

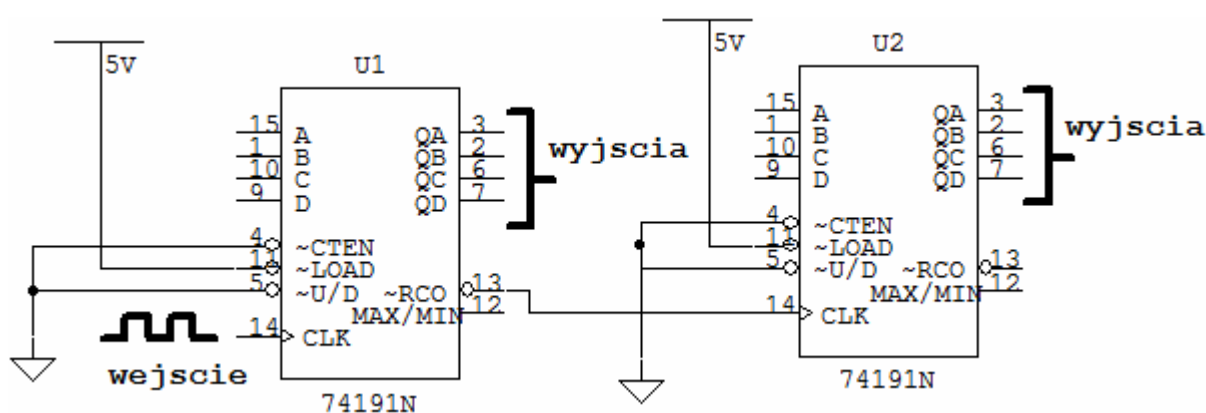
### DODATEK III - Łączenie liczników scalonych

Skoro wiemy (z instrukcji do właściwego ćwiczenia) jak użyć licznika scalonego 74191 do liczenia w dowolnym zakresie, w górę lub w dół, to warto teraz o niewykorzystywanych wcześniej pinach MAX/MIN i ~RCO.

Pin MAX/MIN przechodzi w stan wysoki w momencie gdy licznik jest w stanie 1111, czyli daje on nam informację że licznik doliczył do końca swojego zakresu.

Z kolei pin ~RCO („ripple clock output”) działa odwrotnie od pinu MAX/MIN, czyli gdy licznik doliczył do końca przechodzi on w stan niski (logiczne zero), cały pozostały czas jest w stanie wysokim. **Pin ~RCO jest używany gdy chcemy liczyć dalej niż do 15.** W jaki sposób? Otóż podłączamy go do wejścia CLK następnego licznika i w ten sposób zamiast liczyć na 4 bitach, liczymy na 8, a zatem zamiast liczyć maksymalnie do 15, możemy liczyć do 255 ( $255 = 2^8 - 1$ ).

Układ dwóch liczników skonfigurowanych do liczenia w zakresie 0-255 pokazano na rysunku poniżej:

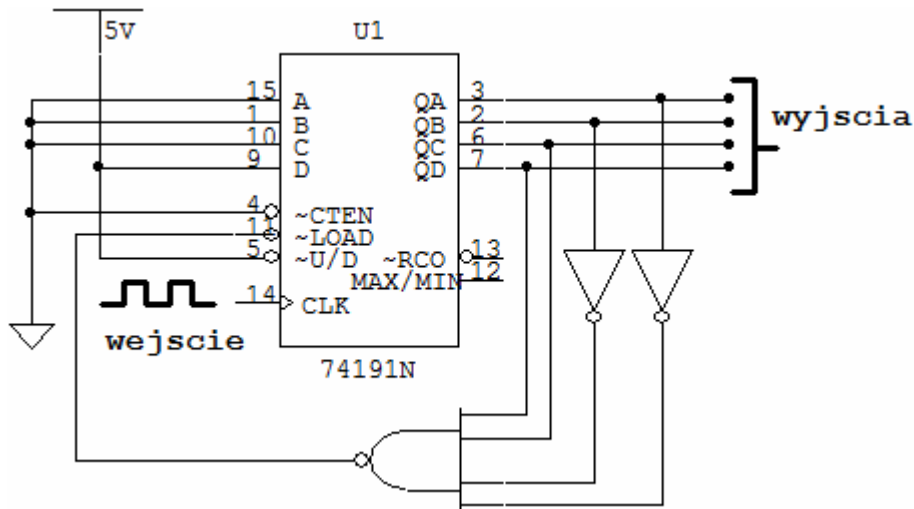


Wyjść układu traktowanego jako całość jest teraz 8, przy czym wyjścia z licznika oznaczonego jako U2 są bitami bardziej znaczącymi ( $Q_D = 2^7$ ,  $Q_C = 2^6$ ,  $Q_B = 2^5$  i  $Q_A = 2^4$ ), natomiast wyjścia z licznika oznaczonego jako U1 to bity mniej znaczące ( $Q_D = 2^3$ ,  $Q_C = 2^2$ ,  $Q_B = 2^1$  i  $Q_A = 2^0$ ). Traktowane razem dają słowo 8-bitowe, którego maksymalna wartość (11111111) to 255. Jak łatwo się domyślić, po dołączeniu kolejnego, czyli trzeciego licznika pojemność układu wzrosłaby do  $2^{12} - 1 = 4095$  i byłby on układem 12-wyjściowym.

Oczywiście po połączeniu liczników cały czas obowiązują te same zasady skracania cyklu co poprzednio. Czyli taki układ jak powyżej możemy spokojnie skonfigurować do liczenia w zakresie np. od 34 do 134, bądź innego, w górę, lub w dół. Różnica jest tylko taka, że zerujemy oba liczniki jednocześnie (czyli wyjście bramki NAND wykrywającej odpowiedni stan podłączone jest do pinów ~LOAD obu liczników jednocześnie) na podstawie właściwego stanu na wyjściach obu liczników jednocześnie. Analogicznie, stan początkowy cyklu jest teraz słowem 8-bitowym i musimy go narzucić wykorzystując wejścia D, C, B i A obu liczników.

W celu utrwalenia informacji o licznikach scalonych (zarówno pojedynczych, jak i łączonych ze sobą) warto przeanalizować poniższe układy.

### PRZYKŁAD 1.



**Dany jest licznik jak na rysunku. W jakim zakresie on liczy?**

**Analiza przebiegów mniej więcej tak:**

Mamy układ. Licznik jest jeden, czyli od razu wiemy, że zliczy on maksymalnie do 15.

Widzimy że jest w nim zerowanie, bo na wejście  $\sim$ LOAD podane jest wyjście bramki NAND.

Widzimy też, że układ nie liczy od 0, bo mamy skonfigurowane (= coś jest do nich podłączone) wejścia DCBA. Powiedzmy, że od tego zaczniemy. Na wejścia DCBA podano stan 1000, czyli **dziesiętnie 8, więc od tej liczby zacznie liczyć ten układ.**

Idziemy dalej. Patrzymy na pin  $\sim$ U/D, jest na niego podana logiczna jedynka. **Aha, zatem licznik liczy w dół.**

Zostało nam sprawdzić „do ilu” liczy ten układ, czyli jaką liczbę wykrywa bramka NAND. Patrzymy na bramkę NAND – na jej wejścia podano bezpośrednio wyjścia  $Q_D$  i  $Q_C$ , czyli tu wykrywamy jedynki. Natomiast wyjścia  $Q_B$  i  $Q_A$  podano przez bramkę NOT zatem tu wykrywamy 0. Dobra, składamy z tego słowo i wychodzi nam:  $Q_D Q_C Q_B Q_A = 1100$  **czyli dziesiętnie 12.**

Jeżeli połączymy pozyskane wyżej informacje, to wynika z nich, że **licznik z rysunku zlicza od 8 do 13 w dół.** Czemu do 13, a nie do 12? Bo analogicznie jak przy przykładzie pokazanym w instrukcji do ćwiczenia, wykrywana jest zawsze liczba następująca po tej, do której planowane jest zliczanie. Dlatego, gdy tam zliczaliśmy do 9 (w górę), to wykrywaliśmy 10, skoro tutaj wykrywamy 12, to zliczanie jest do 13 (o jeden więcej, a nie o jeden mniej, bo liczymy w dół, nie w górę).

Być może zakres taki nie brzmi do końca logicznie, ale spójrzmy jak dokładnie taki układ liczy (przy założeniu, że wszedł już on w pętlę, bo pierwszy cykl ZAWSZE zacznie się od 0).

*Cykl liczenia zaczyna się od 8.*

*Liczymy w dół, zatem dalej, po 8: 7, 6, 5, 4, 3, 2, 1, 0...*

*Co po zerze? Licznik liczy w „kółko” a jest 4-bitowy, zatem po 0 jest: 15, 14, 13...*

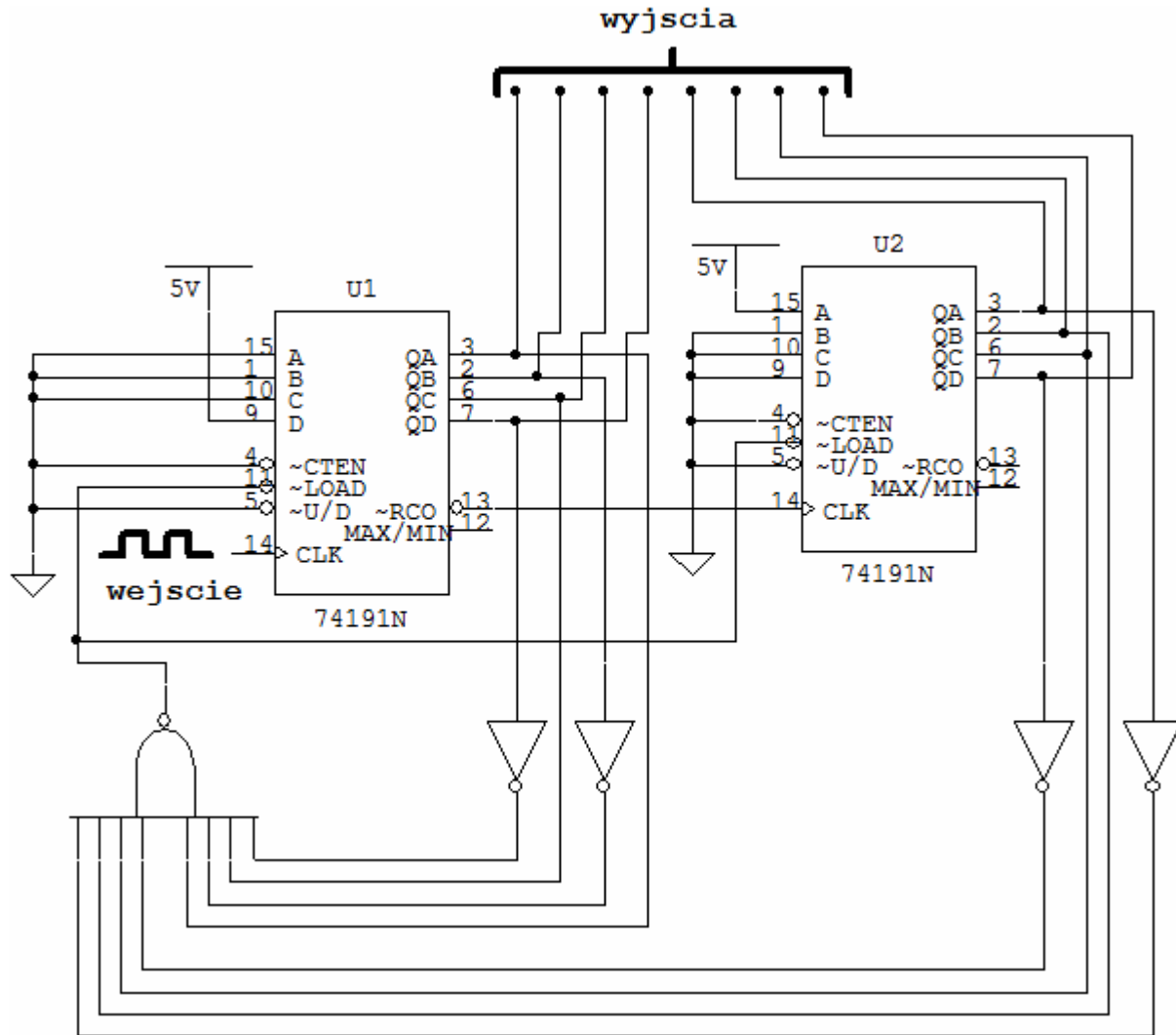
*Co po 13? Po 13 powinno być 12, ale 12, jak wcześniej sprawdziliśmy, jest wykrywana bramką NAND i zeruje układ.*

*Zatem po 13 jest 8 (bo przecież od 8 układ zaczyna).*

*Podsumujmy zatem, licznik w tym układzie liczy: 8, 7, 6, 5, 4, 3, 2, 1, 0, 15, 14, 13, 8, 7, 6... i tak w kółko...*

Kolejnym zadaniem, tym razem do samodzielnej analizy, niech będzie układ pokazany na rysunku poniżej.

Pytanie jest podobne - jak liczy taki układ?



**Jeśli z analizy układu wyszło że od 24 do 100 - gratulacje, umiesz analizować działanie liczników scalonych.**

Ostatnią rzeczą, o której warto wspomnieć w kontekście liczników scalonych jest ta, że w przypadku gdyby zaistniała potrzeba wykorzystania innego licznika, to na podstawie noty katalogowej i logicznego myślenia da się wykonać podobną analizę ich działania, jak też skonfigurować go do działania w pożądanym sposób.