

UKŁADY KOMBINACYJNE

1. Subtraktor 1-bitowy pełny

1.1 Opis

Subtraktor 1 bitowy jest układem kombinacyjnym wykonującym odejmowanie 2 cyfr binarnych A i B oraz cyfry pożyczki z poprzedniej pozycji (P_{i-1}). W wyniku działania układu otrzymujemy różnicę na danej pozycji (R) oraz pożyczkę z kolejnej pozycji (P_i).

1.2 Tabela prawdy

A	B	P_{i-1}	R	P_i
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

A – odjemna

B – odjemnik

P_{i-1} – pożyczka z poprzedniej pozycji

R – różnica

P_i – pożyczka

1.3 Minimalizacja funkcji

Ponieważ projektowany układ subtraktora ma dwa wyjścia wystarczy nam dwie siatki Karnaugh dla wyjść tj. R i P_i .

Siatka Karnaugh dla R

$P_{i-1} \backslash AB$	0 0	0 1	1 1	1 0
0	0	1	0	1
1	1	0	1	0

Siatka Karnaugh dla P_i

$P_{i-1} \backslash AB$	0 0	0 1	1 1	1 0
0	0	1	0	0
1	1	1	1	0

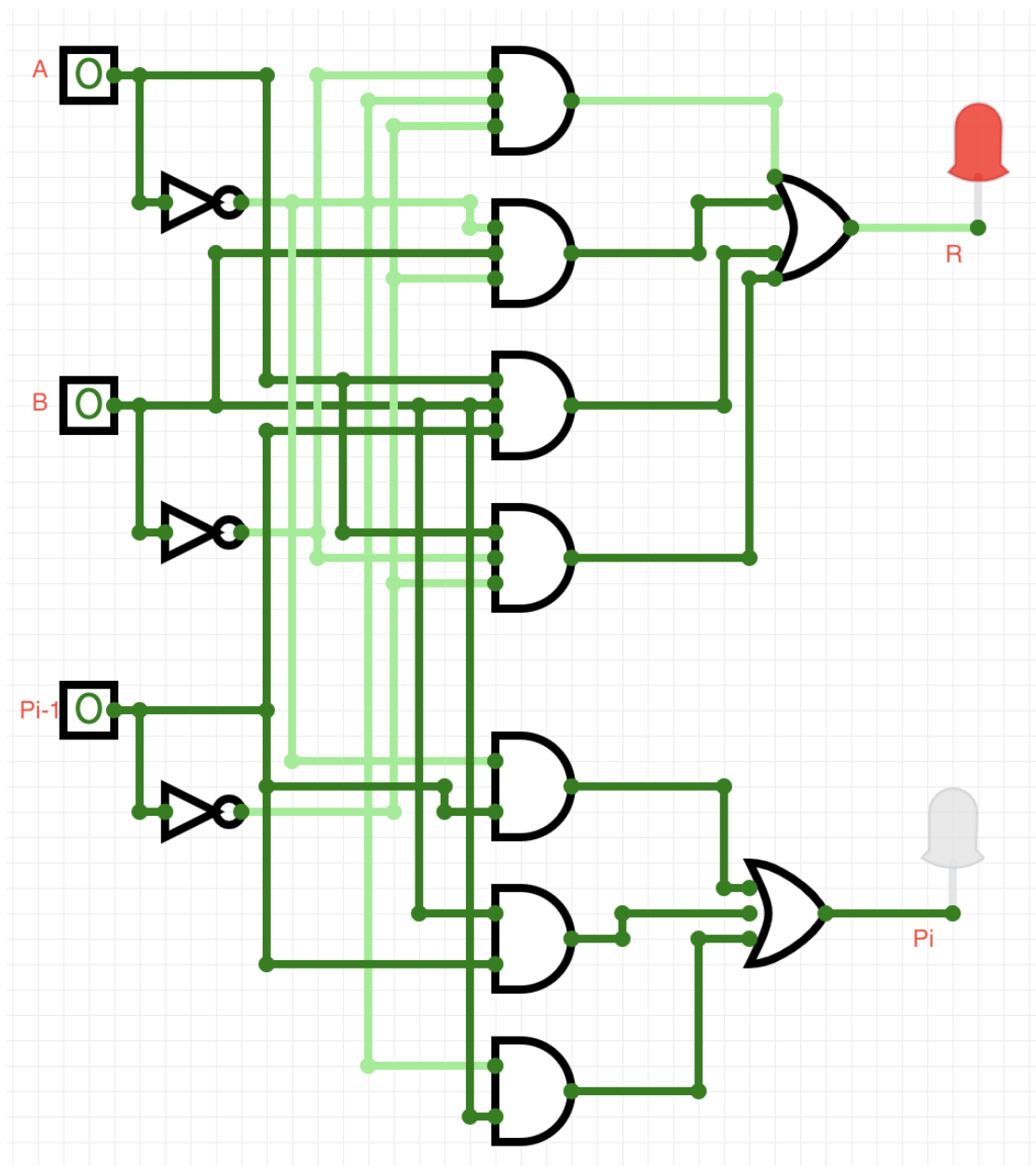
Minimalna postać funkcji wyjściowej dla różnicy na podstawie minimalizacji (dla R)

$$R = \bar{A} \bar{B} P_{i-1} + \bar{A} B \overline{P_{i-1}} + A B P_{i-1} + A \bar{B} \overline{P_{i-1}}$$

Minimalna postać funkcji wyjściowej dla pożyczki na podstawie minimalizacji (dla P_i)

$$P_i = \bar{A} P_{i-1} + B P_{i-1} + \bar{A}$$

1.4 Schemat układu



2. Sterowana bramka 3-bitowa realizująca funkcję NAND lub NOR (na bramkach NAND)

2.1 Opis

Celem drugiego ćwiczenia było skonstruowanie układu sterowanej bramki 3-bitowej realizującej funkcję NAND lub NOR o trzech wejściach (x_2, x_1, x_0), jednym wejściu sterującym (s) oraz jednym wyjściu (y).

- dla $s = 0$ układ realizuje funkcje NAND
- dla $s = 1$ układ realizuje funkcje NOR

2.2 Tabela prawdy

x_2	x_1	x_0	s	y
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	0	0
0	0	0	1	1
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

2.3 Minimalizacja funkcji

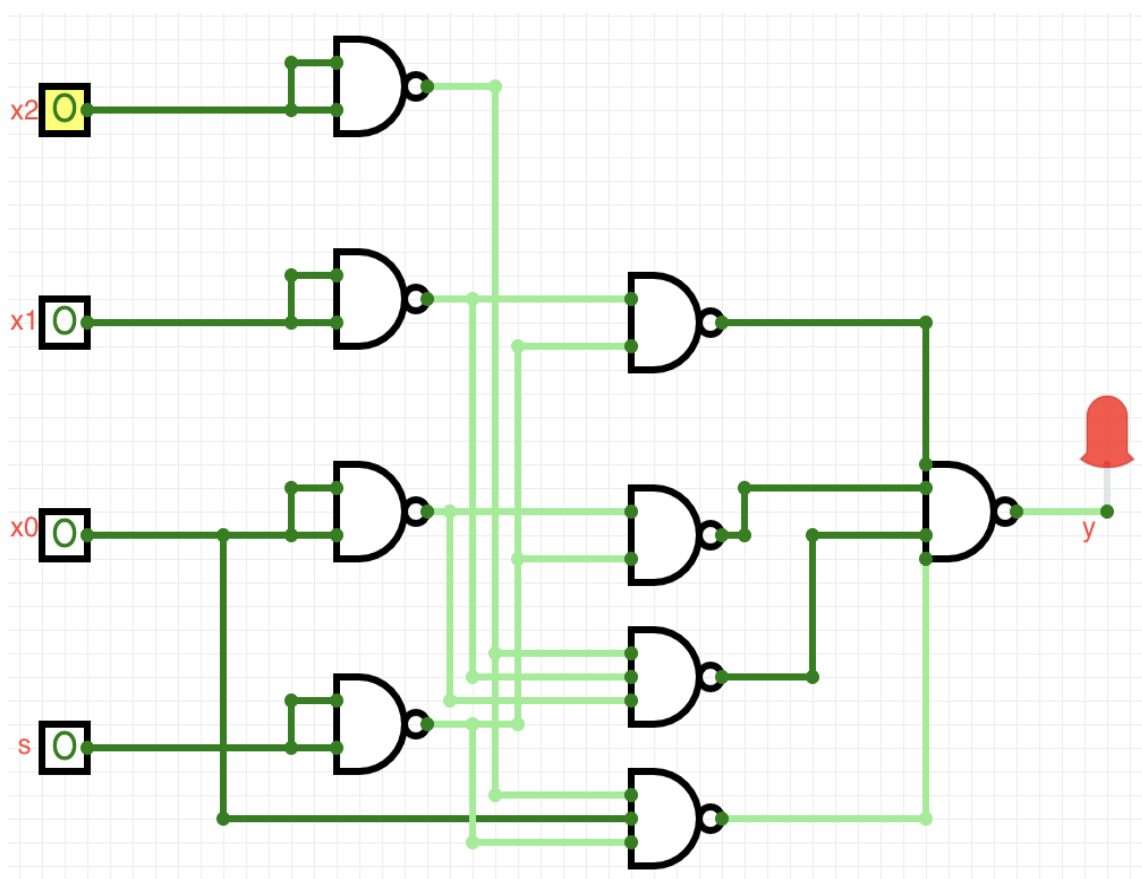
Siatka Karnaugh

$x_2x_1 \backslash x_0s$	0 0	0 1	1 1	1 0
0 0	1	1	0	1
0 1	1	0	0	1
1 0	1	0	0	0
1 1	1	0	0	1

Minimalna postać funkcji wyjściowej na podstawie minimalizacji

$$y = \overline{x_1} \overline{s} + \overline{x_0} \overline{s} + \overline{x_2} \overline{x_1} \overline{x_0} + \overline{x_2} x_0 \overline{s}$$

2.4 Schemat układu



3.Komparator 4-bitowy wykrywający liczby z przedziału [9-13]

3.1 Opis

Układ sprawdza, czy liczba należy do przedziału [9-13], jeśli warunek jest spełniony układ generuje na wyjściu sygnał „1”.

3.2 Tabela prawdy

x_3	x_2	x_1	x_0	y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

x_3 – najstarszy bit liczby
 x_2 – starszy bit liczby
 x_1 – młodszy bit liczby
 x_0 – najmłodszy bit liczby
 y – wynik

3.3 Minimalizacja funkcji

Siatka Karnaugh

$x_1x_0 \backslash x_3x_2$	0 0	0 1	1 1	1 0
0 0	0	0	1	0
0 1	0	0	1	1
1 0	0	0	0	1
1 1	0	0	0	1

Minimalna postać funkcji wyjściowej na podstawie minimalizacji

$$y = \overline{x_1}x_2x_3 + x_0\overline{x_2}x_3 + x_1\overline{x_2}x_3$$

3.4 Schemat układu

