

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC BÁCH KHOA TP.HCM
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



BÁO CÁO BÀI TẬP LỚN
MÔN THIẾT KẾ LUẬN LÝ VỚI HDL

GVHD: Nguyễn Xuân Quang

Nhóm: L03 – 8

Thành phố Hồ Chí Minh, tháng 5 năm 2022

BÁO CÁO BÀI TẬP LỚN

THIẾT KẾ LUẬN LÝ VỚI HDL

ĐỀ TÀI: Thiết kế hệ thống đèn giao thông ngã tư bằng ngôn ngữ Verilog HDL

GVHD: thầy Nguyễn Xuân Quang

Nhóm: L03 – 8

Danh sách thành viên:

<i>HỌ VÀ TÊN</i>	<i>MSSV</i>
Trần Tuấn Anh	2110759
Võ Thành Nhân	2111914

Mục lục

I. GIỚI THIỆU SƠ LƯỢC ĐỀ TÀI	4
II. CƠ SỞ LÝ THUYẾT VÀ ỨNG DỤNG	5
A. Cơ sở lý thuyết	5
1. Verilog HDL:	5
2. Tổng quan về phần mềm Vivado:	5
B. Ứng dụng	7
III. THIẾT KẾ	7
A. Sơ đồ khối	7
B. Hệ thống led mô phỏng	8
C. Bảng trạng thái	10
D. Code Verilog HDL	10
IV. HIỆN THỰC	13
V. KẾT QUẢ	14
VI. KẾT LUẬN	14

I. GIỚI THIỆU SƠ LƯỢC ĐỀ TÀI

Ngày nay, cùng với sự phát triển của khoa học kỹ thuật, xã hội ngày càng văn minh, các đô thị ngày một hiện đại. Nhu cầu sử dụng đèn giao thông càng trở nên cấp thiết, nhất là trong các khu thành thị. Do nhu cầu của đời sống con người đặc biệt là nhu cầu đi lại, số lượng phương tiện đã tăng lên một cách chóng mặt. Riêng tại Việt Nam, số lượng xe máy trong những năm qua tăng một cách đột biến, mật độ xe lưu thông trên đường ngày càng nhiều, trong khi cơ sở hạ tầng còn nhiều hạn chế nên thường gây ra các hiện tượng ùn tắc giao thông và tai nạn giao thông hiện đang là vấn đề được nhiều người quan tâm. Vì lý do đó, các bộ luật giao thông lần lượt được ra đời và đưa vào sử dụng và dần trở nên phổ biến. Trong đó hệ thống đèn giao thông là công cụ điều khiển giao thông công cộng thực tế và có hiệu quả rất lớn trong việc đảm bảo an toàn giao thông và giảm thiểu tai nạn. Mô hình đèn tín hiệu giao thông với 3 màu xanh – vàng – đỏ đã được áp dụng cho hầu hết các nước trên thế giới, là một thiết bị quan trọng, không những an toàn cho các phương tiện mà còn giúp giảm ùn tắc giao thông vào giờ cao điểm. Nhận thấy sự thực tiễn của đề tài và để áp dụng những kiến thức đã học được trong môn Thiết kế luận lí với HDL, chúng em đã chọn đề tài “Thiết kế hệ thống đèn giao thông ngã tư” để làm bài báo cáo này. Đề tài nhằm thiết kế một hệ thống đèn giao thông tại ngã tư thông thường, bao gồm 2 đèn giao thông tại 2 hướng đi vuông góc. Hệ thống có chức năng điều khiển tín hiệu đèn. Các tín hiệu điều khiển được thể hiện bằng 2 đèn led RGB tương ứng với đèn tín hiệu xanh, đỏ, vàng và 1 led 7 đoạn đếm ngược thời gian.

II. CƠ SỞ LÝ THUYẾT VÀ ỨNG DỤNG

A. Cơ sở lý thuyết

1. Verilog HDL:

Là ngôn ngữ mô tả phần cứng (Hardware Description Language), là ngôn ngữ được sử dụng để mô tả hệ thống số, ví dụ như một máy tính hay linh kiện máy tính. Verilog HDL có thể được sử dụng để thiết kế hệ thống số ở nhiều mức khác nhau. Verilog HDL có thể định mô hình bố trí dây, điện trở, transistor trên một mạch tích hợp hoặc nó có thể mô tả các cổng logic, flip-flop trong hệ thống số. Thậm chí ở một mức cao hơn là mô tả thanh ghi và sự di chuyển dữ liệu giữa các thanh ghi gọi là mức chuyển thanh ghi (RTL: Register Transfer Level). Verilog hỗ trợ tất cả các cấp độ thiết kế số.

2. Tổng quan về phần mềm Vivado:

Vivado® Design Suite xuất hiện với nhiều giải pháp để hoàn thành thành tác vụ bao gồm thiết kế và kiểm tra FPGA. Thêm vào đó quá trình thiết kế RTL-to-bitstream, Vivado Design Suite cung cấp các quá trình tích hợp cấp hệ thống mới và tập trung trên thiết kế IP (intellectual property). IP khác nhau có thể được nhanh chóng cấu hình và được kết nối với nhau thành các thiết kế khối hệ thống phụ IP trong môi trường IP integrator của vivado. các thiết kế IP block và custom IP có thể được cấu hình, đóng gói và tạo có sẵn từ catalog IP của vivado. Kiểm tra và phân tích thiết kế được cho phép ở mỗi tầng của quá trình thiết kế. Các tính năng thiết kế bao gồm mô phỏng logic, I/O và clock planning, phân tích nguồn, xác định ràng buộc và phân tích timing, kiểm tra giá trị thiết kế – DRC, ảo hóa của thiết kế logic phân tích và thay đổi các giá trị thực thi, và lập trình và gỡ lỗi.

Toàn bộ giải pháp được tích hợp trong giao diện đồ họa người dùng GUI như môi trường thiết kế IDE vivado. IDE vivado cung cấp 1 giao diện tổng hợp, thực thi, và phê chuẩn thiết kế và IP. Thêm vào đó, tất cả quá trình có thể chạy dùng các lệnh Tcl. Các lệnh Tcl có thể tương tác dùng Tcl shell trên môi trường thiết kế của Vivado IDE. Bạn có thể dùng các kịch bản Tcl để chạy toàn bộ quá trình thiết kế, bao gồm phân tích thiết kế, hoặc chạy chỉ 1 phần của thiết kế.

Board Arty Z7 là một nền tảng phát triển sẵn sàng sử dụng được thiết kế dựa trên Hệ thống trên chip Zynq-7000™ của Xilinx. Kiến trúc Zynq-7000 tích hợp chặt chẽ bộ xử lý ARM Cortex-A9 lõi kép, 650

MHz với logic Xilinx 7-series Field Programmable Gate Array (FPGA). Việc ghép nối này mang lại khả năng của một bộ xử lý mạnh mẽ với một bộ thiết bị ngoại vi và bộ điều khiển được xác định bằng phần mềm độc đáo, có thể được điều chỉnh cho phù hợp với bất kỳ ứng dụng nào đang được hướng tới. Quá trình thiết kế rất đơn giản và cung cấp một con đường có hệ thống giữa việc xác định bộ ngoại vi tùy chỉnh của bạn và đưa chức năng của nó lên hệ điều hành Linux chạy trên bộ xử lý.

Đèn giao thông gồm 3 màu: xanh, đỏ, vàng. Mỗi một màu đều có ý nghĩa riêng cụ thể như:

- Đỏ: Khi gặp đèn đỏ, tất cả các phương tiện đang lưu thông phải dừng lại ở phía trước vạch dừng (trừ trường hợp những xe rẽ phải và những xe được quyền ưu tiên đi làm nhiệm vụ).
- Xanh: Khi gặp đèn xanh, tất cả các phương tiện được phép đi.
- Vàng: Đèn vàng là dấu hiệu của sự chuyển đổi tín hiệu từ xanh sang đỏ.

Khi đèn vàng bật sau đèn xanh nghĩa là các phương tiện chuẩn bị dừng, các phương tiện phải dừng lại trước vạch sơn dừng vì tiếp đó đèn đỏ sẽ sáng, trường hợp đã vượt quá vạch dừng thì phải nhanh chóng cho xe rời khỏi giao lộ.

Khi đèn vàng bật sau đèn đỏ nghĩa là các phương tiện chuẩn bị di chuyển, người lái xe có thể đi trước hoặc chuẩn bị để đi vì tiếp đó đèn xanh sẽ sáng.

Khi đèn vàng nhấp nháy ở tất cả các hướng (hoặc ngừng hoạt động) nghĩa là được đi nhưng người lái xe vẫn phải chú ý.

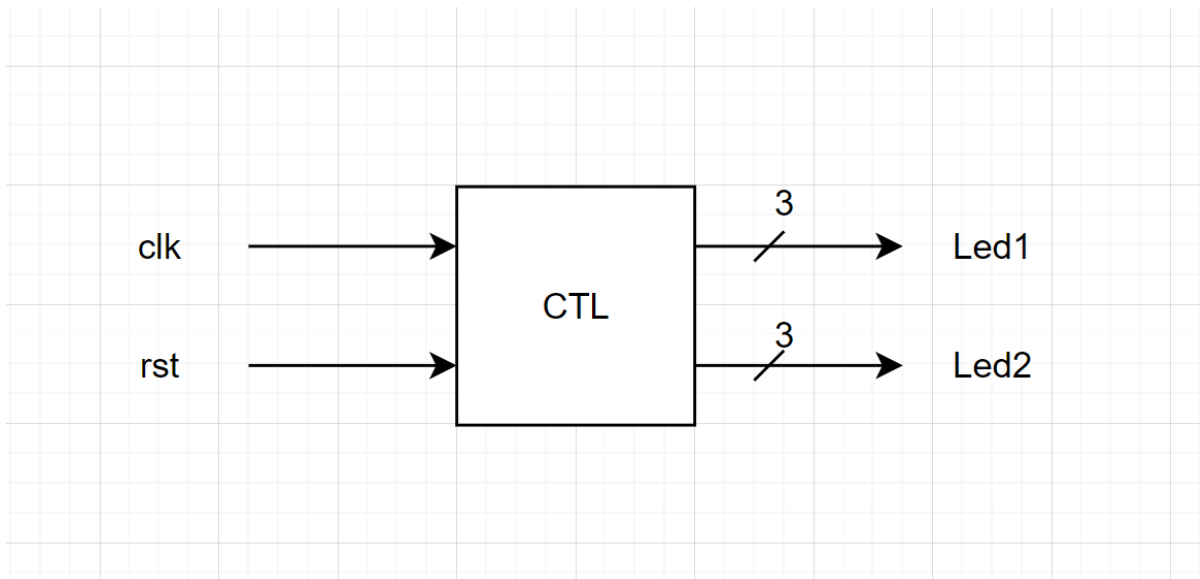
Loại đèn này lắp theo thứ tự: Nếu lắp chiều dọc thì đèn đỏ ở trên, vàng ở giữa, xanh ở dưới. Nếu lắp chiều ngang thì theo thứ tự đỏ ở bên trái, vàng ở giữa, xanh ở bên phải hay ngược lại (đèn xanh luôn luôn hướng về phía vỉa hè hoặc dải phân cách, đèn đỏ hướng xuống lòng đường).

B. Ứng dụng

Đèn giao thông (còn được gọi tên khác là đèn tín hiệu giao thông, đèn điều khiển giao thông, hay đèn xanh đèn đỏ) là một thiết bị được dùng để điều khiển giao thông ở những giao lộ có lượng phương tiện lưu thông lớn (thường là ngã ba, ngã tư đông xe qua lại). Đây là một thiết bị quan trọng không những an toàn cho các phương tiện mà còn giúp giảm ùn tắc giao thông vào giờ cao điểm. Nó được lắp ở tâm giao lộ hoặc trên vỉa hè. Đèn tín hiệu giao thông có thể hoạt động tự động hay cảnh sát giao thông điều khiển.

III. THIẾT KẾ

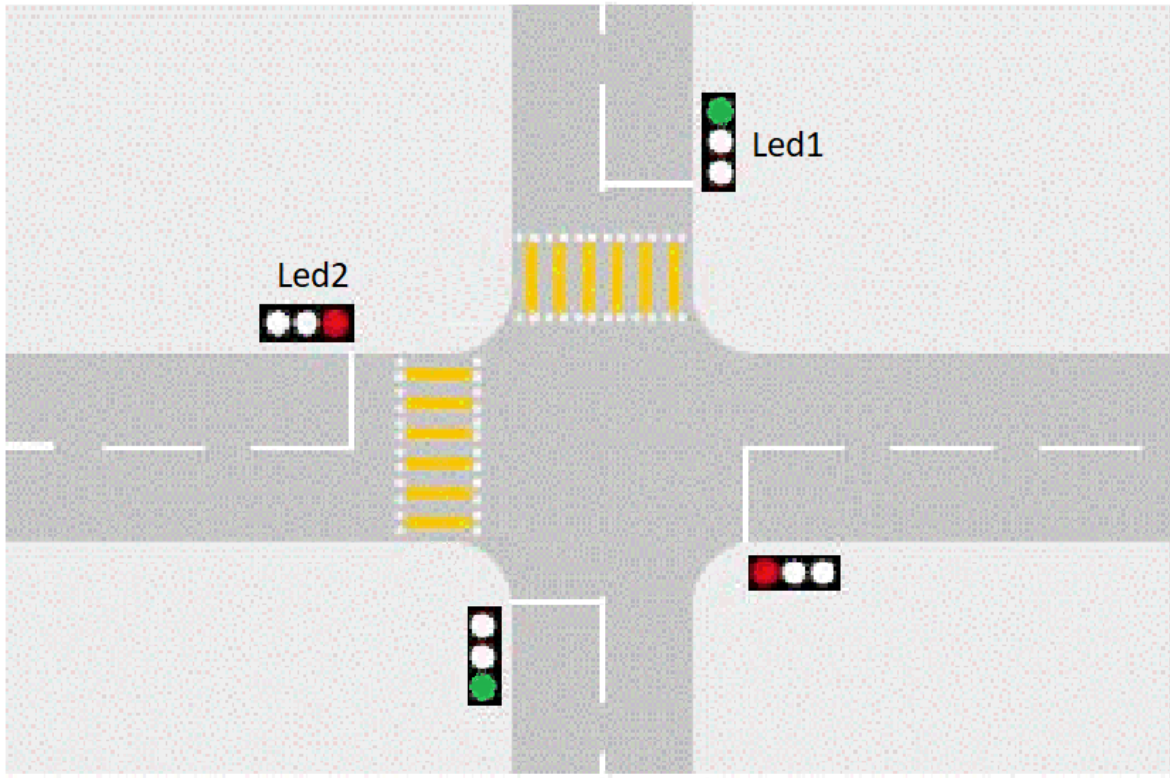
A. Sơ đồ khối



Hình 1: Sơ đồ khối mạch điều khiển đèn giao thông ngã tư.

Mạch xử lí gồm 2 input là xung clock kích cạnh lên và tín hiệu reset tích cực mức cao, 2 output 3-bit để biểu diễn trạng thái của đèn Led RGB.

B. Hệ thống led mô phỏng

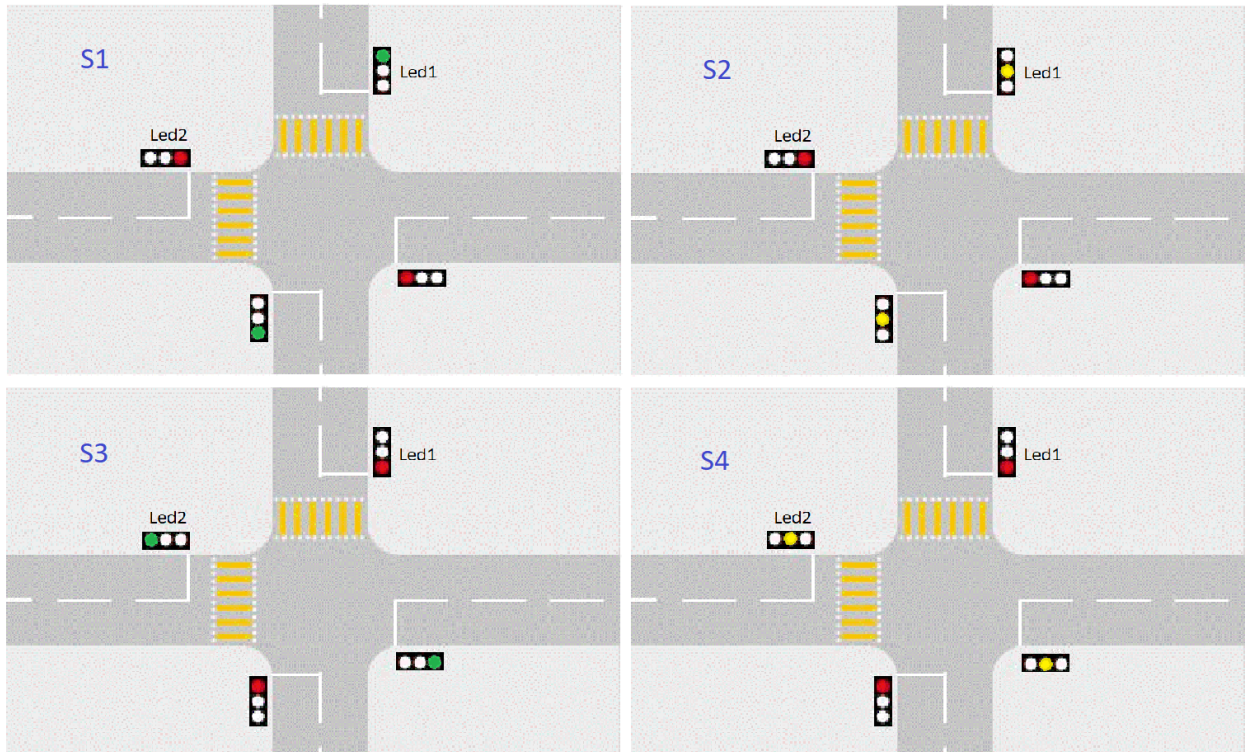


Hình 2: Hệ thống 2 đèn led được mô phỏng trong bài

Khi các đèn làm nhiệm vụ điều khiển giao thông thì các bộ đèn đối diện sẽ có cùng trạng thái về màu đèn, còn các bộ đèn ở đường kẻ sát sẽ ngược lại về màu đèn. Cùng với sự hạn chế về thiết bị nên ta chọn 2 đèn như hình để thiết kế trong mạch mô phỏng.

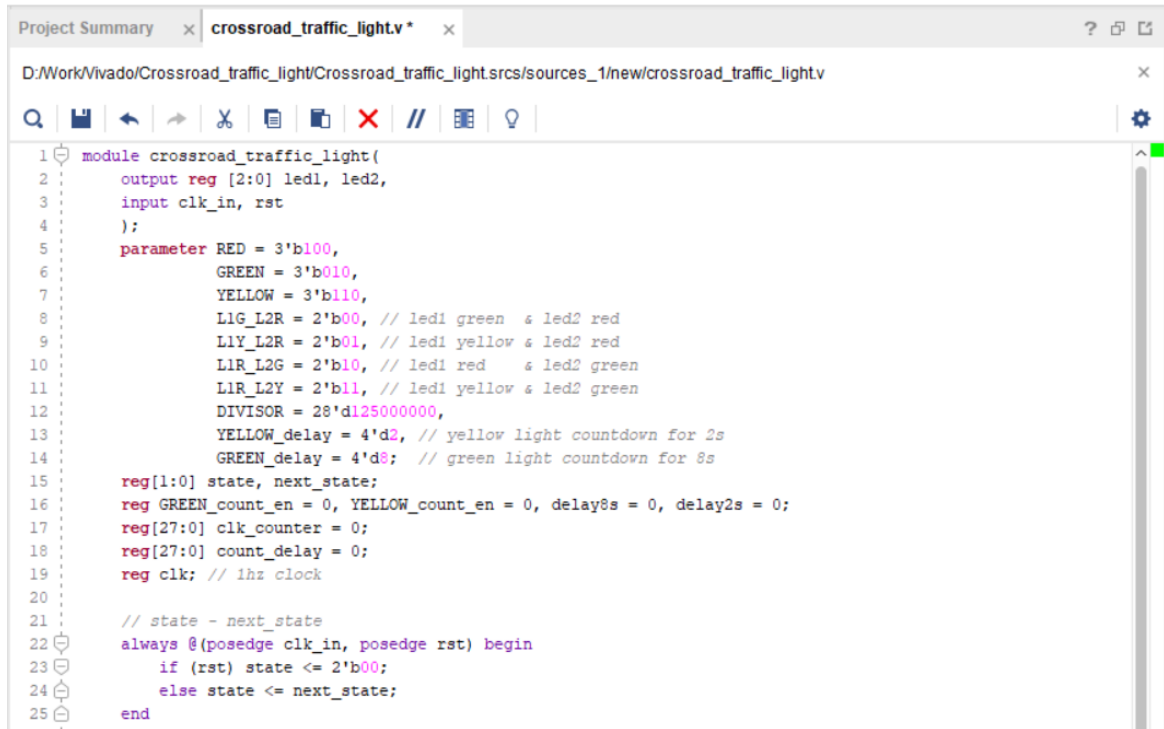
C. Bảng trạng thái

<i>TRẠNG THÁI</i>	<i>TÍN HIỆU</i>	<i>TRẠNG THÁI TIẾP THEO</i>
S0	Led1: Xanh Led2: Đỏ	S1
S1	Led1: Vàng Led2: Đỏ	S2
S2	Led1: Đỏ Led2: Xanh	S3
S3	Led1: Đỏ Led2: Vàng	S0



D. Code Verilog HDL

Đầu tiên, ta khai báo các input, output của module. Sau đó khai báo các tham số, thanh ghi để sử dụng trong chương trình.



```
1 module crossroad_traffic_light(  
2     output reg [2:0] led1, led2,  
3     input clk_in, rst  
4 );  
5     parameter RED = 3'b100,  
6         GREEN = 3'b010,  
7         YELLOW = 3'b110,  
8         L1G_L2R = 2'b00, // led1 green & led2 red  
9         L1Y_L2R = 2'b01, // led1 yellow & led2 red  
10        L1R_L2G = 2'b10, // led1 red & led2 green  
11        L1R_L2Y = 2'b11, // led1 yellow & led2 green  
12        DIVISOR = 28'd125000000,  
13        YELLOW_delay = 4'd2, // yellow light countdown for 2s  
14        GREEN_delay = 4'd8; // green light countdown for 8s  
15    reg[1:0] state, next_state;  
16    reg GREEN_count_en = 0, YELLOW_count_en = 0, delay8s = 0, delay2s = 0;  
17    reg[27:0] clk_counter = 0;  
18    reg[27:0] count_delay = 0;  
19    reg clk; // 1hz clock  
20  
21    // state - next_state  
22    always @(posedge clk_in, posedge rst) begin  
23        if (rst) state <= 2'b00;  
24        else state <= next_state;  
25    end
```

Hình 3.

Mô phỏng mạch bằng mô hình máy trạng thái Moore gồm 4 trạng thái (bảng trạng thái).

```

26 //
27 // FSM
28 always @(*) begin
29     case (state)
30         L1G_L2R: begin
31             led1 = GREEN;
32             led2 = RED;
33             GREEN_count_en = 1;
34             YELLOW_count_en = 0;
35             if (delay8s) next_state = L1Y_L2R;
36             else next_state = L1G_L2R;
37         end
38         L1Y_L2R: begin
39             led1 = YELLOW;
40             led2 = RED;
41             YELLOW_count_en = 1;
42             GREEN_count_en = 0;
43             if (delay2s) next_state = L1R_L2G;
44             else next_state = L1Y_L2R;
45         end
46         L1R_L2G: begin
47             led1 = RED;
48             led2 = GREEN;
49             GREEN_count_en = 1;
50             YELLOW_count_en = 0;
51             if (delay8s) next_state = L1R_L2Y;
52             else next_state = L1R_L2G;
53         end
54         L1R_L2Y: begin
55             led1 = RED;
56             led2 = YELLOW;
57             YELLOW_count_en = 1;
58             GREEN_count_en = 0;
59             if (delay2s) next_state = L1G_L2R;
60             else next_state = L1R_L2Y;
61         end
62         default: next_state = L1G_L2R;
63     endcase
64 end

```

Hình 4.

Vì thời gian đèn đỏ bằng tổng thời gian đèn xanh và đèn vàng nên ta chọn 2 loại màu đèn xanh và vàng để tạo thời gian delay cho mạch.

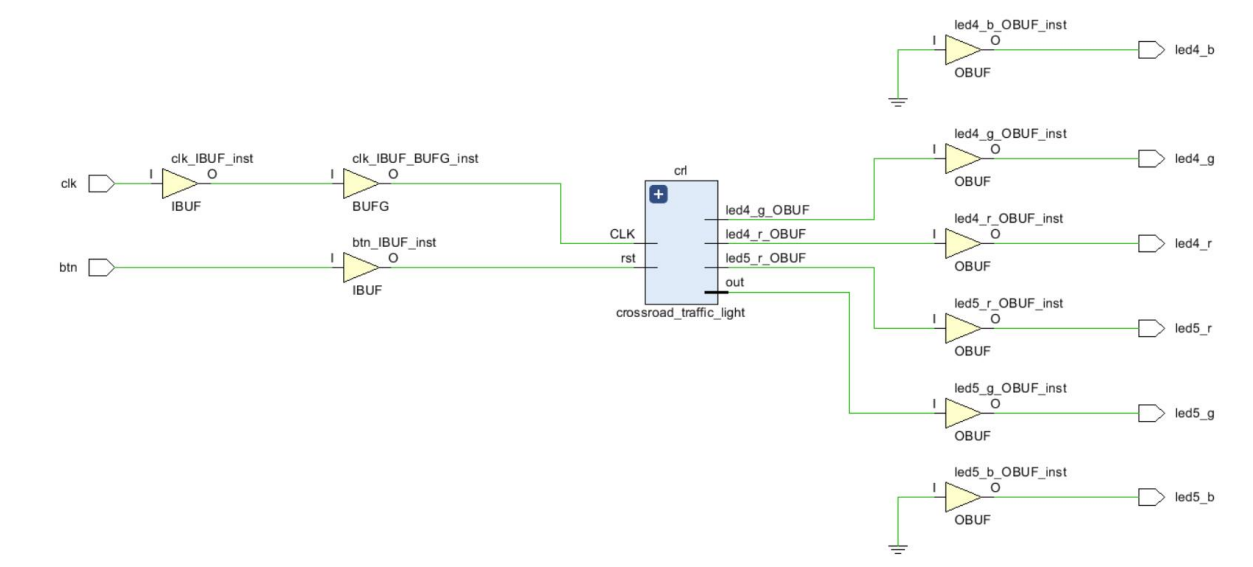
```

66 : // create green & yellow countdown
67 : always @(posedge clk_in) begin
68 :     if (clk == 1) begin
69 :         if (GREEN_count_en || YELLOW_count_en)
70 :             count_delay <= count_delay + 28'b1;
71 :         if ((count_delay == GREEN_delay-1) && GREEN_count_en) begin
72 :             delay8s <= 1;
73 :             delay2s <= 0;
74 :             count_delay <= 0;
75 :         end
76 :         else if ((count_delay == YELLOW_delay-1) && YELLOW_count_en) begin
77 :             delay8s <= 0;
78 :             delay2s <= 1;
79 :             count_delay <= 0;
80 :         end
81 :         else begin
82 :             delay8s <= 0;
83 :             delay2s <= 0;
84 :         end
85 :     end
86 : end
87 :
88 : // create 1hz clock
89 : always @(posedge clk_in) begin
90 :     clk_counter <= clk_counter + 1;
91 :     if (clk_counter >= (DIVISOR-1))
92 :         clk_counter <= 28'd0;
93 :     clk <= (clk_counter < DIVISOR/2)?1'b1:1'b0;
94 :     // clk <= ~clk;
95 : end
96 : endmodule
97 :
98 :
99 :
100 :
101 :

```

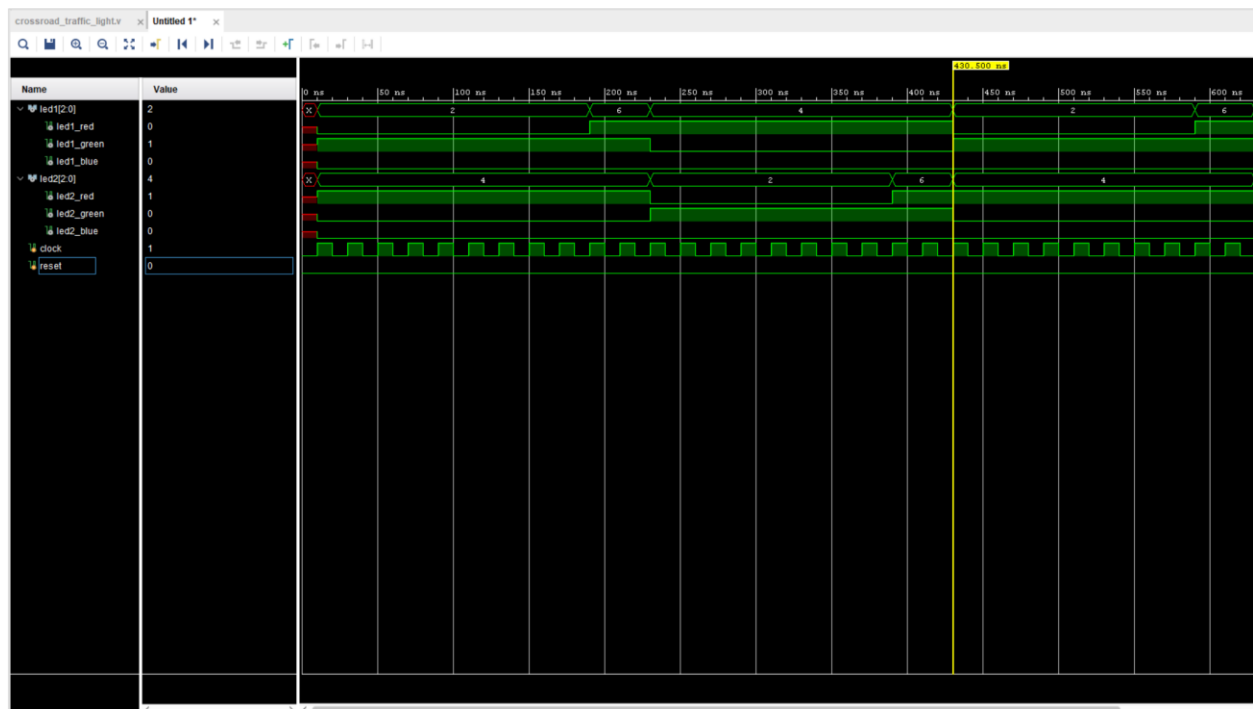
Hình 5.

IV. HIỆN THỰC



Hình 6: Implementation.

V. KẾT QUẢ



Hình 7: Waveform của mạch mô phỏng.

VI. KẾT LUẬN

Thông qua bài tập lớn môn học thiết kế luận lý với HDL, nhóm 8 chúng em đã hiểu biết thêm về cách mô phỏng hệ thống đèn giao thông ngã tư sử dụng đèn Led RGB và hiện thực trên Board Arty Z7-20. Bên cạnh đó, nhờ bài tập lớn chúng em đã được ôn lại cái kiến thức nền tảng của môn học và ngôn ngữ Verilog, hiểu biết về các chức năng của phần mềm Vivado trong quá trình đặc tả phần cứng.

Cuối cùng, nhóm chúng em xin gửi lời cảm ơn đến thầy Nguyễn Xuân Quang đã phụ trách giảng dạy môn học này và đã tận tình giúp đỡ chúng em trong quá trình làm bài tập trong suốt thời gian vừa qua.