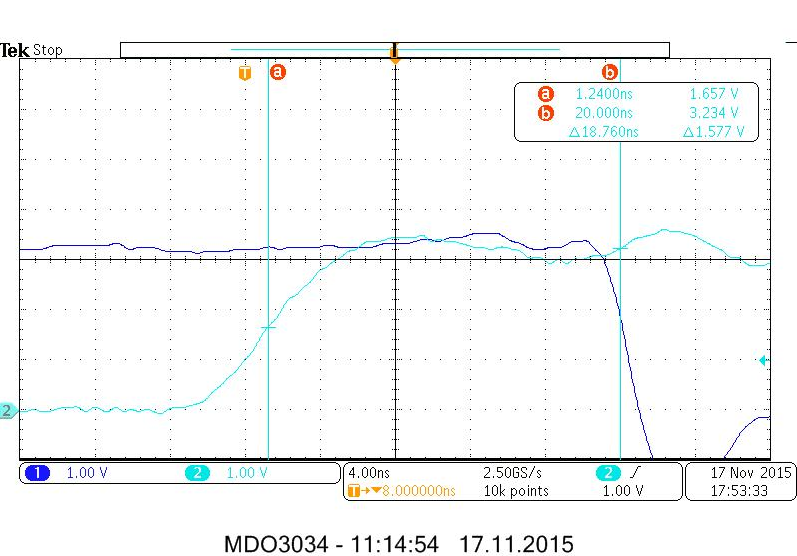
**IE3-DI Digital Circuits LAB REPORT**

|  |  |
| --- | --- |
| Lab session (please tick)  1 2 3 4 | Topic: |
| Lab group (please tick)  01 02 03  Team name / number | Team members Responsible Author   (please tick) |
| 1  |
| 2  |
| 3  |

|  |  |
| --- | --- |
| correctness of **lab preparation** (hard copy) | / 5 pts. |
| **lab tasks** successfully completed (proven by protocol) | / 5 pts. |
| **lab report**  purpose and set-up of the lab tasks are properly introduced and supported by sketches, figures, flowcharts, etc. results are summarized at the end  correctness of information given in report  results are proven by e.g. measurements, simulations, calculations, source code etc. and have been discussed  correctness of formal aspects, e.g. figures with numbers and captions, citation of foreign sources, structure, spelling and style | / 3 pts.  / 2 pts.  / 2 pts.  / 3 pts. |
| **sum** |  |
|  | |

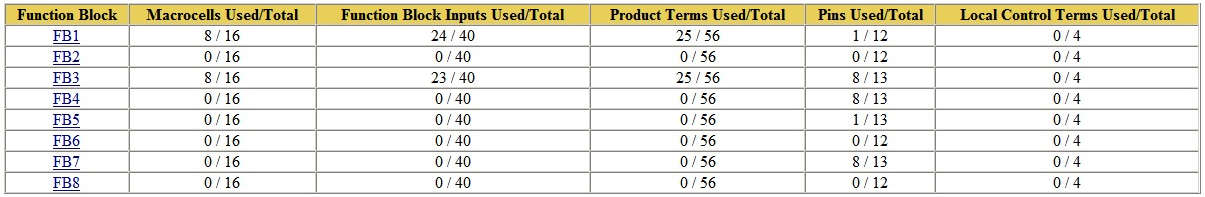
**Lab Task 2.1** (8-bit ripple-carry adder)

The screenshot below shows a result of measuring the time delay using an oscilloscope. Here we have a 18.76ns delay between C\_IN and S(7).



**Lab Task 2.2** (8-bit ripple-carry adder)

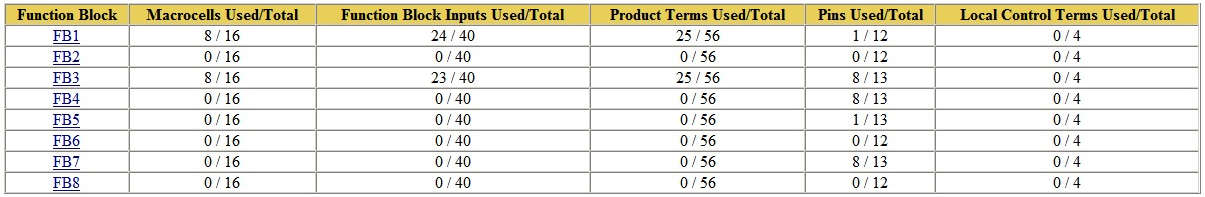
Next screenshot is about Function Block. It's noticeable that CLPD doesn’t use a lot of resources. It might mean that CLPD is capable of doing more complicated things.



**Lab Task 3.1** (8-bit arithmetic adder)

I understand that it is no excuse and I’m terribly sorry about it, but, unfortunately, the screenshot file was somehow corrupted (and I don't know if it is OK to visit lab room after the actual lab sessions). Good thing I remember clearly that for some reasons the delay between C\_IN and S(7) was also 18.76ns which might prove that two implementations (ripple-carry and arithmetic) are working equivalently.

**Lab Task 3.2** (8-bit arithmetic adder)

As a possible proof for abovementioned problem with Task 3.1, here’s a screenshot of Function blocks. Here we also can see that CLPD uses same little amount of resources (if term “little” is appropriate in such cases). 

**Lab Task 3.3** (8-bit arithmetic adder)

This task wasn’t done due to a lack of time (the lab session has finished).

**Summary**

Both Ripple-adder and Arithmetic adder implementations have shown some matches in time delay and use of resources in Function blocks. Thus with just that much data (while not having Boolean Expressions’ Schemes) we might conclude that they are equivalent in theory and practice.