

# Układy cyfrowe – 2 zajęcia on-line 18.04.2020

Prowadzący: Andrzej Kazimierczyk

Materiały i zagadnienia do opracowania w ramach zajęć on-line.

Uwaga! Wszystkie zadania na prace zaliczeniowe wysłałem już do RCKU i są opublikowane w zakładce Prace kontrolne: [http://rcku-namyslow.pl/?page\\_id=43](http://rcku-namyslow.pl/?page_id=43)

Pozdrawiam i życzę jak najwięcej zdrowia i wytrwałości.  
Andrzej Kazimierczyk

## Sterowanie sygnałem PWM.

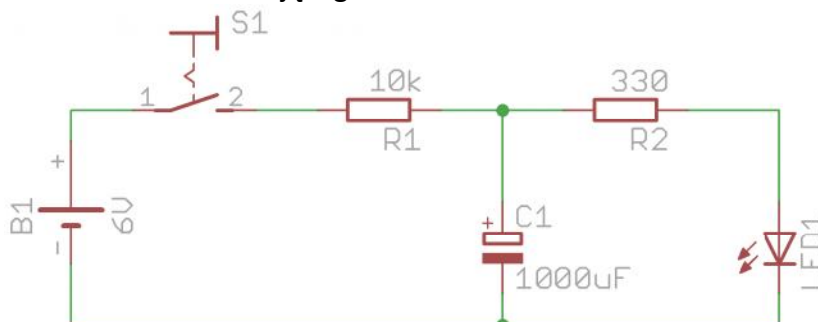
Skrót PWM zna prawie każdy, kto chociaż trochę miał styczności ze sterowaniem silników. Jego rozwinięcie to **Pulse Width Modulation** – modulacja szerokości impulsu. Te trzy słowa brzmią bardzo mądrze, ale co tak naprawdę oznaczają? Zaraz sprawdzimy to w praktyce!

### PWM – idea działania

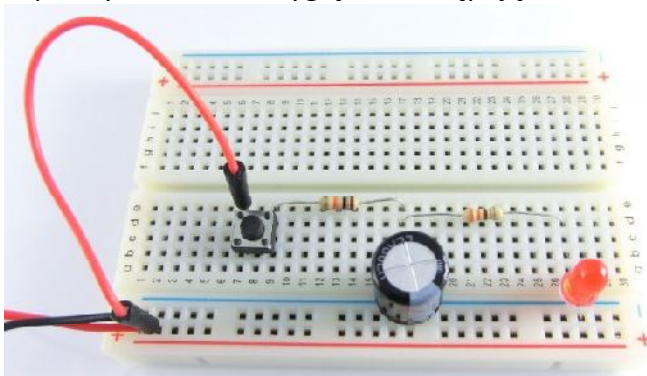
Na początek, proponuję proste doświadczenie. Będziesz potrzebował:

- płytki stykowej,
- baterii wraz z koszykiem,
- kondensatora 1000 $\mu$ F,
- rezystora 10k $\Omega$ ,
- rezystora 330 $\Omega$ ,
- diody świecącej (np. czerwonej),
- microswitcha.

Zmontuj na płytce poniższy układ. Pamiętaj o prawidłowej biegunowości kondensatora!  
Schemat układu testującego działanie PWM.



W praktyce może to wyglądać następująco:

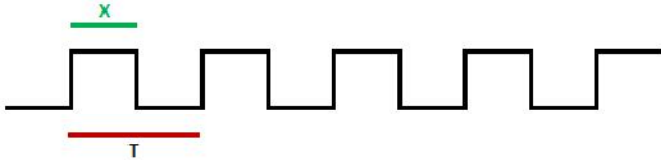


## Przykładowa realizacja ćwiczenia na płytce stykowej.

Naciśnij przycisk, gdy układ będzie gotowy. Dioda rozjaśnia się powoli, po kilku sekundach osiąga jasność maksymalną. Postaraj się ją *zapamiętać*. Teraz puść klawisz i poczekaj kilka sekund, aż dioda niemal całkowicie zgaśnie. Następnie **cyklicznie wciskaj i puszczaj przycisk** w ten sposób, aby każda faza trwała ok. pół sekundy. Zwróć uwagę, jak zachowuje się dioda. Co się dzieje, gdy dwukrotnie wydłużysz czas każdej fazy?

**Dioda świeci nieco ciemniej, ponieważ cyklicznie rozjaśnia się i ciemnieje.**

Takie równomierne wciskanie i puszczanie przycisku sprawiło, że do układu **napięcie docierało w formie poniższego sygnału prostokątnego:**



Sygnał prostokątny (jest napięcie, nie ma, jest, nie ma itd). Gdzie:

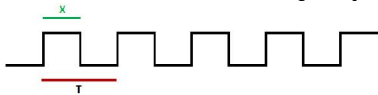
- **X**, to czas dostarczania napięcia do układu
- **T**, to czas jednego cyklu/okresu.

**Rozjaśnianie** zachodzi, gdy wciśniesz microswitch i kondensator ładuje się energią z baterii. Po odłączeniu zasilania, dioda gaśnie, ponieważ jest zasilana **wyłącznie energią zgromadzoną w kondensatorze**, która „kończy się” dość szybko. Jeżeli ładowanie i rozładowywanie kondensatora zachodzi wystarczająco często, wówczas napięcie na nim ustala się na pewnej wartości i zmienia nieznacznie (pulsowanie diody jest słabo widoczne).

**Kiedy czasy trwania tych faz wydłużą się, pulsowanie stanie się wyraźne, ponieważ zmiany napięcia na kondensatorze staną się już znaczące.**

Zakładając, że nasze wciskanie będzie odbywało się w regularnych odstępach, to wygenerujemy sygnał prostokątny, taki jak na wcześniejszym wykresie. Oczywiście czas, gdy przycisk jest wciśnięty i czas, gdy do obwodu nie dostarczamy prądu **nie musi być taki sam**.

**Stosunek czasu wciśnięcia przycisku do okresu sygnału nazwiemy wypełnieniem sygnału.**



Większe wypełnienie – dioda świeci przez większą część cyklu.



Mniejsze wypełnienie – dioda świeci przez krótszą część cyklu

Powyższy eksperyment był dużym uproszczeniem tematu, właściwie była to prosta symulacja generatora PWM. Teraz pora przejść do szczegółowego omówienia tego zagadnienia!

## Do czego wykorzystywany jest PWM?

Jak pewnie wiesz mikrokontrolery (np. te z [Arduino](#)) są układami cyfrowymi. Oznacza to, że sygnały przez nie przetwarzane to logiczne zera oraz jedynki, a patrząc z punktu widzenia elektronika – napięcia.

**Najczęściej logicznemu zeru odpowiada napięcie 0V, a logicznej jedynce napięcie zasilania układu, czyli 5V lub 3.3V.**

Powstaje jednak pytanie – jak np. płynnie regulować jasnością świecenia diody, skoro na wyjściu może pojawić się tylko jedno z dwóch napięć (0V lub 5V)? Łatwo odgadnąć, że z pomocą przyjdzie nam właśnie PWM.

Okazuje się, że sterowanie diodą jest bardzo proste – wystarczy rezystor oraz sama dioda, nie potrzebujemy nawet kondensatora. We wcześniejszym przykładzie był on niezbędnym z innego powodu –

bez niego dioda gasłaby natychmiast po zwolnieniu przycisku, więc widzielibyśmy migającą diodę, a nie łagodnie zapalającą się i gasnącą.

Jednak jeśli zamiast naciskać mikroswitch wykorzystamy procesor, będziemy mogli sterować diodą znacznie szybciej (z większą częstotliwością). Okazuje się, że ludzki wzrok nie jest doskonały i już przy 24 Hz (24 zmianach na sekundę) oko przestaje widzieć kolejne fazy obrazu i uśrednia wynik. **Znamy to działanie z kina oraz telewizji – oczywiście tak niska częstotliwość nie daje idealnego obrazu, więc lepiej jest zastosować wyższą częstotliwość odświeżania (np. 50Hz lub 100Hz).** Dla współczesnych procesorów, które często taktowane są zegarami liczonymi w GHz takie częstotliwości są bardzo małe – ale dla naszego mocno niedoskonałego zmysłu wzroku w zupełności wystarczają.

### Do czego potrzebny był kondensator?

Wróćmy jeszcze raz do naszego przykładu z kondensatorem – do czego taki układ mógłby się przydać? Jak wiemy do sterowania diodą kondensator nie jest potrzebny, nasze oko uśrednia efekt świecenia diody. No właśnie, kondensator można wykorzystać do uśredniania – napięcia. Odpowiednio dobierając wartość kondensatora (oraz 3 rezystorów), na jego zaciskach otrzymalibyśmy napięcie (prawie) stałe, regulowane płynnie w zakresie 0V – 6V w zależności od wypełnienia sygnału PWM.

W przypadku mikroprocesora taki układ jest bardzo popularny do uzyskania płynnie regulowanego napięcia wyjściowego. O ile sam procesor może dostarczyć tylko napięcia 0V oraz powiedzmy 5V, to sterując za pomocą PWM oraz podłączając rezystor i kondensator, możemy na wyjściu układu uzyskać dowolne napięcie z przedziału 0V – 5V. **Połączenie rezystora z kondensatorem tworzy prosty filtr. To właśnie dzięki niemu zamiast skoków napięcia obserwujemy jego wartość średnią.**

Możesz to w prosty sposób sam przetestować: podłącz multimetr do wyprowadzeń kondensatora 1000 $\mu$ F. Klikaj szybko mikroswitchem i zapamiętaj wskazywane wartości napięcia.

Na koniec niestety smutna wiadomość:

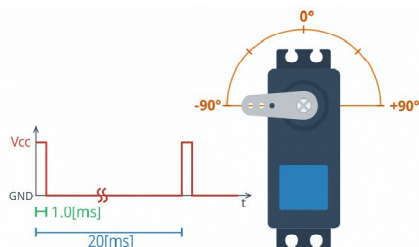
**Nie można tak prostego układu wykorzystać do bezpośredniego zasilania układu pobierającego znaczny prąd.** Aby uzyskać taki układ konieczne byłoby wzmocnienie prądu np. za pomocą tranzystora lub wzmacniacza operacyjnego.

### PWM do przesyłania sygnałów sterujących

Okazuje się, że PWM może być wykorzystany do przekazywania sygnałów między urządzeniami. Jednym z najbardziej znanych układów tego typu jest **serwomechanizm**. Przykładowy serwomechanizm Popularne serwo posiada wbudowany własny sterownik silnika – najczęściej analogowy, ale może być wyposażone w bardziej skomplikowany układ cyfrowy (tzw. *serwo cyfrowe*). W obu przypadkach **PWM jest wykorzystany jako sygnał sterujący**. Na podstawie szerokości impulsu, sterownik ustawia pozycję ramienia serwomechanizmu. Typowy sygnał sterujący serwomechanizmem ma **okres 20 ms**, natomiast szerokość impulsu zmienia się w zakresie **od 1ms do 2ms** – od minimalnego do maksymalnego wychylenia.

Impuls o szerokości 1.5 ms ustawia serwomechanizm w położeniu centralnym

### Zasada działania serwomechanizmu:



Wskaźnik analogowy z wykorzystaniem Arduino Jak pewnie zauważyłeś serwomechanizm taki znajduje się w zestawie elementów do tego kursu. Oznaczać może to tylko jedno – **będziemy nim sterować!** Jednak trochę później, musimy najpierw poznać kilka dodatkowych elementów!