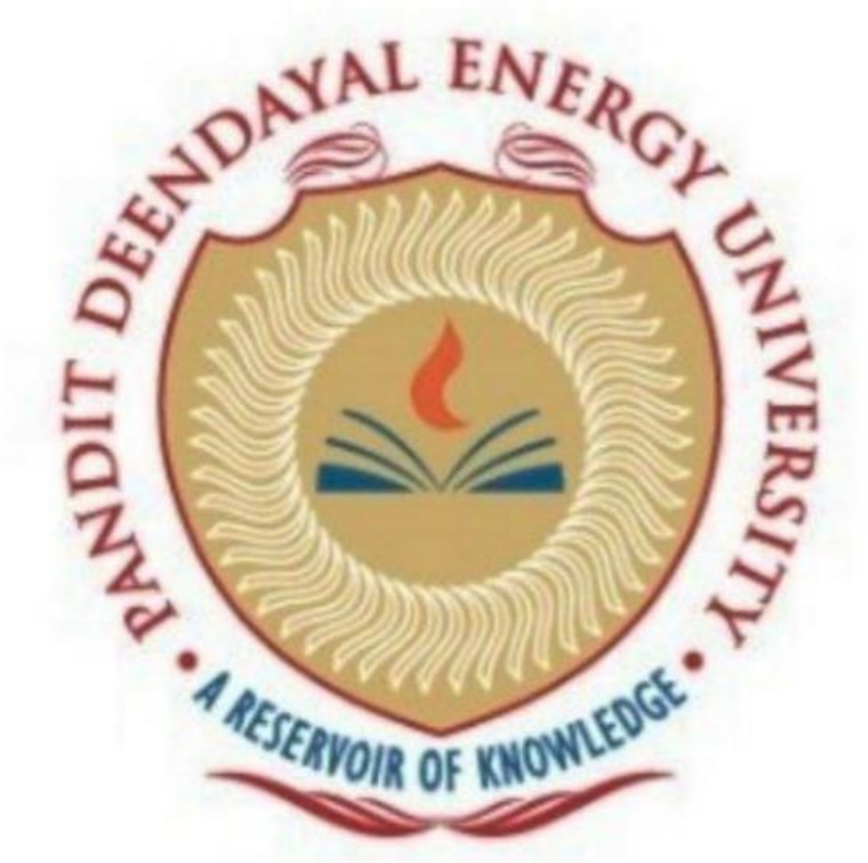
Name - Akshat Shah

Roll No - 21BCP322

Sem – III

Digital Electronics and Computer Organization Lab

Course Code – 20CP203P



Department of Computer Science Engineering
School of Technology,
Pandit Deendayal Energy University

Date: 17-08-2022

Assignment-2

Regarding how basic gates work up on interfacing with a hardware description language (HDL) Verilog.

Table of Contents

List of Figures

1. AND gate

Question: Design an AND gate with Logisim. Use the input and output labels according to given truth table.Write a Verilog code to implement AND gate. Write the corresponding Testbench code for the verification of your Verilog code.

Code:

testbench.sv

```
// Akshat Shah
// 21BCP322

// test bench for AND gate

module andgate_tb;  // creating a module which includes AND gate
  reg A,B;          // declaring A and B as reg to store iput values to be given
  wire Y;           // connecting to get output values
  andgate a1(A,B,Y);    // Instantiating by order as declared in module

  initial
    begin          // beginning execution
      A=0;B=0; #2; // assigning values as input
      $display("a=%b and b=%b",A,B);    // for display of values entered
      $display("y=%b",Y);
      A=0;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=0; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
    end // ending execution
  initial
    begin
      $dumpfile("dump.vcd"); // for storing information of waveform
      $dumpvars(1);          // specifying part of design to be stored as waveform
    end
endmodule       // enclosing the module
```

design.sv

```
module andgate (A,B,Y );
  input A, B;
  output Y;
  assign Y = A & B;
endmodule
```

Fig 1.1 - Testbench and design code for AND gate

Output:

```
[2022-08-20 12:33:44 EDT] iverilog '-Wall' design.sv testbench.sv  && unbuffer vvp a.out
VCD info: dumpfile dump.vcd opened for output.
a=0 and b=0
y=0
a=0 and b=1
y=0
a=1 and b=0
y=0
a=1 and b=1
y=1
Done
```

Fig 1.2 - Output as truth table of AND

Waveform:

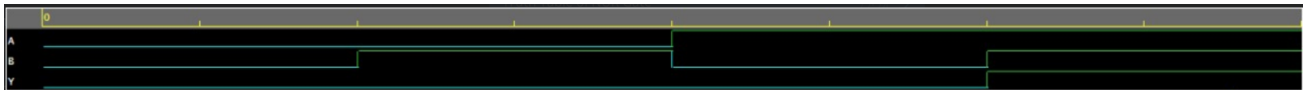


Fig 1.3 - Waveform of AND gate output

2. OR gate

Question: Design an OR gate with Logisim. Use the input and output labels according to given truth table. Write a Verilog code to implement OR gate. Write the corresponding Testbench code for the verification of your Verilog code.

Code:

testbench.sv

```
// Akshat Shah
// 21BCP322

// test bench for OR gate

module orgate_tb;   // creating a module which includes the gate
  reg A,B;          // declaring A and B as reg to store iput values to be given
  wire Y;           // connecting to get output values
  orgate a1(A,B,Y); // Instantiating by order as declared in module

  initial
    begin          // beginning execution
      A=0;B=0; #2; // assigning values as input
      $display("a=%b and b=%b",A,B);    // for display of values entered
      $display("y=%b",Y);
      A=0;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=0; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
    end // ending execution
  initial
    begin
      $dumpfile("dump.vcd"); // for storing information of waveform
      $dumpvars(1);          // specifying part of design to be stored as waveform
    end
endmodule       // enclosing the module
```

design.sv

```
module orgate (A,B,Y );
  input A, B;
  output Y;
  assign Y = A || B;
endmodule
```

Fig 2.1 - Testbench and design code for OR gate

Output:

```
[2022-08-20 12:57:55 EDT] iverilog '-Wall' design.sv testbench.sv  && unbuffer vvp a.out
VCD info: dumpfile dump.vcd opened for output.
a=0 and b=0
y=0
a=0 and b=1
y=1
a=1 and b=0
y=1
a=1 and b=1
y=1
Finding VCD file...
./dump.vcd
[2022-08-20 12:57:55 EDT] Opening EPWave...
Done
```

Fig 2.2 - Output as truth table of OR gate

Waveform:

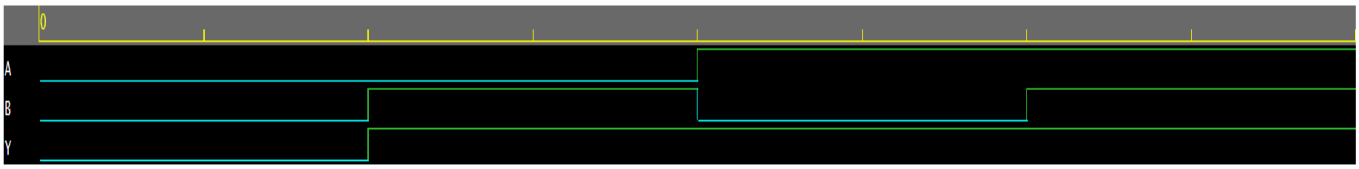


Fig 2.3 - Waveform of OR gate output

3. XOR gate

Question: Design an XOR gate with Logisim. Use the input and output labels according to given truth table. Write a Verilog code to implement XOR gate. Write the corresponding Testbench code for the verification of your Verilog code.

Code:

testbench.sv

```
// Akshat Shah
// 21BCP322

// test bench for XOR gate

module xorgate_tb;  // creating a module which includes the gate
  reg A,B;          // declaring A and B as reg to store iput values to be given
  wire Y;           // connecting to get output values
  xorgate a1(A,B,Y);    // Instantiating by order as declared in module

  initial
    begin          // beginning execution
      A=0;B=0; #2; // assigning values as input
      $display("a=%b and b=%b",A,B);    // for display of values entered
      $display("y=%b",Y);
      A=0;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=0; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
    end // ending execution
  initial
    begin
      $dumpfile("dump.vcd"); // for storing information of waveform
      $dumpvars(1);          // specifying part of design to be stored as waveform
    end
endmodule       // enclosing the module
```

design.sv

```
module xorgate (A,B,Y );
  input A, B;
  output Y;
  assign Y = ~A&&B || A&&~B;
endmodule
```

Fig 3.1 - Testbench and design code for XOR gate

Output:

```
[2022-08-20 13:13:03 EDT] iverilog '-Wall' design.sv testbench.sv  && unbuffer vvp a.out
VCD info: dumpfile dump.vcd opened for output.
a=0 and b=0
y=0
a=0 and b=1
y=1
a=1 and b=0
y=1
a=1 and b=1
y=0
Finding VCD file...
./dump.vcd
[2022-08-20 13:13:04 EDT] Opening EPWave...
Done
```

Fig 3.2 - Output as truth table of XOR gate

Waveform:

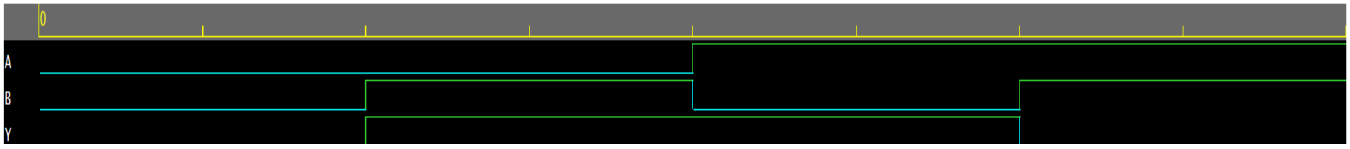


Fig 3.3- Waveform of XOR gate output

4. NOR gate

Question: Design a NOR gate with Logisim. Use the input and output labels according to given truth table.Write a Verilog code to implement NOR gate. Write the corresponding Testbench code for the verification of your Verilog code.

Code:

testbench.sv

```
// Akshat Shah
// 21BCP322

// test bench for NOR gate

module norgate_tb;  // creating a module which includes the gate
  reg A,B;          // declaring A and B as reg to store iput values to be given
  wire Y;           // connecting to get output values
  norgate a1(A,B,Y);    // Instantiating by order as declared in module

  initial
    begin          // beginning execution
      A=0;B=0; #2; // assigning values as input
      $display("a=%b and b=%b",A,B);    // for display of values entered
      $display("y=%b",Y);
      A=0;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=0; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
    end // ending execution
  initial
    begin
      $dumpfile("dump.vcd"); // for storing information of waveform
      $dumpvars(1);          // specifying part of design to be stored as waveform
    end
endmodule       // enclosing the module
```

design.sv

```
module norgate (A,B,Y );
  input A, B;
  output Y;
  assign Y = ~(A || B);
endmodule
```

Fig 4.1 - Testbench and design code for NOR gate

Output:

```
[2022-08-20 13:21:54 EDT] iverilog '-Wall' design.sv testbench.sv  && unbuffer vvp a.out
VCD info: dumpfile dump.vcd opened for output.
a=0 and b=0
y=1
a=0 and b=1
y=0
a=1 and b=0
y=0
a=1 and b=1
y=0
Finding VCD file...
./dump.vcd
[2022-08-20 13:21:54 EDT] Opening EPWave...
Done
```

Fig 4.2 - Output as truth table of NOR gate

Waveform:

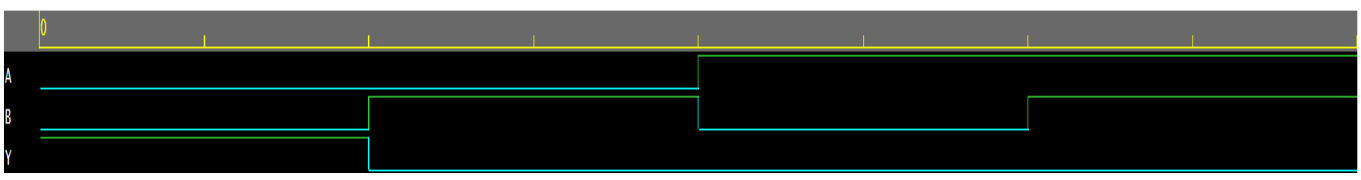


Fig 4.3 - Waveform of NOR gate output

5. NAND gate

Question: Design a NAND gate with Logisim. Use the input and output labels according to given truth table. Write a Verilog code to implement NAND gate. Write the corresponding Testbench code for the verification of your Verilog code.

Code:

testbench.sv

```
// Akshat Shah
// 21BCP322

// test bench for NAND gate

module nandgate_tb; // creating a module which includes the gate
  reg A,B;          // declaring A and B as reg to store iput values to be given
  wire Y;           // connecting to get output values
  nandgate a1(A,B,Y);   // Instantiating by order as declared in module

  initial
    begin          // beginning execution
      A=0;B=0; #2; // assigning values as input
      $display("a=%b and b=%b",A,B);    // for display of values entered
      $display("y=%b",Y);
      A=0;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=0; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
    end // ending execution
  initial
    begin
      $dumpfile("dump.vcd"); // for storing information of waveform
      $dumpvars(1);          // specifying part of design to be stored as waveform
    end
endmodule       // enclosing the module
```

design.sv

```
module nandgate (A,B,Y );
  input A, B;
  output Y;
  assign Y = ~(A & B);
endmodule
```

Fig 5.1 - Test bench and design code for NAND gate

Output:

```
[2022-08-20 13:26:35 EDT] iverilog '-Wall' design.sv testbench.sv  && unbuffer vvp a.out
VCD info: dumpfile dump.vcd opened for output.
a=0 and b=0
y=1
a=0 and b=1
y=1
a=1 and b=0
y=1
a=1 and b=1
y=0
Finding VCD file...
./dump.vcd
[2022-08-20 13:26:35 EDT] Opening EPWave...
Done
```

Fig 5.2 - Output as truth table of NAND gate

Waveform:

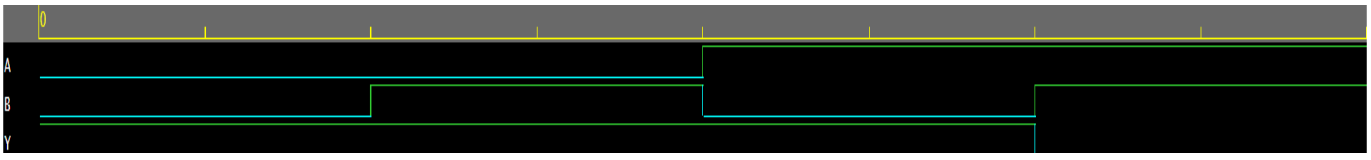


Fig 5.3- Waveform of NAND gate output

6. XNOR gate

Question: Design an XNOR gate with Logisim. Use the input and output labels according to given truth table.Write a Verilog code to implement XNOR gate. Write the corresponding Testbench code for the verification of your Verilog code.

Code:

testbench.sv

```
// Akshat Shah
// 21BCP322

// test bench for XNOR gate

module xnorgate_tb; // creating a module which includes the gate
  reg A,B;          // declaring A and B as reg to store iput values to be given
  wire Y;           // connecting to get output values
  xnorgate a1(A,B,Y);   // Instantiating by order as declared in module

  initial
    begin          // beginning execution
      A=0;B=0; #2; // assigning values as input
      $display("a=%b and b=%b",A,B);    // for display of values entered
      $display("y=%b",Y);
      A=0;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=0; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
      A=1;B=1; #2;
      $display("a=%b and b=%b",A,B);
      $display("y=%b",Y);
    end // ending execution
  initial
    begin
      $dumpfile("dump.vcd"); // for storing information of waveform
      $dumpvars(1);          // specifying part of design to be stored as waveform
    end
endmodule       // enclosing the module
```

SV/Verilog Testbench

design.sv

```
module xnorgate (A,B,Y );
  input A, B;
  output Y;
  assign Y = ~ (~A&&B || A&&~B);
endmodule
```

Fig 6.1 - Testbench and design code for XNOR gate

Output:

```
[2022-08-20 13:33:57 EDT] iverilog '-Wall' design.sv testbench.sv  && unbuffer vvp a.out
VCD info: dumpfile dump.vcd opened for output.
a=0 and b=0
y=1
a=0 and b=1
y=0
a=1 and b=0
y=0
a=1 and b=1
y=1
Finding VCD file...
./dump.vcd
[2022-08-20 13:33:57 EDT] Opening EPWave...
Done
```

Fig 6.2 - Output as truth table of XNOR gate

Waveform:

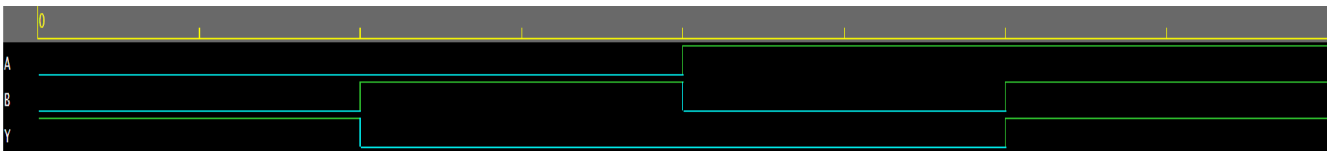


Fig 6.3 - Waveform of XNOR gate output