

## Overview

The Rambus Direct RDRAM™ is a general purpose high-performance memory device suitable for use in a broad range of applications including computer memory, graphics, video, and any other application where high bandwidth and low latency are required.

The 128/144-Mbit Direct Rambus DRAMs (RDRAM®) are extremely high-speed CMOS DRAMs organized as 8M words by 16 or 18 bits. The use of Rambus Signaling Level (RSL) technology permits 600 MHz to 800 MHz transfer rates while using conventional system and board design technologies. Direct RDRAM devices are capable of sustained data transfers at 1.25 ns per two bytes (10 ns per sixteen bytes).

The architecture of the Direct RDRAMs allows the highest sustained bandwidth for multiple, simultaneous randomly addressed memory transactions. The separate control and data buses with independent row and column control yield over 95% bus efficiency. The Direct RDRAM's thirty-two banks support up to four simultaneous transactions.

System oriented features for mobile, graphics and large memory systems include power management, byte masking, and x18 organization. The two data bits in the x18 organization are general and can be used for additional storage and bandwidth or for error correction.

## Features

- Highest sustained bandwidth per DRAM device
  - 1.6 GB/s sustained data transfer rate
  - Separate control and data buses for maximized efficiency
  - Separate row and column control buses for easy scheduling and highest performance
  - 32 banks: four transactions can take place simultaneously at full bandwidth data rates
- Low latency features
  - Write buffer to reduce read latency
  - 3 precharge mechanisms for controller flexibility
  - Interleaved transactions
- Advanced power management:
  - Multiple low power states allows flexibility in power consumption versus time to transition to active state
  - Power-down self-refresh
- Organization: 1 Kbyte pages and 32 banks, x16/18
  - x18 organization allows ECC configurations or increased storage/bandwidth
  - x16 organization for low cost applications
- Uses Rambus Signaling Level (RSL) for up to 800 MHz operation



**Figure 1 Direct RDRAM CSP Package**

The 128/144-Mbit Direct RDRAMs are offered in a CSP horizontal package suitable for desktop as well as low-profile add-in card and mobile applications.

Direct RDRAMs operate from a 2.5 V supply.

**Table 1 Key Timing Parameters/Part Numbers**

Organization	I/O Freq. MHz	Trac	Part Number
<b>Normal Package:</b>			
8M × 18	600	53 ns	HYB25R144180C-653
8M × 18	711	45 ns	HYB25R144180C-745
8M × 18	800	45 ns	HYB25R144180C-845
8M × 18	800	40 ns	HYB25R144180C-840
8M × 16	600	53 ns	HYB25R128160C-653
8M × 16	711	45 n s	HYB25R128160C-745
8M × 16	800	45 ns	HYB25R128160C-845
8M × 16	800	40 ns	HYB25R128160C-840
<b>Mirror Package:</b>			
8M × 18	600	53 ns	HYB25M144180C-653
8M × 18	711	45 ns	HYB25M144180C-745
8M × 18	800	45 ns	HYB25M144180C-845
8M × 18	800	40 ns	HYB25M144180C-840
8M × 16	600	53 ns	HYB25M128160C-653
8M × 16	711	45 n s	HYB25M128160C-745
8M × 16	800	45 ns	HYB25M128160C-845
8M × 16	800	40 ns	HYB25M128160C-840

### Pinouts and Definitions

This tables show the pin assignments of the RDRAM package from the top-side of the package (the view looking down on the package as it is mounted on the circuit board). The mechanical dimensions of this package are shown in a later section. Refer to Section “Center-Bonded FBGA Package” on page 86. Note - pin #1 is at the A1 position. DQA8/DQB8 are used for 144 Mbit only. They are N.C. for 128Mbit.

**Table 2 Normal Package (top view)**

12	GND	–	V <sub>DD</sub>	–	–	–	V <sub>DD</sub>	–	GND
11	–	–	–	–	–	–	–	–	–
10	DQA7	DQA4	CFM	CFMN	RQ5	RQ3	DQB0	DQB4	DQB7
9	GND	V <sub>DD</sub>	GND	GNDa	V <sub>DD</sub>	GND	V <sub>DD</sub>	V <sub>DD</sub>	GND
8	CMD	DQA5	DQA2	V <sub>DDa</sub>	RQ6	RQ2	DQB1	DQB5	SIO1
7	–	–	–	–	–	–	–	–	–
6	–	–	–	–	–	–	–	–	–
5	SCK	DQA6	DQA1	V <sub>REF</sub>	RQ7	RQ1	DQB2	DQB6	SIO0
4	V <sub>CMOS</sub>	GND	V <sub>DD</sub>	GND	GND	V <sub>DD</sub>	GND	GND	V <sub>CMOS</sub>
3	DQA8	DQA3	DQA0	CTMN	CTM	RQ4	RQ0	DQB3	DQB8
2	–	–	–	–	–	–	–	–	–
1	GND	–	V <sub>DD</sub>	–	–	–	V <sub>DD</sub>	–	GND
	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>	<b>G</b>	<b>H</b>	<b>J</b>

**Table 3 Mirrored Package (top view)**

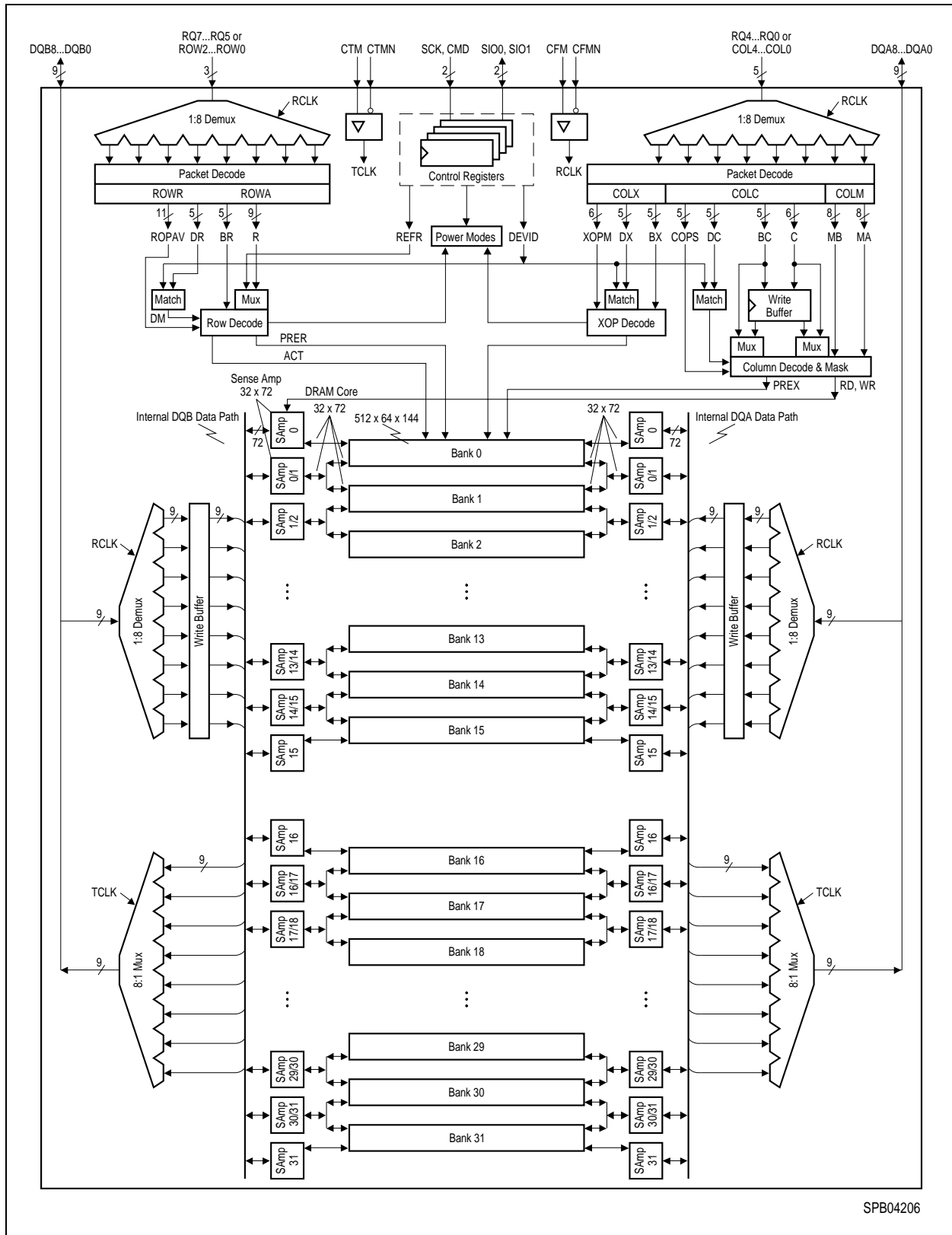
12	GND	–	V <sub>DD</sub>	–	–	–	V <sub>DD</sub>	–	GND
11	–	–	–	–	–	–	–	–	–
10	DQA8	DQA3	DQA0	CTMN	CTM	RQ4	RQ0	DQB3	DQB8
9	V <sub>CMOS</sub>	GND	V <sub>DD</sub>	GND	GND	V <sub>DD</sub>	GND	GND	V <sub>CMOS</sub>
8	SCK	DQA6	DQA1	V <sub>REF</sub>	RQ7	RQ1	DQB2	DQB6	SIO0
7	–	–	–	–	–	–	–	–	–
6	–	–	–	–	–	–	–	–	–
5	CMD	DQA5	DQA2	V <sub>DDa</sub>	RQ6	RQ2	DQB1	DQB5	SIO1
4	GND	V <sub>DD</sub>	GND	GNDa	V <sub>DD</sub>	GND	V <sub>DD</sub>	V <sub>DD</sub>	GND
3	DQA7	DQA4	CFM	CFMN	RQ5	RQ3	DQB0	DQB4	DQB7
2	–	–	–	–	–	–	–	–	–
1	GND	–	V <sub>DD</sub>	–	–	–	V <sub>DD</sub>	–	GND
	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>	<b>G</b>	<b>H</b>	<b>J</b>

**Table 4**

Signal	I/O	Type	# Pins Edge	# Pins Center	Description
SIO1,SIO0	I/O	CMOS <sup>1)</sup>	2	2	Serial input/output. Pins for reading from and writing to the control registers using a serial access protocol. Also used for power management.
CMD	I	CMOS <sup>1)</sup>	1	1	Command input. Pins used in conjunction with SIO0 and SIO1 for reading from and writing to the control registers. Also used for power management.
SCK	I	CMOS <sup>1)</sup>	1	1	Serial clock input. Clock source used for reading from and writing to the control registers.
V <sub>DD</sub>	–	–	14	6	Supply voltage for the RDRAM core and interface logic.
V <sub>DDa</sub>	–	–	2	1	Supply voltage for the RDRAM analog circuitry.
V <sub>CMOS</sub>	–	–	2	2	Supply voltage for CMOS input/output pins.
GND	–	–	19	9	Ground reference for RDRAM core and interface.
GNDa	–	–	2	1	Ground reference for RDRAM analog circuitry.
DQA8 ... DQA0	I/O	RSL <sup>2)</sup>	9	9	Data byte A. Nine pins which carry a byte of read or write data between the Channel and the RDRAM. DQA8 is not used by RDRAMs with a x16 organization.
CFM	I	RSL <sup>2)</sup>	1	1	Clock from master. Interface clock used for receiving RSL signals from the Channel. Positive polarity.
CFMN	I	RSL <sup>2)</sup>	1	1	Clock from master. Interface clock used for receiving RSL signals from the Channel. Negative polarity
V <sub>REF</sub>			1	1	Logic threshold reference voltage for RSL signals
CTMN	I	RSL <sup>2)</sup>	1	1	Clock to master. Interface clock used for transmitting RSL signals to the Channel. Negative polarity.
CTM	I	RSL <sup>2)</sup>	1	1	Clock to master. Interface clock used for transmitting RSL signals to the Channel. Positive polarity.
RQ7 ... RQ5 or ROW2 ... ROW0	I	RSL <sup>2)</sup>	3	3	Row access control. Three pins containing control and address information for row accesses.
RQ4 ... RQ0 or COL4 ... COL0	I	RSL <sup>2)</sup>	5	5	Column access control. Five pins containing control and address information for column accesses.
DQB8 ... DQB0	I/O	RSL <sup>2)</sup>	9	9	Data byte B. Nine pins which carry a byte of read or write data between the Channel and the RDRAM. DQB8 is not used by RDRAMs with a x16 organization.
Total pin count per package			74	54	–

<sup>1)</sup> All CMOS signals are high-true; a high voltage is a logic one and a low voltage is logic zero.

<sup>2)</sup> All RSL signals are low-true; a low voltage is a logic one and a high voltage is logic zero.



**Figure 2 128/144-MBit Direct RDRAM Block Diagram**

## General Description

**Figure 2** is a block diagram of the 128/144 Mbit Direct RDRAM. It consists of two major blocks: a “core” block built from banks and sense amps similar to those found in other types of DRAM, and a Direct Rambus interface block which permits an external controller to access this core at up to 1.6 GB/s.

**Control Registers:** The CMD, SCK, SIO0, and SIO1 pins appear in the upper center of **Figure 2**. They are used to write and read a block of control registers. These registers supply the RDRAM configuration information to a controller and they select the operating modes of the device. The nine bit REFR value is used for tracking the last refreshed row. Most importantly, the five bit DEVID specifies the device address of the RDRAM on the Channel.

**Clocking:** The CTM and CTMN pins (Clock-To-Master) generate TCLK (Transmit Clock), the internal clock used to transmit read data. The CFM and CFMN pins (Clock-From-Master) generate RCLK (Receive Clock), the internal clock signal used to receive write data and to receive the ROW and COL pins.

**DQA, DQB Pins:** These 18 pins carry read (Q) and write (D) data across the Channel. They are multiplexed/de-multiplexed from/to two 72-bit data paths (running at one-eighth the data frequency) inside the RDRAM.

**Banks:** The 16 Mbyte core of the RDRAM is divided into 32 0.5 Mbyte banks, each organized as 512 rows, with each row containing 64 dualocts, and each dualoct containing 16 bytes. A dualoct is the smallest unit of data that can be addressed.

**Sense Amps:** The RDRAM contains 34 sense amps. Each sense amp consists of 512 bytes of fast storage (256 for DQA and 256 for DQB) and can hold one-half of one row of one bank of the RDRAM. The sense amp may hold any of the 512 half-rows of an associated bank. However, each sense amp is shared between two adjacent banks of the RDRAM (except for numbers 0, 15, 30, and 31). This introduces the restriction that adjacent banks may not be simultaneously accessed.

**RQ Pins:** These pins carry control and address information. They are broken into two groups. RQ7 ... RQ5 are also called ROW2 ... ROW0, and are used primarily for controlling row accesses. RQ4 ... RQ0 are also called COL4 ... COL0, and are used primarily for controlling column accesses.

**ROW Pins:** The principle use of these three pins is to manage the transfer of data between the banks and the sense amps of the RDRAM. These pins are de-multiplexed into a 24-bit ROWA (row-activate) or ROWR (row-operation) packet.

**COL Pins:** The principle use of these five pins is to manage the transfer of data between the DQA/DQB pins and the sense amps of the RDRAM. These pins are de-multiplexed into a 23-bit COLC (column-operation) packet and either a 17-bit COLM (mask) packet or a 17-bit COLX (extended-operation) packet.

**ACT Command:** An ACT (activate) command from an ROWA packet causes one of the 512 rows of the selected bank to be loaded to its associated sense amps (two 256 byte sense amps for DQA and two for DQB).

**PRER Command:** A PRER (precharge) command from an ROWR packet causes the selected bank to release its two associated sense amps, permitting a different row in that bank to be activated, or permitting adjacent banks to be activated.

**RD Command:** The RD (read) command causes one of the 64 dualocts of one of the sense amps to be transmitted on the DQA/DQB pins of the Channel.

**WR Command:** The WR (write) command causes a dualoct received from the DQA/DQB data pins of the Channel to be loaded into the write buffer. There is also space in the write buffer for the BC bank address and C column address information. The data in the write buffer is automatically retired (written with optional bytemask) to one of the 64 dualocts of one of the sense amps during a subsequent COP command. A retire can take place during a RD, WR, or NOCOP to another device, or during a WR or NOCOP to the same device. The write buffer will not retire during a RD to the same device. The write buffer reduces the delay needed for the internal DQA/DQB data path turn-around.

**PREC Precharge:** The PREC, RDA and WRA commands are similar to NOCOP, RD and WR, except that a precharge operation is performed at the end of the column operation. These commands provide a second mechanism for performing precharge.

**PREX Precharge:** After a RD command, or after a WR command with no byte masking ( $M = 0$ ), a COLX packet may be used to specify an extended operation (XOP). The most important XOP command is PREX. This command provides a third mechanism for performing precharge.

### Packet Format

**Figure 3** shows the formats of the ROWA and ROWR packets on the ROW pins. **Table 5** describes the fields which comprise these packets. DR4T and DR4F bits are encoded to contain both the DR4 device address bit and a framing bit which allows the ROWA or ROWR packet to be recognized by the RDRAM.

The AV (ROWA/ROWR packet selection) bit distinguishes between the two packet types. Both the ROWA and ROWR packet provide a five bit device address and a five bit bank address. An ROWA packet uses the remaining bits to specify a nine bit row address, and the ROWR packet uses the remaining bits for an eleven bit opcode field. Note the use of the “RsvX” notation to reserve bits for future address field extension.

**Table 5 Field Description for ROWA Packet and ROWR Packet**

Field	Description
DR4T, DR4F	Bits for framing (recognizing) a ROWA or ROWR packet. Also encodes highest device address bit.
DR3 ... DR0	Device address for ROWA or ROWR packet.
BR4 ... BR0	Bank address for ROWA or ROWR packet. RsvB denotes bits ignored by the RDRAM.
AV	Selects between ROWA packet (AV = 1) and ROWR packet (AV = 0).
R8 ... R0	Row address for ROWA packet. RsvR denotes bits ignored by the RDRAM.
ROP10 ... ROP0	Opcode field for ROWR packet. Specifies precharge, refresh, and power management functions.

**Figure 3** also shows the formats of the COLC, COLM, and COLX packets on the COL pins. **Table 6** describes the fields which comprise these packets.

The COLC packet uses the S (Start) bit for framing. A COLM or COLX packet is aligned with this COLC packet, and is also framed by the S bit.

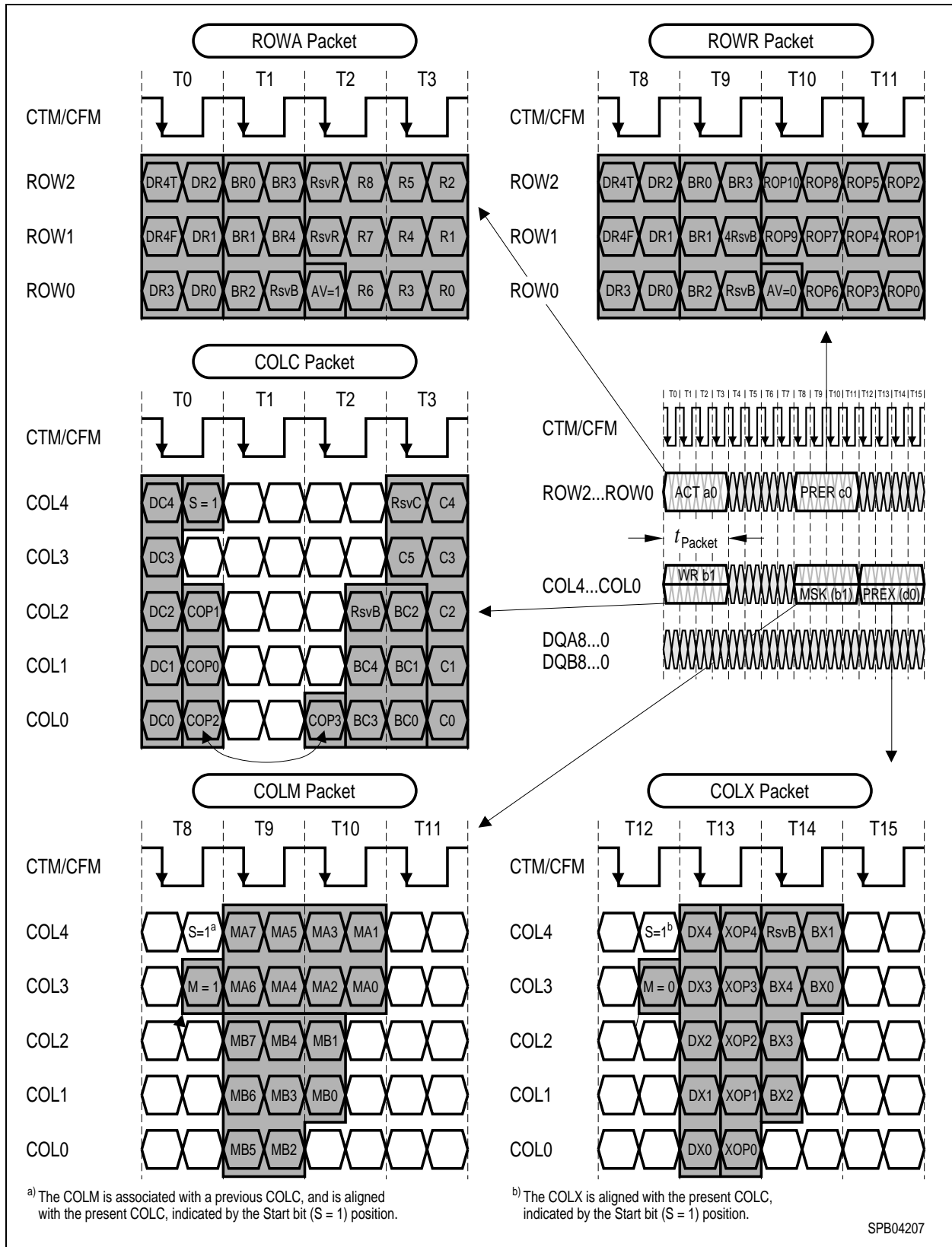
The 23 bit COLC packet has a five bit device address, a five bit bank address, a six bit column address, and a four bit opcode. The COLC packet specifies a read or write command, as well as some power management commands.

The remaining 17 bits are interpreted as a COLM ( $M = 1$ ) or COLX ( $M = 0$ ) packet. A COLM packet is used for a COLC write command which needs bytemask control. The COLM packet is associated with the COLC packet from a time  $t_{RTR}$  earlier. An COLX packet may be used to specify an independent precharge command. It contains a five bit device address, a five bit bank address, and a five bit opcode. The COLX packet may also be used to specify some housekeeping and power management commands. The COLX packet is framed within a COLC packet but is not otherwise associated with any other packet.

**Table 6 Field Description for COLC Packet, COLM Packet, and COLX Packet**

Field	Description
S	Bit for framing (recognizing) a COLC packet, and indirectly for framing COLM and COLX packets.
DC4 ... DC0	Device address for COLC packet.
BC4 ... BC0	Bank address for COLC packet. RsvB denotes bits reserved for future extension (controller drives 0's).
C5 ... C0	Column address for COLC packet. RsvC denotes bits ignored by the RDRAM.
COP3 ... COP0	Opcode field for COLC packet. Specifies read, write, precharge, and power management functions.
M	Selects between COLM packet ( $M = 1$ ) and COLX packet ( $M = 0$ ).
MA7 ... MA0	Bytemask write control bits. 1 = write, 0 = no-write. MA0 controls the earliest byte on DQA8 ... 0.
MB7 ... MB0	Bytemask write control bits. 1 = write, 0 = no-write. MB0 controls the earliest byte on DQB8 ... 0.
DX4 ... DX0	Device address for COLX packet.
BX4 ... BX0	Bank address for COLX packet. RsvB denotes bits reserved for future extension (controller drives 0's).
XOP4 ... XOP0	Opcode field for COLX packet. Specifies precharge, $I_{OL}$ control, and power management functions.





**Figure 3 Packet Formats**

### Field Encoding Summary

**Table 7** shows how the six device address bits are decoded for the ROWA and ROWR packets. The DR4T and DR4F encoding merges a fifth device bit with a framing bit. When neither bit is asserted, the device is not selected. Note that a broadcast operation is indicated when both bits are set. Broadcast operation would typically be used for refresh and power management commands. If the device is selected, the DM (DeviceMatch) signal is asserted and an ACT or ROP command is performed.

**Table 7 Device Field Encodings for ROWA Packet and ROWR Packet**

DR4T	DR4F	Device Selection	Device Match Signal (DM)
1	1	All devices (broadcast)	DM is set to 1
0	1	One device selected	DM is set to 1 if {DEVID4 ... DEVID0} == {0, DR3 ... DR0} else DM is set to 0
1	0	One device selected	DM is set to 1 if {DEVID4 ... DEVID0} == {1, DR3 ... DR0} else DM is set to 0
0	0	No packet present	DM is set to 0

**Table 8** shows the encodings of the remaining fields of the ROWA and ROWR packets. An ROWA packet is specified by asserting the AV bit. This causes the specified row of the specified bank of this device to be loaded into the associated sense amps.

An ROWR packet is specified when AV is not asserted. An 11 bit opcode field encodes a command for one of the banks of this device. The PRER command causes a bank and its two associated sense amps to precharge, so another row or an adjacent bank may be activated. The REFA (refresh-activate) command is similar to the ACT command, except the row address comes from an internal register REFR, and REFR is incremented at the largest bank address. The REFP (refresh-precharge) command is identical to a PRER command.

The NAPR, NAPRC, PDNR, ATTN, and RLXR commands are used for managing the power dissipation of the RDRAM and are described in more detail in “Power State Management” on page 58. The TCEN and TCAL commands are used to adjust the output driver slew rate and they are described in more detail in “Current and Temperature Control” on page 65.

**Table 8 ROWA Packet and ROWR Packet Field Encodings**

DM <sup>1)</sup>	AV	ROP10...ROP0 Field									Name	Command Description
		10	9	8	7	6	5	4	3	2:0		
0	-	-	-	-	-	-	-	-	-	---	-	No operation.
1	1	Row address									ACT	Activate row R8 ... R0 of bank BR4 ... BR0 of device and move device to ATTN <sup>2)</sup> .
1	0	1	1	0	0	0	x <sup>3)</sup>	x	x	000	PRER	Precharge bank BR4 ... BR0 of this device.
1	0	0	0	0	1	1	0	0	x	000	REFA	Refresh (activate) row REFR8 ... REFR0 of bank BR3 ... BR0 of device. Increment REFR if BR4 ... BR0 = 1111 (see <b>Figure 50</b> ).
1	0	1	0	1	0	1	0	0	x	000	REFP	Precharge bank BR4 ... BR0 of this device after REFA (see <b>Figure 50</b> ).
1	0	x	x	0	0	0	0	1	x	000	PDNR	Move this device into the powerdown (PDN) power state (see <b>Figure 47</b> ).
1	0	x	x	0	0	0	1	0	x	000	NAPR	Move this device into the nap (NAP) power state (see <b>Figure 47</b> ).
1	0	x	x	0	0	0	1	1	x	000	NAPRC	Move this device into the nap (NAP) power state conditionally.
1	0	x	x	x	x	x	x	x	0	000	ATTN <sup>2)</sup>	Move this device into the attention (ATTN) power state (see <b>Figure 45</b> ).
1	0	x	x	x	x	x	x	x	1	000	RLXR	Move this device into the standby (STBY) power state (see <b>Figure 46</b> ).
1	0	0	0	0	0	0	0	0	x	001	TCAL	Temperature calibrate this device (see <b>Figure 52</b> ).
1	0	0	0	0	0	0	0	0	x	010	TCEN	Temperature calibrate/enable this device (see <b>Figure 52</b> ).
1	0	0	0	0	0	0	0	0	0	000	NOROP	No operation.

<sup>1)</sup> The DM (Device Match signal) value is determined by the DR4T, DR4F, DR3 ... DR0 field of the ROWA and ROWR packets. See **Table 7**.

<sup>2)</sup> The ATTN command does not cause a RLX-to-ATTN transition for a broadcast operation (DR4T/DR4F = 1/1).

<sup>3)</sup> An "x" entry indicates which commands may be combined. For instance, the three commands PRER/NAPRC/RLXR may be specified in one ROP value (011000111000).

**Table 9** shows the COP field encoding. The device must be in the ATTN power state in order to receive COLC packets. The COLC packet is used primarily to specify RD (read) and WR (write) commands. Retire operations (moving data from the write buffer to a sense amp) happen automatically. See **Figure 17** for a more detailed description.

The COLC packet can also specify a PREC command, which precharges a bank and its associated sense amps. The RDA/WRA commands are equivalent to combining RD/WR with a PREC. RLXC (relax) performs a power mode transition. See "Power State Management" on page 58.

**Table 9 COLC Packet Field Encodings**

S	DC4 ... DC0 (select device) <sup>1)</sup>	COP3 ... 0	Name	Command Description
0	----	-----	–	No operation.
1	/= (DEVID4 ... 0)	-----	–	Retire write buffer of this device.
1	== (DEVID4 ... 0)	x000 <sup>2)</sup>	NOCOP	Retire write buffer of this device.
1	== (DEVID4 ... 0)	x001	WR	Retire write buffer of this device, then write column C5 ... C0 of bank BC4 ... BC0 to write buffer.
1	== (DEVID4 ... 0)	x010	RSRV	Reserved, no operation.
1	== (DEVID4 ... 0)	x011	RD	Read column C5 ... C0 of bank BC4 ... BC0 of this device.
1	== (DEVID4 ... 0)	x100	PREC	Retire write buffer of this device, then precharge bank BC4 ... BC0 (see <b>Figure 14</b> ).
1	== (DEVID4 ... 0)	x101	WRA	Same as WR, but precharge bank BC4 ... BC0 after write buffer (with new data) is retired.
1	== (DEVID4 ... 0)	x110	RSRV	Reserved, no operation.
1	== (DEVID4 ... 0)	x111	RDA	Same as RD, but precharge bank BC4 ... BC0 afterward.
1	== (DEVID4 ... 0)	1xxx	RLXC	Move this device into the standby (STBY) power state (see <b>Figure 46</b> ).

<sup>1)</sup> “/=” means not equal, “==” means equal.

<sup>2)</sup> An “x” entry indicates which commands may be combined. For instance, the two commands WR/RLXC may be specified in one COP value (1001).

**Table 10** shows the COLM and COLX field encodings. The M bit is asserted to specify a COLM packet with two 8 bit bytemask fields MA and MB. If the M bit is not asserted, an COLX is specified. It has device and bank address fields, and an opcode field. The primary use of the COLX packet is to permit an independent PREX (precharge) command to be specified without consuming control bandwidth on the ROW pins. It is also used for the CAL (calibrate) and SAM (sample) current control commands (see “Current and Temperature Control” on page 65), and for the RLXX power mode command (see “Power State Management” on page 58).

**Table 10 COLM Packet and COLX Packet Field Encodings**

M	DX4 ... DX0 (selects device)	XOP4 ... 0	Name	Command Description
1	----	–	MSK	MB/MA bytemasks used by WR/WRA.
0	/= (DEVID4 ... 0)	–	–	No operation.
0	== (DEVID4 ... 0)	00000	NOXOP	No operation.
0	== (DEVID4 ... 0)	1xxx0 <sup>1)</sup>	PREX	Precharge bank BX4 ... BX0 of this device (see <b>Figure 14</b> ).
0	== (DEVID4 ... 0)	x10x0	CAL	Calibrate (drive) $I_{OL}$ current for this device (see <b>Figure 51</b> ).
0	== (DEVID4 ... 0)	x11x0	CAL/SAM	Calibrate (drive) and Sample (update) $I_{OL}$ current for this device (see <b>Figure 51</b> ).
0	== (DEVID4 ... 0)	xxx10	RLXX	Move this device into the standby (STBY) power state (see <b>Figure 46</b> ).
0	== (DEVID4 ... 0)	xxxx1	RSRV	Reserved, no operation.

<sup>1)</sup> An “x” entry indicates which commands may be combined. For instance, the two commands PREX/RLXX may be specified in one XOP value (10010).

### DQ Packet Timing

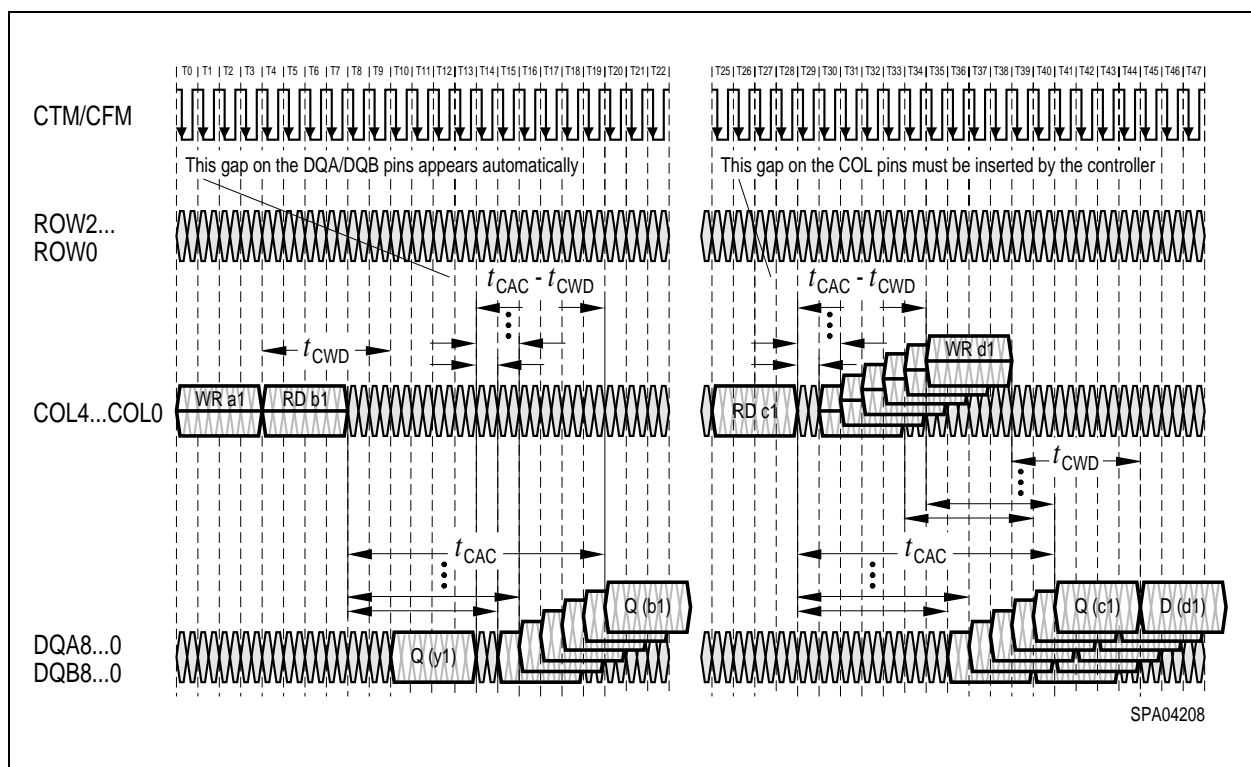
**Figure 4** shows the timing relationship of COLC packets with D and Q data packets. This document uses a specific convention for measuring time intervals between packets: all packets on the ROW and COL pins (ROWA, ROWR, COLC, COLM, COLX) use the trailing edge of the packet as a reference point, and all packets on the DQA/DQB pins (D and Q) use the leading edge of the packet as a reference point.

An RD or RDA command will transmit a dualoct of read data Q a time  $t_{CAC}$  later. This time includes one to five cycles of round-trip propagation delay on the Channel. The  $t_{CAC}$  parameter may be programmed to a one of a range of values (7, 8, 9, 10, 11, or 12  $t_{CYCLE}$ ). The value chosen depends upon the number of RDRAM devices on the Channel and the RDRAM timing bin. See **Figure 39** for more information.

A WR or WRA command will receive a dualoct of write data D a time  $t_{CWD}$  later. This time does not need to include the round-trip propagation time of the Channel since the COLC and D packets are traveling in the same direction.

When a Q packet follows a D packet (shown in the left half of the figure), a gap ( $t_{CAC} - t_{CWD}$ ) will automatically appear between them because the  $t_{CWD}$  value is always less than the  $t_{CAC}$  value. There will be no gap between the two COLC packets with the WR and RD commands which schedule the D and Q packets.

When a D packet follows a Q packet (shown in the right half of the figure), no gap is needed between them because the  $t_{CWD}$  value is less than the  $t_{CAC}$  value. However, a gap of  $t_{CAC} - t_{CWD}$  or greater must be inserted between the COLC packets with the RD WR commands by the controller so the Q and D packets do not overlap.



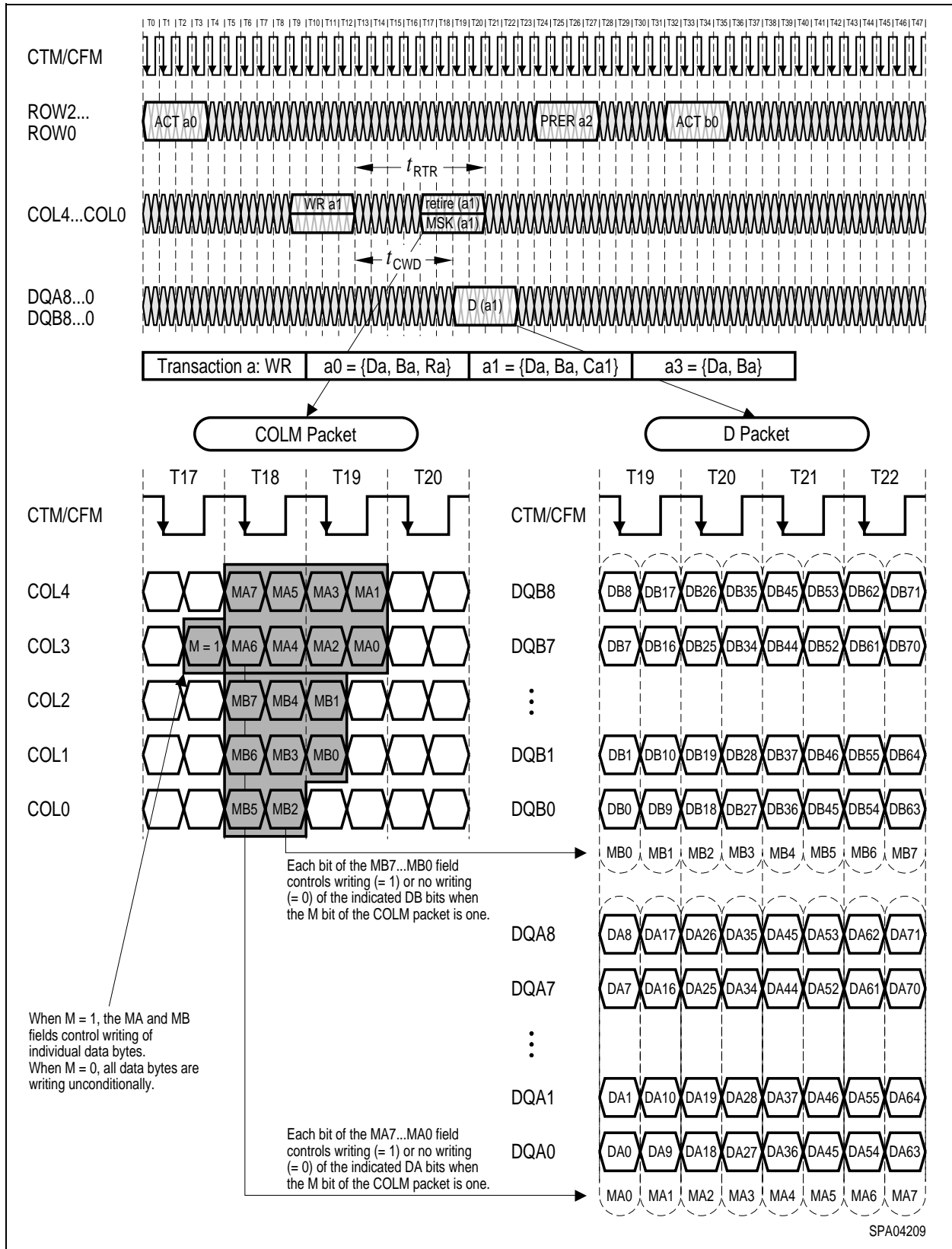
**Figure 4** Read (Q) and Write (D) Data Packet - Timing for  $t_{CAC} = 7, 8, 9, 10, 11,$  or  $12 t_{CYCLE}$

### COLM Packet to D Packet Mapping

**Figure 5** shows a write operation initiated by a WR command in a COLC packet. If a subset of the 16 bytes of write data are to be written, then a COLM packet is transmitted on the COL pins a time  $t_{RTR}$  after the COLC packet containing the WR command. The M bit of the COLM packet is set to indicate that it contains the MA and MB mask fields. Note that this COLM packet is aligned with the COLC packet which causes the write buffer to be retired. See **Figure 17** for more details.

If all 16 bytes of the D data packet are to be written, then no further control information is required. The packet slot that would have been used by the COLM packet ( $t_{RTR}$  after the COLC packet) is available to be used as an COLX packet. This could be used for a PREX precharge command or for a housekeeping command (this case is not shown). The M bit is not asserted in an COLX packet and causes all 16 bytes of the previous WR to be written unconditionally. Note that a RD command will never need a COLM packet, and will always be able to use the COLX packet option (a read operation has no need for the byte-write-enable control bits).

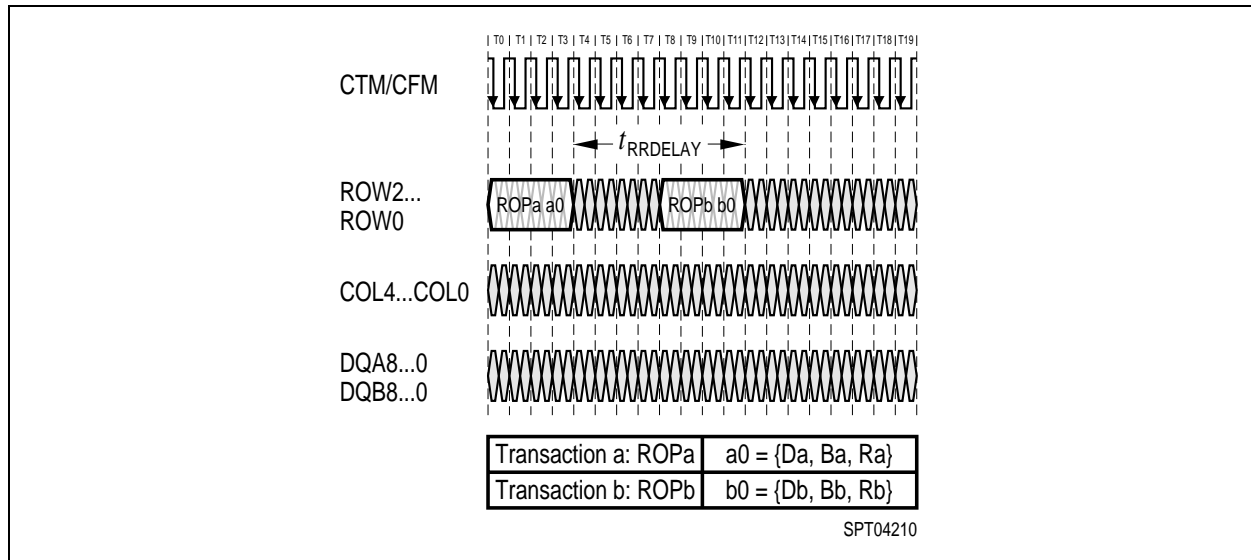
**Figure 5** also shows the mapping between the MA and MB fields of the COLM packet and bytes of the D packet on the DQA and DQB pins. Each mask bit controls whether a byte of data is written (= 1) or not written (= 0).



**Figure 5 Mapping Between COLM Packet and D Packet for WR Command**



### ROW-to-ROW Packet Interaction



**Figure 6** ROW-to-ROW Packet Interaction-Timing

**Figure 6** shows two packets on the ROW pins separated by an interval  $t_{RRDELAY}$  which depends upon the packet contents. No other ROW packets are sent to banks {Ba, Ba+1, Ba-1} between packet “a” and packet “b” unless noted otherwise. **Table 11** summarizes the  $t_{RRDELAY}$  values for all possible cases.

Cases RR1 through RR4 show two successive ACT commands. In case RR1, there is no restriction since the ACT commands are to different devices. In case RR2, the  $t_{RR}$  restriction applies to the same device with non-adjacent banks. Cases RR3 and RR4 are illegal (as shown) since bank Ba needs to be precharged. If a PRER to Ba, Ba+1, or Ba-1 is inserted,  $t_{RRDELAY}$  is  $t_{RC}$  ( $t_{RAS}$  to the PRER command, and  $t_{RP}$  to the next ACT).

Cases RR5 through RR8 show an ACT command followed by a PRER command. In cases RR5 and RR6, there are no restrictions since the commands are to different devices or to non-adjacent banks of the same device. In cases RR7 and RR8, the  $t_{RAS}$  restriction means the activated bank must wait before it can be precharged.

Cases RR9 through RR12 show a PRER command followed by an ACT command. In cases RR9 and RR10, there are essentially no restrictions since the commands are to different devices or to non-adjacent banks of the same device. RR10a and RR10b depend upon whether a bracketed bank (Ba ± 1) is precharged or activated. In cases RR11 and RR12, the same and adjacent banks must all wait  $t_{RP}$  for the sense amp and bank to precharge before being activated.

**Table 11 ROW-to-ROW Packet Interaction - Rules**

Case #	ROPa	Da	Ba	Ra	ROPb	Db	Bb	Rb	$t_{RRDELAY}$	Example
RR1	ACT	Da	Ba	Ra	ACT	/= Da	xxxx	X...X	$t_{PACKET}$	Figure 11
RR2	ACT	Da	Ba	Ra	ACT	== Da	/= {Ba, Ba+1, Ba-1}	X...X	$t_{RR}$	Figure 11
RR3	ACT	Da	Ba	Ra	ACT	== Da	== {Ba+1, Ba-1}	X...X	$t_{RC}$ - illegal unless PRER to Ba/Ba+1/Ba-1	Figure 10
RR4	ACT	Da	Ba	Ra	ACT	== Da	== {Ba}	X...X	$t_{RC}$ - illegal unless PRER to Ba/Ba+1/Ba-1	Figure 10
RR5	ACT	Da	Ba	Ra	PRER	/= Da	xxxx	X...X	$t_{PACKET}$	Figure 11
RR6	ACT	Da	Ba	Ra	PRER	== Da	/= {Ba, Ba+1, Ba-1}	X...X	$t_{PACKET}$	Figure 11
RR7	ACT	Da	Ba	Ra	PRER	== Da	== {Ba+1, Ba-1}	X...X	$t_{RAS}$	Figure 10
RR8	ACT	Da	Ba	Ra	PRER	== Da	== {Ba}	X...X	$t_{RAS}$	Figure 15
RR9	PRER	Da	Ba	Ra	ACT	/= Da	xxxx	X...X	$t_{PACKET}$	Figure 12
RR10	PRER	Da	Ba	Ra	ACT	== Da	/= {Ba, Ba±1, Ba±2}	X...X	$t_{PACKET}$	Figure 12
RR10a	PRER	Da	Ba	Ra	ACT	== Da	== {Ba+2}	X...X	$t_{PACKET}/t_{RP}$ if Ba+1 is precharged/activated.	–
RR10b	PRER	Da	Ba	Ra	ACT	== Da	== {Ba-2}	X...X	$t_{PACKET}/t_{RP}$ if Ba-1 is precharged/activated.	–
RR11	PRER	Da	Ba	Ra	ACT	== Da	== {Ba+1, Ba-1}	X...X	$t_{RP}$	Figure 10
RR12	PRER	Da	Ba	Ra	ACT	== Da	== {Ba}	X...X	$t_{RP}$	Figure 10
RR13	PRER	Da	Ba	Ra	PRER	/= Da	xxxx	X...X	$t_{PACKET}$	Figure 12
RR14	PRER	Da	Ba	Ra	PRER	== Da	/= {Ba, Ba+1, Ba-1}	X...X	$t_{PP}$	Figure 12
RR15	PRER	Da	Ba	Ra	PRER	== Da	== {Ba+1, Ba-1}	X...X	$t_{PP}$	Figure 12
RR16	PRER	Da	Ba	Ra	PRER	== Da	== Ba	X...X	$t_{PP}$	Figure 12

**ROW-to-ROW Interaction (cont'd)**

Cases RR13 through RR16 summarize the combinations of two successive PRER commands. In case RR13 there is no restriction since two devices are addressed. In RR14,  $t_{PP}$  applies, since the same device is addressed. In RR15 and RR16, the same bank or an adjacent bank may be given repeated PRER commands with only the  $t_{PP}$  restriction.

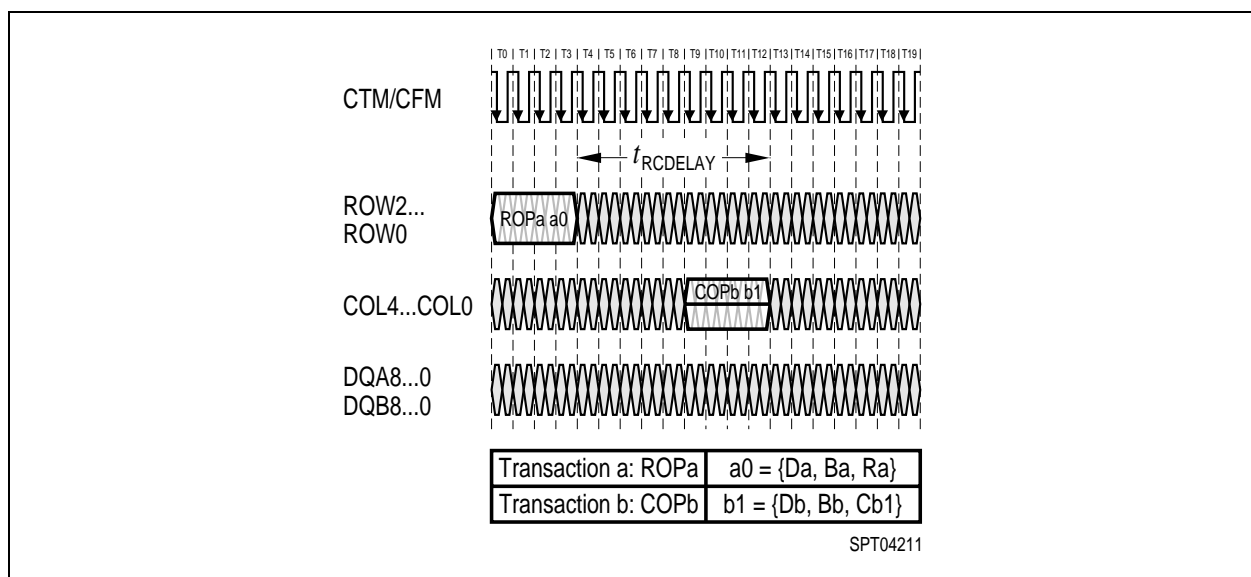
Two adjacent banks can't be activate simultaneously. A precharge command to one bank will thus affect the state of the adjacent banks (and sense amps). If bank Ba is activate and a PRER is directed to Ba, then bank Ba will be precharged along with sense amps Ba-1/Ba and Ba/Ba+1. If bank Ba+1 is activate and a PRER is directed to Ba, then bank Ba+1 will be precharged along with sense amps Ba/Ba+1 and Ba+1/Ba+2. If bank Ba-1 is activate and a PRER is directed to Ba, then bank Ba-1 will be precharged along with sense amps Ba/Ba-1 and Ba-1/Ba-2.

A ROW packet may contain commands other than ACT or PRER. The REFA and REFP commands are equivalent to ACT and PRER for interaction analysis purposes. The interaction rules of the NAPR, NAPRC, PDNR, RLXR, ATTN, TCAL, and TCEN commands are discussed in later sections (see **Table 8** for cross-ref).

### ROW-to-COL Packet Interaction

**Figure 7** shows two packets on the ROW and COL pins. They must be separated by an interval  $t_{RCDELAY}$  which depends upon the packet contents. **Table 12** summarizes the  $t_{RCDELAY}$  values for all possible cases. Note that if the COL packet is earlier than the ROW packet, it is considered a COL-to-ROW packet interaction.

Cases RC1 through RC5 summarize the rules when the ROW packet has an ACT command. **Figure 15** and **Figure 16** show examples of RC5 - an activation followed by a read or write. RC4 is an illegal situation, since a read or write of a precharged banks is being attempted (remember that for a bank to be activated, adjacent banks must be precharged). In cases RC1, RC2, and RC3, there is no interaction of the ROW and COL packets.



**Figure 7 ROW-to-COL Packet Interaction - Timing**

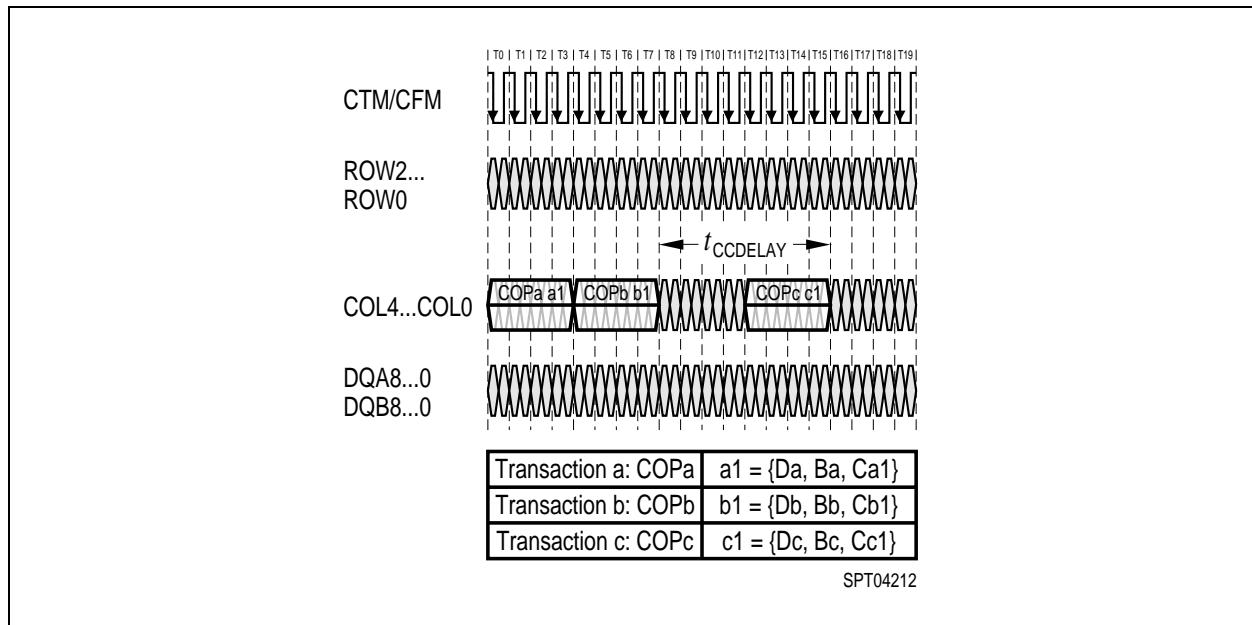
Cases RC6 through RC8 summarize the rules when the ROW packet has a PRER command. There is either no interaction (RC6 through RC9) or an illegal situation with a read or write of a precharged bank (RC9).

The COL pins can also schedule a precharge operation with a RDA, WRA, or PREC command in a COLC packet or a PREX command in a COLX packet. The constraints of these precharge operations may be converted to equivalent PRER command constraints using the rules summarized in **Figure 14**.

**Table 12 ROW-to-COL Packet Interaction - Rules**

Case #	ROPa	Da	Ba	Ra	COPb	Db	Bb	Cb1	$t_{RCDELAY}$	Example
RC1	ACT	Da	Ba	Ra	NOCOP, RD, retire	$\neq$ Da	xxxx	x...x	0	–
RC2	ACT	Da	Ba	Ra	NOCOP	$=$ Da	xxxx	x...x	0	–
RC3	ACT	Da	Ba	Ra	RD, retire	$=$ Da	$\neq$ {Ba, Ba+1, Ba-1}	x...x	0	–
RC4	ACT	Da	Ba	Ra	RD, retire	$=$ Da	$=$ {Ba+1, Ba-1}	x...x	Illegal	–
RC5	ACT	Da	Ba	Ra	RD, retire	$=$ Da	$=$ Ba	x...x	$t_{RCD}$	<b>Figure 15</b>
RC6	PRER	Da	Ba	Ra	NOCOP, RD, retire	$\neq$ Da	xxxx	x...x	0	–
RC7	PRER	Da	Ba	Ra	NOCOP	$=$ Da	xxxx	x...x	0	–
RC8	PRER	Da	Ba	Ra	RD, retire	$=$ Da	$\neq$ {Ba, Ba+1, Ba-1}	x...x	0	–
RC9	PRER	Da	Ba	Ra	RD, retire	$=$ Da	$=$ {Ba+1, Ba-1}	x...x	Illegal	–

**COL-to-COL Packet Interaction**



**Figure 8 COL-to-COL Packet Interaction-Timing**

**Figure 8** shows three arbitrary packets on the COL pins. Packets “b” and “c” must be separated by an interval  $t_{CCDELAY}$  which depends upon the command and address values in all three packets. **Table 13** summarizes the  $t_{CCDELAY}$  values for all possible cases.

Cases CC1 through CC5 summarize the rules for every situation other than the case when COPb is a WR command and COPc is a RD command. In CC3, when a RD command is followed by a WR command, a gap of  $t_{CAC} - t_{CWD}$  must be inserted between the two COL packets. See **Figure 4** for more explanation of why this gap is needed. For cases CC1, CC2, CC4, and CC5, there is no restriction ( $t_{CCDELAY}$  is  $t_{CC}$ ).

In cases CC6 through CC10, COPb is a WR command and COPc is a RD command. The  $t_{CCDELAY}$  value needed between these two packets depends upon the command and address in the packet with COPa. In particular, in case CC6 when there is WR-WR-RD command sequence directed to the same device, a gap will be needed between the packets with COPb and COPc. The gap will need a COLC packet with a NOCOP command directed to any device in order to force an automatic retire to take place. **Figure 18** (right) provides a more detailed explanation of this case.

In case CC10, there is a RD-WR-RD sequence directed to the same device. If a prior write to the same device is unretired when COPa is issued, then a gap will be needed between the packets with COPb and COPc as in case CC6. The gap will need a COLC packet with a NOCOP command directed to any device in order to force an automatic retire to take place.

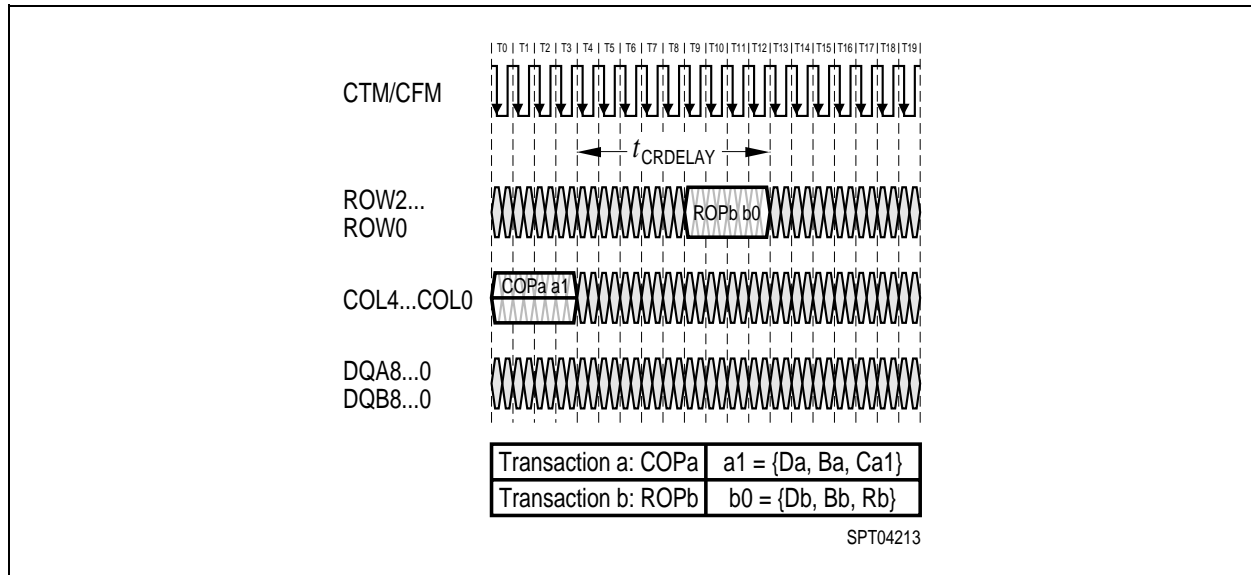
Cases CC7, CC8, and CC9 have no restriction ( $t_{CCDELAY}$  is  $t_{CC}$ ).

For the purposes of analyzing COL-to-ROW interactions, the PREC, WRA, and RDA commands of the COLC packet are equivalent to the NOCOP, WR, and RD commands. These commands also cause a precharge operation PREC to take place. This precharge may be converted to an equivalent PRER command on the ROW pins using the rules summarized in **Figure 14**.

**Table 13 COL-to-COL Packet Interaction - Rules**

Case #	COPa	Da	Ba	Ca1	COPb	Db	Bb	Cb1	COPc	Dc	Bc	Cc1	$t_{CCDELAY}$	Example
CC1	xxxx	xxxxx	x...x	x...x	NOCOP	Db	Bb	Cb1	xxxx	xxxxx	x...x	x...x	$t_{CC}$	–
CC2	xxxx	xxxxx	x...x	x...x	RD,WR	Db	Bb	Cb1	NOCOP	xxxxx	x...x	x...x	$t_{CC}$	–
CC3	xxxx	xxxxx	x...x	x...x	RD	Db	Bb	Cb1	WR	xxxxx	x...x	x...x	$t_{CC}+t_{CAC}-t_{CWD}$	<b>Figure 4</b>
CC4	xxxx	xxxxx	x...x	x...x	RD	Db	Bb	Cb1	RD	xxxxx	x...x	x...x	$t_{CC}$	<b>Figure 15</b>
CC5	xxxx	xxxxx	x...x	x...x	WR	Db	Bb	Cb1	WR	xxxxx	x...x	x...x	$t_{CC}$	<b>Figure 16</b>
CC6	WR	== Db	x	x...x	WR	Db	Bb	Cb1	RD	== Db	x...x	x...x	$t_{RTR}$	<b>Figure 18</b>
CC7	WR	== Db	x	x...x	WR	Db	Bb	Cb1	RD	/= Db	x...x	x...x	$t_{CC}$	–
CC8	WR	/= Db	x	x...x	WR	Db	Bb	Cb1	RD	== Db	x...x	x...x	$t_{CC}$	–
CC9	NOCOP	== Db	x	x...x	WR	Db	Bb	Cb1	RD	== Db	x...x	x...x	$t_{CC}$	–
CC10	RD	== Db	x	x...x	WR	Db	Bb	Cb1	RD	== Db	x...x	x...x	$t_{CC}$	–

### COL-to-ROW Packet Interaction



**Figure 9 COL-to-ROW Packet Interaction - Timing**

**Figure 9** shows arbitrary packets on the COL and ROW pins. They must be separated by an interval  $t_{CRDELAY}$  which depends upon the command and address values in the packets. **Table 14** summarizes the  $t_{CRDELAY}$  value for all possible cases.

Cases CR1, CR2, CR3, and CR9 show no interaction between the COL and ROW packets, either because one of the commands is a NOP or because the packets are directed to different devices or to non-adjacent banks.

Case CR4 is illegal because an already-activated bank is to be re-activated without being precharged. Case CR5 is illegal because an adjacent bank can't be activated or precharged until bank Ba is precharged first.

In case CR6, the COLC packet contains a RD command, and the ROW packet contains a PRER command for the same bank. The  $t_{RDP}$  parameter specifies the required spacing.

Likewise, in case CR7, the COLC packet causes an automatic retire to take place, and the ROW packet contains a PRER command for the same bank. The  $t_{RTP}$  parameter specifies the required spacing.

Case CR8 is labeled "Hazardous" because a WR command should always be followed by an automatic retire before a precharge is scheduled. **Figure 19** shows an example of what can happen when the retire is not able to happen before the precharge.

For the purposes of analyzing COL-to-ROW interactions, the PREC, WRA, and RDA commands of the COLC packet are equivalent to the NOCOP, WR, and RD commands. These commands also cause a precharge operation to take place. This precharge may be converted to an equivalent PRER command on the ROW pins using the rules summarized in **Figure 14**.

A ROW packet may contain commands other than ACT or PRER. The REFA and REFP commands are equivalent to ACT and PRER for interaction analysis purposes. The interaction rules of the NAPR, PDNR, and RLXR commands are discussed in a later section.

**Table 14 COL-to-ROW Packet Interaction - Rules**

Case #	COPa	Da	Ba	Ca1	ROPb	Db	Bb	Rb	$t_{CRDELAY}$	Example
CR1	NOCOP	Da	Ba	Ca1	x...x	xxxxx	xxxx	x...x	0	–
CR2	RD/WR	Da	Ba	Ca1	x...x	/= Da	xxxx	x...x	0	–
CR3	RD/WR	Da	Ba	Ca1	x...x	== Da	/= {Ba, Ba+1, Ba-1}	x...x	0	–
CR4	RD/WR	Da	Ba	Ca1	ACT	== Da	== {Ba}	x...x	Illegal	–
CR5	RD/WR	Da	Ba	Ca1	ACT	== Da	== {Ba+1, Ba-1}	x...x	Illegal	–
CR6	RD	Da	Ba	Ca1	PRER	== Da	== {Ba, Ba+1, Ba-1}	x...x	$t_{RDP}$	<b>Figure 15</b>
CR7	retire <sup>1)</sup>	Da	Ba	Ca1	PRER	== Da	== {Ba, Ba+1, Ba-1}	x...x	$t_{RTP}$	<b>Figure 16</b>
CR8	WR <sup>2)</sup>	Da	Ba	Ca1	PRER	== Da	== {Ba, Ba+1, Ba-1}	x...x	0	<b>Figure 19</b>
CR9	xxxx	Da	Ba	Ca1	NOROP	xxxxx	xxxx	x...x	0	–

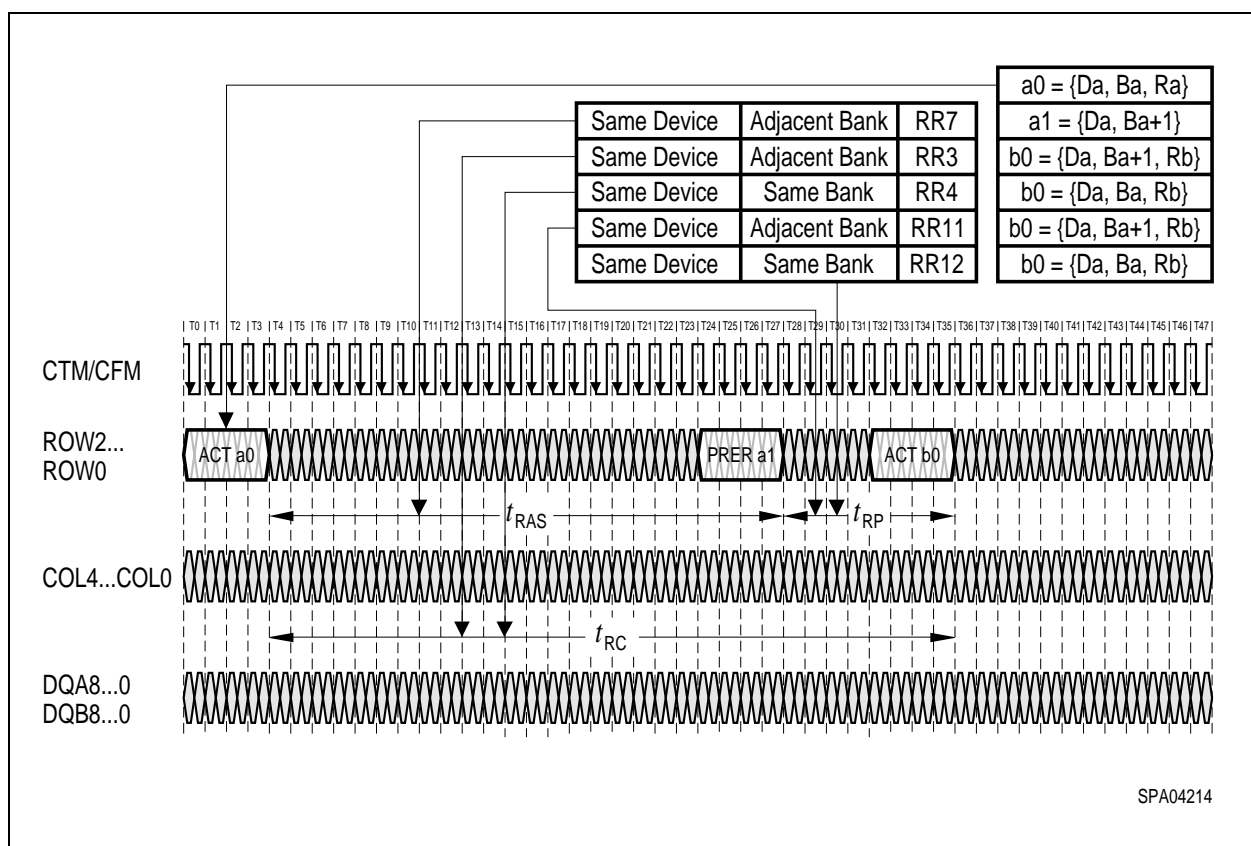
<sup>1)</sup> This is any command which permits the write buffer of device Da to retire (see **Table 9**). “Ba” is the bank address in the write buffer.

<sup>2)</sup> This situation is hazardous because the write buffer will be left unretired while the targeted bank is precharged. See **Figure 19**.

### ROW-to-ROW Examples

**Figure 10** shows examples of some of the ROW-to-ROW packet spacings from **Table 11**. A complete sequence of activate and precharge commands is directed to a bank. The RR8 and RR12 rules apply to this sequence. In addition to satisfying the  $t_{RAS}$  and  $t_{RP}$  timing parameters, the separation between ACT commands to the same bank must also satisfy the  $t_{RC}$  timing parameter (RR4).

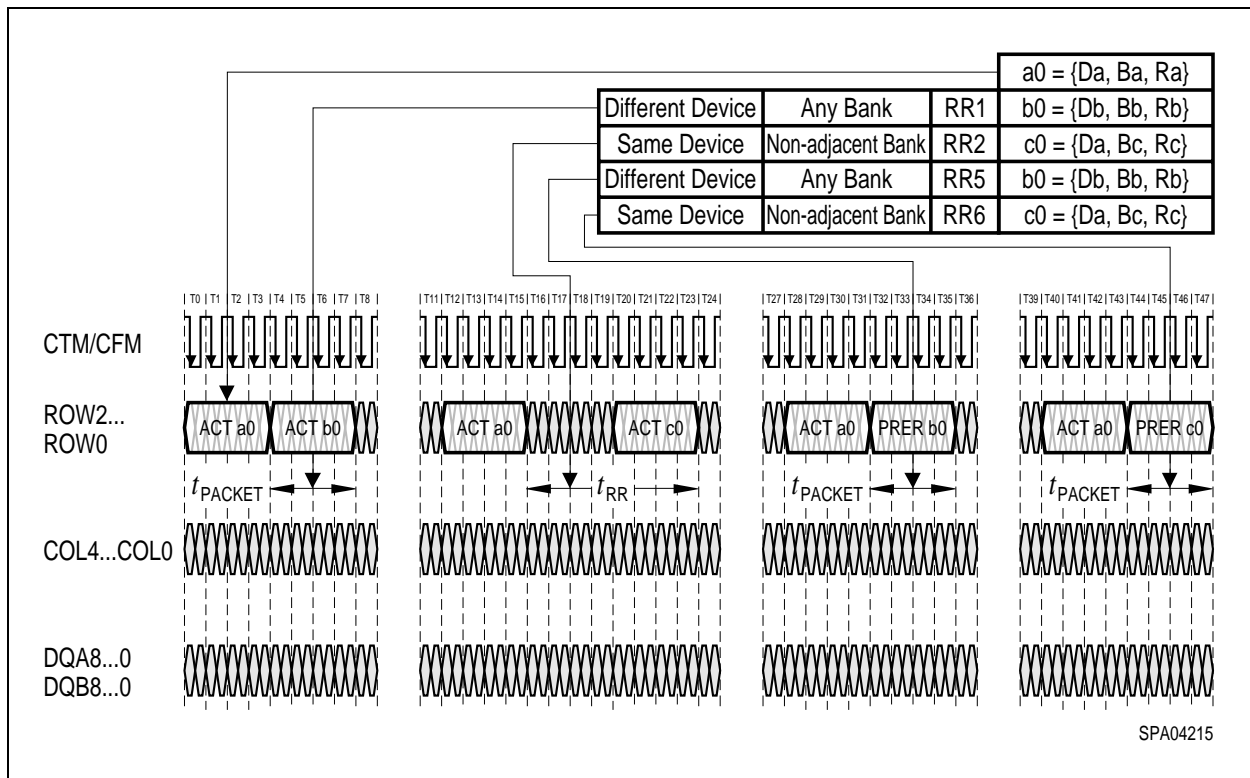
When a bank is activated, it is necessary for adjacent banks to remain precharged. As a result, the adjacent banks will also satisfy parallel timing constraints; in the example, the RR11 and RR3 rules are analogous to the RR12 and RR4 rules.



**Figure 10 Row Packet Example**

**Figure 11** shows examples of the ACT-to-ACT (RR1, RR2) and ACT-to-PRER (RR5, RR6) command spacings from **Table 11**. In general, the commands in ROW packets may be spaced an interval  $t_{PACKET}$  apart unless they are directed to the same or adjacent banks or unless they are a similar command type (both PRER or both ACT) directed to the same device.





**Figure 11** Row Packet Example

**Figure 12** shows examples of the PRER-to-PRER (RR13, RR14) and PRER-to-ACT (RR9, RR10) command spacings from **Table 12**. The RR15 and RR16 cases (PRER-to-PRER to same or adjacent banks) are not shown, but are similar to RR14. In general, the commands in ROW packets may be spaced an interval  $t_{\text{PACKET}}$  apart unless they are directed to the same or adjacent banks or unless they are a similar command type (both PRER or both ACT) directed to the same device.

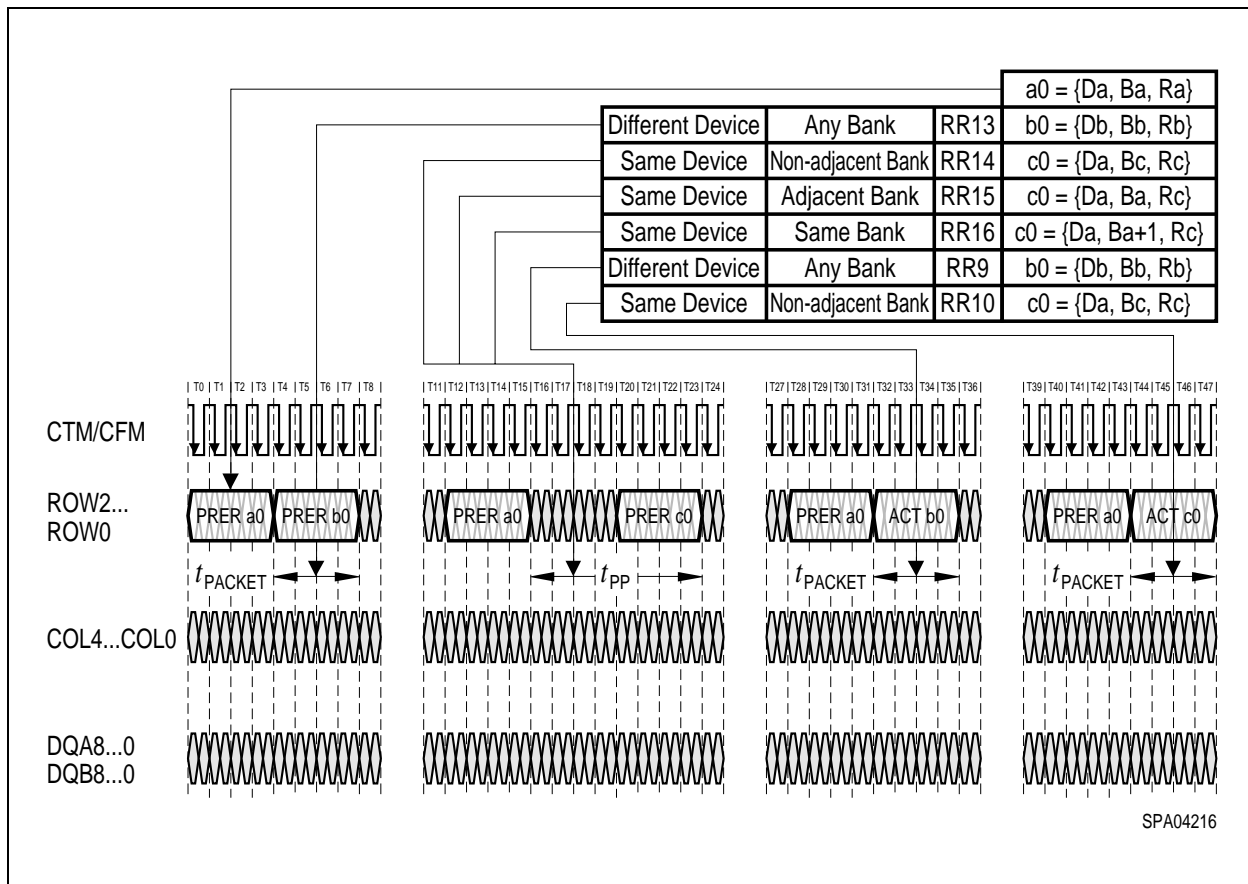


Figure 12 Row Packet Examples

### Row and Column Cycle Description

**Activate:** A row cycle begins with the activate (ACT) operation. The activation process is destructive; the act of sensing the value of a bit in a bank's storage cell transfers the bit to the sense amp, but leaves the original bit in the storage cell with an incorrect value.

**Restore:** Because the activation process is destructive, a hidden operation called restore is automatically performed. The restore operation rewrites the bits in the sense amp back into the storage cells of the activated row of the bank.

**Read/Write:** While the restore operation takes place, the sense amp may be read (RD) and written (WR) using column operations. If new data is written into the sense amp, it is automatically forwarded to the storage cells of the bank so the data in the activated row and the data in the sense amp remain identical.

**Precharge:** When both the restore operation and the column operations are completed, the sense amp and bank are precharged (PRE). This leaves them in the proper state to begin another activate operation.

**Intervals:** The activate operation requires the interval  $t_{RCD,MIN}$  to complete. The hidden restore operation requires the interval  $t_{RAS,MIN} - t_{RCD,MIN}$  to complete. Column read and write operations are also performed during the  $t_{RAS,MIN} - t_{RCD,MIN}$  interval (if more than about four column operations are

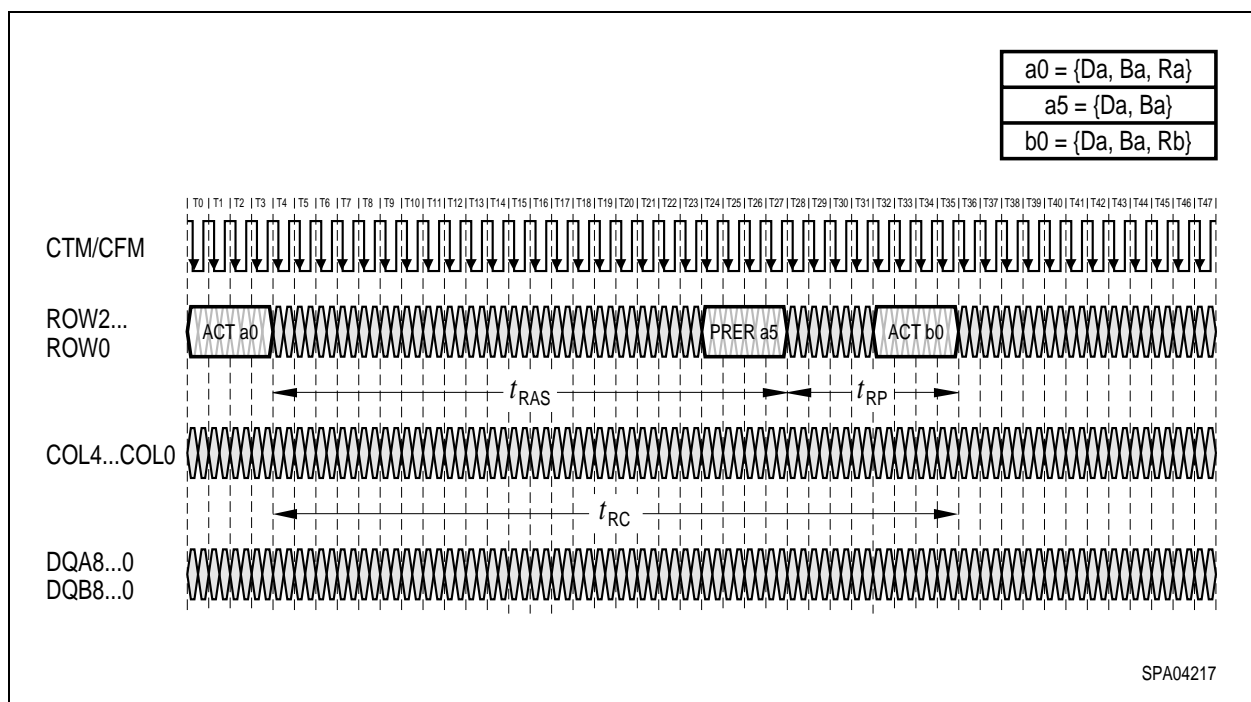
performed, this interval must be increased). The precharge operation requires the interval  $t_{RP,MIN}$  to complete.

**Adjacent Banks:** An RDRAM with a “s” designation (256K × 32s × 16/18) indicates it contains “split banks”. This means the sense amps are shared between two adjacent banks. The only exception is that sense amp 0, 15, 30, and 31 are not shared. When a row in a bank is activated, the two adjacent sense amps are connected to (associated with) that bank and are not available for use by the two adjacent banks. These two adjacent banks must remain precharged while the selected bank goes through its activate, restore, read/write, and precharge operations.

For example (referring to the block diagram of **Figure 2**), if bank 5 is accessed, sense amp 4/5 and sense amp 5/6 will both be loaded with one of the 512 rows (with 512 bytes loaded into each sense amp from the 1 Kbyte row - 256 bytes to the DQA side and 256 bytes to the DQB side). While this row from bank 5 is being accessed, no rows may be accessed in banks 4 or 6 because of the sense amp sharing.

### Precharge Mechanisms

**Figure 13** shows an example of precharge with the ROWR packet mechanism. The PRER command must occur a time  $t_{RAS}$  after the ACT command, and a time  $t_{RP}$  before the next ACT command. This timing will serve as a baseline against which the other precharge mechanisms can be compared.



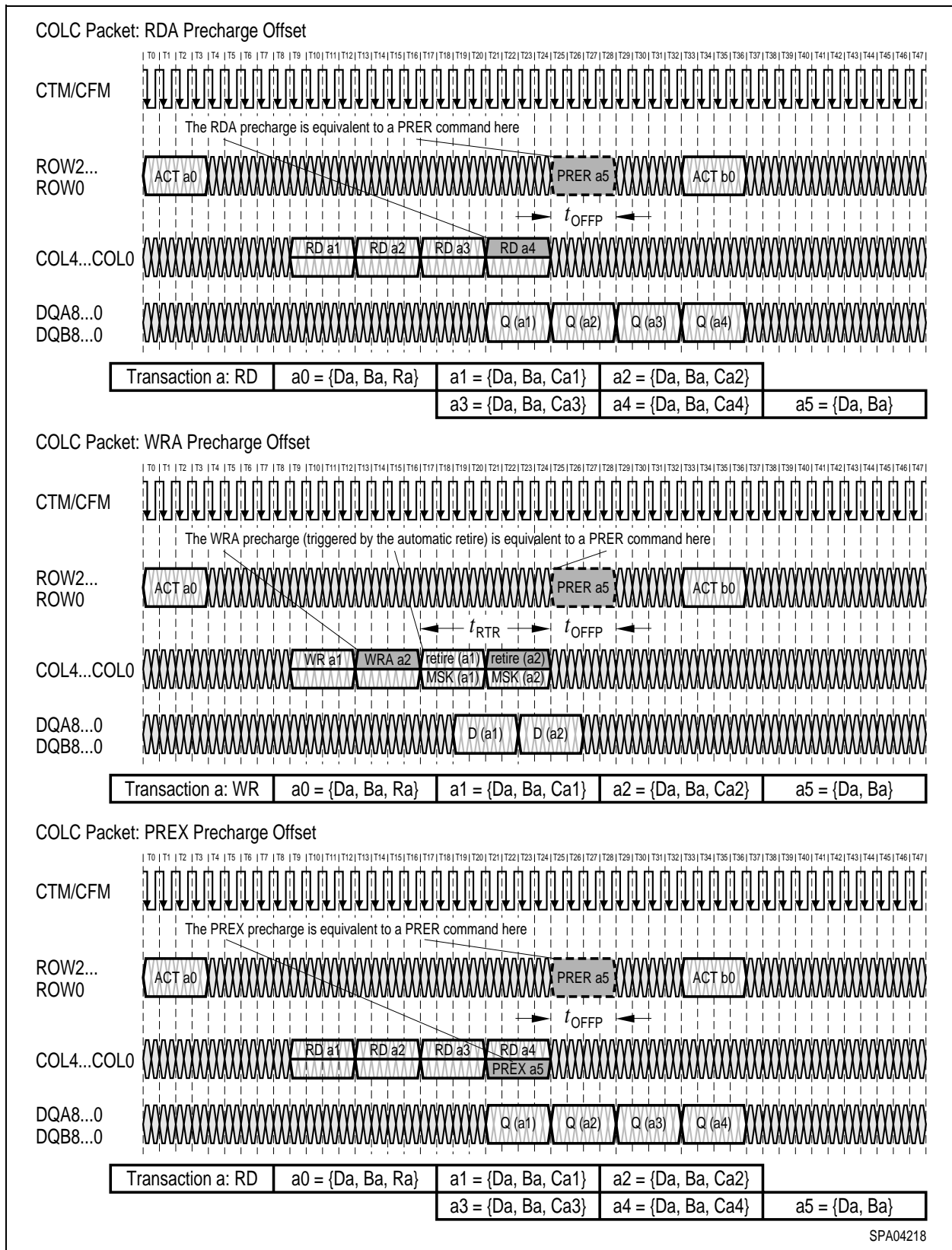
**Figure 13 Precharge via PRER Command in ROWR Packet**

**Figure 14** (top) shows an example of precharge with a RDA command. A bank is activated with an ROWA packet on the ROW pins. Then, a series of four dualocts are read with RD commands in COLC packets on the COL pins. The fourth of these commands is a RDA, which causes the bank to automatically precharge when the final read has finished. The timing of this automatic precharge is equivalent to a PRER command in an ROWR packet on the ROW pins that is offset a time  $t_{OFFP}$

from the COLC packet with the RDA command. The RDA command should be treated as a RD command in a COLC packet as well as a simultaneous (but offset) PRER command in an ROWR packet when analyzing interactions with other packets.

**Figure 14** (middle) shows an example of precharge with a WRA command. As in the RDA example, a bank is activated with an ROWA packet on the ROW pins. Then, two dualocts are written with WR commands in COLC packets on the COL pins. The second of these commands is a WRA, which causes the bank to automatically precharge when the final write has been retired. The timing of this automatic precharge is equivalent to a PRER command in an ROWR packet on the ROW pins that is offset a time  $t_{OFFP}$  from the COLC packet that causes the automatic retire. The WRA command should be treated as a WR command in a COLC packet as well as a simultaneous (but offset) PRER command in an ROWR packet when analyzing interactions with other packets. Note that the automatic retire is triggered by a COLC packet a time  $t_{RTR}$  after the COLC packet with the WR command unless the second COLC contains a RD command to the same device. This is described in more detail in **Figure 17**.

**Figure 14** (bottom) shows an example of precharge with a PREX command in an COLX packet. A bank is activated with an ROWA packet on the ROW pins. Then, a series of four dualocts are read with RD commands in COLC packets on the COL pins. The fourth of these COLC packets includes an COLX packet with a PREX command. This causes the bank to precharge with timing equivalent to a PRER command in an ROWR packet on the ROW pins that is offset a time  $t_{OFFP}$  from the COLX packet with the PREX command.



**Figure 14** Offsets for Alternate Precharge Mechanisms

### Read Transaction - Example

**Figure 15** shows an example of a read transaction. It begins by activating a bank with an ACT a0 command in an ROWA packet. A time  $t_{RCD}$  later a RD a1 command is issued in a COLC packet. Note that the ACT command includes the device, bank, and row address (abbreviated as a0) while the RD command includes device, bank, and column address (abbreviated as a1). A time  $t_{CAC}$  after the RD command the read data dualoct Q(a1) is returned by the device. Note that the packets on the ROW and COL pins use the end of the packet as a timing reference point, while the packets on the DQA/DQB pins use the beginning of the packet as a timing reference point.

A time  $t_{CC}$  after the first COLC packet on the COL pins a second is issued. It contains a RD a2 command. The a2 address has the same device and bank address as the a1 address (and a0 address), but a different column address. A time  $t_{CAC}$  after the second RD command a second read data dualoct Q(a2) is returned by the device.

Next, a PRER a3 command is issued in an ROWR packet on the ROW pins. This causes the bank to precharge so that a different row may be activated in a subsequent transaction or so that an adjacent bank may be activated. The a3 address includes the same device and bank address as the a0, a1, and a2 addresses. The PRER command must occur a time  $t_{RAS}$  or more after the original ACT command (the activation operation in any DRAM is destructive, and the contents of the selected row must be restored from the two associated sense amps of the bank during the  $t_{RAS}$  interval). The PRER command must also occur a time  $t_{RDP}$  or more after the last RD command. Note that the  $t_{RDP}$  value shown is greater than the  $t_{RDP,MIN}$  specification in **Table 23**. This transaction example reads two dualocts, but there is actually enough time to read three dualocts before  $t_{RDP}$  becomes the limiting parameter rather than  $t_{RAS}$ . If four dualocts were read, the packet with PRER would need to shift right (be delayed) by one  $t_{CYCLE}$  (note - this case is not shown).

Finally, an ACT b0 command is issued in an ROWR packet on the ROW pins. The second ACT command must occur a time  $t_{RC}$  or more after the first ACT command and a time  $t_{RP}$  or more after the PRER command. This ensures that the bank and its associated sense amps are precharged. This example assumes that the second transaction has the same device and bank address as the first transaction, but a different row address. Transaction b may not be started until transaction a has finished. However, transactions to other banks or other devices may be issued during transaction a

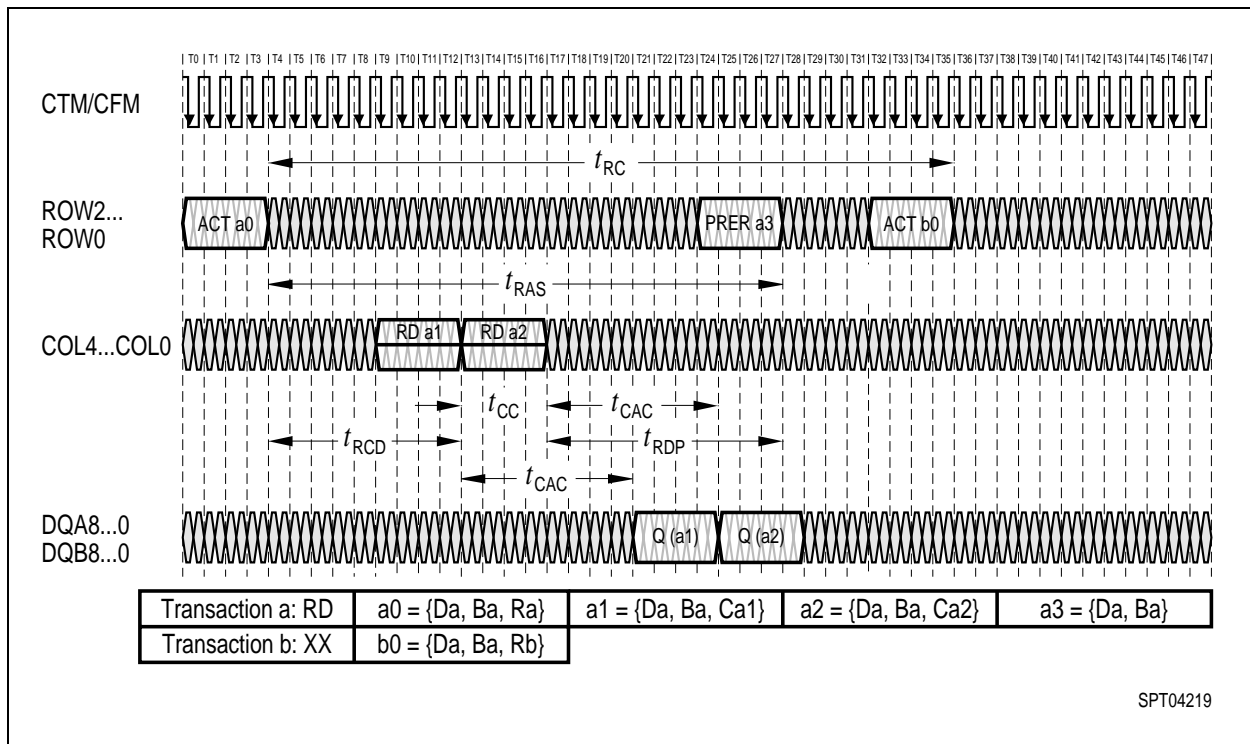


Figure 15 Read Transaction Example

### Write Transaction - Example

Figure 16 shows an example of a write transaction. It begins by activating a bank with an ACT a0 command in an ROWA packet. A time  $t_{RCD} - t_{RTR}$  later a WR a1 command is issued in a COLC packet (note that the  $t_{RCD}$  interval is measured to the end of the COLC packet with the first retire command). Note that the ACT command includes the device, bank, and row address (abbreviated as a0) while the WR command includes device, bank, and column address (abbreviated as a1). A time  $t_{CWD}$  after the WR command the write data dualoct D(a1) is issued. Note that the packets on the ROW and COL pins use the end of the packet as a timing reference point, while the packets on the DQA/DQB pins use the beginning of the packet as a timing reference point.

A time  $t_{CC}$  after the first COLC packet on the COL pins a second COLC packet is issued. It contains a WR a2 command. The a2 address has the same device and bank address as the a1 address (and a0 address), but a different column address. A time  $t_{CWD}$  after the second WR command a second write data dualoct D(a2) is issued.

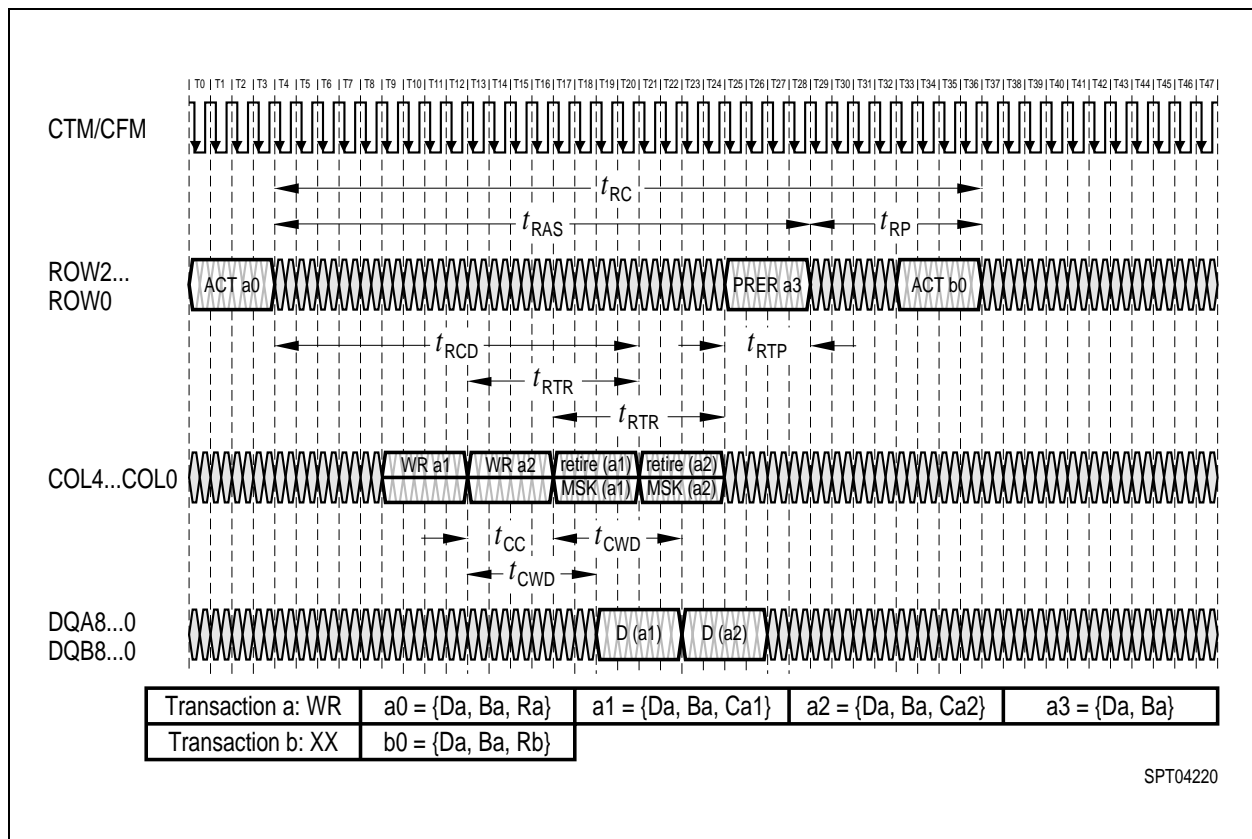
A time  $t_{RTR}$  after each WR command an optional COLM packet MSK (a1) is issued, and at the same time a COLC packet is issued causing the write buffer to automatically retire. See Figure 17 for more detail on the write/retire mechanism. If a COLM packet is not used, all data bytes are unconditionally written. If the COLC packet which causes the write buffer to retire is delayed, then the COLM packet (if used) must also be delayed.

Next, a PRER a3 command is issued in an ROWR packet on the ROW pins. This causes the bank to precharge so that a different row may be activated in a subsequent transaction or so that an adjacent bank may be activated. The a3 address includes the same device and bank address as the a0, a1, and a2 addresses. The PRER command must occur a time  $t_{RAS}$  or more after the original

ACT command (the activation operation in any DRAM is destructive, and the contents of the selected row must be restored from the two associated sense amps of the bank during the  $t_{RAS}$  interval).

A PRER a3 command is issued in an ROWR packet on the ROW pins. The PRER command must occur a time  $t_{RTP}$  or more after the last COLC which causes an automatic retire.

Finally, an ACT b0 command is issued in an ROWR packet on the ROW pins. The second ACT command must occur a time  $t_{RC}$  or more after the first ACT command and a time  $t_{RP}$  or more after the PRER command. This ensures that the bank and its associated sense amps are precharged. This example assumes that the second transaction has the same device and bank address as the first transaction, but a different row address. Transaction b may not be started until transaction a has finished. However, transactions to other banks or other devices may be issued during transaction a.



**Figure 16 Write Transaction Example**

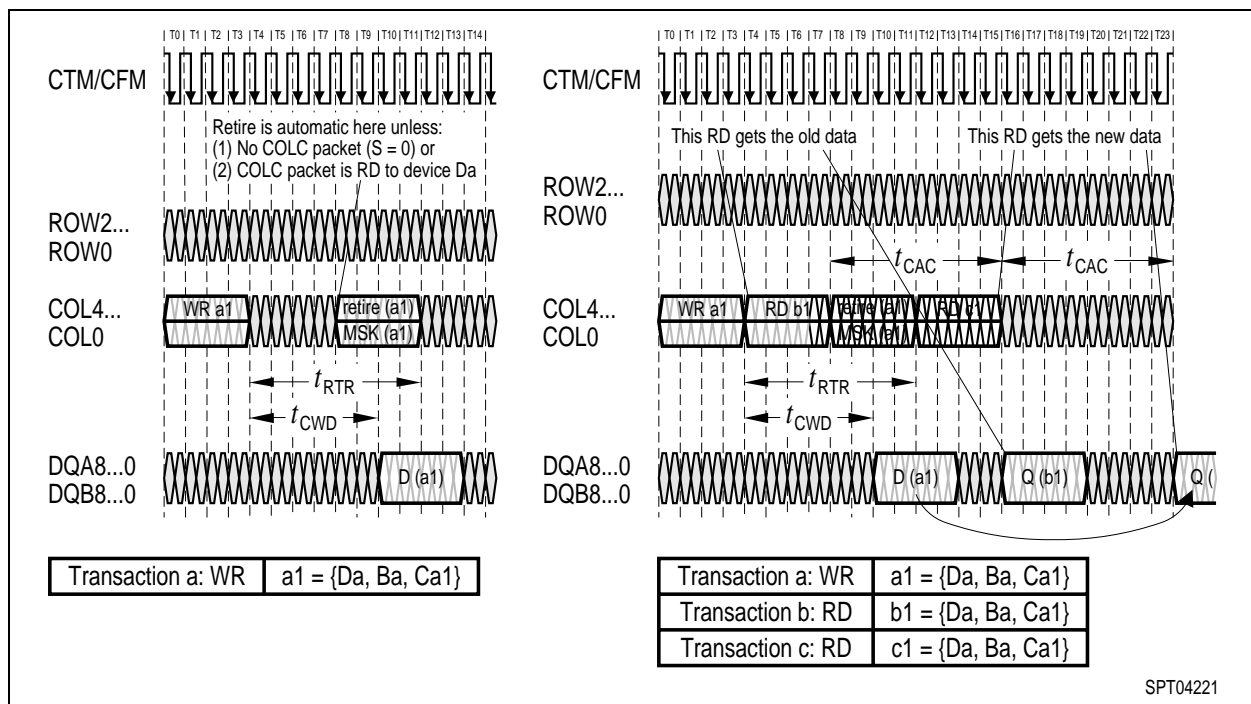


### Write/Retire - Examples

The process of writing a dualoct into a sense amp of an RDRAM bank occurs in two steps. The first step consists of transporting the write command, write address, and write data into the write buffer. The second step happens when the RDRAM automatically retires the write buffer (with an optional bytemask) into the sense amp. This two-step write process reduces the natural turn-around delay due to the internal bidirectional data pins.

**Figure 17** (left) shows an example of this two step process. The first COLC packet contains the WR command and an address specifying device, bank and column. The write data dualoct follows a time  $t_{CWD}$  later. This information is loaded into the write buffer of the specified device. The COLC packet which follows a time  $t_{RTR}$  later will retire the write buffer. The retire will happen automatically unless (1) a COLC packet is not framed (no COLC packet is present and the S bit is zero), or (2) the COLC packet contains a RD command to the same device. If the retire does not take place at time  $t_{RTR}$  after the original WR command, then the device continues to frame COLC packets, looking for the first that is not a RD directed to itself. A bytemask MSK(a1) may be supplied in a COLM packet aligned with the COLC that retires the write buffer at time  $t_{RTR}$  after the WR command.

The memory controller must be aware of this two-step write/retire process. Controller performance can be improved, but only if the controller design accounts for several side effects.

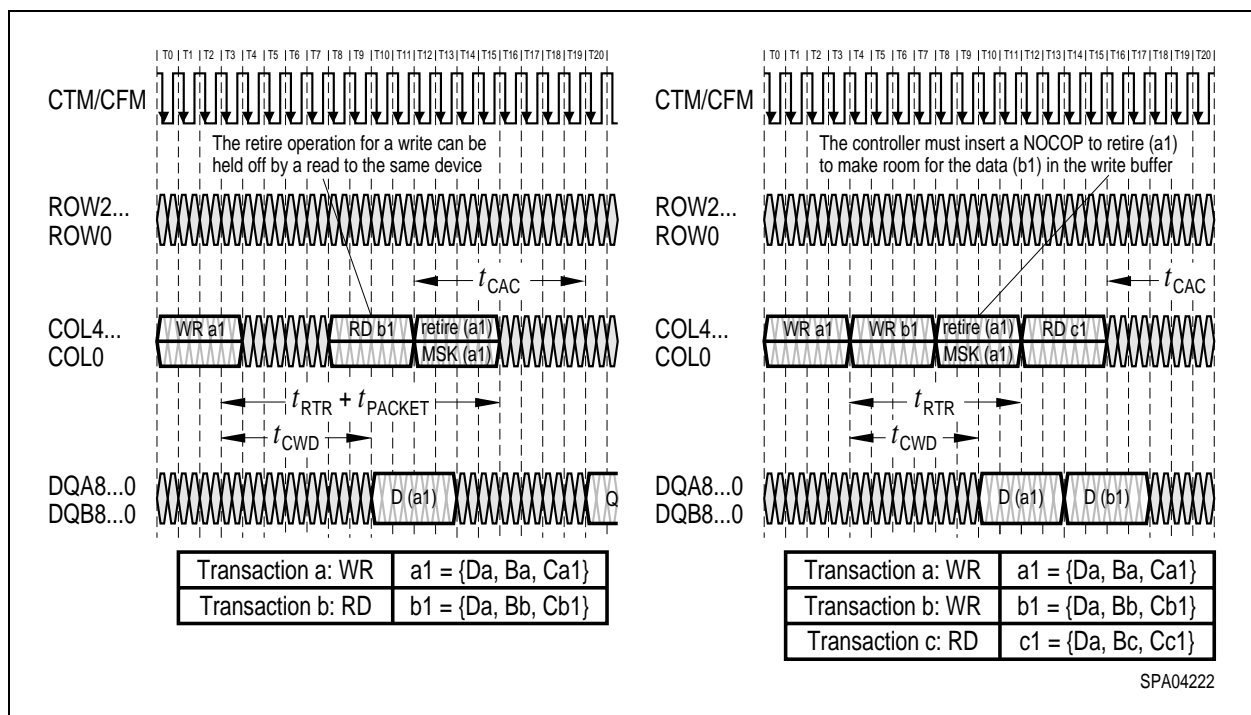


**Figure 17 Normal Retire (left) and Retire/Read Ordering (right)**

**Figure 17** (right) shows the first of these side effects. The first COLC packet has a WR command which loads the address and data into the write buffer. The third COLC causes an automatic retire of the write buffer to the sense amp. The second and fourth COLC packets (which bracket the retire packet) contain RD commands with the same device, bank and column address as the original WR command. In other words, the same dualoct address that is written is read both before and after it is actually retired. The first RD returns the old dualoct value from the sense amp before it is overwritten. The second RD returns the new dualoct value that was just written.

**Figure 18** (left) shows the result of performing a RD command to the same device in the same COLC packet slot that would normally be used for the retire operation. The read may be to any bank and column address; all that matters is that it is to the same device as the WR command. The retire operation and MSK(a1) will be delayed by a time  $t_{\text{PACKET}}$  as a result. If the RD command used the same bank and column address as the WR command, the old data from the sense amp would be returned. If many RD commands to the same device were issued instead of the single one that is shown, then the retire operation would be held off an arbitrarily long time. However, once a RD to another device or a WR or NOCOP to any device is issued, the retire will take place. **Figure 18** (right) illustrates a situation in which the controller wants to issue a WR-WR-RD COLC packet sequence, with all commands addressed to the same device, but addressed to any combination of banks and columns.

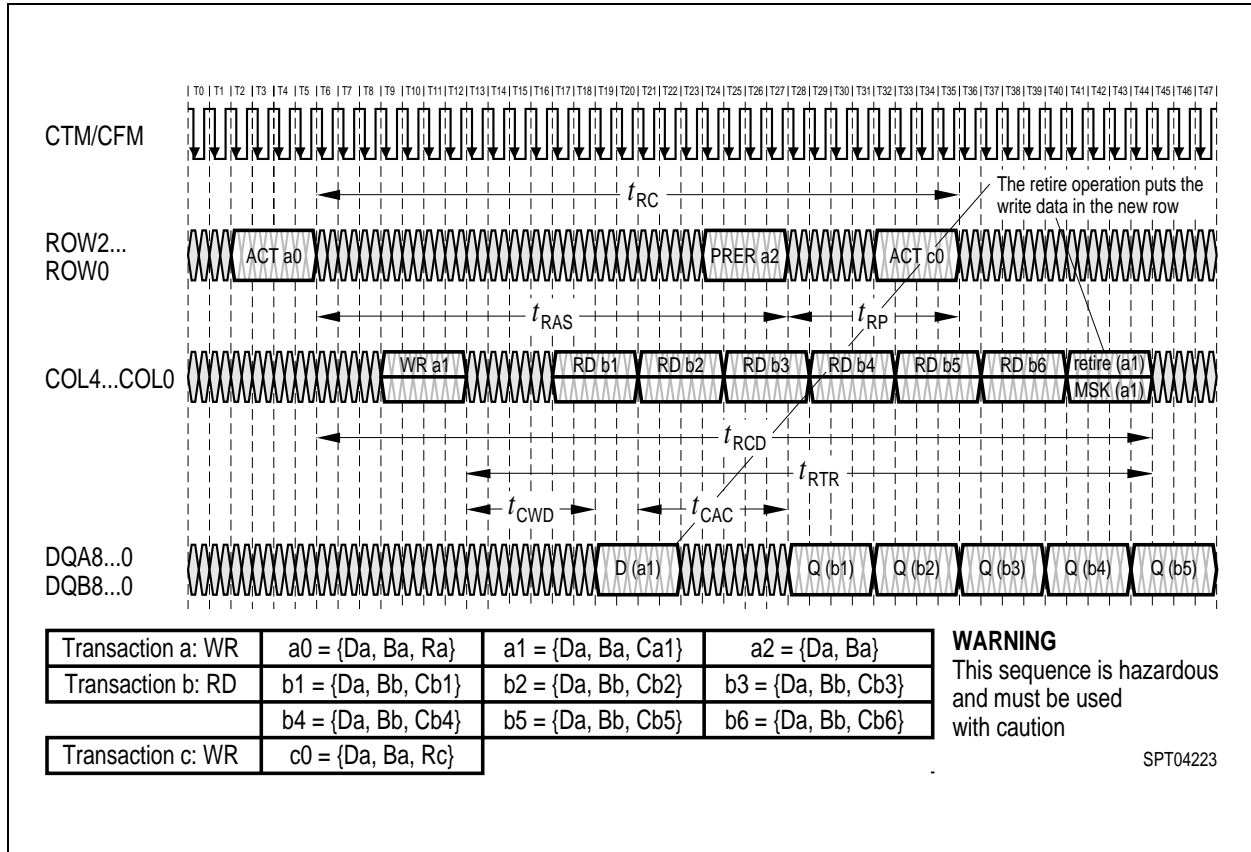
The RD will prevent a retire of the first WR from automatically happening. But the first dualoct D(a1) in the write buffer will be overwritten by the second WR dualoct D(b1) if the RD command is issued in the third COLC packet. Therefore, it is required in this situation that the controller issue a NOCOP command in the third COLC packet, delaying the RD command by a time of  $t_{\text{PACKET}}$ . This situation is explicitly shown in **Table 13** for the cases in which  $t_{\text{CCDELAY}}$  is equal to  $t_{\text{RTR}}$ .



**Figure 18 Retire Held Off by Read (left) and Controller Forces WWR Gap (right)**

**Figure 19** shows a possible result when a retire is held off for a long time (an extended version of **Figure 18**-left). After a WR command, a series of six RD commands are issued to the same device (but to any combination of bank and column addresses). In the meantime, the bank Ba to which the WR command was originally directed is precharged, and a different row Rc is activated. When the retire is automatically performed, it is made to this new row, since the write buffer only contains the bank and column address, not the row address. The controller can insure that this doesn't happen by never precharging a bank with an unretired write buffer. Note that in a system with more than one RDRAM, there will never be more than two RDRAMs with unretired write buffers. This is because

a WR command issued to one device automatically retires the write buffers of all other devices written a time  $t_{RTR}$  before or earlier.



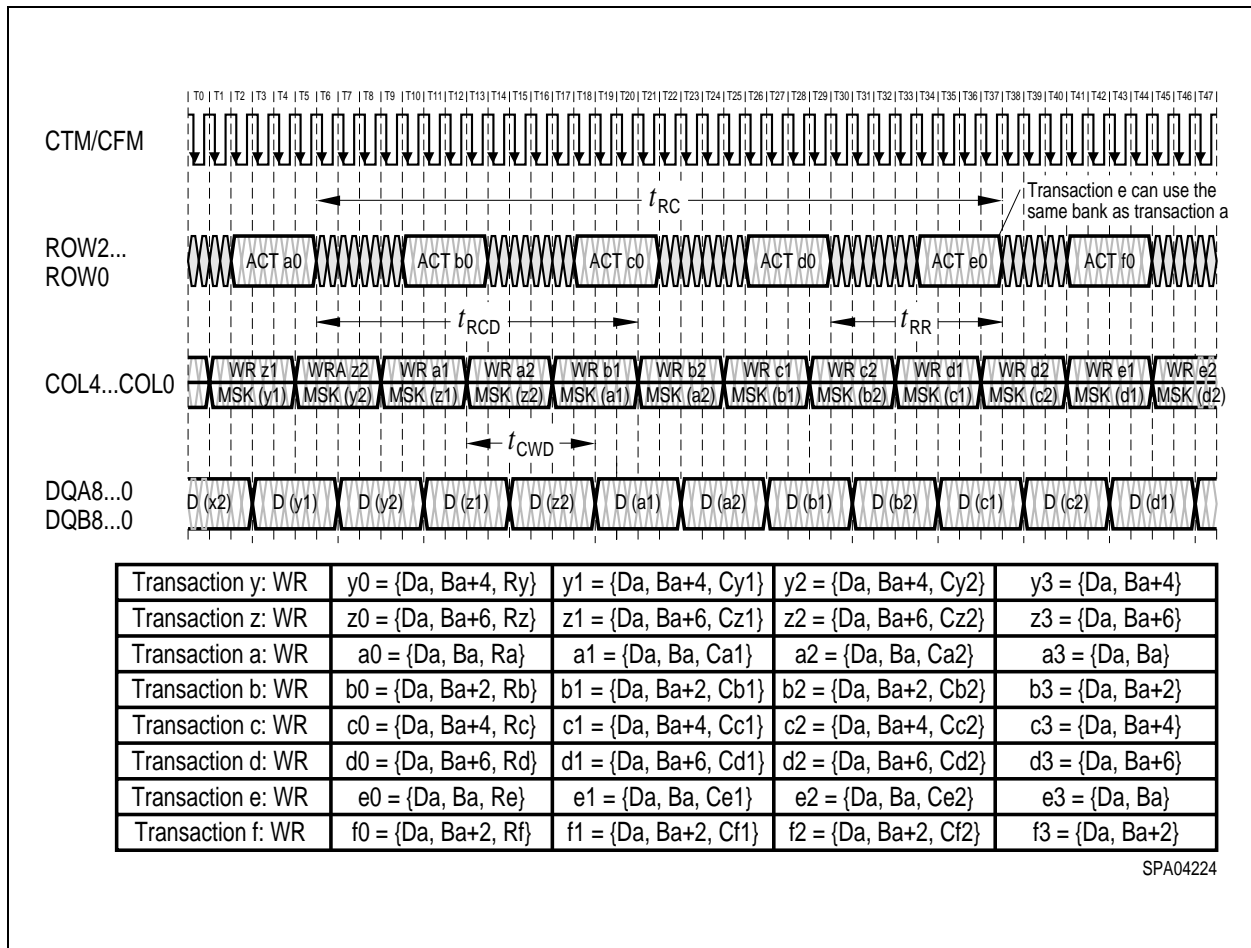
**Figure 19 Retire Held Off by Reads to Same Device, Write Buffer Retired to New Row**

**Interleaved Write - Example**

**Figure 20** shows an example of an interleaved write transaction. Transactions similar to the one presented in **Figure 16** are directed to non-adjacent banks of a single RDRAM. This allows a new transaction to be issued once every  $t_{RR}$  interval rather than once every  $t_{RC}$  interval (four times more often). The DQ data pin efficiency is 100% with this sequence.

With two dualocts of data written per transaction, the COL, DQA, and DQB pins are fully utilized. Banks are precharged using the WRA autoprecharge option rather than the PRER command in an ROWR packet on the ROW pins.

In this example, the first transaction is directed to device Da and bank Ba. The next three transactions are directed to the same device Da, but need to use different, non-adjacent banks Bb, Bc, Bd so there is no bank conflict. The fifth transaction could be redirected back to bank Ba without interference, since the first transaction would have completed by then ( $t_{RC}$  has elapsed). Each transaction may use any value of row address (Ra, Rb, ...) and column address (Ca1, Ca2, Cb1, Cb2, ...).



**Figure 20 Interleaved Write Transaction with Two Dualoct Data Length**

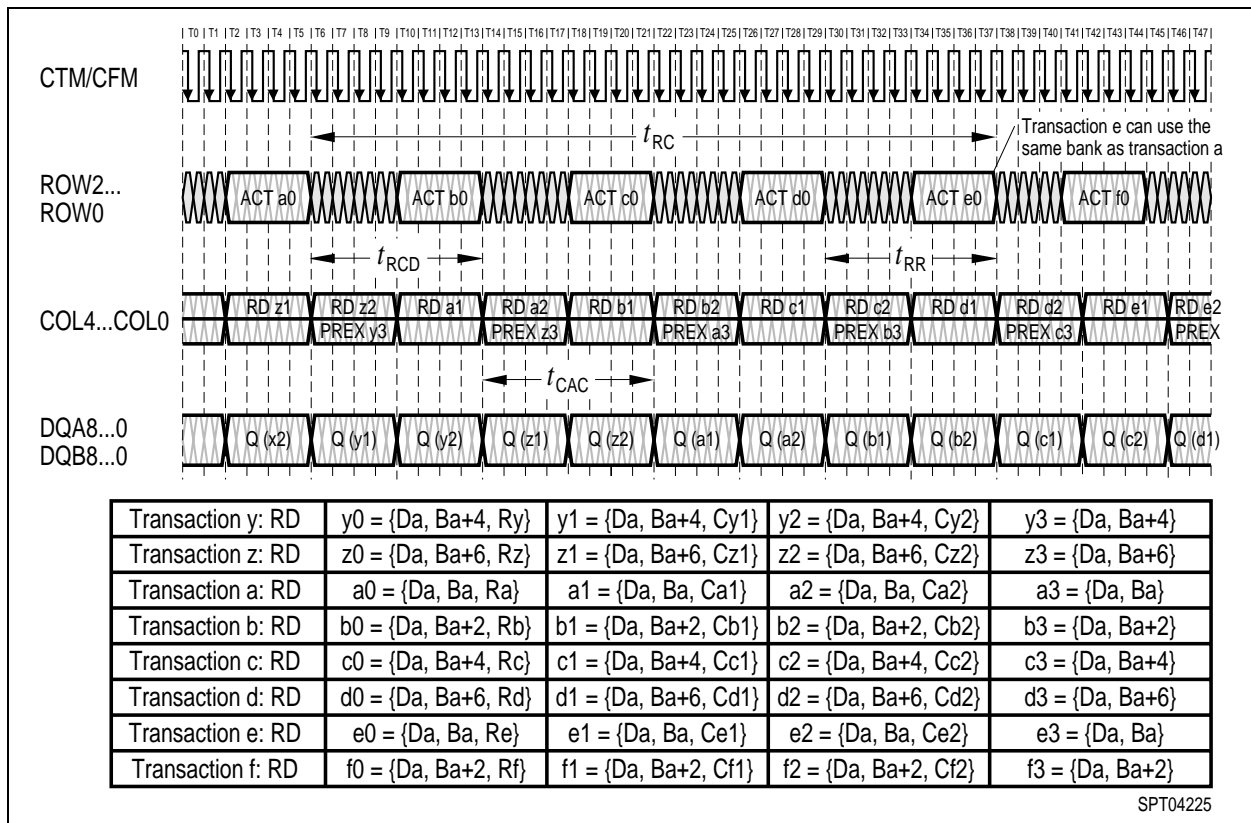
**Interleaved Read - Example**

**Figure 21** shows an example of interleaved read transactions. Transactions similar to the one presented in **Figure 15** are directed to non-adjacent banks of a single RDRAM. The address sequence is identical to the one used in the previous write example. The DQ data pins efficiency is also 100%. The only difference with the write example (aside from the use of the RD command rather than the WR command) is the use of the PREX command in a COLX packet to precharge the banks rather than the RDA command. This is done because the PREX is available for a read transaction but is not available for a masked write transaction.

**Interleaved RRWW - Example**

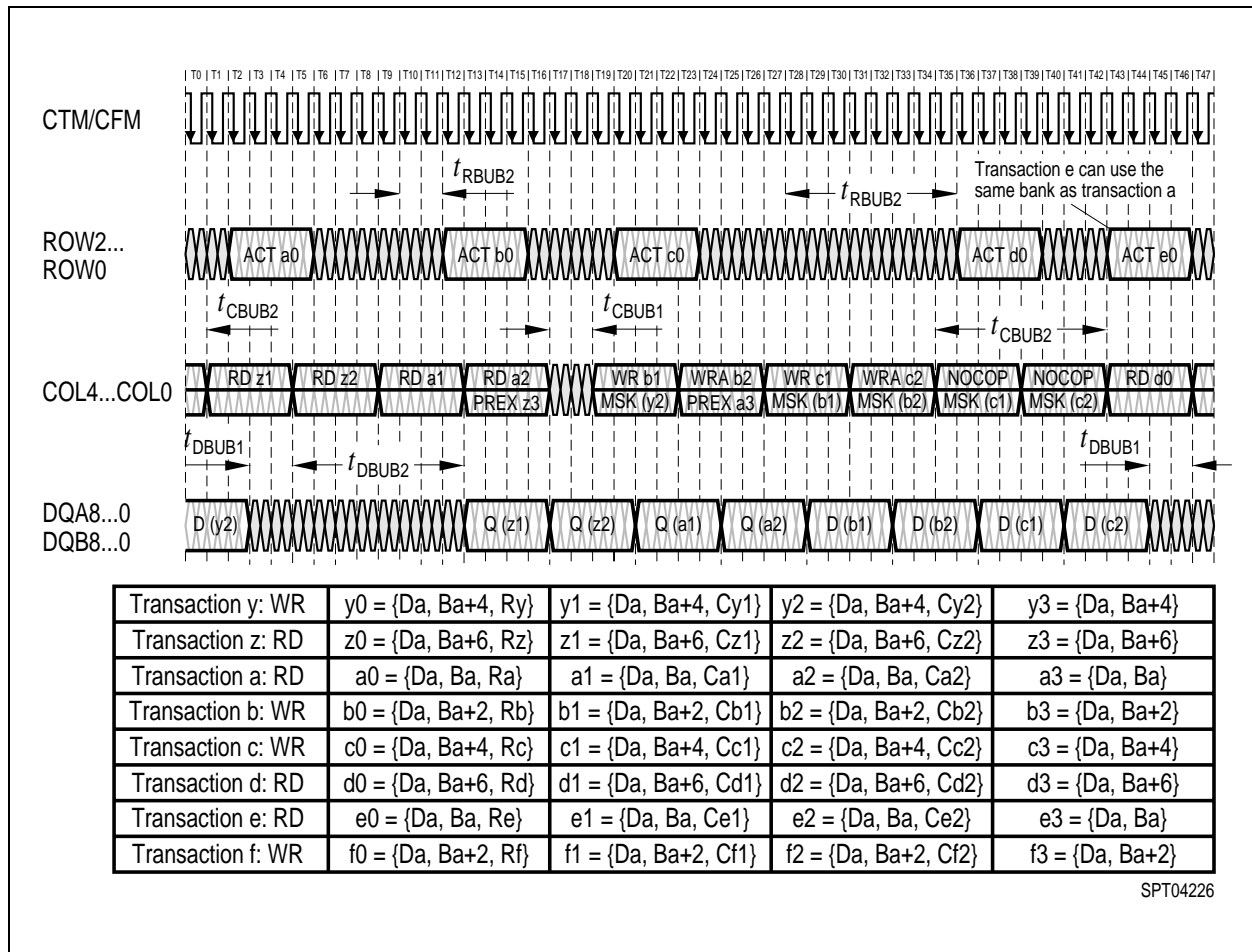
**Figure 22** shows a steady-state sequence of 2-dualloct RD/RD/WR/WR... transactions directed to non-adjacent banks of a single RDRAM. This is similar to the interleaved write and read examples in **Figure 20** and **Figure 21** except that bubble cycles need to be inserted by the controller at read/write boundaries. The DQ data pin efficiency for the example in **Figure 22** is 32/42 or 76%. If there were more RDRAMs on the Channel, the DQ pin efficiency would approach 32/34 or 94% for the two-dualloct RRWW sequence (this case is not shown).

In **Figure 22**, the first bubble type  $t_{CBUB1}$  is inserted by the controller between a RD and WR command on the COL pins. This bubble accounts for the round-trip propagation delay that is seen by read data, and is explained in detail in **Figure 4**. This bubble appears on the DQA and DQB pins as  $t_{DBUB1}$  between a write data dualloct D and read data dualloct Q. This bubble also appears on the ROW pins as  $t_{RBUB1}$ .



**Figure 21 Interleaved Read Transaction with Two Dualloct Data Length**

The second bubble type  $t_{CBUB2}$  is inserted (as a NOCOP command) by the controller between a WR and RD command on the COL pins when there is a WR-WR-RD sequence to the same device. This bubble enables write data to be retired from the write buffer without being lost, and is explained in detail in **Figure 18**. There would be no bubble if address c0 and address d0 were directed to different devices. This bubble appears on the DQA and DQB pins as  $t_{DBUB2}$  between a write data dualoct D and read data dualoct Q. This bubble also appears on the ROW pins as  $t_{RBUB2}$ .

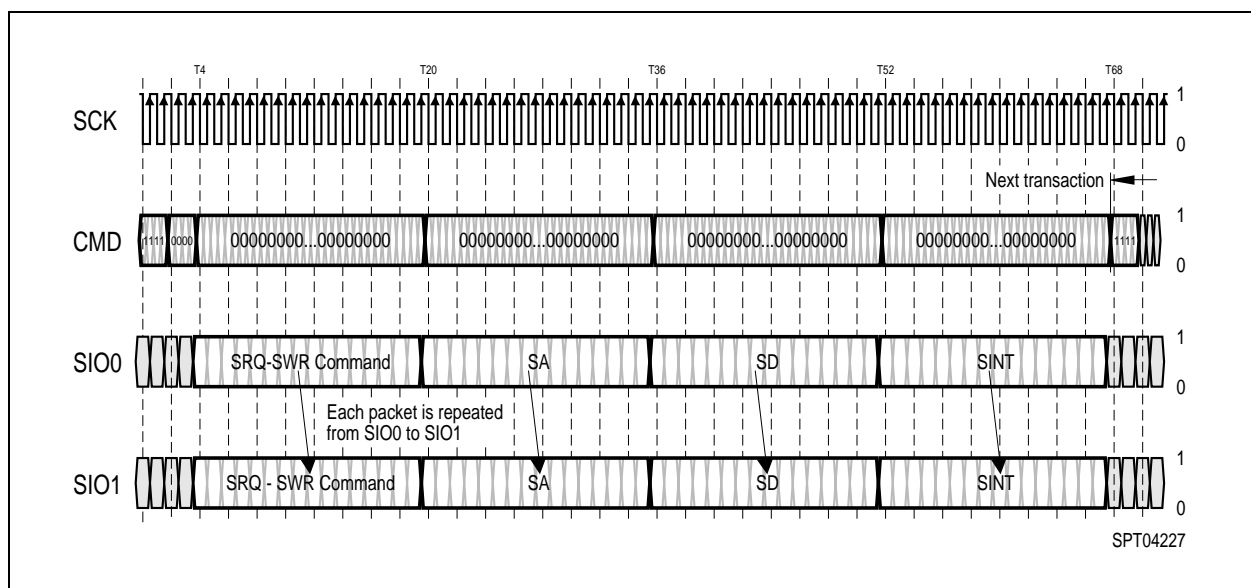


**Figure 22 Interleaved RRWW Sequence with Two Dualoct Data Length**

### Control Register Transactions

The RDRAM has two CMOS input pins SCK and CMD and two CMOS input/output pins SIO0 and SIO1. These provide serial access to a set of control registers in the RDRAM. These control registers provide configuration information to the controller during the initialization process. They also allow an application to select the appropriate operating mode of the RDRAM.

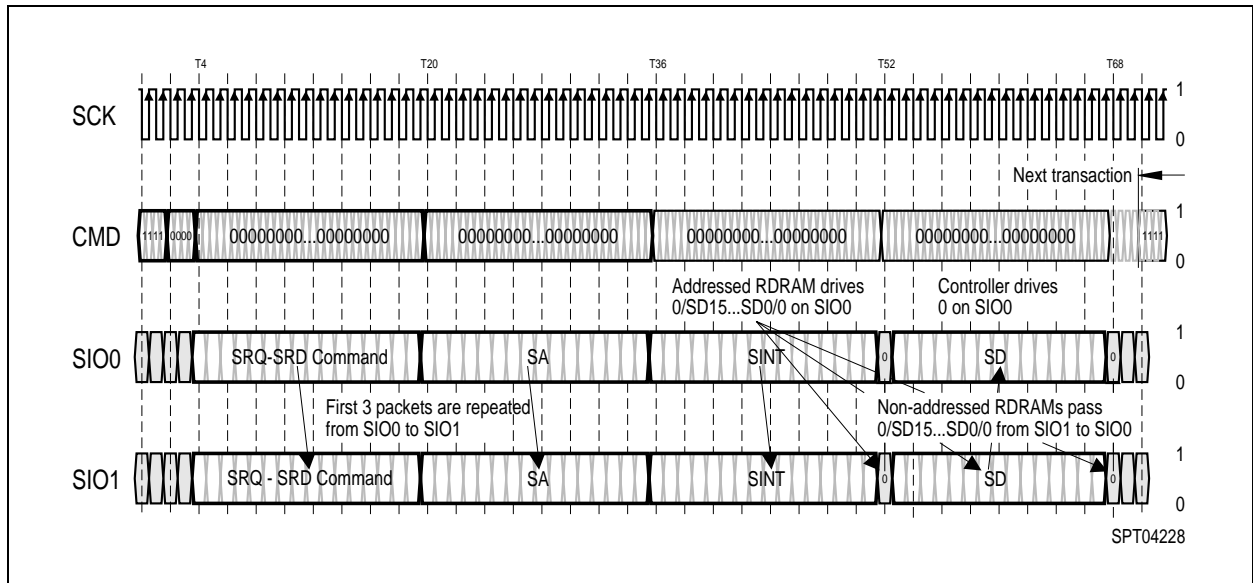
SCK (serial clock) and CMD (command) are driven by the controller to all RDRAMs in parallel. SIO0 and SIO1 are connected (in a daisy chain fashion) from one RDRAM to the next. In normal operation, the data on SIO0 is repeated on SIO1, which connects to SIO0 of the next RDRAM (the data is repeated from SIO1 to SIO0 for a read data packet). The controller connects to SIO0 of the first RDRAM.



**Figure 23 Serial Write (SWR) Transaction to Control Register**

Write and read transactions are each composed of four packets, as shown in **Figure 23** and **Figure 24**. Each packet consists of 16 bits, as summarized in **Figure 15** and **Figure 16**. The packet bits are sampled on the falling edge of SCK. A transaction begins with a SRQ (Serial Request) packet. This packet is framed with a 11110000 pattern on the CMD input (note that the CMD bits are sampled on both the falling edge and the rising edge of SCK). The SRQ packet contains the SOP3...SOP0 (Serial Opcode) field, which selects the transaction type. The SDEV5...SDEV0 (Serial Device address) selects one of the 32 RDRAMs. If SBC (Serial Broadcast) is set, then all RDRAMs are selected. The SA (Serial Address) packet contains a 12 bit address for selecting a control register.

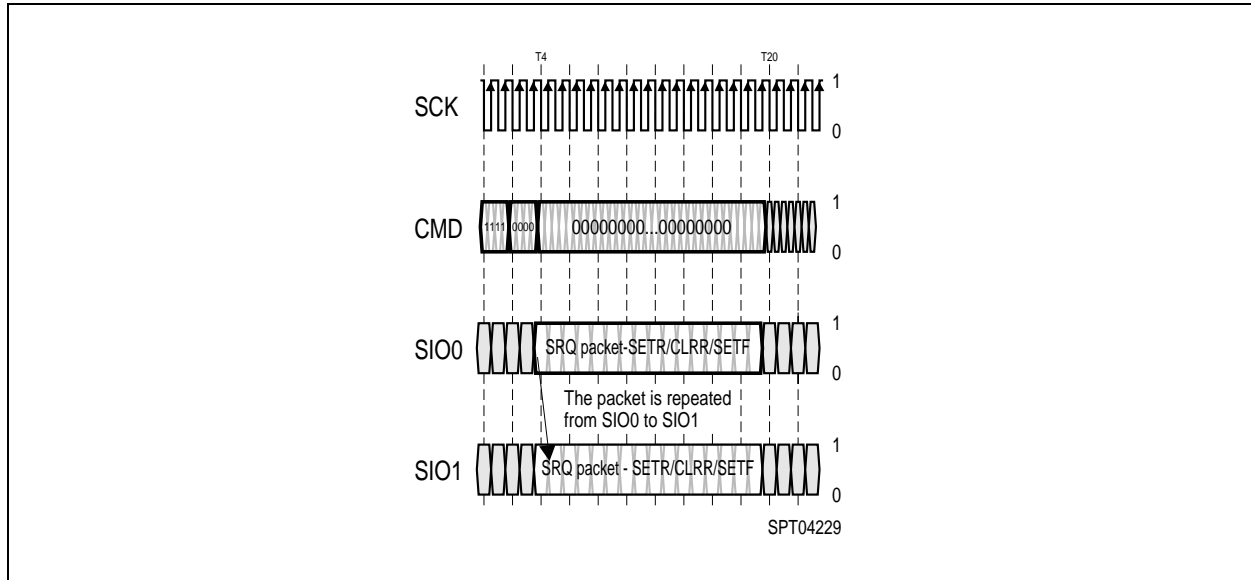
A write transaction has a SD (Serial Data) packet next. This contains 16 bits of data that is written into the selected control register. A SINT (Serial Interval) packet is last, providing some delay for any side-effects to take place. A read transaction has a SINT packet, then a SD packet. This provides delay for the selected RDRAM to access the control register. The SD read data packet travels in the opposite direction (towards the controller) from the other packet types. The SCK cycle time will accommodate the total delay.



**Figure 24 Serial Read (SRD) Transaction Control Register**



### Control Register Packets



**Figure 25 SETR, CLRR, SETF Transaction**

**Table 15** summarizes the formats of the four packet types for control register transactions. **Table 16** summarizes the fields that are used within the packets.

**Figure 25** shows the transaction format for the SETR, CLRR, and SETF commands. These transactions consist of a single SRQ packet, rather than four packets like the SWR and SRD commands. The same framing sequence on the CMD input is used, however. These commands are used during initialization prior to any control register read or write transactions.

**Table 15 Control Register Packet Formats**

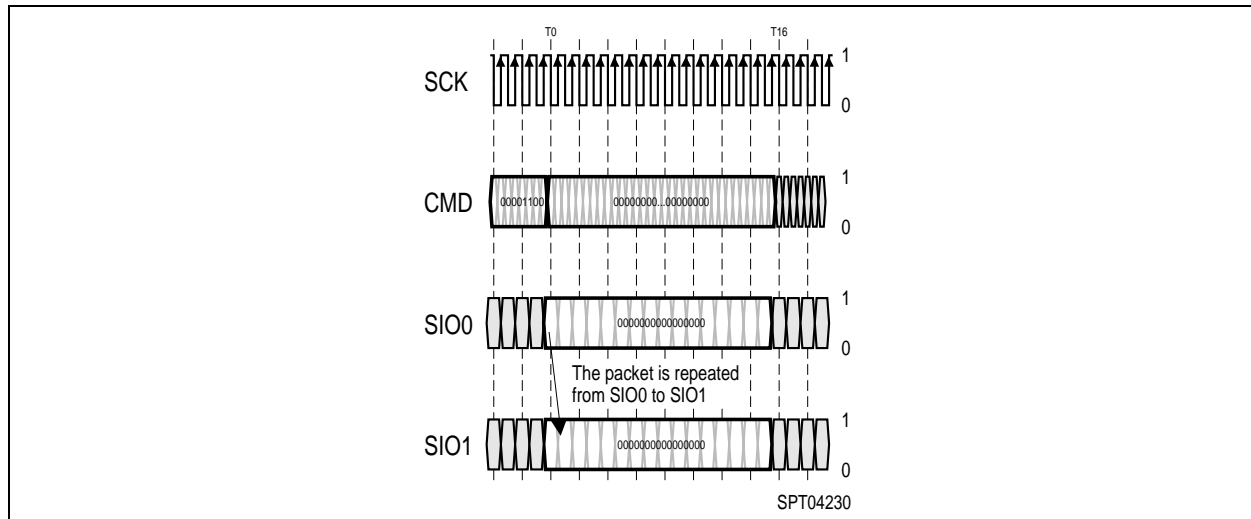
SCK Cycle	SIO0 or SIO1 for SRQ	SIO0 or SIO1 for SA	SIO0 or SIO1 for SINT	SIO0 or SIO1 for SD	SCK Cycle	SIO0 or SIO1 for SRQ	SIO0 or SIO1 for SA	SIO0 or SIO1 for SINT	SIO0 or SIO1 for SD
0	rsrv	rsrv	0	SD15	8	SOP1	SA7	0	SD7
1	rsrv	rsrv	0	SD14	9	SOP0	SA6	0	SD6
2	rsrv	rsrv	0	SD13	10	SBC	SA5	0	SD5
3	rsrv	rsrv	0	SD12	11	SDEV4	SA4	0	SD4
4	rsrv	SA11	0	SD11	12	SDEV3	SA3	0	SD3
5	SDEV5	SA10	0	SD10	13	SDEV2	SA2	0	SD2
6	SOP3	SA9	0	SD9	14	SDEV1	SA1	0	SD1
7	SOP2	SA8	0	SD8	15	SDEV0	SA0	0	SD0

**Table 16 Field Description for Control Register Packets**

Field	Description
rsrv	Reserved. Should be driven as "0" by controller.
SOP3 ... SOP0	0000 - SRD. Serial read of control register {SA11 ... SA0} of RDRAM {SDEV5 ... SDEV0}.
	0001 - SWR. Serial write of control register {SA11 ... SA0} of RDRAM {SDEV5 ... SDEV0}.
	0010 - SETR. Set Reset bit, all control registers assume their reset values. <sup>1)</sup> 16 $t_{\text{SCYCLE}}$ delay until CLRR command.
	0100 - SETF. Set fast (normal) clock mode. 4 $t_{\text{SCYCLE}}$ delay until next command.
	1011 - CLRR. Clear Reset bit, all control registers retain their reset values. <sup>1)</sup> 4 $t_{\text{SCYCLE}}$ delay until next command.
	1111 - NOP. No serial operation.
	0011, 0101-1010, 1100-1110 - RSRV. Reserved encodings.
SDEV5 ... SDEV0	Serial device. Compared to SDEVID5...SDEVID0 field of INIT control register field to select the RDRAM to which the transaction is directed.
SBC	Serial broadcast. When set, RDRAMs ignore {SDEV5 ... SDEV0} for RDRAM selection.
SA11...SA0	Serial address. Selects which control register of the selected RDRAM is read or written.
SD15...SD0	Serial data. The 16 bits of data written to or read from the selected control register of the selected RDRAM.

<sup>1)</sup> The SETR and CLRR commands must always be applied in two successive transactions to RDRAMs; i.e. they may not be used in isolation. This is called "SETR/CLRR Reset".

## Initialization



**Figure 26 SIO Reset Sequence**

Initialization refers to the process that a controller must go through after power is applied to the system or the system is reset. The controller prepares the RDRAM sub-system for normal Channel operation by (primarily) using a sequence of control register transactions on the serial CMOS pins. The following steps outline the sequence seen by the various memory subsystem components (including the RDRAM components) during initialization. This sequence is available in the form of reference code. Contact Rambus Inc. for more information.

- 1. Start Clocks** – This step calculates the proper clock frequencies for PClk (controller logic), SynClk (RAC block), RefClk (DRCG component), CTM (RDRAM component), and SCK (SIO block).
- 2. RAC Initialization** – This step causes the INIT block to generate a sequence of pulses which resets the RAC, performs RAC maintenance operations, and measures timing intervals in order to ensure clock stability.
- 3. RDRAM Initialization** – This stage performs most of the steps needed to initialize the RDRAMs. The rest are performed in stages 5.0, 6.0, and 7.0. All of the steps in 3.0 are carried out through the SIO block interface.
  - 3.1/3.2. SIO Reset** – This reset operation is performed before any SIO control register read or write transactions. It clears six registers (TEST34, CCA, CCB, SKIP, TEST78, and TEST79) and places the INIT register into a special state (all bits cleared except SKP and SDEVID fields are set to ones).
  - 3.3. Write TEST77 Register** – The TEST77 register must be explicitly written with zeros before any other registers are read or written.
  - 3.4. Write TCYCLE Register** – The TCYCLE register is written with the cycle time  $t_{\text{CYCLE}}$  of the CTM clock (for Channel and RDRAMs) in units of 64ps. The  $t_{\text{CYCLE}}$  value is determined in stage 1.0.
  - 3.5. Write SDEVID Register** – The SDEVID (serial device identification) register of each RDRAM is written with a unique address value so that directed SIO read and write transactions can be performed. This address value increases from 0 to 31 according to the distance an RDRAM is from the ASIC component on the SIO bus (the closest RDRAM is address 0).

- 3.6. **Write DEVID Register** – The DEVID (device identification) register of each RDRAM is written with a unique address value so that directed memory read and write transactions can be performed. This address value increases from 0 to 31. The DEVID value is not necessarily the same as the SDEVID value. RDRAMs are sorted into regions of the same core configuration (number of bank, row, and column address bits and core type).
- 3.7. **Write PDNX,PDNXA Registers** – The PDNX and PDNXA registers are written with values that are used to measure the timing intervals connected with an exit from the PDN (powerdown) power state.
- 3.8. **Write NAPX Register** – The NAPX register is written with values that are used to measure the timing intervals connected with an exit from the NAP power state.
- 3.9. **Write TPARM Register** – The TPARM register is written with values which determine the time interval between a COL packet with a memory read command and the Q packet with the read data on the Channel. The values written set each RDRAM to the minimum value permitted for the system. This will be adjusted later in stage 6.0.
- 3.10. **Write TCDLY1 Register** – The TCDLY1 register is written with values which determine the time interval between a COL packet with a memory read command and the Q packet with the read data on the Channel. The values written set each RDRAM to the minimum value permitted for the system. This will be adjusted later in stage 6.0.
- 3.11. **Write TFRM Register** – The TFRM register is written with a value that is related to the  $t_{RCD}$  parameter for the system. The  $t_{RCD}$  parameter is the time interval between a ROW packet with an activate command and the COL packet with a read or write command.
- 3.12. **3.12 SETR/CLRR** – Each RDRAM is given a SETR command and a CLRR command through the SIO block. This sequence performs a second reset operation on the RDRAMs.
- 3.13. **Write CCA and CCB Registers** - These registers are written with a value halfway between their minimum and maximum values. This shortens the time needed for the RDRAMs to reach their steady-state current control values in stage 5.0.
- 3.14. **Powerdown Exit** – The RDRAMs are in the PDN power state at this point. A broadcast PDNExit command is performed by the SIO block to place the RDRAMs in the RLX (relax) power state in which they are ready to receive ROW packets.
- 3.15. **SETF** - Each RDRAM is given a SETF command through the SIO block. One of the operations performed by this step is to generate a value for the AS (autoskip) bit in the SKIP register and fix the RDRAM to a particular read domain.
4. **Controller Configuration** – This stage initializes the controller block. Each step of this stage will set a field of the ConfigRMC[63:0] bus to the appropriate value. Other controller implementations will have similar initialization requirements, and this stage may be used as a guide.
  - 4.1. **Initial Read Data Offset** – The ConfigRMC bus is written with a value which determines the time interval between a COL packet with a memory read command and the Q packet with the read data on the Channel. The value written sets RMC.d1 to the minimum value permitted for the system. This will be adjusted later in stage 6.0.
  - 4.2. **Configure Row/Column Timing** – This step determines the values of the  $t_{RAS,MIN}$ ,  $t_{RP,MIN}$ ,  $t_{RC,MIN}$ ,  $t_{RCD,MIN}$ ,  $t_{RR,MIN}$ , and  $t_{PP,MIN}$  RDRAM timing parameters that are present in the system. The ConfigRMC bus is written with values that will be compatible with all RDRAM devices that are present.
  - 4.3. **Set Refresh Interval** – This step determines the values of the  $t_{REF,MAX}$  RDRAM timing parameter that are present in the system. The ConfigRMC bus is written with a value that will be compatible with all RDRAM devices that are present.

- 4.4. **Set Current Control Interval** – This step determines the values of the  $t_{\text{CTRL,MAX}}$  RDRAM timing parameter that are present in the system. The ConfigRMC bus is written with a value that will be compatible with all RDRAM devices that are present.
- 4.5. **Set Slew Rate Control Interval** – This step determines the values of the  $t_{\text{TEMP,MAX}}$  RDRAM timing parameter that are present in the system. The ConfigRMC bus is written with a value that will be compatible with all RDRAM devices that are present.
- 4.6. **Set Bank/Row/Col Address Bits** – This step determines the number of RDRAM bank, row, and column address bits that are present in the system. It also determines the RDRAM core types (independent, doubled, or split) that are present. The ConfigRMC bus is written with a value that will be compatible with all RDRAM devices that are present.
5. **RDRAM Current Control** – This step causes the INIT block to generate a sequence of pulses which performs RDRAM maintenance operations.
6. **RDRAM Core, Read Domain Initialization** – This stage completes the RDRAM initialization
  - 6.1. **RDRAM Core Initialization** – A sequence of 192 memory refresh transactions is performed in order to place the cores of all RDRAMs into the proper operating state.
  - 6.2. **RDRAM Read Domain Initialization** - A memory write and memory read transaction is performed to each RDRAM to determine which read domain each RDRAM occupies. The programmed delay of each RDRAM is then adjusted so the total RDRAM read delay (propagation delay plus programmed delay) is constant. The TPARM and TCDLY1 registers of each RDRAM are rewritten with the appropriate read delay values. The ConfigRMC bus is also rewritten with an updated value.
7. **Other RDRAM Register Fields** – This stage rewrites the INIT register with the final values of the LSR, NSR, and PSR fields.

In essence, the controller must read all the read-only configuration registers of all RDRAMs (or it must read the SPD device present on each RIMM), it must process this information, and then it must write all the read-write registers to place the RDRAMs into the proper operating mode.

Initialization Note [1]:

During the initialization process, it is necessary for the controller to perform 128 current control operations (3xCAL, 1xCAL/SAM) and one temperature calibrate operation (TCEN/TCAL) after reset or after powerdown (PDN) exit.

Initialization Note [2]:

There are two classes of 64/72Mbit RDRAM. They are distinguished by the “S28IECO” bit in the SPD. The behavior of the RDRAM at initialization is slightly different for the two types:

S28IECO = 0: Upon powerup the device enters ATTN state. The serial operations SETR, CLRR, and SETF are performed without requiring a SDEVID match of the SBC bit (broadcast) to be set.

S28IECO = 1: Upon powerup the device enters PDN state. The serial operations SETR, CLRR, and SETF require a SDEVID match.

See the document detailing the reference initialization procedure for more information on how to handle this in a system.

Initialization Note [3]:

After the step of equalizing the total read delay of each RDRAM has been completed (i.e. after the TCDLY0 and TCDLY1 fields have been written for the final time), a single final memory read

transaction should be made to each RDRAM in order to ensure that the output pipeline stages have been cleared.

Initialization Note [4]:

The SETF command (in the serial SRQ packet) should only be issued once during the Initialization process, as should the SETR and CLRR commands.

Initialization Note [5]:

The CLRR command (in the serial SRQ packet) leaves some of the contents of the memory core in an indeterminate state.

### Control Register Summary

**Table 17** summarizes the RDRAM control registers. Detail is provided for each control register in **Figure 27** through **Figure 43**. Read-only bits which are shaded gray are unused and return zero. Read-write bits which are shaded gray are reserved and should always be written with zero. The RIMM SPD Application Note (DL-0054) describes additional read-only configuration registers which are present on Direct RIMMs.

The state of the register fields are potentially affected by the IO Reset operation or the SETR/CLRR operation. This is indicated in the text accompanying each register diagram.

**Table 17 Control Register Summary**

SA11...SA0	Register	Field	read-write/ read-only	Description
021 <sub>16</sub>	INIT	SDEVID	read-write, 6 bits	Serial device ID. Device address for control register read/write.
		PSX	read-write, 1 bit	Power select exit. PDN/NAP exit with device addr on DQA5 ... 0.
		SRP	read-write, 1 bit	SIO repeater. Used to initialize RDRAM.
		NSR	read-write, 1 bit	NAP self-refresh. Enables self-refresh in NAP mode.
		PSR	read-write, 1 bit	PDN self-refresh. Enables self-refresh in PDN mode.
		LSR	read-write, 1 bit	Low power self-refresh. Enables low power self-refresh.
		TEN	read-write, 1 bit	Temperature sensing enable.
		TSQ	read-write, 1 bit	Temperature sensing output.
		DIS	read-write, 1 bit	RDRAM disable.
022 <sub>16</sub>	TEST34	TEST34	read-write, 16 bits	Test register. Do not read or write after SIO reset.
023 <sub>16</sub>	CNFGA	REFBIT	read-only, 3 bit	Refresh bank bits. Used for multi-bank refresh.
		DBL	read-only, 1 bit	Double. Specifies doubled-bank architecture
		MVER	read-only, 6 bit	Manufacturer version. Manufacturer identification number.
		PVER	read-only, 6 bit	Protocol version. Specifies version of Direct protocol supported.

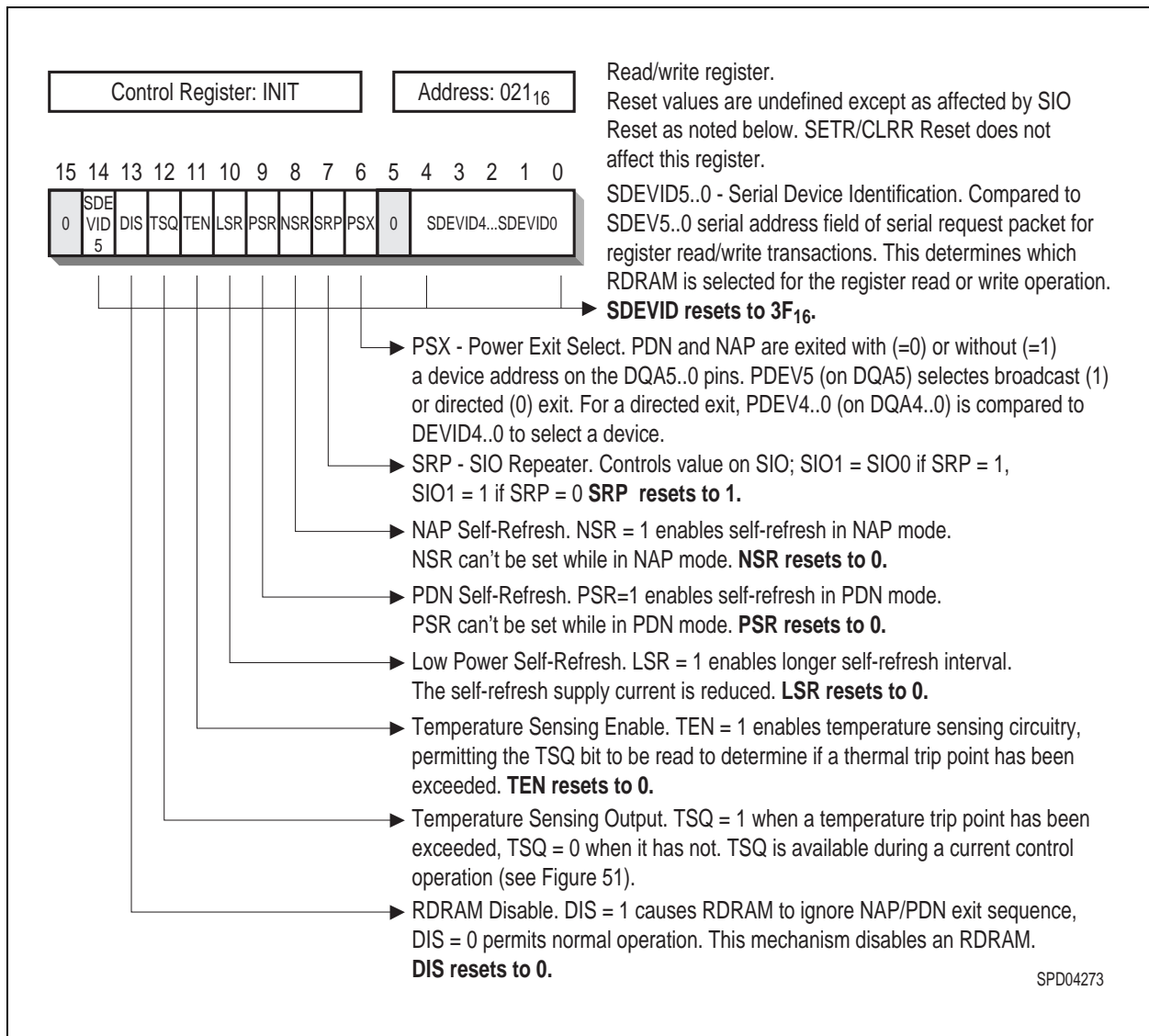
**Table 17 Control Register Summary (cont'd)**

SA11...SA0	Register	Field	read-write/ read-only	Description
024 <sub>16</sub>	CNFGB	BYT	read-only, 1 bit	Byte. Specifies an 8-bit or 9-bit byte size.
		DEVTYP	read-only, 3 bit	Device type. Device can be RDRAM or some other device category.
		SPT	read-only, 1 bit	Split-core. Each core half is an individual dependent core.
		CORG	read-only, 6 bit	Core organization. Bank, row, column address field sizes.
		SVER	read-only, 6 bit	Stepping version. Mask version number.
040 <sub>16</sub>	DEVID	DEVID	read-write, 5 bits	Device ID. Device address for memory read/write.
041 <sub>16</sub>	REFB	REFB	read-write, 4 bits	Refresh bank. Next bank to be refreshed by self-refresh.
042 <sub>16</sub>	REFR	REFR	read-write, 9 bits	Refresh row. Next row to be refreshed by REFA, self-refresh.
043 <sub>16</sub>	CCA	CCA	read-write, 7 bits	Current control A. Controls $I_{OL}$ output current for DQA.
		ASYMA	read-write, 2 bits	Asymmetry control. Controls asymmetry of $V_{OL}/V_{OH}$ swing for DQA.
044 <sub>16</sub>	CCB	CCB	read-write, 7 bits	Current control B. Controls $I_{OL}$ output current for DQB.
		ASYMB	read-write, 2 bits	Asymmetry control. Controls asymmetry of $V_{OL}/V_{OH}$ swing for DQB.
045 <sub>16</sub>	NAPX	NAPXA	read-write, 5 bits	NAP exit. Specifies length of NAP exit phase A.
		NAPX	read-write, 5 bits	NAP exit. Specifies length of NAP exit phase A + phase B.
		DQS	read-write, 1 bits	DQ select. Selects CMD framing for NAP/PDN exit.
046 <sub>16</sub>	PDNXA	PDNXA	read-write, 13 bits	PDN exit. Specifies length of PDN exit phase A.
047 <sub>16</sub>	PDNX	PDNX	read-write, 13 bits	PDN exit. Specifies length of PDN exit phase A + phase B.
048 <sub>16</sub>	TPARM	TCAS	read-write, 2 bits	$t_{CAS-C}$ core parameter. Determines $t_{OFFP}$ data sheet parameter.
		TCLS	read-write, 2 bits	$t_{CLS-C}$ core parameter. Determines $t_{CAC}$ and $t_{OFFP}$ parameters.
		TCDLY0	read-write, 3 bits	$t_{CDLY0-C}$ core parameter. Programmable delay for read data.
049 <sub>16</sub>	TFRM	TFRM	read-write, 4 bits	$t_{FRM-C}$ core parameter. Determines ROW-COL packet framing interval.
04a <sub>16</sub>	TCDLY1	TCDLY1	read-write, 3 bits	$t_{CDLY1-C}$ core parameter. Programmable delay for read data.
04c <sub>16</sub>	TCYCLE	TCYCLE	read-write, 14 bits	$t_{CYCLE}$ data sheet parameter. Specifies cycle time in 64 ps units.

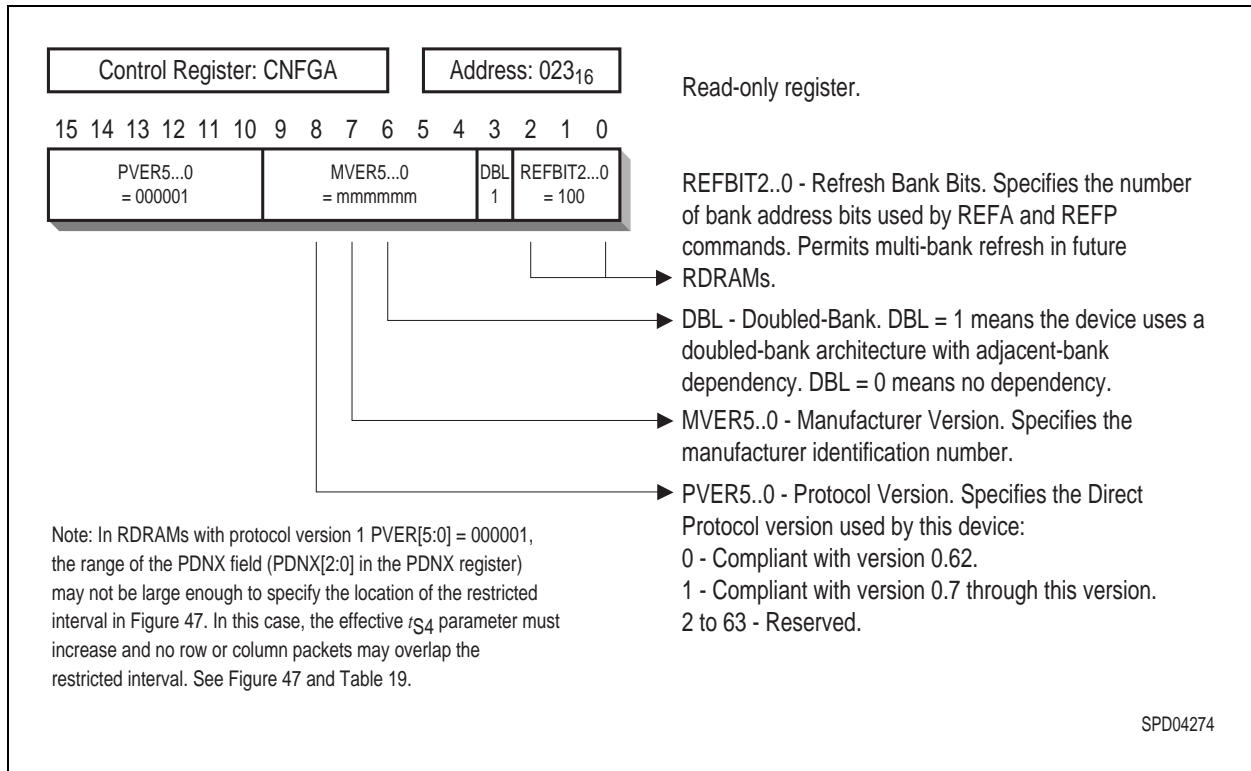
**Table 17 Control Register Summary (cont'd)**

SA11...SA0	Register	Field	read-write/ read-only	Description
04b <sub>16</sub>	SKIP	AS	read-only, 1 bit	Autoskip value established by the SETF command.
		MSE	read-write, 1 bit	Manual skip enable. Allows the MS value to override the AS value.
		MS	read-write, 1 bit	Manual skip value.
04d <sub>16-</sub>	TEST77	TEST77	read-write, 16 bits	Test register. Write with zero after SIO reset.
04e <sub>16-</sub>	TEST78	TEST78	read-write, 16 bits	Test register. Do not read or write after SIO reset.
04f <sub>16-</sub>	TEST79	TEST79	read-write, 16 bits	Test register. Do not read or write after SIO reset.
080 <sub>16</sub> - Off <sub>16</sub>	reserved	reserved	vendor-specific	Vendor-specific test registers. Do not read or write after SIO reset.

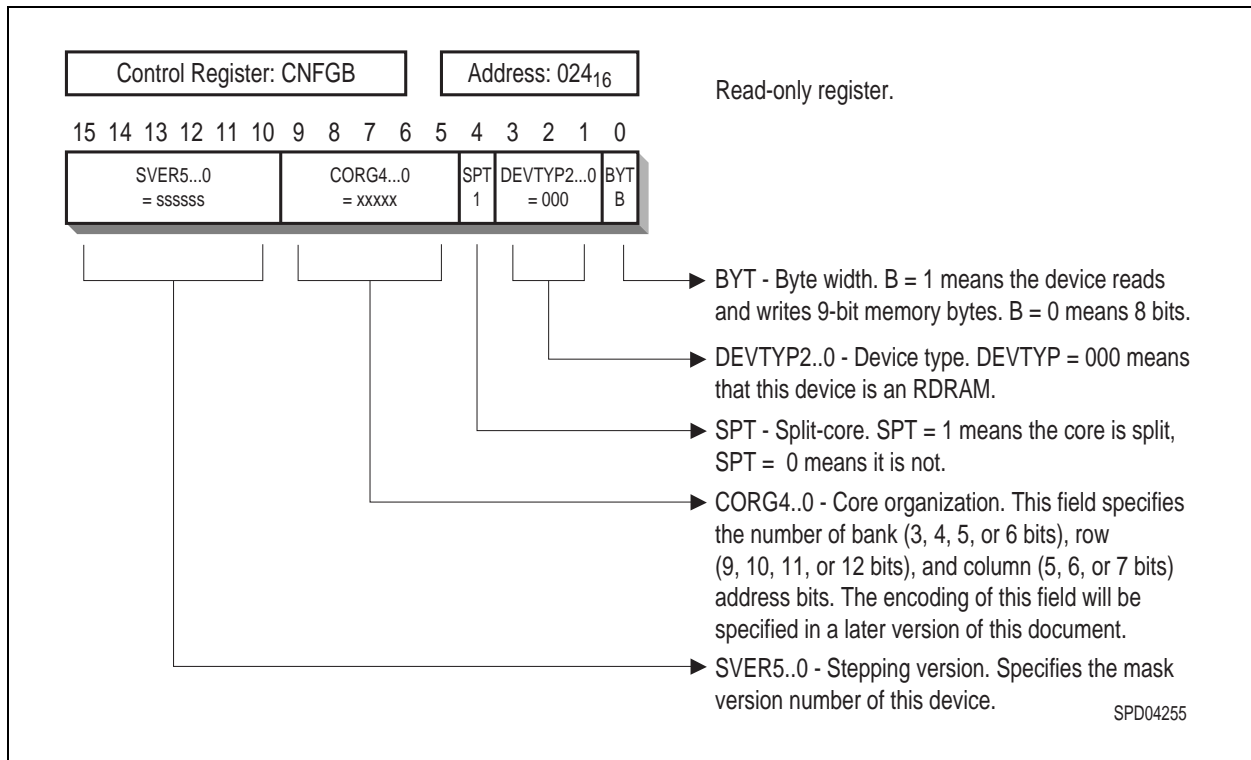




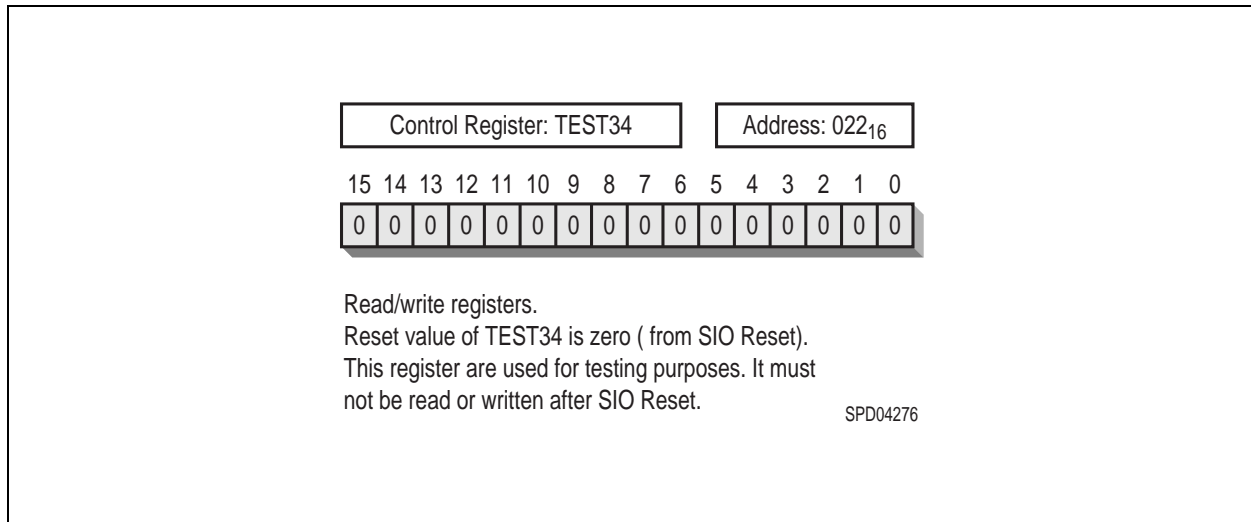
**Figure 27** INIT Register



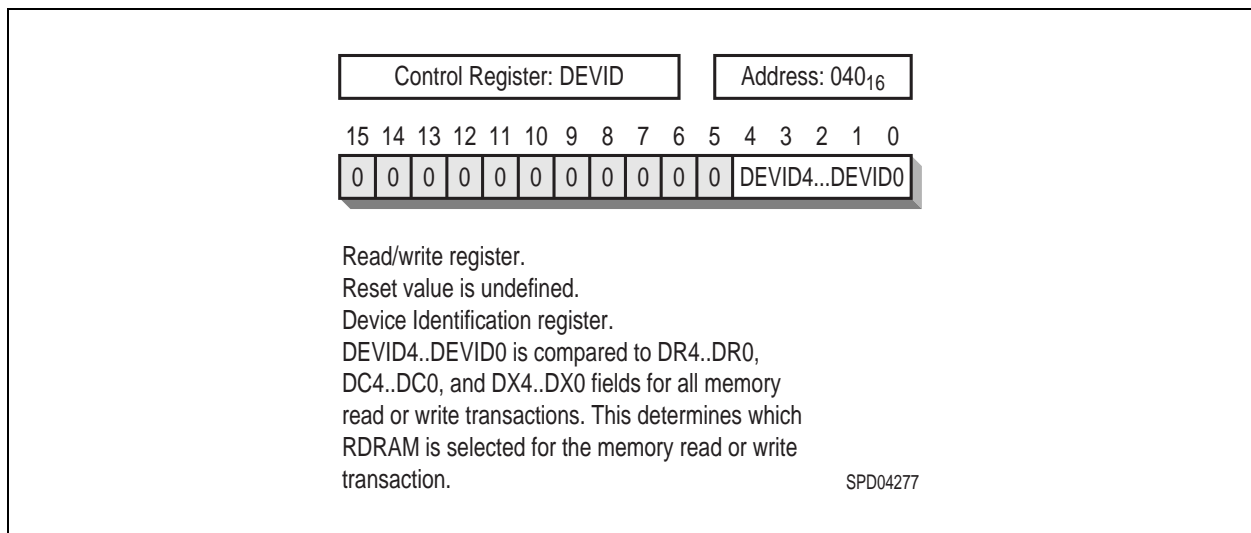
**Figure 28 CNFGA Register**



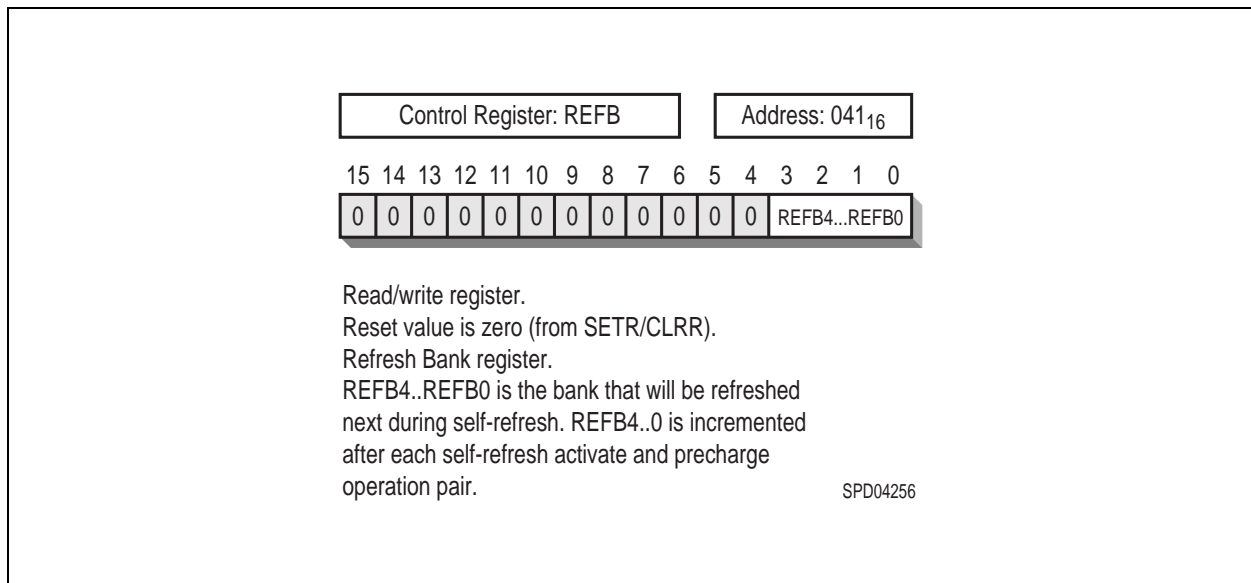
**Figure 29 CNFGB Register**



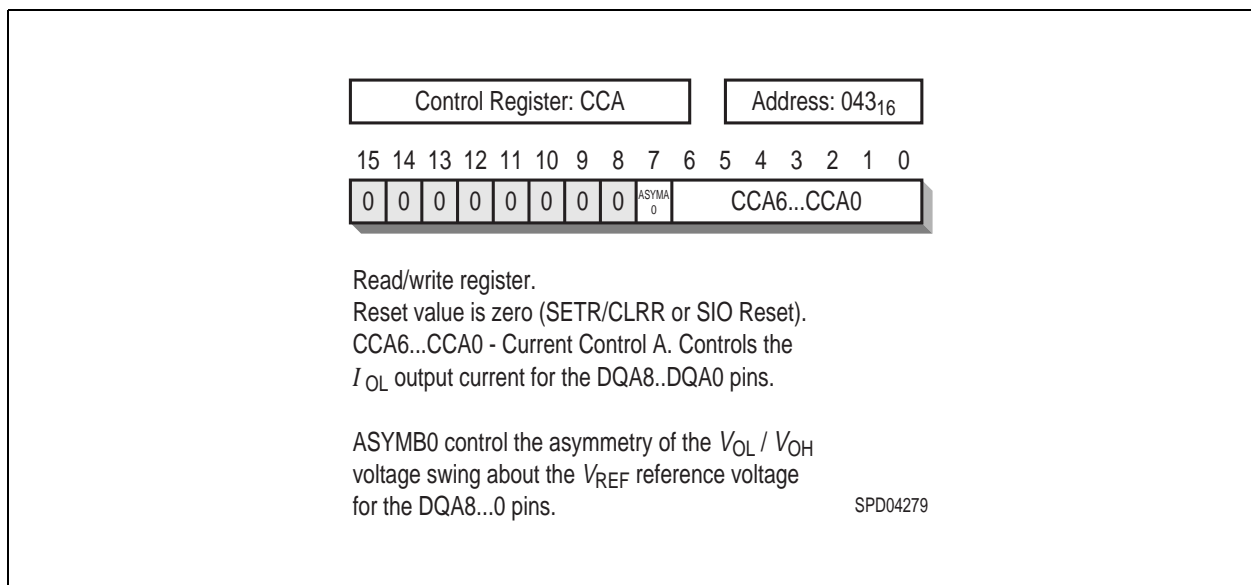
**Figure 30 TEST Register**



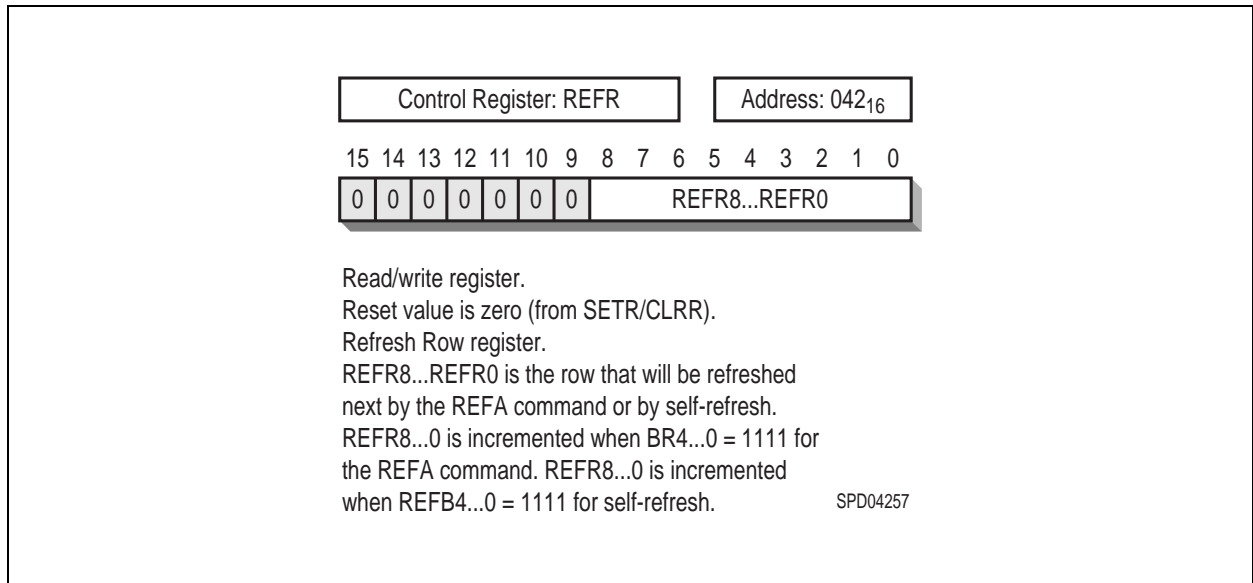
**Figure 31 DEVID Register**



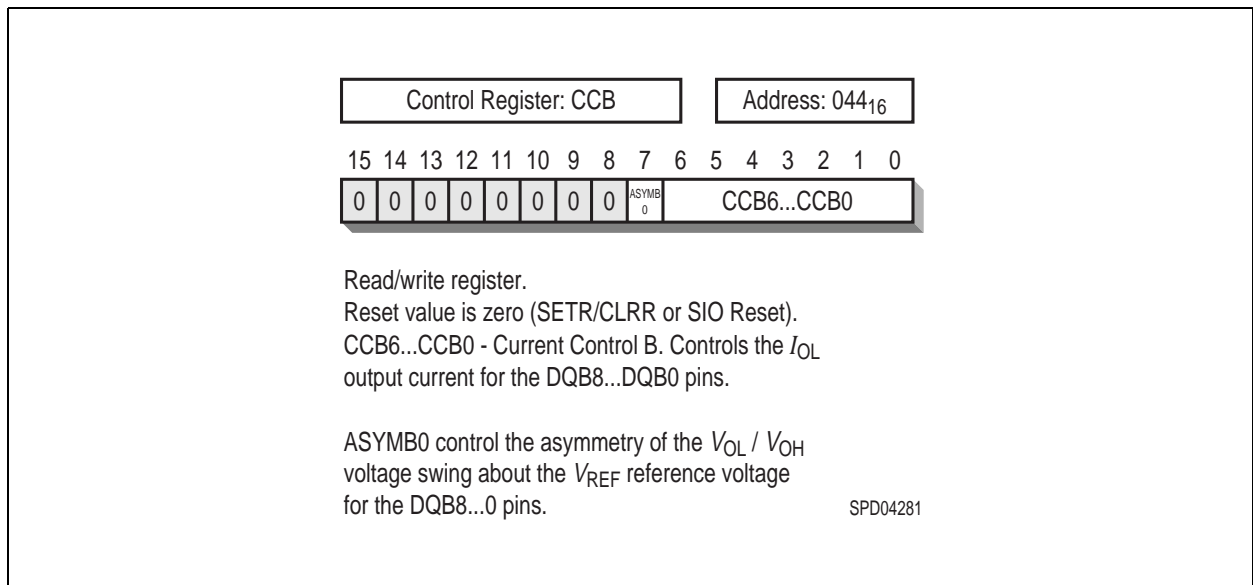
**Figure 32 REFB Register**



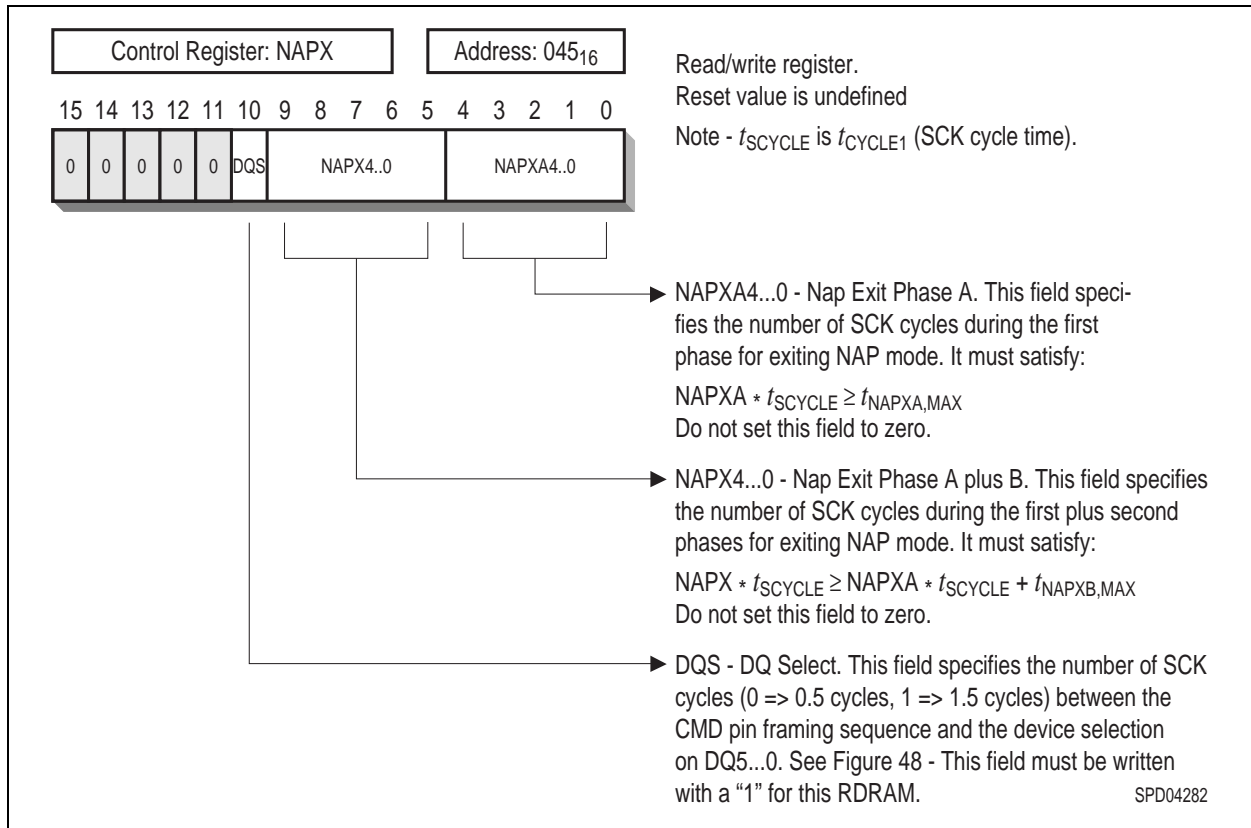
**Figure 33 CCA Register**



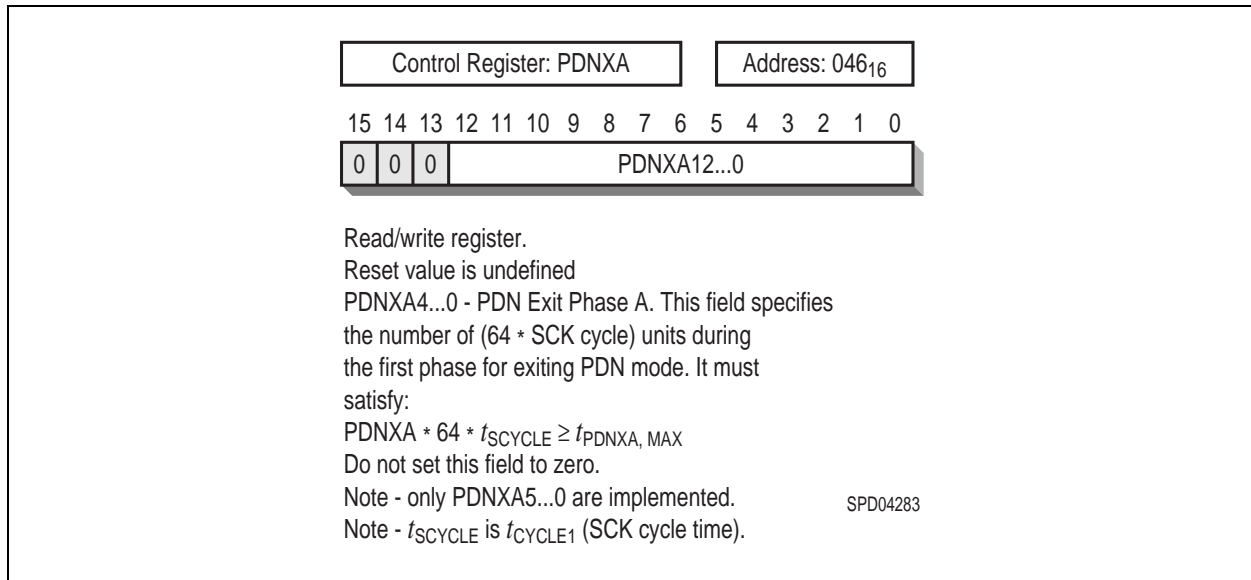
**Figure 34 REFR Register**



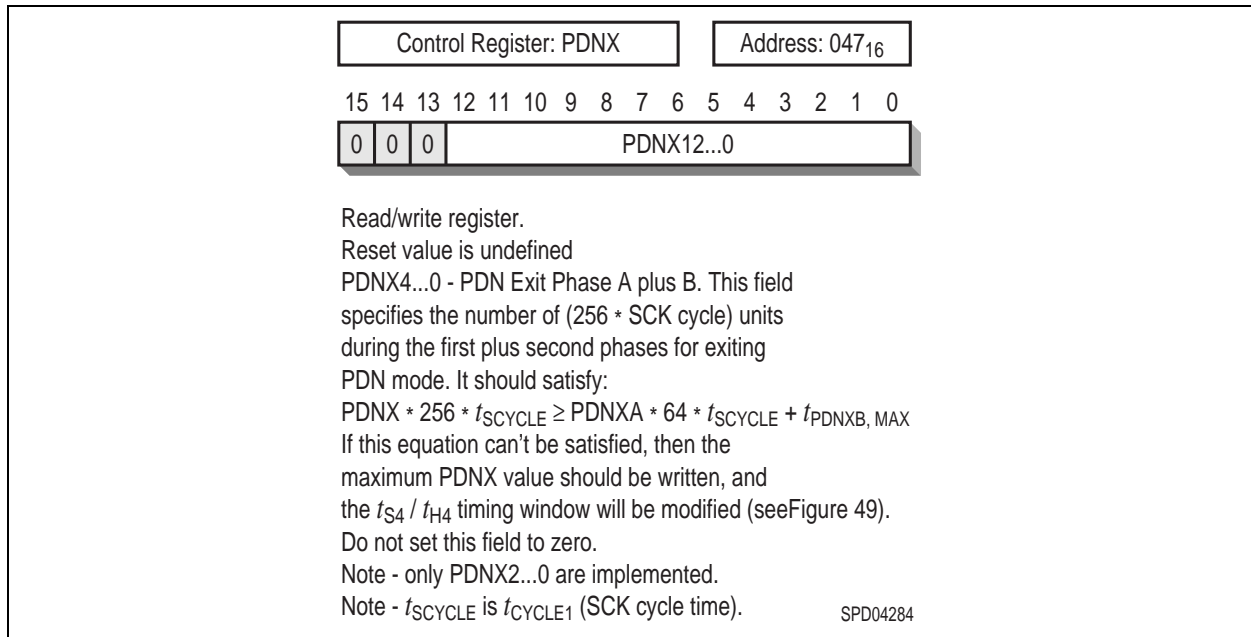
**Figure 35 CCB Register**



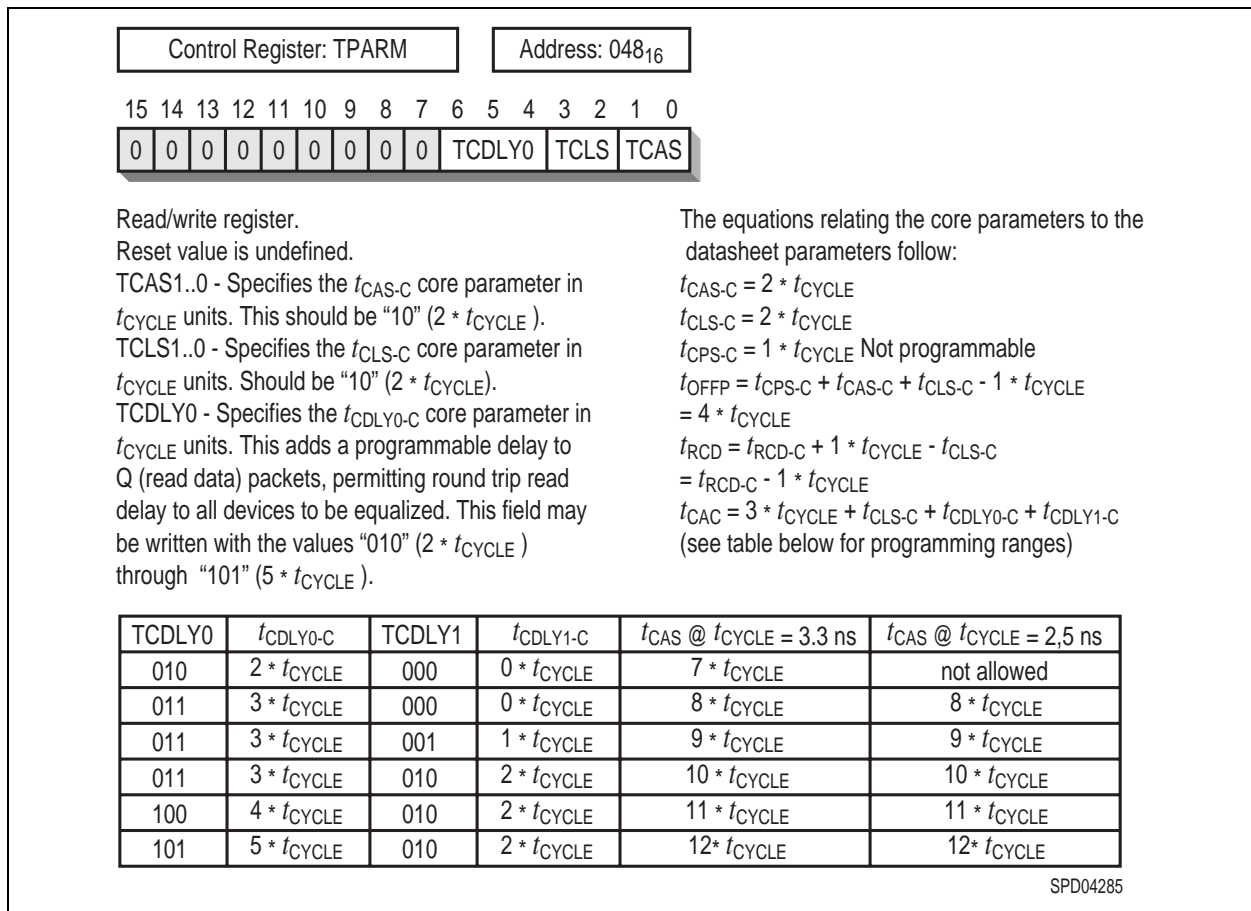
**Figure 36 NAPX Register**



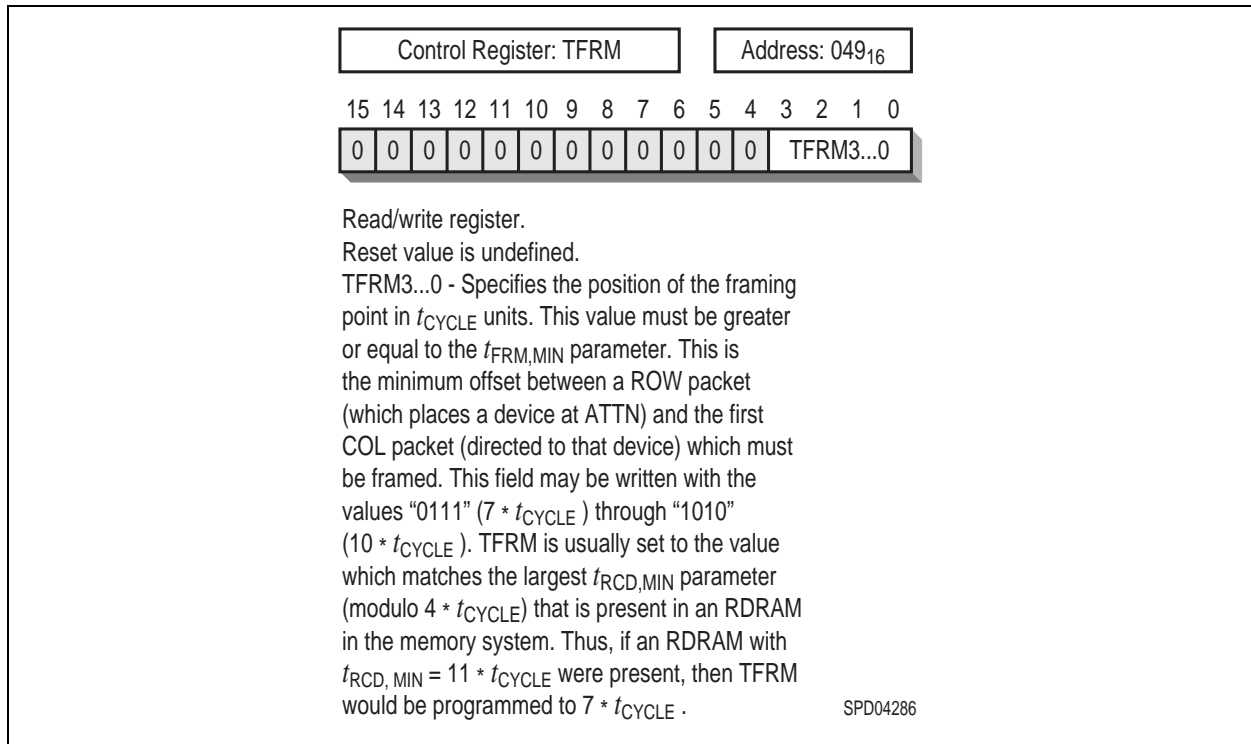
**Figure 37 PDNXA Register**



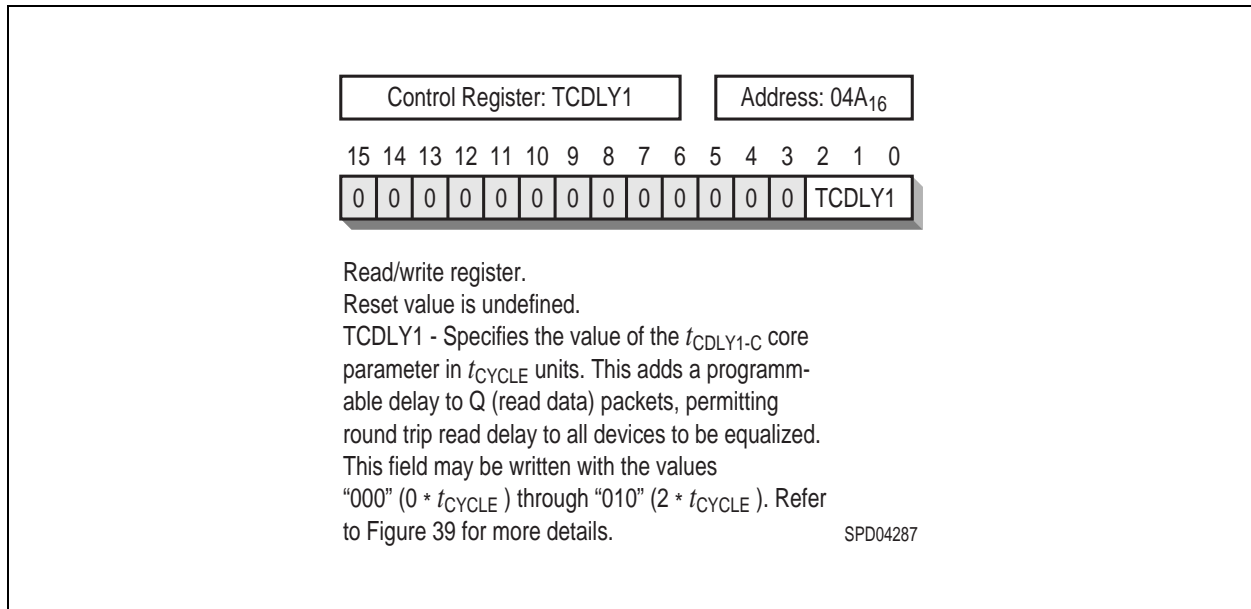
**Figure 38 PDNX Register**



**Figure 39 TPARM Register**

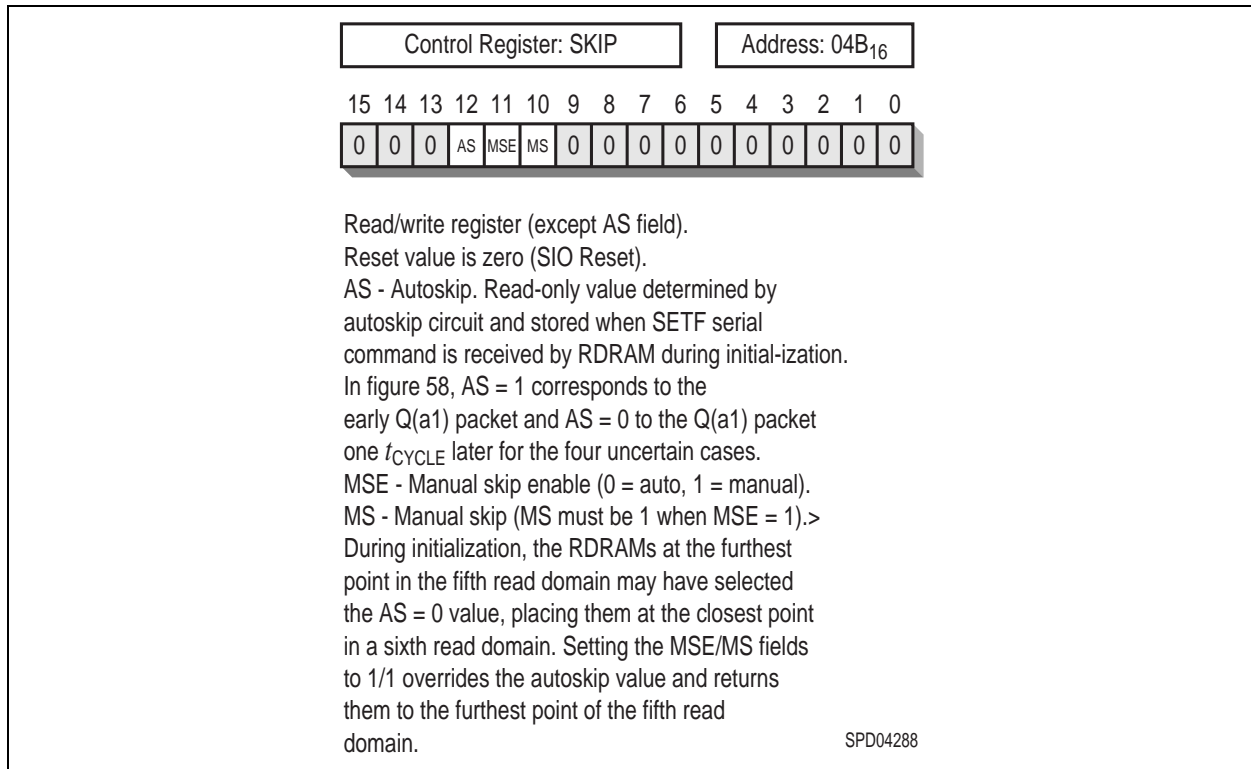


**Figure 40 TFRM Register**

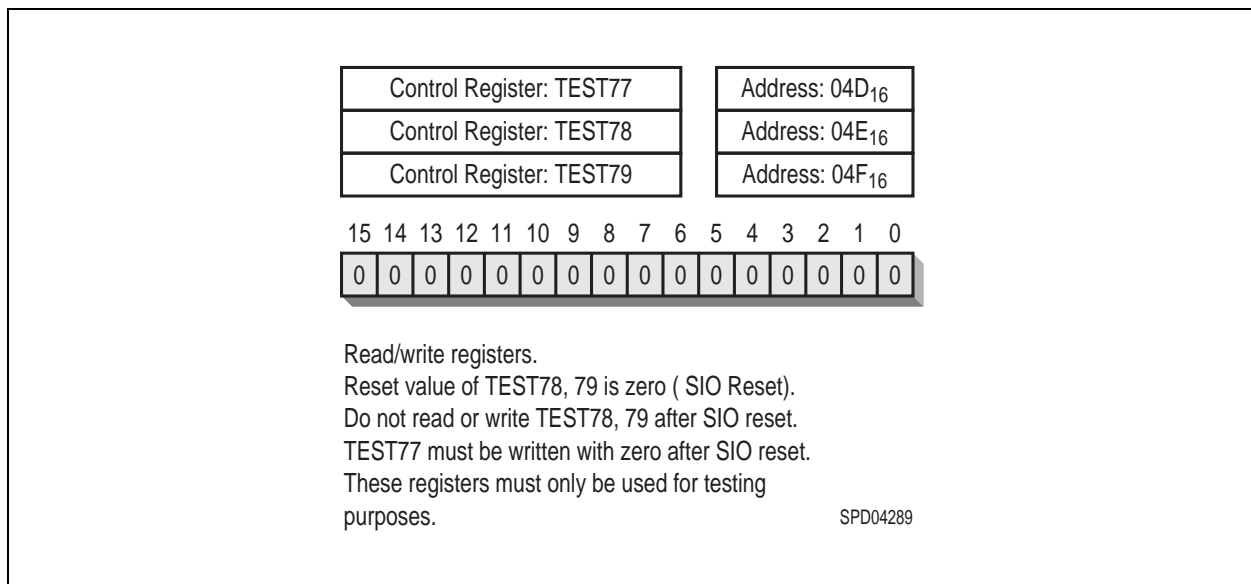


**Figure 41 TRDLY Register**

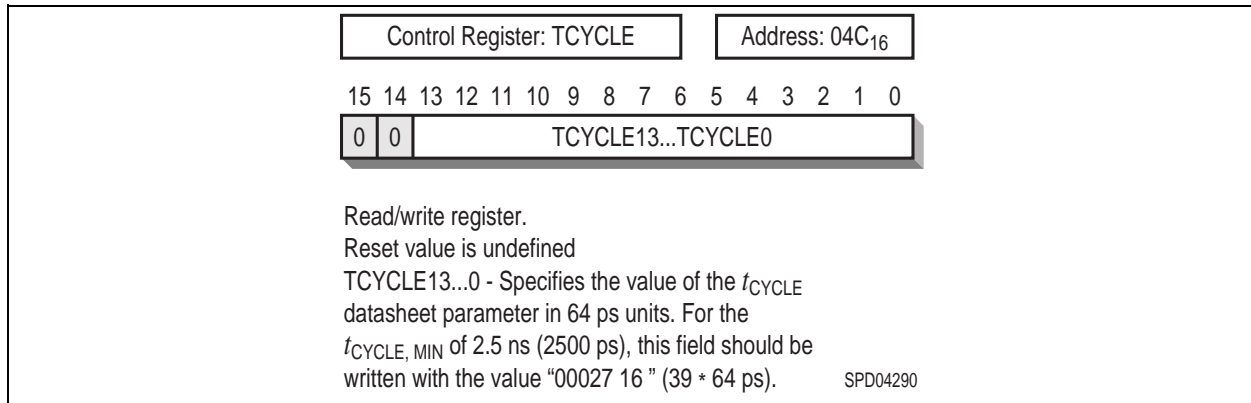




**Figure 42 SKIP Register**



**Figure 43 TEST Registers**



**Figure 44 TCYCLE Register**

### Power State Management

**Table 18** summarizes the power states available to a Direct RDRAM. In general, the lowest power states have the longest operational latencies. For example, the relative power levels of PDN state and STBY state have a ratio of about 1:110, and the relative access latencies to get read data have a ratio of about 250:1.

PDN state is the lowest power state available. The information in the RDRAM core is usually maintained with self-refresh; an internal timer automatically refreshes all rows of all banks. PDN has a relatively long exit latency because the TCLK/RCLK block must resynchronize itself to the external clock signal.

NAP state is another low-power state in which either self-refresh or REFA-refresh are used to maintain the core. See "Refresh" on page 64 for a description of the two refresh mechanisms. NAP has a shorter exit latency than PDN because the TCLK/RCLK block maintains its synchronization state relative to the external clock signal at the time of NAP entry. This imposes a limit ( $t_{\text{NLIMIT}}$ ) on how long an RDRAM may remain in NAP state before briefly returning to STBY or ATTN to update this synchronization state.

**Table 18 Power State Summary**

Power State	Description	Blocks Consuming Power	Power State	Description	Blocks Consuming Power
PDN	Powerdown state.	Self-refresh	NAP	Nap state. Similar to PDN except lower wake-up latency.	Self-refresh or REFA-refresh TCLK/RCLK-Nap
STBY	Standby state. Ready for ROW packets.	REFA-refresh TCLK/RCLK ROW demux receiver	ATTN	Attention state. Ready for ROW and COL packets.	REFA-refresh TCLK/RCLK ROW demux receiver COL demux receiver
ATTNR	Attention read state. Ready for ROW and COL packets. Sending Q (read data) packets.	REFA-refresh TCLK/RCLK ROW demux receiver COL demux receiver DQ mux transmitter Core power	ATTNW	Attention write state. Ready for ROW and COL packets. Ready for D (write data) packets.	REFA-refresh TCLK/RCLK ROW demux receiver COL demux receiver DQ demux receiver Core power

**Figure 45** summarizes the transition conditions needed for moving between the various power states. Note that NAP and PDN have been divided into two substates (NAP-A/NAP-S and PDN-A/PDN-S) to account for the fact that a NAP or PDN exit may be made to either ATTN or STBY states.

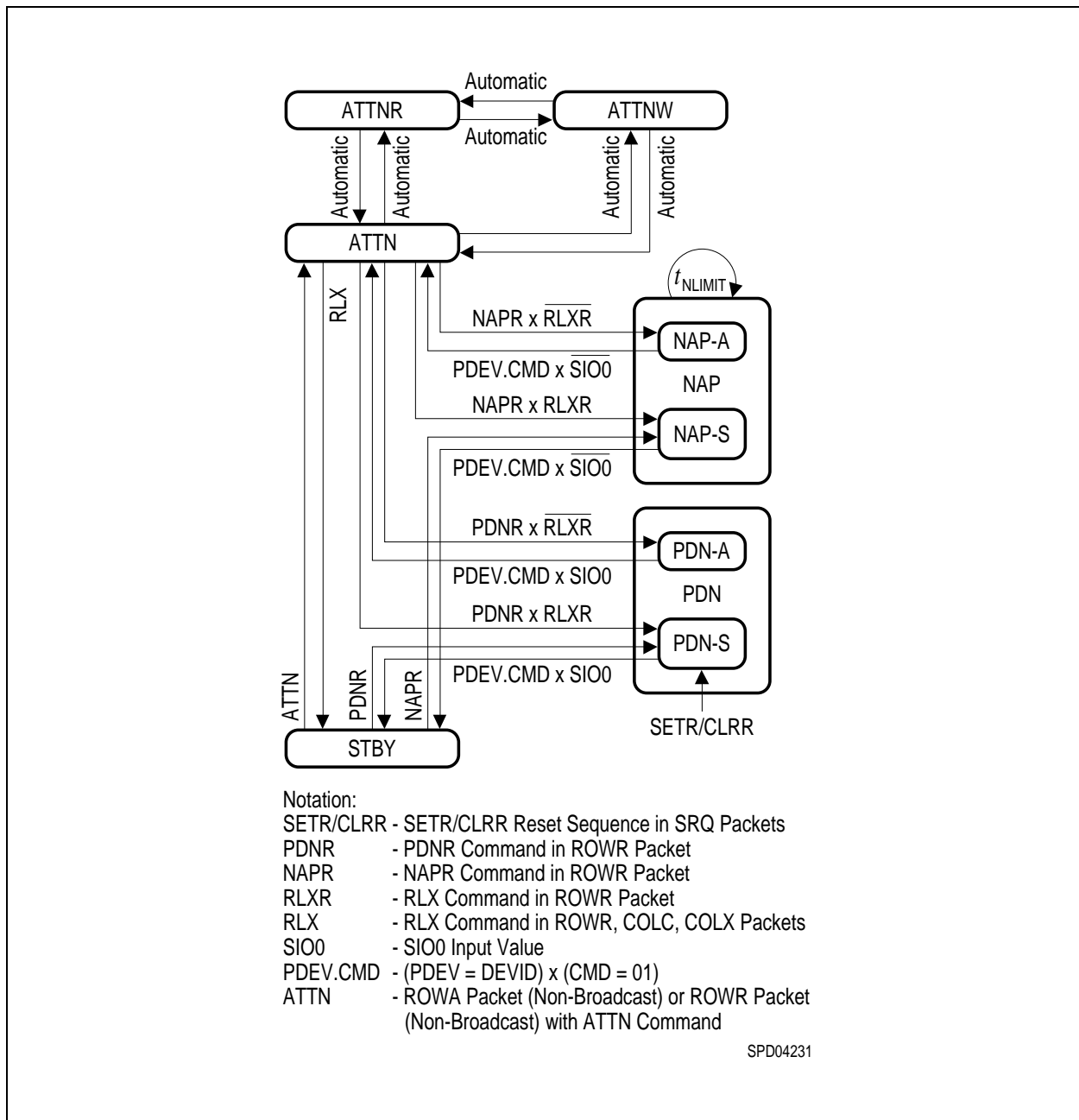
At initialization, the SETR/CLRR Reset sequence will put the RDRAM into PDN-S state. The PDN exit sequence involves an optional PDEV specification and bits on the CMD and SIO<sub>IN</sub> pins.

Once the RDRAM is in STBY, it will move to the ATTN/ATTNR/ATTNW states when it receives a non-broadcast ROWA packet or non-broadcast ROWR packet with the ATTN command. The RDRAM returns to STBY from these three states when it receives a RLX command. Alternatively, it may enter NAP or PDN state from ATTN or STBY states with a NAPR or PDNR command in an ROWR packet. The PDN or NAP exit sequence involves an optional PDEV specification and bits on the CMD and SIO<sub>0</sub> pins. The RDRAM returns to the ATTN or STBY state it was originally in when it first entered NAP or PDN.

An RDRAM may only remain in NAP state for a time  $t_{NLIMIT}$ . It must periodically return to ATTN or STBY.

The NAPRC command causes a napdown operation if the RDRAM's NCBIT is set. The NCBIT is not directly visible. It is undefined on reset. It is set by a NAPR command to the RDRAM, and it is cleared by an ACT command to the RDRAM. It permits a controller to manage a set of RDRAMs in a mixture of power states.

STBY state is the normal idle state of the RDRAM. In this state all banks and sense amps have usually been left precharged and ROWA and ROWR packets on the ROW pins are being monitored. When a non-broadcast ROWA packet or non-broadcast ROWR packet (with the ATTN command) packet addressed to the RDRAM is seen, the RDRAM enters ATTN state (see the right side of **Figure 46**). This requires a time  $t_{SA}$  during which the RDRAM activates the specified row of the specified bank. A time  $TFRM \times t_{CYCLE}$  after the ROW packet, the RDRAM will be able to frame COL packets (TFRM is a control register field - see **Figure 40**). Once in ATTN state, the RDRAM will automatically transition to the ATTNW and ATTNR states as it receives WR and RD commands.



**Figure 45 Power State Transition Diagram**

Once the RDRAM is in ATTN, ATTNW, or ATTNR states, it will remain there until it is explicitly returned to the STBY state with a RLX command. A RLX command may be given in an ROWR, COLC, or COLX packet (see the left side of **Figure 46**). It is usually given after all banks of the RDRAM have been precharged; if other banks are still activated, then the RLX command would probably not be given.

If a broadcast ROWA packet or ROWR packet (with the ATTN command) is received, the RDRAM's power state doesn't change. If a broadcast ROWR packet with RLXR command is received, the RDRAM goes to STBY.

**Figure 47** shows the NAP entry sequence (left). NAP state is entered by sending a NAPR command in a ROW packet. A time  $t_{ASN}$  is required to enter NAP state (this specification is provided for power calculation purposes). The clock on CTM/CFM must remain stable for a time  $t_{CD}$  after the NAPR command.

The RDRAM may be in ATTN or STBY state when the NAPR command is issued. When NAP state is exited, the RDRAM will return to the original starting state (ATTN or STBY). If it is in ATTN state and a RLXR command is specified with NAPR, then the RDRAM will return to STBY state when NAP is exited.

**Figure 47** also shows the PDN entry sequence (right). PDN state is entered by sending a PDNR command in a ROW packet. A time  $t_{ASP}$  is required to enter PDN state (this specification is provided for power calculation purposes). The clock on CTM/CFM must remain stable for a time  $t_{CD}$  after the PDNR command.

The RDRAM may be in ATTN or STBY state when the PDNR command is issued. When PDN state is exited, the RDRAM will return to the original starting state (ATTN or STBY). If it is in ATTN state and a RLXR command is specified with PDNR, then the RDRAM will return to STBY state when PDN is exited. The current- and slew-rate-control levels are re-established.

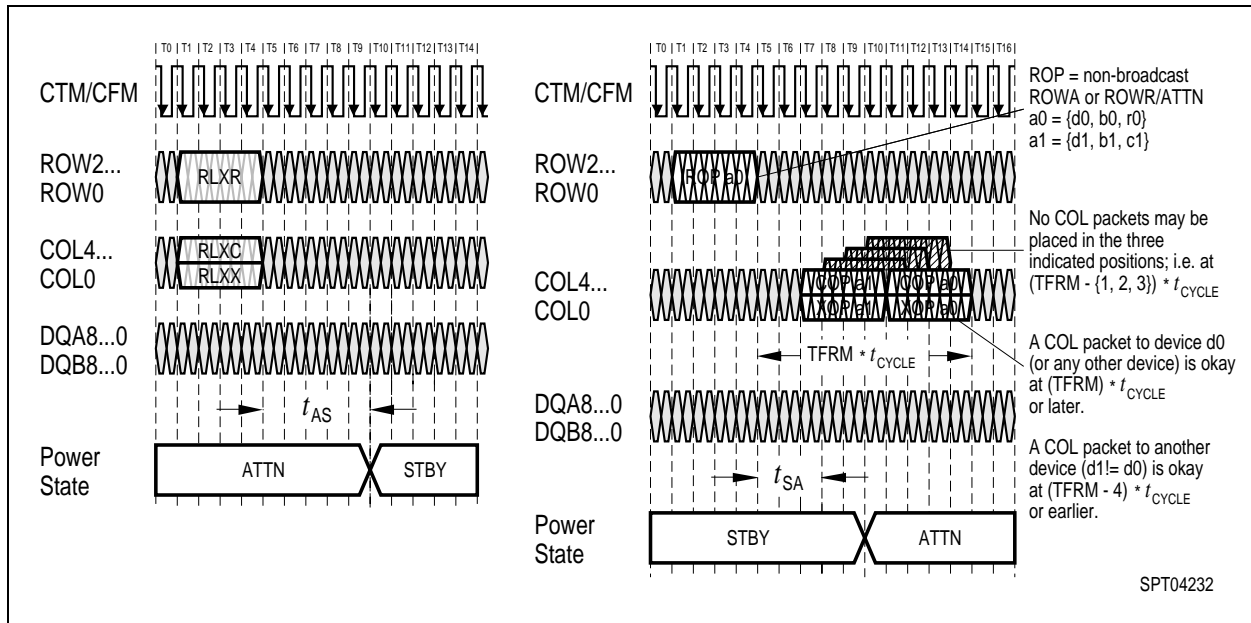
The RDRAM's write buffer must be retired with the appropriate COP command before NAP or PDN are entered. Also, all the RDRAM's banks must be precharged before NAP or PDN are entered. The exception to this is if NAP is entered with the NSR bit of the INIT register cleared (disabling self-refresh in NAP). The commands for relaxing, retiring, and precharging may be given to the RDRAM as late as the ROPa0, COPa0, and XOPa0 packets in **Figure 47**. No broadcast packets nor packets directed to the RDRAM entering Nap or PDN may overlay the quiet window. This window extends for a time  $t_{NPQ}$  after the packet with the NAPR or PDNR command.

**Figure 48** shows the NAP and PDN exit sequences. These sequences are virtually identical; the minor differences will be highlighted in the following description.

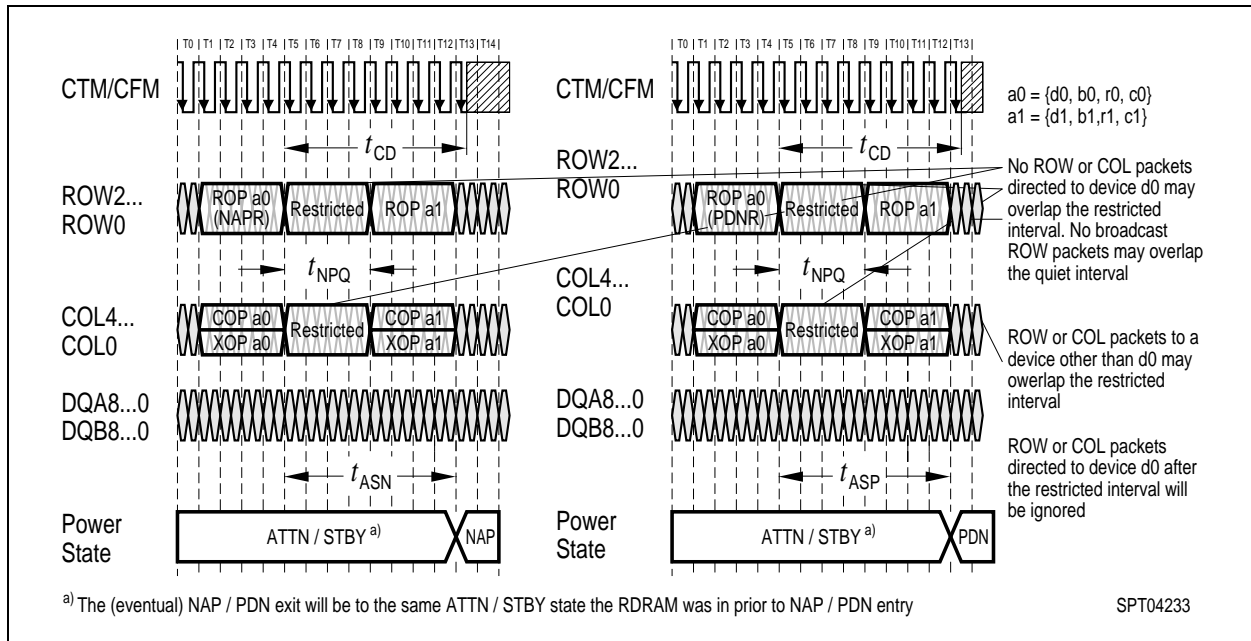
Before NAP or PDN exit, the CTM/CFM clock must be stable for a time  $t_{CE}$ . Then, on a falling and rising edge of SCK, if there is a "01" on the CMD input, NAP or PDN state will be exited. Also, on the falling SCK edge the SIO0 input must be at a 0 for NAP exit and 1 for PDN exit.

If the PSX bit of the INIT register is 0, then a device PDEV5 ... 0 is specified for NAP or PDN exit on the DQA5 ... 0 pins. This value is driven on the rising SCK edge 0.5 or 1.5 SCK cycles after the original falling edge, depending upon the value of the DQS bit of the NAPX register. If the PSX bit of the INIT register is 1, then the RDRAM ignores the PDEV5 ... 0 address packet and exits NAP or PDN when the wake-up sequence is presented on the CMD wire. The ROW and COL pins must be quiet at a time  $t_{S4}/t_{H4}$  around the indicated falling SCK edge (timed with the PDNX or NAPX register fields). After that, ROW and COL packets may be directed to the RDRAM which is now in ATTN or STBY state.

**Figure 49** shows the constraints for entering and exiting NAP and PDN states. On the left side, an RDRAM exits NAP state at the end of cycle  $T_3$ . This RDRAM may not re-enter NAP or PDN state for an interval of  $t_{NU0}$ . The RDRAM enters NAP state at the end of cycle  $T_{13}$ . This RDRAM may not re-exit NAP state for an interval of  $t_{NU1}$ . The equations for these two parameters depend upon a number of factors, and are shown at the bottom of the figure. NAPX is the value in the NAPX field in the NAPX register.

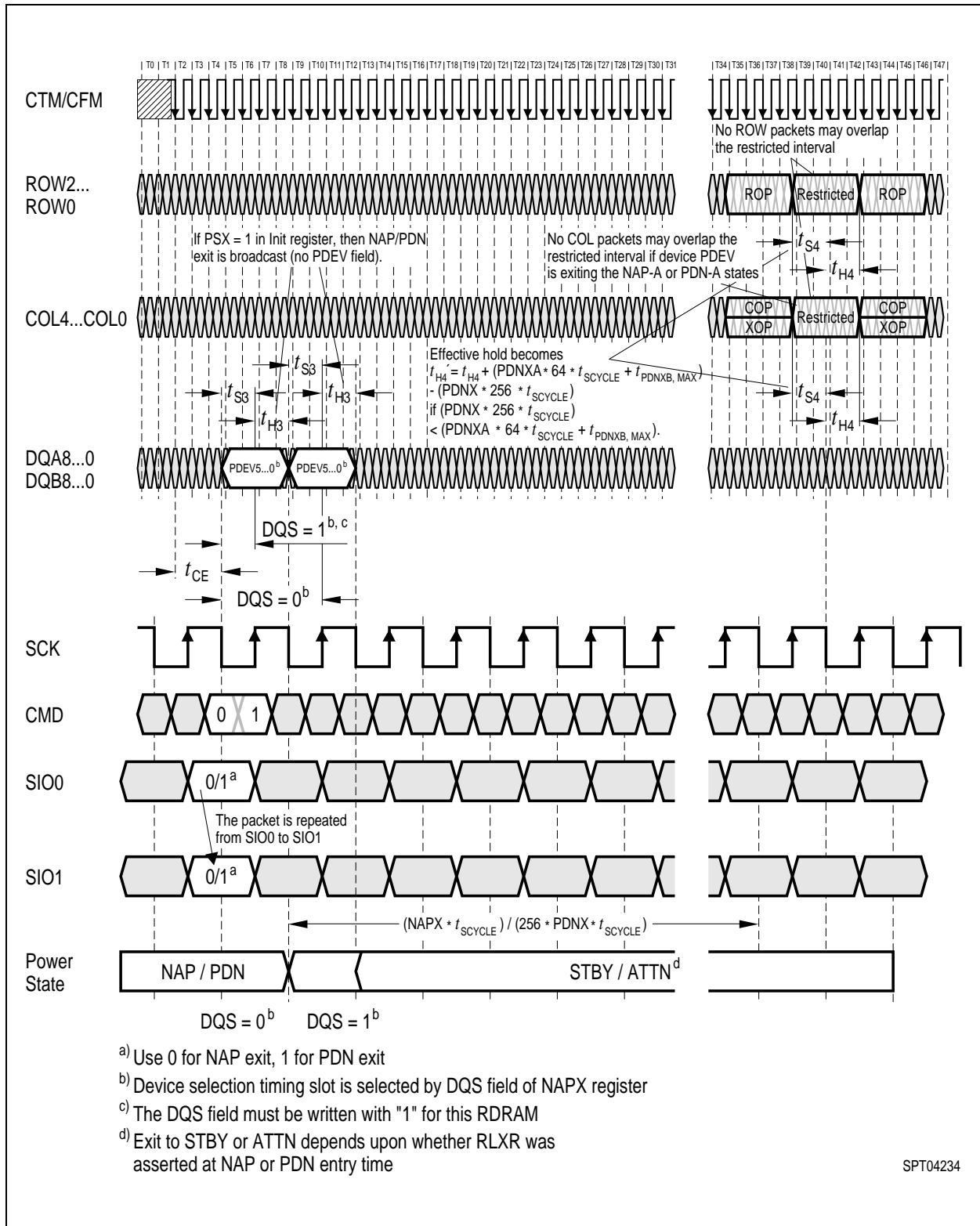


**Figure 46 STBY Entry (left) and STBY Exit (right)**

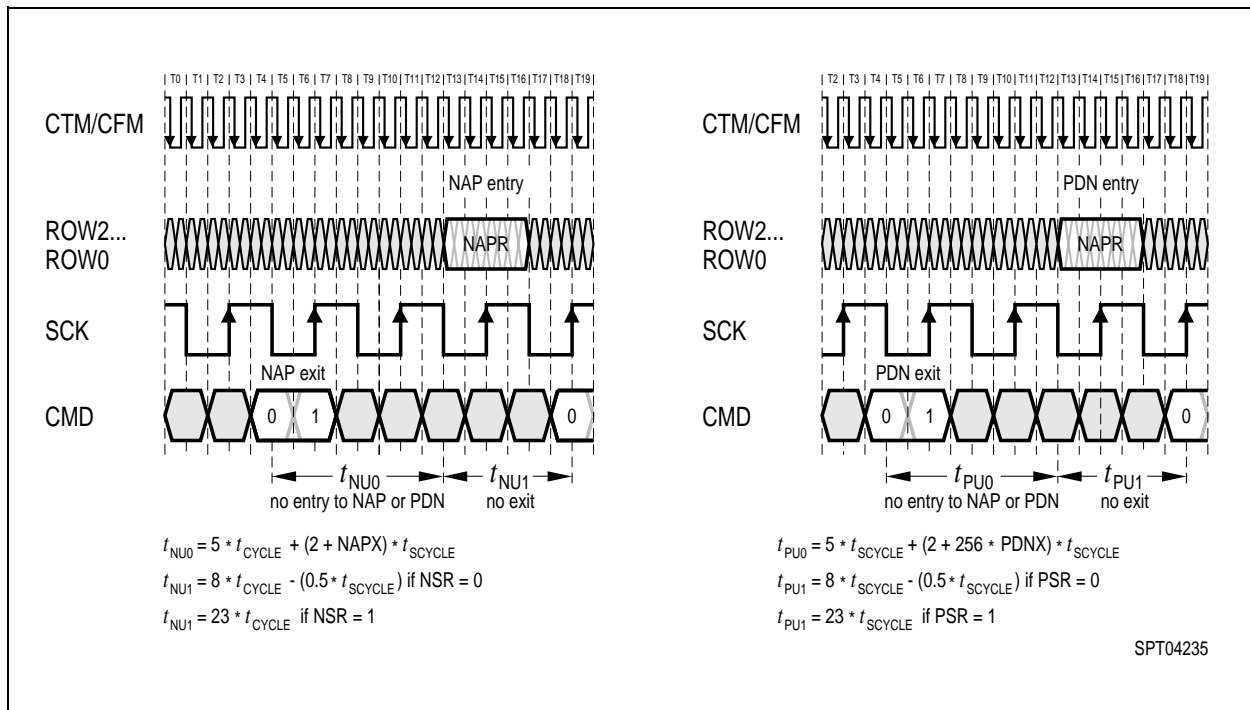


**Figure 47 NAP Entry (left) and PDN Entry (right)**

On the right side of **Figure 48**, an RDRAM exits PDN state at the end of cycle  $T_3$ . This RDRAM may not re-enter PDN or NAP state for an interval of  $t_{PU0}$ . The RDRAM enters PDN state at the end of cycle  $T_{13}$ . This RDRAM may not re-exit PDN state for an interval of  $t_{PU1}$ . The equations for these two parameters depend upon a number of factors, and are shown at the bottom of the figure. PDNX is the value in the PDNX field in the PDNX register.



**Figure 48 NAP and PDN Exit**



**Figure 49 NAP Entry/Exit Windows (left) and PDN Entry/Exit Windows (right)**

### Refresh

RDRAMs, like any other DRAM technology, use volatile storage cells which must be periodically refreshed. This is accomplished with the REFA command. **Figure 50** shows an example of this.

The REFA command in the transaction is typically a broadcast command (DR4T and DR4F are both set in the ROWR packet), so that in all devices bank number Ba is activated with row number REFR, where REFR is a control register in the RDRAM. When the command is broadcast and ATTN is set, the power state of the RDRAMs (ATTN or STBY) will remain unchanged. The controller increments the bank address Ba for the next REFA command. When Ba is equal to its maximum value, the RDRAM automatically increments REFR for the next REFA command.

On average, these REFA commands are sent once every  $t_{REF}/2^{BBIT+RBIT}$  (where BBIT are the number of bank address bits and RBIT are the number of row address bits) so that each row of each bank is refreshed once every  $t_{REF}$  interval.

The REFA command is equivalent to an ACT command, in terms of the way that it interacts with other packets (see **Table 12**). In the example, an ACT command is sent after  $t_{RR}$  to address b0, a different (non-adjacent) bank than the REFA command.

A second ACT command can be sent after a time  $t_{RC}$  to address c0, the same bank (or an adjacent bank) as the REFA command.

Note that a broadcast REFP command is issued a time  $t_{RAS}$  after the initial REFA command in order to precharge the refreshed bank in all RDRAMs. After a bank is given a REFA command, no other core operations (activate or precharge) should be issued to it until it receives a REFP.

It is also possible to interleave refresh transactions (not shown). In the figure, the ACT b0 command would be replaced by a REFA b0 command. The b0 address would be broadcast to all devices, and

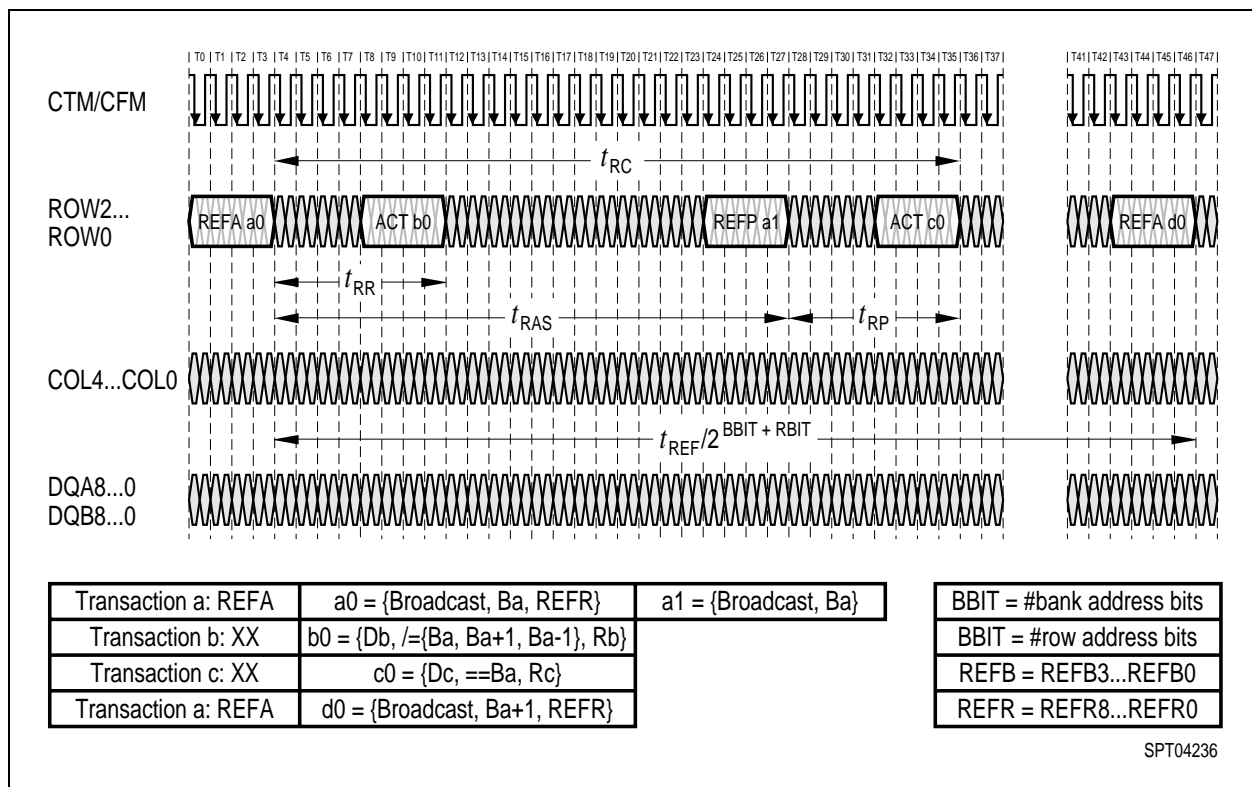


would be {Broadcast, Ba+2, REFR}. Note that the bank address should skip by two to avoid adjacent bank interference. A possible bank incrementing pattern would be: {13, 11, 9, 7, 5, 3, 1, 8, 10, 12, 14, 0, 2, 4, 6, 15, 29, 27, 25, 23, 21, 19, 17, 24, 26, 28, 30, 16, 18, 20, 22, 31}. Every time bank 31 is reached, the REFA command would automatically increment the REFR register.

A second refresh mechanism is available for use in PDN and NAP power states. This mechanism is called self-refresh mode. When the PDN power state is entered, or when NAP power state is entered with the NSR control register bit set, then self-refresh is automatically started for the RDRAM.

Self-refresh uses an internal time base reference in the RDRAM. This causes an activate and precharge to be carried out once in every  $t_{REF}/2^{BBIT+RBIT}$  interval. The REFB and REFR control registers are used to keep track of the bank and row being refreshed.

Before a controller places an RDRAM into self-refresh mode, it should perform REFA/REFP refreshes until the bank address is equal to the maximum value. This ensures that no rows are skipped. Likewise, when a controller returns an RDRAM to REFA/REFP refresh, it should start with the minimum bank address value (zero).



**Figure 50 REFA/REFP Refresh Transaction Example**

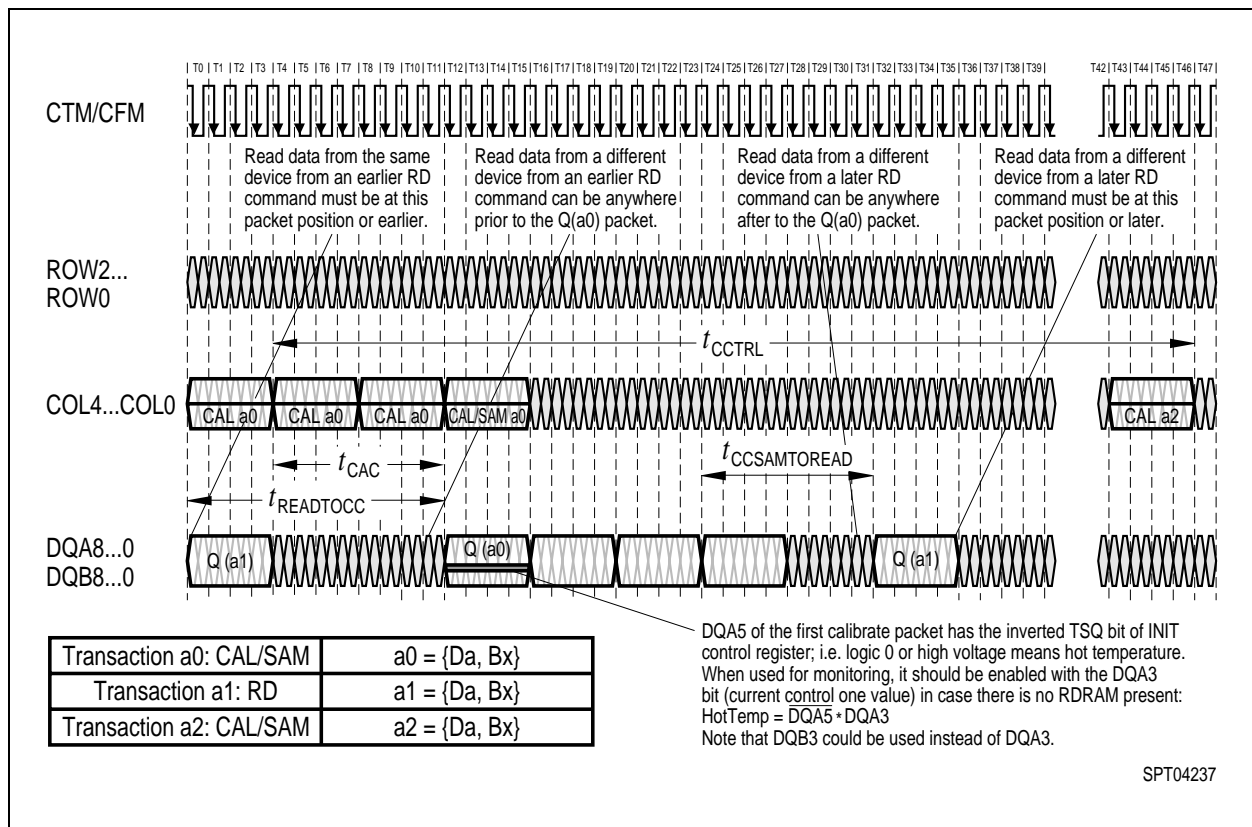
**Current and Temperature Control**

**Figure 51** shows an example of a transaction which performs current control calibration. It is necessary to perform this operation once to every RDRAM in every  $t_{CTRL}$  interval in order to keep the  $I_{OL}$  output current in its proper range.

This example uses four COLX packets with a CAL command. These cause the RDRAM to drive four calibration packets Q(a0) a time  $t_{CAC}$  later. An offset of  $t_{RDTOCC}$  must be placed between the Q(a0) packet and read data Q(a1) from the same device. These calibration packets are driven on the DQA4 ... 3 and DQB4 ... 3 wires. The TSQ bit of the INIT register is driven on the DQA5 wire during same interval as the calibration packets. The remaining DQA and DQB wires are not used during these calibration packets. The last COLX packet also contains a SAM command (concatenated with the CAL command). The RDRAM samples the last calibration packet and adjusts its  $I_{OL}$  current value.

Unlike REF commands, CAL and SAM commands cannot be broadcast. This is because the calibration packets from different devices would interfere. Therefore, a current control transaction must be sent every  $t_{CTRL}/N$ , where N is the number of RDRAMs on the Channel. The device field Da of the address a0 in the CAL/SAM command should be incremented after each transaction.

**Figure 23** shows an example of a temperature calibration sequence to the RDRAM. This sequence is broadcast once every  $t_{TEMP}$  interval to all the RDRAMs on the Channel. The TCEN and TCAL are ROP commands, and cause the slew rate of the output drivers to adjust for temperature drift. During the quiet interval  $t_{TQUIET}$  the devices being calibrated can't be read, but they can be written.



**Figure 51 Current Control CAL/SAM Transaction Example**

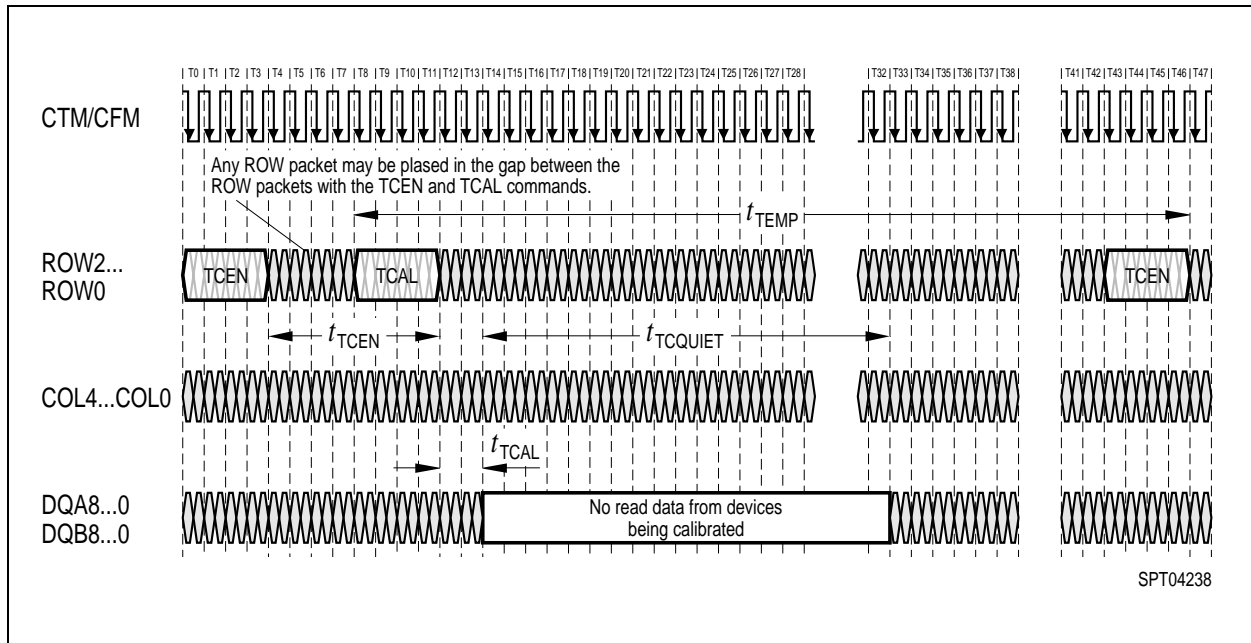


Figure 52 Temperature Calibration (TCEN-TCAL) Transactions to RDRAM

**Table 19 Electrical Conditions**

Parameter and Conditions	Symbol	Limit Values		Unit
		min.	max.	
Junction temperature under bias	$T_J$	-	100	°C
Supply voltage	$V_{DD}, V_{DDA}$	2.50 – 0.13	2.50 + 0.13	V
Supply voltage droop (DC) during NAP interval ( $t_{NLIMIT}$ )	$V_{DD,N}, V_{DDA,N}$	-	2.0	%
Supply voltage ripple (AC) during NAP interval ( $t_{NLIMIT}$ )	$V_{DD,N}, V_{DDA,N}$	- 2.0	2.0	%
Supply voltage for CMOS pins (2.5 V controllers)	$V_{CMOS}$	2.50 – 0.13	2.50 + 0.25	V
Supply voltage for CMOS pins (1.8 V controllers)		1.80 – 0.1	1.80 + 0.2	V
Termination voltage	$V_{TERM}$	1.80 – 0.1	1.80 + 0.1	V
Reference voltage	$V_{REF}$	1.40 – 0.2	1.40 + 0.2	V
RSL data input - low voltage	$V_{DIL}$	$V_{REF} - 0.5$	$V_{REF} - 0.2$	V
RSL data input - high voltage	$V_{DIH}$	$V_{REF} + 0.2$	$V_{REF} + 0.5$	V
RSL data input swing: $V_{DIS} = V_{DIH} - V_{DIL}$	$V_{DIS}$	0.4	1.0	V
RSL data asymmetry: $A_{DI} = [(V_{DIH} - V_{REF}) + (V_{DIL} - V_{REF})]/V_{DIS}$	$A_{DI}$	0	- 20	%
RSL clock input - crossing point of true and complement signals	$V_X$	1.3	1.8	V
RSL clock input - common mode $V_{CM} = (V_{CIH} + V_{CIL})/2$	$V_{CM}$	1.4	1.7	V
RSL clock input swing: $V_{CIS} = V_{CIH} - V_{CIL}$ (CTM, CTMN pins).	$V_{CIS,CTM}$	0.35	0.70	V
RSL clock input swing: $V_{CIS} = V_{CIH} - V_{CIL}$ (CFM, CFMN pins).	$V_{CIS,CFM}$	0.125	0.70	V
CMOS input low voltage	$V_{IL,CMOS}$	- 0.3	$V_{CMOS}/2 - 0.25$	V
CMOS input high voltage	$V_{IH,CMOS}$	$V_{CMOS}/2 + 0.25$	$V_{CMOS} + 0.3$	V

**Table 20 Timing Conditions**

Parameter	Symbol	Limit Values		Unit	Figure
		min.	max.		
CTM and CFM cycle times (-600)	$t_{\text{CYCLE}}$	3.33	3.83	ns	<b>Figure 53</b>
CTM and CFM cycle times (-711)		2.80	3.83	ns	<b>Figure 53</b>
CTM and CFM cycle times (-800)		2.50	3.83	ns	<b>Figure 53</b>
CTM and CFM input rise and fall times	$t_{\text{CR}}, t_{\text{CF}}$	0.2	0.5	ns	<b>Figure 53</b>
CTM and CFM high and low times	$t_{\text{CH}}, t_{\text{CL}}$	40%	60%	$t_{\text{CYCLE}}$	<b>Figure 53</b>
CTM-CFM differential (MSE/MS = 0/0)	$t_{\text{TR}}$	0.0	1.0	$t_{\text{CYCLE}}$	<b>Figure 42</b>
CTM-CFM differential (MSE/MS = 1/1) <sup>1)</sup>		0.9	1.0		<b>Figure 53</b>
Domain crossing window	$t_{\text{DCW}}$	- 0.1	0.1	$t_{\text{CYCLE}}$	<b>Figure 59</b>
DQA/DQB/ROW/COL input rise/fall times	$t_{\text{DR}}, t_{\text{DF}}$	0.2	0.65	ns	<b>Figure 54</b>
DQA/DQB/ROW/COL-to-CFM set/hold @ $t_{\text{CYCLE}} = 3.33$ ns	$t_{\text{S}}, t_{\text{H}}$	0.275 <sup>2),d)</sup>	-	ns	<b>Figure 54</b>
DQA/DQB/ROW/COL-to-CFM set/hold @ $t_{\text{CYCLE}} = 2.81$ ns		0.240 <sup>3),4)</sup>	-	ns	<b>Figure 54</b>
DQA/DQB/ROW/COL-to-CFM set/hold @ $t_{\text{CYCLE}} = 2.50$ ns		0.200 <sup>d)</sup>	-	ns	<b>Figure 54</b>
SIO0, SIO1 input rise and fall times	$t_{\text{DR1}}, t_{\text{DF1}}$	-	5.0	ns	<b>Figure 56</b>
CMD, SCK input rise and fall times	$t_{\text{DR2}}, t_{\text{DF2}}$	-	2.0	ns	<b>Figure 56</b>
SCK cycle time - Serial control register transactions	$t_{\text{CYCLE1}}$	1000	-	ns	<b>Figure 56</b>
SCK cycle time - Power transitions		10	-	ns	<b>Figure 56</b>
SCK high and low times	$t_{\text{CH1}}, t_{\text{CL1}}$	4.25	-	ns	<b>Figure 56</b>
CMD setup time to SCK rising or falling edge <sup>5)</sup>	$t_{\text{S1}}$	1.25	-	ns	<b>Figure 56</b>
CMD hold time to SCK rising or falling edge <sup>c)</sup>	$t_{\text{H1}}$	1	-	ns	<b>Figure 56</b>
SIO0 setup time to SCK falling edge	$t_{\text{S2}}$	40	-	ns	<b>Figure 56</b>
SIO0 hold time to SCK falling edge	$t_{\text{H2}}$	40	-	ns	<b>Figure 56</b>
PDEV setup time on DQA5 ... 0 to SCK rising edge.	$t_{\text{S3}}$	0	-	ns	<b>Figure 48, Figure 57</b>
PDEV hold time on DQA5 ... 0 to SCK rising edge.	$t_{\text{H3}}$	5.5	-	ns	
ROW2 ... 0, COL4 ... 0 setup time for quiet window	$t_{\text{S4}}$	- 1	-	$t_{\text{CYCLE}}$	<b>Figure 48</b>
ROW2 ... 0, COL4 ... 0 hold time for quiet window <sup>6)</sup>	$t_{\text{H4}}$	5	-	$t_{\text{CYCLE}}$	<b>Figure 48</b>

**Table 20 Timing Conditions (cont'd)**

Parameter	Symbol	Limit Values		Unit	Figure
		min.	max.		
CMOS input low voltage - over/undershoot voltage duration is less than or equal to 5 ns	$V_{IL,CMOS}$	-0.7	$V_{CMOS}/2 - 0.4$	V	–
CMOS input high voltage - over/undershoot voltage duration is less than or equal to 5 ns	$V_{IH,CMOS}$	$V_{CMOS}/2 + 0.4$	$V_{CMOS} + 0.7$	V	–
Quiet on ROW/COL bits during NAP/PDN entry	$t_{NPQ}$	4	–	$t_{CYCLE}$	<b>Figure 47</b>
Offset between read data and CC packets (same device)	$t_{READTOCC}$	12	–	$t_{CYCLE}$	<b>Figure 51</b>
Offset between CC packet and read data (same device)	$t_{CCSAMTOREAD}$	8	–	$t_{CYCLE}$	<b>Figure 51</b>
CTM/CFM stable before NAP/PDN exit	$t_{CE}$	2	–	$t_{CYCLE}$	<b>Figure 48</b>
CTM/CFM stable after NAP/PDN entry	$t_{CD}$	100	–	$t_{CYCLE}$	<b>Figure 47</b>
ROW packet to COL packet ATTN framing delay	$t_{FRM}$	7	–	$t_{CYCLE}$	<b>Figure 46</b>
Maximum time in NAP mode	$t_{NLIMIT}$	–	10.0	μs	<b>Figure 45</b>
Refresh interval	$t_{REF}$	–	32	ms	<b>Figure 50</b>
Current control interval	$t_{CTRL}$	$34 t_{CYCLE}$	100 ms	ms/ $t_{CYCLE}$	<b>Figure 51</b>
Temperature control interval	$t_{TEMP}$	–	100	ms	<b>Figure 23</b>
TCE command to TCAL command	$t_{TCEN}$	150	–	$t_{CYCLE}$	<b>Figure 23</b>
TCAL command to quiet window	$t_{TCAL}$	2	2	$t_{CYCLE}$	<b>Figure 23</b>
Quiet window (no read data)	$t_{TCQUIET}$	140	–	$t_{CYCLE}$	<b>Figure 23</b>
RDRAM delay (no RSL operations allowed)	$t_{PAUSE}$	–	200.0	μs	page 43

- 1) MSE/MS are fields of the SKIP register. For this combination (skip override) the  $t_{DCW}$  parameter range is effectively 0.0 to 0.0.
- 2) This parameter also applies to a -800 or -711 part when operated with  $t_{CYCLE} = 3.33$  ns.
- 3) This parameter also applies to a -800 part when operated with  $t_{CYCLE} = 2.81$  ns.
- 4)  $t_{S,MIN}$  and  $t_{H,MIN}$  for other  $t_{CYCLE}$  values can be interpolated between or extrapolated from the timings at the 3 specified  $t_{CYCLE}$  values.
- 5) With  $V_{IL,CMOS} = 0.5 V_{CMOS} - 0.6$  V and  $V_{IH,CMOS} = 0.5 V_{CMOS} + 0.6$  V
- 6) Effective hold becomes  $t_{H4}' = t_{H4} + [PDNXA \times 64 \times t_{SCYCLE} + t_{PDNXB,MAX}] - [PDNX \times 256 \times t_{SCYCLE}]$  if  $[PDNX \times 256 \times t_{SCYCLE}] < [PDNXA \times 64 \times t_{SCYCLE} + t_{PDNXB,MAX}]$ . See **Figure 48**.

**Table 21 Electrical Characteristics**

Parameter	Symbol	Limit Values		Unit
		min.	max.	
$V_{REF}$ current @ $V_{REF,MAX}$	$I_{REF}$	- 10	10	$\mu$ A
RSL output high current @ ( $0 \leq V_{OUT} \leq V_{DD}$ )	$I_{OH}$	- 10	10	$\mu$ A
RSL $I_{OL}$ current @ $V_{OL} = 0.9$ V, $V_{DD,MIN}$ , $T_{J,MAX}$ <sup>1)</sup>	$I_{ALL}$	30.0	90.0	mA
RSL $I_{OL}$ current resolution step	$\Delta I_{OL}$	-	2.0	mA
Dynamic output impedance	$r_{OUT}$	150	-	$\Omega$
CMOS input leakage current @ ( $0 \leq V_{I,CMOS} \leq V_{CMOS}$ )	$I_{I,CMOS}$	- 10.0	10.0	$\mu$ A
CMOS output voltage @ $I_{OL,CMOS} = 1.0$ mA	$V_{OL,CMOS}$	-	0.3	V
CMOS output high voltage @ $I_{OH,CMOS} = - 0.25$ mA	$V_{OH,CMOS}$	$V_{CMOS} - 0.3$	-	V

<sup>1)</sup> This measurement is made in manual current control mode; i.e. with all output device legs sinking current.

**Table 22 Timing Characteristics**

Parameter	Symbol	Limit Values		Unit	Figure
		min.	max.		
$t_Q$	CTM-to-DQA/DQB output time @ $t_{CYCLE} = 3.33$ ns	- 0.350 <sup>1),3)</sup>	+ 0.350 <sup>1),3)</sup>	ns	<b>Figure 55</b> <b>Figure 55</b> <b>Figure 55</b>
	CTM-to-DQA/DQB output time @ $t_{CYCLE} = 2.81$ ns	- 0.300 <sup>2),3)</sup>	+ 0.300 <sup>2),3)</sup>	ns	
	CTM-to-DQA/DQB output time @ $t_{CYCLE} = 2.50$ ns	- 0.260 <sup>3)</sup>	+ 0.260 <sup>3)</sup>	ns	
$t_{QR}, t_{QF}$	DQA/DQB output rise and fall times	0.2	0.45	ns	<b>Figure 55</b>
$t_{Q1}$	SCK(neg)-to-SIO0 delay @ $C_{LOAD,MAX} = 20$ pF (SD read data valid).	-	10	ns	<b>Figure 58</b>
$t_{HR}$	SCK(pos)-to-SIO0 delay @ $C_{LOAD,MAX} = 20$ pF (SD read data hold).	2	-	ns	<b>Figure 58</b>
$t_{QR1}, t_{QF1}$	SIO <sub>OUT</sub> rise/fall @ $C_{LOAD,MAX} = 20$ pF	-	5	ns	<b>Figure 58</b>
$t_{PROP1}$	SIO0-to-SIO1 or SIO1-to-SIO0 delay @ $C_{LOAD,MAX} = 20$ pF	-	10	ns	<b>Figure 58</b>
$t_{NAPXA}$	NAP exit delay - phase A	-	50	ns	<b>Figure 48</b>
$t_{NAPXB}$	NAP exit delay - phase B	-	40	ns	<b>Figure 48</b>
$t_{PDNXA}$	PDN exit delay - phase A	-	4	$\mu$ s	<b>Figure 48</b>
$t_{PDNXB}$	PDN exit delay - phase B	-	9000	$t_{CYCLE}$	<b>Figure 48</b>

**Table 22 Timing Characteristics (cont'd)**

Parameter	Symbol	Limit Values		Unit	Figure
		min.	max.		
$t_{AS}$	ATTN-to-STBY power state delay	–	1	$t_{CYCLE}$	<b>Figure 46</b>
$t_{SA}$	STBY-to-ATTN power state delay	–	0	$t_{CYCLE}$	<b>Figure 46</b>
$t_{ASN}$	ATTN/STBY-to-NAP power state delay	–	8	$t_{CYCLE}$	<b>Figure 47</b>
$t_{ASP}$	ATTN/STBY-to-PDN power state delay	–	8	$t_{CYCLE}$	<b>Figure 47</b>

1) This parameter also applies to a -800 or -711 part when operated with  $t_{CYCLE} = 3.33$  ns.

2) This parameter also applies to a -800 part when operated with  $t_{CYCLE} = 2.81$  ns.

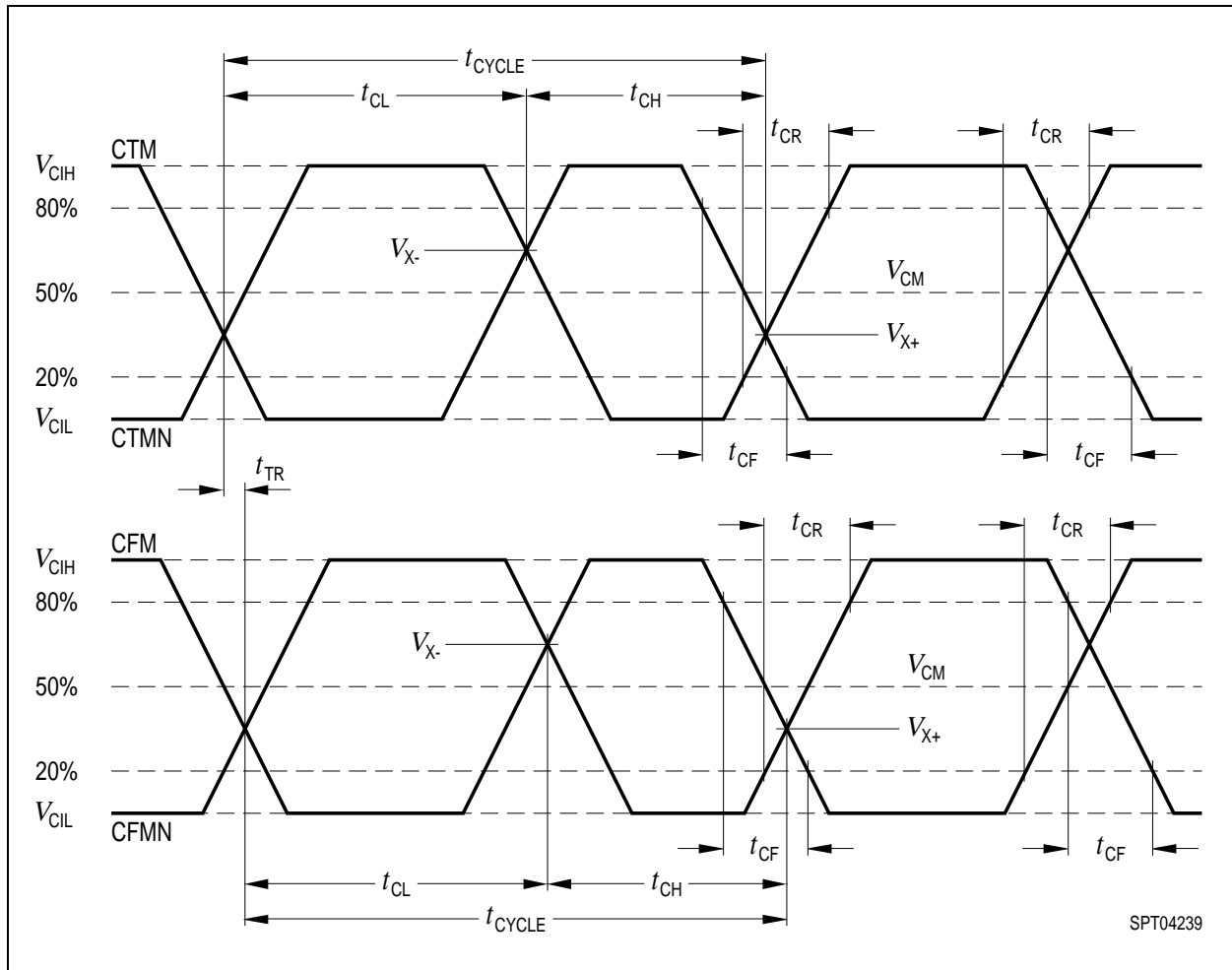
3)  $t_{Q,MIN}$  and  $t_{Q,MAX}$  for other  $t_{CYCLE}$  values can be interpolated between or extrapolated from the timings at the 3 specified  $t_{CYCLE}$  values.

### RSL - Clocking

**Figure 53** is a timing diagram which shows the detailed requirements for the RSL clock signals on the Channel.

The CTM and CTMN are differential clock inputs used for transmitting information on the DQA and DQB, outputs. Most timing is measured relative to the points where they cross. The  $t_{CYCLE}$  parameter is measured from the falling CTM edge to the falling CTM edge. The  $t_{CL}$  and  $t_{CH}$  parameters are measured from falling to rising and rising to falling edges of CTM. The  $t_{CR}$  and  $t_{CF}$  rise- and fall-time parameters are measured at the 20% and 80% points.





**Figure 53 RSL Timing - Clock Signals**

The CFM and CFMN are differential clock outputs used for receiving information on the DQA, DQB, ROW and COL outputs. Most timing is measured relative to the points where they cross. The  $t_{CYCLE}$  parameter is measured from the falling CFM edge to the falling CFM edge. The  $t_{CL}$  and  $t_{CH}$  parameters are measured from falling to rising and rising to falling edges of CFM. The  $t_{CR}$  and  $t_{CF}$  rise- and fall-time parameters are measured at the 20% and 80% points.

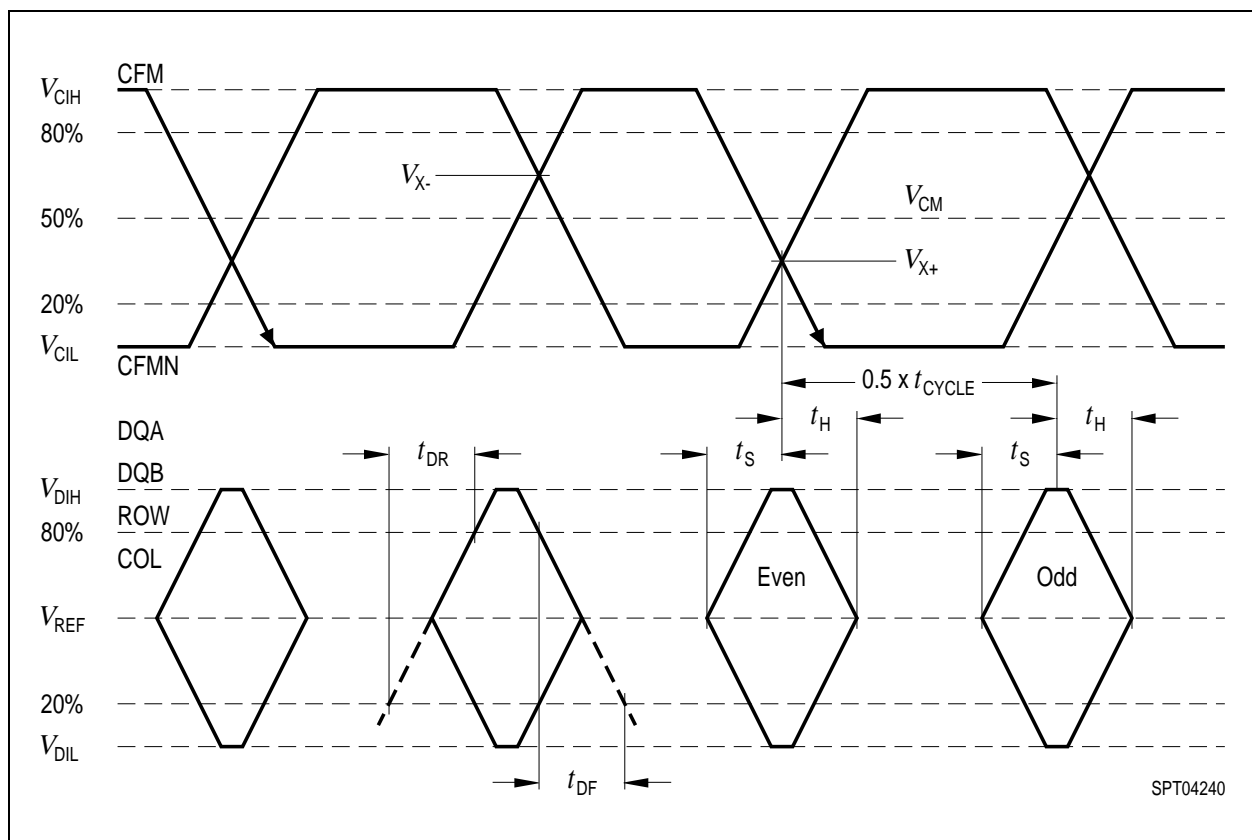
The  $t_{TR}$  parameter specifies the phase difference that may be tolerated with respect to the CTM and CFM differential clock inputs (the CTM pair is always earlier).

**RSL - Receive Timing**

**Figure 54** is a timing diagram which shows the detailed requirements for the RSL input signals on the Channel.

The DQA, DQB, ROW, and COL signals are inputs which receive information transmitted by a Direct RAC on the Channel. Each signal is sampled twice per  $t_{CYCLE}$  interval. The set/hold window of the sample points is  $t_S/t_H$ . The sample points are centered at the 0% and 50% points of a cycle, measured relative to the crossing points of the falling CFM clock edge. The set and hold parameters are measured at the  $V_{REF}$  voltage point of the input transition.

The  $t_{DR}$  and  $t_{DF}$  rise- and fall-time parameters are measured at the 20% and 80% points of the input transition.



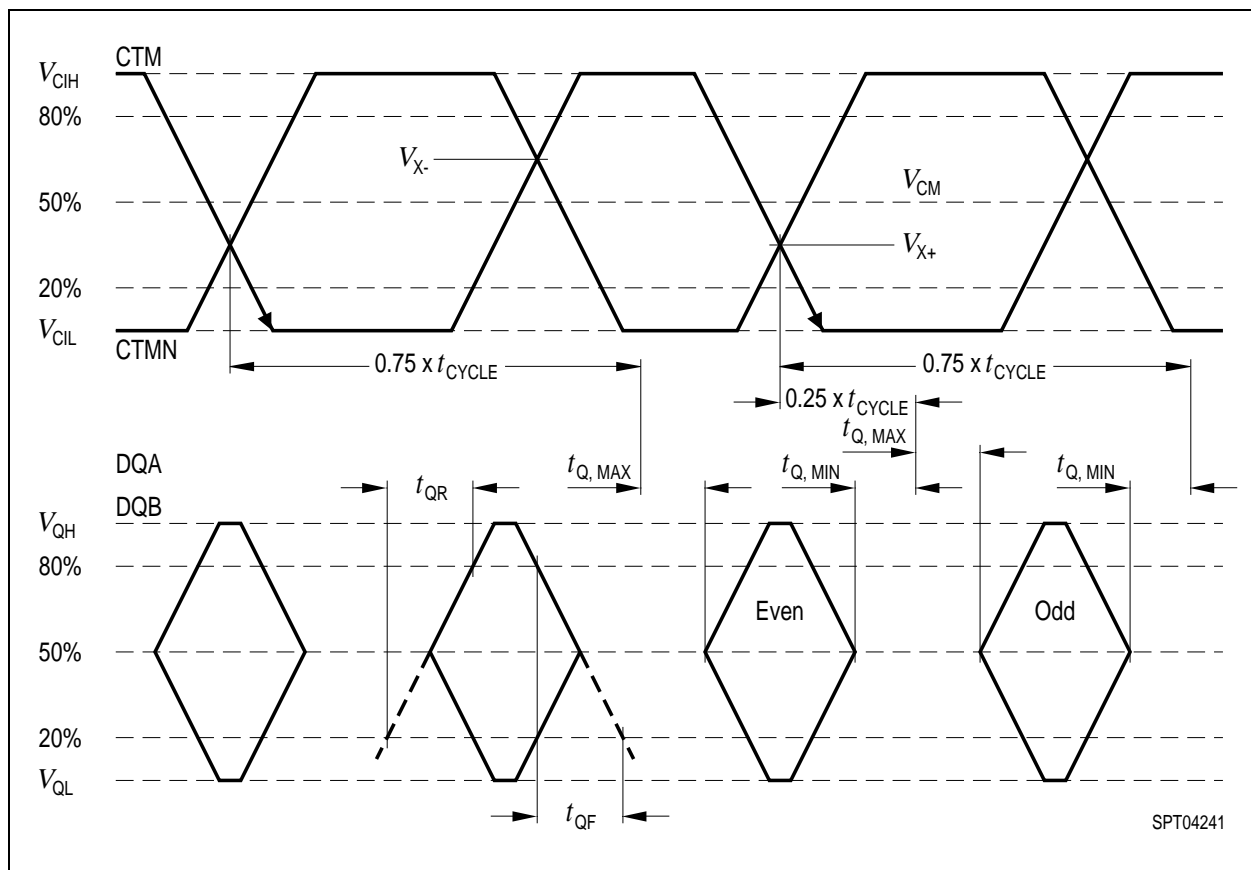
**Figure 54 RSL Timing - Data Signals for Receive**

### RSL - Transmit Timing

**Figure 55** is a timing diagram which shows the detailed requirements for the RSL output signals on the Channel.

The DQA and DQB signals are outputs to transmit information that is received by a Direct RAC on the Channel. Each signal is driven twice per  $t_{\text{CYCLE}}$  interval. The beginning and end of the even transmit window is at the 75% point of the previous cycle and at the 25% point of the current cycle. The beginning and end of the odd transmit window is at the 25% point and at the 75% point of the current cycle. These transmit points are measured relative to the crossing points of the falling CTM clock edge. The size of the actual transmit window is less than the ideal  $t_{\text{CYCLE}}/2$ , as indicated by the non-zero values of  $t_{\text{Q,MIN}}$  and  $t_{\text{Q,MAX}}$ . The  $t_{\text{Q}}$  parameters are measured at the 50% voltage point of the output transition.

The  $t_{\text{QR}}$  and  $t_{\text{QF}}$  rise- and fall-time parameters are measured at the 20% and 80% points of the output transition.



**Figure 55 RSL Timing - Data Signals for Transmit**

### CMOS - Receive Timing

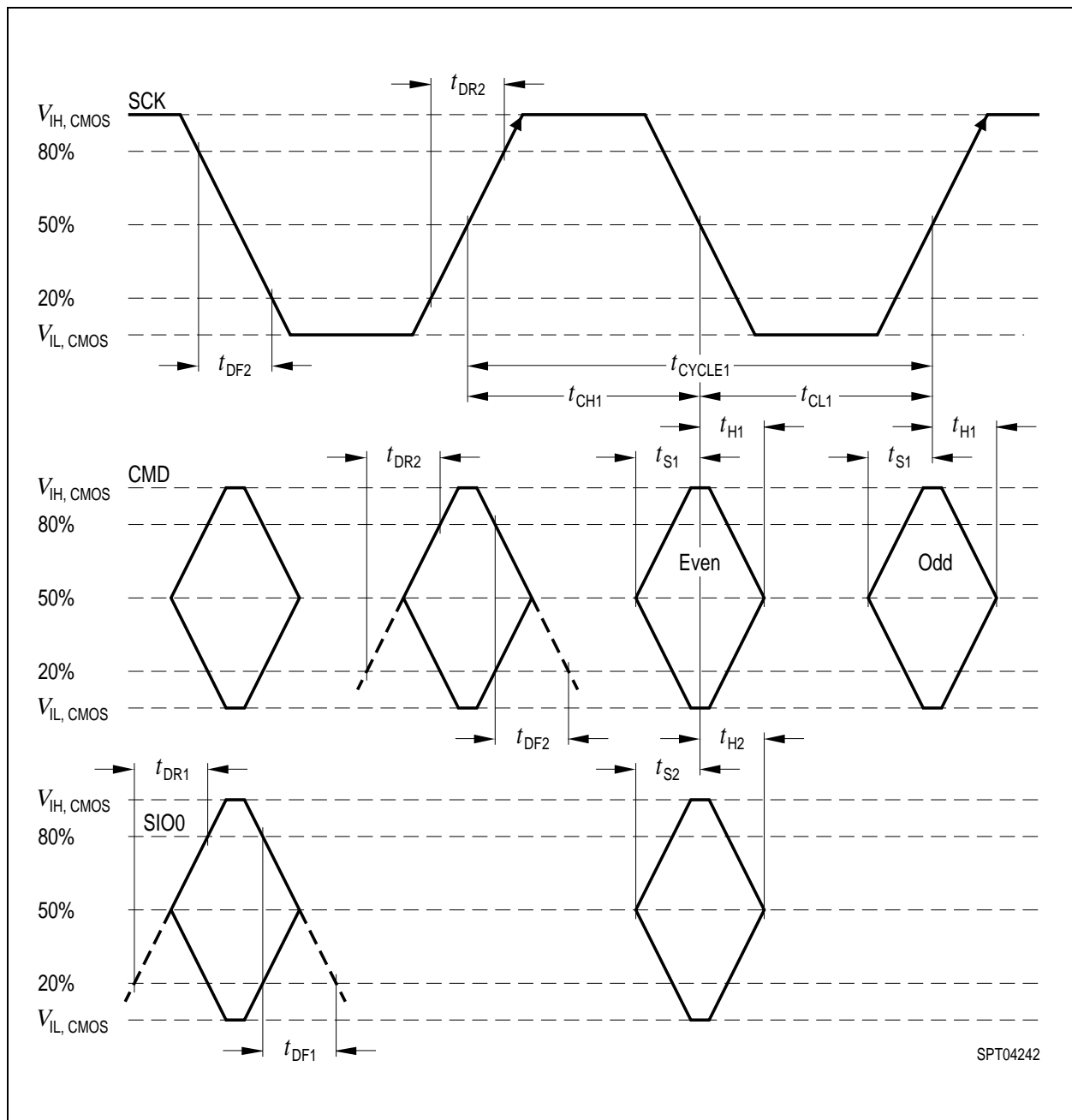
**Figure 56** is a timing diagram which shows the detailed requirements for the CMOS input signals.

The CMD and SIO0 signals are inputs which receive information transmitted by a controller (or by another RDRAM's SIO1 output). SCK is the CMOS clock signal driven by the controller. All signals are high true.

The cycle time, high phase time, and low phase time of the SCK clock are  $t_{CYCLE1}$ ,  $t_{CH1}$  and  $t_{CL1}$ , all measured at the 50% level. The rise and fall times of SCK, CMD, and SIO0 are  $t_{DR1}$  and  $t_{DF1}$ , measured at the 20% and 80% levels.

The CMD signal is sampled twice per  $t_{CYCLE1}$  interval, on the rising edge (odd data) and the falling edge (even data). The set/hold window of the sample points is  $t_{S1}/t_{H1}$ . The SCK and CMD timing points are measured at the 50% level.

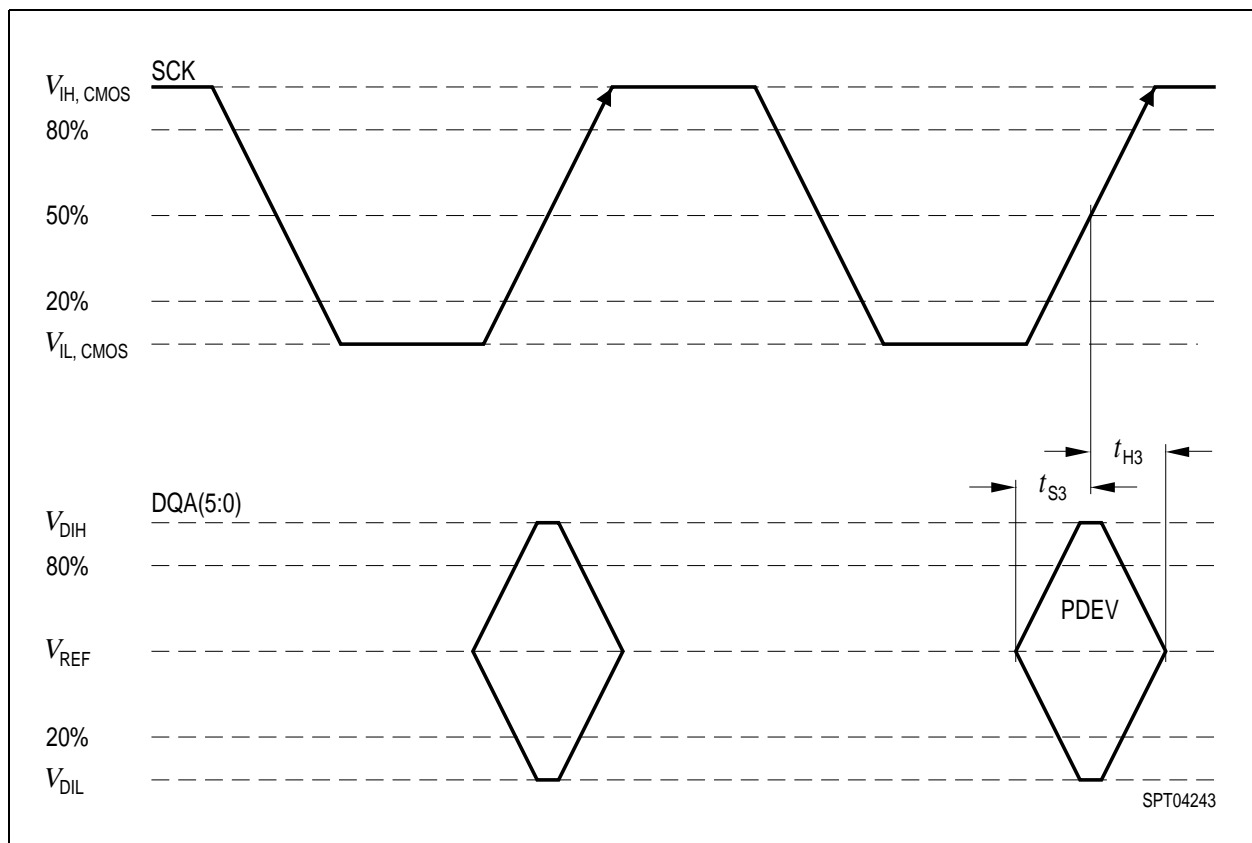
The SIO0 signal is sampled once per  $t_{CYCLE1}$  interval on the falling edge. The set/hold window of the sample points is  $t_{S2}/t_{H2}$ . The SCK and SIO0 timing points are measured at the 50% level.



**Figure 56 CMOS Timing - Data Signals for Receive**

The SCK clock is also used for sampling data on RSL inputs in one situation. **Figure 48** shows the PDN and NAP exit sequences. If the PSX field of the INIT register is one (see **Figure 27**), then the PDN and NAP exit sequences are broadcast; i.e. all RDRAMs that are in PDN or NAP will perform the exit sequence. If the PSX field of the INIT register is zero, then the PDN and NAP exit sequences are directed; i.e. only one RDRAM that is in PDN or NAP will perform the exit sequence.

The address of that RDRAM is specified on the DQA[5:0] bus in the set hold window  $t_{S3}/t_{H3}$  around the rising edge of SCK. This is shown in **Figure 57**. The SCK timing point is measured at the 50% level, and the DQA[5:0] bus signals are measured at the  $V_{REF}$  level.



**Figure 57 CMOS Timing - Device Address for NAP or PDN Exit**

### CMOS - Transmit Timing

**Figure 58** is a timing diagram which shows the detailed requirements for the CMOS output signals. The SIO0 signal is driven once per  $t_{CYCLE1}$  interval on the falling edge. The clock-to-output window is  $t_{Q1,MIN}/t_{Q1,MAX}$ . The SCK and SIO0 timing points are measured at the 50% level. The rise and fall times of SIO0 are  $t_{QR1}$  and  $t_{QF1}$ , measured at the 20% and 80% levels.

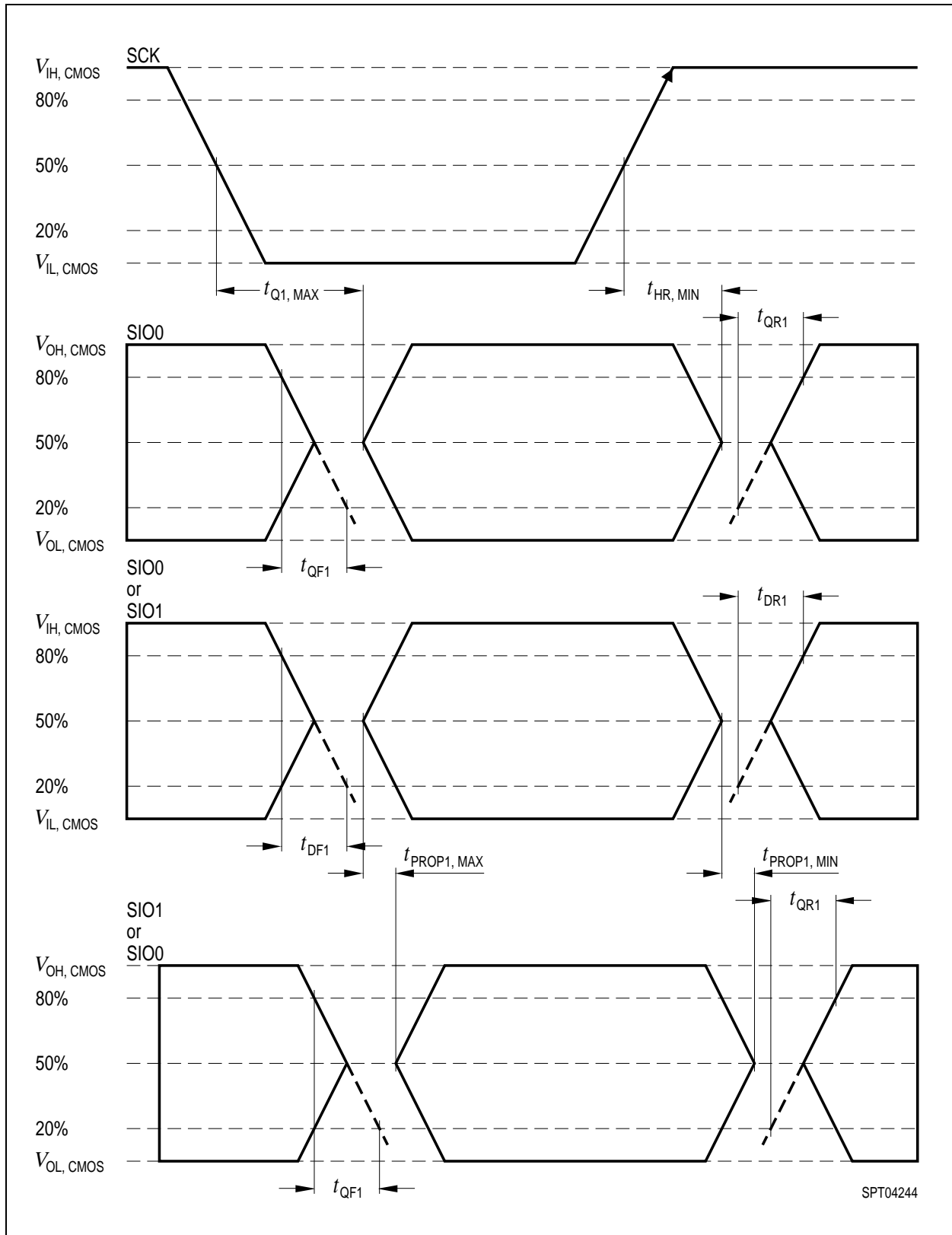


Figure 58 CMOS Timing - Data Signals for Transmit

**Figure 58** also shows the combinational path connecting SIO0 to SIO1 and the path connecting SIO1 to SIO0 (read data only). The  $t_{PROP1}$  parameter specified this propagation delay. The rise and fall times of SIO0 and SIO1 inputs must be  $t_{DR1}$  and  $t_{DF1}$ , measured at the 20% and 80% levels. The rise and fall times of SIO0 and SIO1 outputs are  $t_{QR1}$  and  $t_{QF1}$ , measured at the 20% and 80% levels.

### RSL - Domain Crossing Window

When read data is returned by the RDRAM, information must cross from the receive clock domain (CFM) to the transmit clock domain (CTM). The  $t_{TR}$  parameter permits the CFM to CTM phase to vary through an entire cycle; i.e. there is no restriction on the alignment of these two clocks. A second parameter  $t_{DCW}$  is needed in order to describe how the delay between a RD command packet and read data packet varies as a function of the  $t_{TR}$  value.

**Figure 59** shows this timing for five distinct values of  $t_{TR}$ . Case A ( $t_{TR} = 0$ ) is what has been used throughout this document. The delay between the RD command and read data is  $t_{CAC}$ . As  $t_{TR}$  varies from zero to  $t_{CYCLE}$  (cases A through E), the command to data delay is  $(t_{CAC} - t_{TR})$ . When the  $t_{TR}$  value is in the range 0 to  $t_{DCW,MAX}$ , the command to data delay can also be  $(t_{CAC} - t_{TR} - t_{CYCLE})$ . This is shown as cases A' and B' (the gray packets). Similarly, when the  $t_{TR}$  value is in the range  $(t_{CYCLE} + t_{DCW,MIN})$  to  $t_{CYCLE}$ , the command to data delay can also be  $(t_{CAC} - t_{TR} + t_{CYCLE})$ . This is shown as cases D' and E' (the gray packets). The RDRAM will work reliably with either the white or gray packet timing. The delay value is selected at initialization, and remains fixed thereafter.





## Timing Parameters

**Table 23 Timing Parameter Summary**

Parameter	Description	Min -40 -800	Min -45 -800	Min -45 -711	Min -53 -600	Max	Unit	Figure
$t_{RC}$	Row Cycle time of RDRAM banks - the interval between ROWA packets with ACT commands to the same bank.	28	28	28	28	–	$t_{CYCLE}$	Figure 15 Figure 16
$t_{RAS}$	$\overline{RAS}$ -asserted time of RDRAM bank - the interval between ROWA packet with ACT command and next ROWR packet with PRER <sup>1</sup> command to the same bank.	20	20	20	20	64 $\mu s^2$	$t_{CYCLE}$	Figure 15 Figure 16
$t_{RP}$	Row Precharge time of RDRAM banks - the interval between ROWR packet with PRER <sup>a</sup> command and next ROWA packet with ACT command to the same bank.	8	8	8	8	–	$t_{CYCLE}$	Figure 15 Figure 16
$t_{PP}$	Precharge-to-precharge time of RDRAM device - the interval between successive ROWR packets with PRER <sup>a</sup> commands to any banks of the same device.	8	8	8	8	–	$t_{CYCLE}$	Figure 12
$t_{RR}$	$\overline{RAS}$ -to- $\overline{RAS}$ time of RDRAM device - the interval between successive ROWA packets with ACT commands to any banks of the same device.	8	8	8	8	–	$t_{CYCLE}$	Figure 13
$t_{RCD}$	$\overline{RAS}$ -to- $\overline{CAS}$ Delay - the interval from ROWA packet with ACT command to COLC packet with RD or WR command). Note - the $\overline{RAS}$ -to- $\overline{CAS}$ delay seen by the RDRAM core ( $t_{RCD-C}$ ) is equal to $t_{RCD-C} = 1 + t_{RCD}$ because of differences in the row and column paths through the RDRAM interface.	7	9	7	7	–	$t_{CYCLE}$	Figure 15 Figure 16
$t_{CAC}$	$\overline{CAS}$ Access delay - the interval from RD command to Q read data. The equation for $t_{CAC}$ is given in the TPARM register in Figure 39.	8	8	8	8	12	$t_{CYCLE}$	Figure 4 Figure 39
$t_{CWD}$	$\overline{CAS}$ Write Delay (interval from WR command to D write data).	6	6	6	6	6	$t_{CYCLE}$	Figure 4
$t_{CC}$	$\overline{CAS}$ -to- $\overline{CAS}$ time of RDRAM bank - the interval between successive COLC commands).	4	4	4	4	-	$t_{CYCLE}$	Figure 15 Figure 16
$t_{PACKET}$	Length of ROWA, ROWR, COLC, COLM or COLX packet.	4	4	4	4	4	$t_{CYCLE}$	Figure 3
$t_{RTR}$	Interval from COLC packet with WR command to COLC packet which causes retire, and to COLM packet with bytemask.	8	8	8	8	-	$t_{CYCLE}$	Figure 17
$t_{OFFP}$	The interval (offset) from COLC packet with RDA command, or from COLC packet with retire command (after WRA automatic precharge), or from COLC packet with PREC command, or from COLX packet with PREX command to the equivalent ROWR packet with PRER. The equation for $t_{OFFP}$ is given in the TPARM register in Figure 39.	4	4	4	4	4	$t_{CYCLE}$	Figure 14 Figure 39

**Table 23 Timing Parameter Summary (cont'd)**

Parameter	Description	Min -40 -800	Min -45 -800	Min -45 -711	Min -53 -600	Max	Unit	Figure
$t_{RDP}$	Interval from last COLC packet with RD command to ROWR packet with PRER.	4	4	4	4	–	$t_{CYCLE}$	Figure 15
$t_{RTP}$	Interval from last COLC packet with automatic retire command to ROWR packet with PRER.	4	4	4	4	–	$t_{CYCLE}$	Figure 16

1) Or equivalent PREC or PREX command. See **Figure 14**.

2) This is a constraint imposed by the core, and is therefore in units of  $\mu\text{s}$  rather than  $t_{CYCLE}$ .

**Table 24 Absolute Maximum Ratings**

Parameter	Symbol	Limit Values		Unit
		min.	max.	
Voltage applied to any RSL or CMOS pin with respect to GND	$V_{I,ABS}$	– 0.3	$V_{DD} + 0.3$	V
Voltage on $V_{DD}$ and $V_{DDA}$ with respect to GND	$V_{DD,ABS}$ , $V_{DDA,ABS}$	– 0.5	$V_{DD} + 1.0$	V
Storage temperature	$T_{STORE}$	– 50	100	°C

**$I_{DD}$  - Supply Current Profile**

**Table 25 Supply Current Profile**

RDRAM Blocks Consuming Power <sup>1)</sup>	$I_{DD}$ Value	Limit Values			Unit
		max. -800	max. -711	max. -600	
Self-refresh only for INIT.LSR = 0	$I_{DD,PDN}$	3000	3000	3000	μA
T/RCLK-Nap	$I_{DD,NAP}$	4	4	4	mA
T/RCLK, ROW-demux	$I_{DD,STBY}$	100	95	90	mA
T/RCLK, ROW-demux, COL-demux	$I_{DD,ATTN}$	150	145	140	mA
T/RCLK, ROW-demux, COL-demux, DQ-demux, 1 × WR-SenseAmp, 4 × ACT-Bank	$I_{DD,ATTN-W}$	575/635 <sup>2)</sup>	515/570	450/495 <sup>2)</sup>	mA
T/RCLK, ROW-demux, COL-demux, DQ-demux, 1 × RD-SenseAmp, 4 × ACT-Bank <sup>3)</sup>	$I_{DD,ATTN-R}$	520/575 <sup>2)</sup>	470/520	410/450	mA

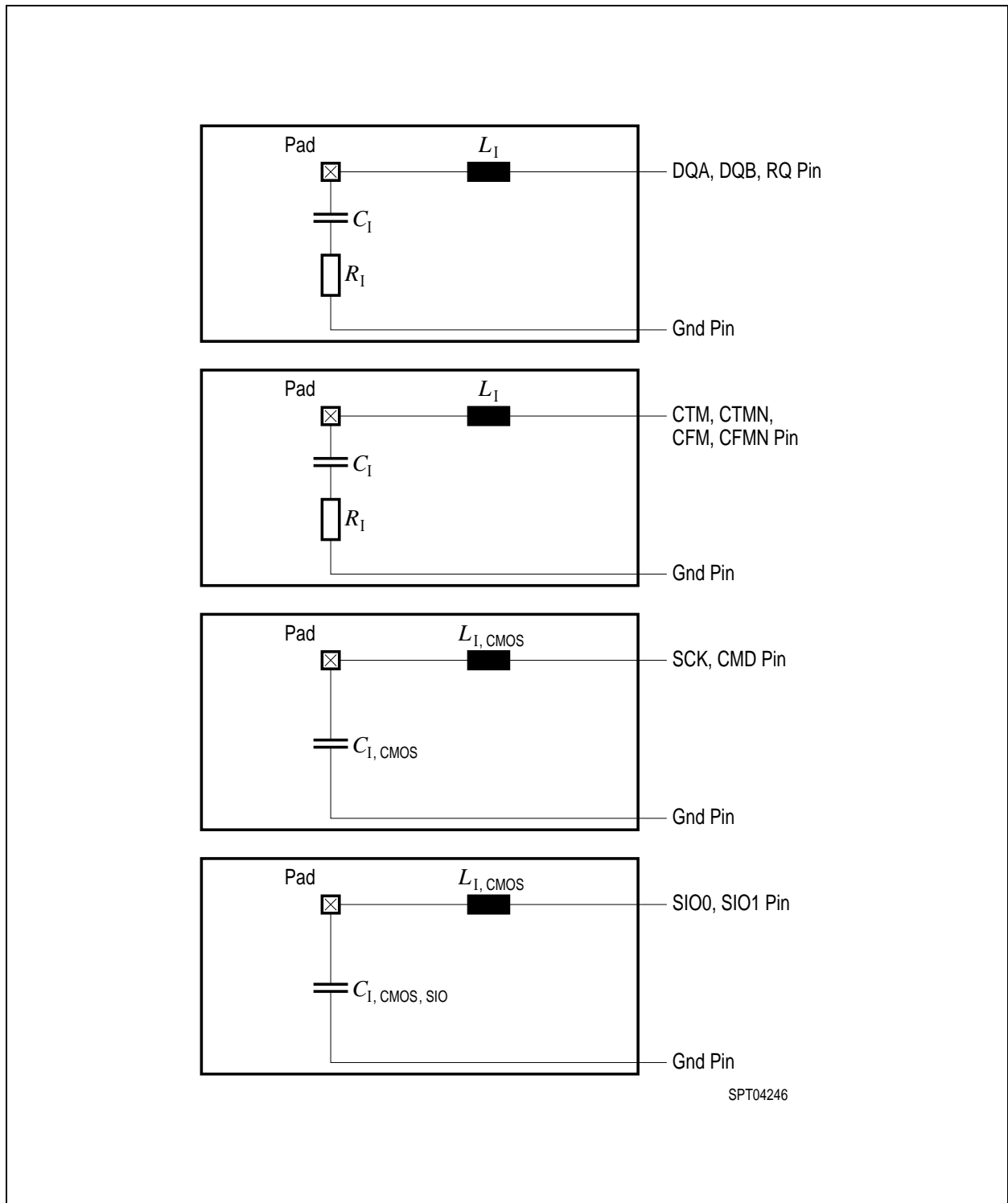
<sup>1)</sup> The CMOS interface consumes power in all power states.

<sup>2)</sup> x16/x18 RDRAM data width.

<sup>3)</sup> This does not include the  $I_{OL}$  sink current. The RDRAM dissipates  $I_{OL} \times V_{OL}$  in each output driver when a logic one is driven.

**Capacitance and Inductance**

Figure 60 shows the equivalent load circuit of the RSL and CMOS pins. The circuit models the load that the device presents to the Channel.



**Figure 60 Equivalent Load Circuit for RSL Pins**

This circuit does not include pin coupling effects that are often present in the packaged device. Because coupling effects make the effective single-pin inductance  $L_1$ , and capacitance  $C_1$ , a function of neighboring pins, these parameters are intrinsically data-dependent. For purposes of specifying the device electrical loading on the Channel, the effective  $L_1$  and  $C_1$  are defined as the worst-case values over all specified operating conditions.

$L_1$  is defined as the effective pin inductance based on the device pin assignment. Because the pad assignment places each RSL signal adjacent to an AC ground (a GND or  $V_{DD}$  pin), the effective inductance must be defined based on this configuration. Therefore,  $L_1$  assumes a loop with the RSL pin adjacent to an AC ground.

$C_1$  is defined as the effective pin capacitance based on the device pin assignment. It is the sum of the effective package pin capacitance and the IO pad capacitance.

**Table 26 RSL Pin Parasitics**

Parameter and Conditions - RSL Pins	Symbol	Limit Values		Unit
		min.	max.	
RSL effective input inductance	$L_1$	–	4.0	nH
Mutual inductance between any DQA or DQB RSL signals.	$L_{12}$	–	0.2	nH
Mutual inductance between any ROW or COL RSL signals.		–	0.6	nH
Difference in $L_1$ value between any RSL pins of a single device.	$\Delta L_1$	–	1.8	nH
RSL effective input capacitance <sup>1)</sup> -800	$C_1$	2.0	2.4	pF
RSL effective input capacitance <sup>1)</sup> -711		2.0	2.4	pF
RSL effective input capacitance <sup>1)</sup> -600		2.0	2.6	pF
Mutual capacitance between any RSL signals.	$C_{12}$	–	0.1	pF
Difference in $C_1$ value between average of CTM/CFM and any RSL pins of a single device.	$\Delta C_1$	–	0.06	pF
RSL effective input resistance	$R_1$	4	15	$\Omega$

<sup>1)</sup> This value is a combination of the device IO circuitry and package capacitances.

**Table 27 CMOS Pin Parasitics**

Parameter and Conditions - CMOS Pins	Symbol	Limit Values		Unit
		min.	max.	
CMOS effective input inductance	$L_{1,CMOS}$	–	8.0	nH
CMOS effective input capacitance (SCK,CMD) <sup>1)</sup>	$C_{1,CMOS}$	1.7	2.1	pF
CMOS effective input capacitance (SIO1, SIO0) <sup>1)</sup>	$C_{1,CMOS,SIO}$	–	7.0	pF

<sup>1)</sup> This value is a combination of the device IO circuitry and package capacitances.

### Center-Bonded FBGA Package

Figure 61 shows the form and dimensions of the recommended package for the center-bonded CSP device class.

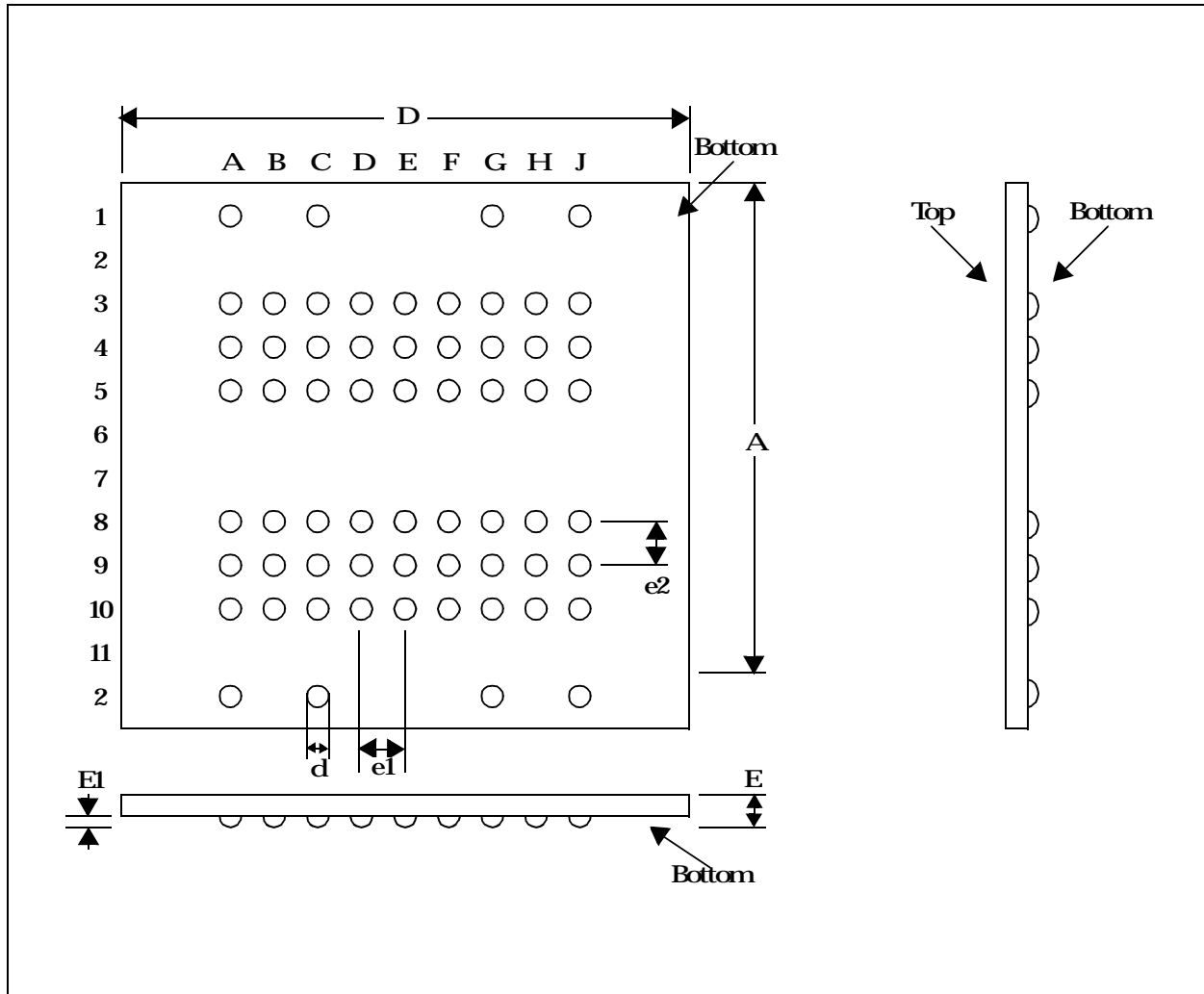


Figure 61 Center-Bonded FBGA Package

Table 28 lists the numerical values corresponding to dimensions shown in Figure 61.

**Table 28 Center-Bonded FBGA Package Dimensions**

Parameter	Symbol	Limit Values		Unit
		min.	max.	
Ball pitch (x-axis)	e1	1.00	1.00	mm
Ball pitch (y-axis)	e2	0.8	0.8	mm
Package body length	A	10.9	11.1	mm
Package body width	D	10.4	10.6	mm
Package total thickness	E	0.65	1.05	mm
Ball height	E1	0.18	0.35	mm
Ball diameter	d	0.3	0.4	mm

## Glossary of Terms

<b>ACT</b>	Activate command from AV field.
<b>activate</b>	To access a row and place in sense amp.
<b>adjacent</b>	Two RDRAM banks which share sense amps (also called doubled banks).
<b>ASYM</b>	CCA register field for RSL $V_{OL}/V_{OH}$ .
<b>ATTN</b>	Power state - ready for ROW/COL packets.
<b>ATTNR</b>	Power state - transmitting Q packets.
<b>ATTNW</b>	Power state - receiving D packets.
<b>AV</b>	Opcode field in ROW packets.
<b>bank</b>	A block of $2^{RBIT} \times 2^{CBIT}$ storage cells in the core of the RDRAM.
<b>BC</b>	Bank address field in COLC packet.
<b>BBIT</b>	CNFGA register field - # bank address bits.
<b>broadcast</b>	An operation executed by all RDRAMs.
<b>BR</b>	Bank address field in ROW packets.
<b>bubble</b>	Idle cycle(s) on RDRAM pins needed because of a resource constraint.
<b>BYT</b>	CNFGB register field - 8/9 bits per byte.
<b>BX</b>	Bank address field in COLX packet.
<b>C</b>	Column address field in COLC packet.
<b>CAL</b>	Calibrate ( $I_{OL}$ ) command in XOP field.
<b>CBIT</b>	CNFGB register field - # column address bits.
<b>CCA</b>	Control register - current control A.
<b>CCB</b>	Control register - current control B.
<b>CFM,CFMN</b>	Clock pins for receiving packets.
<b>Channel</b>	ROW/COL/DQ pins and external wires.
<b>CLRR</b>	Clear reset command from SOP field.
<b>CMD</b>	CMOS pin for initialization/power control.
<b>CNFGA</b>	Control register with configuration fields.
<b>CNFGB</b>	Control register with configuration fields.
<b>COL</b>	Pins for column-access control.
<b>COL</b>	COLC,COLM,COLX packet on COL pins.
<b>COLC</b>	Column operation packet on COL pins.
<b>COLM</b>	Write mask packet on COL pins.
<b>column</b>	Rows in a bank or activated row in sense amps have $2^{CBIT}$ dualocts column storage.
<b>command</b>	A decoded bit-combination from a field.
<b>COLX</b>	Extended operation packet on COL pins.



<b>controller</b>	A logic-device which drives the ROW/COL /DQ wires for a Channel of RDRAMs.
<b>COP</b>	Column opcode field in COLC packet.
<b>core</b>	The banks and sense amps of an RDRAM.
<b>CTM,CTMN</b>	Clock pins for transmitting packets.
<b>current control</b>	Periodic operations to update the proper $I_{OL}$ value of RSL output drivers.
<b>D</b>	Write data packet on DQ pins.
<b>DBL</b>	CNFGB register field - doubled-bank.
<b>DC</b>	Device address field in COLC packet.
<b>device</b>	An RDRAM on a Channel.
<b>DEVID</b>	Control register with device address that is matched against DR, DC, and DX fields.
<b>DM</b>	Device match for ROW packet decode.
<b>doubled-bank</b>	RDRAM with shared sense amp.
<b>DQ</b>	DQA and DQB pins.
<b>DQA</b>	Pins for data byte A.
<b>DQB</b>	Pins for data byte B.
<b>DQS</b>	NAPX register field - PDN/NAP exit.
<b>DR,DR4T,DR4F</b>	Device address field and packet framing fields in ROWA and ROWR packets.
<b>dualoct</b>	16 bytes - the smallest addressable datum.
<b>DX</b>	Device address field in COLX packet.
<b>field</b>	A collection of bits in a packet.
<b>INIT</b>	Control register with initialization fields.
<b>initialization</b>	Configuring a Channel of RDRAMs so they are ready to respond to transactions.
<b>LSR</b>	CNFGA register field - low-power self-refresh.
<b>M</b>	Mask opcode field (COLM/COLX packet).
<b>MA</b>	Field in COLM packet for masking byte A.
<b>MB</b>	Field in COLM packet for masking byte B.
<b>MSK</b>	Mask command in M field.
<b>MVER</b>	Control register - manufacturer ID.
<b>NAP</b>	Power state - needs SCK/CMD wakeup.
<b>NAPR</b>	Nap command in ROP field.
<b>NAPRC</b>	Conditional nap command in ROP field.
<b>NAPXA</b>	NAPX register field - NAP exit delay A.
<b>NAPXB</b>	NAPX register field - NAP exit delay B.
<b>NOCOP</b>	No-operation command in COP field.

<b>NOROP</b>	No-operation command in ROP field.
<b>NOXOP</b>	No-operation command in XOP field.
<b>NSR</b>	INIT register field- NAP self-refresh.
<b>packet</b>	A collection of bits carried on the Channel.
<b>PDN</b>	Power state - needs SCK/CMD wakeup.
<b>PDNR</b>	Powerdown command in ROP field.
<b>PDNXA</b>	Control register - PDN exit delay A.
<b>PDNXB</b>	Control register - PDN exit delay B.
<b>pin efficiency</b>	The fraction of non-idle cycles on a pin.
<b>PRE</b>	PREC,PRER,PREX precharge commands.
<b>PREC</b>	Precharge command in COP field.
<b>precharge</b>	Prepares sense amp and bank for activate.
<b>PRER</b>	Precharge command in ROP field.
<b>PREX</b>	Precharge command in XOP field.
<b>PSX</b>	INIT register field - PDN/NAP exit.
<b>PSR</b>	INIT register field - PDN self-refresh.
<b>PVER</b>	CNFGB register field - protocol version.
<b>Q</b>	Read data packet on DQ pins.
<b>R</b>	Row address field of ROWA packet.
<b>RBIT</b>	CNFGB register field - # row address bits.
<b>RD/RDA</b>	Read (/precharge) command in COP field.
<b>read</b>	Operation of accessing sense amp data.
<b>receive</b>	Moving information from the Channel into the RDRAM (a serial stream is demuxed).
<b>REFA</b>	Refresh-activate command in ROP field.
<b>REFB</b>	Control register - next bank (self-refresh).
<b>REFBIT</b>	CNFGA register field - ignore bank bits (for REFA and self-refresh).
<b>REFP</b>	Refresh-precharge command in ROP field.
<b>REFR</b>	Control register - next row for REFA.
<b>refresh</b>	Periodic operations to restore storage cells.
<b>retire</b>	The automatic operation that stores write buffer into sense amp after WR command.
<b>RLX</b>	RLXC,RLXR,RLXX relax commands.
<b>RLXC</b>	Relax command in COP field.
<b>RLXR</b>	Relax command in ROP field.
<b>RLXX</b>	Relax command in XOP field.
<b>ROP</b>	Row-opcode field in ROWR packet.

<b>row</b>	$2^{\text{CBIT}}$ dualocts of cells (bank/sense amp).
<b>ROW</b>	Pins for row-access control
<b>ROW</b>	ROWA or ROWR packets on ROW pins.
<b>ROWA</b>	Activate packet on ROW pins.
<b>ROWR</b>	Row operation packet on ROW pins.
<b>RQ</b>	Alternate name for ROW/COL pins.
<b>RSL</b>	Rambus Signaling Levels.
<b>SAM</b>	Sample ( $I_{OL}$ ) command in XOP field.
<b>SA</b>	Serial address packet for control register transactions w/ SA address field.
<b>SBC</b>	Serial broadcast field in SRQ.
<b>SCK</b>	CMOS clock pin.
<b>SD</b>	Serial data packet for control register transactions w/ SD data field.
<b>SDEV</b>	Serial device address in SRQ packet.
<b>SDEVID</b>	INIT register field - Serial device ID.
<b>self-refresh</b>	Refresh mode for PDN and NAP.
<b>sense amp</b>	Fast storage that holds copy of bank's row.
<b>SETF</b>	Set fast clock command from SOP field.
<b>SETR</b>	Set reset command from SOP field.
<b>SINT</b>	Serial interval packet for control register read/write transactions.
<b>SIO0,SIO1</b>	CMOS serial pins for control registers.
<b>SOP</b>	Serial opcode field in SRQ.
<b>SRD</b>	Serial read opcode command from SOP.
<b>SRP</b>	INIT register field - Serial repeat bit.
<b>SRQ</b>	Serial request packet for control register read/write transactions.
<b>STBY</b>	Power state - ready for ROW packets.
<b>SVER</b>	Control register - stepping version.
<b>SWR</b>	Serial write opcode command from SOP.
<b>TCAS</b>	TCLSCAS register field - $t_{\text{CAS}}$ core delay.
<b>TCLS</b>	TCLSCAS register field - $t_{\text{CLS}}$ core delay.
<b>TCLSCAS</b>	Control register - $t_{\text{CAS}}$ and $t_{\text{CLS}}$ delays.
<b>TCYCLE</b>	Control register - $t_{\text{CYCLE}}$ delay.
<b>TDAC</b>	Control register - $t_{\text{DAC}}$ delay.
<b>TEST77</b>	Control register - for test purposes.
<b>TEST78</b>	Control register - for test purposes.
<b>TRDLY</b>	Control register - $t_{\text{RDLY}}$ delay.
<b>transaction</b>	ROW,COL,DQ packets for memory access.

<b>transmit</b>	Moving information from the RDRAM onto the Channel (parallel word is muxed).
<b>WR/WRA</b>	Write (/precharge) command in COP field.
<b>write</b>	Operation of modifying sense amp data.
<b>XOP</b>	Extended opcode field in COLX packet



**Direct RDRAM**  
**128/144-MBit (256K×16/18×32s)**

---