Group 24

Gorantla Thoyajakshi – 21CS10026

Ashwin Prasanth – 21CS30009

Computer Organization and Architecture Lab

Verilog Assignment 6, Question 1

- We defined an instruction of 10 bits size, where first 2 bits denote the opcode:
    - 00 (STORE_DATA) => moving 4 bit input data to memory location specified by 4 bit address. In this case, the next 4 bits of the instruction is the data, and the next 4 bits is the memory address
    - 01 (MOVE_TO_MEM) => moving data in register specified by 3 bit register no. to memory location specified by 4 bit address. In this case, leaving the next bit after opcode, the next 3 bits of the instruction is the register number, and the next 4 bits is the memory address
    - 10 (MOVE_FROM_MEM) => it does the reverse as the above – moving data from memory location to register. The other values are stored in the same way as above in the instructions
    - 11 (LOAD_DATA) => Loads data from memory location specified by 4 bit address. In this case, the last 4 bits (least significant) denotes the memory address
- We used a `case` block to execute the required operations accordingly
-
- The block memory was created with the following specifications:
    - Interface Type: Native
    - Memory Type: Single Port RAM
    - Write & Read Width: 4 (Since it stores 4 bit data)
    - Write & Read Width: 16 (Since there are 16 such units)
    - Operating Mode: Write First

- We also made a register module with the 8x4 register structure, similar to the memory block, with the following parameters:
    - dest -> 3 bit input denoting register number
    - en -> 1 bit enable signal input
    - we -> 1 bit write enable signal input
    - data_in -> 3 bit input data
    - data_out -> 3 bit output data
    - clk -> clock

- In the testbench, we tried 1 example of each operation, and it is working successfully