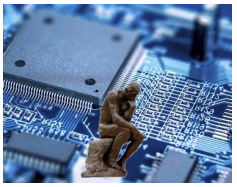
POLITECNICO DI TORINO

Dipartimento di Elettronica e Telecomunicazioni



DIGITAL SYSTEMS ELECTRONICS

Academic year 2022 - 2023

LABORATORY NR. 03

DUE DATE: April 18, 2023
DELIVERY DATE: April 11, 2023

GROUP 20

CONTRIBUTIONS:

- Bianchi Paolo (s272209)
- Youssef Abdelrahman (s262295)
- Mael Madec (s318137)

The members of the group listed above declare under their own responsibility that no part of this document has been copied from other documents and that the associated code is original and has been developed expressly for the assigned project.

Contents

Design-wise, the goal of this lab is the conception of several arithmetic components : the adder, the subtractor and the multiplier, following multiple bit formats.
Tool-wise, it is the introductory lab for TimeQuest Timing Analyser and the determination of the maximum frequency of a FPGA circuit along with the timing analysis of a circuit with Modelsim.

The structure is sticking to the one given in the laboratory document.

1. 4-bit Sequential RCA

DESIGN ENTRY:

This part is designed in a structural way. Part1 is the main file connecting synchronous logical memories (flipflops and n-registers) with the logical part RCA_4bit.vhdl. It is also linking up the physical buttons and switches to the main circuit, as asked.

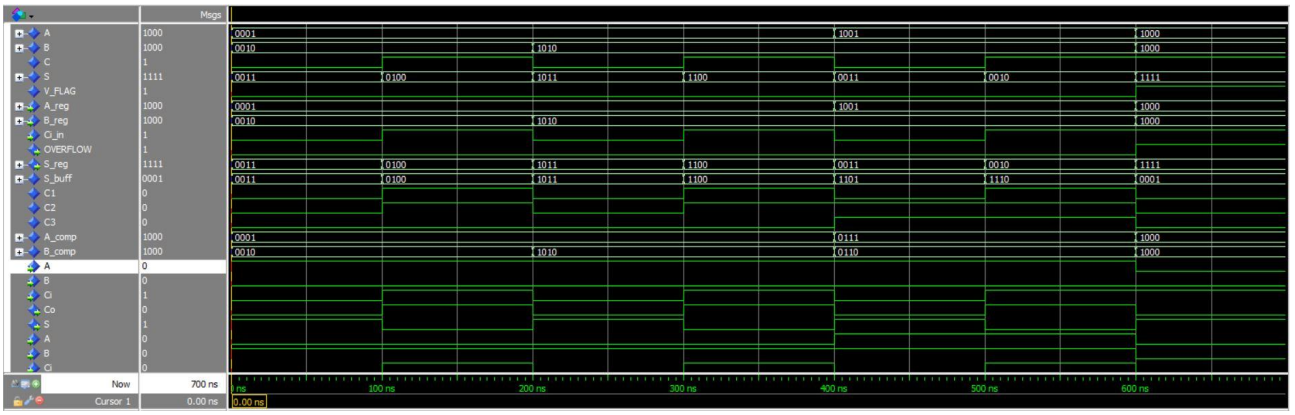
The adder is using the “SIGNED” type in order to be able to complement them easily (-SIGNED + 1 gives the complement quite rapidly) while being compatible to the bit-to-bit addition realised here.

FUNCTIONAL SIMULATION:

The simulation of the adder circuit goes with every combination tested (with and without carry) : positive + positive, negative + negative, positive + negative. At the end, a 7th case is included to test the overflow function, which is returning the last carry.
Part1_tb solely tests the connections, as the adder is tested separately.

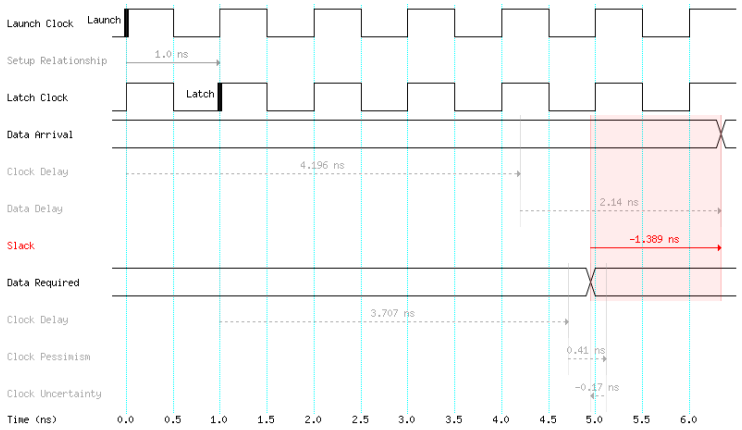
SYNTHESIS:

The simulation reports a functioning adder circuit :



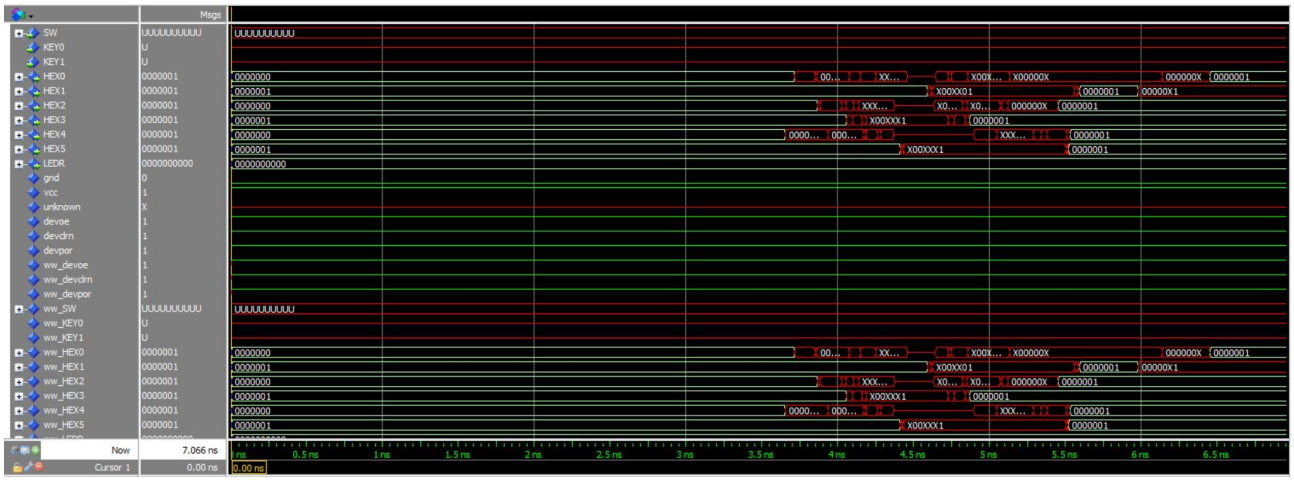
All the functionalities are working properly.

The Quartus timing analysis of Part1.vhdl (with KEY1 being the clock here) :



The slack time (with the default clock) reported here is -1.389 ns, which gives us the Fmax of 1/(1+1.389)ns = 418.585 Mhz.

Timing simulation :



2. 4-bit Sequential Adder/Subtractor

DESIGN ENTRY:

In essence, the design of the adder is the same as the part1 one, but with the addition of an addition/subtraction selector input. Complementing one of the inputs gives us the subtractor (A + (-B) = A - B). Thanks to the use of signed numbers, it is simply done by extending the conditions of the input complementions (here it is B) to accommodate the ADD_SUB input.

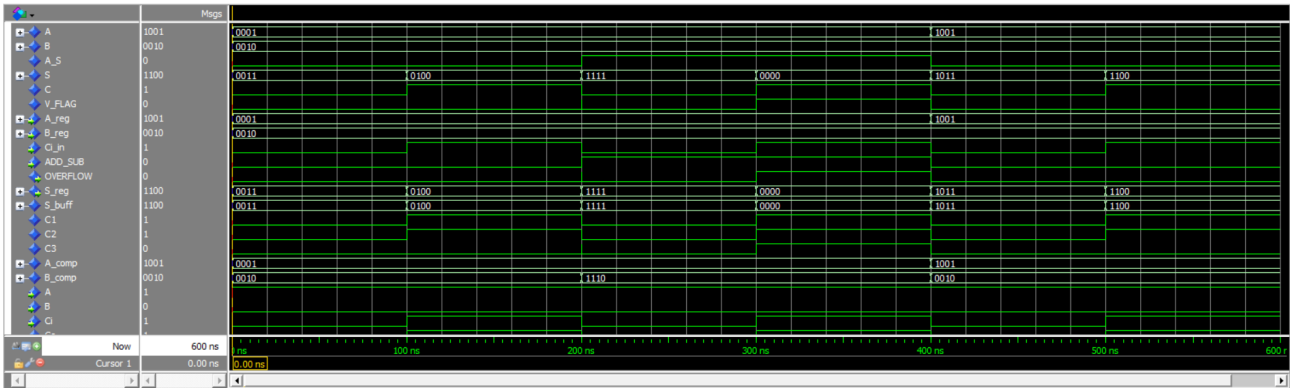
Part2's only change is the connection of SW(8) to the ADD_SUB input.

FUNCTIONAL SIMULATION:

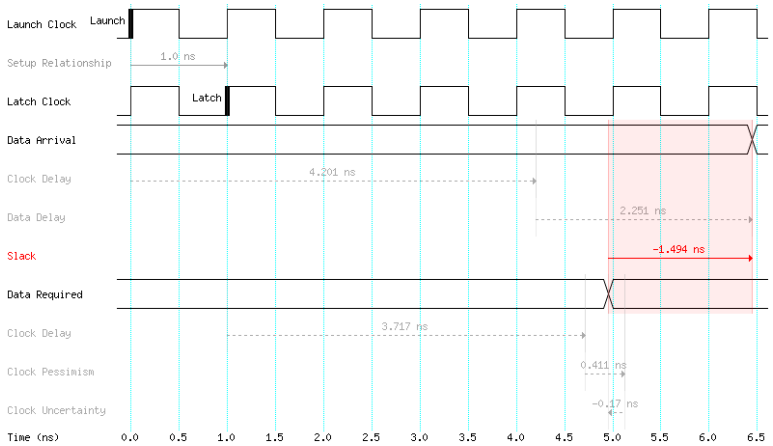
RCA_4bit_part2_tb.vhdl only tests the ADD_SUB input because the design part doesn't change otherwise. Part2_tb does the same job.

SYNTHESIS:

The new adder presents essentially the same results as the first one :



The Timing Analysis report :



The slack time reported is -1.494, which gives an Fmax of 1/(1+1.494) = 401 MHz. This frequency is coherent with Part1 one.

3. 16-bit RCA, Carry Bypass Adder and Carry-Select Adder

DESIGN ENTRY:
In this part, three designs are to be made.
For the 16 bit adder, the full adder circuit is completely replaced by the 4 bit RCA.

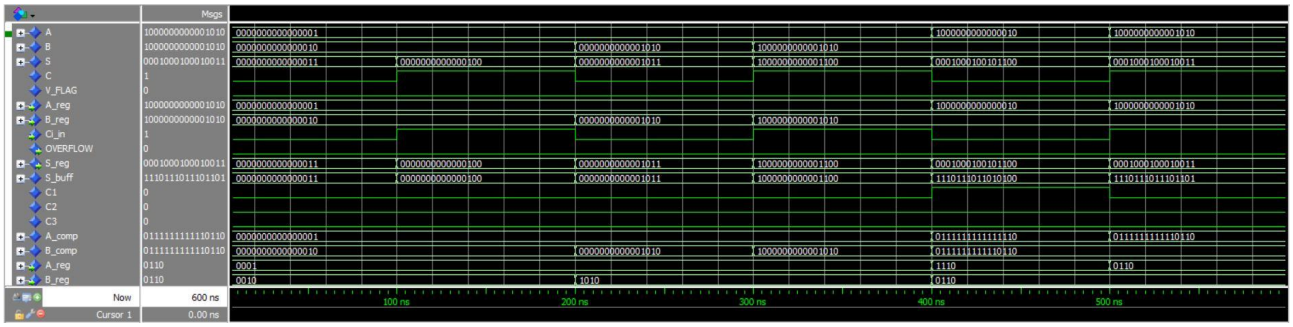
The Carry Bypass introduces multiplexers that choose which carry to propagate, either the one from the RCA or the carry used in input. The selector inputs of these multiplexers are determined by the A and B inputs of the RCAs that precede them. The behavioural way is preferred here as it is only used in this circuit.

Lastly, the Carry Select adder represent the mix of the 16 bit adder and the Carry Bypass, but instead of a choice of propagation, every possibility is calculated (meaning 8 * 4 bit RCAs with forced carry input). The same selector input in addition gives the choice for the S output chosen for the 16 bit.

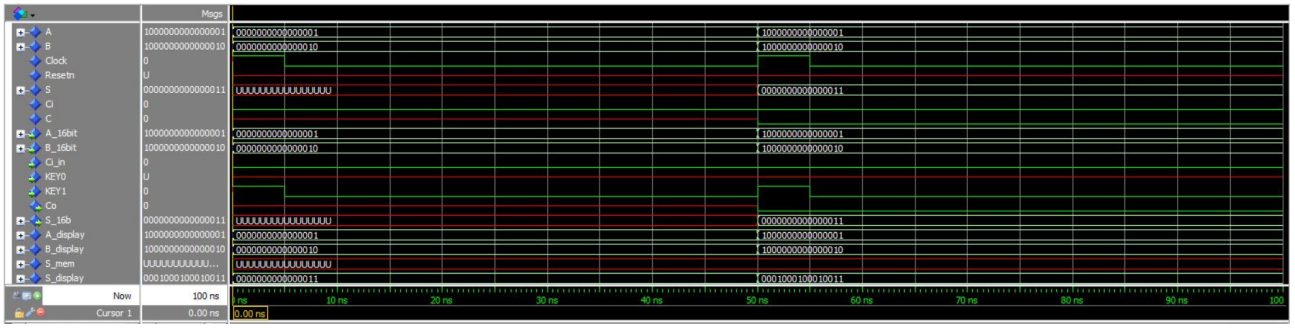
FUNCTIONAL SIMULATION:
Like the first and the second part, every component is tested first, followed by the dedicated part connecting physical interfaces. While the testbenches for the adders are personalised, the test units share the same basis as closest as possible (only the component is replaced here.). Every part should give the same outputs, whatever the adder. They are considered functioning if it's the case.

SYNTHESIS:

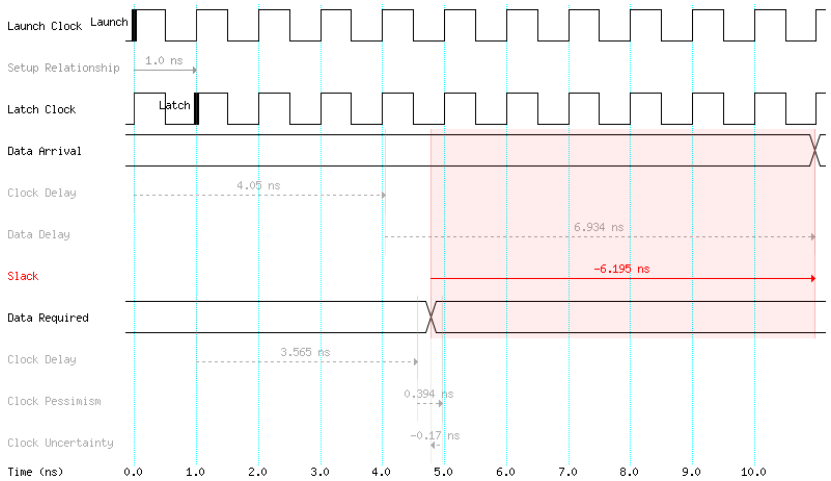
Simulation of the RCA 16bit and the Part3 dedicated to it :



The 16 bit wide adder is working as intended.

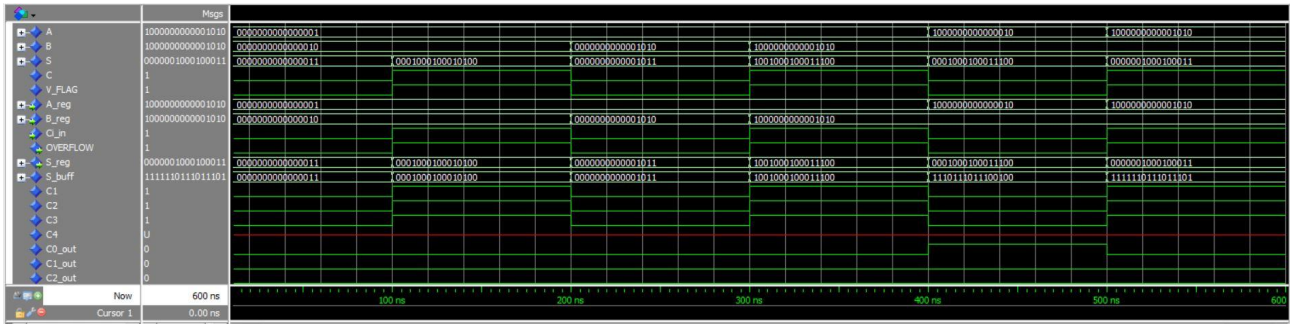


The real result to watch here is S_display, because the S register awaits the clock signal.
It is working correctly.

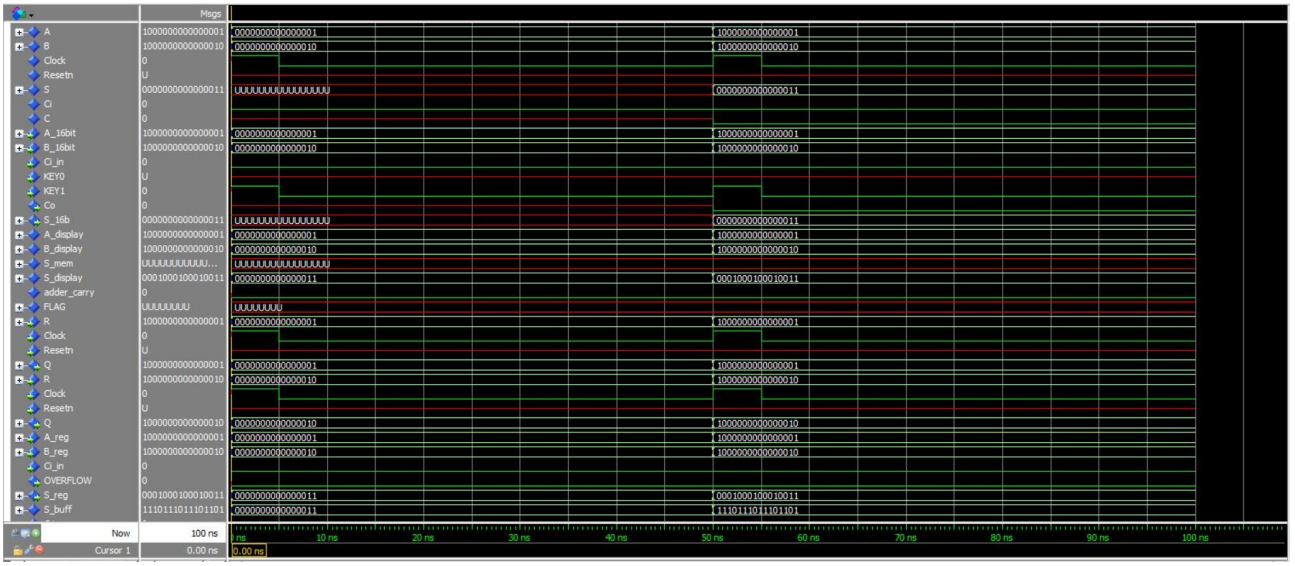


The slack time here is 6.195 ns, which is roughly 4 times the 4 bit wide RCA's time.
The max frequency is about 139 Mhz (1/(1 ns +6.195 ns)).

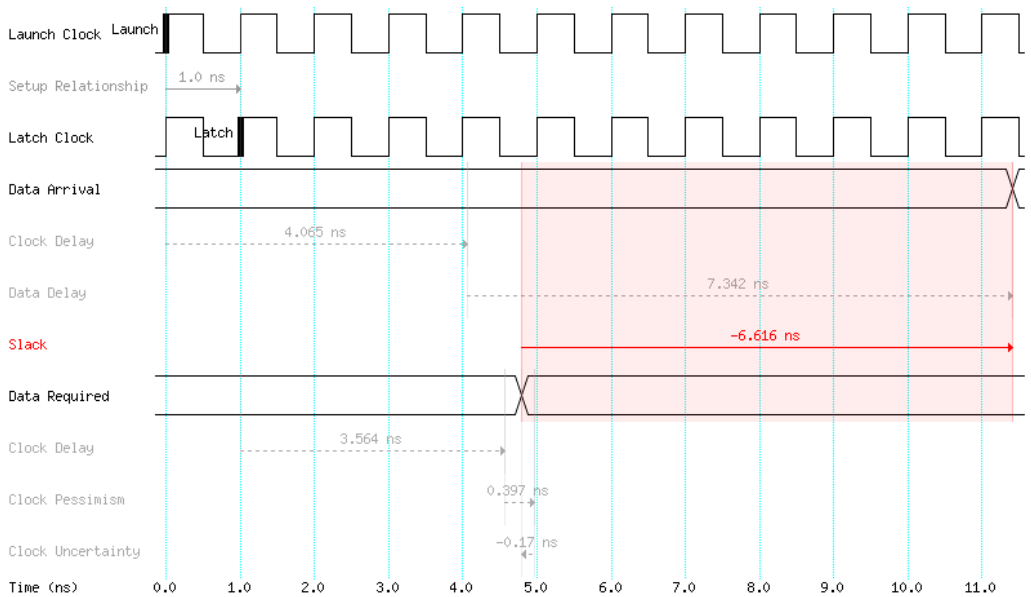
Simulation of the Carry Bypass adder and the part :



The Carry Bypass gives the same result as the RCA, it is working as intended.

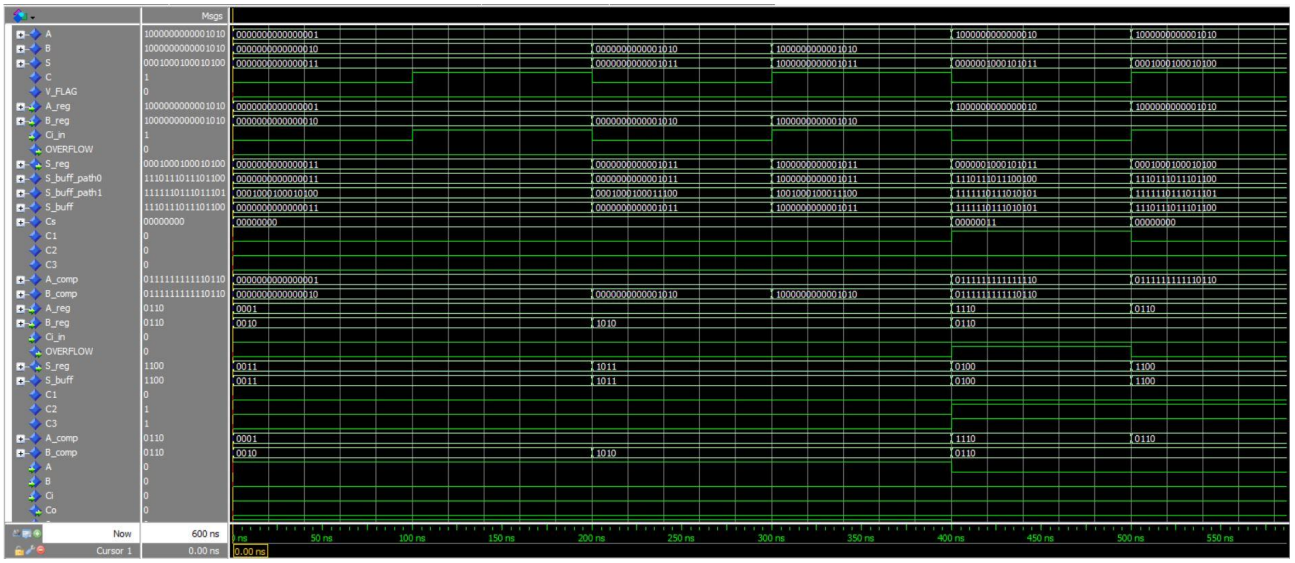


Working as the RCA part => functioning well.

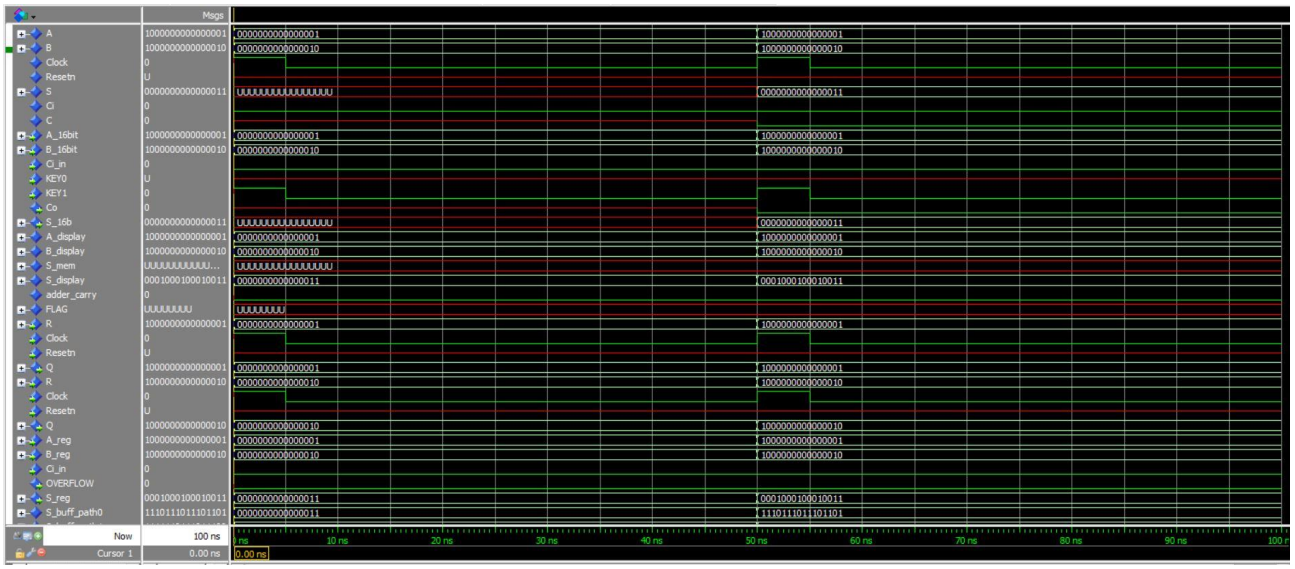


The Fmax is 131.3 Mhz. This outlines a non-optimized code, as the max frequency should be higher given the possibility to shortcut some RCAs.

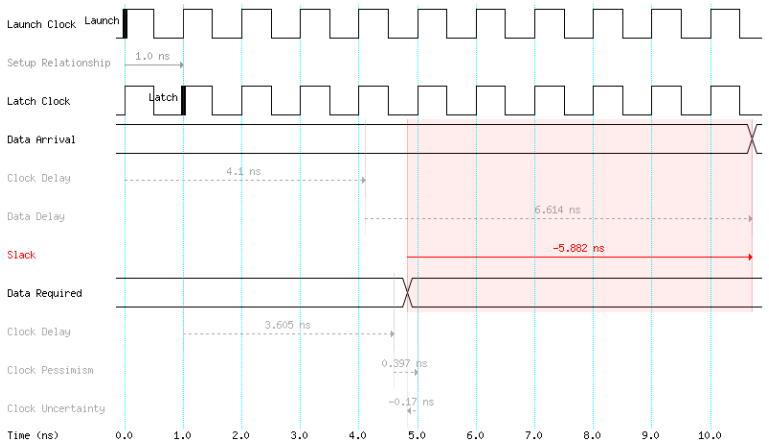
Simulation of the Carry Select adder and the part associated with it :



Similar to the simulation of the RCA and the CB => fine.



Similar => working fine.



In this case, Fmax = 145.3 Mhz. Contrary to the Carry Bypass adder, the slack time is shorter than the 16 bit wide RCA, indicating this time that the general circuit is working properly

4. Multiplier

DESIGN ENTRY:

The multiplier requires many operations and components to accomplish the desired multiplications. Conveniently, these are achieved by ranks (here 3). For each rank, an RCA is used along a 4x1 AND door (dedicated component). The end result is an amalgamation of several results and inputs of the RCAs. Lastly, the numbers used are unsigned to facilitate the operation. The hexadecimal display of numbers is done by a sole component hex_display.vhdl (instead of a structural component, it is done behaviorally).

FUNCTIONAL SIMULATION:

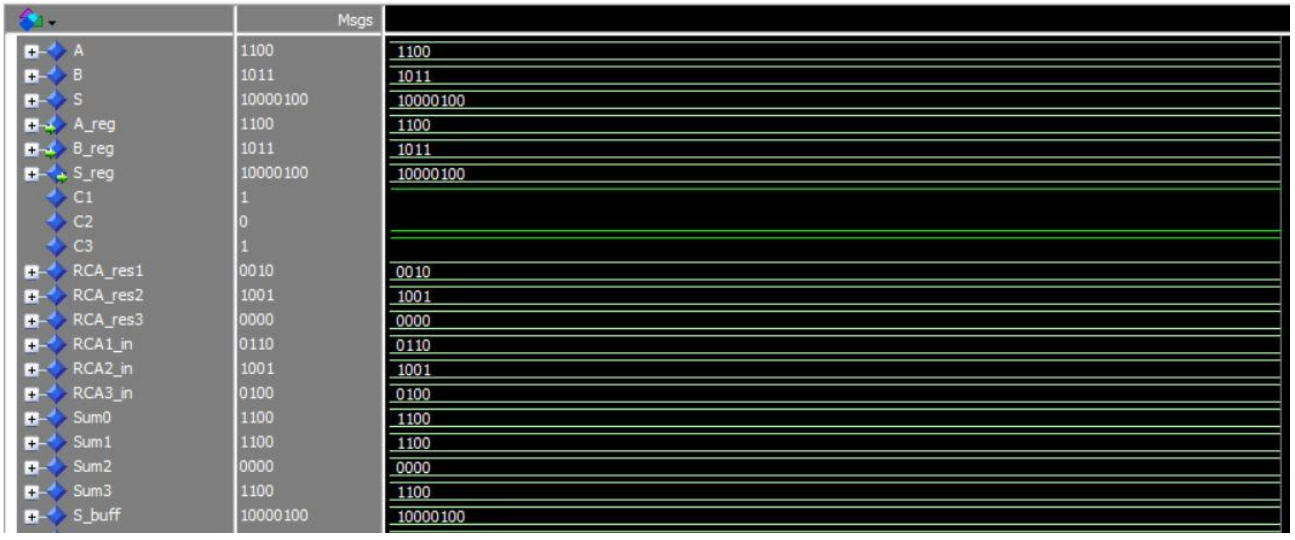
Two tests are done:

- The multiplier, given its complex nature. If the multiplication issued in the LAB doc is correctly performed, then the multiplier is working properly.

- The Part4_tb, to test every interaction between components along with the hex_display.vhdl. Once again, the same multiplication is done. This time, the results and the inputs are to be shown correctly.

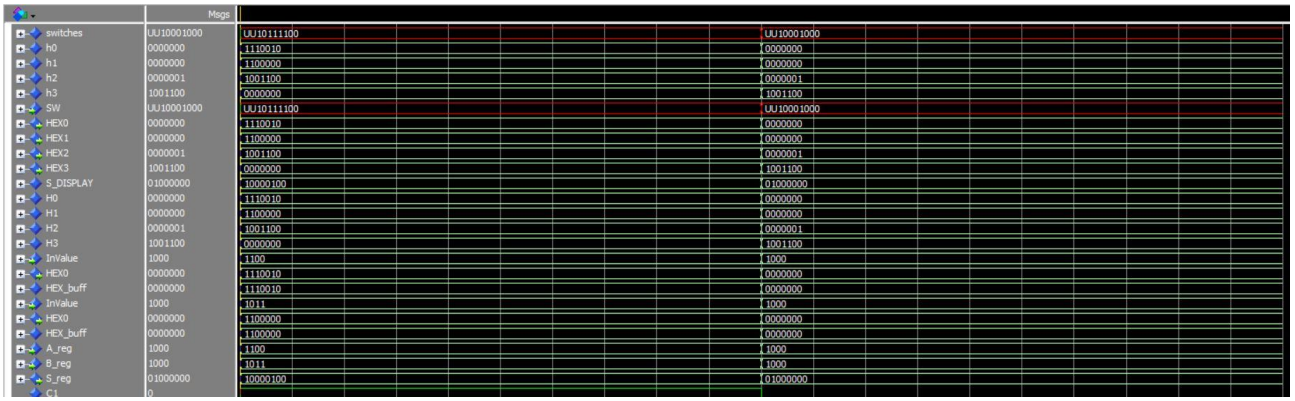
SYNTHESIS:

The testbench simulation of the multiplier :



The result S is correct, hence the multiplier coded here is working.

The testbench simulation of Part4.vhdl :



The two last bits of switches (and SW) are uncoded here, no need to. Once again, the result is correct and every number is displayed correctly on the HEXi displays.