PROGETTO di INFORMATICA INDUSTRIALE

GRUPPO 10:

SILVA EDOARDO 816560

OUTLINE

- Introduction
- Specifications
- Input/Output Ports&Internal Signals Table
- Hierarchical Block
  - VHDL Design of all stages
- Simulation Results
- Conclusions

INTRODUCTION

- The project request is to implement a roadlight with 3 lights in output and 3 working modes, with a clock 100Mhz = 1 clock every 10ns.
  - Red and Yellow in maintenance mode (MOD5) are modulated by 2 different inputs signals.
  - The resulting output have been put in AND with the “enable” signal, that is accountable as a ON/OFF switch.
  - Since signals in the modes are independent, the clock-generator have been omitted for 3 ad-hoc signal components.
  - The 3 modes are: **nominal**, **standby** and **maintenence**; since mode and standby are to be modulated with their own mode-switch input signal and maintenance have to be acted only when the roadlight is turned on or restarted, those 2 execution modes have been tested jointed with a supercomponent “**Normal**” (and for now on i-ll refer as **Normal** as: MOD3 or MOD4 depending by “mode” input value).
  - However, for testing purposes, every 1 second in the FPGA has been converted into 100 ns. Since simulating a 1 sec with ModelSim would take too long.

SYMBOL: I/0 DIAGRAM

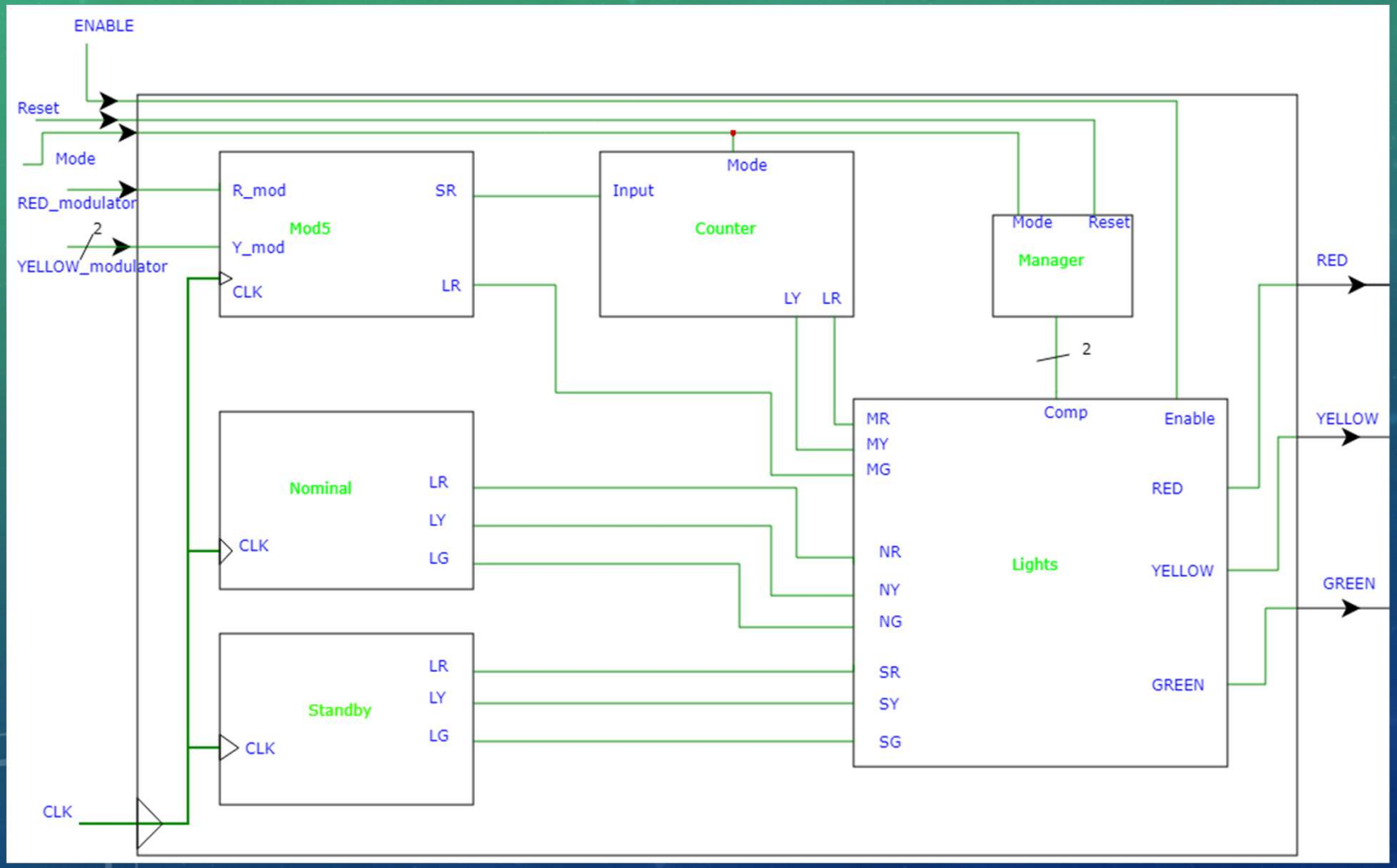
SYMBOL: I/O TABLE

Name	Size	Direction	Comments
Enable	1	input	ON/OFF switch for the circuit
Reset	1	input	Simulate a button that, when relased(falling edge), put the light at default function.
Mode	1	input	Simulate 2 buttons/switch for functioning mode • MOD3:nominal → 0 • MOD4:standby → 1
RED_modulator	1	input	Modulate the length of red-light during maintenance mode (MOD5) • 0→ 0.5s • 1→ 6s
Yellow_modulator	2	input	Modulate the length of yellow-light respect red-light during maintenance mode (MOD5) • 00→ ½ Red • 01→ Red • 10→2 Red • 11→2 Red

SYMBOL: I/O TABLE

Name	Size	Direction	Comments
Clock	1	input	Circuit clock
RED	1	output	Red-light
YELLOW	1	output	Yellow-light
GREEN	1	output	Green-light

TOP-VIEW BLOCK SCHEME



INTERNAL SIGNALS TABLE

Name	Components	Size	Comments
SR-Input	Mod5-Counter	1	Signal counting half the interval of a blink of red-light during Maintenance mode.
LR-MG	Mod5-Lights	1	Signal of green-light during maintenance mode
LY-MY	Counter-Lights	1	Signal of yellow-light during maintenance mode
LR-MR	Counter-Lights	1	Signal of red-light during maintenance mode
LR-NR	Nominal-Lights	1	Signal of red-light during nominal mode
LY-NY	Nominal-Lights	1	Signal of yellow-light during nominal mode
LG-NG	Nominal-Lights	1	Signal of green-light during nominal mode
LR-SR	Standby-Lights	1	Signal of red-light during Standby mode
LY-SY	Standby-Lights	1	Signal of yellow-light during Standby mode
LG-SG	Standby-Lights	1	Signal of green-light during Standby mode
Comp	Manager-Lights	2	A signal that indicate the possible modality in which to operate

As soon as the road-light is powered, it starts as “maintenance” mode.
When Mode gain value, is equal as pressing a button for normal behaviour:

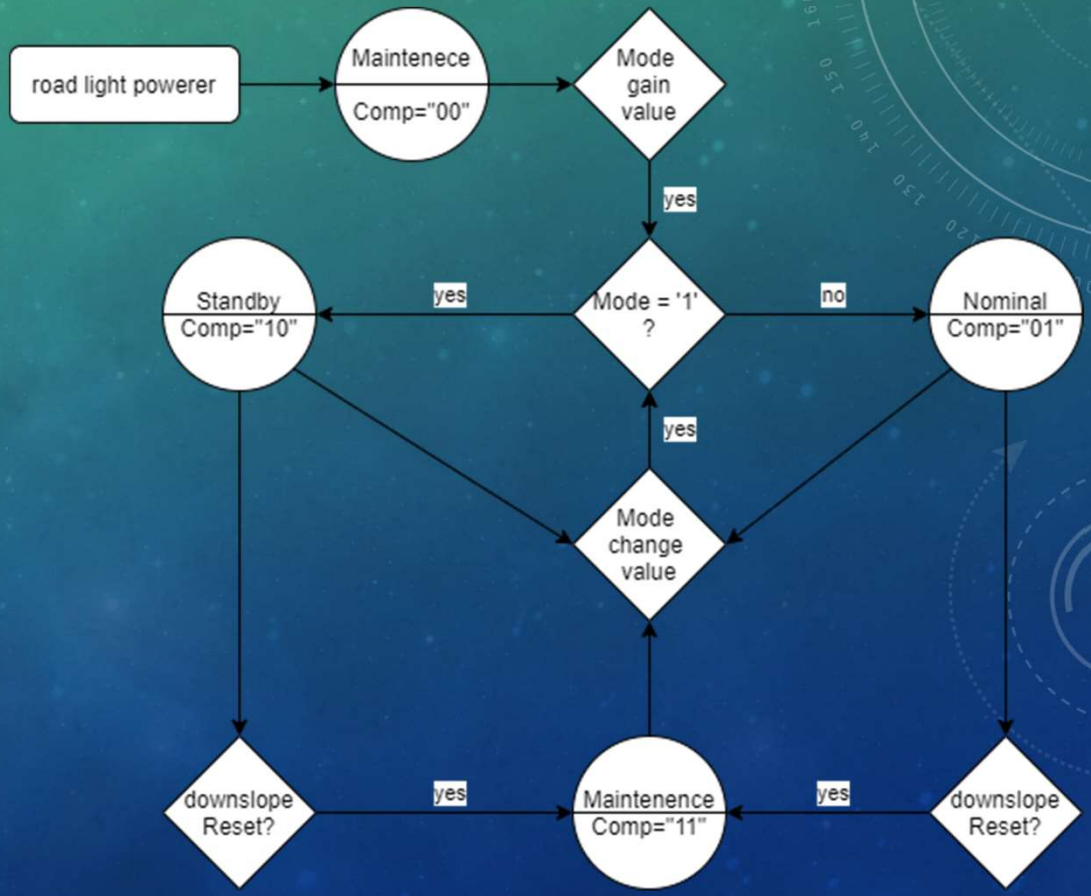
- 0 → Nominal
- 1 → Standby

The value can change any moment.

When «reset» button have a downslope, the lights are resetted to begin status (status as «11» since i cannot «detach» mode signal, so i ignore it untill it change value again»).

Every Mode is equal to different behaviour of lights.

SCHEMA OF ALL STAGES

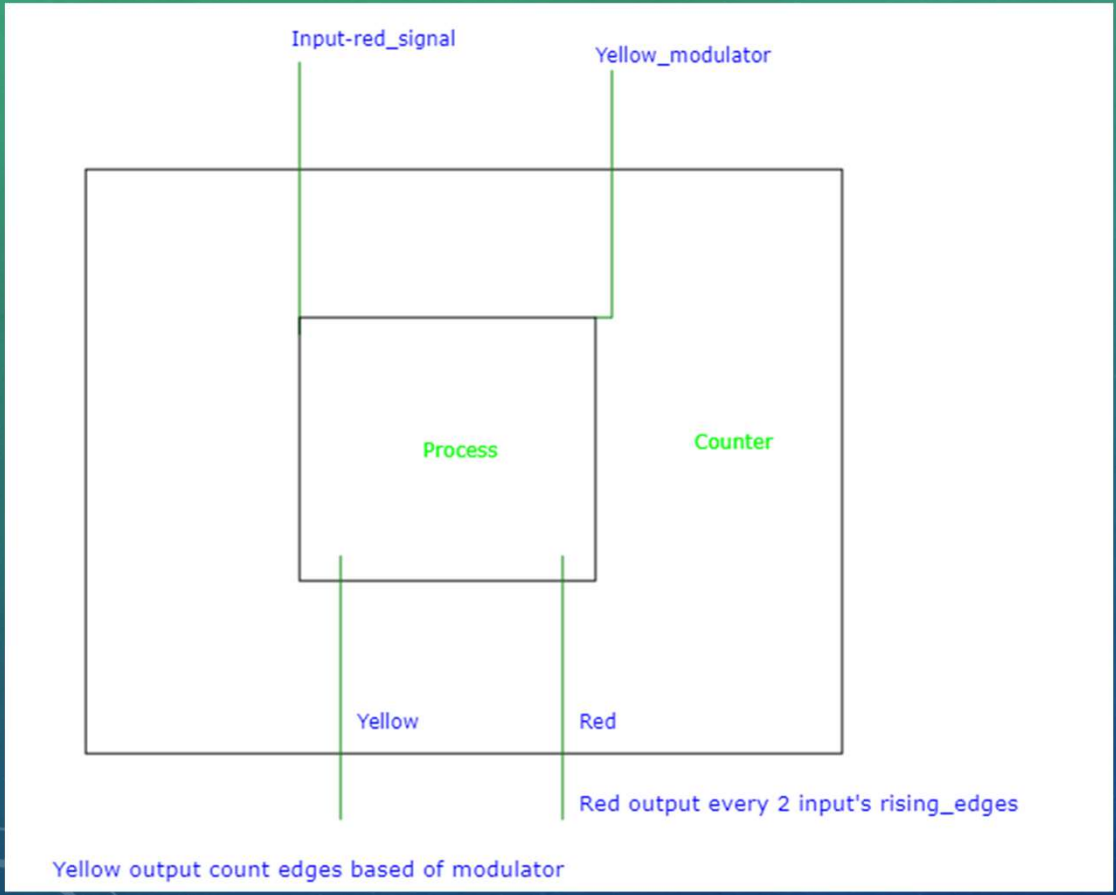


- Nominal(MOD3): red is on for 3 second and off for 5, green turn on when the red id off and vice-versa, yellow is on the last 2 second of green-light.
- Standby(MOD4): red and green are off, yellow is on 1 second and then off for 2.
- Maintenance(MOD5): green is turned on and off every half second, the others based on the value of their modulator signal.
  - Yellow_modulator = '00': half as red.
  - Yellow_modulator = '01': same as red.
  - Yellow_modulator = '10' or '11': twice as red.
  - red_modulator = '0': ½ second.
  - red_modulator = '1': 6 seconds.

SCHEMA OF ALL MODES

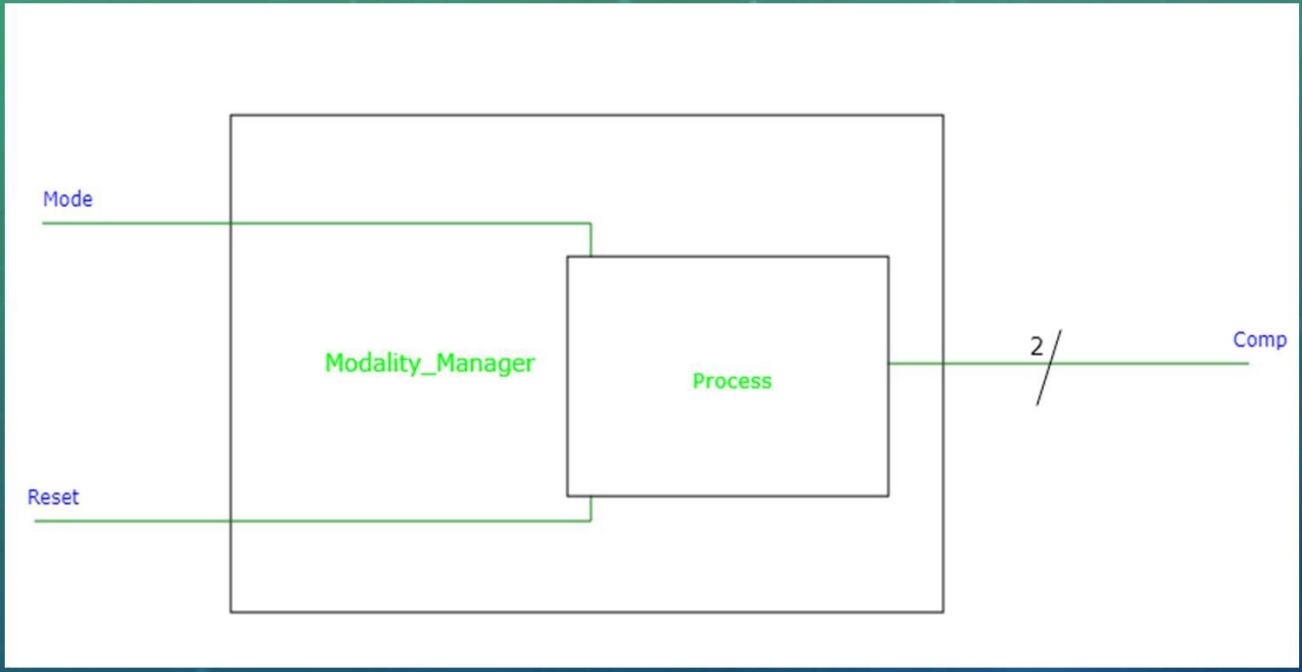
modulator/red yellow	0	1
00	Red: 0.5s Yellow: 0.25s	Red: 6s Yellow: 3s
01	Red: 0.5s Yellow: 0. 5s	Red: 6s Yellow: 6s
10	Red: 0.5s Yellow: 1s	Red: 6s Yellow: 12s
11	Red: 0.5s Yellow: 1s	Red: 6s Yellow: 12s

VHDL DESIGN OF ALL STAGES COUNTER



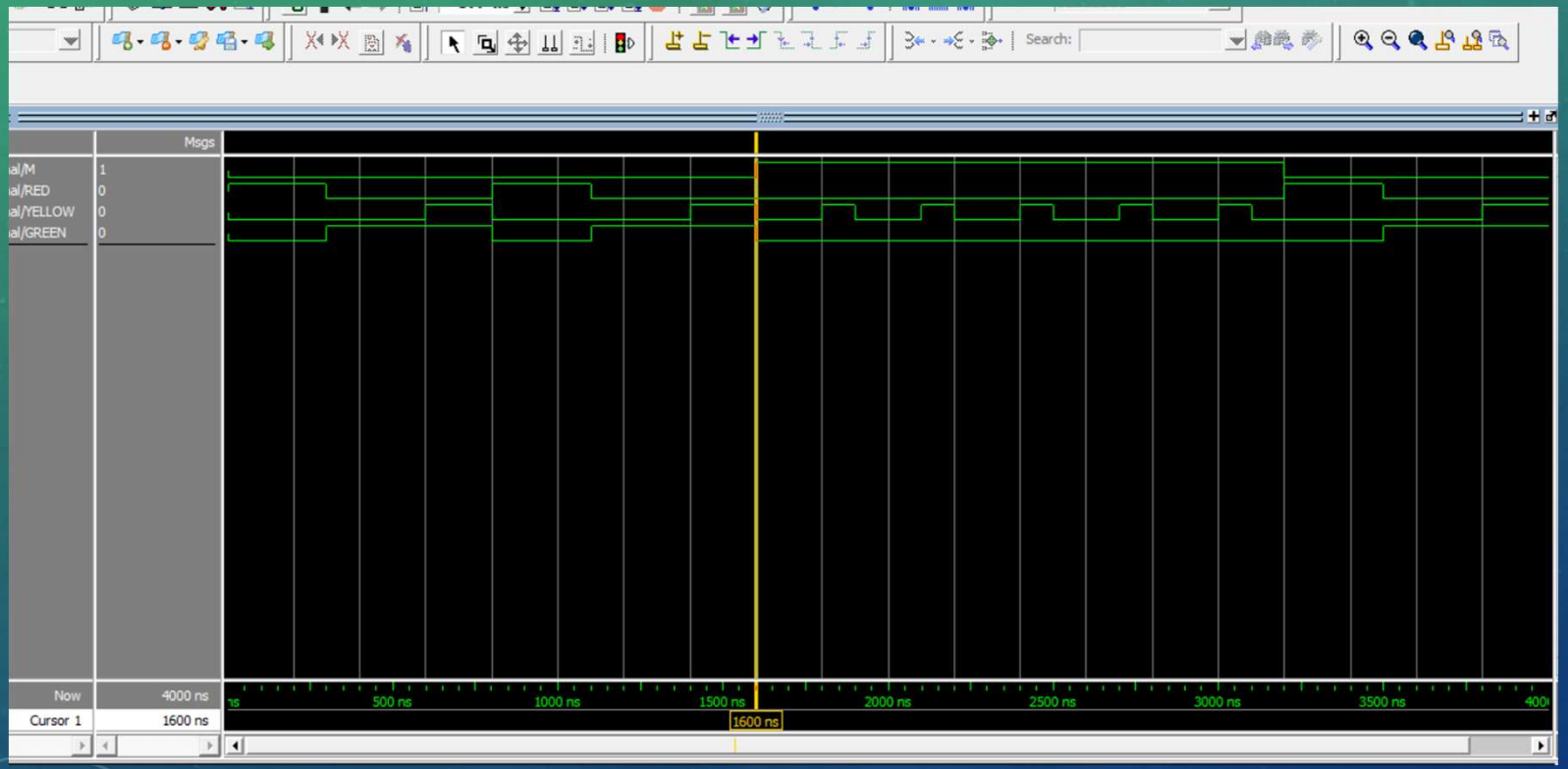
- Input = half red signal interval
- Process contains 2 counters:
  - for the red output signal (out every 2 rising edges of input)
  - Modulable for yellow output signal, that count 1,2 or 4 rising edges depending on the yellow_modulator input value

VHDL DESIGN OF ALL STAGES MODALITY_MANAGER



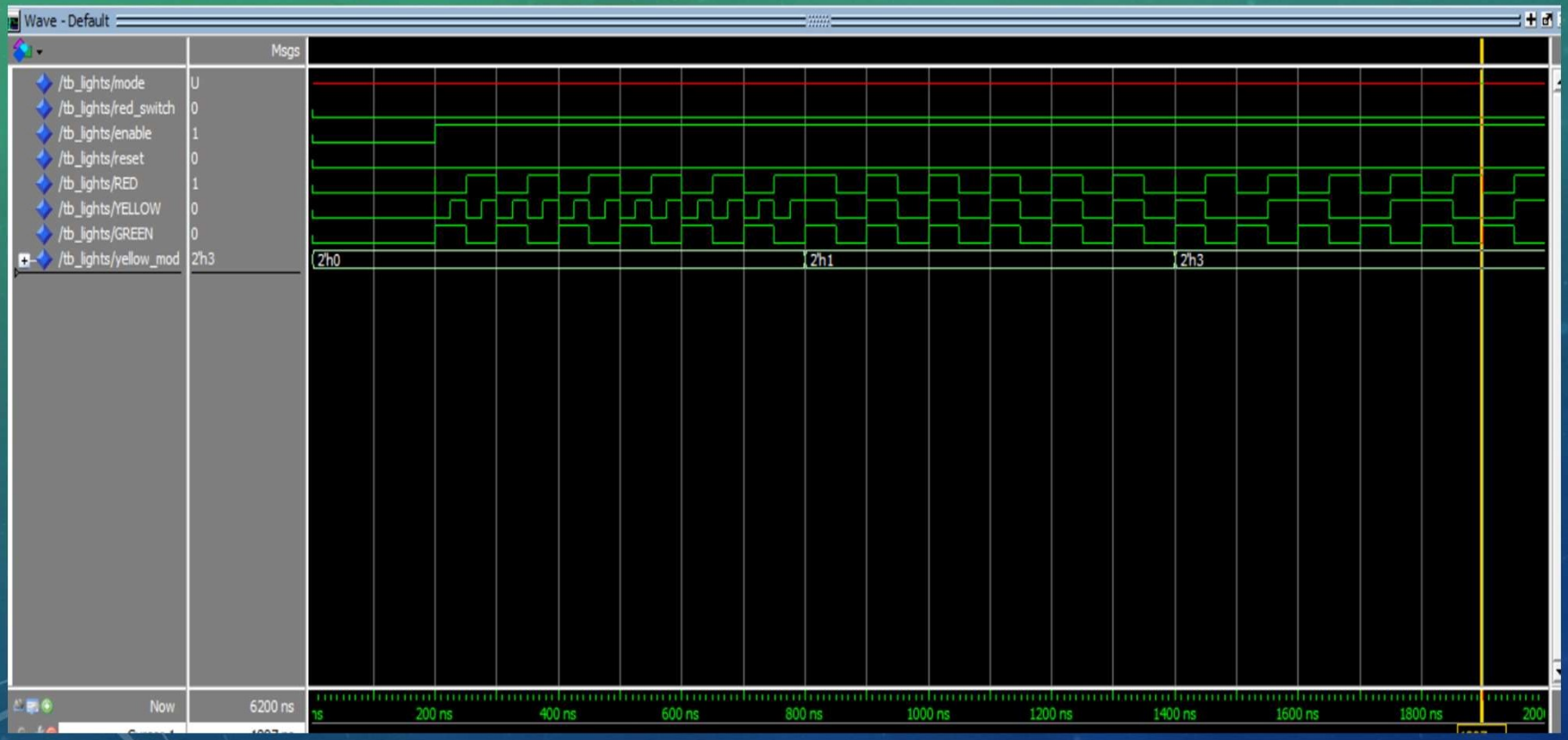
- Mode = input button mode
- Reset = input button reset
- Process a “switch case” for the output (Comp):
  - Starts at “00”.
  - Become “01” if “mode” change to value ‘0’.
  - Become “10” if “mode” change to value ‘1’.
  - Become “11” if “reset” have a downslope.
  - If “mode” change value when “reset” is “released(downlslope)”, comp = “11”.

SIMULATION RESULTS



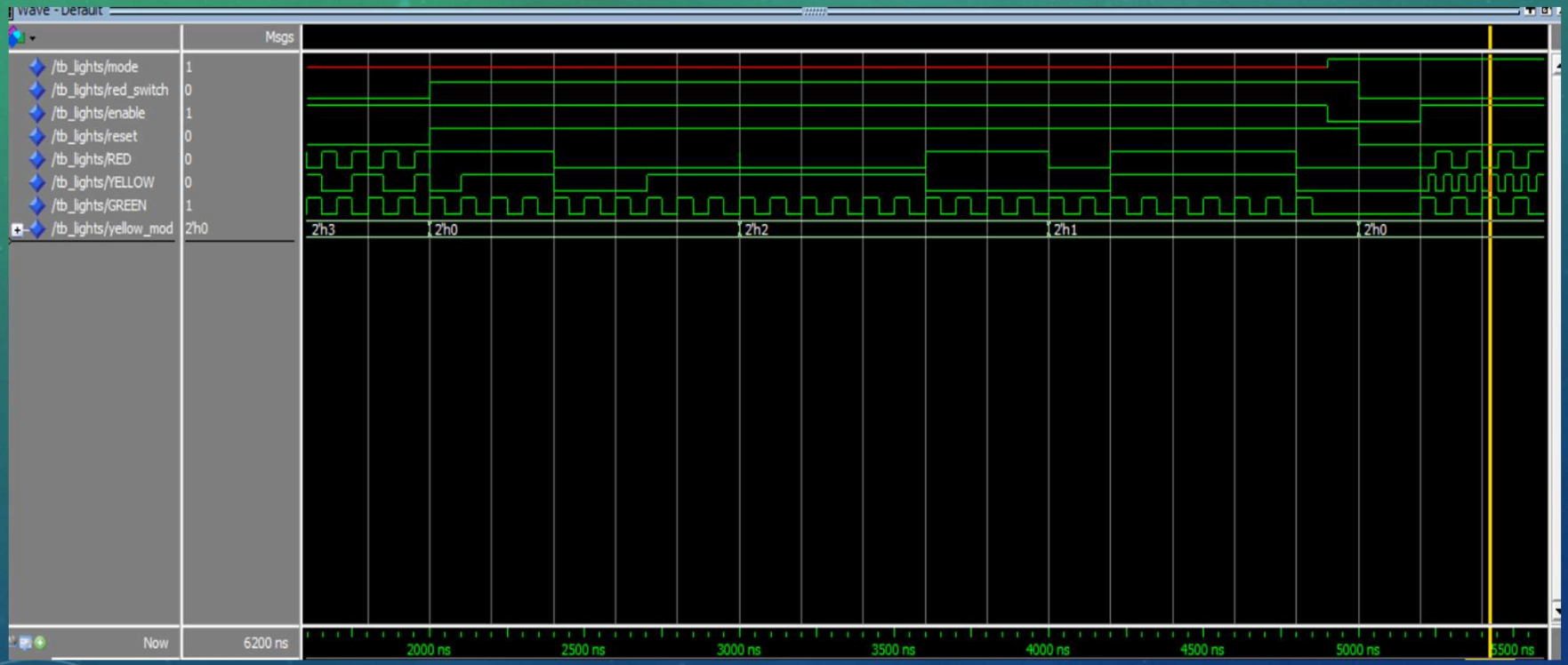
As can be seen here, during normal(MOD3 and MOD4) behavior in an intermediate test, the lights are on and off as described above, depending by the value of “Mode” input.
Prese, remember that 1 second have been transformed in 100 nanoseconds for test reasons.

SIMULATION RESULTS



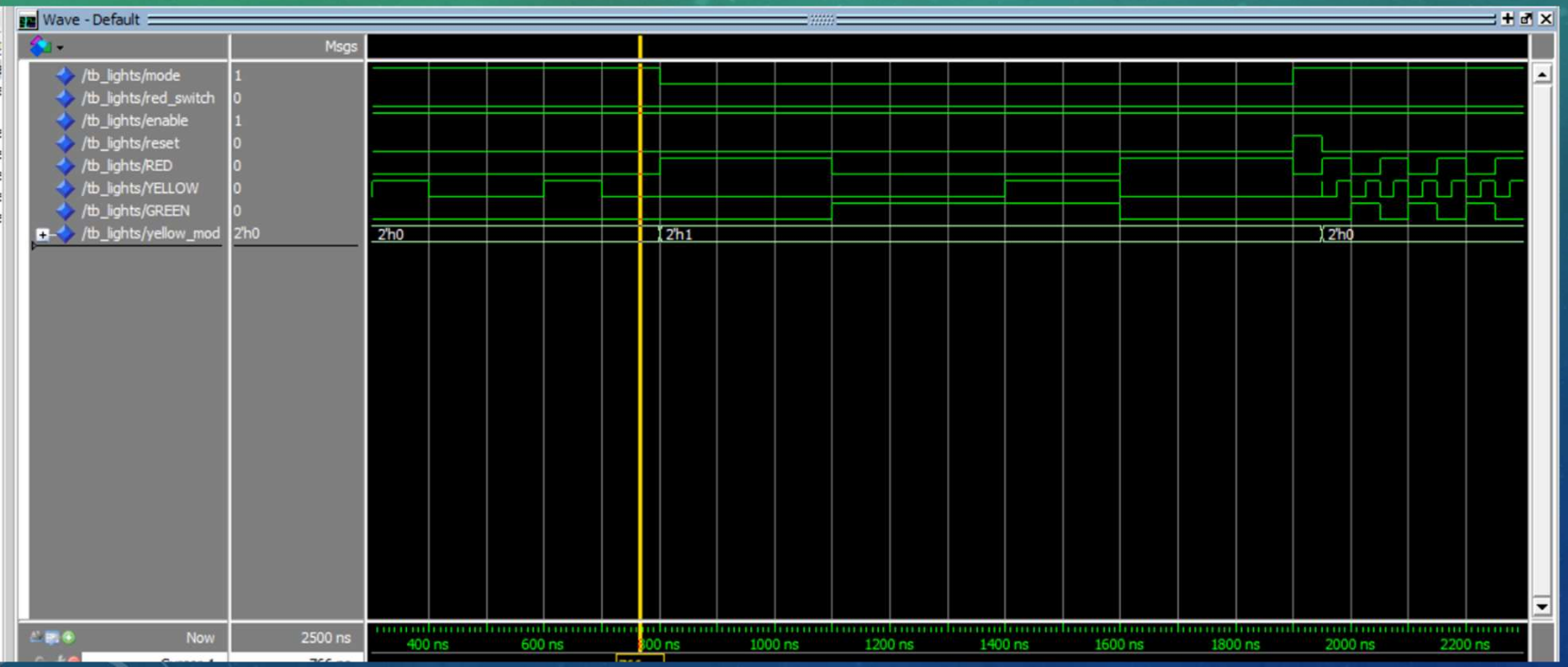
In this image you can see that in **maintenance** mode, as yellow_mod(ulator) change value, Intervals of yellow signals (YELLOW) are altered remaining dependent to RED signal (stable at 50ns with GREEN signal) . Also is possible to see at the beginning of the execution how all output signals are in AND with “enable”.

SIMULATION RESULTS



In this image you can see the same condition as the previous slide; except that red signal output has increased to 600ns (simulating 6S) and YELLOW followed;
As long as red_switch signal (red_modulator) have value of 1.

SIMULATION RESULTS



On the left is possible to see a simulation of "Normal behavior": **standby** since 800ns followed by **nominal**(300ns RED followed by 500ns GREEN and 200ns YELLOW with the last 200ns of GREEN) since 1950ns, when **maintenance** mode starts.

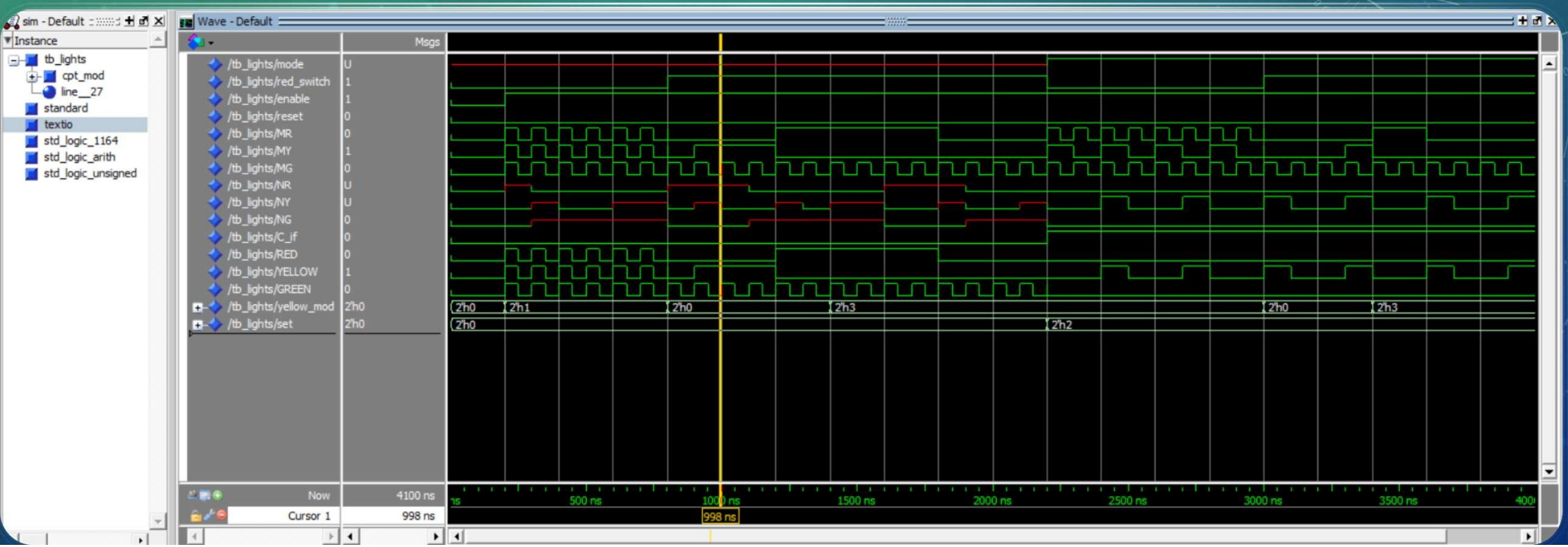
SIMULATION RESULTS

The following images are 2 screenshots taken with “debug variable”:

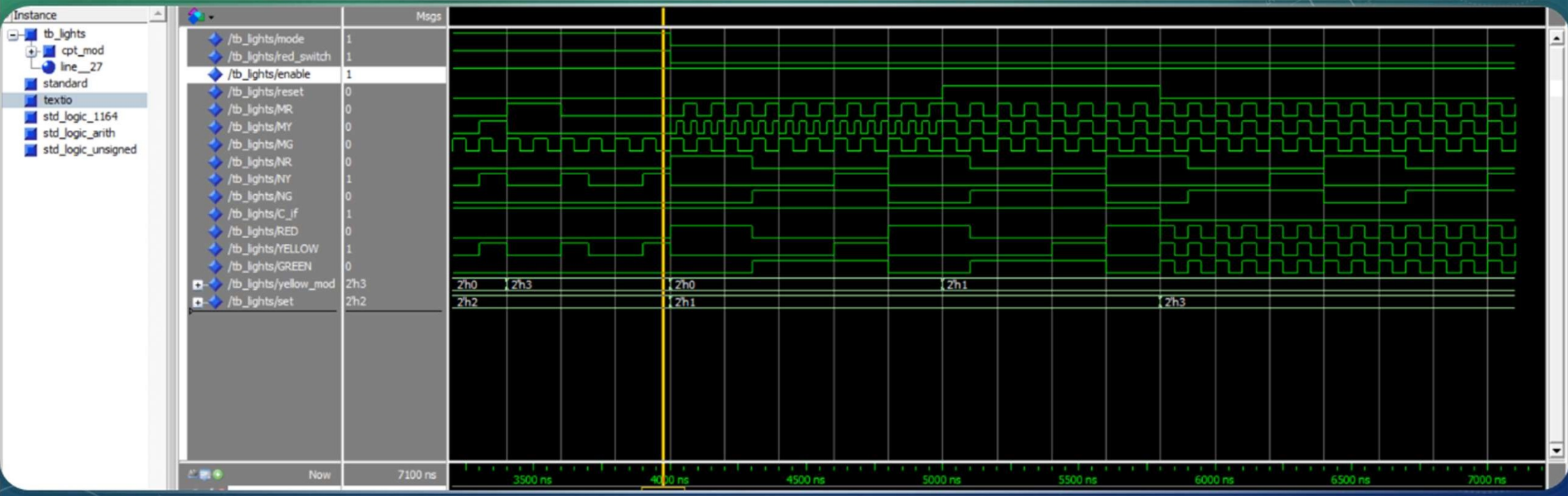
- MR: **Maintenance** mode value for RED signal
- MY: **Maintenance** mode value for YELLOW signal
- MG: **Maintenance** mode value for GREEN signal
- Set: value of “comp” (variable that dictate active mode)

- NR: **Normal** mode value for RED signal
- NY: **Normal** mode value for YELLOW signal
- NG: **Normal** mode value for GREEN signal

[**Normal** means **Nominal** or **Standby** mode, based by the current value of “mode” input signal]



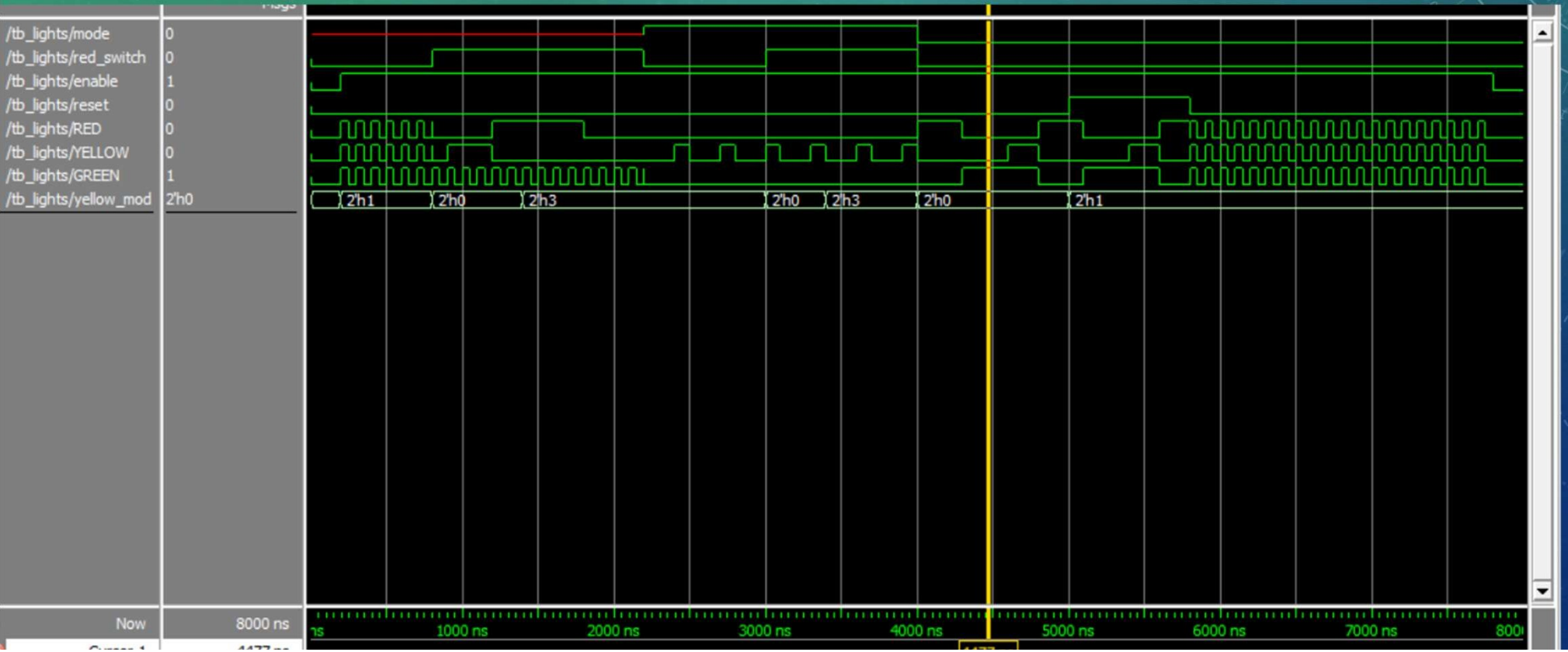
SIMULATION RESULTS



SIMULATION RESULTS

An Example of functioning with complete random input: a result of possible behavior, as can be seen, the lights work as intended also before any action on “Mode” input is performed.

- **Maintenance** since mode is undefined (lights still not active)
- Followed by **Standby** since 4000ns.
- Then **Nominal** since maintenance again at 5600ns, when “reset” input have a downslope.
- At 7300ns, signals are disabled.



CONCLUSION

- Each component in the top-view block scheme has been implemented in VHDL.
- Each entity has been tested independently or with a «supercomponent» before being connected, the **Simulation results** section include the test of all components connected.
- Since nowere was written that the counter has to reset when mode function change, I opted to take all the 3 circuits(MODn) running and abiltate only 1 exit corrisponding to the mode-circuit.
- I gave priority to «resetting» the circuit if one of the buttons to switch it to «**normal** mode» if pressed when «reset» input is released.
- I considered the instruction «blink every X time» as On for X time, then Off for X time, then On again… .
- The Names fot Mod4 and Mod5 have been extrapolated from the context and put as **standby** and **maintenence** respectivly.
- As written at the beginning, for testing purpose every 1 second of «requested time» have been converted as 100 ns.
- As shown by the simulation results, and according to the initial assumptions, the circuit works as expected.