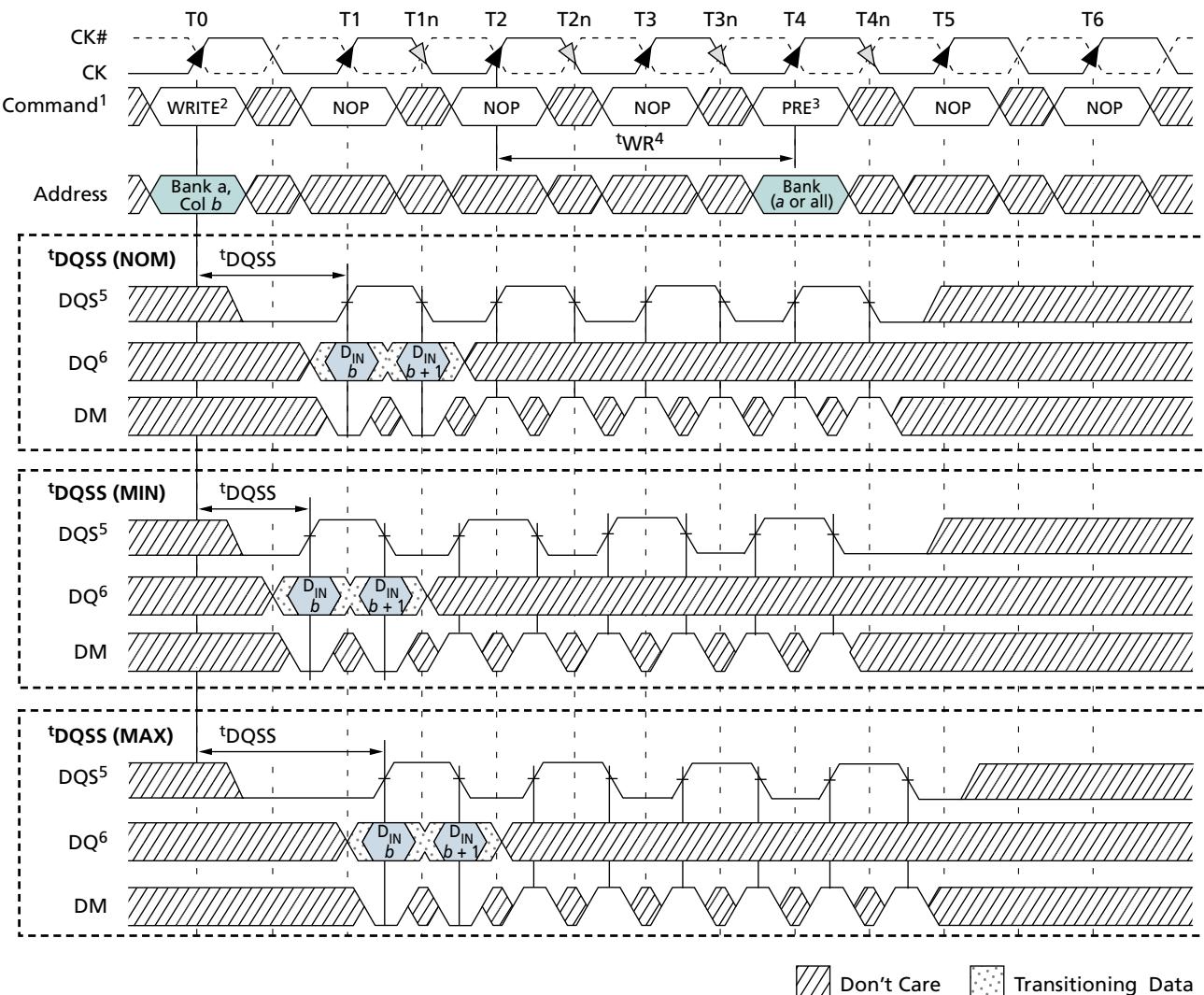
- Notes:
- An interrupted burst of 4 is shown; 2 data elements are written.
  - A10 is LOW with the WRITE command (auto precharge is disabled).
  - $tWTR$  is referenced from the first positive CK edge after the last data-in pair.
  - DQS is required at T2 and T2n (nominal case) to register DM.
  - $D_{IN}^b$  = data-in for column b;  $D_{OUT}^n$  = data-out for column n.

**Figure 136: WRITE-to-READ – Odd Number of Data, Interrupting**


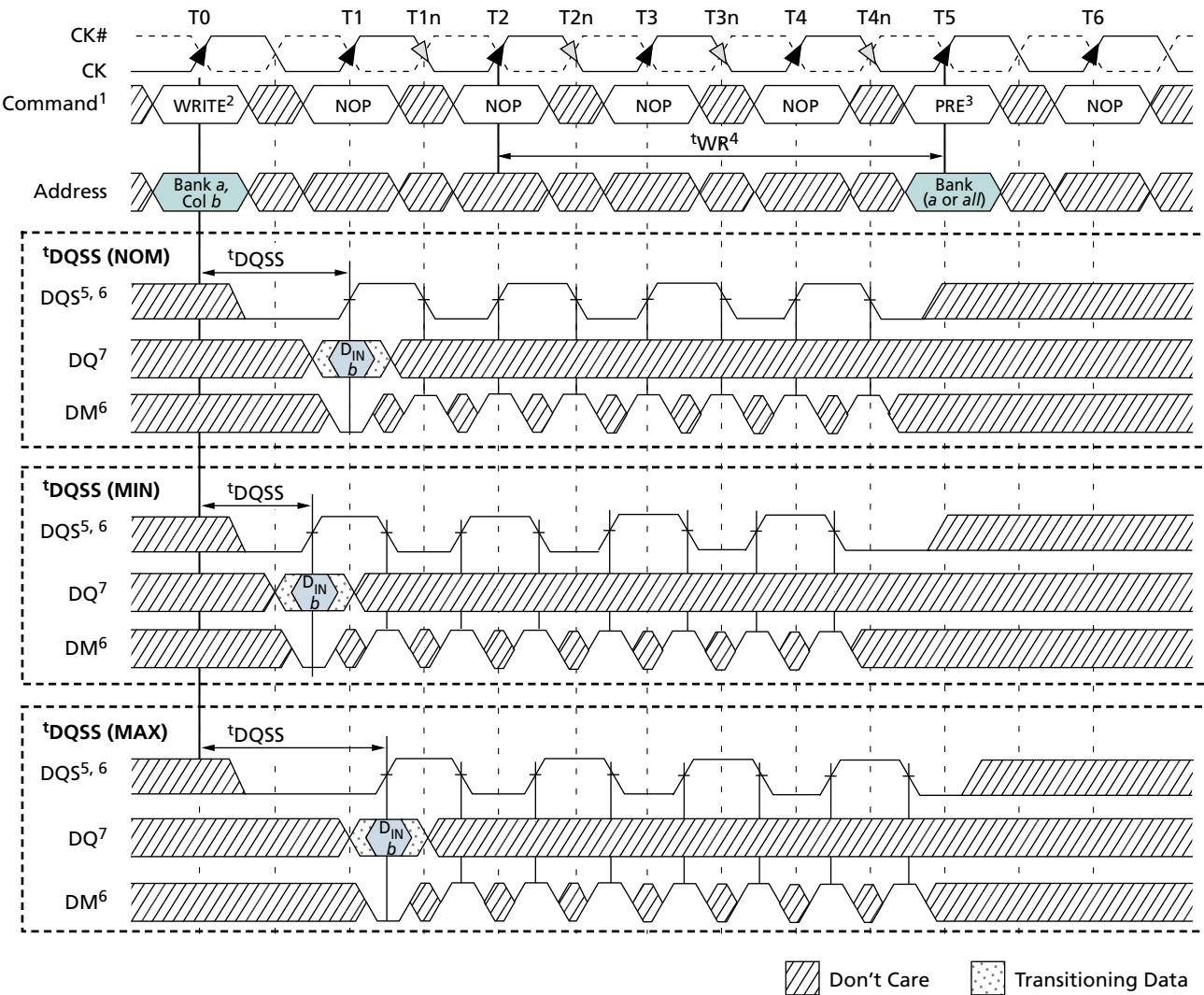
- Notes:
- An interrupted burst of 4 is shown; 1 data element is written, 3 are masked.
  - A10 is LOW with the WRITE command (auto precharge is disabled).
  - $t_{WTR}$  is referenced from the first positive CK edge after the last data-in pair.
  - DQS is required at T2 and T2n (nominal case) to register DM.
  - $D_{IN}^b$  = data-in for column b;  $D_{OUT}^n$  = data-out for column n.

**Figure 137: WRITE-to-PRECHARGE – Uninterrupting**

- Notes:
- An uninterrupted burst 4 of is shown.
  - A10 is LOW with the WRITE command (auto precharge is disabled).
  - PRE = PRECHARGE.
  - The PRECHARGE and WRITE commands are to the same device. However, the PRECHARGE and WRITE commands can be to different devices; in this case,  $t_{WR}$  is not required and the PRECHARGE command can be applied earlier.
  - $t_{WR}$  is referenced from the first positive CK edge after the last data-in pair.
  - $D_{IN}b$  = data-in for column b.

**Figure 138: WRITE-to-PRECHARGE – Interrupting**

- Notes:
- An interrupted burst of 8 is shown; two data elements are written.
  - A10 is LOW with the WRITE command (auto precharge is disabled).
  - PRE = PRECHARGE.
  - $t_{WR}$  is referenced from the first positive CK edge after the last data-in pair.
  - DQS is required at T4 and T4n to register DM.
  - $D_{IN}^b$  = data-in for column b.

**Figure 139: WRITE-to-PRECHARGE – Odd Number of Data, Interrupting**

- Notes:
1. An interrupted burst of 8 is shown; one data element is written.
  2. A10 is LOW with the WRITE command (auto precharge is disabled).
  3. PRE = PRECHARGE.
  4. tWR is referenced from the first positive CK edge after the last data-in pair.
  5. DQS is required at T4 and T4n to register DM.
  6. If a burst of 4 is used, DQS and DM are not required at T3, T3n, T4, and T4n.
  7. D<sub>IN</sub>b = data-in for column b.



## **PRECHARGE Operation**

The PRECHARGE command is used to deactivate the open row in a particular bank or the open row in all banks. The bank(s) will be available for a subsequent row access some specified time ( $t_{RP}$ ) after the PRECHARGE command is issued. Input A10 determines whether one or all banks will be precharged, and in the case where only one bank is precharged (A10 = LOW), inputs BA0 and BA1 select the bank. When all banks are precharged (A10 = HIGH), inputs BA0 and BA1 are treated as “Don’t Care.” After a bank has been precharged, it is in the idle state and must be activated prior to any READ or WRITE commands being issued to that bank. A PRECHARGE command will be treated as a NOP if there is no open row in that bank (idle state), or if the previously open row is already in the process of precharging.

## **Auto Precharge**

Auto precharge is a feature that performs the same individual bank PRECHARGE function described previously, without requiring an explicit command. This is accomplished by using A10 to enable auto precharge in conjunction with a specific READ or WRITE command. A precharge of the bank/row that is addressed with the READ or WRITE command is automatically performed upon completion of the READ or WRITE burst. Auto precharge is nonpersistent; it is either enabled or disabled for each individual READ or WRITE command.

Auto precharge ensures that the precharge is initiated at the earliest valid stage within a burst. This earliest valid stage is determined as if an explicit PRECHARGE command was issued at the earliest possible time without violating  $t_{RAS}$  (MIN), as described for each burst type in Table 53 (page 165). The READ with auto precharge enabled state or the WRITE with auto precharge enabled state can each be broken into two parts: the access period and the precharge period. The access period starts with registration of the command and ends where  $t_{RP}$  (the precharge period) begins. For READ with auto precharge, the precharge period is defined as if the same burst was executed with auto precharge disabled, followed by the earliest possible PRECHARGE command that still accesses all the data in the burst. For WRITE with auto precharge, the precharge period begins when  $t_{WR}$  ends, with  $t_{WR}$  measured as if auto precharge was disabled. In addition, during a WRITE with auto precharge, at least one clock is required during  $t_{WR}$  time. During the precharge period, the user must not issue another command to the same bank until  $t_{RP}$  is satisfied.

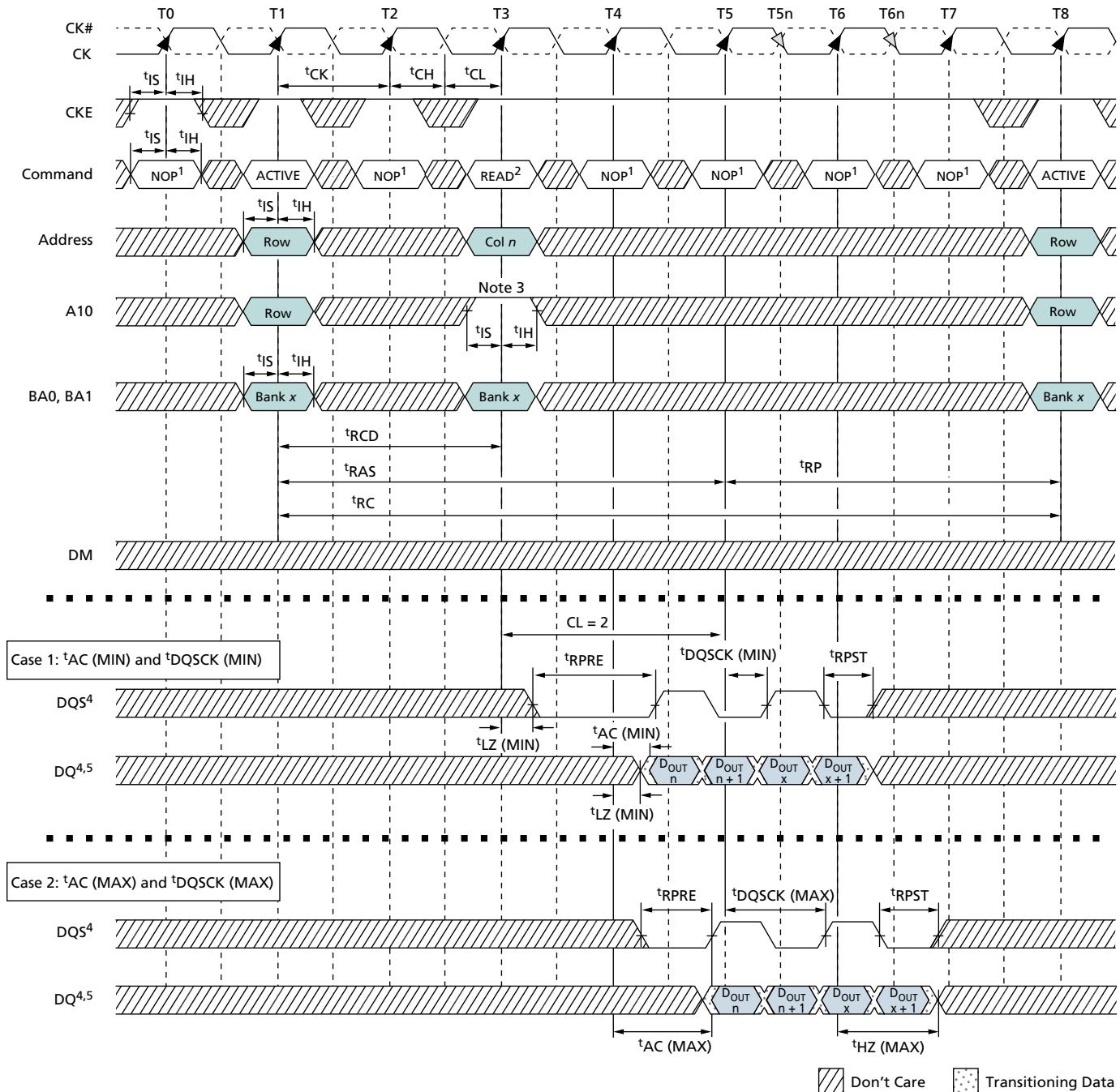
This device supports  $t_{RAS}$  lock-out. In the case of a single READ with auto precharge or single WRITE with auto precharge issued at  $t_{RCD}$  (MIN), the internal precharge will be delayed until  $t_{RAS}$  (MIN) has been satisfied.

Bank READ operations with and without auto precharge are shown in Figure 140 (page 206) and Figure 141 (page 207). Bank WRITE operations with and without auto precharge are shown in Figure 142 (page 208) and Figure 143 (page 209).

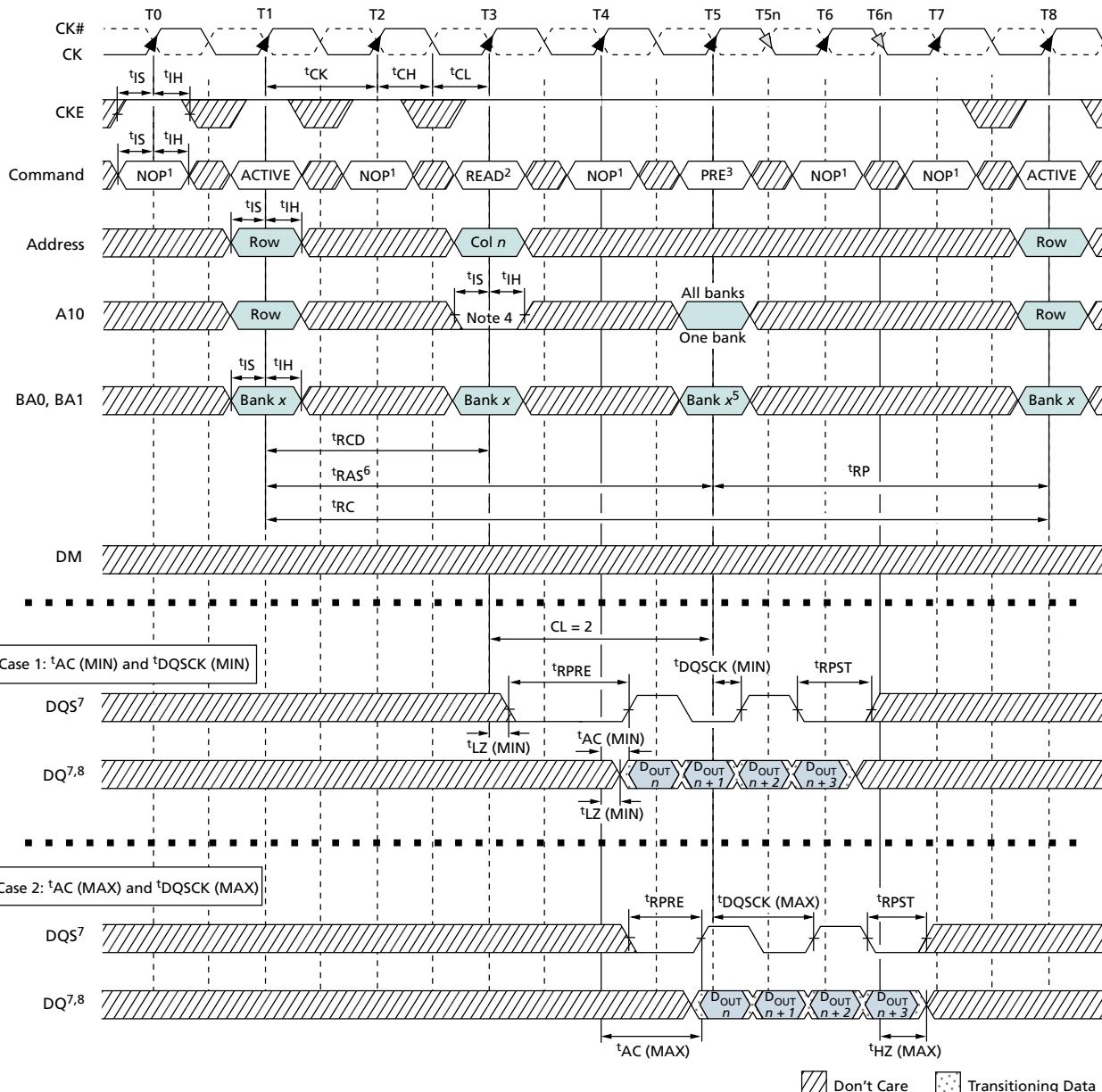


## Concurrent Auto Precharge

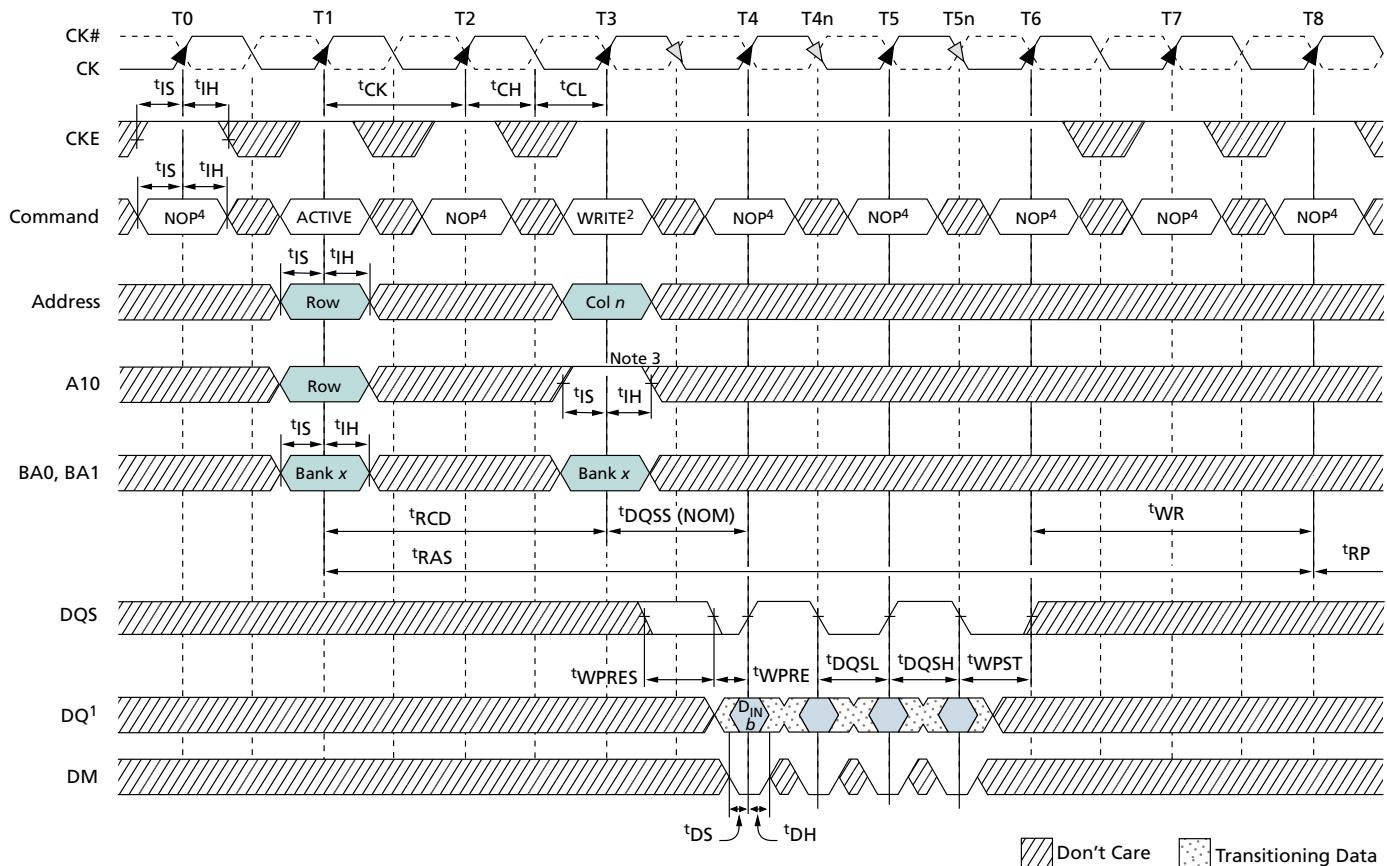
This device supports concurrent auto precharge such that when a READ with auto precharge is enabled or a WRITE with auto precharge is enabled, any command to another bank is supported, as long as that command does not interrupt the read or write data transfer already in process. This feature enables the precharge to complete in the bank in which the READ or WRITE with auto precharge was executed, without requiring an explicit PRECHARGE command, thus freeing the command bus for operations in other banks.

**Figure 140: Bank Read – With Auto Precharge**

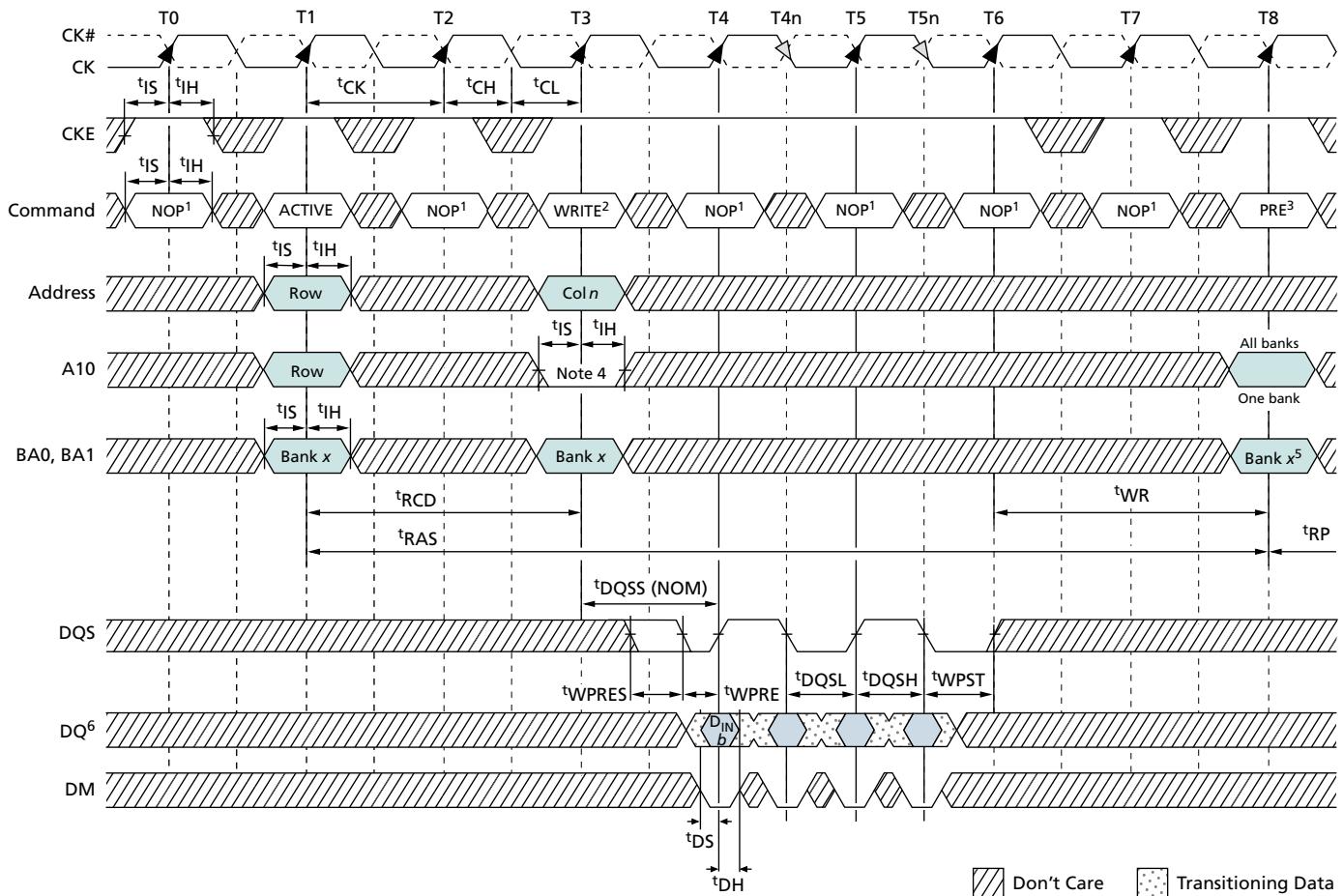
- Notes:
1. NOP commands are shown for ease of illustration; other commands may be valid at these times.
  2. BL = 4 in the case shown.
  3. Enable auto precharge.
  4. Refer to Figure 125 (page 189) and Figure 126 (page 190) for detailed DQS and DQ timing.
  5.  $D_{OUT} n$  = data-out from column  $n$ .

**Figure 141: Bank Read – Without Auto Precharge**

- Notes:
1. NOP commands are shown for ease of illustration; other commands may be valid at these times.
  2. BL = 4 in the case shown.
  3. PRE = PRECHARGE.
  4. Disable auto precharge.
  5. Bank x at T5 is "Don't Care" if A10 is HIGH at T5.
  6. The PRECHARGE command can only be applied at T5 if tRAS (MIN) is met.
  7. Refer to Figure 125 (page 189) and Figure 126 (page 190) for DQS and DQ timing details.
  8. D<sub>OUT</sub><sub>n</sub> = data out from column n.

**Figure 142: Bank Write – With Auto Precharge**

- Notes:
1. NOP commands are shown for ease of illustration; other commands may be valid at these times.
  2. BL = 4 in the case shown.
  3. Enable auto precharge.
  4.  $D_{INn}$  = data-out from column  $n$ .

**Figure 143: Bank Write – Without Auto Precharge**


- Notes:
1. NOP commands are shown for ease of illustration; other commands may be valid at these times.
  2. BL = 4 in the case shown.
  3. PRE = PRECHARGE.
  4. Disable auto precharge.
  5. Bank x at T8 is "Don't Care" if A10 is HIGH at T8.
  6. D<sub>OUT</sub><sup>n</sup> = data-out from column n.

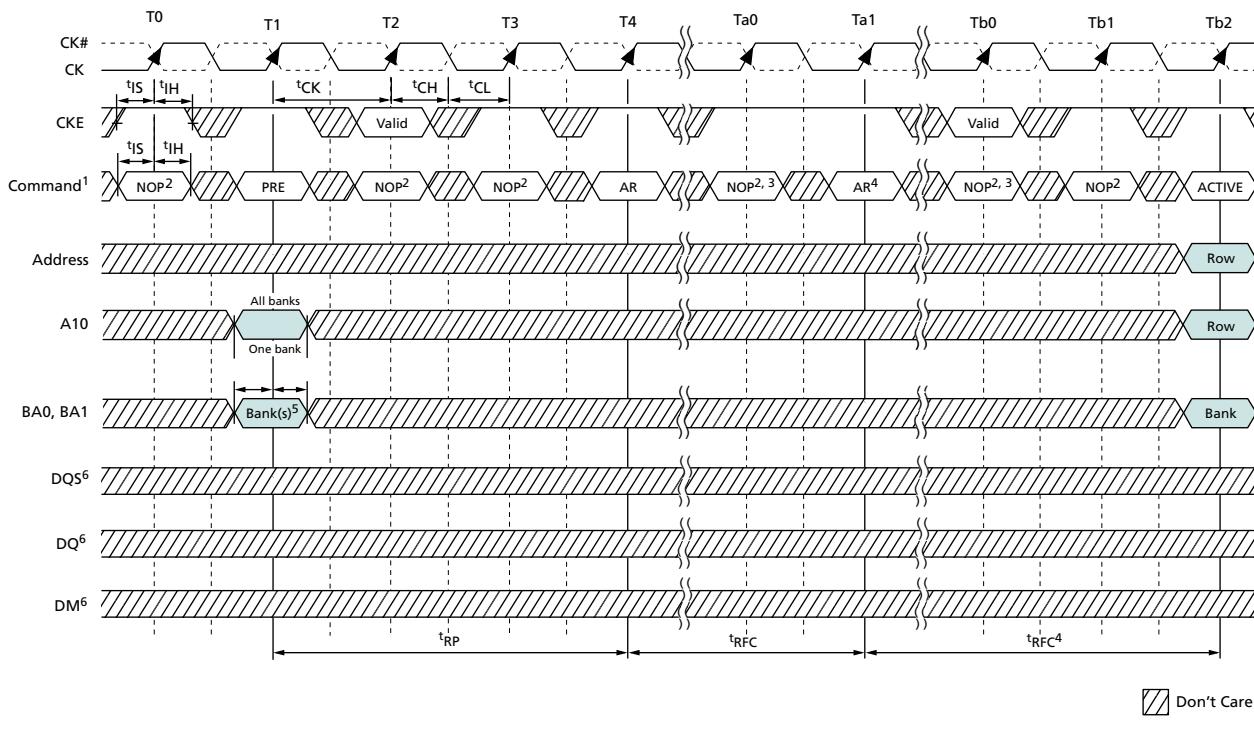
## AUTO REFRESH Operation

Auto refresh mode is used during normal operation of the device and is analogous to CAS#-BEFORE-RAS# (CBR) REFRESH in FPM/EDO DRAM. The AUTO REFRESH command is nonpersistent and must be issued each time a refresh is required.

The addressing is generated by the internal refresh controller. This makes the address bits a “Don’t Care” during an AUTO REFRESH command.

For improved efficiency in scheduling and switching between tasks, some flexibility in the absolute refresh interval is provided. The auto refresh period begins when the AUTO REFRESH command is registered and ends  $t_{RFC}$  later.

**Figure 144: Auto Refresh Mode**



- Notes:
1. PRE = PRECHARGE; AR = AUTO REFRESH.
  2. NOP commands are shown for ease of illustration; other commands may be valid during this time. CKE must be active during clock positive transitions.
  3. NOP or COMMAND INHIBIT are the only commands supported until after  $t_{RFC}$  time; CKE must be active during clock positive transitions.
  4. The second AUTO REFRESH is not required and is only shown as an example of two back-to-back AUTO REFRESH commands.
  5. Bank x at T1 is “Don’t Care” if A10 is HIGH at this point; A10 must be HIGH if more than one bank is active (for example, must precharge all active banks).
  6. DM, DQ, and DQS signals are all “Don’t Care”/High-Z for operations shown.

Although it is not a JEDEC requirement, CKE must be active (HIGH) during the auto refresh period to provide support for future functional features. The auto refresh period begins when the AUTO REFRESH command is registered and ends  $t_{RFC}$  later.



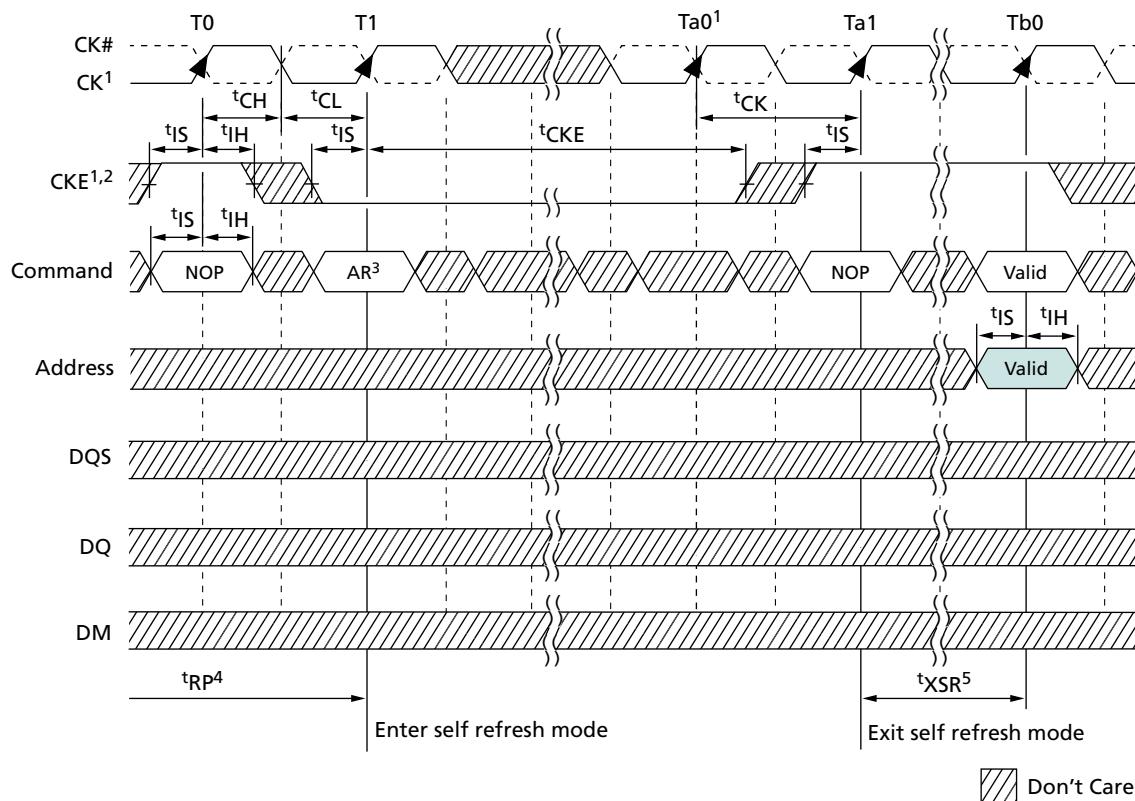
## SELF REFRESH Operation

The SELF REFRESH command can be used to retain data in the device while the rest of the system is powered down. When in self refresh mode, the device retains data without external clocking. The SELF REFRESH command is initiated like an AUTO REFRESH command, except that CKE is disabled (LOW). All command and address input signals except CKE are “Don’t Care” during self refresh.

During self refresh, the device is refreshed as defined in the extended mode register. (see Partial-Array Self Refresh (page 177).) An internal temperature sensor adjusts the refresh rate to optimize device power consumption while ensuring data integrity. (See Temperature-Compensated Self Refresh (page 176).)

The procedure for exiting self refresh requires a sequence of commands. First, CK must be stable prior to CKE going HIGH. When CKE is HIGH, the device must have NOP commands issued for tXSR to complete any internal refresh already in progress.

During SELF REFRESH operation, refresh intervals are scheduled internally and may vary. These refresh intervals may differ from the specified tREFI time. For this reason, the SELF REFRESH command must not be used as a substitute for the AUTO REFRESH command.

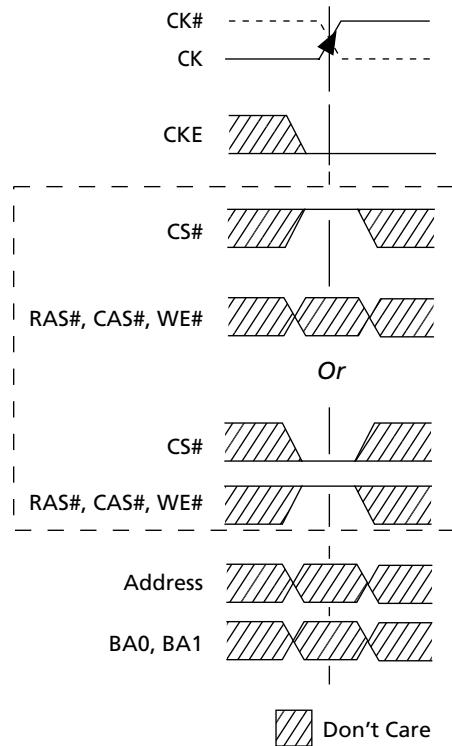
**Figure 145: Self Refresh Mode**

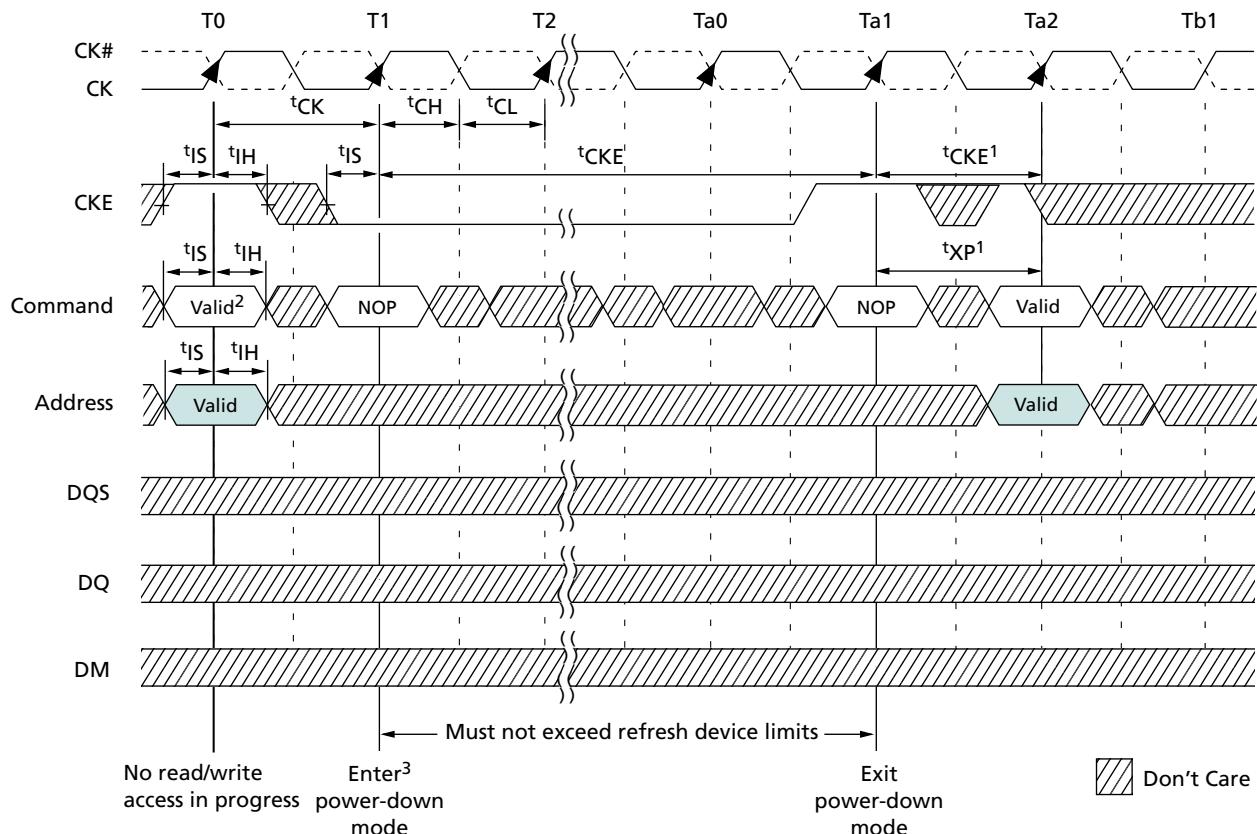
- Notes:**
1. Clock must be stable, cycling within specifications by Ta0, before exiting self refresh mode.
  2. CKE must remain LOW to remain in self refresh.
  3. AR = AUTO REFRESH.
  4. Device must be in the all banks idle state prior to entering self refresh mode.
  5. Either a NOP or DESELECT command is required for t<sub>XSR</sub> time with at least two clock pulses.

## Power-Down

Power-down is entered when CKE is registered LOW. If power-down occurs when all banks are idle, this mode is referred to as precharge power-down; if power-down occurs when there is a row active in any bank, this mode is referred to as active power-down. Entering power-down deactivates all input and output buffers, including CK and CK# and excluding CKE. Exiting power-down requires the device to be at the same voltage as when it entered power-down and received a stable clock. Note that the power-down duration is limited by the refresh requirements of the device.

When in power-down, CKE LOW must be maintained at the inputs of the device, while all other input signals are “Don’t Care.” The power-down state is synchronously exited when CKE is registered HIGH (in conjunction with a NOP or DESELECT command). NOP or DESELECT commands must be maintained on the command bus until t<sub>XP</sub> is satisfied. See Figure 147 (page 214) for a detailed illustration of power-down mode.

**Figure 146: Power-Down Entry (in Active or Precharge Mode)**

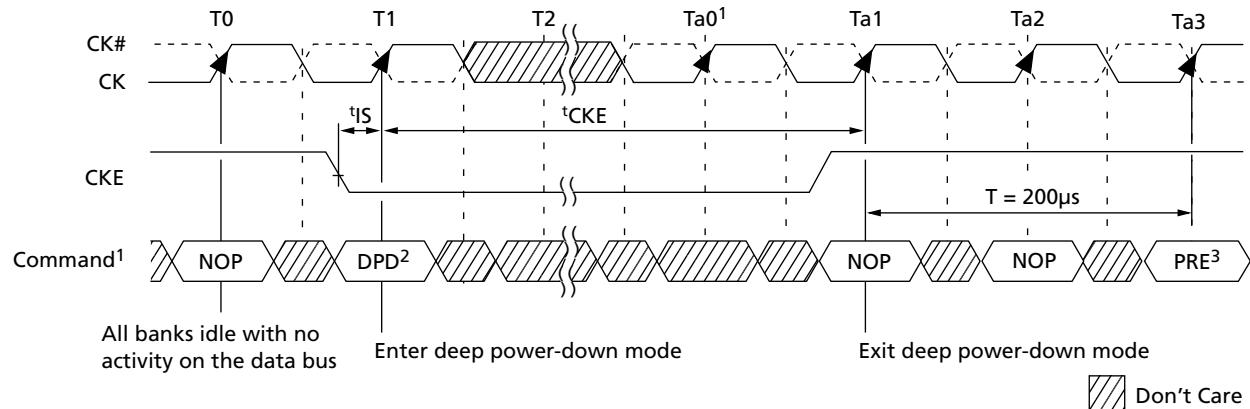
**Figure 147: Power-Down Mode (Active or Precharge)**

- Notes:
1.  $t_{CKE}$  applies if CKE goes LOW at  $Ta2$  (entering power-down);  $t_{XP}$  applies if CKE remains HIGH at  $Ta2$  (exit power-down).
  2. If this command is a PRECHARGE (or if the device is already in the idle state), then the power-down mode shown is precharge power-down. If this command is an ACTIVE (or if at least 1 row is already active), then the power-down mode shown is active power-down.
  3. No column accesses can be in progress when power-down is entered.

## Deep Power-Down

Deep power-down (DPD) is an operating mode used to achieve maximum power reduction by eliminating power to the memory array. Data will not be retained after the device enters DPD mode.

Before entering DPD mode the device must be in the all banks idle state with no activity on the data bus ( $t_{RP}$  time must be met). DPD mode is entered by holding CS# and WE# LOW with RAS# and CAS# HIGH at the rising edge of the clock while CKE is LOW. CKE must be held LOW to maintain DPD mode. The clock must be stable prior to exiting DPD mode. To exit DPD mode, assert CKE HIGH with either a NOP or DESELECT command present on the command bus. After exiting DPD mode, a full DRAM initialization sequence is required.

**Figure 148: Deep Power-Down Mode**

- Notes:
1. Clock must be stable prior to CKE going HIGH.
  2. DPD = deep power-down.
  3. Upon exit of deep power-down mode, a full DRAM initialization sequence is required.

## Clock Change Frequency

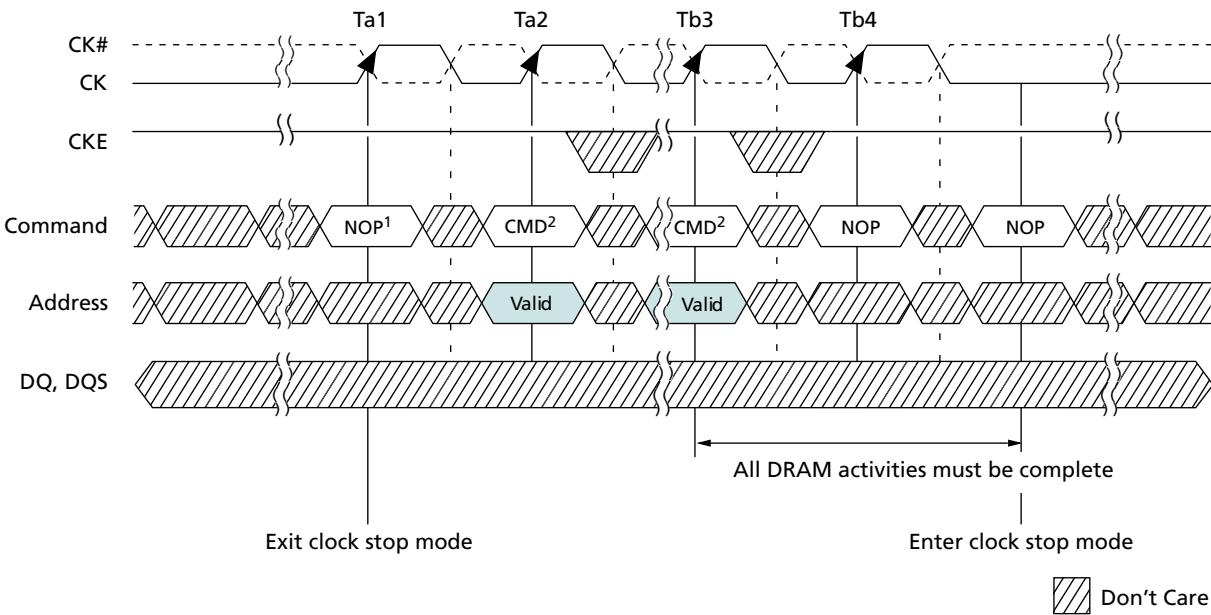
One method of controlling the power efficiency in applications is to throttle the clock that controls the device. The clock can be controlled by changing the clock frequency or stopping the clock.

The device enables the clock to change frequency during operation only if all timing parameters are met and all refresh requirements are satisfied.

The clock can be stopped altogether if there are no DRAM operations in progress that would be affected by this change. Any DRAM operation already in process must be completed before entering clock stop mode; this includes the following timings:  $t_{RCD}$ ,  $t_{RP}$ ,  $t_{RFC}$ ,  $t_{MRD}$ ,  $t_{WR}$ , and  $t_{RPST}$ . In addition, any READ or WRITE burst in progress must be complete. (see READ Operation (page 181), and WRITE Operation (page 192).)

$CKE$  must be held HIGH with  $CK = LOW$  and  $CK\# = HIGH$  for the full duration of the clock stop mode. One clock cycle and at least one NOP or DESELECT is required after the clock is restarted before a valid command can be issued.

**Figure 149: Clock Stop Mode**



- Notes:
- Prior to  $Ta1$ , the device is in clock stop mode. To exit, at least one NOP is required before issuing any valid command.
  - Any valid command is supported; device is not in clock suspend mode.



## Revision History

### Rev. H, Production – 3/11

- Production status

### Rev. G, Preliminary – 6/10

- 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
- Replaced blank with 3 for number of valid address cycles on Block Erase Two-Plane in the Two-Plane Command Set Table
- Updated AT Idd tables
- Added reduced page mode

### Rev. F, Preliminary – 3/10

- Changed OTP DATA LOCK BY PAGE to OTP DATA LOCK BY BLOCK in Command Set.
- Updated Status Register Definition .
- Updated OTP section to show OTP protect by BLOCK, rather than by PAGE.
- Added new notes 1 and 2 to DC Characteristics and Operating Conditions (1.8V) and changed Typ value from 20 to 10 and Max value from 40 to 20 for I<sub>CC1</sub>, I<sub>CC2</sub>, and I<sub>CC3</sub>.

### Rev. E, Preliminary – 2/10

- Add part number to first page: MT29C4G48MAYAPAKQ-5 IT
- Added "x8" to 168-Ball VFBGA (NAND x8, x16; LPDDR x32) Ball Assignments and x8, x16 NAND Ball Descriptions titles.
- Added "I/O[15:8] are RFU for x8 NAND devices." for I/O[15:0] description in x8, x16 NAND Ball Descriptions.

### Rev. D, Preliminary – 1/10

- Modified Electrical Specifications – IDD Parameters: Changed I<sub>DD2P</sub>, I<sub>DD2PS</sub>, I<sub>DD3P</sub>, I<sub>DD3PS</sub>, and I<sub>DD6</sub> values.
- Removed boot block.

### Rev. C, Preliminary – 12/09

- 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


**Rev. B, Preliminary – 10/09**

- Added 4 new MPG part numbers.
- Removed x16 LPDDR ball assignments drawing and removed x16-specific LPDDR values from x32 LPDDR Ball Descriptions.
- Removed part numbers MT29F4G08ABBDAWP and MT29F4G16ABBDAWP.
- Changed  $tR$  to  $tR_{ECC}$  in INTERNAL DATA MOVE (85h-10h) with Internal ECC Enabled.
- Changed  $tR$  to  $tR_{ECC}$  in INTERNAL DATA MOVE (85h-10h) with Random Data Input with Internal ECC Enabled.
- Updated Boot Block Operation to include dual-plane restrictions.
- Added  $tRCBSY$  spec to Electrical Specifications – Program/Erase Characteristics.
- Added note for  $tPROG$  and  $tPROG_{ECC}$  specifications to Electrical Specifications – Program/Erase Characteristics.
- Moved note from  $tRHW$  to  $tRHZ$  in AC Characteristics: Normal Operation (3.3V).

**Rev. A, Preliminary – 7/09**

- Initial release.

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900  
[www.micron.com/productsupport](http://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.