| SDU | Digital Design and Signal Processing - DDS Assignment #2 | IC | 26.01.24 |
|---|---|---|---|

The purpose of this assignment is to create models of simple sequential circuits in the VHDL language. You will be working with flipflops, counters, registers, a state machine and the corresponding timing relations.

All designs will be implemented in the Xilinx Vivado development tools with the ZEDBOARD as the target.

**Question 2.1:**
Write the VHDL code for a simple positive edge triggered D- flipflop. Use std_logic signals. Document the functionality of the circuit with relevant simulations. Describe the timing relations for the flipflop.

Add a Reset and Preset functionality to the flipflop. Document the new functionality with relevant simulations. Decide if Reset/Preset is asynchronous or synchronous and explain the difference in the VHDL model, as well as the timing relations.

**Qustion 2.2:**
Write the VHDL code for an 8 bit up/down counter with Reset. Use std_logic types on all signals. Document the functionality of the circuit with relevant simulations and give a qualified guess of the counters expected maximum clock frequency.

**Question 2.3:**
Create a new VHDL project. Design a simple datapath for a microcontroller with the use of the ALU from assignment #1, and four 8 bit registers with **parallel load and parallel outputs**.

The registers are named A, B, C and D. The circuit must be designed, so that every register can be connected to any of the ALU's data inputs (this requires multiplexing on registers outputs) and any register can be loaded with the value from the ALU's output.

Furthermore: establish input and output possibilities, in the form of two 8 bit input ports and two 8 bits output ports, so data can be loaded into the registers and read out again. Define and implement relevant clock and control signals. Document the functionality and timing with simulations.

**Question 2.4:**
Now we need the control part. Define some very simple assembler instructions. Examples could be: "add two numbers from the input ports and save the result in register A", "subtract two numbers from the input ports and save the result in register B", "Put Register A to the output port A", etc.

Develop a state machine that can read the assembler instruction from a memory component, execute the instruction and continue by reading the next instruction…. The state machine must provide all relevant signals to control the data part developed in question 2.3.

Connect the data part and control part in a structural VHDL and simulate and document the final design. **Use the state machine guideline document (on ItsLearning) for further information.**

**NB!! In a design like this: Draw the block diagram and specify the circuit before writing the VHDL code – and use this in your documentation.**