Queen Mary University of London

School of Electronic Engineering and Computer Science

ECS527U: Digital System Design

Lab 1: Concurrent Statements in VHDL

Introduction

The aim of this set of labs is to introduce the concepts and the advantages of Computer Aided Design (CAD) to design, simulate and implement digital circuits on programmable components such as CPLDs (Complex Programmable Logic Devices) and FPGAs (Field Programmable Gate Arrays).

These labs introduce some basic CAD tools using the Xilinx Vivado Design Suite also includes a VHDL simulator that can be used to give you an insight into the use of programmable logic.

VHDL (Very High Speed Hardware Description Language) is used throughout our labs to allow you model your digital circuits/systems. As VHDL can be readily interpreted by software, you can use a computer to produce your hardware designs on programmable chips. This makes designing complex digital circuits much faster, cheaper and, above all, easier than using discrete physical logic chips (recall your experience from digital logic design laboratories). Basic circuit modeling techniques had been introduced in our lectures.

After completing this lab exercise, you should be able to:

- write concurrent statements in VHDL;
- model behaviour of a digital circuit using the VHDL language;
- implement and download these designs into FPGA;
- verify your design on the lab board.

Specification

You have to design a module that has four single bit inputs: A, B, C and D. The module implements four different Boolean functions at the output bus F. The following table lists the Boolean functions.

ID	Function
0	$F(0) = AB$
1	$F(1) = A + B$
2	$F(2) = A \oplus B = A\bar{B} + \bar{A}B$
3	$F(3) = majority(A, B, C, D)$

The 4-bit majority function is defined as: outputs 1 if there are two or more ones from the four inputs.

The module should be named `Lab1` in the file `Lab1.vhd`. You have to write the whole VHDL source file with both the entity and the architecture.

Restriction: only the following VHDL package(s) and feature(s) is/are allowed in this lab.

- `IEEE.std_logic_1164.all`
- Intermediate signals, logic operators, relational operators
- Signal assignment statements, conditional signal assignment statements (when-else, with-select)
- port map statements

Task 1: Module Design and VHDL Writing

- *Follow Lab 0 to familiarise yourself with Xilinx Vivado if you wish*
- Download the development files (`ECS527U Lab1 Development Files (Vivado Project)`) from QM-Plus
- Extract the archive to your working PC in the lab (e.g. H:)
- Open the project with Xilinx Vivado ver. **2017.03** (best compatibility with the NI DSDB lab board).
- **Create** a new file `Lab1.vhd` and write your VHDL codes (both entity and architecture) based on the specification.
  Note: Recommend to follow the convention to use the same name for your entity and the file (e.g. module `Lab1` in the file `Lab1.vhd`)
- Use the tools to check the syntax of your VHDL.

Task 2: Mapping board devices to module

Add the VHDL file completed in Task 1 to your project.

Then modify `NI_DSDB_demo_core` to declare the module `Lab1` as a component.

In `NI_DSDB_demo_core`, there are signals related to the on-board input and output devices, for instance,
`sw`: On-board switches
`led`: On-board green LEDs
`ss*`: On-board red 7-segment displays

Create an instance of the `Lab1` module and map the appropriate signals such that the following connections are made:

Devices	Ports
`sw(3)`	A
`sw(2)`	B
`sw(1)`	C
`sw(0)`	D
`led(3 downto 0)`	F

Lastly, modify the statements for the 7-segment displays and assign the last four digits of your student IDs (from left to right: ss3, ss2, ss1, ss0) (either member).

You should be able to see the expected file hierarchy as shown in the end of this specification.

If you need help, please approach one of our demonstrators.

You will also be provided a lab board during the lab session. Once you synthesise and implement the top module `NI_DSDB_top` and generate the bitstream following the Vivado design flow. You can download the design to the lab board:

- Connect USB cable to the lab board and the PC.
- Power on the lab board (there can be several switches)
- Choose "Open Hardware Manager" and auto-detect target board.
- Program the design with the generated bitstream.

Expected Hierarchy of VHDL Design Files

```
NI_DSDB_Top.vhd
       \__________(NI_DSDB_demo_core.vhd)
                            \__________________ [Lab1.vhd]

(xxx.vhd): files that you need to modify
[xxx.vhd]: files for you to create and write your code
```

Submission

After the demo, submit the following files to QMPlus:

1. VHDL files: `Lab1.vhd` and `NI_DSDB_demo_core.vhd`
   (note: all other VHDL source files are not supposed to be modified)
2. Half an A4 paper of writing (200-250 words) (in PDF) to conclude what you have learned from this lab exercise.

This lab exercise is formative but your attendance and submission is still **mandatory**. Your submission will be marked with feedbacks in order to prepare for the assessed lab test.

Deadline: **8th Feburary, 2022** Tuesday 12:00 noon

– End of Lab 1 –