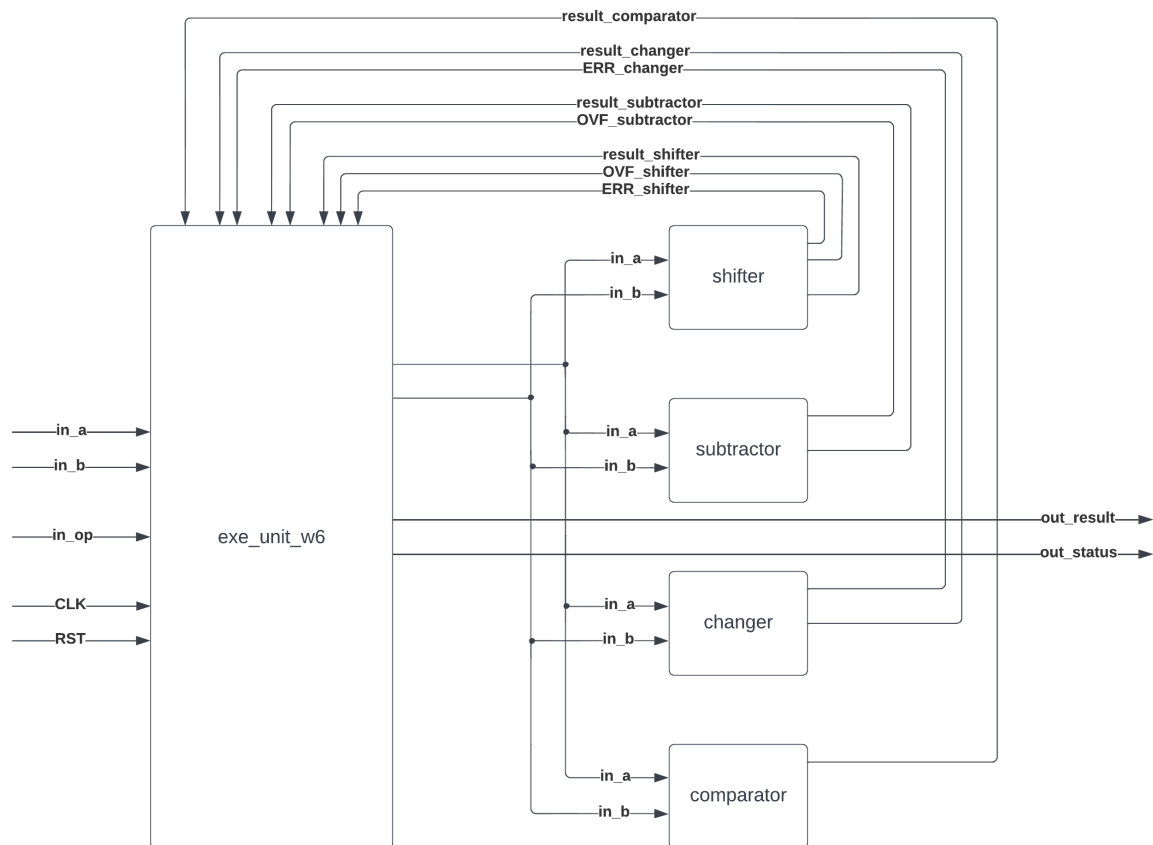


# Specyfikacja modułu exe\_unit\_w6

Patryk Korycki, nr albumu 318529

3 grudnia 2022



Schemat blokowy jednostki `exe_unit_w6`

## Opis jednostki

Zadanie polegało na implementacji modułu `exe_unit_w6`. Zadaniem układu jest wykonywanie kilku zdefiniowanych operacji matematycznych i logicznych.

## Wejścia

Działania są wykonywane na dwóch n-bitowych liczbach A i B (wejścia `in_a` i `in_b`) podanych na wejście. Wyboru operacji dokonuje się za pomocą 2-bitowego wejścia `in_op`. Dodatkowo układ jest wyposażony w wejście zegarowe wyzwalane zboczem narastającym

**CLK** i resetu synchronicznego **RST**. Stan niski na wejściu reset skutkuje przywróceniem stanu układu do stanu początkowego, czyli ustawienia operacji odejmowania dla obydwu wejść wynoszących zero. Pełna lista wejść układu jest przedstawiona w Tabeli 1. Poszczególne operacje wraz z wartościami wejścia **i\_op** zostały wymienione w Tabeli 2.

Wejście	Funcja	Ilość bitów wejściowych
in_a	Pierwszy składnik obliczeń	N-bitów
in_b	Drugi składnik obliczeń	N-bitów
in_op	Wybór operacji	2-bity
RST	Reset synchroniczny	1-bit
CLK	Taktowanie układu	1-bit

Tabela 1: Lista przedstawiająca wszystkie wejścia jednostki

Wartość wejścia in_op	Operacja	Flagi wyjściowe
0b00	Odejmowanie Liczby A od liczby B	OVF, EVEN, SINGLE
0b01	Porównanie, czy liczba $A \geq B$ . Jeśli tak, to wyjście jest dodatnie. Jeśli nie to wyjście jest zerem	EVEN, SINGLE
0b10	Przesunięcie liczby A o B bitów w lewo (z zachowaniem znaku). Gdy B ma wartość ujemną lub jest większe od liczby bitów liczby A, zwróć błąd.	OVF, ERROR, EVEN, SINGLE
0b11	Zmiana bitu liczby A na pozycji B. Gdy B ma wartość ujemną lub jest większe od liczby bitów liczby A, zwróć błąd.	ERROR, EVEN, SINGLE

Tabela 2: Opis poszczególnych operacji wraz z kodami wejściowymi i dostępnymi flagami

## Wyjścia

Na wyjściu modułu dostępne są dwie wartości: **out\_result** i **out\_status**. Wyjście **out\_result** zawiera wynik ostatnio wykonywanej operacji. Na wyjściu **out\_status** pojawiają się flagi informacyjne dotyczące ostatnio wykonanej operacji. Kolejność bitów i ich funkcje zostały opisane w tabeli 2. Każda z wykonywanych operacji ma możliwość zmiany wyłącznie wybranych flag. Pełna lista wyjść znajduje się w Tabeli 3. Flagi wyjściowe z każdej operacji są zawarte w Tabeli 4.

## Flagi

Rejestr wyjściowy **out\_status** składa się z 4 flag sygnalizujących stan wyjścia układu. Dodatkowo należy wspomnieć, że podczas załączenia flagi **ERROR** wyjście z układu jest nieokreślone i wynik będący wtedy na wyjściu w ogóle nie powinien być brany pod uwagę.

Wyjście	Funkcja	Ilość bitów wyjściowych
out_result	Wynik ostatnio wykonanej operacji	N-bitów
out_status	Rejestr z flagami informacyjnymi	4-bitów

Tabela 3: Lista przedstawiająca wszystkie wyjścia układu

Bit	3	2	1	0
	<b>SINGLE</b>	<b>OVF</b>	<b>EVEN</b>	<b>ERROR</b>

Tabela 4: Bity dostępne w wektorze wyjściowym **out\_status**

- **SINGLE** - Flaga informująca, że w wyniku jest tylko jedno zero
- **OVF** - Flaga informująca o przepełnieniu podczas operacji
- **EVEN** - Flaga informująca, że liczba zer w wyniku jest parzysta.
- **ERROR** - Flaga informująca o błędzie podczas wykonywania operacji.

## Instancjonowanie

W kodzie 1 pokazano ostateczne zainstancjonowanie jednostki w użyciu. W przypadku układu przed syntezą konieczne może być zdefiniowanie liczby N, czyli ilości bitów rejestrów wejściowych i wyjściowych. Wartość ta jest przechowywana w parametrze **BITS**. Nazwy podłączonych sygnałów wewnętrznych z przedrostkiem **s\_** są jedynie przykładowe i zostały użyte podczas testowania jednostki przed i po syntezie.<sup>5</sup>

---

**Kod 1** Przykładowe zainstancjonowanie jednostki **exe\_unit\_w6** w ostatecznym kodzie

---

```
exe_unit_w6 #(.BITS(N_BITS)) exe_unit_w6_model (.in_a(s_a), .in_b(s_b),
.i_op(s_op), .i_clk(s_clk), .i_rst(s_rst), .o_out(s_out_model),
.o_status(s_status_model));    // model przed syntezą
```

```
exe_unit_w6_rtl exe_unit_w6_synth (.in_a(s_a), .in_b(s_b),
.i_op(s_op), .i_clk(s_clk), .i_rst(s_rst), .o_out(s_out_synth),
.o_status(s_status_synth));    // model po syntezie
```

---

## Opis Podmodułów

Jednostka składa się z modułu sterującego i 4 modułów wykonawczych:

- **exe\_unit\_w6** - Moduł sterujący
- **subtractor** - Moduł odejmujący
- **comparator** - Moduł porównujący
- **shifter** - Moduł wykonujący operację przesunięcia
- **changer** - Moduł zmieniający bit na 1 na danej pozycji

Każdy z modułów wykonawczych odpowiada za daną operację. Z podmodułów wychodzą sygnały zawierające wynik operacji i ewentualne flagi informacyjne. W module sterującym wchodzi one do multipleksera, który podaje na wyjście wynik wybranych flag i operacji.

## Synteza logiczna

### Raport z syntezy

Pełny raport z syntezy znajduje się w pliku **Pliki\_projektu/synth.log**

## Symulacja i testy

## Struktura plików