

**Wydział Fizyki, Astronomii i Informatyki Stosowanej**  
**Uniwersytetu Mikołaja Kopernika**  
**Pracownia układów programowalnych**

**Ćwiczenie 2**

**Projekt i implementacja liczników synchronicznych i asynchronicznych w strukturze układu programowalnego FPGA Spartan-6 (ATLYS, Schematic, Licznik synchroniczny i asynchroniczny, pamięć ROM)**

### Cel ćwiczenia

Celem ćwiczenia jest implementacja projektu logicznego w strukturze układu programowalnego Spartan-6 przy pomocy programu narzędziowego firmy Xilinx – ISE Design Suite [1]. Student powinien nabyć praktycznej wiedzy na temat funkcjonalnego oraz strukturalnego projektowania liczników synchronicznych i asynchronicznych, a także używanych narzędzi i metod stosowanych do wielopoziomowego opisu projektu.

### Zagadnienia do przygotowania

Budowa i zasada działania przerzutników typu T, D, JK, RS.

Liczniki synchroniczne i asynchroniczne – budowa, zasada działania.

Funkcjonalny i strukturalny sposób opisu projektu.

Mapowanie komponentów oraz realizacja komponentu sparаметryzowanego (instrukcja **generic**).

### Przebieg ćwiczenia

1. Uruchomić środowisko programistyczne ISE Design Suite. Nadać nazwę projektu i ustawić odpowiedni **typ projektu**. W tym celu jako typ projektu (Top-level source type) należy wybrać schematic. Następnie wybrać odpowiedni typ układu (zwrócić szczególną uwagę na prędkość układu) oraz dodać do projektu nowe źródło „VHDL Module”[2]. W kolejnym kroku w języku VHDL zaprojektować przerzutnik typu JK, oraz przeprowadzić symulację funkcjonalną korzystając z narzędzia ISim [3][4]. Przeprowadzić syntezę i obejrzeć w oknie procesów „Synthesize – XST” schemat RTL projektu (*View RTL Schematic*) oraz schemat implementacji z uwzględnieniem użytej technologii (*View Technology Schematic*). Na podstawie dołączonego do ćwiczenia pliku AtlysGeneral.ucf, utworzyć i nadać odpowiednie numery wyprowadzeń w pliku \*.ucf [5]. Do wejścia zegarowego doprowadzić sygnał z przycisku **BTND**, który należy odpowiednio odfiltrować (projekt modułu eliminujący drgania mechaniczne przycisku). Do wejść J i K przerzutnika doprowadzić sygnały z przełączników **SW1** i **SW0**. Stan wyjść przerzutnika wyprowadzić na diody **LD0** i **LD1**. Przeprowadzić implementację i przesłać mapę bitową (plik \*.bit) do zestawu ATLYS. Zaobserwować stan diod LED.
2. Dokonać importu zaprojektowanego wcześniej komponentu przerzutnika JK do biblioteki schematów. Korzystając z edytora schematów dokonać strukturalnego opisu licznika binarnego synchronicznego i asynchronicznego [3]. Szczegóły odnośnie postaci projektów liczników zostaną ustalone indywidualnie z opiekunem ćwiczenia, przy czym wymagana jest realizacja synchronicznego licznika rewersyjnego 2-bitowego. Sprawdzić poprawność zrealizowanego projektu przeprowadzając symulację funkcjonalną i czasową. W dowolny sposób przetestować projekty odpowiednio programując zestaw ATLYS [2].
3. Utworzyć nowy projekt wybierając jako typ projektu VHDL. Posiłkując się „*Language Template*” dodać do projektu element biblioteczny przerzutnika typu D. Dokonać

odpowiedniego mapowania w/w modułu stosując w tym celu język VHDL i instrukcję For – Generate używaną przez narzędzia syntezy do wielokrotnego użycia wskazanego fragmentu kodu. Składnia w/w polecenia dostępna jest w „*VHDL→Synthesis Constructs→Generate*”.

4. Zmodyfikować projekt wprowadzając zmienną parametryzowaną, której wartość (nadana poprzez użycie instrukcji generic) stanowi liczbę bitów licznika.
5. Zrealizować w/w liczniki stosując funkcjonalną formę opisu projektu.
6. (*punkt obowiązkowy dla 60-cio godzinnych grup ćwiczeniowych*) Korzystając z wcześniej nabytej wiedzy samodzielnie zrealizować licznik Johnsona. Zwiększyć funkcjonalność licznika zabezpieczając go przed możliwością wprowadzenia i pozostania licznika w niepożądanym automacie stanu współistniejącym w tym samym module sprzętowym licznika. Sprawdzić poprawność zrealizowanego projektu dokonując symulacji funkcjonalnej i czasowej. Zaprogramować układ i dokonać weryfikacji działania.
7. Zrealizować licznik Johnsona w oparciu o pamięć ROM.

## Literatura

- [1] Spartan-6 Family Overview: **ds160.pdf**.
  - [2] Atlys™ Spartan-6 FPGA Development Board: **Atlys\_rm\_v2.pdf, Atlys\_c2\_sch.pdf**.
  - [3] Język VHDL Projektowanie programowalnych układów logicznych, Skahill Kevin.
  - [4] Kalisz J. - Podstawy elektroniki cyfrowej, WKiŁ, Warszawa 2002.
  - [5] Plik: **AtlysGeneral.ucf**.
- UWAGA**, pozycje literaturowe dostępne są w postaci elektronicznej.