# Flash

# 2 Gbit (256M x 8) 1.8V NAND Flash Memory

## FEATURES

- Voltage Supply: 1.8V (1.7V to 1.95V)
- Open NAND Flash Interface (ONFI) 1.0-compliant1
- Single-level cell (SLC) technology
- Organization
  - Page size: 2176 bytes (2048 + 128 bytes)
  - Block size: 64 pages (128K + 8K bytes)
  - Plane size: 2 planes x 1024 blocks per plane
  - Device size: 2048 blocks
- Asynchronous I/O performance
  - tRC/ tWC: 30ns
- Array performance
  - Read page: 30us
  - Program page: 200us (TYP)
  - Erase block: 2ms(TYP)
- Command set: ONFI NAND Flash Protocol
- Advanced command set
  - Program page cache mode
  - Read page cache mode
  - One-time programmable (OTP) mode
  - Programmable drive strength
  - Two-plane commands (Available with ECC off only)
  - Read unique ID
  - Internal data move
- Operation status byte provides software method for detecting
  - Operation completion
  - Pass/ fail condition
  - Write-protect status
- Ready/ Busy# (R/B#) provides a hardware method of detecting operation completion
- WP#: Write protect entire device
- Block 0 is valid when shipped from factory with ECC. For minimum required ECC, see Error Management.RESET(FFh) required as first command after power-on
- RESET (FFh) required as first command after power- on
- Internal data move operations supported within the plane from which data is read
- Quality and reliability
  - Data retention: JESD47G-compliant
  - Endurance: 100,000 PROGRAM/ERASE cycles
  - Additional: Uncycled data retention: 10 years
- Operating temperature
  - Industrial: -40°C to +85°C

Notes: 1. The ONFI 1.0 specification is available at www.onfi.org.

## ORDERING INFORMATION

| Product ID | Speed | Package | Comments |
|---|---|---|---|
| F59D2G81XA-45TIG2B | 45 ns | 48 pin TSOPI | Pb-free |
| F59D2G81XA-45BIG2B | 45 ns | 63 ball BGA | Pb-free |

## GENERAL DESCRIPTION

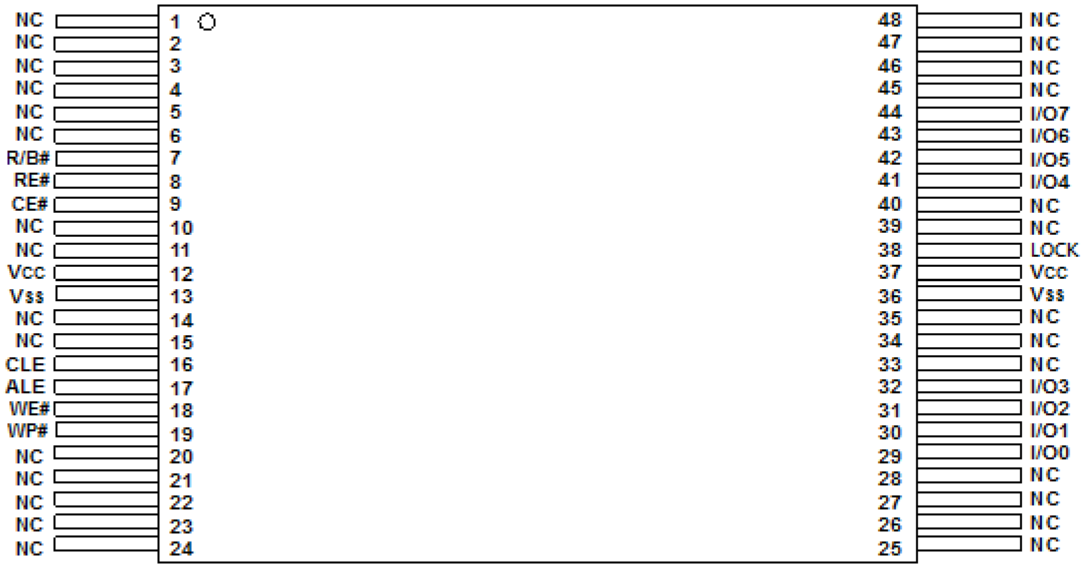
NAND Flash devices include an asynchronous data interface for high-performance I/O operations. These devices use a highly multiplexed 8-bit bus (I/Ox) to transfer commands, address, and data. There are five control signals used to implement the asynchronous data interface: CE#, CLE, ALE, WE#, and RE#.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

## PIN CONFIGURATION (TOP VIEW)

(TSOPI 48L, 12mm X 20mm Body, 0.5mm Pin Pitch)



## BALL CONFIGURATION (x8) (TOP VIEW)

(BGA 63 BALL, 9mm X 11mm Body, 0.8 Ball Pitch)

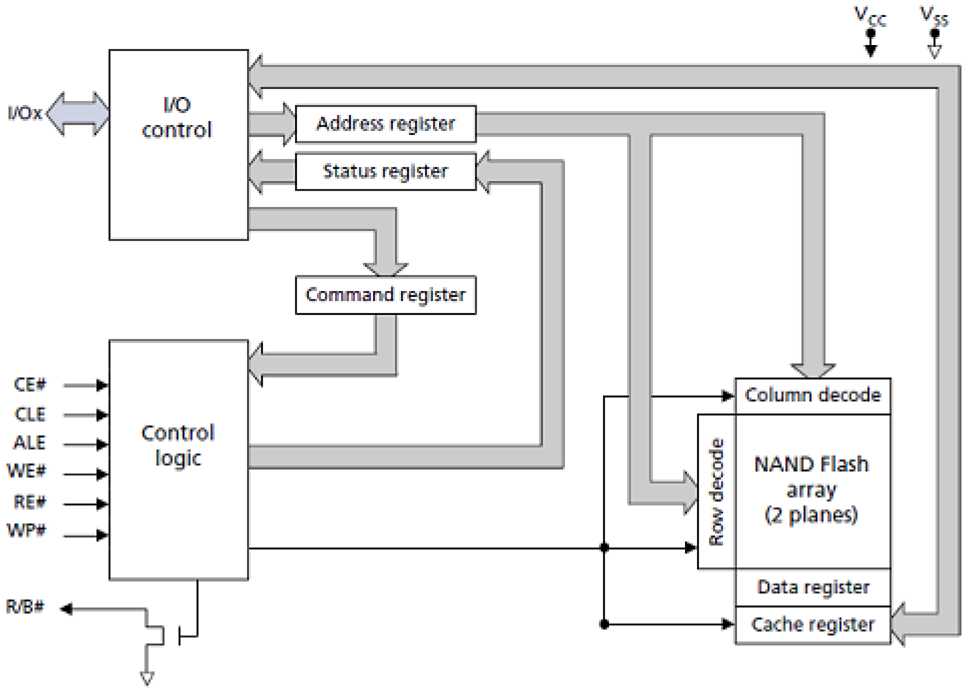
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| A | NC | NC | | | | | | | NC | NC |
| B | NC | | | | | | | | NC | NC |
| C | | | WP# | ALE | Vss | CE# | WE# | R /B# | | |
| D | | | NC | RE# | CLE | NC | NC | NC | | |
| E | | | NC | NC | NC | NC | NC | NC | | |
| F | | | NC | NC | NC | NC | NC | NC | | |
| G | | | NC | NC | NC | NC | NC | NC | | |
| H | | | NC | I/O0 | NC | NC | NC | Vcc | | |
| J | | | NC | I/O1 | NC | Vcc | I/O5 | I/O7 | | |
| K | | | Vss | I/O2 | I/O3 | I/O4 | I/O6 | Vss | | |
| L | NC | NC | | | | | | | NC | NC |
| M | NC | NC | | | | | | | NC | NC |

## PIN NAMES

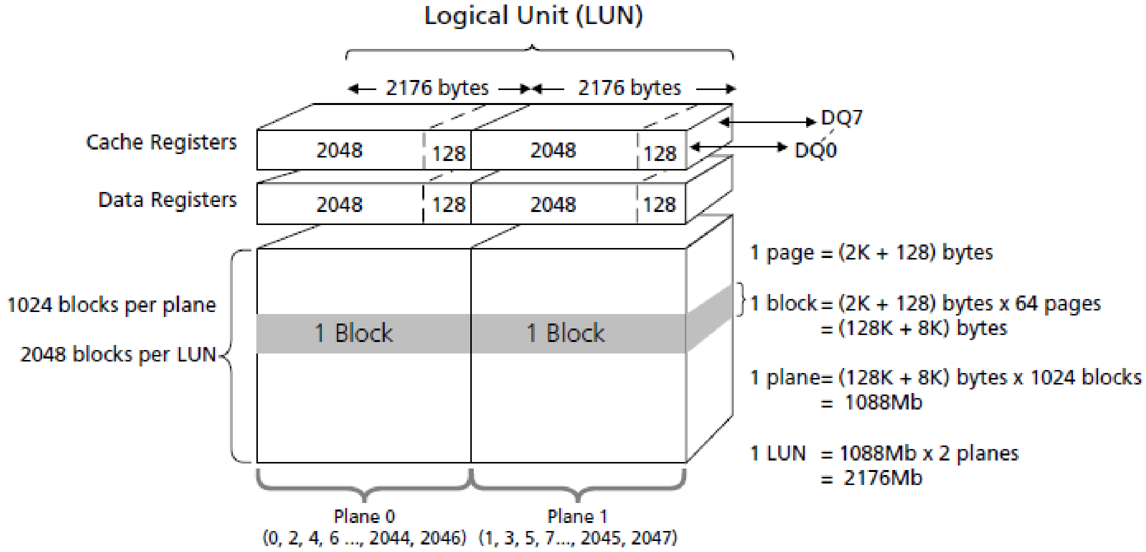
| Pin Name | Type | Function |
|---|---|---|
| $V_{CC}$ | Supply | NAND Power Supply |
| $V_{SS}$ | Supply | Ground |
| I/O1 to I/O7 | Input/output | Data inputs/outputs: The I/O1 to 7 pins are used as a port for transferring address, command and input/output data to and from the device. |
| ALE | Input | Address latch enable: The ALE signal is used to control loading address information into the internal address register. Address information is latched into the address register from the I/O port on the rising edge of WE# while ALE is High. |
| CLE | Input | Command latch enable: The CLE input signal is used to control loading of the operation mode command into the internal command register. The command is latched into the command register from the I/O port on the rising edge of the WE# signal while CLE is High. |
| CE# | Input | Chip enable: The device goes into a low-power Standby mode when CE# goes High during the device is in Ready state. The CE# signal is ignored when device is in Busy state (R/B# = L), such as during a Program or Erase or Read operation, and will not enter Standby mode even if the CE# input goes High. |
| RE# | Input | Read enable: The RE# signal controls serial data output. Data is available $t_{REA}$ after the falling edge of RE#. The internal column address counter is also incremented (Address = Address + l) on this falling edge. |
| WE# | Input | Write enable: The WE# signal is used to control the acquisition of data from the I/O port. |
| WP# | Input | Write protect: The WP# signal is used to protect the device from accidental programming or erasing. The internal voltage regulator is reset when WP# is Low. This signal is usually used for protecting the data during the power-on/off sequence when input signals are invalid. |
| R/B# | Output | Ready/busy: The R/B# output signal is used to indicate the operating condition of the device. The R/B# signal is in Busy state ( R/B# = L) during the Program, Erase and Read operations and will return to Ready state (R/B# = H) after completion of the operation. The output buffer for this signal is an open drain and has to be pulled-up to $V_{CC}$ with an appropriate resister. If R/B# signal is not pulled-up to $V_{CC}$ ("Open" state), device operation can not guarantee. |
| NC | - | No connect: NCs are not internally connected. They can be driven or left unconnected. |
| DNU | - | Do not use: DNUs must be left unconnected. |

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

## Functional Block Diagram




## Device and Array Organization




## Address Cycle Map

| | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0 |
|---|---|---|---|---|---|---|---|---|
| **1st cycle** | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 |
| **2nd cycle** | LOW | LOW | LOW | LOW | CA11 | CA10 | CA9 | CA8 |
| **3rd cycle** | BA7 | BA6 | PA5 | PA4 | PA3 | PA2 | PA1 | PA0 |
| **4th cycle** | BA15 | BA14 | BA13 | BA12 | BA11 | BA10 | BA9 | BA8 |
| **5th cycle** | LOW | LOW | LOW | LOW | LOW | LOW | LOW | BA16 |

**NOTE:**
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[10:7] must be 0.
3. BA6 controls plane selection.

## 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, 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.

## Command Set

| Command | Command 1st Cycle | Number of Valid Address Cycles | Data Input Cycles | Command 2nd Cycle | Valid While Selected LUN is Busy | Valid While Other LUNs are Busy [2] | Notes |
|---|---|---|---|---|---|---|---|
| **Reset Operations** | | | | | | | |
| RESET | FFh | 0 | - | - | Yes | Yes | |
| **Identification Operation** | | | | | | | |
| READ ID | 90h | 1 | - | - | No | No | |
| READ PARAMETER PAGE | ECh | 1 | - | - | No | No | |
| READ UNIQUE ID | EDh | 1 | - | - | No | No | |
| **Feature Operations** | | | | | | | |
| GET FEATURES | EEh | 1 | - | - | No | No | |
| SET FEATURES | EFh | 1 | 4 | - | No | No | |
| **Status Operations** | | | | | | | |
| READ STATUS | 70h | 0 | - | - | Yes | | |
| READ STATUS ENHANCED | 78h | 3 | - | - | Yes | Yes | 2 |
| **Column Address Operations** | | | | | | | |
| 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 |
| **Read Operations** | | | | | | | |
| READ MODE | 00h | 0 | - | - | No | Yes | |
| READ PAGE | 00h | 5 | - | 30h | No | Yes | |
| READ PAGE CACHE SEQUENTIAL | 31h | 0 | - | - | No | Yes | 4 |
| READ PAGE CACHE RANDOM | 00h | 5 | - | 31h | No | Yes | 4 |
| READ PAGE CACHE LAST | 3Fh | 0 | - | - | No | Yes | 4 |

## Command Set (Continued)

| Command | Command 1st Cycle | Number of Valid Address Cycles | Data Input Cycles | Command 2nd Cycle | Valid While Selected LUN is Busy | Valid While Other LUNs are Busy [2] | Notes |
|---|---|---|---|---|---|---|---|
| **Program Operations** | | | | | | | |
| PROGRAM PAGE | 80h | 5 | Yes | 10h | No | Yes | 2 |
| PROGRAM PAGE CACHE | 80h | 5 | Yes | 15h | No | Yes | 2,5 |
| **Erase Operations** | | | | | | | |
| ERASE BLOCK | 60h | 3 | - | D0h | No | Yes | |
| **Internal Data Move Operations** | | | | | | | |
| READ FOR INTERNAL DATA MOVE | 00h | 5 | - | 35h | No | Yes | 3 |
| PROGRAM FOR INTERNAL DATA MOVE | 85h | 5 | Optional | 10h | No | Yes | |
| **One-Time Programmable (OTP) Operations** | | | | | | | |
| OTP DATA LOCK BY BLOCK (ONFI) | 80h | 5 | No | 10h | No | No | 6 |
| OTP DATA PROGRAM (ONFI) | 80h | 5 | Yes | 10h | No | No | 6 |
| OTP DATA READ (ONFI) | 00h | 5 | No | 30h | No | No | 6 |

**NOTE:**
1. Busy means RDY=0.
2. These commands can be used for interleaved die (multi-LUN) operations (see Interleaved Die Multi-LUN Operations).
3. Do not cross plane address boundaries when using READ FOR INTERNAL DATA MOVE and PROGRAM FOR INTERNAL DATA MOVE.
4. 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.
5. 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.
6. OTP commands can be entered only after issuing the SET FEATURES command with the feature address.

## Two-Plane Command Set

| Command | Command 1st Cycle | Number of Valid Address Cycles | Command 2nd Cycle | Number of Valid Address Cycles | Command 3rd Cycle | Valid While Selected LUN is Busy | Valid While Other LUNs are Busy | Notes |
|---|---|---|---|---|---|---|---|---|
| READ PAGE TWO-PLANE | 00h | 5 | 00h | 5 | 30h | No | Yes | |
| READ FOR INTERNAL DATA MOVE TWO-PLANE | 00h | 5 | 00h | 5 | 35h | No | Yes | 1 |
| RANDOM DATA READ TWO-PLANE | 06h | 5 | E0h | - | - | No | Yes | 2 |
| 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 INTERNAL DATA MOVE TWO-PLANE | 85h | 5 | 11h-85h | 5 | 10h | No | Yes | 1 |
| ERASE BLOCK TWO-PLANE | 60h | 3 | D1h-60h | 3 | D0h | No | Yes | 3 |

**NOTE:**
1. Do not cross plane boundaries when using READ FOR INTERNAL DATA MOVE TWO-PLANE or PROGRAM FOR INTERNAL DATA MOVE TWO-PLANE.
2. The RANDOM DATA READ TWO-PLANE command is limited to use with the READ PAGE TWO-PLANE command.
3. D1h command can be omitted.

## 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.

### Absolute Maximum Ratings

Voltage on any pin relative to VSS

| Parameter | Symbol | Min | Max | Unit |
|---|---|---|---|---|
| Voltage input | VIN | -0.6 | +2.4 | V |
| VCC supply voltage | VCC | -0.6 | +2.4 | V |
| Storage Temperature | $T_{STG}$ | -65 | +150 | ℃ |
| Short circuit output current | $I_{OS}$ | - | 5 | mA |

### Recommended Operating Condition

(Voltage reference to GND, $T_A$ = -40 to 85℃)

| Parameter/Condition | Symbol | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| Supply Voltage | VCC | 1.7 | 1.8 | 1.95 | V |
| Supply Voltage | VSS | 0 | 0 | 0 | V |

## DC and Operation Characteristics

| Parameter | Symbol | Test Conditions | Min. | Typ. | Max. | Unit | Notes |
|---|---|---|---|---|---|---|---|
| Sequential READ current | $I_{CC1}$ | $t_{RC}$ = $t_{RC}$ (MIN); CE# = $V_{IL}$; $I_{OUT}$ = 0mA | - | 13 | 20 | mA | 1,2 |
| PROGRAM current | $I_{CC2}$ | - | - | 13 | 20 | mA | 1,2 |
| ERASE current | $I_{CC3}$ | - | - | 15 | 20 | mA | 1,2 |
| Stand-by Current (TTL) | $I_{SB1}$ | CE# =$V_{IH}$; WP# =0V/VCC | - | - | 1 | mA | |
| Stand-by Current (CMOS) | $I_{SB2}$ | CE# = VCC-0.2V; WP# =0V/VCC | - | 15 | 50 | uA | |
| Staggered power-up current | $I_{ST}$ | Rise time=1ms, Line capacitance=0.1uF | - | - | 10 per die | mA | 3 |
| Input Leakage Current | $I_{LI}$ | VIN=0 to VCC | - | - | ±10 | uA | |
| Output Leakage Current | $I_{LO}$ | VOUT=0 to VCC | - | - | ±10 | uA | |
| Input High Voltage | $V_{IH}$ | I/O[7:0], I/O[15:0], CE#, CLE, ALE, WE#, RE#, WP#, R/B# | 0.8 x VCC | - | VCC+0.3 | V | |
| Input Low Voltage, All inputs | $V_{IL}$ | - | -0.3 | - | 0.2 x VCC | V | |
| Output High Voltage Level | $V_{OH}$ | $I_{OH}$= -100uA | VCC - 0.2 | - | - | V | 4 |
| Output Low Voltage Level | $V_{OL}$ | $I_{OL}$= -100uA | - | - | 0.2 | V | 4 |
| Output Low Current | $I_{OL}$ (R /B#) | $V_{OL}$= 0.2V | 3 | 4 | - | mA | 5 |

**NOTE:**
1. Typical and maximum values are for single-plane operation only. 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 VCC reaches VCC(MIN).
4. Test conditions for VOH and VOL.
5. DC characteristics may need to be relaxed if R/B# pull-down strength is not set to full.

## Valid Block

Note 1 applies to all

| Symbol | Min. | Max. | Unit | Notes |
|---|---|---|---|---|
| $N_{VB}$ | 2008 | 2048 | Blocks | 2 |

**NOTE:**

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.

## AC Test Condition

($T_A$= -40 to 85 ℃, VCC= 1.7V~1.95V)

| Parameter | Condition |
|---|---|
| Input Pulse Levels | 0V to VCC |
| Input Rise and Fall Times | 5 ns |
| Input and Output Timing Levels | VCC /2 |
| Output Load* | 1 TTL Gate and $C_L$=30pF |

**NOTE:**

1. Verified in device characterization, not 100% tested.

## Capacitance

($T_A$=25℃, Vin=0V, f=1.0MHz)

| Item | Symbol | Max. | Unit | Notes |
|---|---|---|---|---|
| Input / Output Capacitance | $C_{I/O}$ | 8 | pF | 1,2 |
| Input Capacitance | $C_{IN}$ | 6 | pF | 1,2 |

**NOTE:**

1. These parameters are verified in device characterization and are not 100% tested.

## Program / Erase Characteristics

| Parameter | Symbol | Min. | Typ. | Max. | Unit | Notes |
|---|---|---|---|---|---|---|
| Number of partial-page programs | NOP | - | - | 4 | Cycle | 1 |
| BLOCK ERASE operation time | $t_{BERS}$ | - | 2 | 10 | | 2 |
| Busy time for PROGRAM CACHE operation | $t_{CBSY}$ | - | 7 | 600 | us | 3 |
| Cache read busy time | $t_{RCBSY}$ | - | 5 | 25 | us | |
| Busy time for SET FEATURES and GET FEATURES operations | $t_{FEAT}$ | - | - | 1 | us | |
| LAST PAGE PROGRAM operation time | $t_{LPROG}$ | - | - | - | - | 4 |
| PROGRAM PAGE operation time | $t_{PROG}$ | - | 200 | 600 | us | 2 |
| Power-on reset time | $t_{POR}$ | - | - | 1 | ms | |
| Busy time for OTP DATA PROGRAM operation if OTP is protected | $t_{OBSY}$ | - | - | 50 | us | |
| READ PAGE operation time | $t_R$ | - | - | 30 | us | |
| Busy time for TWO-PLANE PROGRAM PAGE or TWO-PLANE BLOCK ERASE operation | $t_{DBSY}$ | | 0.5 | 1 | us | |

**NOTE:**

1. Four total partial-page programs to the same page.
2. Typical $t_{PROG}$ and $t_{BERS}$ time may increase for two-plane operations.
3. $t_{CBSY}$ MAX time depends on timing between internal program completion and data-in.
4. $t_{LPROG}$ = $t_{PROG}$ (last page) + $t_{PROG}$ (last - 1 page) - command load time (last page) -address load time (last page) - data load time (last page).

## AC Characteristics for Command / Address / Data Input

| Parameter | Symbol | Min. | Max. | Unit | Notes |
|---|---|---|---|---|---|
| ALE to data start | $t_{ADL}$ | 100 | - | ns | 1 |
| 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 | $t_{CS}$ | 25 | - | ns | |
| Data hold time | $t_{DH}$ | 5 | - | ns | |
| Data setup time | $t_{DS}$ | 10 | - | ns | |
| WRITE cycle time | $t_{WC}$ | 30 | - | ns | 1 |
| WE# pulse width HIGH | $t_{WH}$ | 10 | - | ns | 1 |
| WE# pulse width | $t_{WP}$ | 15 | - | ns | 1 |
| WP# transition to WE# LOW | $t_{WW}$ | 100 | | ns | |

**NOTE:**

1. 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.

## AC Characteristics for Normal Operation[1]

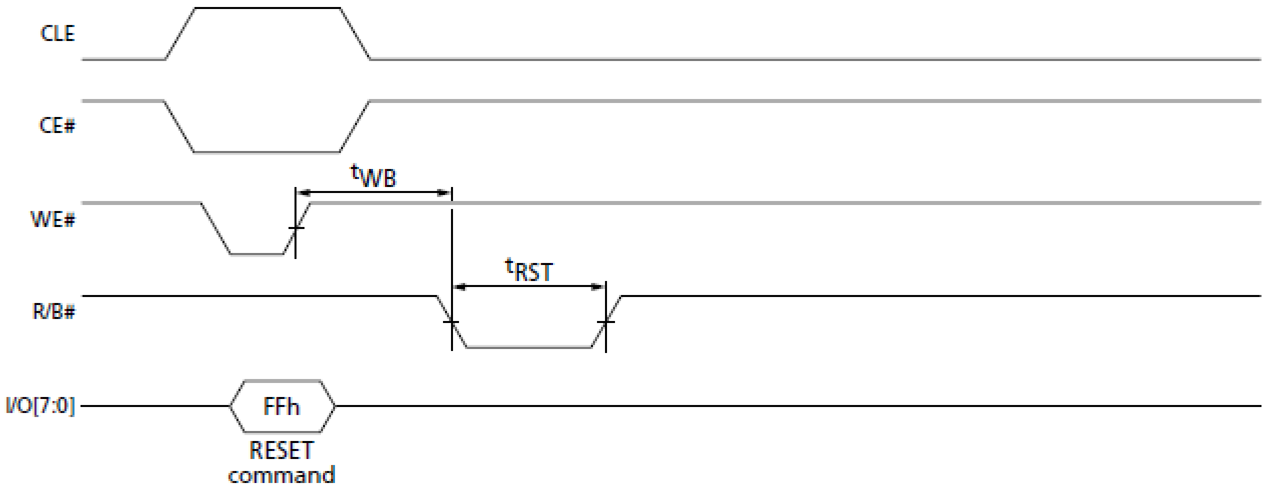
| Parameter | Symbol | Min. | Max. | Unit | Notes |
|---|---|---|---|---|---|
| ALE to RE# dealy | $t_{AR}$ | 10 | - | ns | |
| CE# access time | $t_{CEA}$ | - | 30 | ns | |
| CE# HIGH to output High-Z | $t_{CHZ}$ | - | 50 | ns | 2 |
| CLE to RE# delay | $t_{CLR}$ | 10 | - | ns | |
| CE# HIGH to output hold | $t_{COH}$ | 15 | - | ns | |
| Output High-Z to RE# LOW | $t_{IR}$ | 0 | - | ns | |
| READ cycle time | $t_{RC}$ | 30 | - | ns | |
| RE# access time | $t_{REA}$ | - | 25 | 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 | 2 |
| RE# HIGH to output High-Z | $t_{RHz}$ | - | 65 | ns | |
| RE# pluse width | $t_{RP}$ | 15 | - | ns | |
| Ready to RE# LOW | $t_{RR}$ | 20 | - | ns | |
| Reset time (READ/PROGRAM/ ERASE) | $t_{RST}$ | - | 7/13/600 | us | 3 |
| WE# HIGH to busy | $t_{WB}$ | - | 100 | ns | 4 |
| WE# HIGH to RE# LOW | $t_{WHR}$ | 80 | - | ns | |

**NOTE:**

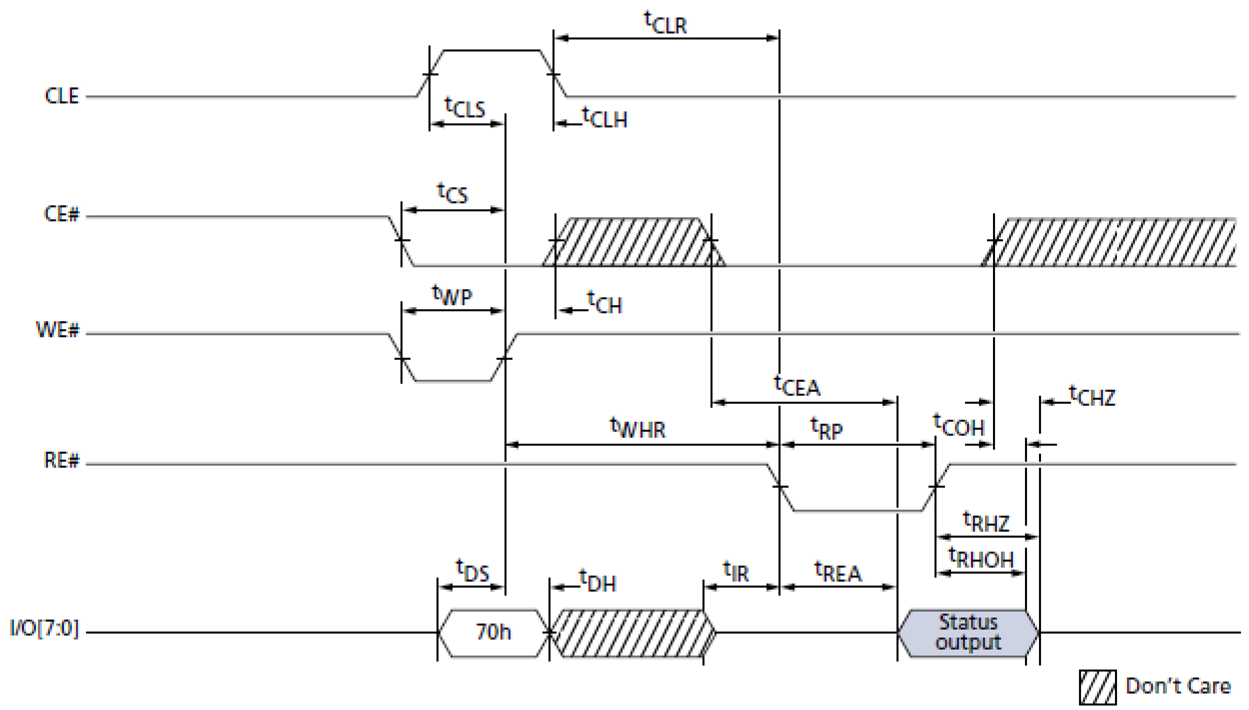
1. AC characteristics may need to be relaxed if I/O drive strength is not set to “full.”
2. Transition is measured ±200mV from steady-state voltage with load. This parameter is sampled and not 100% tested.
3. The first time the RESET (FFh) command is issued while the device is idle, the device will go busy for a maximum of 1ms. Thereafter, the device goes busy for a maximum of 5μs.
4. Do not issue a new command during tWB, even if R/B# is ready.

## Asynchronous Interface Timing Diagrams

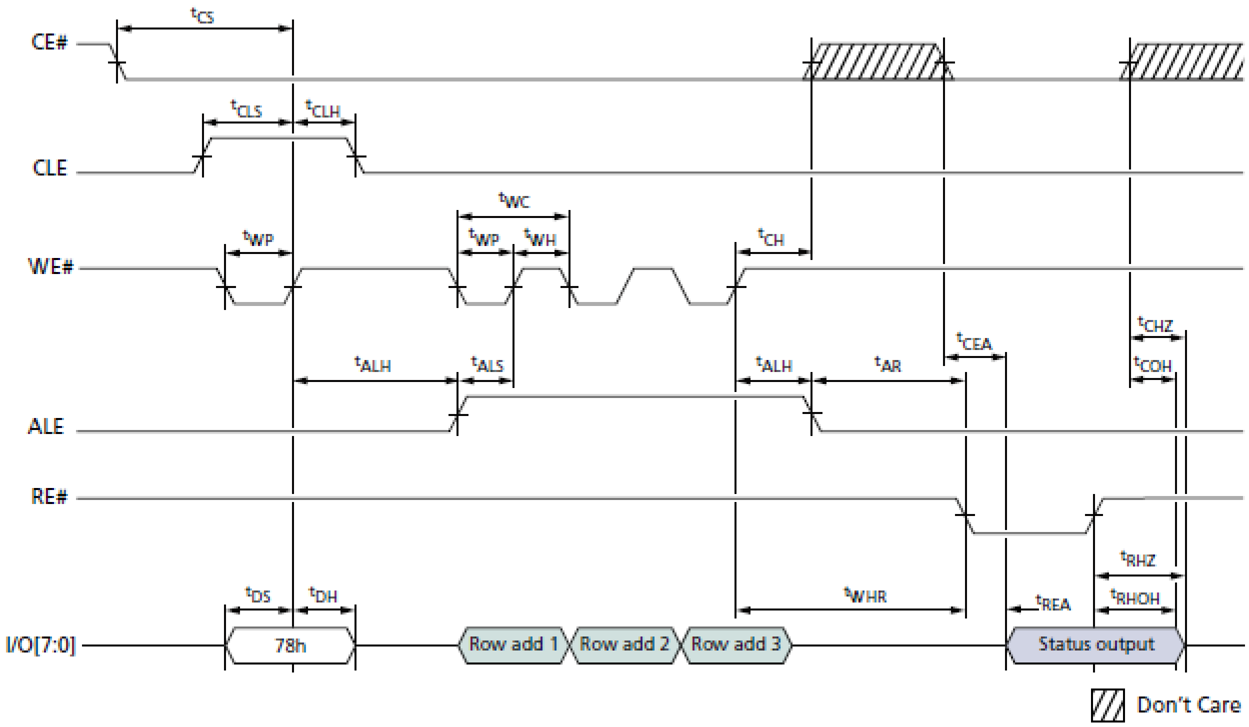
**RESET Operation**



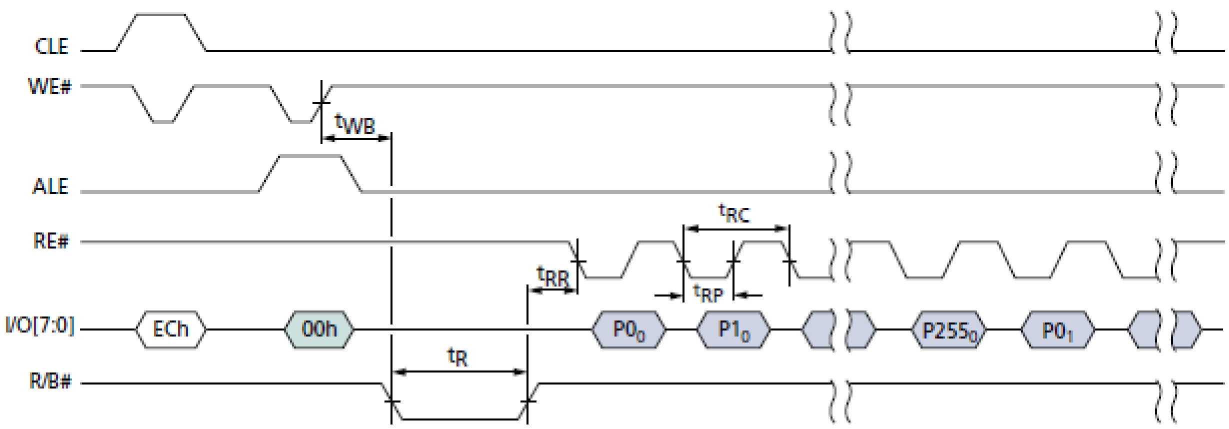

**READ STATUS Cycle**



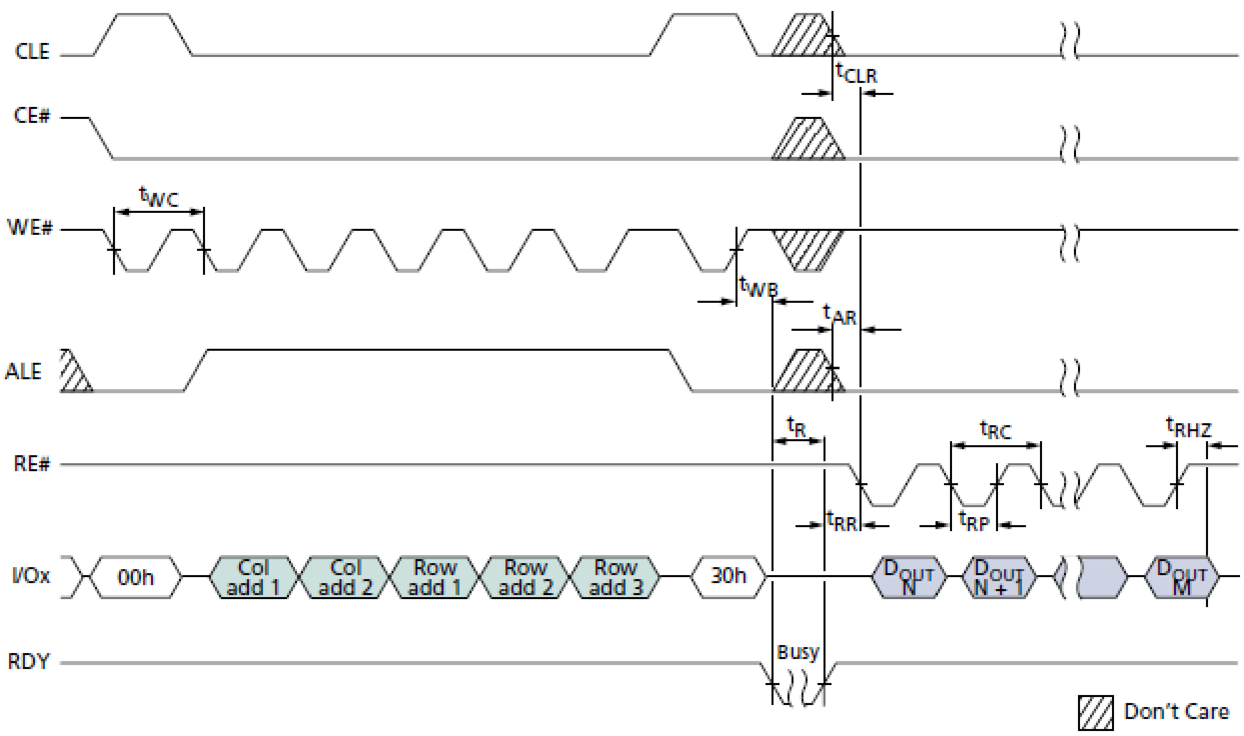
**READ STATUS ENHANCED Cycle**



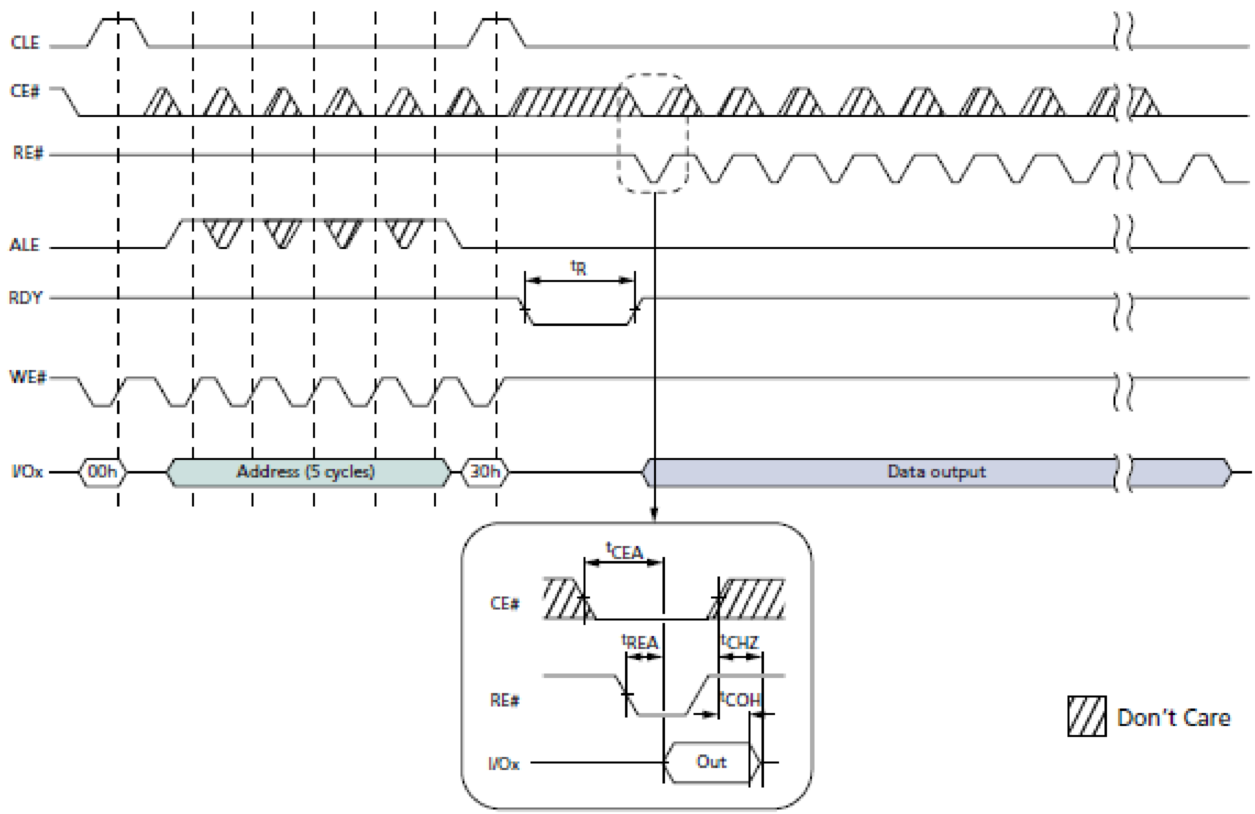

**READ PARAMETER PAGE**



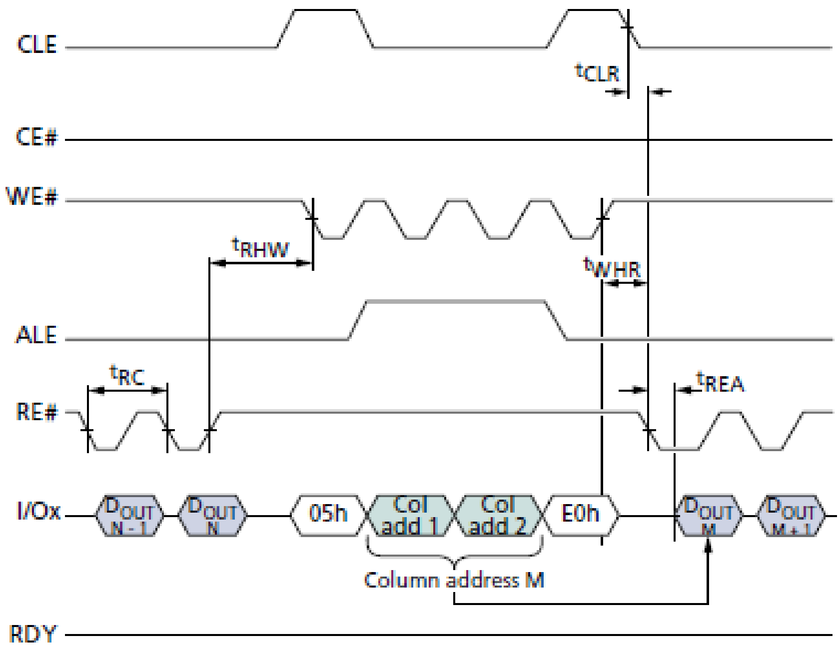
**READ PAGE**



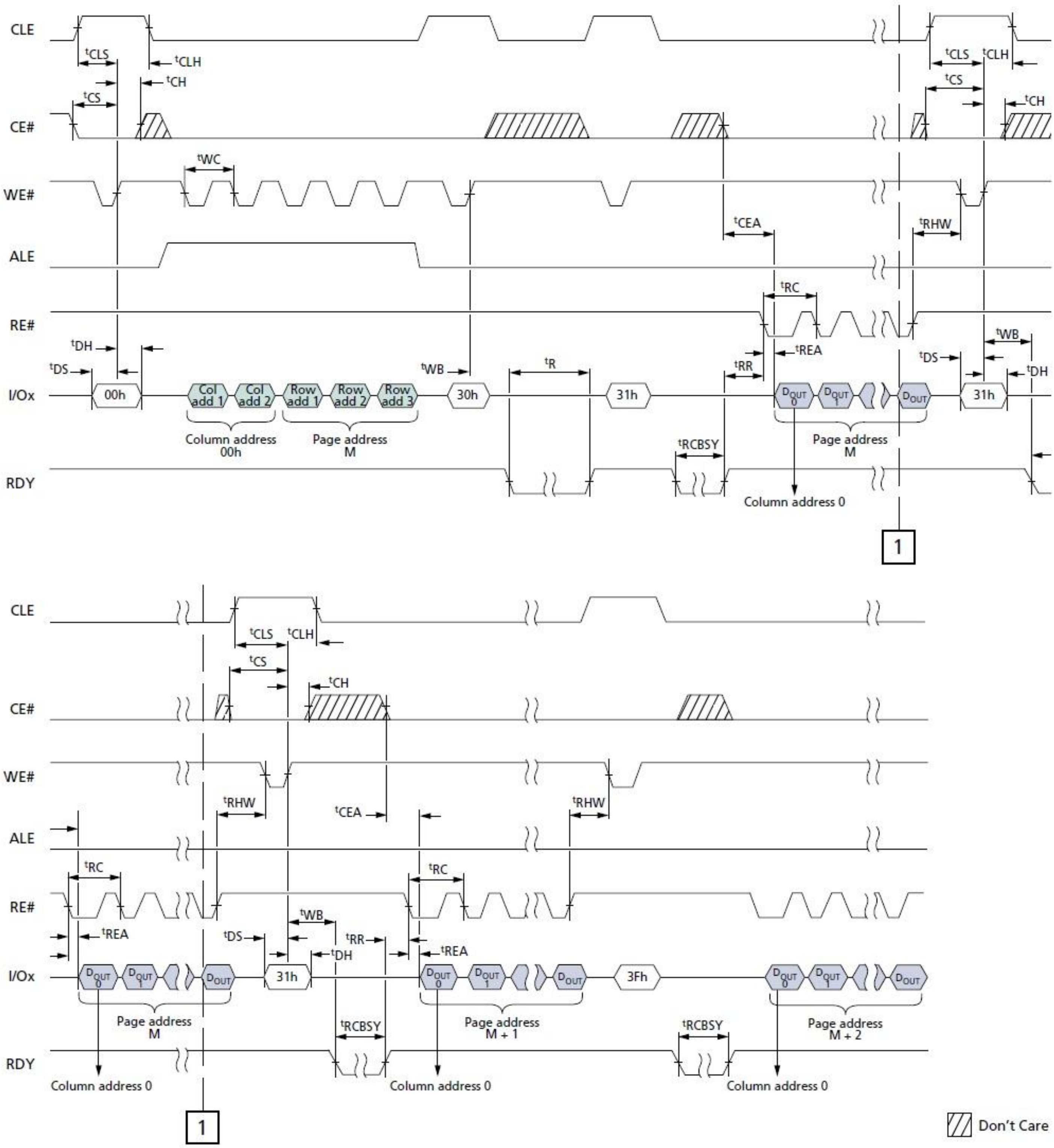

**READ PAGE Operation with CE# "Don't Care"**



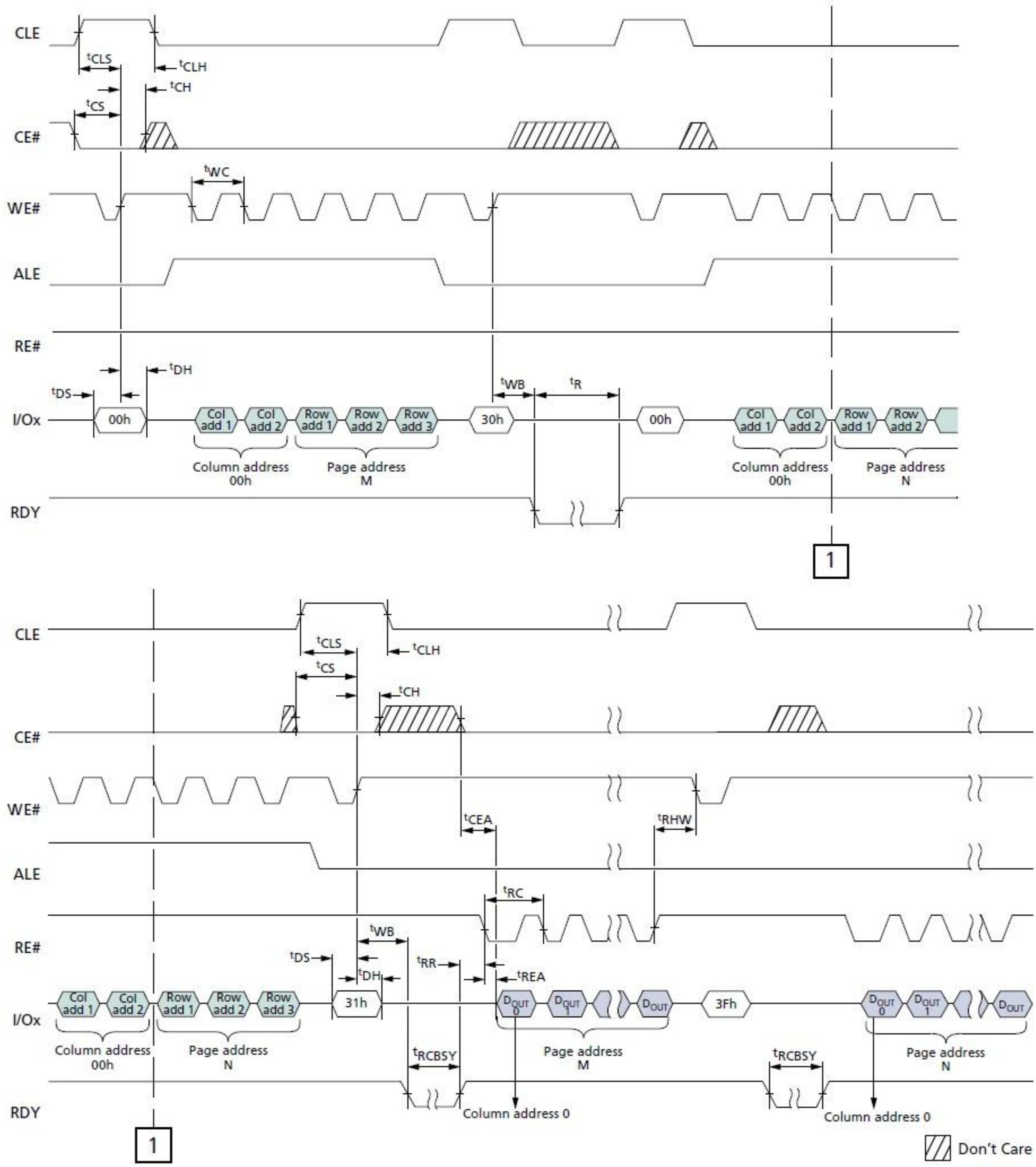
**RANDOM DATA READ**



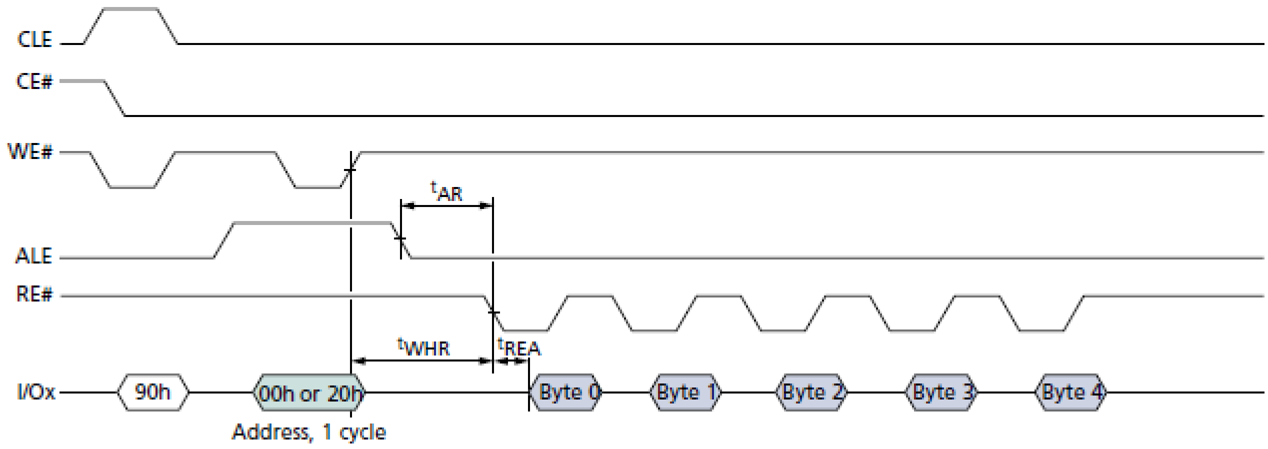
**READ PAGE CACHE SEQUENTIAL**



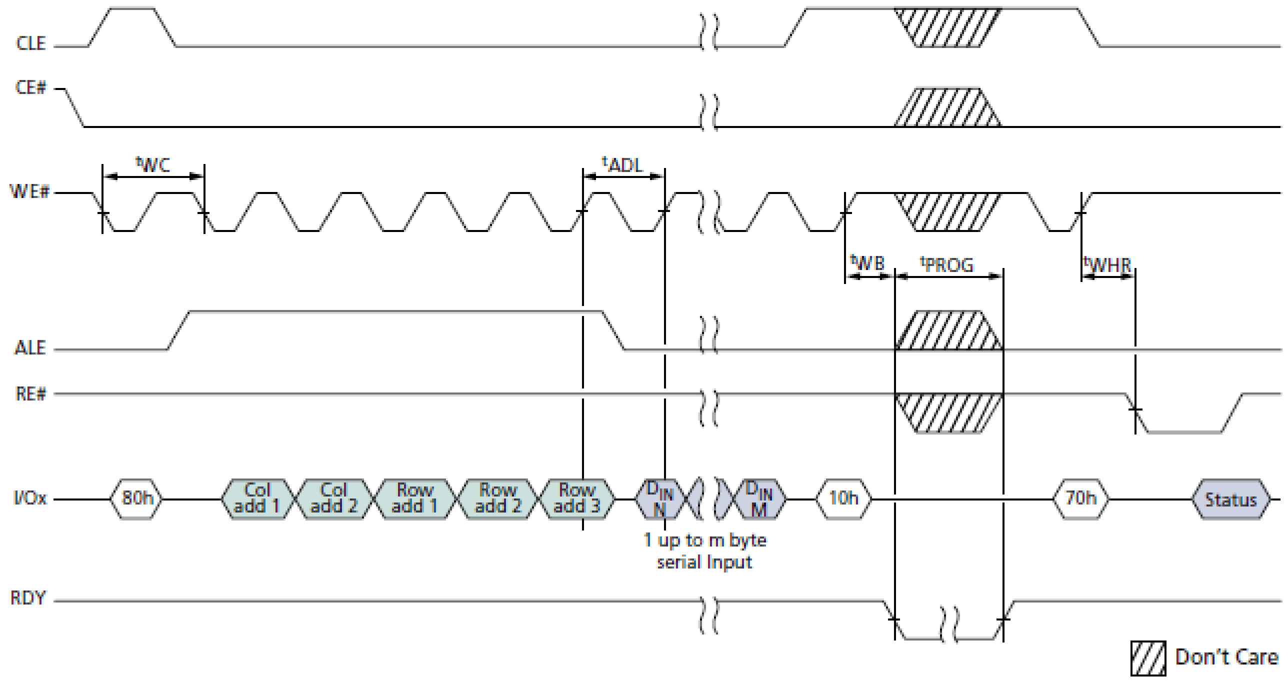
**READ PAGE CACHE RANDOM**



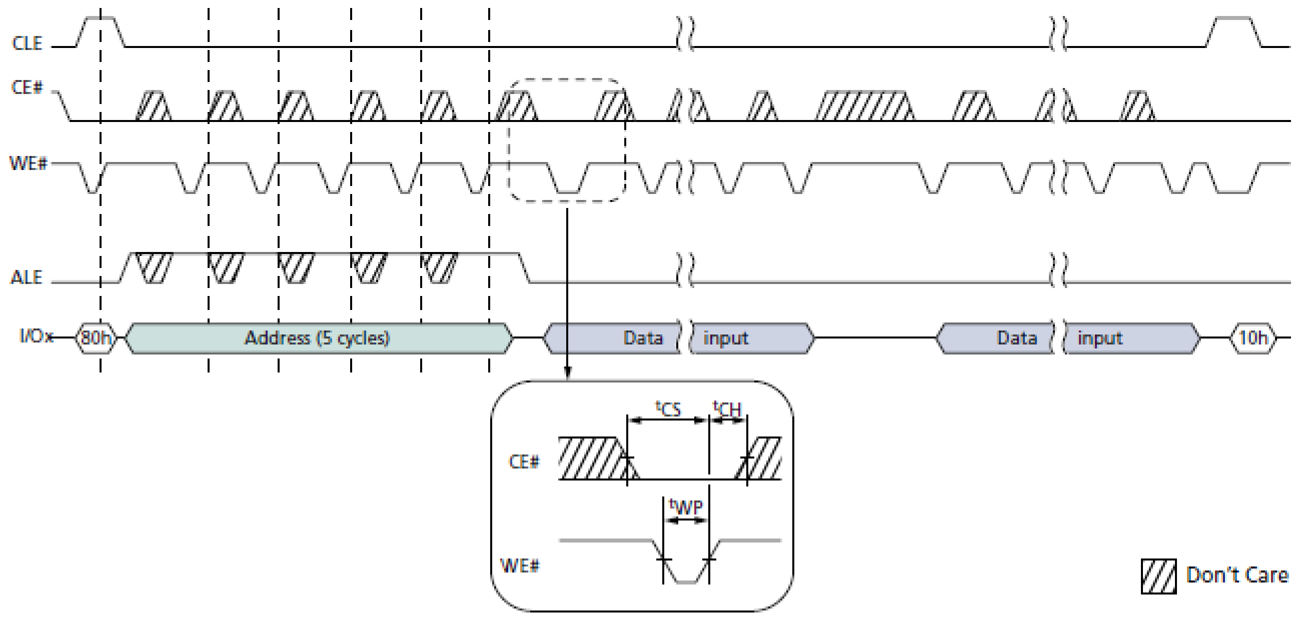
**READ ID Operation**



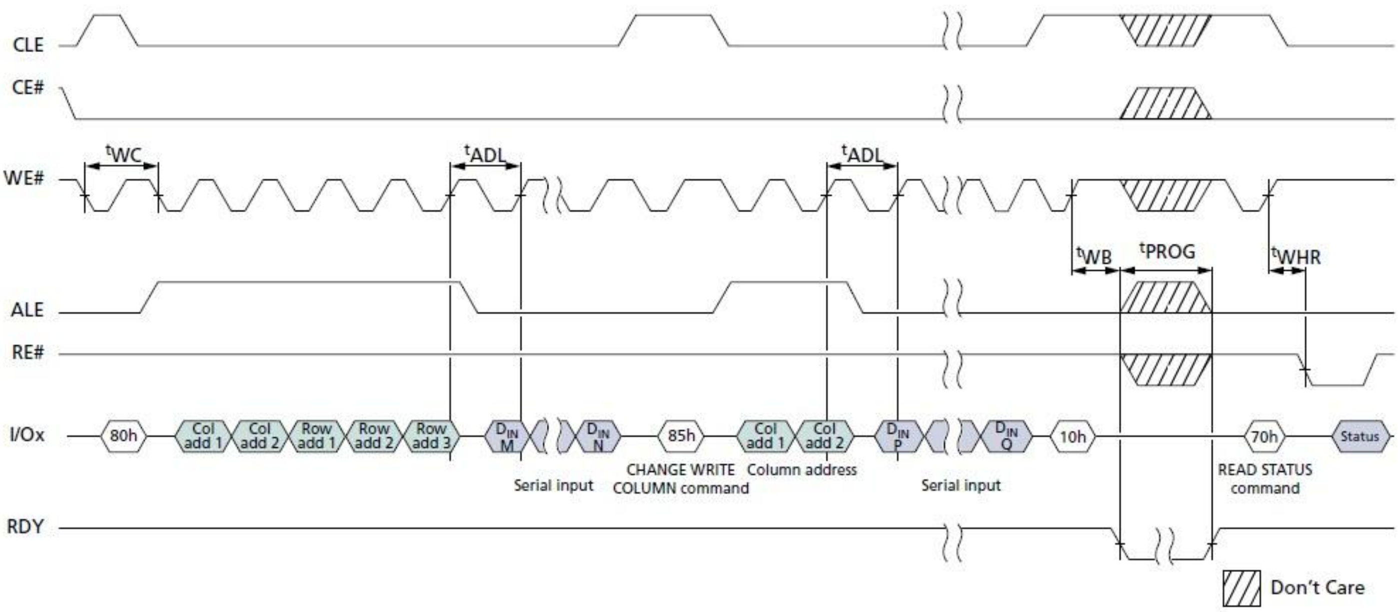

**PROGRAM PAGE Operation**



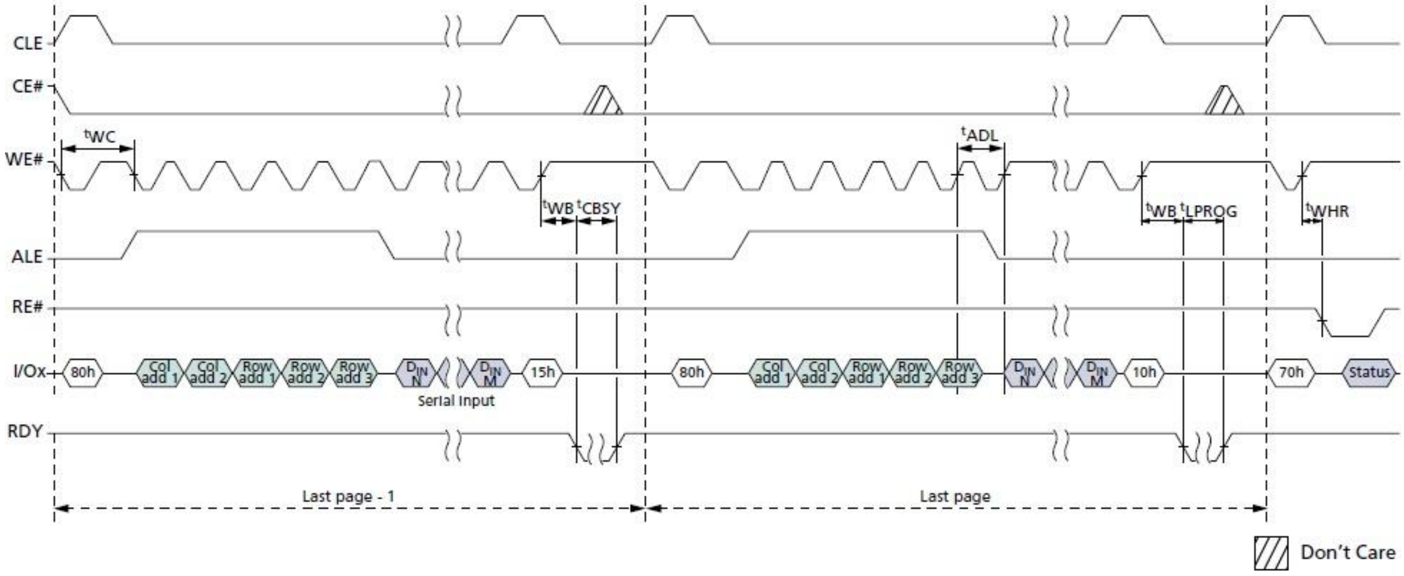
**PROGRAM PAGE Operation with CE# “Don’t Care”**



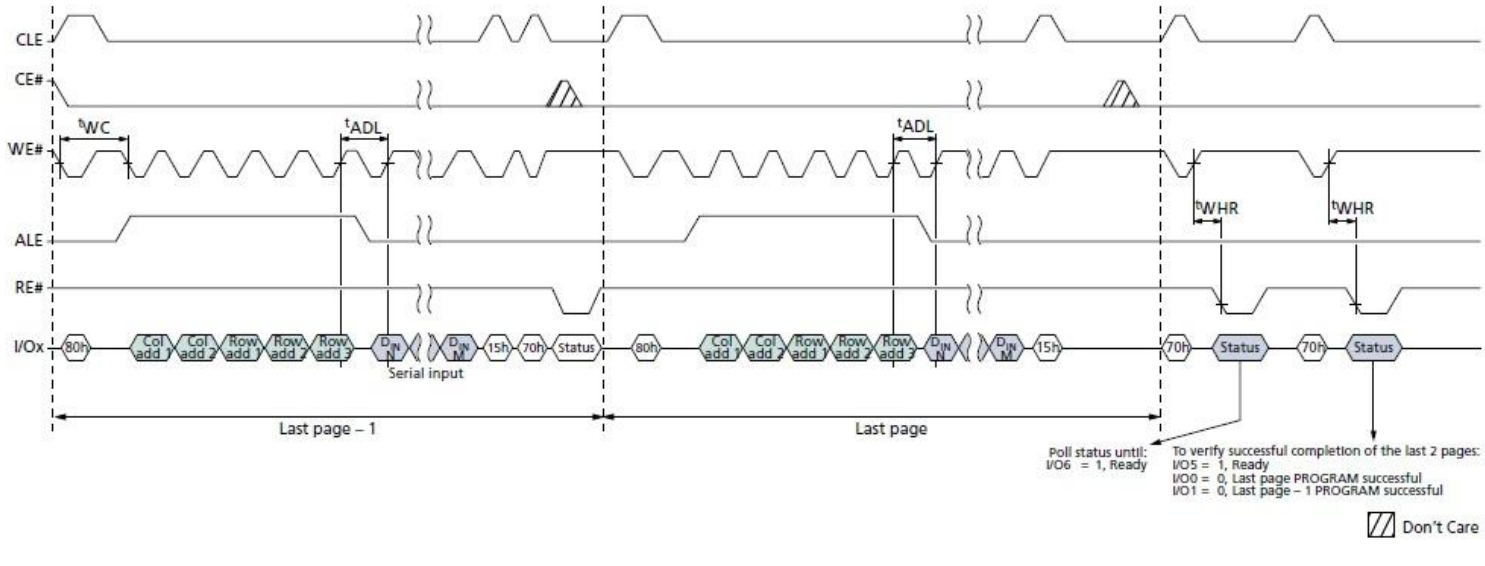

**PROGRAM PAGE Operation with RANDOM DATA INPUT**



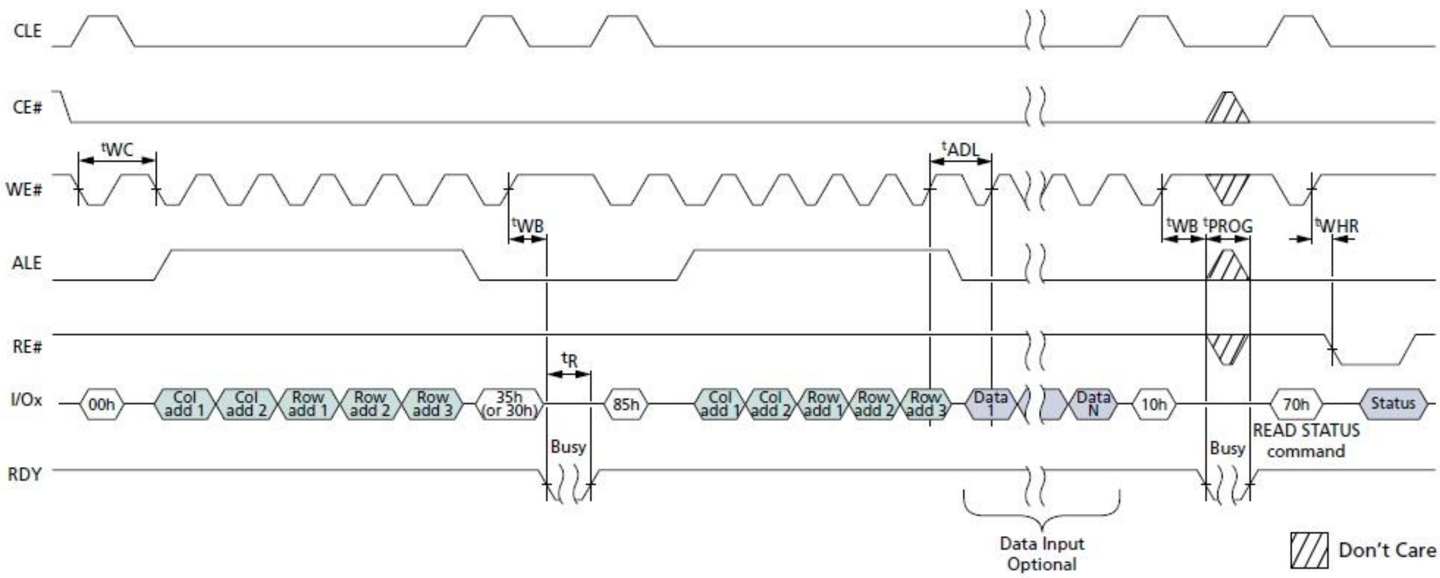
**PROGRAM PAGE CACHE**



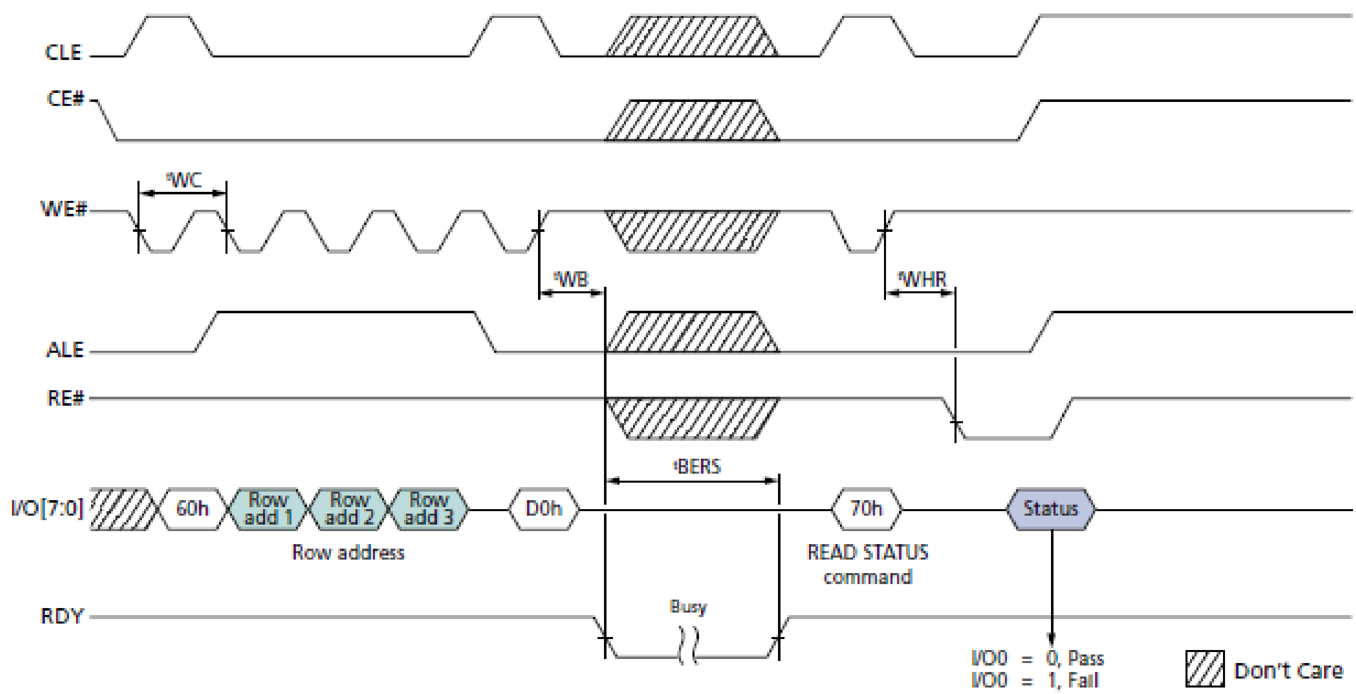

**PROGRAM PAGE CACHE Ending on 15h**



**INTERNAL DATA MOVE**




**ERASE BLOCK Operation**



## Asynchronous Interface Bus Operation

The bus on the device is multiplexed. Data I/O, addresses, and commands all share the same pins 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.

## Asynchronous Interface Mode Selection

| Mode[1] | CE# | CLE | ALE | WE# | RE# | I/Ox | WP# |
|---|---|---|---|---|---|---|---|
| Standby[2] | H | X | X | X | X | X | 0V/VCC |
| 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 |

**NOTE:**
1. Mode selection settings for this table: H = Logic level HIGH; L = Logic level LOW; X = $V_{IH}$ or $V_{IL}$.
2. WP# should be biased to CMOS LOW or HIGH for standby.

## Asynchronous Enable/Standby

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

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

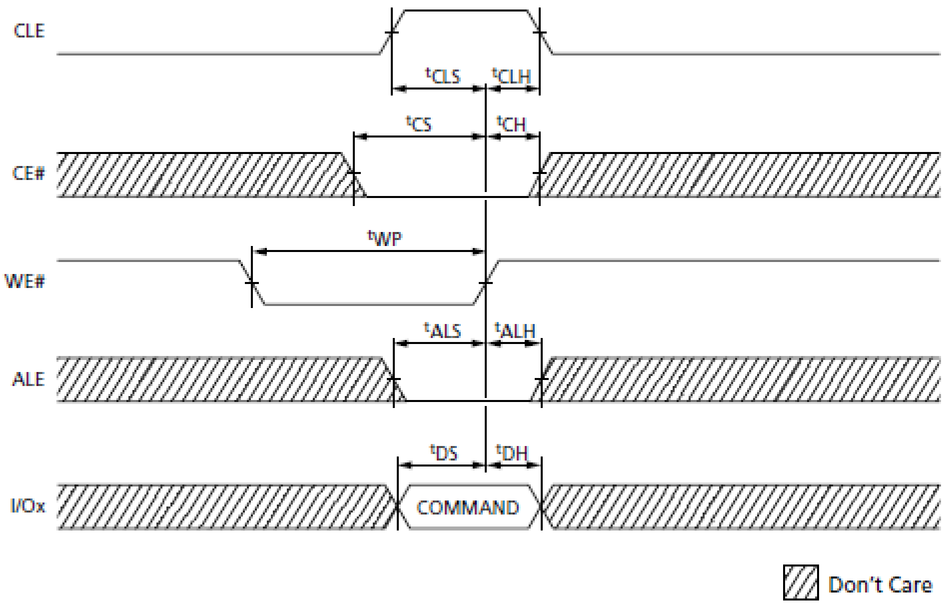
A HIGH CLE signal indicates that a command cycle is taking place. A HIGH ALE signal signifies that an ADDRESS INPUT cycle is occurring.

## Asynchronous Command

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.

**Asynchronous Command Latch Cycle**



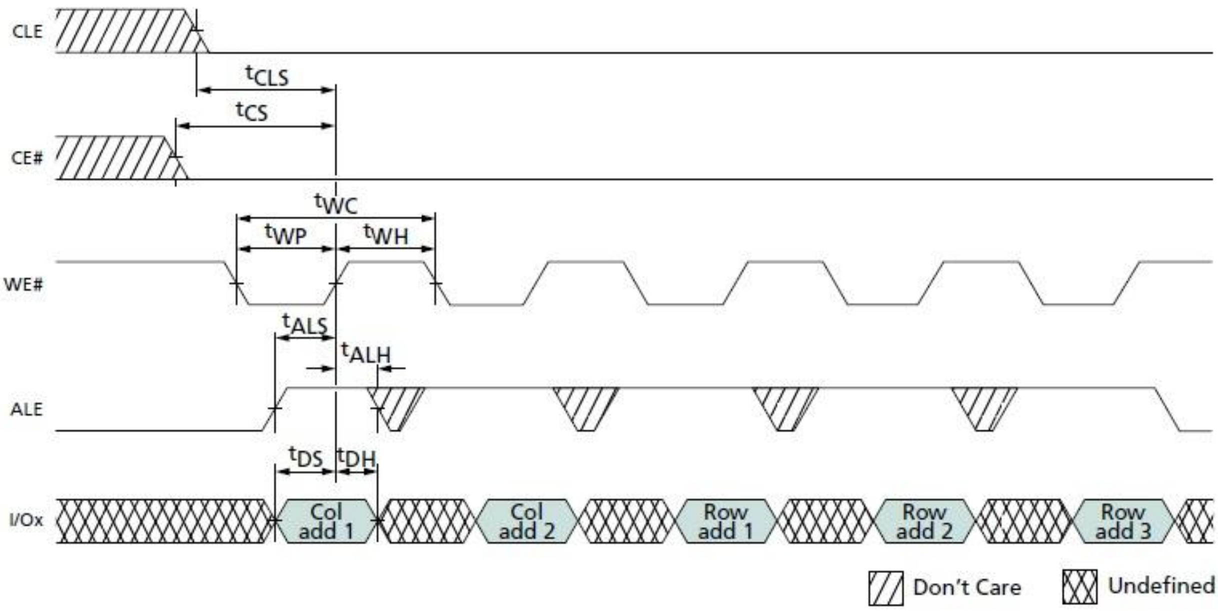
### Asynchronous Addresses

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

**Asynchronous Address Latch Cycle**

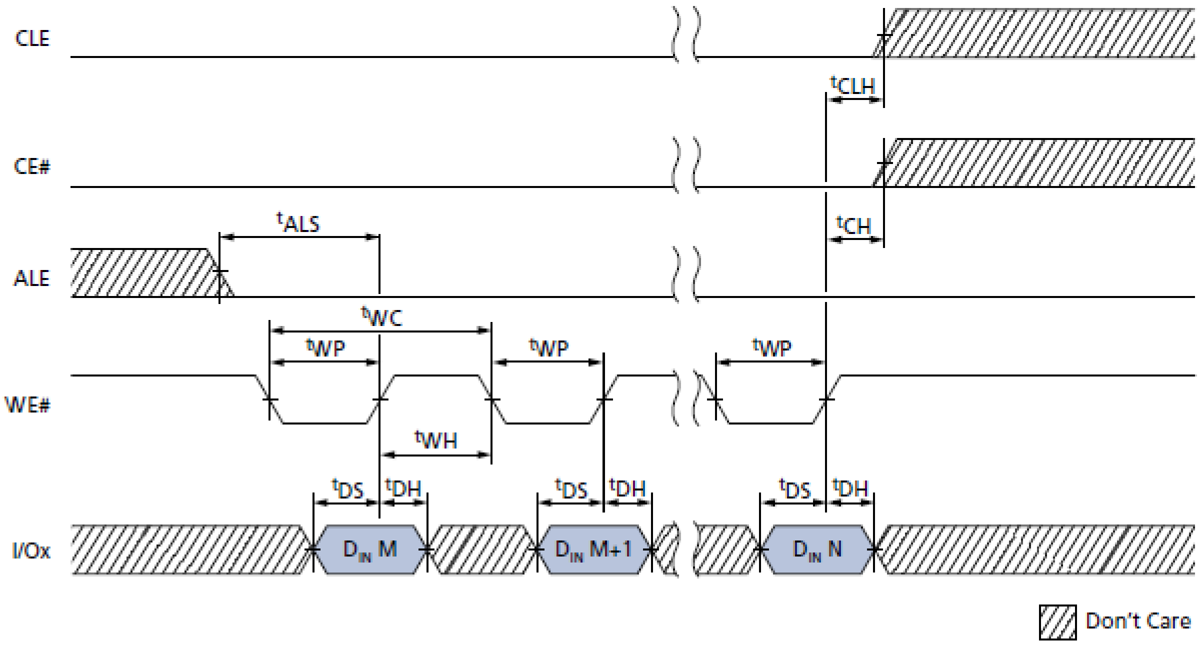


### Asynchronous Data Input

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.

**Asynchronous Data Input Cycles**



## Asynchronous Data Output

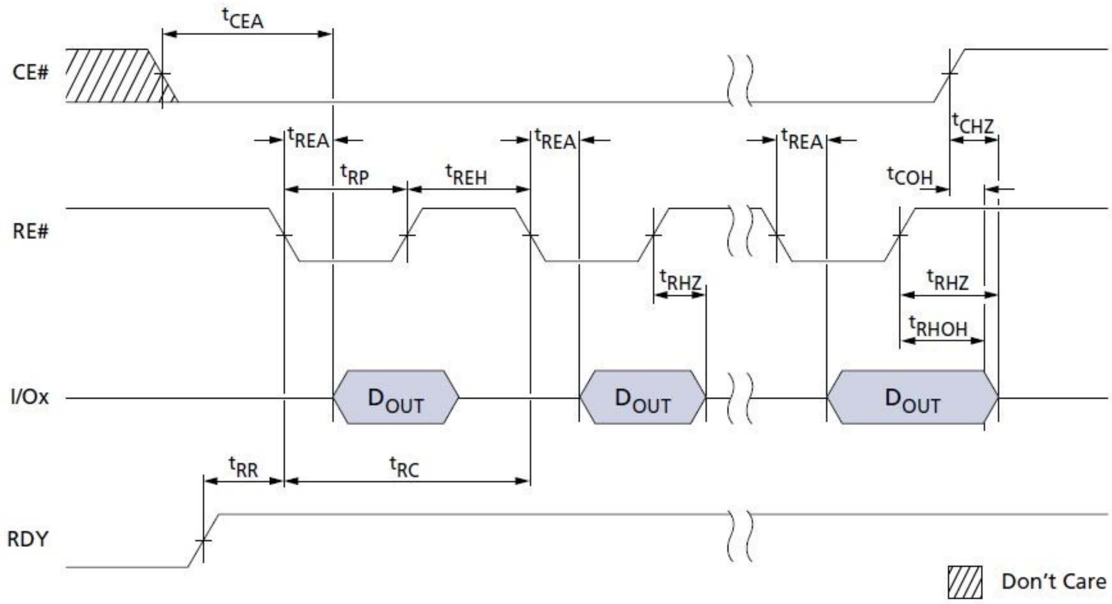
Data can be output from a die (LUN) if it is in a READY state. Data output is supported following a READ operation from the NAND Flash array. Data is output from the cache register of the selected die (LUN) to I/O[7:0] on the falling edge of RE# when CE# is LOW, ALE is LOW, CLE is LOW, and WE# is HIGH.

If the host controller is using a $t_{RC}$ of 30ns or greater, the host can latch the data on th 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#.

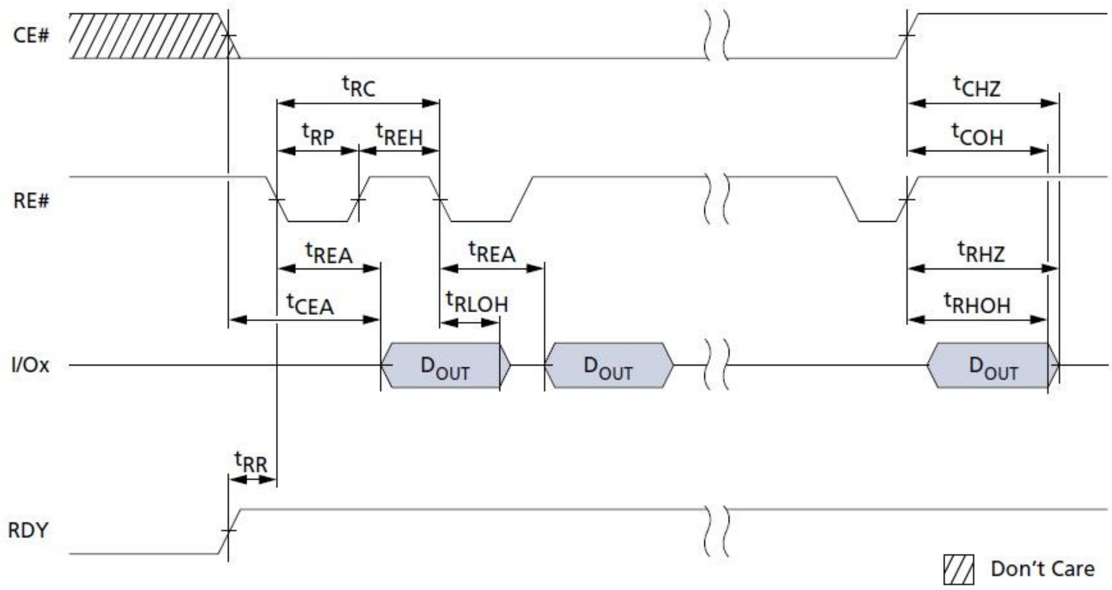
Using the READ STATUS ENHANCED (78h) command prevents data contention following an interleaved die (multi-LUN) operation. After issuing the READ STATUS ENHANCED (78h) command, to enable data output, issue the READ MODE (00h) command.

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.

**Asynchronous Data Output Cycles**




**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 VCC is stable to prevent inadvertent PROGRAM and ERASE operations (see Device Initialization for additional details).

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

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

## Ready/Busy#

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

This signal requires a pull-up resistor, 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.

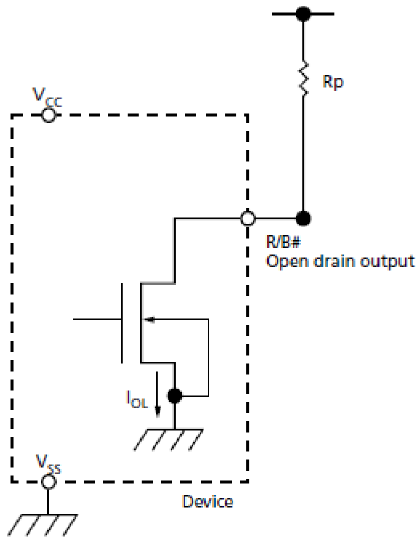
The fall time of the R/B# signal is determined mainly by the output impedance of the R/B# signal and the total load capacitance. Approximate Rp values using a circuit load of 100pF are provided in Figure of “TC vs. Rp”.

The minimum value for Rp is determined by the output drive capability of the R/B# signal, the output voltage swing, and VCC.

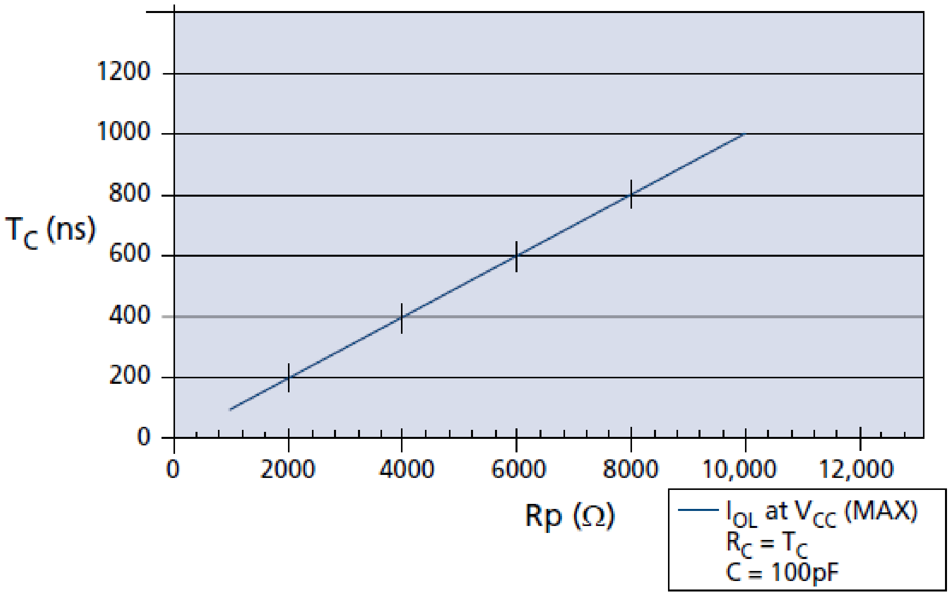
$$R_P = \frac{Vcc(Max) - V_{OL}(MAX)}{I_{OL} + \sum_{IL}}$$

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

**READ/BUSY# Open Drain**



**TC vs. Rp**


1200
1000
800
600
400
200
0
Tc (ns)
0
2000
4000
6000
8000
10,000
12,000
Rp (Ω)
IOL at VCC (MAX)
RC = TC
C = 100pF

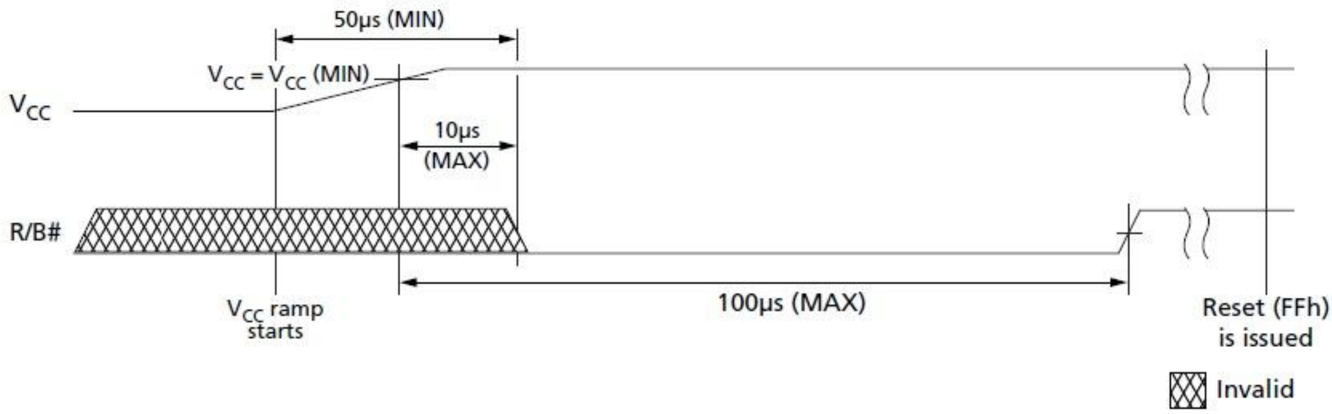
## Device Initialization

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

1. Ramp VCC.
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 50µs has elapsed since the beginning the VCC ramp, and 10µs has elapsed since VCC reaches VCC (MIN).
3. If not monitoring R/B#, the host must wait at least 100µs after VCC reaches VCC (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.

1. The device is now initialized and ready for normal operation.

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




## Power Cycle Requirements

Upon power-down the NAND device requires a maximum voltage and minimum time that the host must hold VCC and VCCQ below the voltage prior to power-on.

**Power Cycle Requirements**

| Parameter | Value | Unit |
|---|---|---|
| **Maximum VCC/VCCQ** | 100 | mV |
| **Minimum time below maximum voltage** | 100 | ns |

## Reset Operations

### RESET (FFh)

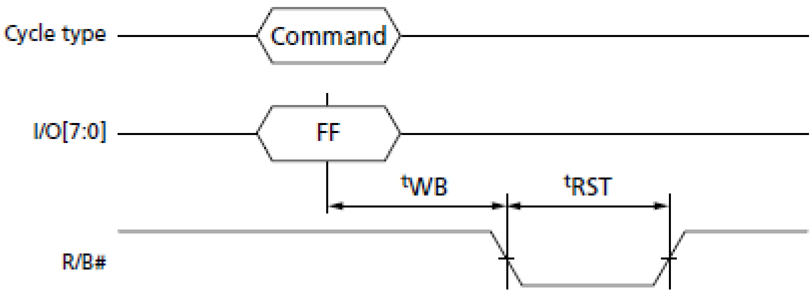
The RESET command is used to put the memory device into a known condition and to abort the command sequence in progress.

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.

**RESET (FFh) Operation**



## Identification Operations

### READ ID (90h)

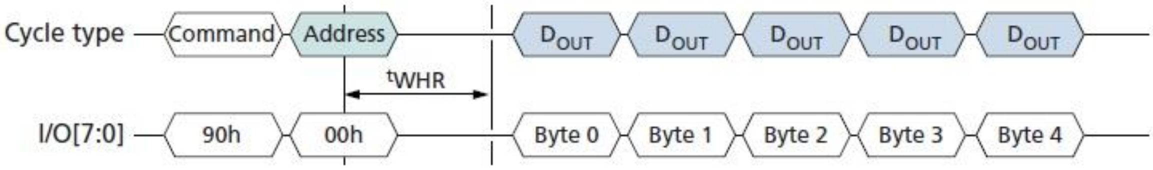
The READ ID (90h) command is used to read identifier codes programmed into the target. This command is accepted by the target only when all die (LUNs) on the target are idle.

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.

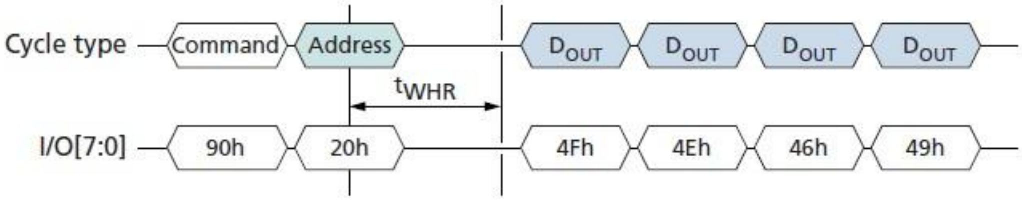
When the 90h command is followed by a 20h address cycle, the target returns the 4-byte ONFI identifier code.

**READ ID (90h) with 00h Address Operation**




**NOTE:**
1. See the Read ID Patameter tables for byte definitions.

**READ ID (90h) with 20h Address Operation**




**NOTE:**
1. See Read ID Parameter tables for byte definitions.

## READ ID Parameter Tables

### READ ID Parameters for Address 00h

| | Options | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0 | Value[1] |
|---|---|---|---|---|---|---|---|---|---|---|
| **Byte 0 - Manufacturer ID** | | | | | | | | | | |
| Manufacturer | | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 2Ch |
| **Byte 1 - Device ID** | | | | | | | | | | |
| Device | 2Gb, x8, 1.8V | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | AAh |
| Byte 2 | | | | | | | | | | |
| Number of die per CE# | 1 | | | | | | | 0 | 0 | 00b |
| 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 | | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 90h |
| Byte 3 | | | | | | | | | | |
| Page size | 2KB | | | | | | | 0 | 1 | 01b |
| Spare area size ( bytes) | 128B | | | | | | 1 | | | 1b |
| Block size ( w/o spare) | 128KB | | | 0 | 1 | | | | | 01b |
| Organization | x8 | | 0 | | | | | | | 0b |
| Serial access(MIN) | 30ns | 1 | | | | 0 | | | | 1xxx0b |
| Byte value | | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 15h |
| **Byte 4** | | | | | | | | | | |
| Reserved | | | | | | | | 1 | 0 | 10b |
| Planes per CE# | 2 | | | | | 0 | 1 | | | 01b |
| Plane size | 1Gb | | 0 | 0 | 0 | | | | | 000b |
| Internal ECC | ECC Disabled | 0 | | | | | | | | 0b |
| Byte value | | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 06h |

Note: 1. b = binary; h = hexadecimal

### READ ID Parameters for Address 20h

| Byte | Options | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0 | Value |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | “O” | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 4Fh |
| 1 | “N” | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 4Eh |
| 2 | “F” | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 46h |
| 3 | “I” | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 49h |
| 4 | Undefined | X | X | X | X | X | X | X | X | XXh |

## READ PARAMETER PAGE (ECh)

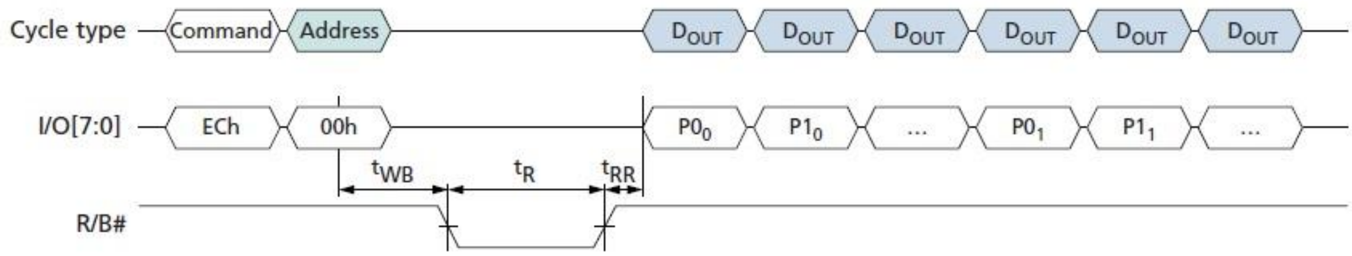
The READ PARAMETER PAGE (ECh) command is used to read the ONFI parameter page programmed into the target. This command is accepted by the target only when all die (LUNs) on the target are idle.

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

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

A minimum of three copies of the parameter page are stored in the device. Each parameter page is 256 bytes. If desired, the RANDOM DATA READ (05h-E0h) command can be used to change the location of data output. Each copy has the CRC value stored at the last two bytes. The software can read the first copy of ONFI parameter page, calculate the CRC and compare it with the stored value. If mis-match found then the 2nd copy should be read and so forth.

**READ PARAMETER (ECh) Operation**




## 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.

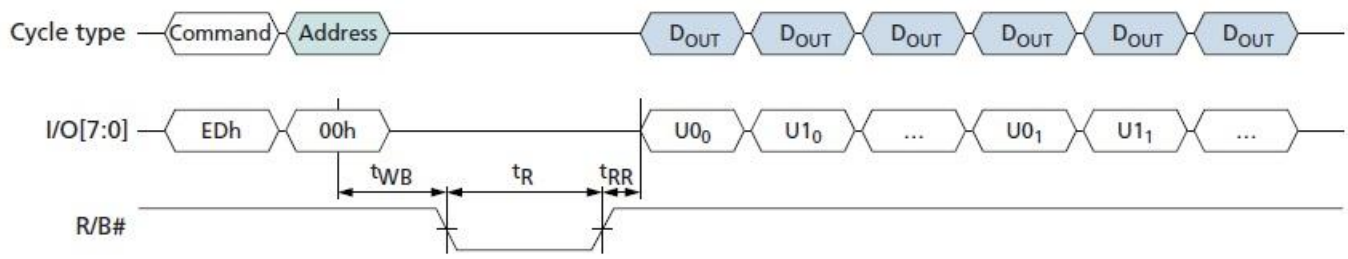
Writing EDh to the command register puts the target in read unique ID mode. The target stays in this mode until another valid command is issued.

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.

**READ UNIQUE ID (EDh) Operation**



## Parameter Page Data Structure Tables

### Parameter Page Data Structure

| Byte | Description | Value |
|---|---|---|
| 0-3 | Parameter page signature | 4Fh, 4Eh, 46h,49h |
| 4-5 | Revision number | 02h, 00h |
| 6-7 | Features supported | 18h, 00h |
| 8-9 | Optional commands supported | 3Fh, 00h |
| 10-31 | Reserved | 00h |
| 32-43 | Device manufacturer | 4Dh, 49h, 43h, 52h, 4Fh, 4Eh, 20h, 20h, 20h, 20h, 20h, 20h |
| 44-63 | Device model | 4Dh, 54h, 32h, 39h, 46h, 32h, 47h, 30h, 38h, 41h, 42h, 42h, 47h, 41h, 33h, 57h, 20h, 20h, 20h, 20h |
| 64 | Manufacturer ID | 2Ch |
| 65-66 | Date code | 00h, 00h |
| 67-79 | Reserved | 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h |
| 80-83 | Number of data bytes per page | 00h, 08h, 00h, 00h |
| 84-85 | Number of spare bytes per page | 80h, 00h |
| 86-89 | Number of data bytes per partial page | 00h, 02h, 00h, 00h |
| 90-91 | Number of spare bytes per partial page | 20h, 00h |
| 92-95 | Number of pages per block | 40h, 00h, 00h, 00h |
| 96-99 | Number of blocks per unit | 00h, 08h, 00h, 00h |
| 100 | Number of logical units | 01h |
| 101 | Number of address cycles | 23h |
| 102 | Number of bits per cell | 01h |
| 103-104 | Bad blocks maximum per unit | 28h, 00h |
| 105-106 | Block Endurance | 01h, 05h |
| 107 | Guaranteed valid blocks at beginning of target | 08h |
| 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 | 08h |
| 113 | Number of interleaved address bits | 01h |
| 114 | Interleaved operation attributes | 0Eh |
| 115-127 | Reserved | 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, |
| 128 | I/O pin capacitance | 08h |
| 129-130 | Timing mode support | 0Fh, 00h |
| 131-132 | Program cache timing mode support | 0Fh, 00h |
| 133-134 | $t_{PROG}$ Maximum page program time | 58h, 02h |
| 135-136 | $t_{BERS}$ Maximum block erase time | 10h, 27h |
| 137-138 | $t_R$ Maximum page read time | 19h, 00h |
| 139-140 | $t_{CCS}$ Minimum | 64h, 00h |
| 141-163 | Reserved | 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, |
| 164-165 | Vendor-specific revision number | 01h, 00h |

**Parameter Page Data Structure (Continued)**

| Byte | Description | Value |
|---|---|---|
| 166-253 | Vendor-specific | 01h, 00h, 00h, 02h, 04h, 80h, 01h, 81h, 04h, 03h, 02h, 01h, 1Eh, 90h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h |
| 254-255 | Integrity CRC | Set at test |
| 256-511 | Value of bytes 0-255 | |
| 512-767 | Value of bytes 0-255 | |
| 768+ | Additional redundant parameter pages | |

Note: 1. h = hexadecimal.

## 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.

### Feature Address Definitions

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

### Feature Address 90h - Array Operation Mode

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

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

## SET FEATURES (EFh)

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

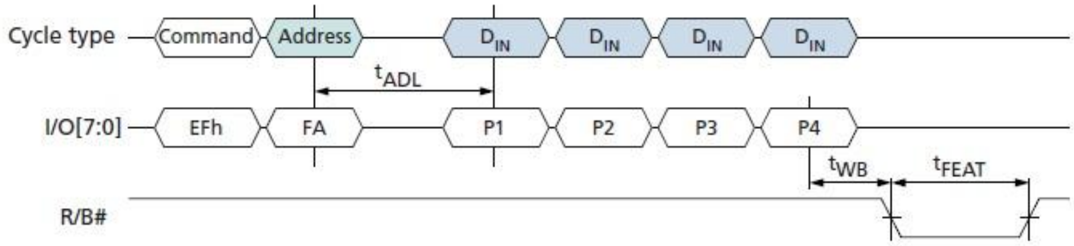
Writing EFh to the command register puts the target in the set features mode. The target stays in this mode until another command is issued.

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

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

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

**SET FEATURES (EFh) Operation**




## GET FEATURES (EEh)

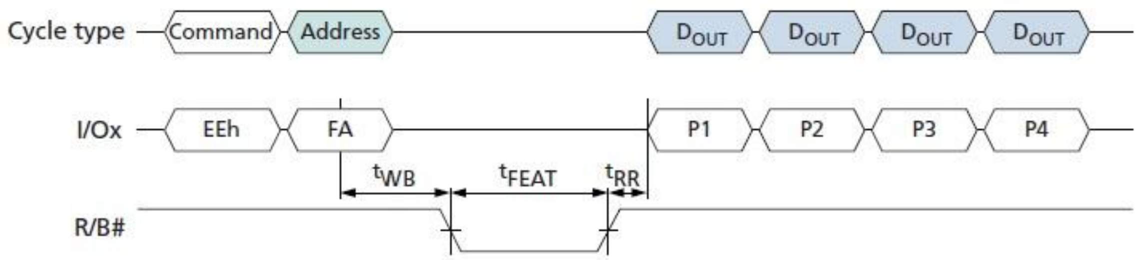
The GET FEATURES (EEh) command reads the subfeature parameters (P1-P4) from the specified feature address. This command is accepted by the target only when all die (LUNs) on the target are idle.

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.

**GET FEATURES (EEh) Operation**



### Feature Addresses 01h: Timing Mode

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

**NOTE:**

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 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.

### Feature Addresses 80h: Programmable I/O Drive Strength

| Subfeature Parameter | Options | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0 | Value | Notes |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **P1** | | | | | | | | | | | |
| 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 | |
| **P2** | | | | | | | | | | | |
| | | Reserved(0) | | | | | | | | 00h | |
| **P3** | | | | | | | | | | | |
| | | Reserved(0) | | | | | | | | 00h | |
| **P4** | | | | | | | | | | | |
| | | 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.

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

| Subfeature Parameter | Options | I/O7 | I/O6 | I/O5 | I/O4 | I/O3 | I/O2 | I/O1 | I/O0 | Value | Notes |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **P1** | | | | | | | | | | | |
| 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 | |
| **P2** | | | | | | | | | | | |
| Reserved | | Reserved(0) | | | | | | | | 00h | |
| **P3** | | | | | | | | | | | |
| Reserved | | Reserved(0) | | | | | | | | 00h | |
| **P4** | | | | | | | | | | | |
| Reserved | | 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 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 ($t_R$) 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).

## Status Register Definition

| SR Bit | Program Page | Program Page Cache Mode | Page Read | Page Read Cache Mode | Block Erase | Description |
|---|---|---|---|---|---|---|
| 7 | Write protect | Write protect | Write protect | Write protect | Write protect | 0 = Protected<br>1 = Not protected |
| 6 | RDY | RDY cache | RDY | RDY cache | RDY | 0 = Busy(PROGRAM operation in progress)<br>1 = Ready(Cache can accept data; R/B# follows) |
| 5 | ARDY | ARDY | ARDY | ARDY | ARDY | 0 = Busy (PROGRAM operation in progress)<br>1 = Ready (Internal operations completed, if cache mode is used) |
| 4 | 0 | 0 | ECC status[1] | ECC status (N–1)[1] | 0 | 00 = Normal or uncorrectable<br>01 = 4~6<br>10 = 1~3<br>11 = 7~8 (Rewrite recommended) |
| 3 | 0 | 0 | | | 0 | |
| 2 | - | - | - | - | - | Don’t Care |
| 1 | FAILC ( N-1) | FAILC ( N-1) | Reserved | - | - | 0 = Pass<br>1 = Fail<br>This bit is valid only when RDY (SR bit 6) is 1. This bit retains the status of the previous valid program operation when the most recent program operation is complete. |
| 0 | FAIL | FAIL (N) | FAIL[2] | FAIL (N-1) | FAIL | 0 = Pass<br>1 = Fail<br>This bit is set if the most recent finished operation on the selected die (LUN) failed. This bit is valid only when ARDY (SR bit 5) is 1. |

**NOTE:**
1. Bit = 11 when a rewrite is recommended because the page includes READ errors per sector (512-Byte [main] + 16-Byte [spare] +16-Byte [parity]). When ECC is enabled, up to 7~8-bit error is corrected automatically.\
2. A status register bit defined as FAIL signifies that an uncorrectable READ error has occurred.

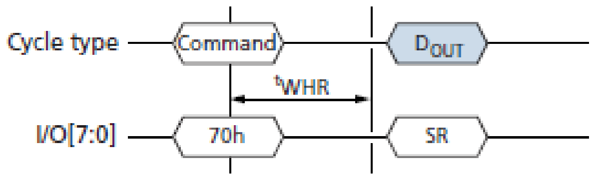
## READ STATUS (70h)

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.

**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.

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.

**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 in the cache register.

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

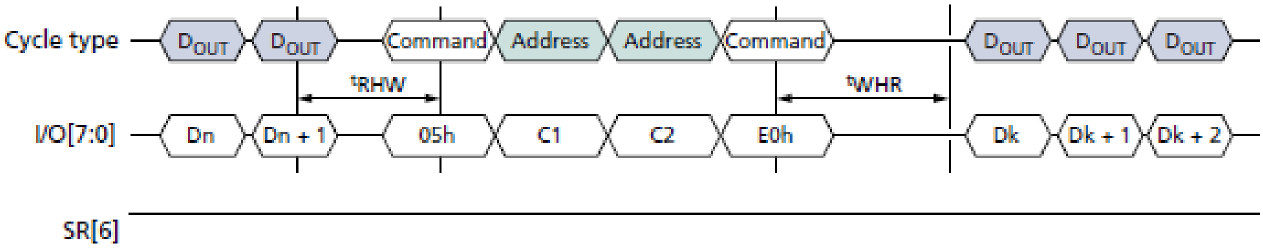
## RANDOM DATA READ (05h-E0h)

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.

**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.

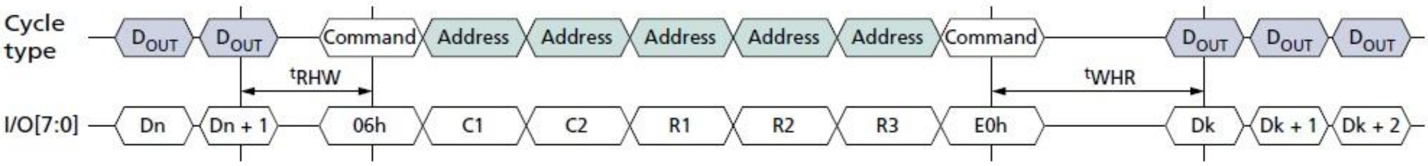
Following a two-plane read page operation, the RANDOM DATA READ TWO-PLANE (06h-E0h) command is used to select the cache register to be enabled for data output. After data output is complete on the selected plane, the command can be issued again to begin data output on another plane.

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.

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



## RANDOM DATA INPUT (85h)

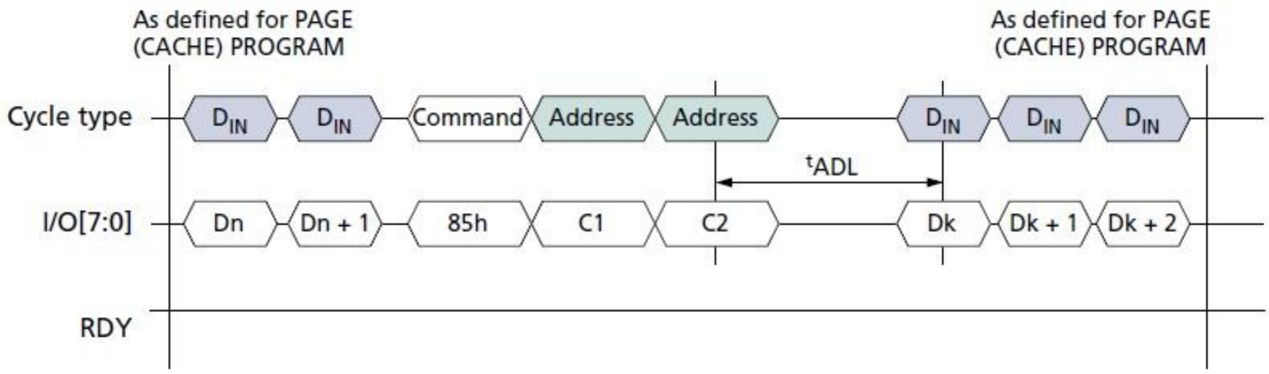
The RANDOM DATA INPUT (85h) command changes the column address of the selected cache register and enables data input on the last-selected die (LUN). This command is accepted by the selected die (LUN) when it is ready (RDY = 1; ARDY = 1). It is also accepted by the selected die (LUN) during cache program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

**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 INTERNAL DATA MOVE TWO-PLANE (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.

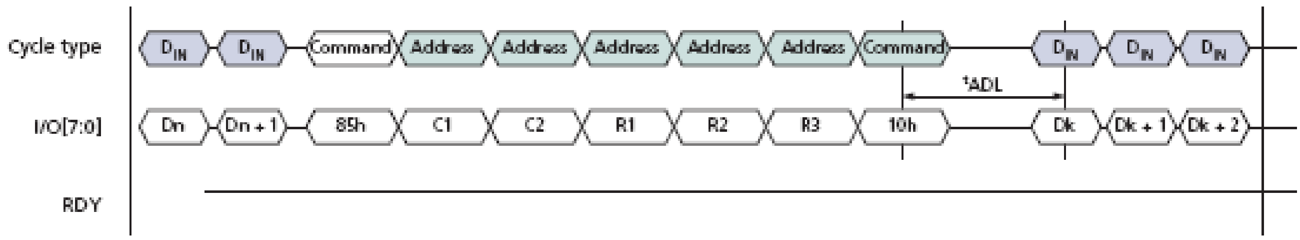
The PROGRAM FOR INTERNAL DATA INPUT (85h) command enables the host to modify the original page and block address for the data in the cache register to a new page and block address.

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.

**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 tR 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 TWO-PLANE CACHE (00h-00h-31h) command.

To begin a two-plane read page cache sequence, begin by issuing a READ PAGE TWOPLANE operation using the READ PAGE TWO-PLANE (00h-00h-30h) and READ PAGE (00h-30h) commands. R/B# goes LOW during tR and the selected die (LUN) is busy (RDY = 0, ARDY = 0). After tR (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 TWO-PLANE CACHE (00h-00h-31h)] – copies the pages specified from the NAND Flash array to the corresponding data registers.

After the READ PAGE CACHE series (31h, 00h-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 TWOPLANE (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-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 tRCBSY 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-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.

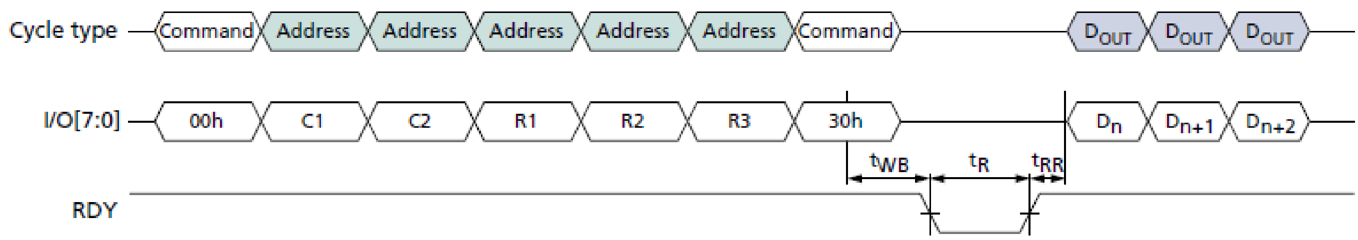
To determine the progress of the data transfer, the host can monitor the target's R/B# signal or, alternatively, the status operations (70h, 78h) can be used. If the status operations are used to monitor the LUN's status, when the die (LUN) is ready (RDY = 1, ARDY = 1), the host disables status output 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.

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.

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



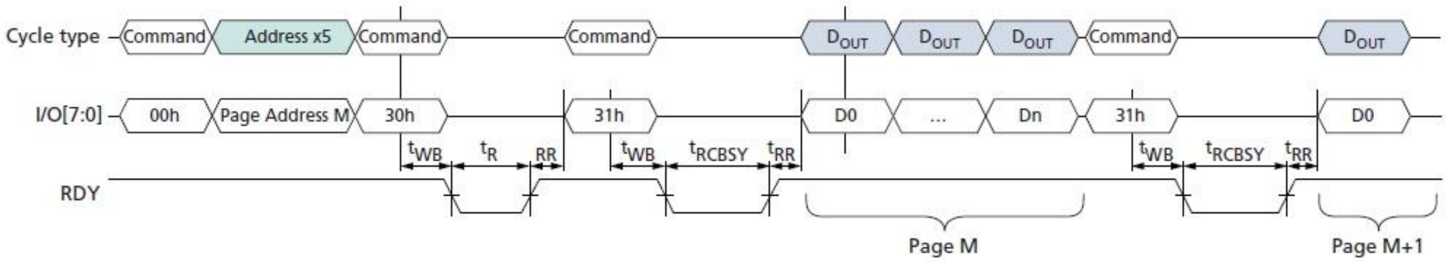
## READ PAGE CACHE SEQUENTIAL (31h)

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 $t_{RCBSY}$. 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.

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.

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



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

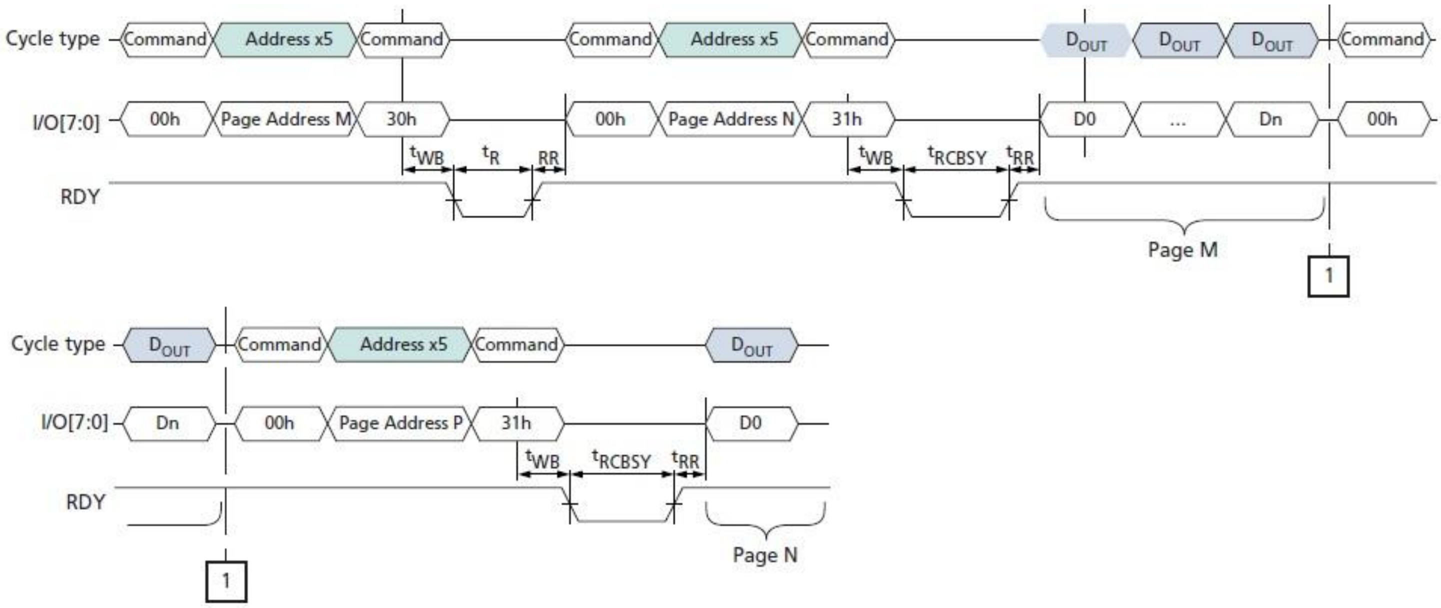
The READ PAGE CACHE RANDOM (00h-31h) command reads the specified block and page into the data register while the previous page is output from the cache register. This command is accepted by the die (LUN) when it is ready (RDY = 1, ARDY = 1). It is also accepted by the die (LUN) during READ PAGE CACHE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

To issue this command, write 00h to the command register, then write 5 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.

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



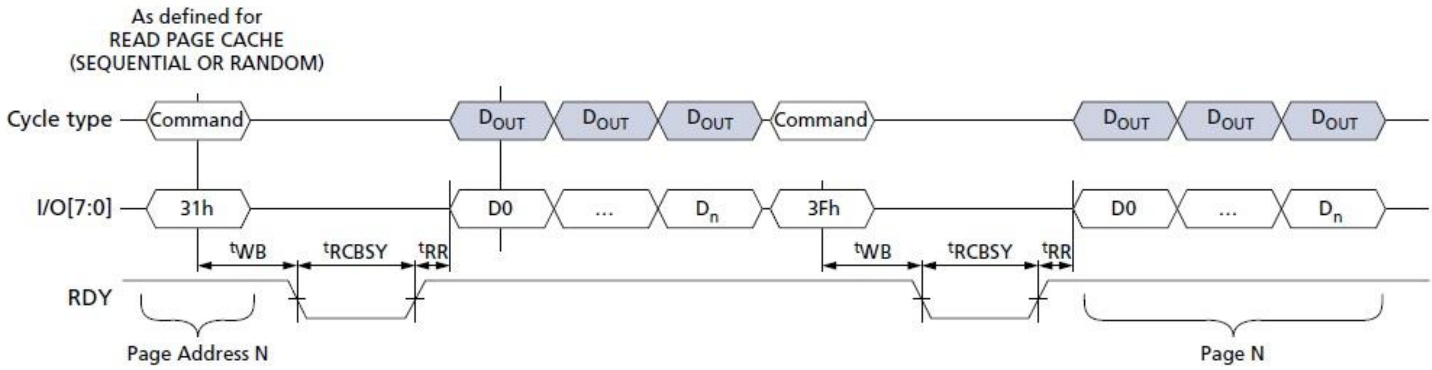
## READ PAGE CACHE LAST (3Fh)

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.

**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.

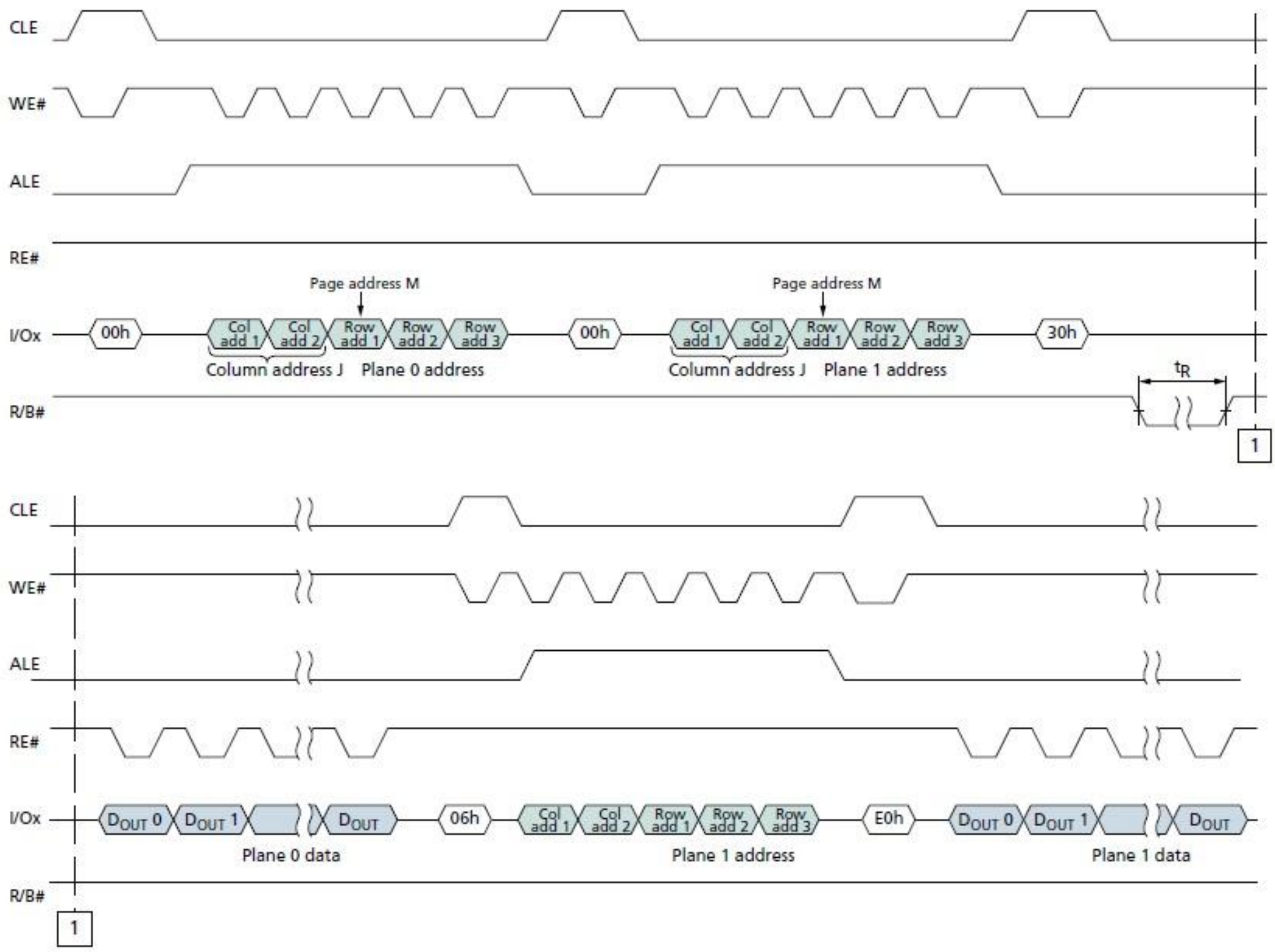
After the 30h command is written, page data is transferred from both planes to their respective data registers in $t_R$. During these transfers, R/B# goes LOW. When the transfers are complete, R/B# goes HIGH. To read out the data from the plane 0 data register, pulse RE# repeatedly. After the data cycle 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 completed, 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.

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



## Program Operations

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

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

### Program Operations

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

### Program Cache Operations

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

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

### Two-Plane Program Operations

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

### Two-Plane Program Cache Operations

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

## PROGRAM PAGE (80h-10h)

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

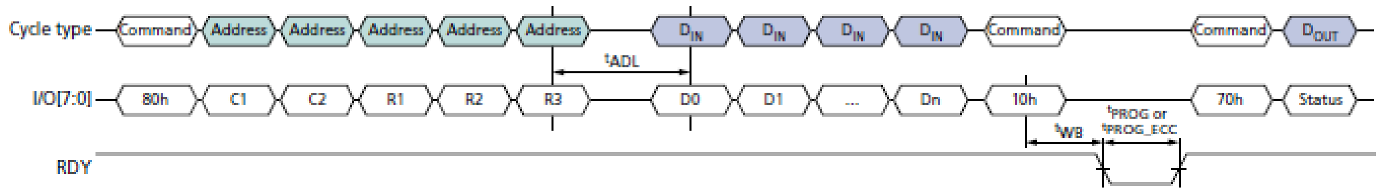
To input a page to the cache register and move it to the NAND array at the block and page address specified, write 80h to the command register. Unless this command has been preceded by a PROGRAM PAGE TWO-PLANE (80h-11h) command, issuing the 80h to the command register clears all of the cache registers' contents on the selected target. Then write 5 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).

**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 available 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

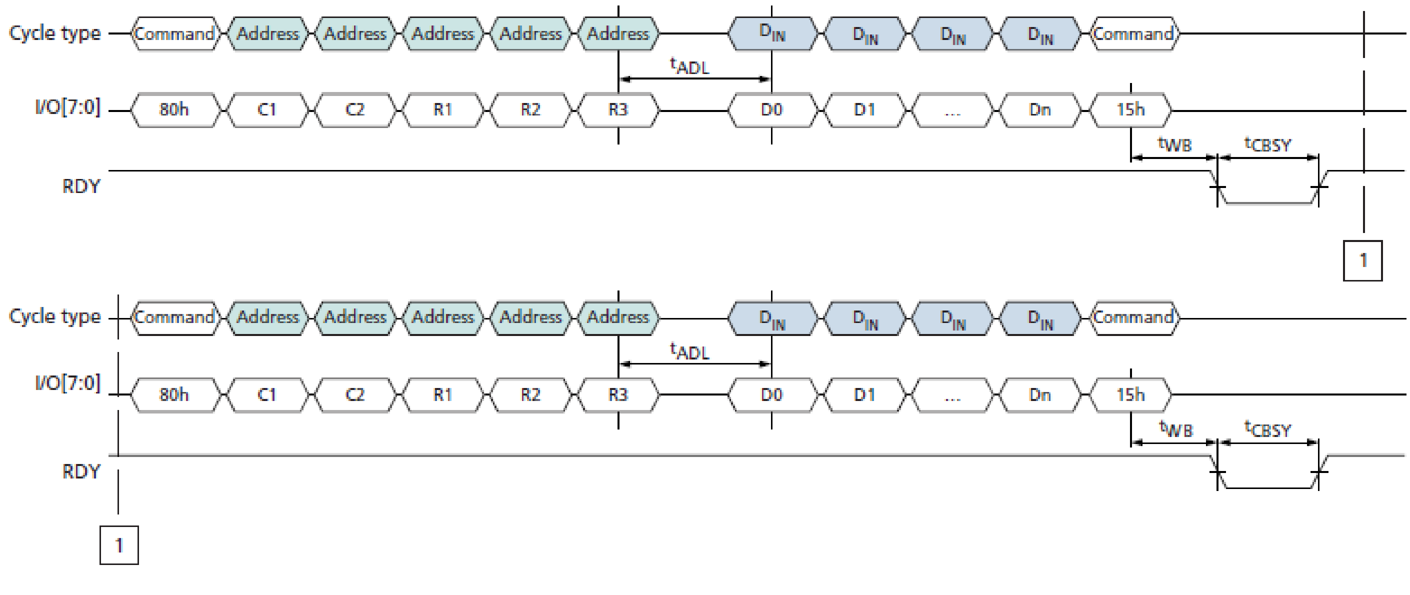
To determine the progress of $t_{CBSY}$, the host can monitor the target's R/B# signal or, alternatively, the status operations (70h, 78h) can be used. When the LUN's status shows that it is busy with a PROGRAM CACHE operation (RDY = 1, ARDY = 0), the host should check the status of the FAILC bit to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

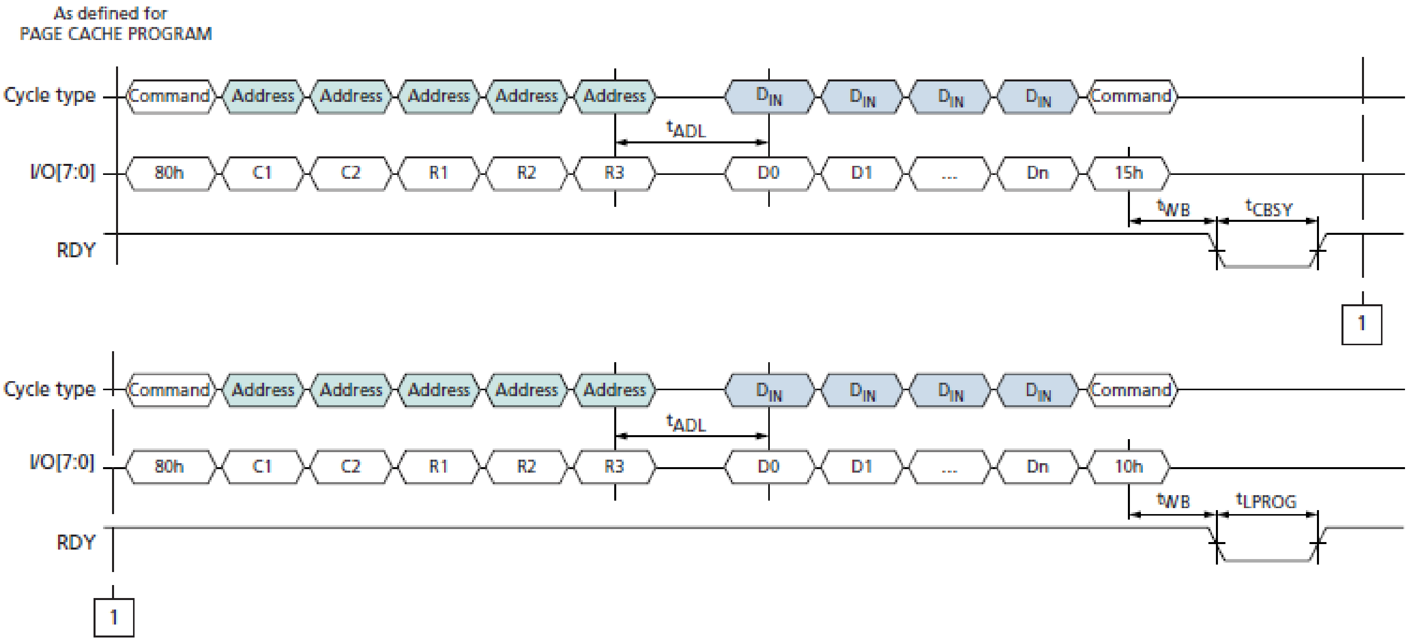
In devices with more than one die (LUN) per target, during and following interleaved die (multi-LUN) operations, the READ STATUS ENHANCED (78h) command must be used to select only one die (LUN) for status output. Use of the READ STATUS (70h) command could cause more than one die (LUN) to respond, resulting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

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



**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 $t_{DBSY}$.

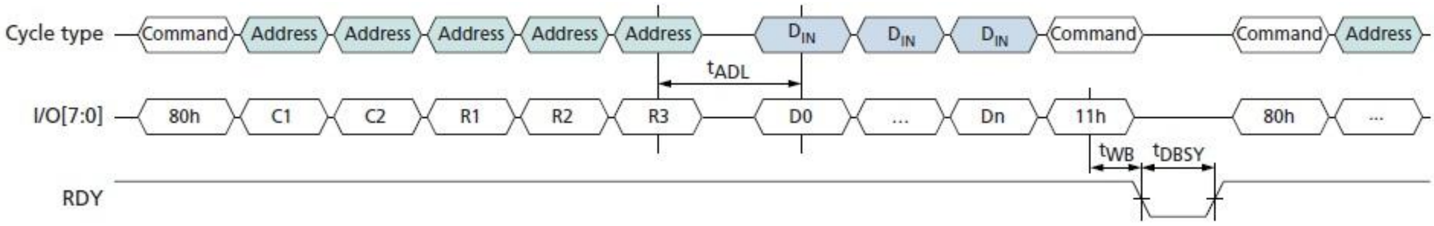
To determine the progress of $t_{DBSY}$, the host can monitor the target's R/B# signal or, alternatively, the status operations (70h, 78h) can be used. When the LUN's status shows that it is ready (RDY = 1), additional PROGRAM PAGE TWO-PLANE (80h-11h) commands can be issued to queue additional planes 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.

**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).

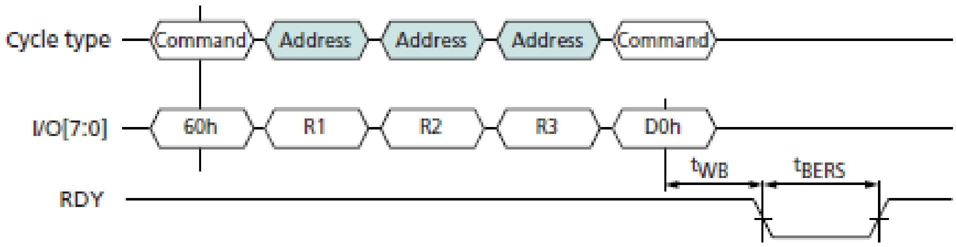
To erase a block, write 60h to the command register. Then write three address cycles containing the row address; the page address is ignored. Conclude by writing D0h to the command register. The selected die (LUN) will go busy (RDY = 0, ARDY = 0) for $t_{BERS}$ while the block is erased.

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.

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



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

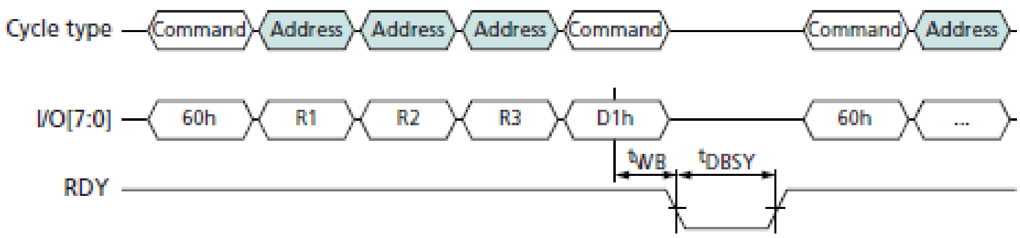
The ERASE BLOCK TWO-PLANE (60h-D1h) command queues a block in the specified plane to be erased in the NAND Flash array. This command can be issued one or more times. Each time a new plane address is specified, that plane is also queued for a block to be erased. To specify the final block to be erased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

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

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

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

**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.

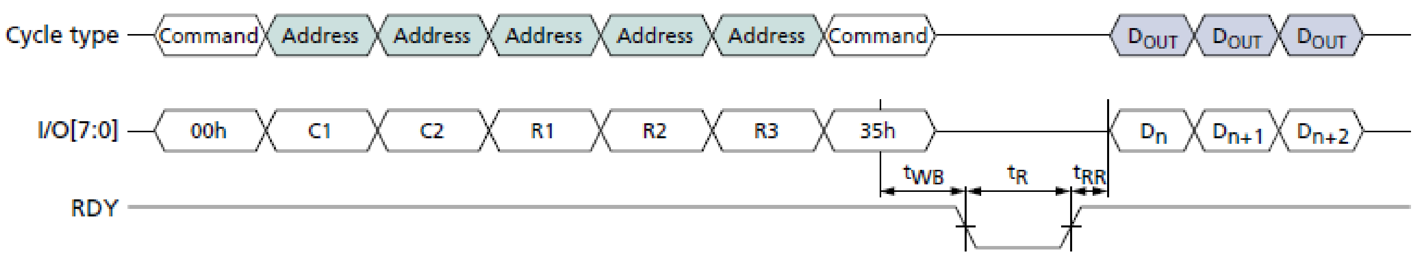
The INTERNAL DATA MOVE PROGRAM TWO-PLANE (85h-11h) command can be used to further system performance of PROGRAM FOR INTERNAL DATA MOVE operations by enabling movement of multiple pages from the cache registers to different planes of the NAND Flash array. This is done by prepending one or more PROGRAM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

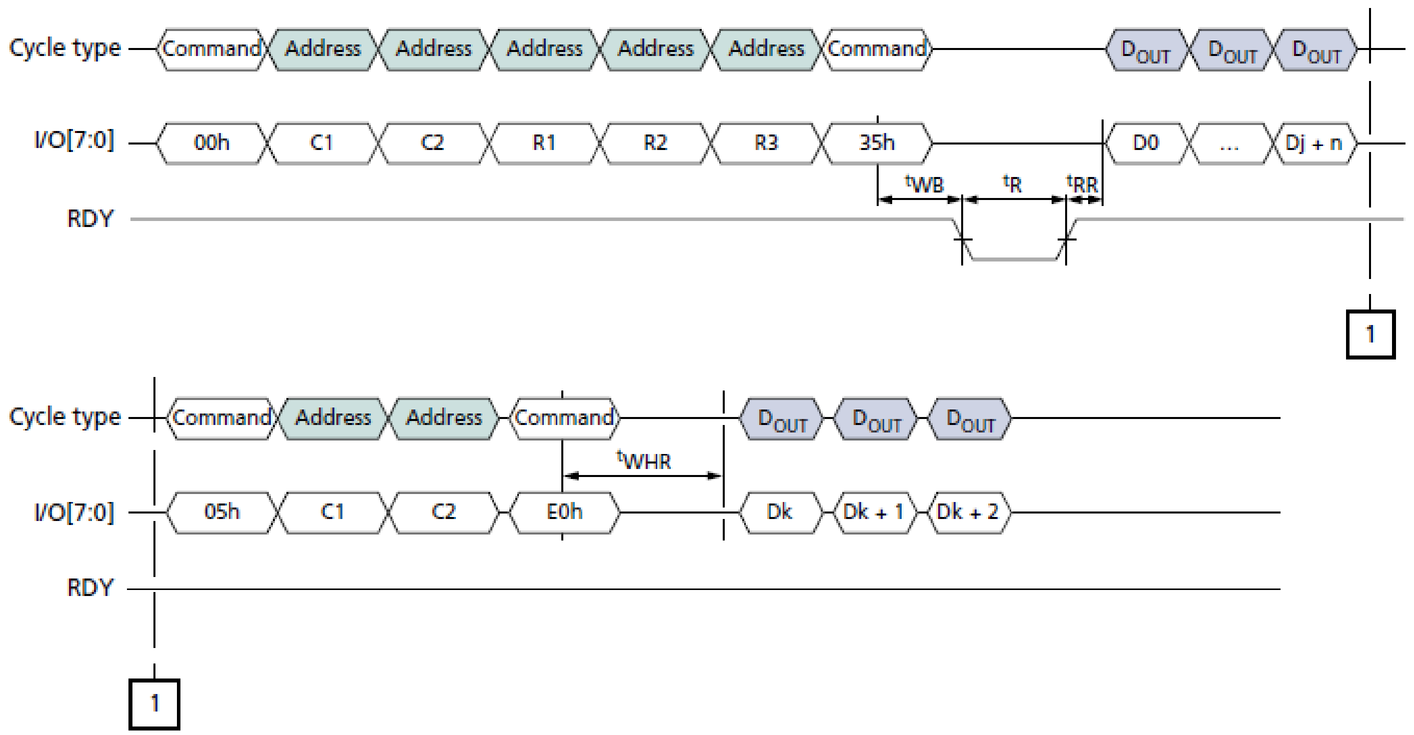
The READ FOR INTERNAL DATA MOVE (00h-35h) command is functionally identical to the READ PAGE (00h-30h) command, except that 35h is written to the command register instead of 30h.

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.

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



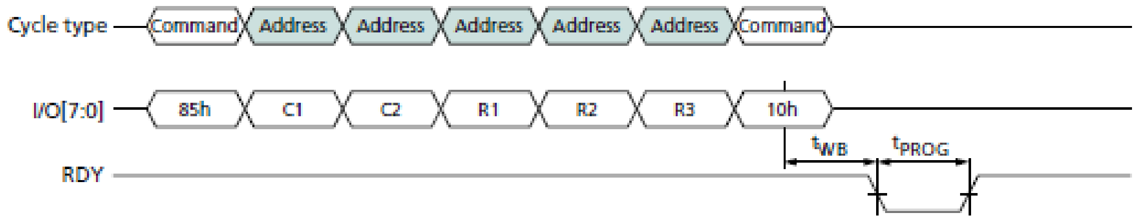

**READ FOR INTERNAL DATA MOVE (00h–35h) with RANDOM DATA READ (05h–E0h)**



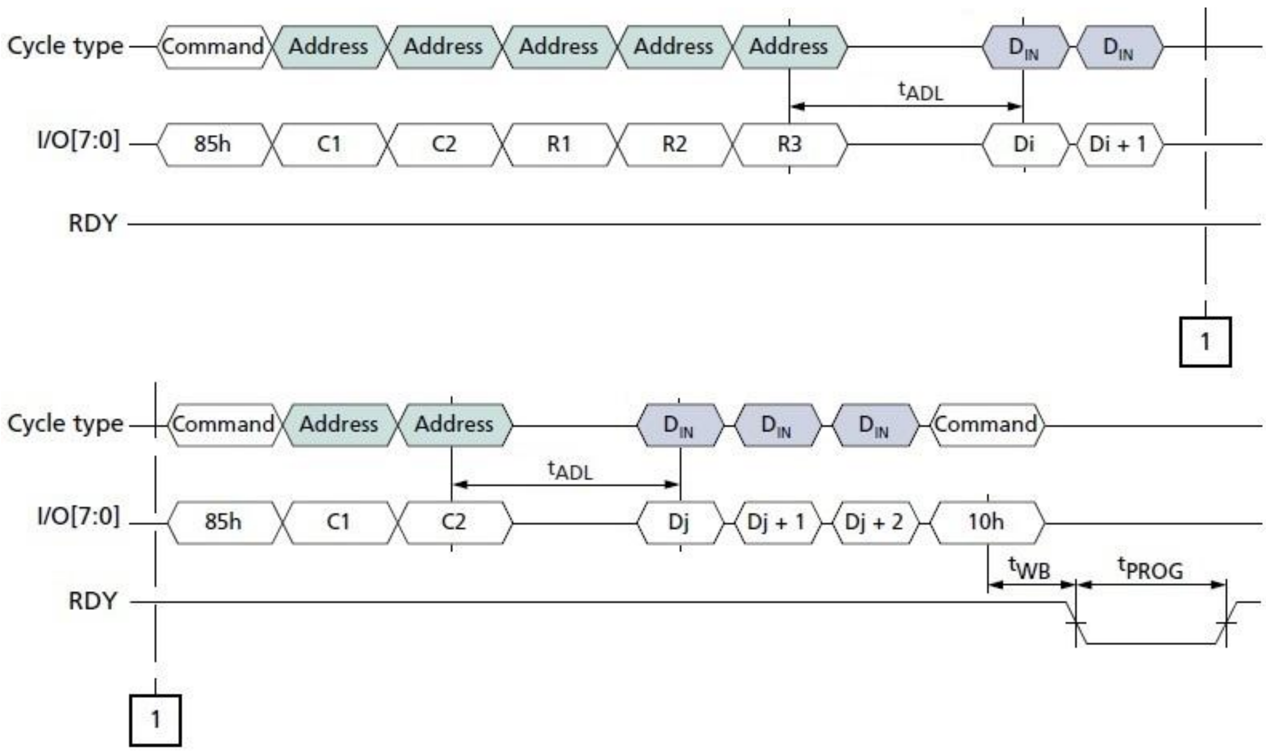
## PROGRAM FOR INTERNAL DATA MOVE (85h–10h)

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.

**PROGRAM FOR INTERNAL DATA MOVE (85h–10h) Operation**



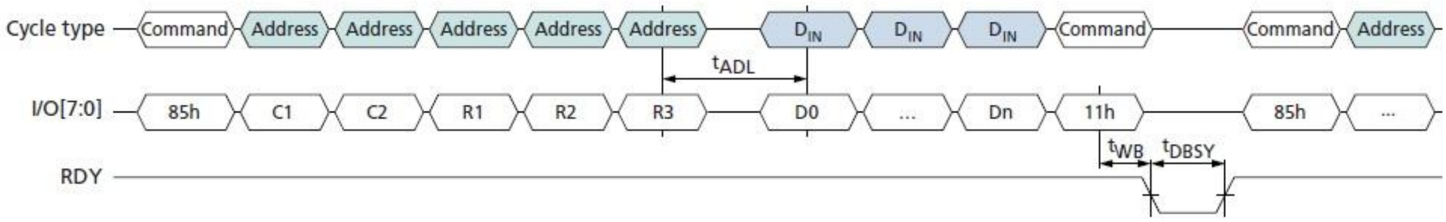

**PROGRAM FOR INTERNAL DATA MOVE (85h-10h) with RANDOM DATA INPUT (85h)**



## 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 (80h-11h) command, except that when 85h is written to the command register, cache register contents are not cleared. See Program Operations for further details.

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



## One-Time Programmable (OTP) Operations

This NAND Flash device offers a protected, one-time programmable NAND Flash memory area. Twenty-eight full pages 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.

This NAND Flash 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-1Dh. 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

The legacy OTP commands are OTP DATA PROGRAM (A0h-10h), OTP DATA PROTECT (A5h-10h), and OTP DATA READ (AFh-30h).

### 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 OTP page allows 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.

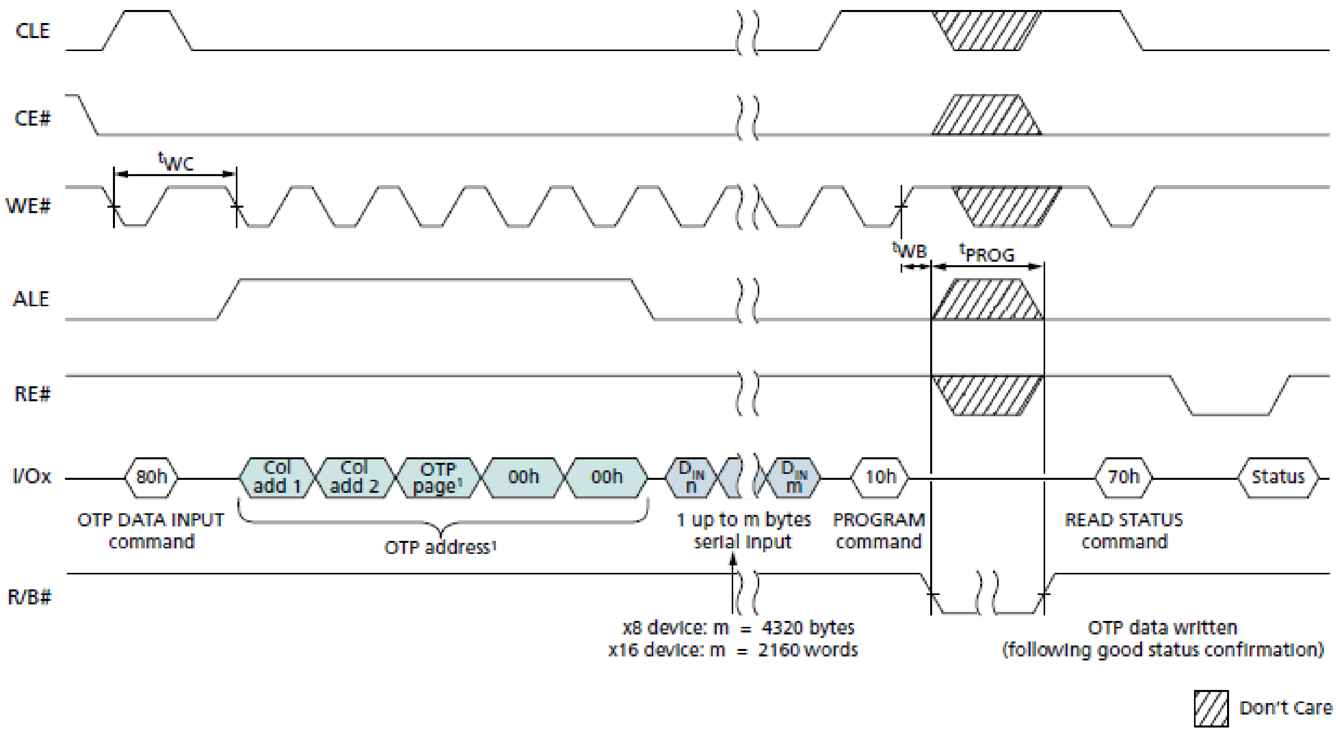
To use the PROGRAM PAGE command, issue the 80h command. Issue n address cycles. The first two address cycles are the column address. For the remaining cycles, select a page in the range of 02h-00h through 1Fh-00h. Next, write n bytes of data. After data input is complete, issue the 10h command. The internal 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 4 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.

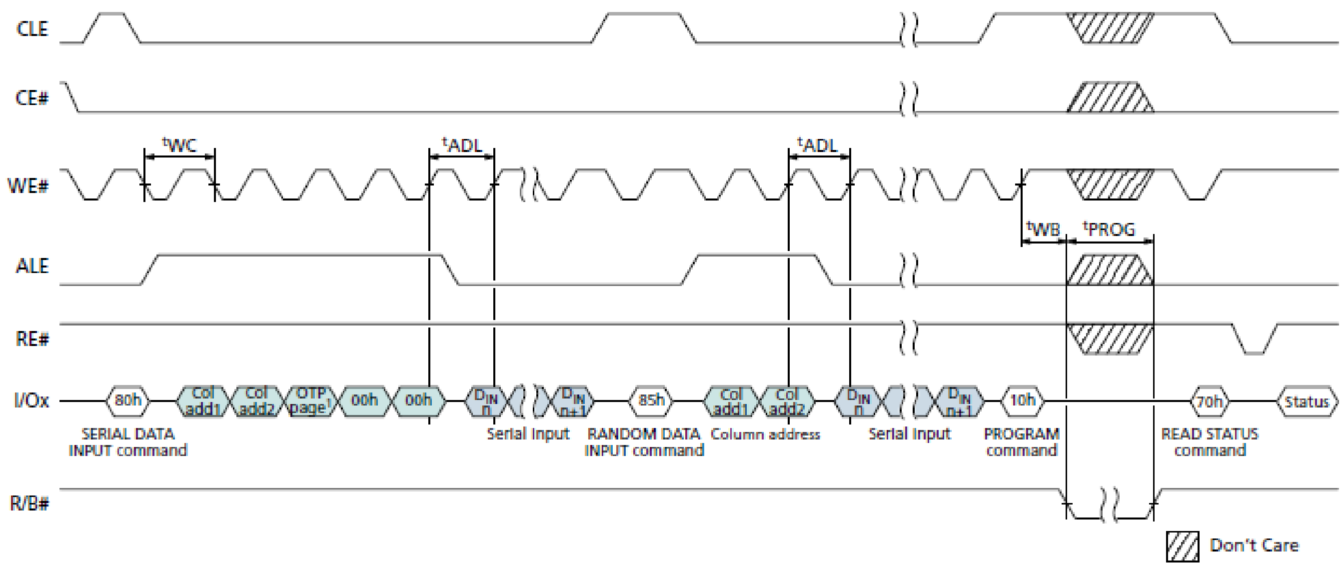
**OTP DATA PROGRAM (After Entering OTP Operation Mode)**




**NOTE:**

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

**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.

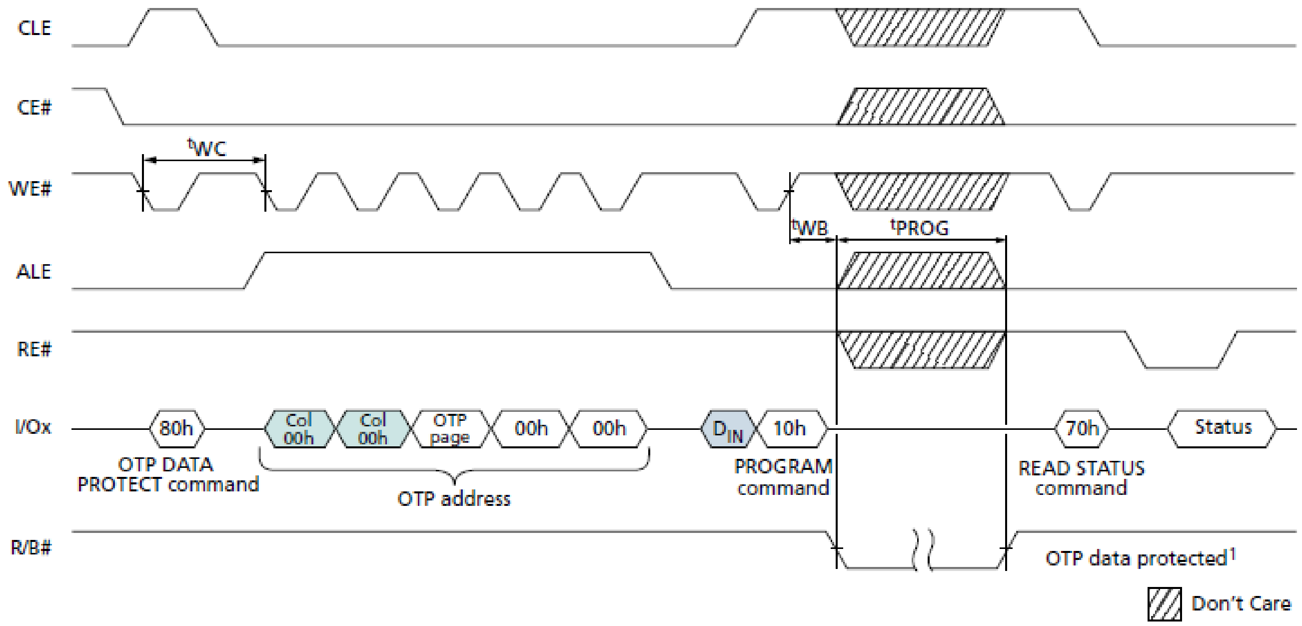
After the data is protected, it cannot be programmed further. When the OTP area is protected, the pages within the area are no longer programmable and cannot be unprotected.

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).

**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.

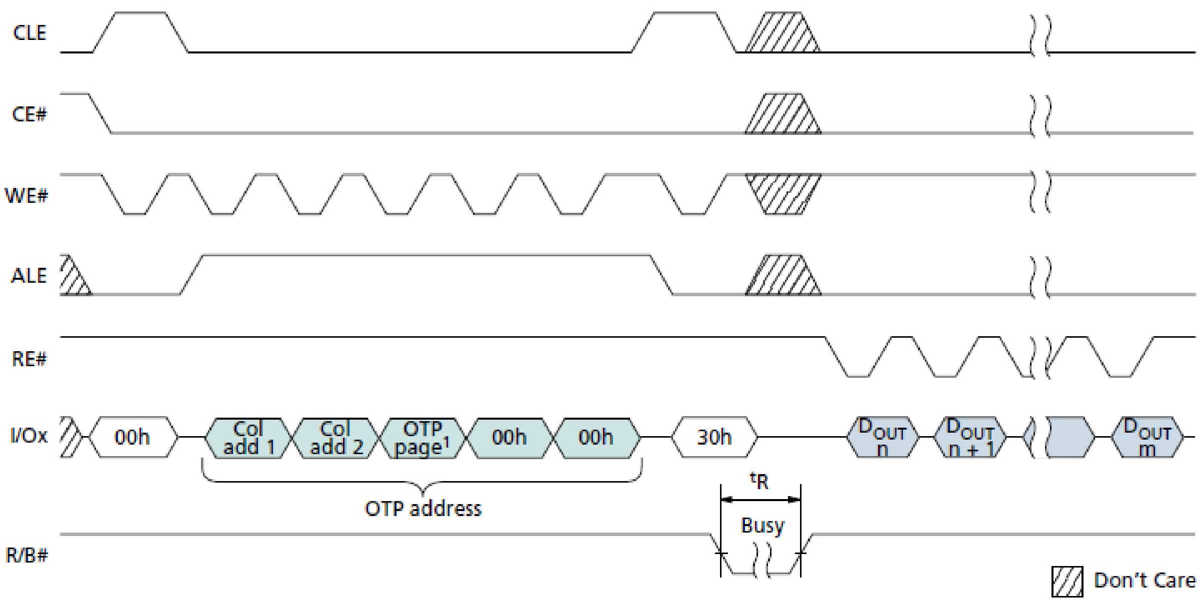
R/B# goes LOW (tR) while the data is moved from the OTP page to the data register. The READ STATUS (70h) command is the only valid command for reading status in OTP operation mode. Bit 5 of the status register reflects the state of R/B# (see Status Operations).

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.

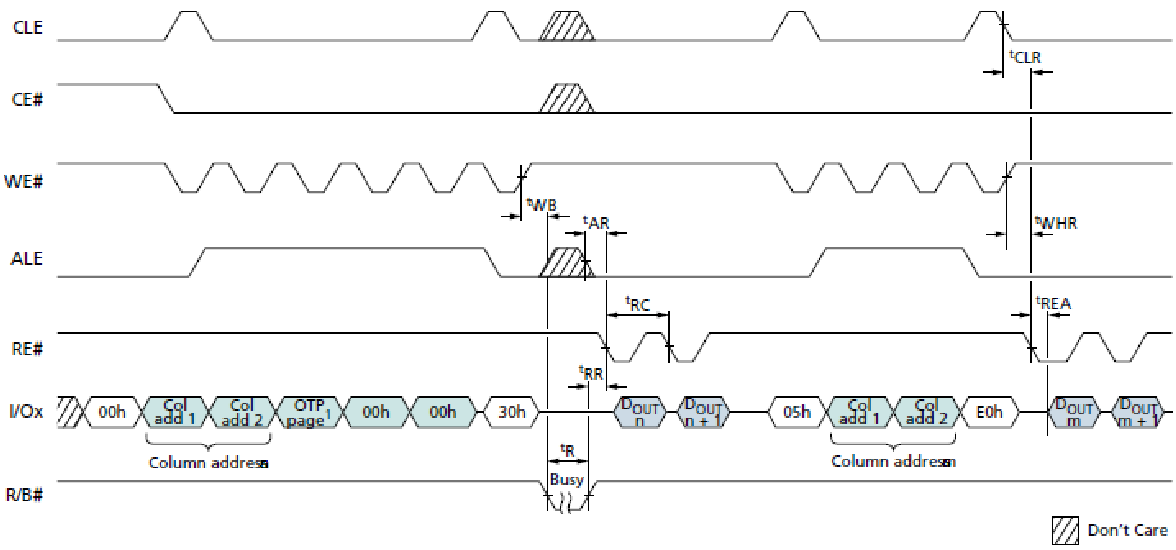
Only data on the current page can be read. Pulsing RE# outputs data sequentially.

**OTP DATA READ**




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

**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.

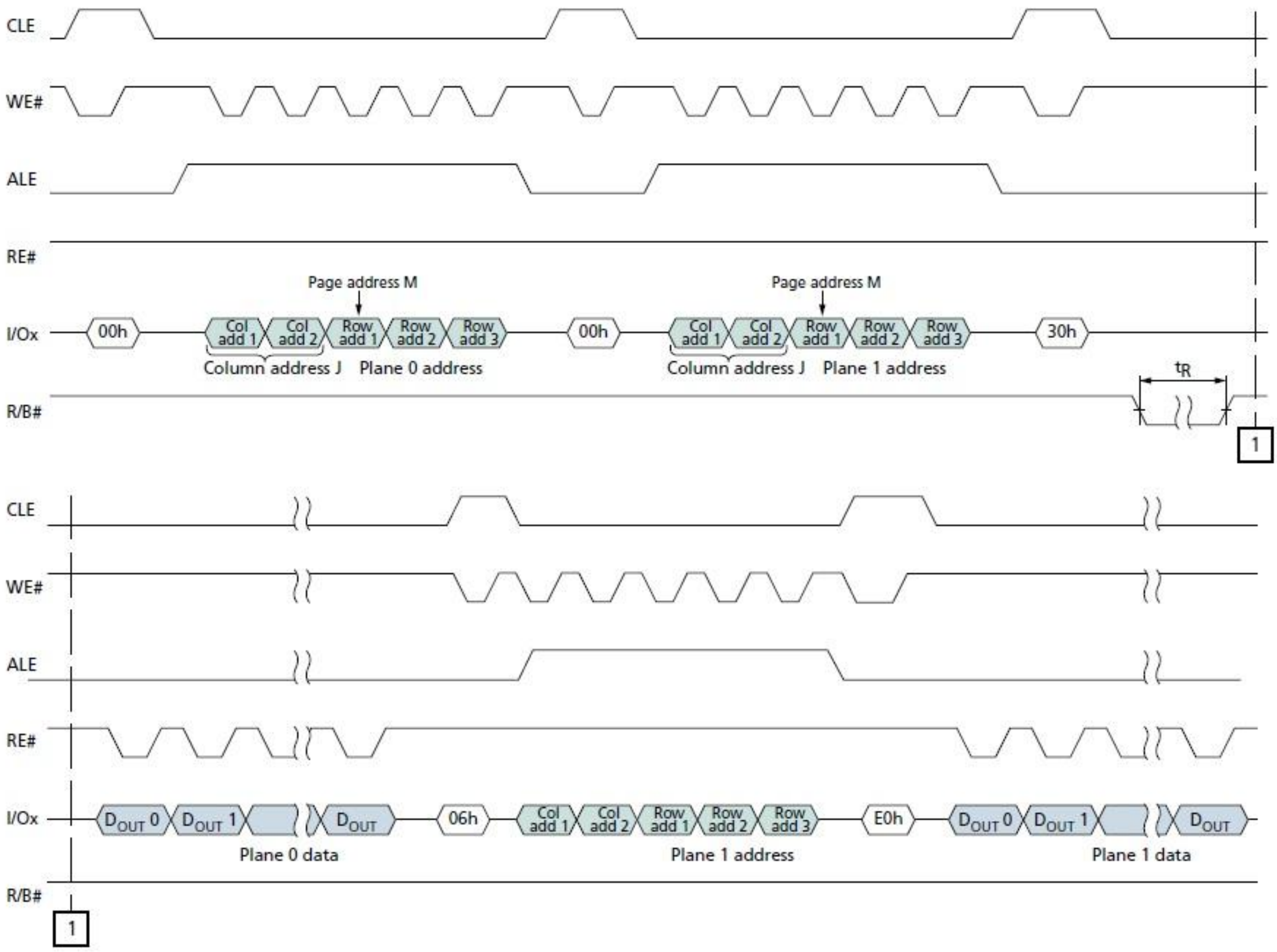
### Two-Plane Addressing

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).

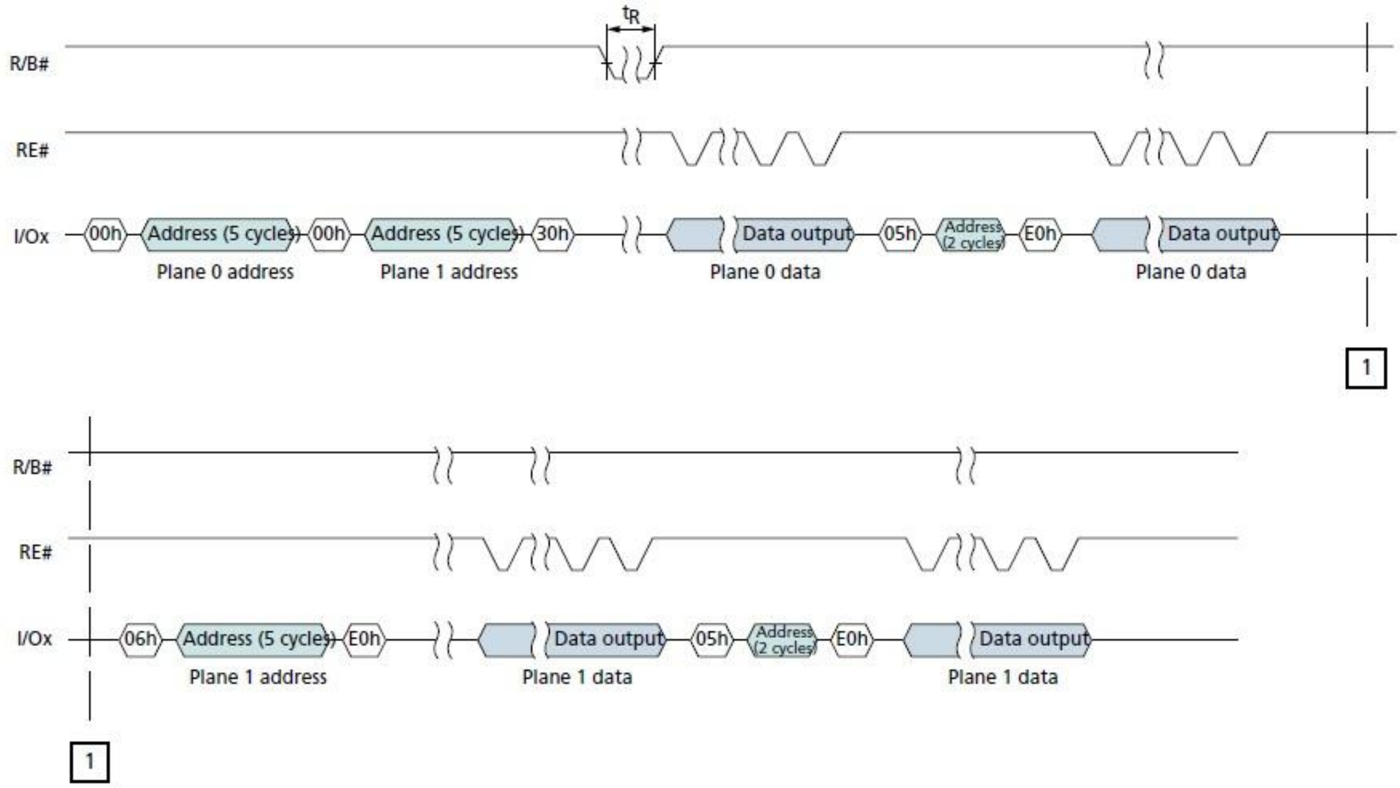
**TWO-PLANE PAGE READ**



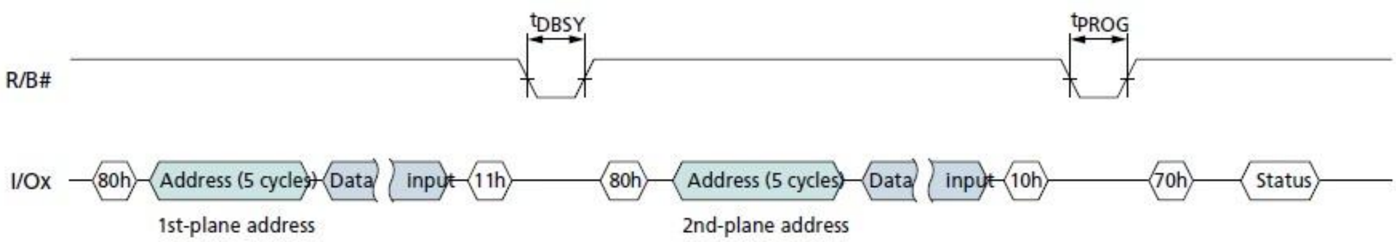

**NOTE:**

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.

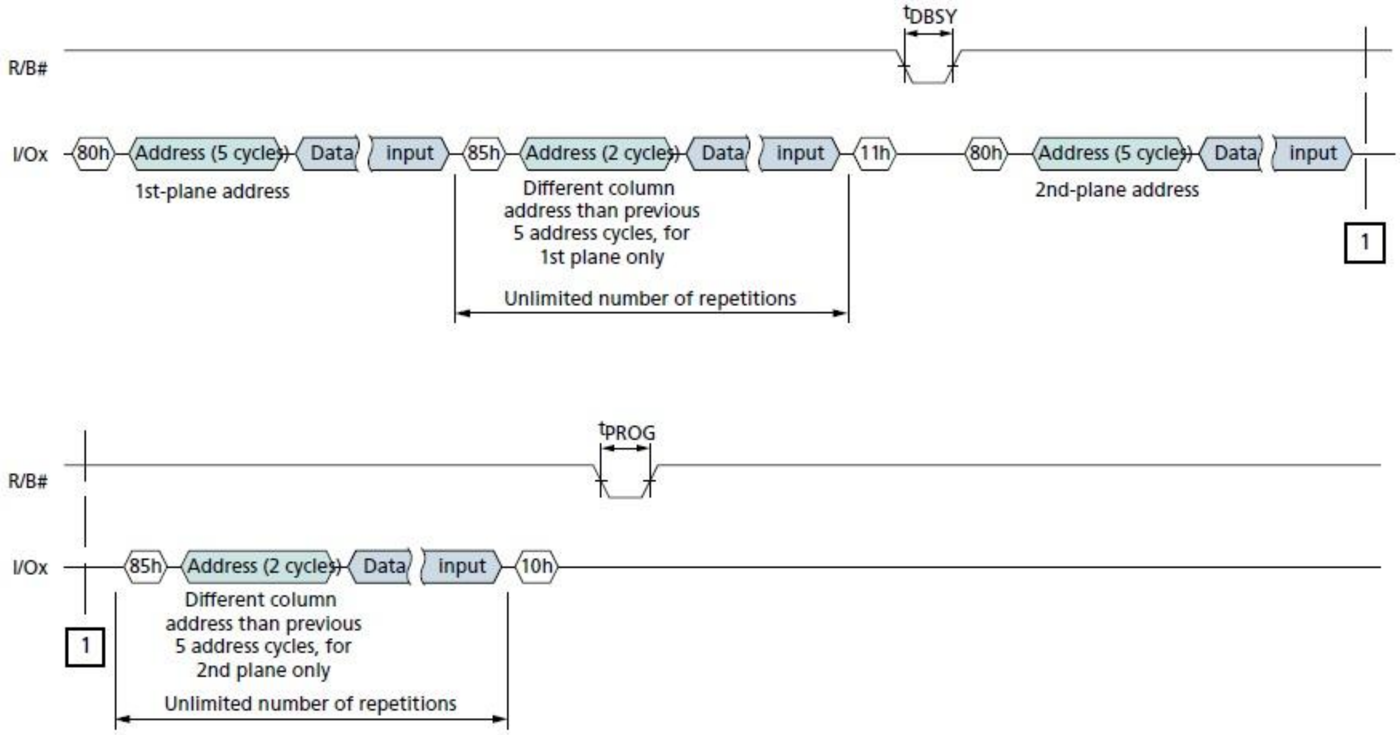
**TWO-PLANE PAGE READ with RANDOM DATA READ**



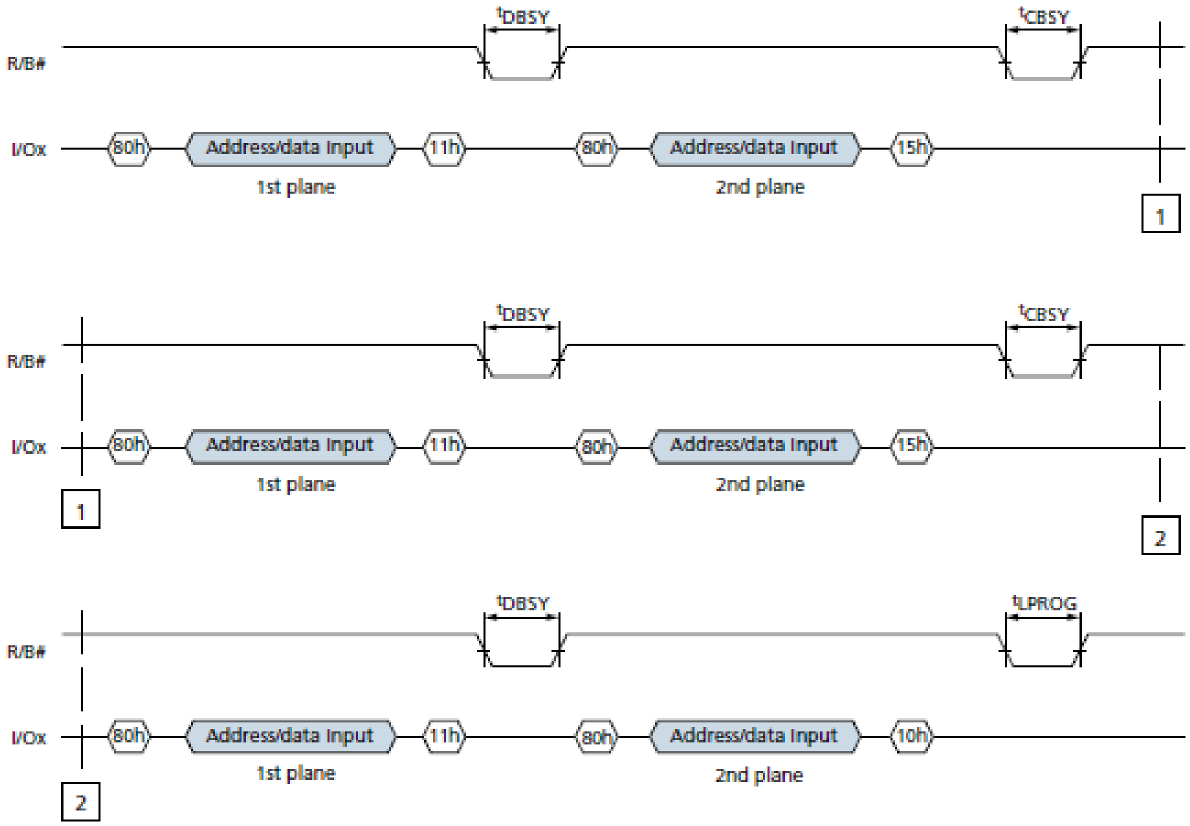

**TWO-PLANE PROGRAM PAGE**



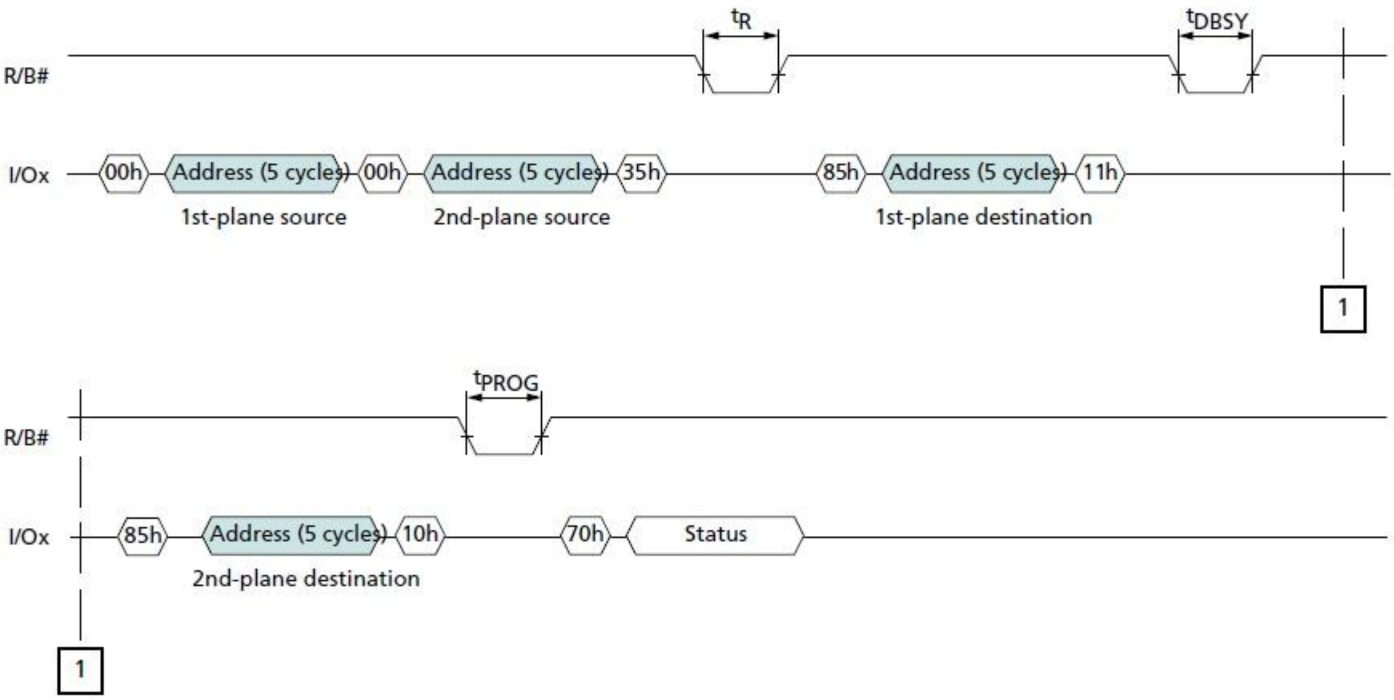
**TWO-PLANE PROGRAM PAGE with RANDOM DATA INPUT**



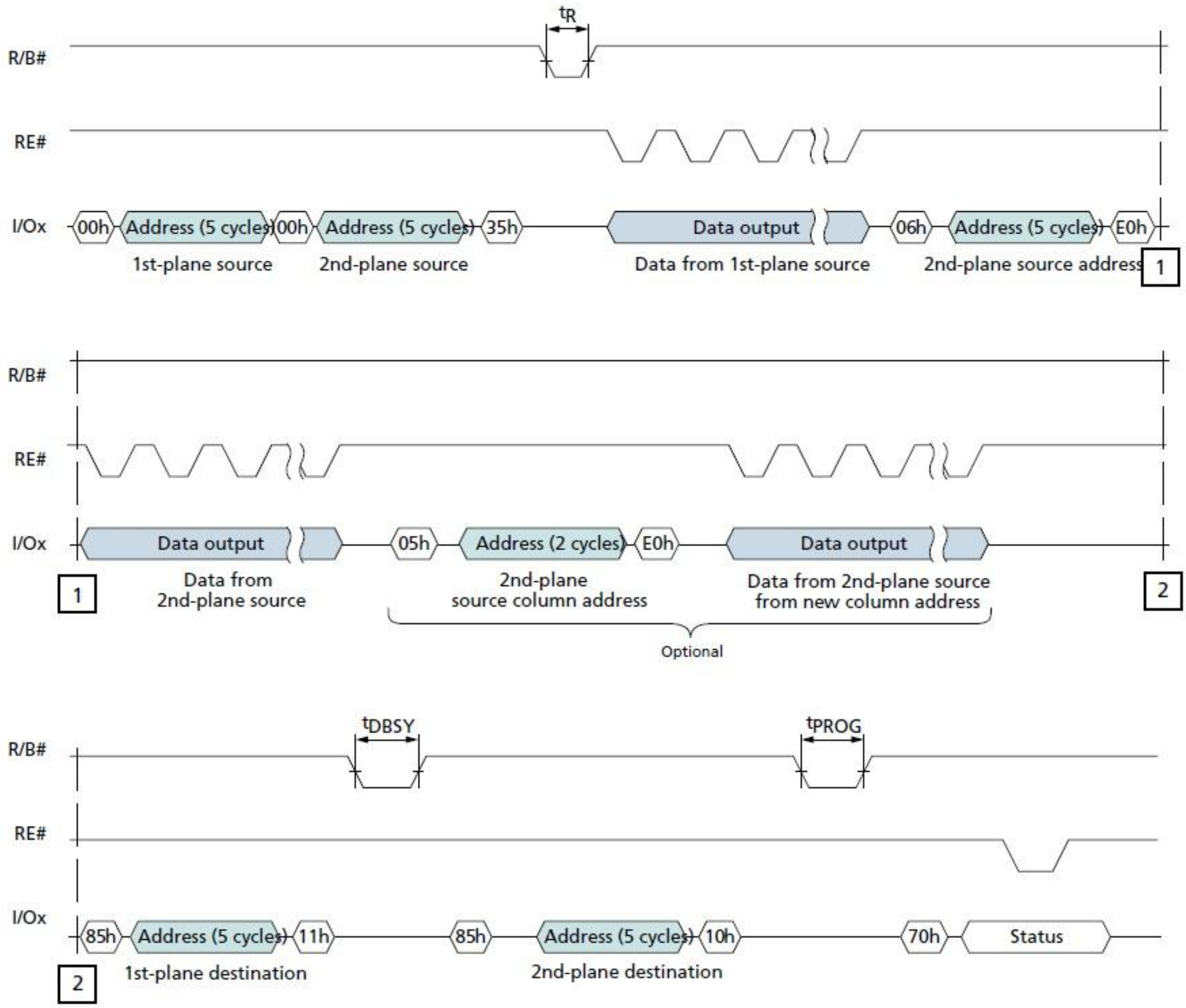
**TWO-PLANE PROGRAM PAGE CACHE MODE**



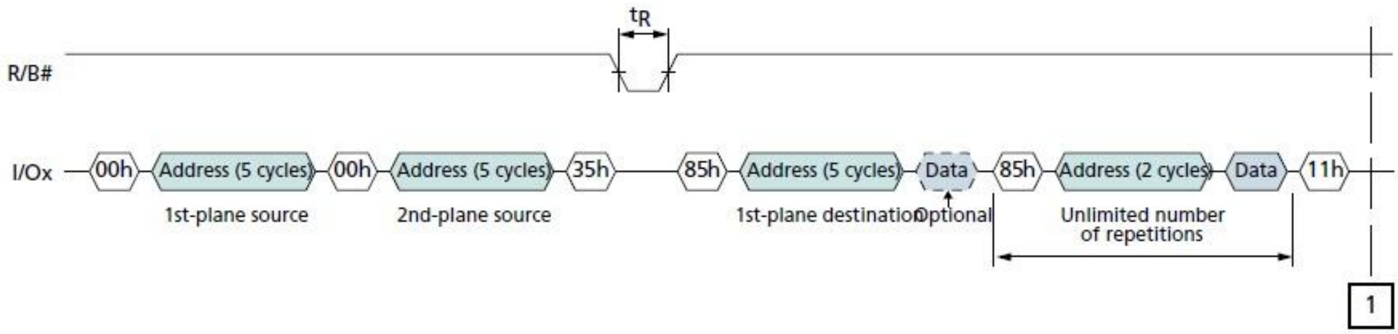

**TWO-PLANE INTERNAL DATA MOVE**

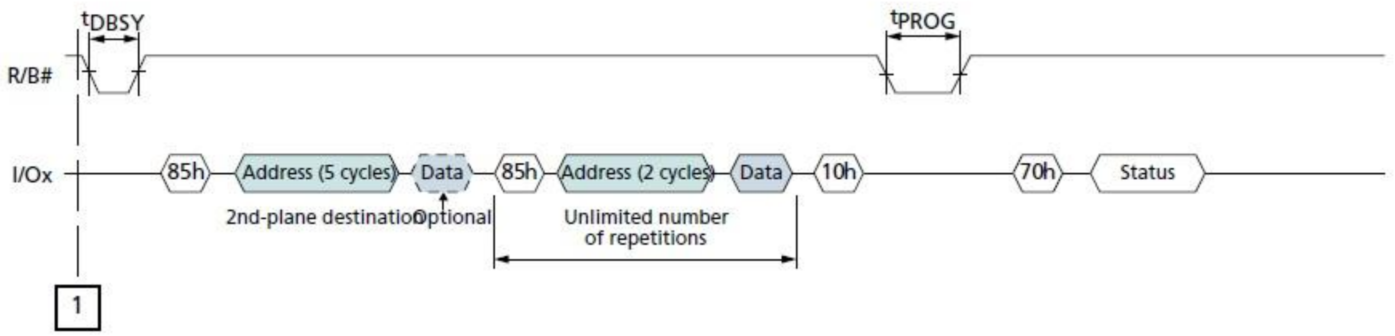


**TWO-PLANE INTERNAL DATA MOVE with TWO-PLANE RANDOM DATA READ**

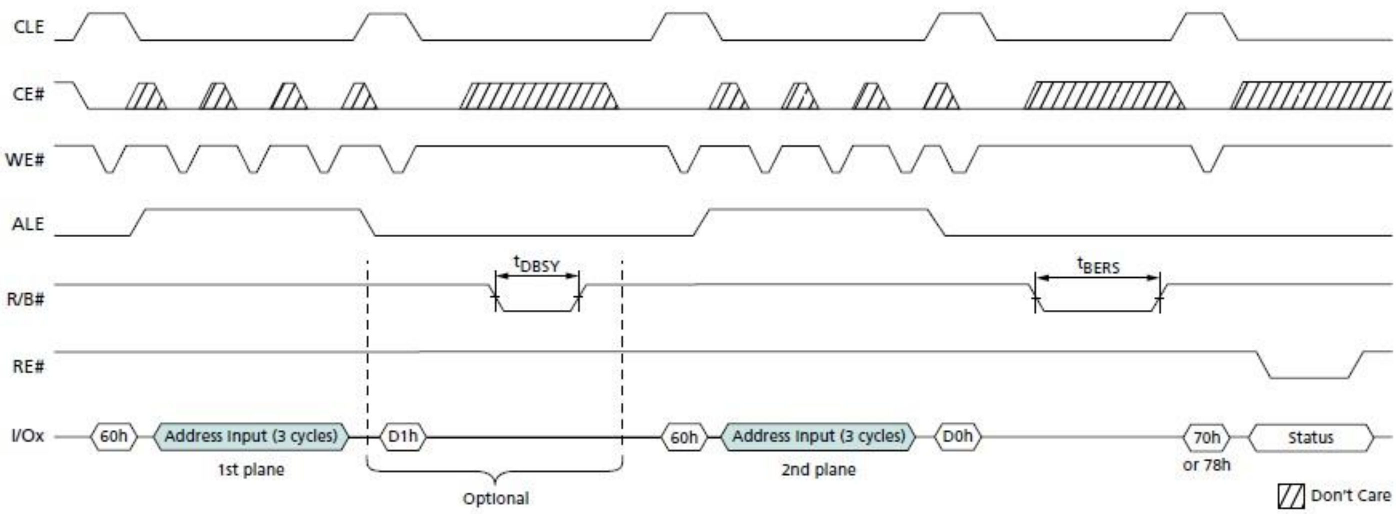


**TWO-PLANE INTERNAL DATA MOVE with RANDOM DATA INPUT**

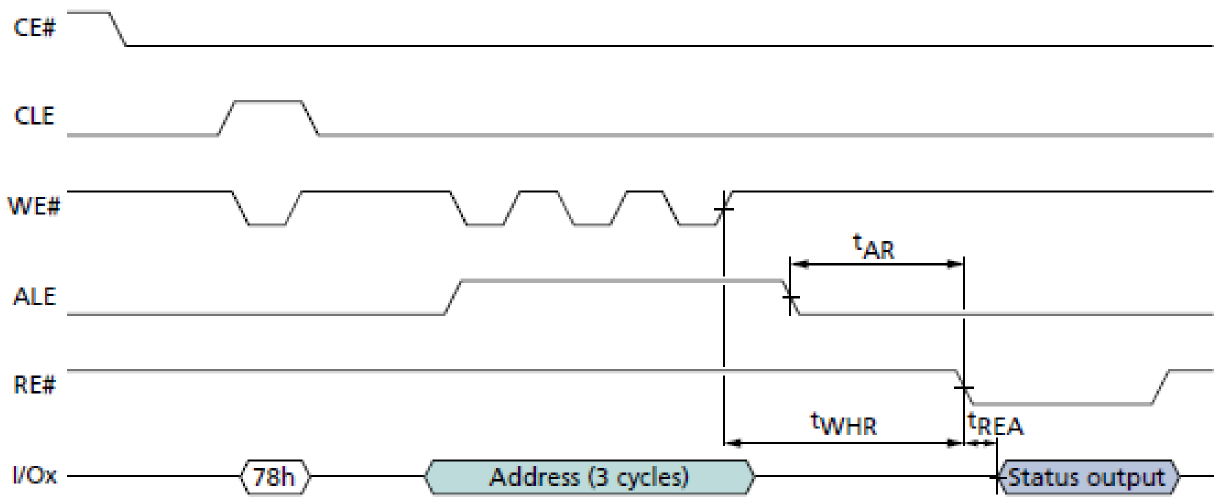


**TWO-PLANE BLOCK ERASE**




**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.

**Spare Area Mapping**

| Max Byte Address | Min Byte Address | ECC Protected | Area | Description |
|---|---|---|---|---|
| **User Data** | | | | |
| 1FFh | 000h | Yes | Main 0 | User data 0 |
| 3FFh | 200h | Yes | Main 1 | User data 1 |
| 5FFh | 400h | Yes | Main 2 | User data 2 |
| 7FFh | 600h | Yes | Main 3 | User data 3 |
| **User Meta Data** | | | | |
| 80Fh | 800h | Yes | Spare 0 | User meta data |
| 81Fh | 810h | Yes | Spare 1 | User meta data |
| 82Fh | 820h | Yes | Spare 2 | User meta data |
| 83Fh | 830h | Yes | Spare 3 | User meta data |
| **ECC** | | | | |
| 84Fh | 840h | Yes | Spare 0 | ECC for main/spare 0 |
| 85Fh | 850h | Yes | Spare 1 | ECC for main/spare 1 |
| 86Fh | 860h | Yes | Spare 2 | ECC for main/spare 2 |
| 87Fh | 870h | Yes | Spare 3 | ECC for main/spare 3 |

**ECC Status**

| Bit 4 | Bit 3 | Bit 0 | Description |
|---|---|---|---|
| 0 | 0 | 0 | No errors |
| 0 | 0 | 1 | Multiple bit errors were detected and not corrected. |
| 0 | 1 | 0 | 4 to 6 bit errors were detected and corrected. Refresh is recommended. |
| 0 | 1 | 1 | Reserved |
| 1 | 0 | 0 | 1 to 3 bit errors/page were detected and corrected. |
| 1 | 0 | 1 | Reserved |
| 1 | 1 | 0 | 7 to 8 bit errors were detected and corrected. Refresh is required to guarantee data retention. |

## Mask Out Initial Invalid Block(s)

Initial invalid blocks are defined as blocks that contain one or more initial invalid bits whose reliability is not guaranteed by ESMT. The information regarding the initial invalid block(s) is called the initial invalid block information. Devices with initial invalid block(s) have the same quality level as devices with all valid blocks and have the same AC and DC characteristics. An initial invalid block(s) does not affect the performance of valid block(s) because it is isolated from the bit line and the common source line by a select transistor. The system design must be able to mask out the initial invalid block(s) via address mapping.

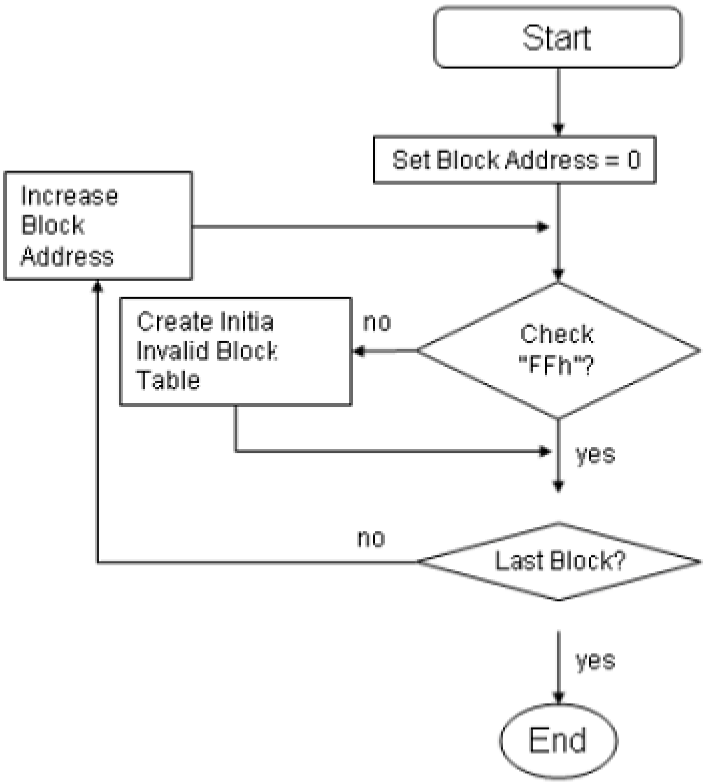
The 1st block, which is placed on 00h block address, is guaranteed to be a valid block up to 1K program/erase cycles with 8bit/544Byte ECC.

## Identifying Initial Invalid Block(s) and Block Replacement Management

All device locations are erased (FFh) except locations where the initial invalid block(s) information is written prior to shipping. The initial invalid block(s) status is defined by the 1st byte in the spare area. ESMT makes sure that either the 1st or 2nd page of every initial invalid block has non-FFh data at the 1st byte column address in the spare area.

Do not erase or program factory-marked bad blocks. The host controller must be able to recognize the initial invalid block information and to create a corresponding table to manage block replacement upon erase or program error when additional invalid blocks develop with Flash memory usage.

**Algorithm for Bad Block Scanning**

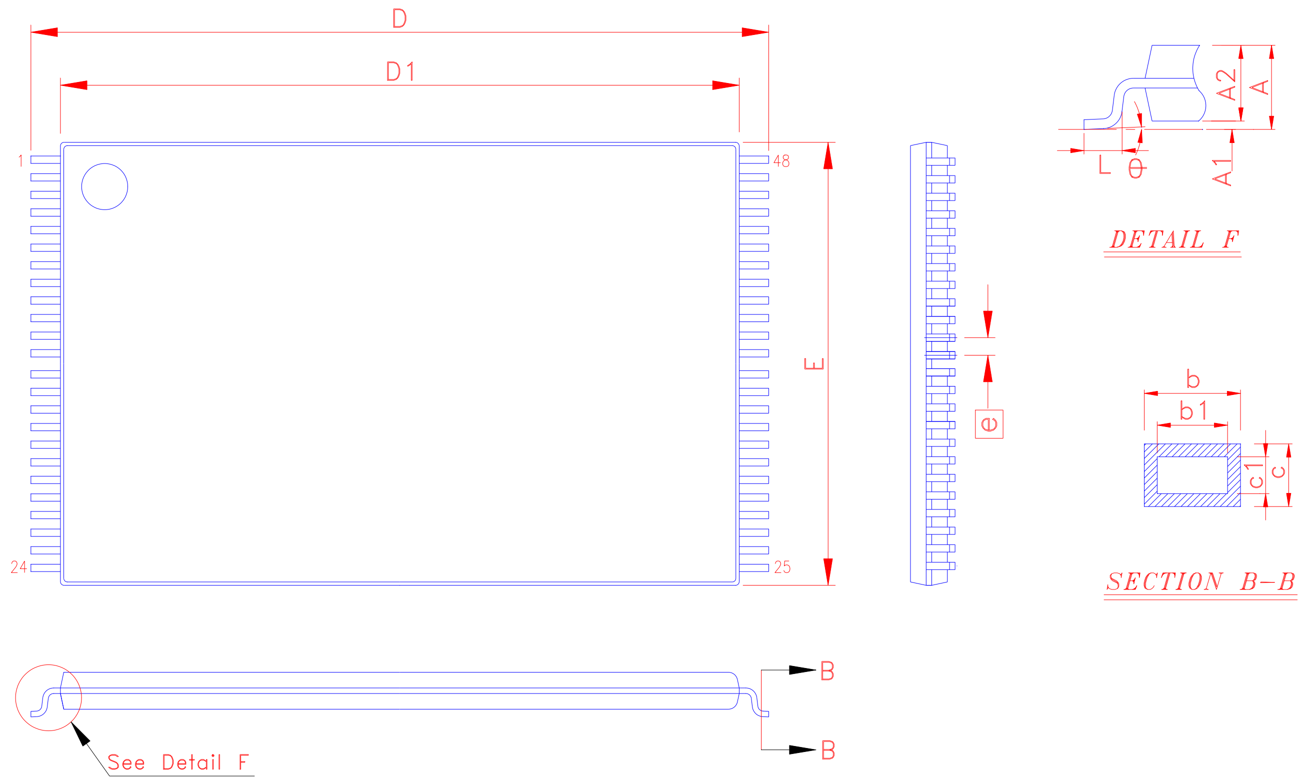



Check “FFh” at the 1st Byte column address in the spare area of the 1st and 2nd page in the block.

```
For (i=0; i<Num_of_LUs; i++)
{
        For (j=0; j<Blocks_Per_LU; j++)
        {
                Defect_Block_Found=False;

                Read_Page(lu=i, block=j, page=0);
                If (Data[coloumn=First_Byte_of_Spare_Area]!=FFh)    Defect_Block_Found=True;

                Read_Page(lu=i, block=j, page=1);
                If (Data[coloumn=First_Byte_of_Spare_Area]!=FFh)    Defect_Block_Found=True;

                If (Defect_Block_Found) Mark_Block_as_Defective(lu=i, block=j);
        }

}
```

## PACKING DIMENSIONS

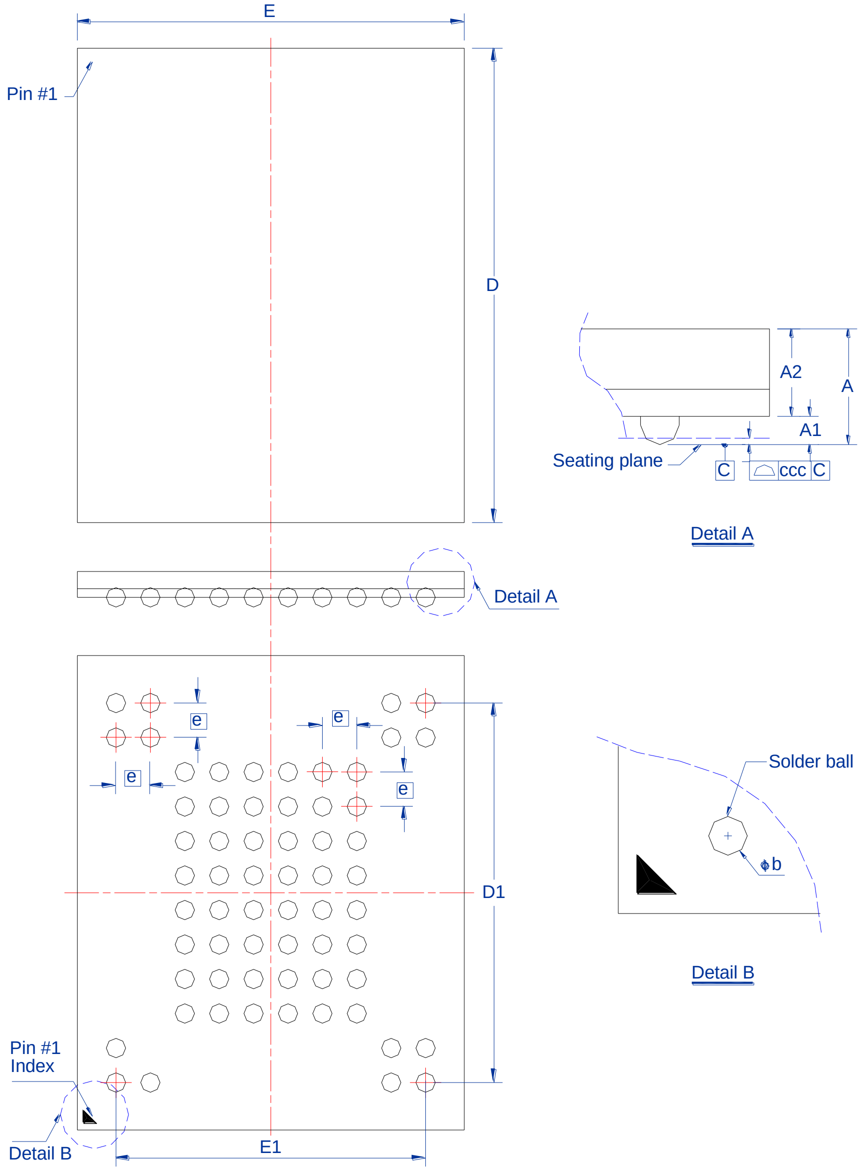
### 48-LEAD TSOP(I) ( 12x20 mm )




| Symbol | Dimension in mm | | | Dimension in inch | | | Symbol | Dimension in mm | | | Dimension in inch | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Min | Norm | Max | Min | Norm | Max | | Min | Norm | Max | Min | Norm | Max |
| A | ------- | ------- | 1.20 | ------- | ------- | 0.047 | D | 20.00 BSC | | | 0.787 BSC | | |
| A 1 | 0.05 | ------- | 0.15 | 0.006 | ------- | 0.002 | D 1 | 18.40 BSC | | | 0.724 BSC | | |
| A 2 | 0.95 | 1.00 | 1.05 | 0.037 | 0.039 | 0.041 | E | 12.00 BSC | | | 0.472 BSC | | |
| b | 0.17 | 0.22 | 0.27 | 0.007 | 0.009 | 0.011 | e | 0.50 BSC | | | 0.020 BSC | | |
| b1 | 0.17 | 0.20 | 0.23 | 0.007 | 0.008 | 0.009 | L | 0.50 | 0.60 | 0.70 | 0.020 | 0.024 | 0.028 |
| c | 0.10 | ------- | 0.21 | 0.004 | ------- | 0.008 | θ | $0^{o}$ | ------- | $8^{o}$ | $0^{o}$ | ------- | $8^{o}$ |
| c1 | 0.10 | ------- | 0.16 | 0.004 | ------- | 0.006 | | | | | | | |

## PACKING DIMENSIONS

## 63-BALL NAND Flash ( 9x11 mm )




| Symbol | Dimension in mm | | | Dimension in inch | | |
|---|---|---|---|---|---|---|
| | Min | Norm | Max | Min | Norm | Max |
| A | | | 1.00 | | | 0.039 |
| $A_1$ | 0.25 | | 0.35 | 0.010 | | 0.014 |
| $A_2$ | 0.60 BSC | | | 0.024 BSC | | |
| Φb | 0.40 | | 0.50 | 0.016 | | 0.020 |
| D | 10.90 | 11.00 | 11.10 | 0.429 | 0.433 | 0.437 |
| E | 8.90 | 9.00 | 9.10 | 0.350 | 0.354 | 0.358 |
| $D_1$ | 8.80 BSC | | | 0.346 BSC | | |
| $E_1$ | 7.20 BSC | | | 0.283 BSC | | |
| e | 0.8 BSC | | | 0.031 BSC | | |
| ccc | | | 0.10 | | | 0.004 |

**Controlling dimension : Millimeter.**

## Revision History

| Revision | Date | Description |
|---|---|---|
| 0.1 | 2018.04.02 | Original |
| 0.2 | 2018.05.07 | Modify the specification of Endurance |
| 0.3 | 2018.08.14 | 1. Modify the description of READ PARAMETER PAGE (ECh)<br>2. Revise the description of Algorithm for Bad Block Scanning |
| 1.0 | 2019.01.04 | Delete Preliminary |
| 1.1 | 2019.04.01 | Add 48 pin TSOPI package |
| 1.2 | 2021.09.03 | 1. Modify Feature description<br>2. Modify the specification of tWHR |

## Important Notice

All rights reserved.

No part of this document may be reproduced or duplicated in any form or by any means without the prior permission of ESMT.

The contents contained in this document are believed to be accurate at the time of publication. ESMT assumes no responsibility for any error in this document, and reserves the right to change the products or specification in this document without notice.

The information contained herein is presented only as a guide or examples for the application of our products. No responsibility is assumed by ESMT for any infringement of patents, copyrights, or other intellectual property rights of third parties which may result from its use. No license, either express , implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of ESMT or others.

Any semiconductor devices may have inherently a certain rate of failure. To minimize risks associated with customer's application, adequate design and operating safeguards against injury, damage, or loss from such failure, should be provided by the customer when making application designs.

ESMT's products are not authorized for use in critical applications such as, but not limited to, life support devices or system, where failure or abnormal operation may directly affect human lives or cause physical injury or property damage. If products described here are to be used for such kinds of application, purchaser must do its own quality assurance testing appropriate to such applications.