Experiment 1

January 5, 2025

Group 3
A Lekhasree - EE23B039
Durga B - EE23B027
K Chandrika - EE23B033

1 Aim

Simulation of half adder, full adder, and ripple carry adder in Xilinx Vivado and realization on FPGA board.

2 Hardware and software used:

Hardware - EDGE Artix7 FPGA Board
Software - Xilinx Vivado

3 Implementation

3.1 Half Adder

Verilog Code:

```
//circuit_half_adder

module half_adder(I1, I2, S, C);
 input I1, I2;
 output S, C;

 xor(S, I1, I2);
 and(C, I1, I2);

endmodule
```

Test-bench for simulation:

```
//testbench_half_adder

‘timescale 1ns/100ps

module half_adder_test;
 wire S, C;
 reg I1, I2;

 half_adder DUT(I1, I2, S, C);

 initial begin
  //$dumpfile("half_add.vcd");
  //$dumpvars(0, half_adder_test);
  I1 = 0;
  I2 = 0;
  #10
  I1 = 1;
  #10
  I2 = 1;
  #10
  I1 = 0;
  #10
  I2 = 0;
  end
endmodule
```

Constraints file:

```
# Switches
set_property -dict { PACKAGE_PIN L5    IOSTANDARD LVCMOS33 } [get_ports { I1 }];#LSB
set_property -dict { PACKAGE_PIN L4    IOSTANDARD LVCMOS33 } [get_ports { I2 }];

# LEDs
set_property -dict { PACKAGE_PIN J3    IOSTANDARD LVCMOS33 } [get_ports { S }];#LSB
set_property -dict { PACKAGE_PIN H3    IOSTANDARD LVCMOS33 } [get_ports { C }];
```

3.2 Full Adder

Verilog Code:

```
//circuit full_adder

module full_adder(s, cout, a, b, cin);
```

```
 input a;
 input b;
 input cin;
 output cout;
 output s;
 wire s1, c1, c2;

 xor(s1, a, b);
 and(c1, s1, cin);
 and(c2, a, b);
 xor(s, cin, s1);
 or(cout, c1, c2);

endmodule
```

Test-bench for simulation:

```
//testbench full adder

module full_adder_test;
 wire s, cout;
 reg a, b, cin;

 full_adder DUT(s, cout, a, b, cin);

 initial begin
  //$dumpfile("full_add.vcd");
  //$dumpvars(0, full_adder_test);
  a = 0;
  b = 0;
  cin = 0;
  #10
  a = 1;
  #10
  b = 1;
  #10
  cin = 1;
  #10
  a = 0;
  end
endmodule
```

Constraints file:

```
# Switches
```

```
set_property -dict { PACKAGE_PIN L5    IOSTANDARD LVCMOS33 } [get_ports { a }];#LSB
set_property -dict { PACKAGE_PIN L4    IOSTANDARD LVCMOS33 } [get_ports { b }];
set_property -dict { PACKAGE_PIN M4    IOSTANDARD LVCMOS33 } [get_ports { cin }];

# LEDs
set_property -dict { PACKAGE_PIN J3    IOSTANDARD LVCMOS33 } [get_ports { s }];#LSB
set_property -dict { PACKAGE_PIN H3    IOSTANDARD LVCMOS33 } [get_ports { cout }];
```

3.3 Ripple Carry Adder

Verilog Code:

```
//circuit 4-bit ripple adder

module FFA(A, B, sum, cout);

  input [3:0] A, B;
  output reg [3:0] sum;
  output reg cout;
  reg cin;
  integer i;

  always @* begin
    cin = 0;
    for (i = 0; i < 4; i = i + 1) begin
      {cout, sum[i]} = A[i] + B[i] + cin;
      cin = cout;
    end
  end
endmodule
```

Test-bench for simulation:

```
//testbench for 4-bit full adder

‘timescale 1ns/100ps

module FFA_test;
 wire[3:0] sum;
 wire cout;
 reg[3:0] A, B;
 FFA DUT(A, B, sum, cout);

 initial begin
  $dumpfile("Four_full_add.vcd");
  $dumpvars(0, FFA_test);
```

```
  A = 4'b1001; B = 4'b1001;
  #10 $finish;
  end
endmodule
```

Constraints file:

```
# Switches
set_property -dict { PACKAGE_PIN L5    IOSTANDARD LVCMOS33 } [get_ports { A[0] }];#LSB
set_property -dict { PACKAGE_PIN L4    IOSTANDARD LVCMOS33 } [get_ports { A[1] }];
set_property -dict { PACKAGE_PIN M4    IOSTANDARD LVCMOS33 } [get_ports { A[2] }];
set_property -dict { PACKAGE_PIN M2    IOSTANDARD LVCMOS33 } [get_ports { A[3] }];
set_property -dict { PACKAGE_PIN M1    IOSTANDARD LVCMOS33 } [get_ports { B[0] }];
set_property -dict { PACKAGE_PIN N3    IOSTANDARD LVCMOS33 } [get_ports { B[1] }];
set_property -dict { PACKAGE_PIN N2    IOSTANDARD LVCMOS33 } [get_ports { B[2] }];
set_property -dict { PACKAGE_PIN N1    IOSTANDARD LVCMOS33 } [get_ports { B[3] }];

# LEDs
set_property -dict { PACKAGE_PIN J3    IOSTANDARD LVCMOS33 } [get_ports { sum[0] }];#LSB
set_property -dict { PACKAGE_PIN H3    IOSTANDARD LVCMOS33 } [get_ports { sum[1] }];
set_property -dict { PACKAGE_PIN J1    IOSTANDARD LVCMOS33 } [get_ports { sum[2] }];
set_property -dict { PACKAGE_PIN K1    IOSTANDARD LVCMOS33 } [get_ports { sum[3] }];
set_property -dict { PACKAGE_PIN L3    IOSTANDARD LVCMOS33 } [get_ports { cout }];
```

4 Observations

1. All the above three worked as expected.
2. On changing the input switches, corresponding changes were seen in the output LEDs.

5 Errors and Debugging

1. **Error:** Bit stream generation failed. We imported the constraints file of the 2021 version, which didn't get supported in the 2022 version of Xilinx Vivado.
2. **Debug:** Changed the constraints file to the 2022 version.