DDR SDRAM Controller White Paper

DDR SDRAM Controller Description

The Double Data Rate(DDR) Synchronous Dynamic Random Access Memory(SDRAM) Controller provides a simplified interface to industry standard DDR SDRAM memory. The SDRAM controller reference design is available in Verilog and VHDL that is optimized for the Altera® APEX™ architecture. The DDR SDRAM controller supports the following features:

- SDRAM burst lengths of 2,4, or 8.
- CAS latency of 1.5, 2.0, 2.5, or 3.0.
- 16 bit programmable refresh counter used for automatic refresh.
- Support for 2 banks of SDRAM devices.
- Supports the following commands: NOP, READA, WRITEA, AUTO_REFRESH, PRECHARGE, ACTIVATE and LOAD_MR.
- Data Mask lines are supported for write operations.
- Supports SDRAM data path widths of 16, 32, and 64 bits.
- Utilizes two PLL's to increase system performance.
- Provides a peak performance of 1.6Gbytes/s with a system clock of 100Mhz and a 64-bit DDR SDRAM.

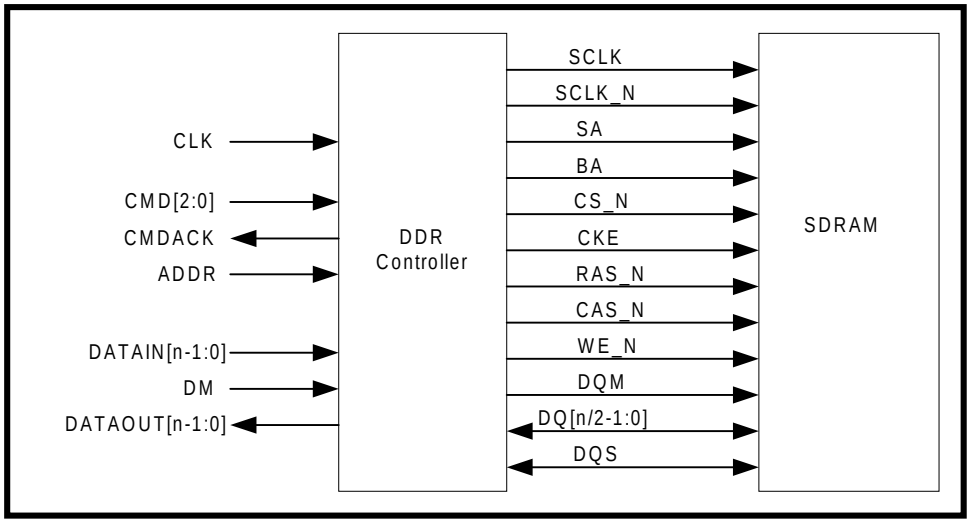


Figure 1 DDR SDRAM Controller System Level Diagram

SDRAM Overview

SDRAM is high-speed Dynamic Random Access Memory (DRAM) with a synchronous interface. The synchronous interface and fully pipelined internal architecture of SDRAM allows extremely fast data rates if used efficiently. SDRAM is organized in banks of memory addressed by row and column. The number of row and column address bits and number of banks depends on the size of the memory.

SDRAM is controlled by bus commands that are formed using combinations of the RASN, CASN, and WEN signals. For instance, on a clock cycle where all three signals are high, the associated command is a No Operation (NOP). A NOP is also indicated when the chip select is not asserted. The standard SDRAM bus commands are shown in Table 1.

Command	RASN	CASN	WEN
No Operation (NOP)	H	H	H
Active (ACT)	L	H	H
Read (RD)	H	L	H
Write (WR)	H	L	L
Burst Terminate (BT) 1	H	H	L
Precharge (PCH)	L	H	L
Autorefresh (ARF)	L	L	H
Load Mode Register (LMR)	L	L	L

Table 1 SDRAM Bus Commands

Notes:

1- Burst Terminate is not supported in this version of the reference design.

SDRAM banks must be opened before a range of addresses can be written to or read from. The row and bank to be opened are registered coincident with the Active (ACT) command. When a bank is accessed for a read or a write it may be necessary to close the bank and re-open it if the row to be accessed is different than the row that is currently opened. Closing a bank is done with the Precharge (PCH) command.

The primary commands used to access SDRAM are Read (RD) and Write (WR). When the WR command is issued, the initial column address and data word is registered. When a RD command is issued, the initial address is registered. The initial data appears on the data bus 1-3 clock cycles later. This is known as CAS latency (CL) and is due to the time required to physically read the internal DRAM core and register the data on the bus. The CAS latency depends on the speed of the SDRAM and the frequency of the memory clock. In general, the faster the clock, the more cycles of CAS latency required. After the initial RD or WR command, sequential read and writes will continue until the burst length is reached or a Burst Terminate (BT) command is issued. DDR SDRAM memory devices support burst lengths (BL) of 2,4, or 8 data cycles. DDR devices double the data rate by clocking data in and out of the device on every clock edge. The Auto Refresh command (ARF) is issued periodically to insure data retention. This function is performed by the SDRAM Controller and is transparent to the user.

The Load Mode Register command (LMR) is used to configure the SDRAM mode register. This register stores the CAS latency, burst length, burst type, and write burst mode. Consult the SDRAM specification for additional details.

SDRAM come in Dual In-line Memory Modules (DIMM) and as chips. To reduce pin count SDRAM row and column addresses are multiplexed onto the same pins. SDRAM often includes more than one bank of memory internally and may DIMMS may require multiple chip selects.

DDR SDRAM Controller Description

DDR SDRAM Controller Interface Signals

The DDR SDRAM Controller interface signals are shown in Table 2 Interface Signals. All signals are synchronous to the system clock and outputs are registered at the output of the SDRAM Controller Core.

Signal	Name	Active	I/O	Description
User interface to the SDRAM controller				
CLK	Clock	NA	Input	System Clock.
RESET_N	Reset	LOW	Input	System Reset.
ADDR[ASIZE-1:0]	Memory Address	NA	Input	Memory address for read/write requests. Width is set by ASIZE.
CMD[2:0]	Command	NA	Input	Command request.
CMDACK	Command Acknowledge	HIGH	Output	Acknowledgment of the requested command.
DATAIN[DSIZE-1:0]	Input Data	NA	Input	Input Data Bus. Width is set by DSIZE.
DATAOUT[DSIZE-1:0]	Output Data	NA	Output	Output Data Bus. Width is set by DSIZE.
DM[(DSIZE/8)-1:0]	Data Mask	HIGH	Input	Masks individual bytes during data write
SDRAM controller interface with DDR SDRAM				
SCLK	Clock	NA	Input	System Clock
SCLK_N	Clock	NA	Input	Inverted System Clock
SA[11:0]	Address Bus	NA	Output	SA[11:0] is sampled during the ACT command to latch the row address. SA[n:0] is sampled during the RD/WR command to latch the column address where n depends on the size of SDRAM used. SA[10] is sampled during the PCH command to determine if all banks are to be precharged or the bank selected by BA[1:0]. The address outputs also provide the op-code during the LMR command.
BA[1:0]	Bank Address	NA	Output	These signals determine to which bank the ACT, RD, WR, or PCH command is being applied.
CS_N[1:0]	Chip Selects	LOW	Output	SDRAM chip selects.

CKE	Clock Enable	HIGH	Output	SDRAM CKE input.
RAS_N	Row Address Strobe	LOW	Output	SDRAM Command input.
CAS_N	Column Address Strobe	LOW	Output	SDRAM Command input.
WE_N	Write Enable	LOW	Output	SDRAM Command input.
DQ[DSIZE/2-1:0]	Data Bus	NA	I/O	SDRAM Data Bus.
DQM[(DSIZE/8)-1:0]	Data Mask	HIGH	Output	SDRAM Data Masks, mask individual bytes during data write.
DQS[DSIZE/8-1:0]	Data Strobe	NA	Output	SDRAM Data strobe, strobes data into the DDR devices during a write operation.

Table 2 Interface Signals

DDR SDRAM Controller Command Interface

The DDR SDRAM Controller provides a synchronous command interface to the SDRAM along with several control registers. The commands are listed in Table 3 with detailed descriptions in following sections. Commands are issued to the controller as follows:

- A command, other than NOP, is driven, by the user, onto CMD[2:0] with ADDR and DATAIN set appropriately for the requested command. The controller registers the command on the next rising clock edge.
- To acknowledge the command the controller will assert CMDACK for one clock period.
- If the command was READA or WRITEA, the user will need to start receiving or writing data on the DATAOUT and DATAIN ports. Refer to the READA and WRITEA timing diagram for more detail.
- The user must drive NOP onto CMD[2:0] by the next rising edge of CLK after CMDACK has been asserted.

Command	Value	Description
NOP	000b	No Operation.
READA	001b	SDRAM Read with Auto precharge.
WRITEA	010b	SDRAM Write with Auto precharge.
REFRESH	011b	SDRAM Auto refresh Refresh
PRECHARGE	100b	SDRAM Precharge all banks.
LOAD_MODE	101b	SDRAM Load Mode Register.
LOAD_REG1	110b	Load controller configuration register.
LOAD_REG2	111b	Load controller refresh period register.

Table 3 Interface Commands

NOP Command

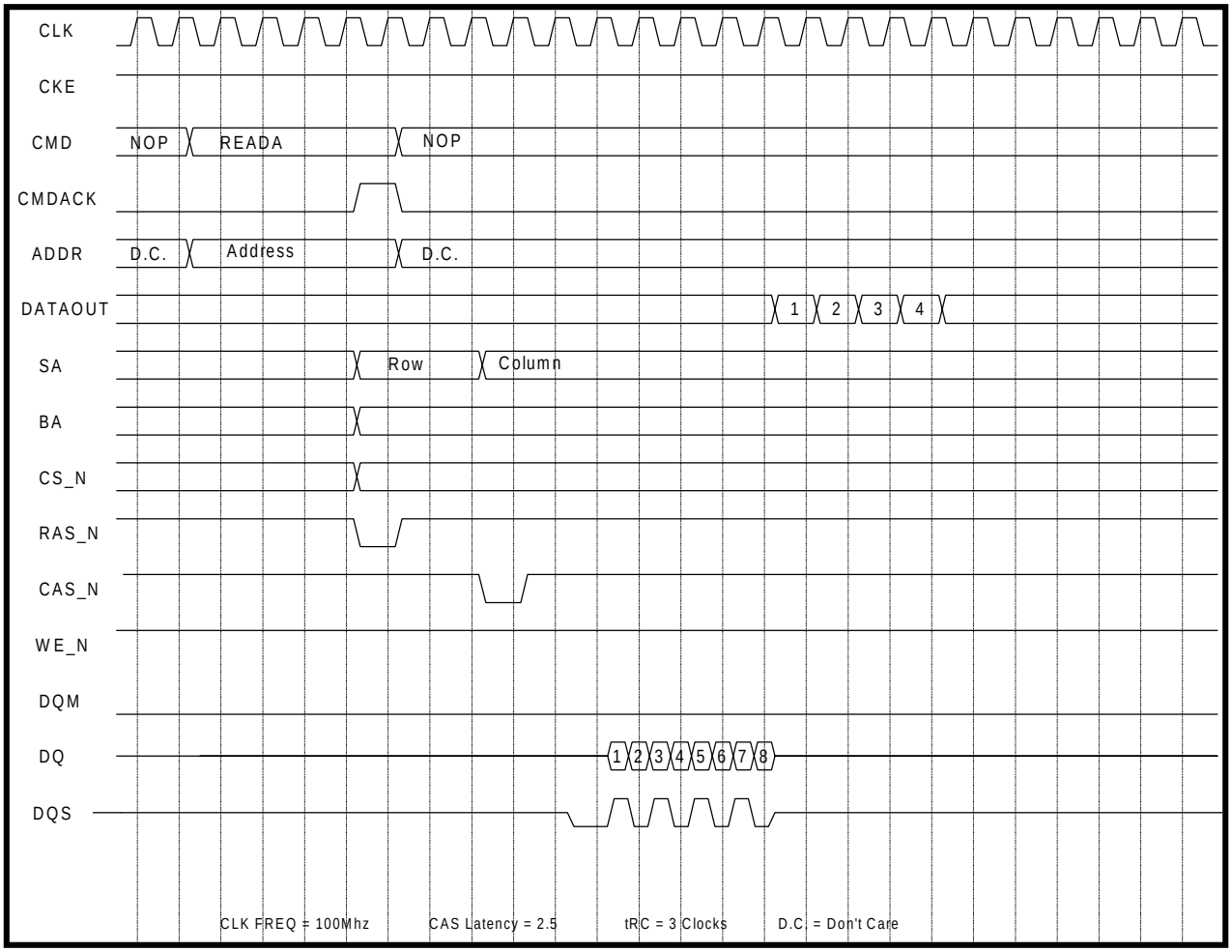
The NOP is a “No Operation” command to the controller. When detected, the controller performs a NOP in the following clock cycle. A NOP must be issued the following clock cycle after the controller has acknowledged a command. The NOP command has no effect on SDRAM accesses that are already in progress.

READA Command

The READA command instructs the controller to perform a burst read with auto-precharge to the SDRAM at the memory address specified by ADDR. The controller will issue an ACTIVATE command to the SDRAM followed by a READA command. The read burst data will first appear on DATAOUT (RC + CL + 4)clock cycles after the controller has asserted command acknowledge (CMDACK), where CL is the integer portion of the actual CAS Latency(i.e. if CAS Latency is 2.5 then CL = 2). Because the data interface to the DDR devices is one half the width of the DATAOUT port, the effective burst length of the data from DATAOUT is one half the burst length that the DDR devices are configured for. During a READA command the user must keep the DM inputs low. The timing for a READA command is shown in Figure 2. The operation of a READA command is as follows:

- The user asserts READA along with ADDR and DM.
- The controller acknowledges the command by asserting CMDACK.
- Simultaneously with asserting CMDACK the controller starts issuing commands to the SDRAM devices.
- One clock after CMDACK was asserted, the user must assert NOP CMD.
- (RC + CL + 4)clock cycles following assertion of CMDACK by the controller, the controller presents the first read burst value on DATAOUT, with the remainder of the burst following every clock there after.

Figure 2 READA Timing Diagram



WRITEA Command

The WRITEA command instructs the controller to perform a burst write with auto-precharge to the SDRAM at the memory address specified by ADDR. The controller will issue an ACTIVATE command to the SDRAM followed by a WRITEA command. The first data value in the burst sequence must be presented with the WRITEA and ADDR address. The host must start clocking data into the controller tRC-2 clocks after the controller has acknowledged the WRITEA command. The controller runs the DQ port to the DDR devices at one half the width and twice the rate of the DATAIN port. This results in burst lengths at the DATAIN port being one half the length that the DDR devices are configured for. The DM inputs are clock through the controller with the data and sent to the SDRAM devices on the DQM outputs. See a SDRAM data sheet for how to use the DQM lines. The timing for a WRITEA command is shown in Figure 3. The operation of a WRITEA command is as follows:

- The user asserts WRITEA along with ADDR and the first write data value on DATAIN and the desired data mask value on DM.
- The controller acknowledges the command by asserting CMDACK.
- Simultaneously with asserting CMDACK the controller starts issuing commands to the SDRAM devices.
- One clock after CMDACK was asserted, the user asserts NOP on CMD.

- tRC-2 following the assertion of CMDACK by the controller, the user clocks data and data mask values into the controller through DATAIN and DM.

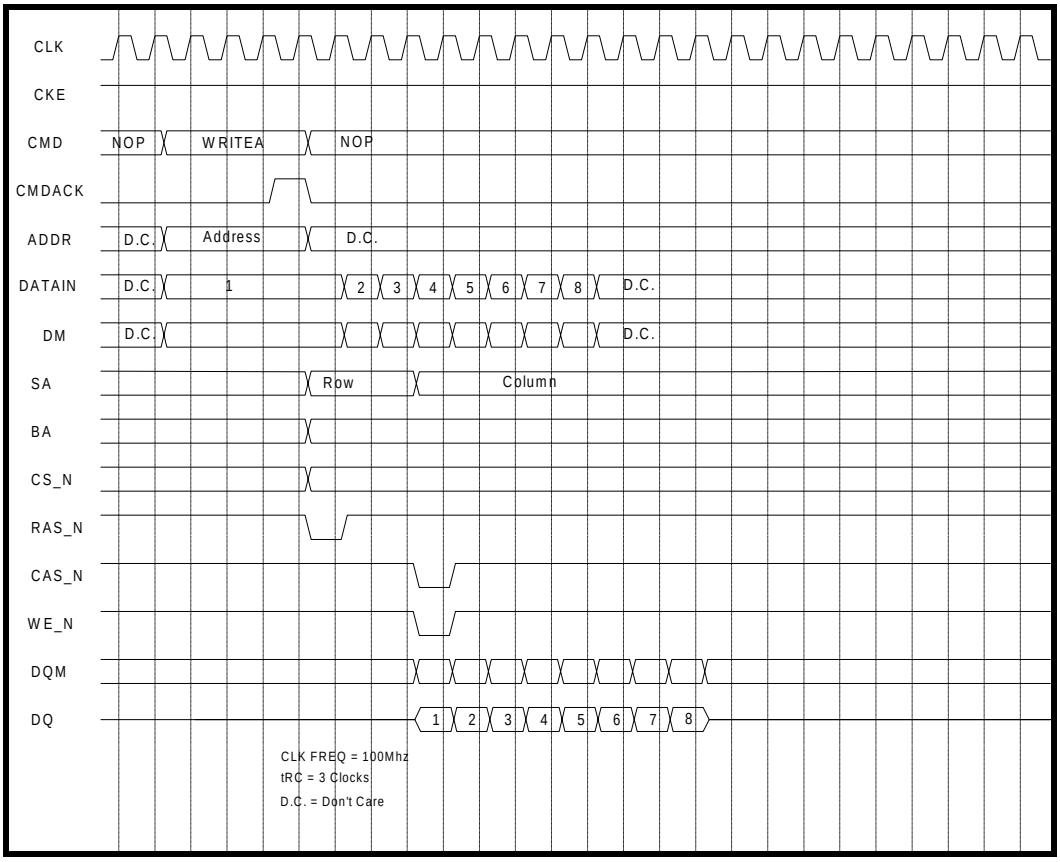


Figure 3 WRITEA Timing Diagram

REFRESH Command

The REFRESH command instructs the controller to perform an AUTO REFRESH command to the SDRAM. The controller will acknowledge the REFRESH command with CMDACK. A timing diagram of the REFRESH command is shown in Figure 4. The operation of a REFRESH command is as follows:

- The user asserts REFRESH on CMD input.
- The controller acknowledges the command by asserting CMDACK.
- Simultaneously with asserting CMDACK the controller starts issuing commands to the SDRAM devices.
- Immediately following the assertion of CMDACK by the controller, the user asserts NOP on CMD.

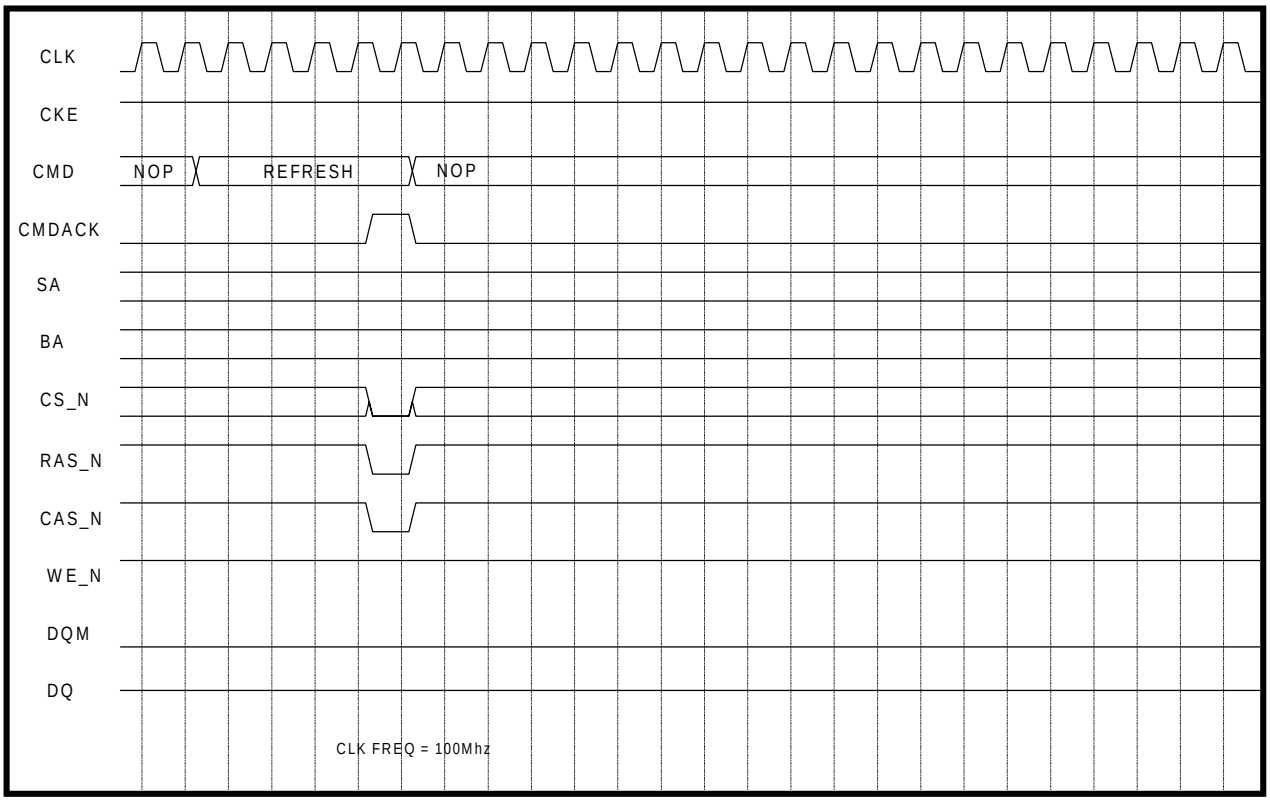


Figure 4 REFRESH Timing Diagram

PRECHARGE Command

The PRECHARGE command instructs the controller to perform a PRECHARGE command to the SDRAM. The controller will acknowledge the command with CMDACK. A timing diagram of the PRECHARGE command is shown in Figure 5. The operation of a PRECHARGE command is as follows:

- The user asserts PRECHARGE on CMD.
- The controller acknowledges the command by asserting CMDACK.
- Simultaneously with asserting CMDACK the controller starts issuing commands to the SDRAM devices.
- Immediately after the controller asserts CMDACK, the user asserts NOP on CMD.

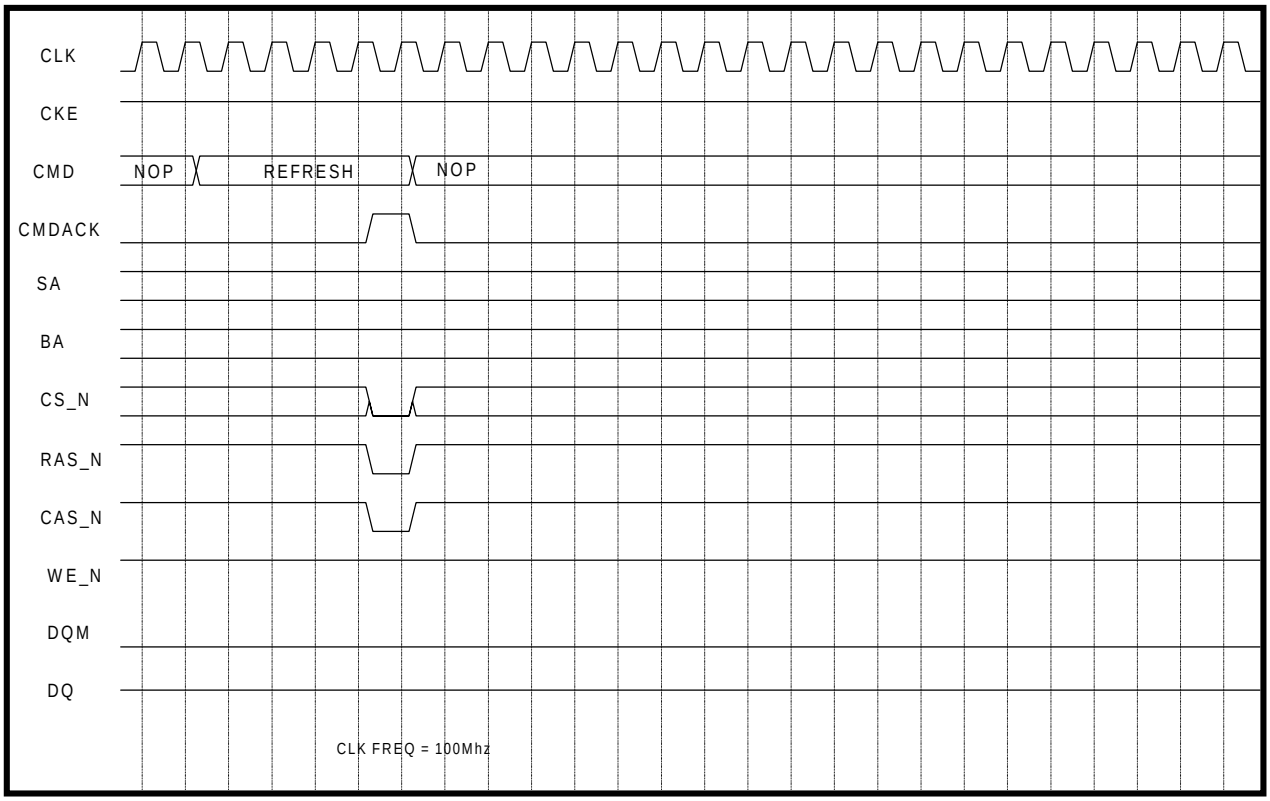


Figure 5 PRECHARGE Timing Diagram

LOAD_MODE Command

The LOAD_MODE command instructs the controller to perform a LOAD MODE REGISTER command to the SDRAM. The value that is to be written into the SDRAM mode register must be present on ADDR[11:0] with the LOAD_MODE command. The value on ADDR[11:0] is mapped directly to the SDRAM pins A11-A0 when the controller issued the LOAD MODE REGISTER to the SDRAM. A timing diagram is shown in Figure 6. DDR devices have an Extended Mode Register that is accessed with the LOAD_MODE command and setting BA[1:0]=01b. Setting BA to 01b is accomplished by setting the appropriate bits in ADDR during a LOAD_MODE command. The mapping of ADDR bits to BA bits is defined in the file PARAMS.V.

- The user asserts LOAD_MODE on CMD.
- The controller acknowledges the command by asserting CMDACK.
- Simultaneously with asserting CMDACK the controller starts issuing commands to the SDRAM devices.
- One clock after the controller asserts CMDACK, the user asserts NOP on CMD.

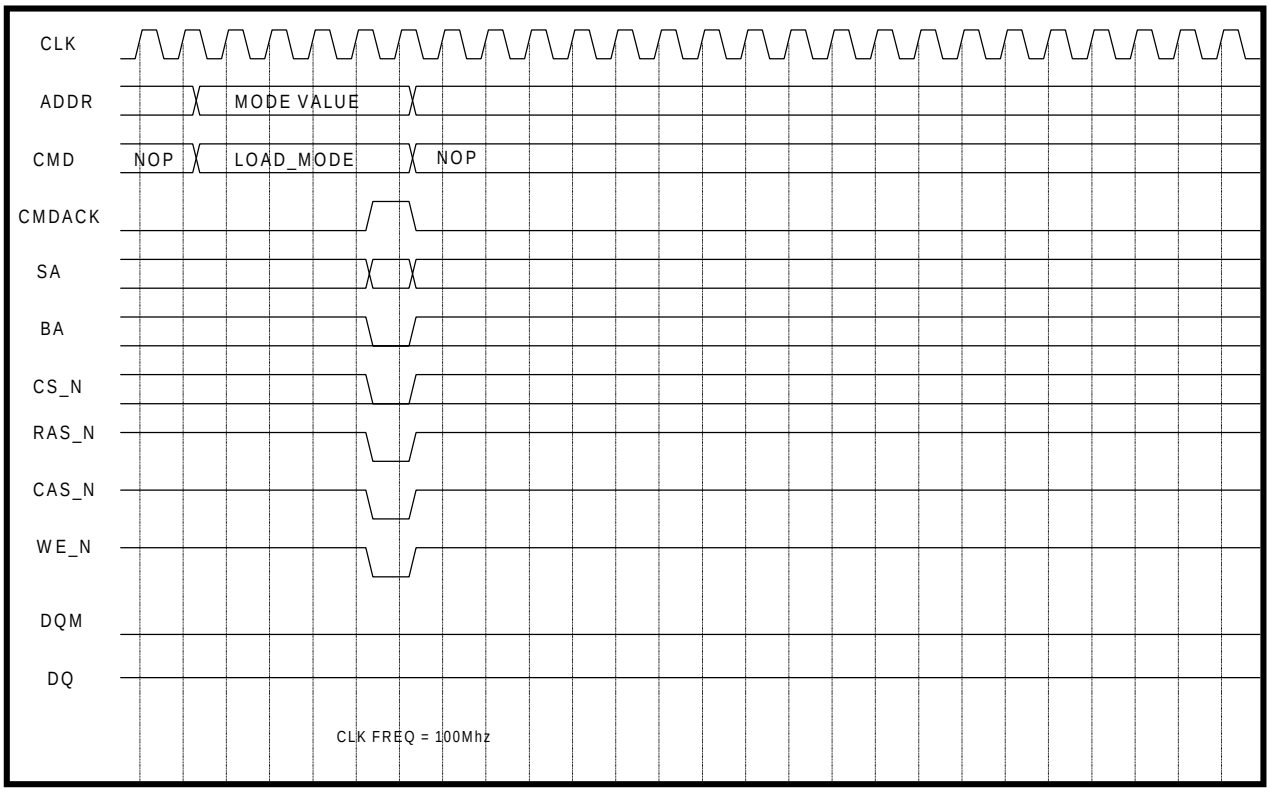


Figure 6 LOAD_MODE Timing Diagram

LOAD_REG1 Command

The LOAD_REG1 command instructs the controller to load the internal configuration register REG1. The value that is to be written into REG1 must be presented on the ADDR input simultaneously with the assertion of the command LOAD_REG1. The bit assignments for REG1 are shown in Table 4.

CL is the CAS latency of the SDRAM memory in clock periods and is dependent on the memory device speed grade and clock frequency. Consult the SDRAM data sheet for appropriate settings. CL must be set to the same value as CL for the SDRAM memory devices, see Table 4.

RC is the RAS to CAS delay in clock periods and is dependent on the SDRAM speed grade and clock frequency. The formula is RC= INT(tRC/ clock_period). Where tRC is the value from the SDRAM data sheet and clock_period is the clock period of the clock that the controller and SDRAM will be running off of.

RRD is the Refresh to RAS delay in clock periods. RRD is dependent on the SDRAM speed grade and clock frequency. The formula is RRD= INT(tRRD / clock_period). Where tRRD is the value from the SDRAM data sheet and clock_period is the clock period of the clock that the controller and SDRAM will be running off of.

BL is the burst length the SDRAM devices have been configured for divided by two to account for the multiplexed nature of the data bus for the DDR Controller.

Label	ADDR Bits	Description
CL	[1:0]	Cas Latency setting for the controller. 00 = 1.5 01=2.0 10=2.5 11=3.0
RC	[3:2]	RAS to CAS delay in number of clocks.
RRD	[7:4]	REFRESH command duration in clocks.
BL	[12:9]	Burst Length, valid values are 1,2,4

Table 4 REG1 Bit Definitions

LOAD_REG2 Command

The LOAD_REG2 command instructs the controller to load the internal configuration register REG2. REG2 is a 16-bit value that represents the period between REFRESH commands that the controller issues. The value is set by the equation int(REF_PERIOD / CLK_PERIOD). For example, if a SDRAM device connected to the controller has a 64ms, 4096 Cycle refresh requirement the device must have a REFRESH command issued to it at least every 64ms/4096 = 15.625 us. If the SDRAM and controller are clocked by a 100Mhz clock then the maximum value of REG2 should be 15.625us / .01us = 1562. The value that is to be written into REG2 must be presented on the ADDR input simultaneously with the assertion of the command LOAD_REG2.

SDRAM Device Initialization and Controller Configuration

The SDRAM devices that are connected to the SDRAM controller must be initialized before they can be accessed. This initialization process set the burst length, CAS latency, burst type, and operation mode for the SDRAM devices. After the SDRAM devices are initialized the SDRAM controller's configurations registers must be set. The procedure for initializing the SDRAM devices and the Controller is as follows:

- The user performs the PRECHARGE command (see section 0).
- Then a LOAD_MODE command (see section 0).
- Then a LOAD_REG2 command (see section 0).
- Then a LOAD_REG1 command (see section 0).

Architecture

The DDR SDRAM Controller Core consists of four main modules, the SDRAM controller, Control Interface, Command, and Data Path Modules. The DDR SDRAM controller module is the top-level module that instantiates the three lower modules and brings the whole design together. The Control Interface Module accepts commands and related memory addresses from the host, decoding the command and passing the request to the Command module. The Command Module accepts command and address from the Control Interface Module, generating the proper commands to the SDRAM. The Data Path Module handles the data path operations during WRITEA and READA commands. The top-level module also instantiates two PLL's that are used in the CLOCK_LOCK and 2X mode to improve I/O timing and to generate a 2X clock.

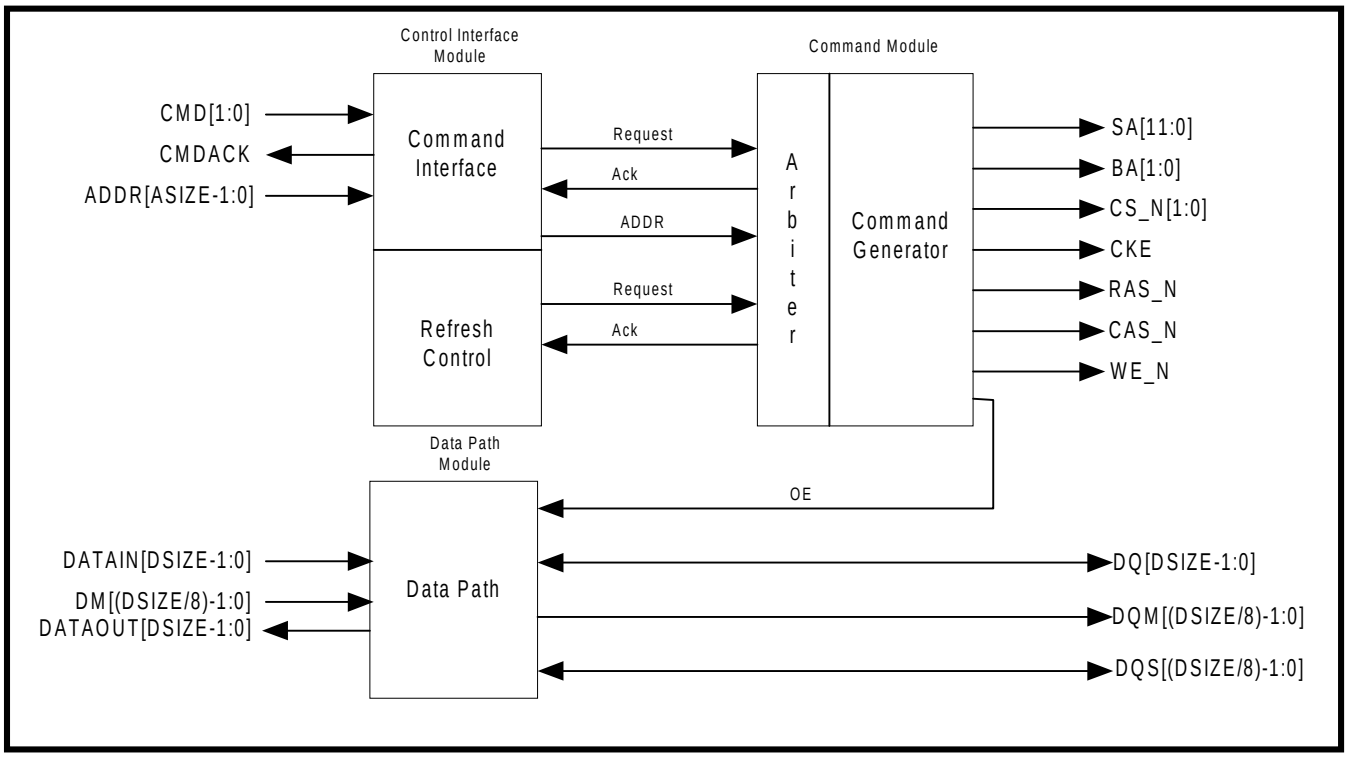
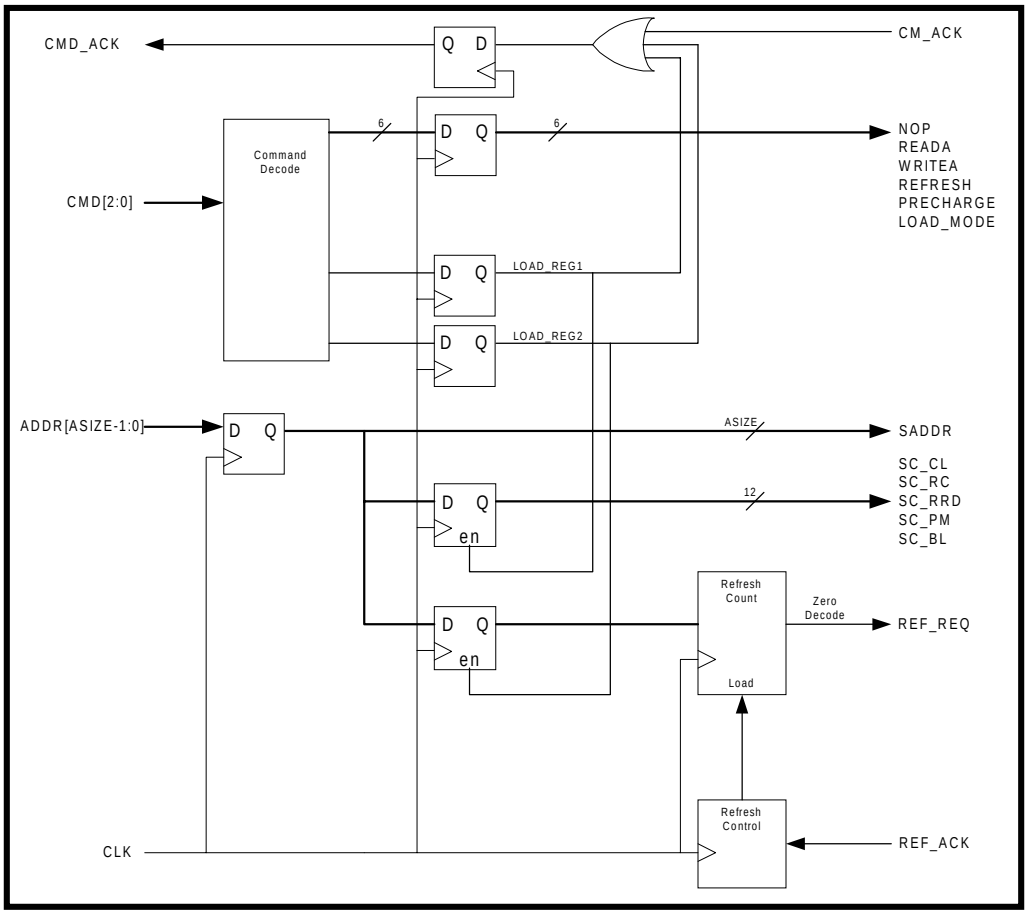


Figure 7 DDR SDRAM Controller Core Block Diagram

Control Interface Module

The control Interface Module decodes and registers commands from the host, passing the decoded NOP, WRITEA, READA, REFRESH, PRECHARGE, and LOAD_MODE commands, along with ADDR, to the Command Module. The LOAD_REG1 and LOAD_REG2 commands are decoded and used internally to load the REG1 and REG2 registers with values from ADDR. A block diagram of the Control Interface Module is shown in the below figure.

The Control Interface Module also contains a 16 bit down counter and control circuit that is used to generate periodic refresh commands to the Command Module. The 16-bit down counter is loaded with the value from REG2 and counts down to zero. The REFRESH_REQ output is asserted with the counter reaches zero and remains asserted until the Command Module acknowledges the request. The acknowledge from the Command Module causes the down counter to be reloaded with REG2 and the process repeats. REG2 is a 16-bit value that represents the period between REFRESH commands that the controller issues. The value is set by the equation int(REF_PERIOD / CLK_PERIOD). For example, if a SDRAM device connected to the controller has a 64ms, 4096 Cycle refresh requirement the device must have a REFRESH command issued to it at least every 64ms/4096 = 15.625 us. If the SDRAM and controller are clocked by a 100Mhz clock then the maximum value of REG2 should be 15.625us / .01us = 1562 Figure 6 shows the timing at the SDRAM interface for a REFRESH sequence.



DATA Path Module

The Data Path Module provides the SDRAM data interface to the host. Host data is accepted on port DATAIN for WRITEA commands and data is provided to the host on port DATAOUT during READA commands. Block diagrams of the Data Path Module are shown in Figure 14 and Figure 15. The data path width into the controller is two times the data path width to the DDR SDRAM devices. The data path module's DATAIN and DATAOUT ports are fixed at 32bits and the DQ port is fixed at 16bits. To build data paths larger than 32bits, Data Path Modules can be cascaded to increase the data bus width in increments of 32bits. The DATAIN write path takes the write data and transfers are over to the CLK200 (two times the frequency of CLK100), multiplexing the data to the DQ port. On the read side, DATAOUT path, the read data is captured using CLK200 to sample the data during the middle of the data valid window. The read data is then demultiplexed and transferred to the CLK100 clock domain, which is one half the frequency of CLK200. The Data Path Module also generates the DQS signal during write operations to the DDR devices.

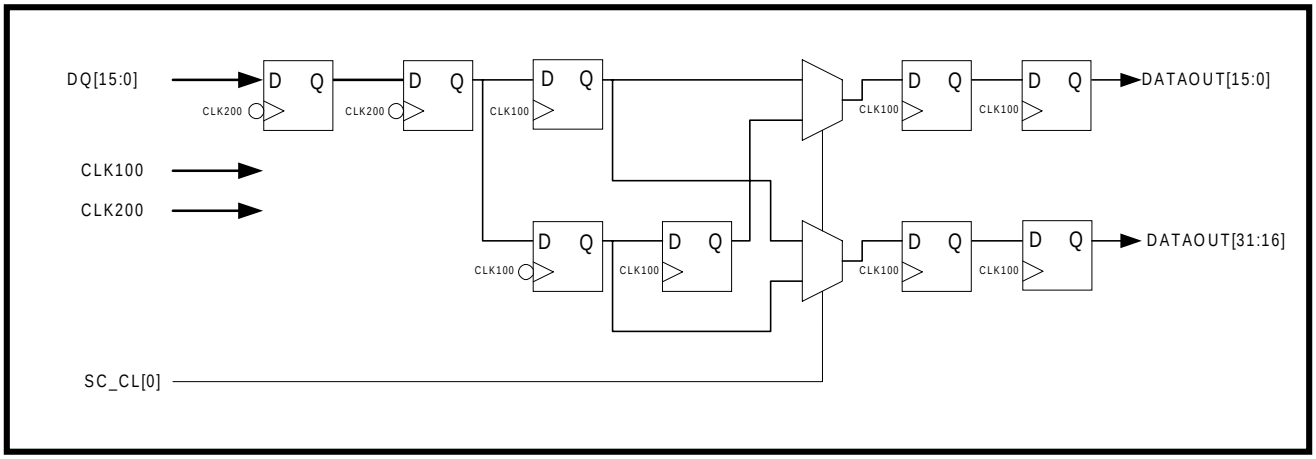


Figure 9 Data Path Module Block Diagram, Read Direction

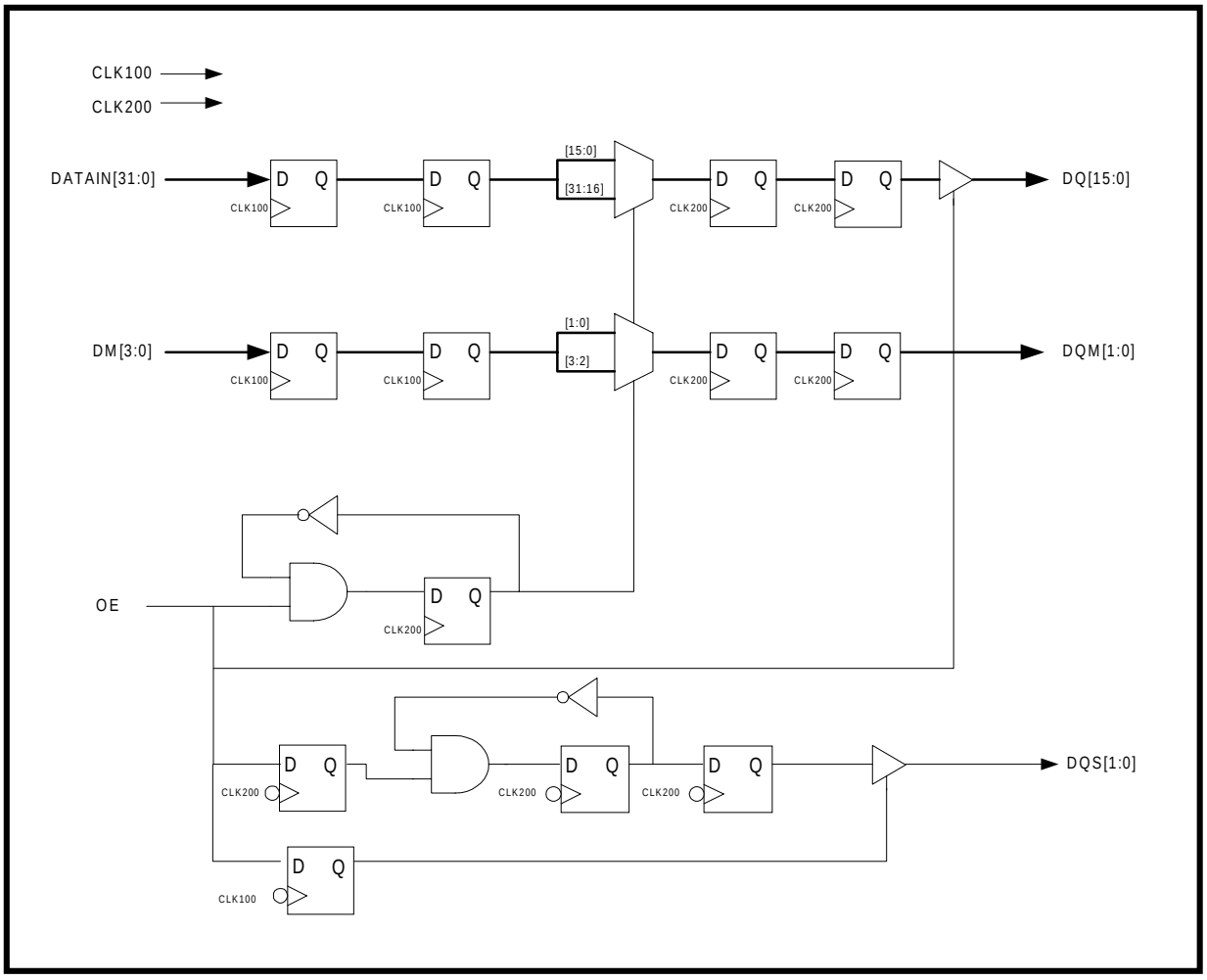


Figure 10 Data Path Module Block Diagram, Write Direction

Command Module

The Command Module accepts decoded commands from the Control Interface Module, along with refresh requests from the refresh control logic and generates the appropriate commands to the SDRAM. The module contains a simple arbiter that arbitrates between the commands from the host interface and the refresh requests from the refresh control logic. The refresh requests from the refresh control logic have priority over the commands from the host interface. If a command from the host arrives at the same time or during a "hidden" refresh operation, the arbiter holds off the host by not asserting CMDACK until the hidden refresh operation is complete. If a hidden refresh command is received while a host operation is in progress then the hidden refresh is held off until the host operation is complete. A block diagram of the Command Module is shown in Figure 18.

After the arbiter has accepted a command from the host, the command is passed onto the command generator portion of the Command Module. The command Module uses three shift registers to generate the appropriate timing between the commands that are issued to the SDRAM. One shift register is used to control the timing the ACTIVATE command while a second is used to control the positioning of the READA or WRITEA commands. A third is used to time command durations, in order for the arbiter to determine if the last requested operation is complete.

The Command Module also performs the multiplexing of the address (ADDR) to the SDRAM. The row portion of the address is multiplexed out to the SDRAM outputs A[11:0] during the ACTIVATE (RAS) command. The column portion is then multiplexed out to the SDRAM address outputs during a READA (CAS) or WRITEA command.

The output signal OE is generated by the Command Module to control tristate buffers in the last stage of the DATAIN path in the Data Path Module.

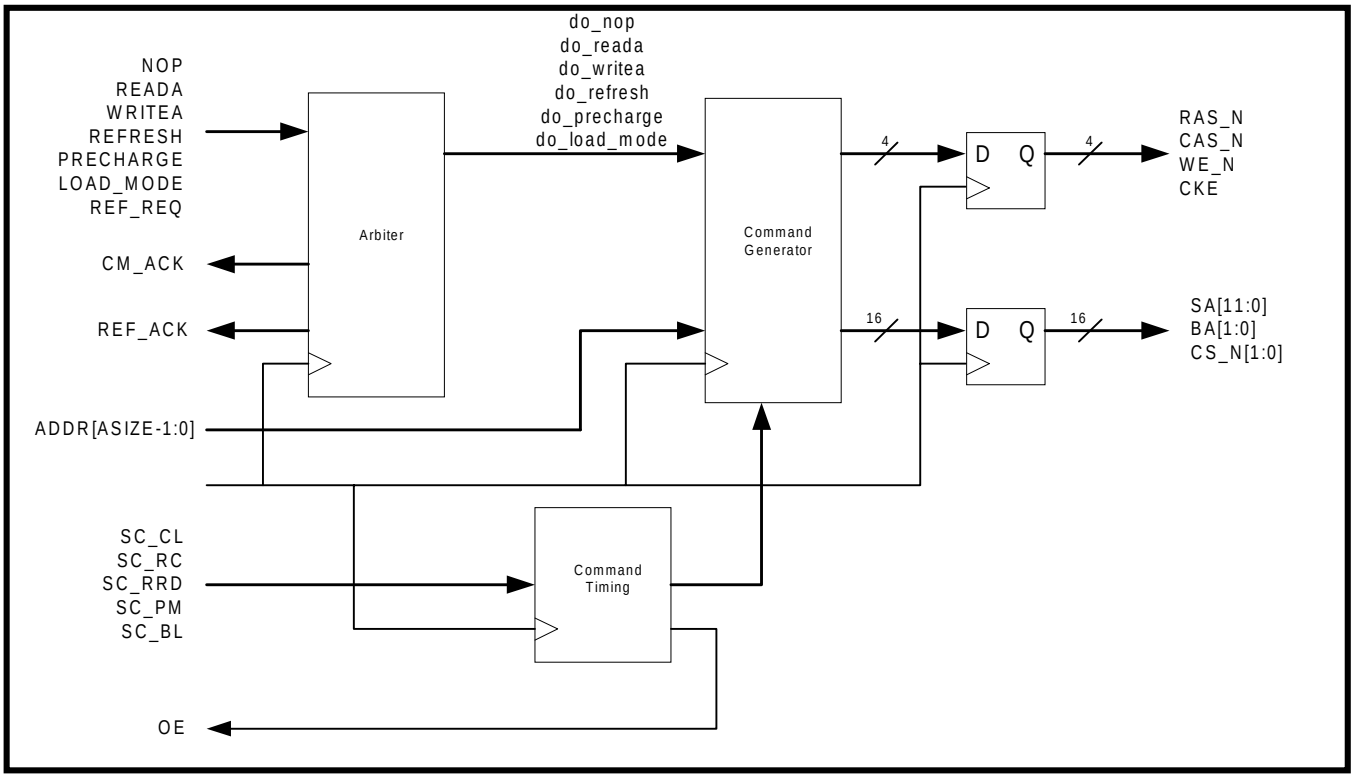


Figure 11 Command Module Block Diagram

Simulation, Synthesis, Place and Route, and Results

The DDR SDRAM Controller includes source files, test bench, synthesis scripts, and support files for place and route using Quartus 2000.02 targeting an EP20K400EFC672-1X. The directory structure for this release is shown in Table 6. The various directories and files are discussed below.

Directory	Description
\source	Contains the source files for the DDR SDRAM controller.
\simulation	Contains the DDR SDRAM controller test bench.
\model	SDRAM memory model.
\synthesis\synplicity	Contains synplicity project file and synthesis results.
\synthesis\fpgaexpress	Contains FPGA express project file and synthesis results.
\route	Contains all files necessary to route the SDR SDRAM controller using Quartus 00.02
\doc	Documentation

Table 5 DDR SDRAM Controller release directory structure

Simulation

The DDR SDRAM Controller can be simulated using the behavioral source file ddr_sdram_tb found in the simulation directory. The test bench instantiates: the DDR SDRAM Controller and two SDRAM memory models. The test bench then configures the SDRAM devices and controller and runs a ramp pattern test using all combinations of CAS latency, burst length, and RAS-CAS delays for a given clock frequency.

Synthesis

The \synthesis\synplicity and \synthesis\fpgaexpress directories contain project files for synplicity and FPGA Express, respectively. These directories also contain the result files from synthesis runs with the DDR SDRAM controller. The only synthesis constraint is the setting of the global fmax to 100/200Mhz. The project file also sets the target device for synthesis to an EP20K400EFC672-1X.

The source file params.v contains several `defines that set constants in the DDR SDRAM Controller design, see Table 8.

`define	Description
DSIZE	Defines the size of the DATAIN and DATAOUT ports. DQ is sized to be ½ DSIZE. Possible values are 32,64,96,128
ASIZE	Defines the size of the ADDR port.
ROWSTART	Defines the starting position within ADDR of the group of bits that are to be used for the row address to the SDRAM devices.
ROWSIZE	Defines the number of bits to use, starting at ROWSTART within ADDR, for the SDRAM row address.
COLSTART	Defines the starting position within ADDR of the group of bits that are to be used for the column address to the SDRAM devices.
COLSIZE	Defines the number of bits to use, starting at COLSTART within ADDR, for the SDRAM column address.
BANKSTART	Defines the starting position within ADDR of the group of bits that are to be used for the bank address to the SDRAM devices.
BANKSIZE	Defines the number of bits to use, starting at COLSTART within ADDR, for the SDRAM bank address.

Table 6 DDR SDRAM Controller `Defines

Place and Route

The place and route directory contains the files from a Quartus 2000.02 build of the DDR SDRAM Controller. The global FMAX constraints are inherited from the PLL's that are instantiated by the controller and are set to 100Mhz and 200Mhz.

In this design for an EP20K400EFC672-1X device, the DQ, DQS, and DQM lines were hand placed in order to meet the timing requirements for 100/200 MHz DDR devices. For read burst operations the DDR controller uses the falling edge of the 200mhz clock to capture the read data coming from the DDR devices. Through the use of an internal PLL, the DDR devices attempt to achieve Tco = 0.0 but the data sheet specs Tco (actually Tac in the Micron DDR SDRAM data sheets) as +-.75ns. This reduces the 5ns(DDR at 100mzh) data valid window to 3.5ns. The I/O cell registers are used in the input path to achieve route independent setup times on the DQ pins of 1.2ns. This leaves 2.3ns left in the timing budget for board related issues. The I/O timing analysis for write operations is a bit more difficult because the I/O cell registers are used in the input path, resulting in Tco on the DQ pins that can vary from route to route of the design. During a burst write operation to DDR devices, DQS is used as a clock by the SDRAM devices to register data and data mask values on DQ and DQM. Therefore, the timing analysis consists of verifying the Tsu and Th of the DQ outputs relative to the DQS outputs. Each 8-DQ lines have a DQS associated with them and the timing requirements separately to each of these groupings. The formula for calculating the worst case Tsu and Th for a DQ,DQS grouping is

$$Tsu = DQS\_Tco + 2.5ns - DQ\_Tco(max)$$
$$Th = DQ\_Tco(min) - DQS\_Tco + 2.5ns$$

The DDR SDRAM controller has 8 byte groupings and the relative timing results from the post route are shown in Table 11. Refer to the Altera Application Note 75, High-Speed Board Designs, for more

information on PCB board design for High-Speed applications.

The other SDRAM interface control signals (RAS, CAS, etc) all use the register in the I/O cell as an output register. This results in route independent Tco of xxx ns. With a main clock period of 10ns(100mhz) this leaves xxx ns in the timing budget for board related timing delays.

DQ Group	DQ Tco Max/Min	DQS Tco	DQ Tsu Relative to DQS	DQ Th Relative to DQS
[7:0]	4.4/5.4ns	5.2ns	2.3ns	2.7ns
[15:8]	4.5/5.4ns	5.2ns	2.3ns	2.7ns
[23:16]	4.5/5.3ns	4.7ns	1.9ns	3.1ns
[31:24]	4.0/5.8ns	5.1ns	1.8ns	3.2ns
[39:32]	4.4/5.3ns	5.3ns	2.5ns	2.5ns
[47:40]	4.2/5.5ns	5.2ns	2.2ns	2.8ns
[55:48]	4.6/5.2ns	4.8ns	2.1ns	2.9ns
[63:56]	4.4/5.4ns	5.1ns	2.2ns	2.8ns

Table 7 DQ and DQS post route timing comparison

Results

In a 20K400E-1X device, the DDR SDRAM controller utilizes 864 Logic Elements and runs at 100Mhz internal and 200Mhz pin.

The post route performance results for the DDR SDRAM controller are shown in Table 6. Throughput is calculated using a 128-bit user data path and 18 clock cycles time for a 4 data burst. READA cycle time in clocks represents the critical path as WRITEA cycle time is 12 clocks. The formula for calculating the throughput in Mbytes/sec is

1 / (clk period ns * number of clocks per access) * (data path width in Bytes) * (burst length of the access)

Device	Internal Fmax	Tco/Tsu	Throughput	Total LEs
20K400E-1X	100/200Mhz	4.8ns/1.2ns	355 Mbytes/sec	864

Table 8 Post Route Performance