CentraleSupélec

entity arch → circuit modelling

pkg pkg body

configuration

fixed point pkg.

ICS M2 Degree - 2021-2022 - Advanced Digital Electronics

Practical Session N°7 & 8

reset active low

n bits

Design of a frequency divider based on a gray code counter

We wish to make a synchronous counter of a first clock in order to make a clock division. We also want to sample the output by means of another asynchronous clock of the first clock.

1°) Interest to use a gray code.

We consider a 4-bit binary counter. The counter changes from value 7 to value 8. Make a chronogram of the state of its outputs at the time of transition.
Its output is sampled at the time of transition. Knowing that each bit can change state with more or less delay, what values can be obtained after sampling its output at this precise moment?

2°) In order to remedy this problem, a gray code counter is used. Show how this code allows to solve the previous difficulty.

3°) Programming the counter :
The coding of a gray code counter is simplified by adding a virtual bit to the right of the LSB. This bit is equal to 1 when the counter is at 0 and changes value at each clock cycle.
Put in a table the values taken by the counter over time by adding the virtual bit.

4°) Deduce a condition of change of state for each bit of the counter

5°) Make a simple diagram based on logic gates and flip-flops of the 3 LSBs of the counter.

6°) Starting from the supplied entity, program the counter in VHDL in the a1 architecture, then test it using the supplied testbench. The factor input, the generic div2 parameter and the clock_out output will not be taken into account. On the other hand, the enable signal allows the counter to change state when it is equal to 1. → condition for termination

7°) We wish to sample the output of the counter by means of an external clock clk2, independent and asynchronous. At the rising edge of this clock, the count_out output must contain the value taken by the counter, recoded in binary.
Using the myfuncs.vhd function library provided by the teacher, start from the previous code to create a new a2 architecture in order to carry out this sampling.

8°) We now wish to create a gray code counter with a programmable modulo.
The value of the modulo will be an entry of this counter called 'factor' in the entity.
In practice the input 'factor' will be equal to the value of the modulo minus 1 because the counter will count from 0 to 'factor'.

Using again the myfuncs.vhd function library provided by the teacher, start again from the previous code to create a new a3 architecture in order to set the modulo of the counter.

9°) We also want to generate a clock signal whose frequency will be the frequency of the initial clock divided by factor+1. Modify the previous code to add this function in order to obtain the a4 architecture.

Subsidiary question :

10°) add a generic parameter called div2 which performs a division by 2 of the initial clock before attacking the counter when div2 is equal to 1. This allows to make the counter work with a frequency 2 times lower and so it allows to work with a much higher initial frequency.
Be careful that dividing the clock by 2 should not change the generated frequency. So the division factors must also be divided by 2.