Systemy cyfrowe i komputerowe

Dokumentacja projektu “exe_unit_w1”

Karol Ambroziński
Nr. albumu: 318488

Spis treści

1 Wejścia, wyjścia, parametry i zakresy ich wartości

1.1 Parametry

- ***m*** - określa wielkość w bitach główne wejścia danych i wyjść,
- ***n*** - określa ilość operacji.

1.2 Wejścia

- ***i_oper*** - n-bitowe wejście określające wykonywaną operację,
- ***i_argA*** - m-bitowe wejście,
- ***i_argB*** - m-bitowe wejście,
- ***i_clk*** - 1 bitowe wejście zegarowe,
- ***i_rsn*** - 1 bitowe wejście resetu synchronicznego.

1.3 Wyjścia

- ***o_status*** - m-bitowe wyjście,
- ***o_result*** - 4 bitowe wyjście.

2 Realizowane funkcje i ich argumenty

Układ realizuje 4 operacje (4 podmoduły):

2.1 Podmoduł *mod1*:

Odejmowanie argumentów (A - B); jeśli operacja nie może zostać wykonana, jednostka zgłasza błąd, a wyjście jest niezdefiniowane. Kod modułu ***i_oper***: 00.

Wejścia

- ***i_argA*** - m-bitowe wejście,
- ***i_argB*** - m-bitowe wejście,

Wyjścia

- ***o_result*** - m-bitowe wyjście,
- ***o_status*** - 4 bitowe wyjście.

2.2 Podmoduł *mod2*:

Porównanie argumentów (A < B); jeśli warunek jest spełniony to wynikiem jest liczba 1, w przeciwnym wypadku wynikiem jest 0. Kod modułu ***i_oper***: 01.

Wejścia

- ***i_argA*** - m-bitowe wejście,
- ***i_argB*** - m-bitowe wejście,

Wyjścia

- ***o_result*** - m-bitowe wyjście,
- ***o_status*** - 4 bitowe wyjście.

2.3 Podmoduł *mod3*:

Ustawienie bitu w argumencie A na wartość 0; numer bitu jest określony w argumencie B; zgłoszenie błędu jeśli wartość B jest ujemna lub przekrasza liczbę bitów argumentu A. Kod modułu ***i_oper***: 10.

Wejścia

- ***i_argA*** - m-bitowe wejście,
- ***i_argB*** - m-bitowe wejście,

Wyjścia

- ***o_result*** - m-bitowe wyjście,
- ***o_status*** - 4 bitowe wyjście.

2.4 Podmoduł *mod4*:

Konwersja argumentu A z kodu ZNAK-MODUŁ na U2; jeśli konwersja nie może zostać wykonana - zgłaszany jest błąd a wynik jest nieokreślony. Kod modułu ***i_oper***: 11.

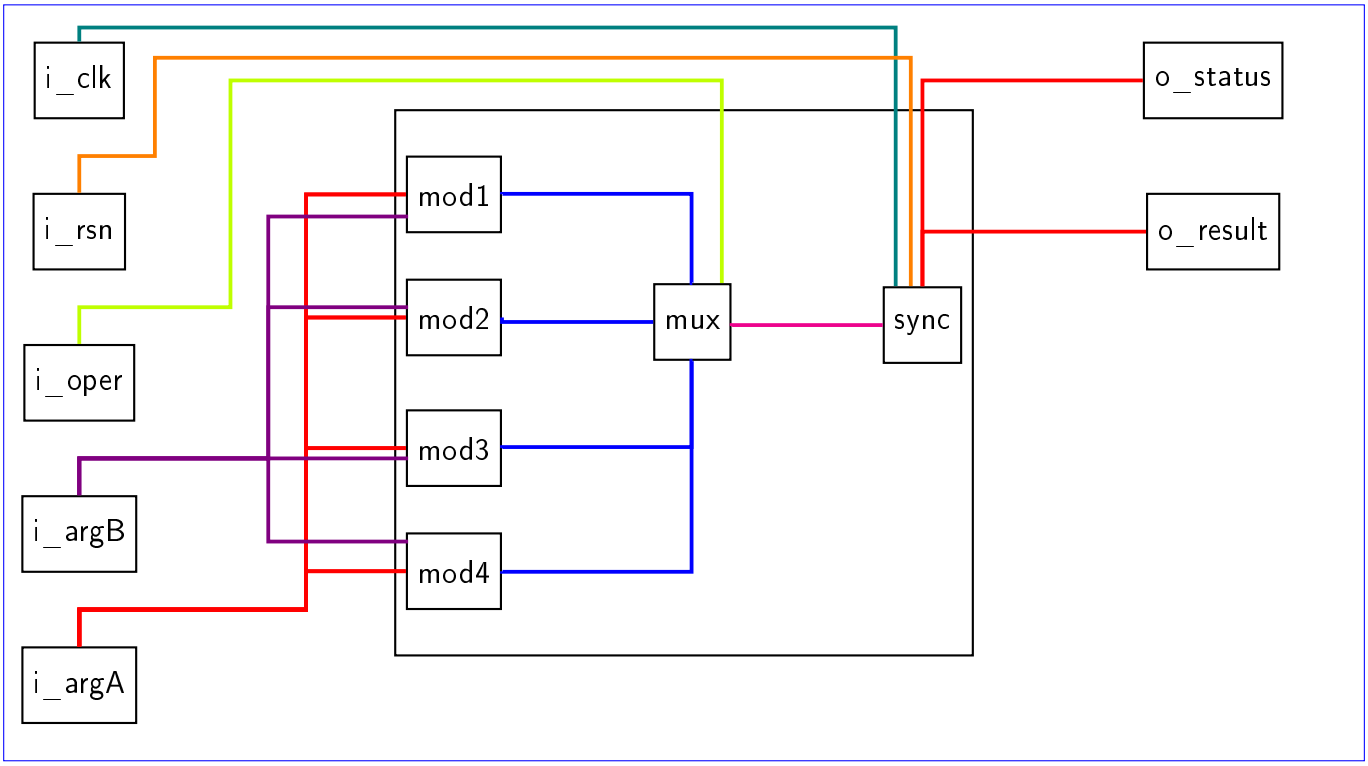
Wejścia

- ***i_argA*** - m-bitowe wejście,

Wyjścia

- ***o_result*** - m-bitowe wyjście,
- ***o_status*** - 4 bitowe wyjście.

3 Schemat blokowy struktury jednostki



Ilustracja nr. 1: Schemat blokowy ***exe_unit_w1***

4 Sygnały zaimplementowanych flag i ich wartości

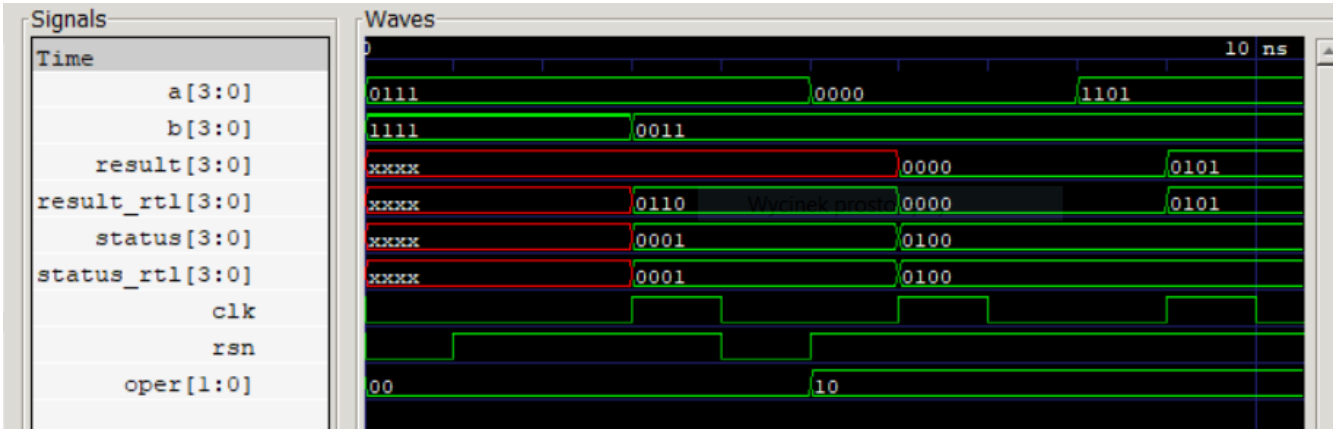
Zaimplementowane flagi o_status:

- ***ERROR*** - operacja nie została wykonana - ***o_status*** = 0b0001,
- ***NEG*** - wynik jest liczbą ujemną; ***o_status*** = 0b0010,
- ***EVEN*** - w wyniku jest parzysta liczba jedynek; ***o_status*** = 0b0100,
- ***ONES*** - wszystkie bity o_result ustawione; ***o_status*** = 0b1000.

Jeśli wynik jest nieokreslony (flaga ***ERROR***) to pozostałe bity nie są ustawiane; warunki pozostałych flag nie są sprawdzane.

5 Przykład użycia modułu

Na ilustracji nr. 2 przedstawiono wykonanie dwóch operacji: odejmowanie liczb A i B oraz zmianę bitu w argumencie A oznaczonego indeksem B (moduły: *mod1* i *mod3*). Przy pierwszej operacji na początku wynik jest niezdefiniowany i flaga błędu ustawiona na 1; przepełnienie wartości. W kolejnej operacji w argumencie B został zmieniony na indeksie B: B równe jest 3, więc bit nr. 3 (liczony od zera) w A został zmieniony na 0. Przy operacji ustawiania bitu poprzez B widać ustawienie flagi 0b0100 która oznacza że wynik posiada parzystą liczbę jedynek (co też jest widoczne na wyjściu result).



Ilustracja nr. 2: Widok wykonania testbenchu RTL i oryginalnych plikow w GTKWave

6 Lista plików

- ***exe_unit_w1.sv*** - plik zawierający główny moduł,
- ***otherModules.sv*** - plik zawierający wszystkie podmoduły głównego modułu (***mod1***, ***mod2***, ***mod3*** i ***mod4***),
- ***exe_unit_w1_rtl.sv*** - plik główny modułu po syntezie,
- ***synth.log*** - plik raportu Yosysa po syntezie.

7 Raport z syntezy logicznej

Podmoduł *mod1*	
Number of wires:	88
Number of wire bits:	115
Number of public wires:	12
Number of public wire bits:	39
Number of memories:	0
Number of memory bits:	0
Number of processes:	0
Number of cells:	84
$_AND_	33
$_NOT_	16
$_OR_	24
$_XOR_	11
Estimated number of transistors:	506

Podmoduł *mod2*	
Number of wires:	50
Number of wire bits:	105
Number of public wires:	11
Number of public wire bits:	66
Number of memories:	0
Number of memory bits:	0
Number of processes:	0
Number of cells:	41
$_AND_	14
$_NOT_	9
$_OR_	15
$_XOR_	3
Estimated number of transistors:	228

Podmoduł *mod3*	
Number of wires:	30
Number of wire bits:	79
Number of public wires:	8
Number of public wire bits:	57
Number of memories:	0
Number of memory bits:	0
Number of processes:	0
Number of cells:	30
$_AND_	13
$_NOT_	7
$_OR_	7
$_XOR_	3
Estimated number of transistors:	170

Podmoduł *mod4*	
Number of wires:	17
Number of wire bits:	32
Number of public wires:	6
Number of public wire bits:	21
Number of memories:	0
Number of memory bits:	0
Number of processes:	0
Number of cells:	16
$_AND_	7
$_NOT_	4
$_OR_	4
$_XOR_	1
Estimated number of transistors:	86

Moduł główny: *exe_unit_w1*	
Number of wires:	256
Number of wire bits:	445
Number of public wires:	54
Number of public wire bits:	243
Number of memories:	0
Number of memory bits:	0
Number of processes:	0
Number of cells:	241
$_AND_	103
$_NOT_	38
$_OR_	74
$_SDFF_PN0_	8
$_XOR_	18
Estimated number of transistors:	1354+