Computer Organization 2019

HOMEWORK 1 Traffic Light

Due date:

Overview

這次作業的主要目的為複習 verilog 語言和熟悉開發環境，請實作出紅綠燈的簡單 verilog 模組，運用有限狀態機的設計方式來設計本作業。

This homework aims to review over Verilog and gets you familiar with the working environment. Please implement a simple Traffic Light using finite state machine (FSM).

General rules for deliverables

- You need to complete this homework INDIVIDUALLY. You can discuss the homework with other students, but you need to do the homework by yourself. You should not copy anything from someone else, and you should not distribute your homework to someone else. If you violate any of these rules, you will get NEGATIVE scores, or even fail this course directly
- When submitting your homework, compress all files into a single **zip** file, and upload the compressed file to Moodle.
  - Please follow the file hierarchy shown in Figure 1.

F740XXXXX (your id) (folder)

src(folder) * Store your source code

report.docx (project report. The report template is already included. Follow the template to complete the report.)

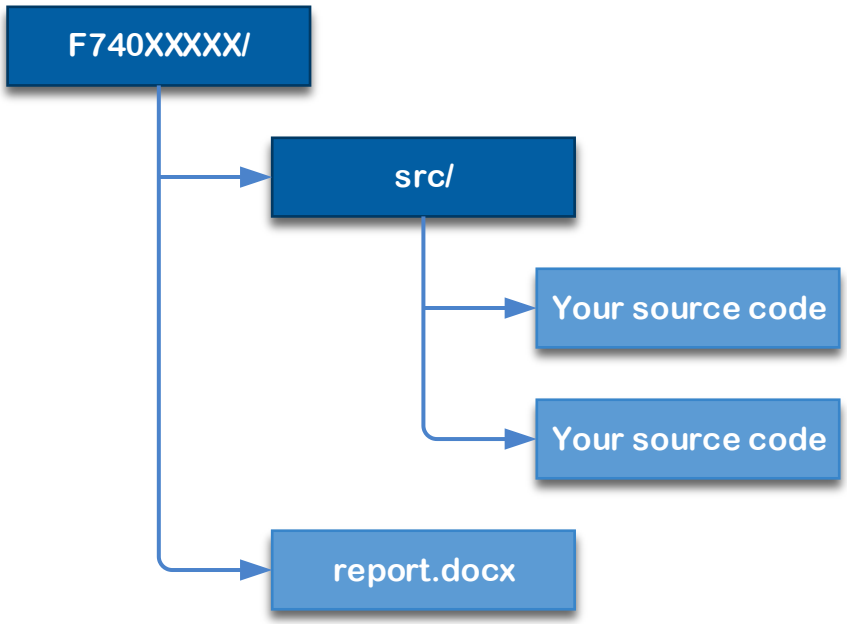

Figure 1. File hierarchy for homework submission

- **Important!** DO NOT submit your homework in the last minute. Late submission is not accepted.
- You should finish all the requirements (shown below) in this homework and Project report.

- If your code can not be recompiled by TA successfully using modelsim, you will receive NO credit.
- Verilog and SystemVerilog generators aren't allowed in this course.

Homework Description

紅綠燈的狀態圖示意如下：

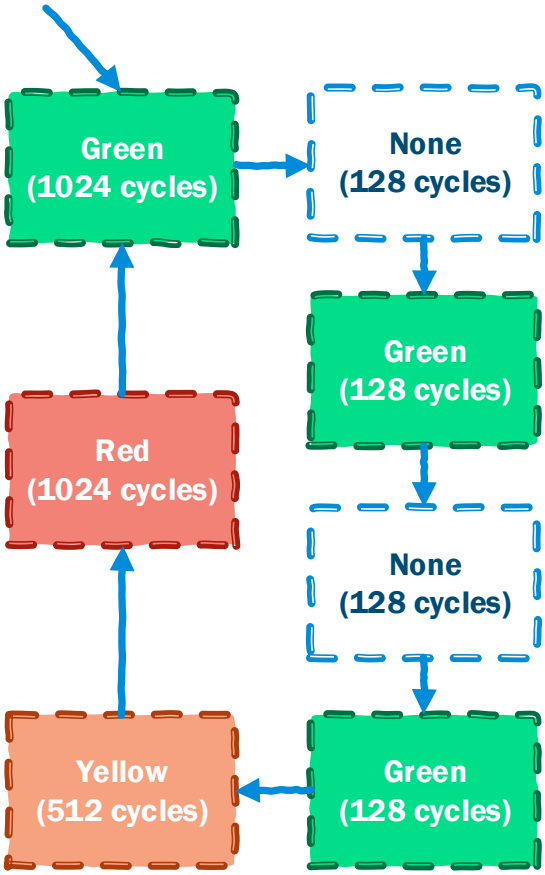

Figure 2. 紅綠燈狀態圖

The state diagram of the traffic light:

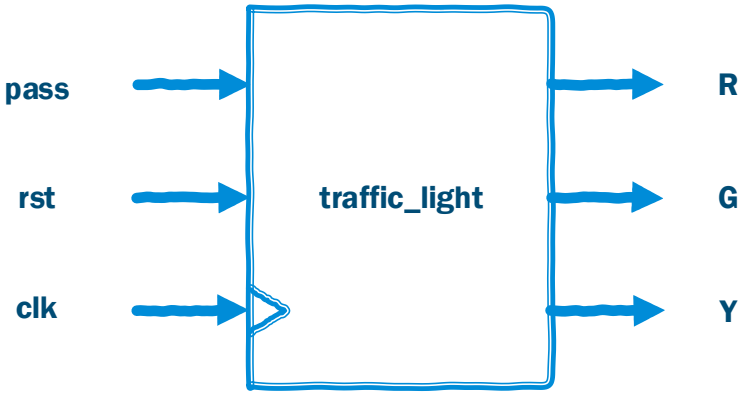

Figure 3. 紅綠燈控制模組的示意圖

The block diagram of the traffic light

作業規則如下：

1. 綠燈維持 1024 個 cycles。（起始狀態）
2. 沒有任何燈號維持 128 個 cycles。
3. 綠燈維持 128 個 cycles。
4. 沒有任何燈號維持 128 個 cycles。
5. 綠燈維持 128 個 cycles。
6. 切換成黃燈維持 512 個 cycles。
7. 再切換成紅燈維持 1024 個 cycles

輸入訊號：(電路為 clock **正緣觸發)**

pass：1bit 訊號，當 pass 為 1 時，若當前狀態非起始狀態之綠燈，強制切換成起始狀態之綠燈第 1 個 cycle，若原本為起始狀態之綠燈則不改變燈號和 cycle。當 pass 為 0 則沒有任何動作。

rst：1bit 訊號，非同步正緣時觸發，將燈號狀態設成綠燈第 1 個 cycle。

clk：1bit clock 訊號。

輸出訊號：

R：1bit 訊號，代表紅燈的輸出訊號。

G：1bit 訊號，代表綠燈的輸出訊號。

Y：1bit 訊號，代表黃燈的輸出訊號。

Specifications:

1. Green light lasts for 1024 cycles (initial state)
2. The state without any light lasts for 128 cycles
3. Green light lasts for 128 cycles
4. The state without any light lasts for 128 cycles
5. Green light lasts for 128 cycles
6. Switch to yellow light and last for 512 cycles
7. Switch to red light and last for 1024 cycles

Input signals: (the clock signal is positive-edge triggered)

1. pass: 1-bit signal.
   When pass == 1:
      If current state != the first green light state(the state with 1024 cycles):
         Switch to the 1st cycle of the first green light state
      Else:
         Nothing happens
   When pass == 0:
      Nothing happens.

2. rst: Asynchronous and positive-edge triggered 1-bit signal.
   When rst == 1: Switch current state to the 1st cycle of the first green light state (the state with 1024 cycles)
3. clk: 1-bit clock signal.

Output signals:

1. R: 1-bit signal, stands for the output signal of red light.
2. G: 1-bit signal, stands for the output signal of green light.
3. Y: 1-bit signal, stands for the output signal of yellow light.

Homework Requirements

1. 完成 traffic_light.v 的設計。
2. 用 modelsim 教學中的步驟，將 traffic_light.v 和 traffic_light_tb.v 放入 modelsim 專案中執行模擬。
3. 根據報告格式完成 report.docx，記得更改檔名。

1. Complete the design of traffic_light.v.
2. Following the Modelsim tutorial, put traffic_light.v and traffic_light_tb.v into Modelsim project and complete the simulation.
3. Complete report.docx according to the described format in hw1.pdf.