

Wydział Fizyki UW

Pracownia fizyczna i elektroniczna (w tym komputerowa)
dla Inżynierii Nanostruktur (1100-1INZ27)
oraz Energetyki i Chemii Jądrowej (1100-1ENPRFIZELEK2)

Ćwiczenie D3

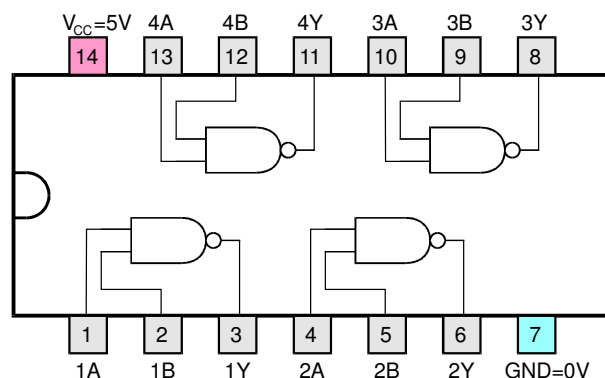
Projekt układu cyfrowego

Streszczenie

Końcowym zadaniem ćwiczeń z działu układy cyfrowe jest zaprojektowanie i zbudowanie dekodera liczb sprzężonego z licznikiem i wyświetlaczem. Projekt układu cyfrowego przygotowuje student i przedstawia asystentowi przed ćwiczeniami. Podstawowym elementem do budowy licznika będą przerzutniki D znajdujące się w układzie scalonym UCY7474. Dekoder zbudowany zostanie na bazie bramek NAND z układu scalonego UCY7400. Efekty pracy licznika czterobitowego wyświetlać będziemy w kodzie ósemkowym na dwóch wyświetlaczach cyfrowych.

Wstęp

Należy zrealizować układ złożony z dekodera liczb sprzężonego z licznikiem i 7-segmentowymi wyświetlaczami cyfrowymi. Licznik budujemy dokładnie tak, jak na poprzednich ćwiczeniach (ćw. D2 "Przerzutniki") na bazie układu scalonego UCY7474 przy użyciu makiety do obwodów cyfrowych i znajdujących się tam diod świecących oraz wyświetlaczy.



Rys. 1. Układ scalony UCY7400: 4 bramki NAND $Y = \overline{A \wedge B} = A \uparrow B$

Do realizacji dekodera wykorzystywać będziemy wyłącznie bramki NAND (układ scalony UCY7400). Bramki NAND są jednymi z najprostszych technicznie układów cyfrowych. Ponadto zgodnie z twierdzeniem Sheffera NAND (dysjunkcja = negacja koniunkcji, \uparrow) jest logiczną funkcją pełną, czyli przy pomocy NAND można zbudować dowolną funkcję logiczną. Korzystamy przy tym z praw de Morgana pozwalających zamieniać alternatywę na koniunkcję (i na odwrót) oraz z faktu, że dysjunkcja dwóch tych samych zdań jest negacją $P \uparrow P = \sim P$.

Zadanie

Każdy student wybiera na zajęciach poprzedzających (ćwiczenie D2), w porozumieniu z asystentem jeden z poniższych dekodów liczb 4-bitowych :

- | | |
|--|---|
| A) podzielnych przez 2 lub przez 5 , | 5 |
| B) podzielnych przez 2 lub przez 7 , | 4 |
| C) podzielnych przez 2 lub przez 9 , | 4 |
| D) podzielnych przez 2 lub przez 11 , | 4 |
| E) podzielnych przez 2 lub przez 13 , | 4 |
| F) podzielnych przez 2 lub przez 15 , | 4 |
| G) podzielnych przez 4 lub przez 9 , | 4 |
| H) podzielnych przez 4 lub przez 10 , | 4 |
| I) podzielnych przez 4 lub przez 13 , | 4 |
| J) podzielnych przez 4 lub przez 14 , | 4 |
| K) mniejszych od 8 lub podzielnych przez 5 , | 5 |
| L) mniejszych od 8 lub podzielnych przez 6 , | 4 |
| M) mniejszych od 8 lub podzielnych przez 7 . | 4 |

Dekoder powinien na wyjściu dawać 1, gdy wykryje, że liczba spełnia zadany warunek. Ograniczamy się do bramek NAND. W ostatniej kolumnie podana jest minimalna liczba bramek NAND potrzebnych do realizacji danej funkcji.

Projekt należy przygotować przed zajęciami zgodnie z formatem zamieszczonym na ostatniej stronie instrukcji.

Realizacja projektu

Projekt rozpoczynamy od ogólnego opisu planowanego działania i narysowania schematu blokowego zawierającego licznik, dekodery i wyświetlacze. Następnie piszemy tabelkę prawdy dla dekodera. Potem zapisujemy funkcje logiczne wynikające z tabelki prawdy w postaci rachunku zdań. Następnie rozwijamy schemat blokowy na schemat logiczny realizowany na elementach NAND (lub innych podanych w założeniach zadania) z zastosowaniem zasad minimalizacji.

Układy w ćwiczeniach należy projektować tak, by do realizacji funkcji wykorzystać możliwie najmniejszą liczbę układów scalonych – czyli doprowadzić funkcję do postaci minimalnej. Można uzyskać to metodą przekształceń algebraicznych.

Warto pamiętać, że sumę (funkcję **lub**), dzięki prawom de Morgana, łatwo można zamienić na koniunkcję, a jeszcze łatwiej na funkcje NAND.

$$A \vee C = \sim(\sim A \wedge \sim C) = \sim A \uparrow \sim C$$

W szczególności na NANDy, \uparrow , bardzo łatwo jest zamienić konstrukcję typu **i-lub-i**.

$$A \wedge B \vee C \wedge D = \sim(\sim(A \wedge B) \wedge \sim(C \wedge D)) = (A \uparrow B) \uparrow (C \uparrow D)$$

Można też zminimalizować układy pamiętając, że negacja negacji zmiennej daje zmienną pierwotną.

Projekt przekazujemy asystentowi do oceny przez zajęciami. Asystent sprawdza, czy w projekcie nie ma błędów i zatwierdza projekt do wykonania.

Przed przystąpieniem do budowania układu należy narysować projekt, z naniesieniem pozycji układów scalonych na makiety i podaniem konkretnych numerów „nózek” bramek wykorzystanych w połączeniach.

Skomplikowany projekt uruchamiamy metoda podziału na bloki - uruchamiamy na przykład blok licznika, blok dekodera, potem łączymy bloki i uruchamiamy całość. Na koniec dobudowujemy dodatkowe funkcje pozwalające na wstępne ustawienie zadanych wartości (SET) lub wyzerowanie układu (RESET). Zaczynamy uruchamianie zbudowanego układu wykorzystując diody LED z makiety jako pomocnicze wyświetlacze stanów binarnych. W początkowej fazie uruchamiania wykorzystujemy generator pojedynczych impulsów z makiety. Następnie podłączamy generator zewnętrzny i obserwujemy przebiegi na oscyloskopie. Do badania stanów wybranych punktów w układzie, przydatna może okazać się „sonda” – długi kabelek podłączony do diody świecącej.

Przykład

Budujemy dekodery wykrywający liczbę 3 bitową podzielną przez 2 lub przez 5. Zakładamy, że liczba 0 jest podzielna przez dowolny czynnik. Dekoder na wyjściu daje 1, gdy wykryje, że liczba spełnia zadany warunek. Ograniczamy się do bramek NAND.

Tabela prawdy

n	C	B	A	Y
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

2) Opis logiczny

W zapisie binarnym podzielność przez 2 oznacza, że najmłodszy bit jest równy zero, czyli mamy warunek $\sim A$. Do liczb podzielnych przez 5 zaliczymy 0 i 5, ale 0 już spełnia warunek podzielności przez 2, więc nie musimy brać go pod uwagę. Pozostanie warunek $n=5$, który zapiszemy jako $C\bar{B}A$, bo $5_{(10)} = 101_{(2)}$ (patrz tabela prawdy). Zatem poszukiwany warunek logiczny zapiszemy:

$$Y = \sim A \vee C \wedge \sim B \wedge A$$

3) Przekształcenia

Korzystając z rozdzielności alternatywy, otrzymujemy:

$$Y = (\sim A \vee C \wedge \sim B) \wedge (\sim A \vee A)$$

$$\sim A \vee A = 1$$

$$(\sim A \vee C \wedge \sim B) \wedge 1 = (\sim A \vee C \wedge \sim B)$$

$$Y = \sim A \vee C \wedge \sim B$$

Korzystając z prawa de Morgana, otrzymujemy:

$$Y = \sim(\sim \sim A \wedge \sim(C \wedge \sim B))$$

$$Y = \sim(A \wedge \sim(C \wedge \sim B))$$

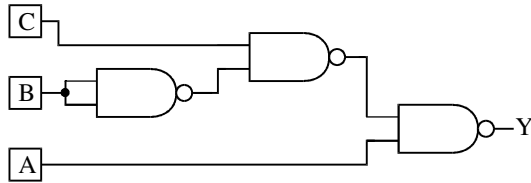
4) Zapisujemy równanie przy pomocy funkcji NAND, korzystając z prawa de Morgana, które mówi, że $(P \uparrow Q) = \sim(P \wedge Q)$:

$$Y = A \uparrow \sim(C \wedge \sim B),$$

$$Y = A \uparrow (C \uparrow \sim B),$$

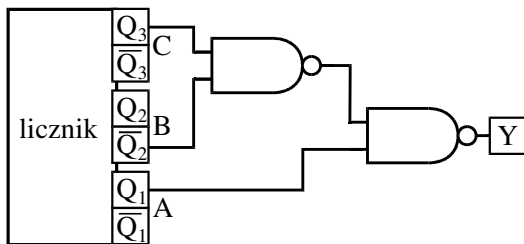
$$Y = A \uparrow (C \uparrow (B \uparrow B)).$$

5) Rysujemy schemat na bramkach:



Rys. 2) Schemat dekodera wykrywający liczbę podzielną przez 2 lub przez 5.

W naszym układzie będziemy mieli do dyspozycji wyjścia licznika dające wartości Q i $\sim Q$. Zatem negacja B nie będzie konieczna. zamiast tego skorzystamy z wyjścia $\sim Q_2$.



Rys. 3) Dekoder wykrywający liczbę podzielną przez 2 lub przez 5, podłączony do licznika.

imię i nazwisko

Projekt układu wykrywającego liczby podzielne przez n i m .

Wstęp

Schemat całego układu (licznik + dekodery + wyświetlacze i diody)

Tabela prawdy

n	D	C	B	A	Y
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
..	
14	1	1	1	0	
15	1	1	1	1	

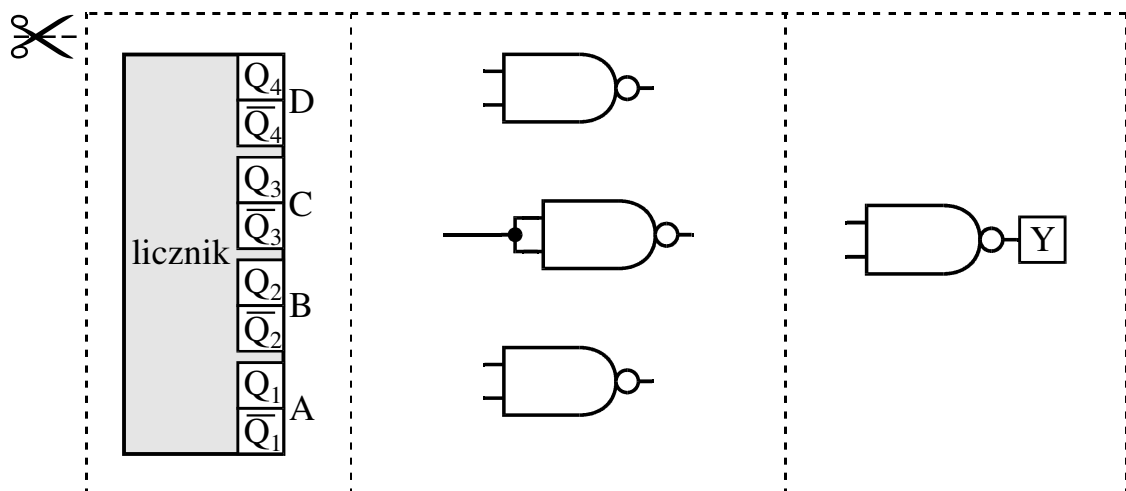
Opis logiczny

$$Y = M \vee P$$

Przekształcenia rachunku zdań dążące do optymalizacji funkcji zapisanej przy pomocy operatorów NAND, \uparrow .

$$Y = Q_n \uparrow Q_m \uparrow Q_k$$

Schemat połączeń bramek dekodera



Narysowane powyżej elementy można wyciąć i wkleić do programu graficznego, a następnie dorysować połączenia.

Ważna uwaga. Na rysunku niektóre połączenia mogą się przecinać. Jeżeli na skrzyżowaniu jest kropka, to znaczy, że przewody są połączone, jeżeli nie ma kropki, to znaczy, że przewody przechodzą jeden nad drugim, nie łącząc się.