Technical Note

Using DDR4 in Networking Subsystems

Introduction

DDR4 is the evolutionary transition from DDR3, supporting features and functions that offer lower power, higher performance, and better manufacturability over earlier DRAM products. The features supported by DDR4 are well suited for networking equipment manufacturers who are under pressure to increase the bandwidth capabilities of their equipment to support the growth of online traffic. Major contributors to this traffic increase include streaming video, cloud-based applications and storage, social networking, and an increase in mobile connectivity users. For example:

According to Facebook:

- 250 million photos are uploaded onto their site every day
- 15 billion images are viewed each day, which equates to 175,000 image views per second

According to YouTube:

- 100 hours of new video content is uploaded every minute
- Six billion hours of video is watched each month, totaling an average of 200 million hours of video watched every day (a 50% increase from the previous year)

According to Cisco Visual Networking Index (VNI):

- Mobile video averages 0.75 exabytes per month and is expected to increase to 7.5 exabytes in 2017
- Mobile web/data averages 0.35 exabytes per month and is expected to increase to 3.0 exabytes in 2017

This technical note focuses on using DDR4 in networking subsystems. It is intended to highlight the main benefits of DDR4 devices, as well as some of the constraints introduced with this new technology, to help system designers maximize the performance of their memory subsystems.

For more information on Micron's DDR4 devices, see the product data sheets.

DDR4 Overview

DDR4 SDRAM is a high-speed dynamic random-access memory internally configured as an 8-bank DRAM for the x16 configuration and as a 16-bank DRAM for the x4 and x8 configurations. The device uses an 8*n*-prefetch architecture to achieve high-speed operation. The 8*n*-prefetch architecture is combined with an interface designed to transfer two data words per clock cycle at the I/O pins.

A single READ or WRITE operation consists of a single 8*n*-bit wide, four-clock data transfer at the internal DRAM core and two corresponding *n*-bit wide, one-half-clock-cycle data transfers at the I/O pins.

This section describes the key features of DDR4, beginning with Table 1, which compares the clock and data rates, density, burst length, and number of banks for the five standard DRAM products offered by Micron.

Table 1: Micron's DRAM Products

	Clock Rate (^{t}CK)		Data Rate				
Product	**Max**	**Min**	**Min**	**Max**	**Density**	**Prefetch (Burst Length)**	**Number of Banks**
SDRAM	10ns	5ns	100 Mb/s	200 Mb/s	64–512Mb	1n	4
DDR	10ns	5ns	200 Mb/s	400 Mb/s	256Mb–1Gb	2n	4
DDR2	5ns	2.5ns	400 Mb/s	800 Mb/s	512Mb–2Gb	4n	4, 8
DDR3	2.5ns	1.25ns	800 Mb/s	1600 Mb/s	1–8Gb	8n	8
DDR4	1.25ns	0.625ns	1600 Mb/s	3200 Mb/s	4–16Gb	8n	8, 16

Density

Micron's first DDR4 product will be a 4Gb device. The JEDEC® standard for DDR4 SDRAM defines densities ranging from 2–16Gb. These higher-density devices enable system designers to take advantage of more available memory with the same number of placements, which can help to increase the bandwidth or supported feature set of a system. It can also enable designers to maintain the same density with fewer placements, which helps to reduce costs.

Prefetch

As shown in Table 1, prefetch (burst length) doubled from one DRAM family to the next. With DDR4, however, burst length remains the same as DDR3 (8). (Doubling the burst length to 16 would result in a x16 device transferring 32 bytes of data on each access, which is good for transferring large chunks of data but inefficient for transferring smaller chunks of data.)

DDR4 devices, like DDR3, offer a burst chop 4 mode (BC4), which is a psuedo burst length of four. Write-to-read or read-to-write transitions get a small timing advantage from using BC4 compared to data masking on the last four bits of a burst length of 8 (BL = 8) access; however, other access patterns do not gain any timing advantage from this mode.

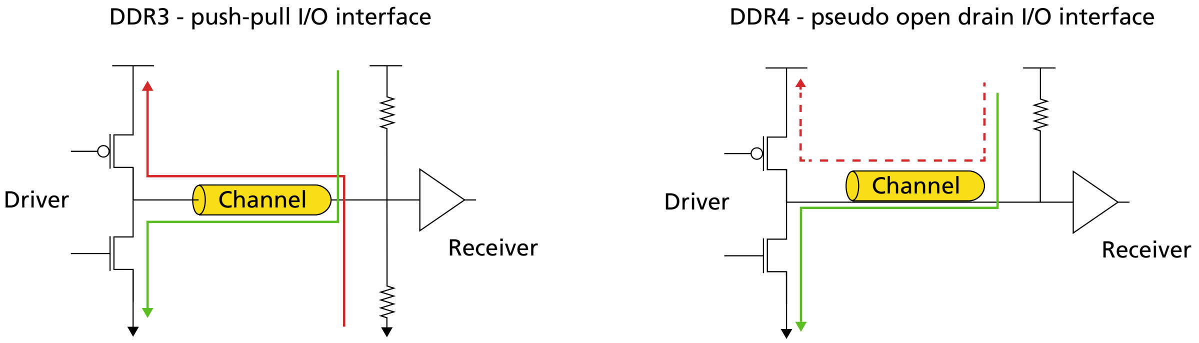
Frequency

The JEDEC DDR4 standard defines clock rates up to 1600 MHz, with data rates up to 3200 Mb/s. Higher clock frequencies translate into the possibility of higher peak bandwidth. However, unless the timing constraints decrease at the same percentage as the clock rate increases, the system may not be able to take advantage of all possible bandwidths. See DRAM Timing Constraints (page 11) for more information

I/O Interface

DDR4 uses a new I/O interface specification: Pseudo Open Drain 1.2V (POD12). This JEDEC specification, new to the DRAM family, is a proven interface specification used by GDDR5. (GDDR5 devices have had success operating at frequencies even higher than those defined for DDR4.) This I/O scheme helps with signal integrity issues associated with higher data rates and enables some power-saving modes. The figure below shows the difference between a DDR3 push-pull I/O interface and a DDR4 psuedo open drain I/O interface.

Figure 1: DDR3 and DDR4 I/O Interfaces


DDR3 - push-pull I/O interface
Driver
Channel
Receiver
DDR4 - pseudo open drain I/O interface
Driver
Channel
Receiver

Error Detection and Data Bus Inversion

Devices that operate at higher clock and data rates make it possible to get more work done in a given period of time. However, higher frequencies also make it more complex to send and receive information correctly. As a result, DDR4 devices offer:

- Two built-in error detection modes: cyclic redundancy cycle (CRC) for the data bus and parity checking for the command and address bits.
- Data bus inversion (DBI) to help improve signal integrity while reducing power consumption.

CRC Error Detection

CRC error detection provides real-time error detection on the DDR4 data bus, improving system reliability during WRITE operations. DDR4 uses an 8-bit CRC header error control: X^8+X^2+X+1 (ATM-8 HEC).

High-level, CRC functions:

- DRAM generates checksum per write burst, per DQS lane
  - 8 bits per write burst (CR0–CR7)
  - CRC using 72 bits of data (unallocated transfer bits are 1s)
- DRAM compares against controller checksum; if two checksums do not match, DRAM flags an error, as shown in Figure 2
- A CRC error sets a flag using the ALERT_n signal (short low pulse; 6–10 clocks)

Figure 2: CRC Error Detection

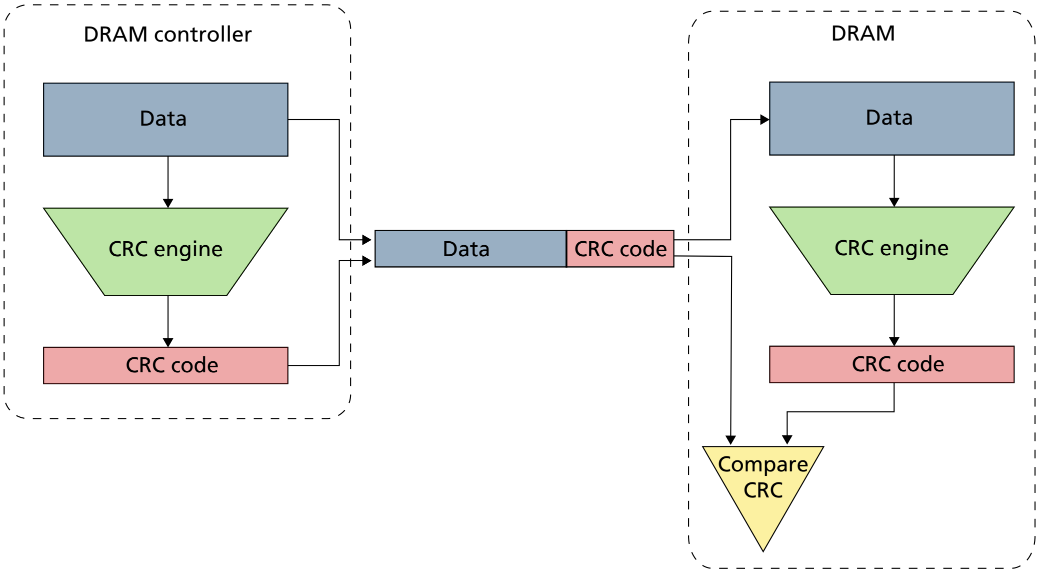


Table 2: CRC Error Detection Coverage

Error Type	Detection Capability
Random single-bit errors	100%
Random double-bit errors	100%
Random odd count errors	100%
Random multi-bit UI error detection (excluding DBI bits)	100%

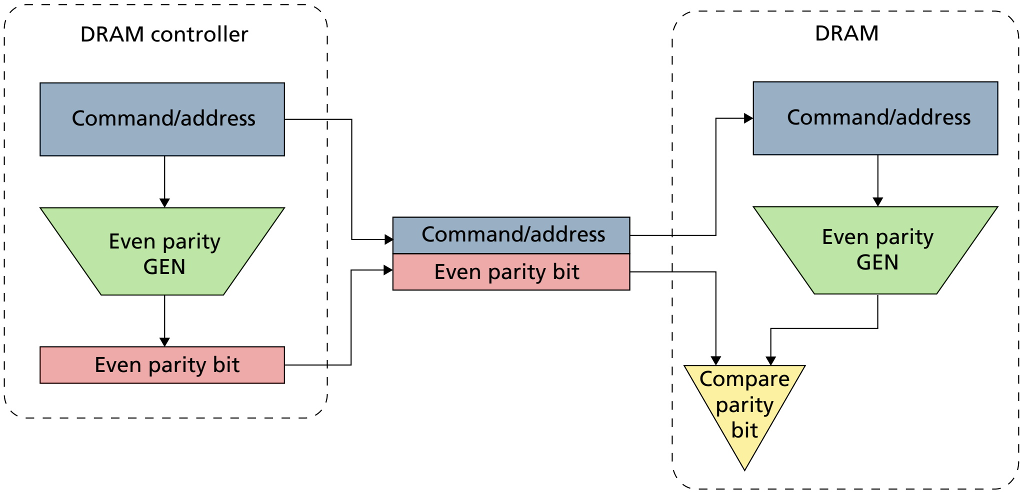
Parity Error Detection

Command/address (CA) parity takes the CA parity signal (PAR) input carrying the parity bit for the generated address and command signals and matches it to the internally generated parity from the captured address and command signals.

High-level, parity error-detection functions:

- CA parity provides parity checking of command and address buses
  - ACT_n, RAS_n, CAS_n, WE_n and the address bus
  - Control signals CKE, ODT, CS_n are not checked
- CA parity uses even parity; the parity bit is chosen so that the total number of 1s in the transmitted signal—including the parity bit—is even
- The device generates a parity bit and compares with controller-sent parity; if parity is not correct, the device flags an error, as shown in Figure 3
- A parity error sets a flag using the ALERT_n signal (long low pulse; 48–144 clocks)

Figure 3: Command/Address Parity Operation



Data Bus Inversion

The data bus inversion (DBI) feature, new to DDR4, is supported on x8 and x16 configurations only (x4 is not supported). The DBI feature shares a common pin with the data mask (DM) and TDQS functions. The DBI feature can apply to both READ and WRITE operations. Write DBI cannot be enabled at the same time the DM function is enabled.

DBI features:

- Opportunistically inverts data bits
- Drives fewer bits LOW (maximum of half of the bits are driven LOW, including the DBI_n pin)
- Consumes less power (power only consumed by bits that are driven LOW)
- Enables fewer bits switching, which results in less noise and a better data eye
- Allows READ and WRITE operations to be enabled separately (controlled by MR5)

DBI is set per-byte:

- One DBI_n pin for x8 configuration
- UDBI_n, LDBI_n pins for x16 configuration

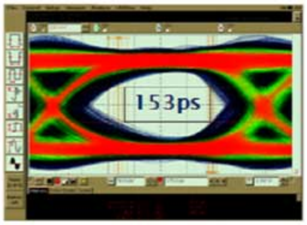
Table 3: DBI Example

Read	Write
If more than four bits of a byte lane are LOW: – Invert output data – Drive DBI_n pin LOW	If DBI_n input is LOW, write data is inverted – Invert data internally before storage
If four or less bits of a byte lane are LOW: – Do not invert output data – Drive DBI_n pin HIGH	If DBI_n input is HIGH, write data is not inverted

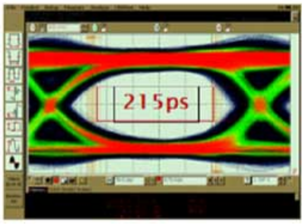
Figure 4: DBI Example

	Controller	Data Bus	Memory
DQ0	0 1 0 0	1 1 0 1	0 1 0 0
DQ1	1 1 0 0	0 1 0 1	1 1 0 0
DQ2	0 0 0 0	1 0 0 1	0 0 0 0
DQ3	0 1 1 0	1 1 1 1	0 1 1 0
DQ4	0 1 0 0	1 1 0 1	0 1 0 0
DQ5	1 0 1 0	0 0 1 1	1 0 1 0
DQ6	1 1 1 0	0 1 1 1	1 1 1 0
DQ7	0 0 1 0	1 0 1 1	0 0 1 0
DBI_n		0 1 1 0	
Number of low bits	5 3 4 8	4 3 4 1	

No DBI



Minimum zeros DBI



Banks and Bank Grouping

DDR4 supports bank grouping:

- x4/x8 DDR4 devices: four bank groups, each comprised of four sub-banks
- x16 DDR4 devices: two bank groups, each comprised of four sub-banks

Figure 5: Bank Groupings—x4 and x8 Configurations

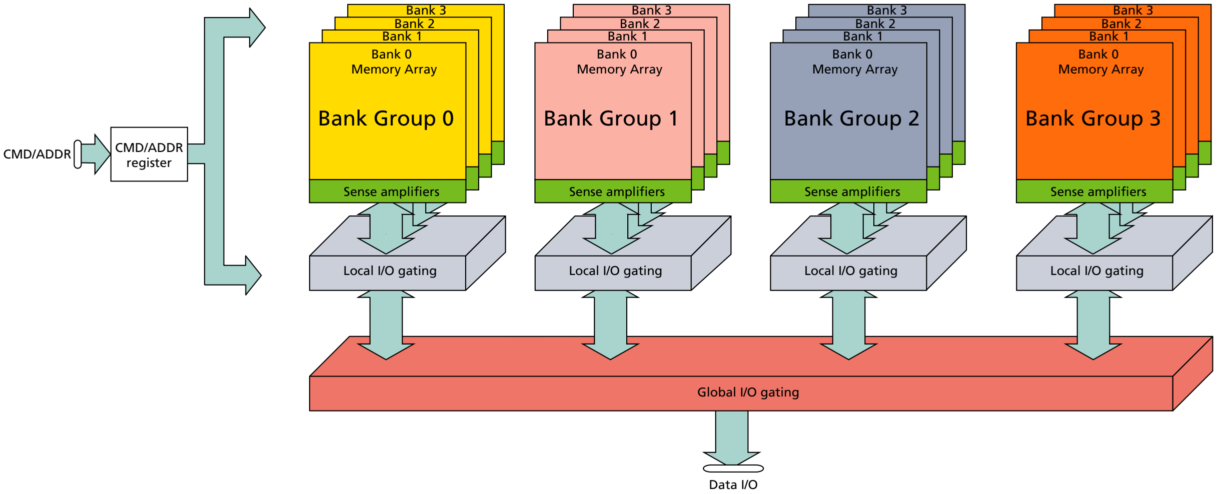
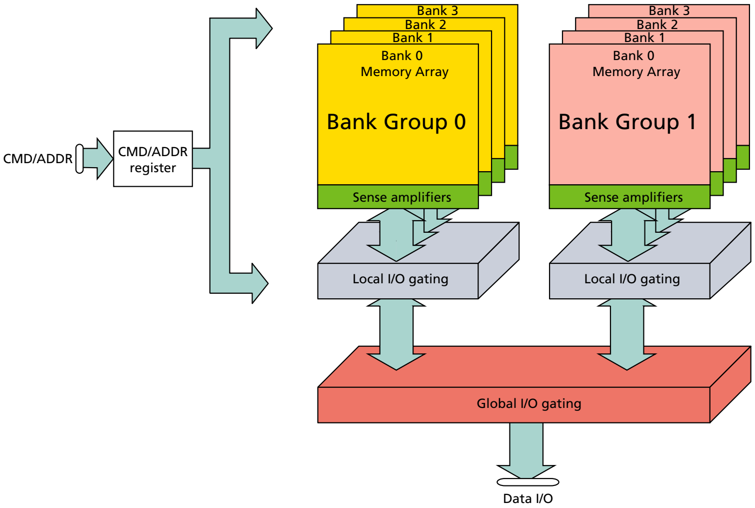


Figure 6: Bank Groupings—x16 Configuration



Bank accesses to a different bank group require less time delay between accesses than bank accesses within the same bank group. Bank accesses to different bank group can use the short timing specification between commands, while bank accesses within the same bank group must use the long timing specifications.

Different timing requirements are supported for accesses within the same bank group and those between different bank groups:

- Long timings (^{t}CCD_L, ^{t}RRD_L, and ^{t}WTR_L): bank accesses within the same bank group
- Short timings (^{t}CCD_S, ^{t}RRD_S, ^{t}WTR_S): bank accesses between different bank groups

Figure 7: Bank Group: Short vs. Long Timing

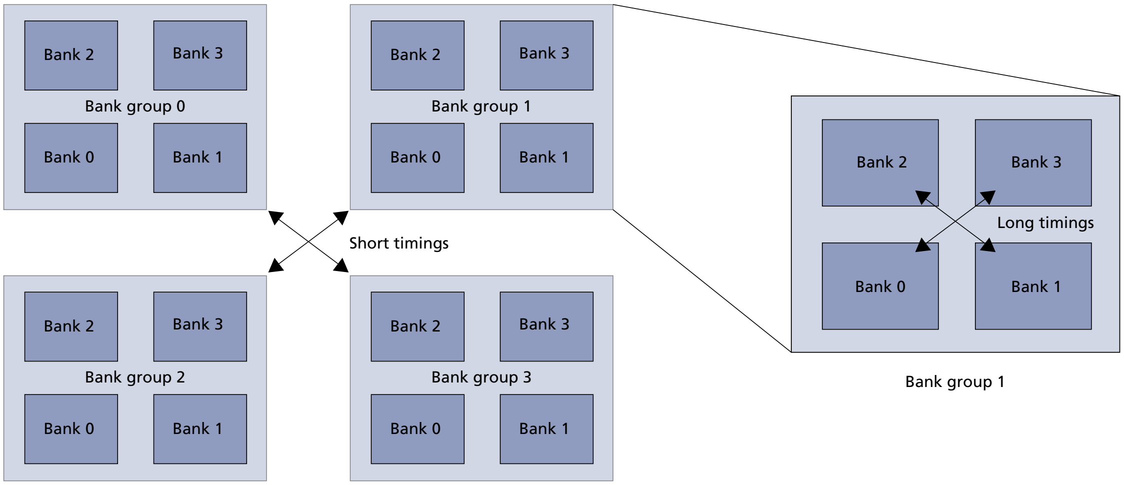


Table 4 summarizes the differences between DDR3 and DDR4 short and long bank-to-bank access timings ^{t}CCD, ^{t}RRD, and ^{t}WTR.

To maximize system performance, it is important that bank-to-bank accesses are to different bank groups. If bank accessing is not controlled properly, it is possible to get less performance with a DDR4-based system versus a DDR3-based system.

Table 4: DDR3 vs. DDR4 Bank Group Timings—^{t}CCD, ^{t}RRD, and ^{t}WTR

Product	Parameter	1600	1866	2133	2400
DDR3	^{t}CCD	4CK	4CK	4CK	N/A
DDR4	^{t}CCD_S	4CK	4CK	4CK	4CK
DDR4	^{t}CCD_L	5CK or 6.25ns	5CK or 5.355ns	6CK or 5.355ns	6CK or 5ns
DDR3	^{t}RRD (1KB)	4CK or 5ns	4 CK or 5ns	4CK or 5ns	N/A
DDR4	^{t}RRD_S (1/2KB, 1KB)	4CK or 5ns	4 CK or 4.2ns	4CK or 3.7ns	4CK or 3.3ns
DDR4	^{t}RRD_L (1/2KB, 1KB)	4CK or 6ns	4CK or 5.3ns	4CK or 5.3ns	4CK or 4.9ns
DDR3	^{t}RRD (2KB)	4CK or 7.5ns	4CK or 6ns	4CK or 6ns	N/A
DDR4	^{t}RRD_S (2KB)	4CK or 6ns	4CK or 5.3ns	4CK or 5.3ns	4CK or 5.3ns
DDR4	^{t}RRD_L (2KB)	4CK or 7.5ns	4CK or 6.4ns	4CK or 6.4ns	4CK or 6.4ns
DDR3	^{t}WTR	4CK or 7.5ns	4CK or 7.5ns	4CK or 7.5ns	N/A

Table 4: DDR3 vs. DDR4 Bank Group Timings—^{t}CCD, ^{t}RRD, and ^{t}WTR (Continued)

Product	Parameter	1600	1866	2133	2400
DDR4	^{t}WTR_S	2CK or 2.5ns	2CK or 2.5ns	2CK or 2.5ns	2CK or 2.5ns
DDR4	^{t}WTR_L	4CK or 7.5ns	4CK or 7.5ns	4CK or 7.5ns	4CK or 7.5ns

Manufacturability

DDR4 has two features that help with manufacturability: multiplexed address pins and connectivity test mode.

Multiplexed Command Pins

To support higher density devices without adding additional address pins, DDR4 defined a method to multiplex addresses on the command pins (RAS, CAS, and WE). The state of the newly defined command pin (ACT_n) determines how the pins are used during an ACTIVATE command.

High-level multiplexed command/address pin functions:

- ACT_n along with CS_n LOW = the input pins RAS_n/A16, CAS_n/A15, and WE_n/A14 used as address pins A16, A15, and A14, respectfully.
- ACT_n HIGH along with CS_n LOW = the input pins RAS_n/A16, CAS_n/A15, and WE_n/A14 used as command pins RAS_n, CAS_n, and WE_n, respectfully for READ, WRITE and other commands defined in the command truth table.

Connectivity Test Mode

Connectivity test (CT) mode is similar to boundary scan testing but is designed to significantly speed up testing of the electrical continuity of pin interconnections between the DDR4 device and the memory controller on a printed circuit board.

Designed to work seamlessly with any boundary scan device, CT mode is supported on all x16 DDR4 devices and on select x4 and x8 DDR4 devices. (JEDEC specifies CT mode for x4 and x8 devices and as an optional feature on 8Gb and above devices.)

Contrary to other conventional shift register-based boundary scan testing, where test patterns are shifted in and out of the memory devices serially during each clock, the DDR4 CT mode allows test patterns to be entered on the test input pins in parallel and the test results to be extracted from the test output pins of the device in parallel. This significantly increases the speed of the connectivity check.

When placed in CT mode, the device appears as an asynchronous device to the external controlling agent. After the input test pattern is applied, the connectivity test results are available for extraction in parallel at the test output pins after a fixed propagation delay time

Table 5: Connectivity Test Mode Pins

Pin Type (CT Mode)	Normal Operation Pin Names
Test Enable	TEN
Chip Select	CS_n
Test Inputs	BA0-1, BG0-1, A0-A9, A10/AP, A11, A12/BC_n, A13, WE_n/A14, CAS_n/A15, RAS_n/A16, CKE, ACT_n, ODT, CLK_t, CLK_c, Parity
	DML_n, DBIL_n, DMU_n/DBIU_n, DM/DBI
	Alert_n
	RESET_n
Test Outputs	DQ0–DQ15, DQSU_t, DQSU_c, DQSL_t, DQSL_c

Logic Equations

The test input and output pins are related by the following equations, where INV denotes a logical inversion operation and XOR a logical exclusive OR operation:

MT0 = XOR (A1, A6, PAR)
MT1 = XOR (A8, ALERT_n, A9)
MT2 = XOR (A2, A5, A13)
MT3 = XOR (A0, A7, A11)
MT4 = XOR (CK_c, ODT, CAS_n/A15)
MT5 = XOR (CKE, RAS_n,/A16, A10/AP)
MT6 = XOR (ACT_n, A4, BA1)
MT7 = x16: XOR (DMU_n / DBIU_n , DML_n / DBIL_n, CK_t)
.... = x8: XOR (BG1, DML_n / DBIL_n, CK_t)
.... = x4: XOR (BG1, CK_t)
MT8 = XOR (WE_n / A14, A12 / BC, BA0)
MT9 = XOR (BG0, A3, RESET_n)

Output Equations for a x16 DDR4 Device:

DQ0 = MT0
DQ1 = MT1
DQ2 = MT2
DQ3 = MT3
DQ4 = MT4
DQ5 = MT5
DQ6 = MT6
DQ7 = MT7
DQ8 = INV DQ0
DQ9 = INV DQ1

DQ10 = INV DQ2
DQ11 = INV DQ3
DQ12 = INV DQ4
DQ13 = INV DQ5
DQ14 = INV DQ6
DQ15 = INV DQ7
LDQS_t = MT8
LDQS_c = MT9
UDQS_t = INV LDQS_t
UDQS_c = INV LDQS_c

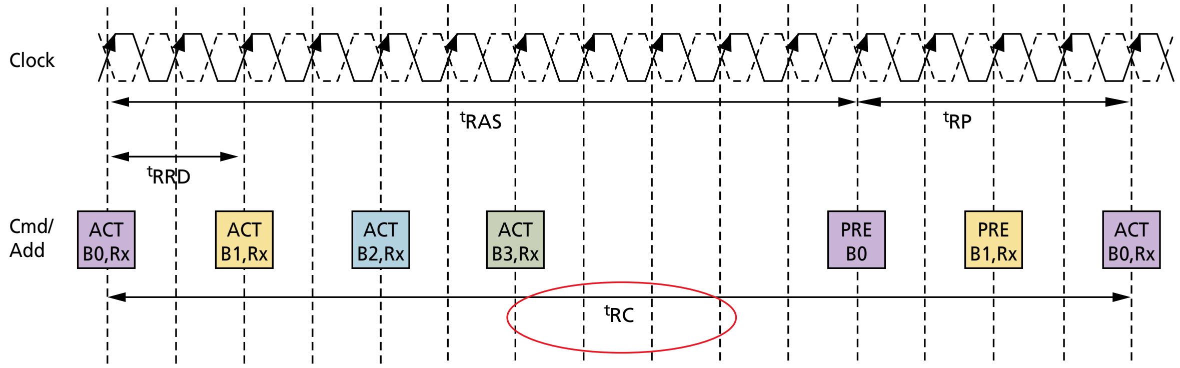
DRAM Timing Constraints

Memory devices have timing parameters that can impact system performance. This section describes the parameters that cause the most significant constraints. The next section provides suggestions for designing around them.

Row Cycle Time

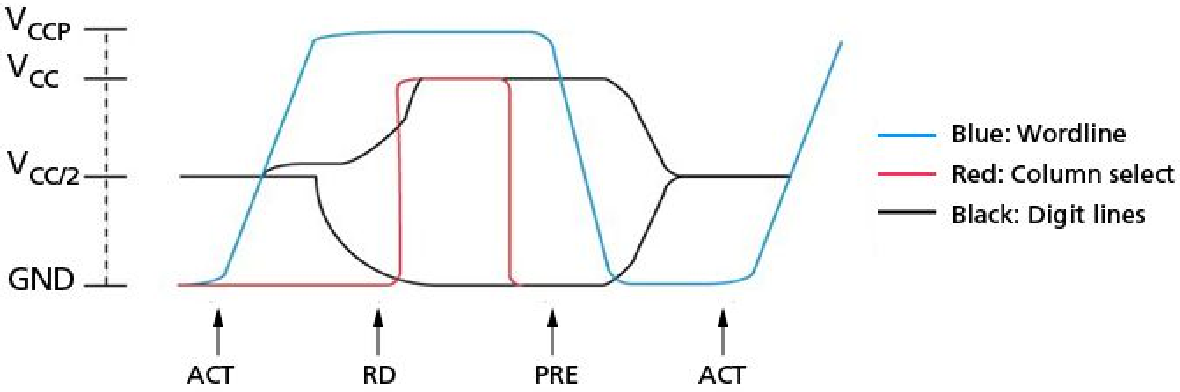
Row cycle time (^{t}RC) is the ACTIVATE-to-ACTIVATE command period within the same bank (that is, the period of waiting time after a row is activated in a bank before another row can be activated in the same bank). For DDR4, ^{t}RC is approximately 44–50ns.

Figure 8: ^{t}RC Timing



^{t}RC is a random-access constraint and a product of the DRAM 1T-1C storage cell architecture. All DRAM devices have a delay based upon the RC time constant associated with separating the digit lines, finishing the requested operation, and bringing the lines back together again.

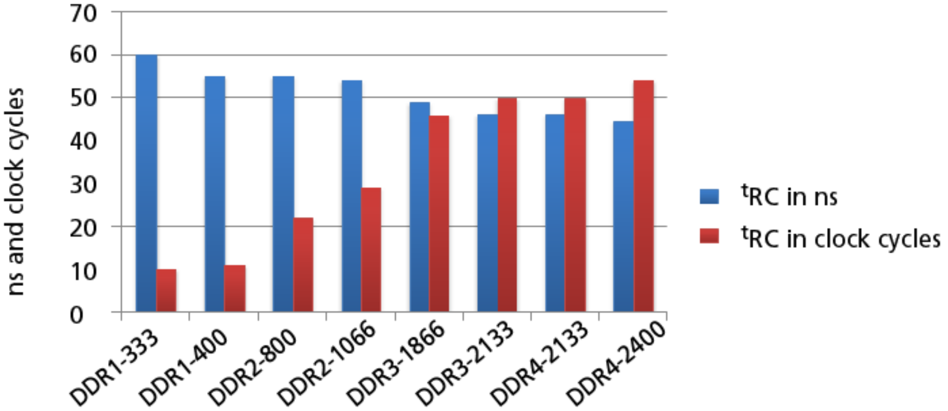
Figure 9: Internal DRAM ^{t}RC Timing



The number of rows connected to the digit lines is a key factor in determining how quickly this can be accomplished. Reducing the number of rows per digit line increases the design complexity, the die size, and the power consumption. Based upon the data in Figure 10, ^{t}RC has only decreased approximately 25% (60ns to 45ns) from DDR1 to DDR4. This is a relatively small reduction compared to the approximate 700% decrease in ^{t}CK (6ns to 0.83ns).

Note: Micron produces memory devices that support a significantly shorter ^{t}RC, known as reduced latency DRAM (RLDRAM®). For more information about RLDRAM memory, send a request to rldram_product_requests@micron.com.

Figure 10: ^{t}RC Across DDR Families



Activation Delay and Four-Bank Activation Window

Activation delay (^{t}RRD) and four-bank activation window (^{t}FAW) are internal power busing constraints. These parameters were created to limit access to the internal DRAM bus to keep die size small.

^{t}RRD is the amount of time that must elapse from one ACTIVATE command to the next ACTIVATE command. For DDR4 devices, the timing specification to use depends upon two factors: the configuration of the device (x4, x8, or x16) and whether the accesses stay within a bank group (long timing—^{t}RRD_L) or go between two different bank groups (short timing—^{t}RRD_S). Each device configuration has a different page size associated with it. The page size of a x4 device is 1/2KB; the page size of a x8 device is 1KB; the page size of a x16 device is 2KB.

^{t}FAW (four-bank activation window) is the amount of time that must elapse after four banks have been activated prior to the activation of another bank. This specification is also based on page size. In most cases, ^{t}FAW is greater than the product of ^{t}RRD x 4. For example, DDR4-2400 ^{t}RRD_S (1KB) = the greater of 4CK or 3.3ns and ^{t}FAW (1KB) = the greater of 20CK or 21ns. At the minimum, ^{t}CK = 0.83ns, ^{t}RRD = 4CK, and ^{t}FAW = 26CK.

As shown in Figure 11, B0–B3 can be activated in only 12 clock cycles. If ^{t}RRD was the only constraint, B4 could be activated on the 16th clock cycle. However, ^{t}FAW must also be considered. This requires an additional 10-clock-cycle delay before the fifth bank can be activated.

Figure 11: tRRD and tFAW Timings

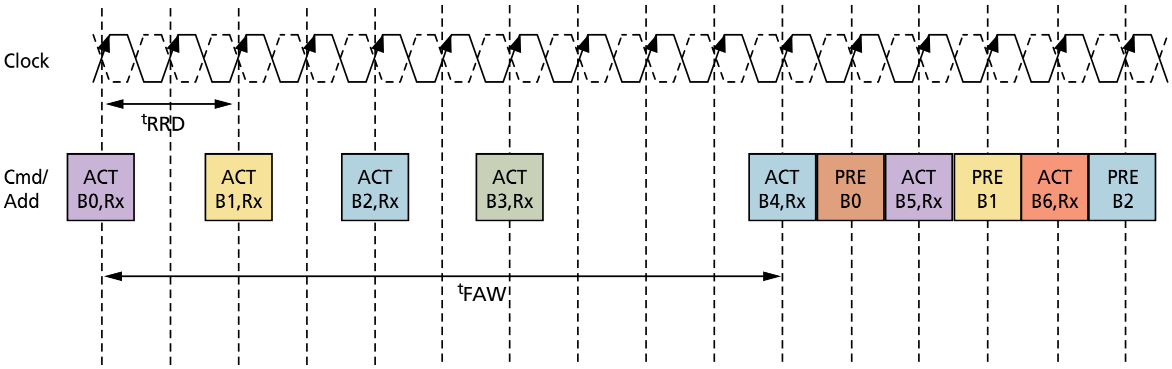


Figure 12: tRRD Across DDR Families

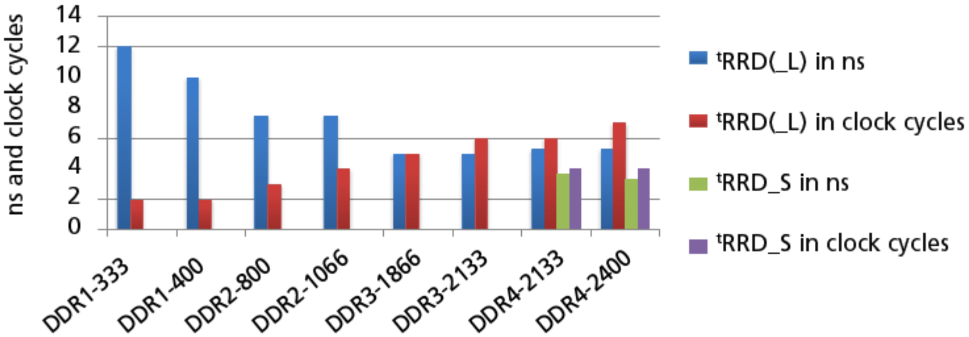
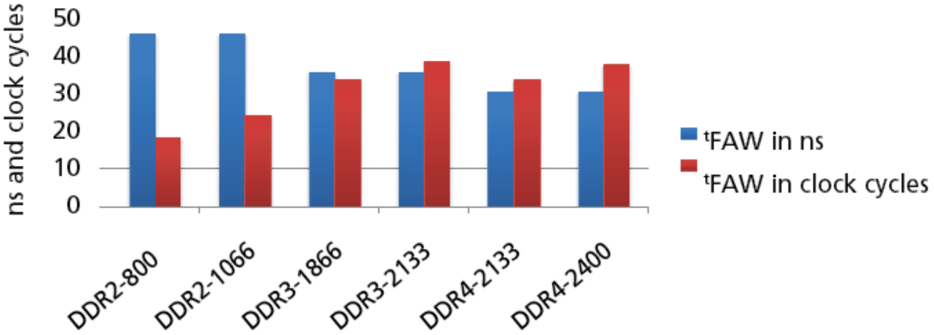


Figure 13: tFAW Across DDR Families

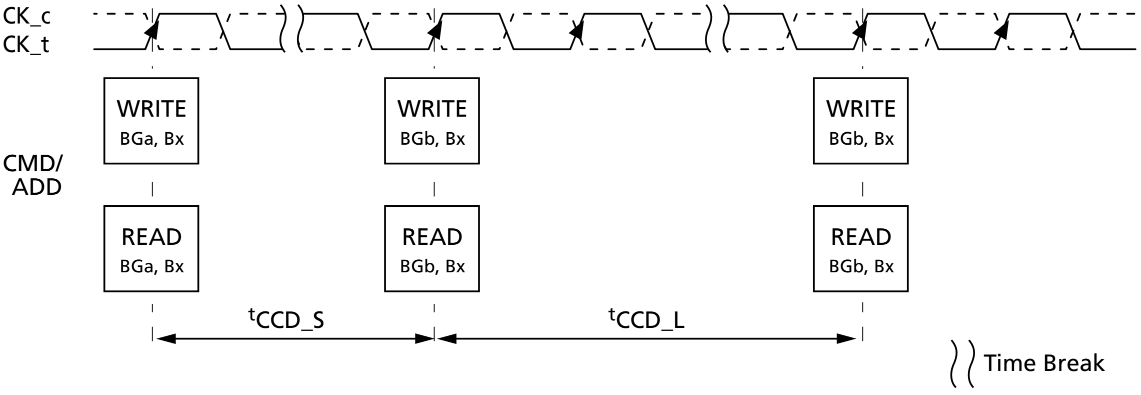


Although ^{t}RRD and ^{t}FAW have decreased more than ^{t}RC, they are still behind the decrease in ^{t}CK. Additionally, the ^{t}RRD_L value for DDR4 is greater than the value of ^{t}RRD for DDR3. DDR4 system designers who do not properly utilize bank grouping will negatively impact their system performance.

Column Access Delay

Column access delay (^{t}CCD) is the delay from one column access (WRITE or READ operation) to the next like-column access. Historically, the number of clock cycles associated with this specification has been simply the burst length divided by 2 (BL/2). The DDR4 definition for ^{t}CCD_s (short timing) follows the historical trend and is BL/2; however, ^{t}CCD_L (long timing) is greater and varies depending on the speed grade.

Figure 14: ^{t}CCD Timing



Write-to-Read Turn Around Time

Write-to-read turn around time (^{t}WTR) specifies how much time must elapse from the last bit of the write data burst prior to a READ command being issued. (^{t}WTR does not specify the total amount of time from a WRITE command to a READ command from a command bus perspective.)

^{t}WTR is necessary because DRAM devices use bidirectional data buses. Common data bus devices have reduced pin count, enabling smaller die size and reduced cost. However, sharing a common data bus for both READ and WRITE operations results in some constraints. When transitioning from a WRITE command to a READ command, the device must ensure that data has been fully written to the array and that the internal data bus is available before read data is placed on it. If this specification is not met, the device places the read data on the internal bus, while still trying to complete a WRITE operation. This causes data corruption.

The total number of clock cycles from a WRITE command to a READ command is WL + BL/2 + ^{t}WTR (in clock cycles). For example, the time required from a WRITE command to the next allowed READ command within the same bank group for a DDR4-083E (^{t}CK = 0.83ns) device is 25 clocks (WL = 12, BL = 8, ^{t}WTR_L = 9 clocks [7.5ns/0.83ns]). The short timing ^{t}WTR_S (as shown in Figure 14) can be used when the WRITE command is in a different bank group than the READ command. If the commands are within the same bank group, ^{t}WTR_L must be used.

Figure 15: ^{t}WTR Timing

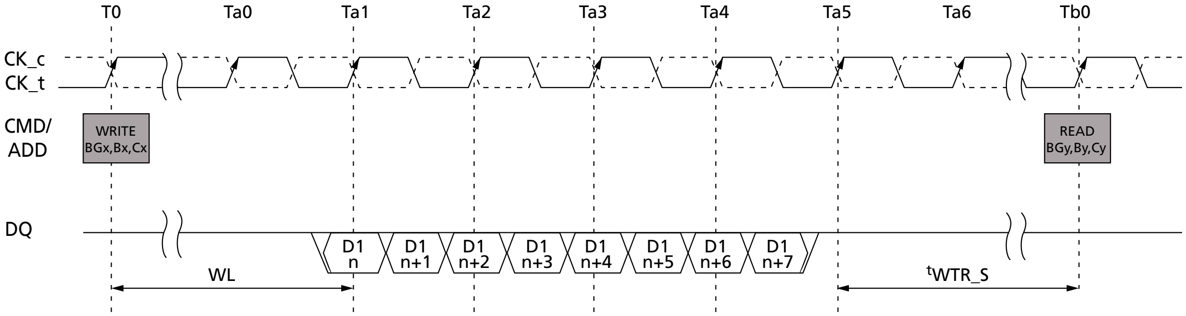
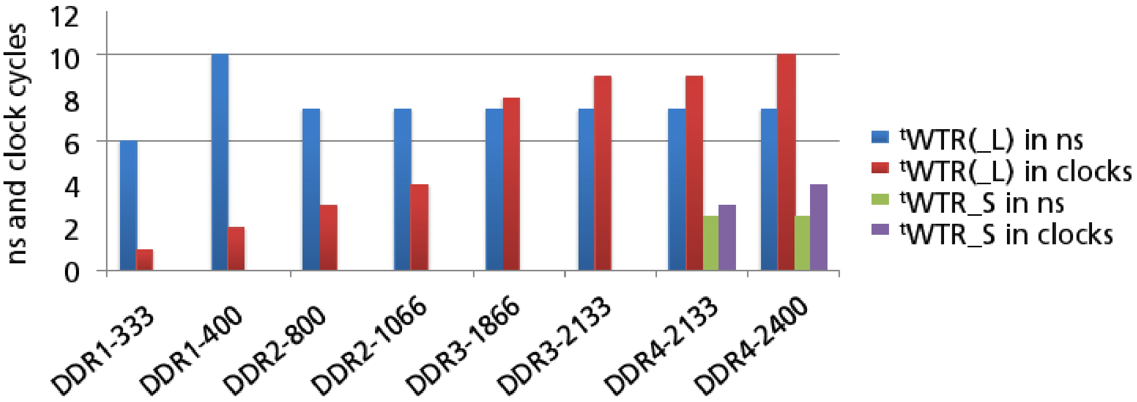


Figure 16: ^{t}WTR Across DDR Families



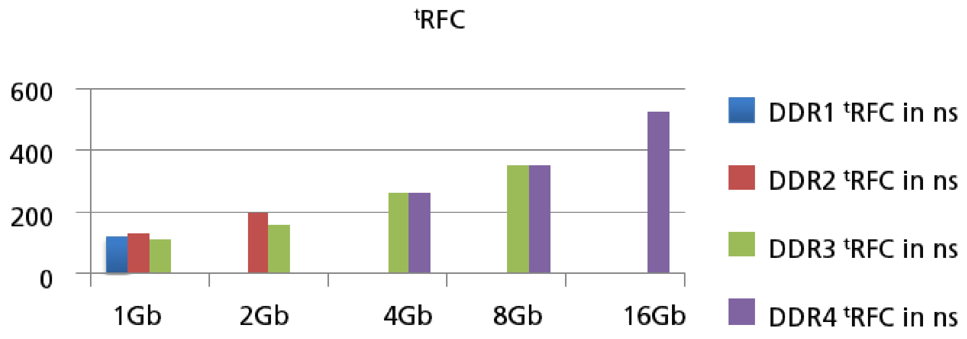
Refresh Cycle Time

All DRAM devices must be refreshed at a periodic rate to maintain the information being stored. A common refresh rate is 64mS, when the temperature of the device is at 85ºC or below. The refresh count of 8K has not changed across DRAM families, even though the densities of the devices continue to grow with each new DRAM family.

To keep every DRAM cell properly refreshed, more physical rows must be accessed during each REFRESH cycle. Accessing more rows takes extra time. ^{t}RFC is the amount of time it takes to access all of the rows in each bank for each REFRESH cycle. After a REFRESH command is issued, there is a minimum delay (^{t}RFC) that must be satisfied before any other command may be issued. ^{t}RFC continues to get larger with each density increase, while ^{t}CK continues to decrease. This is causing REFRESH cycles to steal a noticeable amount of bandwidth from systems. For example, an 8Gb DDR4 device running with a 1200 MHz clock (^{t}CK = 0.83ns) at 85°C requires a periodic refresh cycle once every 7.8µs or once every 9360 clock cycles. A REFRESH cycle takes 350ns or 420 clock cycles. This consumes 4.5% of the device's total number of available clock cycles. A

16Gb device with an anticipated ^{t}RFC value of approximately 525ns will consume 6.7% of the clock cycles under these same conditions.

Figure 17: ^{t}RFC Across DDR Families



Design Considerations

This section provides guidance for working around the timing constraints discussed in the previous section. Three examples from common networking applications are discussed: look-up tables, statistics counters, and packet buffers. Because DDR4 devices require an ACTIVATE and a PRECHARGE command to successfully complete a READ or WRITE operation, any access described in the examples could be an ACTIVATE, READ, or WRITE command.

Example: Look-Up Tables

Look-up tables (LUTs) are a write-once, read-many application. From a worst-case modeling scenario, LUTs are typically considered to be a 100% random/read application because if a look-up table did an update once every few seconds, there would be billions of clock cycles between each update. This would allow for hundreds of millions of reads between each write.

System goal: Use 4Gb DDR4-2400 (^{t}CK = 0.83ns) devices to support 300 million accesses per second (MAPS).

DDR4 architecture: DDR4 uses an 8*n*-prefetch (BL = 8) architecture. To avoid contention on the data bus, the minimum separation from one access to the next must be four clock cycles (BL/2). Using BC4 mode for this access pattern does not provide any timing benefit. This restricts the command bus to one access every four clock cycles, or 300 MAPS (^{t}CK = 0.83ns). In this theoretical scenario, the data bus is 100% utilized.

DDR4 timing constraints: The worst-case scenario for a random read application is accessing data stored in the same bank on every access. Accessing the same bank requires ^{t}RC to be satisfied between accesses. DDR4-2400 (-083E) ^{t}RC is 45.32ns or 55 clocks at 0.83ns ^{t}CK. Doing one access every 55 clock cycles allows only 22 MAPS.

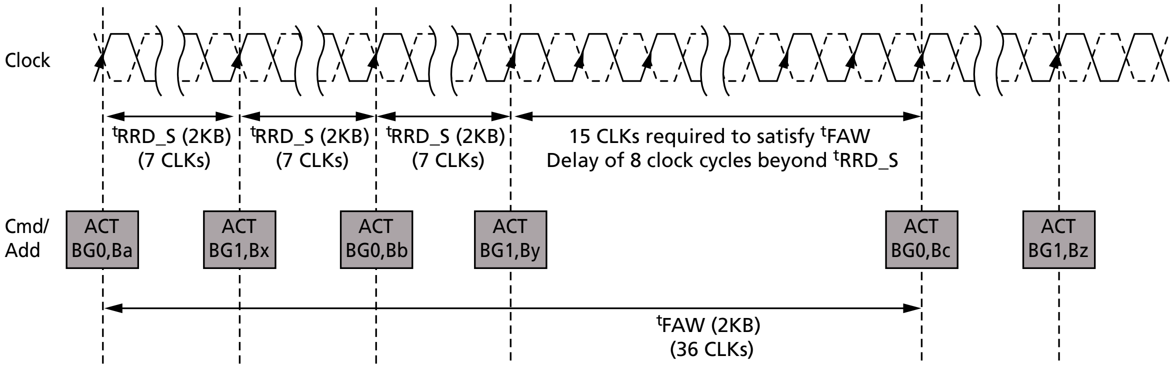
Design considerations: The ^{t}RC delay can be reduced by making copies of the data/table into more than one bank. Having identical copies of the data/table in different banks allows the required information to be retrieved from any of the copies (not limited by ^{t}RC), thus reducing the delay between accesses. The effective ^{t}RC becomes the actual ^{t}RC divided by the number of copies made (in full clock cycles).

DDR4 x4 and x8 devices have 16 banks. Making 16 table copies (one copy in each bank) reduces the ^{t}RC from 45.32ns to 2.8ns, or four clock cycles (45.32/16=2.8, 2.8/0.83 = 3.39, which rounds up to four clock cycles). A delay of only four clock cycles between accesses equates to the theoretical maximum of 300 MAPS. However, some designs will want to utilize x16 devices, which only have eight banks. With only eight banks, only eight copies can be made. The effective ^{t}RC in this case is 5.6ns or seven clock cycles, thus reducing the access rate to 171 MAPS.

In addition to ^{t}RC, other delay constraints (^{t}RRD and ^{t}FAW) must be considered when creating a memory design. The specified value for both of these parameters depends upon the page size, which is dictated by the configuration width of the device to be used (x4 = 1/2KB, x8 = 1KB, x16 = 2KB). In addition to the page size, ^{t}RRD also has a short and a long specification. To take advantage of the short ^{t}RRD timing (^{t}RRD_S), every access must be to a different bank group than the previous access. If accesses are made to the same bank group, the long tRRD timing (^{t}RRD_L) must be used.

A x16 device, which alternates accesses between the two bank groups, would use the following parameters: ^{t}RRD_S(2KB) and ^{t}FAW(2KB). The DDR4-2400 ^{t}RRD_S(2KB) definition is the greater of 4CK or 5.3ns (seven clock cycles with ^{t}CK = 0.83ns). The DDR4-2400 ^{t}FAW(2KB) definition is the greater of 28CK or 30ns (36 clock cycles with ^{t}CK=0.83ns). ^{t}RRD requires the same number of clock cycles as the effective ^{t}RC of the eight-bank device; therefore, ^{t}RRD does not create any additional timing constraints. However, ^{t}FAW causes an eight-cycle delay (see Figure 18), causing the average access rate to drop to one access every nine clock cycles or 133 MAPS.

Figure 18: ^{t}FAW Constraint—x16 (^{t}CK = 0.83ns) Single Device


Clock
^{t}RRD_S (2KB) (7 CLKs)
^{t}RRD_S (2KB) (7 CLKs)
^{t}RRD_S (2KB) (7 CLKs)
15 CLKs required to satisfy ^{t}FAW
Delay of 8 clock cycles beyond ^{t}RRD_S
Cmd/ Add
ACT BG0,Ba
ACT BG1,Bx
ACT BG0,Bb
ACT BG1,By
ACT BG0,Bc
ACT BG1,Bz
^{t}FAW (2KB) (36 CLKs)

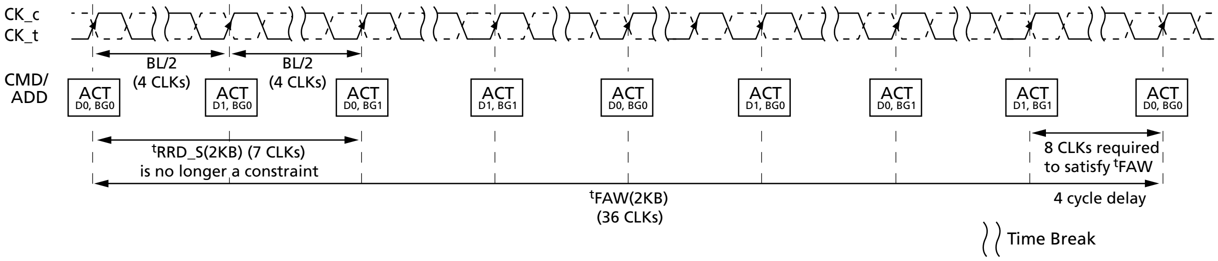
A final consideration is the impact of refresh. A REFRESH command must be issued at an average periodic rate of 7.8µs to maintain the data in all the memory cells. A 4Gb device will reduce available command cycles by approximately 3.3%, or in this specific case, 4.5 MAPS. This reduces the access rate to 128.5 MAPS. (Other methods may be available to reduce or eliminate this constraint; these are system-specific and are not covered in this document.)

Table 6: DDR4 2400 (^{t}CK = 0.83ns) Timing Constraints

	4Gb x4 (1/2KB)	4Gb x8 (1KB)	4Gb x16 (2KB)	8Gb x4 (1/2KB)	8Gb x8 (1KB)	8Gb x16 (2KB)
^{t}RC	45.32ns (55 CLKS)	45.32ns (55 CLKS)	45.32ns (55 CLKS)	45.32ns (55 CLKS)	45.32ns (55 CLKS)	45.32ns (55 CLKS)
Effective ^{t}RC (1 copy/bank)	2.8ns (4 CLKS)	2.8ns (4 CLKS)	5.6ns (7 CLKS)	2.8ns (4 CLKS)	2.8ns (4 CLKS)	5.6ns (7 CLKS)
^{t}RRD_S	3.3ns (4 CLKS)	3.3ns (4 CLKS)	5.3ns (7 CLKS)	3.3ns (4 CLKS)	3.3ns (4 CLKS)	5.3ns (7 CLKS)
^{t}RRD_L	4.9ns (6 CLKS)	4.9ns (6 CLKS)	6.4ns (8 CLKS)	4.9ns (6 CLKS)	4.9ns (6 CLKS)	6.4ns (8 CLKS)
^{t}FAW	13ns (16 CLKS)	21ns (26 CLKS)	30ns (36 CLKS)	13ns (16 CLKS)	21ns (26 CLKS)	30ns (36 CLKS)
Refresh overhead	3.3%	3.3%	3.3%	4.5%	4.5%	4.5%

A single device cannot reach the target of 300 MAPS because of the timing constraints summarized in Table 4. These constraints can be reduced further by using two devices working together in a multirank, shared I/O bus configuration. Ping-ponging back and forth between the two devices reduces all three parameters. The effective ^{t}RC of the two devices working together becomes ^{t}RC/16 (four clocks). Access-to-access timing also improves because from one given access to the next is to a different device. The ^{t}RRD limitation would only apply to every other access, which is to the same device. The minimum time between every other access is limited by burst length (BL/2 * 2 = 8 clocks). This value is greater than the seven clocks needed to satisfy ^{t}RRD; therefore, ^{t}RRD is no longer a constraint. However, the four-bank activate window (^{t}FAW) remains the limiting factor; it allows only eight accesses every 36 clock cycles or 266 MAPS, as shown in Figure 19.

Figure 19: ^{t}FAW Constraints—x16 (^{t}CK = 0.83ns) Two Devices



Completely eliminating the x16 ^{t}FAW constraint requires ping-ponging between three devices, and the desired 300 MAPS can be reached. The ^{t}FAW specification for a x8 device is lower, which makes it possible to reach the 300 MAPS with only two devices. However, the data bus is half as wide, so two additional devices would be required to transfer the same amount of data per access as the x16 device. The decision of whether to use x8 or x16 components is determined based upon how many MAPS are needed, the desired size of each access, and other system considerations.

Example: Statistics Counters

Statistics counters are used for record keeping. They keep track of items such as the total number of packets received, the number of dropped packets, and so on.

Data access patterns for statistic counters are a one-to-one, read-to-write ratio, or what is commonly referred to as read-modify-write. The contents of the counter must first be read. The value is then modified by either incrementing or decrementing it by some value, and then the updated value is written back into the counter.

Some systems may need to maintain hundreds of thousands or even millions of counters. How many and which ones need to be updated depend on the packet type. How often new packets arrive is dependent upon the line rate of the equipment; therefore, the number of counters and how often they get updated is primarily dependent on the capability of the memory technology being used. For example:

- If a 40-byte packet is arriving on a 10 Gb/s link, a new packet is received every 32ns.
- If the arrival of a packet requires two counter updates (each counter update requires two operations: one READ and one WRITE), a memory access once every 8ns is required.

- If the counters being accessed are random, the required 8ns access rate is approximately six times faster than the random access rate (^{t}RC) of a DDR4 device.

In the look-up table example, multiple data copies were made to reduce the value of ^{t}RC. Because counters need to be read and written to, maintaining multiple copies of each counter is not practical; therefore, for this discussion, it will be assumed that counter accesses are not random, and the focus will be on parameters other than ^{t}RC.

DDR4 architecture: DDR4 is a BL = 8 device, similar to DDR3. The DDR4 counter can become extremely large depending on the configuration width selected:

- x4 – BL = 8 -> 32 bits 2^^{32} = 4.29 x 10^{10}
- x8 – BL = 8 -> 64 bits 2^^{64} = 1.84 x 10^{19}
- x16 – BL = 8 -> 128 bits 2^^{128} = 3.40 x 10^{38}

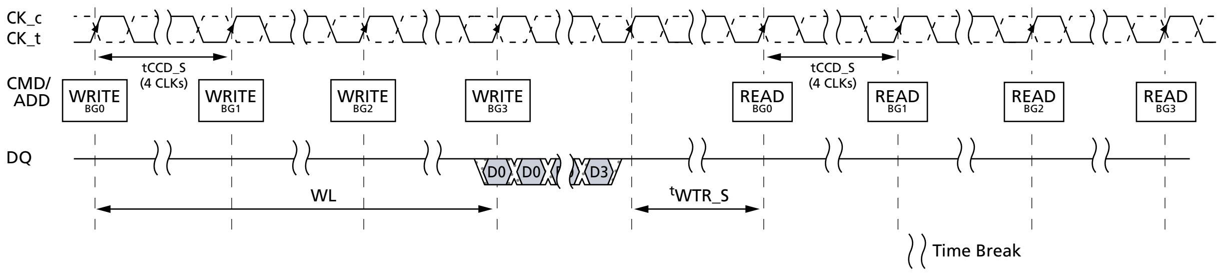
DDR4 timing constraints: If the full counter size is not needed, BC4 mode can be used, which saves two clock cycles on every write-to-read and read-to-write transition. Access-to-access constraints, ^{t}RRD and ^{t}FAW, are also an issue; however, these parameters can be dealt with in a similar manner as shown in the look-up table example.

The major concern is bus turnaround delays. The READ-to-WRITE command delay is RL + BL/2 – WL + 2 * ^{t}CK. For a x16 DDR4-2400 (^{t}CK = 0.83ns) device, this is 10 clock cycles. Although this appears to be a long delay, this only causes two unused clock cycles on the data bus because of the difference in latency between reads and writes.

The WRITE-to-READ command delay is WL + BL/2 + ^{t}WTR [in clock cycles]. For a x16 DDR4-2400 (^{t}CK = 0.83ns) device with both accesses in the same bank group, this is 25 clock cycles. In this case it also causes 25 unused clock cycles on the data bus, allowing two commands every 35 clock cycles, which equates to 68.3 MAPS.

Design considerations: Because the value read from a counter has to travel all the way back to the memory controller or CPU to be modified and then sent back to the memory to be written, a single read-modify-write sequence is actually reading and writing to different counters. This allows several reads and writes to be grouped together, which reduces the number of times the data bus needs to be turned around. If the bank groups on each access are also changed, the bandwidth can be significantly increased (see Figure 20), allowing eight commands every 53 clock cycles, which equates to 150.9 MAPS.

Figure 20: Grouped Writes and Reads to Reduce ^{t}WTR Overhead



Turning the bus around frequently causes severe timing delays due to ^{t}WTR. Accesses to counters can be random. The average number of counter updates per packet is typically in the dozens, and the line rates being designed for are much higher than

10 Gb/s. Taking this into consideration, DRAM devices such as DDR4 are not practical for statistics counters. Counters are typically created using low-latency devices such as RLDRAM or SRAM, which are designed to deal with random accesses, frequent bus turnaround, and high line rates. However, their densities are usually smaller, which forces the counter sizes to be kept small. Some systems may require that the counters be larger than what is practical to store in an RLDRAM or an SRAM. In these cases where the contents of the small counters must periodically be transferred into a larger counter, DDR4 devices would be a good choice for the larger counter implementation.

Example: Packet Buffers

The detailed analysis included in the look-up table and statistic counter examples is not repeated for this packet buffer example. Instead, the goal, the results, and the assumptions used to reach these results are shown below.

System goal: Use 4Gb DDR4-2400 (^{t}CK = 0.833ns) devices to support a 400 Gb/s packet buffer; 800 Gb/s line rate due to ingress and egress.

DDR4 results: A total of 48 x16 devices running at a 2400 Mb/s data rate are required to support the 800Gb/s bandwidth. If ECC is desired, an additional 24 x4 devices are required, for a total of 72 devices, as shown in Figure 21.

Figure 21: DDR4 Packet Buffer Design

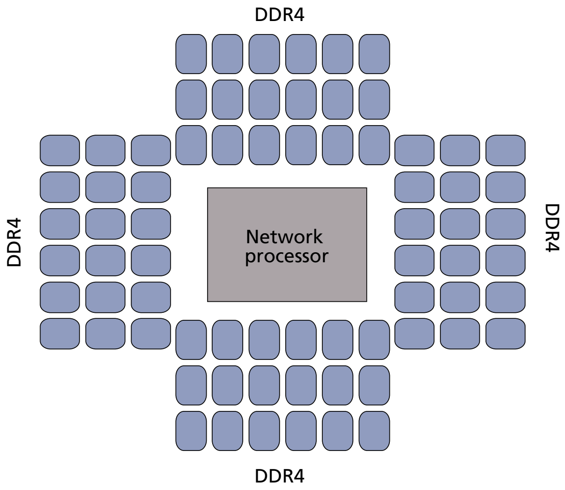


Table 7: DDR4 Packet Buffer Requirements

Parameter	DDR4-2400
Number of memory devices	72 48 x16 / 24 x4 (ECC)
Total number of pins	1896
Power: Host PHY + memory	49W
Memory surface area	10,200mm

DDR4 assumptions:

- 2–4Gb x16 + one 4Gb x4 (ECC) device organized as a 36-bit channel
- 811 Gb/s useful bandwidth (44% of peak bandwidth)
  - 66% data bus utilization (see note below)
  - 67% data efficiency per access (65 bytes/96 bytes access)
- DRAM dimensions:
  - Keepout of 1mm in both X and Y directions
  - x4 = 9 x 11.5mm + keepout = 10 x 12.5mm
  - x16 = 9 x 14mm + keepout = 10 x 15mm
- Power based on:
  - 4Gb device
  - 2400 data rate (0.83ns ^{t}CK)
  - No DBI
  - Approximately 400mW/x16 device
  - Approximately 250mW/x4 device
  - DDR4 memory controller PHY 24W
- Number of pins based on:
  - 1104—I/Os and DQS (40 x 24 + 6 x 24)
  - 792—ADDR/CMD/control/clock pins per 36-bit channel (33 x 24)

Note: The data bus utilization includes the impact of refresh and is based on the following assumptions:

- The access patterns are not random (no ^{t}RC constraint)
- READ/WRITE and ACTIVATE commands are interleaved between bank groups (^{t}CCD_S, ^{t}RRD_S, and ^{t}WTR_S to be used as much as possible)
- Three READ/WRITE operations for each activate
- Reading/writing to four banks between data bus turnaround (reduces the impact of ^{t}WTR)

Changes to any of these assumptions can significantly reduce data bus utilization.

Conclusion

DDR4 features offer the potential for improved system performance and increased bandwidth over DDR3 devices for system designers who are able to properly design around the timing constraints introduced by this technology. The suggestions provided in this technical note mitigating ^{t}RC, ^{t}RRD, ^{t}FAW, ^{t}CCD, and ^{t}WTR can help system designers optimize DDR4 for their memory subsystems.

For system designers who find the increases offered by DDR4 are not enough to provide relief in their memory subsystems, Micron offers a comprehensive line of memory products specifically designed for the networking space. Contact your Micron representative for more information on these products.

Revision History

Rev. A – 02/14

- Initial release

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900
www.micron.com/productsupport Customer Comment Line: 800-932-4992

This data sheet contains minimum and maximum limits specified over the power supply and temperature range set forth herein. Although considered final, these specifications are subject to change, as further product development and data characterization sometimes occur.