

PRURE

Sprawozdanie

Laboratorium nr 2

Data: 21.11.2024

Zespół:

1. Anna Dzieżyk

Zadanie nr 1:

Opis wykonanych czynności:

(co należało wykonać, jakie techniki programowania zostały wykorzystane itp.)

Należało zaprojektować układ tzw. prescalera sygnału zegarowego o następujących właściwościach:

- Generowanie impulsu trwającego dokładnie 1 takt zegara co 1 sekundę

Wskazówka: licznik synchroniczny +1 o zdefiniowanej wartości wyzerowania licznika

Na potrzeby symulacji czas można skrócić: np. do 1 us

- Zaprojektować odpowiedni testbench symulując sygnał zegarowy dostępny na płycie

Basys 3 oraz opracowany komponent.

Niżej opisywane moduły zostały zaprojektowane w języku VHDL. Poniższa tabela ilustruje sygnały i stałe używane w obu modułach:

	licznik.vhd	licznik_tb.vhd
Zegar (wejściowy)	C	C
Reset asynchroniczny (wejściowy)	R	R
Sygnał sterujący wartością zakresu (wejściowy)	S	S
Sygnał wyjściowy	D	D
Zakres (wewnętrzny)	Z	-
Licznik (wewnętrzny)	L	-

Wynik syntezy:

(np. schemat, zajętość zasobów)

Moduł „licznik.vhd” nie był syntezerowany, był wywoływany tylko jako instancja modułu testbencha.

Opracowany kod:

(krótki opis głównych modułów)

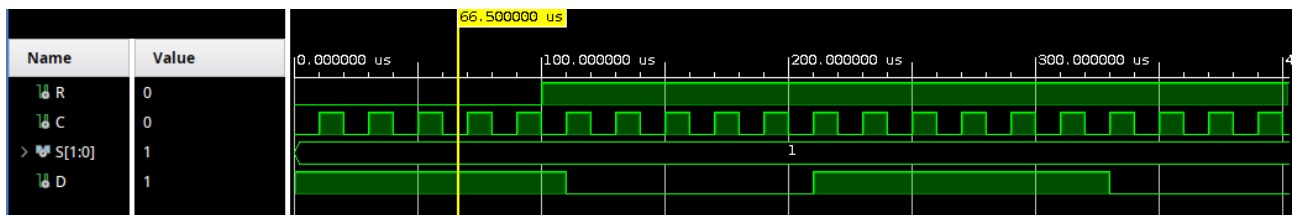
Zostały wykorzystane materiały dostarczone przez prowadzących – plik licznik.vhd, plik

licznik_tb.vhd oraz plik Basys-3-Master_lab2.xdc . Na początek zmodyfikowano plik licznik.vhd, by móc obserwować sygnały w testbenchu porównie – licznik, który miał za zadanie zliczanie taktów zegara został zmniejszony do 10.

Następnie wykonano edycję testbenchu - zainicjalizowano wartości sygnałów wejściowych – nakazano inicjalizację R (resetu), a następnie zwolniono sygnał resetu, po 100µs. Z symulowaniem zegara nic nie zmieniano – ten sygnał zmienia się co 10 µs. Przypisano wejściu S sygnał równy „10” , trwający przez cały czas symulacji.

Podsumowanie oraz wyniki:

(krótki opis działania układu, wynik testów, screenshot np. z programowania itp.)



Zrzut 1.: Przebiegi czasowe sygnałów wejściowych i obserwowanego wyjściowego

Symulacja trwała 5ms, natomiast sama zmiana stanu sygnału wyjściowego D następowała co 100µs. Został przetestowany tylko ten jeden sygnał – odwzorowujący 1:1 wartość stałej ilości taktów zegara T – deklarowanej w module „licznik.vhd” (S ustawiono na wartość „01”).

Zadanie nr 2:

Opis wykonanych czynności:

(co należało wykonać, jakie techniki programowania zostały wykorzystane itp.)

Na bazie komponentu z zadania 2 należało zaprojektować prosty zegarek na 3 diodach:

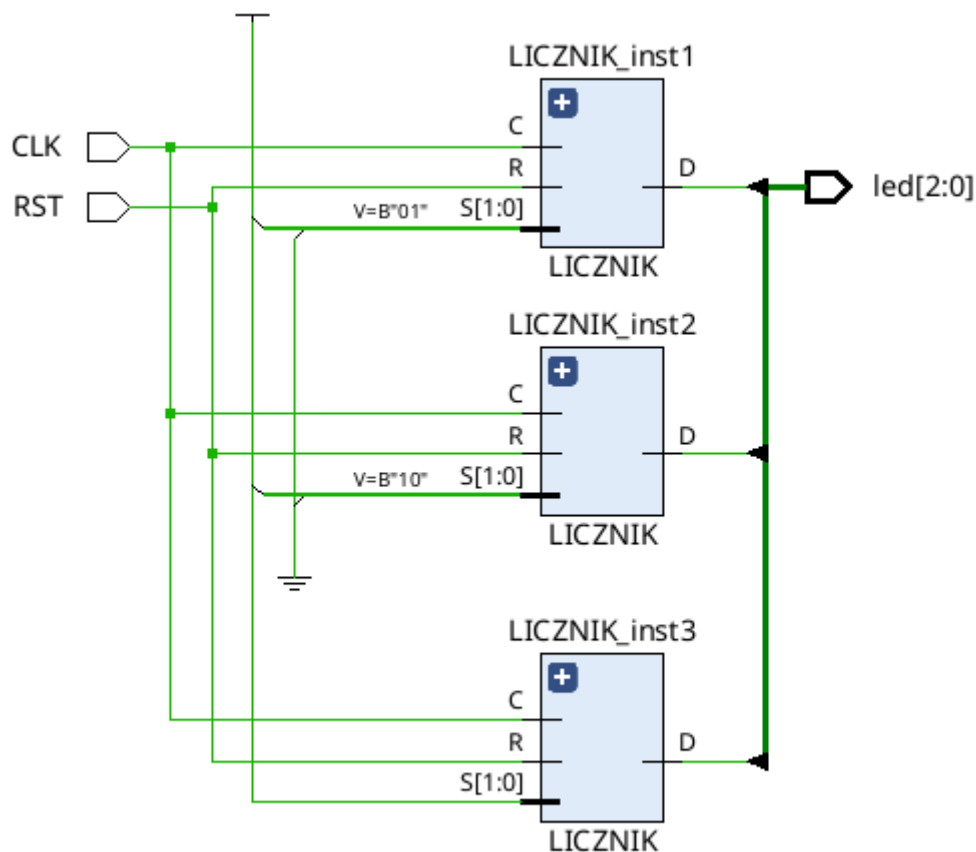
- LED 3 – godziny
- LED 2 – minuty
- LED 1 – sekundy

Zasada działania zegarka jest następująca: stan każdego elementu LED jest negowany (tzn. reverse state) co wskazującą wartość: tzn. co sekundę, co minutę i co godzinę.

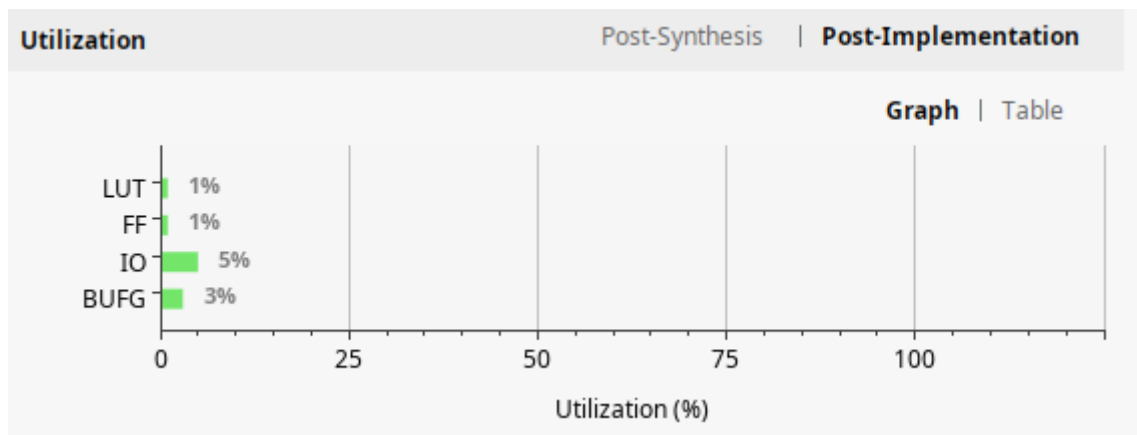
Sygnały obecne w powyższych rozważaniach zostały opisane w tabelce poniżej.

	Basys-3-Master_lab2.xdc	zegar.vhd	zegar_tb.vhd
Sygnał zegara (wejściowy)	CLK	C	CLK
Sygnał resetu asynchronicznego (wejściowy)	RST	R	RST
Sygnał wyjściowy	led	D1, D2, D3	led
Sygnał sterujący (wejściowy)	-	S_L1, S_L2, S_L3	-

Wynik syntezy:
(np. schemat, zajętość zasobów)



Zrzut 2.: Schemat blokowy modułu „zegar.vhd”



Zrzut 3.: Wykres zajętości zasobów.

Opracowany kod:
(krótki opis głównych modułów)

Aby zaimplementować ten concept, stworzono nowy moduł – „zegar.vhd” – w nim zainstancjonowano moduł „licznik.vhd”, wykorzystując jego cechę – możliwość skalowania sygnału wyjściowego, sygnałem wejściowym S. Skoro wymóg był taki, że trzeba było

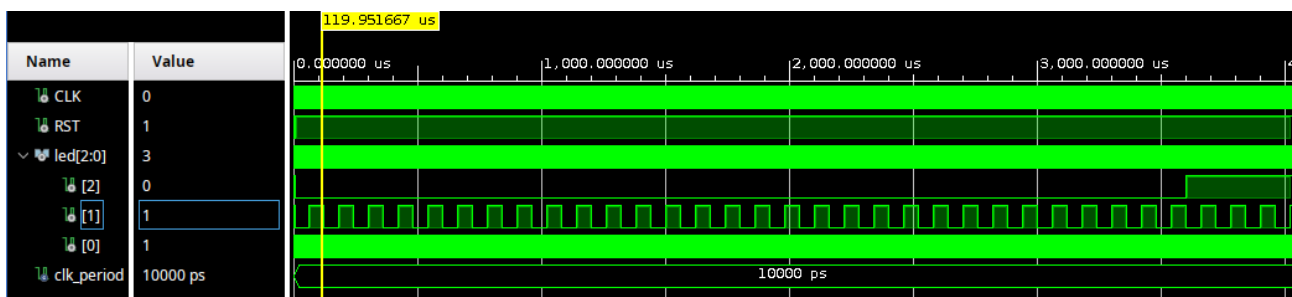
zaimplementować zegar z sekundami, minutami i godzinami – utworzono trzy instancje modułu „licznik.vhd”, każdej przypisując inne wartości sygnału skalującego S i inne wyjścia sygnału D na diody. Reset każdej z tych instancji został zmapowany na jeden z przycisków – czyli jeden przycisk resetuje cały układ.

Moduł „licznik.vhd” został też zmodyfikowany – sygnał S = „01” koduje sekundy, więc zakres jest równy tyle, co stała zakresu zliczania licznika T. Sekundy mają zakres równy $T \cdot 60$, a godziny $3600 \cdot T$. Ponadto zmieniono wartość T na 200.

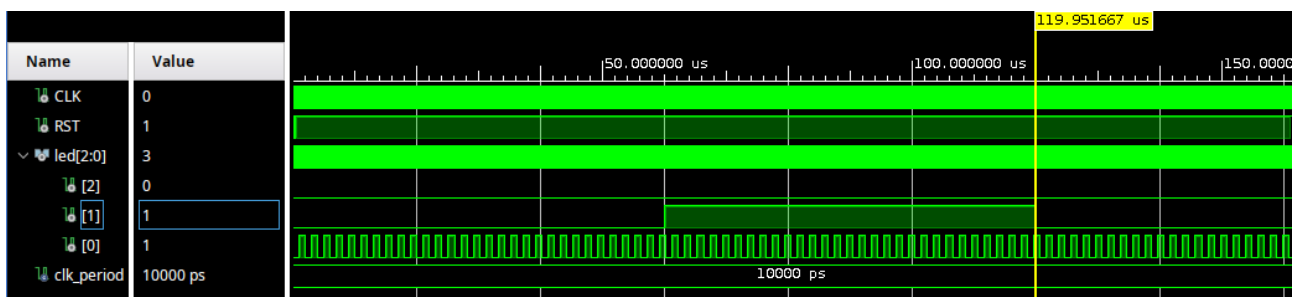
Aby przedstawić poprawność działania zaimplementowanego rozwiązania – napisano do niego też testbench. Aby przyspieszyć działanie zegara – skala czasowa wynosi 5ms czasu trwania symulacji.

Podsumowanie oraz wyniki:

(krótki opis działania układu, wynik testów, screenshot np. z programowania itp.)



Zrzut 4.: Przebiegi zegarowe porównujące sygnały wyjściowe diody led odpowiedzialnej za godziny i tej odpowiedzialnej za minuty



Zrzut 5.: Przebiegi zegarowe porównujące sygnały wyjściowe diody led odpowiedzialnej za minuty i tej odpowiedzialnej za sekundy

Proporcje między sygnałami są zachowane – a to było głównym celem. Skalę czasową można zmieniać zgodnie z życzeniem w module „licznik.vhd” , zmieniając tylko stałą T. Aby testbench zadziałał – należało zakomentować linię przypisującą reset do przycisku w głównym module „zegar.vhd”.