DDR2 SDRAM CONTROLLER

Authors:-

Avinash Prabhu
ECE
IIIT Hyderabad
Hyderabad,India
avinash.prabhu@students.iiit.ac.in

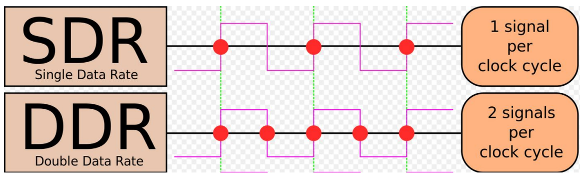
Rahul Kashyap
ECE
IIIT Hyderabad
Hyderabad,India
rahul.kashyap@students.iiit.ac.in

Karan Mirakhor
ECE
IIIT Hyderabad
Hyderabad,India
karan.mirakhor@students.iiit.ac.in

Tushar Patra
ECE
IIIT Hyderabad
Hyderabad,India
tushar.patra@students.iiit.ac.in

Abstract**—In this project we have attempted to design a DDR2 SDRAM Controller. Such a controller has significant benefits compared to the traditional SDR controller.**

I.Introduction



Double Data Rate 2 Synchronous Dynamic Random Access Memory allows higher bus speed as a DDR SDRAM with a certain clock frequency achieves nearly twice the bandwidth of a SDR DRAM; these factors combine to produce a total of four data transfers per internal clock cycle; in addition to transferring the data on the rising and falling edges of the bus clock signal as in DDR SDRAM . DDR2 provides better latency compared to DDR .

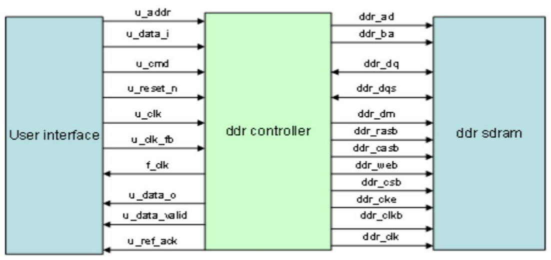


Figure 1: Top Level block diagram

The **memory controller** is a digital circuit that manages the flow of data going to and from the computer's main memory. A memory controller can be a separate chip or integrated into another chip, such as being placed on the same die or as an integral part of a microprocessor.

II.Design

A.Control Interface Unit

A control interface module accepts the commands from the processor and decodes the command and provides the request to the command module. It is basically an 8 state FSM with each state representing the command like NOP ,REFRESH, READ, WRITE, etc .

NOP command: The No Operation (NOP) is a command used to instruct the controller to perform a NOP in the following clock cycle. Operations that are already in progress are not affected by the NOP command.

READA command: The READA command is used to instruct the controller to perform burst read operation. The controller will issue an ACTIVATE (ACT) command on the SDRAM to activate a specific row address followed by the READA command. The burst data will appear on DATAOUT.

WRITEA command: The WRITEA command is an instruction to controller to perform a burst write operation. In this the ACTIVATE(ACT) command is given before the write operation to select the required row and then the burst write operation is performed using the WRITEA command.

REFRESH command: SDRAM is a volatile memory and to retain its data it needs to get refreshed periodically. The REFRESH command is used to instruct the controller to perform an Auto REFRESH operation.The auto refresh command is periodically issued to avoid data retention. Even this command is preceded by precharge.

PRECHARGE command: The PRECHARGE command is used to instruct the controller to perform a deactivation operation on one bank or on all banks. This is issued before a READ or WRITE operation or REFRESH command. Basically, precharge operation closes the row currently in operation so that the bank can again go back to its idle state and perform the

READ or WRITE operation on the required row or column.

LOAD_MODE Register command: The Load mode register command is used to define the specific mode of the SDRAM mode register. The register stores the burst type, CAS latency, burst length and burst write mode. There are two control registers REG1 and REG2. REG2 is used to provide an auto refresh operation.

The controller will acknowledge the commands with CMDACK.

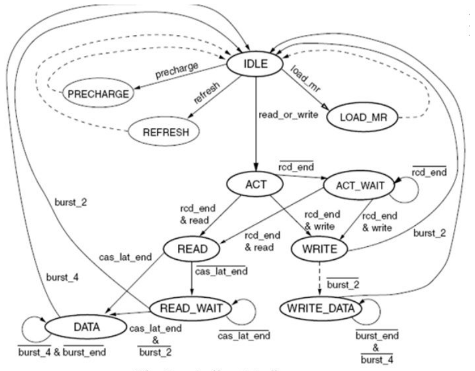


Fig. 6 controller state diagram

B.Command Module

The basic function of the command module is to get its control signal from the control interface module and generate the addresses for the SDRAM.
 Command module consists of an arbiter ,multiplexer and three shift registers :

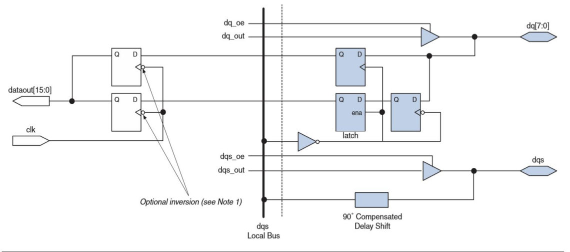
1)Arbiter : An arbiter is used in case if multiple requests are arriving from different- different devices. If a high priority and low priority device both need to transmit data then high priority device is given priority and low priority device holds data unless high priority device has sent all its data.
Eg : High priority is given to REFRESH command and the other command are put in hold state with the help of not asserting the CMDACK .

2) MULTIPLEXER: These are used to select the specific Address line for the given operation. The ROW Address is multiplexed to SDRAM during the ACTIVATE (RAS) command and column portion is multiplexed to SDRAM during the READ or WRITE command.

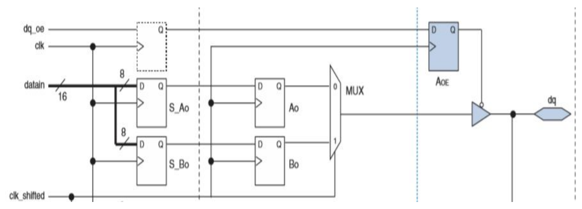
3) SHIFT REGISTERS : These are used to maintain the timing in the SDRAM commands like the latency etc.

C.Data Path Module

Read Operation



Write Operation



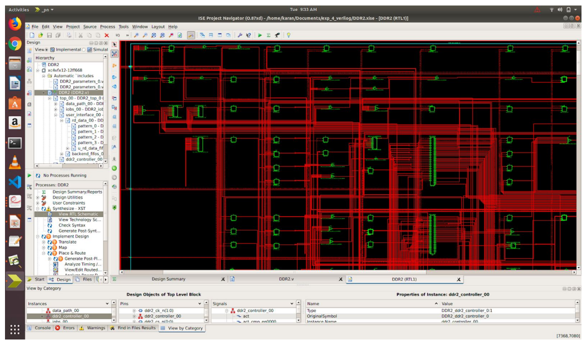
III.Comparison with other implementations

Device	**Power(W)**	**Timing Delay(ns)**
Spartan 3 (XC3S1400AN)	0.064	7.054
Virtex 4 (XC4VFX12 -12ff668)	2.997	6.949

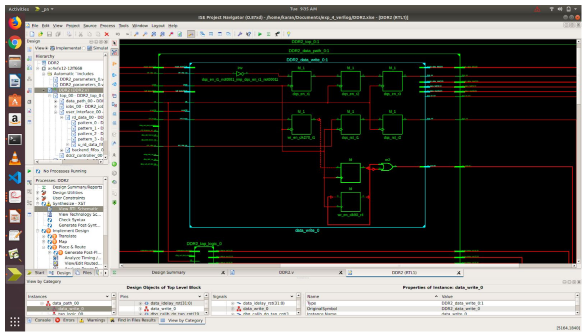
IV.Results

The following are the RTL schematics, timing diagram, power and timing delays of the DDR2 SDRAM controller :-

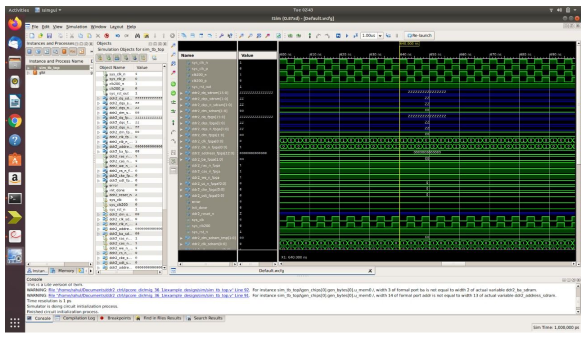
1) The overall RTL Schematic of the DDR2 SDRAM controller

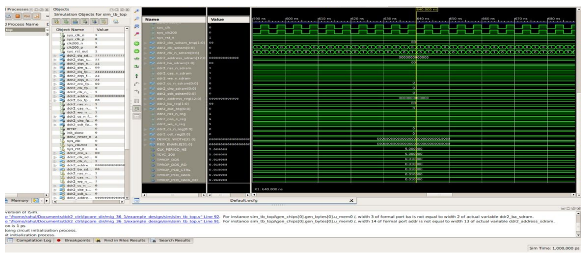


2) The RTL schematic of the DATA PATH MODULE (WRITE)

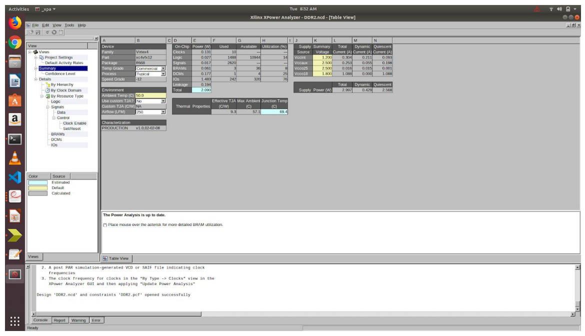


3) Timing Diagrams of the various commands executed by the DDR2 SDRAM controller





4) The Power analysis report of the design as generated by Xilinx ISE 13.7 on Virtex-4 XC4VFX12 -12 ff668 is



5) Timing report of the design as generated by Xilinx ISE 13.7 on Virtex-4 XC4VFX12 -12 ff668 is

```
Clock Information:
------------------
-----------------------------------+------------------------------------+-------+
Clock Signal                       | Clock buffer(FF name)              | Load  |
-----------------------------------+------------------------------------+-------+
sys_clk                            | infrastructure0/DCM_BASE0:CLK0     | 1449  |
sys_clk                            | infrastructure0/DCM_BASE0:CLK90|   113   |
idly_clk_200                       | IBUFG+BUFG                         | 25    |
-----------------------------------+------------------------------------+-------+
```

```
Timing Summary:
---------------
Speed Grade: -12

   Minimum period: 3.458ns (Maximum Frequency: 289.172MHz)
   Minimum input arrival time before clock: 2.228ns
   Maximum output required time after clock: 4.721ns
   Maximum combinational path delay: No path found
```

V. Conclusion

DDR2 SDRAM controller has been implemented using Verilog HDL. High speed, pipeline and burst access features makes it the most popular form of memory used in system design. Use of DDR 2 SDRAM has two advantages that it reduces the system cost and increases the data transfer throughput. The synchronous interface provides the operation to be pipelined and arranged in queue as they requests. This DDR2 SDRAM controller is used in embedded system in which high speed is of great concern.The highest-rated DDR2 modules operate at 533 MHz (1066 MT/s), compared to the highest-rated DDR modules operating at 200 MHz (400 MT/s).

VI. Acknowledgement

The notable contributions of Prof. Azeemuddin Syed and the TA's namely Mayank Awasthi and Kunal wadhwani was very valuable. They helped us complete the project fully and on time with full understanding of the basic concepts to complex implementation of the project.

VI. References

1. J.Bhaskar " A verilog HDL primer – 2nd Edition."
2. John Wakerly "Digital systems design – 8th Edition."
3. Jedec solid state technology association -Double data rate SDRAM specification
4. Neil H.weste "Principles of CMOS VLSI design- 2nd edition."
5. RTL encounter user guide and Cadence user guide.
6. Samir palnitkar " Verilog HDL A guide to digital design and synthesis – 2nd Edition."
7. Xilinx Inc., XAPP179, using select IO interfaces, application note
8. LatticeSC/M DDR/DDR2 SDRAM Memory interface User'sGuide, TechnicalNoteTN1099, July2008
9. J.Bhaskar "A verilog HDL primer- 2nd edition"
10. John Wakerly "Digital system design – 8th edition."
11. Burchardt, A., Hekstra-Nowacka, E., and Chauhan, A., "A Real-time Streaming Memory Controller", Design,
12. Automation and Test in Europe 2005 Proceedings, 2005, vol.3, pp. 20-25
13. Heithecker, S and Ernst, R., "Traffic Shaping for An FPGA Based SDRAM Controller with Complex QoS Requirements", Design Automation Conference 2005, June 2005, pp. 575-578.