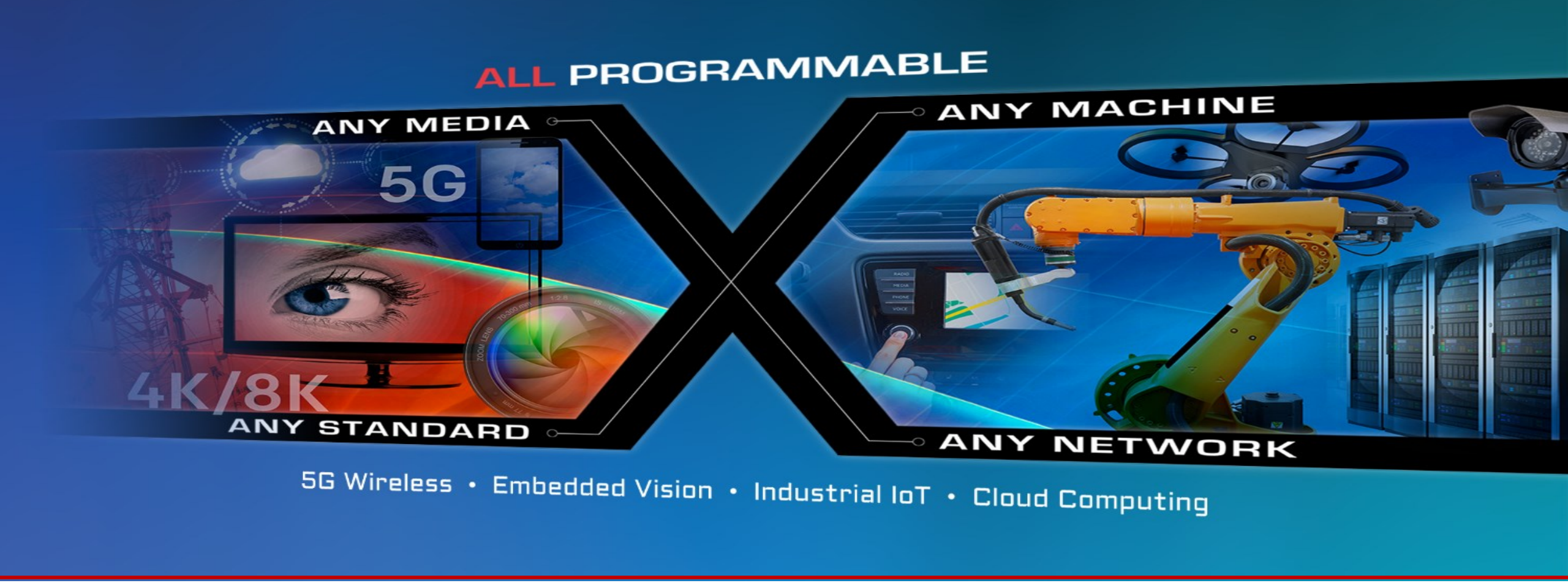


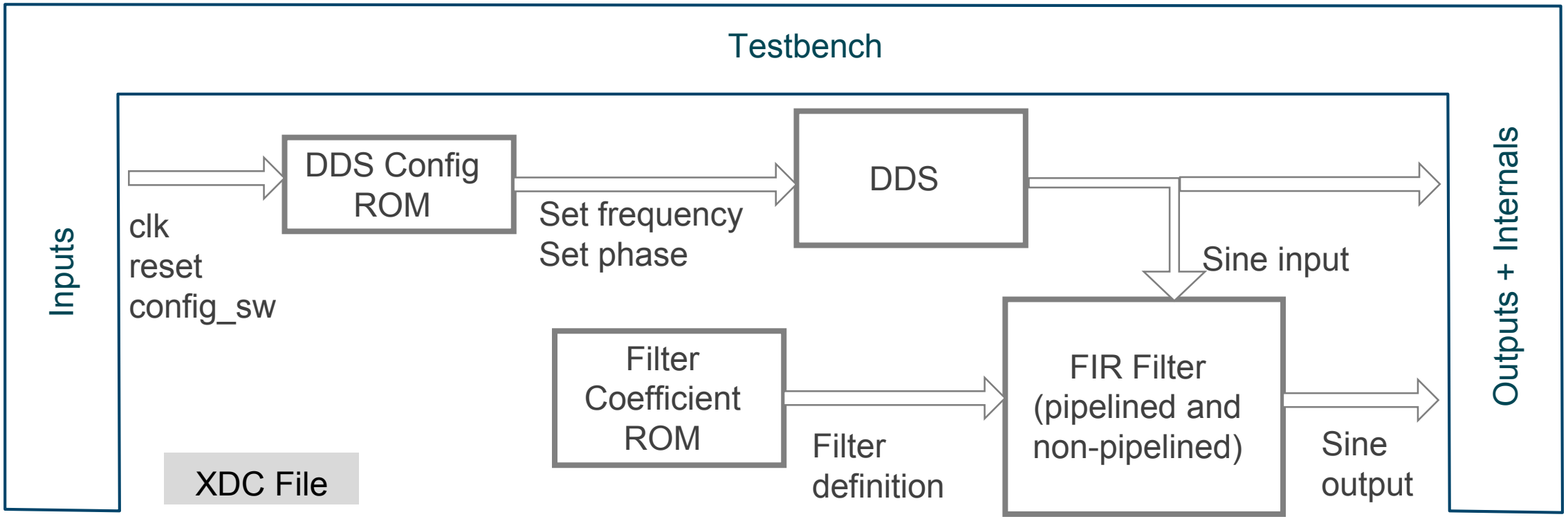
POLITÉCNICO DE LEIRIA

XILINX
ALL PROGRAMMABLE™

Project 1 Support

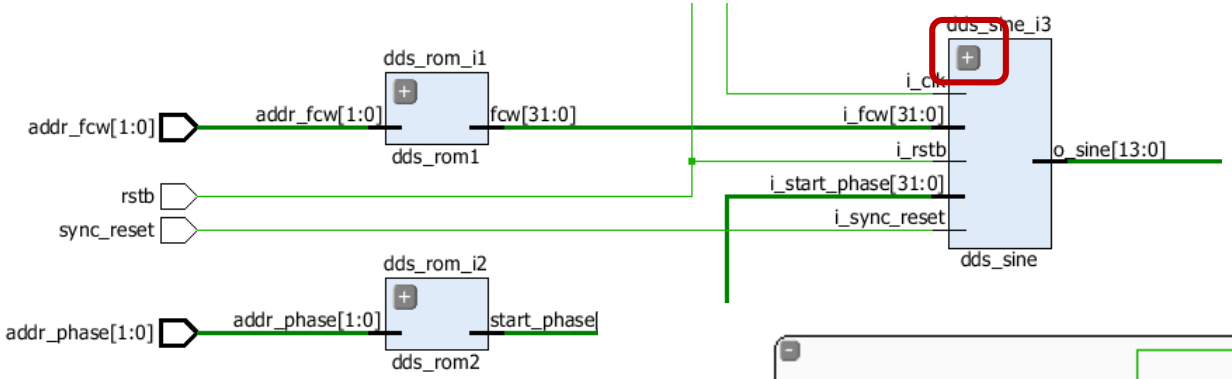
Project 1

- Each group will analyze a different design, but all with the same structure.
- Each design includes a DDS generator and a FIR filter, implemented in VHDL.

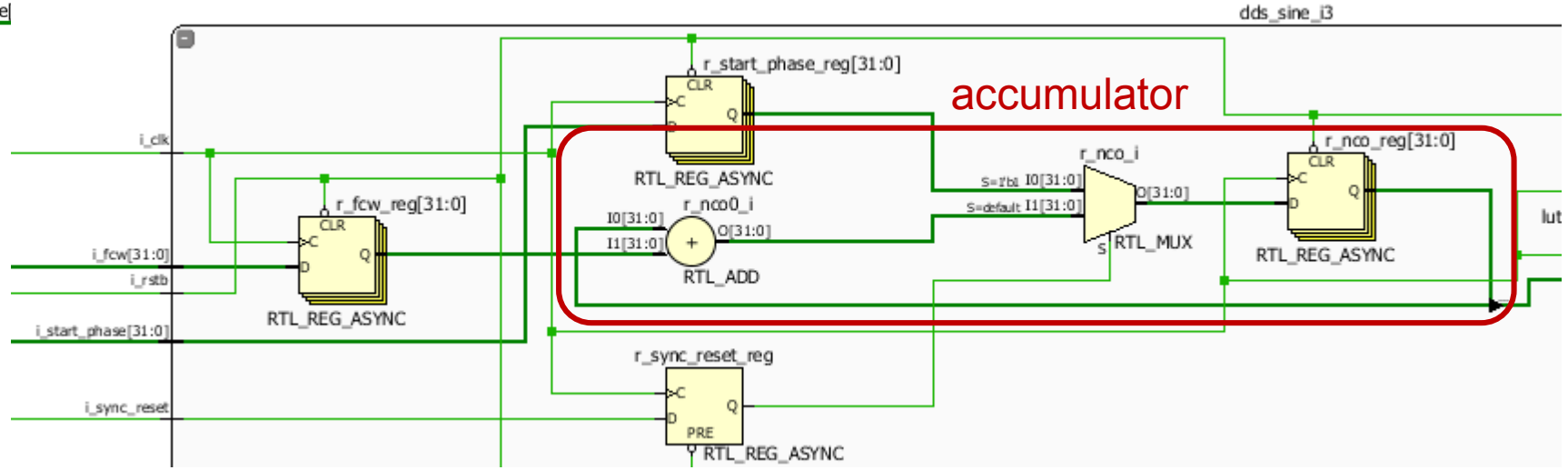


Project1 - DDS_sine

- The DDS frequency is set by a 32 bit vector (fcw), which is the input of an accumulator



- The DDS LUT is read at 100MHz rate but with a step defined by fcw

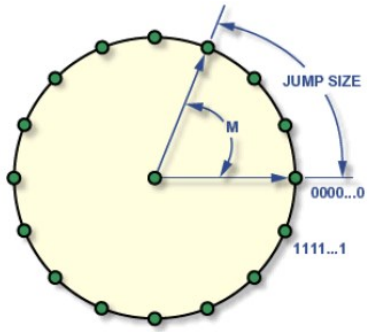


Project1 - DDS_sine

- With the initial value:

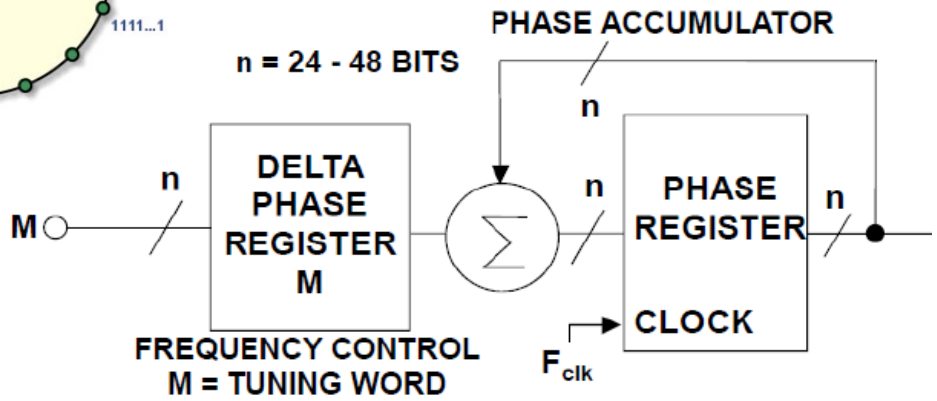
$$fcw = 028F5C28x = 42949672_{10}$$

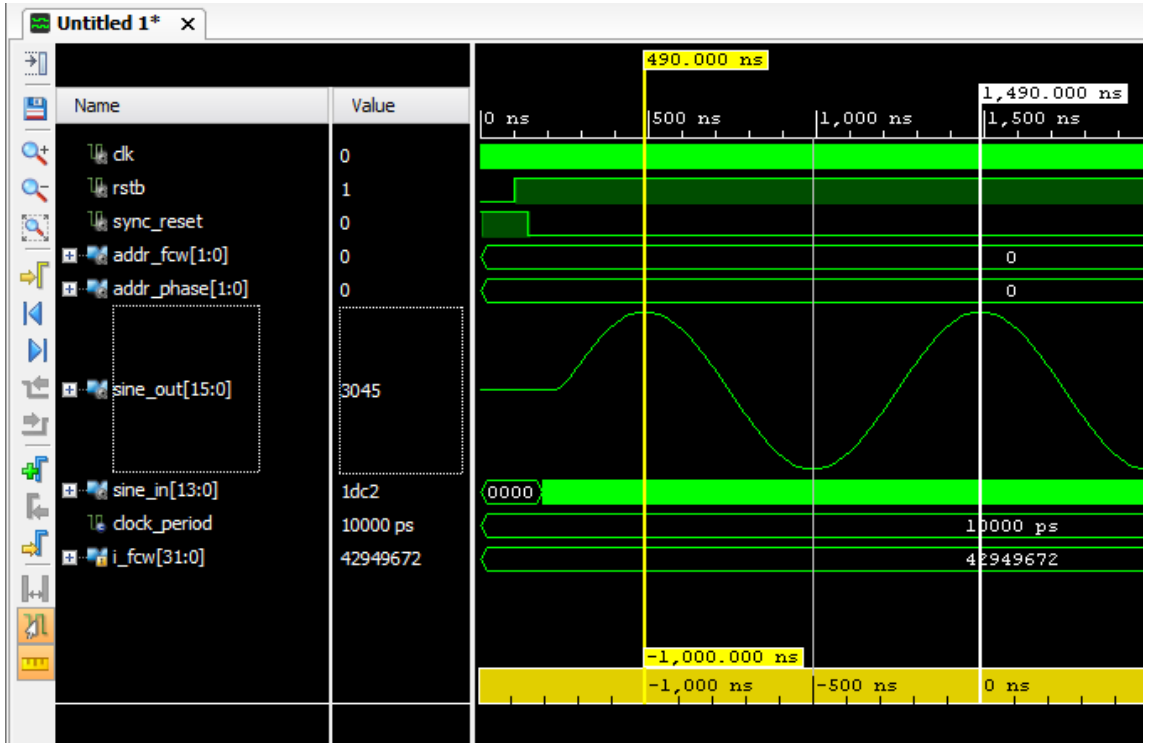
- $Fout = 10MHz \rightarrow Tout = 1us$

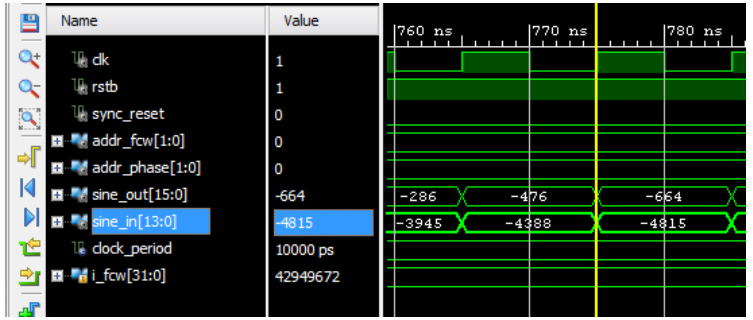


$$F_{out} = M \cdot F_{REF} / 2^n$$

n = # bits of phase accumulator
M = binary tuning word



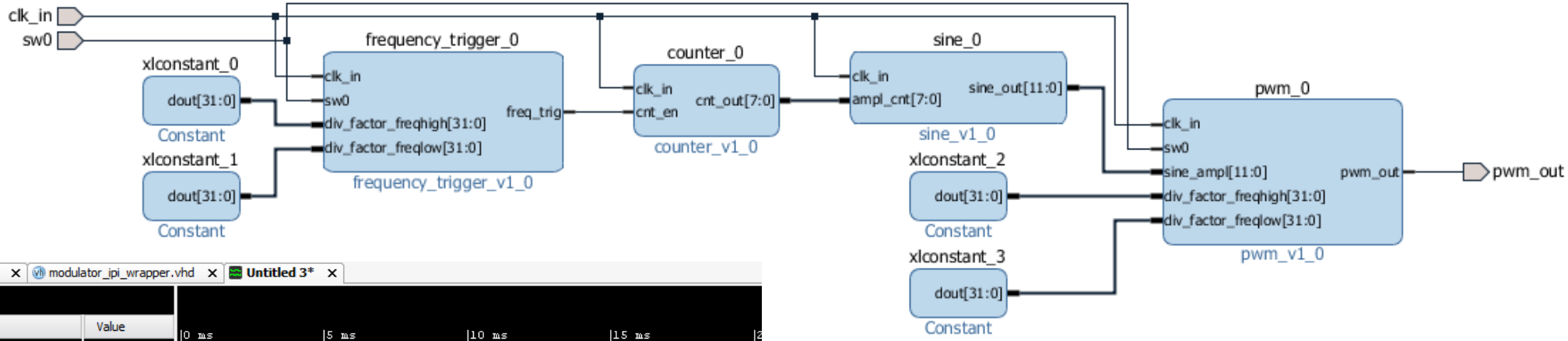


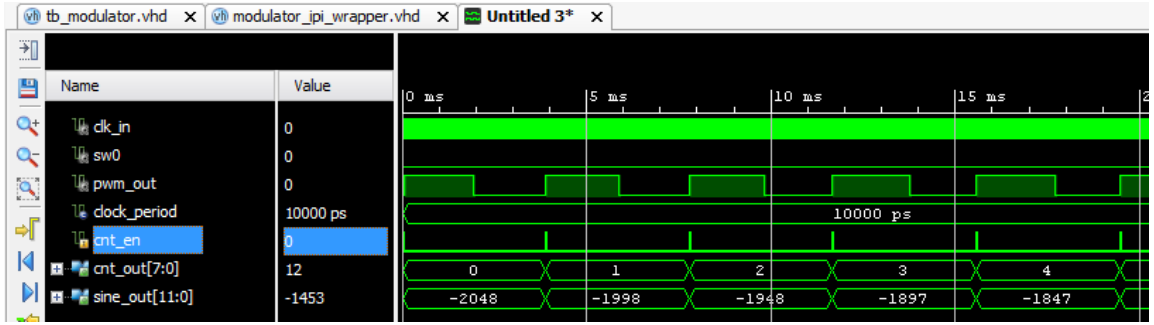


Each clock cycle there is a new sine sample at the DDS LUT output

Lab4 - DDS_sine

- In lab 4 you have used a DDS that works in a different way
- The sine LUT has only just 256 samples, that are read at a slower or faster rate, depending on the counter enable input (cnt_en)



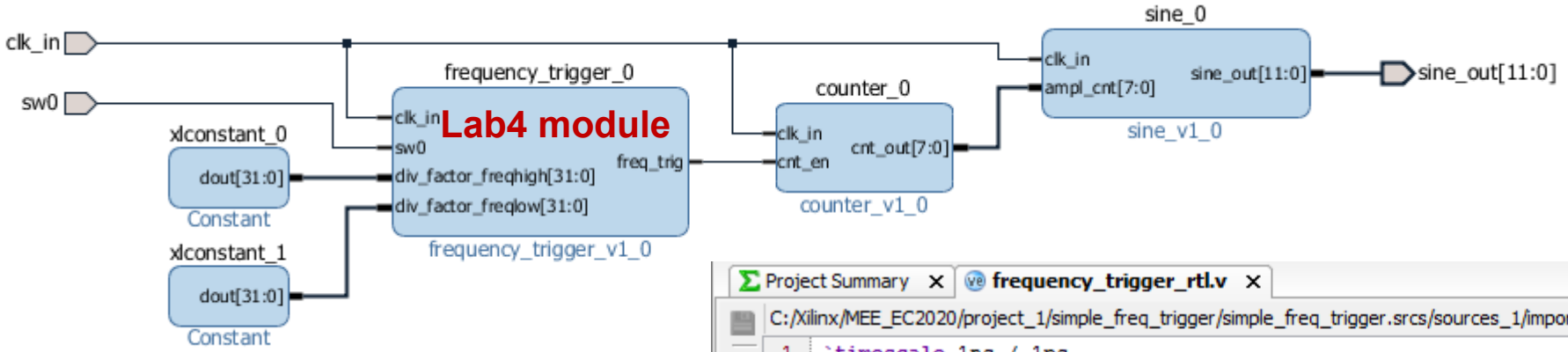


Tasks

1. Elaborate the design and analyze the design's architecture
2. Simulate the design, observe it's behavior and discuss results
3. Synthesize the design using different settings and observe the effects of the options taken ("flatten hierarchy" and "max_dsp" options as minimum goals).
4. Analyze the various reports, especially with regard to resource utilization and timing performance, taking into consideration the synthesis options.
5. Analyze the impact of using a pipelined vs. non-pipelined FIR architecture, in the maximum clock frequency. Identify the critical path.
6. Analyze the post implementation reports, especially with regard to resource utilization and timing performance.
7. Final step … (next slide)

7. Integrate your design with an IP Integrator module

a) Modify the frequency trigger module (given in lab4) so that it accepts input parameters for frequency definition. You can add more than 2 if sw0 has more bits!



The modified frequency trigger module should have a different name!

Project Summary | frequency_trigger_rtl.v

C:/Xilinx/MEE_EC2020/project_1/simple_freq_trigger/simple_freq_trigger.srcs/sources_1/imports/lab4_sources/frequency_trigger_rtl.v

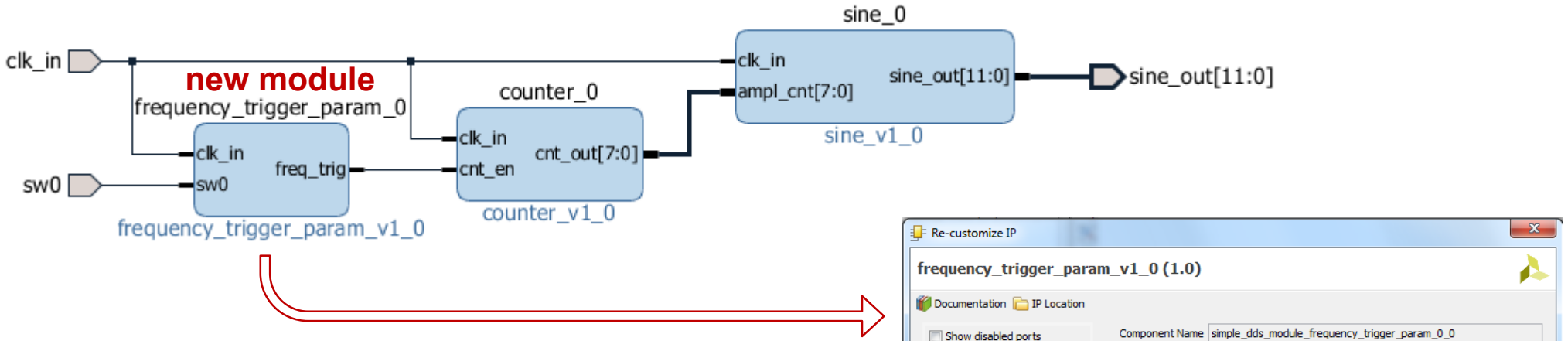
```
1  `timescale 1ns / 1ps
2  module frequency_trigger_param
3  # (parameter    div_factor_freqhigh = 32'd110592,     // replaces constant 0
4                  div_factor_freqlow = 32'd389120       // replaces constant 1
5    )
6     (  input clk_in,                    // input clock signal
7       input sw0,                        // signal used for selecting frequency
8  //     input [31:0] div_factor_freqhigh, // input clock division factor when sw0 = '1'
9  //     input [31:0] div_factor_freqlow,  // input clock division factor when sw0 = '0'
10      output reg freq_trig              // output signal which frequency depends on the sw0 state
11      );
12
```

Define input parameters

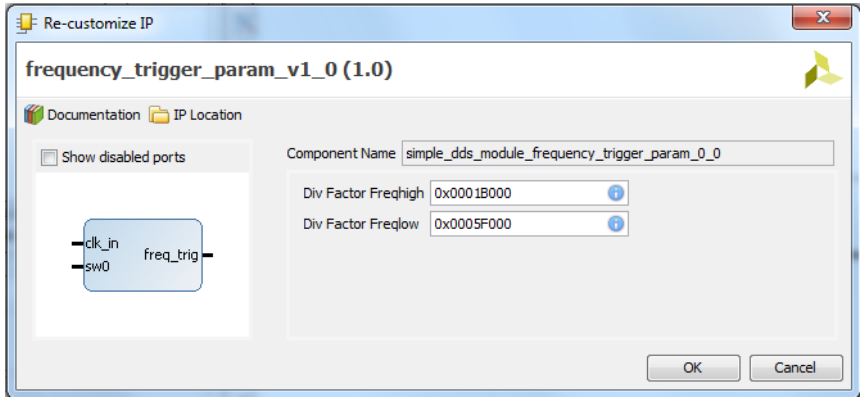
7. Integrate your design with an IP Integrator module

b) Generate a block design for a simple DDS with a modified frequency trigger module (so that it accepts input parameters)

c) Simulate this block design to see if it works as expected.



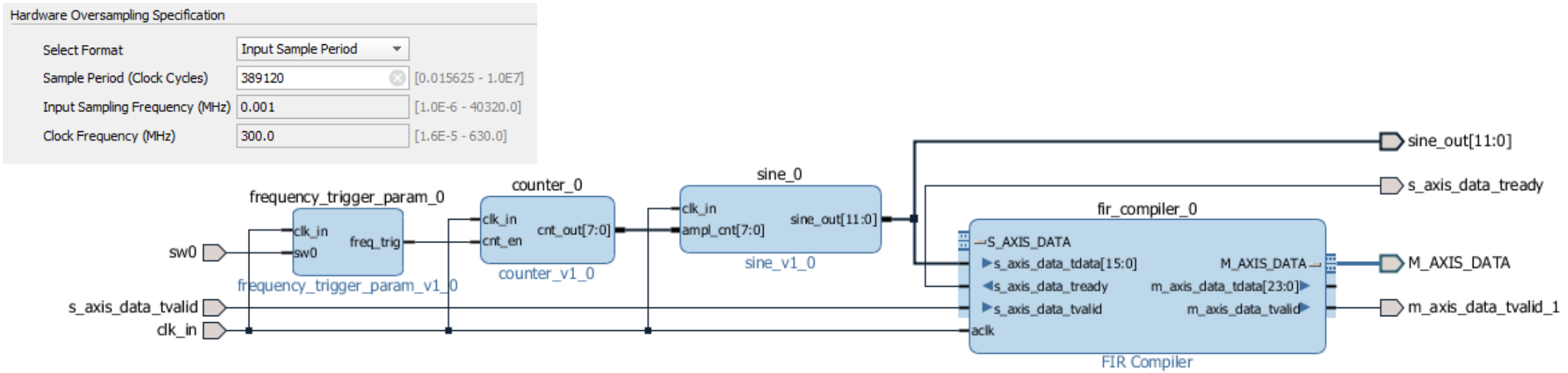
Input constants are now parameters you can modify in block design



7. Integrate your design with an IP Integrator module

d) Add a FIR Filter from the IP Catalog (FIR Compiler) and experiment different configurations.

1. Use different Number of Coefficients and observe its influence in FPGA resource usage
2. Use different Coefficient Width and observe its influence in FPGA resource usage
3. Use symmetric and non-symmetric coefficients and observe its influence in FPGA resource usage
4. Use different Hardware Oversampling specifications and observe its influence in FPGA resource usage and performance



e) Comment these results in light if the ones obtained with the previous filter and DDS

More info

- Project 1 **new delivery date** → 4th april
- Next on-line sessions (project 1 support):
  - Tuesday 24rd march between 10h00 – 12h30 am (I will be available in ZOOM if you need help)
  - Tuesday 31th march between 10h00 – 12h30 am (I will be available in ZOOM if you need help)
- TP classes scheduled for the 26th of march and 2nd of april will be postponed until after Easter Interruption. We will resume with on the 9th of April with a faster pace.
- Next subject will be System Generator tool. Please give me feedback if you have this tool installed in your PC (complete survey in moodle)