Tytuł: Gra ping-pong

Autor: Marek Kuros

Ostatnia modyfikacja: 09.08.2023

Spis treści

[1. Repozytorium git 1](#__RefHeading___Toc1311_2642343945)

[2. Wstęp 1](#__RefHeading___Toc375_832888478)

[3. Specyfikacja](#__RefHeading___Toc377_832888478) 2

[3.1. Opis ogólny algorytmu](#__RefHeading___Toc379_832888478) 2

[3.2. Tabela zdarzeń](#__RefHeading___Toc381_832888478) 3

[4. Architektura](#__RefHeading___Toc383_832888478) 4

[4.1. Moduł: top](#__RefHeading___Toc385_832888478) 4

[4.1.1. Schemat blokowy](#__RefHeading___Toc387_832888478) 4

[4.1.2. Porty](#__RefHeading___Toc389_832888478) 5

[4.1.3. Interfejsy](#__RefHeading___Toc453_832888478) 7

[4.2. Rozprowadzenie sygnału zegara](#__RefHeading___Toc1150_832888478) 8

[5. Implementacja](#__RefHeading___Toc1313_2642343945) 8

[5.1. Lista zignorowanych ostrzeżeń Vivado.](#__RefHeading___Toc1315_2642343945) 8

[5.2. Wykorzystanie zasobów](#__RefHeading___Toc1317_2642343945) 8

[5.3. Marginesy czasowe](#__RefHeading___Toc1319_2642343945) 9

[6. Film.](#__RefHeading___Toc1249_1436539759) 9

# Repozytorium git

Adres repozytorium GITa:

<https://github.com/marek-kuros/Project_UEC2.git>

<https://github.com/marek-kuros/Project_UEC2/tree/files_to_send>

(w branchu **main** znajdują się dodatkowe pliki np. program do użycia akcelerometru, natomiast w **files\_to\_send** pliki są ułożone zgodnie ze specyfikacją)

W przypadku repozytorium prywatnego należy zaprosić użytkownika zewnętrznego o adresie mailowym: [kaczmarczyk@agh.edu.pl](mailto:kaczmarczyk@agh.edu.pl)

# Wstęp

Dawno temu usłyszałem, że najlepszym sposobem na sprawdzenie, czy bardzo dobrze opanowało się dany język programowania, jest napisanie gry w ping-ponga. Dzięki temu projektowi mogłem dokonać takowego testu moich umiejętności pisania w języku System Verilog. Dodatkowo, w celu rozbudowy gry, zaimplementowałem ciekawy tryb dla pojedynczego gracza. Polega on na tym, że zamiast drugiego gracza jest paletka poruszająca się odwrotnie do pierwszej, a zadanie polega na jak najdłuższym odbijaniu piłki.

# Specyfikacja

## Opis ogólny algorytmu

Gra = Idle

Czy sw[1]=1

Czy sw[0]=1

Gra = Single

Gra = Multi

Czy sw[0]=0

Czy sw[1]=0

## Tabela zdarzeń

|  |  |  |
| --- | --- | --- |
| **Zdarzenie** | **Kategoria** | **Reakcja systemu** |
| Sw[0] przesunięte w stan HIGH | Idle | Uruchomienie trybu singleplayer |
| Sw[1] przesunięte w stan HIGH | Idle | Uruchomienie trybu multiplayer |
| Sw[0:1] przesunięte w stan HIGH | Idle | Pozostanie w trybie ekranu startowego |
| Naciśnięcie btnR | Single | Piłka zostaje zaserwowana (leci w randomową stronę, po 15% prawdopodobieństwa, że poleci po skosie i po 20%, że poleci prostopadle do linii paletek) |
| Odbicie od górnej lub dolnej krawędzi | Single | Zmiana kierunku lotu piłki |
| Odbicie od paletki | Single | Piłka zmienia kierunek lotu, zależnie od punktu, w którym paletka została uderzona. |
| Piłka przelatuje za linię paletek | Single | Piłka dolatuje do końca, a następnie znika, by zasygnalizować przegraną (7 segment wyświetla kod 0101) |
| Naciśnięcie btnR po przegranej | Single | Gra zaczyna się od nowa |
| Naciśnięcie btnR | Multi | Piłka zostaje zaserwowana (leci w randomową stronę, po 15% prawdopodobieństwa, że poleci po skosie i po 20%, że poleci prostopadle do linii paletek) |
| Odbicie od górnej lub dolnej krawędzi | Multi | Zmiana kierunku lotu piłki |
| Odbicie od paletki | Multi | Piłka zmienia kierunek lotu, zależnie od punktu, w którym paletka została uderzona. |
| Piłka przelatuje za linię paletek | Multi | Piłka dolatuje do końca, a następnie ponownie pojawia się na starcie. Gracz otrzymuje punkt.  (aktualna liczba punktów jest wyświetlana na disp[0] oraz disp[2] – 7segment) |
| Gracz osiąga 11 punktów | Multi | Piła znika w celu zasygnalizowania końca rozgrywki. Wyświetlacz siedmiosegmentowy disp[1] oraz disp[3] wyświetla, który gracz jest zwycięzcą, a disp[0] oraz disp[2] wskazuje ostateczny wynik |
| Naciśnięcie btnR po przegranej | Multi | Gra zaczyna się od nowa. Wynik zostaje wyzerowany. |
| Sw[0] przesunięte w stan LOW | Single | Wyjście do ekranu głównego(reset gry) |
| Sw[1] przesunięte w stan LOW | Multi | Wyjście do ekranu głównego(reset gry) |
| Sw[1] przesunięte w stan HIGH | Single | Wyjście do ekranu głównego(reset gry) |
| Sw[0] przesunięte w stan HIGH | Multi | Wyjście do ekranu głównego(reset gry) |

# Architektura

## Moduł: top, ball\_control, select\_game\_sm, draw\_bg, ball, char\_rom\_16x16, draw\_rect\_char, vga\_timing, vga\_if, draw\_rect\_ctl, sync, vga\_pkg

Osoba odpowiedzialna: MK

### Schemat blokowy

Vga\_if

Vga\_if

Draw\_bg

Vga\_timing

Ypos\_out

Sync

PS2\_data

Ypos

MouseCtl

Y\_pos\_player\_2

Vga\_if

End\_of\_frame

Draw\_rect\_ctl

Ypos\_out\_player2

Ball\_control

Screen\_idle

Screen\_multi

Screen\_single

Vga\_if

Draw\_ball

sw

Y\_pos\_of\_ball

Select\_game\_sm

Vga\_if

Draw\_rect\_char

Vga\_if

X\_pos\_of\_ball

Char\_line

Char\_xy

serve

Char\_rom\_16x16

addr

Points\_player\_1

Font\_rom

Who\_won

Points\_player\_2

Char\_pixel

Disp\_hex\_mux

7segment

An

### Porty

#### Vga\_timing, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| End\_of\_frame | Sygnał informujący o zakończeniu rysowania klatki |

#### Draw\_bg, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Screen\_single | Wybranie tła do gry |
| Screen\_multi | Wybranie tła do gry |

#### MouseCtl, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| PS2\_data | Szeregowe wejście danych(port komunikacyjny PS2) |

#### MouseCtl, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Ypos | Informacja o pozycji gracza |

#### Sync, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Ypos | Informacja o pozycji gracza |

#### Sync, outpu

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Ypos\_out | Informacja o pozycji gracza |

#### Draw\_rect\_ctl, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Ypos\_player1 | Informacja o pozycji gracza |
| Screen\_multi | Wybranie kontroli rysowania paletek trybu miltiplayer |
| Screen\_single | Wybranie kontroli rysowania paletek trybu singleplayer |

#### Draw\_rect\_ctl, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Ypos\_out\_player2 | Informacja o pozycji gracza do komunikacji |

#### Draw\_ball, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Screen\_idle | Sygnał aktywujący rysowanie piłki |
| X\_pos\_of\_ball | Pozycja piłki w osi X |
| Y\_pos\_of\_ball | Pozycja piłki w osi Y |

#### Draw\_rect\_char, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Screen\_idle | Sygnał aktywujący moduł piszący |
| Char\_pixel | Rysowanie danej literki |

#### Draw\_rect\_char, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Char\_xy | Wybieranie litery z tablicy |
| Char\_line | Sterowanie linią litery |

#### Char\_rom\_16x16, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Char\_xy | Wybieranie litery z tablicy |
| Char\_line | Sterowanie linią litery |

#### Char\_rom\_16x16, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| addr | Addres litery |

#### Font\_rom, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| addr | Addres litery |

#### Font\_rom, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Char\_pixel | Rysowanie danej literki |

#### Disp\_hex\_mux, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Points\_player\_1 | Ilość punktów do wyświetlenia dla gracza 1 |
| Points\_player\_2 | Ilość punktów do wyświetlenia dla gracza 2 |
| Who\_won | Wyświetlenie zwycięzcy rozgrywki |

#### Disp\_hex\_mux, Output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| 7segment | Sterowanie wyświetlaczem 7segment |
| An | Sterowanie zmultipleksowanymi 4 wyświetlaczami |

#### Select\_game\_sm, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Sw | Sterowanie trybem gry |

#### Select\_game\_sm, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Screen\_idle | Sygnał gry stop |
| Screen\_multi | Sygnał gry dla dwóch graczy |
| Screen\_single | Sygnał gry dla pojedynczego gracza |

#### Ball\_control, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Serve | Sygnał do zaserwowania piłki |
| Screen\_idle | Sygnał aktywujący moduł |
| Screen\_multi | Sygnał do wyboru trybu gry |
| Y\_pos\_player\_1 | Pozycja gracza 1 |
| Y\_pos\_player\_2 | Pozycja gracza 2 |
| End\_of\_frame | Sygnał aktualizacji pozycji piłki |

#### Ball\_control, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| Points\_player\_1 | Ilość punktów dla gracza 1 |
| Points\_player\_2 | Ilość punktów dla gracza 2 |
| Who\_won | Wyświetlenie zwycięzcy rozgrywki |
| X\_pos\_of\_ball | Pozycja piłki w osi X |
| Y\_pos\_of\_ball | Pozycja piłki w osi Y |

### Interfejsy

#### Vga\_if

|  |  |
| --- | --- |
| **nazwa sygnału** | **opis** |
| Rgb[11:0] | Sygnał sterujący kolorem piksela |
| Hblnk | Sygnał sterujący blankiem horyzontalnym |
| Hsync | Sygnał synchronizacji horyzontalnej |
| Hcount [10:0] | Licznik horyzontalny |
| Vblnk | Sygnał sterujący blankiem wertykalnym |
| Vsync | Sygnał synchronizacji wertykalnej |
| Vcount [10:0] | Licznik wertykalny |

## Rozprowadzenie sygnału zegara

Osoba odpowiedzialna: MK

Vga\_timing

Draw\_bg

Sync

Draw\_rect\_ctl

MouseCtl

Ball\_control

Draw\_ball

Char\_rom\_16x16

100MHz

Draw\_rect\_char

65MHz

Clk\_gen

Font\_rom

Disp\_hex\_mux

Select\_game\_sm

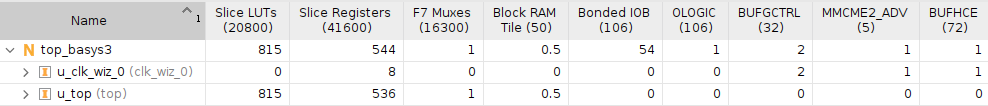
# Implementacja

## Lista zignorowanych ostrzeżeń Vivado.

|  |  |  |
| --- | --- | --- |
| Identyfikator ostrzeżenia | Liczba wystąpień | Uzasadnienie |
| [Synth 8-7080] | 1 | <https://support.xilinx.com/s/article/72150?language=en_US#:~:text=This%20warning%20indicates%20that%20the,flow%20will%20not%20be%20run>.  Projekt jest zbyt mały by skorzystać z równoległej syntezy |
|  |  |  |

## Wykorzystanie zasobów

Tabela z wykorzystaniem zasobów z Vivado:



## Marginesy czasowe

Worst negative slack (WNS) dla Setup = 7.368ns

Worst negative slack (WNS) dla Hold = 0.007ns

# Film.

Link do ściągnięcia filmu:

https://