Introduction to Digital Logic

EECS/CSE 31L

**Assignment 4: Register File**

Prepared by: Jack Melcher

Student ID: 67574625

EECS Department

Henry Samueli School of Engineering

University of California, Irvine

February 15, 2015

**1 Block Description**

* Construct a register file that can be parameterized
* Per rising edge of clock, the register file can:
  + Write to up to one register
  + Read from up to two registers

**2 Input/Output Port Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Generic Name | Data Type | Default Value | Description |
| NBIT | INTEGER | 32 | Size of register |
| NSEL | INTEGER | 8 | Number of Registers |

|  |  |  |  |
| --- | --- | --- | --- |
| Port name | Port size | Port Type | Description |
| Clk | 1 | IN | Triggers the register process |
| Rst\_s | 1 | IN | Will reset the register when active and clock rises |
| We | 1 | IN | Enables writing to the registers |
| Raddr\_1 | NSEL-1 downto 0 | IN | The address of the register to be read from |
| Raddr\_2 | NSEL-1 downto 0 | IN | The address of the register to be read from |
| Waddr | NSEL-1 downto 0 | IN | The address of the register to be written to |
| Rdata\_1 | NBIT-1 downto 0 | OUT | The data value from the register that was read |
| Rdata\_2 | NBIT-1 downto 0 | OUT | The data value from the register that was read |
| Wdata | NBIT-1 downto 0 | IN | The data value to be written to the register |

|  |  |  |  |
| --- | --- | --- | --- |
| Variable Name | Data Type | Port size | Description |
| regfile | Array (0 to 2\*\*NSEL) | NBIT-1 downto 0 | The registers |

**3 Design Schematics**

**Truth Table:**

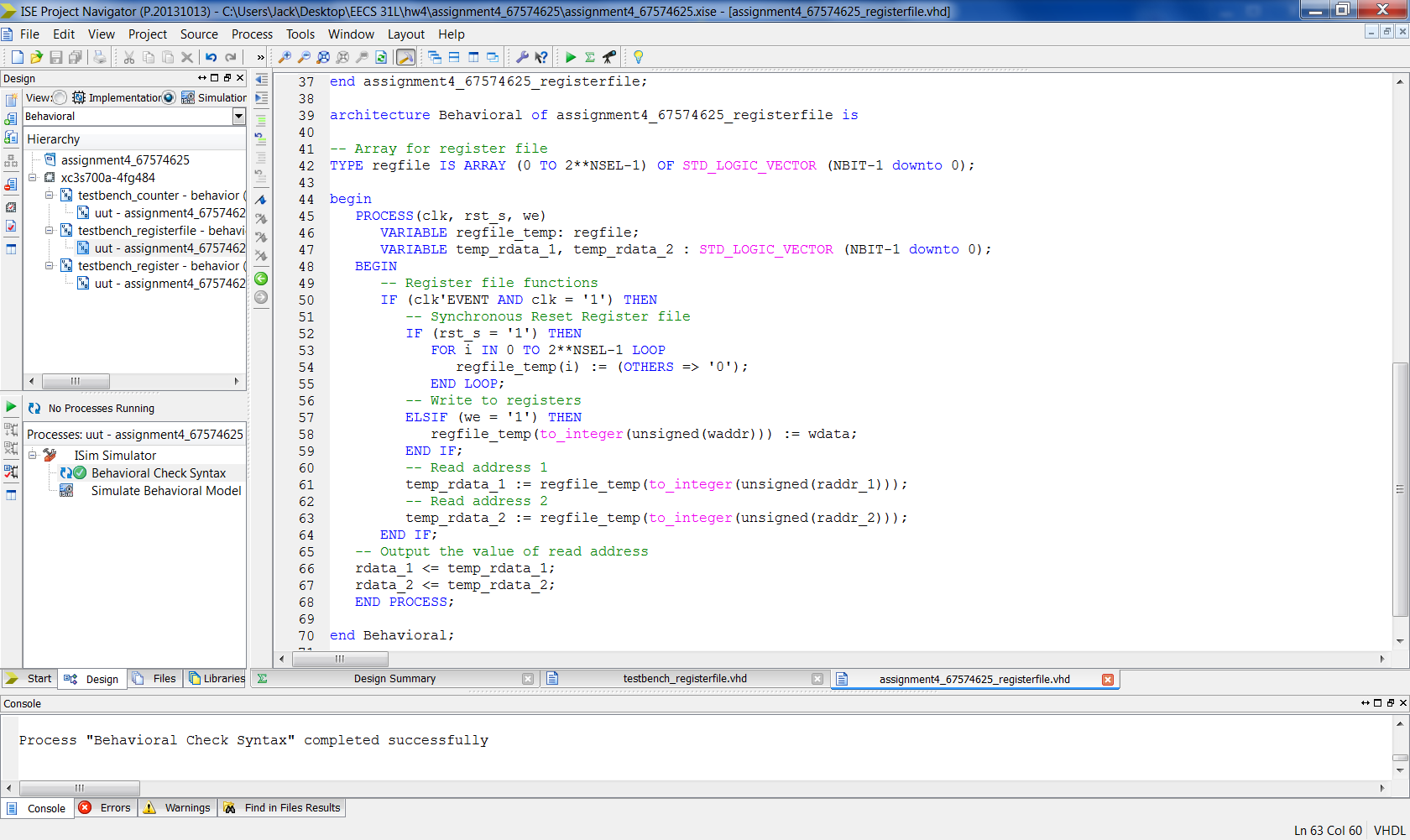
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Clk | Rst\_s | We | Raddr\_1 | Raddr\_2 | Waddr | Function |
| 1 | 1 | X | XXX | XXX | XXX | Synchronous Reset |
| 1 | 0 | 0 | XXX | XXX | XXX | Prevent writing  Also can read from any address |
| 1 | 0 | 1 | XXX | XXX | XXX | Allow writing to any address  Also can read from any address |

**Design:**

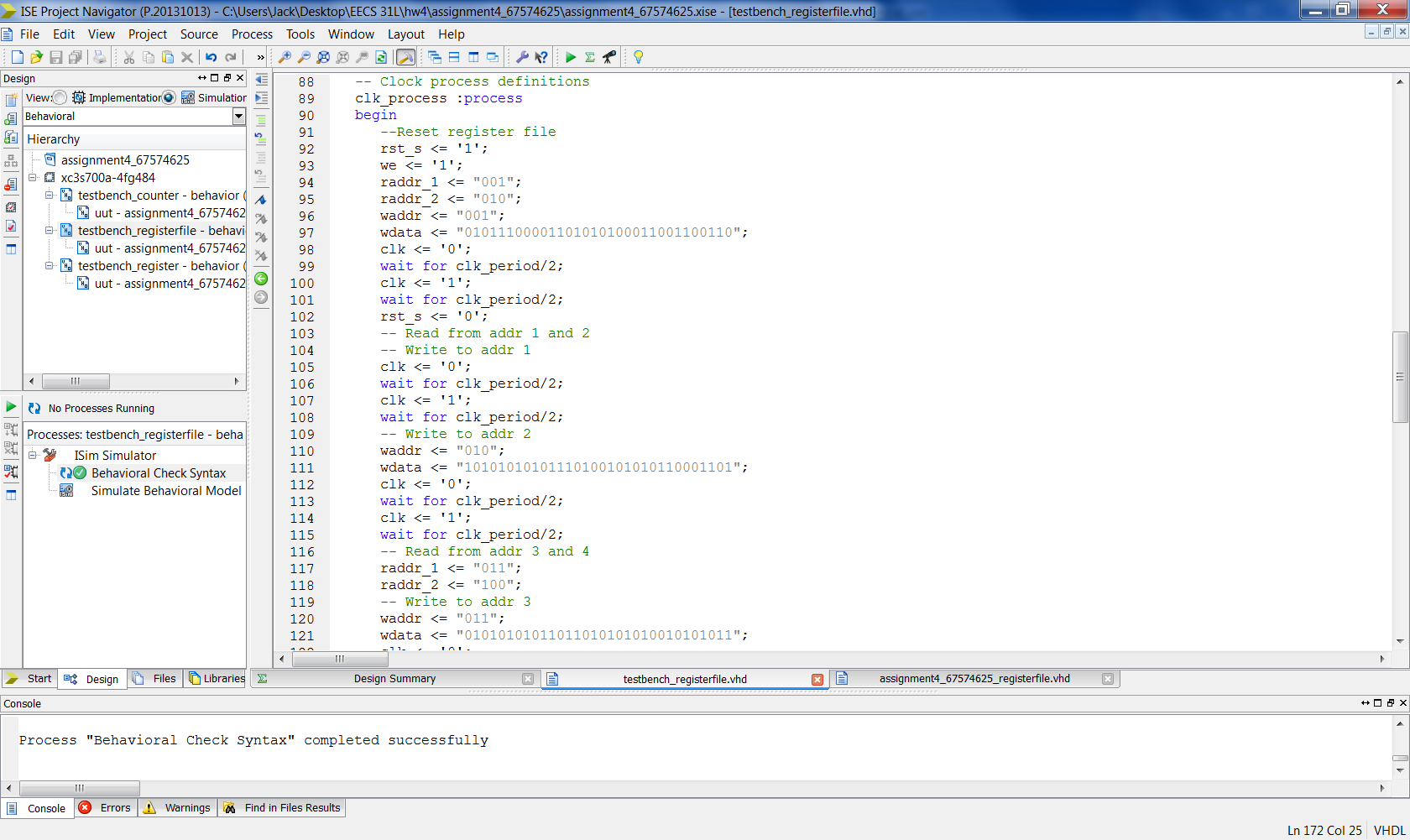


**4 Compilation**

Register File



Testbench



**5 Elaboration**

**Assumptions:**

* I didn’t need to use the register block to construct the register file.
* Instead, I implemented an array of std\_logic\_vectors that is basically a set of registers
* Utilizing the array and process, I was able to construct the register file

Regarding how I wrote my vhdl code

* I used a process with variables
* Used sequential if and elsif statements

**Errors and Challenges:**

* Misusing assignment operators; tried using signal assignment operator on a variable
* Understanding how to access the array properly. I realized that I needed to type convert

**Simulation Log:**

Started : "Simulate Behavioral Model".

Determining files marked for global include in the design...

Running fuse...

Command Line: fuse -intstyle ise -incremental -o {C:/Users/Jack/Desktop/EECS 31L/hw4/assignment4\_67574625/testbench\_registerfile\_isim\_beh.exe} -prj {C:/Users/Jack/Desktop/EECS 31L/hw4/assignment4\_67574625/testbench\_registerfile\_beh.prj} testbench\_registerfile {}

Running: C:\Xilinx\14.7\ISE\_DS\ISE\bin\nt64\unwrapped\fuse.exe -intstyle ise -incremental -o C:/Users/Jack/Desktop/EECS 31L/hw4/assignment4\_67574625/testbench\_registerfile\_isim\_beh.exe -prj C:/Users/Jack/Desktop/EECS 31L/hw4/assignment4\_67574625/testbench\_registerfile\_beh.prj testbench\_registerfile

ISim P.20131013 (signature 0x7708f090)

Number of CPUs detected in this system: 4

Turning on mult-threading, number of parallel sub-compilation jobs: 8

Determining compilation order of HDL files

Parsing VHDL file "C:/Users/Jack/Desktop/EECS 31L/hw4/assignment4\_67574625/assignment4\_67574625\_registerfile.vhd" into library work

Parsing VHDL file "C:/Users/Jack/Desktop/EECS 31L/hw4/assignment4\_67574625/testbench\_registerfile.vhd" into library work

Starting static elaboration

Completed static elaboration

Compiling package standard

Compiling package std\_logic\_1164

Compiling package numeric\_std

Compiling architecture behavioral of entity assignment4\_67574625\_registerfile [\assignment4\_67574625\_registerfi...]

Compiling architecture behavior of entity testbench\_registerfile

Time Resolution for simulation is 1ps.

Waiting for 1 sub-compilation(s) to finish...

Compiled 6 VHDL Units

Built simulation executable C:/Users/Jack/Desktop/EECS 31L/hw4/assignment4\_67574625/testbench\_registerfile\_isim\_beh.exe

Fuse Memory Usage: 35932 KB

Fuse CPU Usage: 498 ms

Launching ISim simulation engine GUI...

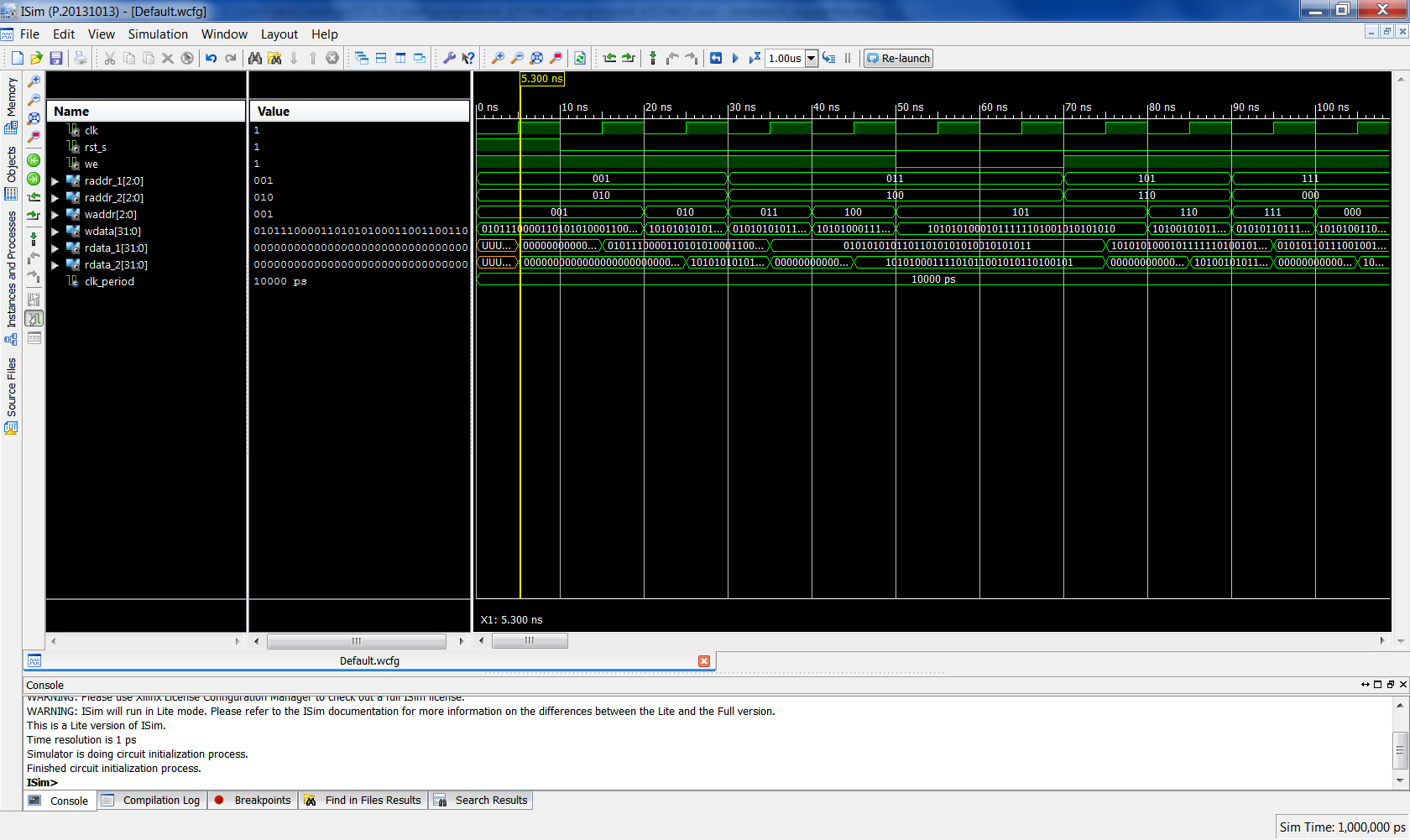
"C:/Users/Jack/Desktop/EECS 31L/hw4/assignment4\_67574625/testbench\_registerfile\_isim\_beh.exe" -intstyle ise -gui -tclbatch isim.cmd -wdb "C:/Users/Jack/Desktop/EECS 31L/hw4/assignment4\_67574625/testbench\_registerfile\_isim\_beh.wdb"

ISim simulation engine GUI launched successfully

Process "Simulate Behavioral Model" completed successfully

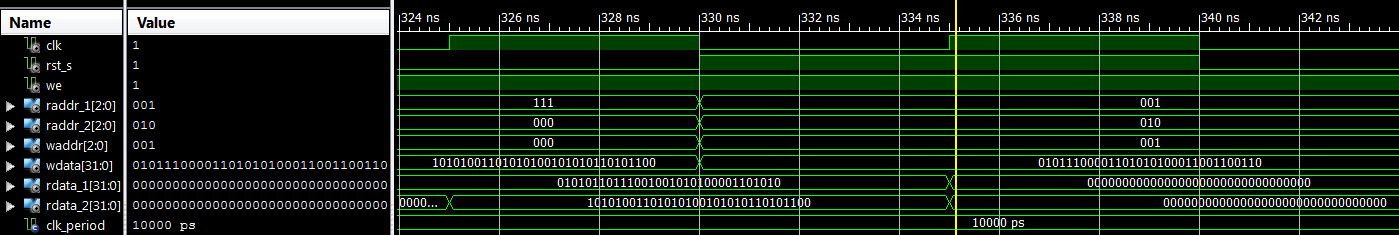
**6 Waveforms**

*Full Waveform Snapshot*

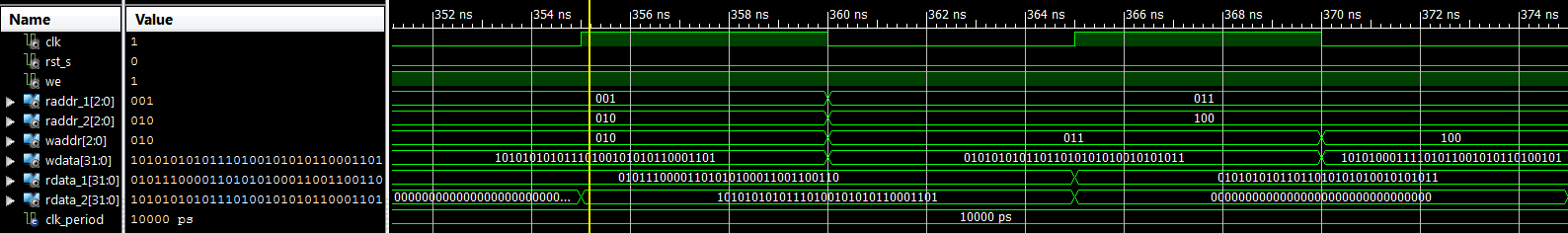


*Snapshots of Waveform*

Reset



Writing and Reading from various Addresses



Hold (write enable off)

