

Projekt indywidualny 1 - wariant 55

Projekt jednostki wykonawczej **exe_unit**
operującej na danych w kodzie **U2**

Semestr zima 2021/22

Cel projektu

Celem projektu jest zrealizowanie jednostki wykonawczej **exe_unit** wykonującej operacje arytmetyczne, logiczne i inne na danych zapisanych w kodzie **U2**.

Realizowane operacje

Jednostka **exe_unit** ma realizować następujące operacje na danych wejściowych (operacje i punktacja za poprawną implementację):

- dodawanie (0.1 pkt)
- or - suma logiczna argumentów (0.1 pkt)
- nor - zanegowana suma logiczna argumentów (0.1 pkt)
- logiczne przesunięcie argumentów w lewo (0.1 pkt)
- arytmetyczne przesunięcie argumentów w lewo (0.1 pkt)
- konwersja danej wejściowej z kodu **U2** na kod **GRAY** (0.6 pkt)
- konwersję danej wejściowej z kodu **U1** na kod **U2** (0.6 pkt)
- sprawdzenie zgodności kodu CRC-4 (0.2 pkt)
- wyznaczenie kodu CRC-4 (0.2 pkt)
- zliczanie sumarycznej liczby jedynek w obu argumentach wejściowych (0.5 pkt)
- dekodery termometrowy (thermometer to binary encoder) (0.2 pkt)
- dekodery priorytetowy (0.2 pkt)

Wejścia i wyjścia

Rodzaj wykonywanej operacji jest określony przez **n-bitowe** wejście sterujące **i_oper**. Dane **m-bitowe**, na których wykonywane są operacje są pobierane z dwóch wejść **i_argA** i **i_argB**. Wynik jest podawany na **m-bitowe** wyjście **o_result**.

Jednostka **exe_unit** ma dodatkowe wyjścia jednobitowe będącymi znacznikami wykonanej operacji i stanu wyjścia **o_result** (znacznik i punktacja za poprawną implementację):

- **BF** - znacznik informujący, że w wyniku jest tylko jedna jedyńska (0.5 pkt)
- **NF** - znacznik uzupełnienia do nieparzystej liczby jedynek (0.5 pkt)
- **PF** - znacznik uzupełnienia do parzystej liczby jedynek (0.5 pkt)
- **BF** - znacznik informujący, że w wyniku jest tylko jedno zero (0.5 pkt)

Wymagania i punktacja

- Synteza logiczna całego modułu **exe_unit** wraz ze wszystkimi plikami musi przebiegać poprawnie - warunek konieczny
- Wszystkie pliki projektu i dokumenty muszą być umieszczone w repozytorium gitlab-stud.elka.pw.edu.pl w odpowiednich katalogach - warunek konieczny
- W repozytorium musi być widoczny systematyczny rozwój i praca nad projektem w postaci historii poszczególnych operacji commit na projekcie - warunek konieczny
- Sporządź specyfikację zaprojektowanej jednostki w postaci pliku pdf zawierającej niezbędne informacje umożliwiające wykorzystanie jednostki bez potrzeby szczególnej analizy opisu systemverilog (4 pkt):
 - listę wejść i wyjść, parametrów jednostki oraz zakresy ich wartości
 - listę realizowanych funkcji wraz z ich argumentami oraz kodem jaki należy podać na wejście **i_oper** aby wykonać zadaną funkcję
 - schemat blokowy struktury jednostki
 - specyfikacje sygnałów określających zaimplementowane flagi wraz z przykładowymi sytuacjami kiedy dana flaga przyjmie wartość 0 i 1
 - przykładowe użycie modułu pokazującego jak go instancjonować
 - listę wszystkich plików .sv wraz z informacją jakie moduły są w nich zdefiniowane
 - pozostałe informacje wedle uznania

- Poprawny i syntezywalny opis modułu **exe_unit**(1 pkt) - struktura projektu wraz z podziałem na poszczególne moduły, zależności między nimi i pliki źródłowe .sv pozostaje w indywidualnej gestii projektanta

Liczba punktów do zdobycia - 10 pkt w tym:

- 3 pkt - implementacja operacji
- 2 pkt - implementacja znaczników
- 4 pkt - specyfikacja spełniająca wymagania
- 1 pkt - poprawna implementacja modułu **exe_unit**

Punkty bonusowe

Sporządzenie sprawozdania wskazującego zgodność opisu z rezultatem syntezy logicznej. W każdym z analizowanych i wykazywanych przypadków w sprawozdaniu powinien znaleźć się rezultat syntezy dla której wykazywana jest zgodność opisu i wyniku syntezy wraz z opisem źródłowym. Wykazanie zgodności wyniku syntezy i opisu źródłowego może polegać np. porównanie tablicy prawdy dla obu opisów przy zadanych parametrach.

- Za każde poprawne wykazanie zgodności opisu i syntezy dla danej operacji lub znacznik liczba użytych punktów za tę operację lub znacznik jest podwajana