FINAL PROJECT – INFORMATICA INDUSTRIALE

Group 2:

- Mattia Vincenzi (860579)
- Francesco Porto (816042)

Link to GitHub repository: https://github.com/HopedWall/LightControlSystem

OUTLINE

- Introduction
- Symbol
  - I/O ports
- Top-view block scheme
  - Internal signals
  - VHDL Design of all stages
- Simulation results
- Conclusion

OUTLINE

- **Introduction**
- Symbol
  - I/O ports
- Top-view block scheme
  - Internal signals
  - VHDL Design of all stages
- Simulation results
- Conclusion

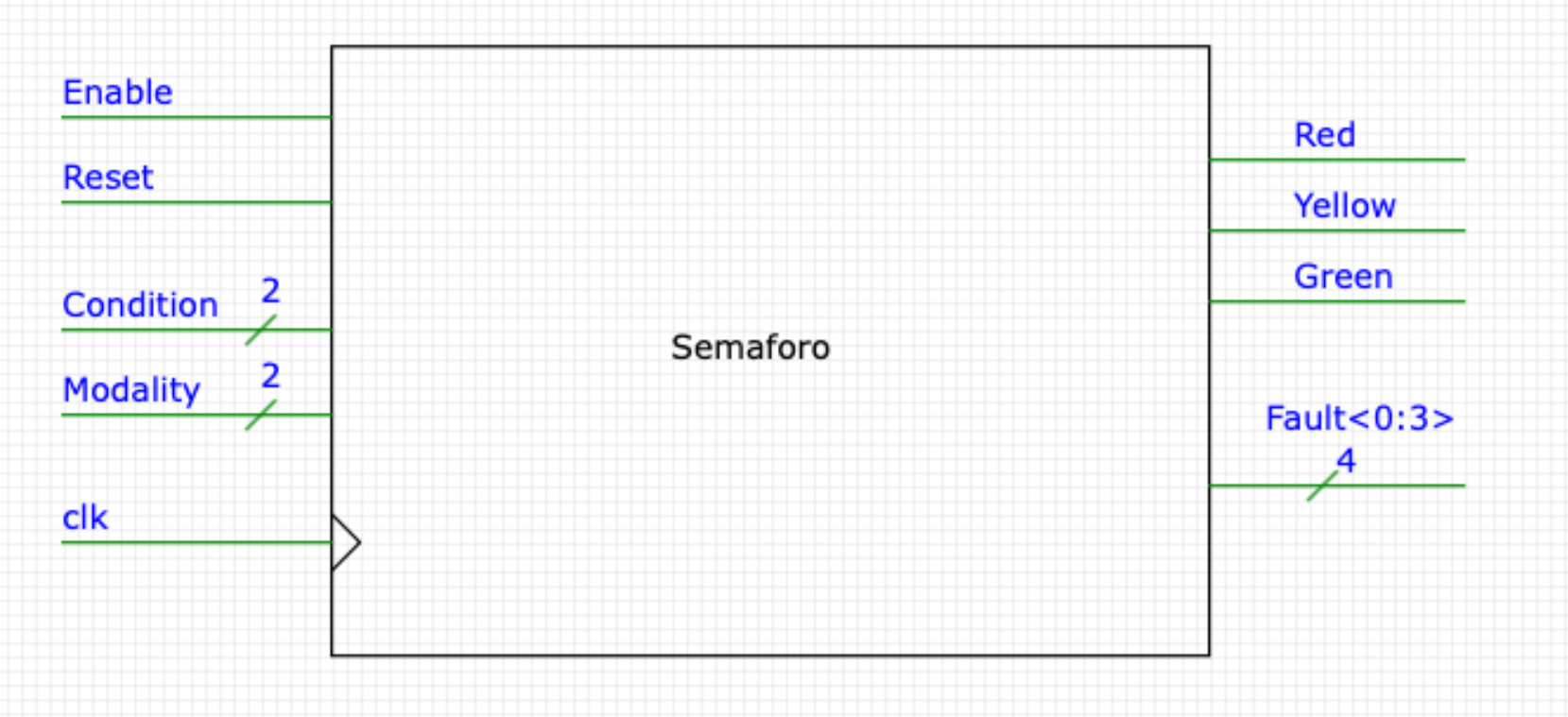
INTRODUCTION

- Initial assumptions:
  - Red will be modulated along with Green
    - (i.e. setting MOD12 makes **both** Red and Green last 12 seconds)
  - We decided to add an initial **OFF state**
    - It can only be left by setting the MAINTENANCE state
  - A **FAULT** signal has been added to report errors (due to setting conflicting inputs)
  - We decided to set the clock period to 1 sec (1 Hz)
    - However, for testing purposes, we set it to 20 ns
    - Simulating a 1 sec clock period with ModelSim was too expensive

OUTLINE

- Introduction
- **Symbol**
  - I/O ports
- Top-view block scheme
  - Internal signals
  - VHDL Design of all stages
- Simulation results
- Conclusion

SYMBOL


Enable
Reset
Condition
2
Modality
2
clk
Semaforo
Red
Yellow
Green
Fault<0:3>
4

OUTLINE

- Introduction
- **Symbol**
  - **I/O ports**
- Top-view block scheme
  - Internal signals
  - VHDL Design of all stages
- Simulation results
- Conclusion

I/O PORTS

Name	Direction	Resolution	Comment
Enable	Input	1	Active High, Synchronous.
Reset	Input	1	Active Low, Asynchronous.
Condition	Input	2	00 -> Maintenance 01 -> Nominal 11 -> Standby 10 not used
Modality	Input	2	00 -> MOD5 01 -> MOD12 11 -> MOD15 10 not used
Clk	Input	1	

I/O PORTS

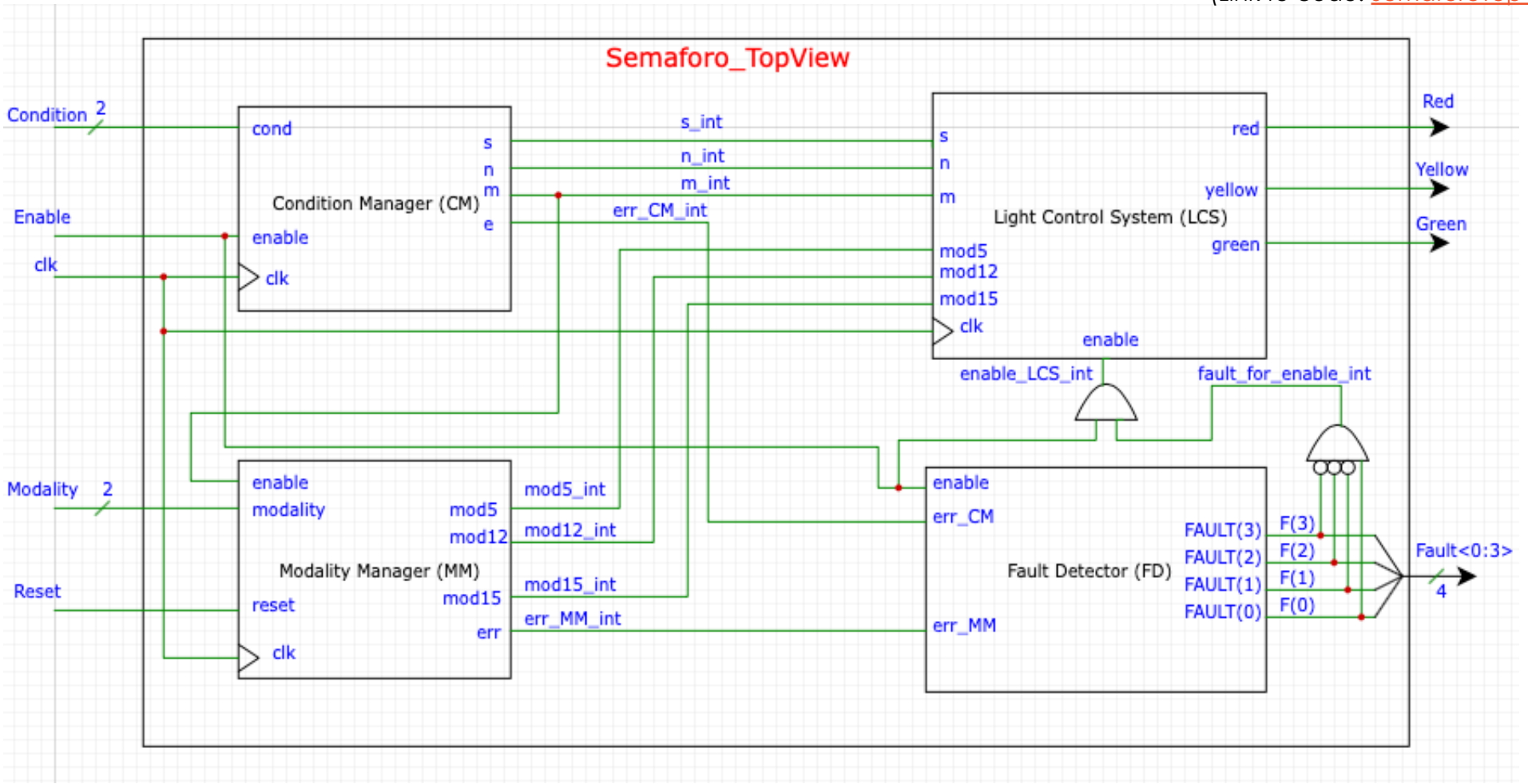
Name	Direction	Resolution	Comment
Red	Output	1	
Yellow	Output	1	
Green	Output	1	
Fault	Output	4	0000 -> Enable is set to 0 0001 -> Everything working correctly 0010 -> Condition error 0100 -> Modality error 1000 -> Condition AND Modality error

OUTLINE

- Introduction
- Symbol
  - I/O ports
- **Top-view block scheme**
  - Internal signals
  - VHDL Design of all stages
- Simulation results
- Conclusion

TOP-VIEW BLOCK SCHEME

(Link to code: SemaforoTopView.vhdl)



OUTLINE

- Introduction
- Symbol
  - I/O ports
- Top-view block scheme
  - **Internal signals**
  - VHDL Design of all stages
- Simulation results
- Conclusion

INTERNAL SIGNALS

Name	Resolution	Comment
mod5_int	1	Connects MM and LCS. It is 1 when mod5 is set.
mod12_int	1	Connects MM and LCS. It is 1 when mod12 is set.
mod15_int	1	Connects MM and LCS. It is 1 when mod15 is set.
err_MM_int	1	Connects MM and FD. It is 1 when the input modality is 10.
m_int	1	Connects CM and LCS. It is 1 when MAINTENANCE is set.
n_int	1	Connects CM and LCS. It is 1 when NOMINAL is set.
s_int	1	Connects CM and LCS. It is 1 when STANDBY is set.
err_CM_int	1	Connects CM and FD. It is 1 when the input condition is 10.
fault_for_enable_int	1	It is 1 when FAULT is 0001 (everything works correctly)
enable_LCS_int	1	Enables LCS. It is 1 iff fault_for_enable_int AND (global) Enable.

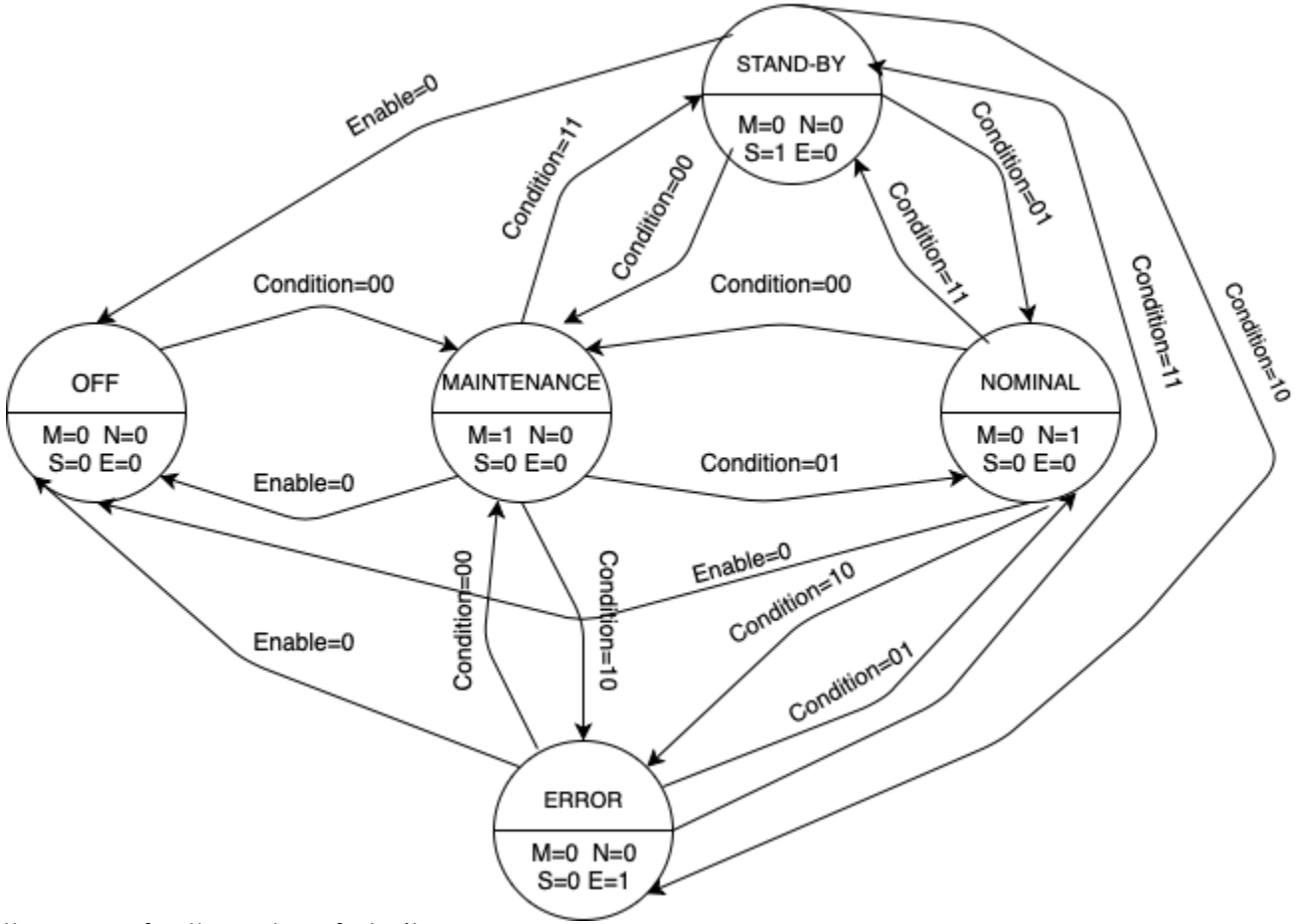
OUTLINE

- Introduction
- Symbol
  - I/O ports
- Top-view block scheme
  - Internal signals
  - **VHDL Design of all stages**
- Simulation results
- Conclusion

CONDITION MANAGER

(Link to code: ConditionManager.vhdl)

- It is a **Moore FSM**
- The starting state is **OFF,** which can be left only by setting Condition to 00, thus entering the **MAINTENANCE** state
- **OFF** can be reached again only by setting Enable=0 from each state
- **Gray code** has been used to minimize transitions to a unwanted states

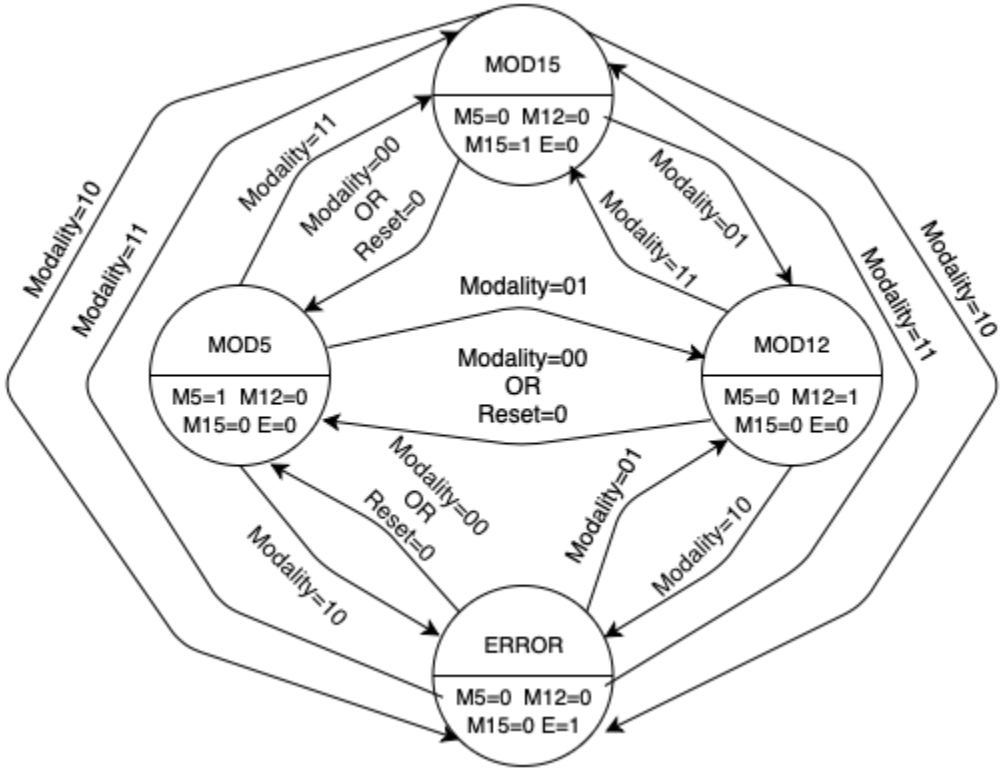


Note: We omitted the loops when ENABLE=1 or when the condition remains the same for the sake of clarity

MODALITY MANAGER

(Link to code: ModalityManager.vhdl)

- It is a **Moore FSM**
- The starting state is **MOD5**
- The modality can only be changed when the circuit is in the **MAINTENANCE** state
  - Otherwise the circuit ignores the modality change
- When Reset is set to 0 (Active Low) the FSM returns to MOD5

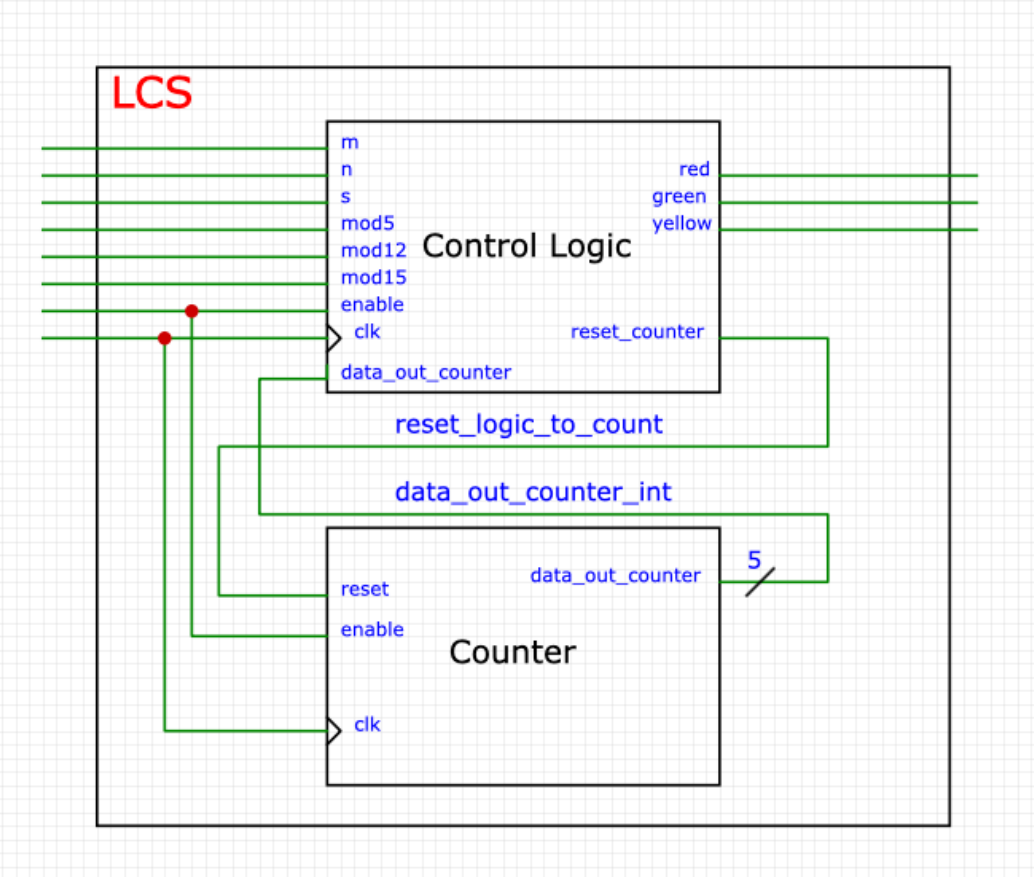


Note: We omitted the loops when RESET=1 or when the modality remains the same for the sake of clarity

LIGHT CONTROL SYSTEM

(Link to code: LightControlSystem.vhdl)

- It is made by two sub-components:
  - **Control Logic:** a process that sets the lights according to the current condition and modality.
  - **Counter (w/ Reset):** a simple counter that keeps track of time
- By using both components, it is possible to handle the timings for the three lights
- The two components cooperate as follows:
  - The Control Logic can reset the Counter
  - The Counter is read by the Control Logic



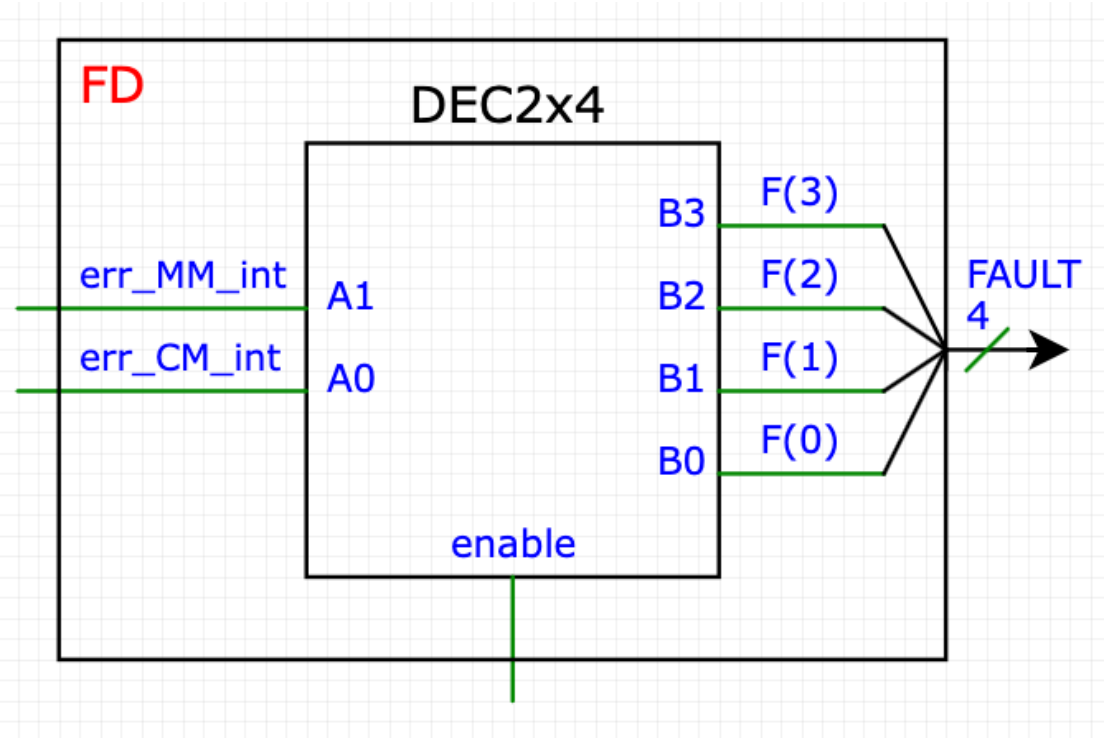
FAULT DETECTOR

(Link to code: FaultDetector.vhdl)

- Detects whether or not the circuit is working correctly
- Takes as input the error bit from CM and MM
- Uses a **2x4 Decoder** for returning a bit sequence that represents the error type
- It is a **combinatory component**
- When an error is found, the **Enable** for the LCS is set to 0

Truth Table

enable	A1	A0	B3	B2	B1	B0
0	x	x	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

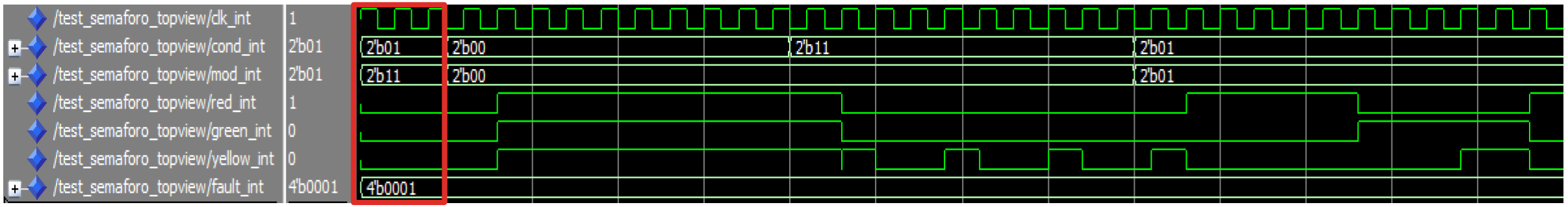


OUTLINE

- Introduction
- Symbol
  - I/O ports
- Top-view block scheme
  - Internal signals
  - VHDL Design of all stages
- **Simulation results**
- Conclusion

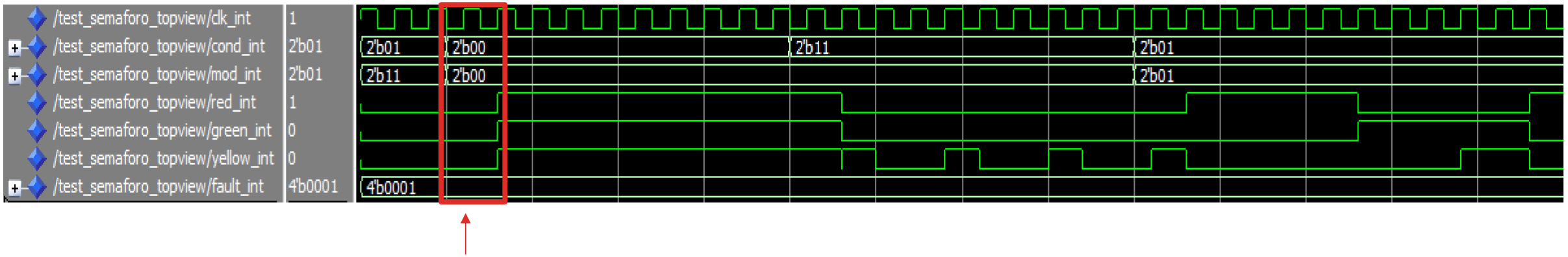
SIMULATION RESULTS

Scenario #1: Everything working correctly



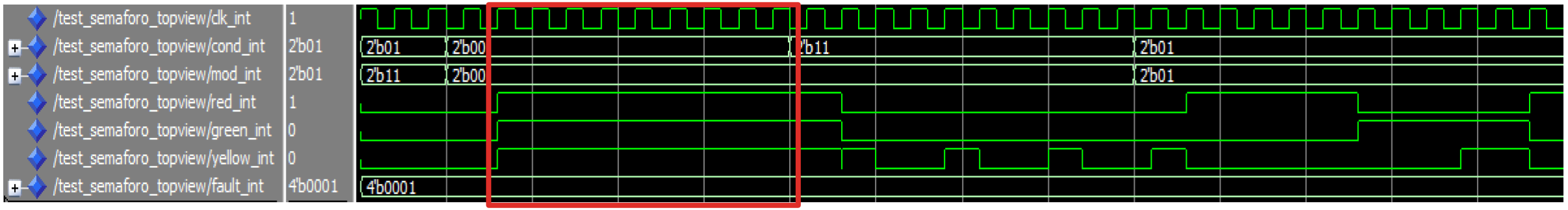
OFF

SIMULATION RESULTS



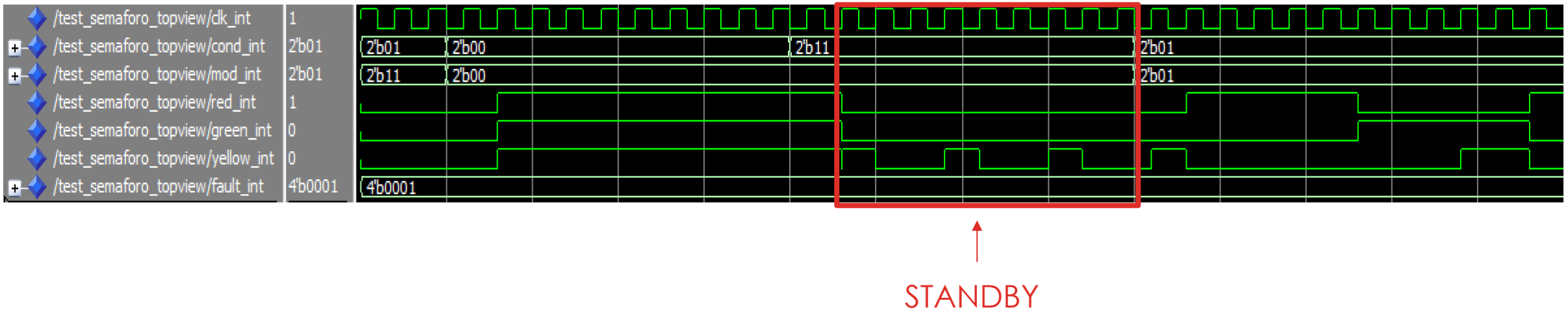
Delay between internal state change and lights update

SIMULATION RESULTS

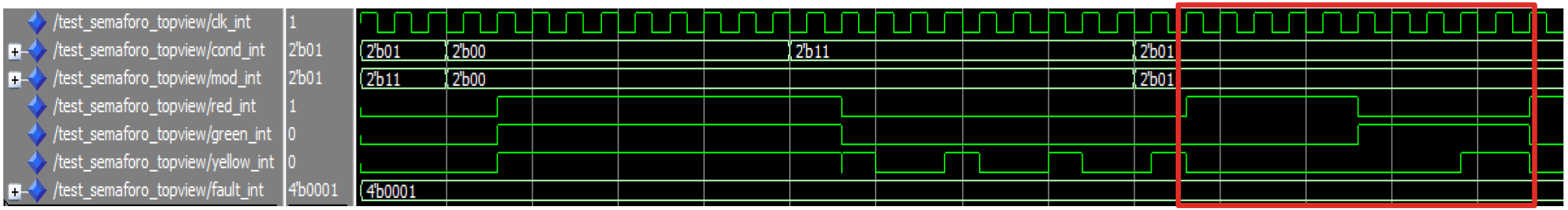


MAINTENANCE

SIMULATION RESULTS

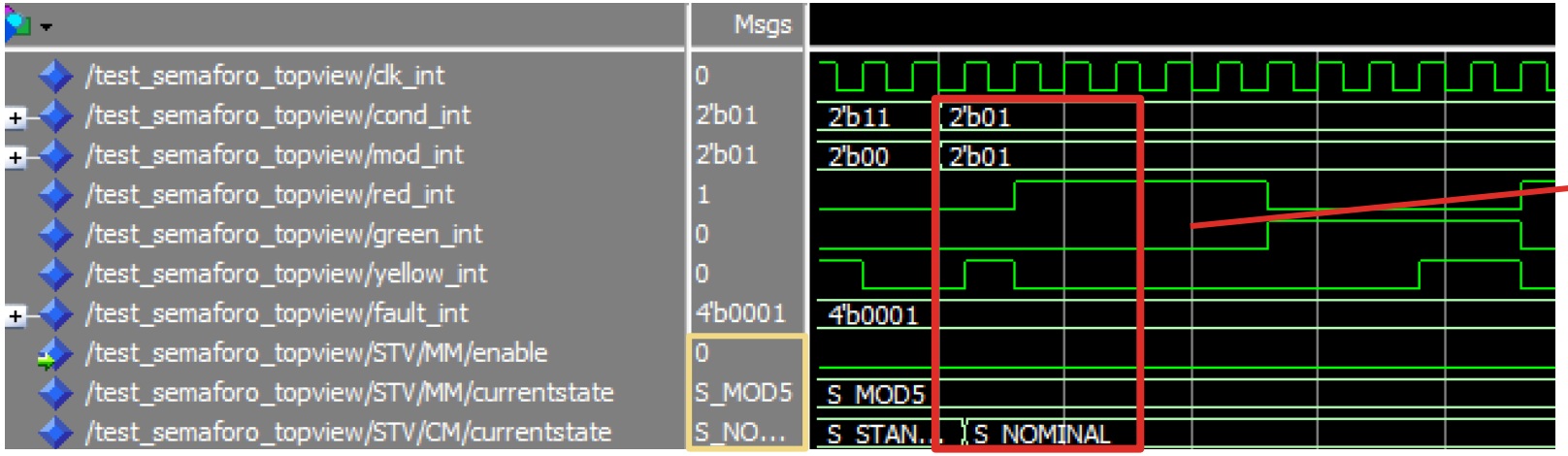


SIMULATION RESULTS

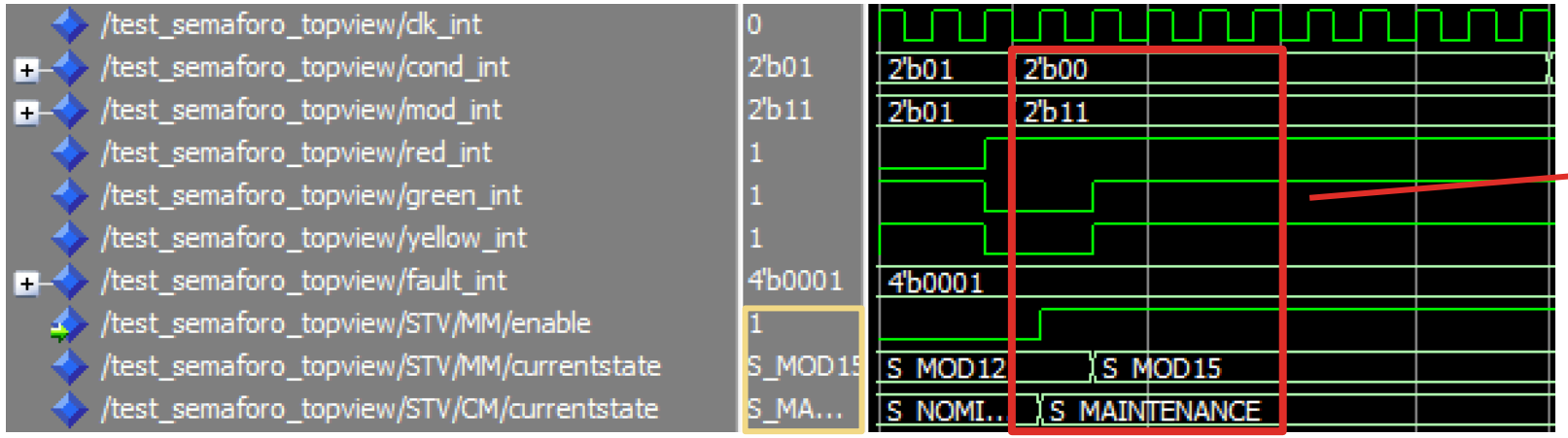


NOMINAL
(MOD-5)

SIMULATION RESULTS

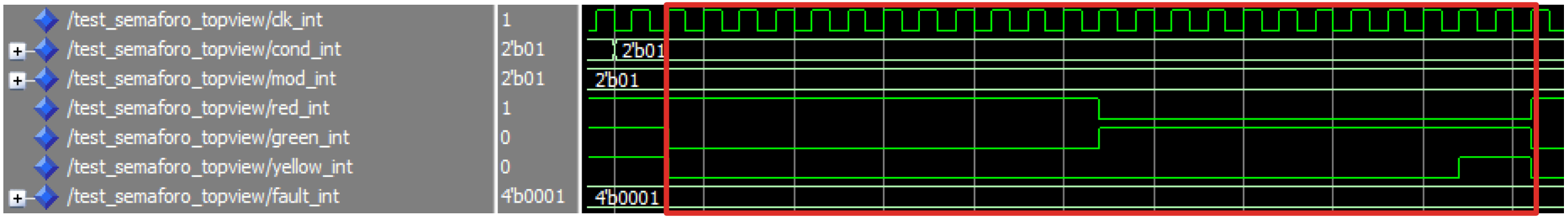


The modality does not change since the traffic light is not in the MAINTENANCE state, therefore the Enable for Modality Manager is 0



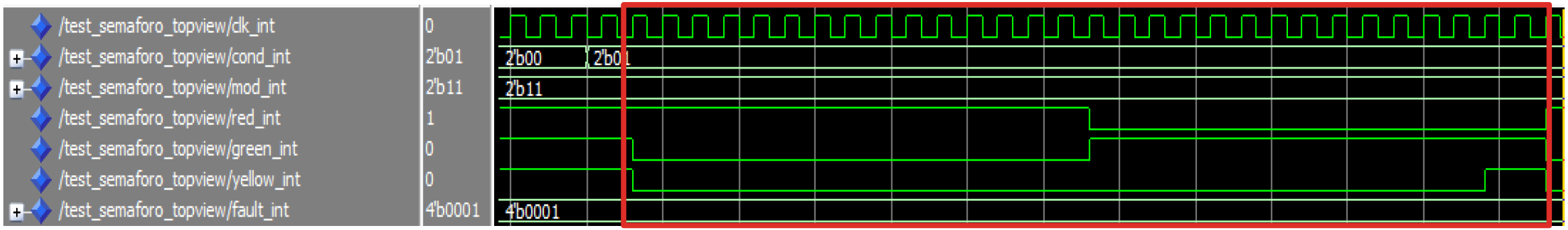
The modality does change since the traffic light is in the MAINTENANCE state, therefore the Enable for Modality Manager is 1

SIMULATION RESULTS



NOMINAL
(MOD-12)

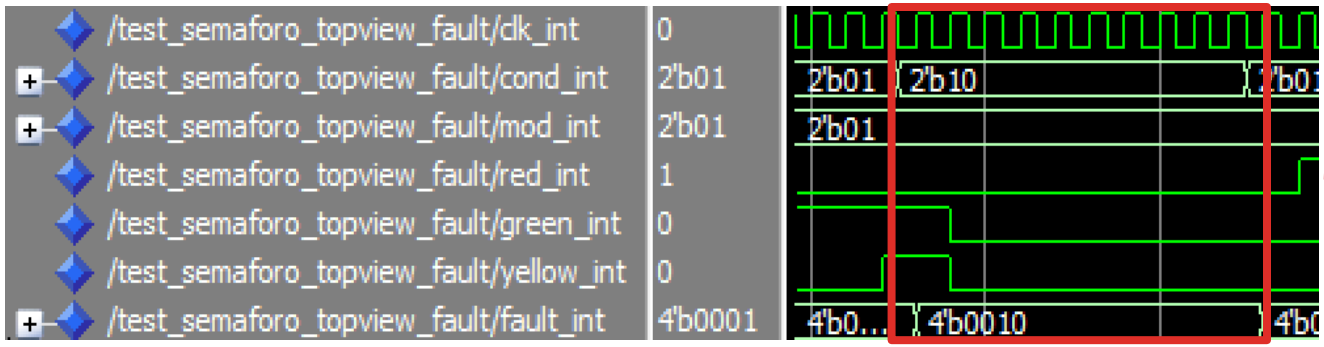
SIMULATION RESULTS



NOMINAL
(MOD-15)

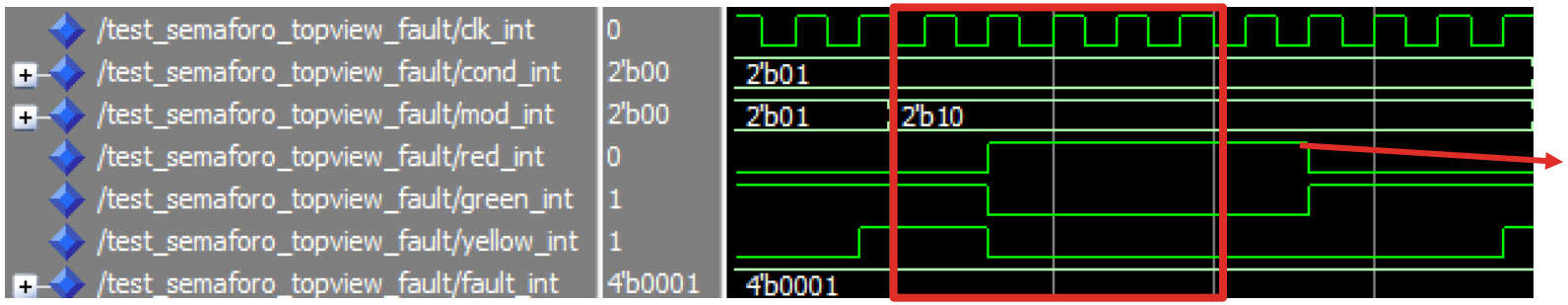
SIMULATION RESULTS

Scenario #2: FAULT Configuration



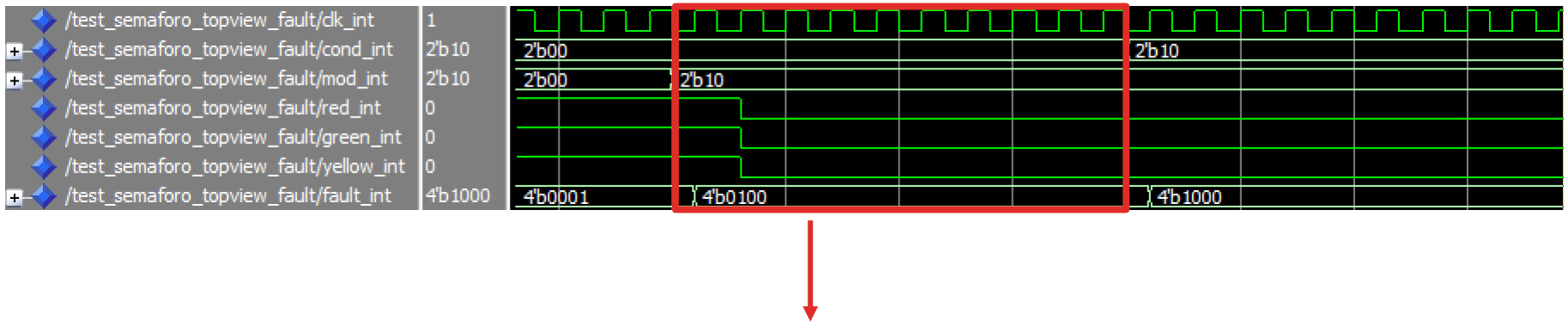
- 10 is an invalid input Condition, so Fault is set to 0010 that represents this type of error
- Lights are turned off until a valid Condition is set

SIMULATION RESULTS



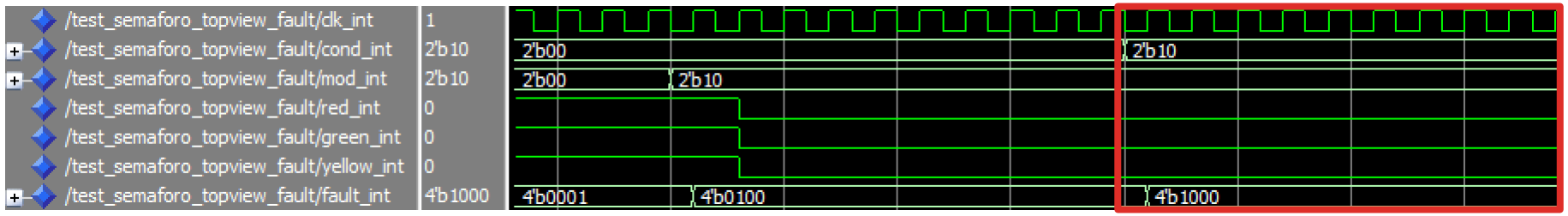
- 10 is an invalid input Modality, but it is ignored because the modality cannot be changed since the traffic light is not in the MAINTENANCE state

SIMULATION RESULTS



- 10 is an invalid input Modality, so Fault is set to 0100 that represents this type of error.
  - This is detected because the traffic light is in the MAINTENANCE state and therefore the Modality can be modified.
- Lights are turned off until a valid Condition is set

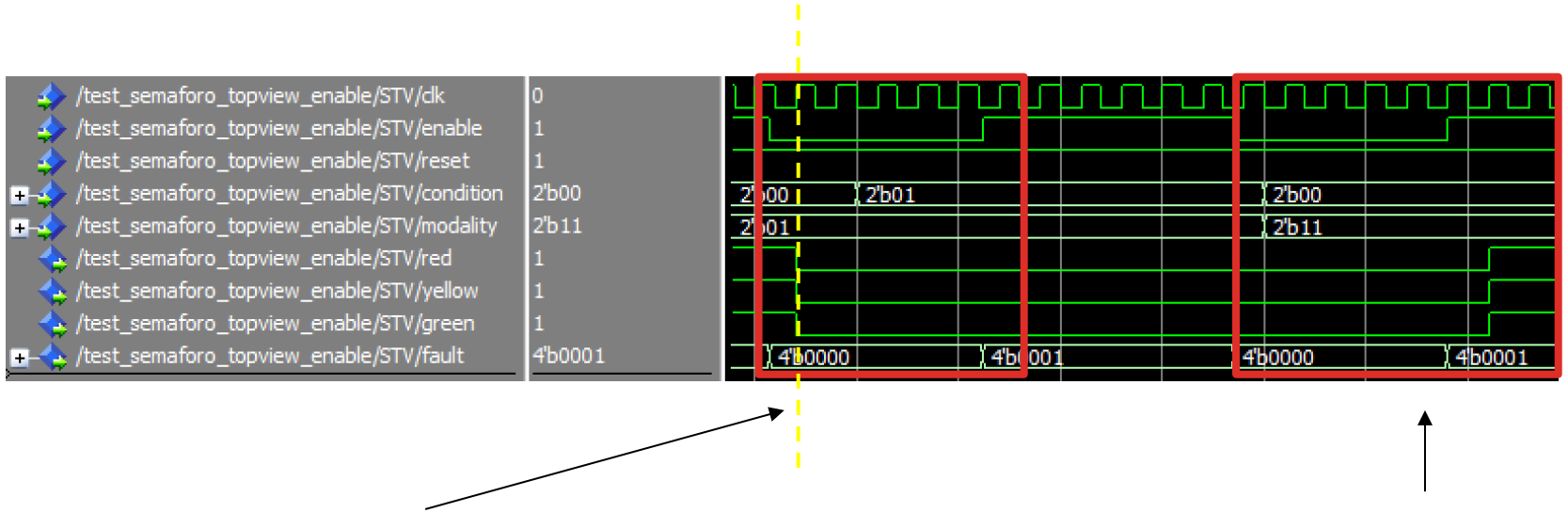
SIMULATION RESULTS



- 10 is an invalid input Condition
- 10 is an invalid input Modality
  - Fault is set to 1000 that represents an error on both
- Lights are turned off until a valid Condition is set

SIMULATION RESULTS

Scenario #3: Enable



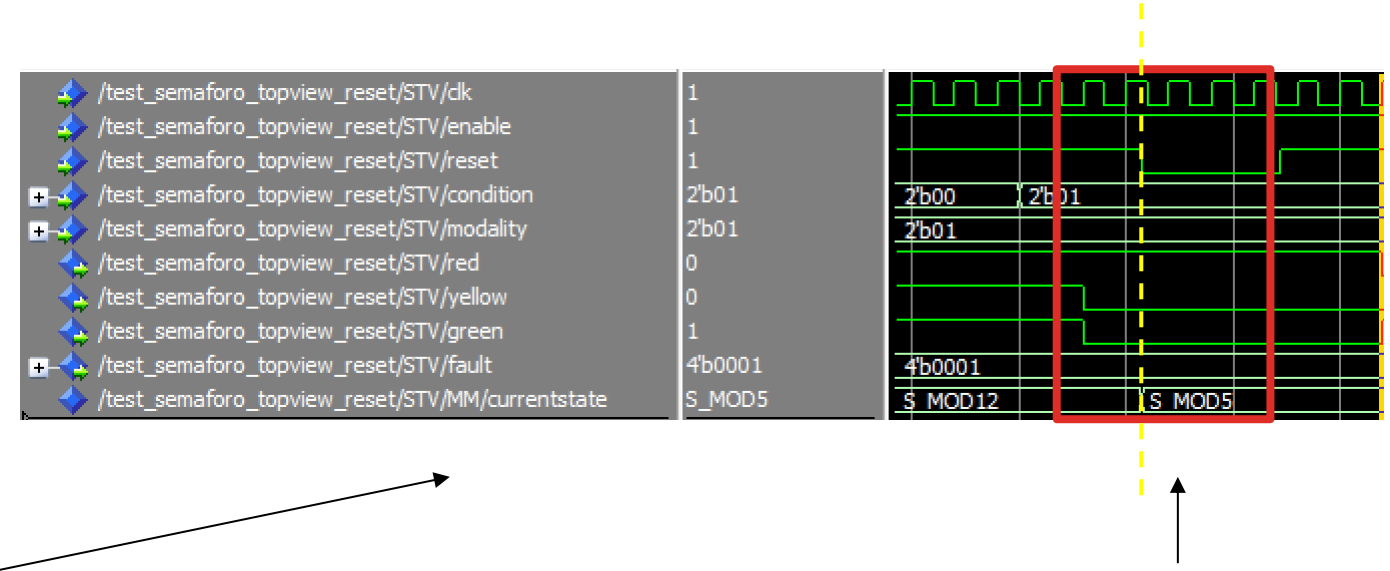
ENABLE:
- Active High
- Synchronous

All lights are off and it is not sensitive to condition changes

When ENABLE becomes 1 the OFF condition can be changed only by setting MAINTENANCE

SIMULATION RESULTS

Scenario #4: Reset



RESET:

- Active Low
- Asynchronous

When RESET becomes 0 the modality becomes MOD5

SIMULATION RESULTS

Scenario #5: Enable and Reset

- RESET (Asynchronous) is prioritary over ENABLE (Synchronous)
- ENABLE = 0 makes the circuit insensitive to external signals and turns all the lights off
  - But when RESET = 0, the modality changes nonetheless

OUTLINE

- Introduction
- Symbol
  - I/O ports
- Top-view block scheme
  - Internal signals
  - VHDL Design of all stages
- Simulation results
- **Conclusion**

CONCLUSION

- Starting by the top-view block scheme, each component has been implemented in VHDL
- Each entity has been tested independently with its own testbench, before being connected
- As shown by the simulation results, and according to the initial assumptions, the circuit works as expected

OUTLINE

- Introduction
- Symbol
  - I/O ports
- Top-view block scheme
  - Internal signals
  - VHDL Design of all stages
- Simulation results
- Conclusion