Computer Aided Design Course Projects

Project #1

Title: Floating-point Arithmetic Unit (First Part)

Definition: Floating point operations are widely used in large set of scientific and signal processing computation. IEEE has set a standard for these operations numbered IEEE-754. In this project, a floating point arithmetic operation unit need to be developed and tested according to IEEE specifications. This unit is a combination of a control logic and data path. The operations need to be done on single and double precision floating point numbers. The concept of floating point numbers need to be understood well and implementation of the arithmetic unit should undergo exhaustive tests according to standards.

Specifications: According to IEEE-754-2008 for single (SP) and double precision (DP) floating point numbers

Blocks to develop (in optimized behavioral/dataflow Verilog):

1. MAC (SP: 1cycle, DP: 2 cycles) + TB
2. MUL (SP: 1cycle, DP: 2 cycles) + TB
3. NEG, ABS, COPY (1 cycle) + TB
4. A central controller for selection between operations + TB

Work:

1. Writing a golden model in C or Python or Java; whichever is easier for team
2. Unit test
3. Integration test
    a. Integration of all blocks into one system
    b. An overall testbench for the whole system
    c. Verification against golden model
4. Synthesis using Xilinx Vivado Design Suite

Deliverables:

1. Verilog RTL at behavioral/data flow level for each block
2. Testbench in Verilog for every block according to the specifications. The standalone testbench should cover all possible cases.
3. All source codes and test-benches should be put in zip file.
4. Simulation results in ModelSim/Vivado simulator
5. Performing synthesis on FPGA and generating meaningful reports
6. Performing Post-synthesis timing simulation (Verilog Netlist + SDF); *has an extra mark*
7. Technical report in Persian/English

Project #2

Title: Floating-point Arithmetic System (Second Part)

Definition: Floating point operations are widely used in large set of scientific and signal processing computation. IEEE has set a standard for these operations numbered IEEE-754. In this project, a floating point arithmetic operation unit need to be developed and tested according to IEEE specifications. This unit is a combination of a control logic and data path. The operations need to be done on single and double precision floating point numbers. The concept of floating point numbers need to be understood well and implementation of the arithmetic unit should undergo exhaustive tests according to standards.

Specifications: According to IEEE-754-2008 for single (SP) and double precision (DP) floating point numbers

Blocks to develop (in optimized behavioral/dataflow Verilog):

1. DIV (SP: 14 cycles, DP: 28 cycles) + TB
2. SQRT (SP: 14 cycles, DP: 28 cycles) + TB
3. A central controller for selection between operations + TB

Work:

1. Writing a golden model in C or Python or Java; whichever is easier for team
2. Unit test
3. Integration test
    a. Integration of all blocks into one system
    b. An overall testbench for the whole system
    c. Verification against golden model
4. Synthesis using Xilinx Vivado Design Suite

Deliverables:

1. Verilog RTL at behavioral/data flow level for each block
2. Testbench in Verilog for every block according to specifications. The standalone testbench should cover all possible cases including invalid operations such as divide by zero, overflow, underflow, and inexact.
3. Simulation results in ModelSim/Vivado simulator
4. Performing synthesis on FPGA and generating meaningful reports
5. Performing Post-synthesis timing simulation (Verilog Netlist + SDF); *has an extra mark*
6. Technical report in Persian/English

Project #3

Title: A Configurable Asynchronous FIFO

Definition: First-In First-Out (FIFO) memory structures have been the dominant mechanism to control the flow of data between a source and a destination. The function of FIFO runs around memory element and control circuitry. In hardware it is either an array of flops or Read/Write memory where data is written to memory in one clock domain and on request, data is read from memory in another clock domain following the same order as written. FIFO is said to be SYNCHRONOUS if read and write operations are performed using same clock signal and FIFO is said to be ASYNCHRONOUS if read and write operations are performed using different clock signals In this project, we are going to develop a FIFO including the controller and a dual port memory element, with independent clocks for read and write operations.

Specifications: No specific standard exists. Available resources on the web can be used.

Blocks to develop (in behavioral Verilog):

1. A FIFO controller
2. A configurable dual-port memory (width and depth) with independent read and write clocks
3. A complete testbench

Work:

1. Writing a golden model in C or Python or Java; whichever is easier for team
2. Unit test
3. Integration test
    a. Integration of all blocks into one system
    b. An overall testbench for the whole system
    c. Verification against golden model
4. Synthesis using Xilinx Vivado Design Suite

Deliverables:

1. Verilog RTL at behavioral/data flow level for each block
2. Testbench in Verilog for every block according to specifications.
3. Simulation results in ModelSim/Vivado simulator
4. Performing synthesis on FPGA and generating meaningful reports
5. Performing Post-synthesis timing simulation (Verilog Netlist + SDF); *has an extra mark*
6. Technical report in Persian/English