JEDEC STANDARD

Wide I/O 2 (WideIO2)

JESD229-2

AUGUST 2014

JEDEC SOLID STATE TECHNOLOGY ASSOCIATION

NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or refer to www.jedec.org under Standards and Documents for alternative contact information.

Published by

3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

PRICE: Contact JEDEC

WIDE I/O 2 (WideIO2) STANDARD

(From JEDEC Board Ballot JCB-14-40, formulated under the cognizance of the JC-42.6 Subcommittee on Low Power Memories.)

1 Scope

This standard defines Wide I/O 2 (WideIO2), including features, functionality, AC and DC characteristics, packages, and micropillar signal assignments. The purpose of this standard is to define the minimum set of requirements for JEDEC compliant, 8 Gb through 32 Gb SDRAM devices with 4 or 8 64-bit wide channels using direct chip-to-chip attach methods for between 1 and 4 memory devices and a controller/buffer device. This standard was created using aspects of the following standards: DDR2 (JESD79-2), DDR3 (JESD79-3), LPDDR (JESD209), LPDDR2 (JESD209-2), LPDDR3 (JESD209-3) and WIO (JESD229-1). Each aspect of the standard will require approval by committee ballot(s). The accumulation of these ballots will then be incorporated into the WideIO2 standard.

The WideIO2 architecture is an evolution of the WIO architecture to enable bandwidth scaling with capacity.

2 General Description

2.1 Terms and Definitions

Within the WideIO2 standard, these terms have particular meanings:

Stack: All memory chips in the memory system taken together in one assembly. The WideIO2 standard supports memory stacks of up to 4 memory chips.

Slice: One memory chip in the stack of memory chips as shown in Figure 1.

Quadrant: A single memory chip is divided into 4 quadrants as shown in Figure 8.

Micropillar: An electrical connection between two stacked die. The connection is made between the lower die's top metal layer and an upper die's pad by cutting a hole in the passivation on the lower die and inserting a conducting pillar.

Rank: Multiple slices can be connected to a single channel in a multidrop fashion within the memory stack. The DRAM array connected to the channel is referred to as a rank. WideIO2 supports single and dual rank configurations.

Channel: A set of physically discrete connections within the WideIO2 interface that independently control a partition of the WideIO2 device. (see Figure 1).

NOTE The WideIO2 interface supports 4 or 8 physical channels. Each channel contains all the control, data, and clock signals necessary to independently control a partition of the WideIO2 device. Each channel can have different DRAM pages open, can be independently clocked, and can be in different power states. The physical channel also includes I/O power and ground signals. All power and ground signals for all channels must be at their appropriate levels for any portion of the WideIO2 device to operate correctly. The physical channel also contains a reset signal but the WideIO2 interface defines reset to be per slice rather than per channel.

2.1 Terms and Definitions (cont'd)

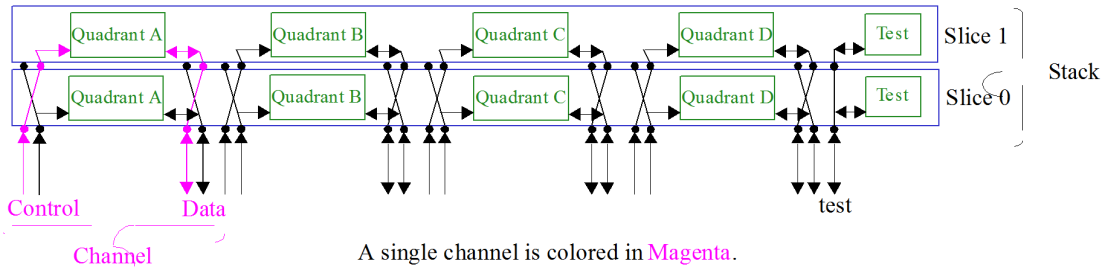


Figure 1 — Definition of Terms for WideIO2 stack

2.2 Key Features

. Support for 800MT/s and 1067MT/s data rates.

. 25.6GB/s and 34.1GB/s with four 64b channels (4x64 die)

. 51.2GB/s and 68.3GB/s with eight 64b channels (8x64 die)

. DRAM core frequencies of 200MHz and 266MHz.

. Configurable with1, 2, or 4 stacked die for bandwidth and capacity scaling.

. 4KB page size with 8 banks per channel for 4x64 die

. 2KB page size with 4 banks per channel for 8x64 die

. 64 Data Bits per channel.

. Support for burst lengths of 4 and 8.

. Complementary data strobe for every 16 data bits.

. Double Data Rate for command and data.

. Unterminated CMOS I/O signaling

. No PLL or DLL in the DRAM

. Per byte write data mask and data bus inversion

. Multiplexed Command Address (11 CA signals over 2 UI).

. Each rank in each channel will have its own set of Mode Registers.

. Each channel is independent.

. Support for 8, 16, and 32Gb DRAM die density.

. Per slice scan chain.

. Per slice Reset.

. Support for supplier specific Direct Access Mode test feature with 10 digital and 1 analog signals per quadrant.

. Support for GPIO Mode test access.

. Support for Post Package Repair

. Power micropillar count supports current requirements of low-power memory space.

. VDDQ = VDD2 = 1.1V

. VDD1 = 1.8V

2.3 Bandwidth vs. Capacity Relationship

The WideIO2 device is targeted to operate up to 800 or 1067MT/s with 64b per channel. The per die density will be either 8, 16, or 32 Gb. Depending on the configuration, the bandwidth and capacity scale as shown in Table 1. The bandwidth is calculated by the transfers x bytes/channel x number of channels. P22P configurations will double the capacity while the bandwidth stays constant.

Table 1 — Capacity vs. Bandwidth (8 - 32Gb Die Density)

Die	Configuration	Capacity	800 MT/s Band-width	1067 MT/s Band-width	Figure
4x64 Die	1 Slice, 4 Channel	1 - 4GB	25.6GB/s	34.1GB/s	Figure 2
	2 Slice, 8 Channel	2 - 8GB	51.2GB/s	68.3GB/s	Figure 3
	4 Slice, 8 Channel, P22P Dual Rank	4 - 16GB	51.2GB/s	68.3GB/s	Figure 4
8x64 Die	1 Slice, 8 Channel	1 - 4GB	51.2GB/s	68.3GB/s	Figure 5
	2 Slice, 8 Channel, P22P Dual Rank	2 - 8GB	51.2GB/s	68.3GB/s	Figure 6

2.4 WideIO2 Topologies

WideIO2 topologies require a shifting of signals through the stack. This shifting is done in the metal layers of the lower die. The WideIO2 topologies shown in this section are from an SOC point of view (SOC at bottom of stack). A 4 channel topology will only have a single channel (channel 0) for each of the 4 quadrant (A-D). The 4 channels will be referred to as: 0A, 0B, 0C, and 0D. An 8 channel topology will have a channel 0 and channel 1 for each of the 4 quadrants (A-D). The 8 channels will be referred to as: 0A, 1A, 0B, 1B, 0C, 1C, 0D, and 1D. All of the figures in this section will show one of the channels in a magenta color.

2.4.1 WideIO2 Topologies with 4x64 Die

2.4.1.1 1-High 4Ch x 64b

Figure 2 shows the baseline WideIO2 1 slice, 4 channel P2P topology. There are a total of 128 DQs per quadrant with only 64 being used in this topology.

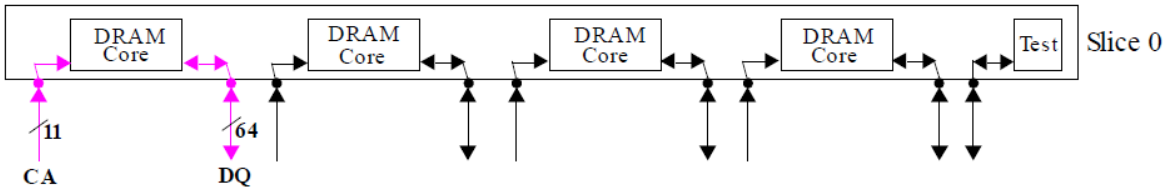


Figure 2 — P2P WideIO2: 1 Slice, 4 Channel

2.4.1.2 2-High 8Ch x 64b Capacity and Bandwidth Scaling

Figure 3 shows a WideIO2 2 slice, 8 channel P2P topology. Since this is an 8 channel, P2P topology, each channel from the second slice increases the overall bandwidth with a constant capacity per channel.

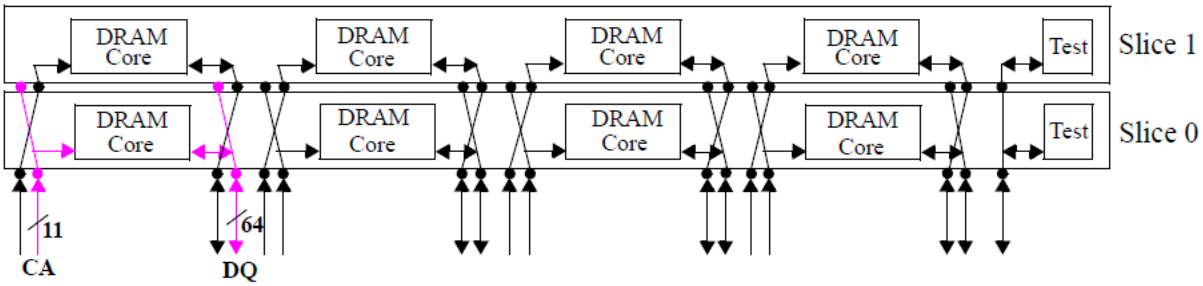


Figure 3 — P2P WideIO2: 2 Slice, 8 Channel

2.4.1.3 4-High, 8Ch x 64b, P22P Capacity and Bandwidth Scaling

Figure 4 shows a WideIO2 4 slice, 2 rank, 8 channel P22P topology with separate CA buses to avoid P24P loading on CA. This configuration scales both capacity and bandwidth.

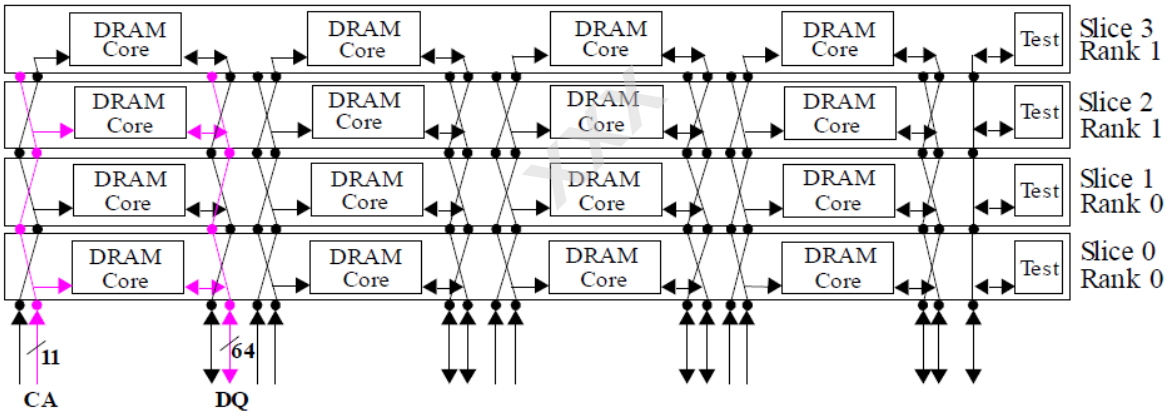


Figure 4 — P22P WideIO2: 4 Slice, 2 Rank, 8 Channel

2.4.2 WideIO2 Topologies with 8x64 Die

2.4.2.1 1-High 8Ch x 64b

Figure 5 shows a WideIO2 1-slice, 8-channel P2P topology.

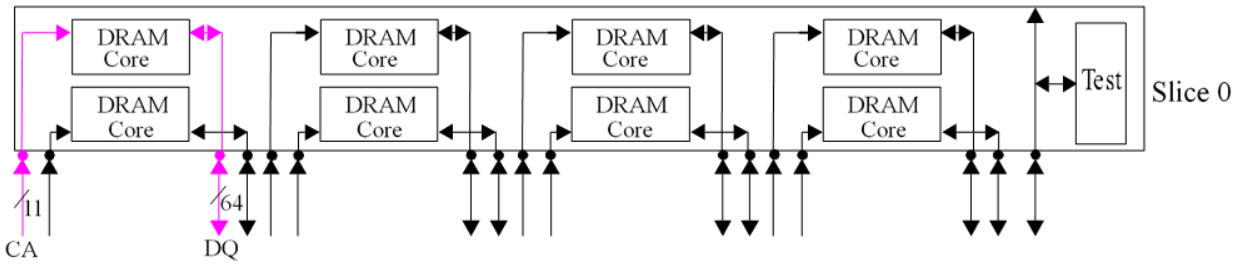


Figure 5 — P2P WideIO2: 1 Slice, 8 Channels

2.4.2.2 2-High 8Ch x 64b

Figure 6 shows a WideIO2 2-slice, 8-channel, P22P topology. The CS and CKE signals are staggered in pairs. CS2/CKE2 is connected to CS0/CKE0 on upper die and CS3/CKE3 is connected to CS1/CKE1 on upper die.

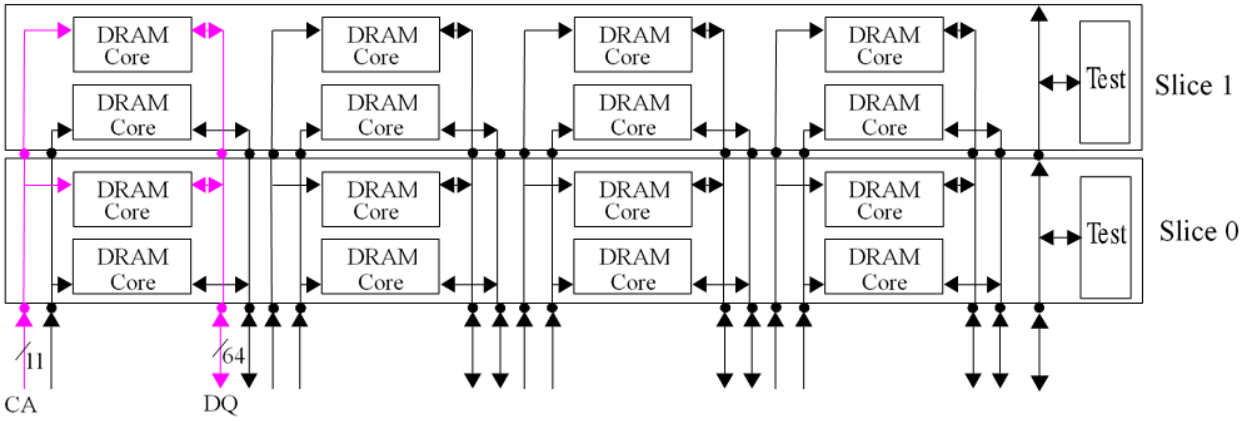


Figure 6 — P22P WideIO2: 2 Slice, 8 Channels

2.5 Micropillar-out

2.5.1 Micropillar Definition and Description

Note on nomenclature: unless otherwise designated, each channel is independent and implements independent sets of the designated micropillars. Signal naming convention designates quadrant and channel within each quadrant as follows:
<SignalName>[1:0][D:A][msb:lsb]

. [1:0] denotes which channel within the quadrant
. [D:A] denotes which quadrant
. [msb:lsb] denotes index of bus
. _n for active low, _t for true, and _c for complement polarity

For example DQ1A13 would be DQ13 in quadrant A channel 1.

Table 2 — Micropillar Definition and Description

Name Signal[Ch][Quad][bit]	Type	Description
CK[1:0][D:A]_t, CK[1:0][D:A]_c	Input	**Clock:** CK_t and CK_c are Complementary clock inputs to each channel. All Command signals are sampled on the positive edges of CK_t and CK_c providing a double rate command bus. The clocks are not free running, they will toggle as commands are latched into the device and long enough after the command is latched to allow the command to complete. Clock is defined as the complementary pair CK_t and CK_c.
CA[1:0][D:A][10:0]	Input	**Command/Address:** There are 11 command/address signals per channel and a command packet is 2UI in length giving a payload of 22 bits of information. The CA bus will be clocked by CK_t/CK_c clocks.
CS[D:A][3:0]_n	Input	**Chip Select:** CS_n is considered part of the command code. Each of CS[D:A][3:0]_n address a single rank on each of the channels. See Table 20 - Command Truth Table for command code descriptions. See Table 6 for CS_n signal mapping.
CKE[D:A][3:0]	Input	**Clock Enables:** CKE HIGH activates and CKE LOW deactivates internal clock signals, device input buffers and output drivers. Power savings modes are entered and exited through CKE transitions. CKE is considered part of the command code. See Table 21 - Command Truth Table for command code descriptions. See Table 6 for CKE signal mapping.
DQ[1:0][D:A][63:0]	I/O	**Data Inputs/Output:** Bi-directional data bus. 64 DQs per channel.
DQS[1:0][D:A][3:0]_t DQS[1:0][D:A][3:0]_c	I/O	**Data Strobe (Bi-directional, Complementary):** The data strobe is bidirectional (used for read and write data) and complimentary (DQS_t and DQS_c). It is output with read data and input with write data. DQS_t and DQS_c are edge-aligned to read data and centered with write data. Each DQS pair strobes 16 DQ I/Os.
DMI[1:0][D:A][7:0]	I/O	**Data Mask and Data Bus Inversion:** DMI is a bidirectional signal sampled on the rising edges of DQS_t and DQS_c. See section 3.3.
RST[3:0]_n	Input	**Reset:** Unidirectional reset inputs. These are per slice reset signals.
SEN[2:0]	Input	**Boundary Scan Enable:** Used to enable normal operation, boundary scan, or 1 of 6 vendor specific scan chains. Boundary scan modes may be operational in all normal and test modes, as defined by each individual memory vendor.
SSH_n	Input	**Boundary Scan Shift:** There is one SSH_n provided to all slices in the WideIO2 stack using Through routing.
SDI	Input	**Boundary Scan Serial Data In:** There is one SDI provided to all slices in the WideIO2 stack using Through routing.
SCK	Input	**Boundary Scan Clock:** There is one SCK provided to all slices in the WideIO2 stack using Through routing.

Table 2 — Micropillar Definition and Description

Name Signal[Ch] [Quad][bit]	Type	Description
SCS[3:0]_n	Input	**Boundary Scan Chip Selects:** There is a one SCS[3:0]_n per slice using Staggered routing.
SDO[3:0]	Output	**Boundary Scan Output:** There is one SDO output per slice using Staggered routing.
TEST	Input	**TEST:** This input enables memory GPIO test mode. It may be routed through a controller I/O buffer before driving the memory I/O pad.
DAA[D:A]	I/O	**Direct Access Analog:** This is a vendor specific test feature. These I/Os provide an analog direct access path to DRAM core signals for test/debug purposes. They must be routed directly to external package I/O pads to allow unbuffered visibility to an internal analog signal. They can be connected through the vertical stack in a multidrop topology as defined by each individual memory vendor. There is one DAA per quadrant. DAA pins may be operational in all normal and test modes, as defined by each individual memory vendor.
DA[D:A][9:0]	I/O	**Direct Access:** This is a vendor specific test feature. These I/Os provide digital direct access to internal DRAM core signals for test/debug purposes. They can be connected through the vertical stack in a multidrop topology as defined by each individual memory vendor. There are 10 digital DA's per quadrant. One of these I/Os may be used to enabled Direct Access test mode. If DA Mode is supported by an SOC design, they must be routed directly to external package I/O pads to allow unbuffered access to these signals. When not in DA mode each DA pin on the device will either be left floating or have a keeper to pull it low. The SOC is not required to pull up or pull down any of the DA pins and should leave these signals floating. The device must account for up to 5uA of SOC leakage current on these signals.
NC	NA	**No Connection:** This micropillar is a spare and not connected in the WideIO2 device.
VDD1	Supply	**Core Voltage Supply 1:** Core power supply.
VDD2	Supply	**Core Voltage Supply 2:** Core power supply.
VDDQ	Supply	**I/O Voltage Supply:** Power supply for the DQ, DQS, and DMI I/O buffers.
VSS	Supply	**Ground**
VSSQ	Supply	**I/O Ground**

NOTE 1 All DA pins must be ESD harden, in the WideIO2 device, up to 2KV HBM (Human Body Model) level or 200V for MM (Machine Model) level, as stated in ANSI/ESDA/JEDEC JS-001-2012 spec.

2.5.2 Quadrant Micropillars

Table 3 — Micropillar Definitions (Per Quadrant)

Micropillar Type	Count	Description
VDD1	12	Core Voltage 1
VDD2	60	Core Voltage 2
VSS	72	Core Ground
VDDQ	42	I/O Voltage
VSSQ	42	I/O Ground
DQ	128	Data
DMI	16	Data Mask and Data Bus Inversion
DQS	16	Complementary Data Strobe: DQS_t, DQS_c
CA	22	Encoded address and command
CS_n	4	Chip Select
CKE	4	Clock Enable
CK	4	Complementary Clock: CK_t, CK_c
Misc	5	Miscellaneous (RST_n, Serial Port, TEST (average. per quadrant))
DAA/DA	11	Direct Access for Test (1 analog, 10 digital)
Total	438	Per Quadrant

NOTE 1 The power and ground signal counts remain the same for both 4 channel and 8 channel configurations (all power and grounds need to be connected for all configurations).
NOTE 2 For configurations with 4 channels the SOC will need to connect their unused signals to a known value.

2.5.3 Channel Interface Signals

Table 4 — Channel Interface Signals

Micropillar Type	Count	Description
DQ	64	Data
DMI	8	Data Mask and Data Bus Inversion
DQS	8	Complementary Data Strobe: DQS_t, DQS_c
CA	11	Encoded address and command
CKE	2	Clock Enable
CS_n	2	Chip Select
CK	2	Complementary Clock: CK_t, CK_c
Total	97	**Per Quadrant**

2.5.3 Channel Interface Signals (cont'd)

Table 5 — DQS and DMI Mapping

DQ Signal	DQS Mapping	DMI Mapping
DQ[7:0]	DQS[0]_t, DQS[0]_c	DMI[0]
DQ[15:8]	DQS[0]_t, DQS[0]_c	DMI[1]
DQ[23:16]	DQS[1]_t, DQS[1]_c	DMI[2]
DQ[31:24]	DQS[1]_t, DQS[1]_c	DMI[3]
DQ[39:32]	DQS[2]_t, DQS[2]_c	DMI[4]
DQ[47:40]	DQS[2]_t, DQS[2]_c	DMI[5]
DQ[55:48]	DQS[3]_t, DQS[3]_c	DMI[6]
DQ[63:56]	DQS[3]_t, DQS[3]_c	DMI[7]

NOTE 1 The table does not show quadrant and slice information. For example, the DQS true signal for DQ39, Channel 1, Quadrant C would be DQS1C2 (Micropillar Definition and Description).

2.5.4 CKE and CS Mapping

Table 6 — CKE and CS Mapping

Configuration	4x64	8x64
1 High P2P	CKE[0], CS[0] for Ch. 0	CKE[0], CS[0] for Ch. 0 CKE[1], CS[1] for Ch. 1
2 High P2P	CKE[0], CS[0] for Ch. 0 CKE[1], CS[1] for Ch. 1	NA
2 High P22P	NA	CKE[0], CS[0] for Ch. 0 of Rank 0 CKE[1], CS[1] for Ch. 1 of Rank 0 CKE[2], CS[2] for Ch. 0 of Rank 1 CKE[3], CS[3] for Ch. 1 of Rank 1
4 High P22P	CKE[0], CS[0] for Ch. 0 of Rank 0 CKE[1], CS[1] for Ch. 1 of Rank 0 CKE[2], CS[2] for Ch. 0 of Rank 1 CKE[3], CS[3] for Ch. 1 of Rank 1	NA

2.5.5 Test Sub-Block Micropillar Signals

The Test Sub-Block allocates 5 test pins per quadrant. Table 7 shows the signals contained in the Test Sub-Block and Figure 12 shows their micropillar assignments.

Table 7 — Test Sub-Block Micropillar Signals

Signal Name	Type	Number	Description
SEN[2:0]	Input	3	Scan chain enable
SSH_n	Input	1	Boundary scan chain shift
SDI	Input	1	Boundary scan data input
SCK	Input	1	Boundary scan clock
SCS[3:0]_n	Input	4	Boundary scan chip select
SDO[3:0]	Output	4	Boundary scan chain data out
TEST	Input	1	Test mode enable
NC	NA	1	No Connection
RST[3:0]_n	Input	4	Reset Slice (all ranks, quadrants, and channels)
Totals		20	

2.5.6 Bump Plan

The WideIO2 micro-bumps will be a 40um x 40um rectangular pattern as shown in Figure 7. The bumps are divided into 4 quadrants with signal assignments mirrored both horizontally and vertically with the exception of the Test Sub-Block of each quadrant. Figure 8 shows a high level quadrant bump plan. There will be a space between quadrants in the x-direction of 1000um and in the y-direction of 120um (Micropillar center-to-center). Note: Figure 8 is a bottom view looking down upon the memory micropillars, i.e., with the memory micropillars facing out of the page. In the anticipated mounting orientation, this will be looking up from the board.

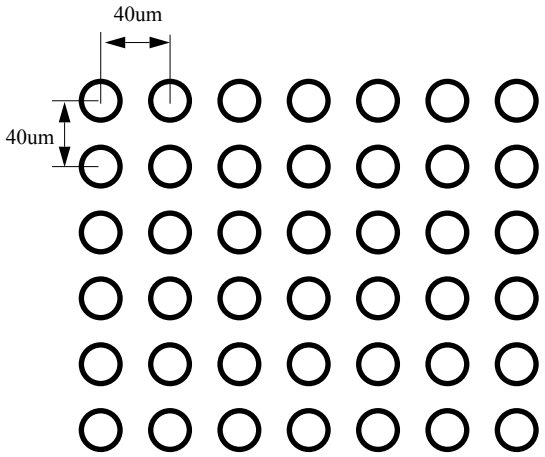
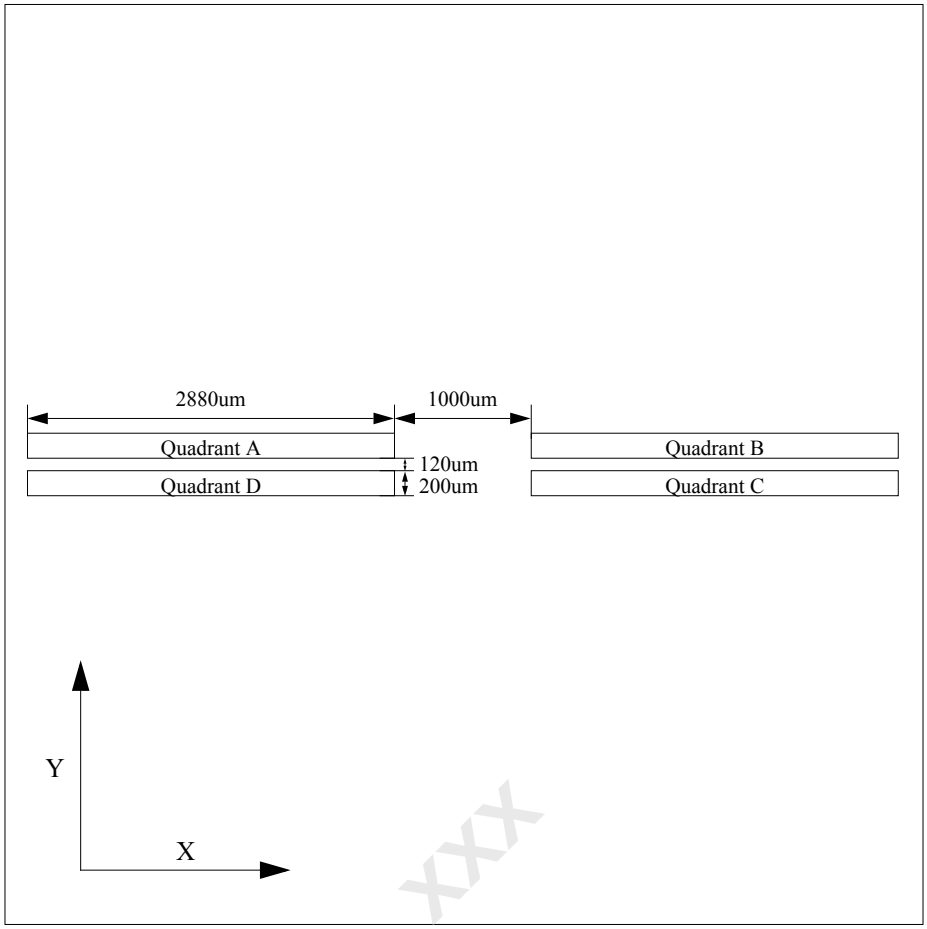


Figure 7 — Bump Pattern

2.5.6 Bump Plan (cont'd)



NOTE 1 Dimensions are micropillar center-to-center.

Figure 8 — Bottom View Quadrant Placement

2.6 TSV Signal Routing

Signal routing through the slices can be Swizzled, Staggered, or Through (see Figure 9). Swizzled routing is used as P2P for 1 or 2-high stack and P22P for 4-high stack. Staggered routing is used to deliver unique signals to each slice. To match capacitive loading, signals will continue up the stack regardless of where they are terminated. Through routing delivers common signals to all slices. Shaded micropillars indicate the active connections in each slice.

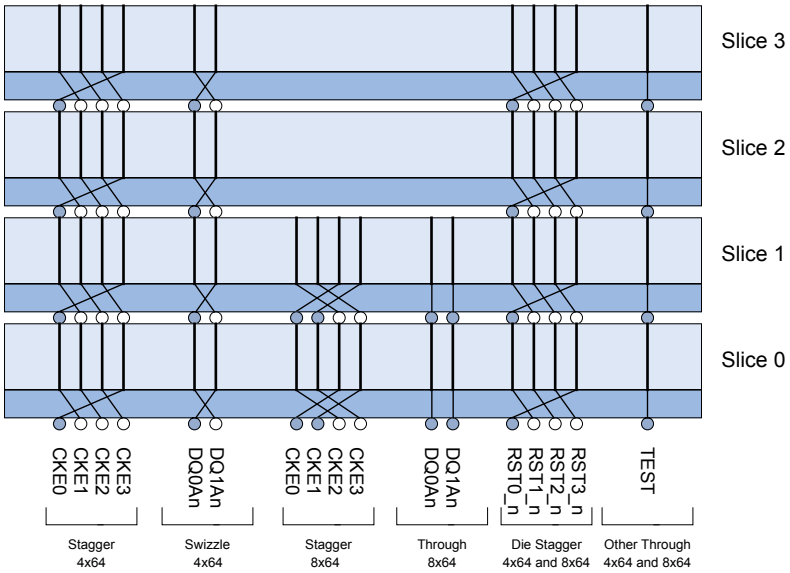


Figure 9 — TSV Routing Example

Table 8 — TSV Signal Routing for 4ch x64b

Signal Name	Routing	Signal Name	Routing
CK[1:0][D:A]_t, CK[1:0][D:A]_c	Swizzled	SSH_n	Through
CA[1:0][D:A][10:0]	Swizzled	SDI	Through
CS[D:A][3:0]_n	Staggered	SCK	Through
CKE[D:A][3:0]	Staggered	SCS[3:0]_n	Staggered
DQ[1:0][D:A][63:0]	Swizzled	SDO[3:0]	Staggered
DQS[1:0][D:A][3:0]_t, DQS[1:0][D:A][3:0]_c	Swizzled	TEST	Through
DMI[1:0][D:A][7:0]	Swizzled	DAA[D:A]	Through
RST[3:0]_n	Staggered	DA[D:A][9:0]	Through
SEN[2:0]	Through		

2.6 TSV Signal Routing (cont'd)

Table 9 — TSV Signal Routing for 8ch x64b

Signal Name	Routing	Signal Name	Routing
CK[1:0][D:A]_t, CK[1:0][D:A]_c	Through	SSH_n	Through
CA[1:0][D:A][10:0]	Through	SDI	Through
CS[D:A][3:0]_n	Staggered	SCK	Through
CKE[D:A][3:0]	Staggered	SCS[3:0]_n	Staggered
DQ[1:0][D:A][63:0]	Through	SDO[3:0]	Staggered
DQS[1:0][D:A][3:0]_t, DQS[1:0][D:A][3:0]_c	Through	TEST	Through
DMI[1:0][D:A][7:0]	Through	DAA[D:A]	Through
RST[3:0]_n	Staggered	DA[D:A][9:0]	Through
SEN[2:0]	Through		

2.6.1 Micropillar Locations

Each WideIO2 channel signal (e.g., DQ0) takes two micropillars; one for channel 0 and one for channel 1 (see Micropillar Definition and Description for signal naming methodology). These signals are swizzled to the slice(s) above as needed (see Table 8). The columns are 6 bumps high alternating between signal and power which allows area on both sides of signal columns for routing. Each quadrant contains one 14-column CA sub-block, one 56-column data sub-block, and one 3-column test sub-block (these sub-blocks column counts include the power supply columns).

Figure 10 shows the CA sub-block for quadrants A and D, Figure 11 shows the data sub-block for quadrant A, and Figure 12 shows the test sub-block for quadrants A, B, C, and D. Since quadrants are flipped images of each other with the only variation in test sub-block, all micropillar locations can be derived from these three figures.

NOTE 1 All views are the bottom views looking down upon the memory micropillars, i.e., with the memory micropillars facing out of the page. In the anticipated mounting orientation, this will be looking up from the board.

							Quadrant A									
VDD1			1	2	3	4	5	6	7	8	9	10	11	12	13	14
VDD2		um	0	40	80	120	160	200	240	280	320	360	400	440	480	520
VSS	M	520	VDD1	VSS	VSS	DAA8	VDD2	CA0A6	VSS	CA0A3	VDD2	CA0A0	VSS	CK0A_t	VDD2	CK0A_c
VDDQ	L	480	VDD1	VSS	VSS	CA0A9	VDD2	CA1A6	VSS	CA1A3	VDD2	CA1A0	VSS	CK1A_t	VDD2	CK1A_c
VSSQ	K	440	VDD1	VSS	VSS	CA1A9	VDD2	CA0A7	VSS	CA0A4	VDD2	CA0A1	VSS	CKEA0	VDD2	CSA0_n
DQ	J	400	VDD1	VSS	VSS	CA0A10	VDD2	CA1A7	VSS	CA1A4	VDD2	CA1A1	VSS	CKEA1	VDD2	CSA1_n
DMI	H	360	VDD1	VSS	VSS	CA1A10	VDD2	CA0A8	VSS	CA0A5	VDD2	CA0A2	VSS	CKEA2	VDD2	CSA2_n
DQS	G	320	VDD1	VSS	VSS	DAA9	VDD2	CA1A8	VSS	CA1A5	VDD2	CA1A2	VSS	CKEA3	VDD2	CSA3_n
CA		280														
CS		240														
CKE	F	200	VDD1	VSS	VSS	DAD9	VDD2	CA1D8	VSS	CA1D5	VDD2	CA1D2	VSS	CKED3	VDD2	CSD3_n
CK	E	160	VDD1	VSS	VSS	CA1D10	VDD2	CA0D8	VSS	CA0D5	VDD2	CA0D2	VSS	CKED2	VDD2	CSD2_n
Misc	D	120	VDD1	VSS	VSS	CA0D10	VDD2	CA1D7	VSS	CA1D4	VDD2	CA1D1	VSS	CKED1	VDD2	CSD1_n
DA	C	80	VDD1	VSS	VSS	CA1D9	VDD2	CA0D7	VSS	CA0D4	VDD2	CA0D1	VSS	CKED0	VDD2	CSD0_n
	B	40	VDD1	VSS	VSS	CA0D9	VDD2	CA1D6	VSS	CA1D3	VDD2	CA1D0	VSS	CK1D_t	VDD2	CK1D_c
	A	0	VDD1	VSS	VSS	DAD8	VDD2	CA0D6	VSS	CA0D3	VDD2	CA0D0	VSS	CK0D_t	VDD2	CK0D_c
		um	0	40	80	120	160	200	240	280	320	360	400	440	480	520
			1	2	3	4	5	6	7	8	9	10	11	12	13	14
								Quadrant D								

Figure 10 — Quadrant A and D CA Sub-Block Micropillar Locations

2.6.1 Micropillar Locations (cont'd)

15	16	17	18	19	20	21	22	23	24	25	26	27	28
560	600	640	680	720	760	800	840	880	920	960	1000	1040	1080
VSS	DQ0A61	VDDQ	DMI0A7	VSSQ	DQ0A56	VDD2	DAA6	VSS	DQ0A53	VDDQ	DMI0A6	VSSQ	DQ0A48
VSS	DQ1A61	VDDQ	DMI1A7	VSSQ	DQ1A56	VDD2	DQS0A3_t	VSS	DQ1A53	VDDQ	DMI1A6	VSSQ	DQ1A48
VSS	DQ0A62	VDDQ	DQ0A59	VSSQ	DQ0A57	VDD2	DQS1A3_t	VSS	DQ0A54	VDDQ	DQ0A51	VSSQ	DQ0A49
VSS	DQ1A62	VDDQ	DQ1A59	VSSQ	DQ1A57	VDD2	DQS0A3_c	VSS	DQ1A54	VDDQ	DQ1A51	VSSQ	DQ1A49
VSS	DQ0A63	VDDQ	DQ0A60	VSSQ	DQ0A58	VDD2	DQS1A3_c	VSS	DQ0A55	VDDQ	DQ0A52	VSSQ	DQ0A50
VSS	DQ1A63	VDDQ	DQ1A60	VSSQ	DQ1A58	VDD2	DAA7	VSS	DQ1A55	VDDQ	DQ1A52	VSSQ	DQ1A50

29	30	31	32	33	34	35	36	37	38	39	40	41	42
1120	1160	1200	1240	1280	1320	1360	1400	1440	1480	1520	1560	1600	1640
VDD2	DQ0A45	VSS	DMI0A5	VDDQ	DQ0A40	VSSQ	DAA4	VDD2	DQ0A37	VSS	DMI0A4	VDDQ	DQ0A32
VDD2	DQ1A45	VSS	DMI1A5	VDDQ	DQ1A40	VSSQ	DQS0A2_t	VDD2	DQ1A37	VSS	DMI1A4	VDDQ	DQ1A32
VDD2	DQ0A46	VSS	DQ0A43	VDDQ	DQ0A41	VSSQ	DQS1A2_t	VDD2	DQ0A38	VSS	DQ0A35	VDDQ	DA0A33
VDD2	DQ1A46	VSS	DQ1A43	VDDQ	DQ1A41	VSSQ	DQS0A2_c	VDD2	DQ1A38	VSS	DQ1A35	VDDQ	DQ1A33
VDD2	DQ0A47	VSS	DQ0A44	VDDQ	DQ0A42	VSSQ	DQS1A2_c	VDD2	DQ0A39	VSS	DQ0A36	VDDQ	DQ0A34
VDD2	DQ1A47	VSS	DQ1A44	VDDQ	DQ1A42	VSSQ	DAA5	VDD2	DQ1A39	VSS	DQ1A36	VDDQ	DQ1A34

43	44	45	46	47	48	49	50	51	52	53	54	55	56
1680	1720	1760	1800	1840	1880	1920	1960	2000	2040	2080	2120	2160	2200
VSSQ	DQ0A29	VDD2	DMI0A3	VSS	DQ0A24	VDDQ	DAA2	VSSQ	DQ0A21	VDD2	DMI0A2	VSS	DQ0A16
VSSQ	DQ1A29	VDD2	DMI1A3	VSS	DQ1A24	VDDQ	DQS0A1_t	VSSQ	DQ1A21	VDD2	DMI1A2	VSS	DQ1A16
VSSQ	DQ0A30	VDD2	DQ0A27	VSS	DQ0A25	VDDQ	DQS1A1_t	VSSQ	DQ0A22	VDD2	DQ0A19	VSS	DQ0A17
VSSQ	DQ1A30	VDD2	DQ1A27	VSS	DQ1A25	VDDQ	DQS0A1_c	VSSQ	DQ1A22	VDD2	DQ1A19	VSS	DQ1A17
VSSQ	DQ0A31	VDD2	DQ0A28	VSS	DQ0A26	VDDQ	DQS1A1_c	VSSQ	DQ0A23	VDD2	DQ0A20	VSS	DQ0A18
VSSQ	DQ1A31	VDD2	DQ1A28	VSS	DQ1A26	VDDQ	DAA3	VSSQ	DQ1A23	VDD2	DQ1A20	VSS	DQ1A18

57	58	59	60	61	62	63	64	65	66	67	68	69	70
2240	2280	2320	2360	2400	2440	2480	2520	2560	2600	2640	2680	2720	2760
VDDQ	DQ0A13	VSSQ	DMI0A1	VDD2	DQ0A8	VSS	DAA0	VDDQ	DQ0A5	VSSQ	DMI0A0	VDD2	DQ0A0
VDDQ	DQ1A13	VSSQ	DMI1A1	VDD2	DQ1A8	VSS	DQS0A0_t	VDDQ	DQ1A5	VSSQ	DMI1A0	VDD2	DQ1A0
VDDQ	DQ0A14	VSSQ	DQ0A11	VDD2	DQ0A9	VSS	DQS1A0_t	VDDQ	DQ0A6	VSSQ	DQ0A3	VDD2	DQ0A1
VDDQ	DQ1A14	VSSQ	DQ1A11	VDD2	DQ1A9	VSS	DQS0A0_c	VDDQ	DQ1A6	VSSQ	DQ1A3	VDD2	DQ1A1
VDDQ	DQ0A15	VSSQ	DQ0A12	VDD2	DQ0A10	VSS	DQS1A0_c	VDDQ	DQ0A7	VSSQ	DQ0A4	VDD2	DQ0A2
VDDQ	DQ1A15	VSSQ	DQ1A12	VDD2	DQ1A10	VSS	DAA1	VDDQ	DQ1A7	VSSQ	DQ1A4	VDD2	DQ1A2

Figure 11 — Quadrant A Data Sub-Block Micropillar Locations

2.6.1 Micropillar Locations (cont'd)

Quadrant A				Quadrant B			
71	72	73			74	75	76
2800	2840	2880			3880	3920	3960
VSS	DAAA	VDD1			VDD1	DAAB	VSS
VSS	SDI	VDD1			VDD1	SDO0	VSS
VSS	SCK	VDD1			VDD1	SDO1	VSS
VSS	SSH_n	VDD1			VDD1	SDO2	VSS
VSS	SEN0	VDD1			VDD1	SDO3	VSS
VSS	SEN1	VDD1			VDD1	NC	VSS
VSS	SEN2	VDD1			VDD1	TEST	VSS
VSS	SCS3_n	VDD1			VDD1	RST3_n	VSS
VSS	SCS2_n	VDD1			VDD1	RST2_n	VSS
VSS	SCS1_n	VDD1			VDD1	RST1_n	VSS
VSS	SCS0_n	VDD1			VDD1	RST0_n	VSS
VSS	DAAD	VDD1			VDD1	DAAC	VSS
2800	2840	2880			3880	3920	3960
71	72	73			74	75	76
Quadrant D				Quadrant C			

Figure 12 — Quadrants A, B, C, and D Test Sub-Block Micropillar Locations

2.7 Addressing

2.7.1 Addressing for 4x64 Die

Table 10 shows WideIO2 addressing for 4x64 die supported slice densities.

Table 10 — Address Table (4x64 Die)

Parameter	Density					Notes
	8Gb	12Gb	16Gb	24Gb	32Gb	
Prefetch Size	256b	256b	256b	256b	256b	
Row Address	RA[12:0]	RA[13:0] 3)	RA[13:0]	RA[14:0] 4)	RA[14:0]	
Column Address	CA[8:0]	CA[8:0]	CA[8:0]	CA[8:0]	CA[8:0]	1,2
Bank Address	BA[2:0]	BA[2:0]	BA[2:0]	BA[2:0]	BA[2:0]	
Page Size	4KB	4KB	4KB	4KB	4KB	

NOTE 1 The least-significant column address C0 is not transmitted on the CA bus and is implied to be zero.
NOTE 2 CA[1] is used to drive 16B critical data first in burst order for BL4 transfers. CA[2:1] are used to drive 16B critical data first in burst order for BL8.
NOTE 3 No memory present at addresses with R12=R13=HIGH. ACT command with R12=R13=HIGH is ignored (NOP).Write to R12=R13=HIGH is ignored (NOP).
NOTE 4 No memory present at addresses with R13=R14=HIGH. ACT command with R13=R14=HIGH is ignored (NOP). Write to R13=R14=HIGH is ignored (NOP).

2.7.2 Addressing for 8x64 Die

Table 11 shows WideIO2 addressing for 8x64 supported slice densities.

Table 11 — Address Table (8x64 Die)

Parameter	Density					Notes
	8Gb	12Gb	16Gb	24Gb	32Gb	
Prefetch Size	256b	256b	256b	256b	256b	
Row Address	RA[13:0]	RA[14:0] 3)	RA[14:0]	RA[15:0] 4)	RA[15:0]	
Column Address	CA[7:0]	CA[7:0]	CA[7:0]	CA[7:0]	CA[7:0]	1,2
Bank Address	BA[1:0]	BA[1:0]	BA[1:0]	BA[1:0]	BA[1:0]	
Page Size	2KB	2KB	2KB	2KB	2KB	

NOTE The least-significant column address C0 is not transmitted on the CA bus and is implied to be zero.
NOTE 2 CA[1] is used to drive 16B critical data first in burst order for BL4 transfers. CA[2:1] are used to drive 16B critical data first in burst order for BL8.
NOTE 3 No memory present at addresses with R13=R14=HIGH. ACT command with R13=R14=HIGH is ignored (NOP).Write to R13=R14=HIGH is ignored (NOP).
NOTE 4 No memory present at addresses with R14=R15=HIGH. ACT command with R14=R15=HIGH is ignored (NOP). Write to R14=R15=HIGH is ignored (NOP).

3 Functional Descriptions

3.1 State Diagram

WideIO2 state diagram provides a simplified illustration of allowed state transitions and the related commands to control them. Unless otherwise specified, Figure 13 describes the state and commands for only one channel in a slice.

For a complete definition of the device behavior, the information provided in Figure 13 should be integrated with the truth tables and timing specification.

Table 20 provides complementary information to Figure 13, they clarify the device behavior and the applied restrictions when considering the actual state of all the banks.

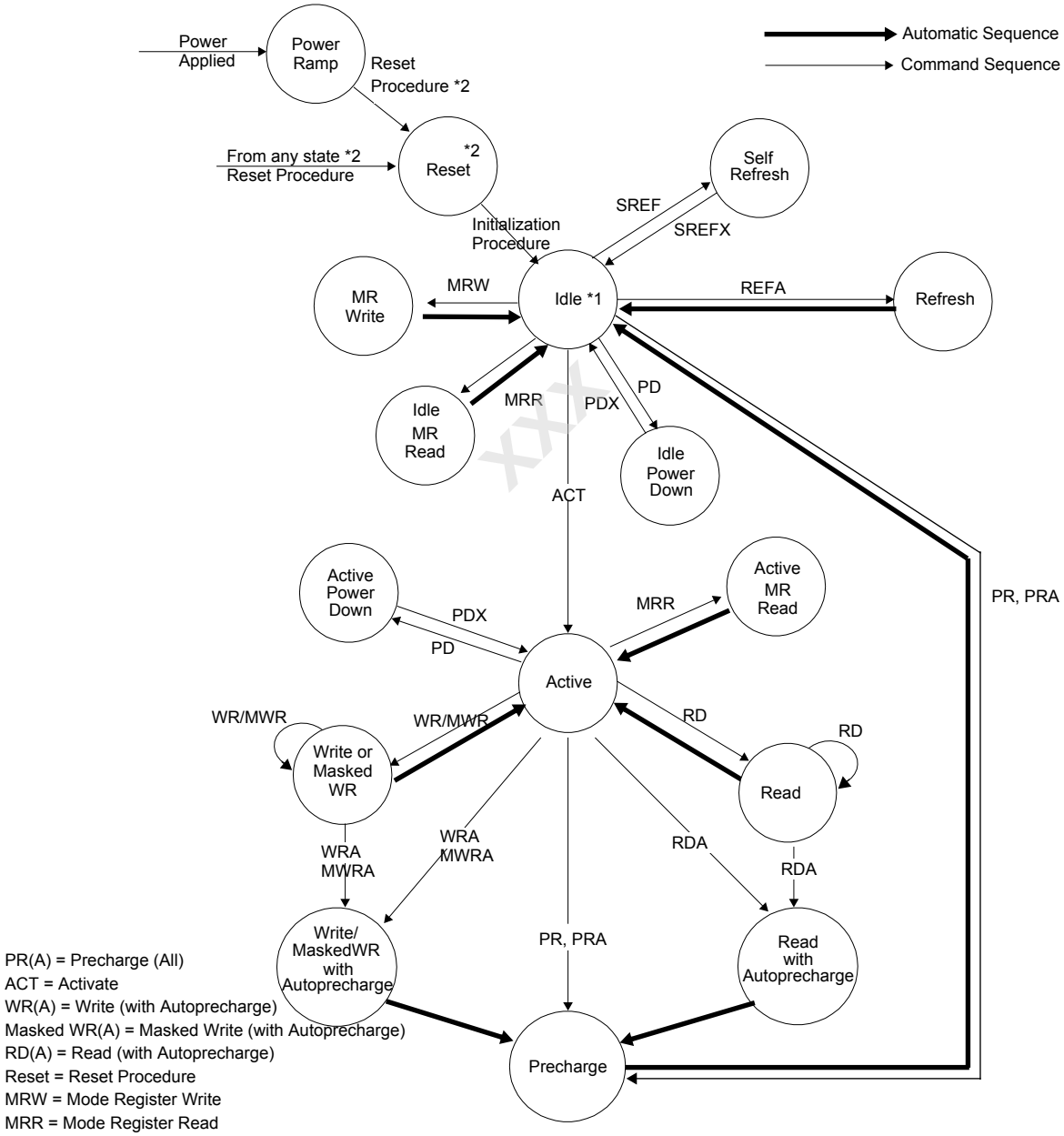


Figure 13 — WideIO2 Device Simplified State Diagram

3.2 Power-up, Initialization, and Power-off

3.2.1 Voltage Ramp and Device Initialization

The following sequence must be used to power up the device. Unless specified otherwise, this procedure is mandatory to avoid permanent damage to the WideIO2 device.

1. Voltage Ramp: While applying power, RST_n is recommended to be maintained below 0.2 x VDD2; all other inputs shall be between VILmin and VIHmax. Voltage ramp power supply requirements are provided in Table 12.

Table 12 — Voltage Ramp Conditions

After...	Applicable Conditions
Ta is reached	V_{DD1} must be greater than V_{DD2}—200mV
	V_{DD1} and V_{DD2} must be greater than V_{DDQ}—200mV

NOTE 1 Ta is the point when any power supply first reaches 300mV.
NOTE 2 Noted conditions apply between Ta and power-off (controlled or uncontrolled).
NOTE 3 Tb is the point at which all supply and reference voltages are within their defined operating ranges.
NOTE 4 Power ramp duration t_{INIT0} (Tb - Ta) must not exceed 20ms.
NOTE 5 The voltage difference between any of $V_{SS}$and V_{SSQ} pins must not exceed 100mV.
NOTE 6 The Wide IO2 DRAM device will guarantee that outputs are in a high impedance state while RST_n is held low.

2. RST_n: RST_n must be maintained for a minimum 200 us with stable power. CKE must be pulled “Low” a minimum of 10ns before RST_n is de-asserted.

3. CKE and Clock: After RST_n is de-asserted, CKE must remain deasserted for 2 ms. During this time, the DRAM will initialize internal state independent of external clocks. The clock must be started and stable for at least 5 clocks before CKE goes active. Since CKE is a synchronous signal, the corresponding set up time to clock (tIS) must also be met. Also, a NOP or Deselect command must be registered (with tIS set up time to clock) the clock before CKE goes active. After CKE is registered high, tINIT5 must pass before issuing the MRS command to load a mode register.

4. MR: Issue MRS Command(s) to load mode registers with all application settings.

5. **Normal Operation:** The Wide IO2 DRAM device is now ready for normal operation.

3.2.1 Voltage Ramp and Device Initialization (cont'd)

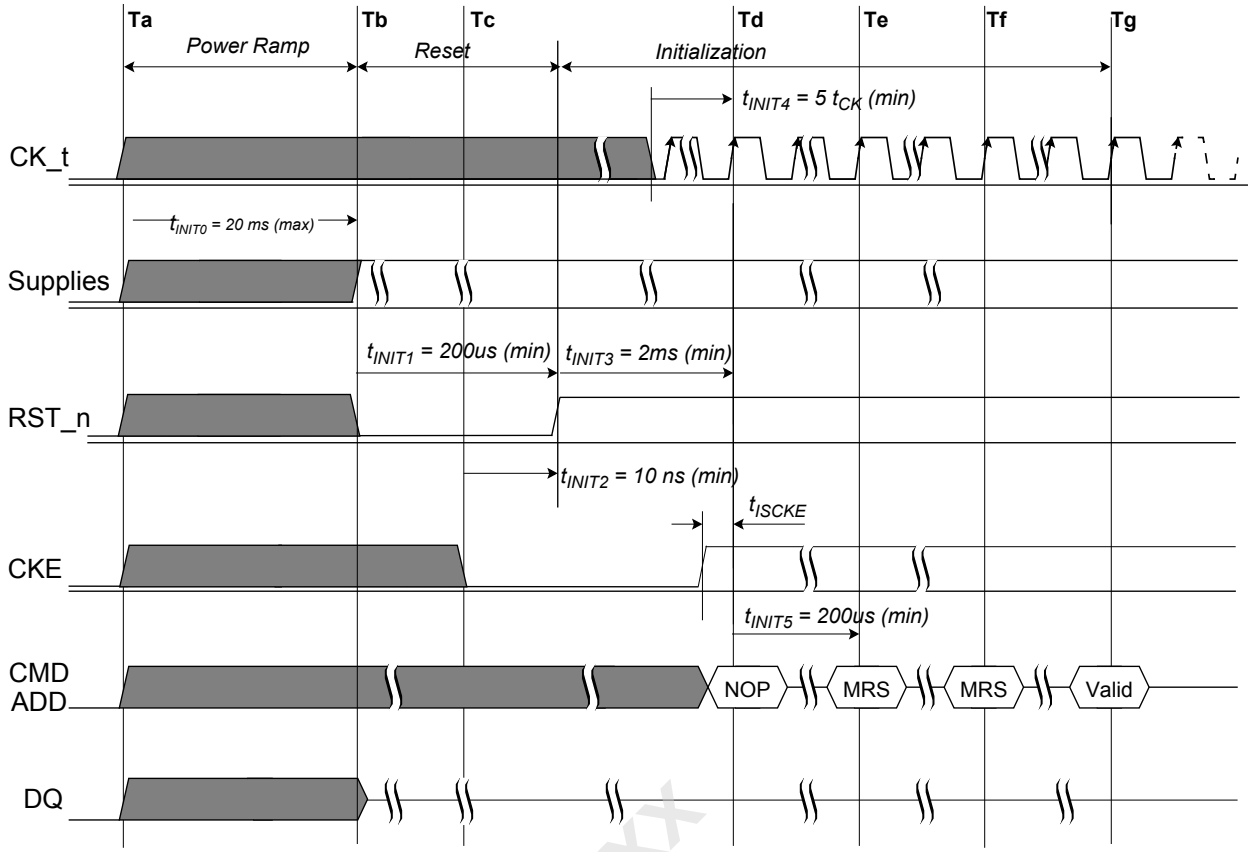


Figure 14 — Voltage Ramp and Initialization Sequence

NOTE 1 High-Z on the CA bus indicates NOP.
NOTE 2 For t_{INIT} values, see Table 13.

Table 13 — Initialization Timing Parameters

Parameter	Value		Unit	Comment
	Min	Max		
t_{INIT0}		20	ms	Maximum Power Ramp Time
t_{INIT1}	200		us	Minimum RST_n low Time with stable power
t_{INIT2}	10		ns	Minimum CKE low Time before RST_n high
t_{INIT3}	2		ms	Minimum CKE low Time after RST_n high
t_{INIT4}	5		tCK	Minimum stable clock before first CKE high
t_{INIT5}	200		us	Minimum idle time after first CKE assertion

3.2.2 Reset Initialization with stable power

The following sequence is required for RESET with no power interruption initialization.

1. Assert RST_n (active low) anytime when reset is needed. All other inputs may be undefined.
RST_n must be maintained for at least 100 ns. CKE must be pulled “LOW” 10ns before RST_n is deasserted.

2. Follow the Power-up and Initialization Sequence steps 3 and 4.

3. The Reset sequence is now completed; The Wide IO2 DRAM device is ready for normal operation.

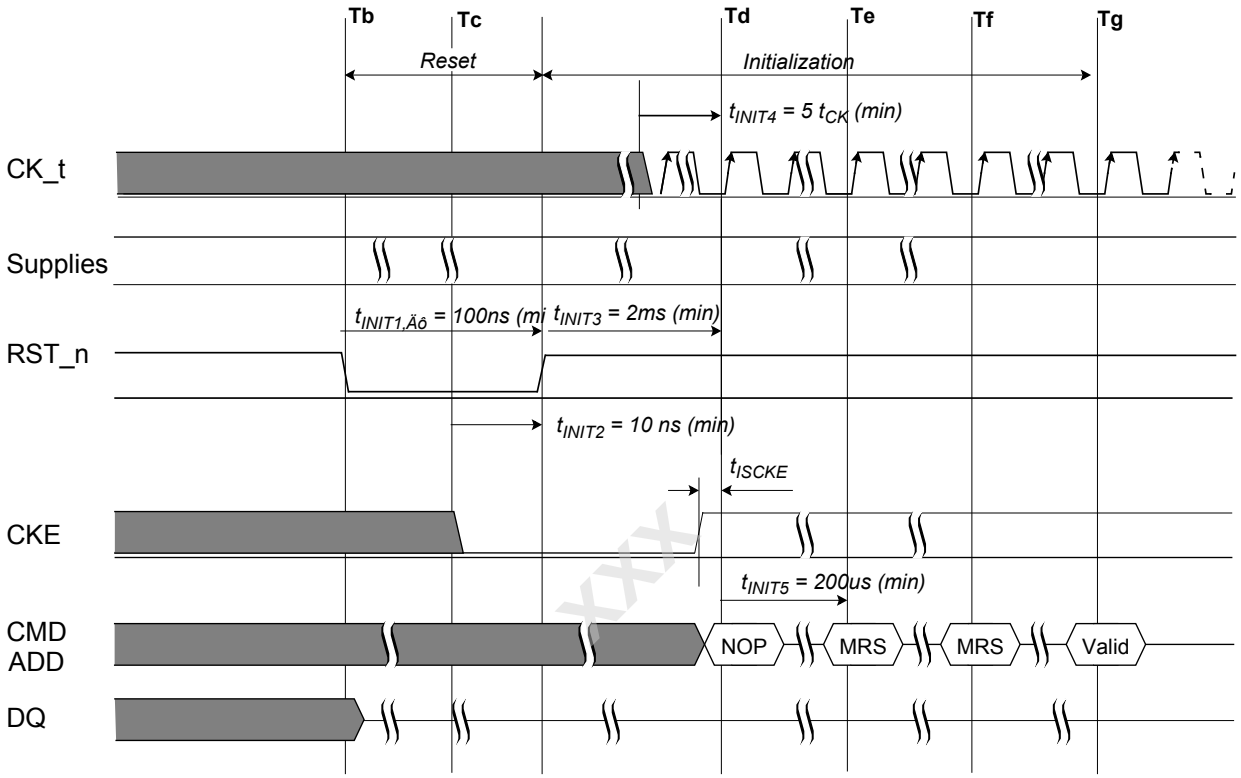


Figure 15 — Reset Initialization with stable Power

Table 14 — Timing Parameters for Reset Initialization with stable Power

Symbol	Value		Unit	Comment
	min	max		
t_{INIT1}	100		ns	Minimum RST_n low Time for Reset Initialization with stable power
t_{INIT2}	10		ns	Minimum CKE low Time before RST_n high
t_{INIT3}	2		ms	Minimum CKE low Time after RST_n high
t_{INIT4}	5		tCK	Minimum stable clock before first CKE high
t_{INIT5}	200		us	Minimum idle time after first CKE assertion

3.2.3 Power-off Sequence

The following sequence shall be used to power off the Wide IO2 DRAM device. Unless specified otherwise, these steps are mandatory.

If system requires DQ remain Hi-Z while removing power, RST_n shall be held at a logic low level (=< 0.2 x VDD2) or CKE shall be held in logic low level. All other inputs shall be between VILmin and VIHmax. The Wide IO2 DRAM device will guarantee that outputs are in a high impedance state while RST_n is held low.

Tx is the point where any power supply decreases under its minimum value specified in Table 42, DC operating conditions.

Tz is the point where all power supplies are below 300 mV. After Tz, the device is powered off.
The time between Tx and Tz (tPOFF) shall be less than 2s.

Table 15 — Power Supply Conditions

Between...	Applicable Conditions
Tx and Tz	V_{DD1} must be greater than V_{DD2}—200mV
Tx and Tz	V_{DD1} and V_{DD2} must be greater than V_{DDQ}—200mV

The voltage difference between VSS and VSSQ micropillars may not exceed 100 mV.

3.2.3 Power-off Sequence (cont'd)

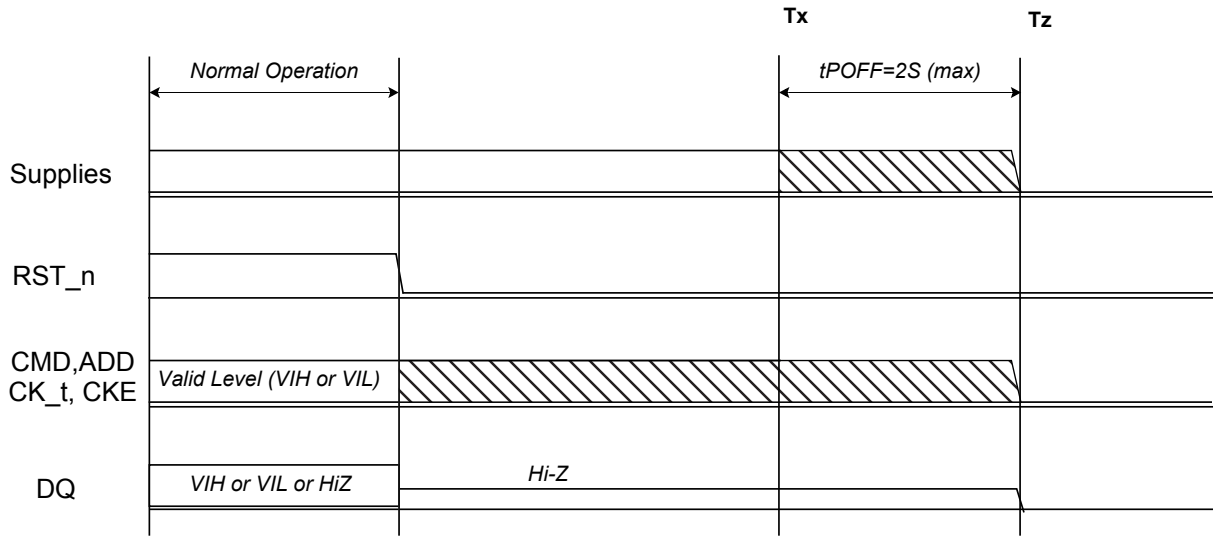


Figure 16 — Power-Off Sequence with RST_n

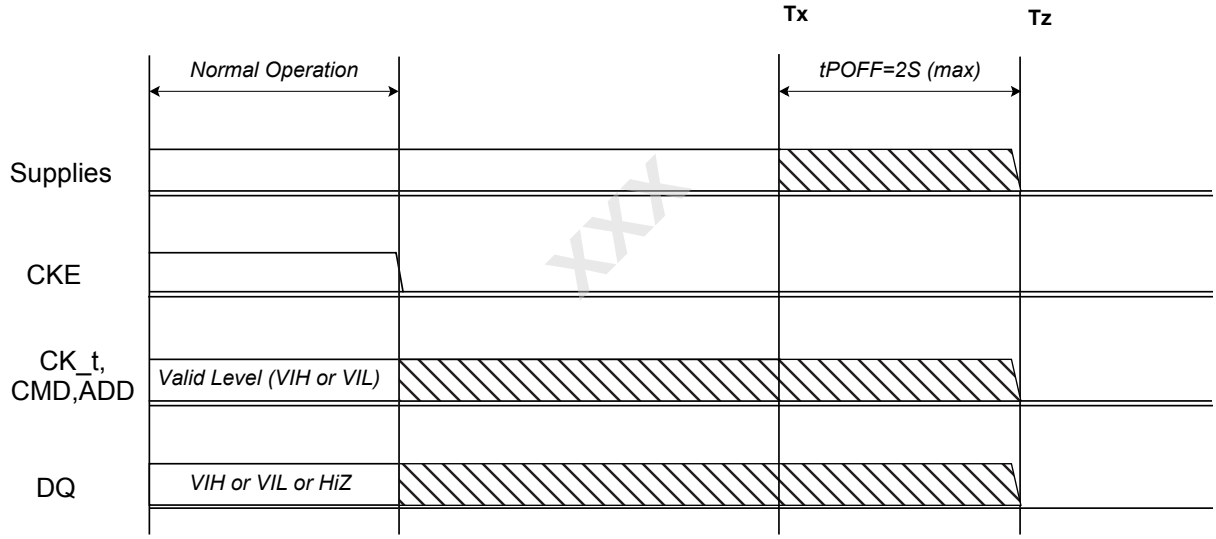


Figure 17 — Power-Off Sequence with CKE

3.2.4 Uncontrolled Power-Off Sequence

When an uncontrolled power-off occurs, the following conditions must be met:

At Tx, when the power supply drops below the minimum values specified, all power supplies must be turned off and all power-supply current capacity must be at zero, except for any static charge remaining in the system.

After Tz (the point at which all power supplies first reach 300mV), the device must power off. During this period, the relative voltage between power supplies is uncontrolled. V_{DD1} and V_{DD2} must decrease with a slope lower than 0.5 V/µs between Tx and Tz.

An uncontrolled power-off sequence can occur a maximum of 400 times over the life of the device.

Table 16 — Timing Parameters Power-Off

Symbol	Value		Unit	Comment
	min	max		
t_{POFF}	-	2	s	Maximum Power-Off ramp time

3.3 Input Clock Stop and Frequency Change

WideIO2 devices support input clock frequency change during CKE LOW under the following conditions:
. tCK(MIN) and tCK(MAX) are met for each clock cycle;
. Refresh Requirements apply during clock frequency change;
. During clock frequency change, only REF commands may be executing;
. Any Activate or Precharge commands have executed to completion prior to changing the frequency;
. The related timing conditions (tRCD, tRP) have been met prior to changing the frequency;
. The initial clock frequency shall be maintained for a minimum of 2 clock cycles after CKE goes LOW;
. The clock satisfies tCH(abs) and tCL(abs) for a minimum of 2 clock cycles prior to CKE going HIGH.

After the input clock frequency is changed and CKE is held HIGH, additional MRS commands may be required to set the WR, RL, etc. These settings may need to be adjusted to meet minimum timing requirements at the target clock frequency.

WideIO2 devices support clock stop during CKE LOW under the following conditions:
. CK_t is held LOW and CK_c is held HIGH during clock stop;
. Refresh Requirements apply during clock stop;
. During clock stop, only REF commands may be executing;
. Any Activate or Precharge commands have executed to completion prior to stopping the clock;
. The related timing conditions (tRCD, tRP) have been met prior to stopping the clock;
. The initial clock frequency shall be maintained for a minimum of 2 clock cycles after CKE goes LOW;
. The clock satisfies tCH(abs) and tCL(abs) for a minimum of 2 clock cycles prior to CKE going HIGH.

WideIO2 devices support input clock frequency change during CKE HIGH under the following conditions:
. tCK(MIN) and tCK(MAX) are met for each clock cycle;
. Refresh Requirements apply during clock frequency change;
. Any Activate, Read, Write, Precharge, Mode Register Set, or Status Register Read commands must have executed to completion, including any associated data bursts prior to changing the frequency;
. The related timing conditions (tRCD, tWR, tRP, tMRD, tSRR, etc.) have been met prior to changing the frequency;
. CS_n shall be held HIGH during clock frequency change;
. During clock frequency change, only REF commands may be executing;
. The WideIO2 device is ready for normal operation after the clock satisfies tCH(abs) and tCL(abs) for a minimum of 2tCK + tXP.

After the input clock frequency is changed, additional MRS commands may be required to set the WR, RL, etc. These settings may need to be adjusted to meet minimum timing requirements at the target clock frequency.

WideIO2 devices support clock stop during CKE HIGH under the following conditions:
. CK_t is held LOW and CK_c is held HIGH during clock stop;
. CS_n shall be held HIGH during clock stop;
. Refresh Requirements apply during clock stop;
. During clock stop, only REF commands may be executing;
. Any Activate, Read, Write, Precharge, Mode Register Write, or Mode Register Read commands must have executed to completion, including any associated data bursts prior to stopping the clock;
. The related timing conditions (tRCD, tWR, tRP, tMRD, tSRR, etc.) have been met prior to stopping the clock;

The WideIO2 device is ready for normal operation after the clock is restarted and satisfies tCH(abs) and tCL(abs) for a minimum of 2tCK + tXP.

3.4 Mode Register Definition

For application flexibility, various functions, features, and modes are programmable in WideIO2 Mode Registers. Values are programmed into Mode Registers via a Mode Register Write (MRW) command. The default values of the Mode Registers (MR#) are defined so they must be programmed during the reset and initialization sequence. The Mode Registers are defined for one channel.

The contents of the Mode Registers can be altered by re-executing the MRW command during normal operation. When programming the Mode Registers, even if the user chooses to modify only a sub-set of the MRW fields, all address fields within the accessed Mode Register will be redefined when the MRW command is issued.

MRW commands do not affect array contents regardless of when these commands are executed.

After the Mode Register write command, tMRD is required to complete the write operation to the Mode Register. tMRW is the minimum time required between two MRW commands.

Table 17 shows the mode registers for WideIO2 SDRAM. Each register is denoted as “R” if it can be read but not written, “W” if it can be written but not read, and “R/W” if it can be read and written. A Mode Register Read command is used to read a mode register. A Mode Register Write command is used to write a mode register.

Table 17 — Mode Register Assignment in WideIO2 SDRAM

	Function	Access	OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
0	(Reserved)	(RFU)	(RFU)							
1	Device Feature-1	W	nWR			RRO	Programmable postamble	BL		
2	Device Feature-2	W	SRA	PPRE	WL			RL & nRTP		
3	I/O Config-1	W	DBI WE	DBI RE	Thermal offset		DME	DS		
4	Refresh Rate	R	TUF	(RFU)			Refresh Mode	SDRAM Refresh Rate		
5	Basic Config-1	R	Wide IO2 manufacturer ID							
6	Basic Config-2	R	Revision ID1							
7	Basic Config-3	R	Revision ID2							
8	Basic Config-4	R	(RFU)	IO Width / channel		Density / die			Channel / die	
9	Test Mode	W	Test Mode							
10	(Reserved)	(RFU)	(RFU)							
11	PASR-1	W	Bank Masking							
12	PASR-2	W	Segment Masking							
13	PPR Resources	R	Post Package Repair							
14	TRR-1	W	TRR Mode	TRR Mode - Bank			(RFU)			
15	TRR-2	R	(RFU)				Mac Cap.	Mac Value		

NOTE 1 RFU bits shall be set to ‘0’ during mode register writes.

NOTE 2 RFU bits shall be read as ‘0’ during mode register reads.

NOTE 3 All mode registers that are specified as RFU or write-only shall return undefined data when read and DQS_t, DQS_c shall be toggled.

NOTE 4 All mode registers that are specified as RFU shall not be written.

NOTE 5 See vendor device datasheets for details on vendor specific mode registers

NOTE 6 Writes to read-only registers shall have no impact on the functionality of the device.

NOTE 7 Mode Register Read data OP0 is delivered on DQ0 and so forth.

3.4.1 MR0_Reserved (MA<7:0> = 00_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
(RFU)							

3.4.2 MR1_Device Feature 1 (MA<7:0> = 01_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
nWR (for AP)			RRO	Programmable Postamble	BL		

BL	Write-only	OP<2:0>	**010_B:** BL4 (default) **011_B:** BL8 **111_B:** BL4 or BL8 (on the fly enabled) **All others:** Reserved	
Programmable Read Post-amble		OP<3>	**0_B:** Standard Post-amble (default) **1_B:** Extended Post-amble	
RRO (Refresh Rate Option)		OP<4>	**0_B** : Disable codes 001 and 010 in MR4 OP[2:0] (default) **1_B** : Enable all codes in MR4 OP[2:0]	1,2,3
nWR		OP<7:5>	**010_B:** nWR=6 (default) **011_B:** nWR=7 **100_B:** nWR=8 **101_B:** nWR=10 **110_B:** nWR=11 **All others:** Reserved	4

NOTE 1 MR1 OP4 RRO bit is valid only when MR4 OP3 = 1. For WideIO2 devices with MR4 OP3 = 0, MR4 OP[2:0] bits are not dependent on MR1 OP4.
NOTE 2 When OP[4] = 0, only 001b and 010b in MR4 OP[2:0] are disabled. WideIO2 devices must report 011b instead of 001b or 010b in this case.
NOTE 3 TCSR function does not depend on RRO setting.
NOTE 4 Programmed value in nWR register determines when to start internal precharge operation for a write burst with AP (nWR).

Table 18 — Burst Sequence

C2	C1	C0	BT	BL	Burst Cycle Number and Burst Address Sequence							
					1	2	3	4	5	6	7	8
X	0_B	0_B	seq	4	0	1	2	3				
X	1_B	0_B			2	3	0	1				
0_B	0_B	0_B	seq	8	0	1	2	3	4	5	6	7
0_B	1_B	0_B			2	3	0	1	6	7	4	5
1_B	0_B	0_B			4	5	6	7	0	1	2	3
1_B	1_B	0_B			6	7	4	5	2	3	0	1

NOTE 1 C0 input is not present on CA bus. It is implied zero.
NOTE 2 For BL=4, the burst address represents C1 - C0.
NOTE 3 For BL=8, the burst address represents C2 - C0.

3.4.3 MR2_Device Feature 2 (MA<7:0> = 02_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
SRA	PPRE	WL			RL & nRTP		

RL & nRTP	Write-only	OP<2:0>	DBI disable (MR3 OP[6] = 0B) **000_B**: RL=5, nRTP=2 (default) **001_B**: RL=6, nRTP=3 **010_B**: RL=7, nRTP=3 **011_B**: RL=8, nRTP=4 **100_B**: RL=9, nRTP=4 **All others:** reserved DBI enable (MR3 OP[6] = 1B) **000_B**: RL=6, nRTP=2 **001_B**: RL=7, nRTP=3 **010_B**: RL=8, nRTP=3 **011_B**: RL=9, nRTP=4 **100_B**: RL=10, nRTP=4 **All others:** reserved
WL		OP<5:3>	**000_B:** WL=3 (default) **001_B:** WL=4 **010_B:** WL=5 **011_B:** WL=6 **100_B**: WL=7 **All others:** reserved
PPRE (Post Package Repair Entry/Exit)		OP<6>	**0_B:** PPR mode exit (default) **1_B:** PPR mode entry
SRA (Self Refresh Abort)		OP<7>	**0_B:** Self-Refresh Abort disabled (default) **1_B:** Self-Refresh Abort enabled

NOTE 1 Self Refresh Abort is only implemented for 12Gb densities and above.

Read Latency		Write Latency	nWR	nRTP	Lower Frequency limit (Greater than)	Upper Frequency Limit (Same or less than)	Unit
No DBI	W/ DBI						
5	6	3	6	2	10	266	MHz
6	7	4	7	3	266	333	
7	8	5	8	3	333	400	
8	9	6	10	4	400	466	
9	10	7	11	4	466	533	

NOTE 1 The Wide I/O2 SDRAM device should not be operated at a frequency below the Lower-Frequency Limit shown for each RL, WL, or nWR value.
NOTE 2 The Wide I/O2 SDRAM device should not be operated at a frequency above the Upper Frequency Limit shown for each RL, WL, or nWR value.

3.4.4 MR3_I/O Configuration 1 (MA<7:0> = 03_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
DBI WE	DBI RE	Thermal Offset		DME	DS		

DS (Driver Strength)	Write-only	OP<2:0>	000_B : Weak **001_B** : Strong (default) **All others :** Reserved
DME		OP<3>	0_B : Data Mask Disable 1_B : Data Mask Enable (default)
Thermal Offset		OP<5:4>	**00_B**: No offset, 0~5°C gradient (default) **01_B**: 5 °C offset, 5~10°C gradient **10_B**: 10°C offset, 10~15°C gradient **11_B:** Reserved
DBI RE		OP<6>	**0_B** : Data Bus Inversion on Reads Disable (default) **1_B**: Data Bus Inversion on Reads Enable
DBI WE		OP<7>	**0_B**: Data Bus Inversion on Writes Disable (default) **1_B**: Data Bus Inversion on Writes Enable

3.4.4.1 Thermal Offset

Because of their tight thermal coupling with WideIO2 DRAMs, hot spots on WideIO2 controllers can induce thermal gradients into the DRAMs. As these hot spots may not be located near the DRAM thermal sensor, the DRAM's temperature compensated selfrefresh circuit may not generate enough refresh cycles to guarantee memory retention. To address this shortcoming, the controller can provide a thermal offset that the memory uses to adjust its TCSR circuit to ensure reliable operation.

This offset is provided through MR3 in channel A0 and it will not be affected through MR3 in the other channels. This temperature offset will modify refresh behavior for all channels in the slice. If the induced thermal gradient from controller's corresponding location for temperature sensor location of DRAM to hot spot location of controller is larger than 15 degrees C, then self-refresh mode will not reliably maintain memory contents.

During normal operation, the controller will update this register in channel A0 whenever it detects changes in the induced thermal gradient. Updating this register may modify the self-refresh behavior for all channels, including channels in self-refresh operation. It will take max tRRU(Refresh Rate Update) to change self-refresh period and have the change reflected in MR4.

To accurately determine the controller die's temperature gradient between the memory thermal sensor and the induced hot spot, the memory thermal sensor location must be provided to the WideIO2 memory controller.

To ensure that the memory thermal sensor is located above an X-Y location on the controller, the memory thermal sensor must be located within a rectangle centered on the memory-controller physical interface. Additionally, the minimum size of a controller that wants to use this feature is within that rectangle as otherwise the memory's thermal sensor could be located outside the controller die's perimeter.

3.4.5 MR4_Device Temperature (MA<7:0> = 04_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
TUF	(RFU)			Refresh mode	SDRAM Refresh Rate		

SDRAM Refresh Rate	Read-only	OP<2:0>	**000_B**: SDRAM Low temperature operating limit exceeded **001_B**: 4x t_{REFI}, 4x t_{REFIpb}, 4x t_{REFW} **010_B**: 2x t_{REFI}, 2x t_{REFIpb}, 2x t_{REFW} **011_B**: 1x t_{REFI}, 1x t_{REFIpb}, 1x t_{REFW} (<=85'C) **100_B**: 0.5x t_{REFI}, 0.5x t_{REFIpb}, 0.5x t_{REFW}, do not de-rate SDRAM AC timing (optional) **101_B**: 0.25x t_{REFI}, 0.25x t_{REFIpb}, 0.25x t_{REFW}, do not de-rate SDRAM AC timing **110_B**: 0.25x t_{REFI}, 0.25x t_{REFIpb}, 0.25x t_{REFW}, de-rate SDRAM AC timing **111_B**: SDRAM High temperature operating limit exceeded
Refresh mode		OP<3>	**0_B** : Both legacy & modified refresh mode supported **1_B** : Only modified refresh mode supported
Temperature Update Flag (TUF)		OP<7>	**0_B**: OP<2:0> value has not changed since last read of MR4. **1_B**: OP<2:0> value has changed since last read of MR4.

NOTE 1 A Mode Register Read from MR4 will reset OP7 to '0'.
NOTE 2 OP7 is reset to '0' at power-up. OP[2:0] bits are undefined after power-up.
NOTE 3 If OP2 equals '1', the device temperature is greater than 85 'C.
NOTE 4 OP7 is set to '1' if OP[2:0] has changed at any time since the last read of MR4.
NOTE 5 WideIO2 might not operate properly when OP[2:0] = 000B or 111B
NOTE 6 WideIO2 devices shall be de-rated by adding 1.875 ns to the following core timing parameters: tRCD, tRC, tRAS, tRP, and tRRD. tDQSCK shall be de-rated according to the tDQSCK de-rating in temperature derating table[1]. Prevailing clock frequency spec and related setup and hold timings shall remain unchanged.
NOTE 7 See 3.4.5.1, Temperature Sensor, for information on the recommended frequency of reading MR4.
NOTE 8 MR4 in channel 0A will only provide the information, and the other channels will not provide.

1. As of publication of this document, under discussion by the formulating committee.

3.4.5.1 Temperature Sensor

WideIO2 devices feature a temperature sensor whose status can be read from MR4. This sensor can be used to determine an appropriate refresh rate, determine whether AC timing de-rating is required in the elevated temperature range, and/or monitor the operating temperature. Either the temperature sensor or the device TOPER may be used to determine whether operating temperature requirements are being met.

WideIO2 devices shall monitor device temperature and update MR4 according to tTSI. Upon exiting self-refresh or power-down, the device temperature status bits shall be no older than tTSI.

When using the temperature sensor, the actual device case temperature may be higher than the TOPER specification that applies for the standard or elevated temperature ranges. For example, TCASE may be above 85 °C when MR4[2:0] equals ‘b011. WideIO2 devices shall allow for 2 °C temperature margin between the point at which the device updates the MR4 value and the point at which the controller re-configures the system accordingly. In the case of tight thermal coupling of the memory device to external hot spots, the maximum device temperature might be higher than what is indicated by MR4.

To assure proper operation using the temperature sensor, applications should consider the following factors:

- TempGradient is the maximum temperature gradient experienced by the memory device at the temperature of interest over a range of 2 °C.
- ReadInterval is the time period between MR4 reads from the system.
- TempSensorInterval (tTSI) is maximum delay between internal updates of MR4.
- SysRespDelay is the maximum time between a read of MR4 and the response by the system.

In order to determine the required frequency of polling MR4, the system shall use the maximum TempGradient and the maximum response time of the system using the following equation:
TempGradient x (ReadInterval + tTSI + SysRespDelay) <= 2 °C

Table 19 — Temperature Sensor

Parameter	Symbol	Max/Min	Value	Unit	Notes
System Temperature Gradient	TempGradient	Max	System Dependent	°C/s	
MR4 Read Interval	ReadInterval	Max	System Dependent	ms	
Temperature Sensor Interval	tTSI	Max	32	ms	
System Response Delay	SysRespDelay	Max	System Dependent	ms	
Device Temperature Margin	TempMargin	Max	2	°C	

For example, if Temp Gradient is 10 °C/s and the SysRespDelay is 1 ms:
(10 °C/s) x (ReadInterval + 32ms + 1ms) <= 2 °C

In this case, ReadInterval shall be no greater than 167 ms.

3.4.5.1 Tempurature Sensor (cont'd)

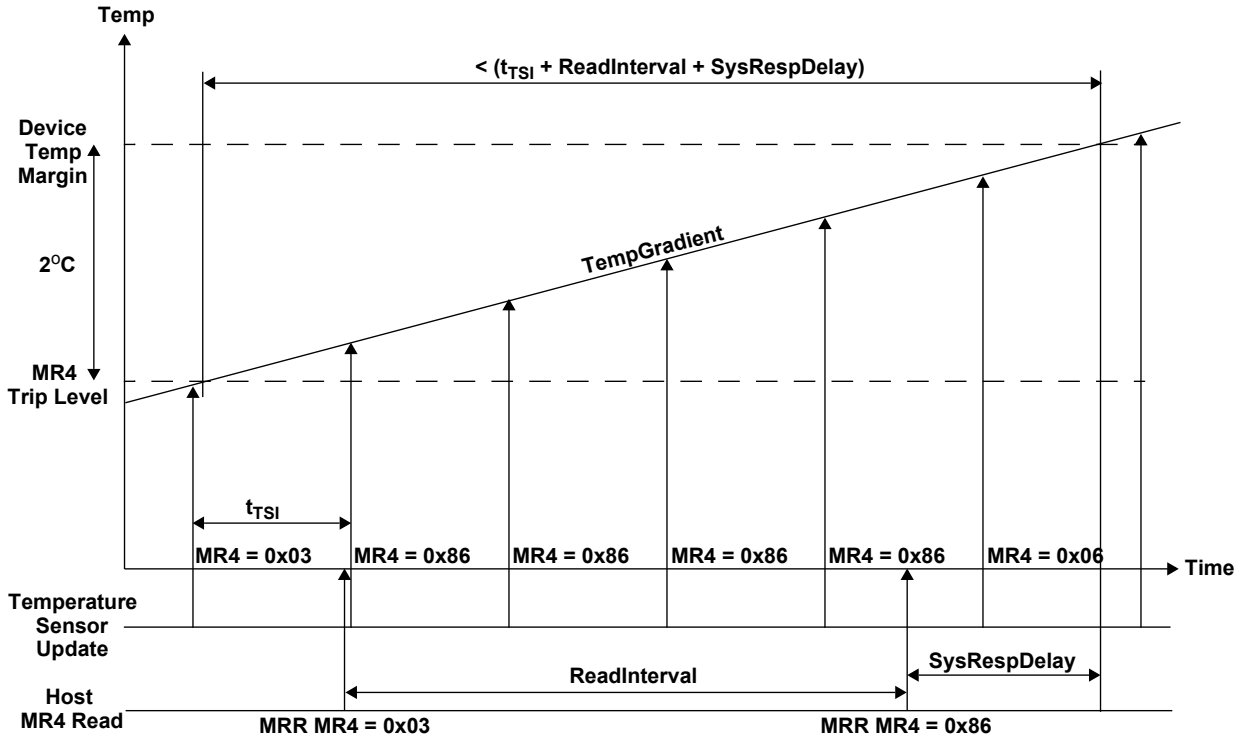


Figure 18 — Temp Sensor Timing

3.4.6 MR5_Basic Configuration 1 (MA<7:0> = 05_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
Wide IO2 Manufacturer ID							

Wide IO2 Manufacturer ID	Read-only	OP<7:0>	See JEP166, WideIO2 Manufacturer ID Codes

3.4.7 MR6_Basic Configuration 2 (MA<7:0> = 06_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
Revision ID1							

Revision ID1	Read-only	OP<7:0>	**00000000_B:** A-version (Vendor specific)

3.4.8 MR7_Basic Configuration 3 (MA<7:0> = 07_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
Revision ID2							

Revision ID2	Read-only	OP<7:0>	**00000000_B:** A-version (Vendor specific)

3.4.9 MR8_Basic Configuration 4 (MA<7:0> = 08_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
(RFU)	I/O width		Die density			Channel	

Channel / die	Read-only	OP<1:0>	**01_B**: 4 channel **10_B**: 8 channel **all others**: reserved
Density / die	Read-only	OP<4:2>	**001_B**: 8Gb **010_B**: 12Gb **011_B**: 16Gb 100_B: 24Gb **101_B**: 32Gb **all others**: reserved
I/O width / channel	Read-only	OP<6:5>	**00_B:** x64 **all others** : Reserved

3.4.10 MR9_Test Mode (MA<7:0> = 09_H)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
Test Mode							

NOTE 1 MR9 is vendor specific.

3.4.11 MR10_Reserved (MA<7:0> = $0A_H$)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
(RFU)							

3.4.12 MR11_PASR-1 (MA<7:0> = $0B_H$)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
Bank Masking							

Bank Masking	Write-only	OP<7:0>	0_B: refresh enable to the bank (=unmasked, default) 1_B: refresh blocked (=masked)

OP	Bank Mask	8-Bank SDRAM	4-Bank SDRAM
0	XXXXXXX1	BANK 0	BANK 0
1	XXXXXX1X	BANK 1	BANK 1
2	XXXXX1XX	BANK 2	BANK 2
3	XXXX1XXX	BANK 3	BANK 3
4	XXX1XXXX	BANK 4	N/A
5	XX1XXXXX	BANK 5	N/A
6	X1XXXXXX	BANK 6	N/A
7	1XXXXXXX	BANK 7	N/A

3.4.13 MR12_PASR-2 (MA<7:0> = $0C_H$)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
Segment Masking							

Segment Masking	Write-only	OP<7:0>	0_B: refresh enable to the segment (=unmasked, default) 1_B: refresh blocked (=masked)

Segment	OP	Segment Mask	Rmsb:Rmsb-2
0	0	XXXXXXX1	000_B
1	1	XXXXXX1X	001_B
2	2	XXXXX1XX	010_B
3	3	XXXX1XXX	011_B
4	4	XXX1XXXX	100_B
5	5	XX1XXXXX	101_B
6	6	X1XXXXXX	110_B
7	7	1XXXXXXX	111_B

NOTE 1 The table indicates the range of row addresses in each masked segment. X is do not care for a particular segment.
NOTE 2 "Rmsb:Rmsb-2" means the most significant three Row address bits are used as the segment address independent of the bit density or bank configuration.
NOTE 3 For 12Gb and 24Gb device, no memory present at addresses with R13=R14=HIGH, and segment masks 6 and 7 are ignored.

3.4.14 MR13_Post Package Repair Resource (MA<7:0> = $0D_H$)

Mode Register 13 contains one bit of readout per bank indicating that at least one resource is available for Post Package Repair programming.

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
Bank7	Bank6	Bank5	Bank4	Bank3	Bank2	Bank1	Bank0

Post Package Repair Resource	Read-only	OP<7:0>	$\mathbf{0_B}$: PPR Resource is not available $\mathbf{1_B}$: PPR Resource is available (One or more)

NOTE 1 For 4x64 die, OP[7:4]=RFU.

3.4.15 MR14_TRR-1 (MA<7:0> = $0E_H$)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
TRR Mode	TRR Mode - Bank			(RFU)			

TRR Mode	Write-only	OP<7>	$\mathbf{0_B}$: Disabled $\mathbf{1_B}$: Enabled
TRR Mode - BAn	Write-only	OP<6:4>	$\mathbf{000_B}$: Bank 0 $\mathbf{001_B}$: Bank 1 $\mathbf{010_B}$: Bank 2 $\mathbf{011_B}$: Bank 3 $\mathbf{100_B}$: Bank 4 $\mathbf{101_B}$: Bank 5 $\mathbf{110_B}$: Bank 6 $\mathbf{111_B}$: Bank 7

3.4.16 MR15_TRR-2 (MA<7:0> = $0F_H$)

OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
(RFU)				Mac Cap.	Mac Value		

Mac Capabil-ites	Read-only	OP<3>	**0_B**: Bits 2:0 define MAC limit **1_B**: Unlimited MAC limit[1,2]
Mac Value	Read-only	OP<2:0>	**000_B**: Unknown when bit OP3=0[3], Unlimited when bit OP3=1[4] **001_B**: 700K **010_B**: 600K **011_B**: 500K **100_B**: 400K **101_B**: 300K **110_B**: 200K **111_B**: Reserved

NOTE 1 Unlimited MAC means that there is no restriction to the number of Activates in a refresh period.
NOTE 2 Bits OP2:OP0 are set to zero.
NOTE 3 Unknown means that device is not tested for tMAC and pass/fail value is unknown.
NOTE 4 There is no restriction to the number of activates.

4 Command Definitions and Timing

4.1 Command Truth Table

WideIO2 devices use a packetized, double data rate architecture on the Command/Address (CA) bus to reduce the number of input pins in the system. The 11-bit CA bus contains command, bank address, row address, column address, and mode register address. Each command uses one clock cycle, during which command information is transferred on the positive edge CK_t and positive edge of CK_c (2UI) for a 22-bit payload. Each channel will include a CA[10:0] which is independent of other channels. Table 20 shows how CA Bits are mapped to the 22-bit payload.

Table 20 — Command Truth Table

Function	Symbol	CKE		CS_n	Rising Edge of Clock	DDR CA Row Pins (11)										
		CK_t (n-1)	CK_t (n)			CA [0]	CA [1]	CA [2]	CA [3]	CA [4]	CA [5]	CA [6]	CA [7]	CA [8]	CA [9]	CA [10]
Deselect	DES	H	H	H	CK_t	V	V	V	V	V	V	V	V	V	V	V
				V	CK_c	V	V	V	V	V	V	V	V	V	V	V
NOP	NOP	H	H	L	CK_t	H	H	H	V	V	V	V	V	V	V	V
				V	CK_c	V	V	V	V	V	V	V	V	V	V	V
Activate	ACT	H	H	L	CK_t	L	H	R11	R12	R13	R14	R15	BA0	BA1	BA2	V
				V	CK_c	R0	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
PreCharge (per Bank, all)	PRE	H	H	L	CK_t	H	H	L	H	AB	V	V	BA0	BA1	BA2	V
				V	CK_c	V	V	V	V	V	V	V	V	V	V	V
Refresh (per Bank, all)	REFA	H	H	L	CK_t	L	L	H	V	AB	V	V	BA0	BA1	BA2	V
				V	CK_c	V	V	V	V	V	V	V	V	V	V	V
Power Down Entry	PDE	H	L	H	CK_t	V	V	V	V	V	V	V	V	V	V	V
				V	CK_c	V	V	V	V	V	V	V	V	V	V	V
Self Refresh Entry	SRE	H	L	L	CK_t	L	L	H	V	V	V	V	V	V	V	V
				V	CK_c	V	V	V	V	V	V	V	V	V	V	V
Power Down & Self Refresh Exit	PDX/SRX	L	H	H	CK_t	X	X	X	X	X	X	X	X	X	X	X
				V	CK_c	X	X	X	X	X	X	X	X	X	X	X
Maintain Power Down or Self Refresh	MLPM	L	L	X	CK_t	X	X	X	X	X	X	X	X	X	X	X
				X	CK_c	X	X	X	X	X	X	X	X	X	X	X
Read	RD	H	H	L	CK_t	H	L	H	V	BL	C1	C2	BA0	BA1	BA2	V
				V	CK_c	AP	C3	C4	C5	C6	C7	C8	V	V	V	V
Write	WR	H	H	L	CK_t	H	L	L	L	BL	C1	C2	BA0	BA1	BA2	V
				V	CK_c	AP	C3	C4	C5	C6	C7	C8	V	V	V	V
Masked Write	MWR	H	H	L	CK_t	H	L	L	H	L	C1	C2	BA0	BA1	BA2	V
				V	CK_c	AP	C3	C4	C5	C6	C7	C8	V	V	V	V
Mode Register Write	MRW	H	H	L	CK_t	L	L	L	L	MA0	MA1	MA2	MA3	V	V	V
				V	CK_c	V	V	OP0	OP1	OP2	OP3	OP4	OP5	OP6	OP7	V
Mode Register Read	MRR	H	H	L	CK_t	L	L	L	H	MA0	MA1	MA2	MA3	V	V	V
				V	CK_c	V	V	V	V	V	V	V	V	V	V	V

NOTE 1 X = Don't care and not driven to a logic level

NOTE 2 V= Don't care but must be driven to a logic value

NOTE 3 C0 is not defined in the command table and is assumed to be 0. C1 is used to drive critical data first for BL4.

NOTE 4 BL = Burst Length on-the-fly. When on-the-fly is enabled, the Burst Length equals 4 when BL is LOW and equals 8 when BL is HIGH. BL must be held to a valid level when on-the-fly is disabled. The number of bursts will be 4 for Masked Write commands.

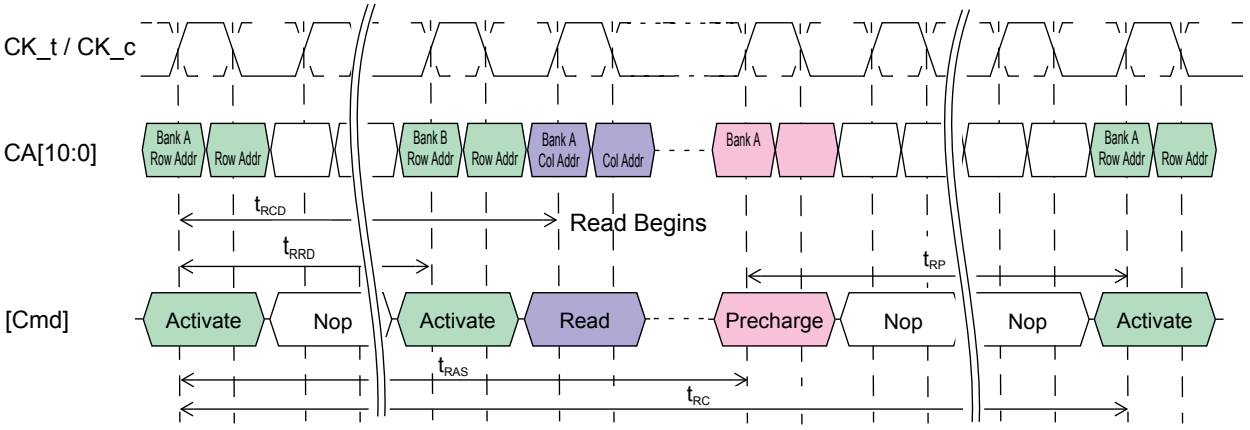
NOTE 5 For 2KB page size (51.2GB/s die), C8 and BA2 are don't cares (V).

NOTE 6 AB HIGH during Precharge command indicates that all bank Precharge will occur. In this case, Bank Address is a do-not-care. AB LOW during Precharge command indicates that a precharge of the bank selected by the Bank Address will occur.

NOTE 7 AB HIGH during Refresh command indicates that all bank Refresh will occur. In this case, Bank Address is do-not-care. AB LOW during Refresh command indicates that a refresh of the bank selected by the Bank Address will occur.

4.2 Activate Command

The Activate command is issued by holding CS_n LOW, CA0 LOW, and CA1 HIGH at the rising edge of the clock. The bank addresses are used to select the desired bank. Row addresses are used to determine which row to activate in the selected bank. The Activate command must be applied before any Read, Write, or Masked Write operation can be executed. The device can accept a Read, Write, or Masked Write command at tRCD after the Activate command is issued. After a bank has been activated it must be precharged before another Activate command can be applied to the same bank. The bank active and precharge times are defined as tRAS and tRP, respectively. The minimum time interval between successive Activate commands to the same bank is determined by the RAS cycle time of the device (tRC). The minimum time interval between Activate commands to different banks is tRRD (see Figure 19).



NOTE 1 A PRECHARGE-all command uses t_{RPab} timing, while a single-bank PRECHARGE command uses t_{RPpb} timing. In this figure, t_{RP} is used to denote either an all-bank PRECHARGE or a single-bank PRECHARGE.

Figure 19 — Activate Command

4.2.1 Device Operation

Certain restrictions on operation of the WideIO2 devices must be observed. There are two rules: One rule restricts the number of sequential ACTIVATE commands that can be issued; the other provides more time for RAS precharge for a PRECHARGE ALL command. The rules are as follows:

Sequential Bank Activation Restriction: No more than 4 banks may be activated (or refreshed, in the case of REFpb) in a rolling t_{FAW} window. The number of clocks in a t_{FAW} period is dependent upon the clock frequency, which may vary. If the clock frequency is not changed over this period, converting to clocks is done by dividing t_{FAW}[ns] by t_{CK}[ns], and rounding up to the next integer value. As an example of the rolling window, if RU(t_{FAW}/t_{CK}) is 10 clocks, and an ACTIVATE command is issued in clock n, no more than three further ACTIVATE commands can be issued at or between clock $n + 1$ and $n + 9$. REFpb also counts as bank activation for purposes of t_{FAW}. If the clock frequency is changed during the t_{FAW} period, the rolling t_{FAW} window may be calculated in clock cycles by adding up the time spent in each clock period. The t_{FAW} requirement is met when the previous n clock cycles exceeds the t_{FAW} time.

The Device Precharge-All Allowance: t_{RP} for a PRECHRGE ALL command must equal t_{RPab}, which is greater than t_{RPpb}.

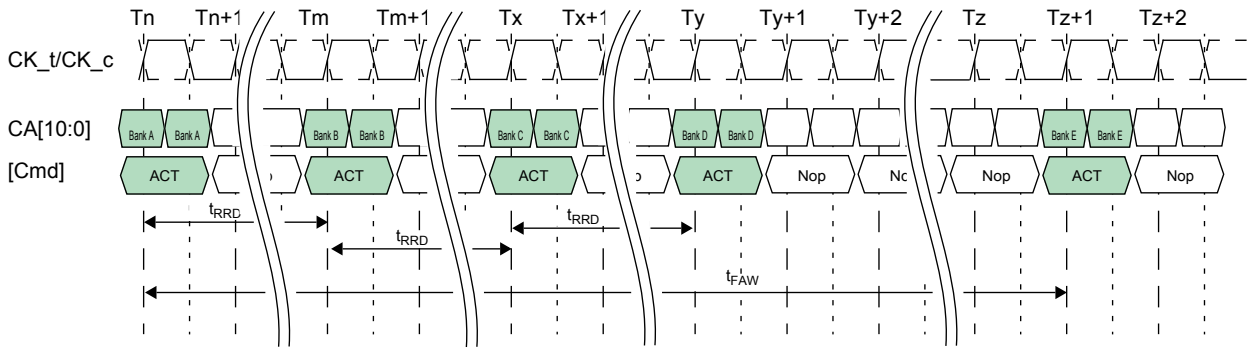


Figure 20 — WideIO2 t_{FAW} Timing

4.3 Read And Write Access Modes

After a bank has been activated, a read or write cycle can be executed. This is accomplished by setting CS_n LOW, CA0 HIGH, and CA1 LOW at the rising edge of the clock. CA2 and CA3 must also be defined at this time to determine whether the access cycle is a read operation (CA2 HIGH, CA3 VALID that means it is HIGH or LOW) or a write operation (CA2 LOW, CA3 Low).

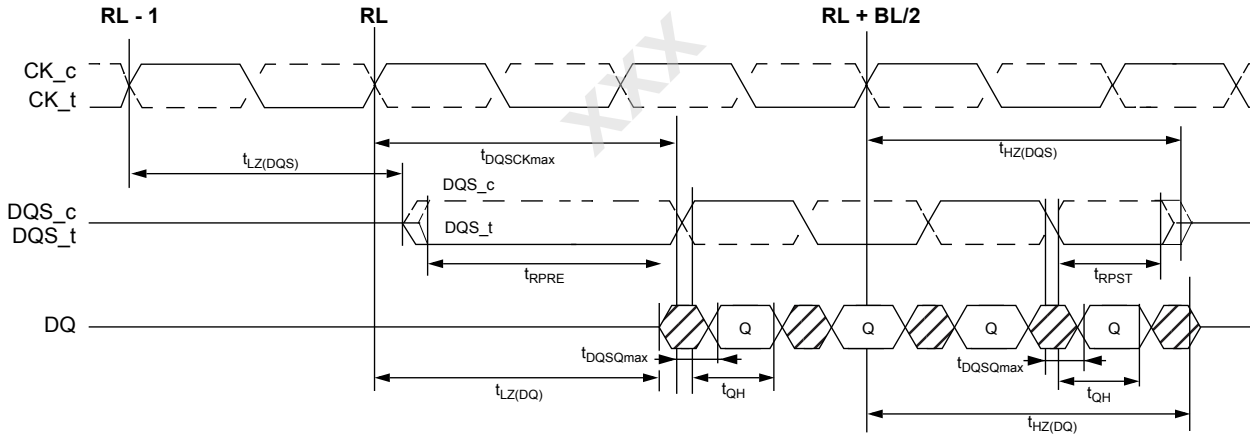
The WideIO2 SDRAM provides a fast column access operation. A single Read or Write Command will initiate a burst read or write operation on successive clock cycles.

During a READ or WRITE command, WIDEIO2 SDRAM will support BL4 and BL8 on the fly using address CA4 at Clock rising edge during the READ or WRITE (AUTO PRECHARGE can be enabled or disabled).

4.4 Read Command

4.4.1 Burst Read Command

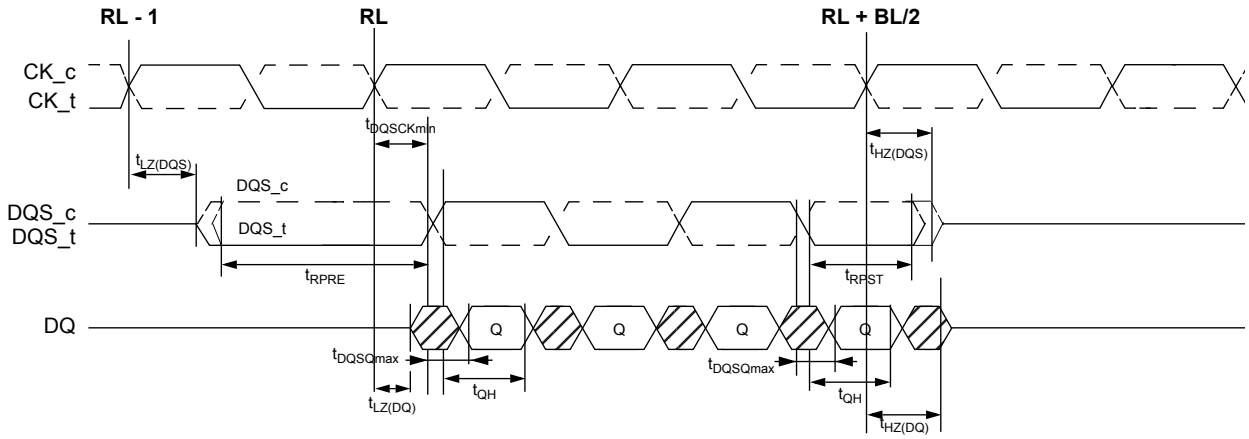
The Burst Read command is initiated by having CS_n LOW, CA0 HIGH, CA1 LOW and CA2 HIGH at the rising edge of the clock. The command address bus inputs, CA5r-CA6r and CA1f-CA6f, determine the starting column address for the burst. The Read Latency (RL) is defined from the rising edge of the clock on which the Read Command is issued to the rising edge of the clock from which the tDQSCK delay is measured. The first valid datum is available RL × t_{CK} + t_{DQSCK} + t_{DQSQ} after the rising edge of the clock where the Read Command is issued. The data strobe output is driven LOW t_{RPRE} before the first rising valid strobe edge. The first bit of the burst is synchronized with the first rising edge of the data strobe. Each subsequent data-out appears on each DQ pin edge aligned with the data strobe. The RL is programmed in the mode registers.Timings for the data strobe are measured relative to the crosspoint of DQS_t and its complement, DQS_c.



NOTE 1 t_{DQSCK} may span multiple clock periods.
NOTE 2 An effective Burst Length of 4 is shown.

Figure 21 — Data output (read) timing (tDQSCKmax) (AMD_RB13327)

4.4.1 Burst Read Command (cont'd)



NOTE 1 An effective Burst Length of 4 is shown.

Figure 22 — Data output (read) timing (tDQSCKmin) (AMD_RB13327)

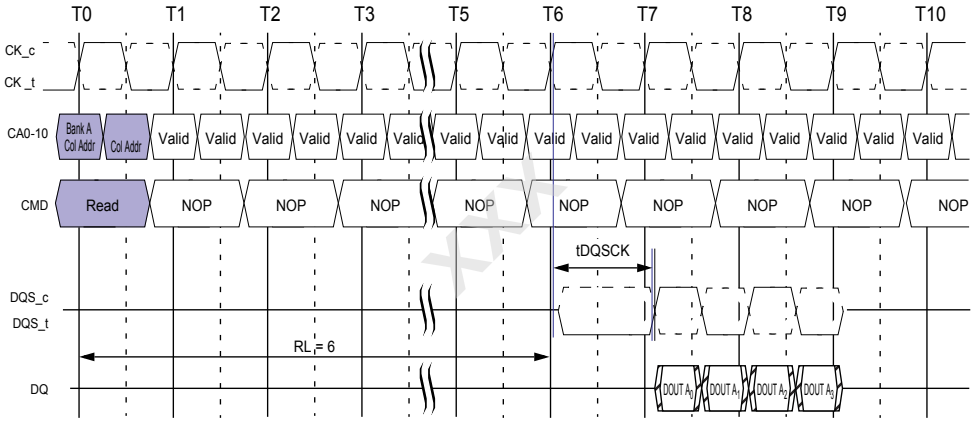


Figure 23 — WIDEIO2: Burst read : RL = 6, BL = 4, tDQSCK > tCK(AMD_RB13327)

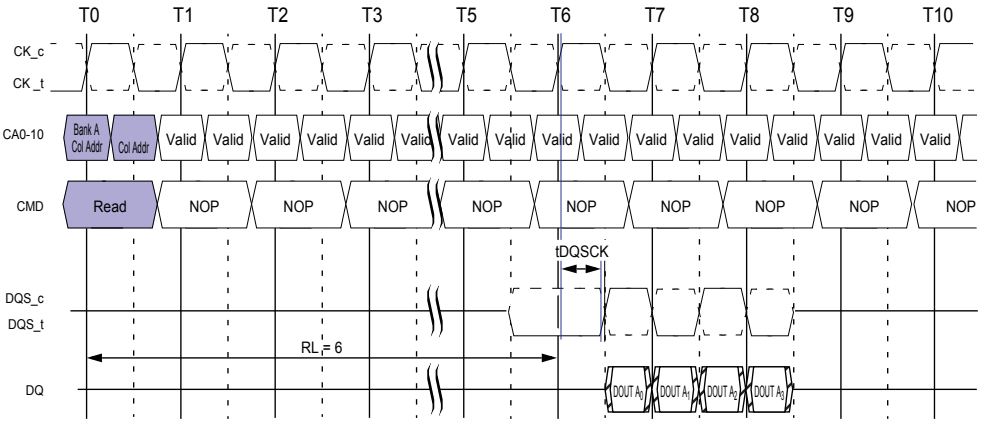


Figure 24 — WIDEIO2: Burst read : RL = 6, BL = 4, tDQSCK < tCK(AMD_RB13327)

4.4.1 Burst Read Command (cont'd)

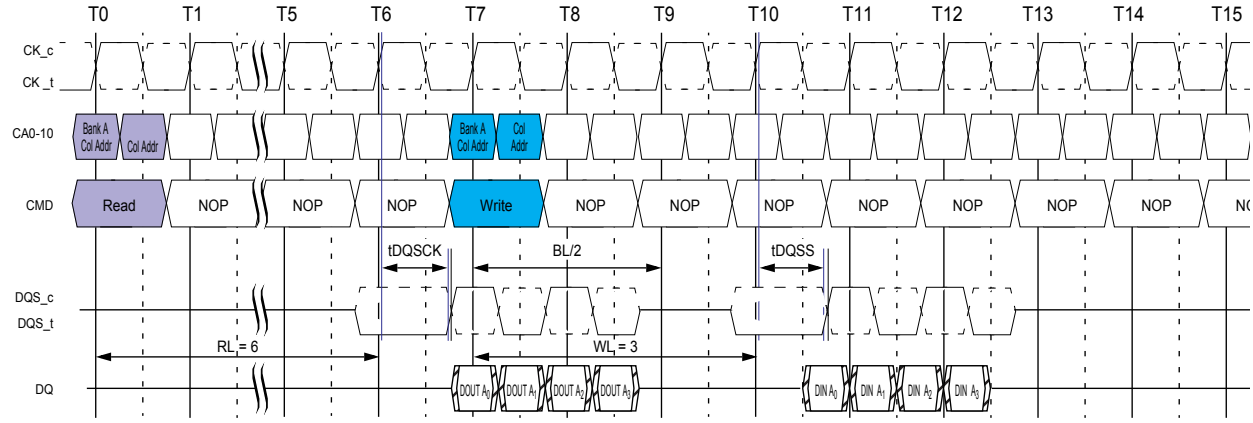


Figure 25 — WIDEIO2: Burst read followed by burst write: RL = 6, WL = 3, BL = 4(AMD_RB13327)

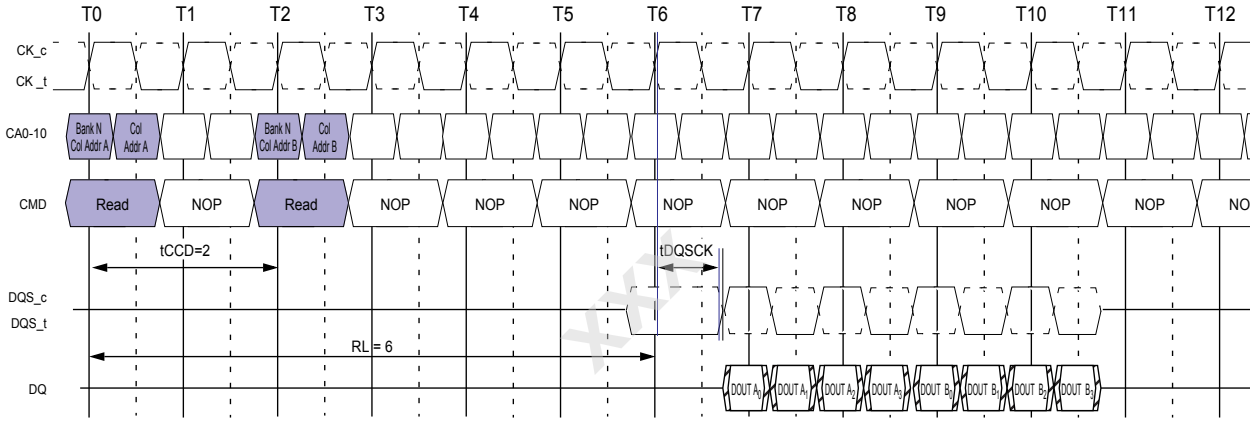


Figure 26 — WIDEIO2: Seamless burst read: RL = 6, BL = 4, tCCD = 2(AMD_RB13327)

4.4.2 Back-to-Back Reads from Different Ranks

Figure 27 and Figure 28 are example waveform only. The waveform may vary based on cycle time and tDQSCK min/max value.

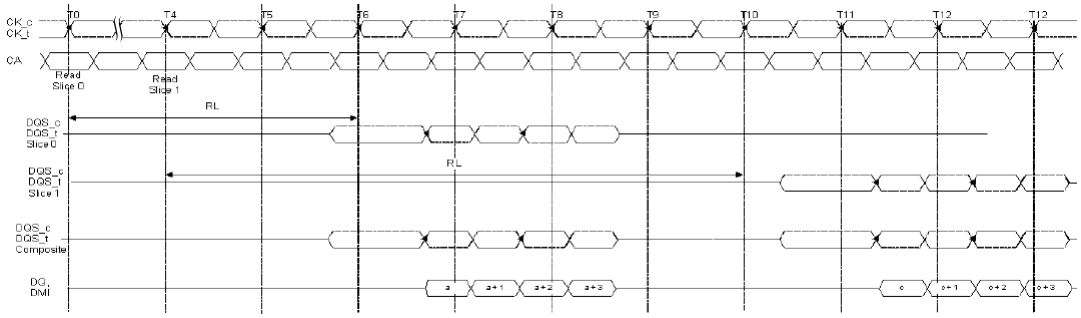


Figure 27 — Consecutive burst read from Different Slice : RL = 6, BL = 4, Slice 0 - Early tDQSCK, Slice 1 - Late tDQSCK(AMD_RB13327)

4.4.2 Back-to-Back Reads from Different Ranks (cont'd)

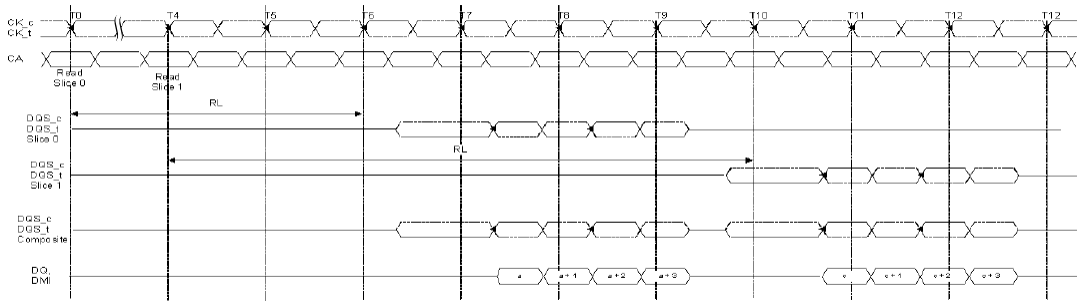


Figure 28 — Consecutive burst read from Different Slice : RL = 6, BL = 4, Slice 0 - Late tDQSCK, Slice 1 - Early tDQSCK(AMD_RB13327)

4.4.3 Read preamble

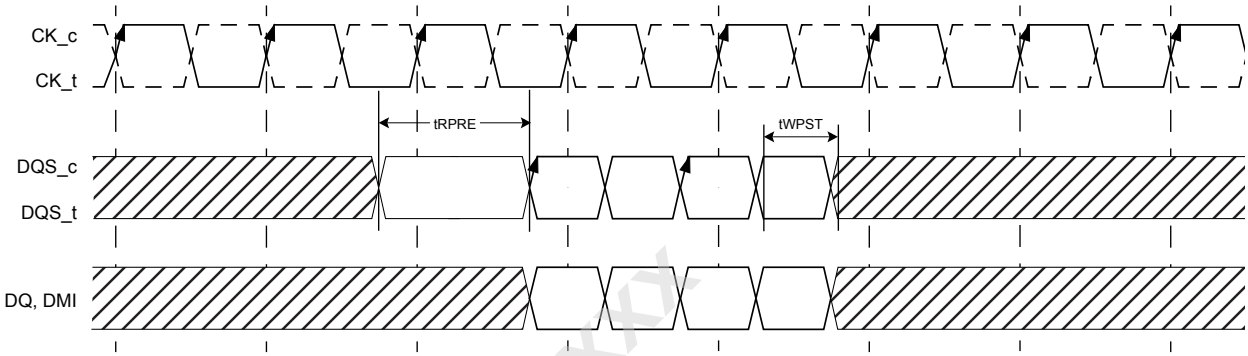


Figure 29 — Read Preamble

4.5 Write command

4.5.1 Burst Write Command

The Burst Write command is initiated by having CS_n LOW, CA0 HIGH, CA1 LOW and CA2 LOW at the rising edge of the clock. The command address bus inputs, CA5r-CA6r and CA1f-CA6f, determine the starting column address for the burst. The Write Latency (WL) is defined from the rising edge of the clock on which the Write Command is issued to the rising edge of the clock from which the tDQSS delay is measured. The first valid datum shall be driven WL × tCK + tDQSS from the rising edge of the clock from which the Write command is issued. The data strobe signal (DQS) should be driven LOW tWPRE prior to the data input. The data bits of the burst cycle must be applied to the DQ pins tDS prior to the respective edge of the DQS_t, DQS_c and held valid until tDH after that edge.

The burst data are sampled on successive edges of the DQS_t, DQS_c until the burst length is completed, which is 4 or 8 bit burst.

For WideIO2 SDRAM devices, tWR must be satisfied before a precharge command to the same bank may be issued after a burst write operation.

4.5.1 Burst Write Command (cont'd)

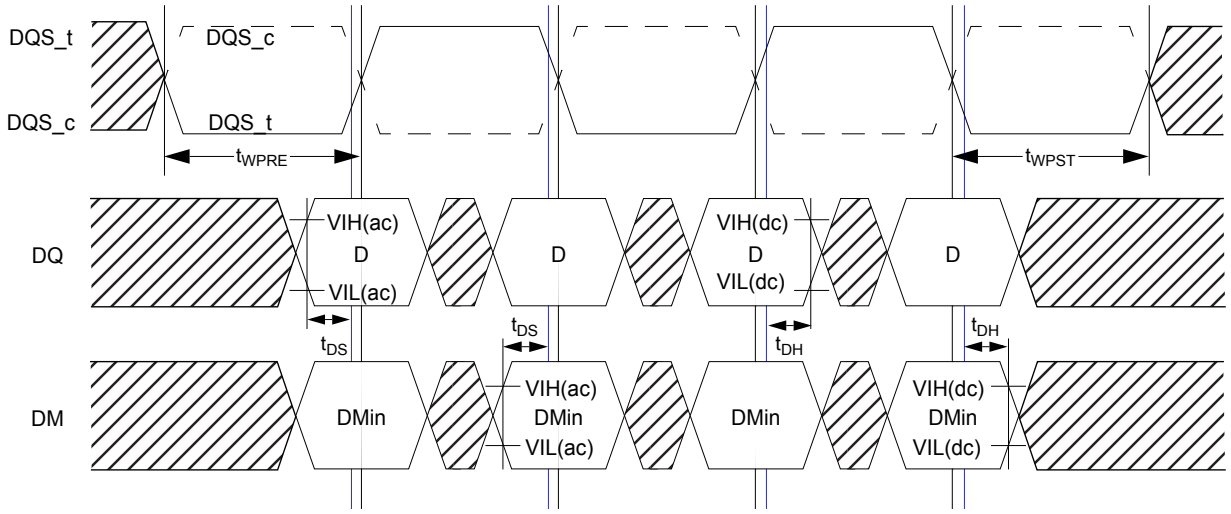


Figure 30 — Data input (write) timing (Elpida RB13252)

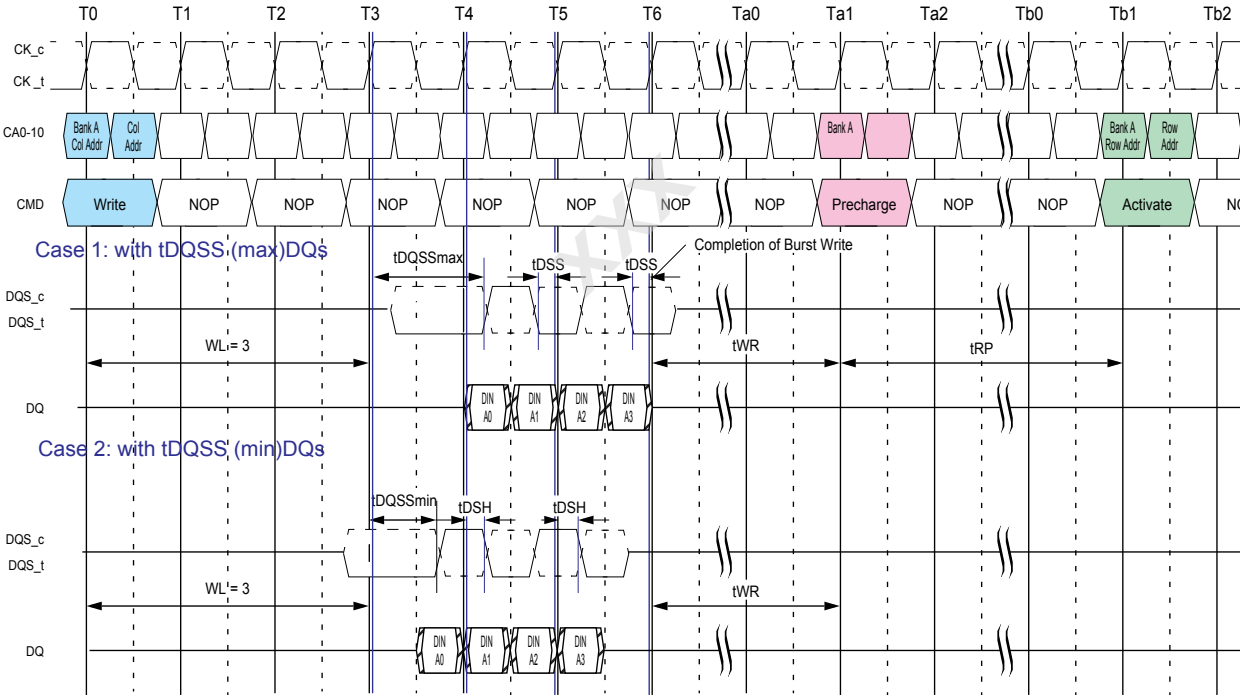
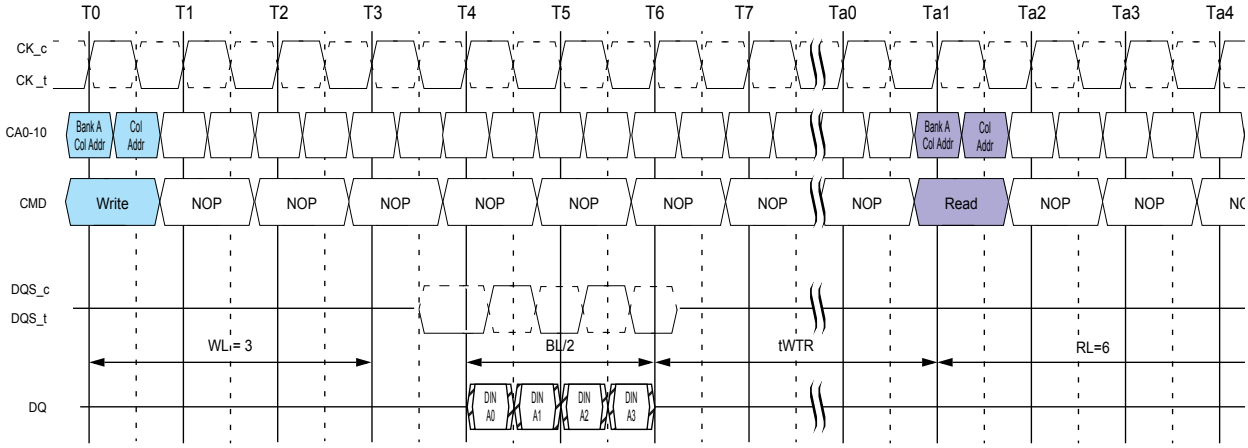


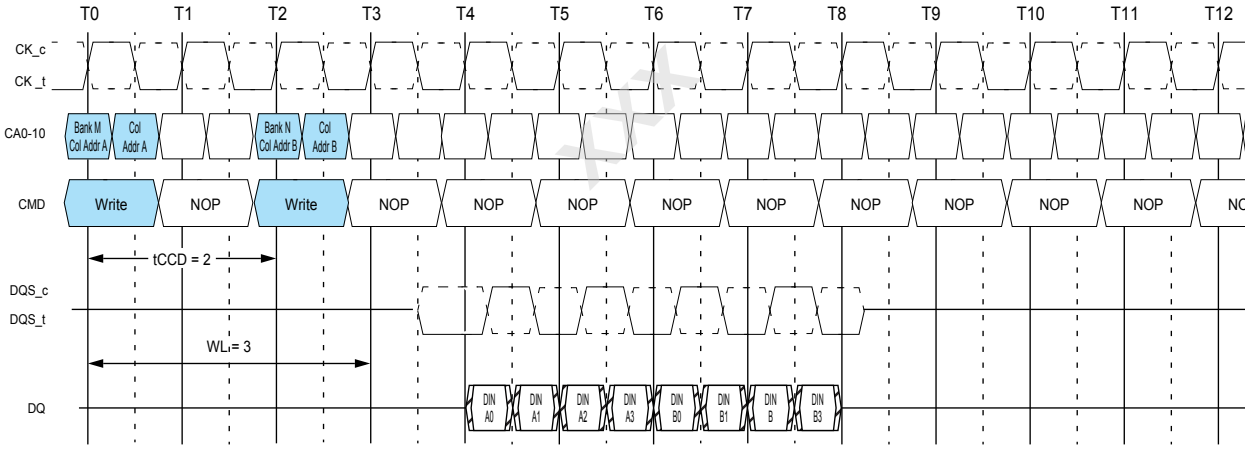
Figure 31 — WIDEIO2: Burst write: WL = 3, BL = 4 (Elpida RB13252)

4.5.1 Burst Write Command (cont'd)



NOTE 1 The minimum number of clock cycles from the burst write command to the burst read command for any bank is [WL + 1 + BL/2 + Roundup(tWTR/tCK)].
NOTE 2 tWTR starts at the rising edge of the clock after the last valid input datum.

Figure 32 — WIDEIO2: Burst write followed by burst read: RL=6, WL = 3, BL = 4 (Elpida RB13252)



NOTE 1 The seamless burst write operation is supported by enabling a write command every other clock for BL = 4 operation. This operation is allowed regardless of same or different banks as long as the banks are activated.

Figure 33 — Seamless burst write: WL = 3, BL = 4, tCCD = 2 (Elpida RB13252)

4.5.2 Write Preamble

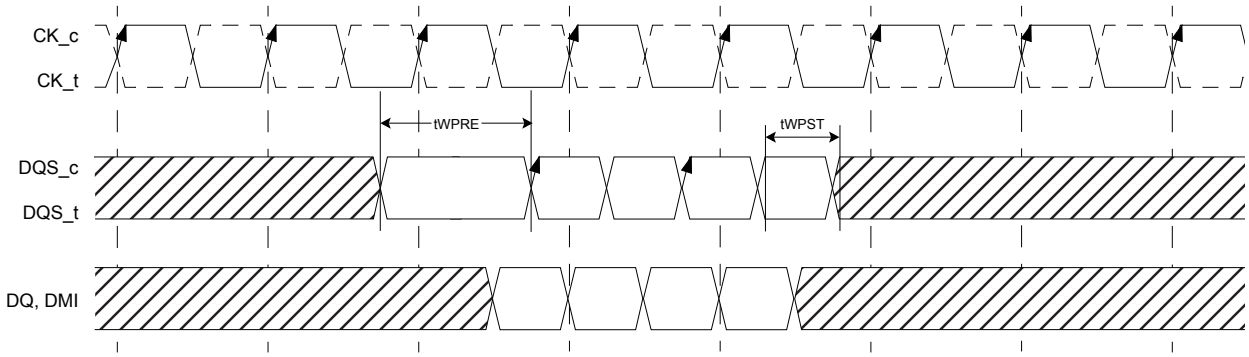
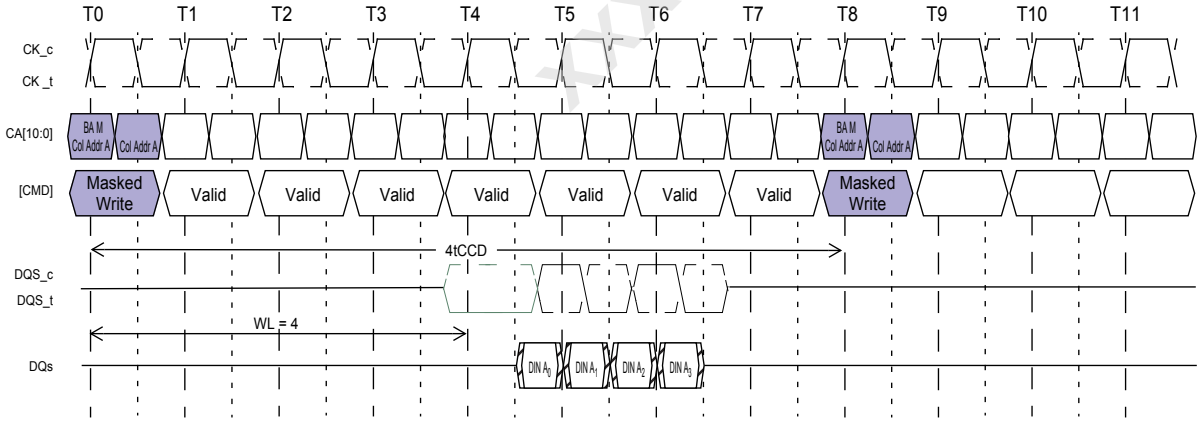


Figure 34 — Write Preamble

4.6 Masked Write Command

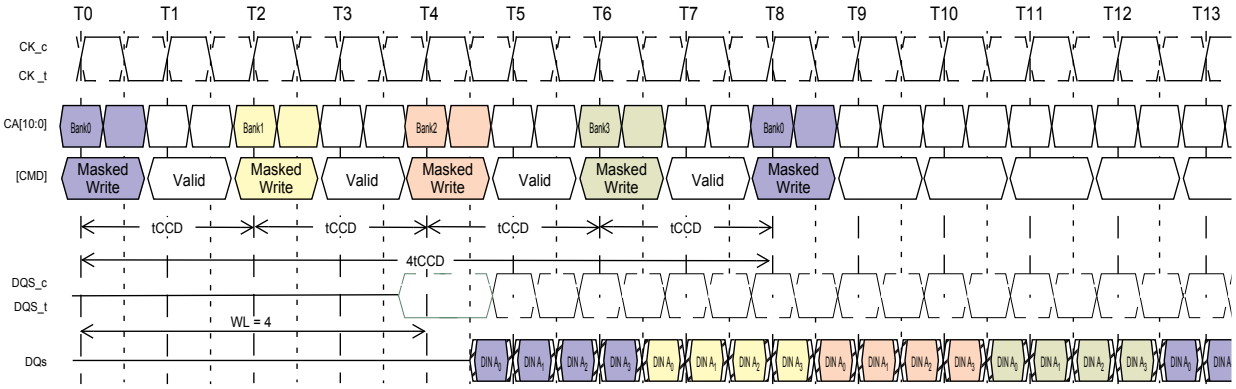
The WideIO2 SDRAM requires that Write operations which include a byte mask anywhere in the burst sequence must use the Masked Write command. This allows the DRAM to implement efficient data protection schemes based on larger data blocks. A minimum time, tCCDMW, must be satisfied between a write (masked or unmasked) and a subsequent Masked Write to the same bank, to allow the Masked Write to perform its internal Read.Modify.Write. One Data Mask-Invert (DMI) pin is provided per byte lane, and the Data Mask-Invert timings match data bit (DQ) timing. See the section on "Data Mask Invert" for more information on the use of the DMI signal.



NOTE 1 Masked write command supports only BL=4 operation. For BL=8 configuration, the system needs to insert only 4 bit wide data for masked write operation.

Figure 35 — Masked Write command Same Bank

4.6 Masked Write Command (cont'd)



NOTE 1 Masked write command supports only BL=4 operation. For BL=8 configuration, the system needs to insert only 4 bit wide data for masked write operation.

Figure 36 — Masked Write command Different Bank

4.6.1 Masked Write Timing Constraints

4.6.1.1 Same bank

[Units: tCK]

Next CMD / Current CMD	Active	Read (BL=4 or 8)	Write (BL=4 or 8)	Masked Write	Precharge
Active	illegal	RU(t_{RCD}/t_{CK})	RU(t_{RCD}/t_{CK})	RU(t_{RCD}/t_{CK})	RU(t_{RAS}/t_{CK})
Read with BL = 4	illegal	2 1)	RL+RU($t_{DQSCK(max)}/t_{CK}$) +BL/2+1-WL	RL+RU($t_{DQSCK(max)}/t_{CK}$) +BL/2+1-WL	BL/2+max{(2,RU(t_{RTP}/t_{CK})}-2
Read with BL = 8	illegal	4 2)	RL+RU($t_{DQSCK(max)}/t_{CK}$) +BL/2+1-WL	RL+RU($t_{DQSCK(max)}/t_{CK}$) +BL/2+1-WL	BL/2+max{(2,RU(t_{RTP}/t_{CK})}-2
Write with BL = 4	illegal	WL+1+BL/2 +RU(t_{WTR}/t_{CK})	2 1)	t_{CCDMW} 3)	WL+ 1 + BL/2 +RU(t_{WR}/t_{CK})
Write with BL = 8	illegal	WL+1+BL/2 +RU(t_{WTR}/t_{CK})	4 2)	t_{CCDMW} +2 4)	WL+ 1 + BL/2 +RU(t_{WR}/t_{CK})
Masked Write	illegal	WL+1+BL/2 +RU(t_{WTR}/t_{CK})	t_{CCD}	t_{CCDMW} 3)	WL+ 1 + BL/2 +RU(t_{WR}/t_{CK})
Precharge	RU(t_{RP}/t_{CK}), RU(t_{RPab}/t_{CK})	illegal	illegal	illegal	2

NOTE 1 In the case of BL = 4, t_{CCD} is 2 × t_{CK}.
NOTE 2 In the case of BL = 8, t_{CCD} is 4 × t_{CK}.
NOTE 3 t_{CCDMW} = 8 × tCK (4 × t_{CCD} at BL=4)
NOTE 4 Write with BL=8 operation has 2 × tCK longer than BL =4.

4.6.1.2 Different bank

[Units: tCK]

Next CMD Current CMD	Active	Read (BL=4 or 8)	Write (BL=4 or 8)	Masked Write	Precharge
Active	RU(t_{RRD}/t_{CK})	1	1	1	1
Read with BL = 4	1	2 1)	RL+RU($t_{DQSCK(max)}$ /t_{CK}) +BL/2+1-WL	RL+RU($t_{DQSCK(max)}$ /t_{CK}) +BL/2+1-WL	1
Read with BL = 8	1	4 2)	RL+RU($t_{DQSCK(max)}$ /t_{CK}) +BL/2+1-WL	RL+RU($t_{DQSCK(max)}$ /t_{CK}) +BL/2+1-WL	1
Write with BL = 4	1	WL+1+BL/2 +RU(t_{WTR}/t_{CK})	2 1)	2 1)	1
Write with BL = 8	1	WL+1+BL/2 +RU(t_{WTR}/t_{CK})	4 2)	4 2)	1
Masked Write	1	WL+1+BL/2 +RU(t_{WTR}/t_{CK})	2 1)	2 1)	1
Precharge	1	1	1	1	2

NOTE 1 In the case of BL = 4, t_{CCD} is 2 × tCK.

NOTE 2 In the case of BL = 8, t_{CCD} is 4 × tCK.

4.7 WideIO2 Data Mask (DM) and Data Bus Inversion (DBIac) Function

WideIO2 device supports Data Mask (DM) function for Masked Write operation and Data Bus Inversion (DBIac) function for Write or Masked Write and Read operation. WideIO2 supports DM and DBIac function with a byte granularity. WideIO2 device has one Data Mask Inversion (DMI) signal pin per byte; total of 8 DMI signals per channel. DMI signal is a bi-directional DDR signal and is sampled along with the DQ signals for Read and Write or Masked Write operation.

Table 21 — Odd/Even Byte Groups

DQ Signals	DMI Signal	Byte Type
DQ[7:0]	DMI0	Even Byte
DQ[15:8]	DMI1	Odd Byte
DQ[23:16]	DMI2	Even Byte
DQ[31:24]	DMI3	Odd Byte
DQ[39:32]	DMI4	Even Byte
DQ[47:40]	DMI5	Odd Byte
DQ[55:48]	DMI6	Even Byte
DQ[63:56]	DMI7	Odd Byte

DM function can be enabled or disabled per WideIO2 Mode Register MR3 OP[3]. DBIac function can be enabled or disabled independently for Write/Masked Write and Read operation per WideIO2 mode register MR3 OP[7] and MR3 OP[6] respectively. DM and Write or Masked Write DBIac function can be enabled simultaneously in WideIO2 mode register.

Throughout the WideIO2 spec, word DBIac refers to internal state of the device and does not reflect the pin state. The word DMI is used to reflect pin state.

4.7.1 DBIac States (Either Write/Masked Write DBIac or Read DBIac is enabled)

The WideIO2 device resets DBI value to Low (All eight DQs and DMI signals within a byte group) whenever either of the following events:

. RST_n signal de-assertion
. The WideIO2 device registers Mode Register Set command (MRS)
. The WideIO2 device registers Masked Write command after Read command
. The WideIO2 device registers back to back Masked Write command with a gap (> tCCDmin)
. The WideIO2 device registers back to back Write followed by Masked Write command with a gap (> tCCDmin)
. Power Down exit, Self Refresh exit

4.7.1.1 Write Operation

The WideIO2 device requires DQ bus to be preconditioned during Write commands to avoid simultaneous DQ switching that might cause violation of the VDDQ specifications.

DM Function Disable - When a Write command is registered, WideIO2 device treats DMI signal as DBI signal and determines if it needs to invert the Write data on its DQ inputs.

DM Function Enable - When a Write command is registered, WideIO2 device treats DMI signal as DBI signal and determines if it needs to invert the Write data on its DQ inputs.

For back to back Write followed by Masked Write operation; if a gap is required (i.e., > tCCDmin), then WideIO2 device uses DBI reset value (Low) for DBIac calculation for the first beat of Masked Write command burst. If there is no gap required for Write followed by Masked Write (i.e., =tCCDmin), then WideIO2 device uses the last beat of the previous burst for DBIac calculation for the first beat of Masked Write command burst.

Back to back Write followed by Write operation to a different Rank requires a gap and WideIO2 device treats DMI signal as DBI signal.

Back to back Write followed by Masked Write operation to a different Rank requires a gap and hence treated the same way as back to back Write followed by Masked Write operation with a gap from the same rank as noted above.

Write followed by Read - The WideIO2 device pre-conditions the bus as if it was a first Read command (4.7.1.3) once it registers Read command after Write command.

4.7.1.2 Masked Write Operation

The WideIO2 device requires DQ bus to be preconditioned during Masked Write commands to avoid simultaneous DQ switching that might cause violation of the VDDQ specifications.

When a first Masked Write command is registered after DBI reset state, WideIO2 device uses DBI reset value (Low) as the previous value for DBIac calculation for the first beat of Masked Write command burst.

For back to back Masked Write followed by Masked Write operation; if a gap is required (i.e., > tCCDmin), then WideIO2 device uses DBI reset value (Low) for DBIac calculation for the first beat of second Masked Write command burst. If there is no gap required for back to back Masked Write command (i.e., = tCCDmin), the WideIO2 device uses the last beat of the previous burst for DBIac calculation for the first beat of Masked Write command burst.

Back to back Masked Write followed by Write to a different rank requires a gap and DMI signal is treated as DBI signal for Write command.

Back to back Masked Write followed by Masked Write operation to a different Rank requires a gap and hence treated the same way as back to back Masked Write followed by Masked Write operation with a gap from the same rank as noted above.

4.7.1.2 Masked Write Operation (cont'd)

Masked Write followed by Read - The WideIO2 device pre-conditions the bus as if it was a first Read command (4.7.1.3) once it registers Read command after Masked Write command.

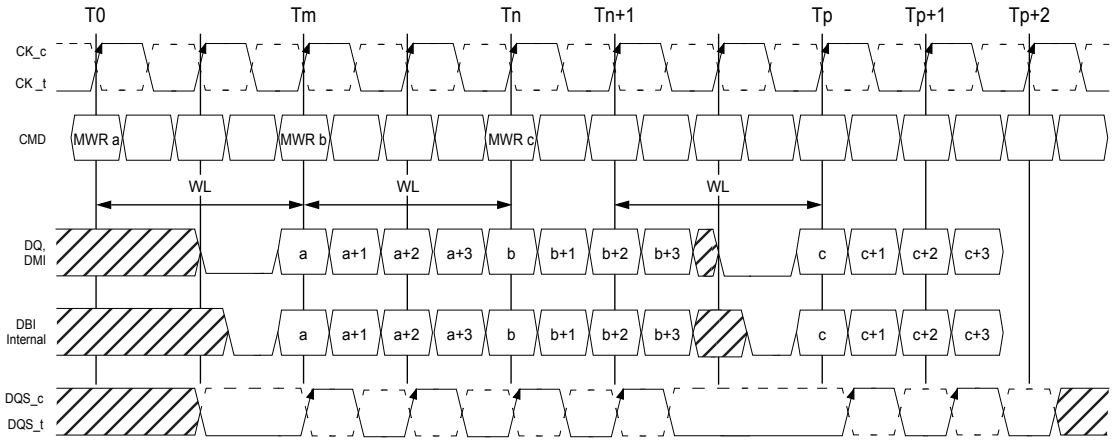


Figure 37 — Masked Write Timing; BL=4; DM Enable (AMD RB13292)

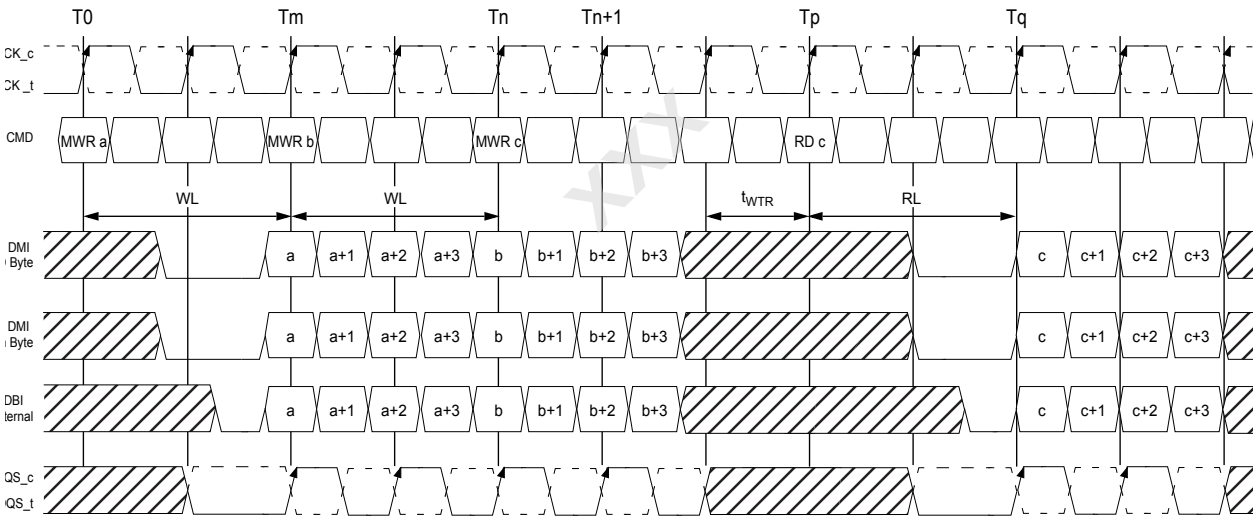


Figure 38 — Masked Write followed by Read Timing; BL=4; DM Enable(AMD RB13292)

4.7.1.3 Read Operation

When a first Read command is registered, WideIO2 device pre-conditions the bus 2 UI prior to Read data with either valid HIGH or LOW. When Read DBIac function is disabled in MR3 bit OP6, WideIO2 device does not pre-condition the bus as described.

Consecutive Read Commands (Seamless and non-seamless):
Once the Read DQ burst is complete, the WideIO2 device tri-states DMI and all DQ output drivers.

For back to back Read operation without a gap, the WideIO2 device uses the last beat of the previous burst for DBIac calculation for second Read operation.

For back to back Read operation with a gap, the WideIO2 device pre-conditions the bus over 2 UI period prior to Read data as if it was a first Read command.

4.7.1.3 Read Operation (cont'd)

Back to back Read operation from a different Rank requires a gap and hence treated the same way as back to back Read operation with a gap from the same rank as noted above.

WideIO2 device resets Masked Write DBI value to Low once it registers Masked Write command after Read command and Read burst is complete.

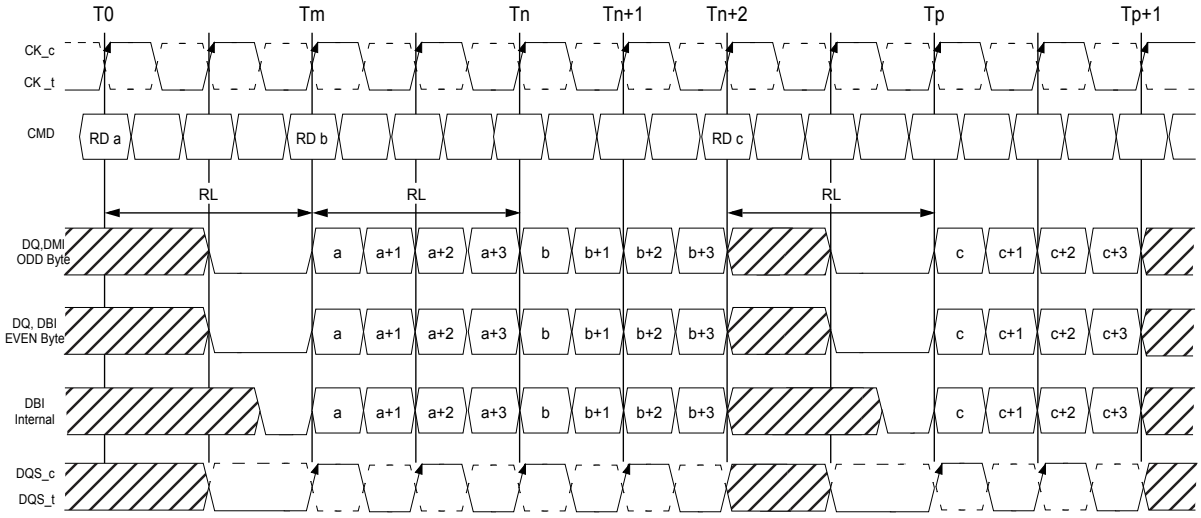


Figure 39 — Figure 3: Read Timing; BL=4 (AMD RB13292)

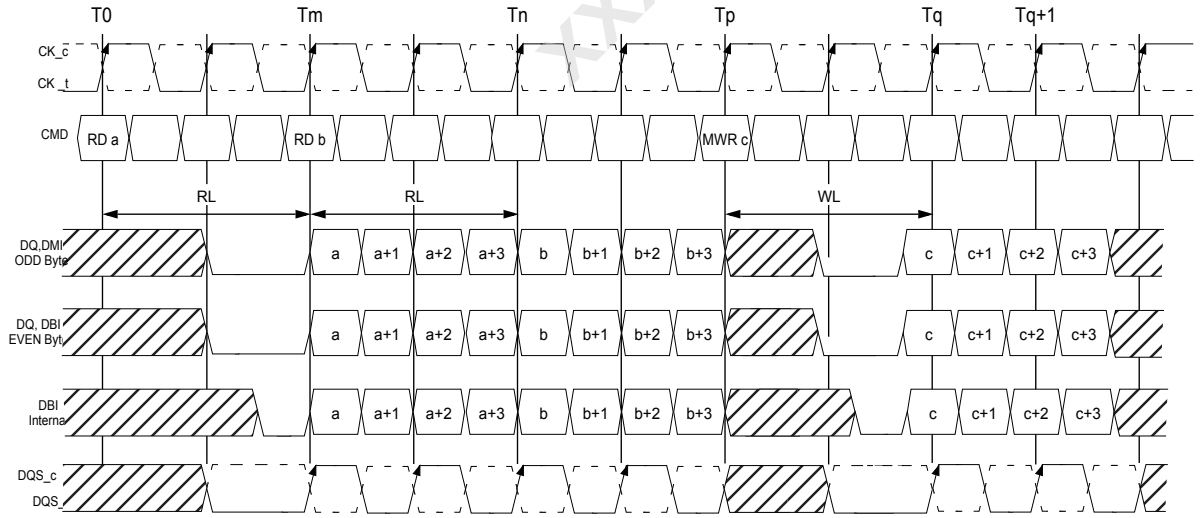


Figure 40 — Read followed by Masked Write Timing; BL=4; DM Enable(AMD RB13292)

4.7.1.4 Mode Register Read (MRR) Operation

When Read DBIac function is enabled, it also applies to MRR command.

4.7.2 DM and DBIac Function Combinations

There are eight possible combinations for WideIO2 device with DM and DBIac function. Table 2 describes the functional behavior for all possible combinations.

Table 22 — Function Behavior of DMI Signal During Write, Masked Write and Read Operation

DM Function	Write DBIac Function	Read DBIac Function	DMI Signal during Write Command	DMI Signal during Masked Write Command	DMI Signal During Read
	Disable	Disable	Notes: 1	Notes: 1, 3	Notes: 2
Disable	Enable	Disable	Notes: 4	Notes: 3	Notes: 2
Disable	Disable	Enable	Notes: 1	Notes: 3	Notes: 5
Disable	Enable	Enable	Notes: 4	Notes: 3	Notes: 5
Enable	Disable	Disable	Notes: 6	Notes: 7	Notes: 2
Enable	Enable	Disable	Notes: 4	Notes: 8	Notes: 2
Enable	Disable	Enable	Notes: 6	Notes: 7	Notes: 5
Enable	Enable	Enable	Notes: 4	Notes: 8	Notes: 5

NOTE 1 DMI signal is a don't care. DMI input receivers are turned OFF.

NOTE 2 DMI output drivers are tunred OFF.

NOTE 3 Masked Write Command is not allowed and is considered an illegal command as DM function is disabled.

NOTE 4 DMI signal is treated as DBI signal and it indicates whether DRAM needs to invert the Write data received on DQs within a byte. The WideIO2 device inverts Write data received on the DQ inputs in case DMI was sampled HIGH, or leaves the Write data non-inverted in case DMI was sampled LOW. See Table 23.

NOTE 5 The WideIO2 device counts the number of DQ signals that are transitioning from previous state. See Read Operation under DBIac section for bus pre-condition. The WideIO2 device inverts Read data and sets DMI HIGH when the number of transitioning data bits within a byte is greater than 4; If previous UI is inverted (DMI High) and the transitions to current UI is equal to 4, then the WideIO2 device will invert the current UI and set DMI HIGH otherwise the WideIO2 device does not invert the Read data and sets DMI LOW.

NOTE 6 The WideIO2 device does not perform any mask operation when it receives Write command. During the Write burst associated with Write command, DMI signal is a don't care and ignored by DRAM.

NOTE 7 The WideIO2 DRAM requires an explicit Masked Write command for all masked write operations. DMI signal is treated as DM signal and it indicates which bit time within the burst is to be masked. When DMI signal is HIGH, DRAM masks that bit time across all DQs associated within a byte. All DQ input signals within a byte are don't care (either HIGH or LOW) when DMI signal is HIGH. When DMI signal is LOW, the WideIO2 DRAM does not perform mask operation and data received on DQ input is written to the array.

NOTE 8 The WideIO2 DRAM requires an explicit Masked Write command for all masked write operations. The WideIO2 device masks the Write data received on the DQ inputs in case DMI is sampled HIGH, DQ[3:0] transition, and DQ[7:4] do not transition (previous DQ[7:0] XOR DQ[7:0] = 0x0F). Otherwise, the WideIO2 device inverts Write data received on the DQ inputs in case DMI is sampled HIGH, or leaves the Write data non-inverted in case DMI is sampled LOW. See Table 24, Table 25 and Table 26 for examples.

Table 23 — DBIac Algorithm during Write Command

	DQ Bus Pre-condition value	Original DQ Pattern Prior to DBIac Algorithm					DQ Pattern on the Bus after Host Performs DBIac Algorithm			
		UI					UI			
		0	1	2	3		0	1	2	3
DQ[0]	0	1	0	0	1		0	0	0	0
DQ[1]	0	0	0	0	1		1	0	0	0
DQ[2]	0	1	0	0	1		0	0	0	0
DQ[3]	0	1	0	0	1		0	0	0	0
DQ[4]	0	0	0	1	0		1	0	1	1
DQ[5]	0	1	0	1	0		0	0	1	1
DQ[6]	0	1	0	1	1		0	0	1	0
DQ[7]	0	1	0	1	1		0	0	1	0
DMI	0						1	0	0	1
Controller/DRAM Internal DBIac State DQ[7:0]	0x00						0x12	0x00	0xF0	0x30

4.7.2 DM and DBIac Function Combinations (cont'd)

Table 24 — DBIac Algorithm During Masked Write Command - Example 1

	DQ Bus Pre-condition value	Original DQ Pattern Prior to DBIac Algorithm					DQ Pattern on the Bus after Host Performs DBIac Algorithm			
		UI					UI			
		0	1	2	3		0	1	2	3
DQ[0]	0	1	0	0	1		1	0	0	0
DQ[1]	0	0	0	0	1		1	0	0	0
DQ[2]	0	1	0	0	1		1	0	0	0
DQ[3]	0	1	0	0	1		1	0	0	0
DQ[4]	0	0	0	1	0		0	0	1	1
DQ[5]	0	1	0	1	0		0	0	1	1
DQ[6]	0	1	0	1	1		0	0	1	0
DQ[7]	0	1	0	1	1		0	0	1	0
DMI	0						1	1	0	1
Data Mask Function		Y	Y	N	N		Y	Y	N	N
Controller/DRAM Internal DBIac State DQ[7:0]	0x00						0x0F	0x00	0xF0	0x30

Table 25 — DBIac Algorithm During Masked Write Command - Example 2

	DQ Bus Pre-condition value	Original DQ Pattern Prior to DBIac Algorithm					DQ Pattern on the Bus after Host Performs DBIac Algorithm			
		UI					UI			
		0	1	2	3		0	1	2	3
DQ[0]	0	1	0	0	1		0	0	0	1
DQ[1]	0	0	0	0	1		1	0	0	1
DQ[2]	0	1	0	0	1		0	0	0	1
DQ[3]	0	1	0	0	1		0	0	0	1
DQ[4]	0	0	0	1	0		1	0	1	1
DQ[5]	0	1	0	1	0		0	0	1	1
DQ[6]	0	1	0	1	1		0	0	1	1
DQ[7]	0	1	0	1	1		0	0	1	1
DMI	0						1	0	0	1
Data Mask Function		N	N	N	Y		N	N	N	Y
Controller/DRAM Internal DBIac State DQ[7:0]	0x00						0x12	0x00	0xF0	0xFF

Table 26 — DBIac Algorithm During Write Command Followed by Masked Write Command

	DQ Bus Pre-condition value	Write Command - Original DQ Pattern Prior to DBIac Algorithm				Masked Write Command - Original DQ Pattern Prior to DBI ac Algorithm					Write Command - DQ Pattern on the Bus after Host Performs DBIac Algorithm				Masked Write Command - DQ Pattern on the Bus after Host Performs DBIac Algorithm			
		UI				UI					UI				UI			
		0	1	2	3	0	1	2	3		0	1	2	3	0	1	2	3
DQ[0]	0	1	0	0	1	1	0	0	1		0	0	0	0	1	0	0	0
DQ[1]	0	0	0	0	1	0	0	0	1		1	0	0	0	1	0	0	0
DQ[2]	0	1	0	0	1	1	0	0	1		0	0	0	0	1	0	0	0
DQ[3]	0	1	0	0	1	1	0	0	1		0	0	0	0	1	0	0	0
DQ[4]	0	0	0	1	0	0	0	1	0		1	0	1	1	1	1	1	1
DQ[5]	0	1	0	1	0	1	0	1	0		0	0	1	1	1	1	1	1
DQ[6]	0	1	0	1	1	1	0	1	1		0	0	1	0	0	0	1	0
DQ[7]	0	1	0	1	1	1	0	1	1		0	0	1	0	0	0	1	0
DMI	0										1	0	0	1	1	1	0	1
Data Mask Function		N/A				Y	Y	N	N		N/A				Y	Y	N	N
Controller/DRAM Internal DBIac State DQ[7:0]	0x00										0x12	0x00	0xF0	0x30	0x3F	0x30	0xF0	0x30

4.8 Refresh Command

The REFRESH command is initiated with CS_n LOW, CA0 LOW, CA1 LOW,CA2 HIGH with CKE high at the rising edge of the clock. Per-bank REFRESH is initiated with CA4 LOW at the rising edge of the clock. All-bank REFRESH is initiated with CA4 HIGH at the rising edge of the clock.

A per-bank REFRESH command (REFpb) performs a per-bank REFRESH operation to each of the banks before it issues second one to any of the banks. A controller that does not use out of order per bank commands simply delivers Directed Per Bank Refresh commands with a round robin bank address. (0-1-2-3-4-5-6-7-0-1-....) The bank count is synchronized between the controller and the SDRAM by resetting the bank count to zero. Synchronization can occur upon issuing one of following three events. (RESET signal, self refresh exit, all bank refresh) Some WideIO2 implementations may not require synchronization. Bank addressing for the per-bank REFRESH count is the same as established for the single-bank PRECHARGE command. A bank must be idle before it can be refreshed. The controller must track the bank being refreshed by the per-bank REFRESH command.

The REFpb command must not be issued to the device until the following conditions are met (see Table 27):

. t_{RFCab} has been satisfied after the prior REFab command

. t_{RFCpb} has been satisfied after the prior REFpb command

. t_{RP} has been satisfied after the prior PRECHARGE command to that bank

. t_{RRD} has been satisfied after the prior ACTIVATE command (if applicable, for example after activating a row in a different bank than the one affected by the REFpb command).

The target bank is inaccessible during per-bank REFRESH cycle time (t_{RFCpb}), however, other banks within the device are accessible and can be addressed during the cycle. During the REFpb operation, any of the banks other than the one being refreshed can be maintained in an active state or accessed by a READ or a WRITE command. When the per-bank REFRESH cycle has completed, the affected bank will be in the idle state.

After issuing REFpb, these conditions must be met (Table 27):

. t_{RFCpb} must be satisfied before issuing a REFab command

. t_{RFCpb} must be satisfied before issuing an ACTIVATE command to the same bank

. t_{RRD} must be satisfied before issuing an ACTIVATE command to a different bank

. t_{RFCpb} must be satisfied before issuing another REFpb command.

An all-bank REFRESH command (REFab) issues a REFRESH command to all banks. All banks must be idle when REFab is issued (for instance, by issuing a PRECHARGE-all command prior to issuing an all-bank REFRESH command). REFab also synchronizes the bank count between the controller and the SDRAM to zero. The REFab command must not be issued to the device until the following conditions have been met (Table 27):

. t_{RFCab} has been satisfied following the prior REFab command

. t_{RFCpb} has been satisfied following the prior REFpb command

. t_{RP} has been satisfied following the prior PRECHARGE commands.

When an all-bank refresh cycle has completed, all banks will be idle. After issuing REFab:

. t_{RFCab} latency must be satisfied before issuing an ACTIVATE command

. t_{RFCab} latency must be satisfied before issuing a REFab or REFpb command.

4.8 Refresh Command (cont'd)

Table 27 — REFRESH Command Scheduling Separation Requirements

Symbol	Minimum Delay From…	To…	Notes
t_{RFCab}	REFab	REFab	
		ACTIVATE command to any bank	
		REFpb	
t_{RFCpb}	REFpb	REFab	
		ACTIVATE command to same bank as REFpb	
		REFpb	
t_{RRD}	REFpb	ACTIVATE command to a different bank than REFpb	
	ACTIVATE	REFpb	
		ACTIVATE command to a different bank than the prior ACTIVATE command	

NOTE 1 A bank must be in the idle state before it is refreshed, so following an ACTIVATE command REFab is prohibited; REFpb is supported only if it affects a bank that is in the idle state.

In general, an all bank refresh command needs to be issued to the WideIO2 SDRAM regularly every tREFI × REFM interval(REFM is the refresh multiplier specified by MR4). To allow for improved efficiency in scheduling and switching between tasks, some flexibility in the absolute refresh interval is provided for postponing and pulling-in refresh command. A maximum of 8 Refresh commands can be postponed during operation of the WideIO2 SDRAM, meaning that at no point in time more than a total of 8 Refresh commands are allowed to be postponed and maximum number of pulled-in or postponed REF command is dependent on refresh rate. It is described in the Table 28 and Table 29. In case that 8 Refresh commands are postponed in a row, the resulting maximum interval between the surrounding Refresh commands is limited to 9 × tREFI × REFM (see Figure 41). A maximum of 8 additional Refresh commands can be issued in advance ("pulled in") , with each one reducing the number of regular Refresh commands required later by one. Note that pulling in more than 8, depending on Refresh mode, Refresh commands in advance does not further reduce the number of regular Refresh commands required later, so that the resulting maximum interval between two surrounding Refresh commands is limited to 9 × tREFI × REFM . At any given time, a maximum of 16 REF commands can be issued within 2 x tREFI × REFM

And for per bank refresh, a maximum 8 x 8 per bank refresh commands can be postponed or pulled in for scheduling efficiency. At any given time, a maximum of 2 x 8 x 8 per bank refresh commands can be issued within 2 x tREFI × REFM.

4.8 Refresh Command (cont'd)

Table 28 — Legacy REFRESH Command Timing Constraints

MR4 OP[2:0]	Refresh rate	Max. No. of pulled-in or postponed REFab	Max. Interval between two REFab	Max. No. of REFab within max(2xtREFI x refresh rate multiplier, 16xtRFC)	Per-bank Refresh
000_B	Low Temp. Limit	N/A	N/A	N/A	N/A
001_B	4× tREFI	8	9 × 4 × tREFI	16	1/8 of REFab
010_B	2× tREFI	8	9 × 2 × tREFI	16	1/8 of REFab
011_B	1× tREFI	8	9 × tREFI	16	1/8 of REFab
100_B	0.5× tREFI	8	9 × 0.5 × tREFI	16	1/8 of REFab
101_B	0.25× tREFI	8	9 × 0.25 × tREFI	16	1/8 of REFab
110_B	0.25× tREFI	8	9 × 0.25 × tREFI	16	1/8 of REFab
111_B	High Temp. Limit	N/A	N/A	N/A	N/A

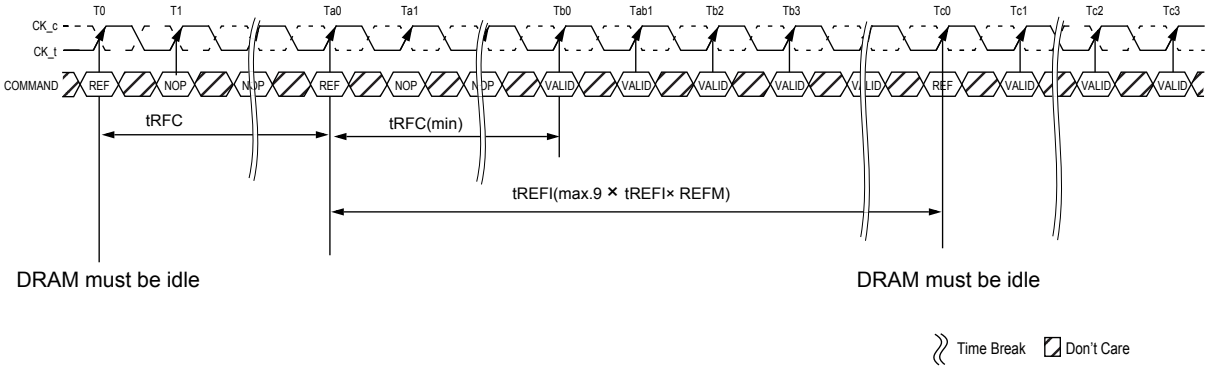
Table 29 — Modified REFRESH Command Timing Constraints

MR4 OP[2:0]	Refresh rate	Max. No. of pulled-in or postponed REFab	Max. Interval between two REFab	Max. No. of REFab within max(2xtREFI x refresh rate multiplier, 16xtRFC)	Per-bank Refresh
000_B	Low Temp. Limit	N/A	N/A	N/A	N/A
001_B	4× tREFI	2	3 × 4 × tREFI	16	1/8 of REFab
010_B	2× tREFI	4	5 × 2 × tREFI	16	1/8 of REFab
011_B	1× tREFI	8	9 × tREFI	16	1/8 of REFab
100_B	0.5× tREFI	8	9 × 0.5 × tREFI	16	1/8 of REFab
101_B	0.25× tREFI	8	9 × 0.25 × tREFI	16	1/8 of REFab
110_B	0.25× tREFI	8	9 × 0.25 × tREFI	16	1/8 of REFab
111_B	High Temp. Limit	N/A	N/A	N/A	N/A

NOTE 1 For any thermal transition phase where Refresh mode is transitioned to either 2× tREFI or 4× tREFI, DRAM will support the previous postponed refresh requirement provided the number of postponed refreshes is monotonically reduced to meet the new requirement. However, the pulled-inrefresh commands in previous thermal phase are not applied in new thermal phase.Entering new thermal phase the controller must count the number of pulled-in refresh commands as zero, regardless of remaining pulled-in refresh commands in previous thermal phase.

NOTE 2 WideIO2 devices are refreshed properly if memory controller issues refresh commands with same or shorter refresh rate than reported by MR4 OP[2:0]. If shorter refreshperiod is applied, the corresponding requirements from Table apply. For example, when MR4 OP[2:0]=001B, controller can be in any refresh rate from 4xtREFI to 0.25xtREFI. When MR4 OP[2:0]=010B, the only prohibited refresh rate is 4xtREFI.

4.8 Refresh Command (cont'd)



NOTE 1 Time interval between two Refresh commands may be extended to a maximum of 9 × tREFI × REFM.

Figure 41 — Refresh Command Timing

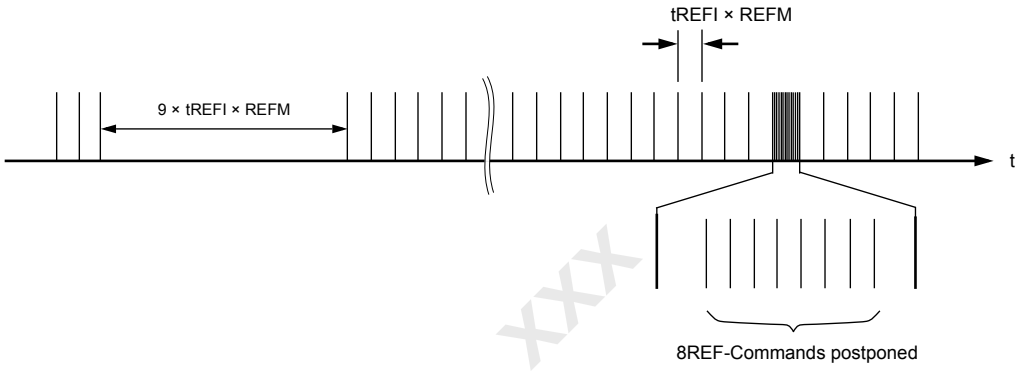


Figure 42 — Postponing Refresh Commands

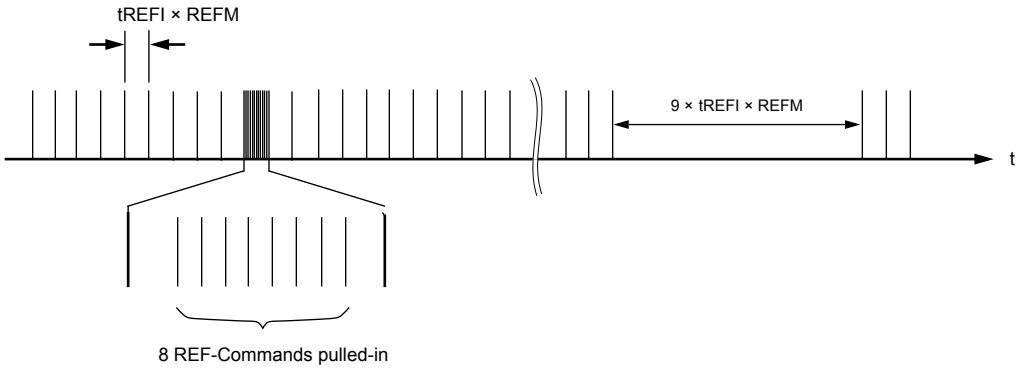


Figure 43 — Pulling-in Refresh Commands

4.8.1 Refresh Requirements

4.8.1.1 Minimum number of REFRESH commands

WideIO2 requires a minimum number, R, of REFRESH (REFab) commands within any rolling refresh window (tREFW = 32 ms @ MR4[2:0] = 011 or TC @ 85 °C). For tREFW and tREFI refresh multipliers at different MR4 settings, refer to the MR4 definition.

When using per-bank REFRESH, a REFab command can be replaced by a full cycle of eight REFpb commands.

4.8.1.2 REFRESH Requirements and SELF REFRESH

Self refresh mode may be entered with a maximum of eight refresh commands being postponed. After exiting self-refresh mode with one or more refresh commands postponed, additional refresh commands may be postponed to the extent that the total number of postponed refresh commands (before and after the self refresh) will never exceed eight. During self-refresh mode, the number of postponed or pulled-in REF commands does not change.

The use of self refresh mode introduces the possibility that an internally timed refresh event can be missed when CKE is raised for exit from self refresh mode. Upon exit from self refresh, the WideIO2 SDRAM requires a minimum of one extra refresh command before it is put back into self refresh mode.

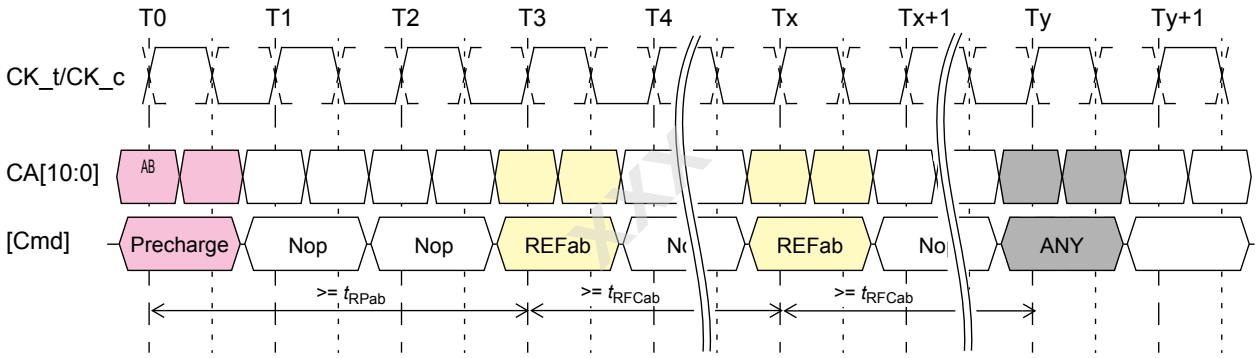
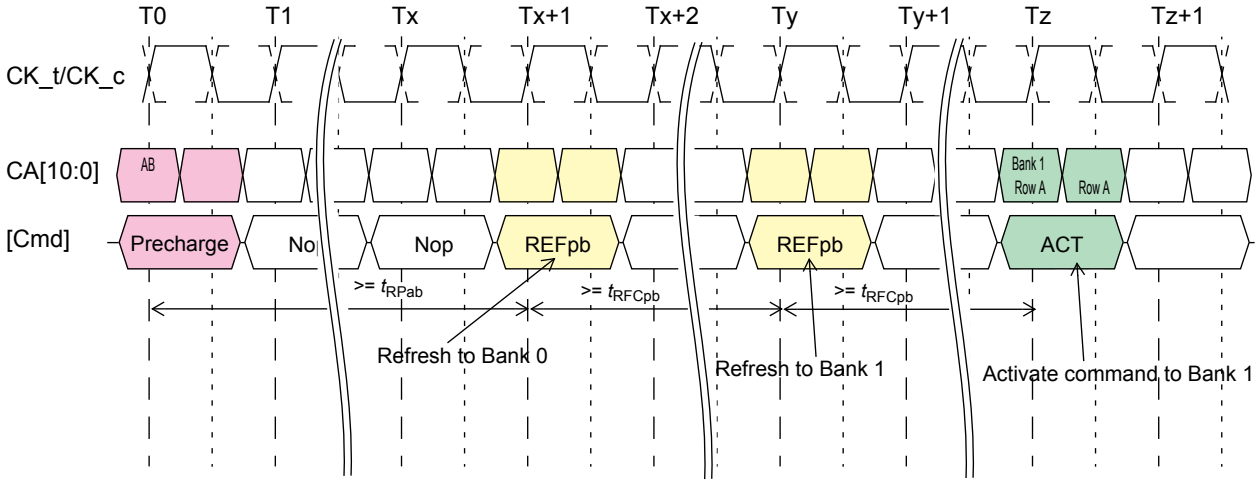


Figure 44 — All-Bank REFRESH Operation



NOTE 1 In the beginning of this example, the REFpb bank is pointing to bank 0.

NOTE 2 Operations to banks other than the bank being refreshed are supported during the tRFCpb period.

Figure 45 — Per-Bank REFRESH Operation

4.8.1.2 REFRESH Requirements and SELF REFRESH(cont'd)

Table 30 — Refresh Requirements

Parameter	Symbol	4ch Die	8ch Die	Unit
Refresh Cycle Time (all bank)	$tRFC_{ab}$ 8Gb	180	180	ns
Refresh Cycle Time (all bank)	$tRFC_{ab}$ 12Gb	TBD	TBD	ns
Refresh Cycle Time (all bank)	$tRFC_{ab}$ 16Gb	TBD	TBD	ns
Average periodic refresh interval (Tcase <= 85oC)	tREFI	3.9	3.9	us
Required number of refresh commands in Refresh Time tREFW	R	8,192	8,192	-
Refresh Time	tREFW	32	32	ms
Refresh Cycle Time (per bank)	$tRFC_{pb}$ 8Gb	90	90	ns
Refresh Cycle Time (per bank)	$tRFC_{pb}$ 12Gb	TBD	TBD	ns
Refresh Cycle Time (per bank)	$tRFC_{pb}$ 16Gb	TBD	TBD	ns

4.9 Power Down

Power-down is entered synchronously when CKE is registered LOW and CS_n is HIGH at the rising edge of clock. CKE must not go LOW while MRR, MRW, READ, or WRITE operations are in progress. CKE can go LOW while any other operations such as row activation, PRECHARGE, auto precharge, or REFRESH are in progress, but the power-down IDD specification will not be applied until such operations are complete. Power-down entry and exit are shown in Figure 46 through Figure 57.

Entering power-down deactivates the input and output buffers, excluding CKE. To ensure that there is enough time to account for internal delay on the CKE signal path, two NOP commands or Deselects are required after CKE is driven LOW. This
timing period is defined as t_{CPDED}. CKE LOW will result in deactivation of input receivers after t_{CPDED} has expired. In power-down mode, CKE must be held LOW; all other input signals are "Don't Care." CKE LOW must be maintained until $t_{CKE,min}$ is satisfied.

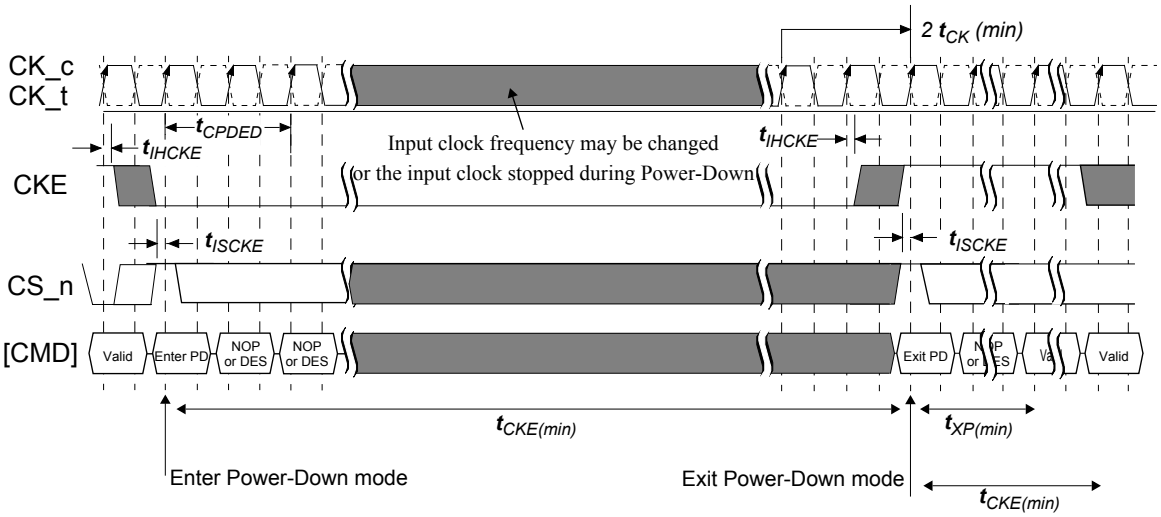
V_{DDQ} can be turned off during power-down. Prior to exiting power-down, V_{DDQ} must be within its respective minimum/maximum operating ranges.

No refresh operations are performed in power-down mode. The maximum duration in power-down mode is only limited by the refresh requirements outlined in the Refresh command section.

The power-down state is exited when CKE is registered HIGH. The controller must drive CS_n HIGH in conjunction with CKE HIGH when exiting the power-down state. CKE HIGH must be maintained until $t_{CKE,min}$ is satisfied. A valid, executable command can be applied with power-down exit latency t_{XP} after CKE goes HIGH. During tXP, NOP or DES commands are required. Power-down exit latency is defined in the AC timing parameter table.

If power-down occurs when all banks are idle, this mode is referred to as idle power-down; if power-down occurs when there is a row active in any bank, this mode is referred to as active power-down.

4.9 Power Down (cont'd)



NOTE 1 Input clock frequency can be changed or the input clock can be stopped or floated during power-down, provided that upon exiting power-down, the clock is stable and within specified limits for a minimum of 2 clock cycles prior to power-down exit and the clock frequency is between the minimum and maximum specified frequency for the speed grade in use.

Figure 46 — Basic Power-Down Entry and Exit Timing

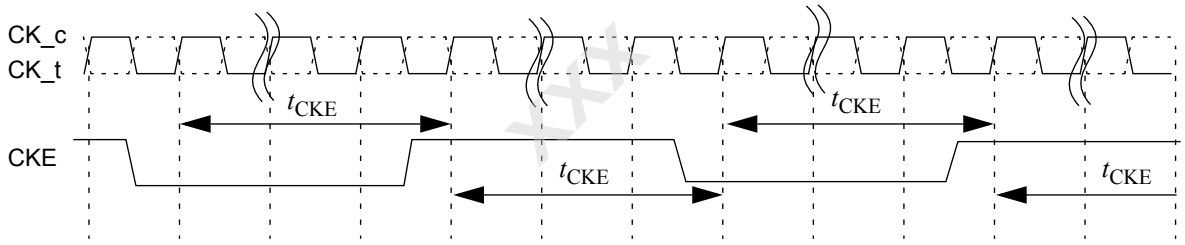
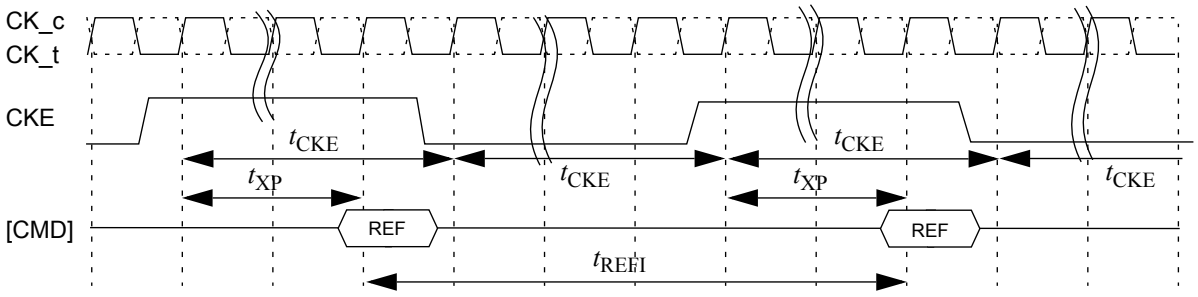


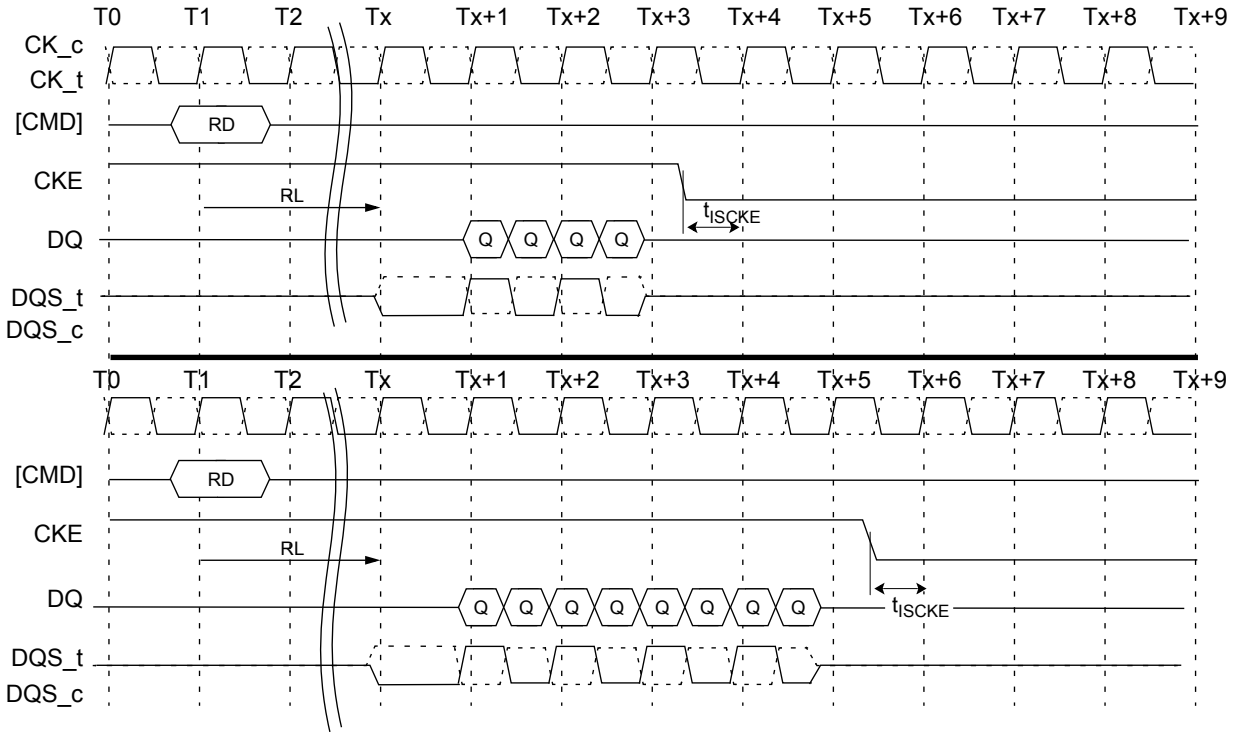
Figure 47 — CKE-Intensive Environment



NOTE 1 The pattern shown can repeat over an extended period of time. With this pattern, all AC and DC timing and voltage specifications with temperature and voltage drift are ensured.

Figure 48 — REFRESH-to-REFRESH Timing in CKE-Intensive Environments

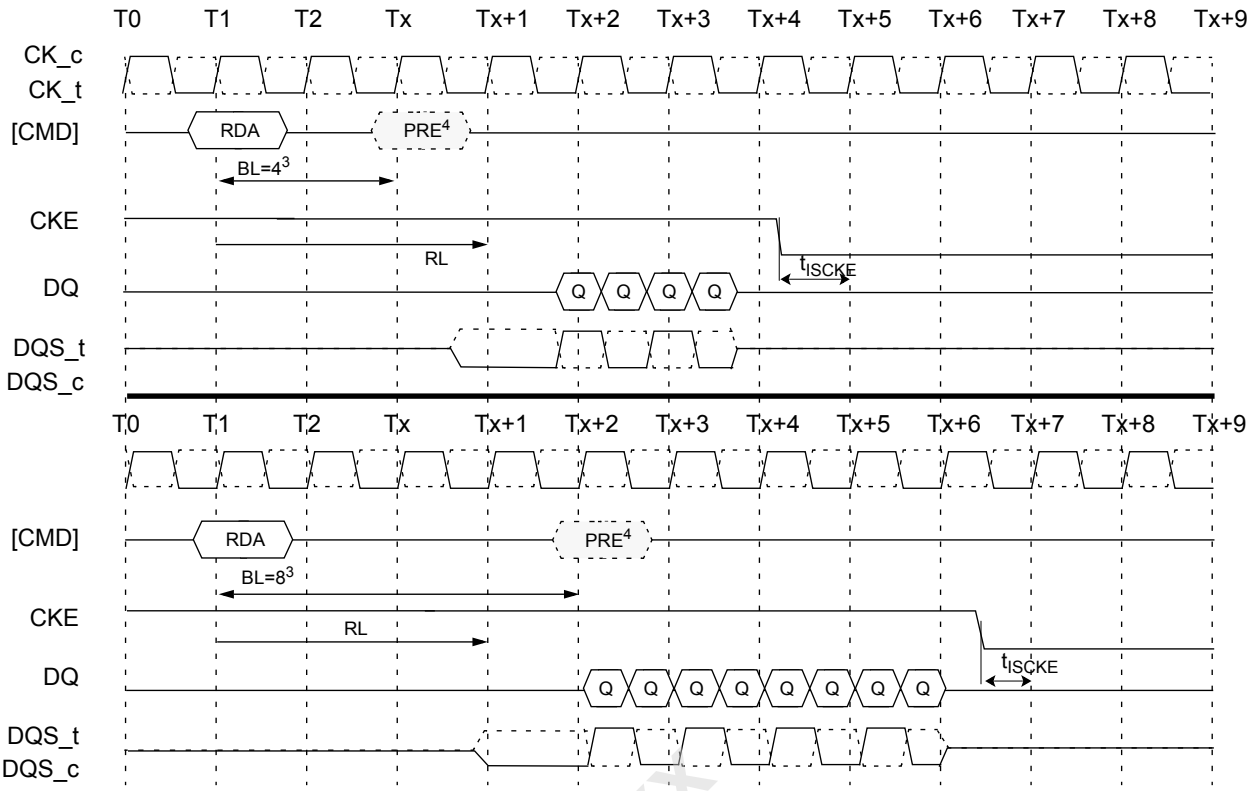
4.9 Power Down (cont'd)



NOTE 1 CKE must be held HIGH until the end of the burst operation.
NOTE 2 CKE can be registered LOW RL + RU($t_{DQSCK(MAX)}/t_{CK}$) + BL/2 + 1 clock cycles after the clock on which the READ command is registered.

Figure 49 — READ to Power-Down Entry

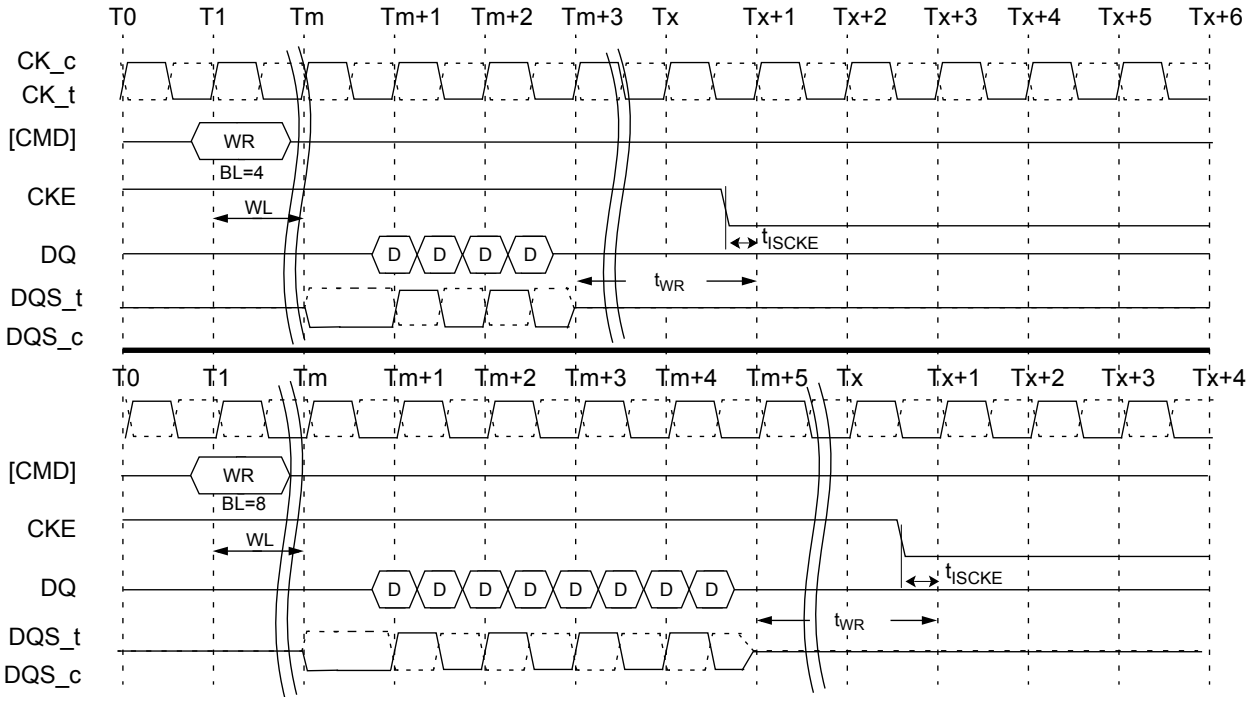
4.9 Power Down (cont'd)



NOTE 1 CKE must be held HIGH until the end of the burst operation.
NOTE 2 CKE can be registered LOW at RL + RU(t_{DQSCK}/t_{CK})+ BL/2 + 1 clock cycles after the clock on which the READ command is registered.
NOTE 3 BL/2 with t_{RTP} = 7.5ns and $t_{RAS(MIN)}$ is satisfied.
NOTE 4 Start internal PRECHARGE.

Figure 50 — READ with Auto Precharge to Power-Down Entry

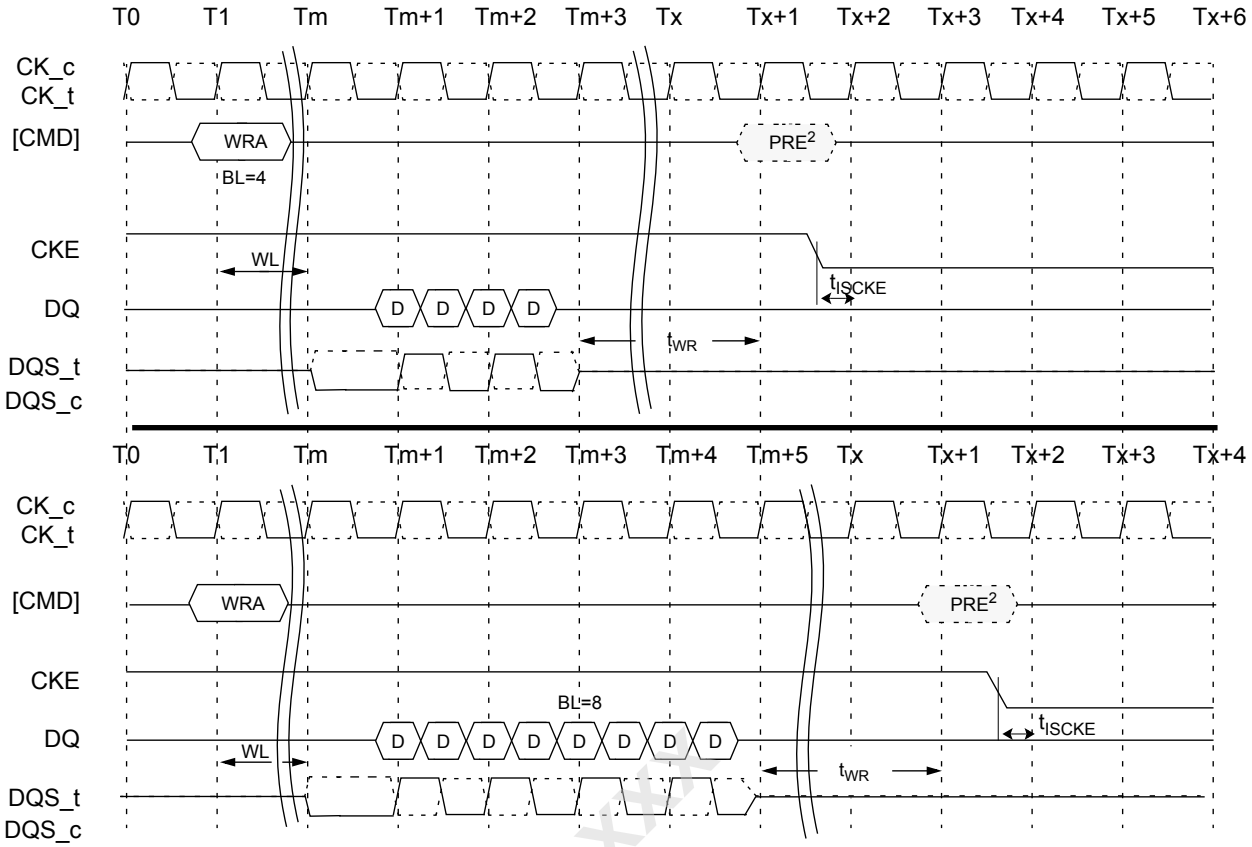
4.9 Power Down (cont'd)



NOTE 1 CKE can be registered LOW WL + 1 + BL/2 + RU(t_{WR}/t_{CK}) clock cycles after the clock on which the WRITE command is registered.

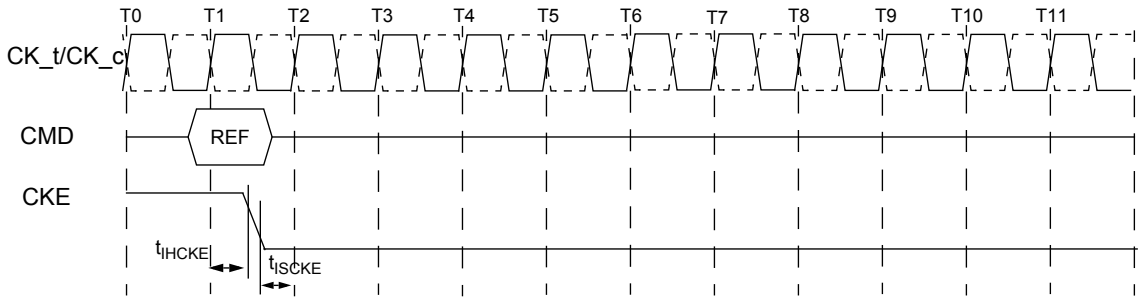
Figure 51 — WRITE to Power-Down Entry

4.9 Power Down (cont'd)



NOTE 1 CKE can be registered LOW WL+1+BL/2+RU(t_{WR}/t_{CK})+1 clock cycles after the WRITE command is registered.
NOTE 2 Start internal PRECHARGE.

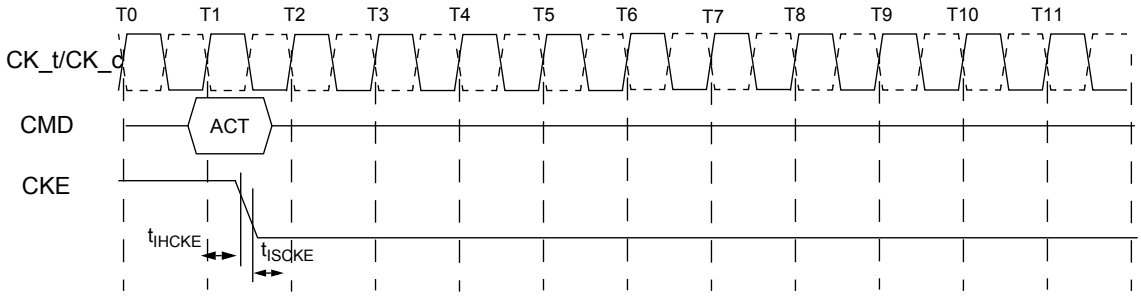
Figure 52 — WRITE with Auto Precharge to Power-Down Entry



NOTE 1 CKE can go LOW t_{IHCKE} after the clock on which the REFRESH command is registered.

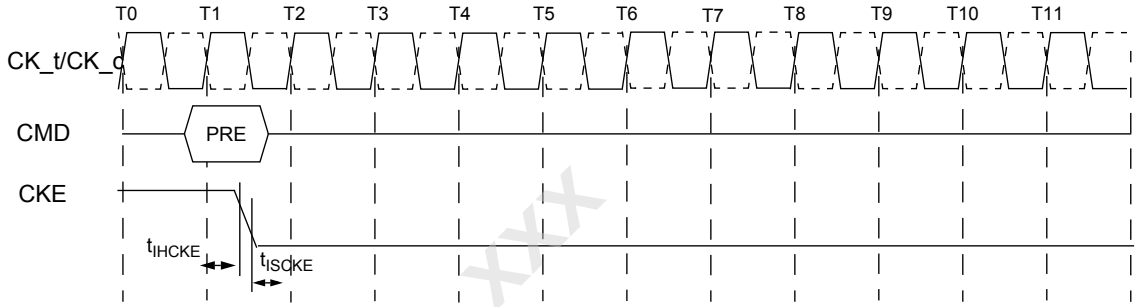
Figure 53 — REFRESH Command to Power-Down Entry

4.9 Power Down (cont'd)



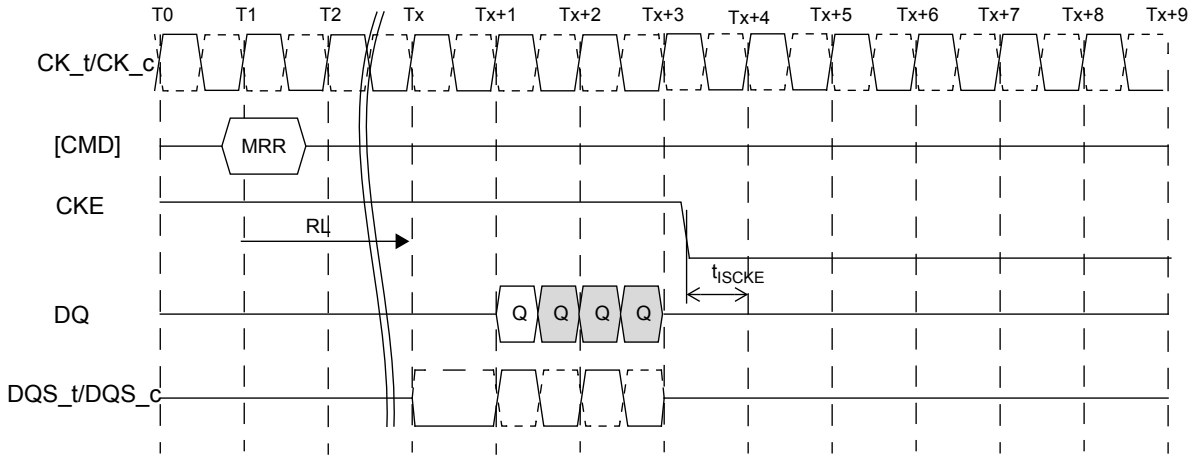
NOTE 1 CKE can go LOW t_{IHCKE} after the clock on which the ACTIVATE command is registered.

Figure 54 — ACTIVATE Command to Power-Down Entry



NOTE 1 CKE can go LOW t_{IHCKE} after the clock on which the PRECHARGE command is registered.

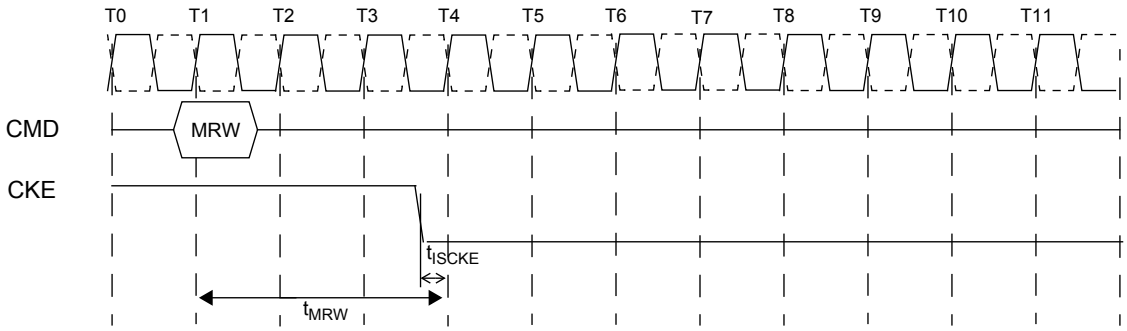
Figure 55 — PRECHARGE Command to Power-Down Entry



NOTE 1 CKE can be registered LOW RL + RU(tDQSCK(MAX)/tCK)+ 4/2 + 1 clock cycles after the clock on which the MRR command is registered.
NOTE 2 CKE should be held high until the end of the burst operation.

Figure 56 — MRR to Power-Down Entry

4.9 Power Down (cont'd)



NOTE 1 CKE can be registered LOW tMRW after the clock on which the MRW command is registered.

Figure 57 — MRW to Power-Down Entry

4.10 Self Refresh Operation

The Self Refresh command can be used to retain data in the WideIO2 SDRAM, even if the rest of the system is powered down. When in the Self Refresh mode, the SDRAM retains data without external clocking. The device has a built-in timer to accommodate Self Refresh operation. The Self Refresh Command is defined by having CKE LOW, CS_n LOW, CA0 LOW, CA1 LOW, and CA2 HIGH at the rising edge of the clock. CKE must be HIGH during the previous clock cycle. CKE must not go LOW while MRR, MRW, READ, or WRITE operations are in progress. To ensure that there is enough time to account for internal delay on the CKE signal path, two NOP commands are required after CKE is driven LOW, this timing period is defined as tCPDED. CKE LOW will result in deactivation of input receivers after tCPDED has expired. Once the command is registered, CKE must be held LOW to keep the device in Self Refresh mode.

WideIO2 SDRAM devices can operate in Self Refresh in both the standard or elevated temperature ranges. WideIO2 devices will also manage Self Refresh power consumption when the operating temperature changes, lower at low temperatures and higher at high temperatures.

Once the SDRAM has entered Self Refresh mode, all of the external signals except CKE, are "don't care". For proper self refresh operation, power supply pins (VDD1 and VDD2) must be at valid levels. VDDQ may be turned off(floated) during Self-Refresh. Prior to exiting Self-Refresh, VDDQ must be within specified limits. The SDRAM initiates a minimum of one all-bank refresh command internally within tCKESR period once it enters Self Refresh mode. The clock is internally disabled during Self Refresh Operation to save power. The minimum time that the SDRAM must remain in Self Refresh mode is tCKESR,min. The user may change the external clock frequency or halt the external clock one clock after Self Refresh entry is registered; however, the clock must be restarted and stable before the device can exit Self Refresh operation.

The procedure for exiting Self Refresh requires a sequence of commands. First, the clock shall be stable and within specified limits for a minimum of 2 tCK prior to the positive clock edge that registers CKE HIGH. Once Self Refresh Exit is registered, a delay of at least tXSR must be satisfied before a valid command can be issued to the device to allow for any internal refresh in progress. CKE must remain HIGH for the entire Self Refresh exit period tXSR for proper operation. NOP commands must be registered on each positive clock edge during the Self Refresh exit interval tXSR.

The use of Self Refresh mode introduces the possibility that an internally timed refresh event can be missed when CKE is raised for exit from Self Refresh mode. Upon exit from Self Refresh, it is required that at least one REFRESH command (8 per-bank or 1 all-bank) is issued before entry into a subsequent Self Refresh.

4.10 Self Refresh Operation (cont'd)

NOTE 1 Input clock frequency may be changed or can be stopped or floated during self-refresh, provided that upon exiting self-refresh, the clock is stable and within specified limits for a minimum of 2 clocks of stable clock are provided and the clock frequency is between the minimum and maximum frequency for the speed grade in use.

NOTE 2 Device must be in the "All banks idle" state prior to entering Self Refresh mode.

NOTE 3 t_{XSR} begins at the rising edge of the clock after CKE is driven HIGH.

NOTE 4 A valid command may be issued only after t_{XSR} is satisfied. "DES or NOP commands shall be issued during t_{XSR}.

Figure 58 — WideIO2: Self-Refresh Operation

4.10.1 Self-Refresh Abort

If Self-Refresh Abort, MR2 OP[7], is enabled then DRAM aborts any ongoing refresh during Self-Refresh exit and does not increment the internal refresh counter. Controller can issue valid commands after a delay of tXSR_abort instead of tXSR. Self-Refresh Abort is only applicable to 12Gb densities and above.

The value of tXSR_abort(min) is defined in Table 53, AC timing parameter table.

Upon exit from Self-Refresh mode, DRAM requires a minimum of one extra REFRESH command before entry into a subsequent Self-Refresh mode. This requirement remains the same irrespective of the setting of the MR bit for Self-Refresh Abort.

Feature	Access	Addressing	Values
Self-Refresh Abort	Write only	OP[7]	$\mathbf{0_B}$: Self-Refresh Abort disabled $\mathbf{1_B}$: Self-Refresh Abort enabled

4.11 Partial Array Self Refresh

4.11.1 PASR Bank Masking

The WideIO2 SDRAM has eight banks (4 channel with 25.6 GB/s) or four banks(8 channel with 51.2 GB/s). Each bank of an WideIO2 SDRAM can be independently configured whether a self refresh operation is taking place. One mode register unit of 8 bits, accessible via MRW command, is assigned to program the bank masking status of each bank up to 8 banks. For bank masking bit assignments, see 3.4.12, Mode Register 11.

The mask bit to the bank controls a refresh operation of entire memory within the bank. If a bank is masked via MRW, a refresh operation to the entire bank is blocked and data retention by a bank is not guaranteed in self refresh mode. To enable a refresh operation to a bank, a coupled mask bit has to be programmed, "unmasked". When a bank mask bit is unmasked, a refresh to a bank is determined by the programmed status of segment mask bits, which is decribed in the following chapter.

4.11.2 PASR Segment Masking

A segment masking scheme may be used in lieu of or in combination with the bank masking scheme in WideIO2 SDRAM. WideIO2 devices utilize eight segments per bank. For segment masking bit assignments, see Mode Register 12.

For those refresh-enabled banks, a refresh operation to the address range which is represented by a segment is blocked when the mask bit to this segment is programmed, “masked”. Programming of segment mask bits is similar to the one of bank mask bits. Eight segments are used as listed in Mode Register 12, see 3.4.13. One mode register unit is used for the programming of segment mask bits up to 8 bits. One more mode register unit may be reserved for future use. Programming of bits in the reserved registers has no effect on the device operation.

Table 31 — Example of Bank and Segment Masking use in WIO2 devices

	Segment Mask (MR12)	Bank 0	Bank 1	Bank 2	Bank 3	Bank 4	Bank 5	Bank 6	Bank 7
Bank Mask (MR11)		0	1	0	0	0	0	0	1
Segment 0	0		M						M
Segment 1	0		M						M
Segment 2	1	M	M	M	M	M	M	M	M
Segment 3	0		M						M
Segment 4	0		M						M
Segment 5	0		M						M
Segment 6	0		M						M
Segment 7	1	M	M	M	M	M	M	M	M

NOTE 1 Table 31 illustrates an example of an 8-bank WideIO2 device, when a refresh operation to bank 1 and bank 7, as well as segment 2 and segment 7 are masked.

4.12 TRR Mode - Target Row Refresh

A WideIO2 SDRAM's row has a limited number of times a given row can be accessed within a refresh period (tREFW * 2) prior to requiring adjacent rows to be refreshed. The Maximum Activate Count (MAC) is the maximum number of activates that a single row can sustain within a refresh period before the adjacent rows need to be refreshed. The row receiving the excessive actives is the Target Row (TRn), the adjacent rows to be refreshed are the victim rows. When the MAC limit is reached on TRn, either the WideIO2 SDRAM receive all (R * 2) Refresh Commands before another row activate is issued, or the WideIO2 SDRAM should be placed into Targeted Row Refresh (TRR) mode. The TRR Mode will refresh the rows adjacent to the TRn that encountered tMAC limit.

There could be a maximum of two target rows to a victim row in a bank. The cumulative value of the activates from the two target rows on a victim row in a bank should not exceed MAC value as well.

Setting MR14 [OP7=1] enables TRR Mode and setting MR14 [OP7=0] disables TRR Mode. MR14 [OP6:OP4] defines which bank (BAn) the target row is located in.

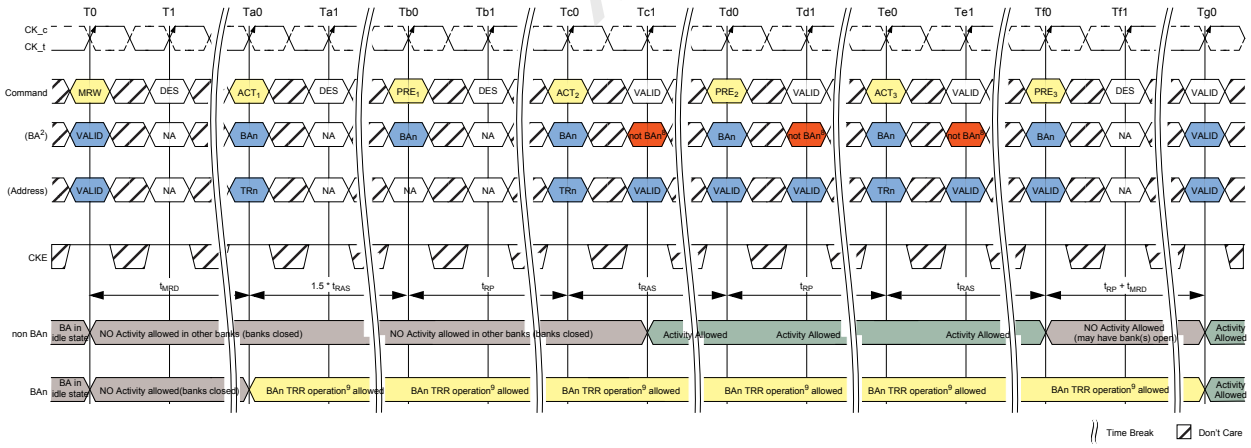
The TRR mode must be disabled during initialization as well as any other WideIO2 SDRAM calibration modes. The TRR mode is entered from a DRAM Idle State, once TRR mode has been entered, no other Mode Register commands are allowed until TRR mode is completed, except setting MR14 [OP7=0] to interrupt and reissue the TRR mode is allowed .

When enabled; TRR Mode is self-clearing; the mode will be disabled automatically after the completion of defined TRR flow; after the 3rd BAn precharge has completed plus tMRD. Optionally the TRR mode can also be exited via another MRS command at the completion of TRR by setting MR14 [OP7=0]; if the TRR is exited via another MRS command, the value written to MR14 [OP6:OP4] are don’t cares.

4.12 TRR Mode - Target Row Refresh (cont'd)

TRR Mode Operation

1. The timing diagram in Figure 59 depicts TRR mode. The following steps must be performed when TRR mode is enabled. This mode requires all three ACT (ACT1, ACT2 and ACT3) and three corresponding PRE commands (PRE1, PRE2 and PRE3) to complete TRR mode. A Precharge All (PREA) commands issued while WIO2 SDRAM is in TRR mode will also perform precharge to BAn and counts towards a PREn command.
2. Prior to issuing the MRW command to enter TRR mode, the SDRAM should be in the idle state. A MRW command must be issued with MR14 [OP7=1] and MR14 [OP6:OP4] defining the bank in which the targeted row is located. All other MR14 bits should remain unchanged.
3. No activity is to occur in the DRAM until tMRD has been satisfied. Once tMRD has been satis-fied, the only commands to BAn allowed are ACT and PRE until the TRR mode has been completed.
4. The first ACT to the BAn with the TRn address can now be applied, no other command is allowed at this point. All other banks must remain inactive from when the first BAn ACT command is issued until [(1.5 * tRAS) + tRP] is satisfied.
5. After the first ACT to the BAn with the TRn address is issued, a PRE to BAn is to be issued (1.5 * tRAS) later; and then followed tRP later by the second ACT to the BAn with the TRn address. Once the 2nd activate to the BAn is issued, nonBAn banks are allowed to have activity.
6. After the second ACT to the BAn with the TRn address is issued, a PRE to BAn is to be issued tRAS later and then followed tRP later by the third ACT to the BAn with the TRn address.
7. After the third ACT to the BAn with the TRn address is issued, a PRE to BAn would be issued tRAS later; and once the third PRE has been issued, nonBAn bank groups are not allowed to have activity until TRR mode is exited. The TRR mode is completed once tRP plus tMRD is satisfied.
8. TRR mode must be completed as specified to guarantee that adjacent rows are refreshed. Any-time the TRR mode is interrupted and not completed, the interrupted TRR Mode must be cleared and then subsequently performed again. To clear an interrupted TRR mode, an MR14 change is required with setting MR14 [OP7=0], MR14 [OP6:OP4] are don't care, followed by three PRE to BAn, tRP time in between each PRE command. The complete TRR sequence (Steps 2-7) must be then re-issued and completed to guarantee that the adjacent rows are refreshed.
9. Refresh command to the WIO2 SDRAM or entering Self-Refresh mode is not allowed while the DRAM is in TRR mode.



NOTE 1 TRn is targeted row.

NOTE 2 Bank group BAn represents the bank in which the targeted row is located.

NOTE 3 TRR mode self-clears after tMRD + tRP measured from 3rd BAn precharge PRE3 at clock edge Tg0.

NOTE 4 TRR mode or any other activity can be re-engaged after tRP + tMRD from 3rd BAn precharge PRE3. PRE_ALL also counts if issued instead of PREn. TRR mode is cleared by DRAM after PRE3 to the BAn bank.

NOTE 5 Activate commands to BAn during TRR mode do not provide refreshing support, i.e. the Refresh counter is unaffected.

NOTE 6 The DRAM must restore the degraded row(s) caused by excessive activation of the targeted row (TRn) neccessary to meet refresh requirements.

NOTE 7 A new TRR mode must wait tMRD+tRP time after the third precharge.

NOTE 8 BAn may not be used with any other command.

NOTE 9 ACT and PRE are the only allowed commands to BAn during TRR Mode.

NOTE 10 Refresh commands are not allowed during TRR mode.

NOTE 11 All DRAM timings are to be met by DRAM during TRR mode such as tFAW. Issuing of ACT1, ACT2 and ACT3 counts towards tFAW budget.

Figure 59 — TRR Mode

4.13 Precharge Operation

The PRECHARGE command is used to precharge or close a bank that has been activated. The PRECHARGE command is initiated with CS_n LOW, CA0 HIGH, CA1 HIGH, CA2 LOW and CA3 HIGH at the rising edge of the clock. The PRECHARGE command can be used to precharge each bank independently or all banks simultaneously. For 4-bank devices(8ch x64), the AB flag, and the bank address bits, BA0 and BA1, are used to determine which bank(s) to precharge. For 8-bank devices(4ch x64), the AB flag, and the bank address bits, BA0, BA1, and BA2, are used to determine which bank(s) to precharge. The bank(s) will be available for a subsequent row access tRPab after an All-Bank Precharge command is issued and tRPpb after a Single-Bank Precharge command is issued.

In order to ensure that 8-bank devices do not exceed the instantaneous current supplying capability of 4-bank devices, the Row Precharge time (tRP) for an All-Bank Precharge for 8-bank devices (tRPab) will be longer than the Row Precharge time for a Single-Bank Precharge (tRPpb). For 4-bank devices, the Row Precharge time (tRP) for an All-Bank Precharge (tRPab) is equal to the Row Precharge time for a Single-Bank Precharge (tRPpb).

Table 32 — Bank selection for Precharge by address bits

AB (CA4r)	BA2 (CA9r)	BA1 (CA8r)	BA0 (CA7r)	Precharged Bank(s) 4-bank device (8ch x64)	Precharged Bank(s) 8-bank device (4ch x64)
0	0	0	0	Bank 0 only	Bank 0 only
0	0	0	1	Bank 1 only	Bank 1 only
0	0	1	0	Bank 2 only	Bank 2 only
0	0	1	1	Bank 3 only	Bank 3 only
0	1	0	0	Bank 0 only	Bank 4 only
0	1	0	1	Bank 1 only	Bank 5 only
0	1	1	0	Bank 2 only	Bank 6 only
0	1	1	1	Bank 3 only	Bank 7 only
1	don’t care	don’t care	don’t care	All Banks	All Banks

4.13.1 Burst Read followed by Precharge

For the earliest possible precharge, the precharge command may be issued BL/2 clock cycles after a Read command. A new bank active (command) may be issued to the same bank after the Row Precharge time (tRP). A precharge command cannot be issued until after tRAS is satisfied.

The minimum Read to Precharge spacing has also to satisfy a minimum analog time from the rising clock edge that initiates the last 4-bit prefetch of a Read command. This time is called tRTP (Read to Precharge).
tRTP begins BL/2 - 2 clock cycles after the Read command.

See Table 33 for Read to Precharge timings.

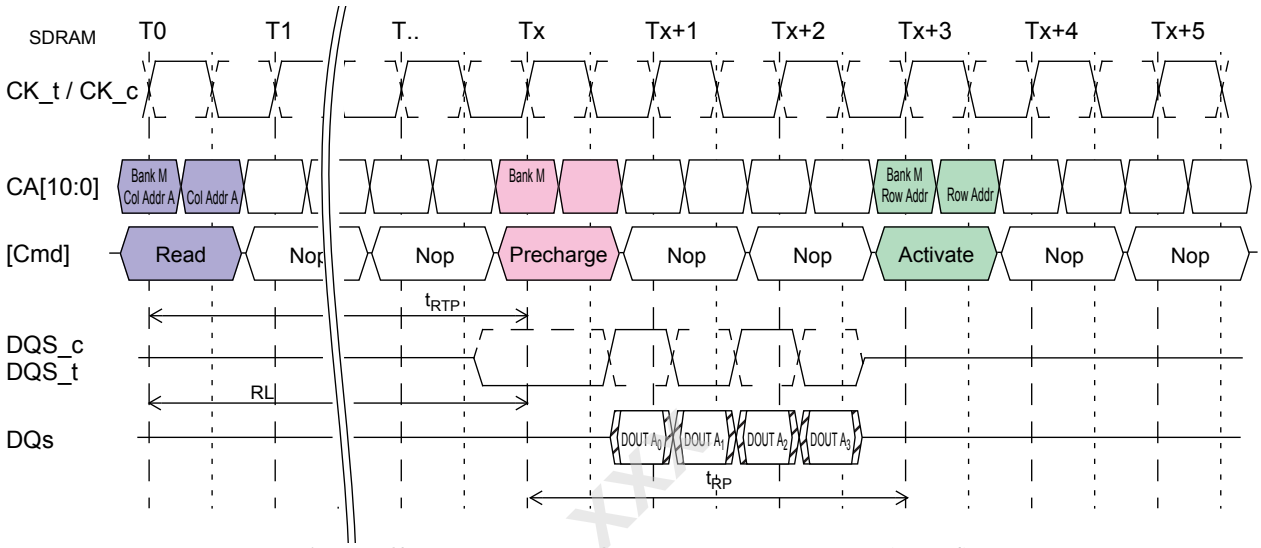


Figure 60 — Burst read followed by Precharge (BL=4)

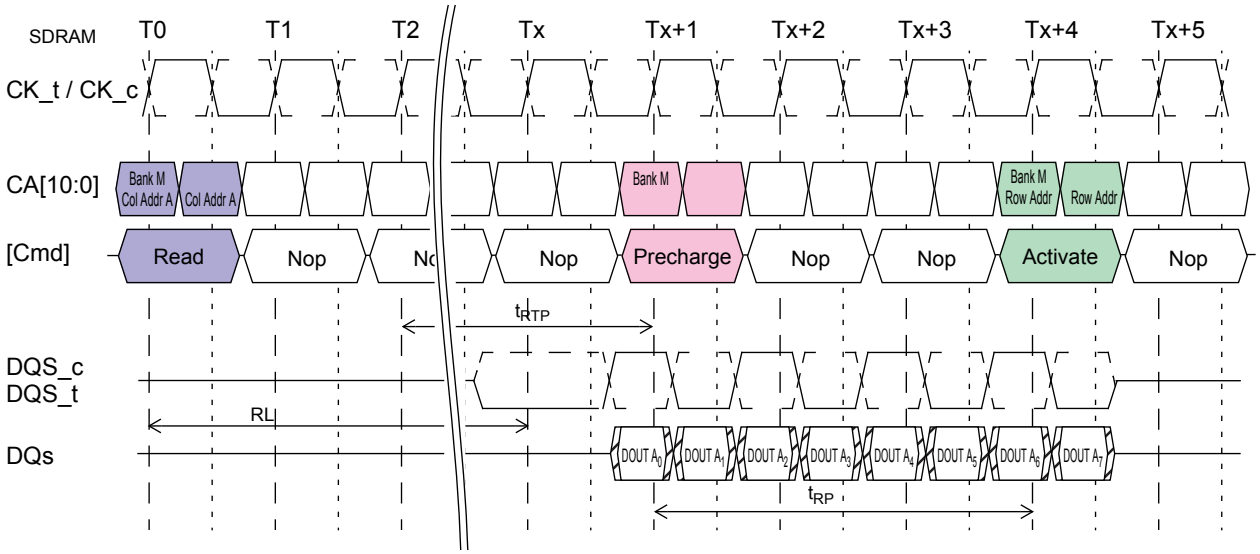


Figure 61 — Burst read followed by Precharge (BL=8)

4.13.2 Burst Write followed by Precharge

For WRITE cycles, a WRITE recovery time (tWR) must be provided before a PRECHARGE command can be issued. This delay is referenced from the last valid burst input data to the completion of the burst WRITE. A PRECHARGE command must not be issued prior to the tWR delay. For the Write-to-Precharge timings see the table "Precharge and Auto Precharge clarification".

WideIO2 devices write data to the array in prefetch quadruples (prefetch=4). An internal WRITE operation can only begin after a prefetch group has been completely latched, so tWR starts at prefetch boundaries. The minimum WRITE-to-PRECHARGE time for commands to the same bank is WL + BL/2 + 1 + RU(tWR/tCK) clock cycles.

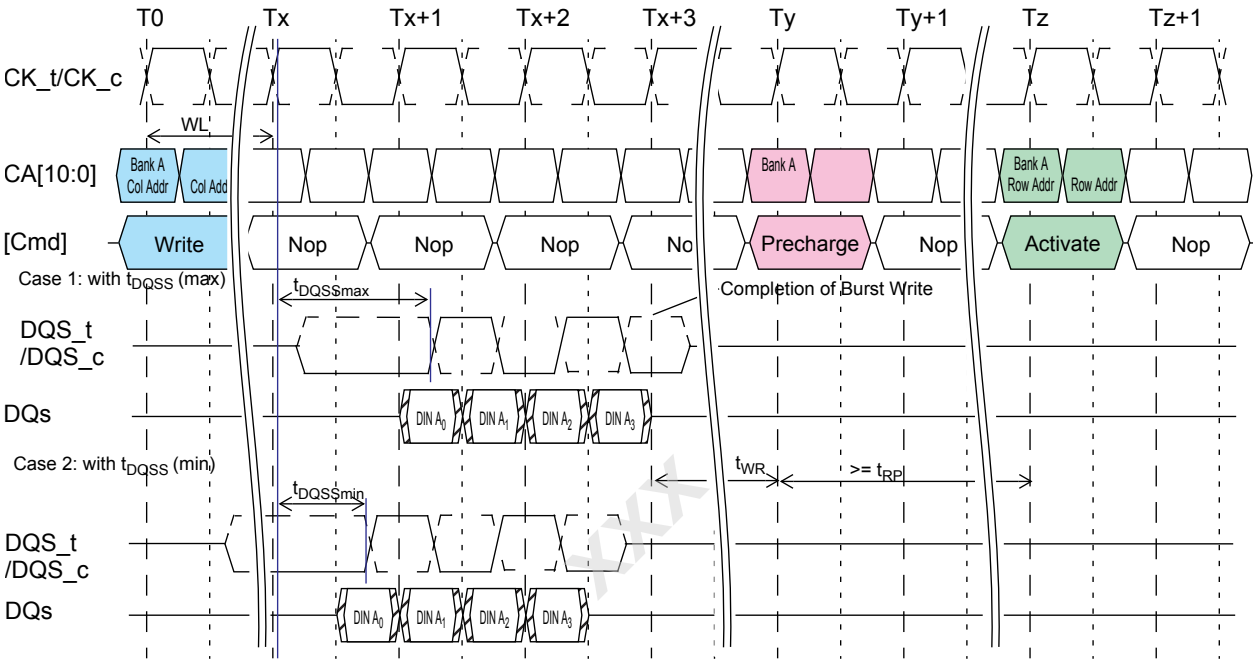


Figure 62 — Burst write followed by precharge

4.13.3 Auto Precharge operation

Before a new row can be opened in an active bank, the active bank must be precharged using either the PRECHARGE command or the auto precharge function. When a READ or a WRITE command is issued to the device, the AP bit (CA0f) can be set to enable the active bank to automatically begin precharge at the earliest possible moment during the burst READ or WRITE cycle.

If AP is LOW when the READ or WRITE command is issued, then normal READ or WRITE burst operation is executed and the bank remains active at the completion of the burst.

If AP is HIGH when the READ or WRITE command is issued, the auto precharge function is engaged. This feature enables the PRECHARGE operation to be partially or completely hidden during burst READ cycles (dependent upon READ or WRITE latency) thus improving system performance for random data access.

4.13.4 Burst Read with Auto-Precharge

If AP (CA0f) is HIGH when a Read Command is issued, the Read with Auto-Precharge function is engaged.

Wide IO2 devices start an Auto-Precharge operation on the rising edge of the clock BL/2 - 2 + nRTP clock cycles later than the Read with auto precharge command. For Wide IO2 auto-precharge calculations see the table “Precharge and Auto Precharge clarification”. Following an auto-precharge operation, an ACTIVATE command can be issued to the same bank if the following two conditions are satisfied simultaneously:
a) The RAS precharge time (tRP) has been satisfied from the clock at which the auto-precharge begins.
b) The RAS cycle time (tRC) from the previous bank activation has been satisfied.

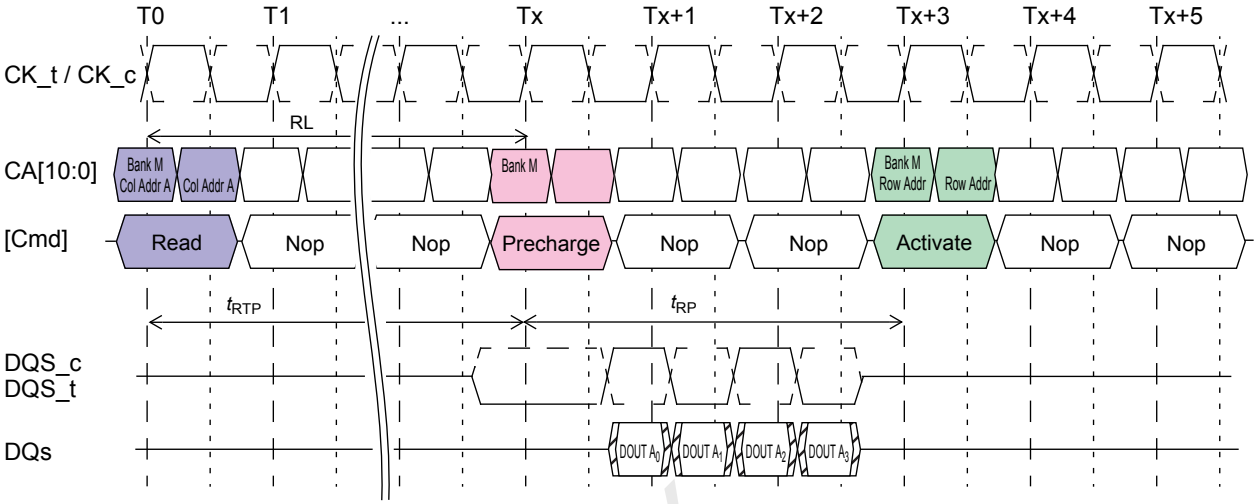


Figure 63 — Burst Read with Auto-Precharge

4.13.5 Burst Write with Auto-Precharge

If AP (CA0f) is HIGH when a Write Command is issued, the Write with Auto-Precharge function is engaged. The device starts an Auto Precharge operation on the rising edge tWR cycles after the completion of the burst WRITE.

Following a WRITE with auto precharge, an ACTIVATE command can be issued to the same bank if the following two conditions are met:
a) The RAS precharge time (tRP) has been satisfied from the clock at which the auto-precharge begins.
b) The RAS cycle time (tRC) from the previous bank activation has been satisfied.

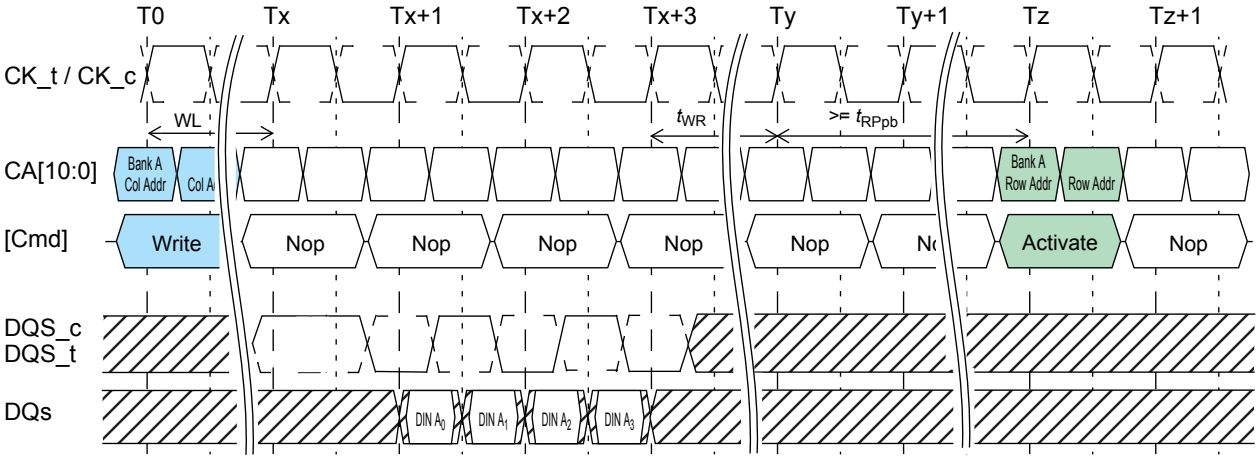


Figure 64 — Burst Write with Auto-Precharge

4.13.5 Burst Write with Auto-Precharge (cont'd)

Table 33 — Precharge & Auto Precharge clarification

From Command	To Command	Minimum Delay between "From Command" to "To Command"	Unit	Notes
Read	Precharge (to same Bank as Read)	BL/2 + nRTP - 2	clks	1
	Precharge All	BL/2 + nRTP - 2	clks	1
Read w/AP	Precharge (to same Bank as Read w/AP)	BL/2 + nRTP - 2	clks	1,2
	Precharge All	BL/2 + nRTP - 2	clks	1
	Activate (to same Bank as Read w/AP)	BL/2 + nRTP - 2 + RU(t_{RPpb}/t_{CK})	clks	1
	Write or Write w/AP (same bank)	Illegal	clks	3
	Mask-WR or MASK-WR w/AP (same bank)	Illegal	clks	
	Write or Write w/AP (different bank)	RL + BL/2 + RU(tDQSCKmax/tCK) - WL + 1	clks	3
	Mask-WR or MASK-WR w/AP (different bank)	RL + BL/2 + RU(tDQSCKmax/tCK) - WL + 1	clks	3
	Read or Read w/AP (same bank)	Illegal	clks	3
	Read or Read w/AP (different bank)	BL/2	clks	3
Write	Precharge (to same Bank as Write)	WL + BL/2 + nWR + 1	clks	1
	Precharge All	WL + BL/2 + nWR + 1	clks	1
MASK-WR	Precharge (to same Bank as Masked-WR)	WL + BL/2 + nWR + 1	clks	
	Prechareg All	WL + BL/2 + nWR + 1	clks	
Write w/AP	Precharge (to same Bank as Write w/AP)	WL + BL/2 + nWR + 1	clks	1
	Precharge All	WL + BL/2 + nWR + 1	clks	1
	Activate (to same Bank as Write w/AP)	WL + BL/2 + nWR + 1 + RU(t_{RPpb}/t_{CK})	clks	1
	Write or Write w/AP (same bank)	Illegal	clks	3
	Write or Write w/AP (different bank)	BL/2	clks	3
	Read or Read w/AP (same bank)	Illegal	clks	3
	Read or Read w/AP (different bank)	WL + BL/2 + RU(t_{WTR}/t_{CK}) + 1	clks	3
Mask-WR w/AP	Precharge (to same Bank as Write w/AP)	WL + BL/2 + nWR + 1	clks	1
	Precharge All	WL + BL/2 + nWR + 1	clks	1
	Activate (to same Bank as Write w/AP)	WL + BL/2 + nWR + 1 + RU(t_{RPpb}/t_{CK})	clks	1
	Write or Write w/AP (same bank)	Illegal	clks	3
	Mask-WR or Mask-WR w/AP (same bank)	Illegal	clks	3
	Write or Write w/AP (different bank)	BL/2	clks	3
	Mask-WR or Mask-WR w/AP (different bank)	BL/2	clks	3
	Read or Read w/AP (same bank)	Illegal	clks	3
	Read or Read w/AP (different bank)	WL + BL/2 + RU(tWTR/tCK) + 1	clks	3
Precharge	Precharge (to same Bank as Precharge)	2	clks	1,4
	Precharge All	2	clks	1,4
Precharge All	Precharge	2	clks	1,4
	Precharge All	2	clks	1,4

NOTE 1 For a given bank, the precharge period should be counted from the latest precharge command, either one bank precharge or precharge all, issued to that bank. The precharge period is satisfied after t_{RP} depending on the latest precharge command issued to that bank.

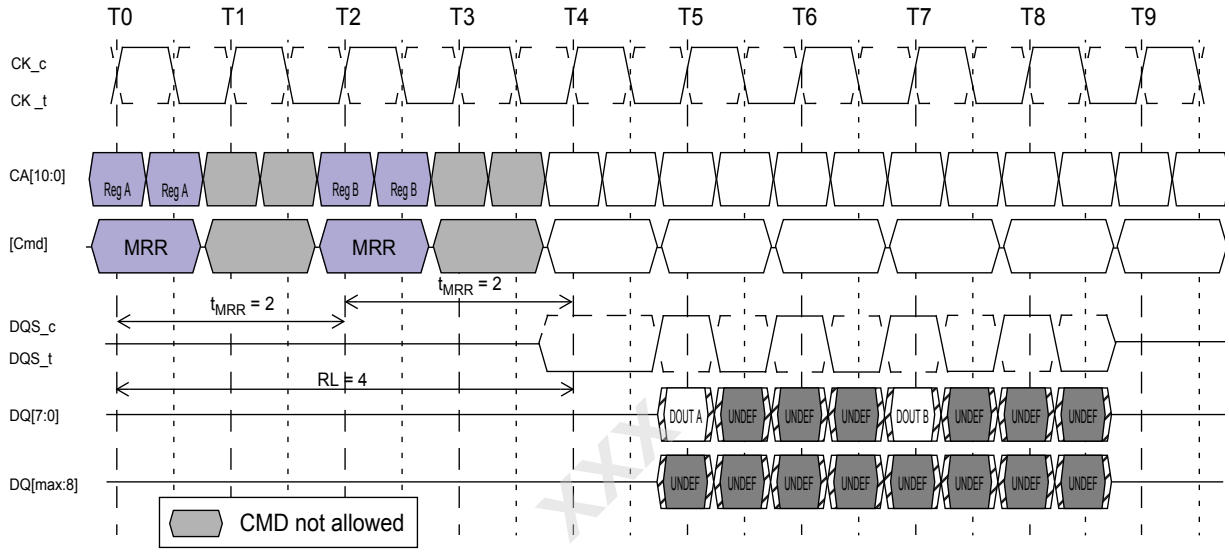
NOTE 2 Any command issued during the minimum delay time as specified in the table above is illegal.

NOTE 3 After READ w/AP, seamless read operations to different banks are supported. After WRITE w/AP or MASK-WR w/AP, seamless write operations to different banks are supported. READ, WRITE, and MASK-WR operations may not be truncated or interrupted.

NOTE 4 Precharge command (all bank or per bank) delay to Precharge command (all bank or per bank) tPPD, does not impose any timing restrictions relative to the internal Auto Precharge function initiated by read or write commands with auto precharge. tPPD does not impose any Precharge command to other (non-Precharge) command timing restrictions.

4.14 Mode Register Read (MRR) Command

The Mode Register Read (MRR) command is used to read configuration and status data from SDRAM mode registers. The MRR command is initiated with CS_n LOW, CA0 LOW, CA1 LOW, CA2 LOW, and CA3 HIGH at the rising edge of the clock. The mode register is selected by CA7r–CA4r. The mode register contents are available on the first data beat of DQ0-DQ7, RL * tCK + tDQSCK + tDQSQ after the rising edge of the clock where the Mode Register Read Command is issued. Subsequent data beats contain valid, but undefined content. All DQS, DQS shall be toggled for the duration of the Mode Register Read burst. The MRR command has a burst length of four. The Mode Register Read operation (consisting of the MRR command and the corresponding data traffic) shall not be interrupted. The MRR command period (tMRR) is 2 clock cycles. Mode Register Reads to reserved and write-only registers shall return valid, but undefined content on all data beats and DQS, DQS shall be toggled.



NOTE 1 Mode Register Read has a burst length of four.
NOTE 2 Mode Register Read operation shall not be interrupted.
NOTE 3 Mode Register data is valid only on DQ[0-7] on the first beat. Subsequent beats contain valid, but undefined data. DQ[8-max] contain valid, but undefined data for the duration of the MRR burst.
NOTE 4 If DBI is enabled, DBI function is applied to mode register data.
NOTE 5 OP to DQ mapping

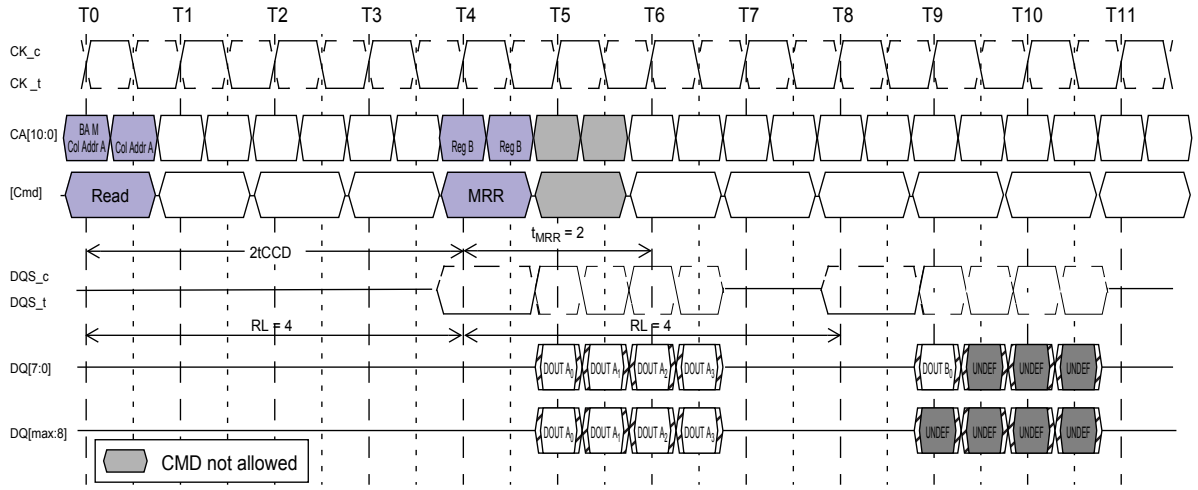
OP0	OP1	OP2	OP3	OP4	OP5	OP6	OP7
DQ0	DQ1	DQ2	DQ3	DQ4	DQ5	DQ6	DQ7

NOTE 6 The Mode Register Command period is tMRR. No command (other than Nop) is allowed during this period.
NOTE 7 Minimum Mode Register Read to write latency is RL + RU(tDQSCKmax/tCK) + 4/2 + 1 - WL clock cycles.
NOTE 8 Minimum Mode Register Read to Mode Register Write latency is RL + RU(tDQSCKmax/tCK) + 4/2 + 1clock cycles.
NOTE 9 Minimum Mode register read to Read latency is 2*tCCD
NOTE 10 Minimum Read to Mode register read latency is 2*tCCD.

Figure 65 — Mode Register Read timing example: RL = 4, t_{MRR} = 2

4.14 Mode Register Read (MRR) Command (cont'd)

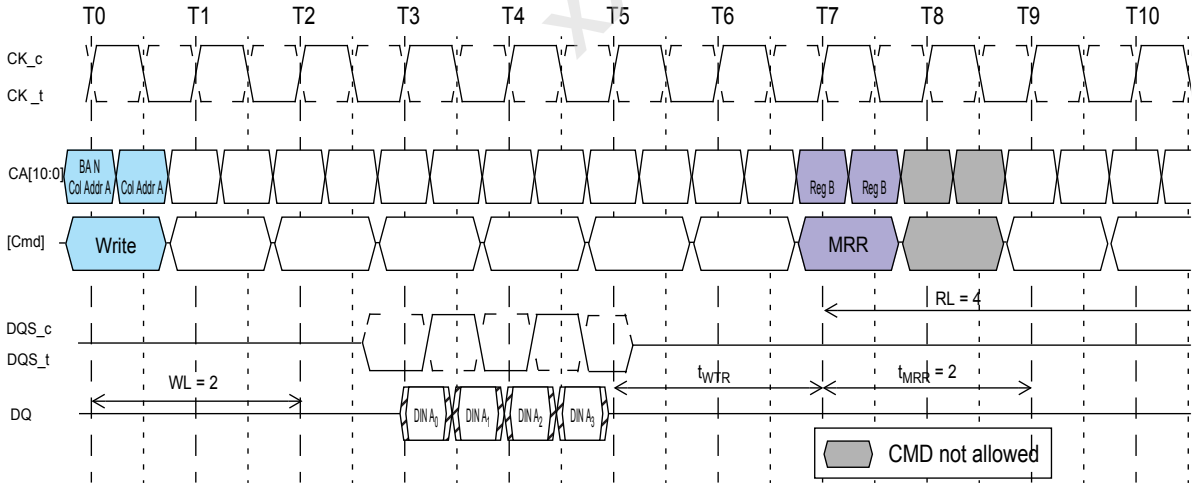
The MRR command shall not be issued earlier than BL/2 clock cycles after a prior Read command and WL + 1 + BL/2 + RU(tWTR/tCK) clock cycles after a prior Write command, because read-bursts and write-bursts shall not be truncated by MRR.



NOTE 1 The minimum number of clocks from the burst read command to the Mode Register Read command is $2*t_{CCD}$.

NOTE 2 The Mode Register Read Command period is t_{MRR}. No command (other than Nop) is allowed during this period.

Figure 66 — Wide IO2: Read to MRR timing example: RL = 4, t_{MRR} = 2



NOTE 1 The minimum number of clock cycles from the burst write command to the Mode Register Read command is [WL + 1 + BL/2 + RU(t_{WTR} /t_{CK})].

NOTE 2 The Mode Register Read Command period is t_{MRR}. No command (other than Nop) is allowed during this period.

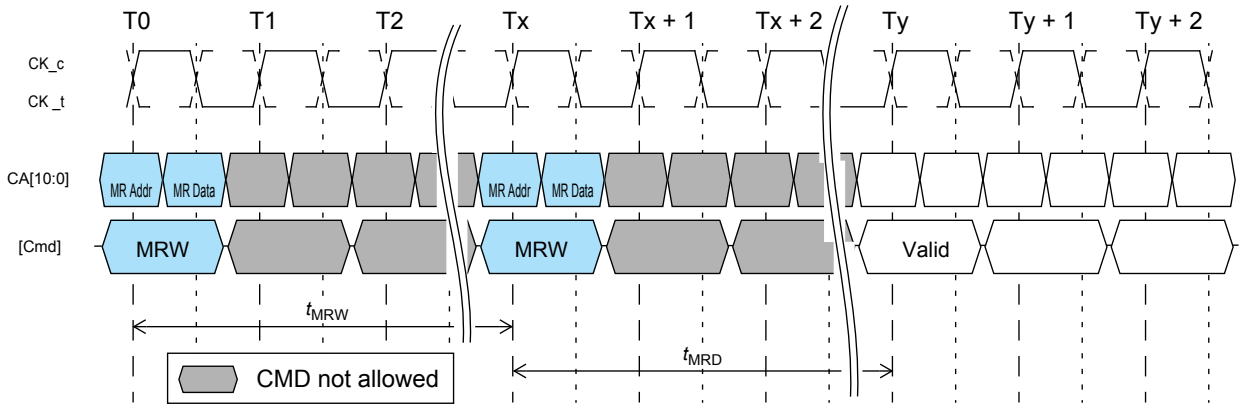
Figure 67 — Wide IO2: Burst Write Followed by MRR: RL = 4, WL = 2, BL = 4

4.14.1 MRR Following Idle Power-Down State

Following the idle power-down state, an additional time, *t*MRRI, is required prior to issuing the mode register read (MRR) command. This additional time (equivalent to *t*RCD) is required in order to be able to maximize power-down current savings by allowing more power-up time for the MRR data path after exit from standby, idle power-down mode.

4.15 Mode Register Write (MRW) Command

The Mode Register Write (MRW) command is used to write configuration data to mode registers. The MRW command is initiated with CS_n LOW, CA0 LOW, CA1 LOW, CA2 LOW, and CA3 LOW at the rising edge of the clock. The mode register is selected by CA7r-CA4r. The data to be written to the mode register is contained in CA9f-CA2f. The MRW command period is defined by t_{MRW}. Mode register WRITEs to read-only registers have no impact on the functionality of the device.



NOTE 1 At time Ty, the device is in the idle state.

NOTE 2 Only DE-Select command is supported during t_{MRW} and t_{MRD} periods.

Figure 68 — Mode Register Write Timing

4.15.1 MRR and MRW Restrictions

MRW can only be issued when all banks are in the idle precharge state. One method of ensuring that the banks are in this state is to issue a PRECHARGE-ALL command.

Table 34 — Truth Table for Mode Register Read (MRR) and Mode Register Write (MRW)

Current State	Command	Intermediate State	Next State
All Banks Idle	MRR	Mode Register Reading (All Banks Idle)	All Banks Idle
	MRW	Mode Register Writing (All Banks Idle)	All Banks Idle
Bank(s) Active	MRR	Mode Register Reading (Bank(s) Active)	Bank(s) Active
	MRW	Not Allowed	Not Allowed

4.16 Deselect and No Operation Command

The Deselect command (DES) is to prevent the WideIO2 device from registering any unwanted command between operations. Only when the CKE level is constant for clock cycle N-1 and clock cycle N, a DES command may be issued at clock cycle N. The DES command is encoded as CS_n High at the clock rising edge N.

The purpose of the No Operation command (NOP) is to prevent the WideIO2 device from registering any unwanted command between operations. Only when the CKE level is constant for clock cycle N-1 and clock cycle N, a NOP command may be issued at clock cycle N. The NOP command is encoded as CS_n LOW and CA0, CA1, CA2 HIGH at the clock rising edge N.

The Deselect and No Operation command will not terminate a previous operation that is still executing, such as a burst read or write cycle.

5 WideIO2 Boundary Scan Interface

5.1 Overview

The WideIO2 Boundary Scan interface provides test access to the CA and Data IO buffers, enabling stuck node and connectivity testing within the DRAM die/stack and between the DRAM die/stack and the SoC. Notable features of the Boundary Scan implementation include:

- Shared/common scan mode controls and Scan Data In across all slices.
- Individual Boundary Scan test mode Chip Select signals per slice (SCS[3:0]_n) which control scan chain shifting and parallel output of the scan chain data on the corresponding Ca and Data Tx buffers.
- The logic state of the CA and Data nodes may be captured into the scan chains, enabling stuck-node and connectivity testing.
- Individual Scan Data Out signals per slice (SDO[3:0]), enabling parallel observation and fewer shifts when doing boundary scan testing.
- Scan enable signals (SEN[2:0]) select normal operation, I/O boundary scan, or 1 of 6 vendor specific scan chains.

Boundary scan mode is entered by asserting and deasserting RST_n per the 3.2, Power-up, Initialization, and Power-off, through the tINIT3 stage and then asserting SEN[2:0] = ‘b001. The CK and CKE signals need not be asserted or toggled in order to enter boundary scan mode. Boundary scan mode entry at any time after reset is undefined. The intent of only requiring the minimal reset sequence for use of boundary scan modes is to allow verification of connectivity between the WideIO2 DRAM and the SOC without depending on a successful full reset of the WideIO2 DRAM.

While in boundary scan mode (SEN[2:0]='b001) the internal state of the DRAM is effectively forced into Deselect for all channels per Table 20, Command Truth Table, while the IO pins follow Table 35, Boundary Scan Truth Table.

To exit boundary scan modes and to enable normal operation, one of the full initialization sequences must be executed as defined in 3.2, Power-up, Initialization, and Power-off.

5.2 Slice and Stack Level Configuration

The Boundary Scan logic operates independently in each slice of the WideIO2 two and four-die stack configurations. Each slice is presented the same SEN[2:0], SDI, SCK, and SSH_n signals using Through routing. The per-slice SCS_n signals use Staggered routing and determine if and how each slice processes the boundary scan mode operations. Each slice presents an independent SDO signal, routed to the bottom of the DRAM die stack using Staggered routing. When SEN[2:0] selects normal operation mode, the SDO signals equal L for all present slices. The SDO signals from non-present slices will float, for example SDO1, SDO2, and SDO3 are floating signals for a single die WideIO2 configuration. It is suggested that the SoC pull the SDO signals low with a weak pull-down to avoid current flow in the corresponding SoC receivers.

Within a slice, the scan chain is connected in series across all quadrants.

In 4x64 die and die stack configurations, each slice implements one channel within each quadrant, and connects to one half of the CA and Data micro-bumps within the quadrant. In 8x64 die and die stack configurations, each slice implements two channels within each quadrant, and connects to all CA and Data micro-bumps within the quadrant. The scan chain orders are different between 4x64 and 8x64 configurations.

5.2 Slice and Stack Level Configuration (cont'd)

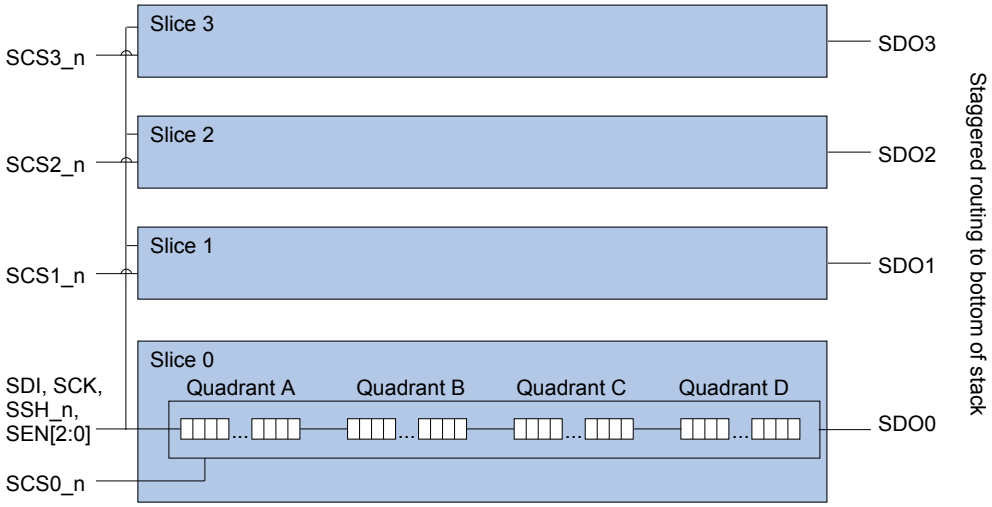


Figure 69 — Slice and Stack Level Configuration

5.3 Within-Slice Scan Chain Configuration and Operation

Figure 70 represents the logical behavior of the scan chain within a slice. Specific implementations may vary but must adhere to the described logical behavior. Boundary Scan logic is shaded. Note that the normally input-only CA pins implement a Tx buffer for Boundary Scan test purposes.

SEN[2:0]='b001 enables Boundary Scan mode operations. When Boundary Scan mode is enabled, two signals are derived from the combination of the defined Boundary Scan signals:

- Output Enable - Within a slice, the Tx buffers are enabled when not scan shifting (SSH_n=H) and when the slice is selected (SCS_n=L). The content of the scan chain flip-flops is driven on the corresponding CA and Data pins.
- Clock - The Boundary Scan mode clock (SCK) is applied to the scan chain flip-flops in two conditions. 1) Scan shifting occurs when scan shift mode is enabled (SSH_n=L) and the slice is selected (SCS_n=L), and 2) Boundary Scan Capture occurs when not shifting (SSH_n=H) and on SCK rising.

Note that the described Output Enable and Clock signals are internal signals and not part of the WideIO2 specification. They represent an example WideIO2-compliant implementation. The specific implementation may vary but must adhere to the logical behavior described in Table 35, "Boundary Scan Truth Table".

5.3 Within-Slice Scan Chain Configuration and Operation (cont'd)

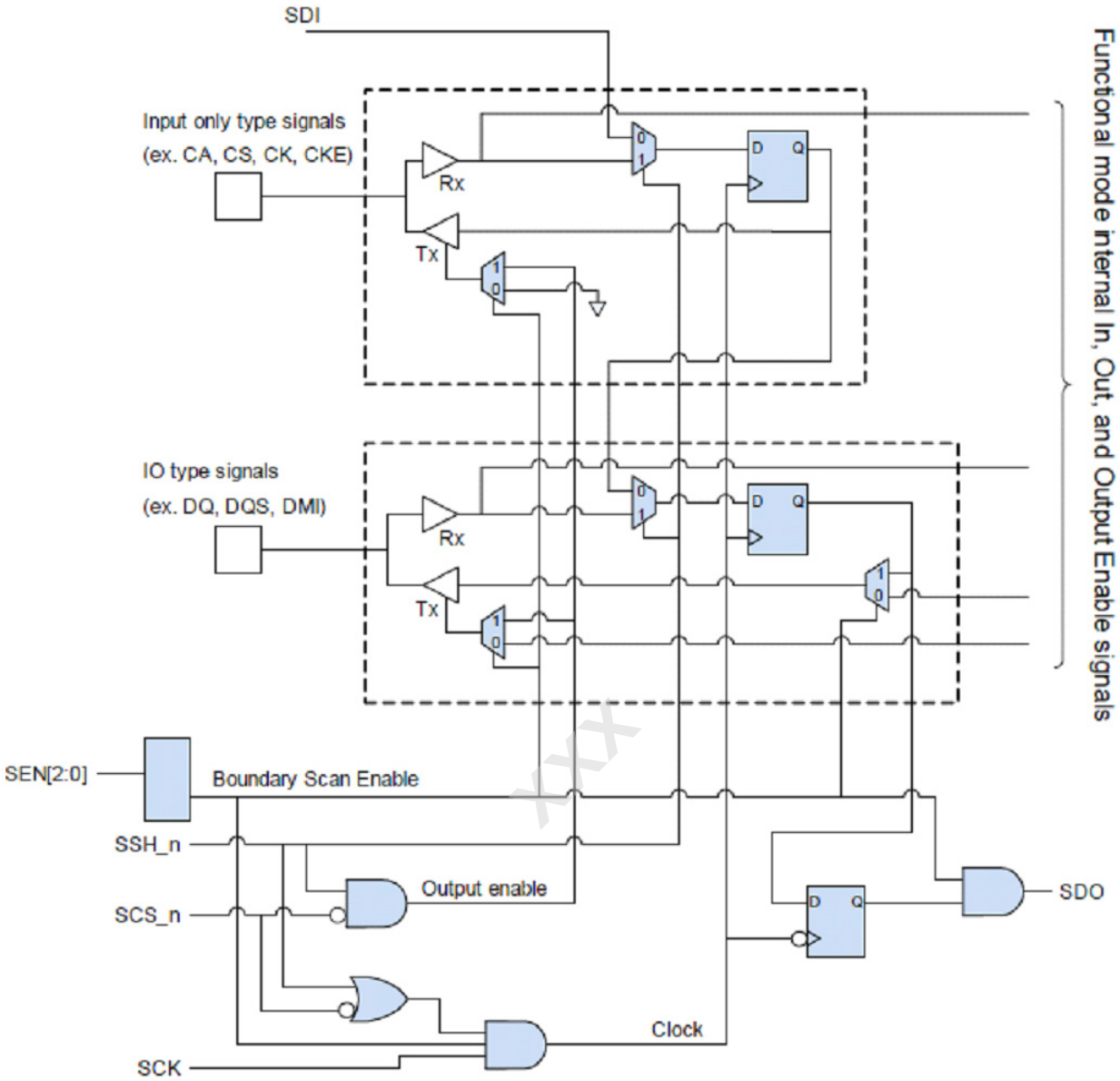


Figure 70 — Within-Slice Scan Chain Example Implementation

5.3 Within-Slice Scan Chain Configuration and Operation (cont'd)

Table 35 — Boundary Scan Truth Table

Description	SEN[2:0]	SCS[3:0]_n	SSH_n	SCK	SDI	SDO[3:0] (Notes 1, 2)	CA/Data	Notes
Normal operation	'b000	X	X	X	X	L	Normal operation	Boundary scan logic is disabled. SDO signals for present slices are driven low.
NOP	'b001	H	X	L	X	X	Z	This state may be held indefinitely.
Serial Data In/out	'b001	L for slices to be shifted	L	Clock	Valid	Valid	Z	The scan chain(s) shift on each rising edge of SCK.
	'b001	H for non-shifted slices	L	Clock	X	X	Z	Non-chip-selected slices do not shift.
Parallel Output	'b001	L for slice(s) that are enabled for Parallel Out	H	X	X	X	Output scan chain data for slices with SCS_n=L	SCS_n=L enables Tx buffers on selected slice(s) when SSH_n=H
	'b001	H for slices not driven	H	X	X	X	Z	Tx buffers float for slices with SCS_n=H
Parallel Input/Capture	'b001	X	H	Clock Rising Edge	X	Captured state of last CA pin in the scan chain is driven on SDO after SCK falling.	CA/Data node states are captured to scan chains on SCK rising	All chains capture, including any chains doing Parallel Output. More than one capture pulse may be issued on SCK. The scan chains retain the states captured on the last SCK rising edge.
Vendor Specific Modes	!= 'b00X	X	X	X	X	X	X	Modes not defined. Reserved for vendor specific use.

NOTE 1 When in Boundary Scan mode (SEN[2:0]='b001), SDO signals will actively drive H or L. When not scan shifting (SSH_n=H) the SDO driven states are not specified, and are implementation specific. When not scan shifting the SDOs should be considered invalid (X).

NOTE 2 SDO is only valid for present slices. For example, in a single slice WideIO2 configuration only SDO0 is valid, and SDO1, 2, and 3 float.

5.4 Boundary Scan Operational Timing Diagram

Figure 71 illustrates the Boundary Scan operations sequence, based on a four-slice P22P WideIO2 configuration. In this example, data is scanned into Slice 0 and driven onto the Slice 0 CA and Data pins, then captured into both Slice 0 and 2 scan chains and finally scanned out of both slices.

The example test application sequence is as follows:

1. Data is shifted into the Slice 0 scan chain by asserting SCS0_n=L, SSH_n=L, and clocking data in from SDI.
2. The CA and Data Tx buffers on Slice 0 are enabled when SSH_n=H and SCS0_n=L.
3. In a P22P configuration, the CA and Data nodes are connected between Slice 0 and Slice 2. The logic states driven by Slice 0 are also seen at the Slice 2 buffers.
4. The logic states on all CA and Data nodes on all slices are captured when SCK rises and not scan shifting (SSH_n=H). Note that the captured state of the last pin in the scan chain order (Bit 1) is driven on all respective SDO pins on SCK falling. The SOC should capture Bit 1 before the first SCK falling edge of the following scan shift operation.
5. The captured results are shifted out of Slice 2 with SCS2_n=L. Data on SDI fills the Slice 2 scan chain.
6. The captured results are shifted out of Slice 0 with SCS0_n=L. Data on SDI fills the Slice 0 scan chain.

5.4 Boundary Scan Operational Timing Diagram (cont'd)

Note that Bit 1 shifted into Slice 0 will appear on SDO0 once all bits have been shifted into the Slice 0 scan chain - "Bit 1 (shifted)". Once the Parallel Input / Capture pulse is applied, the data captured on the last IO buffer node, per the scan chain order, will be driven on SDO0 and SDO2 - "Bit 1 (captured)". For a non-defective device, Bit 1 captured would be the same value as Bit 1 shifted.

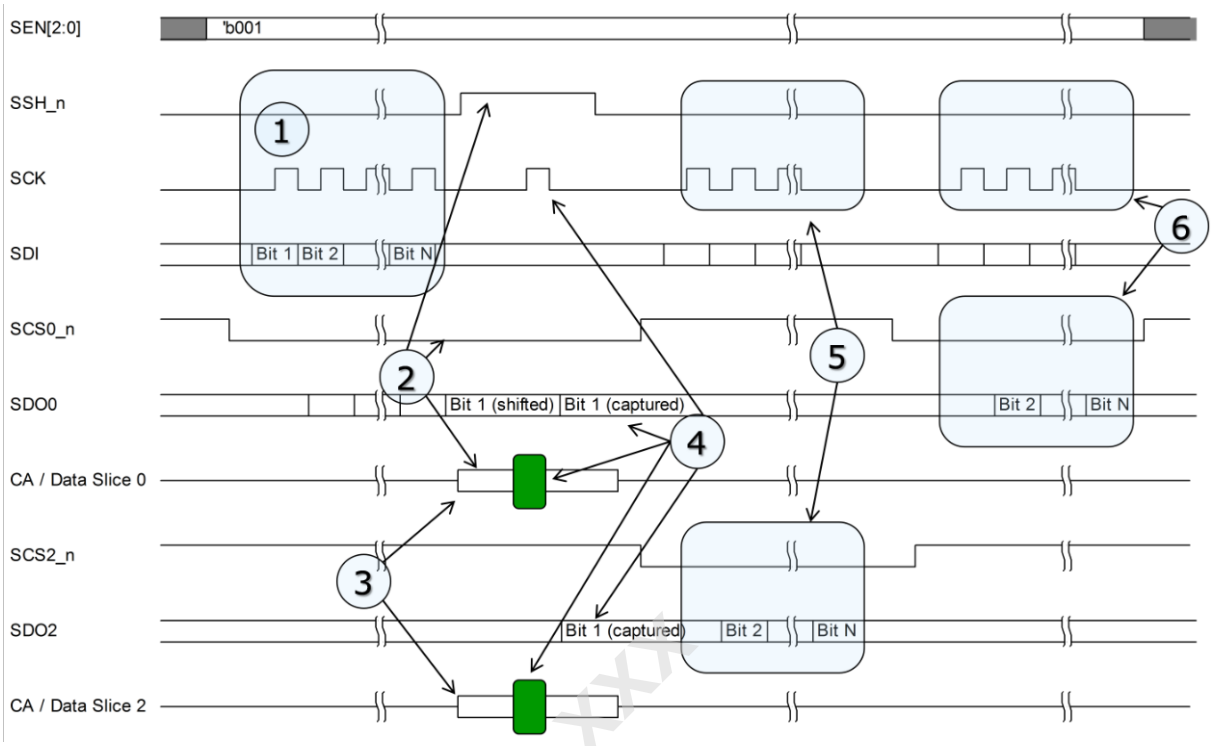


Figure 71 — Boundary Scan Operational Timing Diagram

Simultaneous operation of Boundary Scan mode (SEN[2:0]='b001) and DA or GPIO Bypass Mode (TEST=H) is undefined.

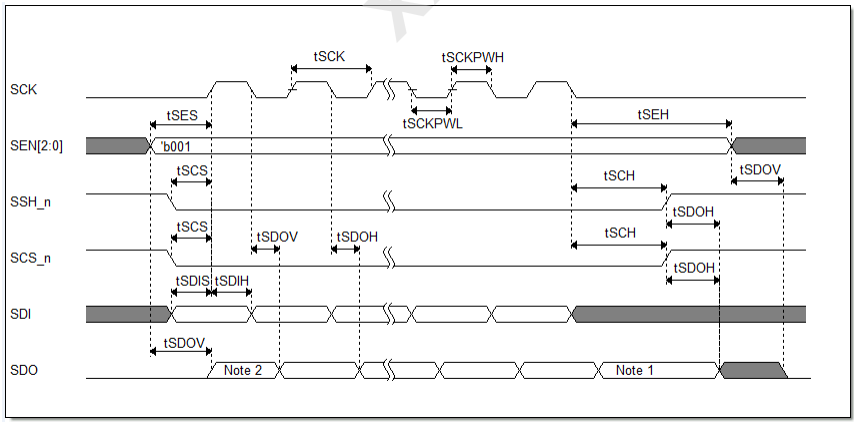
5.5 Test Area Micro-Bump Assignments

The assignment of Boundary Scan related signals is shown in Figure 12, along with other signals not related to boundary scan.

5.6 Boundary Scan AC Timing

Table 36 — Boundary Scan AC Timings

Parameter	Symbol	Min	Max	Unit
Scan clock cycle time	t_{SCK}	20		ns
Scan clock pulse width high	t_{SCKPWH}	8		ns
Scan clock pulse width low	t_{SCKPWL}	8		ns
Scan enable setup time	t_{SES}	10		ns
Scan enable hold time	t_{SEH}	10		ns
Scan command setup time	t_{SCS}	5		ns
Scan command hold time	t_{SCH}	5		ns
Scan Data In setup time	t_{SDIS}	5		ns
Scan Data In hold time	t_{SDIH}	5		ns
Scan Data Out propagation	t_{SDOV}		10	ns
Scan Data Out hold time	t_{SDOH}	1		ns
Parallel output valid	t_{SPV}		10	ns
Parallel output high Z	t_{SPZ}		10	ns
Parallel input setup time	t_{SPS}	5		ns
Parallel input hold time	t_{SPH}	5		ns



NOTE 1 The SOC may need an additional clock to capture this last data bit on SDO. The t_{SCH} specification ensures proper scan shift operation; however, if t_{SCH} is set to the minimum specification the last SDO data bit may be too short to be correctly sampled by the SOC. SSH_n and SCS_n should be held low long enough after the last scan shift cycle for the SOC to capture this last data bit.

NOTE 2 When boundary scan mode is activated, SDO will drive the state that is retained in the final falling edge flip flop of the scan chain. This state may have been set by a prior Serial Data In/Out operation or a Parallel Input/Capture operation, or is otherwise undefined.

NOTE 3 In order to meet setup and hold time specifications it may be necessary to have a clock cycle gap between the capture pulse and scan chain shift clock cycles.

Figure 72 — Scan Shift Timing

5.6 Boundary Scan AC Timing (cont'd)

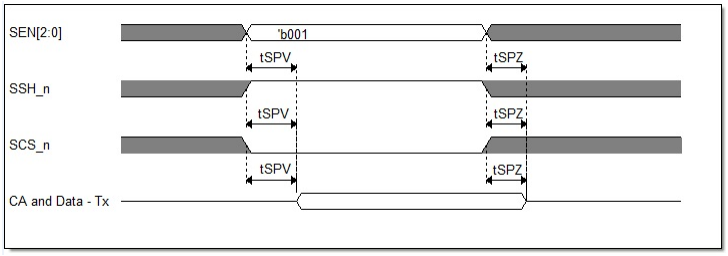


Figure 73 — Parallel Output Timing

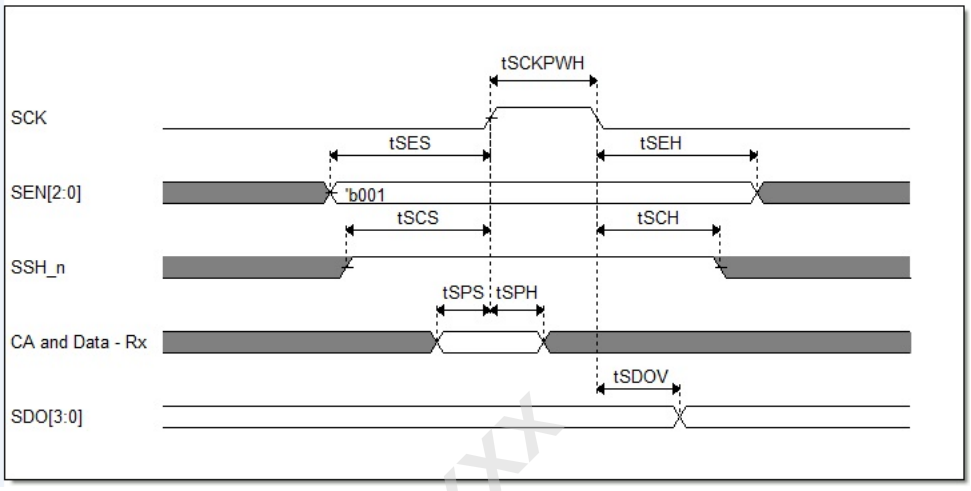


Figure 74 — Parallel Input Timing

5.7 Boundary Scan Chain Order

These signals are included in the Boundary Scan chains: DQ, DQS, DMI, CK, CKE, CS_n, CA.

These signals are not included in the Boundary Scan chains: SEN, SSH_n, SCK, SDI, SDO, SCS_n, TEST, RST_n, DA, DAA.

5.7.1 4x64 Die Boundary Scan Chain Order

Table 37 shows the boundary scan chain order for a 4x64 die. This table only shows the signals for quadrant A. Since quadrants B, C, and D are flipped images of quadrant A, their chain order will be the same (just swap out the quadrant letter in the signal name). The scan order for a 8 x 64 die is shown in Table 38. Figure 75 shows how the boundary scan chain is routed through a 4 channel slice and Figure 76 shows how the boundary scan chain is routed through an 8 channel slice (quadrant order is A - D - C - B).

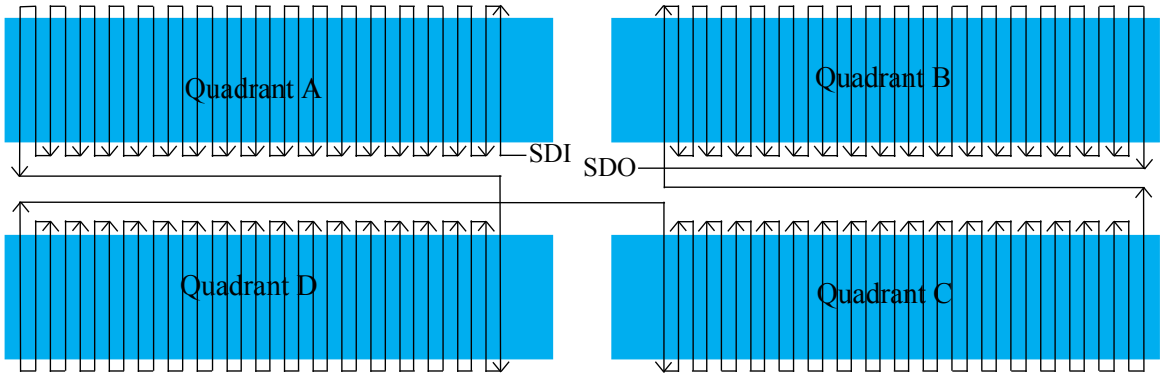


Figure 75 — Boundary Scan Chain 4 Channel Routing Through Quadrants

Table 37 — 4x64 Die Boundary Scan Exit Order (Quadrant A)

Bit#	Location	Signal Name	Bit#	Location	Signal Name	Bit#	Location	Signal Name	Bit#	Location	Signal Name	Bit#	Location	Signal Name
1	J4	CA0A10	20	K18	DQ0A59	39	H32	DQ0A44	58	M44	DQ0A29	77	K58	DQ0A14
2	L4	CA0A9	21	H18	DQ0A60	40	K32	DQ0A43	59	M46	DMI0A3	78	H58	DQ0A15
3	M6	CA0A6	22	H20	DQ0A58	41	M32	DMI0A5	60	K46	DQ0A27	79	H60	DQ0A12
4	K6	CA0A7	23	K20	DQ0A57	42	M34	DQ0A40	61	H46	DQ0A28	80	K60	DQ0A11
5	H6	CA0A8	24	M20	DQ0A56	43	K34	DQ0A41	62	H48	DQ0A26	81	M60	DMI0A1
6	H8	CA0A5	25	L22	DQS0A3_t	44	H34	DQ0A42	63	K48	DQ0A25	82	M62	DQ0A8
7	K8	CA0A4	26	J22	DQS0A3_c	45	J36	DQS0A2_c	64	M48	DQ0A24	83	K62	DQ0A9
8	M8	CA0A3	27	H24	DQ0A55	46	L36	DQS0A2_t	65	L50	DQS0A1_t	84	H62	DQ0A10
9	M10	CA0A0	28	K24	DQ0A54	47	M38	DQ0A37	66	J50	DQS0A1_c	85	J64	DQS0A0_c
10	K10	CA0A1	29	M24	DQ0A53	48	K38	DQ0A38	67	H52	DQ0A23	86	L64	DQS0A0_t
11	H10	CA0A2	30	M26	DMI0A6	49	H38	DQ0A39	68	K52	DQ0A22	87	M66	DQ0A5
12	K12	CKEA0	31	K26	DQ0A51	50	H40	DQ0A36	69	M52	DQ0A21	88	K66	DQ0A6
13	M12	CK0A_t	32	H26	DQ0A52	51	K40	DQ0A35	70	M54	DMI0A2	89	H66	DQ0A7
14	M14	CK0A_c	33	H28	DQ0A50	52	M40	DMI0A4	71	K54	DQ0A19	90	H68	DQ0A4
15	K14	CSA0_n	34	K28	DQ0A49	53	M42	DQ0A32	72	H54	DQ0A20	91	J68	DQ0A3
16	H16	DQ0A63	35	M28	DQ0A48	54	K42	DQ0A33	73	H56	DQ0A18	92	M68	DMI0A0
17	K16	DQ0A62	36	M30	DQ0A45	55	H42	DQ0A34	74	K56	DQ0A17	93	M70	DQ0A0
18	M16	DQ0A61	37	K30	DQ0A46	56	H44	DQ0A31	75	M56	DQ0A16	94	K70	DQ0A1
19	M18	DMI0A7	38	H30	DQ0A47	57	K44	DQ0A30	76	M58	DQ0A13	95	H70	DQ0A2

5.7.2 8x64 Die Boundary Scan Chain Order

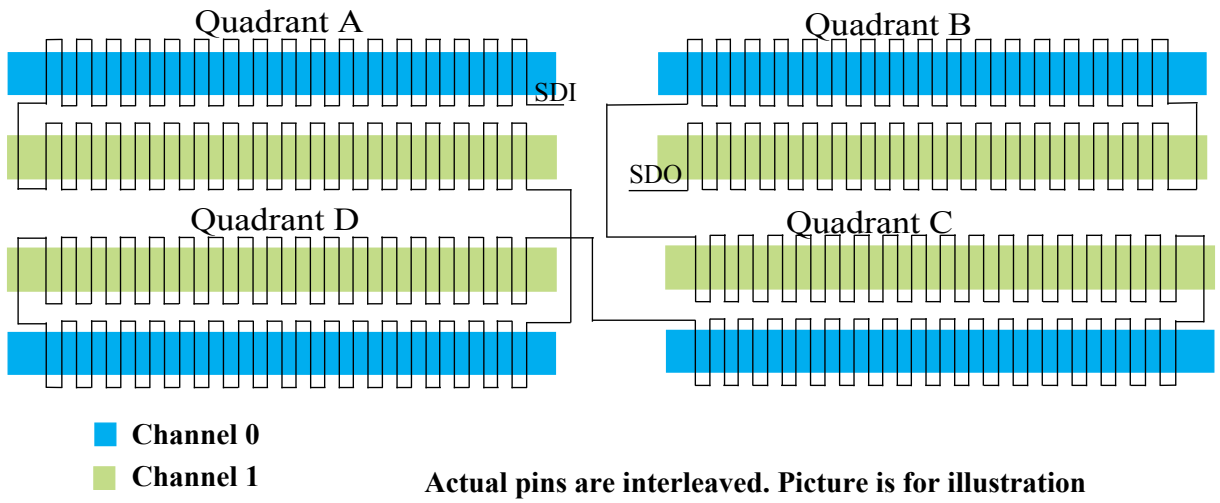


Figure 76 — Boundary Scan Chain 8-Channel Routing Through Quadrants

Table 38 — 8x64 Die Boundary Scan Exit Order (Quadrant A)

Bit#	Location	Signal Name	Bit#	Location	Signal Name	Bit#	Location	Signal Name	Bit#	Location	Signal Name	Bit#	Location	Signal Name
1	G70	DQ1A2	39	J44	DQ1A30	77	L18	DMI1A7	115	K18	DQ0A59	153	M44	DQ0A29
2	J70	DQ1A1	40	G44	DQ1A31	78	L16	DQ1A61	116	H18	DQ0A60	154	M46	DMI0A3
3	L70	DQ1A0	41	G42	DQ1A34	79	J16	DQ1A62	117	H20	DQ0A58	155	K46	DQ0A27
4	L68	DMI1A0	42	J42	DQ1A33	80	G16	DQ1A63	118	K20	DQ0A57	156	H46	DQ0A28
5	J68	DQ1A3	43	L42	DQ1A32	81	J14	CSA1_n	119	M20	DQ0A56	157	H48	DQ0A26
6	G68	DQ1A4	44	L40	DMI1A4	82	L14	CK1A_c	120	L22	DQS0A3_t	158	K48	DQ0A25
7	G66	DQ1A7	45	J40	DQ1A35	83	L12	CK1A_t	121	J22	DQS0A3_c	159	M48	DQ0A24
8	J66	DQ1A6	46	G40	DQ1A36	84	J12	CKEA1	122	H24	DQ0A55	160	L50	DQS0A1_t
9	L66	DQ1A5	47	G38	DQ1A39	85	G10	CA1A2	123	K24	DQ0A54	161	J50	DQS0A1_c
10	K64	DQS1A0_	48	J38	DQ1A38	86	J10	CA1A1	124	M24	DQ0A53	162	H52	DQ0A23
11	H64	DQS1A0_c	49	L38	DQ1A37	87	L10	CA1A0	125	M26	DMI0A6	163	K52	DQ0A22
12	G62	DQ1A10	50	K36	DQS1A2_t	88	L8	CA1A3	126	K26	DQ0A51	164	M52	DQ0A21
13	J62	DQ1A9	51	H36	DQS1A2_c	89	J8	CA1A4	127	H26	DQ0A52	165	M54	DMI0A2
14	L62	DQ1A8	52	G34	DQ1A42	90	G8	CA1A5	128	H28	DQ0A50	166	K54	DQ0A19
15	L60	DMI1A1	53	J34	DQ1A41	91	G6	CA1A8	129	K28	DQ0A49	167	H54	DQ0A20
16	J60	DQ1A11	54	L34	DQ1A40	92	J6	CA1A7	130	M28	DQ0A48	168	H56	DQ0A18
17	G60	DQ1A12	55	L32	DMI1A5	93	L6	CA1A6	131	M30	DQ0A45	169	K56	DQ0A17
18	G58	DQ1A15	56	J32	DQ1A43	94	K4	CA1A9	132	K30	DQ0A46	170	M56	DQ0A16
19	J58	DQ1A14	57	G32	DQ1A44	95	H4	CA1A10	133	H30	DQ0A47	171	M58	DQ0A13
20	L58	DQ1A13	58	G30	DQ1A47	96	J4	CA0A10	134	H32	DQ0A44	172	K58	DQ0A14
21	L56	DQ1A16	59	J30	DQ1A46	97	L4	CA0A9	135	K32	DQ0A43	173	H58	DQ0A15
22	J56	DQ1A17	60	L30	DQ1A45	98	M6	CA0A6	136	M32	DMI0A5	174	H60	DQ0A12
23	G56	DQ1A18	61	L28	DQ1A48	99	K6	CA0A7	137	M34	DQ0A40	175	K60	DQ0A11
24	G54	DQ1A20	62	J28	DQ1A49	100	H6	CA0A8	138	K34	DQ0A41	176	M60	DMI0A1
25	J54	DQ1A19	63	G28	DQ1A50	101	H8	CA0A5	139	H34	DQ0A42	177	M62	DQ0A8
26	L54	DMI1A2	64	G26	DQ1A52	102	K8	CA0A4	140	J36	DQS0A2_c	178	K62	DQ0A9
27	L52	DQ1A21	65	J26	DQ1A51	103	M8	CA0A3	141	L36	DQS0A2_t	179	H62	DQ0A10
28	J52	DQ1A22	66	L26	DMI1A6	104	M10	CA0A0	142	M38	DQ0A37	180	J64	DQS0A0_c
29	G52	DQ1A23	67	L24	DQ1A53	105	K10	CA0A1	143	K38	DQ0A38	181	L64	DQS0A0_t
30	H50	DQS1A1_c	68	J24	DQ1A54	106	H10	CA0A2	144	H38	DQ0A39	182	M66	DQ0A5
31	K50	DQS1A1_t	69	G24	DQ1A55	107	K12	CKEA0	145	H40	DQ0A36	183	K66	DQ0A6
32	L48	DQ1A24	70	H22	DQS1A3_c	108	M12	CK0A_t	146	K40	DQ0A35	184	H66	DQ0A7
33	J48	DQ1A25	71	K22	DQS1A3_t	109	M14	CK0A_c	147	M40	DMI0A4	185	H68	DQ0A4
34	G48	DQ1A26	72	L20	DQ1A56	110	K14	CSA0_n	148	M42	DQ0A32	186	J68	DQ0A3
35	G46	DQ1A28	73	J20	DQ1A57	111	H16	DQ0A63	149	K42	DQ0A33	187	M68	DMI0A0
36	J46	DQ1A27	74	G20	DQ1A58	112	K16	DQ0A62	150	H42	DQ0A34	188	M70	DQ0A0
37	L46	DMI1A3	75	G18	DQ1A60	113	M16	DQ0A61	151	H44	DQ0A31	189	K70	DQ0A1
38	L44	DQ1A29	76	J18	DQ1A59	114	M18	DMI0A7	152	K44	DQ0A30	190	H70	DQ0A2

6 General Purpose IO Test Mode

GPIO Mode defines a reduced pin count test interface on the WideIO2 interface. The intention of this test mode is to enable a DRAM vendor test access path from a reduced number of SOC GPIO (General Purpose IO) pins through the SOC logic to the WideIO2 interface functional pins. GPIO Mode is an alternative test access method to Direct Access Mode.

The SOC logic may, for example, duplicate the CA bus to all channels and provide mux selection of which channel data is output on read cycles. The SOC implementation is not specified, and the WideIO2 device functionality in GPIO Mode is not specified. SOC designers should review their implementation plans with their DRAM vendor(s) to ensure that the SOC implementation will support the DRAM vendors' testing needs.

6 General Purpose IO Test Mode (cont'd)

The WideIO2 device internal function while in GPIO Mode is vendor-specific - only the active pins on the interface are specified:

- TEST = H puts the WideIO2 device into GPIO mode.
- On Write cycles, valid data is received only on Byte 4, along with the corresponding DQS_t, DQS_c, and DMI pins. How this single input byte is distributed to the other bytes in the channel is DRAM vendor specific and not specified.
- On Read cycles, valid data is driven out on all bytes of the channel.
- All CK_t/CK_c, CA, CS_n, CKE, and RST_n signals operate as in normal mode.

Note that only Byte 4 is used on write cycles. Using a single input byte avoids the difficulty of the SOC replicating a single write byte to all eight bytes in the channel while also meeting the tDQSS specification across the width of the channel.

The following WideIO2 interface signals are specified as active and functional in GPIO Mode. "Inputs" signals are driven by the SOC to the WideIO2 device inputs. "Output" signals are driven from the WideIO2 device to the SOC. The SOC implementation for these signals is not specified.

Table 39 — WideIO2 Interface Signals Active in GPIO Mode

WideIO2 interface functional signals in GPIO Mode	Micropillars per quadrant	Micropillars for full WideIO2 interface	Direction	Comments
CK[1:0][D:A]_t, CK[1:0][D:A]_c	4 (2 signals x 2 channels)	16 (4 x 4 quadrants)	Input	Normal function
CA[1:0][D:A][10:0]	22 (11 signals x 2 channels)	88 (22 x 4 quadrants)	Input	Normal function
CS[D:A][3:0]_n	4 (1 signal x 2 ranks /channel x 2 channels)	16 (4 x 4 quadrants)	Input	Normal function
CKE[D:A][3:0]	4 (1 signal x 2 ranks per channel, x 2 channels)	16 (4 x 4 quadrants)	Input	Normal function
Write cycles: DQ[1:0][D:A][39:32] (Byte 4)	16 (8 signals x 2 channels)	64 (16 x 4 quadrants)	Input	One active byte per channel on Byte 4. The DRAM supplier specific test mode implementation may use this byte for full coverage across the full channel.
Read cycles: DQ[1:0][D:A][63:0]	128 (64 signals x 2 channels)	512 (128 x 4 quadrants)	Output	Normal function. Valid data is output on all bytes of the channel.
Write cycles: DQS[1:0][D:A]2_t DQS[1:0][D:A]2_c	4 (2 signals x 2 channels)	16 (4 x 4 quadrants)	Input	One active DQS-pair per channel, associated with Byte 4. The DRAM supplier specific test mode implementation may use this pair for full coverage across the full channel.
Read cycles: DQS[1:0][D:A][3:0]_t DQS[1:0][D:A][3:0]_c	16 (8 signals x 2 channels)	64 (16 x 4 quadrants)	Output	Normal function. Valid DQS-pair are ouput per byte-pair.
Write cycles: DMI[1:0][D:A]4	2 (1 signal x 2 channels)	8 (2 x 4 quadrants)	Input	One active DMI per channel, associated with Byte 4. The DRAM supplier specific test mode implementation may use this pin for full coverage across the full channel.
Read cycles: DMI[1:0][D:A][7:0]	16 (8 signal x 2 channels)	64 (16 x 4 quadrants)	Output	Normal function. Valid DMI output per byte.
RST[3:0]_n	-	4 in quadrant C. 1 per slice	Input	Normal function.
TEST	-	1 in quadrant C	Input	Asserting TEST = H causes the WideIO2 device to enter GPIO mode.
Total GPIO Mode active micropillars	**56 Writes 194 Reads**	**229 Writes 781 Reads 5 in quadrant C**		

7 Post Package Repair (PPR)

WideIO2 supports Fail Row address repair as optional feature and it is readable through MR0 OP[5]

PPR provides a repair method in the system and Fail Row address can be repaired by the electrical programming of Electrical-fuse scheme.

With PPR, WideIO2 can correct at least1Row per Bank.

Electrical-fuse cannot be switched back to un-fused states once it is programmed. The controller should prevent unintended the PPR mode entry and repair.

7.1 Fail Row Address Repair

The following is procedure of PPR.

1. Before entering 'PPR' mode, the user needs to check the availability of repairable row with MR13 [OP7:OP0]. If there is no available row to the target bank, then user should not enable PPR. And then, all banks must be Precharged
2. Enable PPR using MR2 bit "OP6=1" and wait tMRD
3. Issue ACT command with Fail Row address
4. Wait tPGM to allow DRAM repair target Row Address internally and issue PRE
5. Wait tPGM_Exit after PRE which allow DRAM to recognize repaired Row address
6. Exit PPR with setting MR2 bit "OP6=0"
7. WideIO2 will accept any valid command after tPGMPST
8. In More than one fail address repair case, Repeat Step 2 to 7

Once PPR mode is exited, to confirm if target row is repaired correctly, host can verify by writing data into the target row and reading it back after PPR exit with MR2 [OP6=0] and tPGMPST

Table 40 and Timing diagram show PPR related MR bits and its operation

Table 40 — PPR Setting

MR2 [OP6]	Description
0	PPR mode exit
1	PPR mode entry

NOTE 1 During tPGM, any other commands (including refresh) are not allowed on each slice.

NOTE 2 With one PPR command, only one row can be repaired at one time per slice.

NOTE 3 When PPR procedure is done, reset command is required before normal operation.

NOTE 4 During PPR, memory contents is not refreshed and may be lost.

7.1 Fail Row Address Repair (cont'd)

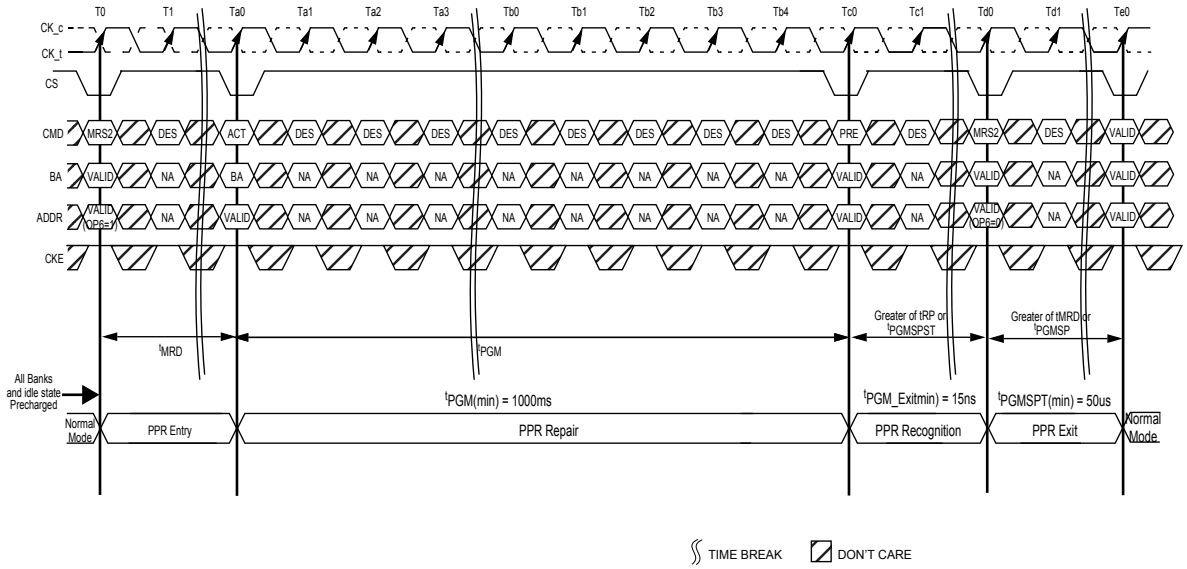


Figure 77 — Fail Row Repair

7.2 Programming PPR support in MR13

PPR is an optional feature of WideIO2 so Host can recognize if DRAM is supporting PPR or not by reading out MR13. If no resources are avilable then PPR is not supported.

7.3 Required Timing Parameters

Repair requires additional time period to repair Fail Row Address into spare Row address and the followings are requirement timing parameters for PPR

Table 41 — PPR Timing Parameters

Parameter	Symbol	min	Unit	Note
PPR Programming Time	tPGM	1000	ms	
PPR Exit Time	tPGM_Exit	15	ns	
New Address Setting time	tPGMPST	50	us	

8 AC & DC Operating Conditions

8.1 Absolute Maximum DC Rating

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

Parameter	Symbol	Rating	Unit	Notes
Voltage on VDD1 relative to VSS	VDD1	-0.4 ~ 2.3	V	1
Voltage on VDD2 relative to VSS	VDD2	-0.4 ~ 1.6	V	1
Voltage on VDDQ relative to VSSQ	VDDQ	-0.4 ~ 1.6	V	1
Storage Temperature	TSTG	-55 ~ 125	°C	2

NOTE 1 See "Power-Ramp" section in "Power-up, Initialization, and Power-Off" for relationships between power supplies.
NOTE 2 Storage Temperature is the case surface temperature on the center/top side of the WideIO2 device. For the measurement conditions, please refer to JESD51-2 standard.
Operation or timing that is not specified is illegal, and after such an event, in order to guarantee proper operation, the WideIO2 Device must be powered down and then restarted through the specialized initialization sequence before normal operation can continue.

8.2 DC Operating Conditions

Table 42 — DC operating conditions

Parameter	Symbol	Min	Typ	Max	Unit
Core Power 1	VDD1	1.70	1.80	1.95	V
Core Power 2	VDD2	1.06	1.10	1.17	V
I/O Buffer Power	VDDQ	1.06	1.10	1.17	V

NOTE 1 When VDD2 is used, VDD1 uses significantly less power than VDD2.
NOTE 2 The voltage range is for DC voltage only. DC is defined as the voltage supplied at the DRAM nearest to the SOC and is inclusive of all noise up to 1 MHz at the DRAM micropillar.

8.3 Input Leakage Current

Table 43 — Input Leakage Current

Parameter	Symbol	Min	Max	Unit	Note
Input Leakage current For CA, CKE, CS_n, CK_t, CK_c, and DQ, DQS_t, DQS_c, DMI when in input mode. Any input 0V ≤ VIN ≤ VDD2(VDDQ) (All other pins not under test = 0V)	IL	-5	5	uA	1

NOTE 1 Although DM is for input only, the DM leakage shall match the DQ and DQS_t/DQS_c output leakage specification.

8.4 Operating Temperature

Table 44 — Operating Temperature

Parameter		Symbol	Min	Max	Unit	Note
Operating Temperature	Standard	T_{OPER}	-25	85	°C	1
	Elevated		85	105		1

NOTE 1 Operating Temperature is thebackside temperature of center of WideIO2 DRAM device
NOTE 2 Some applications require operating the WideIO2 DRAM in the maximum temperature conditons of the High Temperature Range between 85 C and 105 C case temperature. For WideIO2 DRAM devices, some derating is neccessary to operate in this range. See Status Register Read section.
NOTE 3 Either the device case temperature or the temperature sensor may be used to set an appropriate refresh rate, determine the need for AC timing de-rating and/or monitor the operating temperature.

9 AC and DC Input Specifications

9.1 AC and DC Input Measurement Levels

Table 45 — Receiver Input Voltage Level Specification

Parameter	Symbol	Min	Max	Unit
Input high Voltage	VIH	0.70 * VDDQ	VDDQ + 0.2	V
Input low Voltage	VIL	-0.2	0.30 * VDDQ	V

NOTE 1 VDD2 for CA micropillars, VDDQ for DQ/DMI/DQS micropillars.

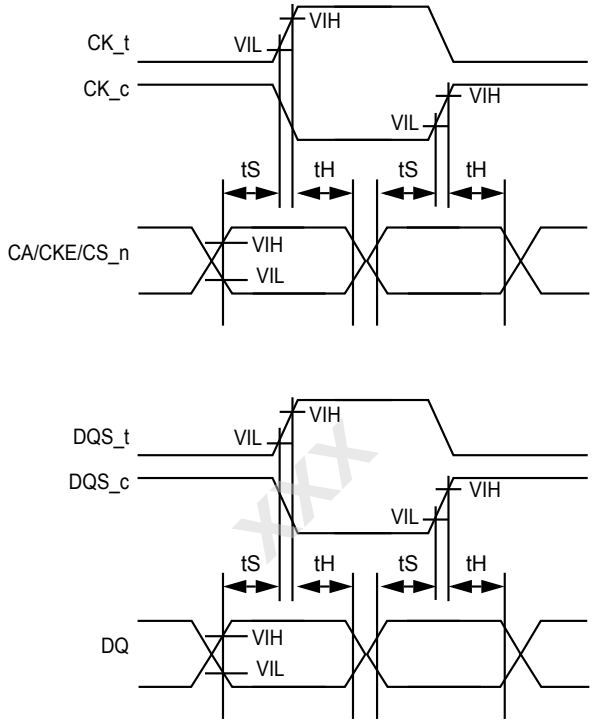


Figure 78 — Definition of receiver input levels VIH/VIL

10 AC and DC Output Specifications

10.1 Output Timing Reference Load

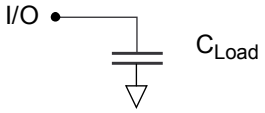


Figure 79 — Output Timing Reference Load

Table 46 — Output Timing Reference Load

Parameter	Symbol	Max	Unit
Output Timing Reference Load	CLOAD	1.4[1]	pF

NOTE 1 Self capacitance of driving device not included.

10.2 Definition of Output Voltage Levels and Output Rise/Fall Times (T_R, T_F)

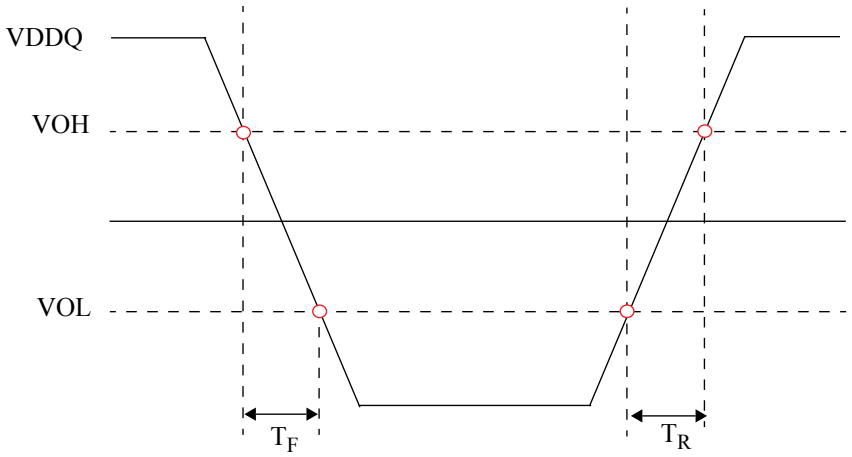


Figure 80 — Output Voltage Levels VOH/VOL and Output Timing T_R/T_F Definition

Table 47 — Transmitter Output Voltage Level Specification

Parameter	Symbol	Min	Max	Unit
Output High Voltage	VOH	0.70 * VDDQ	NA	V
Output Low Voltage	VOL	NA	0.30 * VDDQ	V

With the reference load for timing measurements, the output rise/fall time for falling and rising edges is defined and measured between VOL and VOH for single ended signals.

Table 48 — Output Rise/Fall Time Measurement Definition

Description	Measured		Defined by
	from	to	
Output rise time	VOL	VOH	T_{VOH} - T_{VOL}
Output fall time	VOH	VOL	T_{VOL} - T_{VOH}

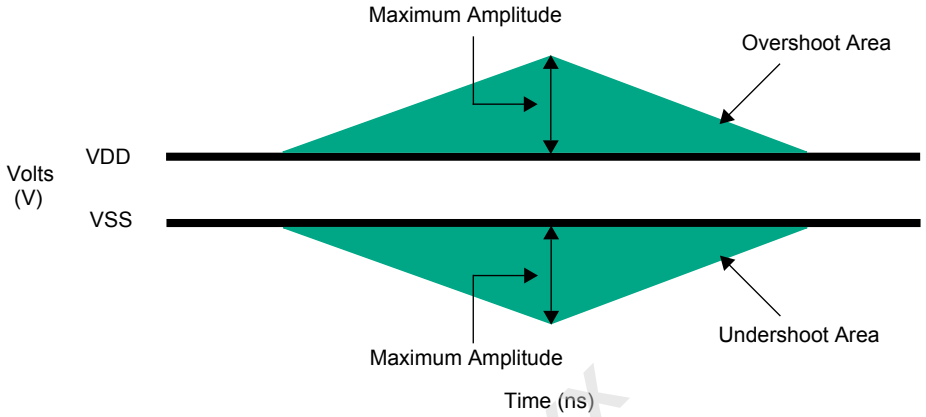
Table 49 — Transmitter Output Rise/Fall Time Specification

Parameter	Symbol	800 MT/s		1067 MT/s		Unit
		Min	Max	Min	Max	
Transmitter Output Rise/Fall Time	TRF	N/A	180	N/A	135	ps

10.3 Over/Undershoot

Table 50 — Over/Undershoot

Parameter	800	1067	Units
	Max	Max	
Maximum peak amplitude allowed for overshoot area (see Figure)	0.35	0.35	V
Maximum peak amplitude allowed for undershoot area (see Figure)	0.35	0.35	V
Maximum area above VDD2/VDDQ (see Figure)	0.12	0.10	V-ns
Maximum area below VSS/VSSQ (see Figure)	0.12	0.10	V-ns



NOTE 1 VDD stands for VDD2 for CA[10:0], CK_t, CK_c, CS_n, CKE, and RST_n. VDD stands for VDDQ for DQ, DMI, DQS_t, and DQS_c.
NOTE 2 VSS stands for VSS for CA[10:0], CK_t, CK_c, CS_n, and CKE. VSS stands for VSSQ for DQ, DMI, DQS_t, and DQS_c.
NOTE 3 Absolute maximum requirements apply.
NOTE 4 Maximum peak amplitude values are referenced from actual VDD and VSS values.
NOTE 5 Maximum area values are referenced from maximum operating VDD and VSS values.

Figure 81 — AC Overshoot and Undershoot Definition

11 Input/Output Capacitance

Table 51 — DRAM Input/Output Capacitance

Parameter	Symbol	Min	Max	Unit
Input/Output capacitance, 1-high DRAM stack (4x64)	Cio1h	N/A	0.5	pF
Input/Output capacitance, 2-high DRAM stack (4x64) P2P	Cio2h (P2P)	N/A	0.7	pF
Input/Output capacitance, 4-high DRAM stack (4x64) P22P	Cio4h (P22P)	N/A	1.4	pF
Input/Output capacitance, 1-high DRAM stack (8x64)	Cio1h	N/A	0.5	pF
Input/Output capacitance, 2-high DRAM stack (8x64) P22P	Cio2h (P22P)	N/A	1.0	pF

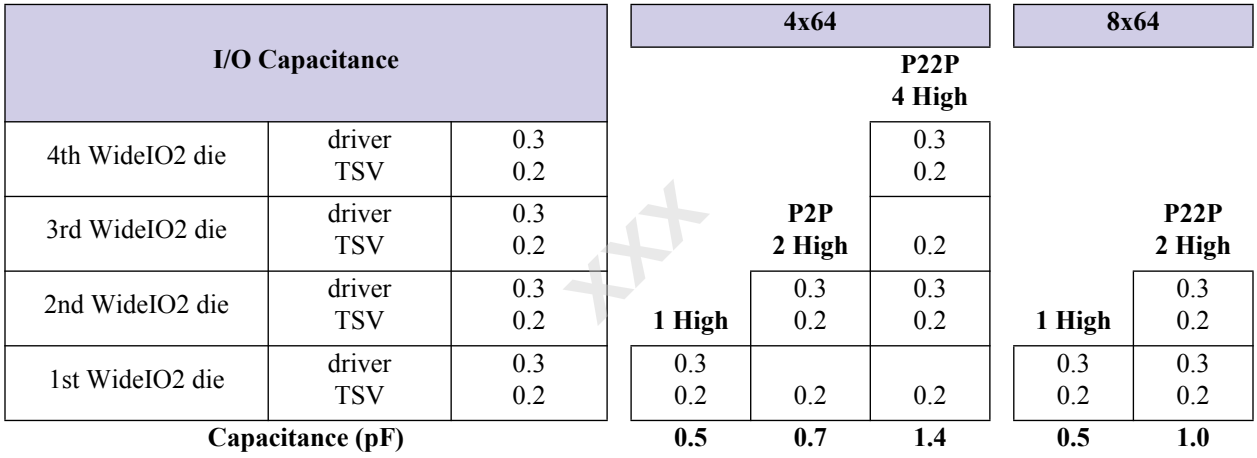


Figure 82 — DRAM Input/Output Capacitance for Stack Option

12 Electrical Characteristics and AC Timings

12.1 Command input signal timing definition

12.1.1 Command input signal timing definition for CS_n

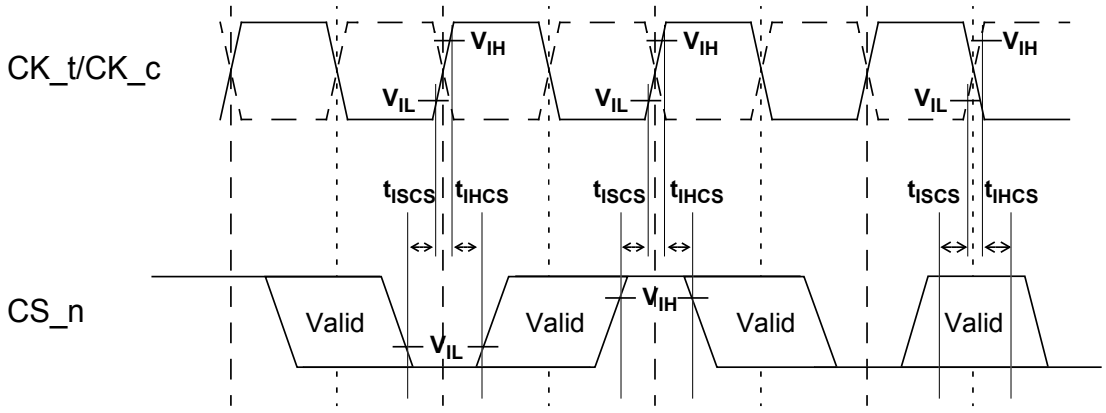


Figure 83 — Command Input Setup and Hold Timing for CS_n

12.1.2 Command input signal timing definition for CA

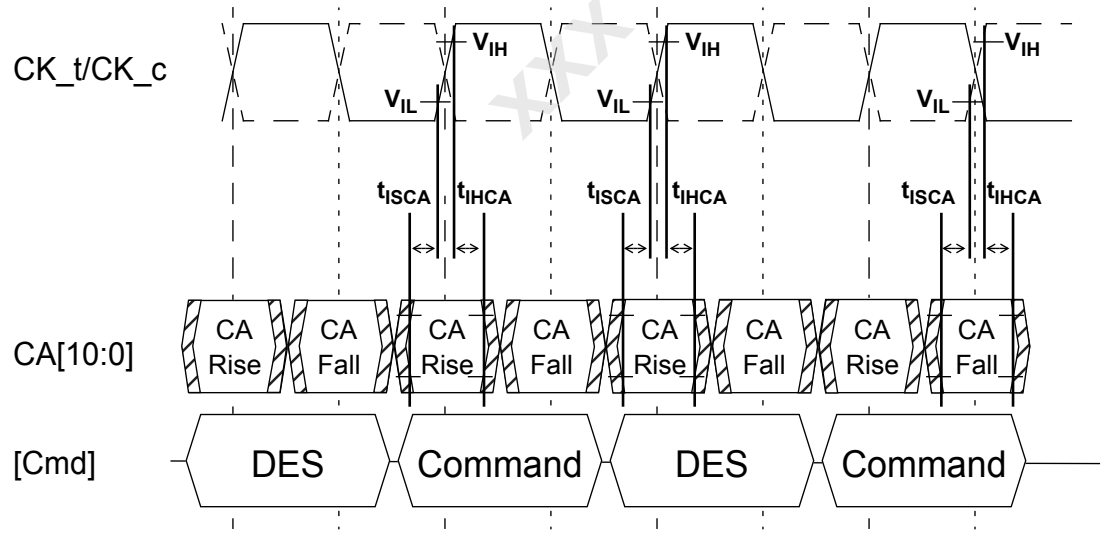
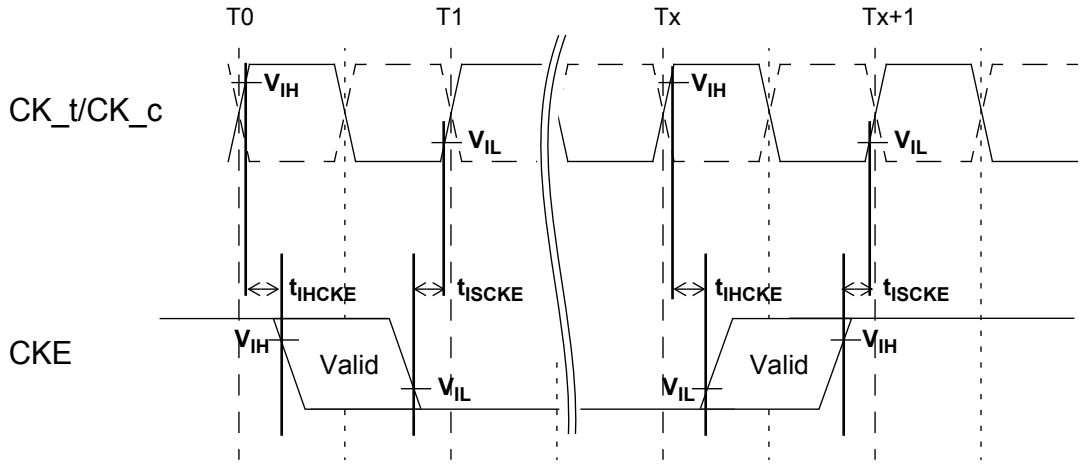


Figure 84 — Command Input Setup and Hold Timing for CA

12.1.3 Command input signal timing definition for CKE



NOTE 1 After CKE is registered LOW, CKE signal level shall be maintained below VILCKE for tCKE specification (LOW pulse width).
NOTE 2 After CKE is registered HIGH, CKE signal level shall be maintained above VIHCKE for tCKE specification (HIGH pulse width).

Figure 85 — Command Input Setup and Hold Timing for CKE

12.2 Clock Specification

12.2.1 Definition for tCK(avg) and nCK

tCK(avg) is calculated as the average clock period across any consecutive 200 cycle window, where each clock period is calculated from rising edge to rising edge.

$$t_{CK(avg)} = \frac{\left(\sum_{j=1}^{N} t_{CK_j}\right)}{N} \quad , where\ N=200$$

The unit ‘tCK(avg)’ represents the actual clock average tCK(avg) of the input clock under operation. Unit ‘nCK’ represents one clock cycle of the input clock, counting the actual clock edges.

tCK(avg) may change by up to +/-1% within a 100 clock cycle window, provided that all jitter and timing specs are met.

12.2.2 Definition for tCK(abs)

tCK(abs) is defined as the absolute clock period, as measured from one rising edge to the next consecutive rising edge. tCK(abs) is not subject to production test.

12.2.3 Definition for tCHH(avg), tCHM(avg), tCLL(avg) and tCLM(avg)

tCHH(avg) and tCHM(avg) are defined as average high pulse width, as calculated across any consecutive 200 high pulses.

$$t_{CHH(avg)} = \frac{\left(\sum_{j=1}^{N} t_{CHHj}\right)}{N * t_{CK(avg)}} \text{ , where } N=200$$

$$t_{CHM(avg)} = \frac{\left(\sum_{j=1}^{N} t_{CHMj}\right)}{N * t_{CK(avg)}} \text{ , where } N=200$$

tCLL(avg) and tCLM(avg) are defined as average low pulse width, as calculated across any consecutive 200 high pulses.

$$t_{CLL(avg)} = \frac{\left(\sum_{j=1}^{N} t_{CLLj}\right)}{N * t_{CK(avg)}} \text{ , where } N=200$$

$$t_{CLM(avg)} = \frac{\left(\sum_{j=1}^{N} t_{CLMj}\right)}{N * t_{CK(avg)}} \text{ , where } N=200$$

12.2.4 Definition for tJIT(per)

tJIT(per) is the single period jitter defined as the largest deviation of any signal tCK from tCK(avg).
tJIT(per) = Min/max of {tCKi - tCK(avg) where i = 1 to 200}.
tJIT(per) is not subject to production test.

12.2.5 Definition for tJIT(cc)

tJIT(cc) is defined as the absolute difference in clock period between two consecutive clock cycles.
tJIT(cc) = Max of |{tCKi +1 - tCKi}|.
tJIT(cc) defines the cycle to cycle jitter.
tJIT(cc) is not subject to production test.

12.2.6 Definition for tERR(nper)

tERR(nper) is defined as the cumulative error across n multiple consecutive cycles from tCK(avg).
tERR(nper),act is the actual clock jitter over n cycles for a given system. tERR(nper),allowed is the specified allowed clock period jitter over n cycles. tERR(nper) is not subject to production test.

$$t_{ERR(nper)} = \left(\sum_{j=1}^{i+n-1} t_{CKj} \right) - n \times t_{CK(avg)}$$

tERR(nper),min can be calculated by the formula shown below:

$t_{ERR(nper),\,min} = (1+ 0.68LN(n)) \times t_{JIT}(per),\,min$

tERR(nper),max can be calculated by the formula shown below:

$t_{ERR(nper),\,max} = (1+ 0.68LN(n)) \times t_{JIT}(per),\,max$

Using these equations, tERR(nper) tables can be generated for each tJIT(per),act value.

12.2.7 Definition for duty cycle jitter tJIT(duty)

tJIT(duty) is defined with absolute and average specification of tCHM / tCLM.

$t_{CHM}(abs),\,min - t_{CHM}(avg),min),\,(t_{CLM}(abs),min - t_{CLM}(avg),min)) \times t_{CK}(avg)$

$t_{CHM}(abs),max - t_{CHM}(avg),max),\,(t_{CLM}(abs),max - t_{CLM}(avg),max)) \times t_{CK}(avg)$

12.2.8 Definition for tCK(abs), tCHM(abs) and tCLM(abs)

These parameters are specified per their average values, however it is understood that the following relationship between the average timing and the absolute instantaneous timing holds at all times.

Table 52 — Definition for tCK(abs), tCHM(abs), tCLM(abs)

Parameter	Symbol	Min	Unit	Notes
Absolute Clock Period	tCK(abs)	tCK(avg),min + tJIT(per),min	ps	
Absolute Clock HIGH Pulse Width	tCHM(abs)	tCHM(avg),min + tJIT(duty),min / tCK(avg),min	tCK(avg)	1
Absolute Clock LOW Pulse Width	tCLM(abs)	tCLM(avg),min + tJIT(duty),min / tCK(avg),min	tCK(avg)	1

12.3 Period Clock Jitter

WideIO2 devices can tolerate some clock period jitter without core timing parameter de-rating. This section describes device timing requirements in the presence of clock period jitter (tJIT(per)) in excess of the values found in Table (AC Timing) and how to determine cycle time de-rating and clock cycle de-rating.

12.3.1 Clock period jitter effects on core timing parameters (tRCD, tRP, tRTP, tWR, tWTR, tRC, tRAS, tRRD, tFAW)

Core timing parameters extend across multiple clock cycles. Period clock jitter will impact these parameters when measured in numbers of clock cycles. When the device is operated with clock jitter within the specification limits, the WideIO2 device is characterized and verified to support tnPARAM = RU{tPARAM / tCK(avg)}.

When the device is operated with clock jitter outside specification limits, the number of clocks or tCK(avg) may need to be increased based on the values for each core timing parameter.

12.3.1.1 Cycle time de-rating for core timing parameters

For a given number of clocks (tnPARAM), for each core timing parameter, average clock period (tCK(avg)) and actual cumulative period error (tERR(tnPARAM),act) in excess of the allowed cumulative period error (tERR(tnPARAM),allowed), the equation below calculates the amount of cycle time de-rating (in ns) required if the equation results in a positive value for a core timing parameter.

$$CycleTimeDerating = MAX\left\{\left(\frac{t_{PARAM} + t_{ERR(tnPARAM),\,act} - t_{ERR(tnPARAM),\,allowed}}{t_{nPARAM}} - t_{CK(avg)}\right), 0\right\}$$

A cycle time derating analysis should be conducted for each core timing parameter. The amount of cycle time derating required is the maximum of the cycle time de-ratings determined for each individual core timing parameter.

12.3.1.2 Clock Cycle de-rating for core timing parameters

For a given number of clocks (tnPARAM) for each core timing parameter, clock cycle de-rating should be specified with amount of period jitter (tJIT(per)).

For a given number of clocks (tnPARAM), for each core timing parameter, average clock period (tCK(avg)) and actual cumulative period error (tERR(tnPARAM),act) in excess of the allowed cumulative period error (tERR(tnPARAM),allowed), the equation below calculates the clock cycle derating (in clocks) required if the equation results in a positive value for a core timing parameter.

$$CycleTimeDerating = RU\left\{\frac{t_{PARAM} + t_{ERR(tnPARAM),\,act} - t_{ERR(tnPARAM),\,allowed}}{t_{CK(avg)}}\right\} - t_{nPARAM}$$

12.3.2 Clock jitter effects on Command/Address timing parameters (tISCA, tIHCA, tISCS, tIHCS,tISCKE, tIHCKE, tISb, tIHb, tISCKEb, tIHCKEb)

These parameters are measured from a command/address signal (CKE, CS, CA0 - CA9) transition edge to its respective clock signal (CK_t/CK_c) rising edge. The spec values are not affected by the amount of clock jitter applied (i.e., tJIT(per), as the setup and hold are relative to the clock rising signal edge that latches the command/address. Regardless of clock jitter values, these values shall be met.

12.3.3 Clock jitter effects on Read timing parameters

12.3.3.1 tRPRE

When the device is operated with input clock jitter, tRPRE needs to be de-rated by the actual period jitter (tJIT(per),act,max) of the input clock in excess of the allowed period jitter (tJIT(per),allowed,max). Output de-ratings are relative to the input clock.

$$tRPRE\ (min,\ derated) = 0.9 - \left(\frac{t_{JIT(per),act,max} - t_{JIT(per),allowed,max}}{t_{CK(avg)}}\right)$$

12.3.3.2 tLZ(DQ), tHZ(DQ), tDQSCK, tLZ(DQS), tHZ(DQS)

These parameters are measured from a specific clock edge to a data signal (DMn, DQm.: n=0,1,2,3. m=0.31) transition and will be met with respect to that clock edge. Therefore, they are not affected by the amount of clock jitter applied (i.e., tJIT(per)).

12.3.4 Clock jitter effects on Write timing parameters

12.3.4.1 tDS, tDH

These parameters are measured from a data signal (DMn, DQm.: n=0,1,2,3. m=0 .31) transition edge to its respective data strobe signal (DQSn_t, DQSn_c : n=0,1,2,3) rising edge. The spec values are not affected by the amount of clock jitter applied (i.e., tJIT(per), as the setup and hold are relative to the data strobe signal edge that latches the data. Regardless of clock jitter values, these values shall be met.

12.3.5 tDSS, tDSH

These parameters are measured from a data strobe signal (DQSx_t, DQSx_c) rising edge to its respective clock signal (CK_t/CK_c) rising edge. The spec values are not affected by the amount of clock jitter applied (i.e. tJIT(per), as the setup and hold of the data strobes are relative to the corresponding clock rising signal edge. Regardless of clock jitter values, these values shall be met.

12.3.5.1 tDQSS

This parameter is measured from a data strobe signal (DQSx_t, DQSx_c) rising edge to the subsequent clock signal (CK_t/CK_c) rising edge. When the device is operated with input clock jitter, this parameter needs to be de-rated by the actual period jitter tJIT(per),act of the input clock in excess of the allowed period jitter tJIT(per),allowed.

$$t_{DQSS\ (min,\ derated)} = 0.75 - \frac{t_{JIT(per),act,min} - t_{JIT(per),allowed,min}}{t_{CK(avg)}}$$

$$t_{DQSS\ (max,\ derated)} = 1.25 - \frac{t_{JIT(per),act,max} - t_{JIT(per),allowed,max}}{t_{CK(avg)}}$$

12.4 AC Timing Definitions

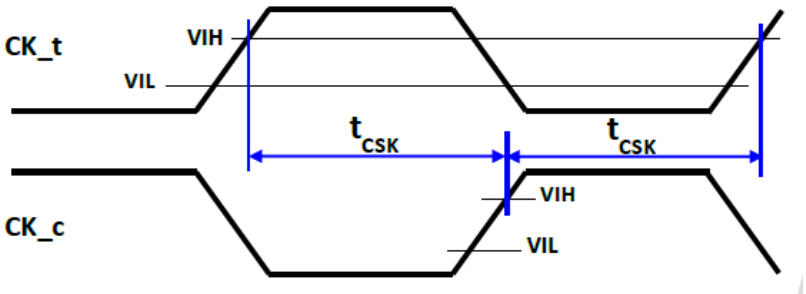


Figure 86 — tCSK definition (Intel 1821.04)

12.4 AC Timing Definitions (cont'd)

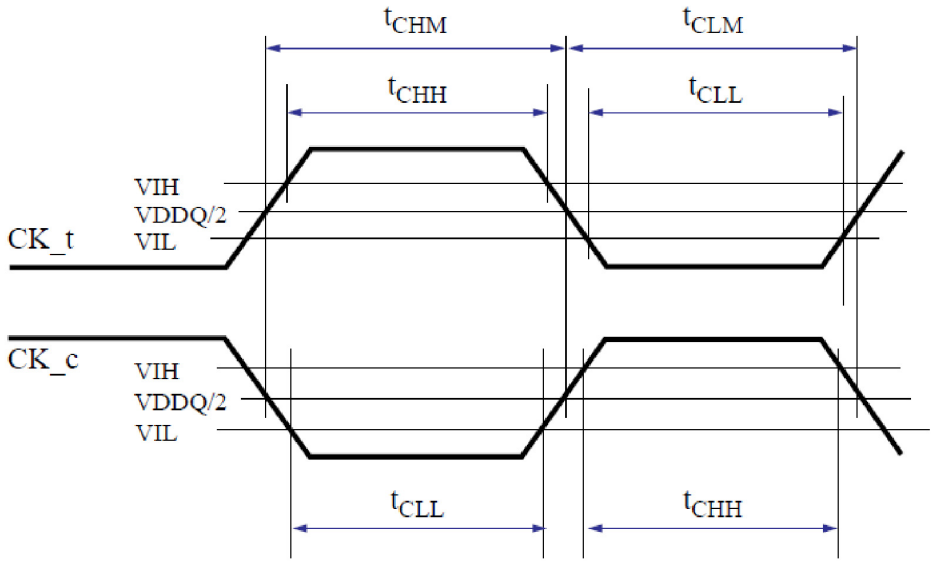


Figure 87 — CK pulse width definitions

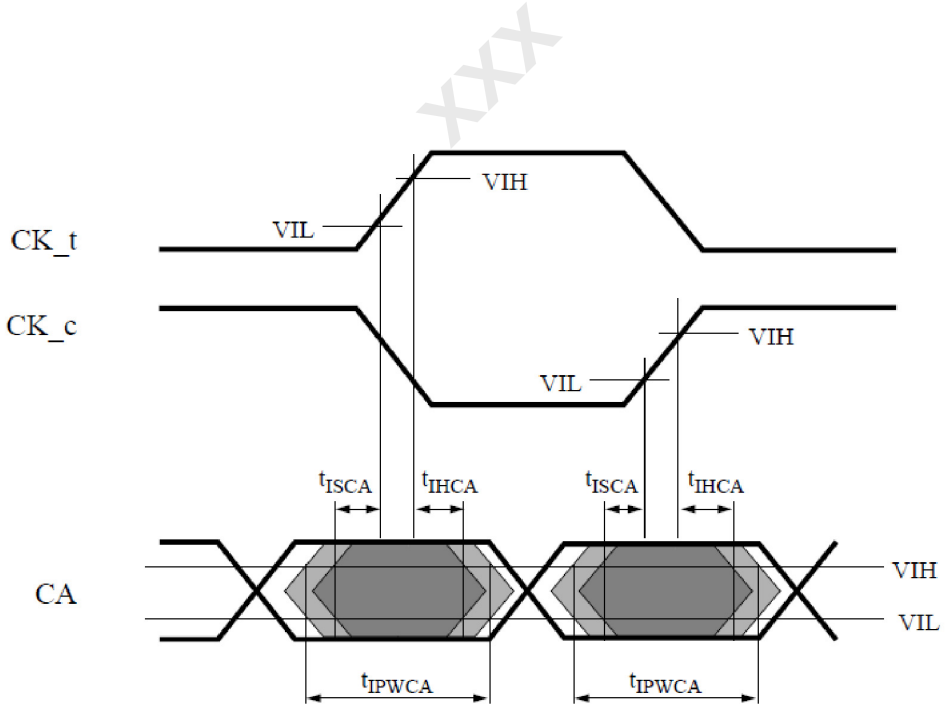


Figure 88 — CA/CKE Input setup, hold and CA pulse width definition

12.4 AC Timing Definitions (cont'd)

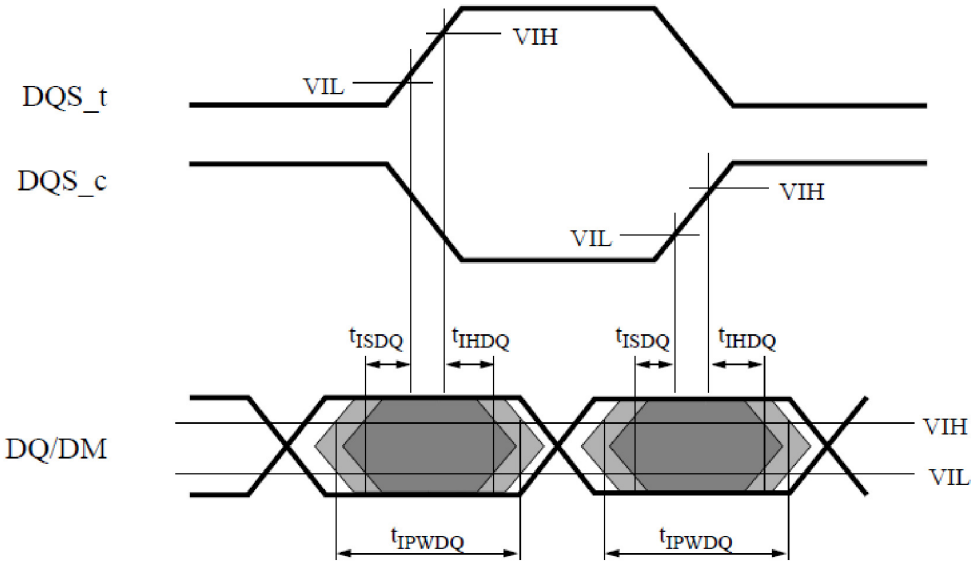


Figure 89 — DQ/DM input setup, hold and pulse width definition

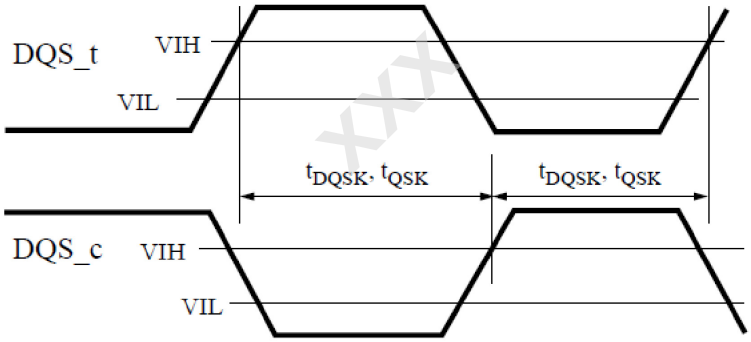


Figure 90 — tDQSK , tQSK definition

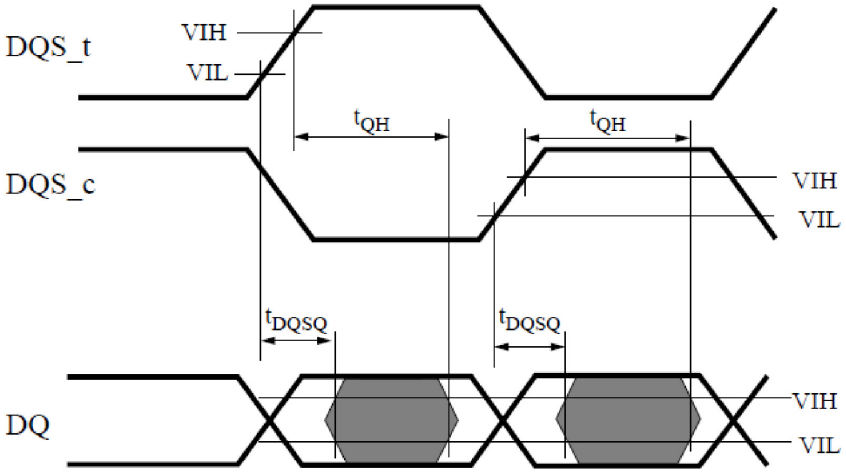


Figure 91 — tDQSQ, tQH definition

12.5 AC Timings

Table 53 — AC Timings

Parameter	Symbol	min tCK	Min/ Max	Data Rate		Unit	Note
				800 MT/s	1066 MT/s		
Clock Timing							
Average clock period	$t_{CK(avg)}$		MIN	2.5	1.876	ns	
			MAX	100		ns	
Average high pulse width at VIH	$t_{CHH(avg)}$		MIN	0.38	0.38	$t_{CK(avg)}$	
Average low pulse width at VIL	$t_{CLL(avg)}$		MIN	0.38	0.38	$t_{CK(avg)}$	
Average high pulse width at VDDQ/2	$t_{CHM(avg)}$		MIN	0.45	0.45	$t_{CK(avg)}$	
			MAX	0.55	0.55	$t_{CK(avg)}$	
Average low pulse width at VDDQ/2	$t_{CLM(avg)}$		MIN	0.45	0.45	$t_{CK(avg)}$	
			MAX	0.55	0.55	$t_{CK(avg)}$	
Absolute Clock Period	$t_{CK(abs)}$			$t_{CK(avg)}$ MIN + $t_{JIT(per)}$ MIN		ns	
Absolute high pulse width at VIH	$t_{CHH(abs)}$		MIN	0.36	0.36	$t_{CK(avg)}$	
Absolute low pulse width at VIL	$t_{CLL(abs)}$		MIN	0.36	0.36	$t_{CK(avg)}$	
Absolute high pulse width at VDDQ/2	$t_{CHM(abs)}$		MIN	0.43	0.43	$t_{CK(avg)}$	
			MAX	0.57	0.57	$t_{CK(avg)}$	
Absolute low pulse width at VDDQ/2	$t_{CLM(abs)}$		MIN	0.43	0.43	$t_{CK(avg)}$	
			MAX	0.57	0.57	$t_{CK(avg)}$	
CK_t to CK_c skew	t_{CSK}		MIN	0.43	0.43	$t_{CK(avg)}$	
			MAX	0.57	0.57	$t_{CK(avg)}$	
Clock period jitter	$t_{JIT(per)}$		MIN	-90	-80	ps	
			MAX	90	80	ps	
Max Clock Cycle to Cycle period jitter	$t_{JIT(cc)}$		MIN	-180	-160	ps	
			MAX	180	160	ps	
Duty cycle jitter (with supported jitter)	$t_{JIT(duty),allowed}$		MIN	$\min((t_{CH(abs),min} - t_{CH(avg),min}),(t_{CL(abs),min} - t_{CL(avg),min})) \times t_{CK(avg)}$		ps	
			MAX	$\max((t_{CH(abs),max} - t_{CH(avg),max}),(t_{CL(abs)}, \max - t_{CL(avg),max}) \times tC_{K(avg)}$		ps	
Cumulative errors across 2 cycles	$t_{ERR(2per),allowed}$		MIN	-147	-132	ps	
			MAX	147	132	ps	
Cumulative errors across 3 cycles	$t_{ERR(3per),allowed}$		MIN	-175	-157	ps	
			MAX	175	-157	ps	
Cumulative errors across 4 cycles	$t_{ERR(4per),allowed}$		MIN	-194	-175	ps	
			MAX	194	175	ps	
Cumulative errors across 5 cycles	$t_{ERR(5per),allowed}$		MIN	-209	-188	ps	
			MAX	209	188	ps	
Cumulative errors across 6 cycles	$t_{ERR(6per),allowed}$		MIN	-222	-200	ps	
			MAX	222	200	ps	

Table 53 — AC Timings

Parameter	Symbol	min tCK	Min/ Max	Data Rate		Unit	Note
				800 MT/s	1066 MT/s		
Cumulative errors across 7 cycles	$t_{ERR(7per),allowed}$		MIN	-232	-209	ps	
			MAX	232	209	ps	
Cumulative errors across 8 cycles	$t_{ERR(8per),allowed}$		MIN	-241	-217	ps	
			MAX	241	217	ps	
Cumulative errors across 9 cycles	$t_{ERR(9per),allowed}$		MIN	-249	-224	ps	
			MAX	249	224	ps	
Cumulative errors across 10 cycles	$t_{ERR(10per),allowed}$		MIN	-257	-231	ps	
			MAX	257	231	ps	
Cumulative errors across 11 cycles	$t_{ERR(11per),allowed}$		MIN	-263	-237	ps	
			MAX	263	237	ps	
Cumulative errors across 12 cycles	$t_{ERR(12per),allowed}$		MIN	-269	-242	ps	
			MAX	269	242	ps	
Cumulative errors across n = 13, 14, 15..., 19, 20 cycles	$t_{ERR(nper),allowed}$		MIN	$(1 + 0.68\ln(n)) \times t_{JIT(per),allowed,min}$		ps	
			MAX	$(1 + 0.68\ln(n)) \times t_{JIT(per),allowed,max}$		ps	
Read Parameters							
DQS Output Access Time from CK_t/CK_c rising edge	t_{DQSCK}		MIN	2		ns	1,4
			MAX	5		ns	1,4
DQS Output Access Time from CK_t/CK_c - Temperature Variation	t_{DQSCK_temp}		MAX	TBD	TBD	ps/C	2,4
DQS Output Access Time from CK_t/CK_c - Voltage Variation	t_{DQSCK_volt}		MAX	TBD	TBD	ps/mV	3,4
DQS_t to DQS_c output skew	t_{QSK}		MIN	0.38	0.38	$t_{CK(avg)}$	
			MAX	0.62	0.62	$t_{CK(avg)}$	
DQ to DQS_t/DQS_c output skew	t_{DQSQ}		MAX	387	291	ps	
DQ output hold time from DQS_t/DQS_c	t_{QH}		MIN	812	610	ps	
Read postamble	t_{RPST} standard		MIN	0.3	0.3	$t_{CK(avg)}$	
	t_{RPST} extended		MIN	1 + 0.3	1 + 0.3	$t_{CK(avg)}$	5
Read preamble	t_{RPRE}		MIN	0.9	0.9	$t_{CK(avg)}$	
DQS low-Z from clock	$t_{LZ(DQS)}$		MIN	$t_{DQSCK(min)}$ -3 00		ps	
DQ low-Z from clock	$t_{LZ(DQ)}$		MIN	$t_{DQSCK(min)}$ - 300		ps	
DQS high-Z from clock	$t_{HZ(DQS)}$		MAX	$t_{DQSCK(max)}$ - 100		ps	
DQ high-Z from clock	$t_{HZ(DQ)}$		MAX	$t_{DQSCK(max)} + (1.4 \times t_{DQSQ(max)})$		ps	
Command path disable delay	t_{CPDED}		MIN	2	2	$t_{CK(avg)}$	
Write Parameters							
DQ and DMI input setup time	t_{ISDQ}		MIN	270	200	ps	6
DQ and DMI input hold time	t_{IHDQ}		MIN	270	200	ps	6

Table 53 — AC Timings

Parameter	Symbol	min tCK	Min/ Max	Data Rate		Unit	Note
				800 MT/s	1066 MT/s		
DQ and DMI input pulse width	t_{IPWDQ}		MIN	0.315	0.315	$t_{CK(avg)}$	6
DQS_t to DQS_c input skew	t_{DQSK}		MIN	0.4	0.4	$t_{CK(avg)}$	
			MAX	0.6	0.6	$t_{CK(avg)}$	
Write command to first DQS latching transition	t_{DQSS}		MIN	0.75	0.75	$t_{CK(avg)}$	
			MAX	1.25	1.25	$t_{CK(avg)}$	
DQS rising edge to CK setup time	t_{DSS}		MIN	0.2	0.2	$t_{CK(avg)}$	
DQS rising edge hold time from CK	t_{DSH}		MIN	0.2	0.2	$t_{CK(avg)}$	
Write postamble	t_{WPST}		MIN	0.4	0.4	$t_{CK(avg)}$	
Write preamble	t_{WPRE}		MIN	0.9	0.9	$t_{CK(avg)}$	
CKE Input parameters							
CKE min. pulse width (high and low pulse width)	t_{CKE}		MIN	3	3	$t_{CK(avg)}$	6
CKE input setup time	t_{ISCKE}		MIN	0.2	0.2	$t_{CK(avg)}$	6
CKE input hold time	t_{IHCKE}		MIN	0.2	0.2	$t_{CK(avg)}$	6
Command path disable delay	t_{CPDED}		MIN	2	2	$t_{CK(avg)}$	
Command Address Input Parameters							
Address and control input setup time	t_{ISCA}		MIN	270	200	ps	6
Address and control input hold time	t_{IHCA}		MIN	270	200	ps	6
CS_n input setup time	t_{ISCS}		MIN	337	253	ps	6
CS_n input hold time	t_{IHCS}		MIN	337	253	ps	6
Address and control input pulse width	t_{IPWCA}		MIN	0.315	0315	$t_{CK(avg)}$	6
CS_n input pulse width	t_{IPWCS}		MIN	0.67	0.67	$t_{CK(avg)}$	6
Mode Register Parameters							
Mode Register Write command period	t_{MRW}		MIN	5	5	$t_{CK(avg)}$	
Mode Register Read command period	t_{MRR}		MIN	2	2	$t_{CK(avg)}$	
Additional time after tXP has expired until MRR command may be issued	t_{MRRI}		MIN	t_{RCD}	t_{RCD}	ns	
Mode Register Set Command Delay	t_{MRD}		MIN	Max(14ns, 10tCK)	Max(14ns, 10tCK)	ns	
Refresh Rate Update	t_{RRU}		MIN	200	200	us	
Core Parameters							
Read Latency (no DBI)	RL	5	MIN	7	9	$t_{CK(avg)}$	
Read Latency (with DBI)	RL	6	MIN	8	10	$t_{CK(avg)}$	
Write Latency	WL	3	MIN	5	7	$t_{CK(avg)}$	
ACTIVATE to ACTIVATE command period (with single bank PRECHARGE)	t_{RCpb}		MIN	tRAS+tRPpb		ns	
ACTIVATE to ACTIVATE command period (with all banks PRECHARGE)	t_{RCab}		MIN	tRAS+tRPab		ns	
CKE min. pulse width during Self-Refresh (low pulse width during Self-Refresh)	t_{CKESR}	3	MIN	15		ns	
Self refresh exit to next valid command delay	t_{XSR}	2	MIN	tRFCab+10		ns	

Table 53 — AC Timings

Parameter	Symbol	min tCK	Min/ Max	Data Rate		Unit	Note
				800 MT/s	1066 MT/s		
Self refresh exit to next valid command delay with Self Refresh abort	t_{XSR_abort}		MIN	TBD		ns	
Exit power down to next valid command delay	t_{XP}	2	MIN	7.5		ns	
CAS to CAS delay BL = 4	t_{CCD4}	2	MIN	2		$t_{CK(avg)}$	
CAS to CAS delay BL = 8	t_{CCD8}	4	MIN	4		$t_{CK(avg)}$	
Internal Read to Precharge command delay	t_{RTP}	2	MIN	7.5		ns	
RAS to CAS Delay	t_{RCD}	3	MIN	18		ns	
Row Precharge Time (single bank)	t_{RPpb}	3	MIN	18		ns	
Row Precharge Time (all banks)	$t_{RPab\ 4\text{-}bank}$	3	MIN	18		ns	
Row Precharge Time (all banks)	$t_{RPab\ 8\text{-}bank}$	3	MIN	21		ns	
Row Active Time	t_{RAS}	3	MIN	42		ns	
			MAX	Min (9 × tREFI × REFM, 70.2)		us	
Write Recovery Time	t_{WR}	3	MIN	20		ns	
Internal Write to read Command Delay	t_{WTR}	2	MIN	10		ns	
Active bank A to Active Bank B	t_{RRD}	2	MIN	10		ns	
Four Bank Activate Window	t_{FAW}	8	MIN	60		ns	

NOTE 1 Includes DRAM process, voltage and temperature variation.
NOTE 2 t_{DQSCK_temp} max delay variation as a function of Temperature at a fixed voltage.
NOTE 3 t_{DQSCK_volt} max delay variation as a function of DC voltage variation for V_{DDQ} and V_{DD2} at a fixed temperature. It includes the AC noise impact for frequencies > 250 KHz at a fixed temperature on the die. The voltage variation is defined as the Max[abs{$t_{DQSCKmin}$@1-$tD_{QSCKmax}$@2},abs{$t_{DQSCKmax}$@1-$t_{DQSCKmin}$@2}].
NOTE 4 Values are guaranteed by design and characterization.
NOTE 5 1 × tCK indicates one cycle toggling of extendded Read postamble.
NOTE 6 Measured at V_{IH}/V_{IL}

13 IDD Specification Parameter and Test Condition

13.1 IDD Specifications

IDD values are for the entire operating voltage range, and all of them are for the entire standard temperature range. All specifications apply to the device on a per-die basis, with a single channel in the required state for the applicable measurement parameter, and all other channels in the Idle Power-Down Standby state.

Table 54 — IDD Specification Parameters

Parameter / Condition	Symbol	Power Supply	Notes
Operating one bank active-precharge current: tCK = tCKmin; tRC = tRCmin; CKE is HIGH; CS_n is HIGH between valid commands; CA bus inputs are switching; Data bus inputs are stable	IDD01	VDD1	
	IDD02	VDD2	
	IDD0in	VDDQ	
Idle power-down standby current: tCK = tCKmin; CKE is LOW; CS_n is HIGH; All banks are idle; CA bus inputs are switching; Data bus inputs are stable	IDD2P1	VDD1	
	IDD2P2	VDD2	
	IDD2P,in	VDDQ	
Idle power-down standby current with clock stop: CK = LOW; CKE is LOW; CS_n is HIGH; All banks are idle; CA bus inputs are stable; Data bus inputs are stable	IDD2PS1	VDD1	
	IDD2PS2	VDD2	
	IDD2PS,in	VDDQ	
Idle non-power-down standby current: tCK = tCKmin; CKE is HIGH; CS_n is HIGH; All banks are idle; CA bus inputs are switching; Data bus inputs are stable	IDD2N1	VDD1	
	IDD2N2	VDD2	
	IDD2N,in	VDDQ	
Idle non-power-down standby current with clock stop: CK = LOW; CKE is HIGH; CS_n is HIGH; All banks are idle; CA bus inputs are stable; Data bus inputs are stable	IDD2NS1	VDD1	
	IDD2NS2	VDD2	
	IDD2NS,in	VDDQ	
Active power-down standby current: tCK = tCKmin; CKE is LOW; CS_n is HIGH; One bank is active; CA bus inputs are switching; Data bus inputs are stable	IDD3P1	VDD1	
	IDD3P2	VDD2	
	IDD3P,in	VDDQ	
Active power-down standby current with clock stop: CK = LOW; CKE is LOW; CS_n is HIGH; One bank is active; CA bus inputs are stable; Data bus inputs are stable	IDD3PS1	VDD1	
	IDD3PS2	VDD2	
	IDD3PS,in	VDDQ	

Table 54 — IDD Specification Parameters

Active non-power-down standby current: tCK = tCKmin; CKE is HIGH; CS_n is HIGH; One bank is active; CA bus inputs are switching; Data bus inputs are stable	IDD3N1	VDD1	
	IDD3N2	VDD2	
	IDD3N,in	VDDQ	
Active non-power-down standby current with clock stop: CK = LOW; CKE is HIGH; CS_n is HIGH; One bank is active; CA bus inputs are stable; Data bus inputs are stable	IDD3NS1	VDD1	
	IDD3NS2	VDD2	
	IDD3NS,in	VDDQ	
Operating burst READ current: tCK = tCKmin; All bank active; BL = 4; RLmin; CA bus inputs are switching; 50% data change each burst transfer	IDD4R1	VDD1	
	IDD4R2	VDD2	
	IDD4RQ	VDDQ	5
Operating burst WRITE current: tCK = tCKmin; All bank active; BL = 4; WL = WLmin; CA bus inputs are switching; 50% data change each burst transfer	IDD4W1	VDD1	
	IDD4W2	VDD2	
	IDD4W,in	VDDQ	

All-bank REFRESH burst current:
tCK = tCKmin;
CKE is HIGH between valid commands;
tRC = tRFCmin;
Burst refresh;
CA bus inputs are switching;
Data bus inputs are stable
	IDD51	VDD1	

	IDD52	VDD2	
	IDD5IN	VDDQ	
All-bank REFRESH average current: tCK = tCKmin; CKE is HIGH between valid commands; tRC = tREFI × REFM; CA bus inputs are switching; Data bus inputs are stable	IDD5A1	VDD1	
	IDD5A2	VDD2	
	IDD5A,in	VDDQ	
Self refresh current (Standard Temperature Range): CK_t = LOW; CK_c = HIGH CKE is LOW; CA bus inputs are stable; Data bus inputs are stable Maximum 1x self refresh rate	IDD61	VDD1	4,6
	IDD62	VDD2	
	IDD6IN	VDDQ	

SPECIAL NOTE: NOTE 1, NOTE 2 and NOTE 3 apply for all values. CA bus is comprised of RAS_n, CAS_n, WE_n, and A[msb:lsb].

NOTE 1 Published IDD values are the maximum of the distribution of the arithmetic mean.
NOTE 2 IDD current specifications are tested after the device is properly initialized.
NOTE 3 All other channels are in idle power-down standby with clock-stopped state (IDD2PS)
NOTE 4 The 1x self refresh rate is the rate at which the device is refreshed internally during self refresh, before going into the elevated temperature range.
NOTE 5 Guaranteed by design with output reference load and 180ps max rise time.
NOTE 6 This is the general definition that applies to full-array SELF REFRESH.

13.2 IDD Measurement Conditions

The following definitions are used within the IDD measurement tables:

LOW: $V_{IN} \leq V_{IL}$(DC) MAX

HIGH: VIN $\geq V_{IH}$(DC) MIN

STABLE: Inputs are stable at a HIGH or LOW level

SWITCHING: See the tables below.

Table 55 — Definition of Switching for CA Input Signals

Switching for CA								
	CK_t (RISING)	CK_c (RISING)	CK_t (RISING)	CK_c (RISING)	CK_t (RISING)	CK_c (RISING)	CK_t (RISING)	CK_c (RISING)
Cycle	N		N+1		N+2		N+3	
CS_n	HIGH		HIGH		HIGH		HIGH	
CA0	HIGH	LOW	LOW	LOW	LOW	HIGH	HIGH	HIGH
CA1	HIGH	HIGH	HIGH	LOW	LOW	LOW	LOW	HIGH
CA2	HIGH	LOW	LOW	LOW	LOW	HIGH	HIGH	HIGH
CA3	HIGH	HIGH	HIGH	LOW	LOW	LOW	LOW	HIGH
CA4	HIGH	LOW	LOW	LOW	LOW	HIGH	HIGH	HIGH
CA5	HIGH	HIGH	HIGH	LOW	LOW	LOW	LOW	HIGH
CA6	HIGH	LOW	LOW	LOW	LOW	HIGH	HIGH	HIGH
CA7	HIGH	HIGH	HIGH	LOW	LOW	LOW	LOW	HIGH
CA8	HIGH	LOW	LOW	LOW	LOW	HIGH	HIGH	HIGH
CA9	HIGH	HIGH	HIGH	LOW	LOW	LOW	LOW	HIGH
CA10	HIGH	LOW	LOW	LOW	LOW	HIGH	HIGH	HIGH

NOTE 1 CS_n must always be driven HIGH.

NOTE 2 50% of CA bus is changing between HIGH and LOW once per clock for the CA bus.

NOTE 3 The above pattern (N, N+1, N+2, N+3...) is used continuously during IDD measurement for IDD values that require SWITCHING on the CA bus.

Table 56 — Definition of Switching for IDD4R

Loop	Clock	CKE	CS_n	Clock Cycle Number	Command	CA0 - CA2	CA3 - CA10	All DQs	CA7-CA9
0	CK_t(RISING)	HIGH	LOW	0	Read	HLH	HLHLVVVL	L	000
0	CK_c(RISING)	HIGH	LOW			LLL	LLLLLLLL	L	
0	CK_t(RISING)	HIGH	HIGH	1	NOP	LLL	LLLLLLLL	H	
0	CK_c(RISING)	HIGH	HIGH			HLH	HLHLVVVL	L	000
0	CK_t(RISING)	HIGH	LOW	2	Read	HLH	HLHLVVVL	H	000
0	CK_c(RISING)	HIGH	LOW			LLL	HHHHHHHL	H	
0	CK_t(RISING)	HIGH	HIGH	3	NOP	LLL	HHHHHHHL	H	
0	CK_c(RISING)	HIGH	HIGH			HLH	HLHLVVVL	L	000
1	Repeat Loop 0 with CA7-CA9 = 001								
2	Repeat Loop 0 with CA7-CA9 = 010								
3	Repeat Loop 0 with CA7-CA9 = 011								
4	Repeat Loop 0 with CA7-CA9 = 100								
5	Repeat Loop 0 with CA7-CA9 = 101								
6	Repeat Loop 0 with CA7-CA9 = 110								
7	Repeat Loop 0 with CA7-CA9 = 111								

NOTE 1 Data strobe (DQS) is changing between HIGH and LOW every clock cycle.

NOTE 2 The above pattern (N, N+1, ...) is used continuously during IDD measurement for IDD4R.

13.2 IDD Measurement Conditions (cont'd)

Table 57 — Definition of Switching for IDD4W

Loop	Clock	CKE	CS_n	Clock Cycle Number	Command	CA0 - CA2	CA3 - CA10	All DQs	CA7-CA9
0	CK_t(RISING)	HIGH	LOW	N	Write	HLL	LLHHVVVL	L	000
0	CK_c(RISING)	HIGH	LOW			LLL	LLLLLLLL	L	
0	CK_t(RISING)	HIGH	HIGH	N+1	NOP	LLL	LLLLLLLL	H	
0	CK_c(RISING)	HIGH	HIGH			HLH	LLHHVVVL	L	000
0	CK_t(RISING)	HIGH	LOW	N+2	Write	HLL	LLHHVVVL	H	000
0	CK_c(RISING)	HIGH	LOW			LLL	HHHHHHHL	H	
0	CK_t(RISING)	HIGH	HIGH	N+3	NOP	LLL	HHHHHHHL	H	
0	CK_c(RISING)	HIGH	HIGH			HLH	LLHHVVVL	L	000
1	Repeat Loop 0 with CA7-CA9 = 001								
2	Repeat Loop 0 with CA7-CA9 = 010								
3	Repeat Loop 0 with CA7-CA9 = 011								
4	Repeat Loop 0 with CA7-CA9 = 100								
5	Repeat Loop 0 with CA7-CA9 = 101								
6	Repeat Loop 0 with CA7-CA9 = 110								
7	Repeat Loop 0 with CA7-CA9 = 111								

NOTE 1 Data strobe (DQS) is changing between HIGH and LOW every clock cycle.
NOTE 2 Data masking (DMI) must always be driven LOW.
NOTE 3 The above pattern (N, N+1...) is used continuously during IDD measurement for IDD4W.

Standard Improvement Form

JEDEC _____

The purpose of this form is to provide the Technical Committees of JEDEC with input from the industry regarding usage of the subject standard. Individuals or companies are invited to submit comments to JEDEC. All comments will be collected and dispersed to the appropriate committee(s).

If you can provide input, please complete this form and return to:

JEDEC
Attn: Publications Department
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

Fax: 703.907.7583

1. I recommend changes to the following:

☐ Requirement, clause number _____

☐ Test method number _____ Clause number _____

The referenced clause number has proven to be:

☐ Unclear ☐ Too Rigid ☐ In Error

☐ Other _____

2. Recommendations for correction:

_____

_____

_____

_____

3. Other suggestions for document improvement:

_____

_____

_____

_____

Submitted by

Name: _____ Phone: _____

Company: _____ E-mail: _____

Address: _____

City/State/Zip: _____ Date: _____