

# MASZYNA SORTUJĄCA KOLOROWE CUKIERKI

SPECYFIKACJA SYSTEMU WBUDOWANEGO

KACPER SMYCZYK

PAWEŁ ZELENT-WITOWSKI

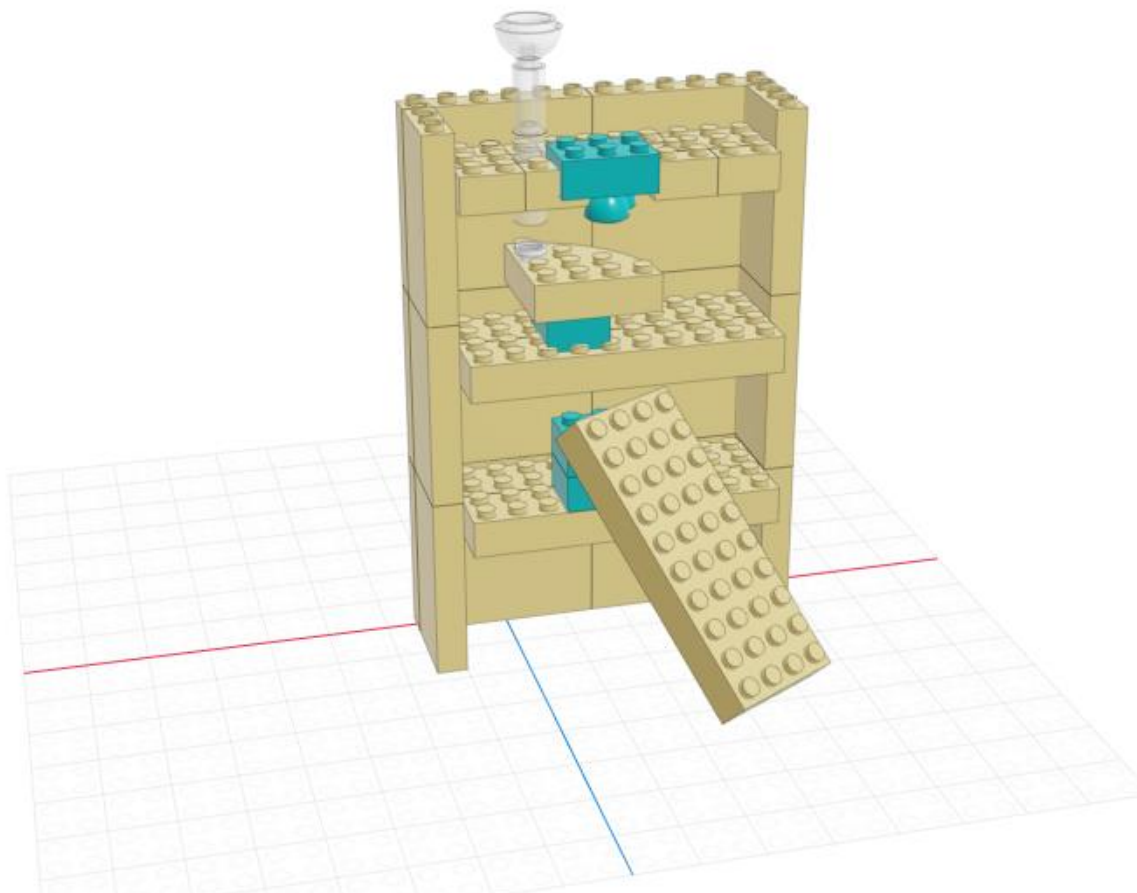
SEMESTR LETNI 2019

## OPIS URZĄDZENIA

Głównymi elementami systemu są:

- ruchomy podajnik,
- sensor kolorów,
- ruchomy odbiornik

Celem urządzenia jest sortowanie małych cukierków ze względu na ich kolor. Po włączeniu urządzenia cukierki załadowane w górnej tubce zaczynają pojedynczo opadać na przymocowaną do górnego serwomechanizmu platformę i czekają aż zwolni się na niej miejsce (poprzedni zostanie posortowany). Następnie serwomechanizm obraca platformę i podstawia cukierek pod sensor sprawdzający jego kolor. Po zbadaniu koloru dolny serwomechanizm obraca odbiornik podstawiając go pod odpowiedni pojemnik, a górny serwomechanizm obraca platformę do miejsca zrzutu cukierka na odbiornik po czym wraca do stanu początkowego.



Niebieskie elementy od góry: sensor, górne serwo, dolne serwo.

## KOMPONENTY SYSTEMU

### ARDUINO NANO



Główny komponent systemu odpowiedzialny za zbieranie danych z sensora kolorów i sterowanie silnikami.

Płytkę o rozmiarze: 45 mm x 18 mm zawiera mikrokontroler ATmega328 wyposażony w 22 cyfrowe wejścia/wyjścia z czego 6 można wykorzystać jako wyjścia PWM (np. do sterowania silnikami) oraz 8 jako wejścia analogowe. Układ taktowany jest sygnałem zegarowym o częstotliwości 16 MHz, posiada 32 kB pamięci programu Flash oraz 2 kB pamięci operacyjnej SRAM.

## WAVESHARE TCS3200



Komponent systemu odpowiedzialny za pomiar koloru.

Moduł ten umożliwia pomiar koloru jako składowe RGB, a jego wyjściem jest natężenie światła danej barwy. Czujnik ten można podłączyć do dowolnego zestawu uruchomieniowego z mikrokontrolerem wyposażonym w licznik z wejściem częstotliwościowym (kompatybilny z Arduino).

## SERVO TOWERPRO SG-90 - MICRO



Komponent systemu odpowiedzialny za sterowanie pozycją podajnika oraz odbiornika. Lekkie, 9-gramowe serwo o zakresie 180°.

Parametry dla napięcia 4,8 V:

- Moment: 1,8 kg\*cm (0,18 Nm)
- Prędkość: 0,1 s/60°

Wymiary urządzenia: 22 x 11,5 x 27 mm.

## WYŁĄCZNIK TYPU ON/OFF



### Specyfikacja:

- Prąd: 6A Napięcie: 230V
- 18 x 10 mm

## GNIAZDO MONTAŻOWE JACK 3,5MM



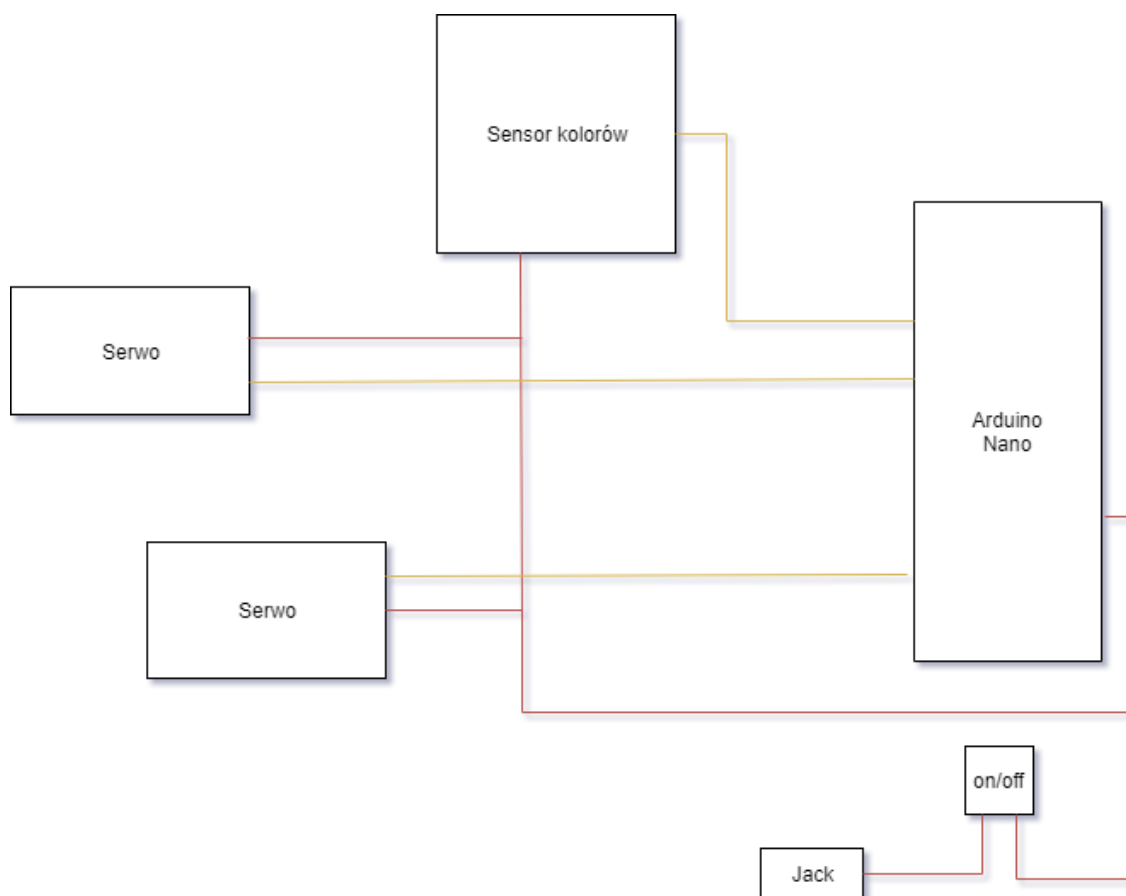
## ZASILACZ IMPULSOWY



### Specyfikacja:

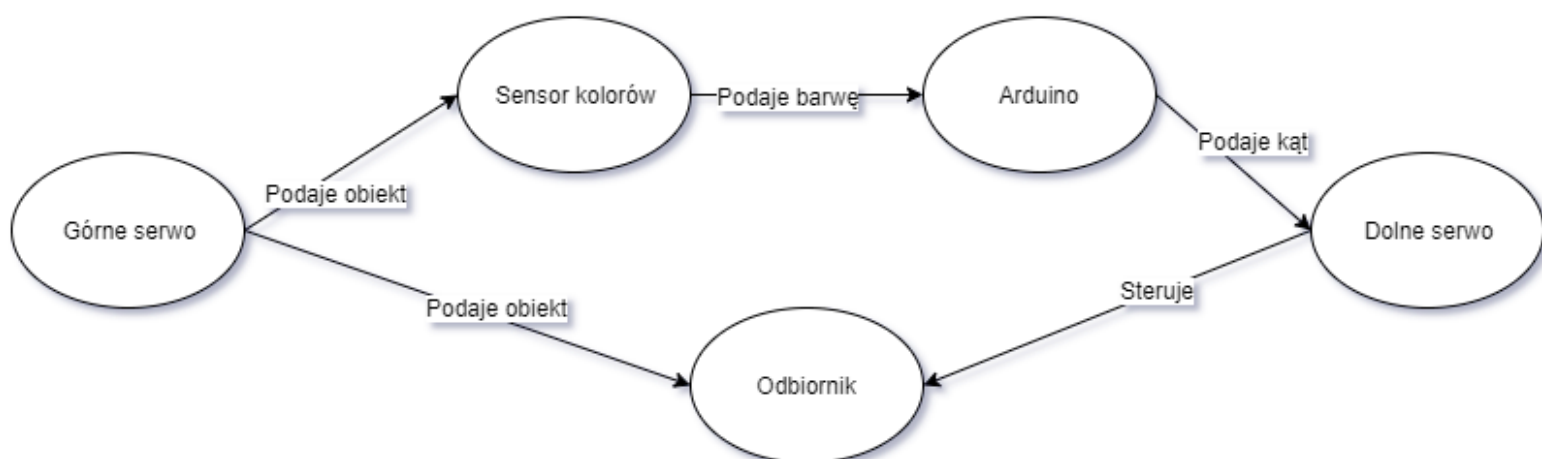
- Napięcie zasilania: 100 do 240 V AC
- Napięcie wyjściowe: 5 V DC
- Prąd wyjściowy: 2 A
- Złącze: wtyk DC 3,5 / 1,5 mm

## DIAGRAMY



Na powyższym diagramie w prosty sposób zostało przedstawione połączenie elementów ze sobą. Widać, że wszystkie są połączone z płytką Arduino (**żółta linia**). Można także zauważyć, że układ jest zasilany napięciem (**czerwona linia**), którego wymagają wszystkie jego elementy.

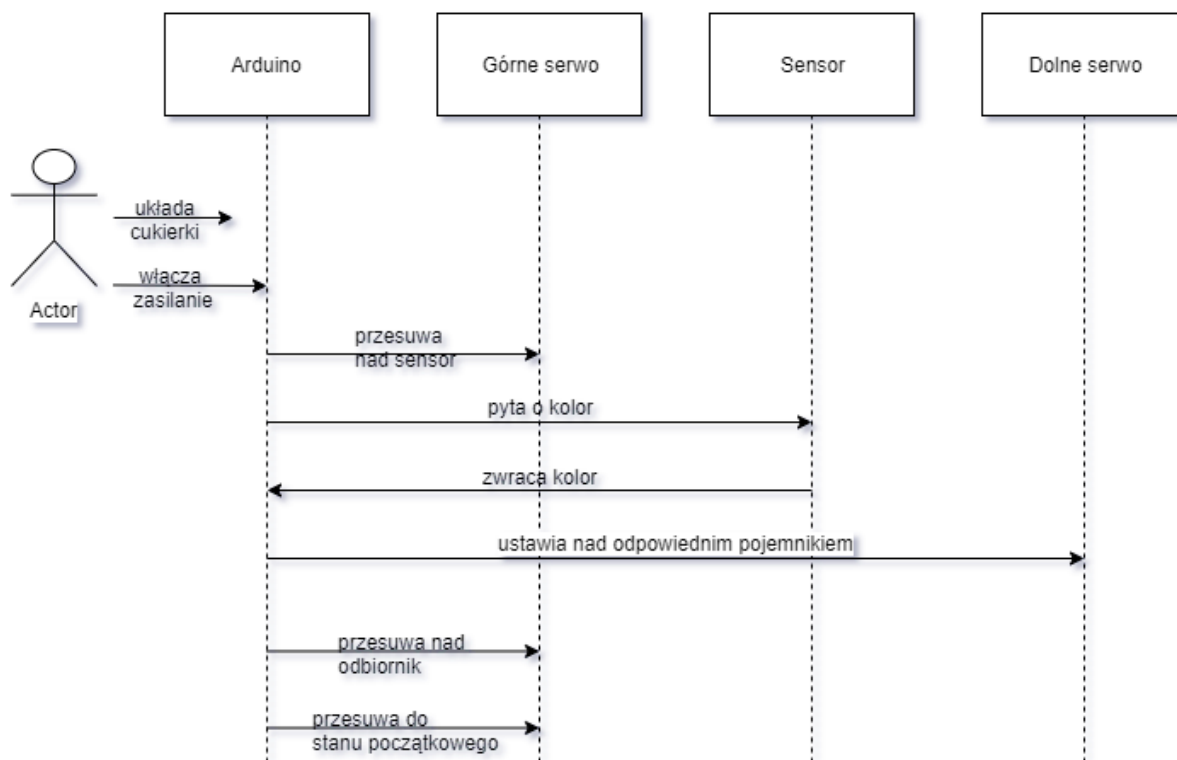
Poniższy diagram przedstawia natomiast współpracę urządzeń.



Poniższy diagram stanów reprezentuje logikę działania systemu.



Poniższy diagram interakcji przedstawia sposób interakcji użytkownika z systemem oraz działanie systemu w czasie. Po przesunięciu górnego serwo do stanu początkowego diagram zapętla się aż do odłączenia przez użytkownika zasilania.



## PSEUDOKOD

Nasz program działa w nieskończonej pętli, która ciągle sprawdza kolor i obsługuje górne oraz dolne serwo.

W naszym programie oprócz oczywistych zmiennych typu `Servo` oraz odpowiednio zdefiniowanych pinów dla naszych urządzeń należy także zdefiniować zmienne typu `int` dla częstotliwości dostarczonych z sensora (`int frequency`) oraz rodzaju koloru.

```
1. while true:
2.     topServo.write(startPosition);
3.
4.     topServo.write(sensorPosition);
5.
6.     colour = readColour();
7.
8.     switch(colour){
9.         case colourType:
10.            bottomServo.write(colourTypePosition);
11.            break;
12.
13.         case 0:
14.            break;
15.     }
16.
17.     topServo.write(dropPosition);
18.
19.     topServo.write(startPosition);
20.
21.     colour = 0;
22. end while
```

Sprawdzanie koloru odbywa się w osobnej funkcji `readColour()`.

```
1. int readColour(){
2.     /*
3.     Sensor's Digital Pin values for various photodiodes:
4.     RED - LOW LOW
5.     GREEN - HIGH HIGH
6.     BLUE - LOW HIGH
7.     */
8.     digitalWrite(sensorPin1, HIGH);
9.     digitalWrite(sensorPin2, HIGH);
10.
11.     frequency = pulseIn(sensorOut, LOW);
12.     int R = frequency;
```

W funkcji zainicjowane są zmienne R, G oraz B przechowujące częstotliwości dla odczytów odpowiednich fotodiod(czerwona, zielona, niebieska).



Zbieranie częstotliwości należy powtórzyć także dla zmiennych *G* oraz *B* zmieniając wartości na odpowiednich pinach.

```
1. if(statement on R,G,B values){  
2.     return colourType;  
3. }
```

Następnie zmienne *R*, *G* oraz *B* podawane są testowi za pomocą bloków *if()* w celu ustalenia koloru.

Należy także pamiętać o ustawieniu odpowiednich odstępów czasowych pomiędzy akcjami każdego z urządzeń za pomocą funkcji *delay()* w celu zapewnienia poprawnego funkcjonowania całego systemu.

## KOSZTORYS

Poniższa tabela przedstawia informacje o cenie całego systemu. Podane ceny są w złotych.

Produkt	Cena
Arduino Nano	95,00
Waveshare TCS3200	23,50
Serwo TowerPro SG-90 - micro	9,90
Serwo TowerPro SG-90 - micro	9,90
Wyłącznik ON/OFF	3,95
Gniazdo JACK 3,5mm stereo montażowe	3,50
Zasilacz impulsowy	16,90
<b>Razem</b>	<b>162,65</b>