

Tytuł: **Tetris**

Autorzy: **Michał Kowalik (MK), Monika Dutkowska (MD)**

Ostatnia modyfikacja: 10.06.2017

Spis treści

1. Wstęp.....	1
2. Specyfikacja.....	1
2.1. Opis ogólny algorytmu.....	1
2.2. Tabela zdarzeń.....	2
3. Architektura.....	2
3.1. Moduł: top.....	2
3.1.1. Schemat blokowy.....	2
3.1.2. Porty.....	2
a) Keybrd – Keyboard_Ctl, input.....	2
b) VGA – VGA_Ctl, output.....	2
3.1.3. Interfejsy.....	3
a) Ct2C – Keyboard_Ctl to core.....	3
b) C2Ct – core to Keyboard_Ctl.....	3
3.2. Moduł: Keyboard_Ctl (external IP).....	3
3.3. Moduł: core.....	3
3.3.1. Schemat blokowy.....	3
3.3.2. Porty.....	3
3.3.3. Interfejsy.....	3
3.4. Moduł: VGA_Ctl.....	3
3.5. Rozprowadzenie sygnału zegara.....	3
4. Implementacja. Zaawansowanie na 10.06.2017 – 99%.....	4
5. Film. Zaawansowanie na 10.06.2017 – 0%.....	4

1 Wstęp

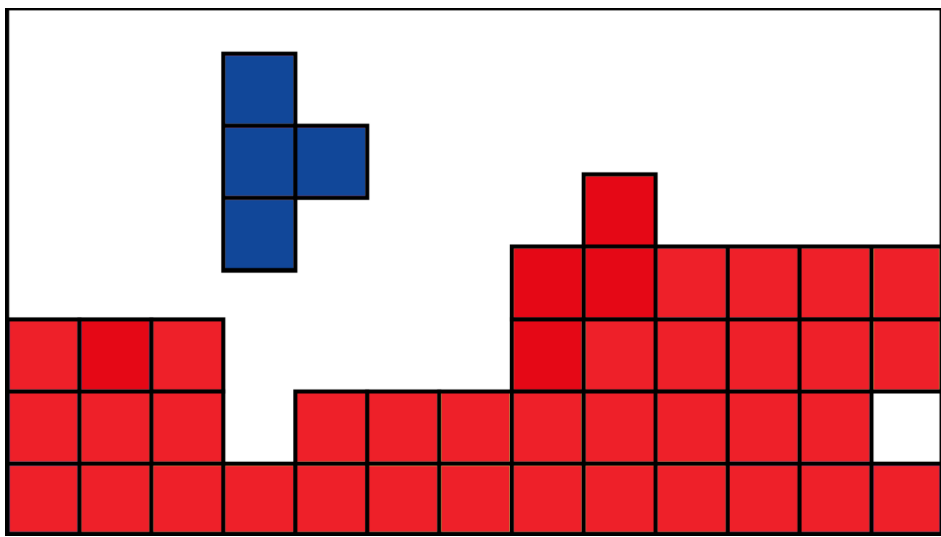
W ramach projektu zdecydowaliśmy się na stworzenie własnej wersji bardzo popularnej gry Tetris. Jest to gra polegająca na układaniu w poziomie spadających obiektów o różnych kształtach.

2 Specyfikacja

2.1 Opis ogólny algorytmu

Po wygenerowaniu klocka moduł New_brick sprawia że spada on z ustaloną prędkością pionowo, dodatkowo pozwala on graczowi na sterowanie nim w poziomie. Po otrzymaniu sygnału o dotknięciu od dołu jednego z klocków nieruchomych, zapisanych się w pamięci, następuje sprawdzenie, czy wszystkie pola w tym poziomie są wypełnione. Następuje wtedy aktualizacja stanu nieruchomych klocków.

Przykładowy cel realizacji naszego projektu:



2.2 Tabela zdarzeń

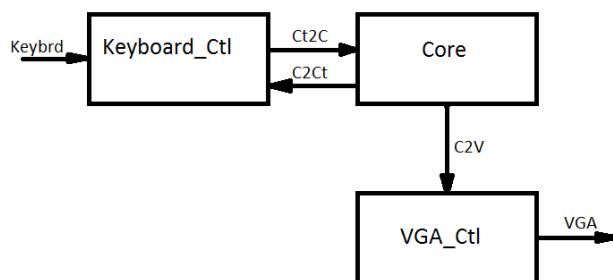
Zdarzenie	Kategoria	Reakcja systemu
Wciśnięcie strzałki w dół	Ekran startowy	Uruchomienie gry
Wciśnięcie strzałki lewo/prawo	Gra	Zmiana położenia spadającego klocka w poziomie
Dotknięcie spadającego klocka z nieruchomymi	Gra	Zatrzymanie spadającego klocka, sprawdzenie, czy każde pole w poziomie jest zajęte, jeśli nie, zapisanie stanu klocków do pamięci
Zajęty cały poziom	Gra	Usunięcie tego jednego poziomu i zapisanie stanu klocków, zaktualizowanie stanu punktów
Przekroczenie poziomej kreski	Gra	Game over, włączenie ekranu startowego

3 Architektura

3.1 Moduł: top

Osoba odpowiedzialna:

3.1.1 Schemat blokowy



3.1.2 Porty

a) Keyboard – Keyboard_Ctl, input

nazwa portu	opis
Keyboard_data	szeregowe wejście danych
Keyboard_clk	Zegar dla PS2

b) VGA – VGA_Ctl, output

nazwa portu	opis
h_sync	sygnał synchronizacji pionowej VGA
v_sync	Sygnał synchronizacji poziomej VGA
Red[3:0]	Wyjście koloru czerwonego
Blue[3:0]	Wyjście koloru niebieskiego
Green[3:0]	Wyjście koloru zielonego
h_blnk	Sygnał wygaszania pionowy
v_blnk	Sygnał wygaszania poziomy

3.1.3 Interfejsy

a) Ct2C – Keyboard_Ctl to core

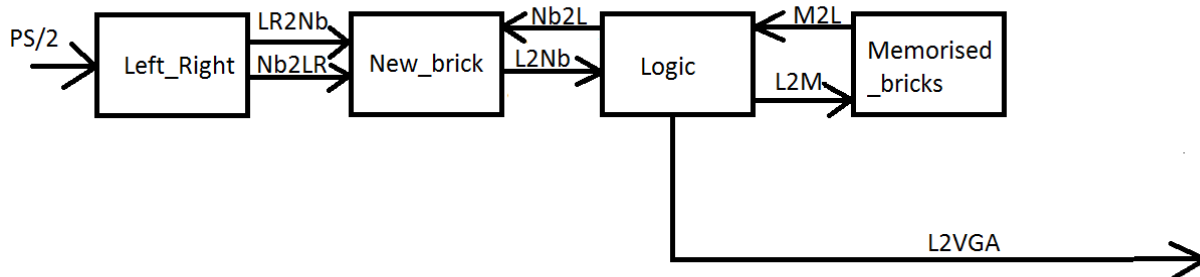
nazwa sygnału	opis
Key_pressed	Wciśnięty klawisz na klawiaturze
Key_code[15:0]	Kod ACSII wciśnietego klawisza

3.2 Moduł: Keyboard_Ctl (external IP)

Osoba odpowiedzialna: Michał Kowalik

3.3 Moduł: core

3.3.1 Schemat blokowy



3.3.2 Porty

PS/2 – Left_Right, input

nazwa sygnału	opis
Key_pressed	Wciśnięty klawisz na klawiaturze
Key_code[15:0]	Kod ACSII wciśnietego klawisza

3.3.2.1

L2VGA – Logic, output

nazwa portu	opis
old_brick_tab[18x25]	Tablica z istniejącymi klockami
new_brick_tab[18x25]	Tablica z nowym klockiem

3.3.3 Interfejsy

a) Left_Right – New Brick (LR2Nb)

nazwa sygnału	opis
Left	Wciśnięty klawisz strzałki w lewo
Right	Wciśnięty klawisz strzałki w prawo

b) New Brick – Logic (Nb2L)

nazwa sygnału	opis
New_brick_tab[18x25]	Tablica z koordynatami spadającego klocka

c) Logic – Memory (L2M)

nazwa sygnału	opis
New_brick_tab[18x25]	Tablica z koordynatami spadającego klocka – propagacja sygnału
Tab_save_signal	Informacja o spadnięciu klocka i zapamiętaniu nowej wartości opadniętych klocków do pamięci
Tab_save_tab [18x25]	Tablica z koordynatami opadniętych klocków – należy zapisać w pamięci

d) Logic – New Brick (L2Nb)

nazwa sygnału	opis
new_brick_signal	Informacja o spadnięciu klocka i konieczności wygenerowaniu nowego

e) Logic – Memoet (L2M)

nazwa sygnału	opis
Old_brick_tab[18x25]	Tablica z koordynatami opadniętych klocków

3.3.4 Architektura wewnętrzna

3.3.4.1 Moduł Left_Right

Osoba odpowiedzialna: Monika Dutkowska

nazwa sygnału	opis
Left	Wciśnięty klawisz strzałki w lewo
Right	Wciśnięty klawisz strzałki w prawo
Key_pressed	Wciśnięty klawisz na klawiaturze
Key_code[15:0]	Kod ACSII wciśnietego klawisza

3.3.4.2 Moduł New_Brick

Osoba odpowiedzialna: Michał Kowalik

nazwa sygnału	opis
Left	Wciśnięty klawisz strzałki w lewo
Right	Wciśnięty klawisz strzałki w prawo
New_brick_tab[18x25]	Informacja o koordynatach spadającego klocka
New_brick_signal	Informacja czy stworzyć nowy klocek

3.3.4.3 Moduł Logic

Osoba odpowiedzialna: Michał Kowalik

nazwa sygnału	opis
Old_brick_tab_in[18x25]	Wysłanie istniejących klocków do sprawdzenia czy cały poziom wypełniony
Tab_save_tab[18x25]	Zapisanie stanu istniejących klocków
New_brick_tab_in[18x25]	Informacja o koordynatach spadającego klocka
New_brick_signal / Tab_save_signal	Informacja czy stworzyć nowy klocek
New_brick_tab_out[18x25]	Tablica z nowym klockiem (propagacja sygnału)

3.3.4.4 Moduł Memory

Osoba odpowiedzialna: Monika Dutkowska

nazwa sygnału	opis
Old_brick_tab[18x25]	Wysłanie istniejących klocków do zapisu
New_brick_tab_out[18x25]	Tablica z nowym klockiem (propagacja sygnału)
Old_brick_tab_in[18x25]	Tablica ze starymi klockami do zapisania
New_brick_signal	Informacja czy stworzyć nowy klocek

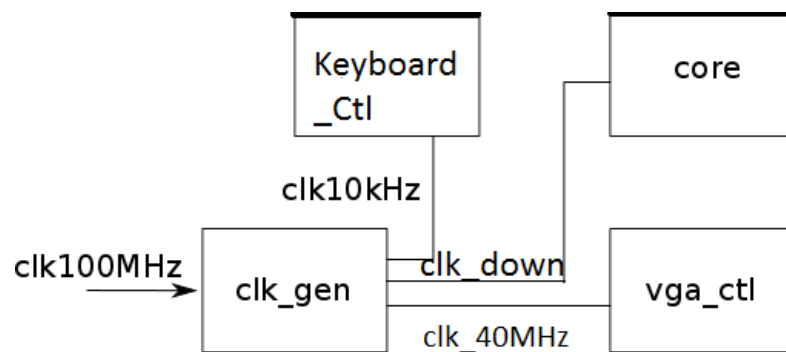
3.4 Moduł: VGA_Ctl

Osoba odpowiedzialna: Monika Dutkowska

nazwa portu	opis
h_sync	sygnał synchronizacji pionowej VGA
v_sync	Sygnał synchronizacji poziomej VGA
Red[3:0]	Wyjście koloru czerwonego
Blue[3:0]	Wyjście koloru niebieskiego
Green[3:0]	Wyjście koloru zielonego
h_blnk	Sygnał wygaszania pionowy
v_blnk	Sygnał wygaszania poziomy

3.5 Rozprowadzenie sygnału zegara

Osoba odpowiedzialna: Monika Dutkowska



4 Implementacja. Zaawansowanie na 10.06.2017 – 99%

<https://github.com/mkowalik/TetrisFPGA>

5 Film. Zaawansowanie na 10.06.2017 – 0%