S73WS256N Based MCPs

Stacked Multi-Chip Product (MCP)

512/256 Megabit (32M/16M x 16-bit) CMOS 1.8 Volt-only, Simultaneous Read/Write, Burst Mode Flash Memory with 256/128 Megabit (4M/2M x 16-bit x 4 Banks) Mobile SDRAM on Shared Data Bus

SPANSION™

Data Sheet

ADVANCE INFORMATION

Notice to Readers: This document states the current technical specifications regarding the Spansion product(s) described herein. Each product described herein may be designated as Advance Information, Preliminary, or Full Production. See "Notice On Data Sheet Designations" for definitions.

Publication Number S73WS256N_00 **Revision** A **Amendment** 3 **Issue Date** December 16, 2005

Notice On Data Sheet Designations

Spansion LLC issues data sheets with Advance Information or Preliminary designations to advise readers of product information or intended specifications throughout the product life cycle, including development, qualification, initial production, and full production. In all cases, however, readers are encouraged to verify that they have the latest information before finalizing their design. The following descriptions of Spansion data sheet designations are presented here to highlight their presence and definitions.

Advance Information

The Advance Information designation indicates that Spansion LLC is developing one or more specific products, but has not committed any design to production. Information presented in a document with this designation is likely to change, and in some cases, development on the product may discontinue. Spansion LLC therefore places the following conditions upon Advance Information content:

> "This document contains information on one or more products under development at Spansion LLC. The information is intended to help you evaluate this product. Do not design in this product without contacting the factory. Spansion LLC reserves the right to change or discontinue work on this proposed product without notice."

Preliminary

The Preliminary designation indicates that the product development has progressed such that a commitment to production has taken place. This designation covers several aspects of the product life cycle, including product qualification, initial production, and the subsequent phases in the manufacturing process that occur before full production is achieved. Changes to the technical specifications presented in a Preliminary document should be expected while keeping these aspects of production under consideration. Spansion places the following conditions upon Preliminary content:

> "This document states the current technical specifications regarding the Spansion product(s) described herein. The Preliminary status of this document indicates that product qualification has been completed, and that initial production has begun. Due to the phases of the manufacturing process that require maintaining efficiency and quality, this document may be revised by subsequent versions or modifications due to changes in technical specifications."

Combination

Some data sheets will contain a combination of products with different designations (Advance Information, Preliminary, or Full Production). This type of document will distinguish these products and their designations wherever necessary, typically on the first page, the ordering information page, and pages with the DC Characteristics table and the AC Erase and Program table (in the table notes). The disclaimer on the first page refers the reader to the notice on this page.

Full Production (No Designation on Document)

When a product has been in production for a period of time such that no changes or only nominal changes are expected, the Preliminary designation is removed from the data sheet. Nominal changes may include those affecting the number of ordering part numbers available, such as the addition or deletion of a speed option, temperature range, package type, or V_{IO} range. Changes may also include those needed to clarify a description or to correct a typographical error or incorrect specification. Spansion LLC applies the following conditions to documents in this category:

> "This document states the current technical specifications regarding the Spansion product(s) described herein. Spansion LLC deems the products to have been in sufficient production volume such that subsequent versions of this document are not expected to change. However, typographical or specification corrections, or modifications to the valid combinations offered may occur."

Questions regarding these document designations may be directed to your local AMD or Fujitsu sales office.

S73WS256N based MCPs

Stacked Multi-Chip Product (MCP)

512/256 Megabit (32M/16M x 16-bit) CMOS 1.8 Volt-only, Simultaneous Read/Write, Burst Mode Flash Memory with 256/128 Megabit (4M/2M x 16-bit x 4 Banks) Mobile SDRAM on Shared Data Bus

Data Sheet

ADVANCE INFORMATION

Distinctive Characteristics

MCP Features

- Power supply voltage of 1.7 to 1.95V

High Performance

- Flash access time: 80ns
- Flash burst frequency: 54 MHz, 66MHz, 80MHz
- Mobile SDRAM burst frequency: 104 MHz
- Package:
  - 9.0 x 12.0 mm
- Operating Temperature
  - –25°C to +85°C (wireless)

General Description

The S73WS series is a product line of stacked Multi-Chip Product (MCP) packages and consists of:

- One or two (in this case, one die is used as code and the other as data) flash memory die
- One Mobile SDRAM die
- Shared address/data bus for Flash and Mobile SDRAM: 96-ball pinout

The products covered by this document are listed in the table below. For details about their specifications, please refer to the individual constituent datasheets for further details:

		Flash Memory Density	
		256Mb	512Mb
Mobile SDRAM Density	128Mb	S73WS256ND0	S73WS256NDE
	256Mb		S73WS256NEE

1 Product Selector Guide

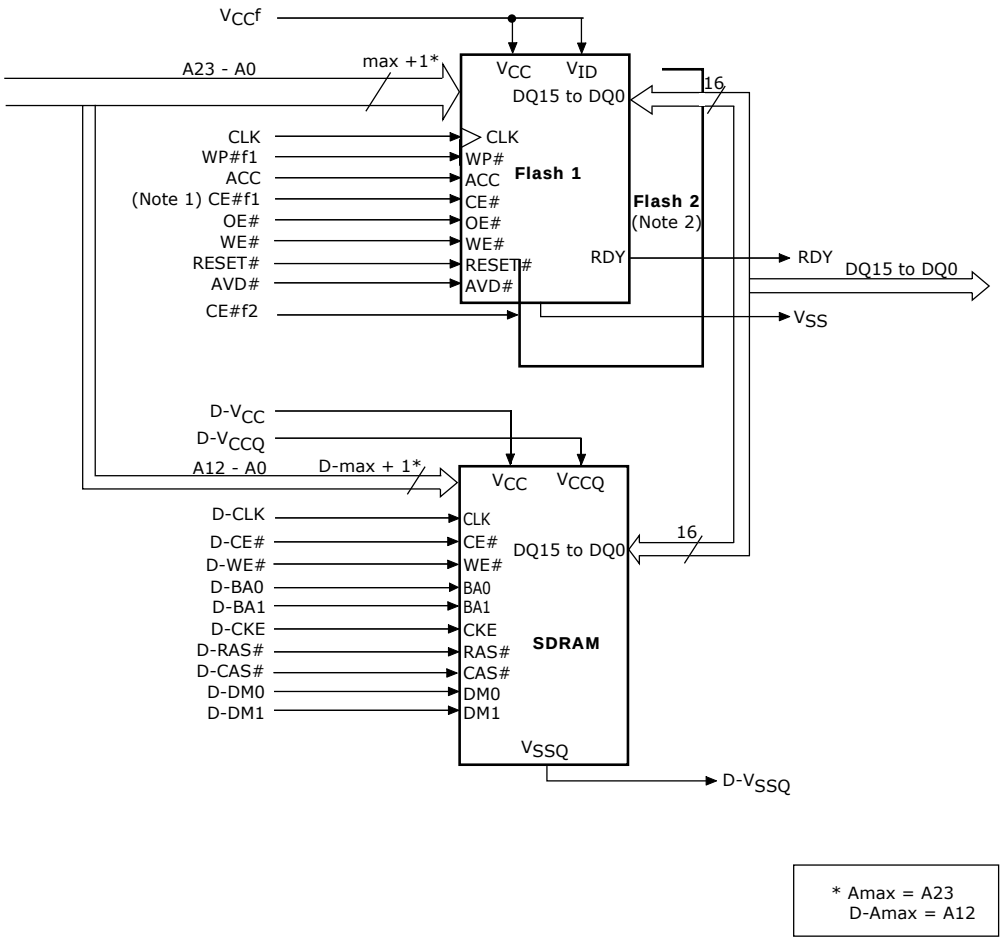
Device-Model#	Flash Density (Code)	Flash Density (Data)	Flash Initial/Burst Speed (ns/MHz)	SDRAM Density	SDRAM burst Speed (MHz)	Supplier	DYB	Package
S73WS256ND0BAWA7	256 Mb	—	80ns/54MHz	128	104 MHz	1	sector unprotected	9x12x1.2 137-ball
S73WS256ND0BAWAB						2		
S73WS256NDEBAWT7	256 Mb	256 Mb	80ns/54MHz	128	104 MHz	1	sector unprotected	9x12x1.4 137-ball
S73WS256NDEBAWTB						2		
S73WS256NEEBAWT7				256		1		
S73WS256NEEBAWTB						2		
S73WS256ND0BFWA7	256 Mb	—	80ns/54MHz	128	104 MHz	1	sector unprotected	9x12x1.2 137-ball
S73WS256ND0BFWAB						2		
S73WS256NDEBFWT7	256 Mb	256 Mb	80ns/54MHz	128	104 MHz	1	sector unprotected	9x12x1.4 137-ball
S73WS256NDEBFWTB						2		
S73WS256NEEBFWT7				256		1		
S73WS256NEEBFWTB						2		

Table of Contents

List of Tables

List of Figures

2 MCP Block Diagram



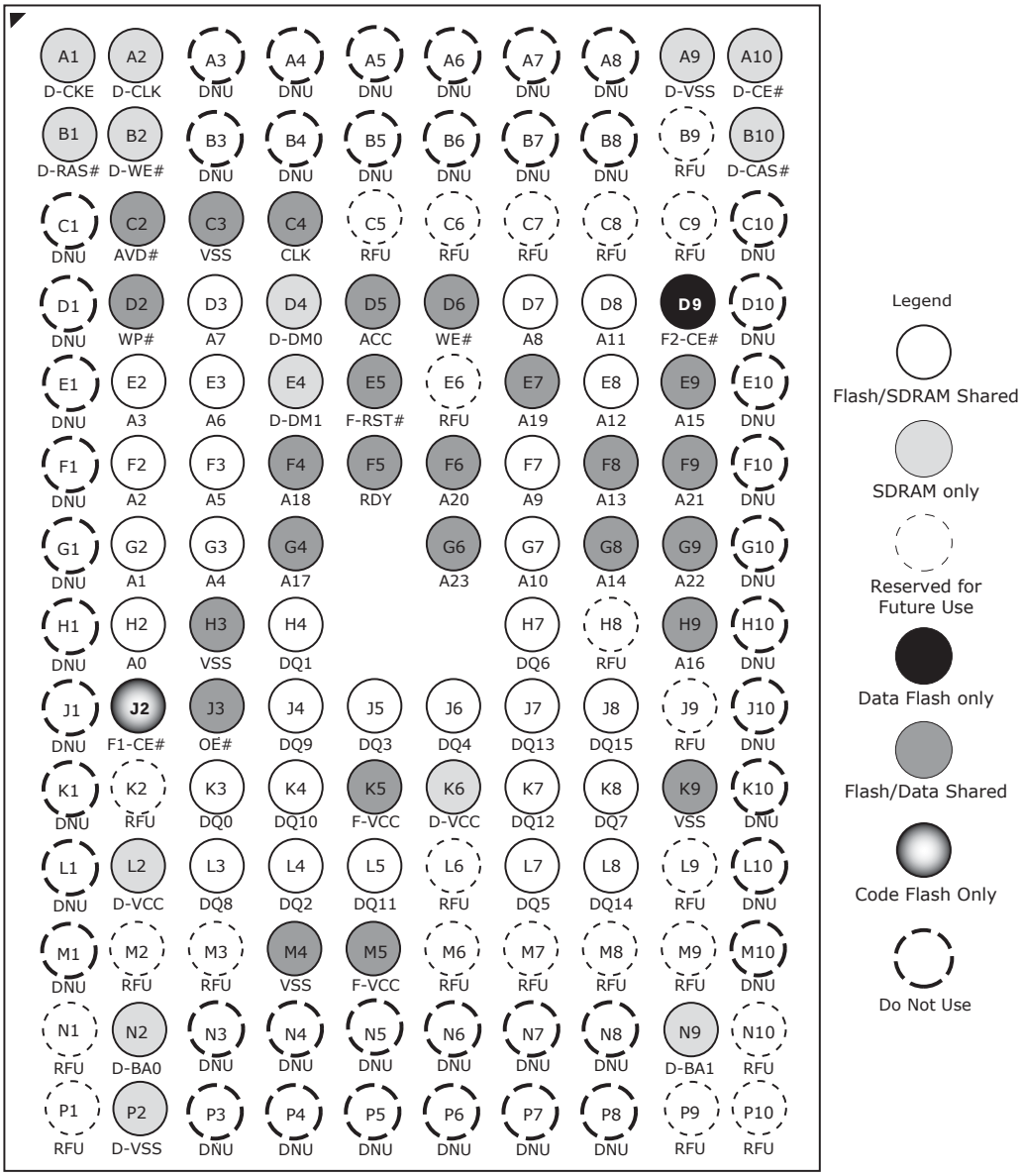
Notes:

1. *For a one-Flash configuration, CE#f1 = CE#.*
   For a two-Flash configuration, CE#f1 = CE for Flash 1 and CE#f2 = CE for Flash 2; CE#f2 is the chip-enable pin for the second Flash.
2. *Flash and SDRAM share common A12-A0 and DQ15-DQ0.*
3. *Applicable only for the S73WS256NDE and S73WS256NEE devices.*

3 Connection Diagrams

3.1 2 x 256Mb Flash with 256Mb SDRAM

137-ball Fine-Pitch Ball Grid Array
(Top View, Balls Facing Down)



Notes:

1. *Tie the SDRAM V_{SS} and V_{SSQ} to D-V_{SS}.*
2. *Tie the SDRAM V_{CC} and V_{CCQ} to D-V_{CC}.*

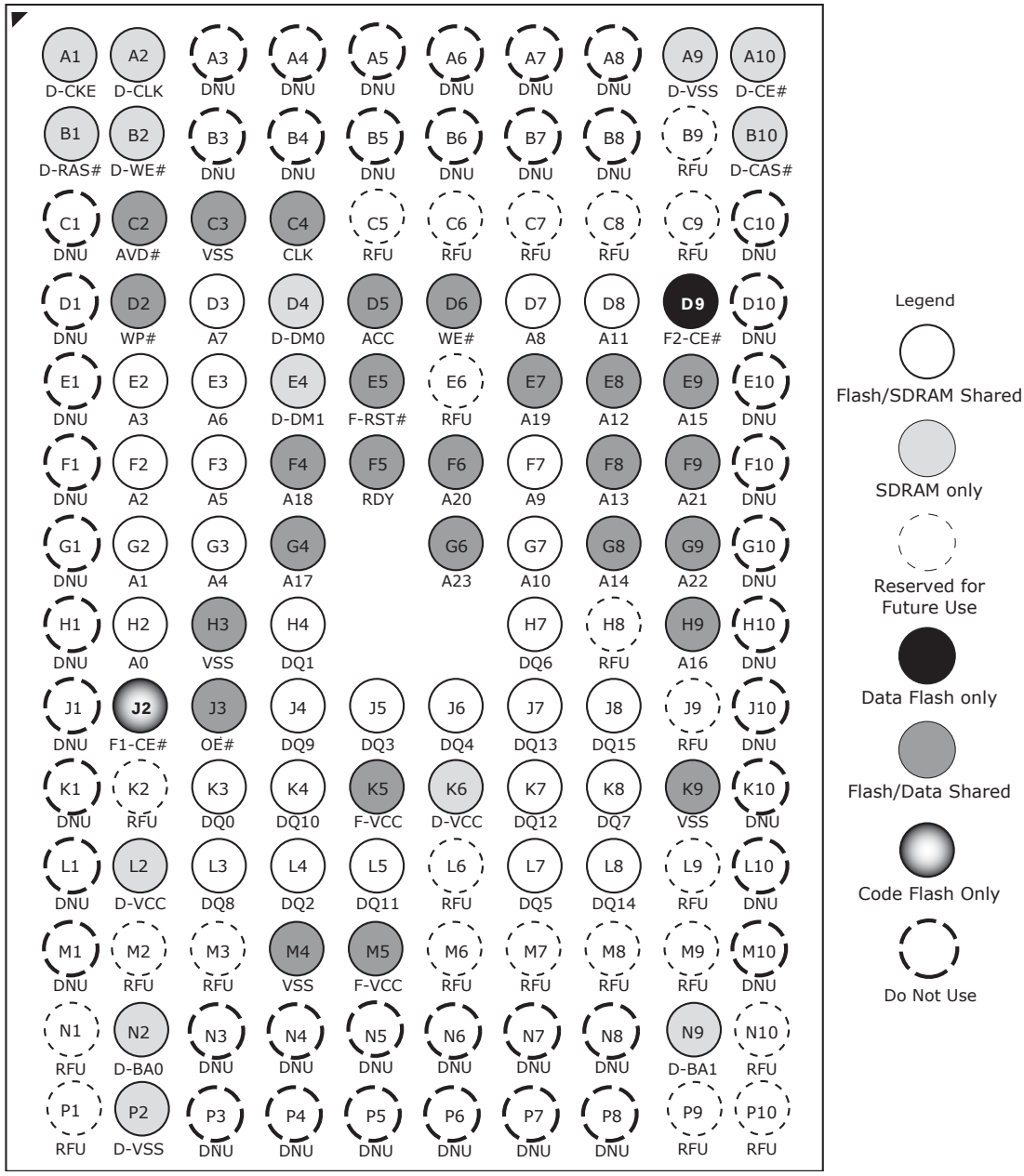
Special Handling Instructions For FBGA Package

Special handling is required for Flash Memory products in FBGA packages.

Flash memory devices in FBGA packages may be damaged if exposed to ultrasonic cleaning methods. The package and/or data integrity may be compromised if the package body is exposed to temperatures above 150°C for prolonged periods of time.

3.2 2 x 256Mb Flash with I28Mb SDRAM

137-ball Fine-Pitch Ball Grid Array
(Top View, Balls Facing Down)



Notes:

1. *Tie the SDRAM V_{SS} and V_{SSQ} to D-V_{SS}.*
2. *Tie the SDRAM V_{CC} and V_{CCQ} to D-V_{CC}.*

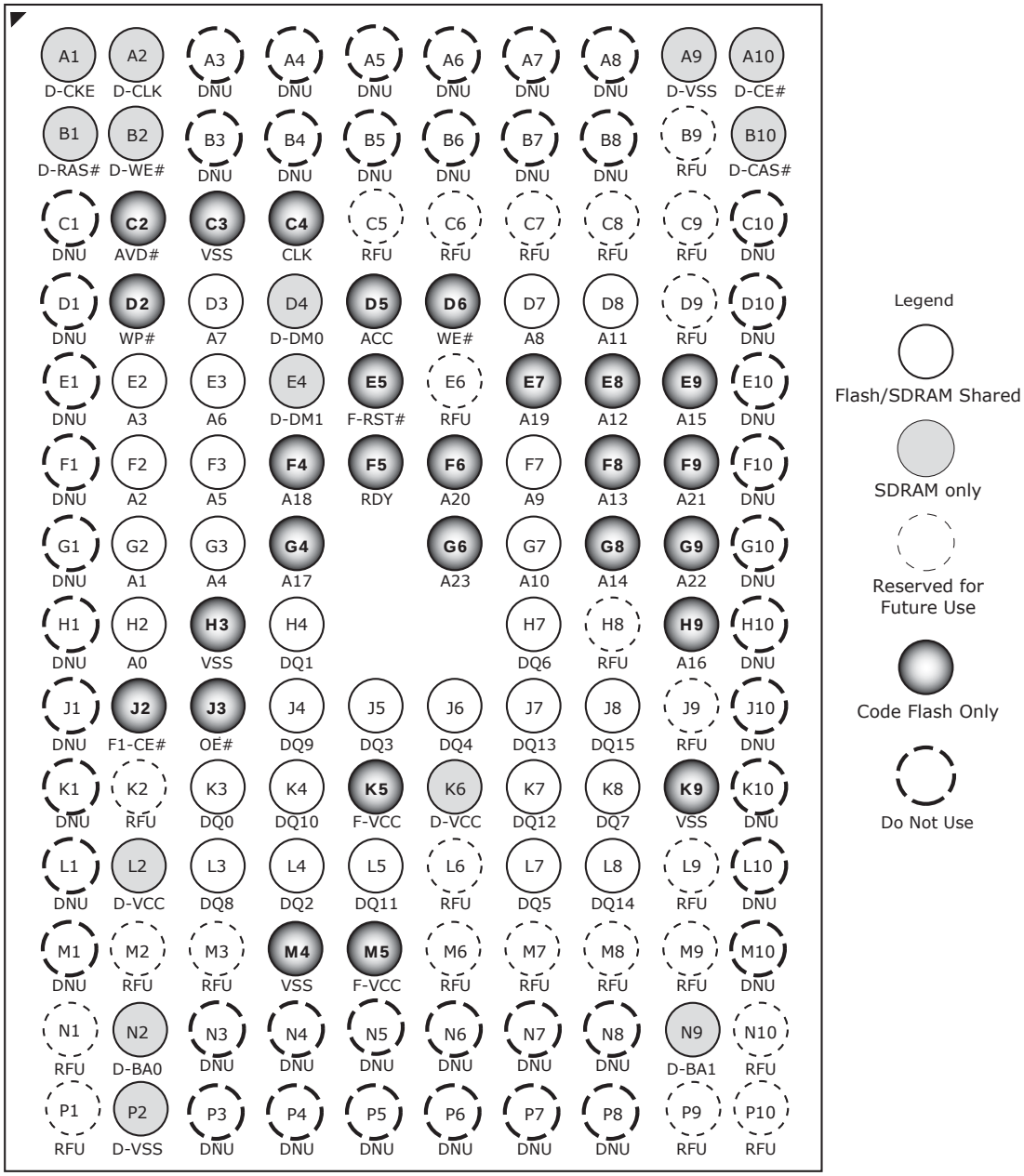
Special Handling Instructions For FBGA Package

Special handling is required for Flash Memory products in FBGA packages.

Flash memory devices in FBGA packages may be damaged if exposed to ultrasonic cleaning methods. The package and/or data integrity may be compromised if the package body is exposed to temperatures above 150°C for prolonged periods of time.

3.3 256MbFlash with I28Mb SDRAM

137-ball Fine-Pitch Ball Grid Array
(Top View, Balls Facing Down)



Notes:

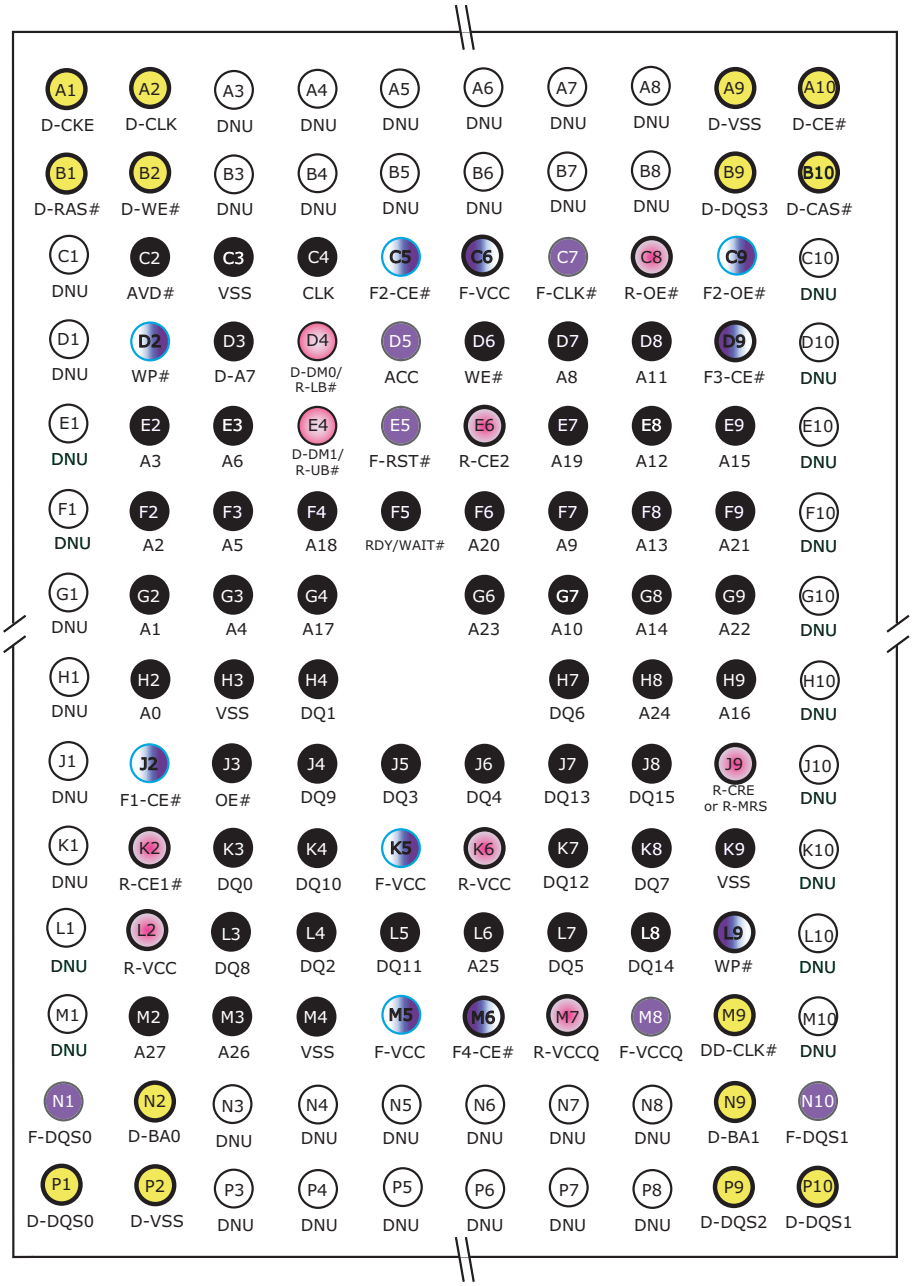
1. *Tie the SDRAM V_{SS} and V_{SSQ} to D-V_{SS}.*
2. *Tie the SDRAM V_{CC} and V_{CCQ} to D-V_{CC}.*

Special Handling Instructions For FBGA Package

Special handling is required for Flash Memory products in FBGA packages.

Flash memory devices in FBGA packages may be damaged if exposed to ultrasonic cleaning methods. The package and/or data integrity may be compromised if the package body is exposed to temperatures above 150°C for prolonged periods of time.

3.4 Lookahead Diagram on Shared Bus



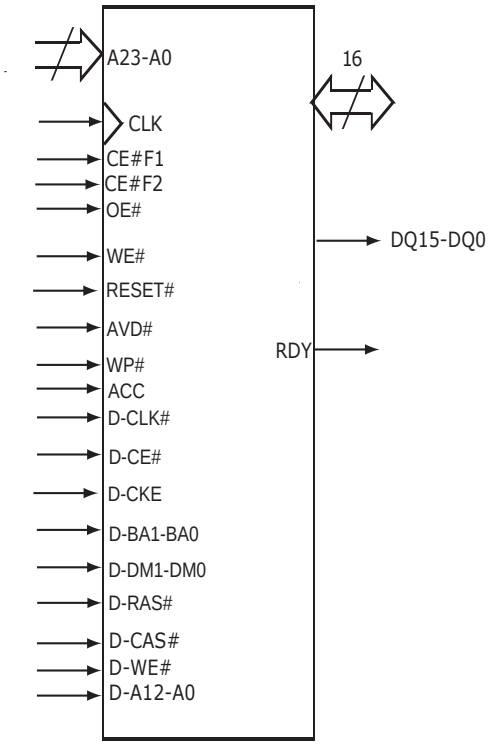
Notes:

1. *In addition to being defined as F2-CE#, Ball C5 can also be assigned as F1-CE2# for code flash that has two chip enable signals.*
2. *F1 and F2 denote XIP/Code Flash, while F3 and F4 denote Data/Companion Flash.*

4 Input/Output Descriptions

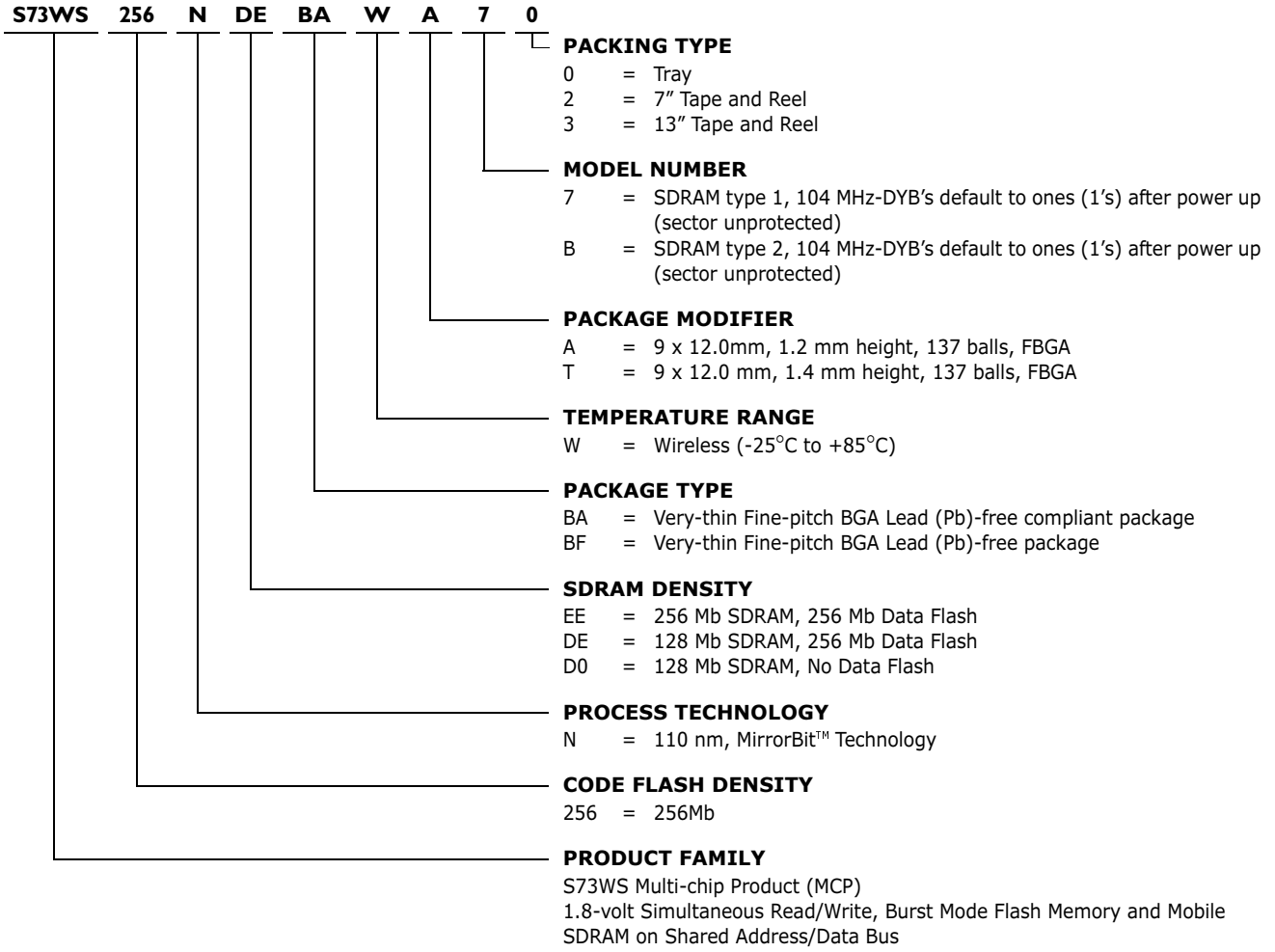
A23-A0	=	Flash Address inputs
A12-A0	=	SDRAM Address inputs
DQ15-DQ0	=	Flash and SDRAM shared bus input/output
CE#f2	=	Flash Chip-enable input # 2. Asynchronous relative to CLK for burst mode.
CE#f1	=	Flash Chip-enable input #1. Asynchronous relative to CLK for Burst Mode.
OE#	=	Flash Output Enable input. Asynchronous relative to CLK for Burst mode.
WE#	=	Flash Write Enable input.
$V_{CC}f$	=	Flash device power supply (1.7 V - 1.95V).
$V_{CC}q$	=	Input/Output Buffer power supply.
V_{SS}	=	Ground
RFU	=	Reserved for Future Use
RDY	=	Flash ready output. Indicates the status of the Burst read. VOL = data valid.
CLK	=	Flash Clock. The first rising edge of CLK in conjunction with AVD# low latches the address input and activates burst mode operation. After the initial word is output, subsequent rising edges of CLK increment the internal address counter. CLK should remain low during asynchronous access.
AVD#	=	Flash Address Valid input. Indicates to device that the valid address is present on the address inputs. VIL = for asynchronous mode, indicates valid address; for burst mode, causes starting address to be latched on rising edge of CLK. VIH= device ignores address inputs
RESET#	=	Flash hardware reset input. VIL= device resets and returns to reading array data
WP#	=	Flash hardware write protect input. VIL = disables program and erase functions in the four outermost sectors.
ACC	=	Flash accelerated input. At VHH, accelerates programming; automatically places device in unlock bypass mode. At VIL, disables all program and erase functions. Should be at VIH for all other conditions.
D-CLK	=	SDRAM System Clock
D-CE#	=	SDRAM Chip Select
D-CKE	=	SDRAM Clock Enable
D-BA1-BA0	=	SDRAM Bank Select
D-RAS#	=	SDRAM Row Address Strobe
D-CAS#	=	SDRAM Column Address Strobe
D-DM1-D-DM0	=	SDRAM Data Input/Output Mask
D-WE#	=	SDRAM Write Enable input
D-VSS	=	SDRAM Ground
D-VSSQ	=	SDRAM Input/Output Buffer ground
D-VCCQ	=	SDRAM Input/Output Buffer power supply
D-VCC	=	SDRAM device power supply

5 Logic Symbol for MCP



6 Ordering Information

The order number is formed by a valid combinations of the following:



S73WS256ND0 Valid Combinations				FLash Initial/ Burst Speed	SDRAM Supplier	SDRAM Burst Speed	Package Type	Package Marking
Base Ordering Part Number	Package & Temperature	Model Number	Packing Type					
S73WS256ND0	BAW, BFW	A7	0, 2, 3 (Note 1)	70ns/54 MHz	Supplier 1	104 MHz	9x12x1.2mm 137-ball	(Note 2)
		AB			Supplier 2			

Notes:
1. *Type 0 is standard. Specify other options as required.*
2. *BGA package marking omits leading "S" and packing type designator from ordering part number.*

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult your local sales office to confirm availability of specific valid combinations and to check on newly released combinations.

S73WS256NDE Valid Combinations				FLash Initial/ Burst Speed	SDRAM Supplier	SDRAM Burst Speed	Package Type	Package Marking
Base Ordering Part Number	Package & Temperature	Model Number	Packing Type					
S73WS256NDE	BAW, BFW	T7	0, 2, 3 (Note 1)	70ns/54 MHz	Supplier 1	104 MHz	9x12x1.4mm 137-ball	(Note 2)
		TB			Supplier 2			

Notes:
1. *Type 0 is standard. Specify other options as required.*
2. *BGA package marking omits leading "S" and packing type designator from ordering part number.*

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult your local sales office to confirm availability of specific valid combinations and to check on newly released combinations.

S73WS256NEE Valid Combinations				FLash Initial/ Burst Speed	SDRAM Supplier	SDRAM Burst Speed	Package Type	Package Marking
Base Ordering Part Number	Package & Temperature	Model Number	Packing Type					
S73WS256NEE	BAW, BFW	T7	0, 2, 3 (Note 1)	70ns/54 MHz	Supplier 1	104 MHz	9x12x1.4mm 137-ball	(Note 2)
		TB			Supplier 2			

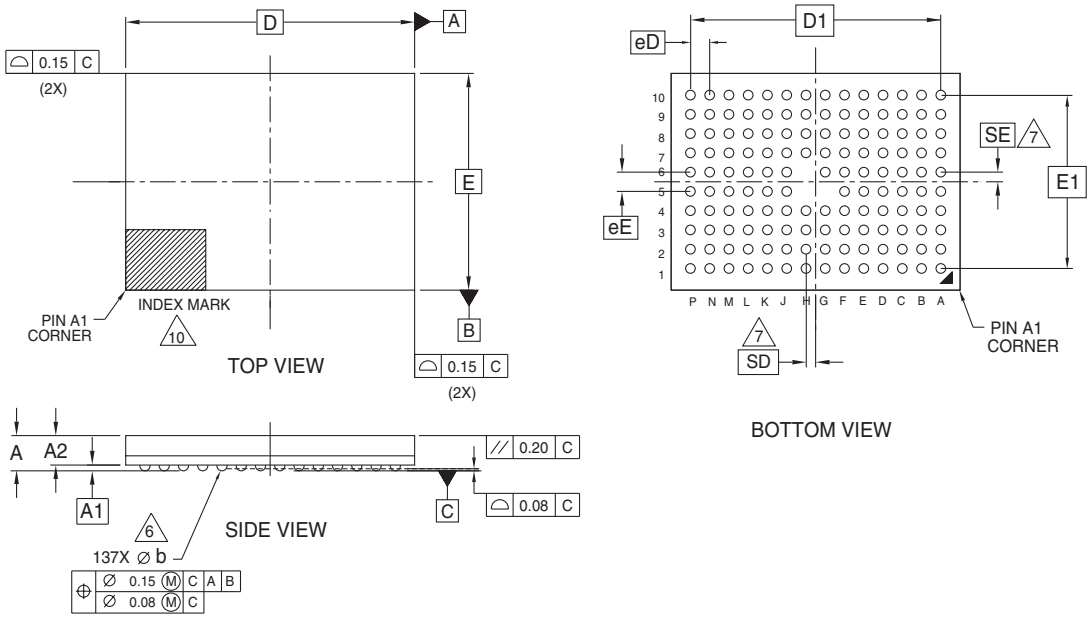
Notes:
1. *Type 0 is standard. Specify other options as required.*
2. *BGA package marking omits leading "S" and packing type designator from ordering part number.*

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult your local sales office to confirm availability of specific valid combinations and to check on newly released combinations.

7 Physical Dimensions

7.1 TLD137—137-ball Fine-Pitch Ball Grid Array (FBGA) 9 x 12.0 mm Package



PACKAGE	TLD 137			
JEDEC	N/A			
D x E	12.00 mm x 9.00 mm PACKAGE			
SYMBOL	MIN	NOM	MAX	NOTE
A	---	---	1.20	PROFILE
A1	0.17	---	---	BALL HEIGHT
A2	0.81	---	0.97	BODY THICKNESS
D		12.00 BSC.		BODY SIZE
E		9.00 BSC.		BODY SIZE
D1		10.40 BSC.		MATRIX FOOTPRINT
E1		7.20 BSC.		MATRIX FOOTPRINT
MD		14		MATRIX SIZE D DIRECTION
ME		10		MATRIX SIZE E DIRECTION
n		137		BALL COUNT
ϕb	0.35	0.40	0.45	BALL DIAMETER
eE		0.80 BSC.		BALL PITCH
eD		0.80 BSC		BALL PITCH
SD / SE		0.40 BSC.		SOLDER BALL PLACEMENT
		G5,H5,H6		DEPOPULATED SOLDER BALLS

NOTES:

1. DIMENSIONING AND TOLERANCING METHODS PER ASME Y14.5M-1994.
2. ALL DIMENSIONS ARE IN MILLIMETERS.
3. BALL POSITION DESIGNATION PER JESD 95-1, SPP-010.
4. e REPRESENTS THE SOLDER BALL GRID PITCH.
5. SYMBOL "MD" IS THE BALL MATRIX SIZE IN THE "D" DIRECTION.

   SYMBOL "ME" IS THE BALL MATRIX SIZE IN THE "E" DIRECTION.

   n IS THE NUMBER OF POPULTED SOLDER BALL POSITIONS FOR MATRIX SIZE MD X ME.

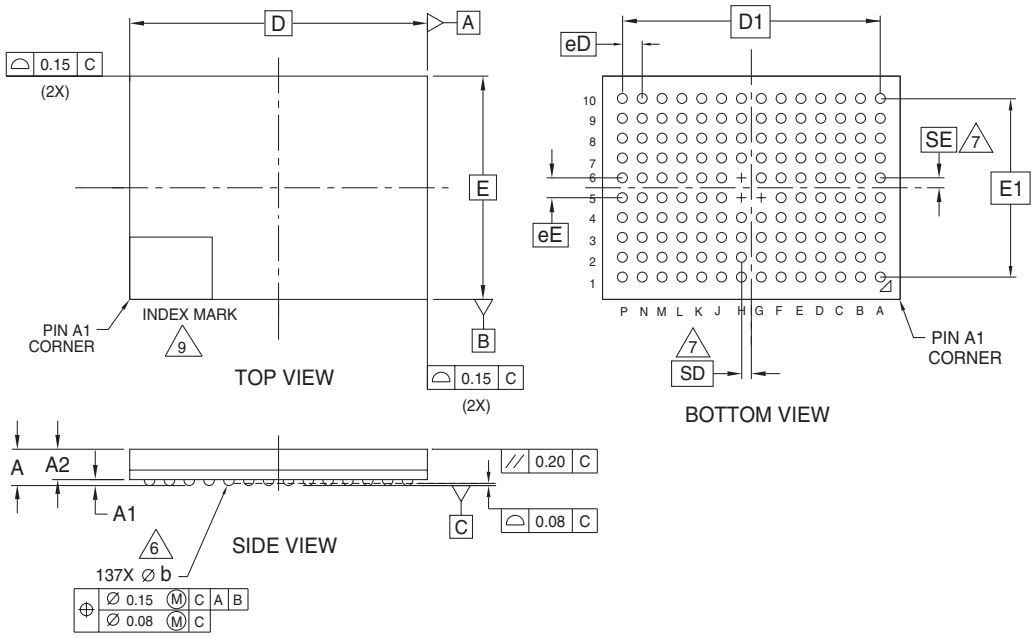
6. DIMENSION "b" IS MEASURED AT THE MAXIMUM BALL DIAMETER IN A PLANE PARALLEL TO DATUM C.
7. SD AND SE ARE MEASURED WITH RESPECT TO DATUMS A AND B AND DEFINE THE POSITION OF THE CENTER SOLDER BALL IN THE OUTER ROW.

   WHEN THERE IS AN ODD NUMBER OF SOLDER BALLS IN THE OUTER ROW SD OR SE = 0.000.

   WHEN THERE IS AN EVEN NUMBER OF SOLDER BALLS IN THE OUTER ROW, SD OR SE = e/2
8. "+" INDICATES THE THEORETICAL CENTER OF DEPOPULATED BALLS.
9. N/A
10. A1 CORNER TO BE IDENTIFIED BY CHAMFER, LASER OR INK MARK, METALLIZED MARK INDENTATION OR OTHER MEANS.

3393\ 16-038.22a

7.2 FTFI37—I37-ball Fine-Pitch Ball Grid Array (FBGA) 9 x I2.0 x I.4 mm Package



PACKAGE	FTF 137			NOTE
JEDEC	N/A			
D x E	12.00 mm x 9.00 mm PACKAGE			
SYMBOL	MIN	NOM	MAX	
A	---	---	1.40	PROFILE
A1	0.17	---	---	BALL HEIGHT
A2	1.02	---	1.17	BODY THICKNESS
D	12.00 BSC.			BODY SIZE
E	9.00 BSC.			BODY SIZE
D1	10.40 BSC.			MATRIX FOOTPRINT
E1	7.20 BSC.			MATRIX FOOTPRINT
MD	14			MATRIX SIZE D DIRECTION
ME	10			MATRIX SIZE E DIRECTION
n	137			BALL COUNT
Ø b	0.35	0.40	0.45	BALL DIAMETER
eE	0.80 BSC.			BALL PITCH
eD	0.80 BSC			BALL PITCH
SD / SE	0.40 BSC.			SOLDER BALL PLACEMENT
	G5,H5,H6			DEPOPULATED SOLDER BALLS

NOTES:

1. DIMENSIONING AND TOLERANCING METHODS PER ASME Y14.5M-1994.
2. ALL DIMENSIONS ARE IN MILLIMETERS.
3. BALL POSITION DESIGNATION PER JEP95, SECTION 4.3, SPP-010.
4. e REPRESENTS THE SOLDER BALL GRID PITCH.
5. SYMBOL "MD" IS THE BALL MATRIX SIZE IN THE "D" DIRECTION.

   SYMBOL "ME" IS THE BALL MATRIX SIZE IN THE "E" DIRECTION.

   n IS THE NUMBER OF POPULTED SOLDER BALL POSITIONS FOR MATRIX SIZE MD X ME.
6. DIMENSION "b" IS MEASURED AT THE MAXIMUM BALL DIAMETER IN A PLANE PARALLEL TO DATUM C.
7. SD AND SE ARE MEASURED WITH RESPECT TO DATUMS A AND B AND DEFINE THE POSITION OF THE CENTER SOLDER BALL IN THE OUTER ROW.

   WHEN THERE IS AN ODD NUMBER OF SOLDER BALLS IN THE OUTER ROW SD OR SE = 0.000.

   WHEN THERE IS AN EVEN NUMBER OF SOLDER BALLS IN THE OUTER ROW, SD OR SE = e/2
8. "+" INDICATES THE THEORETICAL CENTER OF DEPOPULATED BALLS.
9. A1 CORNER TO BE IDENTIFIED BY CHAMFER, LASER OR INK MARK, METALLIZED MARK INDENTATION OR OTHER MEANS.

3532 \ 16-038.21 \ 12.13.05

S29WS-N MirrorBit™ Flash Family

S29WS256N, S29WSI28N
256/128 Megabit (16/8 M x 16 bit) CMOS 1.8 Volt-only Simultaneous Read/Write, Burst-mode Flash Memory

Data Sheet

PRELIMINARY

General Description

The Spansion S29WS256/128 are Mirrorbit™ Flash products fabricated on 110 nm process technology. These burst mode Flash devices are capable of performing simultaneous read and write operations with zero latency on two separate banks using separate data and address pins. These products can operate up to 80 MHz and use a single V_{CC} of 1.7 V to 1.95 V that makes them ideal for today's demanding wireless applications requiring higher density, better performance and lowered power consumption.

Distinctive Characteristics

- **Single 1.8 V read/program/erase (1.70–1.95 V)**
- **110 nm MirrorBit™ Technology**
- **Simultaneous Read/Write operation with zero latency**
- **32-word Write Buffer**
- **Sixteen-bank architecture consisting of 16/8 Mwords for WS256N/128N, respectively**
- **Four 16 Kword sectors at both top and bottom of memory array**
- **254/126 64 Kword sectors (WS256N/128N)**
- **Programmable linear (8/16/32) with or without wrap around and continuous burst read modes**
- **Secured Silicon Sector region consisting of 128 words each for factory and customer**
- **20-year data retention (typical)**
- **Cycling Endurance: 100,000 cycles per sector (typical)**
- **RDY output indicates data available to system**
- **Command set compatible with JEDEC (42.4) standard**
- **Hardware (WP#) protection of top and bottom sectors**
- **Dual boot sector configuration (top and bottom)**
- **Low V_{CC} write inhibit**
- **Persistent and Password methods of Advanced Sector Protection**
- **Write operation status bits indicate program and erase operation completion**
- **Suspend and Resume commands for Program and Erase operations**
- **Unlock Bypass program command to reduce programming time**
- **Synchronous or Asynchronous program operation, independent of burst control register settings**
- **ACC input pin to reduce factory programming time**
- **Support for Common Flash Interface (CFI)**

Performance Characteristics

Read Access Times			
Speed Option (MHz)	**80**	**66**	**54**
Max. Synch. Latency, ns (t_{IACC})	80	80	80
Max. Synch. Burst Access, ns (t_{BACC})	9	11.2	13.5
Max. Asynch. Access Time, ns (t_{ACC})	80	80	80
Max. Asynch. Page Access Time, ns (t_{PACC})	20	20	20
Max CE# Access Time, ns (t_{CE})	80	80	80
Max OE# Access Time, ns (t_{OE})	13.5	13.5	13.5

Current Consumption (typical values)	
Continuous Burst Read @ 80 MHz	38 mA
Simultaneous Operation (asynchronous)	50 mA
Program (asynchronous)	19 mA
Erase (asynchronous)	19 mA
Standby Mode (asynchronous)	20 µA

Typical Program & Erase Times	
Single Word Programming	40 µs
Effective Write Buffer Programming (V_{CC}) Per Word	9.4 µs
Effective Write Buffer Programming (V_{ACC}) Per Word	6 µs
Sector Erase (16 Kword Sector)	150 ms
Sector Erase (64 Kword Sector)	600 ms

Publication Number S29WS-N_m0 **Revision** I **Amendment** 0 **Issue Date** December 3, 2005

8 Input/Output Descriptions & Logic Symbol

Table 8.1 identifies the input and output package connections provided on the device.

Table 8.1 Input/Output Descriptions

Symbol	Type	Description
A23–A0	Input	Address lines for WS256N (A22-A0 for WS128).
DQ15–DQ0	I/O	Data input/output.
CE#	Input	Chip Enable. Asynchronous relative to CLK.
OE#	Input	Output Enable. Asynchronous relative to CLK.
WE#	Input	Write Enable.
V_{CC}	Supply	Device Power Supply.
V_{SS}	I/O	Ground.
NC	No Connect	Not connected internally.
RDY	Output	Ready. Indicates when valid burst data is ready to be read.
CLK	Input	Clock Input. In burst mode, after the initial word is output, subsequent active edges of CLK increment the internal address counter.
AVD#	Input	Address Valid. Indicates to device that the valid address is present on the address inputs. When low during asynchronous mode, indicates valid address; when low during burst mode, causes starting address to be latched at the next active clock edge. When high, device ignores address inputs.
RESET#	Input	Hardware Reset. Low = device resets and returns to reading array data.
WP#	Input	Write Protect. At V_{IL}, disables program and erase functions in the four outermost sectors. Should be at V_{IH} for all other conditions.
ACC	Input	Acceleration Input. At V_{HH}, accelerates programming; automatically places device in unlock bypass mode. At V_{IL}, disables all program and erase functions. Should be at V_{IH} for all other conditions.
RFU	Reserved	Reserved for future use (see MCP look-ahead pinout for use with MCP).

9 Block Diagram

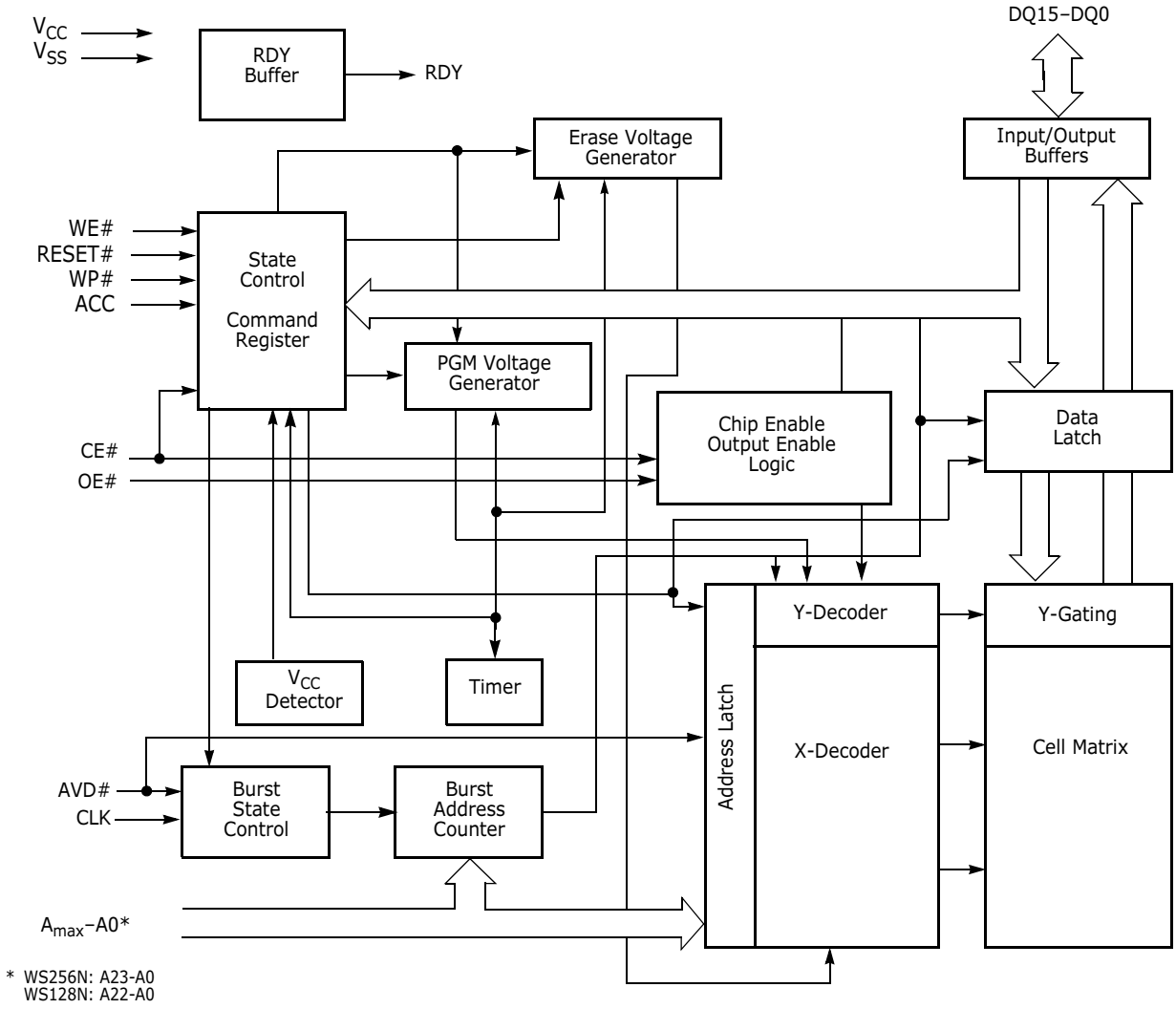


Figure 9.1. S29WS-N Block Diagram

10 Additional Resources

Visit www.spansion.com to obtain the following related documents:

Application Notes

- Using the Operation Status Bits in AMD Devices
- Understanding Burst Mode Flash Memory Devices
- Simultaneous Read/Write vs. Erase Suspend/Resume
- MirrorBit™ Flash Memory Write Buffer Programming and Page Buffer Read
- Design-In Scalable Wireless Solutions with Spansion Products
- Common Flash Interface Version 1.4 Vendor Specific Extensions

Specification Bulletins

Contact your local sales office for details.

Drivers and Software Support

- Spansion low-level drivers
- Enhanced Flash drivers
- Flash file system

CAD Modeling Support

- VHDL and Verilog
- IBIS
- ORCAD

Technical Support

Contact your local sales office or contact Spansion LLC directly for additional technical support:

US: (408) 749-5703
Japan (03) 5322-3324

Spansion LLC Locations

915 DeGuigne Drive, P.O. Box 3453
Sunnyvale, CA 94088-3453, USA
Telephone: 408-962-2500 or
1-866-SPANSION

Spansion Japan Limited
Cube-Kawasaki 9F/10F,
1-14 Nisshin-cho, Kawasaki-ku, Kawasaki-shi,
Kanagawa, 210-0024, Japan
Phone : 044-223-1700

II Product Overview

The S29WS-N family consists of 256, 128 Mbit, 1.8 volts-only, simultaneous read/write burst mode Flash device optimized for today's wireless designs that demand a large storage array, rich functionality, and low power consumption.

These devices are organized in 16 or 8 Mwords of 16 bits each and are capable of continuous, synchronous (burst) read or linear read (8-, 16-, or 32-word aligned group) with or without wrap around. These products also offer single word programming or a 32-word buffer for programming with program/erase and suspend functionality. Additional features include:

- Advanced Sector Protection methods for protecting sectors as required
- 256 words of Secured Silicon area for storing customer and factory secured information. The Secured Silicon Sector is One Time Programmable.

II.I Memory Map

The S29WS256/128N Mbit devices consist of 16 banks organized as shown in Table 11.1–Table 11.2.

Table II.I S29WS256N Sector & Memory Address Map

Bank Size	Sector Count	Sector Size (KB)	Bank	Sector/ Sector Range	Address Range	Notes
2 MB	4	32	0	SA000	000000h–003FFFh	Contains four smaller sectors at bottom of addressable memory.
				SA001	004000h–007FFFh	
				SA002	008000h–00BFFFh	
				SA003	00C000h–00FFFFh	
	15	128		SA004 to SA018	010000h–01FFFFh to 0F0000h–0FFFFFh	All 128 KB sectors. Pattern for sector address range is xx0000h–xxFFFFh. (see note)
2 MB	16	128	1	SA019 to SA034	100000h–10FFFFh to 1F0000h–1FFFFFh	
2 MB	16	128	2	SA035 to SA050	200000h–20FFFFh to 2F0000h–2FFFFFh	
2 MB	16	128	3	SA051 to SA066	300000h–30FFFFh to 3F0000h–3FFFFFh	
2 MB	16	128	4	SA067 to SA082	400000h–40FFFFh to 4F0000h–4FFFFFh	
2 MB	16	128	5	SA083 to SA098	500000h–50FFFFh to 5F0000h–5FFFFFh	
2 MB	16	128	6	SA099 to SA114	600000h–60FFFFh to 6F0000h–6FFFFFh	
2 MB	16	128	7	SA115 to SA130	700000h–70FFFFh to 7F0000h–7FFFFFh	
2 MB	16	128	8	SA131 to SA146	800000h–80FFFFh to 8F0000h–8FFFFFh	
2 MB	16	128	9	SA147 to SA162	900000h–90FFFFh to 9F0000h–9FFFFFh	
2 MB	16	128	10	SA163 to SA178	A00000h–A0FFFFh to AF0000h–AFFFFFh	
2 MB	16	128	11	SA179 to SA194	B00000h–B0FFFFh to BF0000h–BFFFFFh	
2 MB	16	128	12	SA195 to SA210	C00000h–C0FFFFh to CF0000h–CFFFFFh	
2 MB	16	128	13	SA211 to SA226	D00000h–D0FFFFh to DF0000h–DFFFFFh	
2 MB	16	128	14	SA227 to SA242	E00000h–E0FFFFh to EF0000h–EFFFFFh	
2 MB	15	128	15	SA243 to SA257	F00000h–F0FFFFh to FE0000h–FEFFFFh	
	4	32		SA258	FF0000h–FF3FFFh	Contains four smaller sectors at top of addressable memory.
				SA259	FF4000h–FF7FFFh	
				SA260	FF8000h–FFBFFFh	
				SA261	FFC000h–FFFFFFh	

Note: *This table has been condensed to show sector-related information for an entire device on a single page. Sectors and their address ranges that are not explicitly listed (such as SA005–SA017) have sector starting and ending addresses that form the same pattern as all other sectors of that size. For example, all 128 KB sectors have the pattern xx00000h–xxFFFFh.*

Table 11.2. S29WS128N Sector & Memory Address Map

Bank Size	Sector Count	Sector Size (KB)	Bank	Sector/ Sector Range	Address Range	Notes
1 MB	4	32	0	SA000	000000h–003FFFh	Contains four smaller sectors at bottom of addressable memory.
		32		SA001	004000h–007FFFh	
		32		SA002	008000h–00BFFFh	
		32		SA003	00C000h–00FFFFh	
	7	128		SA004 to SA010	010000h–01FFFFh to 070000h–07FFFFh	All 128 KB sectors. Pattern for sector address range is xx0000h–xxFFFFh. (see note)
1 MB	8	128	1	SA011 to SA018	080000h–08FFFFh to 0F0000h–0FFFFFh	
1 MB	8	128	2	SA019 to SA026	100000h–10FFFFh to 170000h–17FFFFh	
1 MB	8	128	3	SA027 to SA034	180000h–18FFFFh to 1F0000h–1FFFFFh	
1 MB	8	128	4	SA035 to SA042	200000h–20FFFFh to 270000h–27FFFFh	
1 MB	8	128	5	SA043 to SA050	280000h–28FFFFh to 2F0000h–2FFFFFh	
1 MB	8	128	6	SA051 to SA058	300000h–30FFFFh to 370000h–37FFFFh	
1 MB	8	128	7	SA059 to SA066	380000h–38FFFFh to 3F0000h–3FFFFFh	
1 MB	8	128	8	SA067 to SA074	400000h–40FFFFh to 470000h–47FFFFh	
1 MB	8	128	9	SA075 to SA082	480000h–48FFFFh to 4F0000h–4FFFFFh	
1 MB	8	128	10	SA083 to SA090	500000h–50FFFFh to 570000h–57FFFFh	
1 MB	8	128	11	SA091 to SA098	580000h–58FFFFh to 5F0000h–5FFFFFh	
1 MB	8	128	12	SA099 to SA106	600000h–60FFFFh to 670000h–67FFFFh	
1 MB	8	128	13	SA107 to SA114	680000h–68FFFFh to 6F0000h–6FFFFFh	
1 MB	8	128	14	SA115 to SA122	700000h–70FFFFh to 770000h–77FFFFh	
1 MB	7	128	15	SA123 to SA129	780000h–78FFFFh to 7E0000h–7EFFFFh	
	4	32		SA130	7F0000h–7F3FFFh	Contains four smaller sectors at top of addressable memory.
		32		SA131	7F4000h–7F7FFFh	
		32		SA132	7F8000h–7FBFFFh	
		32		SA133	7FC000h–7FFFFFh	

Note: *This table has been condensed to show sector-related information for an entire device on a single page. Sectors and their address ranges that are not explicitly listed (such as SA005–SA009) have sector starting and ending addresses that form the same pattern as all other sectors of that size. For example, all 128 KB sectors have the pattern xx00000h–xxFFFFh.*

12 Device Operations

This section describes the read, program, erase, simultaneous read/write operations, handshaking, and reset features of the Flash devices.

Operations are initiated by writing specific commands or a sequence with specific address and data patterns into the command registers (see Tables 17.1 and 17.2). The command register itself does not occupy any addressable memory location; rather, it is composed of latches that store the commands, along with the address and data information needed to execute the command. The contents of the register serve as input to the internal state machine and the state machine outputs dictate the function of the device. Writing incorrect address and data values or writing them in an improper sequence may place the device in an unknown state, in which case the system must write the reset command to return the device to the reading array data mode.

12.1 Device Operation Table

The device must be setup appropriately for each operation. Table 12.1 describes the required state of each control pin for any particular operation.

Table 12.1 Device Operations

Operation	CE#	OE#	WE#	Addresses	DQ15–0	RESET#	CLK	AVD#
Asynchronous Read - Addresses Latched	L	L	H	Addr In	Data Out	H	X	⎍
Asynchronous Read - Addresses Steady State	L	L	H	Addr In	Data Out	H	X	L
Asynchronous Write	L	H	L	Addr In	I/O	H	X	L
Synchronous Write	L	H	L	Addr In	I/O	H	⎍	⎍
Standby (CE#)	H	X	X	X	HIGH Z	H	X	X
Hardware Reset	X	X	X	X	HIGH Z	L	X	X
Burst Read Operations (Synchronous)								
Load Starting Burst Address	L	X	H	Addr In	X	H	⎍	⎍
Advance Burst to next address with appropriate Data presented on the Data Bus	L	L	H	X	Burst Data Out	H	⎍	H
Terminate current Burst read cycle	H	X	H	X	HIGH Z	H	⎍	X
Terminate current Burst read cycle via RESET#	X	X	H	X	HIGH Z	L	X	X
Terminate current Burst read cycle and start new Burst read cycle	L	X	H	Addr In	I/O	H	⎍	⎍

Legend: *L = Logic 0, H = Logic 1, X = Don't Care, I/O = Input/Output.*

12.2 Asynchronous Read

All memories require access time to output array data. In an asynchronous read operation, data is read from one memory location at a time. Addresses are presented to the device in random order, and the propagation delay through the device causes the data on its outputs to arrive asynchronously with the address on its inputs.

The device defaults to reading array data asynchronously after device power-up or hardware reset. To read data from the memory array, the system must first assert a valid address on A_{max}–A0, while driving AVD# and CE# to V_{IL}. WE# must remain at V_{IH}. The rising edge of AVD# latches the address. The OE# signal must be driven to V_{IL}, once AVD# has been driven to V_{IH}. Data is output on A/DQ15-A/DQ0 pins after the access time (t_{OE}) has elapsed from the falling edge of OE#.

12.3 Page Read Mode

The device is capable of fast page mode read and is compatible with the page mode Mask ROM read operation. This mode provides faster read access speed for random locations within a page. The random or initial page access is t_{ACC} or t_{CE} and subsequent page read accesses (as long as the locations specified by the microprocessor falls within that page) is equivalent to t_{PACC}. When CE# is deasserted (= V_{IH}), the reassertion of CE# for subsequent access has access time of t_{ACC} or t_{CE}. Here again, CE# selects the device and OE# is the output control and should be used to gate data to the output inputs if the device is selected. Fast page mode accesses are obtained by keeping A_{max} – A2 constant and changing A1 – A0 to select the specific word within that page.

Address bits A_{max} – A2 select a 4-word page, and address bits A1 – A0 select a specific word within that page. This is an asynchronous operation with the microprocessor supplying the specific word location. See Table 12.2 for details on selecting specific words.

The device is automatically set to reading array data after device power-up. No commands are required to retrieve data. Each bank is ready to read array data after completing an Embedded Program or Embedded Erase algorithm.

Reads from the memory array may be performed in conjunction with the Erase Suspend and Program Suspend features. After the device accepts an Erase Suspend command, the corresponding bank enters the erase-suspend-read mode, after which the system can read data from any non-erase-suspended sector within the same bank. The system can read array data using the standard read timing, except that if it reads at an address within erase-suspended sectors, the device outputs status data. After completing a programming operation in the Erase Suspend mode, the system may once again read array data with the same exception. After the device accepts a Program Suspend command, the corresponding bank enters the program-suspend-read mode, after which the system can read data from any non-program-suspended sector within the same bank.

The de-assertion and re-assertion of AVD# creates a new t_{ACC}. The user must keep AVD# low during and between page reads on address A(1:0).

During Simultaneous Operation (SO), the user needs to de-assert and re- assert either /CE# or /AVD# when performing data polling to SO read.

Table 12.2 Word Selection within a Page

Word	A1	A0
Word 0	0	0
Word 1	0	1
Word 2	1	0
Word 3	1	1

12.4 Synchronous (Burst) Read Mode & Configuration Register

When a series of adjacent addresses needs to be read from the device (in order from lowest to highest address), the synchronous (or burst read) mode can be used to significantly reduce the overall time needed for the device to output array data. After an initial access time required for the data from the first address location, subsequent data is output synchronized to a clock input provided by the system.

The device offers both continuous and linear methods of burst read operation, which are discussed in subsections 12.4.1 and 12.4.2, and 12.4.3.

Since the device defaults to asynchronous read mode after power-up or a hardware reset, the configuration register must be set to enable the burst read mode. Other Configuration Register settings include the number of wait states to insert before the initial word (t_{IACC}) of each burst

access, the burst mode in which to operate, and when RDY indicates data is ready to be read. Prior to entering the burst mode, the system should first determine the configuration register settings (and read the current register settings if desired via the Read Configuration Register command sequence), and then write the configuration register command sequence. See Section 12.4.4, Configuration Register, and Table 17.1, Memory Array Commands for further details.

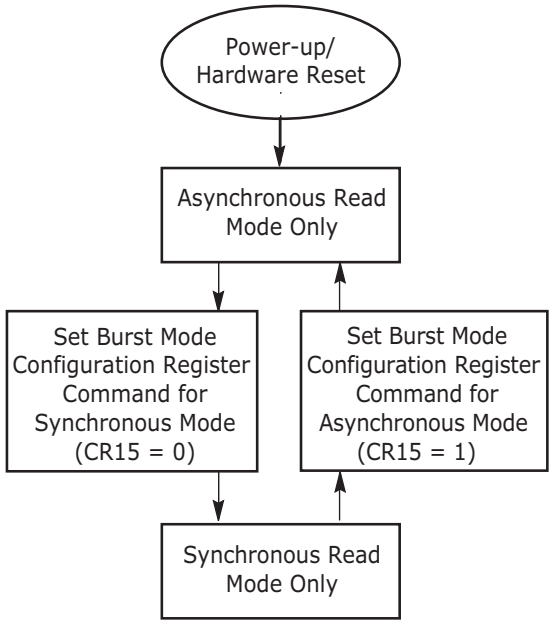


Figure I2.I. Synchronous/Asynchronous State Diagram

The device outputs the initial word subject to the following operational conditions:

- t_{IACC} specification: the time from the rising edge of the first clock cycle after addresses are latched to valid data on the device outputs.
- configuration register setting CR13–CR11: the total number of clock cycles (wait states) that occur before valid data appears on the device outputs. The effect is that t_{IACC} is lengthened.

The device outputs subsequent words t_{BACC} after the active edge of each successive clock cycle, which also increments the internal address counter. The device outputs burst data at this rate subject to the following operational conditions:

- starting address: whether the address is divisible by four (where A[1:0] is 00). A divisible-by-four address incurs the least number of additional wait states that occur after the initial word. The number of additional wait states required increases for burst operations in which the starting address is one, two, or three locations above the divisible-by-four address (i.e., where A[1:0] is 01, 10, or 11).
- boundary crossing: There is a boundary at every 128 words due to the internal architecture of the device. One additional wait state must be inserted when crossing this boundary if the memory bus is operating at a high clock frequency. Please refer to the tables below.
- clock frequency: the speed at which the device is expected to burst data. Higher speeds require additional wait states after the initial word for proper operation.

In all cases, with or without latency, the RDY output indicates when the next data is available to be read.

Tables 12.3-12.8 reflect wait states required for S29WS256/128N devices. Refer to the Configuration Register table (CR11 - CR14) and timing diagrams for more details.

Table 12.3 Address Latency (S29WS256N)

Word	Wait States	Cycle								
0	x ws	D0	D1	D2	D3	D4	D5	D6	D7	D8
1	x ws	D1	D2	D3	1 ws	D4	D5	D6	D7	D8
2	x ws	D2	D3	1 ws	1 ws	D4	D5	D6	D7	D8
3	x ws	D3	1 ws	1 ws	1 ws	D4	D5	D6	D7	D8

Table 12.4 Address Latency (S29WS128N)

Word	Wait States	Cycle								
0	5, 6, 7 ws	D0	D1	D2	D3	D4	D5	D6	D7	D8
1	5, 6, 7 ws	D1	D2	D3	1 ws	D4	D5	D6	D7	D8
2	5, 6, 7 ws	D2	D3	1 ws	1 ws	D4	D5	D6	D7	D8
3	5, 6, 7 ws	D3	1 ws	1 ws	1 ws	D4	D5	D6	D7	D8

Table 12.5 Address/Boundary Crossing Latency (S29WS256N @ 80MHz)

Word	Wait States	Cycle								
0	7 ws	D0	D1	D2	D3	1 ws	1 ws	D4	D5	D6
1	7 ws	D1	D2	D3	1 ws	1 ws	1 ws	D4	D5	D6
2	7 ws	D2	D3	1 ws	1 ws	1 ws	1 ws	D4	D5	D6
3	7 ws	D3	1 ws	1 ws	1 ws	1 ws	1 ws	D4	D5	D6

Table 12.6 Address/Boundary Crossing Latency (S29WS256N @ 66 MHz)

Word	Wait States	Cycle								
0	6 ws	D0	D1	D2	D3	1 ws	D4	D5	D6	D7
1	6 ws	D1	D2	D3	1 ws	1 ws	D4	D5	D6	D7
2	6 ws	D2	D3	1 ws	1 ws	1 ws	D4	D5	D6	D7
3	6 ws	D3	1 ws	1 ws	1 ws	1 ws	D4	D5	D6	D7

Table 12.7 Address/Boundary Crossing Latency (S29WS256N @ 54MHz)

Word	Wait States	Cycle								
0	5 ws	D0	D1	D2	D3	D4	D5	D6	D7	D8
1	5 ws	D1	D2	D3	1 ws	D4	D5	D6	D7	D8
2	5 ws	D2	D3	1 ws	1 ws	D4	D5	D6	D7	D8
3	5 ws	D3	1 ws	1 ws	1 ws	D4	D5	D6	D7	D8

Table 12.8 Address/Boundary Crossing Latency (S29WS128N)

Word	Wait States	Cycle								
0	5, 6, 7 ws	D0	D1	D2	D3	1 ws	D4	D5	D6	D7
1	5, 6, 7 ws	D1	D2	D3	1 ws	1 ws	D4	D5	D6	D7
2	5, 6, 7 ws	D2	D3	1 ws	1 ws	1 ws	D4	D5	D6	D7
3	5, 6, 7 ws	D3	1 ws	1 ws	1 ws	1 ws	D4	D5	D6	D7

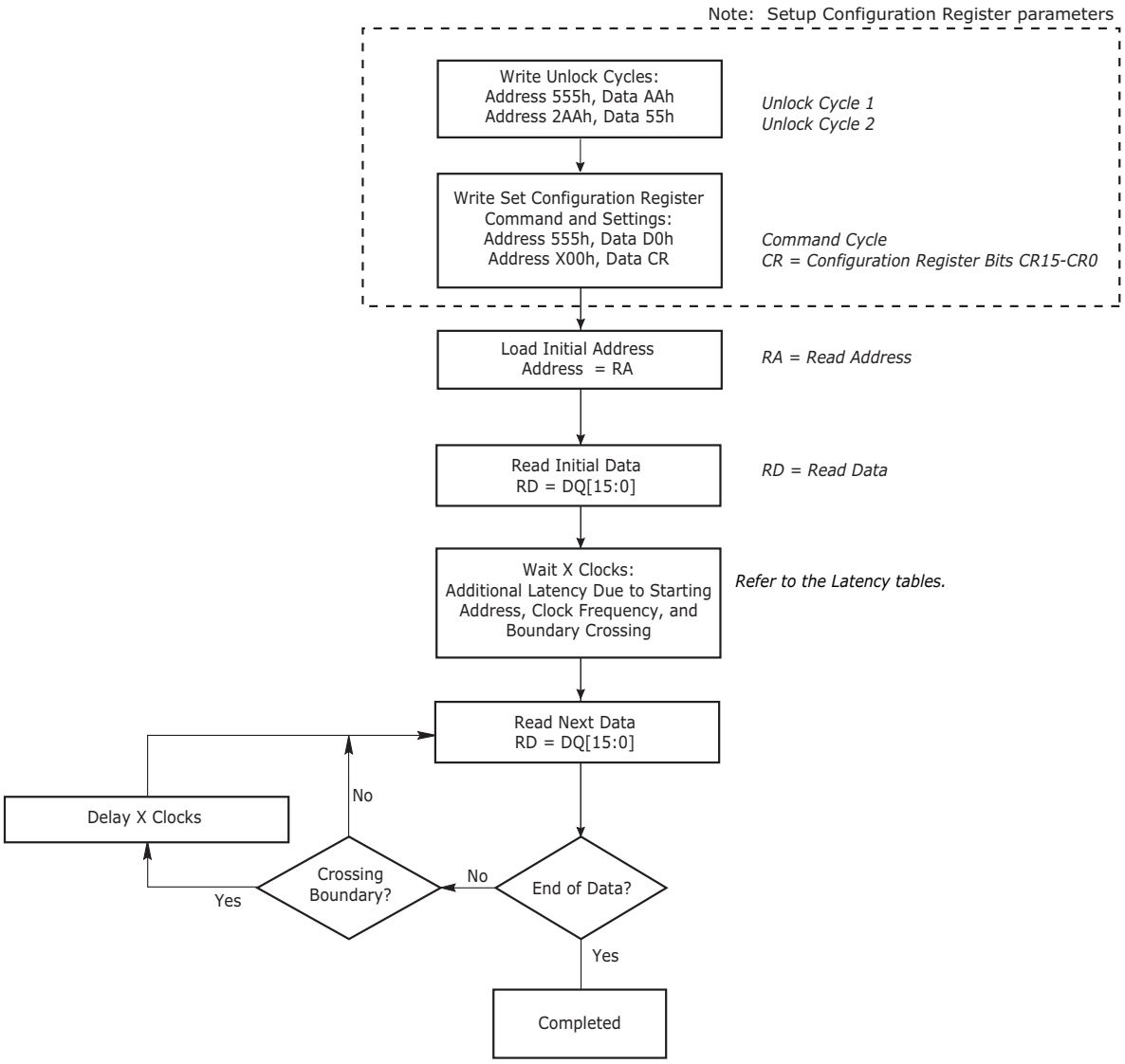


Figure I2.2. Synchronous Read

12.4.1 Continuous Burst Read Mode

In the continuous burst read mode, the device outputs sequential burst data from the starting address given and then wrap around to address 000000h when it reaches the highest addressable memory location. The burst read mode continues until the system drives CE# high, or RESET= V_{IL}. Continuous burst mode can also be aborted by asserting AVD# low and providing a new address to the device.

If the address being read crosses a 128-word line boundary (as mentioned above) and the subsequent word line is not being programmed or erased, additional latency cycles are required as reflected by the configuration register table (Table 12.10).

If the address crosses a bank boundary while the subsequent bank is programming or erasing, the device provides read status information and the clock is ignored. Upon completion of status read or program or erase operation, the host can restart a burst read operation using a new address and AVD# pulse.

12.4.2 8-, 16-, 32-Word Linear Burst Read with Wrap Around

In a linear burst read operation, a fixed number of words (8, 16, or 32 words) are read from consecutive addresses that are determined by the group within which the starting address falls. The groups are sized according to the number of words read in a single burst sequence for a given mode (see Table 12.9).

For example, if the starting address in the 8-word mode is 3Ch, the address range to be read would be 38-3Fh, and the burst sequence would be 3C-3D-3E-3F-38-39-3A-3Bh. Thus, the device outputs all words in that burst address group until all word are read, regardless of where the starting address occurs in the address group, and then terminates the burst read.

In a similar fashion, the 16-word and 32-word Linear Wrap modes begin their burst sequence on the starting address provided to the device, then wrap back to the first address in the selected address group.

Note that in this mode the address pointer does not cross the boundary that occurs every 128 words; thus, no additional wait states are inserted due to boundary crossing.

Table 12.9 Burst Address Groups

Mode	Group Size	Group Address Ranges
8-word	8 words	0-7h, 8-Fh, 10-17h,...
16-word	16 words	0-Fh, 10-1Fh, 20-2Fh,...
32-word	32 words	00-1Fh, 20-3Fh, 40-5Fh,...

12.4.3 8-, 16-, 32-Word Linear Burst without Wrap Around

If wrap around is not enabled for linear burst read operations, the 8-word, 16-word, or 32-word burst executes up to the maximum memory address of the selected number of words. The burst stops after 8, 16, or 32 addresses and does not wrap around to the first address of the selected group.

For example, if the starting address in the 8- word mode is 3Ch, the address range to be read would be 39-40h, and the burst sequence would be 3C-3D-3E-3F-40-41-42-43h if wrap around is not enabled. The next address to be read requires a new address and AVD# pulse. Note that in this burst read mode, the address pointer may cross the boundary that occurs every 128 words, which will incur the additional boundary crossing wait state.

12.4.4 Configuration Register

The configuration register sets various operational parameters associated with burst mode. Upon power-up or hardware reset, the device defaults to the asynchronous read mode, and the configuration register settings are in their default state. The host system should determine the proper settings for the entire configuration register, and then execute the Set Configuration Register command sequence, before attempting burst operations. The configuration register is not reset after deasserting CE#. The Configuration Register can also be read using a command sequence (see Table 17.1). The following list describes the register settings.

Table 12.10 Configuration Register

CR Bit	Function					Settings (Binary)
CR15	Set Device Read Mode					0 = Synchronous Read (Burst Mode) Enabled 1 = Asynchronous Read Mode (default) Enabled
CR14	Reserved					1 = S29WS256N at 6 or 7 Wait State setting 0 = All others
			54 MHz	**66 Mhz**	**80 MHz**	
CR13	Programmable Wait State	S29WS128N	0	1	1	011 = Data valid on 5th active CLK edge after addresses latched 100 = Data valid on 6th active CLK edge after addresses latched 101 = Data valid on 7th active CLK edge after addresses latched (default) 110 = Reserved 111 = Reserved Inserts wait states before initial data is available. Setting greater number of wait states before initial data reduces latency after initial data. (Notes 1, 2)
		S29WS256N				
CR12		S29WS128N	1	0	0	
		S29WS256N				
CR11		S29WS128N	1	0	1	
		S29WS256N				
CR10	RDY Polarity					0 = RDY signal active low 1 = RDY signal active high (default)
CR9	Reserved					1 = default
CR8	RDY					0 = RDY active one clock cycle before data 1 = RDY active with data (default) When CR13-CR11 are set to 000, RDY is active with data regardless of CR8 setting.
CR7	Reserved					1 = default
CR6	Reserved					1 = default
CR5	Reserved					0 = default
CR4	Reserved					0 = default
CR3	Burst Wrap Around					0 = No Wrap Around Burst 1 = Wrap Around Burst (default) Ignored if in continuous mode
CR2 CR1 CR0	Burst Length					000 = Continuous (default) 010 = 8-Word Linear Burst 011 = 16-Word Linear Burst 100 = 32-Word Linear Burst (All other bit settings are reserved)

Notes:

1. *Refer to Tables 12.3 - 12.8 for wait states requirements.*
2. *Refer to Synchronous Burst Read timing diagrams*
3. *Configuration Register is in the default state upon power-up or hardware reset.*

Reading the Configuration Table. The configuration register can be read with a four-cycle command sequence. See Table 17.1 for sequence details. A software reset command is required after reading or setting the configuration register to set the device into the correct state.

12.5 Autoselect

The Autoselect is used for manufacturer ID, Device identification, and sector protection information. This mode is primarily intended for programming equipment to automatically match a device with its corresponding programming algorithm. The Autoselect codes can also be accessed in-system. When verifying sector protection, the sector address must appear on the appropriate highest order address bits (see Table 12.11). The remaining address bits are don't care. The most significant four bits of the address during the third write cycle selects the bank from which the Autoselect codes are read by the host. All other banks can be accessed normally for data read without exiting the Autoselect mode.

- To access the Autoselect codes, the host system must issue the Autoselect command.
- The Autoselect command sequence may be written to an address within a bank that is either in the read or erase-suspend-read mode.
- The Autoselect command may not be written while the device is actively programming or erasing. Autoselect does not support simultaneous operations or burst mode.
- The system must write the reset command to return to the read mode (or erase-suspend-read mode if the bank was previously in Erase Suspend).

See Table 17.1 for command sequence details.

Table 12.11 Autoselect Addresses

Description	Address	Read Data
Manufacturer ID	(BA) + 00h	0001h
Device ID, Word 1	(BA) + 01h	227Eh
Device ID, Word 2	(BA) + 0Eh	2230 (WS256N) 2231 (WS128N)
Device ID, Word 3	(BA) + 0Fh	2200
Indicator Bits (See Note)	(BA) + 03h	DQ15 - DQ8 = Reserved DQ7 (Factory Lock Bit): 1 = Locked, 0 = Not Locked DQ6 (Customer Lock Bit): 1 = Locked, 0 = Not Locked DQ5 (Handshake Bit): 1 = Reserved, 0 = Standard Handshake DQ4, DQ3 (WP# Protection Boot Code): 00 = WP# Protects both Top Boot and Bottom Boot Sectors. 01, 10, 11 = Reserved DQ2 = Reserved DQ1 (DYB Power up State [Lock Register DQ4]): 1 = Unlocked (user option), 0 = Locked (default) DQ0 (PPB Eraseability [Lock Register DQ3]): 1 = Erase allowed, 0 = Erase disabled
Sector Block Lock/ Unlock	(SA) + 02h	0001h = Locked, 0000h = Unlocked

Note: *For WS128N and WS064, DQ1 and DQ0 are reserved.*

Software Functions and Sample Code

Table 12.12 Autoselect Entry

(LLD Function = lld_AutoselectEntryCmd)

Cycle	Operation	Byte Address	Word Address	Data
Unlock Cycle 1	Write	BAxAAAh	BAx555h	0x00AAh
Unlock Cycle 2	Write	BAx555h	BAx2AAh	0x0055h
Autoselect Command	Write	BAxAAAh	BAx555h	0x0090h

Table 12.13 Autoselect Exit

(LLD Function = lld_AutoselectExitCmd)

Cycle	Operation	Byte Address	Word Address	Data
Unlock Cycle 1	Write	base + XXXh	base + XXXh	0x00F0h

Notes:

1. *Any offset within the device works.*
2. *BA = Bank Address. The bank address is required.*
3. *base = base address.*

The following is a C source code example of using the autoselect function to read the manufacturer ID. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Here is an example of Autoselect mode (getting manufacturer ID) */
/* Define UINT16 example: typedef unsigned short UINT16; */

UINT16 manuf_id;

/* Auto Select Entry */

*( (UINT16 *)bank_addr + 0x555 ) = 0x00AA; /* write unlock cycle 1 */
*( (UINT16 *)bank_addr + 0x2AA ) = 0x0055; /* write unlock cycle 2 */
*( (UINT16 *)bank_addr + 0x555 ) = 0x0090; /* write autoselect command */


/* multiple reads can be performed after entry */

manuf_id = *( (UINT16 *)bank_addr + 0x000 ); /* read manuf. id */

/*  Autoselect exit */

*( (UINT16 *)base_addr + 0x000 ) = 0x00F0; /* exit autoselect (write reset command) */
```

12.6 Program/Erase Operations

These devices are capable of several modes of programming and or erase operations which are described in detail in the following sections. However, prior to any programming and or erase operation, devices must be setup appropriately as outlined in the configuration register (Table 12.9).

For any program and or erase operations, including writing command sequences, the system must drive AVD# and CE# to V_{IL}, and OE# to V_{IH} when providing an address to the device, and drive WE# and CE# to V_{IL}, and OE# to V_{IH} when writing commands or programming data.

Addresses are latched on the last falling edge of WE# or CE#, while data is latched on the 1st rising edge of WE# or CE#.

Note the following:

- When the Embedded Program algorithm is complete, the device returns to the read mode.
- The system can determine the status of the program operation by using DQ7 or DQ6. Refer to the Write Operation Status section for information on these status bits.
- A "0" cannot be programmed back to a "1." Attempting to do so causes the device to set DQ5 = 1 (halting any further operation and requiring a reset command). A succeeding read shows that the data is still "0." Only erase operations can convert a "0" to a "1."
- Any commands written to the device during the Embedded Program Algorithm are ignored except the Program Suspend command.
- Secured Silicon Sector, Autoselect, and CFI functions are unavailable when a program operation is in progress.
- A hardware reset immediately terminates the program operation and the program command sequence should be reinitiated once the device has returned to the read mode, to ensure data integrity.
- Programming is allowed in any sequence and across sector boundaries for single word programming operation.

12.6.1 Single Word Programming

Single word programming mode is the simplest method of programming. In this mode, four Flash command write cycles are used to program an individual Flash address. The data for this programming operation could be 8-, 16- or 32-bits wide. While this method is supported by all Spansion devices, in general it is not recommended for devices that support Write Buffer Programming. See Table 17.1 for the required bus cycles and Figure 12.3 for the flowchart.

When the Embedded Program algorithm is complete, the device then returns to the read mode and addresses are no longer latched. The system can determine the status of the program operation by using DQ7 or DQ6. Refer to the Write Operation Status section for information on these status bits.

- During programming, any command (except the Suspend Program command) is ignored.
- The Secured Silicon Sector, Autoselect, and CFI functions are unavailable when a program operation is in progress.

- A hardware reset immediately terminates the program operation. The program command sequence should be reinitiated once the device has returned to the read mode, to ensure data integrity.

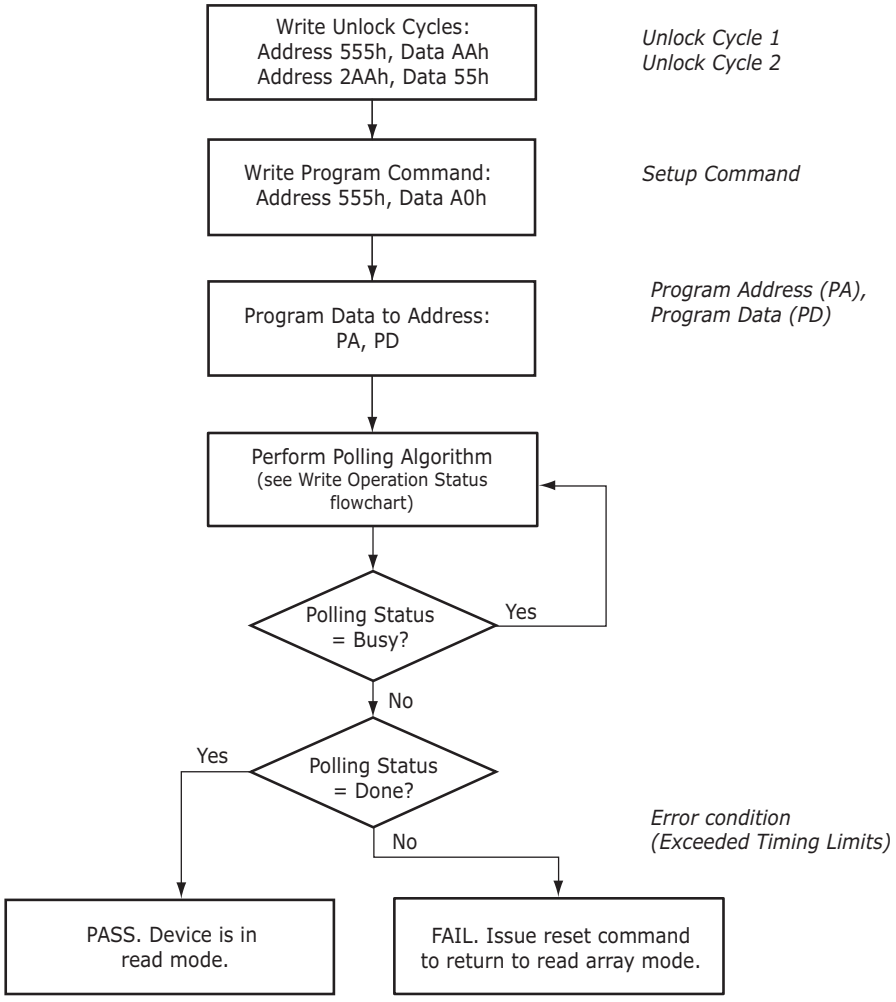


Figure I2.3. Single Word Program

Software Functions and Sample Code

Table 12.14. Single Word Program

(LLD Function = lld_ProgramCmd)

Cycle	Operation	Byte Address	Word Address	Data
Unlock Cycle 1	Write	Base + AAAh	Base + 555h	00AAh
Unlock Cycle 2	Write	Base + 554h	Base + 2AAh	0055h
Program Setup	Write	Base + AAAh	Base + 555h	00A0h
Program	Write	Word Address	Word Address	Data Word

Note: *Base = Base Address.*

The following is a C source code example of using the single word program function. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: Program Command    */
  *( (UINT16 *)base_addr + 0x555 ) = 0x00AA;   /* write unlock cycle 1            */
  *( (UINT16 *)base_addr + 0x2AA ) = 0x0055;   /* write unlock cycle 2            */
  *( (UINT16 *)base_addr + 0x555 ) = 0x00A0;   /* write program setup command     */
  *( (UINT16 *)pa )                = data;     /* write data to be programmed     */
/* Poll for program completion */
```

12.6.2 Write Buffer Programming

Write Buffer Programming allows the system to write a maximum of 32 words in one programming operation. This results in a faster effective word programming time than the standard "word" programming algorithms. The Write Buffer Programming command sequence is initiated by first writing two unlock cycles. This is followed by a third write cycle containing the Write Buffer Load command written at the Sector Address in which programming occurs. At this point, the system writes the number of "word locations minus 1" that are loaded into the page buffer at the Sector Address in which programming occurs. This tells the device how many write buffer addresses are loaded with data and therefore when to expect the "Program Buffer to Flash" confirm command. The number of locations to program cannot exceed the size of the write buffer or the operation aborts. (Number loaded = the number of locations to program minus 1. For example, if the system programs 6 address locations, then 05h should be written to the device.)

The system then writes the starting address/data combination. This starting address is the first address/data pair to be programmed, and selects the "write-buffer-page" address. All subsequent address/data pairs must fall within the elected-write-buffer-page.

The "write-buffer-page" is selected by using the addresses A_{MAX} - A5.

The "write-buffer-page" addresses must be the same for all address/data pairs loaded into the write buffer. (This means Write Buffer Programming cannot be performed across multiple "write-buffer-pages." This also means that Write Buffer Programming cannot be performed across multiple sectors. If the system attempts to load programming data outside of the selected "write-buffer-page", the operation ABORTs.)

After writing the Starting Address/Data pair, the system then writes the remaining address/data pairs into the write buffer.

Note that if a Write Buffer address location is loaded multiple times, the "address/data pair" counter is decremented for every data load operation. Also, the last data loaded at a location before the "Program Buffer to Flash" confirm command is programmed into the device. It is the software's responsibility to comprehend ramifications of loading a write-buffer location more than once. The counter decrements for each data load operation, NOT for each unique write-

buffer-address location. Once the specified number of write buffer locations have been loaded, the system must then write the "Program Buffer to Flash" command at the Sector Address. Any other address/data write combinations abort the Write Buffer Programming operation. The device goes "busy." The Data Bar polling techniques should be used while monitoring the last address location loaded into the write buffer. This eliminates the need to store an address in memory because the system can load the last address location, issue the program confirm command at the last loaded address location, and then data bar poll at that same address. DQ7, DQ6, DQ5, DQ2, and DQ1 should be monitored to determine the device status during Write Buffer Programming.

The write-buffer "embedded" programming operation can be suspended using the standard suspend/resume commands. Upon successful completion of the Write Buffer Programming operation, the device returns to READ mode.

The Write Buffer Programming Sequence is ABORTED under any of the following conditions:

- Load a value that is greater than the page buffer size during the "Number of Locations to Program" step.
- Write to an address in a sector different than the one specified during the Write-Buffer-Load command.
- Write an Address/Data pair to a different write-buffer-page than the one selected by the "Starting Address" during the "write buffer data loading" stage of the operation.
- Write data other than the "Confirm Command" after the specified number of "data load" cycles.

The ABORT condition is indicated by DQ1 = 1, DQ7 = DATA# (for the "last address location loaded"), DQ6 = TOGGLE, DQ5 = 0. This indicates that the Write Buffer Programming Operation was ABORTED. A "Write-to-Buffer-Abort reset" command sequence is required when using the write buffer Programming features in Unlock Bypass mode. Note that the Secured Silicon sector, autoselect, and CFI functions are unavailable when a program operation is in progress.

Write buffer programming is allowed in any sequence of memory (or address) locations. These flash devices are capable of handling multiple write buffer programming operations on the same write buffer address range without intervening erases.

Use of the write buffer is strongly recommended for programming when multiple words are to be programmed. Write buffer programming is approximately eight times faster than programming one word at a time.

Software Functions and Sample Code

Table I2.I5. Write Buffer Program

(LLD Functions Used = lld_WriteToBufferCmd, lld_ProgramBufferToFlashCmd)

Cycle	Description	Operation	Byte Address	Word Address	Data
1	Unlock	Write	Base + AAAh	Base + 555h	00AAh
2	Unlock	Write	Base + 554h	Base + 2AAh	0055h
3	Write Buffer Load Command	Write	Program Address		0025h
4	Write Word Count	Write	Program Address		Word Count (N–1)h
Number of words (N) loaded into the write buffer can be from 1 to 32 words.					
5 to 36	Load Buffer Word N	Write	Program Address, Word N		Word N
Last	Write Buffer to Flash	Write	Sector Address		0029h

Notes:

1. *Base = Base Address.*
2. *Last = Last cycle of write buffer program operation; depending on number of words written, the total number of cycles may be from 6 to 37.*
3. *For maximum efficiency, it is recommended that the write buffer be loaded with the highest number of words (N words) possible.*

The following is a C source code example of using the write buffer program function. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: Write Buffer Programming Command            */
/* NOTES: Write buffer programming limited to 16 words. */
/*        All addresses to be written to the flash in   */
/*        one operation must be within the same flash   */
/*        page. A flash page begins at addresses       */
/*        evenly divisible by 0x20.                     */
  UINT16 *src = source_of_data;               /* address of source data          */
  UINT16 *dst = destination_of_data;          /* flash destination address       */
  UINT16 wc   = words_to_program -1;          /* word count (minus 1)            */
  *( (UINT16 *)base_addr + 0x555 ) = 0x00AA;  /* write unlock cycle 1            */
  *( (UINT16 *)base_addr + 0x2AA ) = 0x0055;  /* write unlock cycle 2            */
  *( (UINT16 *)sector_address )    = 0x0025;  /* write write buffer load command */
  *( (UINT16 *)sector_address )    = wc;      /* write word count (minus 1)      */
loop:
  *dst = *src; /* ALL dst MUST BE SAME PAGE */ /* write source data to destination */
  dst++;                                      /* increment destination pointer    */
  src++;                                      /* increment source pointer         */
  if (wc == 0) goto confirm                   /* done when word count equals zero */
  wc--;                                       /* decrement word count             */
  goto loop;                                  /* do it again                      */
confirm:
  *( (UINT16 *)sector_address )    = 0x0029;  /* write confirm command           */
  /* poll for completion */


/* Example: Write Buffer Abort Reset */
  *( (UINT16 *)addr + 0x555 ) = 0x00AA;   /* write unlock cycle 1           */
  *( (UINT16 *)addr + 0x2AA ) = 0x0055;   /* write unlock cycle 2           */
  *( (UINT16 *)addr + 0x555 ) = 0x00F0;   /* write buffer abort reset       */
```

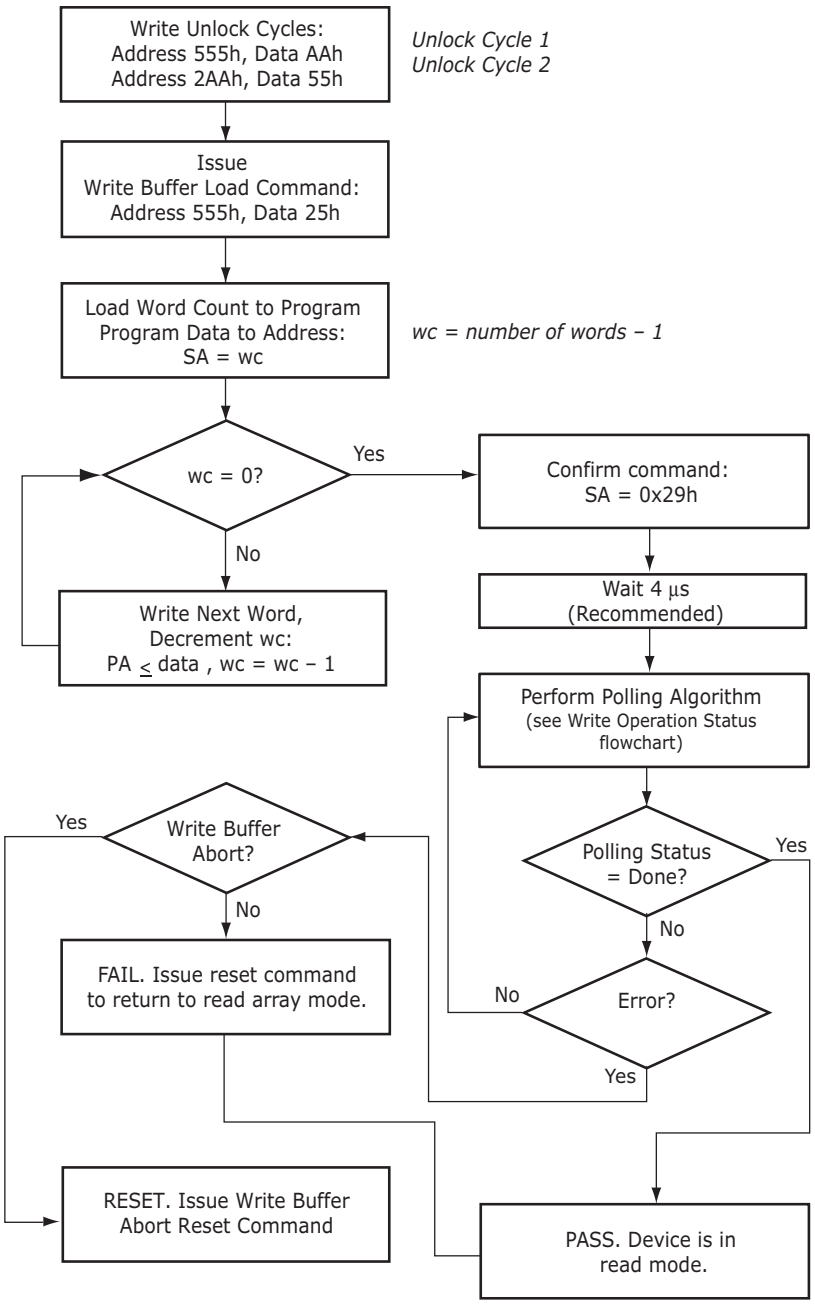


Figure I2.4. Write Buffer Programming Operation

12.6.3 Sector Erase

The sector erase function erases one or more sectors in the memory array. (See Table 17.1, Memory Array Commands; and Figure 12.5, Sector Erase Operation.) The device does not require the system to preprogram prior to erase. The Embedded Erase algorithm automatically programs and verifies the entire memory for an all zero data pattern prior to electrical erase. After a successful sector erase, all locations within the erased sector contain FFFFh. The system is not required to provide any controls or timings during these operations.

After the command sequence is written, a sector erase time-out of no less than t_{SEA} occurs. During the time-out period, additional sector addresses and sector erase commands may be written. Loading the sector erase buffer may be done in any sequence, and the number of sectors may be from one sector to all sectors. The time between these additional cycles must be less than t_{SEA}. Any sector erase address and command following the exceeded time-out (t_{SEA}) may or may not be accepted. Any command other than Sector Erase or Erase Suspend during the time-out period resets that bank to the read mode. The system can monitor DQ3 to determine if the sector erase timer has timed out (See the DQ3: Sector Erase Timeout State Indicator section.) The time-out begins from the rising edge of the final WE# pulse in the command sequence.

When the Embedded Erase algorithm is complete, the bank returns to reading array data and addresses are no longer latched. Note that while the Embedded Erase operation is in progress, the system can read data from the non-erasing banks. The system can determine the status of the erase operation by reading DQ7 or DQ6/DQ2 in the erasing bank. Refer to Write Operation Status for information on these status bits.

Once the sector erase operation has begun, only the Erase Suspend command is valid. All other commands are ignored. However, note that a hardware reset immediately terminates the erase operation. If that occurs, the sector erase command sequence should be reinitiated once that bank has returned to reading array data, to ensure data integrity.

Figure 12.5 illustrates the algorithm for the erase operation. Refer to the Erase and Programming Performance section for parameters and timing diagrams.

Software Functions and Sample Code

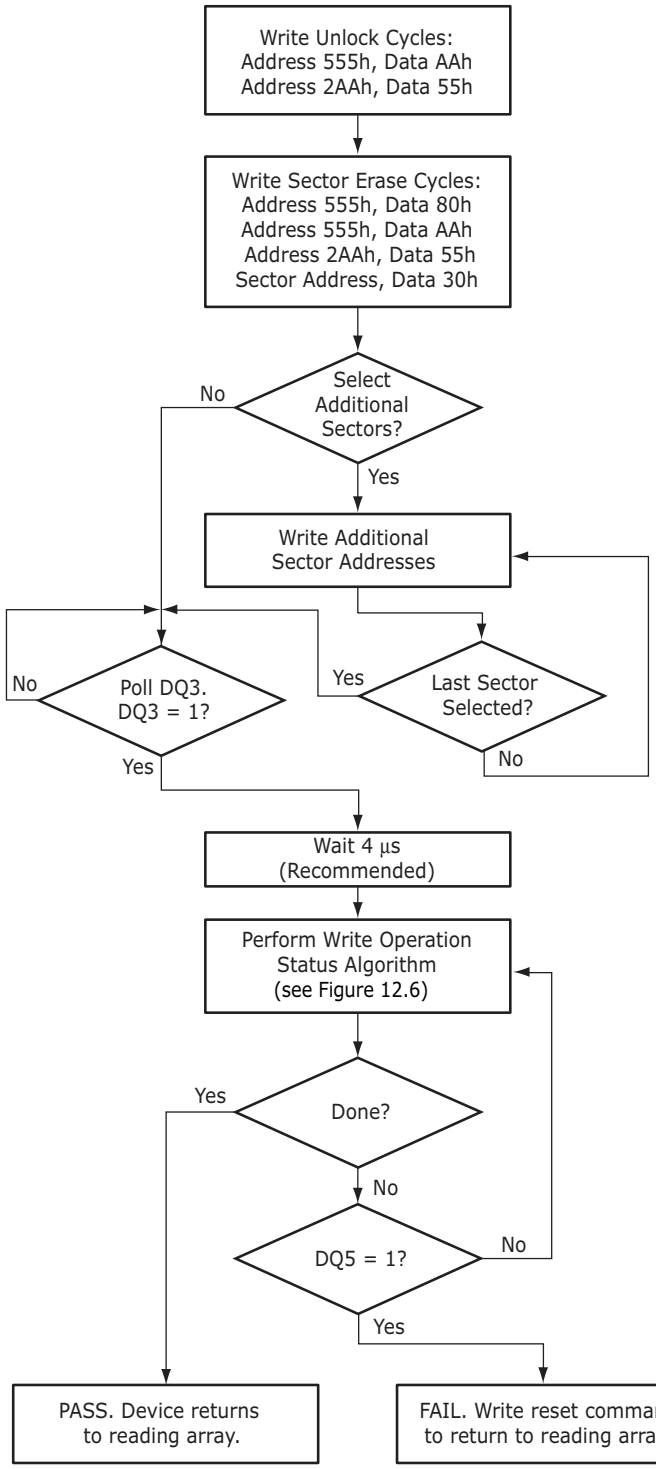
Table 12.16. Sector Erase

(LLD Function = lld_SectorEraseCmd)

Cycle	Description	Operation	Byte Address	Word Address	Data
1	Unlock	Write	Base + AAAh	Base + 555h	00AAh
2	Unlock	Write	Base + 554h	Base + 2AAh	0055h
3	Setup Command	Write	Base + AAAh	Base + 555h	0080h
4	Unlock	Write	Base + AAAh	Base + 555h	00AAh
5	Unlock	Write	Base + 554h	Base + 2AAh	0055h
6	Sector Erase Command	Write	Sector Address	Sector Address	0030h
Unlimited additional sectors may be selected for erase; command(s) must be written within t_{SEA}.					

The following is a C source code example of using the sector erase function. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: Sector Erase Command */
  *( (UINT16 *)base_addr + 0x555 ) = 0x00AA;   /* write unlock cycle 1            */
  *( (UINT16 *)base_addr + 0x2AA ) = 0x0055;   /* write unlock cycle 2            */
  *( (UINT16 *)base_addr + 0x555 ) = 0x0080;   /* write setup command             */
  *( (UINT16 *)base_addr + 0x555 ) = 0x00AA;   /* write additional unlock cycle 1 */
  *( (UINT16 *)base_addr + 0x2AA ) = 0x0055;   /* write additional unlock cycle 2 */
  *( (UINT16 *)sector_address )    = 0x0030;   /* write sector erase command      */
```



Unlock Cycle 1
Unlock Cycle 2

Command Cycle 1
Command Cycle 2
Command Cycle 3
Specify first sector for erasure

- *Each additional cycle must be written within t_{SEA} timeout*
- *Timeout resets after each additional cycle is written*
- *The host system may monitor DQ3 or wait t_{SEA} to ensure acceptance of erase commands*
- *No limit on number of sectors*
- *Commands other than Erase Suspend or selecting additional sectors for erasure during timeout reset device to reading array data*

Status may be obtained by reading DQ7, DQ6 and/or DQ2.

Error condition (Exceeded Timing Limits)

Notes:
1. *See Table 17.1 for erase command sequence.*
2. *See the section on DQ3 for information on the sector erase timeout.*

Figure 12.5. Sector Erase Operation

12.6.4 Chip Erase Command Sequence

Chip erase is a six-bus cycle operation as indicated by Table 17.1. These commands invoke the Embedded Erase algorithm, which does not require the system to preprogram prior to erase. The Embedded Erase algorithm automatically preprograms and verifies the entire memory for an all zero data pattern prior to electrical erase. After a successful chip erase, all locations of the chip contain FFFFh. The system is not required to provide any controls or timings during these operations. The "Command Definition" section in the appendix shows the address and data requirements for the chip erase command sequence.

When the Embedded Erase algorithm is complete, that bank returns to the read mode and addresses are no longer latched. The system can determine the status of the erase operation by using DQ7 or DQ6/DQ2. Refer to "Write Operation Status" for information on these status bits.

Any commands written during the chip erase operation are ignored. However, note that a hardware reset immediately terminates the erase operation. If that occurs, the chip erase command sequence should be reinitiated once that bank has returned to reading array data, to ensure data integrity.

Software Functions and Sample Code

Table 12.17. Chip Erase

(LLD Function = lld_ChipEraseCmd)

Cycle	Description	Operation	Byte Address	Word Address	Data
1	Unlock	Write	Base + AAAh	Base + 555h	00AAh
2	Unlock	Write	Base + 554h	Base + 2AAh	0055h
3	Setup Command	Write	Base + AAAh	Base + 555h	0080h
4	Unlock	Write	Base + AAAh	Base + 555h	00AAh
5	Unlock	Write	Base + 554h	Base + 2AAh	0055h
6	Chip Erase Command	Write	Base + AAAh	Base + 555h	0010h

The following is a C source code example of using the chip erase function. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: Chip Erase Command */
/* Note: Cannot be suspended   */
  *( (UINT16 *)base_addr + 0x555 ) = 0x00AA;   /* write unlock cycle 1            */
  *( (UINT16 *)base_addr + 0x2AA ) = 0x0055;   /* write unlock cycle 2            */
  *( (UINT16 *)base_addr + 0x555 ) = 0x0080;   /* write setup command             */
  *( (UINT16 *)base_addr + 0x555 ) = 0x00AA;   /* write additional unlock cycle 1 */
  *( (UINT16 *)base_addr + 0x2AA ) = 0x0055;   /* write additional unlock cycle 2 */
  *( (UINT16 *)base_addr + 0x000 ) = 0x0010;   /* write chip erase command        */
```

12.6.5 Erase Suspend/Erase Resume Commands

When the Erase Suspend command is written during the sector erase time-out, the device immediately terminates the time-out period and suspends the erase operation. The Erase Suspend command allows the system to interrupt a sector erase operation and then read data from, or program data to, any sector not selected for erasure. The bank address is required when writing this command. This command is valid only during the sector erase operation, including the minimum t_{SEA} time-out period during the sector erase command sequence. The Erase Suspend command is ignored if written during the chip erase operation.

When the Erase Suspend command is written after the t_{SEA} time-out period has expired and during the sector erase operation, the device requires a maximum of t_{ESL} (erase suspend latency) to suspend the erase operation. Additionaly, when an Erase Suspend command is written during an active erase operation, status information is unavailable during the transition from the sector erase operation to the erase suspended state.

After the erase operation has been suspended, the bank enters the erase-suspend-read mode. The system can read data from or program data to any sector not selected for erasure. (The device "erase suspends" all sectors selected for erasure.) Reading at any address within erase-suspended sectors produces status information on DQ7-DQ0. The system can use DQ7, or DQ6, and DQ2 together, to determine if a sector is actively erasing or is erase-suspended. Refer to Table 12.26 for information on these status bits.

After an erase-suspended program operation is complete, the bank returns to the erase-suspend-read mode. The system can determine the status of the program operation using the DQ7 or DQ6 status bits, just as in the standard program operation.

In the erase-suspend-read mode, the system can also issue the Autoselect command sequence. Refer to the "Write Buffer Programming Operation" section and the "Autoselect Command Sequence" section for details.

To resume the sector erase operation, the system must write the Erase Resume command. The bank address of the erase-suspended bank is required when writing this command. Further writes of the Resume command are ignored. Another Erase Suspend command can be written after the chip has resumed erasing.

Software Functions and Sample Code

Table 12.18. Erase Suspend

(LLD Function = lld_EraseSuspendCmd)

Cycle	Operation	Byte Address	Word Address	Data
1	Write	Bank Address	Bank Address	00B0h

The following is a C source code example of using the erase suspend function. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: Erase suspend command */
  *( (UINT16 *)bank_addr + 0x000 ) = 0x00B0;   /* write suspend command          */
```

Table 12.19. Erase Resume

(LLD Function = lld_EraseResumeCmd)

Cycle	Operation	Byte Address	Word Address	Data
1	Write	Bank Address	Bank Address	0030h

The following is a C source code example of using the erase resume function. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: Erase resume command */
  *( (UINT16 *)bank_addr + 0x000 ) = 0x0030;   /* write resume command           */
  /* The flash needs adequate time in the resume state */
```

12.6.6 Program Suspend/Program Resume Commands

The Program Suspend command allows the system to interrupt an embedded programming operation or a "Write to Buffer" programming operation so that data can read from any non-suspended sector. When the Program Suspend command is written during a programming process, the device halts the programming operation within t_{PSL} (program suspend latency) and updates the status bits. Addresses are "don't-cares" when writing the Program Suspend command.

After the programming operation has been suspended, the system can read array data from any non-suspended sector. The Program Suspend command may also be issued during a programming operation while an erase is suspended. In this case, data may be read from any addresses not in Erase Suspend or Program Suspend. If a read is needed from the Secured Silicon Sector area, then user must use the proper command sequences to enter and exit this region.

The system may also write the Autoselect command sequence when the device is in Program Suspend mode. The device allows reading Autoselect codes in the suspended sectors, since the codes are not stored in the memory array. When the device exits the Autoselect mode, the device reverts to Program Suspend mode, and is ready for another valid operation. See "Autoselect Command Sequence" for more information.

After the Program Resume command is written, the device reverts to programming. The system can determine the status of the program operation using the DQ7 or DQ6 status bits, just as in the standard program operation. See "Write Operation Status" for more information.

The system must write the Program Resume command (address bits are "don't care") to exit the Program Suspend mode and continue the programming operation. Further writes of the Program Resume command are ignored. Another Program Suspend command can be written after the device has resumed programming.

Software Functions and Sample Code

Table 12.20. Program Suspend

(LLD Function = lld_ProgramSuspendCmd)

Cycle	Operation	Byte Address	Word Address	Data
1	Write	Bank Address	Bank Address	00B0h

The following is a C source code example of using the program suspend function. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: Program suspend command */
  *( (UINT16 *)base_addr + 0x000 ) = 0x00B0;   /* write suspend command          */
```

Table 12.21. Program Resume

(LLD Function = lld_ProgramResumeCmd)

Cycle	Operation	Byte Address	Word Address	Data
1	Write	Bank Address	Bank Address	0030h

The following is a C source code example of using the program resume function. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: Program resume command */
  *( (UINT16 *)base_addr + 0x000 ) = 0x0030;   /* write resume command           */
```

12.6.7 Accelerated Program/Chip Erase

Accelerated single word programming, write buffer programming, sector erase, and chip erase operations are enabled through the ACC function. This method is faster than the standard chip program and erase command sequences.

The accelerated chip program and erase functions must not be used more than 10 times per sector. In addition, accelerated chip program and erase should be performed at room temperature (25°C ±10°C).

If the system asserts V_{HH} on this input, the device automatically enters the aforementioned Unlock Bypass mode and uses the higher voltage on the input to reduce the time required for program and erase operations. The system can then use the Write Buffer Load command sequence provided by the Unlock Bypass mode. Note that if a "Write-to-Buffer-Abort Reset" is required while in Unlock Bypass mode, the full 3-cycle RESET command sequence must be used to reset the device. Removing V_{HH} from the ACC input, upon completion of the embedded program or erase operation, returns the device to normal operation.

- Sectors must be unlocked prior to raising ACC to V_{HH}.
- The ACC pin must not be at V_{HH} for operations other than accelerated programming and accelerated chip erase, or device damage may result.
- The ACC pin must not be left floating or unconnected; inconsistent behavior of the device may result.
- ACC locks all sector if set to V_{IL}; ACC should be set to V_{IH} for all other conditions.

12.6.8 Unlock Bypass

The device features an Unlock Bypass mode to facilitate faster word programming. Once the device enters the Unlock Bypass mode, only two write cycles are required to program data, instead of the normal four cycles.

This mode dispenses with the initial two unlock cycles required in the standard program command sequence, resulting in faster total programming time. The "Command Definition Summary" section shows the requirements for the unlock bypass command sequences.

During the unlock bypass mode, only the Read, Unlock Bypass Program and Unlock Bypass Reset commands are valid. To exit the unlock bypass mode, the system must issue the two-cycle unlock bypass reset command sequence. The first cycle must contain the bank address and the data 90h. The second cycle need only contain the data 00h. The bank then returns to the read mode.

Software Functions and Sample Code

The following are C source code examples of using the unlock bypass entry, program, and exit functions. Refer to the *Spansion Low Level Driver User's Guide* (available soon on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

Table 12.22. Unlock Bypass Entry

(LLD Function = lld_UnlockBypassEntryCmd)

Cycle	Description	Operation	Byte Address	Word Address	Data
1	Unlock	Write	Base + AAAh	Base + 555h	00AAh
2	Unlock	Write	Base + 554h	Base + 2AAh	0055h
3	Entry Command	Write	Base + AAAh	Base + 555h	0020h

```
/* Example: Unlock Bypass Entry Command    */
  *( (UINT16 *)bank_addr + 0x555 ) = 0x00AA;   /* write unlock cycle 1            */
  *( (UINT16 *)bank_addr + 0x2AA ) = 0x0055;   /* write unlock cycle 2            */
  *( (UINT16 *)bank_addr + 0x555 ) = 0x0020;   /* write unlock bypass command     */
  /* At this point, programming only takes two write cycles.   */
  /* Once you enter Unlock Bypass Mode, do a series of like    */
  /* operations (programming or sector erase) and then exit    */
  /* Unlock Bypass Mode before beginning a different type of   */
  /* operations.                                               */
```

Table I2.23. Unlock Bypass Program

(LLD Function = lld_UnlockBypassProgramCmd)

Cycle	Description	Operation	Byte Address	Word Address	Data
1	Program Setup Command	Write	Base + xxxh	Base +xxxh	00A0h
2	Program Command	Write	Program Address	Program Address	Program Data

```
/* Example: Unlock Bypass Program Command  */
/* Do while in Unlock Bypass Entry Mode!   */
  *( (UINT16 *)bank_addr + 0x555 ) = 0x00A0;   /* write program setup command     */
  *( (UINT16 *)pa )                = data;     /* write data to be programmed     */
  /* Poll until done or error.    */
  /* If done and more to program, */
  /* do above two cycles again.   */
```

Table I2.24. Unlock Bypass Reset

(LLD Function = lld_UnlockBypassResetCmd)

Cycle	Description	Operation	Byte Address	Word Address	Data
1	Reset Cycle 1	Write	Base + xxxh	Base +xxxh	0090h
2	Reset Cycle 2	Write	Base + xxxh	Base +xxxh	0000h

```
/* Example: Unlock Bypass Exit Command */
  *( (UINT16 *)base_addr + 0x000 ) = 0x0090;
  *( (UINT16 *)base_addr + 0x000 ) = 0x0000;
```

12.6.9 Write Operation Status

The device provides several bits to determine the status of a program or erase operation. The following subsections describe the function of DQ1, DQ2, DQ3, DQ5, DQ6, and DQ7.

DQ7: Data# Polling. The Data# Polling bit, DQ7, indicates to the host system whether an Embedded Program or Erase algorithm is in progress or completed, or whether a bank is in Erase Suspend. Data# Polling is valid after the rising edge of the final WE# pulse in the command sequence. Note that the Data# Polling is valid only for the last word being programmed in the write-buffer-page during Write Buffer Programming. Reading Data# Polling status on any word other than the last word to be programmed in the write-buffer-page returns false status information.

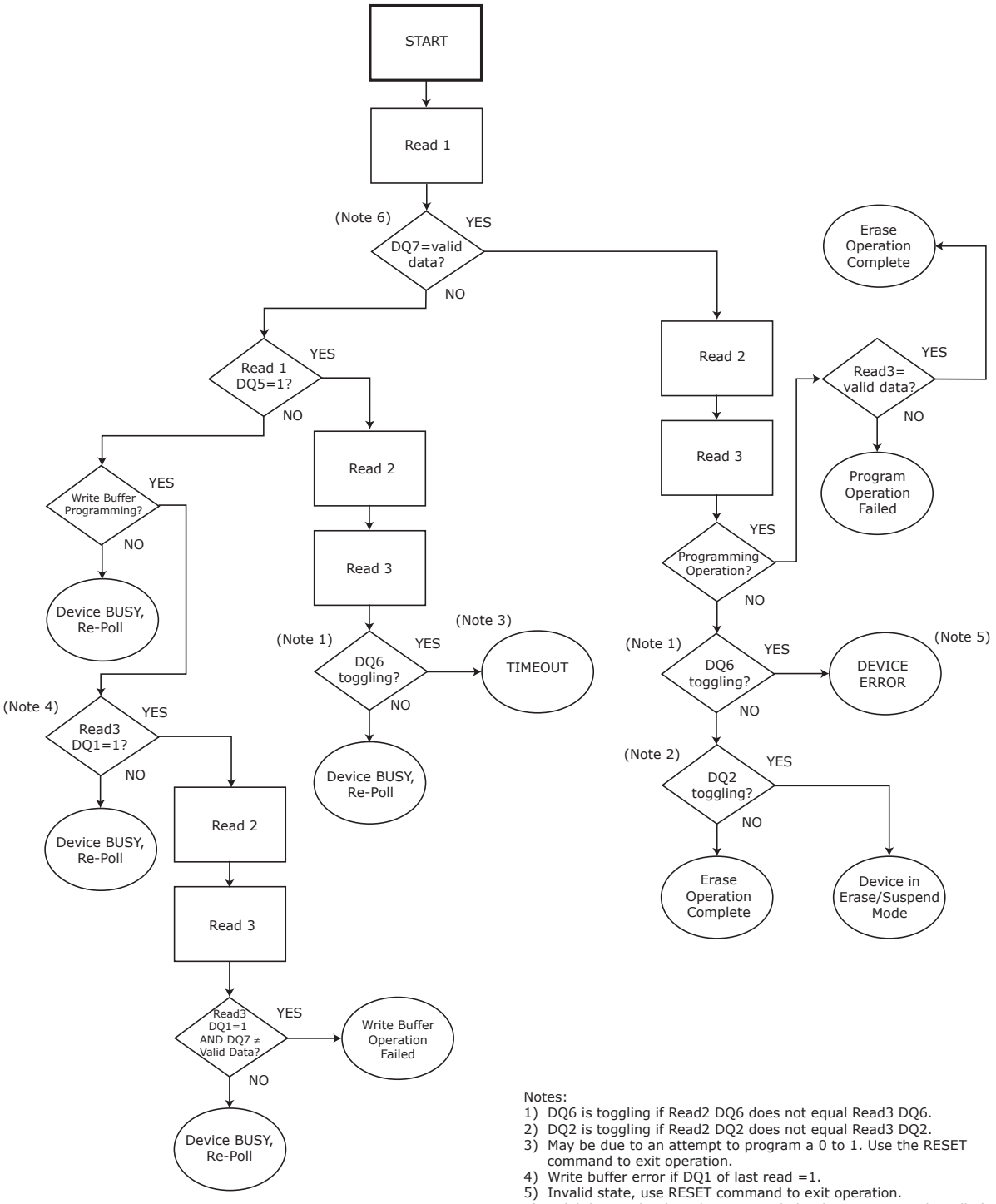
During the Embedded Program algorithm, the device outputs on DQ7 the complement of the datum programmed to DQ7. This DQ7 status also applies to programming during Erase Suspend. When the Embedded Program algorithm is complete, the device outputs the datum programmed to DQ7. The system must provide the program address to read valid status information on DQ7. If a program address falls within a protected sector, Data# polling on DQ7 is active for approximately t_{PSP}, then that bank returns to the read mode.

During the Embedded Erase Algorithm, Data# polling produces a "0" on DQ7. When the Embedded Erase algorithm is complete, or if the bank enters the Erase Suspend mode, Data# Polling produces a "1" on DQ7. The system must provide an address within any of the sectors selected for erasure to read valid status information on DQ7.

After an erase command sequence is written, if all sectors selected for erasing are protected, Data# Polling on DQ7 is active for approximately t_{ASP}, then the bank returns to the read mode. If not all selected sectors are protected, the Embedded Erase algorithm erases the unprotected sectors, and ignores the selected sectors that are protected. However, if the system reads DQ7 at an address within a protected sector, the status may not be valid.

Just prior to the completion of an Embedded Program or Erase operation, DQ7 may change asynchronously with DQ6-DQ0 while Output Enable (OE#) is asserted low. That is, the device may change from providing status information to valid data on DQ7. Depending on when the system samples the DQ7 output, it may read the status or valid data. Even if the device has completed the program or erase operation and DQ7 has valid data, the data outputs on DQ6-DQ0 may be still invalid. Valid data on DQ7-D00 appears on successive read cycles.

See the following for more information: Table 12.26, Write Operation Status, shows the outputs for Data# Polling on DQ7. Figure 12.6, Write Operation Status Flowchart, shows the Data# Polling algorithm; and Figure 16.18, Data# Polling Timings (During Embedded Algorithm), shows the Data# Polling timing diagram.



Notes:
1) DQ6 is toggling if Read2 DQ6 does not equal Read3 DQ6.
2) DQ2 is toggling if Read2 DQ2 does not equal Read3 DQ2.
3) May be due to an attempt to program a 0 to 1. Use the RESET command to exit operation.
4) Write buffer error if DQ1 of last read =1.
5) Invalid state, use RESET command to exit operation.
6) Valid data is the data that is intended to be programmed or all 1's for an erase operation.
7) Data polling algorithm valid for all operations except advanced sector protection.

Figure 12.6. Write Operation Status Flowchart

DQ6: Toggle Bit I . Toggle Bit I on DQ6 indicates whether an Embedded Program or Erase algorithm is in progress or complete, or whether the device has entered the Erase Suspend mode. Toggle Bit I may be read at any address in the same bank, and is valid after the rising edge of the final WE# pulse in the command sequence (prior to the program or erase operation), and during the sector erase time-out.

During an Embedded Program or Erase algorithm operation, successive read cycles to any address cause DQ6 to toggle. When the operation is complete, DQ6 stops toggling.

After an erase command sequence is written, if all sectors selected for erasing are protected, DQ6 toggles for approximately t_{ASP} [all sectors protected toggle time], then returns to reading array data. If not all selected sectors are protected, the Embedded Erase algorithm erases the unprotected sectors, and ignores the selected sectors that are protected.

The system can use DQ6 and DQ2 together to determine whether a sector is actively erasing or is erase-suspended. When the device is actively erasing (that is, the Embedded Erase algorithm is in progress), DQ6 toggles. When the device enters the Erase Suspend mode, DQ6 stops toggling. However, the system must also use DQ2 to determine which sectors are erasing or erase-suspended. Alternatively, the system can use DQ7 (see the subsection on DQ7: Data# Polling).

If a program address falls within a protected sector, DQ6 toggles for approximately t_{PAP} after the program command sequence is written, then returns to reading array data.

DQ6 also toggles during the erase-suspend-program mode, and stops toggling once the Embedded Program Algorithm is complete.

See the following for additional information: Figure 12.6, Write Operation Status Flowchart; Figure 16.19, Toggle Bit Timings (During Embedded Algorithm), and Tables 12.25 and 12.26.

Toggle Bit I on DQ6 requires either OE# or CE# to be de-asserted and reasserted to show the change in state.

DQ2: Toggle Bit II . The "Toggle Bit II" on DQ2, when used with DQ6, indicates whether a particular sector is actively erasing (that is, the Embedded Erase algorithm is in progress), or whether that sector is erase-suspended. Toggle Bit II is valid after the rising edge of the final WE# pulse in the command sequence. DQ2 toggles when the system reads at addresses within those sectors that have been selected for erasure. But DQ2 cannot distinguish whether the sector is actively erasing or is erase-suspended. DQ6, by comparison, indicates whether the device is actively erasing, or is in Erase Suspend, but cannot distinguish which sectors are selected for erasure. Thus, both status bits are required for sector and mode information. Refer to Table 12.25 to compare outputs for DQ2 and DQ6. See the following for additional information: Figure 12.6, the "DQ6: Toggle Bit I" section, and Figures 16.18–16.25.

Table I2.25. DQ6 and DQ2 Indications

If device is	and the system reads	then DQ6	and DQ2
programming,	at any address,	toggles,	does not toggle.
actively erasing,	at an address within a sector selected for erasure,	toggles,	also toggles.
	at an address within sectors *not* selected for erasure,	toggles,	does not toggle.
erase suspended,	at an address within a sector selected for erasure,	does not toggle,	toggles.
	at an address within sectors *not* selected for erasure,	returns array data,	returns array data. The system can read from any sector not selected for erasure.
programming in erase suspend	at any address,	toggles,	is not applicable.

Reading Toggle Bits DQ6/DQ2. Whenever the system initially begins reading toggle bit status, it must read DQ7–DQ0 at least twice in a row to determine whether a toggle bit is toggling. Typically, the system would note and store the value of the toggle bit after the first read. After the second read, the system would compare the new value of the toggle bit with the first. If the toggle bit is not toggling, the device has completed the program or erases operation. The system can read array data on DQ7–DQ0 on the following read cycle. However, if after the initial two read cycles, the system determines that the toggle bit is still toggling, the system also should note whether the value of DQ5 is high (see the section on DQ5). If it is, the system should then determine again whether the toggle bit is toggling, since the toggle bit may have stopped toggling just as DQ5 went high. If the toggle bit is no longer toggling, the device has successfully completed the program or erases operation. If it is still toggling, the device did not complete the operation successfully, and the system must write the reset command to return to reading array data. The remaining scenario is that the system initially determines that the toggle bit is toggling and DQ5 has not gone high. The system may continue to monitor the toggle bit and DQ5 through successive read cycles, determining the status as described in the previous paragraph. Alternatively, it may choose to perform other system tasks. In this case, the system must start at the beginning of the algorithm when it returns to determine the status of the operation. Refer to Figure 12.6 for more details.

Note:

- When verifying the status of a write operation (embedded program/erase) of a memory bank, DQ6 and DQ2 toggle between high and low states in a series of consecutive and contiguous status read cycles. In order for this toggling behavior to be properly observed, the consecutive status bit reads must not be interleaved with read accesses to other memory banks. If it is not possible to temporarily prevent reads to other memory banks, then it is recommended to use the DQ7 status bit as the alternative method of determining the active or inactive status of the write operation.
- Data polling provides erroneous results during erase suspend operation using DQ2 or DQ6 for any address changes after CE# asseration or without AVD# pulsing low. THe user is required to pulse AVD# following an address change or assert CE# after address is stable during status polling. See Figure 16.21 through 16.24 .

DQ5: Exceeded Timing Limits. DQ5 indicates whether the program or erase time has exceeded a specified internal pulse count limit. Under these conditions DQ5 produces a "1," indicating that the program or erase cycle was not successfully completed. The device may output a "1" on DQ5 if the system tries to program a "1" to a location that was previously programmed to "0." Only an erase operation can change a "0" back to a "1." Under this condition, the device halts the operation, and when the timing limit has been exceeded, DQ5 produces a "1."Under both these conditions, the system must write the reset command to return to the read mode (or to the erase-suspend-read mode if a bank was previously in the erase-suspend-program mode).

DQ3: Sector Erase Timeout State Indicator. After writing a sector erase command sequence, the system may read DQ3 to determine whether or not erasure has begun. (The sector erase timer does not apply to the chip erase command.) If additional sectors are selected for erasure, the entire time-out also applies after each additional sector erase command. When the time-out period is complete, DQ3 switches from a "0" to a "1." If the time between additional sector erase commands from the system can be assumed to be less than t_{SEA}, the system need not monitor DQ3. See Sector Erase Command Sequence for more details.

After the sector erase command is written, the system should read the status of DQ7 (Data# Polling) or DQ6 (Toggle Bit I) to ensure that the device has accepted the command sequence, and then read DQ3. If DQ3 is "1," the Embedded Erase algorithm has begun; all further commands (except Erase Suspend) are ignored until the erase operation is complete. If DQ3 is "0," the device accepts additional sector erase commands. To ensure the command has been accepted, the system software should check the status of DQ3 prior to and following each subsequent sector erase command. If DQ3 is high on the second status check, the last command might not have been accepted. Table 12.26 shows the status of DQ3 relative to the other status bits.

DQ1: Write to Buffer Abort. DQ1 indicates whether a Write to Buffer operation was aborted. Under these conditions DQ1 produces a "1". The system must issue the Write to Buffer Abort Reset command sequence to return the device to reading array data. See Write Buffer Programming Operation for more details.

Table 12.26. Write Operation Status

Status			DQ7 (Note 2)	DQ6	DQ5 (Note 1)	DQ3	DQ2 (Note 2)	DQ1 (Note 4)
Standard Mode	Embedded Program Algorithm		DQ7#	Toggle	0	N/A	No toggle	0
	Embedded Erase Algorithm		0	Toggle	0	1	Toggle	N/A
Program Suspend Mode (Note 3)	Reading within Program Suspended Sector		INVALID *(Not Allowed)*	INVALID *(Not Allowed)*	INVALID *(Not Allowed)*	INVALID *(Not Allowed)*	INVALID *(Not Allowed)*	INVALID *(Not Allowed)*
	Reading within Non-Program Suspended Sector		Data	Data	Data	Data	Data	Data
Erase Suspend Mode (Note 6)	Erase-Suspend-Read	Erase Suspended Sector	1	No toggle	0	N/A	Toggle	N/A
		Non-Erase Suspended Sector	Data	Data	Data	Data	Data	Data
	Erase-Suspend-Program		DQ7#	Toggle	0	N/A	N/A	N/A
Write to Buffer (Note 5)	BUSY State		DQ7#	Toggle	0	N/A	N/A	0
	Exceeded Timing Limits		DQ7#	Toggle	1	N/A	N/A	0
	ABORT State		DQ7#	Toggle	0	N/A	N/A	1

Notes:

1. *DQ5 switches to '1' when an Embedded Program or Embedded Erase operation has exceeded the maximum timing limits. Refer to the section on DQ5 for more information.*
2. *DQ7 and DQ2 require a valid address when reading status information. Refer to the appropriate subsection for further details.*
3. *Data are invalid for addresses in a Program Suspended sector.*
4. *DQ1 indicates the Write to Buffer ABORT status during Write Buffer Programming operations.*
5. *The data-bar polling algorithm should be used for Write Buffer Programming operations. Note that DQ7# during Write Buffer Programming indicates the data-bar for DQ7 data* ***for the LAST LOADED WRITE-BUFFER ADDRESS location****.*
6. *For any address changes after CE# assertion, re-assertion of CE# might be required after the addresses become stable for data polling during the erase suspend operation using DQ2/DQ6.*

12.7 Simultaneous Read/Write

The simultaneous read/write feature allows the host system to read data from one bank of memory while programming or erasing another bank of memory. An erase operation may also be suspended to read from or program another location within the same bank (except the sector being erased). Figure 16.29, Back-to-Back Read/Write Cycle Timings, shows how read and write cycles may be initiated for simultaneous operation with zero latency. Refer to the DC Characteristics (CMOS Compatible) table for read-while-program and read-while-erase current specification.

12.8 Writing Commands/Command Sequences

When the device is configured for Asynchronous read, only Asynchronous write operations are allowed, and CLK is ignored. When in the Synchronous read mode configuration, the device is able to perform both Asynchronous and Synchronous write operations. CLK and AVD# induced address latches are supported in the Synchronous programming mode. During a synchronous write operation, to write a command or command sequence (which includes programming data to the device and erasing sectors of memory), the system must drive AVD# and CE# to V_{IL}, and OE# to V_{IH} when providing an address to the device, and drive WE# and CE# to V_{IL}, and OE# to V_{IH} when writing commands or data. During an asynchronous write operation, the system must drive CE# and WE# to V_{IL} and OE# to V_{IH} when providing an address, command, and data. Addresses are latched on the last falling edge of WE# or CE#, while data is latched on the 1st rising edge of WE# or CE#. An erase operation can erase one sector, multiple sectors, or the entire device. Tables 11.1–11.2 indicate the address space that each sector occupies. The device address space is divided into sixteen banks: Banks 1 through 14 contain only 64 Kword sectors, while Banks 0 and 15 contain both 16 Kword boot sectors in addition to 64 Kword sectors. A "bank address" is the set of address bits required to uniquely select a bank. Similarly, a "sector address" is the address bits required to uniquely select a sector. I_{CC2} in "DC Characteristics" represents the active current specification for the write mode. "AC Characteristics-Synchronous" and "AC Characteristics-Asynchronous" contain timing specification tables and timing diagrams for write operations.

12.9 Handshaking

The handshaking feature allows the host system to detect when data is ready to be read by simply monitoring the RDY (Ready) pin, which is a dedicated output and controlled by CE#.

When the device is configured to operate in synchronous mode, and OE# is low (active), the initial word of burst data becomes available after either the falling or rising edge of the RDY pin (depending on the setting for bit 10 in the Configuration Register). It is recommended that the host system set CR13–CR11 in the Configuration Register to the appropriate number of wait states to ensure optimal burst mode operation (see Table 12.10, Configuration Register).

Bit 8 in the Configuration Register allows the host to specify whether RDY is active at the same time that data is ready, or one cycle before data is ready.

12.10 Hardware Reset

The RESET# input provides a hardware method of resetting the device to reading array data. When RESET# is driven low for at least a period of t_{RP}, the device immediately terminates any operation in progress, tristates all outputs, resets the configuration register, and ignores all read/write commands for the duration of the RESET# pulse. The device also resets the internal state machine to reading array data.

To ensure data integrity the operation that was interrupted should be reinitiated once the device is ready to accept another command sequence.

When RESET# is held at V_{SS}, the device draws CMOS standby current (I_{CC4}). If RESET# is held at V_{IL}, but not at V_{SS}, the standby current is greater.

RESET# may be tied to the system reset circuitry which enables the system to read the boot-up firmware from the Flash memory upon a system reset.

See Figures 16.5 and 16.13 for timing diagrams.

12.11 Software Reset

Software reset is part of the command set (see Table 17.1) that also returns the device to array read mode and must be used for the following conditions:

1. to exit Autoselect mode
2. when DQ5 goes high during write status operation that indicates program or erase cycle was not successfully completed
3. exit sector lock/unlock operation.
4. to return to erase-suspend-read mode if the device was previously in Erase Suspend mode.
5. after any aborted operations

Software Functions and Sample Code

Table 12.27. Reset

(LLD Function = lld_ResetCmd)

Cycle	Operation	Byte Address	Word Address	Data
Reset Command	Write	Base + xxxh	Base + xxxh	00F0h

Note: *Base = Base Address.*

The following is a C source code example of using the reset function. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: Reset (software reset of Flash state machine) */
  *( (UINT16 *)base_addr + 0x000 ) = 0x00F0;
```

The following are additional points to consider when using the reset command:

- This command resets the banks to the read and address bits are ignored.
- Reset commands are ignored once erasure has begun until the operation is complete.
- Once programming begins, the device ignores reset commands until the operation is complete
- The reset command may be written between the cycles in a program command sequence before programming begins (prior to the third cycle). This resets the bank to which the system was writing to the read mode.

- If the program command sequence is written to a bank that is in the Erase Suspend mode, writing the reset command returns that bank to the erase-suspend-read mode.
- The reset command may be also written during an Autoselect command sequence.
- If a bank has entered the Autoselect mode while in the Erase Suspend mode, writing the reset command returns that bank to the erase-suspend-read mode.
- If DQ1 goes high during a Write Buffer Programming operation, the system must write the "Write to Buffer Abort Reset" command sequence to RESET the device to reading array data. The standard RESET command does not work during this condition.
- To exit the unlock bypass mode, the system must issue a two-cycle unlock bypass reset command sequence [see command table for details].

I3 Advanced Sector Protection/Unprotection

The Advanced Sector Protection/Unprotection feature disables or enables programming or erase operations in any or all sectors and can be implemented through software and/or hardware methods, which are independent of each other. This section describes the various methods of protecting data stored in the memory array. An overview of these methods in shown in Figure 13.1.

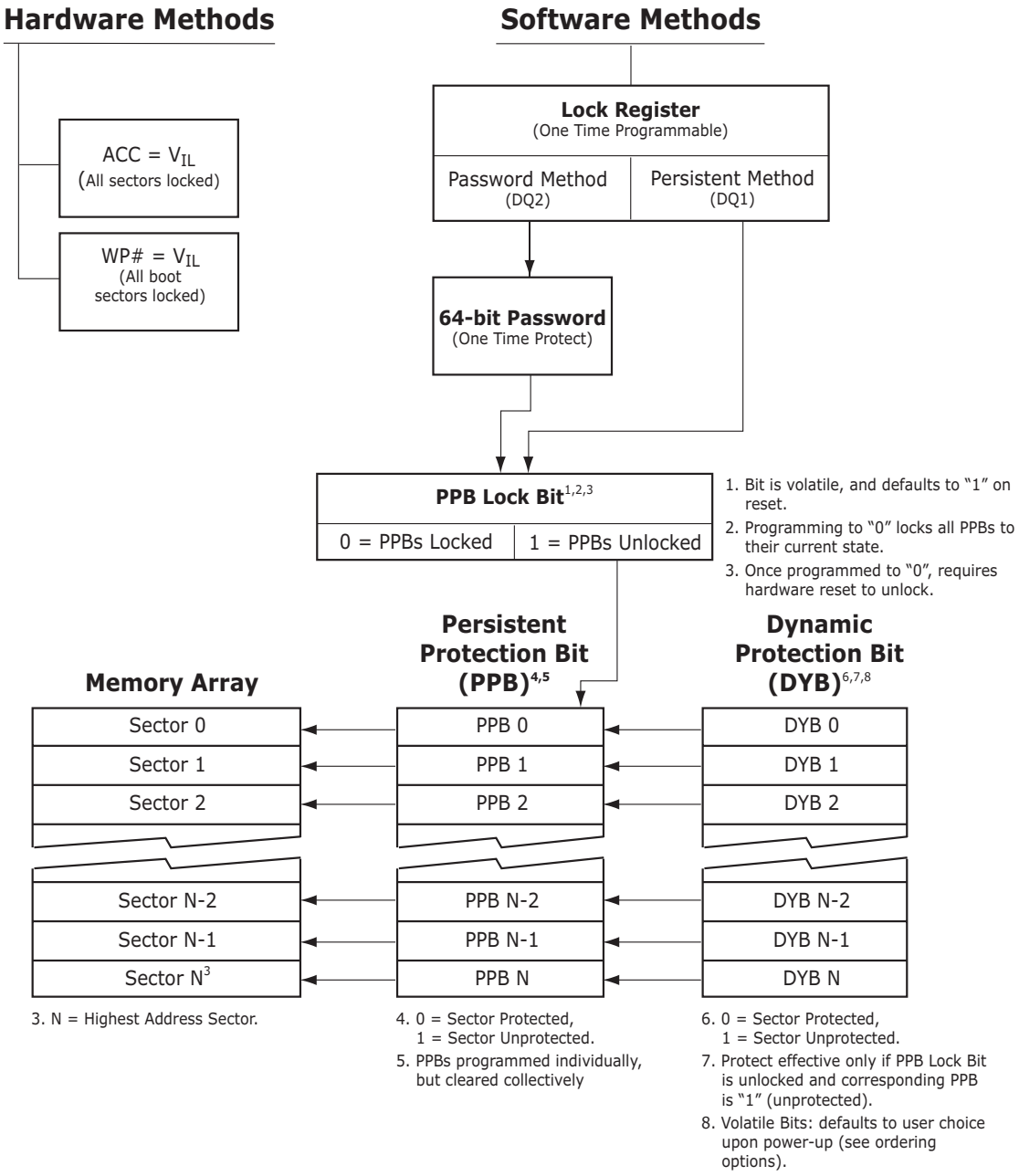


Figure I3.I. Advanced Sector Protection/Unprotection

13.1 Lock Register

As shipped from the factory, all devices default to the persistent mode when power is applied, and all sectors are unprotected, unless otherwise chosen through the DYB ordering option. The device programmer or host system must then choose which sector protection method to use. Programming (setting to "0") any one of the following two one-time programmable, non-volatile bits locks the part permanently in that mode:

- Lock Register Persistent Protection Mode Lock Bit (DQ1)
- Lock Register Password Protection Mode Lock Bit (DQ2)

Table 13.1 Lock Register

Device	DQ15-05	DQ4	DQ3	DQ2	DQ1	DQ0
S29WS256N	1	1	1	Password Protection Mode Lock Bit	Persistent Protection Mode Lock Bit	Customer SecSi Sector Protection Bit
S29WS128N	Undefined	DYB Lock Boot Bit 0 = sectors power up protected 1 = sectors power up unprotected	PPB One-Time Programmable Bit 0 = All PPB erase command disabled 1 = All PPB Erase command enabled	Password Protection Mode Lock Bit	Persistent Protection Mode Lock Bit	SecSi Sector Protection Bit

For programming lock register bits refer to Table 17.2.

Notes

1. If the password mode is chosen, the password must be programmed before setting the corresponding lock register bit.
2. After the Lock Register Bits Command Set Entry command sequence is written, reads and writes for Bank 0 are disabled, while reads from other banks are allowed until exiting this mode.
3. If both lock bits are selected to be programmed (to zeros) at the same time, the operation aborts.
4. Once the Password Mode Lock Bit is programmed, the Persistent Mode Lock Bit is permanently disabled, and no changes to the protection scheme are allowed. Similarly, if the Persistent Mode Lock Bit is programmed, the Password Mode is permanently disabled.
5. During erase/program suspend, ASP entry commands are not allowed.
6. When the device lock register is programmed (PPB mode lock bit is programmed, password mode lock bit is programmed, or the Secured Silicon Sector lock bit is programmed) all DYBs revert to the power-on default state.
7. Lock register programming operation:

   A. Data Polling can be done immediately after the lock register programming command sequence (no delay required). Note that status polling can be done only in bank 0 and the recommended 4-µs delay is for backward compatibility and is not required. This recommendation will be noted as such in the next revision of the data sheet.

   B. Reads from other banks (simultaneous operation) are not allowed during lock register programming. This restriction applies to both synchronous and asynchronous read operations.

   C. The above clarifications are true for programming any bits of the Lock Register.

After selecting a sector protection method, each sector can operate in any of the following three states:

1. *Constantly locked.* The selected sectors are protected and can not be reprogrammed unless PPB lock bit is cleared via a password, hardware reset, or power cycle.
2. *Dynamically locked.* The selected sectors are protected and can be altered via software commands.
3. *Unlocked.* The sectors are unprotected and can be erased and/or programmed.

These states are controlled by the bit types described in Sections 13.2–13.6.

13.2 Persistent Protection Bits

The Persistent Protection Bits are unique and nonvolatile for each sector and have the same endurances as the Flash memory. Preprogramming and verification prior to erasure are handled by the device, and therefore do not require system monitoring.

Notes

1. Each PPB is individually programmed and all are erased in parallel.
2. PPB program/erase operation: Reads from other banks (simultaneous operation) are not allowed during PPB programming/erase operation. This restriction applies for both synchronous and asynchronous read operations.
3. Entry command disables reads and writes for the bank selected.
4. Reads within that bank return the PPB status for that sector.
5. All Reads must be performed using the Asynchronous mode.
6. The specific sector address (A23-A14 WS256N, A22-A14 WS128N) are written at the same time as the program command.
7. If the PPB Lock Bit is set, the PPB Program or erase command does not execute and times-out without programming or erasing the PPB.
8. There are no means for individually erasing a specific PPB and no specific sector address is required for this operation.
9. Exit command must be issued after the execution which resets the device to read mode and re-enables reads and writes for Bank 0
10. The programming state of the PPB for a given sector can be verified by writing a PPB Status Read Command to the device as described by the flow chart shown in Figure 13.2.

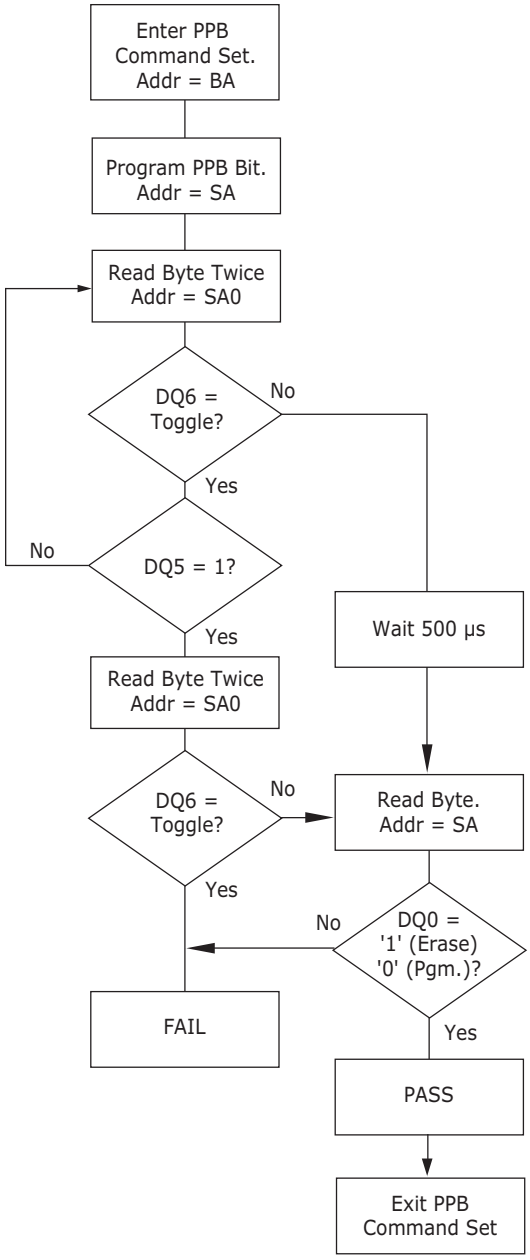


Figure I3.2. PPB Program/Erase Algorithm

13.3 Dynamic Protection Bits

Dynamic Protection Bits are volatile and unique for each sector and can be individually modified. DYBs only control the protection scheme for unprotected sectors that have their PPBs cleared (erased to "1"). By issuing the DYB Set or Clear command sequences, the DYBs are set (programmed to "0") or cleared (erased to "1"), thus placing each sector in the protected or unprotected state respectively. This feature allows software to easily protect sectors against inadvertent changes yet does not prevent the easy removal of protection when changes are needed.

Notes

1. The DYBs can be set (programmed to "0") or cleared (erased to "1") as often as needed.

When the parts are first shipped, the PPBs are cleared (erased to "1") and upon power up or reset, the DYBs can be set or cleared depending upon the ordering option chosen.

2. If the option to clear the DYBs after power up is chosen, (erased to "1"), then the sectors may be modified depending upon the PPB state of that sector (see Table 13.2).
3. The sectors would be in the protected state If the option to set the DYBs after power up is chosen (programmed to "0").
4. It is possible to have sectors that are persistently locked with sectors that are left in the dynamic state.
5. The DYB Set or Clear commands for the dynamic sectors signify protected or unprotected state of the sectors respectively. However, if there is a need to change the status of the persistently locked sectors, a few more steps are required. First, the PPB Lock Bit must be cleared by either putting the device through a power-cycle, or hardware reset. The PPBs can then be changed to reflect the desired settings. Setting the PPB Lock Bit once again locks the PPBs, and the device operates normally again.
6. To achieve the best protection, it is recommended to execute the PPB Lock Bit Set command early in the boot code and protect the boot code by holding WP# = V_{IL}. Note that the PPB and DYB bits have the same function when ACC = V_{HH} as they do when ACC =V_{IH}.

13.4 Persistent Protection Bit Lock Bit

The Persistent Protection Bit Lock Bit is a global volatile bit for all sectors. When set (programmed to "0"), it locks all PPBs and when cleared (programmed to "1"), allows the PPBs to be changed. There is only one PPB Lock Bit per device.

Notes

1. No software command sequence unlocks this bit unless the device is in the password protection mode; only a hardware reset or a power-up clears this bit.
2. The PPB Lock Bit must be set (programmed to "0") only after all PPBs are configured to the desired settings.

13.5 Password Protection Method

The Password Protection Method allows an even higher level of security than the Persistent Sector Protection Mode by requiring a 64 bit password for unlocking the device PPB Lock Bit. In addition to this password requirement, after power up and reset, the PPB Lock Bit is set "0" to maintain the password mode of operation. Successful execution of the Password Unlock command by entering the entire password clears the PPB Lock Bit, allowing for sector PPBs modifications.

Notes

1. There is no special addressing order required for programming the password. Once the Password is written and verified, the Password Mode Locking Bit must be set in order to prevent access.
2. The Password Program Command is only capable of programming "0"s. Programming a "1" after a cell is programmed as a "0" results in a time-out with the cell as a "0".
3. The password is all "1"s when shipped from the factory.
4. All 64-bit password combinations are valid as a password.
5. There is no means to verify what the password is after it is set.
6. The Password Mode Lock Bit, once set, prevents reading the 64-bit password on the data bus and further password programming.
7. The Password Mode Lock Bit is not erasable.
8. The lower two address bits (A1–A0) are valid during the Password Read, Password Program, and Password Unlock.
9. The exact password must be entered in order for the unlocking function to occur.
10. The Password Unlock command cannot be issued any faster than 1 µs at a time to prevent a hacker from running through all the 64-bit combinations in an attempt to correctly match a password.
11. Approximately 1 µs is required for unlocking the device after the valid 64-bit password is given to the device.
12. Password verification is only allowed during the password programming operation.
13. All further commands to the password region are disabled and all operations are ignored.
14. If the password is lost after setting the Password Mode Lock Bit, there is no way to clear the PPB Lock Bit.
15. Entry command sequence must be issued prior to any of any operation and it disables reads and writes for Bank 0. Reads and writes for other banks excluding Bank 0 are allowed.
16. If the user attempts to program or erase a protected sector, the device ignores the command and returns to read mode.
17. A program or erase command to a protected sector enables status polling and returns to read mode without having modified the contents of the protected sector.
18. The programming of the DYB, PPB, and PPB Lock for a given sector can be verified by writing individual status read commands DYB Status, PPB Status, and PPB Lock Status to the device.

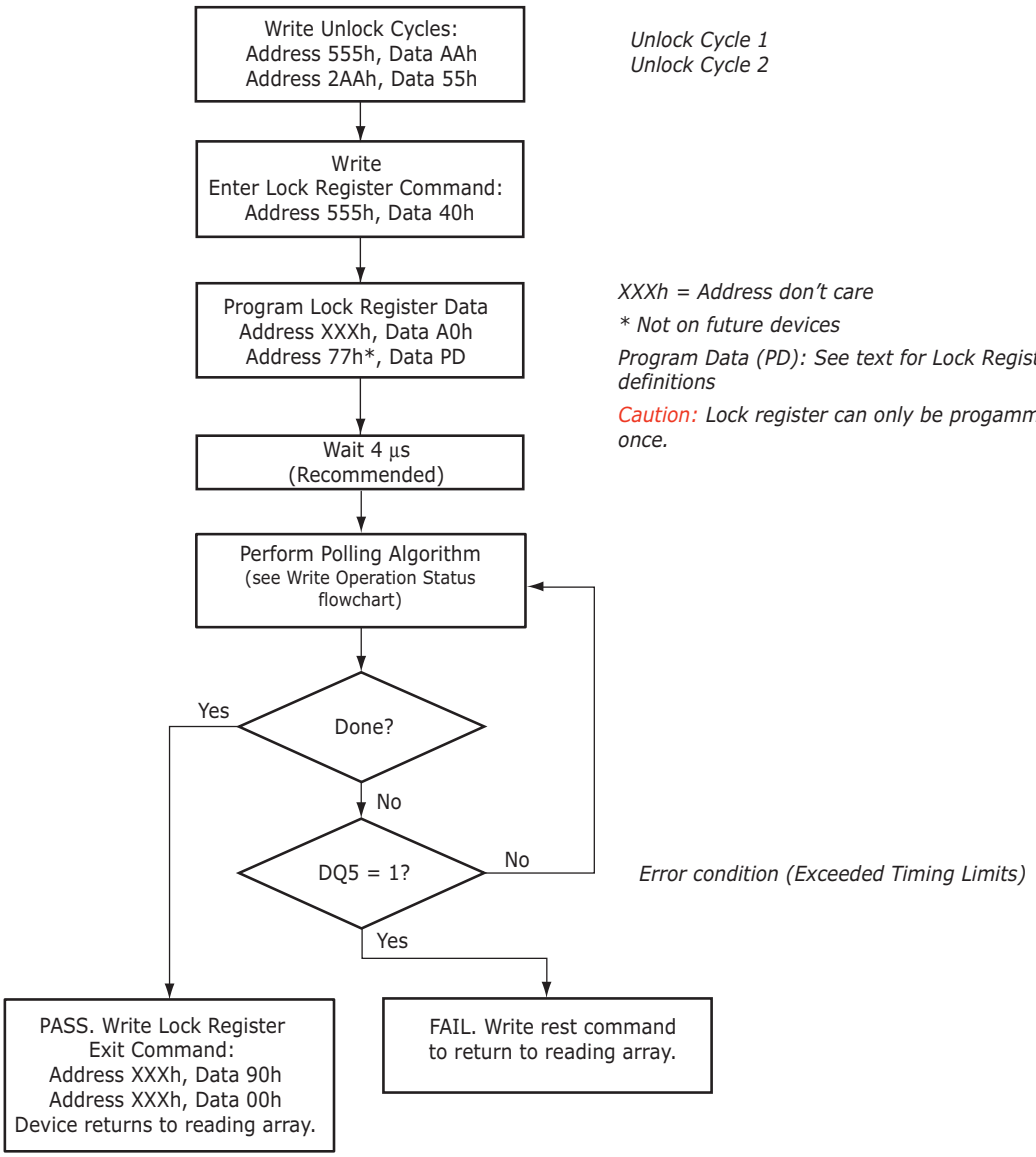


Figure I3.3. Lock Register Program Algorithm

13.6 Advanced Sector Protection Software Examples

Table 13.2 Sector Protection Schemes

Unique Device PPB Lock Bit 0 = locked 1 = unlocked		Sector PPB 0 = protected 1 = unprotected	Sector DYB 0 = protected 1 = unprotected	Sector Protection Status
Any Sector	0	0	x	Protected through PPB
Any Sector	0	0	x	Protected through PPB
Any Sector	0	1	1	Unprotected
Any Sector	0	1	0	Protected through DYB
Any Sector	1	0	x	Protected through PPB
Any Sector	1	0	x	Protected through PPB
Any Sector	1	1	0	Protected through DYB
Any Sector	1	1	1	Unprotected

Table 13.2 contains all possible combinations of the DYB, PPB, and PPB Lock Bit relating to the status of the sector. In summary, if the PPB Lock Bit is locked (set to "0"), no changes to the PPBs are allowed. The PPB Lock Bit can only be unlocked (reset to "1") through a hardware reset or power cycle. See also Figure 13.1 for an overview of the Advanced Sector Protection feature.

13.7 Hardware Data Protection Methods

The device offers two main types of data protection at the sector level via hardware control:

- When WP# is at V_{IL}, the four outermost sectors are locked (device specific).
- When ACC is at V_{IL}, all sectors are locked.

There are additional methods by which intended or accidental erasure of any sectors can be prevented via hardware means. The following subsections describes these methods:

13.7.1 WP# Method

The Write Protect feature provides a hardware method of protecting the four outermost sectors. This function is provided by the WP# pin and overrides the previously discussed Sector Protection/Unprotection method.

If the system asserts V_{IL} on the WP# pin, the device disables program and erase functions in the "outermost" boot sectors. The outermost boot sectors are the sectors containing both the lower and upper set of sectors in a dual-boot-configured device.

If the system asserts V_{IH} on the WP# pin, the device reverts to whether the boot sectors were last set to be protected or unprotected. That is, sector protection or unprotection for these sectors depends on whether they were last protected or unprotected.

Note that the WP# pin must not be left floating or unconnected as inconsistent behavior of the device may result.

The WP# pin must be held stable during a command sequence execution

13.7.2 ACC Method

This method is similar to above, except it protects all sectors. Once ACC input is set to V_{IL}, all program and erase functions are disabled and hence all sectors are protected.

13.7.3 Low V_{CC} Write Inhibit

When V_{CC} is less than V_{LKO}, the device does not accept any write cycles. This protects data during V_{CC} power-up and power-down.

The command register and all internal program/erase circuits are disabled, and the device resets to reading array data. Subsequent writes are ignored until V_{CC} is greater than V_{LKO}. The system must provide the proper signals to the control inputs to prevent unintentional writes when V_{CC} is greater than V_{LKO}.

13.7.4 Write Pulse "Glitch Protection"

Noise pulses of less than 3 ns (typical) on OE#, CE# or WE# do not initiate a write cycle.

13.7.5 Power-Up Write Inhibit

If WE# = CE# = RESET# = V_{IL} and OE# = V_{IH} during power up, the device does not accept commands on the rising edge of WE#. The internal state machine is automatically reset to the read mode on power-up.

14 Power Conservation Modes

14.1 Standby Mode

When the system is not reading or writing to the device, it can place the device in the standby mode. In this mode, current consumption is greatly reduced, and the outputs are placed in the high impedance state, independent of the OE# input. The device enters the CMOS standby mode when the CE# and RESET# inputs are both held at V_{CC} ± 0.2 V. The device requires standard access time (t_{CE}) for read access, before it is ready to read data. If the device is deselected during erasure or programming, the device draws active current until the operation is completed. I_{CC3} in "DC Characteristics" represents the standby current specification

14.2 Automatic Sleep Mode

The automatic sleep mode minimizes Flash device energy consumption while in asynchronous mode. the device automatically enables this mode when addresses remain stable for t_{ACC} + 20 ns. The automatic sleep mode is independent of the CE#, WE#, and OE# control signals. Standard address access timings provide new data when addresses are changed. While in sleep mode, output data is latched and always available to the system. While in synchronous mode, the automatic sleep mode is disabled. Note that a new burst operation is required to provide new data. I_{CC6} in DC Characteristics (CMOS Compatible) represents the automatic sleep mode current specification.

14.3 Hardware RESET# Input Operation

The RESET# input provides a hardware method of resetting the device to reading array data. When RESET# is driven low for at least a period of t_{RP}, the device immediately terminates any operation in progress, tristates all outputs, resets the configuration register, and ignores all read/write commands for the duration of the RESET# pulse. The device also resets the internal state machine to reading array data. The operation that was interrupted should be reinitiated once the device is ready to accept another command sequence to ensure data integrity.

When RESET# is held at V_{SS} ± 0.2 V, the device draws CMOS standby current (I_{CC4}). If RESET# is held at V_{IL} but not within V_{SS} ± 0.2 V, the standby current is greater.

RESET# may be tied to the system reset circuitry and thus, a system reset would also reset the Flash memory, enabling the system to read the boot-up firmware from the Flash memory.

14.4 Output Disable (OE#)

When the OE# input is at V_{IH}, output from the device is disabled. The outputs are placed in the high impedance state.

15 Secured Silicon Sector Flash Memory Region

The Secured Silicon Sector provides an extra Flash memory region that enables permanent part identification through an Electronic Serial Number (ESN). The Secured Silicon Sector is 256 words in length that consists of 128 words for factory data and 128 words for customer-secured areas. All Secured Silicon reads outside of the 256-word address range returns invalid data. The Factory Indicator Bit, DQ7, (at Autoselect address 03h) is used to indicate whether or not the Factory Secured Silicon Sector is locked when shipped from the factory. The Customer Indicator Bit (DQ6) is used to indicate whether or not the Customer Secured Silicon Sector is locked when shipped from the factory.

Please note the following general conditions:

- While Secured Silicon Sector access is enabled, simultaneous operations are allowed except for Bank 0.
- On power-up, or following a hardware reset, the device reverts to sending commands to the normal address space.
- Reads can be performed in the Asynchronous or Synchronous mode.
- Burst mode reads within Secured Silicon Sector wrap from address FFh back to address 00h.
- Reads outside of sector 0 return memory array data.
- Continuous burst read past the maximum address is undefined.
- Sector 0 is remapped from memory array to Secured Silicon Sector array.
- Once the Secured Silicon Sector Entry Command is issued, the Secured Silicon Sector Exit command must be issued to exit Secured Silicon Sector Mode.
- The Secured Silicon Sector is not accessible when the device is executing an Embedded Program or Embedded Erase algorithm.

Table 15.1 Secured Silicon Sector Addresses

Sector	Sector Size	Address Range
Customer	128 words	000080h-0000FFh
Factory	128 words	000000h-00007Fh

15.1 Factory Secured Silicon Sector

The Factory Secured Silicon Sector is always protected when shipped from the factory and has the Factory Indicator Bit (DQ7) permanently set to a "1". This prevents cloning of a factory locked part and ensures the security of the ESN and customer code once the product is shipped to the field.

These devices are available pre programmed with one of the following:

- A random, 8 Word secure ESN only within the Factory Secured Silicon Sector
- Customer code within the Customer Secured Silicon Sector through the Spansion™ programming service.
- Both a random, secure ESN and customer code through the Spansion programming service.

Customers may opt to have their code programmed through the Spansion programming services. Spansion programs the customer's code, with or without the random ESN. The devices are then shipped from the Spansion factory with the Factory Secured Silicon Sector and Customer Secured Silicon Sector permanently locked. Contact your local representative for details on using Spansion programming services.

I5.2 Customer Secured Silicon Sector

The Customer Secured Silicon Sector is typically shipped unprotected (DQ6 set to "0"), allowing customers to utilize that sector in any manner they choose. If the security feature is not required, the Customer Secured Silicon Sector can be treated as an additional Flash memory space.

Please note the following:

- Once the Customer Secured Silicon Sector area is protected, the Customer Indicator Bit is permanently set to "1."
- The Customer Secured Silicon Sector can be read any number of times, but can be programmed and locked only once. The Customer Secured Silicon Sector lock must be used with caution as once locked, there is no procedure available for unlocking the Customer Secured Silicon Sector area and none of the bits in the Customer Secured Silicon Sector memory space can be modified in any way.
- The accelerated programming (ACC) and unlock bypass functions are not available when programming the Customer Secured Silicon Sector, but reading in Banks 1 through 15 is available.
- Once the Customer Secured Silicon Sector is locked and verified, the system must write the Exit Secured Silicon Sector Region command sequence which return the device to the memory array at sector 0.

I5.3 Secured Silicon Sector Entry/Exit Command Sequences

The system can access the Secured Silicon Sector region by issuing the three-cycle Enter Secured Silicon Sector command sequence. The device continues to access the Secured Silicon Sector region until the system issues the four-cycle Exit Secured Silicon Sector command sequence.

See Command Definition Table [Secured Silicon Sector Command Table, Appendix
Table 17.1 for address and data requirements for both command sequences.

The Secured Silicon Sector Entry Command allows the following commands to be executed

- Read customer and factory Secured Silicon areas
- Program the customer Secured Silicon Sector

After the system has written the Enter Secured Silicon Sector command sequence, it may read the Secured Silicon Sector by using the addresses normally occupied by sector SA0 within the memory array. This mode of operation continues until the system issues the Exit Secured Silicon Sector command sequence, or until power is removed from the device.

Software Functions and Sample Code

The following are C functions and source code examples of using the Secured Silicon Sector Entry, Program, and exit commands. Refer to the *Spansion Low Level Driver User's Guide* (available soon on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

Table I5.2. Secured Silicon Sector Entry

(LLD Function = lld_SecSiSectorEntryCmd)

Cycle	Operation	Byte Address	Word Address	Data
Unlock Cycle 1	Write	Base + AAAh	Base + 555h	00AAh
Unlock Cycle 2	Write	Base + 554h	Base + 2AAh	0055h
Entry Cycle	Write	Base + AAAh	Base + 555h	0088h

Note: *Base = Base Address.*

```
/* Example: SecSi Sector Entry Command */
  *( (UINT16 *)base_addr + 0x555 ) = 0x00AA;   /* write unlock cycle 1            */
  *( (UINT16 *)base_addr + 0x2AA ) = 0x0055;   /* write unlock cycle 2            */
  *( (UINT16 *)base_addr + 0x555 ) = 0x0088;   /* write Secsi Sector Entry Cmd    */
```

Table I5.3. Secured Silicon Sector Program

(LLD Function = lld_ProgramCmd)

Cycle	Operation	Byte Address	Word Address	Data
Unlock Cycle 1	Write	Base + AAAh	Base + 555h	00AAh
Unlock Cycle 2	Write	Base + 554h	Base + 2AAh	0055h
Program Setup	Write	Base + AAAh	Base + 555h	00A0h
Program	Write	Word Address	Word Address	Data Word

Note: *Base = Base Address.*

```
/* Once in the SecSi Sector mode, you program */
/* words using the programming algorithm.     */
```

Table I5.4. Secured Silicon Sector Exit

(LLD Function = lld_SecSiSectorExitCmd)

Cycle	Operation	Byte Address	Word Address	Data
Unlock Cycle 1	Write	Base + AAAh	Base + 555h	00AAh
Unlock Cycle 2	Write	Base + 554h	Base + 2AAh	0055h
Exit Cycle	Write	Base + AAAh	Base + 555h	0090h

Note: *Base = Base Address.*

```
/* Example: SecSi Sector Exit Command */
  *( (UINT16 *)base_addr + 0x555 ) = 0x00AA;   /* write unlock cycle 1            */
  *( (UINT16 *)base_addr + 0x2AA ) = 0x0055;   /* write unlock cycle 2            */
  *( (UINT16 *)base_addr + 0x555 ) = 0x0090;   /* write SecSi Sector Exit cycle 3 */
  *( (UINT16 *)base_addr + 0x000 ) = 0x0000;   /* write SecSi Sector Exit cycle 4 */
```

16 Electrical Specifications

16.1 Absolute Maximum Ratings

Storage Temperature
Plastic Packages . . . . . .–65°C to +150°C
Ambient Temperature
with Power Applied . . . . . .–65°C to +125°C
Voltage with Respect to Ground:
All Inputs and I/Os except
as noted below (Note 1) . . . . . .–0.5 V to V_{CC} + 0.5 V
V_{CC} (Note 1). . . . . –0.5 V to +2.5 V
ACC (Note 2) . . . . . –0.5 V to +9.5 V
Output Short Circuit Current (Note 3) . . . . . 100 mA

Notes:

1. *Minimum DC voltage on input or I/Os is –0.5 V. During voltage transitions, inputs or I/Os may undershoot V_{SS} to –2.0 V for periods of up to 20 ns. See Figure 16.1. Maximum DC voltage on input or I/Os is V_{CC} + 0.5 V. During voltage transitions outputs may overshoot to V_{CC} + 2.0 V for periods up to 20 ns. See Figure 16.2.*
2. *Minimum DC input voltage on pin ACC is -0.5V. During voltage transitions, ACC may overshoot V_{SS} to –2.0 V for periods of up to 20 ns. See Figure 16.1. Maximum DC voltage on pin ACC is +9.5 V, which may overshoot to 10.5 V for periods up to 20 ns.*
3. *No more than one output may be shorted to ground at a time. Duration of the short circuit should not be greater than one second.*
4. *Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational sections of this data sheet is not implied. Exposure of the device to absolute maximum rating conditions for extended periods may affect device reliability.*

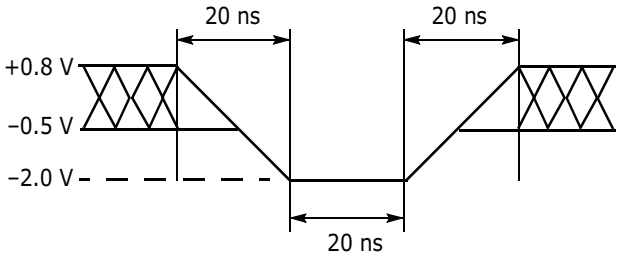


Figure 16.1. Maximum Negative Overshoot Waveform

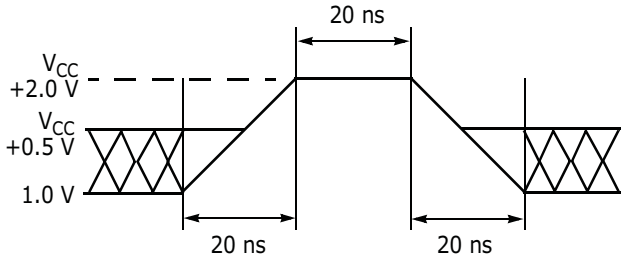


Figure 16.2. Maximum Positive Overshoot Waveform

Note: *The content in this document is Advance information for the S29WS128N. Content in this document is Preliminary for the S29W256N.*

16.2 Operating Ranges

Wireless (W) Devices

Ambient Temperature (T_A) . . . . . –25°C to +85°C

Supply Voltages

V_{CC} Supply Voltages . . . . . +1.70 V to +1.95 V

Notes: *Operating ranges define those limits between which the functionality of the device is guaranteed.*

16.3 Test Conditions

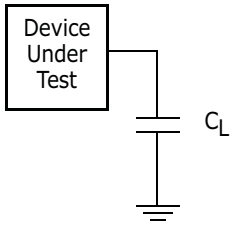


Figure 16.3. Test Setup

Table 16.1 Test Specifications

Test Condition	All Speed Options	Unit
Output Load Capacitance, C_L (including jig capacitance)	30	pF
Input Rise and Fall Times	3.0 @ 54, 66 MHz 2.5 @ 80 MHz	ns
Input Pulse Levels	0.0–V_{CC}	V
Input timing measurement reference levels	$V_{CC}/2$	V
Output timing measurement reference levels	$V_{CC}/2$	V

16.4 Key to Switching Waveforms

Waveform	Inputs	Outputs
	Steady	
	Changing from H to L	
	Changing from L to H	
	Don't Care, Any Change Permitted	Changing, State Unknown
	Does Not Apply	Center Line is High Impedance State (High Z)

16.5 Switching Waveforms

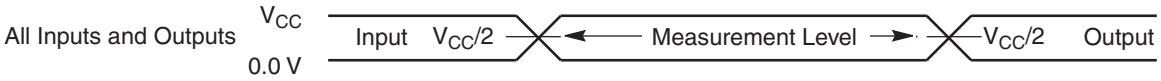


Figure 16.4. Input Waveforms and Measurement Levels

16.6 V_{CC} Power-up

Parameter	Description	Test Setup	Speed	Unit
t_{VCS}	V_{CC} Setup Time	Min	1	ms

Notes:

1. *All V_{CC} signals must be ramped simultaneously to ensure correct power-up.*
2. *S29WS128N: V_{CC} ramp rate is > 1V/ 100 µs and for V_{CC} ramp rate of < 1 V / 100 µs a hardware reset is required.*

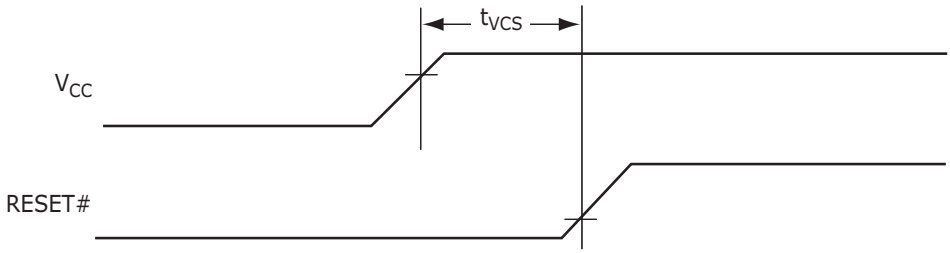


Figure 16.5. V_{CC} Power-up Diagram

16.7 DC Characteristics (CMOS Compatible)

Parameter	Description (Notes)	Test Conditions (Notes 1, 8)		Min	Typ	Max	Unit
I_{LI}	Input Load Current	$V_{IN} = V_{SS}$ to V_{CC}, $V_{CC} = V_{CC}$max				±1	µA
I_{LO}	Output Leakage Current (2)	$V_{OUT} = V_{SS}$ to V_{CC}, $V_{CC} = V_{CC}$max				±1	µA
I_{CCB}	V_{CC} Active burst Read Current	CE# = V_{IL}, OE# = V_{IH}, WE# = V_{IH}, burst length = 8	54 MHz		27	54	mA
			66 MHz		28	60	mA
			80 MHz		30	66	mA
		CE# = V_{IL}, OE# = V_{IH}, WE# = V_{IH}, burst length = 16	54 MHz		28	48	mA
			66 MHz		30	54	mA
			80 MHz		32	60	mA
		CE# = V_{IL}, OE# = V_{IH}, WE# = V_{IH}, burst length = 32	54 MHz		29	42	mA
			66 MHz		32	48	mA
			80 MHz		34	54	mA
		CE# = V_{IL}, OE# = V_{IH}, WE# = V_{IH}, burst length = Continuous	54 MHz		32	36	mA
			66 MHz		35	42	mA
			80 MHz		38	48	mA
I_{CC1}	V_{CC} Active Asynchronous Read Current (3)	CE# = V_{IL}, OE# = V_{IH}, WE# = V_{IH}	10 MHz		34	45	mA
			5 MHz		17	26	mA
			1 MHz		4	7	mA
I_{CC2}	V_{CC} Active Write Current (4)	CE# = V_{IL}, OE# = V_{IH}, ACC = V_{IH}	V_{ACC}		1	5	µA
			V_{CC}		24	52.5	mA
I_{CC3}	V_{CC} Standby Current (5, 6)	CE# = RESET# = V_{CC} ± 0.2 V	V_{ACC}		1	5	µA
			V_{CC}		20	70	µA
I_{CC4}	V_{CC} Reset Current (6)	RESET# = V_{IL}, CLK = V_{IL}			70	250	µA
I_{CC5}	V_{CC} Active Current (Read While Write) (6)	CE# = V_{IL}, OE# = V_{IH}, ACC = V_{IH} @ 5 MHz			50	60	mA
I_{CC6}	V_{CC} Sleep Current (6)	CE# = V_{IL}, OE# = V_{IH}			2	70	µA
I_{CC7}	V_{CC} Page Mode Read Current	OE# = V_{IH}, CE# = V_{IL}				12	mA
I_{ACC}	Accelerated Program Current (7)	CE# = V_{IL}, OE# = V_{IH}, V_{ACC} = 9.5 V	V_{ACC}		6	20	mA
			V_{CC}		14	20	mA
V_{IL}	Input Low Voltage	V_{CC} = 1.8 V		−0.5		0.4	V
V_{IH}	Input High Voltage	V_{CC} = 1.8 V		V_{CC} − 0.4		V_{CC} + 0.4	V
V_{OL}	Output Low Voltage	I_{OL} = 100 µA, $V_{CC} = V_{CC\ min}$				0.1	V
V_{OH}	Output High Voltage	I_{OH} = −100 µA, $V_{CC} = V_{CC\ min}$		V_{CC}			V
V_{HH}	Voltage for Accelerated Program			8.5		9.5	V
V_{LKO}	Low V_{CC} Lock-out Voltage					1.4	V

Notes:

1. *Maximum I_{CC} specifications are tested with $V_{CC} = V_{CC}$max.*
2. *CE# must be set high when measuring the RDY pin.*
3. *The I_{CC} current listed is typically less than 3.5 mA/MHz, with OE# at V_{IH}.*
4. *I_{CC} active while Embedded Erase or Embedded Program is in progress.*
5. *Device enters automatic sleep mode when addresses are stable for t_{ACC} + 20 ns. Typical sleep mode current is equal to I_{CC3}.*
6. *$V_{IH} = V_{CC}$ ± 0.2 V and V_{IL} > −0.1 V.*
7. *Total current during accelerated programming is the sum of V_{ACC} and V_{CC} currents.*
8. *$V_{ACC} = V_{HH}$ on ACC input.*

16.8 AC Characteristics

16.8.1 CLK Characterization

Parameter	Description		54 MHz	66 MHz	80 MHz	Unit
f_{CLK}	CLK Frequency	Max	54	66	80	MHz
t_{CLK}	CLK Period	Min	18.5	15.1	12.5	ns
t_{CH}	CLK High Time	Min	7.4	6.1	5.0	ns
t_{CL}	CLK Low Time					
t_{CR}	CLK Rise Time	Max	3	3	2.5	ns
t_{CF}	CLK Fall Time					

Note: *Not 100% tested.*

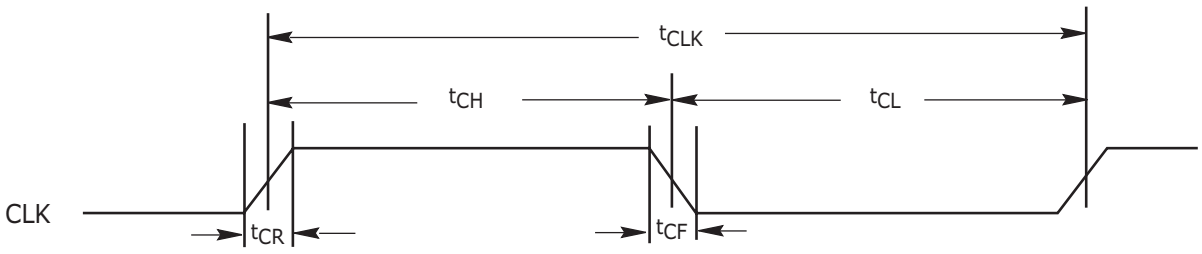


Figure 16.6. CLK Characterization

16.8.2 Synchronous/Burst Read

Parameter		Description		54 MHz	66 MHz	80 MHz	Unit
JEDEC	Standard						
	t_{IACC}	Latency	Max	80			ns
	t_{BACC}	Burst Access Time Valid Clock to Output Delay	Max	13.5	11.2	9	ns
	t_{ACS}	Address Setup Time to CLK (Note 1)	Min	5	4		ns
	t_{ACH}	Address Hold Time from CLK (Note 1)	Min	7	6		ns
	t_{BDH}	Data Hold Time from Next Clock Cycle	Min	4	3		ns
	t_{CR}	Chip Enable to RDY Valid	Max	13.5	11.2	9	ns
	t_{OE}	Output Enable to Output Valid	Max	13.5	11.2		ns
	t_{CEZ}	Chip Enable to High Z (Note 2)	Max	10			ns
	t_{OEZ}	Output Enable to High Z (Note 2)	Max	10			ns
	t_{CES}	CE# Setup Time to CLK	Min	4			ns
	t_{RDYS}	RDY Setup Time to CLK	Min	5	4	3.5	ns
	t_{RACC}	Ready Access Time from CLK	Max	13.5	11.2	8.5	ns
	t_{CAS}	CE# Setup Time to AVD#	Min	0			ns
	t_{AVC}	AVD# Low to CLK	Min	4			ns
	t_{AVD}	AVD# Pulse	Min	7			ns
	t_{AVDH}	AVD# Hold	Min	3			ns
	f_{CLK}	Minimum clock frequency	Min	1	1	1	MHz

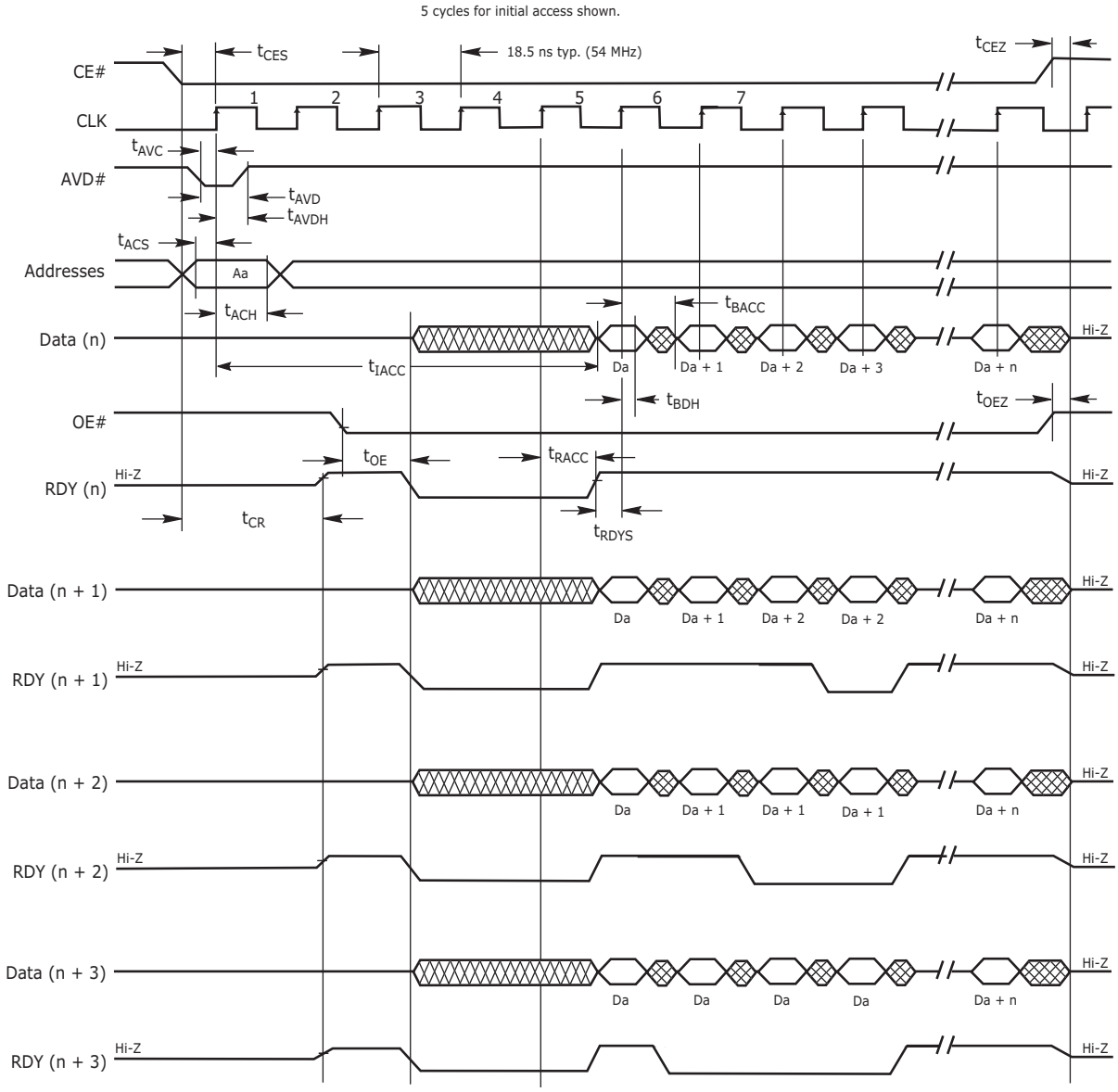
Notes:

1. *Addresses are latched on the first rising edge of CLK.*
2. *Not 100% tested.*

Table 16.2 Synchronous Wait State Requirements

Max Frequency	Wait State Requirement
01 MHz < Freq. ≤ 14 MHz	2
14 MHz < Freq. ≤ 27 MHz	3
27 MHz < Freq. ≤ 40 MHz	4
40 MHz < Freq. ≤ 54 MHz	5
54 MHz < Freq. ≤ 67 MHz	6
67 MHz < Freq. ≤ 80 MHz	7

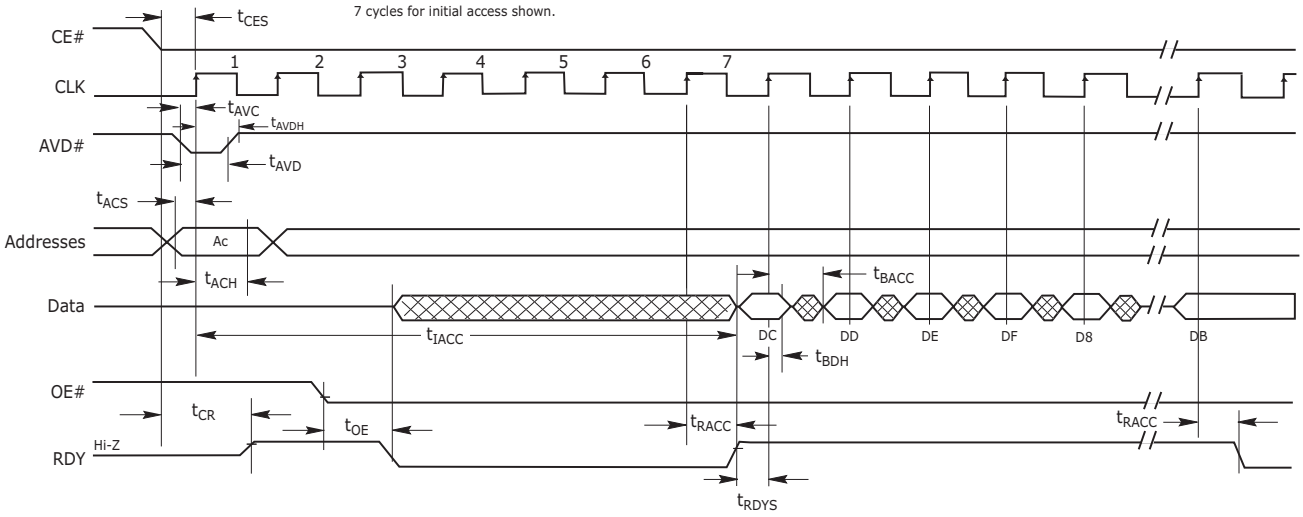
16.8.3 Timing Diagrams



Notes:

1. *Figure shows total number of wait states set to five cycles. The total number of wait states can be programmed from two cycles to seven cycles.*
2. *If any burst address occurs at "address + 1", "address + 2", or "address + 3", additional clock delay cycles are inserted, and are indicated by RDY.*
3. *The device is in synchronous mode.*

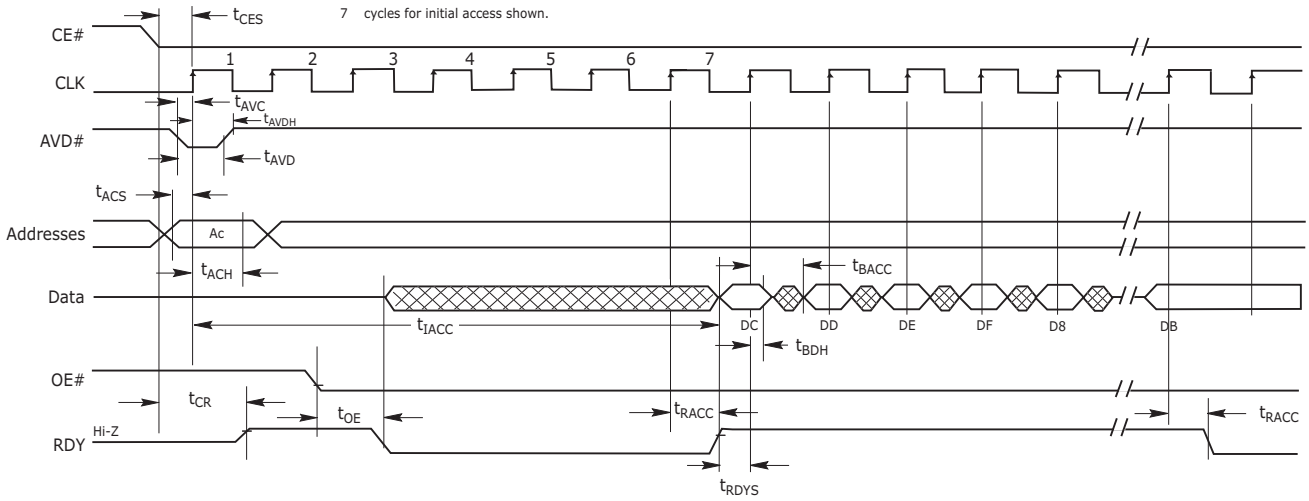
Figure 16.7. CLK Synchronous Burst Mode Read



Notes:

1. *Figure shows total number of wait states set to seven cycles. The total number of wait states can be programmed from two cycles to seven cycles.*
2. *If any burst address occurs at "address + 1", "address + 2", or "address + 3", additional clock delay cycles are inserted, and are indicated by RDY.*
3. *The device is in synchronous mode with wrap around.*
4. *D8–DF in data waveform indicate the order of data within a given 8-word address range, from lowest to highest. Starting address in figure is the 4th address in range (0-F).*

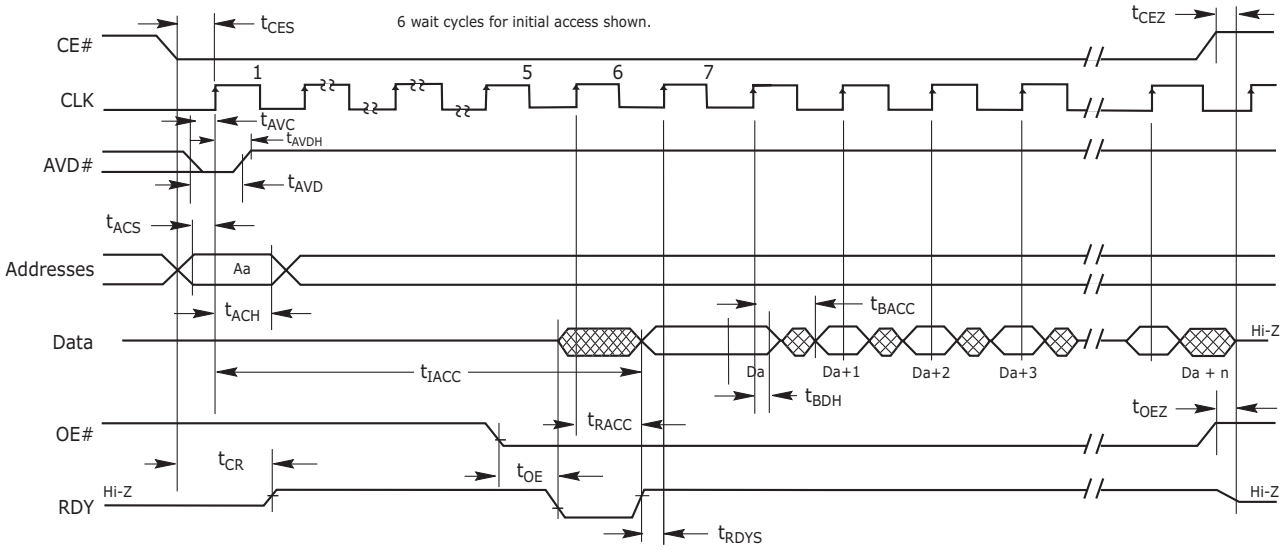
Figure 16.8. 8-word Linear Burst with Wrap Around



Notes:

1. *Figure shows total number of wait states set to seven cycles. The total number of wait states can be programmed from two cycles to seven cycles. Clock is set for active rising edge.*
2. *If any burst address occurs at "address + 1", "address + 2", or "address + 3", additional clock delay cycles are inserted, and are indicated by RDY.*
3. *The device is in asynchronous mode with out wrap around.*
4. *DC–D13 in data waveform indicate the order of data within a given 8-word address range, from lowest to highest. Starting address in figure is the 1st address in range (c-13).*

Figure 16.9. 8-word Linear Burst without Wrap Around



Notes:

1. *Figure assumes 6 wait states for initial access and synchronous read.*
2. *The Set Configuration Register command sequence has been written with CR8=0; device outputs RDY one cycle before valid data.*

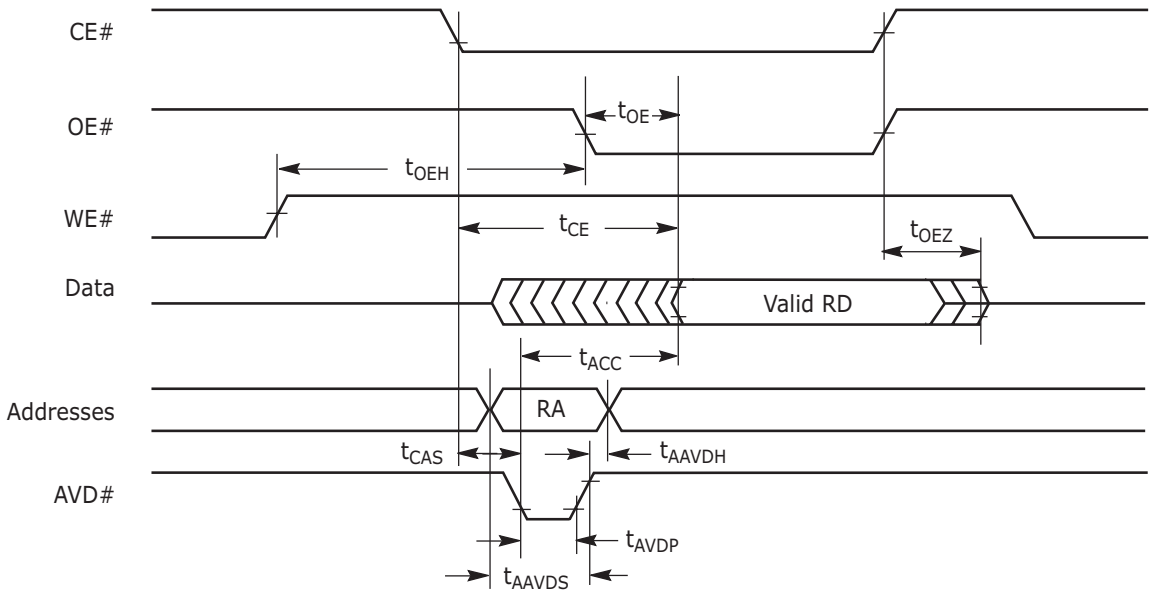
Figure 16.10. Linear Burst with RDY Set One Cycle Before Data

16.8.4 AC Characteristics—Asynchronous Read

Parameter		Description			54 MHz	66 MHz	80 MHz	Unit
JEDEC	Standard							
	t_{CE}	Access Time from CE# Low		Max	80			ns
	t_{ACC}	Asynchronous Access Time		Max	80			ns
	t_{AVDP}	AVD# Low Time		Min	8			ns
	t_{AAVDS}	Address Setup Time to Rising Edge of AVD#		Min	4			ns
	t_{AAVDH}	Address Hold Time from Rising Edge of AVD#		Min	7	6		ns
	t_{OE}	Output Enable to Output Valid		Max	13.5			ns
	t_{OEH}	Output Enable Hold Time	Read	Min	0			ns
			Toggle and Data# Polling	Min	10			ns
	t_{OEZ}	Output Enable to High Z (see Note)		Max	10			ns
	t_{CAS}	CE# Setup Time to AVD#		Min	0			ns
	t_{PACC}	Page Access Time		Max	20			ns
	t_{OH}	Output Hold Time From Addresses, CE# or OE#, whichever occurs first (Note 2)		Min	0			ns
	t_{CEZ}	Chip Enable to Output Tristate		Max	10			ns

Notes:

1. *Not 100% tested.*
2. *t_{OEH} = 1 ns for S29WS128N.*



Note: *RA = Read Address, RD = Read Data.*

Figure 16.11. Asynchronous Mode Read

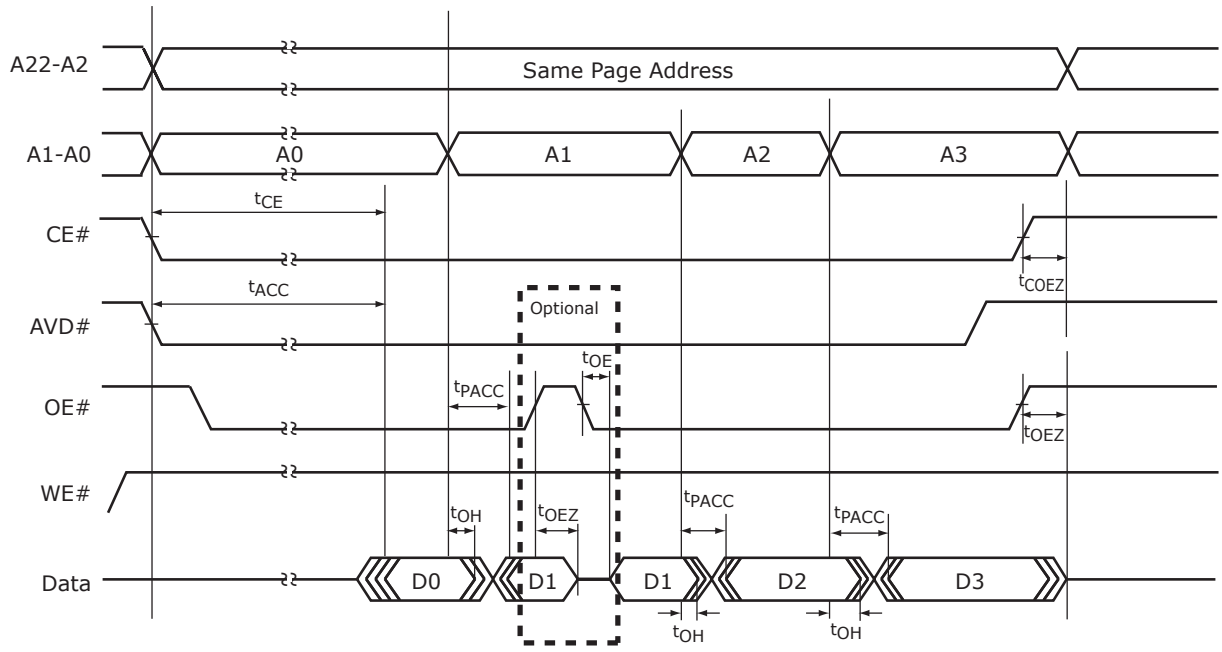


Figure 16.12. Four-Word Page-Mode Operation

16.8.5 Hardware Reset (RESET#)

Parameter		Description		All Speed Options	Unit
JEDEC	Std.				
	t_{RP}	RESET# Pulse Width	Min	30	µs
	t_{RH}	Reset High Time Before Read (See Note)	Min	200	ns

Note: *Not 100% tested.*

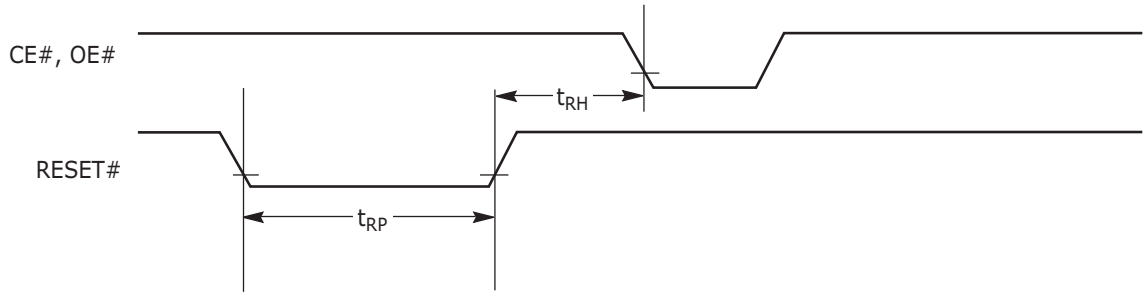


Figure 16.13. Reset Timings

16.8.6 Erase/Program Timing

Parameter		Description			54 MHz	66 MHz	80 MHz	Unit
JEDEC	Standard							
t_{AVAV}	t_{WC}	Write Cycle Time (Note 1)		Min	80			ns
t_{AVWL}	t_{AS}	Address Setup Time (Notes 2, 3)	Synchronous	Min	5			ns
			Asynchronous		0			ns
t_{WLAX}	t_{AH}	Address Hold Time (Notes 2, 3)	Synchronous	Min	9			ns
			Asynchronous		20			
	t_{AVDP}	AVD# Low Time		Min	8			ns
t_{DVWH}	t_{DS}	Data Setup Time		Min	45	20		ns
t_{WHDX}	t_{DH}	Data Hold Time		Min	0			ns
t_{GHWL}	t_{GHWL}	Read Recovery Time Before Write		Min	0			ns
	t_{CAS}	CE# Setup Time to AVD#		Min	0			ns
t_{WHEH}	t_{CH}	CE# Hold Time		Min	0			ns
t_{WLWH}	t_{WP}	Write Pulse Width		Min	30			ns
t_{WHWL}	t_{WPH}	Write Pulse Width High		Min	20			ns
	$t_{SR/W}$	Latency Between Read and Write Operations		Min	0			ns
	t_{VID}	V_{ACC} Rise and Fall Time		Min	500			ns
	t_{VIDS}	V_{ACC} Setup Time (During Accelerated Programming)		Min	1			µs
t_{ELWL}	t_{CS}	CE# Setup Time to WE#		Min	5			ns
	t_{AVSW}	AVD# Setup Time to WE#		Min	5			ns
	t_{AVHW}	AVD# Hold Time to WE#		Min	5			ns
	t_{AVSC}	AVD# Setup Time to CLK		Min	5			ns
	t_{AVHC}	AVD# Hold Time to CLK		Min	5			ns
	t_{CSW}	Clock Setup Time to WE#		Min	5			ns
	t_{WEP}	Noise Pulse Margin on WE#		Max	3			ns
	t_{SEA}	Sector Erase Accept Time-out		Max	50			µs
	t_{ESL}	Erase Suspend Latency		Max	20			µs
	t_{PSL}	Program Suspend Latency		Max	20			µs
	t_{ASP}	Toggle Time During Erase within a Protected Sector		Typ	0			µs
	t_{PSP}	Toggle Time During Programming Within a Protected Sector		Typ	0			µs

Notes:

1. *Not 100% tested.*
2. *Asynchronous read mode allows Asynchronous program operation only. Synchronous read mode allows both Asynchronous and Synchronous program operation.*
3. *In asynchronous program operation timing, addresses are latched on the falling edge of WE#. In synchronous program operation timing, addresses are latched on the rising edge of CLK.*
4. *See the "Erase and Programming Performance" section for more information.*
5. *Does not include the preprogramming time.*

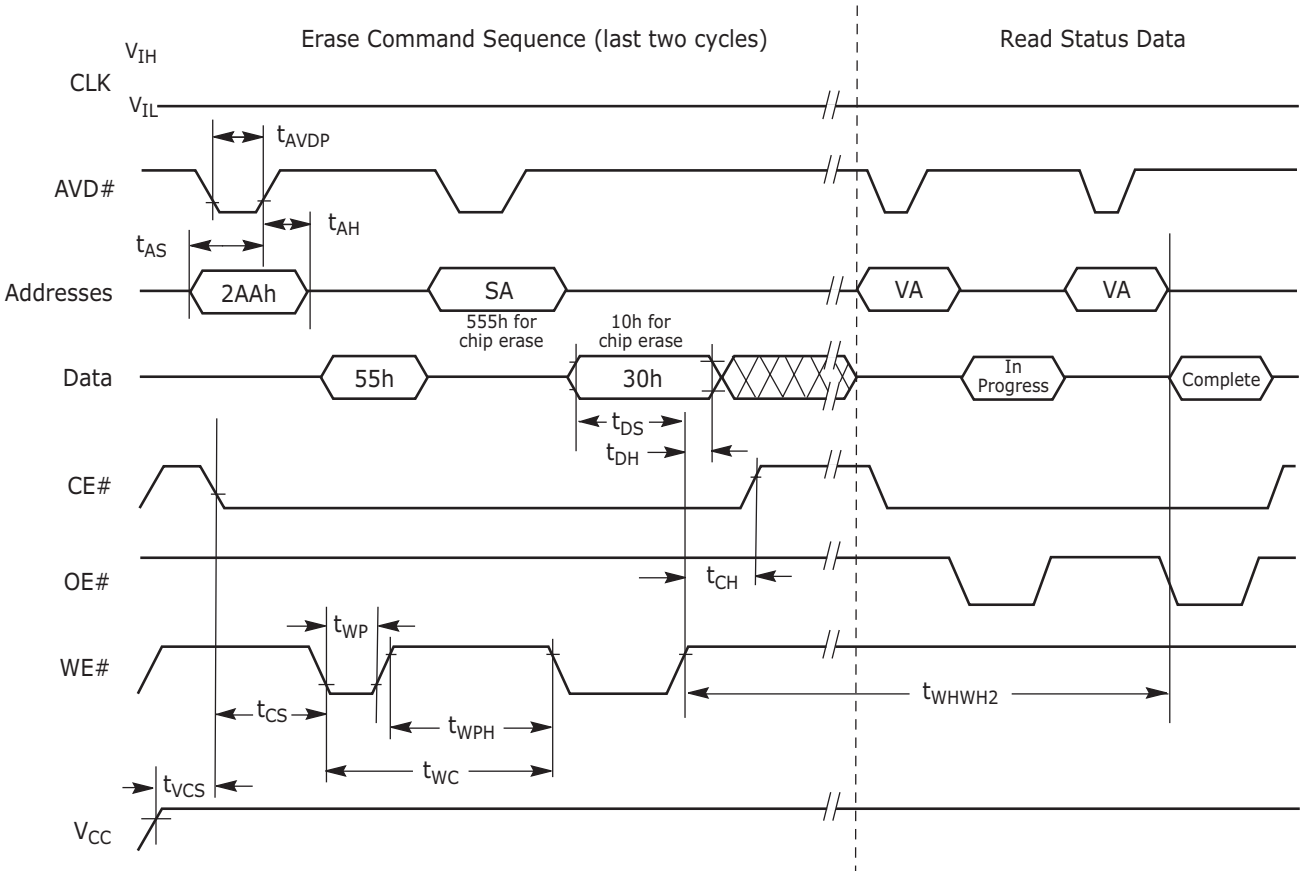
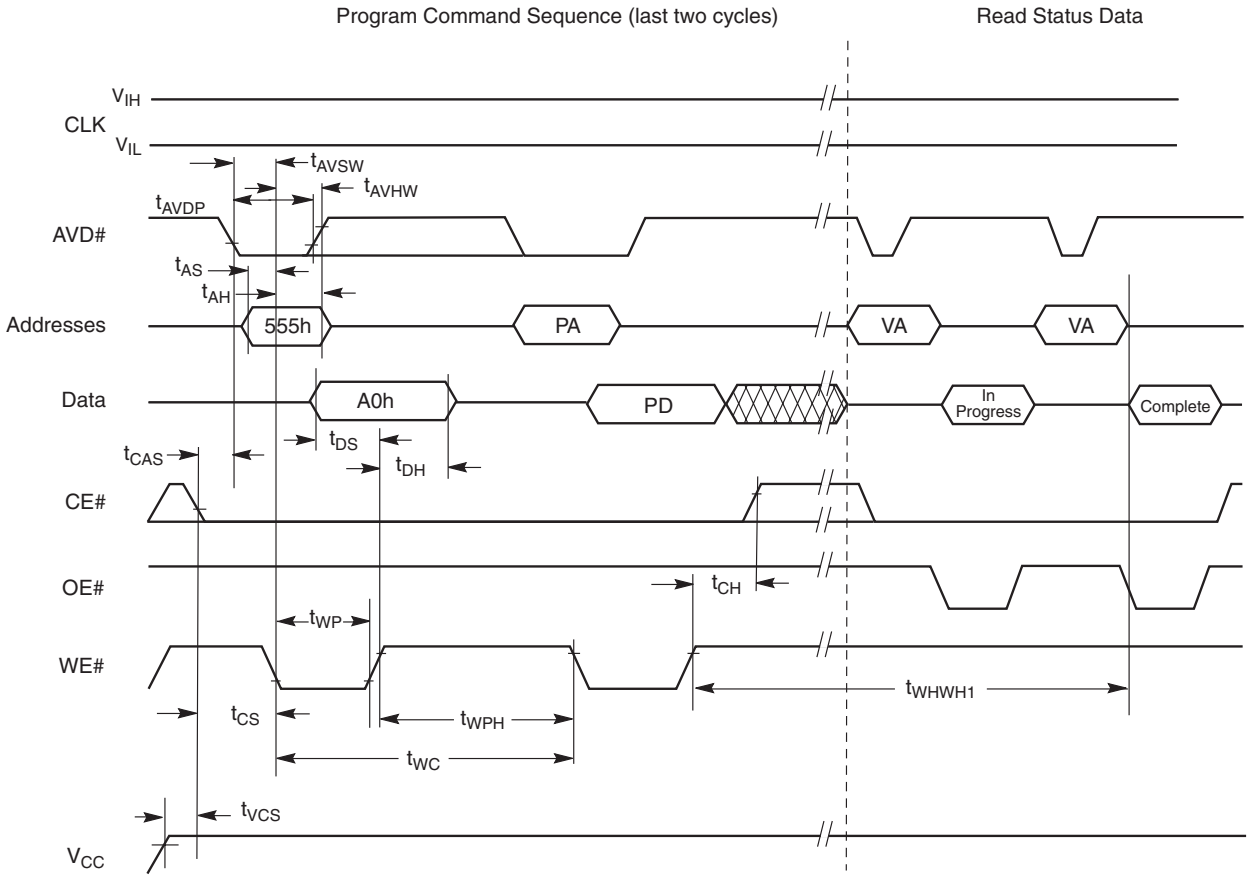


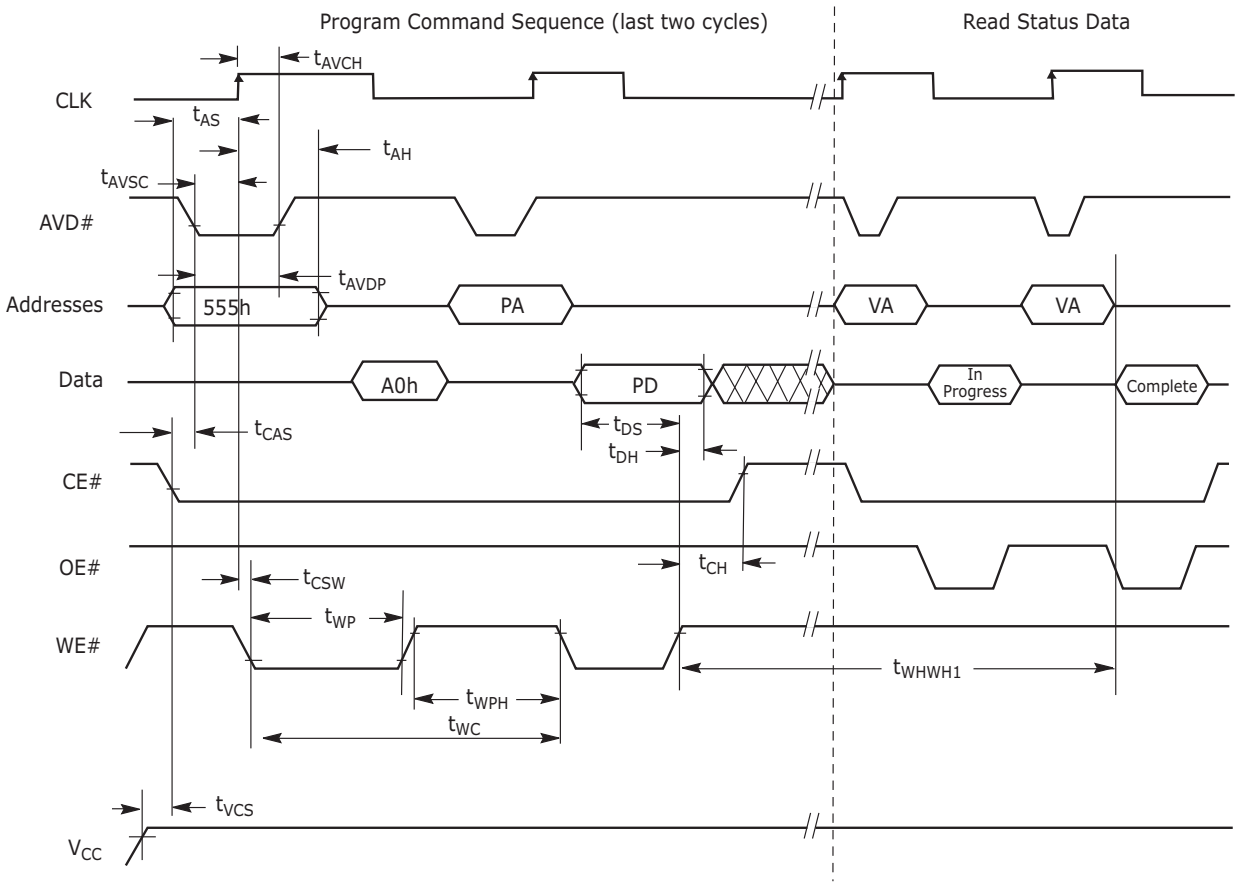
Figure 16.14. Chip/Sector Erase Operation Timings



Notes:

1. *PA = Program Address, PD = Program Data, VA = Valid Address for reading status bits.*
2. *"In progress" and "complete" refer to status of program operation.*
3. *A23–A14 for the WS256N (A22–A14 for the WS128N) are don't care during command sequence unlock cycles.*
4. *CLK can be either V_{IL} or V_{IH}.*
5. *The Asynchronous programming operation is independent of the Set Device Read Mode bit in the Configuration Register.*

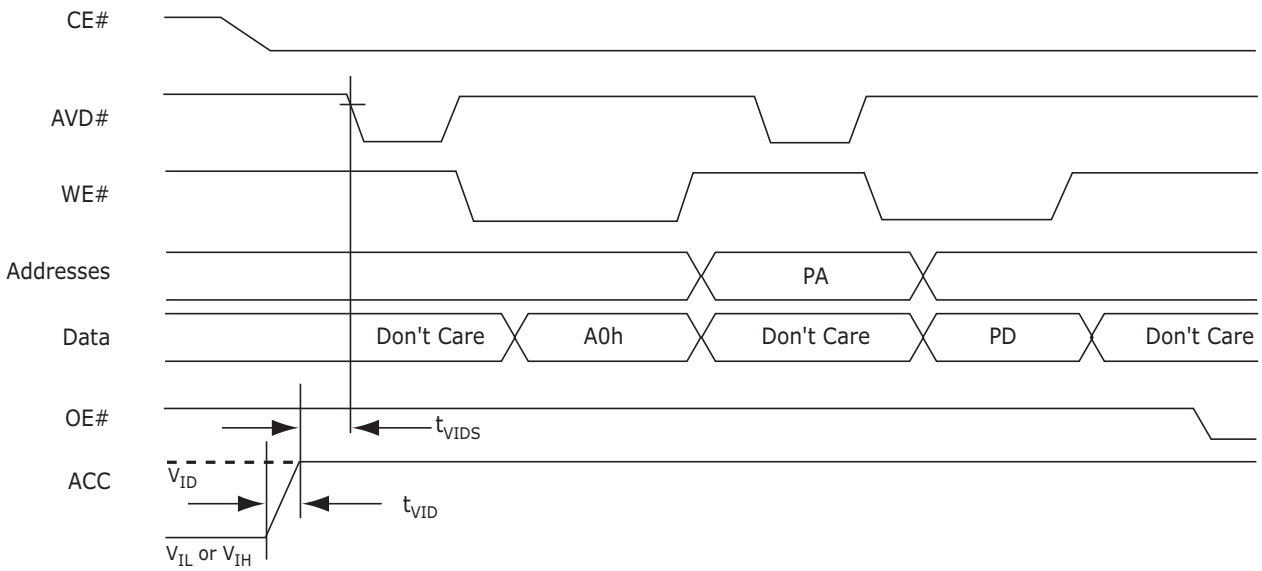
Figure 16.15. Program Operation Timing Using AVD#



Notes:

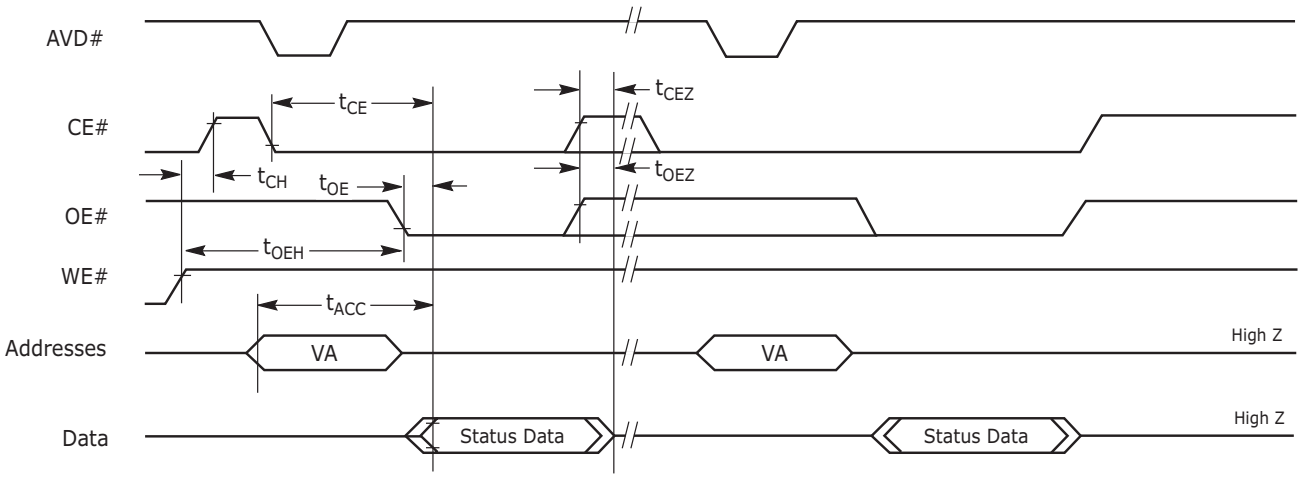
1. *PA = Program Address, PD = Program Data, VA = Valid Address for reading status bits.*
2. *"In progress" and "complete" refer to status of program operation.*
3. *A23–A14 for the WS256N (A22–A14 for the WS128N) are don't care during command sequence unlock cycles.*
4. *Addresses are latched on the first rising edge of CLK.*
5. *Either CE# or AVD# is required to go from low to high in between programming command sequences.*
6. *The Synchronous programming operation is dependent of the Set Device Read Mode bit in the Configuration Register. The Configuration Register must be set to the Synchronous Read Mode.*

Figure 16.16. Program Operation Timing Using CLK in Relationship to AVD#



Note: *Use setup and hold times from conventional program operation.*

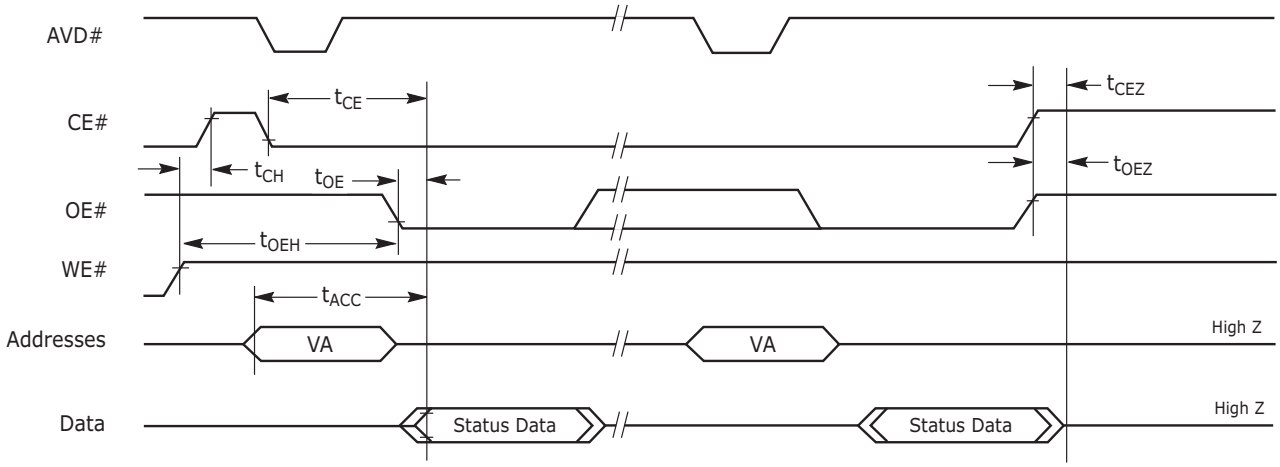
Figure 16.17. Accelerated Unlock Bypass Programming Timing



Notes:

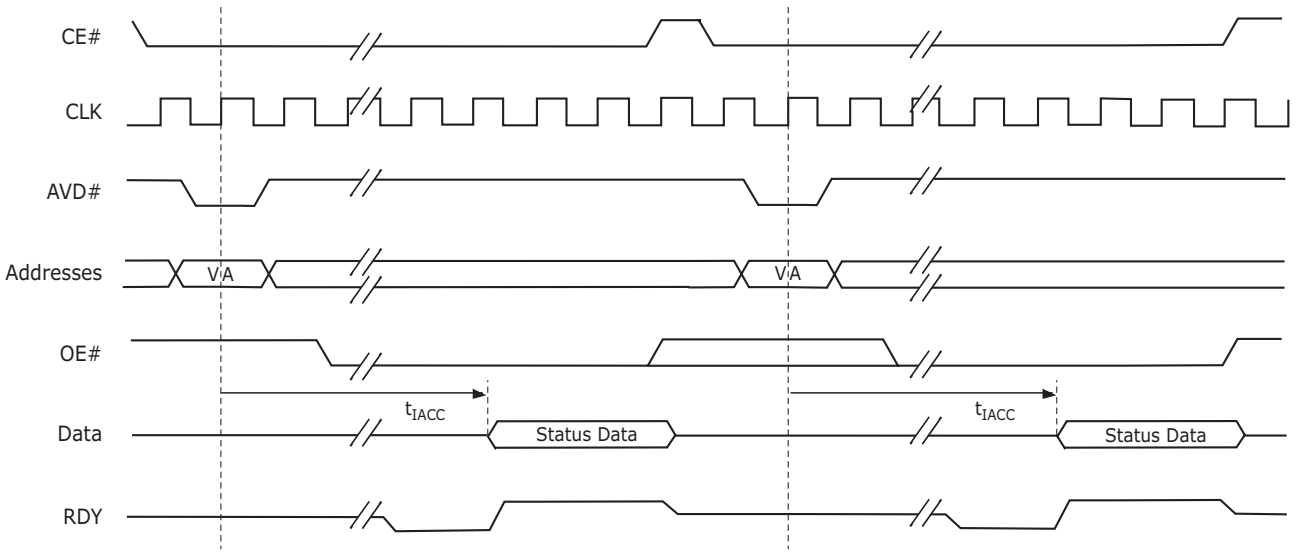
1. *Status reads in figure are shown as asynchronous.*
2. *VA = Valid Address. Two read cycles are required to determine status. When the Embedded Algorithm operation is complete, and Data# Polling outputs true data.*

Figure 16.18. Data# Polling Timings (During Embedded Algorithm)



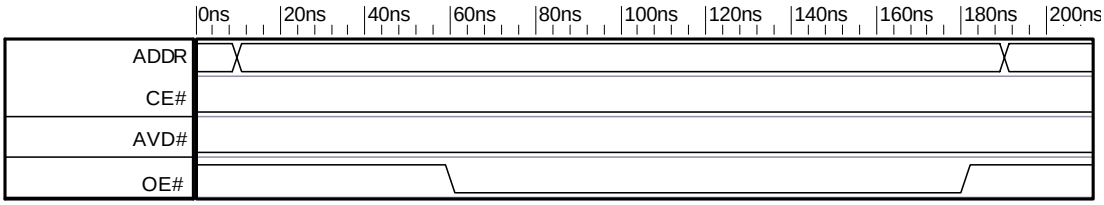
Notes:
1. *Status reads in figure are shown as asynchronous.*
2. *VA = Valid Address. Two read cycles are required to determine status. When the Embedded Algorithm operation is complete, the toggle bits stop toggling.*

Figure 16.19. Toggle Bit Timings (During Embedded Algorithm)



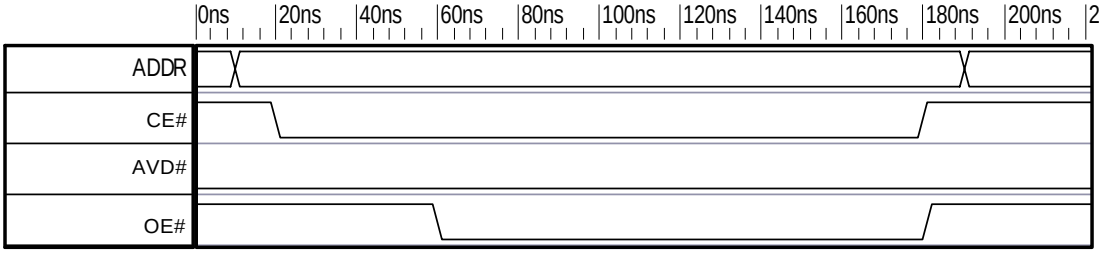
Notes:
1. *The timings are similar to synchronous read timings.*
2. *VA = Valid Address. Two read cycles are required to determine status. When the Embedded Algorithm operation is complete, the toggle bits stop toggling.*
3. *RDY is active with data (D8 = 1 in the Configuration Register). When D8 = 0 in the Configuration Register, RDY is active one clock cycle before data.*

Figure 16.20. Synchronous Data Polling Timings/Toggle Bit Timings



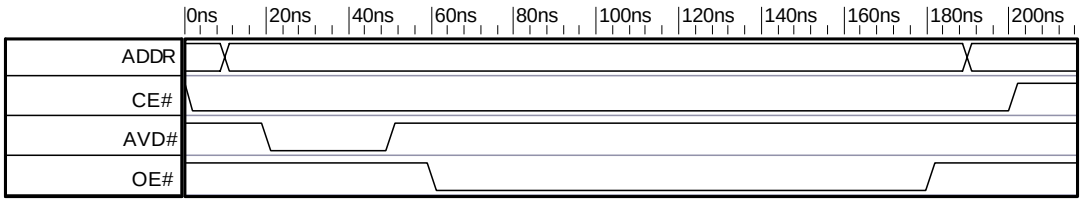
Note: *DQ2 does not toggle correctly during erase suspend if AVD# or CE# are held low after valid address.*

Figure 16.21. Conditions for Incorrect DQ2 Polling During Erase Suspend



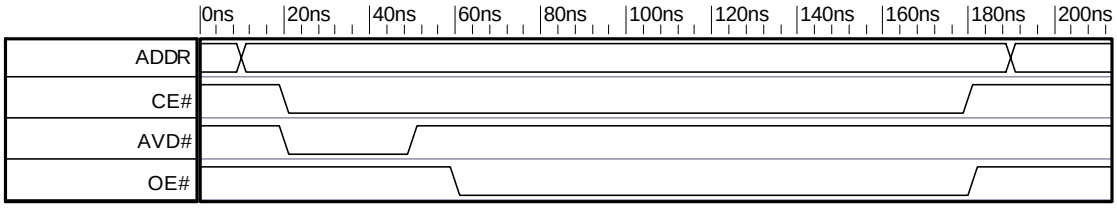
Note: *DQ2 polling during erase suspend behaves normally if CE# pulses low at or after valid Address, even if AVD# does not.*

Figure 16.22. Correct DQ2 Polling during Erase Suspend #I



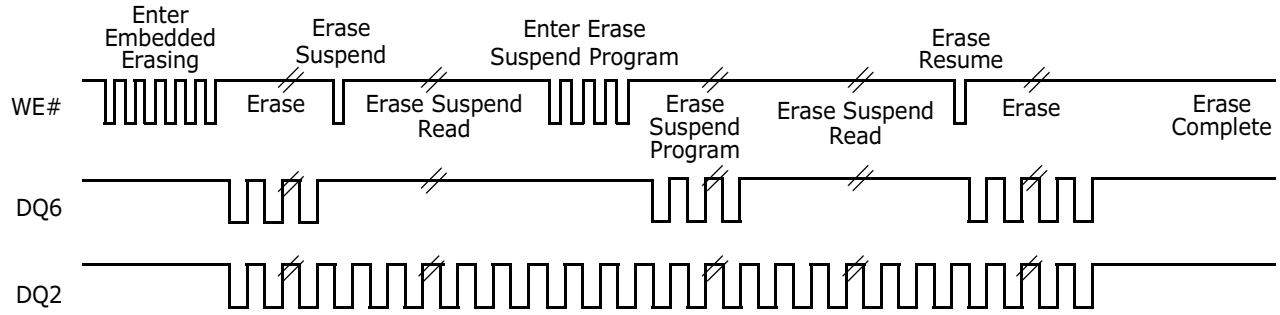
Note: *DQ2 polling during erase suspend behaves normally if AVD# pulses low at or after valid Address, even if CE# does not.*

Figure 16.23. Correct DQ2 Polling during Erase Suspend #2



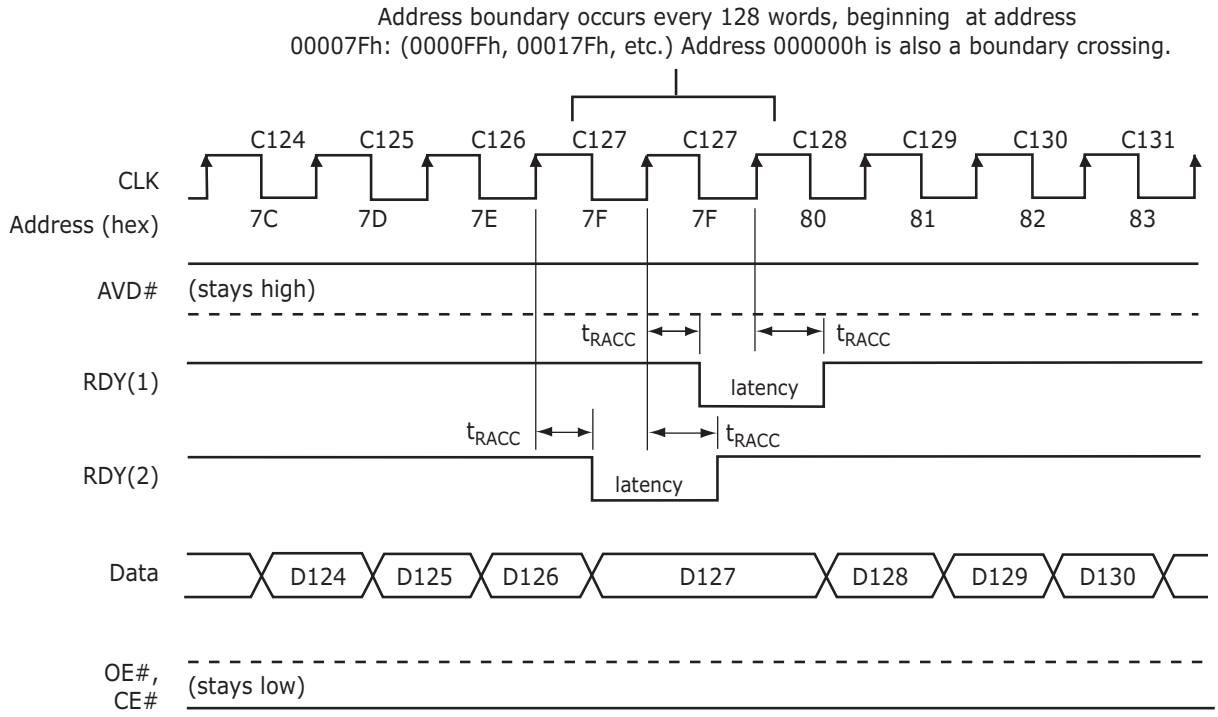
Note: *DQ2 polling during erase suspend behaves normally if both AVD# and CE# pulse low at or after valid Address.*

Figure 16.24. Correct DQ2 Polling during Erase Suspend #3



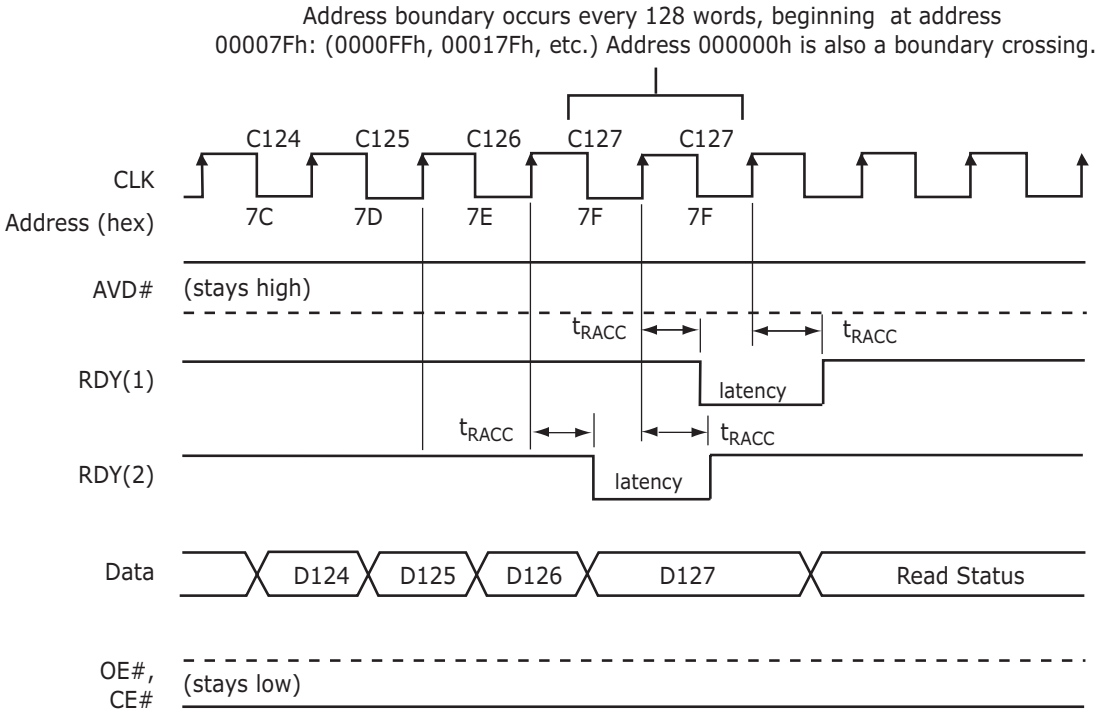
Note: *DQ2 toggles only when read at an address within an erase-suspended sector. The system may use OE# or CE# to toggle DQ2 and DQ6.*

Figure 16.25. DQ2 vs. DQ6



Notes:
1. *RDY(1) active with data (D8 = 1 in the Configuration Register).*
2. *RDY(2) active one clock cycle before data (D8 = 0 in the Configuration Register).*
3. *Cxx indicates the clock that triggers Dxx on the outputs; for example, C60 triggers D60.*
4. *Figure shows the device not crossing a bank in the process of performing an erase or program.*
5. *RDY does not go low and no additional wait states are required for WS ≤ 5.*

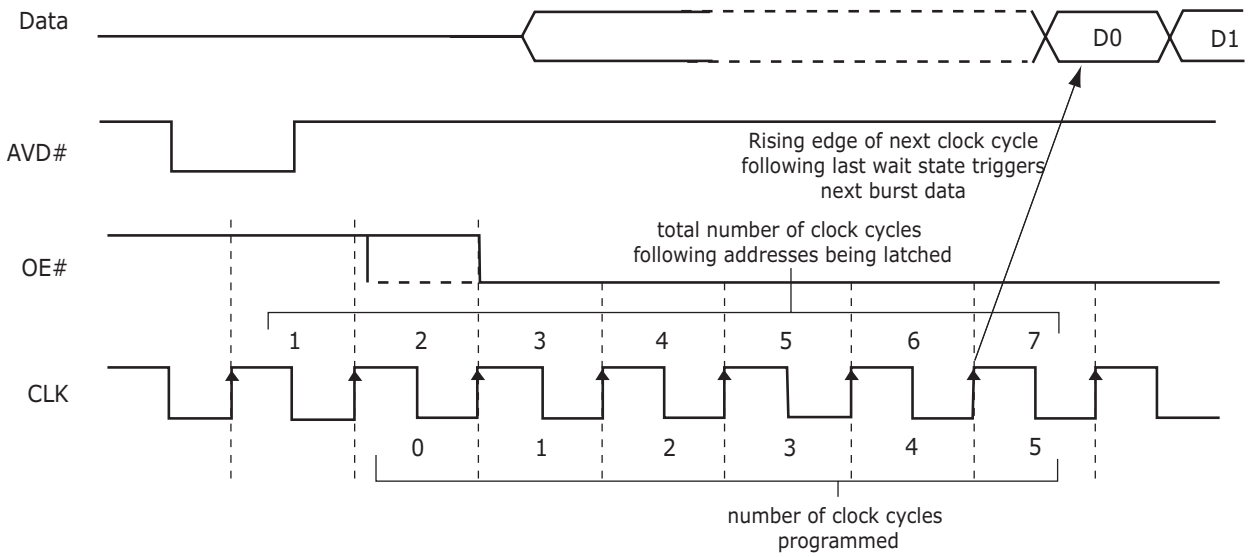
Figure 16.26. Latency with Boundary Crossing when Frequency > 66 MHz



Notes:

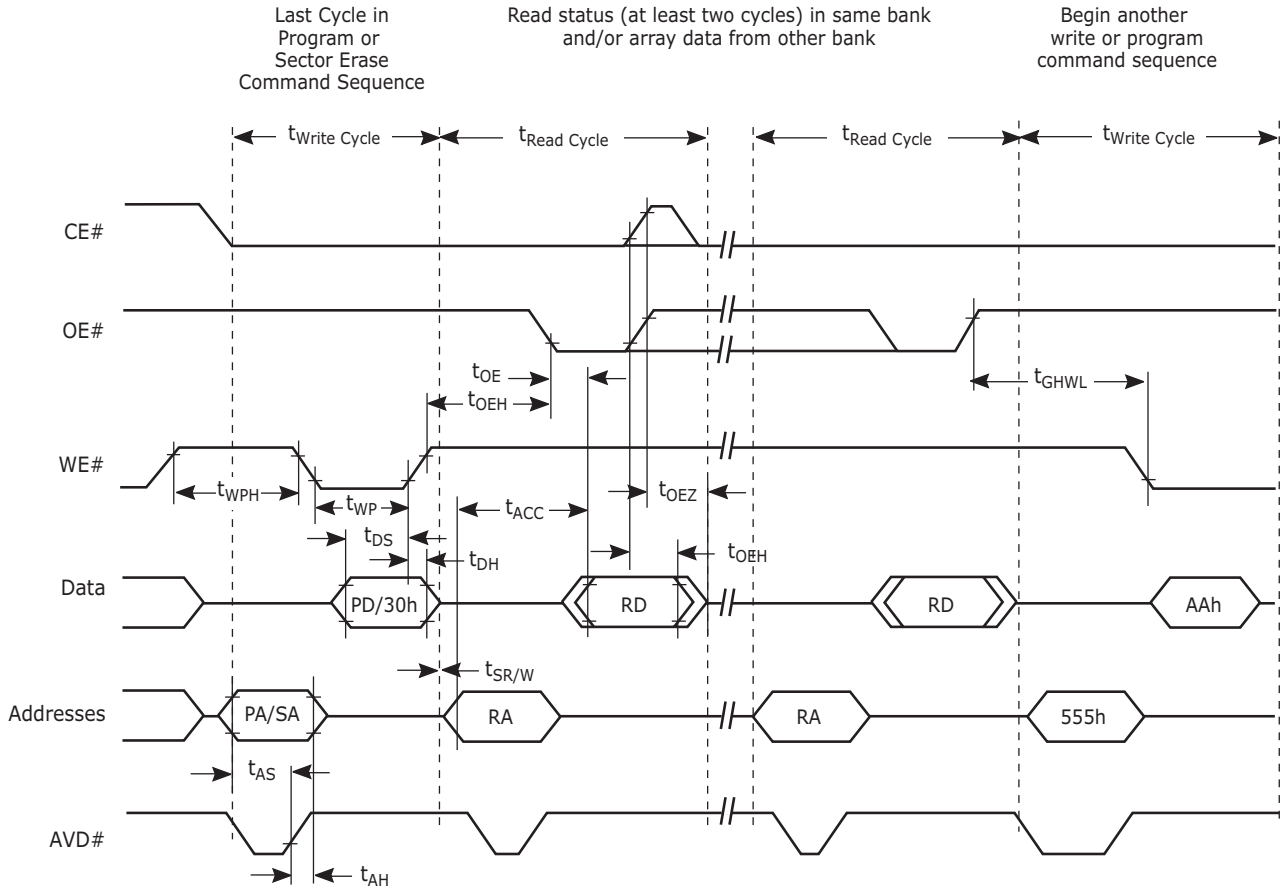
1. *RDY(1) active with data (D8 = 1 in the Configuration Register).*
2. *RDY(2) active one clock cycle before data (D8 = 0 in the Configuration Register).*
3. *Cxx indicates the clock that triggers Dxx on the outputs; for example, C60 triggers D60.*
4. *Figure shows the device crossing a bank in the process of performing an erase or program.*
5. *RDY does not go low and no additional wait states are required for WS ≤ 5.*

Figure 16.27. Latency with Boundary Crossing into Program/Erase Bank



Wait State Configuration Register Setup:
D13, D12, D11 = "111" ⇒ Reserved
D13, D12, D11 = "110" ⇒ Reserved
D13, D12, D11 = "101" ⇒ 5 programmed, 7 total
D13, D12, D11 = "100" ⇒ 4 programmed, 6 total
D13, D12, D11 = "011" ⇒ 3 programmed, 5 total
Note: *Figure assumes address D0 is not at an address boundary, and wait state is set to "101".*

Figure 16.28. Example of Wait States Insertion



Note: *Breakpoints in waveforms indicate that system may alternately read array data from the "non-busy bank" while checking the status of the program or erase operation in the "busy" bank. The system should read status twice to ensure valid information.*

Figure 16.29. Back-to-Back Read/Write Cycle Timings

16.8.7 Erase and Programming Performance

Parameter			Typ (Note 1)	Max (Note 2)	Unit	Comments
Sector Erase Time	64 Kword	V_{CC}	0.6	3.5	s	Excludes 00h programming prior to erasure (Note 4)
	16 Kword	V_{CC}	<0.15	2		
Chip Erase Time		V_{CC}	153.6 (WS256N) 77.4 (WS128N)	308 (WS256N) 154 (WS128N)	s	
		ACC	130.6 (WS256N) 65.8 (WS128N)	262 (WS256N) 132 (WS128N)		
Single Word Programming Time (Note 7)		V_{CC}	40	400	µs	
		ACC	24	240		
Effective Word Programming Time utilizing Program Write Buffer		V_{CC}	9.4	94	µs	
		ACC	6	60		
Total 32-Word Buffer Programming Time		V_{CC}	300	3000	µs	
		ACC	192	1920		
Chip Programming Time (Note 3)		V_{CC}	157.3 (WS256N) 78.6 (WS128N)	314.6 (WS256N) 157.3 (WS128N)	s	Excludes system level overhead (Note 5)
		ACC	100.7 (WS256N) 50.3 (WS128N)	201.3 (WS256N) 100.7 (WS128N)		

Notes:

1. *Typical program and erase times assume the following conditions: 25°C, 1.8 V V_{CC}, 10,000 cycles; checkerboard data pattern.*
2. *Under worst case conditions of 90°C, V_{CC} = 1.70 V, 100,000 cycles.*
3. *Typical chip programming time is considerably less than the maximum chip programming time listed, and is based on utilizing the Write Buffer.*
4. *In the pre-programming step of the Embedded Erase algorithm, all words are programmed to 00h before erasure.*
5. *System-level overhead is the time required to execute the two- or four-bus-cycle sequence for the program command. See the Appendix for further information on command definitions.*
6. *Refer to Application Note "Erase Suspend/Resume Timing" for more details.*
7. *Word programming specification is based upon a single word programming operation not utilizing the write buffer.*

16.8.8 BGA Ball Capacitance

Parameter Symbol	Parameter Description	Test Setup	Typ.	Max	Unit
C_{IN}	Input Capacitance	V_{IN} = 0	5.3	6.3	pF
C_{OUT}	Output Capacitance	V_{OUT} = 0	5.8	6.8	pF
C_{IN2}	Control Pin Capacitance	V_{IN} = 0	6.3	7.3	pF

Notes:

1. *Sampled, not 100% tested.*
2. *Test conditions T_A = 25°C; f = 1.0 MHz.*

17 Appendix

This section contains information relating to software control or interfacing with the Flash device. For additional information and assistance regarding software, see the Additional Resources section on page 25, or explore the Web at www.amd.com and www.fujitsu.com.

Table 17.1 Memory Array Commands

Command Sequence (Notes)		Cycles	Bus Cycles (Notes 1–5)											
			First		Second		Third		Fourth		Fifth		Sixth	
			Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Asynchronous Read (6)		1	RA	RD										
Reset (7)		1	XXX	F0										
Auto-select (8)	Manufacturer ID	4	555	AA	2AA	55	[BA]555	90	[BA]X00	0001				
	Device ID (9)	6	555	AA	2AA	55	[BA]555	90	[BA]X01	227E	BA+X0E	Data	BA+X0F	2200
	Indicator Bits (10)	4	555	AA	2AA	55	[BA]555	90	[BA]X03	Data				
Program		4	555	AA	2AA	55	555	A0	PA	PD				
Write to Buffer (11)		6	555	AA	2AA	55	PA	25	PA	WC	PA	PD	WBL	PD
Program Buffer to Flash		1	SA	29										
Write to Buffer Abort Reset (12)		3	555	AA	2AA	55	555	F0						
Chip Erase		6	555	AA	2AA	55	555	80	555	AA	2AA	55	555	10
Sector Erase		6	555	AA	2AA	55	555	80	555	AA	2AA	55	SA	30
Erase/Program Suspend (13)		1	BA	B0										
Erase/Program Resume (14)		1	BA	30										
Set Configuration Register (18)		4	555	AA	2AA	55	555	D0	X00	CR				
Read Configuration Register		4	555	AA	2AA	55	555	C6	X00	CR				
CFI Query (15)		1	[BA]555	98										
Unlock Bypass Mode	Entry	3	555	AA	2AA	55	555	20						
	Program (16)	2	XXX	A0	PA	PD								
	CFI (16)	1	XXX	98										
	Reset	2	XXX	90	XXX	00								
Secured Silicon Sector	Entry	3	555	AA	2AA	55	555	88						
	Program (17)	4	555	AA	2AA	55	555	A0	PA	PD				
	Read (17)	1	SA	Data										
	Exit (17)	4	555	AA	2AA	55	555	90	XXX	00				

Legend:

X = Don't care.

RA = Read Address.

RD = Read Data.

PA = Program Address. Addresses latch on the rising edge of the AVD# pulse or active edge of CLK, whichever occurs first.

PD = Program Data. Data latches on the rising edge of WE# or CE# pulse, whichever occurs first.

SA = Sector Address. WS256N = A23–A14; WS128N = A22–A14.

BA = Bank Address. WS256N = A23–A20; WS128N = A22–A20.

CR = Configuration Register data bits D15–D0.

WBL = Write Buffer Location. Address must be within the same write buffer page as PA.

WC = Word Count. Number of write buffer locations to load minus 1.

Notes:

1. *See Table 12.1 for description of bus operations.*
2. *All values are in hexadecimal.*
3. *Shaded cells indicate read cycles.*
4. *Address and data bits not specified in table, legend, or notes are don't cares (each hex digit implies 4 bits of data).*
5. *Writing incorrect address and data values or writing them in the improper sequence may place the device in an unknown state. The system must write the reset command to return the device to reading array data.*
6. *No unlock or command cycles required when bank is reading array data.*
7. *Reset command is required to return to reading array data (or to the erase-suspend-read mode if previously in Erase Suspend) when a bank is in the autoselect mode, or if DQ5 goes high (while the bank is providing status information) or performing sector lock/unlock.*
8. *The system must provide the bank address. See Autoselect section for more information.*
9. *Data in cycle 5 is 2230 (WS256N) or 2231 (WS128N).*
10. *See Table 12.10 for indicator bit values.*
11. *Total number of cycles in the command sequence is determined by the number of words written to the write buffer.*
12. *Command sequence resets device for next command after write-to-buffer operation.*
13. *System may read and program in non-erasing sectors, or enter the autoselect mode, when in the Erase Suspend mode. The Erase Suspend command is valid only during a sector erase operation, and requires the bank address.*
14. *Erase Resume command is valid only during the Erase Suspend mode, and requires the bank address.*
15. *Command is valid when device is ready to read array data or when device is in autoselect mode. Address equals 55h on all future devices, but 555h for WS256N/128N.*
16. *Requires Entry command sequence prior to execution. Unlock Bypass Reset command is required to return to reading array data.*
17. *Requires Entry command sequence prior to execution. Secured Silicon Sector Exit Reset command is required to exit this mode; device may otherwise be placed in an unknown state.*
18. *Requires reset command to configure the Configuration Register.*

Table 17.2. Sector Protection Commands

Command Sequence (Notes)		Cycles	Bus Cycles (Notes 1–4)													
			First		Second		Third		Fourth		Fifth		Sixth		Seventh	
			Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Lock Register Bits	Command Set Entry (5)	3	555	AA	2AA	55	555	40								
	Program (6, 12)	2	XX	A0	77/00	data										
	Read (6)	1	77	data												
	Command Set Exit (7)	2	XX	90	XX	00										
Password Protection	Command Set Entry (5)	3	555	AA	2AA	55	555	60								
	Program [0-3] (8)	2	XX	A0	00	PWD[0-3]										
	Read (9)	4	0...00	PWD0	0...01	PWD1	0...02	PWD2	0...03	PWD3						
	Unlock	7	00	25	00	03	00	PWD0	01	PWD1	02	PWD2	03	PWD3	00	29
	Command Set Exit (7)	2	XX	90	XX	00										
Non-Volatile Sector Protection (PPB)	Command Set Entry (5)	3	555	AA	2AA	55	[BA]555	C0								
	PPB Program (10)	2	XX	A0	SA	00										
	All PPB Erase (10, 11)	2	XX	80	00	30										
	PPB Status Read	1	SA	RD(0)												
	Command Set Exit (7)	2	XX	90	XX	00										
Global Volatile Sector Protection Freeze (PPB Lock)	Command Set Entry (5)	3	555	AA	2AA	55	[BA]555	50								
	PPB Lock Bit Set	2	XX	A0	XX	00										
	PPB Lock Bit Status Read	1	BA	RD(0)												
	Command Set Exit (7)	2	XX	90	XX	00										
Volatile Sector Protection (DYB)	Command Set Entry (5)	3	555	AA	2AA	55	[BA]555	E0								
	DYB Set	2	XX	A0	SA	00										
	DYB Clear	2	XX	A0	SA	01										
	DYB Status Read	1	SA	RD(0)												
	Command Set Exit (7)	2	XX	90	XX	00										

Legend:

X = Don't care.

RA = Address of the memory location to be read.

PD(0) = Secured Silicon Sector Lock Bit. PD(0), or bit[0].

PD(1) = Persistent Protection Mode Lock Bit. PD(1), or bit[1], must be set to '0' for protection while PD(2), bit[2] must be left as '1'.

PD(2) = Password Protection Mode Lock Bit. PD(2), or bit[2], must be set to '0' for protection while PD(1), bit[1] must be left as '1'.

PD(3) = Protection Mode OTP Bit. PD(3) or bit[3].

SA = Sector Address. WS256N = A23–A14; WS128N = A22–A14.

BA = Bank Address. WS256N = A23–A20; WS128N = A22–A20.

PWD3–PWD0 = Password Data. PD3–PD0 present four 16 bit combinations that represent the 64-bit Password

PWA = Password Address. Address bits A1 and A0 are used to select each 16-bit portion of the 64-bit entity.

PWD = Password Data.

RD(0), RD(1), RD(2) = DQ0, DQ1, or DQ2 protection indicator bit. If protected, DQ0, DQ1, or DQ2 = 0. If unprotected, DQ0, DQ1, DQ2 = 1.

Notes:

1. *All values are in hexadecimal.*
2. *Shaded cells indicate read cycles.*
3. *Address and data bits not specified in table, legend, or notes are don't cares (each hex digit implies 4 bits of data).*
4. *Writing incorrect address and data values or writing them in the improper sequence may place the device in an unknown state. The system must write the reset command to return the device to reading array data.*
5. *Entry commands are required to enter a specific mode to enable instructions only available within that mode.*
6. *If both the Persistent Protection Mode Locking Bit and the Password Protection Mode Locking Bit are set at the same time, the command operation aborts and returns the device to the default Persistent Sector Protection Mode during 2nd bus cycle. Note that on all future devices, addresses equal 00h, but is currently 77h for the WS256N only. See Tables 13.1 and 13.2 for explanation of lock bits.*
7. *Exit command must be issued to reset the device into read mode; device may otherwise be placed in an unknown state.*
8. *Entire two bus-cycle sequence must be entered for each portion of the password.*
9. *Full address range is required for reading password.*
10. *See Figure 13.2 for details.*
11. *"All PPB Erase" command pre-programs all PPBs before erasure to prevent over-erasure.*
12. *The second cycle address for the lock register program operation is 77 for S29WS256N; however, for WS128N this address is 00.*

17.1 Common Flash Memory Interface

The Common Flash Interface (CFI) specification outlines device and host system software interrogation handshake, which allows specific vendor-specified soft-ware algorithms to be used for entire families of devices. Software support can then be device-independent, JEDEC ID-independent, and forward- and back-ward-compatible for the specified flash device families. Flash vendors can standardize their existing interfaces for long-term compatibility.

This device enters the CFI Query mode when the system writes the CFI Query command, 98h, to address (BA)555h any time the device is ready to read array data. The system can read CFI information at the addresses given in Tables 17.3–17.6) within that bank. All reads outside of the CFI address range, within the bank, returns non-valid data. Reads from other banks are allowed, writes are not. To terminate reading CFI data, the system must write the reset command.

The following is a C source code example of using the CFI Entry and Exit functions. Refer to the *Spansion Low Level Driver User's Guide* (available on www.amd.com and www.fujitsu.com) for general information on Spansion Flash memory software development guidelines.

```
/* Example: CFI Entry command */
  *( (UINT16 *)bank_addr + 0x555 ) = 0x0098;   /* write CFI entry command         */

/* Example: CFI Exit command  */
  *( (UINT16 *)bank_addr + 0x000 ) = 0x00F0;   /* write cfi exit command          */
```

For further information, please refer to the CFI Specification (see JEDEC publications JEP137-A and JESD68.01and CFI Publication 100). Please contact your sales office for copies of these documents.

Table 17.3. CFI Query Identification String

Addresses	Data	Description
10h 11h 12h	0051h 0052h 0059h	Query Unique ASCII string "QRY"
13h 14h	0002h 0000h	Primary OEM Command Set
15h 16h	0040h 0000h	Address for Primary Extended Table
17h 18h	0000h 0000h	Alternate OEM Command Set (00h = none exists)
19h 1Ah	0000h 0000h	Address for Alternate OEM Extended Table (00h = none exists)

Table 17.4. System Interface String

Addresses	Data	Description
1Bh	0017h	V_{CC} Min. (write/erase) D7–D4: volt, D3–D0: 100 millivolt
1Ch	0019h	V_{CC} Max. (write/erase) D7–D4: volt, D3–D0: 100 millivolt
1Dh	0000h	V_{PP} Min. voltage (00h = no V_{PP} pin present)
1Eh	0000h	V_{PP} Max. voltage (00h = no V_{PP} pin present)
1Fh	0006h	Typical timeout per single byte/word write 2^N µs
20h	0009h	Typical timeout for Min. size buffer write 2^N µs (00h = not supported)
21h	000Ah	Typical timeout per individual block erase 2^N ms
22h	0000h	Typical timeout for full chip erase 2^N ms (00h = not supported)
23h	0004h	Max. timeout for byte/word write 2^N times typical
24h	0004h	Max. timeout for buffer write 2^N times typical
25h	0003h	Max. timeout per individual block erase 2^N times typical
26h	0000h	Max. timeout for full chip erase 2^N times typical (00h = not supported)

Table I7.5. Device Geometry Definition

Addresses	Data	Description
27h	0019h (WS256N) 0018h (WS128N)	Device Size = 2^N byte
28h 29h	0001h 0000h	Flash Device Interface description
2Ah 2Bh	0006h 0000h	Max. number of bytes in multi-byte write = 2^N (00h = not supported)
2Ch	0003h	Number of Erase Block Regions within device
2Dh 2Eh 2Fh 30h	0003h 0000h 0080h 0000h	Erase Block Region 1 Information
31h	00FDh (WS256N) 007Dh (WS128N)	Erase Block Region 2 Information
32h 33h 34h	0000h 0000h 0002h	
35h 36h 37h 38h	0003h 0000h 0080h 0000h	Erase Block Region 3 Information
39h 3Ah 3Bh 3Ch	0000h 0000h 0000h 0000h	Erase Block Region 4 Information

Table I7.6. Primary Vendor-Specific Extended Query

Addresses	Data	Description
40h 41h 42h	0050h 0052h 0049h	Query-unique ASCII string "PRI"
43h	0031h	Major version number, ASCII
44h	0034h	Minor version number, ASCII
45h	0100h	Address Sensitive Unlock (Bits 1-0) 0 = Required, 1 = Not Required Silicon Technology (Bits 5-2) 0100 = 0.11 µm
46h	0002h	Erase Suspend 0 = Not Supported, 1 = To Read Only, 2 = To Read & Write
47h	0001h	Sector Protect 0 = Not Supported, X = Number of sectors in per group
48h	0000h	Sector Temporary Unprotect 00 = Not Supported, 01 = Supported
49h	0008h	Sector Protect/Unprotect scheme 08 = Advanced Sector Protection
4Ah	00F3h (WS256N) 007Bh (WS128N)	Simultaneous Operation Number of Sectors in all banks except boot bank
4Bh	0001h	Burst Mode Type 00 = Not Supported, 01 = Supported
4Ch	0000h	Page Mode Type 00 = Not Supported, 01 = 4 Word Page, 02 = 8 Word Page, 04 = 16 Word Page
4Dh	0085h	ACC (Acceleration) Supply Minimum 00h = Not Supported, D7-D4: Volt, D3-D0: 100 mV
4Eh	0095h	ACC (Acceleration) Supply Maximum 00h = Not Supported, D7-D4: Volt, D3-D0: 100 mV
4Fh	0001h	Top/Bottom Boot Sector Flag 0001h = Dual Boot Device
50h	0001h	Program Suspend. 00h = not supported
51h	0001h	Unlock Bypass 00 = Not Supported, 01=Supported
52h	0007h	Secured Silicon Sector (Customer OTP Area) Size 2^N bytes
53h	0014h	Hardware Reset Low Time-out during an embedded algorithm to read mode Maximum 2^N ns
54h	0014h	Hardware Reset Low Time-out not during an embedded algorithm to read mode Maximum 2^N ns
55h	0005h	Erase Suspend Time-out Maximum 2^N ns
56h	0005h	Program Suspend Time-out Maximum 2^N ns
57h	0010h	Bank Organization: X = Number of banks
58h	0013h (WS256N) 000Bh (WS128N)	Bank 0 Region Information. X = Number of sectors in bank
59h	0010h (WS256N) 0008h (WS128N)	Bank 1 Region Information. X = Number of sectors in bank

Table I7.6. Primary Vendor-Specific Extended Query (Continued)

Addresses	Data	Description
5Ah	0010h (WS256N) 0008h (WS128N)	Bank 2 Region Information. X = Number of sectors in bank
5Bh	0010h (WS256N) 0008h (WS128N)	Bank 3 Region Information. X = Number of sectors in bank
5Ch	0010h (WS256N) 0008h (WS128N)	Bank 4 Region Information. X = Number of sectors in bank
5Dh	0010h (WS256N) 0008h (WS128N)	Bank 5 Region Information. X = Number of sectors in bank
5Eh	0010h (WS256N) 0008h (WS128N)	Bank 6 Region Information. X = Number of sectors in bank
5Fh	0010h (WS256N) 0008h (WS128N)	Bank 7 Region Information. X = Number of sectors in bank
60h	0010h (WS256N) 0008h (WS128N)	Bank 8 Region Information. X = Number of sectors in bank
61h	0010h (WS256N) 0008h (WS128N)	Bank 9 Region Information. X = Number of sectors in bank
62h	0010h (WS256N) 0008h (WS128N)	Bank 10 Region Information. X = Number of sectors in bank
63h	0010h (WS256N) 0008h (WS128N)	Bank 11 Region Information. X = Number of sectors in bank
64h	0010h (WS256N) 0008h (WS128N)	Bank 12 Region Information. X = Number of sectors in bank
65h	0010h (WS256N) 0008h (WS128N)	Bank 13 Region Information. X = Number of sectors in bank
66h	0010h (WS256N) 0008h (WS128N)	Bank 14 Region Information. X = Number of sectors in bank
67h	0013h (WS256N) 000Bh (WS128N)	Bank 15 Region Information. X = Number of sectors in bank

I8 Revisions

Revision F (October 29, 2004)

Data sheet completely revised. Changed arrangement of sections; edited explanatory text, added flowcharts. This document supersedes Revision E+1, issued August 9, 2004; only the changes specified for Revision F in this section affect the document or device. All other device specifications remain the same as presented in Revision E+1.

Deleted product selector guide.

16.8.2, Synchronous/Burst Read

Deleted t_{AAS} and t_{AAH} from table. Modified Note 1.

Table 17.4, System Interface String

Changed data at address 23h from 0003h to 0004h.

Revision G (January 27, 2005)

Global

Updated t_{IACC}, t_{BACC}, t_{OE}, CFI address 4Ah, and the Configuration Register.

Added Figure 13.2, PPB Program/Erase Algorithm.

Revision H (July 8, 2005)

All references to V_{IO} changed to V_{CC}

Removed all references to the S29WS064N

Changed CR14 to Reserved. It is pre-programmed at the factory

Updated: t_{ASP}, t_{PSP}, I_{CC1}, I_{CC3}, I_{CC4}, I_{CC6} specifications

Removed t_{AOE} and added f_{MIN} specifications

Updated address latency tables

Revision I (December 3, 2005)

Added *Page Mode Read* information

Revised t_{RACC} from 9 to 8.5 ns and t_{AVD} from 8 to 7 ns

Incorporated specification bulletin's changes, clarifications and errata

Added the 128 Mbit 80 MHz OPN and revised from Advanced to Preliminary Specifications

Mobile SDRAM Type I

128 Mbit (8M x 16Bit) SDRAM

PRELIMINARY

Features

- **Temperature compensated self refresh (TCSR)**
- **Fully synchronous; all signals registered on positive edge of system clock**
- **Internal pipelined operation; column address can be changed every clock cycle**
- **Internal banks for hiding row access/ precharge**
- **Programmable burst lengths: 1, 2, 4, 8, or full page**
- **Auto Precharge, includes concurrent auto precharge, and Auto Refresh Modes**
- **Self Refresh Mode; standard and low power**
- **64ms, 4,096-cycle refresh**
- **LVTTL-compatible inputs and outputs**
- **Low voltage power supply**
- **Partial array self refresh power-saving mode**
- **Deep power-down mode**
- **Programmable output drive strength**
- **Operating temperature range:**
  - Extended (-25°C to +85°C)
- **V_{DD}/V_{DDQ}: 1.8 V/1.8 V**

Speed Options

Speed Grade	Clock Frequency	Access Time		Setup Time	Hold Time
		CL = 2	CL = 3		
-75	133 MHz	—	5.4 ns	2.5 ns	1 ns
	100 MHz	6 ns	—	2.5 ns	1 ns
-10	104 MHz	—	7 ns	2.5 ns	1 ns
	83.3 MHz	8 ns	—	2.5 ns	1 ns

Address Table

	8M x 16
Configuration	2 M x 16 x 4 banks
Refresh Count	4K
Row Addressing	4K (A0–A11)
Bank Addressing	4 (BA0, BA1)
Column Addressing	512 (A1–A8)

I9 General Description

The 128Mb SDRAM is a high-speed CMOS, dynamic random-access memory containing 134,217,728 bits. It is internally configured as a quad bank DRAM with a synchronous interface (all signals are registered on the positive edge of the clock signal, CLK). Each of the x16s 32,554,432-bit banks is organized as 4,096 rows by 512 columns by 16 bits.

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, BA1 select the bank; A0–A11select the row). The address bits registered coincident with the Read or Write command are used to select the starting column location for the burst access.

The SDRAM provides for programmable read or write burst lengths of 1, 2, 4, or 8 locations, or the full page, with a burst terminate option. An auto precharge function may be enabled to provide a self-timed row precharge that is initiated at the end of the burst sequence.

The 128Mb SDRAM uses an internal pipelined architecture to achieve high-speed operation. This architecture is compatible with the 2n rule of prefetch architectures, but it also allows the column address to be changed on every clock cycle to achieve a high speed, fully random access. Precharging one bank while accessing one of the other three banks will hide the precharge cycles and provide seamless high-speed, random-access operation.

The 128Mb SDRAM is designed to operate in 1.8V, low-power memory systems. An auto refresh mode is provided, along with a power-saving, Deep Power-Down Mode. All inputs and outputs are LVTTL-compatible.

SDRAMs offer substantial advances in DRAM operating performance, including the ability to synchronously burst data at a high data rate with automatic column-address generation, the ability to interleave between internal banks in order to hide precharge time and the capability to randomly change column addresses on each clock cycle during a burst access.

20 Functional Block Diagram

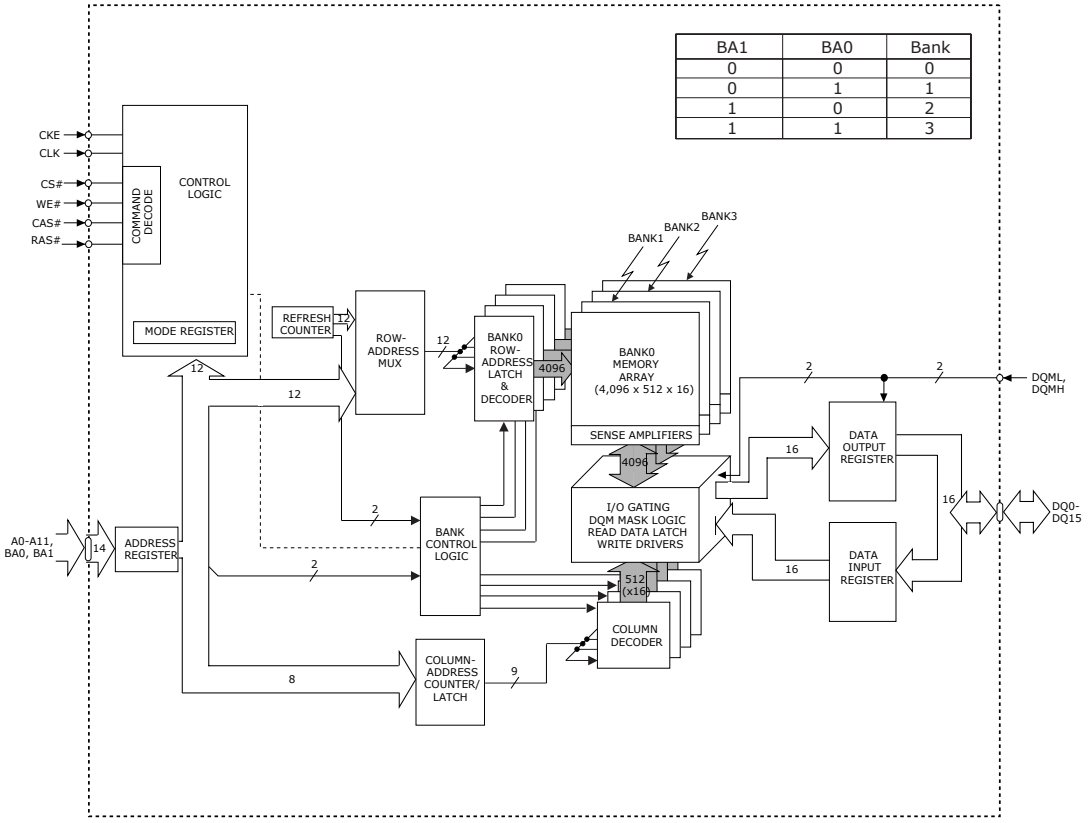


Figure 20.1 Block Diagram

21 Pin Descriptions

Symbol	Type	Description
CLK	Input	Clock: CLK is driven by the system clock. All SDRAM input signals are sampled on the positive edge of CLK. CLK also increments the internal burst counter and controls the output registers.
CKE	Input	Clock Enable: CKE activates (High) and deactivates (Low) the CLK signal. Deactivating the clock provides Precharge Power-Down and Self Refresh operation (all banks idle), Active Power-Down (row active in any bank), Deep Power Down (all banks idle), or Clock Suspend operation (burst/ access in progress). CKE is synchronous except after the device enters power-down and self refresh modes, where CKE becomes asynchronous until after exiting the same mode. The input buffers, including CLK, are disabled during power-down and self refresh modes, providing low standby power. CKE may be tied High.
CS#	Input	Chip Select: CS# enables (registered Low) and disables (registered High) the command decoder. All commands are masked when CS# is registered High. CS# provides for external bank selection on systems with multiple banks. CS# is considered part of the command code.
CAS#, RAS#, WE#	Input	Command Inputs: CAS#, RAS#, and WE# (along with CS#) define the command being entered.
LDQM, UDQM	Input	Input/Output Mask: DQM is sampled High and is an input mask signal for write accesses and an output enable signal for read accesses. Input data is masked during a Write cycle. The output buffers are placed in a High-Z state (two-clock latency) when during a Read cycle. LDQM corresponds to DQ0–DQ7, UDQM corresponds to DQ8–DQ15. LDQM and UDQM are considered same state when referenced as DQM.
BA0, BA1	Input	Bank Address Input(s): BA0 and BA1 define to which bank the Active, Read, Write or Precharge command is being applied. These pins also select between the mode register and the extended mode register.
A0-A11	Input	Address Inputs: A0–A11 are sampled during the Active command (row address A0–A11) and Read/Write command (column-address 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.
DQ0-DQ15	I/O	Data Input/Output: Data bus
NC	—	Internally Not Connected: These could be left unconnected, but it is recommended they be connected or VSS. G1 is a no connect for this part but may be used as A12 in future designs.
V_{DDQ}	Supply	DQ Power: Provide isolated power to DQs for improved noise immunity.
V_{SSQ}	Supply	DQ Ground: Provide isolated ground to DQs for improved noise immunity.
V_{DD}	Supply	Core Power Supply.
V_{SS}	Supply	Ground.

22 Functional Description

In general, the 128Mb SDRAMs (2 M x 16 x 4 banks) are quad-bank DRAMs that operate at 1.8V and include a synchronous interface (all signals are registered on the positive edge of the clock signal, CLK). Each bank (x16 32,554,432-bit) is organized as 4,096 rows by 512 columns by 16 bits.

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–A11 select the row). The address bits (A1–A8) 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.

23 Initialization

SDRAMs must be powered up and initialized in a predefined manner. Operational procedures other than those specified may result in undefined operation. Power should be applied to V_{DD} and V_{DDQ} simultaneously. This time delay should not exceed 2ms. Once the power is applied to V_{DD} and V_{DDQ}, and the clock is stable (stable clock is defined as a signal cycling within timing constraints specified for the clock pin), the SDRAM requires a 100µs delay prior to issuing any command other than a Command Inhibit or NOP. Starting at some point during this 100µs period and continuing at least through the end of this period, Command Inhibit or NOP commands should be applied.

Once the 100µs delay has been satisfied with at least one Command Inhibit or NOP command having been applied, a Precharge command should be applied. All banks must then be precharged, thereby placing the device in the all banks idle state.

Once in the idle state, two Auto refresh cycles must be performed. After the Auto refresh cycles are complete, the SDRAM is ready for mode register programming. Because the mode register will power up in an unknown state, it should be loaded prior to applying any operational command.

24 Mode Register Definition

In order to achieve low power consumption, there are two mode registers in the Mobile component, Mode Register and Extended Mode Register. The mode register is illustrated in Figure 25.1, Mode Register Definition, on page 111 (the extended mode register is illustrated in Figure 33.1).

The mode register defines the specific mode of operation of the SDRAM, including burst length, burst type, CAS latency, operating mode and write burst mode. 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 should be set to zero. M12 and M13 should be set to zero to prevent extended mode register.

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.

25 Burst Length

Read and write accesses to the SDRAM are burst oriented, with the burst length being programmable, as shown in Figure 33.1, "Mode Register Definition" section on page 111. 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, meaning that the burst will wrap within the block if a boundary is reached. The block is uniquely selected by A1–A8 when the burst length is set to two; by A2–A8 when the burst length is set to four; and by A3–A8 when the burst length is set to eight.

Table 25.1 Burst Definition

Burst Length	Starting Column Address			Order of Accesses Within A Burst	
				Type = Sequential	Type = Interleaved
2			A0		
			0	0-1	0-1
			1	1-0	1-0
4		A1	A0		
		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
8	A2	A1	A0		
	0	0	0	0-1-2-3-4-5-6-7	0-1-2-3-4-5-6-7
	0	0	1	1-2-3-4-5-6-7-0	1-0-3-2-5-4-7-6
	0	1	0	2-3-4-5-6-7-0-1	2-3-0-1-6-7-4-5
	0	1	1	3-4-5-6-7-0-1-2	3-2-1-0-7-6-5-4
	1	0	0	4-5-6-7-0-1-2-3	4-5-6-7-0-1-2-3
	1	0	1	5-6-7-0-1-2-3-4	5-4-7-6-1-0-3-2
	1	1	0	6-7-0-1-2-3-4-5	6-7-4-5-2-3-0-1
	1	1	1	7-0-1-2-3-4-5-6	7-6-5-4-3-2-1-0
Full Page (y)	n = A0–A8 (location 0-y)			Cn, Cn+1, Cn+2, Cn+3, Cn+4..., …Cn-1, Cn…	Not Supported

Notes:

1. *For full-page accesses: y = 512.*
2. *For a burst length of two, A1–A8 select the block-of-two burst; A0 selects the starting column within the block.*
3. *For a burst length of four, A2–A8 select the block-of-four burst; A0–A1 select the starting column within the block.*
4. *For a burst length of eight, A3–A8 select the block-of-eight burst; A0–A2 select the starting column within the block.*
5. *For a full-page burst, the full row is selected and A0–A8 select the starting column.*
6. *Whenever a boundary of the block is reached within a given sequence above, the following access wraps within the block.*
7. *For a burst length of one, A0–A8 select the unique column to be accessed, and mode register bit M3 is ignored.*

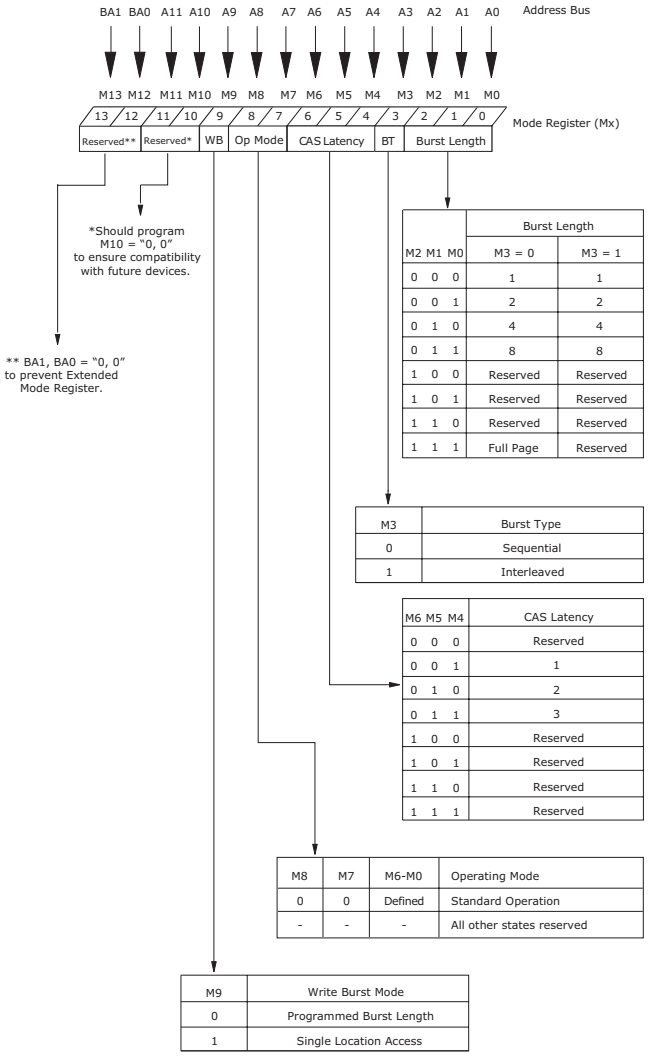


Figure 25.1 Mode Register Definition

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.

26 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 Table 25.1.

27 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 one, 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 T0 and the latency is programmed to two clocks, the DQs will start driving after T1 and the data will be valid by T2, as shown in Figure 27.1, CAS Latency. Table 27.1: CAS Latency, 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.

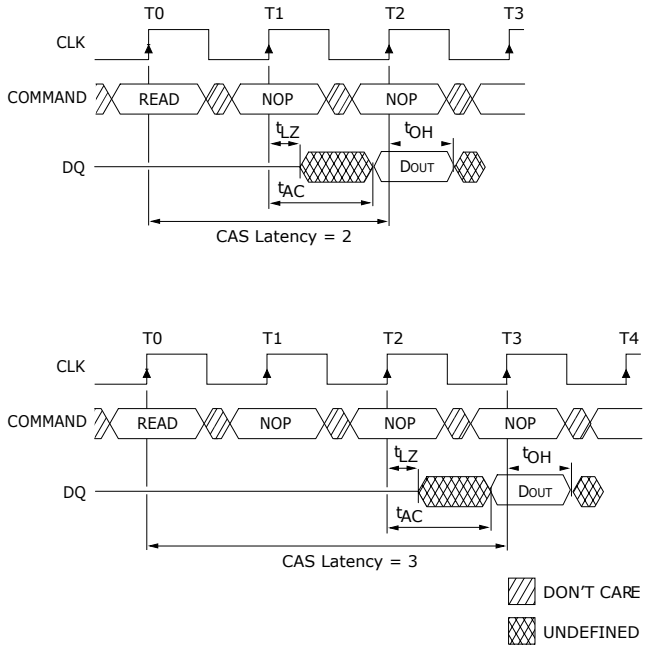


Figure 27.1 CAS Latency

Table 27.1 CAS Latency

Speed	Allowable Operating Frequency MHz	
	CAS Latency = 2	CAS Latency = 3
-75	≤ 100	≤ 133
-10	≤ 83.3	≤ 104

28 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.

29 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.

30 Extended Mode Register

The Extended Mode Register controls the functions beyond those controlled by the Mode Register. These additional functions are special features of the Mobile device. They include Temperature Compensated Self Refresh (TCSR) Control, Partial Array Self Refresh (PASR), and Output Drive Strength. Not programming the Extended Mode Register upon initialization, will result in default settings for the Low Power features. The Extended Mode will default to the +85°C setting for TCSR, full drive strength, and full array refresh.

The Extended Mode Register is programmed via the Mode Register Set command (BA1 = 1, BA0 = 0) and retains the stored information until it is programmed again or the device loses power.

The Extended Mode Register must be programmed with E6 through E11 set to "0". It must be loaded when all banks are idle and no bursts are in progress, and the controller must wait the specified time before initiating any subsequent operation. Violating either of these requirements results in unspecified operation.

Once the values are entered the Extended Mode Register settings will be retained even after exiting Deep Power-Down.

31 Temperature Compensated Self Refresh

Temperature Compensated Self Refresh (TCSR) allows the controller to program the Refresh interval during Self Refresh mode, according to the case temperature of the Mobile device. This allows great power savings during Self Refresh during most operating temperature ranges. Only during extreme temperatures would the controller have to select the maximum TCSR level. This would guarantee data during Self Refresh.

Every cell in the SDRAM requires refreshing due to the capacitor losing its charge over time. The refresh rate is dependent on temperature. At higher temperatures a capacitor loses charge quicker than at lower temperatures, requiring the cells to be refreshed more often. Historically, during Self Refresh, the refresh rate has been set to accommodate the worst case, or highest temperature range expected.

Thus, during ambient temperatures, the power consumed during refresh was unnecessarily high, because the refresh rate was set to accommodate the higher temperatures. Adjusting the refresh rate by setting E4 and E3 allows the SDRAM to accommodate more specific temperature regions during Self Refresh. There are four temperature settings, which will vary the Self Refresh current according to the selected temperature. This selectable refresh rate will save power when the SDRAM is operating at normal temperatures.

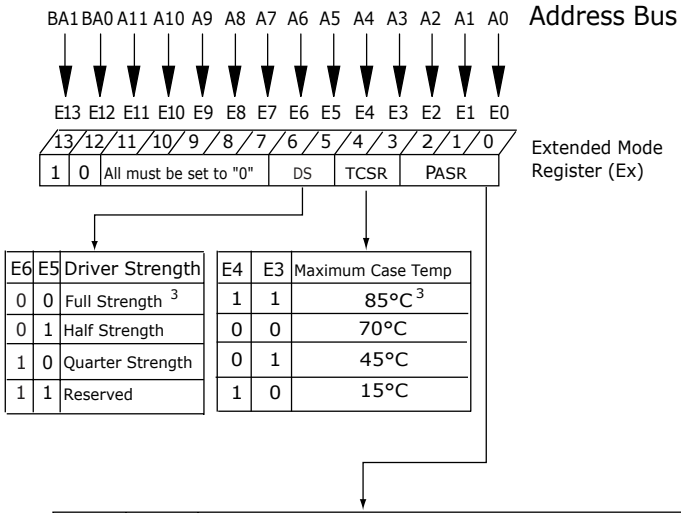
32 Partial Array Self Refresh

For further power savings during Self Refresh, the Partial Array Self Refresh (PASR) feature allows the controller to select the amount of memory that will be refreshed during Self Refresh. The refresh options are all banks (banks 0, 1, 2,

and 3); two banks (banks 0 and 1); and one bank (bank 0). Also included in the refresh options are the 1/2 bank and 1/4 bank partial array self refresh (bank 0). Write and Read commands occur to any bank selected during standard operation, but only the selected banks in PASR will be refreshed during Self Refresh. It is important to note that data in unused banks, or portions of banks, will be lost when PASR is used. Data will be lost in banks 1, 2, and 3 when the one bank option is used.

33 Driver Strength

Bits E5 and E6 of the extended mode register can be used to select the driver strength of the DQ outputs. This value should be set according to the application requirements. Full drive strength was carried over from standard SDRAM and is suitable to drive higher load systems. Full drive strength is not recommended for loads under 30pF. Half drive strength is intended for multi-drop systems with various loads. This drive option is not recommended for loads under 15pF. Quarter drive strength is intended for lighter loads or point-to-point systems.



E2	E1	E0	Self Refresh Coverage
0	0	0	Four Banks [3]
0	0	1	Two Banks (Bank 0,1)
0	1	0	One Bank (Bank 0)
0	1	1	RFU
1	0	0	RFU
1	0	1	1/2 Bank (Bank 0) [4]
1	1	0	1/4 Bank (Bank 0) [5]
1	1	1	RFU

Figure 33.1 Extended Mode Register

Notes:

1. *E13 and E12 (BA1 and BA0) must be "1, 0" to select the extended mode register (vs. the base mode register).*
2. *RFU: Reserved for future use.*
3. *Default EMR values are full array for PASR, Full Drive Strength, and 85° for TCSR.*
4. *E11 = 0.*
5. *E10, E11 = 0.*

34 Commands

Table 34.1 provides a quick reference of available commands. This is followed by a written description of each command. Three additional Truth Tables appear following the Operation section; these tables provide current state/next state information.

Table 34.1 Truth Table I - Commands and DQM Operation

Name (Function)								
Command Inhibit (NOP)	H	X	X	X	X	X	X	
No Operation	L	H	H	H	X	X	X	
Active (Select bank and activate row)	L	L	H	H	X	Bank/ Row	X	3
Read (Select bank and column, and start Read burst)	L	H	L	H	L/H	Bank/Col	X	4
Write (Select bank and column, and start Write burst)	L	H	L	L	L/H	Bank/Col	Valid	4
Burst Terminate or Deep Power Down (Enter deep power down mode)	L	H	H	L	X	X	X	9, 10
Precharge (Deactivate row in bank or banks)	L	L	H	L	X	Bank, A10	X	5
Auto Refresh or Self Refresh (Enter self refresh mode)	L	L	L	H	X	X	X	6, 7
Load Mode Register/Load Extended Mode Register	L	L	L	L	X	Op-Code	X	2
Write Enable/Output Enable	X	X	X	X	L	X	Active	8
Write Inhibit/Output High-Z	X	X	X	X	H	X	High-Z	8

Notes:

1. *CKE is High for all commands shown except Self Refresh and Deep Power Down.*
2. *A0-A11 define op-code written to mode register.*
3. *A0–A11 provide row address, and BA0, BA1 determine which bank is made active.*
4. *A1–A8 provide column address; A10 High enables the auto precharge feature (non persistent), while A10 Low disables the auto precharge feature; 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. *This command is Auto Refresh if CKE is High, Self Refresh if CKE is Low.*
7. *Internal refresh counter controls row addressing; all inputs and I/Os are "Don't Care" except for CKE*
8. *Activates or deactivates the DQs during Writes (zero-clock delay) and Reads (two-clock delay). DQML controls DQ0-7, DQMH controls DQ8-15.*
9. *This command is Burst Terminate when CKE is high and Deep Power Down when CKE is low.*
10. *The purpose of the Burst Terminate command is to stop a data burst, thus the command could coincide with data on the bus. However the DQs column reads a don't care state to illustrate that the Burst Terminate command can occur when there is no data present.*

35 Command Inhibit

The Command Inhibit function prevents new commands from being executed by the SDRAM, regardless of whether the CLK signal is enabled. The SDRAM is effectively deselected. Operations already in progress are not affected.

36 No Operation (NOP)

The No Operation (NOP) command is used to perform a NOP to an SDRAM which is selected (CS# is Low). This prevents unwanted commands from being registered during idle or wait states. Operations already in progress are not affected.

37 Load Mode Register

The mode register is loaded via inputs A0–A11, BA0, BA1. See mode register heading in the Register Definition section. The Load Mode Register and Load Extended Mode Register commands can only be issued when all banks are idle, and a subsequent executable command cannot be issued until t_{MRD} is met.

The values of the mode register and extended mode register will be retained even when exiting deep power-down.

38 Active

The Active command is used to open (or activate) a row in a particular bank for a subsequent access. The value on the BA0, BA1 inputs selects the bank, and the address provided on inputs A0–A11 selects the row. This row remains active (or open) for accesses until a precharge command is issued to that bank. A precharge command must be issued before opening a different row in the same bank.

39 Read

The Read command is used to initiate a burst read access to an active row. The value on the BA0, BA1 inputs selects the bank, and the address provided on inputs A1–A8 selects the starting column location. The value on input A10 determines whether or not auto precharge is used. If auto precharge is selected, the row being accessed will be precharged at the end of the read burst; if auto precharge is not selected, the row will remain open for subsequent accesses. Read data appears on the DQ subject to the logic level on the DQM inputs two clocks earlier. If a given DQM signal was registered High, the corresponding DQ will be High-Z two clocks later; if the DQM signal was registered Low, the DQ will provide valid data.

40 Write

The Write command is used to initiate a burst write access to an active row. The value on the BA0, BA1 inputs selects the bank, and the address provided on inputs A1–A8 selects the starting column location. The value on input A10 determines whether or not auto precharge is used. If auto precharge is selected, the row being accessed will be precharged at the end of the write burst; if auto precharge is not selected, the row will remain open for subsequent accesses. Input data appearing on the DQ is written to the memory array subject to the DQM input logic level appearing coincident with the data. If a given DQM signal is registered Low, the corresponding data will be written to memory; if the DQM signal is registered High, the corresponding data inputs will be ignored, and a write will not be executed to that byte/column location.

41 Precharge

The Precharge command is used to deactivate the open row in a particular bank or the open row in all banks. The bank(s) will be available for a subsequent row access a specified time (t_{RP}) after the precharge command is issued. Input A10 determines whether one or all banks are to be precharged, and in the case where only one bank is to be precharged, inputs BA0, BA1 select the bank. Otherwise BA0, BA1 are treated as "Don't Care." 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.

42 Auto Precharge

Auto precharge is a feature which performs the same individual-bank precharge function described above, without requiring an explicit command. This is accomplished by using A10 to enable auto precharge in conjunction with a specific Read or Write command. A precharge of the bank/row that is addressed with the Read or Write command is automatically performed upon completion of the Read or Write burst, except in the full-page burst mode, where auto precharge does not apply. Auto precharge is non persistent in that it is either enabled or disabled for each individual Read or Write command.

Auto precharge ensures that the precharge is initiated at the earliest valid stage within a burst. The user must not issue another command to the same bank until the precharge time (t_{RP}) is completed. This is determined as if an explicit Precharge command was issued at the earliest possible time, as described for each burst type in the Operation section of this data sheet.

43 Burst Terminate

The Burst Terminate command is used to truncate either fixed-length or full-page bursts. The most recently registered Read or Write command prior to the Burst Terminate command will be truncated, as shown in the Operation section of this data sheet.

44 Auto Refresh

Auto Refresh is used during normal operation of the SDRAM and is analogous to CAS#-BEFORE-RAS# (CBR) refresh in conventional DRAMs. This command is non persistent, so it must be issued each time a refresh is required. All active banks must be PRECHARGED prior to issuing an Auto Refresh command. The Auto Refresh command should not be issued until the minimum t_{RP} has been met after the Precharge command as shown in the operation section.

The addressing is generated by the internal refresh controller. This makes the address bits "Don't Care" during an Auto Refresh command. The 128Mb SDRAM requires 4,096 Auto Refresh cycles every 64ms (t_{REF}). Providing a distributed Auto Refresh command every 15.625µs will meet the refresh requirement and ensure that each row is refreshed. Alternatively, 4,096 Auto Refresh commands can be issued in a burst at the minimum cycle rate (t_{RFC}), once every 64ms.

45 Self Refresh

The Self Refresh command can be used to retain data in the SDRAM, even if the rest of the system is powered down, as long as power is not completely removed from the SDRAM. When in the self refresh mode, the SDRAM retains data without external clocking. The Self Refresh command is initiated like an Auto Refresh command except CKE is disabled (Low). Once the Self Refresh command is registered, all the inputs to the SDRAM become "Don't Care" with the exception of CKE, which must remain Low.

Once self refresh mode is engaged, the SDRAM provides its own internal clocking, causing it to perform its own auto refresh cycles. The SDRAM must remain in self

refresh mode for a minimum period equal to t_{RAS} and may remain in self refresh mode for an indefinite period beyond that.

The procedure for exiting self refresh requires a sequence of commands. First, CLK must be stable (stable clock is defined as a signal cycling within timing constraints specified for the clock pin) prior to CKE going back High. Once CKE is High, the SDRAM must have NOP commands issued (a minimum of two clocks) for t_{XSR} because time is required for the completion of any internal refresh in progress.

Upon exiting the self refresh mode, Auto Refresh commands must be issued every 15.625µs or less as both Self Refresh and Auto Refresh utilize the row refresh counter.

46 Deep Power-down

The operating mode deep power-down achieves maximum power reduction by eliminating the power of the whole memory array of the device. Array data will not be retained once the device enters deep power-down mode.

This mode is entered by having all banks idle then CS# and WE# held low with RAS# and CAS# held high at the rising edge of the clock, while CKE is low. This mode is exited by asserting CKE high.

47 Operation

47.1 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 Figure 47.1, Activating a Specific Row in a Specific Bank Register).

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. $t_{RCD\ (MIN)}$ 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 Figure 47.2, 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}.

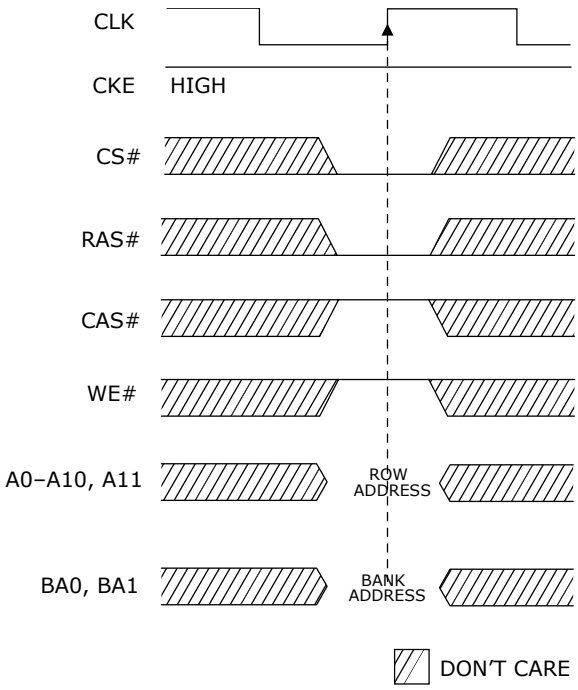


Figure 47.1 Activating a Specific Row in a Specific Bank Register

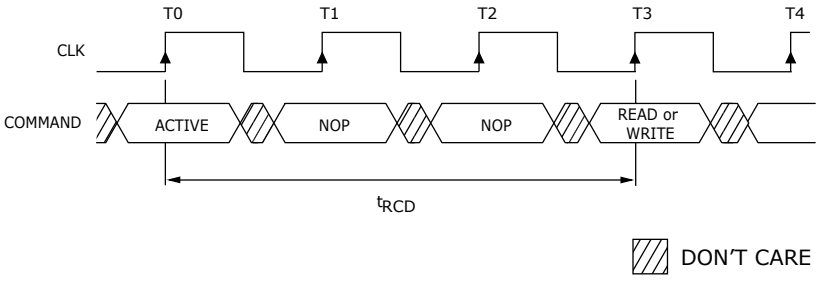


Figure 47.2. Meeting $t_{RCD\ (MIN)}$ when $2 < t_{RCD\ (MIN)}/t_{CK} \leq 3$

47.2 Reads

Read bursts are initiated with a Read command, as

shown in Figure 47.2.

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. Figure 27.1, "CAS Latency," on page 112, shows general timing for each possible CAS latency setting.

Upon completion of a burst, assuming no other commands have been initiated, the DQ 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 that 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 Figure 47.2, "Consecutive Read Bursts," on page 122, 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 128Mb 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 Figure 47.3, "Random Read Accesses," on page 122, 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 I/O 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 DQ go High-Z. In this case, at least a single cycle delay should occur between the last read data and the Write command.

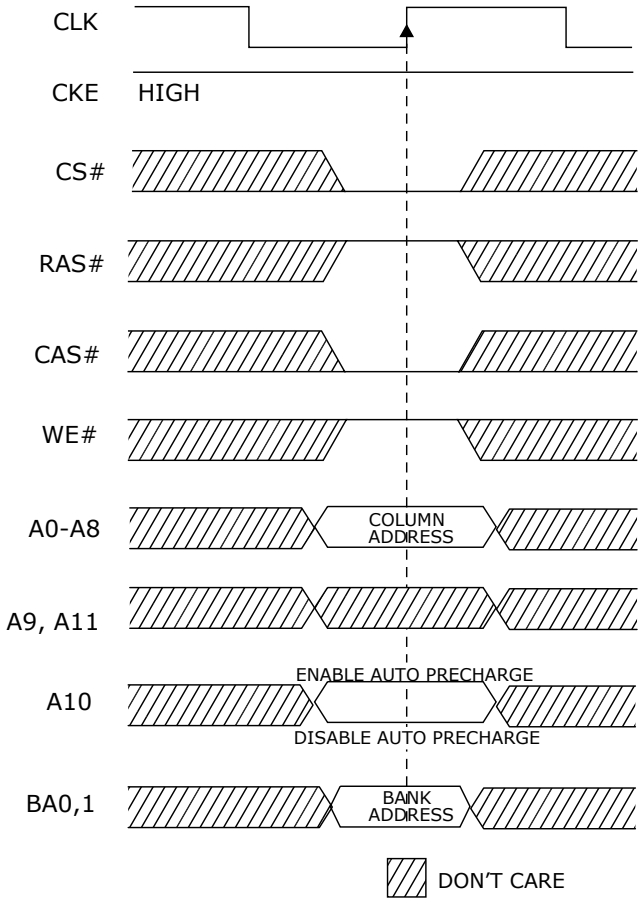


Figure 47.1 Read Command

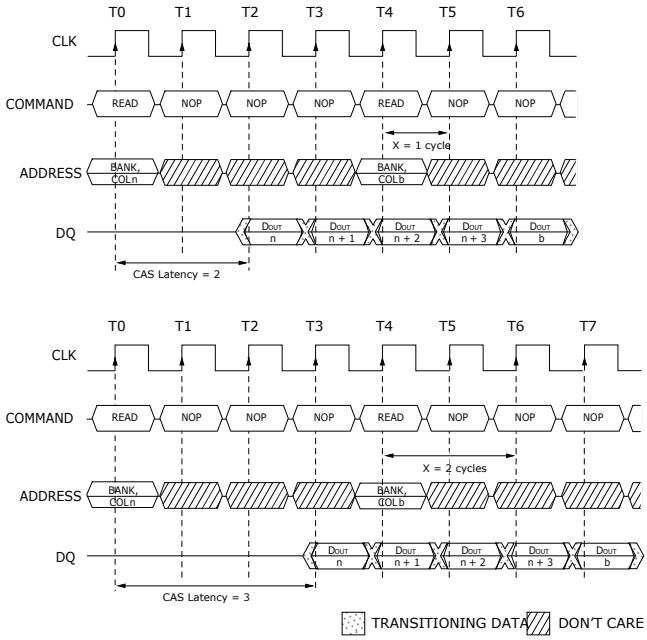


Figure 47.2. Consecutive Read Bursts

Note: *Each Read command may be to any bank. DQM is Low.*

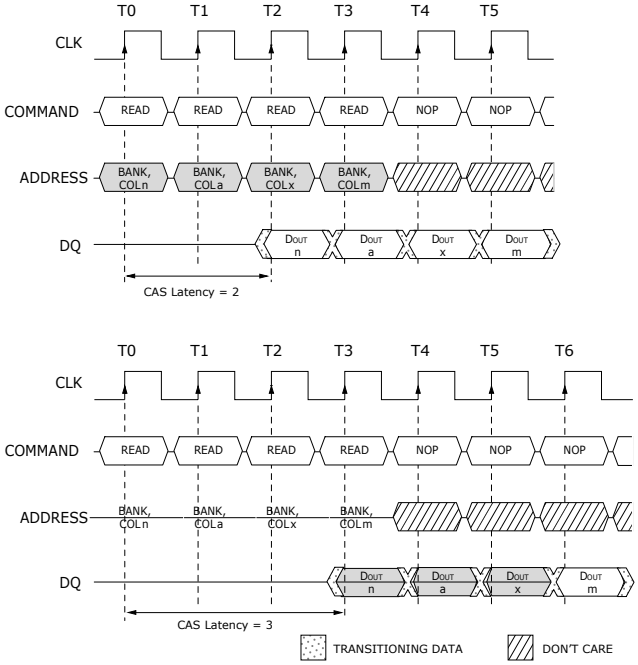


Figure 47.3. Random Read Accesses

Note: *Each Read command may be to any bank. DQM is low.*

The DQM input is used to avoid I/O contention, as shown in Figure 47.4 and Figure 47.5. 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 DQ 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 47.6, 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 47.5 shows the case where the clock frequency allows for bus contention to be avoided without adding a NOP cycle, and 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 Figure 47.6 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 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.

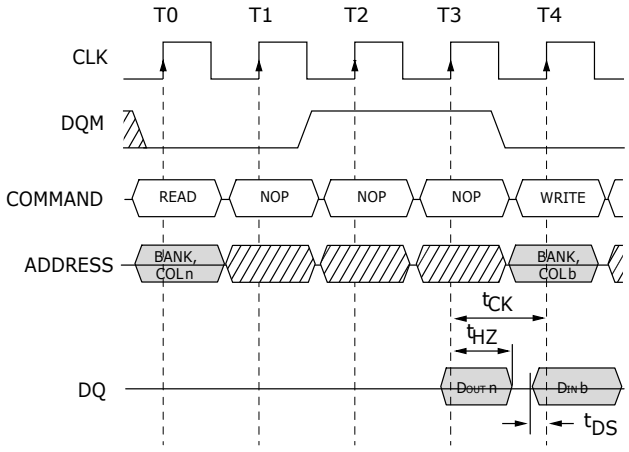


Figure 47.4. Read to Write

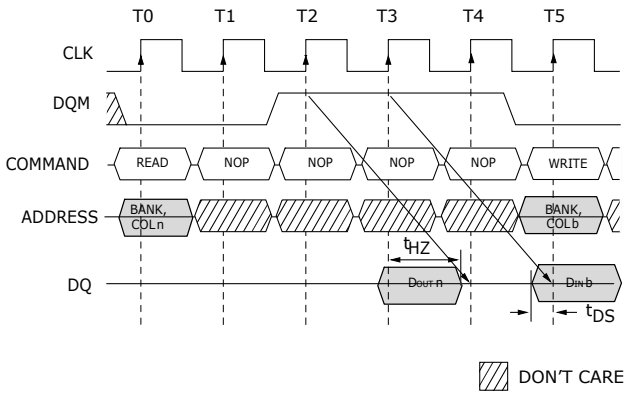


Figure 47.5. Read to Write with Extra Clock Cycle

Note: *A CAS latency of three is used for illustration. The Read command may be to any bank, and the Write command may be to any bank.*

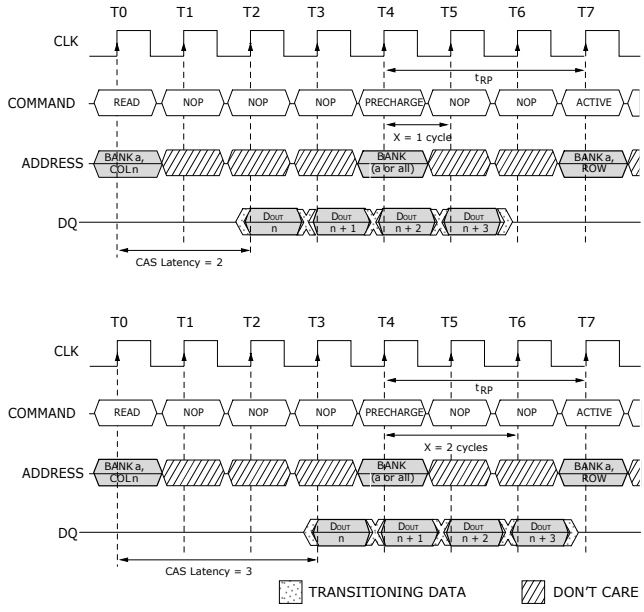


Figure 47.6. Read to Precharge

Note: *DQM is low.*

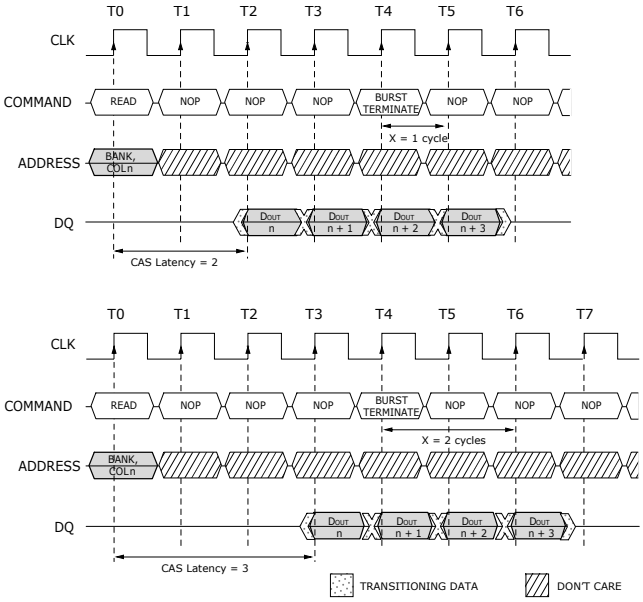


Figure 47.7. Terminating a Read Burst

Note: *DQM is low.*

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 Figure 47.7 for each possible CAS latency; data element *n + 3* is the last desired data element of a longer burst.

47.3 Writes

Write bursts are initiated with a Write command, as shown in Figure 47.1.

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 DQ will remain High-Z and any additional input data will be ignored (see Figure 47.3). A full-page burst will continue until terminated. (At the end of the page, it will wrap to column 0 and continue.)

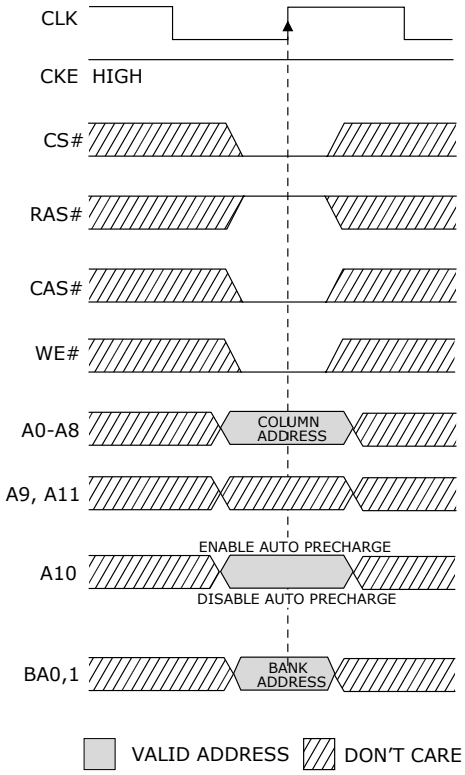


Figure 47.1 Write Command

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 Figure 47.4. Data *n + 1* is either the last of a burst of two or the last desired of a longer burst. The 128Mb 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 Figure 47.4, or each subsequent Write may be performed to a different bank.

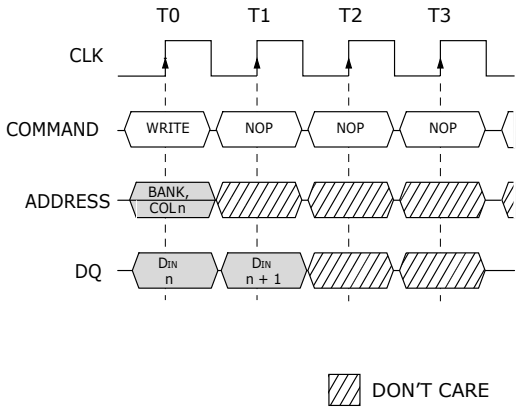


Figure 47.2. Write Burst

Note: *Burst length = 2. DQM is low.*

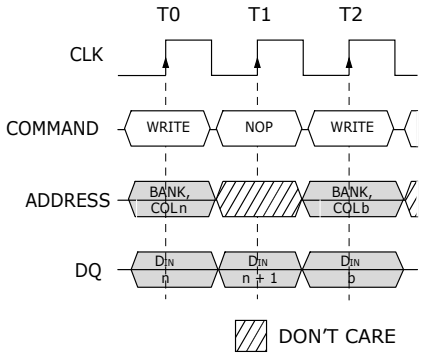


Figure 47.3. Write to Write

Note: *DQM is low. Each Write command may be to any 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 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 Figure 47.5. 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 tWR 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 Figure 47.6. 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.

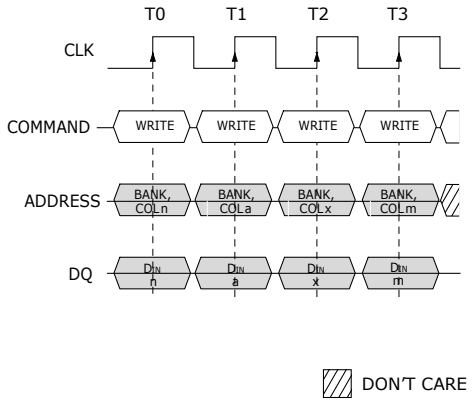


Figure 47.4. Random Write Cycles

Note: *Each Write command may be to any bank. DQM is low.*

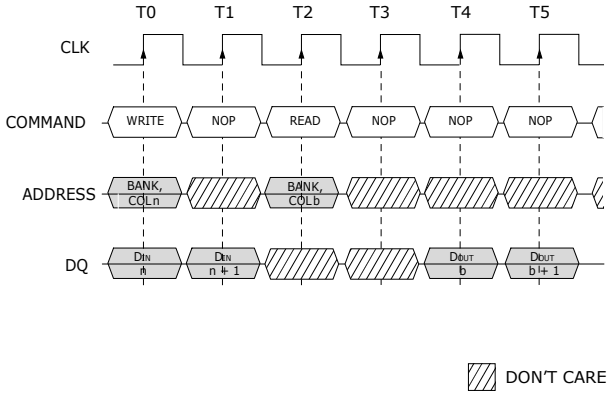


Figure 47.5. Write to Read

Note: *The Write command may be to any bank, and the Read command may be to any bank. DQM is low. CAS latency = 2 for illustration.*

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 pre-charge. 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 Figure 47.2, where data *n* is the last desired data element of a longer burst.

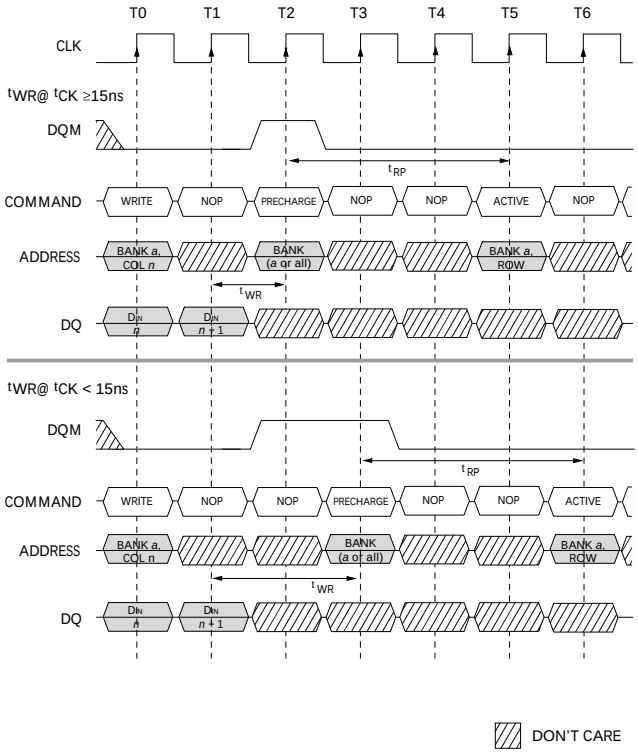


Figure 47.6. Write to Precharge

Note: *DQM could remain low in this example if the Write burst is a fixed length of two.*

47.4 Precharge

The Precharge command (see Figure 47.3) is used to deactivate the open row in a particular bank or the open row in all banks. The bank(s) will be available for a subsequent row access some specified time (t_{RP}) after the precharge command is issued. Input A10 determines whether one or all banks are to be precharged, and in the case where only one bank is to be precharged, inputs BA0, BA1 select the bank. When all banks are to be precharged, inputs BA0, BA1 are treated as "Don't Care." 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.

47.5 Power-down

Power-down occurs if CKE is registered low coincident with a NOP or Command Inhibit when no accesses are in progress. If power-down occurs when all banks are idle, this mode is referred to as precharge power-down; if power-down occurs when there is a row active in any bank, this mode is referred to as active power-down. Entering power-down deactivates the input and output buffers, excluding CKE, for maximum power savings while in standby. The device may not remain in the power-down state longer than the refresh period (64ms) since no refresh operations are performed in this mode. The power-down state is exited by registering a NOP or Command Inhibit and CKE High at the desired clock edge (meeting t_{CKS}). See Figure 47.1.

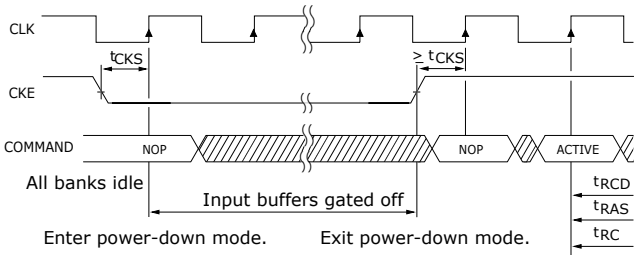


Figure 47.1. Power-Down

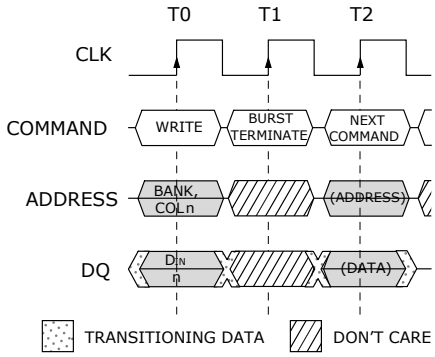


Figure 47.2. Terminating a Write Burst

Note: *DQMs are low.*

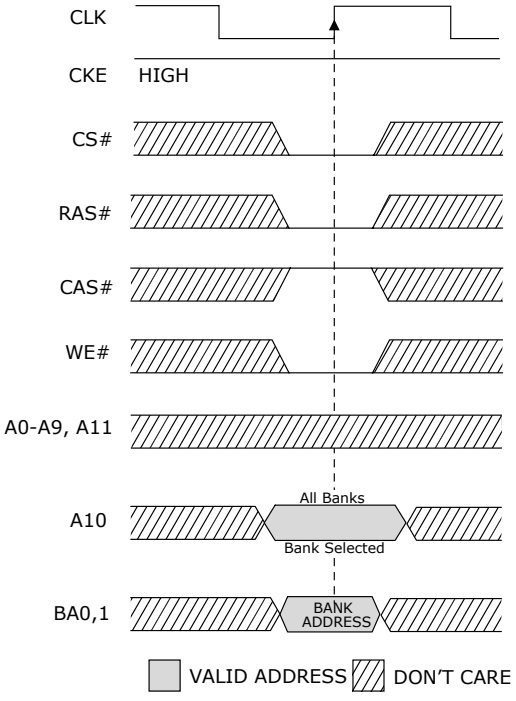


Figure 47.3. Precharge Command

47.6 Deep Power-down

Deep Power Down mode is a maximum power savings feature achieved by shutting off the power to the entire memory array of the device. Data on the memory array will not be retained once Deep Power Down mode is executed. Deep Power Down mode is entered by having all banks idle then CS# and WE# held low with RAS# and CAS# high at the rising edge of the clock, while CKE is low. CKE must be held low during deep power-down.

In order to exit Deep Power Down mode, CKE must be asserted high. After exiting, the following sequence is needed in order to enter a new command. Maintain NOP input conditions for a minimum of 100us. Issue Precharge commands for all banks. Issue eight or more AUTOREFRESH commands. The values of the mode register and extended mode register will be retained upon exiting deep power-down.

47.7 Clock Suspend

The clock suspend mode occurs when a column access/burst is in progress and CKE is registered low. In the clock suspend mode, the internal clock is deactivated, "freezing" the synchronous logic.

For each positive clock edge on which CKE is sampled Low, the next internal positive clock edge is suspended. Any command or data present on the input pins at the time of a suspended internal clock edge is ignored; any data present on the DQ pins remains driven; and burst counters are not incremented, as long as the clock is suspended. (See examples in Figure 47.4, and Figure 47.5.)

Clock suspend mode is exited by registering CKE High; the internal clock and related operation will resume on the subsequent positive clock edge.

47.8 Burst Read/Single Write

The burst read/single write mode is entered by programming the write burst mode bit (M9) in the mode register to a logic 1. In this mode, all Write commands result in the access of a single column location (burst of one), regardless of the programmed burst length. Read commands access columns according to the programmed burst length and sequence, just as in the normal mode of operation (M9 = 0).

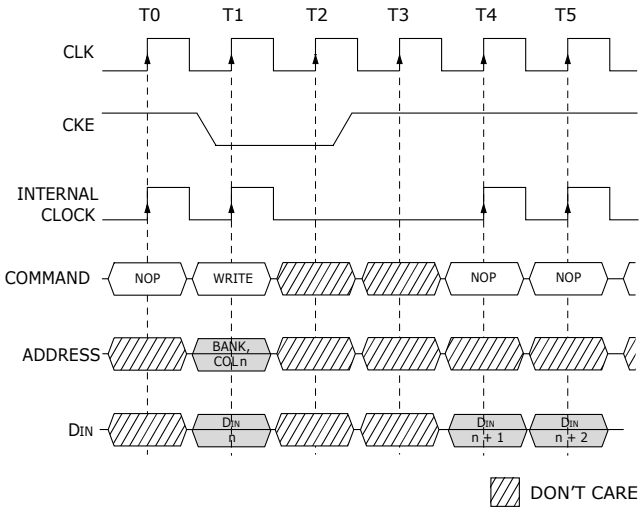


Figure 47.4. Clock Suspend During Write Burst

Note: *For this example, burst length = 4 or greater, and DM is low.*

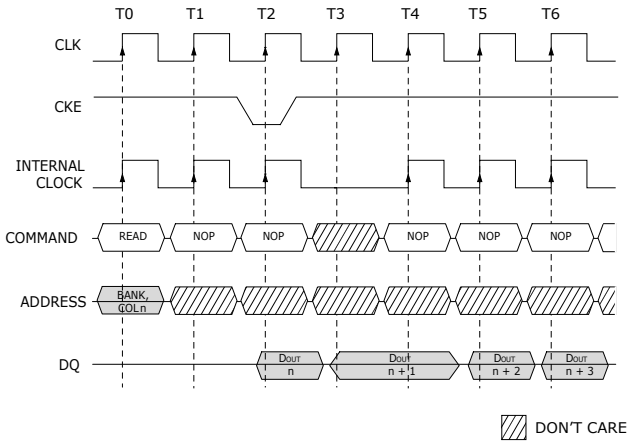


Figure 47.5. Clock Suspend During Read Burst

Note: *For this example, CAS latency = 2, burst length = 4 or greater, and DQM is low.*

47.9 Concurrent Auto Precharge

The SDRAM devices support Concurrent Auto precharge, which allows an access command (Read or Write) to another bank while an access command with auto precharge enabled is executing. Four cases where concurrent auto precharge occurs are defined below.

47.9.1 Read with Auto Precharge

1. Interrupted by a Read (with or without auto precharge): A Read to bank m will interrupt a Read on bank *n*, two or three clocks later, depending on CAS latency. The precharge to bank *n* will begin when the Read to bank *m* is registered (Figure 47.6).
2. Interrupted by a Write (with or without auto precharge): When a Write to bank *m* registers, a Read on bank *n* will be interrupted. DQM should be used two clocks prior to the Write command to prevent bus contention. The pre-

charge to bank *n* will begin when the Write to bank m is registered (Figure 47.7).

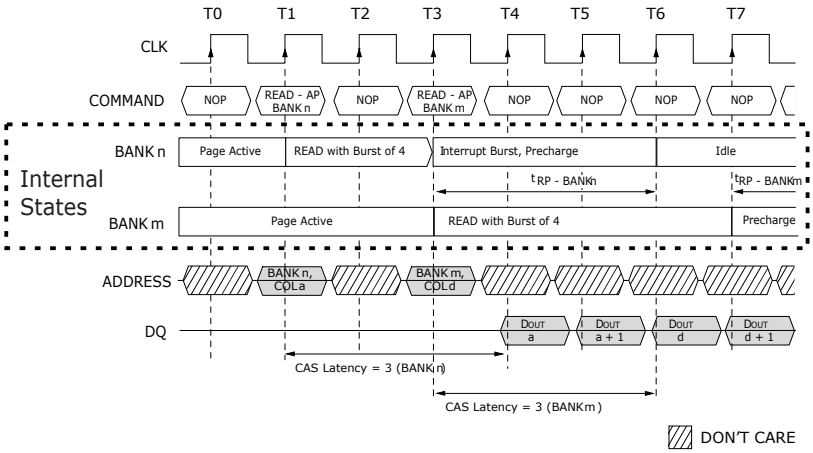


Figure 47.6. Read with Auto Precharge Interrupted by a Read

Note: *DQM is low.*

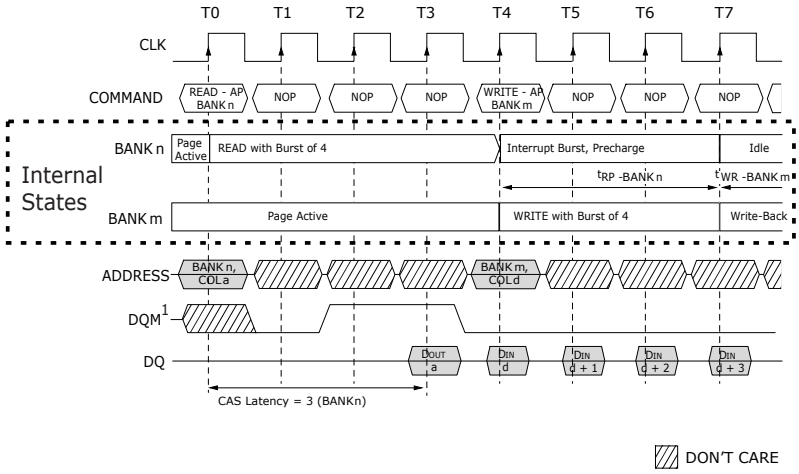


Figure 47.7. Read with Auto Precharge Interrupted by a Write

Note: *DQM is high at T2 to prevent D_{OUT}-a+1 from contending with D_{IN}-dat T4.*

47.10 Write with Auto Precharge

1. Interrupted by a Read (with or without auto precharge): When a Read to bank *m* registers, it will interrupt a Write on bank n, with the data-out appearing 2 or 3 clocks later, (depending on CAS latency). 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* (Figure 47.8).
2. Interrupted by a Write (with or without auto precharge): When a Write to bank *m* registers, it will interrupt a Write on bank *n*. 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 data Write to bank *n* will be data registered one clock prior to a Write to bank *m* (Figure 47.9).

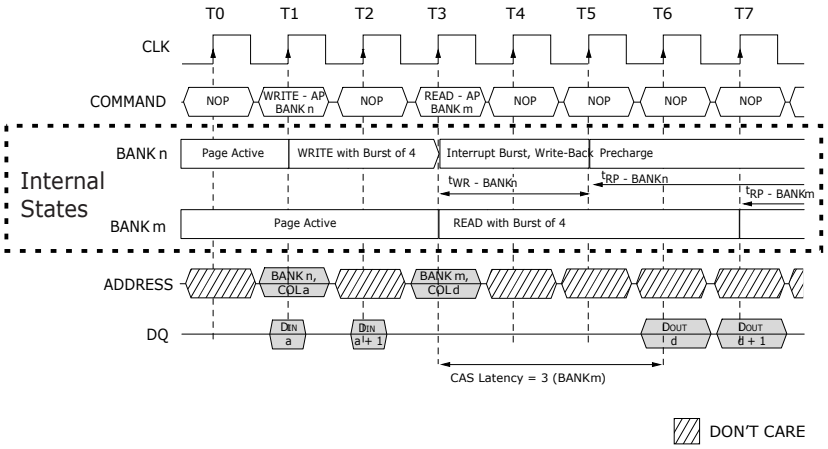


Figure 47.8. Write with Auto Precharge Interrupted by a Read

Note: *DQM is low.*

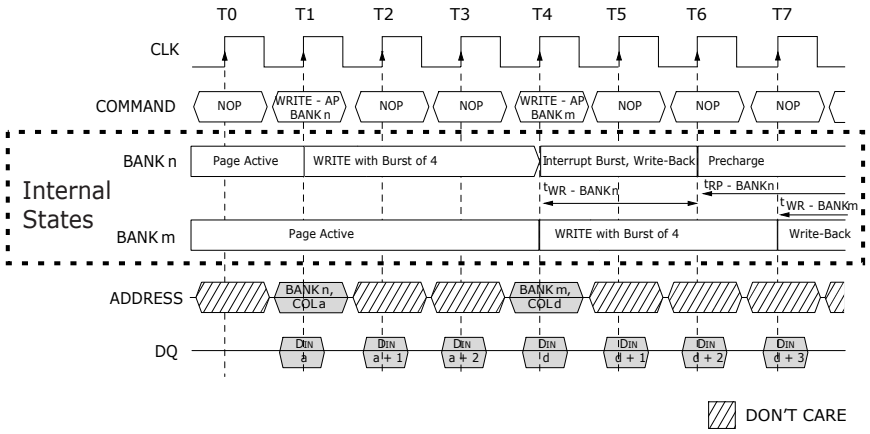


Figure 47.9. Write with Auto Precharge Interrupted by a Write

Note: *DQM is low.*

Table 47.I Truth Table 2 - CKE

CKE_{n-1}	CKE_n	Current State	$Command_n$	$Action_n$	Notes
L	L	Power-Down	X	Maintain Power-Down	
		Self Refresh	X	Maintain Self Refresh	
		Clock Suspend	X	Maintain Clock Suspend	
		Deep Power-Down	X	Maintain Deep Power-Down	8
L	H	Power-Down	Command Inhibit or NOP	Exit Power-Down	5
		Deep Power-Down	X	Exit Deep Power-Down	8
		Self Refresh	Command Inhibit or NOP	Exit Self Refresh	6
		Clock Suspend	X	Exit Clock Suspend	7
H	L	All Banks Idle	Command Inhibit or NOP	Power-Down Entry	
		All Banks Idle	Burst Terminate	Deep Power-Down Entry	8
		All Banks Idle	Auto Refresh	Self Refresh Entry	
		Reading or Writing	Valid	Clock Suspend Entry	
H	H		See Truth Table 3		

Notes:

1. *CKE_n is the logic state of CKE at clock edge n; CKE_{n-1} 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. *$Command_n$ is the command registered at clock edge n, and $Action_n$ is a result of $Command_n$.*
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_{CKS} is met).*
6. *Exiting self refresh at clock edge n will put the device in the all banks idle state once t_{XSR} is met. Command Inhibit or NOP commands should be issued on any clock edges occurring during the t_{XSR} period. A minimum of two NOP commands must be provided during t_{XSR} 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.*
8. *Deep power-down is a power-saving feature of this Mobile SDRAM device. This command is Burst Terminate when CKE is high and Deep Power Down when CKE is low.*

Table 47.2 Truth Table 3 - Current State Bank n, Command to Bank n

Current State	CS#	RAS #	CAS#	WE #	Command (Action)	Notes
Any	H	X	X	X	Command Inhibit (NOP/Continue previous operation)	
	L	H	H	H	No Operation (NOP/Continue previous operation)	
Idle	L	L	H	H	Active (Select and activate row)	
	L	L	L	H	Auto Refresh	7
	L	L	L	L	Load Mode Register	7
	L	L	H	L	Precharge	11
Row Active	L	H	L	H	Read (Select column and start Read burst)	10
	L	H	L	L	Write (Select column and start Write burst)	10
	L	L	H	L	Precharge (Deactivate row in bank or banks)	8
Read (Auto precharge disabled)	L	H	L	H	Read (Select column and start new Read burst)	10
	L	H	L	L	Write (Select column and start Write burst)	10
	L	L	H	L	Precharge (Truncate Read burst, start Precharge)	8
	L	H	H	L	Burst Terminate	9
Write (Auto precharge disabled)	L	H	L	H	Read (Select column and start Read burst)	10
	L	H	L	L	Write (Select column and start new Write burst)	10
	L	L	H	L	Precharge (Truncate Write burst, start Precharge)	8
	L	H	H	L	Burst Terminate	9

Notes:

1. *This table applies when CKE_{n-1} was High and CKE_n is High (see Table 47.1) and after t_{XSR} 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_{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.
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 Table 47.2, and according to Table 47.3.*
   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_{RCD} is met. Once t_{RCD} 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. *Deep Power-Down is power-saving feature of this Mobile SDRAM device. This command is Burst Terminate when CKE is high and Deep Power Down when CKE is low.*
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.*

Table 47.3 Truth Table 4 - Current State Bank n, Command to Bank m

Current State	CS#	RAS #	CAS#	WE #	Command (Action)	Notes
Any	H	X	X	X	Command Inhibit (NOP/Continue previous operation)	
	L	H	H	H	No Operation (NOP/Continue previous operation)	
Idle	X	X	X	X	Any Command Otherwise Allowed to Bank *m*	
Row Activating, Active, or Precharging	L	L	H	H	Active (Select and activate row)	7
	L	H	L	H	Read (Select column and start Read burst)	7
	L	H	L	L	Write (Select column and start Write burst)	
	L	L	H	L	Precharge	
Read (Auto precharge disabled)	L	L	H	H	Active (Select and activate row)	
	L	H	L	H	Read (Select column and start Read burst)	7, 10
	L	H	L	L	Write (Select column and start Write burst)	7, 11
	L	L	H	L	Precharge	9
Write (Auto precharge disabled)	L	L	H	H	Active (Select and activate row)	
	L	H	L	H	Read (Select column and start Read burst)	7, 12
	L	H	L	L	Write (Select column and start new Write burst)	7, 13
	L	L	H	L	Precharge	9
Read (with Auto precharge)	L	L	H	H	Active (Select and activate row)	
	L	H	L	H	Read (Select column and start new Read burst)	7, 8 14
	L	H	L	L	Write (Select column and start Write burst)	7, 8, 15
	L	L	H	L	Precharge	9
Write (with Auto precharge)	L	L	H	H	Active (Select and activate row)	
	L	H	L	H	Read (Select column and start Read burst)	7, 8, 16
	L	H	L	L	Write (Select column and start new Write burst)	7, 8, 17
	L	L	H	L	Precharge	9

Notes:

1. *This table applies when CKE_{n-1} was high and CKE_n is high (see Table 47.1) and after t_{XSR} has been met (if the previous state was self refresh).*
2. *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.*
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_{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.

4. *Auto Refresh, Self Refresh and Load Mode Register commands may only be issued when all banks are idle.*
5. *A Burst Terminate command cannot be issued to another bank; it applies to the bank represented by the current state only.*
6. *All states and sequences not shown are illegal or reserved.*
7. *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 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 (Figure 47.2).*
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 (Figure 47.4, and Figure 47.5). 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 (Figure 47.5), 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 (Figure 47.3). 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 (Figure 47.6). The Precharge to bank n will begin when the Read to bank m is registered.*
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 (Figure 47.7). 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.*
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 (Figure 47.8). 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 bank n will be data-in registered one clock prior to the Read to bank m.*
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 (Figure 47.9). The last valid Write to bank n will be data registered one clock to the Write to bank m.*

48 Absolute Maximum Ratings

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

Voltage on V_{DD}/V_{DDQ} Supply Relative to V_{SS} . . . . . -0.35V to +2.8V
Voltage on Inputs, NC or I/O Pins Relative to VSS . . . . . -0.35V to +2.8V
Operating Temperature T_A (Extended) . . . . . -25°C to +85°C
Storage Temperature (plastic) . . . . . -55°C to +150°C

Table 48.1 DC Electrical Characteristics and Operating Conditions

Parameter	Symbol	-75		-10		Units	Notes
		Min	Max	Min	Max		
Supply Voltage	V_{DD}	1.7	1.95	1.7	1.9	V	
I/O Supply Voltage	V_{DDQ}	1.7	1.95	1.7	1.9		
Input High Voltage: Logic 1; All inputs	V_{IH}	0.8 x V_{DDQ}	V_{DD} + 0.3	0.8 x V_{DDQ}	V_{DD} + 0.3		4
Input Low Voltage: Logic 0; All inputs	V_{IL}	-0.3	+0.3	-0.3	+0.3		4
Output High Voltage: All inputs	V_{OH}	0.9 x V_{DDQ}	—	0.9 x V_{DDQ}	—		
Output Low Voltage: All inputs	V_{OL}	—	0.2	—	0.2		
Input Leakage Current: Any input 0V ≤ V_{IN} ≤ V_{DD} (All other pins not under test = 0V)	I_I	-1.0	1.0	-1.0	1.0	µA	
Output Leakage Current: DQ disabled; 0V ≤ V_{OUT} ≤ V_{DDQ}	I_{OZ}	-1.5	1.5	-1.5	1.5	µA	

Notes:

1. *All voltages referenced to V_{SS}. V_{DD} = V_{DDQ} = +1.8V ±0.1V.*
2. *The minimum specifications are used only to indicate cycle time at which proper operation over the full temperature range (-25°C ≤ T_A ≤ +85°C for standard parts; -40°C ≤ T_A ≤ +85°C for IT parts) is ensured.*
3. *An initial pause of 100µs is required after power up, followed by two Auto Refresh commands, before proper device operation is ensured. (V_{DD} and V_{DDQ} must be powered up simultaneously. V_{SS} and V_{SSQ} must be at same potential.) The two Auto Refresh command wake-ups should be repeated any time the t_{REF} refresh requirement is exceeded.*
4. *V_{IH} overshoot: $V_{IH\ (MAX)}$ = V_{DDQ} + 2V for a pulse width ≤ 3ns, and the pulse width cannot be greater than one third of the cycle rate. V_{IL} undershoot: $V_{IL\ (MIN)}$ = -2V for a pulse width ≤ 3ns.*
5. ***ESD Immunity:** Spansion Flash memory Multi-Chip Products (MCPs) may contain component devices that are developed by Spansion and component devices that are developed by a third party (third-party components). Spansion components are tested and guaranteed to the ESD immunity levels listed in the corresponing Spanion Flash memory Qualification Database. Third-party components are neither tested nor guaranteed by Spansion for ESD immunity. However, ESD test results for third-party components may be available from the component manufacturer. Component manufacturer contact information is listed in the Spansion MCP Qualification Report, when available. The Spanion Flash memory Qualification Database and Spansion MCP Qualification Report are available from Spansion and Fujitsu sales offices.*

Table 48.2 AC Electrical Characteristics and Operating Conditions

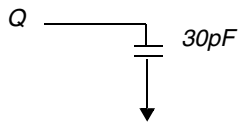
Parameter	Symbol	Min	Max	Units	Notes
Input High Voltage: Logic 1; All inputs	V_{IH}	1.4	—	V	
Input Low Voltage: Logic 0; All inputs	V_{IL}	—	+0.4	V	

Table 48.3 Electrical Characteristics and Recommended AC Operating Conditions

AC Characteristics			-75		-I0		Units	Notes
Parameter		Symbol	Min	Max	Min	Max		
Access time from CLK (pos. edge)	CL=3	t_{AC} (3)		5.4		7	ns	27
	CL=2	t_{AC} (2)		6		8		
Address hold time		t_{AH}	1		1			
Address setup time		t_{AS}	2.5		2.5			
CLK high-level width		t_{CH}	2.5		3			
CLK low-level width		t_{CL}	2.5		3			
Clock cycle time	CL=3	t_{CK} (3)	7.5	100	9.6	100		6
	CL=2	t_{CK} (2)	10	100	12	100		6
CKE hold time		t_{CKH}	1		1			
CKE setup time		t_{CKS}	2.5		2.5			
CS#, RAS#, CAS#, WE#, DQM hold time		t_{CMH}	1		1			
CS#, RAS#, CAS#, WE#, DQM setup time		t_{CMS}	2.5		2.5			
Data-in hold time		t_{DH}	1		1			
Data-in setup time		t_{DS}	2.5		2.5			
Data-out high-impedance time	CL=3	t_{HZ} (3)		5.4		7		7
	CL=2	t_{HZ} (2)		6		8		7
Data-out low-impedance time		t_{LZ}	1		1			
Data-out hold time (load)		t_{OH}	3		2.5			
Data-out hold time (no load)		t_{OHN}	1.8		1.8			8
Active to Precharge command		t_{RAS}	45	120,000	50	120,000		
Active to Active command period		t_{RC}	80		100			
Active to Read or Write delay		t_{RCD}	22.5		20			
Refresh period (4,096 rows)		t_{REF}		64		64	ms	
Auto Refresh period		t_{RFC}	80		100		ns	
Precharge command period		t_{RP}	22.5		20			
Active bank a to Active bank b command		t_{RRD}	15		20			
Transition time		t_T	0.3	1.2	0.3	1.2		9
Write recovery time		t_{WR} (a)	1 CLK + 7.5 ns		1 CLK + 5ns		—	10
		t_{WR} (m)	15		15		ns	11
Exit Self Refresh to Active command		t_{XSR}	80		100			12

Notes:

1. *The minimum specifications are used only to indicate cycle time at which proper operation over the full temperature range (-25°C ≤ T_A ≤ +85°C for standard parts; -40°C ≤ T_A ≤ +85°C for IT parts) is ensured.*
2. *An initial pause of 100µs is required after power up, followed by two Auto Refresh commands, before proper device operation is ensured. (V_{DD} and V_{DDQ} must be powered up simultaneously. V_{SS} and V_{SSQ} must be at same potential.) The two Auto Refresh command wake-ups should be repeated any time the t_{REF} refresh requirement is exceeded.*
3. *In addition to meeting the transition rate specification, the clock and CKE must transit between V_{IH} and V_{IL} (or between V_{IL} and V_{IH}) in a monotonic manner.*
4. *Outputs measured for 1.8V at 0.9V with equivalent load:*



5. *AC timing and IDD tests have V_{IL} and V_{IH}, with timing referenced to $V_{IH}/2$ = cross-over point. If the input transition time is longer than $t_{T\ (MAX)}$, then the timing is referenced at $V_{IL\ (MAX)}$ and $V_{IH\ (MIN)}$ and no longer at the $V_{IH}/2$ crossover point.*
6. *The clock frequency must remain constant (stable clock is defined as a signal cycling within timing constraints specified for the clock pin) during access or precharge states (Read, Write, including t_{WR}, and Precharge commands). CKE may be used to reduce the data rate*
7. *t_{HZ} defines the time at which the output achieves the open circuit condition; it is not a reference to V_{OH} or V_{OL}. The last valid data element will meet t_{OH} before going High-Z.*
8. *Parameter guaranteed by design.*
9. *AC characteristics assume t_T = 1ns.*
10. *Auto precharge mode only. May not exceed limit set for precharge mode.*
11. *Precharge mode only.*
12. *CLK must be toggled a minimum of two times during this period.*

Table 48.4 AC Functional Characteristics

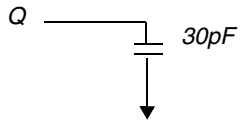
Parameter		Symbol	-75 & -10	Units	Notes
Read/Write command to Read/Write command		t_{CCD}	1	t_{CK}	7
CKE to clock disable or power-down entry mode		t_{CKED}	1		8
CKE to clock enable or power-down exit setup mode		t_{PED}	1		8
DQM to input data delay		t_{DQD}	0		7
DQM to data mask during Writes		t_{DQM}	0		7
DQM to data high-impedance during Reads		t_{DQZ}	2		7
Write command to input data delay		t_{DWD}	0		7
Data-in to Active command		t_{DAL}	5		9, 10
Data-in to Precharge command		t_{DPL}	2		11, 10
Last data-in to burst Stop command		t_{BDL}	1		7
Last data-in to new Read/Write command		t_{CDL}	1		7
Last data-in to Precharge command		t_{RDL}	2		11, 10
Load Mode Register command to Active or Refresh command		t_{MRD}	2		12
Data-out to high-impedance from Precharge command	CL = 3	t_{ROH} (3)	3		7
	CL = 2	t_{ROH} (2)	2		7

Notes:

1. *The minimum specifications are used only to indicate cycle time at which proper operation over the full temperature range (-25°C ≤ T_A ≤ +85°C for standard parts; -40°C ≤ T_A ≤ +85°C for IT parts) is ensured.*
2. *An initial pause of 100µs is required after power up, followed by two Auto Refresh commands, before proper device operation is ensured. (V_{DD} and V_{DDQ} must be powered up simultaneously. V_{SS} and V_{SSQ} must be at same potential.) The two*

Auto Refresh command wake-ups should be repeated any time the t_{REF} refresh requirement is exceeded.

3. *AC characteristics assume t_T = 1ns.*
4. *In addition to meeting the transition rate specification, the clock and CKE must transit between V_{IH} and V_{IL} (or between V_{IL} and V_{IH}) in a monotonic manner.*
5. *Outputs measured for 1.8V at 0.9V with equivalent load:*



6. *AC timing and I_{DD} tests have V_{IL} and V_{IH}, with timing referenced to VIH/2 = crossover point. If the input transition time is longer than $t_{T\ (MAX)}$, then the timing is referenced at $V_{IL\ (MAX)}$ and $V_{IH\ (MIN)}$ and no longer at the V_{IH}/2 crossover point.*
7. *Required clocks are specified by JEDEC functionality and are not dependent on any timing parameter.*
8. *Timing actually specified by t_{CKS}; clock(s) specified as a reference only at minimum cycle rate.*
9. *Timing actually specified by t_{WR} plus t_{RP}; clock(s) specified as a reference only at minimum cycle rate.*
10. *Based on t_{CK} = 9.6ns for -10.*
11. *Timing actually specified by t_{WR}.*
12. *JEDEC and PC100 specify three clocks.*

Table 48.5 IDD Specifications and Conditions

Parameter/Condition		Symbol	-75 Max	-10 Max	Units	Notes
Operating Current: Active Mode; Burst = 2; Read or Write; $t_{RC} = t_{RC\ (MIN)}$		I_{DD1}	50	50	mA	6, 7, 8, 9
Standby Current: Power-Down Mode; All banks idle; CKE = Low		I_{DD2}	220	150	µA	9
Standby Current: Active Mode; CKE = High; CS# = High; All banks active after t_{RCD} met; No accesses in progress		I_{DD3}	5	30	mA	6, 12, 8, 9
Operating Current: Burst Mode; Continuous burst; Read or Write; All banks active		I_{DD4}	50	80		6, 7, 8, 9
Auto Refresh Current CKE = High; CS# = High	$t_{RC} = t_{RFC\ (MIN)}$	I_{DD5}	105	120		6, 12, 7, 8, 9, 10
	t_{RFC} = 15.625µs	I_{DD6}	3	2		
Deep Power Down		I_{ZZ}	10	10	µA	11

Notes:

1. *All voltages referenced to V_{SS}. $V_{DD} = V_{DDQ}$ = +1.8V ±0.1V.*
2. *The minimum specifications are used only to indicate cycle time at which proper operation over the full temperature range (-25°C ≤ T_A ≤ +85°C for standard parts; -40°C ≤ T_A ≤ +85°C for IT parts) is ensured.*
3. *An initial pause of 100µs is required after power up, followed by two Auto Refresh commands, before proper device operation is ensured. (V_{DD} and V_{DDQ} must be powered up simultaneously. V_{SS} and V_{SSQ} must be at same potential.) The two*

Auto Refresh command wake-ups should be repeated any time the t_{REF} refresh requirement is exceeded.

4. *AC timing and I_{DD} tests have V_{IL} and V_{IH}, with timing referenced to VIH/2 = cross-over point. If the input transition time is longer than $t_{T\ (MAX)}$, then the timing is referenced at $V_{IL\ (MAX)}$ and $V_{IH\ (MIN)}$ and no longer at the V_{IH}/2 crossover point.*
5. *I_{DD} specifications are tested after the device is properly initialized.*
6. *I_{DD} is dependent on output loading and cycle rates. Specified values are obtained with minimum cycle time and the outputs open.*
7. *The I_{DD} current will increase or decrease proportionally according to the amount of frequency alteration for the test condition.*
8. *Address transitions average one transition every two clocks.*
9. *For -10, CL = 3 and t_{CK} = 9.6ns.*
10. *CKE is High during refresh command period $t_{RFC\ (MIN)}$ else CKE is Low. The I_{DD6} limit is actually a nominal value and does not result in a fail value.*
11. *Deep power down current is a nominal value at 25°C. The parameter is not tested.*

Table 48.6 I_{DD7} - Self Refresh Current Options

Temperature Compensated Self Refresh Parameter/Condition	Max Temperature	-75	-10	Units	Notes
Self Refresh Current: CKE < 0.2V – 4 Banks Open	85°C	220	200	µA	2
	70°C	175	160		
	45°C	150	140		
	15°C	125	120		
Self Refresh Current: CKE < 0.2V – 2 Banks Open	85°C	160	160		
	70°C	125	130		
	45°C	125	120		
	15°C	85	110		
Self Refresh Current: CKE < 0.2V – 1 Bank Open	85°C	130	130		
	70°C	105	120		
	45°C	90	110		
	15°C	75	100		
Self Refresh Current: CKE < 0.2V – 1/2 Bank Open	85°C	TBD	120		
	70°C	TBD	110		
	45°C	TBD	100		
	15°C	TBD	90		
Self Refresh Current: CKE < 0.2V – 1/4 Bank Open	85°C	TBD	115		
	70°C	TBD	105		
	45°C	TBD	95		
	15°C	TBD	90		

Notes:

1. *V_{DD} = V_{DDQ} = +1.8V ±0.1V.*
2. *Enables on-chip refresh and address counters.*

Table 48.7 Capacitance

Parameter	Symbol	Min	Max	Units	Notes
Input Capacitance: CLK	C_{I1}	1.5	4.0	pF	229
Input Capacitance: All other input-only pins	C_{I2}	1.5	4.0	pF	3
Input/Output Capacitance: DQ	C_{IO}	3.0	6.0	pF	4

Notes:

1. *This parameter is sampled. V_{DD}, V_{DDQ} = +1.8V; TA = 25°C; pin under test biased at 1.4V. f = 1 MHz.*
2. *PC100 specifies a maximum of 4pF.*
3. *PC100 specifies a maximum of 5pF.*
4. *PC100 specifies a maximum of 6.5pF*

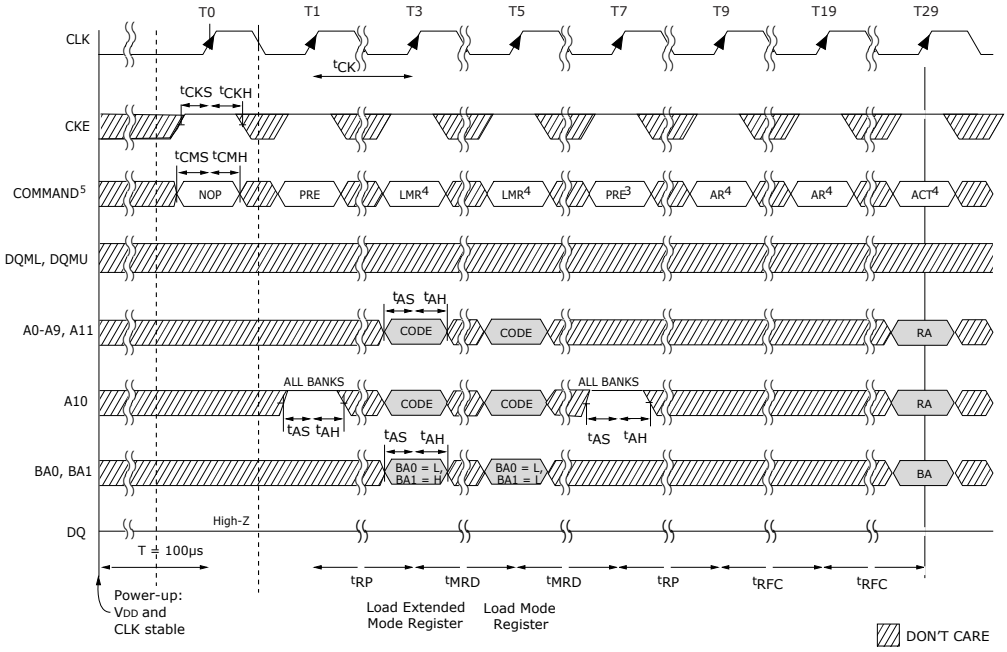


Figure 48.1 Initialize and Load Mode Register

Notes:

1. *The two Auto Refresh commands at T9 and T19 may be applied before either Load Mode Register (LMR) command.*
2. *PRE = Precharge command, LMR = Load Mode Register command, AR = Auto Refresh command, ACT = Active command, RA = Row Address, BA = Bank Address.*
3. *Optional refresh command.*
4. *The Load Mode Register for both MR/EMR and 2 Auto Refresh commands can be in any order. However, all must occur prior to an Active command.*
5. *Device timing is -10 with 104 MHz clock.*

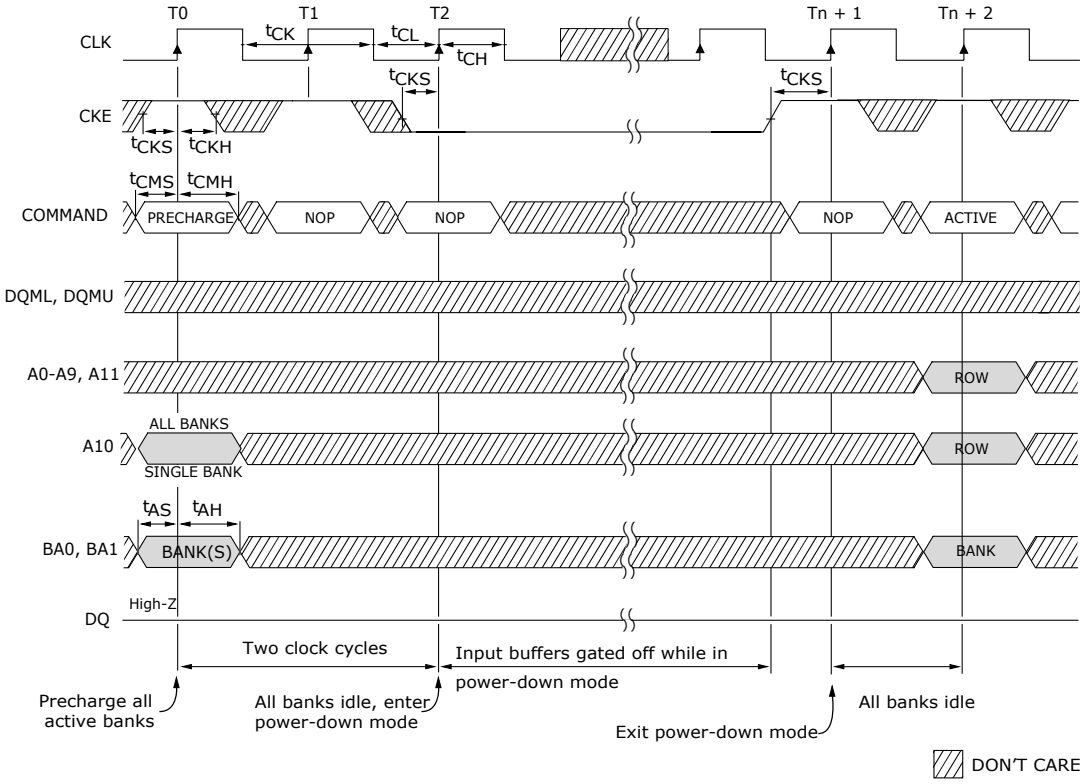


Figure 48.2. Power Down Mode

Note: *Violating refresh requirements during power-down may result in a loss of data.*

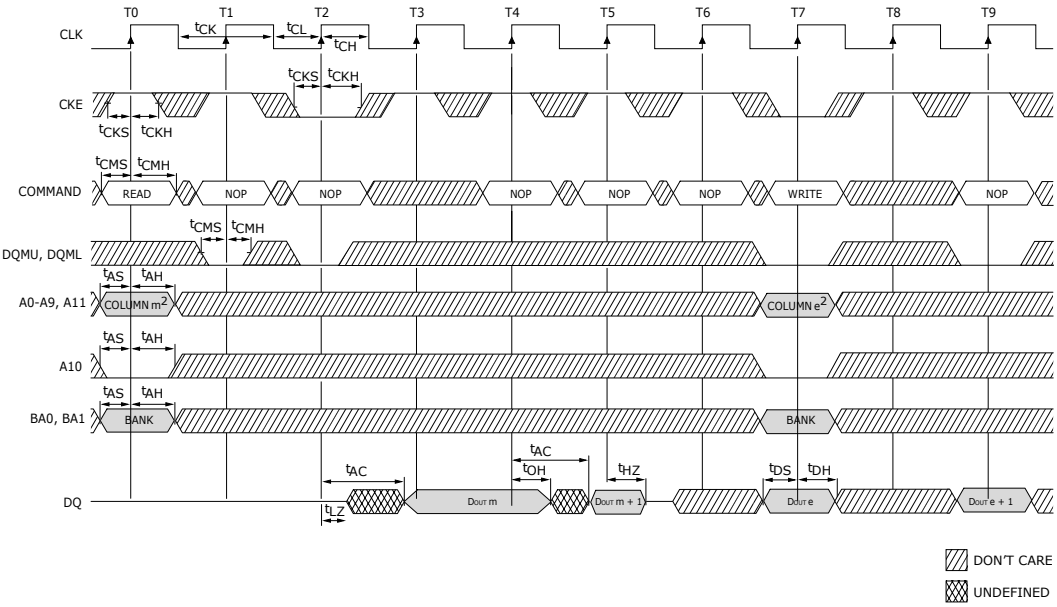


Figure 48.3. Clock Suspend Mode

Notes:

1. *For this example, the burst length = 2, the CAS latency = 3, and auto precharge is disabled.*
2. *A9 and A11 = "Don't Care."*

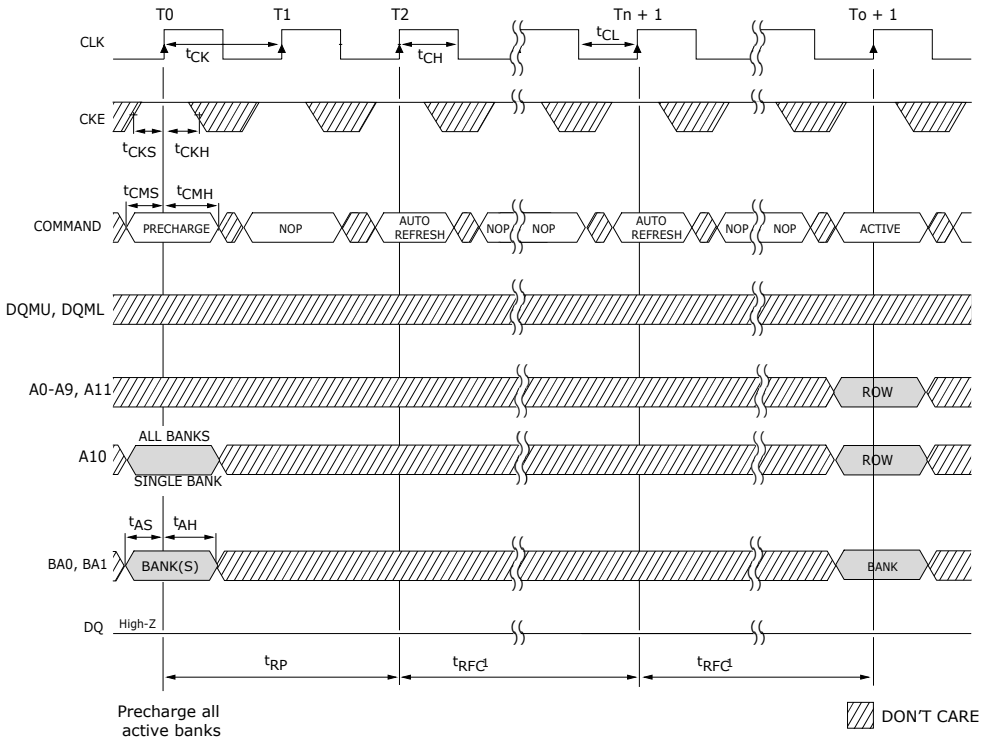


Figure 48.4. Auto Refresh Mode

Note: *Each Auto Refresh command performs a refresh cycle. Back-to-back commands are not required.*

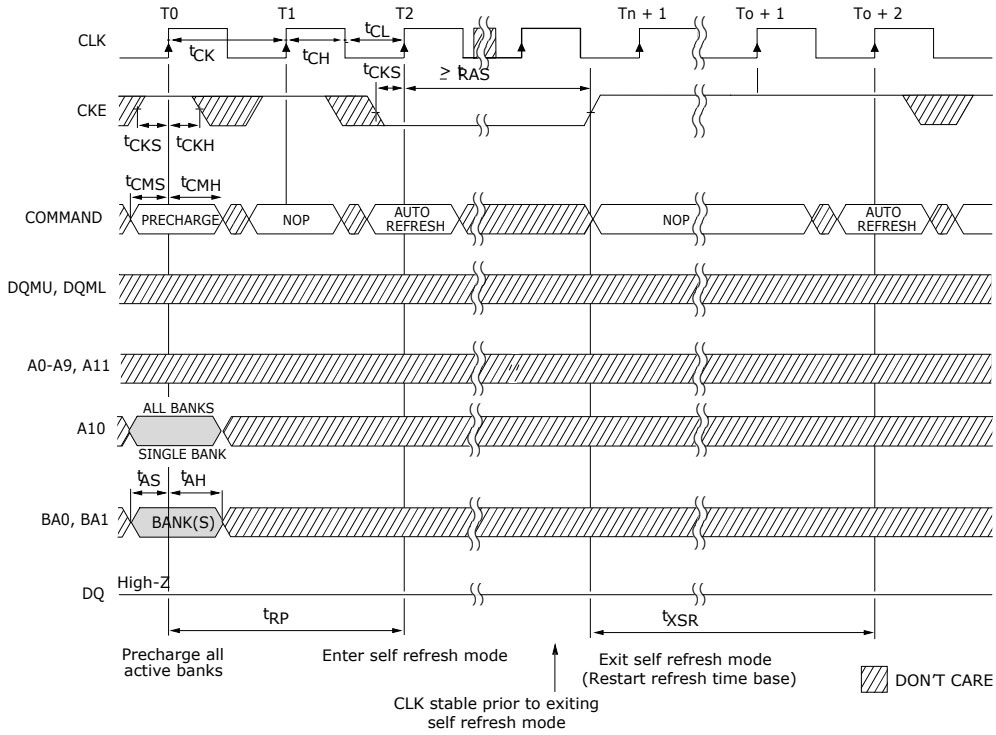


Figure 48.5. Self Refresh Mode

Note: *Each Auto Refresh command performs a refresh cycle. Back-to-back commands are not required.*

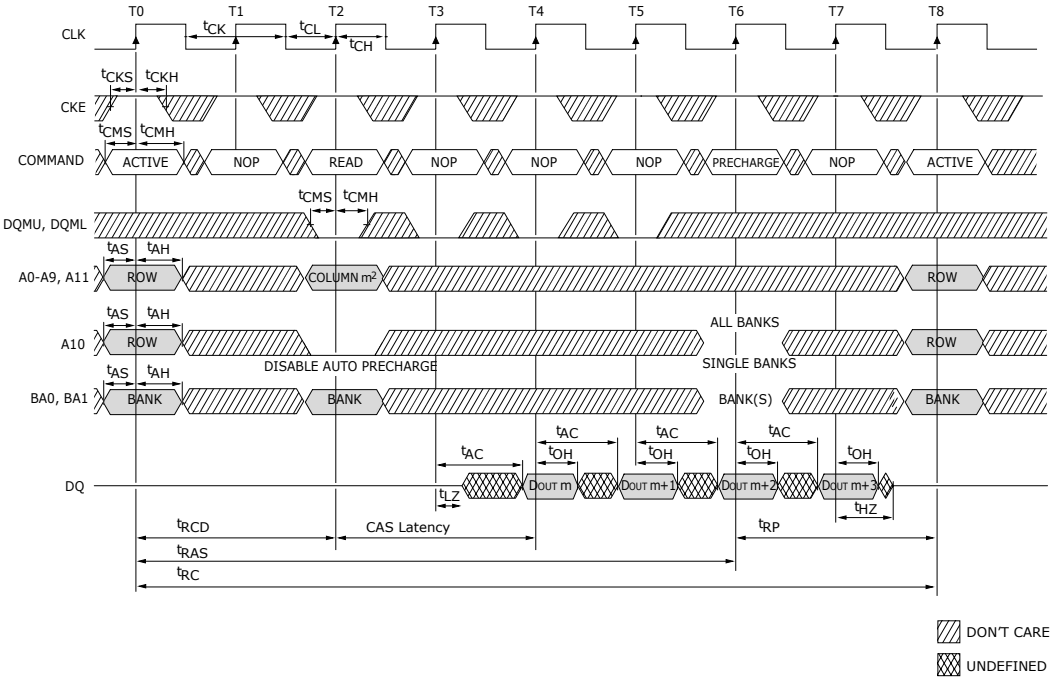


Figure 48.6. Read - Without Auto Precharge

Notes:

1. *For this example, the burst length = 4, the CAS latency = 2, and the Read burst is followed by a "manual" Precharge.*
2. *A9 and A11 = "Don't Care."*

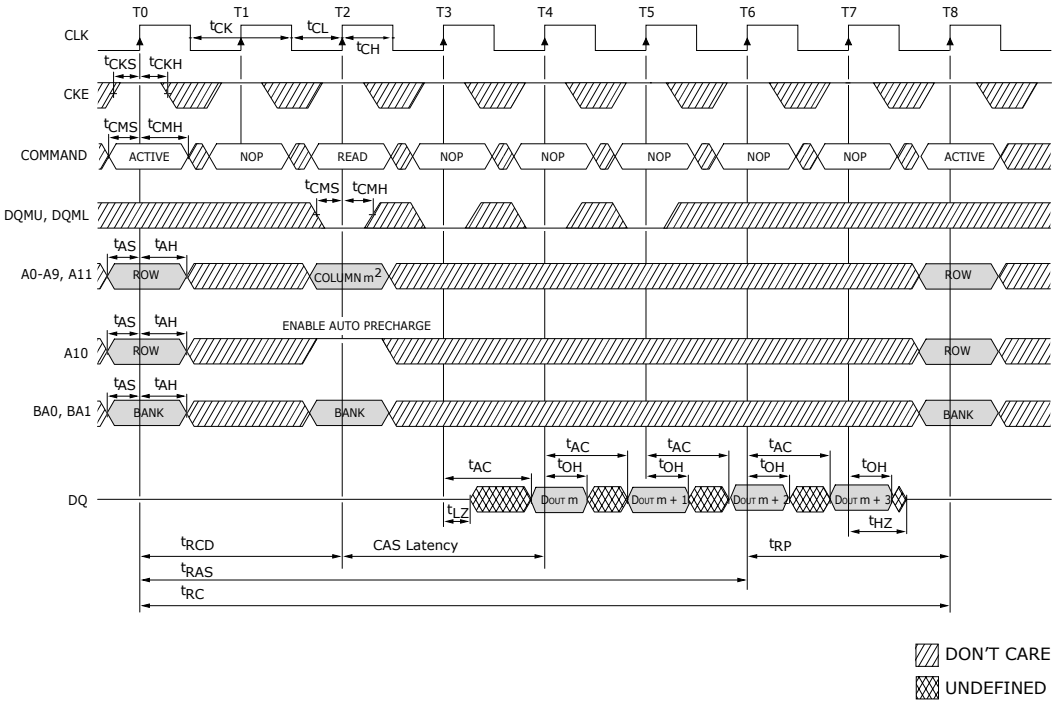


Figure 48.7. Read - With Auto Precharge

Notes:

1. *For this example, the burst length = 4, the CAS latency = 2.*
2. *A9 and A11 = "Don't Care."*

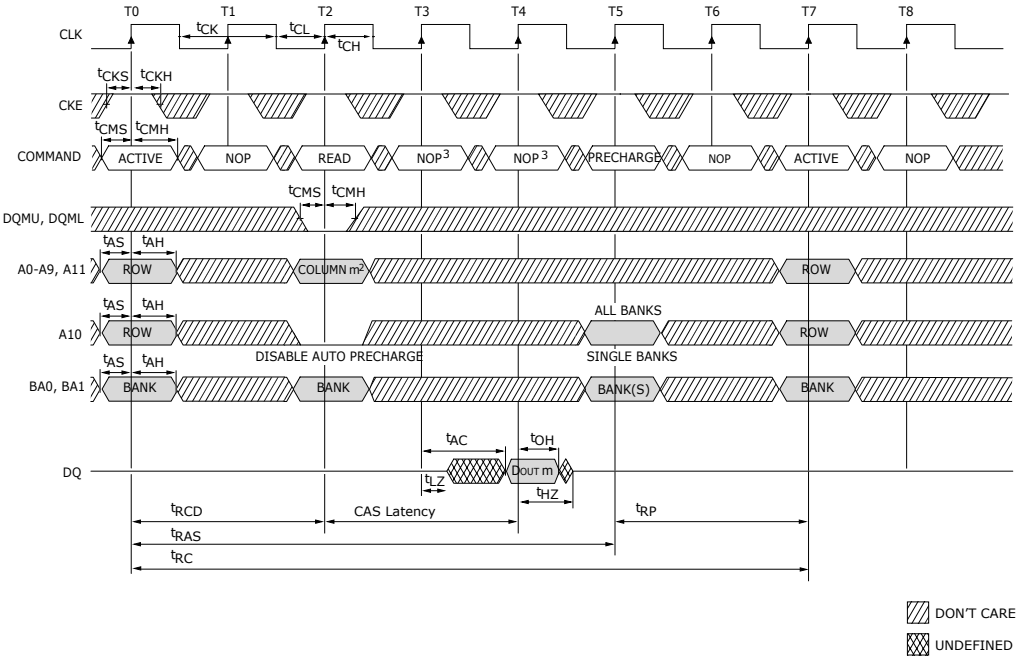


Figure 48.8. Single Read - Without Auto Precharge

Notes:

1. *For this example, the burst length = 4, the CAS latency = 2, and the Read burst is followed by a "manual" Precharge.*
2. *A9 and A11 = "Don't Care."*
3. *Precharge command not allowed or t_{RAS} would be violated.*

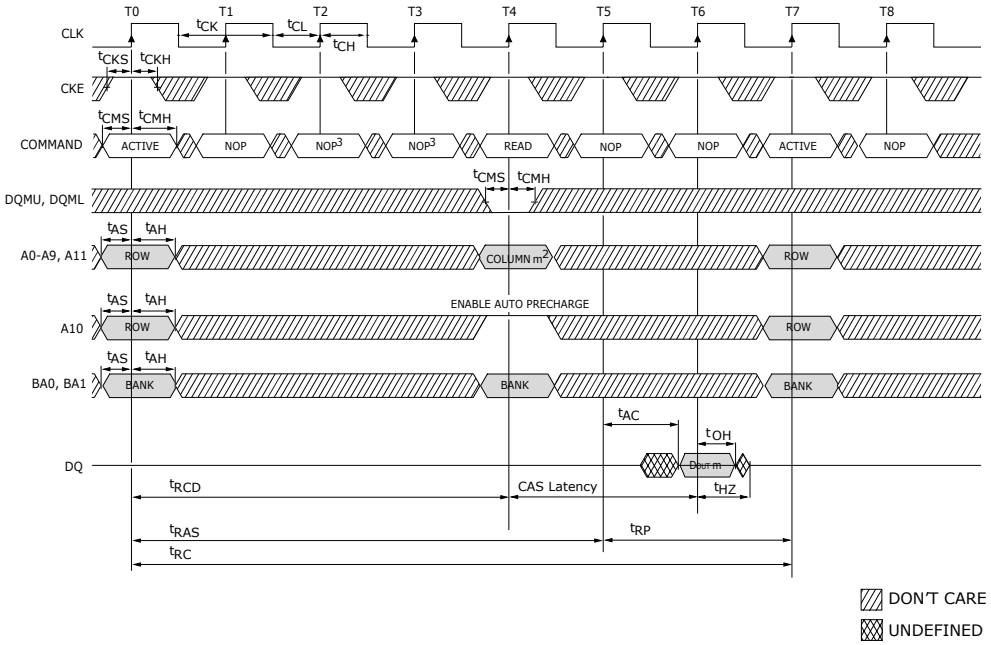


Figure 48.9. Single Read - With Auto Precharge

Notes:

1. *For this example, the burst length = 4, the CAS latency = 2, and the Read burst is followed by a "manual" Precharge.*
2. *A9 and A11 = "Don't Care."*
3. *Precharge command not allowed or t_{RAS} would be violated.*

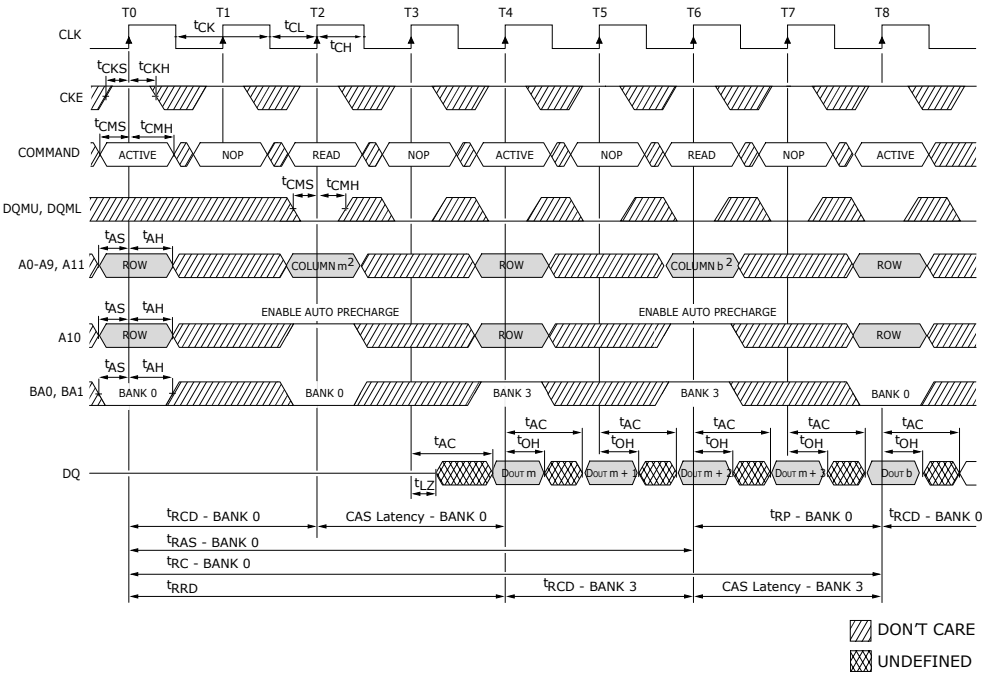


Figure 48.10. Alternating Bank Read Accesses

Notes:

1. *For this example, the burst length = 4, the CAS latency = 2.*
2. *A9 and A11 = "Don't Care."*

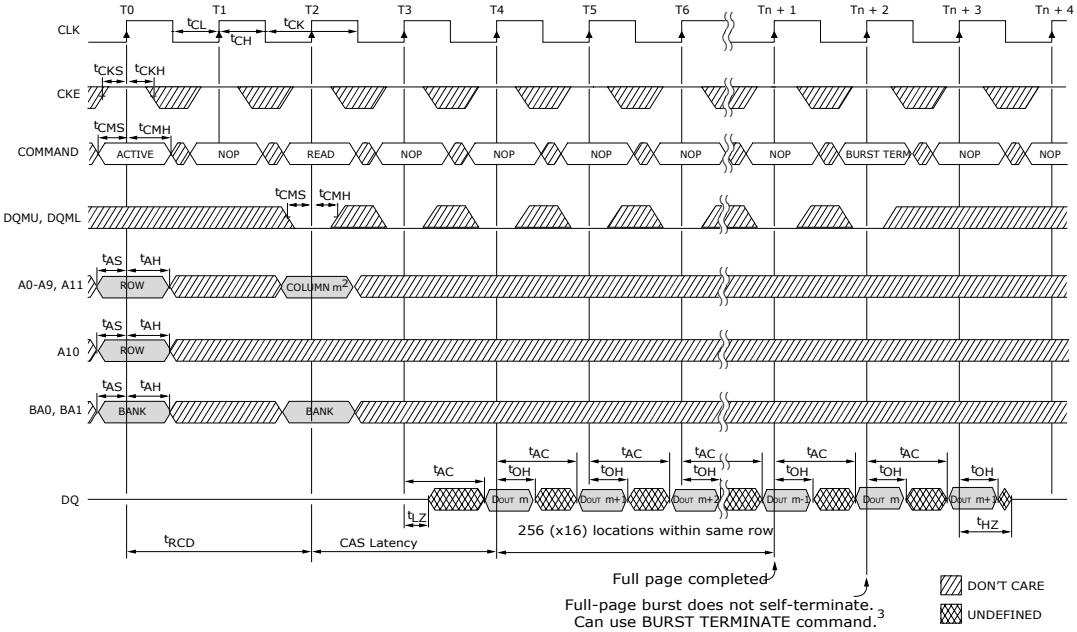


Figure 48.11. Read - Full-Page Burst

Notes:

1. *For this example, the CAS latency = 2.*
2. *A9 and A11 = "Don't Care."*
3. *Page left open; no t_{RP}.*

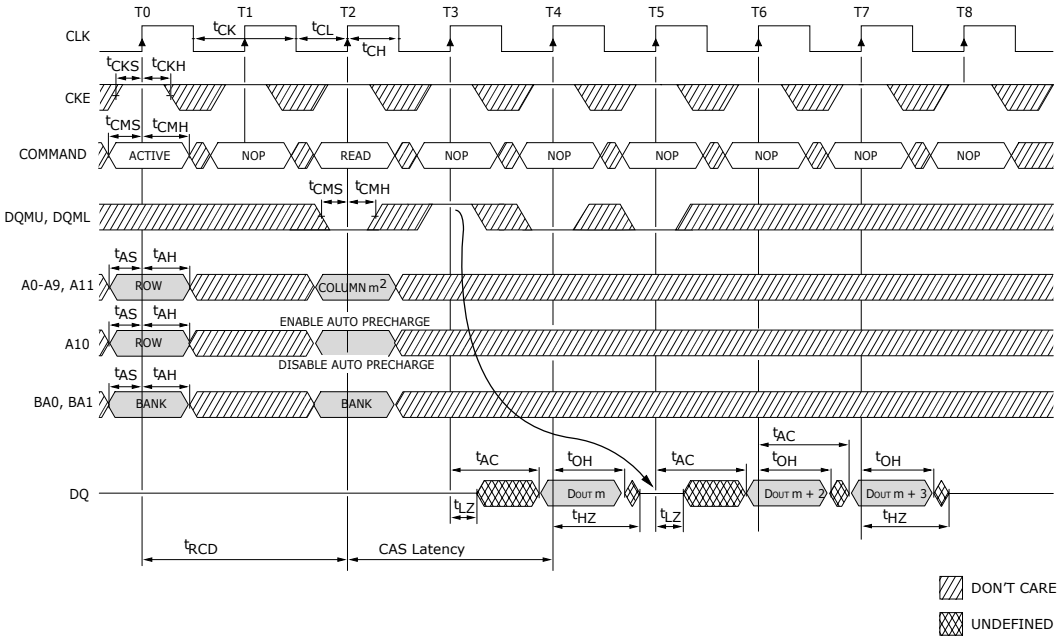


Figure 48.12. Read - DQM Operation

Notes:

1. *For this example, the CAS latency = 2.*
2. *A9 and A11 = "Don't Care."*

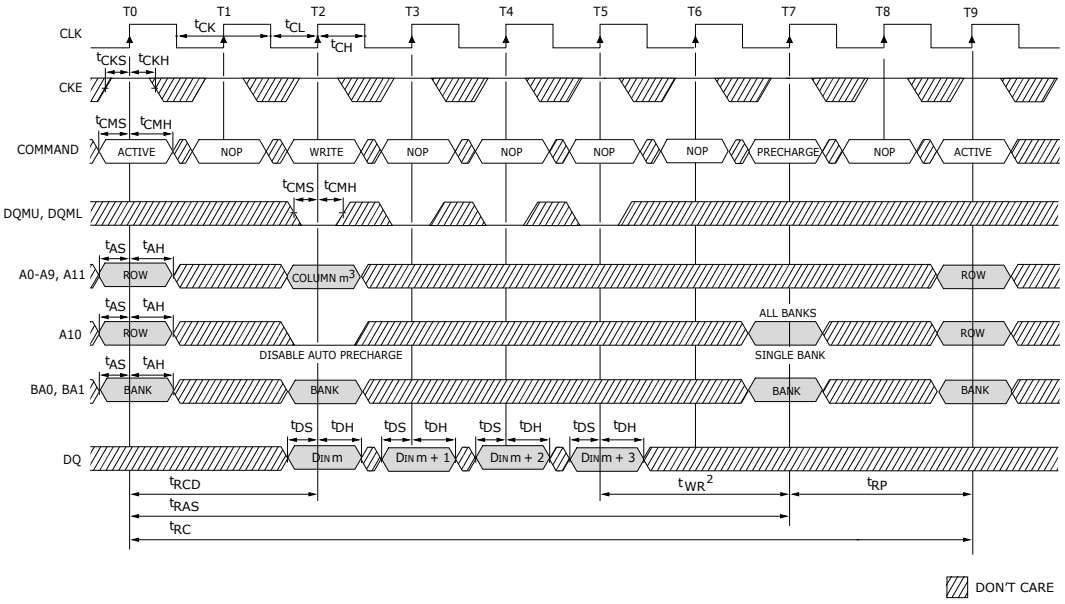


Figure 48.I3. Write - Without Auto Precharge

Notes:

1. *For this example, the burst length = 4, and the Write burst is followed by a "manual" Precharge.*
2. *15ns is required between <DIN m + 3> and the Precharge command, regardless of frequency.*
3. *A9 and A11 = "Don't Care."*

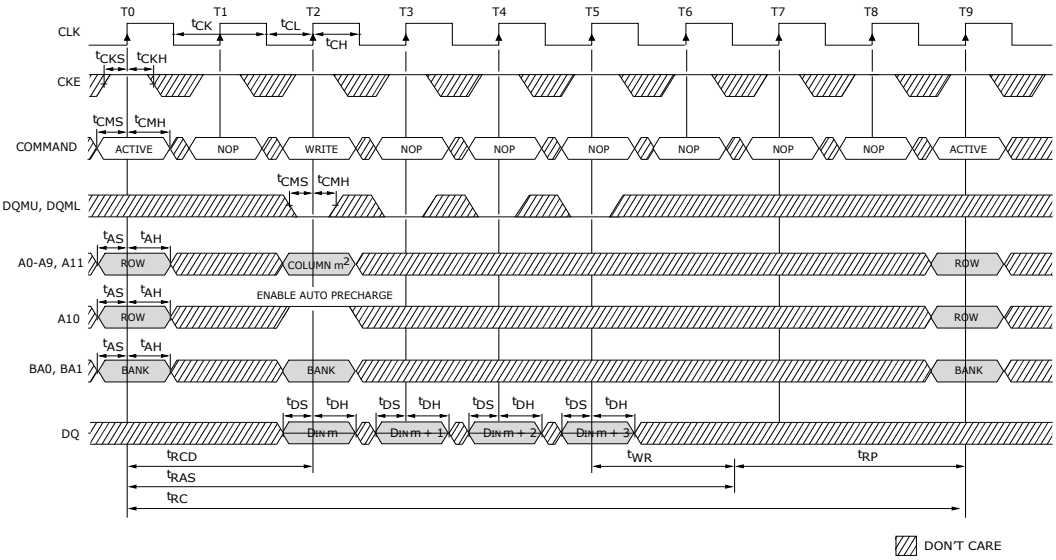


Figure 48.I4. Write - With Auto Precharge

Notes:

1. *For this example, the burst length = 4.*
2. *A9 and A11 = "Don't Care."*

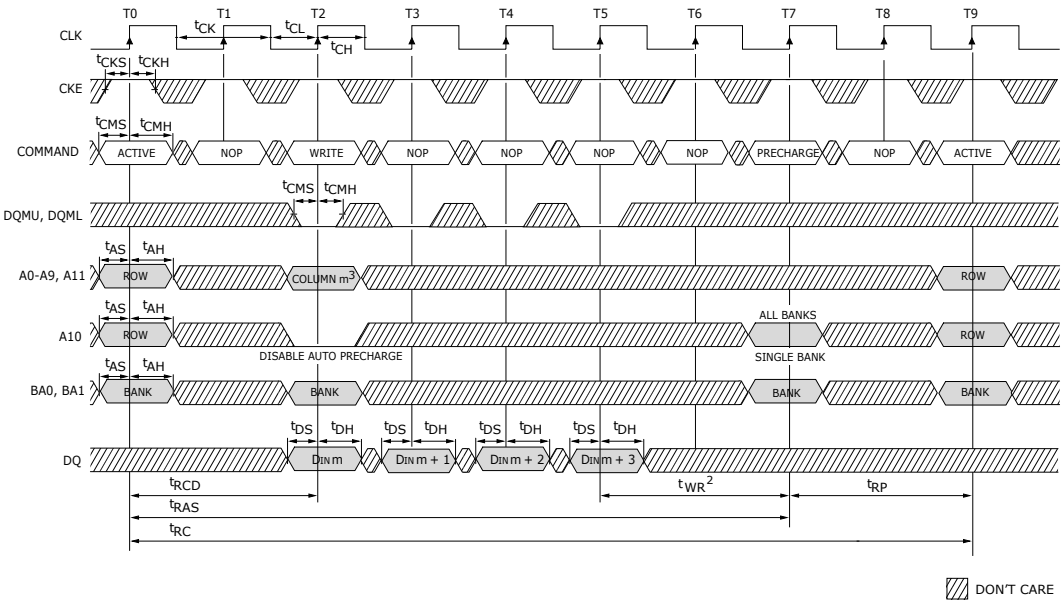


Figure 48.15. Single Write - Without Auto Precharge

Notes:

1. *For this example, the burst length = 1, and the Write burst is followed by a "manual" Precharge.*
2. *15ns is required between <DIN m> and the Precharge command, regardless of frequency.*
3. *A9 and A11 = "Don't Care."*
4. *Precharge command not allowed else t_{RAS} would be violated.*

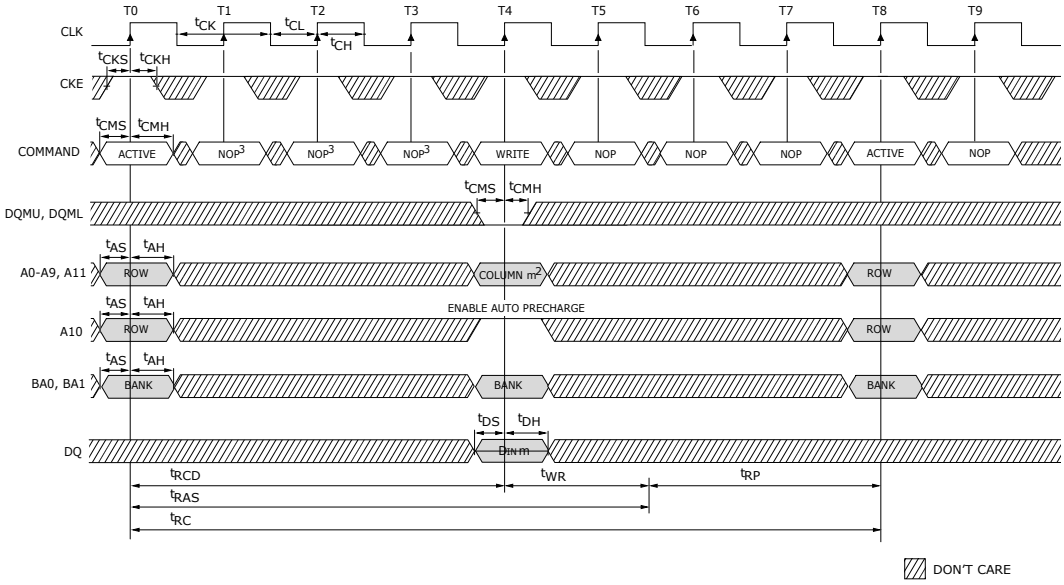


Figure 48.16. Single Write with Auto Precharge

Notes:

1. *For this example, the burst length = 1, and the Write burst is followed by a "manual" Precharge.*
2. *15ns is required between <DIN m> and the Precharge command, regardless of frequency.*
3. *A9 and A11 = "Don't Care."*
4. *Write command not allowed else t_{RAS} would be violated.*

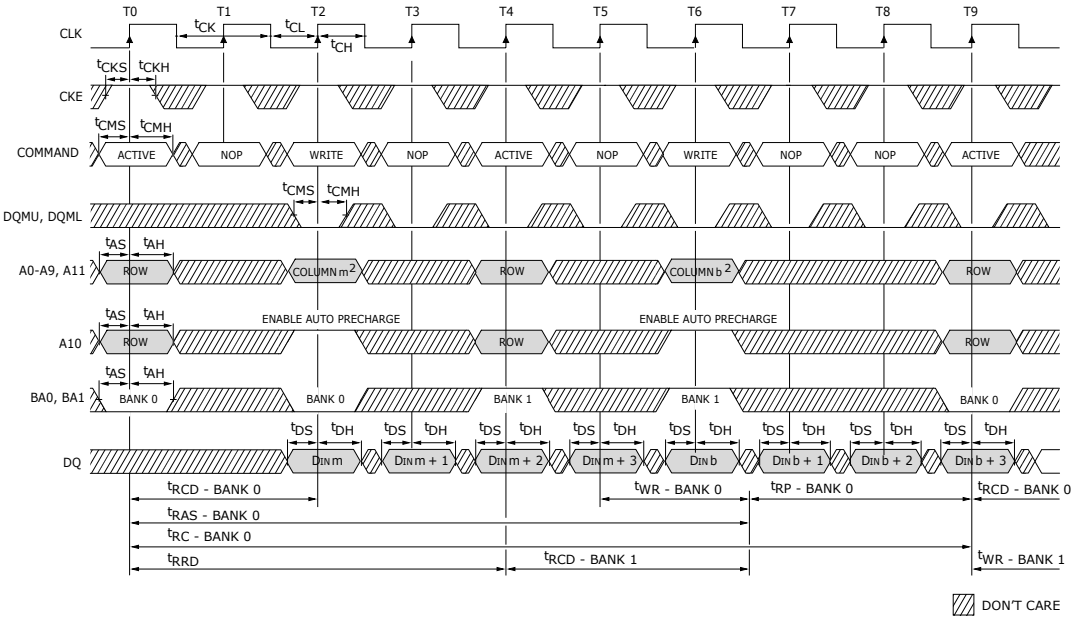


Figure 48.17. Alternating Bank Write Accesses

Notes:

1. *For this example, the burst length = 4.*
2. *A9 and A11 = "Don't Care."*

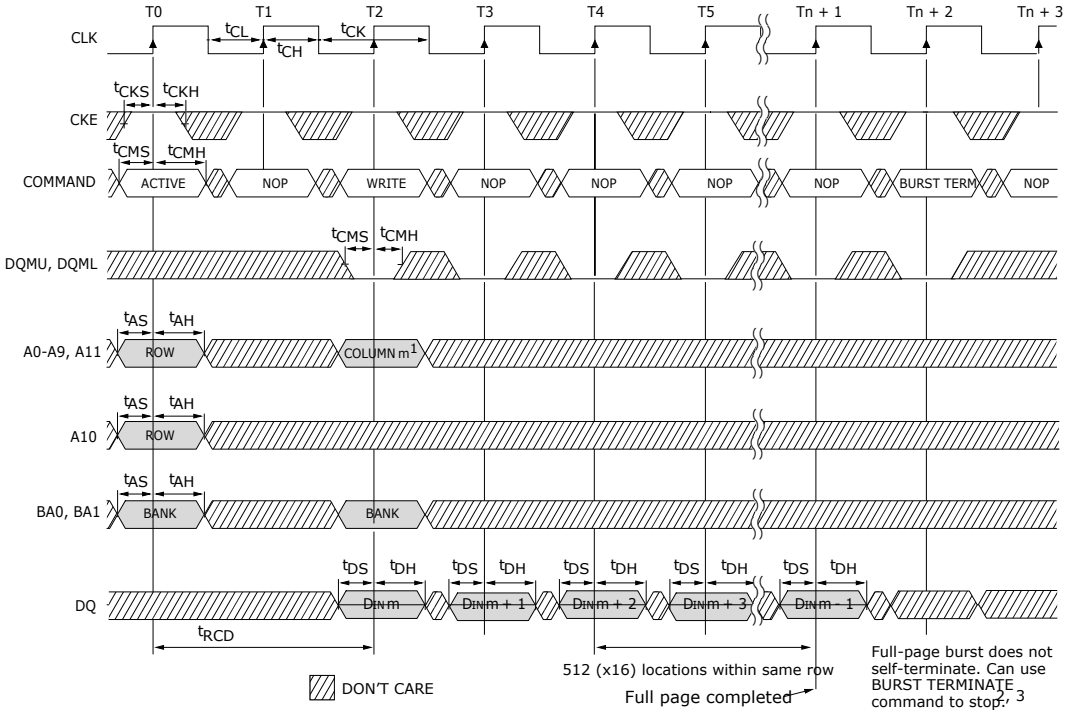


Figure 48.18. Write - Full Page Burst

Notes:

1. *A9 and A11 = "Don't Care."*
2. *t_{WR} must be satisfied prior to Precharge command.*
3. *Page left open; no t_{RP}.*

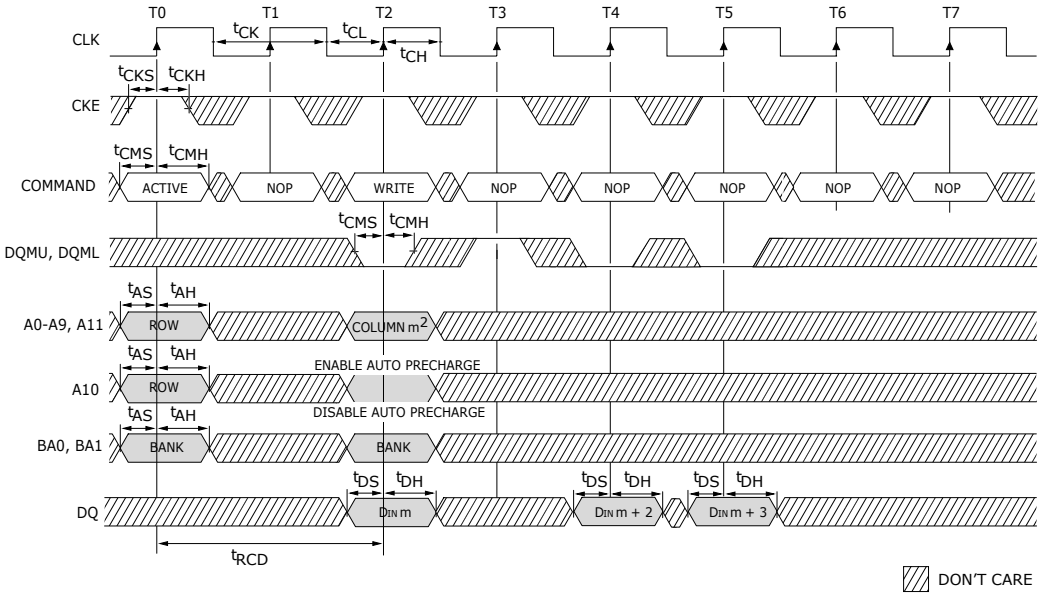


Figure 48.19. Write - DQM Operation

Notes:

1. *For this example, the burst length = 4.*
2. *A9 and A11 = "Don't Care."*

49 Revision Summary

49.1 Revision A0 (April 1, 2005)

Initial Release

49.2 Revision A1 (April 25, 2005)

Added 133 MHz speed grade option

49.3 Revision A2 (April 25, 2005)

Added revision history

49.4 Revision A3 (April 25, 2005)

Added a note on ESD Immunity to the *DC Electrical Characteristics and Operating Conditions* table.

Mobile SDRAM Type 2

2M x l6Bit x 4 Banks SDRAM

PRELIMINARY

Features

- **1.8V power supply.**
- **LVCMOS compatible with multiplexed address.**
- **Four banks operation.**
- **MRS cycle with address key programs.**
  - CAS latency (1, 2 & 3).
  - Burst length (1, 2, 4, 8 & Full page).
  - Burst type (Sequential & Interleave).
- **EMRS cycle with address key programs.**
- **All inputs are sampled at the positive going edge of the system clock.**
- **Burst read single-bit write operation.**
- **Special Function Support.**
  - PASR (Partial Array Self Refresh).
  - Internal TCSR (Temperature Compensated Self Refresh)
  - DS (Driver Strength)
- **DQM for masking.**
- **Auto refresh.**
- **64ms refresh period (4K cycle).**
- **Extended Temperature Operation (-25°C ~ 85°C)**

50 Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Voltage on any pin relative to V_{SS}	V_{IN}, V_{OUT}	-1.0 ~ 2.6	V
Voltage on V_{DD} supply relative to V_{SS}	V_{DD}, V_{DDQ}	-1.0 ~ 2.6	V
Storage temperature	T_{STG}	-55 ~ +150	°C
Power dissipation	P_D	1.0	W
Short circuit current	I_{OS}	50	mA

Notes:

1. *Permanent device damage may occur if ABSOLUTE MAXIMUM RATINGS are exceeded.*
2. *Functional operation should be restricted to recommended operating condition.*
3. *Exposure to higher than recommended voltage for extended periods of time could affect device reliability.*

5I DC Operating Conditions

Recommended operating conditions (Voltage referenced to V_{SS} = 0V, T_A = -25°C ~ +85°C.

Parameter	Symbol	Min	Typ	Max	Unit	Note
Supply voltage	V_{DD}	1.65	1.8	1.95	V	
	V_{DDQ}	1.65	1.8	1.95		
Input logic high voltage	V_{IH}	0.8 x V_{DDQ}	1.8	V_{DDQ} + 0.3		1
Input logic low voltage	V_{IL}	-0.3	0	0.3		2
Output logic high voltage	V_{OH}	V_{DDQ} -0.2	—	—		I_{OH} = -0.1mA
Output logic low voltage	V_{OL}	—	—	0.2		I_{OL} = 0.1mA
Input leakage current	I_{LI}	-10	—	10	µA	

Notes:

1. *V_{IH} (max) = 2.2V AC.The overshoot voltage duration is ≤ 3ns.*
2. *V_{IL} (min) = -1.0V AC. The undershoot voltage duration is ≤ 3ns.*
3. *Any input 0V ≤ V_{IN} ≤ V_{DDQ}. Input leakage currents include Hi-Z output leakage for all bi-directional buffers with tri-state outputs.*
4. *Dout is disabled, 0V ≤ V_{OUT} ≤ V_{DDQ}.*

52 Capacitance

(V_{DD} = 1.8V, TA = 23°C, f = 1MHz, V_{REF} =0.9V ± 50 mV)

Pin	Symbol	Min	Max	Unit	Note
Clock	C_{CLK}	2.0	4.0	pF	
RAS#, CAS#, WE#, CS#, CKE, DQM	C_{IN}	2.0	4.0		
Address	C_{ADD}	2.0	4.0		
DQ0 ~ DQ15	C_{OUT}	3.0	6.0		

53 DC Characteristics

Parameter	Symbol	Test Condition		Value	Unit	Note
Operating Current (One Bank Active)	I_{CC1}	Burst length = 1 $t_{RC} \geq t_{RC}$ (min) I_O = 0 mA		35	mA	1
Precharge Standby Current in power-down mode	$I_{CC2}P$	CKE ≤ V_{IL}(max), t_{CC} = 10ns		0.3	mA	
	$I_{CC2}PS$	CKE & CLK ≤ V_{IL}(max), t_{CC} = ∞		0.3	mA	
Precharge Standby Current in non power-down mode	$I_{CC2}N$	CKE ≥ V_{IH}(min), CS# ≥ V_{IH}(min), t_{CC} = 10ns Input signals are changed one time during 20ns		10	mA	
	$I_{CC2}NS$	CKE ≥ V_{IH}(min), CLK ≤ V_{IL}(max), t_{CC} = ∞ Input signals are stable		1	mA	
Active Standby Current in power-down mode	$I_{CC3}P$	CKE ≤ V_{IL}(max), t_{CC} = 10ns		5	mA	
	$I_{CC3}PS$	CKE & CLK ≤ V_{IL}(max), t_{CC} = ∞		1	mA	
Active Standby Current in non power-down mode (One Bank Active)	$I_{CC3}N$	CKE ≥ V_{IH}(min), CS ≥ V_{IH}(min), t_{CC} = 10ns Input signals are changed one time during 20ns		20	mA	
	$I_{CC3}NS$	CKE ≥ V_{IH}(min), CLK ≤ V_{IL}(max), t_{CC} = ∞ Input signals are stable		5	mA	
Operating Current (Burst Mode)	$I_{CC}4$	IO = 0 mA Page burst 4Banks Activated t_{CCD} = 2CLKs		50	mA	1
Refresh Current	$I_{CC}5$	$t_{ARFC} \geq t_{ARFC}$(min)		85	mA	2
Self Refresh Current	$I_{CC}6$	CKE ≤ 0.2V	**Internal TCSR**	**Max 40**	**°C**	
			Full Array	100	µA	
			1/2 of full array	90		
			1/4 of full array	85		

Notes:

1. *Measured with outputs open.*
2. *Refresh period is 64ms.*
3. *Unless otherwise noted, input swing IeveI is CMOS(V_{IH} /V_{IL}=V_{DDQ}/V_{SSQ}).*

54 AC Operating Test Conditions

VDD = 1.8V ± 0.15V, TA = -25°C ~ +85°C

Parameter	Value	Unit
AC input levels (V_{IH}/V_{IL})	0.9 x V_{DDQ} / 0.2	V
Input timing measurement reference level	0.5 x V_{DDQ}	
Input rise and fall time	tr/tf = 1/1	ns
Output timing measurement reference level	0.5 x V_{DDQ}	V
Output load condition	See Figure 54.2	

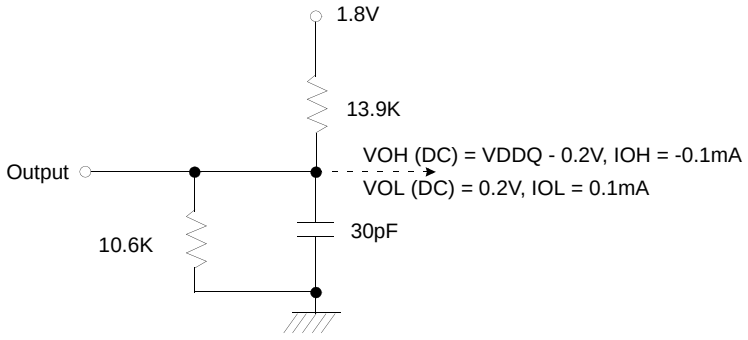


Figure 54.1 DC Output Load Circuit

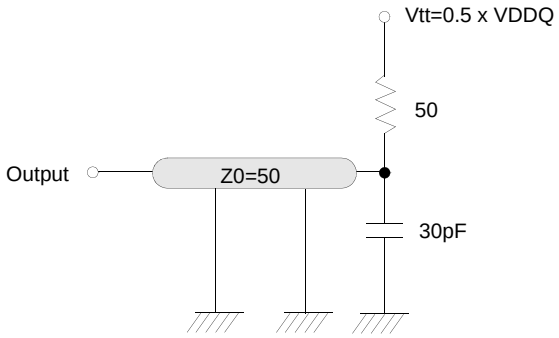


Figure 54.2 AC Output Load Circuit

55 Operating AC Parameter

(AC operating conditions unless otherwise noted)

Parameter	Symbol	Value	Unit	Note
Row active to row active delay	t_{RRD}(min)	19	ns	1
RAS# to CAS# delay	t_{RCD}(min)	28.5		
Row precharge time	t_{RP}(min)	28.5		
Row active time	t_{RAS}(min)	60		
	t_{RAS}(max)	100	µs	
Row cycle time	t_{RC}(min)	88.5	ns	1
Last data in to row precharge	t_{RDL}(min)	15	ns	2
Last data in to Active delay	t_{DAL}(min)	t_{RDL} + t_{RP}	—	
Last data in to new col. address delay	t_{CDL}(min)	1	CLK	2
Last data in to burst stop	t_{BDL}(min)	1	CLK	
Auto refresh cycle time	t_{ARFC}(min)	80	ns	
Exit self refresh to active command	t_{SRFX}(min)	120	ns	
Col. address to col. address delay	t_{CCD}(min)	1	CLK	3
Number of valid output data	CAS latency=3	2	ea	4
Number of valid output data	CAS latency=2	1		
Number of valid output data	CAS latency=1	0		

Notes:

1. *The minimum number of clock cycles is determined by dividing the minimum time required with clock cycle time and then rounding off to the next higher integer.*
2. *Minimum delay is required to complete write.*
3. *All parts allow every cycle column address change.*
4. *In case of row precharge interrupt, auto precharge and read burst stop.*

56 AC Characteristics

(AC operating conditions unless otherwise noted)

Parameter		Symbol	Min	Max	Unit	Note
CLK cycle time	CAS latency=3	t_{CC}	9.5	1000	ns	1
CLK cycle time	CAS latency=2		15			
CLK cycle time	CAS latency=1		25			
CLK to valid output delay	CAS latency=3	t_{SAC}		7	ns	1, 2
CLK to valid output delay	CAS latency=2			8		
CLK to valid output delay	CAS latency=1			20		
Output data hold time	CAS latency=3	t_{OH}	2.5		ns	2
Output data hold time	CAS latency=2					
Output data hold time	CAS latency=1					
CLK high pulse width		t_{CH}	3.5		ns	3
CLK low pulse width		t_{CL}				3
Input setup time		t_{SS}	3.0			3
Input hold time		t_{SH}	1.5			3
CLK to output in Low-Z		t_{SLZ}	1			2
CLK to output in Hi-Z	CAS latency=3	t_{SHZ}		7		
	CAS latency=2			8		
	CAS latency=1			20		

Notes:

1. *Parameters depend on programmed CAS latency.*
2. *If clock rising time is longer than 1ns, (tr/2-0.5)ns should be added to the parameter.*
3. *Assumed input rise and fall time (tr & tf) = 1ns.*
   If tr & tf is longer than 1ns, transient time compensation should be considered, i.e., [(tr + tf)/2-1]ns should be added to the parameter.

57 Simplified Truth Table

Command			CKEn-1	CKEn	CS#	RAS#	CAS#	WE#	DQM	BA0,1	A10/AP	A11, A9 ~ A0	Note
Register	Mode Register Set		H	X	L	L	L	L	X	OP CODE			1, 2
Refresh	Auto Refresh		H	H	L	L	L	H	X	X			3
	Self Refresh	Entry		L									3
		Exit	L	H	L	H	H	H	X	X			3
					H	X	X	X					3
Bank Active & Row Addr.			H	X	L	L	H	H	X	V	Row Address		
Read & Column Address	Auto Precharge Disable		H	X	L	H	L	H	X	V	L	Column Address (A0~A8)	4
	Auto Precharge Enable										H		4, 5
Write & Column Address	Auto Precharge Disable		H	X	L	H	L	L	X	V	L	Column Address (A0~A8)	4
	Auto Precharge Enable										H		4, 5
Burst Stop			H	X	L	H	H	L	X	X			6
Precharge	Bank Selection		H	X	L	L	H	L	X	V	L	X	
	All Banks									X	H		
Clock Suspend or Active Power Down		Entry	H	L	H	X	X	X	X	X			
					L	V	V	V					
		Exit	L	H	X	X	X	X	X				
Precharge Power Down Mode		Entry	H	L	H	X	X	X	X	X			
					L	H	H	H					
		Exit	L	H	H	X	X	X	X				
					L	V	V	V					
DQM			H	X					V	X			7
No Operation Command			H	X	H	X	X	X	X	X			
					L	H	H	H					

Legend: *V = Valid, X = Don't Care, H = Logic High, L = Logic Low*

Notes:

1. *OP Code: Operand Code*
   A0 ~ A11 & BA0 ~ BA1: Program keys. (@MRS)
2. *MRS can be issued only at all banks precharge state.*
   A new command can be issued after 2 CLK cycles of MRS.
3. *Auto refresh functions are the same as CBR refresh of DRAM.*
   The automatic precharge without row precharge command is meant by "Auto".
   Auto/self refresh can be issued only at all banks precharge state.
   Partial self refresh can be issued only after setting partial self refresh mode of EMRS.
4. *BA0 ~ BA1: Bank select addresses.*
5. *During burst read or write with auto precharge, new read/write command can not be issued.*
   Another bank read/write command can be issued after the end of burst.
   New row active of the associated bank can be issued at tRP after the end of burst.
6. *Burst stop command is valid at every burst length.*
7. *DQM sampled at the positive going edge of CLK masks the data-in at that same CLK in write operation (Write DQM latency is 0), but in read operation, it makes the data-out Hi-Z state after 2 CLK cycles. (Read DQM latency is 2).*

58 Mode Register Field Table to Program Modes

Address	BA0 ~ BA1	A11 ~ A10/AP	A9 (Note 2)	A8	A7	A6	A5	A4	A3	A2	A1	A0
Function	"0" Setting for Normal MRS	RFU	W.B.L	Test Mode		CAS Latency			BT	Burst Length		

59 Normal MRS Mode

Test Mode			CAS Latency				Burst Type			Burst Length				
A8	A7	Type	A6	A5	A4	Latency	A3	Type		A2	AI	A0	BT=0	BT=I
0	0	Mode Register Set	0	0	0	Reserved	0	Sequential		0	0	0	1	1
0	1	Reserved	0	0	1	1	1	Interleave		0	0	1	2	2
1	0	Reserved	0	1	0	2	Mode Select			0	1	0	4	4
1	1	Reserved	0	1	1	3	BAI	BA0	Mode	0	1	1	8	8
Write Burst Length			1	0	0	Reserved	0	0	Setting for Normal MRS	1	0	0	Reserved	Reserved
A9	Length		1	0	1	Reserved				1	0	1	Reserved	Reserved
0	Burst		1	1	0	Reserved				1	1	0	Reserved	Reserved
1	Single Bit		1	1	1	Reserved				1	1	1	Full Page	Reserved

Note: *Full Page Length x 16: 64Mb(256), 128Mb(512), 256Mb(512), 512Mb(1024)*

Register Programmed with Extended MRS

Address	BAI	BA0	AII ~ AI0/AP	A9	A8	A7	A6	A5	A4	A3	A2	AI	A0
Function	Mode Select		RFU (Note 1)				DS		RFU (Note 1)		PASR		

60 EMRS for PASR (Partial Array Self Ref) & DS (Driver Strength)

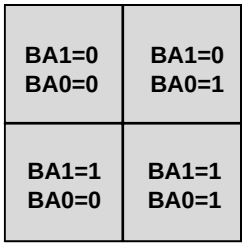
Mode Select			Driver Strength			PASR			
BAI	BA0	Mode	A6	A5	Driver Strength	A2	AI	A0	Refreshed Area
0	0	Normal MRS	0	0	Full	0	0	0	Full Array
0	1	Reserved	0	1	1/2	0	0	1	1/2 of full array
1	0	EMRS for Mobile SDRAM	1	0	1/4	0	1	0	1/4 of full array
1	1	Reserved	1	1	1/8	0	1	1	Reserved
Reserved Address						1	0	0	Reserved
AII~AI0/AP	A9	A8	A7	A4	A3	1	0	1	Reserved
0	0	0	0	0	0	1	1	0	Reserved
						1	1	1	Reserved

Notes:

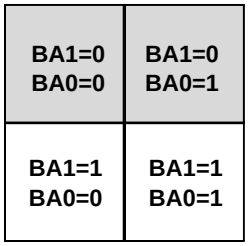
1. *RFU(Reserved for future use) should stay "0" during MRS cycle.*
2. *If A9 is high during MRS cycle, "Burst Read Single Bit Write" function will be enabled.*

6I Partial Array Self Refresh

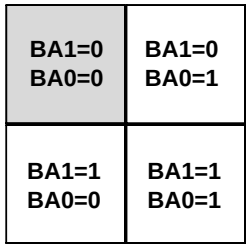
1. In order to save power consumption, Mobile SDRAM has PASR option.
2. Mobile SDRAM supports 3 kinds of PASR in self refresh mode: Full Array, 1/2 of full array, and 1/4 of full array.



- Full Array



- 1/2 of full Array



- 1/4 of full Array

Partial Self Refresh Area

62 Internal Temperature Compensated Self Refresh (TCSR)

Temperature Range	Self Refresh Current (Icc 6)			Unit
	Full Array	I/2 of full array	I/4 of full array	
Max. 40°C	100	90	85	µA
Max. 85°C	250	210	190	

Notes:

1. *In order to save power consumption, Mobile-SDRAM includes the internal temperature sensor and control units to control the self refresh cycle automatically according to the two temperature ranges; Max. 40°C, Max. 85°C.*
2. *If the EMRS for external TCSR is issued by the controller, this EMRS code for TCSR is ignored.*

63 Power Up Sequence

1. Apply power and attempt to maintain CKE at a high state and all other inputs may be undefined.
   - Apply VDD before or at the same time as VDDQ.
2. Maintain stable power, stable clock and NOP input condition for a minimum of 200us.
3. Issue precharge commands for all banks of the devices.
4. Issue 2 or more auto-refresh commands.
5. Issue a mode register set command to initialize the mode register.
6. Issue a extended mode register set command to define DS or PASR operating type of the device after normal MRS.

EMRS cycle is not mandatory and the EMRS command needs to be issued only when DS or PASR is used.

The default state without EMRS command issued is half driver strength and all full array refreshed.

The device is now ready for the operation selected by EMRS.

For operating with DS or PASR, set DS or PASR mode in EMRS setting stage.

In order to adjust another mode in the state of DS or PASR mode, additional EMRS set is required but power up sequence is not.

needed again at this time. In that case, all banks have to be in idle state prior to adjusting EMRS set.

64 Burst Sequence

64.1 Burst Length = 4

Initial Address		Sequential				Interleave		
AI	A0							
0	0	0	1	2	3	0	1	3
0	1	1	2	3	0	1	0	2
1	0	2	3	0	1	2	3	1
1	1	3	0	1	2	3	2	0

64.2 Burst Length = 8

Initial Address			Sequential								Interleave							
A2	AI	A0																
0	0	0	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
0	0	1	1	2	3	4	5	6	7	0	1	0	3	2	5	4	7	6
0	1	0	2	3	4	5	6	7	0	1	2	3	0	1	6	7	4	5
0	1	1	3	4	5	6	7	0	1	2	3	2	1	0	7	6	5	4
1	0	0	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3
1	0	1	5	6	7	0	1	2	3	4	5	4	7	6	1	0	2	3
1	1	0	6	7	0	1	2	3	4	5	6	7	4	5	2	3	0	1
1	1	1	7	0	1	2	3	4	5	6	7	6	5	4	3	2	1	0

65 Device Operations

65.I Addresses of 64Mb

65.1.1 Bank Addresses (BA0 ~ BA1)

:In case x 16. This SDRAM is organized as four independent banks of 1,048,576 words x 16 bits memory arrays. The BA0 ~ BA1 inputs are latched at the time of assertion of RAS# and CAS# to select the bank to be used for the operation. The bank addresses BA0 ~ BA1 are latched at bank active, read, write, mode register set and precharge operations.

: In case x 32. This SDRAM is organized as four independent banks of 524,288 words x 32 bits memory arrays. The BA0 ~ BA1 inputs are latched at the time of assertion of RAS# and CAS# to select the bank to be used for the operation. The bank addresses BA0 ~ BA1 are latched at bank active, read, write, mode register set and precharge operations.

65.1.2 Address Inputs (A0 ~ A11)

: In case x 16. The 20 address bits are required to decode the 1,048,576 word locations are multiplexed into 12 address input pins (A0 ~ A11). The 12 bit row addresses are latched along with RAS# and BA0 ~ BA1 during bank activate command. The 8 bit column addresses are latched along with CAS#, WE# and BA0 ~ BA1 during read or write command.

: In case x 32. The 19 address bits are required to decode the 524,288 word locations are multiplexed into 11 address input pins (A0 ~ A10). The 11 bit row addresses are latched along with RAS# and BA0 ~ BA1 during bank activate command. The 8 bit column addresses are latched along with CAS#, WE# and BA0 ~ BA1 during read or write command.

65.2 Addresses of I28Mb

65.2.1 Bank Addresses (BA0 ~ BA1)

:In case x 16. This SDRAM is organized as four independent banks of 2,097,152 words x 16 bits memory arrays. The BA0 ~ BA1 inputs are latched at the time of assertion of RAS# and CAS# to select the bank to be used for the operation. The bank addresses BA0 ~ BA1 are latched at bank active, read, write, mode register set and precharge operations.

:In case x 32. This SDRAM is organized as four independent banks of 1,048,576 words x 32 bits memory arrays. The BA0 ~ BA1 inputs are latched at the time of assertion of RAS# and CAS# to select the bank to be used for the operation. The bank addresses BA0 ~ BA1 are latched at bank active, read, write, mode register set and precharge operations.

65.2.2 Address Inputs (A0 ~ A11)

: In case x 16. The 21 address bits are required to decode the 2,097,152 word locations are multiplexed into 12 address input pins (A0 ~ A11). The 12 bit row addresses are latched along with RAS# and BA0 ~ BA1 during bank activate command. The 9 bit column addresses are latched along with CAS#, WE# and BA0 ~ BA1 during read or write command.

:In case x 32. The 20 address bits are required to decode the 1,048,576 word locations are multiplexed into 12 address input pins (A0 ~ A11). The 12 bit row addresses are latched along with RAS# and BA0 ~ BA1 during bank activate command. The 8 bit column addresses are latched along with CAS#, WE# and BA0 ~ BA1 during read or write command.

65.3 Clock

The clock input is used as the reference for all SDRAM operations. All operations are synchronized to the positive going edge of the clock. The clock transitions must be monotonic between V_{IL} and V_{IH}. During operation with CKE high all inputs are assumed to be in a valid state (low or high) for the duration of set-up and hold time around positive edge of the clock in order to function well Q perform and ICC specifications

65.4 Clock Enable (CKE)

The clock enable(CKE) gates the clock onto SDRAM. If CKE goes low synchronously with clock (set-up and hold time are the same as other inputs), the internal clock is suspended from the next clock cycle and the state of output and burst address is frozen as long as the CKE remains low. All other inputs are ignored from the next clock cycle after CKE goes low. When all banks are in the idle state and CKE goes low synchronously with clock, the SDRAM enters the power down mode from the next clock cycle. The SDRAM remains in the power down mode ignoring the other inputs as long as CKE remains low. The power down exit is synchronous as the internal clock is suspended. When CKE goes high at least "1CLK + t_{SS}"before the high going edge of the clock, then the SDRAM becomes active from the same clock edge accepting all the input commands.

65.5 NOP and Device Deselect

When RAS#, CAS# and WE# are high, the SDRAM performs no operation (NOP). NOP does not initiate any new operation, but is needed to complete operations which require more than single clock cycle like bank activate, burst read, auto refresh, etc. The device deselect is also a NOP and is entered by asserting CS# high. CS# high disables the command decoder so that RAS#, CAS#, WE# and all the address inputs are ignored.

65.6 DQM Operation

The DQM is used to mask input and output operations. It works similar to OE# during read operation and inhibits writing during write operation. The read latency is two cycles from DQM and zero cycle for write, which means DQM masking occurs two cycles later in read cycle and occurs in the same cycle during write cycle. DQM operation is synchronous with the clock. The DQM signal is important during burst interruptions of write with read or precharge in the SDRAM. Due to asynchronous nature of the internal write, the DQM operation is critical to avoid unwanted or incomplete writes when the complete burst write is not required. Please refer to DQM timing diagram also.

65.7 Mode Register Set (MRS)

The mode register stores the data for controlling the various operating modes of SDRAM. It programs the CAS latency, burst type, burst length, test mode and various vendor specific options to make SDRAM useful for variety of different applications. The default value of the mode register is not defined, therefore the mode register must be written after power up to operate the SDRAM. The mode register is written by asserting low on CS#, RAS#, CAS# and WE# (The SDRAM should

be in active mode with CKE already high prior to writing the mode register). The state of address pins A0 ~ An and BA0 ~ BA1 in the same cycle as CS#, RAS#, CAS# and WE# going low is the data written in the mode register. Two clock cycles is required to complete the write in the mode register. The mode register contents can be changed using the same command and clock cycle requirements during operation as long as all banks are in the idle state. The mode register is divided into various fields depending on the fields of functions. The burst length field uses A0 ~ A2, burst type uses A3, CAS latency (read latency from column address) use A4 ~ A6, vendor specific options or test mode use A7 ~ A8, A10/AP ~ An and BA0 ~ BA1. The write burst length is programmed using A9. A7 ~ A8, A10/AP ~ An and BA0 ~ BA1 must be set to low for normal SDRAM operation. Refer to the table for specific codes for various burst length, burst type and CAS latencies.

65.8 Extended Mode Register Set (EMRS)

The extended mode register stores the data for selecting driver strength and partial self refresh. EMRS cycle is not mandatory and the EMRS command needs to be issued only when DS or PASR is used. The default state without EMRS command issued is half driver strength, and full array refreshed. The extended mode register is written by asserting low on CS#, RAS#, CAS#, WE# and high on BA1,low on BA0(The SDRAM should be in all bank precharge with CKE already high prior to writing into the extended mode register). The state of address pins A0 ~ A11 in the same cycle as CS#, RAS#, CAS# and WE# going low is written in the extended mode register. Two clock cycles are required to complete the write operation in the extended mode register. The mode register contents can be changed using the same command and clock cycle requirements during operation as long as all banks are in the idle state. A0 - A2 are used for partial self refresh, A5 - A6 are used for Driver strength, "Low" on BA1 and "High" on BA0 are used for EMRS. All the other address pins except A0~A2, A5~A6, BA1, BA0 must be set to low for proper EMRS operation. Refer to the table for specific codes.

65.9 Bank Activate

The bank activate command is used to select a random row in an idle bank. By asserting low on RAS# and CS# with desired row and bank address, a row access is initiated. The read or write operation can occur after a time delay of t_{RCD}(min) from the time of bank activation. t_{RCD} is an internal timing parameter of SDRAM, therefore it is dependent on operating clock frequency. The minimum number of clock cycles required between bank activate and read or write command should be calculated by dividing t_{RCD}(min) with cycle time of the clock and then rounding off the result to the next higher integer.

The SDRAM has four internal banks in the same chip and shares part of the internal circuitry to reduce chip area, therefore it restricts the activation of four banks simultaneously. Also the noise generated during sensing of each bank of SDRAM is high, requiring some time for power supplies to recover before another bank can be sensed reliably. t_{RRD}(min) specifies the minimum time required between activating different bank. The number of clock cycles required between different bank activation must be calculated similar to t_{RCD} specification. The minimum time required for the bank to be active to initiate sensing and restoring the complete row of dynamic cells is determined by t_{RAS}(min). Every SDRAM bank activate command must satisfy t_{RAS}(min) specification before a precharge command to that active bank can be asserted. The maximum time any bank can be in the active state is determined by t_{RAS}(max). The number of cycles for both t_{RAS}(min) and t_{RAS}(max) can be calculated similar to t_{RCD} specification.

65.10 Burst Read

The burst read command is used to access burst of data on consecutive clock cycles from an active row in an active bank. The burst read command is issued by asserting low on CS# and CAS# with WE# being high on the positive edge of the clock. The bank must be active for at least t_{RCD}(min) before the burst read command is issued. The first output appears in CAS# latency

number of clock cycles after the issue of burst read command. The burst length, burst sequence and latency from the burst read command is determined by the mode register which is already programmed. The burst read can be initiated on any column address of the active row. The address wraps around if the initial address does not start from a boundary such that number of outputs from each I/O are equal to the burst length programmed in the mode register. The output goes into high-impedance at the end of the burst, unless a new burst read was initiated to keep the data output gapless. The burst read can be terminated by issuing another burst read or burst write in the same bank or the other active bank or a precharge command to the same bank. The burst stop command is valid at every page burst length.

65.11 Burst Write

The burst write command is similar to burst read command and is used to write data into the SDRAM on consecutive clock cycles in adjacent addresses depending on burst length and burst sequence. By asserting low on CS#, CAS# and WE# with valid column address, a write burst is initiated. The data inputs are provided for the initial address in the same clock cycle as the burst write command. The input buffer is deselected at the end of the burst length, even though the internal writing can be completed yet. The writing can be completed by issuing a burst read and DQM for blocking data inputs or burst write in the same or another active bank. The burst stop command is valid at every burst length. The write burst can also be terminated by using DQM for blocking data and procreating the bank tRDL after the last data input to be written into the active row. See DQM OPERATION also.

65.12 All Banks Precharge

All banks can be precharged at the same time by using Precharge all command. Asserting low on CS#, RAS#, and WE# with high on A10/AP after all banks have satisfied t_{RAS}(min) requirement, performs precharge on all banks. At the end of t_{RP} after performing precharge to all the banks, all banks are in idle state.

65.13 Precharge

The precharge operation is performed on an active bank by asserting low on CS#, RAS#, WE# and A10/AP with valid BA0 ~ BA1 of the bank to be precharged. The precharge command can be asserted anytime after t_{RAS}(min) is satisfied from the bank active command in the desired bank. t_{RP} is defined as the minimum number of clock cycles required to complete row precharge is calculated by dividing t_{RP} with clock cycle time and rounding up to the next higher integer. Care should be taken to make sure that burst write is completed or DQM is used to inhibit writing before precharge command is asserted. The maximum time any bank can be active is specified by t_{RAS}(max). Therefore, each bank activate command. At the end of precharge, the bank enters the idle state and is ready to be activated again. Entry to Power down, Auto refresh, Self refresh and Mode register set etc. is possible only when all banks are in idle state.

65.14 Auto Precharge

The precharge operation can also be performed by using auto precharge. The SDRAM internally generates the timing to satisfy t_{RAS}(min) and "t_{RP}" for the programmed burst length and CAS latency. The auto precharge command is issued at the same time as burst read or burst write by asserting high on A10/AP. If burst read or burst write by asserting high on A10/AP, the bank is left active until a new command is asserted. Once auto precharge command is given, no new commands are possible to that particular bank until the bank achieves idle state.

65.15 Auto Refresh

The storage cells of 64Mb, 128Mb, 256Mb and 512Mb SDRAM need to be refreshed every 64ms to maintain data. An auto refresh cycle accomplishes refresh of a single row of storage cells. The internal counter increments automatically on every auto refresh cycle to refresh all the rows. An

auto refresh command is issued by asserting low on CS#, RAS# and CAS# with high on CKE and WE#. The auto refresh command can only be asserted with all banks being in idle state and the device is not in power down mode (CKE is high in the previous cycle). The time required to complete the auto refresh operation is specified by t_{RC}(min). The minimum number of clock cycles required can be calculated by driving t_{ARFC} with clock cycle time and them rounding up to the next higher integer. The auto refresh command must be followed by NOP's until the auto refresh operation is completed. All banks will be in the idle state at the end of auto refresh operation. The auto refresh is the preferred refresh mode when the SDRAM is being used for normal data transactions. The 64Mb and 128Mb SDRAM's auto refresh cycle can be performed once in 15.6µs or a burst of 4096 auto refresh cycles once in 64ms. The 256Mb and 512Mb SDRAM's auto refresh cycle can be performed once in 7.8µs or a burst of 8192 auto refresh cycles once in 64ms.

65.16 Self Refresh

The self refresh is another refresh mode available in the SDRAM. The self refresh is the preferred refresh mode for data retention and low power operation of SDRAM. In self refresh mode, the SDRAM disables the internal clock and all the input buffers except CKE. The refresh addressing and timing are internally generated to reduce power consumption.

The self refresh mode is entered from all banks idle state by asserting low on CS#, RAS#, CAS# and CKE with high on WE#. Once the self refresh mode is entered, only CKE state being low matters, all the other inputs including the clock are ignored in order to remain in the self refresh mode.

The self refresh is exited by restarting the external clock and then asserting high on CKE. This must be followed by NOP's for a minimum time of t_{SRFX} before the SDRAM reaches idle state to begin normal operation. In case that the system uses burst auto refresh during normal operation, it is recommended to use burst 8192 auto refresh cycles for 256Mb and 512Mb, and burst 4096 auto refresh cycles for 128Mb and 64Mb immediately before entering self refresh mode and after exiting in self refresh mode. On the other hand, if the system uses the distributed auto refresh, the system only has to keep the refresh duty cycle.

65.17 Basic Feature and Function Descriptions

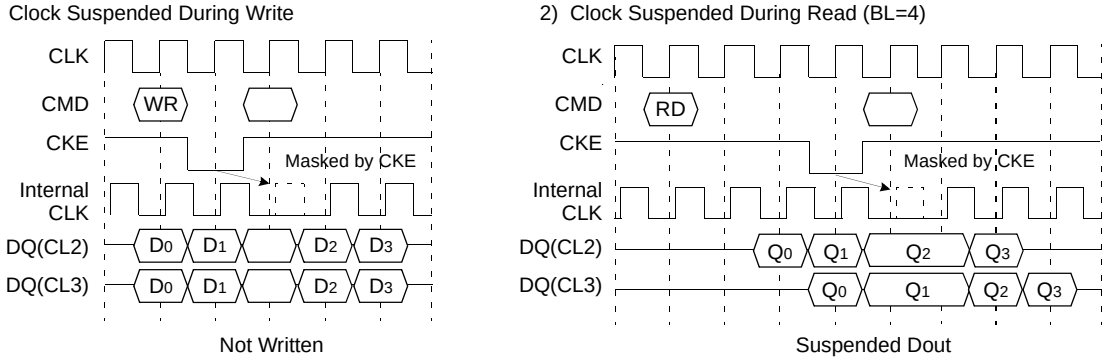
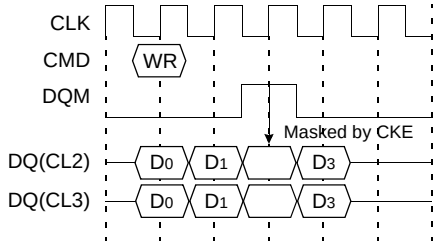


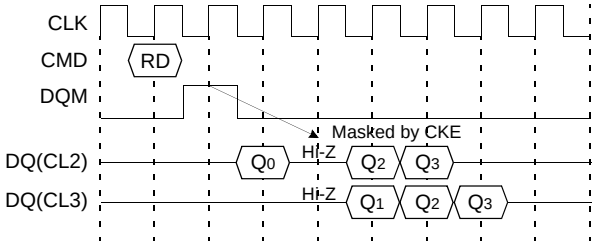
Figure 65.1 Clock Suspend

1) Write Mask (BL=4)



DQM to Data-in Mask = 0

2) Read Mask (BL=4)



DQM to Data-out Mask = 2

3) DQM with Clock Suspended (Full Page Read) -Note 2

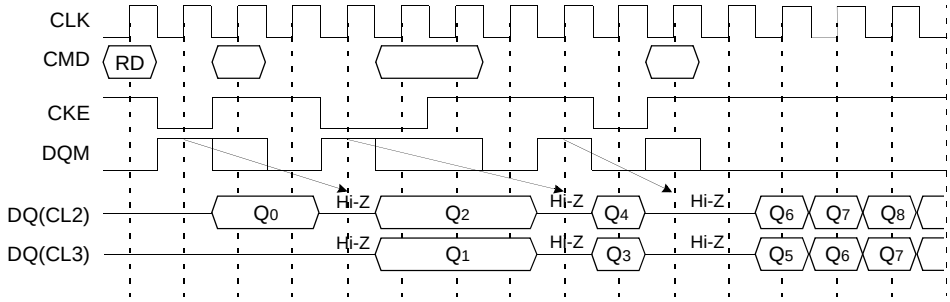
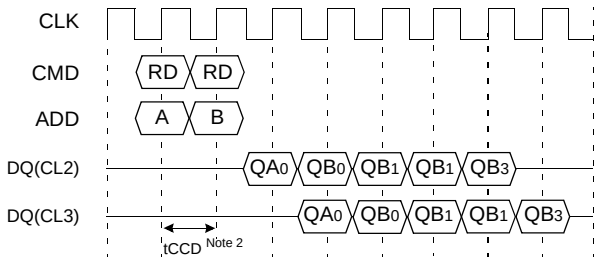


Figure 65.2 DQM Operation

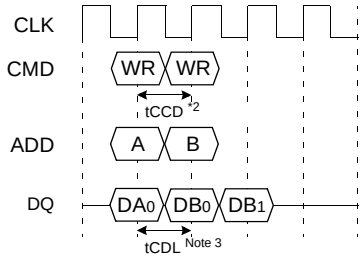
Notes:

1. *CKE to CLK disable/enable = 1CLK.*
2. *DQM makes data out Hi-Z after 2CLKs which should masked by CKE "L"*
3. *DQM masks both data-in and data-out.*

1) Read interrupted by Read (BL=4) -Note 1



2) Write interrupted by Write (BL=2)



3) Write interrupted by Read (BL=2)

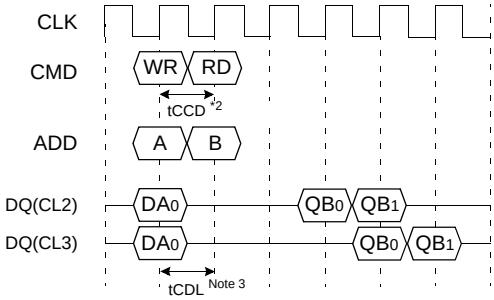


Figure 65.3 CAS# Interrupt (I)

Notes:

1. *By "Interrupt", it is meant to stop burst read/write by external command before the end of burst. By "CAS# Interrupt", to stop burst read/write by CAS# access; read and write.*
2. *t_{CCD}: CAS# to CAS# delay. (=1CLK).*
3. *t_{CDL}: Last data in to new column address delay. (=1CLK).*

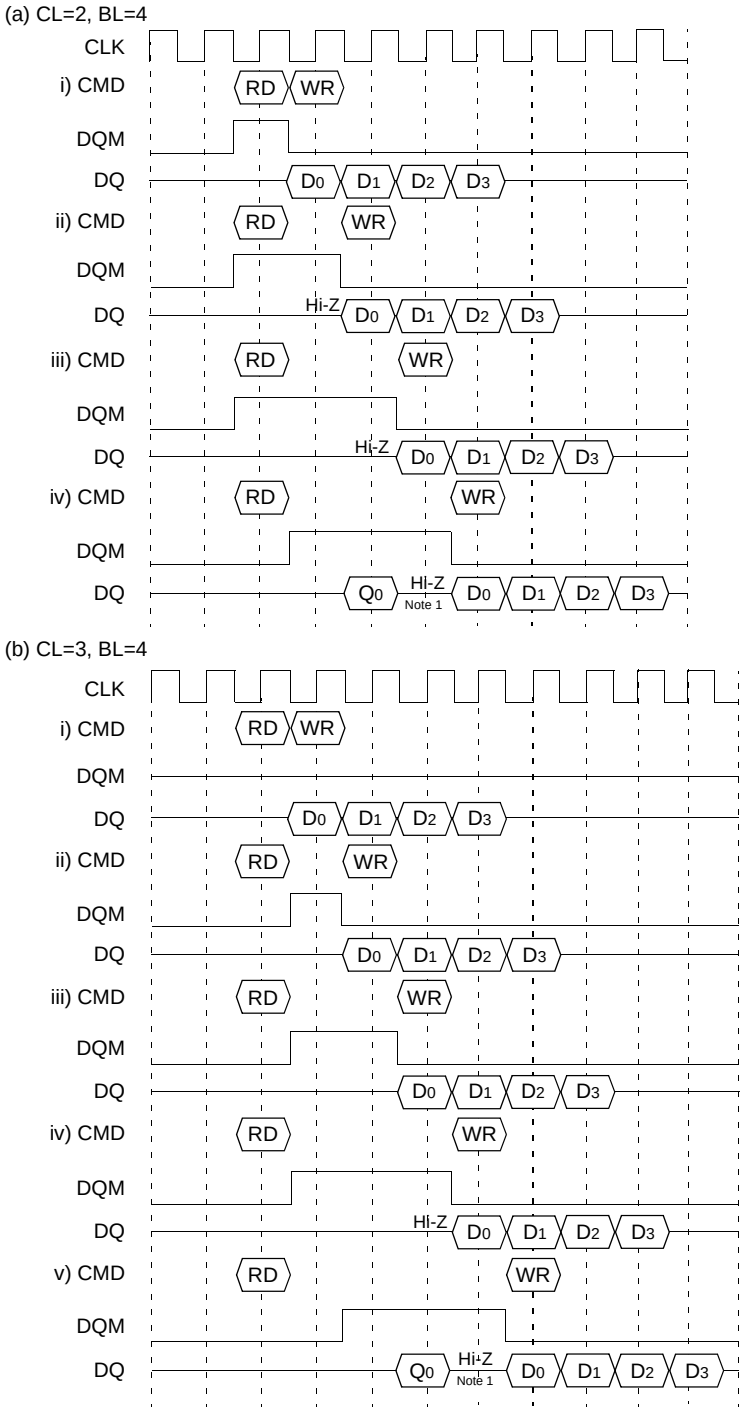


Figure 65.4 CAS# Interrupt (2): Read Interrupted by Write & DQM

Notes:

1. To prevent bus contention, there should be at least one gap between data in and data out.

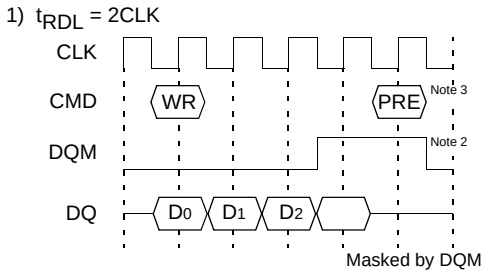


Figure 65.5 Write Interrupted by Precharge & DQM

Notes:

1. *To prevent bus contention, DQM should be issued which makes at least one gap between data in and data out.*
2. *To inhibit invalid write, DQM should be issued.*
3. *This precharge command and burst write command should be of the same bank, otherwise it is not precharge interrupt but only another bank precharge of four banks operation.*

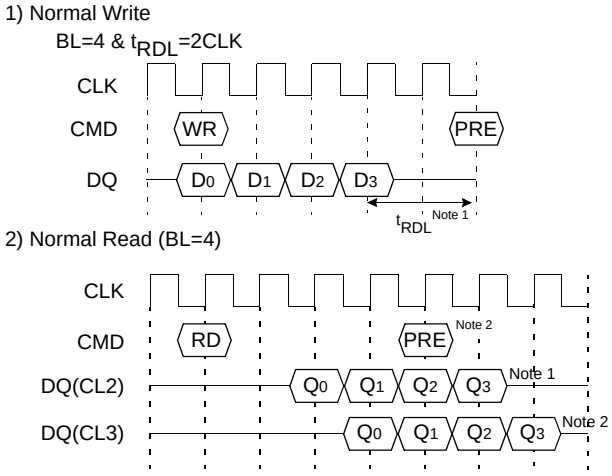


Figure 65.6 Precharge

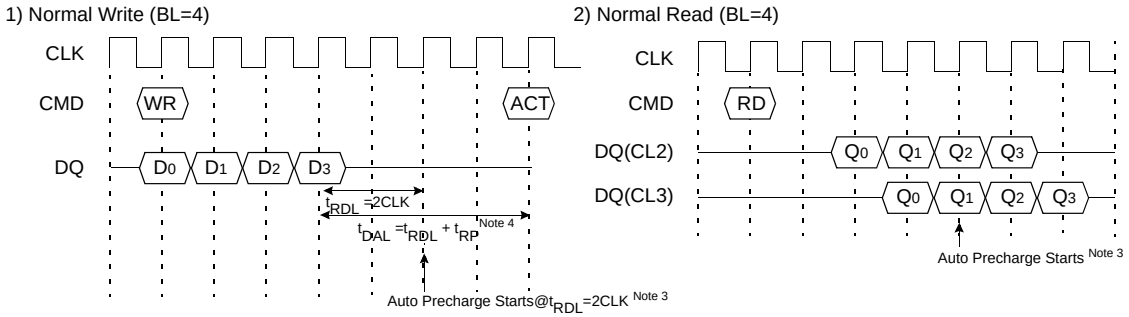


Figure 65.7 Auto Precharge

Notes:

1. *t_{RDL}=2 CLK can be supported.*
2. *Number of valid output data after row precharge: 1, 2 for CAS Latency = 2, 3 respectively.*
3. *The row active command of the precharge bank can be issued after t_{RP} from this point. The new read/write command of other activated bank can be issued from this point. At burst read/write with auto precharge, CAS interrupt of the same bank is illegal*
4. *t_{DAL} defined Last data in to Active delay. t_{DAL}=t_{RDL}+ t_{RP} can be supported.*

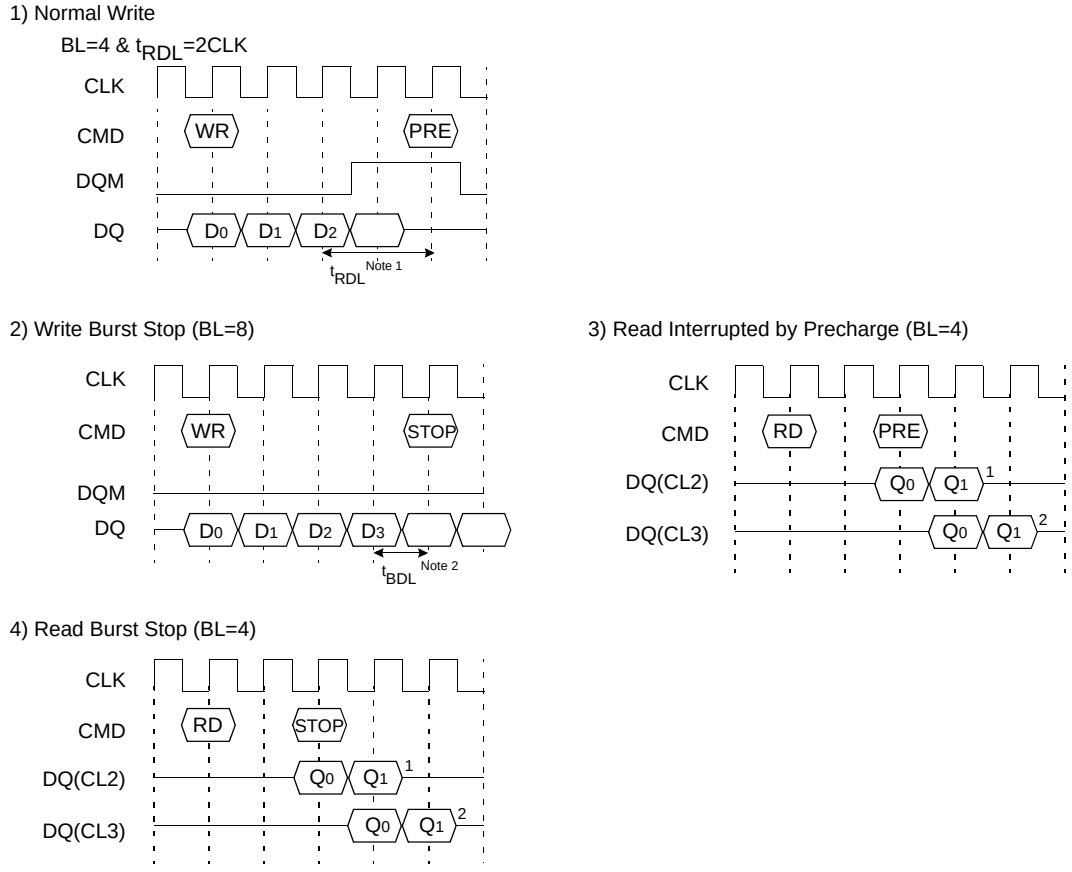


Figure 65.8 Burst Stop and Interrupted by Precharge

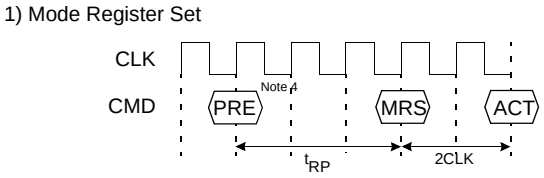


Figure 65.9 MRS

Notes:

1. *t_{RDL}=2 CLK can be supported.*
2. *t_{BDL}: 1 CLK; Last data in to burst stop delay. Read or write burst stop command is valid at every burst length.*
3. *Number of valid output data after row precharge or burst stop: 1, 2 for CAS latency= 2, 3 respectively.*
4. *PRE: All banks precharge is necessary. MRS can be issued only at all banks precharge state.*

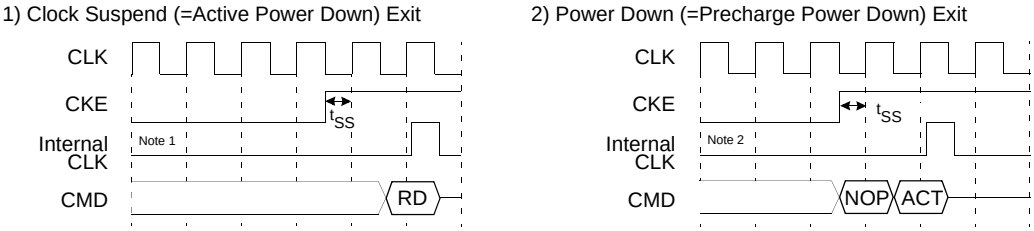


Figure 65.10 Clock Suspend Exit and Power Down Exit

65.17.1 Auto Refresh

An auto refresh command is issued by having CS#, RAS# and CAS# held low with CKE and WE# high at the rising edge of the clock (CLK). All banks must be precharged and idle for $t_{RP(min)}$ before the auto refresh command is applied. No control of the external address pins is required once this cycle has started because of the internal address counter. When the refresh cycle has completed, all banks will be in the idle state. A delay between the auto refresh command and the next activate command or subsequent auto refresh command must be greater than or equal to the t_{ARFC}(min).

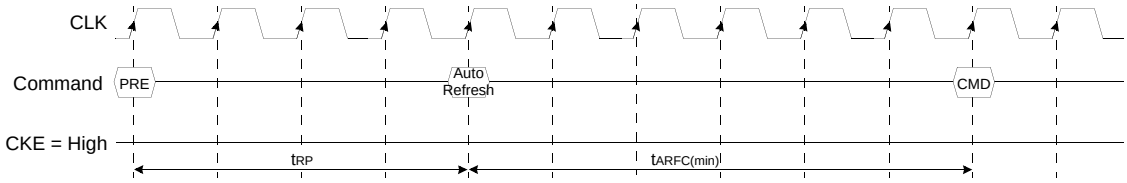


Figure 65.11 Auto Refresh

65.17.2 Self Refresh

A Self Refresh command is defined by having CS#, RAS#, CAS# and CKE held low with WE# high at the rising edge of the clock. Once the self Refresh command is initiated, CKE must be held low to keep the device in Self Refresh mode. After 1 clock cycle from the self refresh command, all of the external control signals including system clock (CLK) can be disabled except CKE. The clock is internally disabled during Self Refresh operation to reduce power. To exit the Self Refresh mode, supply stable clock input before returning CKE high, assert deselect or NOP command and then assert CKE high. In case that the system uses burst auto refresh during normal operation, it is recommended to use burst 4096 auto refresh cycle immediately before entering self refresh mode and after exiting in self refresh mode. On the other hand, if the system uses the distributed auto refresh, the system only has to keep the refresh duty cycle.

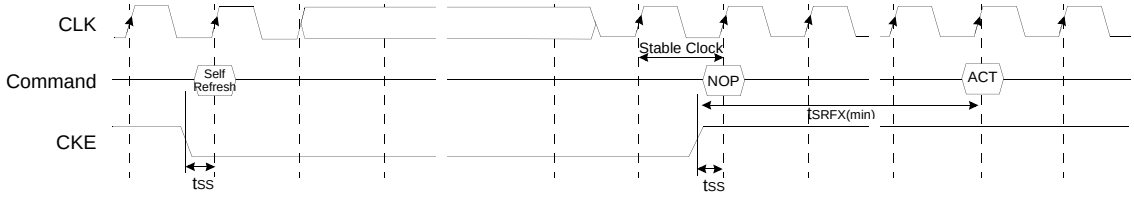


Figure 65.12 Self Refresh

66 About Burst Type Control

Basic Mode	Sequential Counting	At MRS A3 = "0". See the Burst Sequence. (BL=4, 8) BL=1, 2, 4, 8 and full page.
	Interleave Counting	At MRS A3 = "1". See the Burst Sequence. (BL=4, 8) BL=4, 8. At BL=1, 2 Interleave Counting = Sequential Counting.
Random Mode	Random column Access t_{CCD} = 1 CLK	Every cycle Read/Write Command with random column address can realize Random Column Access. That is similar to Extended Data Out (EDO) Operation of conventional DRAM.

67 About Burst Length Control

Basic Mode	1	At MRS A2,1,0 = "000". At auto precharge, tRAS should not be violated.
	2	At MRS A2,1,0 = "001". At auto precharge, tRAS should not be violated.
	4	At MRS A2,1,0 = "010".
	8	At MRS A2,1,0 = "011".
	Full Page	At MRS A2,1,0 = "111". Wrap around mode (infinite burst length) should be stopped by burst stop. RAS# interrupt or CAS# interrupt.
Special Mode	BRSW	At MRS A9 = "1". Read burst =1, 2, 4, 8, full page write Burst =1. At auto precharge of write, tRAS should not be violated.
Random Mode	Burst Stop	t_{BDL}= 1, Valid DQ after burst stop is 1, 2 for CAS latency 2, 3 respectively Using burst stop command, any burst length control is possible.
Interrupt Mode	RAS# Interrupt (Interrupted by Precharge)	Before the end of burst, Row precharge command of the same bank stops read/write burst with Row precharge. t_{RDL}= 2 with DQM, valid DQ after burst stop is 1, 2 for CAS latency 2, 3 respectively. During read/write burst with auto precharge, RAS# interrupt can not be issued.
	CAS# Interrupt	Before the end of burst, new read/write stops read/write burst and starts new read/write burst. During read/write burst with auto precharge, CAS# interrupt can not be issued.

68 Function Truth Table (I)

Current State	CS#	RAS#	CAS#	WE#	BA	Address	Action	Note
Idle	H	X	X	X	X	X	NOP	
	L	H	H	H	X	X	NOP	
	L	H	H	L	X	X	ILLEGAL	2
	L	H	L	X	BA	CA, A_{10}/AP	ILLEGAL	2
	L	L	H	H	BA	RA	Row (& Bank) Active; Latch RA	
	L	L	H	L	BA	A_{10}/AP	NOP	4
	L	L	L	H	X	X	Auto Refresh or Self Refresh	5
	L	L	L	L	OP code	OP code	Mode Register Access	5
Row Active	H	X	X	X	X	X	NOP	
	L	H	H	H	X	X	NOP	
	L	H	H	L	X	X	ILLEGAL	2
	L	H	L	H	BA	CA, A_{10}/AP	Begin Read; latch CA; determine AP	
	L	L	L	L	BA	CA, A_{10}/AP	Begin Read; latch CA; determine AP	
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A_{10}/AP	Precharge	
	L	L	L	X	X	X	ILLEGAL	
Read	H	X	X	X	X	X	NOP (Continue Burst to End --> Row Active)	
	L	H	H	H	X	X	NOP (Continue Burst to End --> Row Active)	
	L	H	H	L	X	X	Term burst --> Row active	
	L	H	L	H	BA	CA, A_{10}/AP	Term burst, New Read, Determine AP	
	L	H	L	L	BA	CA, A_{10}/AP	Term burst, New Write, Determine AP	3
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A_{10}/AP	Term burst, Precharge timing for Reads	
	L	L	L	X	X	X	ILLEGAL	
Write	H	X	X	X	X	X	NOP (Continue Burst to End --> Row Active)	
	L	H	H	H	X	X	NOP (Continue Burst to End --> Row Active)	
	L	H	H	L	X	X	Term burst --> Row active	
	L	H	L	H	BA	CA, A_{10}/AP	Term burst, New Read, Determine AP	3
	L	H	L	L	BA	CA, A_{10}/AP	Term burst, New Write, Determine AP	3
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A_{10}/AP	Term burst, Precharge timing for Reads	3
	L	L	L	X	X	X	ILLEGAL	
Read with Auto Precharge	H	X	X	X	X	X	NOP (Continue Burst to End --> Precharge)	
	L	H	H	H	X	X	NOP (Continue Burst to End --> Precharge)	
	L	H	H	L	X	X	ILLEGAL	
	L	H	L	X	BA	CA, A_{10}/AP	ILLEGAL	
	L	L	H	X	BA	RA, RA_{10}	ILLEGAL	2
	L	L	L	X	X	X	ILLEGAL	
Write with Auto Precharge	H	X	X	X	X	X	NOP (Continue Burst to End --> Precharge)	
	L	H	H	H	X	X	NOP (Continue Burst to End --> Precharge)	
	L	H	H	L	X	X	ILLEGAL	
	L	H	L	X	BA	CA, A_{10}/AP	ILLEGAL	
	L	L	H	X	BA	RA, RA_{10}	ILLEGAL	2
	L	L	L	X	X	X	ILLEGAL	
Precharging	H	X	X	X	X	X	NOP --> Idle after t_{RP}	
	L	H	H	H	X	X	NOP --> Idle after t_{RP}	
	L	H	H	L	X	X	ILLEGAL	2
	L	H	L	X	BA	CA	ILLEGAL	2
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A_{10}/AP	NOP --> Idle after t_{RP}	4

Current State	CS#	RAS#	CAS#	WE#	BA	Address	Action	Note
Row Activating	L	L	L	X	X	X	ILLEGAL	
	H	X	X	X	X	X	NOP --> Row Active after t_{RCD}	
	L	H	H	H	X	X	NOP --> Row Active after t_{RCD}	
	L	H	H	L	X	X	ILLEGAL	2
	L	H	L	X	BA	CA	ILLEGAL	2
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A_{10}/AP	ILLEGAL	2
	L	L	L	X	X	X	ILLEGAL	
Refreshing	H	X	X	X	X	X	NOP --> Idle after t_{RC}	
	L	H	H	X	X	X	NOP --> Idle after t_{RC}	
	L	H	L	X	X	X	ILLEGAL	
	L	L	H	X	X	X	ILLEGAL	
	L	L	L	X	X	X	ILLEGAL	
Mode Register Refreshing	H	X	X	X	X	X	NOP --> Idle after 2 clocks	
	L	H	H	H	X	X	NOP --> Idle after 2 clocks	
	L	H	H	L	X	X	ILLEGAL	
	L	H	L	X	X	X	ILLEGAL	
	L	L	X	X	X	X	ILLEGAL	

Legend: *RA = Row Address, BA = Bank Address, NOP = No Operation Command, CA = Column Address, AP = Auto Precharge*

Notes:

1. *All entries assume the CKE was active (High) during the precharge clock and the current clock cycle.*
2. *Illegal to bank in specified state; Function may be legal in the bank indicated by BA, depending on the state of that bank.*
3. *Must satisfy bus contention, bus turn around, and/or write recovery requirements.*
4. *NOP to bank precharging or in idle state. May precharge bank indicated by BA (and A_{10}/AP).*
5. *Illegal if any bank is not idle.*

69 Function Truth Table (2)

Current State	CKE (n-l)	CKE n	CS#	RAS#	CAS#	WE#	Address	Action	Note
Self Refresh	H	X	X	X	X	X	X	Exit Self Refresh --> Idle after t_{sRFX}(ABI)	
	L	H	H	X	X	X	X	Exit Self Refresh --> Idle after t_{sRFX} (ABI)	6
	L	H	L	H	H	H	X	Exit Self Refresh --> Idle after t_{sRFX} (ABI)	6
	L	H	L	H	H	L	X	ILLEGAL	
	L	H	L	H	L	X	X	ILLEGAL	
	L	H	L	L	X	X	X	ILLEGAL	
	L	L	X	X	X	X	X	NOP (Maintain Self Refresh)	
All Banks Precharge Power Down	H	X	X	X	X	X	X	INVALID	
	L	H	H	X	X	X	X	Exit Power Down --> ABI	
	L	H	L	H	H	H	X	Exit Power Down --> ABI	7
	L	H	L	H	H	L	X	ILLEGAL	7
	L	H	L	H	L	X	X	ILLEGAL	
	L	H	L	L	X	X	X	ILLEGAL	
	L	L	X	X	X	X	X	NOP (Maintain Low Power Mode)	
All Banks Idle	H	H	X	X	X	X	X	Refer to theFunction Truth Table (1)	
	H	L	H	X	X	X	X	Enter Power Down	
	H	L	L	H	H	H	X	Enter Power Down	8
	H	L	L	H	H	L	X	ILLEGAL	8
	H	L	L	H	L	X	X	ILLEGAL	
	H	L	L	L	H	H	RA	Row (& Bank) Active	
	H	L	L	L	L	H	X	Enter Self Refresh	8
	H	L	L	L	L	L	OP Code	Mode Register Access	
	L	L	X	X	X	X	X	NOP	
Any State other than Listed above	H	H	X	X	X	X	X	Refer to Operations in Function Truth Table (1)	
	H	L	X	X	X	X	X	Begin Clock Suspend next cycle	9
	L	H	X	X	X	X	X	Exit Clock Suspend next cycle	9
	L	L	X	X	X	X	X	Maintain Clock Suspend	

Legend: *ABI = All Banks Idle, RA = Row Address*

Notes:

6. *CKE low to high transition is asynchronous.*
7. *CKE low to high transition is asynchronous if restarts internal clock. A minimum setup time 1CLK + t_{SS} must be satisfied before any command other than exit.*
8. *Power down and self refresh can be entered only from the all banks idle state.*
9. *Must be a legal command.*

70 AC Characteristics

Figure 70.1 Power Up Sequence for Mobile SDRAM

Notes:

1. *Apply power and attempt to maintain CKE at a high state and all other inputs may be undefined. - Apply V_{DD} before or at the same time as V_{DDQ}.*
2. *Maintain stable power, stable clock and NOP input condition for a minimum of 200µs.*
3. *Issue precharge commands for all banks of the devices.*
4. *Issue 2 or more auto-refresh commands.*
5. *Issue a mode register set command to initialize the mode register.*
6. *Issue a extended mode register set command to define DS or PASR operating type of the device after normal MRS. EMRS cycle is not mandatory and the EMRS command needs to be issued only when DS or PASR is used.The default state without EMRS command issued is half driver strength, all 4 banks refreshed. The device is now ready for the operation selected by EMRS. For operating with DS or PASR, set DS or PASR mode in EMRS setting stage. In order to adjust another mode in the state of DS or PASR mode, additional EMRS set is required but power up sequence is not*

needed again at this time. In that case, all banks have to be in idle state prior to adjusting EMRS set.

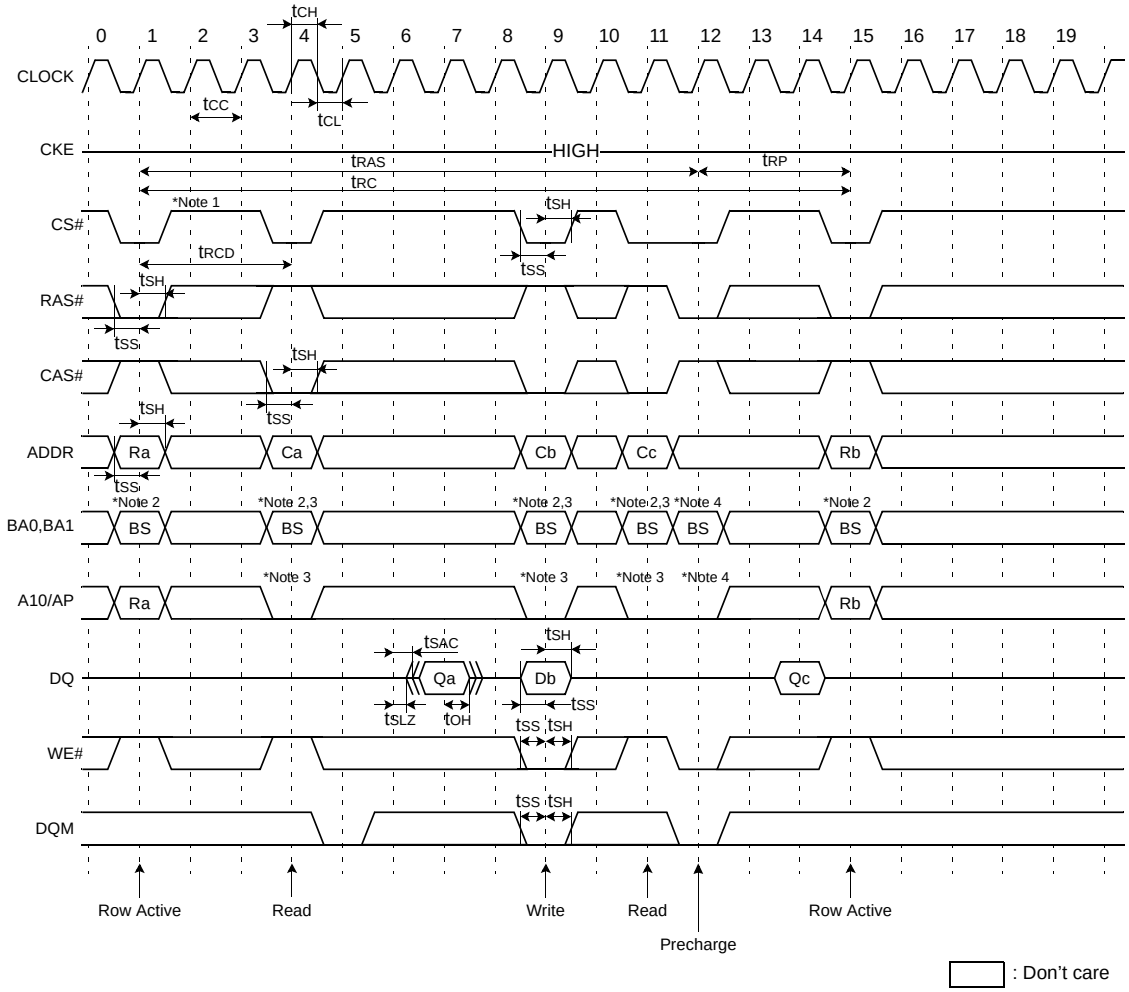


Figure 70.2 Single Bit Read-Write-Read Cycle (Same Page) @CAS Latency=3, Burst Length=l

Notes:

1. *All input except CKE & DQM can be don't care when CS# is high at the CLK high going edge.*
2. *Bank active and read/write are controlled by BA0, BA1.*

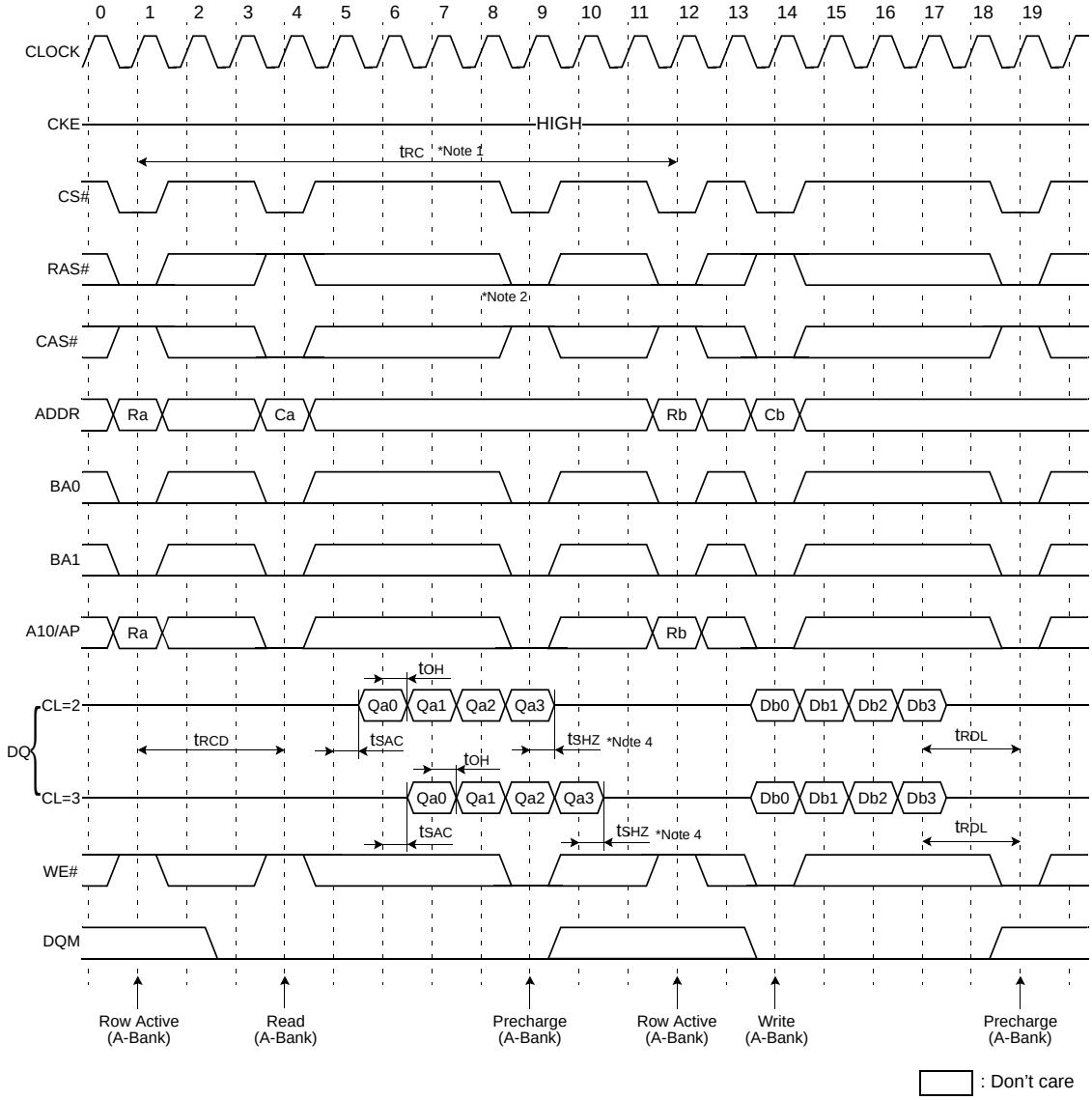


Figure 70.3 Read & Write Cycle at Same Bank @Burst Length=4, t_{RDL}=2CLK

Notes:

1. *Minimum row cycle times is required to complete internal DRAM operation.*
2. *Row precharge can interrupt burst on any cycle. [CAS Latency - 1] number of valid output data is available after Row precharge. Last valid output will be Hi-Z(t_{SHZ}) after the clock.*
3. *Output will be Hi-Z after the end of burst. (1, 2, 4, 8 & Full page bit burst).*

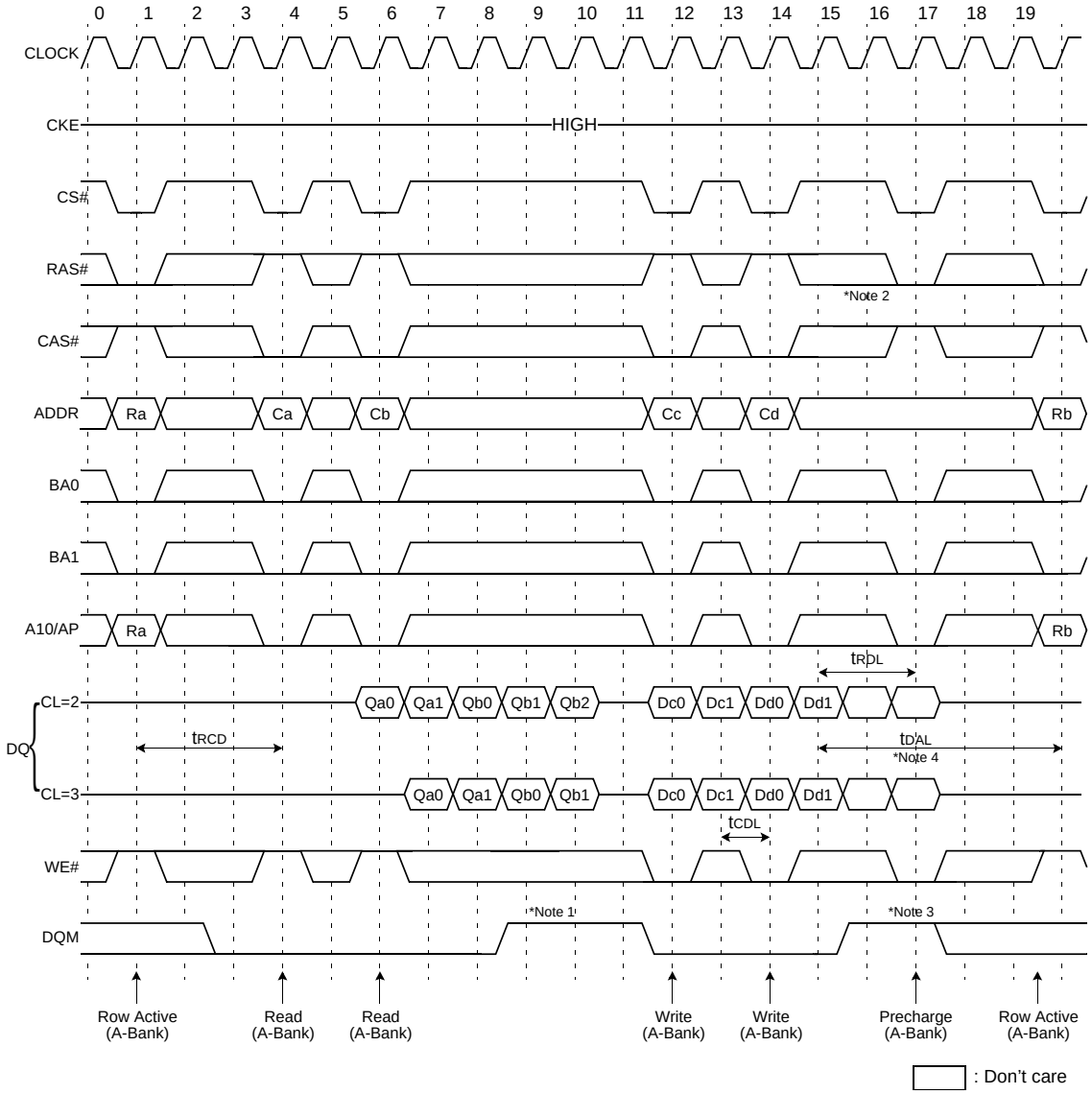


Figure 70.4 Page Read & Write Cycle at Same Bank @Burst Length=4, t_{RDL}=2CLK

Notes:

1. *To write data before burst read ends, DQM should be asserted three cycle prior to write command to avoid bus contention.*
2. *Row precharge will interrupt writing. Last data input, t_{RDL} before Row precharge, will be written.*
3. *DQM should mask invalid input data on precharge command cycle when asserting precharge before end of burst. Input data after Row precharge cycle will be masked internally.*
4. *t_{DAL}, last data in to active delay, is 2CLK + t_{RP}.*

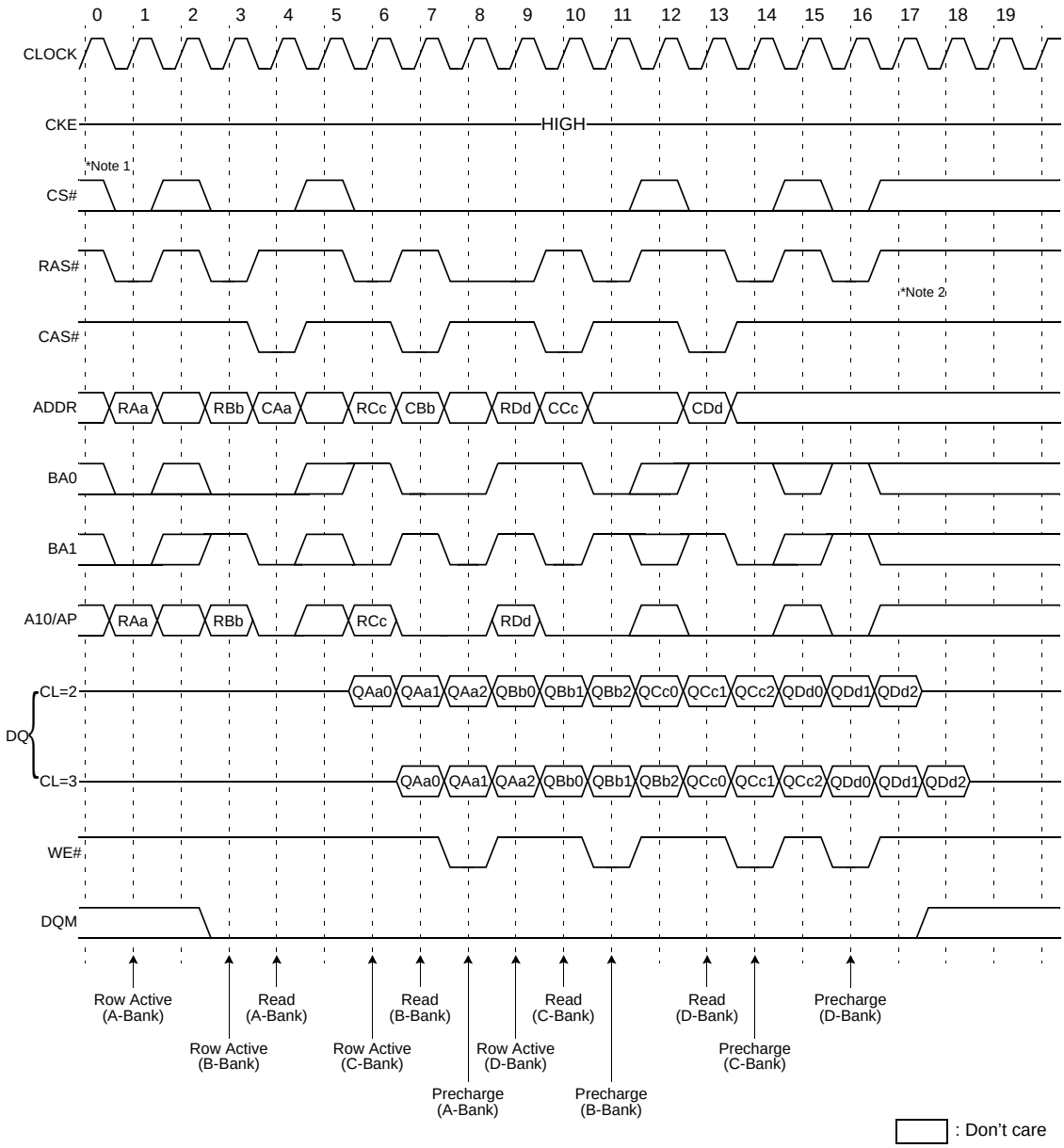


Figure 70.5 Page Read Cycle at Different Bank @Burst Length=4

Notes:

1. *CS# can be don't cared when RAS#, CAS# and WE# are high at the clock high going edge.*
2. *To interrupt a burst read by row precharge, both the read and the precharge banks must be the same.*

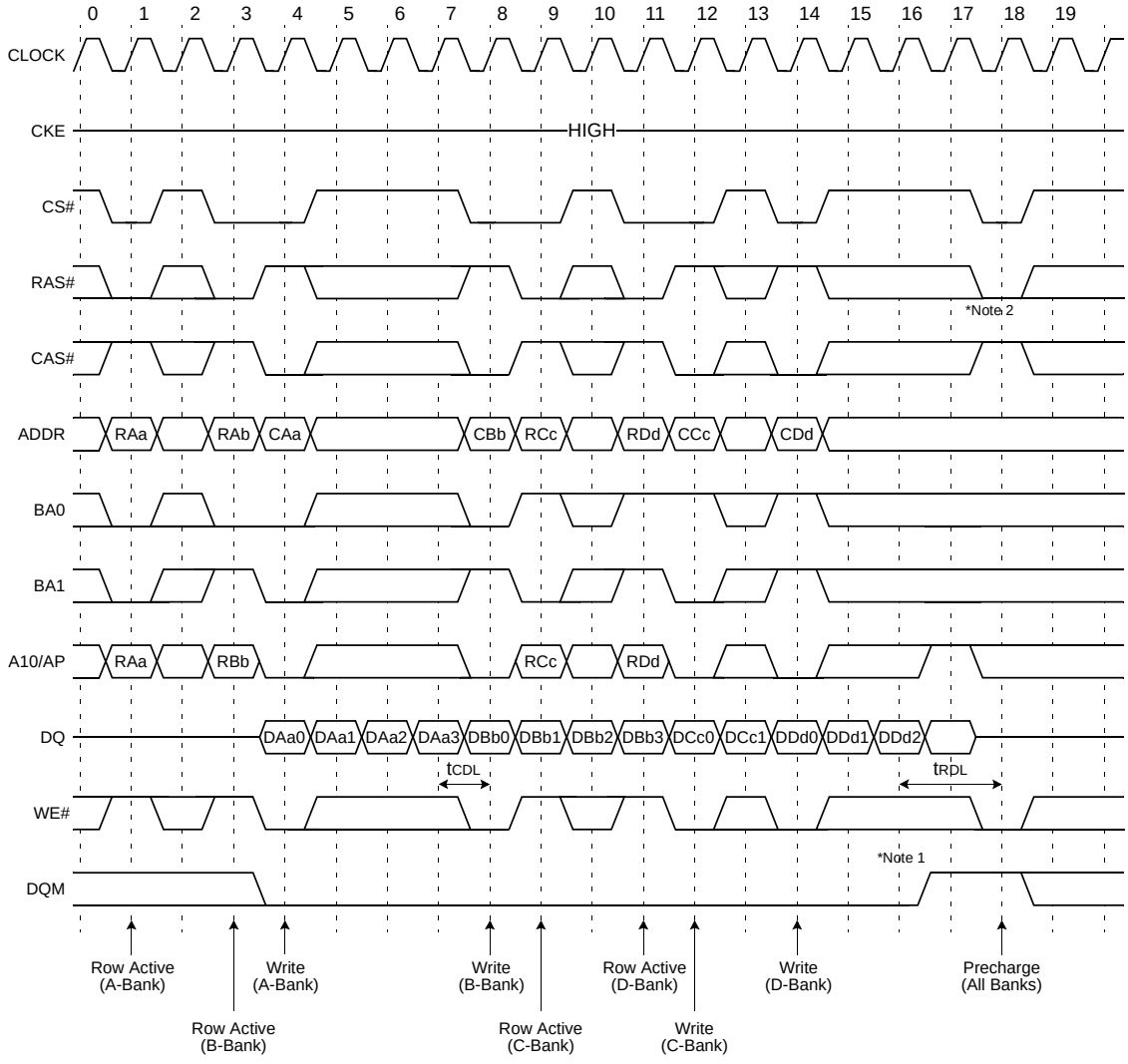


Figure 70.6 Page Write Cycle at Different Bank @Burst Length=4, t_{RDL}=2CLK

Notes:

1. *To interrupt burst write by Row precharge, DQM should be asserted to mask invalid input data.*
2. *To interrupt burst write by Row precharge, both the write and the precharge banks must be the same.*

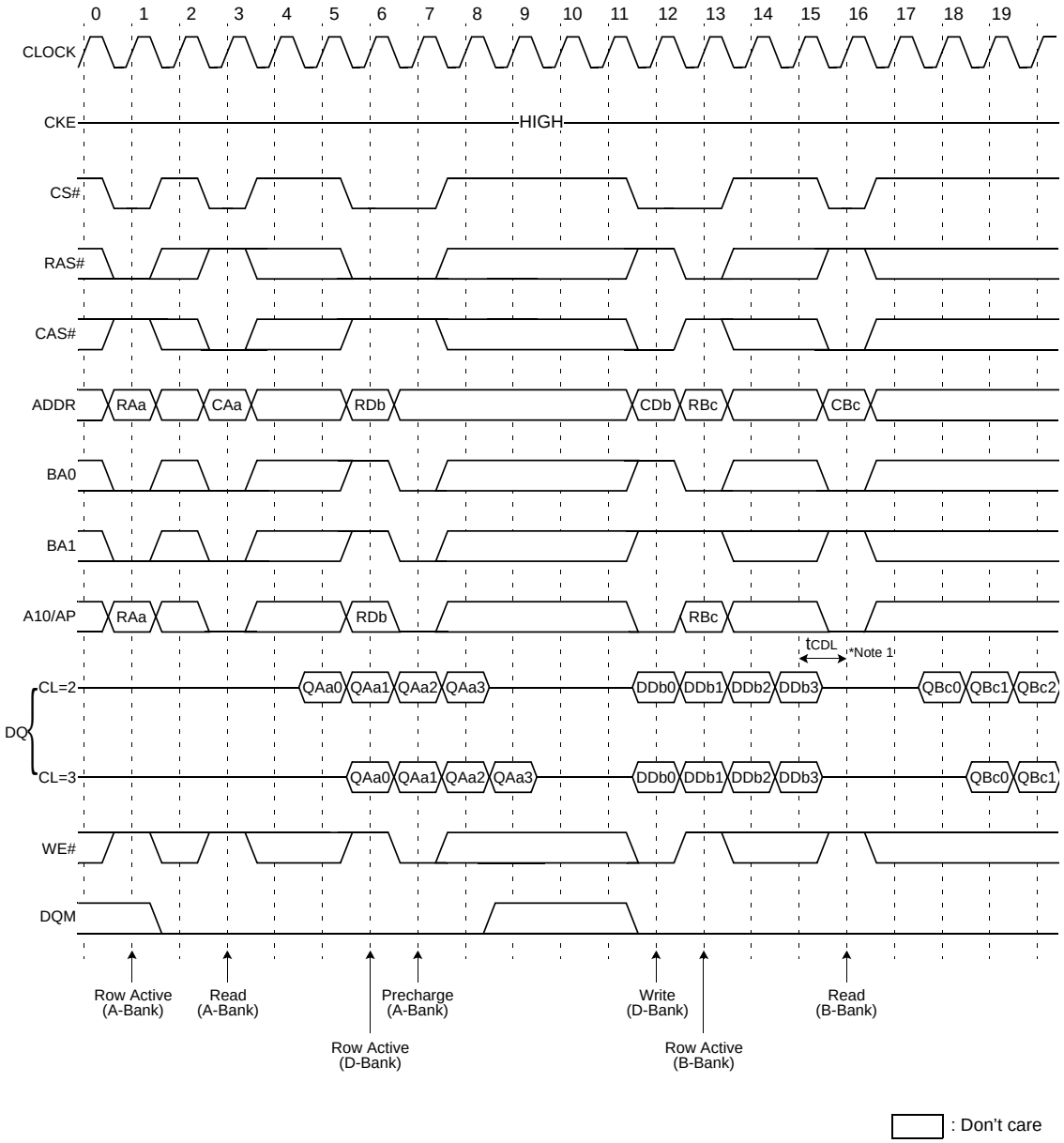


Figure 70.7 Read & Write Cycle at Different Bank @Burst Length=4

Notes:

1. *t_{CDL} should be met to complete write.*

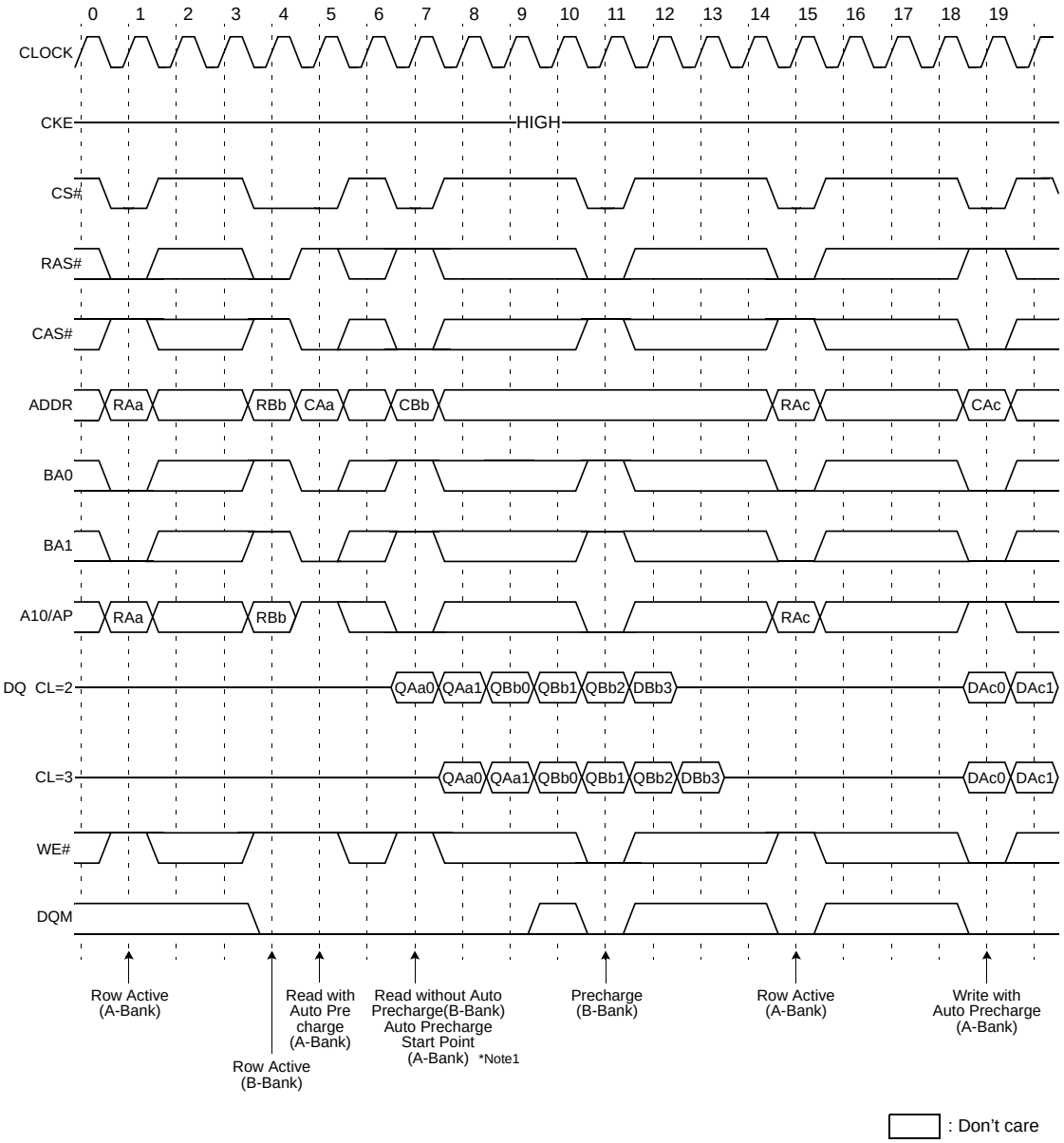


Figure 70.8 Read & Write Cycle with Auto Precharge I @Burst Length=4

Notes:

1. *When Read (Write) command with auto precharge is issued at A-Bank after A and B Bank activation.*
   - if Read (Write) command without auto precharge is issued at B-Bank before A-Bank auto precharge starts, A-Bank auto precharge will start at B-Bank read command input point.
   - any command can not be issued at A-Bank during tRP after A-Bank auto precharge starts.

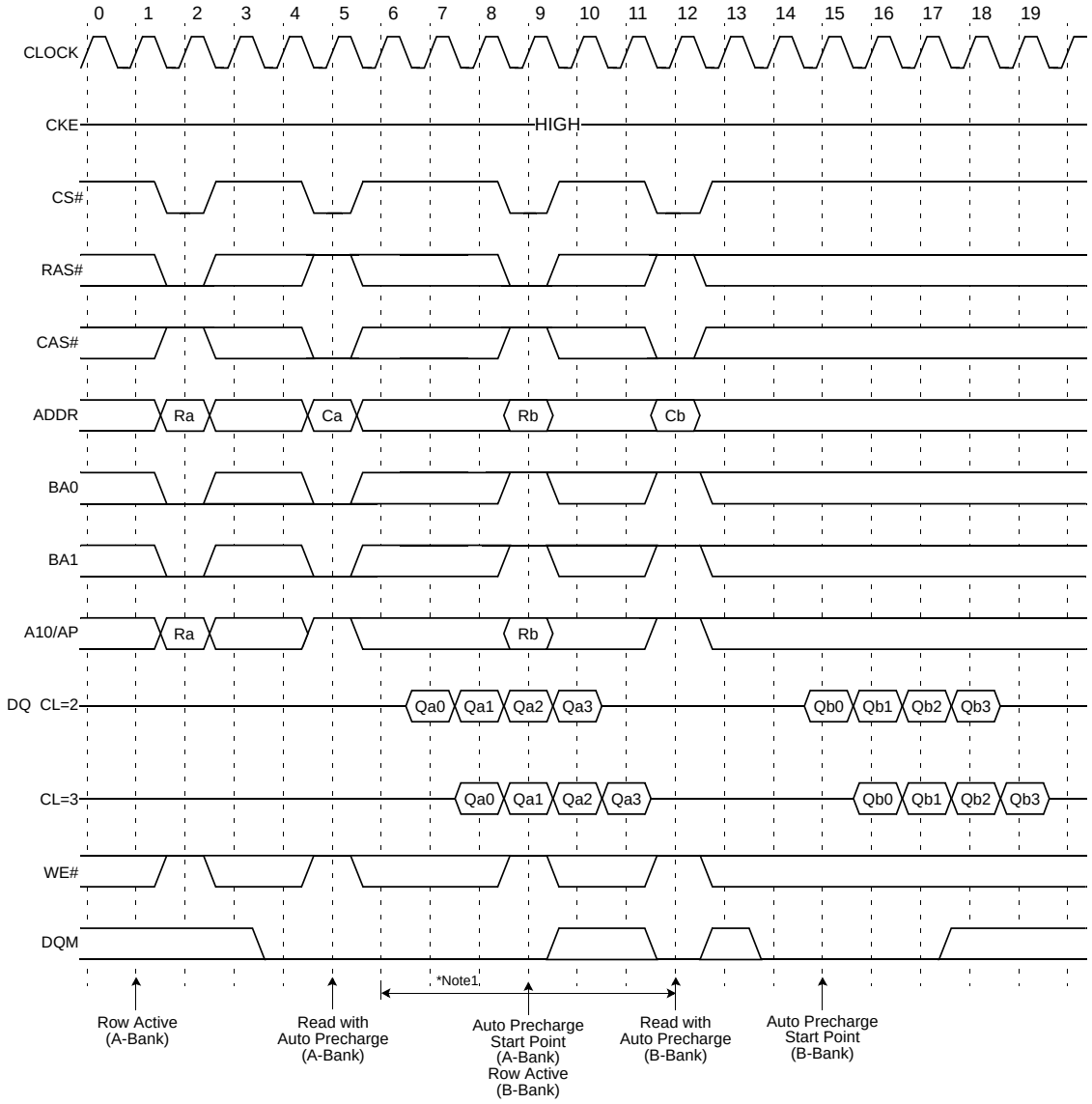


Figure 70.9 Read & Write Cycle with Auto Precharge 2 @Burst Length=4

Notes:

1. *Any command to A-bank is not allowed in this period. t_{RP} is determined from at auto precharge start point.*

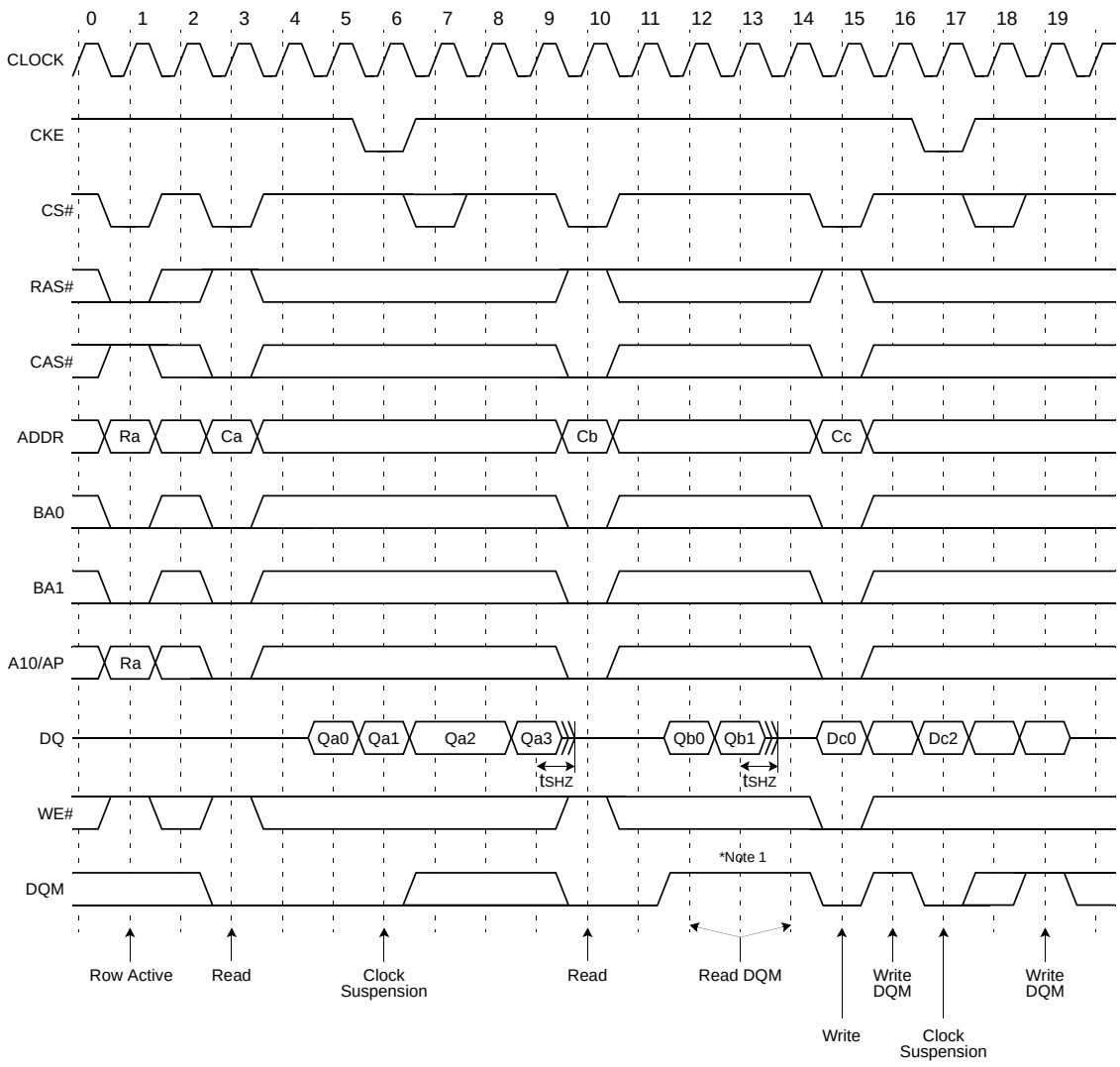


Figure 70.10 Clock Suspension & DQM Operation Cycle @CAS Latency=2, Burst Length=4

Notes:

1. DQM is needed to prevent bus contention.

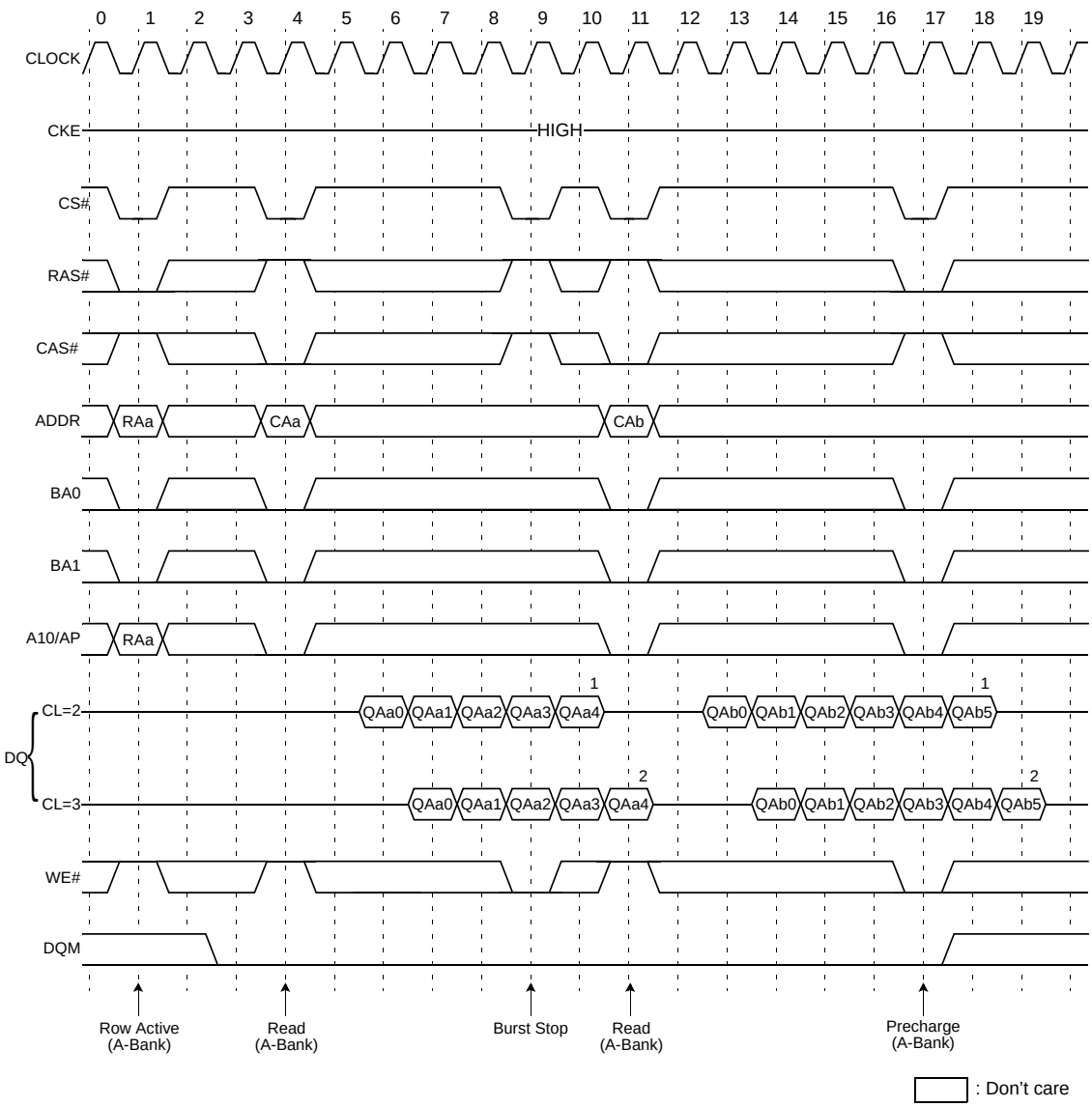


Figure 70.11 Read Interrupted by Precharge Command & Read Burst Stop Cycle @Full Page Burst

Notes:

1. *At full page mode, burst is finished by burst stop or precharge.*
2. *About the valid DQs after burst stop, it is same as the case of RAS# interrupt. Both cases are illustrated above timing diagram. See the label 1, 2 on them. But at burst write, Burst stop and RAS# interrupt should be compared carefully. Refer the timing diagram of "Full page write burst stop cycle".*
3. *Burst stop is valid at every burst length.*

Figure 70.12 Write Interrupted by Precharge Command & Write Burst Stop Cycle @ Full Page Burst, t_{RDL}=2CLK

Notes:

1. *At full page mode, burst is finished by burst stop or precharge.*
2. *Data-in at the cycle of interrupted by precharge can not be written into the corresponding memory cell. It is defined by AC parameter of t_{RDL}.*
   DQM at write interrupted by precharge command is needed to prevent invalid write.
   DQM should mask invalid input data on precharge command cycle when asserting precharge before end of burst. Input data after Row precharge cycle will be masked internally.
3. *Burst stop is valid at every burst length.*

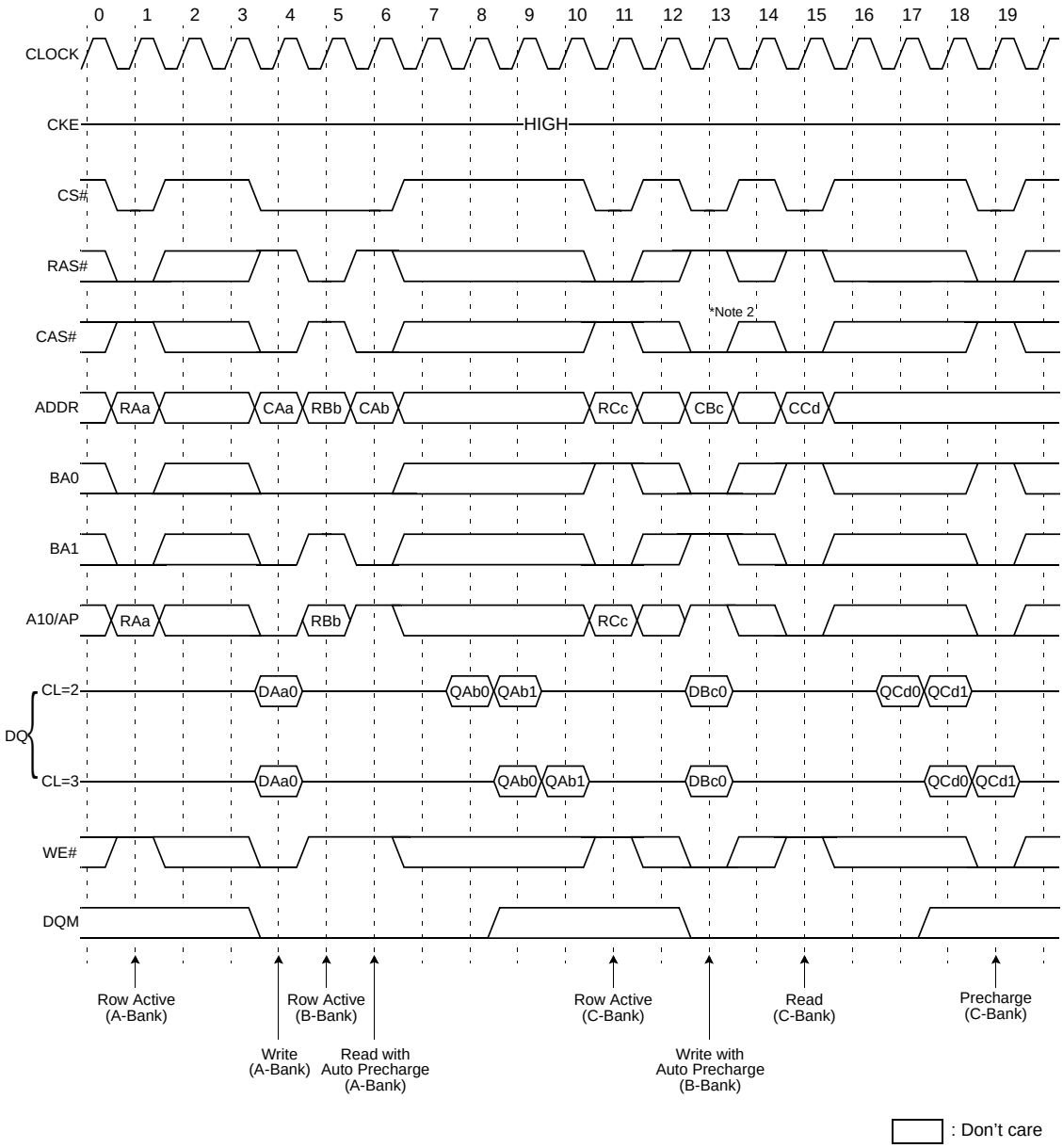


Figure 70.13 Burst Read Single bit Write Cycle @Burst Length=2

Notes:

1. *BRSW modes is enabled by setting A9 "High" at MRS (Mode Register Set). At the BRSW Mode, the burst length at write is fixed to "1" regardless of programmed burst length.*
2. *When BRSW write command with auto precharge is executed, keep it in mind that tRAS should not be violated. Auto precharge is executed at the burst-end cycle, so in the case of BRSW write command, the next cycle starts the precharge.*

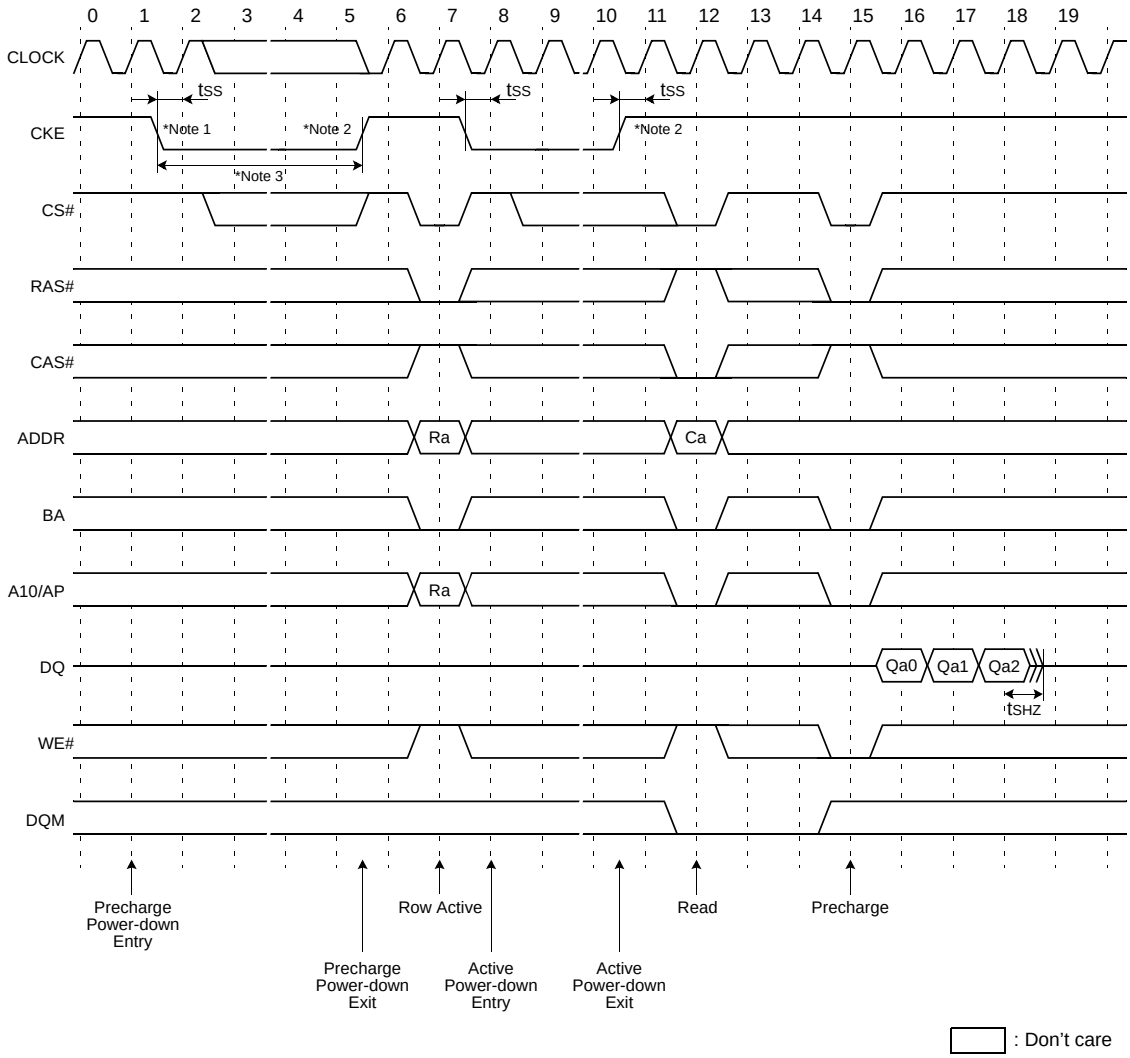


Figure 70.14 Active/Precharge Power Down Mode @CAS Latency=2, Burst Length=4

Notes:

1. *All banks should be in idle state prior to entering precharge power down mode.*
2. *CKE should be set high at least 1CLK + t_{SS} prior to Row active command.*
3. *Can not violate minimum refresh specification (64ms).*

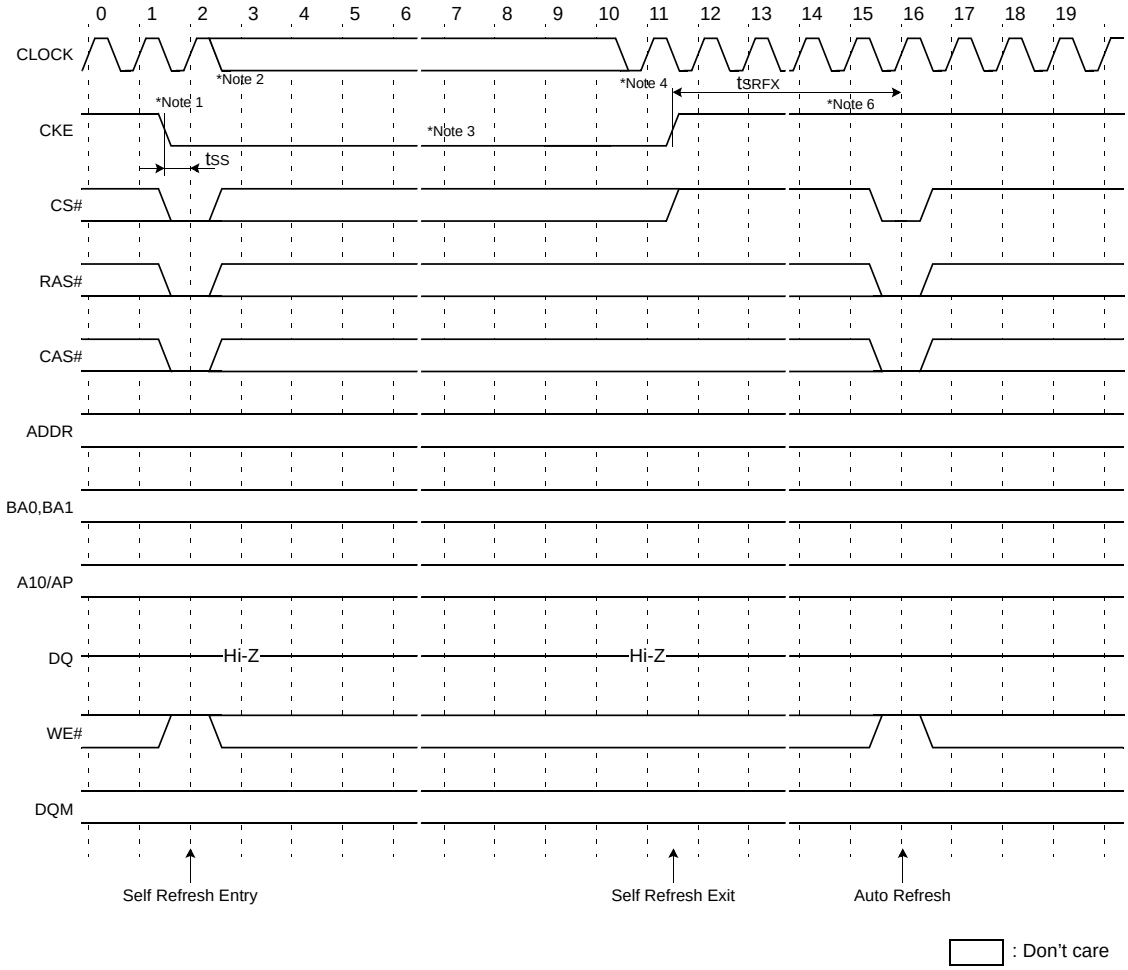


Figure 70.15 Self Refresh Entry & Exit Cycle

Notes:

TO ENTER SELF REFRESH MODE

1. *CS#, RAS# & CAS# with CKE should be low at the same clock cycle.*
2. *After 1 clock cycle, all the inputs including the system clock can be don't care except for CKE.*
3. *The device remains in self refresh mode as long as CKE stays "Low". Once the device enters self refresh mode, minimum t_{RAS} is required before exit from self refresh.*

TO EXIT SELF REFRESH MODE

4. *System clock restart and be stable before returning CKE high.*
5. *CS# starts from high.*
6. *Minimum t_{SRFX} is required after CKE going high to complete self refresh exit.*
7. *4K cycle of burst auto refresh is required before self refresh entry and after self refresh exit if the system uses burst refresh.*

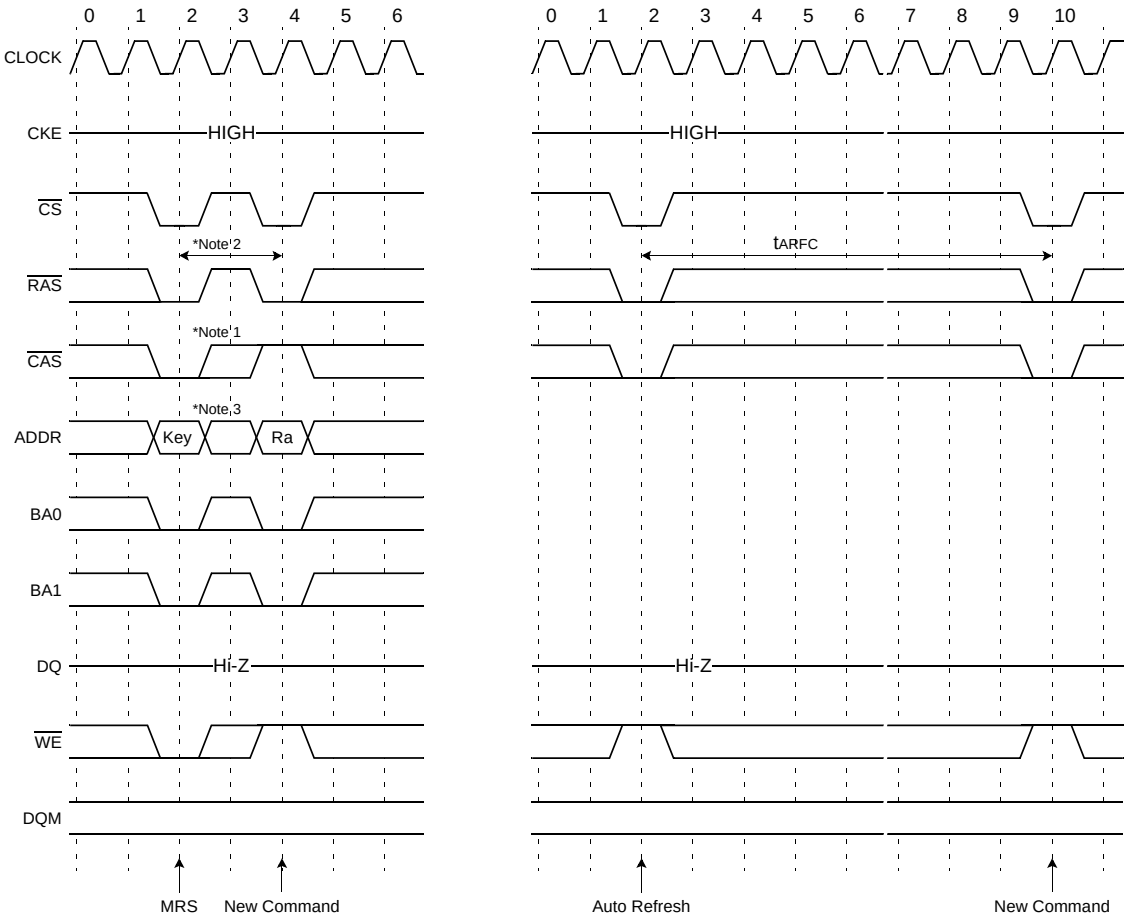


Figure 70.16 Mode Register Set Cycle and Auto Refresh Cycle

Note: *All banks precharge should be completed before Mode Register Set cycle and auto refresh cycle.*

Notes:

1. *CS#, RAS#, CAS#, BA0, BA1 & WE# activation at the same clock cycle with address key will set internal mode register.*
2. *Minimum 2 clock cycles should be met before new RAS# activation.*
3. *Please refer to the Mode Register Field Table to Program Modes*

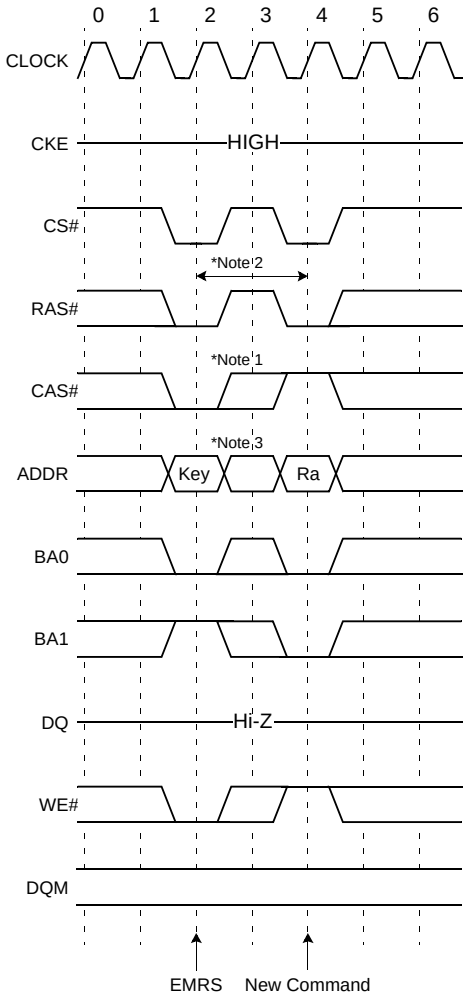


Figure 70.17 Extended Mode Register Set Cycle

Notes:

1. *CS#, RAS#, CAS#, BA0, BA1 & WE# activation at the same clock cycle with address key will set internal mode register.*
2. *Minimum 2 clock cycles should be met before new RAS# activation.*
3. *Please refer to the Mode Register Field Table to Program Modes.*

71 SDRAM Type 2 Revision Summary

Revision A0 (May 25, 2004)

Initial Release

Mobile SDRAM Type 2

256Mbit (16M x 16 bit) SDRAM

PRELIMINARY

Features

- **1.8V power supply**
- **LVCMOS compatible with multiplexed address**
- **Four banks operation**
- **MRS cycle with address key programs**
  - CAS latency (1, 2 & 3)
  - Burst length (1, 2, 4, 8 & Full page)
  - Burst type (Sequential & Interleave)
- **EMRS cycle with address key programs**
- **All inputs are sampled at the positive going edge of the system clock**
- **Burst read single-bit write operation**
- **Special Function Support**
  - PASR (Partial Array Self Refresh)
  - Internal TCSR (Temperature Compensated Self Refresh)
  - DS (Driver Strength)
- **DQM for masking**
- **Auto refresh.**
- **64ms refresh period (8K cycle)**
- **Commercial Temperature Operation (-25°C - 70°C)**
- **Extended Temperature Operation (-25°C - 85°C)**

General Description

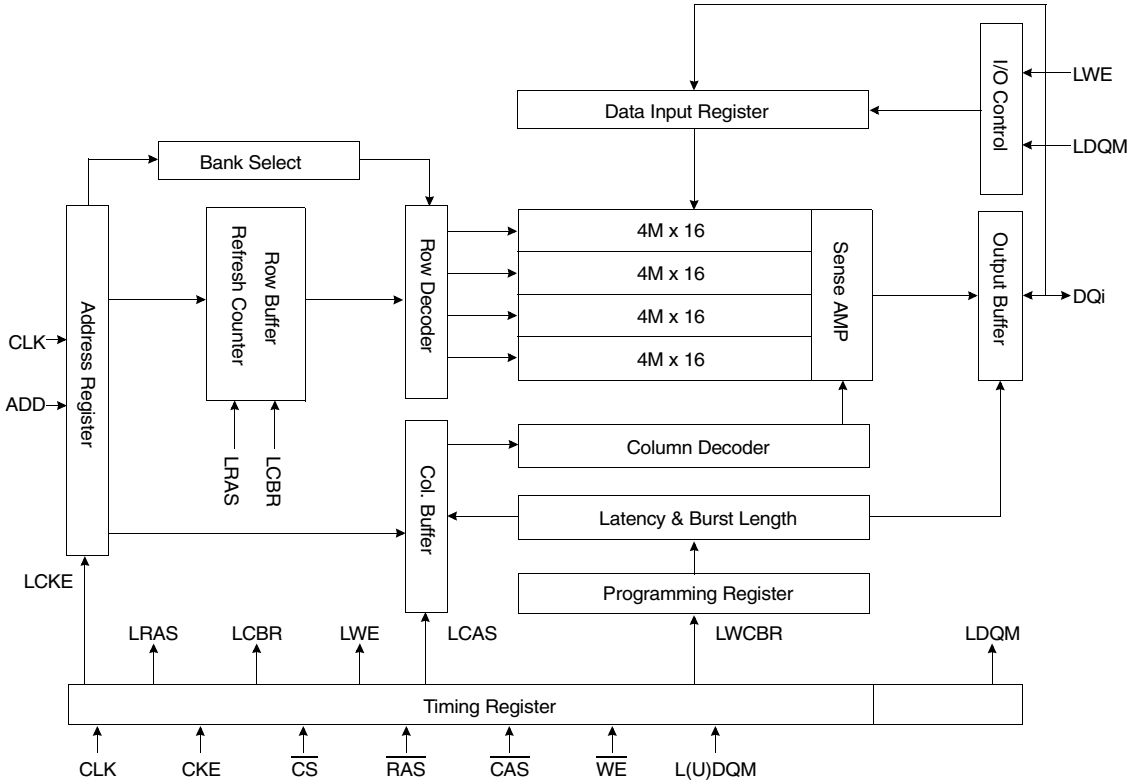
The Mobile SDRAM Type 2 is 268,435,456 bits synchronous high data rate Dynamic RAM organized as 4 x 4,196,304 words by 16 bits, fabricated with CMOS technology. Synchronous design allows precise cycle control with the use of system clock and I/O transactions are possible on every clock cycle. Range of operating frequencies, programmable burst lengths and programmable latencies allow the same device to be useful for a variety of high bandwidth and high performance memory system applications.

Publication Number SDRAM_04 **Revision** A **Amendment** 0 **Issue Date** May 12, 2005

72 Address Table

	16M x 16
Configuration	4 M x 16 x 4 banks
Refresh Count	4K
Row Addressing	4K (A0–A12)
Bank Addressing	4 (BA0, BA1)
Column Addressing	512 (A0–A8)

73 Functional Block Diagram



74 Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Voltage on any pin relative to V_{SS}	V_{IN}, V_{OUT}	-1.0 ~ 2.6	V
Voltage on V_{DD} supply relative to V_{SS}	V_{DD}, V_{DDQ}	-1.0 ~ 2.6	V
Storage temperature	T_{STG}	-55 ~ +150	°C
Power dissipation	P_D	1.0	W
Short circuit current	I_{OS}	50	mA

Notes:

1. *Permanent device damage may occur if Absolute Maximum Ratings are exceeded.*
2. *Functional operation should be restricted to recommended operating condition.*
3. *Exposure to higher than recommended voltage for extended periods of time could affect device reliability.*

75 DC Operating Conditions

Recommended operating conditions (Voltage referenced to V_{SS} = 0V, T_A = -25°C to 85°C for Extended, -25°C to 70°C for Commercial).

Parameter	Symbol	Min	Typ	Max	Unit	Note
Supply voltage	VDD	1.7	1.8	1.95	V	
	VDDQ	1.7	1.8	1.95	V	
Input logic high voltage	VIH	0.8 x VDDQ	1.8	VDDQ + 0.3	V	1
Input logic low voltage	VIL	-0.3	0	0.3	V	2
Output logic high voltage	VOH	VDDQ -0.2	-	-	V	IOH = -0.1mA
Output logic low voltage	VOL	-	-	0.2	V	IOL = 0.1mA
Input leakage current	ILI	-2	-	2	µA	3

Notes:

1. *$V_{IH\ (max)}$ = 2.2V AC.The overshoot voltage duration is ≤ 3ns.*
2. *$V_{IL\ (min)}$ = -1.0V AC. The undershoot voltage duration is ≤ 3ns.*
3. *Any input 0V ≤ V_{IN} ≤ V_{DDQ}. Input leakage currents include Hi-Z output leakage for all bi-directional buffers with tri-state outputs.*
4. *D_{OUT} is disabled, 0V ≤ V_{OUT} ≤ V_{DDQ}.*

76 Capacitance

(VDD = 1.8V, TA = 23°C, f = 1MHz, VREF =0.9V ± 50 mV).

Pin	Symbol	Min	Max	Unit	Note
Clock	C_{CLK}	1.5	3.0	pF	
RAS#, CAS#, WE#, CS#, CKE, DQM	C_{IN}	1.5	3.0	pF	
Address	C_{ADD}	1.5	3.0	pF	
DQ0 - DQ15	C_{OUT}	3.0	5.0	pF	

77 DC Characteristics

Recommended operating conditions (Voltage referenced to V_{SS} = 0V, T_A = -25°C to 85°C for Extended, -25°C to 70°C for Commercial).

Parameter	Symbol	Test Condition		Value		Unit	Note
Operating Current (One Bank Active)	I_{CC1}	Burst length = 1 $t_{RC} \geq t_{RC(min)}$ I_O = 0 mA		45		mA	1
Precharge Standby Current in power-down mode	I_{CC2P}	CKE ≤ $V_{IL(max)}$, t_{CC} = 10ns		0.3		mA	
	I_{CC2PS}	CKE & CLK ≤ $V_{IL(max)}$, t_{CC} = ∞		0.3			
Precharge Standby Current in non power-down mode	I_{CC2N}	CKE ≥ $V_{IH(min)}$, CS# ≥ $V_{IH(min)}$, t_{CC} = 10ns Input signals are changed one time during 20ns		10		mA	
	I_{CC2NS}	CKE ≥ $V_{IH(min)}$, CLK ≤ $V_{IL(max)}$, t_{CC} = ∞ Input signals are stable		1			
Active Standby Current in power-down mode	I_{CC3P}	CKE ≤ $V_{IL(max)}$, t_{CC} = 10ns		5		mA	
	I_{CC3PS}	CKE & CLK ≤ $V_{IL(max)}$, t_{CC} = ∞		1			
Active Standby Current in non power-down mode (One Bank Active)	I_{CC3N}	CKE ≥ $V_{IH(min)}$, CS# ≥ $V_{IH(min)}$, t_{CC} = 10ns Input signals are changed one time during 20ns		20		mA	
	I_{CC3NS}	CKE ≥ $V_{IH(min)}$, CLK ≤ $V_{IL(max)}$, t_{CC} = ∞ Input signals are stable		5		mA	
Operating Current (Burst Mode)	I_{CC4}	I_O = 0 mA Page burst 4Banks Activated t_{CCD} = 2CLKs		60		mA	1
Refresh Current	I_{CC5}	$t_{ARFC} \geq t_{ARFC(min)}$		85		mA	2
Self Refresh Current	I_{CC6}	CKE ≤ 0.2V	**Internal TCSR**	Max 40	Max 70/85	°C	
			Full Array	200	450	µA	
			1/2 of Full Array	160	300		
			1/4 of Full Array	140	250		

Notes:

1. *Measured with outputs open.*
2. *Refresh period is 64ms.*
3. *Unless otherwise noted, input swing IeveI is CMOS(V_{IH} /V_{IL}=V_{DDQ}/V_{SSQ}).*

78 AC Operating Test Conditions

(V_{DD} = 1.7V to 1.95V, T_A = -25°C to 85°C for Extended, -25°C to 70°C for Commercial).

Parameter	Value	Unit
AC input levels (V_{IH}/V_{IL})	0.9 x V_{DDQ} / 0.2	V
Input timing measurement reference level	0.5 x V_{DDQ}	V
Input rise and fall time	tr/tf = 1/1	ns
Output timing measurement reference level	0.5 x V_{DDQ}	V
Output load condition	See Figure 78.2	

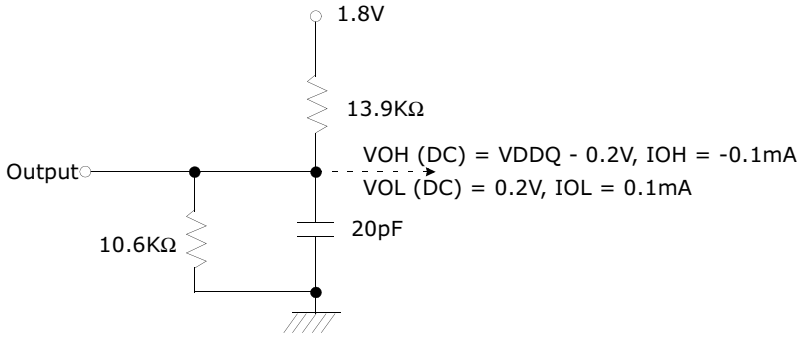


Figure 78.1 DC Output Load Circuit

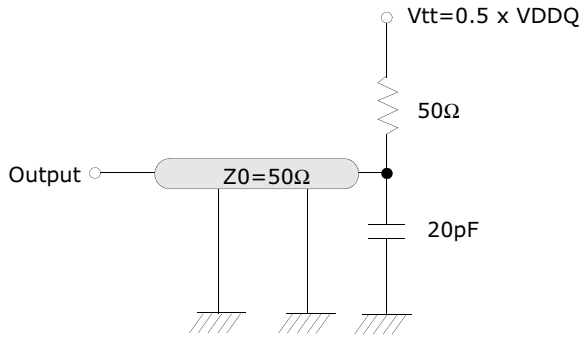


Figure 78.2 AC Output Load Circuit

79 Operating AC Parameters

AC operating conditions unless otherwise noted.

Parameter		Symbol	Value	Unit	Note
Row active to row active delay		$t_{RRD(min)}$	18	ns	1
RAS# to CAS# delay		$t_{RCD(min)}$	24	ns	1
Row precharge time		$t_{RP(min)}$	24	ns	1
Row active time		$t_{RAS(min)}$	50	ns	1
		$t_{RAS(max)}$	100	us	
Row cycle time		$t_{RC(min)}$	74	ns	1
Last data in to row precharge		$t_{RDL(min)}$	15	ns	2
Last data in to Active delay		$t_{DAL(min)}$	t_{RDL} + t_{RP}	-	
Last data in to new col. address delay		$t_{CDL(min)}$	1	CLK	2
Last data in to burst stop		$t_{BDL(min)}$	1	CLK	2
Auto refresh cycle time		$t_{ARFC(min)}$	80	ns	
Exit self refresh to active command		$t_{SRFX(min)}$	120	ns	
Col. address to col. address delay		$t_{CCD(min)}$	1	CLK	3
Number of valid output data	CAS latency=3		2	ea	4
Number of valid output data	CAS latency=2		1		
Number of valid output data	CAS latency=1		—		

Notes:

1. *The minimum number of clock cycles is determined by dividing the minimum time required with clock cycle time and then rounding off to the next higher integer.*
2. *Minimum delay is required to complete write.*
3. *All parts allow every cycle column address change.*
4. *In case of row precharge interrupt, auto precharge and read burst stop.*

80 AC Characteristics

AC operating conditions unless otherwise noted.

Parameter		Symbol	Min	Max	Unit	Note
CLK cycle time	CAS latency=3	t_{CC}	9	1000	ns	1
	CAS latency=2	t_{CC}	12			
CLK to valid output delay	CAS latency=3	t_{SAC}		7	ns	1,2s
	CAS latency=2	t_{SAC}		9		
Output data hold time	CAS latency=3	t_{OH}	2.0		ns	2
	CAS latency=2	t_{OH}	2.0			
CLK high pulse width		t_{CH}	3.0		ns	3
CLK low pulse width		t_{CL}	3.0		ns	3
Input setup time		t_{SS}	2.0		ns	3
Input hold time		t_{SH}	1		ns	3
CLK to output in Low-Z		t_{SLZ}	1		ns	2
CLK to output in Hi-Z	CAS latency=3	t_{SHZ}		7	ns	
	CAS latency=2			9		

Notes:

1. *Parameters depend on programmed CAS latency.*
2. *If clock rising time is longer than 1ns, (tr/2-0.5)ns should be added to the parameter.*
3. *Assumed input rise and fall time (tr & tf) = 1ns. If tr & tf is longer than 1ns, transient time compensation should be considered, i.e., [(tr + tf)/2-1]ns should be added to the parameter.*

81 Simplified Truth Table

Command			CKEn-1	CKEn	CS#	RAS#	CAS#	WE#	DQM	BA0,1	A10/AP	A12, A11, A9 ~ A0	Note
Register	Mode Register Set		H	X	L	L	L	L	X	OP CODE			1, 2
Refresh	Auto Refresh		H	H	L	L	L	H	X	X			3
				L									3
	Self Refresh	Entry											
		Exit	L	H	L	H	H	H	X	X			3
					H	X	X	X					3
Bank Active & Row Addr.			H	X	L	L	H	H	X	V	Row Address		
Read & Column Address	Auto Precharge Disable		H	X	L	H	L	H	X	V	L	Column Address (A0-A8)	4
	Auto Precharge Enable										H		4, 5
Write & Column Address	Auto Precharge Disable		H	X	L	H	L	L	X	V	L	Column Address (A0-A8)	4
	Auto Precharge Enable										H		4, 5
Burst Stop			H	X	L	H	H	L	X	X			6
Precharge	Bank Selection		H	X	L	L	H	L	X	V	L	X	
	All Banks									X	H		
Clock Suspend or Active Power Down		Entry	H	L	H	X	X	X	X	X			
					L	V	V	V					
		Exit	L	H	X	X	X	X	X				
Precharge Power Down Mode		Entry	H	L	H	X	X	X	X	X			
					L	H	H	H					
		Exit	L	H	H	X	X	X	X				
					L	V	V	V					
DQM			H	X					V	X			7
No Operation Command			H	X	H	X	X	X	X	X			
					L	H	H	H					

Legend: *V = Valid, X = Don't Care, H = Logic High, L = Logic Low*

Notes:

1. *OP Code: Operand Code A0 - A12 & BA0 - BA1: Program keys. (@MRS)*
2. *MRS can be issued only at all banks precharge state. A new command can be issued after 2 CLK cycles of MRS.*
3. *Auto refresh functions are the same as CBR refresh of DRAM. The automatically precharge without row precharge command is meant by* Auto. *Auto/self refresh can be issued only at all banks precharge state. Partial self refresh can be issued only after setting partial self refresh mode of EMRS.*
4. *BA0 - BA1: Bank select addresses.*
5. *During burst read or write with auto precharge, new read/write command can not be issued. Another bank read/write command can be issued after the end of burst. New row active of the associated bank can be issued at t_{RP} after the end of burst.*
6. *Burst stop command is valid at every burst length.*
7. *DQM sampled at the positive going edge of CLK masks the data-in at that same CLK in write operation (Write DQM latency is 0), but in read operation, it makes the data-out Hi-Z state after 2 CLK cycles. (Read DQM latency is 2).*

82 Mode Register Field Table to Program Modes

Address	BA0 ~ BA1	A12 ~ A10/AP	A9[2]	A8	A7	A6	A5	A4	A3	A2	A1	A0
Function	*0* Setting for Normal MRS	RFU	W.B.L	Test Mode		CAS Latency			BT	Burst Length		

Note: *Register Programmed with Normal MRS.*

82.1 Normal MRS Mode

Test Mode			CAS Latency				Burst Type			Burst Length				
A8	A7	Type	A6	A5	A4	Latency	A3	Type		A2	AI	A0	BT=0	BT=I
0	0	Mode Register Set	0	0	0	Reserved	0	Sequential		0	0	0	1	1
0	I	Reserved	0	0	1	1	1	Interleave		0	0	1	2	2
I	0	Reserved	0	1	0	2	Mode Select			0	1	0	4	4
I	I	Reserved	0	1	1	3	BAI	BA0	Mode	0	1	1	8	8
Write Burst Length			1	0	0	Reserved	0	0	Setting for Normal MRS	1	0	0	Reserved	Reserved
A9	Length		1	0	1	Reserved				1	0	1	Reserved	Reserved
0	Burst		1	1	0	Reserved				1	1	0	Reserved	Reserved
1	Single Bit		1	1	1	Reserved				1	1	1	Full Page	Reserved

Note: Full Page Length x 16: 64Mb (256), 128Mb (512), 256Mb (512), 512Mb (1024).

Table 82.1 Register Programmed with Extended MRS

Address	BAI BA0	BA0	AI2 ~ AI0/AP AI0	A9	A8	A7	A6	A5	A4	A3	A2	AI	A0
Function	Mode Select		RFU				DS		RFU		PASR		

82.2 EMRS for PASR (Partial Array Self Refresh) and DS (Driver Strength)

Mode Select			Driver Strength			PASR			
BAI	BA0	Mode	A6	A5	Driver Strength	A2	AI	A0	Size of Refreshed Array
0	0	Normal MRS	0	0	Full	0	0	0	Full Array
0	1	Reserved	0	1	1/2	0	0	1	1/2 of Full Array
1	0	EMRS for Mobile SDRAM	1	0	1/4	0	1	0	1/4 of Full Array
1	1	Reserved	1	1	1/8	0	1	1	Reserved
Reserved Address						1	0	0	Reserved
AI2-AI0/AP	A9	A8	A7	A4	A3	1	0	1	Reserved
0	0	0	0	0	0	1	1	0	Reserved
						1	1	1	Reserved

Notes:

1. *RFU (Reserved for future use) should stay* 0 *during MRS cycle.*
2. *If A9 is high during MRS cycle,* Burst Read Single Bit Write *function will be enabled.*

83 Partial Array Self Refresh

In order to save power consumption, Mobile SDRAM has a PASR option. Mobile SDRAM supports 3 kinds of PASR in self refresh mode: full array, 1/2 of full array, 1/4 of full array. See Figure 83.1.

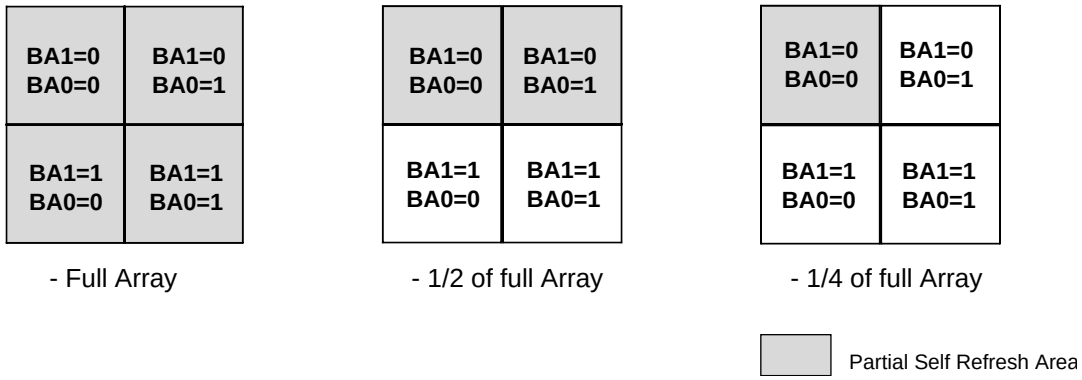


Figure 83.1 Partial Array Self Refresh Areas

83.1 Internal Temperature Compensated Self Refresh (TCSR)

Temperature Range	Self Refresh Current (Icc 6)			Unit
	Full Array	1/2 of Full Array	1/4 of Full Array	
Max. 40 °C	200	160	140	µA
Max. 70/85 °C	450	300	250	

Notes:

1. *In order to save power consumption, Mobile-SDRAM includes the internal temperature sensor and control units to control the self refresh cycle automatically according to the two temperature ranges: Max 40°C and Max 85°C (for Extended), Max 70°C (for Commercial).*
2. *If the EMRS for external TCSR is issued by the controller, this EMRS code for TCSR is ignored.*

84 Power Up Sequence

7. Apply power and attempt to maintain CKE at a high state and all other inputs may be undefined.
   - Apply VDD before or at the same time as VDDQ.
8. Maintain stable power, stable clock and NOP input condition for a minimum of 200us.
9. Issue precharge commands for all banks of the devices.
10. Issue 2 or more auto-refresh commands.
11. Issue a mode register set command to initialize the mode register.
12. Issue a extended mode register set command to define DS or PASR operating type of the device after normal MRS

EMRS cycle is not mandatory and the EMRS command needs to be issued only when DS or PASR is used.

The default state without EMRS command issued is half driver strength and full array refreshed.

The device is now ready for the operation selected by EMRS.

For operating with DS or PASR, set DS or PASR mode in EMRS setting stage.

In order to adjust another mode in the state of DS or PASR mode, additional EMRS set is required but power up sequence is not needed again at this time. In that case, all banks have to be in idle state prior to adjusting EMRS set.

85 Burst Sequence

85.1 Burst Length = 4

Initial Address		Sequential				Interleave			
AI	A0								
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

85.2 Burst Length = 8

Initial Address			Sequential								Interleave							
A2	AI	A0																
0	0	0	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
0	0	1	1	2	3	4	5	6	7	0	1	0	3	2	5	4	7	6
0	1	0	2	3	4	5	6	7	0	1	2	3	0	1	6	7	4	5
0	1	1	3	4	5	6	7	0	1	2	3	2	1	0	7	6	5	4
1	0	0	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3
1	0	1	5	6	7	0	1	2	3	4	5	4	7	6	1	0	3	2
1	1	0	6	7	0	1	2	3	4	5	6	7	4	5	2	3	0	1
1	1	1	7	0	1	2	3	4	5	6	7	6	5	4	3	2	1	0

86 Device Operations

86.1 Addresses of 256Mb

86.1.1. Bank Addresses (BA0-BA1)

x 16

This SDRAM is organized as four independent banks of 4,194,304 words x 16 bits memory arrays. The BA0-BA1 inputs are latched at the time of assertion of RAS# and CAS# to select the bank to be used for the operation. The bank addresses BA0-BA1 are latched at bank active, read, write, mode register set and precharge operations.

x 32

This SDRAM is organized as four independent banks of 2,097,152 words x 32 bits memory arrays. The BA0-BA1 inputs are latched at the time of assertion of RAS# and CAS# to select the bank to be used for the operation. The bank addresses BA0-BA1 are latched at bank active, read, write, mode register set and precharge operations.

86.1.2 Address Inputs (A0-A12)

x 16

The 22 address bits are required to decode the 4,194,304 word locations are multiplexed into 13 address input pins (A0-A12). The 13 bit row addresses are latched along with RAS# and BA0-BA1 during bank activate command. The 9 bit column addresses are latched along with CAS#, WE# and BA0-BA1 during read or write command.

x 32

The 21 address bits are required to decode the 2,097,152 word locations are multiplexed into 12 address input pins (A0-A11). The 12 bit row addresses are latched along with RAS# and BA0 - BA1 during bank activate command. The 9 bit column addresses are latched along with CAS#, WE# and BA0-BA1 during read or write command.

86.2 Addresses of 5l2Mb

86.2.1. Bank Addresses (BA0-BA1)

x 16

This SDRAM is organized as four independent banks of 8,388,608 words x 16 bits memory arrays. The BA0-BA1 inputs are latched at the time of assertion of RAS# and CAS# to select the bank to be used for the operation. The bank addresses BA0-BA1 are latched at bank active, read, write, mode register set and precharge operations.

x 32

This SDRAM is organized as four independent banks of 4,194,304 words x 32 bits memory arrays. The BA0-BA1 inputs are latched at the time of assertion of RAS3# and CAS# to select the bank to be used for the operation. The bank addresses BA0-BA1 are latched at bank active, read, write, mode register set and precharge operations.

86.2.2 Address Inputs (A0-A12)

x 16

The 23 address bits are required to decode the 8,388,608 word locations are multiplexed into 13 address input pins (A0-A12). The 13 bit row addresses are latched along with RAS# and BA0-BA1 during bank activate command. The 10 bit column addresses are latched along with CAS#, WE# and BA0-BA1 during read or write command.

x 32

The 22 address bits are required to decode the 8,388,608 word locations are multiplexed into 13 address input pins (A0-A12). The 13 bit row addresses are latched along with RAS# and BA0-BA1 during bank activate command. The 9 bit column addresses are latched along with CAS#, WE# and BA0-BA1 during read or write command.

86.3 Clock (CLK)

The clock input is used as the reference for all SDRAM operations. All operations are synchronized to the positive going edge of the clock. The clock transitions must be monotonic between V_{IL} and V_{IH}. During operation with CKE high all inputs are assumed to be in a valid state (low or high) for the duration of set-up and hold time around positive edge of the clock in order to function well Q perform and ICC specifications.

86.4 Clock Enable (CKE)

The clock enable (CKE) gates the clock onto SDRAM. If CKE goes low synchronously with clock (set-up and hold time are the same as other inputs), the internal clock is suspended from the next clock cycle and the state of output and burst address is frozen as long as the CKE remains low. All other inputs are ignored from the next clock cycle after CKE goes low. When all banks are in the idle state and CKE goes low synchronously with clock, the SDRAM enters the power down mode from the next clock cycle. The SDRAM remains in the power down mode ignoring the other inputs as long as CKE remains low. The power down exit is synchronous as the internal clock is suspended. When CKE goes high at least *1CLK* + t_{SS} before the high going edge of the clock, then the SDRAM becomes active from the same clock edge accepting all the input commands.

86.5 NOP and Device Deselect

When RAS#, CAS# and WE# are high, the SDRAM performs no operation (NOP). NOP does not initiate any new operation, but is needed to complete operations which require more than single clock cycle like bank activate, burst read, auto refresh, etc. The device deselect is also a NOP and is entered by asserting CS# high. CS# high disables the command decoder so that RAS#, CAS#, WE# and all the address inputs are ignored.

86.6 DQM Operation

The DQM is used to mask input and output operations. It works similar to OE# during read operation and inhibits writing during write operation. The read latency is two cycles from DQM and zero cycle for write, which means DQM masking occurs two cycles later in read cycle and occurs in the same cycle during write cycle. DQM operation is synchronous with the clock. The DQM signal is important during burst interruptions of write with read or precharge in the SDRAM. Due to asynchronous nature of the internal write, the DQM operation is critical to avoid unwanted or incomplete writes when the complete burst write is not required. Please refer to Figure 87.3 also.

86.7 Mode Register Set (MRS)

The mode register stores the data for controlling the various operating modes of SDRAM. It programs the CAS latency, burst type, burst length, test mode and various vendor specific options to make SDRAM useful for variety of different applications. The default value of the mode register is not defined, therefore the mode register must be written after power up to operate the SDRAM. The mode register is written by asserting low on CS#, RAS#, CAS# and WE# (The SDRAM should be in active mode with CKE already high prior to writing the mode register). The state of address pins A0-An and BA0-BA1 in the same cycle as CS#, RAS#, CAS# and WE# going low is the data written in the mode register. Two clock cycles is required to complete the write in the mode register. The mode register contents can be changed using the same command and clock cycle requirements during operation as long as all banks are in the idle state. The mode register is divided into various fields depending on the fields of functions. The burst length field uses A0-A2, burst type uses A3, CAS latency (read latency from column address) use A4-A6, vendor specific options or test mode use A7-A8, A10/AP-An and BA0-BA1. The write burst length is programmed using A9. A7-A8, A10/AP-An and BA0-BA1 must be set to low for normal SDRAM operation. Refer to the table for specific codes for various burst length, burst type and CAS latencies.

86.8 Extended Mode Register Set (EMRS)

The extended mode register stores the data for selecting driver strength and partial self refresh. EMRS cycle is not mandatory and the EMRS command needs to be issued only when DS or PASR is used. The default state without EMRS command issued is half driver strength, and full array refreshed. The extended mode register is written by asserting low on CS#, RAS#, CAS#, WE# and high on BA1,low on BA0 (The SDRAM should be in all bank precharge with CKE already high prior to writing into the extended mode register). The state of address pins A0-A11 in the same cycle as CS#, RAS#, CAS# and WE# going low is written in the extended mode register. Two clock cycles are required to complete the write operation in the extended mode register. The mode register contents can be changed using the same command and clock cycle requirements during operation as long as all banks are in the idle state. A0-A2 are used for partial self refresh, A5-A6 are used for Driver strength, *Low* on BA0 and *High* on BA1 are used for EMRS. All the other address pins except A0-A2, A5-A6, BA1, BA0 must be set to low for proper EMRS operation. Refer to the table for specific codes.

86.9 Bank Activate

The bank activate command is used to select a random row in an idle bank. By asserting low on RAS# and CS# with desired row and bank address, a row access is initiated. The read or write operation can occur after a time delay of $t_{RCD(min)}$ from the time of bank activation. t_{RCD} is an internal timing parameter of SDRAM, therefore it is dependent on operating clock frequency. The minimum number of clock cycles required between bank activate and read or write command should be calculated by dividing $t_{RCD(min)}$ with cycle time of the clock and then rounding off the result to the next higher integer.

The SDRAM has four internal banks in the same chip and shares part of the internal circuitry to reduce chip area, therefore it restricts the activation of four banks simultaneously. Also the noise generated during sensing of each bank of SDRAM is high, requiring some time for power supplies to recover before another bank can be sensed reliably. $t_{RRD(min)}$ specifies the minimum time required between activating different bank. The number of clock cycles required between different bank activation must be calculated similar to t_{RCD} specification. The minimum time required for the bank to be active to initiate sensing and restoring the complete row of dynamic cells is determined by $t_{RAS(min)}$. Every SDRAM bank activate command must satisfy $t_{RAS(min)}$ specification before a precharge command to that active bank can be asserted. The maximum time any bank can be in the active state is determined by $t_{RAS(min)}$. The number of cycles for both $t_{RAS(min)}$ and $t_{RAS(max)}$ can be calculated similar to t_{RCD} specification.

86.10 Burst Read

The burst read command is used to access burst of data on consecutive clock cycles from an active row in an active bank. The burst read command is issued by asserting low on CS# and CAS# with WE# being high on the positive edge of the clock. The bank must be active for at least $t_{RCD(min)}$ before the burst read command is issued. The first output appears in CAS latency number of clock cycles after the issue of burst read command. The burst length, burst sequence and latency from the burst read command is determined by the mode register which is already programmed. The burst read can be initiated on any column address of the active row. The address wraps around if the initial address does not start from a boundary such that number of outputs from each I/O are equal to the burst length programmed in the mode register. The output goes into high-impedance at the end of the burst, unless a new burst read was initiated to keep the data output seamless. The burst read can be terminated by issuing another burst read or burst write in the same bank or the other active bank or a precharge command to the same bank. The burst stop command is valid at every page burst length

86.11 Burst Write

The burst write command is similar to burst read command and is used to write data into the SDRAM on consecutive clock cycles in adjacent addresses depending on burst length and burst sequence. By asserting low on CS#, CAS# and WE# with valid column address, a write burst is initiated. The data inputs are provided for the initial address in the same clock cycle as the burst write command. The input buffer is deselected at the end of the burst length, even though the internal writing can be completed yet. The writing can be completed by issuing a burst read and DQM for blocking data inputs or burst write in the same or another active bank. The burst stop command is valid at every burst length. The write burst can also be terminated by using DQM for blocking data and procreating the bank t_{RDL} after the last data input to be written into the active row. See DQM Operation.

86.12 All Banks Precharge

All banks can be precharged at the same time by using Precharge all command. Asserting low on CS#, RAS#, and WE# with high on A10/AP after all banks have satisfied $t_{RAS(min)}$ requirement, performs precharge on all banks. At the end of t_{RP} after performing precharge to all the banks, all banks are in idle state.

86.13 Precharge

The precharge operation is performed on an active bank by asserting low on CS#, RAS#, WE# and A10/AP with valid BA0-BA1 of the bank to be precharged. The precharge command can be asserted anytime after $t_{RAS(min)}$ is satisfied from the bank active command in the desired bank. t_{RP} is defined as the minimum number of clock cycles required to complete row

precharge is calculated by dividing t_{RP} with clock cycle time and rounding up to the next higher integer. Care should be taken to make sure that burst write is completed or DQM is used to inhibit writing before precharge command is asserted. The maximum time any bank can be active is specified by $t_{RAS(max)}$. Therefore, each bank activate command. At the end of precharge, the bank enters the idle state and is ready to be activated again. Entry to Power down, Auto refresh, Self refresh and Mode register set etc. is possible only when all banks are in idle state.

86.14 Auto Precharge

The precharge operation can also be performed by using auto precharge. The SDRAM internally generates the timing to satisfy $t_{RAS(min)}$ and t_{RP} for the programmed burst length and CAS latency. The auto precharge command is issued at the same time as burst read or burst write by asserting high on A10/AP. If burst read or burst write by asserting high on A10/AP, the bank is left active until a new command is asserted. Once auto precharge command is given, no new commands are possible to that particular bank until the bank achieves idle state.

86.15 Auto Refresh

The storage cells of 64Mb, 128Mb, 256Mb and 512Mb SDRAM need to be refreshed every 64ms to maintain data. An auto refresh cycle accomplishes refresh of a single row of storage cells. The internal counter increments automatically on every auto refresh cycle to refresh all the rows. An auto refresh command is issued by asserting low on CS#, RAS# and CAS# with high on CKE and WE#. The auto refresh command can only be asserted with all banks being in idle state and the device is not in power down mode (CKE is high in the previous cycle). The time required to complete the auto refresh operation is specified by $t_{ARFC(min)}$. The minimum number of clock cycles required can be calculated by driving t_{ARFC} with clock cycle time and them rounding up to the next higher integer. The auto refresh command must be followed by NOP's until the auto refresh operation is completed. All banks will be in the idle state at the end of auto refresh operation. The auto refresh is the preferred refresh mode when the SDRAM is being used for normal data transactions. The 64Mb and 128Mb SDRAM’s auto refresh cycle can be performed once in 15.6µs or a burst of 4096 auto refresh cycles once in 64ms. The 256Mb and 512Mb SDRAM’s auto refresh cycle can be performed once in 7.8µs or a burst of 8192 auto refresh cycles once in 64ms.

86.16 Self Refresh

The self refresh is another refresh mode available in the SDRAM. The self refresh is the preferred refresh mode for data retention and low power operation of SDRAM. In self refresh mode, the SDRAM disables the internal clock and all the input buffers except CKE. The refresh addressing and timing are internally generated to reduce power consumption.

The self refresh mode is entered from all banks idle state by asserting low on CS#, RAS#, CAS# and CKE with high on WE#. Once the self refresh mode is entered, only CKE state being low matters, all the other inputs including the clock are ignored in order to remain in the self refresh mode.

The self refresh is exited by restarting the external clock and then asserting high on CKE. This must be followed by NOP's for a minimum time of t_{SRFX} before the SDRAM reaches idle state to begin normal operation. In case that the system uses burst auto refresh during normal operation, it is recommended to use burst 8192 auto refresh cycles for 256Mb and 512Mb, and burst 4096 auto refresh cycles for 128Mb and 64Mb immediately before entering self refresh mode and after exiting in self refresh mode. On the other hand, if the system uses the distributed auto refresh, the system only has to keep the refresh duty cycle.

87 Basic Feature and Function Descriptions

87.1 Clock Suspend

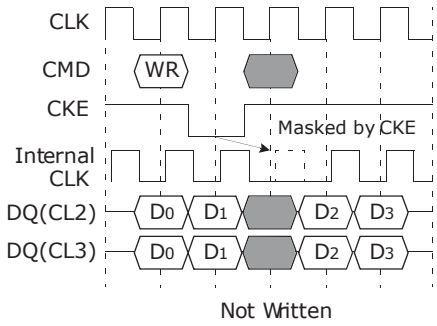


Figure 87.1 Clock Suspend During Write

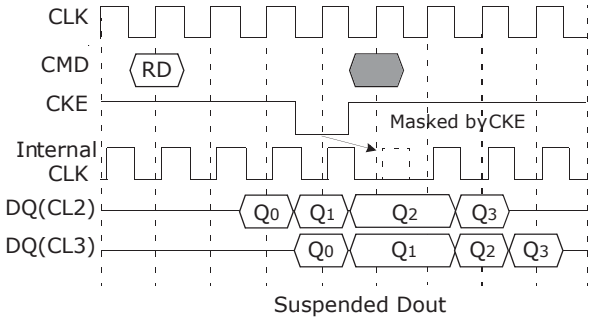


Figure 87.2 Clock Suspend During Read (BL = 4)

87.2 DQM Operation

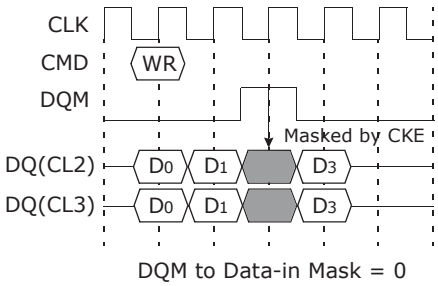


Figure 87.1 Write Mask (BL = 4)

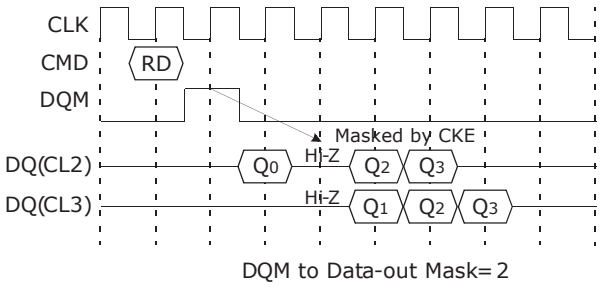
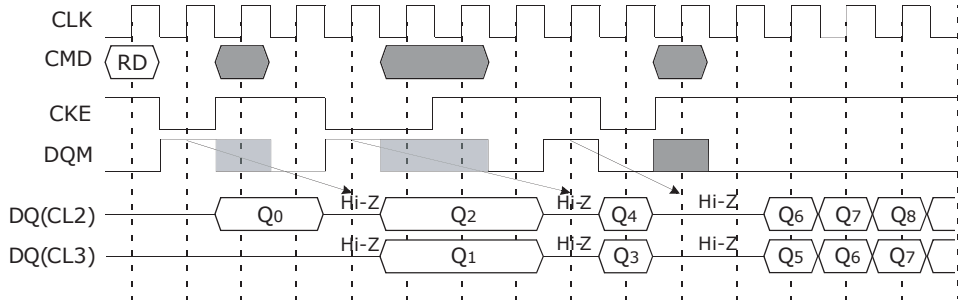


Figure 87.2 Read Mask (BL = 4)

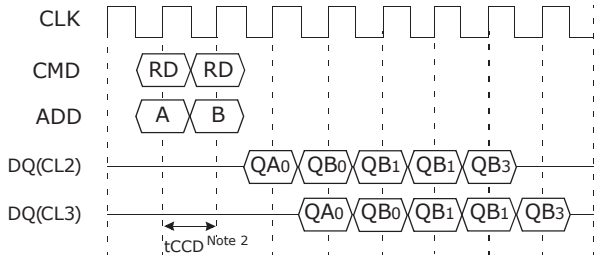


Notes:

1. *CKE to CLK disable/enable = 1CLK.*
2. *DQM makes data out Hi-Z after 2CLKs which should masked by CKE* L.
3. *DQM masks both data-in and data-out.*

Figure 87.3 DQM with CLock Suspended (Full Page Read)

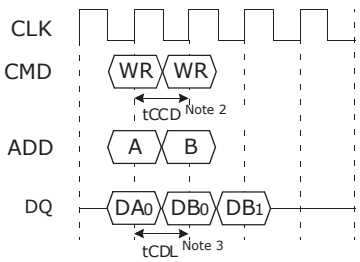
87.3 CAS# Interrupt I



Notes:

1. *By* Interrupt*, It is meant to stop burst read/write by external command before the end of burst. By* CAS# Interrupt*, to stop burst read/write by CAS# access; read and write.*
2. t_{CCD}*: CAS# to CAS# delay (= 1CLK).*
3. t_{CDL}*: Last data in to new column address delay (= 1CLK).*

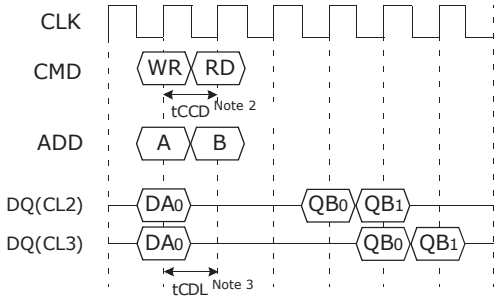
Figure 87.4 Read Interrupted by Read (BL = 4)



Notes:

1. *By* Interrupt*, It is meant to stop burst read/write by external command before the end of burst. By* CAS# Interrupt*, to stop burst read/write by CAS# access; read and write.*
2. t_{CCD}*: CAS# to CAS# delay (= 1CLK).*
3. t_{CDL}*: Last data in to new column address delay (= 1CLK).*

Figure 87.5 Write Interrupted by Write (BL = 2)

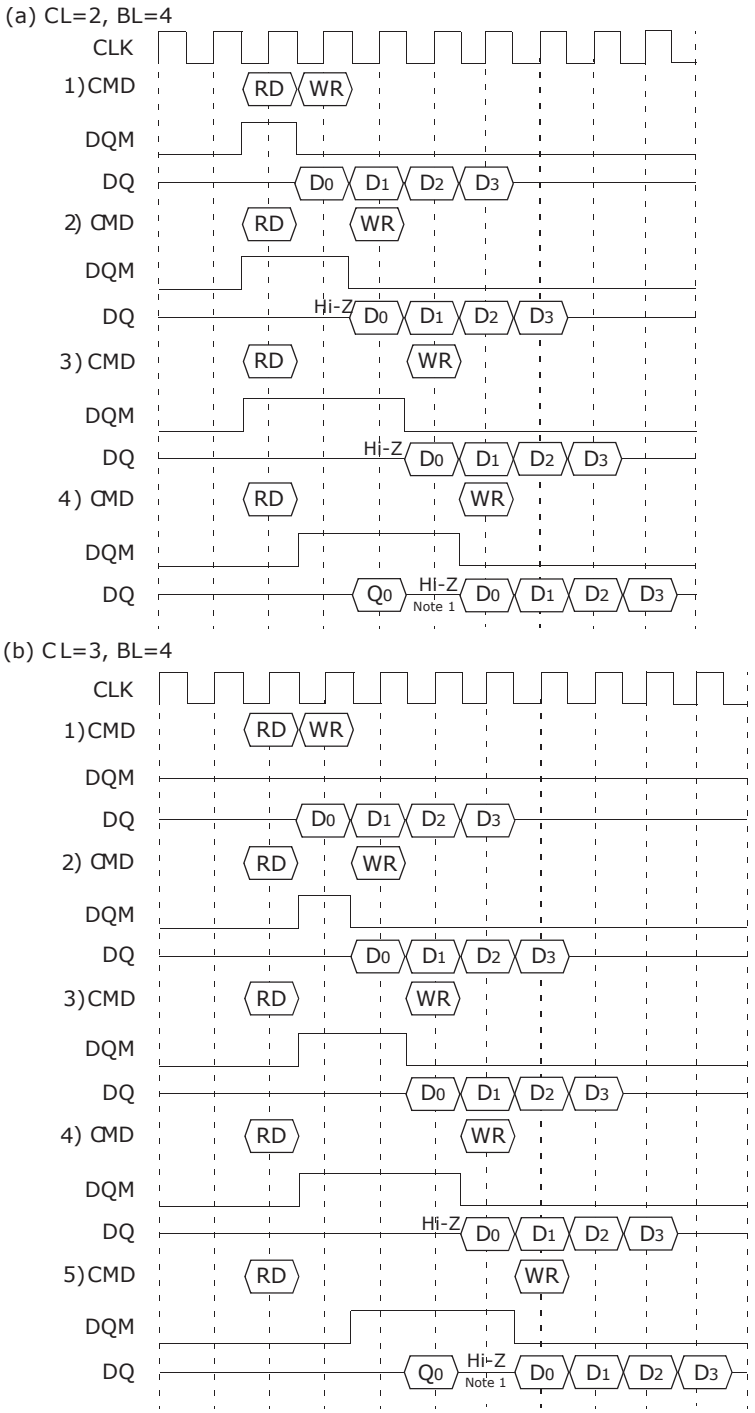


Notes:

1. *By* Interrupt*, It is meant to stop burst read/write by external command before the end of burst. By* CAS# Interrupt*, to stop burst read/write by CAS# access; read and write.*
2. t_{CCD}*: CAS# to CAS# delay (= 1CLK).*
3. t_{CDL}*: Last data in to new column address delay (= 1CLK).*

Figure 87.6 Write Interrupted by Read (BL = 2)

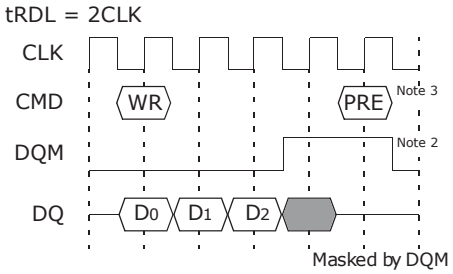
87.4 CAS# Interrupt 2



Notes:

1. To prevent bus contention, there should be at least one gap between data in and data out.

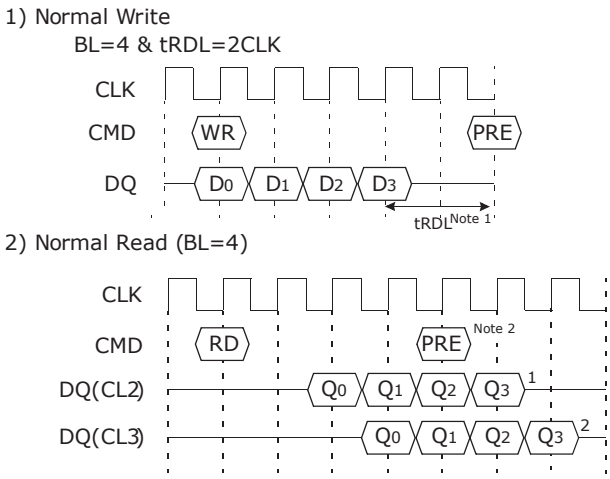
Figure 87.7 Read Interrupted by Write and DQM



Notes:

1. *To prevent bus contention, DQM should be issued which makes at least one gap between data in and data out.*
2. *To inhibit invalid write, DQM should be issued.*
3. *This precharge command and burst write command should be of the same bank, otherwise it is not precharge interrupt but only another bank precharge of four banks operation.*

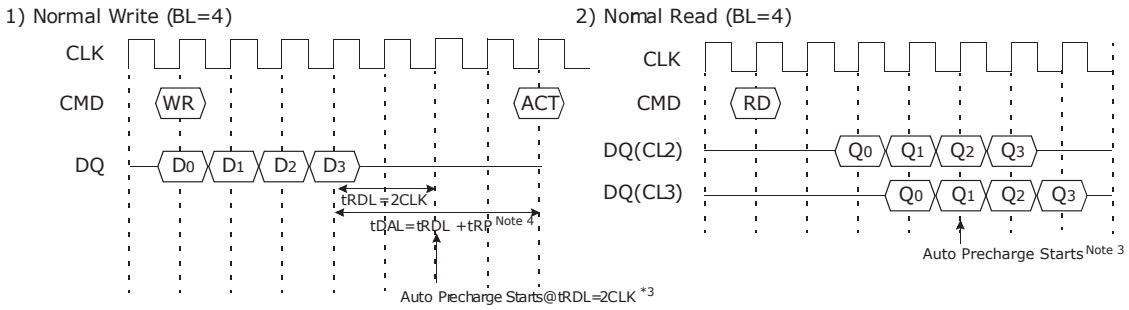
Figure 87.8 Write Interrupted by Precharge and DQM



Notes:

1. *t_{RDL} = 2 CLK is supported.*
2. *Number of valid output data after row precharge: 1, 2 for CAS Latency = 2, 3 respectively.*
3. *The row active command of the precharge bank can be issued after t_{RP} from this point. The new read/write command of other activated bank can be issued from this point. At burst read/write with auto precharge, CAS interrupt of the same bank is illegal.*
4. *t_{DAL} defined Last data in to Active delay. $t_{DAL} = t_{RDL} + t_{RP}$ is supported.*

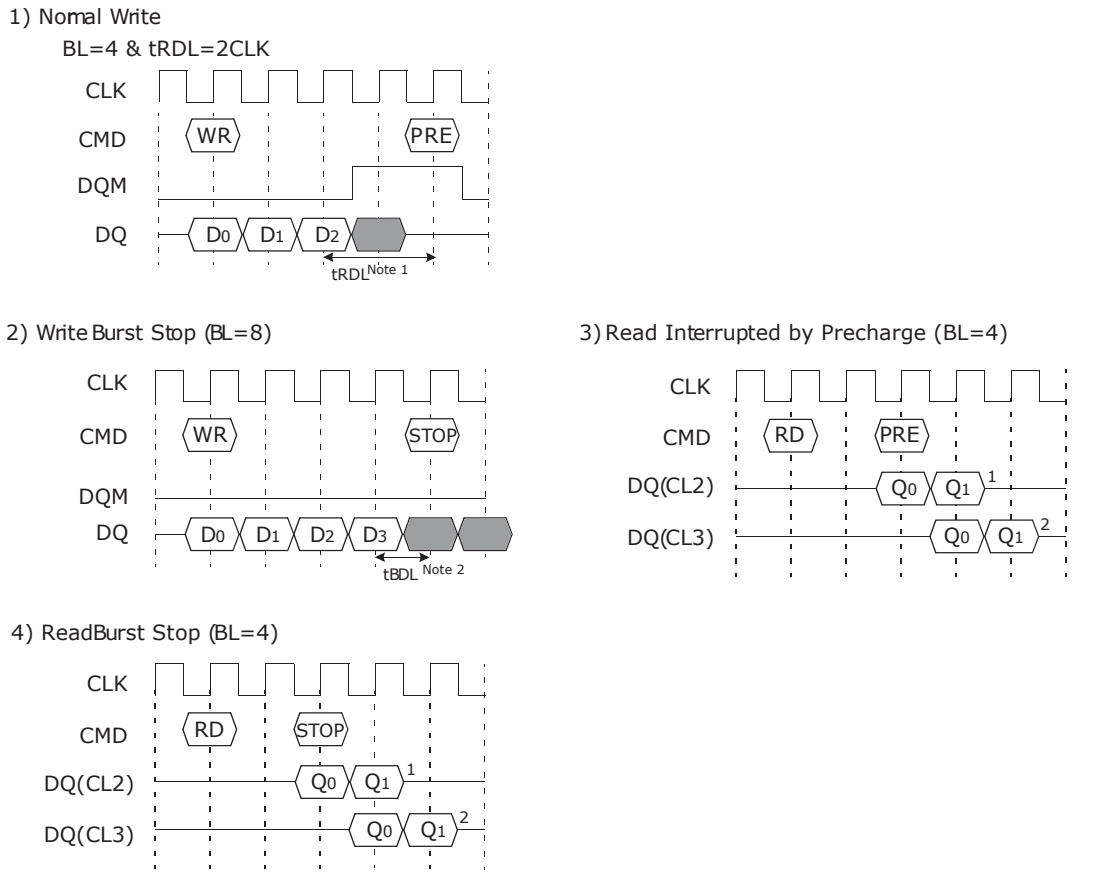
Figure 87.9 Precharge



Notes:

1. *t_{RDL} = 2 CLK is supported.*
2. *Number of valid output data after row precharge: 1, 2 for CAS Latency = 2, 3 respectively.*
3. *The row active command of the precharge bank can be issued after t_{RP} from this point. The new read/write command of other activated bank can be issued from this point. At burst read/write with auto precharge, CAS interrupt of the same bank is illegal.*
4. *t_{DAL} defined Last data in to Active delay. t_{DAL} = t_{RDL} + t_{RP} is supported.*

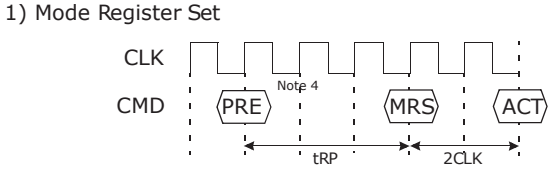
Figure 87.10 Auto Precharge



Notes:

1. *t_{RDL} = 2 CLK is supported.*
2. *t_{BDL}: 1 CLK; Last data in to burst stop delay. Read or write burst stop command is valid at every burst length.*
3. *Number of valid output data after row precharge or burst stop: 1, 2 for CAS latency= 2, 3 respectively.*
4. *PRE: All banks precharge is necessary. MRS can be issued only at all banks precharge state.*

Figure 87.11 Burst Stop and Interrupted by Precharge



Notes:

1. *t_{RDL} = 2 CLK is supported.*
2. *t_{BDL}: 1 CLK; Last data in to burst stop delay. Read or write burst stop command is valid at every burst length.*
3. *Number of valid output data after row precharge or burst stop: 1, 2 for CAS latency= 2, 3 respectively.*
4. *PRE: All banks precharge is necessary. MRS can be issued only at all banks precharge state.*

Figure 87.12 Mode Register Set

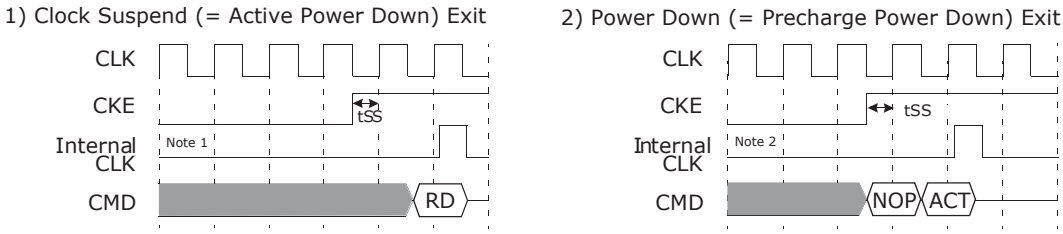


Figure 87.13 Clock Suspend Exit and Power Down Exit

87.5 Auto Refresh

An auto refresh command is issued by having CS#, RAS# and CAS# held low with CKE and WE# high at the rising edge of the clock (CLK). All banks must be precharged and idle for $t_{RP(min)}$ before the auto refresh command is applied. No control of the external address pins is required once this cycle has started because of the internal address counter. When the refresh cycle has completed, all banks will be in the idle state. A delay between the auto refresh command and the next activate command or subsequent auto refresh command must be greater than or equal to the $t_{ARFC(min)}$.

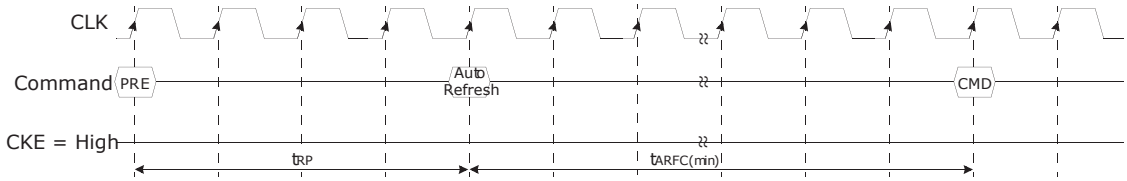


Figure 87.14 Auto Refresh

87.6 Self Refresh

A Self Refresh command is defined by having CS#, RAS#, CAS# and CKE held low with WE# high at the rising edge of the clock. Once the self Refresh command is initiated, CKE must be held low to keep the device in Self Refresh mode. After 1 clock cycle from the self refresh command, all of the external control signals including system clock (CLK) can be disabled except CKE. The clock is internally disabled during Self Refresh operation to reduce power. To exit the Self Refresh mode, supply stable clock input before returning CKE high, assert deselect or NOP command and then assert CKE high. In case that the system uses burst auto refresh during normal operation, it is recommended to use burst auto refresh cycle immediately before entering self refresh mode and after exiting in self refresh mode. On the other hand, if the system uses the distributed auto refresh, the system only has to keep the refresh duty cycle.

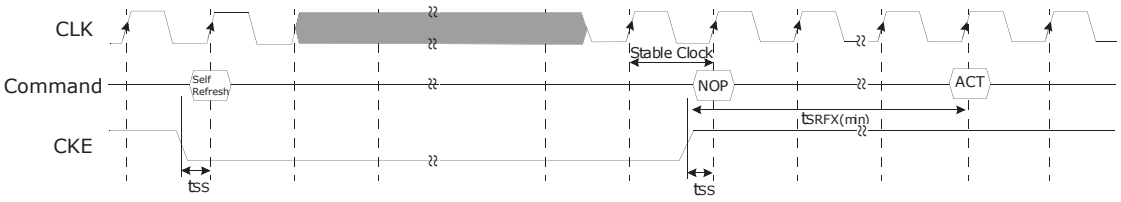


Figure 87.15 Self Refresh

88 Burst Type Control

Basic Mode	Sequential Counting	At MRS A3 = *0*. See the Burst Sequence Table (BL=4, 8) BL=1, 2, 4, 8 and full page.
	Interleave Counting	At MRS A3 = *1*. See the Burst Sequence Table (BL=4, 8) BL=4, 8. At BL=1, 2 Interleave Counting = Sequential Counting.
Random Mode	Random column Access t_{CCD} = 1 CLK	Every cycle Read/Write Command with random column address can realize Random Column Access. That is similar to Extended Data Out (EDO) Operation of conventional DRAM.

89 Burst Length Control

Basic Mode	1	At MRS A2,1,0 = *000*. At auto precharge, t_{RAS} should not be violated.
	2	At MRS A2,1,0 = *001*. At auto precharge, t_{RAS} should not be violated.
	4	At MRS A2,1,0 = *010*.
	8	At MRS A2,1,0 = *011*.
	Full Page	At MRS A2,1,0 = *111*. Wrap around mode (infinite burst length) should be stopped by burst stop. RAS# interrupt or CAS# interrupt.
Special Mode	BRSW	At MRS A9 = *1*. Read burst =1, 2, 4, 8, full page write Burst =1. At auto precharge of write, t_{RAS} should not be violated.
Random Mode	Burst Stop	t_{BDL}= 1, Valid DQ after burst stop is 1, 2 for CAS latency 2, 3 respectively Using burst stop command, any burst length control is possible.
Interrupt Mode	RAS# Interrupt (Interrupted by Precharge)	Before the end of burst, Row precharge command of the same bank stops read/write burst with Row precharge. t_{RDL}= 2 with DQM, valid DQ after burst stop is 1, 2 for CAS latency 2, 3 respectively. During read/write burst with auto precharge, RAS# interrupt can not be issued.
	CAS# Interrupt	Before the end of burst, new read/write stops read/write burst and starts new read/write burst. During read/write burst with auto precharge, CAS# interrupt can not be issued.

90 Function Truth Table I

Current State	CS#	RAS#	CAS#	WE#	BA	Address	Action	Note
IDLE	H	X	X	X	X	X	NOP	
	L	H	H	H	X	X	NOP	
	L	H	H	L	X	X	ILLEGAL	2
	L	H	L	X	BA	CA, A10/AP	ILLEGAL	2
	L	L	H	H	BA	RA	Row (& Bank) Active; Latch RA	
	L	L	H	L	BA	A10/AP	NOP	4
	L	L	L	H	X	X	Auto Refresh or Self Refresh	5
	L	L	L	L	OP code	OP code	Mode Register Access	5
Row Active	H	X	X	X	X	X	NOP	
	L	H	H	H	X	X	NOP	
	L	H	H	L	X	X	ILLEGAL	2
	L	H	L	H	BA	CA, A10/AP	Begin Read; latch CA; determine AP	
	L	H	L	L	BA	CA, A10/AP	Begin Read; latch CA; determine AP	
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A10/AP	Precharge	
	L	L	L	X	X	X	ILLEGAL	
Read	H	X	X	X	X	X	NOP (Continue Burst to End --> Row Active)	
	L	H	H	H	X	X	NOP (Continue Burst to End --> Row Active)	
	L	H	H	L	X	X	Term burst --> Row active	
	L	H	L	H	BA	CA, A10/AP	Term burst, New Read, Determine AP	
	L	H	L	L	BA	CA, A10/AP	Term burst, New Write, Determine AP	3
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A10/AP	Term burst, Precharge timing for Reads	
	L	L	L	X	X	X	ILLEGAL	
Write	H	X	X	X	X	X	NOP (Continue Burst to End --> Row Active)	
	L	H	H	H	X	X	NOP (Continue Burst to End --> Row Active)	
	L	H	H	L	X	X	Term burst --> Row active	
	L	H	L	H	BA	CA, A10/AP	Term burst, New read, Determine AP	3
	L	H	L	L	BA	CA, A10/AP	Term burst, New Write, Determine AP	3
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A10/AP	Term burst, precharge timing for Writes	3
	L	L	L	X	X	X	ILLEGAL	
Read with Auto Precharge	H	X	X	X	X	X	NOP (Continue Burst to End --> Precharge)	
	L	H	H	H	X	X	NOP (Continue Burst to End --> Precharge)	
	L	H	H	L	X	X	ILLEGAL	
	L	H	L	X	BA	CA, A10/AP	ILLEGAL	
	L	L	H	X	BA	RA, RA10	ILLEGAL	2
	L	L	L	X	X	X	ILLEGAL	
Write with Auto Precharge	H	X	X	X	X	X	NOP (Continue Burst to End --> Precharge)	
	L	H	H	H	X	X	NOP (Continue Burst to End --> Precharge)	
	L	H	H	L	X	X	ILLEGAL	
	L	H	L	X	BA	CA, A10/AP	ILLEGAL	
	L	L	H	X	BA	RA, RA10	ILLEGAL	2
	L	L	L	X	X	X	ILLEGAL	
Precharging	H	X	X	X	X	X	NOP --> Idle after t_{RP}	
	L	H	H	H	X	X	NOP --> Idle after t_{RP}	
	L	H	H	L	X	X	ILLEGAL	2
	L	H	L	X	BA	CA	ILLEGAL	2
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A10/AP	NOP --> Idle after t_{RP}	4

Row Activating	L	L	L	X	X	X	ILLEGAL	
	H	X	X	X	X	X	NOP --> Row Active after t_{RCD}	
	L	H	H	H	X	X	NOP --> Row Active after t_{RCD}	
	L	H	H	L	X	X	ILLEGAL	2
	L	H	L	X	BA	CA	ILLEGAL	2
	L	L	H	H	BA	RA	ILLEGAL	2
	L	L	H	L	BA	A10/AP	ILLEGAL	2
	L	L	L	X	X	X	ILLEGAL	
Refreshing	H	X	X	X	X	X	NOP --> Idle after tRC	
	L	H	H	X	X	X	NOP --> Idle after tRC	
	L	H	L	X	X	X	ILLEGAL	
	L	L	H	X	X	X	ILLEGAL	
	L	L	L	X	X	X	ILLEGAL	
Mode Register Accessing	H	X	X	X	X	X	NOP --> Idle after 2 clocks	
	L	H	H	H	X	X	NOP --> Idle after 2 clocks	
	L	H	H	L	X	X	ILLEGAL	
	L	H	L	X	X	X	ILLEGAL	
	L	L	X	X	X	X	ILLEGAL	

Legend: *RA = Row Address, BA = Bank Address, NOP = No Operation Command, CA= Column Address, AP = Auto Precharge*

Notes:

1. *All entries assume the CKE was active (High) during the precharge clock and the current clock cycle.*
2. *Illegal to bank in specified state; Function may be legal in the bank indicated by BA, depending on the state of that bank.*
3. *Must satisfy bus contention, bus turn around, and/or write recovery requirements.*
4. *NOP to bank precharging or in idle state. May precharge bank indicated by BA (and A10/AP).*
5. *Illegal if any bank is not idle.*

9I Function Truth Table 2

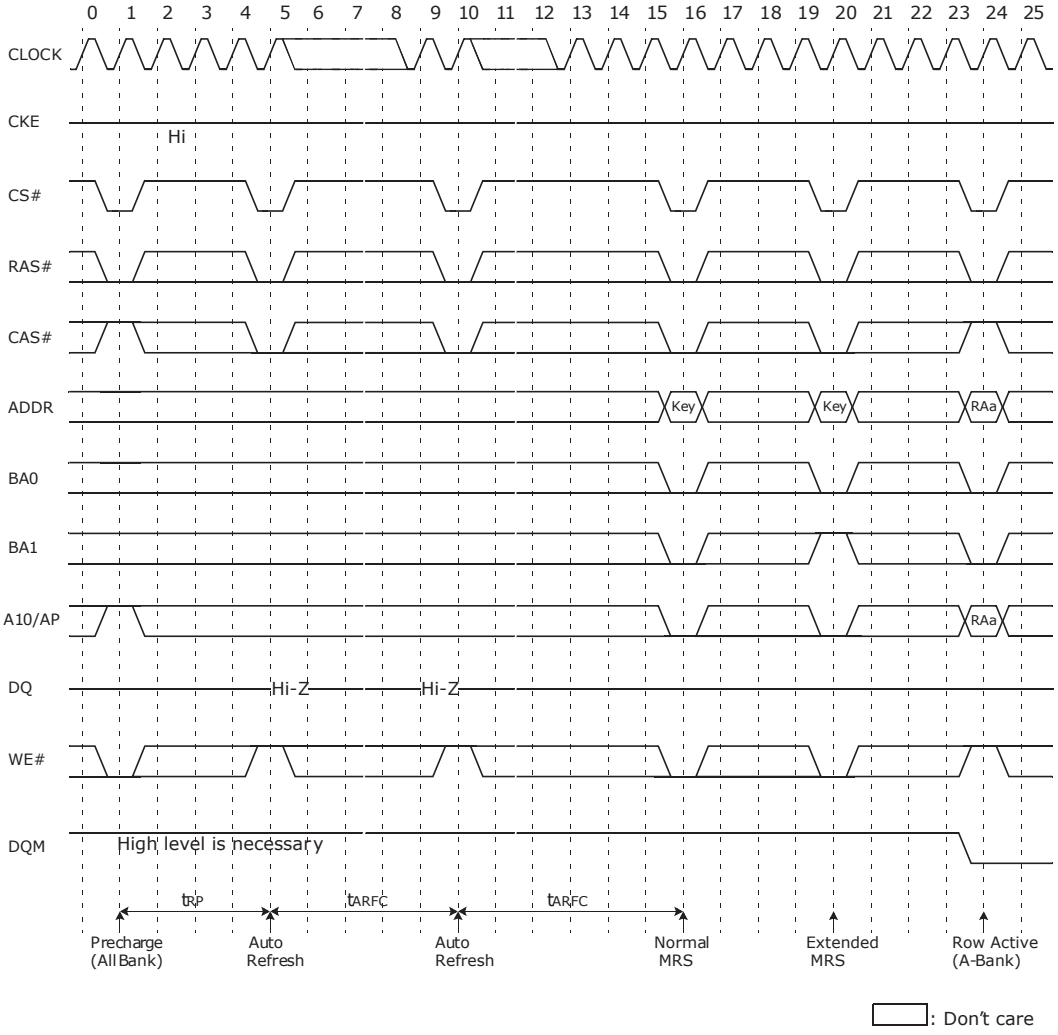
Current State	CKE (n-l)	CKE n	CS#	RAS#	CAS#	WE#	Address	Action	Note
Self Refresh	H	X	X	X	X	X	X	Exit Self Refresh --> Idle after	
	L	H	H	X	X	X	X	Exit Self Refresh --> Idle after t_{sRFX}	1
	L	H	L	H	H	H	X	Exit Self Refresh --> Idle after t_{sRFX}	1
	L	H	L	H	H	L	X	ILLEGAL	
	L	H	L	H	L	X	X	ILLEGAL	
	L	H	L	L	X	X	X	ILLEGAL	
	L	L	X	X	X	X	X	NOP (Maintain Self Refresh)	
All Banks Precharge Power Down	H	X	X	X	X	X	X	INVALID	
	L	H	H	X	X	X	X	Exit Power Down --> ABI	
	L	H	L	H	H	H	X	Exit Power Down --> ABI	2
	L	H	L	H	H	L	X	ILLEGAL	2
	L	H	L	H	L	X	X	ILLEGAL	
	L	H	L	L	X	X	X	ILLEGAL	
	L	L	X	X	X	X	X	NOP (Maintain Low Power Mode)	
All Banks Idle	H	H	X	X	X	X	X	Refer to Function Truth Table 1	
	H	L	H	X	X	X	X	Enter Power Down	
	H	L	L	H	H	H	X	Enter Power Down	3
	H	L	L	H	H	L	X	ILLEGAL	3
	H	L	L	H	L	X	X	ILLEGAL	
	H	L	L	L	H	H	RA	Row (& Bank) Active	
	H	L	L	L	L	H	X	Enter Self Refresh	3
	H	L	L	L	L	L	OP Code	Mode Register Access	
	L	L	X	X	X	X	X	NOP	
Any State other thanListeda bove	H	H	X	X	X	X	X	Refer to Function Truth Table 1	
	H	L	X	X	X	X	X	Begin Clock Suspend next cycle	4
	L	H	X	X	X	X	X	Exit Clock Suspend next cycle	4
	L	L	X	X	X	X	X	Maintain Clock Suspend	

Legend: *ABI = All Banks Idle, RA = Row Address*

Notes:

1. *CKE low to high transition is asynchronous.*
2. *CKE low to high transition is asynchronous if restarts internal clock. A minimum setup time 1CLK + t_{SS} must be satisfied before any command other than exit.*
3. *Power down and self refresh can be entered only from the all banks idle state.*
4. *Must be a legal command.*

92 Timing Diagrams

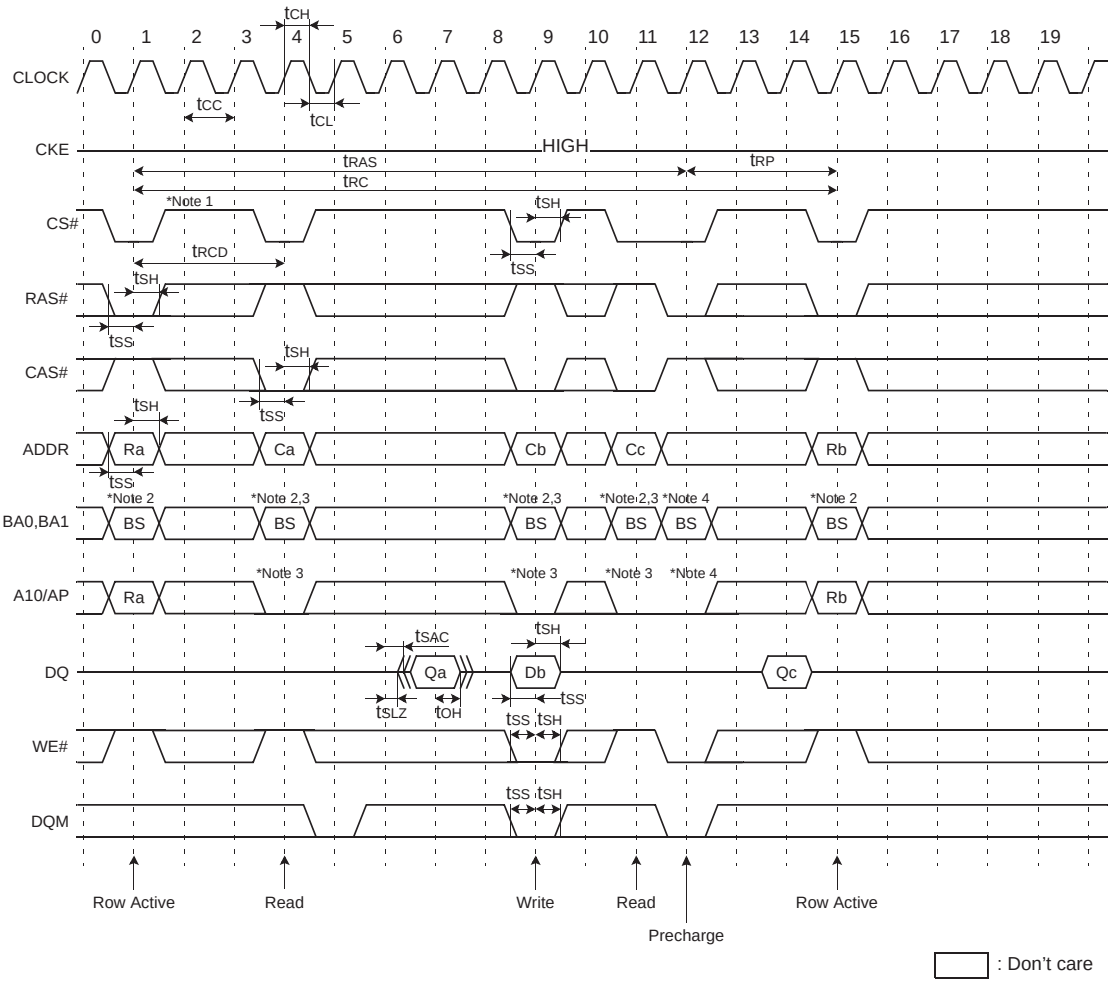


Notes:

1. *Apply power and attempt to maintain CKE at a high state and all other inputs may be undefined.*
   - Apply V_{DD} before or at the same time as V_{DDQ}.
2. *Maintain stable power, stable clock and NOP input condition for a minimum of 200µs.*
3. *Issue precharge commands for all banks of the devices.*
4. *Issue 2 or more auto-refresh commands.*
5. *Issue a mode register set command to initialize the mode register.*
6. *Issue a extended mode register set command to define DS or PASR operating type of the device after normal MRS.*

EMRS cycle is not mandatory and the EMRS command needs to be issued only when DS or PASR is used.
The default state without EMRS command issued is half driver strength and full array refreshed.
The device is now ready for the operation selected by EMRS.
For operating with DS or PASR, set DS or PASR mode in EMRS setting stage.
In order to adjust another mode in the state of DS or PASR mode, additional EMRS set is required but power up sequence is not needed again at this time. In that case, all banks have to be in idle state prior to adjusting EMRS set.

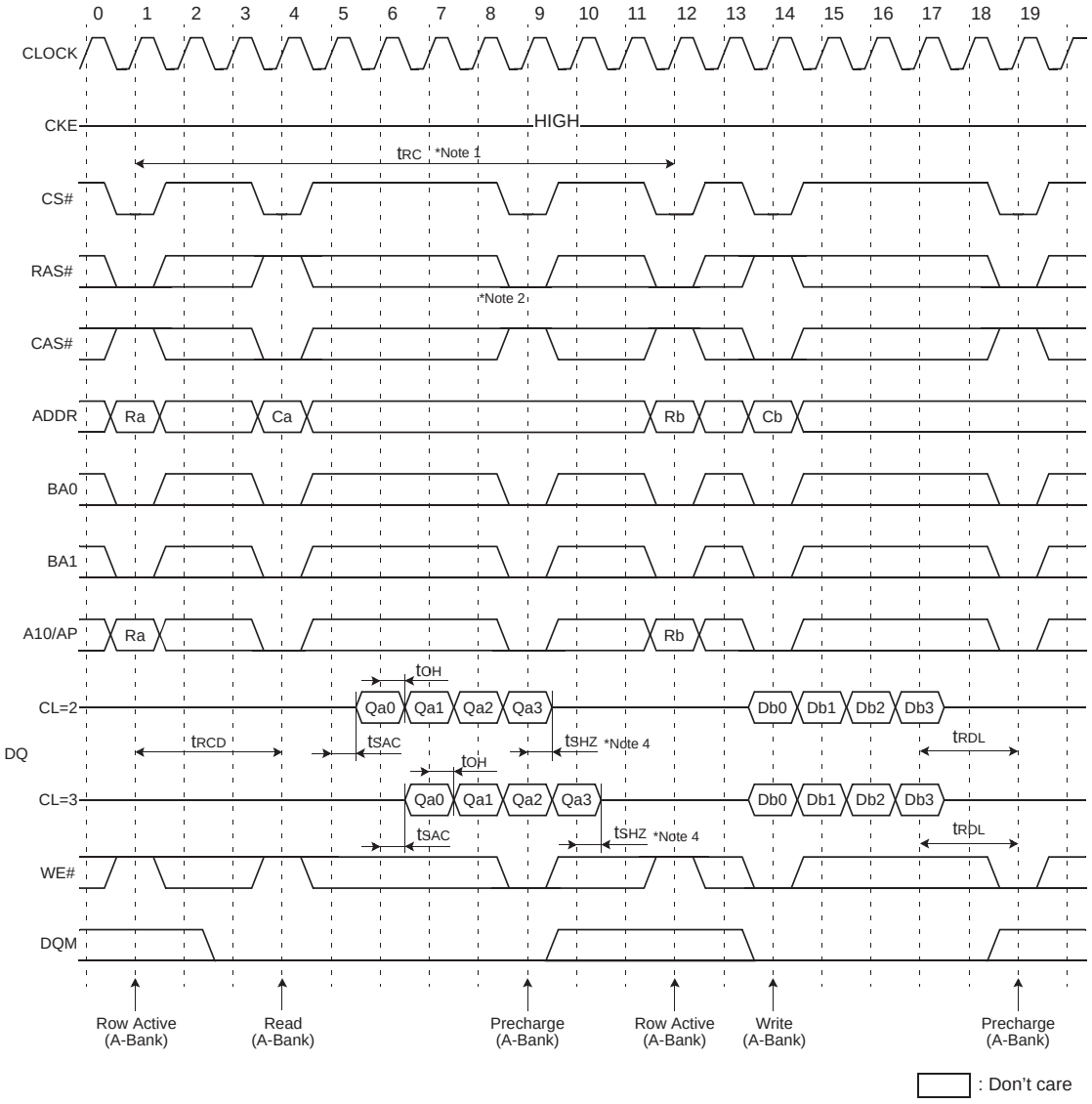
Figure 92.1 Power Up Sequence



Notes:

1. *All input except CKE & DQM can be don't care when CS# is high at the CLK high going edge.*
2. *Bank active & read/write are controlled by BA0, BA1.*

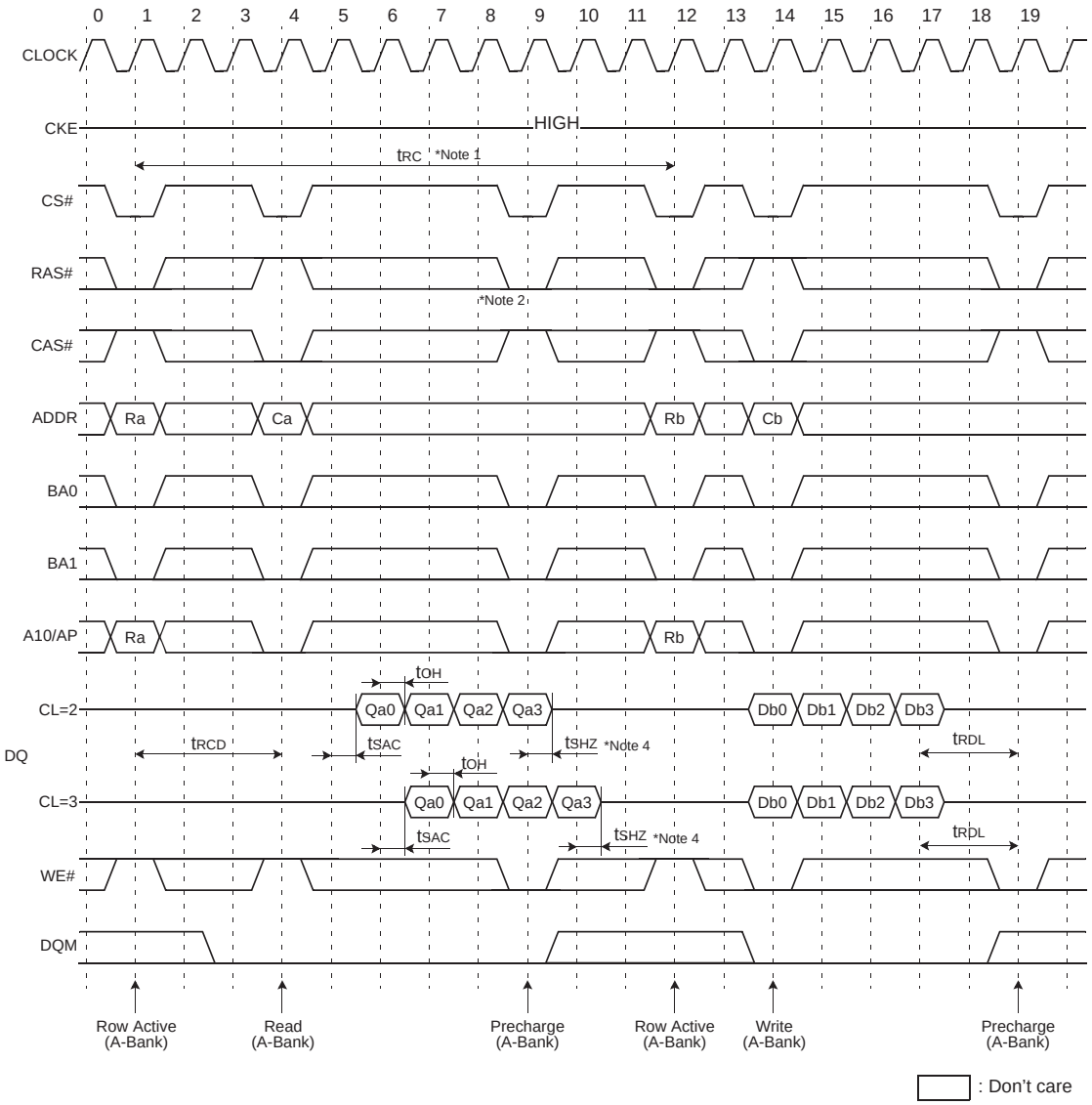
Figure 92.2 Single Bit Read-Write-Read Cycle (Same Page) @CAS Latency=3, Burst Length=l



Notes:

1. *Minimum row cycle times is required to complete internal DRAM operation.*
2. *Row precharge can interrupt burst on any cycle. [CAS Latency - 1] number of valid output data is available after Row precharge. Last valid output will be Hi-Z (t_{SHZ}) after the clock.*
3. *Output will be Hi-Z after the end of burst. (1, 2, 4, 8 & Full page bit burst).*

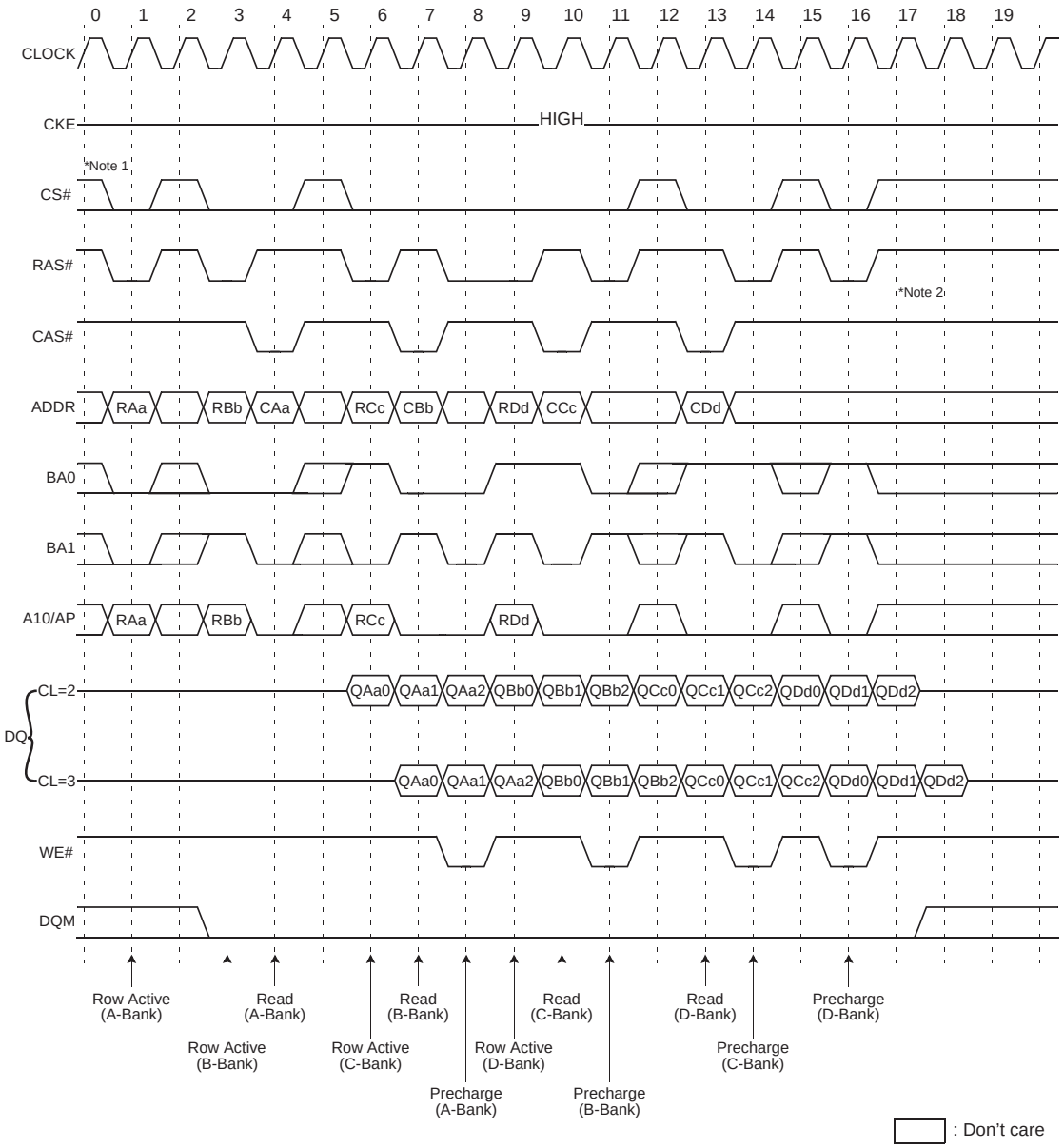
Figure 92.3 Read & Write Cycle at Same Bank @Burst Length=4, t_{RDL}=2CLK



Notes:

1. *To write data before burst read ends, DQM should be asserted three cycle prior to write command to avoid bus contention.*
2. *Row precharge will interrupt writing. Last data input, t_{RDL} before Row precharge, will be written.*
3. *DQM should mask invalid input data on precharge command cycle when asserting precharge before end of burst. Input data after Row precharge cycle will be masked internally.*
4. *t_{DAL}, last data in to active delay, is 2CLK + t_{RP}.*

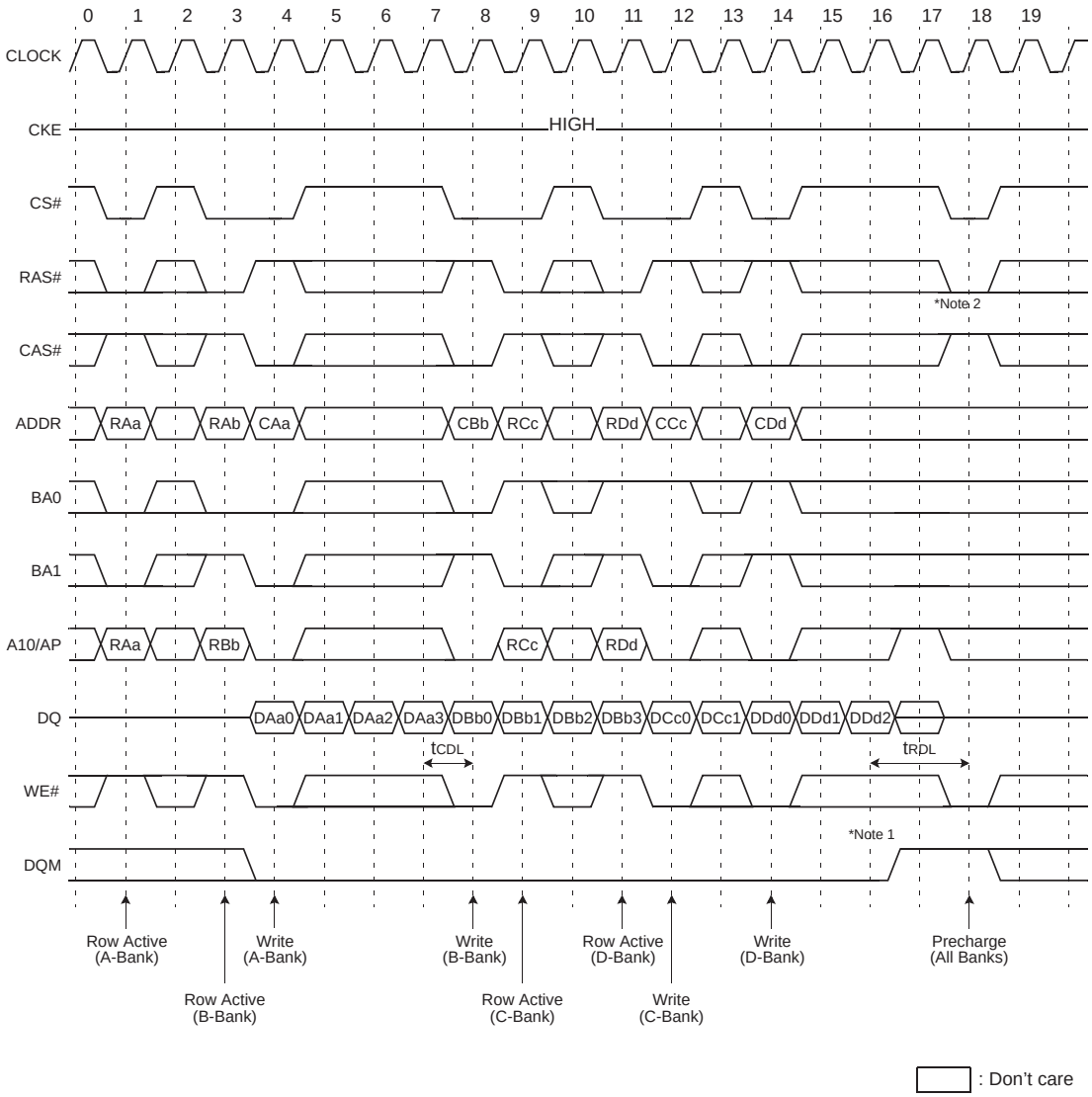
Figure 92.4 Read & Write Cycle at Same Bank @Burst Length=4, t_{RDL}=2CLK



Notes:

1. *CS# can be don't cared when RAS#, CAS# and WE# are high at the clock high going edge.*
2. *To interrupt a burst read by row precharge, both the read and the precharge banks must be the same.*

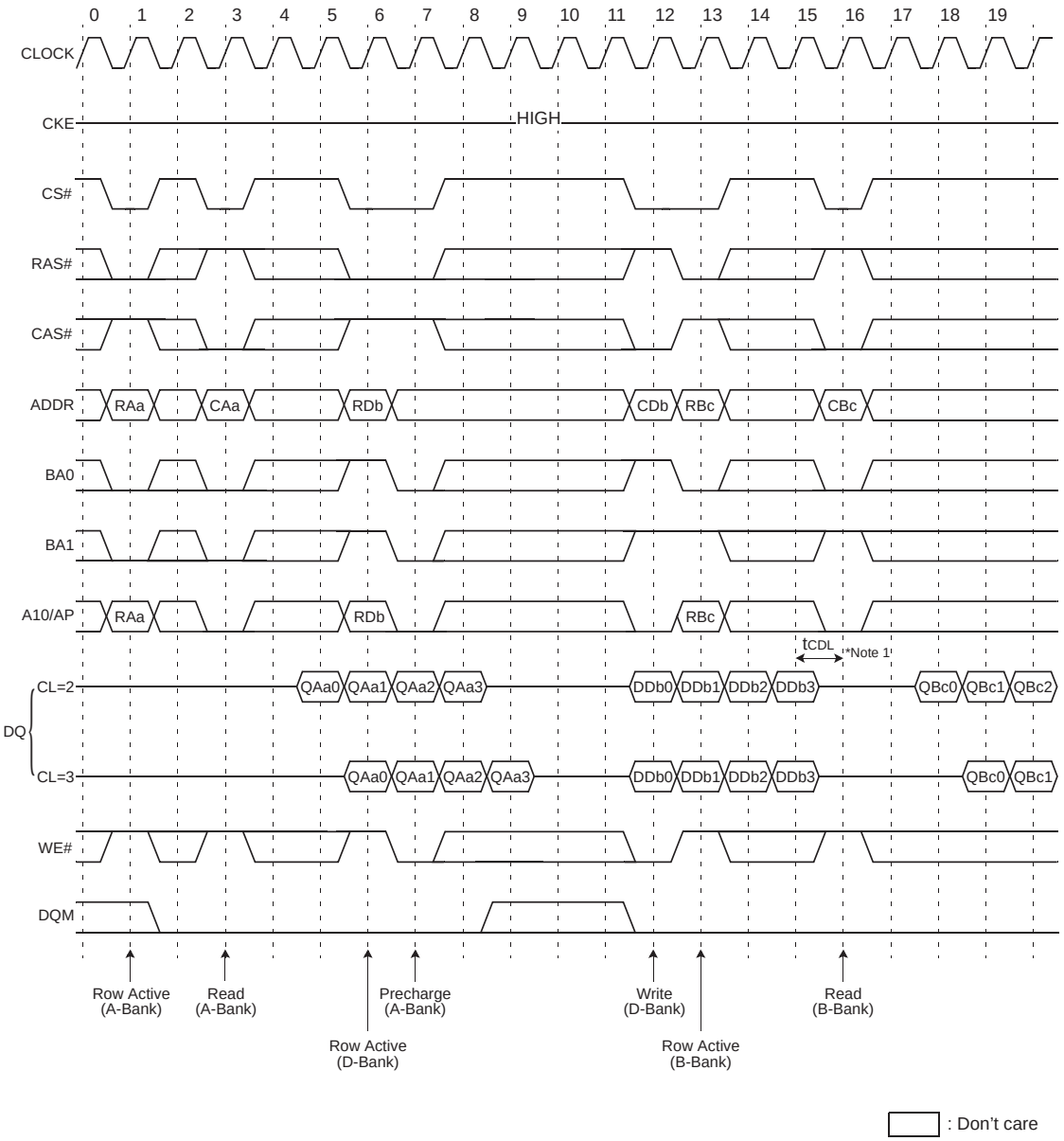
Figure 92.5 Page Read Cycle at Different Bank @Burst Length=4



Notes:

1. *To interrupt burst write by Row precharge, DQM should be asserted to mask invalid input data.*
2. *To interrupt burst write by Row precharge, both the write and the precharge banks must be the same.*

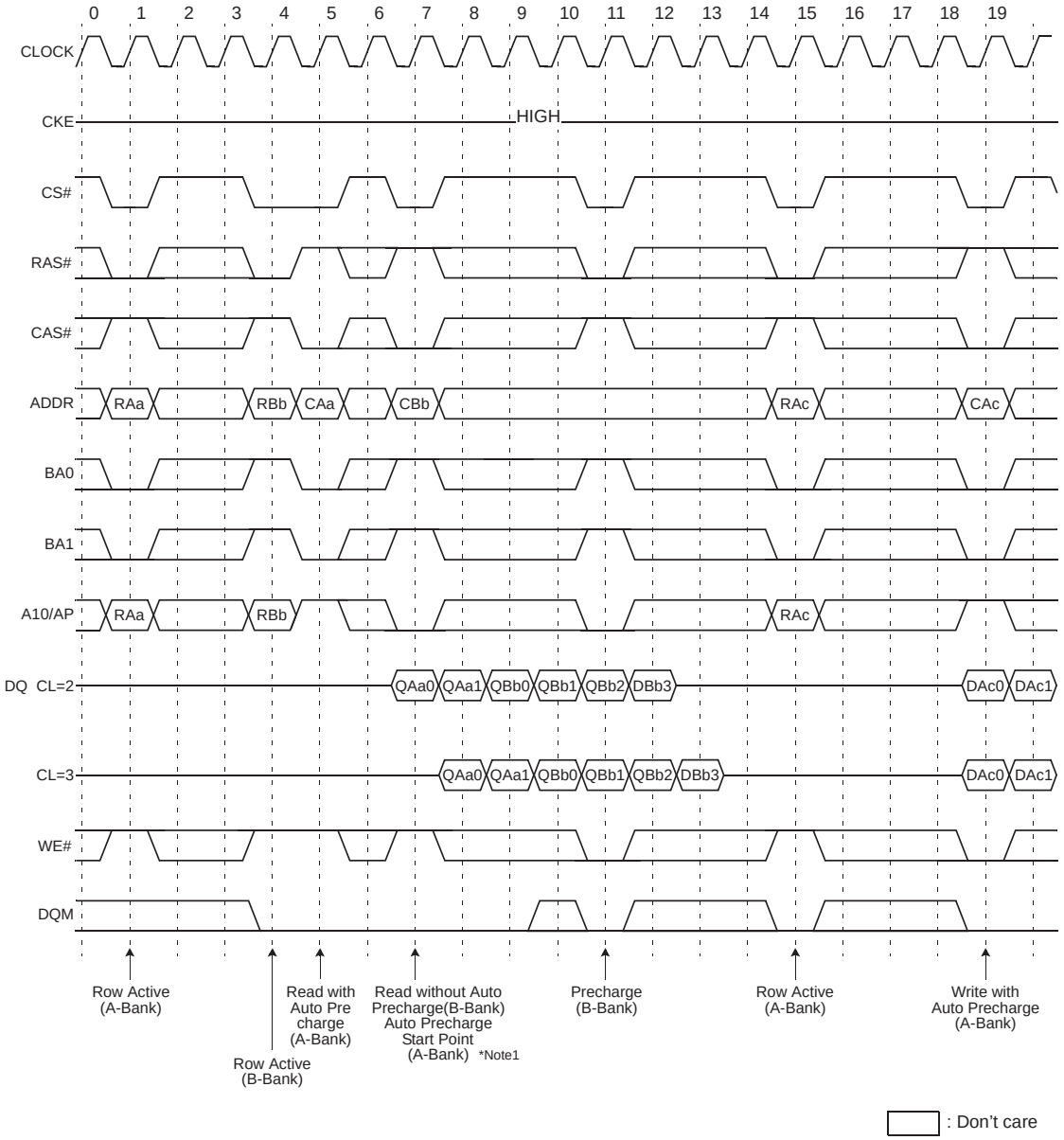
Figure 92.6 Page Write Cycle at Different Bank @Burst Length=4, t_{RDL}=2CLK



Notes:

1. *t_{CDL} should be met to complete write.*

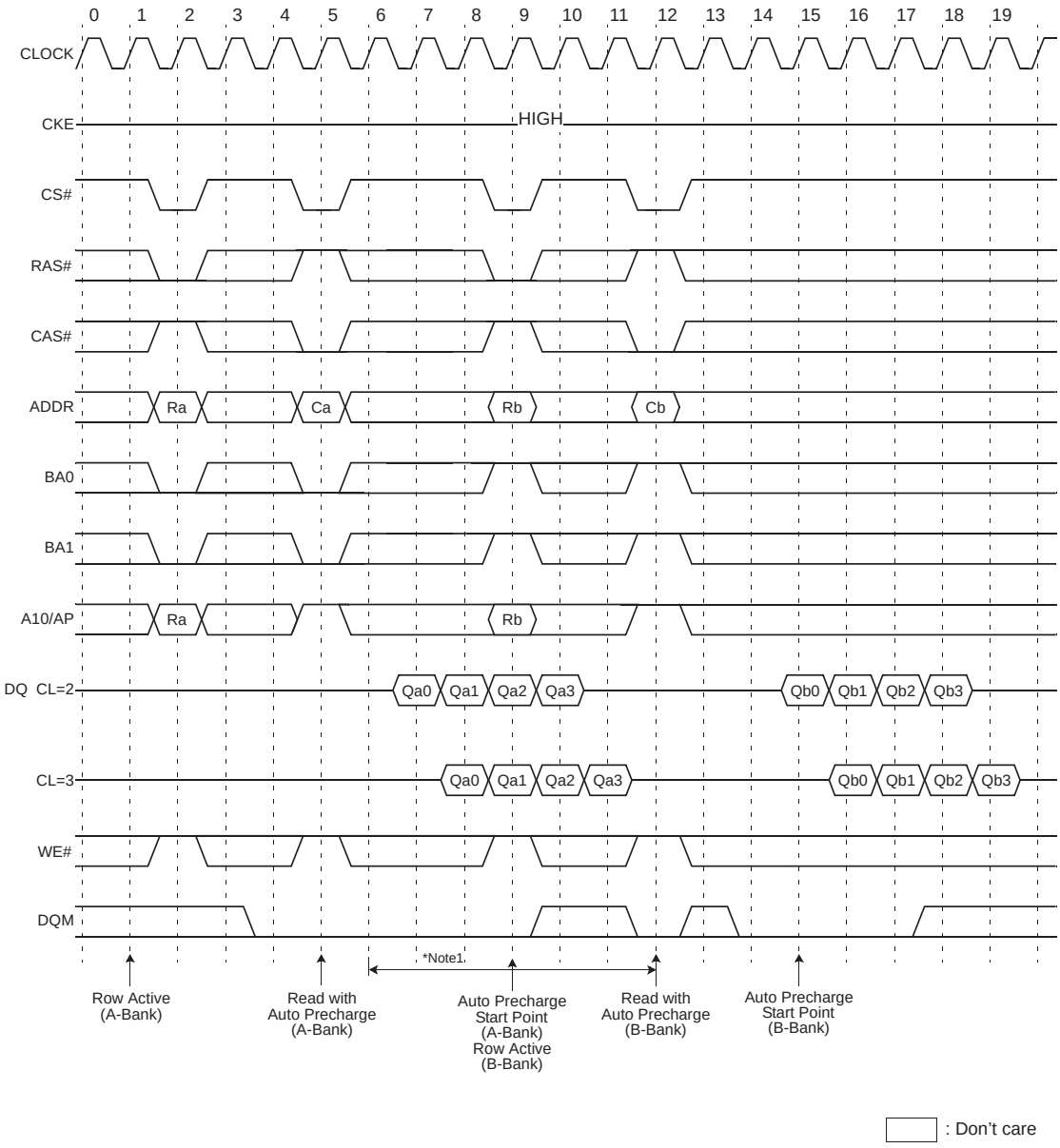
Figure 92.7 Read & Write Cycle at Different Bank @Burst Length=4



Notes:

1. *When Read (Write) command with auto precharge is issued at A-Bank after A and B Bank activation. If Read (Write) command without auto precharge is issued at B-Bank before A-Bank auto precharge starts, A-Bank auto precharge will start at B-Bank read command input point.*
   Any command can not be issued at A-Bank during t_{RP} after A-Bank auto precharge starts.

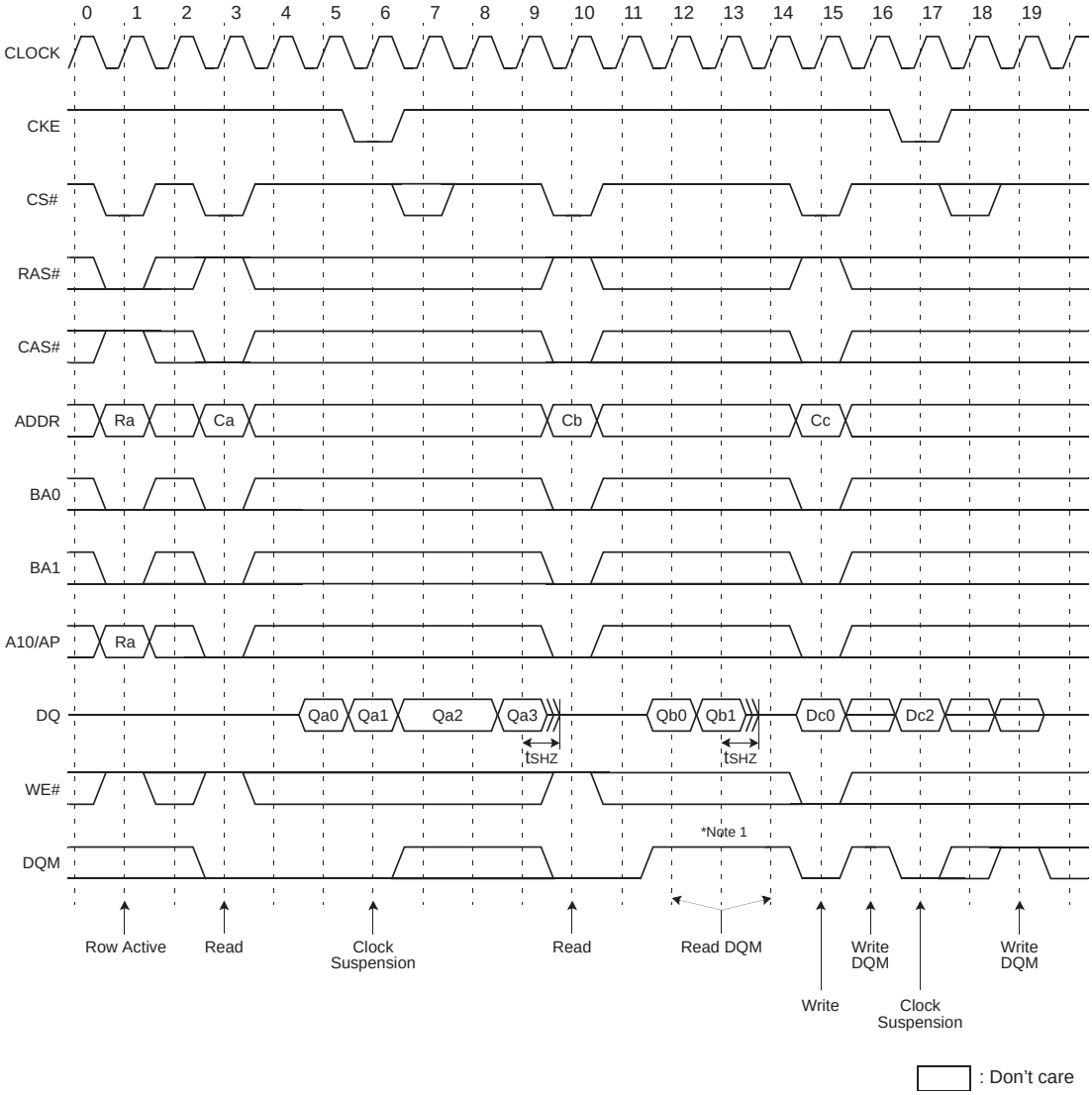
Figure 92.8 Read & Write Cycle with Auto Precharge I @Burst Length=4



Notes:

1. *Any command to A-bank is not allowed in this period. t_{RP} is determined from at auto precharge start point.*

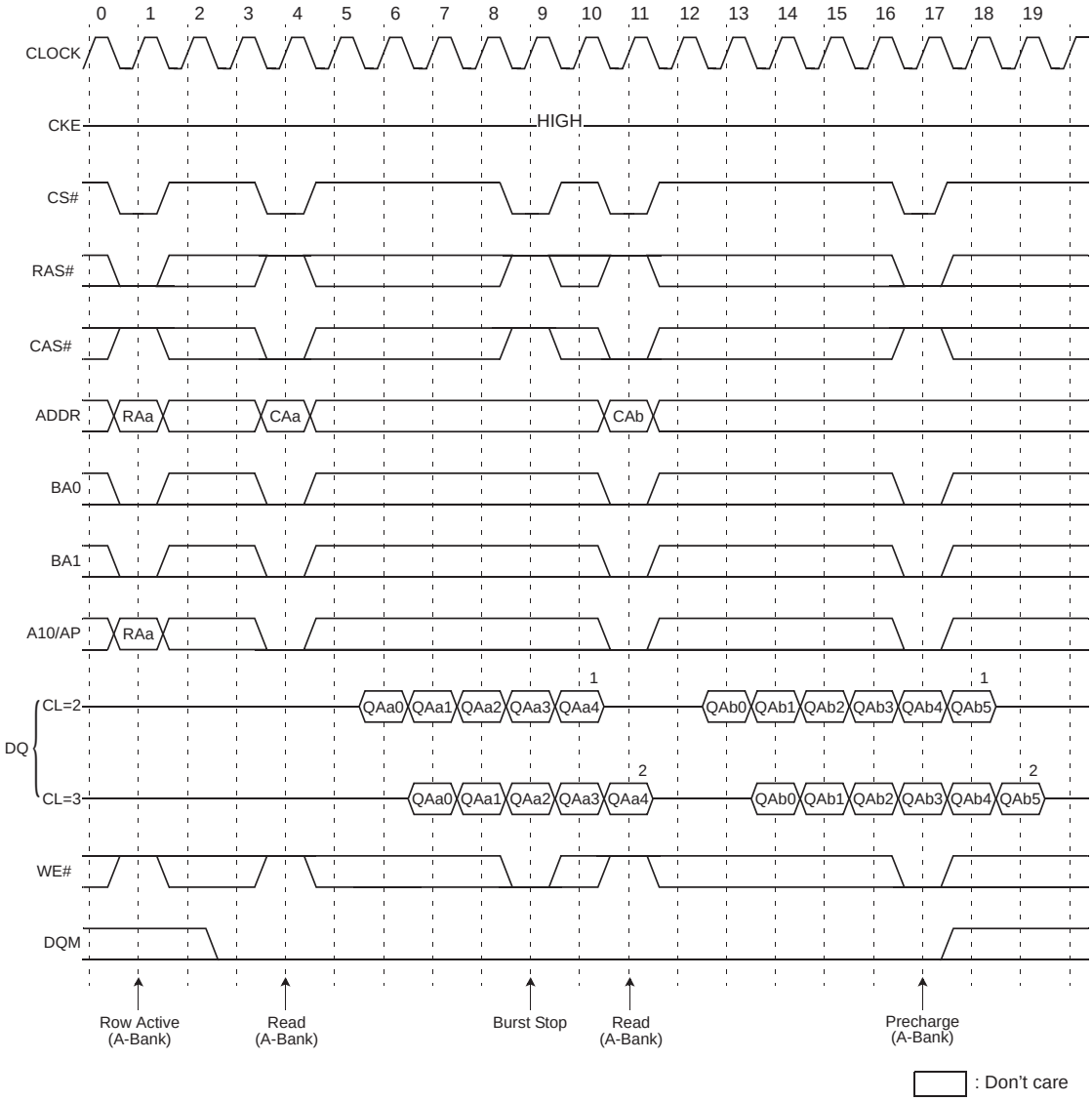
Figure 92.9 Read & Write Cycle with Auto Precharge 2 @Burst Length=4



Notes:

1. DQM is needed to prevent bus contention.

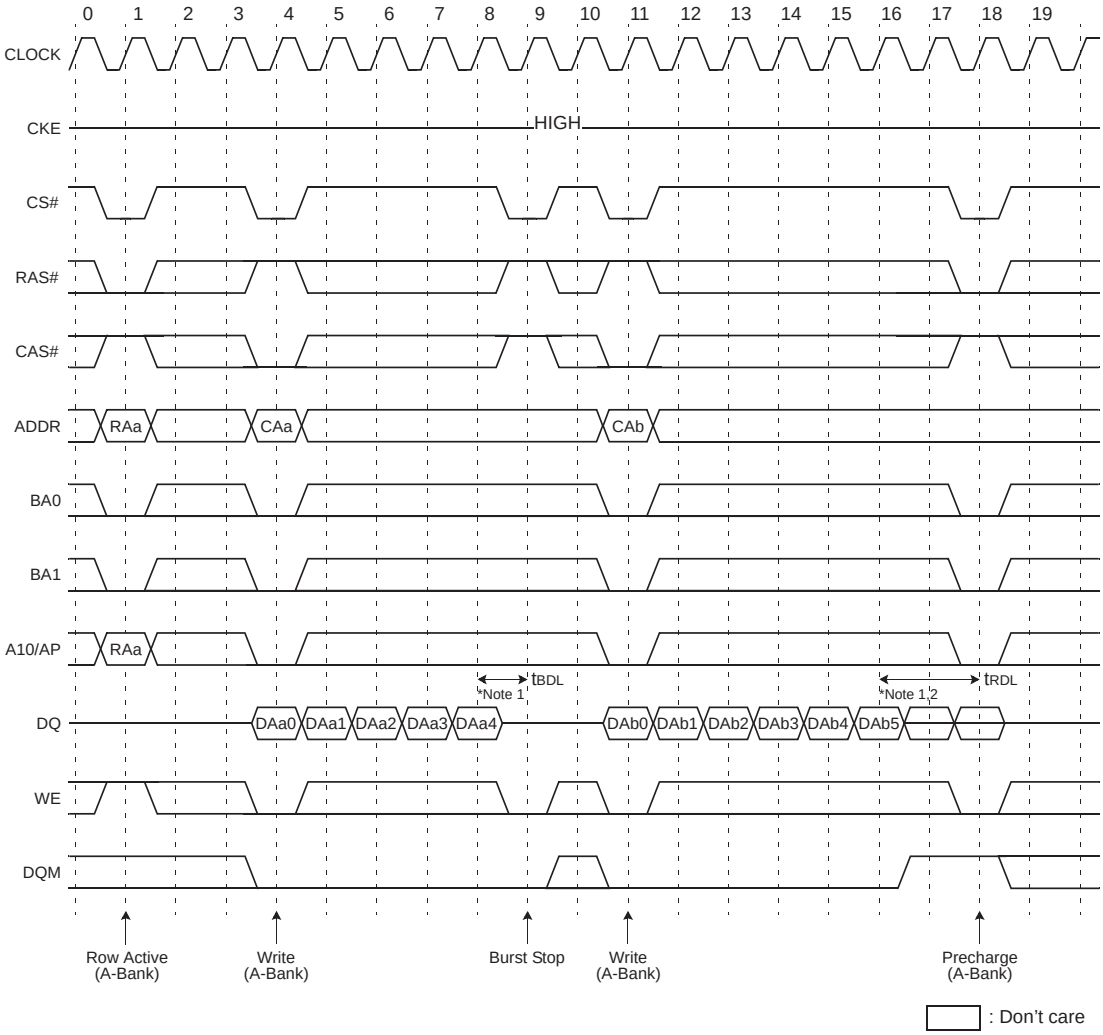
Figure 92.10 Clock Suspension & DQM Operation Cycle @CAS Latency=2, Burst Length=4



Notes:

1. *At full page mode, burst is finished by burst stop or precharge.*
2. *About the valid DQs after burst stop, it is same as the case of RAS# interrupt. Both cases are illustrated above timing diagram. See the label 1, 2 on them. But at burst write, Burst stop and RAS# interrupt should be compared carefully.*
3. *Burst stop is valid at every burst length.*

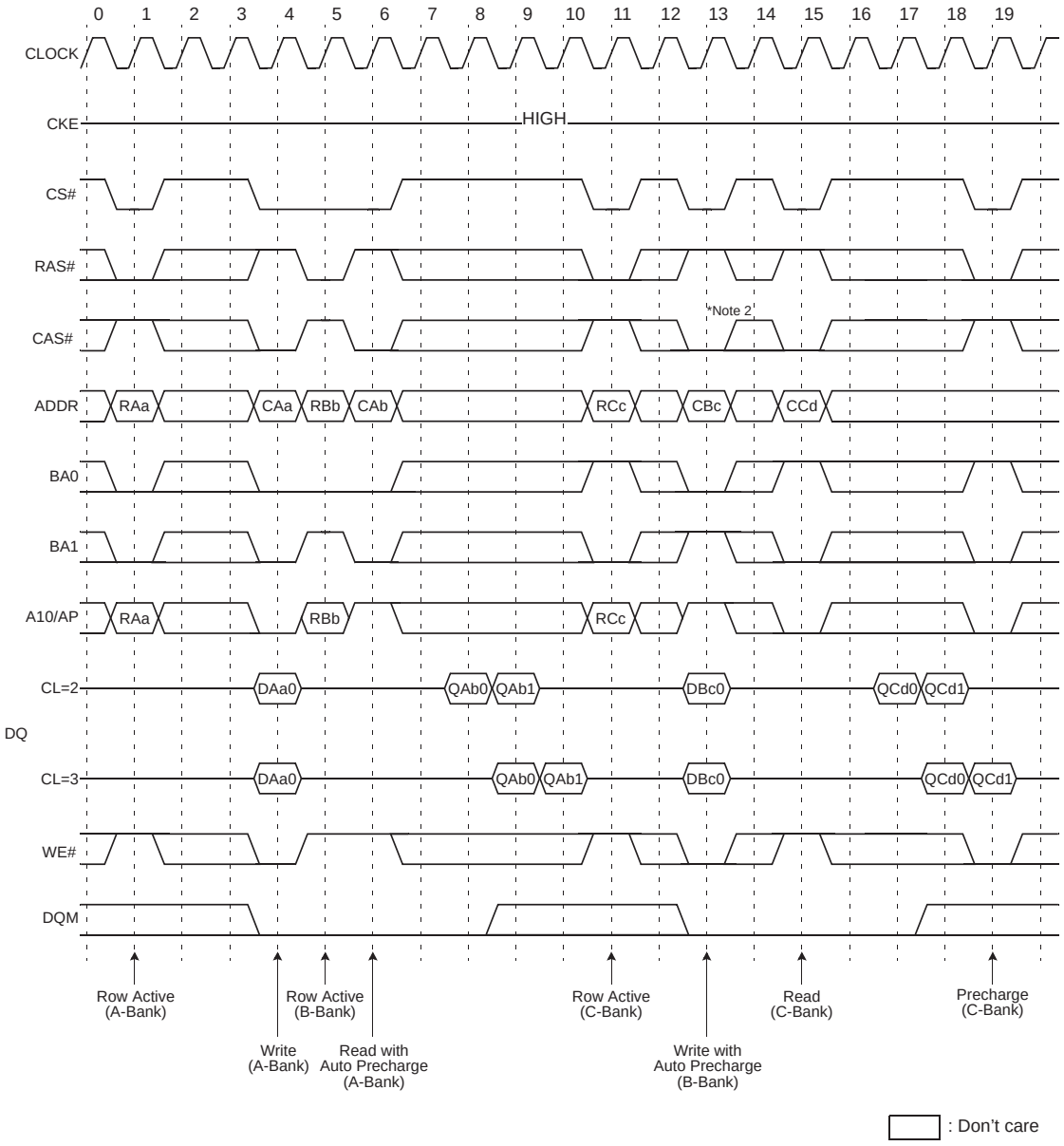
Figure 92.11 Read Interrupted by Precharge Command & Read Burst Stop Cycle @Full Page Burst



Notes:

1. *At full page mode, burst is finished by burst stop or precharge.*
2. *Data-in at the cycle of interrupted by precharge can not be written into the corresponding memory cell. It is defined by AC parameter of t_{RDL}. DQM at write interrupted by precharge command is needed to prevent invalid write. DQM should mask invalid input data on precharge command cycle when asserting precharge before end of burst. Input data after Row precharge cycle will be masked internally.*
3. *Burst stop is valid at every burst length.*

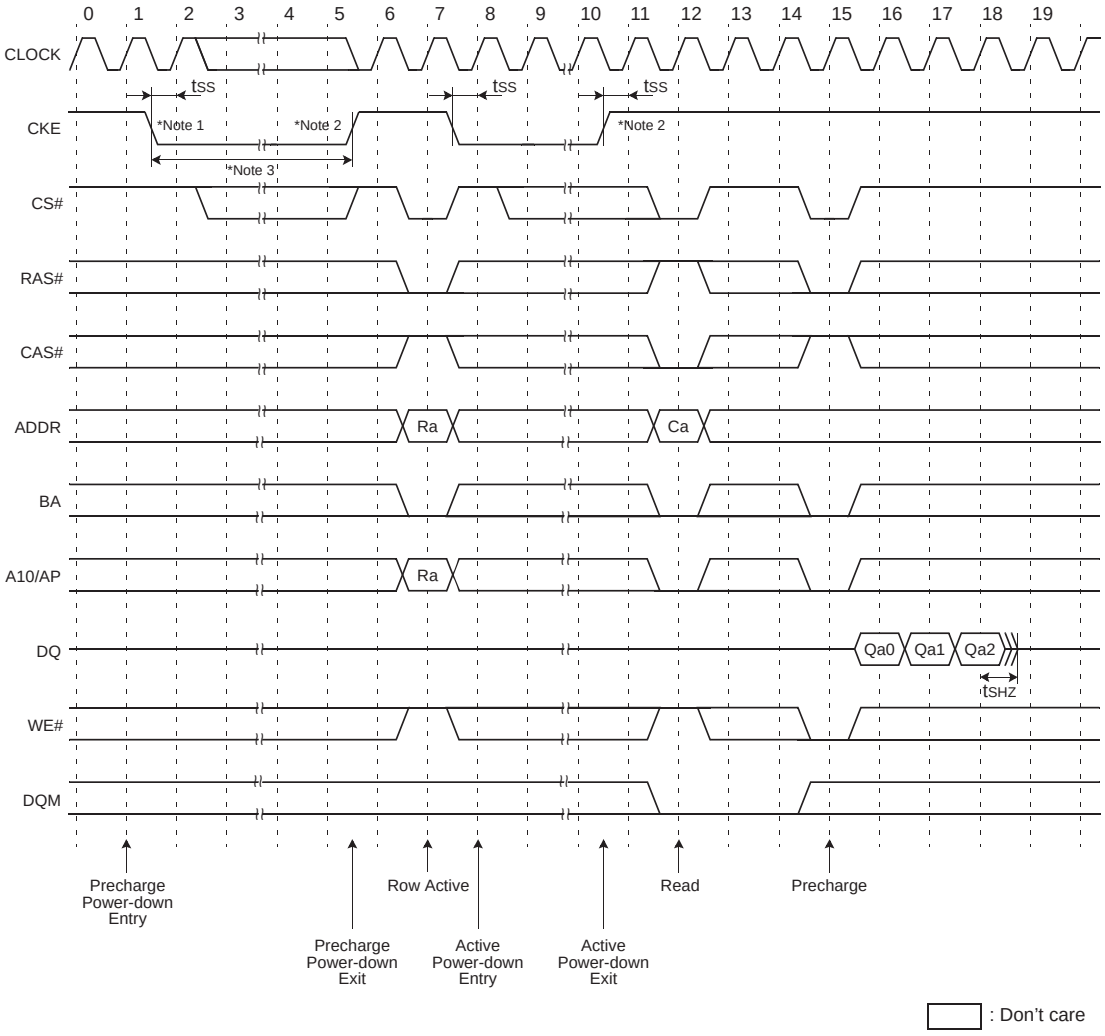
Figure 92.12 Write Interrupted by Precharge Command & Write Burst Stop Cycle @ Full Page Burst, tRDL=2CLK



Notes:

1. *BRSW modes is enabled by setting A9* High *at MRS (Mode Register Set). At the BRSW Mode, the burst length at write is fixed to* 1 *regardless of programmed burst length.*
2. *When BRSW write command with auto precharge is executed, keep it in mind that t_{RAS} should not be violated. Auto precharge is executed at the burst-end cycle, so in the case of BRSW write command, the next cycle starts the precharge.*

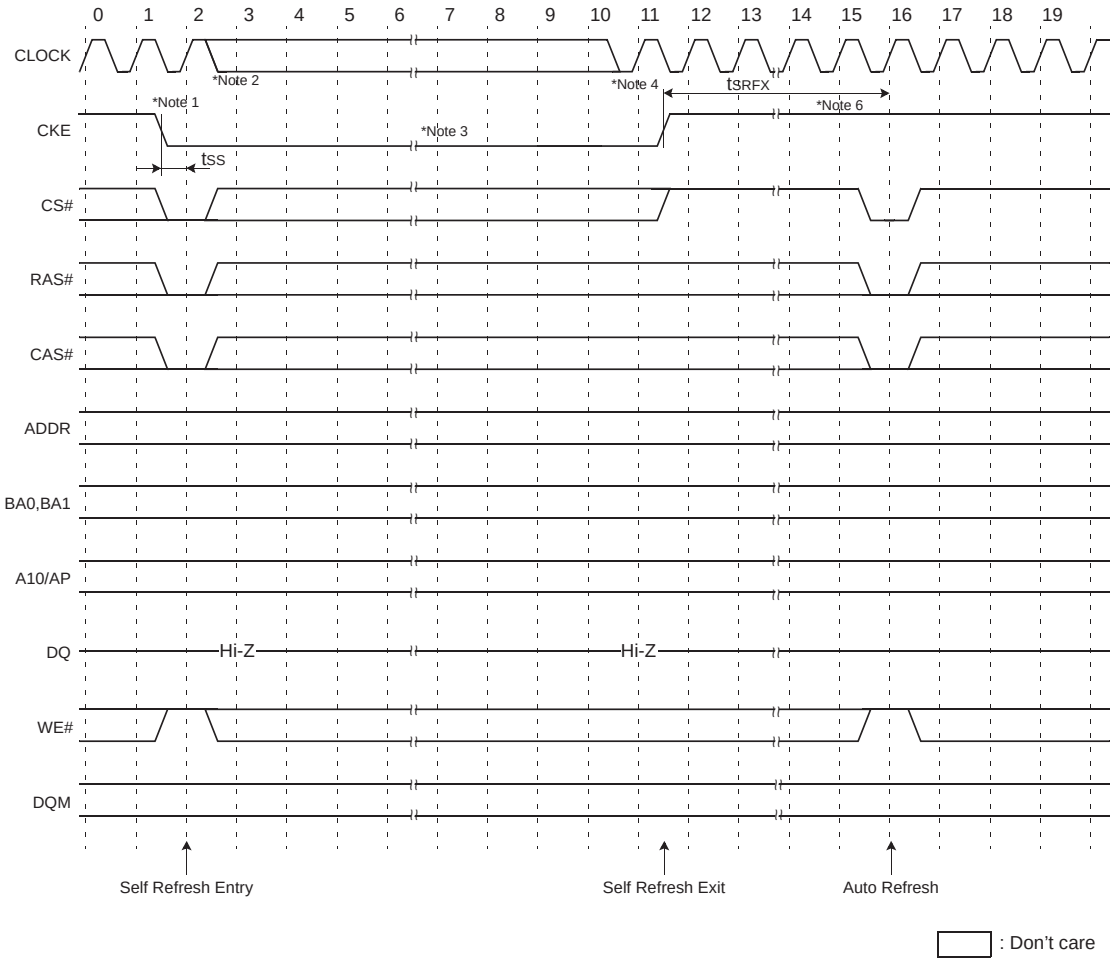
Figure 92.13 Burst Read Single bit Write Cycle @Burst Length=2



Notes:

1. *All banks should be in idle state prior to entering precharge power down mode.*
2. *CKE should be set high at least 1CLK + t_{SS} prior to Row active command.*
3. *Can not violate minimum refresh specification (64ms).*

Figure 92.14 Active/Precharge Power Down Mode @CAS Latency=2, Burst Length=4



Notes:

To Enter Self Refresh Mode

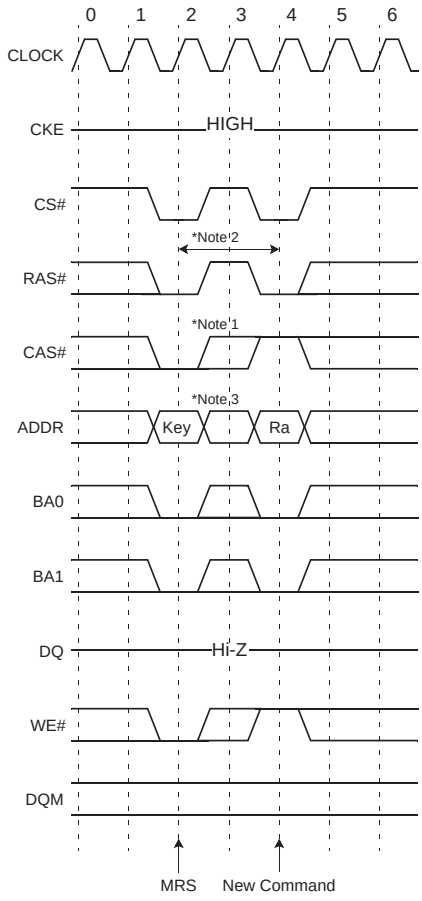
1. *CS#, RAS# & CAS# with CKE should be low at the same clock cycle.*
2. *After 1 clock cycle, all the inputs including the system clock can be don't care except for CKE.*
3. *The device remains in self refresh mode as long as CKE stays* Low.
   Once the device enters self refresh mode, minimum t_{RAS} is required before exit from self refresh.

To Exit Self Refresh Mode

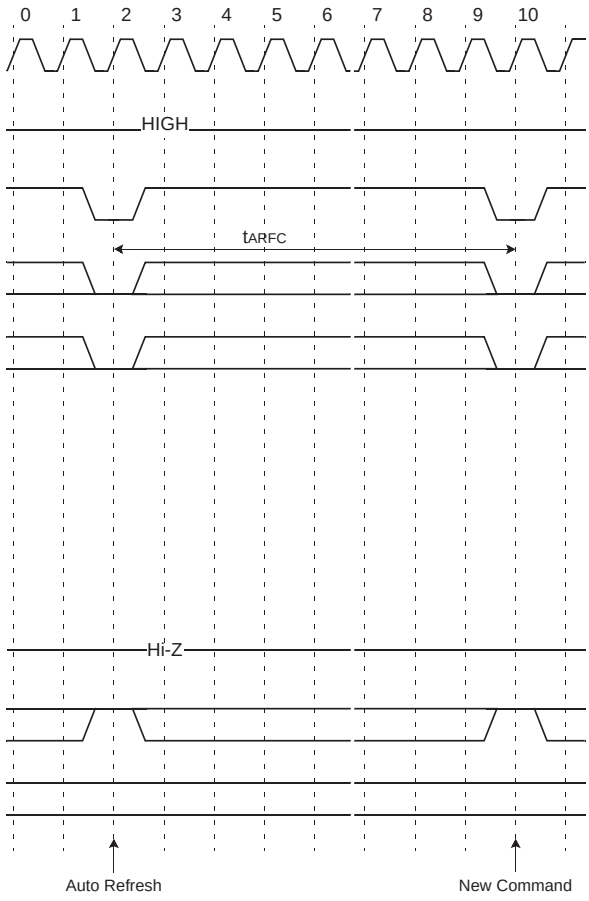
4. *System clock restart and be stable before returning CKE high.*
5. *CS# starts from high.*
6. *Minimum t_{SRFX} is required after CKE going high to complete self refresh exit.*
7. *4K cycle (64Mb, 128Mb) or 8K cycle (256Mb, 512Mb) of burst auto refresh is required before self refresh entry and after self refresh exit if the system uses burst refresh.*

Figure 92.15 Self Refresh Entry & Exit Cycle

Mode Register Set Cycle



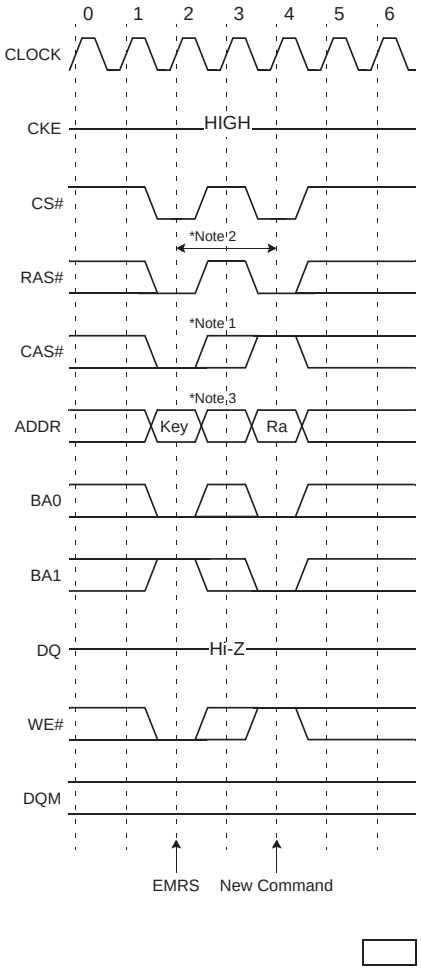
Auto Refresh Cycle



: Don't care

Notes:
Mode Register Set Cycle

1. *CS#, RAS#, CAS#, BA0, BA1 & WE# activation at the same clock cycle with address key will set internal mode register.*
2. *Minimum 2 clock cycles should be met before new RAS activation.*
3. *Please refer to Mode Register Set (MRS).*
4. *All banks precharge should be completed before Mode Register Set cycle and auto refresh cycle.*



Notes:

1. *CS#, RAS#, CAS#, BA0, BA1 & WE# activation at the same clock cycle with address key will set internal mode register.*
2. *Minimum 2 clock cycles should be met before new RAS# activation.*
3. *Please refer to Mode Register Set (MRS).*

Figure 92.16 Extended Mode Register Set Cycle

93 SDRAM Type 2 Revision Summary

Revision A (May 12, 2005)

Initial release.

94 MCP Revision Summary

Revision A (October 19, 2004)

Initial release.

Revision A1 (May 31, 2005)

Added SDRAM Type 2 module

Added FEA137 package diagram

Revision A2 (November 9, 2005)

Updated Flash module and the SDRAM Type 2 module

Changed all RAM module status to Preliminary form Advanced Information

Revision A3 (December 16, 2005)

Corrected package from FEA137 to FTF137