

Trójwymiarowy labirynt

Ta prosta konstrukcja może dostarczyć wiele uciechy, tak podczas budowania i programowania systemu, jak i przechodzenia labiryntu. Została zbudowana z zastosowaniem modułu Arduino Uno, który steruje ruchami labiryntu poprzez serwo mechanizmy. Zabawkę kontroluje się za pomocą dwóch joysticków.

Do budowy urządzenia potrzebnych będzie kilka modułów elektronicznych oraz elementy i materiały niezbędne do wykonania podstawy i zastępujące obudowę. Najważniejsze z nich to:

- moduł Arduino Uno,
- płytki stykowe,
- dwa moduły joysticka,
- cztery serwo silniki SG90,
- kable do połączenia elementów,
- sztywny karton bądź cienki MDF,
- śruby, nakrętki itp. elementy połączeniowe.

Joysticki

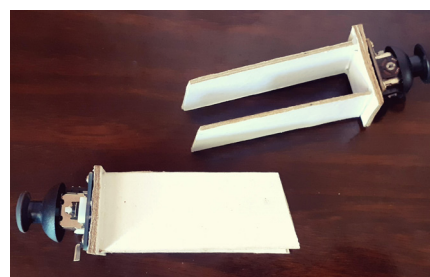
W pierwszej kolejności zmontujemy kontroler do sterowania zabawką. Składa się on z dwóch modułów joysticków analogowych oraz podstawek, wykonanych z kartonu bądź MDF-u. Konstrukcja jest bardzo

prosta – należy umieścić moduł joysticka na tekturze i dodać dwa boczne elementy kartonowe tak, jak to zostało pokazane na **fotografii 1**. Joysticki posłużą do sterowania serwo mechanizmami poruszającymi labiryntem.

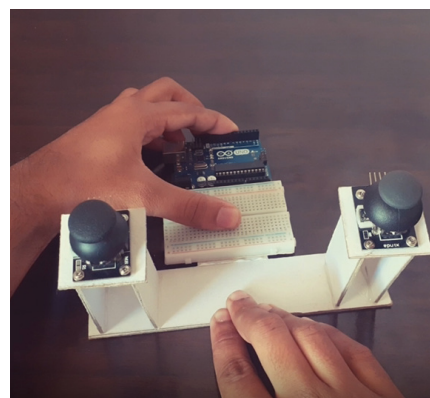
W następnej kolejności należy połączyć elementy kontrolera z elektroniką sterującą. Na wspólnej podstawie montowana jest płytka stykowa oraz moduły z joystickami na podstawkach. Zmontowany fragment został pokazany na **fotografii 2**.

Mechanizm

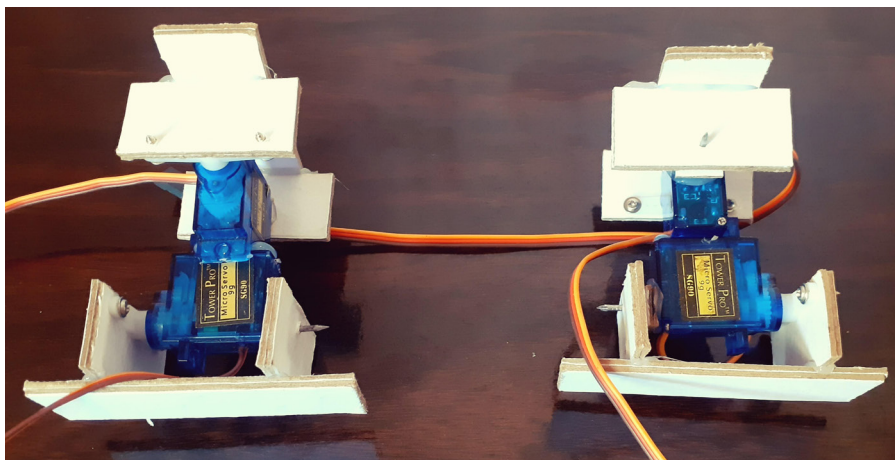
W pierwszej kolejności trzeba przygotować karton – pociąć go na fragmenty, pozwalające na zamontowanie serwo mechanizmów. Gotowe moduły z ułożonymi prostopadle serwo mechanizmami zostały pokazane na **fotografii 3**. Zestawione w ten



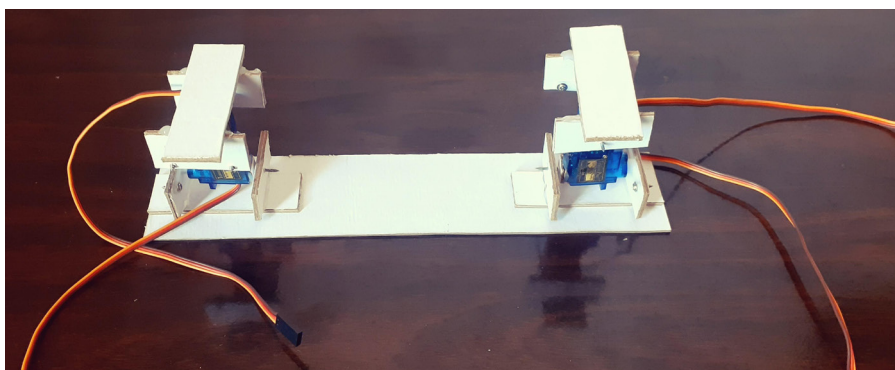
Fotografia 1. Zamontowane na podstawkach joysticki



Fotografia 2. Zmontowany kontroler z manipulatorami i elektroniką sterującą



Fotografia 3. Zmontowane serwomechanizmy



Fotografia 4. Serwomechanizmy zainstalowane na wspólnej podstawie

sposób serwa mogą niezależnie poruszać się w dwóch osiach – X i Y, którym odpowiadają dwa joysticki. Zmontowane mechanizmy umieszczone są na wspólnej podstawie, co zostało pokazane na **fotografii 4**.

Labirynt

Ostatnim elementem jest labirynt. Jego projekt w dużym stopniu zależy od pomysłu konstruktora budującego urządzenie. Może być dowolnie skomplikowany i dopasowany do naszych potrzeb i poziomu trudności, przykładowa realizacja została pokazana na **fotografii 5**. Po zmontowaniu kartonowego labiryntu należy umieścić go na zmontowanych wcześniej serwomechanizmach tak, aby zapewnić im swobodę poruszania labiryntem.

Połączenia elektryczne

Ostatnim etapem jest podłączenie poszczególnych elementów do sterującego systemem modułu Arduino. W pierwszej kolejności podłączamy serwomechanizmy od S1...S4 do pinów cyfrowych Arduino (należy pamiętać, która oś jest podłączona do którego pinu, aby odpowiednio skonfigurować to w programie).

Następnie podłączane są wyjścia joysticków. Oprócz napięcia zasilania do joysticka podłącza się wejścia analogowe modułu Arduino. Napięcie wyjściowe z modułu jest proporcjonalne do wychylenia joysticka. Cztery wyjścia z joysticków podpinamy do pinów

oznaczonych A0, A1, A2 oraz A3 (lub dowolnych innych pinów analogowych, jeśli odpowiednio zmienimy konfigurację w szkicu Arduino).

Oprogramowanie

Kod programu sterującego labiryntem został napisany w Arduino IDE. Jego główne zadanie polega na monitorowaniu pozycji joysticków, podłączonych do wejść analogowych i zgodnie z ich wychyleniem porusza serwomechanizmami. Kod programu zaprezentowano na **listingu 1**.

Do obsługi joysticków wykorzystano wejścia analogowe, ponieważ manipulatory te mają wbudowane dwa potencjometry, które sprzęgnięte są z drążkiem joysticka. Poruszając nim, zmienia się położenie suwaków potencjometrów, które są skonfigurowane

Listing 1. Kod programu sterującego labiryntem

```
#INCLUDE<SERVO.H>

// INICJALIZACJA WYJŚC SERW
SERVO s1;
SERVO s2;
SERVO s3;
SERVO s4;

// DEFINICJE PINÓW ANALOGOWYCH
INT x1 = 0; // PIN A0
INT y1 = 1; // PIN A1
INT x2 = 2; // PIN A2
INT y2 = 3; // PIN A3

INT VAL;
VOID SETUP(){
  // DEFINICJE PINÓW WYJŚCIOWYCH SERW
  s1.ATTACH(13); // PIN 13
  s2.ATTACH(12); // PIN 12
  s3.ATTACH(8); // PIN 8
  s4.ATTACH(7); // PIN 7
}

VOID LOOP(){
  // ODCZYT WARTOŚCI Z JOYSTICKA
  VAL = ANALOGREAD(x1);
  // PRZEMAPOWANIE WARTOŚCI
  VAL = MAP(VAL, 0, 1023, 135, 45);
  // RUCH SERWEM
  s1.WRITE(VAL);
  DELAY(15);

  VAL = ANALOGREAD(y1);
  VAL = MAP(VAL, 0, 1023, 135, 45);
  s2.WRITE(VAL);
  DELAY(15);

  VAL = ANALOGREAD(x1);
  VAL = MAP(VAL, 0, 1023, 135, 45);
  s2.WRITE(VAL);
  DELAY(15);

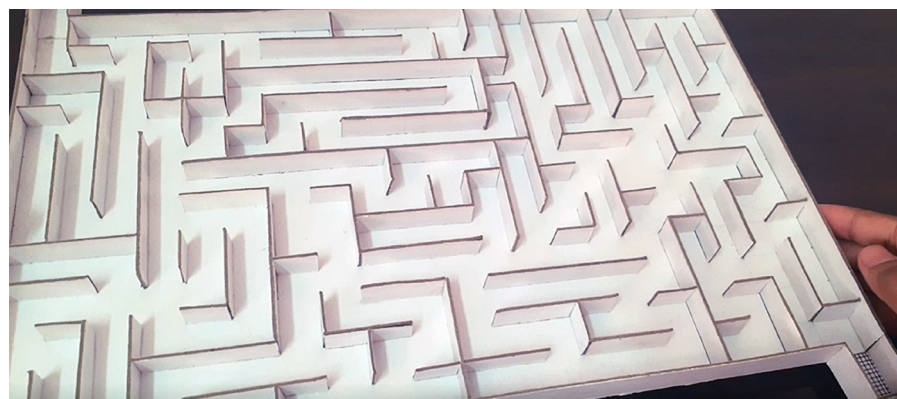
  VAL = ANALOGREAD(y1);
  VAL = MAP(VAL, 0, 1023, 135, 45);
  s2.WRITE(VAL);
  DELAY(15);
}
```

jako regulowane dzielniki napięcia – na zewnętrzne nóżki potencjometrów podawane jest napięcie i masa, a suwaki są wyjściami dzielników. W ten sposób zmieniając pozycję manipulatora, zmienia się napięcia wyjściowe z joysticka, proporcjonalnie do wychylenia.

Każdy joystick ma dwa wyjścia napięciowe – jedno dla osi X, a jedno Y. Napięcia są monitorowane przez przetwornik analogowo-cyfrowy w mikrokontrolerze, a następnie służą do sterowania położeniem serwomechanizmów. Aby przekonwertować pomiar z ADC (w zakresie od 0 do 1023, ponieważ przetwornik ma 10 bitów) na kąt, który ma być ustawiony na serwomechanizmie (od 45 do 135 stopni), używana jest funkcja *map*.

Nikodem Czechowski, EP

Źródło: <http://bit.ly/3t0yuVQ>



Fotografia 5. Labirynt