Oblig 2

VHDL

IN3160 / IN4160

Version 2.0 2023-02-01

The goal of this exercise is to to get started designing using VHDL, and creating your first testbench. *With GHDL/Cocotb it is not a clear cut method to explicitly specify which architecture of the entity you want to use. With Questa the compile order would take of this but it is not the case with GHDL/Cocotb. The following is a suggested workaround that only compile the files needed for your chosen architecture. This is mosly relevant for task b.*

The makefile may be modified like the following:

```
# TOPLEVEL is the name of the toplevel module in your VHDL file
TOPLEVEL ?= decoder

# The architecture you want to use, default given.
# Override default: make ARCH=select
# Remember to make clean in between different architechture simulations.
ARCH ?= case

VHDL_SOURCES += $(PWD)/../src/$(TOPLEVEL)_ent.vhd
VHDL_SOURCES += $(PWD)/../src/$(TOPLEVEL)_$(ARCH).vhd

SIM_ARGS +=--wave=$(TOPLEVEL).ghw

# MODULE is the basename of the Python test file
MODULE = tb_$(TOPLEVEL)_$(ARCH)
```

This assumes a strict naming structure:

```
src
├── decoder_case.vhd
├── decoder_ent.vhd
└── decoder_select.vhd
test
├── makefile
├── tb_decoder_case.py
└── tb_decoder_select.py
```

decoder_ent.vhd only contains the entity declaration, no architecture.
decoder _case only contains the architecture declaration for the architecture using case (task a)
decoder _select only contains the architecture declaration for the architecture using select (task b)
Beware that you will need to make clean in between simulations of the different architectures: *make ARCH=case*
make clean
make ARCH=select

a)

In this exercise, you will design a 2-to-4 bit decoder. Use a **case** statement in a process to achieve this. The outputs on the decoder should be active low, see truth table below. The decoder shall be implemented on the test board. Use SW1 and SW2 as input, and LD1, LD2, LD3 and LD4 as output.

Test all the possible options for the switches in the simulator (GHDL). Create a python testbench and a makefile that runs the simulation and displays the waveform.

SW2	**SW1**	**LD4**	**LD3**	**LD2**	**LD1**
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	1

Approval:

VHDL design file, makefile for compilation and simulation, waveform file for architecture using case (E.g. .ghw) , constraints file (.XDC)

The exercise must be demonstrated to the laboratory supervisor for approval
(A video demonstrating that you program the board using your own design files may also be used when permitted by the lab supervisor).

b)

Create separate files for the entity and architecture. Create another architecture that implements the decoder in a concurrent statement using the **select** statement. Make exactly one deliberate change in the LED output, and make a comment for it in the VHDL architecture. Modify the .do file to compile the newest files and display the new waveform.

Approval:

VHDL design files, makefile, waveform file for architecture using select (E.g .ghw)