SIMULATIVE INVESTIGATIONS OF DDR2 (SDRAM) MODEL IN HDL

Ajay Kumar[1], V.Sulochana Verma[2]ACSD Department, CDAC, Mohali

Abstract

This paper presents a simulation based implementation of DDR2 memory model in HDL language. The study covers single data rate, double data rate and their comparisons. This model consists of five blocks i.e. Instruction decoder, DDR2 interface, SRAM interface, bank control and adder reorder. This work is evaluated using Xilinx 12.4 ISE and ISIM is used for simulation.

Keywords: SDR, DDR, DDR2, SDRAM.

Introduction

With the increase of computer performance in recent years, the requirements of higher bandwidth and better memory performance are growing day by day. There is a frequent demand for computer memories to be faster, larger, lower powered and physically small [2]. To compare with the predecessors of double data rate RAM, DDR2 SDRAM (DDR2) has higher speed, higher bandwidth, higher power efficiency and higher stability. A recent study show that DDR3 will replace DDR, DDR2 in one or two years, but DDR2 is still occurs in general use. If this compatible memory model can be included in the computer architecture, the system will be more flexible and transplantable.

DDR2 was first published in 2004; it has higher stability and higher speed [5]. DDR2 is a new memory standard approved by JEDEC (Joint Electronic Device Engineering Council) that comprises a great number of DIMM, chip, and chipset manufacturers. The standard's earlier versions were published back in March 2003, but it wasn't until January 2004 that it gained the ultimate approval. DDR2 is based on a well-known and time-proved DDR (Double Data Rate) technology. According to a recent survey, 30 percent of the memory market is DDR and 40 percent is DDR2. But it has been forecast that the percentage of DDR will decrease to 20 while that of DDR2 will increase to 50 [3].

This paper provides a brief and detailed analysis of DDR2 and the performance comparisons between DDR and DDR2, and then, a memory model/ architecture based on a common standard bus interface is implemented and synthesized. If this memory model can be applied in the Computer or Digital TV platforms; it will be very beneficial during the period when DDR is completely replaced by DDR2. It will enormously shorten time to market; reduce manual work and lower product cost.

Differences between SDR SDRAM and DDR SDRAM

DDR SDRAM is a type of SDRAM that inherits technologies from SDR SDRAM and realizes faster operation and lower power consumption. It shares many common aspects with SDR SDRAM, which enables easy transition to DDR SDRAM [4].

The table 1 explains the differences in functions and specifications between DDR SDRAM and SDR SDRAM.

Table 1- Differences in Functions and Specifications

Item	DDR SDRAM	SDR SDRAM
Data transfer frequency	Twice the operation frequency	Same as the operation frequency
Data rate	2/tck	1/tck
Clock input	Differential clock	Single clock
Data strobe signal (DQS)	Essential	Not supported
Supply voltage	2.5 V	3.3 V
Burst length	2, 4, 8	1, 2, 4, 8, full-page (256)
Burst sequence	Sequential/Interleave	Sequential/Interleave
Data mask	Write mask only	Write mask/Read mask

DDR SDRAM achieves a double data transfer rate which is twice the clock frequency by implementing a 2-bit prefetch architecture. The size of internal bus is twice the size of the external bus.

For SDR SDRAM, data I/O is synchronized with the clock's rising edges only, whereas for DDR SDRAM, the

data I/O is synchronizes with the clock's both rising and falling edges.

The DDR SDRAM is operated by the command input at the clock's rising edge similar to the SDR SDRAM. A data strobe signal (DQS) is used to have a high speed transfer rate. This signal is output from the driver and it is received by the receiver adjusting the data capture timing.

The power supply of DDR SDRAM is 2.5V, whereas the power supply of SDR SDRAM is 3.3V. This reduction in power supply reduces the power consumption of the embedded circuits. DDR SDRAM has mode register set commands for the latency, burst length, and burst sequence similar to the SDR SDRAM.

The DDR SDRAM has write mask only, it uses a write mask enable signal (DM) which masks write data, but the read mask is not supported here. Whereas SDR SDRAM can mask both read and write data [6].

DDR2 Memory Model

The DDR2 model is a configurable and synthesizable model which uses external memories in order to store data. DDR2 frequency is higher than DDR1 (DDR) frequency [1]. Furthermore DDR2 has better efficiency because it has been extended with new features such as "additive latency" feature. DDR2 SDRAM is a synchronous dynamic memory which uses double data rate architecture to attain high speed operation. The memory transfers two data words every clock cycle at the input/output pins. The write and read accesses are burst oriented. For sending or receiving data, the memory controller provides an address and a command.

A. Specifications

- Model supports all DDR2 configurations and all SRAM implementations.
- It provides an advance debugger mode.
- It is synthesizable for FPGA integration.
- Model is "cycle accurate", which means that the timing is accurately modeled on a clock cycle basis.
- A safe VHDL code is implemented to allow future model update.

B. DDR2 Model interface

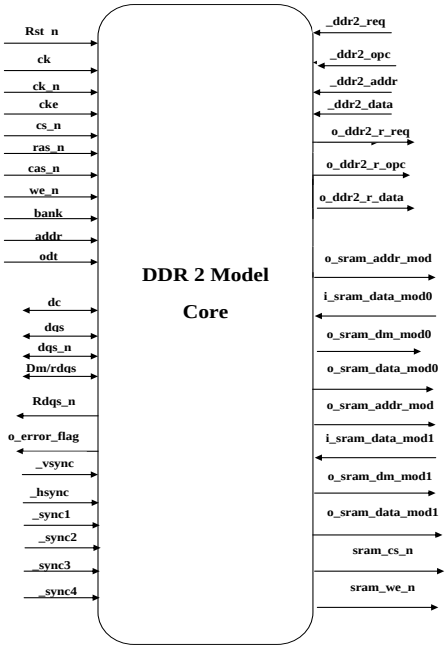

Figure 1- DDR2 Model Core

Design of DDR2 Model

The DDR2 model architecture comprised of mainly five blocks, namely-

A. DDR2 Interface Block

The DDR2 interface block sends data to LMI (Local Memory Interface) or receives data from LMI. The received data is trigged on DQS signal and the sent data is trigged on the following clock value. This block generates the data strobe signals (DQS) when a read command is issued. DQS path is a three state bus. When a rising edge on system clock occurs, then two data is sent to the SRAM block. When the memory uses more than one data strobe signal then an AND function is used. The SRAM block data bus width between DDR2 interface block and SRAM block width is twice the DDR2 data bus width [1].

B. Instruction Decoder Block

The commands (RAS, CAS, CKE, WE) from LMI are transcript into the advanced commands. These commands represent the standard instructions of DDR2. In simple words, it decrypts instructions, which are given by CAS, RAS, WE and CKE commands. These commands are re-grouped in a VHDL package.

C. SRAM Interface Block

This block reads or writes data into ASRAM memories according to the commands issued by the bank control block. The data path in SRAM block is split into two elements, SRAM1 with high bits of data path and SRAM0 with low bits of data path. Thus, the main aim of this block is to read/write data into SRAM0 memory corresponding to even address and data into SRAM1 memory corresponding to odd address. To compensate the dual data rate of DDR2 memory, the SRAM block data bus width is twice the DDR2 data bus width. It generates write enable (WE) and chip select (CS) signals. It also yields data masking signals from control signals.

D. Adder Reorder Block

This block executes an address reordering because a previous block (LMI) may have changed the order of address. The address order is changed to match with the bus address order.

E. Bank Control block

It checks whether a valid sequence command is coming from LMI or not. This block regroups all the FSM (Finite State Machine) i.e. *all bank FSM* and *single bank FSM* to check logical sequence. Power down (active and precharge), Refresh (auto and self), load mode register commands are handled by all bank FSM. Read, write, active, precharge are handled by single bank FSM [1].

F. DDR2 Top Level Diagram

Figure 2 shows the top level architecture of DDR2 memory model. This model consists of five blocks which are grouped to form a whole memory model. The whole model is categorized in two paths-

(i) Subscript $_{(1)}$ represents the control path. The control path consists of two blocks- instruction decoder block and bank control block.

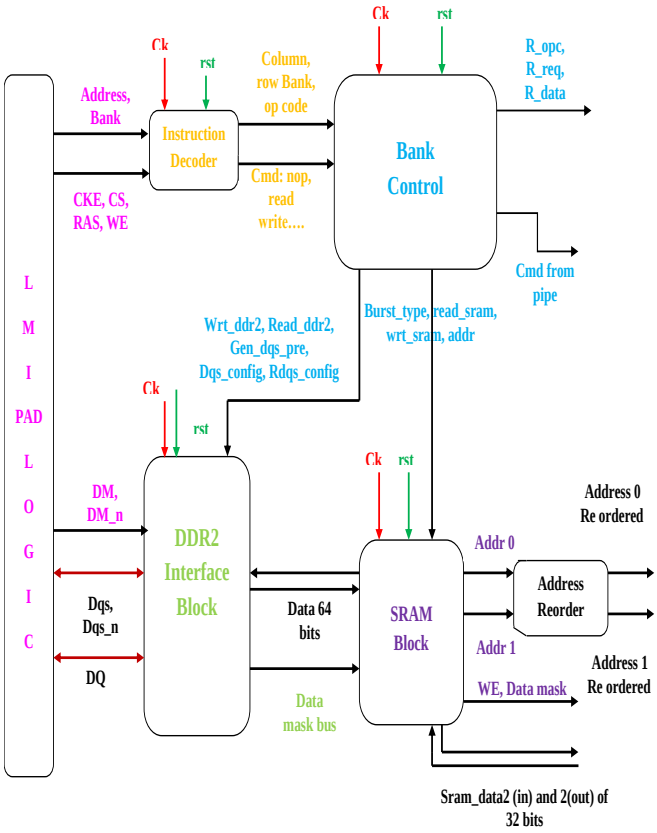

Figure 2 – DDR2 Top Level Diagram

(ii) Subscript $_{(2)}$ represents the data path. The data path consists of three blocks- DDR2 interface block, SRAM interface block and address reorder block.

Simulation results and synthesis

The simulation results of all the five blocks are shown individually. The results consist of RTL Schematic of the block and the simulations. The syntax of the RTL design is checked by using Xilinx tool.

A. RTL Schematic and Simulations of Instruction Decoder Block

Figure 3.1 shows the RTL Schematic of the instruction decoder. The commands (*RAS, CAS, CKE, WE*) generated from LMI are the inputs to this block. *Reset* and *clock* are the global inputs. The output (command) will change according to the conditions given to inputs. The values of bank address and address will be shown in address outputs i.e. *bank* and *row_addr*.

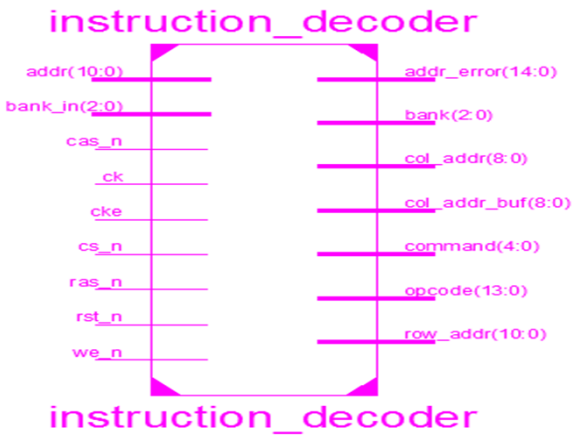

Figure 3.1- RTL view of instruction decoder

Figure 3.2 shows the simulation results which show that if *reset* value is given 0, then the output command (*cmd*) will be *desel* and *opcode*, *row_addr* and *col_addr* are 0. Now when *reset* is equal to 1 and clock event is forced, then the input *addr value* is transferred to *col_addr_tmp* signal. When value of *reset* is 0, the bank signal will be 0, and if *reset* is 1, then the input *bank_in* value will be transferred to bank output address. Initially the *addr_error* contains no value, but after giving reset value and forcing clock, the addr value with 10 bits are transferred to the *adder_error* signal.

B. RTL Schematic and Simulations of DDR2 Interface Block

Figure 4.1 shows the RTL Schematic of the DDR2 interface block which contains the *read_ddr2* and *write_ ddr2* as input signals which are used to read and write data yielded from the LMI controller. It also contains data bus whose width is twice the width of the DDR2 model data bus. When a rising edge on system clock occurs, then double data on single clock cycle is sent to the SRAM block. It uses data strobe signals to send/receive data. It has bidirectional signals i.e. *data_bus_in*, *dqs*, *dm_rdqs*, *dqs_n*, *dq_out*, *dqs_n_out* connected between LMI and DDR2 interface block

Figure 4.2 shows the simulation results which show that the read and write signals are generated by using the buffer signals. Then, all the data strobe signals are generated according to the *reset* and *clock* values. The *dq* signal is generated by forcing *read_ddr2* signal to 1 and forcing *clock* value. Also the *dqs* signal is generated according to the rising and falling edges of the clock. Next, the *rdqs* and *rdqs_n* signals are generated which depends on value of *dm_output*.

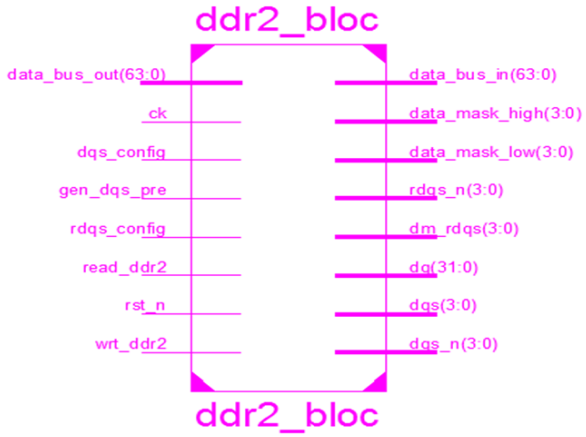

Figure 4.1- RTL view of DDR2 Interface

C. RTL Schematic and Simulations of SRAM Interface Block

Figure 5.1 shows the RTL Schematic of the SRAM interface block. The read and write accesses are burst oriented. Burst access can be sequential and interleaved according to the conditions given to inputs *read_sram* and *write_sram*.

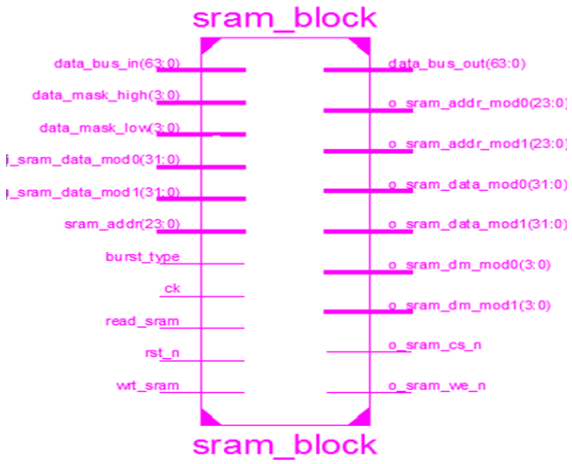

Figure 5.1- RTL view of SRAM block

Figure 5.2 (a) and 5.2(b) shows the simulation results of the SRAM block which shows the burst operation which can be sequential and interleaved. If *burst_type* is 0, then it is sequential and if *burst_type* is 1, then it is interleaved. Then accordingly the read and write signals are generated with the output signals *o_sram_data_mod0* and *o_sram_data_mod1*.

D. RTL Schematic and Simulations of Adder Reorder Block

Figure 6.1 shows the RTL schematic of the adder reorder block with *sram_addr* as input. After reordering of the address, the output *addr_re* is generated.

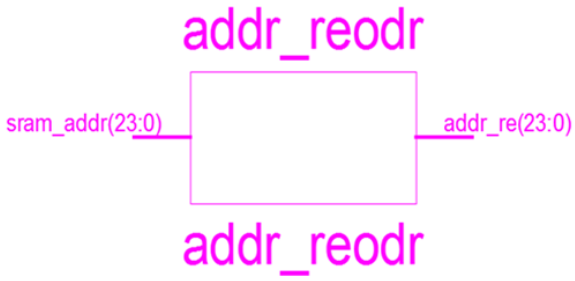

Figure 6.1- RTL view of Adder Reorder

Figure 6.2 shows the simulation results after reordering of the address. It is reordered in order to keep it compatible with the data bus address.

Conclusions

A DDR2 memory model has been designed using the LMI controller and SRAM memories with the desired simulation results. Therefore, if this memory model can be used in the computer architecture, then the system will be more efficient and flexible.

Acknowledgment

As with any enterprise, this research work could not have been completed without help and support of others. I would like to thanks my research advisor Mrs V.Sulochna Verma for their valuable contribution. Finally I wish to thank my family and friends for their support and encouragement.

References

[1] "DDR2 model for prototyping/ emulation" datasheet, STMicroelectronics, March 2007.

[2] Darshan makham "An Innovative Design of the DDR/DDR2 SDRAM Compatible Controller", international conference on nanoscience engineering and technology (ICONSET), 2011.

[3] Vimal Bhalodia "SCALE DRAM Subsystem Power Analysis", M.E. Thesis, MIT, September 2005.

[4] "How to use DDR SDRAM", user's manual, ELPIDA memory inc., August 2007, Japan.

[5] Chen Shuang-yan "An Innovative Design of the DDR/DDR2 SDRAM Compatible Controller", ASIC 2005, ASICON, 6th international conference, IEEE, 2005.

[6] "DDR2 SDRAM Device Operation and Timing Diagram", datasheet, HYNIX.

[7] B. Goossens "Architectures and modeling of predictable memory controllers for improved system integration", Design, automation and test in Europe conference & exhibition, 2011, IEEE.

[8] Shimojo, Y. Konno "High-density and high-speed 128 Mb chains FeRAM with SDRAM-compatible DDR2 interface", Symposium on VLSI Technology, 2009, IEEE.

Simulation results

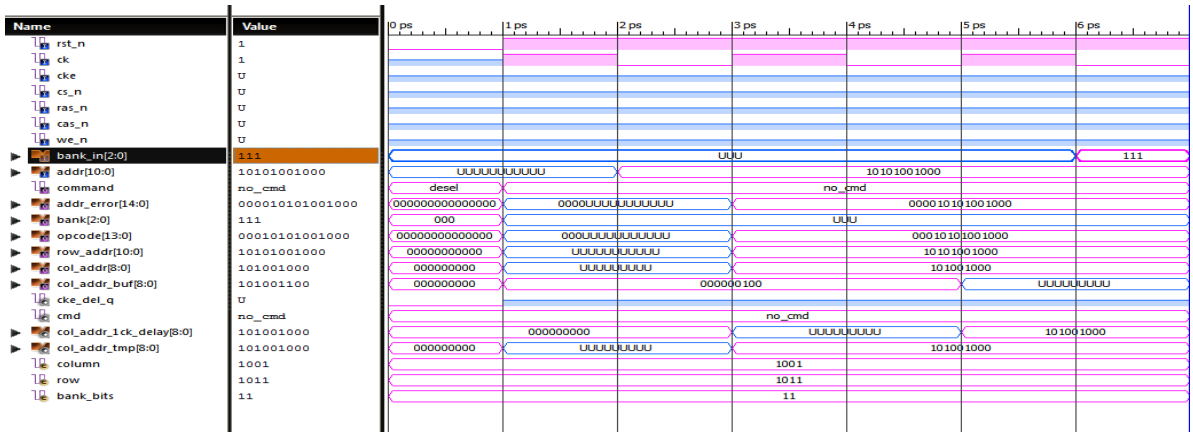

Figure 3.2 Simulation result of instruction decoder

Figure 4.2 Simulation results of DDR2 interface

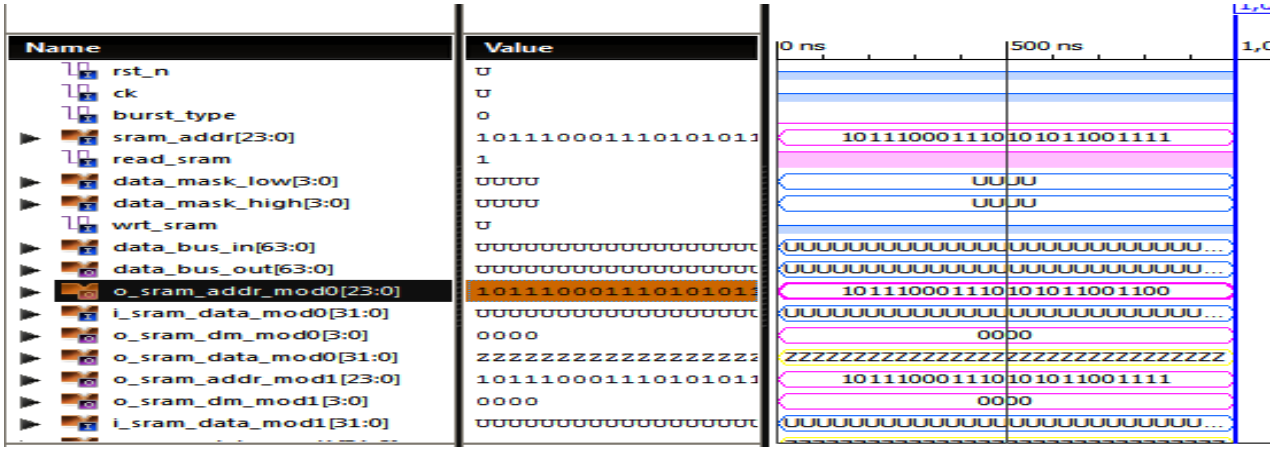

Figure 5.2 (a) Simulation results of SRAM block (Sequential Type)

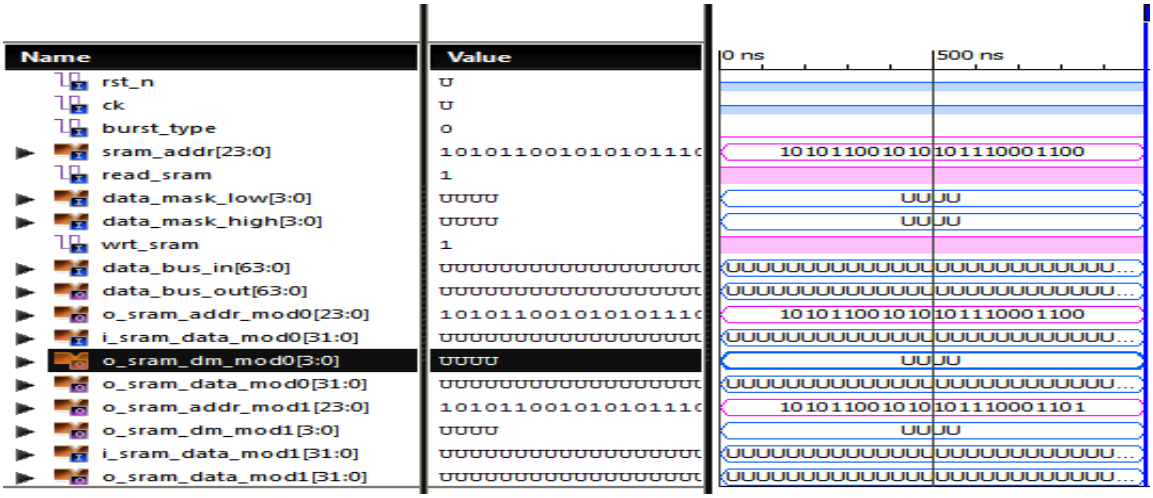

Figure 5.2 (b) Simulation results of SRAM Block (Interleaved type)

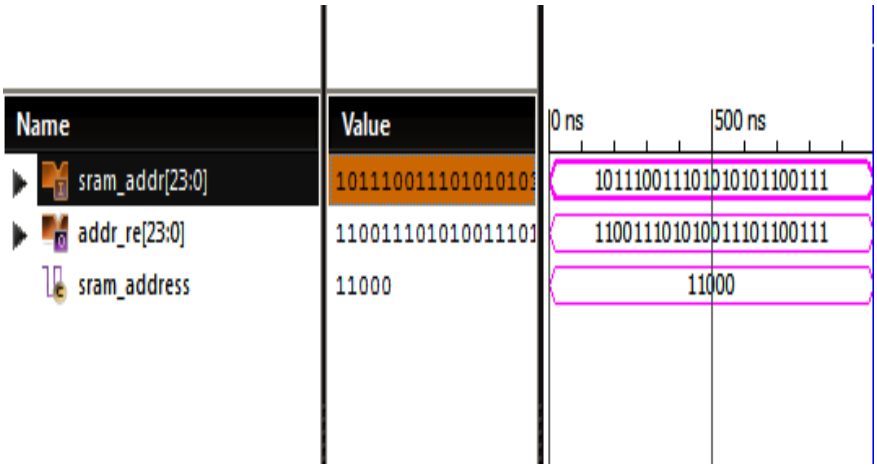

Figure 6.2 Simulation results of adder reorder

Bibliographies

Ajay Kumar has done his Bachelor of Technology degree in Electronics and Communication Engineering from Thapar University in 2009 and currently pursuing his Master of Technology degree in VLSI Design from CDAC, Mohali. His areas of interests are Analog and Digital VLSI Design, VHDL modeling and hardware implementation.

Vemu Sulochana has obtained her Bachelor of Technology degree in Electronics & Communication Engineering from JNTU Kakinada and Master of Technology degree in VLSI Design Automation & Techniques from NIT, Hamirpur in 2004 and 2009 respectively. She is working as a Project consultant at C-DAC, Mohali to conduct innovative research in the area of VLSI design. Her research interests include low power VLSI design, Computer-aided design (CAD), Digital & Analog VLSI Design. She enjoys teaching VLSI design, Device modelling, Low-power VLSI Design, Analog & mixed signal VLSI Design.