

Sprawozdanie

Ćwiczenie 5

Jan Kwinta, grupa 12

Prowadzący ćwiczenia: dr. Szymon Niedźwiedzki

Data wykonania: 24 maja 2023

Wstęp

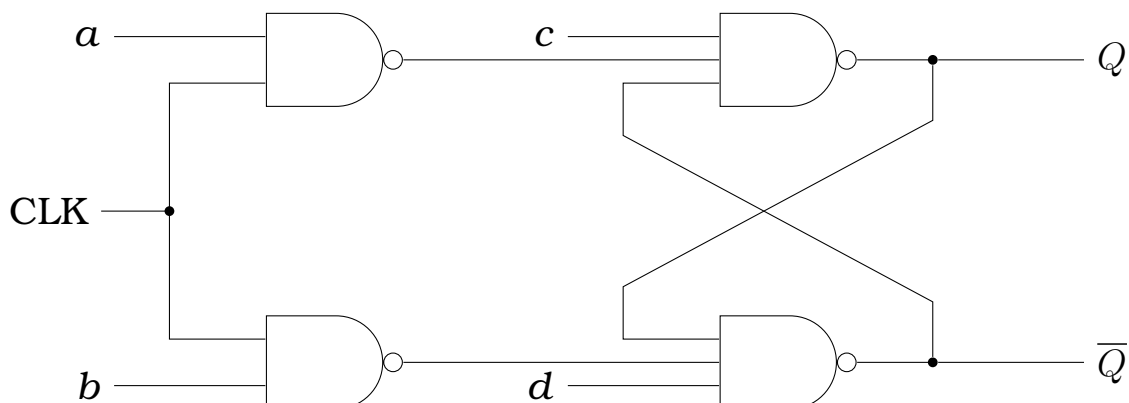
Przedmiotem piątych laboratoriów z elektroniki były układy cyfrowe o większym stopniu złożoności niż na poprzednich zajęciach. Montowaliśmy układy synchroniczne — przerzutniki synchroniczne, liczniki binarne, dzielniki częstotliwości i rejestry. Używaliśmy do ich budowy wiele różnych układów scalonych z serii 74: 7400 i 7410 (bramki NAND dwu- i trzywejściowe), 7474 (przerzutnik jednoboczowy D), 7475 (D *latch*), 7493 (przerzutnik JK) oraz 74164 i 74165 (rejestry).

Układy synchroniczne to takie układy sekwencyjne, których zmiana stanu wewnętrznego następuje wyłącznie w określonych chwilach. Chwile te wyznacza sygnał zegarowy (ang. *clock*), czyli generowany przez osobny układ sygnał o kształcie prostokątnym o stałej częstotliwości i amplitudzie wachającej się pomiędzy logiczną prawdą a logicznym fałszem.

Do realizacji zadań używałem jako sygnału zegarowego fali prostokątnej generatora, którą ustawiałem na małą częstotliwość (w okolicach 0.5 Hz), aby móc operować wejściami układów (impulsatorami na płytce UC-2) w różnych momentach okresu zegara. Amplituda sygnału wynosiła 5 V_{pp} . Wartości sygnału zawierały się pomiędzy 0 V a 5 V , co jest równoważne sygnałom logicznym fałsz i prawda na płytkach UC-2.

Ćwiczenie 5.1

Zmontowanie synchronicznego przerzutnika RS korzystając z układów scalonych 7400 i 7410. Zbadanie działania przerzutnika oraz własności wejść a , b , c i d .



Zmontowałem układ na płytce UC-2 według powyższego schematu, korzystając z bramek NAND: dwuwejściowych (7400) i trzywejściowych (7410). Układ ten posiada pięć wejść i dwa wyjścia. Pod wejście zegarowe (CLK) podłączyłem sygnał o częstotliwości 0.1 Hz z generatora. Wejścia a oraz b połączyłem z wyjściami Q impulsatorów na płytce. Na wejścia c oraz d podawałem stan logiczny prawda i fałsz spinając je odpowiednio do $+5\text{ V}$ przez opornik $1\text{ k}\Omega$ i do 0 V przez opornik $390\ \Omega$. Wyjście Q przerzutnika podłączyłem do próbnika stanów logicznych.

Z zegarem, zmieniając stany wejść c i d oraz operując impulsatorami zmierzyłem stan wyjścia Q przy wszystkich różnych stanach wejść a , b , c , d i CLK. Nagrania z pomiarów znajdują się w filmikach MP4 w pliku Sprawozdanie-cwiczenie5-Jan-Kwinta-Zalacznik1.zip. Na filmikach taktowanie zegara pokazuje skrajnie prawa dioda na płytce.

Wyniki pomiarów przedstawiłem w czterech tabelach zamieszczonych poniżej. Litera p oznacza stan pamiętania.

$c = 0 \quad d = 0$					
CLK = 1			CLK = 0		
a	b	Q	a	b	Q
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	1

$c = 0 \quad d = 1$					
CLK = 1			CLK = 0		
a	b	Q	a	b	Q
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	1

$$c = 1 \quad d = 0$$

CLK = 1			CLK = 0		
a	b	Q	a	b	Q
0	0	0	0	0	0
0	1	0	0	1	0
1	0	1	1	0	0
1	1	1	1	1	0

$$c = 1 \quad d = 1$$

CLK = 1			CLK = 0		
a	b	Q	a	b	Q
0	0	p	0	0	p
0	1	0	0	1	p
1	0	1	1	0	p
1	1	-	1	1	p

Analizując stany wyjścia w zależności od stanów wejść jestem w stanie zaobserwować kilka faktów:

- Gdy na wejście zegarowe podany jest sygnał niski (0) układ nie reaguje na zmiany stanów wejść a i b .
- Gdy na wejście c podany jest stan logiczny 0 na wyjściu Q przerzutnika jest stan logiczny 1 niezależnie od innych wejść, co czyni przerzutnik transparentnym w logice NAND.
- Gdy na obydwa wejścia c i d podany jest sygnał wysoki to przerzutnik zachowuje się tak jak zwykły synchroniczny przerzutnik RS.

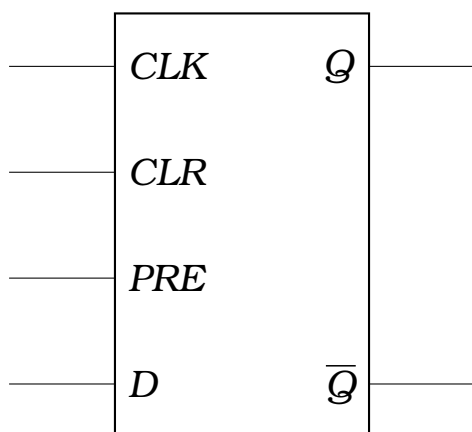
Można z tych obserwacji wyciągnąć następujące wnioski: Po pierwsze — synchroniczny przerzutnik RS może zmieniać swój stan wewnętrzny tylko wtedy, gdy na wejściu zegarowym jest stan logicznej jedynki. Po drugie — podanie fałszu na wejścia c i d powoduje odjęcie funkcjonalności układowi. Spięcie razem tych wejść stworzyłoby wejście włączające, czyli tak zwane *chip enable* lub *chip select* oznaczane zazwyczaj jako CE albo CS, które pozwala na selektywne włączanie i wyłączanie rejestru. Gdy na wejście CE podany jest sygnał 1 to cały układ działa, zachowując się jak normalny przerzutnik. Gdy wyłączymy układ, czyli podamy na wejście CE sygnał 0, nasz przerzutnik staje się dla jakiegoś większego układu, którego może być częścią, transparentny, czyli podaje na swoje wejście 1 niezależnie od stanu zegara czy wejść a oraz b .

Wiele bardziej zaawansowanych komponentów elektronicznych: mikroprocesory, układy pamięci, itp. posiada wejście CE lub CS, co pozwala na wygodne ich wykorzystywanie w dużych układach i urządzeniach. Umożliwia to na przykład sterowanie jednemu procesorowi wieloma układami pamięci podłączonymi do jednej magistrali komunikacyjnej.

Ćwiczenie 5.2

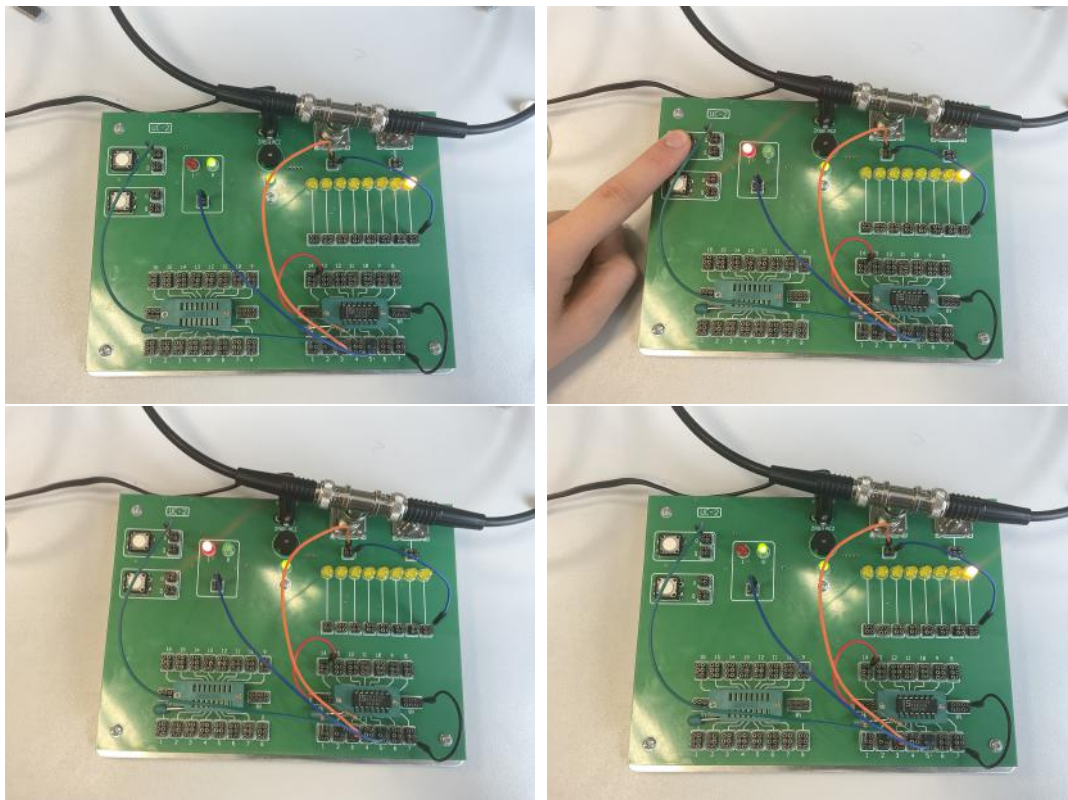
Zbadanie działania przerzutnika jednozboczowego D (układ scalony 7474), podając na odpowiednie wejścia sygnał zegara, sygnały z impulsatora lub spinając z 0 V albo 5 V.

Układ przerzutnika D znajdującego się na układach 7474 można przedstawić schematycznie tak:

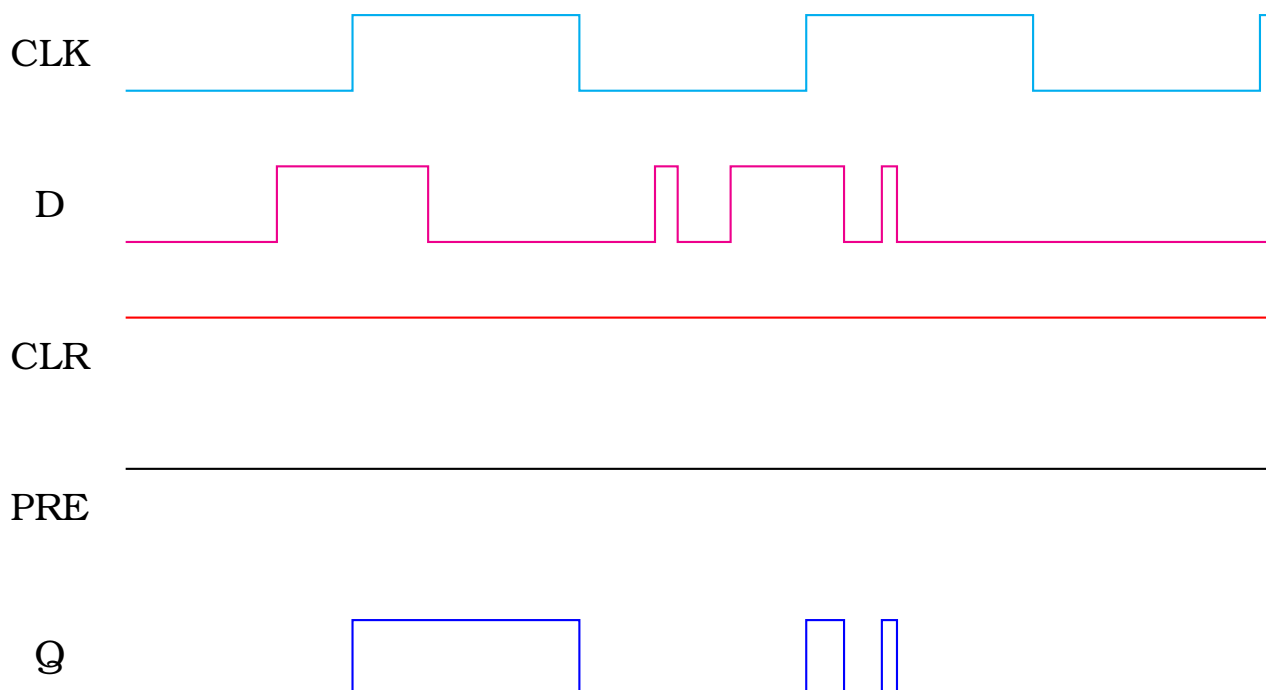


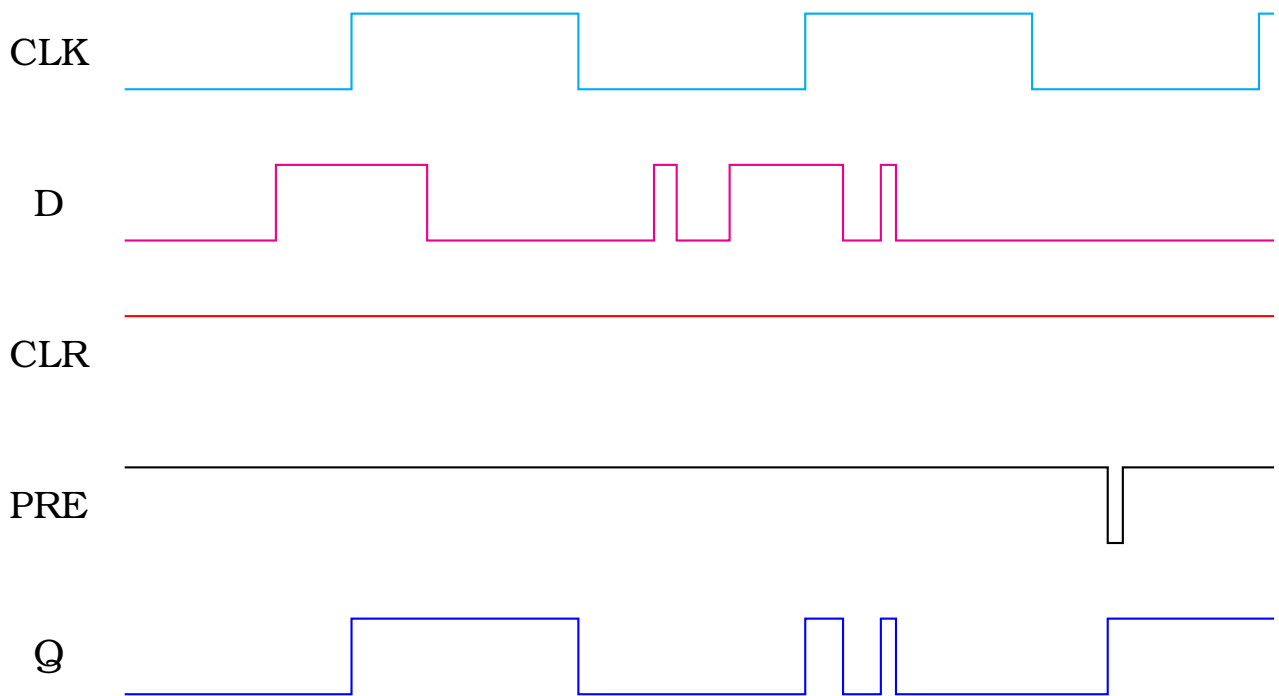
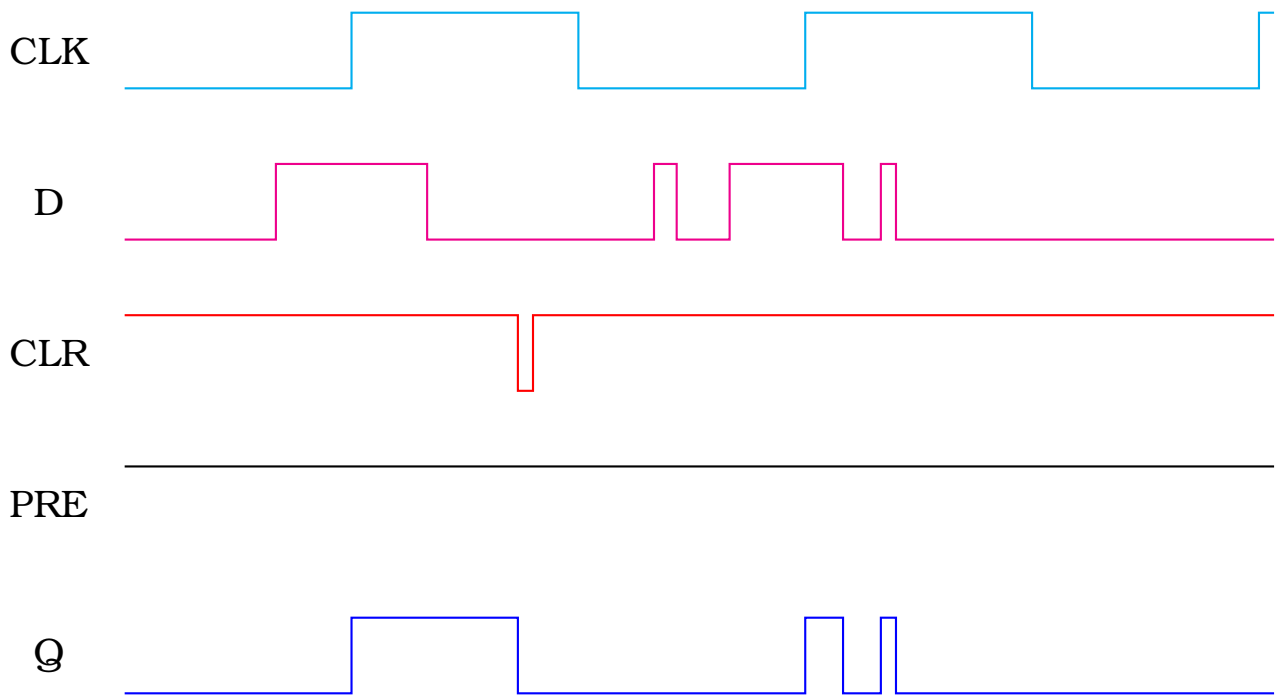
Przerzutnik ten posiada 4 wejścia i dwa wyjścia. Wejście zegarowe oznaczone jest CLK (*clock*), wejście na dane — D (*data*). Wejścia CLR i PRE to odpowiednio *clear* oraz *preset*. Z dokumentacji producenta możemy dowiedzieć się, że są to wejścia normalizowane do stanu wysokiego (czyli przy podaniu na obydwa sygnału prawdziwego układ 7474 jest operacyjny jako przerzutnik D). Podanie sygnału niskiego na wejście CLR "czyści" przerzutnik, czyli ustawia wyjście Q na 0 niezależnie od wejść CLK i D. Podanie sygnału niskiego na wejście PRE "ustawia" przerzutnik, czyli ustawia wyjście Q na 1 niezależnie od wejść CLK i D. Podanie na obydwa wejścia CLR i PRE sygnału 0 jest stanem zabronionym.

Aby zbadać układ przerzutnika podłączyłem obydwa wejścia PRE i CLR "na sztywno" pod sygnał wysoki. Do wejścia CLK podpiąłem sygnał z generatora (na zdjęciach stan zegara widać na skrajnie prawej żółtej diodzie). Jeden z impulsatorów przyłączyłem do wejścia D. Skonstruowałem układ jak na zdjęciach na kolejnej stronie. Następnie podłączyłem wejście CLR a potem wejście PRE pod drugi impulsator (niestety z tych badań nie mam zdjęć).



Zbadane przeze mnie przebiegi wyjścia Q w zależności od wejść CLK, CLR, PRE i D wyglądały następująco:





Na podstawie tych obserwacji można więc wnioskować, że tabela stanów logicznych układu 7474 wygląda następująco:

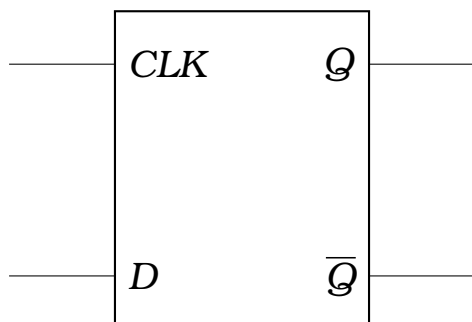
PRE	CLR	CLK	D	Q
0	0	x	x	-
0	1	x	x	1
1	0	x	x	0
1	1	0	x	p
1	1	1	0	0
1	1	1	1	1

gdzie p oznacza stan pamiętania, - oznacza stan niedozwolony, a x oznacza dowolną wartość logiczną (czyli, że w danej chwili wartość wyjścia Q nie zależy od stanu tego wejścia).

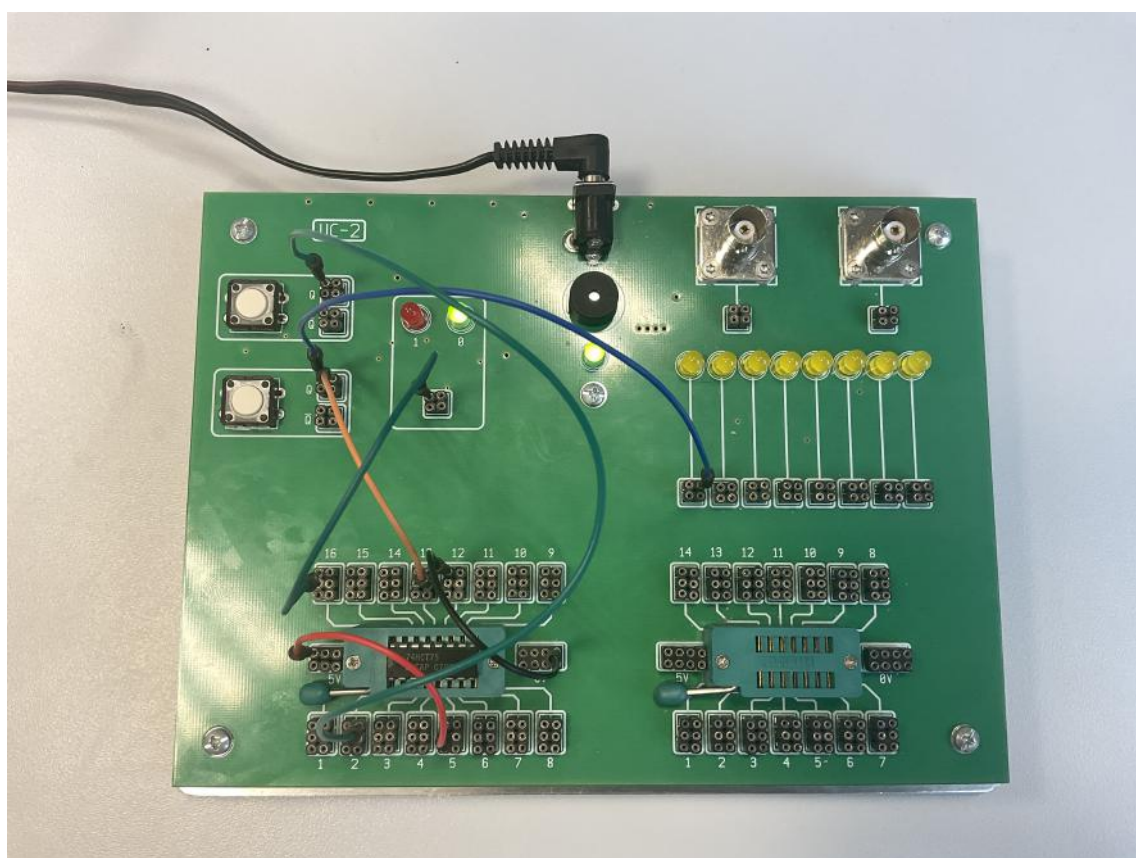
Ćwiczenie 5.3

Zbadanie działania przerzutnika *D-latch* (układ scalony 7475), podając na odpowiednie wejścia sygnał zegara lub sygnały z impulsatora.

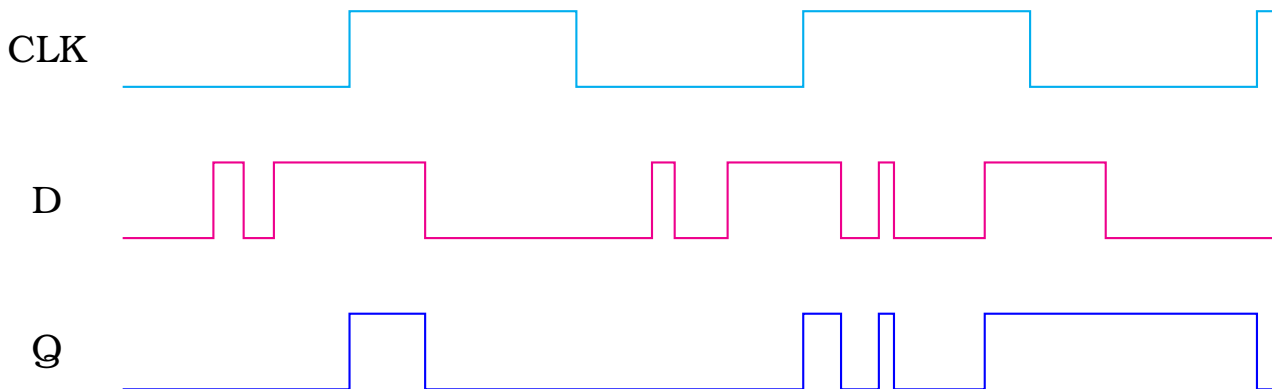
Układ przerzutnika *D-latch* znajdującego się na układach 7475 można przedstawić schematycznie tak:



Skonstruowałem następujący układ: wyjście *Q* przerzutnika podłączyłem pod próbnik stanów logicznych. Wyjście pierwszego impulsatora podpiąłem pod wejście zegarowe *CLK* oraz diodę sygnalizującą stan zegara. Wejście drugiego impulsatora podłączyłem do wejścia *D* przerzutnika.



Zmieniając stany obydwu wejść obserwowałem wartości na wyjściu Q . Zdjęcia załączam w pliku Sprawozdanie-cwiczenie5-Jan-Kwinta-Zalacznik2.zip. Przebiegi, które obserwowałem na wyjściu w zależności od wejść CLK i D wyglądały następująco:



Na podstawie tych pomiarów można wywnioskować tabelę stanów logicznych układu 7475:

CLK	D	Q
0	x	p
1	0	0
1	1	1

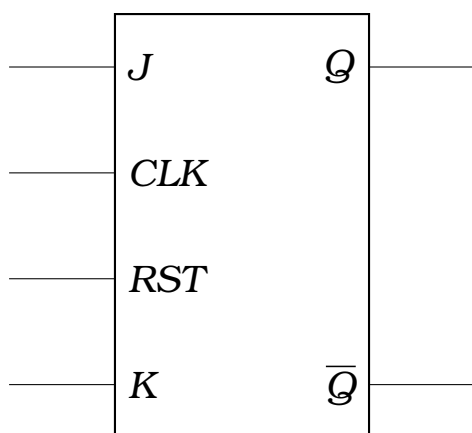
gdzie p oznacza stan pamiętania, a x oznacza dowolną wartość.

Największa różnica pomiędzy przerzutnikiem D -latch wyzwalanym poziomem a przerzutnikiem synchronicznym RS jest taka, że D -latch nie posiada stanów zabronionych. Analizując tabelę stanów logicznych obydwu tych przerzutników można dojść do wniosku, że gdyby spiąć razem wejścia przerzutnika RS zaprzeczając jedno z nich (np. S i \bar{R}) to otrzymalibyśmy właśnie przerzutnik D wyzwalany poziomem.

Ćwiczenie 5.4

Zmontowanie, w oparciu o przerzutnik JK (7493) układu dzielącego częstotliwość przez dwa.

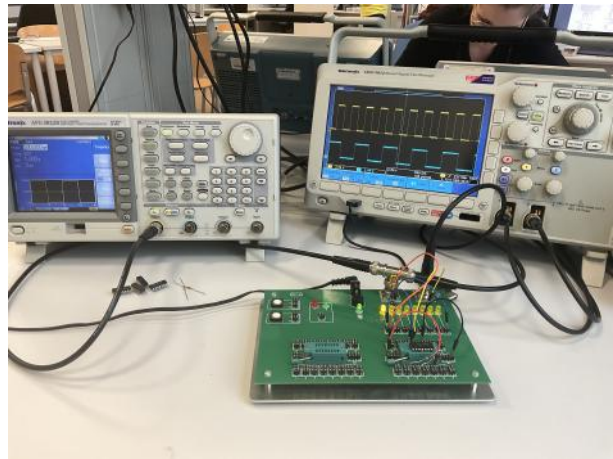
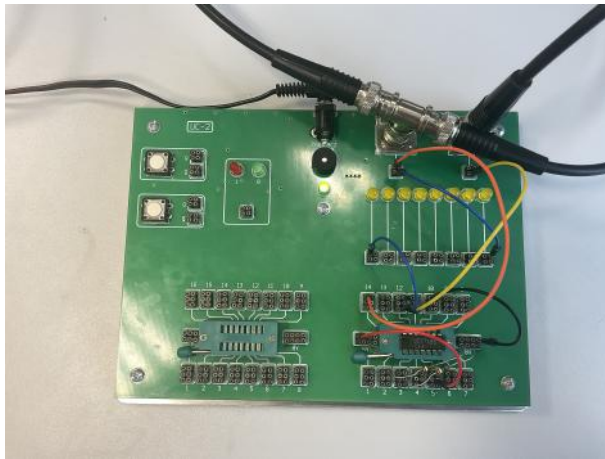
Przerzutnik JK można schematycznie przedstawić schematycznie:



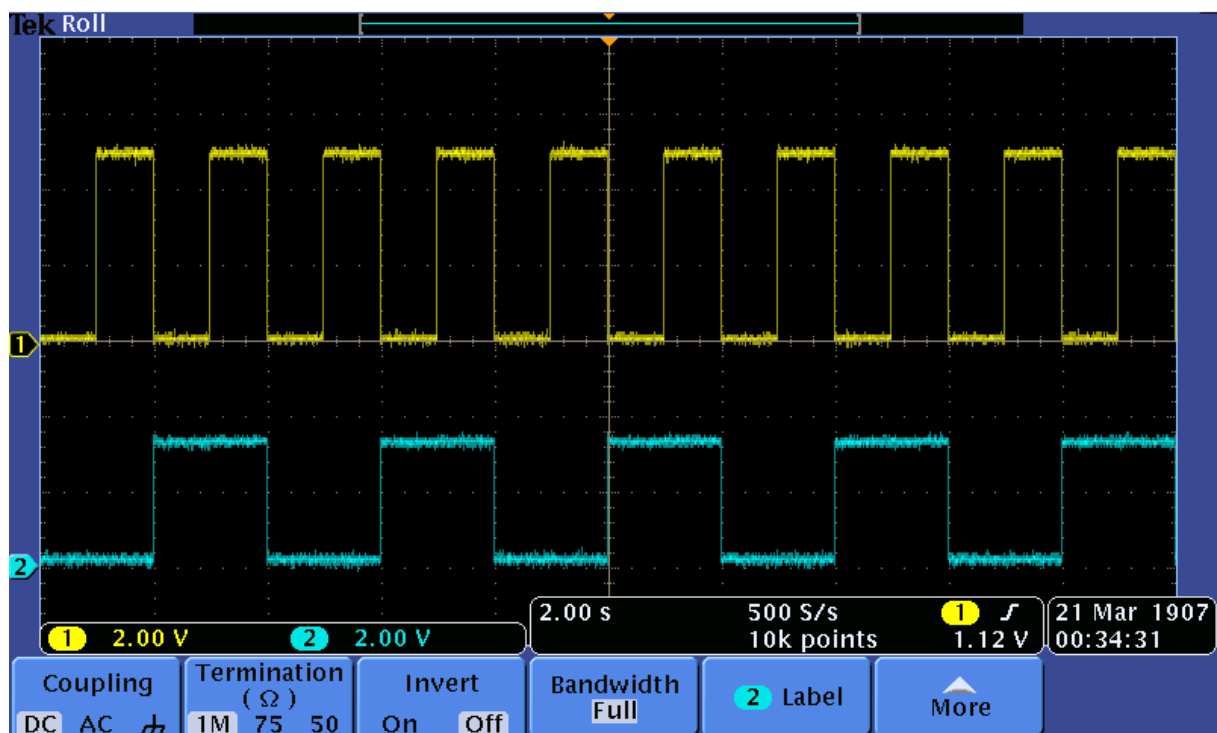
Przerzutnik ma wejścia informacyjne J i K, zegarowe CLK, wyjście Q i jego negację \overline{Q} . Posiada także asynchroniczne wejście resetujące RST.

Podanie jedynki logicznej na wejście J powoduje ustawienie przerzutnika (co skutkuje pojawieniem się stanu wysokiego na wyjściu Q). Ustawienie wejścia K w stan wysoki przestawia przerzutnik w stan niski. Jeżeli jedynka logiczna zostanie ustawiona na obydwu wejściach J i K to nastąpi zmiana stanu przerzutnika na przeciwny (czyli jeżeli układ był w stanie wysokim to przejdzie w stan niski i odwrotnie). JK jest przerzutnikiem synchronicznym, co oznacza, że reaguje na stan wejść tylko przy podaniu odpowiedniego sygnału na wejście zegarowe.

Na układach scalonych 7493 wejścia J oraz K "na sztywno" spięte są z 5 V , czyli logiczną jedynką. Sprawia to, że za każdym razem, kiedy na wejściu zegarowym zmienia się sygnał z 0 na 1 to przerzutnik zmienia stan wyjścia Q . Zamienia to nasz przerzutnik JK w dokładnie taki układ, jaki jest nam potrzebny do zadania — jeżeli Q przerzutnika JK będzie zmieniało stan za każdym razem, kiedy na zegarze będzie zbocze rosnące, znaczy, że Q będzie miało dwa razy mniejszą częstotliwość niż wejście CLK.



Zmontowałem powyższy układ i podałem na wejście zegarowe sygnał z generatora o częstotliwości 0.5 Hz . Sygnał z generatora oraz sygnał na wyjściu układu obserwowałem na oscyloskopie.



MSO3012 - 10:57:12 24.05.2023

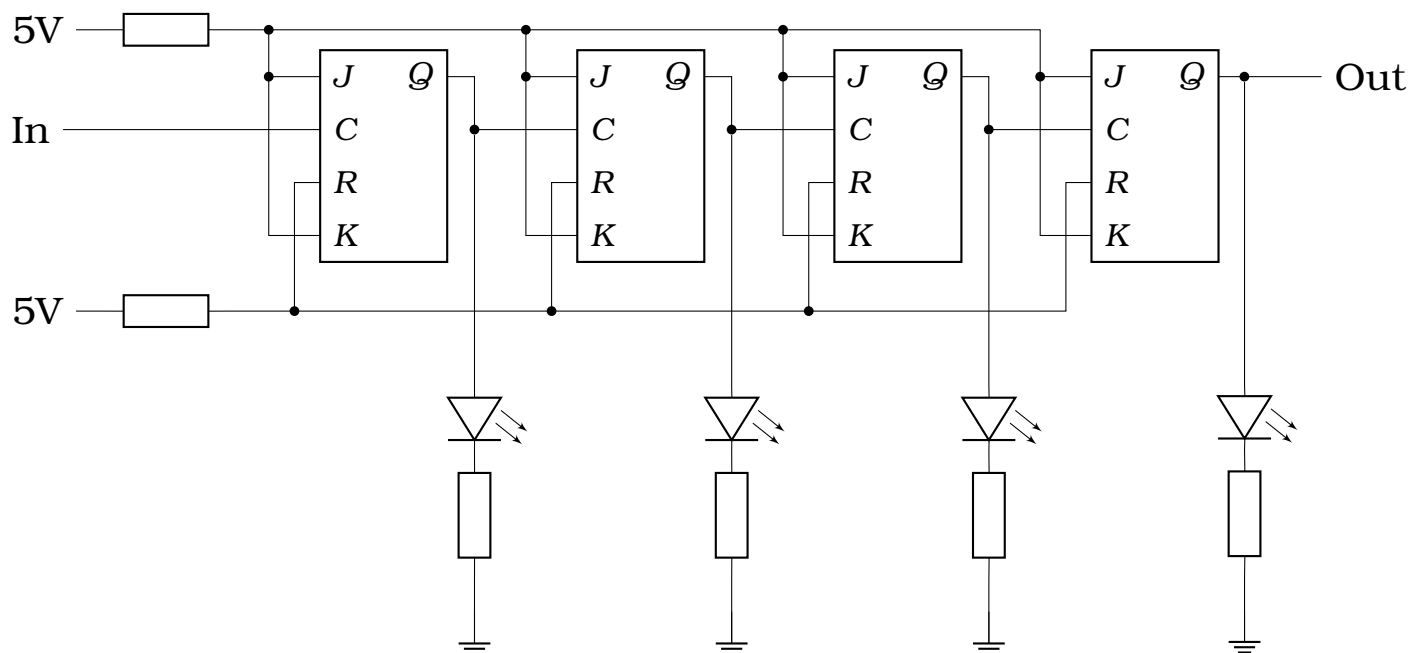
Ćwiczenie 5.5

Zbudowanie licznika modulo 16.

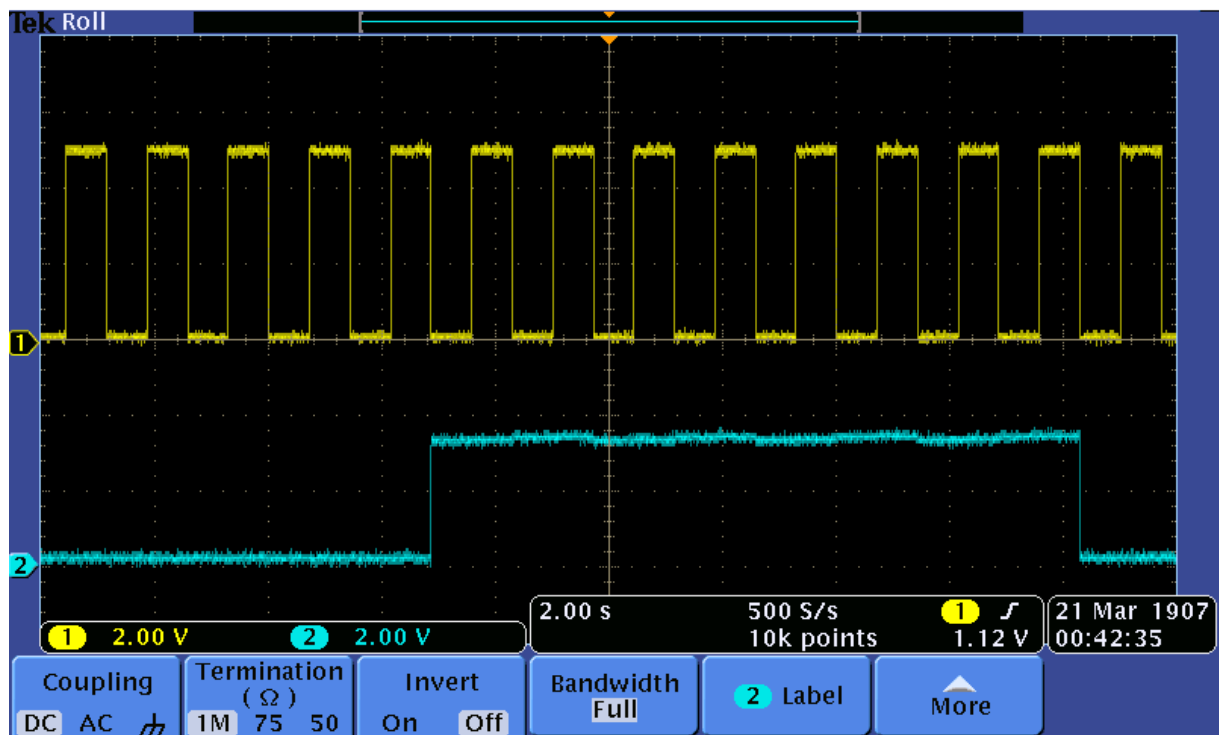
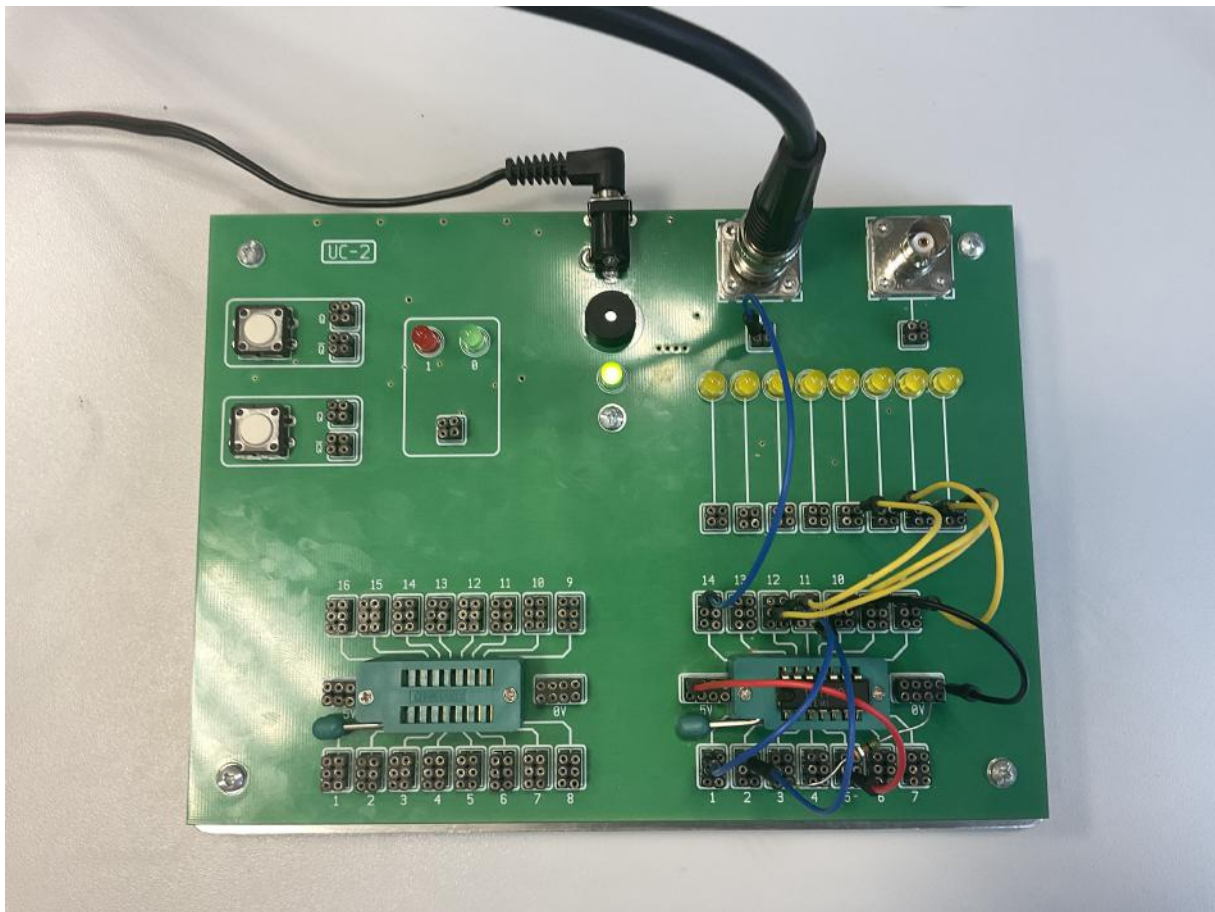
Kiedy sygnał przechodzi przez przerzutnik JK, którego wejścia J i K są spięte do jedynki częstotliwość tego sygnału dzieli się dwukrotnie. Można więc skonstruować układ, który dzieli częstotliwość sygnału czterokrotnie, jeżeli podepnimy wyjście pierwszego przerzutnika JK pod wejście zegarowe drugiego. Albo układ który dzieli częstotliwość sygnału ośmiokrotnie, dodając jeszcze jeden przerzutnik JK. Albo szesnastokrotnie, dodając kolejny przerzutnik.

Na układzie scalonym 7493 znajdują się 4 przerzutniki JK, połączone ze sobą w sprytny sposób: oprócz tego, że wejścia każdego z nich są spięte z logiczną jedynką to wyjście pierwszego jest połączone z wejściem zegarowym drugiego, wyjście drugiego z wejściem zegarowym trzeciego, itd.

Możemy z tego oczywiście skorzystać i zbudować dzielnik częstotliwości przez 16 posługując się układem 7493. Dodając diodę świecącą na wyjściu każdego z przerzutników osiągniemy licznik binarny liczący od 0 do 15. Schemat takiego licznika będzie wyglądał następująco:



Zbudowałem powyższy układ na płytce UC-2. Zbadałem oscyloskopem przebiegi na wejściu i wyjściu licznika. Filmik z działania licznika załączam w pliku Sprawozdanie-cwiczenie5-Jan-Kwinta-Zalacznik3.zip.



MSO3012 - 11:05:14 24.05.2023

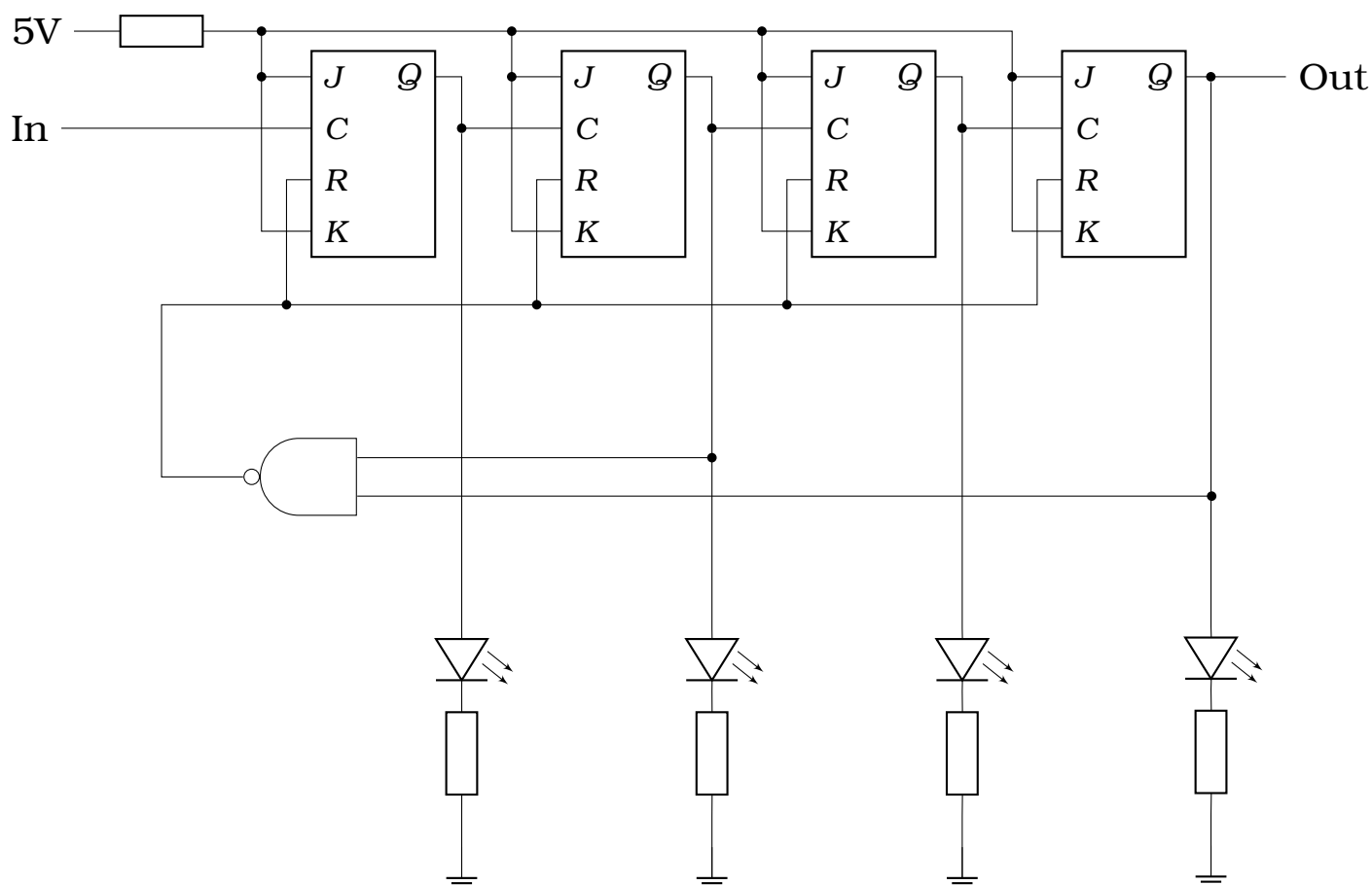
Ćwiczenie 5.6

Zbudowanie licznika modulo 10.

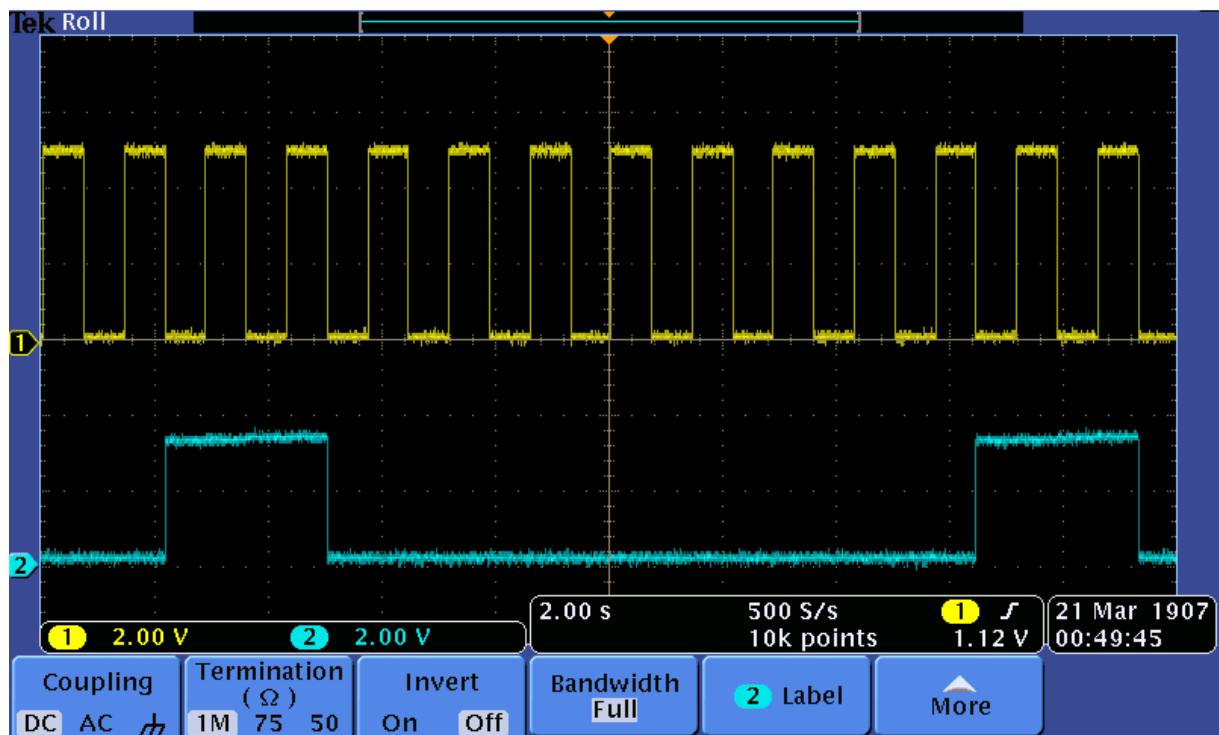
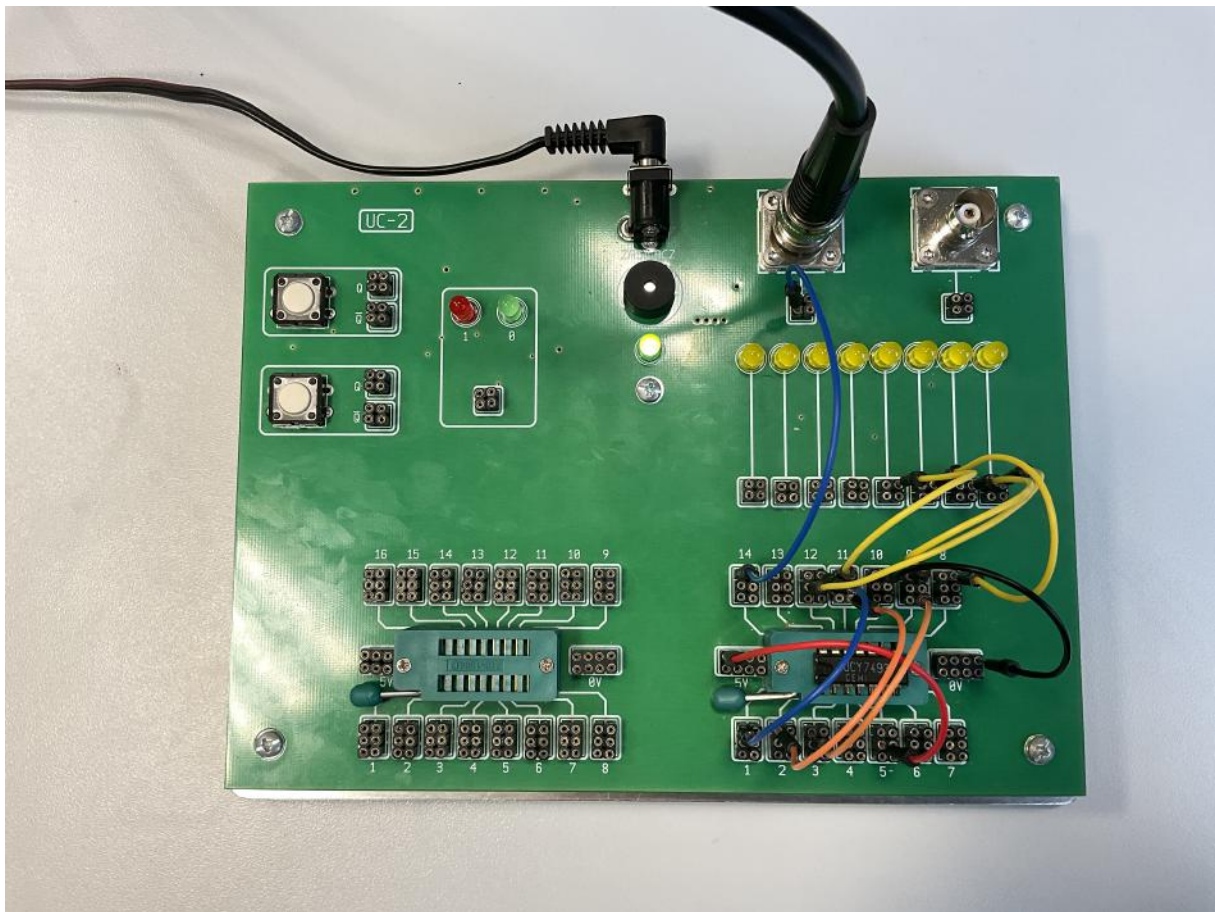
Moglibyśmy skorzystać ze zbudowanego w poprzednim zadaniu licznika modulo 16 i zerować licznik wtedy, gdy wyjścia przerzutników będą oznaczały stan "10". Licznik osiąga stan 10 kiedy przerzutniki czwarty i drugi (licząc od lewej) mają swoje wyjścia Q ustawione na logiczne 1.

Na nasze szczęście, na układzie scalonym 7493 znajduje się, oprócz 4 przerzutników JK jeszcze jedna bramka NAND. Pozwoli nam ona podać sygnał resetujący (logiczne zero na wejściu RST) do wszystkich czterech przerzutników wtedy, gdy licznik osiągnie stan "10". Nandując wyjścia czwartego i drugiego przerzutnika otrzymamy zero logiczne tylko, jeśli obydwa będą wysokie.

Wtedy licznik będzie wyliczał wszystkie liczby od 0 do 9. Schemat takiego układu wygląda następująco:



Zbudowałem powyższy układ na płytce UC-2. Zbadałem oscyloskopem przebiegi na wejściu i wyjściu licznika. Filmik z działania licznika załączam w pliku Sprawozdanie-cwiczenie5-Jan-Kwinta-Zalacznik3.zip.

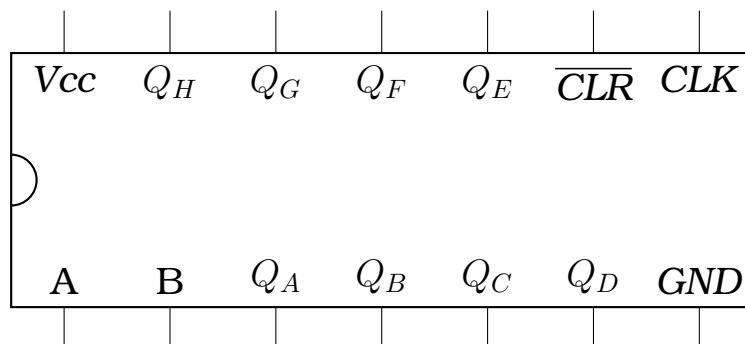


MSO3012 - 11:12:24 24.05.2023

Ćwiczenie 5.7

Sprawdzenie działania rejestrów 74164 i 74165.

Układ scalony ośmiobitowego rejestru 74164 ma następujący schemat:



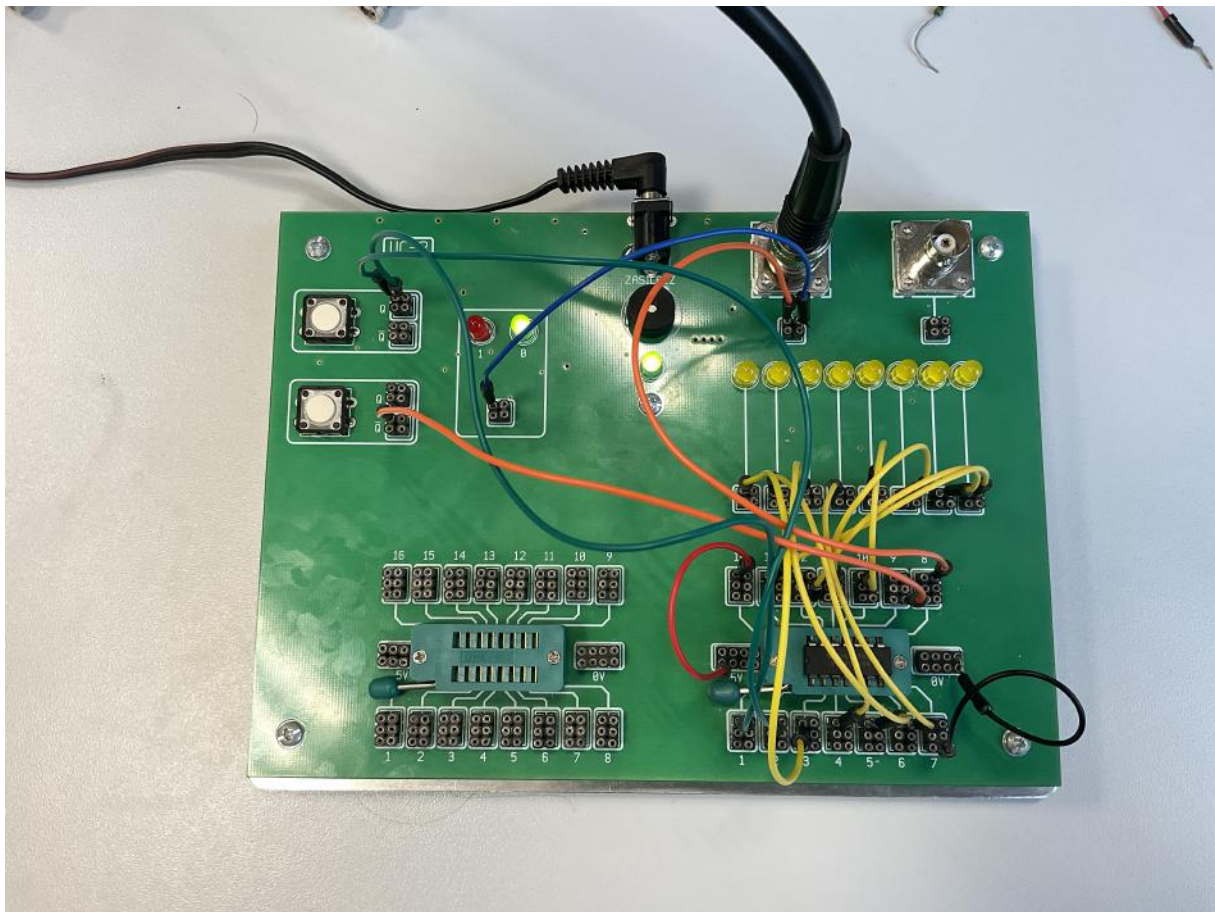
Rejestr ten posiada 4 wejścia: zegarowe (CLK) i czyszczące (CLR) oraz wejścia danych A i B. Układ posiada także osiem wyjść: Q_A , Q_B , ..., Q_H , które reprezentują osiem komórek - 8 bitów pamięci rejestru.

Z każdym cyklem zegara, podczas zbocza wznoszącego zapisywana jest nowa wartość do komórki A rejestru. Jeżeli na obydwu wejściach A i B podane są logiczne jedynki do komórki A zostaje wpisana jedynka. W przeciwnym wypadku A przyjmuje wartość 0. Wartości pozostałych komórek przesuwane są sekwencyjnie, to znaczy, że przy zboczu wznoszącym zegara komórka H przyjmuje wartość, którą wcześniej miała komórka G, komórka G przyjmuje wartość, którą wcześniej miała komórka F i tak dalej. Z każdym cyklem zegara wartość, która była w komórce H jest tracona.

Niezależnie od stanu wejść CLK, A oraz B podanie sygnału niskiego na wejście CLR powoduje wyczyszczenie pamięci, to jest nadanie wszystkim komórkom A, B, ..., H wartości logicznego zera.

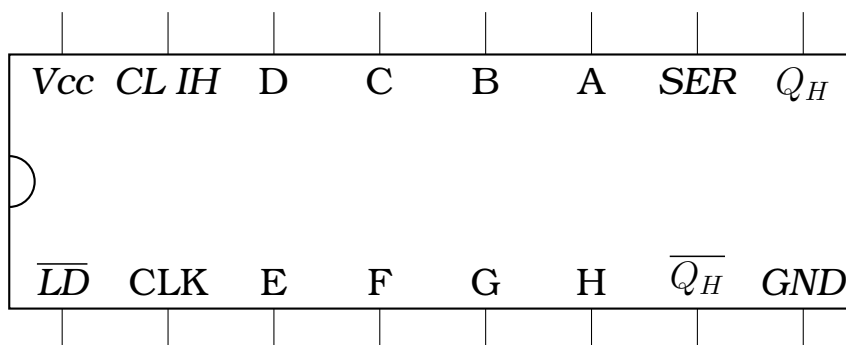
Zmontowałem następujący układ: sygnał zegarowy z generatora podałem na wejście CLK układu oraz do próbnika stanów logicznych, by można było widzieć stan zegara. Wyjście Q pierwszego impulsatora podałem na obydwa wejścia A i B układu 74164. Wyjście \overline{Q} drugiego impulsatora podałem na wejście CLR rejestru.

Za pomocą pierwszego impulsatora jestem w stanie zapisywać bity do pamięci rejestru, a kiedy nacisnę przycisk drugiego impulsatora następuje wyczyszczenie rejestru.



Filmik pokazujący działanie układu zamieszczam w załączniku
Sprawozdanie-cwiczenie5-Jan-Kwinta-zalacznik4.zip.

Układ scalony ośmiobitowego rejestru 74165 ma następujący schemat:



Jest to układ całkiem podobny do poprzedniego. Główną różnicą jest to, że 74165 posiada wejścia szeregowo: oznaczone są jako A, B, ... H. W dowolnej chwili pracy rejestru możemy podać na wejścia szeregowo osiem bitów i gdy podamy sygnał niski na wejście *load* (LD) to zostaną one zapisane w pamięci rejestru.

Dostępne wyjścia to jedynie wyjście Q_H oraz komplementarne $\overline{Q_H}$. Oznacza to, że w danej chwili widzimy jedynie wierzch stosu pamięci tego układu. Podobnie jak w 74164 bity przesuwają się sekwencyjnie od A do H. W każdym cyklu zegara możemy zapisać nowy bit do komórki A podając go na wejście *serial* (SER).

Wejście *clock inhibit* (CL IH) służy do wstrzymywania pracy układu — kiedy zostanie podany na nie sygnał wysoki stan komórek pamięci pozostaje niezmienny w kolejnych cyklach zegara.

Na laboratorium udało mi się podłączyć układ i operować wejściem szeregowym, wejściem blokady zegara (CL IH) i obserwować wyjście Q_H . Jednakże, ze względu na małą ilość czasu nie zrealizowałem wpisania 8-bitowego słowa przez wejścia A, B, ... H, ani nie połączyłem rejestrów 74164 i 74165 ze sobą. Także ze względu na czas nie udało mi się uwiecznić mojego układu z chipem 74165 na zdjęciach ani na filmiku.

Notatki z zeszytu laboratoryjnego

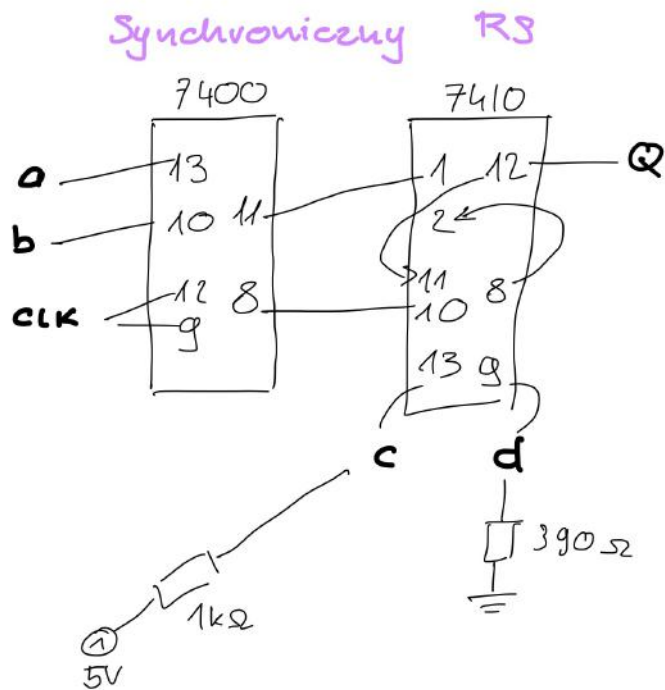
Poniżej załączam notatki z zeszytu laboratoryjnego.

ĆWICZENIE 5

24 MAY

Stacja robocza
nr 1

ZADANIE 1



TABELKA 1

c = 1

d = 1

CLK = 1			CLK = 0		
a	b	Q	a	b	Q
0	0	pamięta	0	0	pamięta
0	1	0	0	1	
1	0	1	1	0	stan
1	1	—	1	1	

TABELKA 2

$c = 0$

$d = 1$

CLK = 1		
a	b	Q
0	0	1
0	1	1
1	0	1
1	1	1

CLK = 0

a	b	Q
0	0	1
0	1	1
1	0	1
1	1	1

TABELKA 3

$c = 1$

$d = 0$

CLK = 1		
a	b	Q
0	0	0
0	1	0
1	0	1
1	1	1

CLK = 0

a	b	Q
0	0	0
0	1	0
1	0	0
1	1	0

TABELKA 4

$c = 0$

$d = 0$

CLK = 1		
a	b	Q
0	0	1
0	1	1
1	0	1
1	1	1

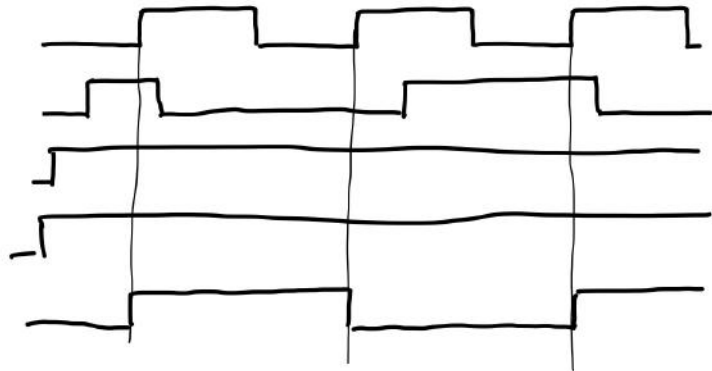
CLK = 0

a	b	Q
0	0	1
0	1	1
1	0	1
1	1	1

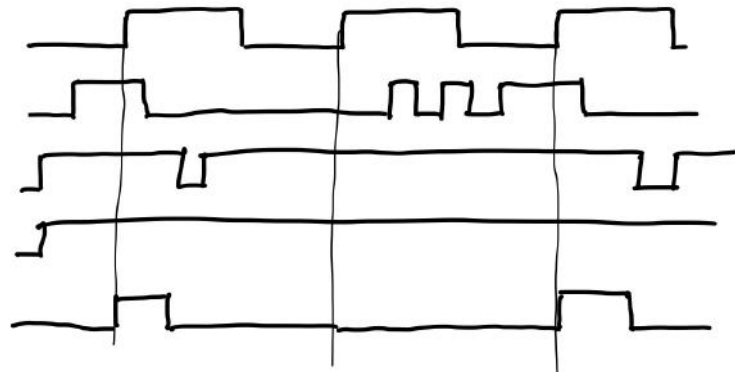
ZADANIE 2.

Przerzutnik D

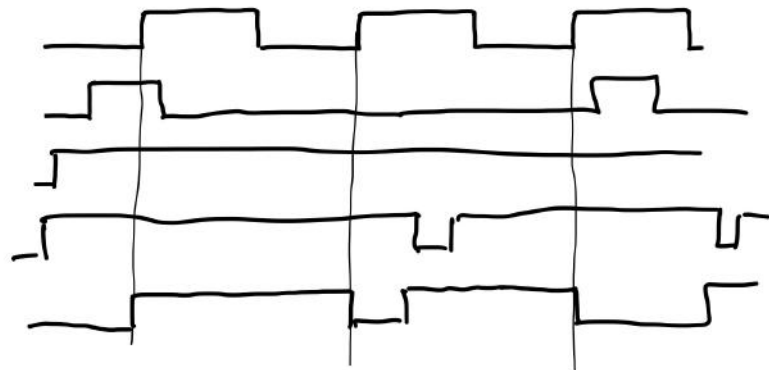
CLK
 D
 CLR
 PRE
 Q



CLK
 D
 CLR
 PRE
 Q



CLK
 D
 CLR
 PRE
 Q



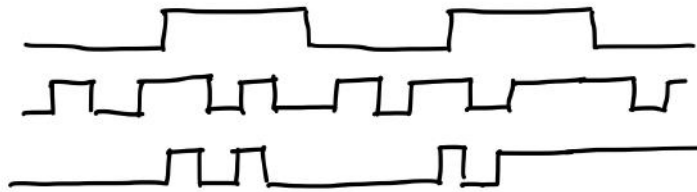
ZADANIE 3.

D- LATCH

C

D

Q



TABELA

STANÓW

C	D	Q
0	0	pozostaje
0	1	pozostaje
1	0	0
1	1	1