

LABORATORIUM 4

Sterowanie urządzeniami wykonawczymi

1. Cel ćwiczenia.

Celem ćwiczenia jest poznanie podstaw sterowania cyfrowego i analogowego różnymi urządzeniami wykonawczymi. W ćwiczeniu w praktyce poznasz sposób użycia przełącznika elektromagnetycznego (przełącznika) oraz elektronicznego (tranzystora). Zbudujesz obwód i poznasz sposób sterowania napędami prądu stałego.

2. Zadania do wykonania.

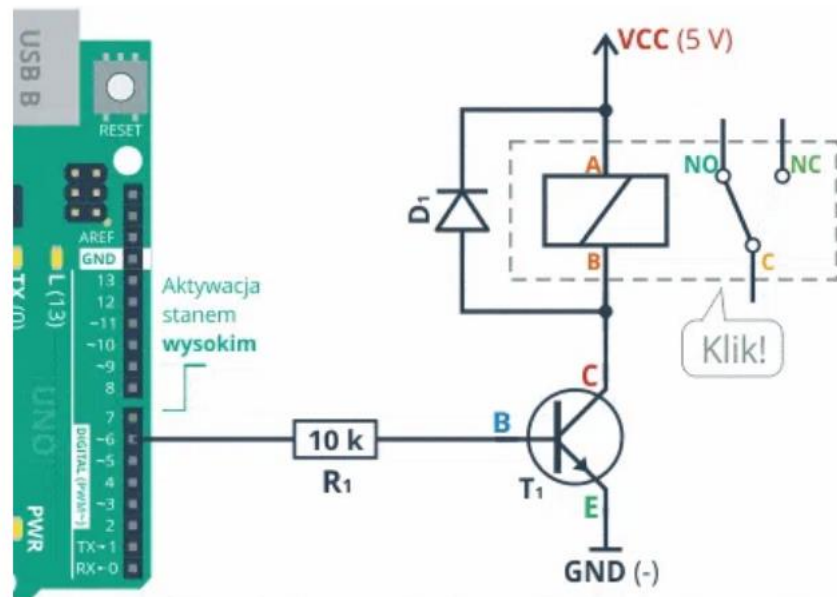
Zadanie 1. Sterowanie dwustanowe za pomocą przełącznika i tranzystora

Krok 1: Sprawdzenie funkcjonowania przełącznika

- umieść przełącznik na płytce stykowej
- doprowadź do linii zasilających płytki stykowej napięcie 5V z zasilacza laboratoryjnego lub innego źródła napięcia
- styki elektromagnesu przełącznika połącz z odpowiednio z biegunem dodatnim oraz ujemnym zasilania
- umieść na płytce stykowej diodę półprzewodnikową tak, aby zabezpieczyć układ przed zwarciami przy przełączaniu
- do styku NO przełącznika doprowadź „+” zasilania
- do styku C przełącznika podłącz anodę LED (pamiętaj o dobraniu rezystora ograniczającego prąd), katodę LED do „-” zasilania
- uruchom zasilacz i ustaw napięcie zasilania 0V (Uwaga: upewnij się, że po włączeniu zasilacza napięcie nie przekracza 5V przed podłączeniem zasilania do płytki stykowej)
- zwiększaj stopniowo wartość napięcia aż do około 5V obserwuj efekt załączenia przełącznika po doprowadzeniu do cewki przełącznika napięcia

Krok 2: Sterowanie dwustanowe za pomocą mikrokontrolera.

- odczytaj symbol z obudowy tranzystora bipolarnego i odszukaj jego dokumentację w Internecie
- odczytaj w dokumentacji podstawowe parametry tranzystora oraz zidentyfikuj jego wyprowadzenia
- zbuduj układ sterowania jak na rysunku 1



Rys. 1. Układ sterowania dwustanowego za pomocą przełącznika i mikrokontrolera (źródło: forbot)

- doprowadź zasilanie do układu ze znajdującego się w zestawie modułu konwertera napięć
- wykorzystaj układ do załączania silnika prądu stałego

Zadanie 2. Sterowanie silnikiem DC za pomocą mostka H

Krok 1: Budowa układu sterowania

- odszukaj w Internecie dokumentację mostka H L293D
- odczytaj z dokumentacji podstawowe parametry układu oraz zidentyfikuj jego wyprowadzenia. Do jakich silników może być użyty ten układ?
- umieść układ mostka na płytce stykowej

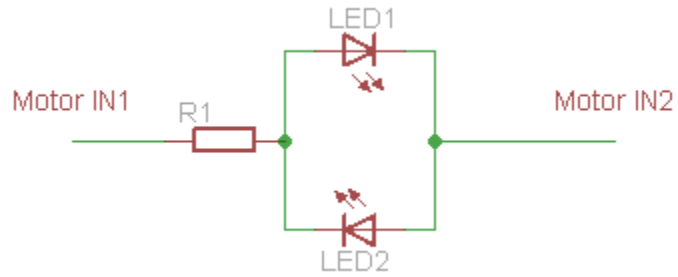
- połącz piny 4, 5, 12, 13 mostka z „masą” zasilania
- doprowadź do pinu 16 Vss zasilanie części logicznej: 5V bezpośrednio z Arduino
- doprowadź do pinu 8 Vs zasilanie silników: użyj zasilacza laboratoryjnego lub innego źródła zasilania, ustaw odpowiednią dla Twojego silnika wartość napięcia zasilania
- piny 7 (INPUT1) oraz 8 (INPUT2) podłącz do wyjść cyfrowych Arduino
- pin 1 (ENABLE1) podłącz do pinu Arduino z oznaczeniem „~” (PWM)
- do pinów 3 oraz 6 podłącz LED tak, jak na rysunku 2
- przykładowy sposób połączenia pokazany jest na rysunku 3
- napisz program przełączający cyklicznie kierunek obrotów sygnalizowany LED
- uruchom układ i zaobserwuj jego funkcjonowanie. Dokonaj pomiaru napięć na wyjściach mostka

```
#define speed 10
#define right 9
#define left 8

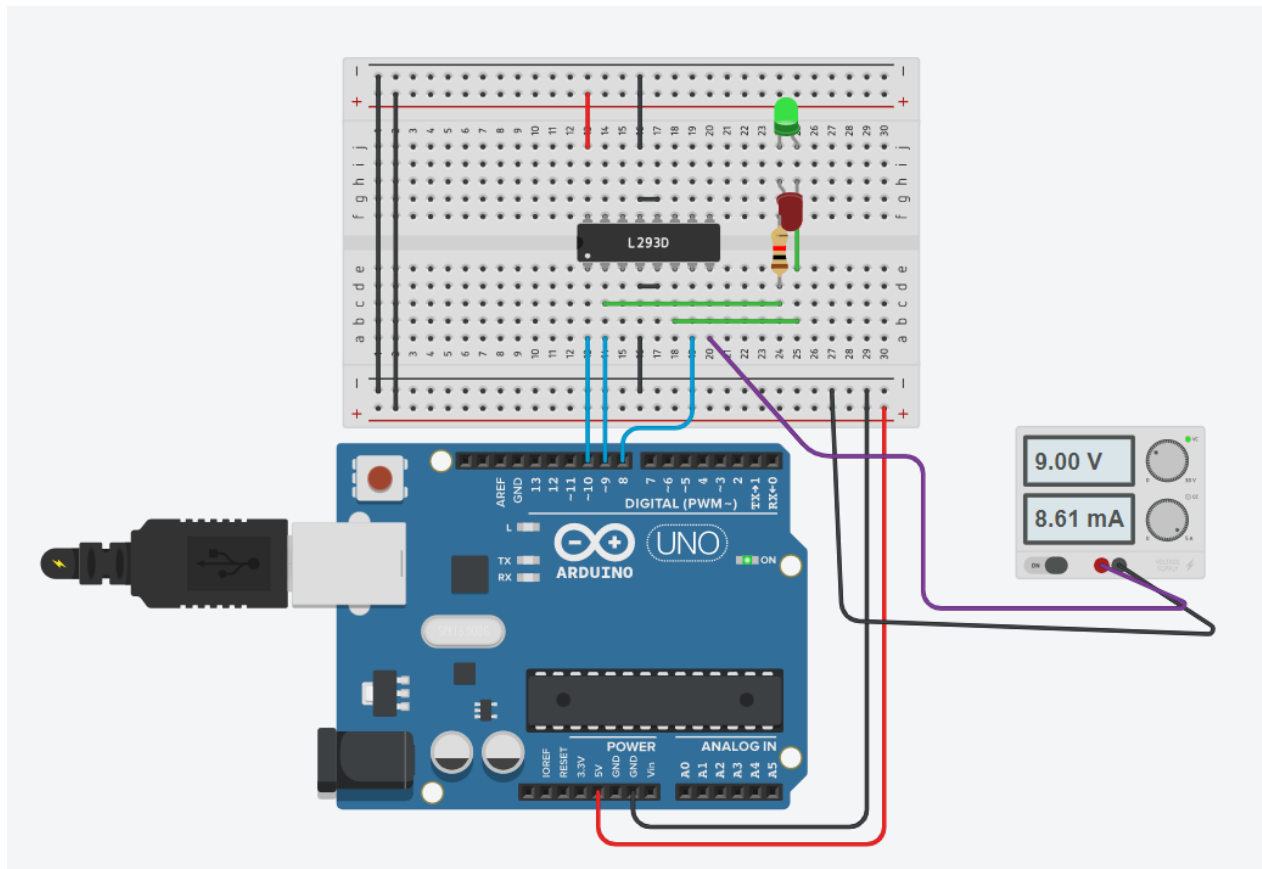
void setup()
{
  pinMode(speed, OUTPUT);
  pinMode(right, OUTPUT);
  pinMode(left, OUTPUT);

  digitalWrite(speed, HIGH);
  digitalWrite(right, LOW);
  digitalWrite(left, LOW);
}

void loop()
{
  digitalWrite(right, HIGH);
  digitalWrite(left, LOW);
  delay(3000);
  digitalWrite(right, LOW);
  digitalWrite(left, HIGH);
  delay(3000);
}
```



Rys. 2. Sposób połączenia LED do wyjść mostka H



Rys. 3 Przykładowy sposób układu sterowania kierunkiem obrotów silnika za pomocą mostka H L293D

- zastęp LED silnikiem DC

Krok 2: Regulacja prędkości obrotowej silnika.

- użyj funkcji **analogWrite(nr_pinu, wsp_wypełnienia)** dla pinu Arduino, do którego podłączone jest wyprowadzenie **ENABLE1** mostka H
- przetestuj funkcję dla różnych wartości współczynnika wypełnienia (drugi argument funkcji w zakresie 0–255)
- podłącz oscyloskop do wejścia **ENABLE** mostka H. Zarejestruj przebiegi na oscyloskopie dla różnych wartości współczynnika wypełnienia. Jaka jest częstotliwość generowanego sygnału PWM?

Krok 3: Opcjonalnie:

- zaproponuj użycie potencjometru do cyfrowej regulacji prędkości obrotowej silnika DC: podłącz potencjometr do przetwornika A/C Arduino, odczytaną wartość użyj do ustawienia współczynnika wypełnienia PWM (przydatna będzie funkcja *map()*)

Zadanie 3. Sterowanie serwonapędem

Krok 1: Podłączenie serwomechanizmu do Arduino.

- zidentyfikuj wyprowadzenia serwomechanizmu
- doprowadź zasilanie do serwomechanizmu z zewnętrznego źródła zasilania
- podłącz sygnał sterujący serwomechanizmu do pinu PWM Arduino

Krok 2: Regulacja położenia serwomechanizmu.

- w programie Arduino, dodaj bibliotekę **Servo.h**
- utwórz obiekt klasy Servo, np. **Servo serwomechanizm**
- wewnątrz funkcji **Setup()**, wywołaj funkcję **attach(nr_pinu)** dla obiektu klasy Servo, podając jako argument nr pinu Arduino do którego podłączone jest wejście sterujące serwomechanizmu
- za pomocą funkcji **write** wywołanej dla obiektu klasy **Servo** steruj wychyleniem serwomechanizmu, podając jako argument kąt w przedziale 0–180

– przykładowy program

```

4 #include <Servo.h> //Biblioteka odpowiedzialna za serwa
5 Servo serwomechanizm; //Tworzymy obiekt, dzięki któremu możemy odwołać się do serwa
6 int pozycja = 0; //Aktualna pozycja serwa 0-180
7 int zmiana = 6; //Co ile ma się zmieniać pozycja serwa?
8
9 void setup()
10{
11   serwomechanizm.attach(9); //Serwomechanizm podłączony do pinu 9
12}
13
14void loop()
15{
16   if (pozycja < 180) { //Jeśli pozycja mieści się w zakresie
17     serwomechanizm.write(pozycja); //Wykonaj ruch
18   } else { //Jeśli nie, to powrót na początek
19     pozycja = 0;
20   }
21
22   pozycja = pozycja + zmiana; //Zwiększenie aktualnej pozycji serwa
   delay(200); //Opóźnienie dla lepszego efektu
}

```

Zadanie 3. Sterowanie silnikiem skokowym unipolarnym

Krok 1: Identyfikacja wyprowadzeń silnika skokowego

- zidentyfikuj rodzaj silnika skokowego: 4 wyprowadzenia – silnik bipolarny, 6 wyprowadzeń – silnik unipolarny
- dokonaj pomiaru rezystancji pomiędzy poszczególnymi wyprowadzeniami i umieść wyniki w odpowiedniej tabeli (X oznacza brak połączenia)

- dla silnika unipolarnego

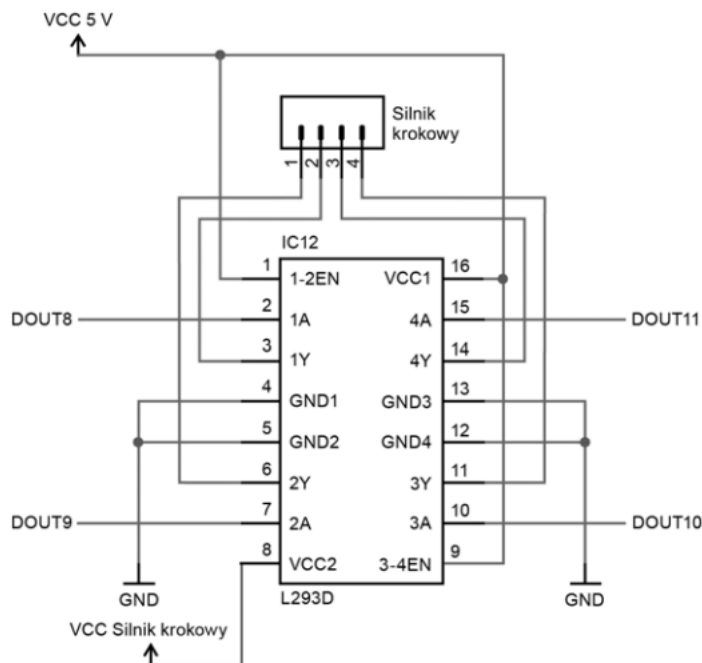
kolor przewodu						

- dla silnika bipolarnego

kolor przewodu				

Krok 2: Budowa układu sterowania (s 123)

- zbuduj układ sterowania silnikiem krokowym w oparciu o mostek H i Arduino tak, jak na schemacie poniżej



- w przypadku używania silnika unipolarnego, zewrzyj środkowe wyprowadzenia do masy

Krok 3: Program Arduino

- dołącz bibliotekę **Stepper.h** w nagłówku programu
- utwórz obiekt klasy **Stepper**, w konstruktorze podając kolejno następujące parametry: liczba kroków na obrót, piny Arduino 8, 9, 10, 11

- wewnątrz funkcji **Setup()**, dla obiektu klasy **Stepper** wywołaj funkcję **setSpeed** podając jako argument prędkość obrotu silnika równą 30 obr/min
- za pomocą funkcji **step** wywołanej dla obiektu klasy **Stepper** dokonaj obrotu wału silnika o określoną liczbę kroków (argument funkcji **step**)
- przykładowy program do sprawdzenia funkcjonowania silnika skokowego może wyglądać tak, jak poniżej

```
#include <Stepper.h>

stepsPerRevolution = 200;
Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);
int stepCount = 0;
void setup() {
    Serial.begin(9600);
}

void loop() {
    myStepper.step(1);
    Serial.print("Kroki: ");
    Serial.println(stepCount);
    stepCount++;
    delay(500);
}
```