Programowanie mikrokontrolerów 2.0

Sterowanie fazowe

Marcin Engel Marcin Peczarski

Instytut Informatyki Uniwersytetu Warszawskiego

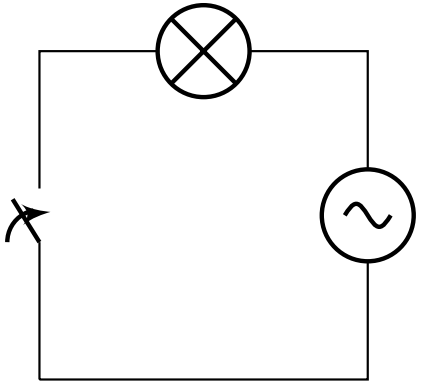
2 października 2022

Triak

- Triak jest półprzewodnikowym elementem przełączającym z trzema końcówkami: MT1, MT2, G
- Włącza się go w obwód prądu przemiennego (anody MT1 i MT2) i steruje się go za pomocą bramki G
- Triak włącza się, gdy przez bramkę popłynie prąd
- Triak pozostaje włączony tak długo, jak długo płynie prąd anodowy (nawet jeśli wcześniej przestanie płynąć prąd bramki)

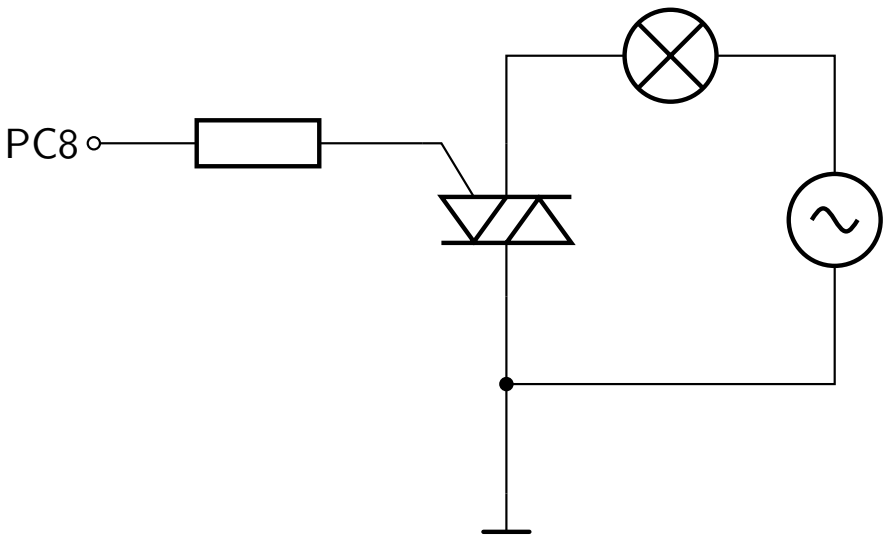
Prosty obwód prądu przemiennego

- Rozważmy prosty obwód prądu przemiennego
- Jako odbiornik wykorzystamy żarówkę



Sterowanie za pomocą triaka

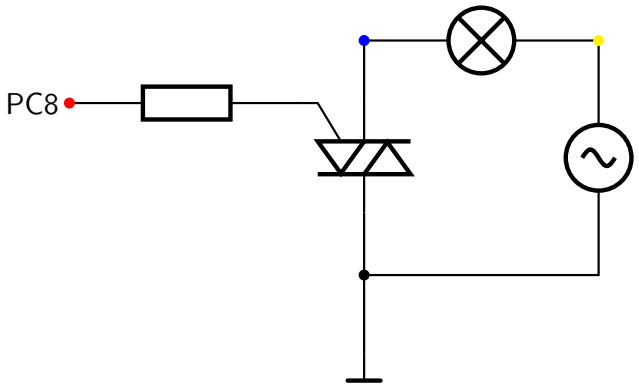
- Zamiast zwykłego włącznika użyjmy triaka sterowanego przez mikrokontroler (wyprowadzenie PC8)



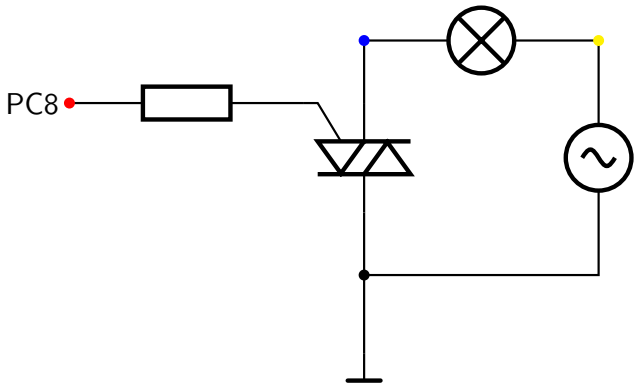
Włączanie i wyłączanie triaka

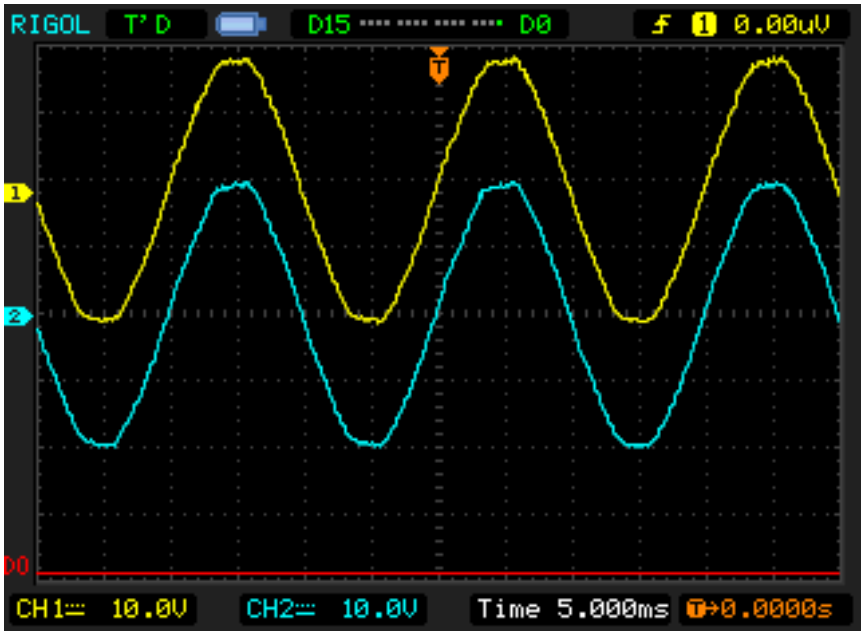
- Co się stanie, gdy na wyprowadzeniu PC8 mikrokontrolera będzie:
  - stan niski?
  - stan wysoki?
- Jak będą wyglądać przebiegi napięcia na końcówkach odbiornika w obu powyższych sytuacjach?

Triak wyłączony

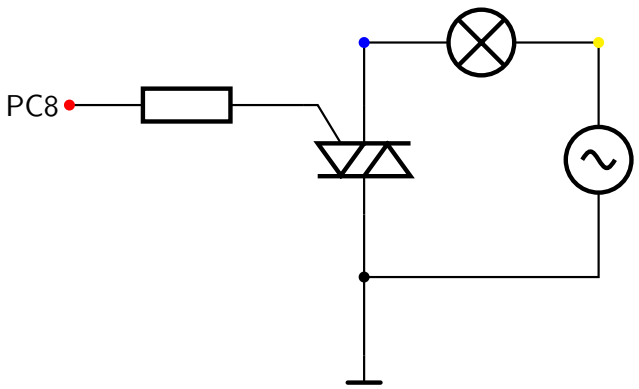


Triak wyłączony

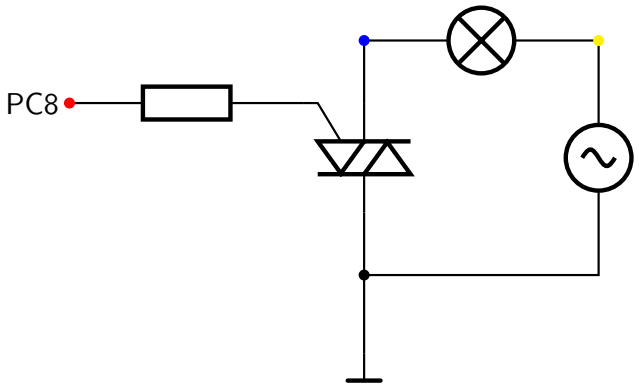


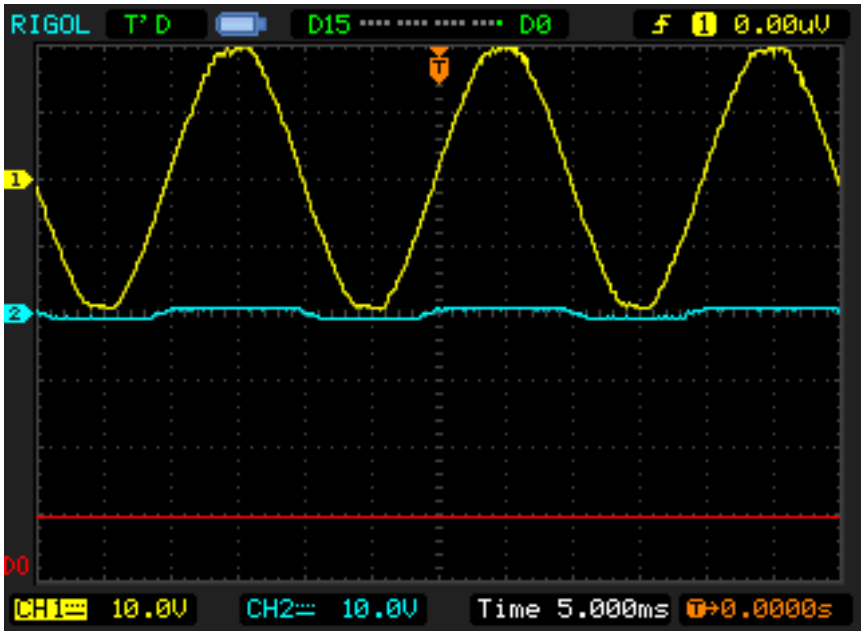


Triak włączony



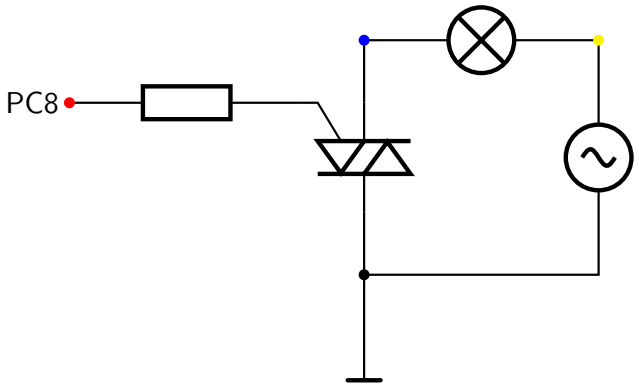
Triak włączony





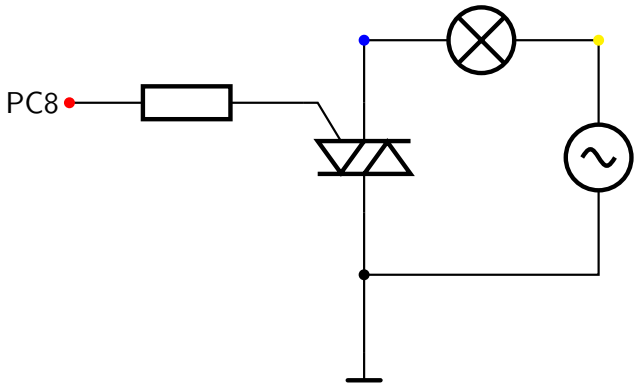
Wyłączenie triaka

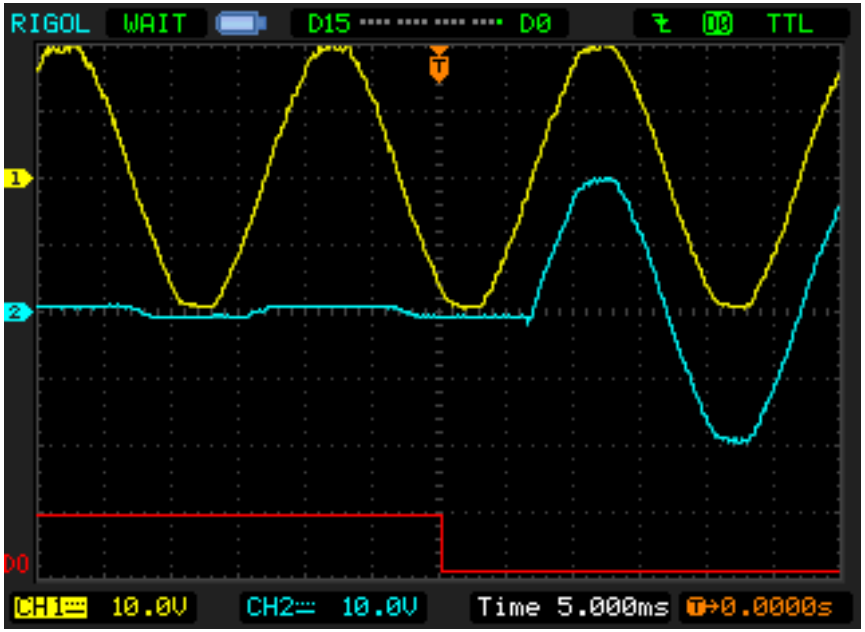
Przyjrzyjmy się dokładniej chwili, gdy stan PC8 zmienia się z wysokiego na niski



Wyłączenie triaka

Przyjrzyjmy się dokładniej chwili, gdy stan PC8 zmienia się z wysokiego na niski



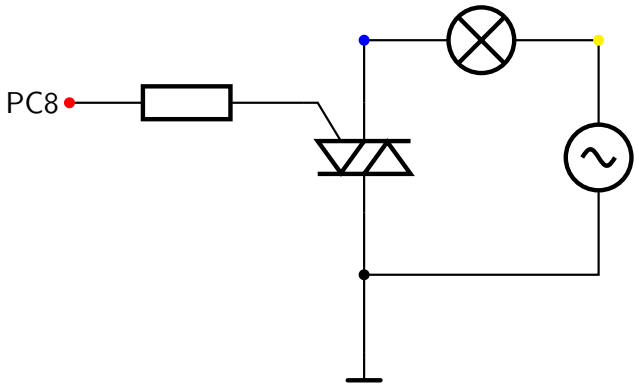


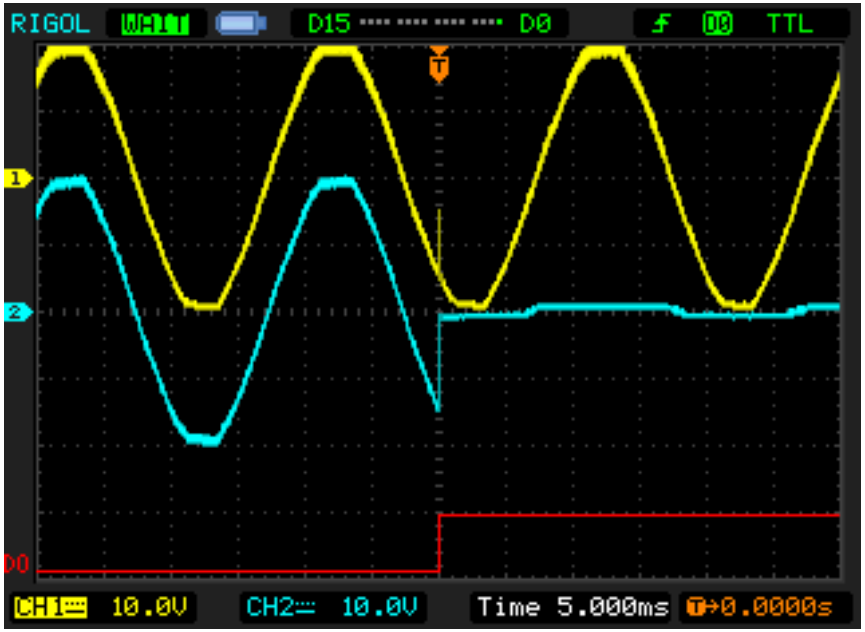
Wyłączenie triaka, cd.

- Żarówka gaśnie
- Przebieg napięcia na końcówce odbiornika od strony triaka przybiera postać sinusoidy
- Napięcie na odbiorniku osiąga zero
- Nie ma gwałtownego skoku napięcia — wyłączenie następuje dopiero, gdy napięcie zasilania przechodzi przez zero

Włączenie triaka, cd.

Przyjrzyjmy się dokładniej chwili, gdy stan wyjścia zmienia się z niskiego na wysoki





Włączenie triaka

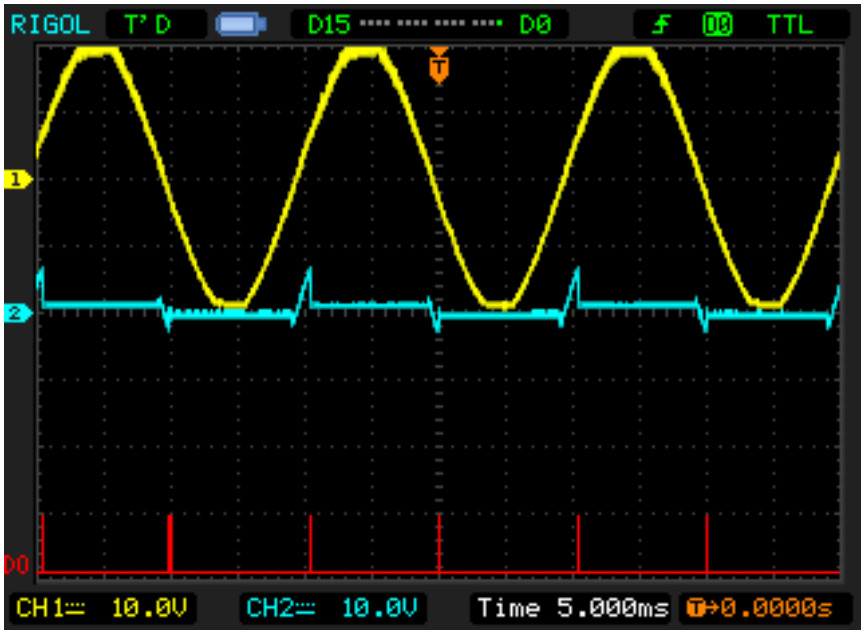
- Przez bramkę triaka zaczyna płynąć prąd
- Żarówka zaświeca się
- Napięcie na końcówce odbiornika od strony triaka spada do (prawie) zera
- Przebieg napięcia na odbiorniku ma postać sinusoidy

Sterowanie impulsami

- Zamiast podawać na wyjście stale stan wysoki spróbujmy generować krótki impuls (rzędu 10 μs)
- Czy żarówka zaświeci się?
- Jak będą wyglądać przebiegi napięcia na zaciskach odbiornika?

Sterowanie impulsami

- Zamiast podawać na wyjście stale stan wysoki spróbujmy generować krótki impuls (rzędu 10 μs)
- Czy żarówka zaświeci się?
- Jak będą wyglądać przebiegi napięcia na zaciskach odbiornika?

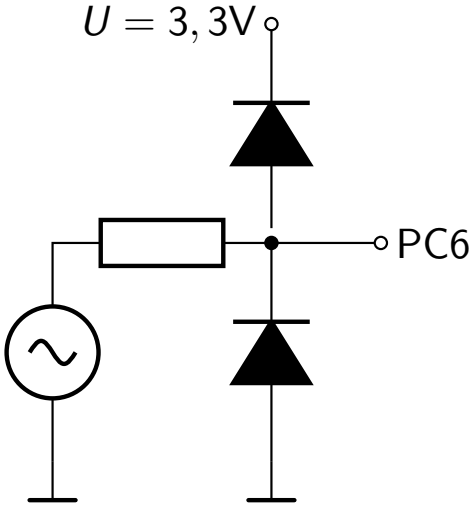


Wnioski

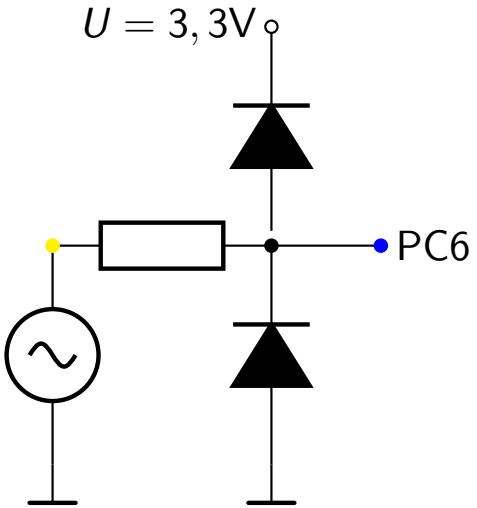
- Triakiem możemy sterować za pomocą mikrokontrolera
- Sterowanie może odbywać się poprzez ciągłe podawanie na bramkę odpowiednio dużego prądu
- Można też podawać na bramkę odpowiednio zsynchronizowane impulsy prądowe
- Załączenie triaka powoduje zakłócenia
- Można temu zapobiegać, włączając triak w chwili przejścia przez zero napięcia zasilającego!

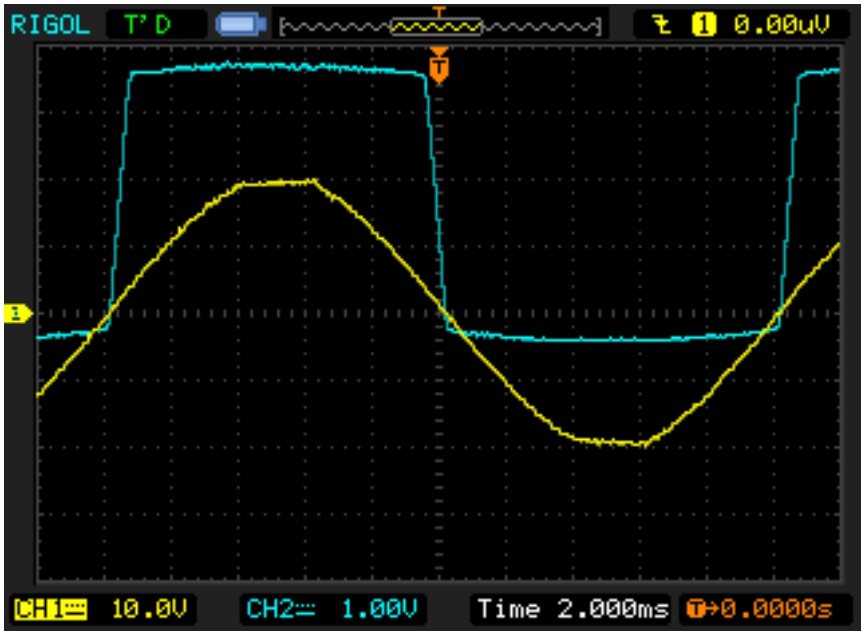
Wykrywanie zera

- Napięcie przemienne podajemy przez rezystor o dużej wartości na wejście mikrokontrolera
- Zabezpieczamy wejście diodami odprowadzającymi nadmiar prądu, które ukształtują przebieg w prawie prostokątny



Wykrywanie zera, cd.





Wykrywanie zera, przykładowa realizacja

- Konfigurujemy przerwanie zewnętrzne na wejściu tak, aby było wyzwalane zboczem opadającym
- Po pojawieniu się przerwania upewniamy się, że nie zostało wywołane przez zakłócenia, np. czekamy przez jakiś czas i sprawdzamy jeszcze raz stan wejścia
- W ten sposób wykryjemy przejście napięcia z fazy dodatniej do ujemnej

Sterowanie impulsami, przykładowa realizacja

- Gdy wykryjemy przejście przez zero, generujemy impuls na wyjście — włączamy triak na pół okresu
- Generujemy kolejny sygnał po upływie 10 ms (dla częstotliwości sieci 50 Hz) — to załączy triak na kolejne pół okresu
- Innymi słowy: generujemy impulsy co 10 ms, synchronizując je cały czas z częstotliwością napięcia w obwodzie anodowym

Sterowanie fazowe

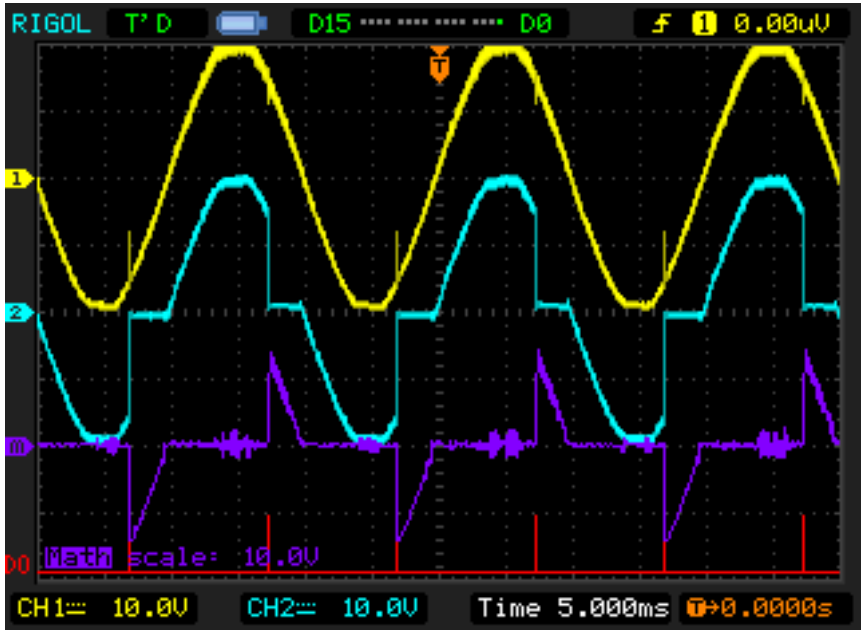
- Co się stanie, jeśli opóźnimy impulsy w stosunku do wykrycia zera?
- Sprawdźmy!

Sterowanie fazowe

- Co się stanie, jeśli opóźnimy impulsy w stosunku do wykrycia zera?
- Sprawdźmy!
  - Jasność żarówki zależy od opóźnienia momentu włączenia triaka
  - Przebieg napięcia na odbiorniku ma w każdym półokresie postać fragmentu sinusoidy
  - Takie sterowanie nosi nazwę sterowania fazowego

Sterowanie fazowe

- Co się stanie, jeśli opóźnimy impulsy w stosunku do wykrycia zera?
- Sprawdźmy!
  - Jasność żarówki zależy od opóźnienia momentu włączenia triaka
  - Przebieg napięcia na odbiorniku ma w każdym półokresie postać fragmentu sinusoidy
  - Takie sterowanie nosi nazwę sterowania fazowego



Sterowanie fazowe, przykładowa realizacja

- Można postąpić tak jak przy wykrywaniu przejścia przez zero
- Można jednak robić to sprytniej, bez aktywnego czekania

Sterowanie fazowe, przykładowa realizacja

- Można postąpić tak jak przy wykrywaniu przejścia przez zero
- Można jednak robić to sprytniej, bez aktywnego czekania i bez konieczności obsługi przerwań!

Sterowanie fazowe, przykładowa realizacja

- Można postąpić tak jak przy wykrywaniu przejścia przez zero
- Można jednak robić to sprytniej, bez aktywnego czekania i bez konieczności obsługi przerwań!

```
int main () {
  init_timer();
  for(;;) {}
}
```

Sterowanie fazowe, przykładowa realizacja

- Można postąpić tak jak przy wykrywaniu przejścia przez zero
- Można jednak robić to sprytniej, bez aktywnego czekania i bez konieczności obsługi przerwań!

```
int main () {
  init_timer();
  for(;;) {}
}
```

- Znów przyda się odpowiednio skonfigurowany licznik – wykorzystamy licznik `TIM3`

Konfiguracja licznika

- Przypomnienie:
  - licznik jest wyposażony w kanały
  - z kanałem jest związana linia wewnętrzna
  - na linii wewnętrznej można tworzyć przebiegi, wykorzystując rejestr zgodności `TIM3->CCR`*y*
  - linie wewnętrzne można podłączyć do wyprowadzeń mikrokontrolera

Konfiguracja licznika

- Przypomnienie:
  - licznik jest wyposażony w kanały
  - z kanałem jest związana linia wewnętrzna
  - na linii wewnętrznej można tworzyć przebiegi, wykorzystując rejestr zgodności `TIM3->CCR`*y*
  - linie wewnętrzne można podłączyć do wyprowadzeń mikrokontrolera
- Licznik może być tak ustawiony, aby wykonywał tylko jeden cykl
- Tę funkcjonalność wykorzystamy do generowania impulsów

Generowanie pojedynczych impulsów

- Włączamy taktowanie portu `GPIOC` i licznika `TIM3`

```
RCC->AHB1ENR |= RCC_AHB1ENR_GPIOCEN;
RCC->APB1ENR |= RCC_APB1ENR_TIM3EN;
```

Generowanie pojedynczych impulsów

- Włączamy taktowanie portu GPIOC i licznika TIM3

```
RCC->AHB1ENR |= RCC_AHB1ENR_GPIOCEN;
RCC->APB1ENR |= RCC_APB1ENR_TIM3EN;
```

- Do wyprowadzeń PC6 i PC8 będą podłączone linie wewnętrzne licznika (kanał pierwszy i trzeci)

```
GPIOafConfigure(GPIOC, 6, GPIO_OType_PP,
      GPIO_Low_Speed, GPIO_PuPd_NOPULL, 2);
GPIOafConfigure(GPIOC, 8, GPIO_OType_PP,
      GPIO_Low_Speed, GPIO_PuPd_NOPULL, 2);
```

Generowanie pojedynczych impulsów

- Włączamy taktowanie portu GPIOC i licznika TIM3

```
RCC->AHB1ENR |= RCC_AHB1ENR_GPIOCEN;
RCC->APB1ENR |= RCC_APB1ENR_TIM3EN;
```

- Do wyprowadzeń PC6 i PC8 będą podłączone linie wewnętrzne licznika (kanał pierwszy i trzeci)

```
GPIOafConfigure(GPIOC, 6, GPIO_OType_PP,
      GPIO_Low_Speed, GPIO_PuPd_NOPULL, 2);
GPIOafConfigure(GPIOC, 8, GPIO_OType_PP,
      GPIO_Low_Speed, GPIO_PuPd_NOPULL, 2);
```

- Konfigurujemy licznik tak, aby jego zwiększanie odbywało się co 10 μs (przy domyślnym taktowaniu 16 MHz):

```
TIM3->PSC = 159;
```

Generowanie impulsów

- Na kanale 3 chcemy utworzyć pojedynczy impuls

Generowanie impulsów

- Na kanale 3 chcemy utworzyć pojedynczy impuls
- Użyjemy trybu PWM2 z licznikiem liczącym w górę

Generowanie impulsów

- Na kanale 3 chcemy utworzyć pojedynczy impuls
- Użyjemy trybu PWM2 z licznikiem liczącym w górę (linia jest nieaktywna wtw., gdy `TIM3->CNT` $<$ `TIM3->CCR3`)

Generowanie impulsów

- Na kanale 3 chcemy utworzyć pojedynczy impuls
- Użyjemy trybu PWM2 z licznikiem liczącym w górę (linia jest nieaktywna wtw., gdy `TIM3->CNT` $<$ `TIM3->CCR3`)
- Wartość rejestru `TIM3->CCR3` wyznacza opóźnienie rozpoczęcia impulsu
- Wartość `TIM3->ARR` $-$ `TIM3->CCR3` $+ 1$ wyznacza czas jego trwania

Generowanie impulsów

- Na kanale 3 chcemy utworzyć pojedynczy impuls
- Użyjemy trybu PWM2 z licznikiem liczącym w górę (linia jest nieaktywna wtw., gdy `TIM3->CNT` $<$ `TIM3->CCR3`)
- Wartość rejestru `TIM3->CCR3` wyznacza opóźnienie rozpoczęcia impulsu
- Wartość `TIM3->ARR` $-$ `TIM3->CCR3` $+ 1$ wyznacza czas jego trwania
- Impuls długości 10 μs zaczynający się po 1 ms

```
TIM3->ARR = 100;
TIM3->CCR3 = 100;
```

Generowanie impulsów

- Na kanale 3 chcemy utworzyć pojedynczy impuls
- Użyjemy trybu PWM2 z licznikiem liczącym w górę (linia jest nieaktywna wtw., gdy `TIM3->CNT` $<$ `TIM3->CCR3`)
- Wartość rejestru `TIM3->CCR3` wyznacza opóźnienie rozpoczęcia impulsu
- Wartość `TIM3->ARR` $-$ `TIM3->CCR3` $+ 1$ wyznacza czas jego trwania
- Impuls długości 10 μs zaczynający się po 1 ms

```
TIM3->ARR = 100;
TIM3->CCR3 = 100;
```

- Zerowanie licznika, aktualizacja rejestrów

```
TIM3->EGR = TIM_EGR_UG;
```

Konfiguracja kanału wyjściowego

- Konfigurujemy tryb PWM2 na kanale 3:

```
TIM3->CCMR2 = TIM_CCMR2_OC3M_2 |
              TIM_CCMR2_OC3M_1 |
              TIM_CCMR2_OC3M_0;
```

Konfiguracja kanału wyjściowego

- Konfigurujemy tryb PWM2 na kanale 3:

```
TIM3->CCMR2 = TIM_CCMR2_OC3M_2 |
              TIM_CCMR2_OC3M_1 |
              TIM_CCMR2_OC3M_0;
```

- Podłączamy wyjście do wyprowadzenia:

```
TIM3->CCER = TIM_CCER_CC3E;
```

Konfiguracja kanału wyjściowego

- Konfigurujemy tryb PWM2 na kanale 3:

```
TIM3->CCMR2 = TIM_CCMR2_OC3M_2 |
              TIM_CCMR2_OC3M_1 |
              TIM_CCMR2_OC3M_0;
```

- Podłączamy wyjście do wyprowadzenia:

```
TIM3->CCER = TIM_CCER_CC3E;
```

- Ustawiamy tryb jednokrotnego wyzwalania licznika:

```
TIM3->CR1 = TIM_CR1_OPM;
```

Konfiguracja kanału wyjściowego

- Konfigurujemy tryb PWM2 na kanale 3:

```
TIM3->CCMR2 = TIM_CCMR2_OC3M_2 |
              TIM_CCMR2_OC3M_1 |
              TIM_CCMR2_OC3M_0;
```

- Podłączamy wyjście do wyprowadzenia:

```
TIM3->CCER = TIM_CCER_CC3E;
```

- Ustawiamy tryb jednokrotnego wyzwalania licznika:

```
TIM3->CR1 = TIM_CR1_OPM;
```

- Ale nie włączamy licznika!

Jak zsynchronizować impulsy z przejściem przez zero?

- Licznik może być uruchamiany stanem lub zmianą stanu kanału, skonfigurowanego jako wejście
- Użyjemy kanału 1, wyprowadzenie `PC6`

Jak zsynchronizować impulsy z przejściem przez zero?

- Licznik może być uruchamiany stanem lub zmianą stanu kanału, skonfigurowanego jako wejście
- Użyjemy kanału 1, wyprowadzenie `PC6`
- Licznik będzie uruchamiany zarówno zboczem opadającym, jak i narastającym:

```
TIM3->CCER |= TIM_CCER_CC1P | TIM_CCER_CC1NP;
```

Jak zsynchronizować impulsy z przejściem przez zero?

- Licznik może być uruchamiany stanem lub zmianą stanu kanału, skonfigurowanego jako wejście
- Użyjemy kanału 1, wyprowadzenie PC6
- Licznik będzie uruchamiany zarówno zboczem opadającym, jak i narastającym:

```
TIM3->CCER |= TIM_CCER_CC1P | TIM_CCER_CC1NP;
```

- Ustalamy tryb pracy licznika i kanał wejściowy:

```
TIM3->SMCR = TIM_SMCR_TS_2 | TIM_SMCR_TS_0 |
             TIM_SMCR_SMS_2 | TIM_SMCR_SMS_1;
```

Jak zsynchronizować impulsy z przejściem przez zero?

- Licznik może być uruchamiany stanem lub zmianą stanu kanału, skonfigurowanego jako wejście
- Użyjemy kanału 1, wyprowadzenie PC6
- Licznik będzie uruchamiany zarówno zboczem opadającym, jak i narastającym:

```
TIM3->CCER |= TIM_CCER_CC1P | TIM_CCER_CC1NP;
```

- Ustalamy tryb pracy licznika i kanał wejściowy:

```
TIM3->SMCR = TIM_SMCR_TS_2 | TIM_SMCR_TS_0 |
             TIM_SMCR_SMS_2 | TIM_SMCR_SMS_1;
```

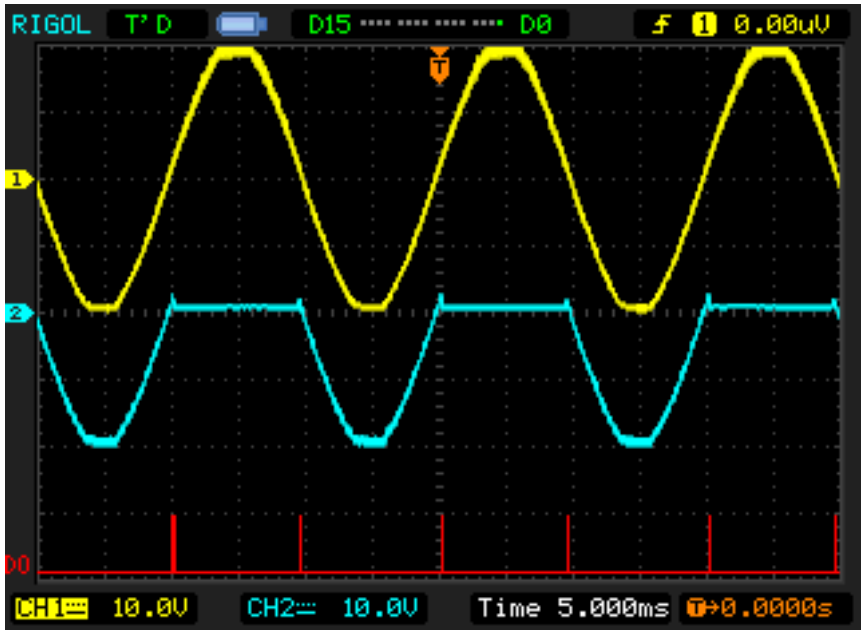
- Już!

Problem

- Przyjrzyjmy się przebiegom przy małych wartościach opóźnień impulsu

Problem

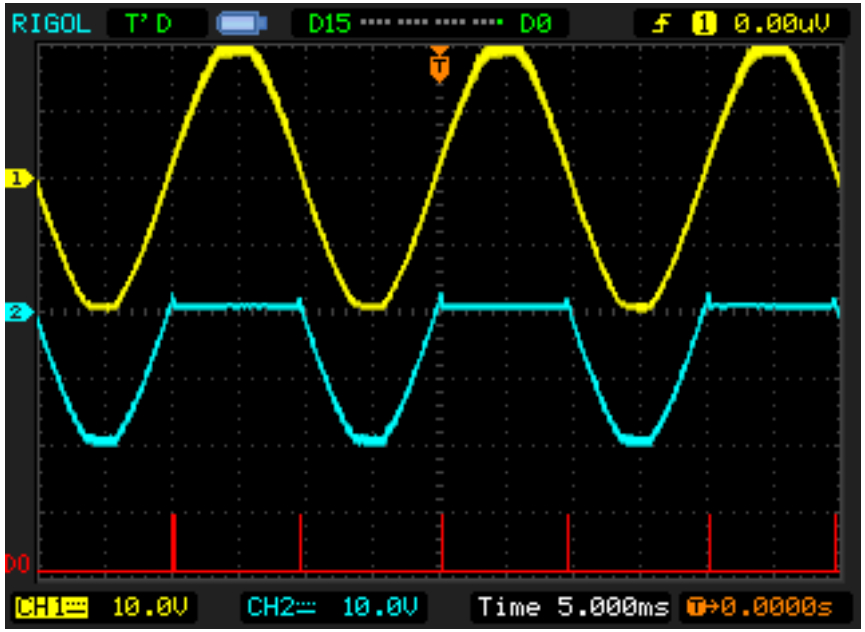
- Przyjrzyjmy się przebiegom przy małych wartościach opóźnień impulsu



- Triak nie włącza obwodu zasilania na ujemny półokres - dlaczego?

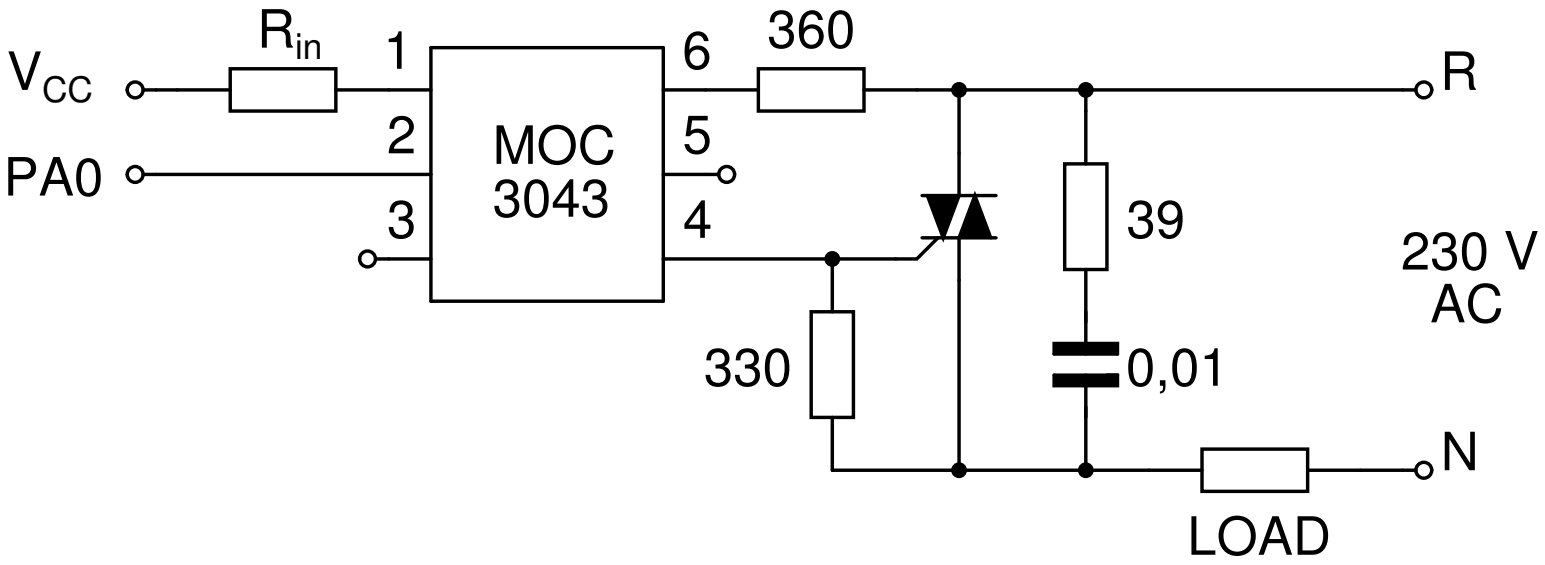
Problem

- Przyjrzyjmy się przebiegom przy małych wartościach opóźnień impulsu



- Triak nie włącza obwodu zasilania na ujemny półokres - dlaczego?
  - zbocze opadające sygnału podawanego na wejście wyprzedza moment przejścia napięcia przez zero
  - impuls jest generowany przed faktycznym przejściem przez zero
- Trzeba to wziąć po uwagę!

Optotriak

- Działa jak triak, ale zapewnia izolację galwaniczną obwodów sterowanego i sterującego
- Jest używany jako układ pośredniczący i zabezpieczający
- Może zawierać w sobie układ wykrywający przejście przez zero

Poprawny układ sterujący obwodem prądu przemiennego