Application Note

Command & Protocol Verification of DDR, DDR2, and DDR3 SDRAM

Keep Pace with More Complex and Shorter Design Cycles

Computer memories are not the only systems that continue to demand larger, faster, lower powered and physically smaller memories. Embedded systems applications have similar requirements. This application note highlights the power of the logic analyzer in verifying DDR, DDR2, and DDR3 SDRAM commands and protocols.

Tektronix®

Introduction

DRAM asynchronous operation caused many design challenges when interfacing to synchronous processors. SDRAM (Synchronous DRAM) was designed to synchronize DRAM operation to the rest of a computer system or embedded system and to eliminate the need to define all the different modes of sequence based memory operations. The technical evolution of SDRAM, such as DDR SDRAM, has progressively increased memory data rate performance. SDRAM, DDR, DDR2, and DDR3 SDRAM are all controlled with memory commands sent by the memory controller host to the memory.

DDR, DDR2 and DDR3 SDRAM

There is a continual demand for memories to be larger, faster, lower powered and physically smaller. These needs are the driving force in the advancement of DRAM technology. Mainstream DRAMs have evolved over the years through several technology enhancements, such as SDRAM (Synchronous DRAM), DDR (Double Data Rate) SDRAM, DDR2 (Double Data Rate 2) SDRAM, and DDR3 (Double Data Rate 3) SDRAM.

DDR (Double Data Rate) SDRAMs increased the memory data rate performance by increasing clock rates, bursting of data and transferring two data bits per clock cycle.

DDR2 SDRAM has several improvements over DDR SDRAM. DDR2 SDRAM clock rates are higher, thus increasing the memory data rates. Signal integrity becomes more important for reliable memory operation as the clock rates increase. As clock rates increase, signal traces on the circuit boards become transmission lines and proper layout and termination at the end of the signal traces becomes more important.

DDR3 SDRAM is a performance evolution and enhancement of SDRAM technology starting at 800 Mb/s. DDR3 SDRAMs support six levels of data rates and clock speeds. DDR3-800/1066/1333 SDRAMs became available in 2007, DDR3-1600 in 2008 and DDR3-1866/2133 SDRAMs are expteced in 2009. DDR3-1066 SDRAM uses less power than DDR2-800 SDRAM because the DDR3 SDRAM operating voltage is 1.5 volts, which is 83% of DDR2 SDRAM's 1.8 volts. Also, the DDR3 SDRAM data DQ drivers are at higher 34 ohms impedance than DDR2 SDRAM's lower 18 ohms impedance. DDR3 SDRAM will start with 512 Mb of memory and will grow to 8 Gb memory in the future. Just like DDR2 SDRAM, DDR3 SDRAM data output configurations include x4, x8 and x16. DDR3 SDRAM has eight banks where as DDR2 SDRAM has four or eight depending upon the memory size.

SDRAM	Data Rate MT/S	Clock MHz	VDD V
DDR-266	266	133	2.5
DDR-333	333	166	2.5
DDR-400	400	200	2.5
DDR2-400	400	200	1.8
DDR2-533	533	267	1.8
DDR2-667	667	334	1.8
DDR2-800	800	400	1.8
DDR2-1066	1066	533	1.8
DDR3-800	800	400	1.5
DDR3-1066	1066	533	1.5
DDR3-1333	1333	667	1.5
DDR3-1600	1600	800	1.5

Table 1. SDRAM Standards

Both DDR2 and DDR3 SDRAMs have four mode registers. DDR2 defined the first two mode registers while the other two were reserved for future use. DDR3 uses all four mode registers. One significant difference is DDR2 mode registers defined CAS latency for read operation and the write latency was one less the mode register read latency setting. DDR3 mode registers have unique settings for both the CAS read latency and write latency. DDR3 SDRAM uses 8n prefetch architecture which transfers 8 data words in 4 clock cycles. DDR2 SDRAM uses 4n prefetch architecture which transfers 4 data words in 2 clock cycles. The DDR3 SDRAM mode registers are programmed to support the on the fly burst chop, which shortens the transfer of 8 data words to 4 data words by setting the address line 12 low during a read or write command. On the fly burst chop is similar in concept to the read and write auto-precharge function of the address line 10 in both DDR2 and DDR3 SDRAMs.

Other noteworthy DDR3 SDRAM attributes include the data strobes DQS which are differential, whereas DDR2 SDRAM data strobes could be programmed by the mode register to be single-ended or differential. DDR3 SDRAM also has a new pin which is the active low asynchronous RESET# pin, which will improve system stability by putting the SDRAM in a known state regardless of the current state. DDR3 SDRAM uses the same type of FBGA packages as DDR2 SDRAM.

DDR3 DIMMs have the terminations for the commands, clock and address on the DIMM. Memory systems using DDR2 DIMM terminate the commands, clock and address on the motherboard. The DDR3 DIMM terminations on the DIMM allow a fly-by topology where each command, clock and address pin on the SDRAM is connected to a single trace which is terminated at the trace end on the DIMM. This improves the signal integrity and results in faster operation than the DDR2 DIMM tree structure. The fly-by topology introduces a new write leveling feature of DDR3 SDRAM for the memory controller to account for the timing skew between the clock CK and data strobes DQS during writes. The DDR3 DIMM is keyed differently than the DDR2 DIMM to prevent the wrong DIMM being plugged into the motherboard.

SDRAM design implementations require complete verification and testing, ranging from circuit board construction to software operation to ensure reliable memory operation. Without full verification of a memory system, product reliability can suffer.

A logic analyzer with its simultaneous state and timing acquisition is the best instrument to verify and debug memory commands and protocols. It can also functionally check the electrical characteristics of the memory signals with its 20 ps high-resolution timing acquisition. This application note focuses on using the logic analyzer to verify correct command and protocol operation of the memory system.

DDR3 Logic Analyzer Solution

The Tektronix logic Analyzer is used with a Nexus Technology DDR3 memory support package to capture DDR3 Read/Write data and DDR3 commands. The TLA7BB4 logic analyzer module is the only module fast enough to address all DDR3 speeds. It's high timing resolution at 20ps(50GS/s) on all channels all the time allows memory designers to capture and analyze glitches, timing faults and signal integrity problems.

Most of the examples within this application note are based on 512Mb DDR2 SDRAM that is configured as 64Meg x 8 outputs. The internal configuration is 16 Meg x 8 outputs x 4 banks.

Techniques shown in this application note are using the TLA7AA4 module and will apply to most SDRAMs, DDR SDRAMs, DDR2 SDRAMs and DDR3 SDRAMs used in embedded systems, computers workstations and servers. For DDR3 SDRAM and many of DDR2 SDRAM, Tektronix recommends the TLA7BB4 module. Please contact your local Tektronix account manager for a detailed recommendation. And as always, check the datasheet of the memory that you are using for its specification, function and operation.

Logic Analyzer Probing

Logic analyzer probing has evolved over the last ten years. Initially, logic analyzer probes were either attached to the IC legs using grabber clips or to square pins that were mounted on the circuit board. As digital designs became more complex and signal speeds increased, logic analyzer probing used quick connect, controlled impedance Mictor connectors on the circuit board.

Leading edge digital designs pushed digital clock rates beyond 1 GHz. As a result, the signal integrity of the probing became extremely critical. At these high signal frequencies the impedance mismatch of connectors and their typically large circuit board foot print size resulted in the evolution of probes to high density connectorless probes. These high-density connectorless probes attach directly to circuit board pads and traces: no connector is used or mounted on the circuit board.

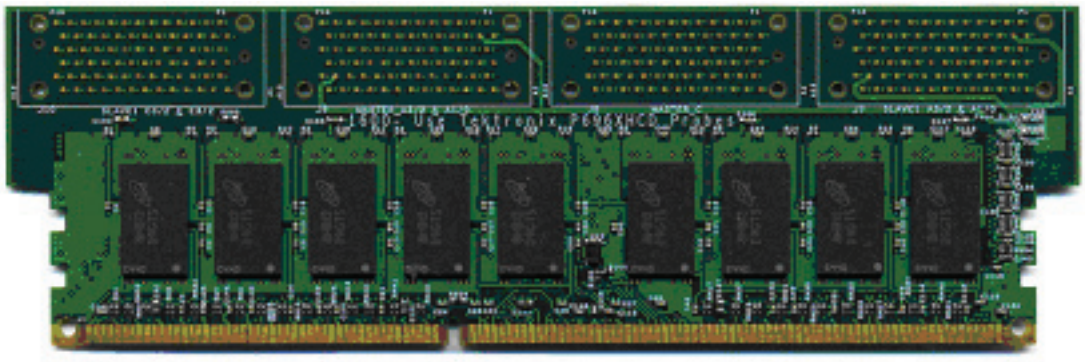


Figure 1. Instrumented DIMM

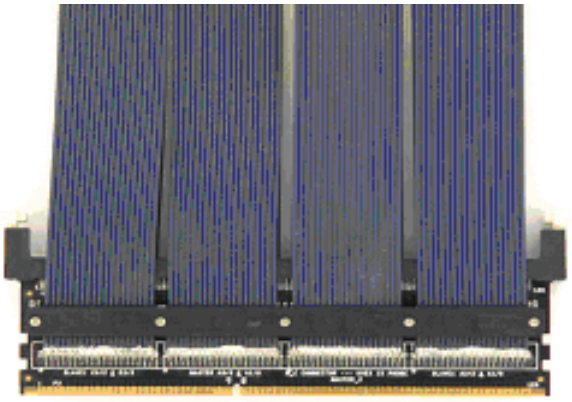


Figure 2. DIMM Interposers

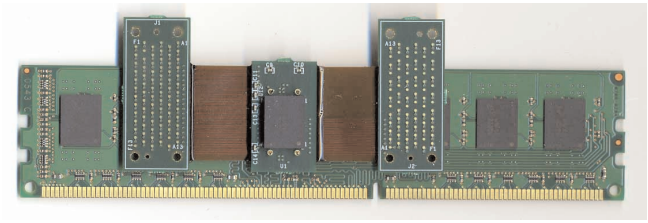


Figure 3. Chip interposer installed between the memory IC and DIMM

Direct Probing

In this type of probing Mictor Connectors or Connectorless footprints are laid out on the circuit board. The DDR signals are routed to these connectors for probing with the Logic Analyzer. This requires careful considerations on how the logic analyzer is going to be used in verifying and debugging the memory system. Analysis of simulations of a logic analyzer probe attached to the memory system will show the best probing location for measuring the signals and the effects of the probe loading on the memory system operation. This simulation is best done before the memory system circuit board is designed.

In some cases, the final product design does not allow for leaving the probing test points on the circuit board. There are several ways to verify and debug these designs. First, a larger validation circuit board is designed with fully instrumented logic analyzer and oscilloscope test points. After validation and debugging is complete, a smaller circuit board with reduced or no test points is designed.

Probing Accessories

An alternate approach is to use probing accessories to probe the DDR SDRAM signals on the circuit board. There are several different probing accessories available from companies such as Nexus Technology and FuturePlus Systems. These include:

Instrumented DIMMs

These extended DIMMs have logic analyzer connectors on them and are designed according to the JEDEC specification. Instrumented SO-DIMMs and FB-DIMMs are available as well. They provide complete access to all the DDR SDRAM signals.

DIMM Interposers

These products install between the Memory DIMM and the circuit board. DIMM interposers use an extender design that does not require an extra connector. Interposers are available for different configurations and different speeds.

Command	S0#	RAS#	CAS#	WE#
Mode Register	0	0	0	0
Refresh	0	0	0	1
Precharge	0	0	1	0
Activate Row	0	0	1	1
Write Column	0	1	0	0
Read Column	0	1	0	1
No Operation	0	1	1	1
Deselect	1	X	X	X

Table 2. SDRAM Commands

.

Chip Interposers

These products install between the memory IC and the circuit board. The design may use a solder down approach where the memory IC is soldered to the interposer or a socketed design where the memory IC is inserted into a socket on the interposer.

Logic Analyzer Channels and Group Setup

SDRAM memory commands are synchronized to the rising edge of the memory clock (CK). The four-command signals are chip select (S0# or CS#), row address select (RAS#), column address select (CAS#) and write enable (WE#). The # symbol indicates these are active low signals (See Table 2). Some memories may have additional signals like the clock enable (CKE), which is an active high signal.

The verification of memory commands requires the logic analyzer probe five signals: CK, S0#, RAS#, CAS# and WE#. The setup of the TLA takes only a minute or two. Setup starts from the default system setup. Next, in the logic analyzer setup window the five command signals – CK, S0#, RAS#, CAS# and WE# – are assigned probe channels and a Command group name is created with the S0#, RAS#, CAS# and WE# channels. By creating a Command group you will be able to use the symbolic command names in the logic analyzer waveform busform, listing window, triggering, filtering and searching.

Not all logic analyzer channels are equal. The clock channels are for external clocking, the qualifiers are used for qualifying the clock edges and the rest of the logic analyzer channels are normal data acquisition channels. For example, it is recommended when using the 136 channel logic analyzer module that the SDRAM memory CK signal is connected to the logic analyzer CK1 or CK3 input channel and the SDRAM memory command signals (S0#, RAS#, CAS# and WE#) are connected to the logic analyzer A1, A3, C3, or E3 channels.

The reason for using these specific logic analyzer channels has to do with the module's ability to be reconfigured to acquire faster and longer recorded lengths using half-channel or quarter-channel acquisition modes. For example, using the TLA7BB4 internal timing at the full 136 channels, the logic analyzer maximum timing resolution is 635 ps (1.6 GS/s) with a maximum record length of 64 Mb per channel. In quarter channel mode, useable channels are reduced to one fourth the original channels at 34 channels and the timing resolution is increased four times to 156 ps (6.4 GS/s) with a record length four times longer at 128 Mb per channel.

For the most flexibility it is best to start with the CK1, CK3, A1, A3, C3, and E3 channels that are used in full, half and quarter channel internal clocking modes. Also, in state acquisition mode the TLA7BB4 logic analyzer module can acquire faster data rates and faster clock rates using half-channel acquisition modes. Check your logic analyzer manual to see if it can be reconfigured to trade off channels or memory to increase its timing resolution. Those channels can then be used first when not using all of the logic analyzer channels in your measurement setup.

Figure 4. DDR2 SDRAM clock transitions indicated by the activity arrow.

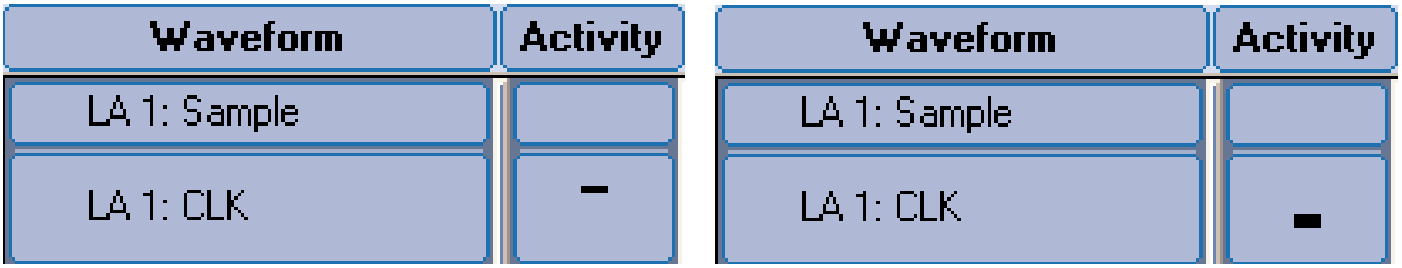


Figure 5 and 6. DDR2 SDRAM clock with no edge transitions at logic high or low, which is indicated by the horizontal activity line.

Logic Analyzer Threshold Setup and Signal Activity

Logic analyzer thresholds depend on the type of signals and the type of memory being measured. The threshold determines the logic high or logic low of the signal depending upon the value of the threshold voltage level. For example, DDR2 SDRAM uses 0.9 V thresholds for S0#, RAS#, CAS# and WE# command signals. The 0.9 V threshold is the same as the DDR2 SDRAM $V_{REF(DC)}$. The DDR2 SDRAM CK is a differential signal and the logic analyzer threshold is set to 0 V, though some designers use a small 50 mV offset from 0 V. It is becoming more common for designs to have differential clock and single-ended data signals. As a result, the Tektronix logic analyzer probes are designed to probe differential clock, differential qualifier signals and single-ended data signals.

DDR3 SDRAM $V_{REF(DC)}$ is 0.750 V and is lower than DDR2 SDRAM.

In review, we started with factory default setup. Next, in the setup window the clock and command signals were added to the probe channels, then a command group was created for the command signals and threshold values were set appropriately. The trigger configuration was left in the trigger immediately. This means that it is set to trigger on the first sample it sees. This is the same as trigger on anything. Trigger immediately acquires data as soon as the Run button is pushed. In the waveform window the Sample marks and Clock signal are added to the waveform display. Click on the Activity button to see signal activity next to each waveform label.

The Activity indicator is an excellent check for live signals before running the logic analyzer. For the clock, there should be a vertical two headed arrow indicating clock transitions (See Figure 4). If there is no waveform activity arrow (See Figures 5 and 6) check for the following: the memory system is powered and running, the correct probe channel is attached to the correct test point in the memory system, and the logic analyzer threshold is the correct level for the differential clock signal.

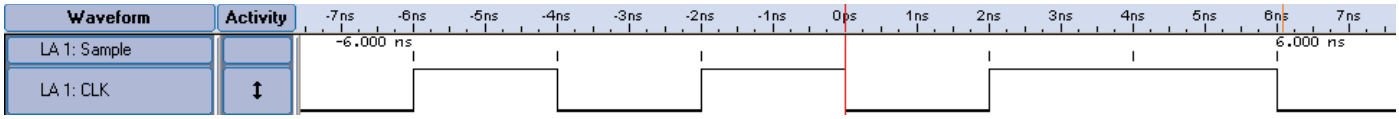


Figure 7. DDR2 SDRAM clock measured at 2 ns timing resolution.

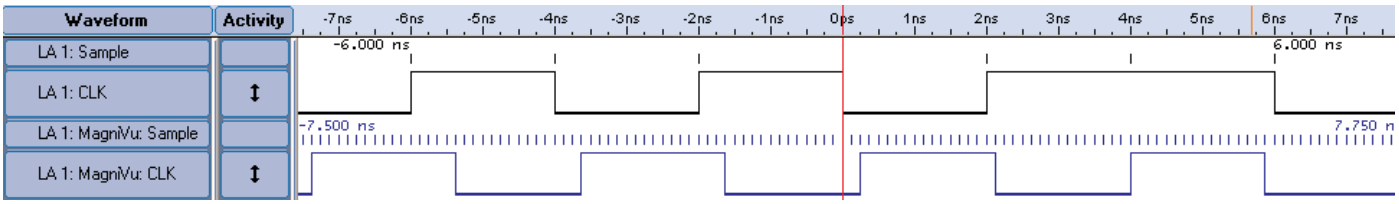


Figure 8. DDR2 SDRAM clock measured with 2 ns timing resolution in the top deep timing waveform and simultaneously measured with MagniVu 125 ps high-resolution timing in the bottom waveform. Notice the better representation of the clock signal with 125 ps timing resolution.

DDR2 SDRAM Clock Waveform

Capture the DDR2 SDRAM clock with the logic analyzer by pushing the run button. The timing waveform is shown in Figure 7. Notice the vertical Sample marks indicating the clock was measured at the default 2 ns timing resolution of the logic analyzer. Greater timing resolution measurements provide more accurate clock measurements. You have two techniques to measure the clock at higher timing resolution. The first technique is to use the half channel or quarter channel timing modes. The second technique is to use Tektronix MagniVu™ acquisition 125 ps high-resolution timing on TLA7AA4 and 20 ps high-resolution timing on TLA7BB4 modules.

MagniVu is a Tektronix patented digital oversampling logic analyzer architecture that uses one high-resolution sampler on every logic analyzer channel to acquire data for all logic analyzer acquisition modes while saving its own high-resolution timing data. It is like have two logic analyzers in one: MagniVu high-resolution timing and normal deep timing or state acquisition. Because there is only one high-resolution sampler for all data, MagniVu, deep timing and state acquisition is perfectly time correlated.

MagniVu sampling resolution on TLA7AA4 modules is 125 ps and its record length is 16K whereas on TLA7BB4 modules the MagniVu sampling resolution is 20 ps and its record length is 128K. MagniVu high-resolution timing is always acquired; it cannot be turned off. To see the MagniVuhigh-resolution timing waveform for DDR2 SDRAM clock waveform, just push the MagniVu button (See Figure 8).

In Figure 8, notice the benefit of the higher timing resolution. The DDR2 SDRAM clock signal is measured with two different timing resolutions with the same logic analyzer probe. With MagniVu waveform the signal edge uncertainty using a TLA7AA4 module is 125 ps, whereas with the deep timing waveform the edge uncertainty is 2 ns. MagniVu high-resolution timing captures the signal more accurately.

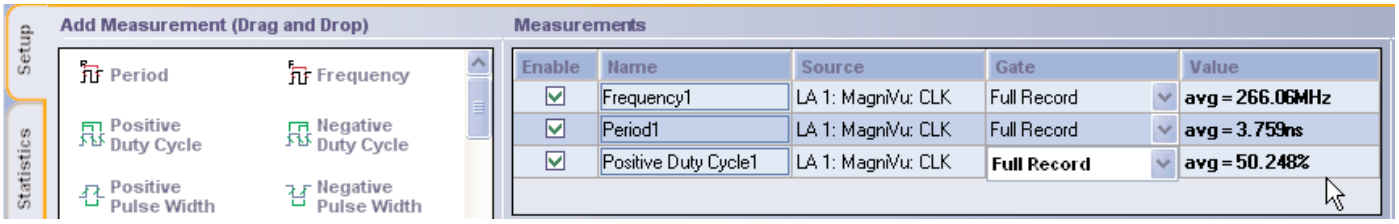


Figure 9. DDR2 SDRAM clock measurements: 266.06 MHz frequency, 3.759 ns period and 50.248% positive duty cycle.

DDR2 SDRAM Clock Measurements

The DDR2 SDRAM clock frequency, period and duty cycle is checked very quickly using the logic analyzer drag and drop measurements. In the Measurement tab at the bottom of the Waveform window, select the Frequency measurement and drag and drop it on the MagniVu DDR2 SDRAM clock waveform. Do the same for Period and Positive Duty Cycle measurements. The MagniVu waveform is used because it has the highest edge resolution. To increase the accuracy of the drag and drop measurements, change the measurement Gate from Display to Full Record (See Figure 9). The Statistics tab for the measurements show the population changes from 4 with a Display Gate to 542 with a Full Record Gate.

The 50.248% positive duty cycle is well in the 45% to 55% specifications of this memory. For example, for a DDR2 533 SDRAM, the clock frequency is 266.06 MHz. The DDR2 SDRAM is commonly referred to by its data rate, which is twice the clock frequency.

For the highest accuracy measurement it is best to measure the clock signal with a Tektronix high performance oscilloscope. Other clock specifications, such as clock jitter, rise time, fall time, and spread spectrum clocking (SSC), are better measured with an oscilloscope with the DPOJET Jitter Analysis and Timing software. Typically, an oscilloscope has higher analog bandwidth, and provides an analog signal measurement. These measurement capabilities make the oscilloscope a better instrument for high precision analog measurements and parametric measurements. However, it is very easy and quick to functionally check the DDR2 SDRAM clock with the logic analyzer drag-and-drop measurements.

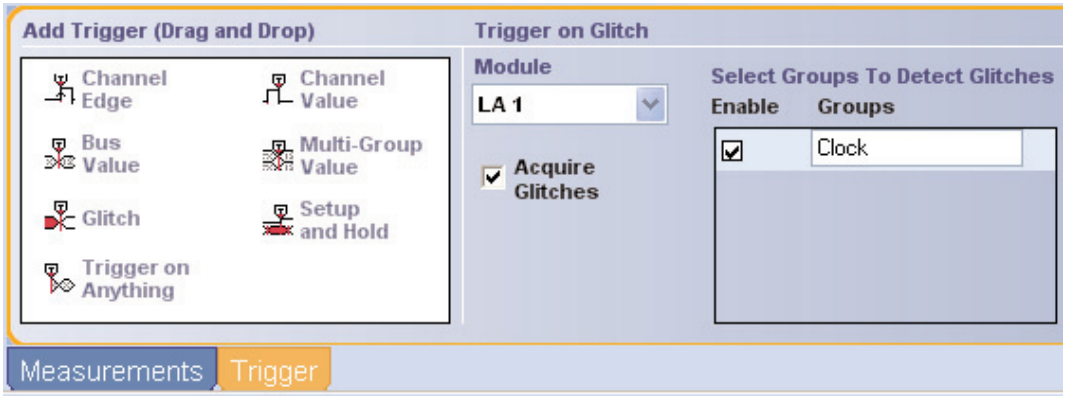


Figure 10. Quick drag and drop trigger setup on the DDR2 SDRAM clock to check it for glitches.

Checking for Clock Glitches

A clean DDR2 SDRAM clock sourced by the memory controller hub is essential for reliable memory operation. Clock glitches can be caused by crosstalk, trace impedance errors, signal termination errors, power supply faults, etc.

For SDRAM, DDR SDRAM, DDR2 SDRAM and DDR3 SDRAM, the logic analyzer can be used to quickly check for clock glitches by using its drag and drop Glitch triggering. In the Waveform window, select the Trigger tab that is next to the Measurement tab and drag and drop the added Glitch trigger on to the Clock group waveform. This drag and drop Glitch trigger will setup the logic analyzer to trigger on Clock glitches only (See Figure 10).

The Tektronix TLA Series logic analyzer defines a glitch to be two or more edges in a deep timing sample period. The Acquire Glitches check box in the drag and drop Trigger window turns the red glitch bar flags on and off. In glitch triggering, the logic analyzer is running and will only stop if it sees a glitch on the SDRAM clock. The key to using logic analyzer glitch triggering for memory clock testing is that the clock positive pulse width and negative pulse width must be greater than the deep timing sample period.

If you start the logic analyzer before lunch or before leaving for the weekend and it is running when you return, you can have confidence that the SDRAM clock signal is glitch free. If there is a glitch, the iView™ display with integrated logic analyzer and oscilloscope operation provides powerful full memory signal visibility, from memory protocols to digital and analog waveforms, for debugging. If the logic analyzer triggers on glitch, you will see two or more edges on the MagniVu waveform in the same time as the deep timing sample period at the trigger point in the Waveform window. To further debug the clock glitch, use the integrated operation mentioned above. In this operation, the logic analyzer will trigger on the clock glitch. The logic analyzer then triggers the oscilloscope to capture the analog characteristics of the glitch. The logic analyzer then copies that analog clock waveform and displays it, time aligned, with the logic analyzer digital waveforms.

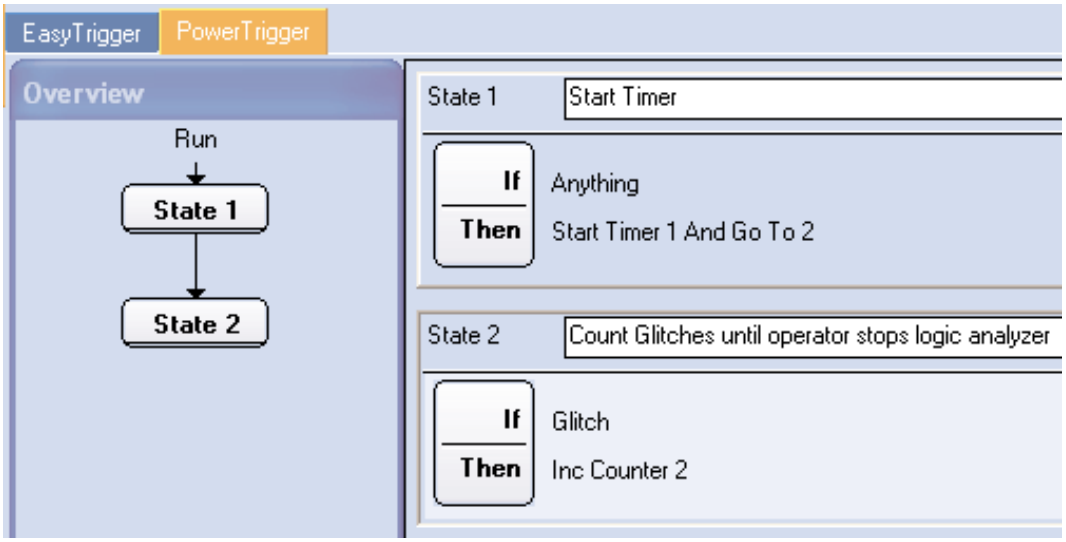


Figure 11. Use the Logic analyzer PowerTrigger program to time the length of time the logic analyzer is running and to count the number of SDRAM clock glitches.

Note that drag and drop triggering works only on deep timing groups and individual waveform channels. Therefore, when adding the clock waveform to the Waveform window, it is best to use the clock group and not add the clock waveform by probe or by name. In the Waveform window, an expansion box to the left of the waveform label indicates a waveform group. A waveform group is shown as a busform waveform when the group has more than one channel in it. To expand the group, click on the expansion box plus sign. To collapse the group, click on the expansion box minus sign.

To determine how numerous your glitches are, create a logic analyzer PowerTrigger program of two states to start a timer and to count the number of glitches until you stop the logic analyzer (See Figure 11). The values of the timer and counter are constantly being updated and are displayed in the logic analyzer Status Monitor. The first state starts a timer as soon as the logic analyzer starts running. The second state counts the glitches until you stop the logic analyzer.

The Tektronix oscilloscopes also have the capability to trigger on a glitch. Therefore, if you are measuring the clock with the oscilloscope, take a minute or two to check out the clock signal by setting its triggering to trigger on a glitch.

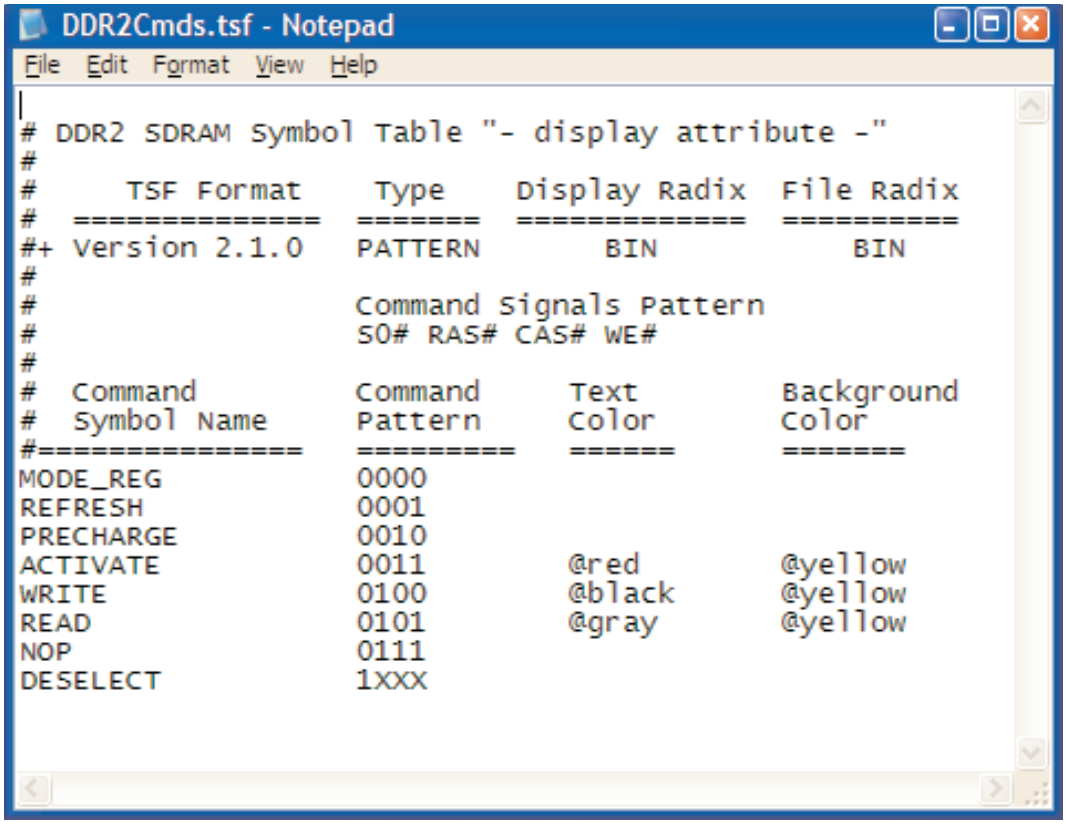


Figure 12. Pattern symbol file example for DDR2 SDRAM.

Capturing & Displaying the SDRAM Command Signals

The memory controller hub sends the clock and memory commands to the SDRAM. Before the introduction of SDRAM, the timing and sequence of the RAS#, CAS# and WE# edges determined the dynamic RAM (DRAM) operating mode. Synchronous DRAM (SDRAM) samples the S0#, RAS#, CAS# and WE# command signals at the rising edge of the SDRAM clock. The orderly SDRAM synchronous command operation replaces the DRAM asynchronous operation.

In order to add the SDRAM command group, do so in the logic analyzer Waveform window. If you set up to trigger immediately, you may not see any changes to the command group. This occurs because the memory controller hub may have deselected the memory and there were no memory commands being sent to the SDRAM at the time you pushed the logic analyzer run button.

The Activate Row command is the first command of a write or read command sequence. To trigger the logic analyzer on the Activate Row command, configure the logic analyzer to trigger on a Command group equal to 0011. This is S0#=0, RAS#=0, CAS#=1 and WE#=1,
as shown in Table 2.

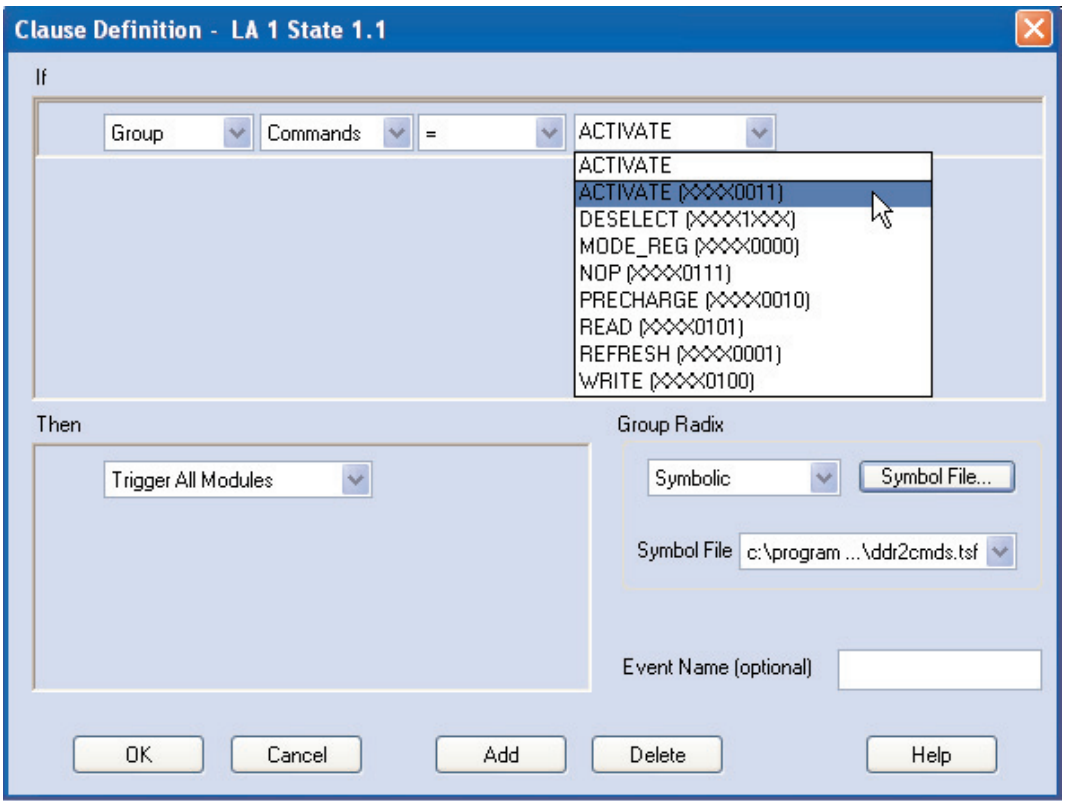


Figure 13. The logic analyzer PowerTrigger program uses pattern symbols to trigger on the SDRAM Activate command.

Dealing with magic numbers like 0011 is error prone. The logic analyzer works with data in several formats: binary, octal, hex, decimal, signed decimal and symbolic. Pattern symbol files are used when a group of signals define a logical state such as the SDRAM command group. Each pattern symbol may also have a color associated with it. Based on the SDRAM command table, as shown in Table 2, the following Tektronix Symbol File (TFS) was created with Microsoft Notepad and saved as DDR2Cmds.tsf (See Figure 12).

The logic analyzer uses these pattern symbols when setting up the logic analyzer to trigger on the SDRAM Activate command (See Figure 13). In the logic analyzer PowerTrigger Clause Definition, the logic analyzer is configured to trigger when the Commands group equals the Activate command. To use pattern symbols in the PowerTrigger Clause Definition, the Group Radix is changed to Symbolic and the DDR2Cmds.tsf file is selected.

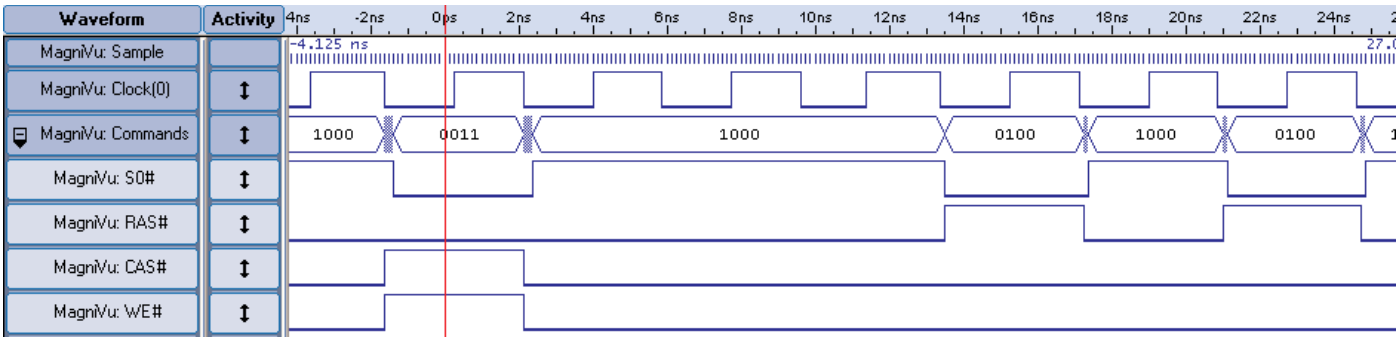


Figure 14. Triggering on the SDRAM Activate command in the Commands group. The Commands group is expanded to show SDRAM command S0#, RAS#, CAS# and WE# signals.

The logic analyzer triggers on the SDRAM Activate command (binary 0011) (See Figure 14). The Commands group waveforms are shown in busform with the values of the Commands bus in binary and the signal edge transitions indicated by crossing lines or vertical lines from top to bottom.

In Figure 14, the Commands group busform is expanded to see the individual signals by clicking on the plus sign to the left of the Commands group label. The Commands group radix is binary. The vertical line at 0ps is the trigger point. At the trigger point, the Commands group is binary 0011, which is the Activate command, as shown in Table 1. The expanded command waveforms are S0#=0, RAS#=0, CAS#=1 and WE#=1 at the trigger point.

As with logic analyzer triggering, change the Radix of the number shown in the Commands group busform. In the Waveform properties, change the radix to symbolic and select the same pattern symbols DDR2Cmds.tsf file that was used in PowerTrigger. A much more readable busform can be viewed with color symbols (See Figure 125. Using the busform with symbols is a powerful tool to verify correct SDRAM command protocol sequence and timing relationships.

Verifying the SDRAM Write Operation

The protocol sequence for a SDRAM writes operation starts with the Activate command followed by one or more Write commands. The Activate command with its row and bank addresses opens a specific row in a specific bank for writes and reads. The Write command with its column and bank addresses opens a specific column in the opened row in a specific bank for writes. It would be a protocol error for the Write command to access a bank that has no open rows. After the Write command, the memory expects at a defined memory cycle that the memory controller hub will write data to it. Typically DDR2 SDRAMs work with data in groups of multiples of four. The row needs to be closed or deactivated with a Precharge command when the writing is completed for the open row and another row is to be accessed. The simplest DDR2 SDRAM command protocol sequence is Activate, Write and Precharge. A consecutive write-to-write sequence is Activate, multiple Writes and Precharge. A write-to-read sequence is Activate, Write, Read, and Precharge. You can have any order of Writes and Reads on an open row.

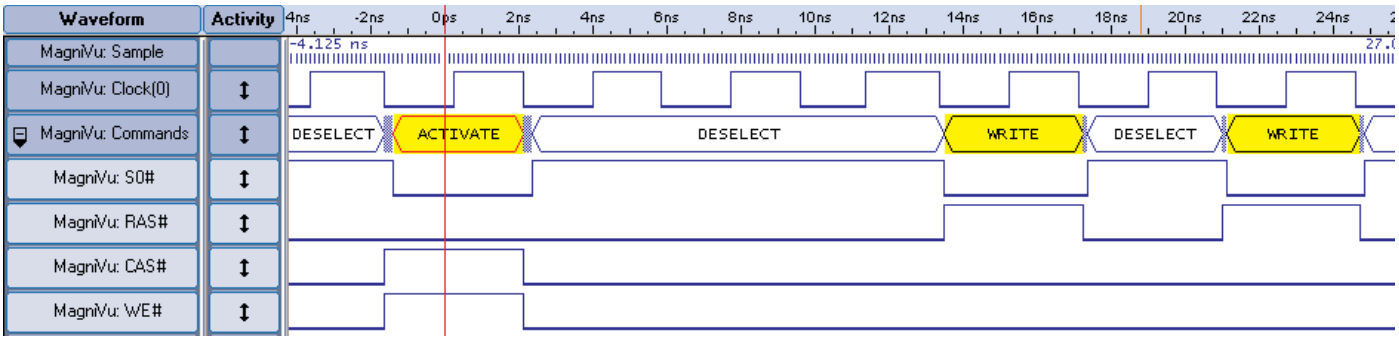


Figure 15. Triggering on the SDRAM Activate command and the Commands group is using command symbols with colors. Also, the Commands group is expanded to show S0#, RAS#, CAS# and WE# signals.

Figure 15 shows the beginning of a write sequence with the Activate command followed by the Write command. DDR2 SDRAMs specify the minimum time ($t_{RCD(MIN)}$) between the Activate and Write command. Also, this minimum time is dependent upon the Additive latency of the memory. Posted CAS Additive latency (AL) makes the command and data bus more efficient for sustainable bandwidths. Additive latency allows the Write command to be sent to the memory before the minimum ($t_{RCD(MIN)}$). Additive latency values are programmed into the memory Extended Mode Register by the memory controller hub. No Additive latency was used in the DDR2 SDRAM examples.

For example, for DDR2 SDRAM the Write command should not be sent within $t_{RCD(MIN)}$ of the Activate command when the Additive latency is zero. In this example, the $t_{RCD(MIN)}$ is specified at 15 ns. A quick way to verify the t_{RCD} value is to right click on therising clock edge of the Activate command and add a marker there, doing the same for the rising clock edge of the Write command. To help readability, you can double click on the marker and change their default names to Activate and Write. Also, if you need to adjust the Activate and Write markers to the clock edges, the markers will snap to the clock waveform edge when you grab the marker with the mouse and move it horizontally on top of the clock waveform. When you do not want to snap the marker to a waveform edge, move it horizontally on an empty waveform space. Next, change the delta time measurement box to display the time difference between the Activate marker and the Write marker. The delta time measurement box can be undocked from the menu bar and moved in a more convenient place.

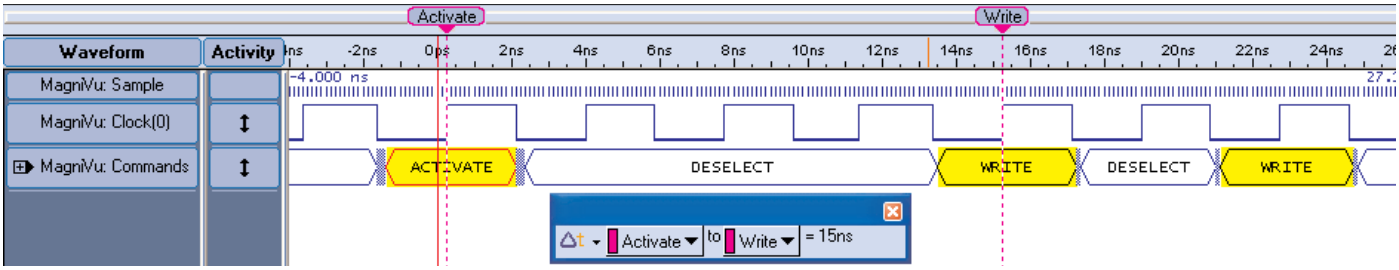


Figure 16. Measuring the t_{RCD} value of 15 ns between the Activate command and the Write command.

The Write command was sent 15 ns after the Activate command (See Figure 16). DDR2 SDRAM specification is violated by the memory controller hub if the measured t_{RCD} is less than 15 ns and the Additive latency is zero.

Figure 16 also shows the beginning of a consecutive Write to Write operation. A second Write command is sent by the memory controller hub two clock cycles after the first Write. There is a Deselect command between the consecutive Write commands. It would be a DDR2 DRAM protocol error if the memory controller hub sent two Write commands in a row without the Deselect command between them. The DDR2 DRAM will respond to the Write command by reading in data that is strobed by the memory controller hub.

DDR2 DRAM has a minimum burst length of four data bits. DDR stands for Double Data Rate, which means two data bits are written per clock cycle. Two clock cycles are needed to complete the write operation with a burst length of four data bits. This is the reason for the Deselect command following the Write command. The Deselect command allows time for the second cycle of writing data that is needed after the Write command.

Another key DDR2 DRAM specification is the minimum t_{RP} time after the Precharge command is sent and before the Activate command is sent to open a row. This can be easily verified by changing the logic analyzer to trigger on the Precharge command and measuring the t_{RP} time between the Precharge and Activate commands to the same bank.

Figure 16 shows a DDR2 DRAM write operation. The same protocol and timing verification techniques are applied to DDR2 DRAM read operations.

Protocol Verification with Logic Analyzer State Acquisition

Logic analyzer state acquisition is a more efficient way to acquire and to verify the SDRAM commands protocol sequencing. State acquisition uses the SDRAM clock to store the values of the SDRAM commands in the logic analyzer. One logic analyzer memory location is used for one SDRAM clock cycle whereas multiple logic analyzer memory locations are used in timing acquisitions. State acquisition is also called external clocking or synchronous acquisition.

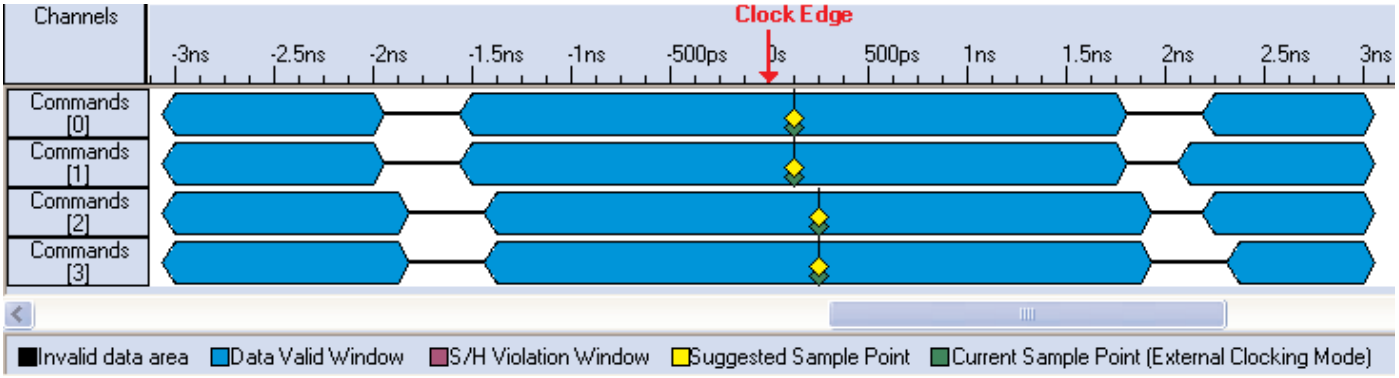


Figure 17. Logic analyzer AutoDeskew analysis shows the suggested sample point position with yellow diamonds. The green diamonds show the current sample point position, which has been set equal to the suggested position.

Clocked signals like the SDRAM commands have setup and hold timing requirements with regard to the SDRAM clock edge. This is also called the data valid window. The memory command signal's data valid window is around the rising SDRAM clock edge. The logic analyzer has a data valid window for the SDRAM command signals in order to acquire SDRAM commands with the SDRAM clock. The logic analyzer is flexible in that the data valid windows for each logic analyzer channel can be adjusted for optimum positioning to ensure reliable signal capture by the logic analyzer.

The logic analyzer AudoDeskew analysis will provide optimum positions of the logic analyzer's data valid windows for each channel.

AutoDeskew analyzes the SDRAM commands signals looking for edge transitions. The edge transitions are in reference to the external SDRAM rising clock edge (See Figure 17). A black horizontal line indicates invalid data because of edge transitions. A blue solid bar indicates a data valid window with no edge transitions. A yellow diamond shows the suggested sample point position for the logic analyzer. A green diamond shows the current sample point position for the logic analyzer, which has been adjusted to match the suggested sample point position. Commands [3, 2, 1, & 0] channels represent S0#, RAS#, CAS# and WE# command signals.

In this example, the S0#, RAS#, CAS# and WE# command signals are well-behaved and have their edges at similar times and the invalid data areas are small. These well-behaved signals are not required because when Chip Select S0# is not activated, the edges of RAS#, CAS# and WE# can occur any time in the clock cycle. AutoDeskew will see these edges and show very small or no data valid windows around the clock edge. AutoDeskew will always work for the Chip Select S0# because it is always an active command signal. Start with using the S0# suggested sample point for RAS#, CAS# and WE# if AutoDeskew shows no data valid windows around the SDRAM clock edge.

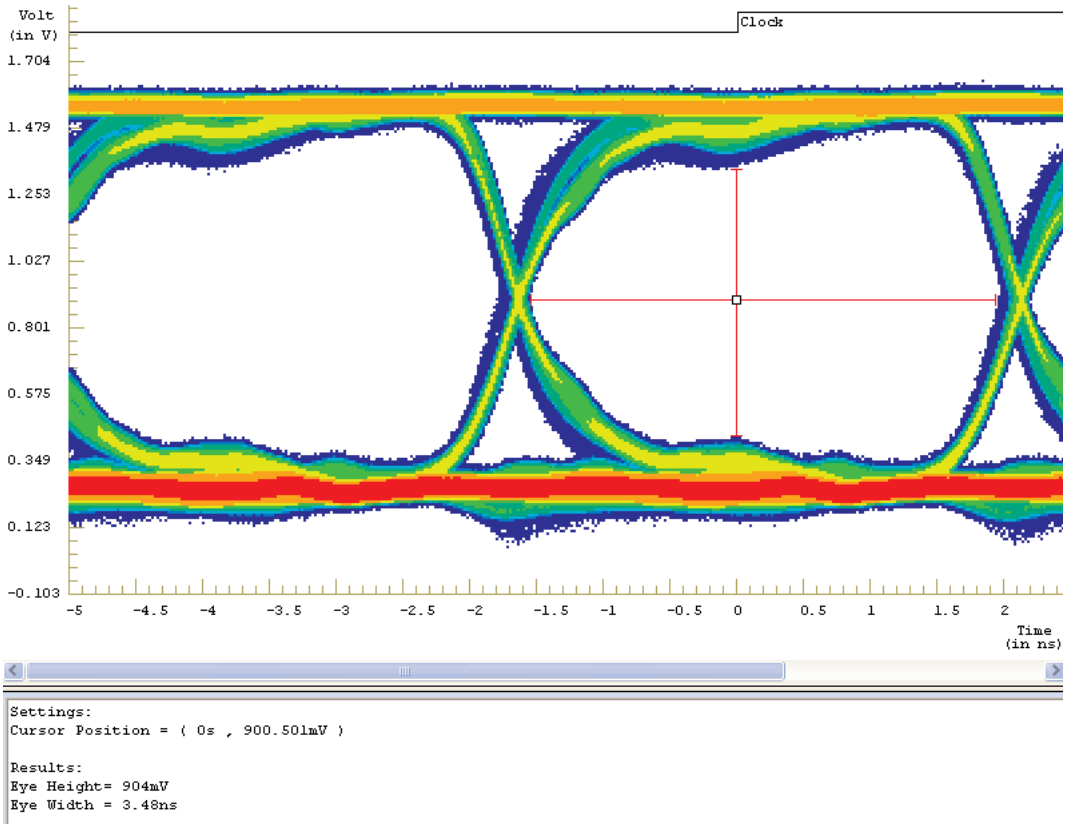


Figure 18. Multi-channel eye diagram of S0#, RAS#, CAS# and WE# related to the SDRAM rising edge clock. The eye opening at the cursor is 904 mV high and 3.48 ns wide.

There are at least four methods to verifying the suggested sample point positions. First, use AutoDeskew Setup/Hold Violation analysis. Second, set up a Setup/Hold Violation PowerTrigger based on the suggested sample point positions. Third, verify these suggested sample point positions by analyzing the MagniVu high-resolution traces of CK, S0#, RAS#, CAS# and WE#. Fourth, analyze the multi-channel eye diagram using the integrated logic analyzer and oscilloscope iVerify™ analysis function (See Figure 18).

A unique Tektronix logic analyzer feature is called iCapture™ multiplexing, where the logic analyzer probes are used simultaneously by the logic analyzer and an external oscilloscope. The logic analyzer controls, triggers and transfers data from the oscilloscope using the iView interface cable. The iVerify analysis function uses iCapture and iView to create multiple channel eye diagrams on the logic analyzer display. All of the eye diagrams can be analyzed at once or one channel can be highlighted. Figure 18 shows the multiple channel eye diagrams for S0#, RAS#, CAS# and WE#. The moveable cursor measures the eye opening at 904 mV high and 3.48 ns wide at the rising clock edge. This type of analysis is quick to set up and use for checking S0#, RAS#, CAS# and WE#.

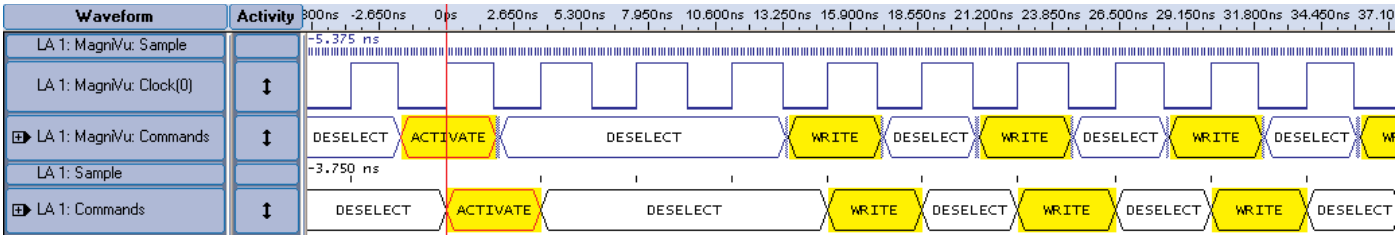


Figure 19. Commands State busform is at the bottom of the Waveform window. At the top waveform and busform are the MagniVu 125ps high-resolution timing of the SDRAM clock and commands.

RAS#, CAS# and WE# can change anywhere in the clock cycle when the chip select S0# is inactive. However, this may cause invalid eye diagrams resulting from edge transitions that occurred when the memory was deselected. The iVerify setup menu provides a way to qualify the rising clock edge with any logic analyzer channel. Therefore, S0# active low is used to qualify the rising clock edges to capture valid eye diagrams.

The MagniVu high-resolution timing of the SDRAM clock and commands are shown at the top of the Waveform window (See Figure 19). At the bottom is the State acquisition Commands busform, which uses the pattern symbol file to decode it. Carefully look at the Sample tick marks above the Commands State busform. Note that they only occur at the rising edge of the SDRAM clock because it is this clock edge that causes the logic analyzer to sample the Commands bus. As a result, the logic analyzer samples the commands only once during the SDRAM clock cycle and the Commands State busform will only change value at the rising edge of the SDRAM clock. The advantage of State acquisition is the logic analyzer is looking at the Commands bus in the same way that the SDRAM is looking at the Commands bus.

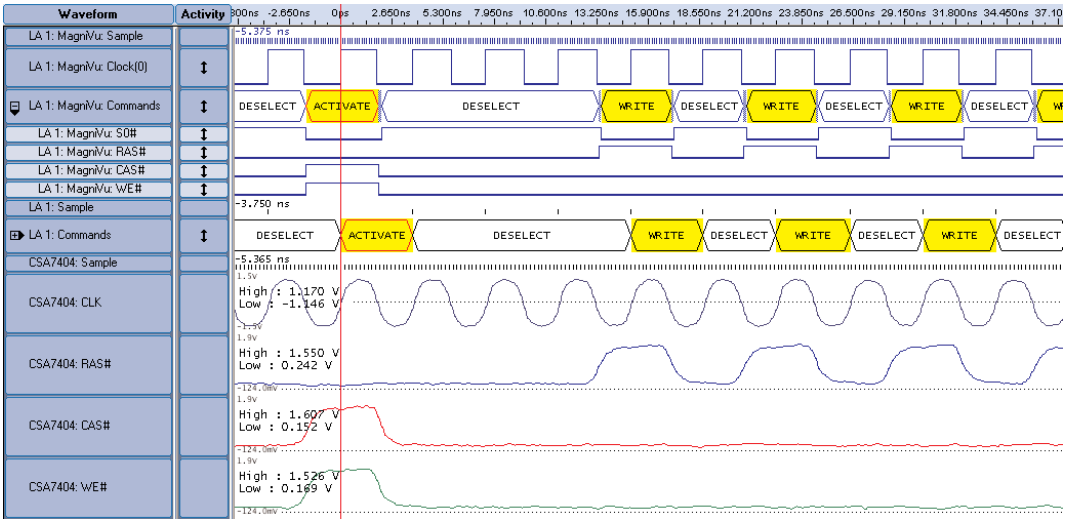


Figure 20. The logic analyzer iCapture probe provides simultaneous probing for three different types of SDRAM clock and commands measurements. The top waveforms are MagniVu 125 ps high-resolution timing. The middle Commands busform is state acquisition. The bottom four waveforms are the CLK, RAS#, CAS# and WE# from the oscilloscope using the iView interface.

iLink™ Toolset: Two Powerful Measurement Tools Team Up

Although Logic Analyzers and Oscilloscopes have long been the tools of choice for digital troubleshooting, not every designer has seen the dramatic benefits that come with integrating these two key instruments.

Logic Analyzers speed up debugging and verification by wading through the digital information stream to trigger on circuit faults and capture related events. Oscilloscopes peer behind digital timing diagrams and show the raw analog waveforms, quickly revealing signal integrity problems.

Several Tektronix Logic Analyzer models offer the iLink™ toolset, a Logic Analyzer/Oscilloscope integration package that is unique in the industry. The iLink™ toolset joins the power of Tektronix TLA Series Logic Analyzers – MagniVu™ acquisition with 20 ps resolution and advanced state machine-based triggering – to selected Tektronix Oscilloscope models.

A powerful set of iLink™ toolset features brings time-correlated digital and analog signals to the logic analyzer display. While the Logic Analyzer acquires and displays a signal in digital form, the attached Tektronix Oscilloscope captures the same signal in its analog form and displays it on the Logic Analyzer screen. Seeing these two views simultaneously makes it easy to see, for example, how a timing problem in the digital domain was a result of a glitch in the analog realm.

The iLink™ Toolset is a comprehensive package designed to speed problem detection and troubleshooting:

- iCapture™ multiplexing provides simultaneous digital and analog acquisition through a single Logic Analyzer probe
- iView™ display delivers time-correlated, integrated Logic Analyzer and Oscilloscope measurements on the Logic Analyzer display
- iVerify™ analysis offers multi-channel bus analysis and validation testing using Oscilloscope-generated eye diagrams

Tektronix iCapture and iView provide an integrated operation between the logic analyzer and oscilloscope. The oscilloscope waveforms for CLK, RAS#, CAS# and WE# time correlated with the logic analyzer Commands State busform and MagniVu high-resolution timing of the SDRAM clock and commands (See Figure 20). This complete waveform visibility of analog, digital timing and digital state is measured through the iCapture logic probe. The logic analyzer can trigger the oscilloscope or the oscilloscope can trigger the logic analyzer. Tektronix iCapture, iView and iVerify are part of the iLink™ ToolSet and they provide powerful features for memory verification and debugging.

The Listing window is a more efficient way to analyze Commands State data (See Figure 21). The first column lists the sample number. The second column lists Commands State data that has been decoded with a previous pattern symbol file. The last column is the Timestamp with, in this case, 125 ps resolution. This can be configured several different ways: time from beginning, time from trigger, time from previous sample (shown in Figure 21), and other variations. Note that the horizontal line in the middle is the trigger point. The trigger was on the Activate command. There is a long sequence of Deselects before the Activate command. The Activate command is the start of a long sequence of Write to Write operations. The three Deselect commands after the Activate command are to satisfy the minimum DDR2 SDRAM $t_{RCD(MIN)}$ before the Write command can be sent. The Deselect command between the Write commands are allows time for the burst of four data bits to be written to memory from the leading Write command.

Sample	Commands	Timestamp
65506	DESELECT	3.750 ns
65507	DESELECT	3.750 ns
65508	DESELECT	3.750 ns
65509	DESELECT	3.750 ns
65510	DESELECT	3.750 ns
65511	DESELECT	3.750 ns
65512	DESELECT	3.875 ns
65513	DESELECT	3.750 ns
65514	DESELECT	3.750 ns
65515	DESELECT	3.750 ns
65516	DESELECT	3.750 ns
65517	DESELECT	3.750 ns
65518	DESELECT	3.750 ns
65519	DESELECT	3.750 ns
65520	DESELECT	3.750 ns
65521	DESELECT	3.750 ns
65522	DESELECT	3.750 ns
65523	DESELECT	3.750 ns
65524	DESELECT	3.750 ns
65525	DESELECT	3.750 ns
65526	DESELECT	3.750 ns
65527	DESELECT	3.875 ns
65528	DESELECT	3.750 ns
65529	DESELECT	3.750 ns
65530	DESELECT	3.750 ns
65531	DESELECT	3.750 ns
65532	DESELECT	3.750 ns
65533	DESELECT	3.750 ns
65534	DESELECT	3.750 ns
65535	ACTIVATE	3.750 ns
65536	DESELECT	3.750 ns
65537	DESELECT	3.750 ns
65538	DESELECT	3.750 ns
65539	WRITE	3.750 ns
65540	DESELECT	3.750 ns
65541	WRITE	3.750 ns
65542	DESELECT	3.875 ns
65543	WRITE	3.750 ns
65544	DESELECT	3.750 ns
65545	WRITE	3.750 ns
65546	DESELECT	3.750 ns
65547	WRITE	3.750 ns
65548	DESELECT	3.750 ns
65549	WRITE	3.750 ns
65550	DESELECT	3.750 ns
65551	WRITE	3.750 ns
65552	DESELECT	3.750 ns
65553	WRITE	3.750 ns
65554	DESELECT	3.750 ns
65555	WRITE	3.750 ns
65556	DESELECT	3.750 ns
65557	WRITE	3.875 ns
65558	DESELECT	3.750 ns
65559	WRITE	3.750 ns
65560	DESELECT	3.750 ns
65561	WRITE	3.750 ns
65562	DESELECT	3.875 ns
65563	WRITE	3.750 ns
65564	DESELECT	3.750 ns
65565	WRITE	3.750 ns
65566	DESELECT	3.750 ns
65567	WRITE	3.750 ns
65568	DESELECT	3.750 ns
65569	WRITE	3.750 ns
65570	DESELECT	3.750 ns
65571	WRITE	3.625 ns

Figure 21. Listing window showing the state acquisition of the SDRAM Commands with commands symbol names.

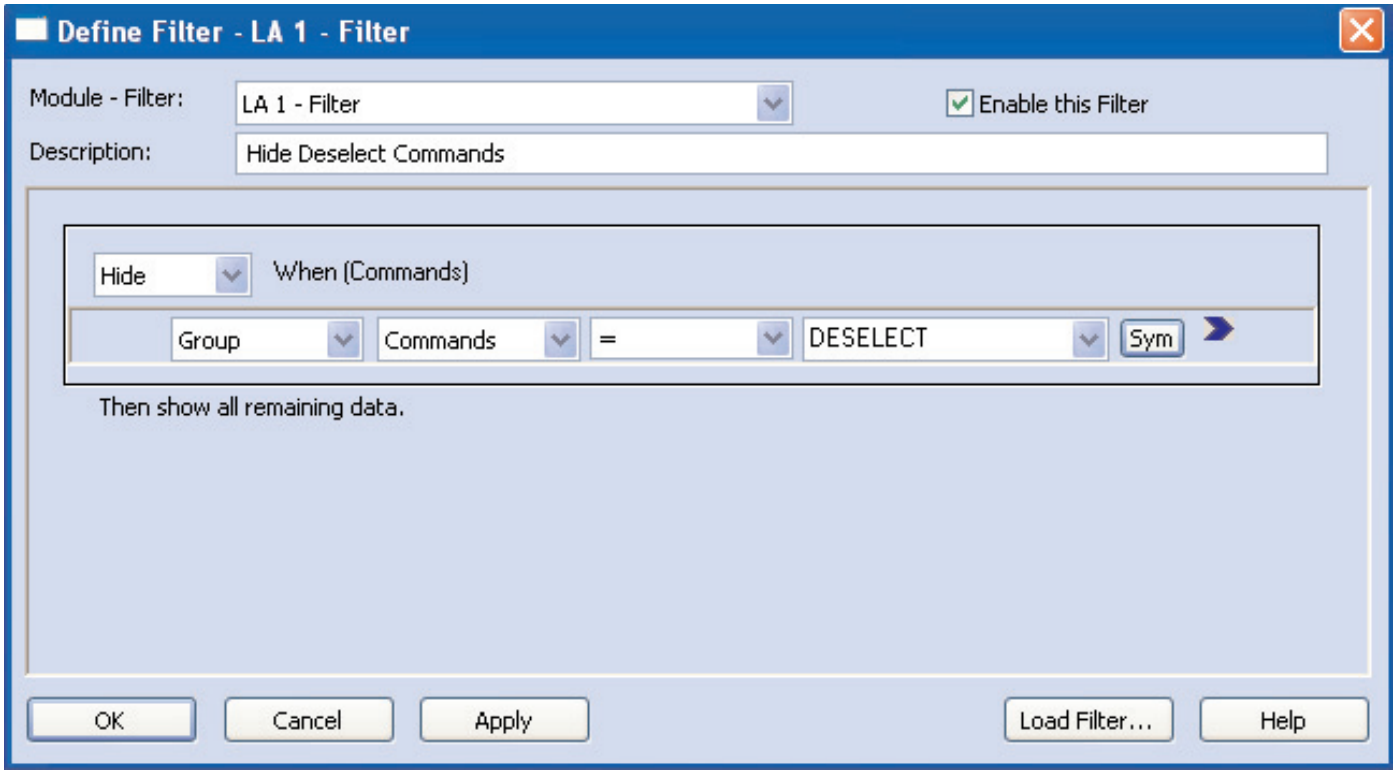


Figure 22. A display filter to hide all SDRAM Deselect commands.

You can define display filters to make your analysis more efficient. For example, Figure 21 shows a large number of Deselect commands. It would be more useful if the Deselect commands were hidden on the Listing window. The Define Filter window is used to hide the Deselect commands (See Figure 22). Notice that the commands symbol names are used in hide when Commands group equals Deselect. The display filter does not change the acquired data; it only affects the way the data is displayed.

The Listing window can now be viewed with the hide Deselect commands display filter applied (See Figure 23). A long list of Refresh commands about 7.8 µs apart is visible before the Activate command and a long list of Write to Write commands after the Activate command. Notice the time-stamp value is to the previous displayed command. The timestamp for the first Write command is 15 ns which is the value that was measured in the Waveform window. For this memory, the minimum time $t_{RCD(MIN)}$ for the Activate to Write command is 15 ns. The resolution of the timestamp for this TLA7AA4 module is 125 ps. In the case of a TLA7BB4 module, the resolution would be 20 ps.

Verifying Refresh Operation

The underlining storage element of DRAM is a capacitor for each memory cell. The charge on the memory cell capacitor will eventually decay and the memory cell will lose its value. Before the memory cell capacitor loses its value the memory cell needs to be read and then have that same value put back in the memory cell. This operation is called refresh.

Sample	Filter Commands	Timestamp
580	REFRESH	0 ps
2656	REFRESH	7.802,125 us
4732	REFRESH	7.802,500 us
6808	REFRESH	7.802,500 us
8884	REFRESH	7.802,625 us
10960	REFRESH	7.802,375 us
13036	REFRESH	7.802,000 us
15112	REFRESH	7.802,250 us
17188	REFRESH	7.802,125 us
19264	REFRESH	7.802,125 us
21340	REFRESH	7.801,750 us
23416	REFRESH	7.801,750 us
25492	REFRESH	7.802,250 us
27568	REFRESH	7.802,000 us
29644	REFRESH	7.802,125 us
31720	REFRESH	7.802,250 us
33796	REFRESH	7.802,125 us
35872	REFRESH	7.802,375 us
37948	REFRESH	7.802,250 us
40024	REFRESH	7.802,750 us
42100	REFRESH	7.802,750 us
44176	REFRESH	7.802,000 us
46252	REFRESH	7.802,125 us
48328	REFRESH	7.802,250 us
50404	REFRESH	7.802,000 us
52480	REFRESH	7.801,625 us
54556	REFRESH	7.802,375 us
56632	REFRESH	7.801,875 us
58708	REFRESH	7.802,375 us
60784	REFRESH	7.802,625 us
62860	REFRESH	7.801,875 us
64936	REFRESH	7.802,500 us
65535	ACTIVATE	2.251,375 us
65539	WRITE	15.000 ns
65541	WRITE	7.500 ns
65543	WRITE	7.625 ns
65545	WRITE	7.500 ns
65547	WRITE	7.500 ns
65549	WRITE	7.500 ns
65551	WRITE	7.500 ns
65553	WRITE	7.500 ns
65555	WRITE	7.500 ns
65557	WRITE	7.625 ns
65559	WRITE	7.500 ns
65561	WRITE	7.500 ns
65563	WRITE	7.625 ns
65565	WRITE	7.500 ns
65567	WRITE	7.500 ns
65569	WRITE	7.500 ns
65571	WRITE	7.375 ns
65573	WRITE	7.625 ns
65575	WRITE	7.500 ns
65577	WRITE	7.500 ns
65579	WRITE	7.625 ns
65581	WRITE	7.500 ns
65583	WRITE	7.500 ns
65585	WRITE	7.500 ns
65587	WRITE	7.500 ns
65589	WRITE	7.500 ns
65591	WRITE	7.500 ns
65593	WRITE	7.500 ns
65595	WRITE	7.625 ns
65597	WRITE	7.500 ns
65599	WRITE	7.500 ns
65601	WRITE	7.500 ns

Figure 23. Listing window with all SDRAM Deselect commands hidden. It is clear that Refresh was the only active memory command before the trigger on the Activate command. After the Activate command, there is a long sequence of Write to Write operations.

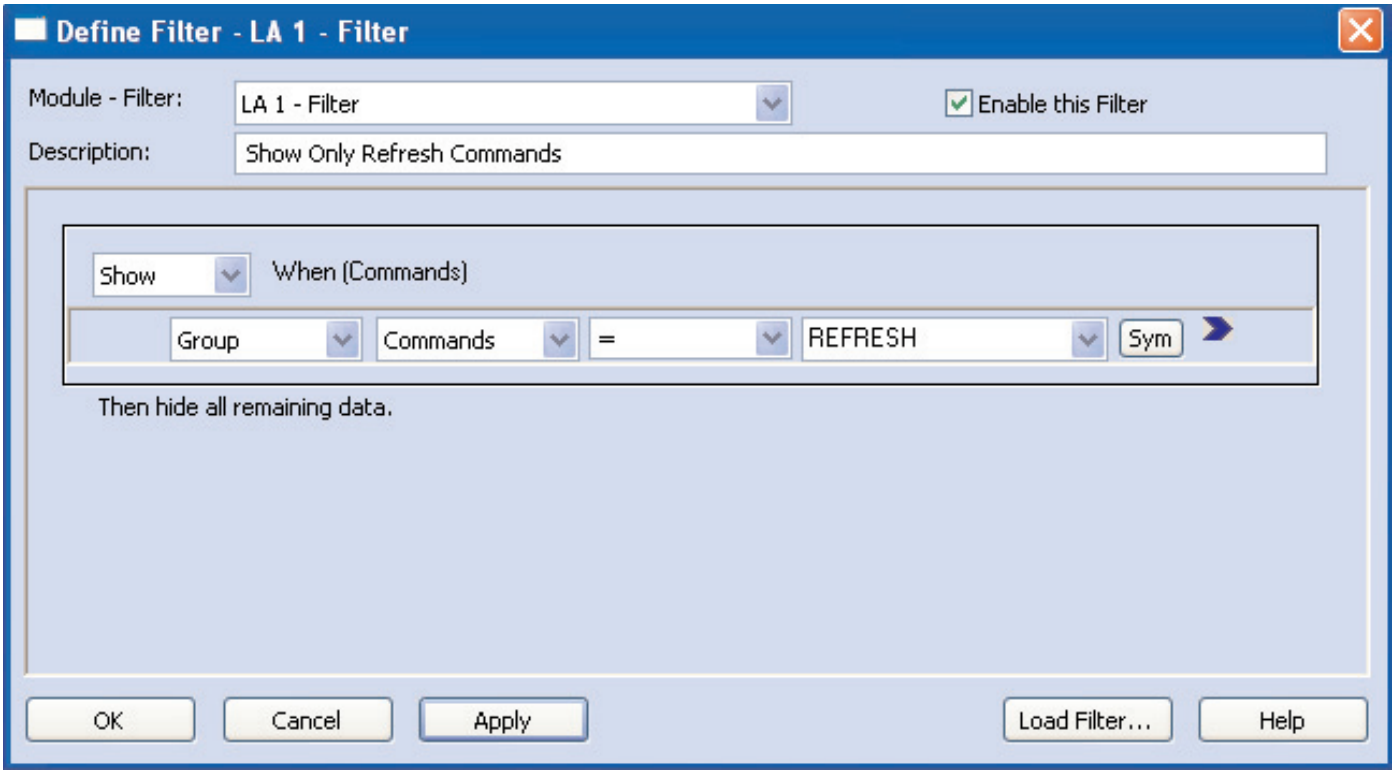


Figure 24. This display filter shows only SDRAM Refresh commands and is used to help verify the refresh operation.

All the memory cells of the DRAM need to be refreshed before the capacitor charge decays too low. It is the responsibility of the memory controller hub to refresh memory. In the early days of DRAM, the memory controller hub had to supply the row address that was to be refreshed during the refresh cycle. Today, the SDRAMs have an internal refresh controller that takes care of providing row addresses that are incremented each time the Refresh command is received by the SDRAM.

The example 512 Mb DDR2 SDRAM refresh cycle time is 64ms and the refresh count is 8K (8192). The refresh cycle time (64 ms) divided by the refresh count (8192) is 7.8125 µs; this is the average intervals of the Refresh-to-Refresh. There is some flexibility in the refresh operation and a maximum of eight consecutive Refresh cycles are allowed for improved efficiency for scheduling and switching between tasks.

Below we will verify that the Refresh cycles are at an average interval of 7.8125 µs. Consecutive Refresh cycles have a minimum time of 105 ns $t_{RFC(MIN)}$ between them. This specification is also for a Refresh-to-Active command interval.

A display filter is created to show only Refresh commands (See Figure 24). This filter is applied to the Listing window (See Figure 25). The listing window shows the entire 128K state acquisition record, which has been filtered down to a complete list. The first Refresh command was acquired at sample 580 and the last one was acquired at sample 129305. The timestamp measures the time between each Refresh command and averages around 7.8 µs.

Sample	Filter Commands	Timestamp
580	REFRESH	0 ps
2656	REFRESH	7.802,125 us
4732	REFRESH	7.802,500 us
6808	REFRESH	7.802,500 us
8884	REFRESH	7.802,625 us
10960	REFRESH	7.802,375 us
13036	REFRESH	7.802,000 us
15112	REFRESH	7.802,250 us
17188	REFRESH	7.802,125 us
19264	REFRESH	7.802,125 us
21340	REFRESH	7.801,750 us
23416	REFRESH	7.801,750 us
25492	REFRESH	7.802,250 us
27568	REFRESH	7.802,000 us
29644	REFRESH	7.802,125 us
31720	REFRESH	7.802,250 us
33796	REFRESH	7.802,125 us
35872	REFRESH	7.802,375 us
37948	REFRESH	7.802,250 us
40024	REFRESH	7.802,750 us
42100	REFRESH	7.802,750 us
44176	REFRESH	7.802,000 us
46252	REFRESH	7.802,125 us
48328	REFRESH	7.802,250 us
50404	REFRESH	7.802,000 us
52480	REFRESH	7.801,625 us
54556	REFRESH	7.802,375 us
56632	REFRESH	7.801,875 us
58708	REFRESH	7.802,375 us
60784	REFRESH	7.802,625 us
62860	REFRESH	7.801,875 us
64936	REFRESH	7.802,500 us
67025	REFRESH	7.851,500 us
69100	REFRESH	7.798,750 us
71177	REFRESH	7.806,125 us
73252	REFRESH	7.797,875 us
75329	REFRESH	7.806,000 us
77404	REFRESH	7.797,625 us
79481	REFRESH	7.805,875 us
81556	REFRESH	7.798,875 us
83633	REFRESH	7.806,500 us
85708	REFRESH	7.799,250 us
87785	REFRESH	7.807,000 us
89860	REFRESH	7.799,125 us
91937	REFRESH	7.805,750 us
94012	REFRESH	7.797,750 us
96089	REFRESH	7.805,250 us
98164	REFRESH	7.798,125 us
100241	REFRESH	7.806,125 us
102316	REFRESH	7.798,625 us
104393	REFRESH	7.806,125 us
106468	REFRESH	7.798,750 us
108545	REFRESH	7.806,000 us
110620	REFRESH	7.797,875 us
112697	REFRESH	7.806,250 us
114772	REFRESH	7.799,375 us
116849	REFRESH	7.806,375 us
118924	REFRESH	7.798,250 us
121001	REFRESH	7.805,250 us
123076	REFRESH	7.797,875 us
125153	REFRESH	7.805,500 us
127228	REFRESH	7.798,500 us
129305	REFRESH	7.806,250 us

Figure 25. Verifying Refresh-to-Refresh times in the Listing window with only Refresh commands shown.

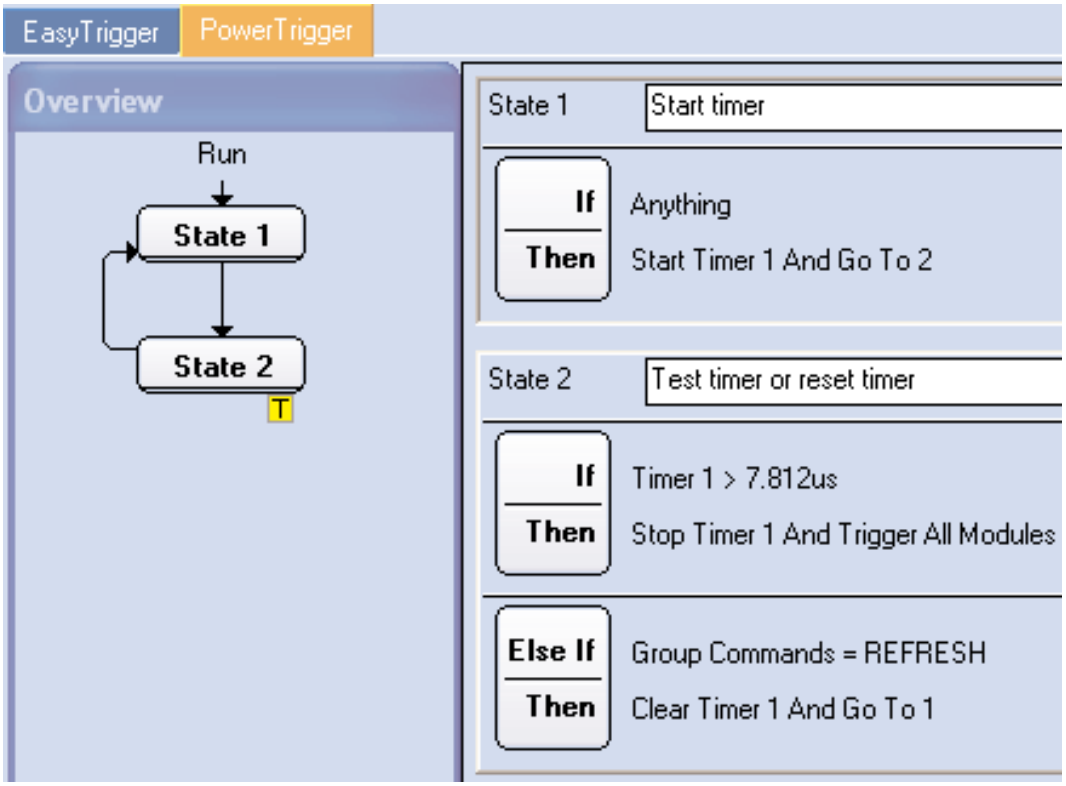
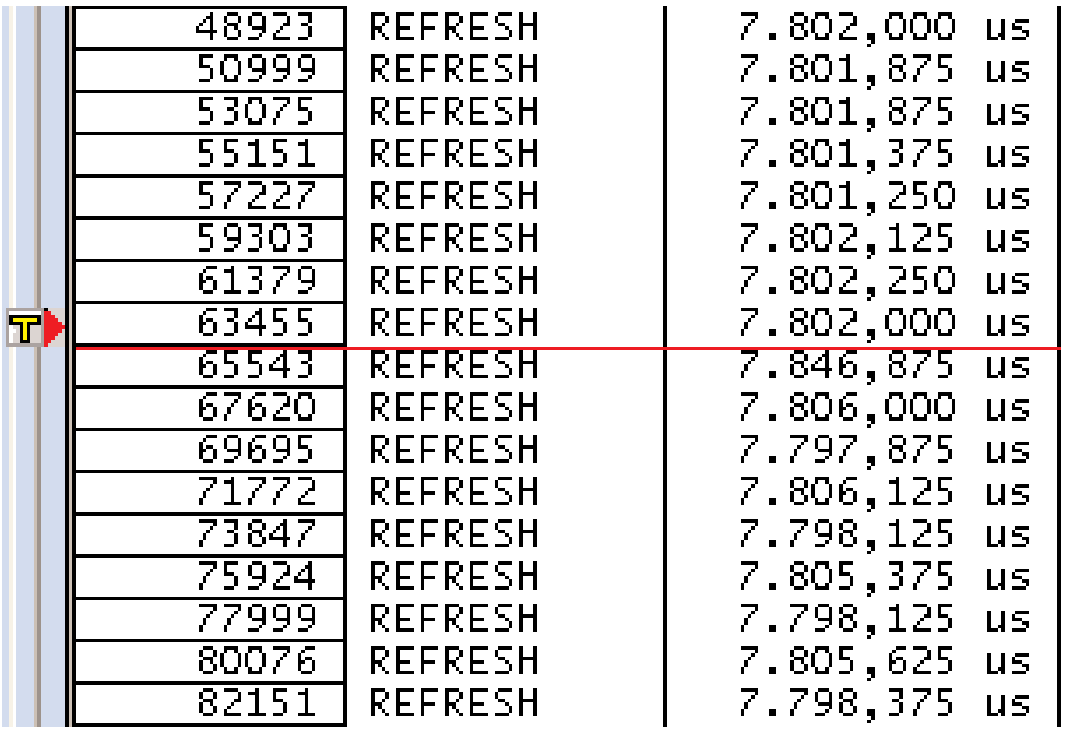


Figure 26. A PowerTrigger program is used to find any Refresh-to-Refresh commands with intervals greater than 7.812 µs.

Grabbing any data and visually checking is not a robust method for verifying the maximum average Refresh-to-Refresh time interval. A better way is by using a simple PowerTrigger program to trigger on a Refresh-to-Refresh time interval greater then a specified value (See Figure 26). The first trigger state starts the timer and the second state tests the timer for greater than 7.812 µs and if a Refresh command is found at that sample point. If the timer is greater than 7.812 µs the logic analyzer triggers (See Figure 27). Here the Refresh-to-Refresh time interval was 7.846,875 µs. The PowerTrigger program was changed to 7.9 µs and left to run for a long period. For this long time period, every Refresh-to-Refresh time interval was checked against 7.9 µs and no triggers occurred. We can, therefore, have confidence that the maximum Refresh-to-Refresh time interval is less than 7.9 µs. Notice in Figure 27 that the sample numbers are not sequential. This is a sure sign that a display filter is being used, which it was. The display filter showing only SDRAM Refresh commands in Figure 24 was used.



	48923	REFRESH	7.802,000 us
	50999	REFRESH	7.801,875 us
	53075	REFRESH	7.801,875 us
	55151	REFRESH	7.801,375 us
	57227	REFRESH	7.801,250 us
	59303	REFRESH	7.802,125 us
	61379	REFRESH	7.802,250 us
T	63455	REFRESH	7.802,000 us
	65543	REFRESH	7.846,875 us
	67620	REFRESH	7.806,000 us
	69695	REFRESH	7.797,875 us
	71772	REFRESH	7.806,125 us
	73847	REFRESH	7.798,125 us
	75924	REFRESH	7.805,375 us
	77999	REFRESH	7.798,125 us
	80076	REFRESH	7.805,625 us
	82151	REFRESH	7.798,375 us

Figure 27. A PowerTrigger program found Refresh-to-Refresh commands with intervals greater than 7.812 µs. In this case the refresh interval was 7.846,875 µs.

Various trigger resources such as counters, timers, setup/hold, glitches, 16 states, range recognizers, word recognizers, and transition recognizers, etc. are available to create PowerTrigger programs. A library of PowerTrigger programs is a valuable tool for verifying correct memory operation. For example, a PowerTrigger program can be created to measure and to trigger on the minimum time violation between the Activate command and the Write command.

Address and Bank Operation

So far only five signals have been used in verifying the DDR2 SDRAM operation. The Address and Bank signals are the next set of signals to be verified. These signals are sourced from the memory controller hub and are clocked into the memory on the rising edge of the SDRAM clock, just like the SDRAM commands.

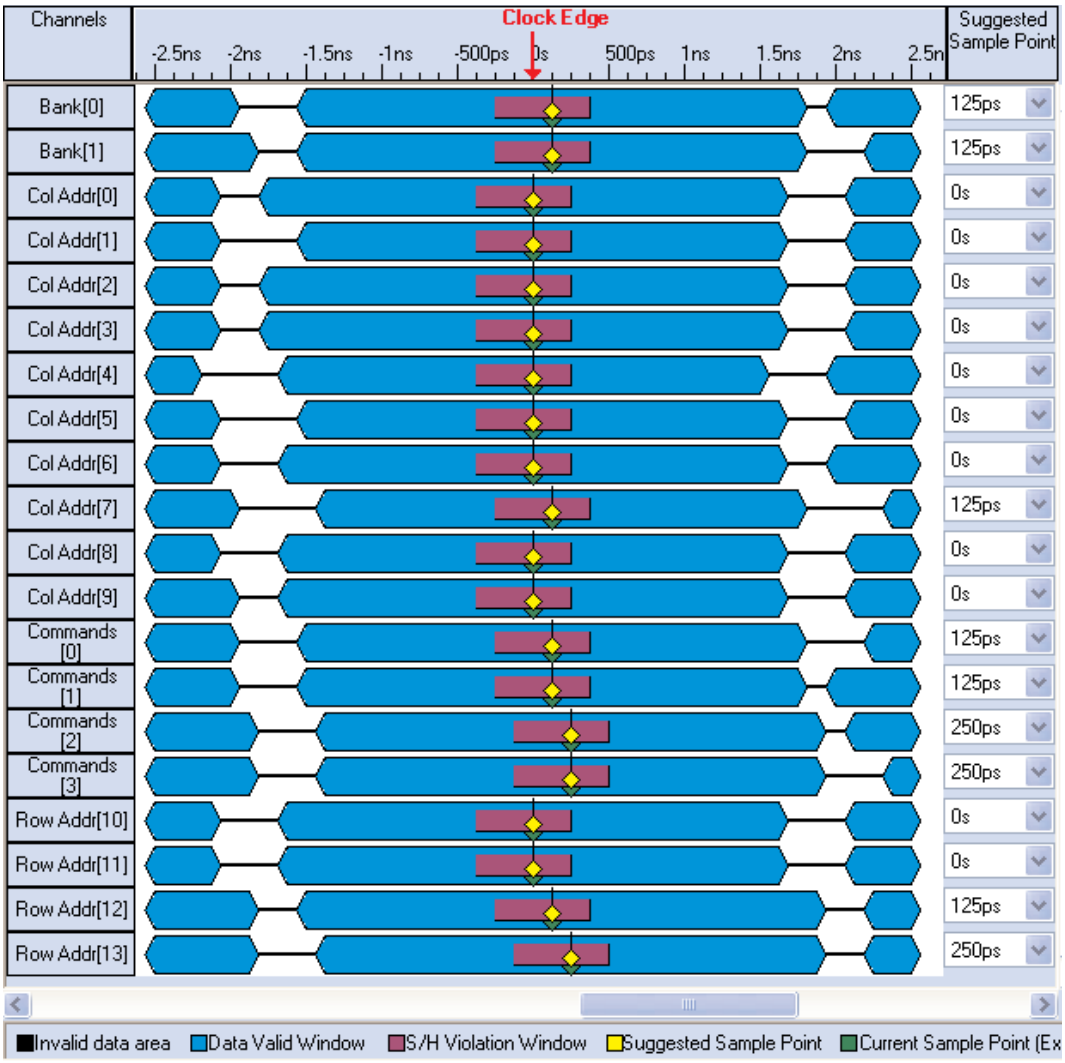


Figure 28. AutoDeskew analysis showing large data valid windows for bank addresses, column addresses, row addresses and commands.

DDR2 SDRAM comes in different sizes and different bank configurations. The 512 Mb DDR2 SDRAM uses A0-A13 for row address, BA0-BA1 for bank address and A0-A9 for column address. There are more row addresses than column addresses to reduce the memory power usage. Memory power increases as the number of columns increase. The row address and column address are multiplex on the same physical pins of the memory. As a result, the Activate command uses all A0-A13 for the row address. The Write command uses A0-A9 for the column address and A10 is used to enable or disable auto precharge. Auto precharge causes the row to be automatically precharged at the end of the write burst. The auto precharge closes the row and if auto precharge is not selected the row will remain open for subsequent write and read accesses.

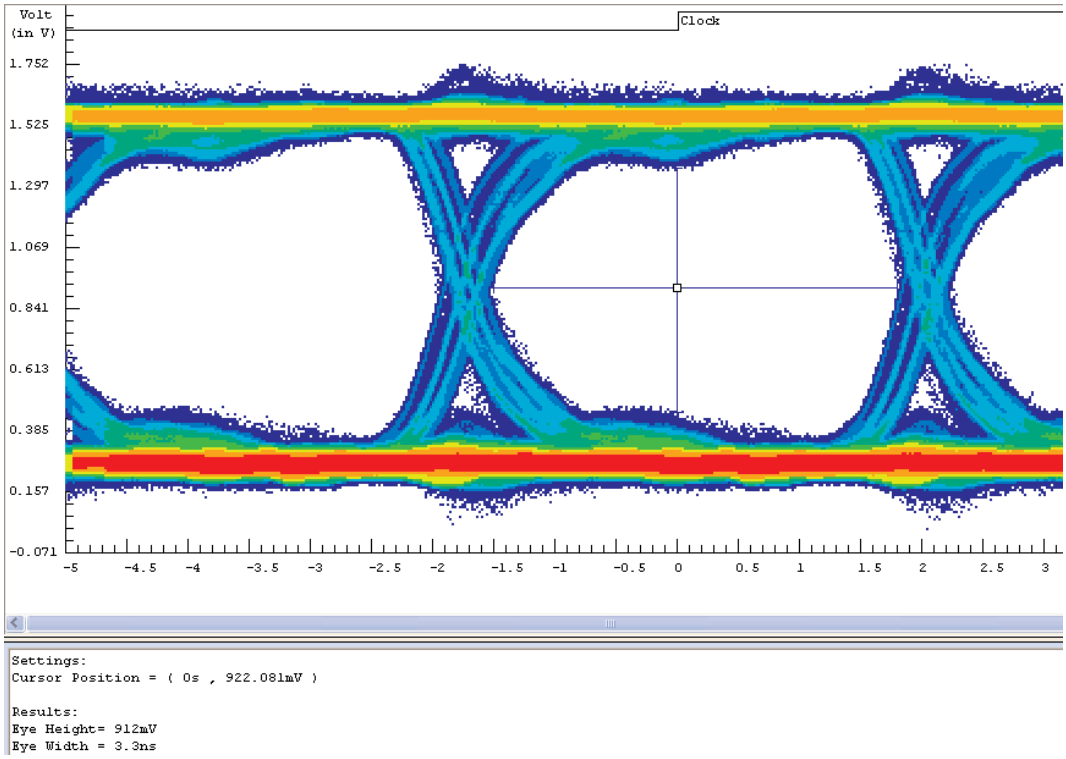
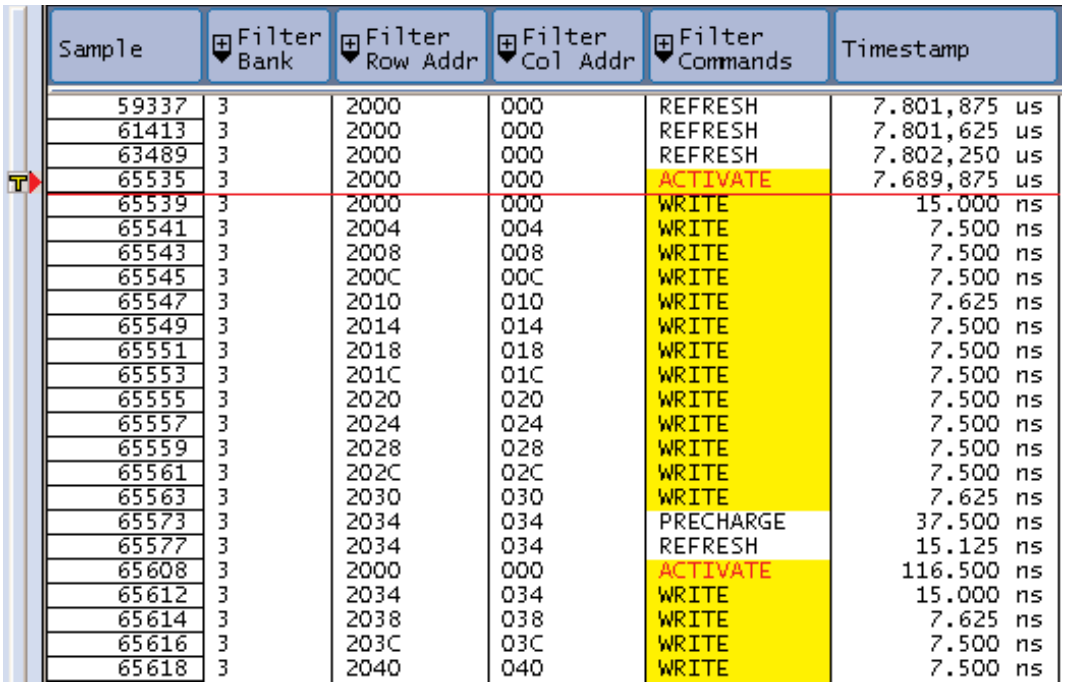


Figure 29. iVerify analysis showing large data valid windows for bank addresses, column addresses, row addresses and commands.

Memory usage statistics show that there are more reads than writes and that then next memory location is more likely to be read than any other memory location. This supports the concept that the computer reads sequential instructions more than any other memory operation. There is latency between the Activate command and Read command, but once the first read command executes, a series of Read commands can provide a continuous flow of data at the maximum data rate until the end of the memory row is reached. Therefore, row addresses are assigned to the least significant address bits of the memory controller hub.

For DDR2 SDRAM address bus and bank bus state acquisition, the logic analyzer data valid window or sample point position is configured the same way as it was for the command bus (See Figures 28 and 29). As expected, the eye diagram closed slightly when adding the two bank addresses and the 14 addresses. The multi-channel eye diagram went from 3.48 ns in Figure 18 for only the four commands signals to 3.3 ns in Figure 29 for all 20 signals.



Sample	Filter Bank	Filter Row Addr	Filter Col Addr	Filter Commands	Timestamp
59337	3	2000	000	REFRESH	7.801,875 us
61413	3	2000	000	REFRESH	7.801,625 us
63489	3	2000	000	REFRESH	7.802,250 us
65535	3	2000	000	ACTIVATE	7.689,875 us
65539	3	2000	000	WRITE	15.000 ns
65541	3	2004	004	WRITE	7.500 ns
65543	3	2008	008	WRITE	7.500 ns
65545	3	200C	00C	WRITE	7.500 ns
65547	3	2010	010	WRITE	7.625 ns
65549	3	2014	014	WRITE	7.500 ns
65551	3	2018	018	WRITE	7.500 ns
65553	3	201C	01C	WRITE	7.500 ns
65555	3	2020	020	WRITE	7.500 ns
65557	3	2024	024	WRITE	7.500 ns
65559	3	2028	028	WRITE	7.500 ns
65561	3	202C	02C	WRITE	7.500 ns
65563	3	2030	030	WRITE	7.625 ns
65573	3	2034	034	PRECHARGE	37.500 ns
65577	3	2034	034	REFRESH	15.125 ns
65608	3	2000	000	ACTIVATE	116.500 ns
65612	3	2034	034	WRITE	15.000 ns
65614	3	2038	038	WRITE	7.625 ns
65616	3	203C	03C	WRITE	7.500 ns
65618	3	2040	040	WRITE	7.500 ns

Figure 30. Listing window showing triggering on the Activate command and displaying the bank address, row address and column address.

The logic analyzer triggers on the Activate command with Deselect commands having been filtered from the display (See Figure 30). “Filter” is in the title of each data column to remind the user that the data is being filtered. The first and second Activate command uses the same row address of 2000 hex and the same bank address of 3. The first Write command uses the column address of 000 hex and the memory controller hub writes a bust of four data values in to the memory. The second Write command uses the column address of 004 hex since the first Write filled addresses 000, 001, 002 and 003 hex. The Write command just before the Precharge command has a column address of 030 hex. The memory controller hub Precharge command closes the open row at 2000 hex in bank 3. A refresh cycle is executed and then an Activate command opens the same row at 2000 hex in bank 3 and the next Write Command starts writing to the next column address 034 in bank 3.

Using the timestamp, you can quickly verify the timing of this command sequence. The first three Refresh commands intervals are the specified average of 7.8 µs. The Refresh to Activate command time is 7.689,875 µs which is much larger than the 105 ns specification minimum. The Activate to Write is at the specification minimum of 15 ns.

The consecutive Writes-to-Writes are the specification minimum of 7.5 ns, which is two clock cycles. With a burst length of four, every write operation takes two clock cycles to write all four data bits to the memory. A Deselect command is in between each Write commands as was discussed earlier in this application note. The timestamp varies slightly because of its 125 ps resolution with respect to the 3.759 ps SDRAM clock period.

The Write to Precharge is composed of three time intervals. The first is write latency (WL), which is 3 clock cycles (3.759 ns*3=11.277 ns). Next is two clock cycles (3.759 ns*2=7.518 ns) to write the burst of four data bits in to the memory. Last is the Write recovery time of 15 ns (MIN). Adding these three intervals together equals to 33.795 ns (MIN), which is less than the actual 37.5 ns measurement in Figure 30.

The Precharge to Refresh time is 15.125 ns, which is very close to the Precharge command period of 15 ns (MIN). The second Refresh to Activate 116 ns interval time is greater than the specified Refresh to Activate command interval of 105 ns (MIN). The rest of the Listing window's consecutive Writes-to-Writes are 7.625 ns to 7.500 ns.

Verifying the last Refresh to Refresh can be done several different ways. First, you can add the timestamp intervals. Secondly, it is easier and less error prone to use the cursors. Move cursor 2 to the last Refresh and cursor 1 to the previous Refresh, which is just above the first Activate command. Then use the delta time readout to measure cursor 1 to cursor 2, which is 7.847 µs and is the specified average refresh time.

The Listing window has to be read carefully. The column addresses A0-A9 are the same physical pins as the A0-A9 addresses of the row address. But the row addresses use A0-A13. Therefore, use the 14-bit row addresses for the Activate commands and the 10-bit column addresses for the Write commands.

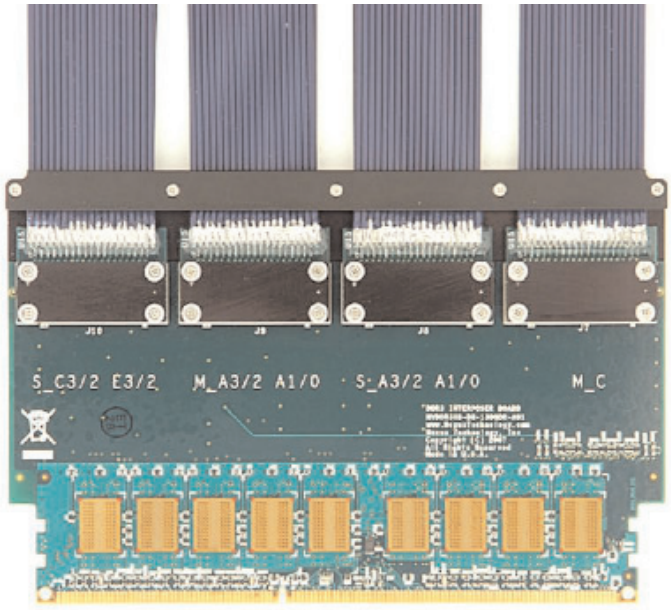


Figure 31. Nexus DDR3 NEXVu DIMM designed to JEDEC standards with logic analyzer probe points. This provides unmatched signal fidelity and allows the capture of the actual eye patterns as seen by the memory components.

Logic Analyzer Memory Supports

Logic analyzer memory supports enhance the operation of the logic analyzer by configuring the logic analyzer setup, providing custom clocking for memory acquisition, memory data analysis software, mnemonics listing, and may include memory probing hardware. Nexus Technology, Inc. (www.NexusTechnology.com/) provides logic analyzer memory supports and is a Tektronix Embedded Systems Tools Partner that provides complementary products for Tektronix logic analyzers and oscilloscopes. Tektronix distributes selected Nexus Technology products.

Nexus logic analyzer DDR3/DDR2 protocol violation software automates the analysis of a DDR2 or DDR3 bus to quickly and easily identify protocol violations, frequency of those violations, and also provide a global view of all DDR commands in the logic analyzer memory.

Determining the appropriate sample point to acquire DDR3 data can be difficult and time consuming because of the small data valid window and low threshold voltages. The Nexus DDR3 Sample Point Analyzer (SPA) Software reviews the logic analyzer's MagniVu data and automatically adjusts the threshold to optimal values. It also determines the best sample point for each data group and allows the user to review the results and apply them. This provides for quick and easy setup of the logic analyzer to acquire accurate data on a DDR3 bus.Nexus logic analyzer memory supports are used for DDR, DDR2, DDR3, DIMM, SO-DIMM, FB-DIMM, and other new emerging memory technologies. The probing hardware ranges from none for direct probing of embedded memory systems to DIMM interposers and instrumented DIMMs. Nexus calls an instrumented DIMM a NEXVu DIMM. (See Figure 31).

DDR2 NEXVu DIMMs are higher than a normal DIMM so that logic analyzer probes can be attached to it with normal height DIMMs in the adjacent slots. The logic analyzer probes are attached to either side of the DDR2 NEXVu DIMM. Two DDR2 NEXVu DIMMs can be in adjacent slots with the logic analyzer probes attaching on opposite sides of each DDR2 NEXVu DIMM.

NEXVu DIMMs have inner layer circuit board isolation resistors close to the FBGA IC packages to reduce the loading affect of the probing circuit board traces attached to them. In Figure 31, you can see the large AMB with its heat sink and 8 of the 18 DDR2 667 SDRAMs.

Also in Figure 31, you can see the 24 high-speed differential circuit board traces that go from the edge connector to the Advanced Memory Buffer (AMB). Notice that the length of a trace increases the further it is from the center of the FB-DIMM. In a FB-DIMM memory system there are no serpentine trace patterns to equalize the trace length of the 24 differential signals. This simplifies circuit board layout. Fewer signals and simpler circuit board layout contribute to having more FB-DIMM channels per memory system as compared to using UDIMMs and RDIMMs. The signal skew caused by unequal circuit board traces is alleviated by the AMB during the training part of the FB-DIMM channel initialization.

Sample	Address	FBDNXV2A Mnemonics
30	00A4A	MRS - MODE REGISTER SET Normal MRS PD Mode: Standard Write Recovery: 6 DLL Reset: No Operating Mode: Normal Latency: 4 Burst Type: Interleaved Burst: 4
31	10380	MRS - MODE REGISTER SET Extended MRS RDQS Enable: No DQS# Enable: Enable OCD Operation: OCD Calibratio> Rtt: Disabled Additive Latency: 0 Output Drive Strength: 100% DLL Enable: Enable (Normal)

Figure 32. The DDR2 SDRAM Mode Register (MR) and Extended Mode Register configurations are captured with the logic analyzer during the Mode Register commands issued by the memory controller hub.

Acquiring Data with NEX-FBD-NEXVu

Configuring NEX-FBD-NEXVu for operation takes only a few steps. The first step is to configure the Command group sample point position for the logic analyzer data valid window. This is done just like setting the Command group sample point earlier in this application note. MagniVu or AutoDeskew is used to determine the optimum logic analyzer sample point position for each of the SDRAM command signals. Use iVerify to check the signals data valid window with a multiple channel eye diagram.

NEX-FBD-NEXVu custom clocking provides several clocking modes to make efficient use of the logic analyzer memory. The first step involves two choices. First, there is a choice between acquiring every DDR2 DRAM rising clock edge or selective clocking based on RAS# and CAS# commands. The second choice is between saving Refresh cycles or reducing the number of Refresh cycles saved.

The second step is adding columns to the Listing window and capturing the SDRAM Mode Register (MR) and Extended Mode Register configurations to determine the CAS Latency (CL), Burst Length (BL) and Additive Latency (AL). A 16-bit value is programmed into these registers by the memory controller hub during the Mode Register command using the addresses A0-13 and bank addresses BA0-1.

Mode Register values are needed in order to know where to look for the read and write data valid windows. The Mode Register values are four for the CAS Latency, four for the Burst Length and zero for the Additive Latency (See Figure 32). These values are used to determine the locations of the write and read data valid windows and to configure the logic analyzer disassembly in the Listing window.

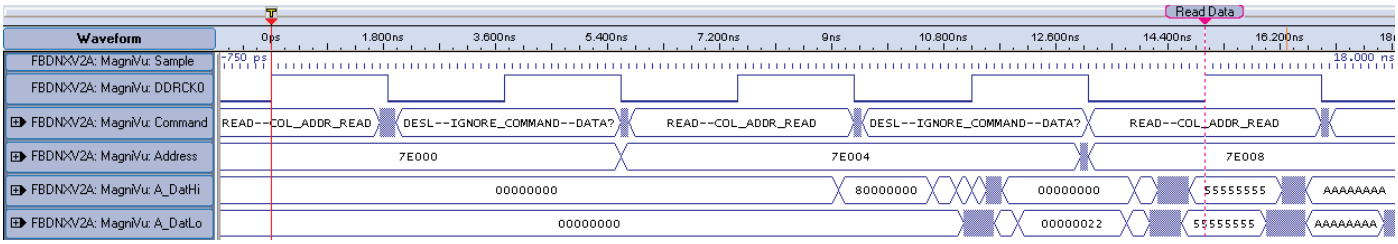


Figure 33. Trigger on the first Read command after the Activate command. Count the four Read Latency (RL) clock edges to find the first read data cycle shown with a Read Data marker. The DDR2 SDRAM 64-bit read data is an alternating pattern of 5555,5555,5555,5555 hex and AAAA,AAAA,AAAA,AAAA hex.

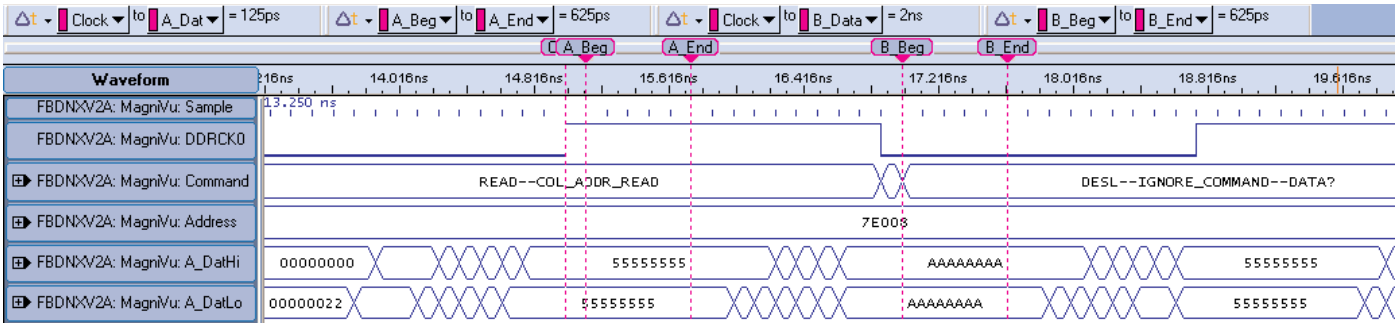


Figure 34. MagniVu high-resolution timing measures the read data valid windows to set the logic analyzer sample point position. The DDR2 SDRAM 64-bit read data is an alternating pattern of 5555,5555,5555,5555 hex and AAAA,AAAA,AAAA,AAAA hex.

The memory support provides bit decoding of the 16-bit Mode Register and Extended Mode Registers. Figure 32 shows how the Mode Register in sample 30 has been decoded into nine lines of mnemonics. The symbol table used in Figure 12 cannot provide complex multi-line mnemonic decoding as shown in Figure 32. Mnemonic decoding is a significant benefit of memory supports.

The third step is to trigger on an Activate-Read command sequence and use MagniVu high-resolution timing waveforms to determine the two read data valid windows per clock cycle. It is usually easier to confirm the first read data at the beginning of the Activate-Read command sequence than triggering on any Read command. This also makes it easier to verify that the Read Latency is correct.

The Read Latency (RL) determines the start of the read burst data. Add CAS Latency (CL) to Additive Latency (AL) to determine the Read Latency (RL) and count the number of RL rising clock edges from the Read command to find the first read data cycle of the Read command (See Figure 33). There are two read data clock cycles and four read data clock cycles for a burst length of four and eight, respectively.

The memory support divides the 64-bit memory data into a 32-bit DatHi group and a 32-bit DatLo group (See Figure 34). In this case the two merged TLA7AA4 logic analyzer modules are operating in External 2X clocking mode for the data lines. This means the logic analyzer uses two memory channels, each with a different sample point position to acquire the DDR2 SDRAM data on a single data I/O line during a single clock cycle. Four sample points, consisting of two groups with two samples points each, are configured in the Setup window custom clocking to acquire the read data.

A Waveform window marker is placed at the rising edge of the clock and renamed Clock. Markers are represented by the five vertical dashed lines visible in Figure 34. A second A_Beg marker is placed behind the start of the first read data valid window that contains 5555,5555,5555,5555 hex. A second A_End marker is placed 625 ps later from the A_Beg marker. This A_Beg to A_End 625 ps time interval is the required time that the read data needs to be stable and unchanging in order for the TLA7AA4 logic analyzer to acquire it reliably.

Sample	Address	FBDNXV2A Mnemonics	FBDNXV2A DataHi	FBDNXV2A DataLo	FBDNXV2A BChekBits	Timestamp
65530	4E3FC	DESL - IGNORE COMMAND	--------	--------	00	3.750 ns
65531	4E3FC	DESL - IGNORE COMMAND	--------	--------	00	3.875 ns
65532	4E3FC	DESL - IGNORE COMMAND	--------	--------	00	3.625 ns
65533	4E000	PRE - PRECHARGE SELECT BANK	--------	--------	00	3.875 ns
65534	7E000	ACTV - ROW ADDRESS STROBE	--------	--------	00	15.000 ns
65535	7E000	READ - COL ADDR READ	--------	--------	00	15.000 ns
65536	7E000	DESL - IGNORE COMMAND	--------	--------	00	3.750 ns
65537	7E004	READ - COL ADDR READ	--------	--------	00	3.750 ns
65538	7E004	DESL - IGNORE COMMAND	--------	--------	00	3.750 ns
65539	7E008	READ - COL ADDR READ	--------	--------	00	3.750 ns
		READ DATA	55555555	55555555	00	
		READ DATA	AAAAAAAA	AAAAAAAA	00	
65540	7E008	READ DATA	55555555	55555555	00	3.750 ns
		READ DATA	AAAAAAAA	AAAAAAAA	00	
65541	7E00C	READ - COL ADDR READ	--------	--------	00	3.750 ns
		READ DATA	55555555	55555555	00	
		READ DATA	AAAAAAAA	AAAAAAAA	00	
65542	7E00C	READ DATA	55555555	55555555	00	3.750 ns
		READ DATA	AAAAAAAA	AAAAAAAA	00	
65543	7E010	READ - COL ADDR READ	--------	--------	00	3.750 ns

Figure 35. Nexus Technology, Inc. NEX-FBD-NEXVu Listing window with DDR2 SDRAM read data of alternating pattern of 5555,5555,5555,5555 hex and AAAA,AAAA,AAAA,AAAA hex.

A Delta time measurement of 125 ps between the Clock and A_Beg marker provides the setup time that is configured in the custom clocking setup. Similarly, B_Beg and B_End markers are used for determining the second data valid window of the DDR2 SDRAM clock cycle. In Figure 34, the first logic analyzer read data valid window A begins 125 ps after the rising edge of the clock. The second logic analyzer read data valid window B begins 2 ns after the rising edge of the clock.

Both write and read data are captured if three-merged TLA7AA4 logic modules are being used. Therefore, to configure the logic analyzer write sample point positions trigger on a Activate-Write command sequence and use MagniVu high-resolution timing waveforms to determine the logic analyzer data valid windows for write data. Note that Write Latency (WL) is one less than the Read Latency (RL) in determining the location of the write data.

The DDR2 SDRAM data line has four data valid windows for a single clock cycle. The logic analyzer is configured to acquire all four data valid windows for each clock cycle. It does this by using custom clocking in External 4X mode for three-merged TLA7AA4 logic analyzer modules. After the logic analyzer completes an acquisition it analyzes the SDRAM Commands to determine if it is a Write or Read cycle and then uses the Additive Latency (AL), CAS Latency (CL) and Burst length (BL) to determine where the valid data cycles are located in logic analyzer acquired data. The user sets the cycle type, Additive Latency, CAS Latency and Burst Length in the Listing window Mnemonics column Disassembly properties.

The memory support Listing window shows three lines of mnemonics for samples 65539 and 65541 (See Figure 35). In one DDR2 SDRAM clock cycle, the memory controller hub sent the Read command to the DDR2 SDRAM and in the same clock cycle two 64-bit data reads occurred from a previous Read command. NEX-FBD-NEXVu memory support provides this analysis, decoding and mnemonics. The NEX-FBD-NEXVu also provides four display modes with the Listing window Mnemonics column Disassembly properties: display all acquired cycles, suppress all idle or wait cycles, show Activate, Write and Read cycles, and show only Write and Read data cycles.

Summary

This application note has shown the power of the logic analyzer in verifying SDRAM protocols sequencing and timing. A lot can be verified by just looking at the SDRAM clock and four SDRAM command signals. Complete verification uses Nexus Technology memory supports, memory interposers and NEXVu DIMMs.

Tektronix also offers a comprehensive tool set including industry-leading oscilloscopes, true differential TDRs, and logic analyzers with Nexus Technology memory supports to enable embedded and computer designers to perform quick and accurate electrical testing and operational validation of their memory designs. Collectively, this tool set provides superior performance with unparalleled ease-of-use, making it an ideal solution for embedded systems and computer memory systems verification and debugging.

Contact Tektronix:

ASEAN / Australasia (65) 6356 3900
Austria +41 52 675 3777
Balkans, Israel, South Africa and other ISE Countries +41 52 675 3777
Belgium 07 81 60166
Brazil +55 (11) 40669400
Canada 1 (800) 661-5625
Central East Europe, Ukraine and the Baltics +41 52 675 3777
Central Europe & Greece +41 52 675 3777
Denmark +45 80 88 1401
Finland +41 52 675 3777
France +33 (0) 1 69 86 81 81
Germany +49 (221) 94 77 400
Hong Kong (852) 2585-6688
India (91) 80-42922600
Italy +39 (02) 25086 1
Japan 81 (3) 6714-3010
Luxembourg +44 (0) 1344 392400
Mexico, Central/South America & Caribbean 52 (55) 54247900
Middle East, Asia and North Africa +41 52 675 3777
The Netherlands 090 02 021797
Norway 800 16098
People's Republic of China 86 (10) 6235 1230
Poland +41 52 675 3777
Portugal 80 08 12370
Republic of Korea 82 (2) 6917-5000
Russia & CIS +7 (495) 7484900
South Africa +27 11 206 8360
Spain (+34) 901 988 054
Sweden 020 08 80371
Switzerland +41 52 675 3777
Taiwan 886 (2) 2722-9622
United Kingdom & Ireland +44 (0) 1344 392400
USA 1 (800) 426-2200
For other areas contact Tektronix, Inc. at: 1 (503) 627-7111

Updated 30 October 2008

For Further Information
Tektronix maintains a comprehensive, constantly expanding collection of application notes, technical briefs and other resources to help engineers working on the cutting edge of technology. Please visit www.tektronix.com

06/09 JS/WWW 52W-19236-2

Tektronix®