**Logbook**

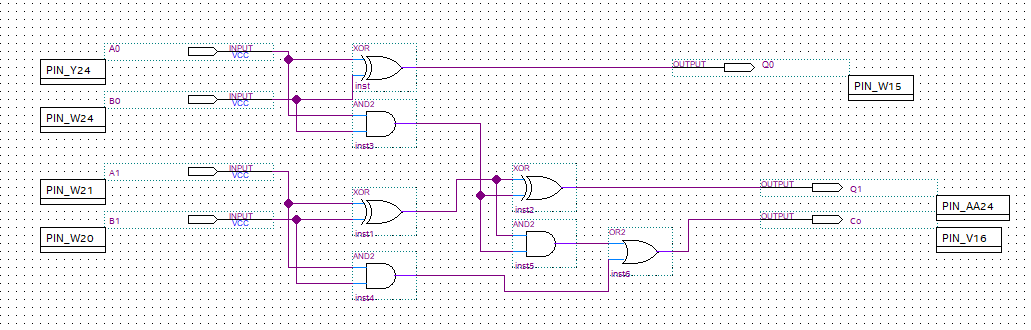
**Embedded systems 2018**

**Miroslav Gechev – 429556 – ETI3IA**

**Lesson 1 – Introduction and test benches**

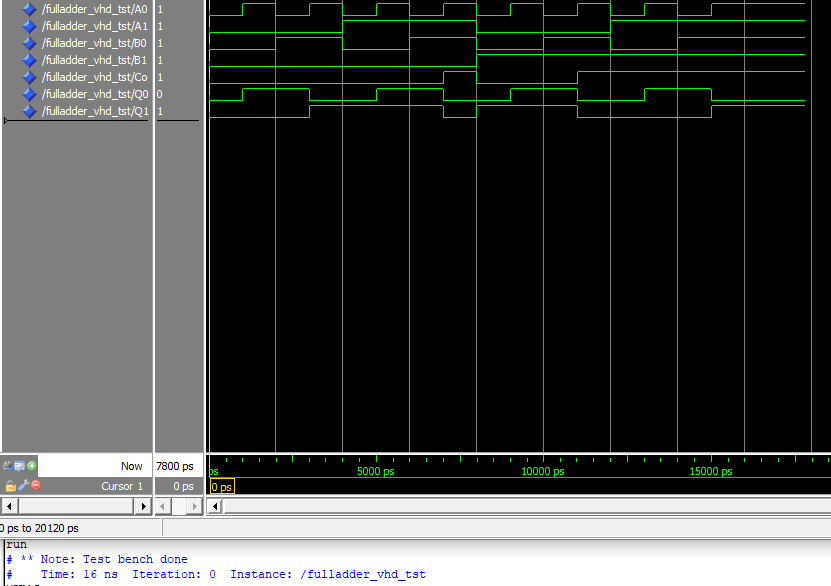
For the first lesson of the class Embedded systems 2018/2019 we had an introduction and we recalled on some basic Digital knowledge, how to work with Quartus Prime (probably the hardest part), some VHDL knowledge.

The first assignment was to create a 2bit full adder. I decided to build the full adder with blocks, since it is easier to understand and see. For started I created a simple 1 bit full adder. Just to get used to working with this software again. Once I got everything running I decided to proceed with building the 2 bit full adder.



I ended up with the solution above. Firstly I wanted to test out this solution on the DE10-nano board. I assigned the pins for the inputs and outputs, uploaded the code to the board and everything was working as expected – 2 switches for A and 2 for B, 3 outputs.

Next, I had to simulate my results in the software Modelsim. Now that was the most time consuming part. The simulation itself is easy to do, once you get to set up the software settings correctly. That took me a lot of time. But once I managed to set everything correctly, the making of the testbench and running it was very easy and quick.



The results of the simulation were the same with the ones I saw on the DE10-nano board.

\*feeback to the course teacher – maybe make a guide on blackboard how to setup all the software settings correctly (incl. modelsim, quartus)

### Lesson 2 - QSys & Nios II

### For the second lesson of Embedded System course, We were introduced to Qsys and Nios.

### We were provided with tutorial for the purpose. This tutorial consists of multiple task.

### First we had to create embedded system with Qsys platform designer. We had to add number of components to our system. In the end, the system looks like this:

### 

From the picture above, we can see that we use 2 buttons, all 4 switches and all 8 leds of the DE10 board.

After we are done with the Qsys, we proceed with the Eclipse software of Quartus. 1st board support package is created. After everything was set up with the BSP, we proceeded with creating the application that is going to be uploaded and run on the DE10-Nano. The 1st application we had to do is to do simple counter.

To upload the application to the board we used the Quartus command line tool. After testing the counter on the board, we proceeded with adding the switches into the system ( We use those with the 2nd application – 2 bit full adder.

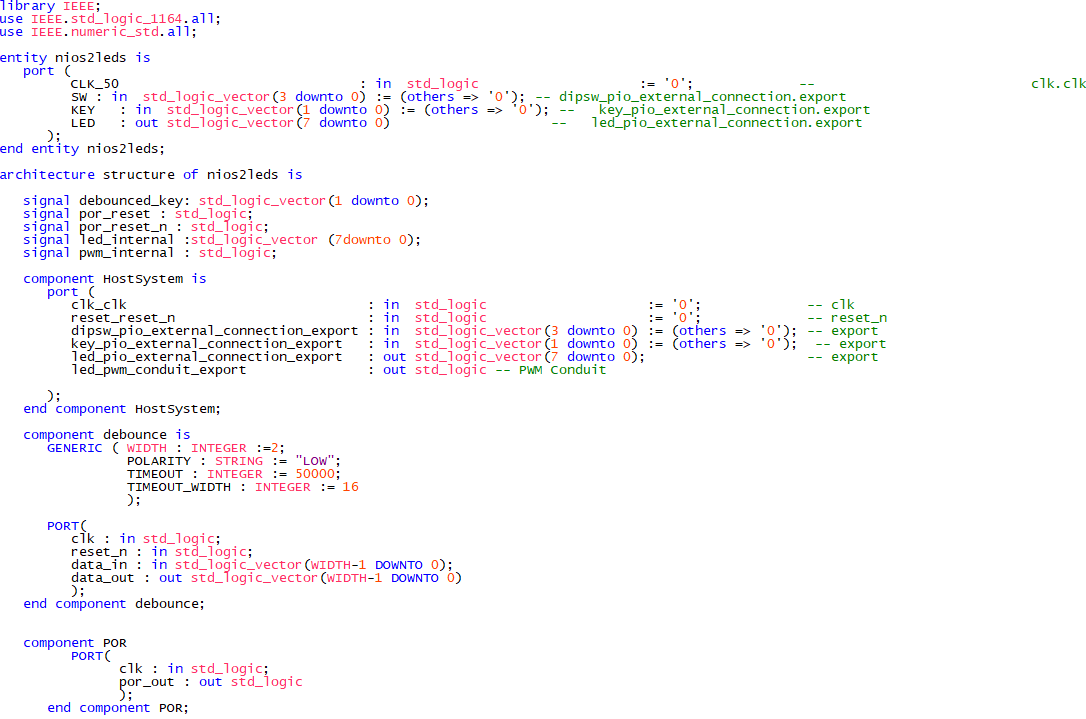
After that we proceeded with adding the buttons to the system. For those we had to take in mind the Debouncing and interrupts .

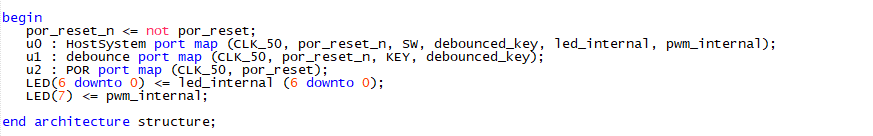
The next assignment from the tutorial was to add PWM component to the system.

In the last assignment we had to combine the PWM and Interrupts in one application.

Long story short – The tutorial was pretty straight forward – Just doing every step from it, was easy enough to do and understand. I did not have troubles doing everything showed in it and to be honest, I think it was much easier than the first Assignment we had. I would recommend to stick with that kind of tutorials – Easy to understand and do ourselves.

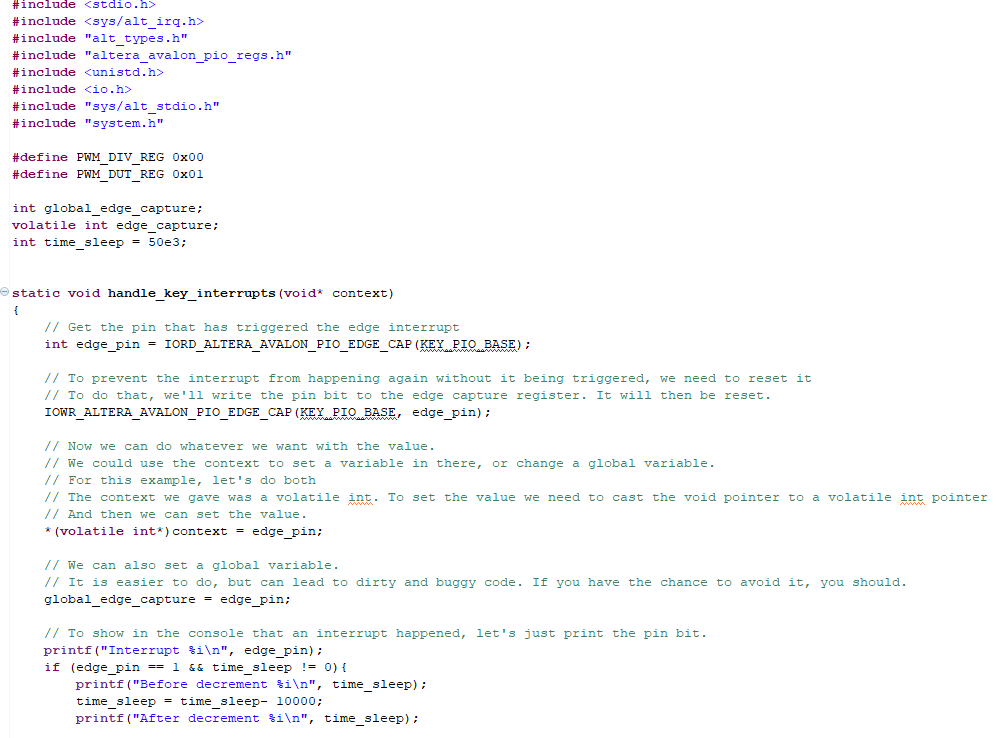
In the picture below, the VHDL code for our system is shown. (That is the last assignment)

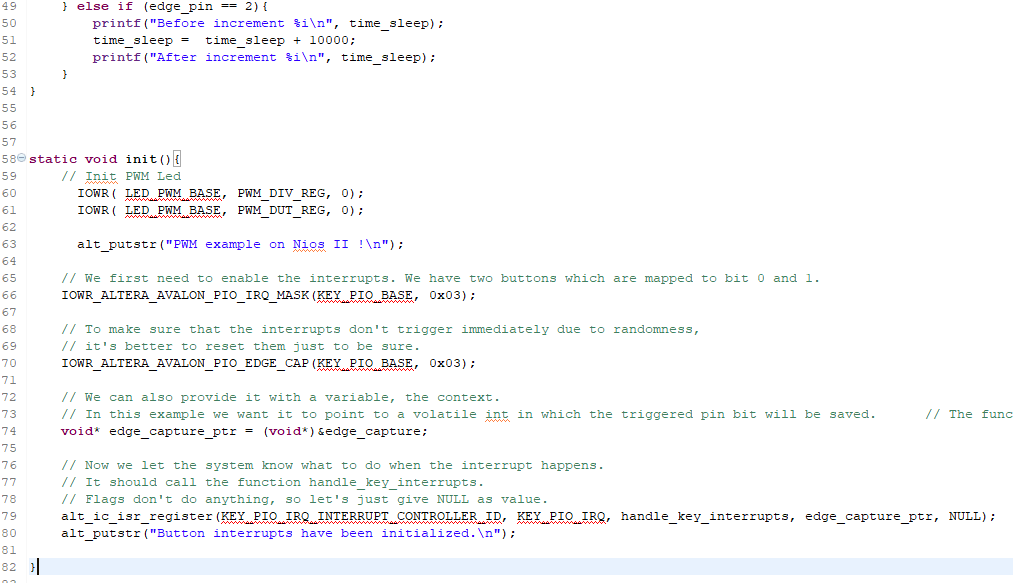


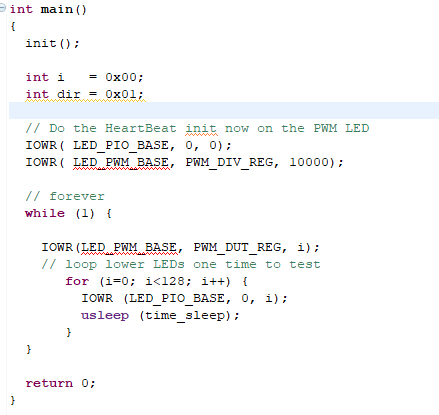


The C code provided below is for the application that combines PWM and Interrupts.

What we see is a function that handles the button interrupts. Depending on which button it is it either slows or speeds up the counter. The 2nd function is init function. We init PWM leds, then the buttons. In the main 1st we call the init so we can use the buttons and leds. We check if the leds are working, for the buttons we can see if they were initialized in the console, from the print in the init. Then we go into infinite loop, which counts up to 255, with speed depending on the buttons pressed. 1 speeds up the counter, 1 slows it down.







### Lesson 3 - Getting started with Linux on the DE10

For the 3rd assignment for the Embedded Systems course, we had to run Linux operating system on the DE10-Nano board. For the purpose, we were provided with a tutorial how to setup everything. The assignment itself weren’t that hard – if the tutorial is followed step by step, it is very easy to complete the assignment. Of course, some basic knowledge how to work with Linux was required, but even if you are not experienced with this, the tutorial gives the basic commands needed for the tutorial. One tricky thing that needs getting used to is working with Vim. Inexperienced people usually find it very confusing when first working with Vim.

We were given a bash script, with which we can control the Leds on the board. It was very easy to upload the code to the Nano.

Overall, I would say that the tutorial is very good written and explains everything that needs to be done to run the script on the DE10. I, personally didn’t have any trouble when doing the tutorial – only 1, but that was mistyped error, which was easy to spot so I didn’t spend too much time on that.

**Self-reflection**

The course embedded system helped me deepen my knowledge in FPGA systems. It was very hard experience, though. I wish we had more information to work with from the beginning – most of the information I was finding on the web was either outdated, or not suitable with the board I was using, but in the end I always managed to do what I was supposed to. One thing I am sad about is not being able to finish the HPS assignment. We had no information about it at all and I couldn’t finish it alone on time, so I had to begin with the final assignment.

**Reccomendations**

I would recommend for the course to have a presentation explaining at least the basic stuff. Setting up the software was very annoying to begin with, so a little tutorial on that would be very useful. I also think its better if we had a theory lesson and then an assignment to do, the tutorials were nice, but to be honest they didn’t teach me a lot.