# 16Gb GDDR6 SGRAM

## *Lead-Free&Halogen-Free*

## *(RoHS Compliant)*

## H56CBM24MIR - xxC

* SK hynix reserves the right to change products or specifications without notice.

## Revision History

| Revision No. | History | Date | Remark |
|---|---|---|---|
| 0.0 | Preliminary version release | May. 2018 | Preliminary |
| 1.0 | IDD & AC timing | Dec. 2018 | |
| 1.1 | IDD7 pattern & IDD Update | Mar. 2019 | |
| 1.2 | DCA inserted | Apr. 2019 | |
| 1.3 | Absolute VDD & tHSRF updte / 1.25V added | Sep. 2019 | |
| 1.4 | Temperature Readout during tRFCab added | Oct. 2019 | |
| 1.5 | IDD spec updated | Mar. 2020 | |
| 1.6 | Package Thermal Characteristics updated | Apr. 2020 | |

# 1. Description

The GDDR6 SGRAM is a high-speed dynamic random-access memory designed for applications requiring high bandwidth. GDDR6 devices contain the following number of bits:

16 Gb has 17,179,869,184 bits

The GDDR6 SGRAM's high-speed interface is optimized for point-to-point connections to a host controller. On-die termination (ODT) is provided for all high-speed interface signals to eliminate the need for termination resistors in the system.

GDDR6 uses a 16n prefetch architecture and a DDR interface to achieve high-speed operation. The device's architecture consists of two 16 bit wide fully independent channels.

GDDR6 operates from a differential clock CK_t and CK_c. CK is common to both channels. Command and Address (CA) are registered at every rising edge of CK and every falling edge of CK. There are both single cycle and multi cycle commands. See command truth table for details.

GDDR6 uses a free running differential forwarded clock (WCK_t/WCK_c) with both input and output data registered and driven respectively at both edges of the forwarded WCK. See Clocking section for details.

Read and write accesses to GDDR6 are burst oriented; accesses start at a selected location and consists of a total of sixteen data words. Accesses begin with the registration of an Activate command, which is then followed by a Read, Write (WOM) or masked Write (WDM, WSM) command.

The row and bank address to be accessed is registered coincident with the Activate command. The address bits registered coincident with the Read, Write or masked Write command are used to select the bank and the starting column location for the burst access.

This specification includes all features and functionality required for GDDR6 SGRAM devices. In many cases the GDDR6 specification describes the behavior of a single channel.

# 2. Device Features and Ordering Information

## 2.1. FEATURES

- 2 separate independent channels with point-to-point interface for data, address and command
- Half CA data rate differential clock inputs CK_t/CK_c for CMD/ ADD (CA) per 2 channels
- Four half data rate differential clock inputs WCK_t/WCK_c, each associated with a data byte (DQ, DBI_n, EDC) in the channel
- Double Data Rate (DDR) data (with regards to the WCK)
- Double Data Rate (DDR) Command Address (with regards to the CK)
- 16 internal banks
- 4 bank groups for $t_{CCDL}$ = 3 $t_{CK}$ and 4 $t_{CK}$
- 16n prefetch architecture: 256 bit per array read or write access per channel
- Burst length: 16 only
- Programmable READ latency: 9 to 27 $t_{CK}$
- Programmable WRITE latency: 5 to 7 $t_{CK}$
- WRITE Data mask function via CA bus (single/double byte mask)
- Data bus inversion (DBI) & Command Address bus inversion (CABI)
- Command Address training: command address input monitoring by DQ/DBI_n/EDC signals
- WCK2CK clock training with phase information by EDC signals
- Data read and write training via READ FIFO (depth 6)
- READ FIFO pattern preload by LDFF command
- Direct write data load to READ FIFO by WRTR command
- Consecutive read of READ FIFO by RDTR command
- Read/Write data transmission integrity secured by cyclic redundancy check using either a half or full data rate CRC
- READ/WRITE EDC on/off mode
- Programmable EDC hold pattern for CDR
- Programmable CRC READ latency = 1 to 4 $t_{CK}$ and CRC WRITE latency = 10 to 14 $t_{CK}$
- Low Power modes
- On-chip temperature sensor with read-out
- Auto precharge option for each burst access
- Auto refresh & self refresh modes
- 32ms, auto refresh (16k cycles)
- Temperature sensor controlled self refresh rate and Partial Array Self Refresh
- Per-Bank / Per-2-Bank Refresh
- Optional digital $t_{RAS}$ lockout
- On-die termination (ODT)
- ODT and output driver strength auto-calibration with external resistor ZQ
- Programmable termination and driver strength offsets (40 ohm to 60ohm)
- Internal $V_{REF}$ for data inputs and CA inputs with programmable levels
- Separate internal $V_{REF}$ for CA (Command / Address) inputs
- Vendor ID1 and ID2 for identification
- x16/x8 mode configuration set at power-up with EDC
- Pseudo-channel mode (PC mode) configuration set at power up with CA6
- 1.35V +/- 0.0405V supply for device operation ($V_{DD}$)
- 1.35V +/- 0.0405V supply for I/O interface ($V_{DDQ}$)
- 1.8 + 0.108V / - 0.054V supply for $V_{PP}$
- 180 ball BGA package with 0.75mm pitch

## 2.2. ORDERING INFORMATION

| Part No | Power Supply VPP/VDD/VDDQ | WCK Frequency | Max Data Rate | Interface |
|---|---|---|---|---|
| TBD | TBD | TBD | TBD | TBD |
| | TBD | TBD | TBD | TBD |
| H56CBM24MIR-S2C | 1.8 / 1.35 / 1.35 | 7.0GHz | 14Gbps/pin | POD_135 |
| | 1.8 / 1.25 / 1.25 | 6.0GHz | 12Gbps/pin | POD_125 |

**Note1) Above SKhynix P/N's are Lead-free, RoHS Compliant and Halogen-free**

## 2.3. DEFINITION OF SIGNAL STATE TERMINOLOGY

GDDR6 SGRAM will be operated in both ODT Enable (terminated) and ODT Disable (unterminated) modes. For highest data rates it is recommended to operate in the ODT Enable mode. ODT Disable mode is designed to reduce power and may operate at reduced data rates. There exist situations where ODT Enable mode can not be guaranteed for a short period of time, i.e., during power up.

Following are four terminologies defined for the state of a device (GDDR6 SGRAM or controller) signal during operation. The state of the bus will be determined by the combination of the device signal connected to the bus in the system. For example, in GDDR6 it is possible for the SGRAM pin to be tristated while the controller signal is HIGH or ODT. In both cases the bus would be HIGH if the ODT is enabled. For details on the device's signals and their function see Sections 9.1 and 9.2.

Device pin signal level:

- High: A device pin is driving the Logic "1" state.
- Low: A device pin is driving the Logic "0" state.
- Hi-Z: A device pin is tristate.
- ODT: A device pin terminates with ODT setting, which could be terminating or tristate depending on Mode Register setting.

Bus signal level:

- High: One device on bus is High and all other devices on bus are either ODT or Hi-Z. The voltage level on the bus would be nominally VDDQ
- Low: One device on bus is Low and all other devices on bus are either ODT or Hi-Z. The voltage level on the bus would be nominally VOL(DC) if ODT was enabled, or VSSQ if Hi-Z.
- Hi-Z: All devices on bus are Hi-Z. The voltage level on bus is undefined as the bus is floating.
- ODT: At least one device on bus is ODT and all others are Hi-Z. The voltage level on the bus would be nominally VDDQ.

## 2.4. DEFINITION OF CLOCKING TERMINOLOGY

- Data refers to the signal being clocked (e.g. DQ by WCK and CA by CK
- Half rate: clock is running at half of the data rate (e.g. WCK 4GHz and DQ at 8Gbps, or CK 1GHz and CA at 2Gbps)
- Quarter rate: clock is running at a quarter of the data rate (e.g. WCK 2GHz and DQ at 8Gbps)
- Eighth rate: clock is running at one eighth of the data rate (e.g. WCK internal 1GHz and DQ at 8Gbps)
- DDR (Double Data Rate): complement to half rate, referring to data relative to clock

## 2.5. CLOCKING

The GDDR6 SGRAM supports DDR operating mode for WCK frequency

Figure 1 illustrates a DDR WCK. Figure 60 illustrates a WRITE command with a DDR WCK clock. Figure 75 illustrates a READ command with DDR WCK clocking. Other figures in the specification are shown with the DDR WCK unless otherwise noted.

GDDR6 SGRAM devices can be designed with a WCK/byte. The ball-out has provisions for a WCK/byte.

The DRAM info bits for WCK Granularity, WCK Frequency and Internal WCK can be read by the host during the initialization process to determine the WCK architecture for the device and for devices that support multiple frequencies, MR2 OP11 allows for the mode to be set. For the frequencies for each mode see Table 69.

In DDR mode the GDDR6 device operates from a differential clock CK_t and CK_c. Command and Address (CA) are registered at every rising and falling CK edge. For WCK DDR ratio the GDDR6 device can support either a full data rate EDC or a half data rate EDC. See EDC section for more details.

A rising CK (or WCK) edge is defined as the crossing of the positive edge of CK_t (or WCK_t) and the negative edge of CK_c (or WCK_c). A falling CK (or WCK) edge is defined as the crossing of the negative edge of CK_t (or WCK_t) and the positive edge of CK_c (or WCK_c).

**Table 1 — Example Clock and Interface Signal Frequency Relationship**

| PIN | DDR WCK | | UNIT |
|---|---|---|---|
| CK_t, CK_c | 1.5 | | GHz |
| CA | 3.0 | | Gbps/pin |
| WCK_t, WCK_c | 6.0 | | GHz |
| DQ, DBI_n | 12.0 | | Gbps/pin |
| EDC | 6.0 | 12.0 | Gbps/pin |

## 2.5 CLOCKING (cont'd)

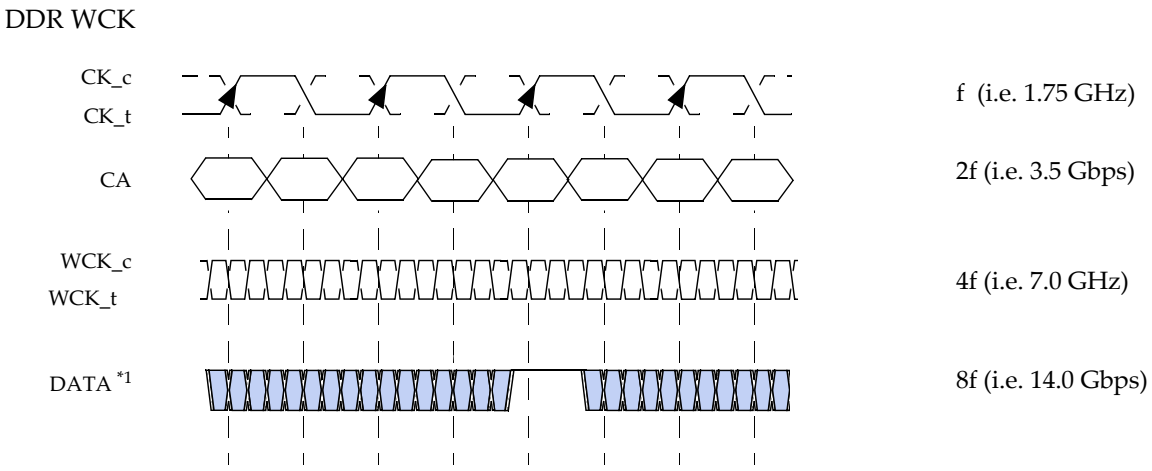



**Figure 1 — GDDR6 Clocking and Interface Relationship**

## 2.5 CLOCKING (cont'd)

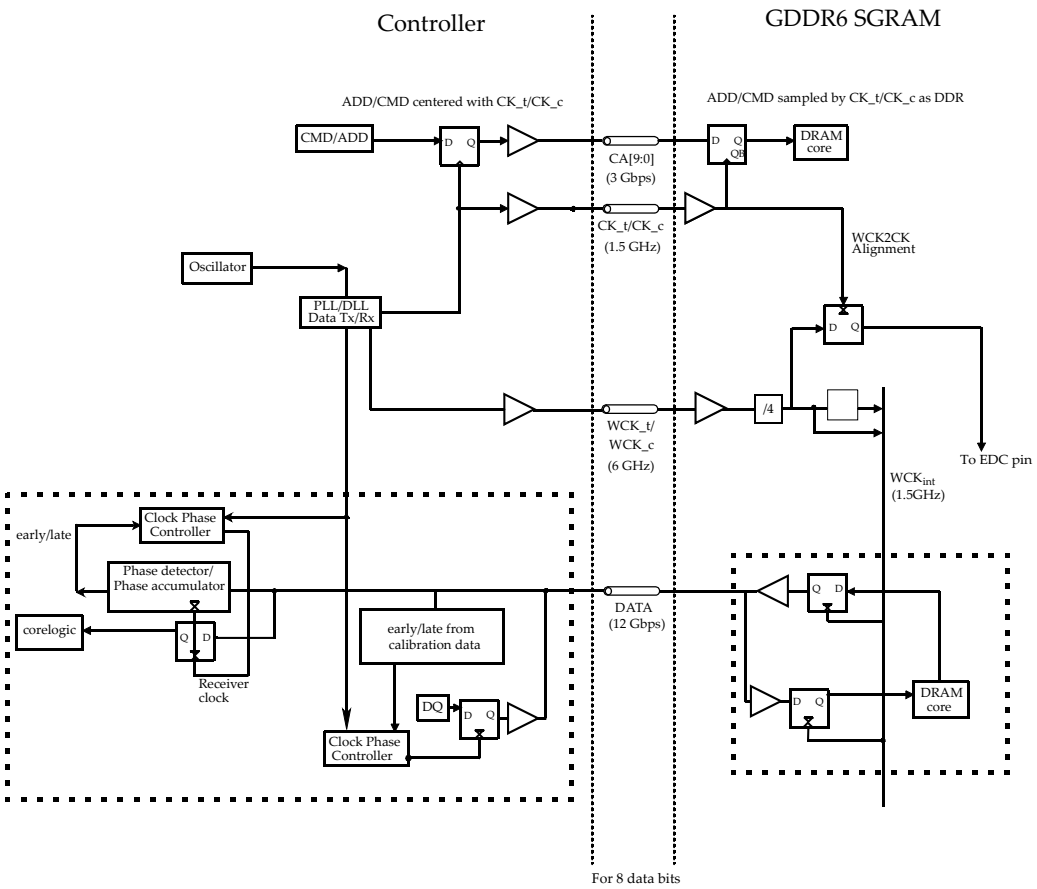



**Figure 2 — Block Diagram of an example clock system**

# 3. INITIALIZATION

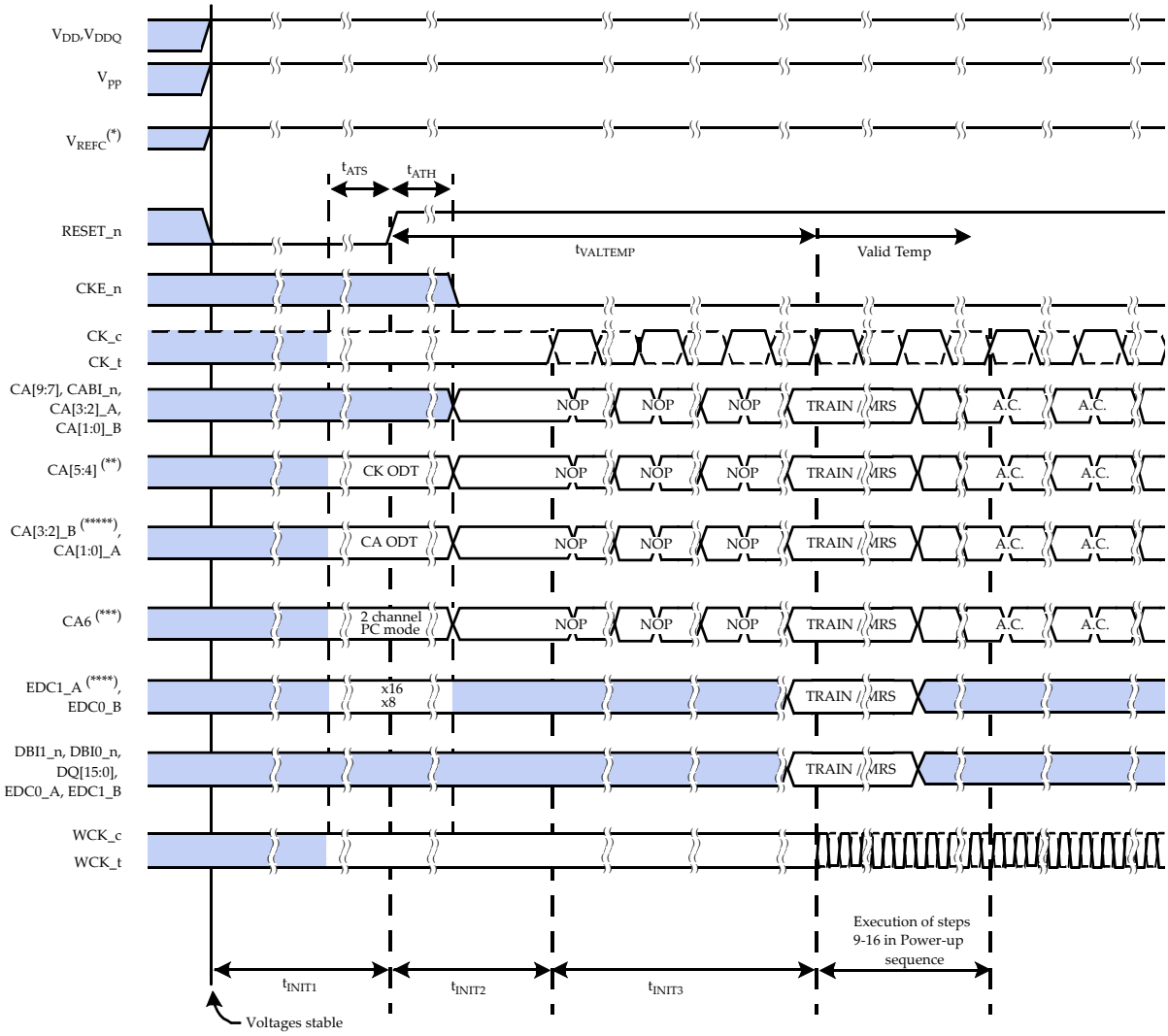
## 3.1. POWER-UP SEQUENCE

GDDR6 SGRAMs must be powered up and initialized in a predefined manner. Operational procedures other than those specified may result in undefined operation. The Mode Registers do not have RESET default values, except for CABI, CA termination, and the EDC hold pattern. If the mode registers are not set during the initialization sequence, it may lead to unspecified operation.

1) Apply power to VPP. Apply power to VDD at the same time or after power is applied to VPP. Apply power to VDDQ at the same time or after power is applied to VDD. VPP must be equal to or higher than VDD at all times the device is powered up except during power-off sequence.Apply $V_{REFC}$ at same time or after power is applied to $V_{DDQ}$, or pull $V_{REFC}$ LOW to select internal $V_{REFC}$.
2) The voltage levels on all signal balls must be less than or equal to $V_{DD}$ and $V_{DDQ}$ on one side and must be larger than or equal to $V_{SS}$ on the other side.
3) Assert RESET_n LOW to ensure all drivers are in Hi-Z and all ODT are off. Maintain RESET_n LOW for a minimum time of $t_{INIT1}$.
4) Set system configuration info at least a time $t_{ATS}$ before RESET_n is driven HIGH:
    a. Drive CA[5:4]_A and CA[5:4]_B both LOW or HIGH for the desired CK ODT
    b. Drive EDC1_A and EDC0_B both HIGH for x16 mode or LOW for x8 mode
    c. Drive CA6_A and CA6_B both HIGH for 2 channel mode or LOW for PC mode
    d. Drive CA1_A and CA0_A per Table 3 to select CA bus termination strength for Channel A.
    e. Drive CA3_B and CA2_B per Table 3 to select CA bus termination strength for Channel B
    d. Drive CK_t, WCK_t to static LOW level, and CK_c, WCK_c to static HIGH level.

    After meeting $t_{ATS}$ requirement, drive RESET_n HIGH
5) After RESET_n is pulled HIGH, maintain CA[6:0] and EDC signals stable for a minimum time of $t_{ATH}$.
6) Pull CKE_n LOW after $t_{ATH}$ is satisfied. Assert and hold NOP command. The device performs the initial impedance calibration during this time; this will be done without external clocks. Latest after $t_{INIT2}$ the device enables the CK and CA ODT as determined in steps 5 and 6.
7) Provide a stable CK clock for a minimum of $t_{INIT3}$ cycles.
8) Issue MRS command to MR15 to set the device into CA training mode (optional).
9) Complete CA training (optional).
10) Issue MRS command to read the Vendor ID (optional).
11) Issue MRS command to set WCK termination values.
12) Provide stable WCK clocks.
13) Issue MRS commands to the mode registers in any order. Issue MRS commands, to select the position of a WCK2CK alignment point and WCK ratio, and to set WLmrs, RLmrs, CRCWL, and CRCRL to appropriate values. All these features must be programmed before WCK2CK training, and $t_{MOD}$ must be met during this procedure.
14) Issue two REFab commands followed by NOP until $t_{RFCab}$ is satisfied.
15) After any necessary training sequences such as WCK2CK training, READ training (LDFF, RDTR) and WRITE training (WRTR, RDTR), the device is ready for normal operation.

## 3.1 POWER-UP SEQUENCE (cont'd)




Notes:

A.C. = Any Command

(*) Pull $V_{REFC}$ ball to $V_{SS}$ to select internal $V_{REFC}$

(**) Drive CA[5:4] HIGH or LOW to select default CK ODT

(***) Drive CA6 HIGH or LOW to select either 2 channel or PC mode

(****) Drive EDC HIGH or LOW to select x8 or x16 mode

(*****) Drive CA[3:2]_B and CA[1:0]_A HIGH or LOW to select default CA ODT

**Figure 3 — Power-up Initialization**

## 3.1 POWER-UP SEQUENCE (cont'd)

**Table 2 — Device Initialization Timings**

| PARAMETER | SYMBOL | VALUES | | UNIT | NOTES |
|---|---|---|---|---|---|
| | | MIN | MAX | | |
| RESET_n initial LOW time after power-up | $t_{INIT1}$ | 200 | – | µs | |
| Time after RESET_n de-assertion before starting the CK clock | $t_{INIT2}$ | 1 | – | ms | |
| Stable CK clock cycles before issuing valid commands | $t_{INIT3}$ | 100 | – | $t_{CK}$ | |
| CA[6:0] and EDC setup time before RESET_n de-assertion | $t_{ATS}$ | 10 | – | ns | |
| CA[6:0] and EDC hold time after RESET_n de-assertion | $t_{ATH}$ | 10 | – | ns | |
| RESET_n LOW time with stable power | $t_{RES}$ | 100 | – | ns | |
| Temperature sensing time after RESET_n deassertion | $t_{VALTEMP}$ | 1.5 | | ms | |

**Table 3 — CA Termination**

| CA[3:2]_B or CA[1:0]_A at RESET_n high transition | Termination for CA[3:0] | Termination with CA6 LOW for CA[9:4], CABI_n & CKE_n | Termination with CA6 HIGH for CA[9:4], CABI_n & CKE_n |
|---|---|---|---|
| LOW, LOW | Disabled | Disabled | Disabled |
| LOW, HIGH | 60 Ohms | 120 Ohms | 60 Ohms |
| HIGH, LOW | 120 Ohms | 240 Ohms | 120 Ohms |
| HIGH, HIGH | Reserved | Reserved | Reserved |

**Table 4 — CK Termination**

| CA5_A CA5_B | CA4_A CA4_B | Default CK Termination Value |
|---|---|---|
| LOW | X | Disabled |
| HIGH | LOW | 60 Ohm |
| HIGH | HIGH | 120 Ohm |

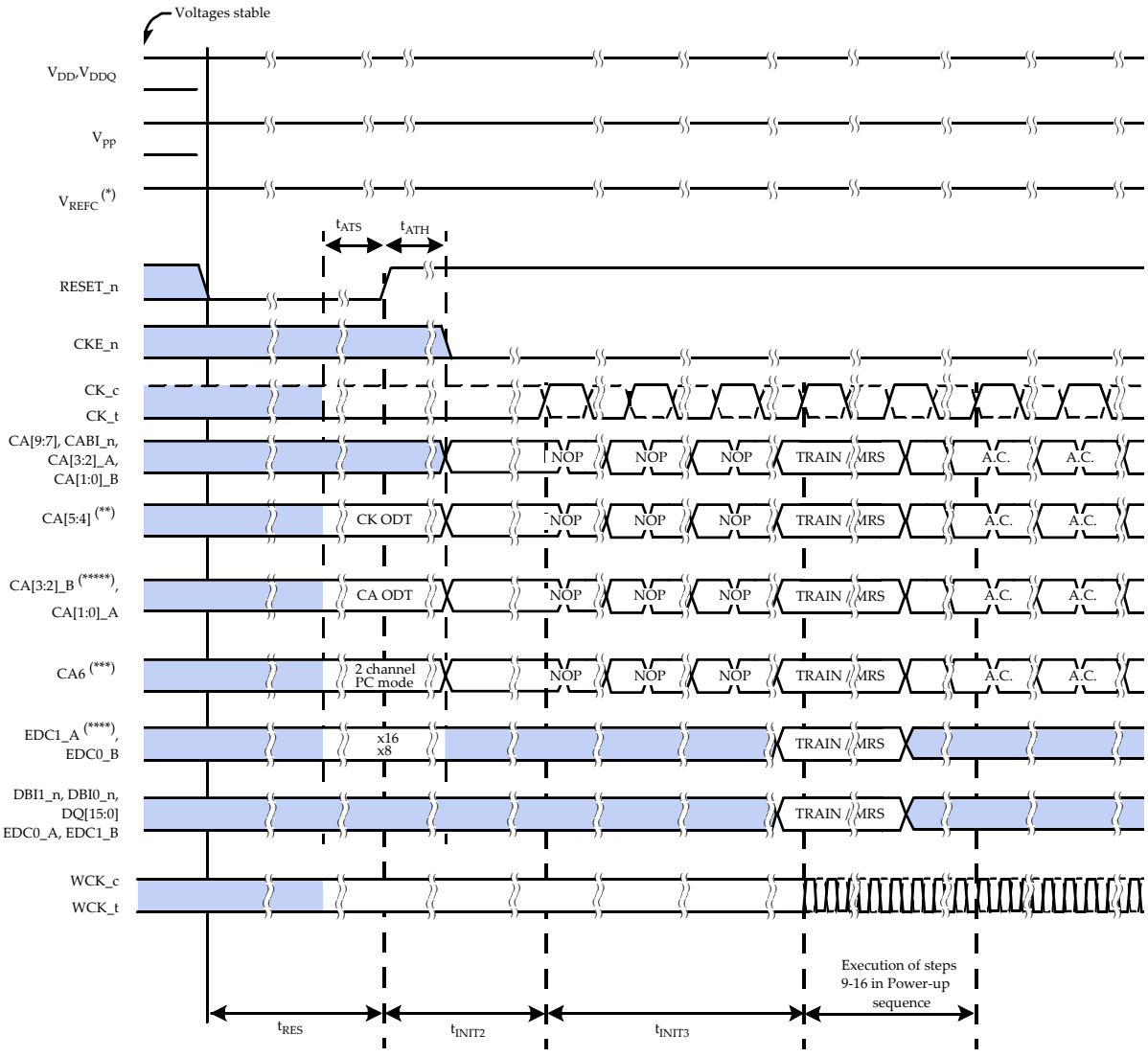
## 3.2 INITIALIZATION WITH STABLE POWER (cont'd)

The following sequence is required for reset subsequent to power-up initialization. This requires that the power has been stable within the specified $V_{DD}$ and $V_{DDQ}$ ranges since power-up initialization (see Figure 4):

1) Assert RESET_n LOW anytime when reset is needed.

2) Hold RESET_n LOW for minimum $t_{RES}$. Assert and hold NOP command.

3) Continue with step 5 of the power-up initialization sequence.




Notes:
A.C. = Any Command
(*) Pull $V_{REFC}$ ball to $V_{SS}$ to select internal $V_{REFC}$
(**) Drive CA[5:4] HIGH or LOW to select default CK ODT
(***) Drive CA6 HIGH or LOW to select either 2 channel or mode
(****) Drive EDC HIGH or LOW to select x8 or x16 mode
(*****) Drive CA[3:2]_B and CA[1:0]_A HIGH or LOW to select default CA ODT

**Figure 4 — Initialization with Stable Power**

## 3.3 VENDOR ID (cont'd)

GDDR6 SGRAMs include a Vendor ID feature that allows the controller to receive information from the device to differentiate between different vendors and different devices using a software algorithm.

Vendor ID is part of the INFO field of Mode Register 3 (MR3) and is selected by issuing a MODE REGISTER SET command with MR3 OP[7:6] = 01 for ID1 and MR3 OP[7:6] = 11 for ID2. MR3 OP[11:8] and OP[5:0] are set to the desired values.

When the Vendor ID function is enabled the device information will be transmitted on the DQ bus as shown in Figure 5. The device will provide the Vendor ID as shown in Table 5 or Table 6 depending on whether ID1 or ID2 is selected. For x8 mode the 16 bits of Vendor ID information comes from 2 devices as only 1 data byte per channel per device is enabled.

Additional information can optionally be provided by the vendor using a vendor specific setting in the Reserved for Vendor Specific Features MR and will follow the same protocol as vendor ID unless explicitly stated in the vendor data sheet.

The Vendor ID will be driven onto the DQ bus after the MRS command that sets bits OP6 to 1 and OP7 to 0 for ID1 or to 1 for ID2. The DQ bus will be continuously driven until an MRS command sets MR3 OP6 and OP7 back to 0. The DQ bus will be in ODT state after $t_{WRIDOFF}$(max). The code can be sampled by the controller after waiting $t_{WRIDON}$(max) and before $t_{WRIDOFF}$(min). DBI is not enabled or ignored during all Vendor ID operations.

The EDC hold pattern is continuously driven on the EDC signals provided a stable WCK clock is applied.

**Table 5 — Vendor ID to DQ Mapping - ID1 (MR3 OP[7:6] = 01)**

| DQ15_A<br>DQ15_B | DQ[14:13]_A<br>DQ[14:13]_B | DQ12_A<br>DQ12_B | DQ11_A<br>DQ11_B | DQ[10:8]_A<br>DQ[10:8]_B | DQ[7:4]_A<br>DQ[7:4]_B | DQ[3:0]_A<br>DQ[3:0]_B |
|---|---|---|---|---|---|---|
| VDDQ OFF<br>See Table 12 | WCK Frequency<br>see Table 11 | WCK Granularity<br>see Table 10 | Internal WCK<br>see Table 8 | Density<br>see Table 9 | Revision ID | Manufacturers Vendor Code<br>see Table 7 |

The Manufacturers Vendor Code, Revision ID and Density allow the host to differentiate which vendor's device, the revision of the device and the density used in the system. Internal WCK tells the host whether to use an invert or quad shift when programming MR10 OP[7:4]. WCK Granularity and WCK Frequency are used by the host to determine if the device in the has WCK/byte or WCK/word to allow the device to be properly trained and configured.

**Table 6 — Vendor ID to DQ Mapping - ID2 (MR3 OP[7:6] = 11)**

| DQ[15:2]_A<br>DQ[15:2]_B | DQ1_A<br>DQ1_B | DQ0_A<br>DQ0_B |
|---|---|---|
| RFU<br>("1") | PRBS<br>See Table 14 | Programmable P2BR<br>See Table 13 |

Programmable P2BR allows the host to determine if the device has a register to select between MSB and LSB Don't Care when using REFp2b commands or the device has just LSB do not care when using REFp2b commands.

PRBS allows the host to determine if the optional PRBS feature in MR12 is supported by the device.

## 3.3 VENDOR ID (cont'd)

**Table 7 — Manufacturers Vendor Code**

| DQ3_A DQ3_B | DQ2_A DQ2_B | DQ1_A DQ1_B | DQ0_A DQ0_B | Manufacturer's Name |
|---|---|---|---|---|
| 0 | 1 | 1 | 0 | SK hynix |

**Table 8 — Internal WCK**

| DQ11_A DQ11_B | Internal WCK |
|---|---|
| 1 | Internal Eighth data rate WCK |

**Table 9 — Channel Density**

| DQ10_A DQ10_B | DQ9_A DQ9_B | DQ8_A DQ8_B | Channel (Device) Density |
|---|---|---|---|
| 0 | 1 | 1 | 8 Gb (16 Gb) |

**Table 10 — WCK Granularity**

| DQ12_A DQ12_B | WCK Granularity |
|---|---|
| 0 | WCK/byte |

**Table 11 — WCK Frequency**

| DQ14_A DQ14_B | DQ13_A DQ13_B | WCK Frequency |
|---|---|---|
| 0 | 1 | Half data rate WCK |

## 3.3 VENDOR ID (cont'd)

### Table 12 — VDDQ Off

| DQ15_A<br>DQ15_B | VDDQ off |
|---|---|
| 1 | Hibernate Self Refresh with VDDQ off supported |

### Table 13 — Programmable P2BR

| DQ0_A<br>DQ0_B | Programmable P2BR |
|---|---|
| 0 | Both LSB (BA0) and MSB (BA3) Don't Care for REFp2b supported |

### Table 14 — PRBS

| DQ1_A<br>DQ1_B | PRBS |
|---|---|
| 0 | Optional PRBS is supported |

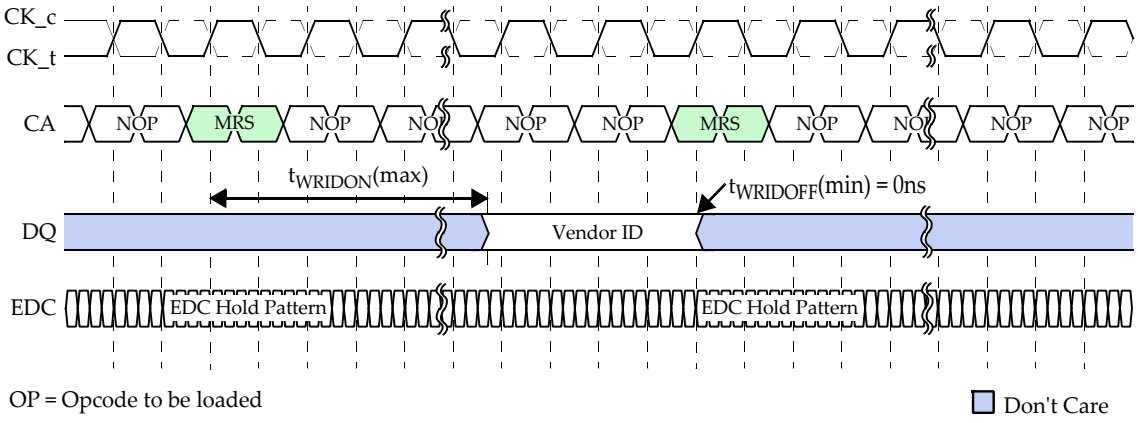



**Figure 5 — Vendor ID Timing Diagram**

# 4. ADDRESS

## 4.1 COMMAND and ADDRESSING

GDDR6 SGRAMs use a double data rate command address scheme to reduce balls required on the device. The command and address is packetized on the 10 CA signals (CA[9:0]) over either single cycle or multi cycles depending on the command (see Command Truth Table). For single cycle commands the command and address data is provided to the device in two parts; the first half is latched on the rising edge of CK; the second half is latched on the falling edge of CK. An 11th CA signal (CA10) is reserved for higher densities.

GDDR6 addressing is defined for a single channel with devices having 2 channels/device.

**Table 15 — Addressing Scheme**

| Memory Density | 16 Gb | |
|---|---|---|
| | 17,179,869,184 | |
| Device Organization | x16 mode | x8 mode |
| Number channels | 2 | |
| Channel Memory Density | 8 Gb | |
| Channel Density | 8,589,934,592 | |
| Array Pre-Fetch (bits, per channel) | 256 | 128 |
| Bank address (per channel) | BA[3:0] | |
| Number Banks (per channel) | 16 | |
| Row address (per channel) | R[13:0] | R[14:0] |
| Number of Rows (per channel) | 16,384 | 32,768 |
| Column address (per channel) | C[6:0] | C[6:0] |
| Page Size (per channel) | 4K | 2K |
| Refresh | 16K/32ms | |
| Refresh period | 1.9us | |

NOTE
1. The column address notation for GDDR6 does not include the lower four address bits as the burst order is always fixed for READ and WRITE commands.
2. Row and column address values on the CA bus that are not used for a particular density must be at valid logic levels.
3. Page Size = 2^COLBITS * (Prefetch_Size/8) where COLBITS is the number of column address bits.
4. Row address range with R[14:13] = 11 for x8 and R[13:12] = 11 for x16 is not present for 12 Gb density

## 4.2 COMMAND ADDRESS BUS INVERSION (CABI)

Command Address Bus Inversion (CABI) reduces the power requirements on Command Address (CA) bus, as the number of CA lines driving a LOW level can be limited to 5 in 2 channel mode or 7 in PC mode.

The Command Address Bus Inversion function is associated with the electrical signalling on the CA lines between a controller and the device, regardless of whether the information conveyed on the CA lines is a row or column address, a command, a mode register op-code, a data mask, or any other pattern.

The CABI_n input is an active LOW double data rate (DDR) signal and sampled by the device at the rising edge of CK and the falling edge of CK along with the CA inputs.

Once enabled by the corresponding CABI Mode Register bit, the GDDR6 SGRAM will invert the pattern received on the CA inputs in case CABI_n was sampled LOW, or leave the pattern non-inverted in case CABI_n was sampled HIGH, as shown in Figure 6.

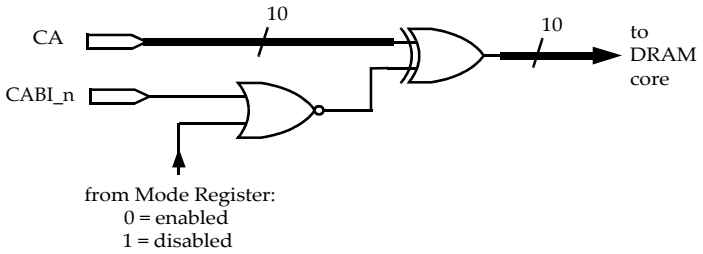
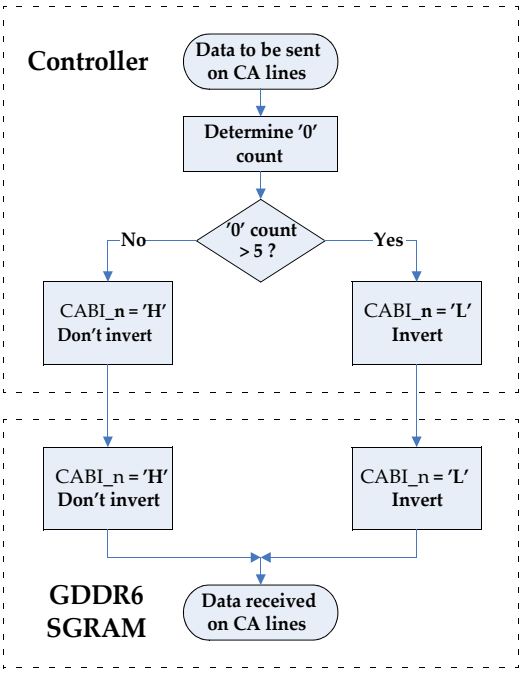



**Figure 6 — Example of Command Address Bus Inversion Logic**

The flow diagram in Figure 7 illustrates the CABI operation. The controller decides whether to invert or not invert the data conveyed on the CA lines. The GDDR6 SGRAM has to perform the reverse operation based on the level of CABI_n. CA input timing parameters are only valid with CABI being enabled and a maximum of 5 or 7 CA inputs driven LOW depending on the channel configuration.




Note: Figure shown for 2 channel configuration. In PC mode check for '0' count > 7

**Figure 7 — Command Address Bus Inversion (CABI) Flow Diagram**

## 4.3 BANK GROUPS

For devices operating at frequencies above a certain threshold ($f_{CKBG}$), the activity within a bank group must be restricted to ensure proper operation of the device. The 16 banks are divided into four bank groups. The bank groups feature is controlled by OP[11:10] in Mode Register 3 (MR3). The assignment of the banks to the bank groups is shown in Table 16.

**Table 16 — Bank Groups**

| Bank | Addressing | | | | 8 Gb - 16 Gb |
|---|---|---|---|---|---|
| | BA3 | BA2 | BA1 | BA0 | 16 banks |
| 0 | 0 | 0 | 0 | 0 | Group A |
| 1 | 0 | 0 | 0 | 1 | |
| 2 | 0 | 0 | 1 | 0 | |
| 3 | 0 | 0 | 1 | 1 | |
| 4 | 0 | 1 | 0 | 0 | Group B |
| 5 | 0 | 1 | 0 | 1 | |
| 6 | 0 | 1 | 1 | 0 | |
| 7 | 0 | 1 | 1 | 1 | |
| 8 | 1 | 0 | 0 | 0 | Group C |
| 9 | 1 | 0 | 0 | 1 | |
| 10 | 1 | 0 | 1 | 0 | |
| 11 | 1 | 0 | 1 | 1 | |
| 12 | 1 | 1 | 0 | 0 | Group D |
| 13 | 1 | 1 | 0 | 1 | |
| 14 | 1 | 1 | 1 | 0 | |
| 15 | 1 | 1 | 1 | 1 | |

These bank groups allow the specification of different command delay parameters depending on whether back-to-back accesses are to banks within one bank group or across bank groups as shown in Table 17.

**Table 17 — Command Sequences Affected by Bank Groups**

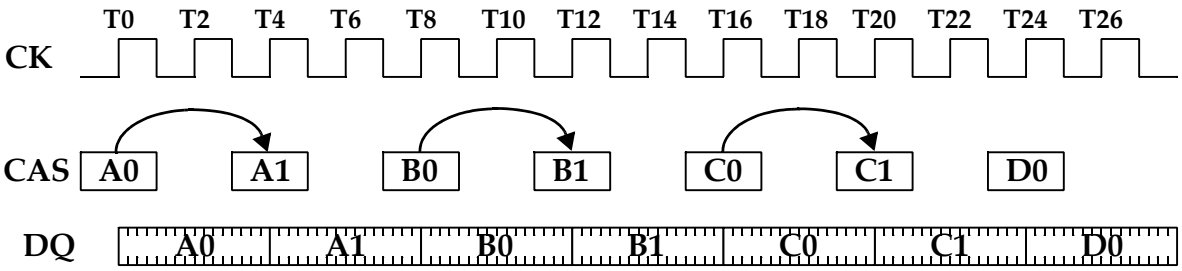
| Command Sequence | Corresponding AC Timing Parameter | | | Notes |
|---|---|---|---|---|
| | Bank Groups Disabled | Bank Groups Enabled | | |
| | | Accesses to a different bank group | Accesses within the same bank group | |
| ACTIVATE to ACTIVATE | $t_{RRDS}$ | $t_{RRDS}$ | $t_{RRDL}$ | |
| WRITE to WRITE | $t_{CCDS}$ | $t_{CCDS}$ | $t_{CCDL}$ | |
| READ to READ | $t_{CCDS}$ | $t_{CCDS}$ | $t_{CCDL}$ | |
| Internal WRITE to READ | $t_{WTRS}$ | $t_{WTRS}$ | $t_{WTRL}$ | |
| READ to PRECHARGE | $t_{RTPS}$ | $t_{RTPS}$ | $t_{RTPL}$ | 1 |

NOTE 1: Parameters $t_{RTPS}$ and $t_{RTPL}$ apply only when READ and PRECHARGE go to the same bank; use $t_{RTPS}$ when BG are disabled, and $t_{RTPL}$ when BG are enabled.

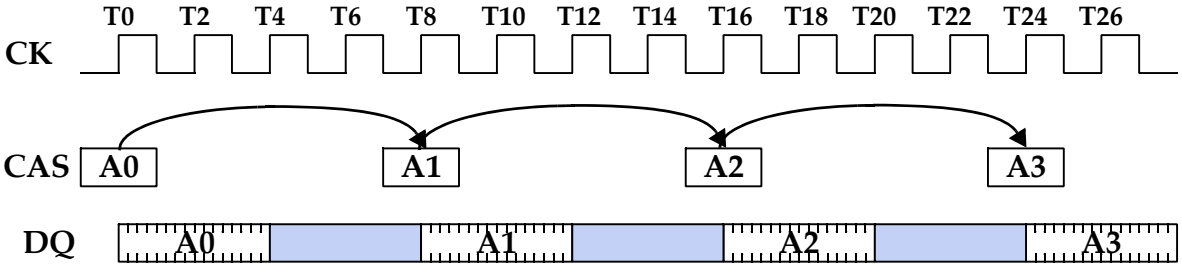
## 4.3 BANK GROUPS (cont'd)

Figure 8 shows back-to-back column accesses based on $t_{CCDL}$ and $t_{CCDS}$ parameters.

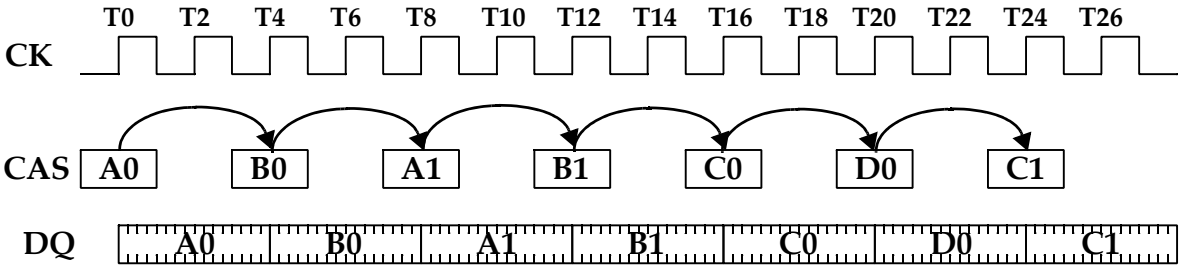
### Example 1 (Bank Groups disabled): $t_{CCDS}$ = 2 * $t_{CK}$




### Example 2: (Bank Groups enabled): $t_{CCDL}$ = 4 * $t_{CK}$




### Example 3: (Bank Groups enabled): $t_{CCDS}$ = 2 * $t_{CK}$




NOTE 1 Column accesses (CAS commands) are to open banks, and $t_{RCD}$ has been met.
NOTE 2 RL = 0 assumed for illustration purposes
NOTE 3 Ax, Bx, Cx, Dx: accesses to bank groups A, B, C or D, respectively
NOTE 4 With bank groups enabled, $t_{CCDL}$ is 3 $t_{CK}$ or 4 $t_{CK}$, as programmed in MR3.

**Figure 8 — $t_{CCDS}$ and $t_{CCDL}$**

# 5. TRAINING

## 5.1 INTERFACE TRAINING SEQUENCE

Due to the high data rates of GDDR6, it is recommended that the interfaces be trained to operate with the optimal timings. GDDR6 SGRAM has features defined which allow for complete and efficient training of the I/O interface without the use of the device's memory array. The interface trainings are required for normal DRAM functionality unless deemed optional by the DRAM vendor or unless running in lower frequency modes as described in the low frequency section. Interface timings will only be guaranteed after all required trainings have been executed.

A recommended order of training sequences has been chosen based on the following criteria:

The Command Address (CA) training must be done first to allow full access to the Mode Registers. (MR15 for CA training is a special single data rate mode register guaranteed to work without training). CA input timing shall function without training as long as $t_{AS/H}$ are met.

WCK2CK training should be done before read training because a shift in WCK relative to CK will cause a shift in all READ timings relative to CK.

READ training should be done before WRITE training because optimal WRITE training depends on correct READ data. As part of WRITE training, the host not only has the ability to find the data-eye optimal position in the horizontal direction but also the ability to adjust VREFD levels using the mode registers to find the data-eye optimal position in the vertical direction.

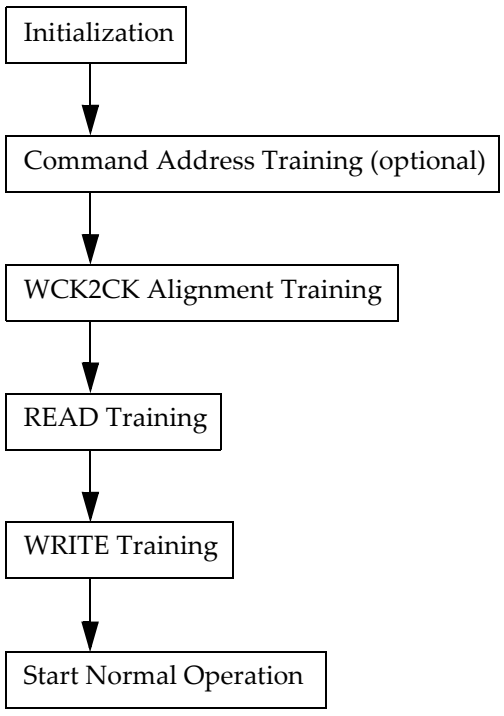



**Figure 9 — Interface Training Sequence**

## 5.2 COMMAND ADDRESS TRAINING

GDDR6 SGRAMs provide a means for Command Address (CA) bus interface training. The controller may use the CA training mode to improve the timing margins on the CA bus.

CA training mode is entered via the CADT bits in Mode Register 15 (MR15). Even though CA9 and CA8 are sampled on both rising and falling edge of CK, the command encoding allows relaxed timing to guarantee the setup and hold times can be met even though the CA bus timing has not yet been trained. When issuing an MRS command the increase in the CA8 timing around the rising edge of clock will lead to one of three different NOP encodings depending on the value on CA8 during the falling edges on either side of the clock rising edge. All NOPs are treated equally.

CA training mode uses an internal bridge between the device's CA inputs and DQ/DBI_n/EDC outputs. It also uses a special COMMAND ADDRESS TRAINING CAPTURE (CAT) command that is encoded using the SDR pin CKE_n. Once the device is placed into CA training mode, the only commands that can be interpreted use CKE_n = HIGH and are the CAT command, a CA training mapping change or CA training exit. When CKE_n = LOW, the CA inputs are do not care, no command is decoded and thus will be treated as a NOP. Once the CA training mode has been entered and $t_{MOD}$ has been met, the CA values registered coincident with the CAT command will be transmitted to the controller on DQ/DBI_n/EDC. The controller is then expected to compare the CA pattern received to the expected value and to adjust the CA transmit timing accordingly. The procedure may be repeated using different CA patterns and interface timings. No WCK clock is required for this special CA training mode. The latched CA values are driven out asynchronously.

The CABI_n signal can be trained directly by using MR15 OP[3:2] = 11 or indirectly as part of a pass with MR15 OP[3:2] = 10 or 01 by enabling the CABI in MR1 OP10. When enabled by the CABI bit in Mode Register 1, command address bus inversion (CABI) is effective during CA training mode. It is suggested to train CABI_n's interface timing together with the other CA lines. CABI is enabled by default on power up or subsequent reset. If CABI is disabled, CABI_n (input) is ignored and CABI_n (output) is undefined during CA training.

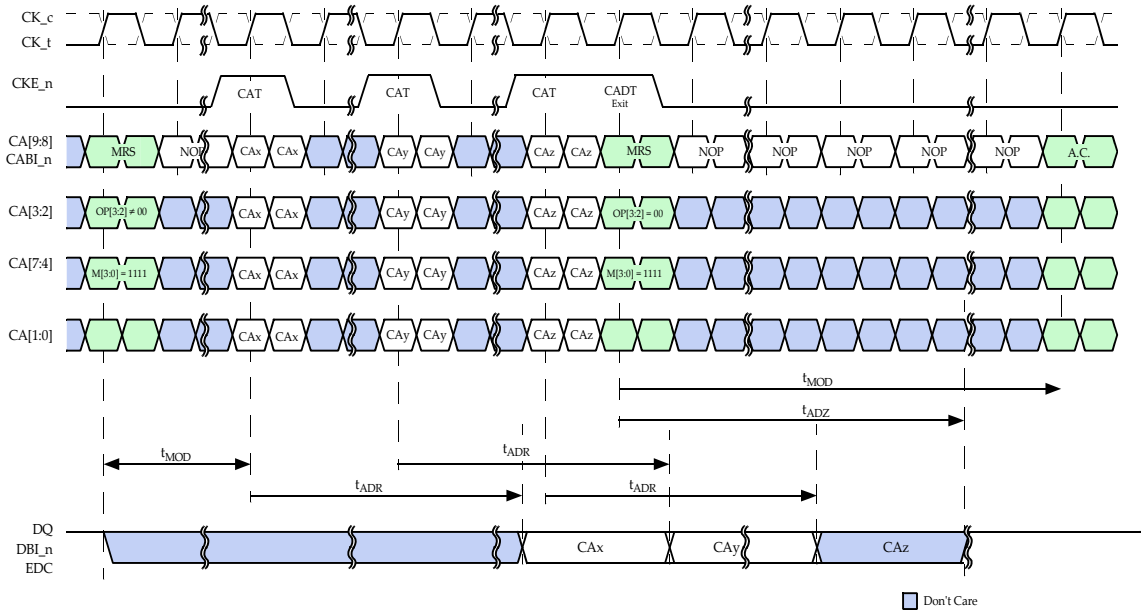
CA training uses multiple passes to train all the CA signals for the rising and falling CK edge setup and hold times. Table 19 defines the correspondence between CA bits and DQ/DBI_n/EDC for all three passes.

When in CA training mode, two cycles of CKE_n = HIGH allows either the change of the mapping while in CA training mode or the exit of CA training mode. The first cycle of CKE_n = HIGH will be interpreted as a CAT command. In the case of a CA training exit, the first cycle of CKE_n = HIGH will be treated as a dummy CAT command as the CA training exit may disable the outputs before the pattern sent to the CA inputs is output on the data bus as $t_{ADZ}$ may be shorter than $t_{ADR}$ and should not be considered a valid training result. The second cycle will be interpreted as either an exit if MR15 OP[3:2] = 00 or a change to the mapping. While in CA training mode, $t_{AS}/t_{AH}$ must be met for the second cycle of CKE_n = HIGH.

Devices configured to x8 mode reflect the CA on the one byte being enabled in that mode, specifically byte 0 in Channel A and byte 1 in Channel B. Devices configured to x16 mode reflect the CA on the same DQ as in x8 mode; in addition they are allowed but not required to reflect the CA on those bytes that are disabled in x8 mode, specifically byte 1 in Channel A and byte 0 in Channel B, thus reflecting each CA twice.

The timing diagram in Figure 10 illustrates the typical command sequence in CA training mode. The DQ/DBI_n/EDC output drivers are enabled as long as the CADT bits are set to either 01, 10 or 11. The minimum spacing between consecutive CAT commands is 2 $t_{CK}$. The minimum spacing between consecutive CA training passes is $t_{MOD}$ as shown in Figure 10. $t_{MOD}$ is also to be met before the next valid command after the CA training exit.

## 5.2 COMMAND ADDRESS TRAINING (cont'd)




NOTE 1 CAT command encoding: CKE_n = H, CA[9:0] = V. A.C. = Any Valid Command

NOTE 2 CA inputs are sampled on rising edge of CK and/or falling edge of CK depending on MR15 OP [3:2]. OP3=0, OP2=1 mapping is shown as an example

NOTE 3 DQs/DBI_n/EDC are enabled when CADT bits in Mode Register 15 set to 01, 10 or 11 (Enter Command Address Training Mode)
DQs/DBI_n/EDC are disabled after $t_{ADZ}$ when CADT is exited (CKE_n = H for 2 cycles and MR15 OP[3:2] = 00 on the second cycle)

NOTE 4 During CA training the CAT is the only valid command with other commands not decoded and thus treated as a NOP

NOTE 5 The initial MRS command to MR15 to enter CA training is shown as DDR, however as timing cannot be guaranteed the command supports relaxed setup and hold timing around the rising edge of CK. If two back-to-back MR15 MRS commands are issued or seen by the DRAM than only the first will be registered. The same two back-to-back MR15 MRS commands are supported at a CA training pass change or at CA training exit.

NOTE 6 Figure shown with CABI enabled. CABI_n is Don't Care if CABI is disabled.

**Figure 10 — Command Address Training Timing - Single Pass**

**Table 18 — AC timings in Command Address Training Mode**

| Parameter | Symbol | Min | Max | Unit |
|---|---|---|---|---|
| CAT command to data out delay | $t_{ADR}$ | 0 | vendor specific | ns |
| CADT off to DQ/DBI_n in ODT state delay | $t_{ADZ}$ | -- | vendor specific | ns |

### Command Address Training with Self Refresh (CADT Self Refresh)

GDDR6 DRAMs support the ability to enable an active Self Refresh mode, here in after CADT Self Refresh mode, to maintain the DRAM contents and enable the required receivers and transmitter that are normally powered down in Self Refresh mode to allow the CA bus to be trained. To enable CADT Self Refresh mode, the CADT SRF register (MR2 OP10) should be set to 1 prior to entering CA training mode with MR15 OP[3:2]. By default CADT Self Refresh will not be enabled when CA training mode is entered. Once the CADT SRF register is set to 1 when MR15 OP[3:2] is set to something other than 00 to enter CA training mode, the DRAM will also enter CADT Self Refresh mode. CA training with Self Refresh does not support Hibernate nor VDDQ off modes and therefore those two Mode Registers are ignored when entering CADT Self Refresh mode.

## 5.2 COMMAND ADDRESS TRAINING (cont'd)

CA Training in Self Refresh works exactly the same as in normal CA training with the exception of exit from CA Training. When CA training mode is exited with two cycles of CKE_n = HIGH and MR15 OP[3:2] = 00 then the device will also exit CADT Self Refresh. After the device has exited CA training with Self Refresh,$t_{XS}$ is required to be met before another valid command as shown in Figure 11.

If multi pass training is planned, the two cycles of CKE_n = HIGH and MR15 OP[3:2] = 01, 10 or 11 will leave the DRAM in the CADT Self Refresh state and change the mapping allowing an additional pass after $t_{MOD}$ is met. The current consumption in CADT Self Refresh mode will be higher than in actual Self Refresh mode because the CK clock, CA and data interface is kept in an active state during CA training.

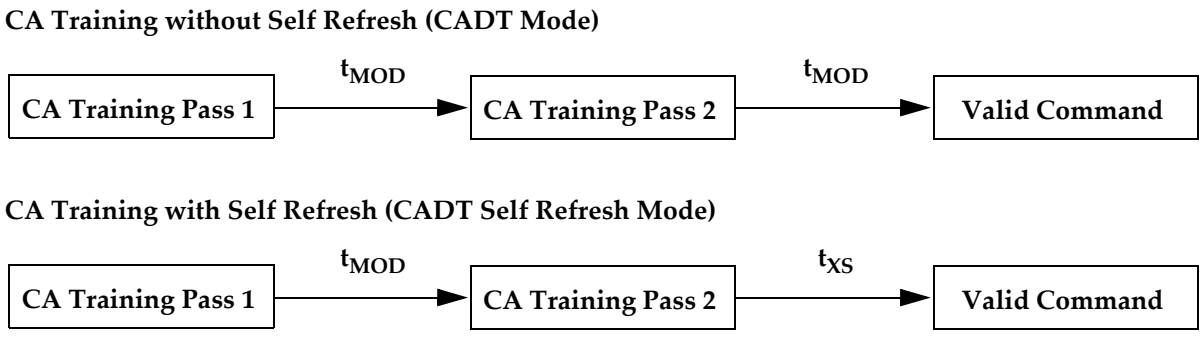
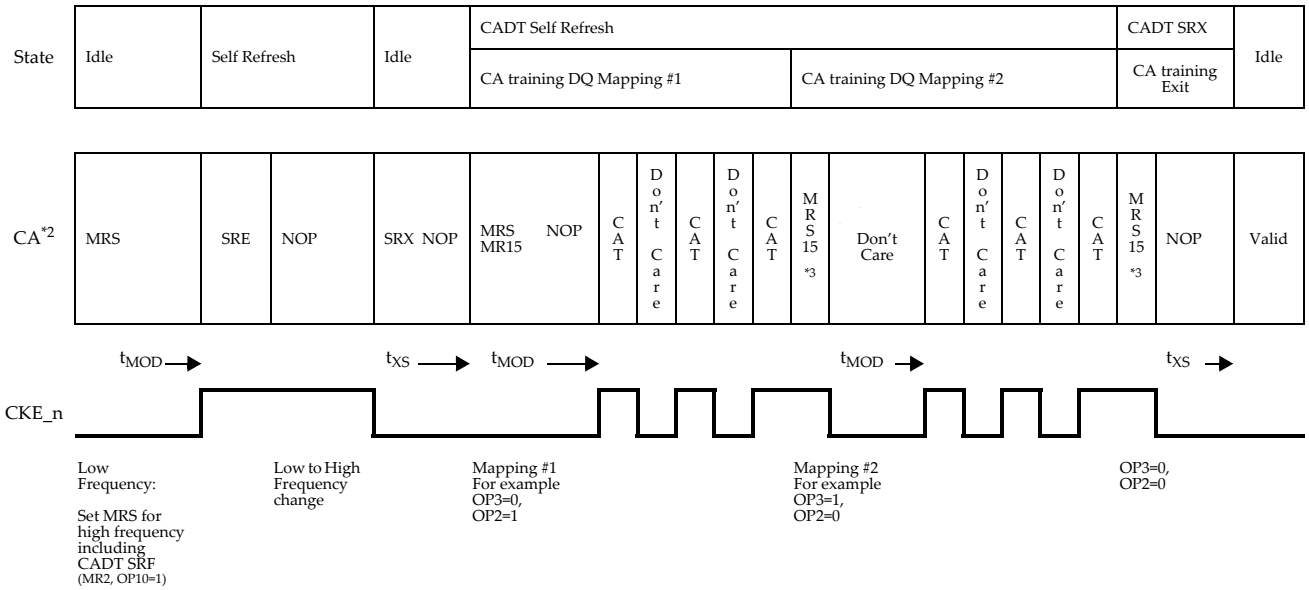



**Figure 11 — Multi pass CA Training Timings**

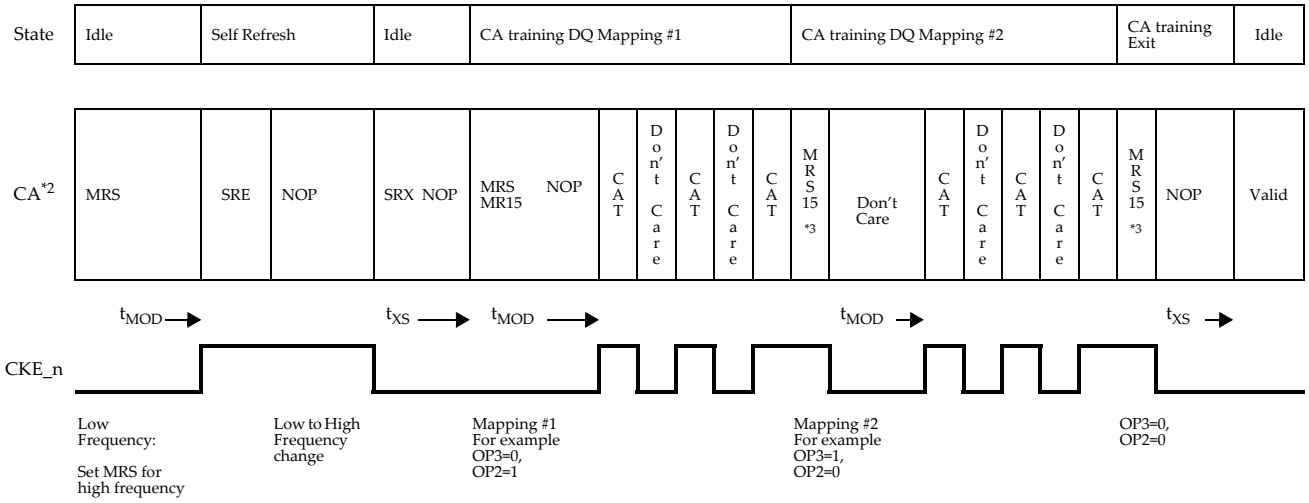
An example of one use of CA Training is shown in Figure 12 and Figure 13. The first figure shows an example that sets the CADT SRF register bit in low frequency where CA bus timing is assumed to be met without prior training. The frequency is changed to a higher frequency using Self Refresh and then at the higher frequency two passes of CA training before the exit of CA Training and CADT Self Refresh. The second figure shows the same example CADT sequence without CADT Self Refresh.

## 5.2 COMMAND ADDRESS TRAINING (cont'd)




Notes:
1) CA training using 2 passes while in Self Refresh is shown as an example.
2) See command truth table for detailed encoding of the Commands shown on the CA bus
3) The second cycle of CKE_n = H signals the DRAM of a mapping change or CA training exit depending on the value of MR15 OP[3:2]. Two back-to-back MR15 MRS commands are supported at CA training entry, pass change or exit as timing cannot be guaranteed.

**Figure 12 — Example of CADT Self Refresh - Multi pass**




Notes:
1) CA training using 2 passes with a frequency change is shown as an example.
2) See command truth table for detailed encoding of the Commands shown on the CA bus
3) The second cycle of CKE_n = H signals the DRAM of a mapping change or CA training exit depending on the value of MR15 OP[3:2]. Two back-to-back MR15 MRS commands are supported at CA training entry, pass change or exit as timing cannot be guaranteed.

**Figure 13 — Example of CA Training - Multi pass**

## 5.2 COMMAND ADDRESS TRAINING (cont'd)

**Table 19 — CA to Data Mapping in CA Training Mode**

| MR15 OP3 = 0 OP2 = 1 | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Output | CA inputs registered at rising edge of CK | | | | | | | | | |
| | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 |
| CH A | EDC0 | DBI0_n | DQ7 | DQ6 | DQ5 | DQ4 | DQ3 | DQ2 | DQ1 | DQ0 |
| CH B | EDC1 | DBI1_n | DQ15 | DQ14 | DQ13 | DQ12 | DQ11 | DQ10 | DQ9 | DQ8 |

| MR15 OP3 = 1 OP2 = 0 | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Output | CA inputs registered at falling edge of CK | | | | | | | | | |
| | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 |
| CH A | EDC0 | DBI0_n | DQ7 | DQ6 | DQ5 | DQ4 | DQ3 | DQ2 | DQ1 | DQ0 |
| CH B | EDC1 | DBI1_n | DQ15 | DQ14 | DQ13 | DQ12 | DQ11 | DQ10 | DQ9 | DQ8 |

| MR15 OP3 = 1 OP2 = 1 | | |
|---|---|---|
| Output | CA inputs registered at rising edge of CK | |
| | CABI_n | |
| CH A | DQ7 | |
| CH B | DQ15 | |
| | | |
| Output | CA inputs registered at falling edge of CK | |
| | CABI_n | |
| CH A | DQ5 | |
| CH B | DQ13 | |

## 5.3 WCK2CK TRAINING

The purpose of WCK2CK training is to align the DATA WCK clock with the COMMAND and ADDRESS (CA) CK clock to aid in the device's internal data synchronization between the logic clocked by CK and WCK. This will help to define both READ and WRITE latencies between the GDDR6 SGRAM and memory controller. WCK2CK training mode is controlled via MRS. The host can read the Vendor ID for info on the WCK characteristics such as WCK Granularity, WCK Frequency and the Internal WCK architecture to ensure the DRAM is configured properly.

Before starting WCK2CK training, the following conditions must be met:

- CK clock is stable and toggling
- The timing of all CA signals must be guaranteed
- The desired WCK2CK alignment point (MR7, OP0) and WCK Ratio (MR10 OP9) are selected.
- 2 Mode Register bits for internal WCK Inversion / Quad Shift (MR10, bits OP[7:4]) must be set to a known state
- All banks are idle and no other command execution is in progress

WCK2CK training must be done after any of the following:

- Device initialization
- Any RLmrs, WLmrs, CRCRL or CRCWL latency change
- CK and WCK frequency changes
- Change of the WCK2CK alignment point (MR7 OP0) or WCK Ratio (MR2 OP11)
- Self Refresh exit or exit from Power-Down when LP2 (OP1 in MR5) is set

Figure 14 and Figure 15 show example WCK2CK training sequences. WCK2CK training is entered via MRS by setting OP8 in MR10. This will initiate the WCK divider circuits associated with WCK0 and WCK1 clocks in the case of WCK/byte or WCK0 in the case of WCK/word. In case the divider circuits on each byte of the channel are not in phase, which is indicated by opposite or multiple U.I. Offset "early/late" phases on the EDC signals, they may be put into phase by using the WCK INV/ Quad shift MR bits to either invert the internal WCK (2UI shift if using internal quarter-rate WCK, 4UI shift if using internal eight-rate WCK) or Quad shift the internal WCK (2UI shift if using internals eighth-rate WCK, not valid with internal quarter-rate WCK). Alternatively, the WCK clocks may be put into a stable inactive state for this initialization event to aid in resetting all dividers to the same output phase as shown in Figure 15. The host is required to restart the WCK clocks in a way that all clock edges meet the WCK clock input specification. Otherwise, the WCK divider circuits for both WCK0 (and WCK1 if WCK/byte) might again have opposite alignment (Internal Quarter data rate) or out of phase alignment (internal eighth data rate). The use of this WCK2CK training method is restricted to operating frequencies up to $f_{WCKSTOP}$. Figure 17 illustrates the difference between a device that uses an internal quarter data rate and a device that uses a internal eighth data rate for the internal WCK divider circuit.

Figure 16 illustrate how the WCK phase information is derived. The phase detectors (PD) sample the internally divided WCK clocks. Only one sample point is shown in the figure for clarity. In reality, when WCK2CK training mode is enabled, a sample will occur every $t_{CK}$ and will be translated to the EDC signals accordingly. If the divided WCK clock arrives early, then the EDC pin outputs a HIGH during the time interval specified in Figure 16. If the divided WCK clock arrives late, then the EDC pin outputs a LOW during the time interval specified in Figure 16. This is shown in Table 20. Finally, if the divided WCK and CK clocks are perfectly aligned at the PD, the EDC signals are indeterminate. For a GDDR6 device with WCK / word the phase information is returned on both EDC0 and EDC1 in x16 mode and on the enabled byte in x8 mode.

## 5.3 WCK2CK TRAINING (cont'd)

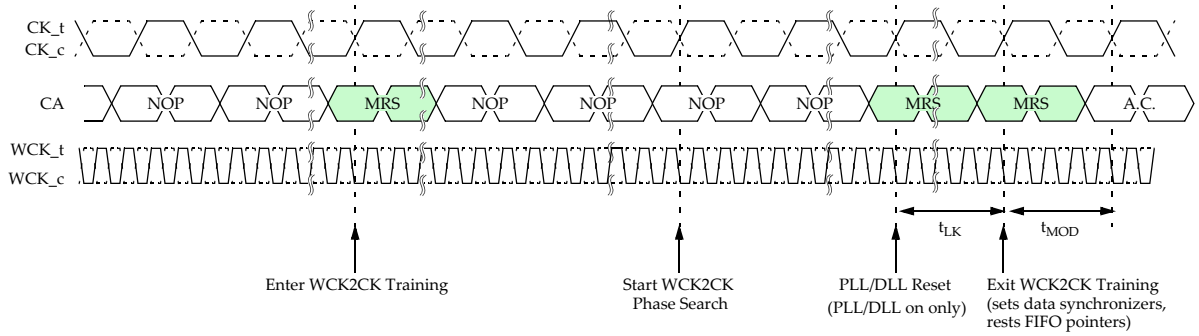



**Figure 14 — Example WCK2CK Training Sequence**

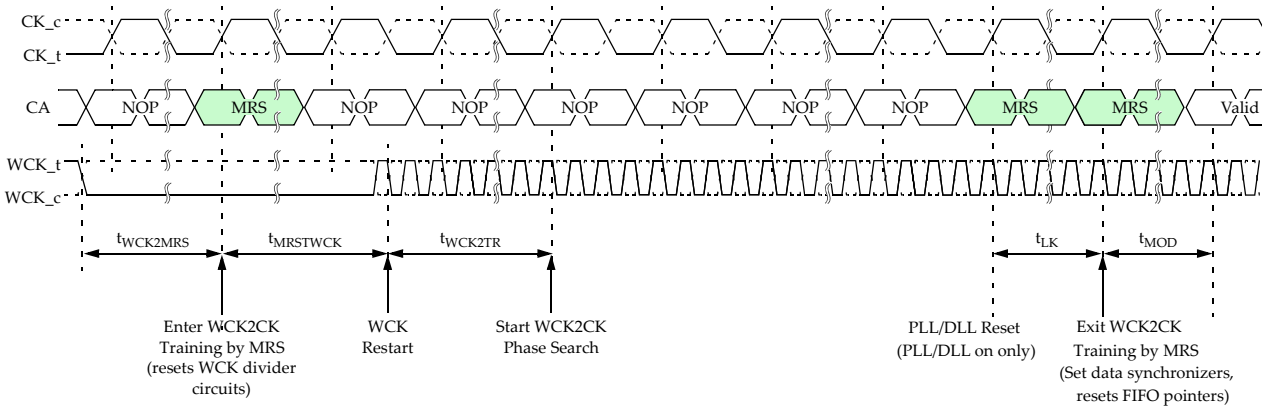
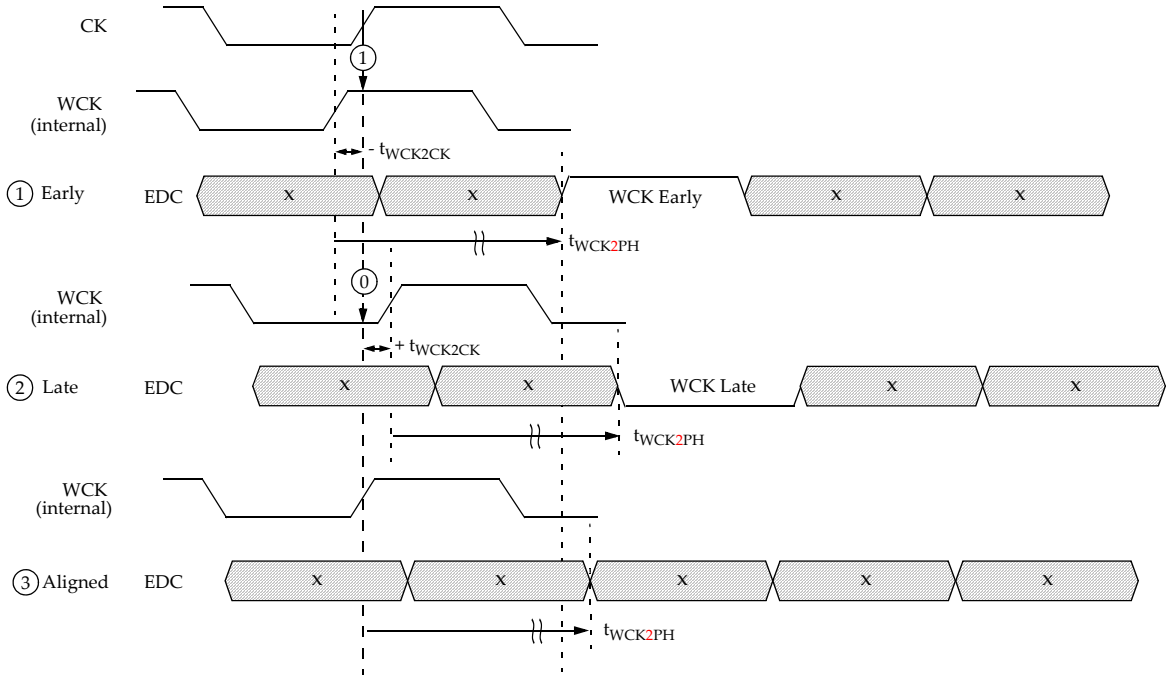


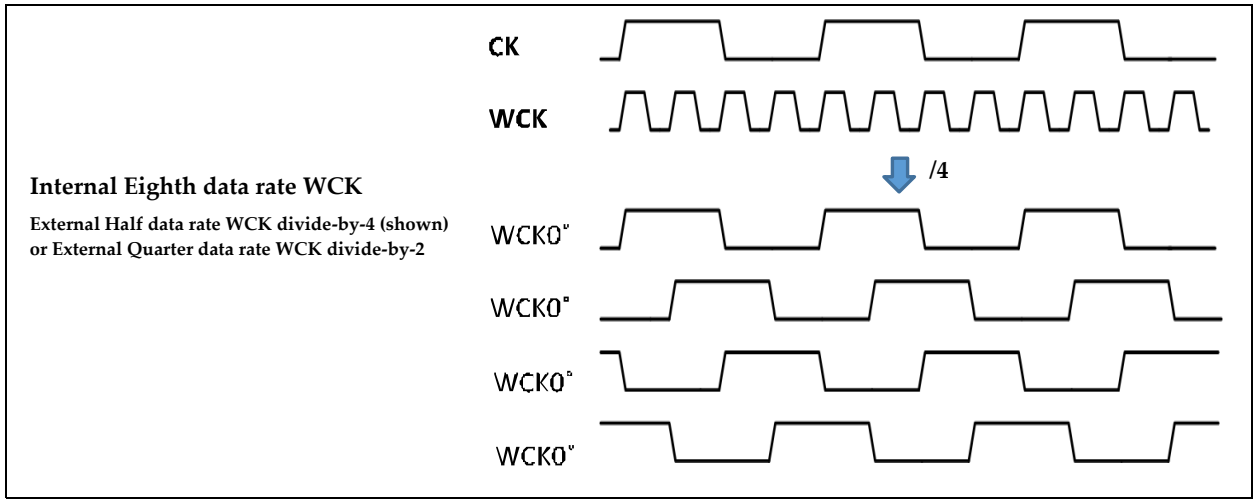

**Figure 15 — Example WCK2CK Training Sequence with WCK Stopping**




Note Example shows an internal eighth data rate WCK.

**Figure 16 — EDC pin Behavior for WCK2CK Training with Internal WCK**

## 5.3 WCK2CK TRAINING (cont'd)




Note 1 External Quarter data rate WCK divide-by-4 is not a valid architecture for a WCK Divider.
Note 2 WCK0° represents the possible positions of the internal divided WCK clock.

**Figure 17 — Internal WCK Divider Circuit**

**Table 20 — Phase Detector and EDC behavior**

| Internal Quarter data rate or Internal Eighth data rate value sampled by CK | WCK2CK Phase | Data on EDC | Action |
|---|---|---|---|
| 'HIGH' | 'Early' | HIGH | Increase Delay on WCK |
| 'LOW' | 'Late' | LOW | Decrease Delay on WCK |

The ideal alignment is indicated by the phase detector output transitioning from "early" to "late" when the delay of the WCK phase is continuously increased. The WCK phase range for ideal alignment is specified by the parameter $t_{WCK2CKPIN}$

## 5.3 WCK2CK TRAINING (cont'd)

WCK2CK training is exited via MRS by resetting OP8 in MR10. For proper reset of the data synchronizers it is required that the WCK and CK clocks are aligned within $t_{WCK2CKSYNC}$ at the time of the WCK2CK training exit.

After exiting WCK2CK training mode, the WCK phase is allowed to further drift from the ideal alignment point by a maximum of $t_{WCK2CK}$ (e.g., due to voltage and temperature variation). Once this WCK phase drift exceeds $t_{WCK2CK}$(min) or $t_{WCK2CK}$(max), it is required to repeat the WCK2CK training and realign the clocks. The values of $t_{WCK2CK}$ and $t_{WCK2CKSYNC}$ may vary with the WCK ratio as defined in the AC timing table.

### WCK2CK alignment at PIN Mode (optional)

The WCK and CK phase alignment point can be changed via MRS by setting OP0 in MR7. In normal mode, when MR7 OP0 is set to '0', the phases of CK and WCK are aligned at the phase detector. On the other hand, when MR7 OP0 is set to '1', the phases of CK and WCK are aligned at the balls as shown in Figure 19. PIN mode is supported up to the max CK clock frequency of $f_{CKPIN}$, and is an option to reduce the time of WCK2CK training at low frequency.

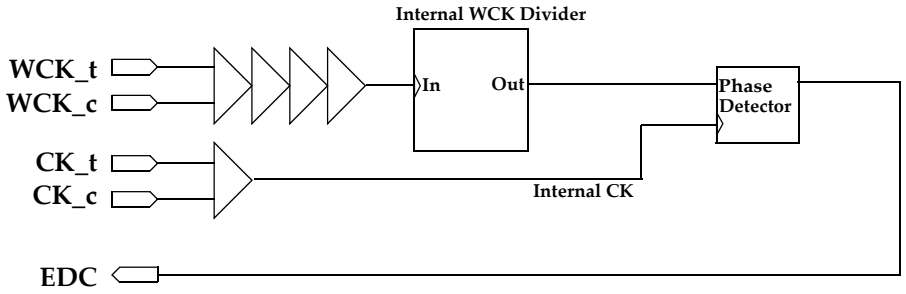



**Figure 18 — Normal Mode**

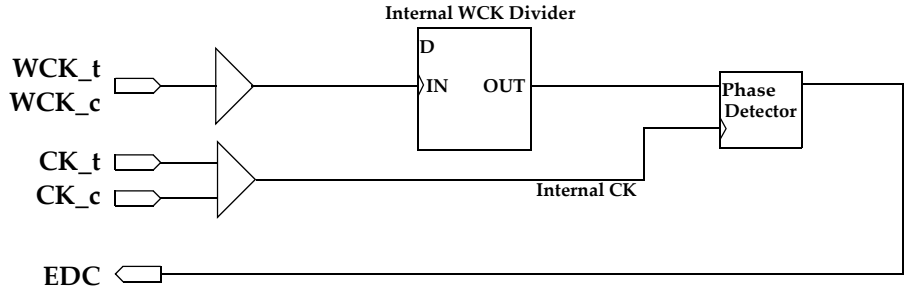



**Figure 19 — Pin Mode (optional)**

## 5.3 WCK2CK TRAINING (cont'd)

### WCK2CK Auto Synchronization

When WCK2CK automatic synchronization mode is enabled using MR7 OP4, WCK2CK training can be limited to meeting the $t_{WCK2CKSYNC}$ specification and issuing two WCK2CK MRS for WCK 2CK training entry and exit. The use of WCK2CK automatic synchronization mode is restricted to operating frequencies up to $f_{CKAUTOSYNC}$ as described in vendor's datasheets and works for both normal and PIN mode.

Table 21 describes WCK2CK training methods for different frequency ranges. Each Frequency range is vendor specific. Normal and PIN mode of WCK2CK training are described in Table 21. Each frequency range is DRAM vendor specific. Divider initialization can be done by training with WCK2CK inversion, WCK2CK stopping, or WCK2CK auto-sync. If the user wants to use WCK2CK stop for divider initialization instead of WCK2CK auto-sync, the user must not set the WCK2CK auto-sync. Below middle frequency, the combined use of PIN and WCK2CK auto-sync modes can minimize WCK2CK training time.

**Table 21 — An example of WCK2CK training simplified for Normal mode and PIN mode**

| | High Frequency (i.e., 12 Gbps) | | Middle Frequency (i.e., 4 Gbps) | | Low Frequency (i.e., 800 Mbps) | |
|---|---|---|---|---|---|---|
| Frequency (Vendor Specific) | | | | | | |
| WCK2CK alignment mode | Normal | PIN | Normal | PIN | Normal | PIN |
| Phase Search | Required | Required | Required | No* | No* | No* |

* NOTE The divided WCK_t/WCK_c should be aligned CK_t/CK_c by WCK2CK Auto Synchronization or WCK stop mode

The following examples describe the WCK2CK training in more detail.

Example 1: outline of a basic WCK2CK training sequence without WCK clock stop:

1) Enable training mode via MRS and wait $t_{MOD}$
2) Sweep and observe the phase independently for WCK0 on EDC0 and WCK1 on EDC1; in case the internal divider circuits are at opposite phase use either the WCK inversion / quad shift bits to flip/shift one of the WCK divider circuits
3) Adjust the WCK phase independently for WCK0 and WCK1 or WCK0 for WCK/word to the optimal point ("ideal alignment")
4) Issue a PLL/DLL reset and wait for $t_{LK}$ (PLL/DLL on mode only)
5) While all WCK and CK are aligned, exit WCK2CK training mode via MRS
6) Wait $t_{MOD}$ for the reset of data synchronizers

Example 2: outline of a basic WCK2CK training sequence with WCK clock stop:

1) Stop WCK clocks with WCK0_t/WCK1_t LOW and WCK0_c/WCK1_c HIGH
2) Wait $t_{WCK2MRS}$ for internal WCK clocks to settle
3) Enable training mode via MRS and wait $t_{MRSTWCK}$ for divider circuits to reset
4) Start WCK clocks without glitches (divider circuits remain in sync)
5) Wait $t_{WCK2TR}$ for internal WCK clocks to stabilize
6) Sweep and observe the phase independently for WCK0 on EDC0 and WCK1 on EDC1 or WCK0 for WCK/word ; adjust the WCK phase to the optimal point ("ideal alignment")
7) Issue a PLL/DLL reset and wait $t_{LK}$ (PLL/DLL on mode only)
8) While all WCK and CK are aligned, exit WCK2CK training mode via MRS
9) Wait $t_{MOD}$ for the reset of data synchronizers

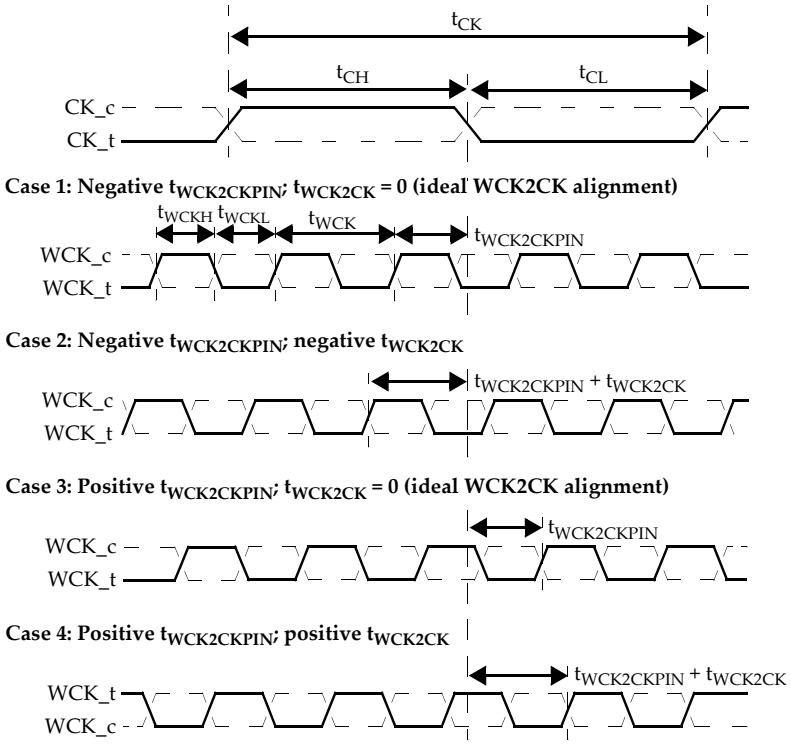
## 5.3 WCK2CK TRAINING (cont'd)

Example 3: WCK2CK training sequence for configurations with 2 devices sharing their WCK clocks:

1) Enable training mode for both DRAMs via MRS and wait $t_{MOD}$
2) For both DRAMs sweep and observe the phase independently for WCK0 on EDC0 and WCK1 on EDC1; in case the internal divider circuits are at opposite phases use either the WCK inversion / quad shift bits to flip/shift one of the WCK divider circuits; use MREA and MREB bits in MR15 (soft chip select) to explicitly direct the MRS command for this operation to one of the two devices.
3) Sweep and observe the phase on the first device; store the setting for the optimal WCK phase.
4) Sweep and observe the phase on the second device; store the setting for the optimal WCK phase.
5) Sweep WCK phase to the midpoint of both device's optimal settings
6) Issue a PLL/DLL reset and wait for $t_{LK}$ (PLL/DLL on mode only)
7) While all WCK and CK are aligned, exit WCK2CK training mode via MRS
8) Wait $t_{MOD}$ for the reset of data synchronizers

READ and WRITE latency timings are defined relative to CK. Any offset in WCK and CK at the balls and/or the phase detector will be reflected in the latency timings. The parameters used to define the relationship between WCK and CK are shown in Figure 20. For more details on the impact on READ and WRITE timings see the OPERATIONS section.

## 5.3 WCK2CK TRAINING (cont'd)




Note: $t_{WCK2CKPIN}$ and $t_{WCK2CK}$ parameter values could be negative or positive numbers, depending on the selected WCK2CK alignment point, PLL-on-or PLL-off mode operation and design implementation. They also vary across PVT. WCK2CK training is required to determine the correct WCK-to-CK phase for stable device operation.

**Figure 21 — WCK2CK Timings for WCK DDR Ratio**

## 5.4 READ TRAINING

Read training allows the memory controller to find the data-eye optimal position (symbol training) and burst frame location (frame training) for each high-speed output of the device. Each pin (DQ[15:0], DBI[1:0]_n, EDC[1:0]) can be individually trained during this sequence.

For Read Training the following conditions must be true:

- at least one bank is active, or a REFab must be in progress and OP2 in Mode Register 5 (MR5) is set to 0 to allow training during a REFab (to disable this special refresh enabling of the WCK clock tree an ACT command must be issued, or the device must be set into power-down or self refresh mode)
- WCK2CK training must be complete
- the PLL/DLL must be locked, if enabled
- RDBI and WDBI must be enabled prior to and during Read Training if the training shall include DBI_n. RDCRC and WRCRC must be enabled prior to and during Read Training if the training shall include the EDC signals.

The following commands are associated with Read Training:

- LDFF to preload the Read FIFO;
- RDTR to read a burst of data directly out of the Read FIFO.

Neither LDFF nor RDTR access the memory core. No MRS is required to enter Read Training.

Figure 22 shows an example of the internal data paths used with LDFF and RDTR. Table 22 lists AC timing parameters associated with Read Training.

Table 22 — LDFF and RDTR TIMINGS

| PARAMETER | SYMBOL | VALUES | | UNIT | NOTES |
|---|---|---|---|---|---|
| | | MIN | MAX | | |
| ACTIVATE to LDFF command delay | $t_{RCDLTR}$ | | – | ns | |
| ACTIVATE to RDTR command delay | $t_{RCDRTR}$ | 18 | – | ns | |
| REFab to RDTR or WRTR command delay | $t_{REFTR}$ | 19 | – | ns | |
| RDTR to RDTR command delay | $t_{CCDS}$ | 2 | – | $t_{CK}$ | 2 |
| LDFF to LDFF command cycle time | $t_{LTLTR}$ | 4 | – | $t_{CK}$ | |
| LDFF15 to RDTR command delay | $t_{LTRTR}$ | 4 | – | $t_{CK}$ | |
| READ or RDTR to LDFF command delay | $t_{RDTLT}$ | RL+CRCRL+4tCK | – | $t_{CK}$ | |
| Note 2: Use $t_{CCDS}$ for gapless consecutive RDTR commands regardless whether Bank Groups is enabled or not. | | | | | |

## 5.4 READ TRAINING (cont'd)

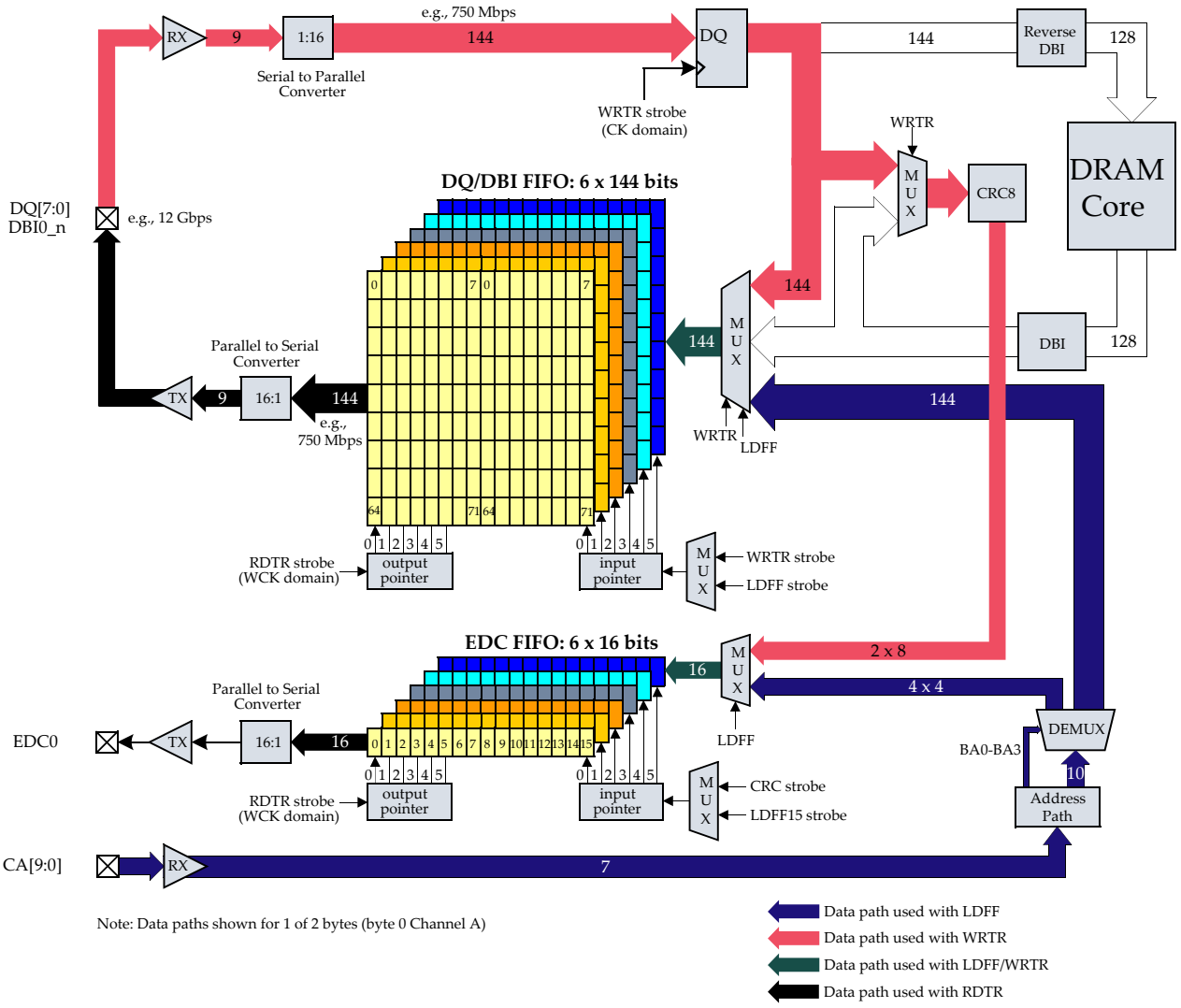



**Figure 22 — Data Paths used for Read and Write Training**

### LDFF Command

The LDFF command (Figure 23) is used to securely load data to the device's Read FIFOs via the CA bus. The READ FIFO has a fixed depth of 6 bursts and thus can store a 6 x 16 = 96 U.I. long bit pattern uniquely to every DQ, DBI_n and EDC pin within a byte.

Sixteen LDFF commands are required to fill one FIFO stage; each LDFF command loads one burst position, and B[3:0] conveyed on CA[7:4] select the burst position from 0 to 15.

The data pattern D[9:0] for DQ[7:0], DBI0_n, and EDC0 is conveyed on CA[5:0]; the data are internally replicated to both bytes in a channel, as shown in Figure 24.

LDFF loads the DBI FIFO when WDBI and RDBI are enabled in the Mode Register. It also loads the EDC FIFO regardless of the WRCRC and RDCRC Mode Register bits, and no CRC is calculated; however, RDBI and RDCRC must be enabled to read the DBI and EDC bits, respectively, with the RDTR command.

If the EDC Mode is set to Half data rate in MR2 OP8, pattern for the EDC pin provided with the LDFF command must be the same for the even and next odd burst position (e.g. burst positions 0-1, 2-3, … , 14-15). If retrieved the EDC FIFO using the RDTR command, such a pattern will appear as a half data rate.

## 5.4 READ TRAINING (cont'd)

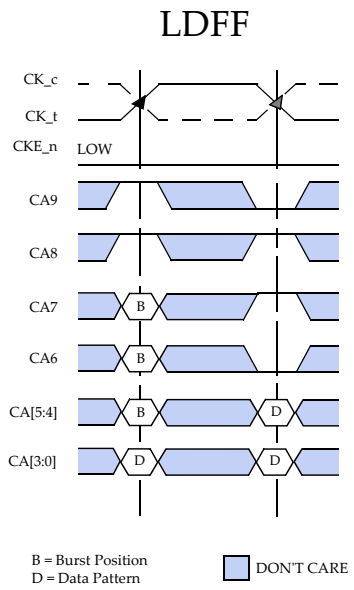



**Figure 23 — LDFF Command**

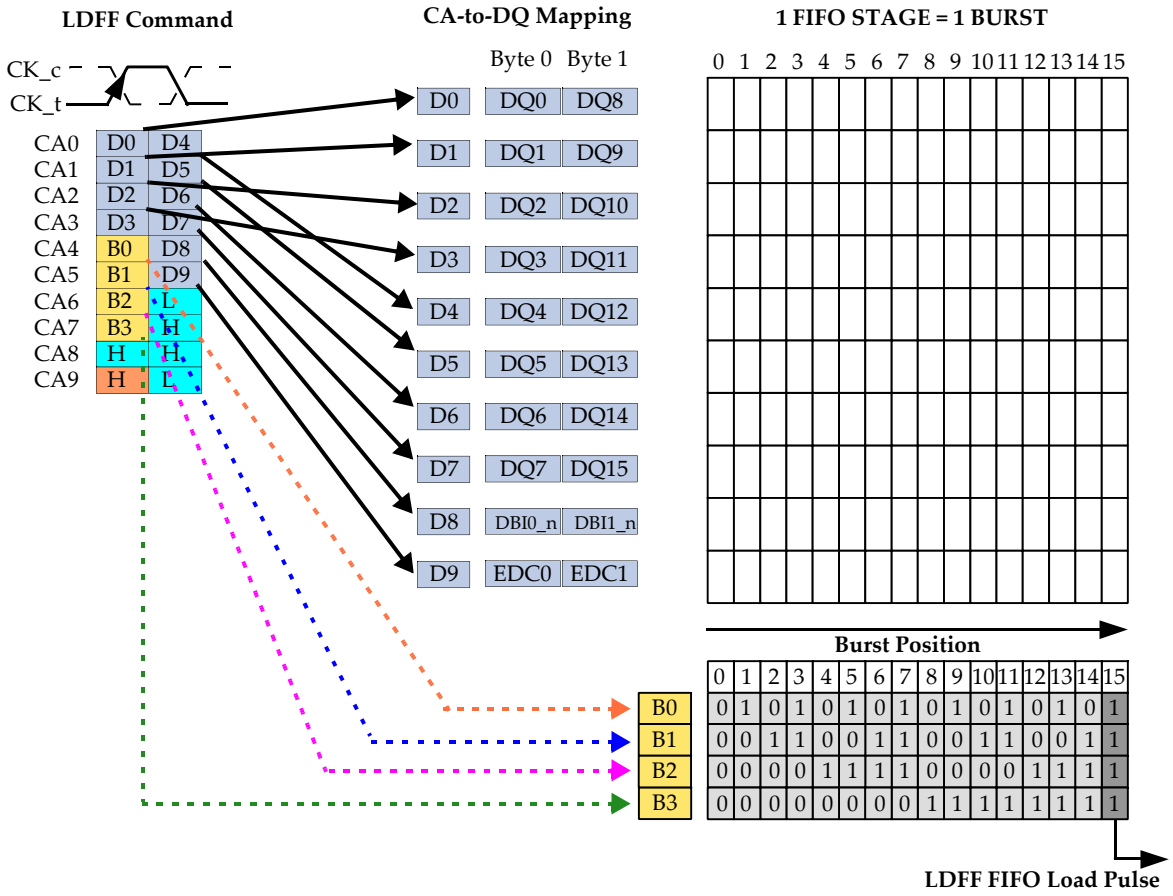



**Figure 24 — LDFF Command Address to DQ/DBI_n/EDC Mapping**

## 5.4 READ TRAINING (cont'd)

All bursts (0 to 15) must be loaded; LDFF commands to burst 0 to 14 may be issued in random order; the LDFF command to burst 15 (LDFF15) must be the last of 16 consecutive LDFF commands, as it effectively loads the data into the FIFO and results in a FIFO pointer increment. Consecutive LDFF commands have to be spaced by at least $t_{LTLTR}$

LDFF pattern may efficiently be replicated to the next FIFO stages by issuing consecutive LDFF commands to burst 15 (with identical data pattern). The data pattern in the scratch memory for LDFF will be available until the first RDTR command.

The DQ/DBI_n output buffers remain in ODT state during LDFF.

An amount of LDFF commands to burst 15 greater than the FIFO depth is allowed and shall result in a looping of the FIFO's data input.

The total number of LDFF commands to burst 15 modulo FIFO depth must equal the total number of RDTR commands modulo FIFO depth when used in conjunction with RDTR. No READ or WRITE commands are allowed between LDFF and RDTR.

The EDC hold pattern is driven on the EDC signals during LDFF.

### RDTR Command

A RDTR burst is initiated with a RDTR command as shown in Figure 25. No bank or column addresses are used as the data is read from the internal READ FIFO, not the array. The length of the burst initiated with a RDTR command is sixteen. There is no interruption nor truncation of RDTR bursts. The FIFO access is suppressed when CE bit is LOW. In two channel mode CE shall be driven HIGH.

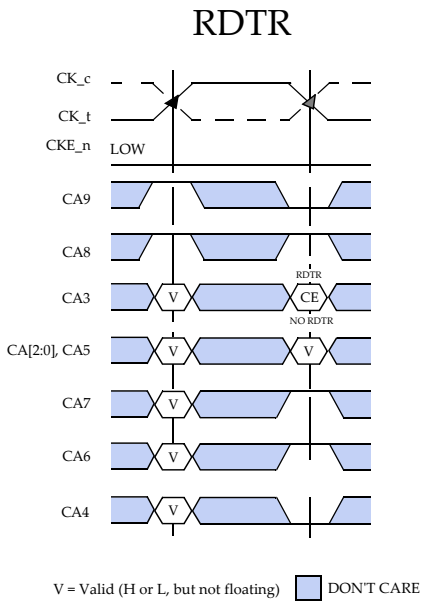



**Figure 25 — RDTR Command**

## 5.4 READ TRAINING (cont'd)

A RDTR command may only be issued when a bank is open or a refresh is in progress and bit OP2 in MR5 is set to 0 to allow training during REFab.

RDBI and RDCRC must be enabled to read the DBI and EDC bits, respectively, with the RDTR command. If not set, DBI_n will remain in ODT state, and the EDC signals will drive the EDC hold pattern.

An amount of RDTR commands greater than the FIFO depth (= 6) is allowed and shall result in a looping of the FIFO's data output.

During RDTR bursts, the first valid data-out element will be available after the Read latency (RL). The latency is the same as for READ. The data on the EDC signals comes with additional CRC latency ($t_{CRCRD}$) after the RL.

Upon completion of a burst, assuming no other RDTR command has been initiated, all DQs and DBI_n will drive a value of '1' and the ODT will be enabled at a maximum of 1 $t_{CK}$ later. The drive value and termination value may be different due to separately defined calibration offsets. If the ODT is disabled, the signals will drive Hi-Z.

Data from any RDTR burst may be concatenated with data from a subsequent RDTR command. A continuous flow of data can be maintained. The first data element from the new burst follows the last element of a completed burst. The new RDTR command should be issued after the first RDTR command according to the $t_{CCDS}$ timing.

A WRTR can be issued any time after a RDTR command as long as the bus turn around time $t_{RTW}$ is met.

The total number of RDTR commands modulo FIFO depth must be equal to total number of WRTR commands modulo FIFO depth when used in conjunction with WRTR. No READ or WRITE commands are allowed between WRTR and RDTR.

The total number of RDTR commands modulo FIFO depth must be equal to the total number of LDFF commands to burst position 15 modulo FIFO depth when used in conjunction with LDFF. No READ or WRITE commands are allowed between LDFF and RDTR.

## 5.5 WRITE TRAINING

Write training allows the memory controller to find the data-eye optimal position (symbol training) and burst frame location (frame training) for each high-speed input of the device. Each signal pin (DQ[15:0] and DBI[1:0]_n) can be individually trained during this sequence.

For Write Training the following conditions must be true:

- at least one bank is active, or a REFab must be in progress and bit OP2 in Mode Register 5 (MR5) is set to 0 to allow training during a REFab (to disable this special REF enabling of the WCK clock tree an ACT command must be issued, or the device must be set into power-down or self refresh mode)
- WCK2CK training should be complete
- Read training should be complete
- RDBI and WDBI must be enabled prior to and during Write Training if the training shall include DBI_n. RDCRC and WRCRC must be enabled prior to and during Write Training if the training shall include the EDC signals.

The following commands are associated with Write Training:

- WRTR to write a burst of data directly into the Read FIFO;
- RDTR to read a burst of data directly out of the Read FIFO.

Neither WRTR nor RDTR access the memory core. No MRS is required to enter Write Training.

Figure 22 shows an example of the internal data paths used with WRTR and RDTR. Figure 27 shows a typical Write training command sequence using WRTR and RDTR. Table 23 lists AC timing parameters associated with WRITE Training.

Table 23 — WRTR and RDTR Timings

| PARAMETER | SYMBOL | VALUES | | UNIT | NOTES |
|---|---|---|---|---|---|
| | | MIN | MAX | | |
| ACTIVATE to WRTR command delay | $t_{RCDWTR}$ | 10 | – | ns | |
| ACTIVATE to RDTR command delay | $t_{RCDRTR}$ | 18 | – | ns | |
| REFab to RDTR or WRTR command delay | $t_{REFTR}$ | 19 | – | ns | |
| RD/WR bank A to RD/WR bank B command delay different bank groups | $t_{CCDS}$ | 2 | – | $t_{CK}$ | 1 |
| WRTR to RDTR command delay | $t_{WTRTR}$ | WL+2tCK+5ns | – | $t_{CK}$ | |
| WRITE to WRTR command delay | $t_{WRWTR}$ | WL+CRCWL+2tCK | – | $t_{CK}$ | |
| READ or RDTR to WRITE or WRTR command delay | $t_{RTW}$ | RL+BL/8+2ns-WL+CRCRL | – | ns | 2 |

Note 1: Use $t_{CCDS}$ for gapless consecutive WRTR and RDTR commands regardless whether Bank Groups is enabled or not.
Note 2: $t_{RTW}$ is not a device limit but determined by the system bus turnaround time. The difference between $t_{WCK2DQO}$ and $t_{WCK2DQI}$ shall be considered in the calculation of the bus turnaround time.

## 5.5 WRITE TRAINING (cont'd)

### WRTR Command

A WRTR burst is initiated with a WRTR command as shown in Figure 26. No bank or column addresses are used as the data is written to the internal READ FIFO, not the array. The length of the burst initiated with a WRTR command is sixteen. There is no interruption nor truncation of WRTR bursts. The FIFO access is suppressed when CE bit is LOW. In two channel mode CE shall be driven HIGH.

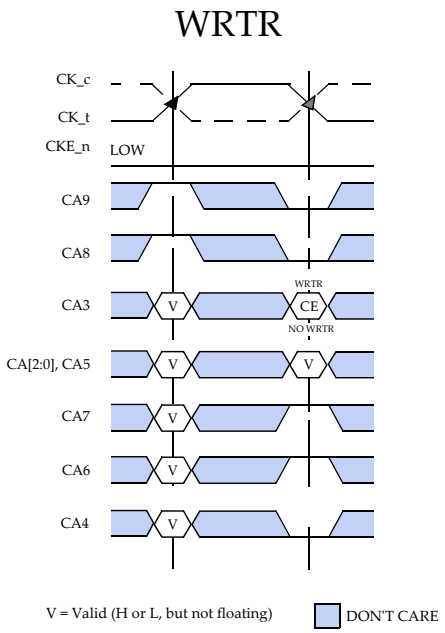



**Figure 26 — WRTR Command**

A WRTR command may only be issued when a bank is open or a refresh is in progress and OP2 in MR5 is set to 0 to allow training during REFab.

WDBI and WRCRC must be enabled to write the DBI and EDC bits, respectively, with the WRTR command. If WDBI is not set, a '1' will be written to the DBI FIFO, and a '1' will be assumed for the DBI_n input in the CRC calculation. In contrast to a normal WRITE, no CRC is returned by the WRTR command and the EDC signals will drive the EDC hold pattern.

Please note that RDCRC must be enabled to read the calculated CRC data with the RDTR command.

An amount of WRTR commands equal to the FIFO depth (= six) is required to fully load the FIFO; any number of WRTR commands greater than the FIFO depth is allowed and shall result in a looping of the FIFO's data input.

During WRTR bursts, the first valid data-in element must be available at the input latch after the Write Latency (WL). The Write Latency is the same as for WRITE.

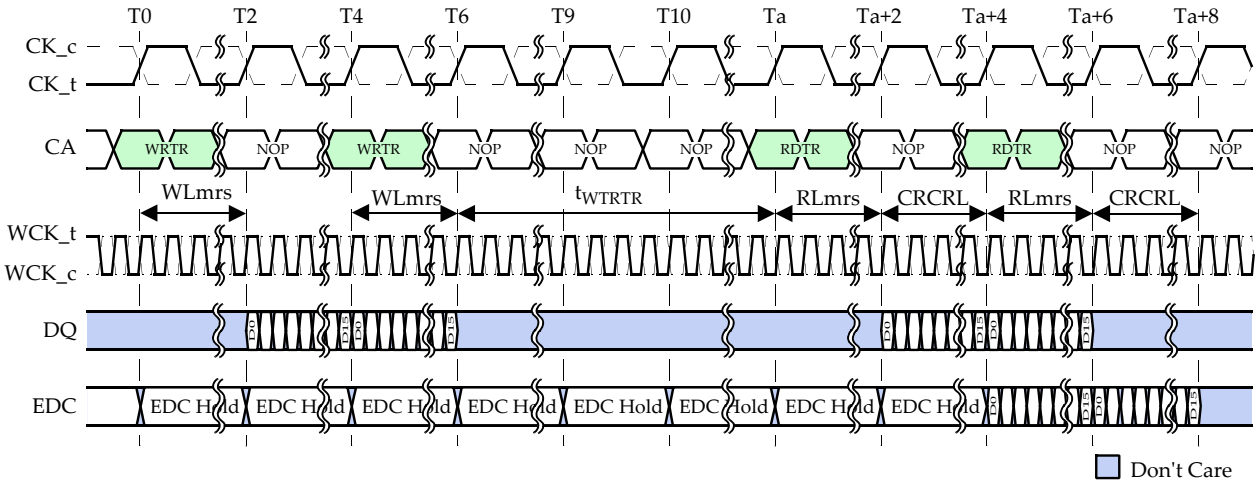
Upon completion of a burst, assuming no other WRTR data is expected on the bus the DQs and DBI_n will be driven according to the ODT state. Any additional input data will be ignored.

## 5.5 WRITE TRAINING (cont'd)

Data from any WRTR burst may be concatenated with data from a subsequent WRTR command. A continuous flow of data can be maintained. The first data element from the new burst follows the last element of a completed burst. The new WRTR command should be issued after the previous WRTR command according to the $t_{CCDS}$ timing.

A RDTR can be issued any time after a WRTR command as long as the internal bus turn around time $t_{RTWTR}$ is met.

The total number of WRTR commands modulo FIFO depth must equal the total number of RDTR commands modulo FIFO depth when used in conjunction with RDTR. No READ or WRITE commands are allowed between WRTR and RDTR.




1. WLmrs, RLmrs and CRCRL set to 2 for ease of illustration; check Mode Register definition for supported settings
2. WCK_t and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

**Figure 27 — Write Training using WRTR and RDTR Commands**

# 6. MODE REGISTERS

GDDR6 specifies 14 Mode Registers to define the specific mode of operation. MR0 to MR12 and MR15 are defined as shown in the overview in Figure 28. MR13 and MR14 are not defined and reserved for vendor specific features. Reprogramming the Mode Registers will not alter the contents of the memory array.

All Mode Registers are programmed via the MODE REGISTER SET (MRS) command and will retain the stored information until they are reprogrammed, chip reset, or the device loses power. Mode Registers must be loaded when all banks are idle and no bursts are in progress, and the controller must wait the specified time $t_{MOD}$ before initiating any subsequent operation. Violating either of these requirements will result in unspecified operation.

No default states are defined for Mode Registers except when otherwise noted. Users therefore must fully initialize all Mode Registers to the desired values upon power-up or after a subsequent chip reset.

Reserved states should not be used, as unknown operation or incompatibility with future versions may result. RFU bits are reserved for future use and must be programmed to 0.

If the user activates bits in an optional field, either the optional field is activated (if option is implemented in the device) or no action is taken by the device (if option is not implemented).

| | M3 | M2 | M1 | M0 | OP11 | OP10 | OP9 | OP8 | OP7 | OP6 | OP5 | OP4 | OP3 | OP2 | OP1 | OP0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MR0 | 0 | 0 | 0 | 0 | Write Recovery (WR) | | | | TM | Read Latency (RLmrs) | | | | Write Latency (WLmrs) | | |
| MR1 | 0 | 0 | 0 | 1 | RFU | CABI | WDBI | RDBI | RFU | Cal Upd | RFU | | Data Termination | | Driver Strength | |
| MR2 | 0 | 0 | 1 | 0 | EDC HR | CADT SRF | RDQS | EDC mode | Self Refresh | | OCD Pullup Driver Offset | | | OCD Pulldown Driver Offset | | |
| MR3 | 0 | 0 | 1 | 1 | Bank Groups | | WR Scaling | | Info | | CA Termination Offset | | | DQ and WCK Termination Offset | | |
| MR4 | 0 | 1 | 0 | 0 | EDC Inv | WR CRC | RD CRC | CRC Read Latency (CRCRL) | | CRC Write Latency (CRCWL) | | | EDC Hold Pattern | | | |
| MR5 | 0 | 1 | 0 | 1 | RFU | | | | | | RFU | | | LP3 | LP2 | RFU |
| MR6 | 0 | 1 | 1 | 0 | Pin Sub-Address | | | | | VREFD Level | | | | | | |
| MR7 | 0 | 1 | 1 | 1 | DCC | | VDD Range | | Half VREFD | HALF VREFC | DQ PreA | Auto Sync | LF Mode | RFU | Hiber-nate | WCK AP |
| MR8 | 1 | 0 | 0 | 0 | CK Termination | | WR EHF | RL EHF | REFpb | CK AC | EDC Hi-Z | CA TO | CAH Terminnation | | CAL Termination | |
| MR9 | 1 | 0 | 0 | 1 | Pin Sub-Address | | | | | RFU | | | Decision Feedback Equalization (DFE) | | | |
| MR10 | 1 | 0 | 1 | 0 | WCK Termination | | WCK Ratio | WCK2 CK | WCK Inv / QS Byte 1 | | WCK Inv / QS Byte 0 | | VREFC Offsets | | | |
| MR11 | 1 | 0 | 1 | 1 | PASR Row Segment Mask | | | | PASR 2-Bank Mask | | | | | | | |
| MR12 | 1 | 1 | 0 | 0 | RFU | | | | | | | | | PRBS | P2BR Addr | VDDQ OFF |
| MR13 | 1 | 1 | 0 | 1 | Reserved for Vendor Specific Features | | | | | | | | | | | |
| MR14 | 1 | 1 | 1 | 0 | DCA | | | | | | | | Reserved for Vendor Specific Features | | | |
| MR15 | 1 | 1 | 1 | 1 | | | | | | | | | | | CADT | MRE |

**Figure 28 — Mode Registers Overview**

## 6.1 MODE REGISTER 0 (MR0)

Mode Register 0 controls operating modes such as Write Latency, Read Latency, Write Recovery and Test Mode as shown in Figure 29. The register is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 0000.

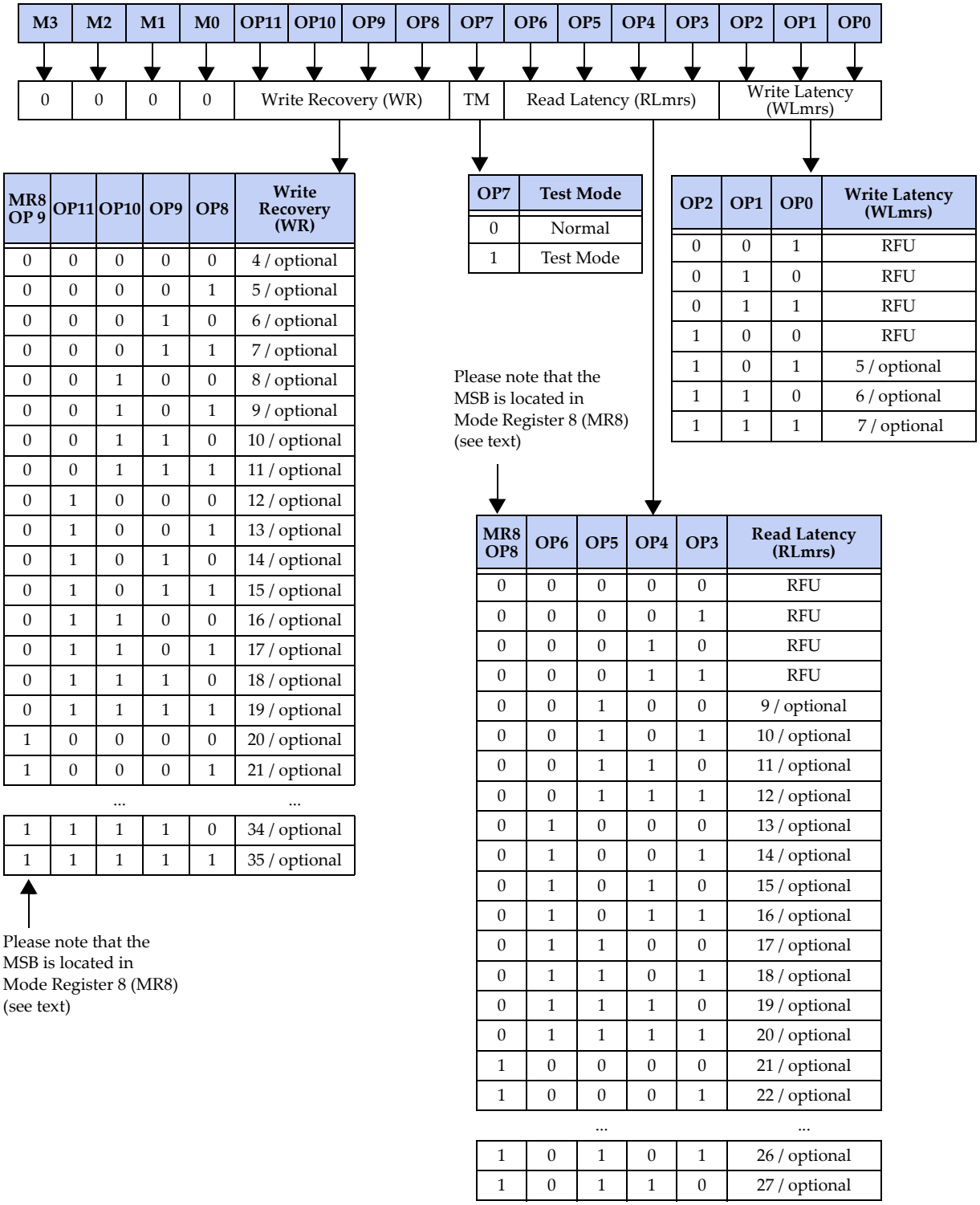



**Figure 29 — Mode Register 0 (MR0) Definition**

| Data Rate (Speed bin, VDD) | RL | | WL | CRC Latency | | Unit |
|---|---|---|---|---|---|---|
| | DBI off | DBI on | | CRCRL | CRCWL | |
| 16Gbps (VDD=1.35V) | TBD | TBD | TBD | TBD | TBD | tCK |
| 14Gbps (VDD=1.35V) | 23 | 24 | 7 | 3 | 14 | tCK |
| 12Gbps (VDD=1.35V) | 21 | 22 | 7 | 2 | 11 | tCK |
| 10Gbps (VDD=1.35V) | 19 | 20 | 7 | 2 | 11 | tCK |
| 8Gbps (VDD=1.35V) | 15 | 15 | 7 | 2 | 11 | tCK |
| 6Gbps (VDD=1.35V) | 14 | 15 | 7 | 2 | 11 | tCK |
| 4Gbps (VDD=1.35V) | 11 | 11 | 7 | 2 | 10 | tCK |
| 2Gbps (VDD=1.35V) | 9 | 9 | 7 | 2 | 10 | tCK |
| 12Gbps (VDD=1.25V) | 23 | 24 | 7 | 3 | 14 | tCK |
| 10Gbps (VDD=1.25V) | 21 | 22 | 7 | 3 | 14 | tCK |
| 8Gbps (VDD=1.25V) | 19 | 20 | 6 | 2 | 14 | tCK |
| 6Gbps (VDD=1.25V) | 16 | 17 | 6 | 2 | 13 | tCK |
| 4Gbps (VDD=1.25V) | 13 | 14 | 5 | 1 | 12 | tCK |
| 2Gbps (VDD=1.25V) | 11 | 11 | 5 | 1 | 10 | tCK |

Note: User should set RL, WL, CRCL for each speed bin according to the above table.

## 6.1 MODE REGISTER 0 (MR0) (cont'd)

### Write Latency (WLmrs)

The Write latency (WLmrs) is the delay in clock cycles used in the calculation of the total Write latency (WL) between the registration of a WRITE or WRTR command and the availability of the first piece of input data. DRAM vendor specifications should be checked for value(s) of WLmrs supported. All WLmrs values are marked as "optional", allowing the DRAM vendor to define the minimum and maximum supported WLmrs values; the supported WLmrs range must be contiguous. As the Write latency is set to higher values (i.e., ... 5, 6, 7, 8 clocks) the input receivers turn on when the WRITE or WRTR command is registered. The full Write latency definition can be found in the section entitled OPERATION.

### Read Latency (RLmrs)

The Read latency (RLmrs) is the delay in clock cycles used in the calculation of the total Read latency (RL) between the registration of a READ or RDTR command and the availability of the first piece of output data.

RLmrs is specified by bits MR0 OP[6:3] and MR8 OP8, defining a RLmrs range of 9 to 36 $t_{CK}$. Please note that the MSB is located in MR8; a Read latency change therefore may require two MRS commands. All RLmrs values are marked as "optional", allowing the DRAM vendor to define the minimum and maximum supported RLmrs values; the supported RLmrs range must be contiguous. The full Read latency definition can be found in the section entitled OPERATION.

### Write Recovery (WR)

The programmed WR value is used to delay the internal auto-precharge until after the tWR timing has been met. The WR register bits are not a required function and may be implemented at the discretion of the DRAM manufacturer.

WR must be programmed with a value greater than or equal to RU{$t_{WR}/t_{CK}$}, where RU stands for round up, $t_{WR}$ is the analog value from the vendor datasheet and $t_{CK}$ is the operating clock cycle time.

WR is specified by MR0 OP[11:8] and MR8 OP9, defining a WR range of 4 to 35 $t_{CK}$. Please note that the MSB is located in MR8; a WR change therefore may require two MRS commands. All WR values are marked as "optional", allowing the DRAM vendor to define the minimum and maximum supported WR values; the supported WR range must be contiguous. The number of clock cycles programmed for WR is multiplied by a factor of 1x (default), 2x or 3x as given by the WR Scaling bits in MR3 OP[9:8].

### Test Mode

The normal operating mode is selected by issuing a MODE REGISTER SET command with bit OP7 set to'0', and bits OP[6:0] and OP[11:8] set to the desired values. Programming bit OP7 to '1' places the device into a test mode that is only to be used by the DRAM manufacturer. No functional operation is specified with test mode enabled.

## 6.2 MODE REGISTER 1 (MR1)

Mode Register 1 controls functions like driver strength, data termination, Read DBI, Write DBI, CABI, control of calibration updates, as shown in Figure 30.

The register is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 0001. OP[3:0], OP6 and OP10 of this register are initialized with '0's.

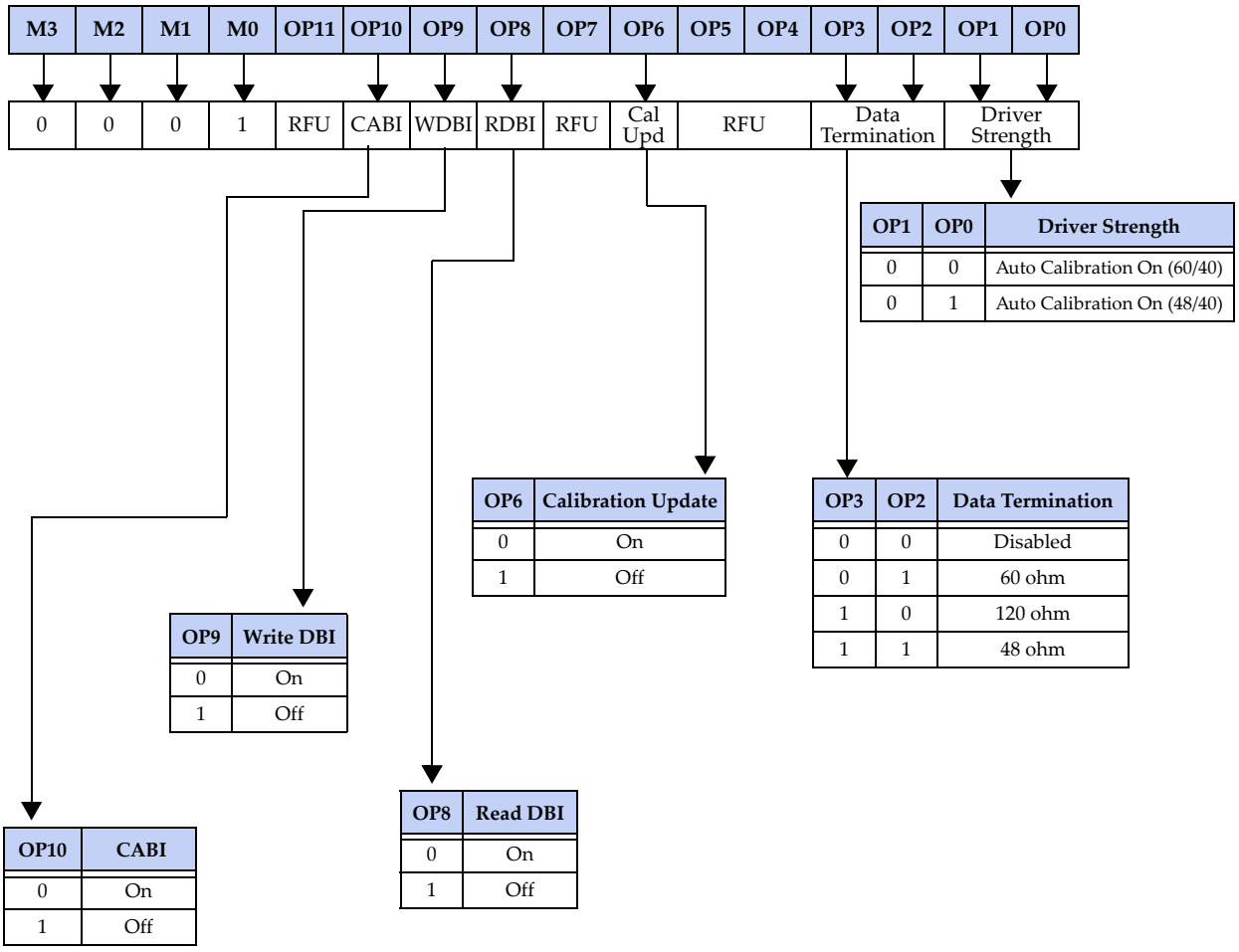



**Figure 30 — Mode Register 1 (MR1) Definition**

### Impedance Auto Calibration of Output Buffer and Active Terminator

GDDR6 SGRAMs offer auto calibrating impedance output buffers and on-die terminations. This enables a user to match the driver impedance and terminations to the system within a given range. To adjust the impedance, an external precision resistor is connected between the ZQ and $V_{SS}$. A nominal resistor value of 120 Ohms is equivalent to the 40 Ohms Pulldown and 60 or 48 Ohms Pullup nominal impedances of devices. RESET_n is not internally terminated.

The output driver and on-die termination impedances are updated during all REFab commands to compensate for variations in supply voltage and temperature. The impedance updates are transparent to the system.

## 6.2 MODE REGISTER 1 (MR1) (cont'd)

### Driver Strength

OP[1:0] define the driver strength. The Auto Calibration setting enables the Auto Calibration functionality for the Pulldown, Pullup and Termination over process, temperature and voltage changes.

### Data Termination

OP[3:2] define the data termination value for the DQs and DBI_n. Data termination is disabled by default; it can be set to a value of 48 Ohm, 60 Ohm, or 120 Ohm depending on system conditions. Data termination may also be turned off.

### Calibration Update

The Calibration Update setting enables the calibration value to be updated automatically by the Auto Calibration engine. The function is enabled upon power-up to reduce update induced jitter. The user may decide to suppress updates from the auto calibration engine by disabling Calibration Update (OP6=1).

The calibration updates can occur with any REFab command. The update is not complete for a time $t_{KO}$ after the latching of the REFab command. During this $t_{KO}$ time, only NOP commands may be issued.

Calibration updates for the CK ODT can be disabled separately by the CK AC bit in MR8 OP6.

### RDBI and WDBI

Bit OP8 controls Data Bus Inversion (DBI) for READs (RDBI), and bit OP9 controls Data Bus Inversion for WRITEs (WDBI). For more details on DBI see READ and WRITE Data Bus Inversion (DBI) in the section entitled OPERATION.

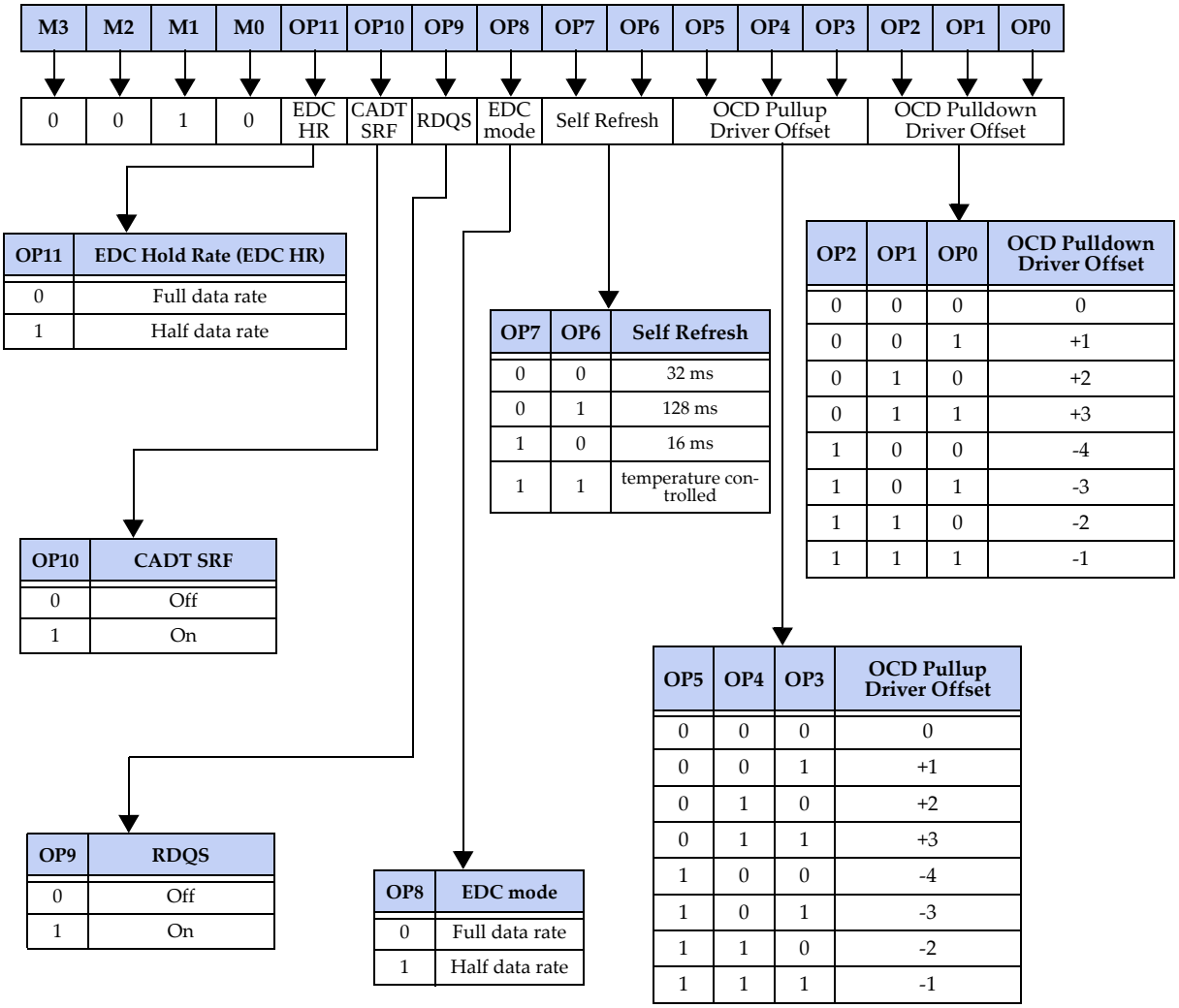
### CABI

Command Address Bus Inversion (CABI) is selected independently from DBI using bit OP10. When enabled any data sent over the CA bus (whether opcode, addresses, LDFF data or DM) is inverted or not inverted based on the state of CABI_n signal. With a change to the CABI register it is required to wait $t_{MOD}$ instead of $t_{MRD}$ with CABI_n held HIGH after an MRS command that changes the CABI and any subsequent MRS command. For more details on CABI see Command Address Bus Inversion (CABI) in the section entitled ADDRESS.

## 6.3 MODE REGISTER 2 (MR2)

Mode Register 2 defines the output driver (OCD) offsets, EDC mode, EDC Hold Rate, Self Refresh, RDQS mode and CADT SRF functions as shown in Figure 31. Bit OP10 of this register is initialized with '0'.

Mode Register 2 is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 0010.




| OP11 | EDC Hold Rate (EDC HR) |
|---|---|
| 0 | Full data rate |
| 1 | Half data rate |

| OP10 | CADT SRF |
|---|---|
| 0 | Off |
| 1 | On |

| OP9 | RDQS |
|---|---|
| 0 | Off |
| 1 | On |

| OP8 | EDC mode |
|---|---|
| 0 | Full data rate |
| 1 | Half data rate |

| OP7 | OP6 | Self Refresh |
|---|---|---|
| 0 | 0 | 32 ms |
| 0 | 1 | 128 ms |
| 1 | 0 | 16 ms |
| 1 | 1 | temperature controlled |

| OP5 | OP4 | OP3 | OCD Pullup Driver Offset |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | +1 |
| 0 | 1 | 0 | +2 |
| 0 | 1 | 1 | +3 |
| 1 | 0 | 0 | -4 |
| 1 | 0 | 1 | -3 |
| 1 | 1 | 0 | -2 |
| 1 | 1 | 1 | -1 |

| OP2 | OP1 | OP0 | OCD Pulldown Driver Offset |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | +1 |
| 0 | 1 | 0 | +2 |
| 0 | 1 | 1 | +3 |
| 1 | 0 | 0 | -4 |
| 1 | 0 | 1 | -3 |
| 1 | 1 | 0 | -2 |
| 1 | 1 | 1 | -1 |

**Figure 31 — Mode Register 2 (MR2) Definition**

### CADT SRF

OP10 enables Self Refresh when CA Training mode is entered. A detailed description of the CA Training while in Self Refresh can be found in the section entitled TRAINING.

### RDQS Mode

OP9 enables the RDQS mode of the device. In this mode the EDC outputs will act as a READ strobe (RDQS). No CRC is supported in RDQS mode, and all related bits in MR4 will be ignored. A detailed description of the RDQS mode can be found in the section entitled OPERATION.

### EDC mode

EDC mode selects between Full data rate and Half data rate EDC modes. See EDC section for more details.

## 6.3 MODE REGISTER 2 (MR2) (cont'd)

### Self Refresh

The refresh interval in Self Refresh mode may be set to 32ms, or being controlled by an integrated temperature sensor.

### EDC Hold Rate

EDC Hold Rate selects between Full data rate and Half data rate for the EDC hold pattern provided the hold pattern is programmed to 1010 or 0101. Programming the hold pattern to 0011, 0110, 1100 or 1001 effectively results in a half or quarter data rate hold pattern depending on the EDC HR setting. EDC Hold Rate is set independently of the EDC mode.

only half data rate PRBS is supported, EDC HR is not to affect the output when the PRBS feature is supported and enabled in MR12 OP2.

### Impedance Offsets (MR2 OP[5:0] & MR3 OP[5:0])

The driver and termination impedances may be offset individually for PD driver, PU driver, DQ/DBI_n/WCK termination and CA termination. The offset impedance step values may be non-linear and will vary across DRAM vendors and across PVT. With negative offset steps the drive strengths will be decreased and Ron will be increased. With positive offset steps the drive strengths will be increased and Ron will be decreased. With negative offset steps the termination value will be increased. With positive offset steps the termination value will be decreased.

IV curves and AC timings are only guaranteed with zero offset.

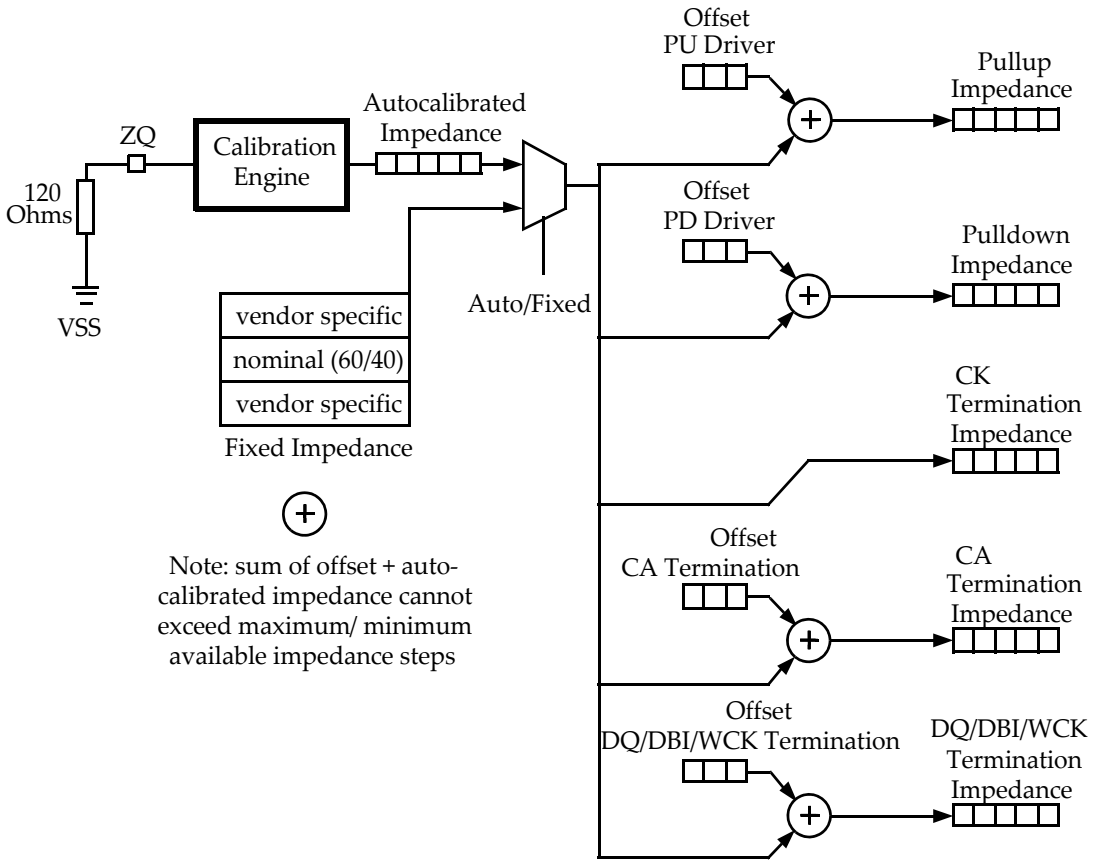



**Figure 32 — Impedance Offsets**

## 6.4 MODE REGISTER 3 (MR3)

Mode Register 3 controls functions including Bank Groups, WR Scaling, DRAM Info, Termination offsets for CA as well as Data and WCK as shown in Figure 33.

Mode Register 3 is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 0011.

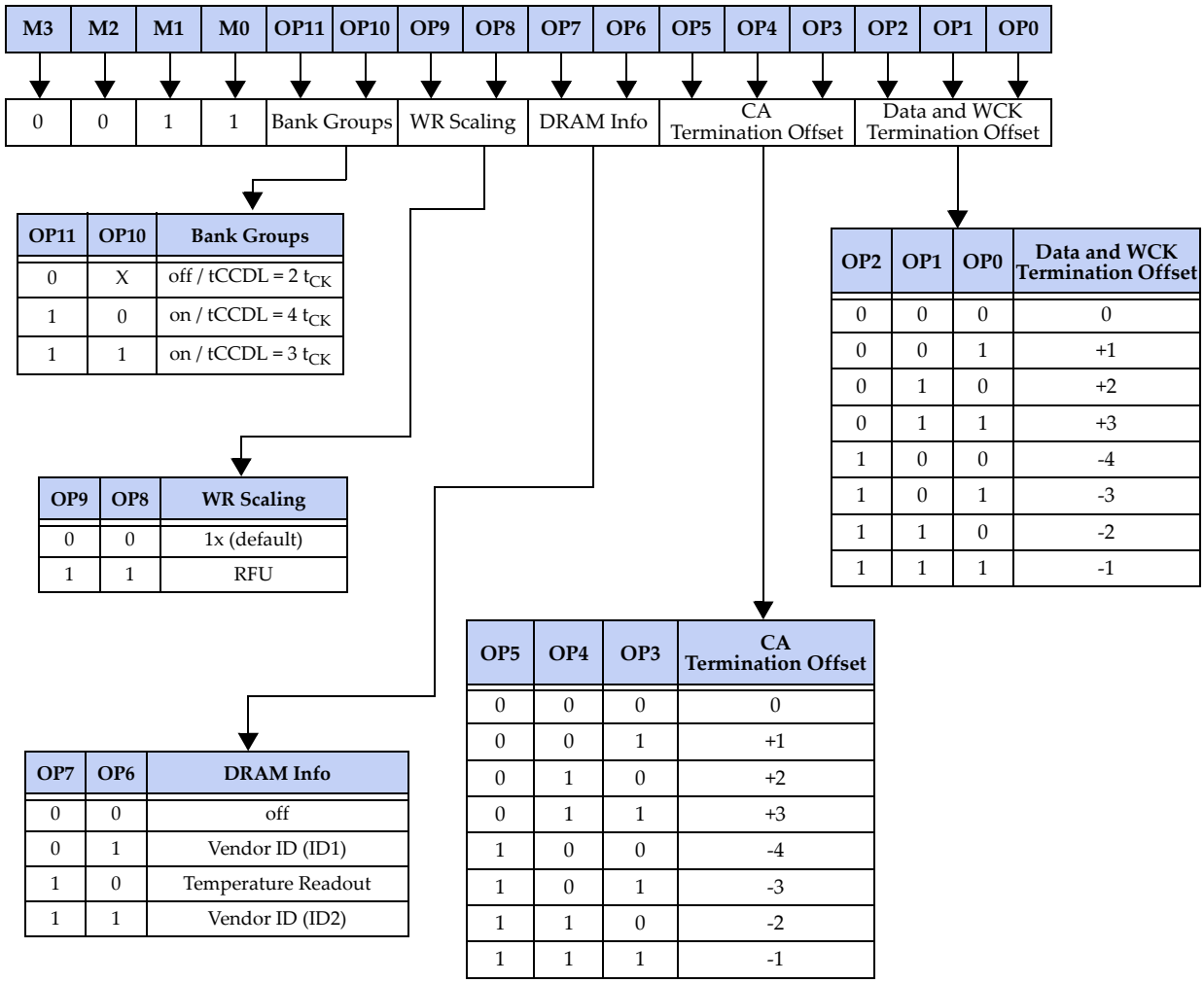



**Figure 33 — Mode Register 3 (MR3) Definition**

### DRAM Info

OP[7:6] enable the DRAM Info mode which is provided to output the Vendor ID or Temperature Readout.

The Vendor ID identifies the manufacturer of the device, and provides the die revision, memory density and other information about the device. See the Vendor ID section for more details.

### WR Scaling

MR9 OP[9:8] enables the number of clock cycles for Write Recovery (WR) programmed in MR8 OP9 and MR0 OP[11:8] to be either the value set or scaled by either 2x or 3x.

## 6.4 MODE REGISTER 3 (MR3) (cont'd)

### Bank Groups

OP11 enables the bank groups feature, and OP10 specifies the min column-to-column command delay ($t_{CCDL}$). With OP11 set to '1', back-to-back column accesses within a bank group have to be spaced by 3 or 4 clocks as defined by bit OP10. With OP11 set to '0', the bank groups feature is disabled and $t_{CCDL}$ equals $t_{CCDS}$.

The DRAM vendor's datasheet specifies the operating frequency limit below which the user may run the device without activating the bank groups feature ($f_{CKBG}$).

## 6.5 MODE REGISTER 4 (MR4)

Mode Register 4 defines the Error Detection Code (EDC) features shown in Figure 34.

The register is programmed via the MODE REGISTER SET (MRS) command with M[3:0]= 0100. OP[3:0] (EDC Hold Pattern) of this register are initialized with '1111'.

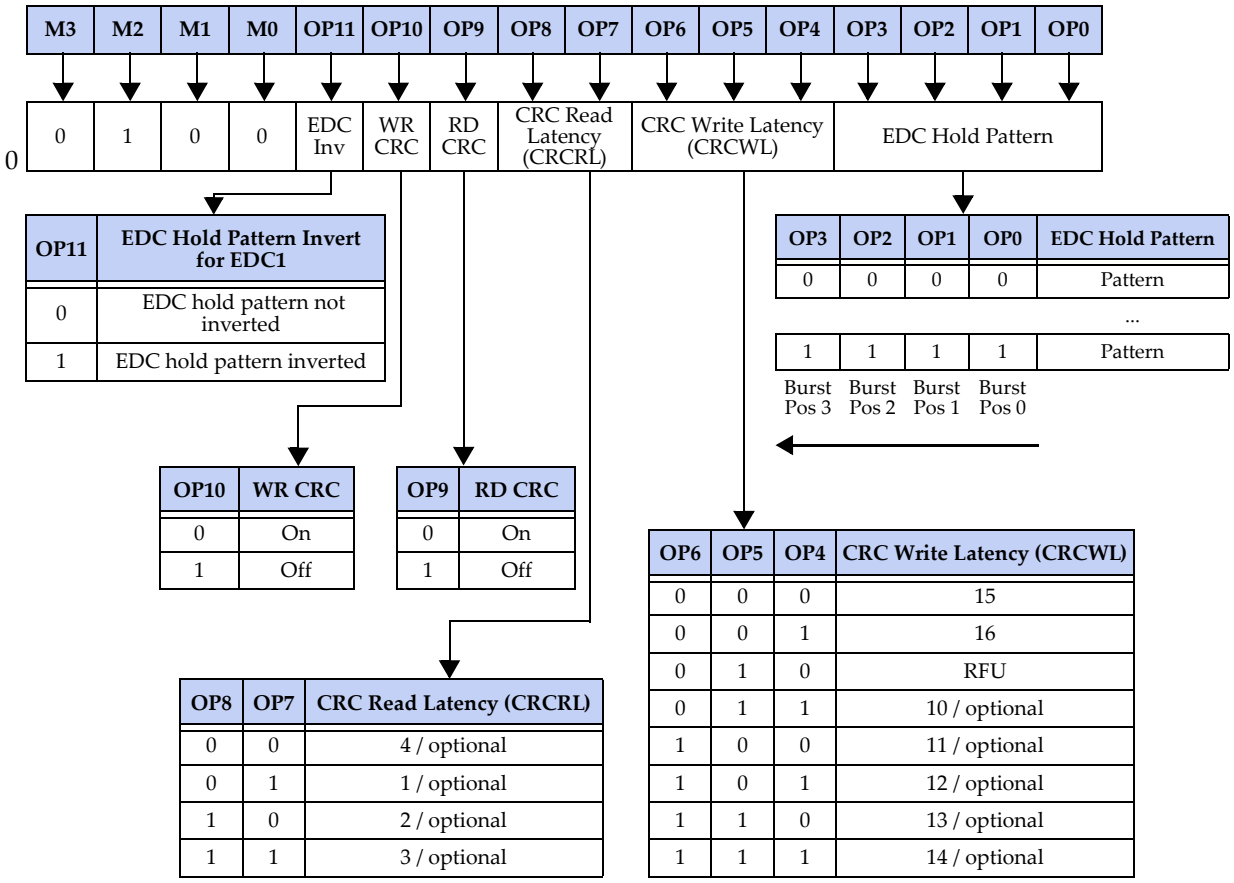



**Figure 34 — Mode Register 4 (MR4) Definition**

### EDC Hold pattern / EDC Invert

The 4-bit EDC hold pattern is considered a background pattern transmitted on the EDC signals. The register is initialized with all '1's. The pattern is shifted from right to left and repeated with every 4 U.I. The output timing is the same as of a READ burst.

CRC bursts calculated from WRITEs or READs will replace the EDC hold pattern for the duration of those bursts, provided CRC is enabled for those bursts.

The EDC hold pattern will be replaced by a pseudo random bit stream when the optional PRBS feature is supported by a device and enabled in MR12 OP2.

The EDC hold pattern will be undefined for $t_{MOD}$ with the following MRS commands:

- MR4 that changes bits OP[3:0] or OP[11:9]
- MR1 that changes bits OP[1:0]
- MR6 that changes TX EQ codes
- MR12 OP2 to disable the optional PRBS feature

## 6.5 MODE REGISTER 4 (MR4) (cont'd)

The EDC hold pattern will not be transmitted when the device is in CA training mode, in WCK2CK training mode, in Self Refresh mode, in reset state, in power-down state with the LP2 bit set, or EDC Hi-Z.

With OP11 set to '1', EDC1 will transmit the inverted EDC hold pattern, resulting in a pseudo-differential pattern. OP11 is ignored for READ, WRITE and RDTR CRC bursts and the clock phase information in WCK2CK training mode.

### Read CRC and CRC Read Latency (CRCRL)

Bit OP9 controls the CRC calculation for READ bursts, and bits OP[8:7] hold the CRC read latency. When enabled, the calculated CRC pattern will be transmitted on the EDC signals with the latency as programmed in the CRCRL field of this register. With Read CRC being off, no CRC will be calculated for READ bursts, and the EDC hold pattern will be transmitted instead.

All CRCRL values are marked as "optional", allowing the DRAM vendor to define the minimum and maximum supported CRCRL values; the supported CRCRL range must be contiguous.

### Write CRC and CRC Write Latency (CRCWL)

Bit OP10 controls the CRC calculation for WRITE bursts, and bits OP[6:4] hold the CRC write latency. When enabled, the calculated CRC pattern will be transmitted on the EDC signals with the latency as programmed in the CRCWL field of this register. With Write CRC being off, no CRC will be calculated for WRITE bursts, and the EDC hold pattern will be transmitted instead.

All CRCWL values are marked as "optional", allowing the DRAM vendor to define the minimum and maximum supported CRCWL values; the supported CRCWL range must be contiguous.

## 6.6 MODE REGISTER 5 (MR5)

Mode Register 5 defines digital RAS, PLL band-width and low power modes as shown in Figure 35.

The register is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 0101.

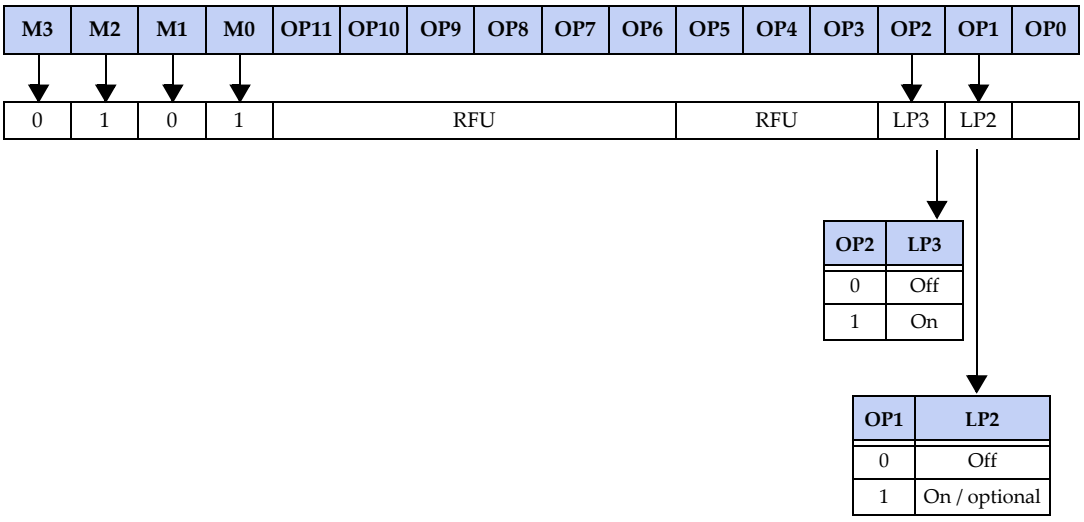



**Figure 35 — Mode Register 5 (MR5) Definition**

### Low Power Modes (LP2, LP3)

OP[2:0] control several low power modes of the device. The modes are independent of each other. LP2 are optional features.

When OP1 (LP2) is set, the WCK receivers may be turned off during power-down.

When OP2 (LP3) is set, RDTR, WRTR and LDFF commands are not allowed while a REFab command is being executed.

## 6.7 MODE REGISTER 6 (MR6) & MODE REGISTER 9 (MR9)

Mode Register 6 and Mode Register 9 control data input receiver properties like VREFD level and DFE as shown in Figure 36 and Figure 37. MR6 also controls the optional data output driver equalization (TX EQ). Both registers use a 5-bit wide sub-address to allow programming of VREFD level and DFE values individually for each data input pin.

Mode Register 6 is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 0110. Mode Register 9 is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 1001.

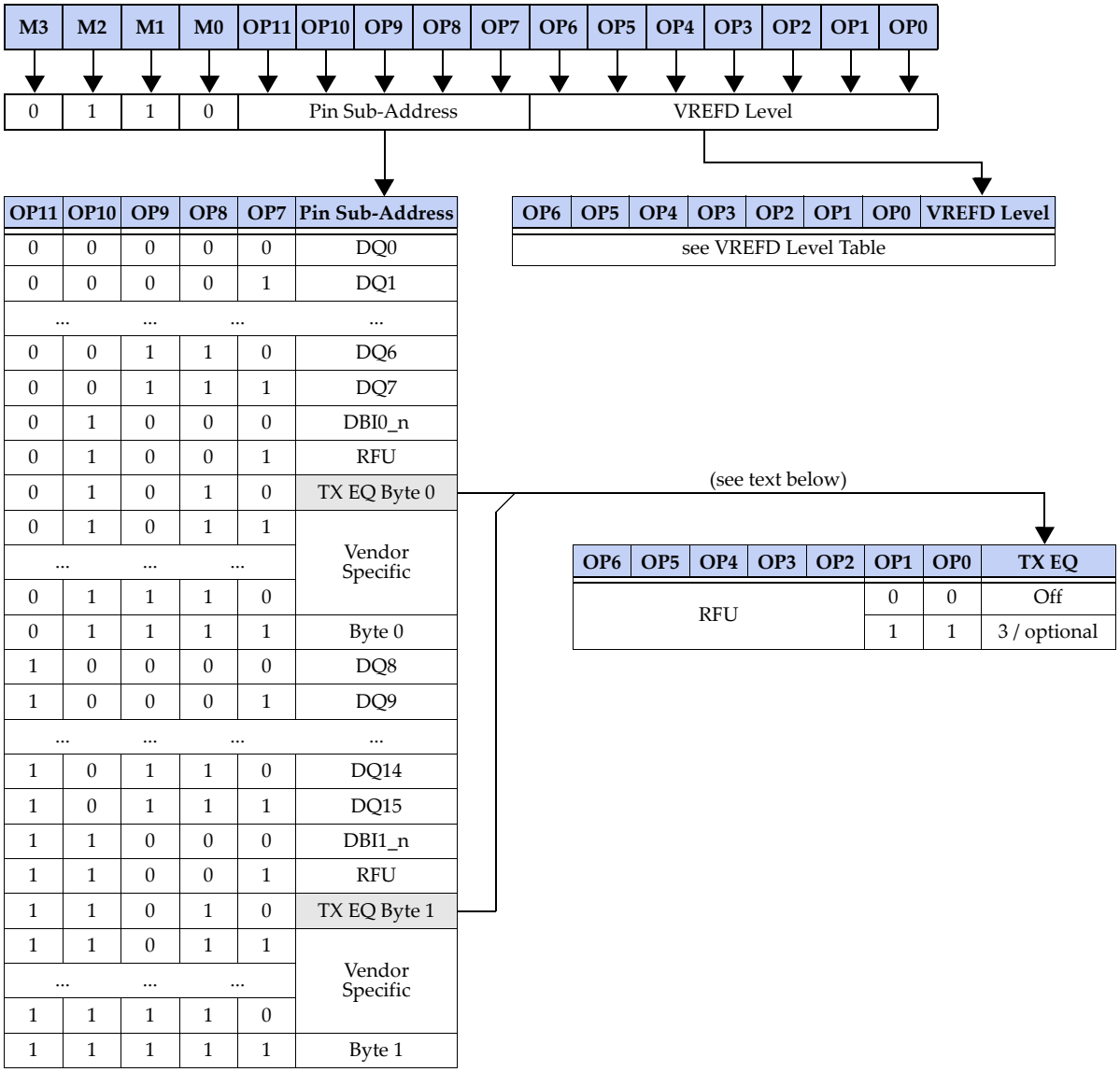
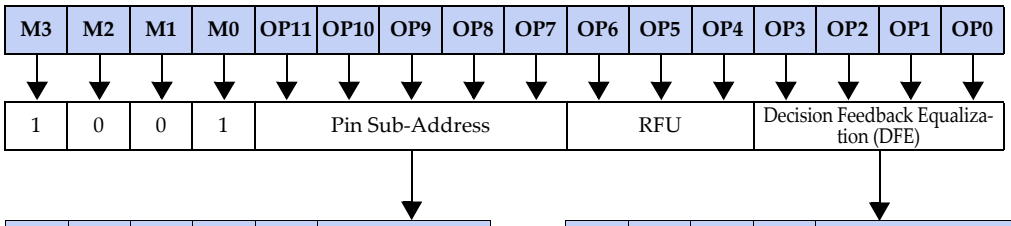



**Figure 36 — Mode Register 6 (MR6) Definition**

## 6.7 MODE REGISTER 6 (MR6) & MODE REGISTER 9 (MR9) (cont'd)




| OP11 | OP10 | OP9 | OP8 | OP7 | Pin Sub-Address |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | DQ0 |
| 0 | 0 | 0 | 0 | 1 | DQ1 |
| ... | | ... | | ... | ... |
| 0 | 0 | 1 | 1 | 0 | DQ6 |
| 0 | 0 | 1 | 1 | 1 | DQ7 |
| 0 | 1 | 0 | 0 | 0 | DBI0_n |
| 0 | 1 | 0 | 0 | 1 | RFU |
| 0 | 1 | 0 | 1 | 0 | Vendor Specific |
| 0 | 1 | 0 | 1 | 1 | |
| ... | | ... | | ... | |
| 0 | 1 | 1 | 1 | 0 | |
| 0 | 1 | 1 | 1 | 1 | Byte 0 |
| 1 | 0 | 0 | 0 | 0 | DQ8 |
| 1 | 0 | 0 | 0 | 1 | DQ9 |
| ... | | ... | | ... | ... |
| 1 | 0 | 1 | 1 | 0 | DQ14 |
| 1 | 0 | 1 | 1 | 1 | DQ15 |
| 1 | 1 | 0 | 0 | 0 | DBI1_n |
| 1 | 1 | 0 | 0 | 1 | RFU |
| 1 | 1 | 0 | 1 | 0 | Vendor Specific |
| 1 | 1 | 0 | 1 | 1 | |
| ... | | ... | | ... | |
| 1 | 1 | 1 | 1 | 0 | |
| 1 | 1 | 1 | 1 | 1 | Byte 1 |

| OP3 | OP2 | OP1 | OP0 | DFE | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | Off | |
| 0 | 0 | 0 | 1 | +/- 0.5% VDDQ | +/- 7mV |
| 0 | 0 | 1 | 0 | +/- 1.0% VDDQ | +/- 14mV |
| ... | | ... | | ... | ... |
| 1 | 1 | 1 | 0 | +/- 7.0% VDDQ | +/- 98mV |
| 1 | 1 | 1 | 1 | +/- 7.5% VDDQ | +/- 105mV |

Note: voltages for DFE are approximated values.

**Figure 37 — Mode Register 9 (MR9) Definition**

### VREFD Level

The reference voltage for the DQ and DBI_n inputs ($V_{REFD}$) is generated internally, and separate $V_{REF}$ circuits are associated with each data input pin.

The VREFD level is linear with a total range of 96 steps and a nominal step size of 1/200 x $V_{DDQ}$ (or 0.5%) in a range from 0.490 x $V_{DDQ}$ to 0.965 x $V_{DDQ}$ as illustrated in Table 24 and Figure 39. The mid point of 0.725 x $V_{DDQ}$ has been set to match the ideal vertical data eye center with nominal 48 Ohms pullup and 40 Ohms pulldown driver strength (see MR1). With nominal 60 Ohms pullup and 40 Ohms pulldown driver strength the ideal vertical data eye center will be at 0.7 x $V_{DDQ}$, only 35mV lower. Both mid point settings are highlighted in Table 24.

## 6.7 MODE REGISTER 6 (MR6) & MODE REGISTER 9 (MR9) (cont'd)

The VREFD level must be set by programming bits OP[6:0] in MR6 individually for each pin; the associated pin sub-address bits in OP[11:7] select the pin. The sub-address field also provides encodings to program the same VREFD level for a byte (byte 0 or byte 1). MRS commands to program the VREFD level can be issued in random order. No valid VREFD level is defined for decimal steps 96 to 127 (OP[6:5] = 11).

The $V_{REFD}$ settling time $t_{VREFD}$ is a constant value for the device, and is referenced from the MRS command to when the 90% level of the delta between old and new $V_{REFD}$ voltage has been reached as illustrated in Figure 40.

Half VREFD mode enables the $V_{REFD}$ level to be adjusted when the DQ and DBI_n inputs operate without termination. When bit OP7 in MR7 is set to '1', a level of nominally 0.5 x $V_{DDQ}$ is generated. The maximum operating frequency for this mode is defined by $f_{CKVREFD2}$. Disabling Half VREFD mode restores the programmed VREFD level and DFE values. A Half VREFD mode reference voltage change requires $t_{VREFD2}$ to settle.

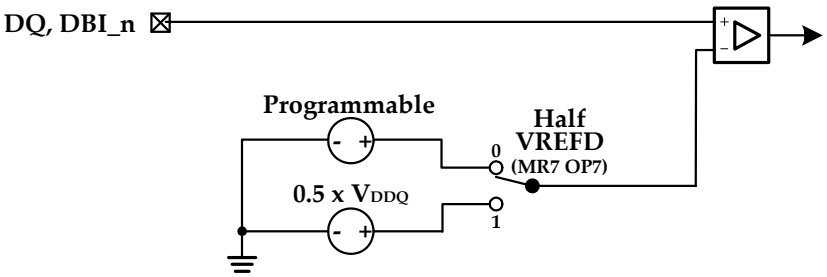



**Figure 38 — VREFD Options**

**Table 24 — VREFD Level**

| VREFD Level Code | | VREFD Level | | |
|---|---|---|---|---|
| MR6 OP[6:0] | Decimal | % $V_{DDQ}$ | Divider | $V_{REFD}$ [V] |
| 1 0 1 1 1 1 1 | 95 | 0.965 | 193 / 2 00 | 1.303 |
| 1 0 1 1 1 1 0 | 94 | 0.960 | 192 / 200 | 1.296 |
| 1 0 1 1 1 0 1 | 93 | 0.955 | 191 / 200 | 1.289 |
| ... | ... | ... | ... | ... |
| 0 1 1 0 0 0 0 | 48 | 0.730 | 146 / 200 | 0.986 |
| 0 1 0 1 1 1 1 | 47 | 0.725 | 145 / 200 | 0.979 |
| 0 1 0 1 1 1 0 | 46 | 0.720 | 144 / 200 | 0.972 |
| 0 1 0 1 1 0 1 | 45 | 0.715 | 143 / 200 | 0.965 |
| 0 1 0 1 1 0 0 | 44 | 0.710 | 142 / 200 | 0.959 |
| 0 1 0 1 0 1 1 | 43 | 0.705 | 141 / 200 | 0.952 |
| 0 1 0 1 0 1 0 | 42 | 0.700 | 140 / 200 | 0.945 |
| 0 1 0 1 0 0 1 | 41 | 0.695 | 139 / 200 | 0.938 |
| ... | ... | ... | ... | ... |
| 0 0 0 0 0 1 0 | 2 | 0.500 | 100 / 200 | 0.675 |
| 0 0 0 0 0 0 1 | 1 | 0.495 | 99 / 200 | 0.668 |
| 0 0 0 0 0 0 0 | 0 | 0490 | 98 / 200 | 0.662 |

## 6.7 MODE REGISTER 6 (MR6) & MODE REGISTER 9 (MR9) (cont'd)

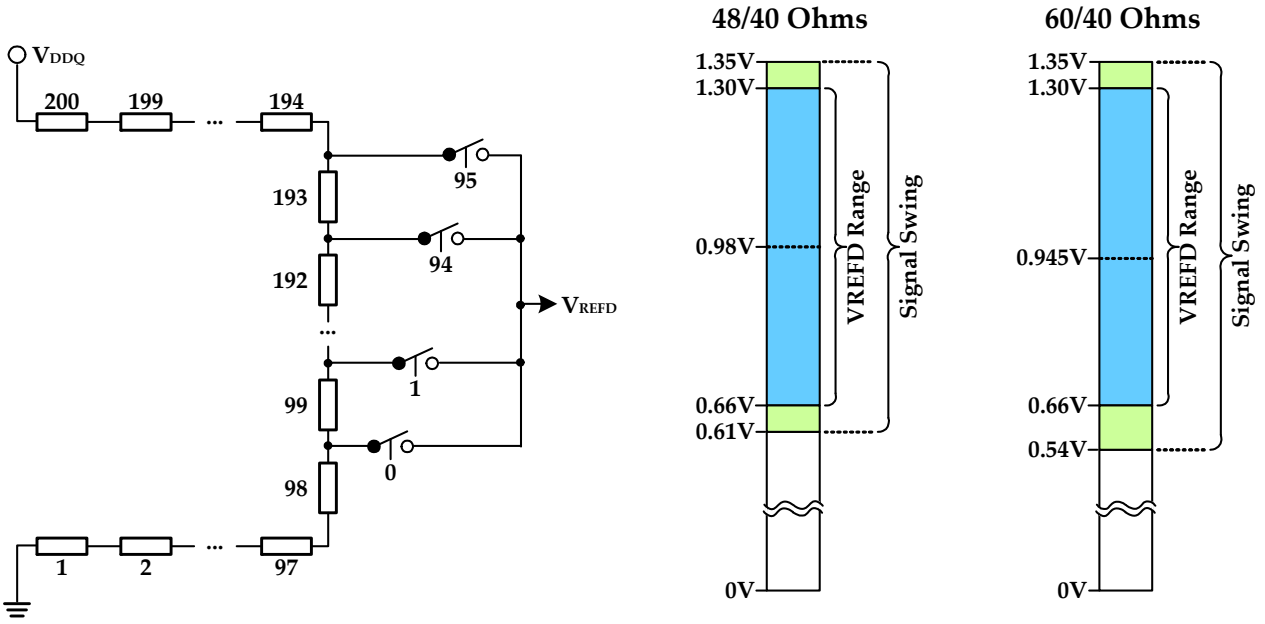



**Figure 39 — VREFD Circuit and Range**

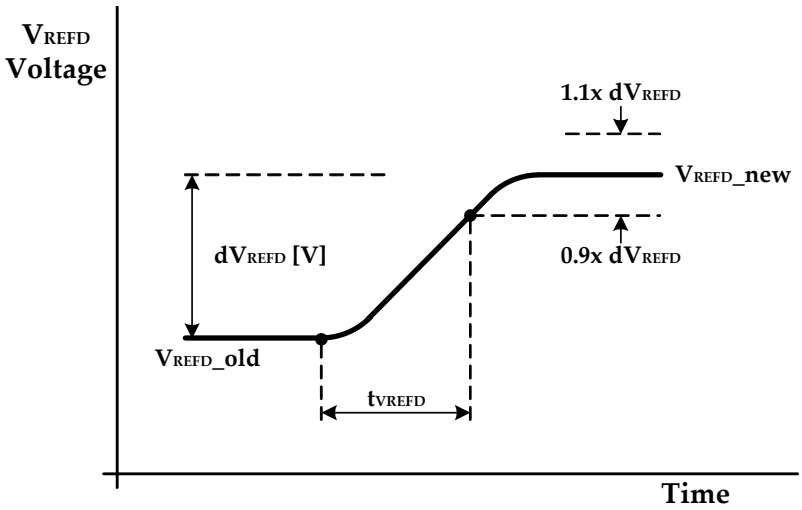



**Figure 40 — VREFD and DFE Settling Time**

## 6.7 MODE REGISTER 6 (MR6) & MODE REGISTER 9 (MR9) (cont'd)

### Decision Feedback Equalization (DFE)

The input data eye may be closed at the DRAM balls due to Inter System Interference (ISI) on the channel. GDDR6 provides means for improving (or opening up) the data eye at the receiver by the use of a 1-tap Decision Feedback Equalization (DFE).

DFE must be set by programming bits OP[3:0] in MR9 individually for each pin; the associated pin sub-address bits in OP[11:7] select the pin. The sub-address field also provides encodings to program the same DFE value for a byte (byte 0 or byte 1). MRS commands to program DFE can be issued in random order.

The DFE settings shall be equivalent to a positive and negative shift of the programmed VREFD level in steps of approximately +/-0.5% x $V_{DDQ}$ or +/-7mV as shown in Figure 37. The programmed VREFD level plus DFE voltage cannot be outside the range of the VREFD level supported in MR6 OP[6:0]. The actual DFE implementation is vendor specific. The $V_{REFD}$ settling time $t_{VREFD}$ must be satisfied for any DFE changes to settle.

### Transmit Equalizer (TX EQ)

The optional transmit equalizer function allows to program each data byte's output drivers to better match the system channel characteristics. The actual TX EQ implementation is vendor specific.

TX EQ uses opcodes in MR6 not used otherwise for programming VREFD as shown in Figure 36. The TX EQ option is selected by bits MR6 OP[1:0], and programmed for byte 0 with OP[11:7] = 01010 and for byte 1 with OP[11:7] = 11010.

## 6.8 MODE REGISTER 7 (MR7)

Mode Register 7 controls features like WCK2CK Alignment Point, Hibernate Self Refresh, Low Frequency mode, Auto Synchronization, DQ Preamble, Half VREFD, Half VREFC, VDD Range and DCC as shown in Figure 41. The register is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 0111.

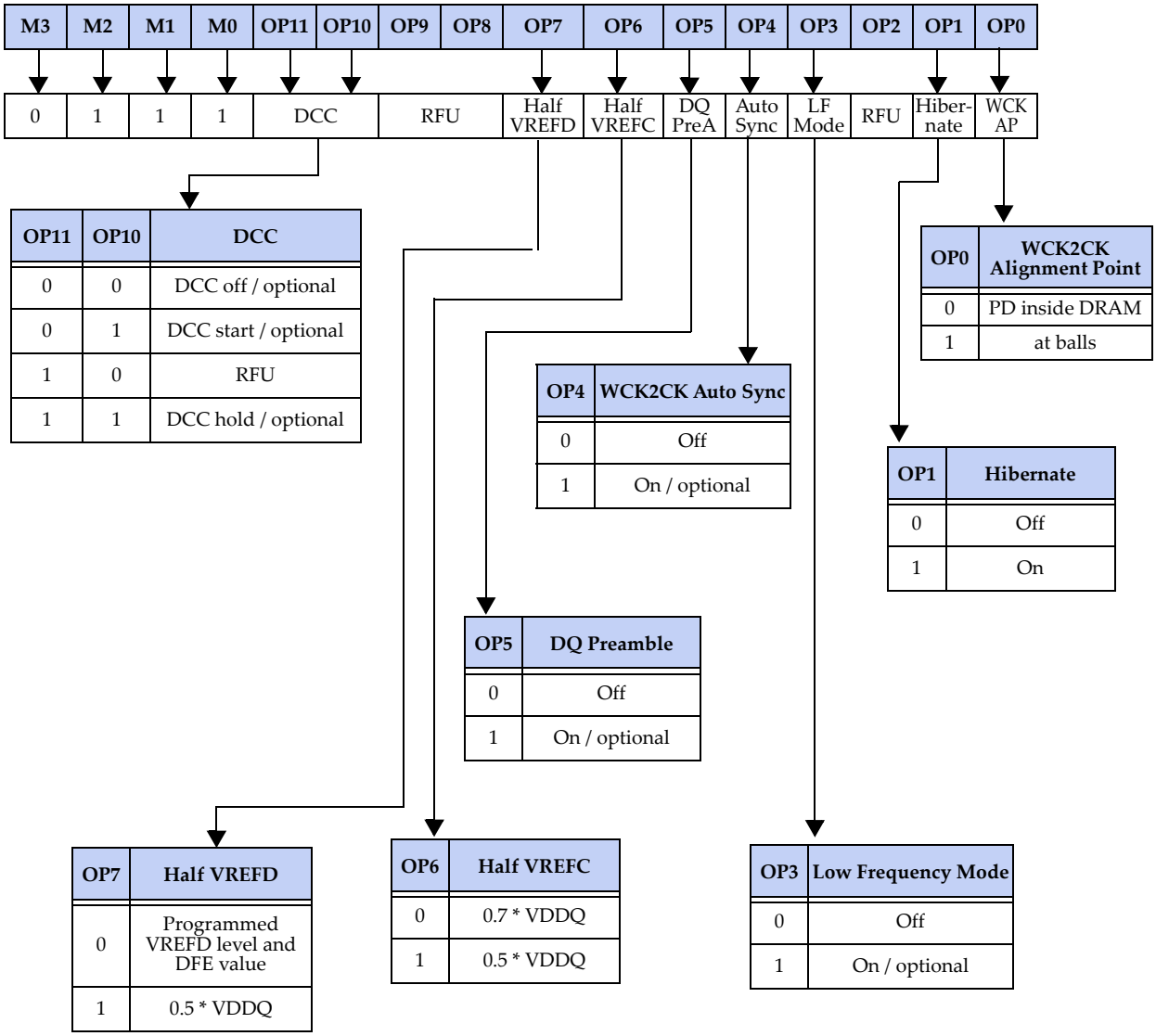



**Figure 41 — Mode Register 7 (MR7) Definition**

### WCK2CK Alignment Point (WCK AP)

OP3 defines the position of the alignment point between CK and WCK. When set to '0', the alignment point will be at the phase detector inside the device. When set to '1', the alignment point will be at the CK and WCK balls.

## 6.8 MODE REGISTER 7 (MR7) (cont'd)

### Hibernate Self Refresh

With OP1 set to '1', the device enters Hibernate Self Refresh mode with the next SELF REFRESH ENTRY command. The bit is self-clearing.

### Low Frequency Mode

When Low Frequency Mode is enabled by OP3, the power consumption of input receivers and clock trees is reduced. The maximum operating frequency for this low frequency mode is given in the vendor's datasheet. Low Frequency Mode is optional.

### WCK2CK Auto Synchronization

GDDR6 SGRAMs may optionally support a WCK2CK automatic synchronization mode that reduces the WCK2CK training to issuing two WCK2CK MRS for WCK2CK training entry and exit. This mode is controlled by OP4. For a detailed description see WCK2CK Auto Synchronization in the section entitled WCK2CK Training.

### DQ Preamble

When enabled by OP5, non-gapless READ bursts will be preceded by a fixed DQ preamble on the DQ and DBI_n signals of 8 U.I. duration. The programmed READ latency does not change when the DQ Preamble is enabled. The pattern is not encoded with RDBI, however, if RDBI is disabled, the DBI_n signals will not toggle and drive a HIGH. DQ Preamble is optional.

### Half VREFC and Half VREFD

See MODE REGISTER 6 (MR6) & MODE REGISTER 9 (MR9) for details on VREF features.

### Duty Cycle Correction (DCC)

OP[11:10] control the operation of the duty cycle corrector (DCC). The DCC can be used to cancel out a static duty cycle error on the WCK clocks. For more details see Duty Cycle Correction (DCC) in section entitled OPERATION.

## 6.9 MODE REGISTER 8 (MR8)

Mode Register 8 defines Clock Termination Override, CAL and CAH Termination, EDC Hi-Z, CK Auto Calibration, REFpb, CK Termination and extensions to Read latency (RLmrs) and Write Recovery (WR) as shown in Figure 42. The register is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 1000. OP4 of this register is initialized with '0's.

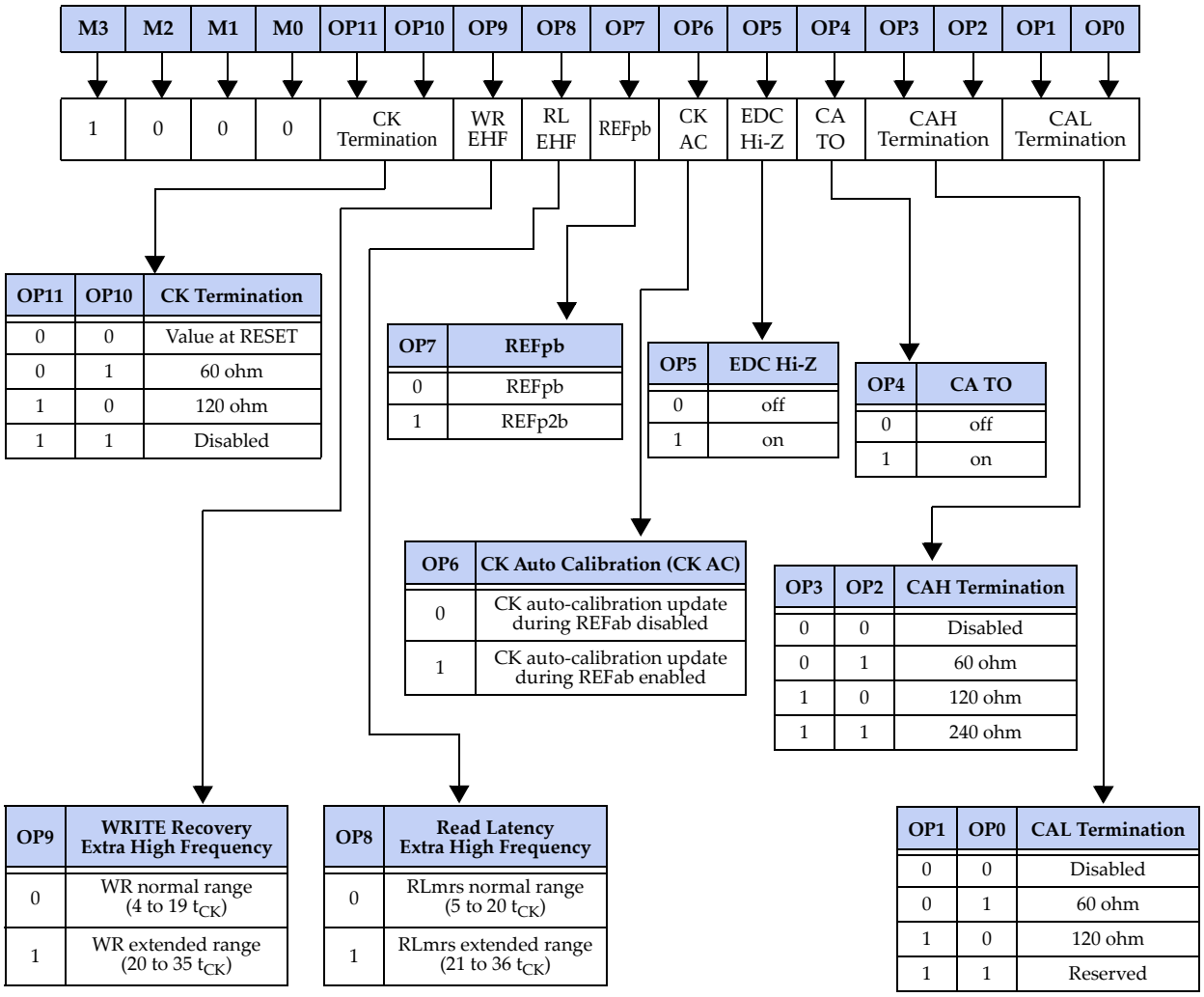



**Figure 42 — Mode Register 8 (MR8) Definition**

### READ Latency Extra High Frequency (RLEHF)

OP8 extends the RLmrs (READ Latency) field in MR0 from 4 bits to 5 bits. See Mode Register 0 for more details.

### WRITE Recovery Extra High Frequency (WREHF)

OP9 extends the WR (WRITE Recovery) field in MR0 from 4 bits to 5 bits. See Mode Register 0 for more details.

### EDC Hi-Z

With bit OP2 set to '1', the EDC signals are in Hi-Z state. The EDC Hi-Z function takes precedence over all other features that define the EDC signal's data pattern.

## 6.9 MODE REGISTER 8 (MR8) (cont'd)

### CK Auto Calibration (CK AC)

With bit OP6 set to '0', the update of the CK impedances by the calibration engine during REFab is disabled resulting in only the other impedances updated. With OP6 set to '1', the CK impedances are updated by the calibration engine during a REFab. If the calibration updates are off (MR1 OP6=1) or the CK Termination is disabled upon reset or by an MRS command that sets MR8 OP[11:10]=11, then this bit is ignored.

### REFpb

OP7 selects between Per-bank (REFpb) and Per-2-bank (REFp2b) Refresh when CA4 = LOW is set in a REFRESH command. See REFRESH and PER-BANK / PER-2-BANK REFRESH section for more details.

### CK Termination

OP[11:10] allows a change to the termination value of CK_t and CK_c that was determined at device initialization. The CK Termination register feature is optional. After any change to OP[11:10] from the default value of 00, the device operation cannot be guaranteed unless defined otherwise. If the feature is not supported by a device, the programming of the register will be ignored and the CK ODT value set during device initialization will remain valid as long as the device is powered or a subsequent chip reset.

The default setting ('00') selects the CK termination as determined by latching CA[5:4] on the rising edge of RESET_n. The CK termination can also be set to a value of 60 Ohm which is intended for a single loaded system, or 120 Ohm which is intended for double loaded configurations with two devices sharing a common CK. The values are the combined termination impedances from both channels as seen by the host. CK termination may also be turned off, for example, if external CK termination is used.

It is required to program the same CK termination value in both channels. In case the associated MRS commands are not issued at the same CK cycle to both channels, the ODT value as seen by the host cannot be guaranteed until a time $t_{MOD}$ has been met after the second MRS command. To allow a termination change to be completed, it is required to wait $t_{MOD}$ instead of $t_{MRD}$ after an MRS command that changes the CK termination and any subsequent MRS command.

### CA Termination Override (CA TO), CAH and CAL Termination

OP4 allows the CA Termination set during the power-up initialization sequence to be changed using CAH and CAL Termination.

OP[3:2] define the termination for the higher CA inputs, specifically CA[9:4], CABI_n and CKE_n. Values of 60 ohm, 120 ohm and 240 ohm are supported. The termination may also be disabled for the higher CA inputs.

OP[1:0] define the termination for the lower CA inputs, specifically CA[3:0]. Values of 60 ohm and 120 ohm are supported. The termination may also be disabled for the lower CA inputs.

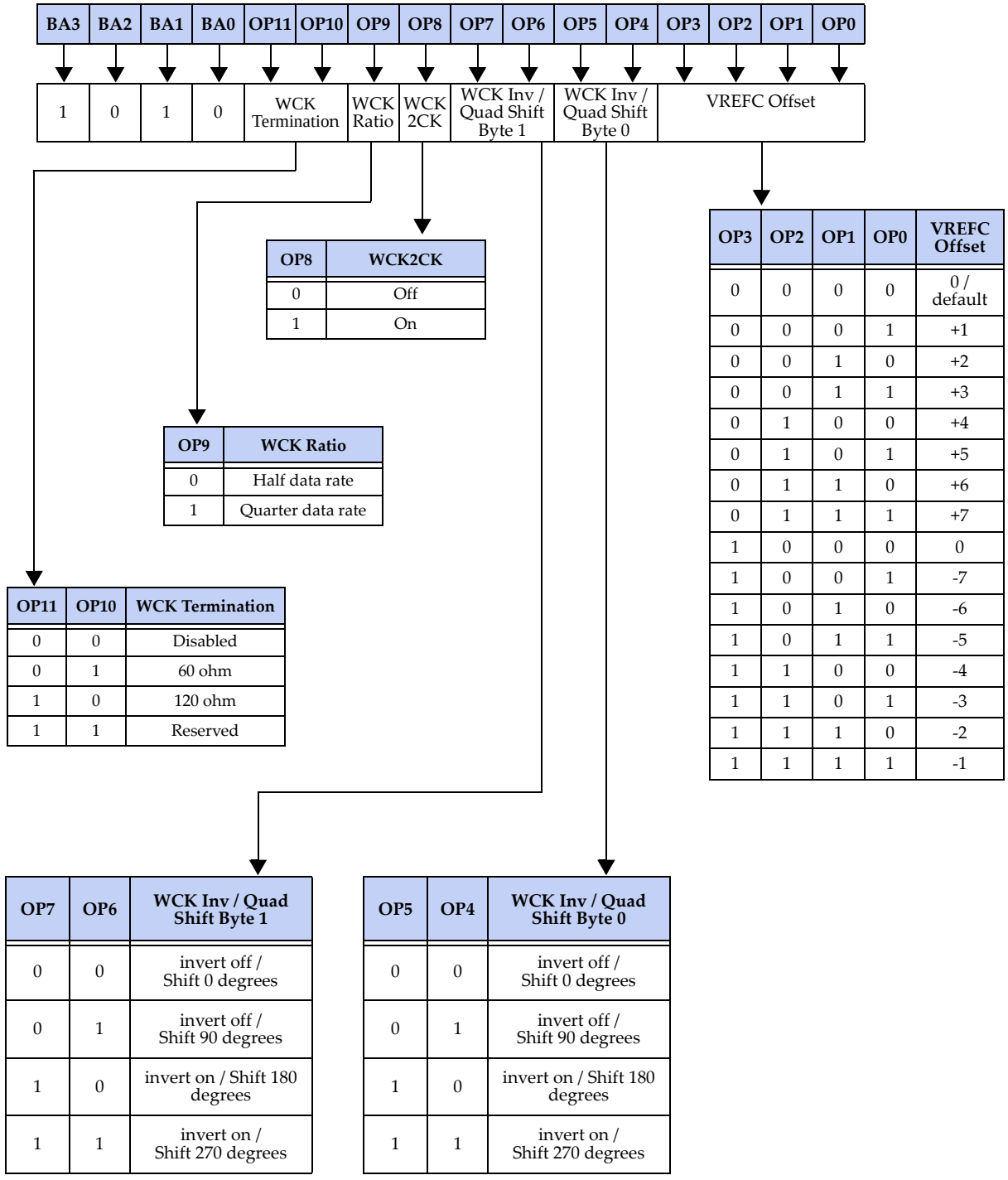
The use of the CA Termination Override, CAH and CAL Termination allow the CA Termination to be set individually for one, two or four loads on a common CA bus, depending on whether the DRAM is in two channel operation or PC mode, or x8 or x16 mode as well as other factors.

If CA Termination Override is set to '0' any change to the CAH and CAL Termination registers will be ignored. To allow a termination change to be completed, it is required to wait $t_{MOD}$ instead of $t_{MRD}$ after an MRS command that changes the CA termination and any subsequent MRS command.

## 6.10 MODE REGISTER 10 (MR10)

Mode Register 10 controls the VREFC Offset, WCK Termination, WCK Ratio, WCK2CK and WCK Inv / Quad Shift functions as shown in Figure 43. Mode Register 10 is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 1010. OP[11:10] of this register is initialized with '0's.

Mode Register 10




| OP8 | WCK2CK |
|---|---|
| 0 | Off |
| 1 | On |

| OP9 | WCK Ratio |
|---|---|
| 0 | Half data rate |
| 1 | Quarter data rate |

| OP11 | OP10 | WCK Termination |
|---|---|---|
| 0 | 0 | Disabled |
| 0 | 1 | 60 ohm |
| 1 | 0 | 120 ohm |
| 1 | 1 | Reserved |

| OP3 | OP2 | OP1 | OP0 | VREFC Offset |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 / default |
| 0 | 0 | 0 | 1 | +1 |
| 0 | 0 | 1 | 0 | +2 |
| 0 | 0 | 1 | 1 | +3 |
| 0 | 1 | 0 | 0 | +4 |
| 0 | 1 | 0 | 1 | +5 |
| 0 | 1 | 1 | 0 | +6 |
| 0 | 1 | 1 | 1 | +7 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | -7 |
| 1 | 0 | 1 | 0 | -6 |
| 1 | 0 | 1 | 1 | -5 |
| 1 | 1 | 0 | 0 | -4 |
| 1 | 1 | 0 | 1 | -3 |
| 1 | 1 | 1 | 0 | -2 |
| 1 | 1 | 1 | 1 | -1 |

| OP7 | OP6 | WCK Inv / Quad Shift Byte 1 |
|---|---|---|
| 0 | 0 | invert off / Shift 0 degrees |
| 0 | 1 | invert off / Shift 90 degrees |
| 1 | 0 | invert on / Shift 180 degrees |
| 1 | 1 | invert on / Shift 270 degrees |

| OP5 | OP4 | WCK Inv / Quad Shift Byte 0 |
|---|---|---|
| 0 | 0 | invert off / Shift 0 degrees |
| 0 | 1 | invert off / Shift 90 degrees |
| 1 | 0 | invert on / Shift 180 degrees |
| 1 | 1 | invert on / Shift 270 degrees |

**Figure 43 — Mode Register 10 (MR10) Definition**

## 6.10 MODE REGISTER 10 (MR10) (cont'd)

### WCK Inversion /Quad Shift

OP[7:4] control whether the internal phase of the WCK clock inputs after internal quarter data rate WCK shall be inverted, corresponding to a 2 U.I. phase shift for devices that use this method. Otherwise the bits control whether the internal phase of the WCK clock inputs after internal eighth data rate WCK shall be shifted for devices that use this method. The bits are used in conjunction with WCK2CK training mode. OP[7:6] on Channel A and OP[5:4] on Channel B are ignored for a device with WCK/word. The WCK Granularity field of the Vendor ID can be read to determine if device is designed with a WCK/byte or WCK/word.

### WCK2CK Training

OP8 (WCK2CK) enables and disables the WCK2CK alignment training. For details on this training sequence, see the section on TRAINING.

### WCK Ratio

OP9 selects between half data rate and quarter data rate WCK when the DRAM supports both ratios. If the DRAM does not support both ratios then WCK ratio is a do not care. WCK frequency ratios support can be determined from the WCK Ratio field of the Vendor ID1.

### WCK Termination

OP[11:10] define the (single ended) termination value for the on-die termination (ODT) for WCK.

The termination is disabled by default and can be set to a value of 60 Ohm or 120 Ohm depending on the system conditions. The WCK termination may also be turned off.

### Input Reference Voltage for CA and CABI_n Pins and VREFC Offset

The reference voltage for the CA bus (VREFC) can be supplied via the external VREFC pin or be generated internally as illustrated in Figure 44. The selection is made with the rising edge of RESET_n: the device selects internal VREFC with a default level of 0.7 x $V_{DDQ}$ when the VREFC pin is pulled LOW in the system; otherwise external VREFC is selected. GDDR6 also provides the capability to offset the internal VREFC by use of the VREFC offset bits OP[3:0]. The offset step values may be non-linear and will vary across DRAM vendors and across PVT.

Half VREFC mode enables the VREFC level to be adjusted when the CA bus operates without termination. When bit OP6 in MR7 is set to '1', a level of nominally 0.5 x $V_{DDQ}$ is generated. The maximum operating frequency for this mode is defined by $f_{CKVREFC2}$. A Half $V_{REFC}$ mode reference voltage change requires $t_{VREFC2}$ to settle. The programmable VREFC offset is not available in Half VREFC mode.

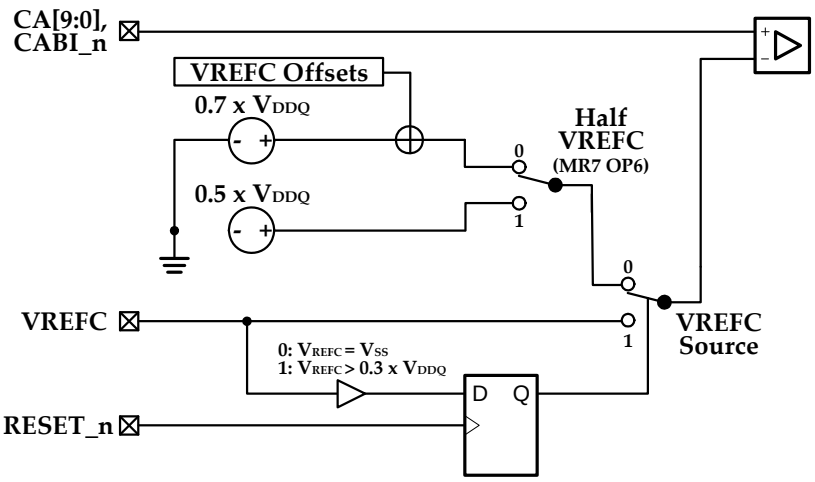
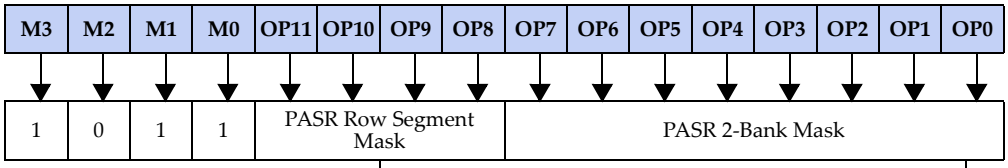



**Figure 44 — VREFC Options**

## 6.11 MODE REGISTER 11 (MR11)

Mode Register 11 controls operating modes such as PASR Row Segment Mask and PASR 2-Bank Mask as shown in Figure 45. The register is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 1011.




| OP11 | OP10 | OP9 | OP8 | PASR Row Segment Mask | 8 Gb | 12 Gb | 16 Gb | 24 Gb | 32 Gb |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | R[13:12] | R[13:12] | R[13:12] | TBD | TBD |
| X | X | X | 1 | Segment 0 | 00 | | | | |
| X | X | 1 | X | Segment 1 | 01 | | | | |
| X | 1 | X | X | Segment 2 | 10 | | | | |
| 1 | X | X | X | Segment 3 | 11 | | | | |

| OP[11:0] | PASR Mask (all bits) |
|---|---|
| 0 | Refresh enabled (= unmasked, default) |
| 1 | Refresh blocked (= masked) |
| X | "Don't Care" for the particular segment or banks |

| OP7 | OP6 | OP5 | OP4 | OP3 | OP2 | OP1 | OP0 | 2-Bank Mask |
|---|---|---|---|---|---|---|---|---|
| X | X | X | X | X | X | X | 1 | Bank 0 and 1 |
| X | X | X | X | X | X | 1 | X | Bank 2 and 3 |
| X | X | X | X | X | 1 | X | X | Bank 4 and 5 |
| X | X | X | X | 1 | X | X | X | Bank 6 and 7 |
| X | X | X | 1 | X | X | X | X | Bank 8 and 9 |
| X | X | 1 | X | X | X | X | X | Bank 10 and 11 |
| X | 1 | X | X | X | X | X | X | Bank 12 and 13 |
| 1 | X | X | X | X | X | X | X | Bank 14 and 15 |

**Figure 45 — Mode Register 11 (MR11) Definition**

## 6.12 MODE REGISTER 12 (MR12)

Mode Register 12 controls operating modes such as VDDQ off, PRBS, P2BR Address and also has register fields that are reserved for future features. The register is programmed via the MRS command with M[3:0] = 1100.

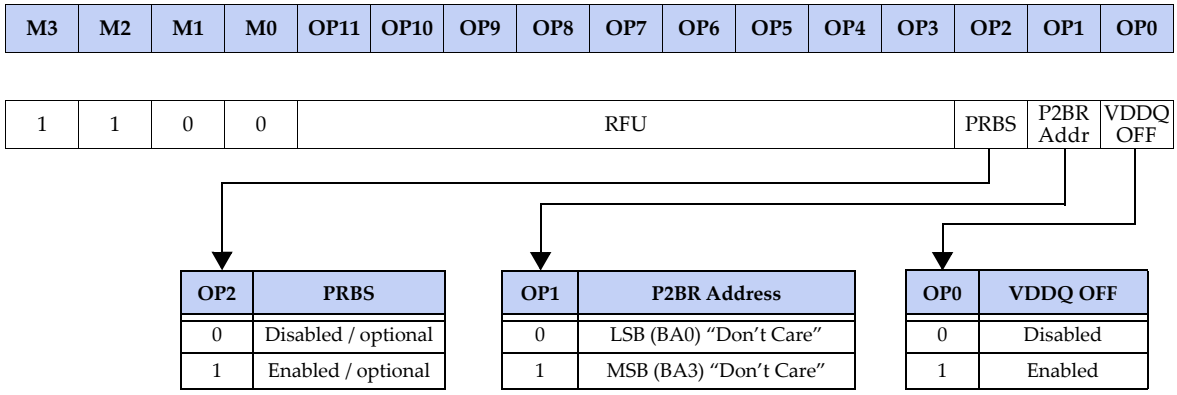



**Figure 46 — Mode Register 12 (MR12) Definition**

### VDDQ OFF

OP1 enables and disables the optional Hibernate Self Refresh with VDDQ Off feature. The VDDQ Off bit must be set on both channels before entering Hibernate Self Refresh. For details on the sequence, see the Hibernate Self Refresh section.

### P2BR Address

OP1 selects between LSB bank address, BA0, as "Don't Care" for REFp2b and MSB bank address, BA3, as "Don't Care" for REFp2b. P2BR Address is an optional register as MSB is an optional feature with only LSB "Don't Care" required to be supported. In the case the device only supports LSB "Don't Care" the programming of this register will be ignored. Support for MSB and the P2BR Address register is determined using the Vendor ID (ID2). For details see the PER-BANK REFRESH (REFpb) and PER-2-BANK REFRESH (REFp2b) Commands section.

### PRBS

OP2 controls the optional PRBS feature. When enabled, the EDC hold pattern will be replaced by a pseudo random bit stream.(PRBS7)

When the feature is enabled, the pseudo random bit stream is driven after $t_{MOD}$. The bit stream may be different any time the feature is enabled.

register bits MR4 OP[11,3:0] are ignored when PRBS is enabled.

The EDC HR bit (MR2 OP11) bit selects between a full and half data rate bit stream if the device supports both options.

## 6.13 MODE REGISTER 13 (MR13)

MR13 is reserved for vendor specific features. The register is programmed via the MRS command with M[3:0] = 1101.

| M3 | M2 | M1 | M0 | OP11 | OP10 | OP9 | OP8 | OP7 | OP6 | OP5 | OP4 | OP3 | OP2 | OP1 | OP0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | Reserved for Vendor Specific Features | | | | | | | | | | | |

**Figure 47 — Mode Register 13 (MR13) Definition**

## 6.14 MODE REGISTER 14 (MR14)

MR14 is reserved for vendor specific features. MR14 is programmed via the MRS command with M[3:0] = 1110.

| M3 | M2 | M1 | M0 | OP11 | OP10 | OP9 | OP8 | OP7 | OP6 | OP5 | OP4 | OP3 | OP2 | OP1 | OP0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | DCA | | | | | | | | Reserved for Vendor Specific Features | | | |

| OP11 | OP10 | OP09 | OP08 | DQ[7:0], EDC0, BDI0_A<br>DQ[15:8], EDC1, DBI1_B | |
|---|---|---|---|---|---|
| | | | | $0^o$-$90^o$ phase,<br>$180^o$-$270^o$ phase | $90^o$-$180^o$ phase<br>$270^o$-$0^o$ phase |
| 0 | 0 | 0 | 0 | OFF | OFF |
| 0 | 0 | 0 | 1 | +1 | -1 |
| 0 | 0 | 1 | 0 | +2 | -2 |
| 0 | 0 | 1 | 1 | +3 | -3 |
| 0 | 1 | 0 | 0 | +4 | -4 |
| 0 | 1 | 0 | 1 | +5 | -5 |
| 0 | 1 | 1 | 0 | +6 | -6 |
| 0 | 1 | 1 | 1 | +7 | -7 |
| 1 | 0 | 0 | 0 | -7 | +7 |
| 1 | 0 | 0 | 1 | -6 | +6 |
| 1 | 0 | 1 | 0 | -5 | +5 |
| 1 | 0 | 1 | 1 | -4 | +4 |
| 1 | 1 | 0 | 0 | -3 | +3 |
| 1 | 1 | 0 | 1 | -2 | +2 |
| 1 | 1 | 1 | 0 | -1 | +1 |
| 1 | 1 | 1 | 1 | OFF | OFF |

| OP7 | OP6 | OP5 | OP4 | DQ[15:8], EDC1, BDI1_A<br>DQ[7:0], EDC0, DBI0_B | |
|---|---|---|---|---|---|
| | | | | $0^o$-$90^o$ phase,<br>$180^o$-$270^o$ phase | $90^o$-$180^o$ phase<br>$270^o$-$0^o$ phase |
| 0 | 0 | 0 | 0 | OFF | OFF |
| 0 | 0 | 0 | 1 | +1 | -1 |
| 0 | 0 | 1 | 0 | +2 | -2 |
| 0 | 0 | 1 | 1 | +3 | -3 |
| 0 | 1 | 0 | 0 | +4 | -4 |
| 0 | 1 | 0 | 1 | +5 | -5 |
| 0 | 1 | 1 | 0 | +6 | -6 |
| 0 | 1 | 1 | 1 | +7 | -7 |
| 1 | 0 | 0 | 0 | -7 | +7 |
| 1 | 0 | 0 | 1 | -6 | +6 |
| 1 | 0 | 1 | 0 | -5 | +5 |
| 1 | 0 | 1 | 1 | -4 | +4 |
| 1 | 1 | 0 | 0 | -3 | +3 |
| 1 | 1 | 0 | 1 | -2 | +2 |
| 1 | 1 | 1 | 0 | -1 | +1 |
| 1 | 1 | 1 | 1 | OFF | OFF |

**Figure 48 — Mode Register 14 (MR14) Definition**

**Duty Cycle Adjuster (DCA)**

OP[11:4] control the operation of the duty cycle adjuster (DCA). The DCA can be used to cancel out a static duty cycle error on the WCK clocks. For more details see Duty Cycle Adjuster (DCA) in section entitled OPERATION.

## 6.15 MODE REGISTER 15 (MR15)

Mode Register 15 controls CA training mode (CADT) and access to Mode Registers 0 to 14 (MRE) as shown in Figure 49.

The register is programmed via the MODE REGISTER SET (MRS) command with M[3:0] = 1111.

Mode Register 15 is a special register that latches data on both edges but only evaluates the rising edge of CK. Therefore nothing is evaluated by the device on the falling edge of CK as shown in Figure 49. Increased CA setup and hold times are assumed to ensure the MRS command to this register is successful while CA training (CADT) has not taken place and the integrity of the DDR CA may not be guaranteed.

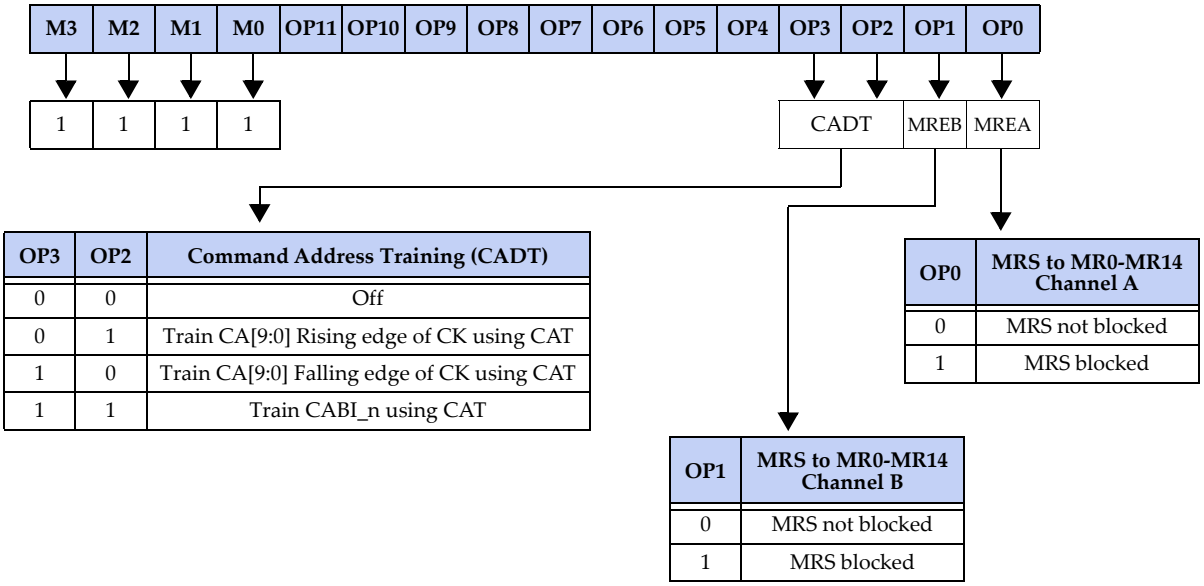



**Figure 49 — Mode Register 15 (MR15) Definition**

### Command Address Training (CADT)

Command Address training mode is controlled by OP[3:2]. See Command Address Training section for details.

### Mode Register 0-14 Enable (MRE)

MRE function allows for the individual configuration of two devices sharing a common CA bus. The function utilizes the fact that in such a configuration the CA inputs of channel A of the first device are shared with the CA inputs of channel B of the second device. The default is to issue MRS to both devices. OP0 is evaluated by channel A only and ignored by channel B. With OP0 = 1 any MRS command to registers MR0 to MR14 is blocked as shown in Figure 50. OP1 is evaluated by channel B only and ignored by channel A. With OP1 = 1 any MRS command to registers MR0 to MR14 is blocked as shown in Figure 50.

## 6.15 MODE REGISTER 15 (MR15) (cont'd)

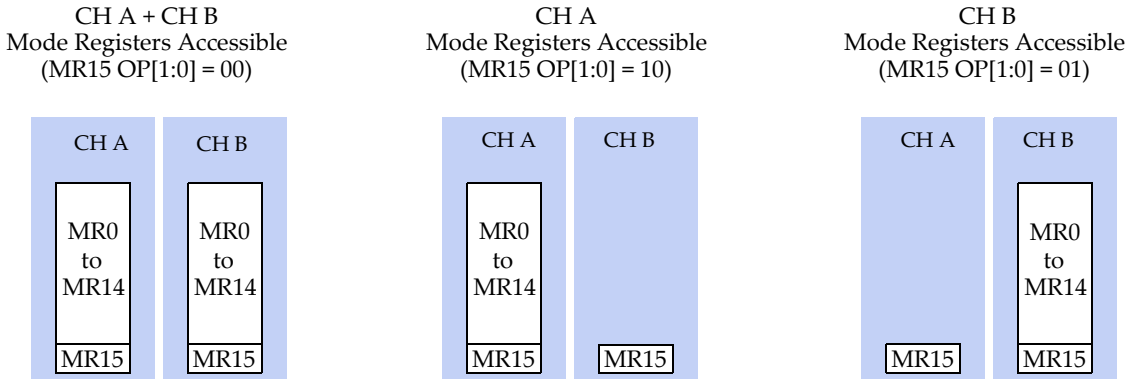



**Figure 50 — Mode Register Enable**

# 7. OPERATION

## 7.1 COMMANDS

**Table 25 — Truth Table - Commands**

| FUNCTION | SYMBOL | Clock Cycle | CKE_n | | CA10 | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 | NOTES |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Previous cycle (n-1) | Current cycle (n) | | | | | | | | | | | | |
| NO OPERATION (NOP) (1) | NOP | R | L | L | | H | H | V | V | V | V | V | V | V | V | 1, 10 |
| | | F | | | | H | H | V | V | V | V | V | V | V | V | |
| NO OPERATION (NOP) (2) | NOP | R | L | L | | H | H | V | V | V | V | V | V | V | V | 1, 10 |
| | | F | | | | H | L | V | V | V | V | V | V | V | V | |
| NO OPERATION (NOP) (3) | NOP | R | L | L | | H | L | V | V | V | V | V | V | V | V | 1, 10 |
| | | F | | | | H | H | V | V | V | V | V | V | V | V | |
| MODE REGISTER SET | MRS | R | L | L | | H | L | M3 | M2 | M1 | M0 | OP3 | OP2 | OP1 | OP0 | 1, 2, 3 |
| | | F | | | | H | L | OP11 | OP10 | OP9 | OP8 | OP7 | OP6 | OP5 | OP4 | |
| ACTIVATE (Select bank & activate row) | ACT | R | L | L | | L | V (R14) | BA3 | BA2 | BA1 | BA0 | R3 | R2 | R1 | R0 | 1, 2, 4 |
| | | F | | | | R13 | R12 | R11 | R10 | R9 | R8 | R7 | R6 | R5 | R4 | |
| READ (Select bank, column & start burst) | RD | R | L | L | | H | H | BA3 | BA2 | BA1 | BA0 | C3 | C2 | C1 | C0 | 1, 2, 5, 6 |
| | | F | | | | L | H | L | L | V | L | CE | C6 | C5 | C4 | |
| READ with Autoprecharge | RDA | R | L | L | | H | H | BA3 | BA2 | BA1 | BA0 | C3 | C2 | C1 | C0 | 1, 2, 5, 6 |
| | | F | | | | L | H | L | L | V | H | CE | C6 | C5 | C4 | |
| Load FIFO | LDFF | R | L | L | | H | H | B3 | B2 | B1 | B0 | D3 | D2 | D1 | D0 | 1, 2, 8 |
| | | F | | | | L | H | H | L | D9 | D8 | D7 | D6 | D5 | D4 | |
| READ Training | RDTR | R | L | L | | H | H | V | V | V | V | V | V | V | V | 1, 2, 6 |
| | | F | | | | L | H | H | H | V | L | CE | V | V | V | |
| WRITE without Mask (Select bank, column, & start burst) | WOM | R | L | L | | H | H | BA3 | BA2 | BA1 | BA0 | C3 | C2 | C1 | C0 | 1, 2, 5, 6 |
| | | F | | | | L | L | L | L | V | L | CE | C6 | C5 | C4 | |
| WRITE without Mask with Autoprecharge | WOMA | R | L | L | | H | H | BA3 | BA2 | BA1 | BA0 | C3 | C2 | C1 | C0 | 1, 2, 5, 6 |
| | | F | | | | L | L | L | L | V | H | CE | C6 | C5 | C4 | |
| WRITE with double-byte mask (WDM) | WDM | Cycle 1 R | L | L | RESERVED | H | H | BA3 | BA2 | BA1 | BA0 | C3 | C2 | C1 | C0 | 1, 2, 5, 6 |
| | | F | | | | L | L | H | L | V | L | CE | C6 | C5 | C4 | |
| | | Cycle 2 R | | | | H | H | BST7 | BST6 | BST5 | BST4 | BST3 | BST2 | BST1 | BST0 | |
| | | F | | | | H | H | BST15 | BST14 | BST13 | BST12 | BST11 | BST10 | BST9 | BST8 | |
| WRITE with double-byte mask with Autoprecharge | WDMA | Cycle 1 R | L | L | | H | H | BA3 | BA2 | BA1 | BA0 | C3 | C2 | C1 | C0 | 1, 2, 5, 6 |
| | | F | | | | L | L | H | L | V | H | CE | C6 | C5 | C4 | |
| | | Cycle 2 R | | | | H | H | BST7 | BST6 | BST5 | BST4 | BST3 | BST2 | BST1 | BST0 | |
| | | F | | | | H | H | BST15 | BST14 | BST13 | BST12 | BST11 | BST10 | BST9 | BST8 | |
| WRITE with single-byte mask | WSM | Cycle 1 R | L | L | | H | H | BA3 | BA2 | BA1 | BA0 | C3 | C2 | C1 | C0 | 1, 2, 5, 6 |
| | | F | | | | L | L | L | H | V | L | CE | C6 | C5 | C4 | |
| | | Cycle 2 R | | | | H | H | Byte 0 BST7 | Byte 0 BST6 | Byte 0 BST5 | Byte 0 BST4 | Byte 0 BST3 | Byte 0 BST2 | Byte 0 BST1 | Byte 0 BST0 | |
| | | F | | | | H | H | Byte 0 BST15 | Byte 0 BST14 | Byte 0 BST13 | Byte 0 BST12 | Byte 0 BST11 | Byte 0 BST10 | Byte 0 BST9 | Byte 0 BST8 | |
| | | Cycle 3 R | | | | H | H | Byte 1 BST7 | Byte 1 BST6 | Byte 1 BST5 | Byte 1 BST4 | Byte 1 BST3 | Byte 1 BST2 | Byte 1 BST1 | Byte 1 BST0 | |
| | | F | | | | H | H | Byte 1 BST15 | Byte 1 BST14 | Byte 1 BST13 | Byte 1 BST12 | Byte 1 BST11 | Byte 1 BST10 | Byte 1 BST9 | Byte 1 BST8 | |
| WRITE with single-byte mask with Autoprecharge | WSMA | Cycle 1 R | L | L | | H | H | BA3 | BA2 | BA1 | BA0 | C3 | C2 | C1 | C0 | 1, 2, 5, 6 |
| | | F | | | | L | L | L | H | V | H | CE | C6 | C5 | C4 | |
| | | Cycle 2 R | | | | H | H | Byte 0 BST7 | Byte 0 BST6 | Byte 0 BST5 | Byte 0 BST4 | Byte 0 BST3 | Byte 0 BST2 | Byte 0 BST1 | Byte 0 BST0 | |
| | | F | | | | H | H | Byte 0 BST15 | Byte 0 BST14 | Byte 0 BST13 | Byte 0 BST12 | Byte 0 BST11 | Byte 0 BST10 | Byte 0 BST9 | Byte 0 BST8 | |
| | | Cycle 3 R | | | | H | H | Byte 1 BST7 | Byte 1 BST6 | Byte 1 BST5 | Byte 1 BST4 | Byte 1 BST3 | Byte 1 BST2 | Byte 1 BST1 | Byte 1 BST0 | |
| | | F | | | | H | H | Byte 1 BST15 | Byte 1 BST14 | Byte 1 BST13 | Byte 1 BST12 | Byte 1 BST11 | Byte 1 BST10 | Byte 1 BST9 | Byte 1 BST8 | |
| WRITE Training | WRTR | R | L | L | | H | H | V | V | V | V | V | V | V | V | 1, 2, 6 |
| | | F | | | | L | L | H | H | V | L | CE | V | V | V | |
| PRECHARGE (per bank) (Deactivate row in a bank) | PREpb | R | L | L | | H | L | BA3 | BA2 | BA1 | BA0 | V | V | V | V | 1, 2, 9 |
| | | F | | | | L | L | V | V | V | L | V | V | V | V | |
| PRECHARGE (all banks) (Deactivate row in all banks) | PREab | R | L | L | | H | L | V | V | V | V | V | V | V | V | 1, 2 |
| | | F | | | | L | L | V | V | V | H | V | V | V | V | |

## Table 25 — Truth Table - Commands

| FUNCTION | SYMBOL | Clock Cycle | CKE_n | | CA10 | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 | NOTES |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Previous cycle (n-1) | Current cycle (n) | | | | | | | | | | | | |
| PER-BANK / PER-2 BANK REFRESH | REFpb / REFp2b | R | L | L | RESERVED | H | L | BA3 | BA2 | BA1 | BA0 | V | V | V | V | 1, 2, 7, 9 |
| | | F | | | | L | H | V | V | V | L | V | V | V | V | |
| REFRESH (all banks) | REFab | R | L | L | | H | L | V | V | V | V | V | V | V | V | 1, 2, 7 |
| | | F | | | | L | H | V | V | V | H | V | V | V | V | |
| POWER DOWN ENTRY | PDE | R | L | H | | H | H | V | V | V | V | V | V | V | V | 1, 2 |
| | | F | | | | H | H | V | V | V | V | V | V | V | V | |
| POWER DOWN EXIT | PDX | R | H | L | | H | H | V | V | V | V | V | V | V | V | 1, 2 |
| | | F | | | | H | H | V | V | V | V | V | V | V | V | |
| SELF REFRESH ENTRY | SRE | R | L | H | | H | L | V | V | V | V | V | V | V | V | 1, 2, 7 |
| | | F | | | | L | H | V | V | V | V | V | V | V | V | |
| SELF REFRESH EXIT | SRX | R | H | L | | H | H | V | V | V | V | V | V | V | V | 1, 2 |
| | | F | | | | H | H | V | V | V | V | V | V | V | V | |
| COMMAND ADDRESS TRAINING CAPTURE | CAT | R | L | H | | V | V | V | V | V | V | V | V | V | V | 1, 2 |
| | | F | | | | V | V | V | V | V | V | V | V | V | V | |

NOTE 1 H = Logic HIGH Level; L = Logic LOW Level; V = Valid, signal may be H or L, but not floating; R, F = Rising, Falling CK clock edge
NOTE 2 Values shown for CA[9:0] are logical values; the physical values are inverted when Command/Address Bus Inversion (CABI) is enabled and CABI_n=L.
NOTE 3 M[3:0] provide the Mode Register address (MRA), OP[11:0] the opcode to be loaded.
NOTE 4 BA[3:0] provide the bank address, R[(14), 13:0] provide the row address.
NOTE 5 BA[3:0] provide the bank address, C[6:0] provide the column address; no sub-word addressing within a burst of 16. BST[15:0] provide the write data mask for each burst position with WDM(A) and WSM(A) commands.
NOTE 6 CE (Channel Enable) is intended for PC mode. The command is active when CE = H. When CE = L the array access is suppressed. In two channel mode CE shall be driven HIGH.
NOTE 7 The command is Refresh (all banks) or Per-Bank / Per-2-Bank Refresh when CKE_n(n) = L and Self Refresh Entry when CKE_n(n) = H. The second bit of CA4 selects between Refresh (all banks) and Per-Bank / Per-2-Bank Refresh.
NOTE 8 B[3:0] select the burst position, and D[9:0] provide the data.
NOTE 9 BA[3:0] provide the bank address.
NOTE 10 All three encodings perform the same NOP. NOP (2) and NOP (3) encodings are only allowed during CA Training.

## 7.2 COMMAND, ADDRESS and WRITE DATA INPUT TIMINGS

Figure 51 and Figure 52 illustrate the timings associated with the Command and Address input as well as Data input.

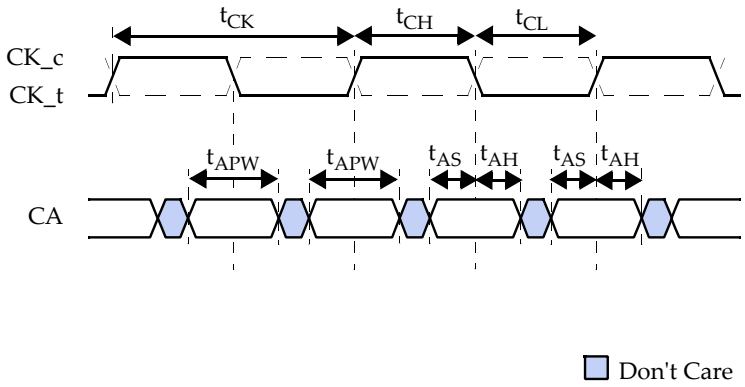



**Figure 51 — Command and Address Input Timings**

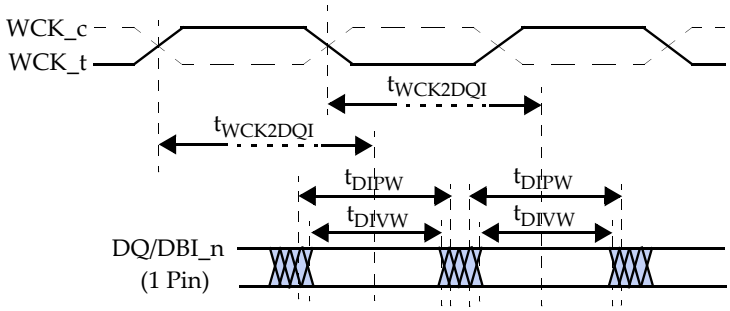



**Figure 52 — Data Input Timings**

## 7.3 NO OPERATION (NOP)

The NO OPERATION (NOP) command is used to instruct the device to perform a NOP. This prevents unwanted commands from being registered during idle or wait states. Operations already in progress are not affected. GDDR6 defines three encodings for the NOP command, and all three encodings perform the same NOP, NOP (2) and NOP (3) encodings are only allowed during CA Training.

## 7.4 MODE REGISTER SET

The MODE REGISTER SET command is used to load the Mode Registers of the device. M[3:0] select the Mode Register, and OP[11:0] determine the opcode to be loaded. See MODE REGISTER for a register definition.

The MODE REGISTER SET (MRS) command can only be issued when all banks are idle and no bursts are in progress. The MRS command cycle time, $t_{MRD}$, is required to complete the write operation to the Mode Register and is the minimum time required between two MRS commands. For the CK Termination, CA Termination and CABI registers, $t_{MOD}$ is required after changing the register and a subsequent MRS command. The MRS command to non-MRS command delay, $t_{MOD}$, is required by the device to update the features, and is the minimum time required from an MRS command to a non-MRS command excluding NOP.

The value of $t_{MRD}$ is less than $t_{MOD}$. The use of the two timing parameter $t_{MRD}$ and $t_{MOD}$ allows the controller to schedule a series of MRS commands more efficiently than with a single timing parameter. However, the same timing is achieved when only parameter $t_{MOD}$ is taken into account.

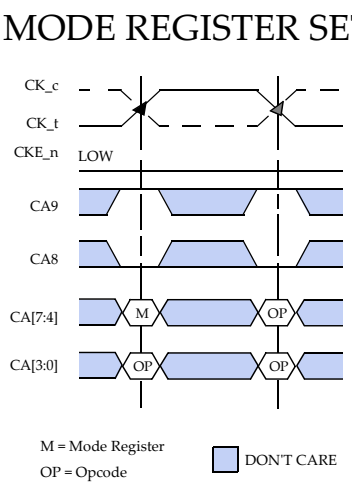
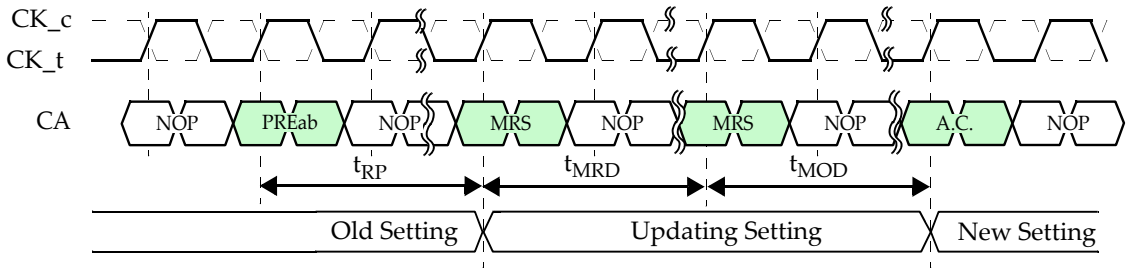



**Figure 53 — MRS Command**




A.C. = any command allowed in bank idle state

**Figure 54 — Mode Register Set Timings**

## 7.5 ROW ACTIVATION

Before any READ or WRITE commands can be issued to a bank in the device, a row in that bank must be “opened”. This is accomplished by the ACTIVATE command (see Figure 55): BA[3:0] select the bank, and R[13:0] or R[14:0] select the row to be activated. Once a row is open, a READ or WRITE command could be issued to that row, subject to the $t_{RCD}$ specification.

A subsequent ACTIVATE command to another row in the same bank can only be issued after the previous row has been closed (precharged). The minimum time interval between two successive ACTIVATE commands on the same bank is defined by $t_{RC}$. A minimum time, $t_{RAS}$, must have elapsed between opening and closing a row.

A subsequent ACTIVATE command to another bank can be issued while the first bank is being accessed, which results in a reduction of total row-access overhead. The minimum time interval between two successive ACTIVATE commands on different banks to different bank groups is defined by $t_{RRDS}$. With bank groups enabled, the minimum time interval between two successive ACTIVATE commands to different banks in the same bank group is defined by $t_{RRDL}$. In all other cases the interval is defined by $t_{RRDS}$. Figure 56 shows the $t_{RCD}$ and $t_{RRD}$ definition.

The row remains active until a PRECHARGE command (or READ or WRITE command with Auto Precharge) is issued to the bank.

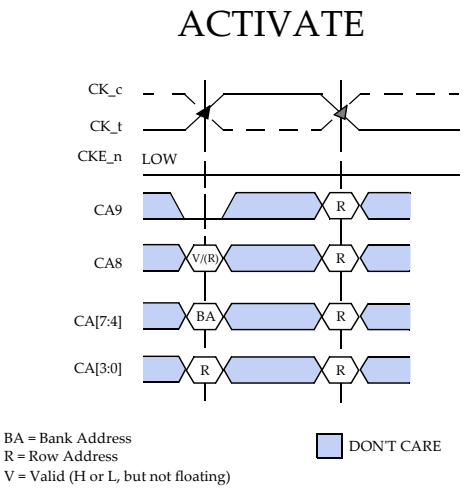
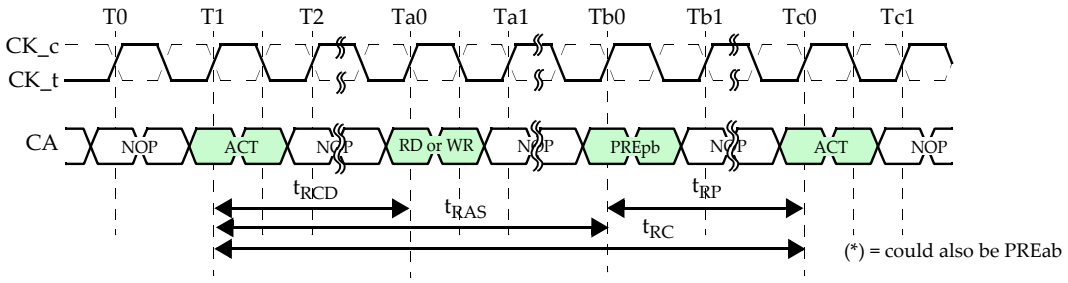



**Figure 55 — ACTIVATE Command**



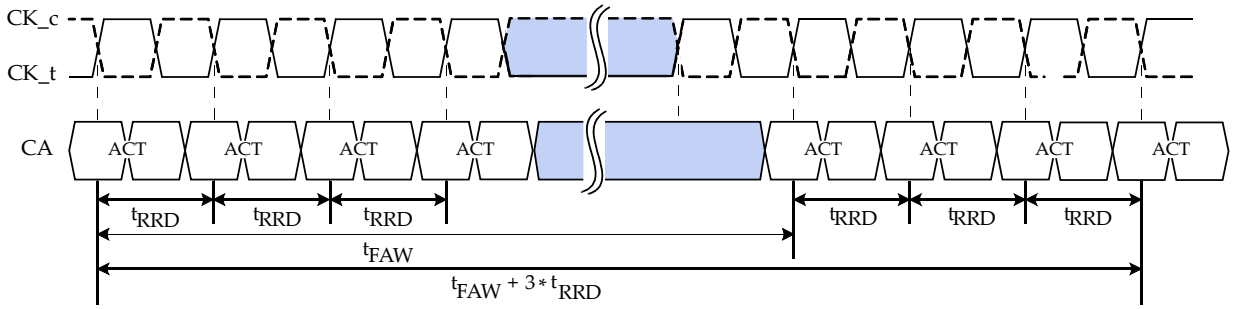

$t_{RCD}$ = $t_{RCDRD}$, $t_{RCDWR}$, $t_{RCDRTR}$, $t_{RCDWTR}$ or $t_{RCDLTR}$, depending on command

**Figure 56 — Bank Activation Command Cycle**

## 7.6 BANK RESTRICTIONS

There may be a need to limit the number of activates in a rolling window to ensure that the instantaneous current supplying capability of the devices is not exceeded. To reflect the short term capability of the device current supply, the parameter $t_{FAW}$ (four activate window) is defined. No more than 4 banks may be activated in a rolling $t_{FAW}$ window. Converting to clocks is done by dividing $t_{FAW}$ (ns) by $t_{CK}$ (ns) and rounding up to next integer value. As an example of the rolling window, if ($t_{FAW}/t_{CK}$) rounds up to 10 clocks, and an ACTIVATE command is issued at clock N, no more than three further ACTIVATE commands may be issued at clocks N+1 through N+9 as illustrated in Figure 57.

It is preferable that GDDR6 SGRAMs have no rolling activation window restrictions ($t_{FAW}$ = 4 * $t_{RRD}$).




$t_{RRD}$ = $t_{RRDL}$ or $t_{RRDS}$ depending on Bank Groups on/off setting and accessed banks

**Figure 57 — $t_{RRD}$ and $t_{FAW}$**

## 7.7 WRITE (WOM)

WRITE bursts are initiated with a WRITE command as shown in Figure 58. The bank and column addresses are provided with the WRITE command and auto precharge is either enabled or disabled for that access with the AP bit (CA4 input falling edge of CK). If auto precharge is enabled, the row being accessed is precharged after $t_{WR}$(min) has been met or after the number of clock cycles programmed in the WR field of MR0, depending on the implementation choice per DRAM vendor. The length of the burst initiated with a WRITE command is sixteen and the column address is unique for this burst of sixteen. There is no interruption nor truncation of WRITE bursts. No write data is received, the array write access is suppressed, and the EDC hold pattern is driven instead of a CRC burst when the CE bit is LOW. In two channel mode CE shall be driven HIGH.

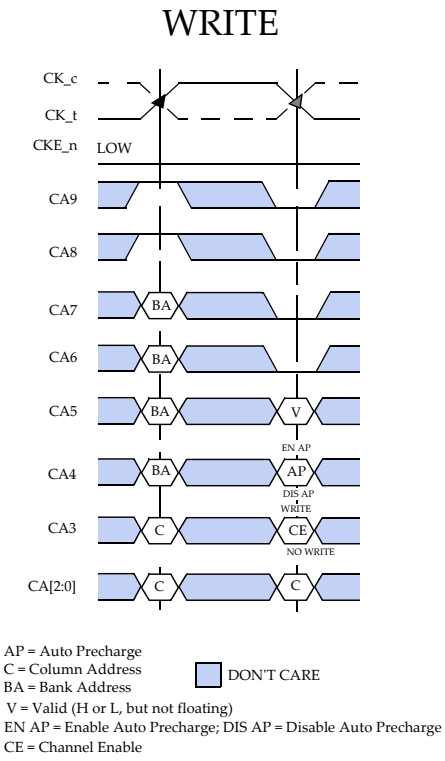



**Figure 58 — WRITE Command**

WRITE timings are shown with DDR WCK ratio in Figure 59. During WRITE bursts, the first valid data-in element must be available at the input latch after the Write Latency (WL). The Write Latency is defined as WLmrs * $t_{CK}$ + $t_{WCK2CKPIN}$ + $t_{WCK2CK}$ + $t_{WCK2DQI}$, where WLmrs is the number of clock cycles programed in MR0, $t_{WCK2CKPIN}$ is the phase offset between WCK and CK at the balls when phase aligned at phase detector, $t_{WCK2CK}$ is the alignment error between WCK and CK at the phase detector, and $t_{WCK2DQI}$ is the WCK to DQ/DBI_n offset as measured at the DRAM balls to ensure concurrent arrival at the latch. The total delay is relative to the data eye center averaged over one byte if WCK/byte or double byte if WCK/word. The maximum skew within a byte is defined by $t_{DQDQI}$.

The data input valid window, $t_{DIVW}$, defines the time region when input data must be valid for reliable data capture at the receiver for any one worst-case channel. It accounts for jitter between data and clock at the latching point introduced in the path between the DRAM pads and the latching point. Any additional jitter introduced into the source signals (i.e., within the system before the DRAM pad) must be accounted for in

## 7.7 WRITE (WOM) (cont'd)

the final timing budget together with the chosen PLL/DLL mode and bandwidth. $t_{DIVW}$ is for DRAM design only and valid on the silicon die. It is not intended to be measured. $t_{DIVW}$ is defined for the PLL/DLL off and on mode separately. In the case of PLL on, $t_{DIVW}$ must be specified for each supported bandwidth. In general $t_{DIVW}$ is smaller than $t_{DIPW}$.

The data input pulse width, $t_{DIPW}$, defines the minimum positive or negative input pulse width for any one worst-case channel required for proper propagation of an external signal to the receiver. $t_{DIPW}$ is for DRAM design only and valid on the silicon die. It is not intended to be measured. $t_{DIPW}$ is independent of the PLL/DLL mode. In general $t_{DIPW}$ is larger than $t_{DIVW}$.

Upon completion of a burst, assuming no other WRITE data is expected on the bus the DQ and DBI_n signals will be driven according to the ODT state. Any additional input data will be ignored. Data for any WRITE burst may not be truncated with a subsequent WRITE command.

Data from any WRITE burst may be concatenated with data from a subsequent WRITE command. A continuous flow of data can be maintained. The first data element from the new burst follows the last element of a completed burst. The new WRITE command should be issued after the previous WRITE command according to the $t_{CCD}$ timing. If that WRITE command is to another bank then an ACTIVATE command must precede the WRITE command and $t_{RCDWR}$ also must be met.

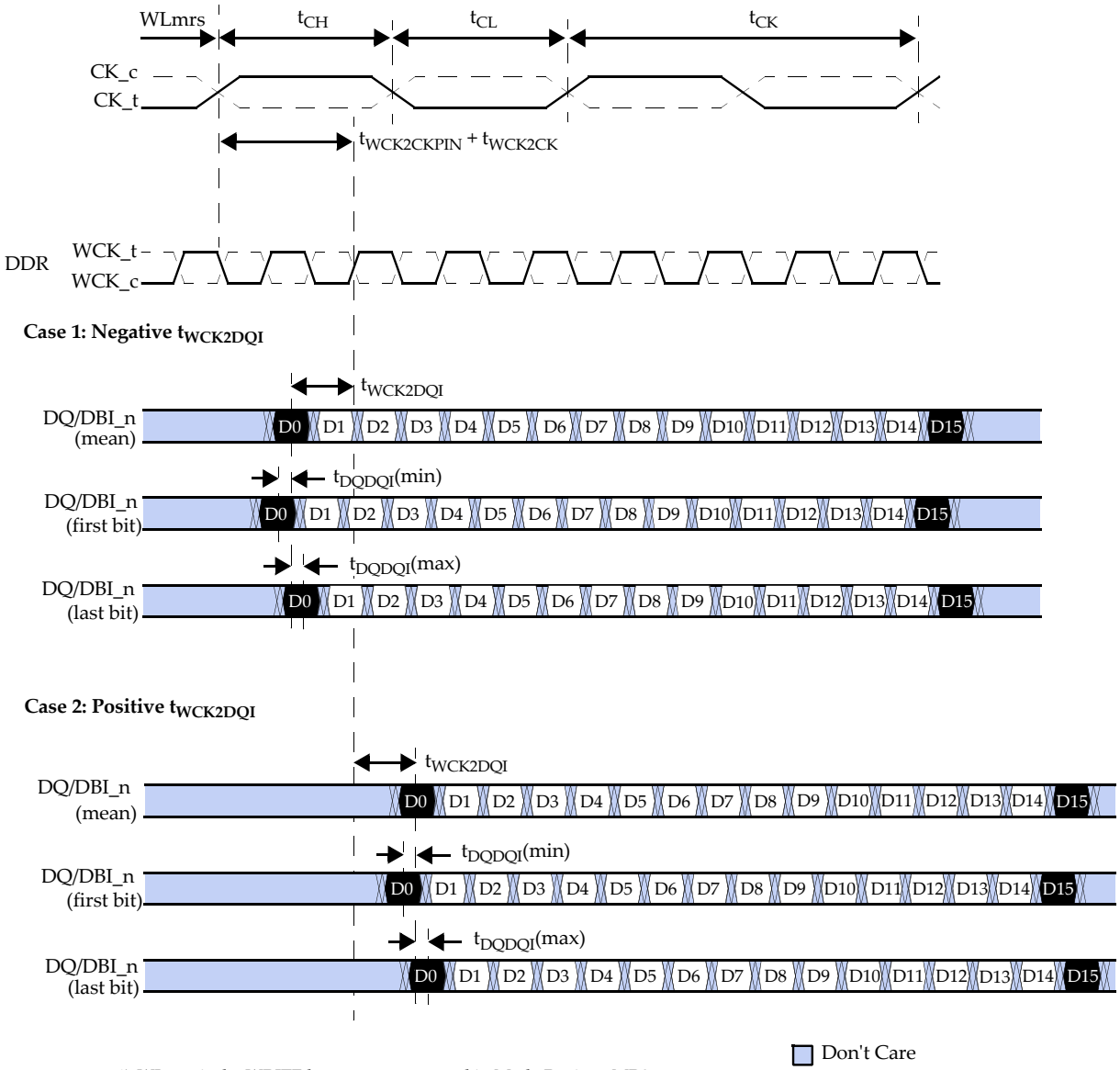
A READ can be issued any time after a WRITE command as long as the internal turn around time $t_{WTR}$ is met. If that READ command is to another bank, then an ACTIVATE command must precede the READ command and $t_{RCDRD}$ also must be met.

A PRECHARGE can also be issued after $t_{WR}$ has been met. After the PRECHARGE command, a subsequent command to the same bank cannot be issued until $t_{RP}$ is met.

The data inversion flag is received on DBI_n. If DBI_n is LOW the data is stored after inversion and not inverted if DBI_n is HIGH. WRITE Data Inversion can be enabled (OP9 = 0) or disabled (OP9 = 1) using WDBI in MR1.

When enabled by the WRCRC flag in MR4, EDC data are returned to the controller with a latency of (WLmrs + CRCWL) * $t_{CK}$ + $t_{WCK2CKPIN}$ + $t_{WCK2CK}$ + $t_{WCK2DQO}$, where CRCWL is the CRC Write latency programmed in MR4 and $t_{WCK2DQO}$ is the WCK to DQ/DBI_n/EDC phase offset at the DRAM balls. GDDR6 supports both a Full data rate and Half data rate EDC as described in the EDC section.

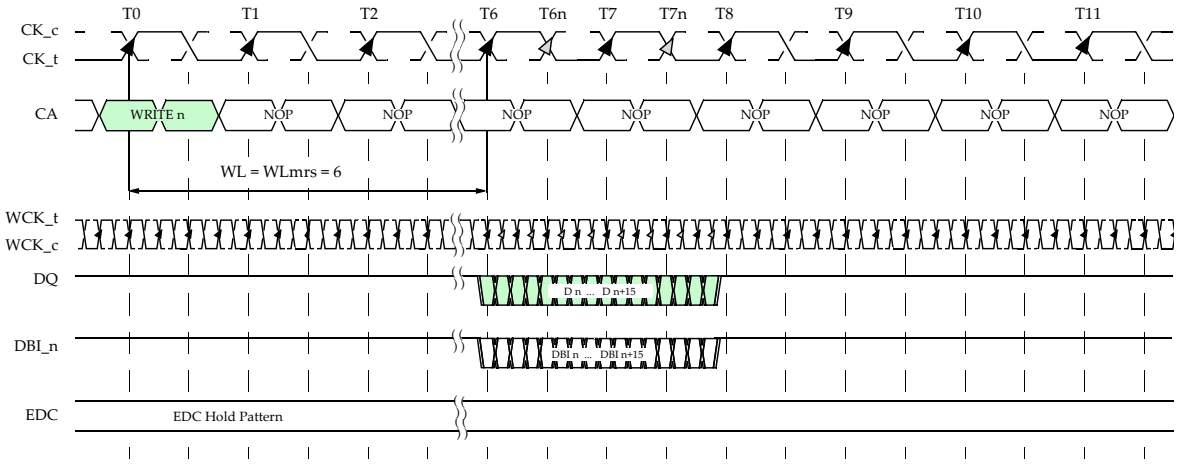
## 7.7 WRITE (WOM) (cont'd)




1) WLmrs is the WRITE latency programmed in Mode Register MR0.
2) Timings are shown with positive $t_{WCK2CKPIN}$ and $t_{WCK2CK}$ values. See WCK2CK timings for $t_{WCK2CKPIN}$ and $t_{WCK2CK}$ ranges.
3) $t_{WCK2DQI}$ parameter values could be negative or positive numbers, depending on PLL-on or PLL-off mode operation and design implementation. They also vary across PVT. Data training is required to determine the actual $t_{WCK2DQI}$ value for stable WRITE operation.
4) $t_{DQDQI}$ defines the minimum to maximum variation of $t_{WCK2DQI}$ within a single byte or double byte.
5) Data Read timings are used for CRC return timing from WRITE commands with CRC enabled.

**Figure 59 — WRITE Lane Timings**

## 7.7 WRITE (WOM) (cont'd)




Notes: 1. WLmrs = 6 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

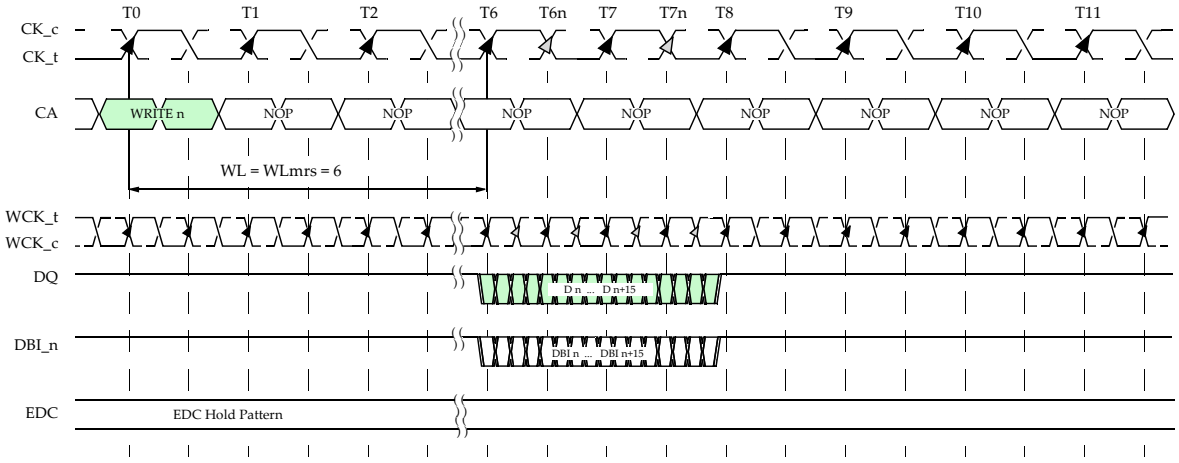
2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

3. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.

4. An ACTIVATE (ACT) command is required to be issued before the WRITE command, and $t_{RCDWR}$ must be met.

5. $t_{WCK2DQI}$, $t_{WCKDQO}$ = 0 is shown for illustration purposes.

**Figure 60 — Single WRITE without EDC (WCK DDR mode)**




Notes: 1. WLmrs = 6 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

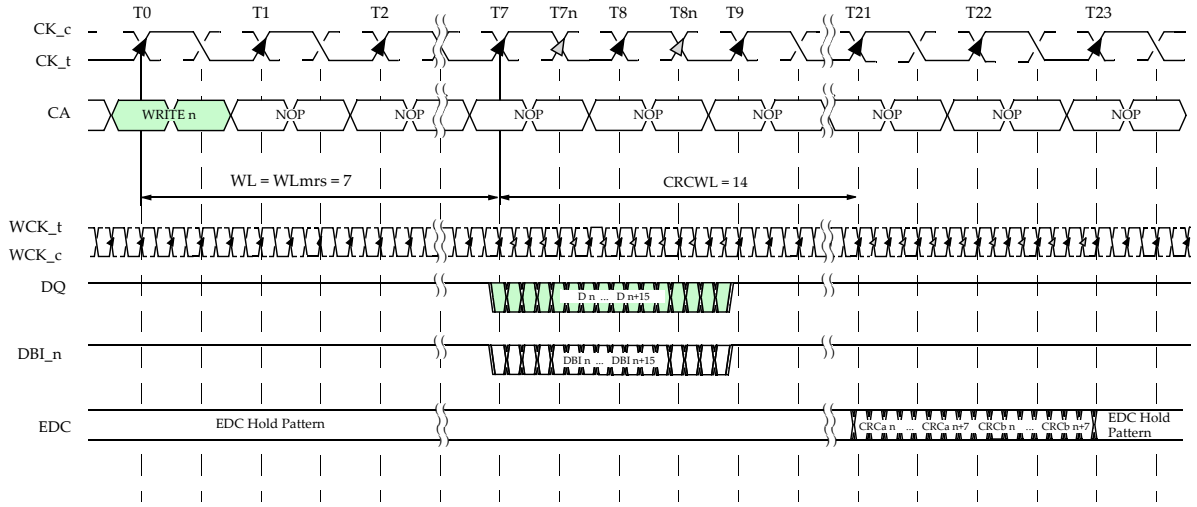
3. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.

4. An ACTIVATE (ACT) command is required to be issued before the WRITE command, and $t_{RCDWR}$ must be met.

5. $t_{WCK2DQI}$, $t_{WCKDQO}$ = 0 is shown for illustration purposes.

**Figure 61 —**

## 7.7 WRITE (WOM) (cont'd)




Notes: 1. WLmrs = 7 and CRCWL = 14 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

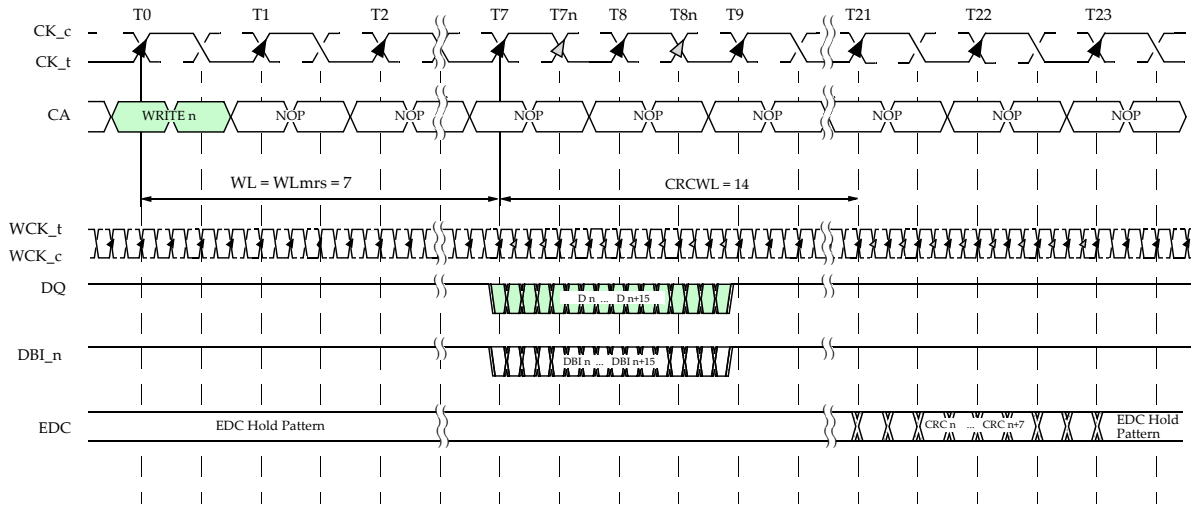
2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

3. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.

4. An ACTIVATE (ACT) command is required to be issued before the WRITE command, and $t_{RCDWR}$ must be met.

5. $t_{WCK2DQI}$, $t_{WCKDQO}$ = 0 is shown for illustration purposes.

**Figure 62 — Single WRITE with EDC (Full data rate)**




Notes: 1. WLmrs = 7 and CRCWL = 14 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

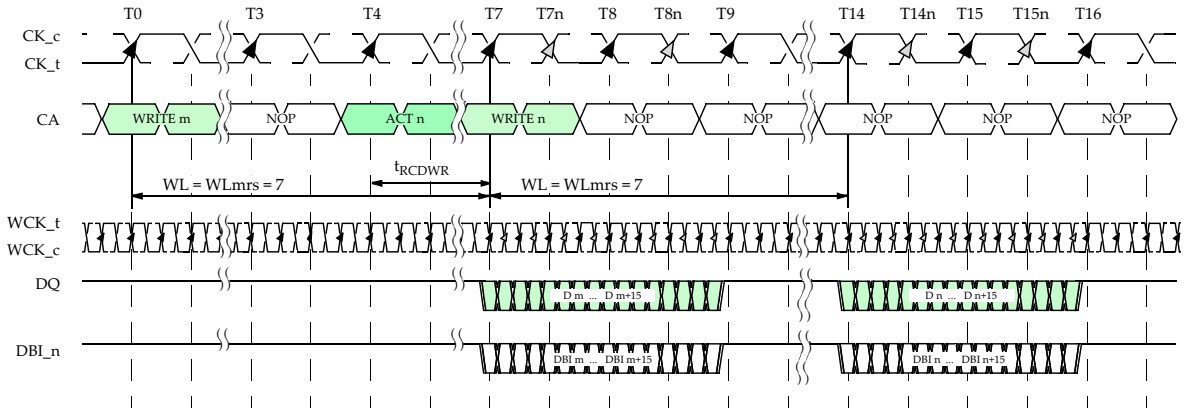
3. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.

4. An ACTIVATE (ACT) command is required to be issued before the WRITE command, and $t_{RCDWR}$ must be met.

5. $t_{WCK2DQI}$, $t_{WCKDQO}$ = 0 is shown for illustration purposes.

**Figure 63 — Single WRITE with EDC (Half data rate)**

## 7.7 WRITE (WOM) (cont'd)




Notes: 1. WLmrs = 7 and $t_{RCDWR}$ = 3 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

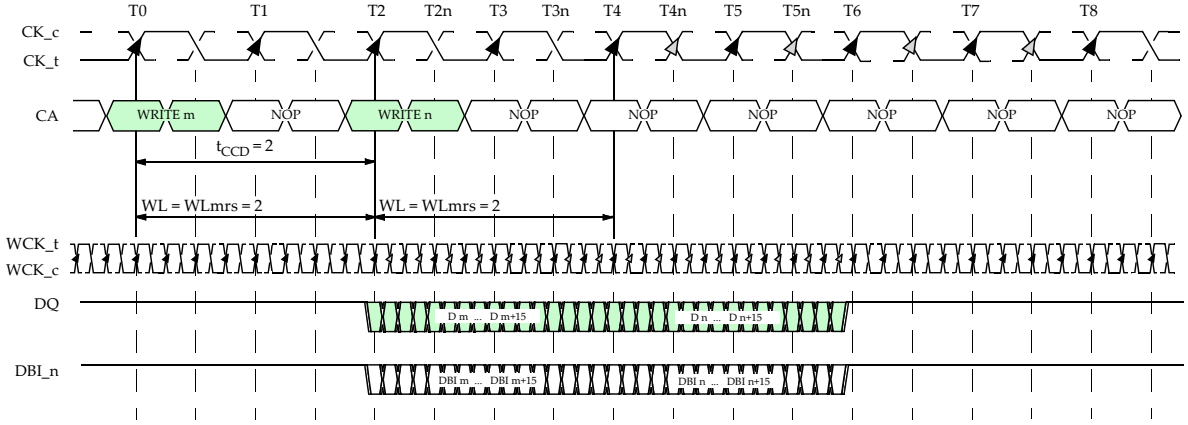
3. EDC may be on or off. See figure entitled "Single WRITE with EDC" for EDC Timing.

4. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.

5. An ACTIVATE (ACT) command is required to be issued before the WRITE commands, and $t_{RCDWR}$ must be met.

6. $t_{WCK2DQI}$ = 0 is shown for illustration purposes.

**Figure 64 — Non-Gapless WRITEs**




Notes: 1. WLmrs = 2 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

3. EDC may be on or off. See figure entitled "Single WRITE with EDC" for EDC Timing.

4. $t_{CCD}$ = $t_{CCDS}$ when bank groups is disabled or the second WRITE is to a different bank group, otherwise $t_{CCD}$=$t_{CCDL}$.

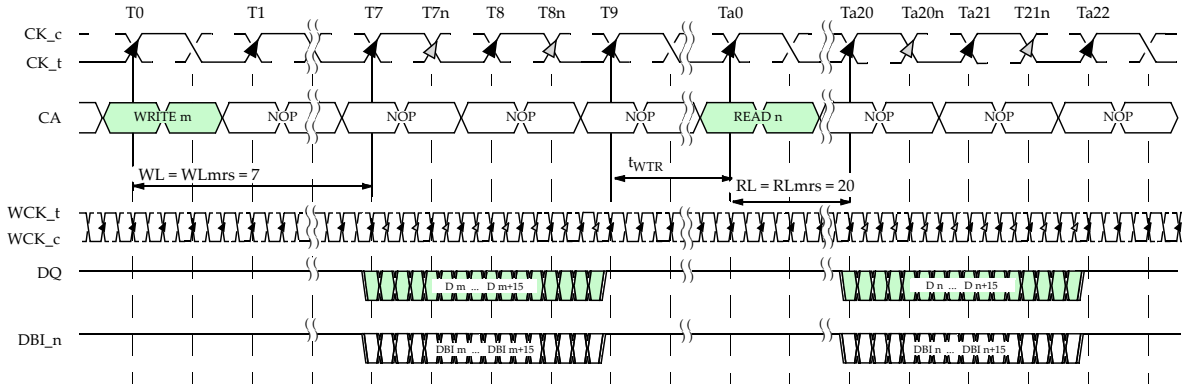
5. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.

6. An ACTIVATE (ACT) command is required to be issued before the WRITE commands, and $t_{RCDWR}$ must be met.

7. $t_{WCK2DQI}$ = 0 is shown for illustration purposes.

**Figure 65 — Gapless WRITEs**

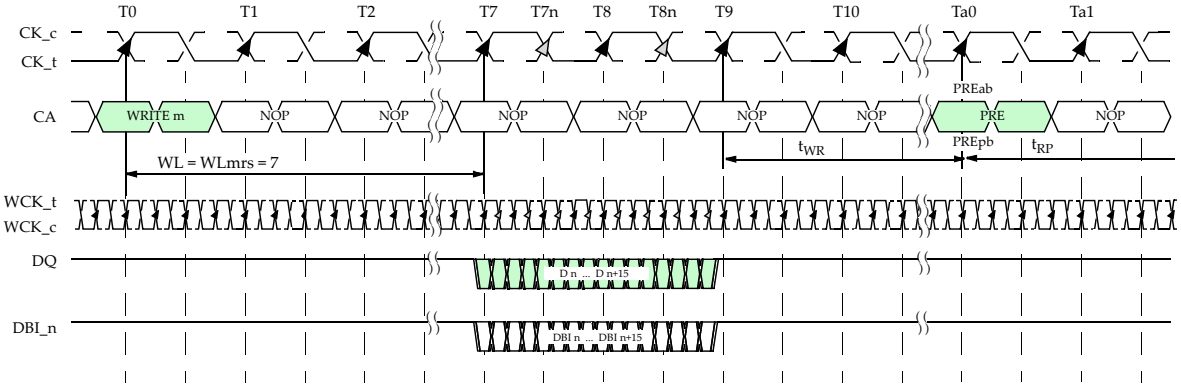
## 7.7 WRITE (WOM) (cont'd)




Notes: 1. WLmrs = 7 and RLmrs = 20 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.
3. EDC may be on or off. See figure entitled "Single WRITE with EDC" for EDC Timing.
4. $t_{WTR}$ = $t_{WTRL}$ when bank groups is enabled and both WRITE and READ access banks in the same bank group, otherwise $t_{WTR}$=$t_{WTRS}$.
5. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.
6. An ACTIVATE (ACT) command is required to be issued before the READ and WRITE commands, and $t_{RCDRD}$ or $t_{RCDWR}$ , respectively, must be met.
7. $t_{WCK2DQI}$, $t_{WCKDQO}$ = 0 is shown for illustration purposes.

**Figure 66 — WRITE to READ**




Notes: 1. WLmrs = 7 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.
3. EDC may be on or off. See figure entitled "Single WRITE with EDC" for EDC Timing.
4. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.
5. An ACTIVATE (ACT) command is required to be issued before the WRITE command, and $t_{RCDWR}$ must be met.
6. $t_{WCK2DQI}$ = 0 is shown for illustration purposes.
7. If auto precharge is enabled with the WRITE command, the precharge is initiated at the earliest possible time an explicit PRECHARGE would be allowed.

**Figure 67 — WRITE to PRECHARGE**

## 7.8 WRITE DATA MASK (DM)

The traditional method of using a DM input for WRITE data mask must be abandoned for a new method. Due to the high data rate, bit errors are expected on the interface and are not recoverable when they occur on the traditional DM input.

In GDDR6 the DM is sent to the SGRAM over the CA bus following the bank/column address cycle associated with the command, during the NOP commands between the WRITE command and the next command. The DM is used to mask the corresponding data according to the following table.

**Table 26 — DM State**

| FUNCTION | DM Value | DQ |
|---|---|---|
| Write Enable | 0 | Valid |
| Write Inhibit | 1 | X |

Two additional WRITE commands that augment the traditional WRITE Without Mask (WOM) are required for proper DM support. The WOM command should be consulted for general WRITE operation and Table 49 should be consulted for timing differences between the 2 additional WRITE commands. The two additional WRITE commands may need an internal Read-Modify-Write if they follow another WRITE command to the same bank. See the detailed description and timing differences in Section 0.30 and Table 49. The commands for proper DM support are:

- WDM: WRITE-With-Doublebyte-Mask:

2 cycle command where the 1st cycle carries command and address information and the 2nd cycle carries data mask information (2 byte granularity);

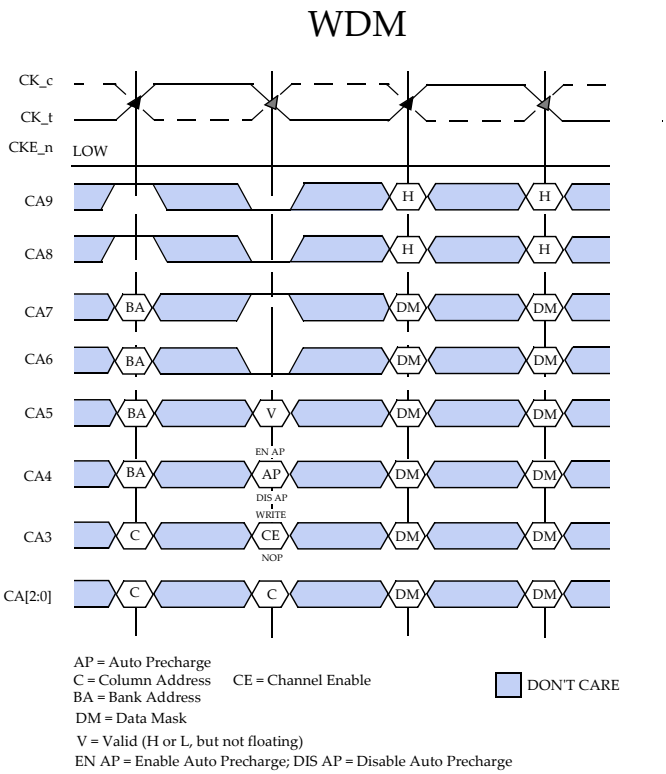



**Figure 68 — WRITE-With-Doublebyte-Mask Command**

## 7.8 WRITE DATA MASK (DM) (cont'd)

### Table 27 — WDM Mapping x16 mode

| | Byte and Burst Position Masked During WDM | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| DQ0 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 |
| DQ1 | | | | | | | | | | | | | | | | |
| DQ2 | | | | | | | | | | | | | | | | |
| DQ3 | | | | | | | | | | | | | | | | |
| DQ4 | | | | | | | | | | | | | | | | |
| DQ5 | | | | | | | | | | | | | | | | |
| DQ6 | | | | | | | | | | | | | | | | |
| DQ7 | | | | | | | | | | | | | | | | |
| DQ8 | | | | | | | | | | | | | | | | |
| DQ9 | | | | | | | | | | | | | | | | |
| DQ10 | | | | | | | | | | | | | | | | |
| DQ11 | | | | | | | | | | | | | | | | |
| DQ12 | | | | | | | | | | | | | | | | |
| DQ13 | | | | | | | | | | | | | | | | |
| DQ14 | | | | | | | | | | | | | | | | |
| DQ15 | | | | | | | | | | | | | | | | |

Device 0

### Table 28 — Example WDM x16 mode

| | Byte and Burst Position Masked During WDM | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| DQ0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| DQ1 | | | | | | | | | | | | | | | | |
| DQ2 | | | | | | | | | | | | | | | | |
| DQ3 | | | | | | | | | | | | | | | | |
| DQ4 | | | | | | | | | | | | | | | | |
| DQ5 | | | | | | | | | | | | | | | | |
| DQ6 | | | | | | | | | | | | | | | | |
| DQ7 | | | | | | | | | | | | | | | | |
| DQ8 | | | | | | | | | | | | | | | | |
| DQ9 | | | | | | | | | | | | | | | | |
| DQ10 | | | | | | | | | | | | | | | | |
| DQ11 | | | | | | | | | | | | | | | | |
| DQ12 | | | | | | | | | | | | | | | | |
| DQ13 | | | | | | | | | | | | | | | | |
| DQ14 | | | | | | | | | | | | | | | | |
| DQ15 | | | | | | | | | | | | | | | | |

Device 0

| | Data out from Example WDM | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1st Cycle Data | | | | | | | | | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| DQ0 | Data | Data | Data | Data | Data | Data | Data | Data | Masked | | | | | | | |
| DQ1 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ2 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ3 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ4 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ5 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ6 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ7 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ8 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ9 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ10 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ11 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ12 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ13 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ14 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ15 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |

Device 0

## 7.8 WRITE DATA MASK (DM) (cont'd)

### Table 29 — WDM Mapping x8 mode

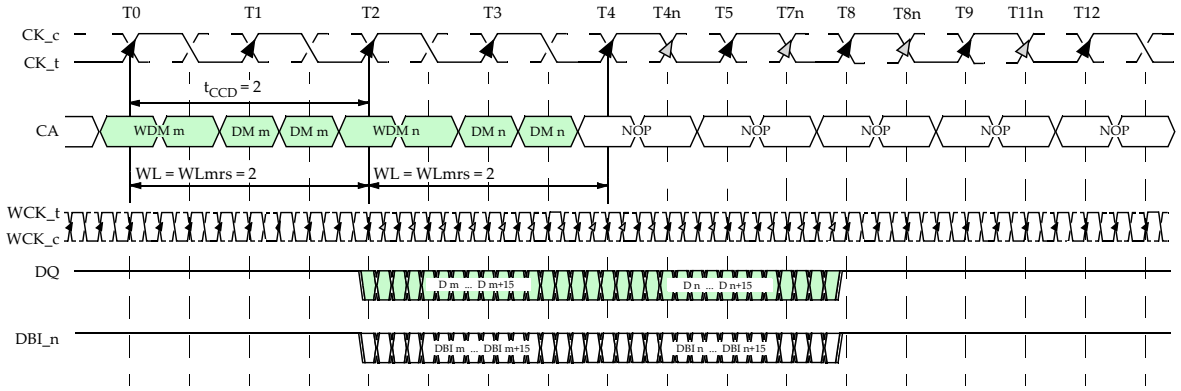
| | | Byte and Burst Position Masked During WDM | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | |
| | Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Device 0 | DQ0 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 |
| | DQ1 | | | | | | | | | | | | | | | | |
| | DQ2 | | | | | | | | | | | | | | | | |
| | DQ3 | | | | | | | | | | | | | | | | |
| | DQ4 | | | | | | | | | | | | | | | | |
| | DQ5 | | | | | | | | | | | | | | | | |
| | DQ6 | | | | | | | | | | | | | | | | |
| | DQ7 | | | | | | | | | | | | | | | | |
| Device 1 | DQ8 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 |
| | DQ9 | | | | | | | | | | | | | | | | |
| | DQ10 | | | | | | | | | | | | | | | | |
| | DQ11 | | | | | | | | | | | | | | | | |
| | DQ12 | | | | | | | | | | | | | | | | |
| | DQ13 | | | | | | | | | | | | | | | | |
| | DQ14 | | | | | | | | | | | | | | | | |
| | DQ15 | | | | | | | | | | | | | | | | |

### Table 30 — Example WDM x8 mode

| | | Byte and Burst Position Masked During WDM | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | |
| | Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Device 0 | DQ0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | DQ1 | | | | | | | | | | | | | | | | |
| | DQ2 | | | | | | | | | | | | | | | | |
| | DQ3 | | | | | | | | | | | | | | | | |
| | DQ4 | | | | | | | | | | | | | | | | |
| | DQ5 | | | | | | | | | | | | | | | | |
| | DQ6 | | | | | | | | | | | | | | | | |
| | DQ7 | | | | | | | | | | | | | | | | |
| Device 1 | DQ8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | DQ9 | | | | | | | | | | | | | | | | |
| | DQ10 | | | | | | | | | | | | | | | | |
| | DQ11 | | | | | | | | | | | | | | | | |
| | DQ12 | | | | | | | | | | | | | | | | |
| | DQ13 | | | | | | | | | | | | | | | | |
| | DQ14 | | | | | | | | | | | | | | | | |
| | DQ15 | | | | | | | | | | | | | | | | |

| | | Data out from Example WDM | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1st Cycle Data | | | | | | | | | | | | | | | |
| | Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Device 0 | DQ0 | Data | Data | Data | Data | Data | Data | Data | Data | Masked | | | | | | | |
| | DQ1 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ2 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ3 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ4 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ5 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ6 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ7 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| Device 1 | DQ8 | Data | Data | Data | Data | Data | Data | Data | Data | Masked | | | | | | | |
| | DQ9 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ10 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ11 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ12 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ13 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ14 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| | DQ15 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |

## 7.8 WRITE DATA MASK (DM) (cont'd)




Notes: 1. WLmrs = 2 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

3. EDC may be on or off. See figure entitled "Single WRITE with EDC" for EDC Timing.

4. $t_{CCD}$ = $t_{CCDS}$ when bank groups is disabled or the second WRITE is to a different bank group, otherwise $t_{CCD}$=$t_{CCDL}$.

5. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.

6. An ACTIVATE (ACT) command is required to be issued before the WRITE command, and $t_{RCDWR}$ must be met.

7. $t_{WCK2DQI}$ = 0 is shown for illustration purposes.

**Figure 69 — WDM Timing**

## 7.8 WRITE DATA MASK (DM) (cont'd)

- WSM: WRITE-With-Singlebyte-Mask:

3 cycle command where the 1st cycle carries command and address information and the 2nd and 3rd cycle carry data mask information (byte granularity);

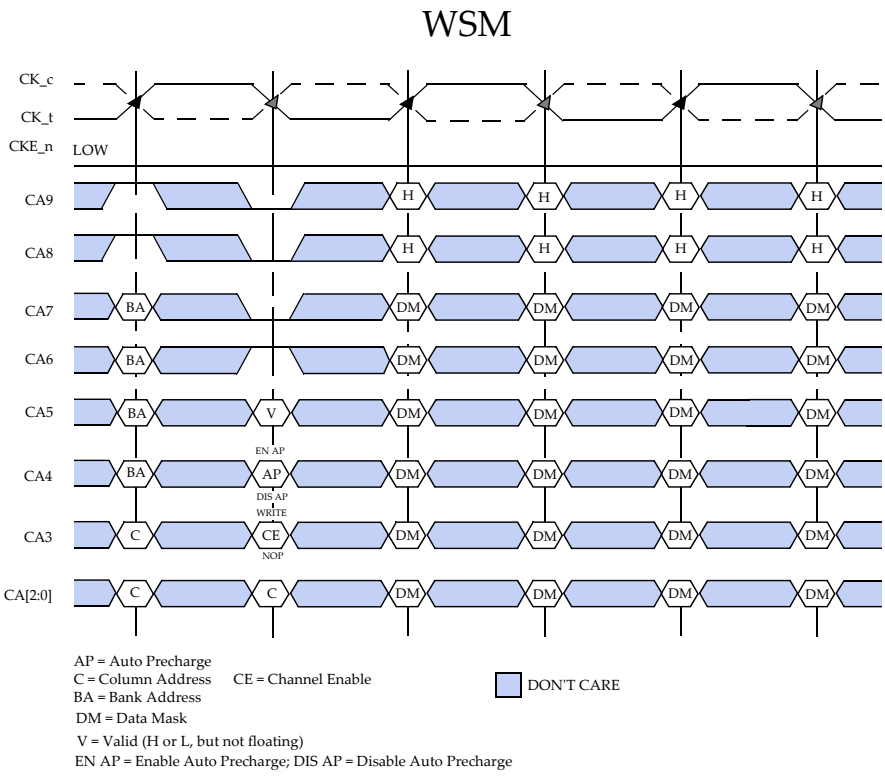



**Figure 70 — WRITE-With-Singlebyte-Mask Command**

## 7.8 WRITE DATA MASK (DM) (cont'd)

### Table 31 — WSM Mapping x16 mode

| Byte and Burst Position Masked During WSM | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | | 3rd Cycle Rising Edge of CK | | | | | | | | 3rd Cycle Falling Edge of CK | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| DQ0 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 |
| DQ1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | DQ8 | DQ9 | DQ10 | DQ11 | DQ12 | DQ13 | DQ14 | DQ15 | | | | | | | | | | | | | | | | | | | | | | | | |

### Table 32 — Example WSM x16 mode

| Byte and Burst Position Masked During WSM | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | | 3rd Cycle Rising Edge of CK | | | | | | | | 3rd Cycle Falling Edge of CK | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| DQ0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| DQ4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DQ7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | DQ8 | DQ9 | DQ10 | DQ11 | DQ12 | DQ13 | DQ14 | DQ15 | | | | | | | | | | | | | | | | | | | | | | | | |

| Data output from Example WSM | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1st Cycle Data | | | | | | | | | | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| DQ0 | Masked | | | | | | | | Data | Data | Data | Data | Data | Data | Data | Data |
| DQ1 | | | | | | | | | Data | Data | Data | Data | Data | Data | Data | Data |
| DQ2 | | | | | | | | | Data | Data | Data | Data | Data | Data | Data | Data |
| DQ3 | | | | | | | | | Data | Data | Data | Data | Data | Data | Data | Data |
| DQ4 | | | | | | | | | Data | Data | Data | Data | Data | Data | Data | Data |
| DQ5 | | | | | | | | | Data | Data | Data | Data | Data | Data | Data | Data |
| DQ6 | | | | | | | | | Data | Data | Data | Data | Data | Data | Data | Data |
| DQ7 | | | | | | | | | Data | Data | Data | Data | Data | Data | Data | Data |
| DQ8 | Data | Data | Data | Data | Data | Data | Data | Data | Masked | | | | | | | |
| DQ9 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ10 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ11 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ12 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ13 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ14 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |
| DQ15 | Data | Data | Data | Data | Data | Data | Data | Data | | | | | | | | |

## 7.8 WRITE DATA MASK (DM) (cont'd)

### Table 33 — WSM Mapping for x8 mode

| | Byte and Burst Position Masked During WSM for x8 mode device | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | | 3rd Cycle Rising Edge of CK | | | | | | | | 3rd Cycle Falling Edge of CK | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Device 0 DQ0 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 | | | | | | | | | | | | | | | | |
| Device 0 DQ1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

### Table 34 — WSM Mapping for x8 mode

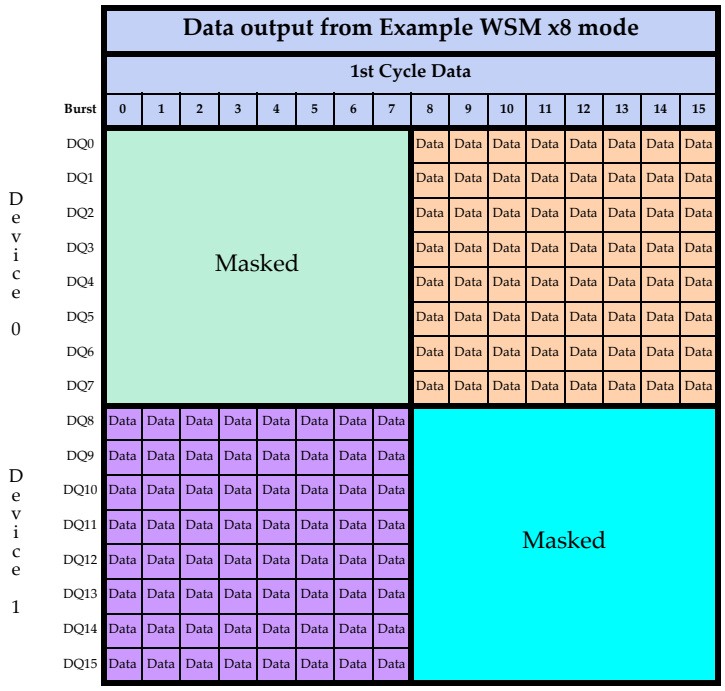
| | Byte and Burst Position Masked During WSM for x8 mode device | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | | 3rd Cycle Rising Edge of CK | | | | | | | | 3rd Cycle Falling Edge of CK | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Device 1 DQ8 | | | | | | | | | | | | | | | | | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 | CA0 | CA1 | CA2 | CA3 | CA4 | CA5 | CA6 | CA7 |
| Device 1 DQ9 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ11 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ13 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ14 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ15 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

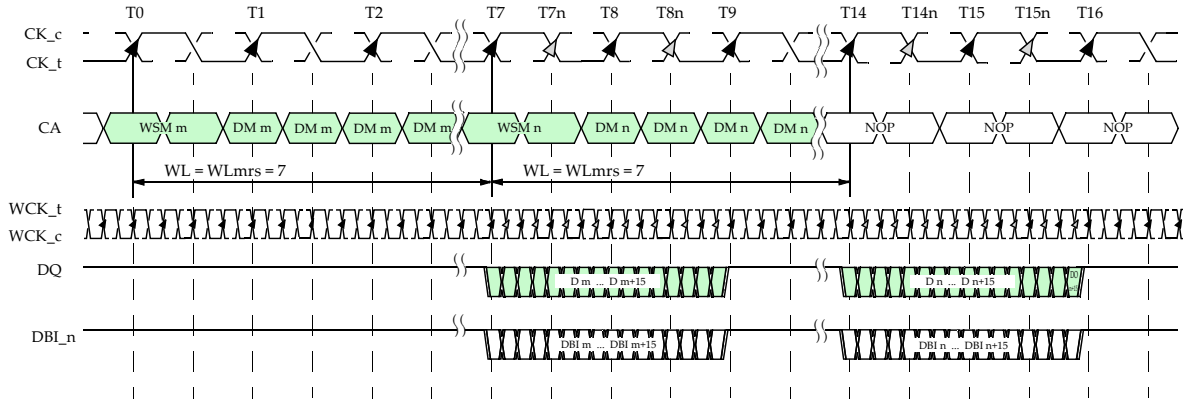
### Table 35 — Example WSM

| | Byte and Burst Position Masked During WSM for x8 mode device | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | | 3rd Cycle Rising Edge of CK | | | | | | | | 3rd Cycle Falling Edge of CK | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Device 0 DQ0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | | | | | | |
| Device 0 DQ1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 0 DQ7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | Byte and Burst Position Masked During WSM for x8 mode device | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2nd Cycle Rising Edge of CK | | | | | | | | 2nd Cycle Falling Edge of CK | | | | | | | | 3rd Cycle Rising Edge of CK | | | | | | | | 3rd Cycle Falling Edge of CK | | | | | | | |
| Burst | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Device 1 DQ8 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Device 1 DQ9 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ11 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ13 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ14 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device 1 DQ15 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

## 7.8 WRITE DATA MASK (DM) (cont'd)




Notes: 1. WLmrs = 7 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

3. EDC may be on or off. See figure entitled "Single WRITE with EDC" for EDC Timing.

4. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.

5. An ACTIVATE (ACT) command is required to be issued before the WRITE command, and $t_{RCDWR}$ must be met.

6. $t_{WCK2DQI}$ = 0 is shown for illustration purposes.

**Figure 71 — WSM Timing**

## 7.10 READ

A READ burst is initiated with a READ command as shown in Figure 73. The bank and column addresses are provided with the READ command and auto precharge is either enabled or disabled for that access with the AP bit (CA4 input falling edge of CK). If auto precharge is enabled, the row being accessed is precharged at a time $t_{RTP}$ after the READ command and after $t_{RAS}$(min) has been met or after the number of clock cycles programmed in the RAS field of MR5, depending on the implementation choice per DRAM vendor. The length of the burst initiated with a READ command is sixteen and the column address is unique for this burst of sixteen. There is no interruption nor truncation of READ bursts. The array read access is suppressed, no read data and the EDC hold pattern instead of a CRC burst are transmitted when the CE bit is LOW. In two channel mode CE shall be driven HIGH.

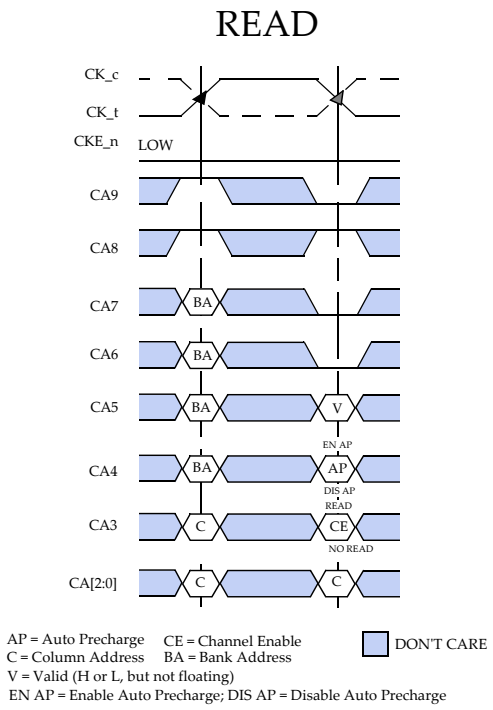



**Figure 73 — READ Command**

READ timings are shown with DDR WCK ratio in Figure 74. During READ bursts, the first valid data-out element will be available after the Read latency (RL). The Read Latency is defined as RLmrs * $t_{CK}$ + $t_{WCK2CKPIN}$ + $t_{WCK2CK}$ + $t_{WCK2DQO}$, where RLmrs is the number of clock cycles programed in MR0, $t_{WCK2CKPIN}$ is the phase offset between WCK and CK at the balls when phase aligned at the phase detector, $t_{WCK2CK}$ is the alignment error between WCK and CK at the phase detector, and $t_{WCK2DQO}$ is the WCK to DQ/DBI_n/EDC offset as measured at the DRAM balls. The total delay is relative to the data eye initial edge averaged over one byte if WCK/byte or double byte if WCK/word. The maximum skew within a byte is defined by $t_{DQDQO}$.

Upon completion of a burst, assuming no other READ command has been initiated, all DQ and DBI_n signals will drive a value of '1' and the ODT will be enabled at a maximum of 1 $t_{CK}$ later. The drive value and termination value may be different due to separately defined calibration offsets. If the ODT is disabled, the signals will drive Hi-Z.

Data from any READ burst may be concatenated with data from a subsequent READ command. A continuous flow of data can be maintained. The first data element from the new burst follows the last element of a completed burst. The new READ command should be issued after the previous READ command according to the $t_{CCD}$ timing. If that READ command is to another bank then an ACTIVATE command must precede the READ command and $t_{RCDRD}$ also must be met.

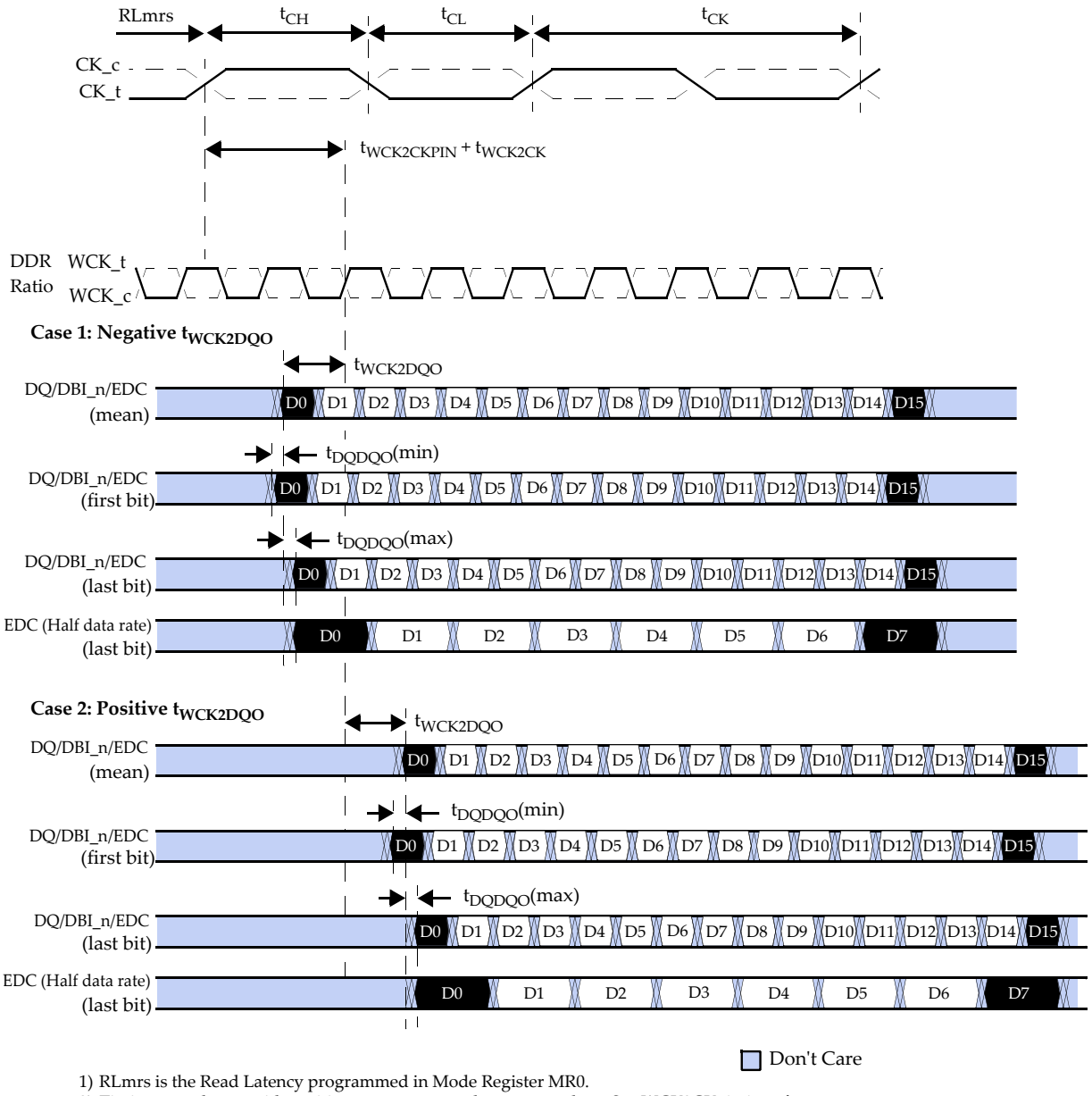
## 7.10 READ (cont'd)

A WRITE can be issued any time after a READ command as long as the bus turn around time $t_{RTW}$ is met. If that WRITE command is to another bank, then an ACTIVATE command must precede the second WRITE command and $t_{RCDWR}$ also must be met. A PRECHARGE can also be issued with the same timing restriction as the new READ command if $t_{RAS}$ is met. After the PRECHARGE command, a subsequent command to the same bank cannot be issued until $t_{RP}$ is met.

The data inversion flag is driven on the DBI_n pin to identify whether the data is true or inverted data. If DBI_n is HIGH, the data is not inverted, and if LOW it is inverted. Read Data Inversion can be enabled (OP8 = 0) or disabled (OP8 = 1) using RDBI in MR1.

When enabled by the RDCRC flag in MR4 EDC data is returned to the controller with a latency of (RLmrs + CRCRL) * $t_{CK}$ + $t_{WCK2CKPIN}$ + $t_{WCK2CK}$ + $t_{WCK2DQO}$, where CRCRL is the CRC Read latency programmed in MR4. GDDR6 supports both a Full data rate and Half data rate EDC as described in the EDC section.

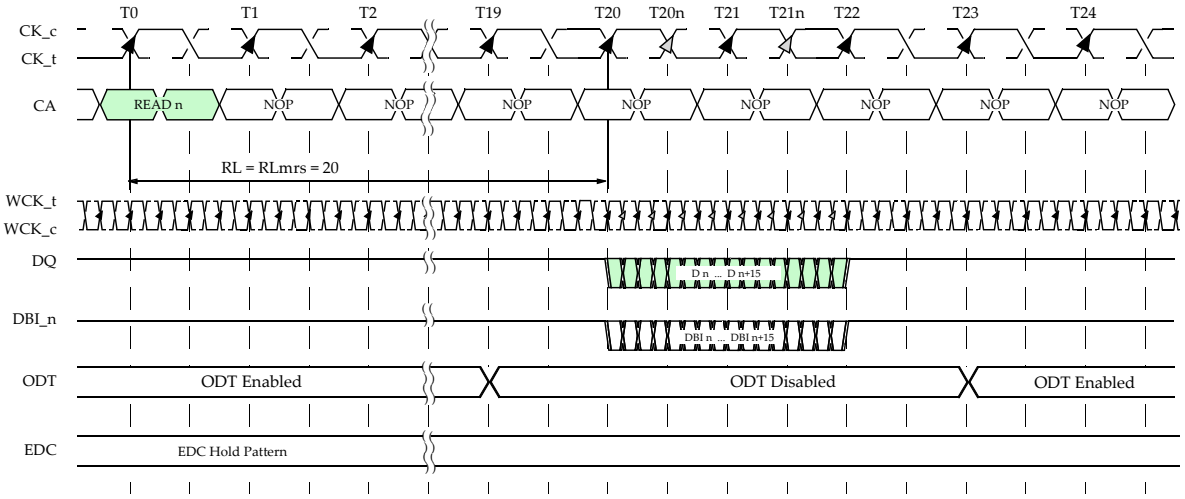
## 7.10 READ (cont'd)




1) RLmrs is the Read Latency programmed in Mode Register MR0.
2) Timings are shown with positive $t_{WCK2CKPIN}$ and $t_{WCK2CK}$ values. See WCK2CK timings for $t_{WCK2CKPIN}$ and $t_{WCK2CK}$ ranges.
3) $t_{WCK2DQO}$ parameter values could be negative or positive numbers, depending on PLL-on or PLL-off mode operation and design implementation. They also vary across PVT. Data training is required to determine the actual $t_{WCK2DQO}$ value for stable READ operation.
4) $t_{DQDQO}$ defines the minimum to maximum variation of $t_{WCK2DQO}$ within a byte or double byte.
5) $t_{DQDQO}$ also applies for CRC data from WRITE and READ commands with CRC enabled and the EDC hold pattern.
6) EDC is shown with Full data rate EDC. Half data rate EDC (last bit) is shown separately for comparison.

**Figure 74 — READ Lane Timing**

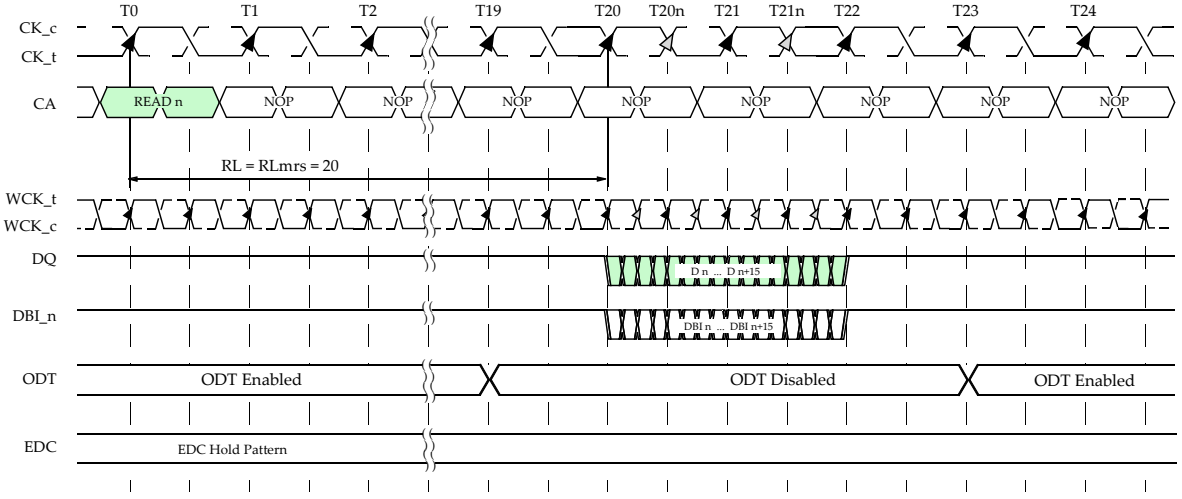
## 7.10 READ (cont'd)




Notes: 1. RLmrs = 20 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.
3. An ACTIVATE (ACT) command is required to be issued before the READ command, and $t_{RCDRD}$ must be met.
4. $t_{WCK2DQO}$ = 0 is shown for illustration purposes.

**Figure 75 — Single READ without EDC (WCK DDR mode)**

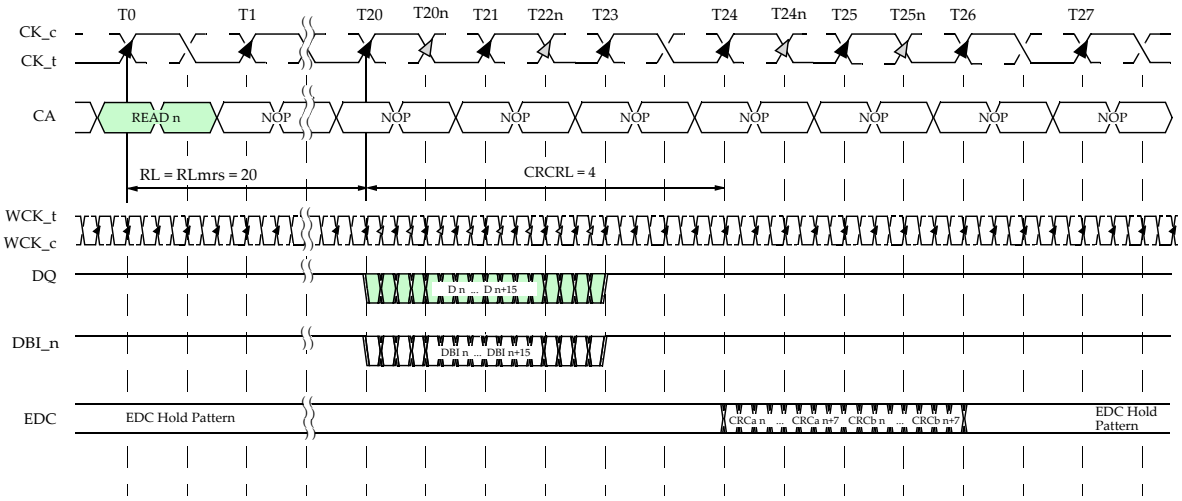



Notes: 1. RLmrs = 20 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.
3. An ACTIVATE (ACT) command is required to be issued before the READ command, and $t_{RCDRD}$ must be met.
4. $t_{WCK2DQO}$ = 0 is shown for illustration purposes.

**Figure 76 —**

## 7.10 READ (cont'd)



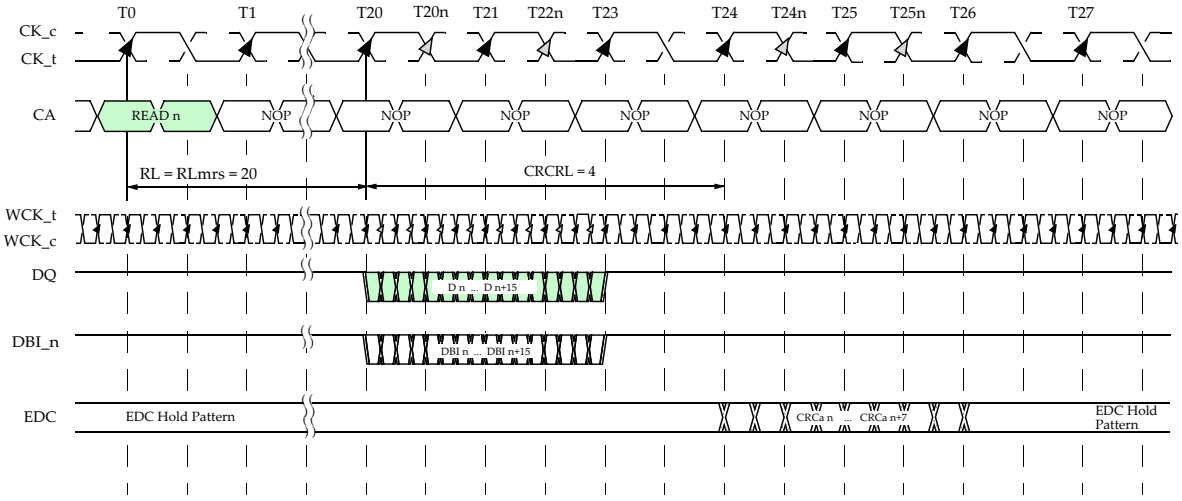

Notes: 1. RLmrs = 20 and CRCRL = 4 are shown as examples. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

3. An ACTIVATE (ACT) command is required to be issued before the READ command, and $t_{RCDRD}$ must be met.

4. $t_{WCK2DQO}$ = 0 is shown for illustration purposes.

**Figure 77 — Single READ with EDC (Full data rate)**




Notes: 1. RLmrs = 20 and CRCRL = 4 are shown as examples. Actual supported values will be found in the MR and AC timings sections.

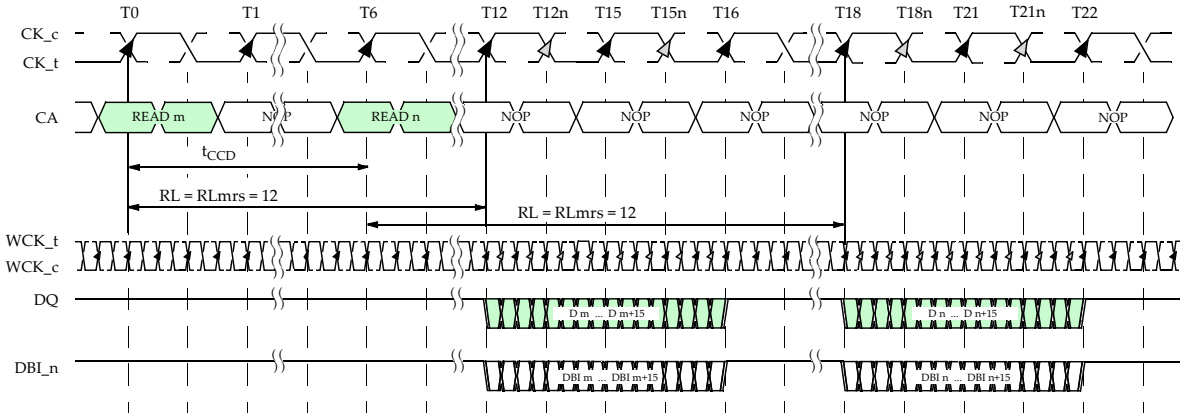
2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

3. An ACTIVATE (ACT) command is required to be issued before the READ command, and $t_{RCDRD}$ must be met.

4. $t_{WCK2DQO}$ = 0 is shown for illustration purposes.

**Figure 78 — Single READ with EDC (Half data rate)**

## 7.10 READ (cont'd)




Notes: 1. RLmrs = 12 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

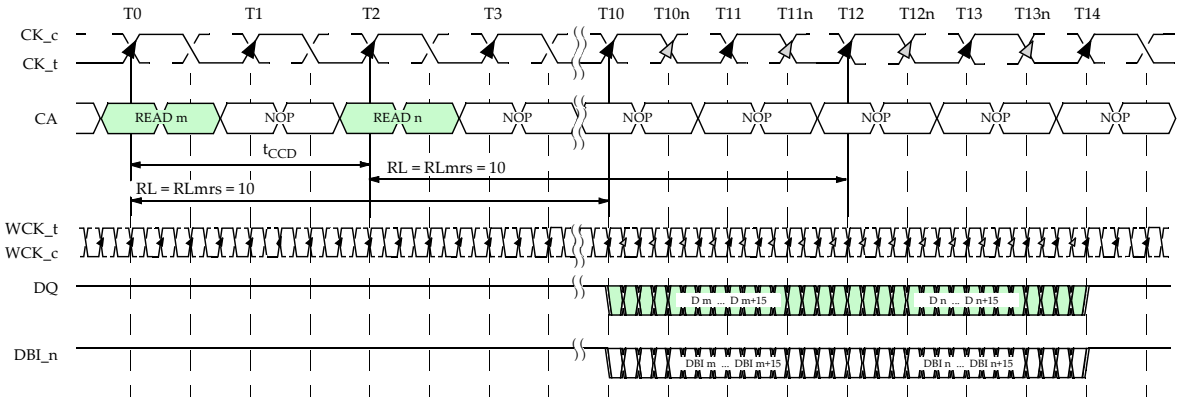
3. EDC may be on or off. See figure entitled "Single READ with EDC" for EDC Timing.

4. $t_{CCD}$ = $t_{CCDL}$ when bank groups are enabled and both READs access banks in the same bank group; otherwise $t_{CCD}$=$t_{CCDS}$.

5. An ACTIVATE (ACT) command is required to be issued before the READ commands, and $t_{RCDRD}$ must be met.

6. $t_{WCK2DQI}$ = 0 is shown for illustration purposes.

**Figure 79 — Non-Gapless READs**




Notes: 1. RLmrs = 10 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

3. EDC may be on or off. See figure entitled "Single READ with EDC" for EDC Timing.

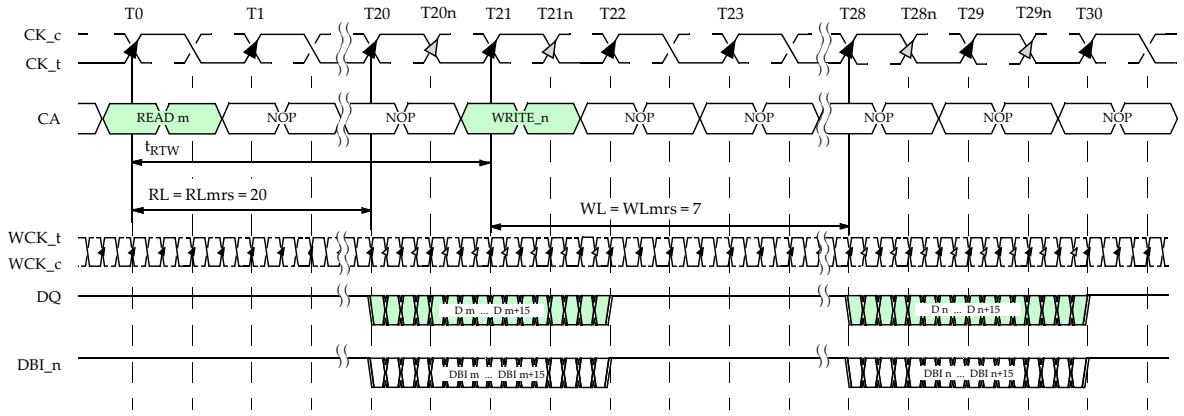
4. $t_{CCD}$ = $t_{CCDS}$ when bank groups are disabled or the second READ is to a different bank group; otherwise $t_{CCD}$=$t_{CCDL}$.

5. An ACTIVATE (ACT) command is required to be issued before the READ commands, and $t_{RCDRD}$ must be met.

6. $t_{WCK2DQI}$ = 0 is shown for illustration purposes.

**Figure 80 — Gapless READs**

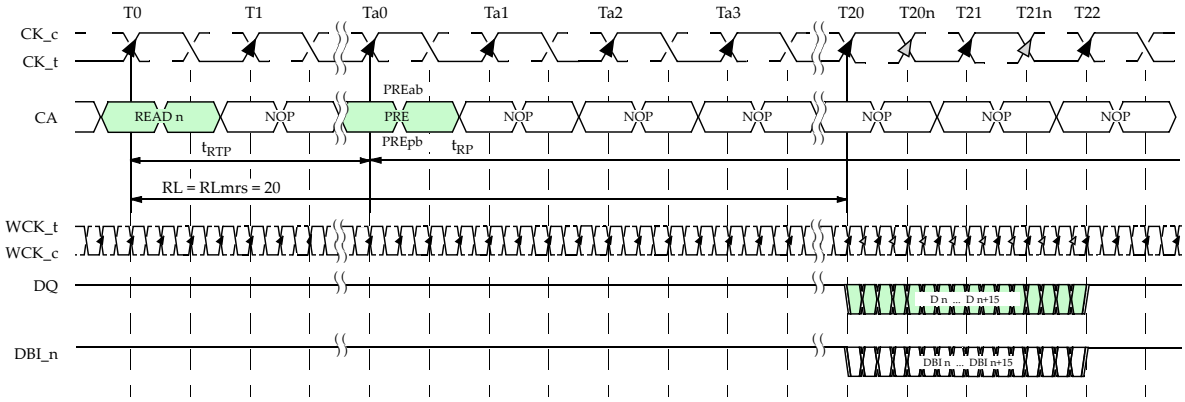
## 7.10 READ (cont'd)




Notes: 1. WLmrs = 7 and RLmrs = 20 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.
3. EDC may be on or off. See figure entitled "Single READ with EDC" for EDC Timing.
4. $t_{RTW}$ is not a device limit but determined by the system bus turnaround time. The difference between $t_{WCK2DQI}$ ,$t_{WCK2DQO}$.shall be considered in the calculation of the bus turnaround time.
5. For WRITE operations it is important that the latching point meet the data valid window requirements, which may or may not be center aligned at the balls.
6. An ACTIVATE (ACT) command is required to be issued before the READ and WRITE commands, and $t_{RCDRD}$ or $t_{RCDWR}$ , respectively, must be met.
7. $t_{WCK2DQI}$, $t_{WCKDQO}$ = 0 is shown for illustration purposes.

**Figure 81 — READ to WRITE**




Notes: 1. RLmrs = 20 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.
3. EDC may be on or off. See figure entitled "Single READ with EDC" for EDC Timing.
4. $t_{RTP}$ = $t_{RTPL}$ when bank groups are enabled and the PRECHARGE command accesses the same bank; otherwise $t_{RTP}$ = $t_{RTPS}$.
5. An ACTIVATE (ACT) command is required to be issued before the READ command, and $t_{RCDRD}$ must be met.
6. $t_{WCK2DQO}$ = 0 is shown for illustration purposes.
7. If auto precharge is enabled with the READ command, the precharge is initiated at the earliest possible time an explicit PRECHARGE would be allowed.

**Figure 82 — READ to PRECHARGE**

## 7.11 DQ PREAMBLE

DQ preamble is an optional feature for GDDR6 SGRAMs that is used for READ data. DQ preamble conditions the DQs for better signal integrity on the initial data of a burst.

Once enabled by OP5 in MR7, the DQ preamble will precede all READ bursts, including non-consecutive READ bursts with a minimum gap of 1 $t_{CK}$, as shown in Figure 79. When enabled, the DQ preamble pattern applies to all DQs and DBI_n in a byte, and the same pattern is used for all bytes as shown in Figure 83. DQ preamble is enabled or disabled for all bytes. The EDC in each byte is not included in the DQ preamble. If ODT is enabled, the ODT is disabled 1 $t_{CK}$ before the start of the preamble pattern as shown in Figure 84.

The preamble pattern on DBI_n is only enabled if the MR for RDBI is enabled (MR1 OP8). During the preamble DBI_n is treated as another DQ and the preamble pattern on the DQs is not encoded with RDBI. If RDBI is disabled, then DBI_n drives ODT.

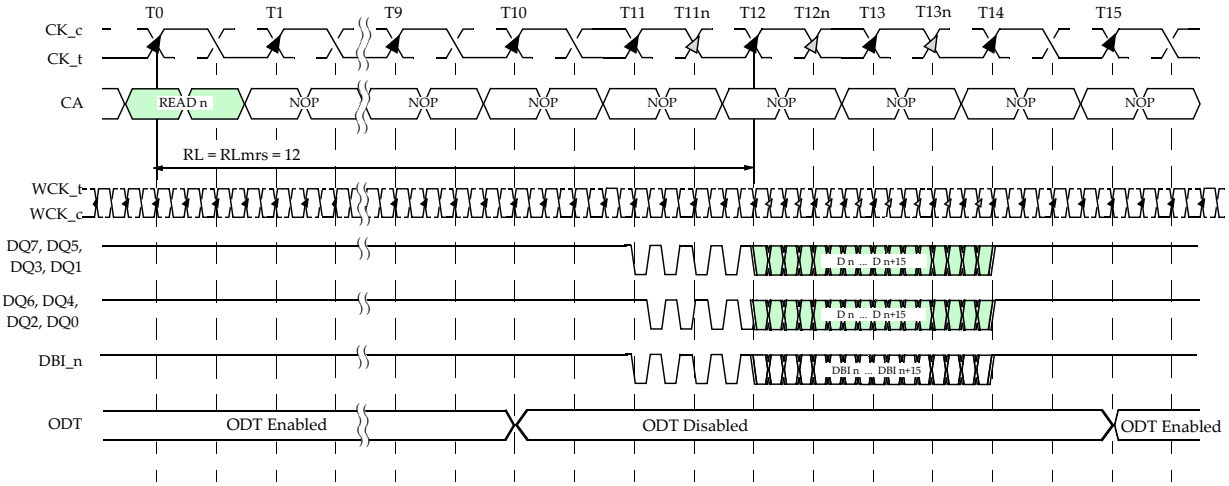
| Byte 0 | Byte 1 | Idle | | | | | | | | Preamble | | | | | | | | Burst | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| DQ7 | DQ15 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | V | V | V | V | V | V | V | V | V | V | V | V | V | V | V |
| DQ6 | DQ14 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | V | V | V | V | V | V | V | V | V | V | V | V | V | V | V |
| DQ5 | DQ13 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | V | V | V | V | V | V | V | V | V | V | V | V | V | V | V |
| DQ4 | DQ12 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | V | V | V | V | V | V | V | V | V | V | V | V | V | V | V |
| DQ3 | DQ11 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | V | V | V | V | V | V | V | V | V | V | V | V | V | V | V |
| DQ2 | DQ10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | V | V | V | V | V | V | V | V | V | V | V | V | V | V | V |
| DQ1 | DQ9 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | V | V | V | V | V | V | V | V | V | V | V | V | V | V | V |
| DQ0 | DQ8 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | V | V | V | V | V | V | V | V | V | V | V | V | V | V | V |
| DBI0_n | DBI1_n | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | V | V | V | V | V | V | V | V | V | V | V | V | V | V | V |
| | Max 0's | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 5 | 4 | 5 | 4 | 5 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |

Time

NOTES
1. The number of Max 0's in the burst is 4 only if RDBI is enabled. Max 0's is on a per byte basis and does not include the EDC.
2. V = Valid signal (H or L, but not floating)

**Figure 83 — DQ Preamble Pattern**

## 7.11 DQ PREAMBLE (cont'd)




Notes: 1. RLmrs = 12 is shown as an example. Actual supported values will be found in the MR and AC timings sections.

2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.

3. EDC may be on or off. See figure entitled "Single READ with EDC" for EDC Timing.

4. DQ6, DQ7 and DBI are shown to illustrate the DQ preamble pattern. RDBI is Enabled (MR1 OP8=0).

5. An ACTIVATE (ACT) command is required to be issued before the READ command, and $t_{RCDRD}$ must be met.

6. $t_{WCK2DQO}$ = 0 is shown for illustration purposes.

**Figure 84 — Preamble Timing Diagram**

## 7.12 RDQS MODE

For device operation at lower clock frequencies the device may be set into RDQS mode in which a READ DATA STROBE (RDQS) will be sent on the EDC signals along with the READ data. The controller will use the RDQS to latch the READ data.

RDQS mode is entered by setting the RDQS Mode bit OP9 in Mode Register 2 (MR2). When the bit is set, the device will asynchronously terminate any EDC hold pattern and drive a logic HIGH after $t_{MOD}$ at the latest. All features controlled by MR4 and the PRBS controlled by MR12 OP2 are ignored by RDQS mode.

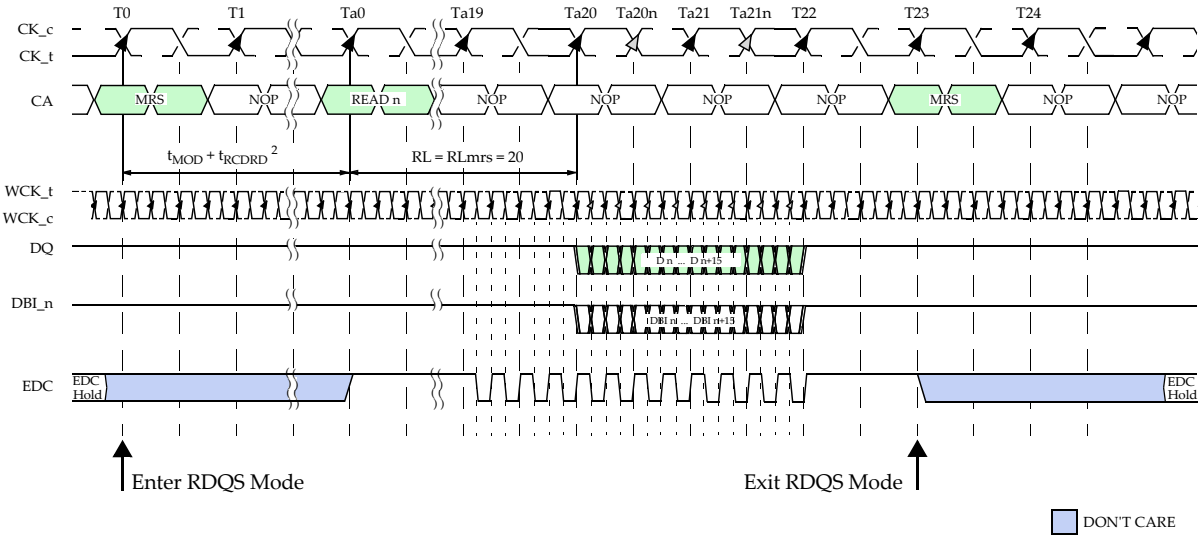
READ commands are executed as in normal mode regarding command to data out delay and programmed READ latencies. A fixed clock-like pattern as shown in Figure 85 is driven on the EDC signals in phase (edge aligned) with the DQ. Prior to the first valid data element, this fixed clock-like pattern or READ preamble is driven for 1 tCK.

No CRC is calculated in RDQS mode, neither for READs nor for WRITEs. The CRC engine is effectively disabled, and the corresponding WRCRC and RDCRC Mode Register bits are ignored. The PLL/DLL may be on or off with RDQS mode, depending on system considerations and the PLL/DLL's minimum clock frequency.

There is no equivalent WDQS mode; WRITE commands to the device are not affected by RDQS mode.

RDQS mode is exited by resetting the RDQS Mode bit. In this case the device will asynchronously start driving the EDC hold pattern after $t_{MOD}$.

The WCK2CK training should be performed prior to entering RDQS mode.




1. WCK_t and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.
2. An ACTIVATE (ACT) command is required to be issued before the READ command, and $t_{RCDRD}$ must be met.
3. $t_{WCK2DQO}$ = 0 is shown for illustration purposes.

**Figure 85 — RDQS Mode Timings**

## 7.13 READ and WRITE DATA BUS INVERSION (DBI)

Data Bus Inversion (DBIdc) reduces the DC power consumption on data signals, as the number of DQ lines driving a LOW level can be limited to 4 within a byte. DBIdc is evaluated per byte.

There is one DBI_n per byte: DBI0_n is associated with DQ[7:0] and DBI1_n with DQ[15:8].

DBI_n are bidirectional active LOW double data rate (DDR) signals. For Writes, they are sampled by the device along with the DQ of the same byte. For Reads, they are driven by the device along with the DQ of the same byte.

Once enabled by the corresponding RDBI Mode Register bit, the device inverts read data and sets DBI_n LOW, when the number of '0' data bits within a byte is greater than 4; otherwise the device does not invert the read data and sets DBI_n HIGH, as shown in Figure 86.

Once enabled by the corresponding WDBI Mode Register bit, the device inverts write data received on the DQ inputs in case DBI_n was sampled LOW, or leaves the data non-inverted in case DBI_n was sampled HIGH, as shown in Figure 87.

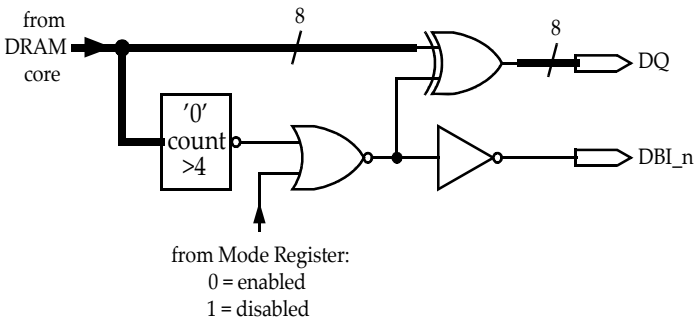



**Figure 86 — Example of Data Bus Inversion Logic for READs**

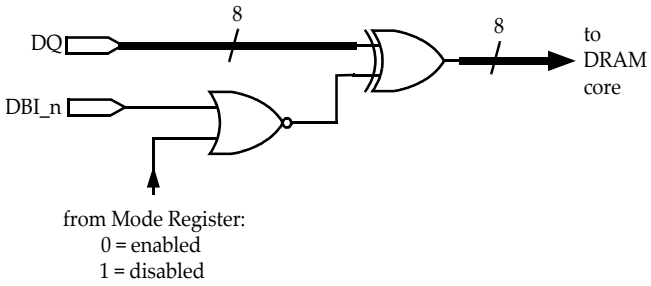



**Figure 87 — Example of Data Bus Inversion Logic for WRITEs**

The flow diagram in Figure 88 illustrates the DBIdc operation. In any case, the transmitter (the controller for WRITEs, the GDDR6 SGRAM for READs) decides whether to invert or not invert the data conveyed on the DQs. The receiver (the GDDR6 SGRAM for WRITEs, the controller for READs) has to perform the reverse operation based on the level on DBI_n. Data input and output timing parameters are only valid with DBI being enabled and a maximum of 4 data lines per byte driven LOW.

## 7.13 READ and WRITE DATA BUS INVERSION (DBI) (cont'd)

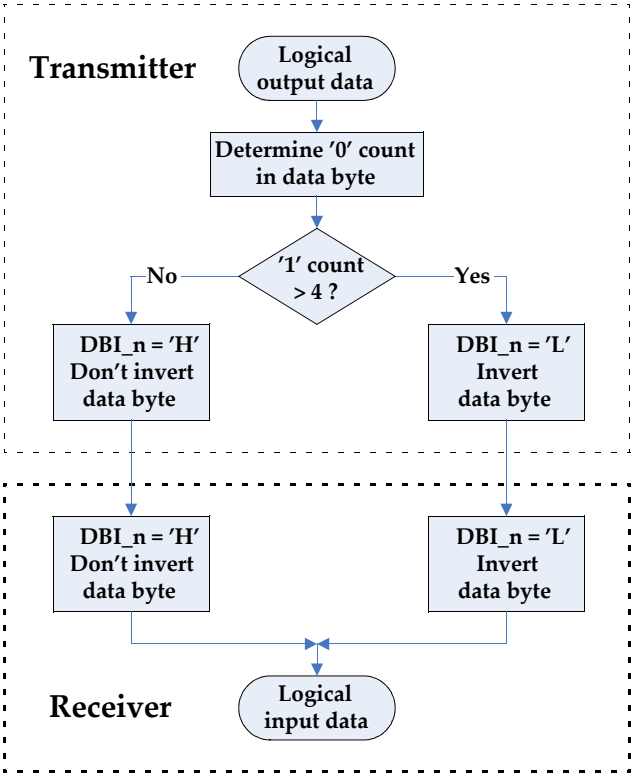



**Figure 88 — DBI Flow Diagram**

### DBI_n Special Function Overview

The DBI_n signal has special behavior compared to DQs because of the ability to enable and disable it via MRS. For either WRITE or READ DBI_n training, both DBI READ and DBI WRITE in MRS must be enabled. The behavior of the DBI_n signals in various mode register settings is summarized below:

If both DBI READ and DBI WRITE are enabled:

- DBI_n drives DBI FIFO data with RDTR command
- DBI_n FIFO accepts WRTR data with the WRTR command

If only DBI READ is enabled:

- DBI_n drives ODT when not READ or RDTR

If only DBI WRITE is enabled:

- DBI_n always drives ODT (unless RESET_n)

If both DBI READ and DBI WRITE are disabled:

- DBI_n drives ODT (unless RESET_n)

## 7.14 ERROR DETECTION CODE (EDC)

The GDDR6 SGRAM provides error detection on the data bus to improve system reliability. The device generates a checksum per byte lane for both READ and WRITE data and returns the checksum to the controller. GDDR6 supports 2 modes for the EDC with either a 16 bit checksum that runs at same rate as the data (Full data rate EDC) or an 8 bit checksum that runs at half the rate of the data (Half data rate EDC). Mode Register 2 OP8 selects between the 2 modes and is undefined at boot. The EDC rate register must be set during the initialization sequence after power up or after a reset.

The 16 bit checksum of the Full data rate EDC is calculated in two halves based on the 2 halves of the 16 bit burst. The first 8 bits of the checksum, CRC-L, are calculated on burst positions 0 thru 7 and the second half of the checksum, CRC-U, are calculated on burst positions 8 thru 15. The 8 bit checksum of the Half data rate EDC is calculated on the full 16bit burst using an XOR of the CRC-L and CRC-U as illustrated in Figure 89.

Based on the checksum, the controller can decide if the data (or the returned CRC) was transmitted in error and retry the READ or WRITE command. The device itself does not perform any error correction. The features of the EDC are:

- 8 bit checksum on 72 bits (9 bit lanes x 8 bit burst) x 2 for 16bit checksum (Full data rate EDC) or 8 bit checksum on 144 bits (9 bit lanes x 16 bit burst) for 8 bit checksum (Half data rate EDC)
- dedicated EDC transfer pin per 9 bit lanes (2x per channel)
- asymmetrical latencies on EDC transfer for READs and WRITEs

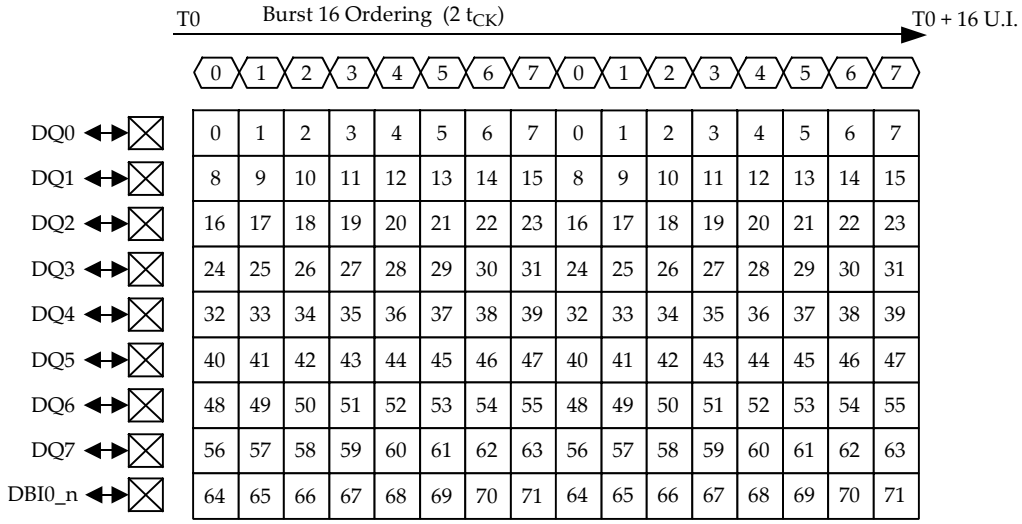
The 8 bit polynomial is 0x83, or X^8+X^2+X^1+1. The starting seed value is set in hardware at “zero”. Table 36 shows the error detection rate for a variety of error types. All error rates are calculated over the entire BL16 of data.

**Table 36 — Error Correction Details**

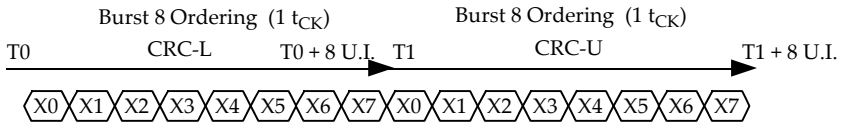
| Error Type | Detection Rate Full data rate | Detection Rate Half data rate |
|---|---|---|
| Random Single Bit | 100 % | 100 % |
| Random Double Bit | 100 % | 100 % |
| Random Triple Bit | 100 % | 99.6 % |
| Random Quad Bit | 99.9 % | 99.6 % |
| Random Odd Count | 100 % | > 99.0 % |
| Burst <= 8 | 100 % | 99.8 % |

## 7.14 ERROR DETECTION CODE (EDC) (cont'd)

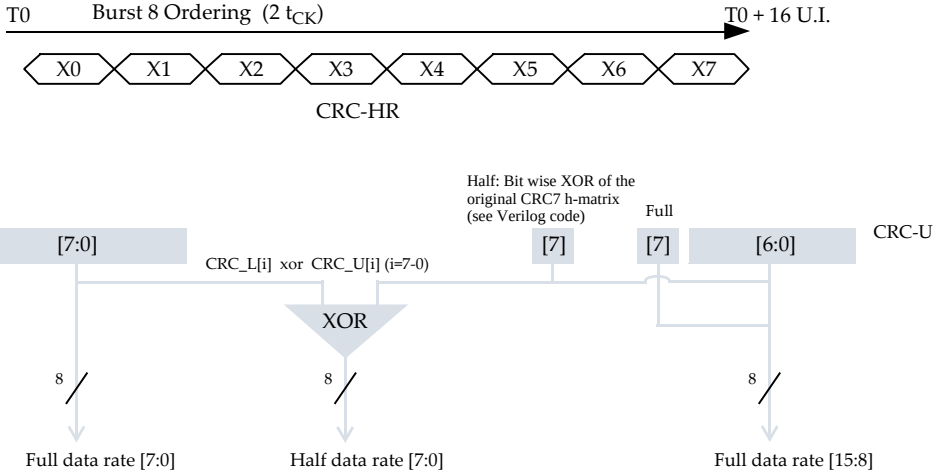
**CRC Data Input**
**DQ/DBI bit ordering**




**Full data rate**
**EDC bit ordering**




**Half data rate**
**EDC bit ordering**




**CRC Polynomial** $X^8 + X^2 + X + 1 = 0 \times 83 = (X + 1)(X^7 + X^6 + X^5 + X^4 + X^3 + X^2 + 1)$

**Figure 89 — EDC Calculation Matrix**

## 7.14 ERROR DETECTION CODE (EDC) (cont'd)

The CRC calculation is embedded into the WRITE and READ data stream as shown in Figure 22:

- for WRITEs, the CRC checksum is calculated on the DQ and DBI_n input data before decoding with DBI
- for READs, the CRC checksum is calculated on the DQ and DBI_n output data after encoding with DBI

The bit ordering is optimized for errors in the time burst direction. Figure 89 shows the bit orientation on a byte lane basis. All '1s' are assumed for DBI_n in the CRC calculation for WRITEs in case WDBI is disabled, and all '1s' are assumed for DBI_n in the CRC calculation for READs in case RDBI is disabled. The CRC calculation is also not affected by any data mask sent along with WDM, WDMA, WSM or WSMA commands.

The EDC latency is based on the READ latency for READ data and the WRITE latency for WRITE data. Table 37 shows the 2 timing parameters associated with the EDC scheme.

Mode Register 4 is used to determine the functionality of the EDC pin. OP[10:9] control the device's CRC calculation independently for READs and WRITEs. With EDC off, the calculated CRC pattern will be replaced by the EDC hold pattern defined in Mode Register 4 OP[3:0]. The EDC hold pattern rate is determined by EDC HR bit. See MODE REGISTERS section for more details.

**Table 37 — EDC Timing**

| Description | Parameter | Value | Units |
|---|---|---|---|
| EDC READ Latency | $t_{EDCRL}$ | RL + CRCRL | $t_{CK}$ |
| EDC WRITE Latency | $t_{EDCWL}$ | WL + CRCWL | $t_{CK}$ |

## 7.14 ERROR DETECTION CODE (EDC) (cont'd)

### EDC Pin Special Function Overview

The EDC pin is used for many different functions. The behavior of the EDC pin in various modes is summarized in Table 38.

**Table 38 — EDC Pin Behavior**

| Device Status | Condition | EDC Pin Status |
|---|---|---|
| Device Power-up | RESET_n = LOW | Hi-Z |
| | RESET_n = HIGH; no WCK clocks | HIGH |
| | RESET_n = HIGH; stable WCK clocks | EDC hold pattern (default = '1111') |
| WCK2CK Training | WCK is sampled HIGH | HIGH |
| | WCK is sampled LOW | LOW |
| Idle | EDCinv (MR4 OP11 = 0) | EDC hold pattern or PRBS |
| | EDCinv (MR4 OP11 = 1) | EDC0: EDC hold pattern or PRBS<br>EDC1: inverted EDC hold pattern or PRBS |
| WRITE Burst | WRCRC on (MR4 OP10 = 0) | CRC data |
| | WRCRC off (MR4 OP10 = 1) | EDC hold pattern or PRBS |
| READ or RDTR burst | RDCRC on (MR4 OP9 = 0) | CRC data |
| | RDCRC off (MR4 OP9 = 1) | EDC hold pattern or PRBS |
| LDFF | WRCRC + RDCRC both on or both off | EDC hold pattern or PRBS |
| WRTR burst | - | EDC hold pattern or PRBS |
| Power-Down | WCK enabled (MR5 OP1 = 0) | EDC hold pattern or PRBS |
| | WCK disabled using MR5 OP1 = 1 (Optional) | HIGH |
| Self Refresh | - | HIGH |
| Vendor ID Mode | WCK is stable | EDC hold pattern or PRBS |
| All except reset | EDC Hi-Z (MR8 OP5 = 1) | Hi-Z |
| CA Training | CADT on (MR15 OP3 =1 or OP2 = 1) | CA data |

NOTE 1 The PRBS feature is optional and set with MR12 OP[3:2].
NOTE 2 The EDC hold pattern related register bits MR2 OP11, MR4 OP11 and MR4 OP[3:0] are not to influence the bit stream when PRBS is enabled.

## 7.14 ERROR DETECTION CODE (EDC) (cont'd)

### CRC Polynomial Logic for Full data rate and Half data rate EDC

Below is an implementation of the 72 bit parallel CRC calculation based on the given polynomial: X^8+X^2+X^1+1. The indices of reg D correspond to the positions in Figure 89.

```
function [7:0] CRC_72TO8;
input [71:0] D;
input        EDC_mode;
reg [7:0]    CRC;
begin
  CRC[0] = D[69] ^ D[68] ^ D[67] ^ D[66] ^ D[64] ^ D[63] ^ D[60] ^ D[56] ^ D[54] ^ D[53] ^ D[52] ^ D[50] ^ D[49] ^
           D[48] ^ D[45] ^ D[43] ^ D[40] ^ D[39] ^ D[35] ^ D[34] ^ D[31] ^ D[30] ^ D[28] ^ D[23] ^ D[21] ^ D[19] ^
           D[18] ^ D[16] ^ D[14] ^ D[12] ^ D[8]  ^ D[7]  ^ D[6]  ^ D[0];
  CRC[1] = D[70] ^ D[66] ^ D[65] ^ D[63] ^ D[61] ^ D[60] ^ D[57] ^ D[56] ^ D[55] ^ D[52] ^ D[51] ^ D[48] ^ D[46] ^
           D[45] ^ D[44] ^ D[43] ^ D[41] ^ D[39] ^ D[36] ^ D[34] ^ D[32] ^ D[30] ^ D[29] ^ D[28] ^ D[24] ^ D[23] ^
           D[22] ^ D[21] ^ D[20] ^ D[18] ^ D[17] ^ D[16] ^ D[15] ^ D[14] ^ D[13] ^ D[12] ^ D[9]  ^ D[6]  ^ D[1]  ^
           D[0];
  CRC[2] = D[71] ^ D[69] ^ D[68] ^ D[63] ^ D[62] ^ D[61] ^ D[60] ^ D[58] ^ D[57] ^ D[54] ^ D[50] ^ D[48] ^ D[47] ^
           D[46] ^ D[44] ^ D[43] ^ D[42] ^ D[39] ^ D[37] ^ D[34] ^ D[33] ^ D[29] ^ D[28] ^ D[25] ^ D[24] ^ D[22] ^
           D[17] ^ D[15] ^ D[13] ^ D[12] ^ D[10] ^ D[8]  ^ D[6]  ^ D[2]  ^ D[1]  ^ D[0];
  CRC[3] = D[70] ^ D[69] ^ D[64] ^ D[63] ^ D[62] ^ D[61] ^ D[59] ^ D[58] ^ D[55] ^ D[51] ^ D[49] ^ D[48] ^ D[47] ^
           D[45] ^ D[44] ^ D[43] ^ D[40] ^ D[38] ^ D[35] ^ D[34] ^ D[30] ^ D[29] ^ D[26] ^ D[25] ^ D[23] ^ D[18] ^
           D[16] ^ D[14] ^ D[13] ^ D[11] ^ D[9]  ^ D[7]  ^ D[3]  ^ D[2]  ^ D[1];
  CRC[4] = D[71] ^ D[70] ^ D[65] ^ D[64] ^ D[63] ^ D[62] ^ D[60] ^ D[59] ^ D[56] ^ D[52] ^ D[50] ^ D[49] ^ D[48] ^
           D[46] ^ D[45] ^ D[44] ^ D[41] ^ D[39] ^ D[36] ^ D[35] ^ D[31] ^ D[30] ^ D[27] ^ D[26] ^ D[24] ^ D[19] ^
           D[17] ^ D[15] ^ D[14] ^ D[12] ^ D[10] ^ D[8]  ^ D[4]  ^ D[3]  ^ D[2];
  CRC[5] = D[71] ^ D[66] ^ D[65] ^ D[64] ^ D[63] ^ D[61] ^ D[60] ^ D[57] ^ D[53] ^ D[51] ^ D[50] ^ D[49] ^ D[47] ^
           D[46] ^ D[45] ^ D[42] ^ D[40] ^ D[37] ^ D[36] ^ D[32] ^ D[31] ^ D[28] ^ D[27] ^ D[25] ^ D[20] ^ D[18] ^
           D[16] ^ D[15] ^ D[13] ^ D[11] ^ D[9]  ^ D[5]  ^ D[4]  ^ D[3];
  CRC[6] = D[67] ^ D[66] ^ D[65] ^ D[64] ^ D[62] ^ D[61] ^ D[58] ^ D[54] ^ D[52] ^ D[51] ^ D[50] ^ D[48] ^ D[47] ^
           D[46] ^ D[43] ^ D[41] ^ D[38] ^ D[37] ^ D[33] ^ D[32] ^ D[29] ^ D[28] ^ D[26] ^ D[21] ^ D[19] ^ D[17] ^
           D[16] ^ D[14] ^ D[12] ^ D[10] ^ D[6]  ^ D[5]  ^ D[4];
  if(EDC_mode === 1'b0) // full data rate
    CRC[7] = D[68] ^ D[67] ^ D[66] ^ D[65] ^ D[63] ^ D[62] ^ D[59] ^ D[55] ^ D[53] ^ D[52] ^ D[51] ^ D[49] ^ D[48] ^
             D[47] ^ D[44] ^ D[42] ^ D[39] ^ D[38] ^ D[34] ^ D[33] ^ D[30] ^ D[29] ^ D[27] ^ D[22] ^ D[20] ^ D[18] ^
             D[17] ^ D[15] ^ D[13] ^ D[11] ^ D[7]  ^ D[6]  ^ D[5];
  if(EDC_mode === 1'b1) // half data rate
    CRC[7] = D[71] ^ D[70] ^ D[69] ^ D[64] ^ D[61] ^ D[60] ^ D[58] ^ D[57] ^ D[56] ^ D[54] ^ D[50] ^ D[46] ^ D[45] ^
             D[43] ^ D[41] ^ D[40] ^ D[37] ^ D[36] ^ D[35] ^ D[32] ^ D[31] ^ D[28] ^ D[26] ^ D[25] ^ D[24] ^ D[23] ^
             D[21] ^ D[19] ^ D[16] ^ D[14] ^ D[12] ^ D[10] ^ D[9]  ^ D[8]  ^ D[4]  ^ D[3]  ^ D[2]  ^ D[1]  ^ D[0];
  CRC_72TO8 = CRC;
  end
endfunction

function [15:0] CRC_GDDR6_FULL;
input   [143:0] D;
reg      [7:0] CRC_L, CRC_U;
begin
  CRC_U = CRC_72TO8(D[143:72],1'b0);
  CRC_L = CRC_72TO8(D[71:0],  1'b0);
  CRC_GDDR6_FULL = {CRC_U, CRC_L};
end
endfunction

function [7:0] CRC_GDDR6_HALF;
input  [143:0] D;
reg     [7:0] CRC_L, CRC_U;
begin
  CRC_U = CRC_72TO8(D[143:72],1'b1);
  CRC_L = CRC_72TO8(D[71:0],  1'b0);
  CRC_GDDR6_HALF = {CRC_U ^ CRC_L};
end
endfunction
```

## 7.15 PRECHARGE

The PRECHARGE command (see Figure 90) is a single cycle command used to deactivate the open row in a particular bank (PREpb) or the open row in all banks (PREab). The bank(s) will be available for a subsequent row access a specified time ($t_{RP}$) after the PRECHARGE command is issued as illustrated in Figure 56.

Input CA4 (falling edge of CK) determines whether one or all banks are to be precharged. In case where only one bank is to be precharged, inputs BA[3:0] select the bank. Otherwise BA[3:0] are treated as “Don’t Care”.

Once a bank has been precharged, it is in the idle state and must be activated prior to any READ or WRITE command being issued. A PRECHARGE command will be treated as a NOP if there is no open row in that bank, or if the previously open row is already in the process of precharging. Sequences of PRECHARGE commands must be spaced by at least $t_{PPD}$.

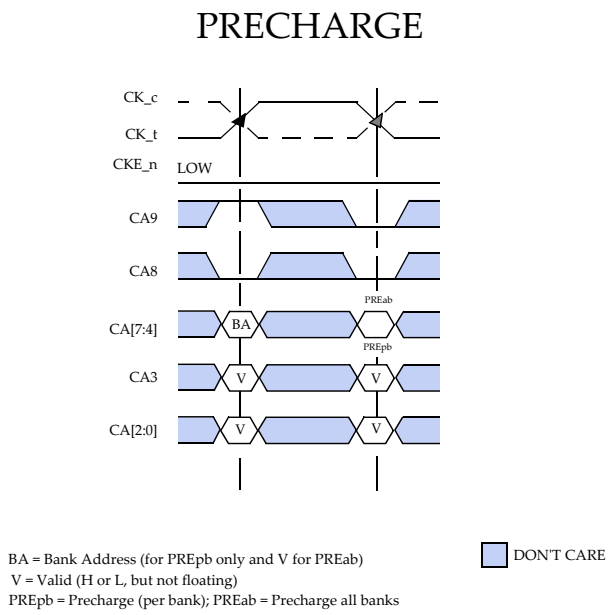



**Figure 90 — PRECHARGE command**

## 7.16 AUTO PRECHARGE

Auto Precharge is a feature which performs the same individual bank precharge function as described above, but without requiring an explicit command. This is accomplished by using CA4 (falling edge of CK = HIGH), to enable Auto Precharge in conjunction with a specific READ or WRITE command. A precharge of the bank / row that is addressed with the READ or WRITE command is automatically performed once the applicable timings $t_{WR}$, $t_{RAS}$ and $t_{RTP}$ have been met as described in the WRITE and READ sections and in Table 49. Auto Precharge is non persistent in that it is either enabled or disabled for each individual READ or WRITE command.

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

## 7.17 REFRESH and PER-BANK / PER-2-BANK REFRESH

The (all-bank) REFRESH (REFab) and the PER-BANK REFRESH (REFpb) / PER-2-BANK REFRESH (REFp2b) commands are used during normal operation. The commands are non persistent, so they must be issued each time a refresh is required. REFab and REFpb / REFp2b commands are distinguished by the level of the CA4 input (falling edge of CK) of the REFRESH command. REFpb and REFp2b are selected by MR8 OP7.

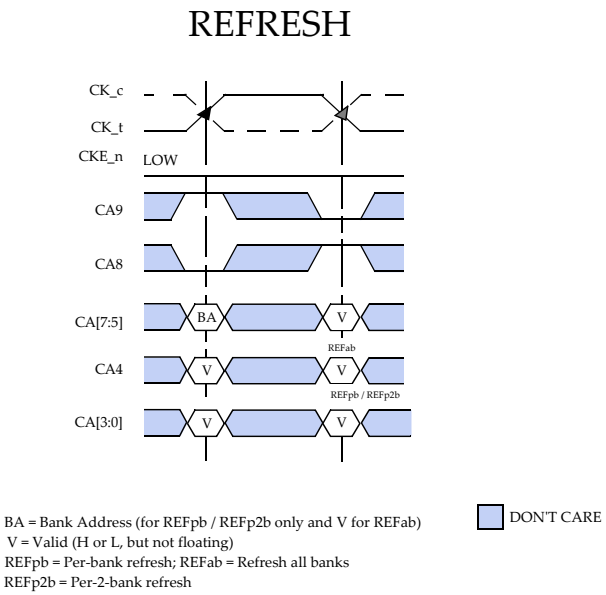



**Figure 91 — REFRESH Command**

### REFRESH (REFab) Command

A minimum time $t_{RFCab}$ is required between two REFab commands. The same rule applies to any access command after the refresh operation. All banks must be precharged prior to the REFab command.

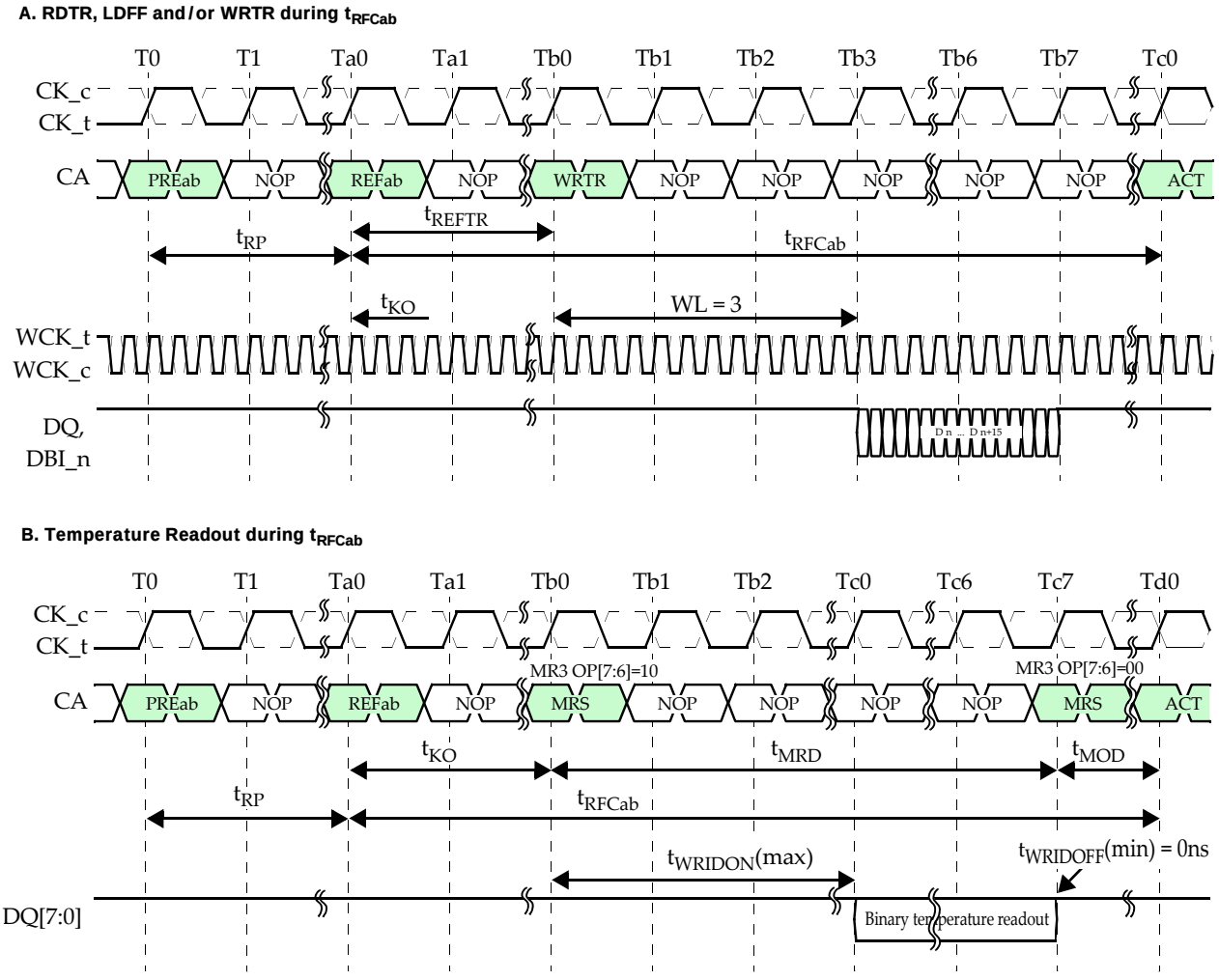
The refresh addressing is generated by the internal refresh controller. This makes the address bits "Don't Care" during a REFab command. The device requires REFab cycles at an average periodic interval of $t_{REFI}$(max). The values of $t_{REFI}$ for different densities are listed in Table 15.

To allow for improved efficiency in scheduling and switching between tasks, some flexibility in the absolute refresh interval is provided. A maximum of 8 REFab commands can be postponed during operation of the device, meaning that at no point in time more than a total of 8 REFab commands are allowed to be postponed. In case that 8 REFab commands are postponed in a row, the resulting maximum interval between the surrounding REFab commands is limited to 9 × $t_{REFI}$ (see Figure 93). A maximum of 8 additional REFab commands can be issued in advance (“pulled in”), with each one reducing the number of regular REFab commands required later by one. Note that pulling in more than 8 REFab commands in advance does not further reduce the number of regular REFab commands required later, so that the resulting maximum interval between two surrounding REFab commands is limited to 9 × $t_{REFI}$ (see Figure 94). At any given time, a maximum of 9 REFab commands can be issued within $t_{REFI}$.

## 7.17 REFRESH and PER-BANK / PER-2-BANK REFRESH (cont'd)

During REFab, and when OP2 in MR5 is set to '0', WRTR, RDTR, and LDFF commands are allowed at time $t_{REFTR}$ after the REFab command, which enable (incremental) data training to occur in parallel with the internal refresh operation and thus without loss of performance on the interface as illustrated in Figure 92A. See READ Training and WRITE Training for details. Additionally, when OP2 in MR5 is set to 0, MRS commands to MR3 OP[7:6] to readout the temperature of the device are allowed as illustrated in Figure 92B. The ability to support temperature readout while MR5 OP2 = 0 is vendor specific, therefore, vendor datasheets should be consulted. MRS commands during $t_{RFCab}$ that make changes to any register other than MR3 OP[7:6] may result in undefined operation.

As impedance updates from the auto-calibration engine may occur with any REFab command, it is safe to only issue NOP commands during $t_{KO}$ period to prevent false command, address or data latching resulting from impedance updates.




NOTES:

1. WRTR, LDFF and RDTR command or MRS for Temp readout are allowed during refresh unless disabled in the Mode Register.
2. WCK and CK are shown aligned ($t_{WCK2CKPIN}$=0, $t_{WCK2CK}$=0) for illustration purposes. WCK2CK training determines the needed offset between WCK and CK.
3. Figure shown with WRTR training and temperature readout durations as examples.

**Figure 92 — REFab Timings**

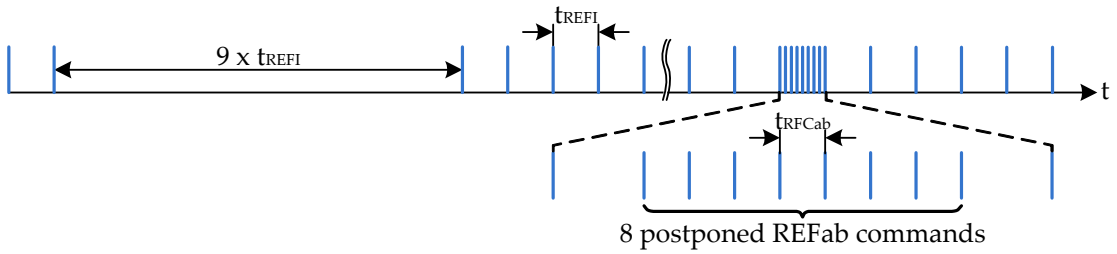



**Figure 93 — Postponing REFab Commands (Example)**

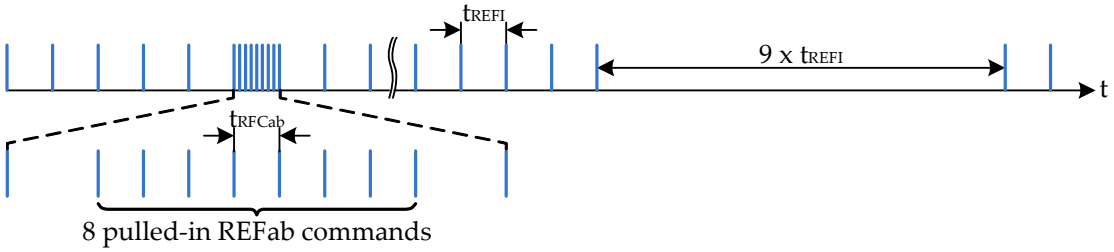



**Figure 94 — Pulling-In REFab Commands (Example)**

## 7.17 REFRESH and PER-BANK / PER-2-BANK REFRESH (cont'd)

### PER-BANK REFRESH (REFpb) and PER-2-BANK REFRESH (REFp2b) Commands

The REFpb / REFp2b command provides an alternative for the refresh of the device. The REFpb command initiates a refresh cycle on a single bank selected by BA[3:0] while accesses to other banks including writes and reads are not affected. The REFp2b command initiates a refresh cycle on a bank selected by BA[3:0] and it's paired bank while accesses to other banks including writes and reads are not affected. Banks for refresh can be selected by MR12 OP1 which determines paired banks by "Don't Care" MSB or LSB. The bank pairing for LSB is shown in Table 40 and for the optional MSB is shown in Table 41. For devices that support both LSB and MSB, MR12 OP1 allows the desired mode to be programmed.

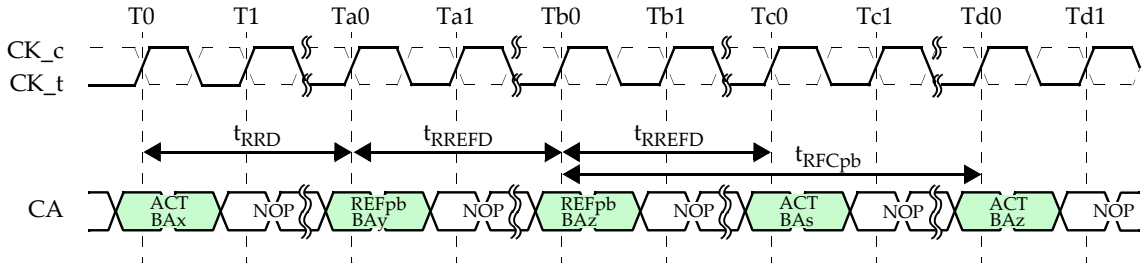
A minimum time $t_{RRD}$ is required between an ACTIVATE command and a REFpb / REFp2b command to a different bank. A minimum time $t_{RREFD}$ is required between any two REFpb / REFp2b commands (see below for an exception requiring $t_{RFCpb}$), and between a REFpb / REFp2b command and an ACTIVATE command to a different bank. A minimum time $t_{RFCpb}$ is required between a REFpb / REFp2b command and an access command to the same bank that follows. The selected bank must be precharged prior to the REFpb / REFp2b command. The row address is generated by an internal counter. This makes the row address bits "Don't Care" during a REFpb / REFp2b command.

A REFpb / REFp2b command can be issued in any order. After all 16 banks have been refreshed using the REFpb / REFp2b command, and after waiting for at least $t_{RFCpb}$, the internal refresh counter is incremented and the controller can issue another set of REFpb / REFp2b commands in the same or a different order. However, it is illegal to send another REFpb / REFp2b command to a bank unless all 16 banks have been refreshed using the REFpb / REFp2b command. The controller must track the banks being refreshed by the REFpb / REFp2b command.

The bank count is synchronized between the controller and the device by resetting the bank count to zero. Synchronization occurs upon exit from reset state or by issuing a REFab or SELF REFRESH ENTRY command. Both commands may be issued at any time even if a preceding sequence of REFpb / REFp2b commands has not completed cycling through all 16 banks. The internal refresh counter is not incremented in case of such incomplete cycling. It is pointed out that multiple occurrences of synchronization events without refresh counter increment may result in an insufficient refresh of the memory array; it is suggested to issue additional REFab commands in that case.

The average rate of REFpb commands $t_{REFIpb}$ is given by $t_{REFI}$ / 16 when MR8 OP7 = 0. The average rate of REFp2b commands $t_{REFIpb}$ is given by $t_{REFI}$ / 8 when MR8 OP7=1.

REFab commands must also be issued during normal operation at a minimum rate of $t_{ABREF}$ = 1ms to allow impedance updates from the auto-calibration engine to occur.




BA = Bank Address
REFpb can be either REFpb or REFp2b depending on MR8 OP7

**Figure 95 — REFpb / REFp2b Timings**

## 7.17 REFRESH and PER-BANK / PER-2-BANK REFRESH (cont'd)

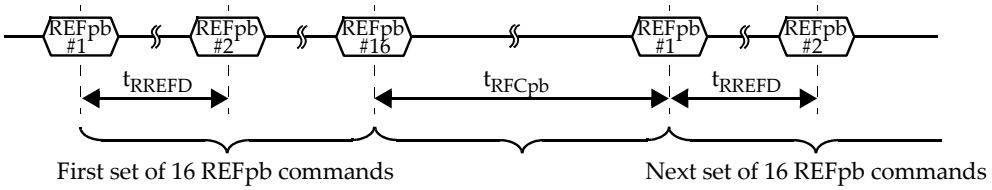



**Figure 96 — Sets of REFpb Commands**

The example in Table 39 shows two full sets of REFpb commands with the bank counter reset to zero and the refresh counter incremented after 16 REFpb commands each. The 3rd set to REFpb commands is interrupted by the REFab command which resets the bank counter to 0 and performs refreshes to all banks indicated by the refresh counter.

**Table 39 — Refresh Counter Increments - REFpb**

| CNT | SUB-CNT | COMMAND | BA[3:0] | REFRESH BANK | BANK COUNTER | REFRESH COUNTER |
|---|---|---|---|---|---|---|
| 0 | 0 | Reset, REFab or SRE command | | | To 0 | |
| 1 | 1 | REFpb | 0000 | 0 | 0 to 1 | |
| 2 | 2 | REFpb | 0001 | 1 | 1 to 2 | |
| 3 | 3 | REFpb | 0010 | 2 | 2 to 3 | |
| 4 | 4 | REFpb | 0011 | 3 | 3 to 4 | n |
| . . . | | | | | | |
| 15 | 15 | REFpb | 1110 | 14 | 14 to 15 | |
| 16 | 16 | REFpb | 1111 | 15 | 15 to 0 | |
| 17 | 1 | REFpb | 0100 | 4 | 0 to 1 | |
| 18 | 2 | REFpb | 0111 | 7 | 1 to 2 | |
| 19 | 3 | REFpb | 1011 | 11 | 2 to 3 | |
| 20 | 4 | REFpb | 0110 | 6 | 3 to 4 | n + 1 |
| . . . | | | | | | |
| 31 | 15 | REFpb | 1100 | 12 | 14 to 15 | |
| 32 | 16 | REFpb | 0001 | 1 | 15 to 0 | |
| 33 | 1 | REFpb | 0010 | 2 | 0 to 1 | |
| 34 | 2 | REFpb | 1001 | 9 | 1 to 2 | n + 2 |
| 35 | 3 | REFpb | 0000 | 0 | 2 to 3 | |
| 36 | 0 | **REFab** | **V** | **all** | **To 0** | **n + 2** |
| 37 | 1 | REFpb | 1010 | 10 | 0 to 1 | n + 3 |
| 38 | 2 | REFpb | 0101 | 5 | 1 to 2 | |
| . . . | | | | | | |

## 7.17 REFRESH and PER-BANK / PER-2-BANK REFRESH (cont'd)

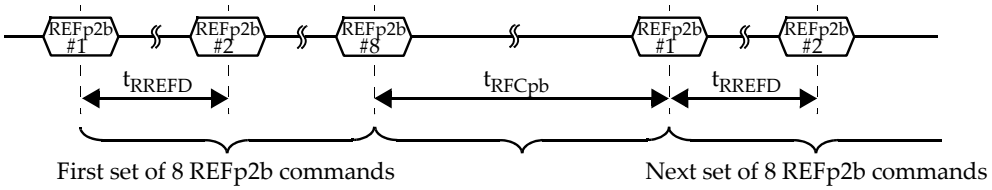



**Figure 97 — Sets of REFp2b Commands**

The examples in Table 40 and Table 41 show two full sets of REFp2b commands with the bank counter reset to zero and the refresh counter incremented after 8 REFp2b commands each. The 3rd set to REFp2b commands is interrupted by the REFab command which resets the bank counter to 0 and performs refreshes to all banks indicated by the refresh counter. The 3rd set also shows the requirement that a set must either be complete or a REFab to reset the bank counter back to 0 before any change to MR8 OP7. The 4th set uses REFpb.

**Table 40 — Refresh Counter Increments - REFp2b (LSB)**

| CNT | SUB-CNT | COMMAND | BA[3:0] | REFRESH BANKS | BANK COUNTER | REFRESH COUNTER |
|---|---|---|---|---|---|---|
| 0 | 0 | Reset, REFab or SRE command followed by MRS to set MR8 OP7 = 1 and MR12 OP1 = 0 | | | To 0 | |
| 1 | 1 | REFp2b | 0000 | 0, 1 | 0 to 1 | n |
| 2 | 2 | REFp2b | 0010 | 2, 3 | 1 to 2 | |
| 3 | 3 | REFp2b | 0100 | 4, 5 | 2 to 3 | |
| 4 | 4 | REFp2b | 0110 | 6, 7 | 3 to 4 | |
| 5 | 5 | REFp2b | 1000 | 8, 9 | 4 to 5 | |
| 6 | 6 | REFp2b | 1010 | 10, 11 | 5 to 6 | |
| 7 | 7 | REFp2b | 1100 | 12, 13 | 6 to 7 | |
| 8 | 8 | REFp2b | 1110 | 14, 15 | 7 to 0 | |
| 9 | 1 | REFp2b | 0100 | 4, 5 | 0 to 1 | n + 1 |
| 10 | 2 | REFp2b | 0111 | 7, 6 | 1 to 2 | |
| 11 | 3 | REFp2b | 1011 | 11, 10 | 2 to 3 | |
| 12 | 4 | REFp2b | 1001 | 9, 8 | 3 to 4 | |
| . . . | | | | | | |
| 15 | 7 | REFp2b | 1100 | 12, 13 | 6 to 7 | |
| 16 | 8 | REFp2b | 1111 | 15, 14 | 7 to 0 | |
| 17 | 1 | REFp2b | 0010 | 2, 3 | 0 to 1 | n + 2 |
| 18 | 2 | REFp2b | 1001 | 9, 8 | 1 to 2 | |
| 19 | 3 | REFp2b | 0000 | 0, 1 | 2 to 3 | |
| 20 | 0 | **REFab** | **V** | **all** | **To 0** | **n + 2** |
| 21 | 0 | MRS (MR8 OP7 = 0) | V | - | 0 | n + 2 |
| 22 | 1 | REFpb | 1010 | 10 | 0 to 1 | n + 3 |
| 23 | 2 | REFpb | 0101 | 5 | 1 to 2 | |
| . . . | | | | | | |

## 7.17 REFRESH and PER-BANK / PER-2-BANK REFRESH (cont'd)

**Table 41 — Refresh Counter Increments - REFp2b (MSB)**

| CNT | SUB-CNT | COMMAND | BA[3:0] | REFRESH BANKS | BANK COUNTER | REFRESH COUNTER |
|---|---|---|---|---|---|---|
| 0 | 0 | Reset, REFab or SRE command followed by MRS to set MR8 OP7=1 and MR12 OP1 = 1 | | | To 0 | |
| 1 | 1 | REFp2b | 0000 | 0, 8 | 0 to 1 | n |
| 2 | 2 | REFp2b | 0001 | 1, 9 | 1 to 2 | |
| 3 | 3 | REFp2b | 0010 | 2, 10 | 2 to 3 | |
| 4 | 4 | REFp2b | 0011 | 3, 11 | 3 to 4 | |
| 5 | 5 | REFp2b | 0100 | 4, 12 | 4 to 5 | |
| 6 | 6 | REFp2b | 0101 | 5 13 | 5 to 6 | |
| 7 | 7 | REFp2b | 0110 | 6, 14 | 6 to 7 | |
| 8 | 8 | REFp2b | 0111 | 7, 15 | 7 to 0 | |
| 9 | 1 | REFp2b | 0100 | 4, 12 | 0 to 1 | n + 1 |
| 10 | 2 | REFp2b | 0111 | 7, 15 | 1 to 2 | |
| 11 | 3 | REFp2b | 1011 | 11, 3 | 2 to 3 | |
| 12 | 4 | REFp2b | 1001 | 9, 1 | 3 to 4 | |
| . . . | | | | | | |
| 15 | 7 | REFp2b | 1100 | 12, 4 | 6 to 7 | |
| 16 | 8 | REFp2b | 1110 | 14, 6 | 7 to 0 | |
| 17 | 1 | REFp2b | 0010 | 2, 10 | 0 to 1 | n + 2 |
| 18 | 2 | REFp2b | 1001 | 9, 1 | 1 to 2 | |
| 19 | 3 | REFp2b | 0000 | 0, 8 | 2 to 3 | |
| 20 | 0 | **REFab** | **V** | **all** | **To 0** | **n + 2** |
| 21 | 0 | MRS (MR8 OP7 = 0) | V | - | 0 | n + 2 |
| 22 | 1 | REFpb | 1010 | 10 | 0 to 1 | n + 3 |
| 23 | 2 | REFpb | 0101 | 5 | 1 to 2 | |
| . . . | | | | | | |

## 7.17 REFRESH and PER-BANK / PER-2-BANK REFRESH (cont'd)

### Table 42 — REFab and REFp2b Command Scheduling Requirements

| From Command | To Command | Minimum Delay between "From Command" to "To Command" | Notes |
|---|---|---|---|
| REFab | REFab or SELF REFRESH ENTRY | $t_{RFCab}$ | |
| | REFpb / REFp2b (any bank) | $t_{RFCab}$ | |
| | ACTIVATE (any bank) | $t_{RFCab}$ | |
| REFpb / REFp2b | REFab or SELF REFRESH ENTRY | $t_{RFCpb}$ | |
| | REFpb / REFp2b (other bank) | $t_{RREFD}$ or $t_{RFCpb}$ | 2 |
| | REFpb / REFp2b (same bank) | $t_{RFCpb}$ | 1 |
| | ACTIVATE (other bank) | $t_{RREFD}$ | |
| | ACTIVATE (same bank) | $t_{RFCpb}$ | |
| ACTIVATE | REFab or SELF REFRESH ENTRY | $t_{RC}$ | 3 |
| | REFpb / REFp2b (other bank) | $t_{RRD}$ | 4 |
| | REFpb / REFp2b (same bank) | $t_{RC}$ | 3 |

NOTE 1 Back-to-back REFpb / REFp2b commands to the same bank are only allowed when the first REFpb / REFp2b command completes a set of 16 per-bank / 8 per-2-bank refresh operations and the second REFpb / REFp2b command initiates the next set of 16 per-bank / 8 per-2-bank refresh operations.

NOTE 2 Use $t_{RFCpb}$ when the first REFpb/ REFp2b command completes a set of 16 per-bank / 8 per-2-bank refresh operations and the second REFpb / REFp2b command initiates the next set of refresh operations. Use $t_{RREFD}$ in all other cases.

NOTE 3 A bank must be in the idle state with $t_{RP}$ satisfied before it is refreshed.

NOTE 4 $t_{FAW}$ parameters must be observed as well.

## 7.18 SELF REFRESH

Self Refresh can be used to retain data in the device, even if the rest of the system is powered down. Self Refresh can also be used to retain data in only one channel while the other channel is in normal operation. The full power savings can only be achieved when both channels are in Self Refresh. When in the Self Refresh mode, the device retains data without external clocking. The SELF REFRESH ENTRY command (see Figure 96) is initiated like a REFRESH command except that CKE_n is pulled HIGH.

SELF REFRESH ENTRY is only allowed when all banks are precharged with $t_{RP}$ satisfied, and when the last data element or CRC data element from a preceding READ or WRITE command has been pushed out ($t_{RDSRE}$ or $t_{WRSRE}$). NOP commands are required until $t_{CKSRE}$ is met after the entering Self Refresh. The PLL/DLL is automatically disabled upon entering Self Refresh and is automatically enabled and reset upon exiting Self Refresh. If the device enters Self Refresh with the PLL/DLL disabled, it will exit Self Refresh with the PLL/DLL disabled.

Once the SELF REFRESH ENTRY command is registered, CKE_n must be held HIGH to keep the device in Self Refresh mode. When the device has entered the Self Refresh mode, all external control signals, except CKE_n and RESET_n are "Don't care". For proper Self Refresh operation, all power supply ($V_{DD}$, $V_{DDQ}$, $V_{PP}$) and $V_{REFC}$ must be at valid levels. The device initiates a minimum of one internal refresh within $t_{SRF}$ period once it enters Self Refresh mode. The CA, data and WCK inputs are in ODT state, and the EDC signals drive a HIGH.

The CK clock is internally disabled during Self Refresh operation to save power, only if both channels are in the Self Refresh state. The WCK clock(s) are also internally disabled during SRF on each channel independently. The user may change the external clock frequency or halt the external CK clock $t_{CKSRE}$ after SELF REFRESH ENTRY is registered on both channels as shown in Figure 97a. Likewise, the host may halt the external WCK clock(s) in a channel $t_{CKSRE}$ after SELF REFRESH ENTRY is registered on that channel. The minimum time that the channel or device must remain in Self Refresh mode is $t_{SRF}$

The procedure for exiting Self Refresh requires a sequence of events. First, the CK clocks must be stable $t_{CKSRX}$ prior to CKE_n going back LOW. WCK clocks could be on or off at SELF REFRESH EXIT depending on the preferred method for resetting the WCK divider in WCK2CK training, which is required upon SELF REFRESH EXIT. For frequency supported by each method, the vendor datasheet should be consulted. A delay of at least $t_{XS}$ must be satisfied before a valid command can be issued to the device to allow for completion of any internal refresh in progress.

During Self Refresh the on-die termination (ODT) and driver will not be auto-calibrated. Recalibration may be automatically performed within $t_{XS}$ after self refresh exit and if not the recalibration will take place on the next REFab command. During this period, commands other than NOP shall be issued with caution, depending on anticipated or measured voltage and temperature changes. No recalibration is performed when calibration update has been set to off in MR1 OP6.

Upon exit from Self Refresh, the device can be put back into Self Refresh mode after waiting at least $t_{XS}$ period and issuing one extra REFab command.

## 7.18 SELF REFRESH (cont'd)

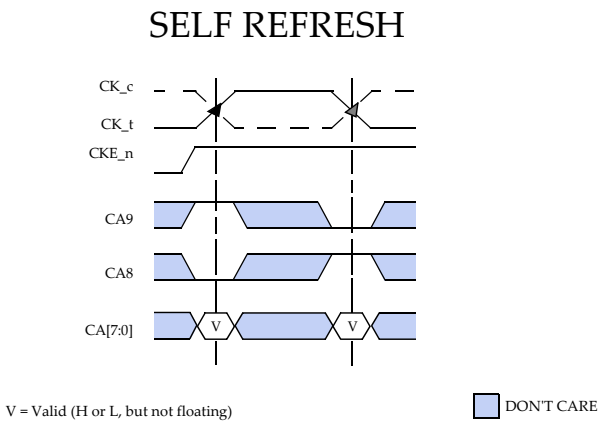


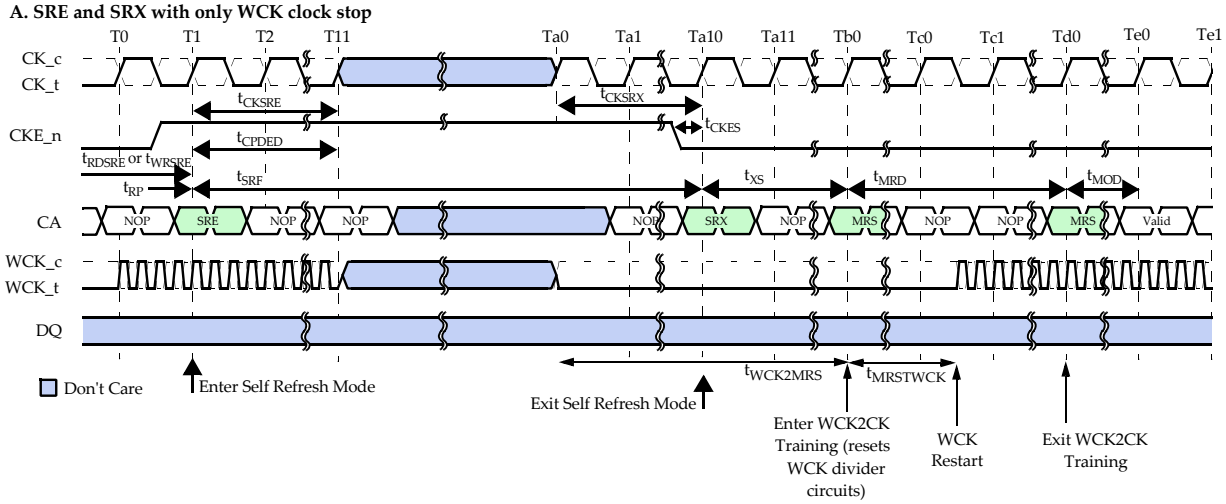

**Figure 98 — SELF REFRESH Entry Command**

**Table 43 — Pin States During Self Refresh**

| Pin | State |
|---|---|
| EDC | HIGH |
| DQ/DBI_n | ODT |
| CA, CABI_n | ODT |
| CKE_n | ODT (Driven HIGH by Controller) |
| WCK_t/WCK_c | ODT |
| CK_t/CK_c | ODT |

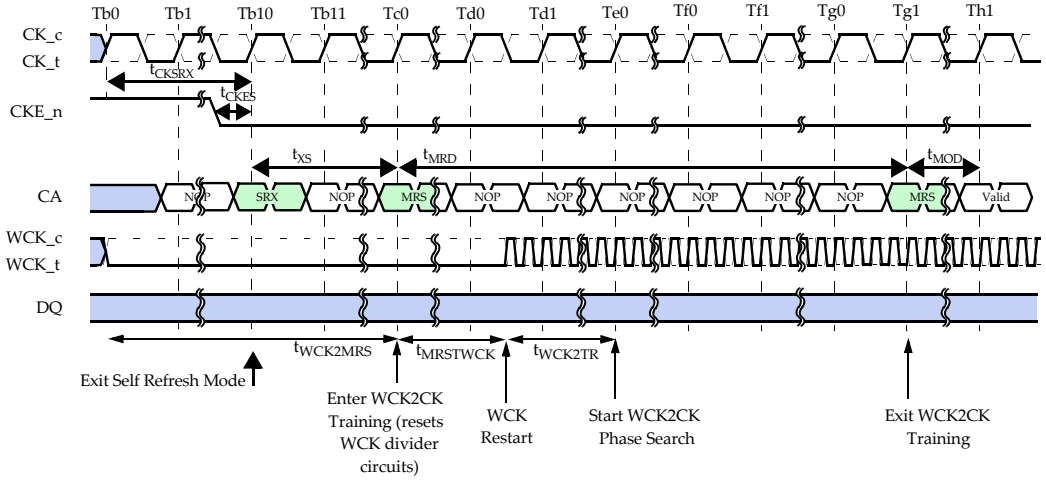
## 7.18 SELF REFRESH

**A. SRE and SRX with only WCK clock stop**




Note: Stopping the CK clock is only allowed if both channels are put in Self Refresh or Hibernate Self Refresh mode.

**B. SRX with WCK clock stop and Phase Search**




**C. SRX without WCK clock stop**

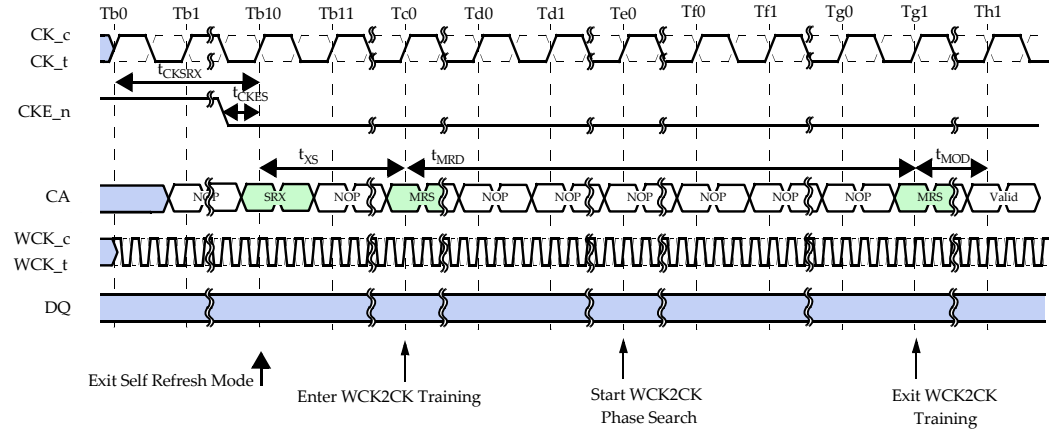



**Figure 99 — Self Refresh Entry and Exit**

## 7.19 PARTIAL ARRAY SELF REFRESH (PASR)

The GDDR6 SGRAM can be programmed to exclude parts of the memory array from refresh when the channel is held in Self Refresh mode. OP[11:0] in MR11 are associated with the Partial Array Self Refresh (PASR) feature.

### PASR Bank Masking

Two banks can individually be configured to be excluded from refresh in Self Refresh mode by programming the bank mask bit(s) in MR11 OP[7:0]. The banks are excluded from refresh when the corresponding 2-bank mask bit is set to “1”. When a 2-bank mask bit is set to “0”, a refresh to the banks is determined by the status of the row segment mask bits as described below.

### PASR Row Segment Masking

The row address space of a channel is virtually divided into 4 row segments along the two MSB row address bits. Each row segment can individually be configured to be excluded from refresh in Self Refresh mode by programming the row segment mask bit(s) in MR11 OP[11:8]. An entire row segment across all 16 banks is excluded from refresh when the corresponding row segment mask bit is set to “1”. When a row segment mask bit is set to “0”, a refresh to the row segment in each group of 2 banks is determined by the status of the corresponding 2-bank mask bits as described above.

An example of using the PASR bank and row segment masking is shown in Table 44.

**Table 44 — Example of PASR 2-Bank and Row Segment Masking in Self Refresh Mode**

| | Row Segment Mask MR11 OP[11:8] | Banks [15:14] | Banks [13:12] | Banks [11:10] | Banks [9:8] | Banks [7:6] | Banks [5:4] | Banks [3:2] | Banks [1:0] |
|---|---|---|---|---|---|---|---|---|---|
| **2-Bank Mask MR11 OP[7:0]** | | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| **Row Segment 0** | 0 | | | | | | M | | |
| **Row Segment 1** | 0 | | | | | | M | | |
| **Row Segment 2** | 1 | M | M | M | M | M | M | M | M |
| **Row Segment 3** | 0 | | | | | | M | | |

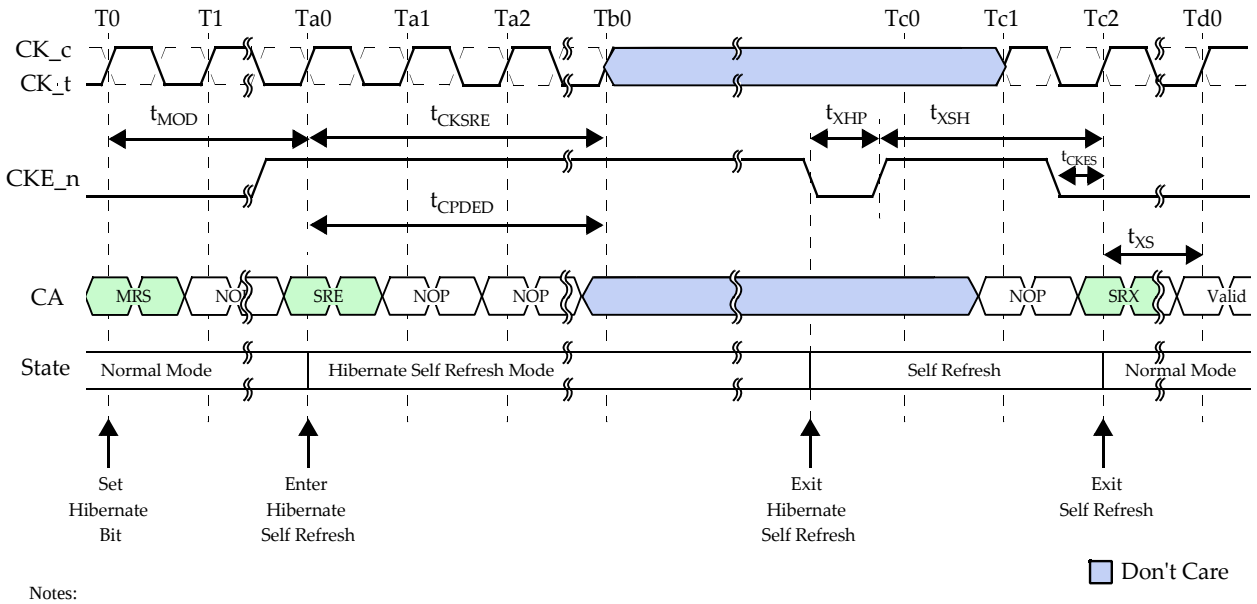
Note: Refresh operation to bank 4 and 5 as well as to row segment 2 in all banks is masked.

## 7.20 HIBERNATE SELF REFRESH

Hibernate Self Refresh is a special mode that provides the same data retention as in the regular Self Refresh mode, but allows the channel to disable additional circuits to achieve an even lower power consumption at the expense of a significantly extended period to return to normal operation. The full power savings can only be achieved when both channels are in Hibernate Self Refresh.

MR7 OP1 is associated with Hibernate Self Refresh. The bit is self-clearing, meaning that an MRS command must set this bit any time the device shall enter Hibernate Self Refresh using the SELF REFRESH ENTRY command.

After the SELF REFRESH ENTRY command is registered, CKE_n must be held HIGH to keep the channel in Hibernate Self Refresh mode. Exiting this mode requires a sequence of events as shown in Figure 98: at first CKE_n must be pulled LOW and again pulled HIGH after $t_{XHP}$ has elapsed. The channel is now in regular Self Refresh mode, and must be held in this mode for at least $t_{XSH}$ period, to retain data during the extended exit time from Hibernate Self Refresh mode. After $t_{XSH}$ period the sequence for SELF REFRESH EXIT shall be followed to return to normal operation.




Notes:
1) Stopping the CK clock is only allowed if both channels are put in Self Refresh or Hibernate Self Refresh mode.
2) Minimum CKE_n pulse width must satisfy $t_{CKE}$.

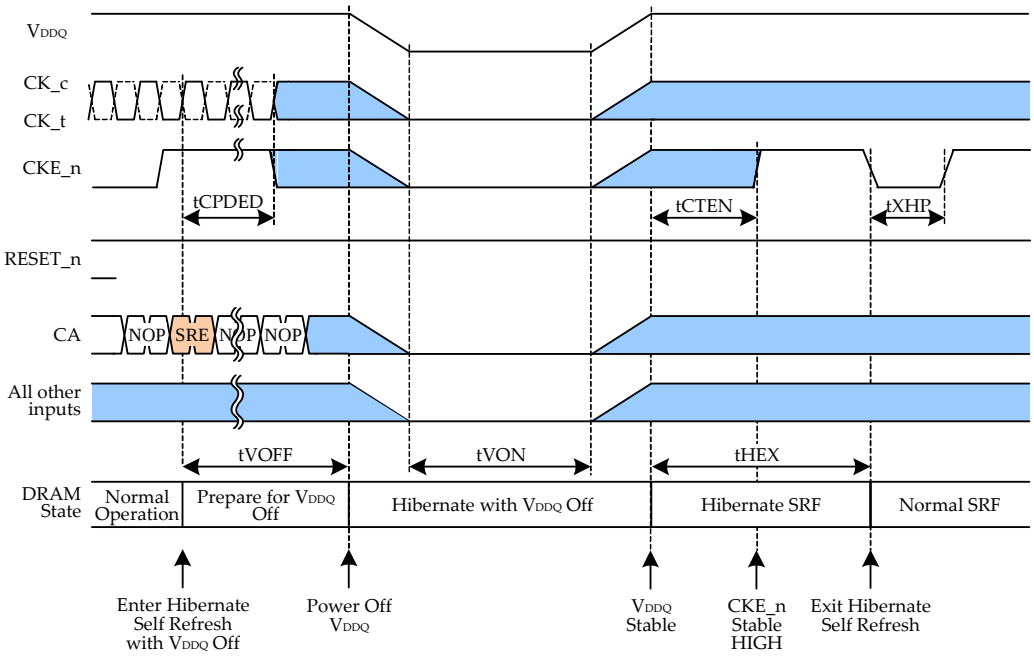
**Figure 100 — Hibernate Self Refresh**

## 7.21 HIBERNATE SELF REFRESH WITH VDDQ OFF

To further reduce the power consumption in self refresh mode, GDDR6 SGRAMs optionally allows to completely power off $V_{DDQ}$ when the device is held in hibernate self refresh. The $V_{DDQ}$ Off Mode Register bit MR12 OP0 is associated with this function; the bit is self-clearing, meaning that an MRS command must set this bit any time the device shall power off $V_{DDQ}$. The device indicates the support of this feature in the Vendor ID, bit 15 of ID1.

The following sequence is required to enter and exit this mode:

1) Issue MRS command(s) on both channels to set the $V_{DDQ}$ Off and Hibernate SRF bits and wait for $t_{MOD}$.
2) Issue a SELF REFRESH ENTRY command followed by NOPs for $t_{CPDED}$. After $t_{CPDED}$ the device is held in hibernate self refresh regardless of the level of the CKE_n signal.
3) After meeting $t_{VOFF}$ timing it is safe to power off $V_{DDQ}$. The voltage level at all device inputs except RESET_n has to be kept at or below $V_{DDQ}$ at all times of the $V_{DDQ}$ power off and on transitions. The RESET_n input must be pulled HIGH to prevent an unwanted chip reset. Maintain $V_{DD}$ and $V_{PP}$ at valid levels. Maintain $V_{DDQ}$ off (below 0.1 x $V_{DD}$) for at least $t_{VON}$.
4) For exiting this mode, apply power to $V_{DDQ}$. The device will detect the $V_{DDQ}$ stable state when $V_{DDQ}$ reaches 0.9 x $V_{DD}$ and will subsequently return to (regular) Hibernate Self Refresh.
5) Drive CKE_n to HIGH latest at time $t_{CTEN}$ after $V_{DDQ}$ is stable (>90% of $V_{DD}$). The device will re-enable the CKE_n receiver after $t_{CTEN}$ has elapsed. Maintain CKE_n HIGH until $t_{HEX}$ has elapsed.
6) Hibernate self refresh can be exited a time $t_{HEX}$ after $V_{DDQ}$ is stable. Follow the procedure described above to exit hibernate self refresh and return to normal operation.




Note: Minimum CKE_n pulse width must satisfy $t_{CKE}$.

**Figure 101 — Hibernate Self Refresh with $V_{DDQ}$ Off Entry and Exit**

## 7.22 POWER-DOWN

GDDR6 SGRAMs requires CKE_n to be LOW at all times an access is in progress: from the issuing of a READ or WRITE command until completion of the burst. For READs, a burst completion is defined as when the last data element including CRC has been transmitted on the DQ outputs, for WRITEs, a burst completion is defined as when the last data element has been written to the memory array and CRC data has been returned to the controller. CKE_n is also required to be LOW for a time $t_{ACTPDE}$ after an ACTIVATE command, for a time $t_{PREPDE}$ after a PRECHARGE command, for a time $t_{REFPDE}$ after a REFRESH command, or for a time $t_{MRSPDE}$ after a MODE REGISTER SET command.

Power-Down is entered when CKE_n is registered HIGH. If POWER-DOWN occurs when all banks are idle, this mode is referred to as precharge power-down; if POWER-DOWN occurs when there is a row active in any bank, this mode is referred to as active power-down. Entering POWER-DOWN deactivates the input and output buffers, excluding CK_t, CK_c, WCK_t, WCK_c, RESET_n, EDC signals and CKE_n if LP2 bit (MR5 OP1) is set LOW. If LP2 bit is set to HIGH, WCK_t, WCK_c input buffers and EDC output buffer are additionally de-activated in power-down.

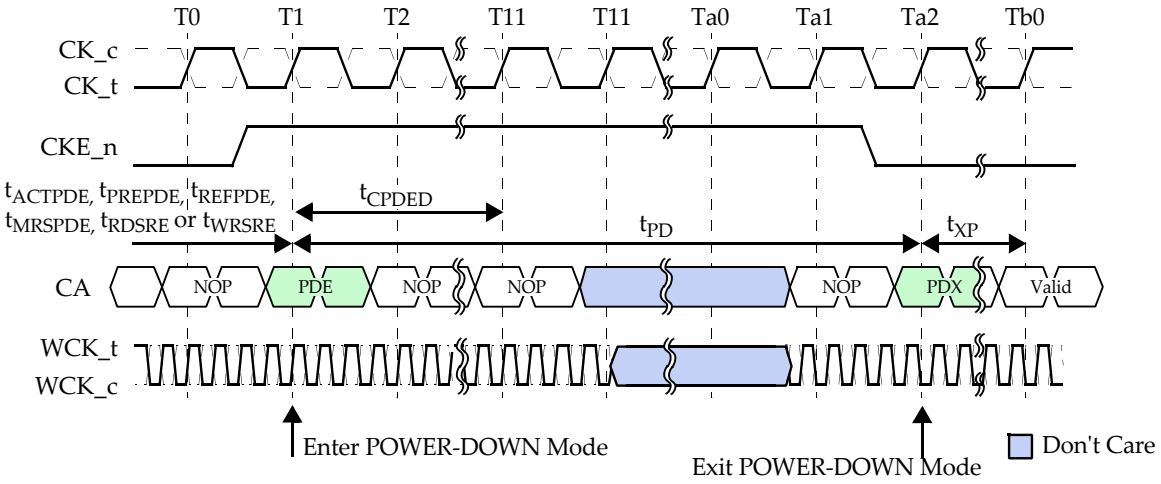
For maximum power savings, the user has the option of disabling the PLL/DLL prior to entering POWER-DOWN. In that case, on exiting POWER-DOWN, WCK2CK training is required to set the internal synchronizers which will include the enabling of the PLL/DLL, PLL/DLL reset, and $t_{LK}$ clock cycles must occur before any READ or WRITE command can be issued.

While in power-down, CKE_n HIGH, a stable CK and RESET_n signals must be maintained at the device inputs if LP2 bit (MR5 OP1) is set to HIGH as shown in Figure 100A. If LP2 bit (MR5 OP1) is set to off, WCK signals must also be maintained in power-down and DRAM continuously drive the EDC hold pattern on the EDC balls as shown in Figure 100B.

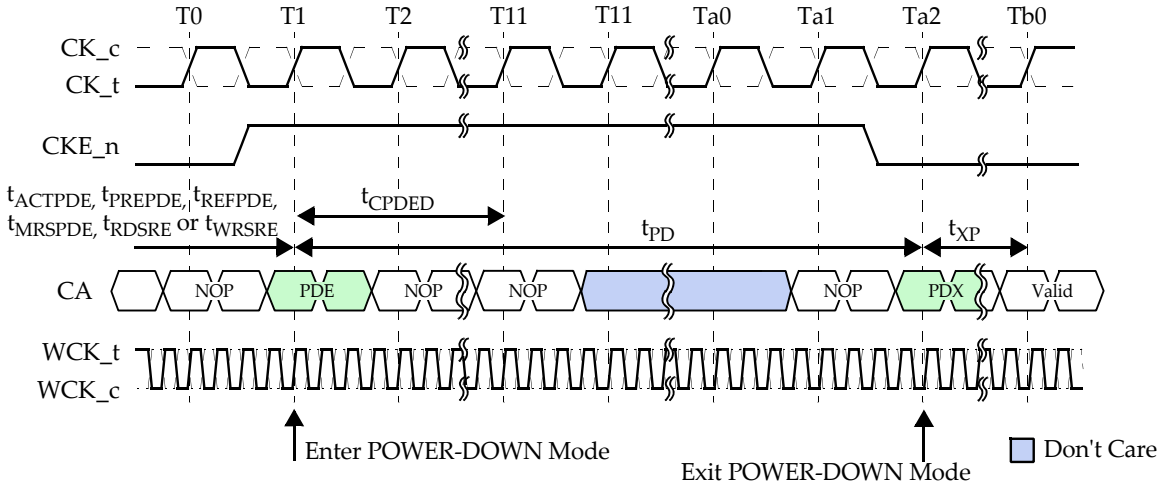
The power-down state is synchronously exited when CKE_n is registered LOW (in conjunction with a NOP command). A valid executable command may be applied $t_{XP}$ cycles later. The power-down duration is specified by $t_{PD}$ and limited by the refresh requirements of the device.

## 7.22 POWER-DOWN (cont'd)

A) LP2 bit On (MR5 OP1 = 1)




B) LP2 bit Off (MR5 OP1 = 0)




NOTE 1 Minimum CKE_n pulse width must satisfy $t_{CKE}$.
NOTE 2 After issuing POWER-DOWN command, NOP commands are required for a minimum of $t_{CPDED}$ after POWER-DOWN entry.

**Figure 102 — Power-Down Entry and Exit**

**Table 45 — Pin States During Power-Down**

| Pin | LP2 on (MR5 OP1 = HIGH) | LP2 off (MR5 OP1 = LOW) |
|---|---|---|
| EDC | HIGH | 'Hold' |
| DQ/DBI_n | ODT | ODT |
| CA | ODT | ODT |
| CKE_n | ODT (Driven HIGH by Controller) | ODT (Driven HIGH by Controller) |
| CK_t/CK_c | ODT | ODT |
| WCK_t/WCK_c | ODT (receiver off) | ODT (receiver active) |

## 7.23 COMMAND TRUTH TABLES

### Table 46 — Truth Table – CKE_n

| CKE_n[2] | | CURRENT STATE [3, 5] | COMMAND(*n*)[4] | ACTION(*n*)[4] | NOTES |
|---|---|---|---|---|---|
| Previous Cycle *(n-1)* | Current Cycle *(n)* | | | | |
| H | H | Power-Down | X | Maintain Power-Down | |
| H | H | Self Refresh | X | Maintain Self Refresh | |
| H | L | Power-Down | NOP | Exit Power-Down | |
| H | L | Self Refresh | NOP | Exit Self Refresh | 6 |
| L | H | All Banks Idle | NOP | Precharge Power-Down Entry | |
| L | H | Bank(s) Active | NOP | Active Power-Down Entry | |
| L | H | All Banks Idle | REFab | Self Refresh Entry | |
| L | L | See Table 47 and Table 48 | | | |

NOTE 1 H = Logic HIGH Level; L = Logic LOW Level; X = Don't Care (command decoder disabled)
NOTE 2 CKE_n(*n*) is the logic state of CKE_n at clock edge *n*; CKE_n(*n-1*) was the state of CKE_n at the previous clock edge.
NOTE 3 Current state is the state of the device immediately prior to clock edge n.
NOTE 4 COMMAND(n) is the command registered at clock edge n, and ACTION*n* is a result of COMMAND*n*.
NOTE 5 All states and sequences not shown are illegal or reserved.
NOTE 6 NOP commands should be issued on any clock edges occurring during the $t_{XS}$ period. A minimum of $t_{LK}$ is needed for the PLL/DLL to lock before applying a READ or WRITE command if the PLL/DLL was disabled.

## 7.23 COMMAND TRUTH TABLES (cont'd)

### Table 47 — Truth Table – Current State Bank *n* – Command To Bank *n*

| CURRENT STATE | CA9 (Rising edge of CK) | CA9 (Falling edge of CK) | CA8 (Rising edge of CK) | CA8 (Falling edge of CK) | COMMAND/ACTION | NOTES |
|---|---|---|---|---|---|---|
| Any | H | H | H | H | NO OPERATION (NOP/continue previous operation) | |
| | | | H | L | | |
| | | | L | H | | |
| Idle | L | V | V | V | ACTIVATE (select and activate row) | |
| | H | L | L | H | REFRESH ALL / PER-2-BANK REFRESH | 8 |
| | H | H | L | L | MODE REGISTER SET | 8 |
| Row Active | H | L | H | H | READ (select column and start READ burst) | 10 |
| | H | L | H | L | WRITE (select column and start WRITE burst) (WOM, WSM or WDM) | 10 |
| | H | L | L | L | PRECHARGE (deactivate row in bank or banks) | 9 |
| Read (Auto Precharge Disabled) | H | L | H | H | READ (select column and start new READ burst) | 10 |
| | H | L | H | L | WRITE (select column and start WRITE burst) (WOM, WSM or WDM) | 10, 11 |
| | H | L | L | L | PRECHARGE (only after the READ burst is complete | 9 |
| Write (Auto Precharge Disabled) (WOM, WSM or WDM) | H | L | H | H | READ (select column and start READ burst) | 10 |
| | H | L | H | L | WRITE (select column and start new WRITE burst) (WOM, WSM or WDM) | 10 |
| | H | L | L | L | PRECHARGE (only after the WRITE burst is complete) | 9 |

NOTE 1 H = Logic HIGH Level; L = Logic LOW Level; V = Valid signal (H or L, but not floating)
NOTE 2 This table applies when CKE_n(*n-1*) was LOW and CKE_n(*n*) is LOW (see Table 46) and after $t_{XS}$ has been met (if the previous state was self refresh).
NOTE 3 This table is bank-specific, except where noted (i.e., the current state is for a specific bank and the commands shown are those allowed to be issued to that bank when in that state). Exceptions are covered in the notes below.
NOTE 4 Current state definitions:
Idle: The bank has been precharged, and $t_{RP}$ has been met.
Row Active: A row in the bank has been activated, and $t_{RCD}$ has been met. No data bursts/accesses and no register accesses are in progress.
Read: A READ burst has been initiated, with auto precharge disabled.
Write: A WRITE burst has been initiated, with auto precharge disabled.
NOTE 5 The following states must not be interrupted by a command issued to the same bank. NOP commands, or allowable commands to the other bank should be issued on any clock edge occurring during these states. Allowable commands to the other bank are determined by its current state and Table 47, and according to Table 48. The minimum delay from a READ or WRITE command with auto precharge enabled, to a command to the same bank is summarized in Table 49.
Precharging: Starts with registration of a PRECHARGE command and ends when $t_{RP}$ is met. Once $t_{RP}$ is met, the bank will be in the idle state.
Row Activating: Starts with registration of an ACTIVATE command and ends when $t_{RCD}$ is met. Once $t_{RCD}$ is met, the bank will be in the "row active" state.
Read w/Auto-Precharge Enabled: Starts with registration of a READ command with auto precharge enabled and ends when $t_{RP}$ has been met. Once $t_{RP}$ is met, the bank will be in the idle state.
Write w/Auto-Precharge Enabled: Starts with registration of a WRITE command with auto precharge enabled and ends when $t_{RP}$ has been met. Once $t_{RP}$ is met, the bank will be in the idle state.
NOTE 6 The following states must not be interrupted by any executable command; NOP commands must be applied on each positive clock edge during these states.
Refreshing: Starts with registration of a REFab command and ends when $t_{RFCab}$ is met. Once $t_{RFCab}$ is met, the device will be in the all banks idle state.
Accessing Mode Register: Starts with registration of a MODE REGISTER SET command and ends when $t_{MOD}$ has been met. Once $t_{MOD}$ is met, the device will be in the all banks idle state.
Precharging All: Starts with registration of a PREall command and ends when $t_{RP}$ is met. Once $t_{RP}$ is met, all banks will be in the idle state.
READ or WRITE: Starts with the registration of the ACTIVATE command and ends with the last valid data nibble.
NOTE 7 All states and sequences not shown are illegal or reserved.
NOTE 8 Not bank-specific (REFab, MODE REGISTER SET) or bank specific (REFpb / REFp2b); requires that all banks (REFab, MODE REGISTER SET) or the current bank (REFpb / REFp2b) are idle, and bursts are not in progress.
NOTE 9 May or may not be bank-specific; if multiple banks are to be precharged, each must be in a valid state for precharging.
NOTE 10 Reads or Writes listed in the Command/Action column include Reads or Writes with auto precharge enabled and Reads or Writes with auto precharge disabled.
NOTE 11 A WRITE command may be applied after the completion of the READ burst

## 7.23 COMMAND TRUTH TABLES (cont'd)

### Table 48 — Truth Table – Current State Bank *n* – Command To Bank *m*

| CURRENT STATE | CA9 (Rising edge of CK) | CA9 (Falling edge of CK) | CA8 (Rising edge of CK) | CA8 (Falling edge of CK) | COMMAND/ACTION | NOTES |
|---|---|---|---|---|---|---|
| Any | H | H | H | H | NO OPERATION (NOP/continue previous operation) | |
| | | | H | L | | |
| | | | L | H | | |
| Idle | X | X | X | X | Any Command Otherwise Allowed to Bank m | |
| Row Activating, Active, or Precharging | L | V | V | V | ACTIVATE (select and activate row) | |
| | H | L | H | H | READ (select column and start READ burst) | 8 |
| | H | L | H | L | WRITE (select column and start WRITE burst) (WOM, WSM or WDM) | 8 |
| | H | L | L | L | PRECHARGE | |
| Read (Auto Precharge Disabled) | L | V | V | V | ACTIVATE (select and activate row) | |
| | H | L | H | H | READ (select column and start new READ burst) | 8 |
| | H | L | H | L | WRITE (select column and start WRITE burst) (WOM, WSM or WDM) | 8, 9 |
| | H | L | L | L | PRECHARGE | |
| Write (Auto Precharge Disabled) | L | V | V | V | ACTIVATE (select and activate row) | |
| | H | L | H | H | READ (select column and start READ burst) | 8 |
| | H | L | H | L | WRITE (select column and start new WRITE burst) (WOM, WSM or WDM) | 8 |
| | H | L | L | L | PRECHARGE | |
| Read (With Auto Precharge) | L | V | V | V | ACTIVATE (select and activate row) | |
| | H | L | H | H | READ (select column and start new READ burst) | 8 |
| | H | L | H | L | WRITE (select column and start WRITE burst) (WOM, WSM or WDM) | 8, 9 |
| | H | L | L | L | PRECHARGE | |
| Write (With Auto Precharge) | L | V | V | V | ACTIVATE (select and activate row) | |
| | H | L | H | H | READ (select column and start READ burst) | 8 |
| | H | L | H | L | WRITE (select column and start new WRITE burst) (WOM, WSM or WDM) | 8 |
| | H | L | L | L | PRECHARGE | |

NOTE 1 H = Logic HIGH Level; L = Logic LOW Level; V = Valid signal (H or L, but not floating)

NOTE 2 This table applies when CKE_n(*n-1*) was LOW and CKE_n(*n*) is LOW (see Table 46) and after $t_{XS}$ has been met (if the previous state was self refresh).

NOTE 3 WRITE in this table refers to WOM/WOMA, WSM/WSMA and WDM/WDMA commands

NOTE 4 This table describes alternate bank operation, except where noted (i.e., the current state is for bank n and the commands shown are those allowed to be issued to bank m, assuming that bank *m* is in such a state that the given command is allowable). Exceptions are covered in the notes below.

NOTE 5 Current state definitions:

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

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

Read: A READ burst has been initiated, with auto precharge disabled.

Write: A WRITE burst has been initiated, with auto precharge disabled.

Read with Auto Precharge Enabled: See following text

Write with Auto Precharge Enabled: See following text

NOTE 5a The read with auto precharge enabled or write with auto precharge enabled states can each be broken into two parts: the access period and the precharge period. For read with auto precharge, the precharge period is defined as if the same burst was executed with auto precharge disabled and then followed with the earliest possible PRECHARGE command that still accesses all of the data in the burst. For write with auto precharge, the precharge period begins when $t_{WR}$ ends, with $t_{WR}$ measured as if auto precharge was disabled. The access period starts with registration of the command and ends where the precharge period (or $t_{RP}$) begins. During the precharge period of the read with auto precharge enabled or write with auto precharge enabled states, ACTIVATE, PRECHARGE, READ and WRITE commands to the other bank may be applied. In either case, all other related limitations apply (e.g., contention between read data and write data must be avoided).

NOTE 5b The minimum delay from a READ or WRITE command with auto precharge enabled, to a command to a different bank is summarized in Table 49.

NOTE 6 REFab and MODE REGISTER SET commands may only be issued when all banks are idle.

NOTE 7 All states and sequences not shown are illegal or reserved.

NOTE 8 READs or WRITEs listed in the Command/Action column include READs or WRITEs with auto precharge enabled and READs or WRITEs with auto precharge disabled.

NOTE 9 A WRITE command may be applied after the completion of the READ burst.

## 7.23 COMMAND TRUTH TABLES (cont'd)

### Table 49 — Auto Precharge Enabled/Disabled and PREab / PREpb Timings

| FROM COMMAND | TO COMMAND | Minimum Delay Between "FROM COMMAND" to "TO COMMAND" [9] | UNIT | NOTES |
|---|---|---|---|---|
| READ | PREpb (same bank) | $t_{RTP}$ | $t_{CK}$ | 5 |
| | PREpb (different bank) | 1 | $t_{CK}$ | |
| | PREab | $t_{RTP}$ | $t_{CK}$ | 5 |
| | WRITE or WRITE w/ AP (any bank) | $t_{RTW}$ | ns | |
| | READ or READ w/ AP (any bank) | $t_{CCD}$ | $t_{CK}$ | 4 |
| READ w/ AP | PREab | $t_{RTP}$ | $t_{CK}$ | 5 |
| | PREpb (different bank) | 1 | $t_{CK}$ | |
| | ACTIVATE or REFpb / REFp2b (same bank) | $t_{RTP} + RU_{(t_{RP}/t_{CK})}$ | $t_{CK}$ | 5, 6 |
| | WRITE or WRITE w/ AP (same bank) | Illegal | | |
| | WRITE or WRITE w/ AP (different bank) | $t_{RTW}$ | ns | |
| | READ or READ w/ AP (same bank) | Illegal | | |
| | READ or READ w/ AP (different bank) | $t_{CCD}$ | $t_{CK}$ | 4 |
| WRITE (WOM, WDM, OR WSM) | PREpb (same bank) | $WLmrs + BL/8 + RU_{(t_{WR}/t_{CK})}$ | $t_{CK}$ | 1, 2, 6 |
| | PREpb (different bank) | WOM = 1, WDM = 2, WSM = 3 | $t_{CK}$ | |
| | PREab | $WLmrs + BL/8 + RU_{(t_{WR}/t_{CK})}$ | $t_{CK}$ | 1, 2, 6 |
| | WRITE or WRITE w/ AP (same bank, w/o Data Mask) | WOM/WDM = $t_{CCD}$<br>WSM = MAX [$t_{CCD}$, 3 * $t_{CK}$] | tCK | 4, 8 |
| | WRITE or WRITE w/ AP (same bank, w Data Mask) | WOM/WDM = MAX [$t_{CCD}$, $t_{CCDMW}$]<br>WSM = MAX [$t_{CCD}$, $t_{CCDWM}$, 3 * $t_{CK}$] | tCK | 4, 8 |
| | WRITE or WRITE w/ AP (different bank) | WOM/WDM = tCCD<br>WSM = MAX [tCCD, 3 * tCK] | tCK | 4, 8 |
| | READ w/ AP (same bank) | $WLmrs + BL/8 + MAX [RU_{(t_{WR}/t_{CK})} - t_{RTP}, RU_{(t_{WTR}/t_{CK})}]$ | $t_{CK}$ | 1, 2, 3, 5, 6 |
| | READ (same bank) | $WLmrs + BL/8 + RU_{(t_{WTR}/t_{CK})}$ | $t_{CK}$ | 1, 2, 3, 6 |
| | READ or READ w/ AP (different bank) | $WLmrs + BL/8 + RU_{(t_{WTR}/t_{CK})}$ | $t_{CK}$ | 1, 2, 3, 6 |
| WRITE w/ AP (WOMA, WDMA, WSMA) | PREab | $WLmrs + BL/8 + RU_{(t_{WR}/t_{CK})}$ | $t_{CK}$ | 1, 2, 6 |
| | PREpb (different bank) | WOM = 1, WDM = 2, WSM = 3 | $t_{CK}$ | |
| | ACTIVATE or REFpb / REFp2b (same bank) | $WLmrs + BL/8 + WR + RU_{(t_{RP}/t_{CK})}$ | $t_{CK}$ | 1, 2, 6 |
| | ACTIVATE or REFpb / REFp2b (different bank) | WOM = 1, WDM = 2, WSM = 3 | $t_{CK}$ | |
| | WRITE or WRITE w/ AP (same bank) | Illegal | | |
| | WRITE or WRITE w/ AP (different bank) | WOM/WDM = $t_{CCD}$<br>WSM = MAX [$t_{CCD}$, 3 * $t_{CK}$] | tCK | 4, 8 |
| | READ or READ w/ AP (same bank) | Illegal | | |
| | READ or READ w/ AP (different bank) | $WLmrs + BL/8 + RU_{(t_{WTR}/t_{CK})}$ | $t_{CK}$ | 1, 2, 3, 6 |
| PREpb | PREpb (any bank) | $t_{PPD}$ | $t_{CK}$ | |
| | PREab | $t_{PPD}$ | $t_{CK}$ | 7 |
| PREab | PREpb (any bank) or PREab | $t_{PPD}$ | $t_{CK}$ | 7 |

NOTES:
1. BL = Burst length
2. WLmrs = Write latency
3. $t_{WTR} = t_{WTRS}$ when Bank Groups disabled or WRITE and READ access different bank groups, otherwise $t_{WTR} = t_{WTRL}$
4. $t_{CCD} = t_{CCDS}$ when bank groups is disabled or the second WRITE or READ is to a different bank group, otherwise $t_{CCD} = t_{CCDL}$; $t_{CCD}$ and $t_{CCDMW}$ are effective for both PC mode and 2-channel operation.
5. $t_{RTP} = t_{RTPL}$ when Bank Groups enabled and the PRECHARGE command accesses the same bank, otherwise $t_{RTP} = t_{RTPS}$
6. RU = round up to next integer
7. PREab is treated as a NOP for a bank when that bank is already idle or in the process of being precharged.
8. WOM / WDM / WSM in the "Minimum Delay between..." column refer to the "From Command".
9. A command issued during the minimum delay time is illegal.

## 7.24 CLOCK FREQUENCY CHANGE SEQUENCE

Step 1) Wait until all commands have finished, all banks are idle.

Step 2) If the new desired clock frequency is below the min frequency supported by PLL/DLL-on mode, turn the PLL/DLL off via an MRS command. Similarly, if the new desired clock frequency is above the maximum frequency supported by Low Frequency mode ($f_{CKLF}$(max)), turn Low Frequency mode off via an MRS command. If the new desired clock frequency requires termination, enable the termination before switching to the new frequency if it is not already enabled.

Step 3) Issue SELF REFRESH ENTRY command on both channels followed by NOP until $t_{CPDED}$ is met.

Step 4) Change the clock frequency and wait until clock is stabilized.

Step 5) Exit Self Refresh and wait for $t_{XS}$.

Step 6) Perform CA training if required.

Step 7) If the new clock frequency is within the PLL/DLL on range and the PLL/DLL on state is desired, enable the PLL/DLL via an MRS Command if it is not already enabled. Similarly, if the new clock frequency is within the Low Frequency mode range and the use of Low Frequency mode is desired, enable Low Frequency mode via an MRS command if it is not already enabled. If the new clock frequency does not require termination, it is now safe to disable termination.

Step 8) Perform WCK2CK training. As defined in the WCK2CK training process, if the PLL/DLL is enabled, then complete steps 8a and 8b:

8a) Reset the PLL/DLL by writing to the MRS register.

8b) Wait $t_{LK}$ clock cycles before issuing any commands to the device.

Step 9) Exit WCK2CK training.

Step 10) Perform READ and WRITE training, if required.

Step 11) Device is ready for normal operation after any necessary interface training.

NOTE: Either in Step 2 or Step 7 other Mode Register settings, for example, RLmrs, WLmrs, LP2, LP3 or RDQS mode, etc. that need to change, should be set for the new frequency before any training and the device is ready for normal operation. Otherwise proper operation cannot be guaranteed.

## 7.25 DYNAMIC VOLTAGE SWITCHING (DVS)

GDDR6 SGRAMs supporting multiple supply voltages allow the supply voltage to be changed during the course of normal operation using the Dynamic Voltage Switching (DVS) feature. By using DVS the device's power consumption can be reduced whenever only a fraction of the maximum available bandwidth is required by the current work load.

DVS requires the device to be properly placed into self refresh before the voltage is changed from the existing stable voltage, $V_{original}$ to the new desired voltage $V_{new}$. The DVS procedure may also require changes to the VDD Range mode register using MR7 bits OP[9:8], depending on whether the feature is supported. The DRAM vendor's datasheet shall be consulted regarding the supported supply voltages for DVS, and any dependencies of AC timing parameters on the selected supply voltage.

Clock frequency changes can also take place after entering self refresh mode using the standard Clock Frequency Change procedure. A clock frequency change in conjunction with DVS is required if $t_{CK}$ is less than $t_{CK}$min supported by $V_{new}$. In this case normal device operation including self refresh exit is not guaranteed without a frequency change. Changing the frequency while in self refresh is the most safe procedure.

Once self refresh is entered, $t_{CKSRE}$ must be met before the supply voltage is allowed to transition from $V_{original}$ to $V_{new}$. After $V_{DD}$ and $V_{DDQ}$ are stable at $V_{new}$, $t_{VS}$ must be met to allow for internal voltages in the device to stabilize before self refresh mode may be exited. During the voltage transition the voltage must not go below $V_{min}$ of the lower voltage of either $V_{original}$ or $V_{new}$ in order to prevent false chip reset. $V_{min}$ is the minimum voltage allowed by $V_{DD}$ or $V_{DDQ}$ in the DC operating conditions table. $V_{REF}$ shall continue to track $V_{DDQ}$.

### DVS Procedure

Step 1) Complete all operations and precharge all banks.

Step 2) Issue MRS commands on both channels to set VDD Range to proper values for $V_{new}$. This step is only required when the VDD Range mode register field is supported by the device. The DRAM vendor's datasheet should be consulted to verify if the feature is supported.

Step 3) Enter self refresh mode on both channels. Self refresh entry procedure must be met.

Step 4) Wait required time $t_{CKSRE}$ before changing voltage to $V_{new}$.

Step 5) Change $V_{DD}$ and $V_{DDQ}$ to $V_{new}$.

Step 6) Wait required time $t_{VS}$ for voltage stabilization.

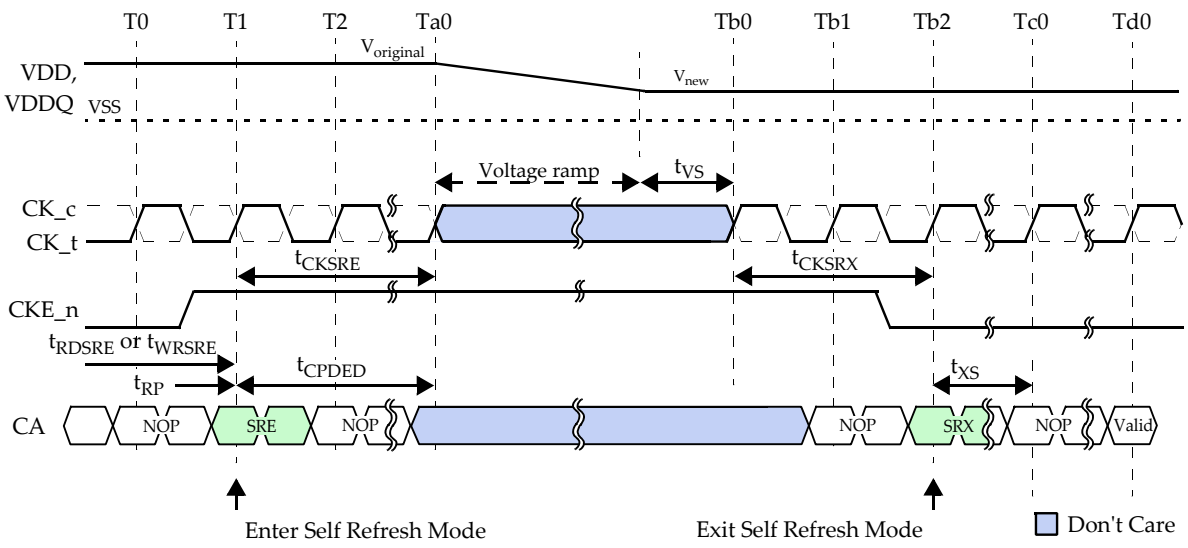
Step 7) Exit self refresh. The self refresh exit procedure must be met.

Step 8) Issue MRS commands to adjust mode register settings as desired (e.g., latencies, PLL/DLL on/off, CRC on/off).

Step 9) Perform any interface training as required.

Step 10) Continue normal operation.

## 7.25 DYNAMIC VOLTAGE SWITCHING (DVS) (cont'd)




Self refresh exit requires WCK2CK training prior to any WRITE or READ operation
At least one REFRESH command shall be issued after $t_{XS}$ for output driver and termination impedance updates
$V_{original} > V_{new}$ shown as an example of a voltage change

**Figure 103 — DVS Sequence**

## 7.26 TEMPERATURE SENSOR

GDDR6 SGRAMs incorporate a temperature sensor with digital temperature readout function. This function allows the controller to monitor the die's junction temperature and use this information to make sure the device is operated within the specified temperature range or to adjust interface timings relative to temperature changes over time.

The temperature sensor is permanently enabled. The temperature readout uses the DRAM Info mode feature. The digital value is driven asynchronously on the DQ bus following the MRS command to Mode Register 3 (MR3) that sets bit OP7 to 1 and bit OP6 to 0. The temperature readout will be continuously driven until an MRS command sets both bits to 0.

The device's junction temperature is linearly encoded as shown in Table 50 and Figure 101. DRAM vendors may restrict the readout to a subset of at least eight digital codes out of Table 50, corresponding to eight temperature thresholds. The sensor's accuracy is vendor specific.

**Table 50 — Temperature Sensor Readout Pattern**

| Temperature [°C] | Decimal Temperature Readout | Binary Temperature Readout DQ[7:0] |
|---|---|---|
| -40 | 0 | 0000 0000 |
| ... | ... | ... |
| 0 | 20 | 0001 0100 |
| 2 | 21 | 0001 0101 |
| 4 | 22 | 0001 0110 |
| 6 | 23 | 0001 0111 |
| 8 | 24 | 0001 1000 |
| ... | ... | ... |
| 120 | 80 | 0101 0000 |
| >120 | 80 | 0101 0000 |

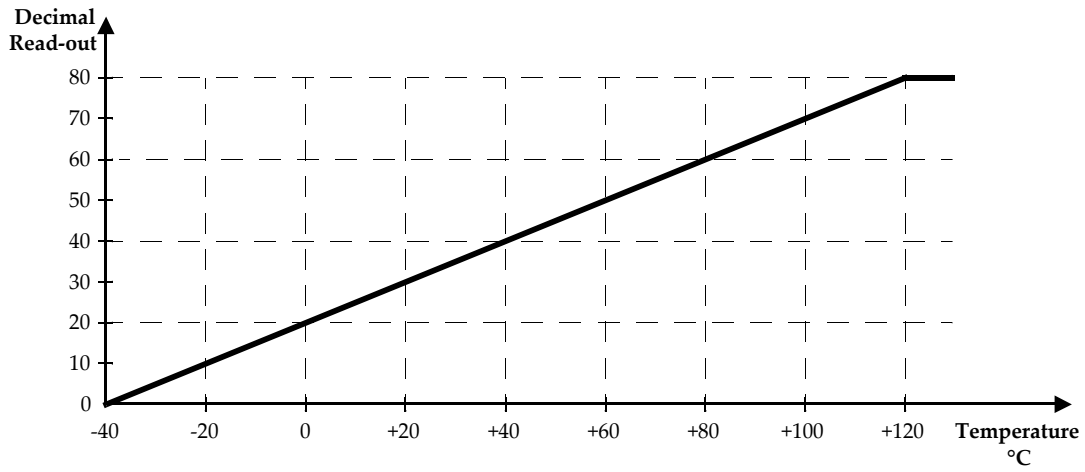



**Figure 104 — Temperature Sensor Readout Characteristic**

## 7.27 DUTY CYCLE CORRECTOR (DCC)

As GDDR6 SGRAM can operate with the PLL/DLL off during normal operation, the use of a Duty Cycle Corrector (DCC) can correct for the duty cycle error of the WCK clocks, resulting in improved timing margins for Reads and Writes. The DCC can be enabled at any time; however it is recommended to enable the DCC prior to WCK2CK training because any shift of rising and falling WCK edges can impact the WCK2CK training results. A new duty cycle correction cycle is required upon a frequency change.

DCC operation is controlled by MR7 OP[11:10]. The duty cycle correction is started by setting bit OP10. The DCC must be held in this state for a minimum duration of $t_{DCC}$. After $t_{DCC}$ is met, bit OP11 shall be set to terminate the duty cycle correction and hold the correction code. The DCC may be disabled at any time by resetting bits OP11 and OP10.

When in the "hold" state, the DCC will remember its correction code during power-down (with LP2 enabled) and self refresh. In this case the WCK2CK training time upon power-down exit or self refresh exit is not gated by $t_{DCC}$, as the correction code will be automatically restored. Alternatively, a new duty cycle correction cycle could be started by resetting bit OP11.

A new duty cycle correction cycle is required upon a frequency change.

**Table 51 — DCC Timings**

| Parameter | Symbol | Min | Max | Unit |
|---|---|---|---|---|
| Required time for duty cycle corrector | $t_{DCC}$ | 2048 | - | $t_{CK}$ (or ns) |

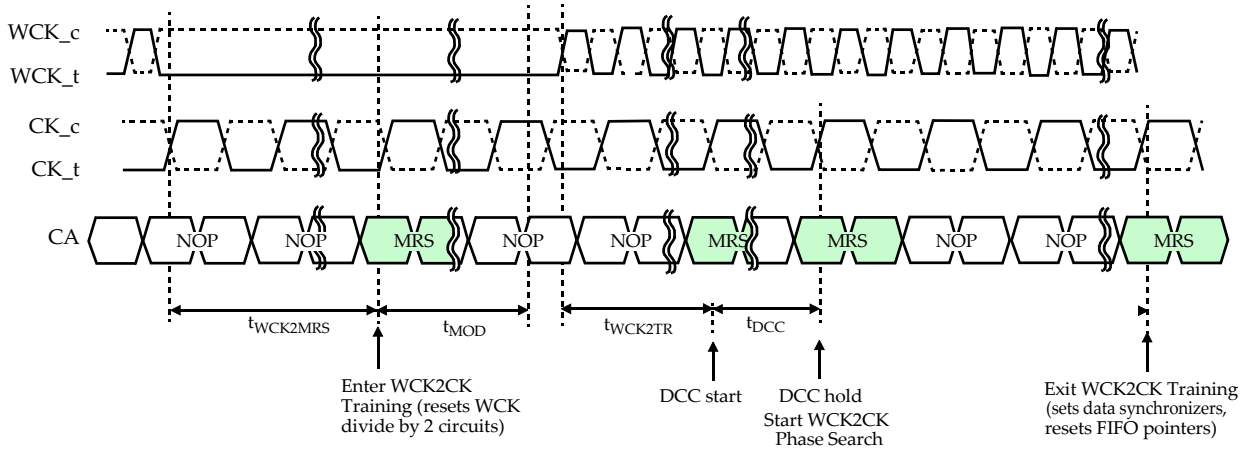



**Figure 105 — WK2CK Training Sequence with DCC**

**Table 52 — DCC Control Signals**

| OP11 | OP10 | DCC |
|---|---|---|
| 0 | 0 | DCC off |
| 0 | 1 | DCC start |
| 1 | 0 | Reserved |
| 1 | 1 | DCC hold |

## 7.28 DUTY CYCLE adjuster (DCA)

As GDDR6 SDRAM can operate with the PLL/DLL off during normal operation, the use of a Duty Cycle Corrector (DCA) can correct for the duty cycle error of the WCK clocks, resulting in improved timing margins for Reads and Writes. The DCA can be enabled at any time before or after WCK2CK training; however it must disable the DCC (MR7 OP[11:10]=00) prior to DCA because DCA code controlled by DCC. A new duty cycle adjuster is required upon a frequency change.

DCA operation is controlled by MR14 OP[11:4]. OP[11:8] controls the offset of channel A byte 0 and Channel B byte 1. OP[7:4] controls the offset of Channel A Byte 1 and Channel B byte 0.

The DCA code update delay time, tWCK2MRS+tWCK2TR, is required to complete the phase adjuster and is the minimum time required between two DCA code changes.

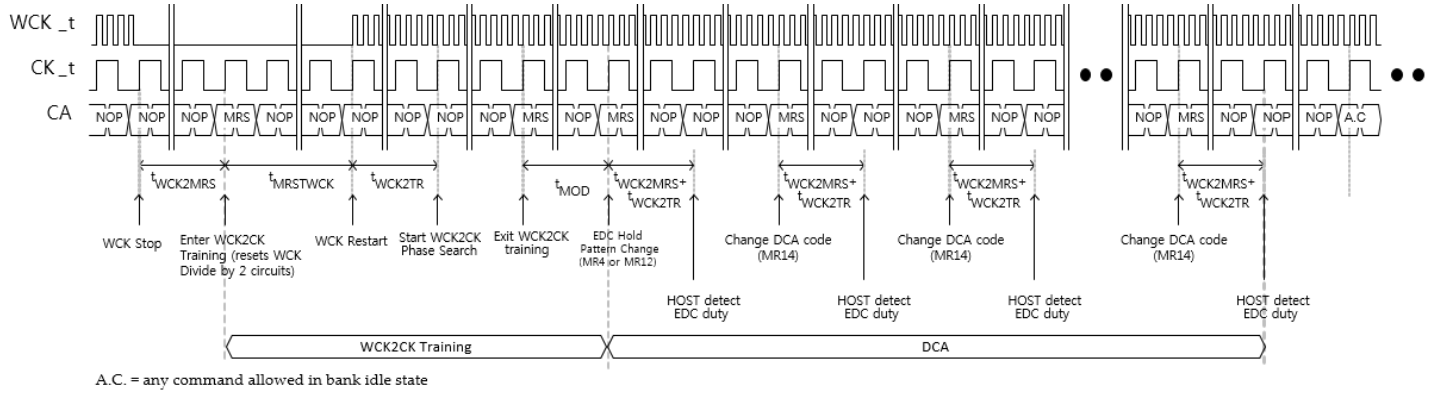



**Figure 106 — DCA after WCK2CK Training**

# 8. OPERATING CONDITIONS

## 8.1 ABSOLUTE MAXIMUM RATINGS

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

**Table 53 — Absolute Maximum Ratings**

| PARAMETER | SYMBOL | MIN | MAX | UNIT | NOTES |
|---|---|---|---|---|---|
| Voltage on $V_{DD}$ relative to $V_{SS}$ | $V_{DD}$ | -0.3 | +1.62 | V | 1 |
| Voltage on $V_{DDQ}$ relative to $V_{SS}$ | $V_{DDQ}$ | -0.3 | +1.62 | V | 1 |
| Voltage on $V_{PP}$ relative to $V_{SS}$ | $V_{PP}$ | -0.3 | +2.3 | V | 2 |
| Voltage on any signal relative to $V_{SS}$ | $V_{IN}$, $V_{OUT}$ | -0.3 | +1.62 | V | |
| Storage temperature | $T_{STG}$ | -55 | +125 | °C | |

NOTE 1 $V_{DD}$ and $V_{DDQ}$ must be within 300 mV of each other at all times the device is powered-up.
NOTE 2 $V_{PP}$ must be equal or greater than $V_{DD}$ and $V_{DDQ}$ at all times the device is powered-up.

## 8.2 PAD CAPACITANCES

**Table 54 — Silicon Pad Capacitance**

| PARAMETER [1] | SYMBOL | MIN | MAX | UNITS | NOTES |
|---|---|---|---|---|---|
| Delta Input/Output Capacitance: DQs, DBI_n, EDC | DCio | | | pF | 2 |
| Delta Input Capacitance: CA, CKE_n, CABI_n | DCi1 | | | pF | 3 |
| Delta Input Capacitance: CK_t, CK_c | DCi2 | | | pF | 4 |
| Delta Input Capacitance: WCK_t, WCK_c | DCi3 | | | pF | 5 |
| Input/Output Capacitance: DQs, DBI_n, EDC | Cio | | | pF | |
| Input Capacitance: CA, CKE_n, CABI_n | Ci1 | | | pF | |
| Input Capacitance: CK_t, CK_c | Ci2 | | | pF | |
| Input Capacitance: WCK_t, WCK_c | Ci3 | | | pF | |

NOTE 1 This parameter is not subject to production test. It is verified by design and characterization. The silicon pad capacitance is validated by deembedding the package L & C parasitics. The capacitance is measured with $V_{DD}$, $V_{DDQ}$ and $V_{SS}$, applied with all other signals floating.
NOTE 2 DCio = Cio (Max) – Cio (Min)
NOTE 3 DCi1 = Ci1 (Max) – Ci1 (Min)
NOTE 4 DCi2 = Absolute value of C CK_t – C CK_c
NOTE 5 DCi3 = Absolute value of C WCK_t – C WCK_c

## 8.3 PACKAGE ELECTRICAL SPECIFICATION

### Table 55 — GDDR6 SGRAM Package Electrical Specifications

| PARAMETER [1, 2, 3] | SYMBOL | MIN | MAX | UNIT | NOTES |
|---|---|---|---|---|---|
| Input/Output package impedance: DQs, DBI_n, EDC | $Z_{IO}$ | | | Ω | |
| Input/Output package delay: DQs, DBI_n, EDC | $Td_{IO}$ | | | ps | |
| Input package impedance: CA, CABI_n, CKE_n | $Z_{I1}$ | | | Ω | |
| Input package delay: CA, CABI_n, CKE_n | $Td_{I1}$ | | | ps | |
| Input package impedance: CK_t, CK_c | $Z_{I2}$ | | | Ω | |
| Input package delay: CK_t, CK_c | $Td_{I2}$ | | | ps | |
| Delta input package impedance: CK_t, CK_c | $DZ_{I2}$ | – | | Ω | 4 |
| Delta input package delay: CK_t, CK_c | $DTd_{I2}$ | – | | ps | 5 |
| Input package impedance: WCK_t, WCK_c | $Z_{I3}$ | | | Ω | |
| Input package delay: WCK_t, WCK_c | $Td_{I3}$ | | | ps | |
| Delta input package impedance: WCK_t, WCK_c | $DZ_{I3}$ | – | | Ω | 6 |
| Delta input package delay: WCK_t, WCK_c | $DTd_{I3}$ | – | | ps | 7 |

NOTE 1 This parameter is not subject to production test. It is verified by design and characterization. The package L & C parasitics are validated using package only samples. The capacitance is measured with $V_{DD}$, $V_{DDQ}$ and $V_{SS}$ shorted with all other signals floating. The inductance is measured with $V_{DD}$, $V_{DDQ}$ and $V_{SS}$ shorted and all other signal balls shorted at the die side (not ball).

NOTE 2 Package only impedance ($Z_{pkg}$) is calculated based on the $L_{pkg}$ and $C_{pkg}$ total for a given pin where

$$Z_{pkg}(\text{total per pin}) = \sqrt{L_{pkg}/C_{pkg}}$$

NOTE 3 Package only delay ($Td_{pkg}$) is calculated based on $L_{pkg}$ and $C_{pkg}$ total for a given pin where

$$Td_{pkg}(\text{total per pin}) = \sqrt{L_{pkg}*C_{pkg}}$$

NOTE 4 $DZ_{I2}$ = Absolute value of $Z_{I2}$ CK_t – $Z_{I2}$ CK_c

NOTE 5 $DTd_{I2}$ = Absolute value of $Td_{I2}$ CK_t – $Td_{I2}$ CK_c

NOTE 6 $DZ_{I3}$ = Absolute value of $Z_{I3}$ WCK_t – $Z_{I3}$ WCK_c

NOTE 7 $DTd_{I3}$ = Absolute value of $Td_{I3}$ WCK_t – $Td_{I3}$ WCK_c

## 8.4 PACKAGE THERMAL CHARACTERISTICS

**Table 56 — Thermal Characteristics**

| Parameter | Description | Value [7] | Units | Notes |
|---|---|---|---|---|
| Theta_JA | Thermal resistance junction to ambient | 25.2 | °C/W | 1,2,3,5 |
| TJ_MAX | Maximum operating junction temperature | 119.48 | °C | 4 |
| TC_MAX | Maximum operating case temperature | 85 | °C | 4 |
| Theta_JC | Thermal resistance from junction to case | 10.2 | °C/W | 1,6 |
| Theta_JB | Thermal resistance junction to board | 8.7 | °C/W | 1,2,6 |

NOTE 1 Measurement procedures for each parameter must follow standard procedures defined in the current JEDEC JESD-51 standard.
NOTE 2 Theta_JA and Theta_JB must be measured with the high effective thermal conductivity test board defined in JESD51-7
NOTE 3 Airflow information must be documented for Theta_JA.
NOTE 4 TJ_MAX and TC_MAX are documented for normal operation in this table. These are not intended to reflect reliability limits.
NOTE 5 Theta_JA should only be used for comparing the thermal performance of single packages and not for system related junction temperature prediction.
NOTE 6 Theta_JB and Theta_JC are derived through a package thermal simulation.
NOTE 7 Values are guaranteed by design but not tested in production

### Notes : DRAM Component Operating Temperature Range

| Symbol | Parameter | Rating | Units | Notes |
|---|---|---|---|---|
| $T_{OPER}$ | Normal Operating Temperature Range | 0 to 85 | °C | 1,2 |

**Notes:**

1. Operating Temperature TOPER is the case surface temperature on the center / top side of the DRAM. For measurement conditions, please refer to the JEDEC document JESD51-2.
2. The Normal Temperature Range specifies the temperatures where all DRAM specifications will be supported. During operation, the DRAM case temperature must be maintained between 0 - 85°C under all operating conditions.

## 8.5 ELECTROSTATIC DISCHARGE SENSITIVITY CHARACTERISTICS

**Table 57 — Electrostatic Discharge Sensitivity Characteristics**

| PARAMETER [1, 2] | SYMBOL | MIN | MAX | UNIT | NOTES |
|---|---|---|---|---|---|
| Human body model (HBM) | $ESD_{HBM}$ | 1000 | – | V | 3 |
| Charged-device model (CDM) | $ESD_{CDM}$ | 250 | – | V | 4 |

NOTE 1 Electrostatic discharge minimum requirements are defined for the high-speed interface balls of the device only. They do not apply to ZQ, RESET_n, TCK, TMS, TDI and TDO.
NOTE 2 State-of-the-art basic ESD control measures have to be in place when handling devices.
NOTE 3 Refer to ESDA / JEDEC Joint Standard JS-001-2014 for measurement procedures.
NOTE 4 Refer to JS-002-2014 for measurement procedures

## 8.6 AC & DC OPERATING CONDITIONS

Initially, all GDDR6 SGRAMs are designed for 1.35V typical VDDQ voltage supplies. The interface of GDDR6 with 1.35V VDDQ termination will follow the POD135 Standard (JESD8-21), Class D. GDDR6 SGRAMs can also be designed for 1.25V typical VDDQ voltage supplies. In that case the interface of GDDR6 with 1.25V VDDQ termination will follow the POD125 Standard (JESD8-30), Class A. GDDR6 SGRAMs can also be designed to support multiple VDDQ supply voltages, e.g. 1.35V and 1.25V. In that case the interface of GDDR6 with follow the related POD standard.

Vendor datasheets should be consulted for actual VDD and VDDQ voltages supported by a GDDR6 device as factors such as process technology and supported system voltage(s) many require typical operating voltages to be added, dropped or maintained over time.

GDDR6 SGRAMs can also support the switching of the VDD and VDDQ supply voltages during self refresh. The Dynamic Voltage Switching (DVS) procedure should be followed when switching between supply voltages. Vendor datasheets should be consulted for details regarding DVS support.

**Table 58 — DC Operating Conditions**

| | | POD135 | | | POD125 | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Parameter | Symbol | Min | Typ | Max | Min | Typ | Max | Unit | Note |
| Device supply voltage | $V_{DD}$ | 1.3095 | 1.35 | 1.3905 | 1.2125 | 1.25 | 1.2875 | V | 1 |
| Output supply voltage | $V_{DDQ}$ | 1.3095 | 1.35 | 1.3905 | 1.2125 | 1.25 | 1.2875 | V | 1 |
| Pump voltage | $V_{PP}$ | 1.746 | 1.8 | 1.908 | 1.746 | 1.8 | 1.908 | V | 2 |
| Reference voltage: DQ and DBI_n pins | $V_{REFD}$ | 0.69 * $V_{DDQ}$ | | 0.71 * $V_{DDQ}$ | 0.69 * $V_{DDQ}$ | | 0.71 * $V_{DDQ}$ | V | 3, 4 |
| | $V_{REFD2}$ | 0.49 * $V_{DDQ}$ | | 0.51 * $V_{DDQ}$ | 0.49 * $V_{DDQ}$ | | 0.51 * $V_{DDQ}$ | V | 3, 4, 5 |
| Reference voltage: CA pins | $V_{REFC}$ | 0.69 * $V_{DDQ}$ | | 0.71 * $V_{DDQ}$ | 0.69 * $V_{DDQ}$ | | 0.71 * $V_{DDQ}$ | V | 6 |
| | $V_{REFC2}$ | 0.49 * $V_{DDQ}$ | | 0.51 * $V_{DDQ}$ | 0.49 * $V_{DDQ}$ | | 0.51 * $V_{DDQ}$ | V | 6, 7 |
| DC input logic HIGH voltage with $V_{REFC}$: CA | $V_{IHA}$(DC) | $V_{REFC}$ + 0.135 | | | $V_{REFC}$ + 0.125 | | | V | |
| DC input logic LOW voltage with $V_{REFC}$: CA | $V_{ILA}$(DC) | | | $V_{REFC}$ - 0.135 | | | $V_{REFC}$ - 0.125 | V | |
| DC input logic HIGH voltage with $V_{REFC2}$: CA | $V_{IHA2}$(DC) | $V_{REFC2}$ + 0.27 | | | $V_{REFC2}$ + 0.25 | | | V | |
| DC input logic LOW voltage with $V_{REFC2}$: CA | $V_{ILA2}$(DC) | | | $V_{REFC2}$ - 0.27 | | | $V_{REFC2}$ - 0.25 | V | |
| DC input logic HIGH voltage with $V_{REFD}$: DQ and DBI_n | $V_{IHD}$(DC) | $V_{REFD}$ + 0.09 | | | $V_{REFD}$ + 0.085 | | | V | |
| DC input logic LOW voltage with $V_{REFD}$: DQ and DBI_n | $V_{ILD}$(DC) | | | $V_{REFD}$ - 0.09 | | | $V_{REFD}$ - 0.085 | V | |
| DC input logic HIGH voltage with $V_{REFD2}$: DQ and DBI_n | $V_{IHD2}$(DC) | $V_{REFD2}$ + 0.27 | | | $V_{REFD2}$ + 0.25 | | | V | |
| DC input logic LOW voltage with $V_{REFD2}$: DQ and DBI_n | $V_{ILD2}$(DC) | | | $V_{REFD2}$ - 0.27 | | | $V_{REFD2}$ - 0.25 | V | |
| RESET_n and Boundary Scan input logic HIGH voltage; EDC and CA input logic high voltage for x16/x8 mode, PC vs. 2-channel mode, CK and CA ODT select at reset | $V_{IHR}$ | 0.8 * $V_{DDQ}$ | | | 0.8 * $V_{DDQ}$ | | | V | 8 |
| RESET_n and Boundary Scan input logic LOW voltage; EDC and CA input logic low voltage for x16/x8 mode, PC vs. 2-channel mode, CK and CA ODT select at reset | $V_{ILR}$ | | | 0.2 * $V_{DDQ}$ | | | 0.2 * $V_{DDQ}$ | V | 8 |
| Input leakage current (any input 0V <= $V_{IN}$ <= $V_{DDQ}$; all other signals not under test = 0V) | $I_L$ | | | | | | | µA | |
| Output Leakage Current (outputs are disabled; 0V <= $V_{OUT}$ <= $V_{DDQ}$) | $I_{OZ}$ | | | | | | | µA | |
| Output logic LOW voltage | $V_{OL}$(DC) | | | 0.56 | | | 0.52 | V | |
| Single ended clock input voltage level: CK_t, CK_c, WCK_t, WCK_c | $V_{IN}$ | -0.30 | | $V_{DDQ}$ + 0.30 | -0.30 | | $V_{DDQ}$ + 0.30 | | 13 |

## Table 58 — DC Operating Conditions

| | | POD135 | | | POD125 | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Parameter | Symbol | Min | Typ | Max | Min | Typ | Max | Unit | Note |
| Clock input mid-point voltage: CK_t, CK_c | $V_{MP}$(DC) | $V_{REFC}$ - 0.10 | | $V_{REFC}$ + 0.10 | $V_{REFC}$ - 0.10 | | $V_{REFC}$ + 0.10 | V | 9, 12 |
| Clock input differential voltage: CK_t, CK_c | $V_{IDCK}$(DC) | 0.198 | | | 0.18 | | | V | 10, 12 |
| Clock input differential voltage: WCK_t, WCK_c | $V_{IDWCK}$(DC) | 0.18 | | | 0.165 | | | V | 11, 14 |

NOTE 1 GDDR6 SGRAMs are designed to tolerate PCB designs with separate $V_{DD}$ and $V_{DDQ}$ power regulators.
NOTE 2 DC bandwidth is limited to 20 MHz.
NOTE 3 AC noise in the system is estimated at 50mV pk-pk for the purpose of DRAM design.
NOTE 4 The reference voltage source and control for DQ and DBI_n pins are determined by Half VREFD and VREFD Level mode register bits.
NOTE 5 Programmable VREFD levels are not supported with $V_{REFD2}$.
NOTE 6 The reference voltage source (external or internal) is determined at power-up; the reference voltage level is determined by Half VREFC and the VREFC Offset mode register bits.
NOTE 7 Programmable VREFC offsets are not supported with $V_{REFC2}$.
NOTE 8 $V_{IHR}$ and $V_{ILR}$ apply to boundary scan input pins TDI, TMS and TCK. $V_{IHR}$ and $V_{ILR}$ apply to EDC and CA inputs at reset when latching default device configurations. $V_{IHR}$ and $V_{ILR}$ also apply to CA, CABI_n, CKE_n, CK, DQ, DBI_n, EDC and WCK inputs when boundary scan mode is active and input data are latched in the Capture-DR TAP controller state.
NOTE 9 This provides a minimum of 0.845 V to a maximum of 1.045 V, and is normally 70% of $V_{DDQ}$. DRAM timings relative to CK_t cannot be guaranteed if these limits are exceeded.
NOTE 10 $V_{IDCK}$ is the magnitude of the difference between the input level in CK_t and the input level on CK_c. The input reference level for signals other than CK_t and CK_c is $V_{REFC}$.
NOTE 11 $V_{IDWCK}$ is the magnitude of the difference between the input level in WCK_t and the input level on WCK_c. The input reference level for signals other than WCK_t and WCK_c is either $V_{REFC}$, $V_{REFC2}$, $V_{REFD}$ or $V_{REFD2}$.
NOTE 12 The CK_t and CK_c input reference level (for timing referenced to CK_t and CK_c) is the point at which CK_t and CK_c cross. Please refer to the applicable timings in the AC Timings table.
NOTE 13 Use $V_{IHR}$ and $V_{ILR}$ when boundary scan mode is active and input data are latched in the Capture-DR TAP controller state.
NOTE 14 The WCK_t and WCK_c input reference level (for timing referenced to WCK_t and WCK_c) is the point at which WCK_t and WCK_c cross. Please refer to the applicable timings in the AC Timings table.

## 8.6 AC & DC OPERATING CONDITIONS (cont'd)

### Table 59 — AC Operating Conditions (For Design only[11])

| Parameter | Symbol | POD135 | | | POD125 | | | Unit | Note |
|---|---|---|---|---|---|---|---|---|---|
| | | Min | Typ | Max | Min | Typ | Max | | |
| AC input logic HIGH voltage with $V_{REFC}$: CA | $V_{IHA}(AC)$ | $V_{REFC}$ + 0.18 | | | $V_{REFC}$ + 0.165 | | | V | |
| AC input logic LOW voltage with $V_{REFC}$: CA | $V_{ILA}(AC)$ | | | $V_{REFC}$ - 0.18 | | | $V_{REFC}$ - 0.165 | V | |
| AC input logic HIGH voltage with $V_{REFC2}$: CA | $V_{IHA2}(AC)$ | $V_{REFC}$ + 0.36 | | | $V_{REFC}$ + 0.333 | | | V | |
| AC input logic LOW voltage with $V_{REFC2}$: CA | $V_{ILA2}(AC)$ | | | $V_{REFC}$ - 0.36 | | | $V_{REFC}$ - 0.333 | V | |
| AC input logic HIGH voltage with $V_{REFD}$: DQ and DBI_n | $V_{IHD}(AC)$ | $V_{REFD}$ + 0.135 | | | $V_{REFD}$ + 0.125 | | | V | |
| AC input logic LOW voltage with $V_{REFD}$: DQ and DBI_n | $V_{ILD}(AC)$ | | | $V_{REFD}$ - 0.135 | | | $V_{REFD}$ - 0.125 | V | |
| AC input logic HIGH voltage with $V_{REFD2}$: DQ and DBI_n | $V_{IHD2}(AC)$ | $V_{REFD2}$ + 0.36 | | | $V_{REFD2}$ + 0.333 | | | V | |
| AC input logic LOW voltage with $V_{REFD2}$: DQ and DBI_n | $V_{ILD2}(AC)$ | | | $V_{REFD2}$ - 0.36 | | | $V_{REFD2}$ - 0.333 | V | |
| Clock input differential voltage: CK_t, CK_c | $V_{IDCK}(AC)$ | 0.36 | | | 0.333 | | | V | 1, 3, 5 |
| Clock input differential voltage: WCK_t, WCK_c | $V_{IDWCK}(AC)$ | 0.27 | | | 0.25 | | | | 1, 4, 6 |
| Clock input crossing point voltage; CK_t, CK_c | $V_{IXCK}(AC)$ | $V_{REFC}$ - 0.108 | | $V_{REFC}$ + 0.108 | $V_{REFC}$ - 0.10 | | $V_{REFC}$ + 0.10 | V | 1, 2, 5 |
| Clock input crossing point voltage: WCK_t, WCK_c | $V_{IXWCK}(AC)$ | $V_{REFD}$ - 0.09 | | $V_{REFD}$ + 0.09 | $V_{REFD}$ - 0.09 | | $V_{REFD}$ + 0.09 | V | 1, 2, 6, 7 |
| Allowed time before ring back of CK/WCK below $V_{IDCK}(AC)/V_{IDWCK}(AC)$ | $t_{DVAC}$ | | | | | | | ps | 8, 9, 10 |

NOTE 1 For AC operations, all DC clock requirements must be satisfied as well.

NOTE 2 The value of $V_{IXCK}$ and $V_{IXWCK}$ is expected to equal 70% $V_{DDQ}$ for the transmitting device and must track variations in the DC level of the same.

NOTE 3 $V_{IDCK}$ is the magnitude of the difference between the input level in CK_t and the input level on CK_c. The input reference level for signals other than CK_t and CK_c is $V_{REFC}$.

NOTE 4 $V_{IDWCK}$ is the magnitude of the difference between the input level in WCK_t and the input level on WCK_c. The input reference level for signals other than WCK_t and WCK_c is either $V_{REFC}$, $V_{REFC2}$, $V_{REFD}$ or $V_{REFD2}$.

NOTE 5 The CK_t and CK_c input reference level (for timing referenced to CK_t and CK_c) is the point at which CK_t and CK_c cross. Please refer to the applicable timings in the AC Timings table.

NOTE 6 The WCK_t and WCK_c input reference level (for timing referenced to WCK_t and WCK_c) is the point at which WCK_t and WCK_c cross. Please refer to the applicable timings in the AC Timings table.

NOTE 7 $V_{REFD}$ is either $V_{REFD}$ or $V_{REFD2}$

NOTE 8 Figure 105 illustrates the exact relationship between (CK_t - CK_c) or (WCK_t - WCK_c) and $V_{ID}(AC)$, $V_{ID}(DC)$ and $t_{DVAC}$.

NOTE 9 Ring back below $V_{ID}(DC)$ is not allowed.

NOTE 10 $t_{DVAC}$ is not measured in and of itself as a compliance specification, but is relied upon in measurement of clock operating conditions and clock related parameters.

NOTE 11 The AC Operating conditions are for DRAM design only and are valid on the silicon at the input of the receiver. They are not intended to be measured.

## 8.6 AC & DC OPERATING CONDITIONS (cont'd)

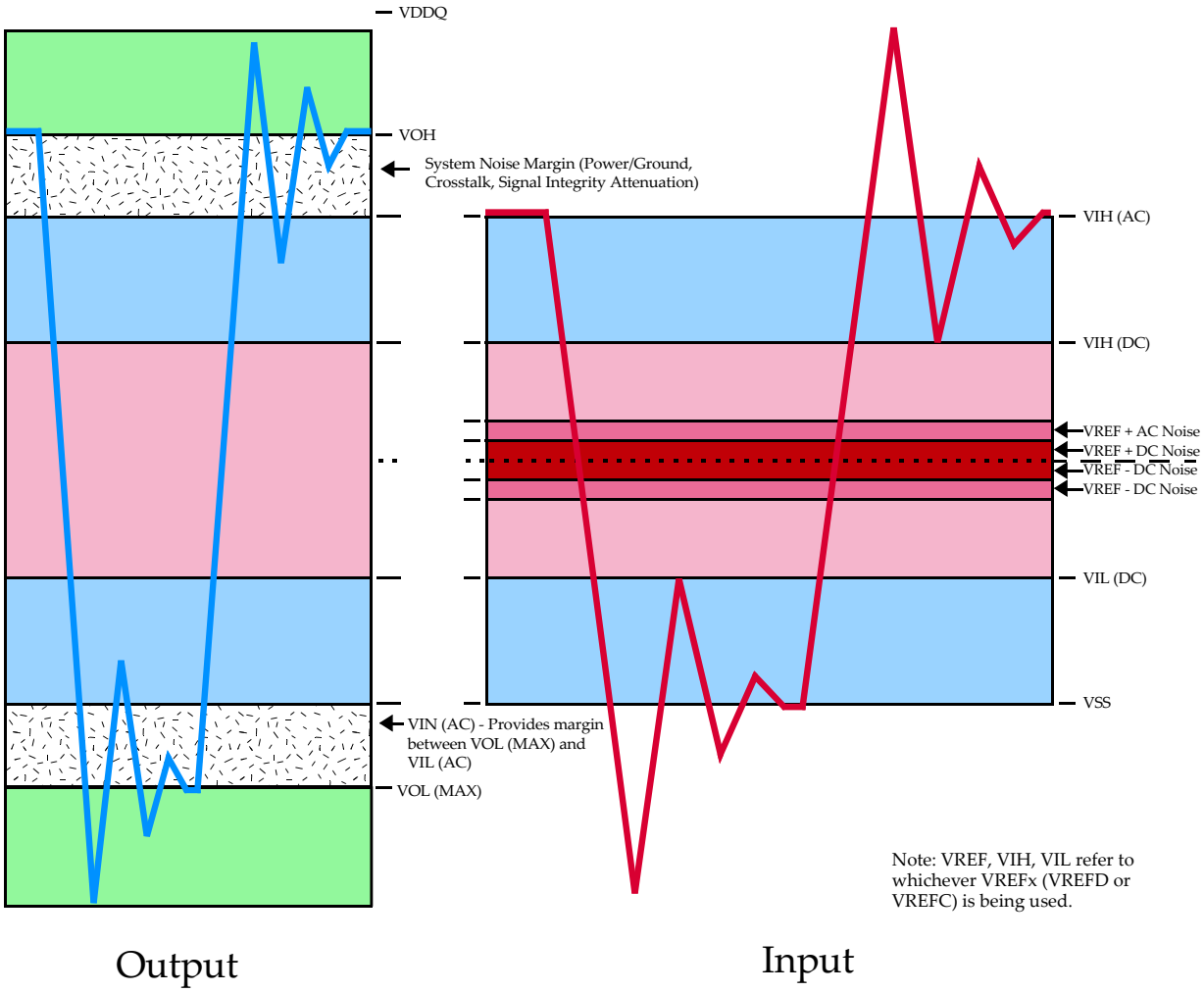



**Figure 107 — Voltage Waveform**

## 8.6 AC & DC OPERATING CONDITIONS (cont'd)

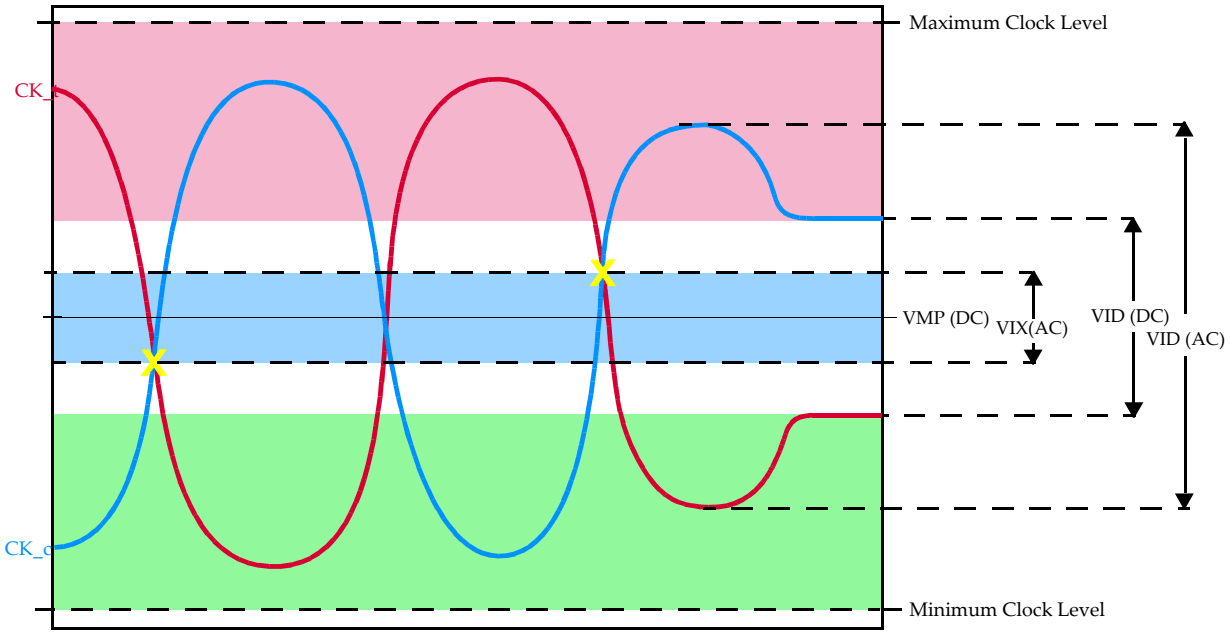



**Figure 108 — Clock Waveform**

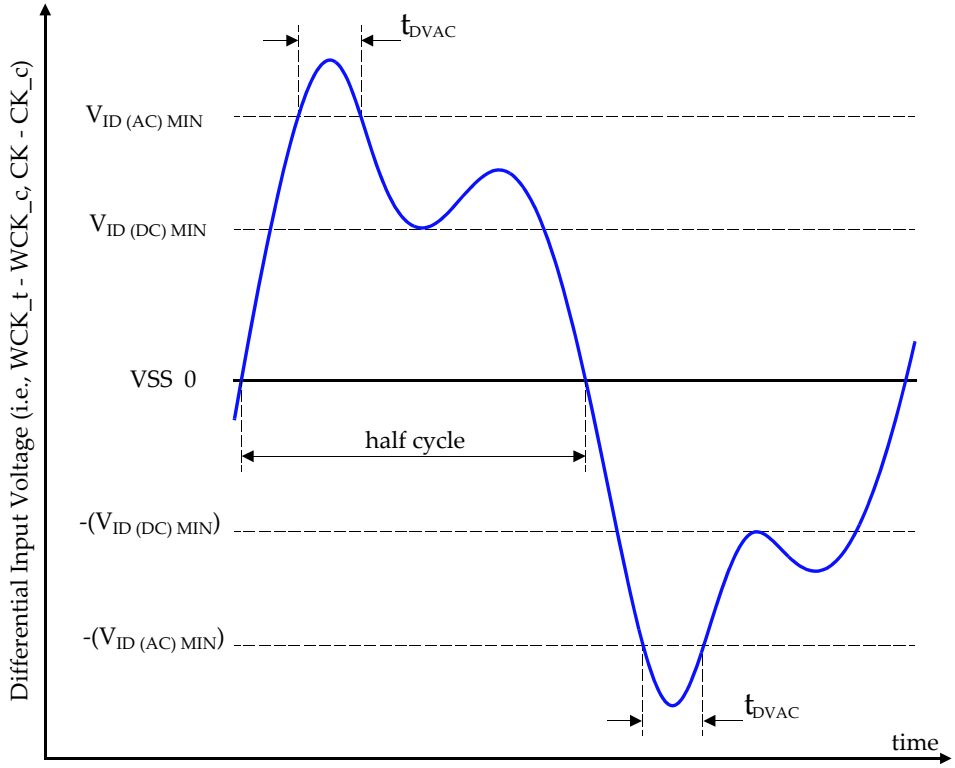



**Figure 109 — Definition of differential ac-swing and "time above ac-level" $t_{DVAC}$**

## 8.7 POD I/O SYSTEM

The POD I/O system is optimized for small systems with very high data rates. The system allows a single Master device to control one or two slaves in the case of GDDR6. The POD driver uses either a 60/40 Ohm output impedance that drives into a 60 Ohm equivalent terminator tied to $V_{DDQ}$ or a 48/40 Ohm output impedance that drives into a 48 Ohm equivalent terminator tied to $V_{DDQ}$. Single and dual load systems are shown as follows:

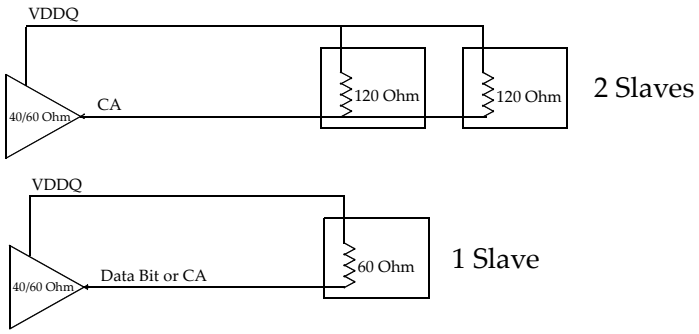



**Figure 110 — System Configurations**

The POD Master I/O cell is comprised of a 60/40 Ohm driver and a terminator of 60 Ohms or a 48/40 Ohm driver and a terminator of 48 Ohms. The Master POD cell's terminator is disabled when the output driver is enabled. The basic cell is shown in Figure 108.

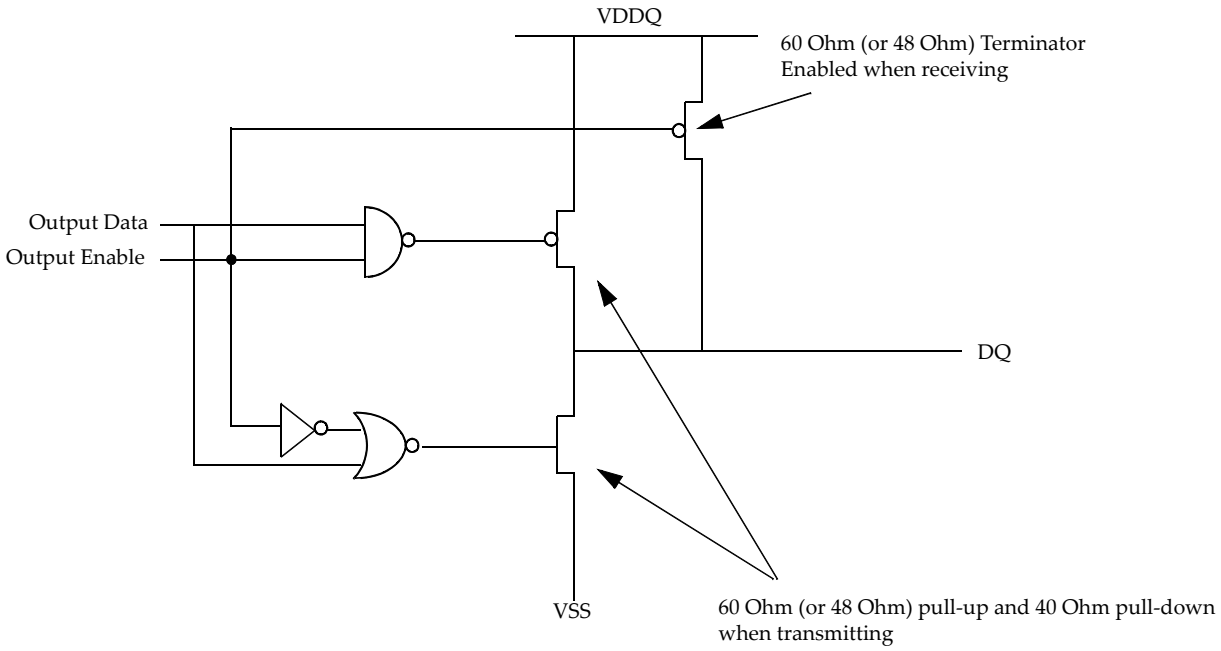



**Figure 111 — Master I/O Cell**

## 8.7 POD I/O SYSTEM (cont'd)

The POD Slave I/O cell is comprised of a 60/40 ohm driver and programmable terminator of 60 or 120 ohms for GDDR6 or a 48/40 ohm driver and programmable terminator of 48 or 120 ohms for GDDR6. The Slave POD cell's terminator is disabled when the output driver is enabled or any other Slave output driver is enabled. The basic cell is shown in Figure 109.

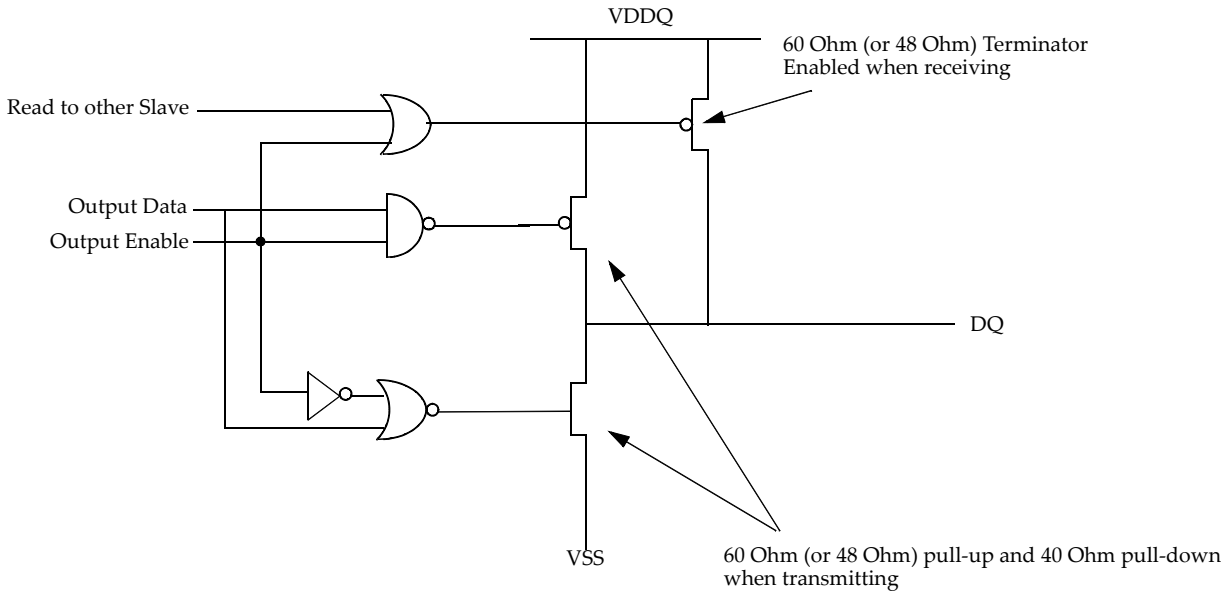



**Figure 112 — Slave I/O Cell**

The POD Master and Slave I/O cells are intended to have their driver and terminators combined together to minimize the area needed to implement the cell and reduce input capacitance. For GDDR6 this is possible by using three 120 Ohm driver/terminator sub cells that are connected in parallel to result in a 60/40 ohm driver and programmable terminator of 60 or 120 ohms. The combinations used are as follows.

**Table 60 — POD I/O Sub Cells, 120 ohm Based**

| # of 120 ohm Sub Cells Enabled | Resulting Impedance | Use |
|---|---|---|
| 1 | 120 Ohms | 2 Slave loads |
| 2 | 60 Ohms | 1 Slave load or Master terminator |
| 3 | 40 Ohms | Master or Slave Driver |

To ensure that the target impedance is achieved the POD I/O cell is designed to be calibrated to an external 1% precision resistor.

The following procedure may be used to calibrate the cell:

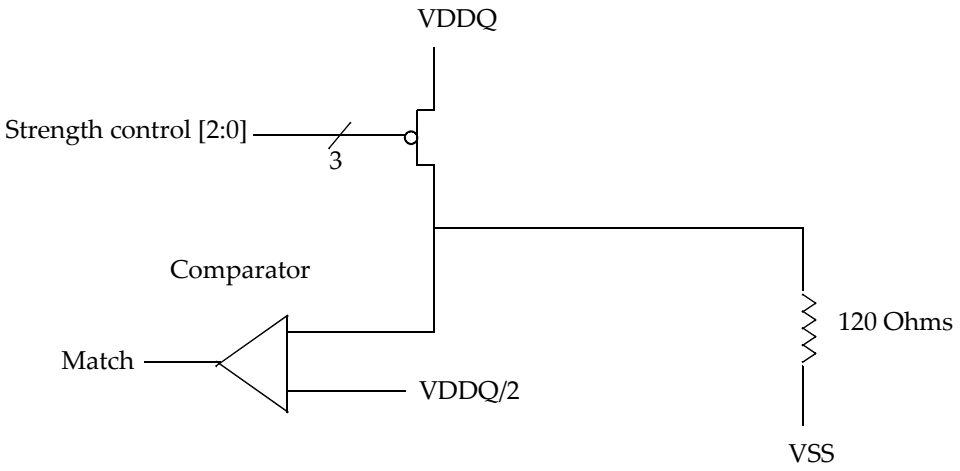
1.) First calibrate the PMOS device against a 120 Ohm resister to VSS via the ZQ pin as illustrated in Figure 110.

- Set Strength Control to minimum setting
- Increase drive strength until comparator detects data bit is greater than $V_{DDQ}/2$
- PMOS device is calibrated to 120 Ohms

2.) Then calibrate the NMOS device against the calibrated 120 Ohm PMOS device as illustrate in Figure 111.

- Set Strength Control to minimum setting
- Increase drive strength until comparator detects data bit is less than $V_{DDQ}/2$
- NMOS device is now calibrated to 120 Ohms

## 8.7 POD I/O SYSTEM (cont'd)




When Match PMOS leg is calibrated to 120 ohms

**Figure 113 — PMOS Calibration**

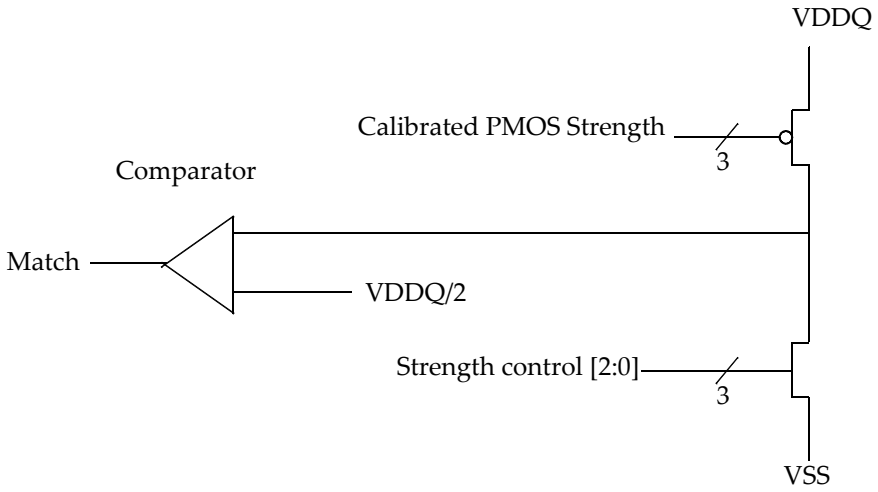



**Figure 114 — NMOS Calibration**

## 8.8 IDD and IPP PARAMETERS and TEST CONDITIONS

This chapter defines IDD and IPP measurement conditions such as test load and patterns. Figure 112 shows the setup and test load for IDD and IPP measurements.

- IDD currents are measured as time-averaged currents, with all $V_{DD}$ balls of the device under test tied together. IDDQ and IPP currents are not included in IDD currents.
- IPP currents are measured as time-averaged currents with all $V_{PP}$ balls of the device under test tied together. IDD and IDDQ currents are not included in IPP currents:

For IDD and IPP measurements, the following definitions apply:

- “0” and “LOW” is defined as VIN <= VILAC(max);
- “1” and “HIGH” is defined as VIN >= VIHAC(min);
- CABI and DBI are enabled;
- All ODTs are enabled with ZQ/2 for Data, CA and WCK inputs;
- WLmrs, RLmrs, RAS, WR are set to appropriate values;
- CRC is enabled for READs and WRITEs with appropriate CRCWL and CRCRL settings;
- the EDC hold pattern is programmed to'0101';
- Bank groups are enabled if required for device operation at $t_{CK}$(min);
- Command Address (CA) inputs include CABI_n;
- Each data byte consists of eight DQs and one DBI_n;
- NOP with all CA inputs pulled HIGH during idle command cycles;
- Data pattern (one burst of 16) used with IDD1, IDD4R, IDD4W and IDD7 pattern:
  - DATA0 is vendor specific. For example, 55h 1Eh 55h 1Eh 55h 1Eh 55h 1Eh 55h 1Eh 55h 1Eh 55h 1Eh 55h 1Eh
  - DATA1 is vendor specific. For example, 1Eh 55h 1Eh 55h 1Eh 55h 1Eh 55h 1Eh 55h 1Eh 55h 1Eh 55h 1Eh 55h
- Basic IDD and IPP Measurement Conditions including timings used for IDD Measurement-Loop Patterns are described in Table 61.
- IDD Measurements are done after properly initializing the device. This includes the pre-load of the memory array with data pattern used with IDD1, IDD4R and IDD7 measurements.
- The IDD and IPP Measurement-Loop patterns shall be executed at least once before actual measurement is started.
- The measurements should be first taken with the device configured to x16 mode first and the x8 mode (2 separate measurements)
- Measurements are taken per device with the same IDD Measurement-Loop Patterns on both channels

## 8.8 IDD and IPP PARAMETERS and TEST CONDITIONS (cont'd)

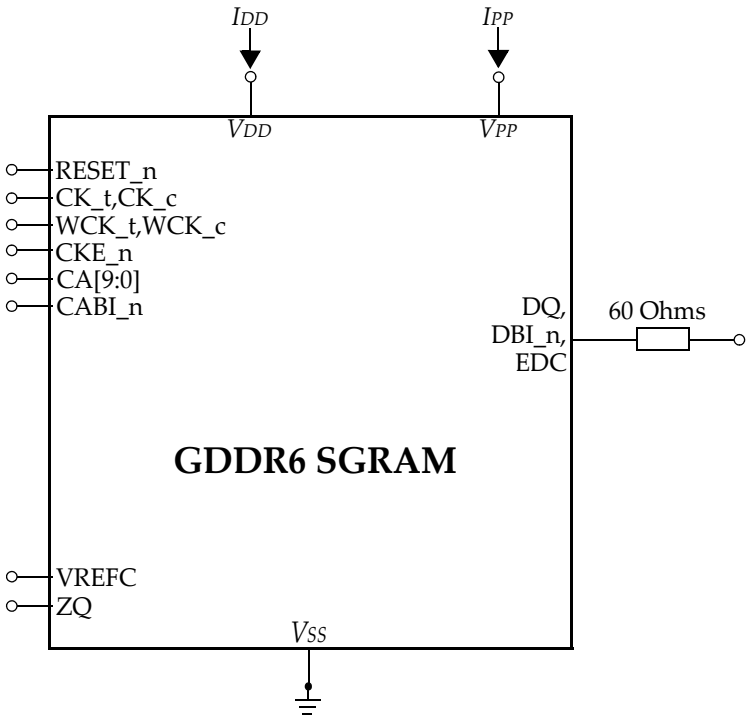



**Figure 115 — Measurement Setup and Test Load for IDD and IPP Measurements**

## 8.8 IDD and IPP PARAMETERS and TEST CONDITIONS (cont'd)

### Table 61 — IDD Specifications and Test Conditions

| PARAMETER/CONDITION | SYMBOL | NOTES |
|---|---|---|
| **One Bank Activate Precharge Current**: $t_{CK} = t_{CK}(min)$; $t_{WCK} = t_{WCK}(min)$; $t_{RC} = t_{RC}(min)$; CKE_n = LOW; DQ, DBI_n are HIGH; bank and row addresses (5 CA inputs set LOW) as defined in Table 62 with ACT command; AC timings as defined in Table 67 | IDD0 | 1 |
| | IPP0 | 1, 2 |
| **One Bank Activate Read Precharge Current**: $t_{CK} = t_{CK}$ (min); $t_{WCK} = t_{WCK}(min)$; $t_{RC} = t_{RC}(min)$; CKE_n = LOW; one bank activated; single read burst with 50% data toggle on each data transfer, with 4 outputs per data byte driven LOW; otherwise DQ, DBI_n are HIGH; bank, row and column addresses (5 CA inputs set LOW) as defined in Table 63 with ACT and READ commands; AC timings as defined in Table 67; $I_{OUT}$ = 0mA | IDD1 | 1 |
| **Precharge Power-down Current**: $t_{CK} = t_{CK}$ (min); $t_{WCK} = t_{WCK}(min)$; all banks idle; CKE_n = HIGH; all other inputs are HIGH; PLL/DLLs are off | IDD2P | |
| **Precharge Standby Current**: $t_{CK} = t_{CK}$ (min); $t_{WCK} = t_{WCK}(min)$; all banks idle; CKE_n = LOW; all other inputs are HIGH | IDD2N | |
| **Active Power-down Current**: $t_{CK} = t_{CK}$ (min); $t_{WCK} = t_{WCK}(min)$; one bank active; CKE_n = HIGH; all other inputs are HIGH | IDD3P | |
| **Active Standby Current**: $t_{CK} = t_{CK}$ (min); $t_{WCK} = t_{WCK}(min)$; one bank active; CKE_n = LOW; all other inputs are HIGH | IDD3N | |
| | IPP3N | 1, 2 |
| **Read Burst Current**: $t_{CK} = t_{CK}$ (min); $t_{WCK} = t_{WCK}(min)$; CKE_n = LOW; one bank in each of the 4 bank groups activated; continuous read burst across bank groups with 50% data toggle on each data transfer, with 4 outputs per data byte driven LOW as defined in Table 64; bank and column addresses (5 CA inputs set LOW) as defined in Table 64; with READ command; $I_{OUT}$ = 0mA | IDD4R | |
| **Write Burst Current**: $t_{CK} = t_{CK}$ (min); $t_{WCK} = t_{WCK}(min)$; CKE_n = LOW; one bank in each of the 4 bank groups activated; continuous write burst across bank groups with 50% data toggle on each data transfer, with 4 inputs per data byte set LOW; as defined in Table 65; bank and column addresses (5 CA inputs set LOW) as defined in Table 65; with WRITE command; no data mask | IDD4W | |
| **Refresh Current**: $t_{CK} = t_{CK}$ (min); $t_{WCK} = t_{WCK}(min)$; $t_{RFCab} = t_{RFCab}(min)$ as defined in Table 67; CKE_n = LOW; DQ, DBI_n are HIGH; CA inputs are HIGH | IDD5 | 1 |
| | IPP5 | 1, 2 |
| **Self Refresh Current**: CKE_n= HIGH; all other inputs are HIGH | IDD6 | |
| **Four Bank Interleave Read Current**: $t_{CK} = t_{CK}(min)$; $t_{WCK} = t_{WCK}(min)$; CKE_n = LOW; one bank in each of the 4 bank groups activated and precharged at $t_{RC}(min)$; continuous read burst across bank groups with 50% data toggle on each data transfer, with 4 outputs per data byte driven LOW as defined in Table 66; bank, row and column addresses (5 CA inputs set LOW) as defined in Table 66; with ACT and READ/READA commands; $I_{OUT}$ = 0mA | IDD7 | |
| | IPP7 | 1, 2 |

NOTE 1: Min $t_{RC}$ or $t_{RFCab}$ for IDD measurements is the smallest multiple of $t_{CK}$ that meets the minimum of the absolute value for the respective parameter.

NOTE 2: IPP currents have the same definition as IDD except that the current on the VPP supply is measured. IPP0 test and limit is applicable for IDD0 and IDD1 conditions. IPP3N test and limit is applicable for all IDD2X, IDD3X, IDD4X and IDD6 conditions.

## Table 61.1 – IDD Specifications and Test Conditions (1.35V)

| Symbol | 12Gbps @1.35V | | 14Gbps @1.35V | |
|---|---|---|---|---|
| | X16 | X8 | X16 | X8 |
| IDD0 | 240 | 210 | 250 | 220 |
| IPP0 | 15 | 10 | 15 | 10 |
| IDD1 | 250 | 220 | 260 | 230 |
| IDD2P | 110 | 100 | 110 | 110 |
| IDD2N | 180 | 180 | 180 | 180 |
| IDD3P | 120 | 110 | 120 | 120 |
| IDD3N | 190 | 190 | 200 | 190 |
| IPP3N | 7 | 7 | 7 | 7 |
| IDD4R | 910 | 580 | 1040 | 650 |
| IDD4W | 760 | 660 | 860 | 740 |
| IDD5 | 550 | 550 | 610 | 610 |
| IPP5 | 90 | 90 | 110 | 110 |
| IDD6N | 110 | 110 | 110 | 110 |
| IDD7 | 1190 | 720 | 1380 | 810 |
| IPP7 | 40 | 20 | 45 | 25 |

## Table 61.2 – IDD Specifications and Test Conditions (1.25V)

| Symbol | 10Gbps @1.25V | | 12Gbps @1.25V | |
|---|---|---|---|---|
| | X16 | X8 | X16 | X8 |
| IDD0 | 200 | 180 | 210 | 190 |
| IPP0 | 10 | 10 | 10 | 10 |
| IDD1 | 210 | 180 | 220 | 190 |
| IDD2P | 100 | 90 | 100 | 100 |
| IDD2N | 150 | 150 | 160 | 150 |
| IDD3P | 100 | 100 | 110 | 110 |
| IDD3N | 170 | 160 | 170 | 170 |
| IPP3N | 7 | 7 | 7 | 7 |
| IDD4R | 710 | 470 | 830 | 530 |
| IDD4W | 600 | 520 | 690 | 590 |
| IDD5 | 450 | 450 | 520 | 520 |
| IPP5 | 80 | 80 | 90 | 90 |
| IDD6N | 100 | 100 | 100 | 100 |
| IDD7 | 940 | 570 | 1100 | 660 |
| IPP7 | 35 | 20 | 40 | 20 |

## 8.8 IDD and IPP PARAMETERS and TEST CONDITIONS (cont'd)

Table 62 — IDD0 Measurement-Loop Pattern

| Sub-Loop | Cycle Number | Command | Clock | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 | CABI_n | Data |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | ACT | Rising | H | L | H | H | H | H | H | L | L | L | L | – |
| | | | Falling | L | L | H | H | H | H | H | H | L | L | H | |
| | 1 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | ... | repeat NOP until cycle (nRAS -1) | | | | | | | | | | | | | |
| | nRAS | PRE | Rising | H | L | L | L | L | L | H | H | H | H | H | – |
| | | | Falling | L | L | H | H | H | H | H | H | H | H | H | |
| | nRAS + 1 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | ... | repeat NOP until cycle (nRC -1) | | | | | | | | | | | | | |
| 1 | nRC | repeat sub-loop 0 pattern until cycle (2* nRC -1); use BA = 03h and RA = 03C7h instead | | | | | | | | | | | | | |
| 2 | 2* nRC | repeat sub-loop 0 pattern until cycle (3* nRC -1); use BA = 04h and RA = 07C7h instead | | | | | | | | | | | | | |
| 3 | 3* nRC | repeat sub-loop 0 pattern until cycle (4* nRC -1); use BA = 07h and RA = 01C7h instead | | | | | | | | | | | | | |
| NOTE 1 IDD test data & command pattern is vendor specific. Users should refer to vendor's GDDR6 datasheet. | | | | | | | | | | | | | | | |

Table 63 — IDD1 Measurement-Loop Pattern

| Sub-Loop | Cycle Number | Command | Clock | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 | CABI_n | Data |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | ACT | Rising | H | L | H | H | H | H | H | L | L | L | L | – |
| | | | Falling | L | L | H | H | H | H | H | H | L | L | H | |
| | 1 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | ... | repeat pattern until cycle (nRCDRD -1) | | | | | | | | | | | | | |
| | nRCDRD | READ | Rising | L | L | H | H | H | H | L | L | H | H | L | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | nRCDRD + 1 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | ... | repeat pattern until cycle (nRAS -1) | | | | | | | | | | | | | |
| | nRAS | PRE | Rising | H | L | L | L | L | L | H | H | H | H | H | – |
| | | | Falling | L | L | H | H | H | H | H | H | H | H | H | |
| | nRAS + 1 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | ... | repeat pattern until cycle (nRC -1) | | | | | | | | | | | | | |
| 1 | nRC | repeat sub-loop 0 pattern until cycle (2* nRC -1); use BA = 03h, RA = 03C7h, CA = 23h and DATA1 instead | | | | | | | | | | | | | |
| 2 | 2* nRC | repeat sub-loop 0 pattern until cycle (3* nRC -1); use BA = 04h, RA = 07C7h, CA = 4Ch and DATA0 instead | | | | | | | | | | | | | |
| 3 | 3* nRC | repeat sub-loop 0 pattern until cycle (4* nRC -1); use BA = 07h, RA = 01C7h, CA = 23h and DATA1 instead | | | | | | | | | | | | | |
| NOTE 1 IDD test data & command pattern is vendor specific. Users should refer to vendor's GDDR6 datasheet. | | | | | | | | | | | | | | | |

## 8.8 IDD and IPP PARAMETERS and TEST CONDITIONS (cont'd)

Table 64 — IDD4R Measurement-Loop Pattern

| Sub-Loop | Cycle Number | Command | Clock | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 | CABI_n | Data |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | READ | Rising | H | H | H | L | L | L | H | H | L | L | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 1 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 2 | READ | Rising | H | H | L | L | H | H | L | L | H | H | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 3 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 4 | READ | Rising | H | H | L | H | L | L | H | H | L | L | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 5 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 6 | READ | Rising | H | H | L | H | H | H | L | L | H | H | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 7 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |

NOTE 1 IDD test data & command pattern is vendor specific. Users should refer to vendor's GDDR6 datasheet.

**Table 65 —** IDD4W Measurement-Loop Pattern

| Sub-Loop | Cycle Number | Command | Clock | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 | CABI_n | Data |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | WRITE | Rising | H | H | H | L | L | L | H | H | L | L | H | DATA0 |
| | | | Falling | H | H | H | H | L | H | L | L | H | H | L | |
| | 1 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 2 | WRITE | Rising | H | H | L | L | H | H | L | L | H | H | H | DATA1 |
| | | | Falling | H | H | H | H | L | H | L | H | L | H | L | |
| | 3 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 4 | WRITE | Rising | H | H | L | H | L | L | H | H | L | L | H | DATA0 |
| | | | Falling | H | H | H | H | L | H | L | L | H | H | L | |
| | 5 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 6 | WRITE | Rising | H | H | L | H | H | H | L | L | H | H | H | DATA1 |
| | | | Falling | H | H | H | H | L | H | L | H | L | H | L | |
| | 7 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |

NOTE 1 IDD test data & command pattern is vendor specific. Users should refer to vendor's GDDR6 datasheet.

## 8.8 IDD and IPP PARAMETERS and TEST CONDITIONS (cont'd)

**Table 66 —** IDD7 Measurement-Loop Pattern

| Sub-Loop | Cycle Number | Command | Clock | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 | CABI_n | Data |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | ACT | Rising | L | H | H | L | L | L | L | H | H | H | H | – |
| | | | Falling | L | L | L | H | H | H | H | H | L | H | L | |
| | 1 | READ | Rising | H | H | L | L | H | H | L | L | H | H | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 2 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 3 | READ | Rising | H | H | L | H | H | H | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 4 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 5 | READ | Rising | H | H | L | H | H | H | L | L | H | H | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 6 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 7 | READ | Rising | H | H | L | L | H | H | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 8 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 9 | READA | Rising | H | H | L | L | H | H | L | L | H | H | H | DATA0 |
| | | | Falling | L | H | L | L | H | H | H | L | H | L | H | |
| | 10 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 11 | READ | Rising | H | H | L | H | H | H | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 12 | ACT | Rising | L | H | L | H | L | L | L | H | H | H | H | – |
| | | | Falling | L | L | L | H | H | H | H | H | L | L | H | |
| | 13 | READ | Rising | H | H | L | H | H | H | L | L | H | H | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 14 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 15 | READ | Rising | H | H | H | L | L | L | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 16 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 17 | READ | Rising | H | H | H | L | L | L | L | L | H | H | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 18 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 19 | READ | Rising | H | H | L | H | H | H | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 20 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 21 | READA | Rising | H | H | L | H | H | H | L | L | H | H | H | DATA0 |
| | | | Falling | L | H | L | L | H | H | H | L | H | L | H | |
| | 22 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 23 | READ | Rising | H | H | L | L | H | H | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |

| Sub-Loop | Cycle Number | Command | Clock | CA9 | CA8 | CA7 | CA6 | CA5 | CA4 | CA3 | CA2 | CA1 | CA0 | CABI_n | Data |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 24 | ACT | Rising | L | H | H | L | L | L | L | H | H | H | H | – |
| | | | Falling | H | H | H | H | L | L | L | L | H | H | L | |
| | 25 | READ | Rising | H | H | H | L | L | L | L | L | H | H | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 26 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| 0 | 27 | READ | Rising | H | H | L | H | L | L | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 28 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 29 | READ | Rising | H | H | L | H | L | L | L | L | H | H | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 30 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 31 | READ | Rising | H | H | H | L | L | L | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 32 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 33 | READA | Rising | H | H | H | L | L | L | L | L | H | H | H | DATA0 |
| | | | Falling | L | H | L | L | H | H | H | L | H | L | H | |
| | 34 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 35 | READ | Rising | H | H | L | H | L | L | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 36 | ACT | Rising | L | H | L | H | H | H | L | H | H | H | H | – |
| | | | Falling | H | L | H | H | H | L | L | L | H | H | L | |
| | 37 | READ | Rising | H | H | L | H | L | L | L | L | H | H | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 38 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 39 | READ | Rising | H | H | L | L | H | H | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | L | |
| | 40 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 41 | READ | Rising | H | H | L | L | H | H | L | L | H | H | H | DATA0 |
| | | | Falling | H | L | H | H | L | H | L | H | L | H | L | |
| | 42 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 43 | READ | Rising | H | H | L | H | L | L | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | L | L | H | H | l | |
| | 44 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 45 | READA | Rising | H | H | L | H | L | L | L | L | H | H | H | DATA0 |
| | | | Falling | L | H | L | L | H | H | H | L | H | L | H | |
| | 46 | NOP | Rising | H | H | H | H | H | H | H | H | H | H | H | – |
| | | | Falling | H | H | H | H | H | H | H | H | H | H | H | |
| | 47 | READ | Rising | H | H | L | L | H | H | H | H | L | L | H | DATA1 |
| | | | Falling | H | L | H | H | L | H | K | K | H | H | L | |

NOTE 1 IDD test data & command pattern is vendor specific. Users should refer to vendor's GDDR6 datasheet.

## 8.8 IDD and IPP PARAMETERS and TEST CONDITIONS

**Table 67 —** AC parameter set for IDD test

| Symbol | Data rate | | | | Unit |
|---|---|---|---|---|---|
| | 10Gbps | 12Gbps | | 14Gbps | |
| | 1.25V | 1.25V | 1.35V | 1.35V | |
| $t_{CK}$ | 0.80 | 0.67 | 0.67 | 0.57 | ns |
| RL | 22 | 24 | 22 | 25 | $t_{CK}$ |
| $t_{RCDRD}$ | 15 | 15 | 15 | 15 | ns |
| $t_{RCDWR}$ | 12 | 12 | 12 | 12 | ns |
| $t_{RAS}$ | 32 | 32 | 32 | 32 | ns |
| $t_{RP}$ | 15 | 15 | 15 | 15 | ns |
| $t_{RC}$ | 47 | 47 | 47 | 47 | ns |
| $t_{FAW}$ | 20 | 20 | 20 | 20 | ns |
| $t_{RRDS}$ | Max(4ns, 4tCK) | Max(4ns, 4tCK) | Max(4ns, 4tCK) | Max(4ns, 4tCK) | ns |
| $t_{RRDL}$ | Max(5ns, 4tCK) | Max(5ns, 4tCK) | Max(5ns, 4tCK) | Max(5ns, 4tCK) | ns |
| $t_{RFCab}$ | 120 | 120 | 120 | 120 | ns |

**Table 68 — Self Refresh Current Definitions**

| Symbol | Temperature Range | Value | | Unit | Note |
|---|---|---|---|---|---|
| | | VDD 1.35V | VDD 1.25V | | |
| IDD6N | $0^oC$ - $T_N$ | 120 | 100 | mA | 2, 3, 8 |
| IDD6E (optional) | $0^oC$ - $T_E$ | | | mA | 1, 3, 4, 9 |
| IDD6R (optional) | $0^oC$ - $T_R$ | | | mA | 5, 10, 14 |
| IDD6A (optional) | $0^oC$ - $T_a$ | | | mA | 3, 5, 6, 7 |
| | $T_b$ - $T_y$ (optional) | | | mA | 3, 5, 6, 7 |
| | $T_z$ - $T_{OPERmax}$[11] | | | mA | 3, 5, 6, 7 |

NOTE 1 Max. values for IDD currents considering worst case conditions of process, temperature and voltage.
NOTE 2 Applicable for MR2 OP[7:6] = 0.
NOTE 3 Supplier data sheets include a max value.
NOTE 4 IDD6E is only specified for devices which support the Extended Temperature Range feature. Refer to the supplier datasheet for the appropriate self refresh setting in MR2 OP[7:6]
NOTE 5 Refer to the supplier data sheet for the value specification method (e.g., max, typical) for IDD6E and IDD6A.
NOTE 6 IDD6A is only specified for devices which support the Temperature Controlled Self Refresh feature enabled by MR2 OP[7:6]
NOTE 7 The number of discrete temperature ranges supported and the associated $T_a$ - $T_z$, and $T_{OPERmax}$ values are supplier/design specific. Temperature ranges are intended to denote the nominal trip points for the internal temperature sensor to bracket discrete self refresh rates internal to the DRAM. Refer to supplier datasheet for more information.
NOTE 8 When TCSR is disabled in the DRAM, $T_N$ represents the temperature limit for normal operation of the DRAM.
NOTE 9 When TCSR is disabled, some DRAMs may support an extended temperature range that is typically 10 $^oC$ higher than $T_N$ and often requires additional refresh cycles. $T_E$ represents this extended temperature limit.
NOTE 10 $T_R$ represents the temperature used to reflect the current consumed in a typical room temperature environment.
NOTE 11 $T_{OPERmax}$ represents the max temperature supported by the DRAM when TCSR is enabled.
NOTE 12 IDD6H is for hibernate self refresh as enabled with MR7 OP=1. IDD6H also applies in the case that the optional hibernate self refresh with VDDQ off is supported and enabled with MR 12 OP0=1. Hibernate self refresh and hibernate self refresh with VDDQ off are allowed to be entered over the full operating temperature range (up to either $T_N$ or $T_E$).
NOTE 13 IPP3N test and limit is applicable for all IDD6 conditions
NOTE 14 IDD6R and IDD6H values are characterized but not necessarily tested, see vendor data sheets for more details.

## 8.9 AC TIMINGS

**Table 69 — AC Timings**

| PARAMETER 1, 2 | | | SYMBOL | 14Gbps @1.35V | | 12Gbps @1.35V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|---|---|
| | | | | MIN | MAX | MIN | MAX | | |
| **CK_t and WCK_t Timings** | | | | | | | | | |
| CK clock cycle time | | PLL off | $t_{CK}$ | 0.571 | 20 | 0.667 | 20 | ns | |
| CK clock HIGH-level width | | | $t_{CH}$ | 0.47 | 0.53 | 0.47 | 0.53 | $t_{CK}$ | |
| CK clock LOW-level width | | | $t_{CL}$ | 0.47 | 0.53 | 0.47 | 0.53 | $t_{CK}$ | |
| Min. CK clock half period | | | $t_{HP}$ | min(tCH,tCL) | - | min(tCH,tCL) | - | $t_{CK}$ | |
| CK clock frequency | DDR WCK | PLL off | $f_{CK}$ | 50 | | 50 | | MHz | |
| CK clock frequency with bank groups disabled | | | $f_{CKBG}$ | fCK(min) | 1750 | fCK(min) | 1750 | MHz | 3, 4 |
| CK clock frequency with bank groups enabled and $t_{CCDL}$ = 3 $t_{CK}$ | | | $f_{CKBG3}$ | fCK(min) | 1750 | fCK(min) | 1750 | MHz | 3 |
| CK clock frequency with WCK2CK alignment at the balls | | | $f_{CKPIN}$ | fCK(min) | 500 | fCK(min) | 500 | MHz | 5 |
| CK clock frequency in RDQS Mode | | | $f_{CKRDQS}$ | fCK(min) | 500 | fCK(min) | 500 | MHz | 6 |
| CK clock frequency for operation with $V_{REFC2}$ | | | $f_{CKVREFC2}$ | fCK(min) | | fCK(min) | | MHz | 6 |
| CK clock frequency for operation with $V_{REFD2}$ | | | $f_{CKVREFD2}$ | fCK(min) | | fCK(min) | | MHz | 7 |
| CK clock frequency for WCK-to-CK auto synchronization in WCK2CK training mode | | | $f_{CKAUTOSYNC}$ | fCK(min) | 500 | fCK(min) | 500 | MHz | 8 |
| CK clock frequency for device operation with Low Frequency mode enabled | | | $f_{CKLF}$ | fCK(min) | 500 | fCK(min) | 500 | MHz | 10 |
| WCK clock cycle time | DDR WCK | PLL off | $t_{WCK}$ | 0.143 | 5 | 0.143 | 5 | ns | 11 |
| WCK clock HIGH-level width | | | $t_{WCKH}$ | 0.47 | 0.53 | 0.47 | 0.53 | $t_{WCK}$ | 12 |
| WCK clock LOW-level width | | | $t_{WCKL}$ | 0.47 | 0.53 | 0.47 | 0.53 | $t_{WCK}$ | 12 |
| Min. WCK clock half period | | | $t_{WCKHP}$ | min(tWCKH, tWCKL) | - | min(tWCKH, tWCKL) | - | $t_{WCK}$ | |
| **CKE_n and CA Input Timings** | | | | | | | | | |
| Command Address (CA) input setup time | | | $t_{AS}$ | -0.09 | - | -0.09 | - | ns | 13, 15, 54 |
| Command Address (CA) input hold time | | | $t_{AH}$ | 0.09 | - | 0.09 | - | ns | 13, 15, 54 |
| Command Address (CA) input pulse width | | | $t_{APW}$ | 0.26 | - | 0.26 | - | ns | 13, 14, 15, 54 |
| CKE_n input setup time | | | $t_{CKES}$ | | - | | - | ns | 15, 54 |

## Table 69 — AC Timings

| PARAMETER 1, 2 | | | SYMBOL | 14Gbps @1.35V | | 12Gbps @1.35V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|---|---|
| | | | | MIN | MAX | MIN | MAX | | |
| CKE_n input hold time | | | $t_{CKEH}$ | | - | | - | ns | 13, 54 |
| CKE_n min. HIGH and LOW pulse width for SELF REFRESH, HIBERNATE SELF REFRESH and POWER-DOWN commands | | | $t_{CKE}$ | 10 | - | 10 | - | ns | 13, 14, 54 |
| CKE_n min. HIGH and LOW pulse width for CAT command | | | $t_{CKEPW}$ | | - | | - | ns | |
| **WCK2CK Timings** | | | | | | | | | |
| WCK stop to MRS delay for entering WCK2CK training | | | $t_{WCK2MRS}$ | 3 | - | 3 | - | ns | |
| MRS to WCK restart delay after entering WCK2CK training | | | $t_{MRSTWCK}$ | 10 | - | 10 | - | ns | 16 |
| WCK clock frequency for WCK2CK training with WCK stop | | | $f_{WCKSTOP}$ | - | | - | | MHz | |
| WCK start to WCK phase movement delay | | | $t_{WCK2TR}$ | 10 | - | 10 | - | $t_{CK}$ | |
| WCK phase change to phase detector out delay | | | $t_{WCK2PH}$ | 5 | - | 5 | - | ns | |
| WCK clock HIGH-level width during WCK2CK training | | | $t_{WCKHTR}$ | 0.43 | 0.57 | 0.43 | 0.57 | $t_{WCK}$ | 12, 17 |
| WCK clock LOW-level width during WCK2CK training | | | $t_{WCKLTR}$ | 0.43 | 0.57 | 0.43 | 0.57 | $t_{WCK}$ | 12, 17 |
| WCK2CK offset when zero offset at phase detector or at balls | PLL off; MR7 OP0=0 (at phase detector) | | $t_{WCK2CKPIN}$ | -0.2 | 0.2 | -0.2 | 0.2 | ns | 18 |
| | PLL off; MR7 OP0=1 (at balls) | | | -0.2 | 0.2 | -0.2 | 0.2 | | |
| WCK2CK phase offset upon WCK2CK training exit | DDR WCK | MR7 OP0=0 (at phase detector) | $t_{WCK2CKSYNC}$ | -0.25 | 0.25 | -0.25 | 0.25 | $t_{CK}$ | 19 |
| | | MR7 OP0=1 (at balls) | | -0.143 | 0.143 | -0.143 | 0.143 | ns | |
| WCK2CK phase offset | DDR WCK | MR7 OP0=0 (at phase detector) | $t_{WCK2CK}$ | -0.4 | 0.4 | -0.4 | 0.4 | $t_{CK}$ | 20 |
| | | MR7 OP0=1 (at balls) | | -0.228 | 0.228 | -0.228 | 0.228 | ns | |
| **Data Input and Output Timings** | | | | | | | | | |

## Table 69 — AC Timings

| PARAMETER 1, 2 | | SYMBOL | 14Gbps @1.35V | | 12Gbps @1.35V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | | |
| WCK to DQ/DBI_n offset for input data | PLL off | $t_{WCK2DQI}$ | 0.3 | 1.3 | 0.3 | 1.3 | ns | 21, 53 |
| WCK to DQ/DBI_n/EDC offset for output data | PLL off | $t_{WCK2DQO}$ | 0.5 | 1.5 | 0.5 | 1.5 | ns | 22, 23, 53 |
| DQ/DBI_n input pulse width | | $t_{DIPW}$ | | - | | - | ns | 24, 25, 26, 54 |
| DQ/DBI_n data input valid window | PLL off | $t_{DIVW}$ | | - | | - | ns | 24, 25, 27, 54 |
| DQ/DBI_n input skew | | $t_{DQDQI}$ | -0.1 | 0.1 | -0.1 | 1.0 | ns | 28, 53 |
| DQ/DBI_n/EDC output skew | | $t_{DQDQO}$ | -0.125 | 0.125 | -0.125 | 0.125 | ns | 19, 53 |
| **Row Access Timings** | | | | | | | | |
| ACTIVATE to ACTIVATE command period | | $t_{RC}$ | 47 | - | 47 | - | ns | |
| ACTIVATE to PREab / PREpb command period | | $t_{RAS}$ | 32 | 9*tREFI | 32 | 9*tREFI | ns | 30 |
| ACTIVATE to READ command delay | | $t_{RCDRD}$ | 15 | - | 15 | - | ns | |
| ACTIVATE to WRITE command delay | | $t_{RCDWR}$ | 12 | - | 12 | - | ns | |
| ACTIVATE to RDTR command delay | | $t_{RCDRTR}$ | 18 | - | 18 | - | ns | |
| ACTIVATE to WRTR command delay | | $t_{RCDWTR}$ | 10 | - | 10 | - | ns | |
| ACTIVATE to LDFF command delay | | $t_{RCDLTR}$ | 10 | - | 10 | - | ns | |
| REFab to RDTR or WRTR command delay | | $t_{REFTR}$ | 19 | - | 19 | - | ns | |
| ACTIVATE bank A to ACTIVATE bank B command delay same bank group | | $t_{RRDL}$ | max(5ns, 4tCK) | - | max(5ns, 4tCK) | - | ns | 31 |
| ACTIVATE bank A to ACTIVATE bank B command delay different bank groups | | $t_{RRDS}$ | max(4ns, 4tCK) | - | max(4ns, 4tCK) | - | ns | 32 |
| Four bank ACTIVATE window | | $t_{FAW}$ | 20 | - | 20 | - | ns | 33 |
| REFpb / REFp2b to ACTIVATE or REFpb / REFp2b command delay | | $t_{RREFD}$ | max(10ns, 10tCK) | - | max(10ns, 10tCK) | - | ns | 34 |
| READ to PREab / PREpb command delay same bank with bank groups enabled | | $t_{RTPL}$ | 3 | - | 3 | - | $t_{CK}$ | 35 |
| READ to PREab / PREpb command delay same bank with bank groups disabled | | $t_{RTPS}$ | 2 | - | 2 | - | $t_{CK}$ | 36 |
| PREpb to PREab / PREpb command delay | | $t_{PPD}$ | 1tCK | - | 1tCK | - | ns | |
| PREab / PREpb command period | | $t_{RP}$ | 15 | - | 15 | - | ns | |
| WRITE recovery time | | $t_{WR}$ | 15 | - | 15 | - | ns | |
| **Column Access Timings** | | | | | | | | |

**Table 69 — AC Timings**

| PARAMETER 1, 2 | SYMBOL | 14Gbps @1.35V | | 12Gbps @1.35V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |
| RD/WR bank A to RD/WR bank B command delay same bank group | $t_{CCDL}$ | 3 | - | 3 | - | $t_{CK}$ | 31, 37 |
| RD/WR bank A to RD/WR bank B command delay different bank groups | $t_{CCDS}$ | 2 | - | 2 | - | $t_{CK}$ | 32, 38 |
| WDM/WSM bank A to WDM/WSM/WOM bank A command delay same bank | $t_{CCDMW}$ | 2 | - | 2 | - | $t_{CK}$ | |
| LDFF to LDFF command cycle time | $t_{LTLTR}$ | 4 | - | 4 | - | $t_{CK}$ | |
| LDFF(1111) to RDTR command delay | $t_{LTRTR}$ | 4 | - | 4 | - | $t_{CK}$ | |
| READ or RDTR to LDFF command delay | $t_{RDTLT}$ | RL+CRCRL+4tCK | - | RL+CRCRL+4tCK | - | $t_{CK}$ | |
| WRITE to LDFF command delay | $t_{WRTLT}$ | WL+5tCK+CRCWL | - | WL+5tCK+CRCWL | - | $t_{CK}$ | |
| WRTR to RDTR command delay | $t_{WTRTR}$ | WL+2tCK+5ns | - | WL+2tCK+5ns | - | $t_{CK}$ | |
| WRITE to WRTR command delay | $t_{WRWTR}$ | WL+CRCWL+2tCK | - | WL+CRCWL+2tCK | - | $t_{CK}$ | |
| Internal WRITE to READ command delay same bank group | $t_{WTRL}$ | 2tCK+5ns | - | 2tCK+5ns | - | ns | 31 |
| Internal WRITE to READ command delay different bank groups | $t_{WTRS}$ | 2tCK+5ns | - | 2tCK+5ns | - | ns | 32 |
| READ or RDTR to WRITE or WRTR command delay | $t_{RTW}$ | RL+BL/8+2-WL+CRCRL | - | RL+BL/8+2-WL+CRCRL | - | ns | 40 |
| **Power-Down and Refresh Timings** | | | | | | | |
| Self Refresh entry to exit time | $t_{CKESR}$ | | - | | - | ns | |
| Hibernate Self Refresh entry to exit time | $t_{HSRF}$ | Max (10ns, 10tCK) | - | Max (10ns, 10tCK) | - | ns | |
| Valid CK clocks required after SELF REFRESH ENTRY | $t_{CKSRE}$ | 10 | - | 10 | - | $t_{CK}$ | |
| Valid CK clocks required before SEF REFRESH EXIT | $t_{CKSRX}$ | 10 | - | 10 | - | $t_{CK}$ | |
| READ to SELF REFRESH ENTRY or POWER-DOWN ENTRY command delay | $t_{RDSRE}$ | RL+CRCRL+4tCK | - | RL+CRCRL+4tCK | - | $t_{CK}$ | 41 |
| WRITE to SELF REFRESH ENTRY or POWER-DOWN ENTRY command delay | $t_{WRSRE}$ | WL+BL/8+1tCK+Max(tRP+tWR, CRCRWL+tCK) | - | WL+BL/8+1tCK+Max(tRP+tWR, CRCRWL+tCK) | - | $t_{CK}$ | 42 |
| ACTIVATE to POWER-DOWN ENTRY command delay | $t_{ACTPDE}$ | 2 | - | 2 | - | $t_{CK}$ | |
| PREab / PREpb to POWER-DOWN ENTRY command delay | $t_{PREPDE}$ | 2 | - | 2 | - | $t_{CK}$ | |
| REFab / REFpb / REFp2b to POWER-DOWN ENTRY command delay | $t_{REFPDE}$ | max(4ns, 4tCK) | - | max(4ns, 4tCK) | - | $t_{CK}$ | |
| MRS to POWER-DOWN ENTRY command delay | $t_{MRSPDE}$ | tMOD(min) | - | tMOD(min) | - | $t_{CK}$ | |

## Table 69 — AC Timings

| PARAMETER 1, 2 | SYMBOL | 14Gbps @1.35V | | 12Gbps @1.35V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |
| REFab command period | $t_{RFCab}$ | 120 | - | 120 | - | ns | |
| REFpb / REFp2b command period | $t_{RFCpb}$ | 120 | - | 120 | - | ns | |
| SELF REFRESH EXIT to any command delay | $t_{XS}$ | 120 | - | 120 | - | $t_{CK}$ | 57 |
| CKE_n LOW pulse width to exit from Hibernate Self Refresh mode | $t_{XHP}$ | 10 | - | 10 | - | ns | |
| Exit Hibernate Self Refresh to SELF REFRESH EXIT command delay | $t_{XSH}$ | 500 | - | 500 | - | µs | |
| REFab / REFpb / REFp2b period | $t_{REF}$ | - | 32 | - | 32 | ms | |
| Average periodic refresh interval with REFab command | $t_{REFI}$ | - | 1.9 | - | 1.9 | µs | 44 |
| Average periodic refresh interval with REFpb / REFp2b command | $t_{REFIPB}$ | - | tREFI / 16 or tREFI / 8 | - | tREFI / 16 or tREFI / 8 | us | |
| REFab to REFab command period required for impedance calibration updates | $t_{ABREF}$ | 1 | - | 1 | - | ms | 45 |
| Power-Down entry to exit time | $t_{PD}$ | | 9 * tREFI | | 9 * tREFI | ns | |
| NOP commands required upon POWER-DOWN and SELF REFRESH ENTRY | $t_{CPDED}$ | 10 | - | 10 | - | $t_{CK}$ | 55 |
| POWER-DOWN exit time | $t_{XP}$ | max(10ns, 5tCK) | - | max(10ns, 5tCK) | - | $t_{CK}$ | 57 |
| Hibernate self refresh entry to $V_{DDQ}$ power-off ramp delay | $t_{VOFF}$ | 100 | | 100 | | ns | |
| Minimum $V_{DDQ}$ off duration in hibernate mode | $t_{VON}$ | 1 | - | 1 | - | ms | |
| $V_{DDQ}$ stable to CKE_n stable HIGH delay | $t_{CTEN}$ | - | 1 | - | 1 | ms | |
| $V_{DDQ}$ stable to exit from hibernate mode delay | $t_{HEX}$ | 2 | - | 2 | - | ms | |
| **Miscellaneous Timings** | | | | | | | |
| MODE REGISTER SET command period | $t_{MRD}$ | max(4ns, 4tCK) | - | max(4ns, 4tCK) | - | $t_{CK}$ | 46 |
| MODE REGISTER SET command update delay | $t_{MOD}$ | max(8ns, 8tCK) | - | max(8ns, 8tCK) | - | $t_{CK}$ | 46 |
| Required time for duty cycle corrector (DCC) | $t_{DCC}$ | 2048 | | 2048 | | $t_{CK}$ or ns | 48 |
| DVS voltage stabilization time | $t_{VS}$ | 50 | - | 50 | - | µs | |
| REFab to calibration update complete delay | $t_{KO}$ | - | 1 * tCK + 12 | - | 1 * tCK + 12 | ns | |
| $V_{REFC}$ to $V_{REFC2}$ reference voltage settling time | $t_{VREFC2}$ | - | 1000 | - | 1000 | ns | 49 |
| $V_{REFC}$ reference voltage level change settling time | $t_{VREFC}$ | - | 300 | - | 300 | ns | 50 |
| $V_{REFD}$ reference voltage level change settling time | $t_{VREFD}$ | - | 300 | - | 300 | ns | 51 |
| $V_{REFD}$ to $V_{REFD2}$ reference voltage settling time | $t_{VREFD2}$ | - | 1000 | - | 1000 | ns | 52 |

**Table 69 — AC Timings**

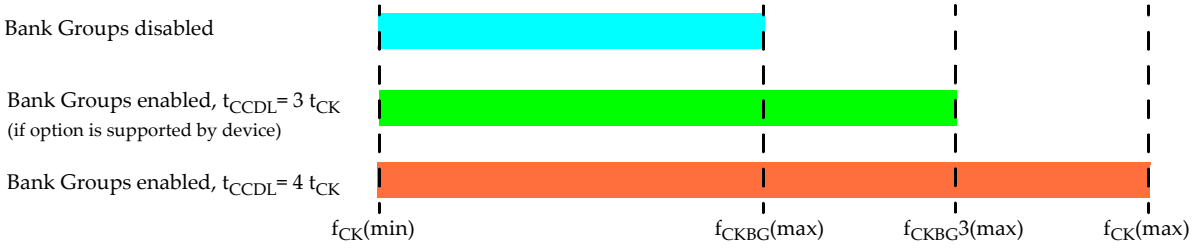
| PARAMETER 1, 2 | SYMBOL | 14Gbps @1.35V | | 12Gbps @1.35V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |
| CA[6:0] and EDC setup time before RESET_n de-assertion | $t_{ATS}$ | 10 | - | 10 | - | ns | |
| CA[6:0] and EDC hold time after RESET_n de-assertion | $t_{ATH}$ | 10 | - | 10 | - | ns | |
| CAT command to data out delay | $t_{ADR}$ | - | 1 * tCK + 10 | - | 1 * tCK + 10 | ns | |
| CADT off to DQ/DBI_n in ODT state delay | $t_{ADZ}$ | - | 1 * tCK + 10 | - | 1 * tCK + 10 | ns | |
| Vendor ID on | $t_{WRIDON}$ | | 1 * tCK + 11 | | 1 * tCK + 11 | ns | |
| Vendor ID off | $t_{WRIDOFF}$ | | 1 * tCK + 11 | | 1 * tCK + 11 | ns | |

## Table 69 — AC Timings

| PARAMETER 1, 2 | SYMBOL | 14Gbps @1.35V | | 12Gbps @1.35V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |

Notes:

1. All parameters assume proper device initialization.
2. Tests for AC timing may be conducted at nominal supply voltage levels, but the related specifications and device operation are guaranteed for the full voltage and temperature range specified.
3. Parameter $f_{CKBG3}$ is required for those devices supporting both 3 * $t_{CK}$ and 4 * $t_{CK}$ settings for bank groups. Devices supporting only 3 * $t_{CK}$ or 4 * $t_{CK}$ need only to specify $f_{CKBG}$.
4. Bank Group Frequency ranges




5. Parameter $f_{CKPIN}$ applies when the alignment point in MR7, OP0 is set to "at balls", the phase difference between the WCK and CK clocks at the DRAM balls is within $t_{WCK2CKSYNC}$ or $t_{WCK2CK}$ for pin mode, and no phase search in WCK2CK training is performed.
6. Parameter $f_{CKVREFC2}$ applies when the CA input reference voltage in MR7, OP6 (Half VREFC) is set to VREFC2.
7. Parameter $f_{CKVREFD2}$ applies when the data input reference voltage in MR7, OP7 (Half VREFD) is set to VREFD2
8. Parameter $f_{CKAUTOSYNC}$ applies when WCK2CK Auto Synchronization is enabled in MR7, OP4.
9. Parameter $f_{CKLF}$ applies when Low Frequency Mode is enabled in MR7, OP3. Low Frequency Mode is optional.
10. By definition the nominal WCK clock cycle time always is 1/2 or 1/4 of the CK clock cycle time depending on the selected WCK Ratio (not including jitter).
11. The phase relationship between WCK_t/WCK_c and CK_t/CK_c clocks must meet the $t_{WCK2CK}$ specification.
12. CA input timings are referenced to $V_{REFC}$.
13. CA and CKE_n input pulse widths are design targets. The values will be characterized but not tested on each device.
14. CA input timings are only valid with CABI_n being enabled and a maximum of 5 CA inputs driven LOW in 2 channel mode or 7 CA inputs driven LOW in PC mode.
15. Parameter may be specified as a combination of $t_{CK}$ and ns.
16. Parameters $t_{WCKHTR}$ and $t_{WCKLTR}$ specify the max. allowed WCK clock-to-clock phase shift during WCK2CK training. For READ and WRITE bursts use $t_{WCKH}$ and $t_{WCKL}$.
17. Parameter $t_{WCK2CKPIN}$ defines the WCK2CK phase offset range at the CK and WCK balls for ideal (phase = 0°) clock alignment at the device's phase detector (when the alignment point in MR7, OP0 is set to "at phase detector"), or at the WCK and CK balls (when the alignment point in MR7, OP0 is set to "at pins"). The minimum and maximum values could be negative or positive numbers, depending on the selected WCK2CK alignment point, PLL-on or PLL-off mode and design implementation.
18. Parameter $t_{WCK2CKSYNC}$ defines the max. phase offset from the ideal (phase = 0°) clock alignment at the device's phase detector (when the alignment point in MR7, OP0 is set to "at phase detector"), or at the WCK and CK balls (when the alignment point in MR7, OP0 is set to "at pins"), where the internal logic synchronizes the CK and WCK clocks; it is expected to be a fraction of $t_{WCK2CK}$.
19. Parameter $t_{WCK2CK}$ defines the max. phase offset from the ideal (phase = 0°) clock alignment at the device's phase detector (when the alignment point in MR7, OP0 is set to "at phase detector"), or at the WCK and CK balls (when the alignment point in MR7, OP0 is set to "at pins"), for stable device operation.
20. Parameter $t_{WCK2DQI}$ defines the WCK to DQ/DBI_n time delay range for WRITEs for PLL-on and PLL-off mode. The minimum and maximum values could be negative or positive numbers, depending on design implementation and PLL-on or PLL-off mode. They also vary across PVT. Data training is required to determine the actual $t_{WCK2DQI}$ value for reliable WRITE operation.
21. Parameter $t_{WCK2DQO}$ defines the WCK to DQ/DBI_n/EDC time delay range for READs for PLL-on and PLL-off mode. The minimum and maximum values could be negative or positive numbers, depending on design implementation and PLL-on or PLL-off mode. They also vary across PVT. Data training is required to determine the actual $t_{WCK2DQO}$ value for reliable READ operation.
22. Outputs measured with equivalent load (vendor specific) terminated to $V_{DDQ}$.
23. DQ/DBI_n input timings are valid only with DBI being enabled and a maximum of 4 data inputs per byte driven LOW.
24. Data input slew rate must be greater than or equal to TBD V/ns. The slew rate is measured between $V_{REFD}$ crossing and $V_{IHD(AC)}$ or $V_{ILD(AC)}$.
25. The data input pulse width, $t_{DIPW}$, defines the minimum positive or negative input pulse width for any one worst-case channel required for proper propagation of an external signal to the receiver. $t_{DIPW}$ is independent of the PLL/DLL mode. In general $t_{DIPW}$ is larger than $t_{DIVW}$.

## Table 69 — AC Timings

| PARAMETER 1, 2 | SYMBOL | 14Gbps @1.35V | | 12Gbps @1.35V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |

26. The data input valid window, $t_{DIVW}$, defines the time region where input data must be valid for reliable data capture at the receiver for any one worst case channel. It accounts for jitter between data and clock at the latching point introduced in the path between the DRAM pads and the latching point. Any additional jitter introduced into the source signals (e.g., within the system before the DRAM pad) must be accounted for in the final timing budget together with the chosen PLL/DLL mode and bandwidth. $t_{DIVW}$ is defined for PLL/DLL off and on mode separately. In the case of PLL/DLL on, $t_{DIVW}$ must be specified for each supported bandwidth. In general $t_{DIVW}$ is smaller than $t_{DIPW}$.
27. $t_{DQDQI}$ defines the maximum skew among all DQ/DBI_n inputs of a byte under worst case conditions. Parameter $t_{WCK2DQI}$ defines the mean value of the earliest and latest DQ/DBI_n pin, $t_{DQDQI}$(min) the negative offset to $t_{WCK2DQI}$ for the earliest DQ/DBI_n pin and $t_{DQDQI}$(max) the positive offset to $t_{WCK2DQI}$ for the latest DQ/DBI_n pin.
28. $t_{DQDQO}$ defines the maximum skew among all DQ/DBI_n/EDC outputs of a byte under worst case conditions. Parameter $t_{WCK2DQO}$ defines the mean value of the earliest and latest DQ/DBI_n/EDC pin, $t_{DQDQO}$(min) the negative offset to $t_{WCK2DQO}$ for the earliest DQ/DBI_n/EDC pin and $t_{DQDQO}$(max) the positive offset to $t_{WCK2DQO}$ for the latest DQ/DBI_n/EDC pin.
29. For READs and WRITEs with AUTO PRECHARGE enabled the device will hold off the internal PRECHARGE until $t_{RAS}$(min) has been satisfied.
30. Parameter applies when bank groups are enabled and consecutive commands access the same bank group.
31. Parameter applies when bank groups are disabled or consecutive commands access different bank groups.
32. Not more than 4 ACTIVATE commands are allowed within period.
33. Parameter applies between any two REFpb / REFp2b commands and between a REFpb / REFp2b command and a subsequent ACTIVATE command to a different bank.
34. Parameter applies when bank groups are enabled and READ and PRECHARGE commands access the same bank.
35. Parameter applies when bank groups are disabled or READ and PRECHARGE commands access the same bank.
36. $t_{CCDL}$ is either for gapless consecutive READ or gapless consecutive WRITE commands.
37. $t_{CCDS}$ is either for gapless consecutive READ or RDTR (any combination), gapless consecutive WRITE, or gapless consecutive WRTR commands.
38. $t_{RTW}$ is not a device limit but determined by the system bus turnaround time. The difference between $t_{WCK2DQO}$ and $t_{WCK2DQI}$ shall be considered in the calculation of the bus turnaround time.
39. Read data including CRC data must have been clocked out before entering self refresh or power-down mode.
40. Write data must have been written to the memory core, and CRC data must have been clocked out before entering self refresh or power-down mode.
41. Time for WCK2CK training and data training not included.
42. A maximum of 8 consecutive REFab commands can be posted to a device, meaning that the maximum absolute interval between any REFab command and the next REFab command is 9 * $t_{REFI}$.
43. $t_{ABREF}$ is relevant only when refresh is normally performed using the REFpb / REFp2b command. REFab commands must be issued at a minimum rate of $t_{ABREF}$ to allow impedance updates from the auto-calibration engine to occur.
44. $t_{MRD}$ is less than or equal to $t_{MOD}$. See MODE REGISTER SET command section for more details on $t_{MOD}$ and $t_{MRD}$
45. Replaces parameter $t_{LK}$ when PLL Fast Lock has been enabled prior to the PLL/DLL enable or reset.
46. The parameter may be specified in $t_{CK}$ or ns (vendor specific)
47. The parameter applies when the CA reference voltage selection in MR7, OP6 (Half VREFC) has changed.
48. The parameter applies when the $V_{REFC}$ level has been changed in MR10. $t_{VREFC}$ is a constant value for the device, and is referenced from the MRS command to when the 90% level of the delta between old and new $V_{REFC}$ voltage has been reached.
49. The parameter applies when the $V_{REFD}$ level has been changed in MR6 or MR9. $t_{VREFD}$ is a constant value for the device, and is referenced from the MRS command to when the 90% level of the delta between old and new $V_{REFD}$ voltage has been reached.
50. The parameter applies when the data reference voltage selection in MR7, OP7 (Half VREFD) has changed.
51. The parameter applies to WCK per byte or WCK per word depending on device implementation.
52. This parameter is for DRAM design only and valid on the silicon die at the input of the receiver. It is not intended to be measured.
53. After issuing PDE or SRE command, NOP (1) commands and CABI_n = HIGH (if CABI is enabled) are required for a minimum of $t_{CPDED}$ after POWER-DOWN or SELF REFRESH entry.
54. This parameter is for DRAM design only and valid on the silicon die at the input of the receiver. It is not intended to be measured.
55. After issuing PDE or SRE command, NOP (1) commands and CABI_n = HIGH (if CABI is enabled) are required for a minimum of tCPDED after POWER-DOWN or SELFREFRESH entry.
56. the scope of tCKEPW is limited to CAT commands in Command Address Training. Refer to parameters tPD and tXP for the minimum CKE_n pin HIGH and LOW pulse width states in conjunction with power-down, and parameters tCKESR, tXS, tHSRF, tXHP for the minimum CKE_n pin HIGH and LOW pulse width states in conjunction with self refresh and hibernate self refresh
57. After issuing PDX or SRX command, NOP (1) commands and CABI_n = HIGH (if CABI is enabled) are required for a minimum of tXP after POWER-DOWN EXIT or tXS after SELF REFRESH EXIT

## 8.9 AC TIMINGS

**Table 70 — AC Timings**

| PARAMETER 1, 2 | | | SYMBOL | 10Gbps @1.25V | | 12Gbps @1.25V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|---|---|
| | | | | MIN | MAX | MIN | MAX | | |
| **CK_t and WCK_t Timings** | | | | | | | | | |
| CK clock cycle time | | PLL off | $t_{CK}$ | 0.8 | 20 | 0.67 | 20 | ns | |
| CK clock HIGH-level width | | | $t_{CH}$ | 0.47 | 0.53 | 0.47 | 0.53 | $t_{CK}$ | |
| CK clock LOW-level width | | | $t_{CL}$ | 0.47 | 0.53 | 0.47 | 0.53 | $t_{CK}$ | |
| Min. CK clock half period | | | $t_{HP}$ | min(tCH,tCL) | - | min(tCH,tCL) | - | $t_{CK}$ | |
| CK clock frequency | DDR WCK | PLL off | $f_{CK}$ | 50 | | 50 | | MHz | |
| CK clock frequency with bank groups disabled | | | $f_{CKBG}$ | fCK(min) | 1750 | fCK(min) | 1750 | MHz | 3, 4 |
| CK clock frequency with bank groups enabled and $t_{CCDL}$ = 3 $t_{CK}$ | | | $f_{CKBG3}$ | fCK(min) | 1750 | fCK(min) | 1750 | MHz | 3 |
| CK clock frequency with WCK2CK alignment at the balls | | | $f_{CKPIN}$ | fCK(min) | 500 | fCK(min) | 500 | MHz | 5 |
| CK clock frequency in RDQS Mode | | | $f_{CKRDQS}$ | fCK(min) | 500 | fCK(min) | 500 | MHz | 6 |
| CK clock frequency for operation with $V_{REFC2}$ | | | $f_{CKVREFC2}$ | fCK(min) | | fCK(min) | | MHz | 6 |
| CK clock frequency for operation with $V_{REFD2}$ | | | $f_{CKVREFD2}$ | fCK(min) | | fCK(min) | | MHz | 7 |
| CK clock frequency for WCK-to-CK auto synchronization in WCK2CK training mode | | | $f_{CKAUTOSYNC}$ | fCK(min) | 500 | fCK(min) | 500 | MHz | 8 |
| CK clock frequency for device operation with Low Frequency mode enabled | | | $f_{CKLF}$ | fCK(min) | 500 | fCK(min) | 500 | MHz | 10 |
| WCK clock cycle time | DDR WCK | PLL off | $t_{WCK}$ | 0.143 | 5 | 0.143 | 5 | ns | 11 |
| WCK clock HIGH-level width | | | $t_{WCKH}$ | 0.47 | 0.53 | 0.47 | 0.53 | $t_{WCK}$ | 12 |
| WCK clock LOW-level width | | | $t_{WCKL}$ | 0.47 | 0.53 | 0.47 | 0.53 | $t_{WCK}$ | 12 |
| Min. WCK clock half period | | | $t_{WCKHP}$ | min(tWCKH, tWCKL) | - | min(tWCKH, tWCKL) | - | $t_{WCK}$ | |
| **CKE_n and CA Input Timings** | | | | | | | | | |
| Command Address (CA) input setup time | | | $t_{AS}$ | -0.09 | - | -0.09 | - | ns | 13, 15, 54 |
| Command Address (CA) input hold time | | | $t_{AH}$ | 0.09 | - | 0.09 | - | ns | 13, 15, 54 |
| Command Address (CA) input pulse width | | | $t_{APW}$ | 0.26 | - | 0.26 | - | ns | 13, 14, 15, 54 |
| CKE_n input setup time | | | $t_{CKES}$ | | - | | - | ns | 15, 54 |

## Table 70 — AC Timings

| PARAMETER 1, 2 | | | SYMBOL | 10Gbps @1.25V | | 12Gbps @1.25V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|---|---|
| | | | | MIN | MAX | MIN | MAX | | |
| CKE_n input hold time | | | $t_{CKEH}$ | | - | | - | ns | 13, 54 |
| CKE_n min. HIGH and LOW pulse width for SELF REFRESH, HIBERNATE SELF REFRESH and POWER-DOWN commands | | | $t_{CKE}$ | 10 | - | 10 | - | ns | 13, 14, 54 |
| CKE_n min. HIGH and LOW pulse width for CAT command | | | $t_{CKEPW}$ | | - | | - | ns | |
| **WCK2CK Timings** | | | | | | | | | |
| WCK stop to MRS delay for entering WCK2CK training | | | $t_{WCK2MRS}$ | 3 | - | 3 | - | ns | |
| MRS to WCK restart delay after entering WCK2CK training | | | $t_{MRSTWCK}$ | 10 | - | 10 | - | ns | 16 |
| WCK clock frequency for WCK2CK training with WCK stop | | | $f_{WCKSTOP}$ | - | | - | | MHz | |
| WCK start to WCK phase movement delay | | | $t_{WCK2TR}$ | 10 | - | 10 | - | $t_{CK}$ | |
| WCK phase change to phase detector out delay | | | $t_{WCK2PH}$ | 5 | - | 5 | - | ns | |
| WCK clock HIGH-level width during WCK2CK training | | | $t_{WCKHTR}$ | 0.43 | 0.57 | 0.43 | 0.57 | $t_{WCK}$ | 12, 17 |
| WCK clock LOW-level width during WCK2CK training | | | $t_{WCKLTR}$ | 0.43 | 0.57 | 0.43 | 0.57 | $t_{WCK}$ | 12, 17 |
| WCK2CK offset when zero offset at phase detector or at balls | PLL off; MR7 OP0=0 (at phase detector) | | $t_{WCK2CKPIN}$ | -0.2 | 0.2 | -0.2 | 0.2 | ns | 18 |
| | PLL off; MR7 OP0=1 (at balls) | | | -0.2 | 0.2 | -0.2 | 0.2 | | |
| WCK2CK phase offset upon WCK2CK training exit | DDR WCK | MR7 OP0=0 (at phase detector) | $t_{WCK2CKSYNC}$ | -0.25 | 0.25 | -0.25 | 0.25 | $t_{CK}$ | 19 |
| | | MR7 OP0=1 (at balls) | | -0.143 | 0.143 | -0.143 | 0.143 | ns | |
| WCK2CK phase offset | DDR WCK | MR7 OP0=0 (at phase detector) | $t_{WCK2CK}$ | -0.4 | 0.4 | -0.4 | 0.4 | $t_{CK}$ | 20 |
| | | MR7 OP0=1 (at balls) | | -0.228 | 0.228 | -0.228 | 0.228 | ns | |
| **Data Input and Output Timings** | | | | | | | | | |

## Table 70 — AC Timings

| PARAMETER 1, 2 | | SYMBOL | 10Gbps @1.25V | | 12Gbps @1.25V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | | |
| WCK to DQ/DBI_n offset for input data | PLL off | $t_{WCK2DQI}$ | 0.3 | 1.3 | 0.3 | 1.3 | ns | 21, 53 |
| WCK to DQ/DBI_n/EDC offset for output data | PLL off | $t_{WCK2DQO}$ | 0.5 | 1.5 | 0.5 | 1.5 | ns | 22, 23, 53 |
| DQ/DBI_n input pulse width | | $t_{DIPW}$ | | - | | - | ns | 24, 25, 26, 54 |
| DQ/DBI_n data input valid window | PLL off | $t_{DIVW}$ | | - | | - | ns | 24, 25, 27, 54 |
| DQ/DBI_n input skew | | $t_{DQDQI}$ | -0.1 | 0.1 | -0.1 | 1.0 | ns | 28, 53 |
| DQ/DBI_n/EDC output skew | | $t_{DQDQO}$ | -0.125 | 0.125 | -0.125 | 0.125 | ns | 19, 53 |
| **Row Access Timings** | | | | | | | | |
| ACTIVATE to ACTIVATE command period | | $t_{RC}$ | 47 | - | 47 | - | ns | |
| ACTIVATE to PREab / PREpb command period | | $t_{RAS}$ | 32 | 9*tREFI | 32 | 9*tREFI | ns | 30 |
| ACTIVATE to READ command delay | | $t_{RCDRD}$ | 15 | - | 15 | - | ns | |
| ACTIVATE to WRITE command delay | | $t_{RCDWR}$ | 12 | - | 12 | - | ns | |
| ACTIVATE to RDTR command delay | | $t_{RCDRTR}$ | 18 | - | 18 | - | ns | |
| ACTIVATE to WRTR command delay | | $t_{RCDWTR}$ | 10 | - | 10 | - | ns | |
| ACTIVATE to LDFF command delay | | $t_{RCDLTR}$ | 10 | - | 10 | - | ns | |
| REFab to RDTR or WRTR command delay | | $t_{REFTR}$ | 19 | - | 19 | - | ns | |
| ACTIVATE bank A to ACTIVATE bank B command delay same bank group | | $t_{RRDL}$ | max(5ns, 4tCK) | - | max(5ns, 4tCK) | - | ns | 31 |
| ACTIVATE bank A to ACTIVATE bank B command delay different bank groups | | $t_{RRDS}$ | max(4ns, 4tCK) | - | max(4ns, 4tCK) | - | ns | 32 |
| Four bank ACTIVATE window | | $t_{FAW}$ | 20 | - | 20 | - | ns | 33 |
| REFpb / REFp2b to ACTIVATE or REFpb / REFp2b command delay | | $t_{RREFD}$ | max(10ns, 10tCK) | - | max(10ns, 10tCK) | - | ns | 34 |
| READ to PREab / PREpb command delay same bank with bank groups enabled | | $t_{RTPL}$ | 3 | - | 3 | - | $t_{CK}$ | 35 |
| READ to PREab / PREpb command delay same bank with bank groups disabled | | $t_{RTPS}$ | 2 | - | 2 | - | $t_{CK}$ | 36 |
| PREpb to PREab / PREpb command delay | | $t_{PPD}$ | 1tCK | - | 1tCK | - | ns | |
| PREab / PREpb command period | | $t_{RP}$ | 15 | - | 15 | - | ns | |
| WRITE recovery time | | $t_{WR}$ | 15 | - | 15 | - | ns | |
| **Column Access Timings** | | | | | | | | |

**Table 70 — AC Timings**

| PARAMETER 1, 2 | SYMBOL | 10Gbps @1.25V | | 12Gbps @1.25V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |
| RD/WR bank A to RD/WR bank B command delay same bank group | $t_{CCDL}$ | 3 | - | 3 | - | $t_{CK}$ | 31, 37 |
| RD/WR bank A to RD/WR bank B command delay different bank groups | $t_{CCDS}$ | 2 | - | 2 | - | $t_{CK}$ | 32, 38 |
| WDM/WSM bank A to WDM/WSM/WOM bank A command delay same bank | $t_{CCDMW}$ | 2 | - | 2 | - | $t_{CK}$ | |
| LDFF to LDFF command cycle time | $t_{LTLTR}$ | 4 | - | 4 | - | $t_{CK}$ | |
| LDFF(1111) to RDTR command delay | $t_{LTRTR}$ | 4 | - | 4 | - | $t_{CK}$ | |
| READ or RDTR to LDFF command delay | $t_{RDTLT}$ | RL+CRCRL+4tCK | - | RL+CRCRL+4tCK | - | $t_{CK}$ | |
| WRITE to LDFF command delay | $t_{WRTLT}$ | WL+5tCK+CRCWL | - | WL+5tCK+CRCWL | - | $t_{CK}$ | |
| WRTR to RDTR command delay | $t_{WTRTR}$ | WL+2tCK+5ns | - | WL+2tCK+5ns | - | $t_{CK}$ | |
| WRITE to WRTR command delay | $t_{WRWTR}$ | WL+CRCWL+2tCK | - | WL+CRCWL+2tCK | - | $t_{CK}$ | |
| Internal WRITE to READ command delay same bank group | $t_{WTRL}$ | 2tCK+5ns | - | 2tCK+5ns | - | ns | 31 |
| Internal WRITE to READ command delay different bank groups | $t_{WTRS}$ | 2tCK+5ns | - | 2tCK+5ns | - | ns | 32 |
| READ or RDTR to WRITE or WRTR command delay | $t_{RTW}$ | RL+BL/8+2-WL+CRCRL | - | RL+BL/8+2-WL+CRCRL | - | ns | 40 |
| **Power-Down and Refresh Timings** | | | | | | | |
| Self Refresh entry to exit time | $t_{CKESR}$ | | - | | - | ns | |
| Hibernate Self Refresh entry to exit time | $t_{HSRF}$ | Max (10ns, 10tCK) | - | Max (10ns, 10tCK) | - | ns | |
| Valid CK clocks required after SELF REFRESH ENTRY | $t_{CKSRE}$ | 10 | - | 10 | - | $t_{CK}$ | |
| Valid CK clocks required before SEF REFRESH EXIT | $t_{CKSRX}$ | 10 | - | 10 | - | $t_{CK}$ | |
| READ to SELF REFRESH ENTRY or POWER-DOWN ENTRY command delay | $t_{RDSRE}$ | RL+CRCRL+4tCK | - | RL+CRCRL+4tCK | - | $t_{CK}$ | 41 |
| WRITE to SELF REFRESH ENTRY or POWER-DOWN ENTRY command delay | $t_{WRSRE}$ | WL+BL/8+1tCK+Max(tRP+tWR, CRCRWL+tCK) | - | WL+BL/8+1tCK+Max(tRP+tWR, CRCRWL+tCK) | - | $t_{CK}$ | 42 |
| ACTIVATE to POWER-DOWN ENTRY command delay | $t_{ACTPDE}$ | 2 | - | 2 | - | $t_{CK}$ | |
| PREab / PREpb to POWER-DOWN ENTRY command delay | $t_{PREPDE}$ | 2 | - | 2 | - | $t_{CK}$ | |
| REFab / REFpb / REFp2b to POWER-DOWN ENTRY command delay | $t_{REFPDE}$ | max(4ns, 4tCK) | - | max(4ns, 4tCK) | - | $t_{CK}$ | |
| MRS to POWER-DOWN ENTRY command delay | $t_{MRSPDE}$ | tMOD(min) | - | tMOD(min) | - | $t_{CK}$ | |

## Table 70 — AC Timings

| PARAMETER 1, 2 | SYMBOL | 10Gbps @1.25V | | 12Gbps @1.25V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |
| REFab command period | $t_{RFCab}$ | 120 | - | 120 | - | ns | |
| REFpb / REFp2b command period | $t_{RFCpb}$ | 120 | - | 120 | - | ns | |
| SELF REFRESH EXIT to any command delay | $t_{XS}$ | 120 | - | 120 | - | $t_{CK}$ | 57 |
| CKE_n LOW pulse width to exit from Hibernate Self Refresh mode | $t_{XHP}$ | 10 | - | 10 | - | ns | |
| Exit Hibernate Self Refresh to SELF REFRESH EXIT command delay | $t_{XSH}$ | 500 | - | 500 | - | µs | |
| REFab / REFpb / REFp2b period | $t_{REF}$ | - | 32 | - | 32 | ms | |
| Average periodic refresh interval with REFab command | $t_{REFI}$ | - | 1.9 | - | 1.9 | µs | 44 |
| Average periodic refresh interval with REFpb / REFp2b command | $t_{REFIPB}$ | - | tREFI / 16 or tREFI / 8 | - | tREFI / 16 or tREFI / 8 | us | |
| REFab to REFab command period required for impedance calibration updates | $t_{ABREF}$ | 1 | - | 1 | - | ms | 45 |
| Power-Down entry to exit time | $t_{PD}$ | | 9 * tREFI | | 9 * tREFI | ns | |
| NOP commands required upon POWER-DOWN and SELF REFRESH ENTRY | $t_{CPDED}$ | 10 | - | 10 | - | $t_{CK}$ | 55 |
| POWER-DOWN exit time | $t_{XP}$ | max(10ns, 5tCK) | - | max(10ns, 5tCK) | - | $t_{CK}$ | 57 |
| Hibernate self refresh entry to $V_{DDQ}$ power-off ramp delay | $t_{VOFF}$ | 100 | | 100 | | ns | |
| Minimum $V_{DDQ}$ off duration in hibernate mode | $t_{VON}$ | 1 | - | 1 | - | ms | |
| $V_{DDQ}$ stable to CKE_n stable HIGH delay | $t_{CTEN}$ | - | 1 | - | 1 | ms | |
| $V_{DDQ}$ stable to exit from hibernate mode delay | $t_{HEX}$ | 2 | - | 2 | - | ms | |
| **Miscellaneous Timings** | | | | | | | |
| MODE REGISTER SET command period | $t_{MRD}$ | max(4ns, 4tCK) | - | max(4ns, 4tCK) | - | $t_{CK}$ | 46 |
| MODE REGISTER SET command update delay | $t_{MOD}$ | max(8ns, 8tCK) | - | max(8ns, 8tCK) | - | $t_{CK}$ | 46 |
| Required time for duty cycle corrector (DCC) | $t_{DCC}$ | 2048 | | 2048 | | $t_{CK}$ or ns | 48 |
| DVS voltage stabilization time | $t_{VS}$ | 50 | - | 50 | - | µs | |
| REFab to calibration update complete delay | $t_{KO}$ | - | 1 * tCK + 12 | - | 1 * tCK + 12 | ns | |
| $V_{REFC}$ to $V_{REFC2}$ reference voltage settling time | $t_{VREFC2}$ | - | 1000 | - | 1000 | ns | 49 |
| $V_{REFC}$ reference voltage level change settling time | $t_{VREFC}$ | - | 300 | - | 300 | ns | 50 |
| $V_{REFD}$ reference voltage level change settling time | $t_{VREFD}$ | - | 300 | - | 300 | ns | 51 |
| $V_{REFD}$ to $V_{REFD2}$ reference voltage settling time | $t_{VREFD2}$ | - | 1000 | - | 1000 | ns | 52 |

**Table 70 — AC Timings**

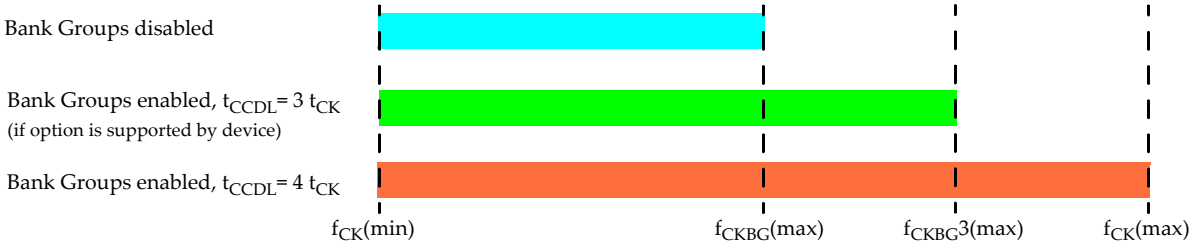
| PARAMETER 1, 2 | SYMBOL | 10Gbps @1.25V | | 12Gbps @1.25V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |
| CA[6:0] and EDC setup time before RESET_n de-assertion | $t_{ATS}$ | 10 | - | 10 | - | ns | |
| CA[6:0] and EDC hold time after RESET_n de-assertion | $t_{ATH}$ | 10 | - | 10 | - | ns | |
| CAT command to data out delay | $t_{ADR}$ | - | 1 * tCK + 10 | - | 1 * tCK + 10 | ns | |
| CADT off to DQ/DBI_n in ODT state delay | $t_{ADZ}$ | - | 1 * tCK + 10 | - | 1 * tCK + 10 | ns | |
| Vendor ID on | $t_{WRIDON}$ | | 1 * tCK + 11 | | 1 * tCK + 11 | ns | |
| Vendor ID off | $t_{WRIDOFF}$ | | 1 * tCK + 11 | | 1 * tCK + 11 | ns | |

## Table 70 — AC Timings

| PARAMETER 1, 2 | SYMBOL | 10Gbps @1.25V | | 12Gbps @1.25V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |

Notes:

1. All parameters assume proper device initialization.
2. Tests for AC timing may be conducted at nominal supply voltage levels, but the related specifications and device operation are guaranteed for the full voltage and temperature range specified.
3. Parameter $f_{CKBG3}$ is required for those devices supporting both 3 * $t_{CK}$ and 4 * $t_{CK}$ settings for bank groups. Devices supporting only 3 * $t_{CK}$ or 4 * $t_{CK}$ need only to specify $f_{CKBG}$.
4. Bank Group Frequency ranges




5. Parameter $f_{CKPIN}$ applies when the alignment point in MR7, OP0 is set to "at balls", the phase difference between the WCK and CK clocks at the DRAM balls is within $t_{WCK2CKSYNC}$ or $t_{WCK2CK}$ for pin mode, and no phase search in WCK2CK training is performed.
6. Parameter $f_{CKVREFC2}$ applies when the CA input reference voltage in MR7, OP6 (Half VREFC) is set to VREFC2.
7. Parameter $f_{CKVREFD2}$ applies when the data input reference voltage in MR7, OP7 (Half VREFD) is set to VREFD2
8. Parameter $f_{CKAUTOSYNC}$ applies when WCK2CK Auto Synchronization is enabled in MR7, OP4.
9. Parameter $f_{CKLF}$ applies when Low Frequency Mode is enabled in MR7, OP3. Low Frequency Mode is optional.
10. By definition the nominal WCK clock cycle time always is 1/2 or 1/4 of the CK clock cycle time depending on the selected WCK Ratio (not including jitter).
11. The phase relationship between WCK_t/WCK_c and CK_t/CK_c clocks must meet the $t_{WCK2CK}$ specification.
12. CA input timings are referenced to $V_{REFC}$.
13. CA and CKE_n input pulse widths are design targets. The values will be characterized but not tested on each device.
14. CA input timings are only valid with CABI_n being enabled and a maximum of 5 CA inputs driven LOW in 2 channel mode or 7 CA inputs driven LOW in PC mode.
15. Parameter may be specified as a combination of $t_{CK}$ and ns.
16. Parameters $t_{WCKHTR}$ and $t_{WCKLTR}$ specify the max. allowed WCK clock-to-clock phase shift during WCK2CK training. For READ and WRITE bursts use $t_{WCKH}$ and $t_{WCKL}$.
17. Parameter $t_{WCK2CKPIN}$ defines the WCK2CK phase offset range at the CK and WCK balls for ideal (phase = 0°) clock alignment at the device's phase detector (when the alignment point in MR7, OP0 is set to "at phase detector"), or at the WCK and CK balls (when the alignment point in MR7, OP0 is set to "at pins"). The minimum and maximum values could be negative or positive numbers, depending on the selected WCK2CK alignment point, PLL-on or PLL-off mode and design implementation.
18. Parameter $t_{WCK2CKSYNC}$ defines the max. phase offset from the ideal (phase = 0°) clock alignment at the device's phase detector (when the alignment point in MR7, OP0 is set to "at phase detector"), or at the WCK and CK balls (when the alignment point in MR7, OP0 is set to "at pins"), where the internal logic synchronizes the CK and WCK clocks; it is expected to be a fraction of $t_{WCK2CK}$.
19. Parameter $t_{WCK2CK}$ defines the max. phase offset from the ideal (phase = 0°) clock alignment at the device's phase detector (when the alignment point in MR7, OP0 is set to "at phase detector"), or at the WCK and CK balls (when the alignment point in MR7, OP0 is set to "at pins"), for stable device operation.
20. Parameter $t_{WCK2DQI}$ defines the WCK to DQ/DBI_n time delay range for WRITEs for PLL-on and PLL-off mode. The minimum and maximum values could be negative or positive numbers, depending on design implementation and PLL-on or PLL-off mode. They also vary across PVT. Data training is required to determine the actual $t_{WCK2DQI}$ value for reliable WRITE operation.
21. Parameter $t_{WCK2DQO}$ defines the WCK to DQ/DBI_n/EDC time delay range for READs for PLL-on and PLL-off mode. The minimum and maximum values could be negative or positive numbers, depending on design implementation and PLL-on or PLL-off mode. They also vary across PVT. Data training is required to determine the actual $t_{WCK2DQO}$ value for reliable READ operation.
22. Outputs measured with equivalent load (vendor specific) terminated to $V_{DDQ}$.
23. DQ/DBI_n input timings are valid only with DBI being enabled and a maximum of 4 data inputs per byte driven LOW.
24. Data input slew rate must be greater than or equal to TBD V/ns. The slew rate is measured between $V_{REFD}$ crossing and $V_{IHD(AC)}$ or $V_{ILD(AC)}$.
25. The data input pulse width, $t_{DIPW}$, defines the minimum positive or negative input pulse width for any one worst-case channel required for proper propagation of an external signal to the receiver. $t_{DIPW}$ is independent of the PLL/DLL mode. In general $t_{DIPW}$ is larger than $t_{DIVW}$.

## Table 70 — AC Timings

| PARAMETER 1, 2 | SYMBOL | 10Gbps @1.25V | | 12Gbps @1.25V | | UNIT | NOTES |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | MAX | | |

26. The data input valid window, $t_{DIVW}$, defines the time region where input data must be valid for reliable data capture at the receiver for any one worst case channel. It accounts for jitter between data and clock at the latching point introduced in the path between the DRAM pads and the latching point. Any additional jitter introduced into the source signals (e.g., within the system before the DRAM pad) must be accounted for in the final timing budget together with the chosen PLL/DLL mode and bandwidth. $t_{DIVW}$ is defined for PLL/DLL off and on mode separately. In the case of PLL/DLL on, $t_{DIVW}$ must be specified for each supported bandwidth. In general $t_{DIVW}$ is smaller than $t_{DIPW}$.
27. $t_{DQDQI}$ defines the maximum skew among all DQ/DBI_n inputs of a byte under worst case conditions. Parameter $t_{WCK2DQI}$ defines the mean value of the earliest and latest DQ/DBI_n pin, $t_{DQDQI}$(min) the negative offset to $t_{WCK2DQI}$ for the earliest DQ/DBI_n pin and $t_{DQDQI}$(max) the positive offset to $t_{WCK2DQI}$ for the latest DQ/DBI_n pin.
28. $t_{DQDQO}$ defines the maximum skew among all DQ/DBI_n/EDC outputs of a byte under worst case conditions. Parameter $t_{WCK2DQO}$ defines the mean value of the earliest and latest DQ/DBI_n/EDC pin, $t_{DQDQO}$(min) the negative offset to $t_{WCK2DQO}$ for the earliest DQ/DBI_n/EDC pin and $t_{DQDQO}$(max) the positive offset to $t_{WCK2DQO}$ for the latest DQ/DBI_n/EDC pin.
29. For READs and WRITEs with AUTO PRECHARGE enabled the device will hold off the internal PRECHARGE until $t_{RAS}$(min) has been satisfied.
30. Parameter applies when bank groups are enabled and consecutive commands access the same bank group.
31. Parameter applies when bank groups are disabled or consecutive commands access different bank groups.
32. Not more than 4 ACTIVATE commands are allowed within period.
33. Parameter applies between any two REFpb / REFp2b commands and between a REFpb / REFp2b command and a subsequent ACTIVATE command to a different bank.
34. Parameter applies when bank groups are enabled and READ and PRECHARGE commands access the same bank.
35. Parameter applies when bank groups are disabled or READ and PRECHARGE commands access the same bank.
36. $t_{CCDL}$ is either for gapless consecutive READ or gapless consecutive WRITE commands.
37. $t_{CCDS}$ is either for gapless consecutive READ or RDTR (any combination), gapless consecutive WRITE, or gapless consecutive WRTR commands.
38. $t_{RTW}$ is not a device limit but determined by the system bus turnaround time. The difference between $t_{WCK2DQO}$ and $t_{WCK2DQI}$ shall be considered in the calculation of the bus turnaround time.
39. Read data including CRC data must have been clocked out before entering self refresh or power-down mode.
40. Write data must have been written to the memory core, and CRC data must have been clocked out before entering self refresh or power-down mode.
41. Time for WCK2CK training and data training not included.
42. A maximum of 8 consecutive REFab commands can be posted to a device, meaning that the maximum absolute interval between any REFab command and the next REFab command is 9 * $t_{REFI}$.
43. $t_{ABREF}$ is relevant only when refresh is normally performed using the REFpb / REFp2b command. REFab commands must be issued at a minimum rate of $t_{ABREF}$ to allow impedance updates from the auto-calibration engine to occur.
44. $t_{MRD}$ is less than or equal to $t_{MOD}$. See MODE REGISTER SET command section for more details on $t_{MOD}$ and $t_{MRD}$
45. Replaces parameter $t_{LK}$ when PLL Fast Lock has been enabled prior to the PLL/DLL enable or reset.
46. The parameter may be specified in $t_{CK}$ or ns (vendor specific)
47. The parameter applies when the CA reference voltage selection in MR7, OP6 (Half VREFC) has changed.
48. The parameter applies when the $V_{REFC}$ level has been changed in MR10. $t_{VREFC}$ is a constant value for the device, and is referenced from the MRS command to when the 90% level of the delta between old and new $V_{REFC}$ voltage has been reached.
49. The parameter applies when the $V_{REFD}$ level has been changed in MR6 or MR9. $t_{VREFD}$ is a constant value for the device, and is referenced from the MRS command to when the 90% level of the delta between old and new $V_{REFD}$ voltage has been reached.
50. The parameter applies when the data reference voltage selection in MR7, OP7 (Half VREFD) has changed.
51. The parameter applies to WCK per byte or WCK per word depending on device implementation.
52. This parameter is for DRAM design only and valid on the silicon die at the input of the receiver. It is not intended to be measured.
53. After issuing PDE or SRE command, NOP (1) commands and CABI_n = HIGH (if CABI is enabled) are required for a minimum of $t_{CPDED}$ after POWER-DOWN or SELF REFRESH entry.
54. This parameter is for DRAM design only and valid on the silicon die at the input of the receiver. It is not intended to be measured.
55. After issuing PDE or SRE command, NOP (1) commands and CABI_n = HIGH (if CABI is enabled) are required for a minimum of tCPDED after POWER-DOWN or SELFREFRESH entry.
56. the scope of tCKEPW is limited to CAT commands in Command Address Training. Refer to parameters tPD and tXP for the minimum CKE_n pin HIGH and LOW pulse width states in conjunction with power-down, and parameters tCKESR, tXS, tHSRF, tXHP for the minimum CKE_n pin HIGH and LOW pulse width states in conjunction with self refresh and hibernate self refresh
57. After issuing PDX or SRX command, NOP (1) commands and CABI_n = HIGH (if CABI is enabled) are required for a minimum of tXP after POWER-DOWN EXIT or tXS after SELF REFRESH EXIT

## 8.10 CLOCK-TO-DATA TIMING SENSITIVITY

The availability of clock-to-data (WCK2DQ) timing sensitivity information provides the controller the opportunity to anticipate the impact to timings from variations in environmental conditions (such as changes in voltage or temperature) allowing the controller to take corrective action if necessary (e.g., realigning WCK and DQ).

Variations in relative timing between WCK and data are reported for READ and WRITE paths. The reported values are characterized but not tested in production.

**Table 71 — WCK-to-Data-In Timing Sensitivity**

| PARAMETER | | SYMBOL | VALUES | UNIT | NOTES |
|---|---|---|---|---|---|
| WCK2DQI Sensitivity to variations in $V_{DDQ}$ | PLL off | $t_{I2VQSens}$ | | ps/V | 1, 2 |
| WCK2DQI Sensitivity to variations in $V_{DD}$ | PLL off | $t_{I2VSens}$ | | ps/V | 3, 4 |
| WCK2DQI Sensitivity to variations in Tcase | PLL off | $t_{I2TSens}$ | | ps/°C | 5, 6 |

NOTE 1 $tI2VQSens = ((t_{WCK2DQI}(V_{DDQ}(max)) - t_{WCK2DQI}(V_{DDQ}(min))) / (V_{DDQ}(max) - V_{DDQ}(min)))$.
NOTE 2 $V_{DD}(typ)$, $T_{case}$ = 85 °C, worst-case process corner.
NOTE 3 $tI2VSens = ((t_{WCK2DQI}(V_{DD}(max)) - t_{WCK2DQI}(V_{DD}(min))) / (V_{DD}(max) - V_{DD}(min)))$.
NOTE 4 $V_{DDQ}(typ)$, $T_{case}$ = 85 °C, worst-case process corner.
NOTE 5 $tI2TSens = ((t_{WCK2DQI}(T_{case}(max)) - t_{WCK2DQI}(T_{case}(min))) / (T_{case}(max) - T_{case}(min)))$.
NOTE 6 $V_{DDQ}(typ)$, $V_{DD}(typ)$, worst-case process corner.

**Table 72 — WCK-to-Data-Out Timing Sensitivity**

| PARAMETER | | SYMBOL | VALUES | UNIT | NOTES |
|---|---|---|---|---|---|
| WCK2DQO Sensitivity to variations in $V_{DDQ}$ | PLL off | $t_{O2VQSens}$ | | ps/V | 1, 2 |
| WCK2DQO Sensitivity to variations in $V_{DD}$ | PLL off | $t_{O2VSens}$ | | ps/V | 3, 4 |
| WCK2DQO Sensitivity to variations in Tcase | PLL off | $t_{O2TSens}$ | | ps/°C | 5, 6 |

NOTE 1 $tO2VQSens = ((t_{WCK2DQO}(V_{DDQ}(max)) - t_{WCK2DQO}(V_{DDQ}(min))) / (V_{DDQ}(max) - V_{DDQ}(min)))$.
NOTE 2 $V_{DD}(typ)$, $T_{case}$ = 85 °C, worst-case process corner.
NOTE 3 $tO2VSens = ((t_{WCK2DQO}(V_{DD}(max)) - t_{WCK2DQO}(V_{DD}(min))) / (V_{DD}(max) - V_{DD}(min)))$.
NOTE 4 $V_{DDQ}(typ)$, $T_{case}$ = 85 °C, worst-case process corner.
NOTE 5 $tO2TSens = ((t_{WCK2DQO}(T_{case}(max)) - t_{WCK2DQO}(T_{case}(min))) / (T_{case}(max) - T_{case}(min)))$.
NOTE 6 $V_{DDQ}(typ)$, $V_{DD}(typ)$, worst-case process corner.

## 8.11 1.35V I/O DRIVER MODELS

The Driver and Termination impedances are derived from the following test conditions under worst case process corners:

1. Nominal 1.35V ($V_{DD}/V_{DDQ}$)
2. Power the DRAM device and calibrate the output drivers and termination to eliminate process variation at 25 °C.
3. Reduce temperature to 10 °C and recalibrate.
4. Reduce temperature to 0 °C and take the fast corner measurement.
5. Raise temperature to 75 °C and recalibrate
6. Raise temperature to 85 °C and take the slow corner measurement
7. Reiterate 2 to 6 with $V_{DD}/V_{DDQ}$ 1.3095V
8. Reiterate 2 to 6 with $V_{DD}/V_{DDQ}$ 1.3905V
9. All obtained Driver and Termination IV characteristics have to be bounded by the specified MIN and MAX IV characteristics

The following values (Ideal with +/- 10% min/max) are targets for the designer and are not required to be met. Vendor datasheets should be consulted for further details. It is expected that the characteristics of the real curves will have some nonlinearity as shown in Figure 116 and Figure 117. This may help to reduce the overall capacitance and boost performance. It is up to the designer to find the optimum combination of linearity and capacitance for best Rx and Tx performance.

**Table 73 — 1.35V I/O Impedances**

| Pull-Down Characteristic at 40 ohms | | | |
|---|---|---|---|
| Voltage (V) | MIN(mA) | Ideal(mA) | MAX(mA) |
| 0.1 | 2.25 | 2.50 | 2.75 |
| 0.2 | 4.50 | 5.00 | 5.50 |
| 0.3 | 6.75 | 7.50 | 8.25 |
| 0.4 | 9.00 | 10.00 | 11.00 |
| 0.5 | 11.25 | 12.50 | 13.75 |
| 0.6 | 13.50 | 15.00 | 16.50 |
| 0.7 | 15.75 | 17.50 | 19.25 |
| 0.8 | 18.00 | 20.00 | 22.00 |
| 0.9 | 20.25 | 22.50 | 24.75 |
| 1.0 | 22.50 | 25.00 | 27.50 |
| 1.1 | 24.75 | 27.50 | 30.25 |
| 1.2 | 27.00 | 30.00 | 33.00 |
| 1.3 | 29.25 | 32.50 | 35.75 |
| 1.35 | 31.15 | 33.75 | 37.12 |

| Pull-Up/Termination Characteristic at 60 ohms | | | |
|---|---|---|---|
| Voltage (V) | MIN(mA) | Ideal(mA) | MAX(mA) |
| 0.1 | -1.50 | -1.67 | -1.83 |
| 0.2 | -3.00 | -3.33 | -3.67 |
| 0.3 | -4.50 | -5.00 | -5.50 |
| 0.4 | -6.00 | -6.67 | -7.33 |
| 0.5 | -7.50 | -8.33 | -9.17 |
| 0.6 | -9.00 | -10.00 | -11.00 |
| 0.7 | -10.50 | -11.67 | -12.83 |
| 0.8 | -12.00 | -13.33 | -14.67 |
| 0.9 | -13.50 | -15.00 | -16.50 |
| 1.0 | -15.00 | -16.67 | -18.33 |
| 1.1 | -16.50 | -18.33 | -20.17 |
| 1.2 | -18.00 | -20.00 | -22.00 |
| 1.3 | -19.50 | -21.67 | -23.83 |
| 1.35 | -20.25 | -22.50 | -24.75 |

## 8.11 1.35V I/O DRIVER MODELS (cont'd)

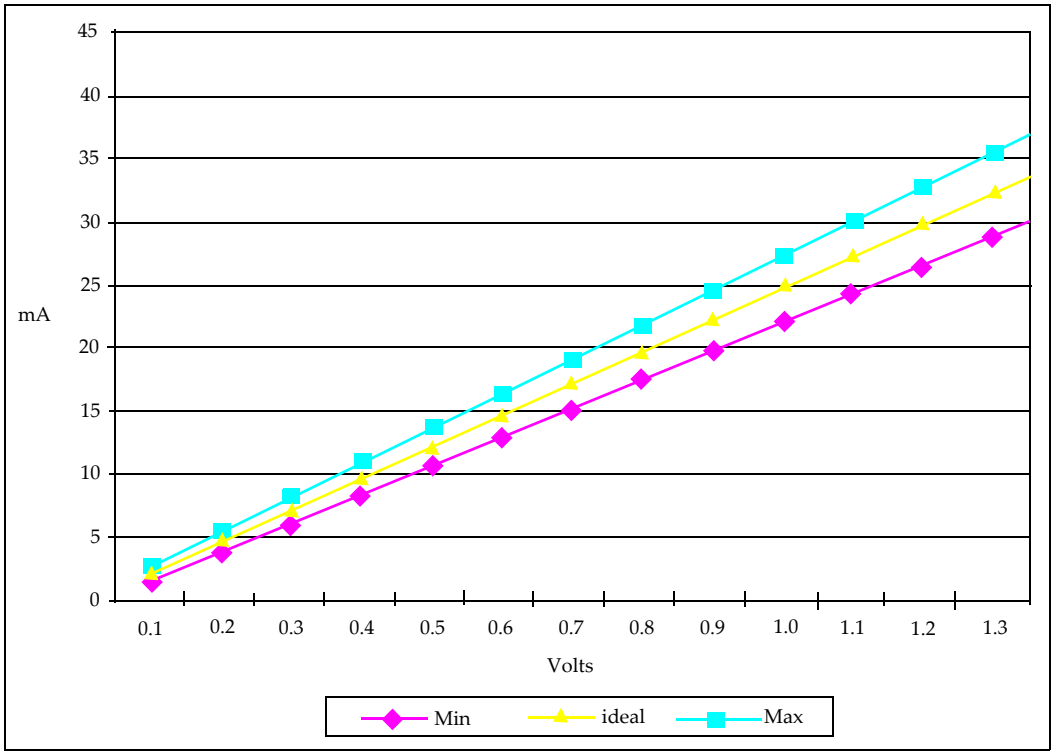



**Figure 117 — Target Pull Down Characteristic at 40 ohms**

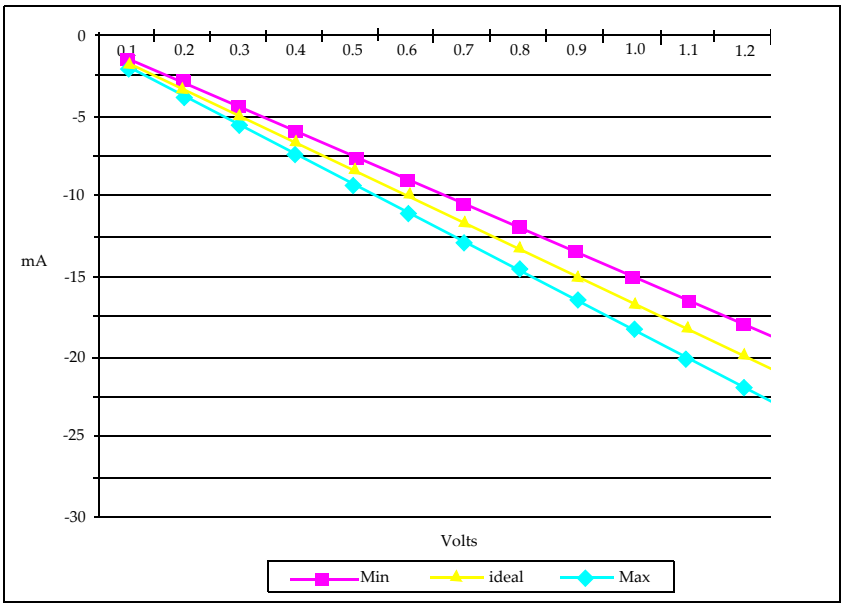



**Figure 118 — Target Pull Up/Termination Characteristic at 60 ohms**

## 8.11 1.35V I/O DRIVER MODELS (cont'd)

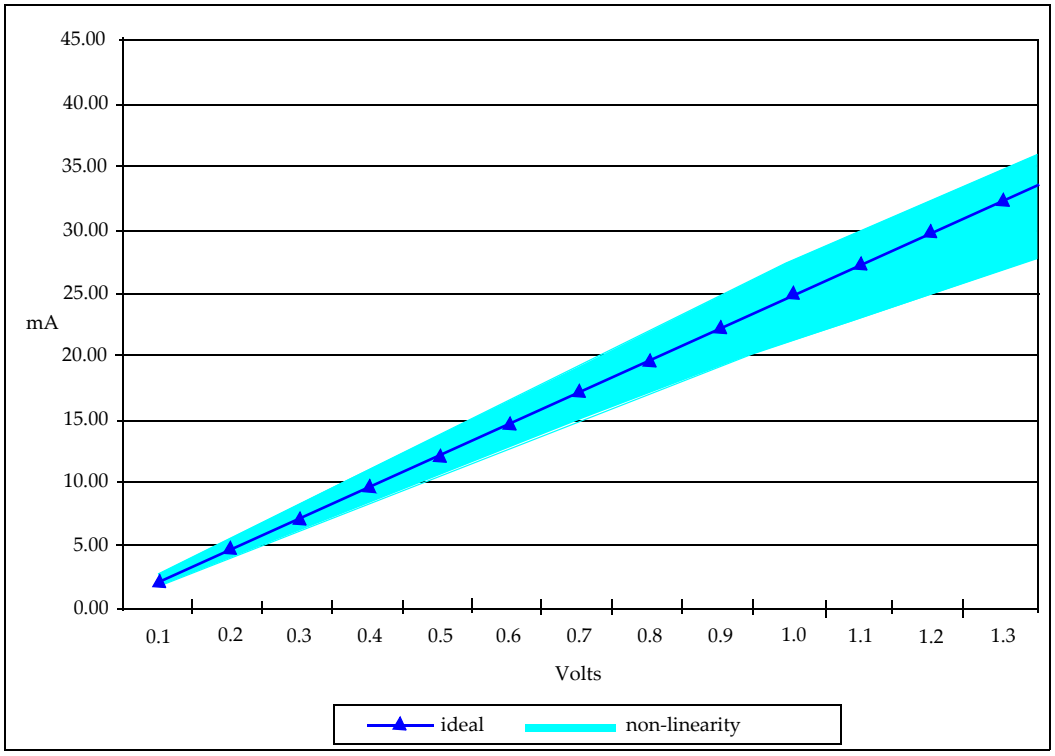



**Figure 119 — Example of Non Linearity, Pull Down Characteristic at 40 ohms**

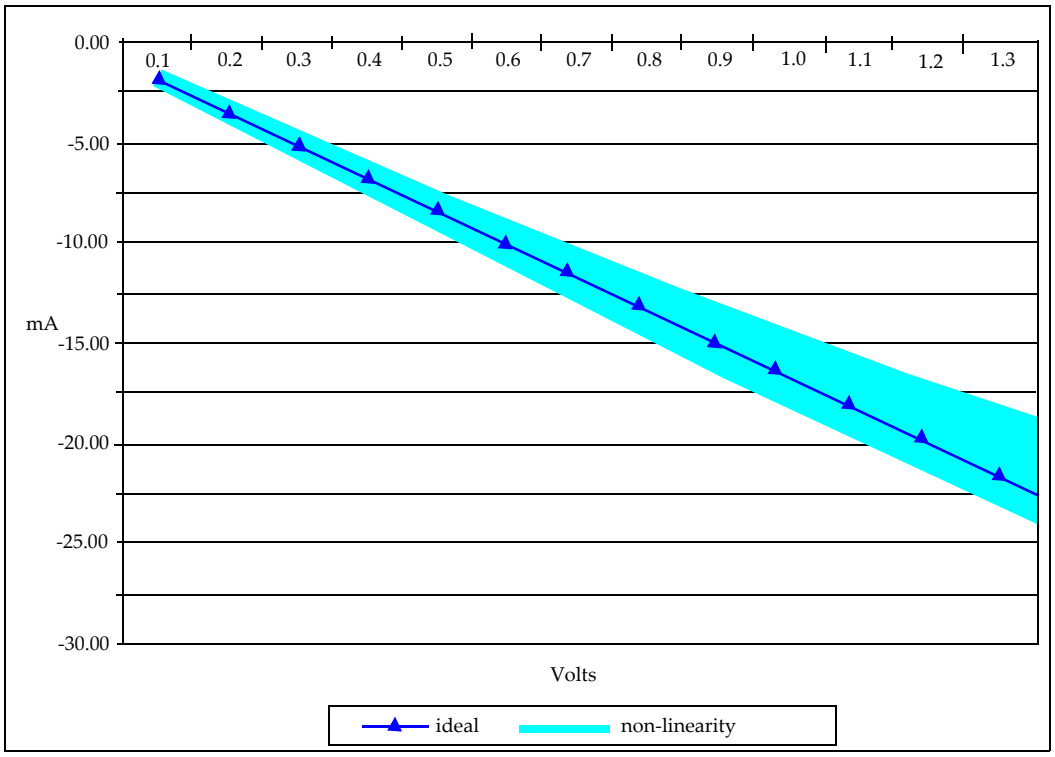



**Figure 120 — Example of Non Linearity, Pull Up/Termination Characteristic at 60 ohms**

## 8.12 1.25V I/O DRIVER MODELS

The Driver and Termination impedances are derived from the following test conditions under worst case process corners:

1. Nominal 1.25V ($V_{DD}/V_{DDQ}$)
2. Power the DRAM device and calibrate the output drivers and termination to eliminate process variation at 25 °C.
3. Reduce temperature to 10 °C and recalibrate.
4. Reduce temperature to 0 °C and take the fast corner measurement.
5. Raise temperature to 75 °C and recalibrate
6. Raise temperature to 85 °C and take the slow corner measurement
7. Reiterate 2 to 6 with $V_{DD}/V_{DDQ}$ 1.2125V
8. Reiterate 2 to 6 with $V_{DD}/V_{DDQ}$ 1.2875V
9. All obtained Driver and Termination IV characteristics have to be bounded by the specified MIN and MAX IV characteristics

The following values (Ideal with +/- 10% min/max) are targets for the designer and are not required to be met. Vendor datasheets should be consulted for further details. It is expected that the characteristics of the real curves will have some nonlinearity as shown in Figure 120 and Figure 121. This may help to reduce the overall capacitance and boost performance. It is up to the designer to find the optimum combination of linearity and capacitance for best Rx and Tx performance.

**Table 74 — 1.25V I/O Impedances**

| Pull-Down Characteristic at 40 ohms | | | |
|---|---|---|---|
| Voltage (V) | MIN(mA) | Ideal(mA) | MAX(mA) |
| 0.1 | 2.25 | 2.50 | 2.75 |
| 0.2 | 4.50 | 5.00 | 5.50 |
| 0.3 | 6.75 | 7.50 | 8.25 |
| 0.4 | 9.00 | 10.00 | 11.00 |
| 0.5 | 11.25 | 12.50 | 13.75 |
| 0.6 | 13.50 | 15.00 | 16.50 |
| 0.7 | 15.75 | 17.50 | 19.25 |
| 0.8 | 18.00 | 20.00 | 22.00 |
| 0.9 | 20.25 | 22.50 | 24.75 |
| 1.0 | 22.50 | 25.00 | 27.50 |
| 1.1 | 24.75 | 27.50 | 30.25 |
| 1.2 | 27.00 | 30.00 | 33.00 |
| 1.25 | 28.125 | 31.25 | 35.375 |

| Pull-Up/Termination Characteristic at 60 ohms | | | |
|---|---|---|---|
| Voltage (V) | MIN(mA) | Ideal(mA) | MAX(mA) |
| 0.1 | -1.50 | -1.67 | -1.83 |
| 0.2 | -3.00 | -3.33 | -3.67 |
| 0.3 | -4.50 | -5.00 | -5.50 |
| 0.4 | -6.00 | -6.67 | -7.33 |
| 0.5 | -7.50 | -8.33 | -9.17 |
| 0.6 | -9.00 | -10.00 | -11.00 |
| 0.7 | -10.50 | -11.67 | -12.83 |
| 0.8 | -12.00 | -13.33 | -14.67 |
| 0.9 | -13.50 | -15.00 | -16.50 |
| 1.0 | -15.00 | -16.67 | -18.33 |
| 1.1 | -16.50 | -18.33 | -20.17 |
| 1.2 | -18.00 | -20.00 | -22.00 |
| 1.25 | -18.75 | -21.835 | -22.915 |

## 8.12 1.25V I/O DRIVER MODELS (cont'd)

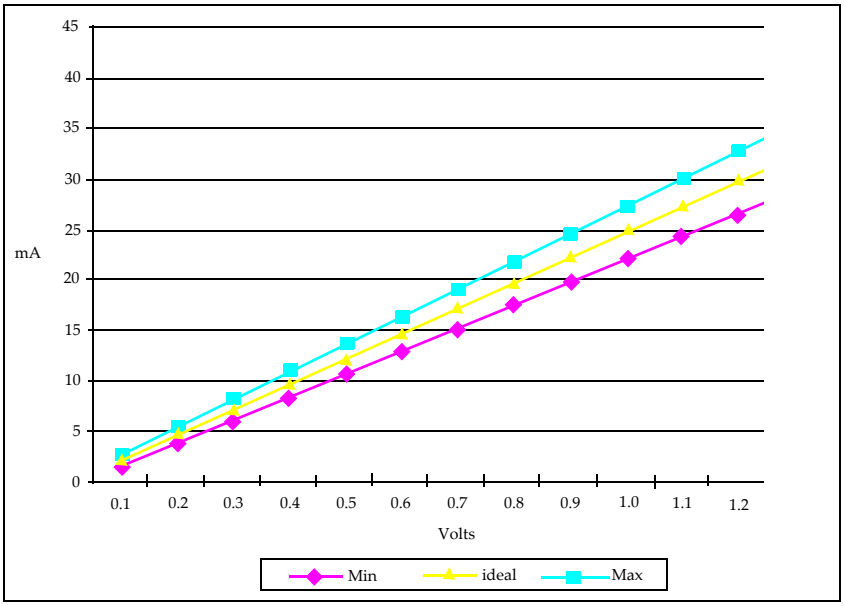



**Figure 121 — Target Pull Down Characteristic at 40 ohms**

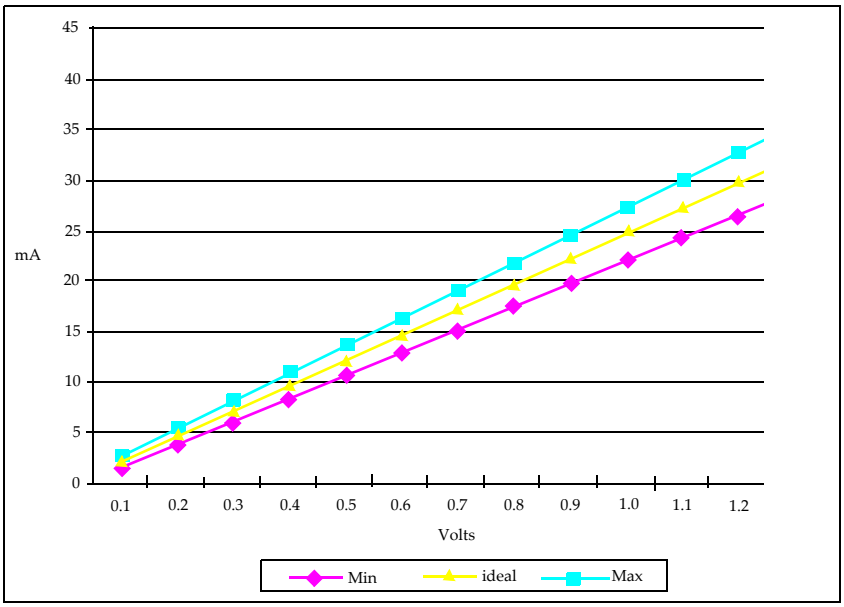



**Figure 122 — Target Pull Up/Termination Characteristic at 60 ohms**

## 8.12 1.25V I/O DRIVER MODELS (cont'd)

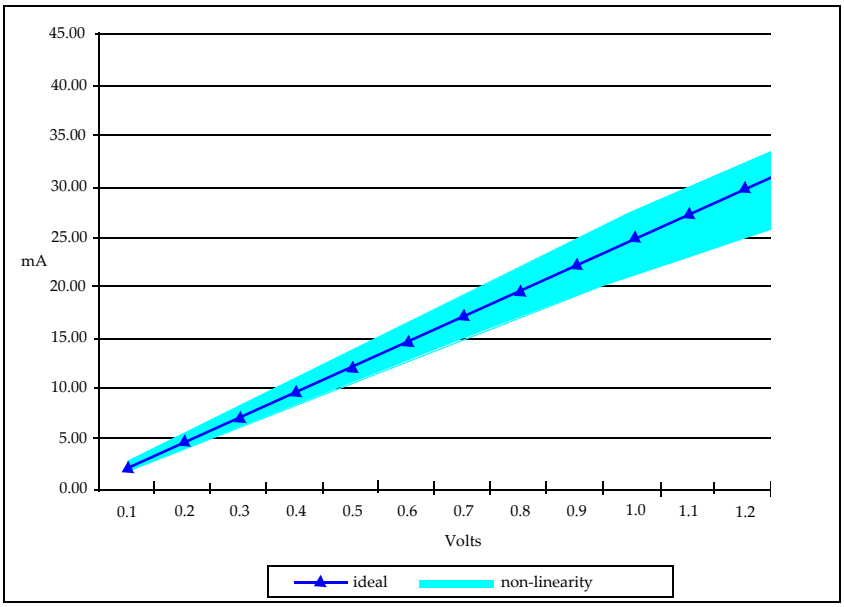



**Figure 123 — Example of Non Linearity, Pull Down Characteristic at 40 ohms**

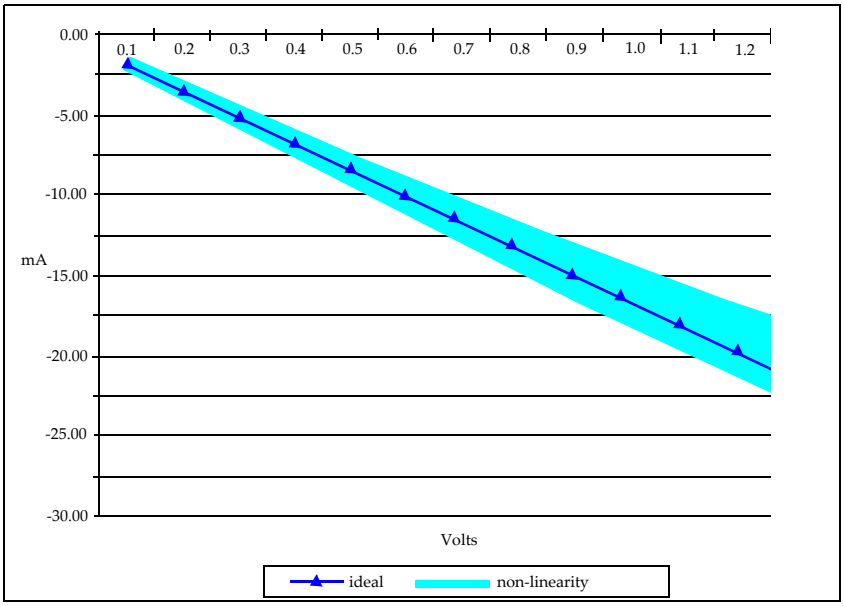



**Figure 124 — Example of Non Linearity, Pull Up/Termination Characteristic at 60 ohms**

# 9. ADDRESS

## 9.1 BALL-OUT

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | VDD | VSS | DQ1_A | VSS | VPP | | | | | VPP | VSS | DQ9_A | VSS | VDD |
| B | VSS | DQ3_A | DQ2_A | DQ0_A | VDDQ | | | | | VDDQ | DQ8_A | DQ10_A | DQ11_A | VSS |
| C | VDDQ | EDC0_A | VSS | VDDQ | VSS | | | | | VSS | VDDQ | VSS | EDC1_A | VDDQ |
| D | VSS | DBI0_n_A | VSS | WCK0_t_A | WCK0_c_A | | | | | WCK1_c_A | WCK1_t_A | VSS | DBI1_n_A | VSS |
| E | VDDQ | DQ5_A | DQ4_A | VSS | VDD | | | | | VDD | VSS | DQ12_A | DQ13_A | VDDQ |
| F | VSS | DQ6_A | VSS | VDDQ | TMS | | | | | TDI | VDDQ | VSS | DQ14_A | VSS |
| G | VSS | DQ7A | VSS | CA2_A | RFU_A, NC | | | | | CKE_n_A | CA1_A | VSS | DQ15_A | VSS |
| H | VDDQ | VDD | CA0_A | VSS | CA4_A | | | | | CA5_A | VSS | CA3_A | VDD | VDDQ |
| J | RESET_n | VDDQ | CA9_A | CA8_A | CABI_n_A | | | | | CK_t | CA7_A | CA6_A | VDDQ | ZQ_A |
| K | VREFC | VDDQ | CA9_B | CA8_B | CABI_n_B | | | | | CK_c | CA7_B | CA6_B | VDDQ | ZQ_B |
| L | VDDQ | VDD | CA0_B | VSS | CA4_B | | | | | CA5_B | VSS | CA3_B | VDD | VDDQ |
| M | VSS | DQ7_B | VSS | CA2_B | RFU_B, NC | | | | | CKE_n_B | CA1_B | VSS | DQ15_B | VSS |
| N | VSS | DQ6_B | VSS | VDDQ | TCK | | | | | TDO | VDDQ | VSS | DQ14_B | VSS |
| P | VDDQ | DQ5_B | DQ4_B | VSS | VDD | | | | | VDD | VSS | DQ12_B | DQ13_B | VDDQ |
| R | VSS | DBI0_n_B | VSS | WCK0_t_B | WCK0_c_B | | | | | WCK1_c_B | WCK1_t_B | VSS | DBI1_n_B | VSS |
| T | VDDQ | EDC0_B | VSS | VDDQ | VSS | | | | | VSS | VDDQ | VSS | EDC1_B | VDDQ |
| U | VSS | DQ3_B | DQ2_B | DQ0_B | VDDQ | | | | | VDDQ | DQ8_B | DQ10_B | DQ11_B | VSS |
| V | VDD | VSS | DQ1_B | VSS | VPP | | | | | VPP | VSS | DQ9_B | VSS | VDD |

NOTE Top View (as seen thru package)

**Figure 125 — GDDR6 SGRAM 180 ball BGA Ball-out**

## 9.2 SIGNALS

**Table 75 — Ball-out Description**

| SYMBOL | TYPE | DESCRIPTION |
|---|---|---|
| CK_t, CK_c | Input | Clock: CK_t and CK_c are differential clock inputs. CK_t and CK_c do not have channel indicators as one clock is shared between both Channel A and Channel B on a device. Command Address (CA) inputs are latched on the rising and falling edge of CK. All latencies are referenced to CK. |
| WCK0_t, WCK0_c, WCK1_t, WCK1_c | Input | Write Clocks: WCK_t and WCK_c are differential clocks used for WRITE data capture and READ data output. WCK0_t/WCK0_c is associated with DQ[7:0], DBI0_n and EDC0. WCK1_t/WCK1_c is associated with DQ[15:8], DBI1_n and EDC1. The ball out has a WCK/byte but devices can be designed that support only a WCK/word. In the case of a WCK/word, for Channel A the WCK0_t and WCK0_c are active and the unused WCK1 clock is a NC; for Channel B the WCK1_t and WCK1_c are active and unused WCK0 clock is a NC. The WCK Granularity of the Vendor ID can be read to determine whether the device has been designed with a WCK/byte or a WCK/word. |
| CKE_n | Input | Clock Enable: CKE_n LOW activates and CKE_n HIGH deactivates the internal clock, device input buffers, and output drivers excluding RESET_n, TDI, TDO, TMS and TCK. Taking CKE_n HIGH provides PRECHARGE POWER-DOWN and SELF REFRESH operations (all banks idle), or ACTIVE POWER-DOWN (row ACTIVE in any bank). CKE_n must be maintained LOW throughout read and write accesses. |
| CA[9:0] | Input | Command Address (CA) Inputs: The CA inputs receive packetized DDR commands, address or other information, for example, the op-code for the MRS command. See Command Truth Table for details. |
| DQ[15:0] | I/O | Data Input/Output: 16-bit data bus |
| DBI[1:0]_n | I/O | Data Bus Inversion. DBI0_n is associated with DQ[7:0], DBI1_n is associated with DQ[15:8]. |
| EDC[1:0] | Output | Error Detection Code. The calculated CRC data is transmitted on these signals. In addition these signals drive a 'hold' pattern when idle. EDC0 is associated with DQ[7:0], EDC1 is associated with DQ[15:8]. |
| CABI_n | Input | Command Address Bus Inversion |
| $V_{DDQ}$ | Supply | I/O Power Supply. Isolated on the die for improved noise immunity. |
| $V_{DD}$ | Supply | Power Supply |
| $V_{SS}$ | Supply | Ground |
| $V_{PP}$ | Supply | Pump Voltage |
| VREFC | Supply | Reference Voltage for CA/CABI_n/CKE_n signals |
| RFU | | Reserved for Future Use |
| NC | | No Connect |
| ZQ | Reference | External Reference for autocalibration |
| TDI | Input | JTAG test data input |
| TDO | Output | JTAG test data output |
| TMS | Input | JTAG test mode select |
| TCK | Input | JTAG test clock |
| RESET_n | Input | RESET_n LOW asynchronously initiates a full chip reset. With RESET_n LOW all ODTs are disabled. A full chip reset may be performed at any time by pulling RESET_n LOW. |
| NOTE 1: Index "_A" or "_B" represents the channel indicator "A" and "B" of the device. Signal names including the channel indicator are used whenever more than one channel is referenced, as e.g., with the ball-out. The channel indicator is omitted whenever features and functions common to both channels are described. | | |

## 9.3 ON DIE TERMINATION (ODT)

GDDR6 SGRAMs support multiple termination modes for its high speed input signals. When the termination is enabled for a receiver, an impedance defined for that termination mode is applied between that input receiver and the $V_{DDQ}$ supply rail. This is commonly referred to as $V_{DDQ}$ termination.

Table 75 includes all the high speed signals whose receivers include on die termination to $V_{DDQ}$ and the corresponding Mode Register bits to control their termination.

**Table 76 — Signals Affected by Termination Control Registers**

| Signal | Termination Control Registers | | Notes |
|---|---|---|---|
| | x16 mode | x8 mode | |
| CA[9:4]_A, CABI_n_A, CKE_n_A<br>CA[9:4]_B, CABI_n_B, CKE_n_B | CAH Termination + CA TO<br>MR8 OP [4:2] | | |
| CA[3:0]_A, CA[3:0]_B | CAL Termination + CA TO<br>MR8 OP [4, 1:0] | | |
| CK_t, CK_c | CK Termination<br>MR8 OP [11:10] | | 1 |
| DQ[7:0]_A, DBI0_n_A,<br>DQ[15:8]_B, DBI1_n_B | Data Termination<br>MR1 OP [3:2] | | |
| DQ[15:8]_A, DBI1_n_A,<br>DQ[7:0]_B, DBI0_n_B | Data Termination<br>MR1 OP [3:2] | Disabled | |
| WCK0_t_A, WCK0_c_A,<br>WCK1_t_B, WCK1_c_B | WCK Termination<br>MR10 OP [11:10] | | |
| WCK1_t_A, WCK1_c_A,<br>WCK0_t_B, WCK0_c_B, | WCK Termination<br>MR10 OP [11:10] | Disabled | 2 |

Notes:
1 CK ODT control via MRS is optional.
2 WCK1_A and WCK0_B are NC with WCK/word.

## 9.4 PACKAGE OUTLINE

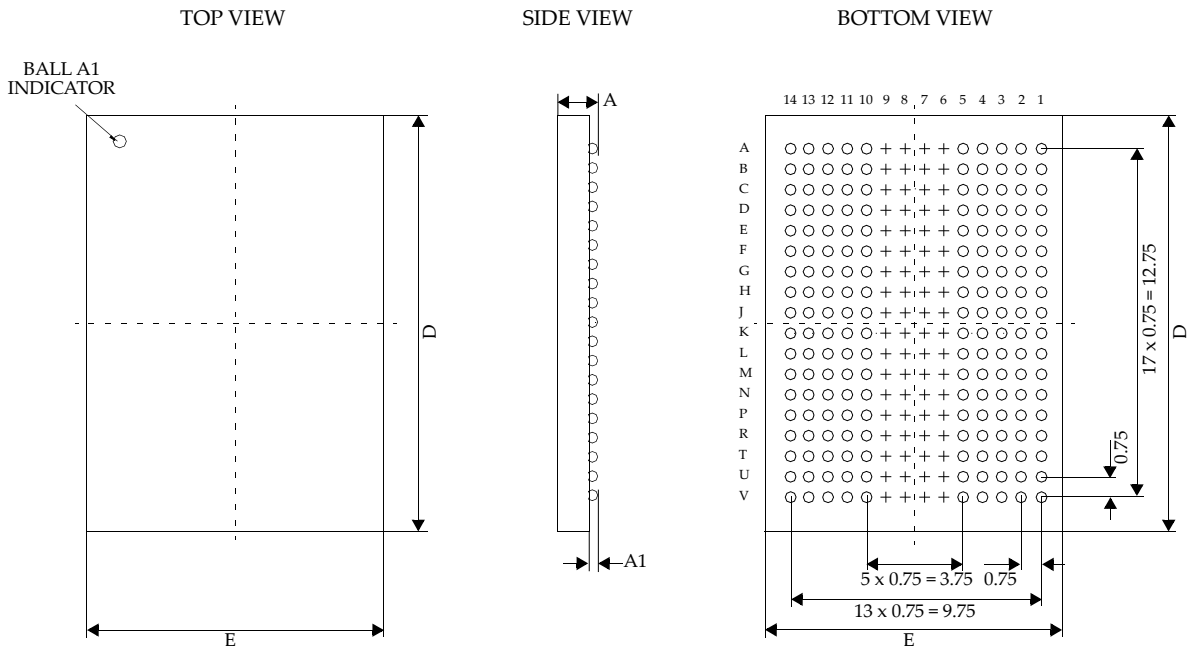



**Figure 126 — Package Dimensions[2]**

**Table 77 — Package Parameters**

| Parameter | Symbol | Minimum | Nominal | Variation |
|---|---|---|---|---|
| Length | D | | 14.00 | +/- 0.100 |
| Width | E | | 12.00 | +/- 0.100 |
| Height | A | | 1.10 | +/- 0.100 |
| Ball diameter | b | | 0.50 | +/- 0.050 |
| Standoff | A1 | 0.24 | | |
| Ball pitch | e | | 0.75 | |

NOTE 1 GDDR6 package size, height and standoff specification is compliant to MO-328, Variation P14.0x12.0-GJ-180A
NOTE 2 All dimensions in mm unless otherwise noted.
NOTE 3 Nominal Ball diameter (b) is vendor specific between 0.45mm and 0.5mm.

## 9.5 x8 MODE ENABLE

A GDDR6 SGRAM based memory system is typically divided into several channels. GDDR6 has been optimized for a 16 bit wide channel. A channel can be comprised of a single device operated in x16 mode, or two devices each operated in x8 mode. For x8 mode the devices are typically assembled on opposite sides of the PCB in what is referred to as a clamshell layout.

Whether in x16 mode or x8 mode the device will operate with a point to point connection on the high speed data signal. The disabled signals in x8 mode should all be in a Hi-Z state, non-terminating.

The x8 mode is detected at power up on EDC1_A and EDC0_B. For x8 mode this ball is tied to $V_{SS}$; the signal is part of the byte that is disabled in this mode and therefore not needed for EDC functionality. For x16 mode this signal is active and always terminated to $V_{DDQ}$ in the system or by the controller. The configuration is set with RESET_n going HIGH. Once the configuration has been set, it cannot be changed during normal operation. Usually the configuration is fixed in the system. Details of the x8 mode detection are depicted in Figure 124. A comparison of x16 mode and x8 mode systems is shown in Figure 125 and Figure 127 show examples of the board topologies that are supported in GDDR6.

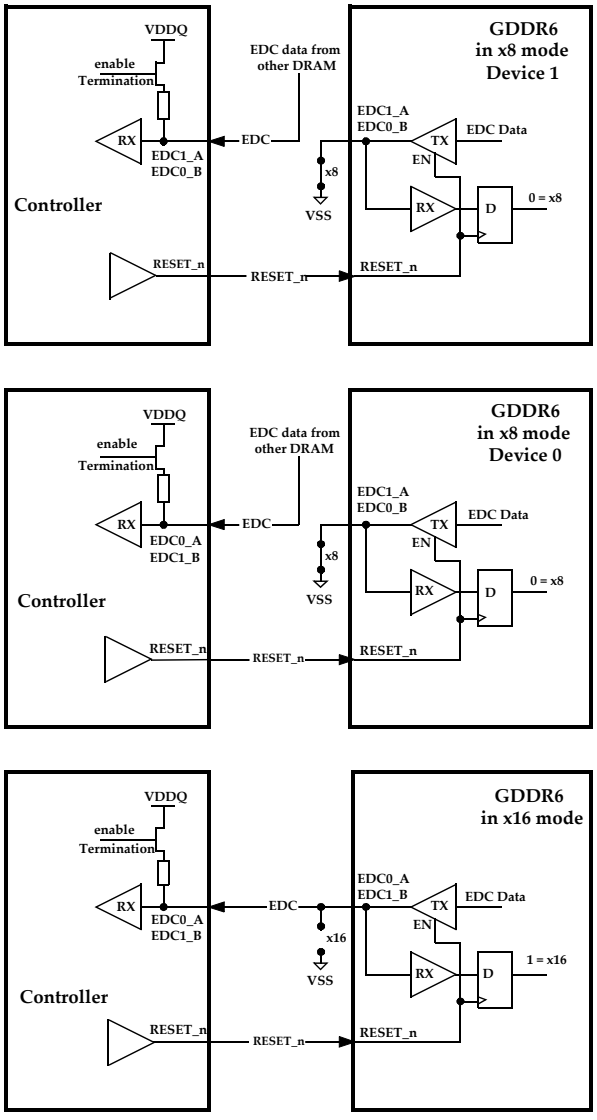
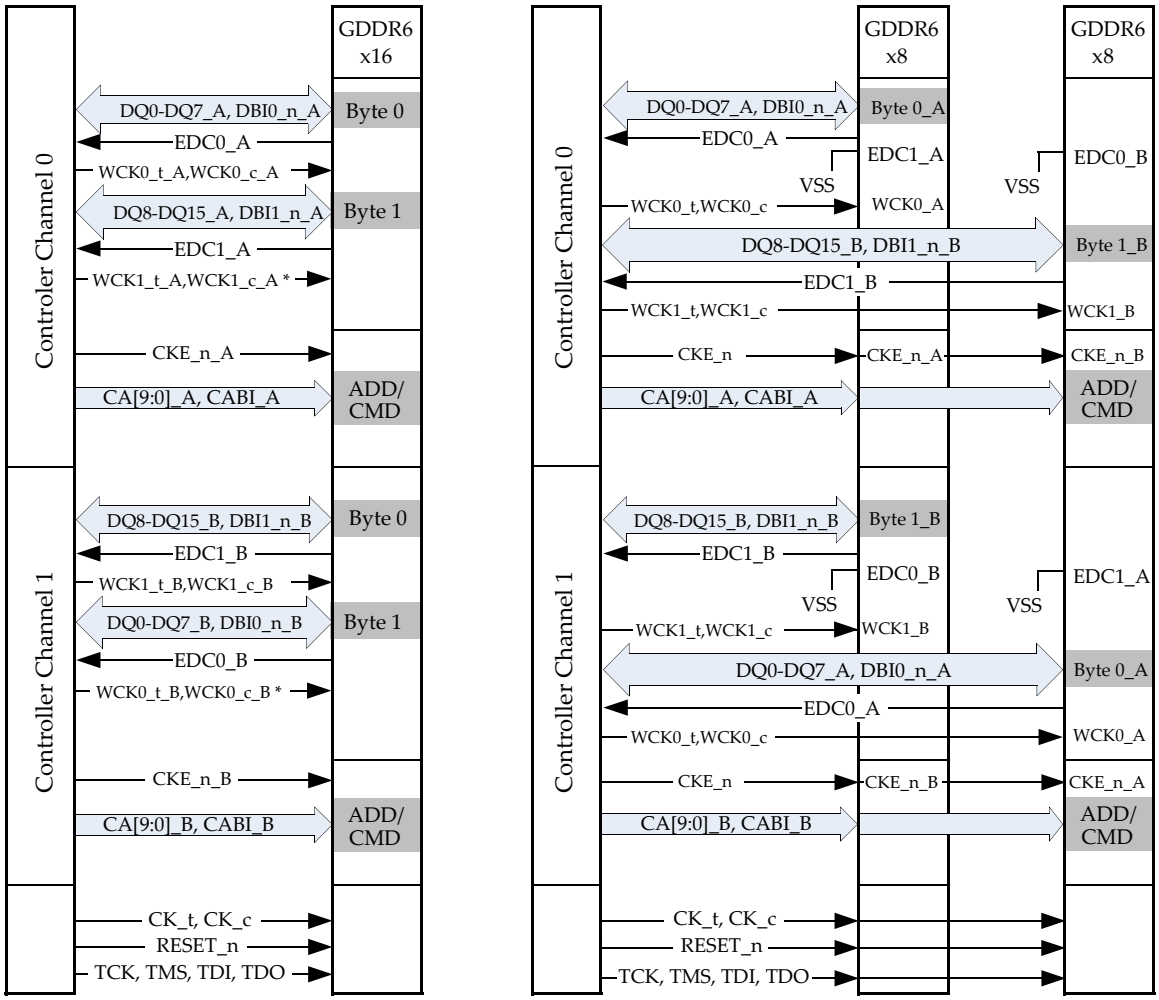



**Figure 127 — Enabling x8 mode**

## 9.5 x8 MODE ENABLE (cont'd)

**Table 78 — x8 mode**

| MODE | EDC0_A | EDC1_A | EDC0_B | EDC1_B |
|---|---|---|---|---|
| x8 | $V_{DDQ}$ | $V_{SS}$ (on board) | $V_{SS}$(on board) | $V_{DDQ}$ |
| x16 | $V_{DDQ}$ (terminated by the system or controller) | | | |




* WCK1_A and WCK0_B are NC with WCK/word.

**Figure 128 — Example System view for x16 mode vs. x8 mode**

## 9.5 x8 MODE ENABLE (cont'd)

Figure 126 is an example that clarifies the use of x8 mode and how the bytes are enabled/disabled to give the controller the view of the same bytes that a controller sees with a single x16 device. For a 16 bit channel using 2 devices in a clamshell design, Byte 0 comes from Channel A from the top device and Byte 1 comes from the bottom Channel B and will look equivalent at the controller to a x16 mode.

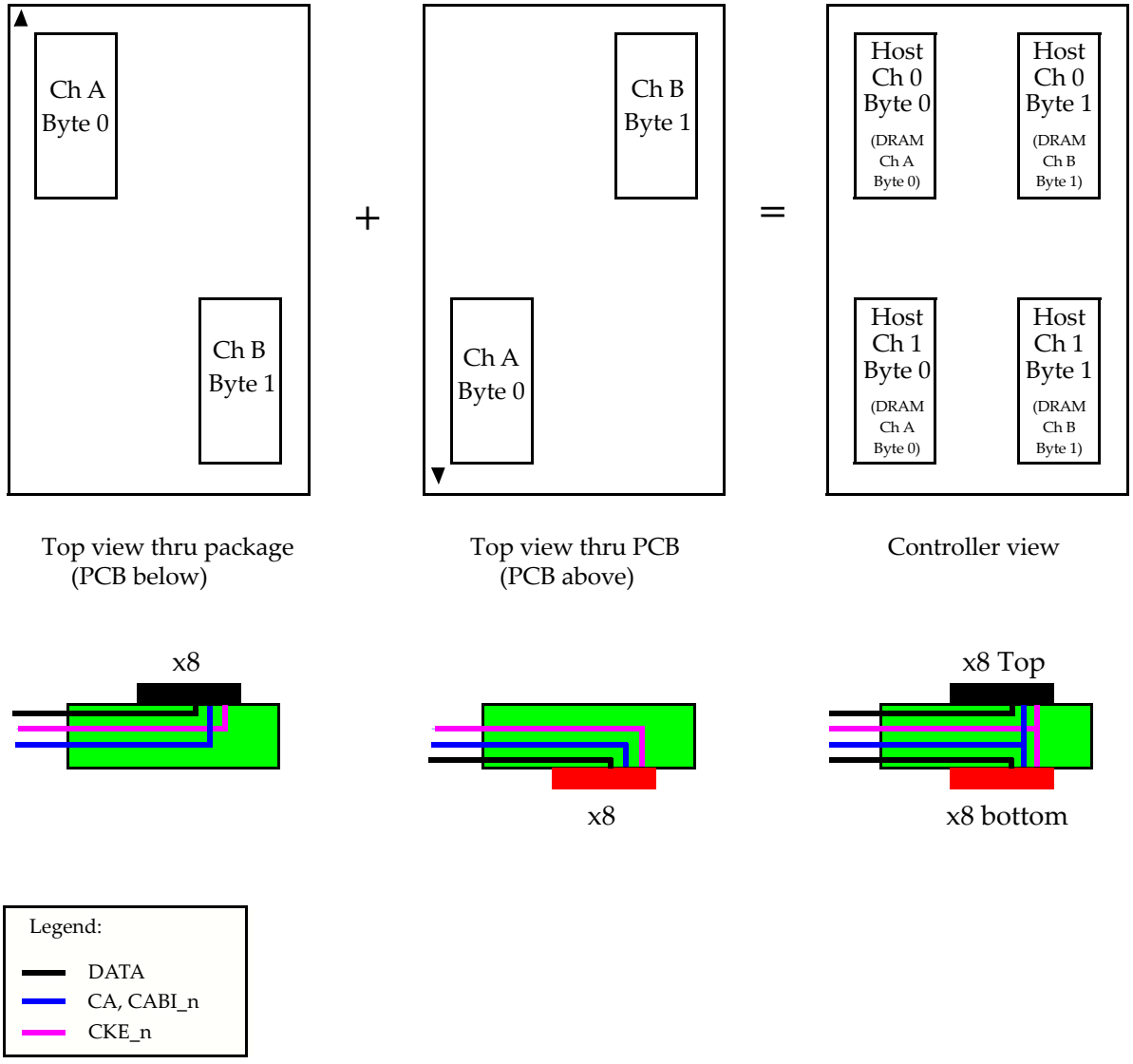



**Figure 129 — Byte Orientation in Clamshell Topology**

## 9.5 x8 MODE ENABLE (cont'd)

The simple block diagrams in Figure 127 demonstrate some of the flexibility of PCB routing.

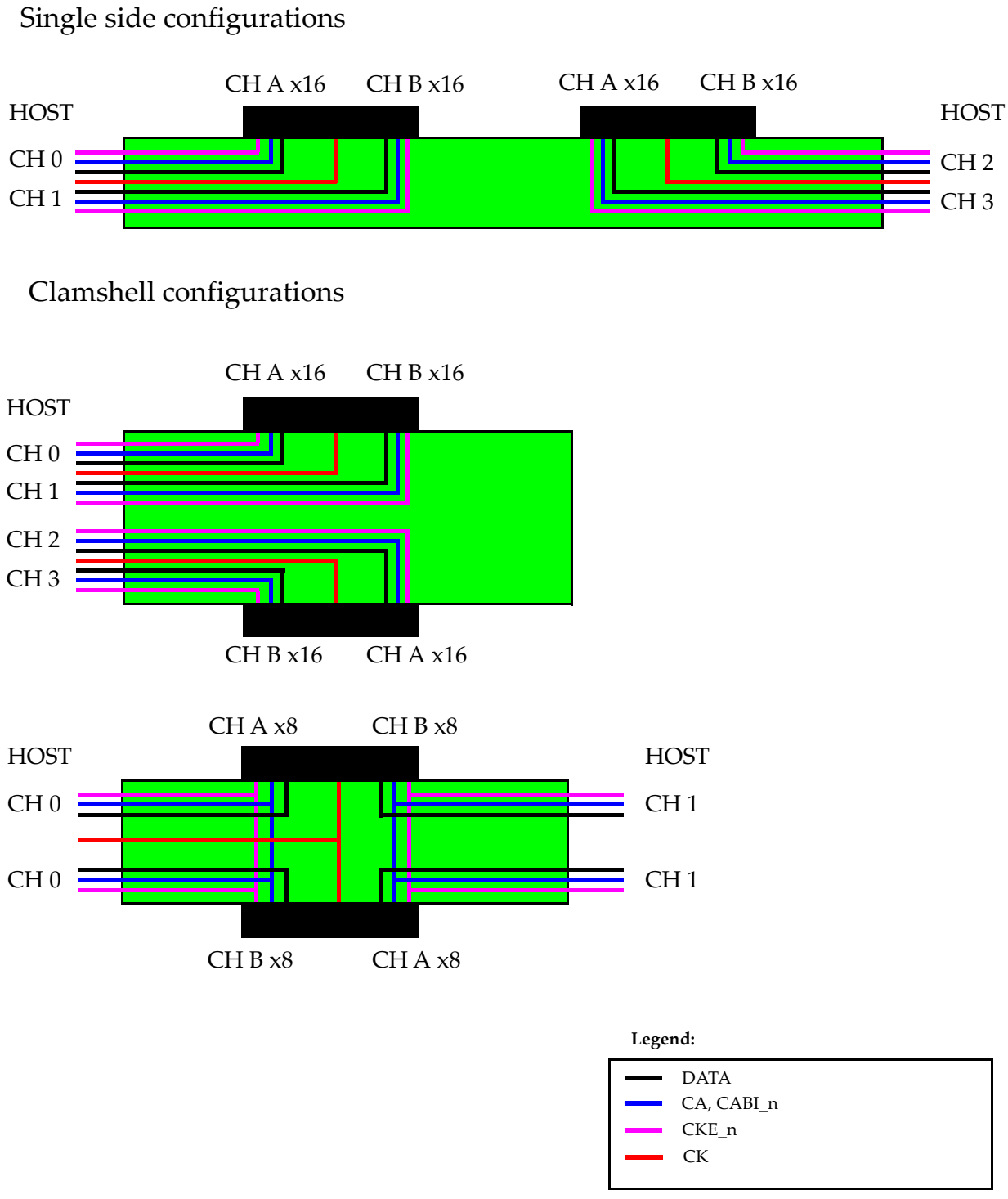



**Figure 130 — Example GDDR6 PCB Layout Topologies**

## 9.6 PSEUDO-CHANNEL (PC) MODE

A GDDR6 SGRAM based memory system is typically divided into several channels. GDDR6 has been optimized for a 32B access across a 16-bit channel by providing a unique CA[9:0] bus to each 16-bit wide channel. For applications requiring fewer CA pins, GDDR6 includes support for a PC mode where CA[9:4] on each channel is connected to a common bus while CA[3:0] of each channel is connected to a separate bus. The command truth table has been organized to enable this mode to support a unique column address to each pseudo-channel on CA[3:0]. In the PC mode, CKE_n and CABI_n are also shared across the pseudo-channels.

In the PC mode, the only difference in the DRAM is that termination on CA[9:4], CKE_n, and CABI_n can be configured differently from CA[3:0] but the functional operation of each channel is the same as in the normal mode.

The PC mode can be selected during initialization by driving CA6 = LOW on both channels when RESET_n is driven HIGH. The termination values are also selected at this time as defined in the initialization section. Mode Register MR8 includes controls for over-riding the termination impedance on the CA interface. MR8 OP4 = 0 selects the values selected during initialization. MR8 OP1 = 1 selects the values define by MR8 OP[3:0] as shown in Table 78 and Table 79.

**Table 79 — CAH Termination for CKE_n, CABI_n, CA[9:4]**

| MR8 OP[3:2] | Termination Strength |
|---|---|
| 00 | Disabled |
| 01 | 60 Ohm |
| 10 | 120 Ohm |
| 11 | 240 Ohm |

**Table 80 — CAL Termination for CA[3:0]**

| MR8 OP[3:2] | Termination Strength |
|---|---|
| 00 | Disabled |
| 01 | 60 Ohm |
| 10 | 120 Ohm |
| 11 | Reserved |

## 9.6 PSEUDO-CHANNEL (PC) MODE (cont'd)

Table 81 — Example System Configuration for Pseudo-Channel Mode

| GDDR6 | Controller | | |
|---|---|---|---|
| Dual channel | PC-A | PC-B | Shared pins |
| CA0_A | CA0_A | | |
| CA1_A | CA1_A | | |
| CA2_A | CA2_A | | |
| CA3_A | CA3_A | | |
| CA4_A | | | CA4_A_B |
| CA5_A | | | CA5_A_B |
| CA6_A | | | CA6_A_B |
| CA7_A | | | CA7_A_B |
| CA8_A | | | CA8_A_B |
| CA9_A | | | CA9_A_B |
| CABI_n_A | | | CABI_n_A_B |
| CKE_n_A | | | CKE_n_A_B |
| CA0_B | | CA0_B | |
| CA1_B | | CA1_B | |
| CA2_B | | CA2_B | |
| CA3_B | | CA3_B | |
| CA4_B | | | CA4_A_B |
| CA5_B | | | CA5_A_B |
| CA6_B | | | CA6_A_B |
| CA7_B | | | CA7_A_B |
| CA8_B | | | CA8_A_B |
| CA9_B | | | CA9_A_B |
| CABI_n_B | | | CABI_n_A_B |
| CKE_n_B | | | CKE_n_A_B |

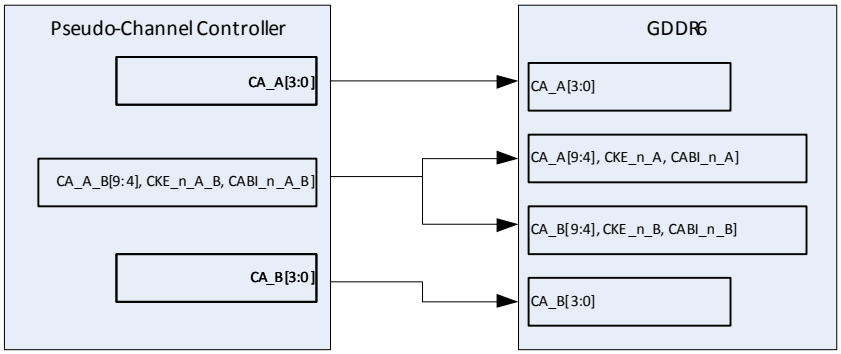



**Figure 131 — Example System Configuration for Pseudo-Channel Mode**

# 10. IEEE.1149.1 BOUNDARY SCAN

The GDDR6 SGRAM incorporates a boundary scan standard test access port that operates in accordance with IEEE Standard 1149.1-2013. It allows monitoring and control of the device's external I/O pins through a dedicated test port, and is controlled by an integrated test access port (TAP) controller.

Both channels A and B are equipped with their own TAP controller. This allows each channel to be tested individually, similar to a standalone device.

DRAM vendors may in addition use this test access port to gain access to proprietary test modes built into the device; this access will be provided through vendor defined instructions as outlined in section 10.4.

## 10.1 TEST PINS

Four dedicated pins are associated with the boundary scan test access port. The pins connect to each channel's TAP as illustrated in Figure 129 and described below.

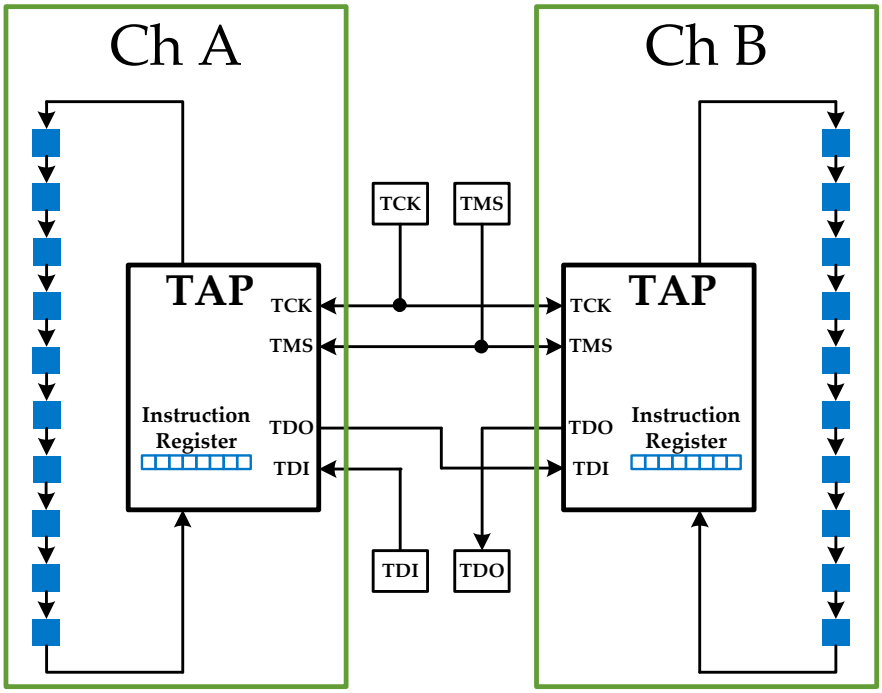



**Figure 132 — GDDR6 Dual TAP Controller Architecture**

### 10.1.1 Test Clock (TCK)

The test clock is used only with the TAP controller. All inputs are captured on the rising edge of TCK. All outputs are driven from the falling edge of TCK. TCK is common to both channels.

### 10.1.2 Test Mode Select (TMS)

Commands to the TAP controller are received through the test mode select input. An internal pull-up resistor ensures that an undriven input is latched as a logic 1. TMS is common to both channels.

### 10.1.3 Test Data Input (TDI)

The pin is used to serially load test instructions and data into the registers and can be connected to the input of any register. The register between TDI and TDO is selected by the instruction that is loaded into the TAP instruction register. An internal pull-up resistor ensures that an undriven input is latched as a logic 1. TDI is connected to the most significant bit (MSB) of any register. TDI is connected to channel A's test data input while channel B's test data input is internally connected to channel A's test data output.

### 10.1.4 Test Data Output (TDO)

The pin is used to clock test instructions and data serially out of the registers. The TDO output driver is only active during the Shift-IR and Shift-DR TAP controller states (see Figure 130) and is Hi-Z in all other states. The output changes on the falling edge of TCK. TDO is connected to the least significant bit (LSB) of any register. TDO is connected to channel B's test data output while channel A's test data output is internally connected to channel B's test data input.

## 10.2 TAP CONTROLLER

The TAP controller is a finite state machine that uses the logic level of the TMS pin at the rising edge of TCK to navigate through its various operating modes as illustrated in Figure 130. The states are described subsequently. Actions of the test logic (e.g., data capture, shift or register updates) occur on the next falling or rising TCK edge in each state.

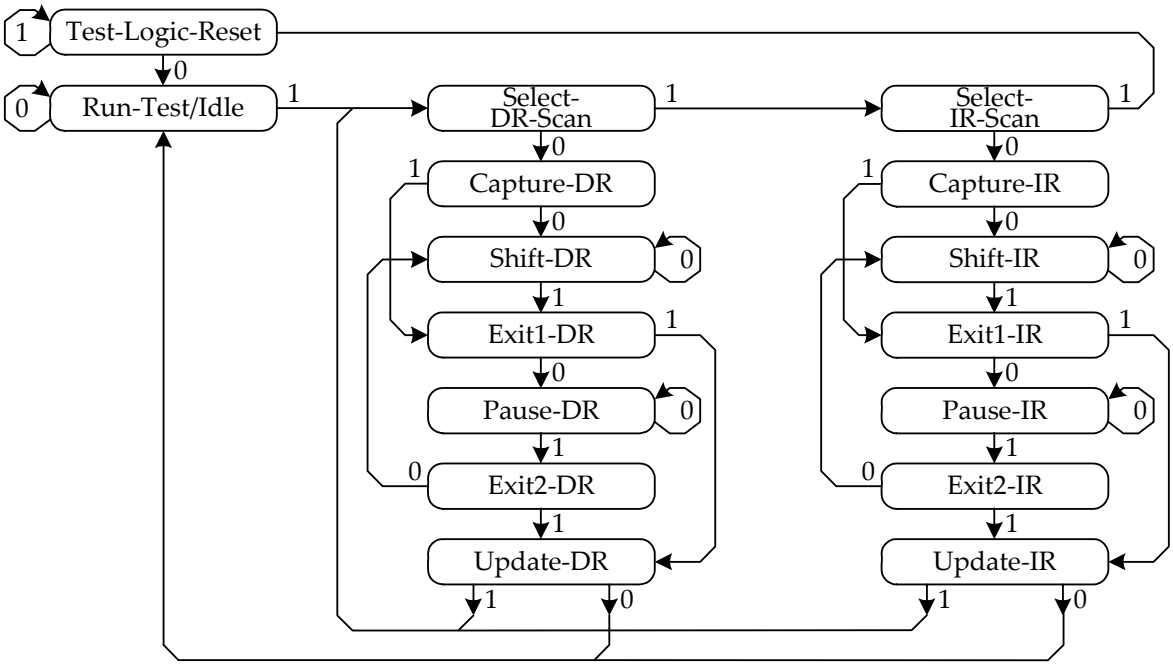



**Figure 133 — TAP Controller**

### 10.2.1 Test-Logic-Reset

The Test-Logic-Reset state is entered when TMS is held HIGH for at least five consecutive rising edges of TCK. As long as TMS remains HIGH, the TAP controller remains in the test-logic-reset state, and the test logic is inactive during this state. The test-logic-reset state is also asychronously entered upon device initialization to prevent any false interaction with the functional I/O of the device when boundary scan mode operation is not desired.

### 10.2.2 Run-Test/Idle

Run-Test/Idle is a controller state in between scan operations. The state can be maintained by holding TMS LOW. From here, either the data register scan, or subsequently, the instruction register scan can be selected.

### 10.2.3 Select-DR-Scan

This is a temporary controller state. All test data registers retain their state while here.

### 10.2.4 Capture-DR

This state is where data is parallel-loaded into the test data registers.

### 10.2.5 Shift-DR

Data is shifted serially through the data register while in this state. As new data is input through the TDI pin, data is shifted out of the TDO pin.

### 10.2.6 Exit1-DR, Pause-DR, and Exit2-DR

The purpose of Exit1-DR is to provide a path for return to the Run-Test/Idle state (through the Update-DR state). The Pause-DR state is entered when there is a need to suspend data shifting through the test registers. When shifting is to reconvene, the controller enters the Exit2-DR state and can then re-enter the Shift-DR state.

### 10.2.7 Update-DR

Data is parallel loaded from the shift register to the parallel output register on the falling edge of TCK in this controller state.

### 10.2.8 Instruction Register States

The instruction register states of the TAP controller are similar to the data register states. The desired instruction is shifted serially into the instruction register during the Shift-IR state and is loaded during the Update-IR state.

## 10.3 TAP REGISTERS

Several registers are provided to serially load and scan out instructions and data through the TDI and TDO pins. Only a single register can be selected at a time; the selection is determined by the active instruction and the TAP controller state.

The LSB (bit 0) of each register is located nearest to the serial output (TDO) and shifted out first.

### 10.3.1 Instruction (IR) Register

Eight-bit instructions (see Table 82) can be loaded serially into the instruction register. This register is loaded during the Update-IR state of the TAP controller. The instruction register is loaded with the IDCODE instruction upon power-up or when the TAP controller is in the Test-Logic-Reset state. When the TAP controller is in the Capture-IR state, the register is loaded with a hexadecimal 0x01 pattern to accommodate fault isolation of the board-level serial test data path.

### 10.3.2 Bypass (BY) Register

The BY register is a single-bit register that can be placed between the TDI and TDO pins. This enables data shifting through the device with minimal delay. The BY register is loaded with 0 in the Capture-DR controller state when the BYPASS, CLAMP or HIGH-Z instruction is executed.

### 10.3.3 Identification (ID) Register

The ID register contains a vendor specific 32-bit hardwired device identification code as shown in Figure 131. The register is loaded during the Capture-DR state when the IDCODE command is loaded in the instruction register. The register is shifted out in the Shift-DR controller state. The manufacturer identity field shall be programmed according to the latest revision of JEP106.

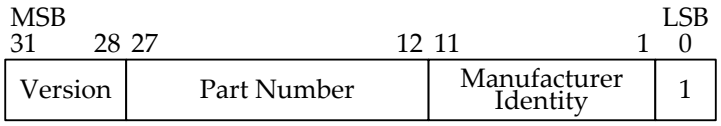



**Figure 134 — Identification Register**

### 10.3.4 Temperature (TR) Register

The 8-bit TR register is loaded with the binary readout of the device's temperature sensor (see TEMPERATURE SENSOR section) in the Capture-DR controller state when the TEMPERATURE instruction is loaded in the instruction register. The register is shifted out in the Shift-DR controller state.

### 10.3.5 Boundary Scan (BS) Register

The BS register is the primary test register that monitors and controls the data flow through the functional I/O of the device. The register is parallel-loaded with the contents of the I/O when the TAP controller is in the Capture-DR state. It is then placed between the TDI and TDO pins when the controller is moved to the Shift-DR state; in that state new test data are serially shifted in from the TDI pin while the captured pin state are shifted out on the TDO pin at the same time.

Different types of boundary scan cells are associated with each device pin. Example implementations of boundary scan cells are shown in Figure 132.

Each input pin is equipped with an observe-only cell for input data capture. Differential clocks (CK, WCK) are equipped with an observe-only input cell on both their true and complement inputs. $V_{REFC}$ and ZQ pins are as well equipped with an observe-only cell; their scan result may be masked (ignored) if, e.g., internal $V_{REFC}$ is used. Shared pins CK_t/CK_c, $V_{REFC}$ and RESET_n can be observed via both channel's TAP.

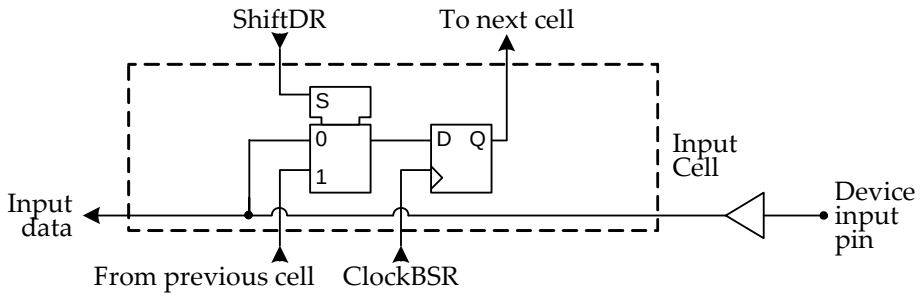
Each DQ pin is equipped with one combined input and output boundary scan cell. The cell performs the same function for input data capture as provided for the input pins, and also holds the output data. An additional boundary scan cell per byte controls the output enable of the 8 DQs of that byte.

Each DBI_n and EDC pin is equipped with two boundary scan cells. The first cell is the same combined input and output boundary scan cell for input data capture and output data as for a DQ pin. The second cell controls the output enable. It is pointed out that the input data capture on EDC pins is defined for EDC0_B and EDC1_A only and should be masked (ignored) for EDC0_A and EDC1_B.

The boundary scan register of each channel has a fixed length of 48 for all densities, and the boundary scan cells are serially connected as outlined in Table 81.

### 10.3.5 Boundary Scan (BS) Register (cont'd)

**Boundary scan input cell**




**Boundary scan cell at bidirectional I/O**

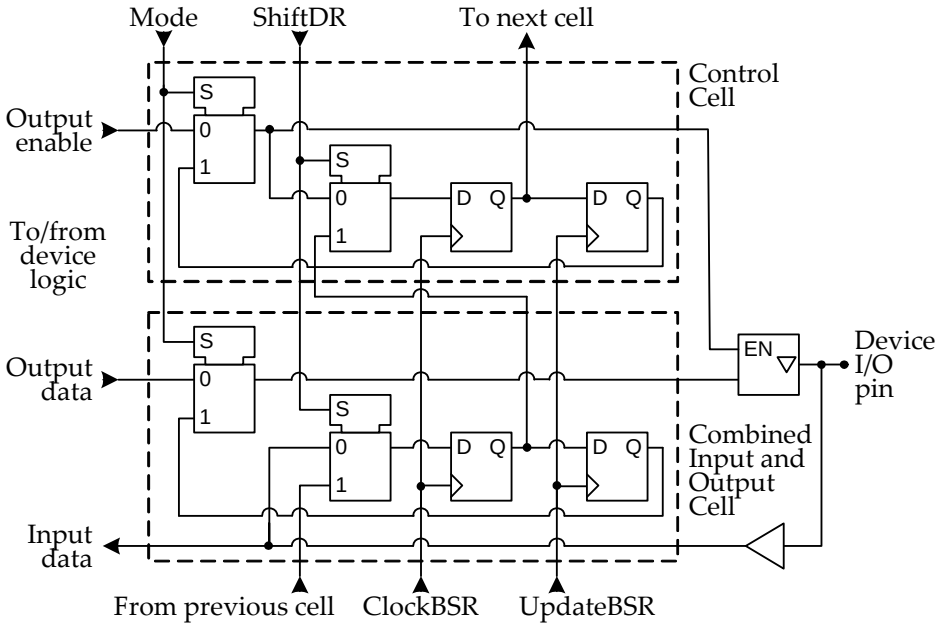



**Figure 135 — Example Boundary Scan Cells**

### 10.3.5 Boundary Scan (BS) Register (cont'd)

**Table 82 — Boundary Scan Register Bit Order**

| BIT | BALL | | TYPE | PIN | BIT | BALL | | TYPE | PIN | BIT | BALL | | TYPE | PIN |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Ch A | Ch B | | | | Ch A | Ch B | | | | Ch A | Ch B | | |
| 0 | K-1 | K-1 | I | VREFC | 16 | D-5 | R-5 | I | WCK0_c | 32 | – | – | OE | DQ[15:8] |
| 1 | J-1 | J-1 | I | RESET_n | 17 | D-4 | R-4 | I | WCK0_t | 33 | D-13 | R-13 | OE | DBI1_n |
| 2 | J-5 | K-5 | I | CABI_n | 18 | C-2 | T-2 | OE | EDC0 | 34 | D-13 | R-13 | I/O | DBI1_n |
| 3 | J-3 | K-3 | I | CA9 | 19 | C-2 | T-2 | I/O | EDC0 | 35 | E-12 | P-12 | I/O | DQ12 |
| 4 | J-4 | K-4 | I | CA8 | 20 | B-2 | U-2 | I/O | DQ3 | 36 | E-13 | P-13 | I/O | DQ13 |
| 5 | H-5 | L-5 | I | CA4 | 21 | B-3 | U-3 | I/O | DQ2 | 37 | F-13 | N-13 | I/O | DQ14 |
| 6 | G-4 | M-4 | I | CA2 | 22 | A-3 | V-3 | I/O | DQ1 | 38 | G-13 | M-13 | I/O | DQ15 |
| 7 | H-3 | L-3 | I | CA0 | 23 | B-4 | U-4 | I/O | DQ0 | 39 | G-10 | M-10 | I | CKE_n |
| 8 | G-5 | M-5 | | RFU | 24 | B-11 | U-11 | I/O | DQ8 | 40 | G-11 | M-11 | I | CA1 |
| 9 | G-2 | M-2 | I/O | DQ7 | 25 | A-12 | V-12 | I/O | DQ9 | 41 | H-12 | L-12 | I | CA3 |
| 10 | F-2 | N-2 | I/O | DQ6 | 26 | B-12 | U-12 | I/O | DQ10 | 42 | H-10 | L-10 | I | CA5 |
| 11 | E-2 | P-2 | I/O | DQ5 | 27 | B-13 | U-13 | I/O | DQ11 | 43 | J-12 | K-12 | I | CA6 |
| 12 | E-3 | P-3 | I/O | DQ4 | 28 | C-13 | T-13 | OE | EDC1 | 44 | J-11 | K-11 | I | CA7 |
| 13 | D-2 | R-2 | OE | DBI0_n | 29 | C-13 | T-13 | I/O | EDC1 | 45 | K-10 | K-10 | I | CK_c |
| 14 | D-2 | R-2 | I/O | DBI0_n | 30 | D-10 | R-10 | I | WCK1_c | 46 | J-10 | J-10 | I | CK_t |
| 15 | – | – | OE | DQ[7:0] | 31 | D-11 | R-11 | I | WCK1_t | 47 | J-14 | K-14 | I | ZQ |

NOTE 1 Register types: I = input cell; I/O = combined input and output cell; OE = output enable control cell.
NOTE 2 Bit order applies to both channels A and B.
NOTE 3 Shared pins CK_t/CK_c, VREFC and RESET_n can be observed via both channel's TAP.
NOTE 4 Input data capture on EDC pins is defined for EDC0_B and EDC1_A only and should be masked (ignored) for EDC0_A and EDC1_B.
NOTE 5 A single shift register cell without connection to a primary input or output is associated with the RFU pin as a placeholder as the type (input or output) of this pin is not yet defined. The state of this cell changes in the SHIFT_DR controller state only.

## 10.4 TAP INSTRUCTION SET

Table 82 summarizes the instructions supported by the TAP controller. Instruction codes not listed in the table may be used by the DRAM vendor for vendor specific test instructions.

**Table 83 — Boundary Scan Instructions**

| Instruction | Op-Code | Active Data Register | Register Length |
|---|---|---|---|
| BYPASS | 0x00 | Bypass | 1 |
| SAMPLE/PRELOAD | 0x01 | Boundary Scan | 48 |
| IDCODE | 0x02 | Identification | 32 |
| CLAMP | 0x04 | Boundary Scan / Bypass | 48 / 1 |
| HIGH-Z | 0x08 | Bypass | 1 |
| EXTEST | 0x10 | Boundary Scan | 48 |
| TEMPERATURE | 0x20 | Temperature | 8 |
| BYPASS | 0xFF | Bypass | 1 |

### 10.4.1 SAMPLE/PRELOAD

The SAMPLE/PRELOAD instruction provides two functions. The SAMPLE function allows a snapshot of the states of the device's input and output signals to be taken without interfering with the system's normal operation; the snapshot is taken and captured in the boundary scan register on the rising edge of TCK when the TAP controller is in the Capture-DR state. The data can then be viewed by shifting through the device's TDO output.

The PRELOAD function allows an initial data pattern to be placed at the latched parallel outputs of boundary scan register cells before the selection of another boundary scan test operation, for example, before the selection of the EXTEST instruction. As soon as the EXTEST instruction has been transferred to the parallel output of the instruction register, the preloaded data are driven through the system output pins.

### 10.4.2 IDCODE

The IDCODE instruction causes loading of a vendor-specific, 32-bit code into the identification register. It also places the identification register between the TDI and TDO pins and enables shifting the IDCODE out of the device when the TAP controller enters the Shift-DR state. The IDCODE instruction is loaded into the instruction register upon power-up or whenever the TAP controller is in the Test-Logic-Reset state.

### 10.4.3 CLAMP

The CLAMP instruction allows the state of the device's output pins to be determined from the boundary scan register, while the bypass register is selected as the serial path between TDI and TDO. Data in the boundary scan register may result from the previous use of the PRELOAD instruction. The signals driven from the output pins do not change while the CLAMP instruction is selected.

### 10.4.4 HIGH-Z

The HIGH-Z instruction places all device output pins into a Hi-Z state; it causes the bypass register to be connected between TDI and TDO.

### 10.4.5 EXTEST

The EXTEST instruction allows testing of off-chip circuitry and board-level interconnections. Test data typically would be loaded onto the latched parallel outputs of boundary scan shift-register stages using the PRELOAD instruction before selection of the EXTEST instruction.

When the EXTEST instruction is selected, the state of all signals driven from the device's output pins changes only on the falling edge of TCK in the Update-DR controller state, and the state of all signals received at the device's input pins shall be loaded into the boundary scan register on the rising edge of TCK in the Capture-DR controller state.

### 10.4.6 BYPASS

When the BYPASS instruction is loaded in the instruction register and the TAP is placed in a Shift-DR state, the bypass register is placed between TDI and TDO. Please note that two op-codes are associated with the BYPASS instruction.

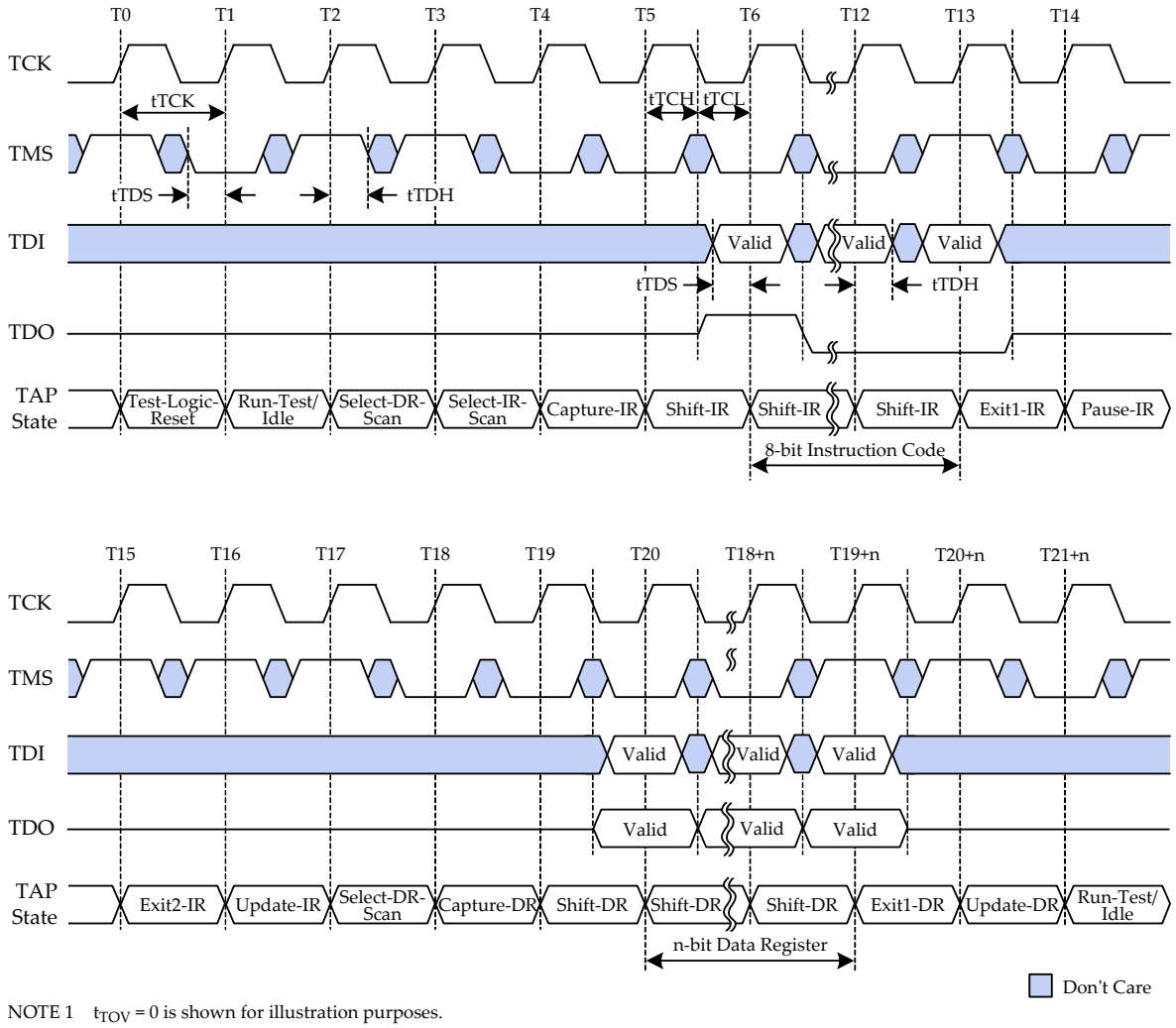
### 10.4.7 TEMPERATURE

The TEMPERATURE instruction provides an alternative method to read the device's junction temperature at any time without interrupting the normal mode operation. It places the temperature register between the TDI and TDO pins.

## 10.5 BOUNDARY SCAN OPERATION

Figure 133 illustrates a typical boundary scan test port operating sequence. The TAP states correspond to those shown in Figure 130. The sequence starts from the Test-Logic-Reset state. At first an instruction is serially loaded into the instruction register at clock edges T6 through T13 and latched in the Update_IR state at T16. The TAP then advances to the data register states, to parallel-load test data into the test data register in the Capture-DR state at T19, serially shift the test data in the Shift-DR state starting at T20, and finally to update the test data register's parallel outputs in the Update-DR state at T20+n.

It is pointed out that this example sequence refers to channel A or channel B separately. Any test pattern shift operation for a GDDR6 device is achieved by serially connecting the test pattern for channels A and B in accordance with the internal connection between both TAPs as shown in Figure 129. As an example, the 8-bit instruction registers of channel A and B are connected in series in the SHIFT_IR state. The 8-bit instruction code for channel B will be shifted in first immediately followed by the 8-bit instruction code for channel A, resulting in a total of 16 shift operations to load both channel's instruction registers.




NOTE 1 $t_{TOV}$ = 0 is shown for illustration purposes.

**Figure 136 — Example Boundary Scan Operation**

## 10.5 BOUNDARY SCAN OPERATION (cont'd)

**Table 84 — Boundary Scan AC Electrical Characteristics**

| PARAMETER/CONDITION | SYMBOL | MIN | MAX | UNIT |
|---|---|---|---|---|
| TCK clock cycle time | $t_{TCK}$ | 20 | - | ns |
| TCK clock HIGH-level width | $t_{TCH}$ | 0.45 | - | $t_{TCK}$ |
| TCK clock LOW-level width | $t_{TCL}$ | 0.45 | - | $t_{TCK}$ |
| TDI, TMS input setup time to rising TCK edge | $t_{TDS}$ | 2.5 | - | ns |
| TDI, TMS input hold time from rising TCK edge | $t_{TDH}$ | 2.5 | - | ns |
| TDO output hold time from falling TCK edge | $t_{TOH}$ | 0 | - | ns |
| TDO output valid time from falling TCK edge | $t_{TOV}$ | - | 5 | ns |

## 10.6 INTERACTIONS BETWEEN BOUNDARY SCAN AND NORMAL DEVICE OPERATION

Boundary scan operation may be initiated at any time after all voltages are stable (step 3 of the power-up sequence). Interactions between normal device operation and boundary scan operation depend on the selected scan instruction (see Table 82):

- Instructions BYPASS, IDCODE, TEMPERATURE and SAMPLE/PRELOAD can be loaded and executed in parallel with normal device operation. The state of the high speed pins is not impacted by these scan instructions, and all commands are executed as normal. It is pointed out that the SAMPLE function of the SAMPLE/PRELOAD instruction is not supported; any data sampled with this instruction shall be ignored.
- Instructions EXTEST, CLAMP and HIGH-Z take control of the high speed pins and therefore cannot be executed in parallel with normal device operation. The device disables the respective channel's command decoder, terminates all ongoing command executions and forces the respective channel into reset state once one of these instructions is loaded into the instruction register.

Once boundary scan operation has been completed, the device may return to normal operation without cycling through power-off and power-on by following this sequence:

1. Assert RESET_n to LOW.

2. Terminate boundary scan operation by setting the TAP controller back to Test-Logic-Reset state; this state will load the IDCODE instruction into the instruction register, enable the command decoder and release control of the high speed pins.

3. Maintain RESET_n LOW for a minimum time of $t_{RES}$, then continue with step 3 of the initialization with stable power sequence