

# Sprawozdanie 1: Bramki i funkcje logiczne

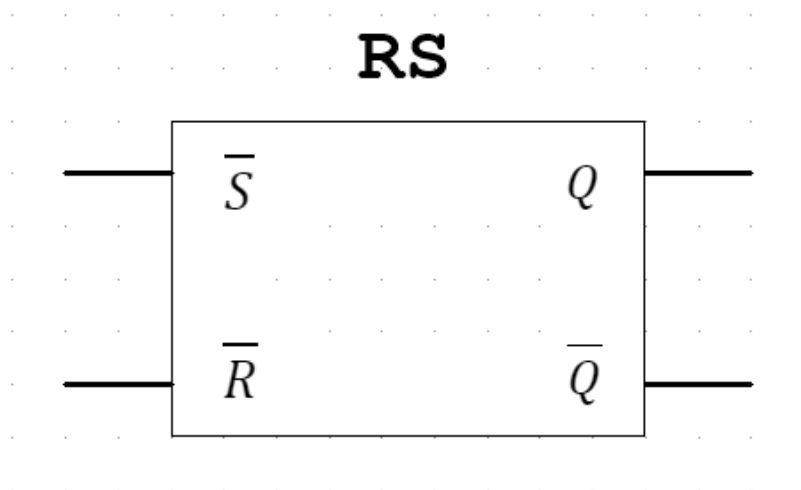
**Jakub Szymczak, Damian Tworek,  
Maksymilian Sulima, Łukasz Wala**

*AGH, Wydział Informatyki, Elektroniki i Telekomunikacji  
Technika Cyfrowa 2021/2022*

Kraków, 4 maja 2022

## 1 Ćwiczenie 2a

Ideą ćwiczenia jest, na podstawie tabel prawdy, zaprojektować i praktycznie zrealizować asynchroniczny przerzutnik RS w oparciu o dwie bramki NAND, po czym jednoznacznie przetestować poprawność jego działania w programie Multisim.



Rysunek 1: Układ 2a

Zasada działania przerzutnika jest następująca: jeżeli na wejście  $\bar{S}$  podane zostanie logiczne zero, wówczas wymusi ona stan jeden na wyjściu  $Q$  (stan

”set”), natomiast logiczne zero na wejściu  $\bar{R}$  wymusi zero na wyjściu  $Q$  (stan ”reset”). Przejście przerzutnika do stanu neutralnego, gdzie na obu wejściach są jedynki, zachowuje poprzedni stan na wyjściu. Stan z dwoma zerami na wejściach jest zabronionym, dla którego wyjście jest nieokreślone. Również przed pierwszą zmianą stanu stan jest nieokreślony.

## 1.1 Rozwiązanie teoretyczne

Rozpocznijmy od stworzenia tabeli przejść, gdzie  $Q_{n-1}$  jest poprzednim stanem przerzutnika

$\bar{S}$	$\bar{R}$	$Q$	$\bar{Q}$
0	1	1	0
1	0	0	1
1	1	$Q_{n-1}$	$\bar{Q}_{n-1}$
0	0	-	-

Tabela 1: Tabela przejść

Na podstawie tabeli 1 skonstruowana zostanie tabela Karnough dla  $Q$

$Q_{n-1} \backslash \bar{S}\bar{R}$	00	10	11	01
0	-	0	0	1
1	-	0	1	1

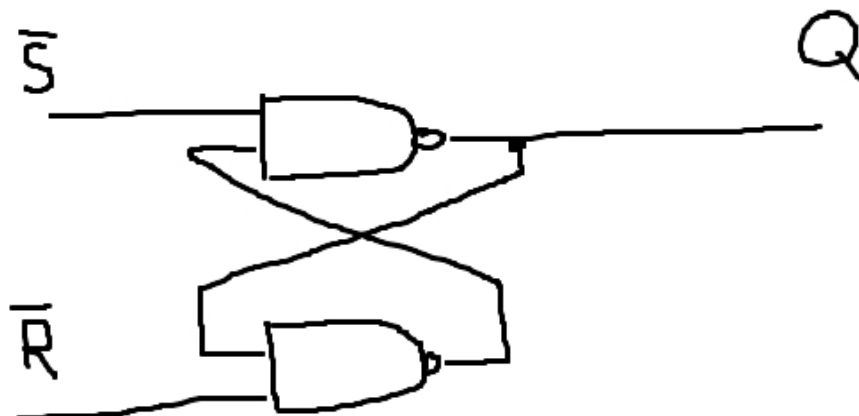
Rysunek 2: Tabela Karnough dla  $Q$

Poniżej wzór na podstawie tabeli Karnough, z prawa podwójnej negacji

$$Q = \bar{\bar{S}} + \bar{R}Q_{n-1} = S + \bar{R}Q_{n-1}$$

Korzystając z prawa de Morgana

$$Q = \overline{\bar{S} \bar{R} Q_{n-1}}$$



Rysunek 3: Projekt układu dla  $Q$

Poniżej tabela Karnough dla  $\bar{Q}$

$Q_{n-1} \backslash \bar{S}\bar{R}$	00	10	11	01
0	-	1	1	0
1	-	1	0	0

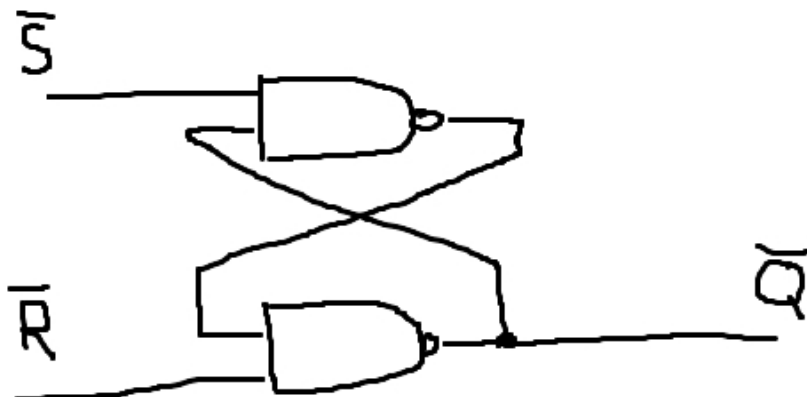
Rysunek 4: Tabela Karnough dla  $\bar{Q}_n$

Poniżej wzór na podstawie tabeli Karnough dla  $\bar{Q}_n$ , z prawa podwójnej negacji

$$\bar{Q} = \bar{\bar{R}} + \bar{S} \bar{Q}_{n-1} = R + \bar{S} \bar{Q}_{n-1}$$

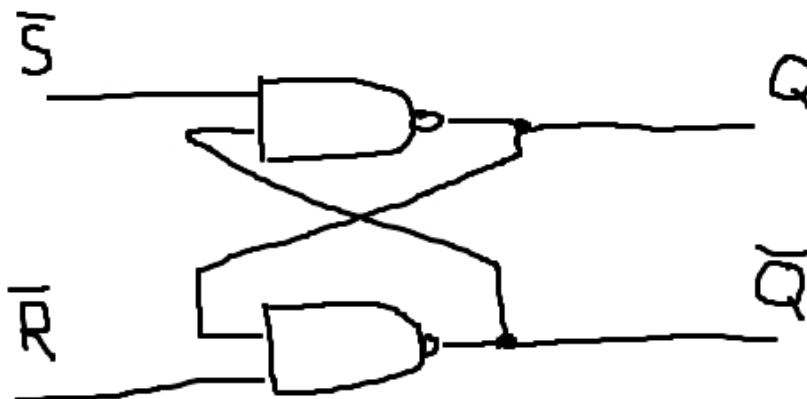
Korzystając z prawa de Morgana

$$\bar{Q} = \overline{\bar{R} \bar{S} \bar{Q}_{n-1}}$$



Rysunek 5: Projekt układu dla  $\overline{Q}$

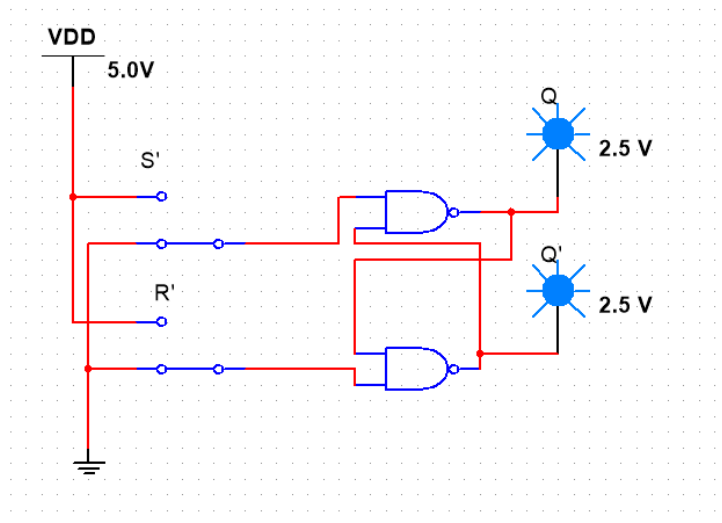
Można zauważyć, że układy te są symetryczne, więc można je połączyć, żeby otrzymać zarówno  $Q$  oraz jego negację.



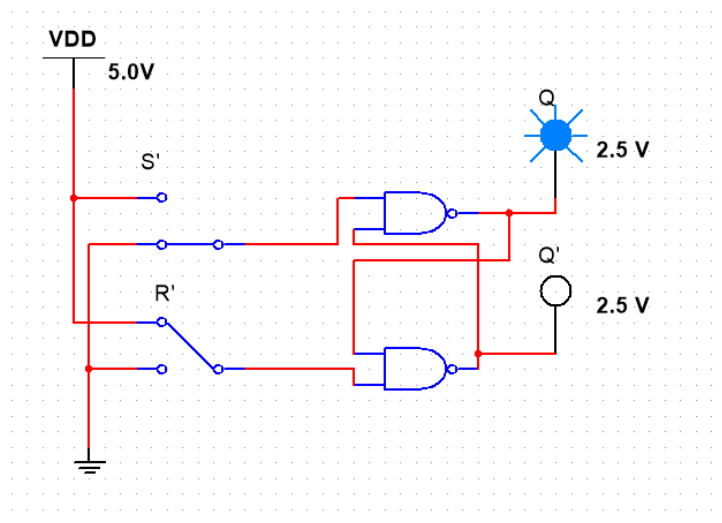
Rysunek 6: Projekt układu

## 1.2 Implementacja w programie Multisim

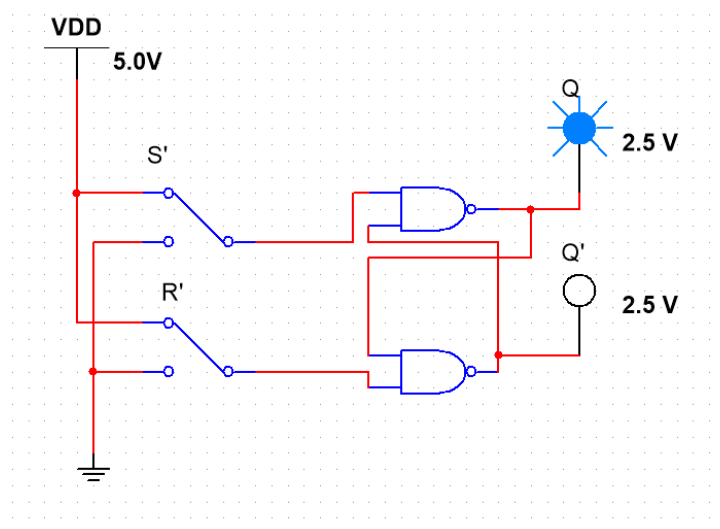
Poniżej implementacja układu w programie Multisim dla wszystkich możliwych stanów



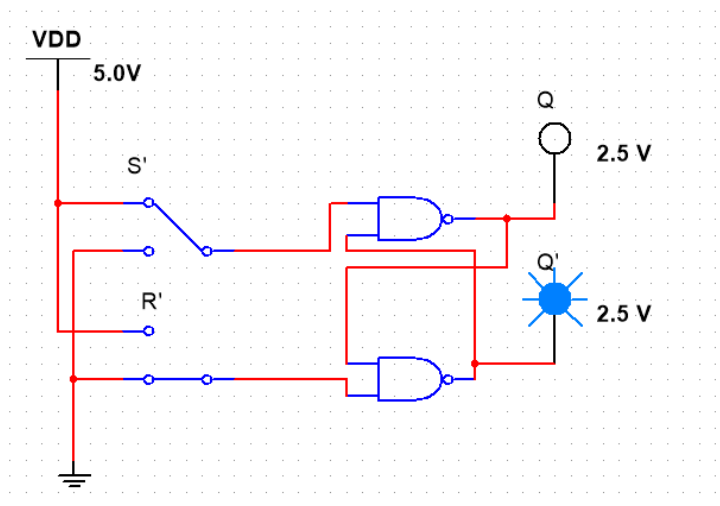
Rysunek 7: Układ w stanie nieokreślonym



Rysunek 8: Układ w stanie "set"

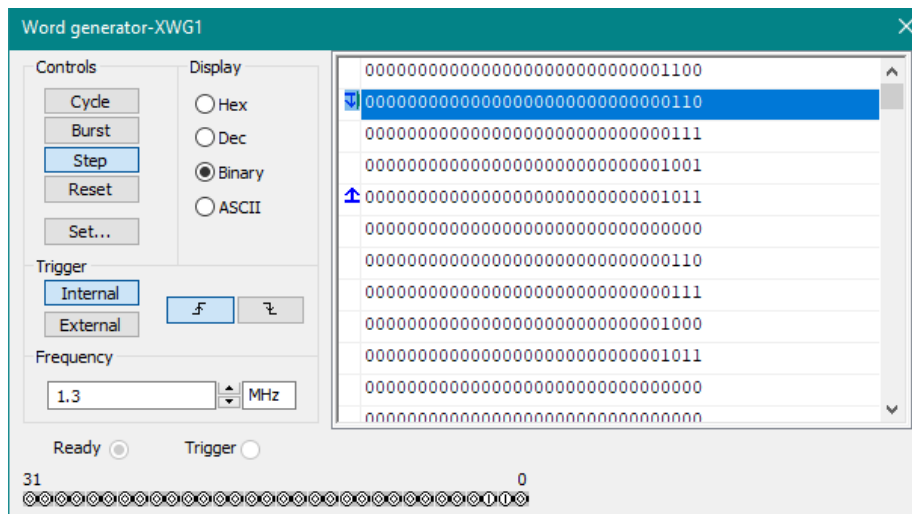


Rysunek 9: Układ w stanie neutralnym, gdy w poprzednim był w stanie "set"

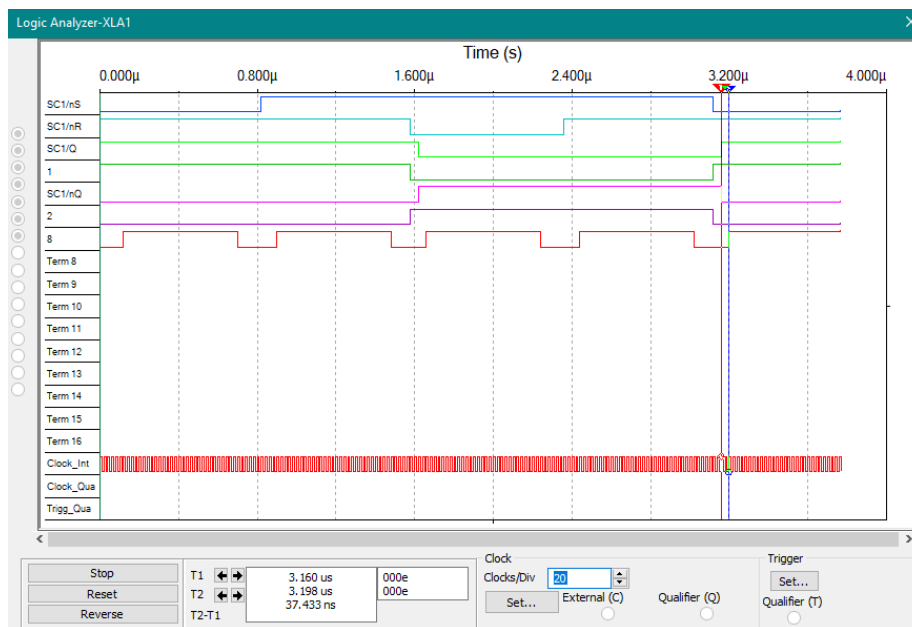


Rysunek 10: Układ w stanie "reset"





Rysunek 13: Generator układu testującego



Rysunek 14: Analizator dla układu testującego

Generator słów na dwóch pierwszych bitach zawiera poszczególne możliwe wejścia do przerzutnika, natomiast na dwóch kolejnych bitach są poprawne wyjścia, które są konfrontowane z wyjściami z przerzutnika przy pomocy bramek



XOR. Ze względu na czas propagacji sygnału użyty został synchroniczny przerzutnik D, którego wejście CLK połączone jest z wyjściem R z generatora, które mówi, że dane z generatora są gotowe, wówczas nie ma sytuacji, że moment zmiany wartości generatora zostanie odczytany jako błąd testowanego przerzutnika. Do tego wykorzystano wbudowany do programu Multisim przerzutnik RS po to, żeby po znalezionym błędzie lampka nie gasła.

### 1.3 Wnioski

- Przerzutnik RS jest przerzutnikiem asynchronicznym, co czyni go podatnym na przypadkowe zmiany wejścia (na przykład spowodowane napięciem wyindukowanym na ścieżce czy przewodzie). Remedium na te problemy są przerzutniki synchroniczne, które zmieniają stan tylko w określonych momentach zmian stanów zegara.
- Alternatywnym podejściem byłoby wyprowadzenie wzorów na podstawie tabeli Karnough zaznaczając grupy zer:

$\overline{S}\overline{R}$ $Q_{n-1}$	00	10	11	01
0	-	0	0	1
1	-	0	1	1

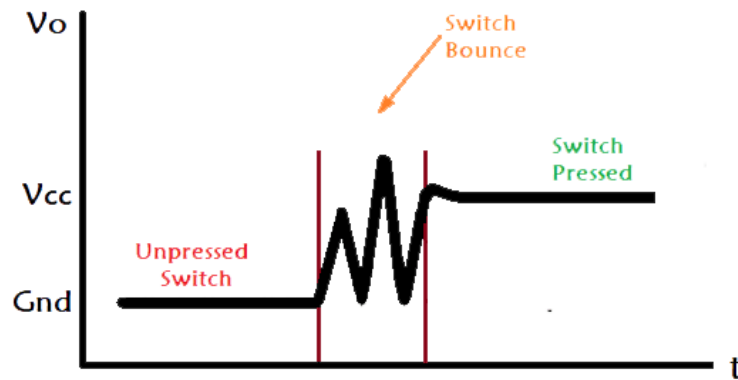
Rysunek 15: Alternatywna tabela Karnough

Wówczas wzór wygląda następująco:

$$Q = \overline{R}(\overline{S} + \overline{Q}_{n-1}) = R(\overline{S} + \overline{Q}_{n-1}) = R(\overline{\overline{S}\overline{Q}_{n-1}}) = R(\overline{SQ}_{n-1})$$

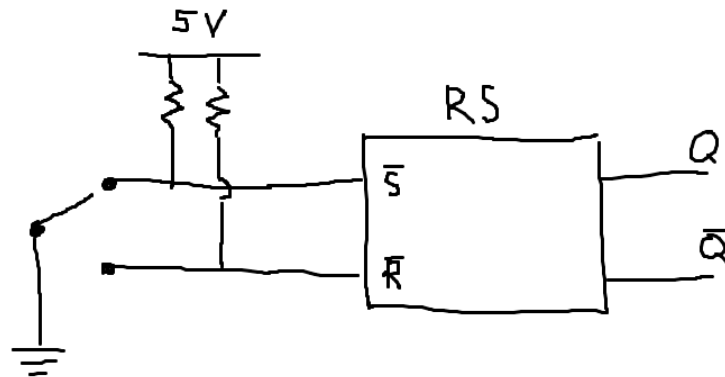
W tym przypadku konieczne byłoby użycie więcej niż dwóch bramek NAND, co eliminuje to podejście.

- Podczas używania fizycznych przycisków w układach elektronicznych często może wystąpić efekt, gdzie metalowe styki odbijają się od siebie podczas przyciśnięcia i układ rejestruje kilka przyciśnięć zamiast jednego:



Rysunek 16: Switch bounce

Asynchroniczny przerzutnik RS może posłużyć jako "debouncer" dla np. takiego przycisku w klawiaturze

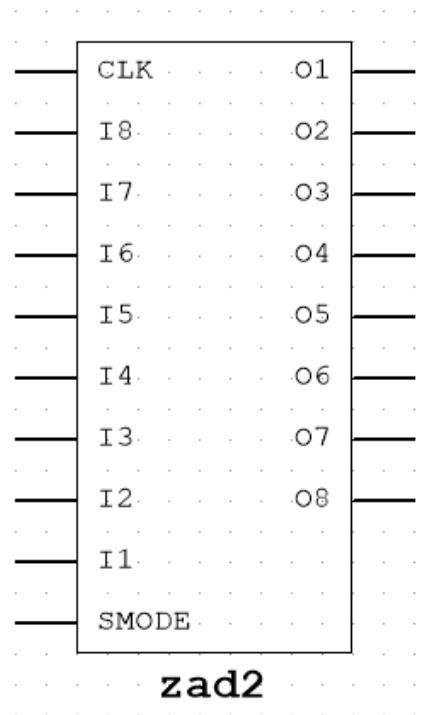


Rysunek 17: Switch debouncer z użyciem przerzutnika RS

## 2 Ćwiczenie 2b

Ideą ćwiczenia jest zbudowanie rejestru pierścieniowego. Powinien on realizować dwie podstawowe funkcje wybierane przy pomocy pojedynczego przełącznika trybu pracy:

- powinien umożliwiać załadowanie do rejestru dowolnej ośmiobitowej wartości przy pomocy ośmiu przełączników,
- powinien sprawić, że wpisana wcześniej wartość będzie w sposób ciągły krążyła w rejestrze. Np. skrajny bit z prawej strony powinien pojawić w następnym cyklu zegarowym na pierwszej pozycji z lewej strony.



Rysunek 18: Układ 2b

**I1...I8** są wejściami, którymi można ustawić poszczególne bity, **O1..O8** to wyjścia, natomiast **CLK** to wejście zegara, **SMODE** to wejście włączające możliwość ustawiania wartości w rejestrze.

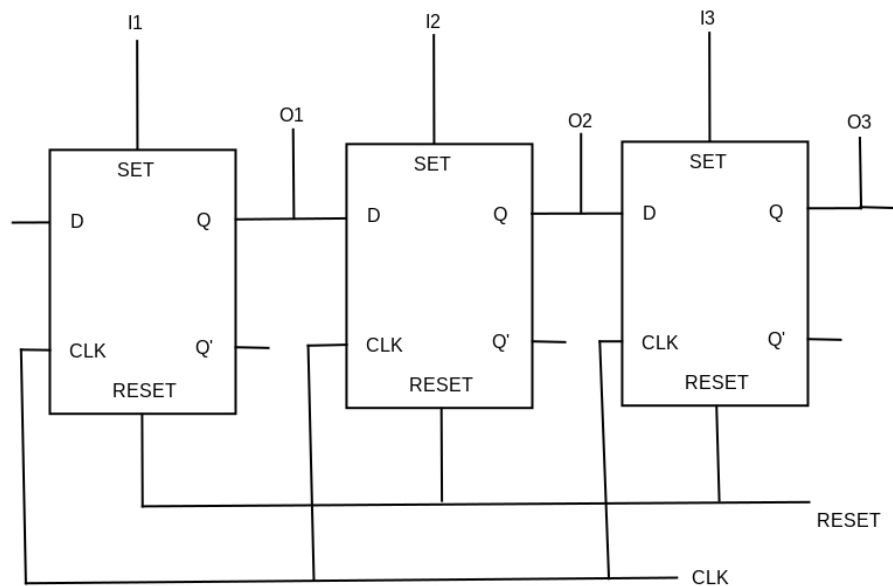
## 2.1 Pomysł 1 - schemat koncepcyjny i implementacja

Do zbudowania układu użyte zostaną przerzutnik typu **D** pokazane na poniższym rysunku. Przerzutnik typu **D** przepisuje stan wejścia informacyjnego **D** na wyjście **Q**, przepisanie informacji następuje tylko przy odpowiednim stanie wejścia zegarowego **CLK**, w tym przypadku przy zmianie jego stanu z 0 na 1. Dodatkowo wejścia **SET** oraz **RESET** działają analogicznie jak w przerzutniku RS.

D	Q(t)	Q(t+1)
0	0	0
0	1	0
1	0	1
1	1	1

Rysunek 19: Tabela przejść dla przerzutnika D

Układ bazowany na pierwszym pomysłe (udoskonalony w sekcjach "Pomysł 2") będzie się składał z ośmiu przerzutników D, gdzie wyjście **Q** przerzutnika poprzedniego połączone jest z wejściem **D** przerzutnika kolejnego, natomiast ostatnie przerzutnik połączony jest z pierwszym. Tym sposobem przerzutniki będą przekazywać sobie kolejne stany. Poniżej wycinek układu:

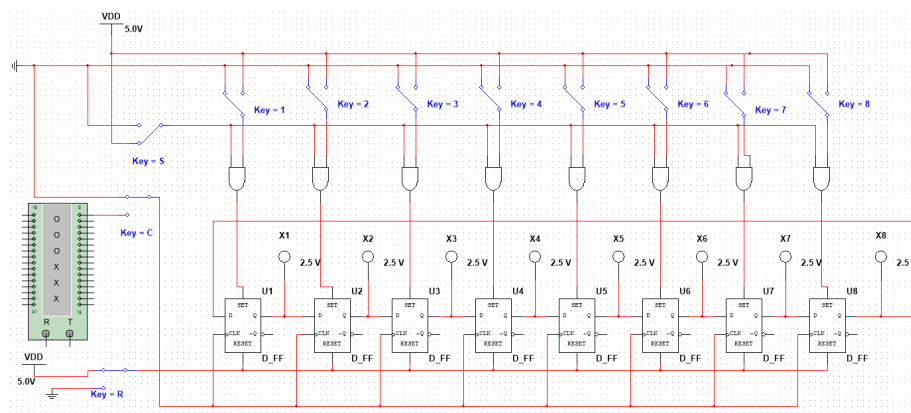


Rysunek 20: Schemat koncepcyjny pierwszej wersji układu 2b

Łaładowanie do rejestru początkowej wartości zostało zrealizowane za pomocą asynchronicznych wejść przerzutników D - **SET**. Do tego została również do-

dana możliwość resetowania wartości w rejestrze zrealizowana za pomocą wejść **RESET**.

Schemat zaimplementowany w programie Multisim wygląda następująco:

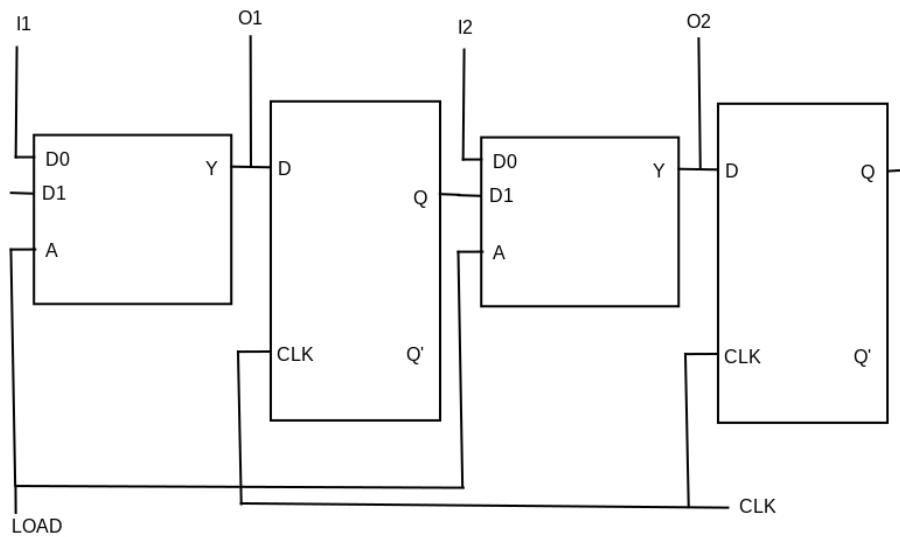


Rysunek 21: Implementacja pierwszej wersji układu 2b

Rozwiązanie to jednak nie jest idealne - do ustawiania wartości w rejestrze wykorzystywane są operacje asynchroniczne, które mogą być ryzykowne, lepsze rozwiązanie zaprezentowane jest w kolejnej sekcji.

## 2.2 Pomysł 2 - schemat koncepcyjny i implementacja

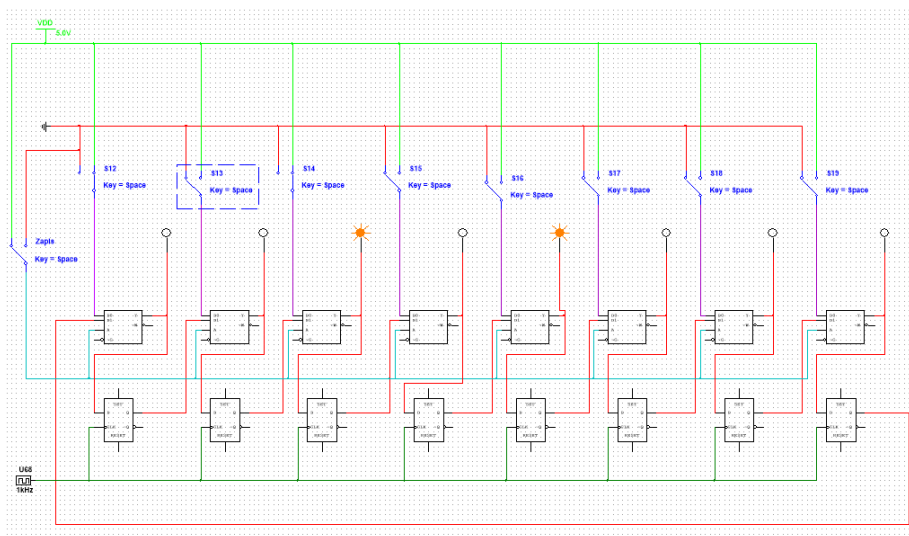
Drugi pomysł odrzuca używanie operacji asynchronicznych, natomiast do wprowadzania wartości do rejestru wykorzystuje multiplexery w sposób zaprezentowany poniżej (również użyto przerzutników D).



Rysunek 22: Schemat ulepszonej wersji układu 2b

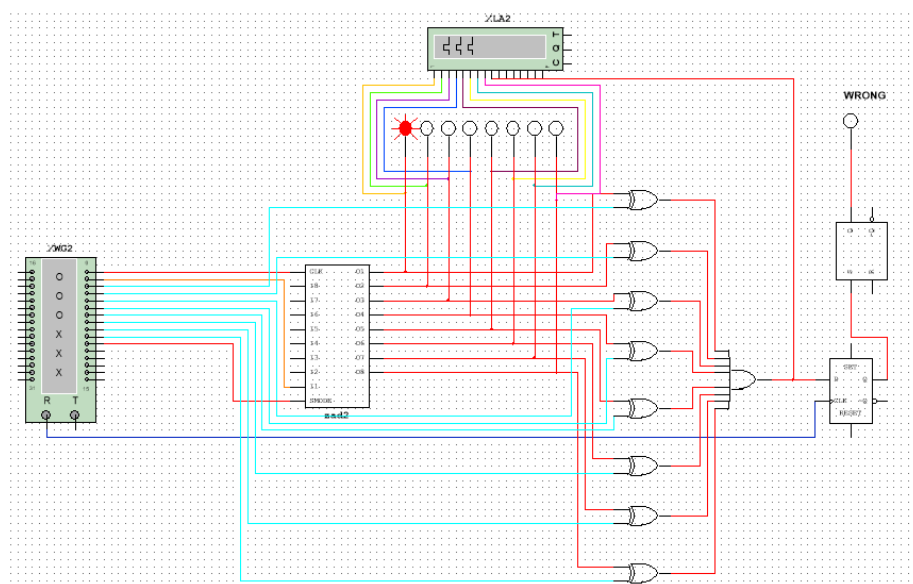
Wejście **LOAD**, gdy ustawione na jedynkę logiczną, pozwala na załadowanie wartości do rejestru, w przeciwnym wypadku pobierane są wartości z poprzedzającego multiplekser przerzutnika D.

Poniżej implementacja układu w programie Multisim:

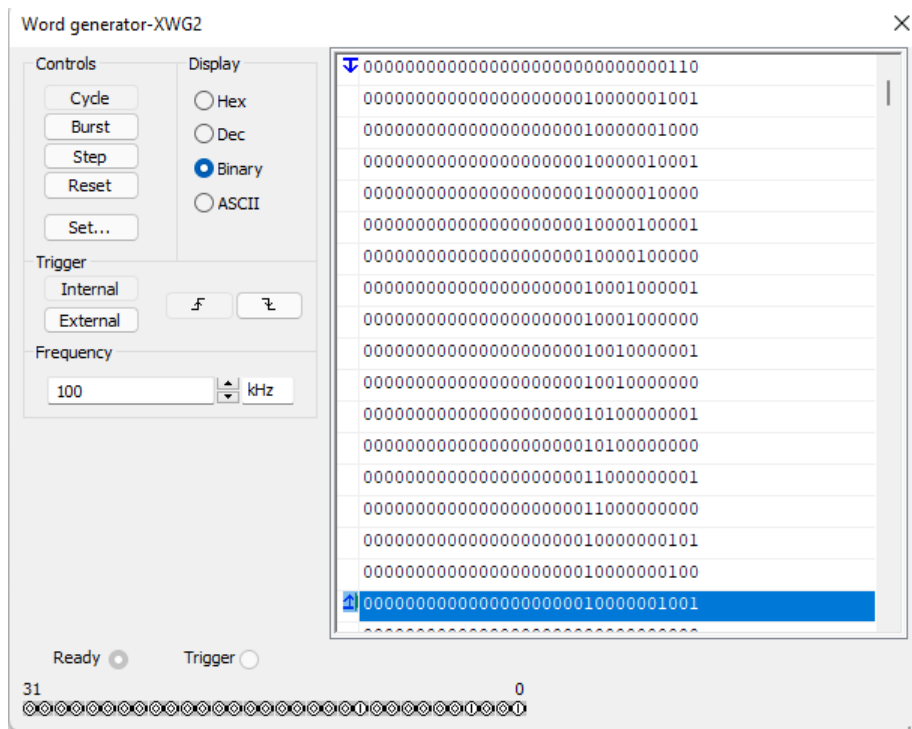


Rysunek 23: Implementacja ulepszonej wersji układu 2b

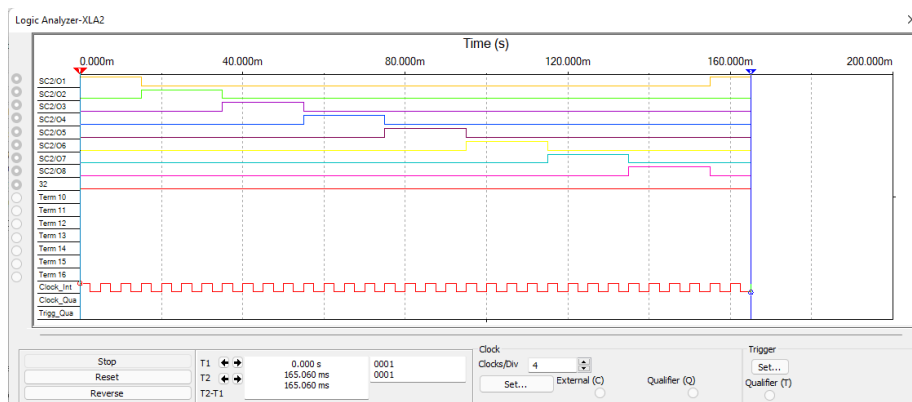
Poniżej automatyczny układ testujący



Rysunek 24: Automatyczny układ testujący



Rysunek 25: Generator słów dla układu testującego



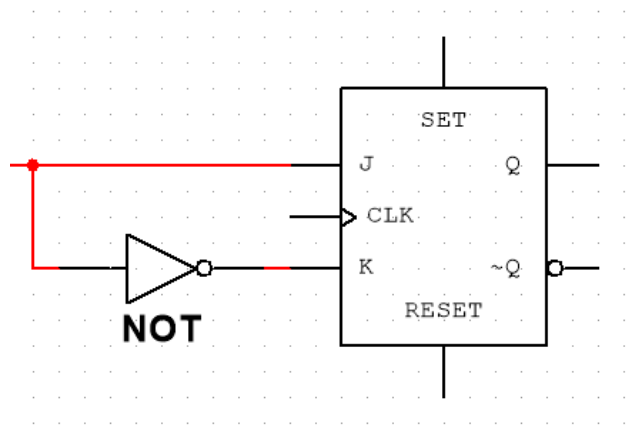
Rysunek 26: Analizator dla układu testującego

## 2.3 Wnioski

- Alternatywnym podejściem byłoby zastosowanie innego rodzaju przerzutnika synchronicznego, np. przerzutnika JK. Przerzutnik JK może być ła-

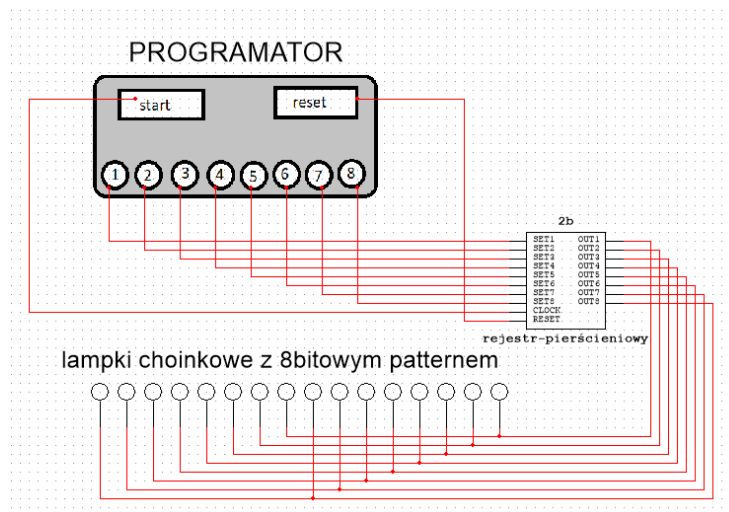


two przekonwertowany tak, żeby spełniał funkcję przerzutnika D potrzebnego w ćwiczeniu.



Rysunek 27: Przerzutnik D stworzony przy pomocy przerzutnika JK

- Rejestru pierścieniowego można użyć jako urządzenia sterującego światłami choinkowymi. Np. niech co ósme światło będzie przypięte do każdego wyjścia rejestru (1 wyjście do 1 światła, 9 światła, 17 światła, 2 wyjście do 2 światła, 10 światła itd.), wówczas można łatwo sterować sekwencjami światełek.



Rysunek 28: Światła choinkowe