Group -8

Assignment 3

Bratin Mondal (21CS10016)
Somya Kumar (21CS30050)
Submitted on: August 24th, 2023

8 Register Implementation

1. **The Design:**

The register_file module has the following parameters:

1. input wire enabled: It is for setting the output port to enabled or disabled mode.
2. input wire res: Checks if the module is to be reset from the beginning. If res is set to 1 for starting with the registers for the first iteration of the loop and set to 0 for further iterations of the loop.
3. input wire [2:0] src_reg_num: It contains the source register number from where we need to pick data
4. input wire [2:0] dest_reg_num: It contains the destination register number where we need to store data
5. input wire move: It contains the flag if we want to move data from one register to another register
6. input wire in: It contains the flag if we want to take data input to a register from the input source.
7. input wire [15:0] data_input: The data input to be entered in some register
8. input wire clock: It is the clock pulse that is sent to the module
9. output reg [15:0] output_port: As output, it will store the value in the source register.

The module contains the following variables:

1. reg [15:0] registers [0:7]: 8 registers of 16-bit each for storage of data. It will be accessed and used as specified by the input commands.

2. **Working:**

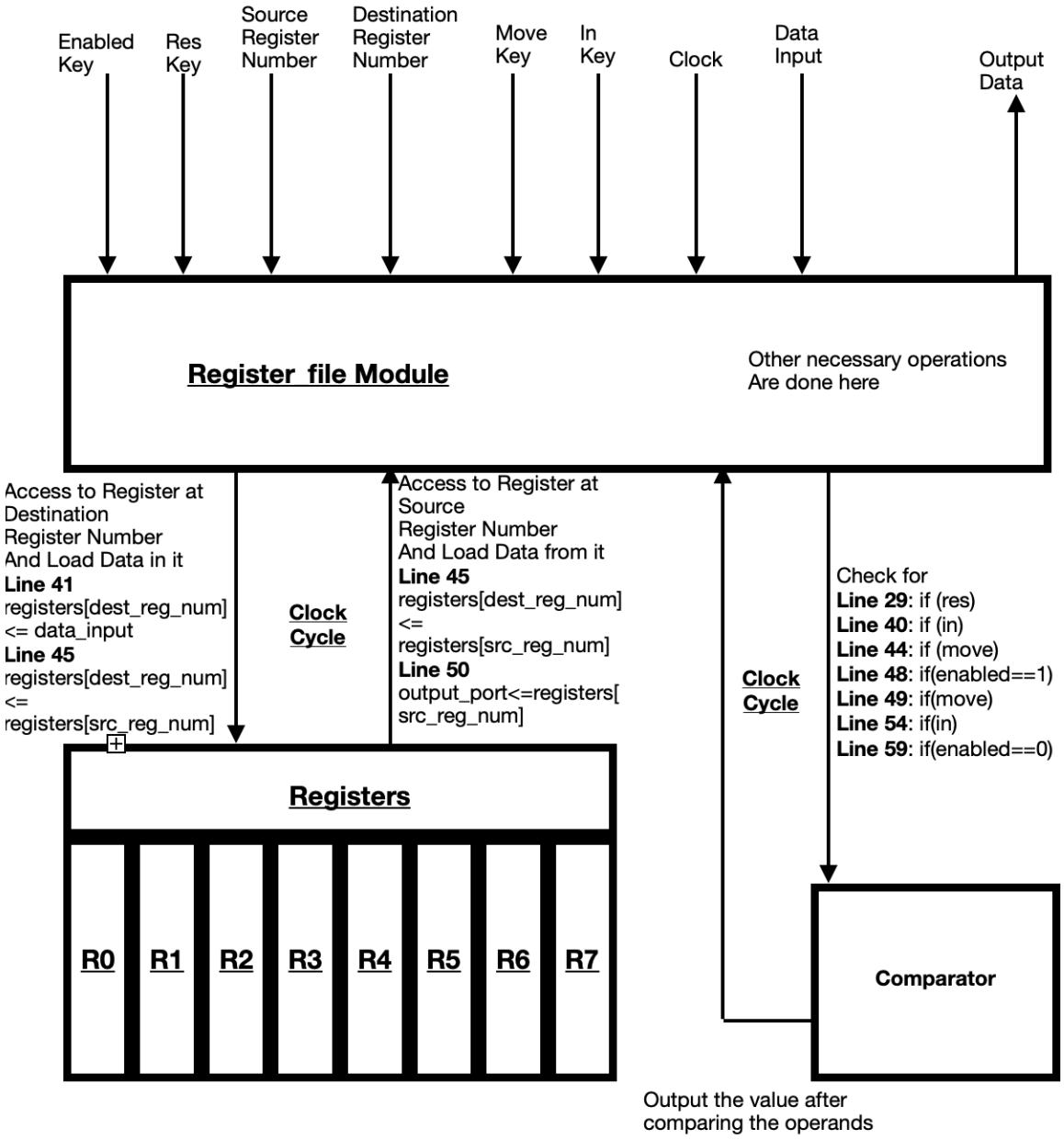
The workings of the module are as follows:

In each clock cycle:

1. If reset mode is on, we do the following. The particular line of the check condition in line 29 will be implemented through a **comparator** in the circuit.
    a. We set all the values of registers in register number 0 to register number 7 to 0.
2. If the in key is set to 1, we do the following.The particular line of the check condition in line 40 will be implemented through a **comparator** in the circuit.
    a. Move the value from data input to the register number loaded at dest_reg_num
3. If the move key is set to 1, we do the following.The particular line of the check condition in line 44 will be implemented through a **comparator** in the circuit.
    a. Move the value from register at src_reg_num to register at dest_reg_num
4. If the enabled key is set to 1, we do the following.The particular line of the check condition in line 48 will be implemented through a **comparator** in the circuit
    a. If the move key is set to 1, we do the following.The particular line of the check condition in line 49 will be implemented through a **comparator** in the circuit.
        i. Move data from register at src_reg_num to output_port
    b. If the in key is set to 1, we do the following.The particular line of the check condition in line 54 will be implemented through a **comparator** in the circuit.
        i. Move data from data_input to output_port
5. If the move key is set to 0, we do the following.The particular line of the check condition in line 59 will be implemented through a **comparator** in the circuit.

a. We set the value of output_pot to 16’bzzzzzzzzzzzzzzz

3. **Structure:**



The Test Bench:

The test bench we submit does a set of operations

We keep toggling the clock after every five time units.
The operations are as follows:

1. Initialize res with 1 for resetting
2. Initialize the clock with 0
3. Initialize enabled outport with 0, starting with disabled outport
4. Give sufficient time for resetting
5. **We load 123 in register number 3**
6. Set res to 0
7. Set *in* to 1 for reading input
8. Give 123 data input
9. Set dest_reg_num as 3
10. Enable the output port
11. Give sufficient time for changes
12. Display the results
13. **We load data from register number 3 to register number 4**
14. Set src_reg_num as 3
15. Set dest_reg_num as 4
16. Set move to 1
17. Set in to 0
18. Give sufficient time for changes
19. Display the results
20. **We load data from register number 4 to register number 6**
21. Set src_reg_num as 4
22. Set dest_reg_num as 6
23. Set move to 1
24. Set in to 0
25. Give sufficient time for changes
26. Display the results
27. **We disable the output line**
28. Set move to 0
29. Set in to 0
30. Disable output line
31. Give sufficient time for changes
32. Display the results