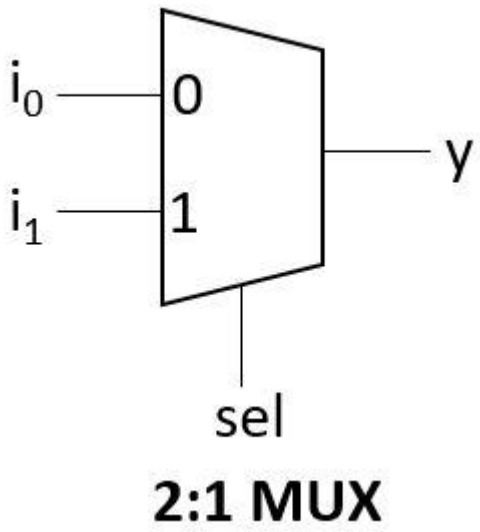
Lab 10: Testbench coding

Goal of this lab is to code a testbench for a simple 2:1 multiplexer module, which is available at `labs/lab10/src/mux.vhd`. The interface of multiplexer are

Port	Direction	Type
i0	IN	std_logic
i1	IN	std_logic
sel	IN	std_logic
y	OUT	std_logic



This is the truth table of the module

sel	y
0	i0
1	i1

Exercise 1. Basic Testbench

Go to the lab folder `~/labs/lab10/`, and open the testbench file `sim/tb_mux.vhd`

```
kate sim/tb_mux.vhd &
```

Design the testbench, following the instructions in the comments of the file. Once the testbench is ready, open the `run_sim.sh` file, and code the script as shown in the lecture.

You can use either XSIM or GHDL, as you prefer.

Once you are done, run the simulation to test your testbench.

```
./run_sim.sh
```

Exercise 2. Self-Checking

Improve the testbench, by adding self-checking and pass/failing functionalities.

- Check the `y` output of the multiplexer
- Report errors when the expected behaviour is incorrect.
- Issue a final pass / fail message based on the results of the test.
    - Tip: You need an extra check in addition to the integrated `assert` function to keep track of errors.

Exercise 3. Write a Procedure

Improve the testbench, by adding a `write` procedure that sends the stimulus, check the results and update the error counts.