**HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY**

**FACULTY OF COMPUTER SCIENCE AND ENGINEERING**

\* \* \* \* \* \* \* \* \* \*



**LOGIC DESIGN WITH HDL**

**Báo Cáo BTL**

**Nhóm 1**

**Nhóm 1**

Thành viên:

* Nhóm trưởng: Lê Nguyễn Đức Huy

MSSV: 2010285

* Nguyễn Phúc Khang

MSSV: 2010318

* Đặng Đình Thông

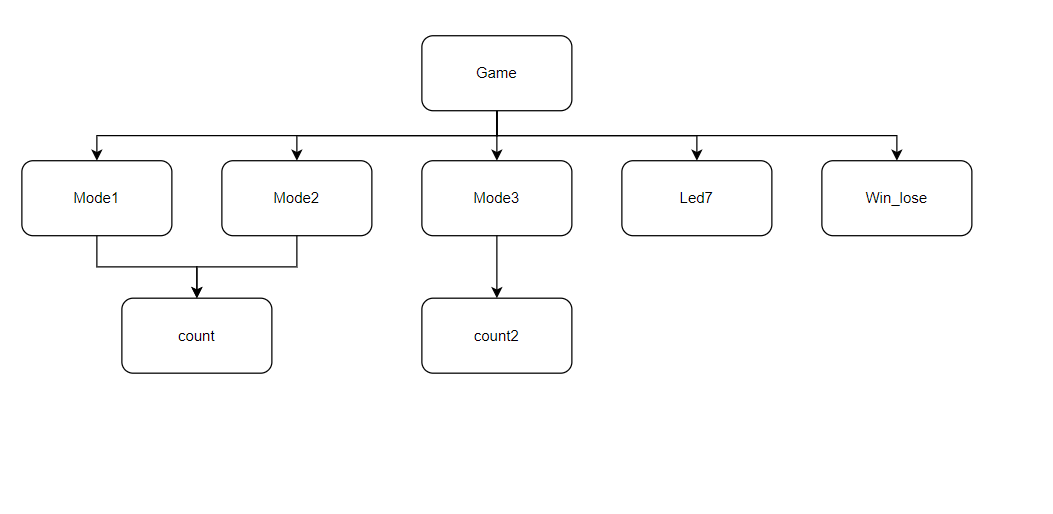
MSSV: 2010660

# 1.Đặc tả đề Tài

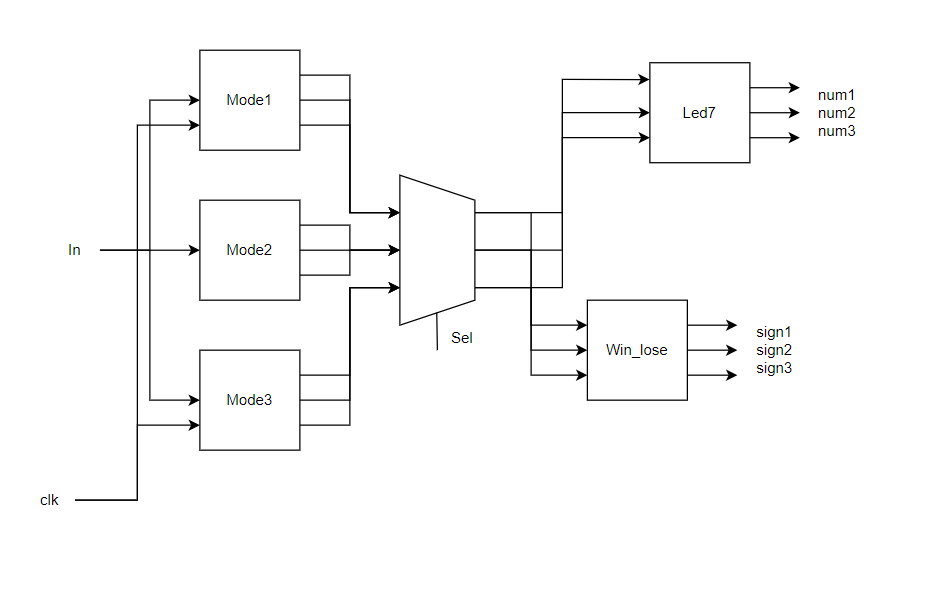
Đề tài của nhóm là thiết kế trò chơi quay số may mắn, thắng nếu ba con số trên màn hình trùng với nhau. Ý tưởng để thực hiên được trò chơi này là việc thiết kế ba bộ đếm đếm từ 0 đến 9 chạy với 3 tần số khác nhau rồi cho hiển thị kết quả ra trên màn hình led 7 đoạn. Có 3 chế độ chơi và có chức năng thông báo thắng thua cho người chơi. Chế độ một, "nhấn một lần quay số ngẫu nhiên" được thực hiện thông qua module mode2 và count. Khi người chơi nhấn nút thì bộ đếm sẽ hoạt động và hiển thị ra trên led7 đoạn với ba con số khác nhau.Ở chế độ thứ hai, các số quay ngẫu nhiên liên tục khi người chơi nhấn giữ nút và sẽ chỉ dừng lại khi người chơi thả nút nhấn ra. Chế độ này, nhóm sử dụng module mode1 và count đển thực hiện. Với mode1 này, nhóm sử dụng thêm input "in" với in = 0 là khi người chơi đang giữ nút nhấn, và in != 0 khi người chơi thả nút nhấn ra và kết thúc trò chơi. Ở chế độ ba, khi người chơi giữ nút nhấn, các số quay ngẫu nhiên với tốc độ tăng dần đến khi người chơi thả nút nhấn thì tốc độ quay sẽ giảm dần sau đó ngừng lại. Với chế độ này, nhóm sử dụng module mode3 và module count2. Với việc count2 sử dụng thanh ghi để đếm lên thay vì 1 số cố định, ta có thể thay đổi tần số của bộ đếm cho nó tăng dần khi người chơi giữ nút và giảm dần khi người chơi thả nút nhấn ra, sau đó kết quả sẽ được hiển thị ra màn hình. Để hiện kết quả thắng thua, nhóm thiết kế module win\_lose để xem các số quay ra có trùng nhau hay không, nếu thắng màn hình sẽ hiện ra hình mặt cười, ngược lại nếu thua sẽ hiện ra hình mặt mếu trên Led7 đoạn. Để kiểm tra tính đúng đắn của trò chơi sau mỗi module tụi em dùng testbench để kiểm tra kết quả có đúng với mong muốn ban đầu hay không.

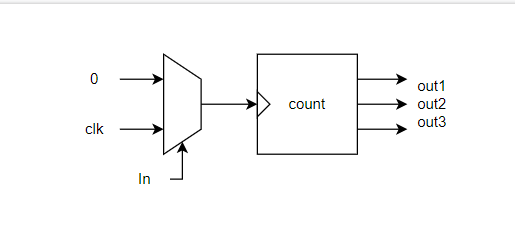
# 2.Thiết kế tổng quan và đặc tả chi tiết module

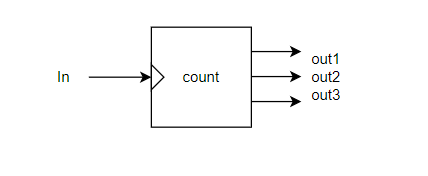
## a. Hierachical tree:

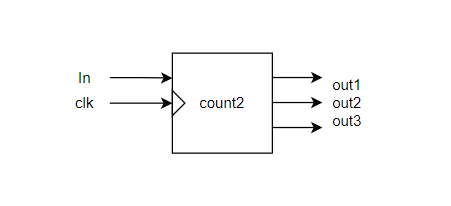


## b. Sơ đồ khối:

Game:

Mode1:

Mode2:

Mode3:

## c. Module:

1. count:

Input: clk

Output: count1, count2, count3

Chức năng: dùng ba mod 10 counter với ba tần số khác nhau để tạo ra ba chữ số khác nhau khi input clk được kích cạnh lên.

Kì vọng output: khi input clk kích cạnh lên thì output count1, count2, count3 sẽ cho ra 3 số bất kì.

2. count2:

Input: clk, in

Output: count1, count2, count3

Chức năng: dùng để tạo ra bộ đếm tương tự với module count nhưng có tốc độ nhảy chữ số thay đổi được (nhanh dần khi giữ, chậm dần rồi dừng khi thả).

Kì vọng output: khác với module count module này có thêm input in để kiểm soát việc thay đổi tốc độ nhảy số là nhanh dần hay chậm dần. Cụ thể, khi ta thả nút nhấn tức là input in sẽ chuyển từ trạng thái 0 sang 1 tức xuất hiện cạnh lên, khi đó mạch chuyển từ đếm nhanh dần sang đếm chậm dần. Do đó, output kì vọng khi nhấn giữ nút nhấn là những số hiển thị nhảy liên tục với tốc độ tang dần, sau đó khi thả ra thì các số sẽ chậm lại rồi một lúc sau thì dừng hẳn.

3. mode1:

Input: in, clk

Output: out1, out2, out3

Chức năng: chế độ thứ nhất, khi ta nhấn giữ nút nhấn thì các số sẽ nhảy liên tục chỉ dừng khi ta thả nút nhấn.

Kì vọng output: khi ta nhấn nút tức khiến input in nhận giá trị 0 thì mạch cho nhảy số liên tục thông qua module count, khi thả tức input in nhận giá trị 1 thì mạch dừng lại.

4. mode2:

Input: in

Output: out1, out2, out3

Chức năng: chế độ chơi thứ hai, khi ta nhấn nút mạch sẽ cho ra một con số bất kì.

Kì vọng output: thông qua việc cho input in, điều khiển bằng nút bấm, làm clk cho module count, khi ta bấm một lần tức một lần kích cạnh lên, module count sẽ cho ra lập tức ba con số bất kì.

5. mode3:

Input: in, clk

Output: out1, out2, out3

Chức năng: chế độ chơi thứ ba, khi ta nhấn giữ nút mạch sẽ cho nhảy số liên tục với tốc độ nhanh dần, khi thả nút ra số sẽ nhảy chậm lại rồi sau một lúc thì dừng hẳn.

Kì vọng output: thông qua module count 2 việc thay đổi tốc độ nhanh hay chậm được kiểm soát bằng input in. Khi ta nhấn giữ thì số hiển thị liên tục với tốc độ nhanh dần cho đến khi ta thả thì số châm dần rồi dừng hẳn.

6. Led7:

Input: in

Output: out

Chức năng: chuyển đổi một số BCD sang tín hiệu hiển thị trên led 7 đoạn.

Kì vọng output: khi input in có giá trị nhị phân 0 (0000) thì output sẽ hiển thị chữ số 0 trên đèn led 7 đoạn. Tương tự, các số khác từ 1 đến 9 ở dạng nhị phân (0001 đến 1001) đều được chuyển sang các chữ số tương ứng trên led 7 đoạn.

7. win\_lose:

Input: inc1, inc2, inc3

Output: sign01, sign02, sign03

Chức năng: hiển thị kết quả thắng thua trên led 7 đoạn, thắng thì sẽ xuất hiện hình ảnh mặt cười, thua thì là mặt mếu.

Kì vọng output: inc1, inc2, inc3 sẽ được nối trực tiếp với số được tạo ra từ kết quả của các chế độ chơi nếu chúng giống nhau thì hệ thống sẽ thông báo thắng ngược lại sẽ thông báo thua.

8. game:

Input: in, sel, clk

Output: num1, num2, num3, sign1, sign2, sign3

Chức năng: là module chính tổng hợp tất cả các module con.

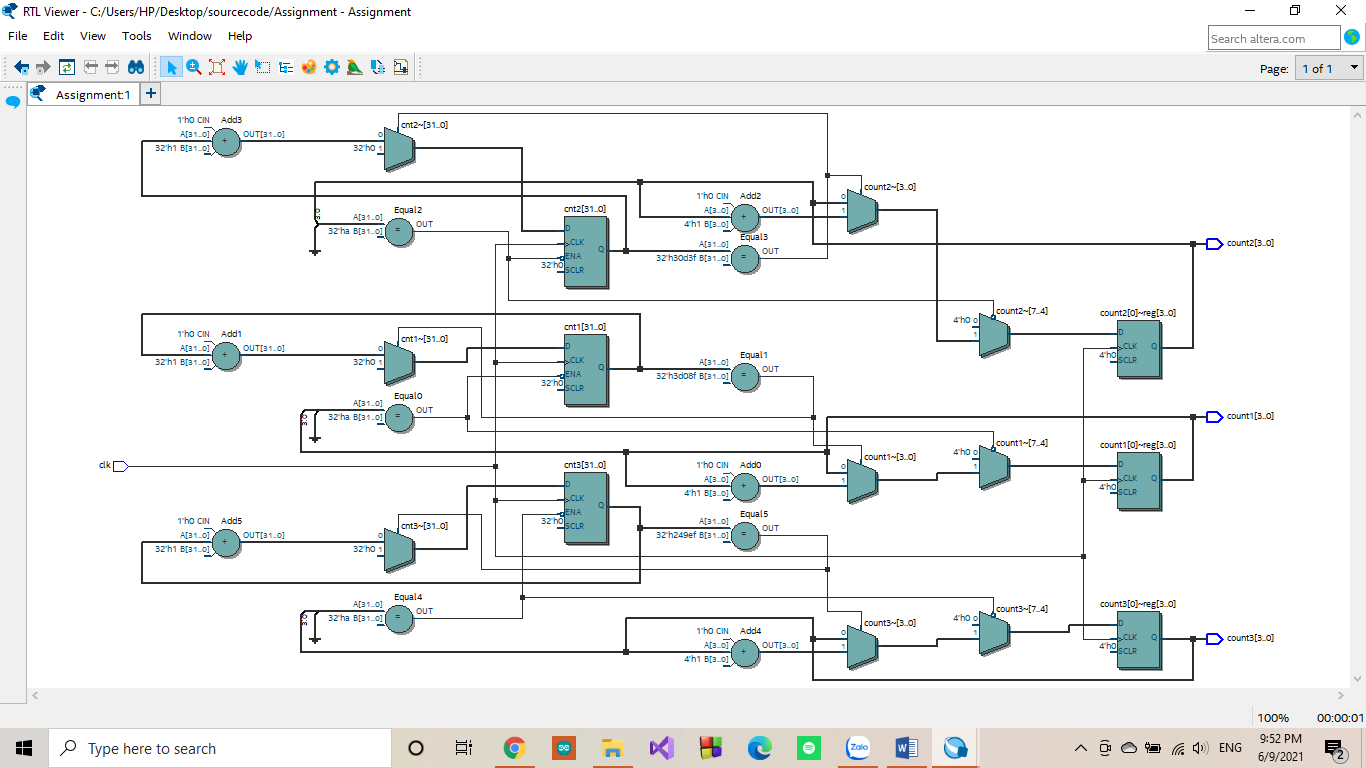
Kì vọng output: sử dụng input sel để chọn chế độ và tùy vào kết quả của mode được chọn để hiện thị lên led 7 thông qua output num1, num2, num3 và hiển thi kết quả thắng thua qua output sign1, sign2, sign3. Cụ thể, module này sẽ gọi cả ba module mode1, mode2, mode3 cho chúng chạy và sẽ thông qua sel để quyết định sử dụng module nào với sel = 0 thì hệ thống sẽ cho cả ba số bằng 0, sel = 1 thì sử dụng module mode1, sel = 2 thì sử dụng module mode2, sel = 3 thì sử dụng module mode3. Sau đó kết quả sau việc chọn lựa sẽ được đưa vào lần lượt module Led7 và win)lóe để hiển thị ra led 7 đoạn.

# 3. Kết quả hiện thực

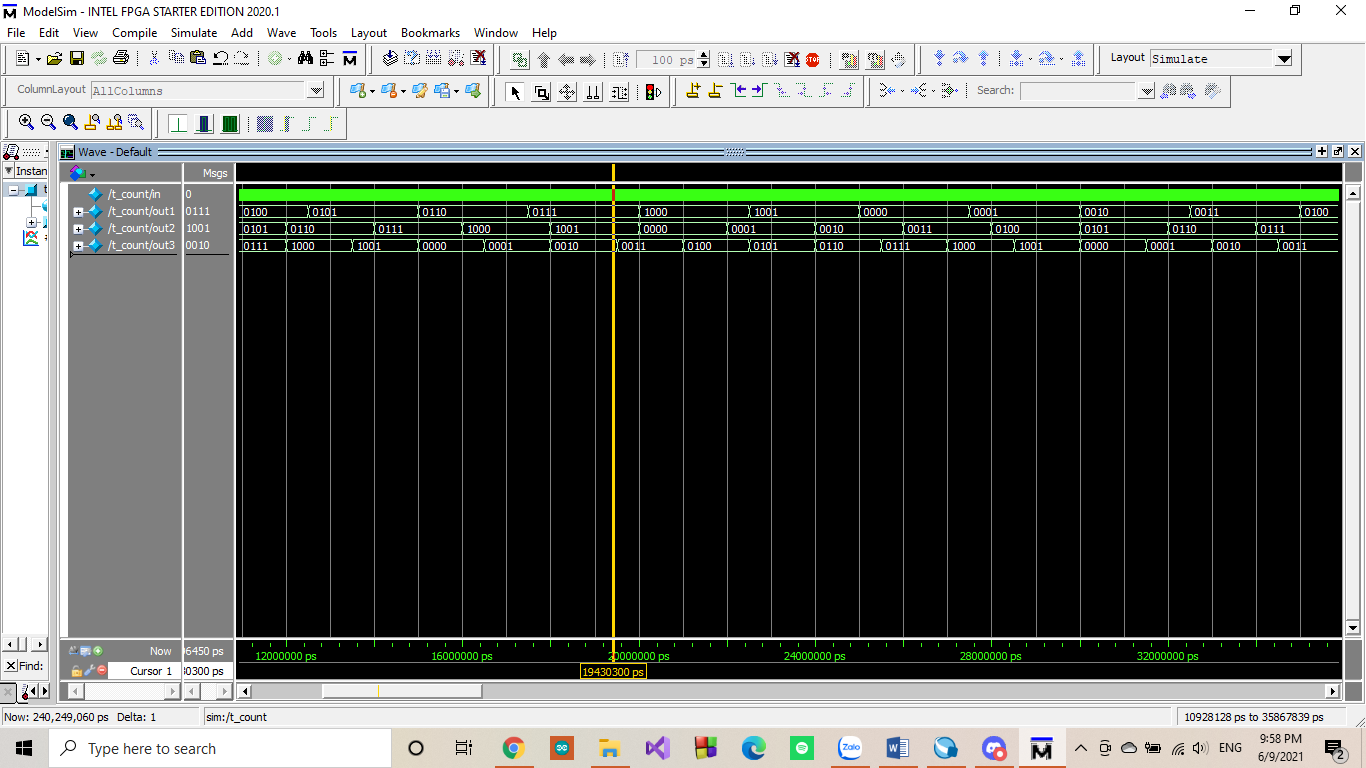
## 1) Tên các file code Verilog:

|  |  |  |
| --- | --- | --- |
| **Tên file** | **Tên file testbench** | **Nơi lưu file** |
| count.v | t\_count.v | Tất cả các file  đều lưu trong thư mục *sourcecode* |
| count2.v | t\_count2.v |
| mode1.v | t\_mode1.v |
| mode2.v | t\_mode2.v |
| mode3.v | t\_mode3.v |
| led7.v | t\_led7.v |
| win\_lose.v | t\_win\_lose.v |
| game.v | t\_game.v |

## 2) RTL view và kết quả chạy testbench:

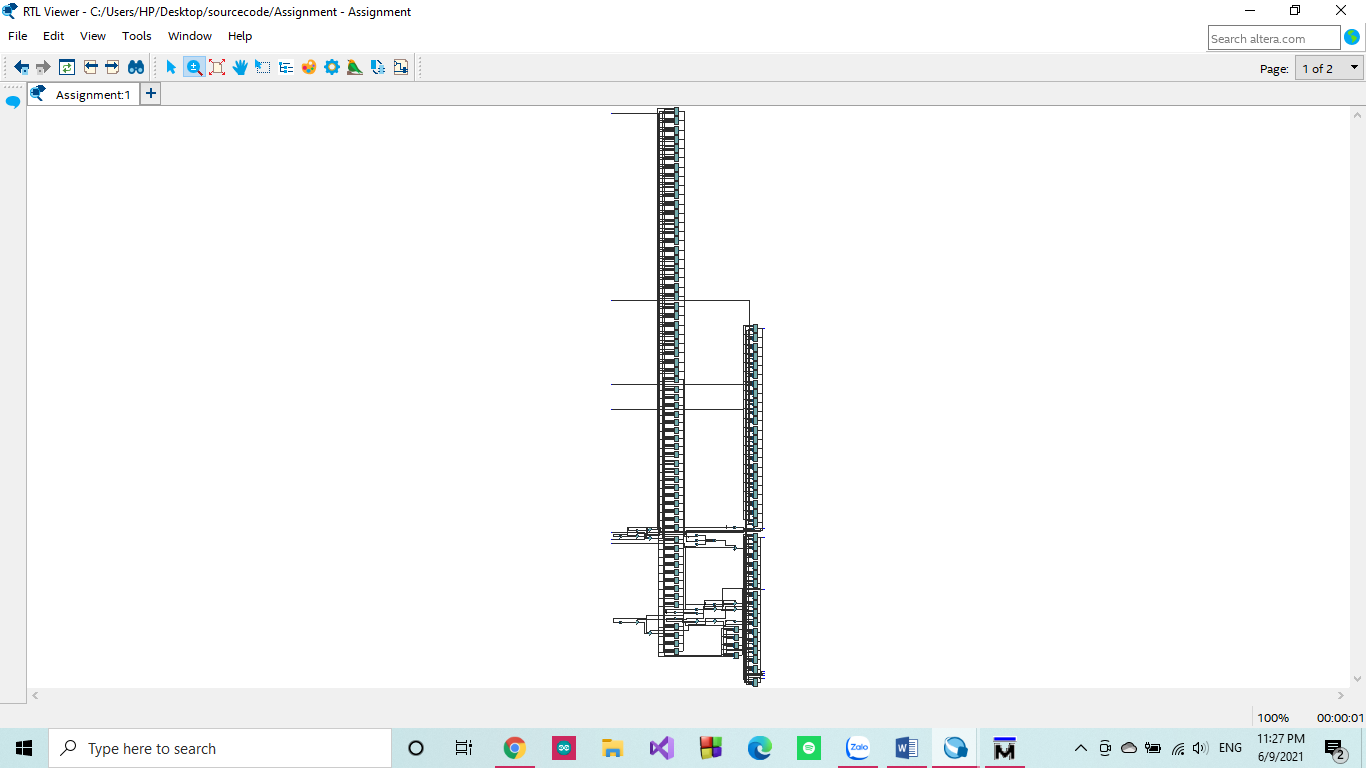
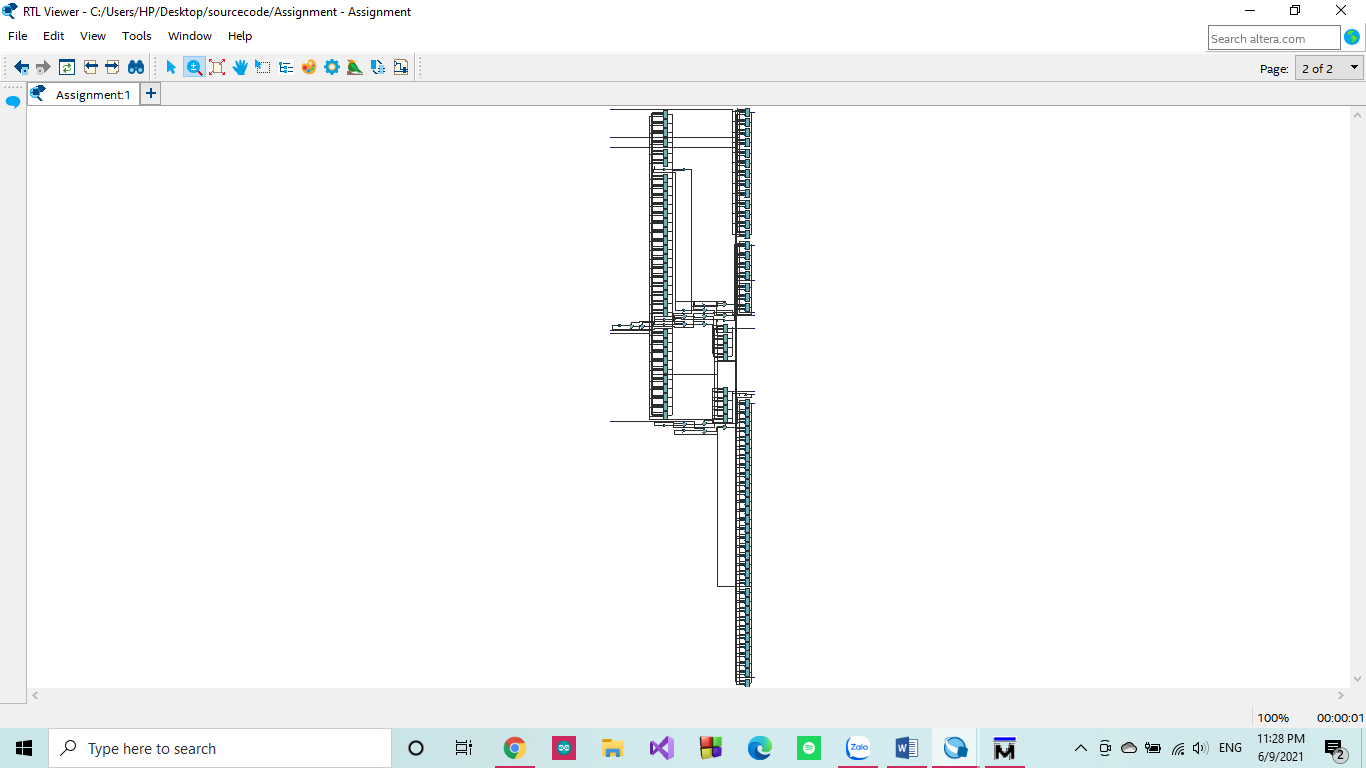
- Module **count**:

*RTL view*

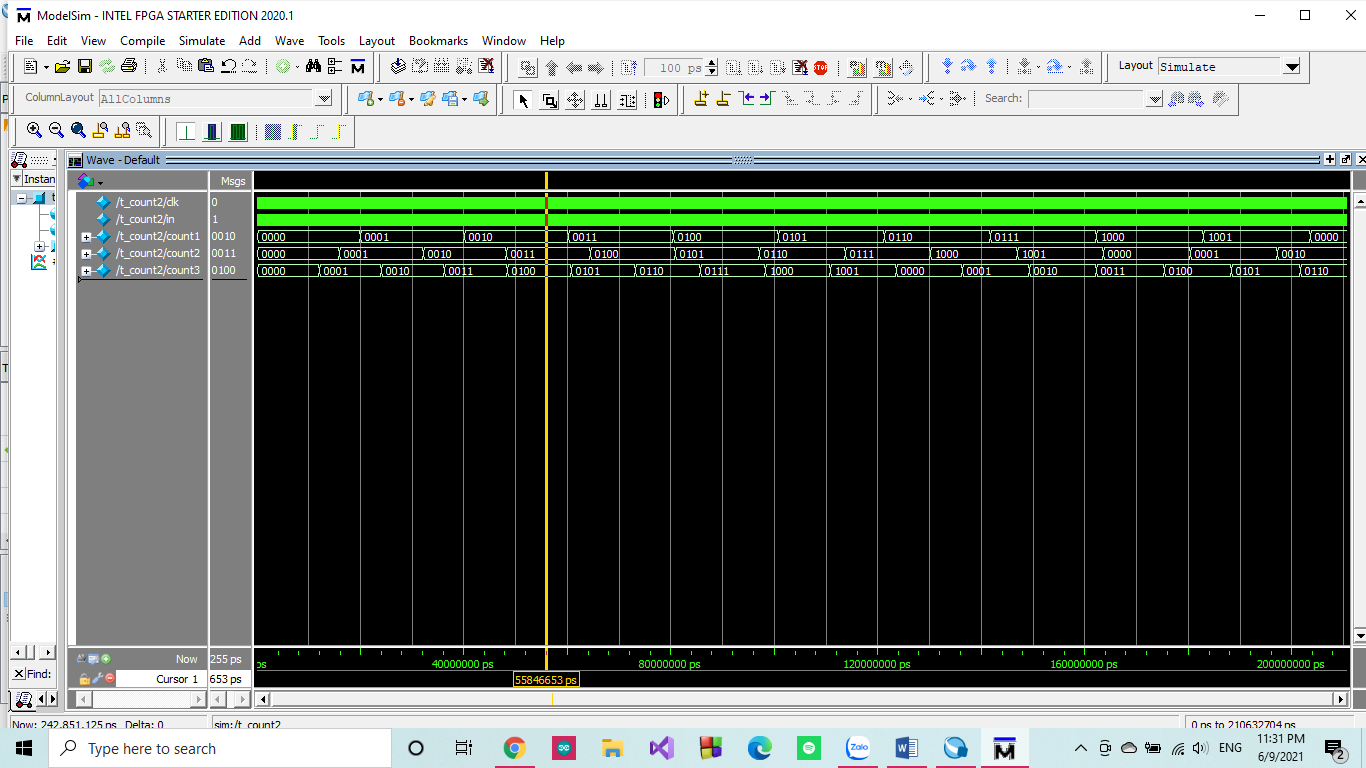


*Kết quả chạy testbench*

- Module **count2**:

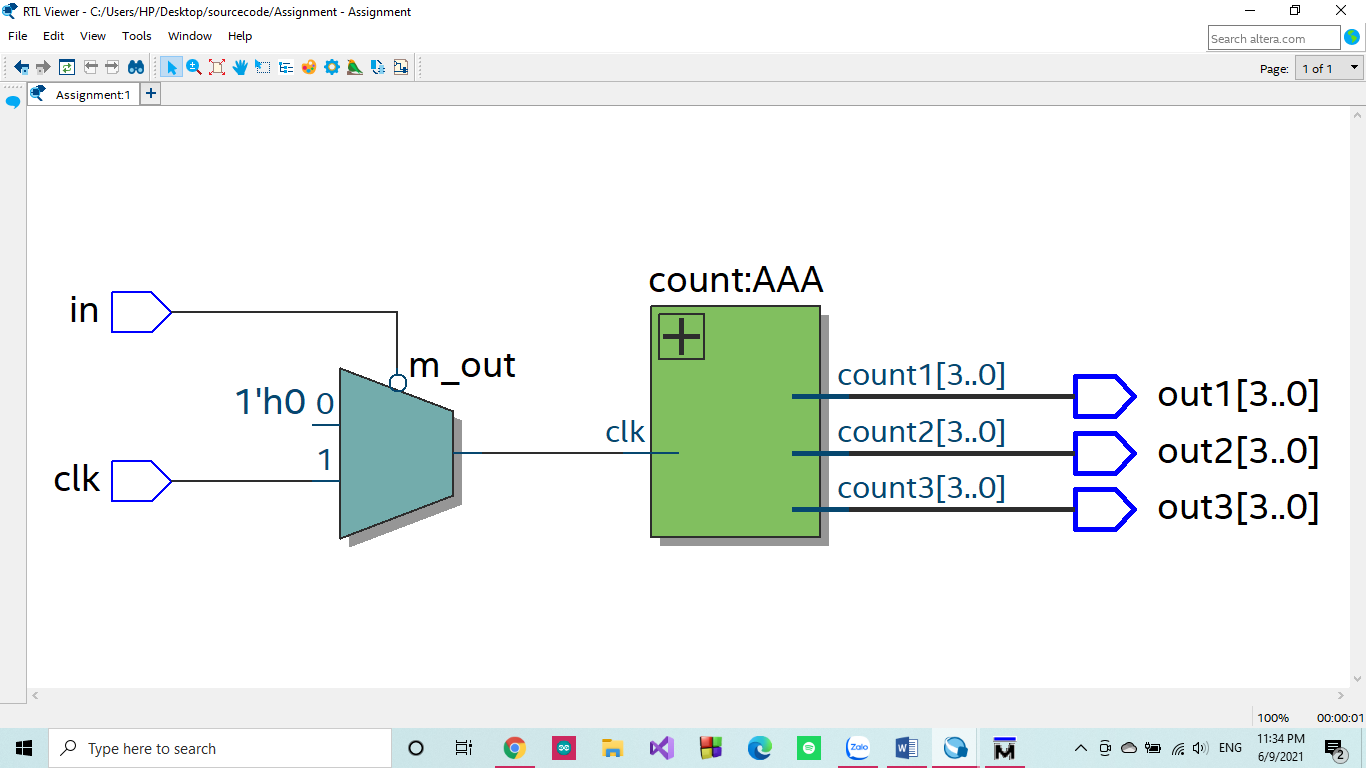


*RTL view*

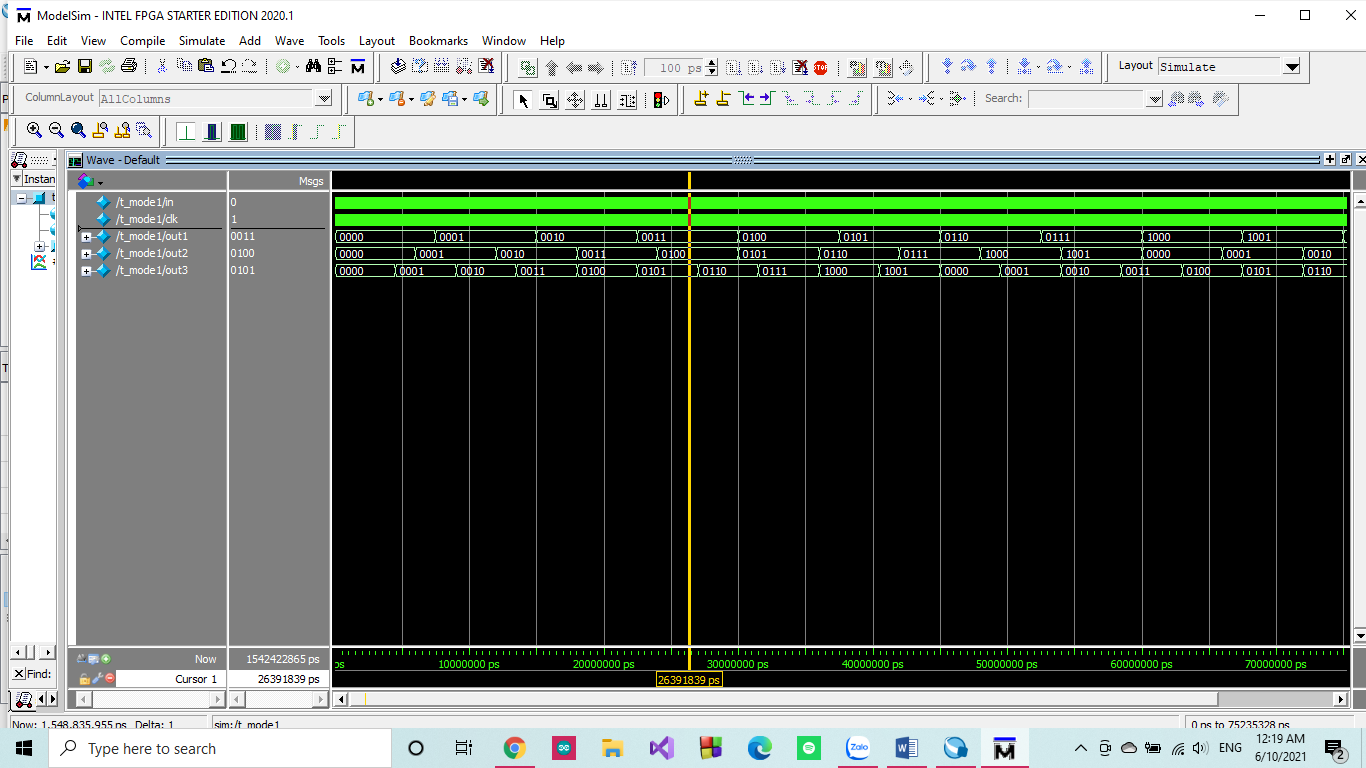


*Kết quả chạy testbench*

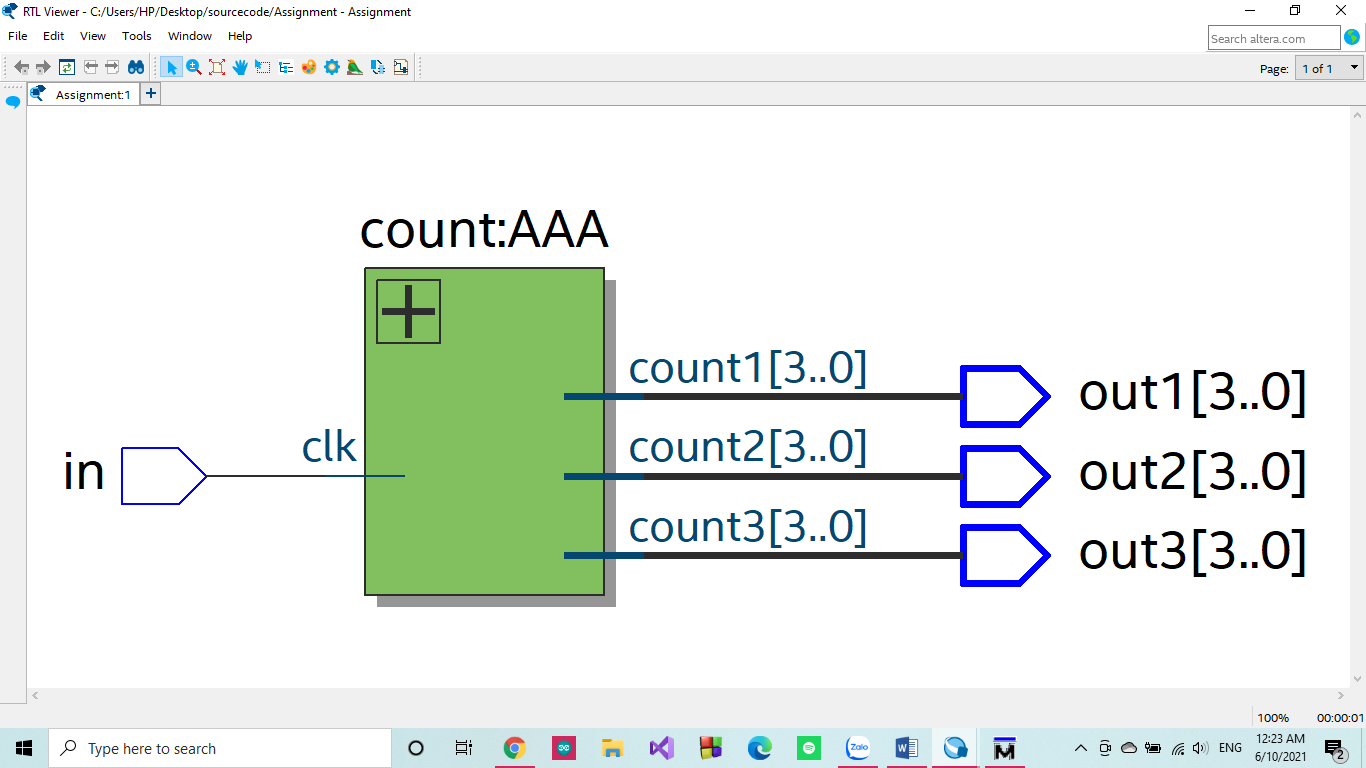
- Module **mode1**:



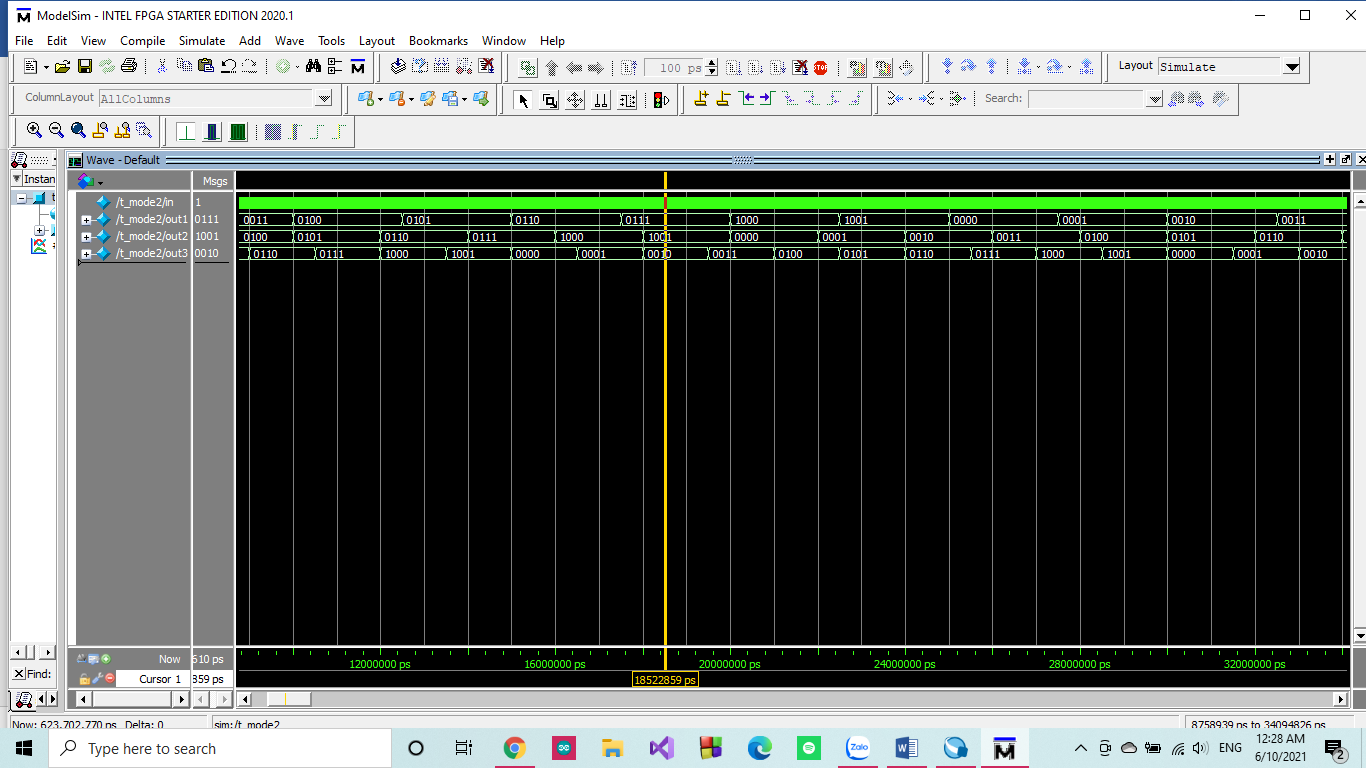
*RTL view*

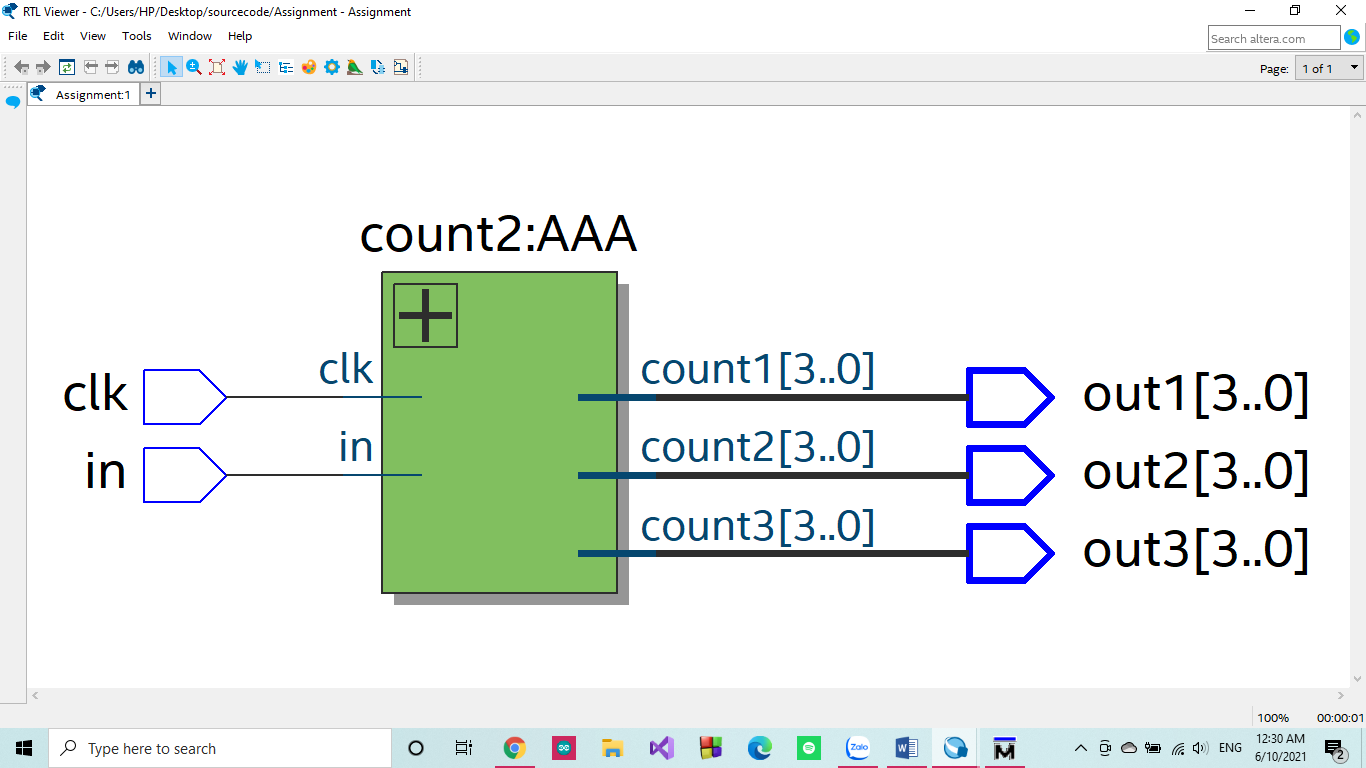


*Kết quả chạy testbench*

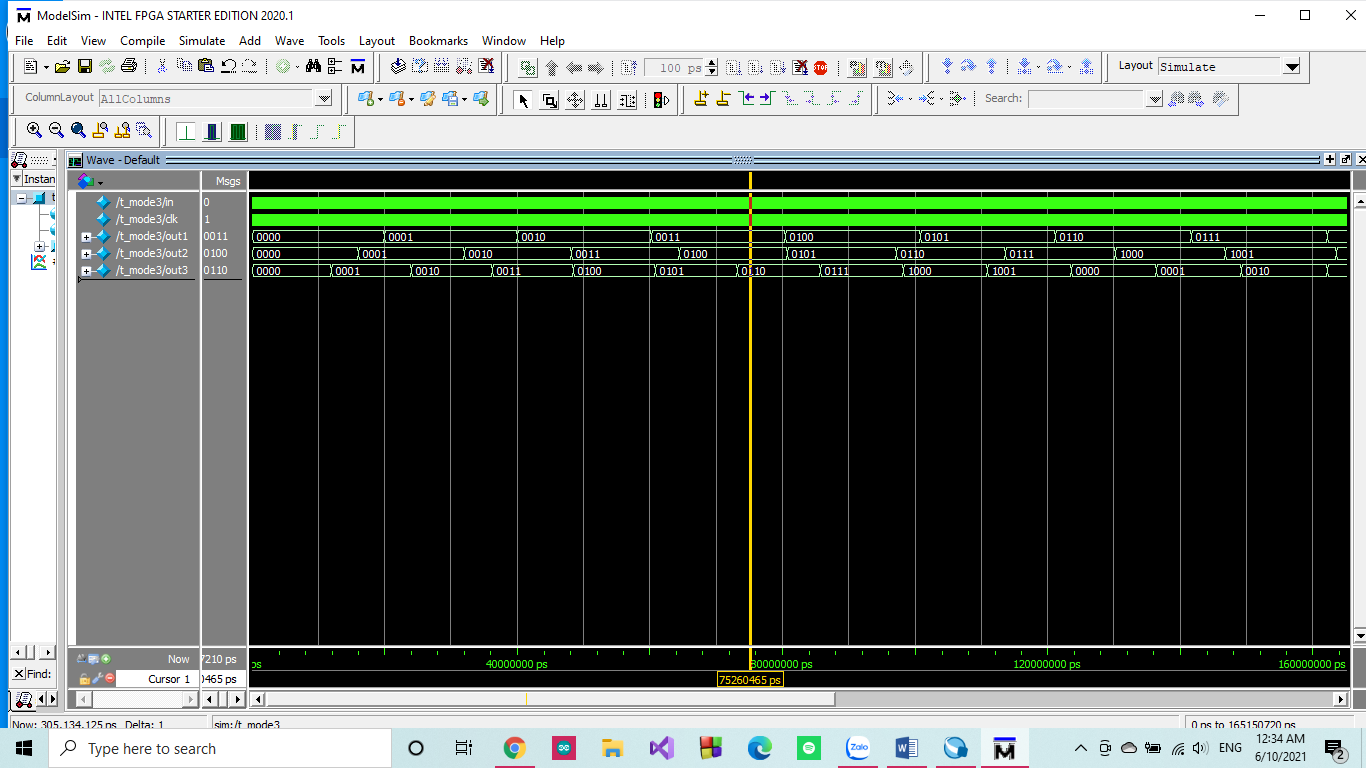
- Module **mode2** :

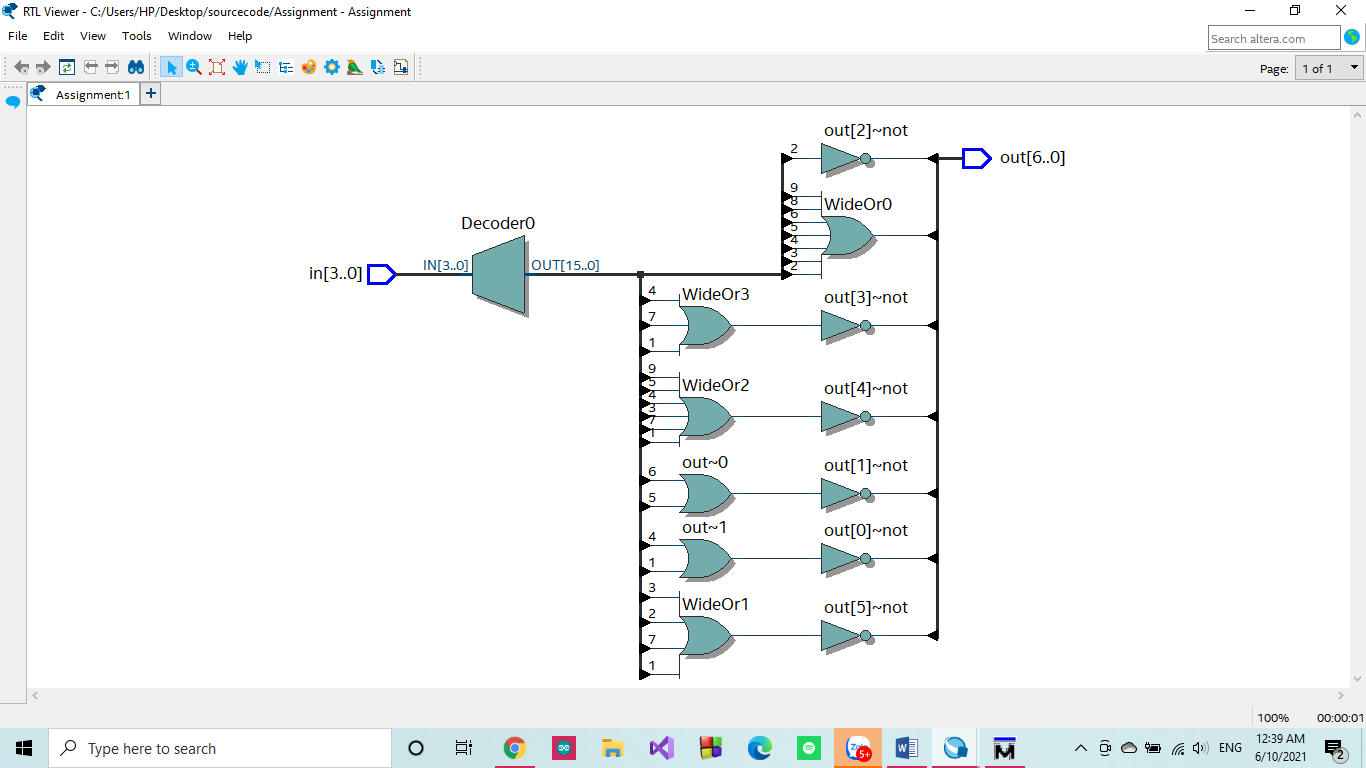
*RTL view*

*Kết quả chạy testbench*

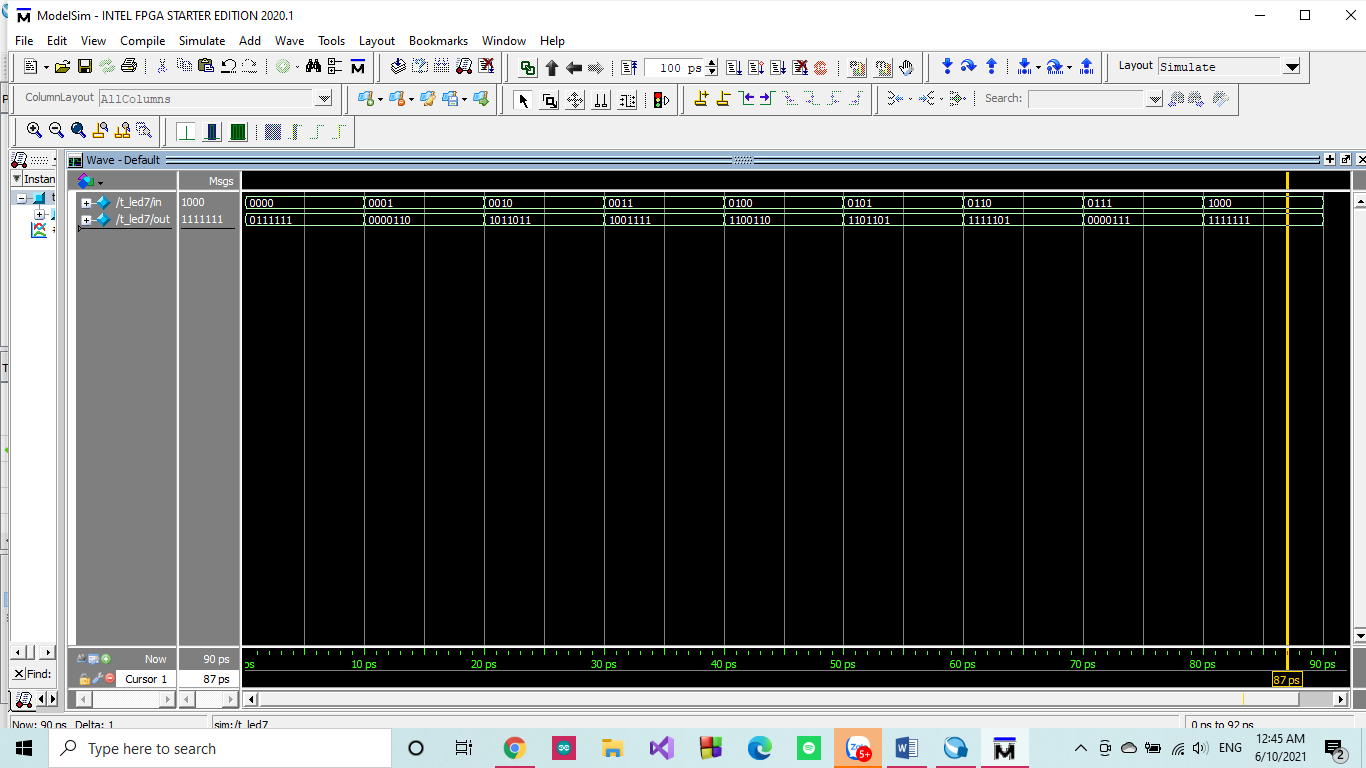
- Module **mode3**:

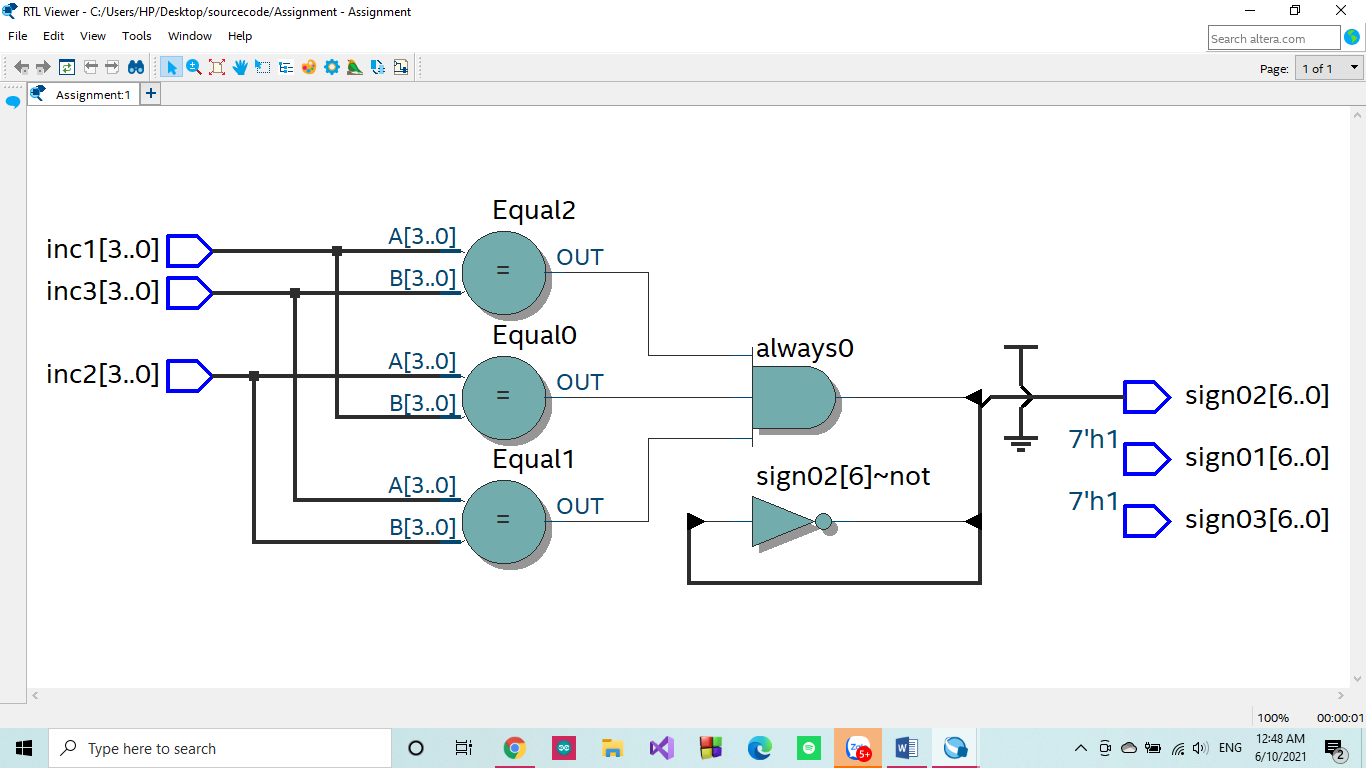
*RTL view*

*Kết quả chạy testbench*

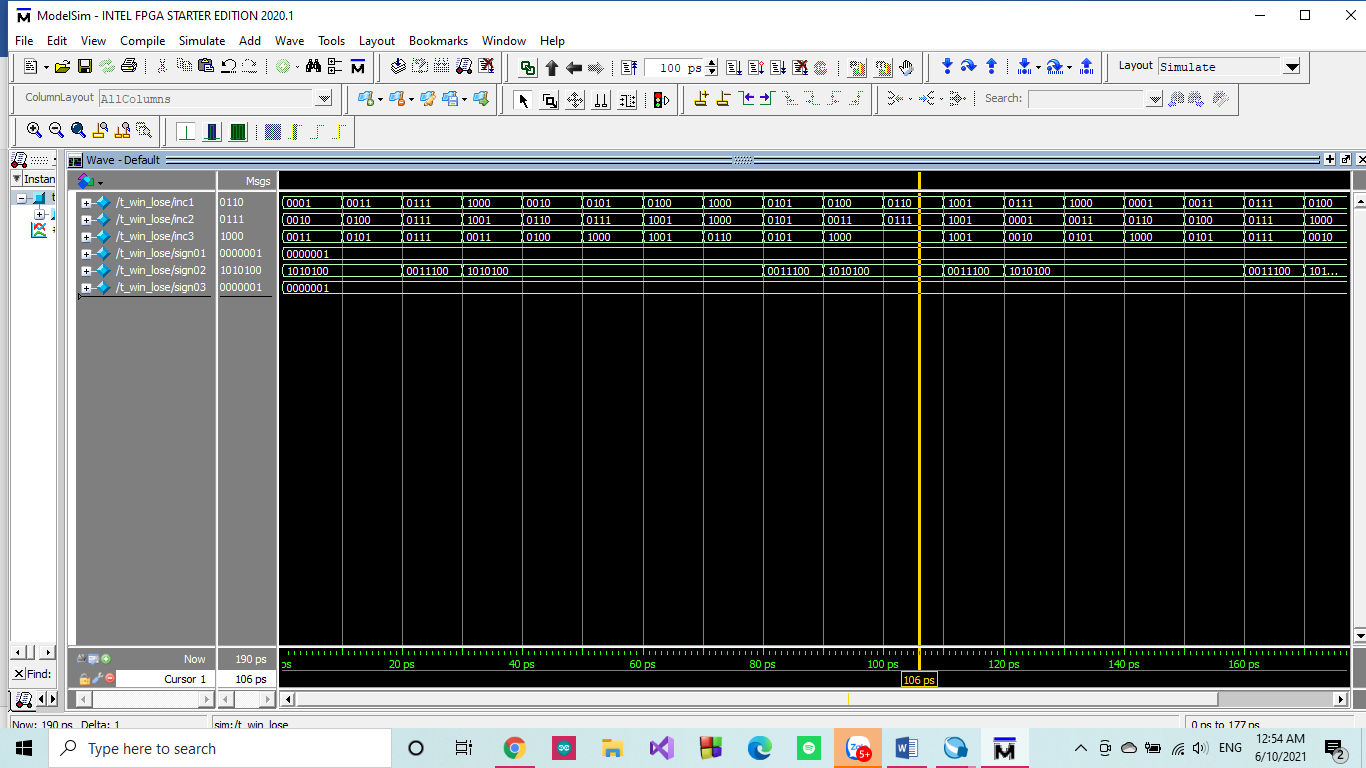
- Module **led7**:

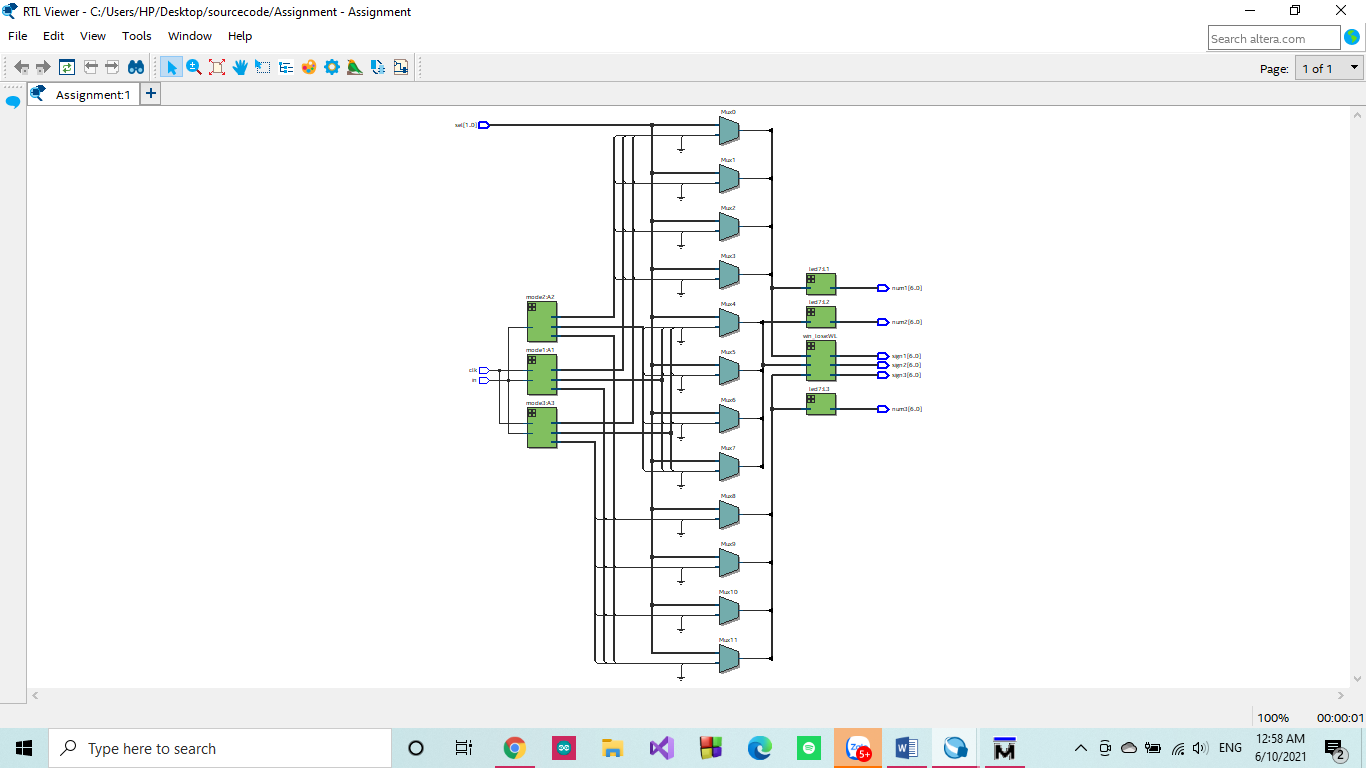
*RTL viewer*

*Kết quả chạy testbench*

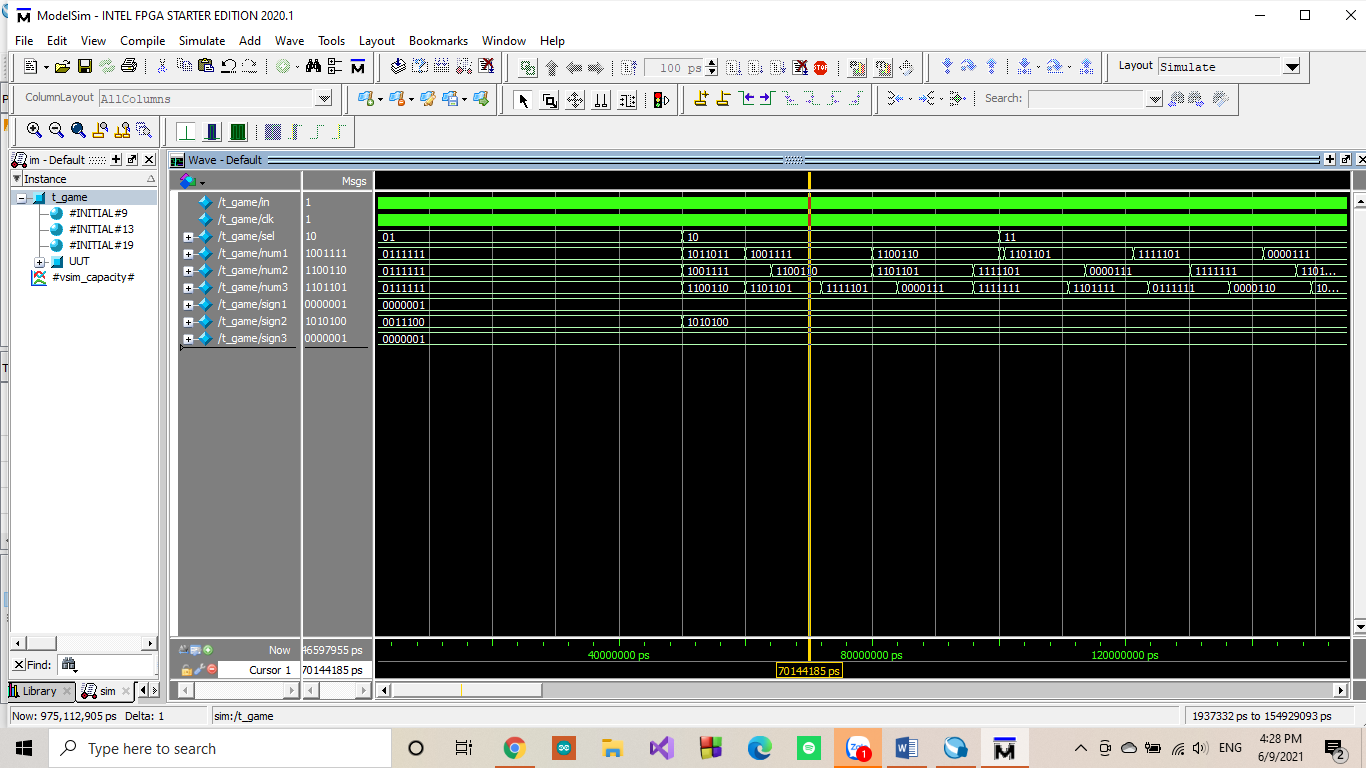
- Module **win\_lose** :

*RTL view*

*Kết quả chạy testbench*

- Module **game** :

*RTL view*

*Kết quả chạy testbench*

# 4. Kết luận và bài học rút ra

- Hệ thống hoàn thiện được các chức năng yêu cầu.

- Qua việc thực hiện đề tài lần này, chúng em đã có nhiều cơ hội được tìm hiểu thêm những kiến thức mới và cũng như nhưng gì đã được học từ môn Hệ thống số và Thiết kế luận lí HDl để áp dụng vào thực tiễn thông qua trò chơi Quay số may mắn. Chúng em đã học hỏi thêm một số kiến thức về Hệ thống số để áp dụng cho verilog HDl và cũng như tìm hiểu về cách trang trí sản phẩm đẹp hơn bằng các đèn led. Bài tập lớn lần này đã rèn giũa chúng em tập cách phân tích, từ ý tưởng chung thành những thành phần nhỏ hơn theo dạng hierarchical tree, biết phân chia các chức năng của hệ thống trò chơi, phân chia các module để thực hiện. Tuy có khó khăn trong việc tìm hiểu và xử lí những kiên thức mới, cũng như lên ý tưởng cho đề tài và phân chia chức năng của nó hợp lí, chúng em đã cùng nhau thảo luân và hoàn thành. Bài tập lớn đã giúp chúng em có thêm kĩ năng làm việc nhóm, và sự đoàn kết để vượt qua khó khăn đã gặp phải. Nếu có điều kiện chúng em sẽ tìm tòi và hiển thị trò chơi ra màn hình thông qua giao tiếp VGA.

# 5. Lịch sử phân công công việc và mức độ hoàn thành công việc

- Lê Nguyễn Đức Huy:

+ Viết code Verilog.

+ Soạn báo cáo phần Thiết kế tổng quan và đặc tả chi tiết Module.

+ Đánh giá: hoành thành tốt.

- Nguyễn Phúc Khang:

+ Viết testbench t\_mode2.v, t\_mode3.v, t\_win\_lose.v, t\_game.v

+ Soạn báo cáo phần Kết quả hiện thực và soạn Mã nguồn.

+ Đánh giá: hoành thành tốt.

- Đặng Đình Thông:

+ Viết testbench t\_count.v, t\_count2.v, t\_led7.v, t\_mode1.v

+ Soạn báo cáo phần Đặc tả đề tài và Kết quả và bài học rút ra.

+ Đánh giá: hoành thành tốt.