

# Akademia ETI

## Instrukcja do laboratorium





Jan Olencki, Jakub Gierowski,  
Mikołaj Barcikowski, Maciej Brzeski

27 lutego 2019

**Uwaga! Wszelkie nazwy utworzonych plików lub folderów nie mogą zawierać polskich znaków!**

**Przed utworzeniem nowego projektu należy zamknąć poprzedni (File > Close project)!**

## 1 Przerzutnik typu JK

$CLK$	$J$	$K$	$Q_{t+1}$	$\overline{Q_{t+1}}$
	0	0	$Q_t$	$\overline{Q_t}$
	0	1	0	1
	1	0	1	0
	1	1	$\overline{Q_t}$	$Q_t$

Tablica 1: Tabela wzbudzeń przerzutnika typu JK.

Tablica prawdy pokazuje, jaka wartość będzie na wyjściu przerzutnika(Q) w zależności od jego wejść J i K oraz sygnału zegarowego CLK. Wartość 1 oznacza stan wysoki, a wartość 0 stan niski. Przerzutnik pracuje w takt zegara, kiedy zmienia się zbocze sygnału zegarowego, czyli kiedy wartość zmienia się z 0 na 1.

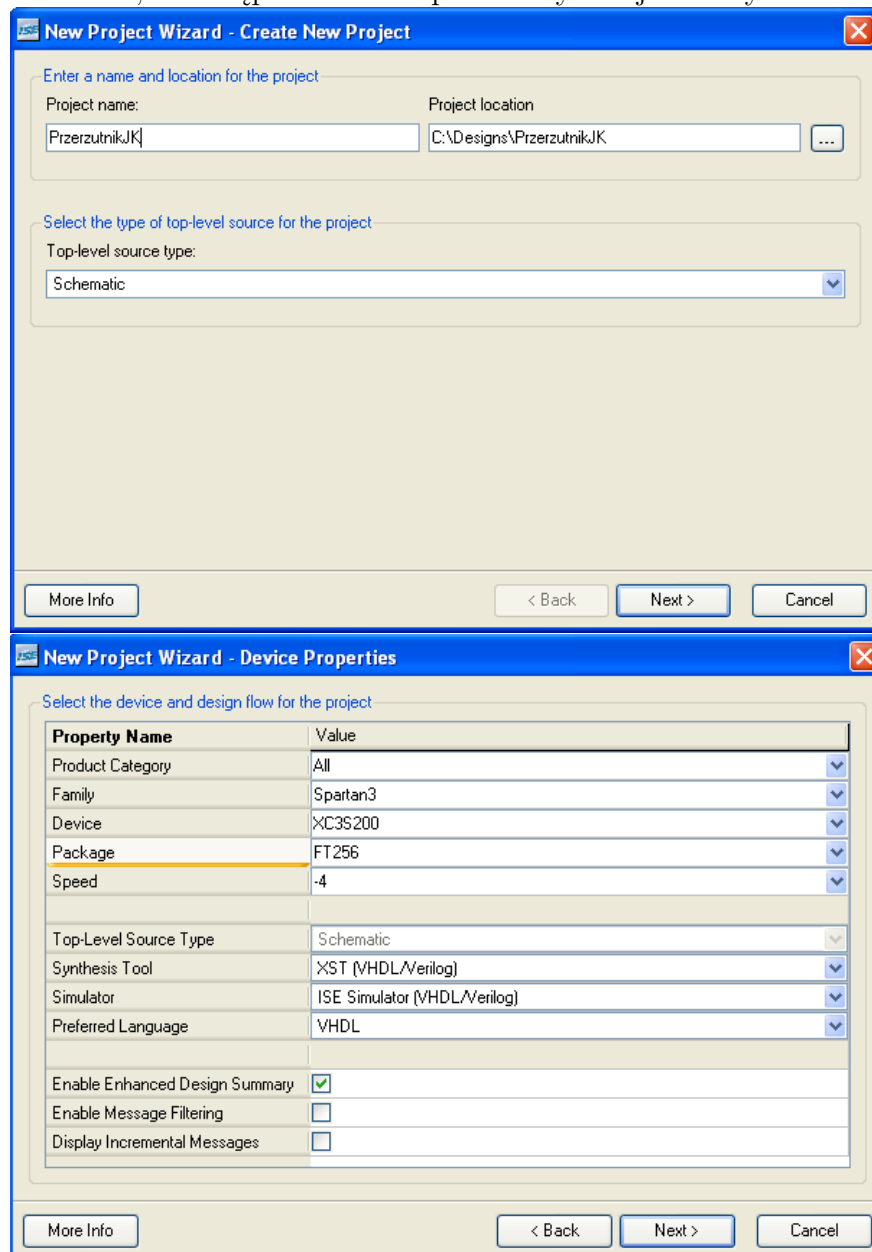
### 1.1 Do wykonania

1. Utworzyć nowy projekt w programie Xilinx ISE 10.1 i dodać do niego odpowiednie pliki źródłowe.
2. Przeanalizować schemat układu. Jaką częstotliwością taktowany jest układ przerzutnika? Które przyciski odpowiadają sygnałom J i K? Jak zostanie wyświetlony sygnał Q?

3. Przeprowadzić symulację układu.
4. Wgrać program do systemu docelowego. Co się dzieje gdy oba przyciski J i K są wciśnięte?

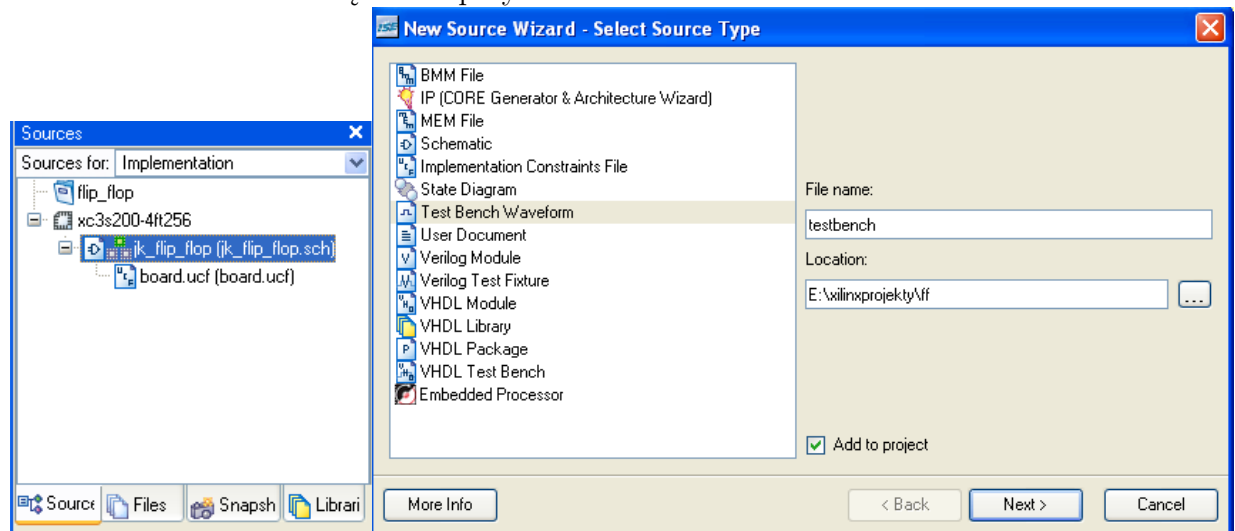
## 1.2 Krok po kroku

1. Ze strony pobrać folder *jk\_flip\_flop*.
2. Należy uruchomić program Xilinx ISE 10.1
3. Utworzyć nowy projekt poprzez File > New Project. Upewnić się, że został wybrany folder taki jak na rysunku. Podać nazwę projektu i wybrać z rozwijanego menu Schematic, nacisnąć przycisk Next, a następnie ustawić parametry tak jak na rysunku.

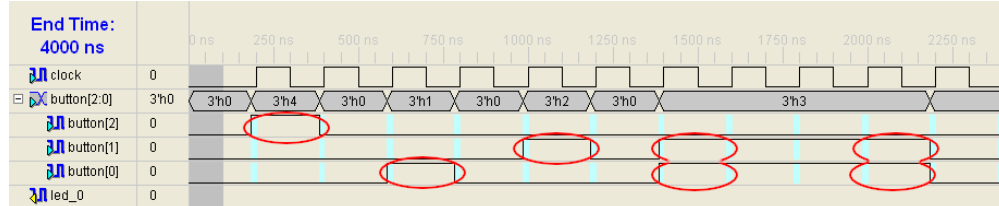


Nacisnąć przycisk Next. Następnie nacisnąć Add Source i wybrać pliki: *jk\_flip\_flop.sch* oraz *board.ucf* z pobranego wcześniej folderu. Następnie nacisnąć kolejne przyciski Next i Finish.

4. Następnie należy utworzyć plik symulacyjny testbench, dzięki czemu będzie można przetestować zaprojektowany układ przed zaprogramowaniem urządzenia docelowego. Aby to uczynić należy w okienku Sources kliknąć prawym przyciskiem myszy na pliku schematu *jk\_flip\_flop.sch* i wybrać opcję New Source. W wyświetlonym oknie wybrać opcję *Test Bench Waveform* i nadać nazwę pliku. Następnie nacisnąć dwukrotnie przycisk Next oraz na koniec Finish. Pojawi się kolejne okienko w którym należy wydłużyć czas trwania symulacji do 4  $\mu$ s (parametr Initial Length w prawym dolnym rogu). Ułatwi to zaobserwowanie działania układu. Zamknąć okno przyciskiem Finish.

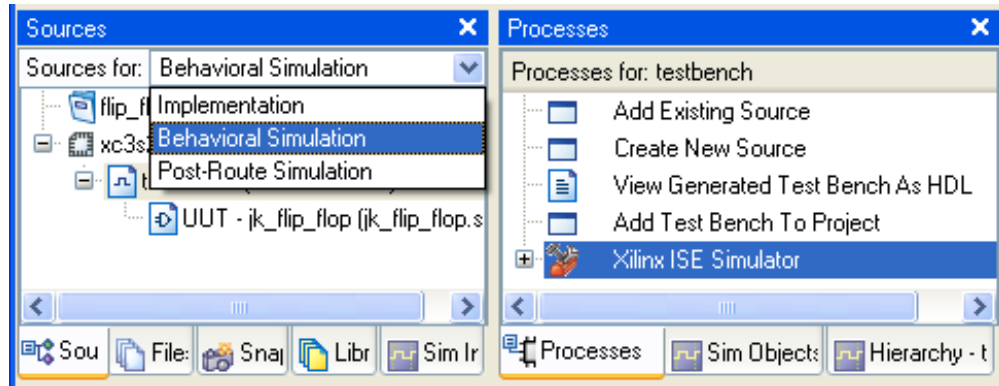


5. We wyświetlonym pliku można dowolnie ustawić sygnały J oraz K, odpowiednio *button[0]* i *button[1]*. Sygnał *button[2]* pełni rolę asynchronicznego resetu. Sygnał zmienia się poprzez naciśnięcie na wykresie czasowym w miejscu w którym chcemy aby nastąpiła zmiana.

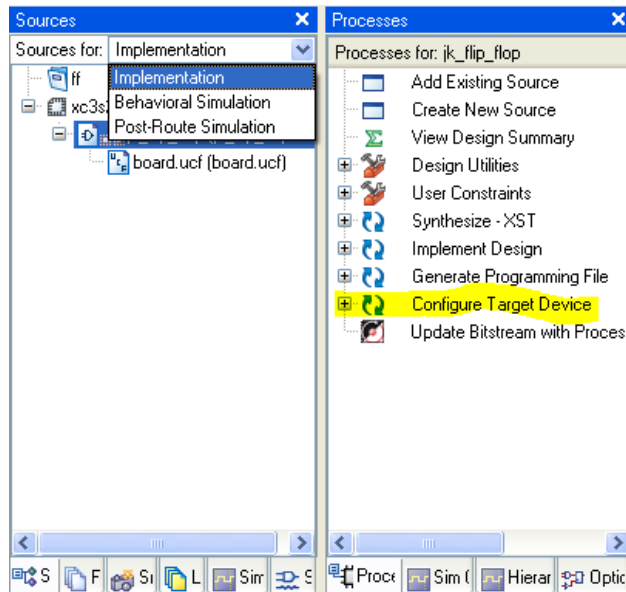


Na powyższym rysunku zaznaczono przykładowe miejsca kliknięć. Plik zapisać (Ctrl+S).

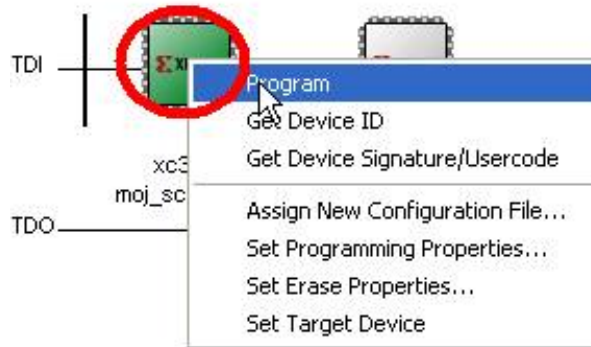
6. Aby uruchomić symulację należy w oknie Sources wybrać opcję Behavioral Simulation, zaznaczyć utworzony plik \*.tbw oraz dwukrotnie kliknąć na pozycji "Simulate Behavioral Model" w oknie Processes.



7. Jeżeli wyniki symulacji zgadzają się z oczekiwaniami można przejść do testowania układu bezpośrednio na płytce. Aby to zrobić w okienku Sources należy wybrać opcję Implementation i zaznaczyć plik *jk\_flip\_flop.sch*. W okienku Processes należy dwukrotnie nacisnąć Configure Target Device.



Zatwierdzić wyświetlony komunikat przyciskiem Ok, a następnie nacisnąć przycisk Finish. W wyświetlonym oknie należy wybrać plik z rozszerzeniem .bit i nacisnąć przycisk Open, a w kolejnym nacisnąć Bypass. W okienku które wyskoczy naciskamy przycisk Ok. Ostatnim krokiem jest naciśnięcie prawym przyciskiem w miejscu jak na rysunku oraz wybrać opcję Program.



8. Można zaobserwować świecenie diody poprzez naciskanie różnych kombinacji przycisków, zgodnie z tablicą prawdy przerzutnika JK.

## 2 Dzielnik częstotliwości

W tym zadaniu poruszony zostanie temat dzielnika częstotliwości i do zbadania jego działania zostanie użyta zwykła dioda. Częstotliwość generowana przez kryształ na płytce rozwojowej to 50 MHz, co oznacza, że dioda migałaby co około  $2 \cdot 10^{-8}$ s, co oczywiście jest niezauważalne dla człowieka. Poprzez zastosowanie układu jakim jest dzielnik częstotliwości, można wydłużyć okres przełączania do np. 1 s. Układ ten zlicza odpowiednią liczbę taktów zegara wejściowego i po przekroczeniu określonej wartości zmienia wartość sygnału na wyjściu, dzięki czemu uzyskujemy odpowiednio „spowolniony” sygnał prostokątny. W układzie przedstawionym na zajęciach oprócz wejścia i wyjścia występuje przycisk reset, który działa on asynchronicznie, czyli wartość na wyjściu układu zostanie wyzerowana w momencie naciśnięcia przycisku, niezależnie od aktualnego stanu wejściowego sygnału zegarowego.

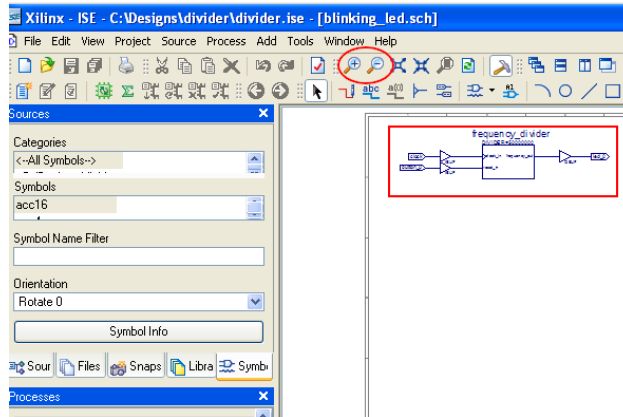
### 2.1 Do wykonania

1. Utworzyć nowy projekt wraz z odpowiednimi plikami źródłowymi.
2. Przeanalizować schemat układu. W jaki sposób można zmienić wartość dzielnika?
3. Wgrać program do systemu docelowego.

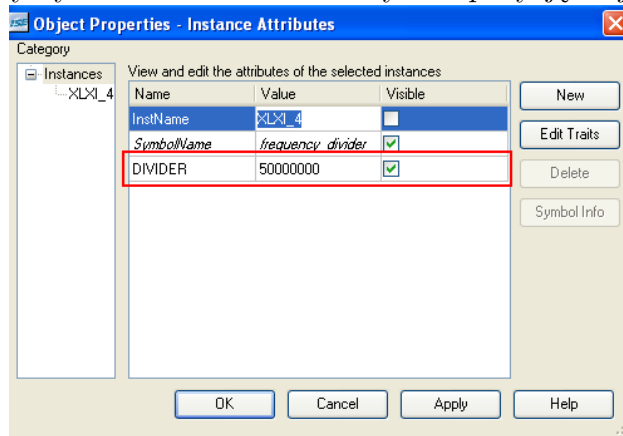
### 2.2 Krok po kroku

1. Zamknąć poprzedni projekt (File > Close project).
2. Utworzyć projekt i dodać do niego pliki z folderu *blinking\_led*, czyli *blinking\_led.sch*, *board.ucf*, *frequency\_divider.sym* oraz *frequency\_divider.vhd*, w podobny sposób, jak do poprzedniego zadania (podpunkt 1.2.3.).

3. W celu zmiany wartości dzielnika należy uruchomić w oknie *Sources* dwukrotnie kliknąć plik *blinking\_led.sch*.



4. Przybliżyć schemat tak aby widoczne były jego wszystkie elementy, następnie kliknąć prawym przyciskiem myszy na bloku dzielnika i wybrać pozycję *Object Properties*.

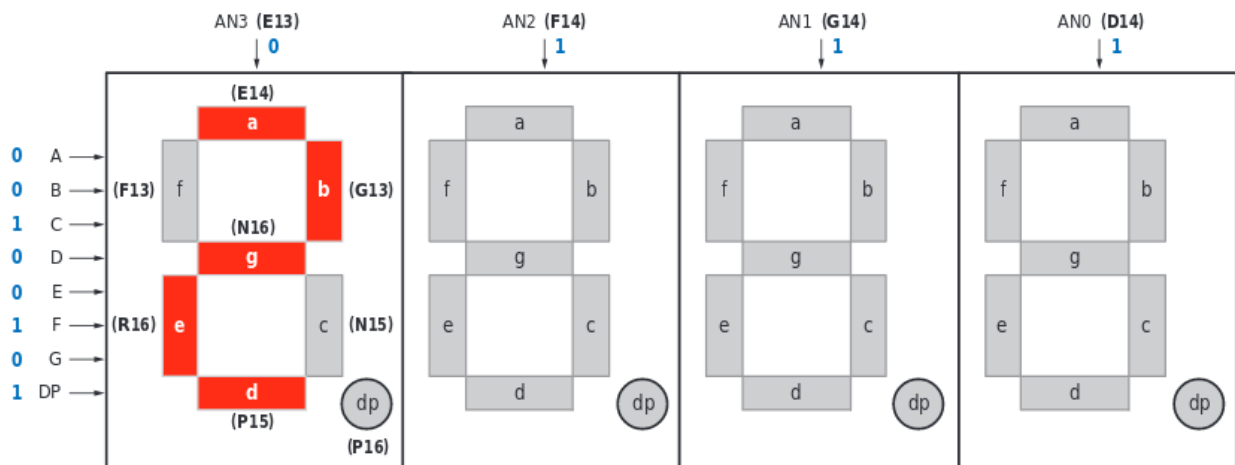


5. Zmienić wartość dzielnika (pole DIVIDER). Domyślna wartość  $5 \cdot 10^7$  odpowiada częstotliwości wyjściowej 1 Hz.

6. Aby zaprogramować układ w systemie docelowym należy postąpić podobnie jak w poprzednim ćwiczeniu (podpunkt 1.2.7.).

### 3 Sterownik wyświetlacza siedmiosegmentowego

Kolejnym układem którego działanie zostanie zaprezentowane jest wyświetlacz siedmiosegmentowy. Taki wyświetlacz, dzięki swojej prostocie, powszechnie używany jest do wyświetlania liczb. Pojedynczy wyświetlacz składa się z siedmiu segmentów, które zostają podświetlane zależnie od danych jakie chcemy wyświetlić.



UG130\_c3\_01\_042704

Na płytce rozwojowej dostępnej w laboratorium występuje poczwórny wyświetlacz, który posiada jedną ośmiobitową linię zapalającą segmenty (zobrazowaną po lewej) oraz cztery linie aktywujące wyświetlanie (widoczne na górze). W konsekwencji jeśli na każdym wyświetlaczu chcemy wyświetlić inny znak (aby przykładowo wyświetlić jakąś liczbę) musimy dokonać multipleksacji, co sprowadza się do sekwencyjnego zapalania każdego z wyświetlaczy. Najpierw ustawiamy dane pierwszego wyświetlacza i go aktywujemy, następnie wyświetlacz ten wygaszamy, zmieniamy dane i zapalamy drugi itd. Aby znak został wyświetlony tylko na jednym wyświetlaczu, na linii aktywującej konkretny wyświetlacz powinna wystąpić wartość 0, a na innych liniach 1 (wejście aktywne stanem niskim). Aby taki układ działał poprawnie, musi działać z odpowiednią częstotliwością. Częstotliwość ta musi być odpowiednio wysoka, aby ludzkie oko nie wychwyciło migotania obrazu, ale też nie za wysoka, aby kolejne wyświetlacze zdążyły się wygasić przed zmianą danych. Na potrzeby tego zadania została przyjęta 1 kHz, więc do tego celu w sterowniku został użyty dzielnik częstotliwości ze zadania poprzedniego.

### 3.1 Do wykonania

1. Utworzyć nowy projekt. Pobrać ze strony folder *display\_controler\_test*. Do projektu dodać następujące pliki: *OBUF4\_magistral.sch*, *OBUF4\_magistral.sym*, *board.ucf*, *display\_controler\_test.sch*, *display\_led7seg\_controler.sym*, *display\_led7seg\_controller.vhd*, *frequency\_divider.sym*, *frequency\_divider.vhd*.
2. Przeanalizować schemat wyświetlacza. W jaki sposób zapamiętywane są wyświetlane znaki?
3. Wgrać program do systemu docelowego. Jaka jest największa liczba możliwa do wyświetlenia?

## 4 Stoper

Ostatnim przewidzianym na zajęcia zadaniem jest zbadanie praktycznego wykorzystania wszystkich poprzednich układów. Stoper składa się on ze sterownika wyświetlania, czterech

liczników modulo 10 połączonych kaskadowo oraz dzielnika częstotliwości wykorzystanego w celu uzyskania stałej częstotliwości 100 Hz. Częstotliwość ustala rozdzielczość pomiaru czasu na poziomie 10 ms.

#### 4.1 Do wykonania

1. Utworzyć nowy projekt. Pobrać ze strony folder *stopwatch*. Do projektu dodać następujące pliki: *OBUF4\_magistral.sch*, *OBUF4\_magistral.sym*, *board.ucf*, *counter\_modulo\_10.sym*, *counter\_modulo\_10.vhd*, *display\_led7seg\_controler.sym*, *display\_led7seg\_controller.vhd*, *frequency\_divider.sym*, *frequency\_divider.vhd*, *stopwatch.sch*.
2. Przeanalizować schemat stopera. Jakie są funkcje poszczególnych przycisków?
3. Wgrać program do systemu docelowego. Co się dzieje, gdy mierzony czas przekracza 99,99 s?