DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

FINAL PROJECT REPORT

PROJECT PROPOSAL FOR CIS 634: OBJECT ORIENTED SOFTWARE ENGINEERING

AUTHOR	BRINDA YENIBILLI (CSU ID: 2860564) KAVYASHREE SARAVANAN (CSU ID: 2861563)
REVIEWER	YONGJIAN FU
APPROVER	YONGJIAN FU
CONTRACTUAL DATE OF DELIVERY	12/08/2022

Table of Contents

Memories:

Computers use memories for storing data for immediate use and also for future use. The storage location where data is stored temporarily is called primary memory while the storage where the data are stored permanently for future use is called secondary memory. The primary memory is usually called as memory and secondary memory is called as storage. The computers will also have internal process memory for third type of storage, this memory is placed inside CPU or near CPU.

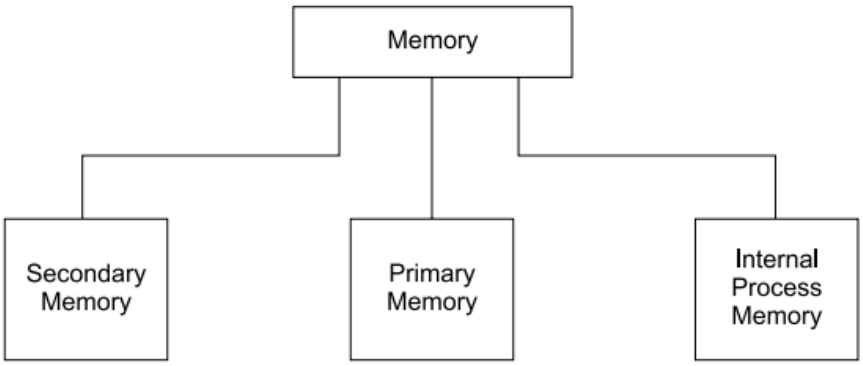


Fig:1

Random access memory (RAM):

Random Access Memory (RAM) is a volatile memory. It loses all its data when the power is switched off. It is the main memory of the computer system that stores the data temporarily and allows the data to be accessed.

RAM is made up of different ICs, which are mounted on a printed circuit board. RAM stores the application programs and the data on which the user is currently working so that the processor can easily access the required application program and data in a less amount of time. RAM is also known as read/write memory because it can perform both read as well as write operations. The speed of RAM is faster than the other memory devices, such as hard disk, floppy disk, etc.

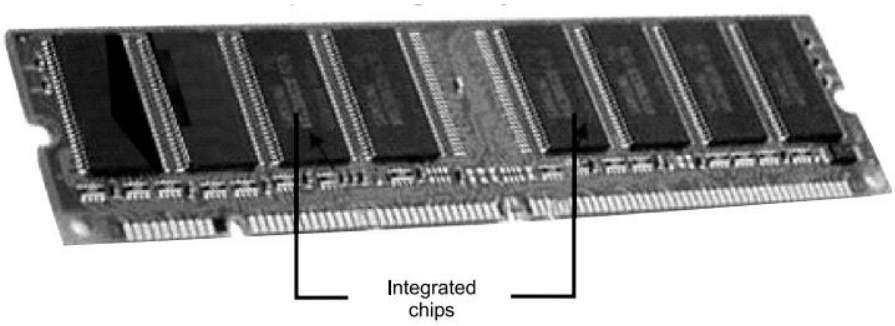


Fig: 2

Types of random-access memory:

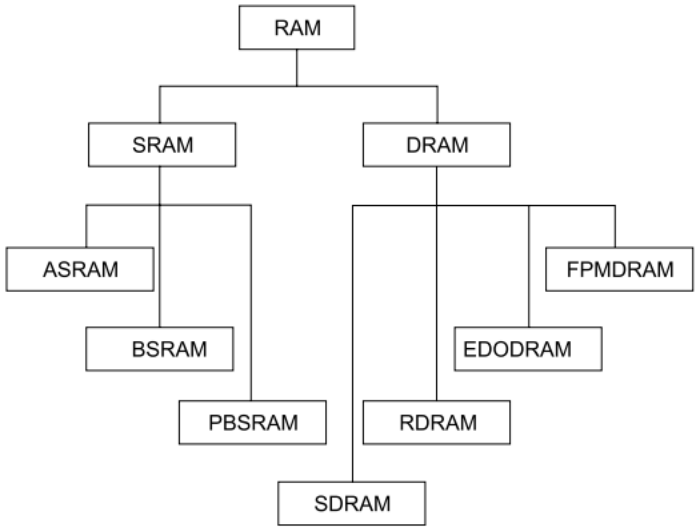


Fig: 3

Static Random-access memory:

Static RAM (SRAM) is a type of RAM which stores data till the power of the computer system is switched on. SRAM uses a number of transistors to store a single bit of digital information.

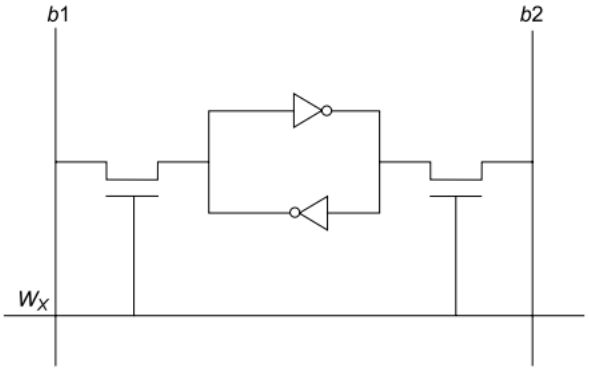


Fig: 4

Scope:

Every device consists of integrated circuits which are made up of semiconductors. Each device has its functions and certain applications to perform which are basically automated. The functions are fed into the devices using internal storage memories. Memory is used in any tiny device to huge equipment that has a CPU which is based on semiconductor technology. The demand for semiconductor memory is very high as the processors and computers are getting complex in nature.

Project Organization:

Steps to follow
Design Specification
Software Requirement
RTL Coding/ Design Model
Verification of Design
Compilation and Simulation
RTL analysis

Table: 1

Risk management:

- Milestones:
  - Develop initial version of design by considering all the required input specifications by 11/01/2022
  - To create test cases according to datasheet and verify the design modules by using test benches by 11/08/2022
  - Design and develop final stage of project by 12/08/2022

- Risk table:

Risk description	Probability of risk involved (%)	Level of risk involved (Level 1 TO 5)
Planning		
Confusion of project objective	50%	3
Confusion of project requirement	40%	3
Change in requirements	20%	2
Change of required tools	10%	1
Change in plan	50%	1
design		

Change in requirement-based design	50%	4
Confusion in design	50%	4
Change in design	30%	3
Implementation		
Inexperienced tools	30%	2
New library/new functions	30%	2
Test		
Improper test cases fed	80%	4
Delivery		
New modules to add	70%	4
Miss deadline	80%	4

Table: 2

	Negligible	Minor	Moderate	Significant	Severe
In percentage (%)	1	2	3	4	5
81% - 100%	Low risk	Moderate risk	High risk	Extreme risk	Extreme risk
61% - 80%	Minimum risk	Low risk	Moderate risk	High risk	Extreme risk
41% - 60%	Minimum risk	Low risk	Moderate risk	High risk	High risk
21% - 40%	Minimum risk	Low risk	Low risk	Moderate risk	High risk
01% - 20%	Minimum risk	Minimum risk	Low risk	Moderate risk	High risk

Table: 3

- Overview of risk mitigation, monitoring, management (RM3):

RM3/RMMM stands for risk mitigation, monitoring and management

where M1 => Mitigation (way to avoid the risk)
M2 => Monitoring (way to keep track of the risk)
M3 => Management (way to eliminate or reduce the risk)

Software tools used:

- ModelSim:

ModelSim

- It is EDA tool developed by mentor graphics.
- It is used for simulation of hardware description languages such as VHDL, Verilog and systemC.
- It simulates behavioral, RTL, and gate-level code.

- VIVADO:

VIVADO

- Vivado is a design tool.
- It is developed by Xilinx.
- It is used for synthesis and analysis of hardware description language (HDL) design.

Timeline chart:

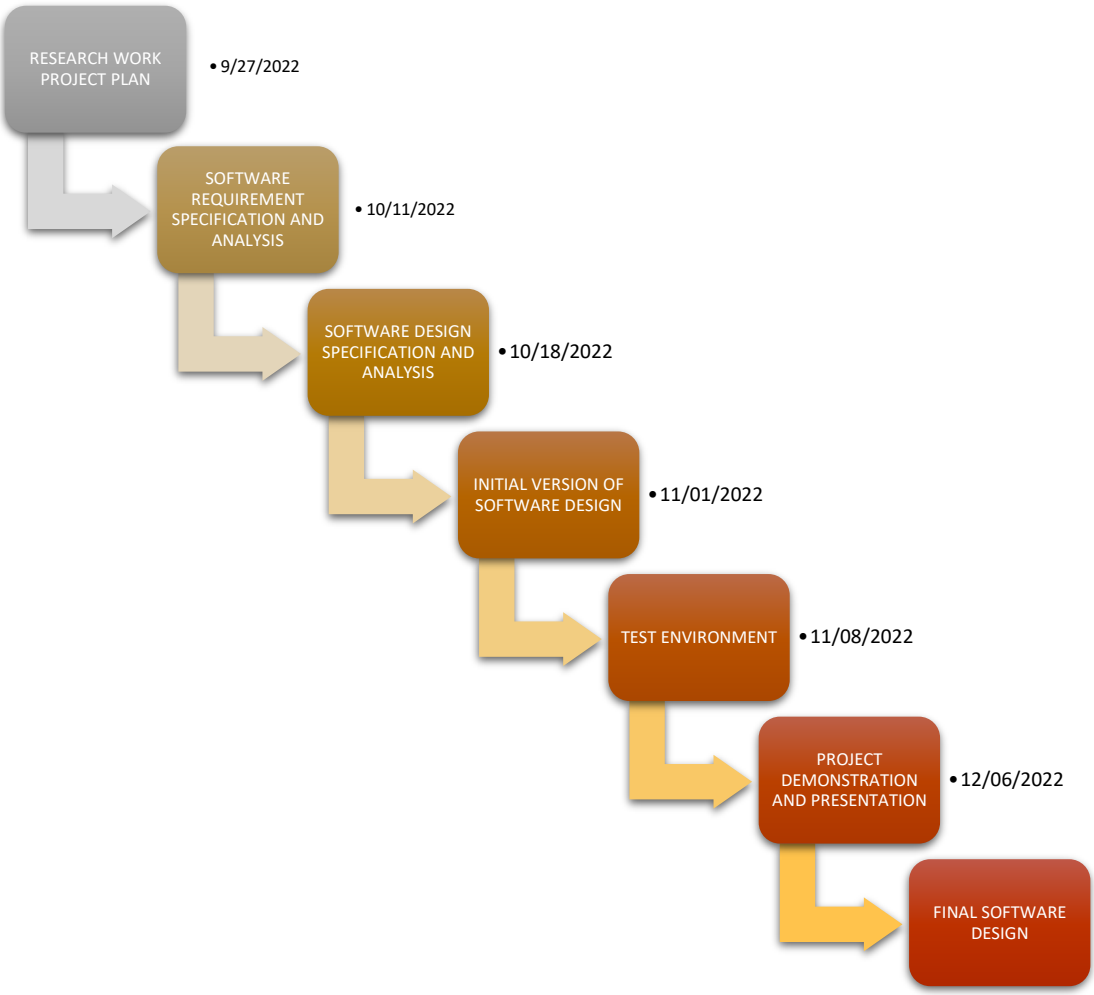


Fig: 5

Features:

- Access time: 8 ns, 10ns
- CMOS power operation: 130mA
- Power supply: 3.3v
- Industrial temperature: -40 ℃ to 85 ℃
- Byte control: it has access to upper bytes and lower bytes separately.
- All inputs and outputs are TTL compatible (transistor-transistor logic), our system contains digital input and digital output devices so the voltage levels are interpreted as 0 = 0.0 to 0.8v, 1 = 2.0 to 5.0v.

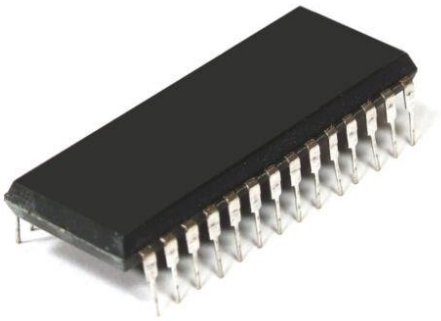


Fig: 6

Fig: 7

Recommended operating conditions:

VDD, dc supply voltage	3.0v to 3.6v
Tc, temperature range(I/O)	-55℃ to 125℃
Vin, DC input voltage	0v to 3.6v

Table: 4

Absolute maximum rating:
Exposure to absolute maximum rated conditions for extended periods may affect/damage device.

- VDD (DC supply voltage(I/O)) - -0.3 v to 3.8v
- Vi/o (voltage on any pin) - -0.3v to 3.8v
- Tstg (storage temperature) - -65℃ to 150℃
- Pd (maximum power dissipation) - 1.2W
- Ii (DC input current) - ±5mA

Component and description:

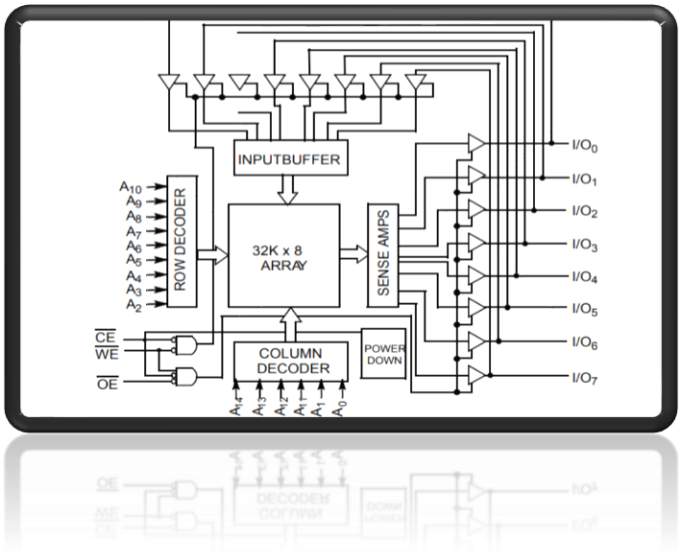


Fig: 9

- Address input buffer block: - The address inputs are used to select location in the sram memory chip. In real, you choose a memory location located in the chip when you select an address line for every input output on the chip. For instance, if we use 24-bit chip then it will have 24 data in/out, we can choose 24 memory locations simultaneously. Likewise, in our project we are using 4-bit data in/out i.e., it will have 4 addresses from each memory location. The number of address input pins is dependent on the size of memory.
- Input/output buffer**: -** This contains data signal which is used to data input and data output. Data signal comes in two forms- input and output are separate or input and output are common which using same signal for data transfer on the sram device. In write operation, a data signal is applied at the data input pin. The data is transferred into the signal and stored in the selected memory buffer block. In read operation, data from the selected memory block appears at the data output signal once operation is completed and output is enabled.
- Row decoder: - Selects a particular row location in the array of memory.
- Column decoder: - Selects a particular column location in the array of memory.

- Memory of array: - Combines both row line and column line and stores the data.
- Controller: - This consists of all the control signals which are responsible for the system operation. According to the applied conditions to control signals, the system operations take place and output is produced.

Black box:

It is a representation of input and output signals of the system.

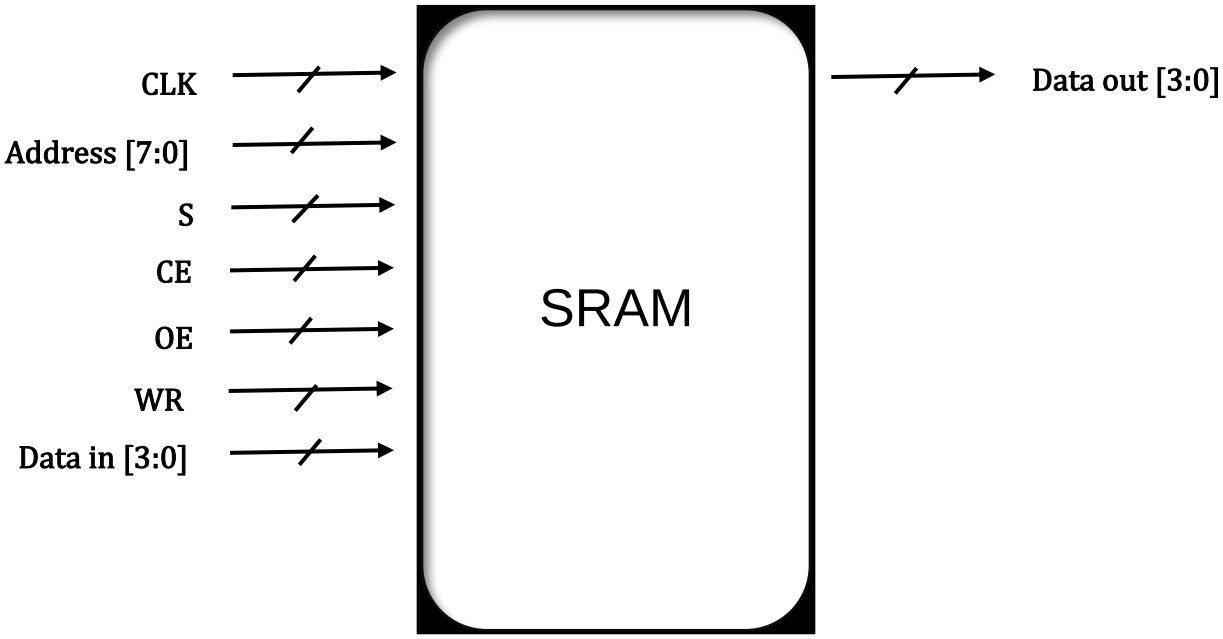


Fig: 10

Pin configuration:

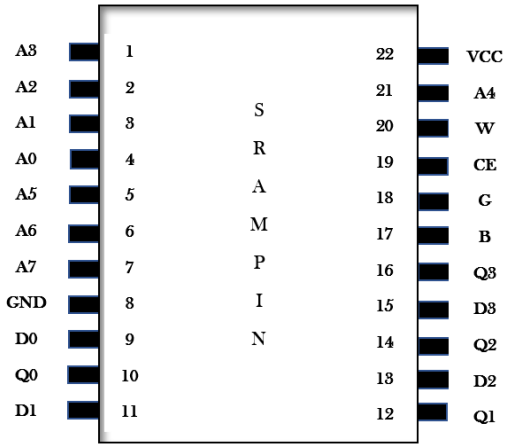


Fig: 11

Pin description:

Symbol	description
A0-A7	address input (only A0 to A3 address lines are used, remaining are unused)
D0-D3	data input/output
CE	chip enable input
W	write enable input
G	output enable input
GND	ground
Q0-Q1	lower bytes input
Q2-Q3	upper bytes input
VCC	power supply
B	chip select

Table: 5

NOTE: B or S = chip select, CE or E = chip enable, W or WE = write enable, OE or G = output enable

Logic symbol:

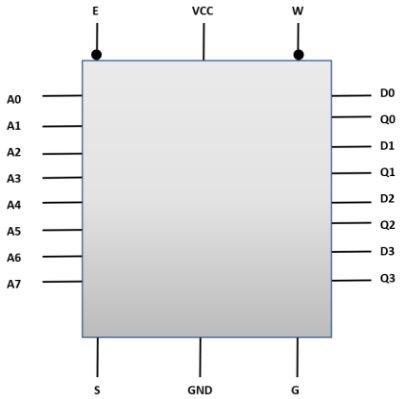


Fig: 12

A0-A7: - address input

S: - chip select

E: - chip enable

D: - data input

W: - write enable

G: - output enable

Q: - data output

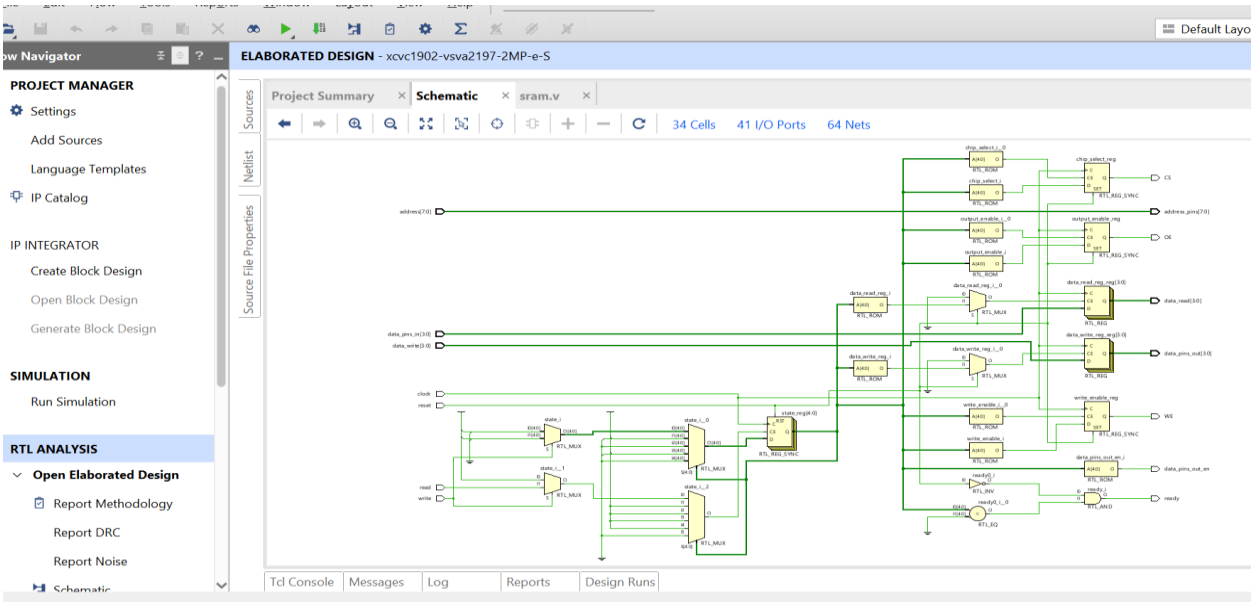
Operations:

- WRITE OPERATION
  - Data is written to the memory by selecting its row and column bocks using block selector, the data that has to be stored must be applied at the data input pins and the information must be stored in the sram cell.
  - Before clock pulse goes low to high, write operation is initialized. The row and column address must be applied to address input pins. Chip must be selected and write enable should be low. Data to be written has to be applied to the input pin.
  - After clock switches, chip select signal (B or S) should be low for enabling the chip select and selecting the chip.
  - Write enable (WE or W) is responsible for read or write operation, write enable is kept low at every clock pulse whenever we want out data to write into the memory. For instance, the data is written to selected address location i.e., data at address A0 corresponds to write DI0 data from the data input signal into the memory.
  - Output enable (OE or G) is made high i.e., it is in tristate mode, there is no use of it as we do not require to send the data from memory to other device.

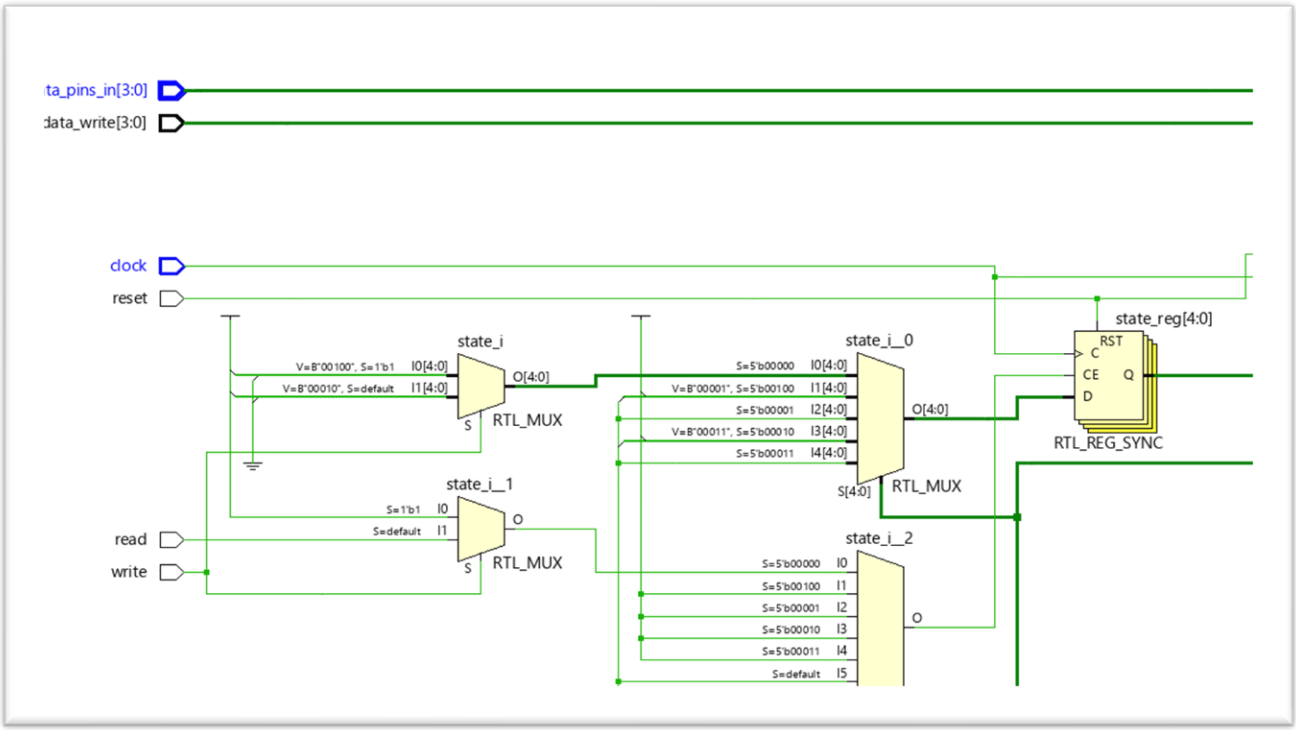
- READ OPERATION
  - The cell must be selected using row and column block selector to read data from sram memory. Particular cell has to be selected to send the information to data output.
  - Clock signal (CLK): These are the pulsating signals generating clock pulses for the system. We are designing sram using sequential registers so every event taking place in the system is dependent on clock signal.
  - Row and column address must be selected and sent to the address input signal before the next rising edge of clock signal i.e., next cycle. At this point, chip must be selected and write enable must be high.
  - After clock switches, chip select signal (B or S) should be low for enabling the chip select and selecting the chip. When chip select signal is high then the chip select pin goes into inactive state, chip cannot accept any input signal.
  - Write enable (WE or W) is responsible for read or write operation, when this signal is low write operation is enabled and read operation is enabled when this signal is high. Here the write enable is high has we want to perform the read operation.
  - Output enable (OE or G) is used to control the passing of data to the output. When OE is low the data appears at the output (DQ) and when this signal is high DQ is in tristate i.e., we get high impedance as the output at DQ. We keep the output enable low as we want our information to appear at the output DQ.

Conclusions:

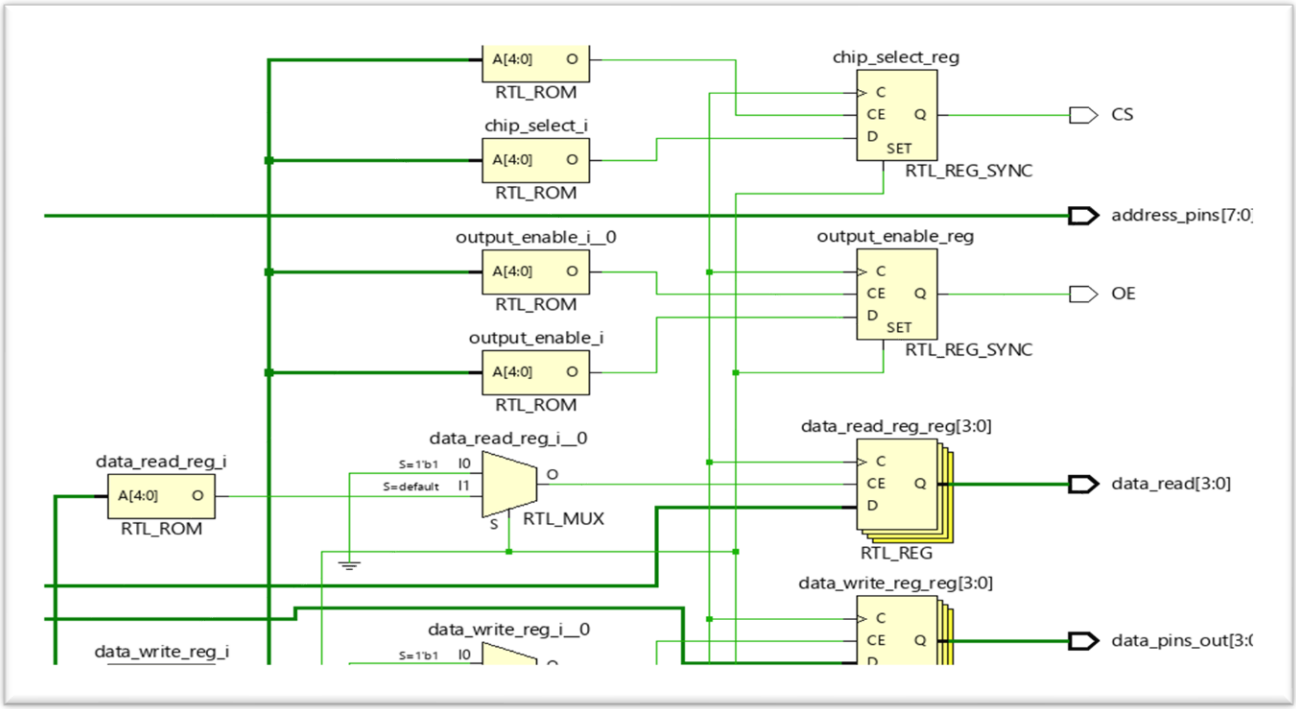
- Outcomes of the project (are all goals achieved?)
  RTL Schematic analysis:



Result: 1

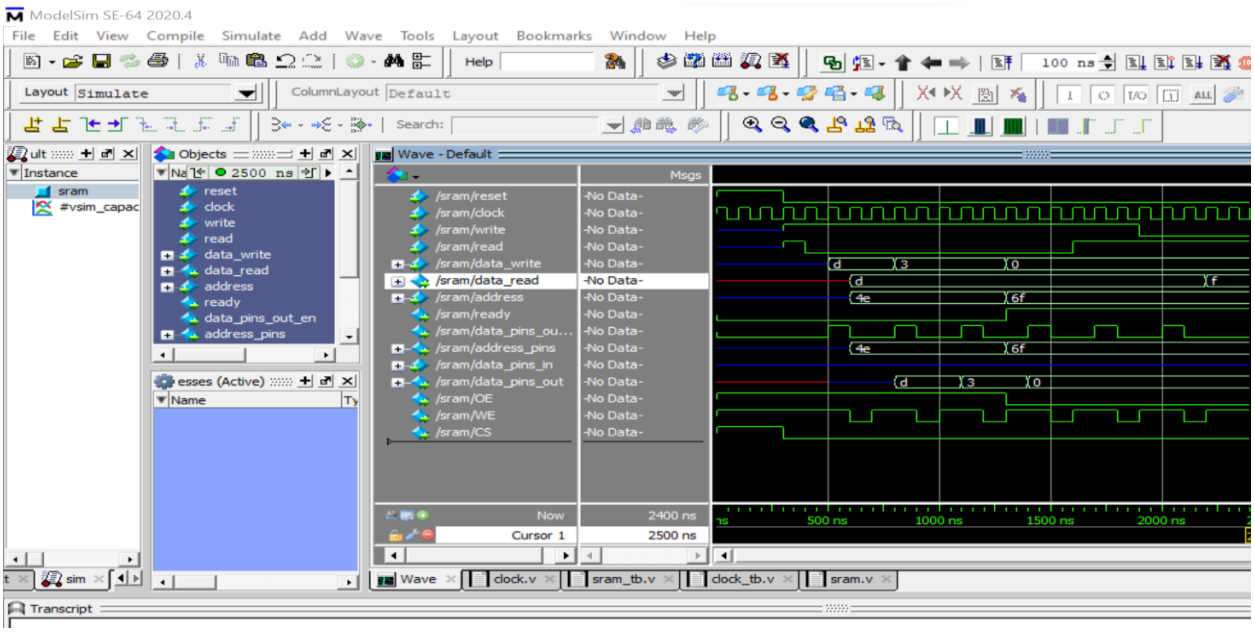


Result: 2

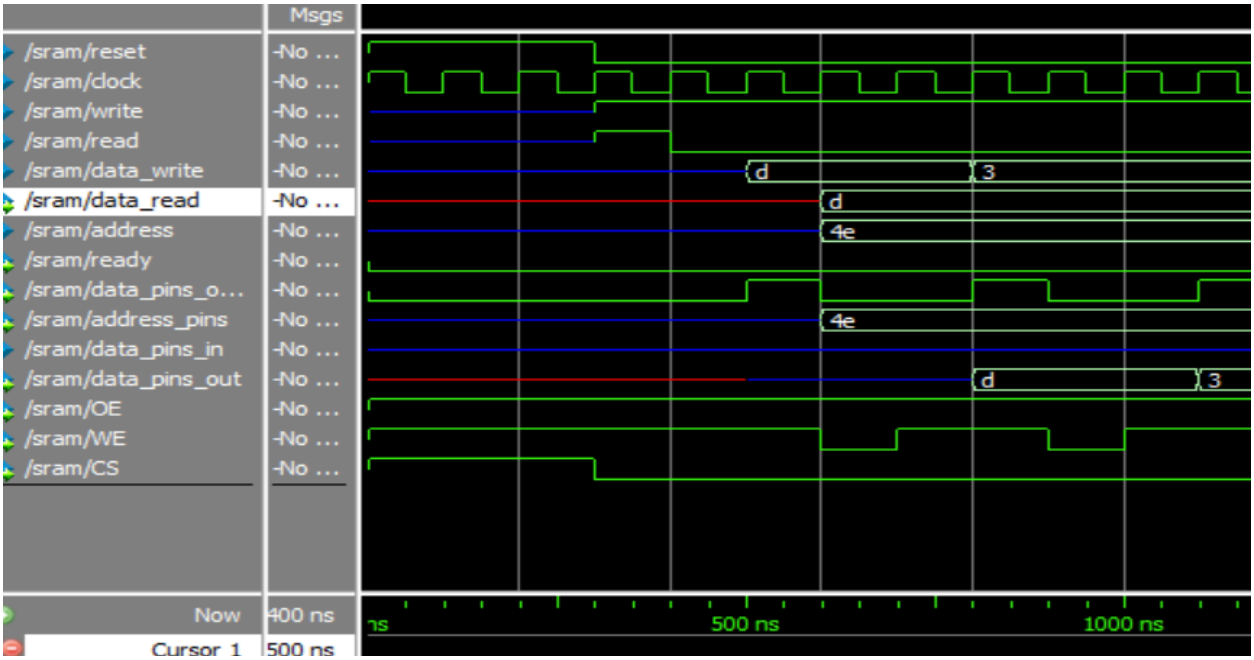


Result: 3

Simulation results:



Result: 4



Result: 5

➢ Lessons learned

- Firstly, understanding design specifications of SRAM memory clearly was 1st milestone.
- Installing the design verification software i.e., ModelSim was the part where we faced little difficulty, we followed the procedure but failed in installing one of its libraries so again had to reinstall it and did set up after installation.

➢ Future implementation

The below figure is FPGA (field programmable gate array) Altera DE2-115 board.

- We need to do the synthesis of our SRAM design by establishing connection to our system with the board.
- We need to give pin planning/configuration to the design.
- Take the synthesis results

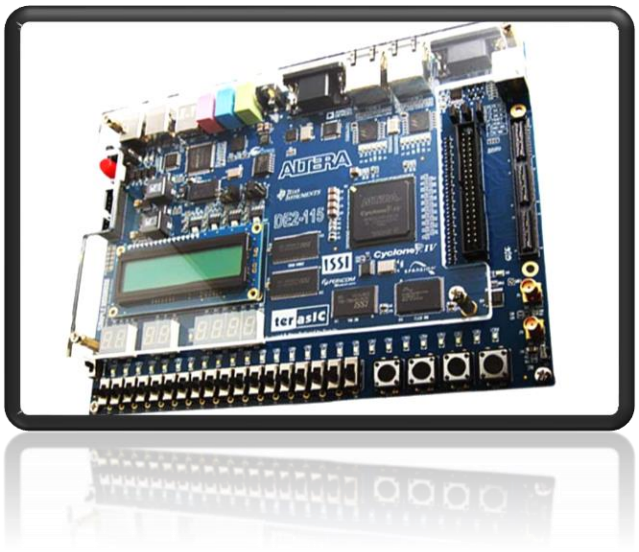


Fig: 13

References:

http://www.princeton.edu/~mae412/HANDOUTS/Datasheets/6116.pdf

https://www.st.com/resource/en/datasheet/stm32h743vi.pdf

https://www.winbond.com/hq/product/mobile-dram/pseudo-sram/?__locale=en

https://www.farnell.com/datasheets/109202.pdf