Tytuł: Basys Invaders

Autorzy: Tomasz Sieja (TS), Antoni Sus (AS)

Ostatnia modyfikacja: 27.08.2025

Spis treści

[1. Repozytorium git 1](#__RefHeading___Toc1311_2642343945)

[2. Wstęp 1](#__RefHeading___Toc375_832888478)

[3. Specyfikacja 1](#__RefHeading___Toc377_832888478)

[3.1. Opis ogólny algorytmu 1](#__RefHeading___Toc379_832888478)

[3.2. Tabela zdarzeń 2](#__RefHeading___Toc381_832888478)

[4. Architektura 2](#__RefHeading___Toc383_832888478)

[4.1. Moduł: top 2](#__RefHeading___Toc385_832888478)

[4.1.1. Schemat blokowy 2](#__RefHeading___Toc387_832888478)

[4.1.2. Porty 3](#__RefHeading___Toc389_832888478)

[a) mou – mouse\_ctl, input 3](#__RefHeading___Toc391_832888478)

[b) vga – vga\_ctl, output 3](#__RefHeading___Toc393_832888478)

[4.1.3. Interfejsy 3](#__RefHeading___Toc453_832888478)

[a) m2c – mouse\_ctl to core 3](#__RefHeading___Toc395_832888478)

[4.2. Rozprowadzenie sygnału zegara 3](#__RefHeading___Toc1150_832888478)

[5. Implementacja 4](#__RefHeading___Toc1313_2642343945)

[5.1. Lista zignorowanych ostrzeżeń Vivado. 4](#__RefHeading___Toc1315_2642343945)

[5.2. Wykorzystanie zasobów 4](#__RefHeading___Toc1317_2642343945)

[5.3. Marginesy czasowe 4](#__RefHeading___Toc1319_2642343945)

[6. Film. 4](#__RefHeading___Toc1249_1436539759)

# Repozytorium git

Adres repozytorium GITa:

https://github.com/Antoni-S/basys-invaders

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

# Wstęp

*Skąd się wziął pomysł i co w ramach tego projektu robimy.*

*Pomysł na tę grę pojawił się w momencie gdy dowiedzieliśmy się, że projekt może być prostą grą na płytkę FPGA. Jest to dobra okazja aby powrócić do gier klasycznych i spróbować naszych sił w odwzorowaniu ich. W ramach tego projektu staramy się odwzorować Space Invaders, ale z dodanym trybem kooperacyjnym i lekko zmienionym wyglądem.*

# Specyfikacja

## Opis ogólny algorytmu

*Uproszczony schemat blokowy działania implementowanego algorytmu. Co się dzieje po starcie, jak wygląda przebieg działania, kiedy i pod jakimi warunkami się kończy.*

*Ewentualnie przykładowe screen-shoty tego, co w przybliżeniu chcielibyśmy uzyskać.*

## Tabela zdarzeń

|  |  |  |
| --- | --- | --- |
| **Zdarzenie** | **Kategoria** | **Reakcja systemu** |
| Przycisk Enter | Ekran startowy | Uruchomienie gry |
| Przycisk „A” | Klawiatura | Statek porusza się w lewo |
| Przycisk „D” | Klawiatura | Statek porusza się w prawo |
| Przycisk „W” | Klawiatura | Statek wypuszcza pocisk, zapisywana jest pozycja gracza |
| Wystrzelony pocisk | Gracz | Pocisk porusza się w prostej linii do góry, w pozycji X w której gracz kliknął przycisk „W”. Gracz nie może wystrzelić kolejnego pocisku |
| Pocisk zderza się z przeciwnikiem | Gracz | Przeciwnik zostaje usunięty z „planszy”. Gracz może wystrzelić kolejny pocisk |
| Pocisk zderza się z górną krawędzią ekranu | Gracz | Pocisk zostaje usunięty. Gracz może wystrzelić kolejny pocisk |
| Wszyscy przeciwnicy usunięci z „planszy” | Gra | Zakończenie gry, ekran zmienia się na ekran wygranej |
| Przeciwnik zdążył przejść na dolną część ekranu (odległość od dolnej części ekranu około 70px) | Gra | Zakończenie gry, ekran zmienia się na ekran przegranej |

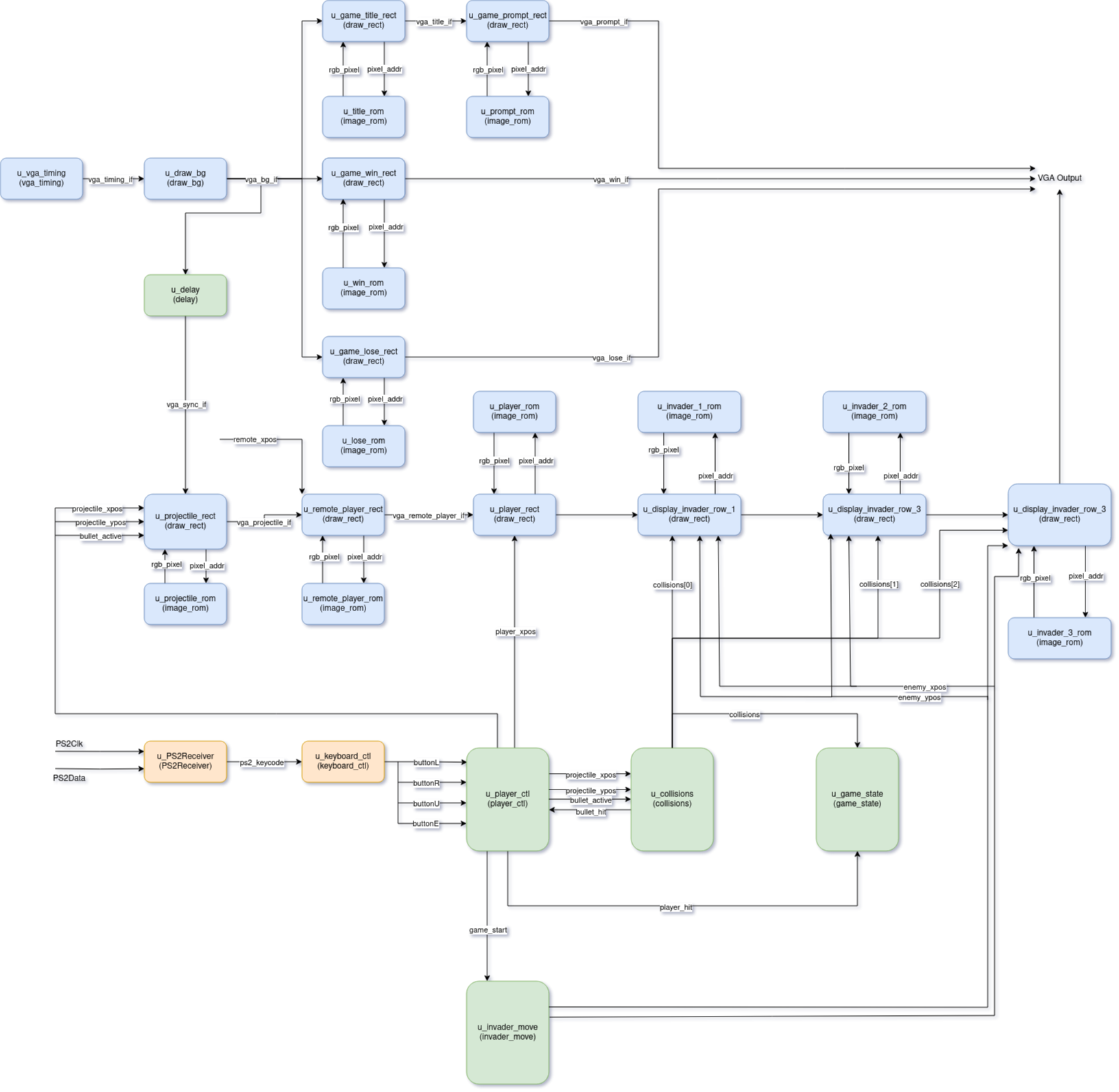
# Architektura

## Moduł: top

Osoba odpowiedzialna: AS, TS

*Moduł top\_vga jest złożony z podmodułów odpowiedzialnych za wyświetlanie elementów na ekranie, kontrolowanie ich oraz z modułów pomocniczych (np. pamięci ROM lub modułu delay).*

### Schemat blokowy



### Porty

#### keyboard – keyboard\_ctl, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| PS2Data | szeregowe wejście danych z klawiatury |
| PS2Clk | zegar klawiatury |

#### vga – vga\_ctl, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| vga\_vs | sygnał synchronizacji pionowej VGA |
| vga\_hs | sygnał synchronizacji poziomej VGA |
| vga\_r [3:0] | sygnał koloru czerwonego VGA |
| vga\_g [3:0] | sygnał koloru zielonego VGA |
| vga\_b [3:0] | sygnał koloru niebieskiego VGA |

#### uart – uart\_ctl, input

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| test\_Rx | sygnał wejściowy UART używany do testu poprawności działania |
| JB1 | sygnał wejściowy UART odpowiedzialny za przekazywanie pozycji gracza |
|  |  |

#### uart – uart\_ctl, output

|  |  |
| --- | --- |
| **nazwa portu** | **opis** |
| test\_Tx | sygnał wyjściowy UART używany do testu poprawności działania |
| JC1 | sygnał wyjściowy UART odpowiedzialny za przekazywanie pozycji gracza |
|  |  |

### Interfejsy

#### vga\_if – vga signal pipeline

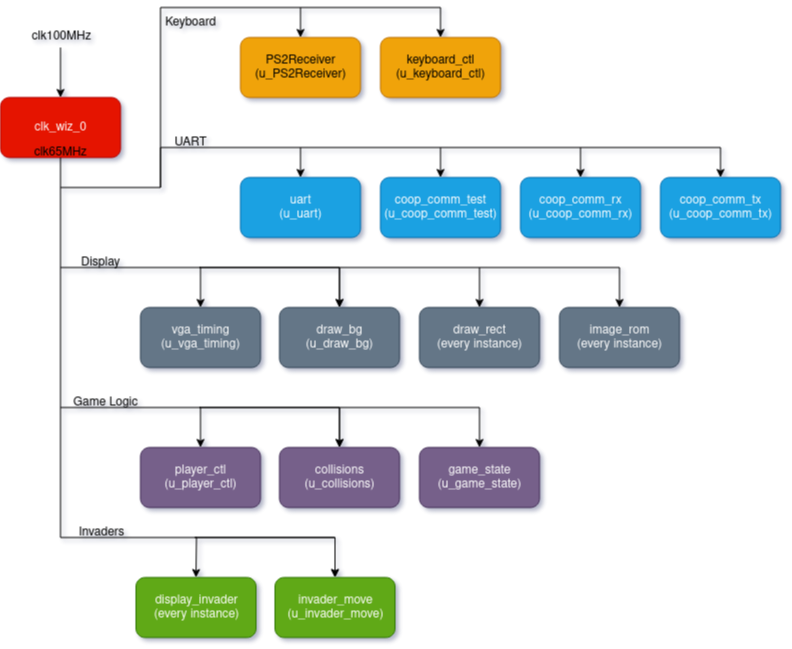
|  |  |
| --- | --- |
| **nazwa sygnału** | **opis** |
| hcount [10:0] | licznik horyzontalny VGA |
| vcount [10:0] | licznik wertykalny VGA |
| hsync | synchronizacja pozioma VGA |
| vsync | synchronizacja pionowa VGA |
| hblnk | sygnał blank poziomy VGA |
| vblnk | sygnał blank pionowy VGA |
| rgb [11:0] | sygnał z informacją o kolorze |

*Każdy moduł, który w jakikolwiek sposób jest rysowany (wyświetlany) na ekranie korzysta z powyższego interfejsu vga\_if. Dla czytelności, każdy interfejs w projekcie jest nazywany zgodnie z konwencją vga\_[element\_wyświetlany]\_if. Dla przejrzystości tego raportu, nie będzie wymieniana każda instancja tego interfejsu.*

## Rozprowadzenie sygnału zegara

Osoba odpowiedzialna: TS

*Cały projekt działa na zegarze o częstotliwości 65 MHz, który został wygenerowany za pomocą Clock Wizard-a w Vivado. Wartość tej częstotliwości wynika z wymagania projektu dyktującego minimalną rozdzielczość monitora jako 1024x768 pikseli.*



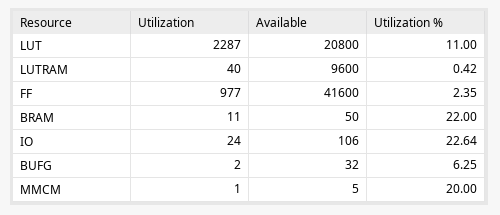
# Implementacja

## Lista zignorowanych ostrzeżeń Vivado.

|  |  |  |
| --- | --- | --- |
| Identyfikator ostrzeżenia | Liczba wystąpień | Uzasadnienie |
| 8-7080 | 1 | Projekt nie jest na tyle rozbudowany, aby spełniać zasady równoległej syntezy |
| 8-7129 | 2 | Zegar 100 MHz nie jest używany w projekcie |

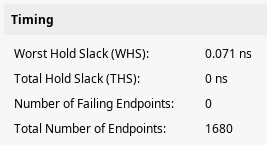
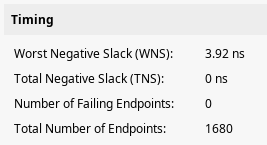
## Wykorzystanie zasobów

Tabela z wykorzystaniem zasobów z Vivado



## Marginesy czasowe

Marginesy czasowe (WNS) dla setup i hold.



# Konfiguracja sprzętu

Schemat połączenia ze sobą płytek Basys3 w trybie multiplayer.

Schematy podłączenia dodatkowych urządzeń peryferyjnych.

Konfiguracja zworek, przełączników, itp., jeśli inna niż domyślna.

# Film.

Link do ściągnięcia filmu:

https://