User's Manual

ELPIDA

HOW TO USE SDRAM

Document No. E0123N10 (Ver.1.0)
(Previous No. M13132EJ2V0UM00)
Date Published May 2001 CP(K)

Elpida Memory, Inc. is a joint venture DRAM company of NEC Corporation and Hitachi, Ltd.

SUMMARY OF CONTENTS

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

Note:

Strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it once, when it has occurred. Environmental control must be adequate. When it is dry, humidifier should be used. It is recommended to avoid using insulators that easily build static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work bench and floor should be grounded. The operator should be grounded using wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with semiconductor devices on it.

② HANDLING OF UNUSED INPUT PINS FOR CMOS

Note:

No connection for CMOS device inputs can be cause of malfunction. If no connection is provided to the input pins, it is possible that an internal input level may be generated due to noise, etc., hence causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using a pull-up or pull-down circuitry. Each unused pin should be connected to V_{DD} or GND with a resistor, if it is considered to have a possibility of being an output pin. All handling related to the unused pins must be judged device by device and related specifications governing the devices.

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

Note:

Power-on does not necessarily define initial status of MOS device. Production process of MOS does not define the initial operation status of the device. Immediately after the power source is turned ON, the devices with reset function have not yet been initialized. Hence, power-on does not guarantee out-pin levels, I/O settings or contents of registers. Device is not initialized until the reset signal is received. Reset operation must be executed immediately after power-on for devices having reset function.

VirtualChannel is trademark of NEC Corporation.

Rambus, RDRAM and the Rambus logo are registered trademarks of Rambus Inc.

Direct Rambus, Direct RDRAM, RIMM and SO-RIMM are trademarks of Rambus Inc.

The information in this document is subject to change without notice. Before using this document, confirm that this is the latest version.

M02 01.2

INTRODUCTION

Target Readers

This manual is intended for users who wish to understand the function of SDRAM and design application systems for them.

Purpose

This manual is intended to give users understanding of the basic function of SDRAM and how to use them.

How to Use This Manual

It is assumed that readers of this manual should have general knowledge in the fields of electrical engineering, logic circuits, and memory products. For details about the functions of individual products, refer to the corresponding data sheet. Since the operation examples that appear in this user's manual are strictly illustrative examples, numerical values that appear are not guaranteed values. Use them only as reference values.

Conventions

Note: Footnote for items marked with **Note** in the text
Caution: Information requiring particular attention
Remark: Supplementary information
Numeric notation: Decimal ... xxxx

Related Documents

The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

Document Name	Document Number
Synchronous DRAM User's Manual	E0124N
μPD4564441, 4564841, 4564163 DATA SHEET	M12621
μPD4564323 DATA SHEET	M12976
μPD45128441, 45128841, 45128163 DATA SHEET	E0031N

Organization This manual consists of the following chapters.

EXPLANATION OF THE FEATURES OF THE 128M SDRAM
CHAPTER 1 PRODUCT OUTLINE

EXPLANATION OF THE SYNCHRONOUS OPERATIONS AND SPECIFIC FUNCTIONS
CHAPTER 2 FEATURES OF PRODUCTS

CHAPTER 3 OPERATION AFTER POWER APPLICATION

CHAPTER 4 BASIC SETTING (MODE REGISTER SETTING)

CHAPTER 5 STATUS TRANSITIONS

CHAPTER 6 COMMAND OPERATIONS

CHAPTER 7 BASIC OPERATION MODE

CHAPTER 8 DQM CONTROL OPERATION

CHAPTER 9 CKE CONTROL OPERATION

CHAPTER 10 BURST OPERATION

CHAPTER 11 MULTIBANK OPERATION

CHAPTER 12 CALCULATION OF CURRENT CONSUMPTION

CONTENTS

LIST OF FIGURES (1/3)

LIST OF FIGURES (2/3)

LIST OF FIGURES (3/3)

LIST OF TABLES

CHAPTER 1 PRODUCT OUTLINE

This chapter provides an outline of SDRAM products, taking the μPD45128163 128M SDRAM (512 K words × 16 bits × 4 banks) as examples. Unless otherwise specified, these models are treated as the representative products in this User's Manual.

1.1 Pin Configuration

This section shows the pin configuration and pin names of the 64M/128M SDRAM.

Figure 1-1. Pin configuration of 64M/128M SDRAM

V_{CC}	V_{CC}	V_{CC}	1	54	V_{SS}	V_{SS}	V_{SS}
DQ0	DQ0	NC	2	53	NC	DQ7	DQ15
$V_{CC}Q$	$V_{CC}Q$	$V_{CC}Q$	3	52	$V_{SS}Q$	$V_{SS}Q$	$V_{SS}Q$
DQ1	NC	NC	4	51	NC	NC	DQ14
DQ2	DQ1	DQ0	5	50	DQ3	DQ6	DQ13
$V_{SS}Q$	$V_{SS}Q$	$V_{SS}Q$	6	49	$V_{CC}Q$	$V_{CC}Q$	$V_{CC}Q$
DQ3	NC	NC	7	48	NC	NC	DQ12
DQ4	DQ2	NC	8	47	NC	DQ5	DQ11
$V_{CC}Q$	$V_{CC}Q$	$V_{CC}Q$	9	46	$V_{SS}Q$	$V_{SS}Q$	$V_{SS}Q$
DQ5	NC	NC	10	45	NC	NC	DQ10
DQ6	DQ3	DQ1	11	44	DQ2	DQ4	DQ9
$V_{SS}Q$	$V_{SS}Q$	$V_{SS}Q$	12	43	$V_{CC}Q$	$V_{CC}Q$	$V_{CC}Q$
DQ7	NC	NC	13	42	NC	NC	DQ8
V_{CC}	V_{CC}	V_{CC}	14	41	V_{SS}	V_{SS}	V_{SS}
LDQM	NC	NC	15	40	NC	NC	NC
/WE	/WE	/WE	16	39	DQM	DQM	UDQM
/CAS	/CAS	/CAS	17	38	CLK	CLK	CLK
/RAS	/RAS	/RAS	18	37	CKE	CKE	CKE
/CS	/CS	/CS	19	36	NC	NC	NC
A13	A13	A13	20	35	A11	A11	A11
A12	A12	A12	21	34	A9	A9	A9
A10	A10	A10	22	33	A8	A8	A8
A0	A0	A0	23	32	A7	A7	A7
A1	A1	A1	24	31	A6	A6	A6
A2	A2	A2	25	30	A5	A5	A5
A3	A3	A3	26	29	A4	A4	A4
V_{CC}	V_{CC}	V_{CC}	27	28	V_{SS}	V_{SS}	V_C

x4 (64M / 128M)

x8 (64M / 128M)

x16 (64M / 128M)

A0 - A13	:	Address inputs	UDQM	:	Upper DQ mask enable
A12 (BA1),	:	Bank address	LDQM	:	Lower DQ mask enable
A13 (BA0)			CKE	:	Clock enable
DQ0 - DQ15	:	Data inputs/outputs	CLK	:	System clock input
/CS	:	Chip select	V_{CC}	:	Supply voltage
/RAS	:	Row address strobe	V_{SS}	:	Ground
/CAS	:	Column address strobe	$V_{CC}Q$	:	Supply voltage for DQ
/WE	:	Write enable	$V_{SS}Q$	:	Ground for DQ
			NC	:	No connection

1.2 Pin Functions

1.2.1 CLK input

An external clock is input to the CLK pin.

All input signals and data input/output signals are synchronized with the rising edge of CLK.

1.2.2 CKE input

CKE determines whether CLK is valid or not. If the CKE signal is high at the rising edge of a given CLK, the rising edge of the next CLK is valid. Otherwise, the rising edge of the next CLK is invalid.

If the rising edge of CLK is invalid, the internal clock does not operate, and the product temporarily stops.

<Burst mode>

The internal burst clock temporarily stops if the CKE signal is made low in the burst mode.

<Self refresh mode>

Self refreshing is executed if the CKE signal is made low in the self refresh mode. In this mode, the CKE signal must be kept low.

<Modes other than burst and self refresh modes>

In a mode other than the burst and self refresh modes, the power down mode is set if the CKE signal is made low. In this mode, the CKE signal must be kept low.

1.2.3 /CS input

Low level: Starts a command input cycle.

High level: Command is ignored but the operation continues.

1.2.4 /RAS, /CAS, /WE input

/RAS, /CAS, and /WE have the same name as the signals of conventional DRAM but they differ in their function. For details, refer to the list of commands.

1.2.5 V_{CC}, V_{SS}

Power supply pins. V_{CC} and V_{SS} are power supply pins for the internal circuit.

1.2.6 $V_{CC}Q$, $V_{SS}Q$

Power supply pins. $V_{CC}Q$ and $V_{SS}Q$ are power supply pins for the output buffer.

1.2.7 Address (A0 through Ax) input

<Row address>
Determined by A0 through Ax input when an active command is input.

<Column address>
Determined by A0 through Ax input when a read or write command is input.

<Bank address (BA)>
The bank to be selected differs depending on the input level of BA when a command is input.

<Precharge mode select address (AP)>
The function of this pin differs depending on the input level of AP when a precharge command is input or when a read command (or write command) is input.

When precharge command is input

AP	Function
High level	Precharging all the banks is started (all bank precharge).
Low level	Precharging only the bank selected by a bank address is started.

When read/write command is input

AP	Function
High level	Precharge is automatically started after burst access (auto precharge).
Low level	Precharge command must be input to start precharge.

Table 1-1. Address Pins

Part Number	Address Pins	Row Address	Column Address	BA	AP
SDRAM					
μPD45128441	A0-A13	A0-A11	A0-A9,A11	A12, A13	A10
μPD45128841	A0-A13	A0-A11	A0-A9	A12, A13	A10
μPD45128163	A0-A13	A0-A11	A0-A8	A12, A13	A10
μPD4564441	A0-A13	A0-A11	A0-A9	A12, A13	A10
μPD4564841	A0-A13	A0-A11	A0-A8	A12, A13	A10
μPD4564163	A0-A13	A0-A11	A0-A7	A12, A13	A10

1.2.8 DQM0 through DQM3 input

DQM is used to control the I/O buffer.

<Read mode (output)>
DQM is used to control the output buffer and is the same as the conventional /OE pin in function.
High level: Output buffer is off.
Low level : Output buffer is on.
DQM latency for read is 2 clocks.

<Write mode (input)>
DQM is used to control the word mask.
High level: Input data is not written to a memory cell.
Low level : Input data is written to a memory cell.
DQM latency for write is zero.

Table 1-2. DQM Pin

Bit Organization	Pin Name	Control Pin	Part Number
×4 bits	DQM	DQ0-DQ3	μPD45128441
×8 bits	DQM	DQ0-DQ7	μPD45128841
×16 bits	LDQM	DQ0-DQ7	μPD45128163
	UDQM	DQ8-DQ15	
×32 bits	DQM0	DQ0-DQ7	μPD4564323
	DQM1	DQ8-DQ15	
	DQM2	DQ16-DQ25	
	DQM3	DQ26-DQ31	

Remark DQM latency is the number of clocks necessary for controlling the I/O buffer after DQM has been made high.

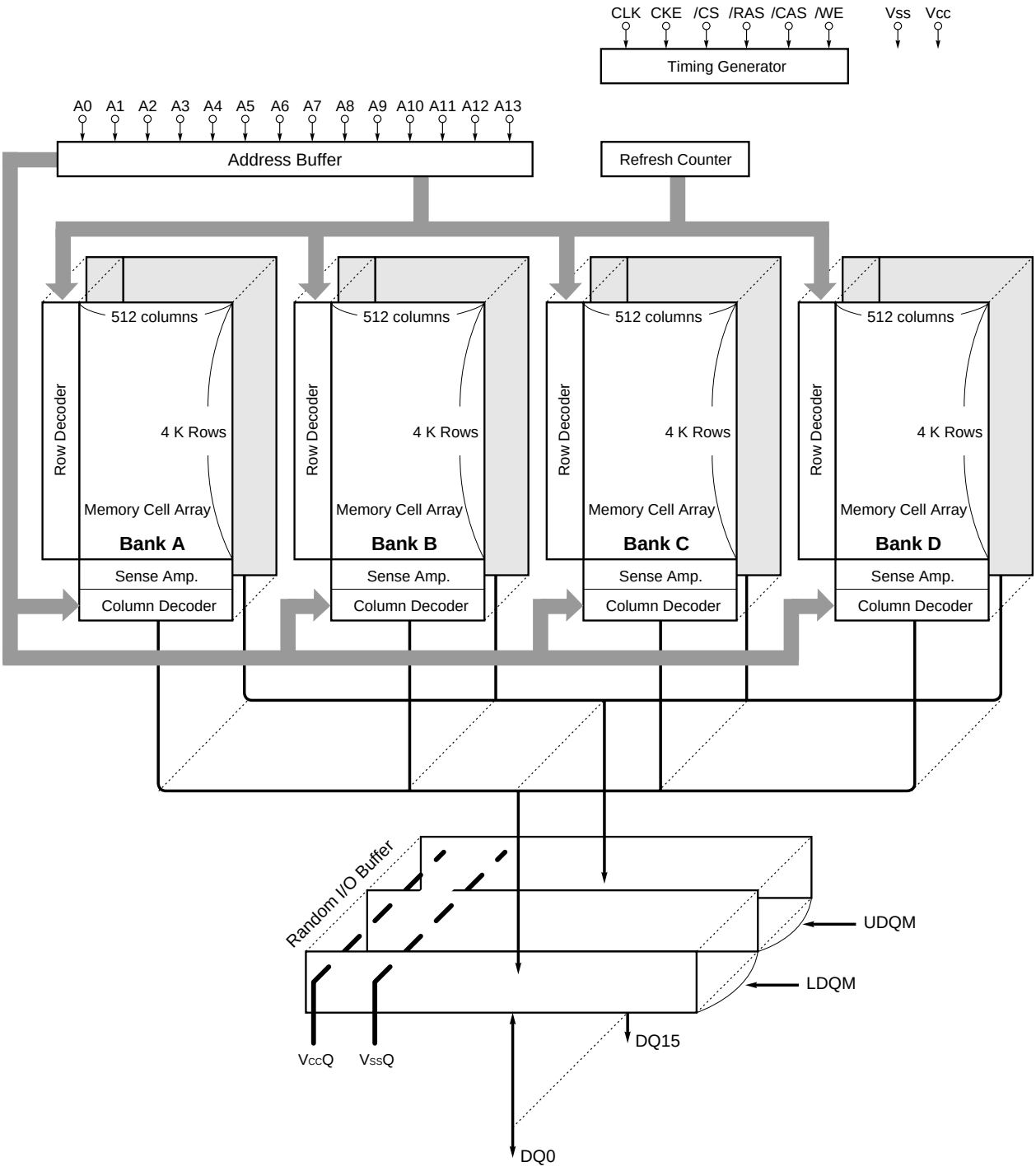
1.2.9 DQ0 through DQx I/O

The function of the DQ pin is the same as that of the I/O pin of conventional DRAM.

1.3 Block Diagram

1.3.1 Block diagram of 128M SDRAM (μPD45128163)

Figure 1-2. Block diagram of 128M SDRAM (μPD45128163)



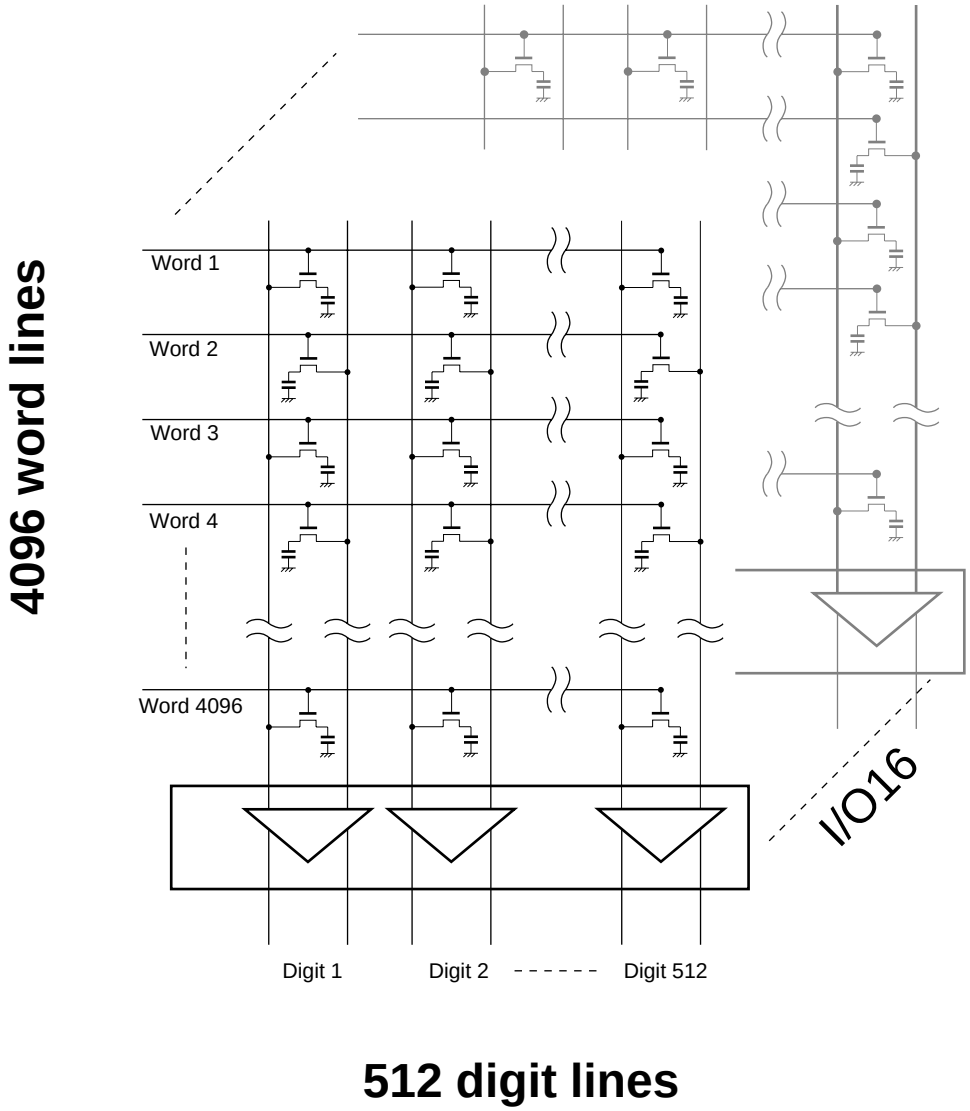
1.4 Description of Block Diagram

1.4.1 Memory cell array of 128M SDRAM (μPD45128163)

The circuit of a memory cell is configured of one transistor and one capacitor in the same manner as a general-purpose DRAM.

The μPD45128163 (×16-bit organization) has a total capacity of 128M bits and consists of 4096 word lines, 512 digit lines, 16 I/O lines, and four banks.

Figure 1-3. Memory Cell Array (128M SDRAM)



1.4.2 Address (Row, Column) decoders

These products employ an address multiplex method. To decode a certain address, a bank select signal and a row address are loaded at the same time as an active command, and the corresponding word line is selected. Next, a bank select signal and a column address are loaded at the same time as a read command or write command, the corresponding digit line is selected, and an address is decoded.

1.4.3 I/O buffer

Buffer for data input/output.

1.4.4 Refresh counter

This counter automatically counts row addresses in the memory.

CHAPTER 2 FEATURES OF PRODUCTS

NEC's SDRAM can access successive data at high speeds in synchronization with a system clock of 133 MHz at a voltage of as low as 3.3 V.

This chapter explains the following basic features of the SDRAM.

(1) Clock synchronization operation
(2) Control by command
(3) Plural bank configuration
(4) Burst transfer
(5) Comparison with general-purpose DRAM

2.1 Clock Synchronization Operation

The SDRAM latch each control signal at the rising edge of a basic input clock (CLK) and input/output data in synchronization with the input clock (CLK). Therefore, designing the timing is easy when the SDRAM operates at high speed.

Examples of timing of the basic input clock, control signals (commands), and input/output data (DQ) are shown below.

Figure 2-1. Read Cycle Timing

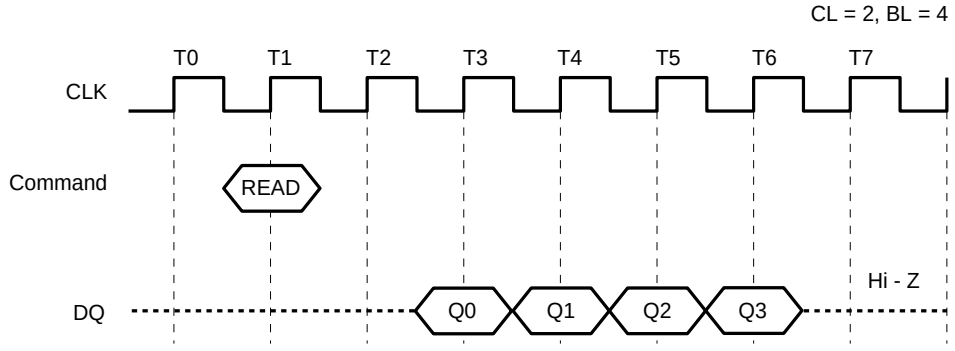
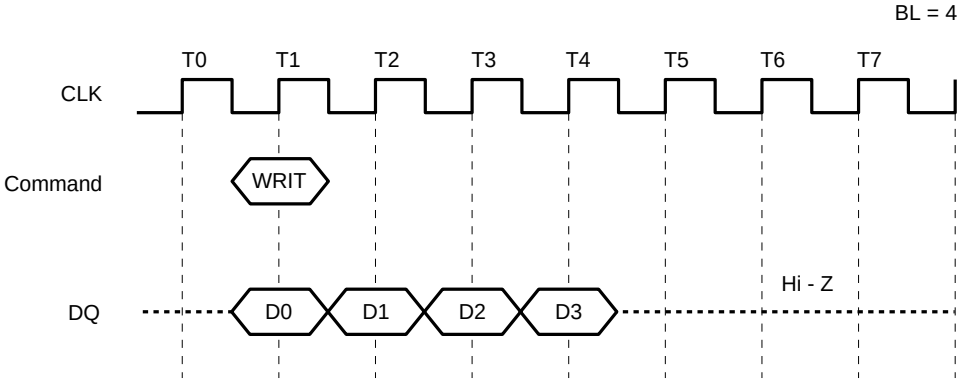


Figure 2-2. Write Cycle Timing



Remark CL: /CAS Latency
BL: Burst Length

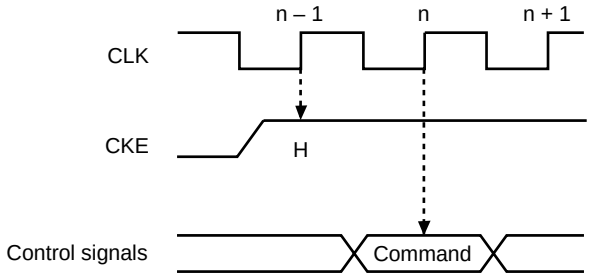
2.2 Control by Command

With SDRAM, combinations of logic levels of control signals are called commands. Typical commands include active command, read command, write command, and precharge command. Although control signals are combined at logic level when controlling general-purpose DRAM, the concept of commands is not used. The commands of the 128M SDRAM is listed on the following page. For examples of command operations, refer to Chapter 7.

2.2.1 Command input timing

All the commands are latched in synchronization with the rising edge of CLK. To activate CLK, signal CKE is used. When inputting a command, a high level must be input to CKE at the timing of CLK “n – 1” where “n” indicates the rising of CLK.

Figure 2-3. Command Input Timing



2.2.2 SDRAM command list

The commands of the 128M SDRAM are listed below.

Table 2-1. 128M SDRAM (×4/×8/×16-bit organization) Command List

No	Command	Symbol	CKE		/CS	/RAS	/CAS	/WE	DQM	Address		
			n – 1	n						BA0, BA1	A10	A9-A0 A11
1	Mode register set	MRS	H	×	L	L	L	L	×	L	L	V
2	CBR (auto) refresh	REF	H	H	L	L	L	H	×	×	×	×
3	Self refresh start	SELF	H	L	L	L	L	H	×	×	×	×
4	Self refresh exit	–	L	H	L	H	H	H	×	×	×	×
					H	×	×	×	×	×	×	×
5	Precharge select bank	PRE	H	×	L	L	H	L	×	V	L	×
6	Precharge all banks	PALL	H	×	L	L	H	L	×	×	H	×
7	Bank active	ACT	H	×	L	L	H	H	×	V	V	V
8	Write	WRIT	H	×	L	H	L	L	×	V	L	V
9	Write (with auto precharge)	WRITA	H	×	L	H	L	L	×	V	H	V
10	Read	READ	H	×	L	H	L	H	×	V	L	V
11	Read (with auto precharge)	READA	H	×	L	H	L	H	×	V	H	V
12	Burst stop	BST	H	×	L	H	H	L	×	×	×	×
13	No operation	NOP	H	×	L	H	H	H	×	×	×	×
14	Device deselect	DESL	H	×	H	×	×	×	×	×	×	×
15	Data input/output enable	-	H	×	×	×	×	×	L	×	×	×
16	Data mask	-	H	×	×	×	×	×	H	×	×	×

Remark H: High level, L: Low level, ×: High or low level (Don't care), V: Valid data

2.3 Plural Bank Configuration

The SDRAM divide the internal circuits of the chip (address decoder, memory cell arrays, and sense amplifier) into plural banks. Each bank can be controlled independently. With this configuration and by making the best use of the interleave operation of each bank, another bank can be accessed even while one bank is precharged.

2.3.1 Four-bank configuration

A four-bank model has four banks: A, B, C, and D. These banks are selected by a bank address (BA0 or BA1). The four-bank model is outlined below through comparison with general-purpose DRAM.

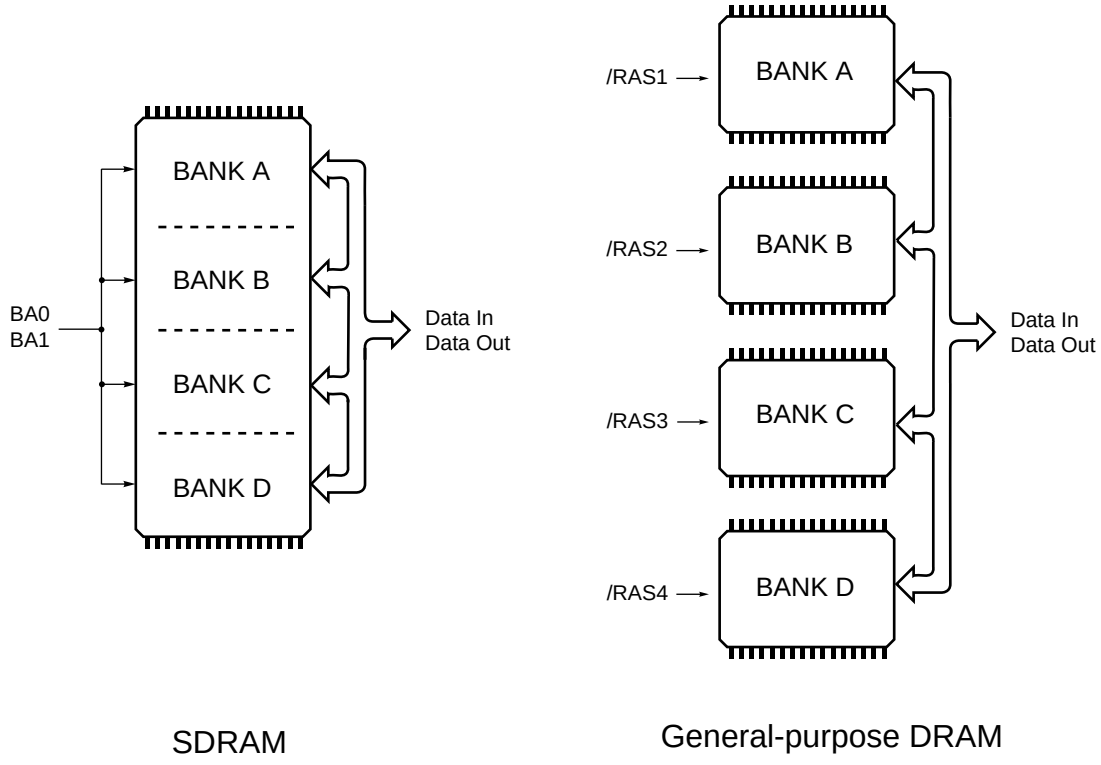
General-purpose DRAM

To use four banks, four devices are necessary. These banks are selected by the /RAS signal.

SDRAM

Because four banks are provided to a device, four banks can be configured with one device.

Figure 2-4. Four-Bank Configuration



2.4 Burst Transfer

Because the SDRAM internally perform pipeline processing, they can successively input/output a fixed number of memory data in synchronization with an external clock.

Pipeline processing divides the operations, including column address input and data input/output, into several blocks and executes these blocks in parallel, in order to enhance the data transfer capability.

Figure 2-6 illustrates the concept of three-stage pipeline architecture, comparing it with that of the general-purpose DRAM.

(1) General-purpose DRAM

The next operation cannot be started until a series of operations, from address input to data output, has been completed.

(2) SDRAM

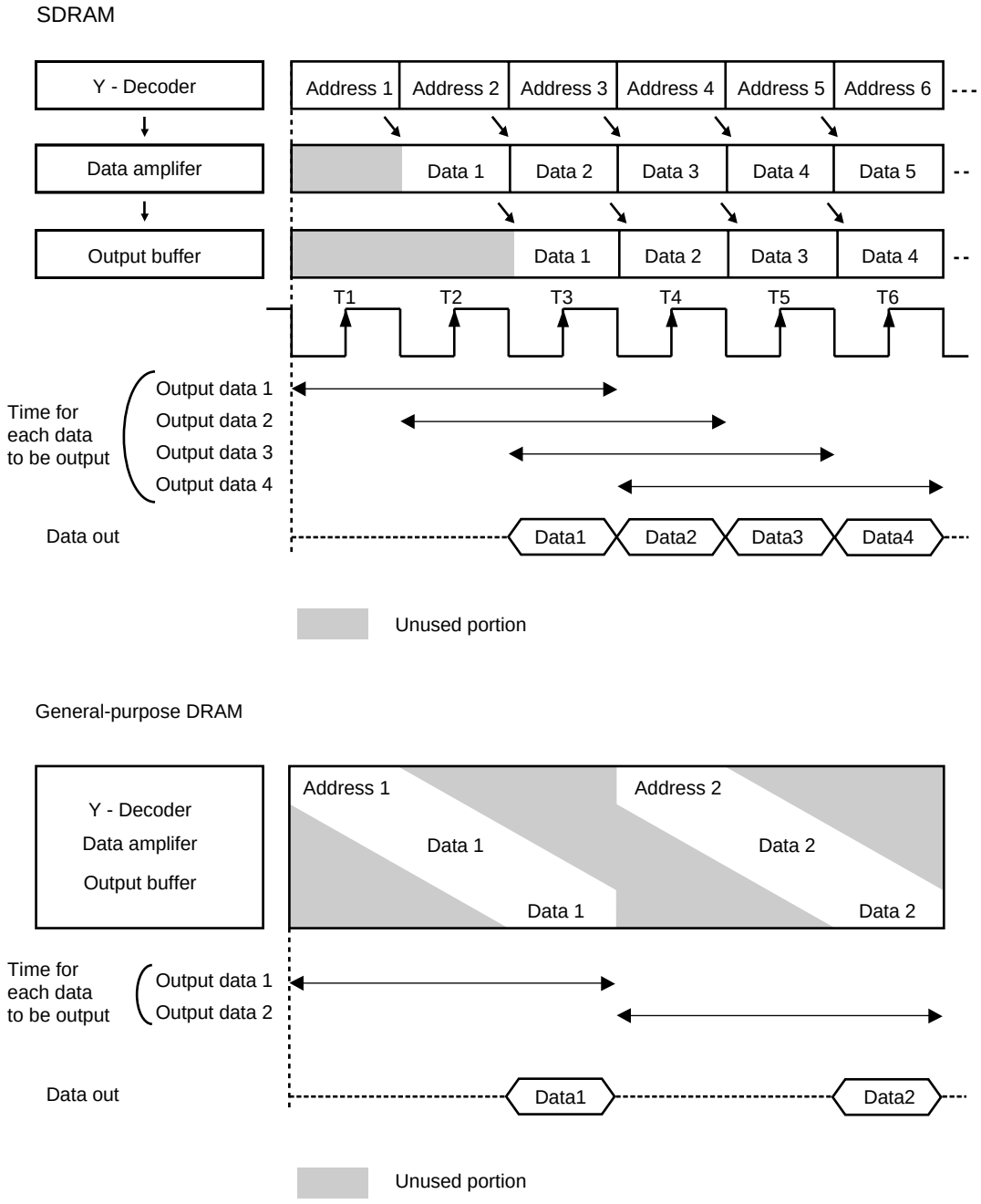
A column operation is divided into three operation blocks. When one operation block has been completed, the operation can proceed to the next operation block. When a column address is input, the internal column address counter automatically increments the internal column address in synchronization with the clock. The number by which the column address is to be incremented is determined by the burst length. This internal structure enables reading or writing of data of successive addresses.

(3) Comparison between general-purpose DRAM and SDRAM

The time for the first data to be output is the same for general-purpose DRAM and SDRAM.

When plural data are successively input/output, however, the SDRAM that perform parallel processing as described in (2) can transfer data at high speeds.

Figure 2-5. Pipeline Architecture



2.5 Comparison with General-Purpose DRAM

2.5.1 Increasing speed of DRAM

The figure below compares the access times of the SDRAM and an EDO DRAM.

Although there is not much difference in /RAS access time (random access time), the burst cycle time of SDRAM is much shorter than the burst cycle time of EDO DRAM.

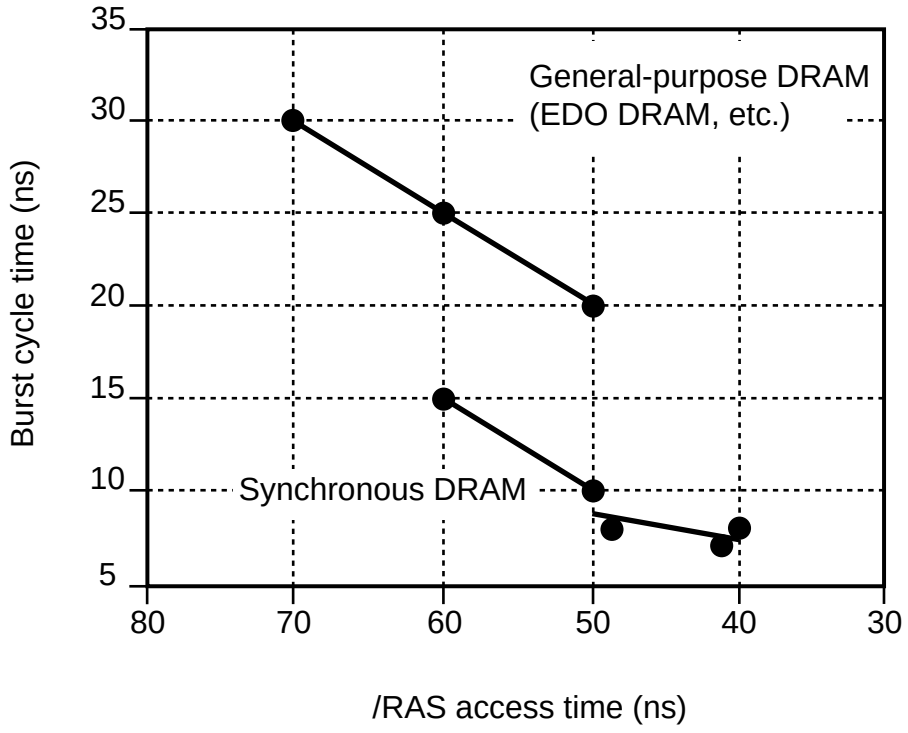
There is not much difference in the /RAS access time between SDRAM and EDO DRAM because their internal basic configurations are almost the same.

However, SDRAM can shorten the burst cycle time, as compared with general-purpose DRAM, by employing techniques different from those of general-purpose DRAM, such as pipelining.

Generally, EDO DRAM is synchronized with a 66-MHz (15-ns) clock. In contrast, SDRAM is planned to be synchronized with a 133-MHz (7.5-ns) clock. Synchronization with a clock of more than 200 MHz is also under study.

As the frequency of the clock system increases in the future, it is expected that the performance of systems can be improved by employing SDRAM instead of EDO DRAM when a system with a memory clock of 75 MHz or more is designed.

Figure 2-6. Increasing Speed of DRAM



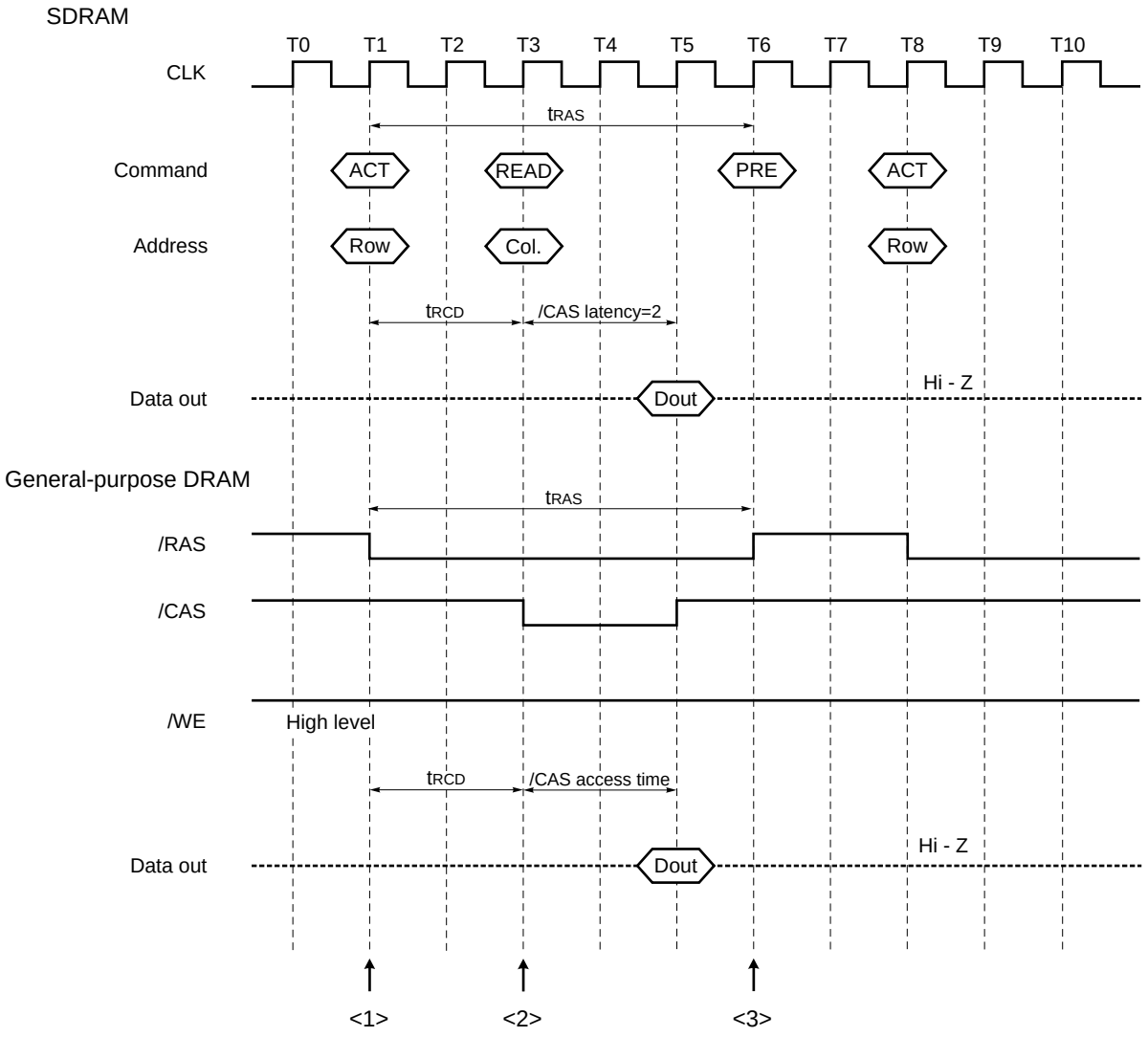
Remark The burst cycle time indicates the /CAS cycle time of DRAM and the clock cycle time of SDRAM.

2.5.2 Basic control method and access time

The actual control method is explained, taking the read cycle as an example.

(1) Basic control method

Figure 2-7. Read Cycle of SDRAM and General-Purpose DRAM



Caution General-purpose DRAM is asynchronous.

Timing <1>
SDRAM: Input of active command (ACT)
General-purpose DRAM: Status in which the /RAS signal goes low when the /CAS signal is high

Timing <2>
SDRAM: Input of read command (READ)
General-purpose DRAM: Status in which the /CAS signal goes low when the /RAS signal is low and when the /WE signal is high

SDRAM: /CAS latency
(Number of clocks since a column address signal has been latched until the valid data is output)
General-purpose DRAM: /CAS access time

SDRAM: Input of burst length (number of words of data successively output)
General-purpose DRAM: Number of page mode cycles

Timing <3>
SDRAM: Input of precharge command (PRE)
General-purpose DRAM: Status in which the /RAS and /CAS signals go high

(2) Access time

The following figure indicates the bust read cycle where burst length = 4. The access time of this SDRAM is compared with the access time of an EDO DRAM with a /RAS access time of 60 ns, assuming that the clock rate of the SDRAM is 66 MHz.

Figure 2-8. Burst Read Cycle

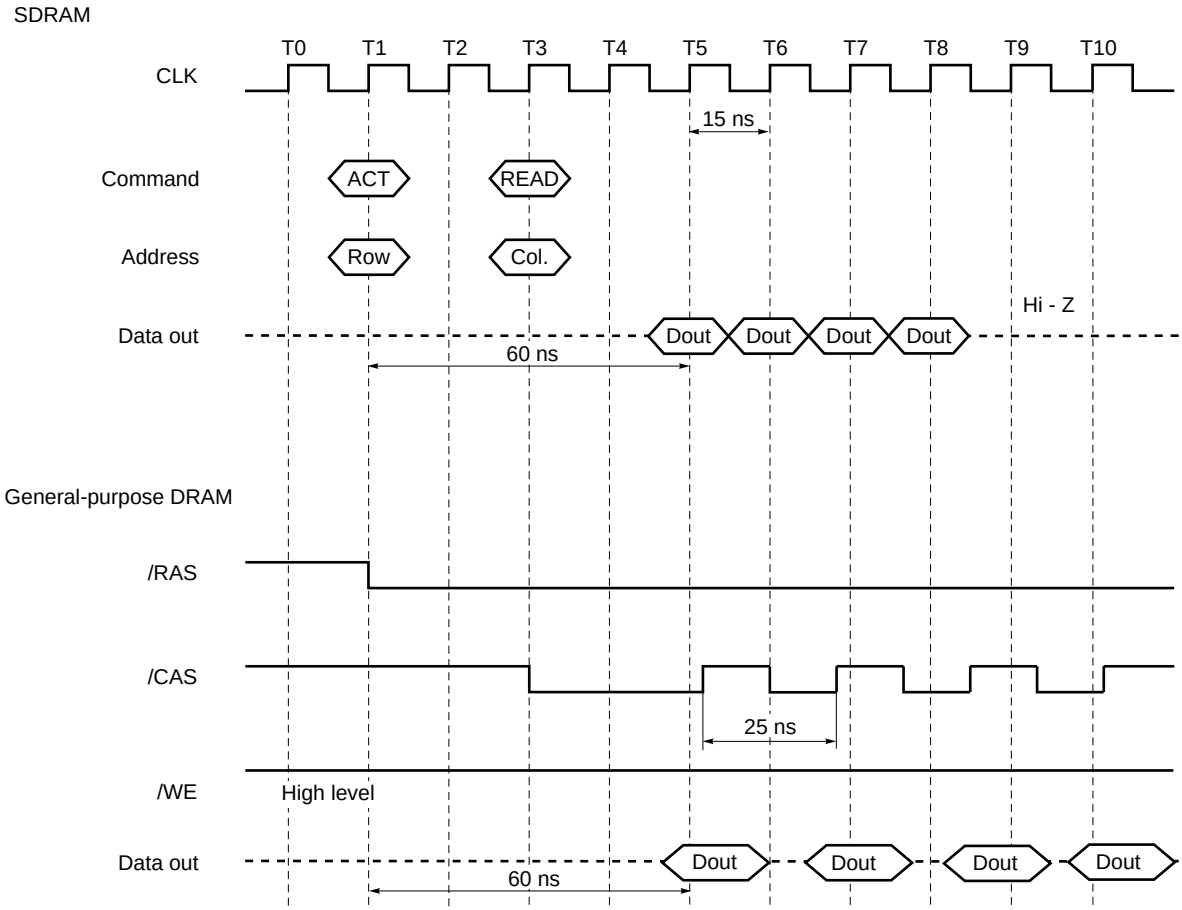


Table 2-2. Comparison of Access Time of SDRAM and EDO DRAM

Access Time	SDRAM	EDO DRAM	Comparison
First access time	60 ns	60 ns	Same
Second access time	75 ns	85 ns	SDRAM is shorter by 10 ns.
Third access time	90 ns	110 ns	SDRAM is shorter by 20 ns.
Fourth access time	105 ns	135 ns	SDRAM is shorter by 30 ns.

The first access times of SDRAM and EDO DRAM are the same. As the burst length increases, however, the transfer speed of SDRAM goes up.

The following table compares the access times of various SDRAMs and EDO DRAMs.

Table 2-3. Access Times of SDRAMs and EDO DRAMs

	Grade	First Access	Second Access	Third Access	Fourth Access
SDRAM	133 MHz (7.5 ns)	45 ns	52.5 ns	59 ns	66.5 ns
	125 MHz (8 ns)	40 ns	48 ns	56 ns	64 ns
	100 MHz (10 ns)	50 ns	60 ns	70 ns	80 ns
	66 MHz (15 ns)	60 ns	75 ns	90 ns	105 ns
EDO DRAM	–60	60 ns	85 ns	110 ns	135 ns
	–50	50 ns	70 ns	90 ns	110 ns

The first access times of SDRAM: 100 MHz (10 ns) and EDO DRAM: –50 are the same. However, the difference between SDRAM and EDO DRAM is evident at the fourth access. Although the above table shows data of up to burst length = 4, SDRAM is superior to EDO DRAM in data transfer capability as the burst length increases.

CHAPTER 3 OPERATION AFTER POWER APPLICATION

This chapter explains initialization after power application.

3.1 Initialization after Power Application

The logical status of the internal circuit of SDRAM is undefined immediately after power application. To ensure correct operation, SDRAM must always be initialized. Unless initialization is correctly executed, the device may not operate correctly.

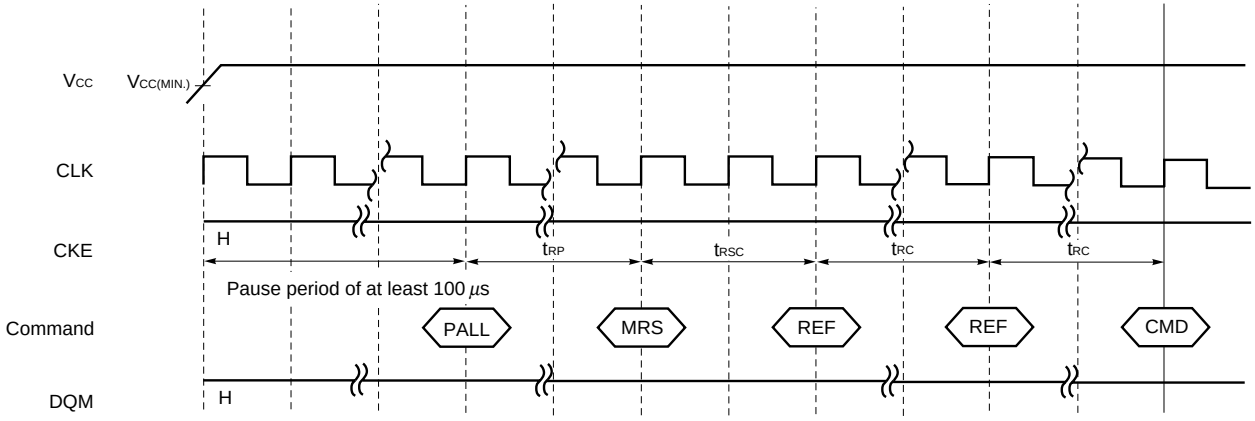
3.2 Initializing

1. Supply power and clock. At this time, make sure that CKE = H, DQM = H, and maintain the other input pins in the NOP or DESL status.
2. After the power and clock have been stabilized, make sure that CKE = H, DQM = H, and maintain the other input pins in the NOP or DESL status for at least 100 μs.
3. Precharge all the banks.
4. Execute the mode register set command to initialize the mode register.
5. Execute the auto refresh command at least twice as a dummy cycle.

Cautions 1. Steps 4 and 5 above are interchangeable.

2. To ensure that the output goes into a high-impedance state after the pause period of at least 100 μs in step 2 above, keep CKE and DQM high until the precharge command is input.

Figure 3-1. Initializing SDRAM



CHAPTER 4 BASIC SETTING (MODE REGISTER SETTING)

4.1 Mode Register Setting

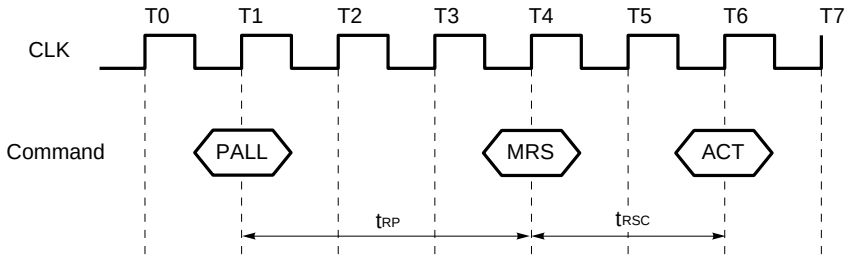
The mode register sets the operation modes of SDRAM, such as the latency mode, wrap type, and burst length. At this time, addresses A0 through Ax are used as input data. Once the mode register has been set, it holds the set data until it is reset or the power is turned off.

4.1.1 Setting

1 Execute the precharge command to all the banks.
→ Set all the banks in the idle status.

2. Execute the mode register set command.

Figure 4-1. Mode Register Setting Cycle



4.1.2 Set parameters

The mode register has the following four functions:

(1) A0 through A2 : Sets a burst length.
(2) A3 : Sets a wrap type.
(3) A4 through A6 : Sets a /CAS latency.
(4) A7 through Ax : Option

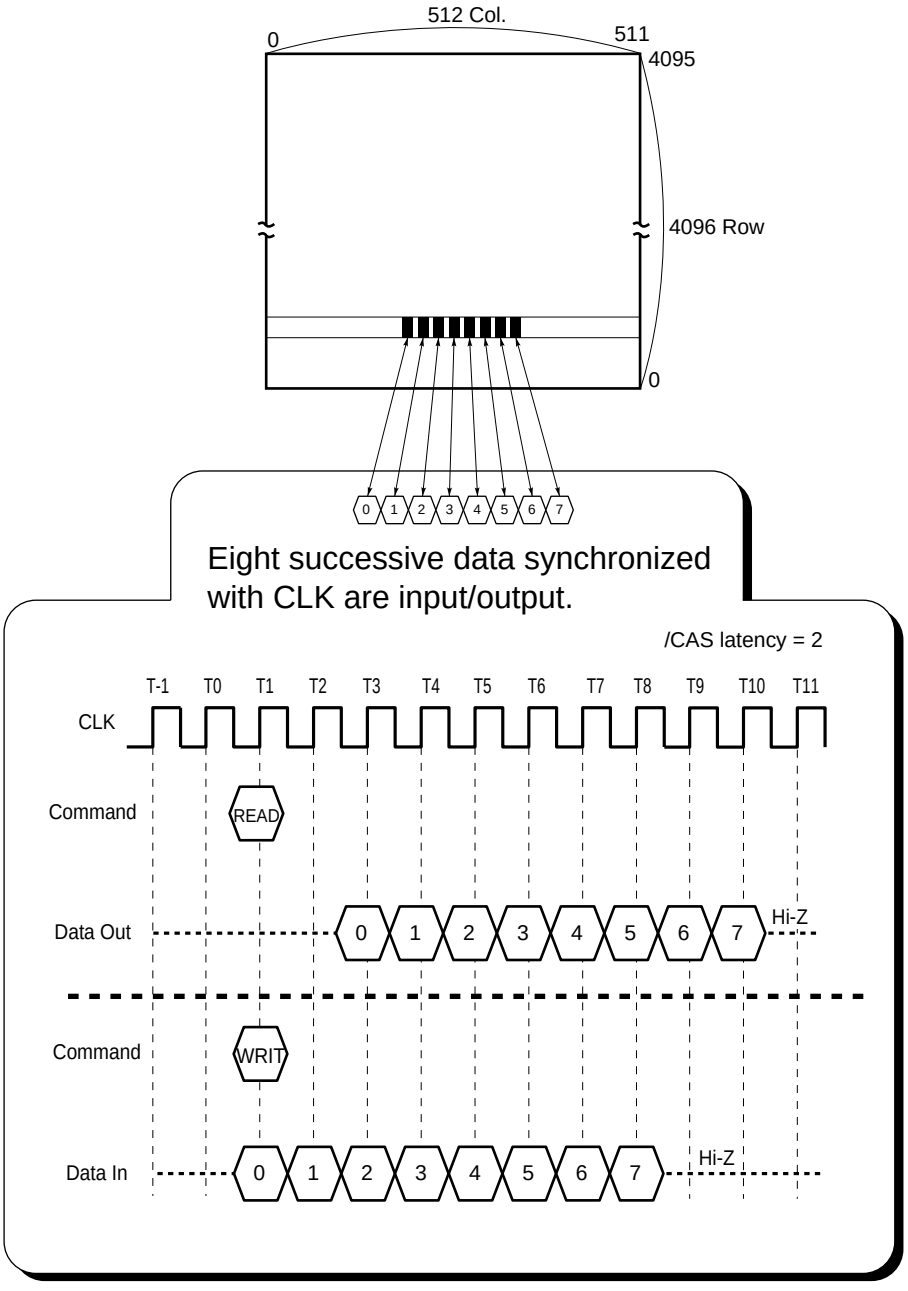
(1) Setting of burst length

The burst length is the number of data that can be successively input or output. The burst length may be 1, 2, 4, 8, or full page.

Example 1) Burst length of 8

Data of eight columns can be successively input or output by inputting a command once.
When the read burst has been completed, the data bus goes into a high-impedance state.

Figure 4-2. Read/Write Cycle with Burst Length of 8

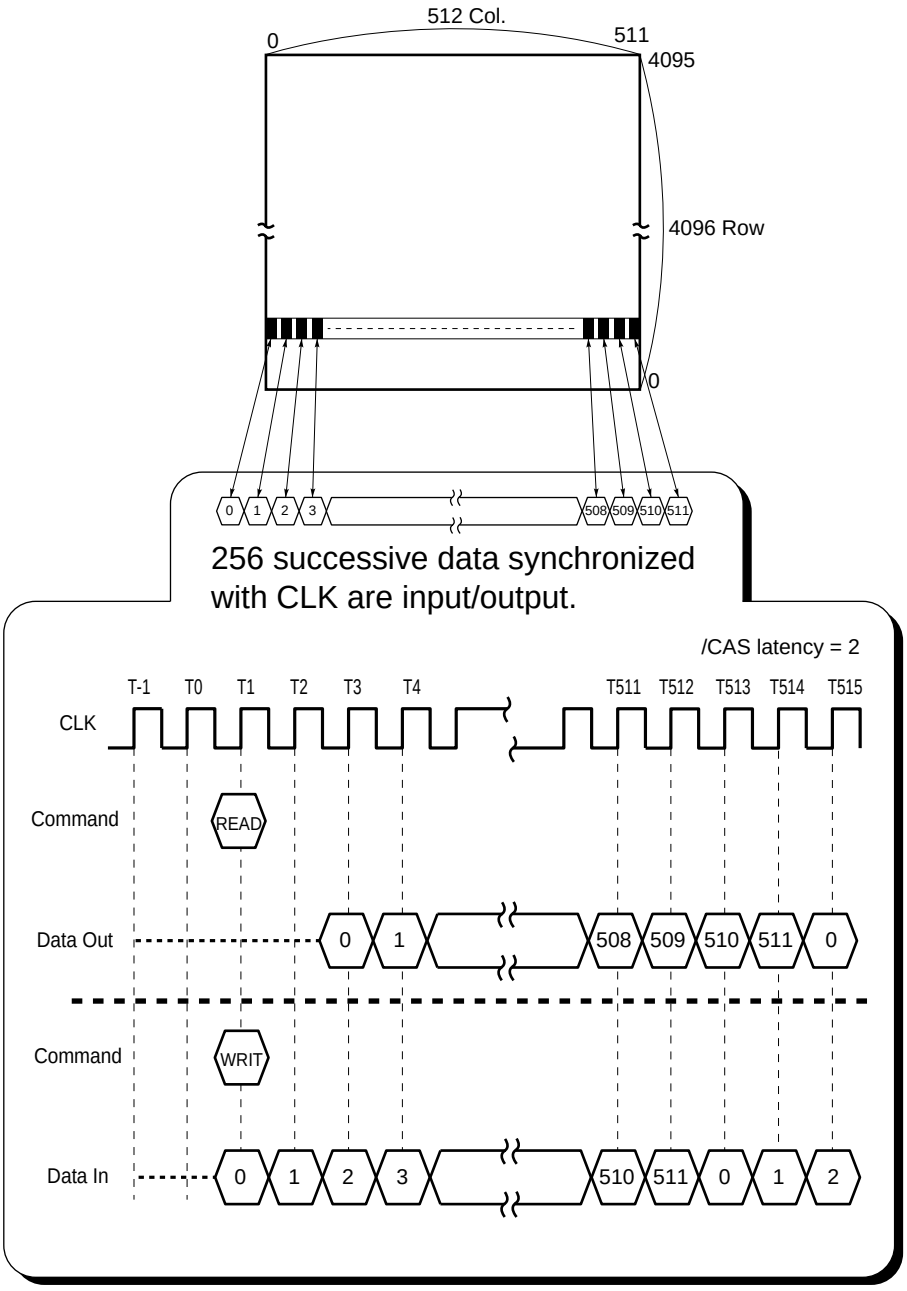


Example 2) Burst length of full page (512)

Data of a full page are successively input or output by inputting a command once.

Because the input/output data perform a wrap-around operation at this time, the burst stop command, read/write command, and precharge command must be used to stop input/output of data.

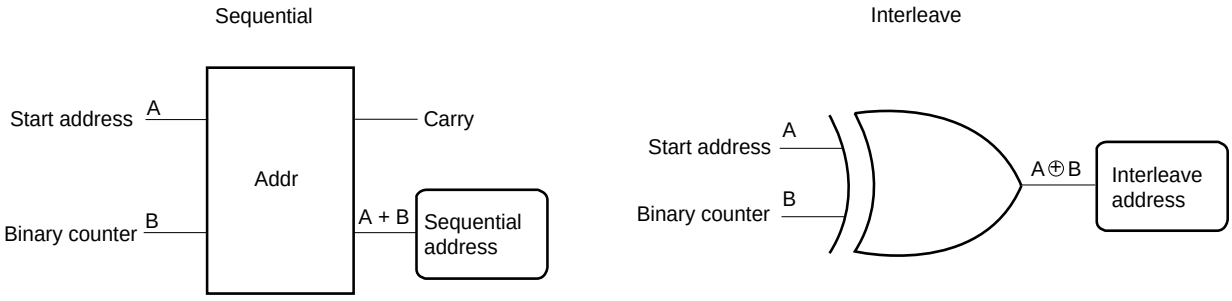
Figure 4-3. Read/Write Cycle with Burst Length of Full Page (256)



(2) Setting of wrap type

The wrap type specifies the sequence in which the address of burst data is incremented. SDRAM supports the sequential type and interleave type. When A3 = 0, the sequential type is selected; when A3 = 1, the interleave type is selected. Which type is to be selected is determined by the type of CPU used in each system.

Figure 4-4. Wrap Type



Burst length and addressing sequence

The following tables indicate the start column address and addressing sequence of each burst length.

[Burst length = 2]

Start Address (column address A0, binary)	Sequential Addressing Sequence (decimal)	Interleave Addressing Sequence (decimal)
0	0, 1	0, 1
1	1, 0	1, 0

[Burst length = 4]

Start Address (column address A1 through A0, binary)	Sequential Addressing Sequence (decimal)	Interleave Addressing Sequence (decimal)
00	0, 1, 2, 3	0, 1, 2, 3
01	1, 2, 3, 0	1, 0, 3, 2
10	2, 3, 0, 1	2, 3, 0, 1
11	3, 0, 1, 2	3, 2, 1, 0

[Burst length = 8]

Start Address (column address A2 through A0, binary)	Sequential Addressing Sequence (decimal)	Interleave Addressing Sequence (decimal)
000	0, 1, 2, 3, 4, 5, 6, 7	0, 1, 2, 3, 4, 5, 6, 7
001	1, 2, 3, 4, 5, 6, 7, 0	1, 0, 3, 2, 5, 4, 7, 6
010	2, 3, 4, 5, 6, 7, 0, 1	2, 3, 0, 1, 6, 7, 4, 5
011	3, 4, 5, 6, 7, 0, 1, 2	3, 2, 1, 0, 7, 6, 5, 4
100	4, 5, 6, 7, 0, 1, 2, 3	4, 5, 6, 7, 0, 1, 2, 3
101	5, 6, 7, 0, 1, 2, 3, 4	5, 4, 7, 6, 1, 0, 3, 2
110	6, 7, 0, 1, 2, 3, 4, 5	6, 7, 4, 5, 2, 3, 0, 1
111	7, 0, 1, 2, 3, 4, 5, 6	7, 6, 5, 4, 3, 2, 1, 0

[Full-page burst (column length: 512)]

Start Address (column address A7 through A0, binary)	Sequential Addressing Sequence (decimal)
000000000	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, ……. 506, 507, 508, 509, 510, 511
000000001	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, ……. 506, 507, 508, 509, 510, 511, 0
000000010	2, 3, 4, 5, 6, 7, 8, 9, 10, ……. 506, 507, 508, 509, 510, 511, 0, 1
000000011	3, 4, 5, 6, 7, 8, 9, 10, ……. 506, 507, 508, 509, 510, 511, 0, 1, 2
: :	: :
111111100	508, 509, 510, 511, 0, 1, 2, 3, 4, 5, 6, ……. 502, 503, 504, 505, 506, 507
111111101	509, 510, 511, 0, 1, 2, 3, 4, 5, 6, 7, ……. 503, 504, 505, 506, 507, 508
111111110	510, 511, 0, 1, 2, 3, 4, 5, 6, 7, 8, ……. 504, 505, 506, 507, 508, 509
111111111	511, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ……. 505, 506, 507, 508, 509, 510

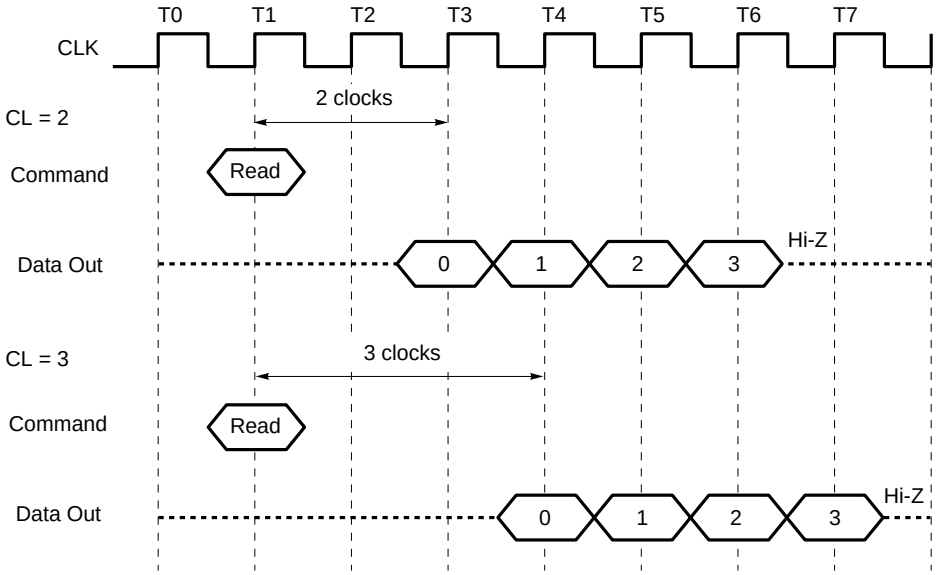
Remark Column length

Part Number	Full Column Length
μPD45128441	2,048 columns
μPD45128841	1,024 columns
μPD45128163	512 columns

(3) Setting of /CAS latency

/CAS latency is the number of clocks required until the first data is read after the read command is input. The value of /CAS latency is limited by the operating frequency of the clock and speed grade of the SDRAM.

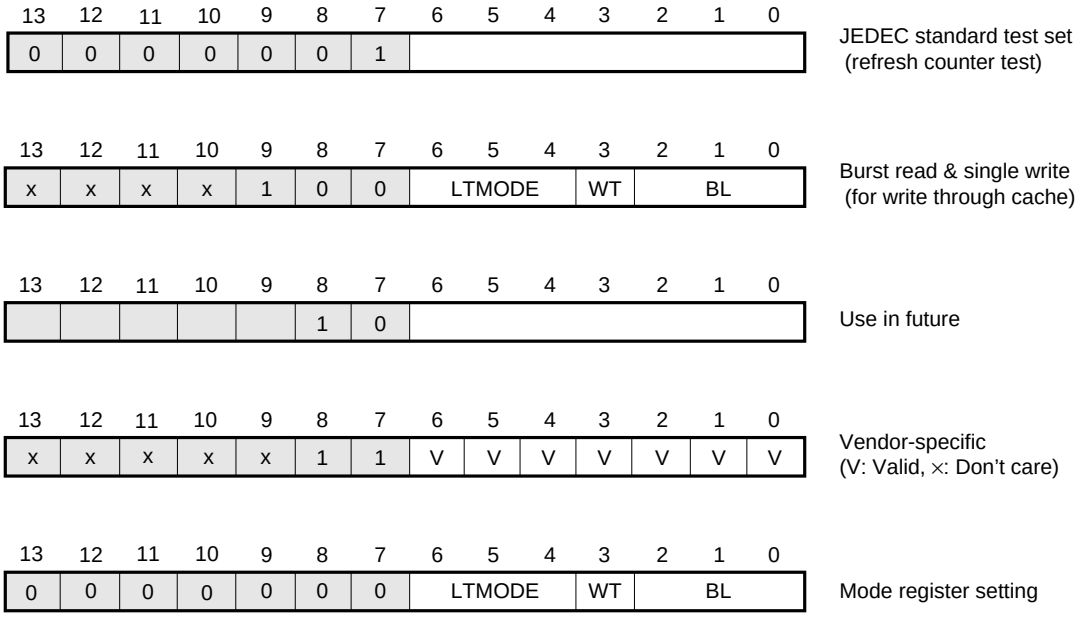
Figure 4-5. Timing Differences between /CAS Latency = 2 and 3



(4) Option

The mode is changed as follows depending on the information on addresses A7 through A13 of the mode register set command.

Figure 4-6. Options in Mode Register (with 128M SDRAM)



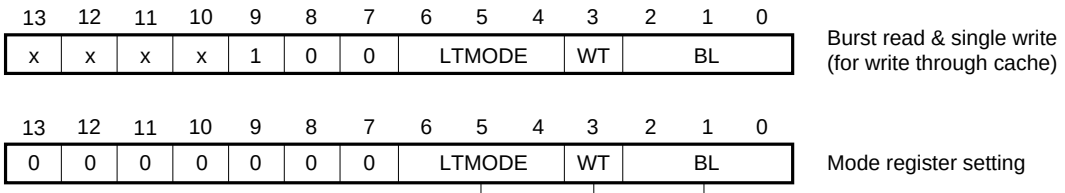
Remark ×: Don't care
V: Valid

4.1.3 Setting of burst length, wrap type, and /CAS latency

To set the burst length, wrap type, and /CAS latency, set the option of mode register setting (A7 through A13: 0000000) by using the mode register set command.

Although the burst length, wrap type, and /CAS latency can also be set by means of burst read and single write (A7, A8, A9: 001), the burst length of the write cycle is set to 1 in this case.

Figure 4-7. Field of Mode Register (with 128M SDRAM)



Burst length

A2	A1	A0	WT = 0	WT = 1
0	0	0	1	1
0	0	1	2	2
0	1	0	4	4
0	1	1	8	8
1	0	0	R (reserved)	R (reserved)
1	0	1	R (reserved)	R (reserved)
1	1	0	R (reserved)	R (reserved)
1	1	1	Full page	R (reserved)

Wrap type

A3	Wrap type (WT)
0	Sequential
1	Interleave

/CAS latency

A6	A5	A4	/CAS latency
0	0	0	R (reserved)
0	0	1	R (reserved)
0	1	0	2
0	1	1	3
1	0	0	R (reserved)
1	0	1	R (reserved)
1	1	0	R (reserved)
1	1	1	R (reserved)

CHAPTER 5 STATUS TRANSITIONS

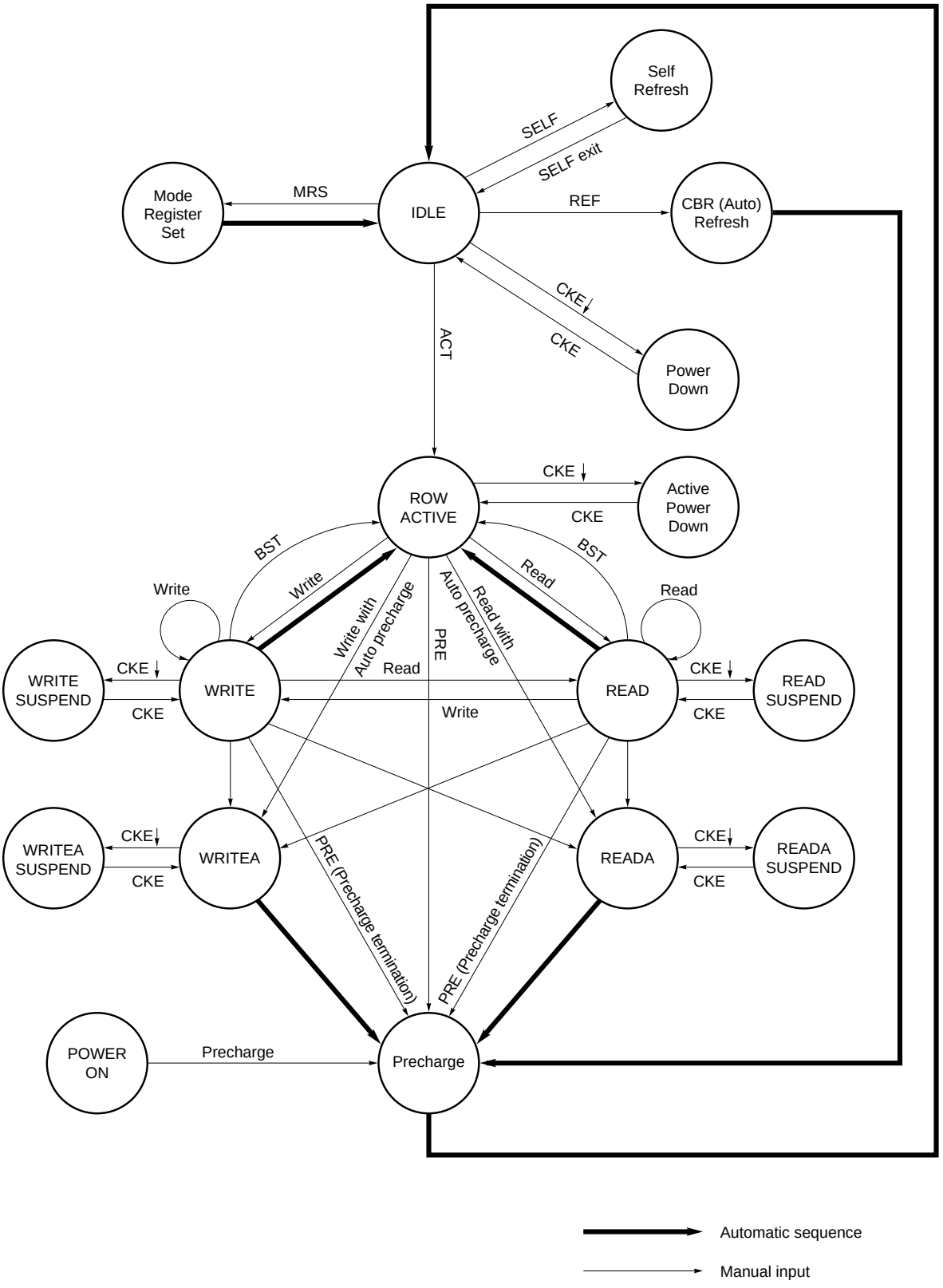
This chapter explains the status transitions of SDRAM.

5.1 Status Transition Diagram

Figure 5-1 shows the status transition diagram of SDRAM. The circles in the figure indicate the device status, and the status is changed in the direction indicated by the arrow.

5.1.1 Status transition diagram of 128M SDRAM (μPD45128163)

Figure 5-1. Status Transition Diagram of 128M SDRAM (μPD45128163)



5.2 Status Description

This section explains the statuses of SDRAM.

5.2.1 Idle

The idle status is the status from which all operations are started. Commands such as the active command, register set command, and refresh command must be input when the device (the bank to be selected) is in the idle status.

5.2.2 Row active

In the row active status, a row address is selected and an operation request (read command or write command) is not made. To change the status from idle to row active, input an active command to the selected row address.

5.2.3 Precharge

In the precharge status, the operation to the current row address ends and an operation to another row address is started. When the precharge command is input, the device automatically returns to the idle status.

5.2.4 Read and write

In the read or write status, the read or write operation is executed. To change the status from row active to this status, input the read or write command to the selected column address. When the read or write operation is completed, the device automatically returns to the row active status.

5.2.5 Read and write with auto precharge

When the read or write command with auto precharge is input, the device automatically starts precharging and returns to the idle status after the read or write operation is completed.

5.2.6 Suspend

If the CKE pin goes low during read operation or write operation (including read and write with auto precharge), the operation is temporarily stopped.

5.2.7 Mode register setting

The mode register can be set when all banks of the device are in the idle status. When data has been written to the mode register, the device automatically returns to the idle status.

5.2.8 CBR (auto) refresh

The CBR refresh command can be executed when all banks of the device are in the idle status. When the CBR refresh command is input, a certain row address of all banks is selected, and refreshing is executed. When CBR refreshing is completed, the device automatically returns to the idle status.

5.2.9 Self refresh

The self refresh command can be executed when all banks of the device are in the idle status, like the CBR refresh command. In the self refresh status, the device automatically performs refreshing. During this time, it is not necessary to execute the refresh command from an external source. When the device exits from the self refresh status, it automatically returns to the idle status.

5.2.10 Power down

If CKE is made low in the idle status or row active status, the power down mode is set. In this mode, all input buffers except CLK and CKE are off, and the power consumption of the device is lowered. To return to the original status (idle or active), make CKE high.

CHAPTER 6 COMMAND OPERATIONS

This chapter explains the points to be noted in executing commands.

For command lists, refer to **2.2.2 SDRAM command list**, **5.1.1 Status transition diagram of 128M SDRAM**.

6.1 Command Execution Condition

The status in which each command can be executed is shown below.

Table 6-1. Command Executable Condition

Command	Symbol	Command Executable (Input) Condition	Remark
Mode register set	MRS	All banks are in idle status.	
CBR (auto) refresh	REF	All banks are in idle status.	
Self refresh	SELF	All banks are in idle status.	
Precharge select bank	PRE	t_{RAS} after active command input (selected bank)	
Precharge all banks	PALL	t_{RAS} after active command input (all banks)	
Bank active	ACT	Selected bank is in idle status.	
Write	WRIT	t_{RCD} after active command input (selected bank)	
Write with auto precharge	WRITA	t_{RCD} after active command input (selected bank)	
Read	READ	t_{RCD} after active command input (selected bank)	
Read with auto precharge	READA	t_{RCD} after active command input (selected bank)	
Burst stop	BST	During read or write operation	
No operation	NOP	All status	
Device deselect	DESL	All status	

6.2 Command Operations of 128M SDRAM (μPD45128163)

The operation status that changes depending on the input command is shown below.

Current status: Idle

Input Command	Action	Notes
DESL	NOP or power down	**Note 1**
NOP	NOP or power down	**Note 1**
BST	NOP or power down	**Note 1**
READ/READA	Illegal	**Note 2**
WRIT/WRITA	Illegal	**Note 2**
ACT	Row activating	
PRE/PALL	NOP	**Note 3**
REF/SELF	CBR (auto) refresh/self refresh	**Note 4**
MRS	Mode register set	**Note 5**

Notes 1. If all banks are in the idle status and CKE is inactive (low level), the power down mode is set.
2. These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).
3. Precharging is enabled depending on the status of the bank specified by the bank address (BA).
4. If all banks are in the idle status and CKE is inactive (low level), the self refresh mode is set.
5. This command is illegal if there is a bank that is not in the idle status.

Current status: Row active

Input Command	Action	Notes
DESL	NOP	
NOP	NOP	
BST	NOP	
READ/READA	Begin read: Determine AP	
WRIT/WRITA	Begin write: Determine AP	
ACT	Illegal	**Note 1**
PRE/PALL	Precharge	**Note 2**
REF/SELF	Illegal	
MRS	Illegal	

Notes 1. These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).
2. This command is illegal if t_{RAS} is not satisfied.

Remark Row active: Status after t_{RCD} from active command (ACT) input (selected bank)

Current status: Read

Input Command	Action	Notes
DESL	Continue burst to end → Row active	
NOP	Continue burst to end → Row active	
BST	Burst stops → Row active	
READ/READA	Burst stops → Start read: Determine AP	**Note 1**
WRIT/WRITA	Burst stops → Start write: Determine AP	**Note 1**
ACT	Illegal	**Note 2**
PRE/PALL	Burst stops → precharging	
REF/SELF	Illegal	
MRS	Illegal	

Notes 1. The burst interrupt condition must be satisfied.

2. These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).

Current status: Write

Input Command	Action	Notes
DESL	Continue burst to end → Write recovering	
NOP	Continue burst to end → Write recovering	
BST	Burst stops → Row active	
READ/READA	Burst stops → Start read: Determine AP	**Note 1**
WRIT/WRITA	Burst stops → Start write: Determine AP	**Note 1**
ACT	Illegal	**Note 2**
PRE/PALL	Burst stops → precharging	**Note 1**
REF/SELF	Illegal	
MRS	Illegal	

Notes 1. The burst interrupt condition must be satisfied.

2. These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).

Current status: Read with auto precharge

Input Command	Action	Notes
DESL	Continue burst to end → Precharging	
NOP	Continue burst to end → Precharging	
BST	Illegal	
READ/READA	Illegal	**Note**
WRIT/WRITA	Illegal	**Note**
ACT	Illegal	**Note**
PRE/PALL	Illegal	**Note**
REF/SELF	Illegal	
MRS	Illegal	

Note These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).

Current status: Write with auto precharge

Input Command	Action	Notes
DESL	Continue burst to end → Precharging	
NOP	Continue burst to end → Precharging	
BST	Illegal	
READ/READA	Illegal	**Note**
WRIT/WRITA	Illegal	**Note**
ACT	Illegal	**Note**
PRE/PALL	Illegal	**Note**
REF/SELF	Illegal	
MRS	Illegal	

Note These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).

Current status: Precharging

Input Command	Action	Notes
DESL	NOP → Enter idle after t_{RP}	
NOP	NOP → Enter idle after t_{RP}	
BST	Illegal	
READ/READA	Illegal	**Note 1**
WRIT/WRITA	Illegal	**Note 1**
ACT	Illegal	**Note 1**
PRE/PALL	NOP → Enter idle after t_{RP}	**Note 2**
REF/SELF	Illegal	
MRS	Illegal	

Notes 1. These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).

2. Precharging is enabled depending on the status of the bank specified by the bank address (BA).

Current status: Row activating

Input Command	Action	Notes
DESL	NOP → Enter bank active after t_{RCD}	
NOP	NOP → Enter bank active after t_{RCD}	
BST	Illegal	
READ/READA	Illegal	**Note 1**
WRIT/WRITA	Illegal	**Note 1**
ACT	Illegal	**Notes 1, 2**
PRE/PALL	Illegal	**Note 1**
REF/SELF	Illegal	
MRS	Illegal	

Notes 1. These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).

2. This command is illegal if t_{RRD} is not satisfied.

Remark Row activating: Status of less than t_{RCD} from active command (ACT) input (selected bank)

Current status: Write recovering

Input Command	Action	Notes
DESL	NOP → Enter row active after t_{DPL}	
NOP	NOP → Enter row active after t_{DPL}	
BST	NOP → Enter row active after t_{DPL}	
READ/READA	Start read, Determine AP	**Note 1**
WRIT/WRITA	Start write, Determine AP	
ACT	Illegal	**Note 2**
PRE/PALL	Illegal	**Note 2**
REF/SELF	Illegal	
MRS	Illegal	

Notes 1. Refer to **10.1.1 Data interrupt by read command**.

2. These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).

Current status: Write recovering with auto precharge

Input Command	Action	Notes
DESL	NOP → Enter precharge after t_{DPL}	
NOP	NOP → Enter precharge after t_{DPL}	
BST	NOP → Enter precharge after t_{DPL}	
READ/READA	Illegal	**Note 1**
WRIT/WRITA	Illegal	
ACT	Illegal	**Note 2**
PRE/PALL	Illegal	**Note 2**
REF/SELF	Illegal	
MRS	Illegal	

Notes 1. Refer to **10.1.1 Data interrupt by read command**.

2. These commands are illegal for the same bank (these commands may be valid depending on the status of the bank specified by the bank address (BA)).

Current status: Refreshing

Input Command	Action	Notes
DESL	NOP → Enter idle after t_{RC}	
NOP	NOP → Enter idle after t_{RC}	
BST	Illegal	
READ/READA	Illegal	
WRIT/WRITA	Illegal	
ACT	Illegal	
PRE/PALL	Illegal	
REF/SELF	Illegal	
MRS	Illegal	

Current status: Mode register set

Input Command	Action	Notes
DESL	NOP → Enter idle after t_{RSC}	
NOP	NOP → Enter idle after t_{RSC}	
BST	Illegal	
READ/READA	Illegal	
WRIT/WRITA	Illegal	
ACT	Illegal	
PRE/PALL	Illegal	
REF/SELF	Illegal	
MRS	Illegal	

CHAPTER 7 BASIC OPERATION MODE

This chapter explains the basic operation in the read, write, and refresh modes.

7.1 Read Mode

The read operation is executed when the read command is input in the row active status. The following series of operations are performed in the read cycle.

(1) The corresponding bank is activated by inputting the active command and a row address.

(2) After the lapse of t_{RCD}, the read command and a column address are input.

(3) The data at the specified address is output in accordance with the burst length (BL) and /CAS latency (CL) set by the mode register.

(4) After the lapse of t_{RAS}, the precharge command is input. The earliest input timing of the precharge command so as to read data without all burst data lost is as follows:
When CL = 2: One clock earlier than the last read data
When CL = 3: Two clocks earlier than the last read data

Caution The input timing of the precharge command differs depending on the model of the SDRAM. For details, refer to the Data Sheet of each model.

(5) After the lapse of t_{RP}, the corresponding bank enters the idle status.

Figure 7-1 shows the timing chart of the basic operation of the read cycle where BL = 4.

Figure 7-1. Read Cycle

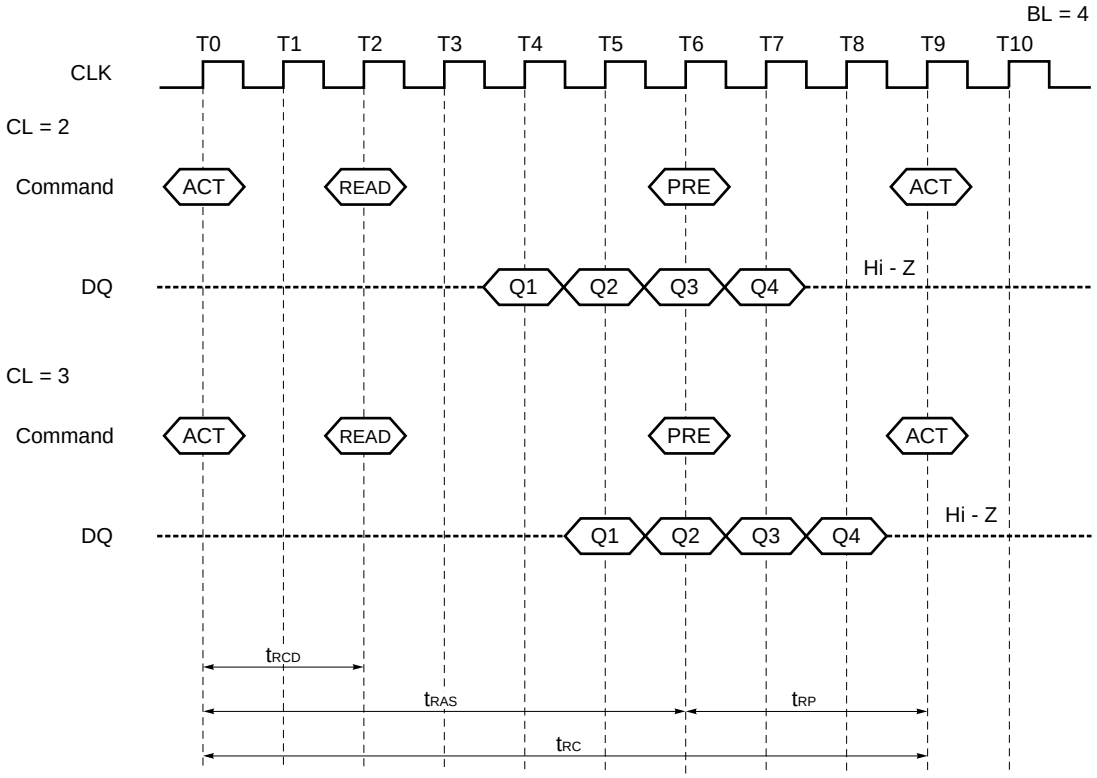


Figure 7-2 shows the read cycle when the read command with auto precharge is selected. When the read command with auto precharge is input, it is not necessary to input the precharge command because the precharge operation is automatically started.

When using auto precharge in the read cycle, it is necessary to know when the precharge operation is started because t_{RAS} and t_{RP} must be satisfied. The next active command for the bank being precharged cannot be executed until the precharge cycle is completed. The active command can be input t_{RP} after the start of auto precharge.

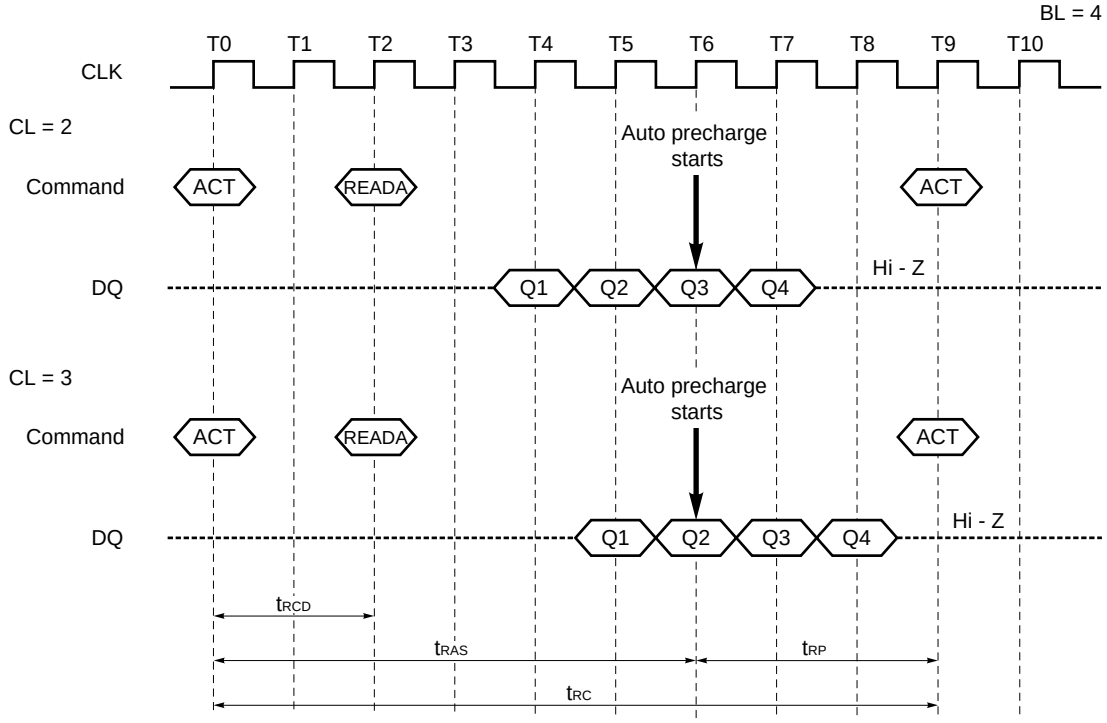
The timing to start auto precharge is as follows:

When CL = 2: One clock earlier than the last read data

When CL = 3: Two clocks earlier than the last read data

Caution The timing to start auto precharge differs depending on the model of the SDRAM. For details, refer to the Data Sheet of each model.

Figure 7-2. Read Cycle with Auto Precharge



7.2 Write Mode

The write operation is executed when the write command is input in the row active status. The following series of operations are performed in the write cycle.

(1) The corresponding bank is activated by inputting the active command and a row address.

(2) After the lapse of t_{RCD}, the write command and a column address are input.

(3) Data is input to the specified address in accordance with the burst length (BL) set by the mode register.

(4) After the lapse of t_{RAS}, the precharge command is input. t_{DPL} must be satisfied to write all data correctly to memory cells.

(5) After the lapse of t_{RP}, the corresponding bank enters the idle status.

Figure 7-3 shows the timing chart of the basic operation of the write cycle where BL = 4.

Figure 7-3. Write Cycle

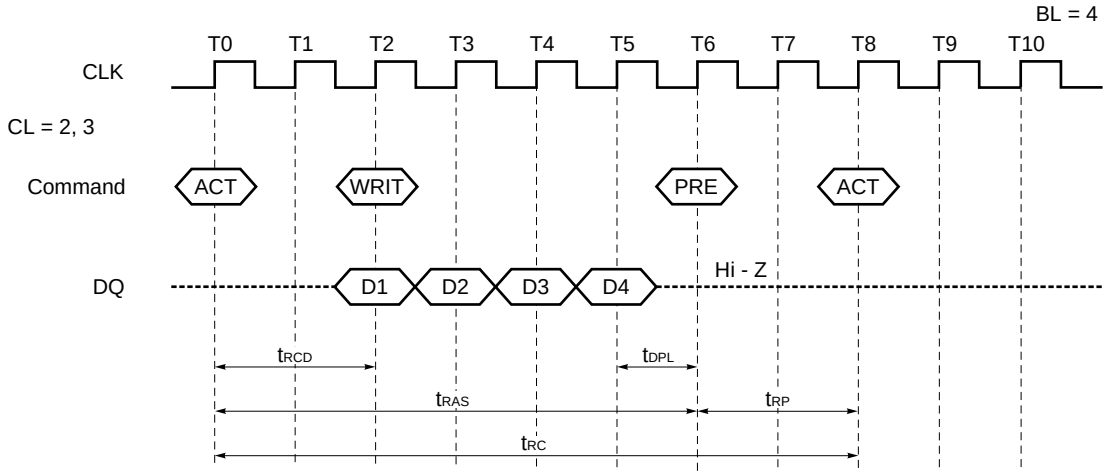


Figure 7-4 shows the write cycle when the write command with auto precharge is selected. When the write command with auto precharge is input, it is not necessary to input the precharge command because the precharge operation is automatically started.

It is not necessary to know when the precharge operation is started in the write cycle because only t_{DAL} has to be satisfied to input the next active command for the bank being precharged.

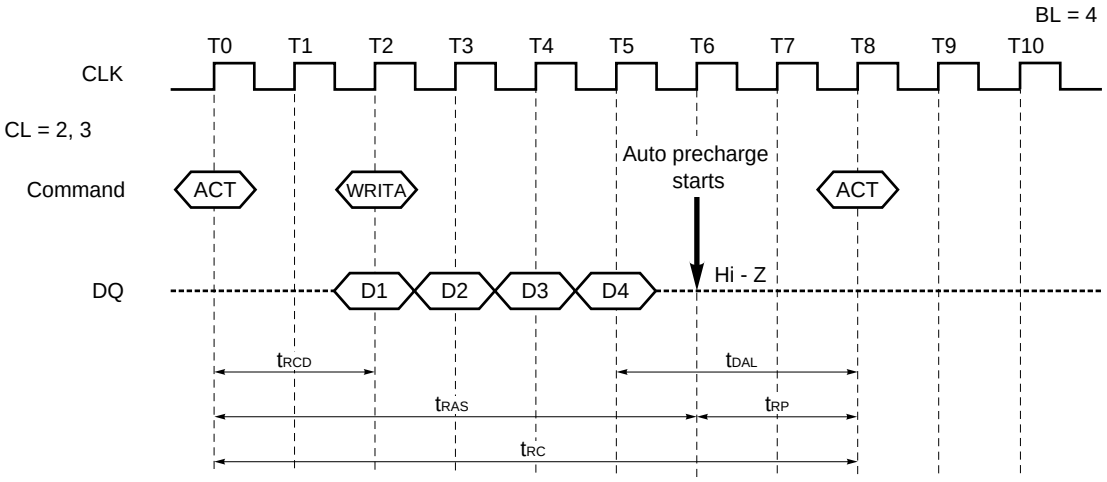
The timing to start auto precharge is as follows:

When CL = 2: One clock after the last write data

When CL = 3: One clock after the last write data

Caution The timing to start auto precharge differs depending on the model of SDRAM. For details, refer to the Data Sheet of each model.

Figure 7-4. Write Cycle with Auto Precharge



7.3 Refresh Mode

Like the conventional DRAM, a refresh operation is necessary for SDRAM. Refreshing is performed in two modes: CBR (auto) refresh[Note 1] and self refresh[Note 2].

The following series of operations are performed in the CBR (auto) refresh cycle.

(1) Because a single refresh operation must be executed in the idle status, if the device is not in the idle status, it must be set in the idle status by executing the precharge operation.

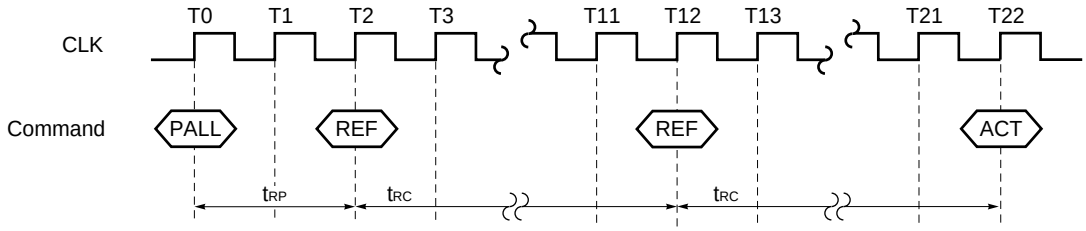
(2) Input the CBR (auto) refresh command.

Because the internal refresh counter of the device automatically generates a refresh address, it is not necessary to specify an address from an external device.

(3) The device enters the idle status after the lapse of t_{RC}.

Figure 7-5 shows the timing chart of the basic operation in the CBR (auto) refresh cycle.

Figure 7-5. CBR (Auto) Refresh Cycle



Notes 1. The refresh operation is completed even if a read or write cycle is executed to all the row addresses within t_{REF}.

2. Refer to **9.2.3 Self refresh mode**.

CHAPTER 8 DQM CONTROL OPERATION

This chapter explains DQM control. The DQM signal masks input/output data. The control timing of the DQM pin differs depending on whether the cycle is read or write.

8.1 DQM Pin

The input/output pins to be controlled by the DQM signal differ depending on the bit organization as follows:

×4-bit organization

Four bits of input/output pins are controlled simultaneously.
DQM controls DQ0 through DQ3.

×8-bit organization

Eight bits of input/output pins are controlled simultaneously.
DQM controls DQ0 through DQ7.

×16-bit organization

The high-order 8 bits and low-order 8 bits of the 16 bits of input/output pins are controlled independently.
LDQM controls DQ0 through DQ7 (low-order 8 bits).
UDQM controls DQ8 through DQ15 (high-order 8 bits).

×32-bit organization

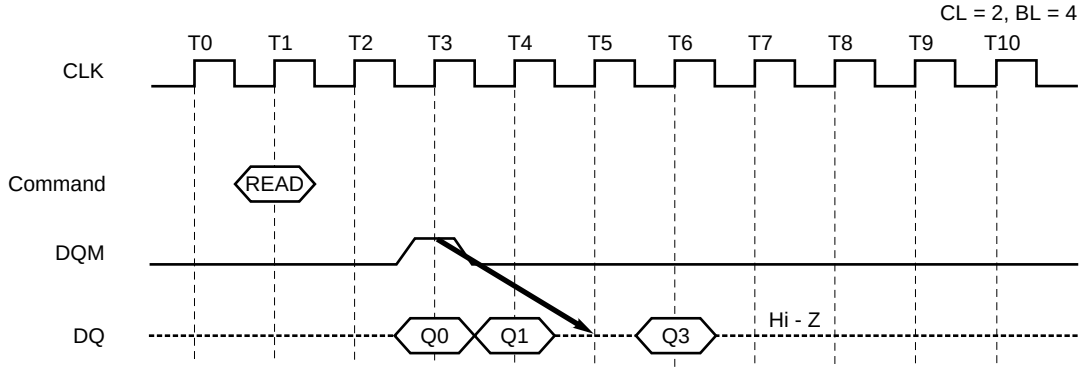
Thirty-two bits of input/output pins are controlled in 8-bit units independently.
DQM0 controls DQ0 through DQ7.
DQM1 controls DQ8 through DQ15.
DQM2 controls DQ16 through DQ23.
DQM3 controls DQ24 through DQ31.

8.2 DQM Control in Read Cycle

The DQM latency in the read cycle is two clocks regardless of the /CAS latency.

As shown in Figure 8-1, the output buffer corresponding to the DQM goes in to a high-impedance in state T5 two clocks after T3 and the read data is stopped when some DQM goes high.

Figure 8-1. DQM Control during Read Operation

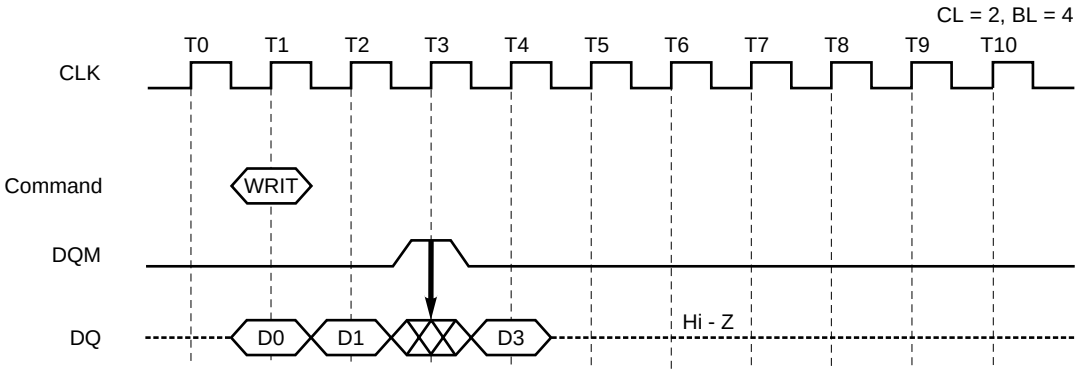


8.3 DQM Control in Write Cycle

The DQM latency in the write cycle is 0 regardless of the /CAS latency.

As shown in Figure 8-2, data corresponding to the DQM is not written when some DQM goes high (T3).

Figure 8-2. DQM Control during Write Operation



8.4 DQM Truth Table

Table 8-1 shows the command truth table of DQM.

Table 8-1. DQM Truth Table

Function	Symbol	CKE		DQM	
		n – 1	n	U	L
Data write/output enable	ENB	H	×	L	
Data write/output disable	MASK	H	×	H	
Upper byte write enable/output enable	ENBU	H	×	L	×
Lower byte write enable/output enable	ENBL	H	×	×	L
Upper byte write disable/output disable	MASKU	H	×	H	×
Lower byte write disable/output disable	MASKL	H	×	×	H

Remark H = High level, L = Low level, × = Don't care

CHAPTER 9 CKE CONTROL OPERATION

This chapter explains the basic control method by the CKE signal and limitations (control timing and control signal level) during operation. CKE is a signal that controls inputting CLK.

When CKE = H at the rising edge of preceding CLK

The rising edge of the next CLK is valid and each signal is input.

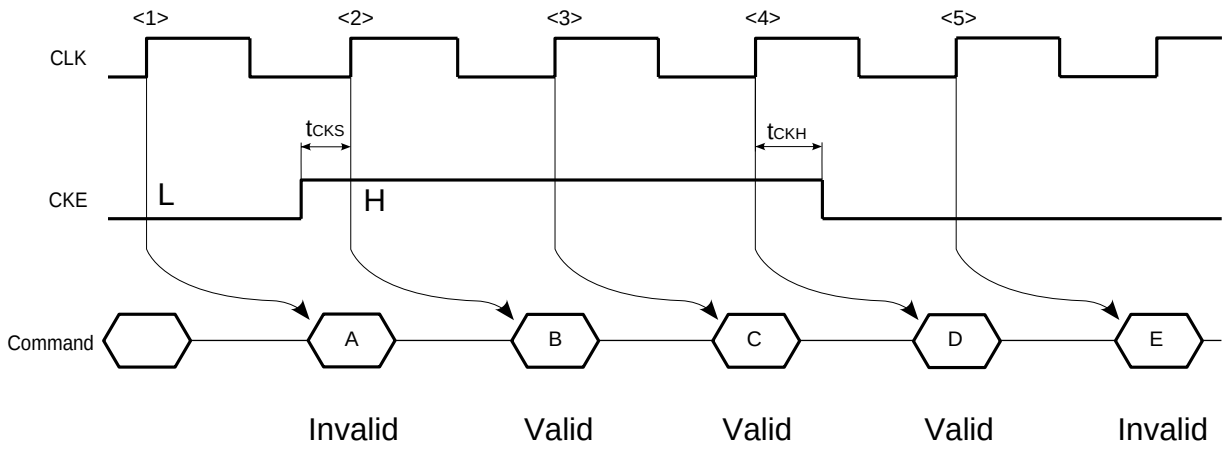
When CKE = L at the rising edge of preceding CLK

The rising edge of the next CLK is invalid and each signal is not input.

9.1 Basic Control

Figure 9-1 shows the signal input timing when CKE = H.

Figure 9-1. Signal Input Timing Controlled by CKE



If CKE = high level at the rising edge of CLK as in <2>, <3>, and <4> in the above figure (where setup time t_{CKS} and hold time t_{CKH} are satisfied), commands B, C, and D input at the rising edge of the next CLK can be loaded. When CKE = low level at the rising edge of CLK as in <1> and <5>, the command input at the rising edge of the next CLK is invalid and is not loaded.

The above control is performed by controlling the internal clock of the device. If CKE goes low in the middle of an operation, the internal operation of the memory is temporarily stopped. When CKE goes high, the internal operation is resumed.

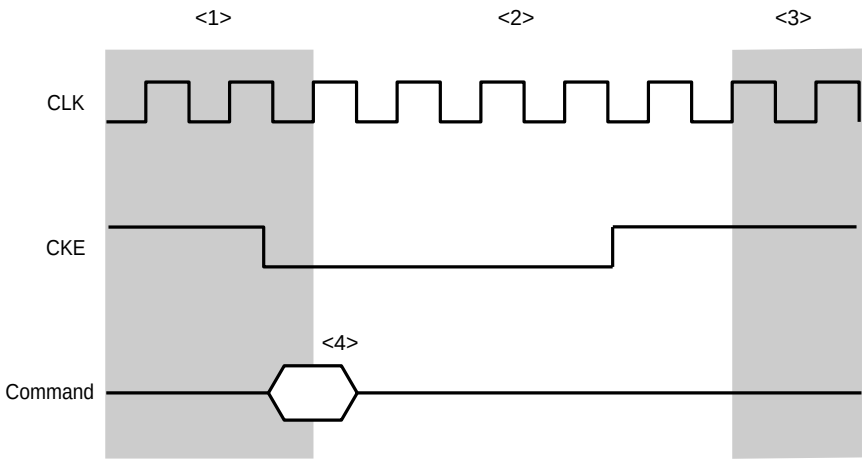
9.2 Example of CKE Control

The CKE control operations are performed in the following modes:

Power down mode
Clock suspend mode
Self refresh mode

Figure 9-2 illustrates the status transition of the device if the CKE signal level is changed after any command <4> has been input. In this figure, the command is loaded only during the period between <1> and <3>, and the command cannot be loaded during period <2>.

Figure 9-2. Example of CKE Control



9.2.1 Power down mode

In the power down mode, the internal clock of the device is deactivated to reduce the power consumption of the device.

Figure 9-3 shows the timing in the power down mode.

Starting power down mode

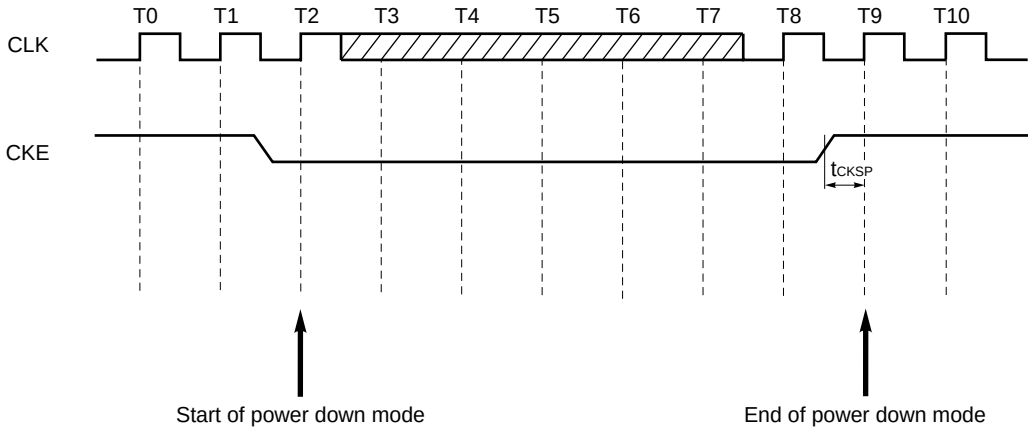
The power down mode is started at T2 if the input level of the CKE signal is changed from high to low (T1-T2) when all the banks are in the idle status or row active status (bank active).

When the power down operation is started, all the input signals other than the CKE signal are in the don't care status (high or low level), and the data bus goes into a high-impedance state. The CKE signal must be kept low in power down status.

Releasing power down mode

The power down mode is released at T9 if the input level of the CKE signal is changed from low to high (T8-T9), and the next command can be input at the timing subsequent to T10. However, at least one clock must be supplied to the device before the CKE signal goes high.

Figure 9-3. Power Down Mode



Remarks 1. Commands cannot be input in the power down mode.

2. Make sure that t_{REF} is satisfied.

9.2.2 Clock suspend mode

The clock suspend operation stops transmission of the clock to the internal circuits of the device during burst transfer of data to stop the operation of the device.

(1) Clock suspend mode during a read operation

Figure 9-4 shows the timing of the clock suspend mode during a read operation.

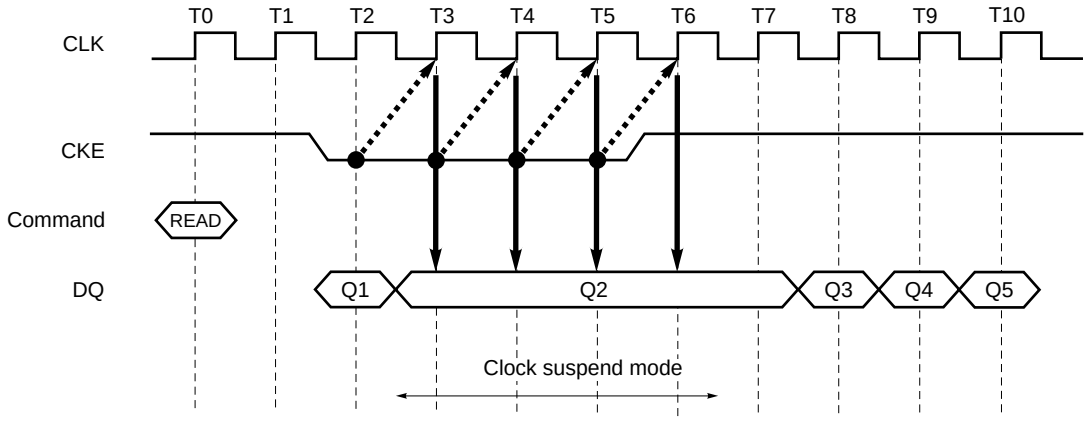
Starting clock suspend mode

The suspend operation is started at T3 if the input level of the CKE signal is changed from high to low (T1-T2), regardless of the value of the /CAS latency. In the clock suspend mode, outputting data (Q2) goes on.

Releasing clock suspend mode

The clock suspend mode is released at T6 if the input level of the CKE signal is changed from low to high (T5-T6), the read operation is resumed at T7, and new data (Q3) is output at T8.

Figure 9-4. Clock Suspend Mode (read cycle: CL = 2)



(2) Clock suspend mode during a write operation

Figure 9-5 shows the timing of the clock suspend mode during a write operation.

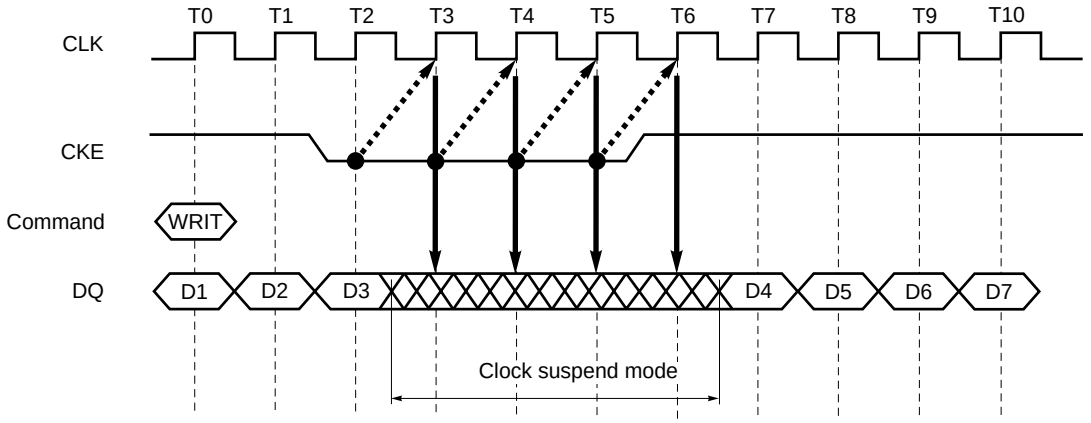
Starting clock suspend mode

The suspend operation is started at T3 if the input level of the CKE signal is changed from high to low (T1-T2). Data cannot be written in the clock suspend mode.

Releasing clock suspend mode

The clock suspend mode is released at T6 if the input level of the CKE signal is changed from low to high (T5-T6), and the write operation is resumed at T7.

Figure 9-5. Clock Suspend Mode (write cycle)



9.2.3 Self refresh mode

The self refresh operation deactivates the clock in the same manner as in the power down mode to reduce the power consumption of the device and automatically executes a refresh operation by using an internal refresh counter.

This mode is effective for not accessing the device for a long time though the memory cell data must be held.

Figure 9-6 illustrates the timing of self refresh.

Starting self refresh mode

The self refresh operation is started at T2 if the input level of the CKE signal is changed from high to low (T1-T2) when the refresh command (REF) is input.

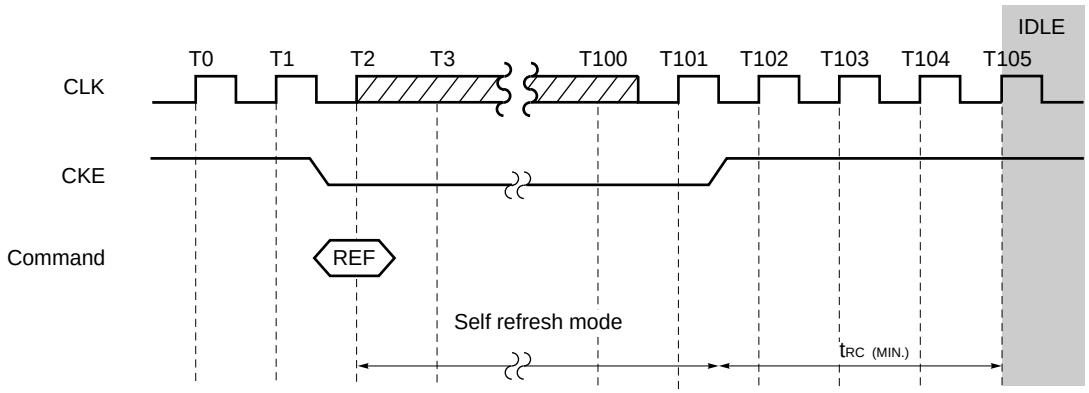
When the self refresh operation has been started, all the input signals except the CKE signal is in the don't care status (high or low level), and the data bus goes into a high-impedance state.

The CKE signal must be kept low in the self refresh mode.

Releasing self refresh mode

The self refresh mode is released if the input level of the CKE signal is changed from low to high (T101-T102). However, at least one clock must be supplied to the device before the CKE signal goes high. In addition, $t_{RC(MIN.)}$ must be satisfied by the NOP or DESL command if the next command is input.

Figure 9-6. Self Refresh Mode



Caution When using concentrated refresh during normal operation, CBR refresh (auto refresh) must be concentrated and executed for the duration of the total number of refresh cycles before and after the self refresh operation.

9.2.4 CKE command truth table (128M SDRAM (μPD45128163))

The CKE command truth table is shown below.

Current status: Self refresh

CKE(n – 1)	CKE(n)	/CS	/RAS	/CAS	/WE	Address	Action	Notes
H	×	×	×	×	×	×	Invalid	
L	H	H	×	×	×	×	Releases self refresh mode → Self refresh recovery	
L	H	L	H	H	×	×	Releases self refresh mode → Self refresh recovery	
L	H	L	H	L	×	×	Illegal	
L	H	L	L	×	×	×	Illegal	
L	L	×	×	×	×	×	Continues self refresh	

Remark H: High level, L: Low level, ×: High or low level (Don't care)

Current status: Self refresh recovery

CKE(n – 1)	CKE(n)	/CS	/RAS	/CAS	/WE	Address	Action	Notes
H	H	H	×	×	×	×	NOP → after t_{RC}, idle status	
H	H	L	H	H	×	×	NOP → after t_{RC}, idle status	
H	H	L	H	L	×	×	Illegal	
H	H	L	L	×	×	×	Illegal	
H	L	H	×	×	×	×	Illegal	
H	L	L	H	H	×	×	Illegal	
H	L	L	H	L	×	×	Illegal	
H	L	L	L	×	×	×	Illegal	

Remark H: High level, L: Low level, ×: High or low level (Don't care)

Current status: Power down

CKE(n – 1)	CKE(n)	/CS	/RAS	/CAS	/WE	Address	Action	Notes
H	×	×	×	×	×	×	Invalid	
L	H	H	×	×	×	×	Releases power down mode → idle status	
L	H	L	H	H	H	×	Releases power down mode → idle status	
L	L	×	×	×	×	×	Continues power down mode.	

Remark H: High level, L: Low level, ×: High or low level (Don't care)

Current status: When all banks are in idle status

CKE(n – 1)	CKE(n)	/CS	/RAS	/CAS	/WE	Address	Action	Notes
H	H	H	×	×	×		Refer to **6.2 Command Operations of 128M SDRAM**	
H	H	L	H	×	×		Refer to **6.2 Command Operations of 128M SDRAM**	
H	H	L	L	H	×		Refer to **6.2 Command Operations of 128M SDRAM**	
H	H	L	L	L	H	×	CBR(auto) refresh entry	
H	H	L	L	L	L	V	Refer to **6.2 Command Operations of 128M SDRAM**	
H	L	H	×	×	×		Refer to **6.2 Command Operations of 128M SDRAM**	
H	L	L	L	H	×		Refer to **6.2 Command Operations of 128M SDRAM**	
H	L	L	L	L	H	×	Self refresh entry	
H	L	L	L	L	L	V	Refer to **6.2 Command Operations of 128M SDRAM**	
L	×	×	×	×	×	×	Power down entry	

Remark H: High level, L: Low level, ×: High or low level (Don't care), V: Valid data

Current status: Row active

CKE(n – 1)	CKE(n)	/CS	/RAS	/CAS	/WE	Address	Action	Notes
H	×	×	×	×	×	×	Refer to **6.2 Command Operations of 128M SDRAM**	
L	×	×	×	×	×	×	Power down entry	

Remark H: High level, L: Low level, ×: High or low level (Don't care), V: Valid data

Current status: Other than above

CKE(n – 1)	CKE(n)	/CS	/RAS	/CAS	/WE	Address	Action	Notes
H	H	×	×	×	×	×	Refer to **6.2 Command Operations of 128M SDRAM**	
H	L	×	×	×	×	×	Starts clock suspend in next cycle.	**Note**
L	H	×	×	×	×	×	Releases clock suspend in next cycle.	
L	L	×	×	×	×	×	Continues clock suspend.	

Note The command indicated as “Illegal” in **6.2 Command Operations of 128M SDRAM (μPD45128163)** cannot be used.

Remark H: High level, L: Low level, ×: High or low level (Don’t care)

CHAPTER 10 BURST OPERATION

This chapter explains the burst operation.

10.1 Ending Burst Operation

The burst operation can be ended in the following ways:

(1) By using read command

(2) By using write command

(3) By using burst stop command

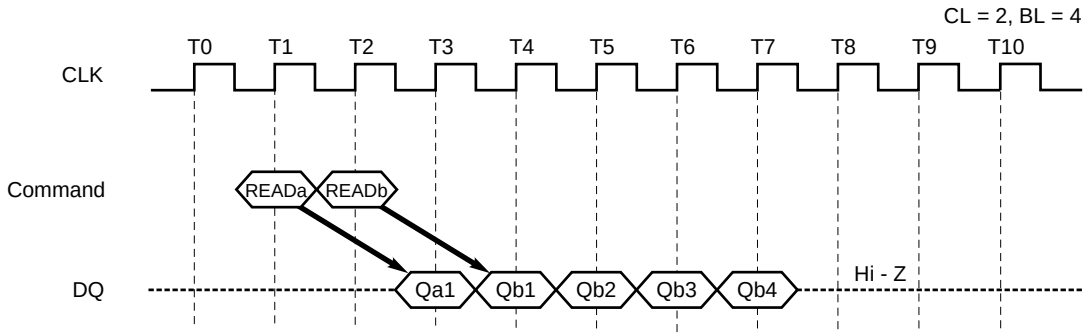
(4) By using precharge command

10.1.1 Data interrupt by read command

(1) Read cycle

The preceding burst read operation can be aborted and a new burst read operation can be started by inputting a new read command in the read cycle. The data for the new read command (READb) is output after the lapse of the /CAS latency.

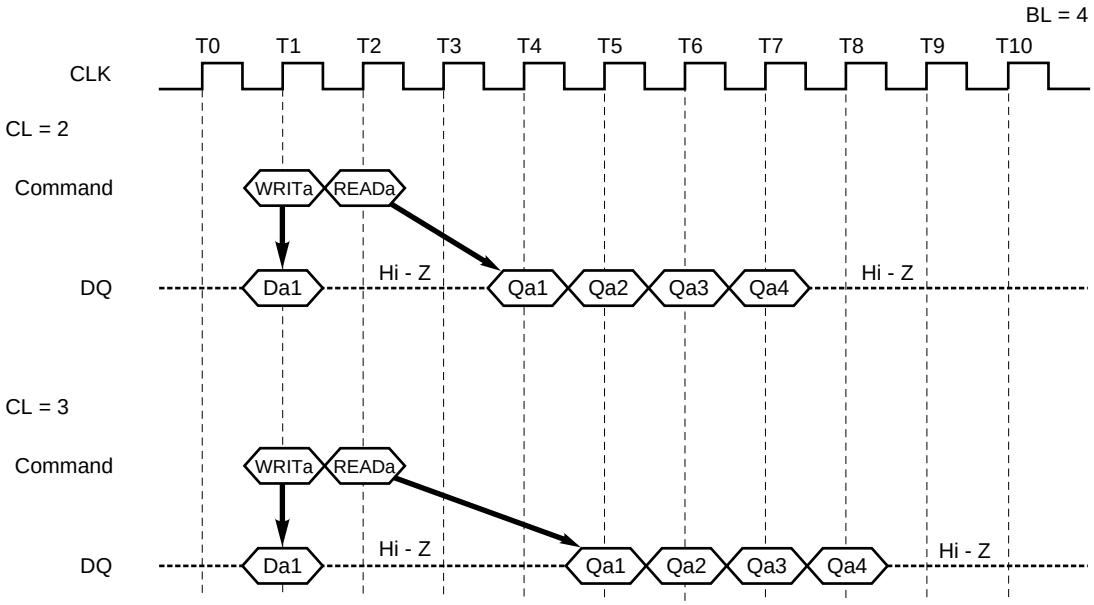
Figure 10-1. Read/Read Command



(2) Write cycle

The preceding burst write operation can be aborted and a new burst read operation can be started by inputting a new read command in the write cycle. The data of the read command (READa) is output after the lapse of the /CAS latency. The preceding write operation (WRITa) writes only the data input before the read command. The data bus must go into a high-impedance state at least one cycle before output of the latest data.

Figure 10-2. Write/Read Command

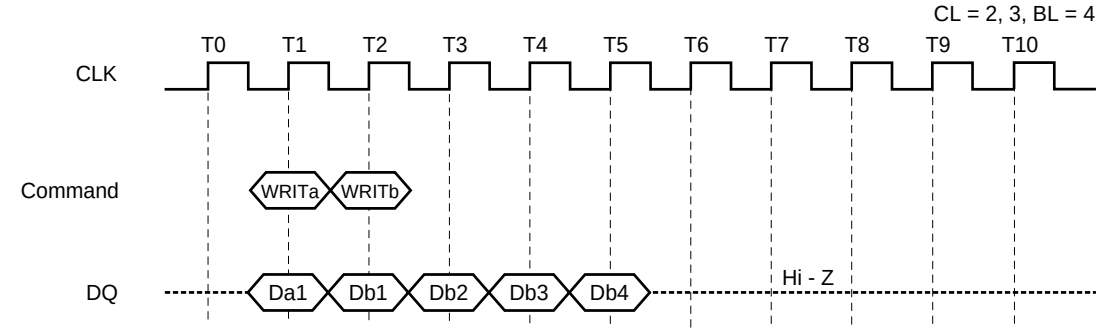


10.1.2 Data interrupt by write command

(1) Write cycle

The preceding burst write operation can be aborted and a new burst write operation can be started by inputting a new write command in the write cycle.

Figure 10-3. Write/Write Command

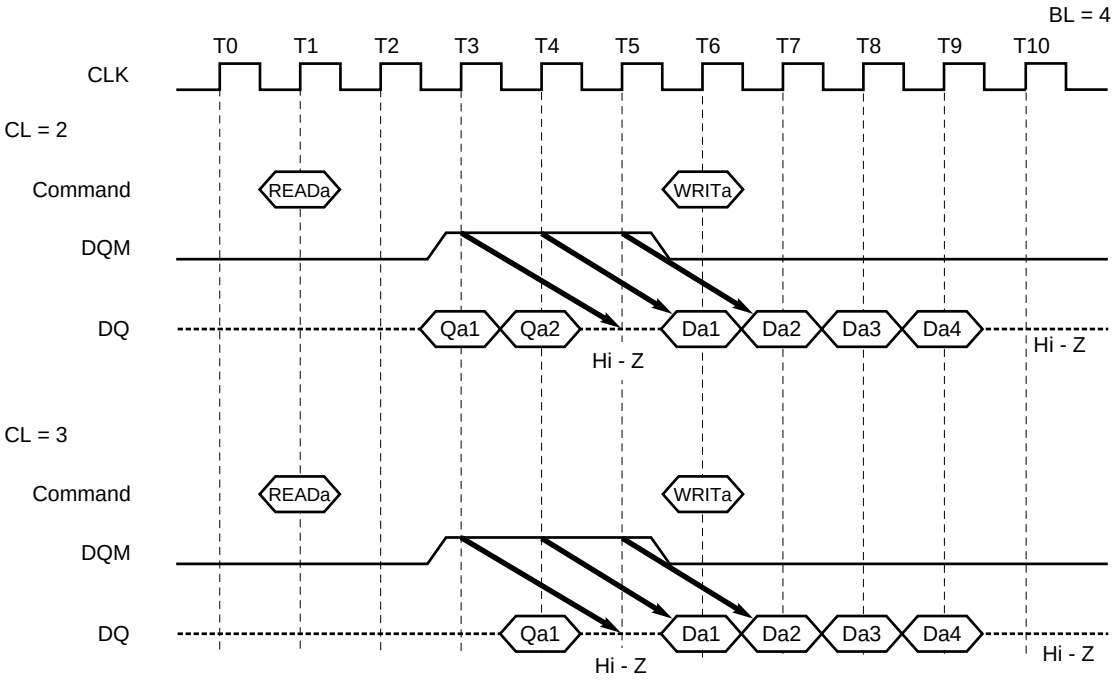


(2) Read cycle

The preceding burst read operation can be aborted and a burst write operation can be started by inputting a new write command in the read cycle. At this time, the data bus must be made to go into a high-impedance state by using DQM before inputting the write command, to avoid data collision. Therefore, make DQM high at least three clocks before input of the write command.

Caution The timing of interrupting data by inputting the write command differs depending on the SDRAM product. For details, refer to the Data Sheet of each model.

Figure 10-4. Read/Write Command

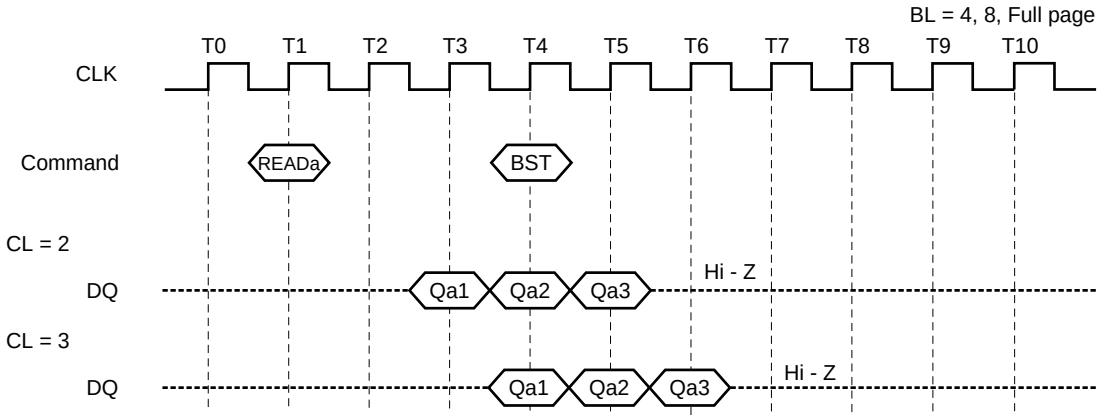


10.1.3 Ending burst operation by burst stop command

(1) Read cycle

The burst read operation can be aborted by inputting the burst stop command in the read cycle. The data bus goes into a high-impedance state after the lapse of the /CAS latency since the burst stop command (BST) has been input.

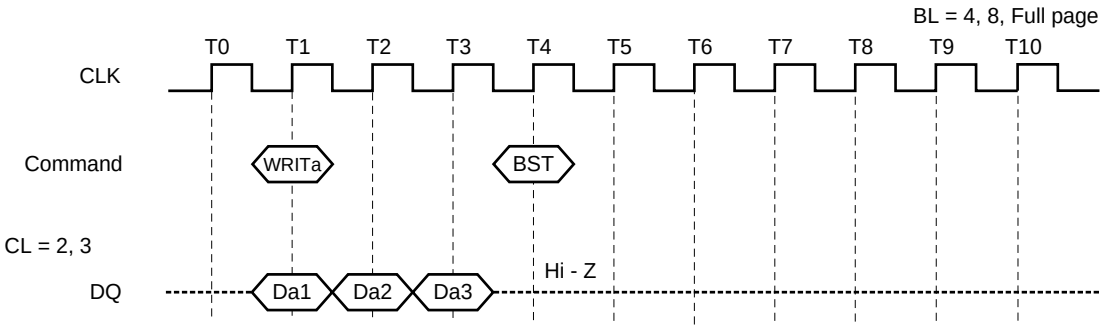
Figure 10-5. Read/Burst Stop Command



(2) Write cycle

The burst write operation ends when the burst stop command is input in the write cycle. The data bus goes into a high-impedance state as soon as the burst stop command (BST) has been input.

Figure 10-6. Write/Burst Stop Command



Remark Ending the burst operation by the burst stop command is valid for SDRAM, even when BL = 2.

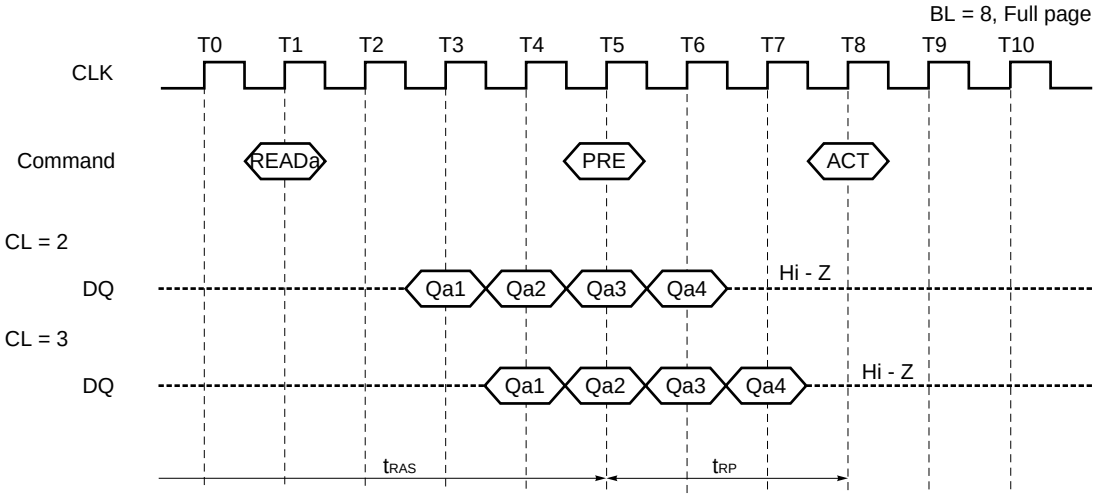
10.1.4 Ending burst operation by precharge command

(1) Read cycle

The burst read operation ends and the precharge operation is started by inputting the precharge command in the read cycle. The same bank can be made active again if t_{RP} has elapsed after input of the precharge command. To input the precharge command, t_{RAS} must be satisfied.

Caution The timing of ending the burst operation by the precharge command in the read cycle differs depending on the SDRAM product. For details, refer to the Data Sheet of each product.

Figure 10-7. Read/Precharge Command



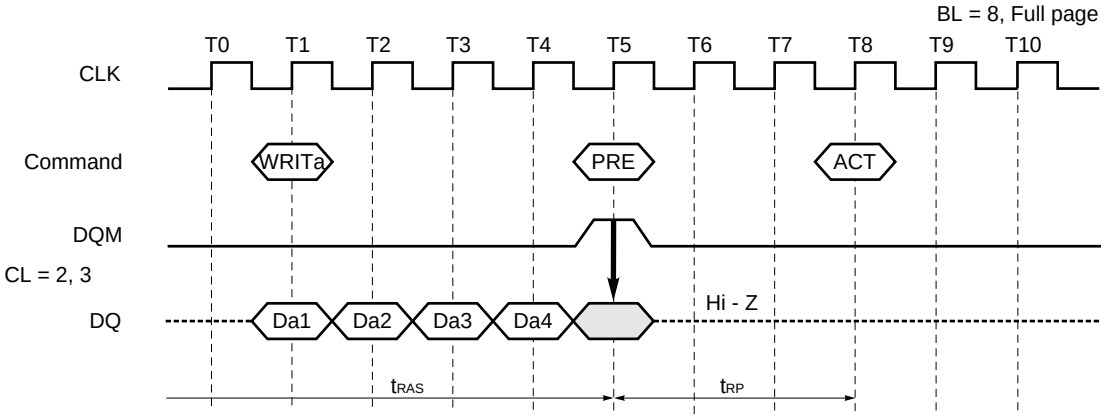
Remark Ending the burst operation by the precharge command is valid even when BL = 2 or 4.

(2) Write cycle

The burst write operation can be ended and precharge can be started by inputting the precharge command in the write cycle. The same bank can be made active again if t_{RP} has elapsed after input of the precharge command. To input the precharge command, t_{RAS} must be satisfied.

The data written before input of the precharge command is correctly written to memory cells. However, invalid data may be written as soon as the precharge command has been executed. To prevent this, make DQM high as soon as the precharge command has been input. This masks the invalid data.

Figure 10-8. Write/Precharge Command

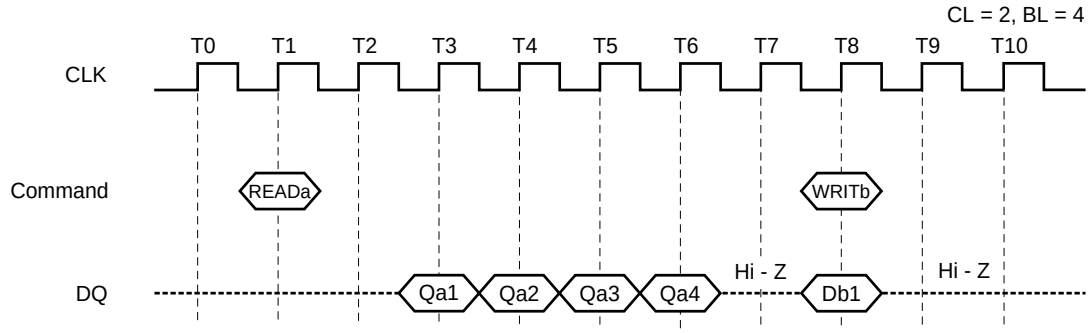


Remark Ending the burst operation by the precharge command is valid even when BL = 2 or 4.

10.2 Burst Read & Single Write

The SDRAM supports the burst read & single write mode. This mode can be used by setting any command to the mode register (refer to **Chapter 4 Basic Setting (Mode Register Setting)**). This is a function suitable for a write through cache system.

Figure 10-9. Burst Read & Single Write



CHAPTER 11 MULTIBANK OPERATION

Multibank operation is a form of operation control that makes efficient use of banks by controlling several banks independently.

This chapter describes the basic timing of multibank operation for three cases.

11.1 Basic Timing Types

11.1.1 Case when, during burst operation (read or write), burst operation (read or write) for a different bank is performed

In this case, there are the following types of operation transitions.

(1) Burst read operation -> Burst read operation
(2) Burst read operation -> Burst write operation
(3) Burst write operation -> Burst read operation
(4) Burst write operation -> Burst write operation

11.1.2 Case when, during burst operation (read or write), a different bank is activated and burst operation (read or write) for that bank is performed

In this case, there are the following types of operation transitions.

(1) Burst read operation -> Activation of different bank -> Burst read operation
(2) Burst read operation -> Activation of different bank -> Burst write operation
(3) Burst write operation -> Activation of different bank -> Burst read operation
(4) Burst write operation -> Activation of different bank -> Burst write operation

11.1.3 Case when, during auto precharge burst operation (read or write), burst operation (read or write) for a different bank is performed

In this case, there are the following types of operation transitions.

(1) Auto precharge burst read operation -> Burst read operation
(2) Auto precharge burst read operation -> Burst write operation
(3) Auto precharge burst write operation -> Burst read operation
(4) Auto precharge burst write operation -> Burst write operation

11.2 Case When, During Burst Operation (Read or Write), Burst Operation (Read or Write) for a Different Bank Is Performed

The timing examples described in this section are examples of multibank operation when burst operation (read or write) is performed for Bank B during execution of burst operation (read or write) for Bank A. Each of the following sub-sections shows two timing examples: the timing for the start of burst operation for Bank B following the end (completion) of burst operation to Bank A, and the timing for the start of burst operation for Bank B after burst operation for Bank A is interrupted.

The timing examples are for /CAS latency (CL) = 2, burst length (BL) = 4, with Bank A and Bank B both active. (An interval of time equal to or greater than tRDC(MIN.) has elapsed following input of the active command to both banks.)

Various other bank combinations in addition to the Bank A and Bank B combination described here are possible.

Current Operation (Bank A)		Next Operation (Bank B)	Timing Example
Burst read	Completed	Burst read	See **Fig. 11-1**
	Interrupted		See **Fig. 11-2**
Burst read	Completed	Burst write	See **Fig. 11-3**
	Interrupted		See **Fig. 11-4**
Burst write	Completed	Burst read	See **Fig. 11-5**
	Interrupted		See **Fig. 11-6**
Burst write	Completed	Burst write	See **Fig. 11-7**
	Interrupted		See **Fig. 11-8**

11.2.1 Burst read operation -> Burst read operation

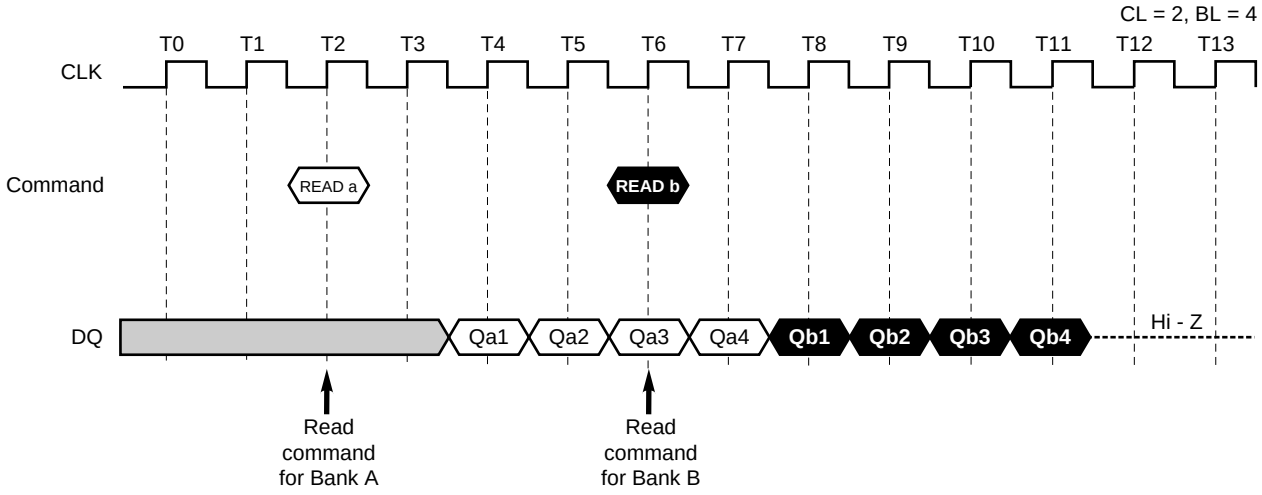
(1) Burst read operation (completion) -> Burst read operation

Figure 11-1 shows a timing example when the data of Bank B is newly output at the clock following the clock for which the last data has been output during burst read operation for Bank A.

If a read command (READa) for Bank A is input at timing T2 and a read command (READb) for Bank B is input at timing T6, data output by READa ends at T7 and data output by READb begins from T8.

Performing burst read operation for Bank B with this timing allows continuous output of burst data for the respective banks and effective use of the data bus.

Figure 11-1. Burst Read Operation (Completion) -> Burst Read Operation



(2) Burst read operation (Interruption) -> Burst read operation

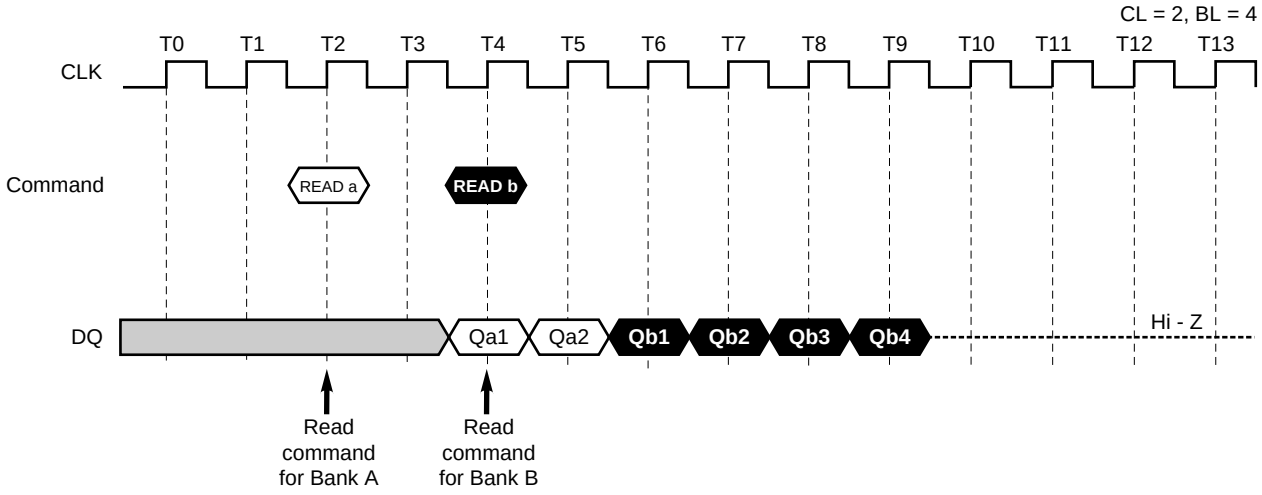
Figure 11-2 shows a timing example when the data of Bank B is newly output following interruption of data output during burst read operation for Bank A.

If a read command (READa) for Bank A is input at timing T2 and a read command (READb) for bank B is input at timing T4, data output by READa ends at T5 and data output by READb begins from T6.

When burst read operation for Bank B is performed during burst read operation for Bank A, the first burst read operation (burst read of Bank A) is interrupted, and burst read operation (burst read of Bank B) by the read command that is input next is given priority.

It is not possible to input a read command for Bank B prior to timing T3.

Figure 11-2. Burst Read Operation (Interruption) -> Burst Read Operation



11.2.2 Burst read operation -> Burst write operation

(1) Burst read operation (Completion) -> Burst write operation

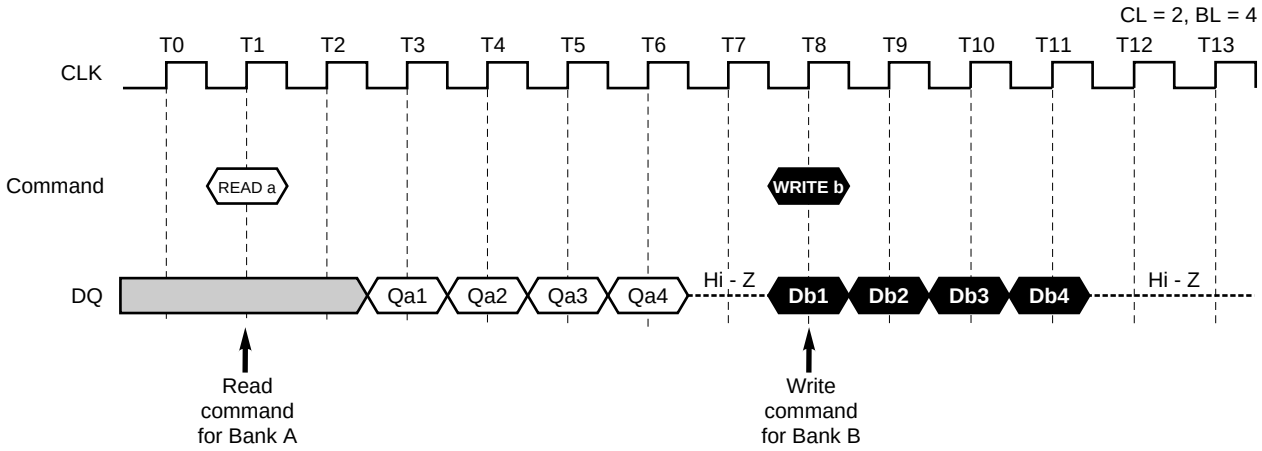
Figure 11-3 shows a timing example when write data is newly input to Bank B following output of the last data during burst read operation for Bank A.

If a read command (READa) is input for Bank A at timing T1 and a write command (WRITEb) is input for Bank B at timing T8, data output by READa ends (completes) at T6 and data input by WRITEb starts from T8.

Performing burst write operation for Bank B with this timing allows the most efficient data input and efficient use of the data bus.

To prevent data conflicts on the data bus, the state of the data bus must be high impedance 1 clock before the write command is input (T7).

Figure 11-3. Burst Read Operation (Completion) -> Burst Write Operation



(2) Burst read operation (Interruption) -> Burst write operation

Figure 11-4 shows a timing example when write data is newly input to Bank B after interrupting data output during burst read operation for Bank A.

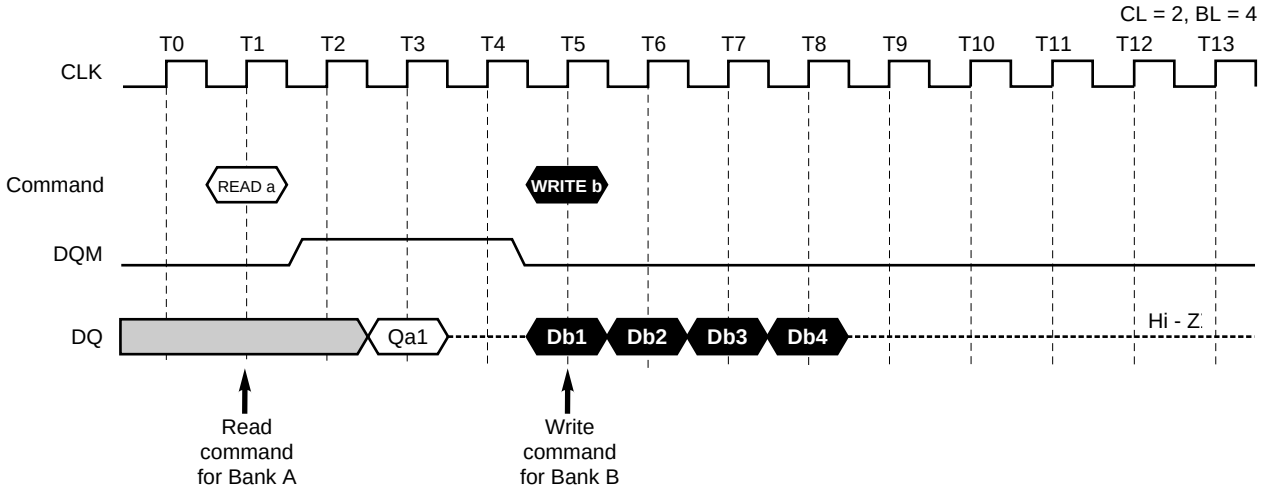
If a read command (READa) for Bank A is input at timing T1 and a write command (WRITEb) for bank B is input at timing T5, data output by READa ends at T3 and data input by WRITEb begins from T5.

To prevent data conflicts on the data bus, the state of the data bus must be made high impedance 1 clock before the write command is input (T4) by making DQM high level.

Since DQM latency for read is 2 clocks, DQM is made high level at T2 to mask data at T4. Also, as data is output at T5 and T6 since the burst length is 4, DQM is similarly made high level at T3 and T4 to mask output data.

The write command cannot be input for Bank B prior to timing T4.

Figure 11-4. Burst Read Operation (Interruption) -> Burst Write Operation



11.2.3 Burst write operation -> Burst read operation

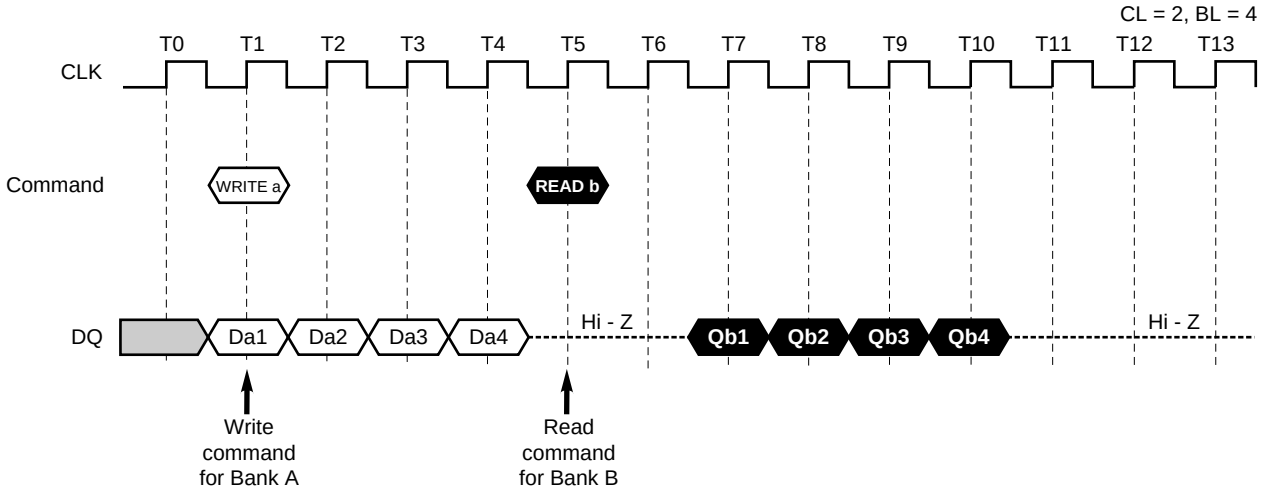
(1) Burst write operation (Completion) -> Burst read operation

Figure 11-5 shows a timing example when data is newly output to Bank B following input of the last data during burst write operation for Bank A.

If a write command (WRITEa) is input for Bank A at timing T1 and a read command (READb) is input for Bank B at timing T5, data input by WRITEa ends (completes) at T4 and data output by READb starts from T7.

Performing burst read operation for Bank B with this timing allows the most efficient data output and efficient use of the data bus.

Figure 11-5. Burst Write Operation (Completion) -> Burst Read Operation



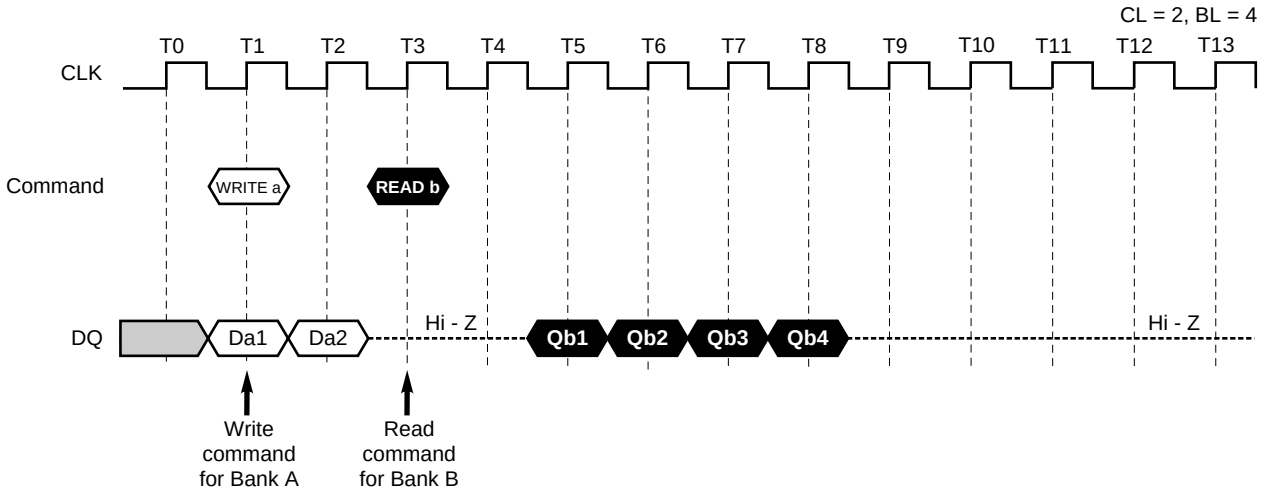
(2) Burst write operation (Interruption) -> Burst read operation

Figure 11-6 shows a timing example when read data is newly output to Bank B after data input is interrupted during burst write operation for Bank A.

If a write command (WRITEa) is input for Bank A at timing T1 and a read command (READb) is input for Bank B at timing T3, data input by WRITEa ends at T2 and data output by READb begins from T5. The burst write operation for Bank A at this time writes only the write data input prior to input of READb.

It is not possible to input a write command for Bank B prior to timing T2.

Figure 11-6. Burst Write Operation (Interruption) -> Burst Read Operation



11.2.4 Burst write operation -> Burst write operation

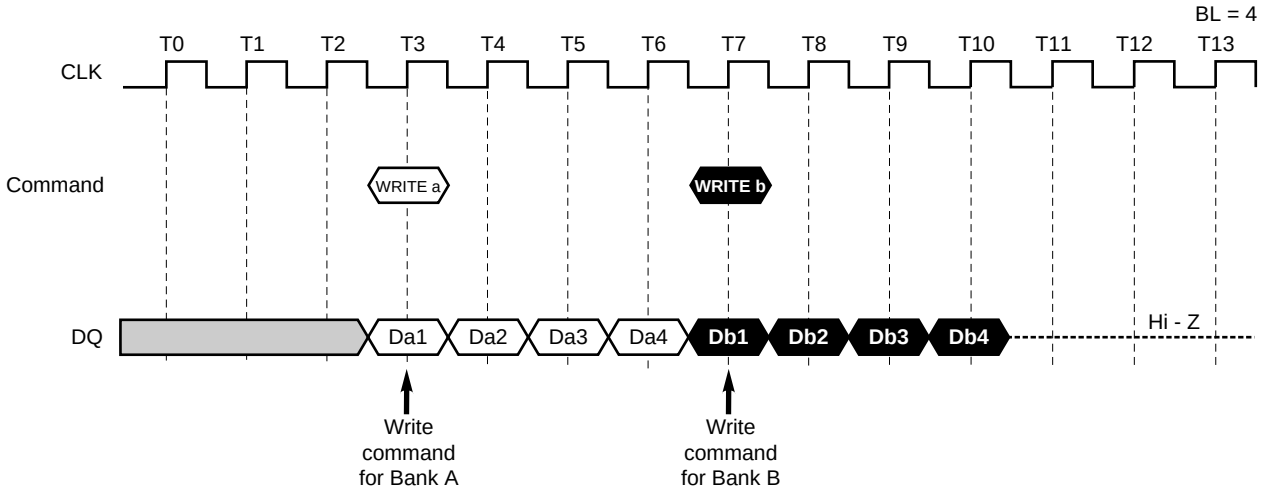
(1) Burst write operation (Completion) -> Burst write operation

Figure 11-7 shows a timing example when data is newly input to Bank B at the next clock following input of the last data during burst write operation for Bank A.

If a write command (WRITEa) for Bank A is input at timing T3 and a write command (WRITEb) for Bank B is input at timing T7, data input by WRITEa ends (completes) at T6 and data input by WRITEb begins from T7.

Performing burst write operation for Bank B with this timing allows continuous input of burst data to the respective banks and efficient user of the data bus.

Figure 11-7. Burst Write Operation (Completion) -> Burst Write Operation



(2) Burst write operation (Interruption) -> Burst write operation

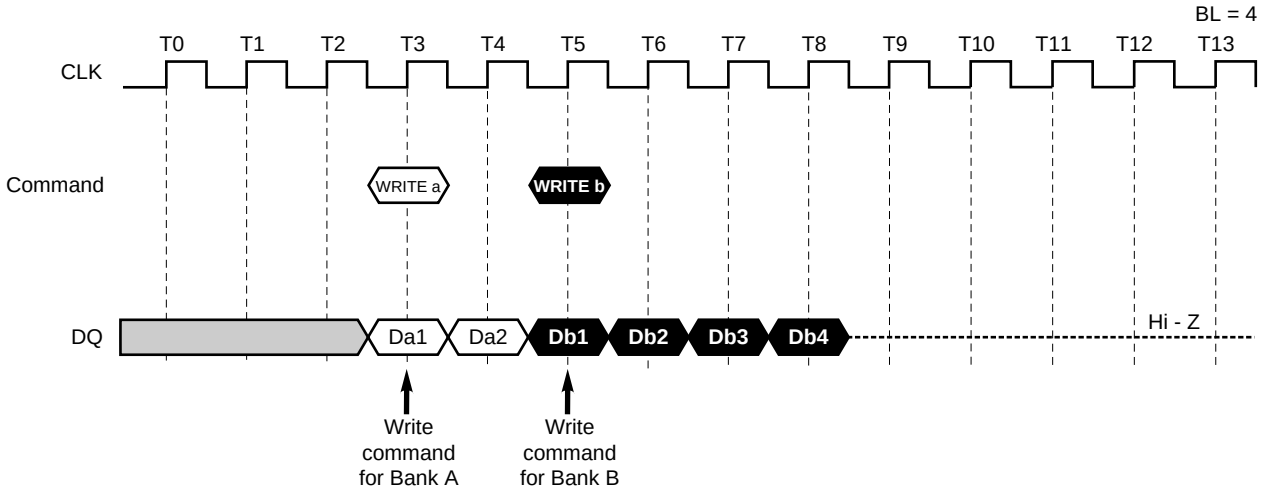
Figure 11-8 shows a timing example when data is newly input to Bank B following interruption of data input during burst write operation for Bank A.

If a write command (WRITEa) for Bank A is input at timing T3 and a write command (WRITEb) for Bank B is input at timing T5, data input by WRITEa ends at T4 and data input by WRITEb starts from T5.

When burst write operation for Bank B is performed during burst write operation for Bank A, the first burst write operation (burst write of Bank A) is interrupted, and burst write operation (burst write to Bank B) by the write command that is input next is given priority.

It is not possible to input a write command for Bank B prior to timing T4.

Figure 11-8. Burst Write Operation (Interruption) -> Burst Write Operation



11.3 Case When, During Burst Operation (Read or Write), a Different Bank Is Activated and Burst Operation (Read or Write) for That Bank Is Performed

The timing examples described in this section are examples of multibank operation when burst operation (read or write) is performed for Bank B following activation of Bank B using the active command during burst operation (read or write) for Bank A. Each of the following sub-sections show two timing examples: The timing for the start of burst operation for Bank B following the end (completion) of burst operation for Bank A and the activation of Bank B, and the timing for the start of burst operation for Bank B following the interruption of burst operation for Bank A and the activation of Bank B.

The timing examples are for /CAS latency (CL) = 2, burst length (BL) = 4, with Bank A active (An interval of time equal to or greater than tRDC(MIN.) has elapsed following input of the active command), and Bank B idle ($t_{RP(MIN.)}$ has elapsed following input of the precharge command).

Various other bank combinations in addition to the Bank A and Bank B combination described here are possible.

Current Operation (Bank A)		Next Operation (Bank B)	Last Operation (Bank B)	Timing Example
Burst read	Completed	Bank activation	Burst read	See **Fig. 11-9**
	Interrupted			See **Fig. 11-10**
Burst read	Completed		Burst write	See **Fig. 11-11**
	Interrupted			See **Fig. 11-12**
Burst write	Completed		Burst read	See **Fig. 11-13**
	Interrupted			See **Fig. 11-14**
Burst write	Completed		Burst write	See **Fig. 11-15**
	Interrupted			See **Fig. 11-16**

11.3.1 Burst read operation -> Activation of different bank -> Burst read operation

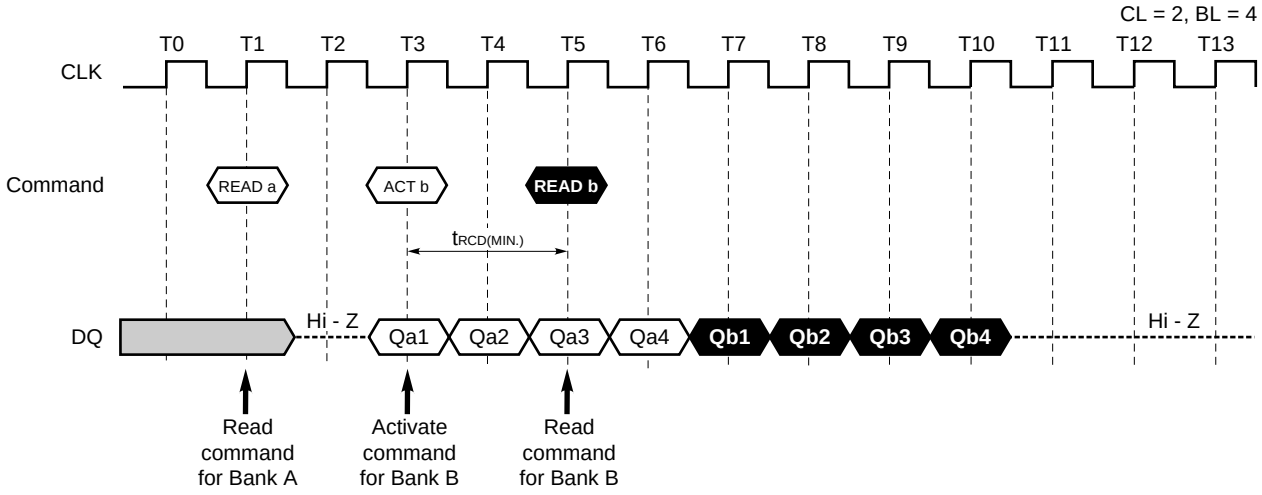
(1) Burst read operation (completion) -> Activation of different bank -> Burst read operation for that bank

Figure 11-9 shows a timing example when the data of Bank B is newly output following activation of Bank B from the idle state, at the next clock following output of the last data during burst read operation for Bank A.

If a read command (READb) for Bank B is input at timing T5 following input of a read command (READa) for Bank A at timing T1 and input of an active command (ACTb) for Bank B at timing T3 to activate Bank B, data output by READa ends (completes) at T6 and data output by READb begins from T7.

Performing burst read operation for Bank B with this timing allows continuous output for the respective banks and effective use of the data bus.

Figure 11-9. Burst Read Operation (Completion) -> Activation of Different Bank -> Burst Read Operation of That Bank



(2) Burst read operation (interruption) -> Activation of different bank -> Burst read operation for that bank

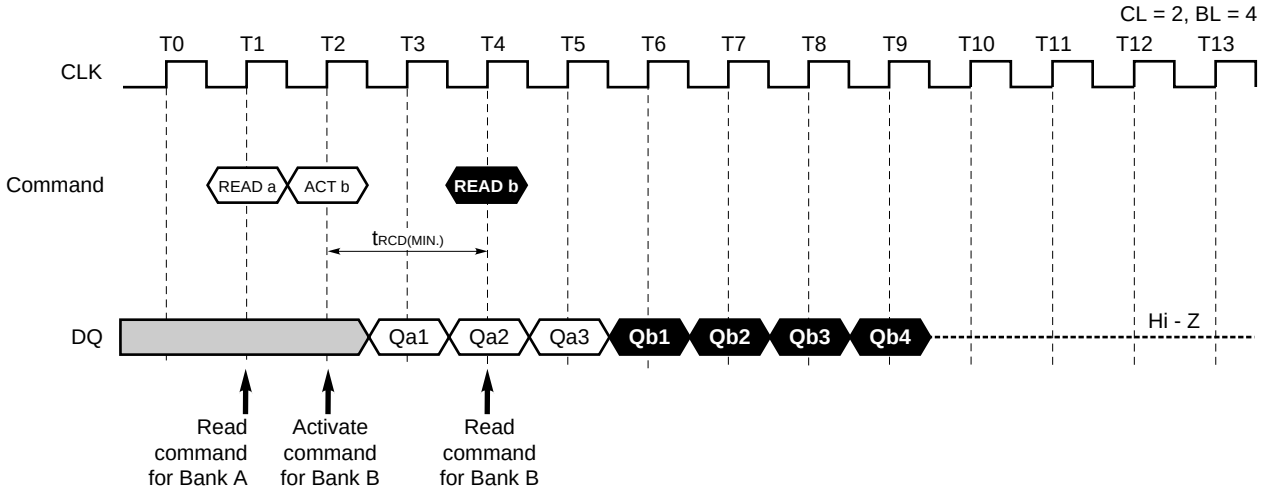
Figure 11-10 shows a timing example when the data of Bank B is newly output following interruption of data output during burst read operation of Bank A and activation of Bank B from the idle state.

If a read command (READb) for Bank B is input at timing T4 following input of a read command (READa) for Bank A at timing T1 and input of an active command (ACTb) for Bank B at timing T2 to activate Bank B, data output by READa ends at T5 and data output by READb begins from T6.

When burst read operation for Bank B is performed during burst read operation for Bank A, the first operation (burst read of Bank A) is interrupted and the burst read operation (burst read of Bank B) by the read command that is input next is given priority.

It is not possible to input a read command for Bank B prior to timing T3.

Figure 11-10. Burst Read Operation (Interruption) -> Activation of Different Bank -> Burst Read Operation for That Bank



11.3.2 Burst read operation -> Activation of different Bank -> Burst write operation

(1) Burst read operation (completion) -> Activation of different Bank -> Burst write operation for that bank

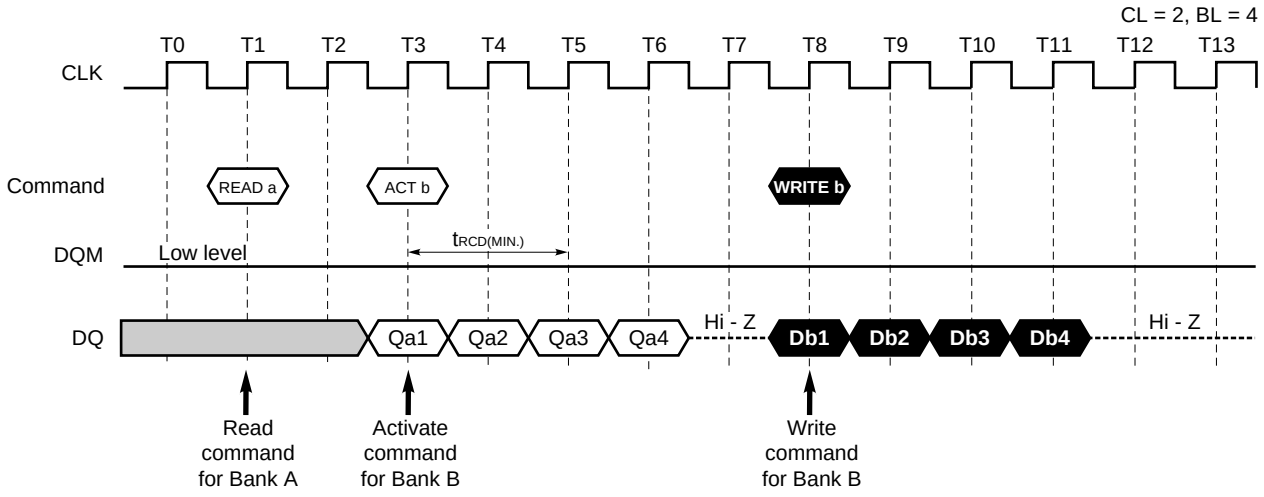
Figure 11-11 shows a timing example when Bank B, which was idle, is activated and the write data for Bank B is newly input following output of the last data during burst read operation for Bank A.

If a write command (WRITEb) for Bank B is input at timing T8 following input of a read command (READa) for Bank A at timing T1 and input of an active command (ACTb) for Bank B at timing T3 to activate Bank B, data output by READa ends (completes) at T6 and data input by WRITEb begins from T8.

Performing burst write operation for Bank B using this timing allows the most efficient data input and efficient use of the data bus.

To prevent data conflicts on the data bus, the state of the data bus must be high impedance 1 clock before the write command is input (T7).

Figure 11-11. Burst Read Operation (Completion) -> Activation of Different Bank -> Burst Write Operation for That Bank



(2) Burst read operation (interruption) -> activation of different bank -> Burst write operation for that bank

Figure 11-12 shows a timing example when write data is newly input to Bank B following interruption of data output during burst read operation for Bank A and activation of Bank B from the idle state.

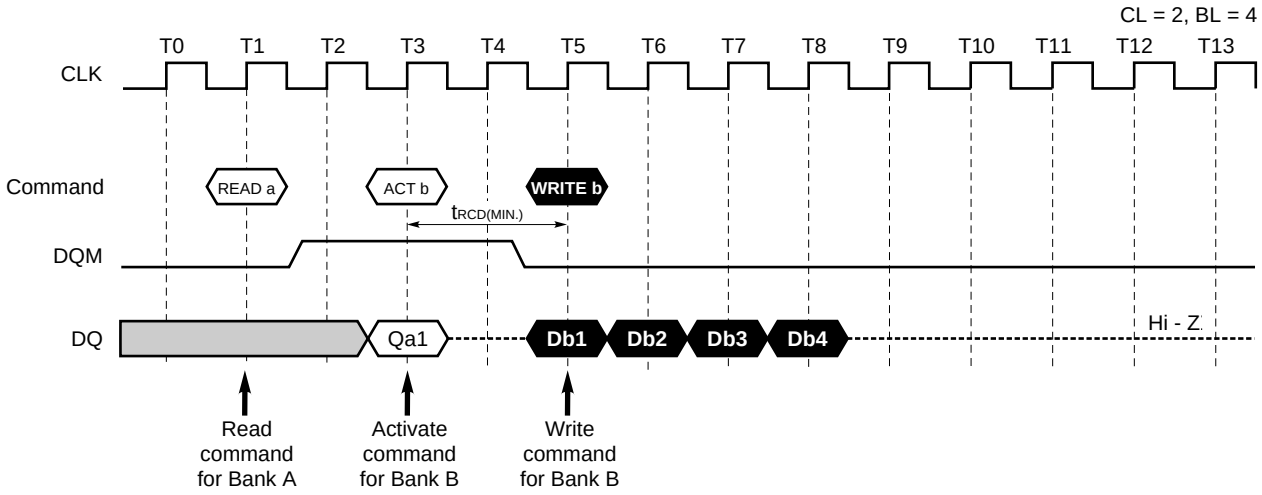
If a write command (WRITEb) for Bank B is input at timing T5 following input of a read command (READa) for Bank A and input of an active command (ACTb) for Bank B at timing T3 to activate Bank B, data output by READa ends at T3 and data input by WRITEb begins from T5.

To prevent data conflicts on the data bus, the state of the data bus must be made high impedance 1 clock before the write command is input (T4) by making DQM high level.

Since DQM latency for read is 2 clocks, DQM is made high level at T2 to mask data at T4. Also, as data is output at T5 and T6 since the burst length is 4, DQM is similarly made high level at T3 and T4 to mask output data.

The write command cannot be input for Bank B prior to timing T4.

Figure 11-12. Burst Read Operation (Interruption) -> Activation of Different Bank -> Burst Write Operation for That Bank



11.3.3 Burst write operation -> Activation of different bank -> Burst read operation

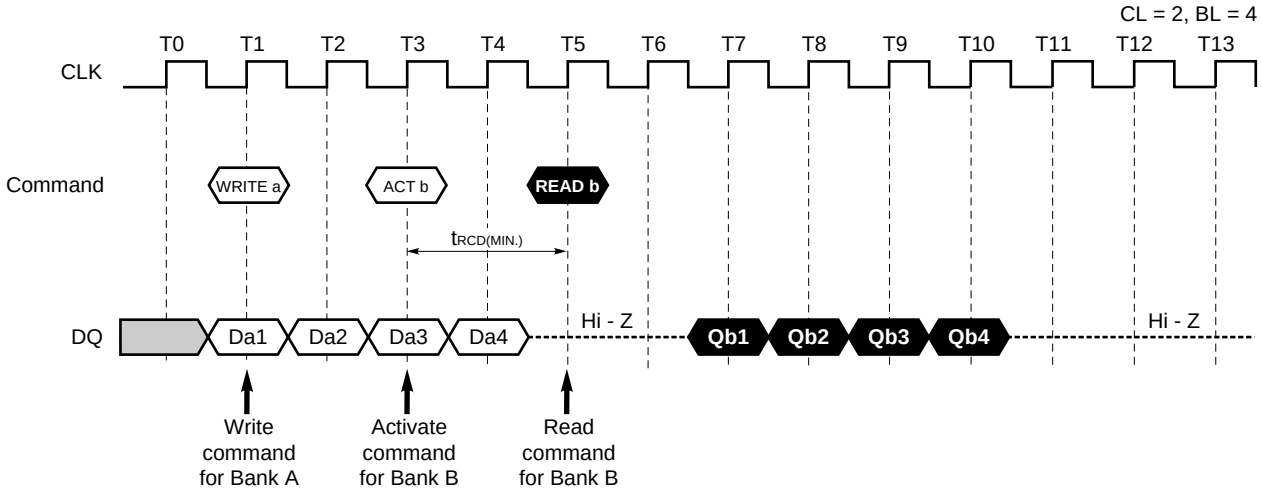
(1) Burst write operation (completion) -> Activation of different bank -> Burst read operation for that bank

Figure 11-13 shows a timing example when data is newly output to Bank B following input of the last data during burst write operation for Bank A and activation of Bank B from idle state.

If a read command (READb) is input for Bank B at timing T5 following input of a write command (WRITEa) for Bank A at timing T1 and input of an active command (ACTb) for Bank B at timing T3 to activate Bank B, data input by WRITEa ends (completes) at T4 and data output by READb begins from T7.

Performing burst read operation for Bank B using this timing allows the most efficient data output and efficient use of the data bus.

Figure 11-13. Burst Write Operation (Completion) -> Activation of Different Bank -> Burst Read Operation for That Bank



(2) Burst write operation (interruption) -> Activation of different bank -> Burst read operation for that bank

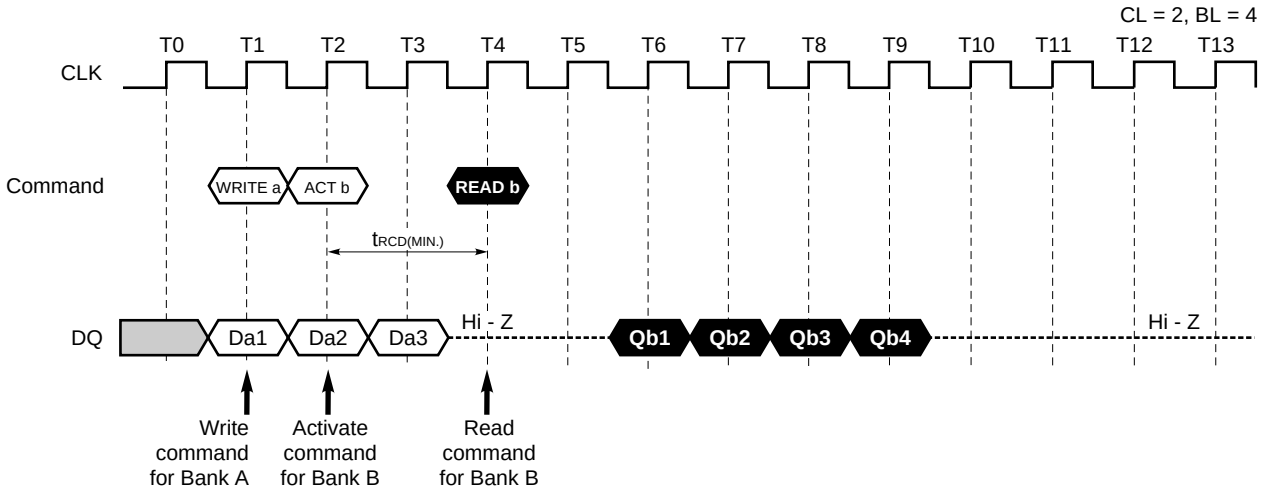
Figure 11-14 shows a timing example when read data of Bank B is newly output following interruption of data input during burst write operation for Bank A and activation of Bank B from idle state.

If a read command (READb) for Bank B is input at timing T4 following input of a write command (WRITEa) for Bank A at timing T1 and input of an active command (ACTb) for Bank B at timing T2 to activate Bank B, data input by WRITEa ends at T3 and data output by READb begins from T6.

The burst write operation for Bank A at this time writes only the write data that has been input prior to input of READb.

It is not possible to input a write command for Bank B prior to timing T3.

Figure 11-14. Burst Write Operation (Interruption) -> Activation of Different Bank -> Burst Read Operation for That Bank



11.3.4 Burst write operation -> Activation of different bank -> Burst write operation

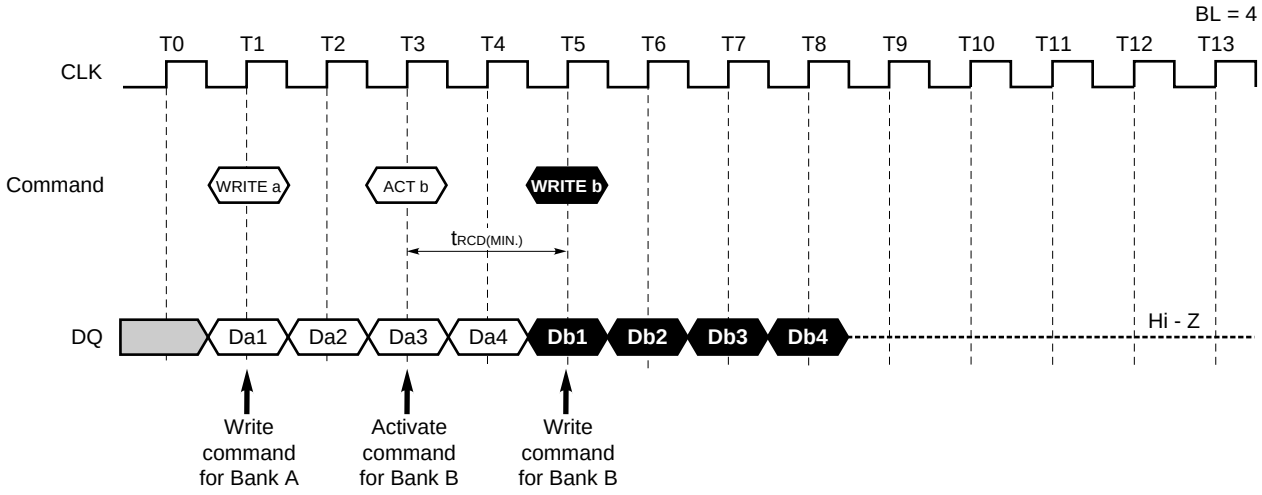
(1) Burst write operation (completion) -> Activation of different bank -> Burst write operation for that bank

Figure 11-15 shows a timing example when data is newly input to Bank B following activation of Bank B from idle state at the next clock following input of the last data during burst write operation for Bank A.

If a write command (WRITEb) is input for Bank B at timing T5 following input of a write command (WRITEa) for Bank A and input of an active command (ACTb) for Bank B at timing T3 to activate Bank B, data input by WRITEa ends (completes) at T4 and data input by WRITEb begins from T5.

Performing burst write to Bank B using this timing allows continuous input of burst data to the respective banks and efficient use of the data bus.

Figure 11-15. Burst Write Operation (Completion) -> Activation of Different Bank -> Burst Write Operation for That Bank



(2) Burst write operation (interruption) -> Activation of different bank -> Burst write operation for that bank

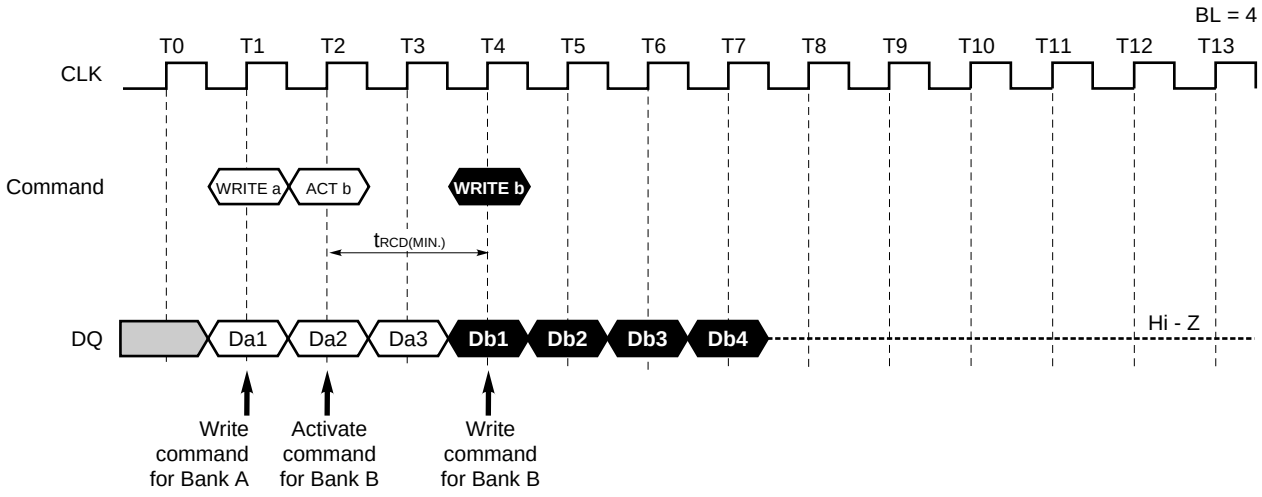
Figure 11-16 show a timing example when data is newly input to Bank B following interruption of data input during burst write operation for Bank A and activation of Bank B from idle state.

If a write command (WRITEb) for Bank B is input at timing T4 following input of a write command (WRITEa) for Bank A at timing T1 and input of an active command (ACTb) for Bank B at timing T2 to activate Bank B, data input by WRITEa ends at T3 and data input by WRITEb begins from T4.

When burst write operation for Bank B is performed during burst write operation for Bank A, the first burst write operation (burst write of Bank A) is interrupted and burst write operation (burst write of Bank B) by the write command that is input next is given priority.

It is not possible to input a write command for Bank B prior to timing T3.

Figure 11-16. Burst Write Operation (Interruption) -> Activation of Different Bank ->Burst Write Operation for That Bank



11.4 Case When, During Auto Precharge Burst Operation (Read or Write), Burst Operation (Read or Write) for a Different Bank Is Performed

The timing examples in this section are examples of multibank operation when burst operation (read or write) is performed for Bank B during execution of auto precharge burst operation (read or write) for Bank A. Each of the following sub-sections shows two timing examples: the timing for the start of burst operation for Bank B following the end (completion) of auto precharge burst operation for Bank A, and the timing for the start of burst operation for Bank B after auto precharge burst operation for Bank A is interrupted.

The timing examples are for /CAS latency (CL) = 2, burst length (BL) = 4, with Bank A and Bank B both active. (An interval of time equal to or greater than $t_{RDC(MIN.)}$ has elapsed following input of the active command to both banks.)

Various other bank combinations in addition to the Bank A and Bank B combination described here are possible.

Current Operation (Bank A)		Next Operation (Bank B)	Timing Example
Burst read	Completed	Burst read	See **Fig. 11-17**
With Auto precharge	Interrupted		See **Fig. 11-18**
Burst read	Completed	Burst write	See **Fig. 11-19**
With Auto precharge	Interrupted		See **Fig. 11-20**
Burst write	Completed	Burst read	See **Fig. 11-21**
With Auto precharge	Interrupted		See **Fig. 11-22**
Burst write	Completed	Burst write	See **Fig. 11-23**
With Auto precharge	Interrupted		See **Fig. 11-24**

11.4.1 Auto precharge burst read operation -> Burst read operation

(1) Auto precharge burst read operation (completion) -> Burst read operation

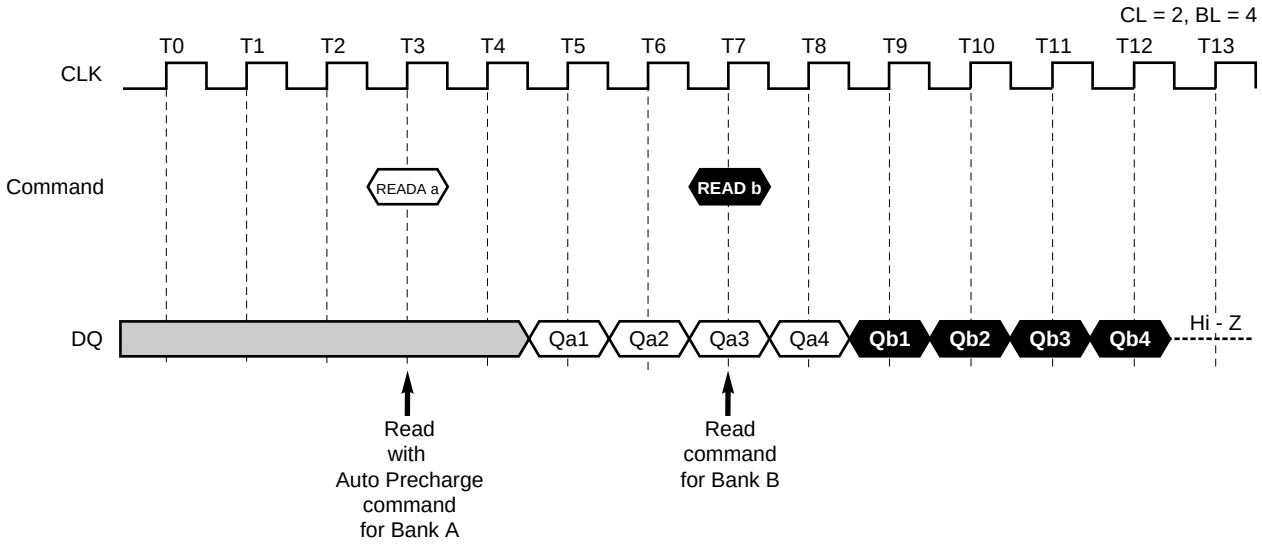
Figure 11-17 shows a timing example when data is newly output from Bank B at the next clock following output of the last data during auto precharge burst read operation for Bank A.

If a read command (READb) is input for Bank B at timing T7 following input of an auto precharge read command (READAa) for Bank A at timing T3, data output by READAa ends (completes) at T8 and data output by READb begins from T9.

Performing burst read operation for Bank B using this timing allows continuous output of the burst data of the respective banks and efficient use of the data bus.

Auto precharge operation for Bank A begins at timing T7.

Figure 11-17. Auto Precharge Burst Read Operation (Completion) -> Burst Read Operation



(2) Auto precharge burst read operation (interruption) -> Burst read operation

Figure 11-18 shows a timing example when data is newly output from Bank B following interruption of data output during auto precharge burst read operation for Bank A.

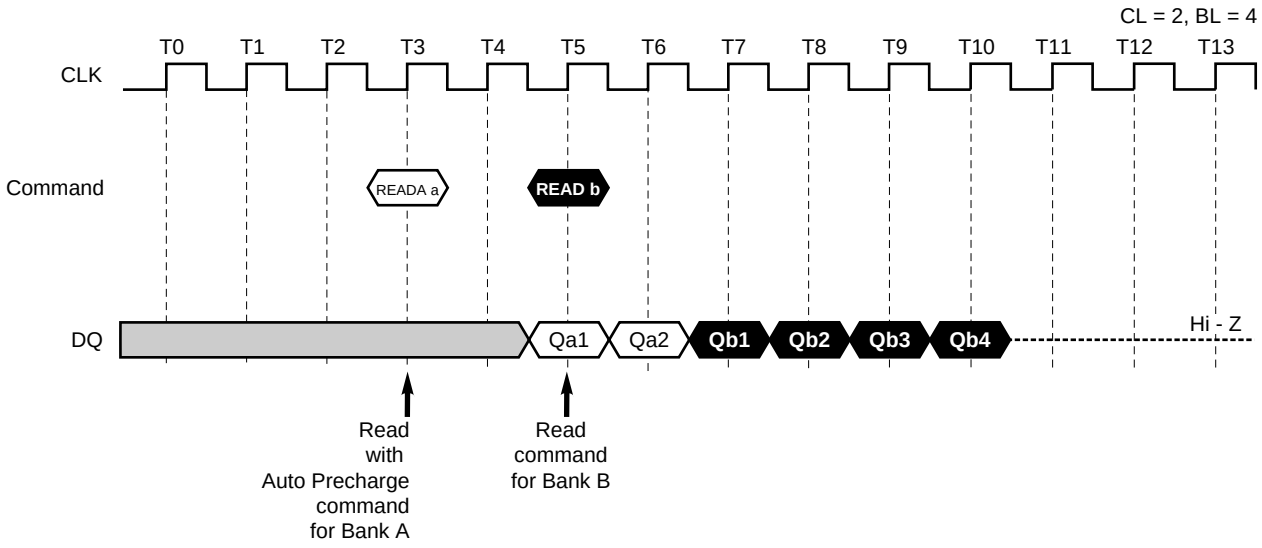
If a read command (READb) for Bank B is input at timing T5 following input of an auto precharge read command (READAa) for Bank A at timing T3, data output by READAa ends at T6 and data output by READb begins from T7.

When a burst read operation for Bank B is performed during auto precharge burst read operation for Bank A, the first burst read operation (auto precharge burst read of Bank A) is interrupted and burst read operation (burst read of Bank B) by the read command that is input next is given priority.

However, the auto precharge function for Bank A is effective even if the read cycle is interrupted, and when a read command for Bank B is input at T5, precharging of Bank A begins immediately and after t_{RP}+1 cycles, taking T5 as reference, Bank A goes in the idle state.

It is not possible to input a read command for Bank B before timing T4.

Figure 11-18. Auto Precharge Burst Read Operation (Interruption) -> Burst Read Operation



11.4.2 Auto precharge burst read operation -> Burst write operation

(1) Auto precharge burst read operation (completion) -> Burst write operation

Figure 11-19 shows a timing example when write data is input to Bank B following output of the last data during auto precharge burst read operation for Bank A.

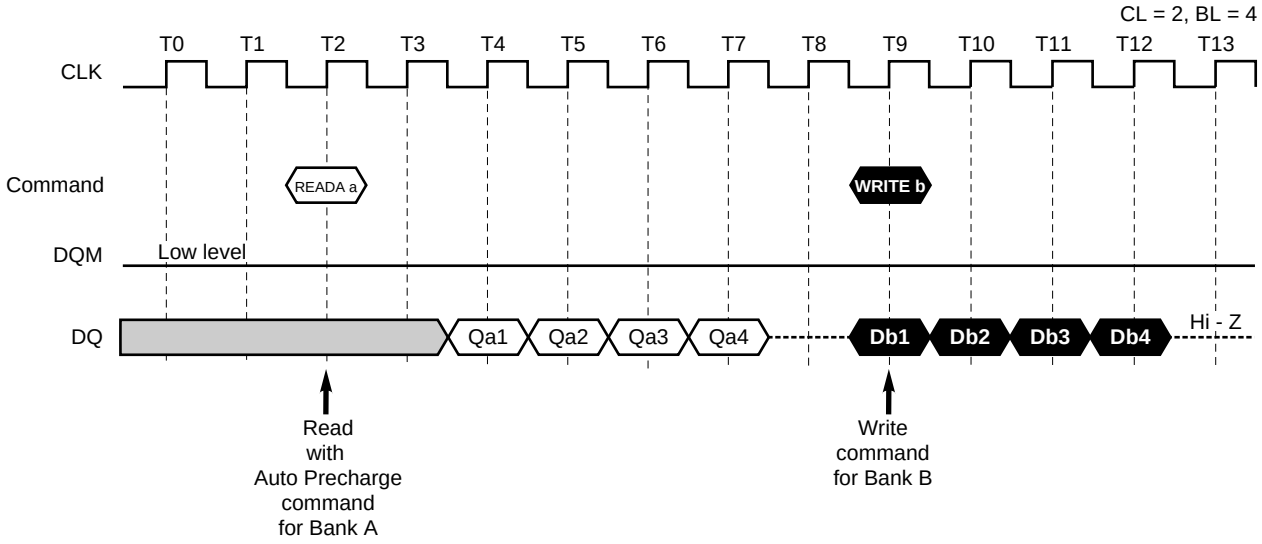
If a auto precharge read command for Bank A (READAa) is input at timing T2 and a write command for Bank B (WRITEb) is input at timing T9, data output by READAa ends (completes) at T7 and data input by WRITEb begins from T9.

Performing burst write operation for Bank B using this timing allows the most efficient data input and efficient use of the data bus.

To prevent data conflicts on the data bus, the state of the data bus must be high impedance 1 clock before the write command is input (T7).

Auto precharge operation for Bank A begins from timing T7.

Figure 11-19. Auto Precharge Burst Read Operation (Completion) -> Burst Write Operation



(2) Auto precharge burst read operation (interruption) -> Burst write operation

Figure 11-20 shows a timing example when write data is newly input to Bank B after interrupting data output during auto precharge burst read operation for Bank A.

If an auto precharge read command (READAa) is input for Bank A at timing T3 and a write command (WRITEb) is input for Bank B at timing T7, data output by READAa ends at T5 and data input by WRITEb begins from T7.

To prevent data conflicts on the data bus, the state of the data bus must be made high impedance 1 clock before the write command is input (T6) by making DQM high level.

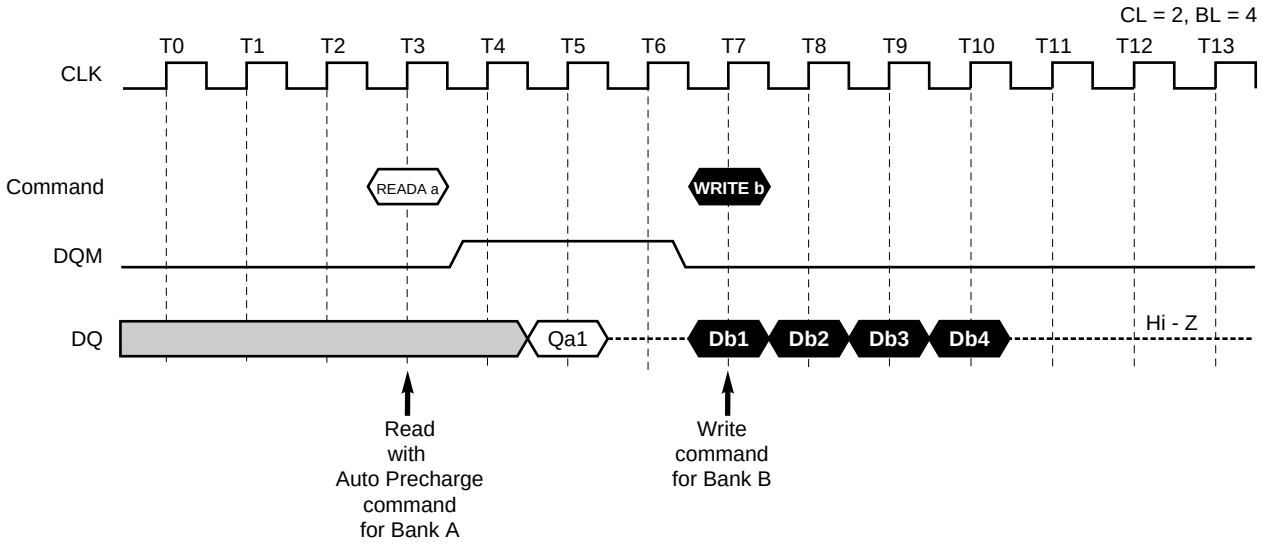
Since DQM latency for read is 2 clocks, DQM is made high level at T4 to mask data at T6. Also, as data is output at T7 and T8 since the burst length is 4, DQM is similarly made high level at T5 and T6 to mask output data.

The write command cannot be input for Bank B prior to timing T4.

However, the auto precharge function for Bank A is effective even if the read cycle is interrupted, and when a read command for Bank B is input at T7, precharging of Bank A begins immediately and after t_{RP}+1 cycles, taking T7 as reference, Bank A goes in the idle state.

It is not possible to input a write command for Bank B before timing T6.

Figure 11-20. Auto Precharge Burst Read Operation (Interruption) -> Burst Write Operation



11.4.3 Auto precharge burst write operation -> Burst read operation

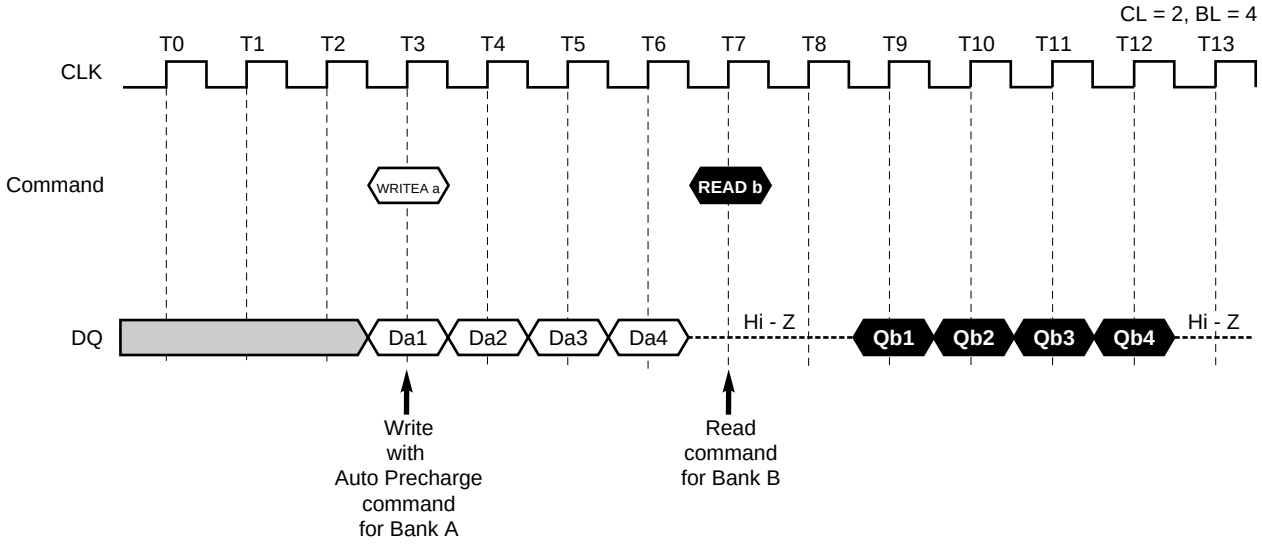
(1) Auto precharge burst write operation (completion) -> Burst read operation

Figure 11-21 shows a timing example when data of Bank B is newly output following input of the last data during auto precharge burst write operation for Bank A.

If an auto precharge write command (WRITEAa) is input for Bank A at timing T3 and a read command (READb) is input for Bank B, data input by WRITEAa ends (completes) at T6 and data output by READb begins from T9.

Performing burst read operation for Bank B using this timing allows the most efficient data output and efficient use of the data bus.

Figure 11-21. Auto Precharge Burst Write Operation (Completion) -> Burst Read Operation



(2) Auto precharge burst write operation (interruption) -> Burst read operation

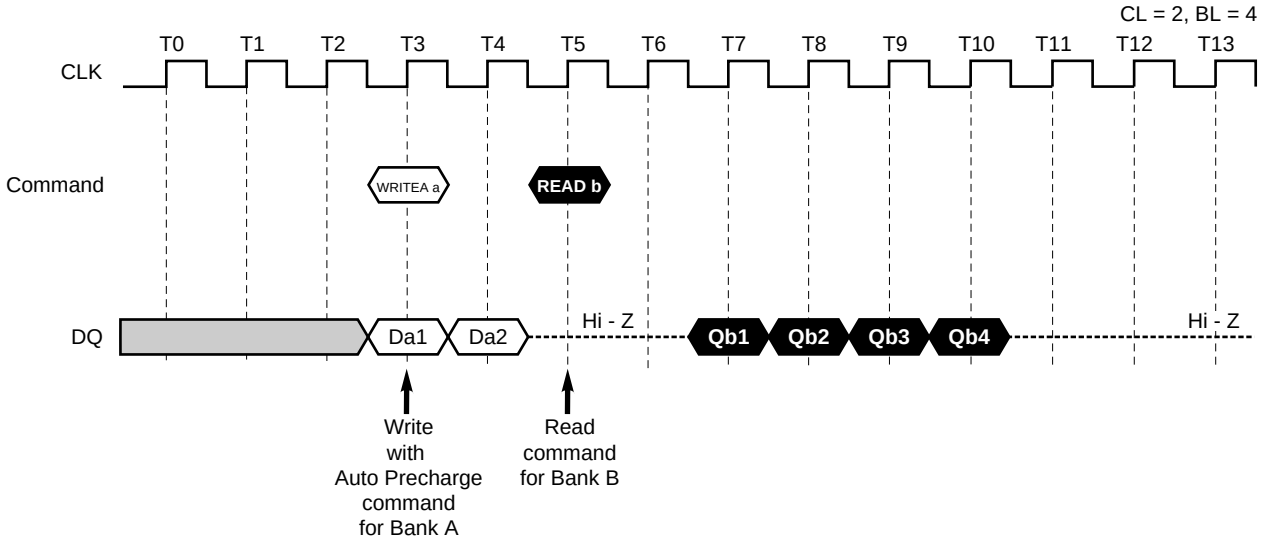
Figure 11-22 shows a timing example when read data is newly output from Bank B following interruption of data input during auto precharge burst write operation for Bank A.

If an auto precharge write command (WRITEAa) is input for Bank A at timing T3 and a read command (READb) is input for Bank B at timing T5, data input by WRITEAa ends at T4 and data output by READb begins from T7. The burst write operation for Bank A at this time writes only the write data input prior to input of READb.

However, the auto precharge function for Bank A is effective even if the write cycle is interrupted, and when a read command for Bank B is input at T5, precharging of Bank A begins immediately and after $t_{DAL(MIN.)}$+1 clock (in IntelTM specs, after $t_{DAL(MIN.)}$+2 clocks), taking T5 as reference, Bank A goes in the idle state.

It is not possible to input a read command for Bank B before timing T4.

Figure 11-22. Auto Precharge Burst Write Operation (Interruption) -> Burst Read Operation



11.4.4 Auto precharge burst write operation -> Burst write operation

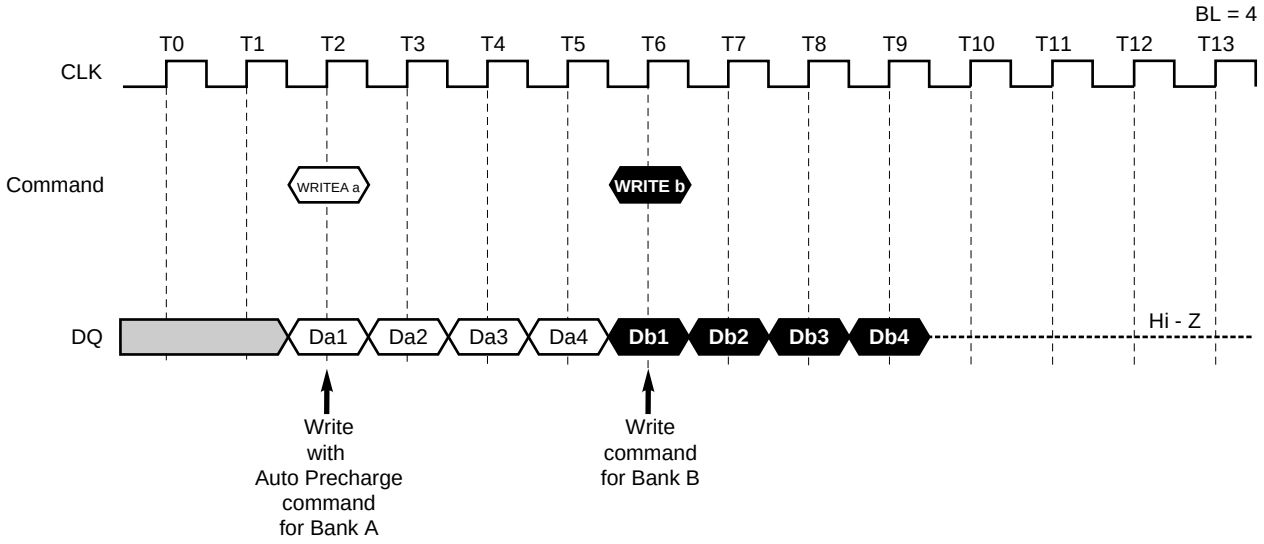
(1) Auto precharge burst write operation (completion) -> Burst write operation

Figure 11-23 shows a timing example when data is newly input to Bank B at the next clock following input of the last data during auto precharge burst write operation for Bank A.

If an auto precharge write command (WRITEAa) is input for Bank A at timing T2 and a write command (WRITEb) for Bank B is input at timing T6, data input by WRITEAa ends (completes) at T5 and data input by WRITEb begins from T6.

Performing burst write operation for Bank B using this timing allows continuous input of burst data to the respective banks and efficient use of the data bus.

Figure 11-23. Auto Precharge Burst Write Operation (Completion) -> Burst Write Operation



(2) Auto precharge burst write operation (interruption) -> Burst write operation

Figure 11-24 shows a timing example when data is newly input to Bank B following interruption of data input during auto precharge burst write operation for Bank A.

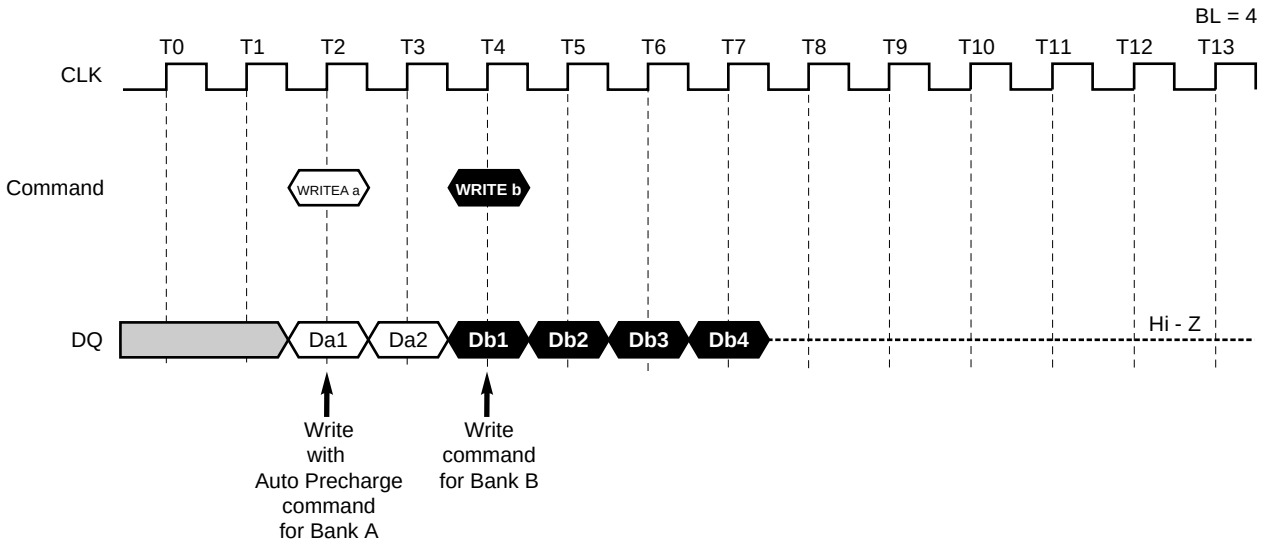
If an auto precharge write command (WRITEAa) is input for Bank A at timing T2 and a write command (WRITEb) is input for Bank B at timing T4, data input by WRITEAa ends at T3 and data input by WRITEb begins from T4.

When burst write operation for Bank B is performed during burst write operation for Bank A, the first burst write operation (auto precharge burst write for Bank A) is interrupted, and the burst write operation (burst write for Bank B) by the write command that is input next is given priority.

However, the auto precharge function for Bank A is effective even if the write cycle is interrupted, and when a read command for Bank B is input at T4, precharging of Bank A begins immediately and after $t_{DAL(MIN.)}$+1 clock (in IntelTM specs, after $t_{DAL(MIN.)}$+2 clocks), taking T4 as reference, Bank A goes in the idle state.

It is not possible to input a read command for Bank B before timing T3.

Figure 11-24 Auto Precharge Burst Write Operation (Interruption) -> Burst Write Operation



CHAPTER 12 CALCULATION OF CURRENT CONSUMPTION

This chapter shows expressions to calculate the average current consumption and actually calculates the average current consumption when a single bank of the 128M SDRAM (μPD45128841-A10) operates and when all the banks operate (i.e., when the two banks alternately operate incessantly), based on the DC characteristic values of the SDRAM.

12.1 Calculation Expression

The average current consumption for single bank operation can be calculated by the expression shown in Figure 12-1, and the average current consumption for all bank operation can be calculated by the expression in Figure 12-2.

12.1.1 Parameters for calculating current consumption

The following parameters are necessary for calculating the average current consumption.

t_{CK}	:	Clock cycle (ns)
t_{RAS}	:	/RAS active time (ns)
t_{RP}	:	/RAS precharge time (ns)
Precharge standby current	:	$I_{CC}2N$ (mA)
Active standby current	:	$I_{CC}3N$ (mA)
Operating current	:	$I_{CC}1$ (mA)
Burst mode current	:	$I_{CC}4$ (mA)
Refresh current	:	$I_{CC}5$ (mA)
Self-refresh current	:	$I_{CC}6$ (mA)
AN	:	Number of single-bank row active commands
BN	:	Number of single-bank burst data transfers
CN	:	Number of dual-bank row active commands
DN	:	Number of dual-bank burst data transfers
RN	:	Number of refresh commands
T	:	Combined total elapsed time of operation
I	:	Average calculated current (mA)

12.1.2 Expression for single bank operation

The average current consumption for the single bank operation can be calculated by the expression below.

Figure 12-1. Expression for Single Bank Operation

SINGLE BANK

$$
\begin{aligned}
I\ (mA) = [\ & I_{CC2N} \times \{ t_{RP} - (AN + RN) \times t_{RP(MIN.)} \} \times t_{CK(MIN.)} / t_{CK} \\
& + I_{CC1} \times AN \times (t_{RAS(MIN.)} + t_{RP(MIN.)}) \\
& + I_{CC4} \times (BN - AN) \times t_{CK(MIN.)} \\
& + I_{CC5} \times RN \times (t_{RAS(MIN.)} + t_{RP(MIN.)}) \\
& + I_{CC3N} \times \{ t_{RAS} - (AN + RN) \times t_{RAS(MIN.)} - (BN - AN) \times t_{CK(MIN.)} \} \times t_{CK(MIN.)} / t_{CK}] / T
\end{aligned}
$$

12.1.3 Expression for two bank operation

The average current consumption for the two bank operation can be calculated by the expression below.

Figure 12-2. Expression for Two Bank Operation

DUAL BANK

$$
\begin{aligned}
I\ (mA) = [\ & I_{CC1} \times CN \times (t_{RAS(MIN.)} + t_{RP(MIN.)}) \\
& + I_{CC4} \times (DN - CN) \times t_{CK(MIN.)} \\
& + I_{CC5} \times RN \times (t_{RAS(MIN.)} + t_{RP(MIN.)}) \\
& + I_{CC3N} \times \{ t_{RAS} - (CN + RN) \times t_{RAS(MIN.)} - (DN - CN) \times t_{CK(MIN.)} \} \times t_{CK(MIN.)} / t_{CK}] / T
\end{aligned}
$$

Caution The actual current consumption changes depending on the design conditions and must be compensated for in accordance with the operating conditions of the device.

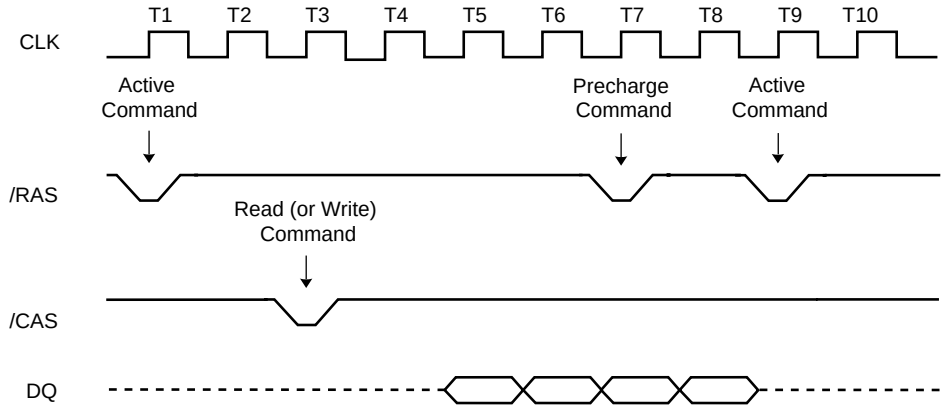
12.2 Average Current Consumption of 128M SDRAM (μPD45128841)

This section calculates the average current consumption of a 128M SDRAM (μPD45128821) when a single bank operates and when the two banks operate (i.e., when the two banks alternately operate incessantly), based on the DC characteristics of the SDRAM.

12.2.1 Example of average current consumption for single bank operation 1

The average current consumption for single bank operation is illustrated below.

Figure 12-3. Example of Average Current Consumption for Single Bank Operation 1



DC Characteristics (-A10)

Parameter		x4	x8	x16
Icc1	CL=3	100	100	100
	CL=2	100	100	110
Icc2P		1		
Icc2PS		0.5		
Icc2N		20		
Icc2NS		8		
Icc3P		5		
Icc3PS		4		
Icc3N		30		
Icc3NS		20		
Icc4	CL=3	85	95	110
	CL=2	110	125	140
Icc5	CL=3	220	220	220
	CL=2	220	220	220

System Parameters

CL = 2
BL = 4
Organization = x 8 bits
t_{CK2} = 10 ns
Constant access with 4 K / 64 ms Auto-Refresh
CKE = high level
/CS = low level

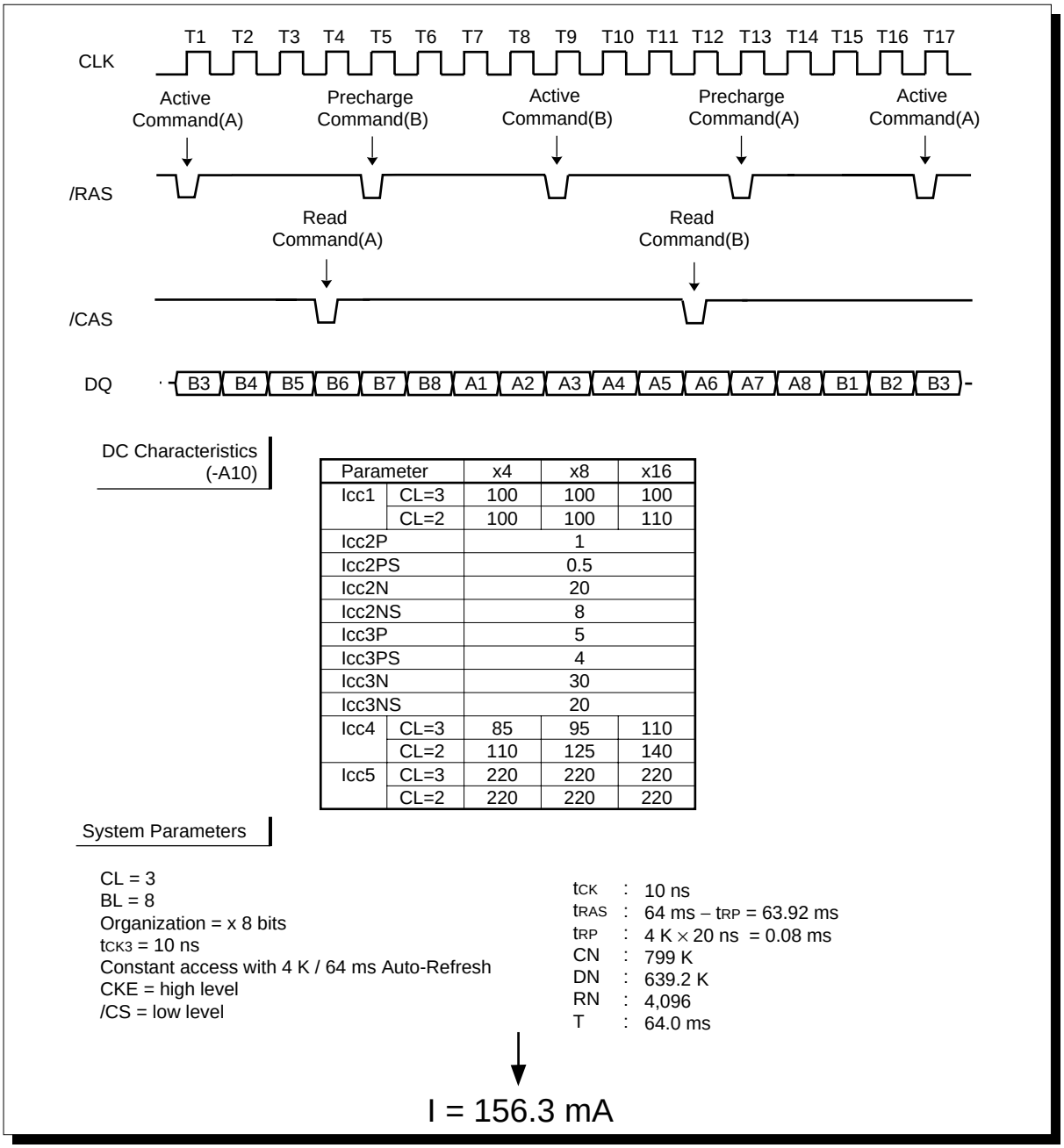
t_{CK} : 10 ns
t_{RAS} : AN × 70 ns + RN × $t_{RAS(MIN.)}$ = 63.48 ms
t_{RP} : 64.0 ms - 63.48 ms = 0.52 ms
AN : 909,721
BN : 3,638,884
RN : 4,096
T : 64 ms

I = 131.2 mA

12.2.2 Example of average current consumption for two-bank operation

The average current consumption for two-bank operation is illustrated below.

Figure 12-4. Example of Average Current Consumption for Two-Bank Operation



DC Characteristics (-A10)

Parameter		x4	x8	x16
Icc1	CL=3	100	100	100
	CL=2	100	100	110
Icc2P		1		
Icc2PS		0.5		
Icc2N		20		
Icc2NS		8		
Icc3P		5		
Icc3PS		4		
Icc3N		30		
Icc3NS		20		
Icc4	CL=3	85	95	110
	CL=2	110	125	140
Icc5	CL=3	220	220	220
	CL=2	220	220	220

System Parameters

CL = 3
BL = 8
Organization = x 8 bits
t_{CK3} = 10 ns
Constant access with 4 K / 64 ms Auto-Refresh
CKE = high level
/CS = low level

t_{CK} : 10 ns
t_{RAS} : 64 ms – t_{RP} = 63.92 ms
t_{RP} : 4 K × 20 ns = 0.08 ms
CN : 799 K
DN : 639.2 K
RN : 4,096
T : 64.0 ms

I = 156.3 mA