

# Logika cyfrowa

## Programistyczna lista zadań nr 4

Termin: 29 marca 2020

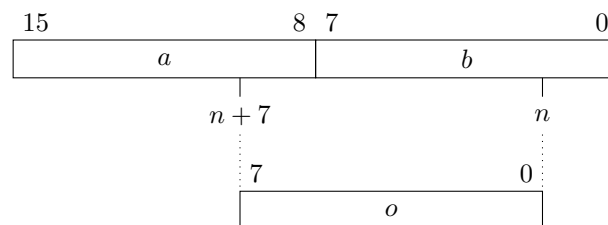
**Uwaga!** Poniższe zadania należy rozwiązać przy użyciu języka SystemVerilog, sprawdzić w DigitalJS oraz wysłać w systemie Web-CAT na SKOS. Należy pamiętać, aby nazwy portów nadesłanego modułu zgadzały się z podanymi w treści zadania. Wysłany plik powinien mieć nazwę `toplevel.sv`. **Nie przestrzeganie tych zasad będzie skutkowało przyznaniem 0 punktów.**

1. Zaimplementuj układ przesuwający wejście czterobitowe o jeden bit w lewo lub w prawo. Układ powinien mieć jedno wejście czterobitowe `i`, dwa wejścia jednobitowe `l` oraz `r`, oraz jedno wyjście czterobitowe `o`. Wyjście powinno zależeć od wejścia następująco:

- Jeśli `l` ma wartość 1, a `r` ma wartość 0, wynikiem powinno być wejście `i` przesunięte o 1 bit w lewo.
- Jeśli `r` ma wartość 1, a `l` ma wartość 0, wynikiem powinno być wejście `i` przesunięte o 1 bit w prawo.
- Jeśli zarówno `l` oraz `r` mają wartość 0, wynikiem powinno być wejście `i` (bez zmian).
- W przeciwnym wypadku wynik nie jest wyspecyfikowany.

Nie używaj wbudowanej arytmetyki SystemVeriloga ani wyrażenia warunkowego. Skonstruowany układ powinien mieć ścieżkę krytyczną o długości 3 bramek.

2. Zaimplementuj układ typu *funnel shifter*. W tym celu zaimplementuj moduł posiadający: dwa wejścia ośmiobitowe `a` oraz `b`, jedno wejście czterobitowe `n`, jedno wyjście ośmiobitowe `o`. Dozwolonymi wartościami wejścia `n` są 0 do 8. Wynikiem powinny być bity o numerach od  $n$  do  $n + 7$  wybrane z konkatencji wejść `a` oraz `b`. Ilustracja:



Wykorzystaj ten moduł do zbudowania ogólnego modułu przesuwająco-obracającego. Moduł ten powinien posiadać: jedno wejście ośmiobitowe `i`, jedno wejście czterobitowe `n`, trzy wejścia jednobitowe `ar`, `lr` oraz `rot`, wyjście ośmiobitowe `o`. Znaczenia wejść jednobitowych:

- stan wysoki `ar` oznacza przesunięcie arytmetyczne, stan niski – logiczne,
- stan wysoki `lr` oznacza przesunięcie w lewo, stan niski – w prawo,
- stan wysoki `rot` oznacza rotację, stan niski – przesunięcie (zgodnie z wartością `ar`).

Wykorzystaj wyrażenia warunkowe, aby na podstawie wartości wejść jednobitowych dobrać wartości wejść układu *funnel shifter* do żądanej operacji. Wyjście układu *funnel shifter* powinno być połączone bezpośrednio z wyjściem głównego układu. W zadaniu można używać wbudowaną arytmetykę SystemVeriloga.