TECHNICAL NOTE

GENERAL DDR SDRAM FUNCTIONALITY

INTRODUCTION

The migration from single data rate synchronous DRAM (SDR) to double data rate synchronous DRAM (DDR) memory is upon us. Although there are many similarities, DDR technology also provides notable product enhancements.

In general, double data rate memory provides source-synchronous data capture at a rate of twice the clock frequency. Therefore, a DDR266 device with a clock frequency of 133 MHz has a peak data transfer rate of 266 Mb/s or 2.1 GB/s for a x64 DIMM. This is accomplished by utilizing a 2*n*-prefetch architecture where the internal data bus is twice the width of the external data bus and data capture occurs twice per clock cycle. To provide high-speed signal integrity, the DDR SDRAM utilizes a bidirectional data strobe, SSTL_2 interface with differential inputs and clocks.

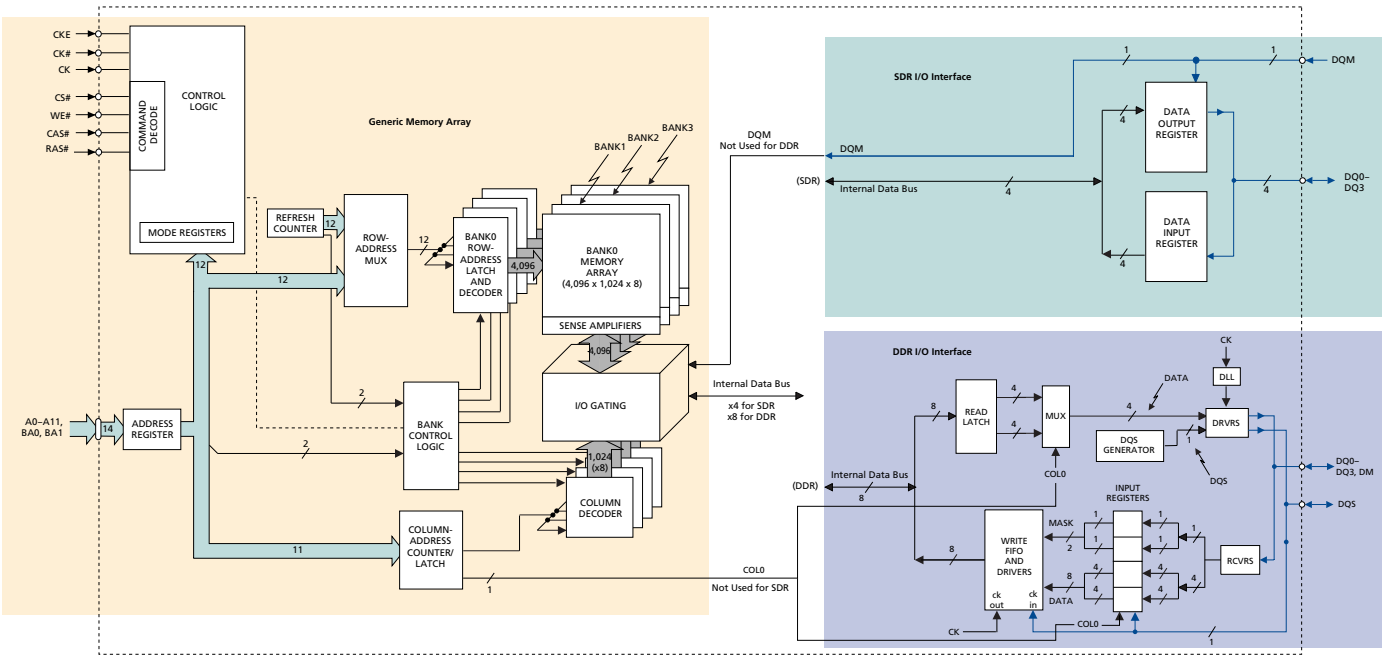
The objective of this technical note is to provide an overview of the 2*n*-prefetch architecture, a strobe-based data bus, and the SSTL_2 interface used with DDR SDRAM. It will also highlight the functional differences between SDR and the improved DDR memory technology. For detailed design and timing criteria for DDR SDRAM-based systems, see Micron's DDR SDRAM data sheets.(http://www.micron.com/ddrsdram.)

Table of Contents

Table 1
SDR to DDR Quick Reference

PARAMETER	SDR	DDR	NOTES
DQM	Yes	No	Used for write data mask and read OE
DM (Data Mask)	No	Yes	Replaces DQM, used to mask write data only
DQS (Data Strobe)	No	Yes	New, used to capture data
CK# (System Clock)	No	Yes	New, DDR utilizes differential clocks
V_{REF}	No	Yes	Reference voltage for differential inputs (1/2 V_{DD})
V_{DD} and $V_{DD}Q$	3.3 Volts	2.5 Volts	Reduced supply and power for DDR
Signal Interface	LVTTL	SSTL_2	DDR utilizes differential I/O
Output Drive	Fixed	Variable	x16 DDR devices offer a reduced drive option
Data Rate	1x Clock	2x Clock	Data transfer is twice the clock rate for DDR
Architecture	Synchronous	Source-Synchronous	DDR utilizes a bidirectional data strobe

Figure 1
Functional Block Diagram
2 Meg x 4 Memory Array with SDR and DDR Interface



DDR VS. SDR FUNCTIONALITY

SDR SDRAM is well established and generally understood, so questions tend to focus where DDR differs from SDR.

An examination of the 32 Meg x 4 SDR and DDR functional block diagrams reveals that the memory core is essentially the same (see Figure 1). Both have an identical addressing and command control interface; both have a four-bank memory array; and both incorporate the same refresh requirements. The fundamental differences are found in the data interface.

The SDR memory data interface is a fully synchronous design where the data is only captured on the positive clock edge. The internal bus is the same width as the external data bus and data latches into the internal memory array sequentially as it passes through the I/O buffers. SDR memory also supports a DQM signal that acts as a data mask during a WRITE operation or an output enable for a READ.

The DDR memory data is a true source-synchronous design, where the data is captured twice per clock cycle with a bidirectional data strobe. This architecture employs a 2*n*-prefetch architecture, where the internal data bus is twice the width of the external bus. This allows the internal memory cell to pass data to the I/O buffers in pairs. With DDR, there is no output enable for READ operations, but DDR does support a BURST TERMINATE command to quickly end a READ in process. During a WRITE operation, the DM signal is available to allow the masking of nonvalid write data.

The DDR command bus consists of a clock enable, chip select, row and column addresses, bank address, and a write enable as shown in Figure 4. Commands are entered on the positive edges of clock, and data occurs on both positive and negative edges of the clock.

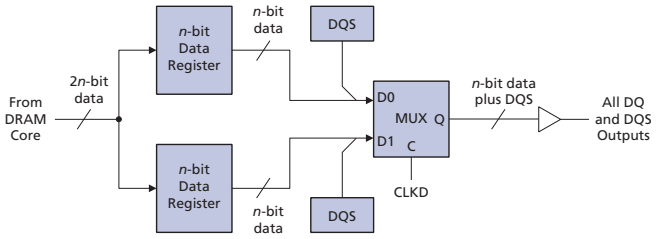
The double data rate memory utilizes a differential pair for the system clock and therefore will have both a true clock (CK) and complementary clock (CK#) signal. The positive clock edge for DDR refers to the point where the rising clock signal crosses with the falling complementary clock signal, and the term negative clock edge indicates the transition of the falling clock and rising complementary clock signals.

2*n*-Prefetch Architecture

The term DDR (or DDRI) should be specifically associated with the 2*n*-prefetch device, as future memory designs (DDRII) will use the 4*n*-prefetch architecture.

To the DRAM vendor, 2*n*-prefetch means that the internal data bus can be twice the width of the external data bus, and therefore the internal column access frequency can be half of the external data transfer rate. That is, for each single read access cycle internal to the device, two external data words are provided (as shown in Figure 2). Similarly, two external data words written to the device are internally combined and written in one internal access (as shown in Figure 3).

Figure 2
Simplified Block Diagram of 2*n*-Prefetch READ



To the user, from a high-level view, 2*n*-prefetch means that data accesses occur in pairs; i.e., a single read access fetches two data words; and for a single write access, two data words (and/or data mask bits) must be provided. This affects both the minimum burst size and nonminimum burst interruptions. The minimum burst size of a 2*n*-prefetch architecture is two external data transfers.

Figure 3
Simplified Block Diagram of 2*n*-Prefetch WRITE

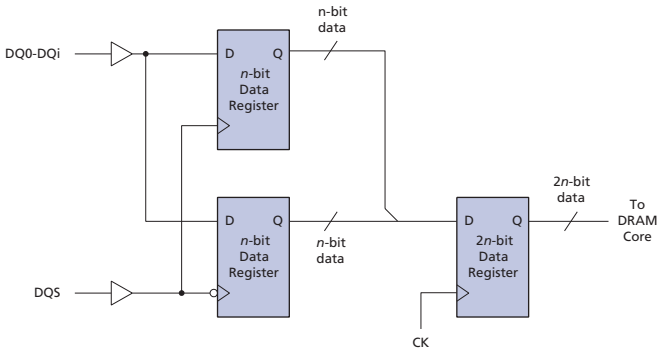
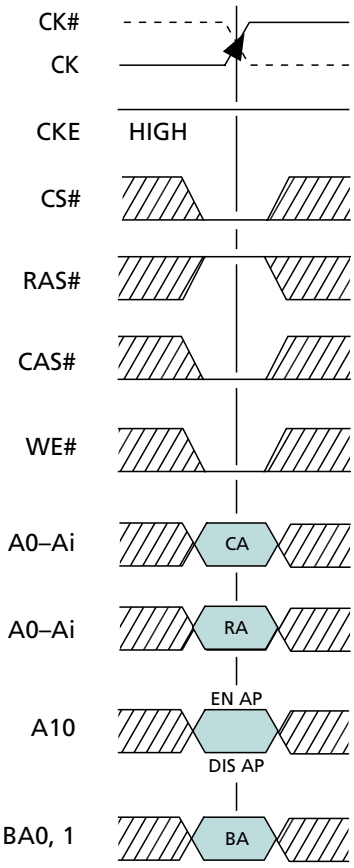


Figure 4
Example of the DDR Command Bus for a WRITE Cycle



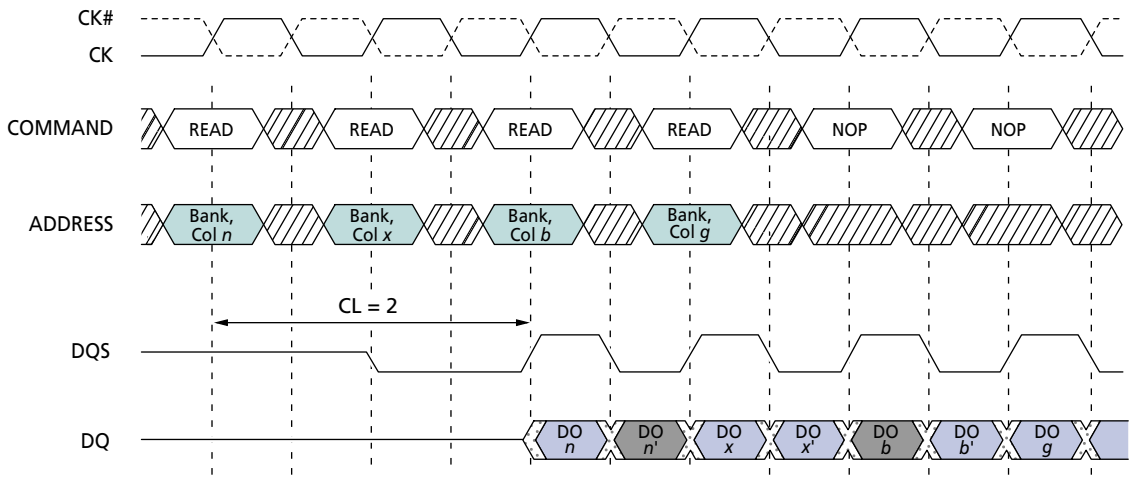
CA = Column Address
RA = Row Address
Ai = Most Significant Address
BA = Bank Address
EN AP = Enable Auto Precharge
DIS AP = Disable Auto Precharge

DON'T CARE

Minimum Time Slots

For READs, the controller can choose to ignore either of the two words, but the time slots for both will be occupied (see Figure 5). Similarly, for WRITEs, the controller can mask either of the two words, but again, the time slots are occupied (see Figure 6). For each READ or WRITE command (and column address) applied, two data words are provided. Because the device is double data rate as well as 2*n*-prefetch, a minimum of two data words is optimal (since commands cannot be applied more frequently).

Figure 5
Minimum Data Time Slot for 2*n*-Prefetch READ



NOTES: 1. DO *n*, etc. = data-out from column *n*, etc.
2. *n*', etc. = the next data-out following DO *n*, etc., according to the programmed burst order.
3. Burst length = 2, 4, or 8 in cases shown.
4. READs are to active rows in any banks.
5. Shown with nominal ^{t}AC and ^{t}DQSQ.
6. The controller wants the first of two words for the first READ command, both words for the second, and the second of two words for the third.

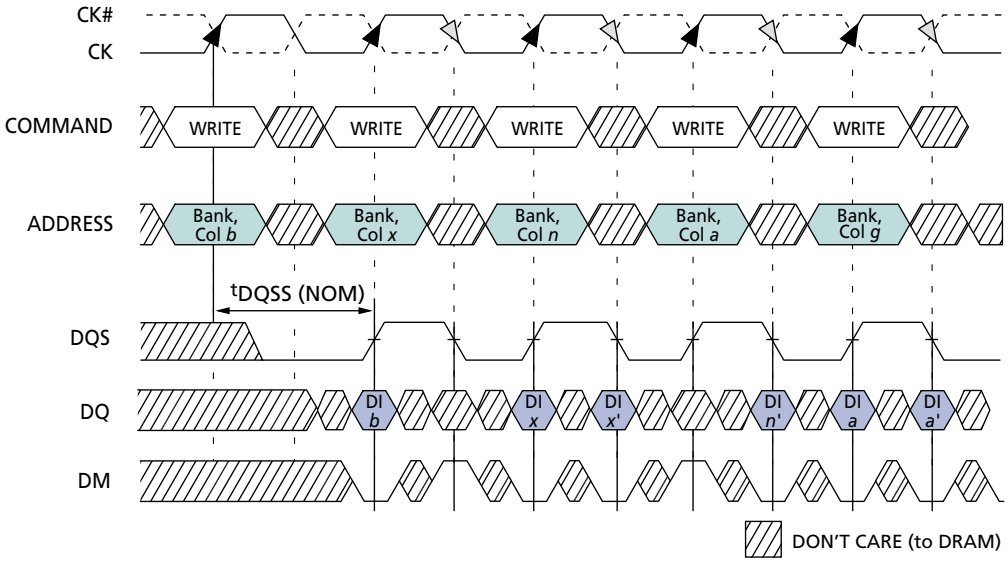
For nonminimum READ bursts (four or eight words), it helps to associate each positive clock edge with a pair of data words. This way, interruptions of READ commands are easily understood. For example, with a burst length of eight, a READ command followed by three uninterrupting commands is needed to access the entire burst. If an interrupting command is applied at the first positive clock edge following the READ command, only two words will be accessed; if an interrupting command is applied at the second positive clock edge following the READ command, only four words will be accessed, etc., (see Figure 7).

The concept of associating pairs of data with positive clock edges applies for WRITEs as well. However, to fully understand the masking and interrupting of write data, WRITE latency and strobe-based data bus timing must be considered. For now it should be noted that the positive clock edges of interest for WRITEs are different than those for READs. This is because of differences in latencies and because the array access occurs at the beginning of a READ operation but at the end of a WRITE operation. In fact, the relevant edges depend on what the interrupting command is, as we will be shown later.

Strobe-Based Data Bus

In a purely synchronous system, data output and capture are referenced to a common, free-running system clock. However, the maximum data rate for such a system is reached when the sum of output access time and flight time approaches the bit time (the reciprocal of the data rate). Although generating delayed clocks for early data launch and/or late data capture will allow for increased data rate, these techniques do not account for the fact that the data valid window (or data eye) moves relative to any fixed clock signal, due to changes in temperature, voltage, or loading. So, to allow for even higher data rates, data strobe signals were added to DDR devices. The data strobes are nonfree-running signals driven by the device, which is driving the data signals (the controller for WRITEs, the DRAMs for READs). At the DRAM device level, for READs, the data strobe (DQS) signals are effectively additional data outputs (DQ) with a predetermined pattern; for WRITEs, the strobe signals are used as clocks to capture the corresponding input data. At the board level, the strobe signals have identical loading to data signals and should be routed similarly.

Figure 6
Minimum Data Time Slot for 2*n*-Prefetch WRITE



NOTE: 1. DI *b*, etc. = data-in for column *b*, etc.
2. *b*', etc. = the next data-in following DI *b*, etc., according to the programmed burst order.
3. Programmed burst length = 2, 4, or 8 in cases shown.
4. Each WRITE command may be to any bank.
5. The controller wants to write the first of two words for the first WRITE command, both words for the second, and the second of two words for the third.

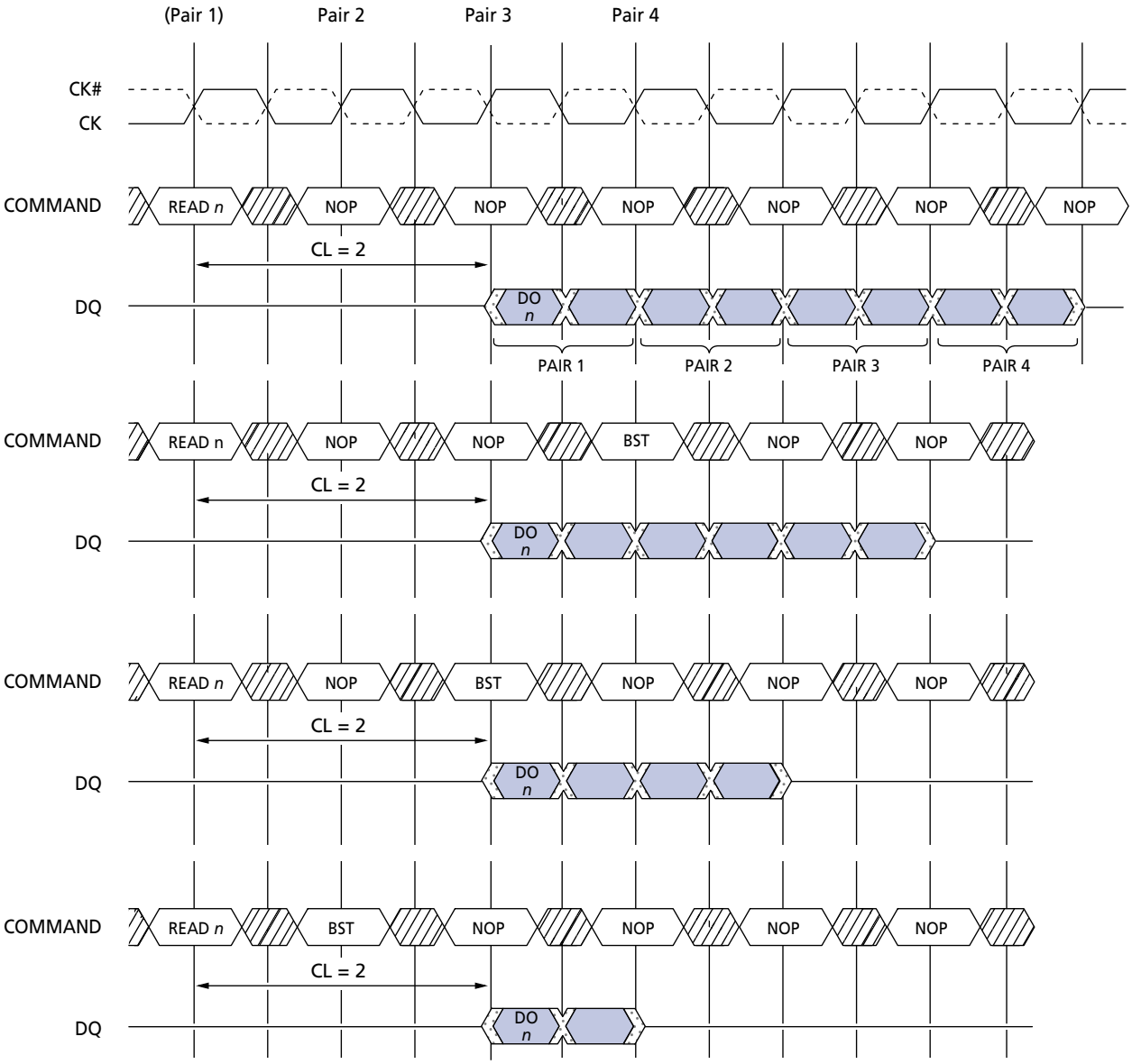
At this point, it may be helpful to digress for a moment to cover the ratio of strobe-to-data signals. This technical note focuses on typical desktop PC main memory applications, which use x64 DIMMs constructed with x8 or x16 DRAM devices. Controllers for these applications will be designed with one strobe per byte; i.e., there is one strobe on each x8 DRAM device and two strobes on each x16 device.

Other types of applications may use different strobe-to-data ratios. For example, servers using x72 DIMMs based on x4 components require controllers with one strobe per four bits, and controllers for graphics or communications applications might use one to four strobes per 32 bits. Creative techniques can be utilized to mix and match different strobe-to-data ratios in a system, but those will not be covered here. Because of the focus of this technical note, a strobe-to-data ratio of one per byte is used. The timing and calculations described apply independently to each group of signals.

For READs, the data strobe signals are edge-aligned with the data signals, meaning that all data and data strobes are clocked out of the device by the same internal clock signal, and all will transition at the outputs at nominally the same time. The controller will internally delay the received strobe to the center of the received data eye.

For WRITEs, the controller must provide the data strobes center-aligned relative to data. That is, strobe transitions occur nominally 90 degrees (relative to the clock frequency) out of phase with data transitions. The DRAM device uses internally matched routing for the strobes and data such that the strobes can be used directly to capture input data. Note that the reason READs and WRITEs use a different alignment scheme is so that the delay circuitry can be centralized in one place (the controller) and does not have to be replicated in every DRAM device in the system. This approach is expected to be carried forward to future generations of DDR to leverage the infrastructure now being established.

Figure 7
Relating Command Slots to Read Data



NOTE: 1. DO *n* = data-out from column *n*.
2. NOP commands represent any valid uninterrupting commands.
3. BST commands represent any valid interrupting commands.
4. Controlling command slots are noted for each pair; if an interrupting command is applied in the controlling command slot for a given data word pair, that pair will not be read out.
5. Pair 1 is always read out (i.e., is the minimum burst) when a READ command is applied.

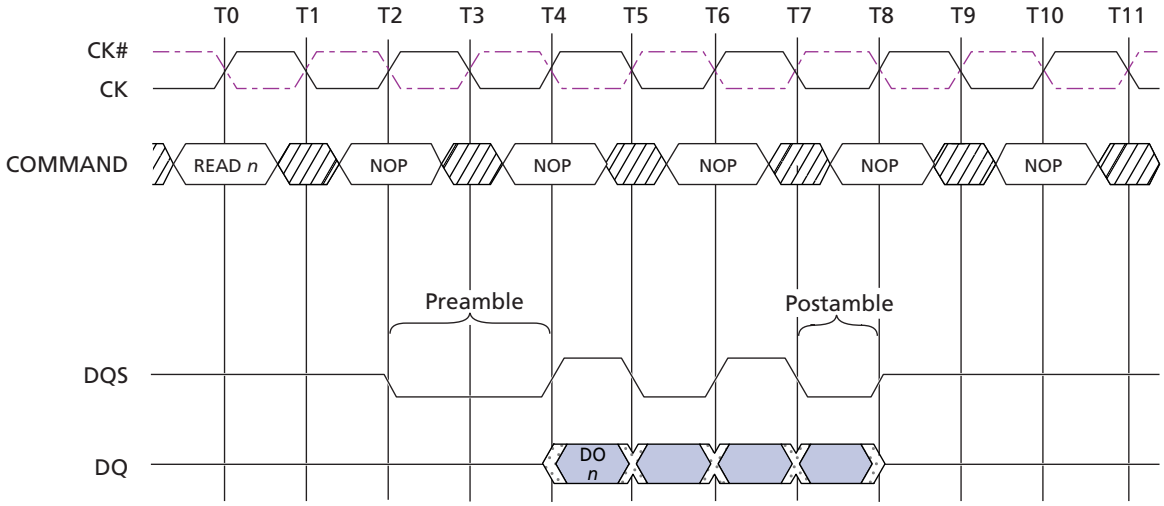
DON'T CARE

Preamble and Postamble

The data strobe timing pattern consists of a preamble, toggling, and postamble portion. Figure 8 shows the strobe pattern and alignment to data for READs, and Figure 9 shows the same for WRITEs. The preamble portion provides a timing window for the receiving device to enable its data capture circuitry while a known/valid level is present on the strobe signal, thus avoiding false triggers of the capture circuit. Following the preamble, the strobes will toggle at the same frequency as the clock signal for the duration of the data burst. Each high transition and each low transition is associated with one data transfer. The low time following the last transition is known as the postamble. Most controllers have an internal clock running at twice the memory clock frequency, so generating a strobe shifted 90 degrees relative to data is fairly straightforward. Typically, data and strobe are clocked out using the 2x clock, with data being driven by edges of one polarity, and the strobe being driven by edges of the opposite polarity. Delaying the incoming strobe for READs is more involved and will be covered in detail in an independent technical note.

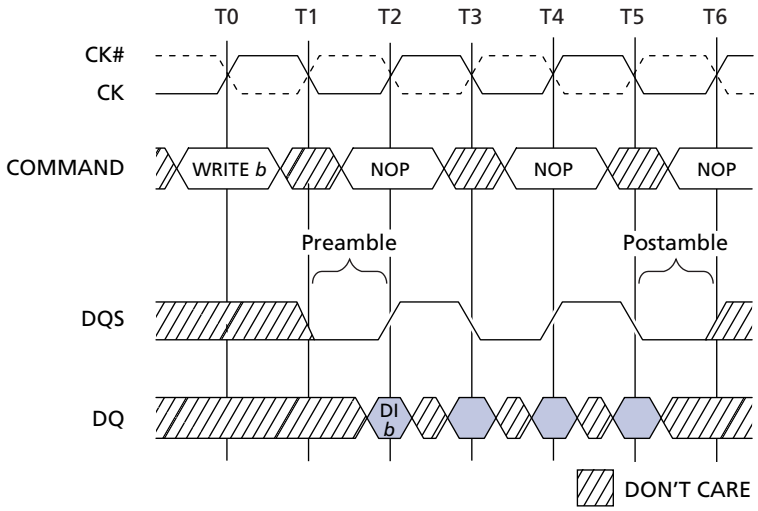
For read-to-read data bus transitions where the READs are from different physical banks of DRAMs, or read-to-write data bus transitions (i.e., transitions from one device driving the data bus to another device driving the data bus), there is a hand-off of strobe signals, and the full strobe pattern (including preambles and postambles) is needed from each source. The conservative approach is to space requests such that the postamble from the first source completes before the preamble from the second source begins. Consecutive READ bursts from the same bank of DRAMs are achieved by extending the toggling portion of the data strobe pattern. (A postamble and preamble are not needed between consecutive READ bursts from the same source.) Consecutive WRITE bursts are also possible, even if to different physical banks of DRAMs. This is a little less intuitive because, unlike the case of consecutive READs from the same source, the destination DRAMs for the second consecutive WRITE burst have no way of knowing that a first WRITE burst to another bank of DRAMs occurred. This means that the DRAMs must be capable of accepting different preamble timing, depending on whether there was prior write activity on the bus.

Figure 8
DQS Pattern for READ Showing Preamble and Postamble



NOTE: 1. DO *n* = data-out from column *n*.
2. Burst length = 4, CAS latency = 2.
3. Three subsequent elements of data-out appear in the programmed order following DO *n*.
4. Shown with ^{t}AC and ^{t}DQSQ = 0 for illustration.

Figure 9
DQS Pattern for WRITE Showing Preamble and Postamble



NOTE:
1. DI *b* = data-in for column *b*.
2. Three subsequent elements of data-in are applied in the programmed order following DI *b*.
3. A burst of four is shown.
4. Shown with nominal ^{t}DQSS and without DM bits for illustration.

SSTL_2 Interface

Previous SDR memory technology used LVTTL and a fixed voltage level for signal interface. DDR SDRAM utilizes differential inputs and a reference voltage for all interface signals. This interface is called SSTL_2, which stands for stub series terminated logic for 2.5 volts. SSTL_2 is an industry standard defined by JEDEC document #EIA/JESD8-9. Although some DRAMs will support a reduced drive output, most will comply with the SSTL_2 Class II drive levels.

Benefits to the SSTL_2 interface include symmetrical low and high logic levels, improved signal integrity, and better noise immunity, as the input levels track minor variations in the supply voltage.

Driver and Receivers

The output buffer and input receivers have changed from LVTTL to SSTL_2. The output buffer logic has not changed, but the $V_{DD}Q$ has moved from 3.3 volts to 2.5 volts for DDR.

The input receivers have migrated from an *n* and *p* channel stacked gate to a differential pair common source amp. The more complex receiver used in DDR provides greater bandwidth and a smaller variation over temperature to increase margin to the tighter input signaling. V_{REF} has been added to improve V_{DD} margin over temperature.

Figure 10
Typical LVCMOS Receiver

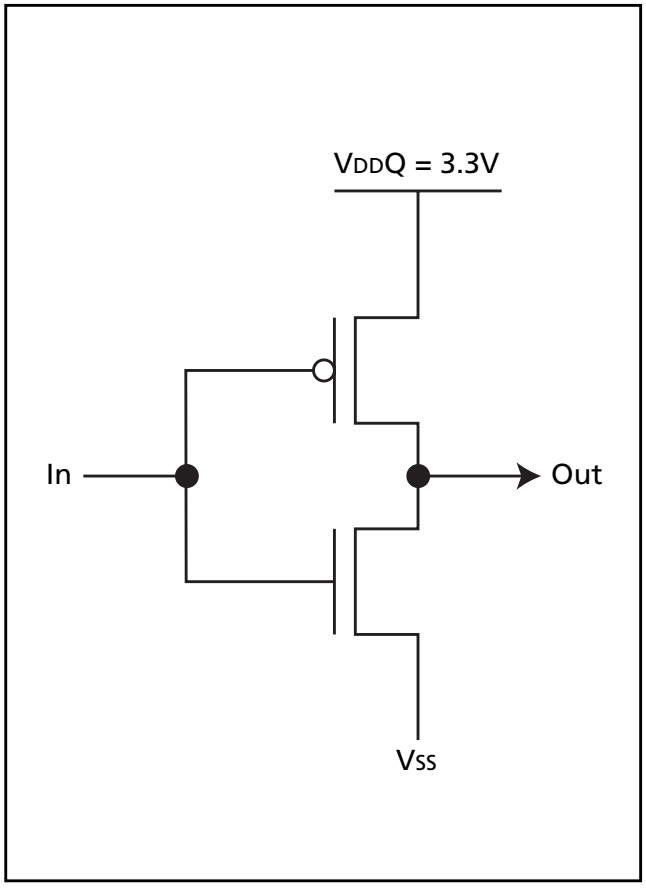
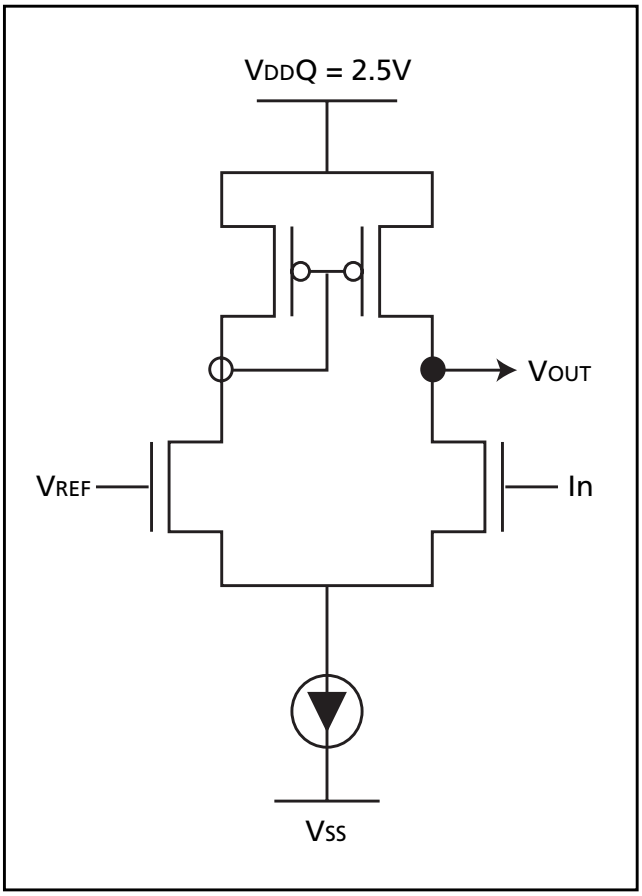


Figure 11
Typical SSTL_2 Receiver

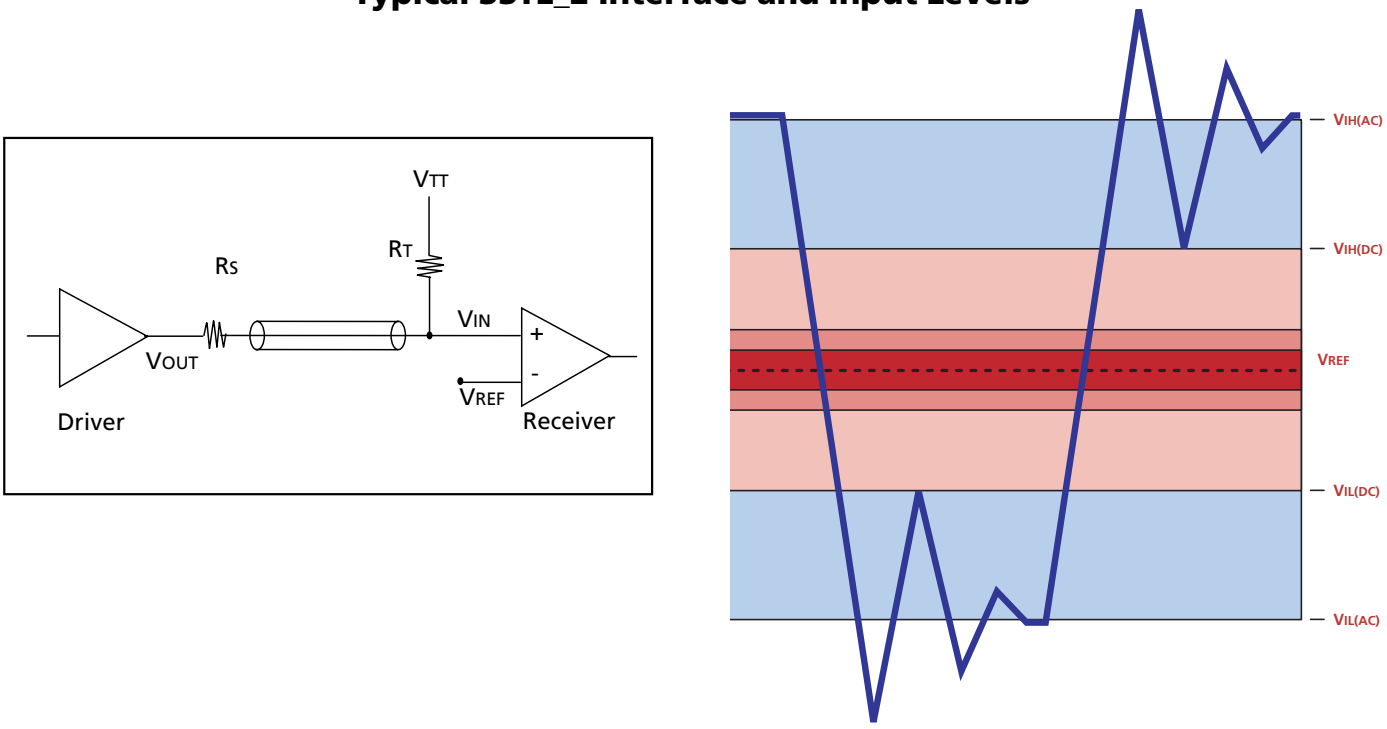


I/O Signaling

The typical SSTL_2 interface includes series termination and a pull-up to the termination voltage (refer to Figure 12). The SSTL_2 interface uses a reference voltage and differential input to determine the logic levels. The reference voltage is defined to be half of the supply voltage and the termination voltage equals the reference voltage. (V_{DD} = 2.5 volts, V_{REF} = V_{TT} = 1.25 volts).

There are both DC and AC input logic levels for the SSTL_2 interface. In general, the DRAM will start to switch to the new logic level when the input signal transitions through the target DC level and it will latch when the input signal crosses through the final AC input level. Once the logic level has been latched, it will remain latched until the input signal transitions back through the DC level. Refer to Figure 12 for a typical SSTL_2 input signal.

Figure 12
Typical SSTL_2 Interface and Input Levels



DC ELECTRICAL CHARACTERISTICS AND OPERATING CONDITIONS

(0°C ≤ T_A ≤ +70°C; V_{DD} = +2.5V ±0.2V, $V_{DD}Q$ = +2.5V ±0.2V)

PARAMETER/CONDITION	SYMBOL	MIN	MAX	UNITS
I/O Reference Voltage	V_{REF}	0.49 x $V_{DD}Q$	0.51 x $V_{DD}Q$	V
I/O Termination Voltage (system)	V_{TT}	V_{REF} - 0.04	V_{REF} + 0.04	V
Input High (Logic 1) Voltage	$V_{IH(DC)}$	V_{REF} + 0.15	V_{DD} + 0.3	V
Input Low (Logic 0) Voltage	$V_{IL(DC)}$	-0.3	V_{REF} - 0.15	V
Input High (Logic 1) AC Voltage	$V_{IH(AC)}$	V_{REF} + 0.310	–	V
Input Low (Logic 0) AC Voltage	$V_{IL(AC)}$	–	V_{REF} - 0.310	V

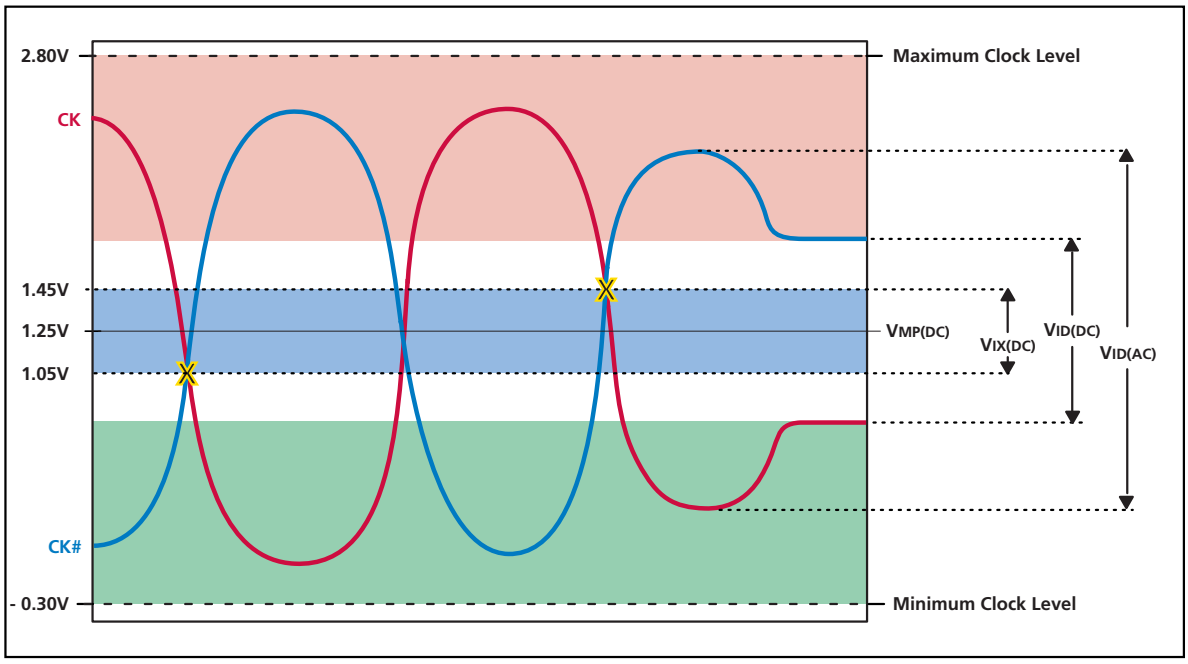
Clock Inputs

To increase accuracy caused by clock jitter, a differential clock was added to DDR. SDR devices use the midpoint of the rising clock edge to latch data and thus are exposed to drift. DDR uses the crossing point of CK and CK#. Using the crossing point instead of the midpoint helps negate the affects of jitter and increase margins. The clocks used for DDR also operate within a set of parameters, which are defined by JEDEC (see Figure 13).

SUMMARY

The similarities between SDR and DDR SDRAM provide the DRAM manufacturer cost advantages and assure high production yields. These similarities also help the designer to better understand DDR and allow the most optimal design techniques from previous designs to be incorporated into the new DDR platforms. Although the addressing schemes, layout requirements, and device configurations are much the same for DDR, the performance gains are remarkable. For example, the power consumption for DDR is significantly less than for a comparable SDR device, yet peak transfer rates can exceed 2.1 GB/s for a standard x64 DDR DIMM.

Figure 13
Typical SSTL_2 Clocks



Reference: *Micron DesignLine*, Volume 8, Issue 3 (3Q99)
DesignLine is available at http:www.micron.com/designline

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