

## Mikroelektronika w technice i medycynie, 2017 Podstawy projektowania układów cyfrowych VLSI

Historia dokumentu

27-03-2017 – wersja 1.1. Dodana kategoria „Open” - o dowolnej architekturze. Informacje o braku back-up'u.

24-02-2017 – wersja 1.0

# Projekt

Tematem projektu jest opracowanie 16-bitowego sumatora w wybranej architekturze.

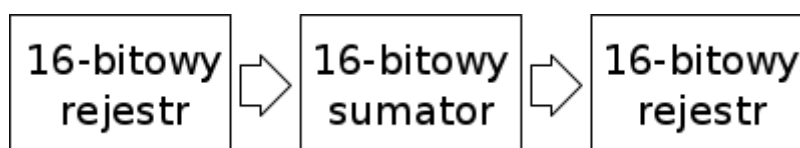
Pracujemy w zespołach 2-osobowych.

## Przebieg projektu

1. Przeczytaj rozdział o sumatorach z książki Weste, Harris, *Integrated Circuit Design 4<sup>th</sup> Ed.*
2. Wybierz jedną z architektur do realizacji:
  - Brent-Kung (maks. 6 zespołów),
  - Sklansky (maks. 6 zespołów),
  - Kogge-Stone (maks. 6 zespołów),
  - Kategoria „Open” - o dowolnej architekturze, brak limitu na liczbę zespołów. :)
3. Znajdź jedną lub więcej publikacji, które opisują wybraną architekturę i przeczytaj ją. Zwróć uwagę na zalety i wady konkretnego rozwiązania.
4. Opracuj schemat blokowy wybranego sumatora.
5. Opracuj schemat tranzystorowy wybranego sumatora.
6. Przeprowadź symulacje schematu.
7. Opracuj layout sumatora.
8. Przeprowadź symulacje post-layout.
9. Opracuj raport.
10. Załaduj raport i bibliotekę z projektem na UPEL.

## Szczegółowe wymagania

1. Projekt wykonujemy w technologii UMC 180 nm (komenda startowa **umc180oa**). Układ symulatora będzie projektowany pod kątem szybkości działania, poboru mocy i zajmowanej powierzchni. Im większa szybkość działania, mniejszy pobór mocy i mniejsza zajmowana powierzchnia – tym lepiej.
2. Wszystkie symulacje będą przeprowadzane w określonym środowisku. Sumator będzie otrzymywał dane z 16-bitowego rejestru, oraz jego wyjście będzie zatraskiwane w 16-bitowym rejestrze.



Schemat i symbol przerzutnika zostanie dostarczony w bibliotece.

3. Rejestry będą zbudowane z przerzutników D, zasilane oddzielnym napięciem i podłączone do idealnego sygnału zegara o czasach narastania i opadania 100 ps oraz wypełnieniu 50%.
4. Napięcie zasilania będzie wynosiło 1,8 V. Sumator będzie posiadał osobne napięcie zasilania w celu zapewnienia pomiaru poboru mocy.
5. Przeprowadzamy 2 rodzaje symulacji:
  - A) Symulacje najgorszej ścieżki, która determinuje maksymalną szybkość, z jaką sumator może działać. Symulacja wymaga takiego ustawienia sygnałów wejściowych, żeby sygnał przechodził ścieżką o największym czasie propagacji. Ścieżka ta zależy oczywiście od wybranej architektury. Symulacja musi uwzględniać obydwie zbocza sygnału. Poprawne działania sumatora zakłada prawidłowe zatrzaśnięcie wyniku w rejestrze wyjściowym. W wyniku symulacji otrzymujemy  **$f_{\max}$**  - **maksymalną częstotliwość zegara, z jaką układ może działać.**
  - B) Symulacje średniego poboru mocy. Przeprowadzamy symulacje dla serii 128 liczb A i B:

	<b>A</b>	<b>B</b>
0	$(2^{16}-1) - (0*511)$	$1+(0*511)$
1	$(2^{16}-1) - (1*511)$	$1+(1*511)$
2	$(2^{16}-1) - (2*511)$	$1+(2*511)$
...		
99	$(2^{16}-1) - (99*511)$	$1+(99*511)$

Symulacje przeprowadzamy dla częstotliwości zegara  $f_{clk} = 10$  MHz. Wynikiem symulacji jest  $P_{AVG}$  – **średnia moc pobierana przez sumator**.

Symulacje mają uwzględniać przypadki brzegowych parametrów technologicznych: **tt**, **ff**, **ss**, **spfn**, **fpsn**, czyli otrzymujemy pięć wartości  $f_{max}$ , spośród których jako wynik końcowy podajemy najniższą, oraz pięć wartości  $P_{AVG}$ , spośród których jako wynik końcowy podajemy najwyższą.

Wszystkie symulacje wykonujemy dla temperatury  $T = 27^{\circ}\text{C}$ .

6. Sumator nie może wykorzystywać żadnych innych sygnałów wejściowych poza dwoma 16-bitowymi liczbami A i B.
7. Wymagania odnośnie layout'u sumatora:
  - maksymalna proporcja układu to 2/1,
  - układ ma mieć 16 wejść z lewej strony na ME3 o minimalnej szerokości i 17 wyjść (wynik +  $C_{OUT}$ ) po prawej stronie na ME3 o minimalnej szerokości,
  - layout musi spełniać reguły DRC i LVS.
8. Przy ekstrakcji elementów pasożytniczych (i symulacjach post-layout) uwzględniamy wyłącznie pojemności (nie uwzględniamy rezystancji).

## Warunki oceniania

### Na ocenę 3.0 należy:

- przedstawić raport zawierający:
  - informację na temat przeczytanych publikacji (tytuł, autorów, wydawnictwo, rok wydania, krótki opis tego, co było opisane w publikacji),
  - schemat blokowy wybranego sumatora, na schemacie należy zaznaczyć najdłuższą ścieżkę propagacji sygnału,
  - opis cech charakterystycznych wybranego rozwiązania (czym się różni od innych),
  - wyniki symulacji schematu układu dla typowych parametrów technologicznych,
- zademonstrować działającą symulację schematu sumatora.

### Na ocenę 3.5 należy ponadto:

- uwzględnić w raporcie:

- wyniki symulacji dla parametrów brzegowych.

**Na ocenę 4.0 należy ponadto:**

- przedstawić opracowany layout sumatora, przechodzący DRC i LVS

**Na ocenę 4.5 należy ponadto:**

- przedstawić wyniki symulacji post-layout'owej,
- uwzględnić w raporcie:
  - porównanie wyników symulacji schematu i post-layout.

**Na ocenę 5.0 należy ponadto:**

- uwzględnić w raporcie:
  - opis metody doboru rozmiarów zastosowanych tranzystorów pod względem poboru mocy, zajmowanej powierzchni i szybkości działania,
  - dyskusję na temat wad i zalet wybranego rozwiązania na podstawie publikacji (tezy uzasadniamy odwołaniami do publikacji).

UWAGA: Końcowa ocena zależy od poprawnego wykonania wymienionych elementów.

**Dodatkowe punkty**

- Dodatkowe 0.5 pkt do oceny projektu otrzymujemy za wykonanie kompletnego projektu w terminie do 1 czerwca 2017 (decyduje data załadowania plików na UPEL).
- Dla ukończonego sumator definiujemy "Figure of merit" dla parametrów otrzymanych z symulacji post-layout:

$$FoM = \frac{f_{max}}{P_{AVG}}$$

Członkowie zespołu, który dla danej kategorii (Brent-Kung, Sklansky, Kogge-Stone, „Open”), uzyskają najwyższy **FoM**, otrzymują +1 punkt do oceny z egzaminu końcowego.

**Inne informacje**

Poproszę o załadowanie na UPEL raportu przez obydwie osoby w zespole (to ułatwia stwierdzenie, kto już oddał projekt).

Bibliotekę wystarczy jak załaduje jedna osoba. Proszę bibliotekę z projektem spakować komendą tar z kompresją gz lub bz2.

## **Back-up**

Uwaga: nie zapewniamy robienia kopii zapasowych projektu. Ponieważ biblioteki w środowisku Virtuoso są w postaci katalogów z plikami, dlatego najlepiej jest robić kopie zapasowe z całych katalogów przy pomocy komendy *tar*. Przykładowo, aby zrobić kopię zapasową biblioteki LIB, wchodzimy do katalogu poziom wyższego i wykonujemy komendę:

```
tar cvjf LIB_20170328.tbz LIB
```

Pliki \*.tbz przechowujemy w bezpiecznym miejscu. :)