Design and Implementation of High Speed DDR SDRAM Controller on FPGA

Veena H K
M.Tech Student, Department of ECE,
Sambhram Institute of Technology, Bangalore,
Karnataka, India

Dr. A H Masthan Ali
Associate Professor, Department of ECE,
Sambhram Institute of Technology, Bangalore,
Karnataka, India

Abstract — **The dedicated memory controller is important is the applications in high end applications where it doesn't contains microprocessors. Command signals for memory refresh, read and write operation and SDRAM initialisation has been provided by memory controller. Our work will focus on FPGA implementation of Double Data Rate (DDR) SDRAM controller. The DDR SDRAM controller is located in between the DDR SDRAM and bus master. The operations of DDR SDRAM controller is to simplify the SDRAM command interface to the standard system read/ write interface and also optimization of the access time of read/write cycle. The proposed design will offers effective power utilization, reduce the gate count, reduce the area of chip and improves the speed of system by reducing the gates. The proposed design is implemented using Xilinx FPGA platform.**

Keywords — Synchronous Dynamic Random Access Memory (SDRAM), Field Programmable Gate Array (FPGA), Double Data Rate (DDR), VHDL/Verilog, Code optimization.

I. INTRODUCTION

Micro chip and in particular embedded real-time systems typically consist of several computational elements. These fundamentals accomplish different tasks for all functions and overall solution. These computational or processing elements are either general-purpose processors or digital signal processors. Currently, several processors are integrated into a Micro-chip solution. There is a interaction with other systems in order to complete a task the interaction systems may include memories and i/o. Currently buses are used to connect between different IP blocks. The research is made between network on chip and buses to connect between different IP blocks and it found that NOCs have more flexibility. In applications where high volume storage is required is usually put off-chip as dynamic memory. The dynamic memories provide high data rates and high storage capabilities. In the manufacturing of dynamic memory the manufacturing process varies differently from one another from standard logic therefore separation between memories is required.

SDRAM is a CMOS high-speed, dynamic random access memory (DRAM) and it is configured as a quad-bank Dynamic RAM internally with a synchronous interface means that all signals are registered on the falling or rising edge of the clock signal CLK. Read/write access to the SDRAM are burst type; accesses start at a selected location (selected by program) and continue for a definite number of locations as in a programmed sequence. Accessing of the data from a memory location begins with an ACTIVE command from the interpreter and it is then followed by a READ/WRITE command. In general memory is represented in banks, rows and column. To point to a location in the memory these three are mandatory. The ACTIVE command signal along with registered address bits point to the specific bank and row to be accessed. And column is given by READ/WRITE signal along with the registered address bits that shall point to a location for burst access. The DDR SDRAM interface makes higher transfer rates possible compared to single data rate (SDR) SDRAM, by more firm control of the timing of the data and clock signals. Implementations frequently have to use schemes such as phase-locked loops (PLL) and self-calibration to reach the required timing precision [1][2]. The interface uses double pumping means transferring data on both the rising as well as falling edges of the clock signal to lower the clock frequency. One of the major uses of keeping the clock frequency down is that it reduces the signal integrity requirements connecting the memory to the controller on the circuit board. The "double data rate" means with a certain clock frequency a DDR SDRAM achieves virtually double the bandwidth of a SDR SDRAM working at the same clock frequency due to double pumping at the interface of memory controller.

To achieve high-speed operation the SDRAM uses an internal pipelined architecture. This pipelined architecture is well-suited with the 2*n* rule of prefetch architectures, but it also makes space for the column address to be changed on every clock cycle to achieve a high-speed and fully random access. Precharging of a bank while accessing any of the other three banks h A read/write access for the DDR SDRAM in fact consists of a single 2n-bit wide, one clock-cycle data transfer at the internal DRAM core and two corresponding n-bit wide, one-half clock cycle data transfers at the I/O pins. Because of the internal bus width is two times the external bus width; DDR SDRAM achieves a data output rate that is almost twice the data rate of the internal bus. Double data rate (DDR) SDRAM—SDRAM, latches command information on the positive or rising edge of the clock; data is driven or latched on both the positive and negative edges of the clock rather than just the rising edge. This method doubles the data throughput rate without an increase in frequency. DDR SDRAM utilizes a differential clock (CLK, CLK') and data strobe signal DQS for high-speed burst mode data transfer. DQS is synchronized with clock signal, and data input and output (DQ) is synchronized with both the positive and negative of the CLK. A bidirectional data strobe (DQS) is transmitted along with data. DQS is used to capture data at the receiver [5]. DQS is a strobe signal transmitted by the memory controller during WRITE operation and by the DDR SDRAM during READ mode. The read/write accesses to the DDR-SDRAM are burst mechanism based, means more than one location data are read at a time and usually accessing starts at a specified location and continue to fetch data for required number of locations.

II. METHODOLOGY

A. DDR SDRAM Controller Architecture

Bus master is responsible for sending addresses and control signals to DDR SDRAM Controller with respect to read and write operations. Following Figure 1 depicts the proposed architectural blocks of the DDR SDRAM controller.

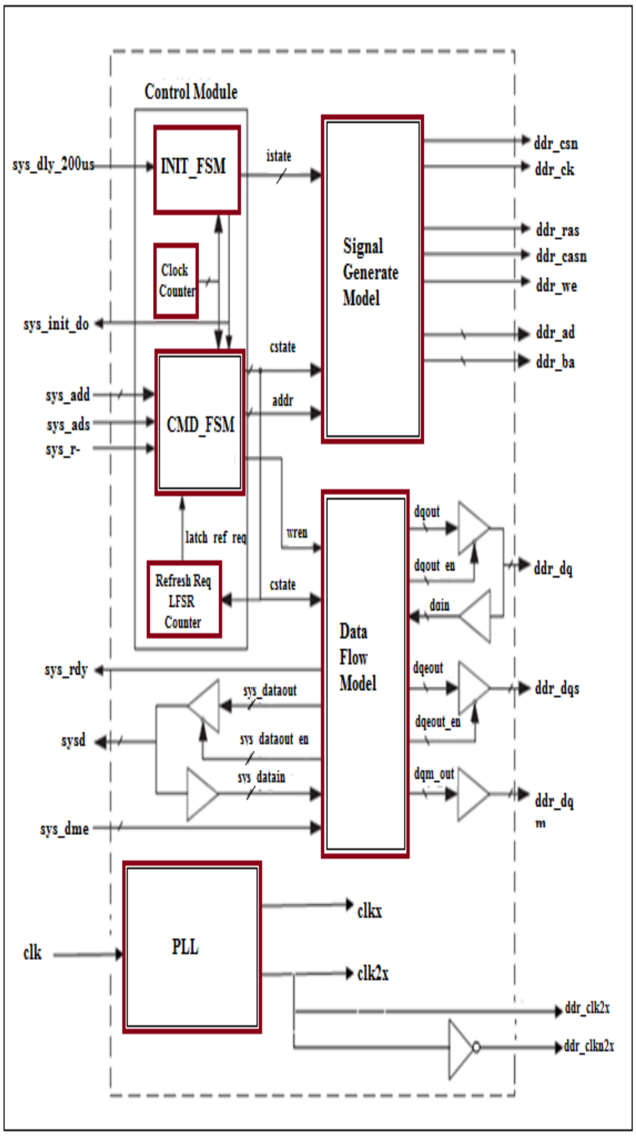

Figure 1: Functional block diagram of proposed DDR SDRAM controller

The modules of DDRAM are

- Main control
- Signal generation
- Data path
- PLL (Phase Locked Loop) modules

The main control module has refresh counter and two state machines. These perform initialization of the SDRAM and issue the commands to the SDRAM. The cState and iState outputs are produced as per the system interface control signals. The Signal generation module is responsible for generating address and command signals based on the iState and cState. The data path module performs the write and read operations between the DDR SDRAM and bus master and the PLL module generates the clock to the all sub modules.

The DDR SDRAM uses DDR architecture to achieve high-speed operation. The DDRAM is based on 2n pre-fetch architecture that can achieve two data words per clock pulse at the I/O pins for a single read or write access.

B. DDR SDRAM Main Controller Block

Before it is operational the DDR SDRAM memory controller shall be initialized with set of parameters. The process involves issuing sequence of command signals before it gives access to memory. The main control module state machine is responsible for the initialization of the memory controller.

Three sub-modules in main control module are

1. INIT_FSM module- Initialization FSM
2. CMD_FSM module- Command FSM
3. Counter module.

C. Design and Operation of Initialization FSM

Design of Initialization FSM Module Using Mealy FSM:

Initialization module design is carried out using Mealy state machine. In this state machine all data transfer occurs around common rising (or falling) clock edge. The outputs of the circuit are sampled only on active clock edge.

As output remains stable before and after active edge clock both setup and hold times are satisfied. Mealy is chosen to design Initialization module because output is valid on occurrence of active clock edge and also output occurs one clock period before the output in equivalent Moore machine. Number of states in Mealy is less compared to Moore machine.

Operation of Initialization FSM module:

Following Figure 2 depicts FSM initialization state machine. In initialization state machine, when the reset signal go low, transits to i_IDLE state irrespective of its current state. When system is in idle state it remains idle without performing any operations. The controller has to wait for clock to stabilize for around 200us, upon reset signal going high. The sys_dly_200us signal monitors this delay. Active high on this signal indicate that clock is stabilized.

Upon completion of the clock stabilization the initialization sequence of DDR controller will begin. The INIT_FSM will change its state from i_IDLE to i_NOP state. The NO OPERATION (NOP) command is used to avoid unnecessary commands from being registered during idle or wait states. It does not affect the operations already in progress.Next clock cycle INIT_FSM transits from the i_NOP state to the i_PRE state.

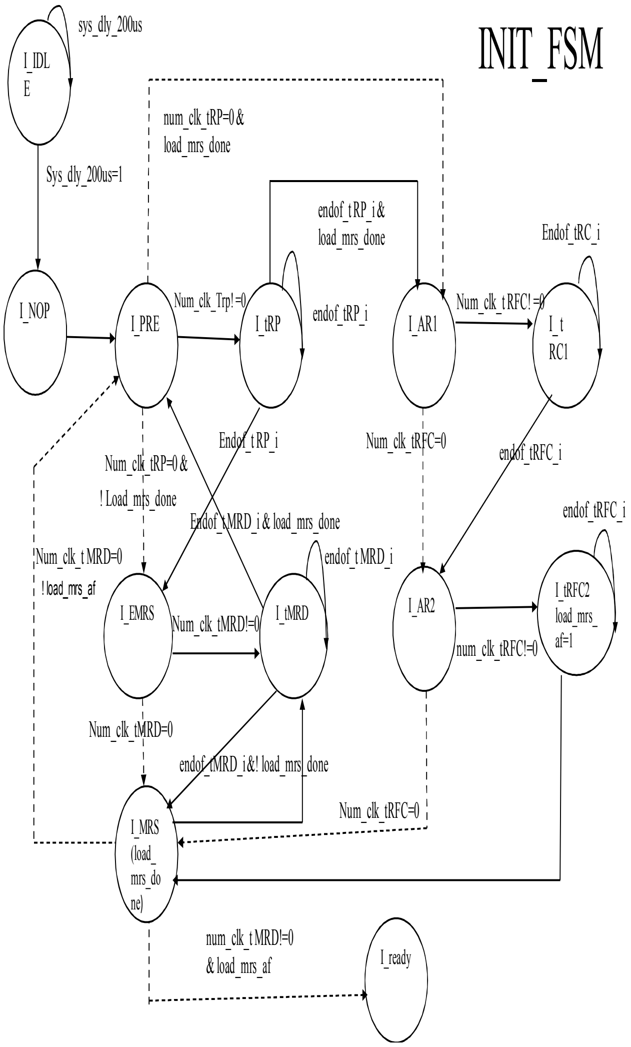

Figure 2: DDR SDRAM Initialization FSM (INIT_FSM) state diagram

While in i_PRE state, PRECHARGE command is generated by main control module. It is applied to all the banks in the device. The command can deactivate the open row corresponding to a particular bank or open the row in all banks depending on the BA0 and BA1 states, and the state of A10 triggers a single bank or all banks to be pre-charged. Upon generation of PRECHARGE command the INIT_FSM transits to next stage i.e. to two AUTO REFRESH commands.

The generation of AUTO REFRESH commands is to refresh the DDRAM memory. This command is issued each time a refresh is needed. In order to issue and perform auto refresh the condition is that all banks shall be in idle state. The refresh is performed for two times, upon finishing INIT_FSM transits to i_MRS state.

In i_MRS state configuration process initiated by issuing LOAD MODE REGISTER command to configure the DDRAM controller for operation. Similar to AUTO REFRESH this command shall also be applied when all banks are in idle state and shall wait until tMRD is met for a subsequent executable command to be issued. Upon completion of i_tMRD delay the INIT_FSM transits to i_ready state. From here on the INIT_FSM remain in this state for memory access operations. During this time signal sys_INIT_DONE is set to high to notifying that initialization process is done. Following delay state are used for issuing DDR commands

i_PRE

i_AR1

i_AR2

i_EMRS and *i_MRS* delay *states*.

Signal Generation Module:

The Figure 3 shows the signal generation module with inputs and outputs as shown in the figure. This module is responsible for generating command signals to the DDR SDRAM. Following are the command signals

- ddr_add: for generating addresses
- ddr_casn and ddr_rasn: for selecting particular column and row address.

These signal are driven by value of the iState and cState received from the CMD_FSM and INIT_FSM part of the Main Control module

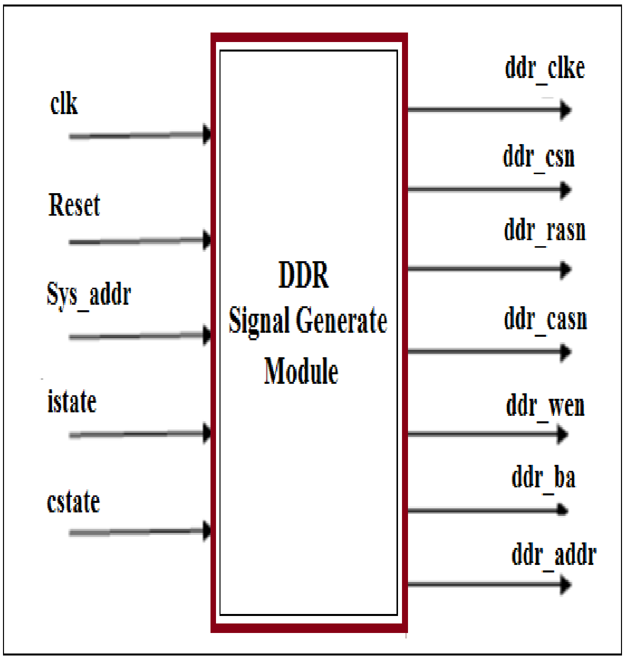

Figure 3: Signal Generation Module

Data Path Module:

The Figure 4 shows the data path module with inputs and outputs as shown in the figure. The data flow design between the SDRAM and the Bus master. The DDR SDRAM Controller design interfaces between the 8-bit data bus and the bus master .with a 16-bit data bus. Bidirectional data bus and the bus master with 16-bit bidirectional data bus. The user should be able to modify this module to customize to fit his or her system bus requirements. The data path module performs the data latching and dispatching of the data between the processor and DDR. The data path module for read and write are shown in figure. The data path module depends on cState for its read/write operation. The cState is generated by the CMD_FSM present in the Main Control module.

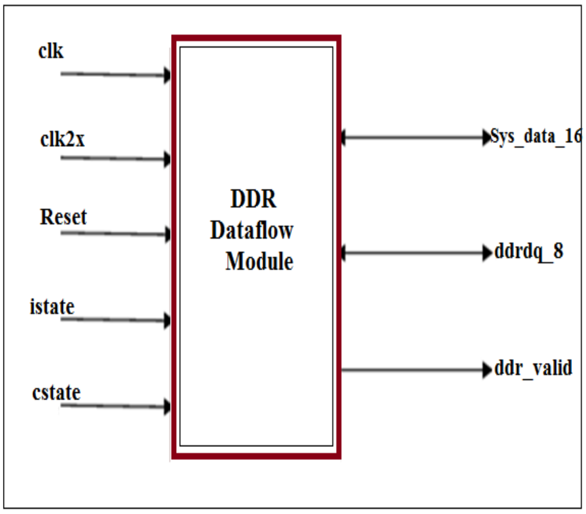

Figure 4: Data Path Module

PLL Module:

PLL is required for generation clk (100MHz) and clk2x (200MHz). And also PLL is required for generating ddr_clk and ddr_clkn. The reason for PLL clk2x (200MHz) is to control the read/write data path delay. ddr_clk and ddr_clkn.

III. EXPERIMENTAL RESULTS

To test the main control module, initially the clock is given 100MHZ. The module will not be initialized until reset_n signal is made high. Once this signal is made high, it is necessary to make the delay signal (sys_dly_200us) high. This implies that the controller is provided with the necessary delay. After providing the delay signal, after some ns delay the controller gets initialized. This is indicated by the signal (sys_init_done) becoming high. During this time all the initialization FSM states are executed. Figure 5 shows the simulation output of main control module.

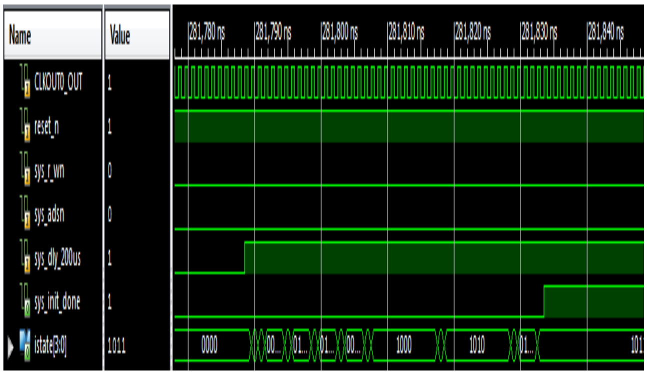

Figure 5: Main Control Module simulation output

Reset is made high and required system delay (200us) is given. Once the system gets initialized, the required data is written to the particular memory location. Now we can read the same data from memory. Figure 6 shows the final simulation output.

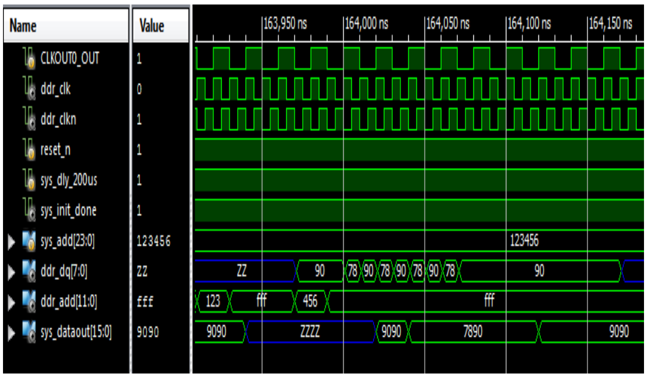

Figure 6: Simulation output of SDRAM memory controller

IV. CONCLUSION

A memory controller is proposed, which can be used for real-time systems. The memory controller is co-designed for the use with Ethereal, but can be used with in any system. The predictability of the system is reached with an interleaved access pattern and a predictable scheduler. This allows a gross-to-net bandwidth translation of the memory.

Double Data Rate (DDR) SDRAM Controller is based on ASIC design methodology. The blocks are designed by using Verilog HDL. The issues were analyzed with help of RTL Simulation and Synthesis using RTL Compile.

Furthermore, a high efficiency is gained with this access pattern, where the efficiency stays above 80%. The maximum latency for an access is calculated analytically at design time and is for highest priority requestor in the worst-case 375ns.

Thus an efficient fully functional memory controller is designed. The controller generates different types of timing and control signals, which synchronizes the timing and control the flow of operation. The memory system operates at double the frequency of processor, without affecting the performance. Thus we can reduce the data bus size. The drawback of this controller is complex schematic with large number of buffers in the circuit increases the amount of delay.

REFERENCES

[1] Z. Peyton Peebles, Jr. "Digital Communication Systems," PrenticeHall, INC

[2] J. Shtargot, “Design Guidelines for High-Performance, Multichannel, Simultaneous Sampling ADCs in Data-Acquisition Systems”, Maxim semiconductors.

[3] P. Murray, A. Corporation, and F. A. Hawari, Cadence Design Systems, Inc. “Challenges in implementing DDR3 memory interface on PCB systems: a methodology for interfacing DDR3 SDRAM DIMM to an FPGA”, Design Con 2008.

[4] Zude Zhou, Songlin Cheng, and Quan Liu, “Application of DDR Controller for High-speed Data Acquisition Board”, Innovative Computing Information and Control International Conference on 2006, pp. 611-614.

[5] Hadley M. Siqueira, Ivan S. Silva, Marcio E. Kreutz, Edgard F. Correa, “DDR SDRAM Memory Controller for Digital TV Decoders”, Brazilian Symposium on Computing System Engineering (SBESC) 2011, pp. 78-82.

[6] Daou, A. Kassem, Hamad, M, El-Moucary C, “FPGA Implementation of a Three-Way Asynchronous DDR2 Memory Controller”, 2012 2nd International Conference on Advances in Computational Tools for Engineering Applications (ACTEA), pp. 334-337.

[7] Gurram Rajesh, K Srinivasa Reddy, U Srinivasa Rao, “ASIC Design Methodology & Implementation of Double Data Rate (DDR) SDRAM

Controller", IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278-2834, p-ISSN: 2278-8735.Volume 9, Issue 4, Ver. IV (Jul - Aug. 2014), pp. 62-67.

[8] Deepali Sharma, Shruti bhargava, Mahendra Vucha, "Design and VLSI Implementation of DDR SDRAM Controller for High Speed Applications", International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 2 (4), 2011, pp.1625-1632.

[9] Stephen Chin, Mohammad Ghayour Attorney, "Under-Sampling Techniques Simplify Digital Radio", 1995.

[10] KarthikChandrasekar, Benny Akesson, Kees Goossens, "Run-Time Power-Down Strategies for Real-Time SDRAM Memory Controllers", In Proc. Design Automation Conference (DAC), June 2012, pp. 988-993.

[11] Santhoshima K G, Shilpa K , "FPGA Implementation of Optimized DDR SDRAM Controller", International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE) ISSN: 0976-1353 Volume 14 Issue 2–APRIL 2015, pp. 44-48.

[12] Datasheet, "Cyclone II Device Family Data Sheet," Altera Corporation. [9] Datasheet, "EV8AQ160 Quad ADC", e2v technologies plc. [10] Double Date Rate (DDR) SDRAM Specification, JEDEC Standard No.79C

[13] www.xilinx.com/apps/xapp.htm.

[14] www.xilinx.com/univ/beginnersbookjune2003ver2.pdf

[15] Micron Technology Inc. Double 'Data Rate controller' white paper.