

## Układy sekwencyjne – licznik

### 1. Cel ćwiczenia

Zapoznanie się ze sposobem projektowania układów sekwencyjnych synchronicznych.  
Badanie liczników równoległych binarnych.

### 2. Podział liczników

#### 2.1. Liczniki szeregowo i równoległe

#### 2.2. Liczniki jednokierunkowe i dwukierunkowe (rewersyjne)

#### 2.3. Konwencja oznaczania wyprowadzeń

- LOAD – loadable
- DIR, UP, DOWN – bidirectional
- CE – clock enable
- CLK, C – clock
- RST, CLR – (a)synchronous reset
- Qx – data output (x – numer pozycji, bitu)
- Dx – data input

### 3. Liczniki równoległe

#### 3.1. Projektowanie licznika „modulo 3”

##### 3.1.1.przerzutnik typu D z wejściem CE;

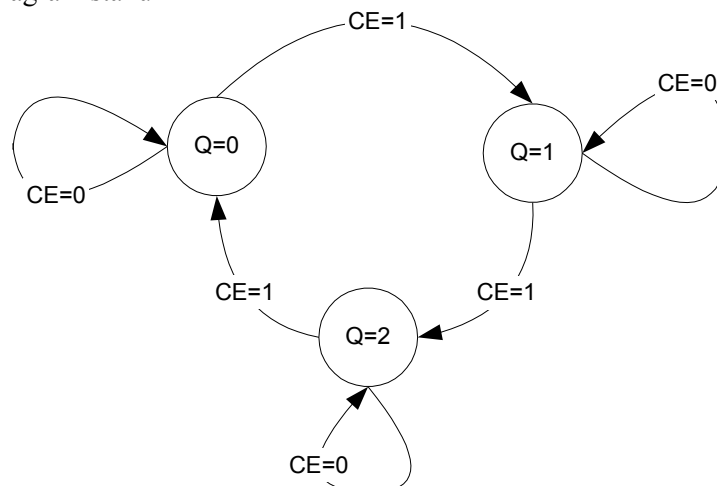
| CE,D \ Q | 0 0 | 0 1 | 1 1 | 1 0 |
|----------|-----|-----|-----|-----|
| 0        | 0   | 0   | 1   | 0   |
| 1        | 1   | 1   | 1   | 0   |

Q'

$$Q' = \neg CE Q + CE D$$

##### 3.1.2.projekt licznika „modulo 3” z wejściem CE (wykorzystuje przerzutniki typu DCE);

- diagram stanu



- tablica przejść i wyjść

| A | CE |   |
|---|----|---|
|   | 0  | 1 |
| 0 | 0  | 1 |
| 1 | 1  | 2 |
| 2 | 2  | 0 |

| kodowanie stanów | A1 A0 |
|------------------|-------|
| 0                | 0 0   |
| 1                | 0 1   |
| 2                | 1 0   |

- synteza układu

| A1 A0 \ CE | 0 | 1 |
|------------|---|---|
|            |   |   |
| 00         | 0 | 1 |
| 01         | 1 | 0 |
| 11         | - | - |
| 10         | 0 | 0 |

A0'

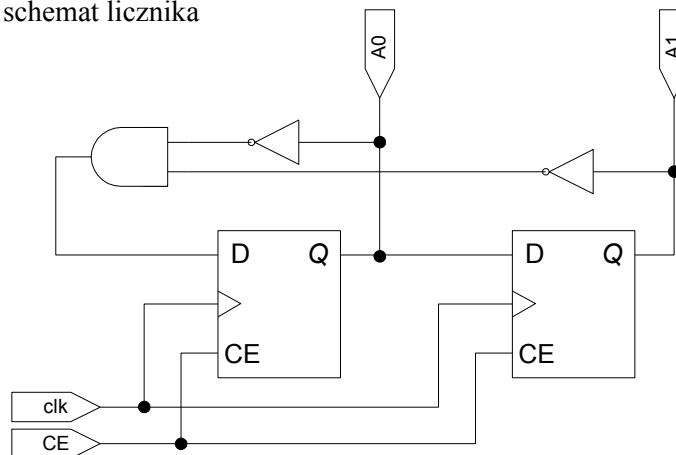
$$A0' = \neg CE A0 + CE \neg A1 \neg A0$$

| A1 A0 \ CE | 0 | 1 |
|------------|---|---|
|            |   |   |
| 00         | 0 | 0 |
| 01         | 0 | 1 |
| 11         | - | - |
| 10         | 1 | 0 |

A1'

$$A1' = \neg CE A1 + CE A0$$

- schemat licznika



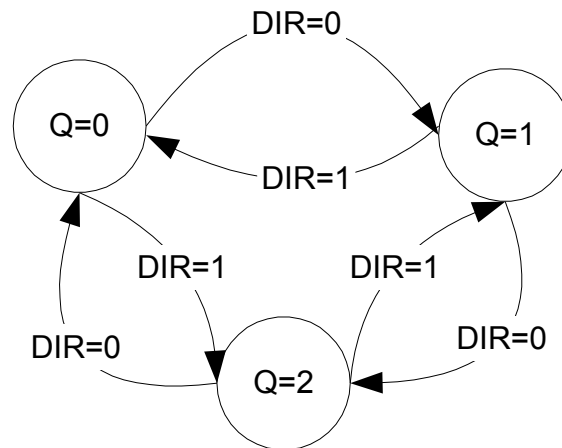
#### 4. Liczniki rewersyjne

Licznik rewersyjny jest układem sekwencyjnym mającym możliwość zliczania w przód lub w tył, w zależności od stanu wejścia sterującego oznaczanego najczęściej DIR lub UP.

##### 4.1. Projekt licznika rewersyjnego „modulo 3”

(wejście sterujące DIR = '0' -> zliczanie w przód, DIR = '1' -> zliczanie w tył);

- diagram stanu



- tablica przejść i wyjść

| Q | DIR |   |
|---|-----|---|
|   | 0   | 1 |
| 0 | 1   | 2 |
| 1 | 2   | 0 |
| 2 | 0   | 1 |

| kodowanie stanów | Q1 Q0 |
|------------------|-------|
| 0                | 0 0   |
| 1                | 0 1   |
| 2                | 1 0   |

- synteza układu

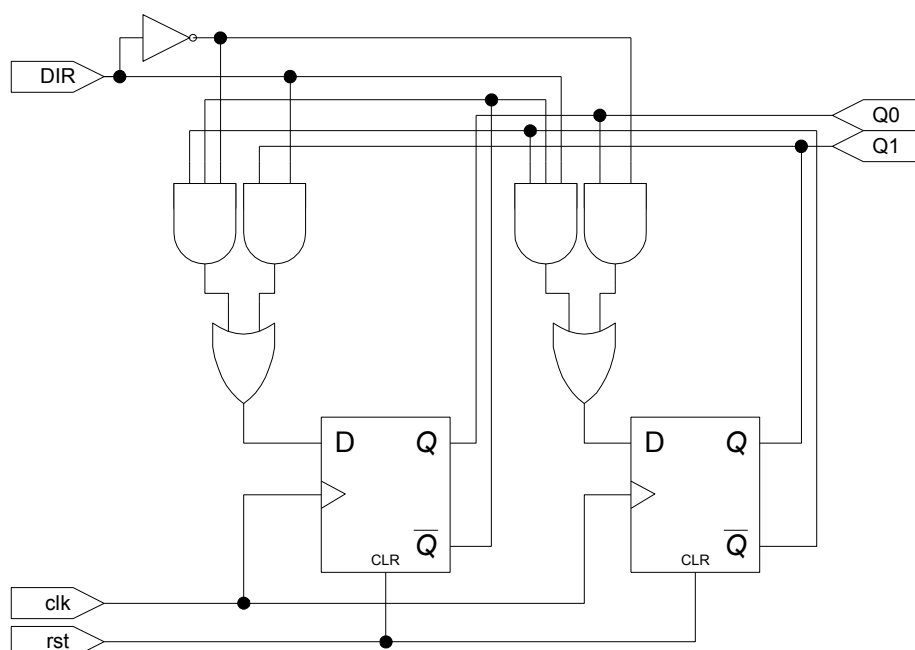
| Q1 Q0 \ DIR | 0 | 1 |
|-------------|---|---|
|             |   |   |
| 00          | 1 | 0 |
| 01          | 0 | 0 |
| 11          | - | - |
| 10          | 0 | 1 |

$$Q0' = \neg \text{DIR} \neg Q1 \neg Q0 + \text{DIR} Q1$$

| Q1 Q0 \ DIR | 0 | 1 |
|-------------|---|---|
|             |   |   |
| 00          | 0 | 1 |
| 01          | 1 | 0 |
| 11          | - | - |
| 10          | 0 | 0 |

$$Q1' = \neg \text{DIR} Q0 + \text{DIR} \neg Q1 \neg Q0$$

- schemat licznika



## 5. Scalone układy licznikowe

W postaci modułów scalonych CMOS najczęściej występują liczniki dodające lub rewersyjne. Ze względu na realizowane funkcje liczniki scalone dzielone są na binarne (zliczające w kodzie naturalnym binarnym NKB) i dziesiętne (zliczające w kodzie BCD).

Jeśli typowe pojemności liczników nie odpowiadają wymaganiom projektanta, można je zmienić za pomocą układów sprzężenia zwrotnego wymuszających określoną zmianę zawartości licznika. W tym celu wykorzystuje się wejście zerujące licznika (RST, CLR).

### 5.1. Licznik binarny

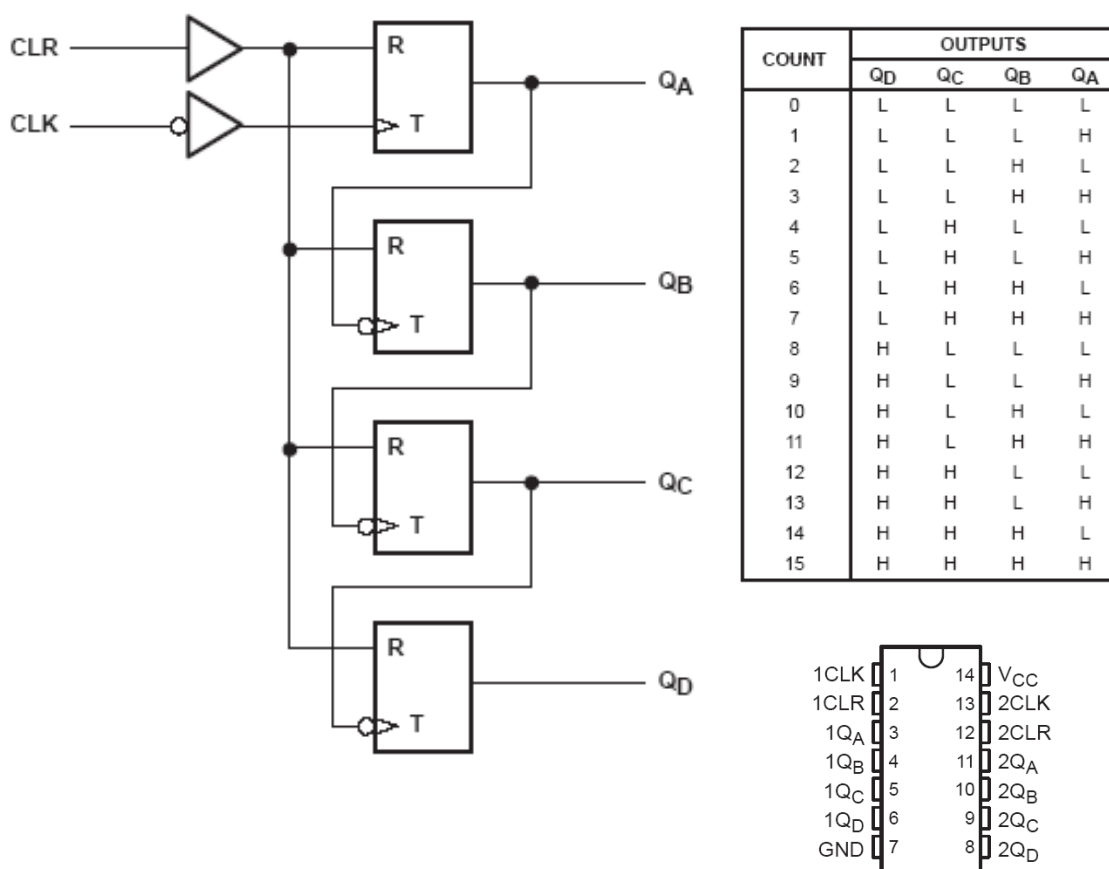
Do realizacji scalonych liczników binarnych bardzo często wykorzystuje się przerzutniki typu T.

5.1.1. działanie przerzutnika typu T;

| Q \ T | 0 | 1 |
|-------|---|---|
| 0     | 0 | 1 |
| 1     | 1 | 0 |

$$Q' = \neg T Q + T \neg Q = T \oplus Q$$

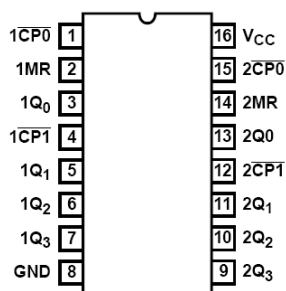
5.1.2. przykład realizacji licznika binarnego: **74HC393**



## 5.2. Licznik dziesiętny (BCD)

Jest to licznik „modulo 10”, zliczający w kodzie BCD. Na jego wyjściu pojawiają się kolejno stany od 0 do 9. Najczęściej realizowany jako szeregowe połączenie dwóch segmentów o pojemnościach 2 i 5.

### 5.2.1. przykład realizacji licznika dziesiętnego: 74HC390



TRUTH TABLE

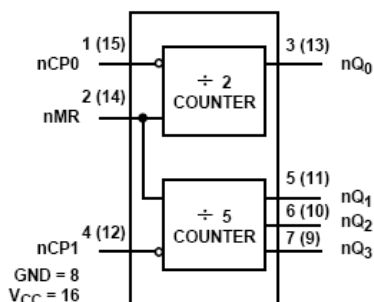
| INPUTS |    | ACTION     |
|--------|----|------------|
| CP     | MR |            |
| ↑      | L  | No Change  |
| ↓      | L  | Count      |
| X      | H  | All Qs Low |

H = High Voltage Level, L = Low Voltage Level, X = Don't Care,  
 ↑ = Transition from Low to High Level, ↓ = Transition from High to Low.

BCD COUNT SEQUENCE FOR 1/2 THE 390

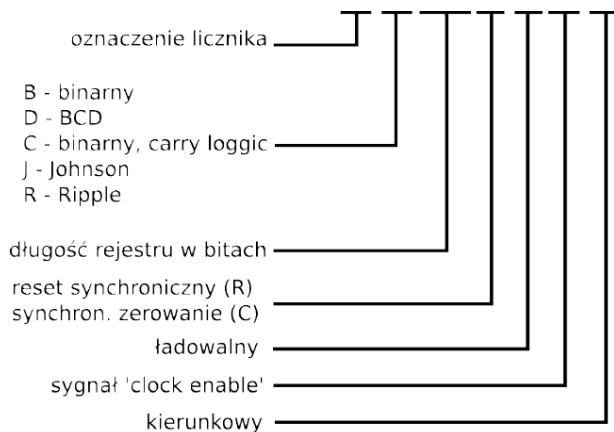
| COUNT | OUTPUTS |    |    |    |
|-------|---------|----|----|----|
|       | Q0      | Q1 | Q2 | Q3 |
| 0     | L       | L  | L  | L  |
| 1     | H       | L  | L  | L  |
| 2     | L       | H  | L  | L  |
| 3     | H       | H  | L  | L  |
| 4     | L       | L  | H  | L  |
| 5     | H       | L  | H  | L  |
| 6     | L       | H  | H  | L  |
| 7     | H       | H  | H  | L  |
| 8     | L       | L  | L  | H  |
| 9     | H       | L  | L  | H  |

Output nQ0 connected to nCP1 with counter input on nCP0.



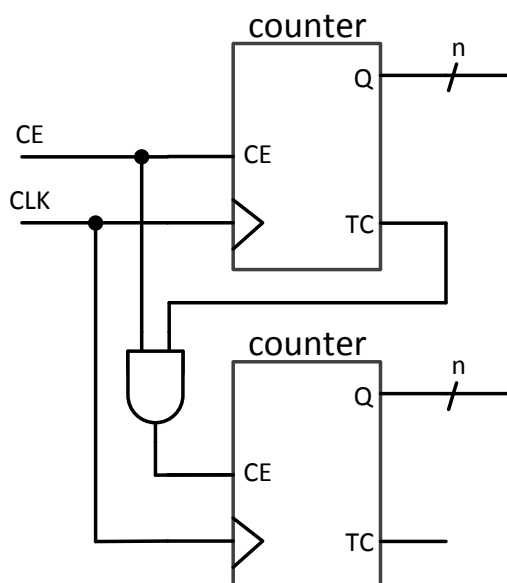
## 6. Oznaczenie liczników stosowanych w oprogramowaniu ISE (układy z biblioteki producenta)

### CB16CLED



## 7. Kaskadowe łączenie liczników

W celu rozszerzenia zakresu zliczania można łączyć wiele modułów licznikowych w kaskadę. Sygnał końca zliczania (TC – *Terminal Count*) poprzedniego modułu powoduje odblokowanie i inkrementację stanu w kolejnym bloku licznika. W elektronice cyfrowej, szczególnie przy implementacji w urządzeniach rekonfigurowalnych, sygnał TC młodsze licznika podawany jest na wejście *clock enable* starszego, odblokowując go na okres jednego taktu sygnału zegarowego. Unikamy w ten sposób wprowadzania dodatkowej logiki w linię zegarową, co może zwiększać przesunięcia fazy między zboczami zegara i powodować powstawanie tzw. ‘glitchy’(krótkie, niechciane impulsy).



Schemat przedstawia typowe połączenie kaskadowe licznika synchronicznego – należy zwrócić uwagę na konieczność powiązania kolejnych sygnałów TC z wejściem CE całego licznika.

## 8. Zadania:

- Zaprojektuj liczniki **L1**, **L2** korzystając z przerzutników typu D z wejściem *clock enable*, funkcje przejść w projektowanych licznikach zrealizuj z dowolnych bramek logicznych. Liczniki muszą posiadać wyjście końca zliczania (**TC**).
- Połącz kaskadowo zbudowane w pkt.a liczniki tak, aby L1 był młodszy a L2 starszym licznikiem w kaskadzie. Wykonaj symulację behawioralną tego układu i wyniki zamieść w karcie projektu.
- Przygotuj implementację układu licznika z pkt.b sterującego wyświetlaczem 7-segmentowym z płyty prototypowej Nexys2 (można wykorzystać dowolny ze sterowników wyświetlacza z serwera lub zaprojektować własny moduł sterujący do wyświetlacza multipleksowanego z płyty Nexys2). Na wejście zegarowe podaj sygnał wolnozmienny z dzielnika zegara (np. `clk_gen_50`).
- Zadanie dodatkowe: w celu ułatwienia testowania na pozostałych dwóch segmentach wyświetlacza LED wyświetlaj numery stanów odpowiedniego segmentu licznika (zadanie można zrealizować używając gotowych liczników binarnych z biblioteki producenta).

W czasie zajęć uruchom i zaprezentuj działanie jednego z zaprojektowanych układów (układ wybrany przez prowadzącego).

*W opracowaniu zamieszczono fragmenty dokumentacji układu SN74HC390 i 74HC393 firmy Texas Instruments Incorporated.*