Indian Institute of Technology Kharagpur

AUTUMN Semester, 2020
COMPUTER SCIENCE AND ENGINEERING

Computer Organization Laboratory

Assignment-4: Verilog Design of Sequential Circuits for Simple Arithmetic Operations

Full Marks: 20

Time allowed: 6 hours

INSTRUCTIONS: Make one submission per group in the form of a single zipped folder containing your Verilog source code files(s) and Verilog testbench(es). Name your submitted zipped folder as `Assgn_4_Grp_<Group_no>.zip` and (e.g. `Assgn_4__Grp_25.zip`). Inside each submitted source and testbench files, there should be a clear header describing the assignment no., problem no., semester, group no., and names of group members. Liberally comment your code to improve its comprehensibility.

1. [**Two's Complement Converter FSM**] Design (using Verilog), simulate (using an appropriate Verilog testbench), and implement (on a FPGA platform supported by your CAD software tool), a simple finite state machine (FSM) that in every clock cycle reads an input bit, and outputs a bit such that the bitstring output till that point is the two's complement of the binary number read till that point, including the most recently read bit (the number is considered to be input from the LSB side). The FSM has one input control signals which resets it to the initial state. Come up with an appropriate interface for your circuit. [Hint: consider a Mealy machine.]   (4 marks)

2. [**Multiple-of-three Detector FSM**] Design (using Verilog), simulate (using an appropriate Verilog testbench), and implement (on a FPGA platform supported by your CAD software tool), a simple finite state machine (FSM) that in every clock cycle reads an input bit, and outputs a bit which indicates whether the binary number read till that point, including the most recently read bit (the number is considered to be input from the LSB side), is divisible by three. The input number is to be considered an unsigned integer. The FSM has one input control signals which resets it to the initial state. Come up with an appropriate interface for your circuit. [Hint: again, consider a Mealy machine.]   (6 marks)

3. [Sequential GCD Calculator] Consider the following C function to calculate the GCD of two positive integer arguments:

```
/* a, b are positive integers, with a >= b */
int gcd (int a, int b) {
   int temp;
   while (1) {
    while ((temp = a - b) > 0) a = temp;
    if (!temp) return b;
    temp = a; a = b; b = temp;
   }
}
```

Design (using Verilog), simulate (using an appropriate Verilog testbench) and implement (on a FPGA platform supported by your CAD software tool) a sequential circuit to execute this scheme and generate the GCD as output. Assume each input operand is a positive 8-bit integer. Come up with an appropriate interface for your circuit. Design the data path and the control path separately, with the control path circuitry generating control signals based on status signals received from the data path circuitry. Design the data path in a *bottom-up* manner, structurally and hierarchically. In the data path, try to reuse adder/subtractor circuits you have designed in a previous assignment.   (10 marks)