

TRƯỜNG ĐẠI HỌC BÁCH KHOA - ĐẠI HỌC QUỐC GIA  
THÀNH PHỐ HỒ CHÍ MINH  
Khoa Khoa học và kĩ thuật máy tính



BÁO CÁO BÀI TẬP LỚN THIẾT KẾ LUẬN LÝ

---

**Đề tài:** Trò chơi quay số may mắn

**Nhóm:** 02

- Nguyễn Xuân Nam - 2012516
  - Hoàng Minh Triết - 2012262
- 

Thành phố Hồ Chí Minh, Ngày 11 tháng 6 năm 2021

# Mục lục

<b>1</b>	<b>Giới thiệu</b>	<b>2</b>
1.1	Trò chơi quay số may mắn . . . . .	2
1.2	Công cụ và thiết bị sử dụng . . . . .	2
1.3	Các chức năng . . . . .	2
<b>2</b>	<b>Thiết kế</b>	<b>3</b>
2.1	Sơ đồ khối . . . . .	3
2.2	Chức năng từng phần . . . . .	3
<b>3</b>	<b>Cách thức hiện thực</b>	<b>4</b>
3.1	Module rand: . . . . .	4
3.2	Module clock-divide . . . . .	5
3.3	Module mode1: . . . . .	6
3.4	Module mode2: . . . . .	8
3.5	Module mode3: . . . . .	10
3.6	Module modex: . . . . .	12
3.7	Module main . . . . .	13
<b>4</b>	<b>Kết luận</b>	<b>15</b>
4.1	Kết luận . . . . .	15
4.2	Hướng phát triển tương lai . . . . .	15
4.3	Khó khăn gặp phải . . . . .	15
4.4	Phân chia công việc . . . . .	15
<b>5</b>	<b>Tài liệu tham khảo</b>	<b>16</b>

# 1 Giới thiệu

## 1.1 Trò chơi quay số may mắn

Trên màn hình hiển thị 3 con số. Khi người chơi nhấn nút bắt đầu, ba con số này sẽ thay đổi ngẫu nhiên và dừng lại sau một khoảng thời gian. Nếu kết quả xuất hiện 3 chữ số giống nhau, người chơi sẽ giành chiến thắng.



Máy quay số may mắn (slot machine)

## 1.2 Công cụ và thiết bị sử dụng

- Verilog HDL
- Phần mềm Quartus II và ModelSim
- Kit phát triển FPGA DE2i-150

## 1.3 Các chức năng

Hệ thống hỗ trợ ít nhất 3 chế độ chơi:

- Người chơi nhấn 1 lần quay số ngẫu nhiên
- Người chơi nhấn giữ nút nhấn, các số quay ngẫu nhiên đến khi người chơi thả nút nhấn
- Người chơi nhấn giữ nút nhấn, các số quay ngẫu nhiên với tốc độ tăng dần đến khi người chơi thả nút nhấn thì giảm dần tốc độ quay đến khi dừng lại.

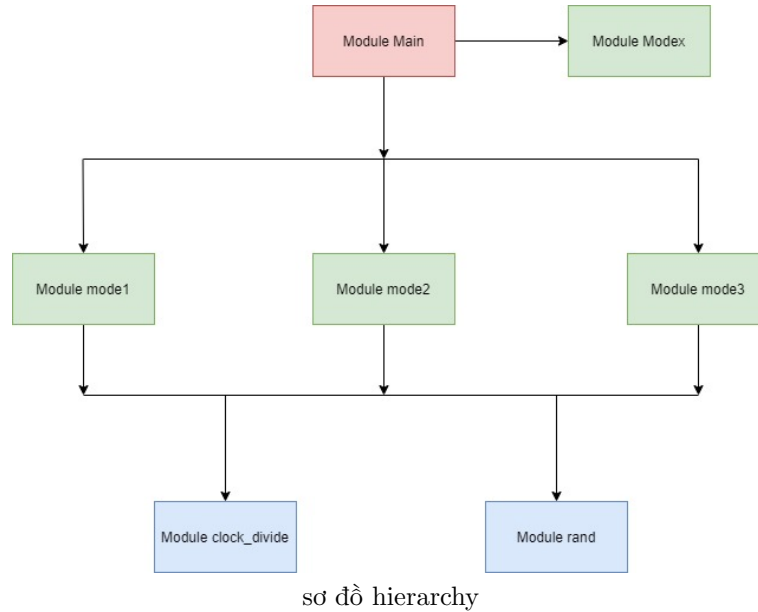
Hệ thống hỗ trợ hiển thị lên kit:

- Sử dụng các Led 7 đoạn để thể hiện các con số
- Sử dụng các Switch, Button để điều chỉnh, thay đổi hệ thống
- Sử dụng các Led đơn để báo hiệu, thông báo kết quả cho người chơi

Ngoài ra, nhóm còn đề xuất thêm tính năng đặt cược để tăng phần thú vị cho người chơi.

## 2 Thiết kế

### 2.1 Sơ đồ khối



### 2.2 Chức năng từng phần

- Module main: module chính, thực hiện chọn trạng thái và chọn số tiền đặt cược, đồng thời thể hiện kết quả và các thông số.
- Module mode1: thực hiện mode1 của máy - nhấn một lần, tiến hành tạo các tổ hợp số ngẫu nhiên trong 3 giây.
- Module mode2: thực hiện mode2 của máy - nhấn giữ, các số quay ngẫu nhiên cho đến khi người chơi thả nút nhấn.
- Module mode3: thực hiện mode3 của máy - khi nhấn giữ nút, số quay nhanh dần, khi thả nút số quay chậm dần cho đến khi dừng lại.
- Module rand: tạo số ngẫu nhiên mỗi xung clock dựa trên cơ chế LFSR (Linear-feedback shift register).
- Module clock divide: tạo xung clock tần số mới dựa trên xung clock của kit (tần số 50MHz).
- Module modex: hiển thị số tiền đặt cược và trong kho còn bao nhiêu tiền. Nếu quay trúng ba số trùng nhau, số tiền cược sẽ được cộng vào kho. Nếu không quay trúng ba số trùng nhau, số tiền trong kho sẽ bị trừ một khoản bằng số tiền cược.

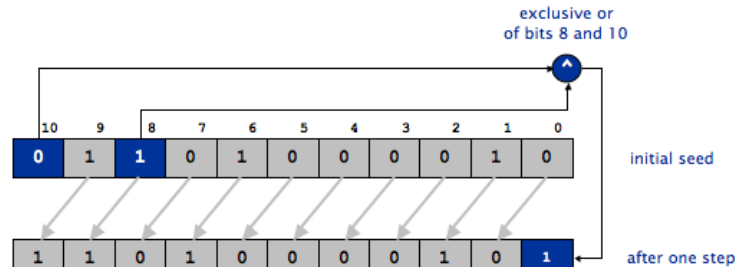
\*Lưu ý:

- Khi nút reset đang bật thì trò chơi sẽ không chạy, chỉ có thể thay đổi mode.
- Không thể đặt cược số tiền (bet) lớn hơn số tiền hiện tại trong kho (sto)
- Khi đang trong quá trình tạo số ngẫu nhiên, không thể thực hiện chọn mode hoặc thay đổi số tiền đặt cược.

### 3 Cách thức hiện thực

#### 3.1 Module rand:

Đã có nhiều công trình nghiên cứu về thuật toán sinh số ngẫu nhiên. Một trong những thuật toán phổ biến và dễ thực thi nhất là thuật toán dịch bit - Linear Feedback Shift Register (LFSR).



One step of an 11-bit LFSR with initial seed 0110100010 and tap at position 8

Ví dụ về thuật toán LFSR

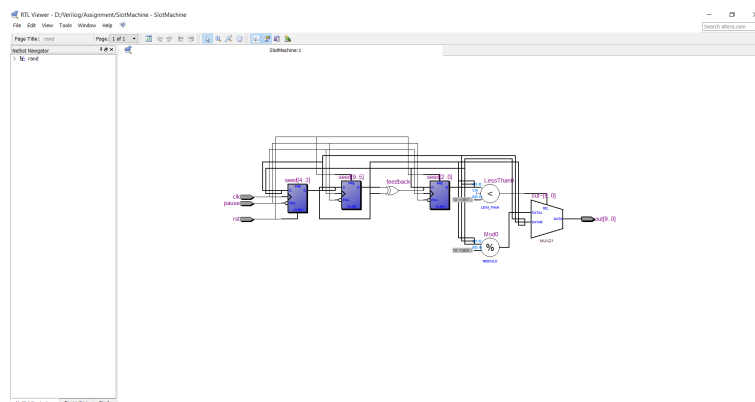
Thuật toán hoạt động như sau:

- Thực hiện phép toán XOR (hoặc XNOR) đối với 2 bit bất kì của số hiện có. Lưu lại giá trị của phép toán (tạm gọi là feedback).
- Dịch tất cả các bit sang trái (hoặc sang phải)
- Thay thế vị trí bit bỏ trống sau khi dịch bit bằng feedback.

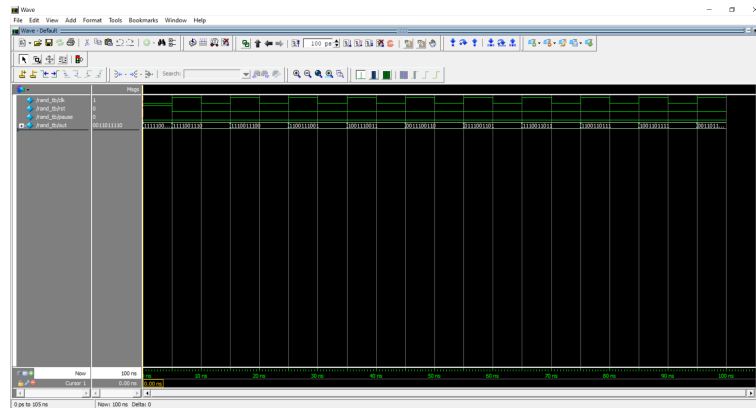
Đối với dữ liệu có  $n$  bit, thuật toán có thể sinh ngẫu nhiên tối đa  $2^n - 1$  giá trị. Tuy nhiên do trên màn hình chỉ hiển thị 3 chữ số, mà dữ liệu đầu vào của chúng ta là 10 bit (vì số lớn nhất có thể là 999) nên có thể sinh số có 4 chữ số (lớn hơn 1000). Ta có thể khắc phục bằng cách lấy phần dư của phép chia các số cho 1000.

Trong đề tài này, chúng ta sẽ tiến hành sinh số ngẫu nhiên theo xung clock. Do đó khi hiện thực bằng Verilog sẽ sử dụng mô hình hành vi (behavior). Module rand gồm các tín hiệu sau:

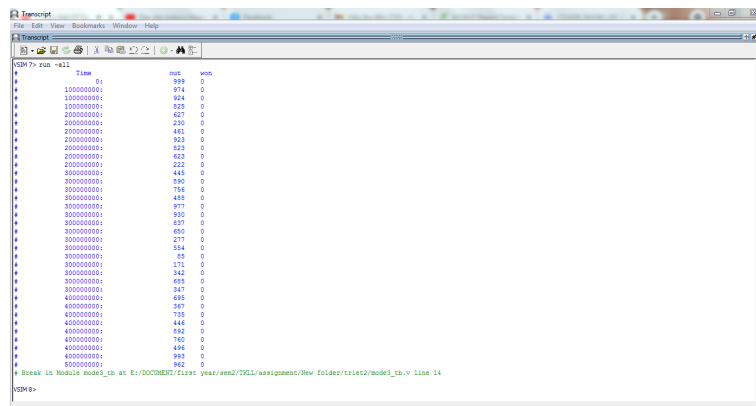
- clk: xung clock đầu vào. Có thể là xung clock của kit hoặc xung clock đã qua chia tần.
- rst: tượng trưng cho nút reset. Khi tín hiệu rst = 1 thì module sẽ reset bất đồng bộ và trả về 999.
- pause: tín hiệu để báo bắt đầu hoặc dừng sinh số ngẫu nhiên.
- out: output của module với các số từ 0 đến 999.



mô phỏng RTL trên modelsim



biểu đồ sóng trên modelsim



kết quả thu được trên modelsim

### 3.2 Module clock-divide

Xung clock của kit là 50MHZ, quá nhanh để người chơi có thể cảm nhận được sự thay đổi của các chữ số .Ngoài ra, các chế độ chơi khác nhau cũng có sự khác biệt về tốc độ nhảy số. Do đó ta cần thiết kế một module chia tần thích hợp để người chơi có thể thấy được sự thay đổi của các số và cũng dễ dàng để ta thiết lập thời gian chạy tự động của trò chơi đồng thời điều khiển tốc độ thay đổi của số.

Module clock-divide gồm các tín hiệu sau:

- clk: xung clock đầu vào 50MHZ từ kit.
- n: input từ người dùng.
- div-clk: xung clock đầu ra với tần số n HZ.

Giải thích cách hoạt động:

- Kit có tần số 50MHZ -> mỗi chu kỳ kéo dài  $2 \times 10^{-8}s$
- Sau  $25 \times 10^6/n$  chu kỳ thì đảo giá trị div clk
- Chu kỳ của div clk kéo dài:  $25 \times 2 \times 10^6 \times 2 \times 10^{-8}/n = 1/n(s)$
- Tần số của div clk:  $n(Hz)$

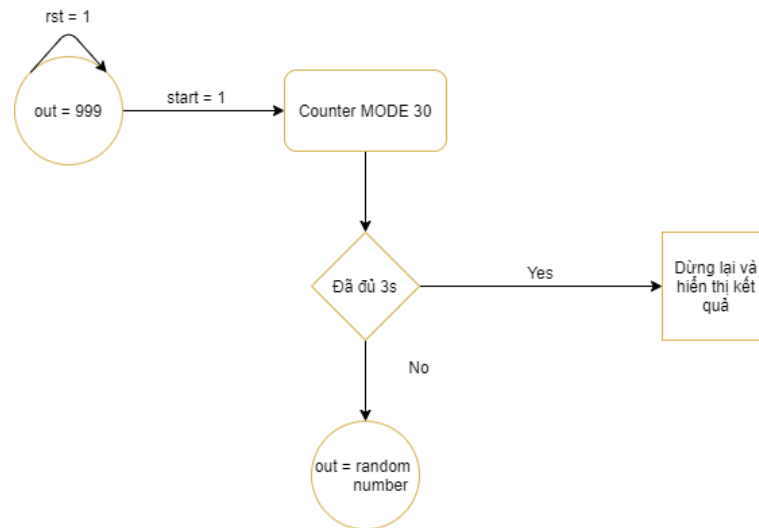
### 3.3 Module model1:

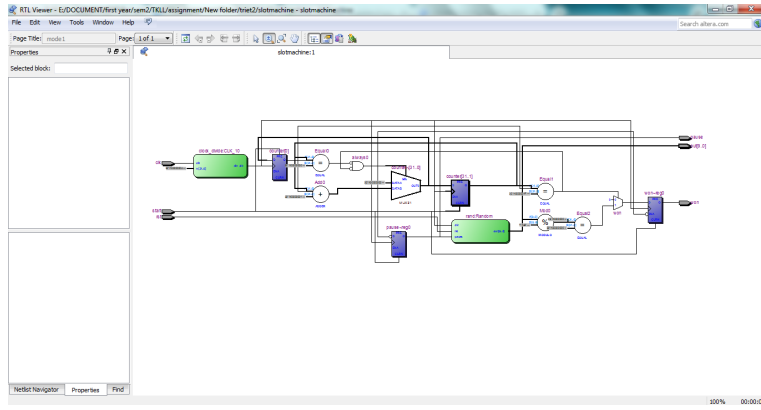
Đối với chế độ 11, người chơi chỉ cần nhấn nút, trò chơi sẽ tự động chạy và tự động dừng lại sau một khoảng thời gian nhất định. Do đó ta chỉ cần chia tần số của kit về tần số thích hợp và thực hiện một bộ đếm để trò chơi tự động dừng lại.

Trong đề tài này, model sẽ sinh 10 số trong vòng 1 giây và tự động dừng lại sau 3 giây. Do đó chúng ta cần đưa tần số 50MHZ về 10HZ, đồng thời thực hiện một bộ đếm MOD 30 để điều khiển thời gian chạy của trò chơi.

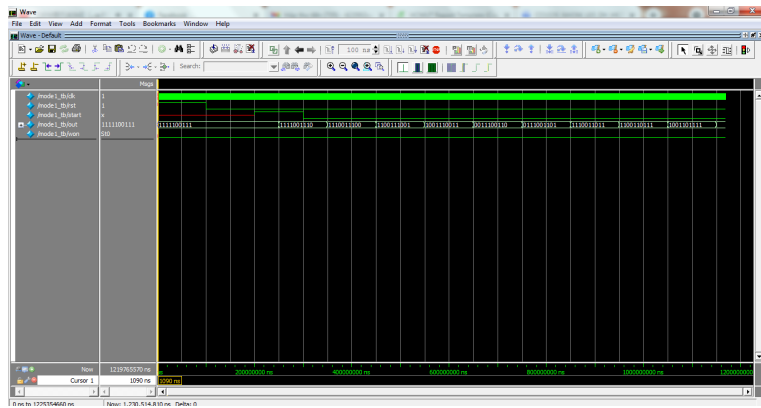
Module model1 gồm các tín hiệu sau:

- clk: xung clock đầu vào với tần số là 10HZ
- rst: tương trưng cho nút reset. Khi tín hiệu rst = 1 thì module sẽ reset bắt đồng bộ và trả về 999.
- start: chính là nút bắt đầu trò chơi. Khi người chơi nhấn nút trò chơi sẽ được bắt đầu ngay lập tức (bắt đồng bộ).
- pause: tín hiệu để báo bắt đầu hoặc dừng sinh số ngẫu nhiên.
- out: output của module với các số từ 0 đến 999. Sơ đồ thể hiện luồng xử lý như sau:





mô phỏng RTL trên modelsim



biểu đồ sóng trên modelsim

```
Transcript
File Edit View Bookmarks Window Help
# -- Compiling module clock_divide
#
# Top level modules:
# clock_divide
# vlog -vlogcompas -work work -incode E:/DOCUMENT/first year/sem2/TKL/assignment/New folder/triet2 (E:/DOCUMENT/first year/sem2/TKL/assignment/New folder/triet2/rand.v)
# Model Technology ModelSim ALTERA vlog 10.1d Compiler 2012.11 Nov 2 2012
# -- Compiling module rand
#
# Top level modules:
# rand
# vlog -vlogcompas -work work -incode E:/DOCUMENT/first year/sem2/TKL/assignment/New folder/triet2 (E:/DOCUMENT/first year/sem2/TKL/assignment/New folder/triet2/model.v)
# Model Technology ModelSim ALTERA vlog 10.1d Compiler 2012.11 Nov 2 2012
# -- Compiling module model
#
# Top level modules:
# model
#
add wave -position insertpoint \
  pin/model_clk/clock \
  pin/model_clk/reset \
  pin/model_clk/out \
  pin/model_clk/nvm
VSM P> run -all
#
# Time Out W00
# 0 999 0
# 0 971 0
# 0 924 0
# 0 825 0
# 100000000 627 0
# 100000000 290 0
# 100000000 461 0
# 100000000 903 0
# 100000000 823 0
# 100000000 623 0
# 100000000 222 0
# 100000000 440 0
# 100000000 890 0
```

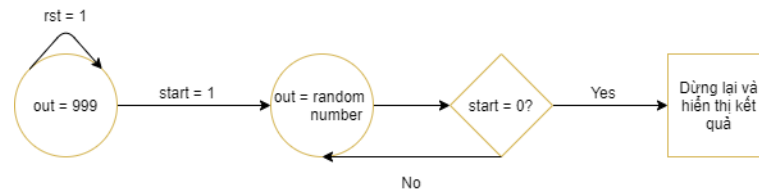
kết quả thu được trên modelsim



### 3.4 Module mode2:

Ở chế độ thứ 2, tính năng cũng tương tự như chế độ 1, chỉ khác biệt là trò chơi sẽ chỉ dừng lại khi người chơi thả nút nhấn. Như vậy, nút start cũng sẽ chính là tín hiệu dừng lại bất đồng bộ của chương trình. Do đó, thay vì sử dụng bộ đếm như ở mode1, ta chỉ cần đặt start là tín hiệu bất đồng bộ ở mode2 là trò chơi sẽ dừng lại theo ý muốn. Module mode2 gồm các tín hiệu sau:

- clk: xung clock đầu vào với tần số là 10HZ
- rst: tượng trưng cho nút reset. Khi tín hiệu rst = 1 thì module sẽ reset bất đồng bộ và trả về 999.
- start: chính là nút bắt đầu trò chơi. Khi người chơi nhấn nút trò chơi sẽ được bắt đầu khi có tín hiệu xung clock. Trò chơi sẽ dừng lại khi người chơi thả nút nhấn (bất đồng bộ).
- pause: tín hiệu để báo bắt đầu hoặc dừng sinh số ngẫu nhiên.
- out: output của module với các số từ 0 đến 999. Sơ đồ thể hiện luồng xử lý như sau:



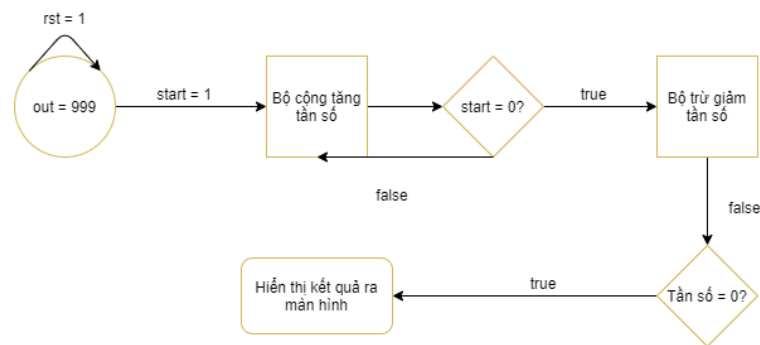


### 3.5 Module mode3:

Đây là chế độ thú vị nhất và cũng là chế độ phức tạp nhất khi tốc độ nhảy số sẽ tăng dần chừng nào người chơi còn giữ nút nhấn và sẽ chậm dần khi người chơi thả nút đến khi dừng lại.

Để có thể thay đổi tần số tăng dần hoặc giảm dần, ta sẽ sử dụng module clock-divide cùng một bộ đếm. Về tín hiệu, module mode3 cũng không khác nhiều so với mode1 và mode2:

- clk: xung clock đầu vào với tần số là 10HZ
- rst: tượng trưng cho nút reset. Khi tín hiệu rst = 1 thì module sẽ reset bắt đồng bộ và trả về 999.
- start: chính là nút bắt đầu trò chơi. Khi người chơi nhấn nút trò chơi sẽ được bắt đầu. Nếu người chơi còn giữ nút thì tốc độ nhảy số sẽ tăng dần. Khi người chơi thả nút, tốc độ nhảy số sẽ chậm dần cho đến khi dừng lại.
- pause: tín hiệu để báo bắt đầu hoặc dừng sinh số ngẫu nhiên.
- out: output của module với các số từ 0 đến 999. Sơ đồ thể hiện luồng xử lý như sau:



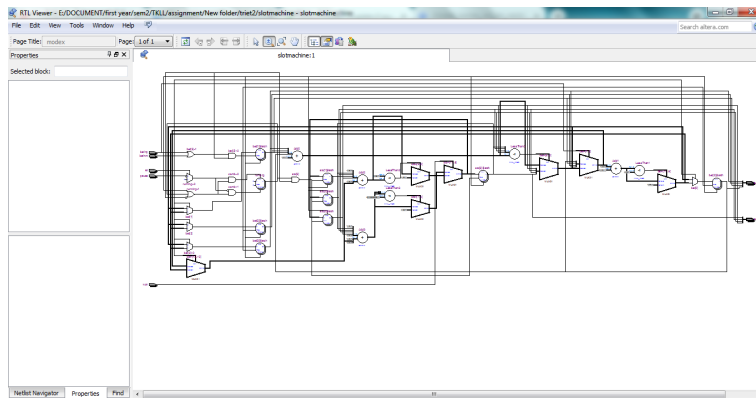


### 3.6 Module modex:

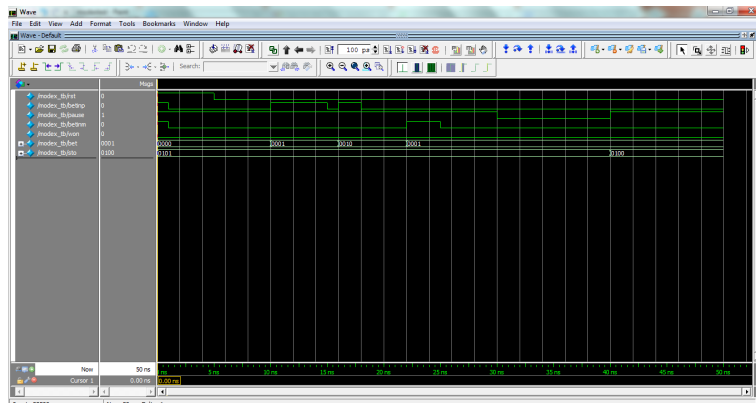
ModeX sẽ được chạy trong cả ba chế độ mode 1, mode 2, mode 3. Lấy cảm hứng từ máy đánh bạc quay số (slot machine), modex được thêm vào để trò chơi có thêm tính năng "đặt cược" giống phiên bản gốc.

Module modex gồm các tín hiệu sau:

- betinp: tăng số tiền cược (bet) lên một đơn vị
- betinm: giảm số tiền cược (bet) xuống một đơn vị
- pause: người chơi chỉ có thể đặt cược khi hệ thống đang không tạo số (pause = 1)
- won: nếu won = 1 thì bet sẽ được cộng vào sto, nếu won = 0 thì sto sẽ bị trừ đi bet
- rst: tượng trưng cho nút reset. Khi tín hiệu rst = 1 thì trả sto về 5/bet về 0
- bet: số tiền đặt cược mỗi vòng chơi
- sto: số tiền còn lại trong "ví" người chơi



mô phỏng RTL trên modelsim



biểu đồ sóng trên modelsim

```

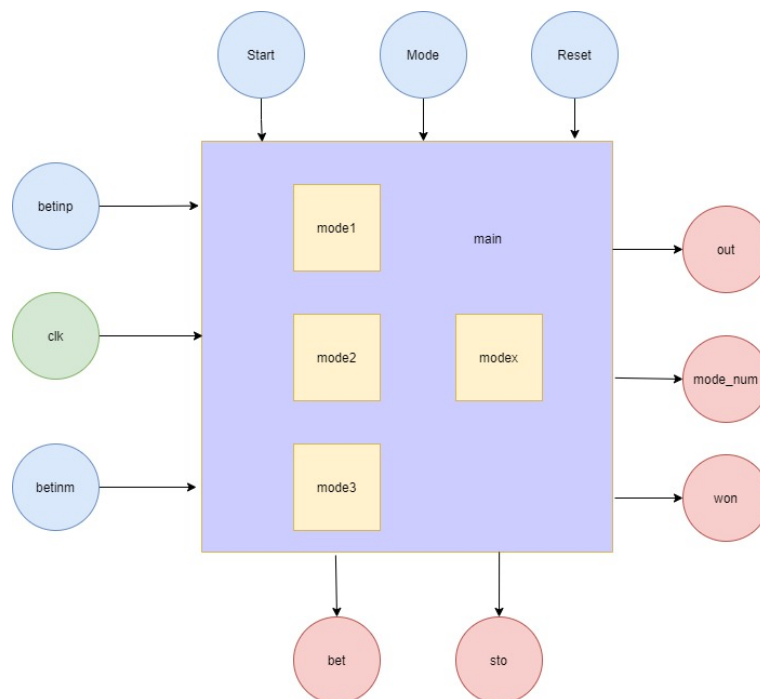
Transcript
File Edit View Bookmarks Window Help
Transcript
# -- Compiling module clock_divide
#
# Top level modules:
#   clock_divide
# Vlog -vlogcompat -work work +incdir+E:/DOCUMENT/first year/sem2/TKL/assignment/Dev folder/triet2 (E:/DOCUMENT/first year/sem2/TKL/assignment/Dev folder/triet2)
# Model Technology ModelSim ALTERA Vlog 10.1d Compiler 2012.11 Nov  2 2012
# -- Compiling module rand
#
# Top level modules:
#   rand
# Vlog -vlogcompat -work work +incdir+E:/DOCUMENT/first year/sem2/TKL/assignment/Dev folder/triet2 (E:/DOCUMENT/first year/sem2/TKL/assignment/Dev folder/triet2)
# Model Technology ModelSim ALTERA Vlog 10.1d Compiler 2012.11 Nov  2 2012
# -- Compiling module model
#
# Top level modules:
#   model
# with wave -position lowerpoints \
# s1st/modex_ch/rst \
# s1st/modex_ch/betinp \
# s1st/modex_ch/pause \
# s1st/modex_ch/betinm \
# s1st/modex_ch/won \
# s1st/modex_ch/bet \
# s1st/modex_ch/sto
VSM> run -all
#
# Time betinp pause betinm won rst bet sto
# 0: 0 0 0 0 0 0 0
# 10000: 0 1 0 0 0 1 0
# 20000: 0 1 0 0 0 1 0
# 30000: 0 1 0 0 0 0 1
# 40000: 1 0 0 0 0 0 1
# 50000: 1 0 0 0 0 0 1
# 60000: 1 0 0 0 0 0 1
# 70000: 1 0 0 0 0 0 1
# 80000: 1 0 0 0 0 0 1
# 90000: 0 1 0 0 0 0 1
# 100000: 0 1 0 0 0 0 1
# 110000: 0 1 0 0 0 0 1
# 120000: 0 1 0 0 0 0 1
# 130000: 0 1 0 0 0 0 1
# 140000: 0 1 0 0 0 0 1
# 150000: 0 1 0 0 0 0 1
# 160000: 0 1 0 0 0 0 1
# 170000: 0 1 0 0 0 0 1
# 180000: 0 1 0 0 0 0 1
# 190000: 0 1 0 0 0 0 1
# 200000: 0 1 0 0 0 0 1
# 210000: 0 1 0 0 0 0 1
# 220000: 0 1 0 0 0 0 1
# 230000: 0 1 0 0 0 0 1
# 240000: 0 1 0 0 0 0 1
# 250000: 0 1 0 0 0 0 1
# 260000: 0 1 0 0 0 0 1
# 270000: 0 1 0 0 0 0 1
# 280000: 0 1 0 0 0 0 1
# 290000: 0 1 0 0 0 0 1
# 300000: 0 1 0 0 0 0 1
# 310000: 0 1 0 0 0 0 1
# 320000: 0 1 0 0 0 0 1
# 330000: 0 1 0 0 0 0 1
# 340000: 0 1 0 0 0 0 1
# 350000: 0 1 0 0 0 0 1
# 360000: 0 1 0 0 0 0 1
# 370000: 0 1 0 0 0 0 1
# 380000: 0 1 0 0 0 0 1
# 390000: 0 1 0 0 0 0 1
# 400000: 0 1 0 0 0 0 1
VSM>
    
```

kết quả thu được trên modelsim

### 3.7 Module main

Sau khi đã hiện thực xong các module con, ta cần kết nối tất cả vào một module chính - module main. Module này cũng sẽ là nơi thể hiện các tín hiệu giao tiếp với người dùng đồng thời hiển thị các kết quả trong trò chơi. Về tín hiệu, module main đáp ứng các yêu cầu về thay đổi chức năng và hiển thị hệ thống:

- clk: nhận tín hiệu xung clock 50MHZ từ kit.
- mode: kết nối với button trên kit, mỗi lần nhấn tương ứng với một lần thay đổi mode.
- start: nút bắt đầu trò chơi.
- rst: nút reset.
- betinp: nút đặt cược. Mỗi lần nhấn sẽ tăng tiền cược
- betinm: nút đặt cược. Mỗi lần nhấn sẽ giảm tiền cược.
- bet: thể hiện số tiền đặt cược thông qua led 7 đoạn.
- sto: thể hiện số tiền người chơi hiện có thông qua led 7 đoạn.
- out: thể hiện các con số được sinh ngẫu nhiên thông qua led 7 đoạn.
- mode-num: thể hiện chế độ người chơi đang chọn thông qua led 7 đoạn.
- won: báo hiệu kết quả cho người chơi bằng led đơn. Ví dụ xanh là thắng, đỏ là thua.



sơ đồ chức năng sau khi đã thiết kế xong



## 4 Kết luận

### 4.1 Kết luận

Qua đề tài, nhóm biết được thêm cơ chế tạo số bán ngẫu nhiên (pseudo - random) và ôn lại những kiến thức cũ của môn học thiết kế luận lý của học kỳ vừa qua.

### 4.2 Hướng phát triển tương lai

Phát triển thêm các chức năng khác:

- Quay kí tự thay vì quay số
- Hiện thị trò chơi thông qua màn hình giao tiếp VGA
- Tăng mức tiền đặt cược và trong kho: số hai chữ số hoặc ba chữ số

### 4.3 Khó khăn gặp phải

Không có công cụ để thí nghiệm vì phòng thí nghiệm không hoạt động  
Giải pháp: mô phỏng bằng modelSim

### 4.4 Phân chia công việc

Hoàng Minh Triết:

- Viết module chính, mode 1, mode 2, mode 3, clock divide và rand
- Vẽ sơ đồ khối

Nguyễn Xuân Nam:

- Viết module modex
- Soạn báo cáo, slides



## 5 Tài liệu tham khảo

- Bài giảng trên lớp
- Tài liệu từ các buổi Lab
- Sách và trang web từ Internet