



**POLITECHNIKA
RZESZOWSKA**

im. IGNACEGO ŁUKASIEWICZA



**WYDZIAŁ
BUDOWY MASZYN
I LOTNICTWA**
POLITECHNIKI RZESZOWSKIEJ

PROJEKT INŻYNIERSKI

System do monitoringu obiektów w czasie rzeczywistym

Autor: Łukasz Undziakiewicz

Koordynator przedmiotu: dr inż. Jacek S. Tutak

Forma i stopień studiów: I stopnia, stacjonarne

Kierunek studiów: Mechatronika

Rok akademicki: 2022/2023

Profil studiów: ogólnoakademicki

Cel i zakres pracy

Celem pracy było zbudowanie kamery monitorującej i śledzącej obiekty w czasie rzeczywistym, która będzie miała możliwość wyboru śledzenia jednego z kilku obiektów. System ma umożliwiać śledzenie kilku obiektów w tym samym czasie. Za napędzanie silników mają odpowiadać dwa silniki krokowe, dzięki którym obroty kamery będą płynne, co jest bardzo ważne w tego typu urządzeniach. Obudowa systemu ma być wykonana przy użyciu druku 3d.

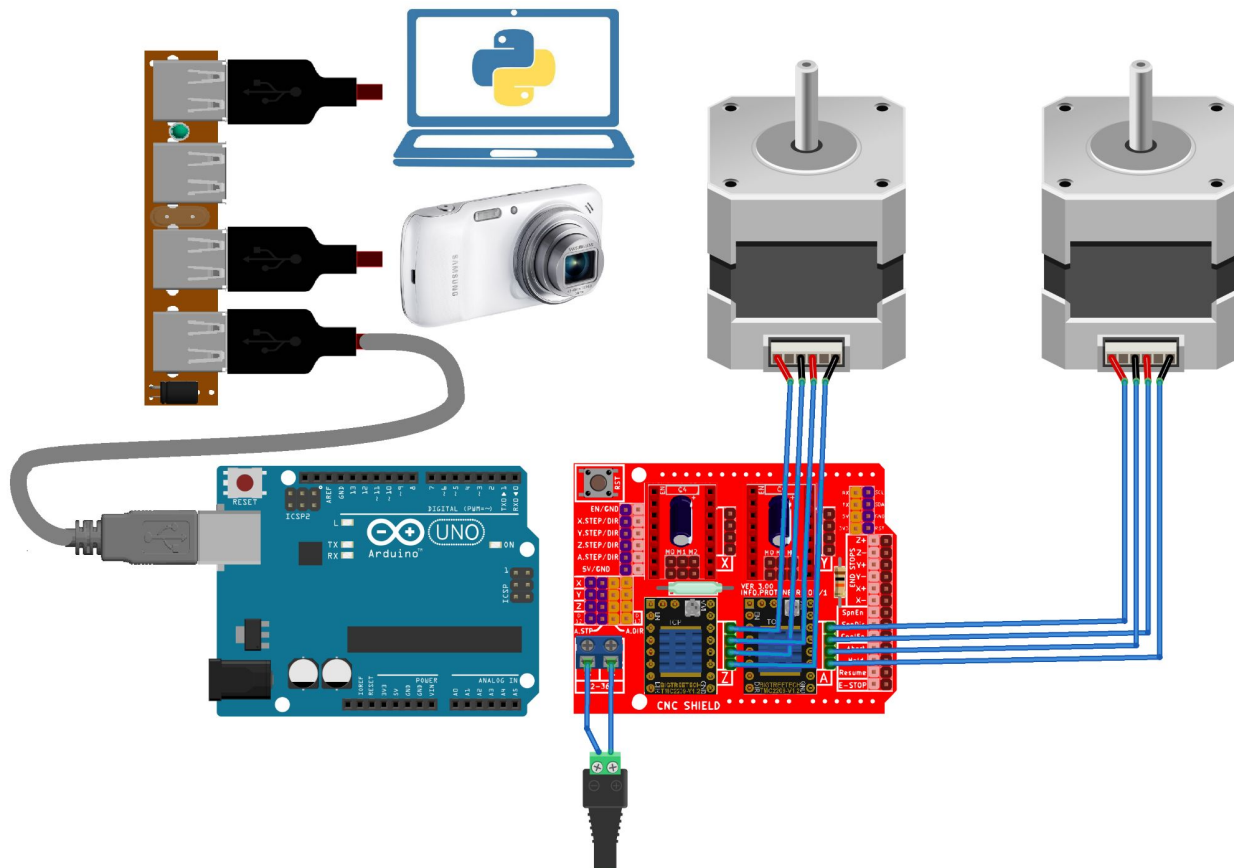
Przegląd istniejących rozwiązań



Filmed with PIX4TEAM



Dobór elektroniki



Przedstawienie kilku fragmetów kodu w Arduino Ide

```
#include <TMC2208Stepper.h> // Biblioteka służąca do obsługi sterowników silników krokowych TMC2208
```

```
#define EN_PIN 8 // LOW: Silnik włączony. HIGH: Silnik wyłączony
```

```
#define STEPx_PIN 2 // Pin sterujący krokiem dla osi x
```

```
#define DIRx_PIN 5 // Pin sterujący kierunkiem ruchu dla osi x
```

```
#define STEPy_PIN 3 // Pin sterujący krokiem dla osi y
```

```
#define DIRy_PIN 6 // Pin sterujący kierunkiem ruchu dla osi y
```

```
if (Serial.available() > 0) //Jeśli dostępne są dane do odczytu z portu szeregowego, wykonaj kod wewnątrz bloku if
```

```
if (x_mid > width / 2 + 30)
```

```
{
```

```
digitalWrite(EN_PIN, LOW); // Włącz silnik
```

```
digitalWrite(DIRx_PIN, LOW); // Ustaw kierunek obrotu silnika na lewo
```

```
digitalWrite(STEPx_PIN, !digitalRead(STEPx_PIN)); // Obróć silnik o jeden krok
```

```
delayMicroseconds(2); // Zatrzymaj się na chwilę
```

```
}
```

Przedstawienie kilku fragemntów kodu w języku python

```
# importowanie niezbędnych bibliotek
```

```
import numpy as np
```

```
import cv2
```

```
import serial,time
```

```
import pytesseract
```

```
# zainicjalizowanie HOG descriptor/detektor osób
```

```
hog = cv2.HOGDescriptor()
```

```
hog.setSVMDetector(cv2.HOGDescriptor_getDefaultPeopleDetector())
```

```
cap = cv2.VideoCapture(0)
```

```
ArduinoSerial=serial.Serial('com10',9600,timeout=0.1)
```

```
#tworzenie zmiennej string
```

```
string='X{0:d}Y{1:d}'.format(centroid_X, centroid_Y)
```

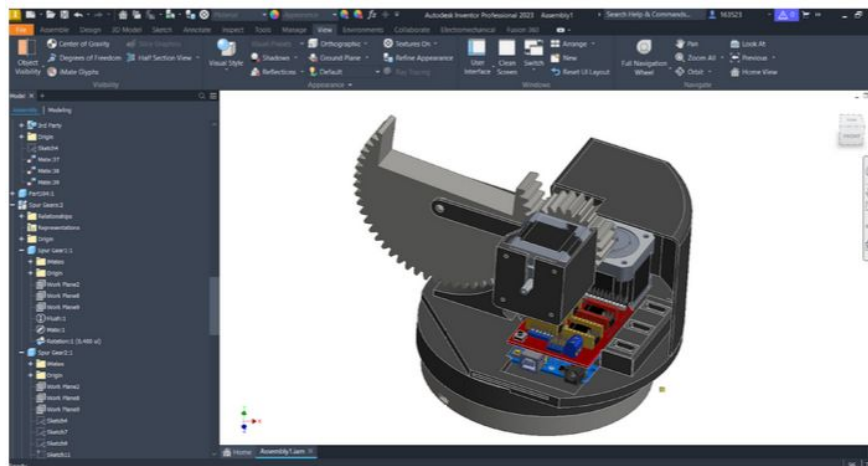
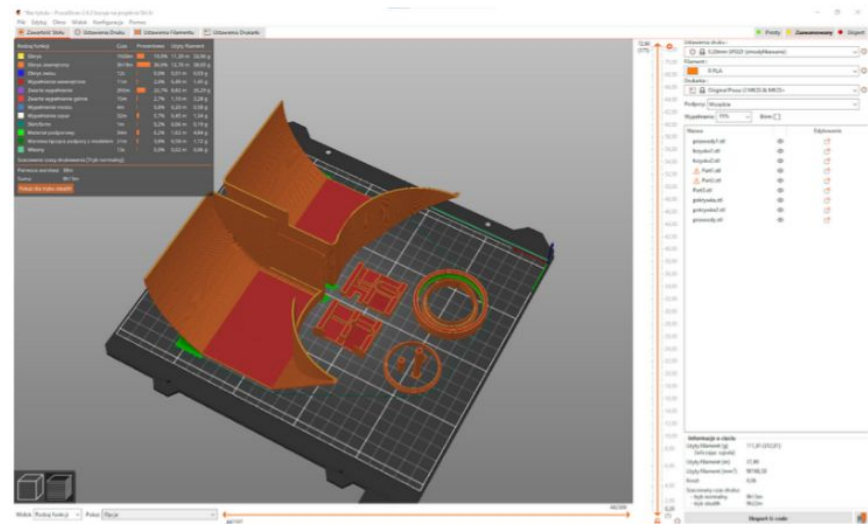
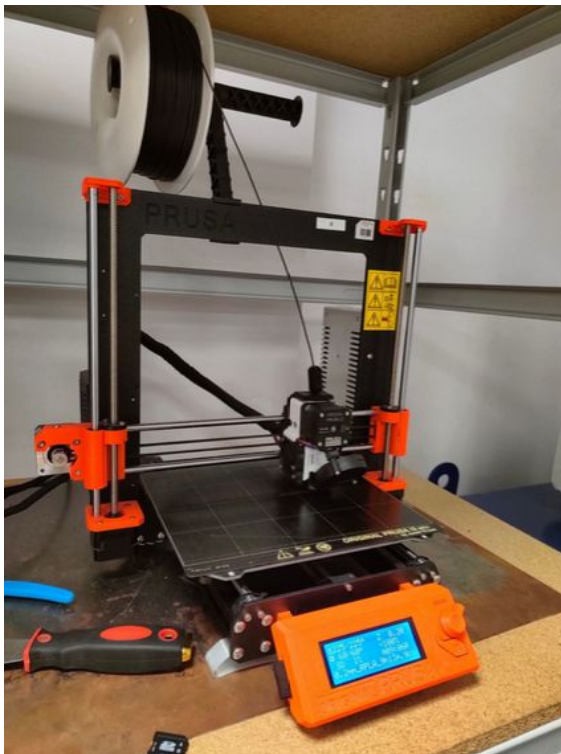
```
#wyświetlenie zawartości string
```

```
print(string)
```

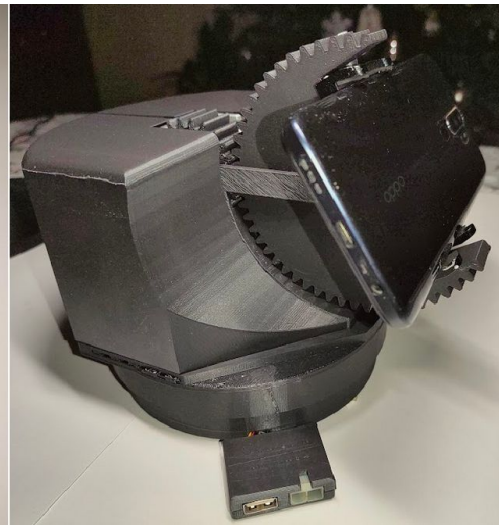
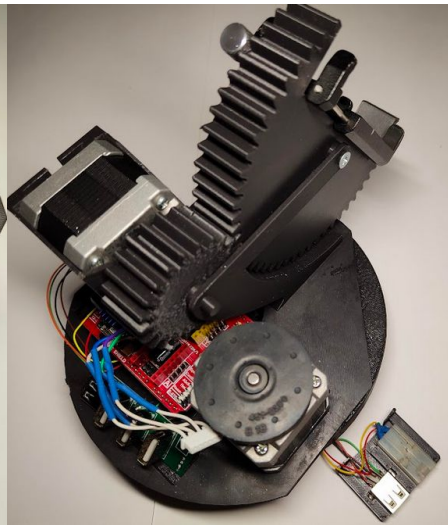
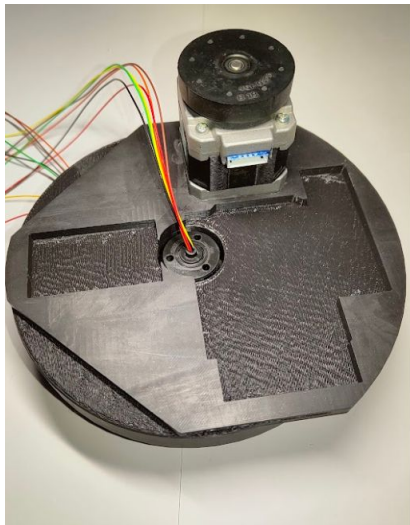
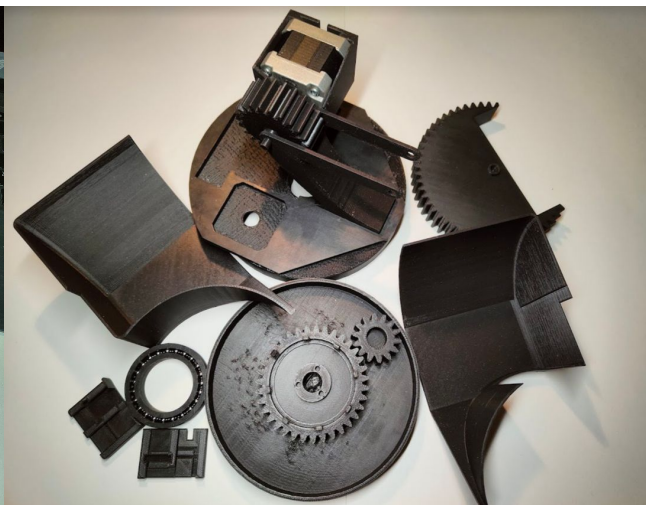
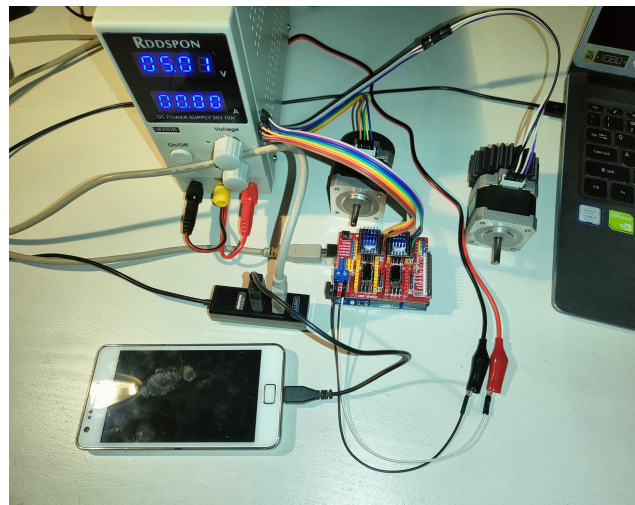
```
#wysłanie zawartości do Arduino
```

```
ArduinoSerial.write(string.encode('utf-8'))
```

Projektowanie Prototypu



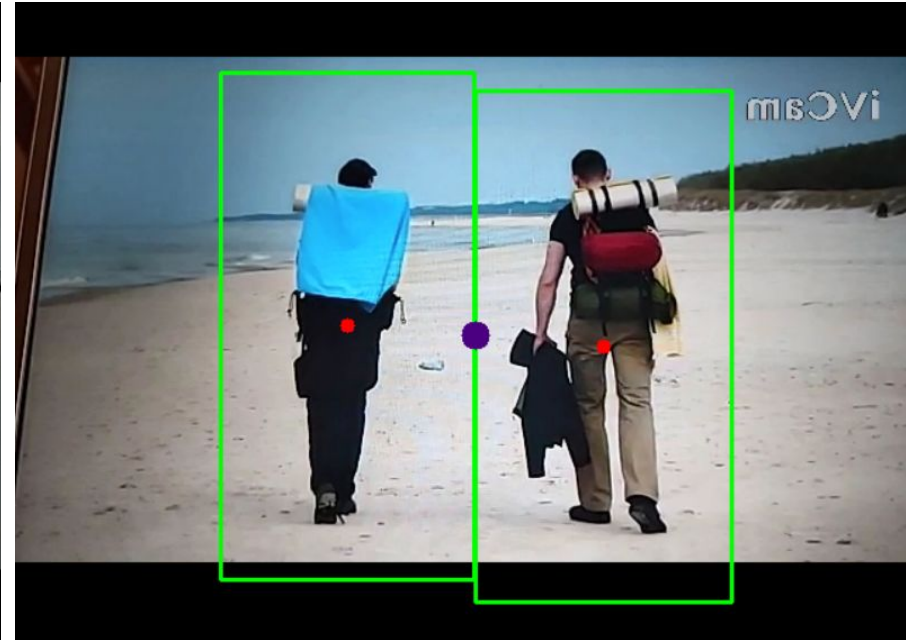
Złożenie całości



Testy i ocena poprawności działania

X574Y139
X570Y156
X569Y159
X575Y156
X574Y160
X585Y155
X581Y154
X582Y155
X583Y155

frame



Bibliografia

- [1] <https://orllo.pl/kamera-obrotowa-zewnetrzna-wifi-ip-orllo-z16.html>. (dostęp 13.01.2022).
- [2] <https://presentation.aver.com/model/ptc310hvv2> (dostęp 13.01.2022).
- [3] <https://shop.movensee.com/en/content/31-pix4team-auto-follow-camera-for-team-sports> (dostęp 13.01.2022).

Dziękuję za uwagę