Date: 07.05.2021
Author: Sławomir Batruch

I declare that this piece of work that is a basis for the grading of edc1 laboratory 3 task was performed on my own without any unallowed help (indirect or direct) from other students.

Sławomir Batruch; ID 303827

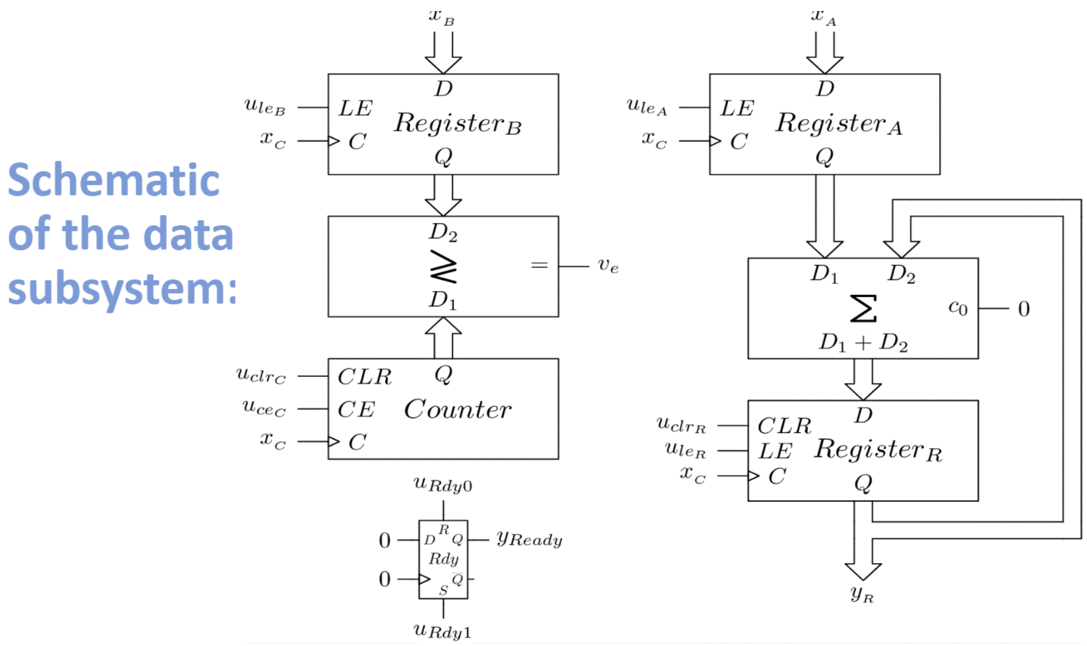
EDC1 – Lab4

For this laboratory the task was to design a schematic in Xilinx based on a schematic given in a .pdf lab introduction file.
Then in the testbench, do proper instructions so that the designed schematic follows the given order of instructions.
The general way the circuit works is that it **performs a multiplication of two unsigned binary numbers.**

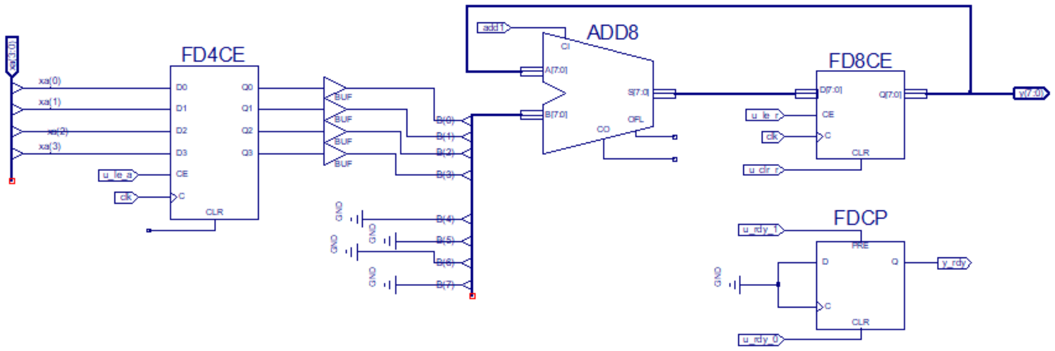
Task A and C

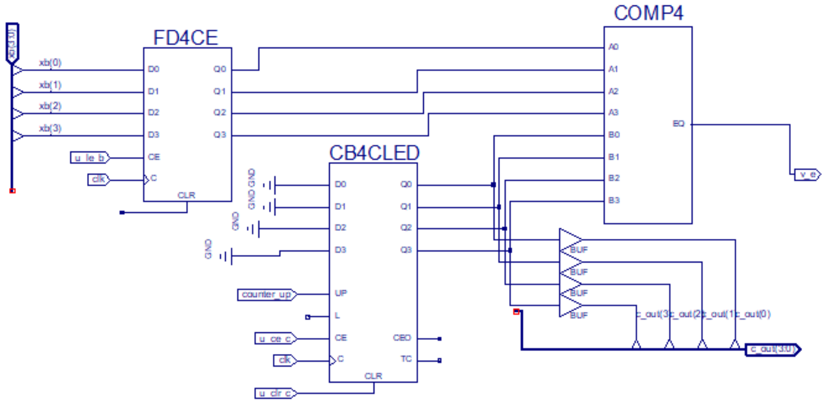
For this task, the schematic was a binary number multiplier that adds a given binary number (xa), (xb) times. For example 3 * 7 would mean that 3 would be added 7 times, that would result in 21 – a correct result.

First, I did make a schematic in Xilinx, based on the schematic from the PDF:


Schematic of the data subsystem:

In Xilinx, it looked like this:

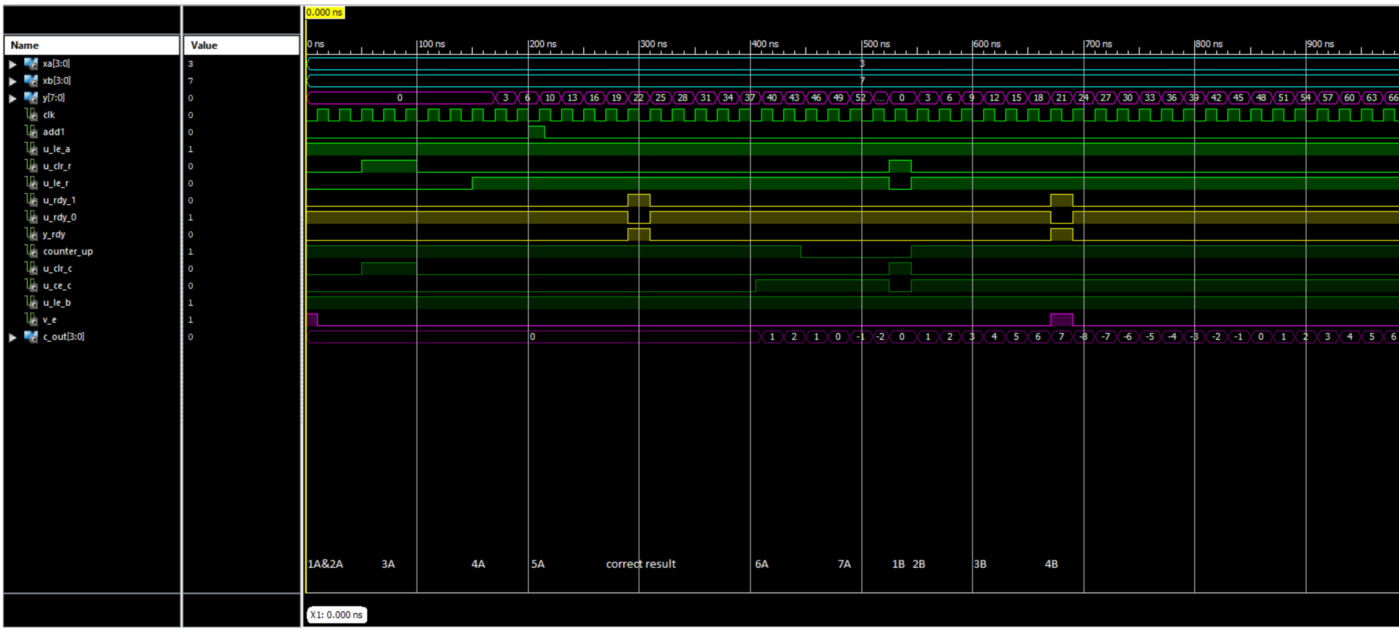




Then I wrote a testbench based on these instructions. No text snippets are pasted in the report, as the testbench is way more complex compared to testbenches from older labs:

1.Take two rightmost digits greater than 2 numbers from your student ID.
2.Load these numbers into registers A and B.
3.Clear register R and counter C.
4.Load register R.
5.Set c0=1 and load register R.
6.Increment the counter 2 times.
7.Decrement the counter 4 times.

Then after checking the testbench and running simulation based on it, and then changing radix, colours, order etc. of the signals, I obtained the following simulation output:



Name	Value
xa[3:0]	3
xb[3:0]	7
y[7:0]	0
clk	0
add1	0
u_le_a	1
u_clr_r	0
u_le_r	0
u_rdy_1	0
u_rdy_0	1
y_rdy	0
counter_up	1
u_clr_c	0
u_ce_c	0
u_le_b	1
v_e	1
c_out[3:0]	0

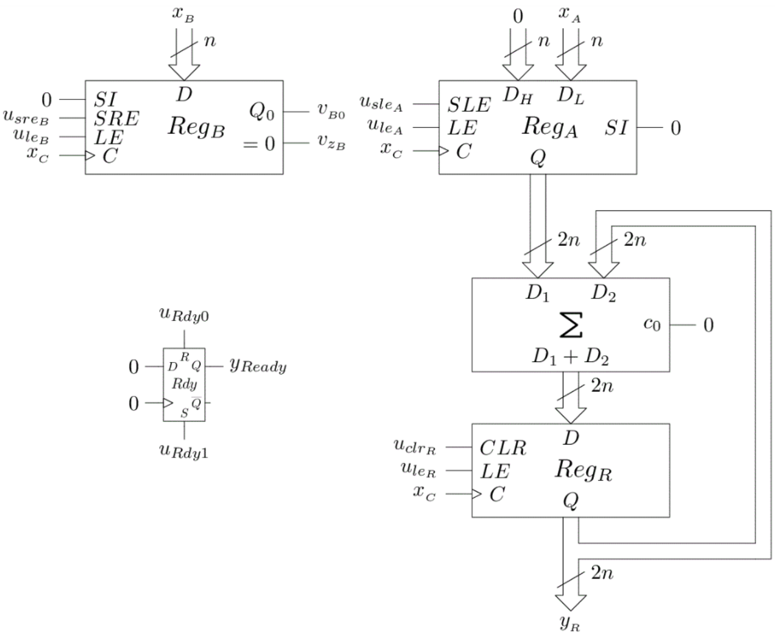
(for better quality please refer to the .zip file provided at the end of the lab)

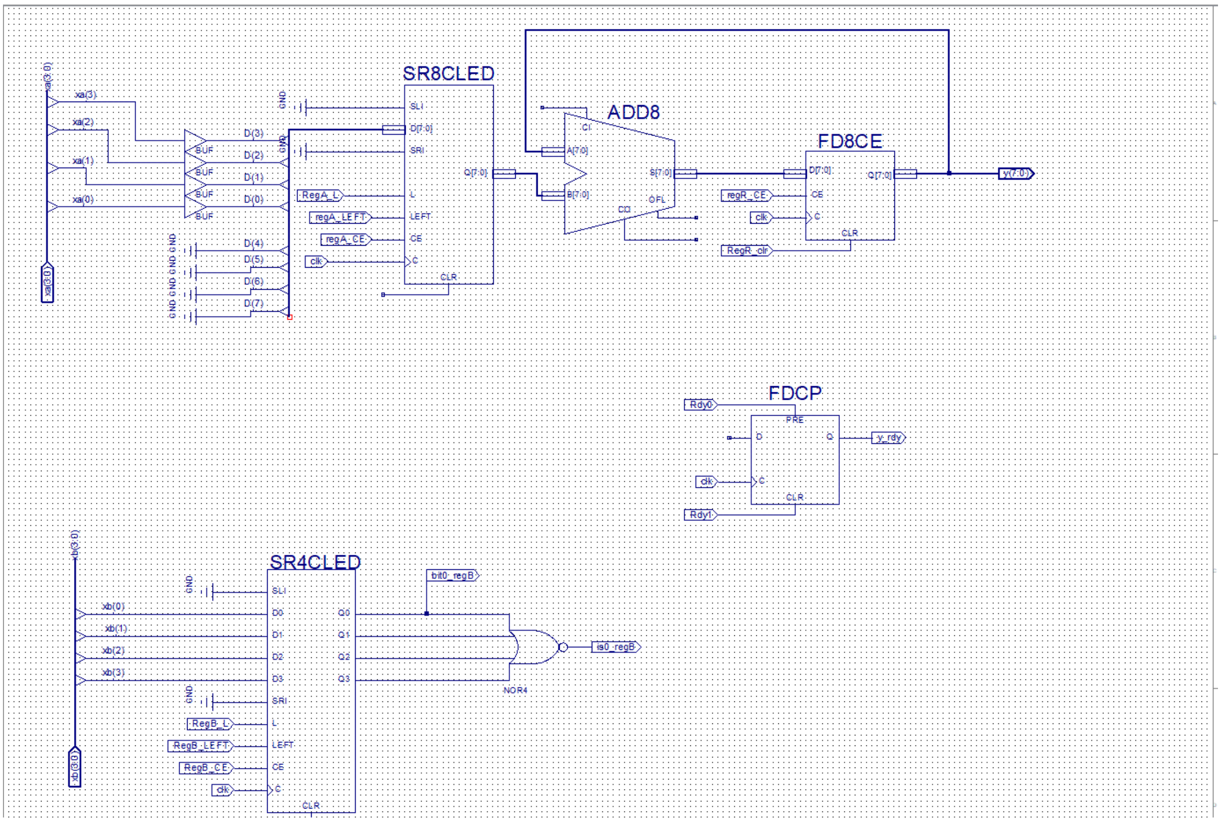
The simulation looks just as it should compared to the schematic outline and the instructions.
The numbers are multiplied (with added +1, hence result is 22), and the SR flip-flops reacts properly on correct result.

Task C

The task here was also to multiply numbers, but it used a non-obsolete technique for it. It's based on shifting binary numbers. Here is the schematic from PDF (1) that I've based my Xilinix schematic (2) on, that is shown below the PDF schematic:

Schematic of the data subsystem:

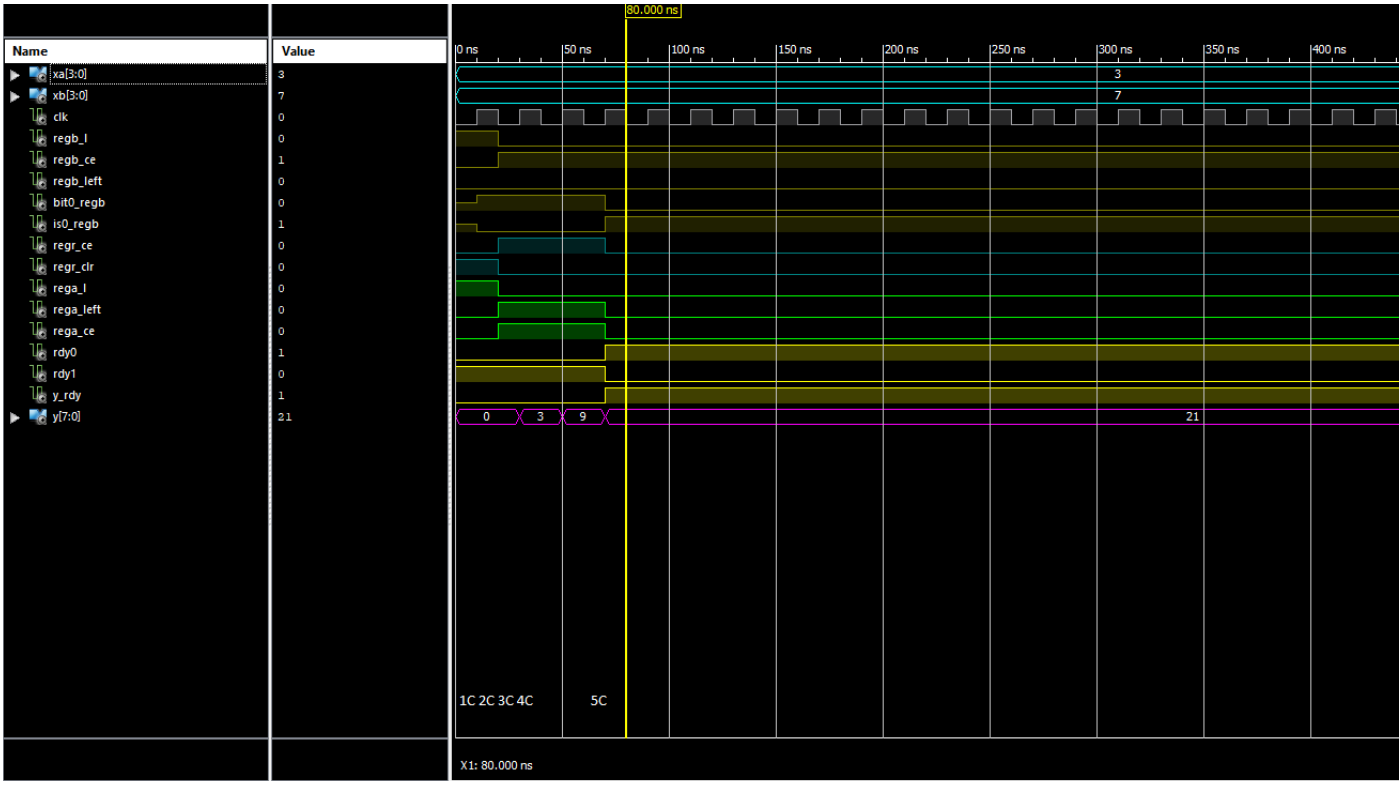




Then, same as for task A and C, I wrote a VHDL testbench based on the set of instructions found in the PDF file:

```
1.Load your numbers into registers A and B.
2.Clear register R and set C0=0.
3.If least significant bit of B is 1 load register R.
4.Shiftleftregister A and shift right register B
5.Repeatsteps 3-4 until output of B becomes 0.
```

Then after checking the testbench and running the simulation, I changed signal order, radix, colours etc. and got the following simulation output:



Name	Value
xa[3:0]	3
xb[3:0]	7
clk	0
regb_l	0
regb_ce	1
regb_left	0
bit0_regb	0
is0_regb	1
regr_ce	0
regr_clr	0
rega_l	0
rega_left	0
rega_ce	0
rdy0	1
rdy1	0
y_rdy	1
y[7:0]	21

(Refer to the .zip file for better quality screenshot)

From the simulation output it can be seen that the output is correct – numbers have been multiplied.

There is only one issue for bit0_regb and is0_regb – they are undefined for the start of the simulation, which I suspect is related to the initialisation issues of a register B.