Flash

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

FEATURES

- Voltage Supply
  - VCC: 1.8V (1.7 V ~ 1.95V)
- Organization
  - Page Size: (2K + 128) bytes
  - Data Register: (2K + 128) bytes
  - Block Size: 64Pages = (128K + 8K) bytes
  - Number of Block per Die (LUN)= 2048
- Automatic Program and Erase
  - Page Program: (2K + 128) bytes
  - Block Erase: (128K + 8K) bytes
- Page Read Operation
  - Random Read: 25us (Max.)
  - Read Cycle: 45ns
- Write Cycle Time
  - Page Program Time: 400us (Typ.)
    700us (Max.)
  - Block Erase Time: 3.5ms (Typ.)
    10ms (Max.)
- 1bit/cell
- Command/Address/Data Multiplexed DQ Port
- Hardware Data Protection
  - Program/Erase Lockout During Power Transitions
- Reliable CMOS Floating Gate Technology
  - ECC Requirement: 8bit / 512Byte
  - Endurance: 50K-P/E Cycle Times
  - Uncycled Data Retention: 10 years of real time use at 55°C
- Command Register Operation
- Number of partial program cycles in the same page (NOP) : 4
- Automatic Page 0 Read at Power-Up Option
  - Boot from NAND support
  - Automatic Memory Download
- Cache Program Operation for High Performance Program
- Cache Read Operation
- Copy-Back Operation
- Two-Plane Operation
- EDO mode
- Page copy

ORDERING INFORMATION

Product ID	Speed	Package	Comments
F59D2G81KA -45TIAG2N	45 ns	48 pin TSOPI	Pb-free
F59D2G81KA -45BIAG2N	45 ns	63 ball BGA	Pb-free
F59D2G81KA -45BCIAG2N	45 ns	67 ball BGA	Pb-free

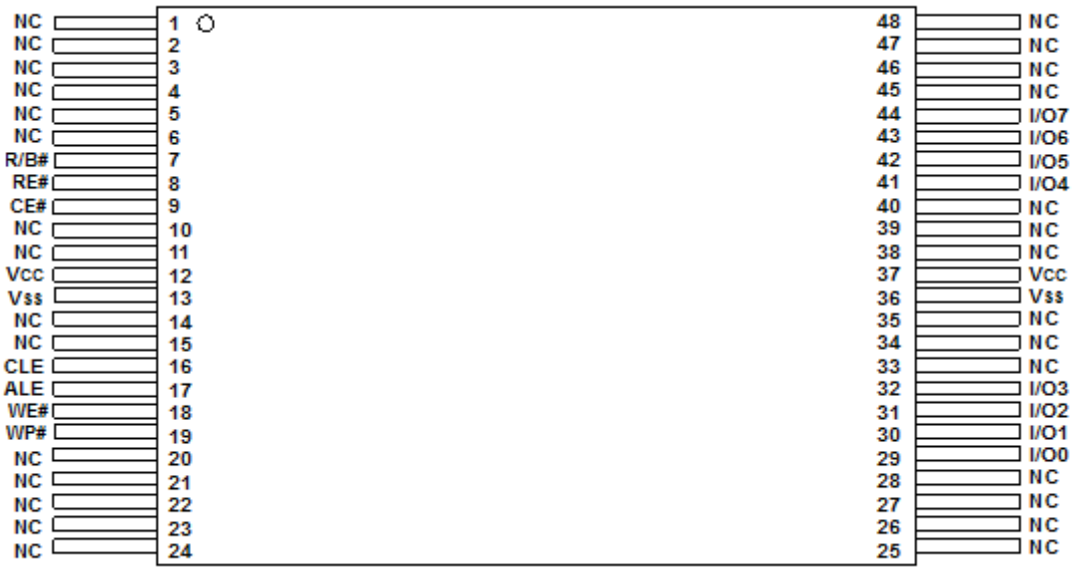
GENERAL DESCRIPTION

The device has two 2176-byte static registers which allow program and read data to be transferred between the register and the memory cell array in 2176-byte increments. The Erase operation is implemented in a single block unit (128Kbytes + 8Kbytes).

The device is a memory device which utilizes the I/O pins for both address and data input/output as well as command inputs. The Erase and Program operations are automatically executed making the device most suitable for applications such as solid state file storage, voice recording, image file memory for still cameras and other systems which require high density non-volatile memory data storage.

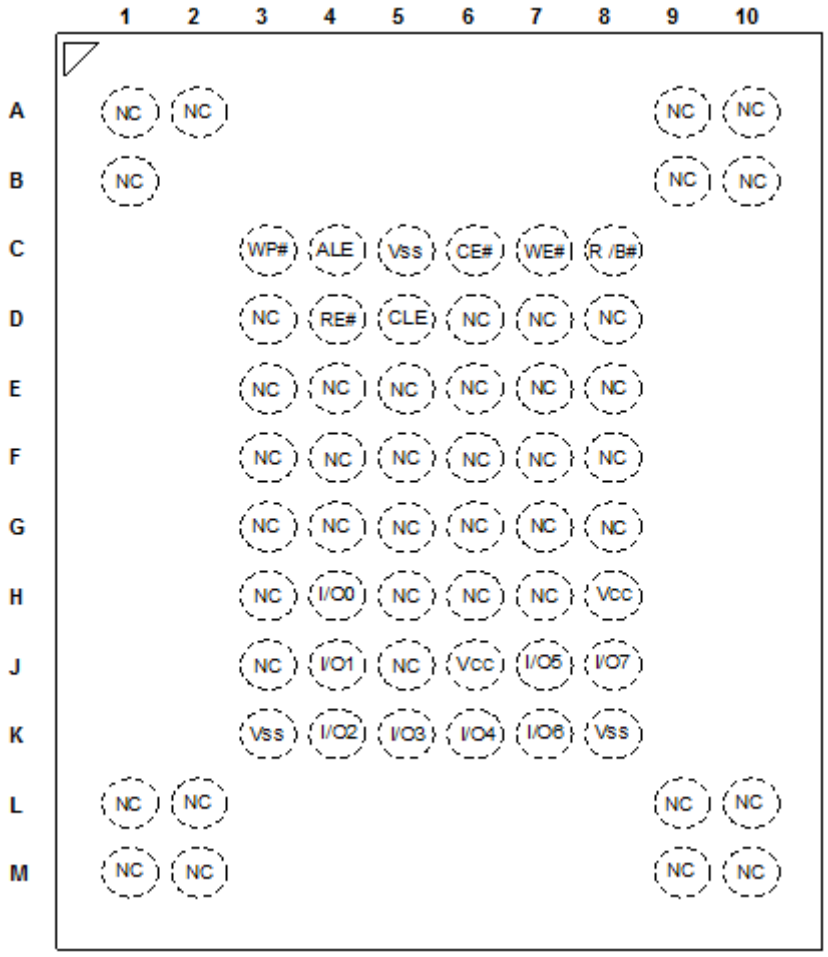
PIN CONFIGURATION (TOP VIEW)

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



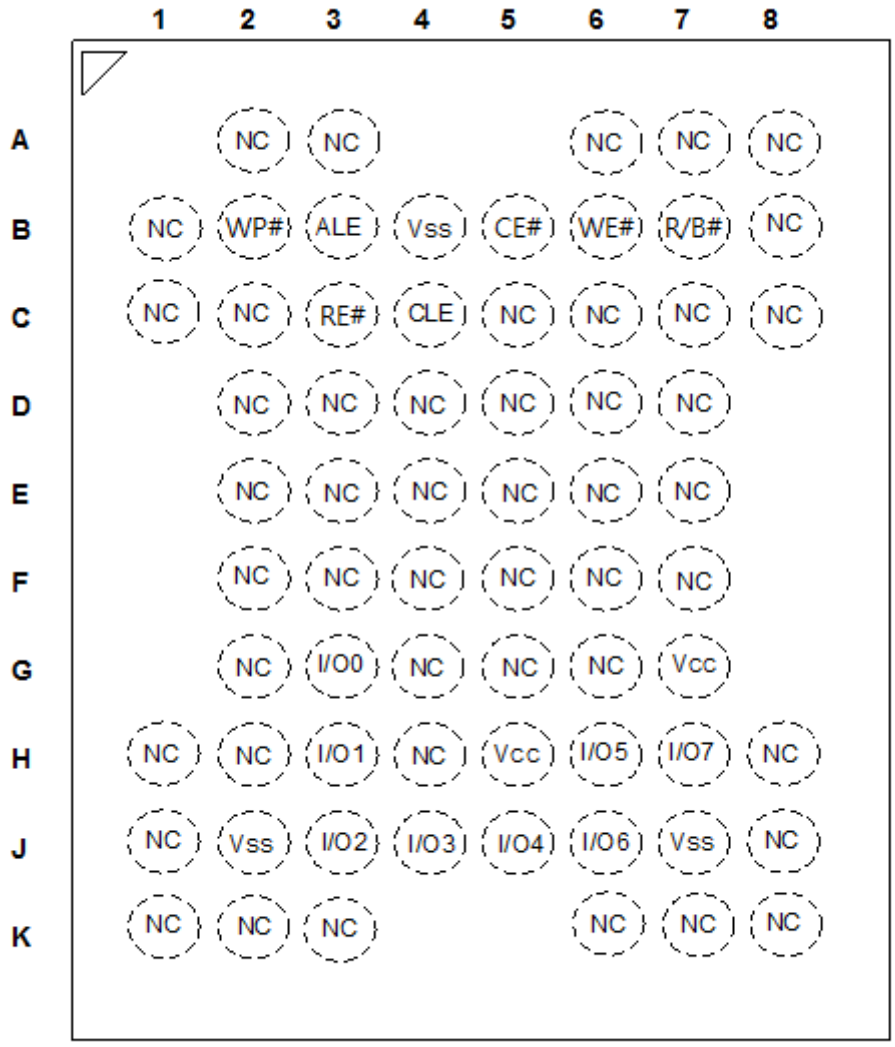
BALL CONFIGURATION (x8) (TOP VIEW)

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



BALL CONFIGURATION (TOP VIEW)

(BGA 67 Ball, 6.5mmx8mmx1.0mm Body, 0.8mm Ball Pitch)



PIN /BALL NAMES

Pin/Ball Name	Type	Function
V_{CC}	Supply	NAND Power Supply
V_{SS}	Supply	Ground
I/O0 to I/O7	Input/output	**Data inputs/outputs:** The I/O0 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.
V_{CC}	Supply	NAND Power Supply

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

Definitions and Abbreviations

LSB
Acronym for the least significant bit.

Address
The address is comprised of a column address 2 cycles and a row address with 3 cycles. The row address identifies the page, block, and LUN to be accessed. The column address identifies the byte within a page to access.

Column
The byte location within the page register.

Row
Refer to the block and page to be accessed.

Page
The smallest addressable unit for the Read and the Program operations.

Block
Consists of multiple pages and is the smallest unit for the Erase operation.

Page register
Register used to transfer data to and from the Flash Array.

Cache register
Register used to transfer data to and from the Host.

Defect area
The defect area is where the factory defects are marked by the manufacturer. It is a reference for initial invalid block(s).

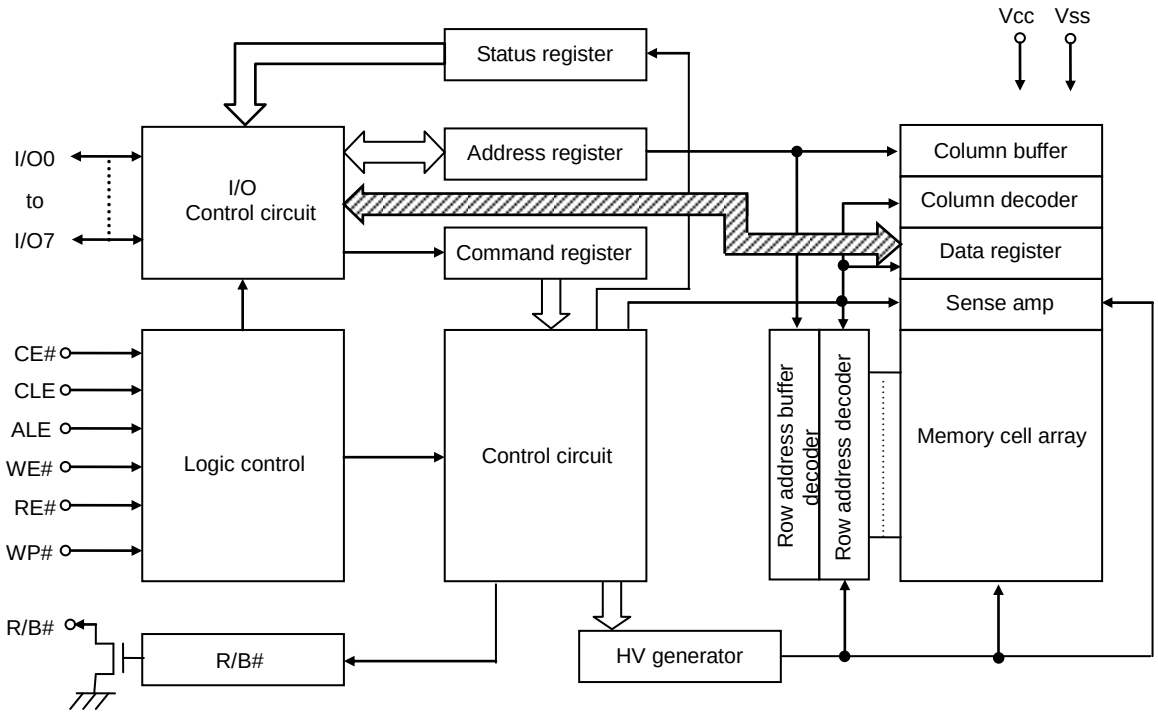
Device
The packaged NAND unit. A device may contain more than a target.

LUN (Logical Unit Number)
The minimum unit that can independently execute commands and report status. There are one or more LUNs per CE#.

Target
An independent NAND Flash component with its own CE# signal.

SR[x] (Status Read)
SR refers to the status register contained within a particular LUN. SR[x] refers to bit x in the status register for the associated LUN.

Block Diagram



Absolute Maximum Rating

Parameter	Symbol	Rating	Unit
Voltage on any pin relative to V_{SS}	V_{CC}	-0.6 to +2.5	V
	V_{IN}	-0.6 to +2.5	
	$V_{I/O}$	-0.6 to Vcc+0.3(≦2.5V)	
Short Circuit Current	I_{OS}	5	mA

Note: Permanent device damage may occur if ABSOLUTE MAXIMUM RATINGS are exceeded. Functional operation should be restricted to the conditions as detailed in the operational sections of this data sheet. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

Operating Temperature Condition

Parameter	Symbol	Rating	Unit
Operating Temperature Range	T_{OPER}	-40 to +105	°C
Soldering Temperature (10s)	T_{SOLDER}	260	°C
Storage Temperature	T_{STG}	-55 to +125	°C

Note:

1. Operating Temperature T_{OPER} is the case surface temperature on the center/top side of the NAND.
2. Operating Temperature Range specifies the temperatures where all NAND specifications will be supported. During operation, the NAND case temperature must be maintained between the range specified in the table under all operating conditions.

Recommended Operating Conditions

(Voltage reference to GND)

Parameter	Symbol	Min	Typ.	Max	Unit
Supply Voltage	V_{CC}	1.7	1.8	1.95	V
High Level Input Voltage	V_{IH}	0.8 V_{CC}	-	V_{CC} + 0.3	V
Low Level Input Voltage	V_{IL}	-0.3	-	0.2 V_{CC}	V
Ground Voltage	V_{SS}	0	0	0	V

Valid Blocks

Symbol	Min	Typ.	Max	Unit
NVB	2,008	-	2,048	Block

Note:

1. The device may include initial invalid blocks when first shipped. The number of valid blocks is presented as first shipped. Invalid blocks are defined as blocks that contain one or more bad bits which cause status failure during program and erase operation. Do not erase or program factory-marked bad blocks. Refer to the attached technical notes for appropriate management of initial invalid blocks.
2. The 1st block, which is placed on 00h block address, is guaranteed to be a valid block at the time of shipment.

DC Operation Characteristics

(Recommended operating conditions otherwise noted)

Parameter		Symbol	Test Conditions	Min	Typ.	Max	Unit
Operating Current	Page Read with Serial Access	I_{CC1}	CE# = V_{IL}, Iout = 0, t_{RC} = t_{RC}(min)	-	15	30	mA
	Program	I_{CC2}	-	-	15		
	Erase	I_{CC3}	-	-	15		
Stand-by Current (TTL)		I_{SB1}	CE# = V_{IH}, WP# = 0V/V_{CC}	-	-	1	
Stand-by Current (CMOS)		I_{SB2}	CE# = V_{CC}-0.2, WP# = 0V/V_{CC}	-	10	50	uA
Input Leakage Current		I_{LI}	V_{IN} = 0 to V_{CC} (max)	-	-	+/-10	
Output Leakage Current		I_{LO}	V_{OUT} = 0 to V_{CC} (max)	-	-	+/-10	
Output High Voltage Level		V_{OH}	I_{OH} = -0.1mA	V_{CC} - 0.2	-	-	V
Output Low Voltage Level		V_{OL}	I_{OL} = 0.1mA	-	-	0.2	
Output Low Current (R/B#)		I_{OL} (R/B#)	V_{OL} = 0.2V	-	4	-	mA

Note :

1. Typical value are measured at V_{CC} = 1.8V, T_A = 25°C. Not 100% tested.
2. I_{CC1} and I_{CC2} are without data cache.
3. I_{CC1}, I_{CC2}, I_{CC3}, and I_{SB2} are the values of one chip.

Capacitance

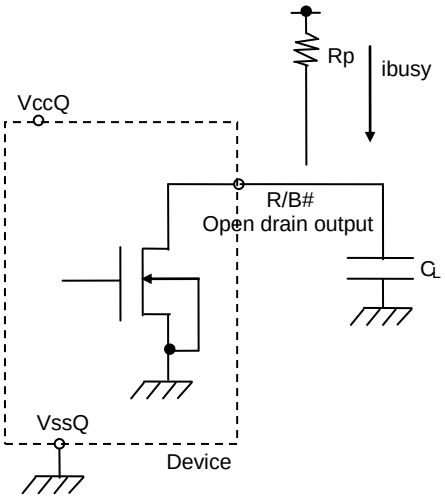
(T_A= 25°C, V_{CC}= 1.8V, f= 1.0MHz)

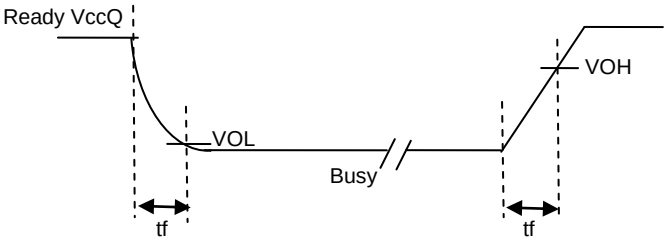
Item	Symbol	Test Condition	Min	Max	Unit
Input/ Output Capacitance	C_{DQ}	V_{OUT}= 0V	-	10	pF
Input Capacitance	C_{IN}	V_{IN}= 0V	-	10	pF

Note: Capacitance is periodically sampled and not 100% tested.

Ready/Busy

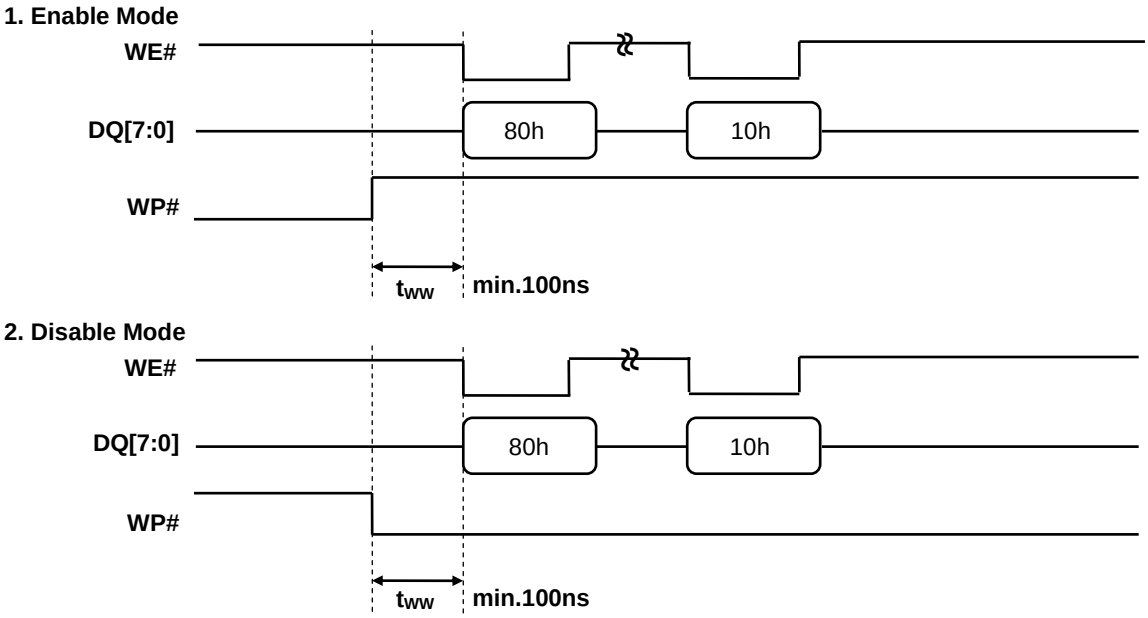
R/B# represents the status of the selected target. R/B# goes busy when only a single LUN is busy while rest of LUNs on the same target are idle.



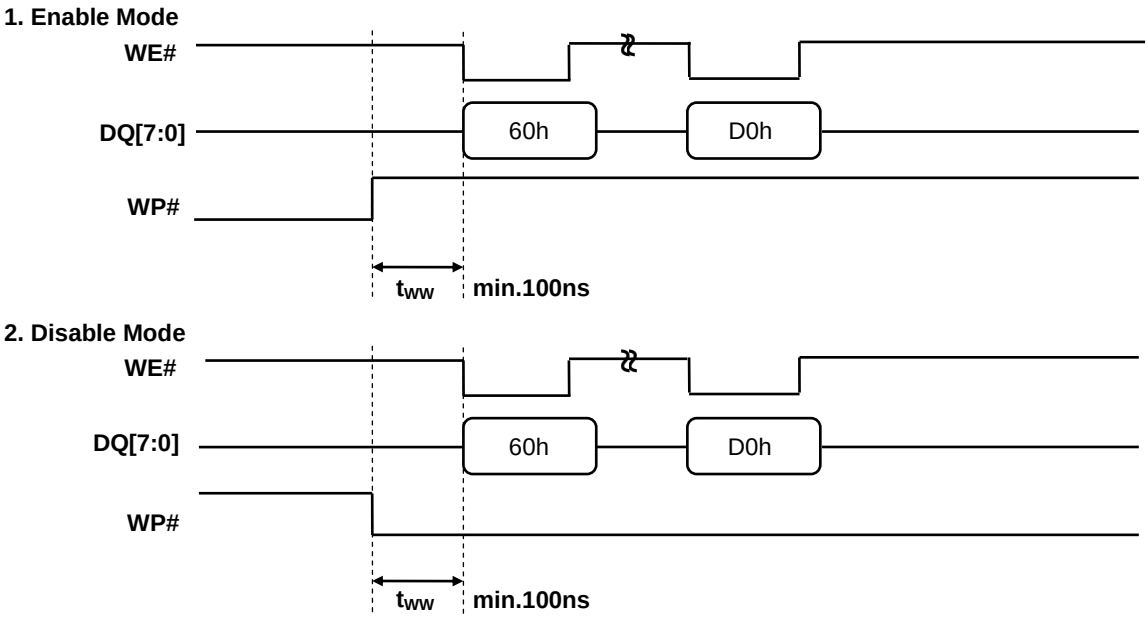


Write Protect

When WP# is enabled, Flash array is blocked from any program and erase operations. This signal shall only transitioned when a target is idle. The host shall be allowed to issue a new command after t_{WW} once WP# is enabled. Figures below describes the t_{WW} timing requirement, shown with the start of a Program command and the start of a Erase command.



Write Protect timing requirements of the Program operation



Write Protect timing requirements of the Erase operation

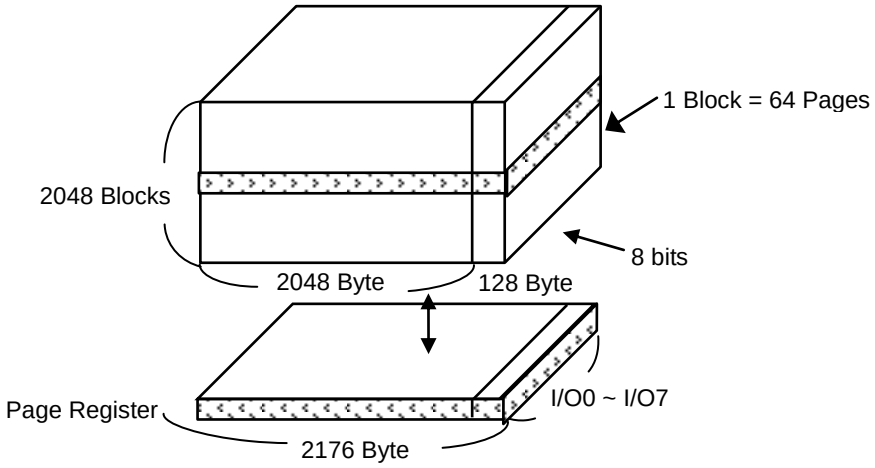
Memory Organization

Addressing

There are two address types used: the column address and the row address. The column address is used to access bytes within a page, i.e. the column address is the byte offset into the page. The row address is used to address pages, blocks, and LUNs (in x8 device, there is only one LUN).

When both the column and row addresses are required to be issued, the column address is always issued first in one or more 8-bit address cycles. The row addresses follow in one or more 8-bit address cycles. There are some functions that may require only row addresses, such as Block Erase. In this case the column addresses shall not be issued.

For both column and row addresses, the first address cycle always contains the least significant bits and the last cycle always contains the most significant bits. If there are bits in the most significant cycles of the column and row addresses that are not used, then they are required to be cleared to zero.



Array Address

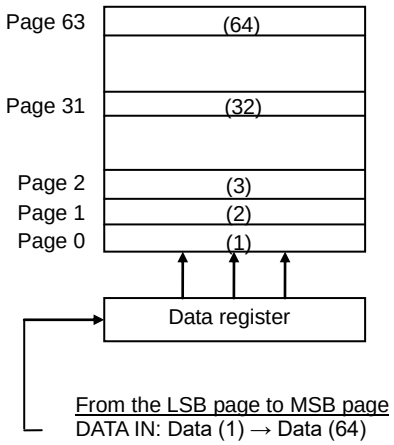
	I/O0	I/O1	I/O2	I/O3	I/O4	I/O5	I/O6	I/O7	Address
1st cycle	A_0	A_1	A_2	A_3	A_4	A_5	A_5	A_7	Column Address
2nd cycle	A_8	A_9	A_{10}	A_{11}	*L	*L	*L	*L	Column Address
3rd cycle	A_{12}	A_{13}	A_{14}	A_{15}	A_{16}	A_{17}	A_{18}	A_{19}	Row Address
4th cycle	A_{20}	A_{21}	A_{22}	A_{23}	A_{24}	A_{25}	A_{26}	A_{27}	Row Address
5th cycle	A_{28}	*L	*L	*L	*L	*L	*L	*L	Row Address

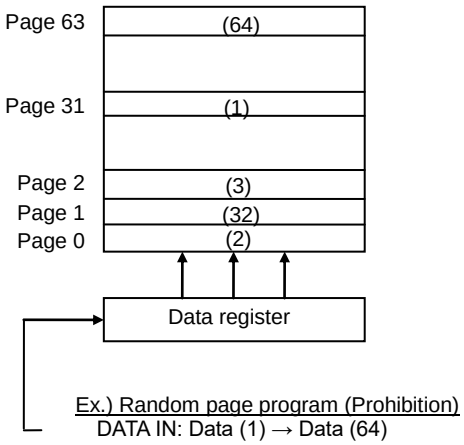
Note:

1. Column address: Starting Address of the Register.
2. *L must be set to 'Low'
3. The device ignores any additional input of address cycles than required.
4. A_{18} is for Plane Address setting, A_{12}~A_{17} are for Page Address, A_{19}~A_{28} are for Block Address.

Addressing for Program Operation

Within a block, the pages must be programmed consecutively from the LSB (least significant bit) page of the block to MSB (most significant bit) pages of the block. Random page address programming is prohibited. In this case, the definition of LSB page is the LSB among the pages to be programmed. Therefore, LSB page doesn't need to be page 0.





Factory Defect Mapping and Error Management

Mask Out Initial Invalid Block(s)

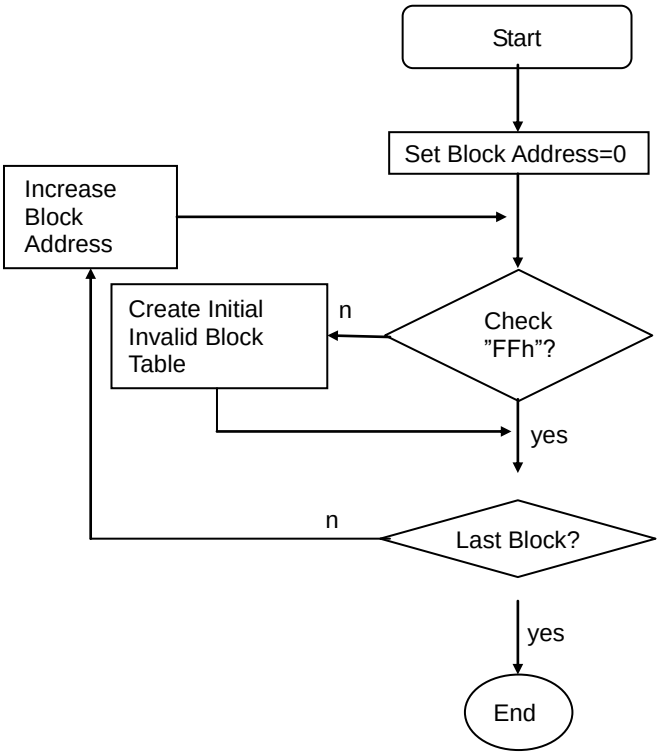
Initial invalid blocks are defined as blocks that contain one or more initial invalid bits whose reliability is not guaranteed. The information regarding the initial invalid block(s) is called the initial invalid block information. Devices with initial invalid block(s) have the same quality level as 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.

Identifying Initial Invalid Block(s) and Block Replacement Management

If a block is defective, the manufacturer shall mark as defective by setting the Defective Block Marking, as shown in figure, of the first or second page of the defective block to a value of non-FFh. The Defective Block Marking is located on the first byte of spare data area in the pages within a block.

The host shall not erase or program blocks marked as defective by the manufacturer, and any attempt to do so yields indeterminate results. Figure below outlines the flow chart how to create an initial invalid block table. It should be performed by the host to create the initial invalid block table prior to performing any erase or programming operations on the target. All pages in non-defective blocks are read FFh with ECC enabled on the controller. A defective block is indicated by the majority of bits being read non-FFh in the Defective Block Marking location of either the first page or second page of the block. The host shall check the Defective Block Marking location of both the first and second page of each block to verify the block is valid prior to any erase or program operations on that block.

Over the lifetime use of a NAND device, the Defective Block Marking of defective blocks may encounter read disturbs that cause bit changes. The initial defect marks by the manufacturer may change value over the lifetime of the device, and are expected to be read by the host and used to create a bad block table during initial use of the part.



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

Algorithm for Bad Block Scanning

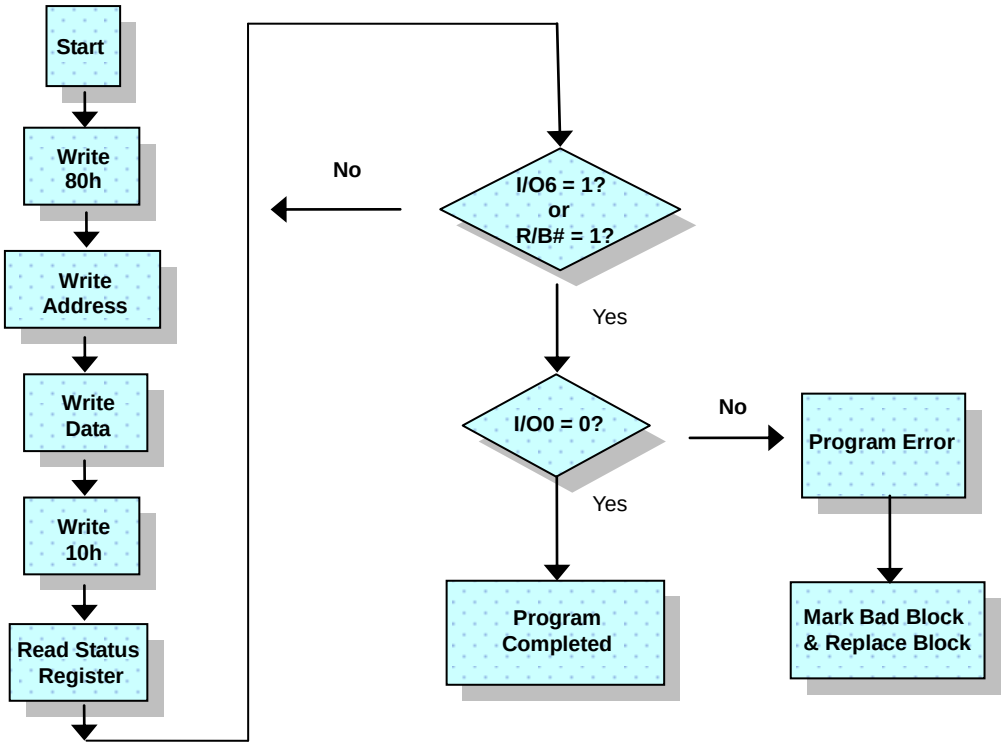
```
For (i=0; i<Num_of_LUs; i++)
{
   For (j=0; j<Blocks_Per_LU; j++)
   {
        Defect_Block_Found=False;

        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);
   }
}
```

Errors in Write or Read Operation

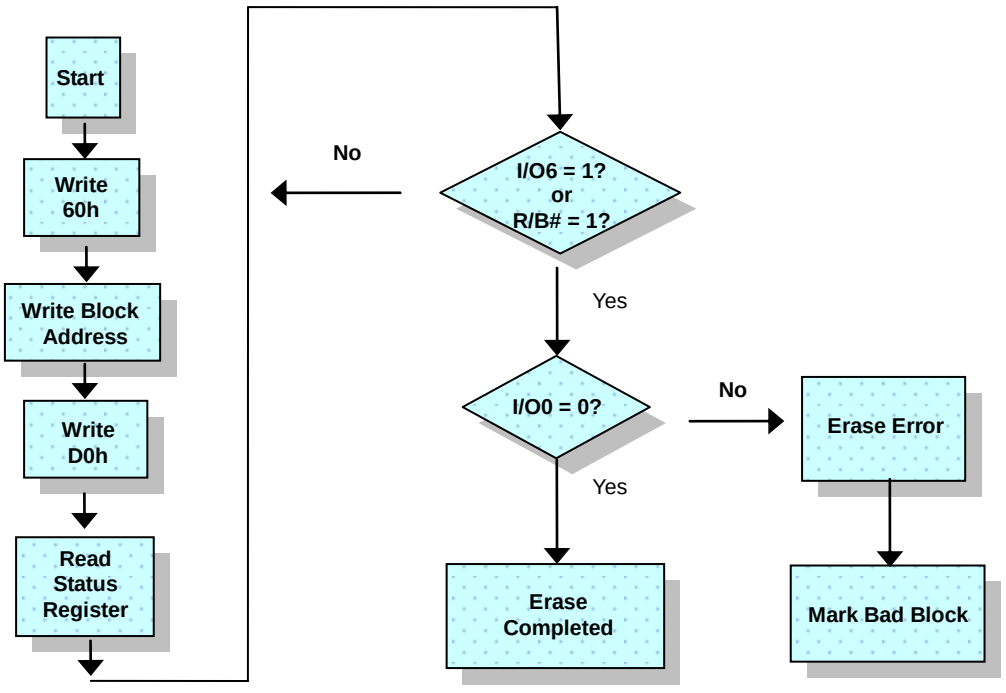
Within its lifetime, additional invalid blocks may develop with NAND Flash memory. Refer to the qualification report for the actual data. The following possible failure modes should be considered to implement a highly reliable system. In the case of status read failure after erase or program, block replacement should be done. Because program status fail during a page program does not affect the data of the other pages in the same block, block replacement can be executed with a page-sized buffer by finding an erased empty block and reprogramming the current target data and copying the rest of the replaced block. In case of Read, ECC must be employed. To improve the efficiency of memory space, it is recommended that the read or verification failure due to bits error (less than 8 bits / 512 byte) be reclaimed by ECC without any block replacement. The additional block failure rate does not include those reclaimed blocks.

Failure Mode		Detection and Countermeasure Sequence
Write	Erase failure	Read Status after Erase → Block Replacement
	Program failure	Read Status after Program → Block Replacement
Read	Up to 8 bits failure	Verify ECC → ECC Correction

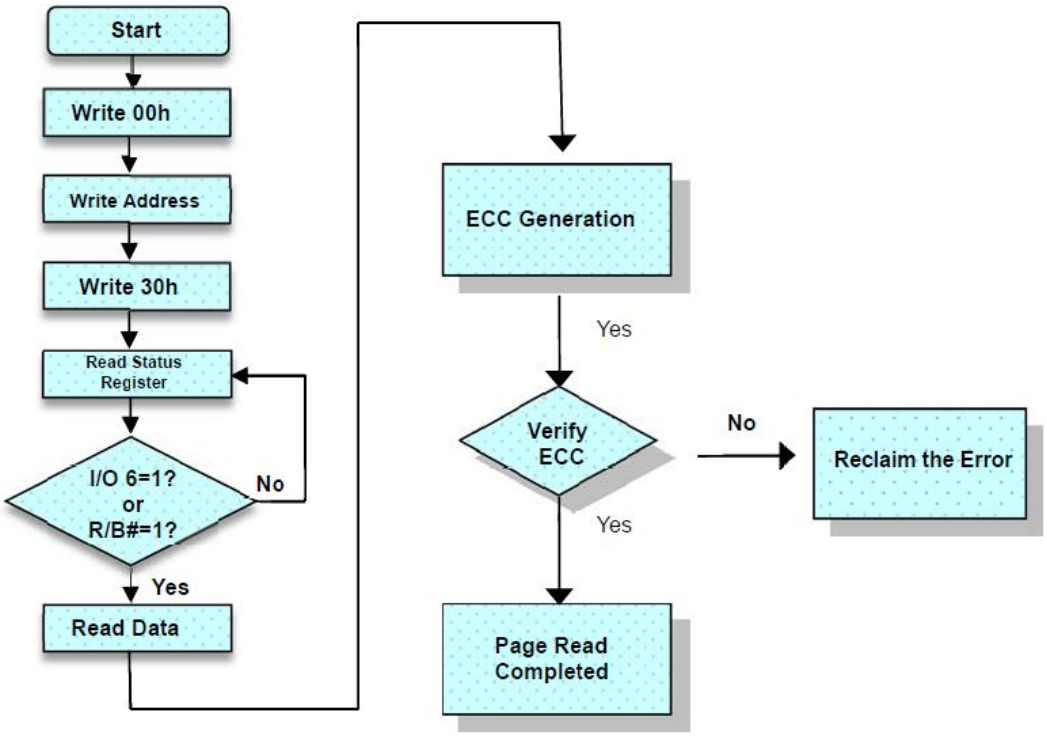
Note: Error Correcting Code → RS Code or BCH Code etc.
Example: 8bit correction / 512Byte



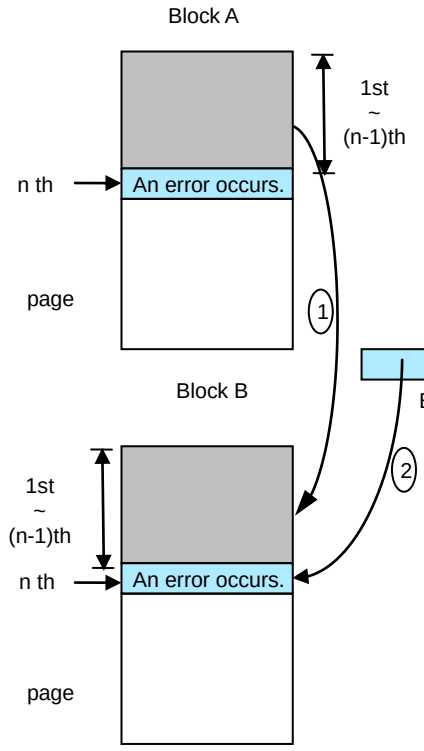
Program Flow Chart



Erase Flow Chart



Read Flow Chart



* Step 1
When an error happens in the nth page of the Block 'A' during erase or program operation.
* Step 2
Copy the data in the 1st ~ (n-1)th page to the same location of another free block. (Block 'B')
* Step 3
Then, copy the nth page data of the Block 'A' in the buffer memory to the nth page of the Block 'B'
* Step 4
Do not erase or program to Block 'A' by creating an 'invalid block' table or other appropriate scheme.

Block Replacement

Function Description

Discovery and Initialization

The device is designed to offer protection from any involuntary program/erase during power transitions. An internal voltage detector disables all functions whenever V_{CC} is below about 1.5V. Max busy time is 5ms after Power-On Reset. During busy time of resetting, the acceptable command is the Read Status (70h).

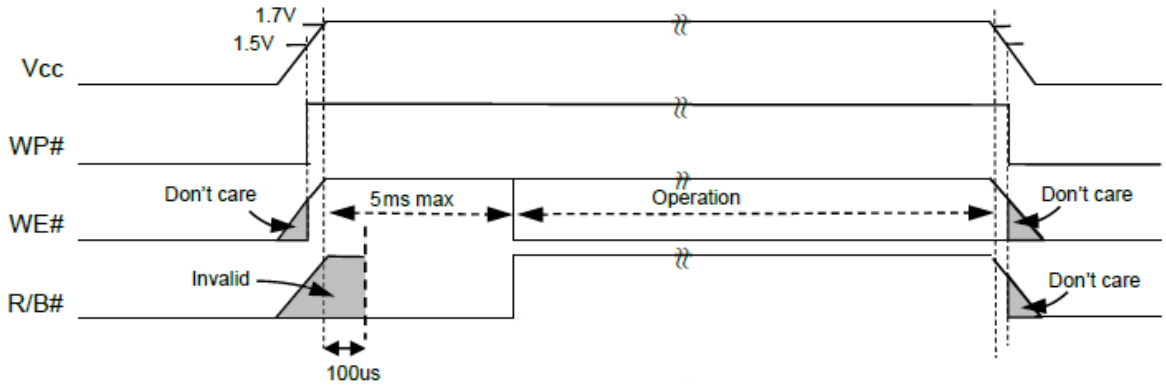
WP# provides hardware protection and is recommended to be kept at V_{IL} during power up and power down. The two step command sequence for program/erase provides additional protection. Figure below defines the Initialization behavior and timings.

Data Protection and Power On Sequence

The timing sequence shown in the figure below is necessary for the power-on/off sequence.

The device internal initialization starts after the power supply reaches an appropriate level in the power on sequence. During the initialization the device R/B# signal indicates the Busy state as shown in the figure below. In this time period, the acceptable commands are 70h.

The WP# signal is useful for protecting against data corruption at power on/off.



AC Waveforms for Power Transition

Mode Selection

SDR

CLE	ALE	CE#	WE#	RE#	WP#	Mode	
H	L	L	↑	H	X	Read Mode	Command Input
L	H	L	↑	H	X		Address Input (5 clock)
H	L	L	↑	H	H	Write Mode	Command Input
L	H	L	↑	H	H		Address Input (5 clock)
L	L	L	↑	H	H	Data Input	
L	L	L	H	↓	X	Data Output	
X	X	X	X	H	X	During Read (Busy)	
X	X	X	X	X	H	During Program (Busy)	
X	X	X	X	X	H	During Erase (Busy)	
X	X(1)	X	X	X	L	Write Protect	
X	X	H	X	X	0V/V_{CC}(2)	Stand-by	

Note :

1. X can be V_{IL} or V_{IH}.
2. WP# should be biased to CMOS high or CMOS low for standby.

AC Test Condition

(V_{CC}= 1.7V ~ 1.95V)

Parameter	Single-ended signaling
Input Pulse	0 to V_{CC}
Input Rise and Fall Times	3ns
Input and Output Timing Levels	V_{CC}/2
Output Load*	C_L(30pF) and 1TTL

Note: Refer to Ready/Busy, R/B# output's Busy to Ready time is decided by the pull-up resistor (Rp) tied to the R/B# pin.

Read / Program / Erase Characteristics

(V_{CC}= 1.7V ~ 1.95V)

Parameter	Symbol	Min	Typ	Max	Unit
Data Transfer from Cell to Register	t_R	-	-	25	us
Program Time	t_{PROG}	-	400	700	us
Last Page Program Time	t_{LPROG}		800	1400	us
Dummy Busy Time for Cache Operation	t_{CBSY}	-	3	750	us
Number of Partial Program Cycles in the Same Page	NOP	-	-	4	cycle
Block Erase Time	t_{BERS}	-	3.5	10	ms
Dummy Busy Time for Two-Plane Page Program (following 11h)	t_{DBSY}	-	0.5	1	us
Data Cache Busy Time in Write Cache (following11h)	$t_{DCBSYW1}$	-	-	10	us
Data Cache Busy Time in Write Cache (following 15h)	$t_{DCBSYW2}$	-	-	700	us

Note:

1. Typical program time is defined as the time within which more than 50% of the whole pages are programmed at 1.8V Vcc and 25°C temperature.
2. t_{CBSY} max. time depends on timing between internal program completion and data-in.
3. $t_{DCBSYW2}$ depends on the timing between internal programming time and data in time.
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 Timing Characteristics

SDR (V_{CC} = 1.7~1.95V)

Parameter	Symbol	Min	Max	Unit
CLE Setup Time	t_{CLS}(1)	12	-	ns
CLE Hold Time	t_{CLH}	5	-	ns
CE# Setup Time	t_{CS}(1)	20	-	ns
CE# Hold Time	t_{CH}	5	-	ns
WE# Pulse Width	t_{WP}	12	-	ns
ALE Setup Time	t_{ALS}(1)	12	-	ns
ALE Hold Time	t_{ALH}	5	-	ns
Data Setup Time	t_{DS}(1)	12	-	ns
Data Hold Time	t_{DH}	5	-	ns
Write Cycle Time	t_{WC}	45	-	ns
WE# High Hold Time	t_{WH}	10	-	ns
Address to Data Loading Time	t_{ADL}(2)	70	-	ns
Data Transfer from Cell to Register	t_R	-	25	us
ALE to RE# Delay	t_{AR}	10	-	ns
CLE to RE# Delay	t_{CLR}	10	-	ns
Ready to RE# Low	t_{RR}	20	-	ns
Ready to WE# Falling Edge	t_{RW}	20		ns
RE# Pulse Width	t_{RP}	12	-	ns
WE# High to Busy	t_{WB}	-	100	ns
WP# Low to WE# Low (disable mode)	t_{WW}	100	-	ns
WP# High to WE# Low (enable mode)				
Read Cycle Time	t_{RC}	45	-	ns
CE# Low to RE# Low	**t_{CR}**	9	-	ns
RE# Access Time	t_{REA}	-	25	ns
CE# Access Time	t_{CEA}	-	25	ns
RE# High to Output Hi-Z	t_{RHZ}	-	100	ns
CE# High to Output Hi-Z	t_{CHZ}	-	30	ns
CLE High to Output Hi-Z	**t_{CLHZ}**	-	30	ns
RE# High to Output Hold	t_{RHOH}	15	-	ns
RE# Low to Output Hold	t_{RLOH}	5		ns
CE# High to Output Hold	t_{COH}	15		ns
RE# High Hold Time	t_{REH}	10	-	ns
Output Hi-Z to RE# Low	t_{IR}	0	-	ns
RE# High to WE# Low	t_{RHW}	100	-	ns
WE# High to RE# Low (Read Status)	t_{WHR1}	60	-	ns

WE# High to RE# Low (Column Address Change in Read)		t_{WHR2}	60	-	ns
Device Resetting Time during…	Ready	t_{RST}	-	5	us
	Read		-	5	us
	Program		-	10	us
	Erase		-	250	us
Data Cache Busy in Read Cache (following 31h and 3Fh)		$t_{DCBSYR1}$		30	us
Data Cache Busy in Page Copy (following 3Ah)		$t_{DCBSYR2}$		30	us

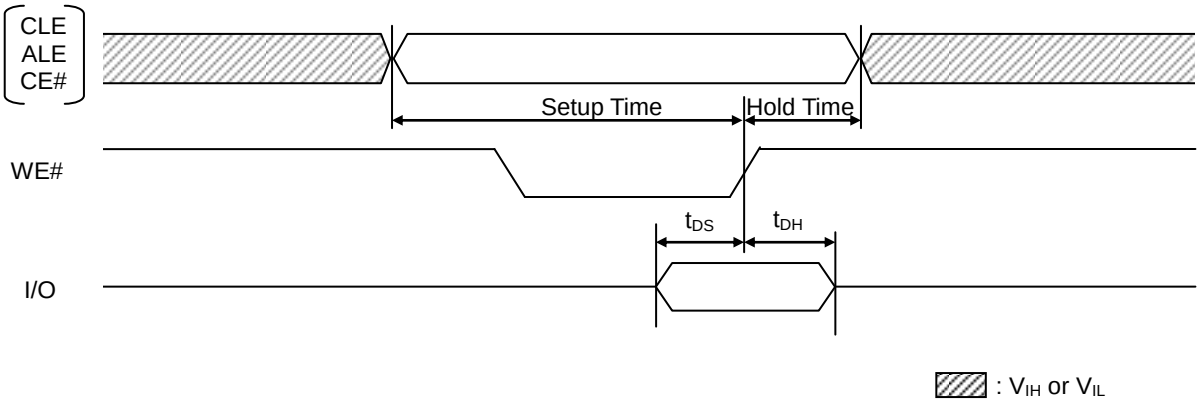
Note:

1. The transition of the corresponding control pins must occur only once while WE# is held low.
2. t_{ADL} is the time from the WE rising edge of final address cycle to the WE# rising edge of first data cycle.

General Timing

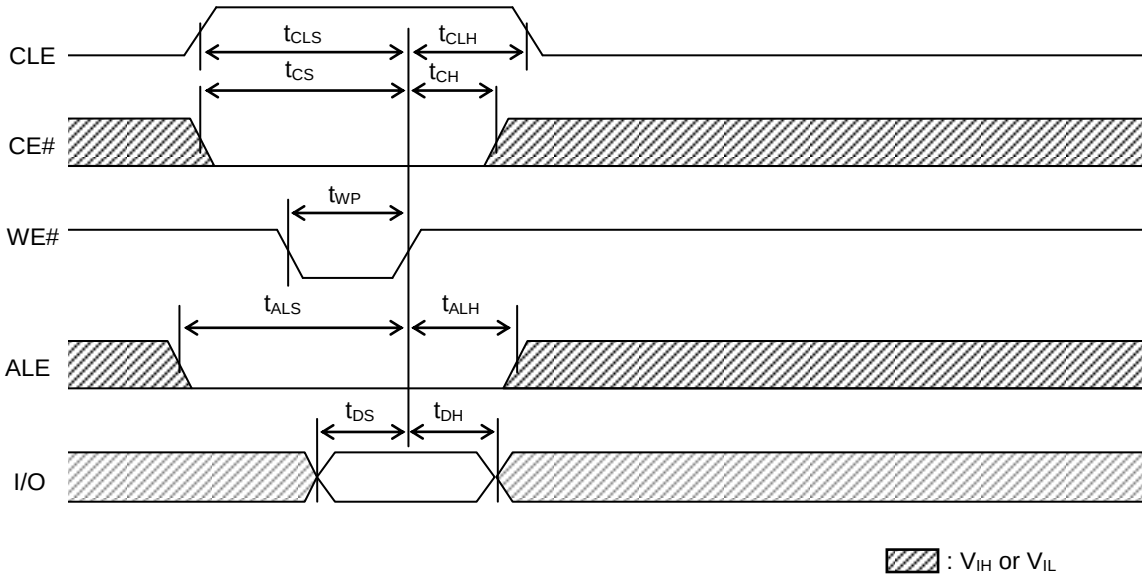
CE bar = CE#
WE bar = WE#
RE bar = RE#
R/ B bar = R/B#

Command/Address/Data Latch Timing



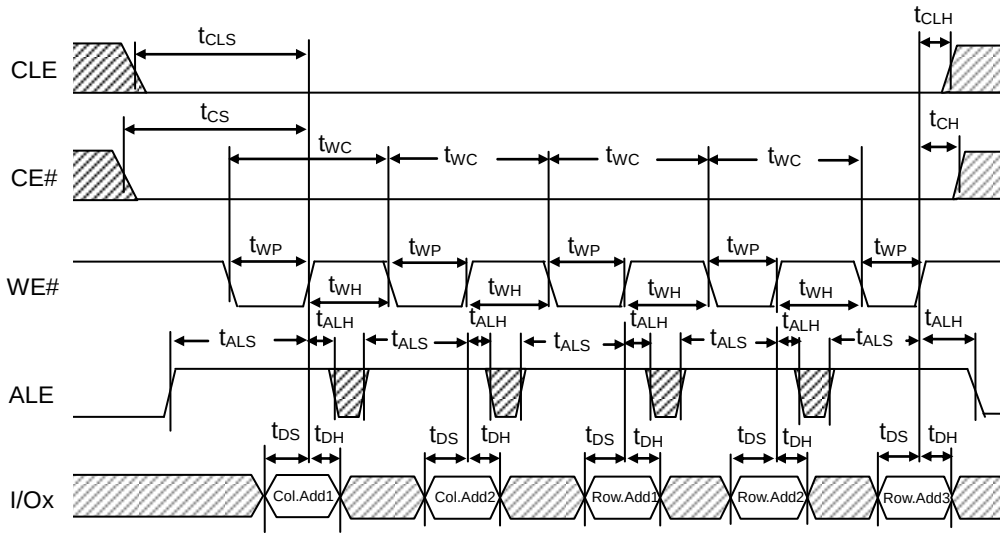
Command/Address/Data Latch Timing

Command Input Cycle



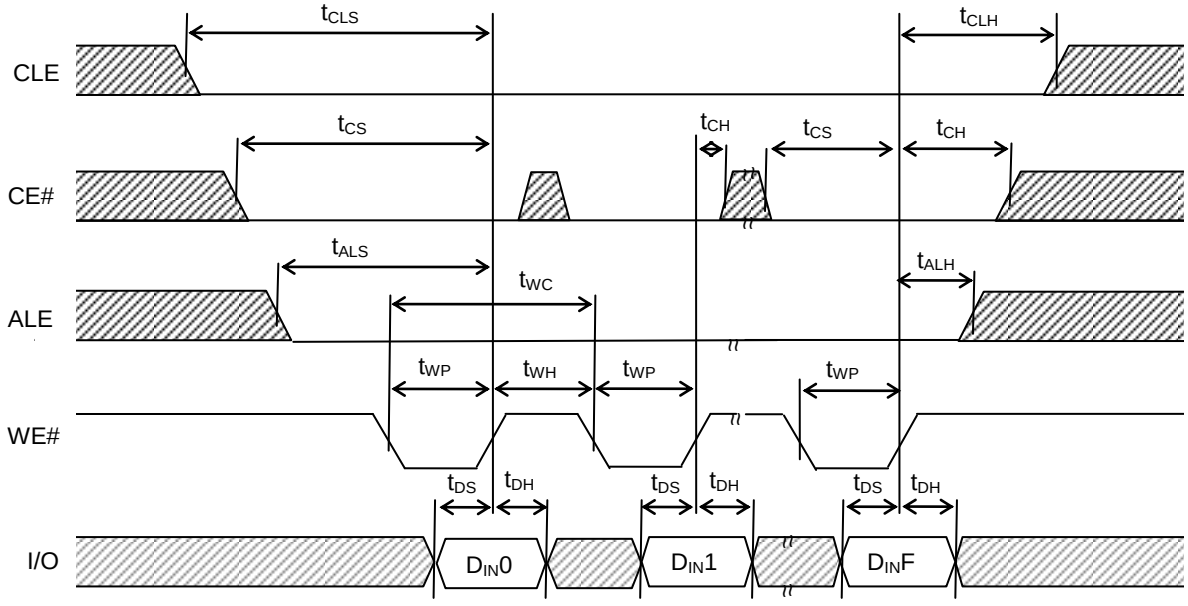
Command Input Cycle Timing

Address Input Cycle



Address Input Cycle Timing

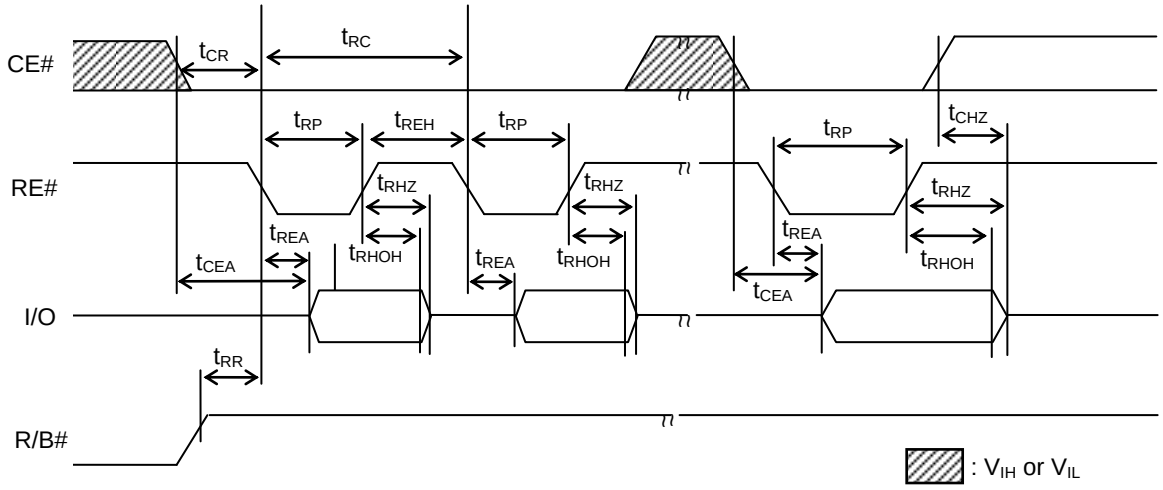
Data Input Cycle



Note: DINF means the Final Data Input.

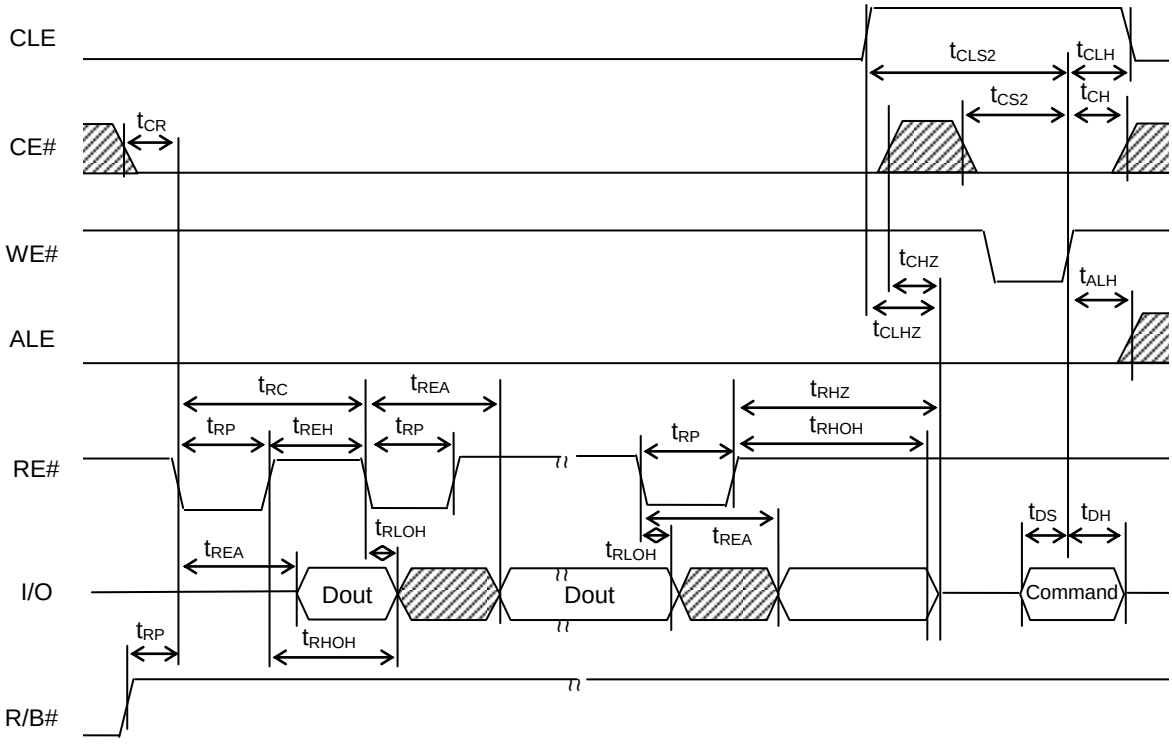
Data Input Cycle Timing

Data Output Cycle



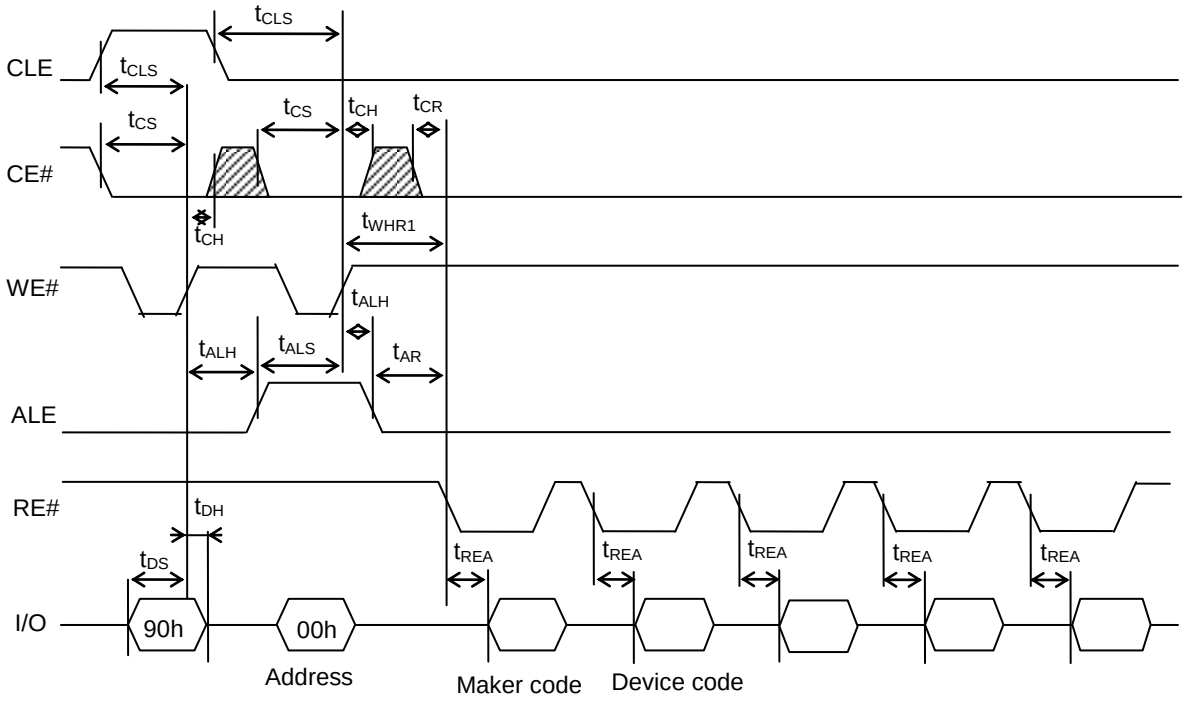
Data Output Cycle Timing

Basic Data Output



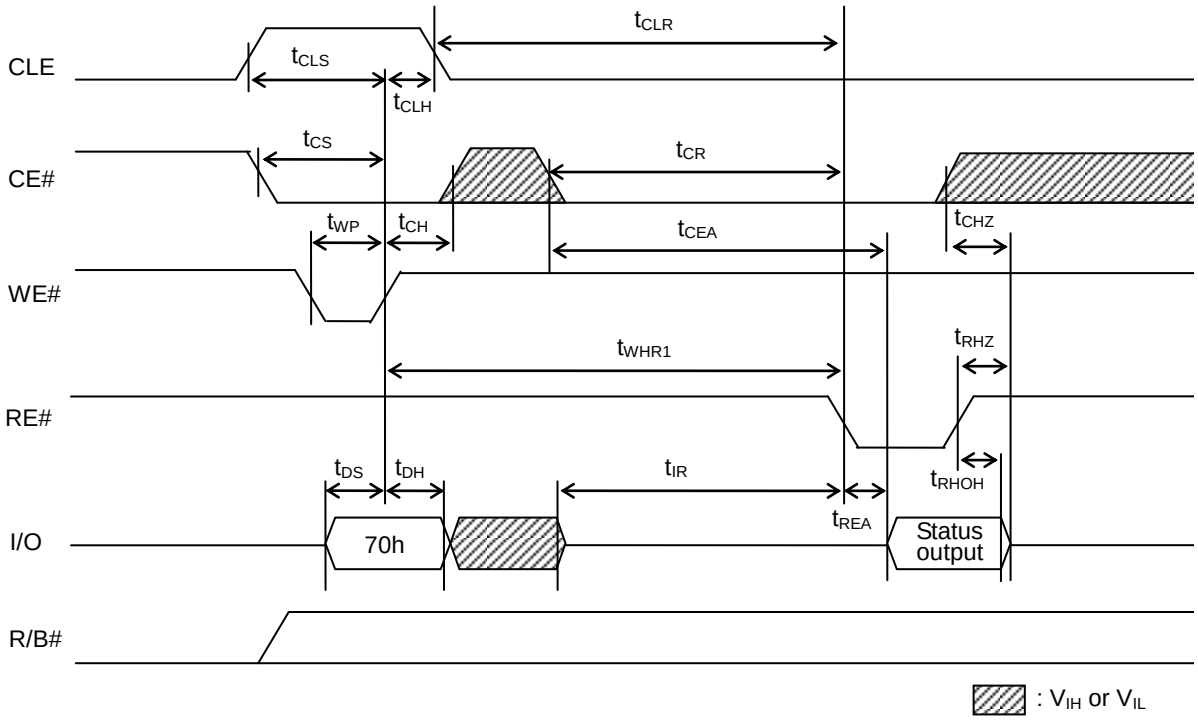
Basic Data Output Timing

Read ID



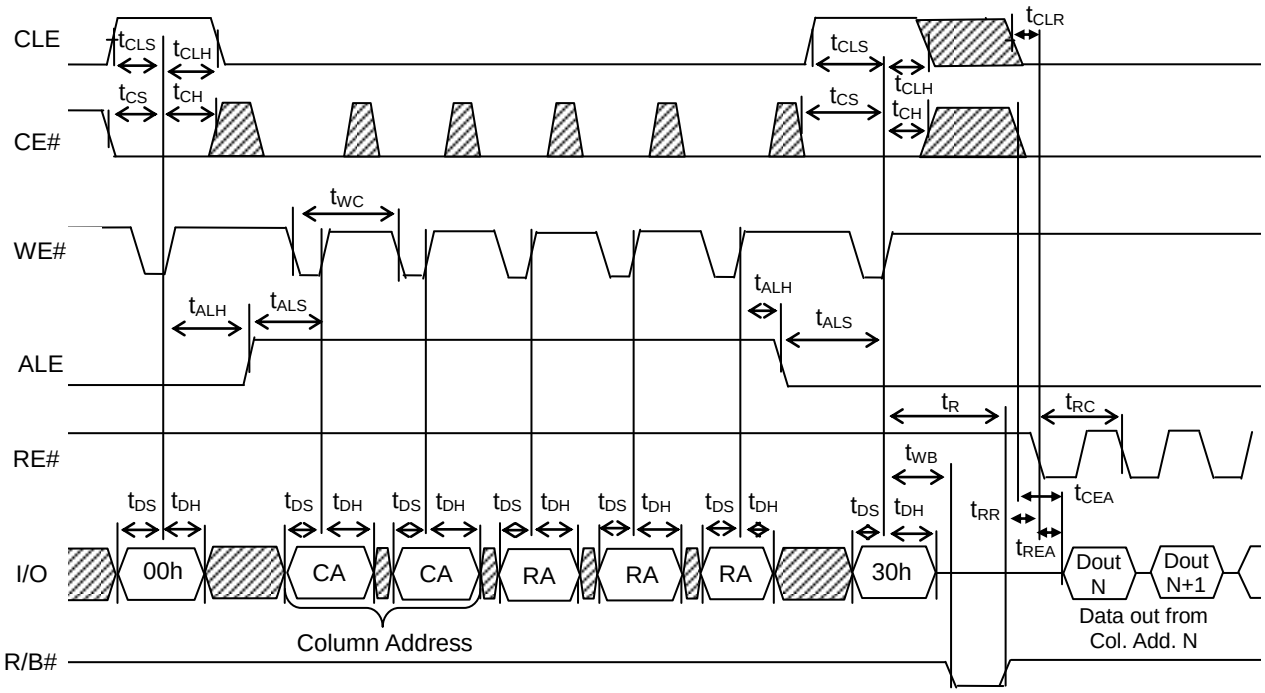
Read ID Operation Timing

Status Read Cycle



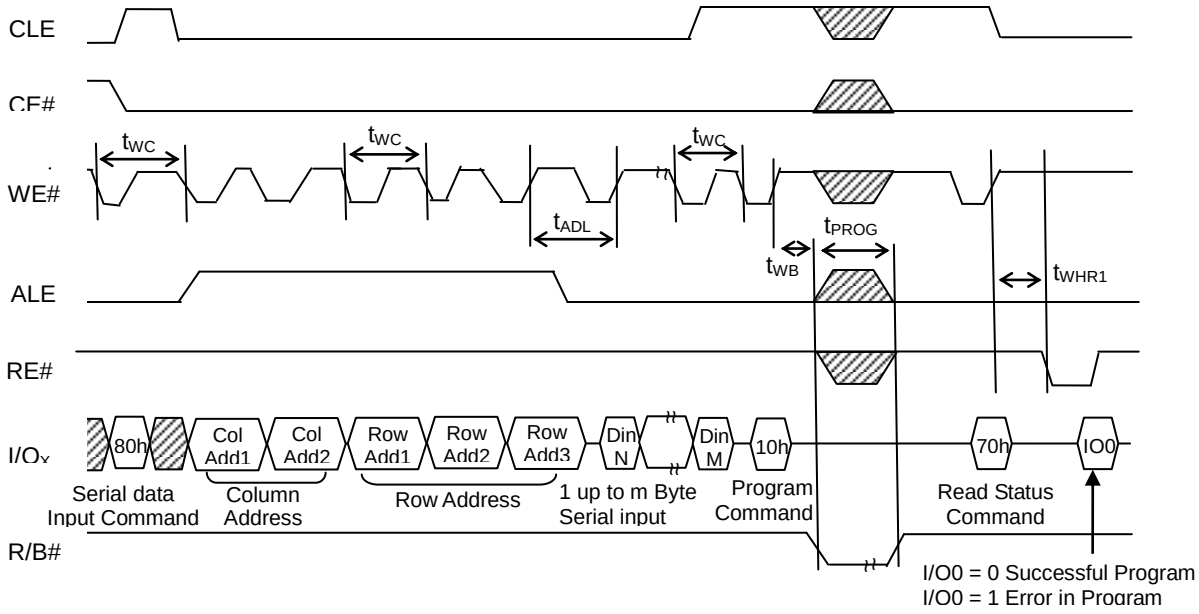
Status Read Cycle Timing

Page Read Operation



Page Read Operation Timing

Page Program Operation



Page Program Operation

Command Description and Device Operation

Command Sets

Function	1st Cycle	2nd Cycle	Acceptable Command during Busy
Read	00h	30h	
Read for Copy-Back	00h	35h	
Read ID	90h	-	
Reset	FFh	-	O
Page Program	80h	10h	
Copy-Back Program	85h	10h	
Block Erase	60h	D0h	
Random Data Input (1)	85h	-	
Random Data Output (1)	05h	E0h	
Read Status	70h	-	O
Read Status2	F1h	-	O
Cache Program	80h	15h	
Cache Read	31h	-	
Read Start for Last Page Cache Read	3Fh	-	
Two-Plane Page Read	60h-60h	30h	
Two-Plane Cache Read	60h-60h	33h	
Two-Plane Read for Copy-Back	60h-60h	35h	
Two-Plane Random Data Output(1)	00h-05h	E0h	
Two-Plane Page Program(2)	80h-11h	81h-10h	
Two-Plane Copy-Back Program(2)	85h-11h	81h-10h	
Two-Plane Block Erase	60h-60h	D0h	
Two-Plane Cache Program(2)	80h-11h	81h-15h	
Read for Page Copy with Data Out	00h	3Ah	
Auto Program with Data Cache during Page Copy	8Ch	15h	
Auto Program for last page during Page Copy	8Ch	10h	
Read Parameter Page	ECh	-	
Read Unique ID	EDh	-	

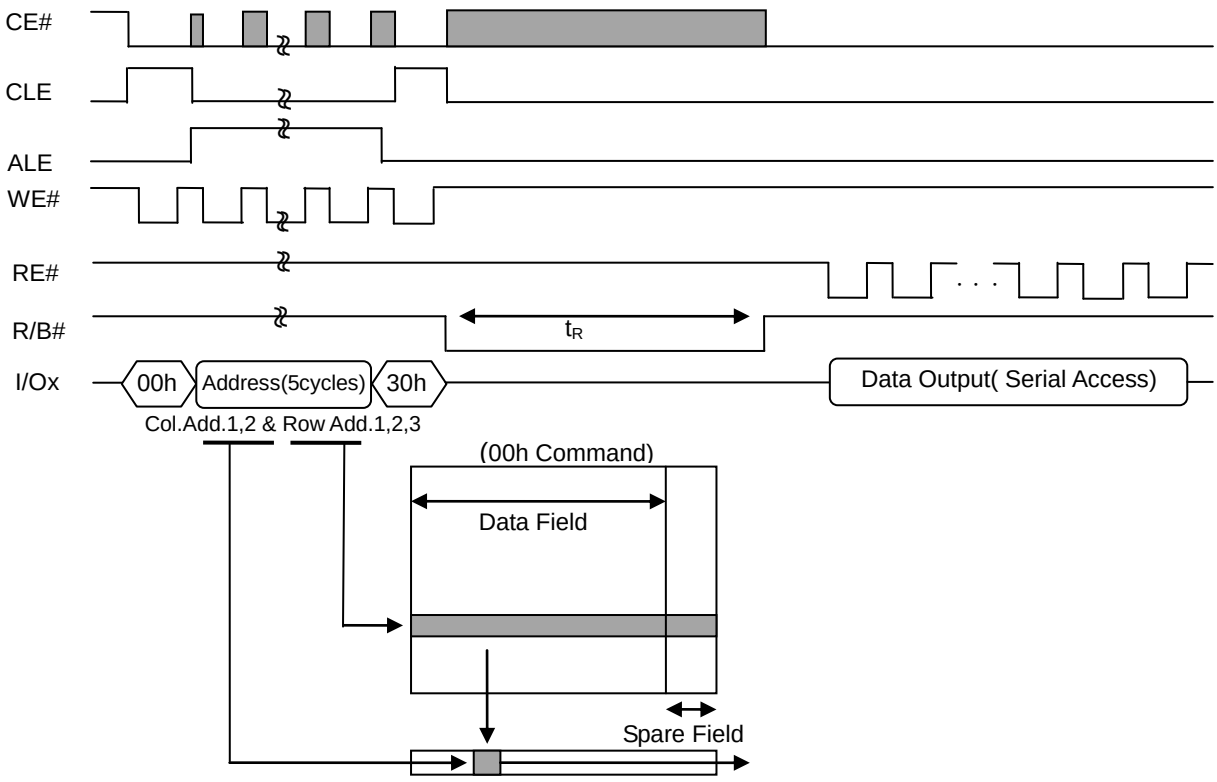
Note:

1. Random Data Input/ Output can be executed in a page.
2. The page address and block address shall be the same in Two-Plane operation.

Operations

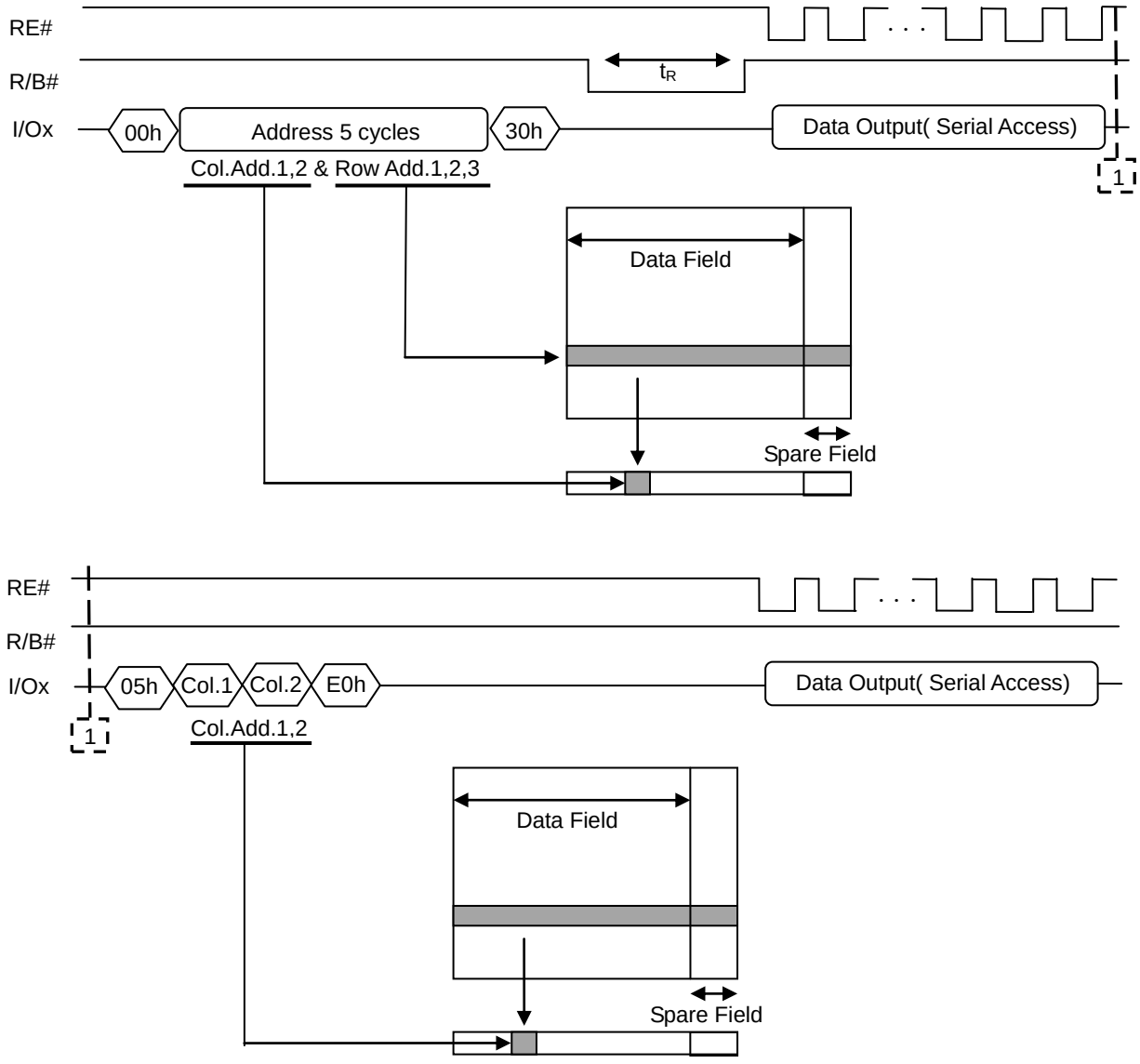
Page Read Operation

The Page Read function reads a page of data identified by row address for the selected LUN. The page of data is made available to be read from the page register starting at the specified column address. Figure below defines the Page Read behavior and timings. Reading beyond the end of a page results in indeterminate values being returned to the host.



Page Read Operation Timing

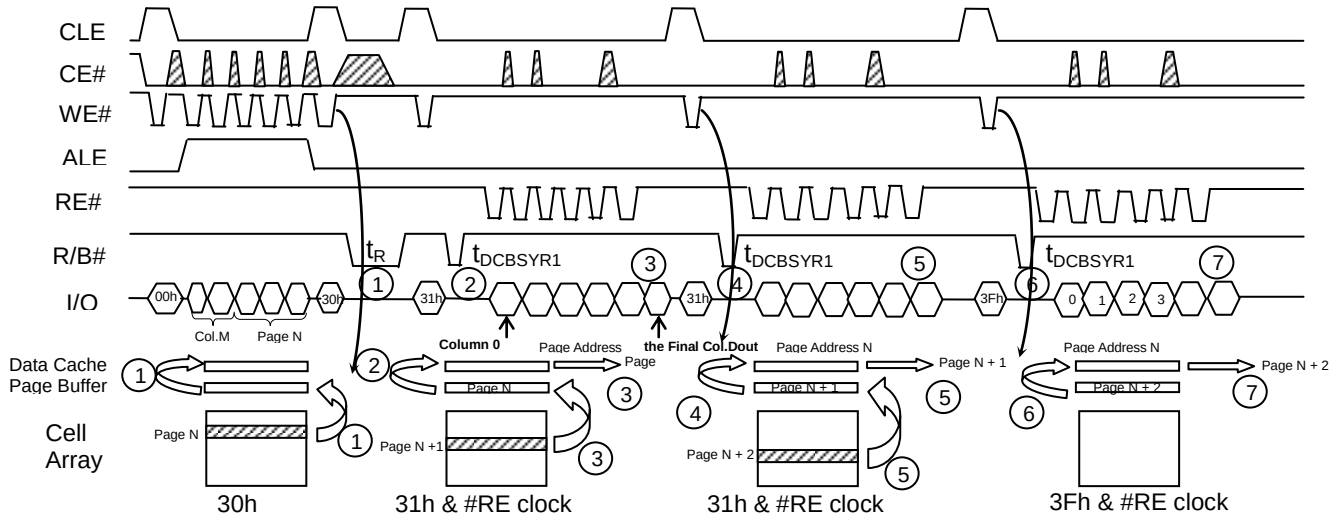
The Random Data Output function changes the column address from which data is being read in the page register for the selected LUN. The Random Data Output command shall only be issued when LUN is in a read idle condition. Figure below defines the Random Data Output behavior and timings. The host shall not read data from the LUN until t_{WHR2}(ns) after the second command (i.e. E0h) is written to the LUN.



Random Data Output in a Page Timing

Cache Read Operation

Cache Read is an extension of Page Read, and is available only within a block. The normal Page Read command (00h-30h) is always issued before invoking Cache Read. After issuing the Cache Read command (31h), read data of the designated page (page N) are transferred from data registers to cache registers in a short time period of $t_{DCBSYR1}$, and then data of the next page (page N+1) is transferred to data registers while the data in the cache registers are being read out. Host controller can retrieve continuous data and achieve fast read performance by iterating Cache Read operation. The Read Start for Last Page Cache Read command (3Fh) is used to complete data transfer from memory cells to data registers.



Cache Read Operation Timing

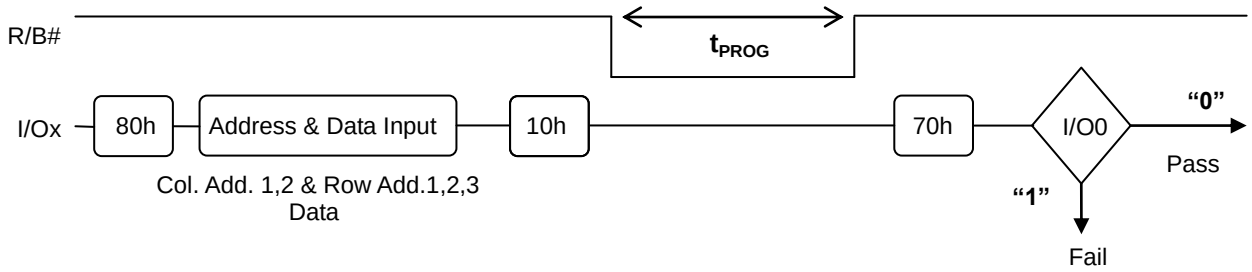
If the 31h command is issued to the device, the data content of the next page is transferred to the Page Buffer during serial data out from the Data Cache, and therefore the t_R (Data transfer from memory cell to data register) will be reduced.

1. Normal read. Data is transferred from Page N to Data cache through Page Buffer. During this time period, the device outputs Busy state for t_R max.
2. After the Ready/Busy returns to Ready, 31h command is issued and data is transferred to Data Cache from Page Buffer again. This data transfer takes $t_{DCBSYR1}$ max and the completion of this time period can be deleted by Ready/Busy signal.
3. Data of Page N + 1 is transferred to Page Buffer from cell while the data of Page N in Data Cache can be read out by RE# clock simultaneously.
4. The 31h command makes data of Page N + 1 transfer to Data Cache from Page Buffer after the completion of the transfer from cell to Page Buffer. The device outputs Busy state for $t_{DCBSYR1}$ max.. This Busy period depends on the combination of the internal data transfer time from cell to Page Buffer and the serial data out time.
5. Data of Page N + 2 is transferred to Page Buffer from cell while the data of Page N + 1 in Data Cache can be read out by RE# clock simultaneously.
6. The 3Fh command makes the data of Page N + 2 transfer to the Data Cache from the Page Buffer after the completion of the transfer from cell to Page Buffer. The device outputs Busy state for $t_{DCBSYR1}$ max.. This Busy period depends on the combination of the internal data transfer time from cell to Page Buffer and the serial data out time.
7. Data of Page N + 2 in Data Cache can be read out, but since the 3Fh command dose not transfer the data from the memory cell to Page Buffer, the device can accept new command input immediately after the completion of serial data out.

Page Program Operation

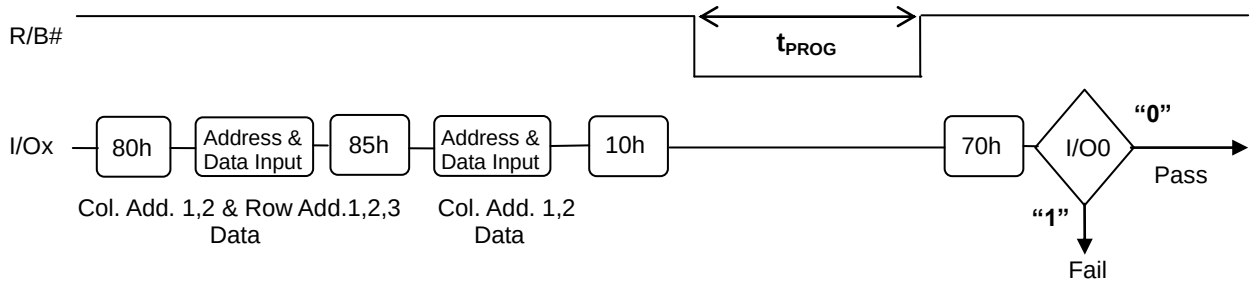
The device is programmed basically on a page basis, and each page shall be programmed only one before being erased. The addressing order shall be sequential within a block. The contents of the page register are programmed into the Flash array specified by row address. After t_{PROG} program time, the R/B# return to ready state. Read Status command (70h) can be issued to read the status register right after 10h.Figure below defines the Page Program behavior and timings.

In a single page, the number of partial program per page must not exceed 4.



Program & Read Status Operation Timing

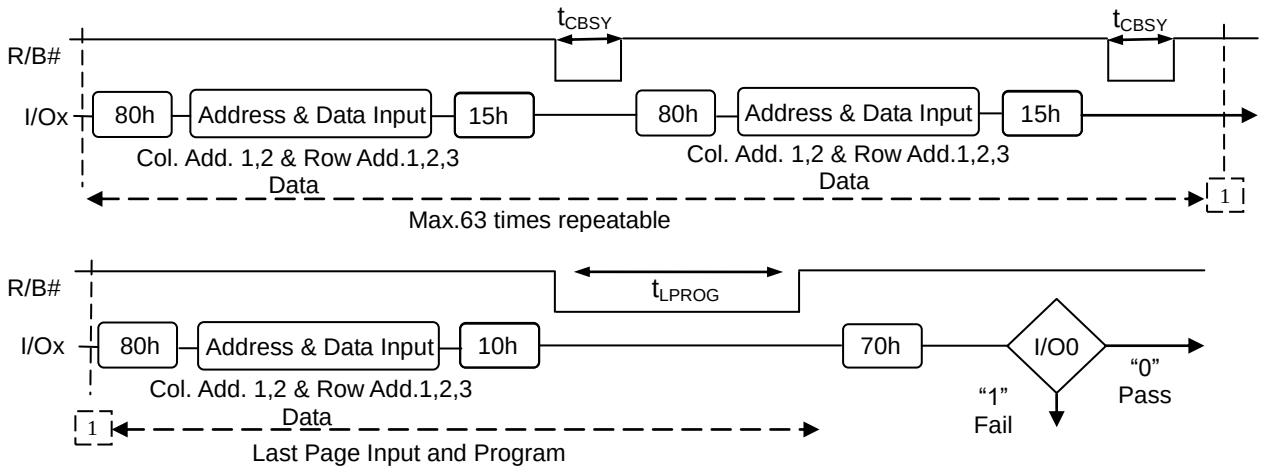
The device supports random data input in a page. The column address for the next data, which will be written, may be changed to the address using Random Data Input command (i.e. 85h). Random data input may be operated multiple times without limitation.



Random Data Input in a Page Timing

Cache Program Operation

The Cache Program function allows the host to write the next data for another page to the page register while a page of data to be programmed to the Flash array for the selected LUN. When command 15h is issued, R/B# returns high (i.e. ready) when a cache register is ready to be written after data in the cache register is transferred to a page register. However, when command 10h is issued for the final page, R/B# turns to high after outstanding program operation performed by previous Cache Program command and the program operation for the final page is completed. SR[0] is valid for this command after SR[5] transitions from zero to one until the next transition. SR[1] is valid for this command after SR[6] transitions from zero to one, and it is invalid after the first Cache Program command completion since there is no previous Cache Program operation. Cache Program operation shall work only within a block. Figure below defines the Cache Program behavior and timings. Writing beyond the end of the page register is undefined.



Cache Program Operation Timing

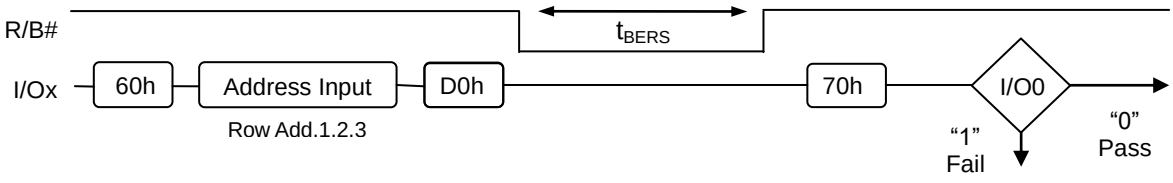
Note:

The last page of the program sequence must use the PROGRAM PAGE (10h) command instead of the CACHE PROGRAM (15h) command.

If the CACHE PROGRAM (15h) command is used every time, including the last page of the programming sequence, monitor I/O6 (Ready / Busy) by issuing Read Status command (70h) and make sure the previous page program operation is completed.

Block Erase Operation

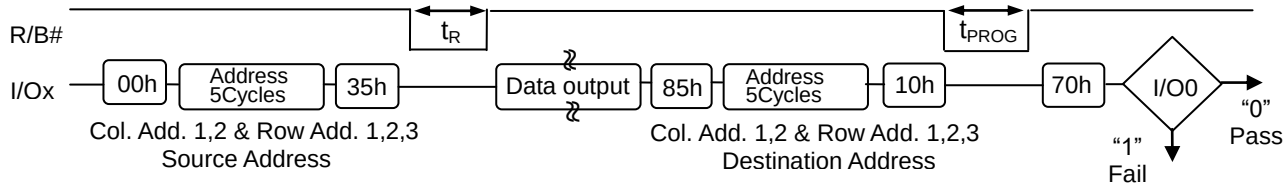
The Block Erase operation is done on a block basis. Only three cycles of row addresses are required for Block Erase operation and a page address within the cycles is ignored while plane and block address are valid. After Block Erase operation passes, all bits in the block shall be set to one. SR[0] is valid for this command after SR[6] transitions from zero to one (i.e. the selected LUN is ready) until the LUN goes in busy state by a next command. Figure below defines the Block Erase behavior and timings.



Block Erase Operation Timing

Copy-Back Program Operation

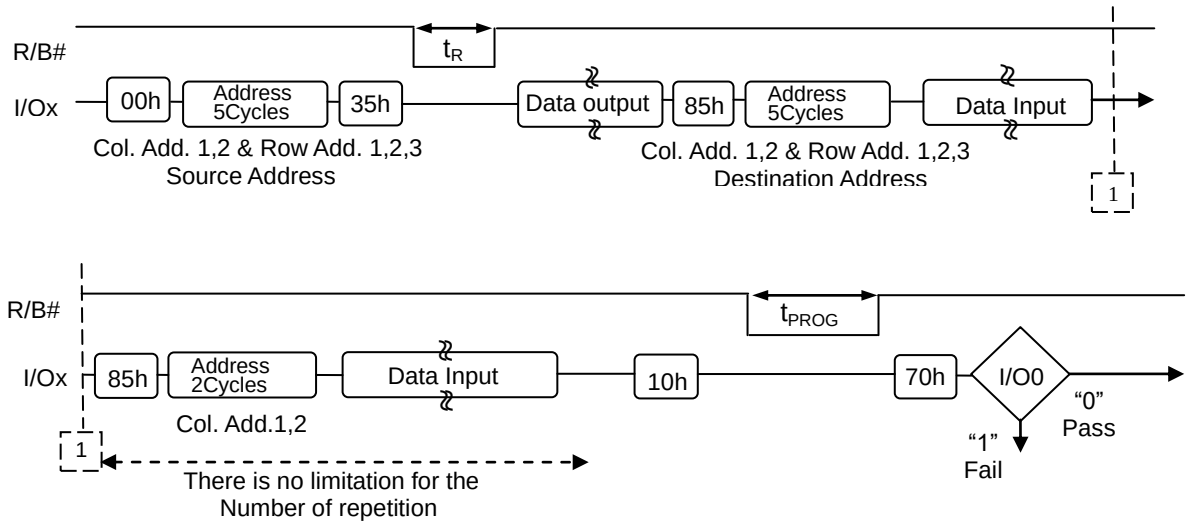
The Copy-Back Program with Read for Copy-Back is configured to efficiently rewrite data stored in a page without data re-loading when no error within the page is found. Since the time consuming re-loading cycles are removed, copy-back operation helps the system performance improve. The benefit is especially obvious when a part of a block is updated and the rest of the block also needs to be copied to the newly assigned free block. The Copy-Back operation consists of Read for Copy-Back and Copy-Back Program. A host reads a page of data from a source page using Read for Copy-Back and copies read data back to a destination page on the same LUN by Copy-Back Program command. Copy-Back Program operation shall work only within the same plane. Figure below defines the Copy-Back Program behavior and timings.



Note: The LSB of page address shall be the same between source and destination pages. In other words, the page of even page address can't be copied to the page of odd page address, and the page of odd page address can't be copied to the page of even page address as well.

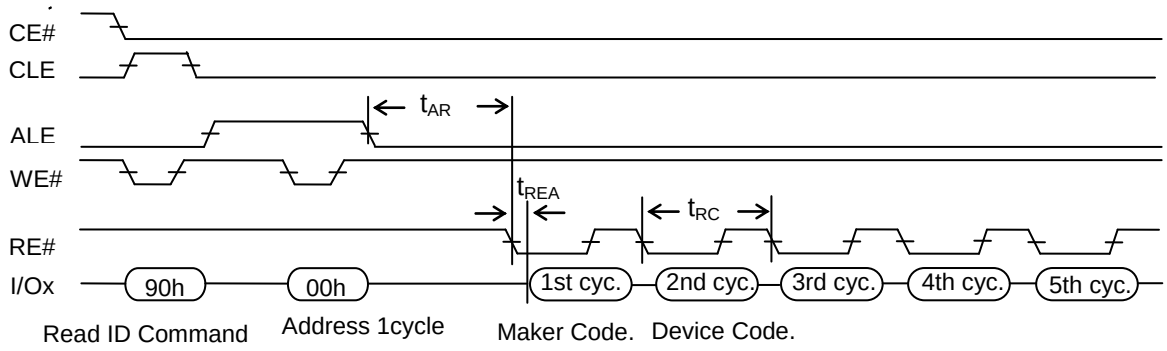
Page Copy-Back Program Operation Timing

After a host completes to read data from a page register, the host may modify data using Random Data Input command if required. Figure below defines Copy-Back Program with Random Data Input behavior and timings.



Page Copy-Back Program Operation with Random Data Input

Read ID



Read ID Timing

Read ID (00h Address ID Cycle)

Users can read five bytes of ID containing manufacturer code, device code and architecture information of the target by command 90h followed by 00h address. The command register remains in Read ID mode until another command is issued.

	Description	X8 device
1st Byte	Maker Code	C8h
2nd Byte	Device Code	5Ah
3rd Byte	Internal Chip Number, Cell Type, etc	90h
4th Byte	Page Size, Block Size, etc	04h
5th Byte	Plane Number, ECC Level	34h

3rd ID Data

Item	Description	DQ7	DQ6	DQ5	DQ4	DQ3	DQ2	DQ1	DQ0
Internal Chip Number	1 2 4 8							0 0 1 1	0 1 0 1
Cell Type	2 Level Cell 4 Level Cell 8 Level Cell 16 Level Cell					0 0 1 1	0 1 0 1		
Number of Simultaneously Programmed Pages	1 2 4 8			0 0 1 1	0 1 0 1				
Interleave Program Between Multiple Chips	Not Support Support		0 1						
Cache Program	Not Support Support	0 1							

4th ID Data

Item	Description	DQ7	DQ6	DQ5	DQ4	DQ3	DQ2	DQ1	DQ0
Page Size (w/o redundant area)	2KB							0	0
	4KB							0	1
	8KB							1	0
	Reserved							1	1
Block Size (w/o redundant area)	128KB	0		0	0				
	256KB	0		0	1				
	512KB	0		1	0				
	1MB	0		1	1				
	Reserved	1		0	0				
	Reserved	1		0	1				
	Reserved	1		1	0				
	Reserved	1		1	1				
Redundant Area Size (Byte / Page Size)	Reserved		0			0	0		
	128B		0			0	1		
	224B		0			1	0		
	400B		0			1	1		
	436B		1			0	0		
	512B		1			0	1		
	640B		1			1	0		
	1KB		1			1	1		

5th ID Data

Item	Description	DQ7	DQ6	DQ5	DQ4	DQ3	DQ2	DQ1	DQ0
Plane Number	1					0	0	0	
	2					0	1	0	
	4					1	0	0	
	8					1	1	0	
	16					1	1	1	
ECC Level	1bit		0	0	0				
	2bit		0	0	1				
	4bit		0	1	0				
	8bit		0	1	1				
	12bit		1	0	0				
	24bit		1	0	1				
	40bit		1	1	0				
	60bit		1	1	1				
Reserved	Reserved	0							0

Read Status and Read Status2

The Read Status function (command 70h) retrieves a status value for the last operation issued in the case of one-plane operations. While the Read Status2 function (command F1h) retrieves plane0 and plane1 status. Both 70h and F1h are followed without address setting. Specifically, Read Status and Read Status 2 return the combined status values of the independent status register bits according to Table below.

Read Status Definition

	I/O0	I/O1	I/O2	I/O3	I/O4	I/O5	I/O6	I/O7
Definition	Pass: 0 Fail: 1	Pass: 0 Fail: 1	Reserved	Reserved	Reserved	Busy: 0 Ready: 1	Busy: 0 Ready: 1	Protected: 0 Not Protected: 1
Read	NA	NA	NA	NA	NA	NA	Busy/Ready	Write Protect
Cache Read	NA	NA	NA	NA	NA	Flash array Busy/Ready	Host Busy/Ready	Write Protect
Page Program	Pass/Fail	NA	NA	NA	NA	NA	Busy/Ready	Write Protect
Cache Program	Pass/Fail	(N-1) Pass/Fail	NA	NA	NA	Flash array Busy/Ready	Host Busy/Ready	Write Protect
Block Erase	Pass/Fail	NA	NA	NA	NA	NA	Busy/Ready	Write Protect

Note:

1. During Block Erase, Page Program or Copy-Back operation, I/O0 is only valid when I/O6 shows the Ready state.
2. During Cache Program operation, I/O0 is only valid when I/O5 shows the Ready state, and I/O1 is only valid when I/O6 shows the Ready state.

Read Status2 Definition

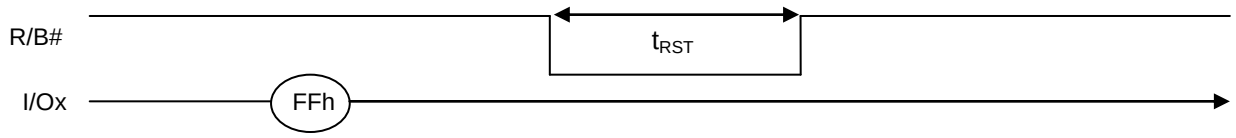
	I/O0	I/O1	I/O2	I/O3	I/O4	I/O5	I/O6	I/O7
Definition	Pass: 0 Fail: 1	Pass: 0 Fail: 1	Pass: 0 Fail: 1	Pass: 0 Fail: 1	Pass: 0 Fail: 1	Busy: 0 Ready: 1	Busy: 0 Ready: 1	Protected: 0 Not Protected: 1
Read	NA	NA	NA	NA	NA	NA	Busy/Ready	Write Protect
Cache Read	NA	NA	NA	NA	NA	Flash array Busy/Ready	Host Busy/Ready	Write Protect
Page Program	Pass/Fail	Plane#0 Pass/Fail	Plane#1 Pass/Fail	NA	NA	NA	Busy/Ready	Write Protect
Cache Program	Pass/Fail	Plane#0 (N) Pass/Fail	Plane#1 (N) Pass/Fail	Plane#0 (N-1) Pass/Fail	Plane#1 (N-1) Pass/Fail	Flash array Busy/Ready	Host Busy/Ready	Write Protect
Block Erase	Pass/Fail	Plane#0 Pass/Fail	Plane#1 Pass/Fail	NA	NA	NA	Busy/Ready	Write Protect

Note:

1. (N) means current page, and (N-1) means previous page.
2. During Block Erase, Page Program or Copy-Back operation, I/O0, I/O1, and I/O2 are only valid when I/O6 shows the Ready state.
3. During Cache Program operation, I/O0, I/O1, and I/O2 are only valid when I/O5 shows the Ready state, and I/O3 and I/O4 is only valid when I/O6 shows the Ready state.

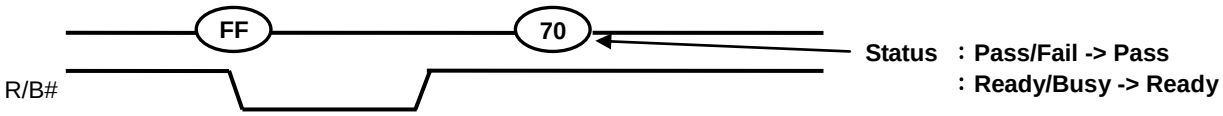
Reset

The device offers a reset function by command FFh. When the device is in 'Busy' state during any operation, the Reset operation will abort these operations except during power-on when Reset shall not be issued until R/B# is set to one (i.e. ready). The contents of memory cells being programmed are no longer valid, as the data will be partially programmed or erased. The response to a "FFh" Reset command input during the various device operations is as follows:



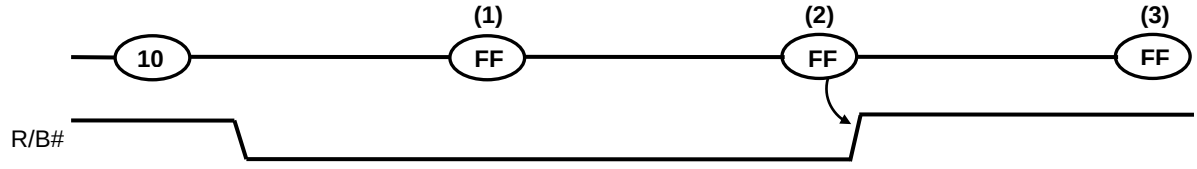
Reset Timing

When Status Read command (70h) is input after Reset Operation



Status Read after Reset operation

When two or more Reset commands are input in succession



The second FF command is invalid, but the third FF command is valid.

Successive Reset operation

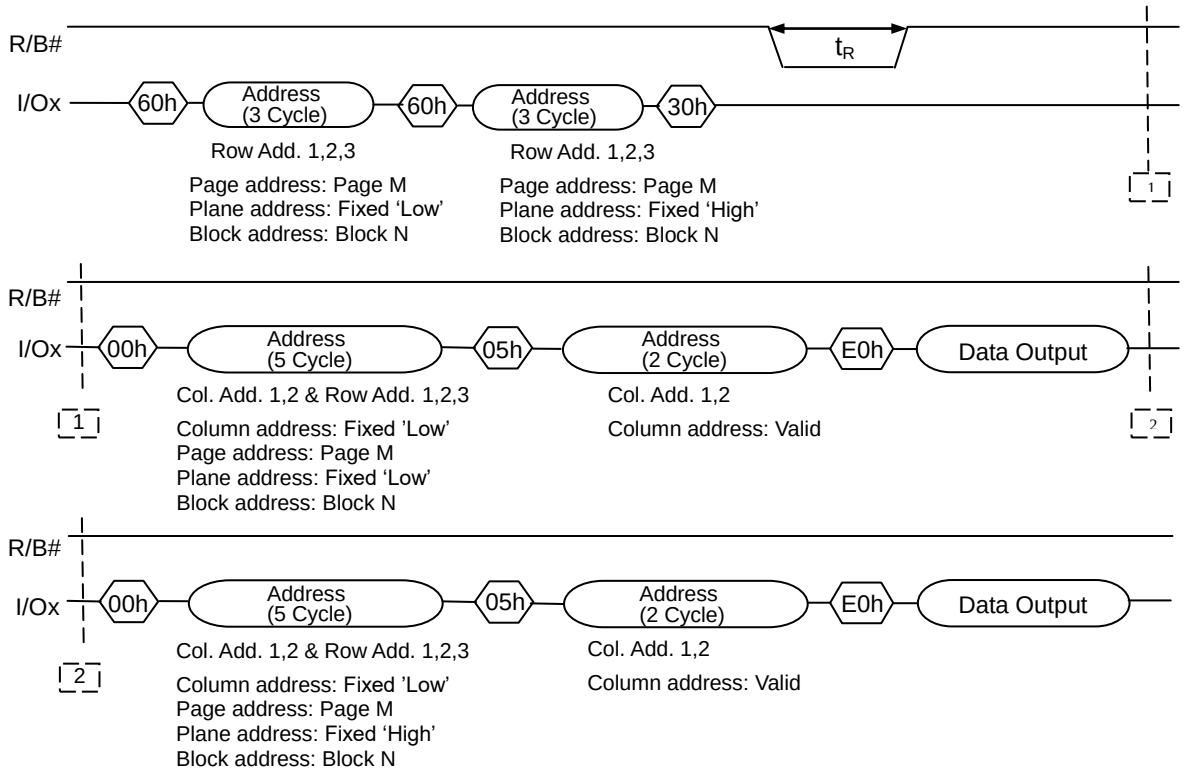
Two-Plane Operation

Two page address may be set over two planes. The page addresses and block address shall be identical when setting each plane. The same plane address shall not be set twice or more within a set of address setting sequence. The number of planes which are set for this operation shall be even. Multi page operation in this mode is also regarded as multi-plane operation.

Two-Plane Page Read

The Two-Plane Page Read operation is and extension of the Page Read operation. The device supporting Two-Plane Page Read operation also allows multiple Random data-output from each page (i.e. Two-Plane Page Random Data Output) once two- plane pages are loaded to page registers.

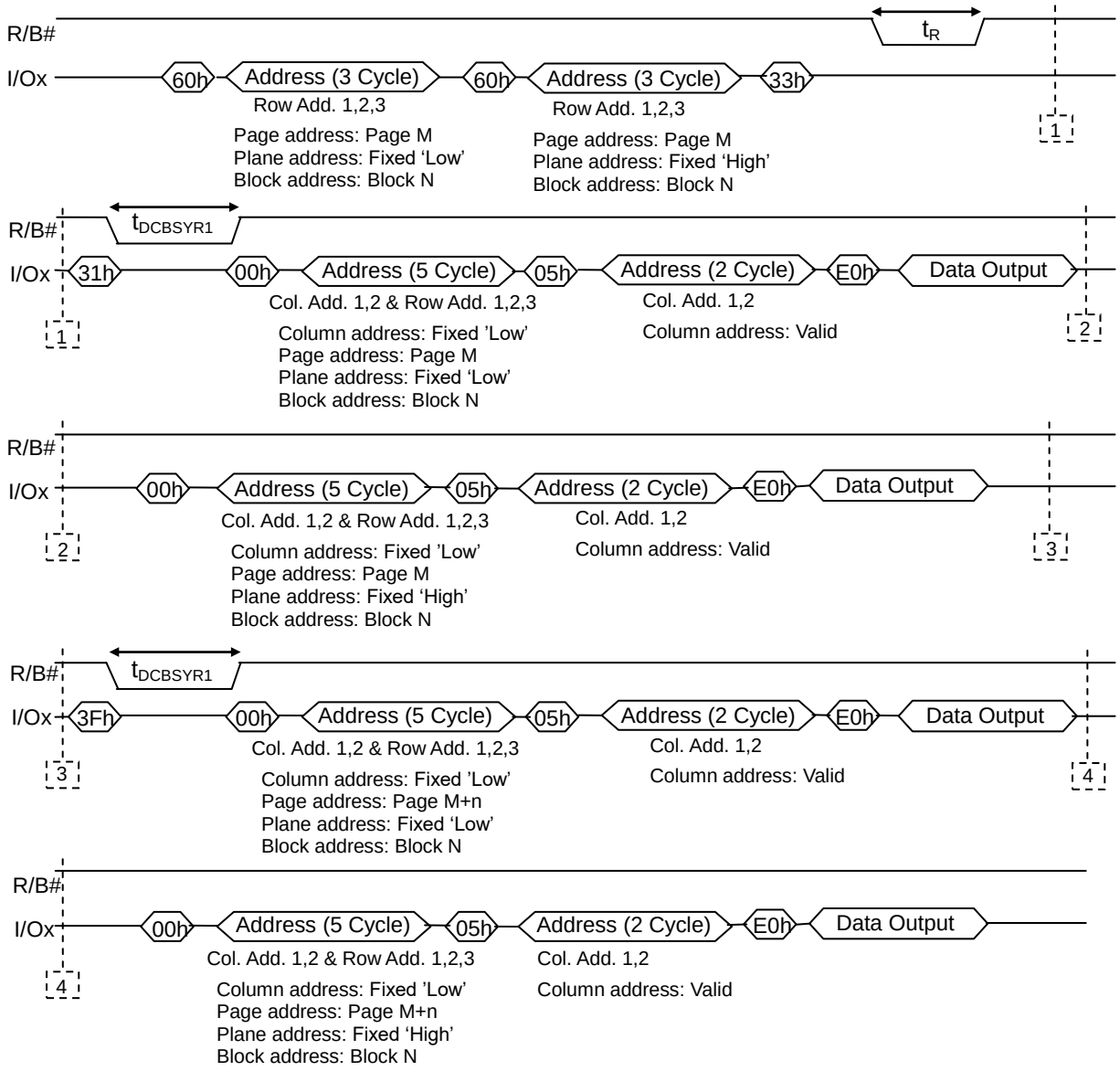
Once the data are loaded into the cache registers, the data on the first page can be read out by issuing the Two-Plane Random Data Output command. The data on other pages can be also read out using the identical command sequences. Figures below define Two-Plane Page Read and Two-Plane Random Data Output behavior and timings.



Two-Plan Page Read Operation

Two-Plane Cache Read

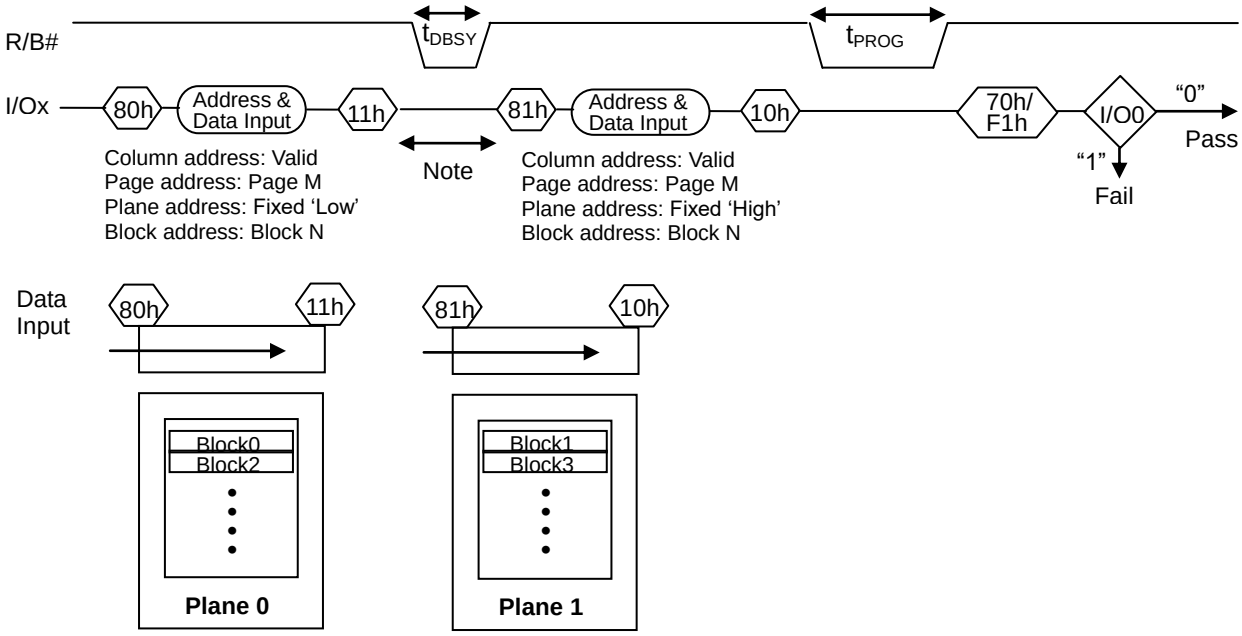
Two-Plane Sequential Cache Read operation provides fast sequential read function after the initial Two-Plane Page Read operation is set. With the primary command, once Two-Plane Page Read operation performs, next page data can be loaded to page register by command 31h without additional address setting while a host reads data, which is loaded by Two-Plane Page Read operation, from cache registers. Since the next page data are loaded to page registers during host read-out period, R/B# turns to high (i.e. ready) in a short time after command 31h although data loading by command 31h is being performed internally. If the previous data is still being loaded after command 31h, R/B# busy state may takes as long as t_R, hence the maximum time of $t_{DCBSYR1}$ is identical to t_R. At the last page, command 3Fh shall be issued to transfer data from page registers to cache registers. Two-Plane Sequential Cache Read operation shall work only within a block of each plane and shall not be continued over the boundary of plane. Figure below define Two-Plane Sequential Cache Read behavior and timings.



Two-Plane Cache Read Operation with Two-Plane Random Data Out

Two-Plane Page Program

Two-Plane Program function extends and effective programmable page size using multiple pages. When a host moves to load data for another page, command 11h for the second command is used. After 11h command, R/B# returns high (i.e. ready) in a short period of time since it is not actual programming operation. At the last page loading, command 81h is issued before loading data and command 10h after data loading is issued for the second command. After command 10h, all loaded data in each page stars to be programmed to Flash array simultaneously. Figure below defines Two-Plane Page Program behavior and timings.

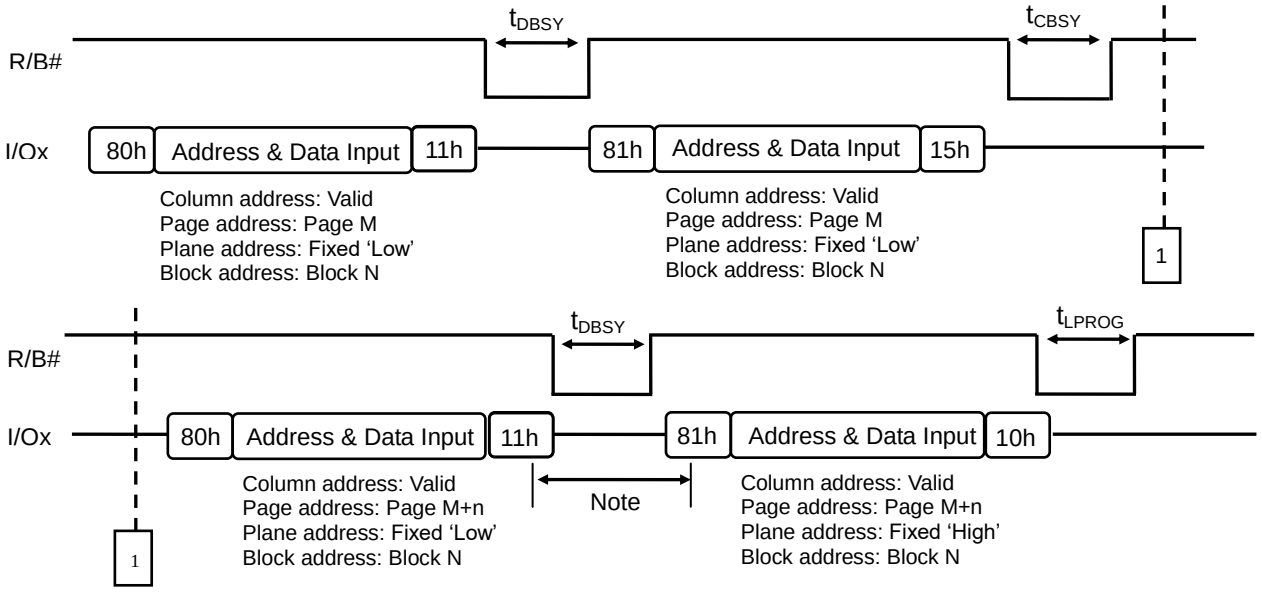


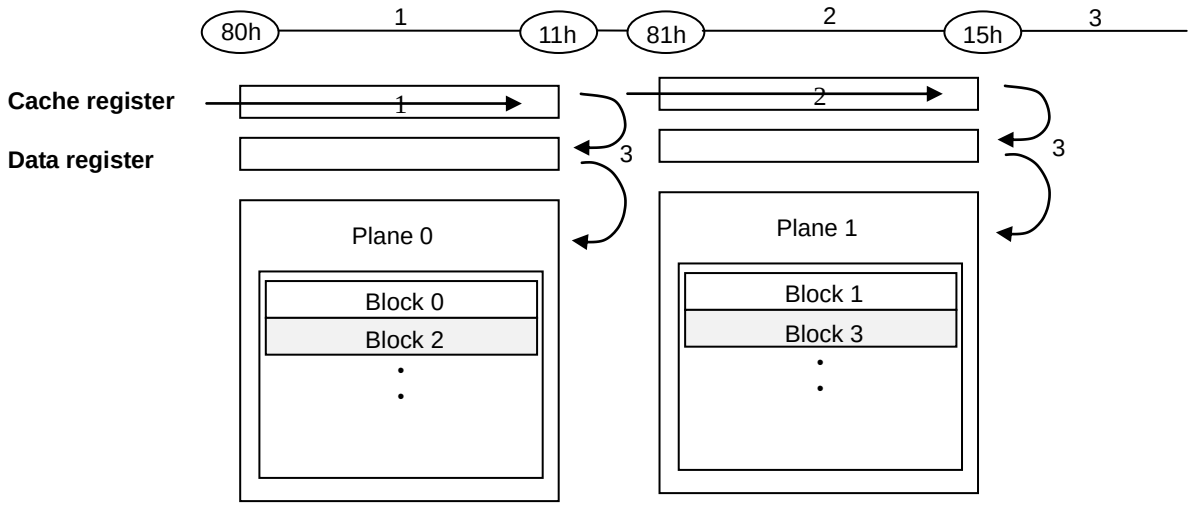
Note: The page address and block address shall be the same in Two-Plane Page Program operation.

Two-Plane Page Program

Two-Plane Cache Program

The Two-Plane Cache Program is an extension of the Cache Program. After loading pages for Two-Plane Cache Program, command 15h is issued. After command 15h, R/B# returns high once transferring data from cache register to page register is completed. Internal program operation is in progress after R/B# returns while other pages are loaded by a host. At the last page loading for the entire Two-Plane Cache Program, command 10h is required to finalize the operation and R/B# stays busy as long as t_{LPROG}. Two-Plane Cache Program operation shall work only within a block of each plane and shall not be continued over the boundary of plane. The activated planes for the first Two-Plane Cache Program shall be kept using in the next address sequence until Two-Plane Cache Program operation is completed by command 10h. Figure below defines Two-Plane Cache Program behavior and timings.



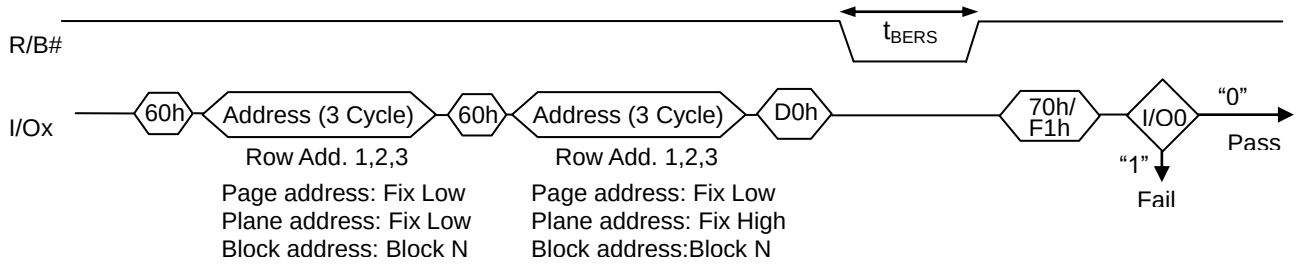


Note: The page address and block address shall be the same in Two-Plane Page Program operation.

Two-Plane Cache Program

Two-Plane Block Erase

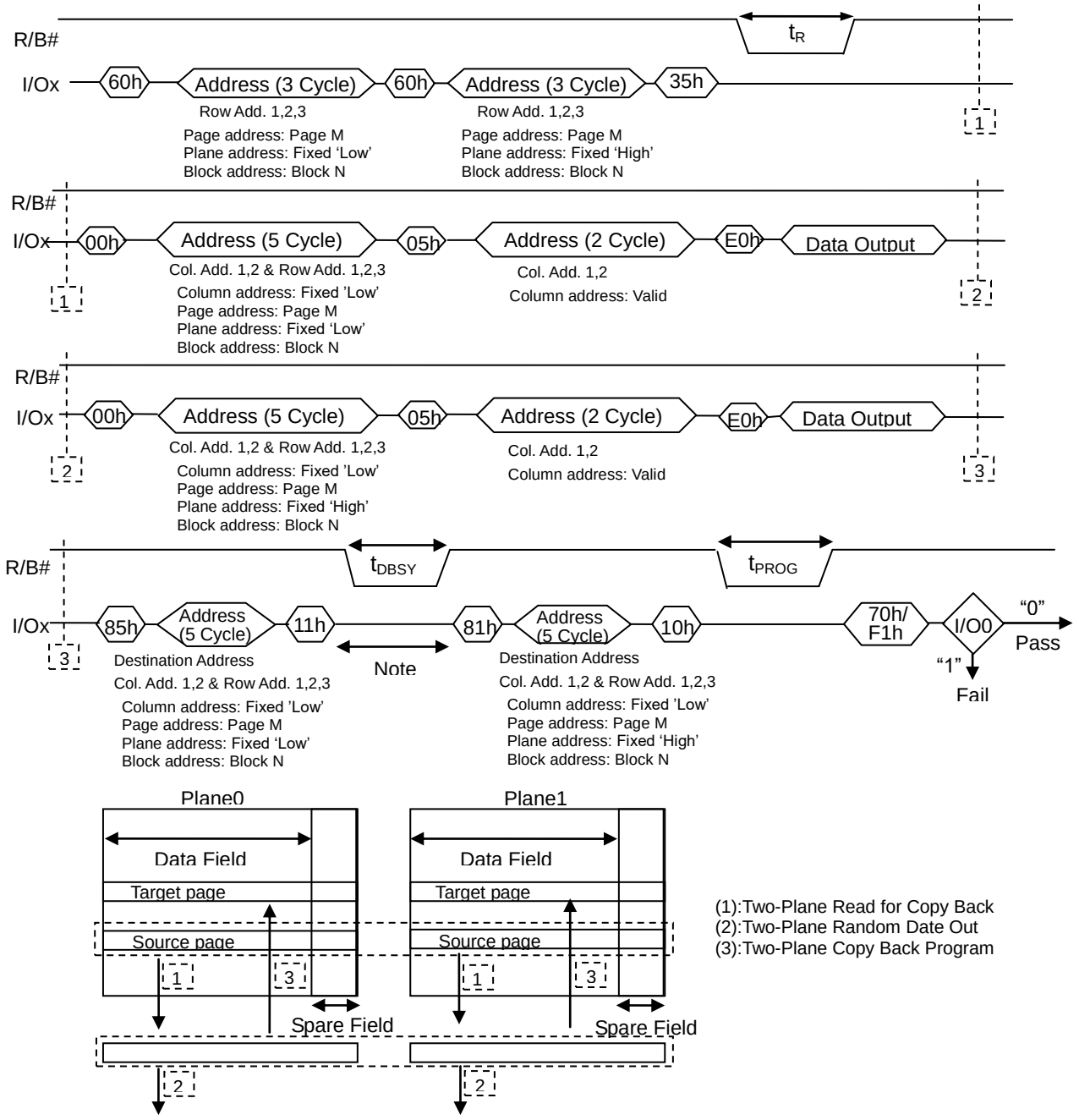
Two-Plane Block Erase allows users to erase multiple blocks comprising a block of each plane simultaneously. The same plane address shall not be set twice within a set of address setting sequence for the Two-Plane Block Erase operation. Figure below defines Two-Plane Block Erase behavior and timings.



Two-Plane Block Erase Operation

Two-Plane Copy-Back Program

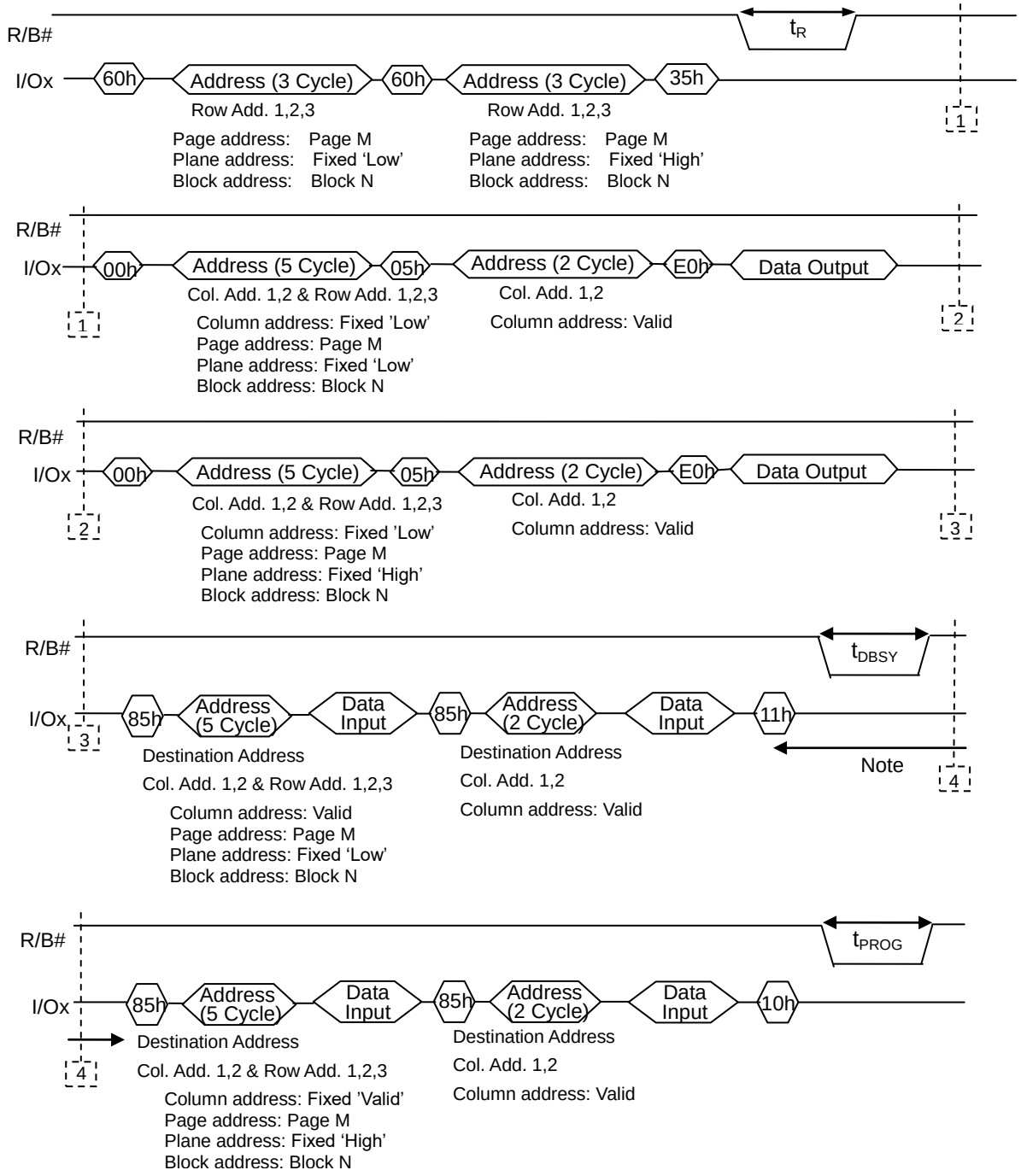
The Two-Plane Copy-Back Program is an extension of the Copy-Back Program. Two-Plane Copy-Back Program operation is executed two sets of commands. Two-Plane Read for Copy-Back and Two-Plane Copy-Back Program. The read data shall be copied back to a page in the same plane. Figures below define Two-Plane Copy-Back Program behavior and timings.



(1):Two-Plane Read for Copy Back
(2):Two-Plane Random Date Out
(3):Two-Plane Copy Back Program

Note: The LSB of page address shall be the same between source and destination pages. In other words, the page of even page address can't be copied to the page of odd page address, and the page of odd page address can't be copied to the page of even page address as well.

Two-Plane Copy-Back Program Operation



Note: The page address and block address shall be the same in Two-Plane Page Program operation.

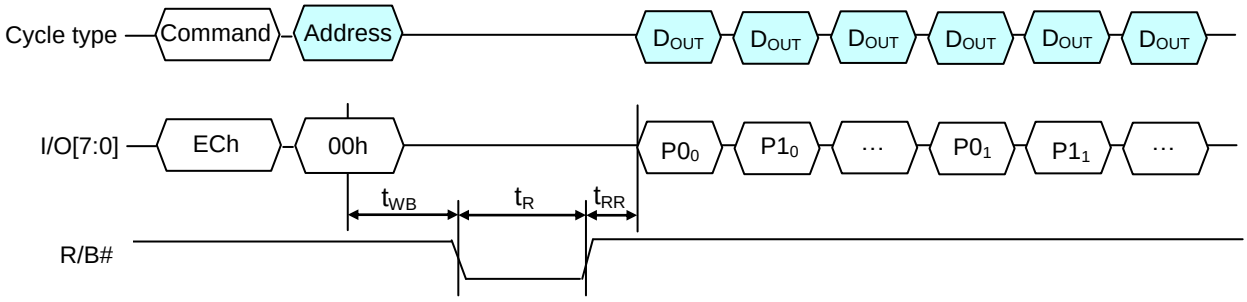
Two-Plane Copy-Back Program with Random Data Input Operation

Read Parameter Page Operation

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 the die(s) on the target is 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 ECh command is followed by one 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.

A minimum of three copies of the parameter page are stored in the device. Each parameter page is 256 bytes. Random Data Output (05h-E0h) 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 Page Operation

Parameter Page Data Structure

Byte	Description	Value
0-3	Parameter page signature ("O", "N", "F", "I")	4Fh, 4Eh, 46h, 49h
4-5	Revision number	02h, 00h
6-7	Features supported	10h, 00h
8-9	Optional commands supported	31h, 00h
10-31	Reserved	All 00h
32-43	Device manufacturer	50h, 4Fh, 57h, 45h, 52h, 43h, 48h, 49h, 50h, 20h, 20h, 20h
44-63	Device model	50h, 53h, 52h, 32h, 47h, 41h, 33h, 30h, 43h, 54h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h
64	Manufacturer ID	C8h
65-66	Date code	00h, 00h
67-79	Reserved	All 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	Number of maximum bad blocks per unit	28h, 00h
105-106	Block endurance	05h, 04h
107	Guaranteed valid blocks at beginning of target	01h
108-109	Block endurance of guaranteed valid blocks	00h, 00h
110	Number of partial programs per page	04h
111	Partial programming attributes	00h
112	Number of bits ECC	08h
113	Number of Interleaved address bits	01h
114	Interleaved operation attributes	0Ch
115-127	Reserved	All 00h
128	I/O pin capacitance	0Ah
129-130	Timing mode support (Reserved)	1Fh, 00h
131-132	Program cache timing mode support (Reserved)	1Fh, 00h
133-134	t_{PROG} (max)	BCh, 02h
135-136	t_{BERS} (max)	10h, 27h

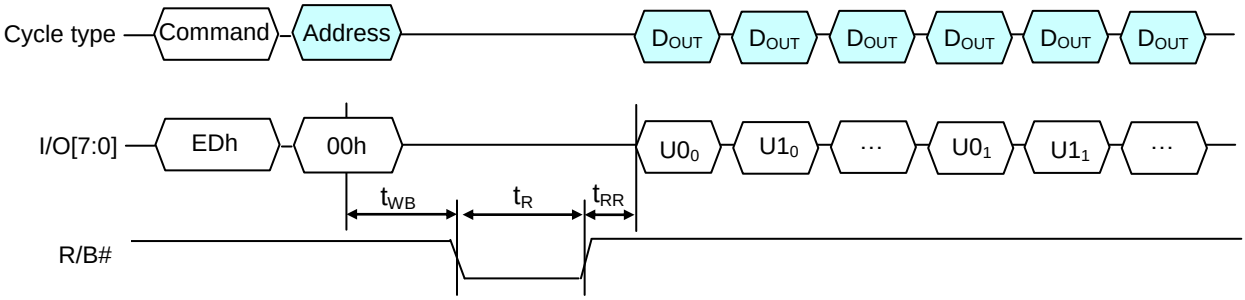
137-138	t_R (max)	19h, 00h
139-140	t_{CCS} (min)	46h, 00h
141-163	Reserved	All 00h
164-165	Vendor-specific revision number	00h, 00h
166	Two-Plane Page Read support Bit[7:1]: Reserved (0) Bit 0: 0= Doesn't support Two Plane Page Read	01h
167	Read cache support Bit[7:1]: Reserved (0) Bit 0: 0= Doesn't support ONFI-specific read cache	01h
168	Read Unique ID support Bit[7:1]: Reserved (0) Bit 0: 0= Doesn't support ONFI-specific Read Unique ID	01h
169	Programmable output impedance support Bit[7:1]: Reserved (0) Bit 0: 0= Doesn't support programmable output impedance support	00h
170	Number of programmable output impedance support settings Bit[7:3]: Reserved (0) Bit[2:0]: Number of programmable IO output impedance settings	00h
171	Reserved	00h
172	Programmable R/B# pull-down strength support Bit[7:1]: Reserved (0) Bit 0: 0= Doesn't support programmable R/B# pull-down strength	00h
173	Reserved	00h
174	Number of programmable R/B# pull-down strength support Bit[7:3]: Reserved (0) Bit[2:0]: Number of programmable R/B# pull-down strength settings	00h
175	OTP mode support Bit[7:2]: Reserved (0) Bit 1: 0= Doesn't support Get/Set Feature command set Bit 0: 1= support OTP mode	01h
176	OTP page start Bit[7:0] = Page where OTP page space begins	00h
177	OTP Data Protect address Bit[7:0] = Page address to use when issuing OTP Data Protect command	00h
178	Number of OTP pages Bit[15:5]: Reserved (0) Bit[4:0] = Number of OTP pages	1Eh
179	OTP Feature Address	90h
180-253	Reserved	All 00h
254-255	Integrity CRC	Set at test
256-511	Values of bytes 0-255	Values of bytes 0-255
512-767	Values of bytes 0-255	Values of bytes 0-255
768+	Additional redundant parameter pages	

Read Unique ID Operation

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 the die(s) on the target is 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 EDh command is followed by one 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.

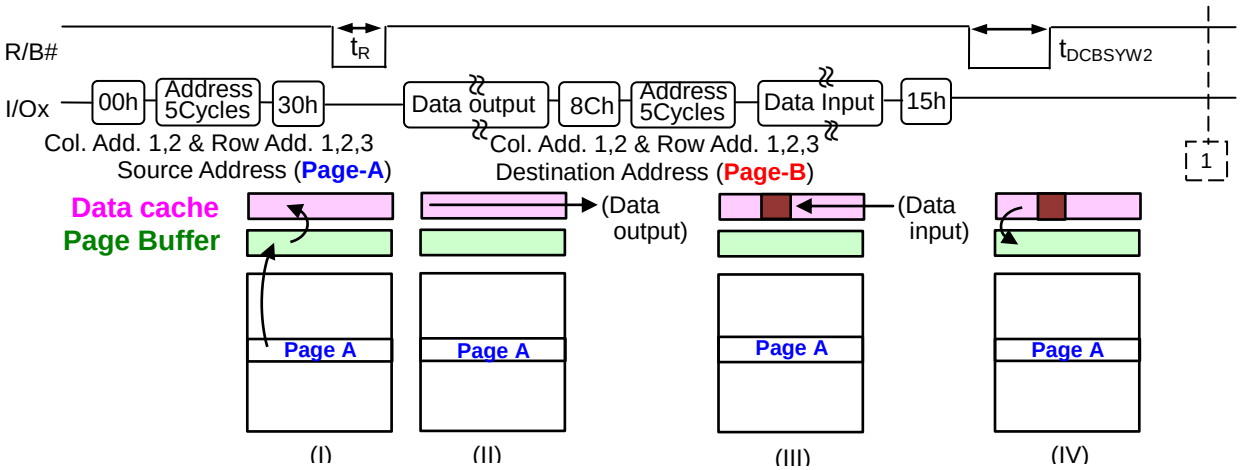
Sixteen copies of the unique ID data are store in the device. Each copy is 32 bytes. The first 16 bytes of a 32-byte copy are unique ID data, and the second 16 bytes are the complement of the first 16 bytes of FFh, then that copy of the unique ID data is correct. In the event that a non-FFh result is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



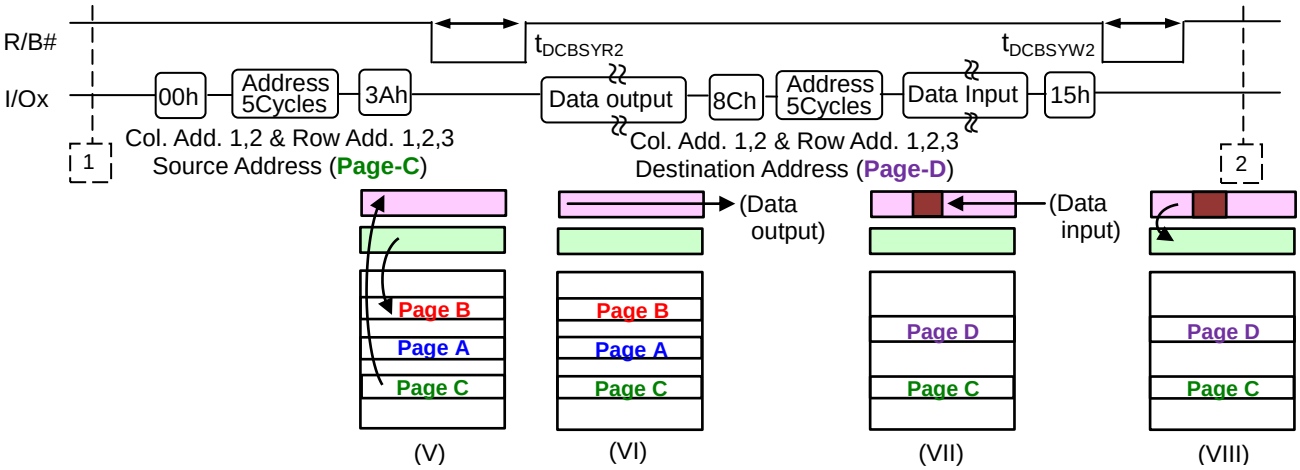
Read Unique ID Operation

Page Copy

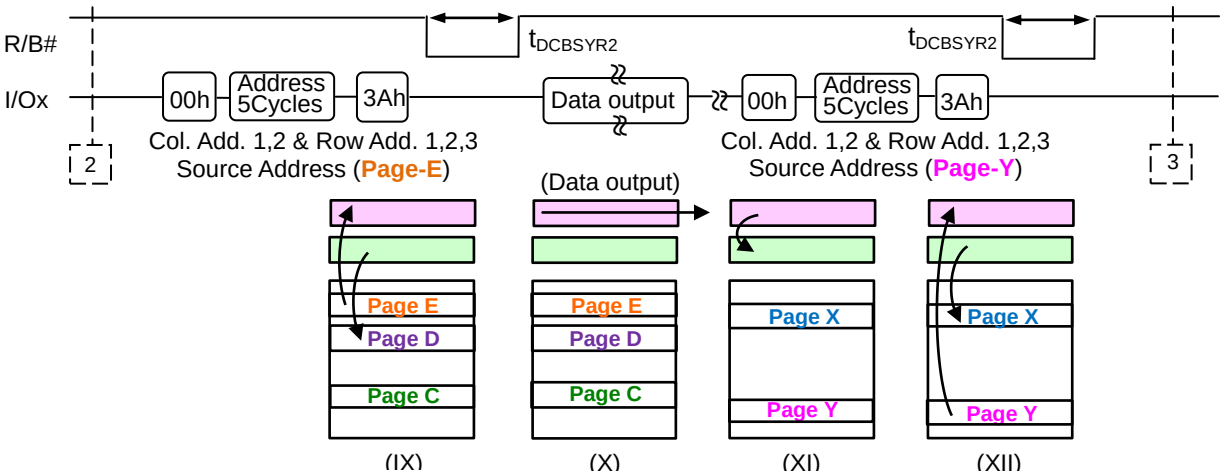
By using Page Copy, data in a page can be copied to another page after the data has been read out. When the block address changes (increments) this sequenced has to be started from the beginning.



I. Data for Page A is transferred to the Data Cache
II. Data for Page A is read out
III. Copy Page address B is input and if the data needs to be changed, changed data is input
IV. Data Cache for Page B is transferred to the Page Buffer



V. Data for Page C is transferred to Data Cache while the data of Page B is being programmed
VI. After the Ready state, Data for Page C is output from the Data Cache
VII. Copy Page address D is input and if the data needs to be changed, changed data is input
VIII. After programming of page B is completed, Data Cache for Page D is transferred to the Page Buffer

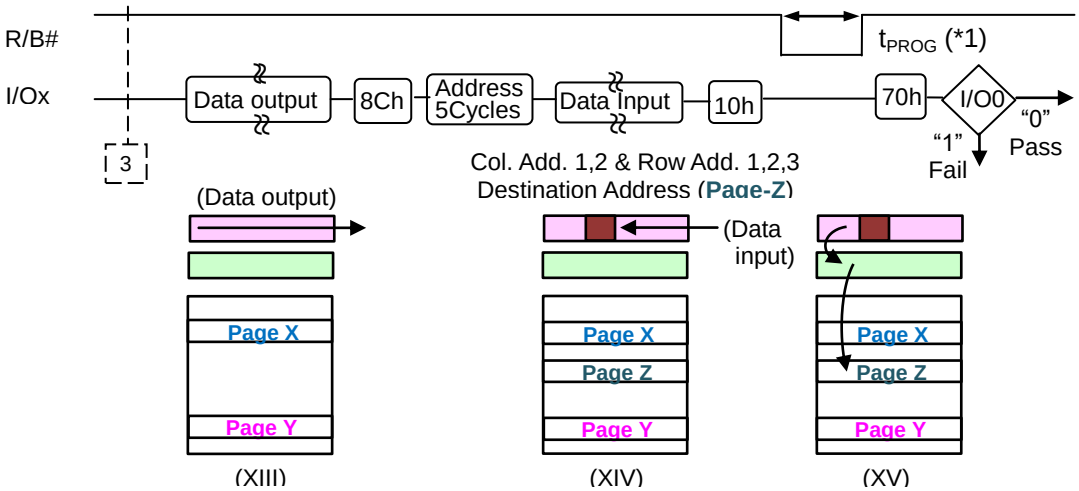


IX. By the 15h command, the data in the Page Buffer is programmed to Page D. Data for Page E is transferred to the Data cache

X. Data for Page E is read out

XI. Data Cache for Page X is transferred to the Page Buffer

XII. The data in the Page Buffer is programmed to Page X. Data for Page Y is transferred to the Data Cache



XIII. After the Ready state, Data for Page Y is output from the Data Cache

XIV. Copy Page address Z is input and if the data needs to be changed, changed data is input

XV. By issuing the 10h command, the data in the Page Buffer is programmed to Page Z

(*1) Since the last page programming by the 10h command is initiated after the previous cache program, the tPROG here will be expected as the following,
tPROG = tPROG of the last page + tPROG of the previous page - (command input cycle + address input cycle + data output/input cycle time of the last page)

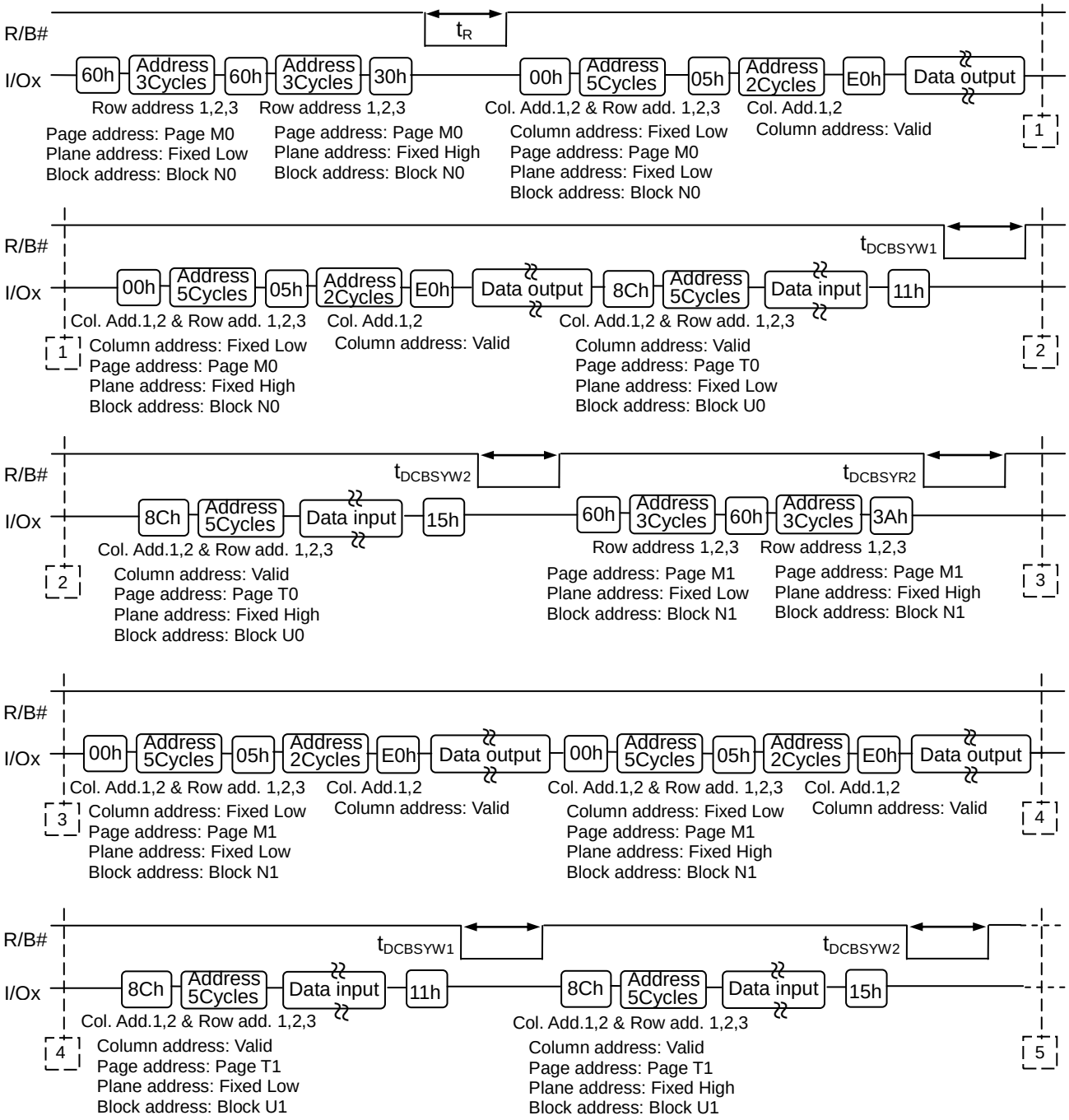
Note) This operation needs to be executed within Plane-0 or Plane-1.

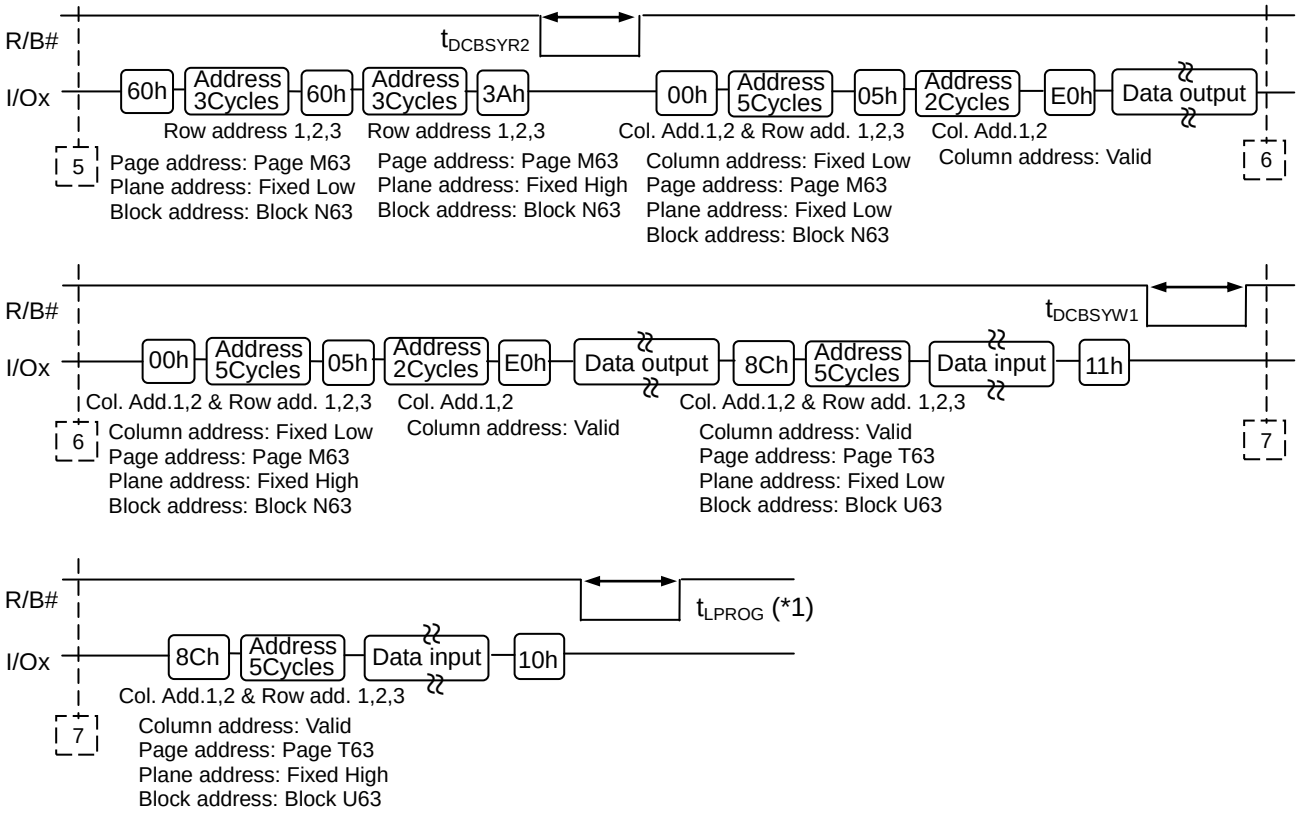
Data input is required only if previous data output needs to be altered.
If the data has to be changed, locate the desired address with the column and page address input after the 8Ch command, and change only the data that needs be changed.
If the data does not have to be changed, data input cycles are not required.

Make sure WP is held to High level when Page Copy operation is performed.
Also make sure the Page Copy operation is terminated with 8Ch-10h command sequence

Two-Plane Page Copy

By using Two-Plane Page Copy, data in two pages on different plane can be copied to other pages after the data has been read out. When each block address changes (increments) this sequence has to be started from the beginning. Same page address (A13 to A18) and same block address (A20~A29) within two plane has to be selected.





(*1) t_{LPROG}: Since the last page programming by 10h command is initiated after the previous cache program, the t_{LPROG}* during cache programming is given by the following equation.

$t_{LPROG} = t_{PROG}$ of the last page + t_{PROG} of the previous page-A

A = (command input cycle + address input cycle + data output/input cycle time of the last page)

If "A" exceeds the t_{PROG} of previous page, t_{PROG} of the last page is t_{LPROG} max.

Note)

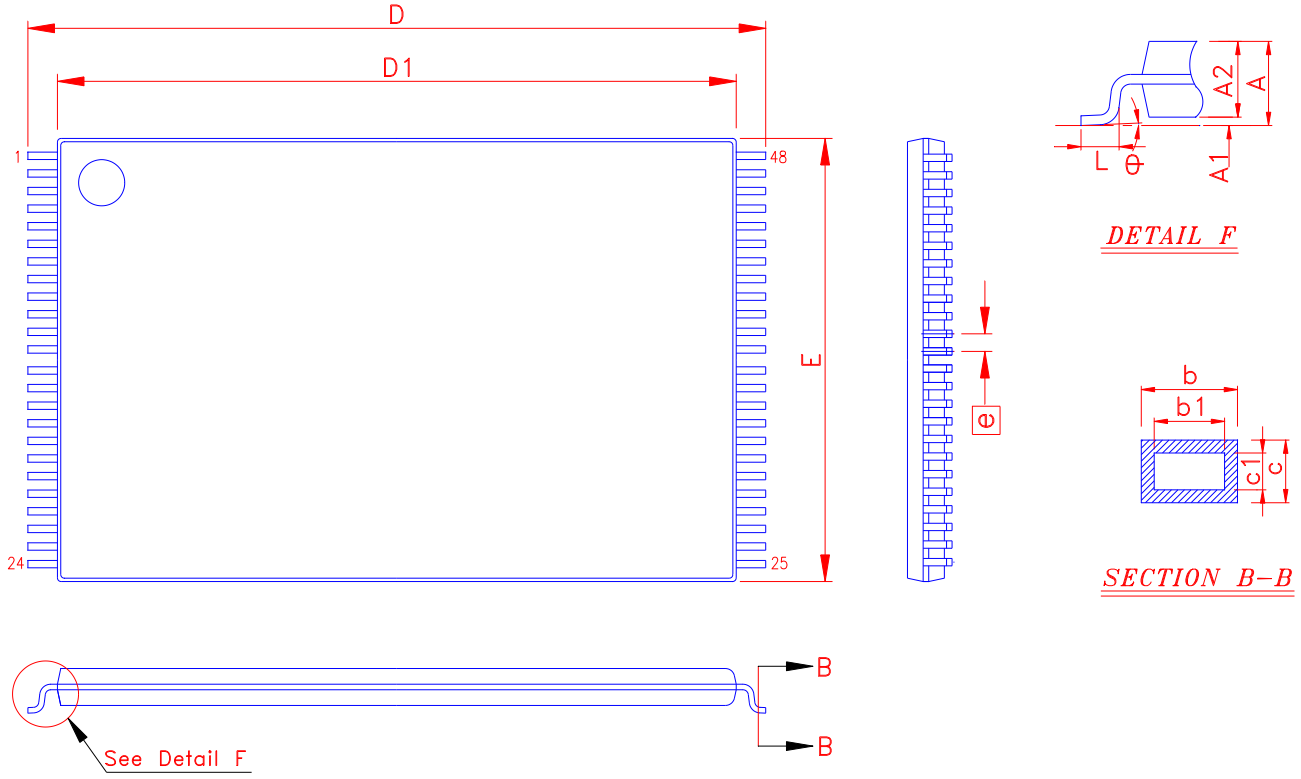
This operation needs to be executed within each Plane.

Data input is required only if previous data output needs to be altered.
If the data has to be changed, locate the desired address with the column and page address input after the 8Ch command, and change only the data that needs be changed.
If the data does not have to be changed, data input cycles are not required.

Make sure WP is held to High level when Two-Plane Page Copy operation is performed.
Also make sure the Multi Page Copy operation is terminated with 8Ch-10h command sequence

PACKING DIMENSION

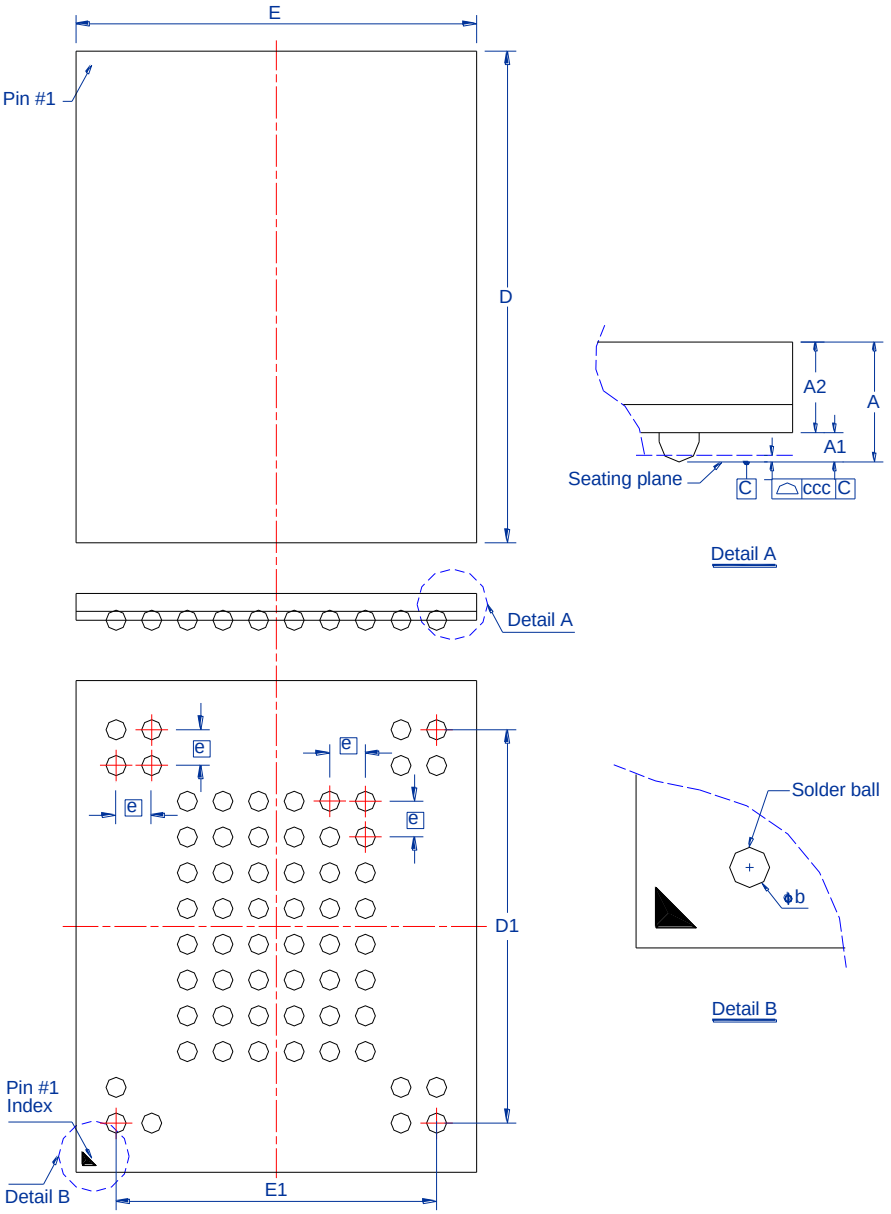
48-LEAD TSOP(I) (12x20 mm)



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)

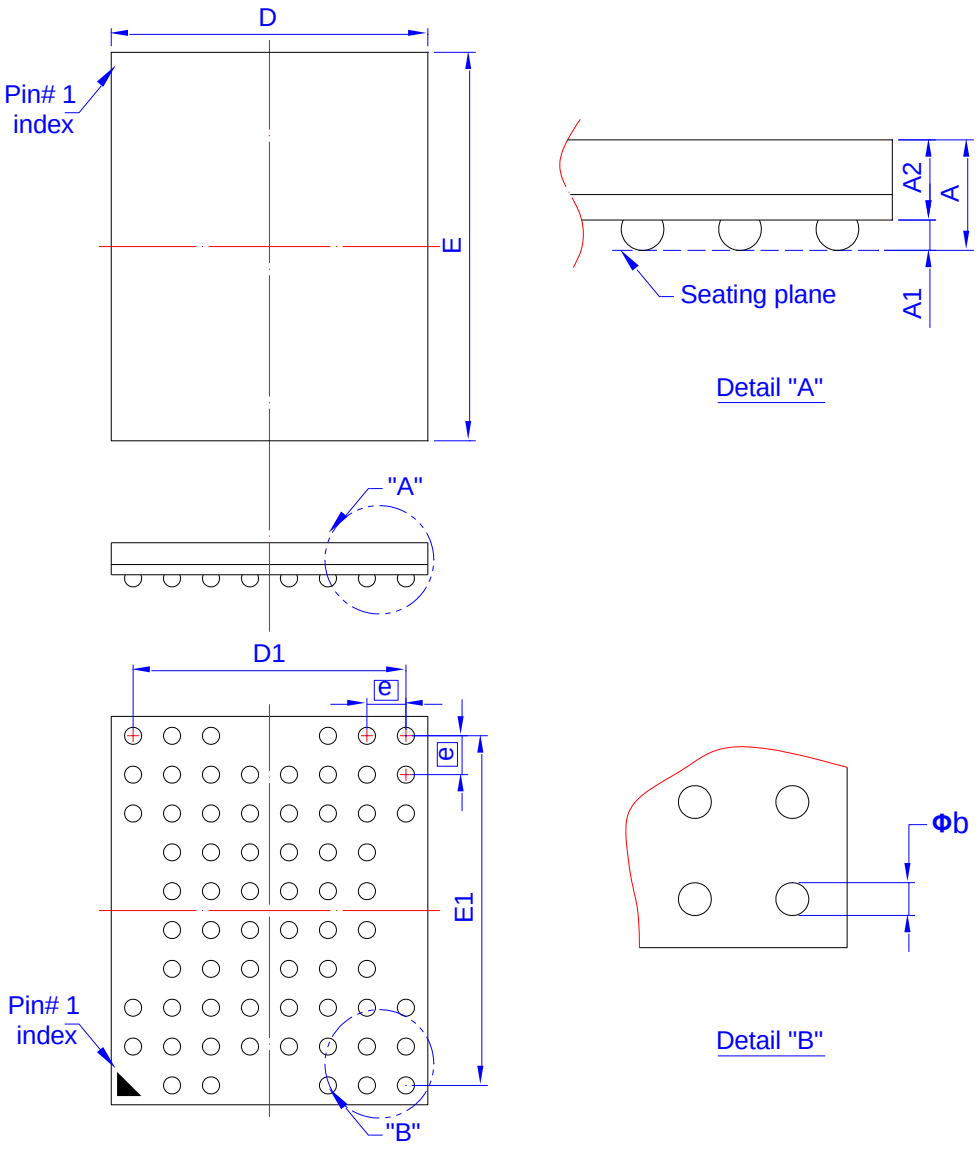


Symbol	Dimension in mm			Dimension in inch		
	Min	Norm	Max	Min	Norm	Max
A			1.00			0.039
A_1	0.25		0.35	0.010		0.014
A_2		0.60 BSC			0.024 BSC	
Φb	0.40		0.50	0.016		0.020
D	10.90	11.00	11.10	0.429	0.433	0.437
E	8.90	9.00	9.10	0.350	0.354	0.358
D_1		8.80 BSC			0.346 BSC	
E_1		7.20 BSC			0.283 BSC	
e	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

Controlling dimension : Millimeter.

PACKING DIMENSIONS

67-BALL Flash (6.5x8 mm)



Symbol	Dimension in mm			Dimension in inch		
	Min	Norm	Max	Min	Norm	Max
A			1.00			0.039
A_1	0.22	0.27	0.32	0.009	0.011	0.013
A_2	0.61	0.66	0.71	0.024	0.026	0.028
Φ_b	0.30	0.35	0.40	0.012	0.014	0.016
D	6.40	6.50	6.60	0.252	0.256	0.260
E	7.90	8.00	8.10	0.311	0.315	0.319
D_1	5.60 BSC			0.220 BSC		
E_1	7.20 BSC			0.283 BSC		
e	0.80 BSC			0.031 BSC		

Controlling dimension : Millimeter.
(Revision date : Jun 29 2014)

Revision History

Revision	Date	Description
1.0	2020.10.30	Original
1.1	2021.04.01	Modify Temperature marking

Important Notice

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.