



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



# IS42S32200L IS45S32200L



## 512K Bits x 32 Bits x 4 Banks (64-MBIT) SYNCHRONOUS DYNAMIC RAM

OCTOBER 2012

### FEATURES

- Clock frequency: 200, 166, 143, 133 MHz
- Fully synchronous; all signals referenced to a positive clock edge
- Internal bank for hiding row access/precharge
- Single 3.3V power supply
- LVTTTL interface
- Programmable burst length: (1, 2, 4, 8, full page)
- Programmable burst sequence: Sequential/Interleave
- Self refresh modes
- 4096 refresh cycles every 16ms (A2 grade) or 64ms (Commercial, Industrial, A1 grade)
- Random column address every clock cycle
- Programmable  $\overline{\text{CAS}}$  latency (2, 3 clocks)
- Burst read/write and burst read/single write operations capability
- Burst termination by burst stop and precharge command

### OPTIONS

- Packages:
  - 86-pin TSOP-II
  - 90-ball TF-BGA
- Operating temperature range:
  - Commercial (0°C to + 70°C)
  - Industrial (-40°C to + 85°C)
  - Automotive Grade, A1 (-40°C to + 85°C)
  - Automotive Grade, A2: (-40°C to +105°C)

### OVERVIEW

ISSI's 64Mb Synchronous DRAM IS42/45S32200L is organized as 524,288 bits x 32-bit x 4-bank for improved performance. The synchronous DRAMs achieve high-speed data transfer using pipeline architecture. All inputs and outputs signals refer to the rising edge of the clock input.

### KEY TIMING PARAMETERS

Parameter	-5	-6	-7	Unit
Clk Cycle Time				
$\overline{\text{CAS}}$ Latency = 3	5	6	7	ns
$\overline{\text{CAS}}$ Latency = 2	7.5	7.5	7.5	ns
Clk Frequency				
$\overline{\text{CAS}}$ Latency = 3	200	166	143	Mhz
$\overline{\text{CAS}}$ Latency = 2	133	133	133	Mhz
Access Time from Clock				
$\overline{\text{CAS}}$ Latency = 3	4.8	5.4	5.4	ns
$\overline{\text{CAS}}$ Latency = 2	5.4	5.4	5.4	ns

Copyright © 2011 Integrated Silicon Solution, Inc. All rights reserved. ISSI reserves the right to make changes to this specification and its products at any time without notice. ISSI assumes no liability arising out of the application or use of any information, products or services described herein. Customers are advised to obtain the latest version of this device specification before relying on any published information and before placing orders for products.

Integrated Silicon Solution, Inc. does not recommend the use of any of its products in life support applications where the failure or malfunction of the product can reasonably be expected to cause failure of the life support system or to significantly affect its safety or effectiveness. Products are not authorized for use in such applications unless Integrated Silicon Solution, Inc. receives written assurance to its satisfaction, that:

- a.) the risk of injury or damage has been minimized;
- b.) the user assume all such risks; and
- c.) potential liability of Integrated Silicon Solution, Inc is adequately protected under the circumstances

## GENERAL DESCRIPTION

The 64Mb SDRAM is a high speed CMOS, dynamic random-access memory designed to operate in 3.3V memory systems containing 67,108,864 bits. Internally configured as a quad-bank DRAM with a synchronous interface. Each 16,777,216-bit bank is organized as 2,048 rows by 256 columns by 32 bits.

The 64Mb SDRAM includes an AUTO REFRESH MODE, and a power-saving, power-down mode. All signals are registered on the positive edge of the clock signal, CLK. All inputs and outputs are LVTTTL compatible.

The 64Mb SDRAM has the ability to synchronously burst data at a high data rate with automatic column-address generation, the ability to interleave between internal banks to hide precharge time and the capability to randomly change column addresses on each clock cycle during burst access.

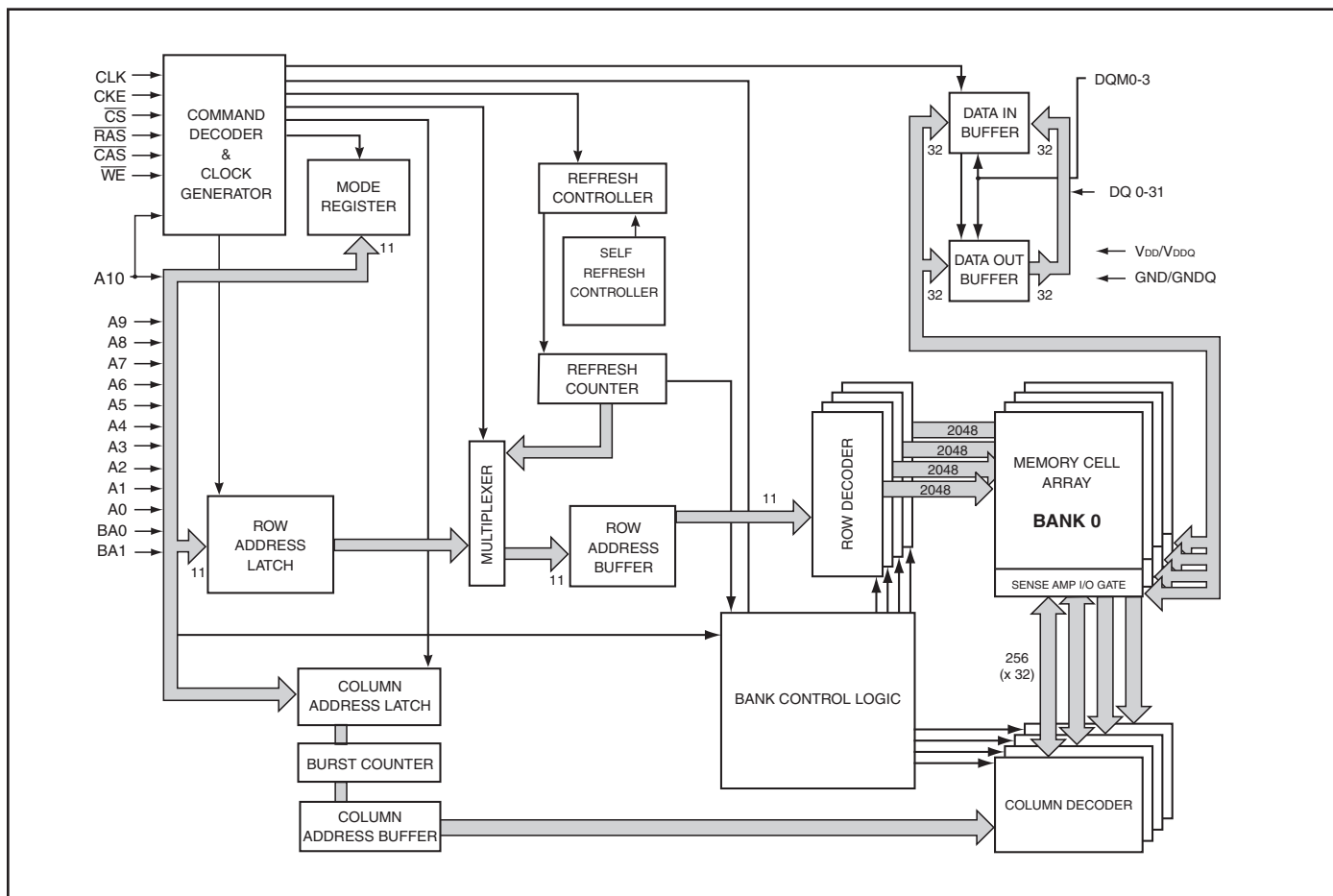
A self-timed row precharge initiated at the end of the burst sequence is available with the AUTO PRECHARGE function enabled. Precharge one bank while accessing one of the

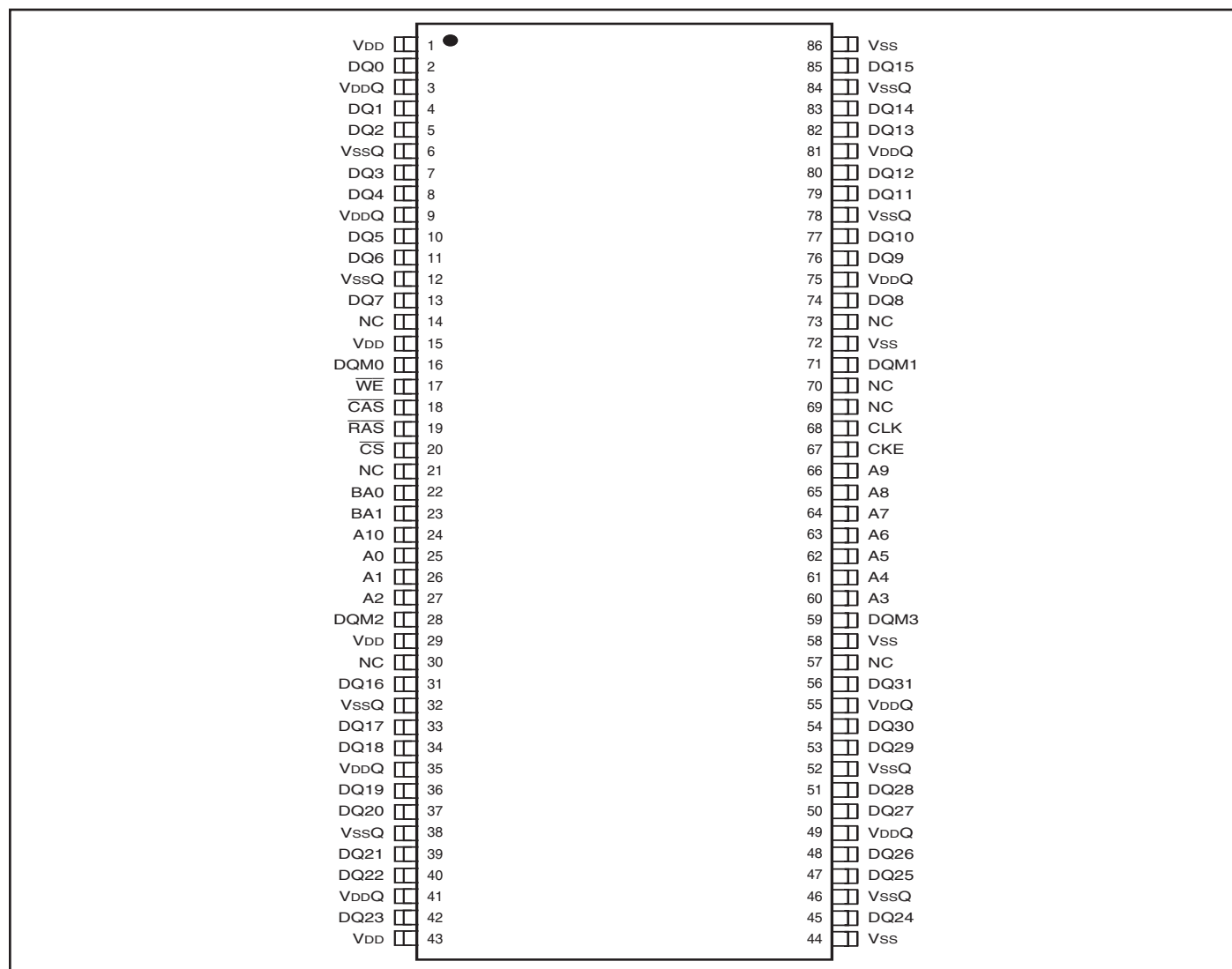
other three banks will hide the precharge cycles and provide seamless, high-speed, random-access operation.

SDRAM read and write accesses are burst oriented starting at a selected location and continuing for a programmed number of locations in a programmed sequence. The registration of an ACTIVE command begins accesses, followed by a READ or WRITE command. The ACTIVE command in conjunction with address bits registered are used to select the bank and row to be accessed (BA0, BA1 select the bank; A0-A10 select the row). The READ or WRITE commands in conjunction with address bits registered are used to select the starting column location for the burst access.

Programmable READ or WRITE burst lengths consist of 1, 2, 4 and 8 locations or full page, with a burst terminate option.

## FUNCTIONAL BLOCK DIAGRAM



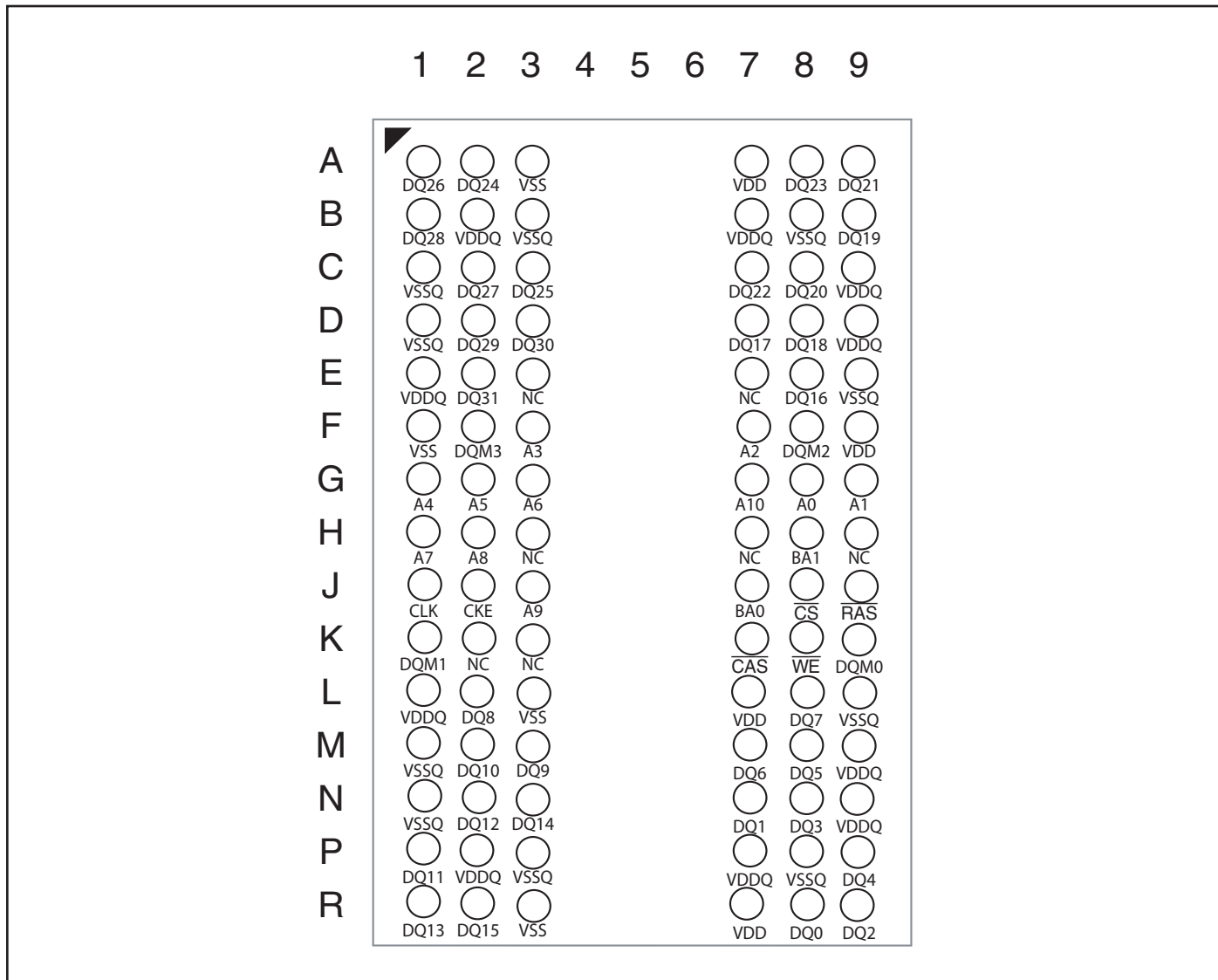
**PIN CONFIGURATIONS**
**86 pin TSOP - Type II for x32**

**PIN DESCRIPTIONS**

A0-A10	Row Address Input
A0-A7	Column Address Input
BA0, BA1	Bank Select Address
DQ0 to DQ31	Data I/O
CLK	System Clock Input
CKE	Clock Enable
CS	Chip Select
RAS	Row Address Strobe Command
CAS	Column Address Strobe Command

WE	Write Enable
DQM0-DQM3	x32 Input/Output Mask
V <sub>DD</sub>	Power
V <sub>SS</sub>	Ground
V <sub>DDQ</sub>	Power Supply for I/O Pin
V <sub>SSQ</sub>	Ground for I/O Pin
NC	No Connection

**PIN CONFIGURATION**

PACKAGE CODE: B 90 BALL TF-BGA (Top View) (8.00 mm x 13.00 mm Body, 0.8 mm Ball Pitch)


**PIN DESCRIPTIONS**

A0-A10	Row Address Input
A0-A7	Column Address Input
BA0, BA1	Bank Select Address
DQ0 to DQ31	Data I/O
CLK	System Clock Input
CKE	Clock Enable
$\overline{CS}$	Chip Select
$\overline{RAS}$	Row Address Strobe Command
$\overline{CAS}$	Column Address Strobe Command

$\overline{WE}$	Write Enable
DQM0-DQM3	x32 Input/Output Mask
VDD	Power
VSS	Ground
VDDQ	Power Supply for I/O Pin
VSSQ	Ground for I/O Pin
NC	No Connection

**PIN FUNCTIONS**

Symbol	Pin No. (TSOP)	Type	Function (In Detail)
A0-A10	25 to 27 60 to 66 24	Input Pin	Address Inputs: A0-A10 are sampled during the ACTIVE command (row-address A0-A10) and READ/WRITE command (A0-A7 with A10 defining auto precharge) to select one location out of the memory array in the respective bank. A10 is sampled during a PRECHARGE command to determine if all banks are to be precharged (A10 HIGH) or bank selected by BA0, BA1 (LOW). The address inputs also provide the op-code during a LOAD MODE REGISTER command.
BA0, BA1	22,23	Input Pin	Bank Select Address: BA0 and BA1 defines which bank the ACTIVE, READ, WRITE or PRECHARGE command is being applied.
CAS	18	Input Pin	CAS, in conjunction with the RAS and WE, forms the device command. See the "Command Truth Table" for details on device commands.
CKE	67	Input Pin	The CKE input determines whether the CLK input is enabled. The next rising edge of the CLK signal will be valid when is CKE HIGH and invalid when LOW. When CKE is LOW, the device will be in either power-down mode, clock suspend mode, or self refresh mode. CKE is an asynchronous input.
CLK	68	Input Pin	CLK is the master clock input for this device. Except for CKE, all inputs to this device are acquired in synchronization with the rising edge of this pin.
CS	20	Input Pin	The CS input determines whether command input is enabled within the device. Command input is enabled when CS is LOW, and disabled with CS is HIGH. The device remains in the previous state when CS is HIGH.
DQ0 to DQ31	2, 4, 5, 7, 8, 10,11,13 74,76,77,79,80,82,83,85 45,47,48,50,51,53,54,56 31,33,34,36,37,39,40,42	DQ Pin	DQ0 to DQ15 are DQ pins. DQ through these pins can be controlled in byte units using the DQM0-DQM3 pins
DQM0 DQM3	16,28,59,71	Input Pin	DQMx control the lower and upper bytes of the DQ buffers. In read mode, the output buffers are placed in a High-Z state. During a WRITE cycle the input data is masked. When DQMx is sampled HIGH and is an input mask signal for write accesses and an output enable signal for read accesses. DQ0 through DQ7 are controlled by DQM0. DQ8 through DQ15 are controlled by DQM1. DQ16 through DQ23 are controlled by DQM2. DQ24 through DQ31 are controlled by DQM3.
RAS	19	Input Pin	RAS, in conjunction with CAS and WE, forms the device command. See the "Command Truth Table" item for details on device commands.
WE	17	Input Pin	WE, in conjunction with RAS and CAS, forms the device command. See the "Command Truth Table" item for details on device commands.
VDDQ	3,9,35,41,49,55,75,81	Supply Pin	VDDQ is the output buffer power supply.
VDD	1,15,29,43	Supply Pin	VDD is the device internal power supply.
GNDQ	6,12,32,38,46,52,78,84	Supply Pin	GNDQ is the output buffer ground.
GND	44,58,72,86	Supply Pin	GND is the device internal ground.

## READ

The READ command selects the bank from BA0, BA1 inputs and starts a burst read access to an active row. Inputs A0-A7 provides the starting column location. When A10 is HIGH, this command functions as an AUTO PRECHARGE command. When the auto precharge is selected, the row being accessed will be precharged at the end of the READ burst. The row will remain open for subsequent accesses when AUTO PRECHARGE is not selected. DQ's read data is subject to the logic level on the DQM inputs two clocks earlier. When a given DQM signal was registered HIGH, the corresponding DQ's will be High-Z two clocks later. DQ's will provide valid data when the DQM signal was registered LOW.

## WRITE

A burst write access to an active row is initiated with the WRITE command. BA0, BA1 inputs selects the bank, and the starting column location is provided by inputs A0-A7. Whether or not AUTO-PRECHARGE is used is determined by A10.

The row being accessed will be precharged at the end of the WRITE burst, if AUTO PRECHARGE is selected. If AUTO PRECHARGE is not selected, the row will remain open for subsequent accesses.

A memory array is written with corresponding input data on DQ's and DQM input logic level appearing at the same time. Data will be written to memory when DQM signal is LOW. When DQM is HIGH, the corresponding data inputs will be ignored, and a WRITE will not be executed to that byte/column location.

## PRECHARGE

The PRECHARGE command is used to deactivate the open row in a particular bank or the open row in all banks. BA0, BA1 can be used to select which bank is precharged or they are treated as "Don't Care". A10 determined whether one or all banks are precharged. After executing this command, the next command for the selected banks(s) is executed after passage of the period  $t_{RP}$ , which is the period required for bank precharging. Once a bank has been precharged, it is in the idle state and must be activated prior to any READ or WRITE commands being issued to that bank.

## AUTO PRECHARGE

The AUTO PRECHARGE function ensures that the precharge is initiated at the earliest valid stage within a burst. This function allows for individual-bank precharge without requiring an explicit command. A10 enables the AUTO PRECHARGE function in conjunction with a specific READ or WRITE command. For each individual READ or WRITE command, auto precharge is either enabled or disabled.

AUTO PRECHARGE does not apply except in full-page burst mode. Upon completion of the READ or WRITE burst, a precharge of the bank/row that is addressed is automatically performed.

## AUTO REFRESH COMMAND

This command executes the AUTO REFRESH operation. The row address and bank to be refreshed are automatically generated during this operation. The stipulated period ( $t_{RC}$ ) is required for a single refresh operation, and no other commands can be executed during this period. This command is executed at least 4096 times every  $t_{REF}$ . During an AUTO REFRESH command, address bits are "Don't Care". This command corresponds to CBR Auto-refresh.

## SELF REFRESH

During the SELF REFRESH operation, the row address to be refreshed, the bank, and the refresh interval are generated automatically internally. SELF REFRESH can be used to retain data in the SDRAM without external clocking, even if the rest of the system is powered down. The SELF REFRESH operation is started by dropping the CKE pin from HIGH to LOW. During the SELF REFRESH operation all other inputs to the SDRAM become "Don't Care". The device must remain in self refresh mode for a minimum period equal to  $t_{RAS}$  or may remain in self refresh mode for an indefinite period beyond that. The SELF-REFRESH operation continues as long as the CKE pin remains LOW and there is no need for external control of any other pins. The next command cannot be executed until the device internal recovery period ( $t_{RC}$ ) has elapsed. Once CKE goes HIGH, the NOP command must be issued (minimum of two clocks) to provide time for the completion of any internal refresh in progress. After the self-refresh, since it is impossible to determine the address of the last row to be refreshed, an AUTO-REFRESH should immediately be performed for all addresses.

## BURST TERMINATE

The BURST TERMINATE command forcibly terminates the burst read and write operations by truncating either fixed-length or full-page bursts and the most recently registered READ or WRITE command prior to the BURST TERMINATE.

## COMMAND INHIBIT

COMMAND INHIBIT prevents new commands from being executed. Operations in progress are not affected, apart from whether the CLK signal is enabled

## NO OPERATION

When  $\overline{CS}$  is low, the NOP command prevents unwanted commands from being registered during idle or wait states.

### **LOAD MODE REGISTER**

During the LOAD MODE REGISTER command the mode register is loaded from A0-A10. This command can only be issued when all banks are idle.

### **ACTIVE COMMAND**

When the ACTIVE COMMAND is activated, BA0, BA1 inputs selects a bank to be accessed, and the address inputs on A0-A10 selects the row. Until a PRECHARGE command is issued to the bank, the row remains open for accesses.



**TRUTH TABLE – COMMANDS AND DQM OPERATION<sup>(1)</sup>**

FUNCTION	$\overline{\text{CS}}$	$\overline{\text{RAS}}$	$\overline{\text{CAS}}$	$\overline{\text{WE}}$	DQM	ADDR	DQs
COMMAND INHIBIT (NOP)	H	X	X	X	X	X	X
NO OPERATION (NOP)	L	H	H	H	X	X	X
ACTIVE (Select bank and activate row) <sup>(3)</sup>	L	L	H	H	X	Bank/Row	X
READ (Select bank/column, start READ burst) <sup>(4)</sup>	L	H	L	H	L/H <sup>(8)</sup>	Bank/Col	X
WRITE (Select bank/column, start WRITE burst) <sup>(4)</sup>	L	H	L	L	L/H <sup>(8)</sup>	Bank/Col	Valid
BURST TERMINATE	L	H	H	L	X	X	Active
PRECHARGE (Deactivate row in bank or banks) <sup>(5)</sup>	L	L	H	L	X	Code	X
AUTO REFRESH or SELF REFRESH <sup>(6,7)</sup> (Enter self refresh mode)	L	L	L	H	X	X	X
LOAD MODE REGISTER <sup>(2)</sup>	L	L	L	L	X	Op-Code	X
Write Enable/Output Enable <sup>(8)</sup>	—	—	—	—	L	—	Active
Write Inhibit/Output High-Z <sup>(8)</sup>	—	—	—	—	H	—	High-Z

**NOTES:**

1. CKE is HIGH for all commands except SELF REFRESH.
2. A0-A10 define the op-code written to the mode register.
3. A0-A10 provide row address, and BA0, BA1 determine which bank is made active.
4. A0-A7 (x32) provide column address; A10 HIGH enables the auto precharge feature (nonpersistent), while A10 LOW disables auto precharge; BA0, BA1 determine which bank is being read from or written to.
5. A10 LOW: BA0, BA1 determine the bank being precharged. A10 HIGH: All banks precharged and BA0, BA1 are “Don’t Care.”
6. AUTO REFRESH if CKE is HIGH, SELF REFRESH if CKE is LOW.
7. Internal refresh counter controls row addressing; all inputs and DQs are “Don’t Care” except for CKE.
8. Activates or deactivates the DQs during WRITES (zero-clock delay) and READS (two-clock delay).

**TRUTH TABLE – CKE** <sup>(1-4)</sup>

CURRENT STATE	COMMAND <sub>n</sub>	ACTION <sub>n</sub>	CKE <sub>n-1</sub>	CKE <sub>n</sub>
Power-Down	X	Maintain Power-Down	L	L
Self Refresh	X	Maintain Self Refresh	L	L
Clock Suspend	X	Maintain Clock Suspend	L	L
Power-Down <sup>(5)</sup>	COMMAND INHIBIT or NOP	Exit Power-Down	L	H
Self Refresh <sup>(6)</sup>	COMMAND INHIBIT or NOP	Exit Self Refresh	L	H
Clock Suspend <sup>(7)</sup>	X	Exit Clock Suspend	L	H
All Banks Idle	COMMAND INHIBIT or NOP	Power-Down Entry	H	L
All Banks Idle	AUTO REFRESH	Self Refresh Entry	H	L
Reading or Writing	VALID	Clock Suspend Entry	H	L
<b>See TRUTH TABLE – CURRENT STATE BANK <i>n</i>, COMMAND TO BANK <i>n</i></b>			H	H

**NOTES:**

1. CKE<sub>n</sub> is the logic state of CKE at clock edge *n*; CKE<sub>n-1</sub> was the state of CKE at the previous clock edge.
2. Current state is the state of the SDRAM immediately prior to clock edge *n*.
3. COMMAND<sub>n</sub> is the command registered at clock edge *n*, and ACTION<sub>n</sub> is a result of COMMAND<sub>n</sub>.
4. All states and sequences not shown are illegal or reserved.
5. Exiting power-down at clock edge *n* will put the device in the all banks idle state in time for clock edge *n+1* (provided that t<sub>CKS</sub> is met).
6. Exiting self refresh at clock edge *n* will put the device in all banks idle state once t<sub>XS<sub>R</sub></sub> is met. COMMAND INHIBIT or NOP commands should be issued on clock edges occurring during the t<sub>XS<sub>R</sub></sub> period. A minimum of two NOP commands must be sent during t<sub>XS<sub>R</sub></sub> period.
7. After exiting clock suspend at clock edge *n*, the device will resume operation and recognize the next command at clock edge *n+1*.

**TRUTH TABLE – CURRENT STATE BANK *n*, COMMAND TO BANK *n*** <sup>(1-6)</sup>

CURRENT STATE	COMMAND (ACTION)	$\overline{CS}$	$\overline{RAS}$	$\overline{CAS}$	$\overline{WE}$
Any	COMMAND INHIBIT (NOP/Continue previous operation)	H	X	X	X
	NO OPERATION (NOP/Continue previous operation)	L	H	H	H
Idle	ACTIVE (Select and activate row)	L	L	H	H
	AUTO REFRESH <sup>(7)</sup>	L	L	L	H
	LOAD MODE REGISTER <sup>(7)</sup>	L	L	L	L
	PRECHARGE <sup>(11)</sup>	L	L	H	L
Row Active	READ (Select column and start READ burst) <sup>(10)</sup>	L	H	L	H
	WRITE (Select column and start WRITE burst) <sup>(10)</sup>	L	H	L	L
	PRECHARGE (Deactivate row in bank or banks) <sup>(8)</sup>	L	L	H	L
Read (Auto Precharge Disabled)	READ (Select column and start new READ burst) <sup>(10)</sup>	L	H	L	H
	WRITE (Select column and start WRITE burst) <sup>(10)</sup>	L	H	L	L
	PRECHARGE (Truncate READ burst, start PRECHARGE) <sup>(8)</sup>	L	L	H	L
	BURST TERMINATE <sup>(9)</sup>	L	H	H	L
Write (Auto Precharge Disabled)	READ (Select column and start READ burst) <sup>(10)</sup>	L	H	L	H
	WRITE (Select column and start new WRITE burst) <sup>(10)</sup>	L	H	L	L
	PRECHARGE (Truncate WRITE burst, start PRECHARGE) <sup>(8)</sup>	L	L	H	L
	BURST TERMINATE <sup>(9)</sup>	L	H	H	L

**NOTE:**

1. This table applies when CKE *n-1* was HIGH and CKE *n* is HIGH (see Truth Table - CKE) and after t<sub>XS<sub>R</sub></sub> has been met (if the previous state was SELF REFRESH).
2. This table is bank-specific, except where noted; i.e., the current state is for a specific bank and the commands shown are those allowed to be issued to that bank when in that state. Exceptions are covered in the notes below.

**3. Current state definitions:**

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

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

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

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

**4. The following states must not be interrupted by a command issued to the same bank. COMMAND INHIBIT or NOP commands, or allowable commands to the other bank should be issued on any clock edge occurring during these states. Allowable commands to the other bank are determined by its current state and CURRENT STATE BANK n truth tables.**

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

Row Activating: Starts with registration of an ACTIVE command and ends when  $t_{RCO}$  is met. Once  $t_{RCO}$  is met, the bank will be in the row active state.

Read w/Auto

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

Write w/Auto

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

**5. The following states must not be interrupted by any executable command; COMMAND INHIBIT or NOP commands must be applied on each positive clock edge during these states.**

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

Accessing Mode

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

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

**6. All states and sequences not shown are illegal or reserved.**

7. Not bank-specific; requires that all banks are idle.

8. May or may not be bank-specific; if all banks are to be precharged, all must be in a valid state for precharging.

9. Not bank-specific; BURST TERMINATE affects the most recent READ or WRITE burst, regardless of bank.

10. READs or WRITEs listed in the Command (Action) column include READs or WRITEs with auto precharge enabled and READs or WRITEs with auto precharge disabled.

11. Does not affect the state of the bank and acts as a NOP to that bank.

**TRUTH TABLE – CURRENT STATE BANK  $n$ , COMMAND TO BANK  $m$  <sup>(1-6)</sup>**

CURRENT STATE	COMMAND (ACTION)	$\overline{CS}$	$\overline{RAS}$	$\overline{CAS}$	$\overline{WE}$
Any	COMMAND INHIBIT (NOP/Continue previous operation)	H	X	X	X
	NO OPERATION (NOP/Continue previous operation)	L	H	H	H
Idle	Any Command Otherwise Allowed to Bank $m$	X	X	X	X
Row	ACTIVE (Select and activate row)	L	L	H	H
Activating, Active, or Precharging	READ (Select column and start READ burst) <sup>(7)</sup>	L	H	L	H
	WRITE (Select column and start WRITE burst) <sup>(7)</sup>	L	H	L	L
	PRECHARGE	L	L	H	L
Read (Auto Precharge Disabled)	ACTIVE (Select and activate row)	L	L	H	H
	READ (Select column and start new READ burst) <sup>(7,10)</sup>	L	H	L	H
	WRITE (Select column and start WRITE burst) <sup>(7,11)</sup>	L	H	L	L
	PRECHARGE <sup>(9)</sup>	L	L	H	L
Write (Auto Precharge Disabled)	ACTIVE (Select and activate row)	L	L	H	H
	READ (Select column and start READ burst) <sup>(7,12)</sup>	L	H	L	H
	WRITE (Select column and start new WRITE burst) <sup>(7,13)</sup>	L	H	L	L
	PRECHARGE <sup>(9)</sup>	L	L	H	L
Read (With Auto Precharge)	ACTIVE (Select and activate row)	L	L	H	H
	READ (Select column and start new READ burst) <sup>(7,8,14)</sup>	L	H	L	H
	WRITE (Select column and start WRITE burst) <sup>(7,8,15)</sup>	L	H	L	L
	PRECHARGE <sup>(9)</sup>	L	L	H	L
Write (With Auto Precharge)	ACTIVE (Select and activate row)	L	L	H	H
	READ (Select column and start READ burst) <sup>(7,8,16)</sup>	L	H	L	H
	WRITE (Select column and start new WRITE burst) <sup>(7,8,17)</sup>	L	H	L	L
	PRECHARGE <sup>(9)</sup>	L	L	H	L

**NOTE:**

- This table applies when CKE  $n-1$  was HIGH and CKE  $n$  is HIGH (Truth Table - CKE) and after  $txsr$  has been met (if the previous state was self refresh).
- This table describes alternate bank operation, except where noted; i.e., the current state is for bank  $n$  and the commands shown are those allowed to be issued to bank  $m$  (assuming that bank  $m$  is in such a state that the given command is allowable). Exceptions are covered in the notes below.
- Current state definitions:
  - Idle: The bank has been precharged, and  $t_{RP}$  has been met.
  - Row Active: A row in the bank has been activated, and  $t_{RCD}$  has been met. No data bursts/accesses and no register accesses are in progress.
  - Read: A READ burst has been initiated, with auto precharge disabled, and has not yet terminated or been terminated.
  - Write: A WRITE burst has been initiated, with auto precharge disabled, and has not yet terminated or been terminated.
  - Read w/Auto Precharge Enabled: Starts with registration of a READ command with auto precharge enabled, and ends when  $t_{RP}$  has been met. Once  $t_{RP}$  is met, the bank will be in the idle state.
  - Write w/Auto Precharge Enabled: Starts with registration of a WRITE command with auto precharge enabled, and ends when  $t_{RP}$  has been met. Once  $t_{RP}$  is met, the bank will be in the idle state.
- AUTO REFRESH, SELF REFRESH and LOAD MODE REGISTER commands may only be issued when all banks are idle.
- A BURST TERMINATE command cannot be issued to another bank; it applies to the bank represented by the current state only.
- All states and sequences not shown are illegal or reserved.
- READs or WRITEs to bank  $m$  listed in the Command (Action) column include READs or WRITEs with auto precharge enabled and READs or WRITEs with auto precharge disabled.

8. CONCURRENT AUTO PRECHARGE: Bank n will initiate the AUTO PRECHARGE command when its burst has been interrupted by bank m's burst.
9. Burst in bank n continues as initiated.
10. For a READ without auto precharge interrupted by a READ (with or without auto precharge), the READ to bank m will interrupt the READ on bank n, CAS latency later (Consecutive READ Bursts).
11. For a READ without auto precharge interrupted by a WRITE (with or without auto precharge), the WRITE to bank m will interrupt the READ on bank n when registered (READ to WRITE). DQM should be used one clock prior to the WRITE command to prevent bus contention.
12. For a WRITE without auto precharge interrupted by a READ (with or without auto precharge), the READ to bank m will interrupt the WRITE on bank n when registered (WRITE to READ), with the data-out appearing CAS latency later. The last valid WRITE to bank n will be data-in registered one clock prior to the READ to bank m.
13. For a WRITE without auto precharge interrupted by a WRITE (with or without auto precharge), the WRITE to bank m will interrupt the WRITE on bank n when registered (WRITE to WRITE). The last valid WRITE to bank n will be data-in registered one clock prior to the READ to bank m.
14. For a READ with auto precharge interrupted by a READ (with or without auto precharge), the READ to bank m will interrupt the READ on bank n, CAS latency later. The PRECHARGE to bank n will begin when the READ to bank m is registered (Fig CAP 1).
15. For a READ with auto precharge interrupted by a WRITE (with or without auto precharge), the WRITE to bank m will interrupt the READ on bank n when registered. DQM should be used two clocks prior to the WRITE command to prevent bus contention. The PRECHARGE to bank n will begin when the WRITE to bank m is registered (Fig CAP 2).
16. For a WRITE with auto precharge interrupted by a READ (with or without auto precharge), the READ to bank m will interrupt the WRITE on bank n when registered, with the data-out appearing CAS latency later. The PRECHARGE to bank n will begin after  $t_{WR}$  is met, where  $t_{WR}$  begins when the READ to bank m is registered. The last valid WRITE to bank n will be data-in registered one clock prior to the READ to bank m (Fig CAP 3).
17. For a WRITE with auto precharge interrupted by a WRITE (with or without auto precharge), the WRITE to bank m will interrupt the WRITE on bank n when registered. The PRECHARGE to bank n will begin after  $t_{WR}$  is met, where  $t_{WR}$  begins when the WRITE to bank m is registered. The last valid WRITE to bank n will be data registered one clock prior to the WRITE to bank m (Fig CAP 4).

## FUNCTIONAL DESCRIPTION

The 64Mb SDRAMs (512K x 32 x 4 banks) are quad-bank DRAMs which operate at 3.3V and include a synchronous interface (all signals are registered on the positive edge of the clock signal, CLK). Each of the 16,777,216-bit banks is organized as 2,048 rows by 256 columns by 32bits.

Read and write accesses to the SDRAM are burst oriented; accesses start at a selected location and continue for a programmed number of locations in a programmed sequence. Accesses begin with the registration of an ACTIVE command which is then followed by a READ or WRITE command. The address bits registered coincident with the ACTIVE command are used to select the bank and row to be accessed (BA0 and BA1 select the bank, A0-A10 select the row). The address bits (A0-A7) registered coincident with the READ or WRITE command are used to select the starting column location for the burst access.

Prior to normal operation, the SDRAM must be initialized. The following sections provide detailed information covering device initialization, register definition, command descriptions and device operation.

## Initialization

SDRAMs must be powered up and initialized in a predefined manner.

The 64M SDRAM is initialized after the power is applied to V<sub>DD</sub> and V<sub>DDQ</sub> (simultaneously) and the clock is stable.

A 100µs delay is required prior to issuing any command other than a COMMAND INHIBIT or a NOP. The COMMAND INHIBIT or NOP may be applied during the 100µs period and continue should at least through the end of the period.

With at least one COMMAND INHIBIT or NOP command having been applied, a PRECHARGE command should be applied once the 100µs delay has been satisfied. All banks must be precharged. This will leave all banks in an idle state where two AUTO REFRESH cycles must be performed. After the AUTO REFRESH cycles are complete, the SDRAM is then ready for mode register programming.

The mode register should be loaded prior to applying any operational command because it will power up in an unknown state.

## REGISTER DEFINITION

### Mode Register

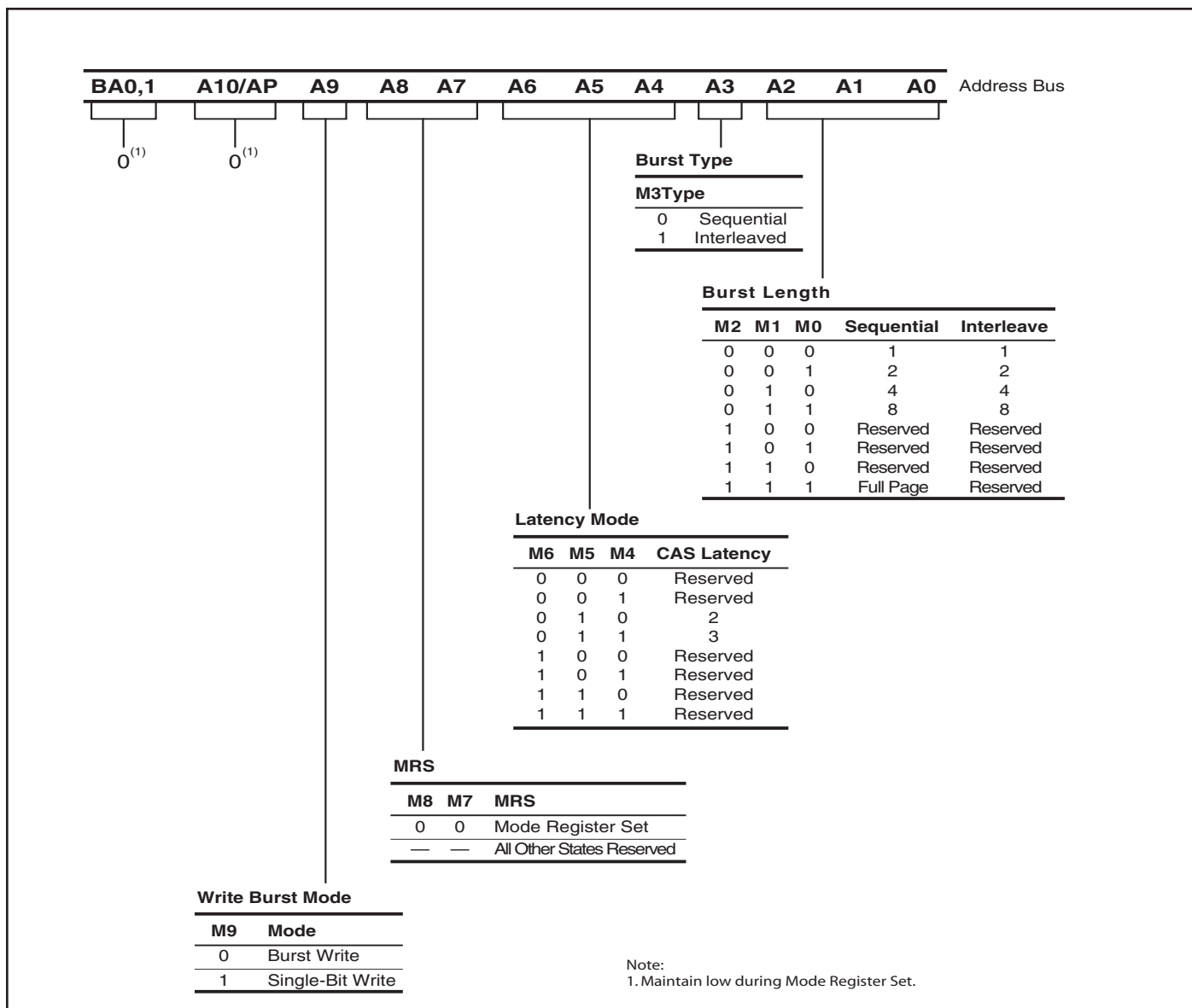
The mode register is used to define the specific mode of operation of the SDRAM. This definition includes the selection of a burst length, a burst type, a CAS\ latency, an operating mode and a write burst mode, as shown in MODE REGISTER DEFINITION.

The mode register is programmed via the LOAD MODE REGISTER command and will retain the stored information until it is programmed again or the device loses power.

Mode register bits M0-M2 specify the burst length, M3 specifies the type of burst (sequential or interleaved), M4- M6 specify the CAS latency, M7 and M8 specify the operating mode, M9 specifies the WRITE burst mode, and M10 and M11 and M12 are reserved for future use.

The mode register must be loaded when all banks are idle, and the controller must wait the specified time before initiating the subsequent operation. Violating either of these requirements will result in unspecified operation.

## MODE REGISTER DEFINITION



## Burst Length

Read and write accesses to the SDRAM are burst oriented, with the burst length being programmable, as shown in MODE REGISTER DEFINITION. The burst length determines the maximum number of column locations that can be accessed for a given READ or WRITE command. Burst lengths of 1, 2, 4 or 8 locations are available for both the sequential and the interleaved burst types, and a full-page burst is available for the sequential type. The full-page burst is used in conjunction with the BURST TERMINATE command to generate arbitrary burst lengths.

Reserved states should not be used, as unknown operation or incompatibility with future versions may result.

When a READ or WRITE command is issued, a block of columns equal to the burst length is effectively selected. All accesses for that burst take place within this block, mean-

ing that the burst will wrap within the block if a boundary is reached. The block is uniquely selected by A1-A7 (x32) when the burst length is set to two; by A2-A7 (x32) when the burst length is set to four; and by A3-A7 (x32) when the burst length is set to eight. The remaining (least significant) address bit(s) is (are) used to select the starting location within the block. Full-page bursts wrap within the page if the boundary is reached.

## Burst Type

Accesses within a given burst may be programmed to be either sequential or interleaved; this is referred to as the burst type and is selected via bit M3.

The ordering of accesses within a burst is determined by the burst length, the burst type and the starting column address, as shown in BURST DEFINITION table.

## BURST DEFINITION

Burst Length	Starting Column Address			Order of Accesses Within a Burst	
				Type = Sequential	Type = Interleaved
	<b>A0</b>				
2		0		0-1	0-1
		1		1-0	1-0
	<b>A1</b>		<b>A0</b>		
4		0	0	0-1-2-3	0-1-2-3
		0	1	1-2-3-0	1-0-3-2
		1	0	2-3-0-1	2-3-0-1
		1	1	3-0-1-2	3-2-1-0
	<b>A2</b>	<b>A1</b>	<b>A0</b>		
8		0	0	0-1-2-3-4-5-6-7	0-1-2-3-4-5-6-7
		0	0	1-2-3-4-5-6-7-0	1-0-3-2-5-4-7-6
		0	1	2-3-4-5-6-7-0-1	2-3-0-1-6-7-4-5
		0	1	3-4-5-6-7-0-1-2	3-2-1-0-7-6-5-4
		1	0	4-5-6-7-0-1-2-3	4-5-6-7-0-1-2-3
		1	0	5-6-7-0-1-2-3-4	5-4-7-6-1-0-3-2
		1	1	6-7-0-1-2-3-4-5	6-7-4-5-2-3-0-1
		1	1	7-0-1-2-3-4-5-6	7-6-5-4-3-2-1-0
Full Page (y)	n = A0-A7 (location 0-y)			Cn, Cn + 1, Cn + 2 Cn + 3, Cn + 4... ...Cn - 1, Cn...	Not Supported



### CAS Latency

The CAS latency is the delay, in clock cycles, between the registration of a READ command and the availability of the first piece of output data. The latency can be set to two or three clocks.

If a READ command is registered at clock edge  $n$ , and the latency is  $m$  clocks, the data will be available by clock edge  $n + m$ . The DQs will start driving as a result of the clock edge one cycle earlier ( $n + m - 1$ ), and provided that the relevant access times are met, the data will be valid by clock edge  $n + m$ . For example, assuming that the clock cycle time is such that all relevant access times are met, if a READ command is registered at  $T_0$  and the latency is programmed to two clocks, the DQs will start driving after  $T_1$  and the data will be valid by  $T_2$ , as shown in CAS Latency diagrams. The **Allowable Operating Frequency** table indicates the operating frequencies at which each CAS latency setting can be used.

Reserved states should not be used as unknown operation or incompatibility with future versions may result.

### Operating Mode

The normal operating mode is selected by setting M7 and M8 to zero; the other combinations of values for M7 and M8 are reserved for future use and/or test modes. The programmed burst length applies to both READ and WRITE bursts.

Test modes and reserved states should not be used because unknown operation or incompatibility with future versions may result.

### Write Burst Mode

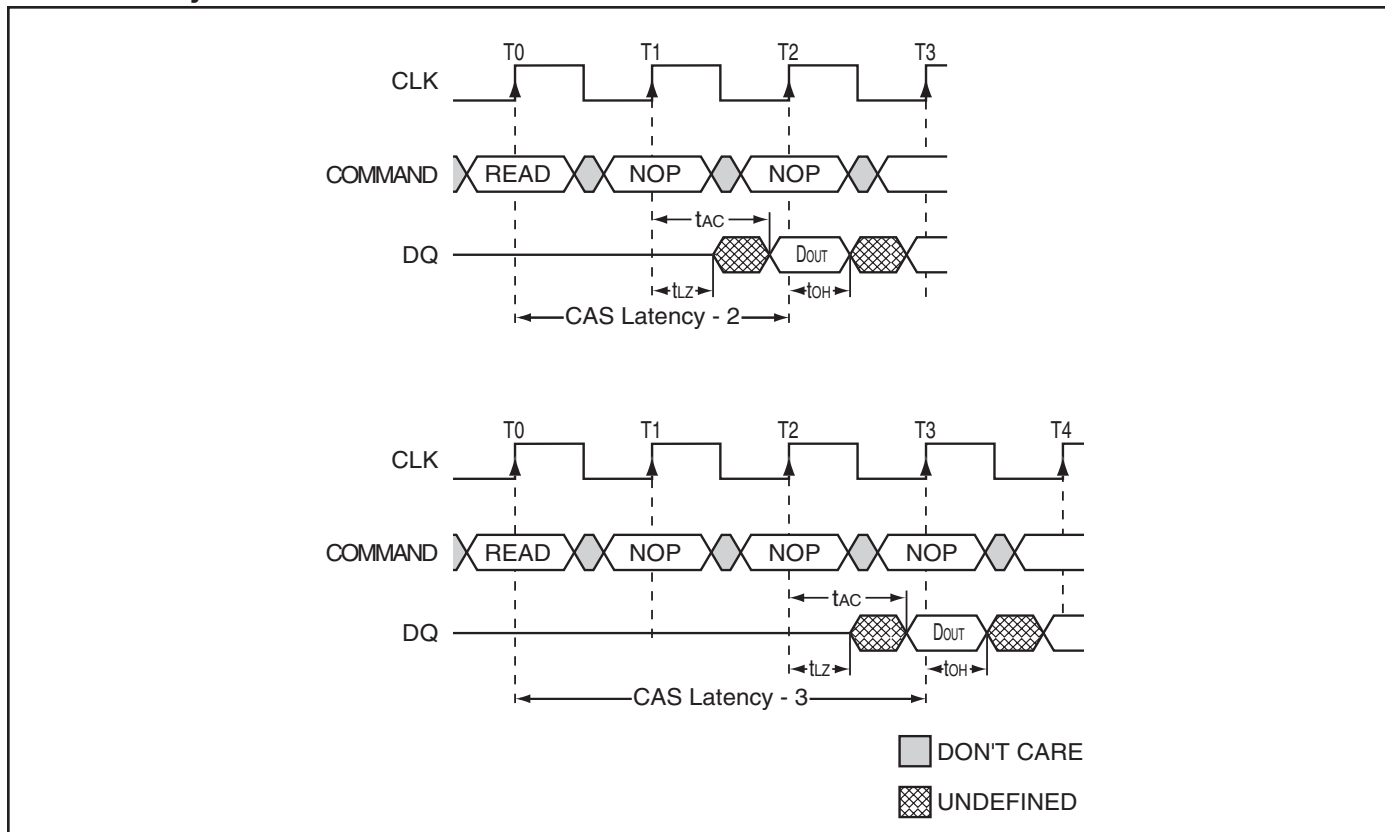
When M9 = 0, the burst length programmed via M0-M2 applies to both READ and WRITE bursts; when M9 = 1, the programmed burst length applies to READ bursts, but write accesses are single-location (nonburst) accesses.

### CAS Latency

**Allowable Operating Frequency (MHz)**

Speed	CAS Latency = 2	CAS Latency = 3
-5	133	200
-6	133	166
-7	133	143

### CAS Latency



## OPERATION

### BANK/ROW ACTIVATION

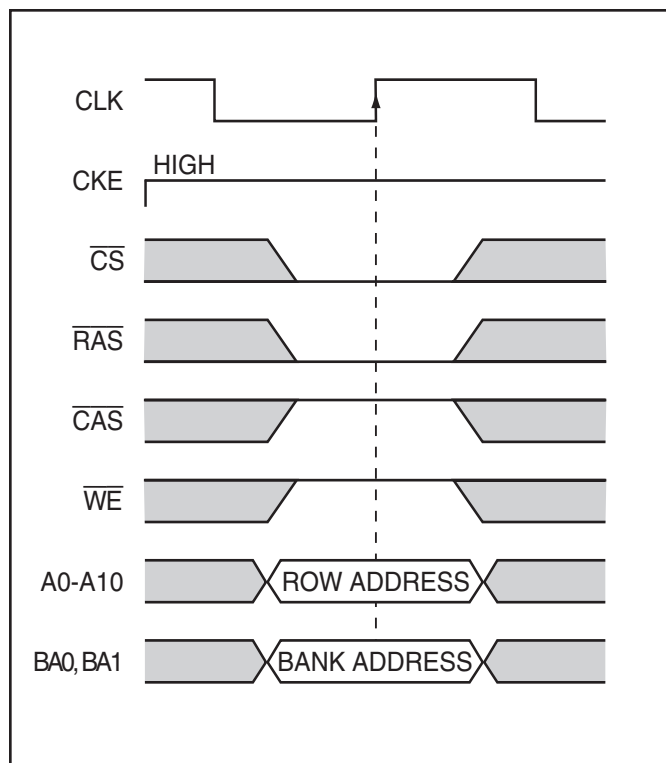
Before any READ or WRITE commands can be issued to a bank within the SDRAM, a row in that bank must be "opened." This is accomplished via the ACTIVE command, which selects both the bank and the row to be activated (see Activating Specific Row Within Specific Bank).

After opening a row (issuing an ACTIVE command), a READ or WRITE command may be issued to that row, subject to the  $t_{RCD}$  specification. Minimum  $t_{RCD}$  should be divided by the clock period and rounded up to the next whole number to determine the earliest clock edge after the ACTIVE command on which a READ or WRITE command can be entered. For example, a  $t_{RCD}$  specification of 20ns with a 125 MHz clock (8ns period) results in 2.5 clocks, rounded to 3. This is reflected in the following example, which covers any case where  $2 < [t_{RCD} (MIN)/t_{CK}] \leq 3$ . (The same procedure is used to convert other specification limits from time units to clock cycles).

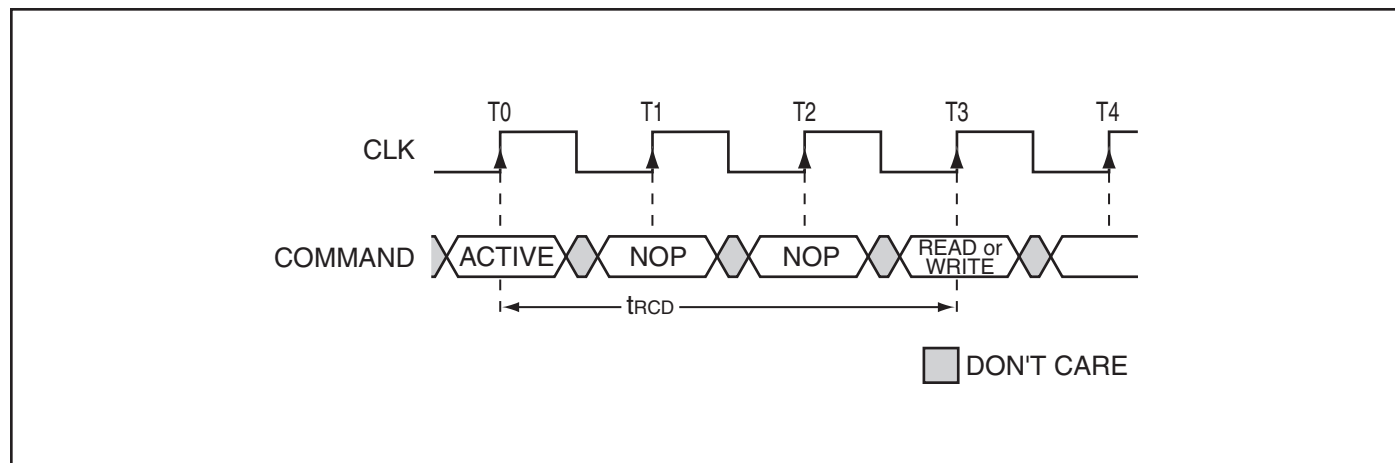
A subsequent ACTIVE command to a different row in the same bank can only be issued after the previous active row has been "closed" (precharged). The minimum time interval between successive ACTIVE commands to the same bank is defined by  $t_{RC}$ .

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

### Activating Specific Row Within Specific Bank



### Example: Meeting $t_{RCD} (MIN)$ when $2 < [t_{RCD} (min)/t_{CK}] \leq 3$



## READS

READ bursts are initiated with a READ command, as shown in the READ COMMAND diagram.

The starting column and bank addresses are provided with the READ command, and auto precharge is either enabled or disabled for that burst access. If auto precharge is enabled, the row being accessed is precharged at the completion of the burst. For the generic READ commands used in the following illustrations, auto precharge is disabled.

During READ bursts, the valid data-out element from the starting column address will be available following the CAS latency after the READ command. Each subsequent data-out element will be valid by the next positive clock edge. The CAS Latency diagram shows general timing for each possible CAS latency setting.

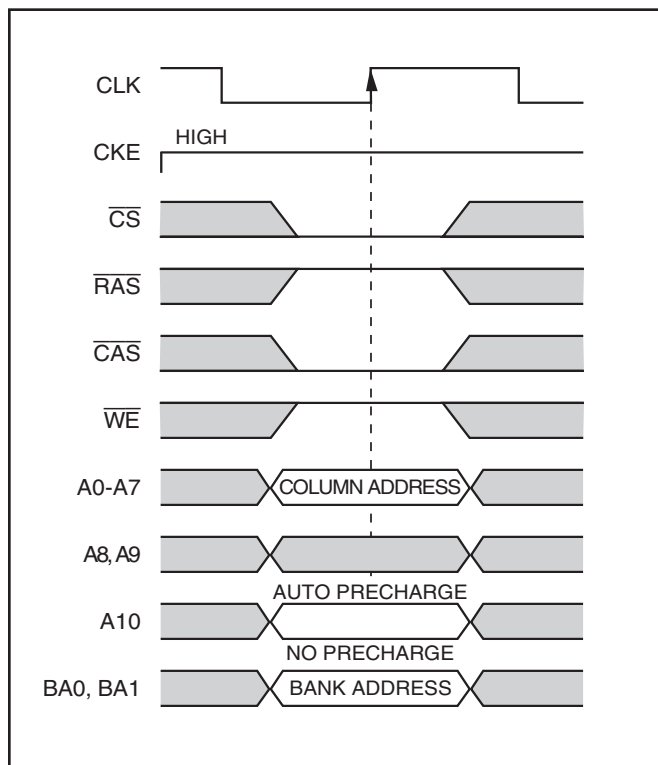
Upon completion of a burst, assuming no other commands have been initiated, the DQs will go High-Z. A full-page burst will continue until terminated. (At the end of the page, it will wrap to column 0 and continue.)

Data from any READ burst may be truncated with a subsequent READ command, and data from a fixed-length READ burst may be immediately followed by data from a READ command. In either case, a continuous flow of data can be maintained. The first data element from the new burst follows either the last element of a completed burst or the last desired data element of a longer burst which is being truncated.

The new READ command should be issued  $x$  cycles before the clock edge at which the last desired data element is valid, where  $x$  equals the CAS latency minus one. This is shown in Consecutive READ Bursts for CAS latencies of two and three; data element  $n + 3$  is either the last of a burst of four or the last desired of a longer burst. The 64Mb SDRAM uses a pipelined architecture and therefore does not require the  $2n$  rule associated with a prefetch architecture. A READ command can be initiated on any clock cycle following a previous READ command. Full-speed random read accesses can be performed to the same bank, as shown in Random READ Accesses, or each subsequent READ may be performed to a different bank.

Data from any READ burst may be truncated with a subsequent WRITE command, and data from a fixed-length READ burst may be immediately followed by data from a WRITE command (subject to bus turnaround limitations). The WRITE burst may be initiated on the clock edge immediately following the last (or last desired) data element from the READ burst, provided that DQ contention can be avoided. In a given system design, there may be a possibility that the device driving the input data will go Low-Z before the SDRAM DQs go High-Z. In this case, at least a single-cycle delay should occur between the last read data and the WRITE command.

## READ COMMAND



The DQM input is used to avoid DQ contention, as shown in Figures RW1 and RW2. The DQM signal must be asserted (HIGH) at least two clocks prior to the WRITE command (DQM latency is two clocks for output buffers) to suppress data-out from the READ. Once the WRITE command is registered, the DQs will go High-Z (or remain High-Z), regardless of the state of the DQM signal, provided the DQM was active on the clock just prior to the WRITE command that truncated the READ command. If not, the second WRITE will be an invalid WRITE. For example, if DQM was LOW during T4 in Figure RW2, then the WRITES at T5 and T7 would be valid, while the WRITE at T6 would be invalid.

The DQM signal must be de-asserted prior to the WRITE command (DQM latency is zero clocks for input buffers) to ensure that the written data is not masked. Figure RW1 shows the case where the clock frequency allows for bus contention to be avoided without adding a NOP cycle, and Figure RW2 shows the case where the additional NOP is needed.

A fixed-length READ burst may be followed by, or truncated with, a PRECHARGE command to the same bank (provided that auto precharge was not activated), and a full-page burst may be truncated with a PRECHARGE command to the

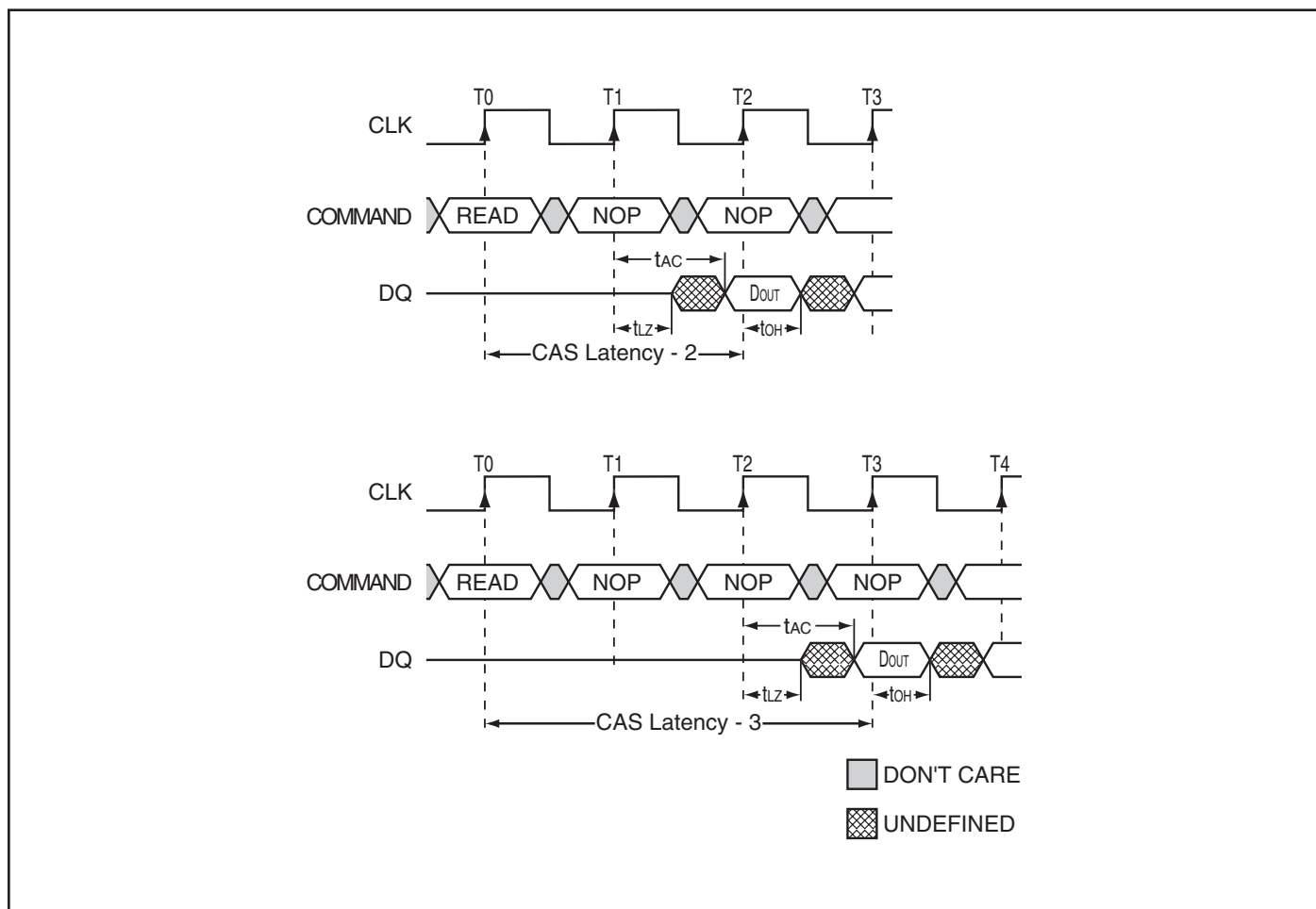
same bank. The PRECHARGE command should be issued  $x$  cycles before the clock edge at which the last desired data element is valid, where  $x$  equals the CAS latency minus one. This is shown in the READ to PRECHARGE diagram for each possible CAS latency; data element  $n + 3$  is either the last of a burst of four or the last desired of a longer burst. Following the PRECHARGE command, a subsequent command to the same bank cannot be issued until  $t_{RP}$  is met. Note that part of the row precharge time is hidden during the access of the last data element(s).

In the case of a fixed-length burst being executed to completion, a PRECHARGE command issued at the optimum time (as described above) provides the same operation that would result from the same fixed-length burst with auto precharge. The disadvantage of the PRE-

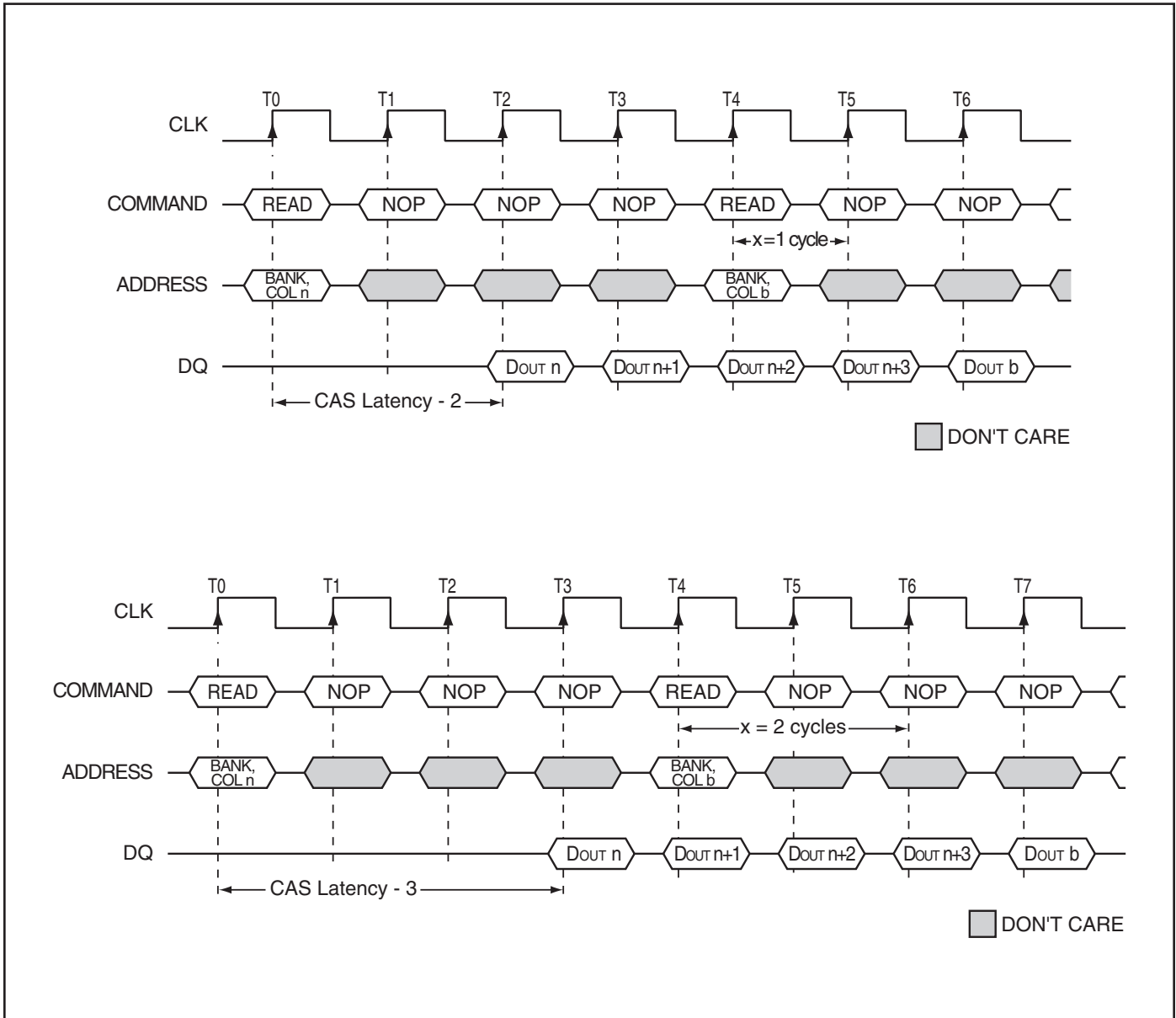
CHARGE command is that it requires that the command and address buses be available at the appropriate time to issue the command; the advantage of the PRECHARGE command is that it can be used to truncate fixed-length or full-page bursts.

Full-page READ bursts can be truncated with the BURST TERMINATE command, and fixed-length READ bursts may be truncated with a BURST TERMINATE command, provided that auto precharge was not activated. The BURST TERMINATE command should be issued  $x$  cycles before the clock edge at which the last desired data element is valid, where  $x$  equals the CAS latency minus one. This is shown in the READ Burst Termination diagram for each possible CAS latency; data element  $n + 3$  is the last desired data element of a longer burst.

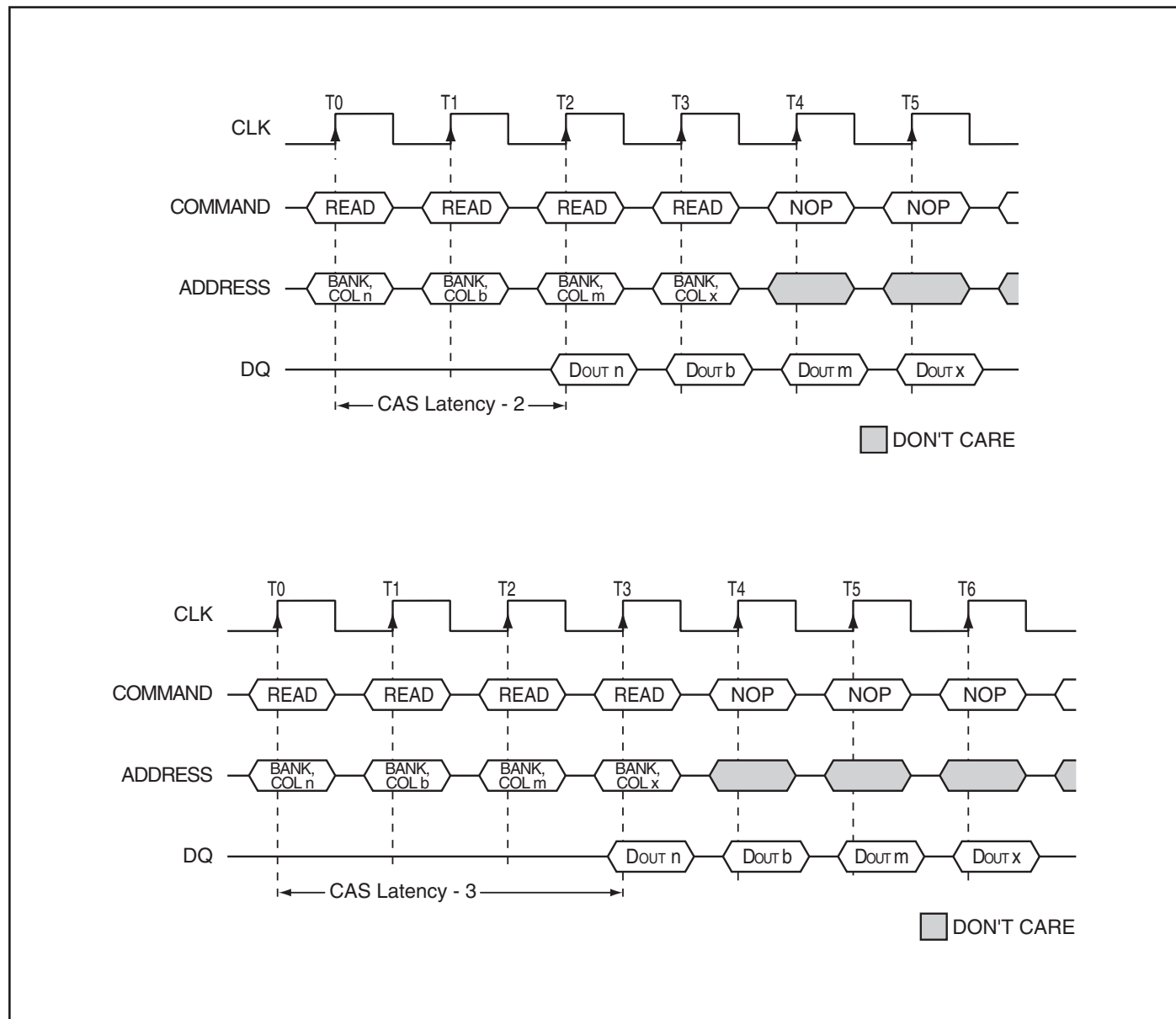
### CAS Latency



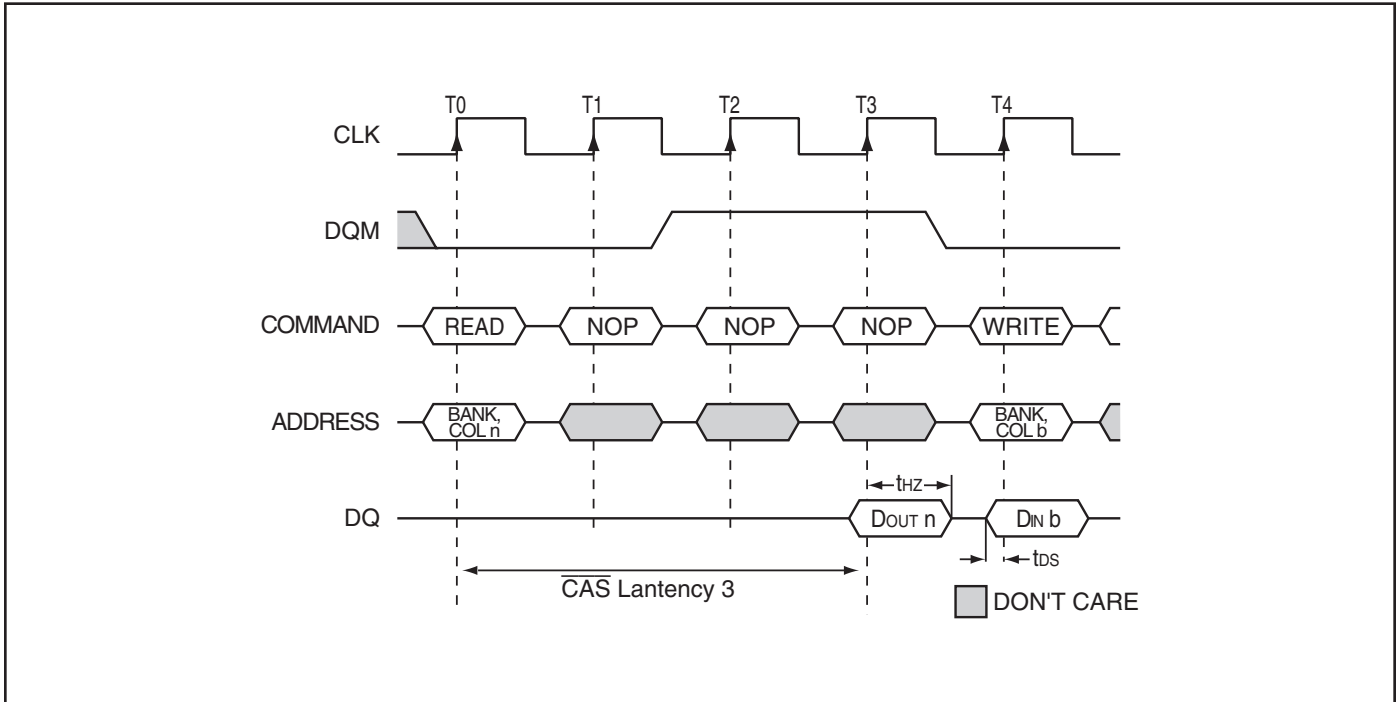
### Consecutive READ Bursts



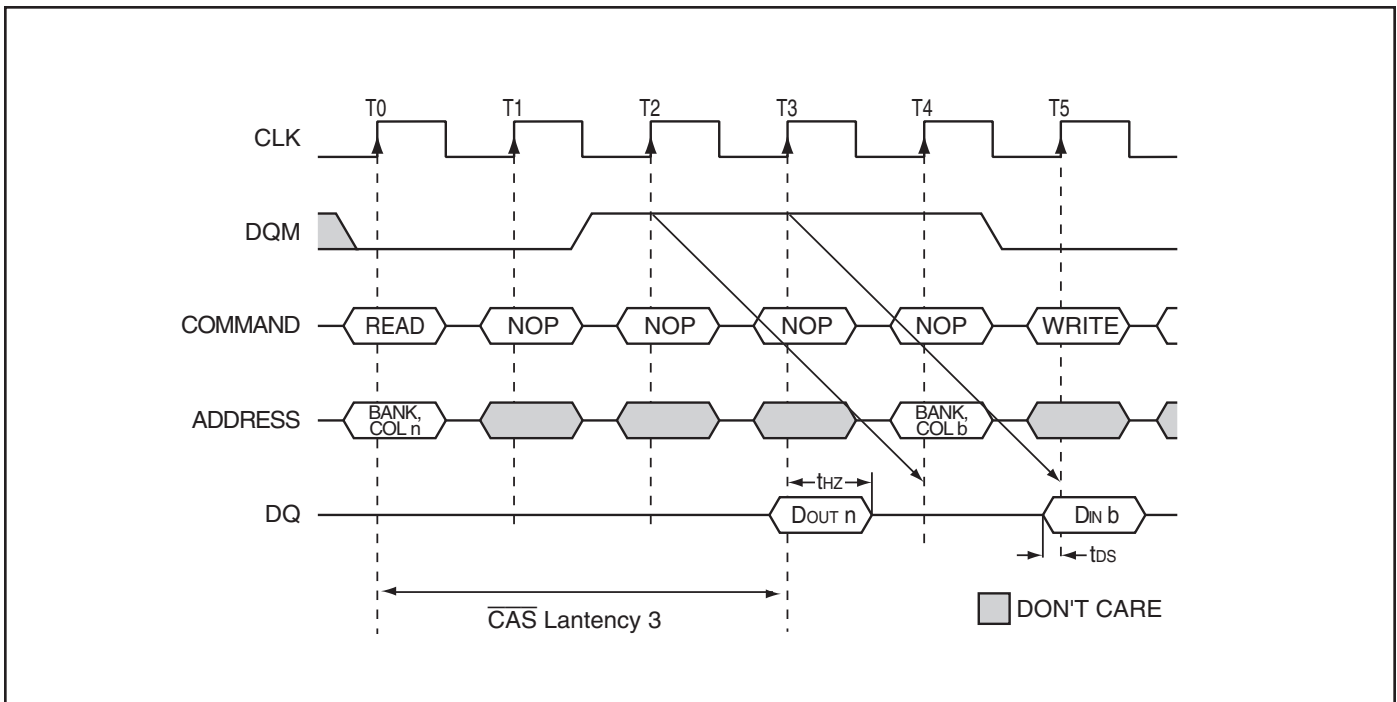
Random READ Accesses



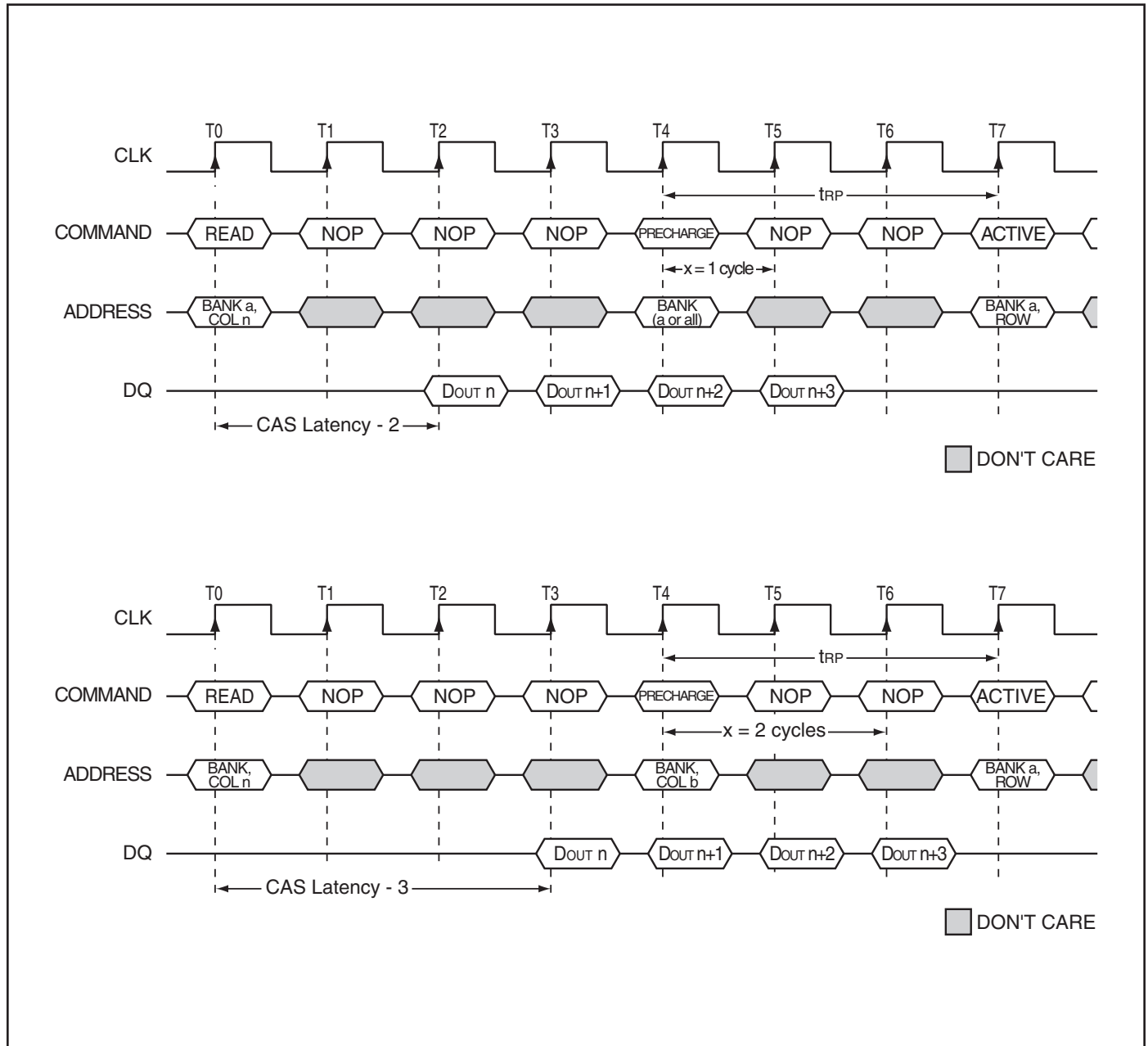
**RW1 - READ to WRITE**



**RW2 - READ to WRITE With Extra Clock Cycle**

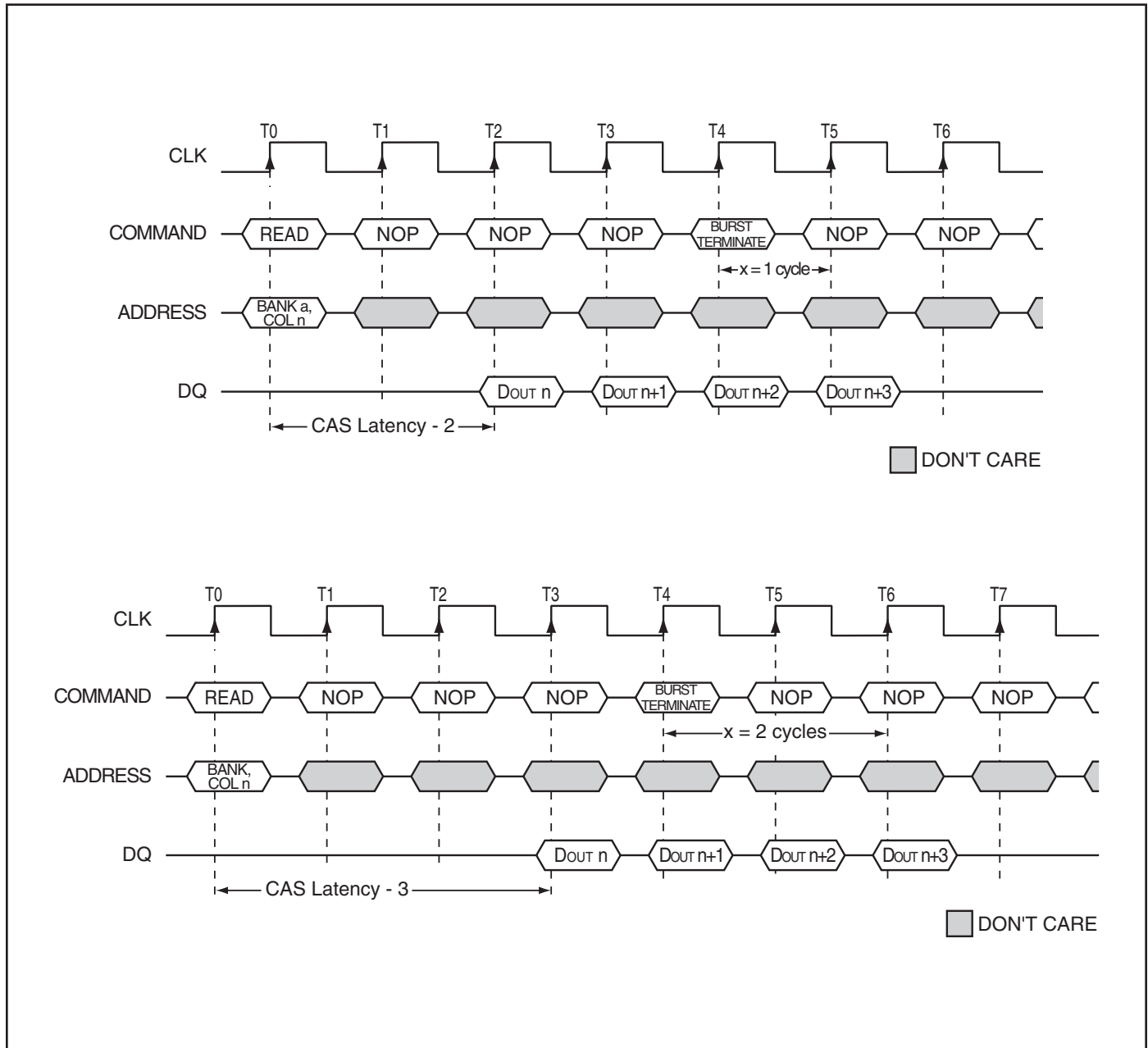


**READ to PRECHARGE**





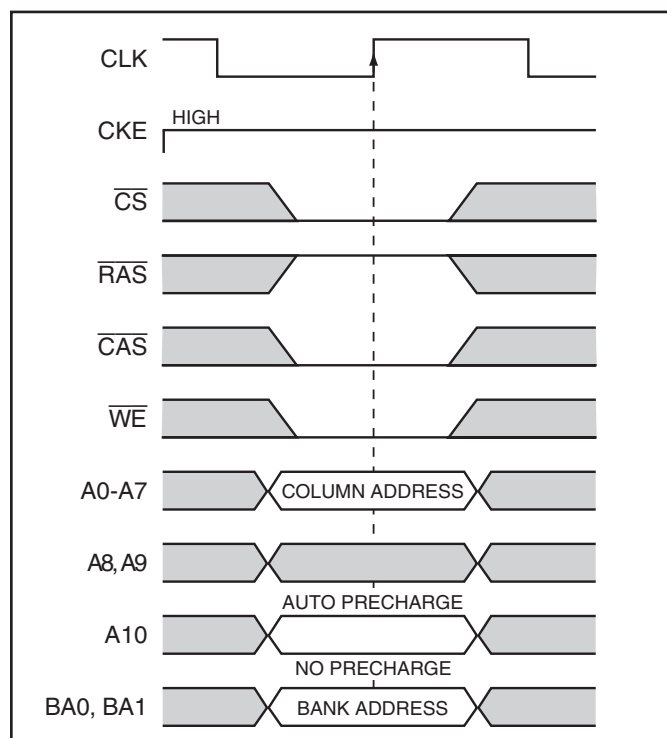
### READ Burst Termination



## WRITEs

WRITE bursts are initiated with a WRITE command, as shown in WRITE Command diagram.

### WRITE Command



The starting column and bank addresses are provided with the WRITE command, and auto precharge is either enabled or disabled for that access. If auto precharge is enabled, the row being accessed is precharged at the completion of the burst. For the generic WRITE commands used in the following illustrations, auto precharge is disabled.

During WRITE bursts, the first valid data-in element will be registered coincident with the WRITE command. Subsequent data elements will be registered on each successive positive clock edge. Upon completion of a fixed-length burst, assuming no other commands have been initiated, the DQs will remain High-Z and any additional input data will be ignored (see WRITE Burst). A full-page burst will continue until terminated. (At the end of the page, it will wrap to column 0 and continue.)

Data for any WRITE burst may be truncated with a subsequent WRITE command, and data for a fixed-length WRITE burst may be immediately followed by data for a WRITE command. The new WRITE command can be issued on any clock following the previous WRITE command, and the data provided coincident with the new command applies to the new command.

An example is shown in WRITE to WRITE diagram. Data  $n + 1$  is either the last of a burst of two or the last desired of a longer burst. The 64Mb SDRAM uses a pipelined architecture and therefore does not require the  $2n$  rule associated with a prefetch architecture. A WRITE command can be initiated on any clock cycle following a previous WRITE command. Full-speed random write accesses within a page can be performed to the same bank, as shown in Random WRITE Cycles, or each subsequent WRITE may be performed to a different bank.

Data for any WRITE burst may be truncated with a subsequent READ command, and data for a fixed-length WRITE burst may be immediately followed by a subsequent READ command. Once the READ command is registered, the data inputs will be ignored, and WRITEs will not be executed. An example is shown in WRITE to READ. Data  $n + 1$  is either the last of a burst of two or the last desired of a longer burst.

Data for a fixed-length WRITE burst may be followed by, or truncated with, a PRECHARGE command to the same bank (provided that auto precharge was not activated), and a full-page WRITE burst may be truncated with a PRECHARGE command to the same bank. The PRECHARGE command should be issued  $t_{WR}$  after the clock edge at which the last desired input data element is registered. The auto precharge mode requires a  $t_{WR}$  of at least one clock plus time, regardless of frequency. In addition, when truncating a WRITE burst, the DQM signal must be used to mask input data for the clock edge prior to, and the clock edge coincident with, the PRECHARGE command. An example is shown in the WRITE to PRECHARGE diagram. Data  $n+1$  is either the last of a burst of two or the last desired of a longer burst. Following the PRECHARGE command, a subsequent command to the same bank cannot be issued until  $t_{RP}$  is met.

In the case of a fixed-length burst being executed to completion, a PRECHARGE command issued at the optimum time (as described above) provides the same operation that would result from the same fixed-length burst with auto precharge. The disadvantage of the PRECHARGE command is that it requires that the command and address buses be available at the appropriate time to issue the command; the advantage of the PRECHARGE command is that it can be used to truncate fixed-length or full-page bursts.

Fixed-length or full-page WRITE bursts can be truncated with the BURST TERMINATE command. When truncating a WRITE burst, the input data applied coincident with the BURST TERMINATE command will be ignored. The last data written (provided that DQM is LOW at that time) will be the input data applied one clock previous to the BURST TERMINATE command. This is shown in WRITE Burst Termination, where data  $n$  is the last desired data element of a longer burst.