

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

**Ćwiczenie 12**

**Opis i implementacja układu cyfrowego z wykorzystaniem Matlab**  
**(System-Generator, Matlab, Virtex4, ML403)**

### Cel ćwiczenia

Celem ćwiczenia jest zapoznanie studenta z narzędziem System-Generator oraz pokazanie możliwości wykorzystania Matlab i Simulinka w opisie układów cyfrowych implementowanych w struktury FPGA. W trakcie tego ćwiczenia prosty projekt układu cyfrowego zostanie opisany w środowisku Matlab a następnie zaimplementowany do układu programowalnego [1]. Student ma możliwość sprawdzić działanie projektu wykorzystując symulację oraz obserwując działający układ.

### Zagadnienia do przygotowania

Zakłada się, że przed przystąpieniem do ćwiczenia student posiada praktyczną umiejętność obsługi Matlab oraz Simulinka. Wymagana jest znajomość reprezentacji liczb stałoprzecinkowych i całkowitych zarówno ze znakiem jak i bez znaku.

### Przebieg ćwiczenia

1. Wykorzystując komponenty In oraz Out zaprojektować przy użyciu Simulinka układ przenoszący stan przycisku SW3 na stan diody LED0 ([2] – schemat-1.png, [3]). Dokonać symulacji wykonanego układu – na wejście reprezentujące przycisk podłączyć generator impulsów, na wyjście reprezentujące diodę podłączyć oscyloskop. Zaimplementować powyższy układ do struktury FPGA oraz sprawdzić jego działanie. **Wybrać strukturę {Virtex4, xc4vfx12, -10, ff668}.**
2. Wykonać układ, który będzie posiadał dwa 32 bitowe, dwukierunkowe (up/down) liczniki binarne z wejściem odblokowującym (en), zliczające impulsy zegara wzorcowego 100 MHz ([2] – schemat-2.png, [3]). Oba liczniki (C0 i C1) powinny zostać, poprzez komponent relacyjny, podłączone do pary przycisków, z których pierwszy zwiększa stan licznika, a drugi ten stan zmniejsza. Wygodne jest następujące przypisanie SW3(C0++), SW4(C0--), SW5(C1++), SW7(C1--), SW6(RESET). Sześć najstarszych bitów każdego licznika powinno zostać, poprzez blok wycinający (Slice), podłączonych do trzech 6 wejściowych komponentów relacyjnych, które będą obrazować następujące stany C0==C1, C0>C1 oraz C1<C1. Do reprezentacji tych stanów wybrać diody LED0 (==), LED1 (>), LED2 (<).
3. *(punkt obowiązkowy dla 60-cio godzinnych grup ćwiczeniowych)* Dokonać symulacji w środowisku Matlab układu z punktu 2.
4. *(punkt obowiązkowy dla 60-cio godzinnych grup ćwiczeniowych)* Zaimplementować dodatkowy 8 bitowy komponent relacyjny, którego zadaniem jest porównywać 8 najstarszych bitów obu liczników i raportować zgodność na diodzie LED3 (==) ([2] – schemat-4.png, [3]).

## Literatura

[1] ML403 Evaluation Platform User Guide – nota producenta. Xilinx. 2004. (ML403ug.pdf).

[2] Schematy blokowe dla poszczególnych punktów ćwiczenia (schemat-1.png, schemat-2.png, schemat-4.png).

[3] Obrazy ważniejszych okien aplikacji (system\_generator.png, gateway\_in-1.png, gateway\_in-2.png, counter.png).

**UWAGA**, wszystkie pozycje literaturowe dostępne są w postaci elektronicznej (plik PDF oraz pliki PNG).