

Instrukcja do ćwiczenia na platformie **CircuitLab** „Cyfrowe układy scalone - bramki logiczne”

1 Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z podstawowymi technikami cyfrowymi, przede wszystkim działaniem bramek logicznych i podstawowych układów cyfrowych. Badanie działania bramek logicznych wykonuje się za pomocą symulatora *Simulation Program With Integrated Circuit Emphasis* (SPICE) na platformie CircuitLab (www.circuitlab.com.)

2 Wymagania

Samodzielne zapoznanie się z wykładami z pracowni fizycznej i elektronicznej (wykład 6 i 7) na temat układów cyfrowych: układy cyfrowe 1 oraz układy cyfrowe 2, oraz skryptami z pracowni elektronicznej (skrypt 4 i 5): Cyfrowe układy scalone 1 i Cyfrowe układy scalone 2. Materiały są dostępne na stronie pracowni fizycznej i elektronicznej w zakładce *Wykłady*.

3 Dostęp do platformy CircuitLab

1. W przeglądarce internetowej otwórz stronę platformy: www.circuitlab.com.
2. Zaloguj się na konto użytkownika **pfe2020**
Password: pfe1234
3. W workbench'u użytkownika utwórz nowy projekt o nazwie <inicjały><nr_albumu> (np.: xx123456)

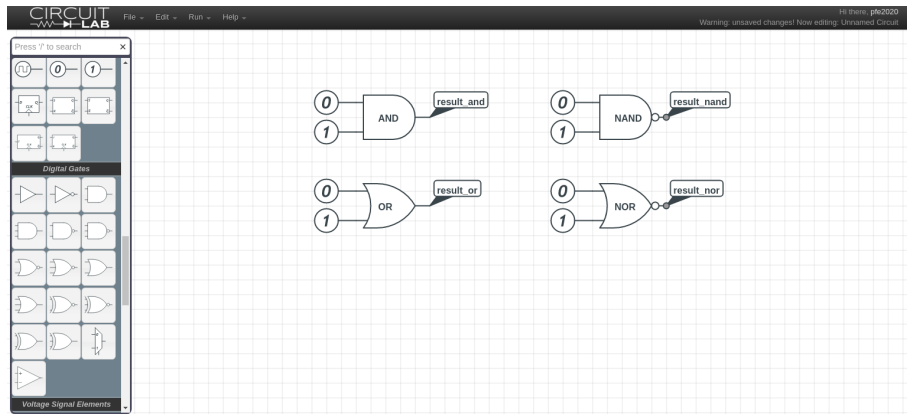
4 Wykonanie ćwiczenia

Wykonanie ćwiczenia polega na wykonaniu symulacji działania elektronicznych układów cyfrowych.

4.1 Zapoznanie się z bramkami logicznymi w środowisku CircuitLab

Platforma CircuitLab umożliwia zmontowanie (w panelu *Built*) prostych układów elektronicznych, w tym układów cyfrowych, i wykonanie symula-

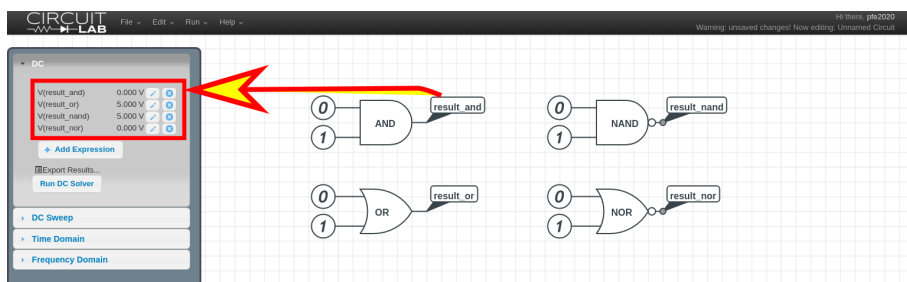
cji ich działania (w panelu *Simulate*). W ramach ćwiczenia, układy cyfrowe będziemy budować z bramek logicznych, przerzutników, generatorów stanów logicznych, cyfrowego generatora zegarowego oraz generatora funkcyjnego. Przykłady najprostszych układów, wykorzystujących bramki logiczne zostały zaprezentowane na rysunku 1. Na wejściach bramek logicznych ustawiono



Rysunek 1: Budowa podstawowych układów logicznych.

stany logiczne “0” i “1”, które odpowiadają napięciom TTL 0V i 5V.

Symulację działania zmontowanych układów możemy wykonać “w chwili włączenia układu”, wykorzystując narzędzia dostępne w zakładce *DC*, gdzie wynikiem symulacji będzie pomiar napięcia na wyjściu bramki logicznej, lub w domenie czasu, za pomocą narzędzi z zakładki *Time Domain*, gdzie wynikiem symulacji będzie przebieg czasowy sygnału na wyjściu bramki logicznej.. Symulacjami “w chwili włączenia układu” posługujemy się, gdy nie spodziewamy się cyklicznego działania układu, tj. przy badaniu działania bramek logicznych. Przykład takiej symulacji przedstawiono na rysunku 2 Symulacje



Rysunek 2: Przykład symulacji działania podstawowych układów logicznych “w chwili włączenia układu”.

w domenie czasu, wykorzystujemy przy badaniu układów z pamięcią – prze-

rzutników, oraz układów pokrewnych (liczniki, dekodery). Przykład ustawień wejściowych dla takiej symulacji załączono do zadania 1 z rozdziału 4.3.

Dodatkowo, przykładowe układy bramek logicznych i przerzutników są dostępne po zalogowaniu w projekcie: GATES_example

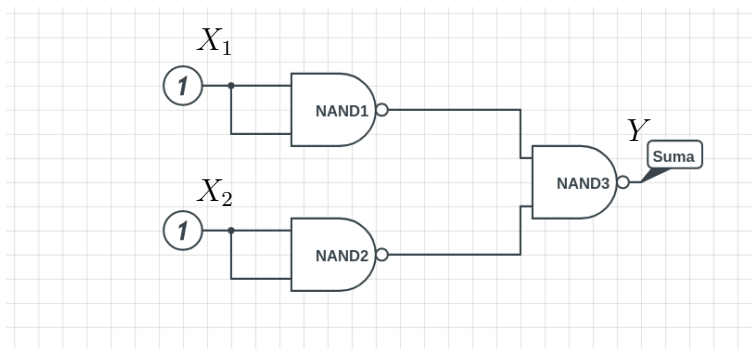
Proszę nie modyfikować tego projektu.

4.2 Zabawy z bramkami logicznymi

W panelu *Built* swojego projektu przygotuj schemat zbudowany z bramek NAND (**wyłącznie**), który posłuży do badania poniższych równań logicznych:

1. Suma logiczna: $Y = X_1 + X_2$

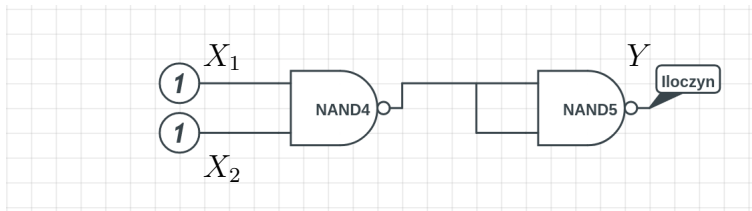
Wskazówka poniżej:



Rysunek 3: Schemat układu sumującego utworzony z bramek NAND.

2. Iloczyn logiczny: $Y = X_1 \bullet X_2$

Wskazówka poniżej:



Rysunek 4: Schemat układu mnożącego utworzony z bramek NAND.

3. Implikacja $[X_2 \rightarrow X_1]$: $Y = X_1 + \sim X_2$

4. Zakaz X_2 przez X_1 : $Y = \sim X_1 \bullet X_2$
5. Suma *Exclusive Or*: $Y = X_1 \bullet \sim X_2 + \sim X_1 \bullet X_2$

W panelu *Simulate*, zakładce *DC*, wykonaj symulacje działanie układu bramek logicznych dla różnych stanów na wejściu układu bramek logicznych. Wypełnij tabelę prawdy dla każdego z równań logicznych. Przykładowa tabela 1 zawiera tabelę prawdy dla bramki NAND.

X_1	X_2	Y
1	1	0
1	0	1
0	1	1
0	0	1

Tablica 1: Przykładowa tablica prawdy dla bramki NAND.

Schematy układów oraz ich tabele prawdy umieść w raporcie.

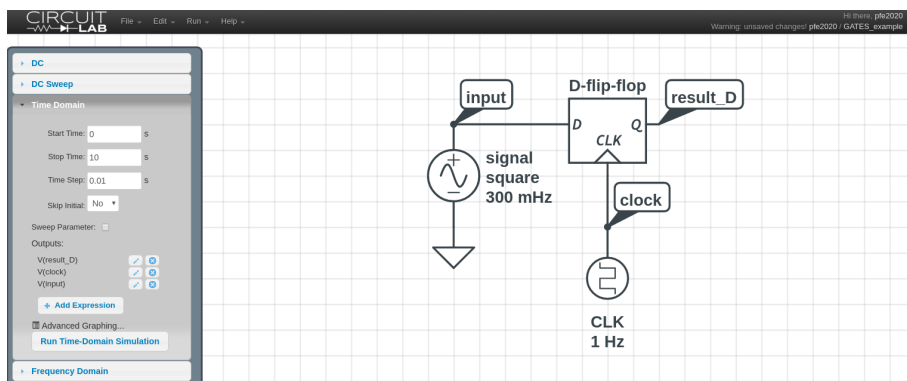
4.3 Przerzutniki

Wykonaj układy ilustrujące działanie przerzutnika D i przerzutnika T.

1. Na wejście zegarowe przerzutnika D podaj sygnał zegarowy o częstotliwości 1 Hz, na wejście D podaj sygnał prostokątny z generatora o częstotliwości 300 mHz. Pamiętaj o tym, że elektronika cyfrowa akceptuje sygnały w standardzie TTL (stan niski = 0 V, stan wysoki = 5 V). Przykład na rysunku 5. Zbadaj stan na wyjściu przerzutnika D w zależności od sygnału na jego wejściu i zbocza sygnału taktowania zegara.

Wskazówka: Aby uzyskać sygnał TTL, skorzystaj z opcji poziomu napięcia stałego (*DC offset*). Przy badaniu przebiegów czasowych na wyjściu z bramki przerzutnika, sygnały wejściowy i wyjściowy umieść na jednym wykresie, natomiast sygnał zegarowy na drugim (opcje dostępne w menu *Advanced Graphing*). Dodatkowo, aby przebiegi czasowe były bardziej przejrzyste, można wprowadzić sztuczne przesunięcie wybranych sygnałów definiując je, jako wyrażenia np.: $V(result_D)+10$.

2. Wykonaj podobną symulację dla przerzutnika T. Na wejście przerzutnika T podaj logiczny stan wysoki (1) i zaobserwuj sygnał wyjściowy z przerzutnika T w funkcji od cyklu zegara.

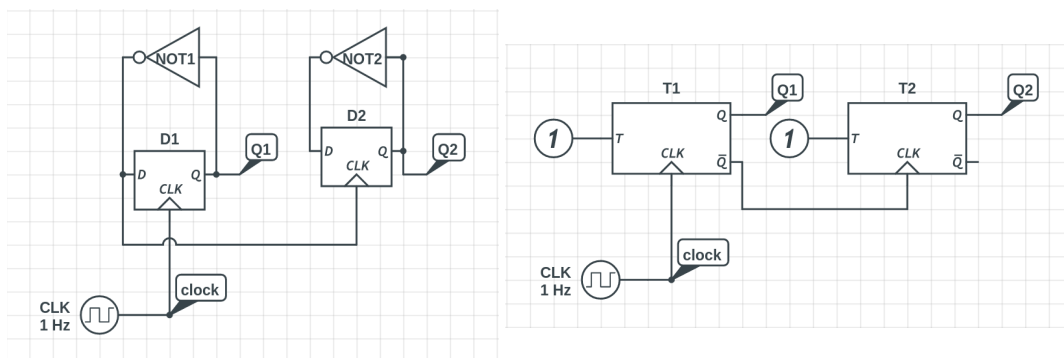


Rysunek 5: Schemat podłączenia przerzutnika D i przykład sprawdzenia zasady jego działania.

Schematy układów z przerzutnikami i symulacje obrazujące ich działanie umieść w raporcie. Na podstawie obserwacji, krótko opisz zasadę działania przerzutników D i T oraz różnicę między nimi.

4.4 Licznik

Bazując na przerzutnikach T lub przerzutnikach D (wraz z negacją wyjścia Q) wykonaj licznik 4-bitowy. Przykładowe schematy dla liczników 2-bitowych przedstawiono na rysunku 6.



Rysunek 6: Przykład licznika 2-bitowego zbudowanego z przerzutników D (po lewej) i przerzutników T (po prawej).

Rozpisz tabelę stanów licznika 4-bitowego (patrz Tab. 3) względem taktowania zegara.

Nr impulsu	Q_4	Q_3	Q_2	Q_1
0	0	0	0	0
1				

⋮

Tablica 2: Przykładowa tablica stanów dla licznika 4-bitowego, gdzie Q_4 jest najstarszym bitem.

Wskazówka: Aby wykonać ćwiczenie poprawnie, symulację należy rozpocząć w momencie, gdy wyjścia wszystkich przerzutników będą w stanie niskim (np. przy taktowaniu zegara z częstotliwością 1 Hz, rozpocznij symulację w momencie 2.5 s, a zakończ w momencie 18 s.) Taka konieczność jest spowodowana brakiem wyprowadzenia wejść R i S dla przerzutników typu D i T w środowisku CircuitLab.

Następnie, przebuduj układ na licznik na liczący “do tyłu” łącząc $Q - CK$ (zamiast $\sim Q - CK$), zaobserwować zmianę stanów i rozpisz ich tabelę dla tego licznika.

Schematy obu liczników oraz prezentacje ich działania umieść w raporcie.

Po wykonaniu tego ćwiczenia powróć do poprzedniego układu licznika.

4.5 Dekoder

Korzystając z licznika z poprzedniego ćwiczenia i 2-wejściowych bramek NAND zmontuj 4-bitowy dekodek liczby 10.

1. Zapisz liczbę 10 binarnie.
2. Rozpisz tabelę prawdy swojego dekodera, zakładając, że na wejście dekodera podajemy 4 wyjścia z licznika.

Q_4	Q_3	Q_2	Q_1	Dekoder

⋮

Tablica 3: Przykładowa tabela prawdy dekodera, na którego wejście podano wyjścia z 4-bitowego licznika (Q_4 jest najstarszym bitem).

3. Zmontuj dekodek liczby 10 z dowolnych bramek logicznych dostępnych na platformie CircuitLab i sprawdź jego działanie.

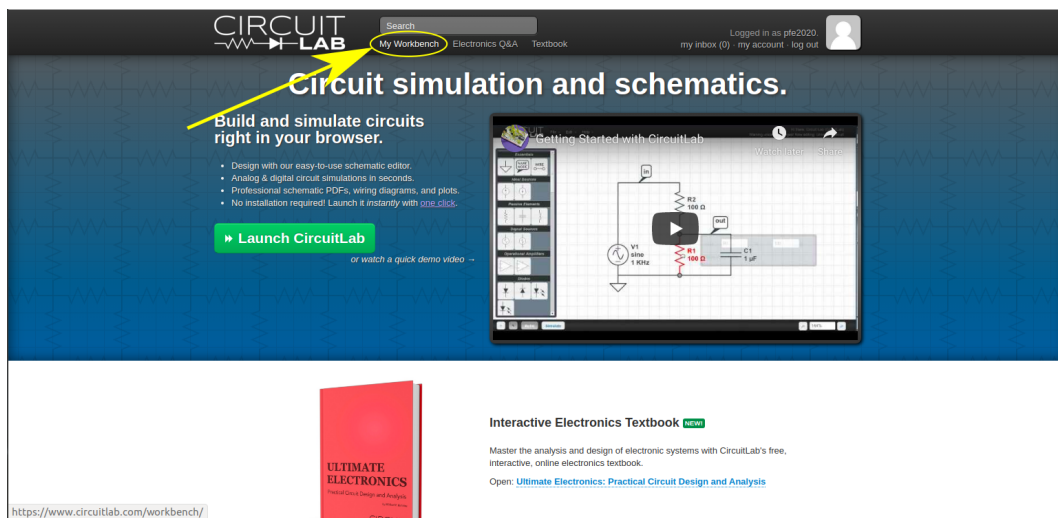
4. Zmontuj dekoder liczby 10 z wyłącznicze z 2-wejściowych bramek NAND i sprawdź jego działanie.

Tabele prawdy i schematy dekoderów umieść w raporcie.

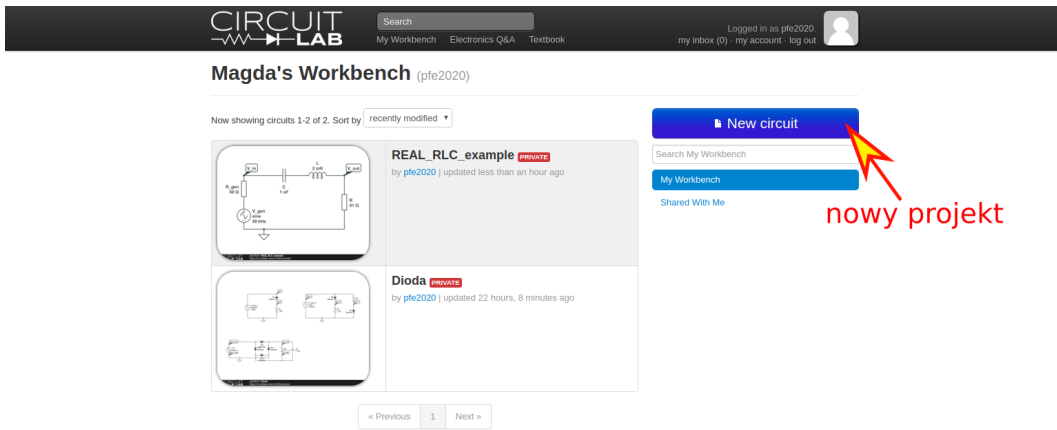
Miłej zabawy!

Dodatkowe informacje

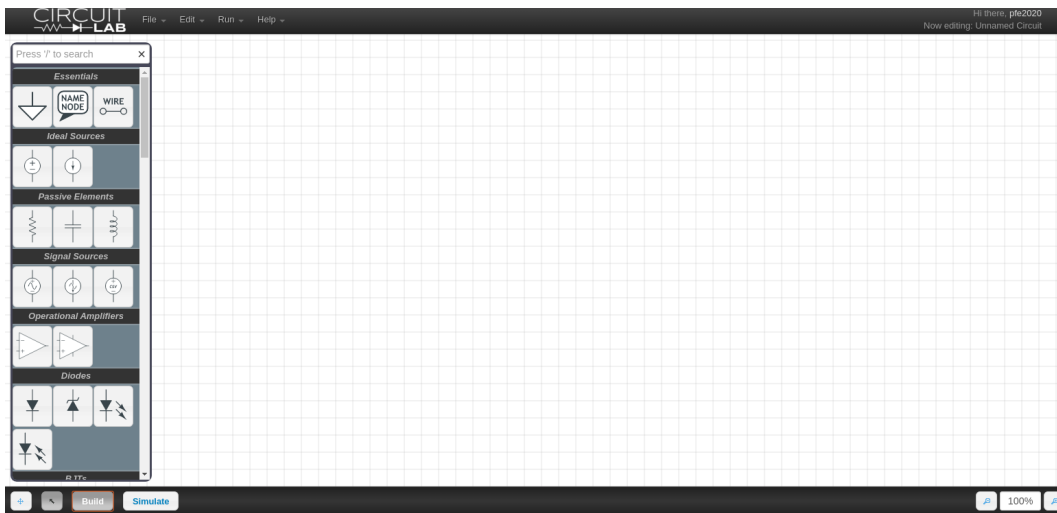
- Przykładowe układy bramek logicznych i przerzutników są dostępne po zalogowaniu w projekcie: GATES_example
Proszę nie modyfikować tego projektu.
- Przydatna może okazać się dokumentacja platformy: <https://www.circuitlab.com/docs/>
- Kilka zrzutów z ekranu pomocnych w nawigacji:



Rysunek 7: Panel startowy.



Rysunek 8: Workbench.



Rysunek 9: Panel *Build* - gotowy do montowania nowego układu elektronicznego.