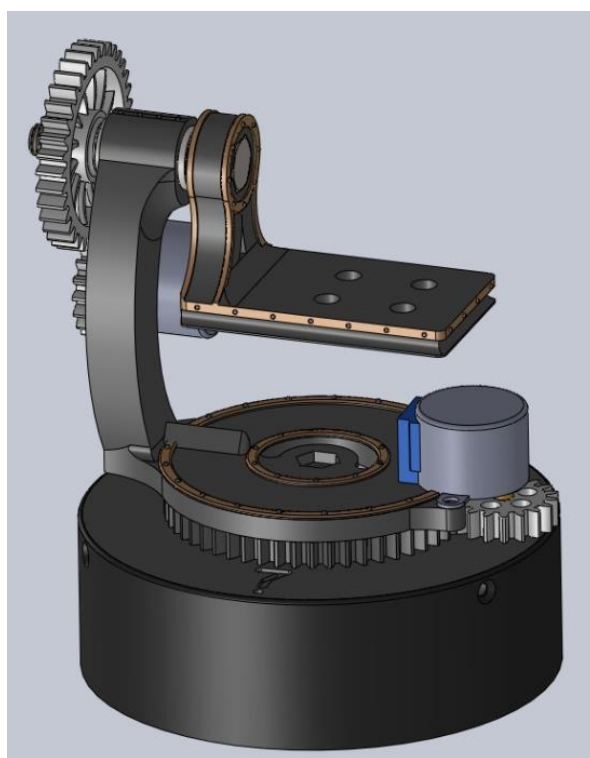


Wydział Elektrotechniki i Informatyki Politechniki Lubelskiej	Laboratorium z przedmiotu Podstawy Robotyki		
Skład osobowy grupy laboratoryjnej GR1: Vasyl Tytiuchenko Volodymyr Sharonin	Rok akademicki: 2022/2023 Semestr: zimowy	Kierunek studiów: PSI Grupa: 7.1	Stanowisko: Pierwsze
Temat laboratorium: IMPLEMENTACJA MODELU ROBOTA NA PLATFORMIE ARDUINO.		Data wykonania: 05.12.2022	Podpis:

Cel laboratorium: zapoznać się z metodami implementacji robota sterowanego przez arduino , oprogramowanie, kinematyka prosta tego robota i technologie użyte przy jego wyrobieniu.

Wybór typu robota do zaimplementowania

Podczas wyboru robota nasza grupa podjęła decyzję o wyborze nie standardowego robota (że to nie jakieś proste ramiennie czy złożony robot ręka). Czymś takim okazał się robot o dwóch stopniach swobody, przeznaczony do zainstalowania np. małej kamery do wideo nadzoru czy jakiegoś czujnika, na silnikach 28BYJ-48 5V.



Rysunek 1 wygląd robota z środowiska programu 3D Builder

Wzięty ten schemat był ze strony <https://www.thingiverse.com/thing:1154361>

Program w języku C do sterowania robotem

```
int i1 = 0;
int i2 = 0;
int t = 0;

int t12 = 14; //odpowiada za prędkość obrotów pierwszego silnika przy małym
dchyleniu dżojstika
int t11 = 4;   //odpowiada za prędkość obrotów pierwszego silnika przy dużym
dchyleniu dżojstika
int t22 = 14; //odpowiada za prędkość obrotów drugiego silnika przy małym
dchyleniu dżojstika
int t21 = 4;   //odpowiada za prędkość obrotów drugiego silnika przy dużym
dchyleniu dżojstika

void setup() {
    //4,5,6,7 kontakty do pierwszego silnika
    pinMode(4, OUTPUT);
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    //13,12,11,10 kontakty do drugiego silnika
    pinMode(13, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(11, OUTPUT);
    pinMode(10, OUTPUT);
}

void prodlit11() { //funkcja dla dzielenia cyklu
    if (analogRead (A0) < (520)) {
        i1++;
    }
    if (analogRead (A0) > (521) && analogRead (A0) < (999)) {
        i1--;
        t = t12;
    }
    if (analogRead (A0) > (1000)) {
        i1--;
        t = t11;
    }
}

void prodlit12() { //funkcja dla dzielenia cyklu
    if (analogRead (A0) > (490)) {
        i1++;
    }
    if (analogRead (A0) > (51) && analogRead (A0) < (489)) {
        i1--;
        t = 12;
    }
    if (analogRead (A0) < (50)) {
        i1--;
        t = t11;
    }
}
```

```

void prodlit21() { //funkcja dla dzielenia cyklu
    if (analogRead (A1) < (520)) {
        i2++;
    }
    if (analogRead (A1) > (521) && analogRead (A1) < (999)) {
        i2--;
        t = t22;
    }
    if (analogRead (A1) > (1000)) {
        i2--;
        t = t21;
    }
}

void prodlit22() { //funkcja dla dzielenia cyklu
    if (analogRead (A1) > (490)) {
        i2++;
    }
    if (analogRead (A1) > (51) && analogRead (A1) < (489)) {
        i2--;
        t = t22;
    }
    if (analogRead (A1) < (50)) {
        i2--;
        t = t21;
    }
}

void stopPins() {
    digitalWrite(4, LOW); digitalWrite(5, LOW); digitalWrite(6, LOW);
    digitalWrite(7, LOW); digitalWrite(10, LOW); digitalWrite(11, LOW);
    digitalWrite(12, LOW); digitalWrite(13, LOW); delay(100);
}

void loop() {
    if (analogRead (A0) > (521)) {
        t = t12;
        while (i1 < 1) { //cykl dla obracania wału pierwszego silnika wprawo
            digitalWrite(7, HIGH); digitalWrite(5, LOW); delay(t);
            digitalWrite(6, HIGH); digitalWrite(4, LOW); delay(t);
            digitalWrite(5, HIGH); digitalWrite(7, LOW); delay(t);
            digitalWrite(4, HIGH); digitalWrite(6, LOW); delay(t - 2);
            i1++; prodlit11();
        } i1 = 0; stopPins();
    }

    if (analogRead (A0) < (489)) {
        t = t12;
        while (i1 < 1) { //cykl dla obracania wału pierwszego silnika wlewo
            digitalWrite(4, HIGH); digitalWrite(6, LOW); delay(t);
            digitalWrite(5, HIGH); digitalWrite(7, LOW); delay(t);
            digitalWrite(6, HIGH); digitalWrite(4, LOW); delay(t);
            digitalWrite(7, HIGH); digitalWrite(5, LOW); delay(t - 2);
            i1++; prodlit12();
        } i1 = 0; stopPins();
    }
}

```

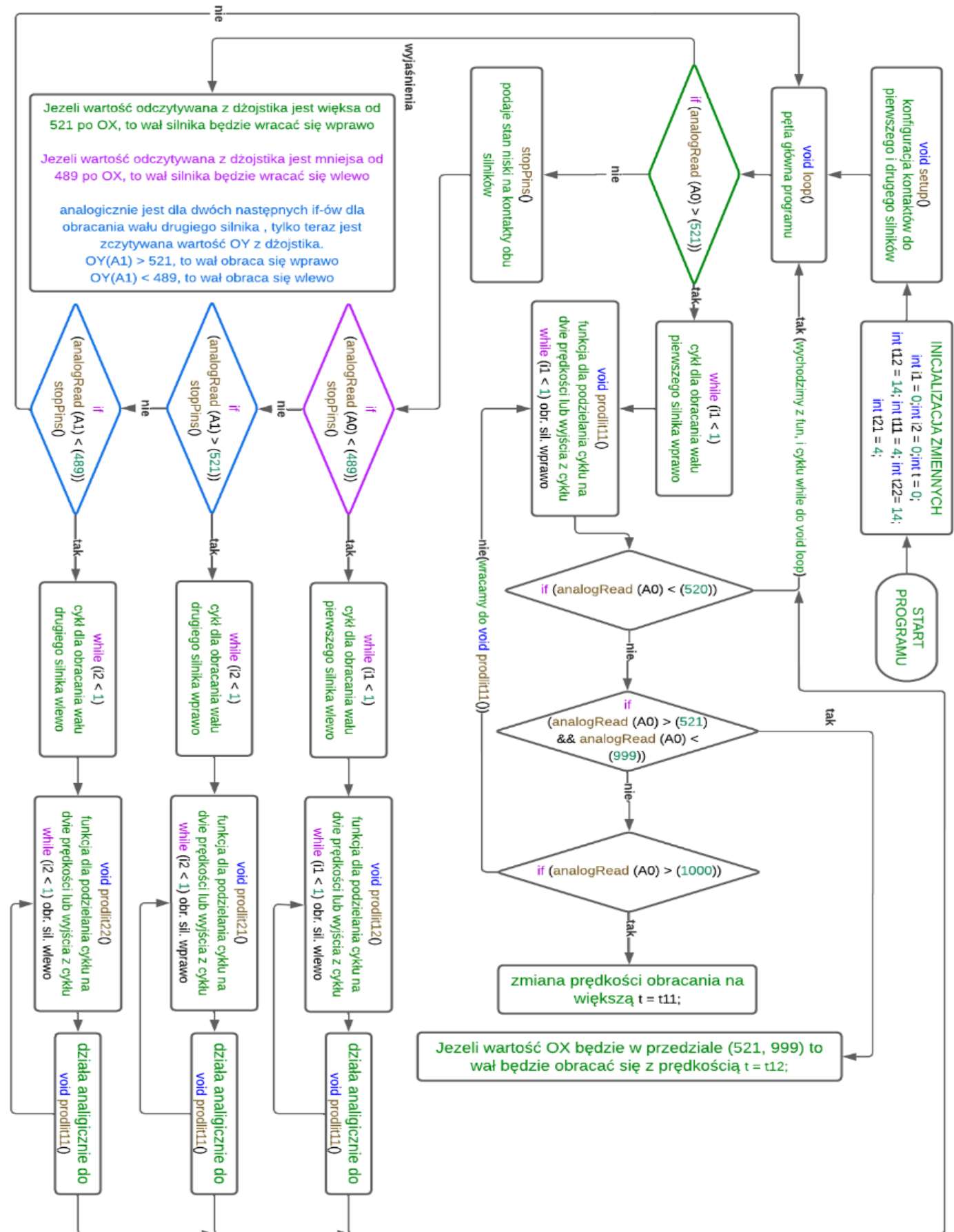
```

if (analogRead (A1) > (521)) {
    t = t22;
    while (i2 < 1) { //cykł dla obracania wału drugiego silnika wprawo
        digitalWrite(13, HIGH); digitalWrite(11, LOW); delay(t);
        digitalWrite(12, HIGH); digitalWrite(10, LOW); delay(t);
        digitalWrite(11, HIGH); digitalWrite(13, LOW); delay(t);
        digitalWrite(10, HIGH); digitalWrite(12, LOW); delay(t - 2);
        i2++; prodLit21();
    } i2 = 0; stopPins();
}

if (analogRead (A1) < (489)) {
    t = t22;
    while (i2 < 1) { //cykł dla obracania wału drugiego silnika wlewo
        digitalWrite(10, HIGH); digitalWrite(12, LOW); delay(t);
        digitalWrite(11, HIGH); digitalWrite(13, LOW); delay(t);
        digitalWrite(12, HIGH); digitalWrite(10, LOW); delay(t);
        digitalWrite(13, HIGH); digitalWrite(11, LOW); delay(t - 2);
        i2++; prodLit22();
    } i2 = 0; stopPins();
}
}

```

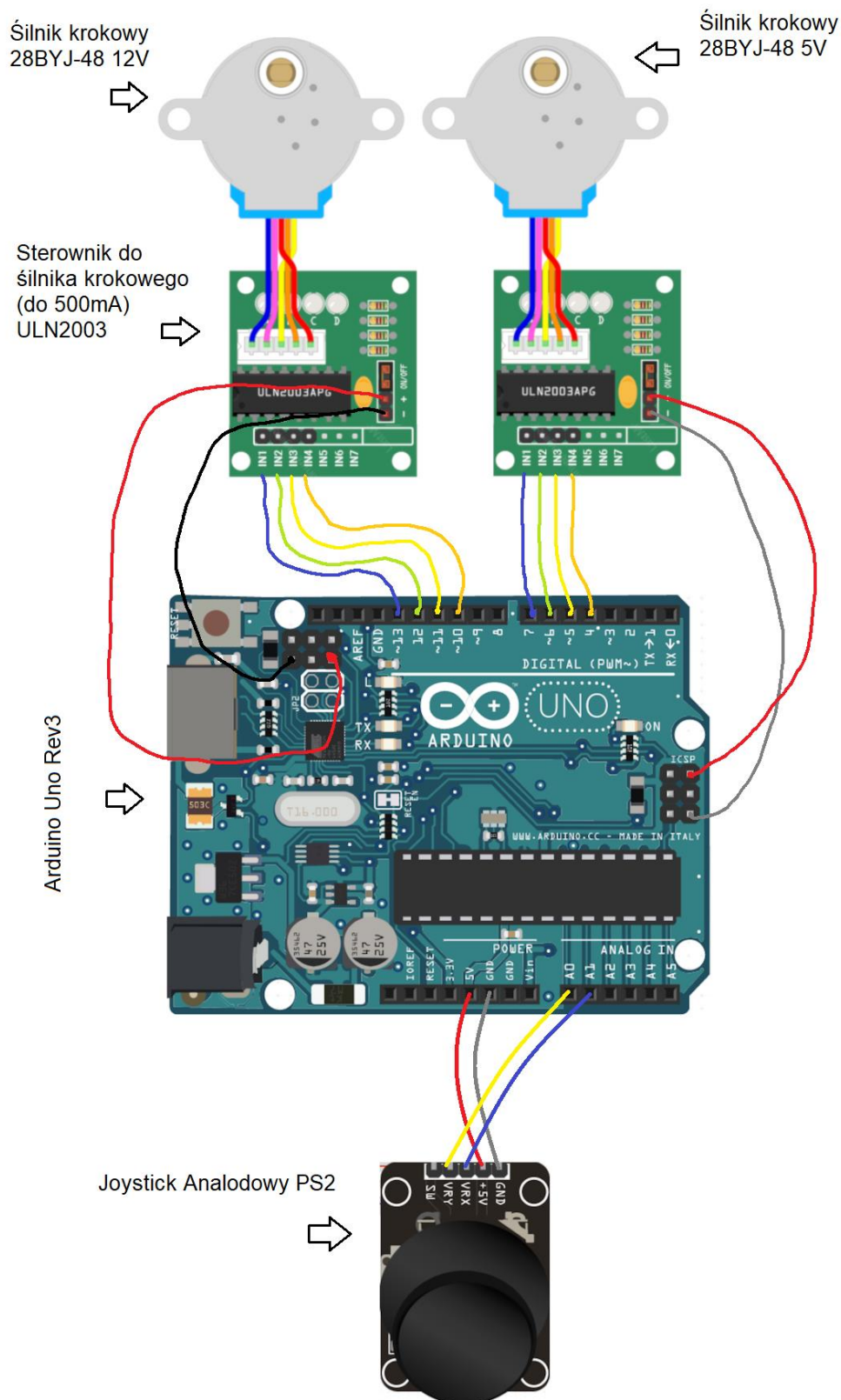
Schemat blokowy działania programu robota



Rysunek 2 Schemat blokowy programu

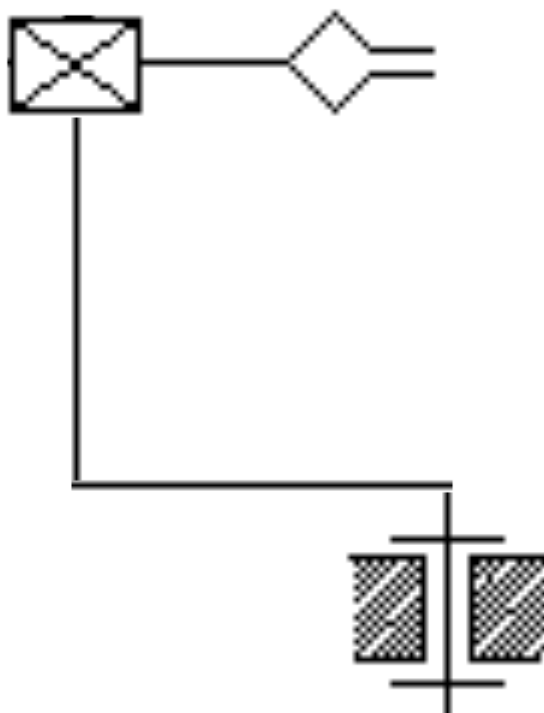
Schemat blokowy zestaw trochę mniej rozbudowany przez ograniczenia serwisu w którym był stworzony, do niektórych bloków po prostu zostały dodane nazwy funkcji, do których oni musieli być podłączone. A także funkcje które były podobne za swoim działaniem zostały zaznaczone jak „działa analogicznie do...”.

Schemat podłączenia i sterowania robotem

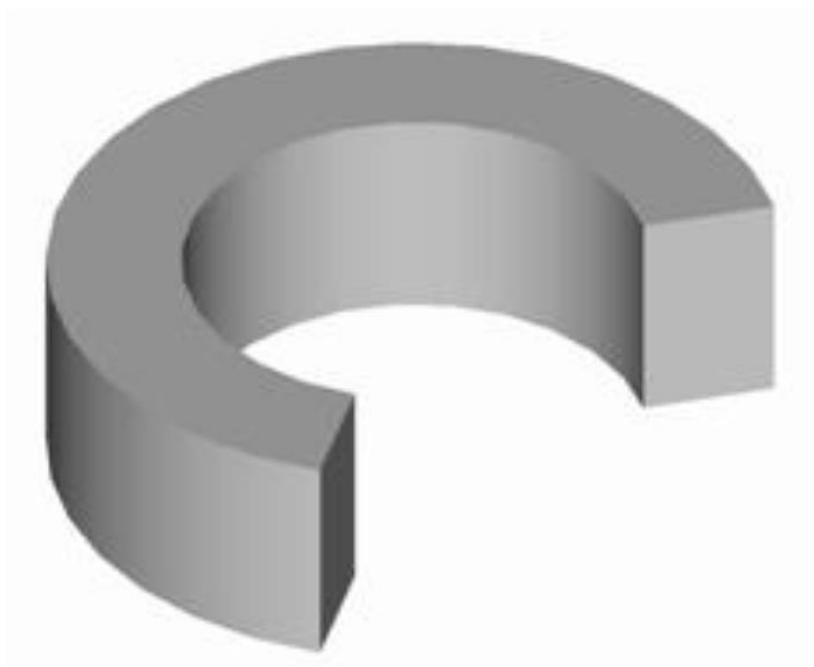


Rysunek 3 Schemat podłączenia i sterowania robotem

Schemat kinematyki prostej robota

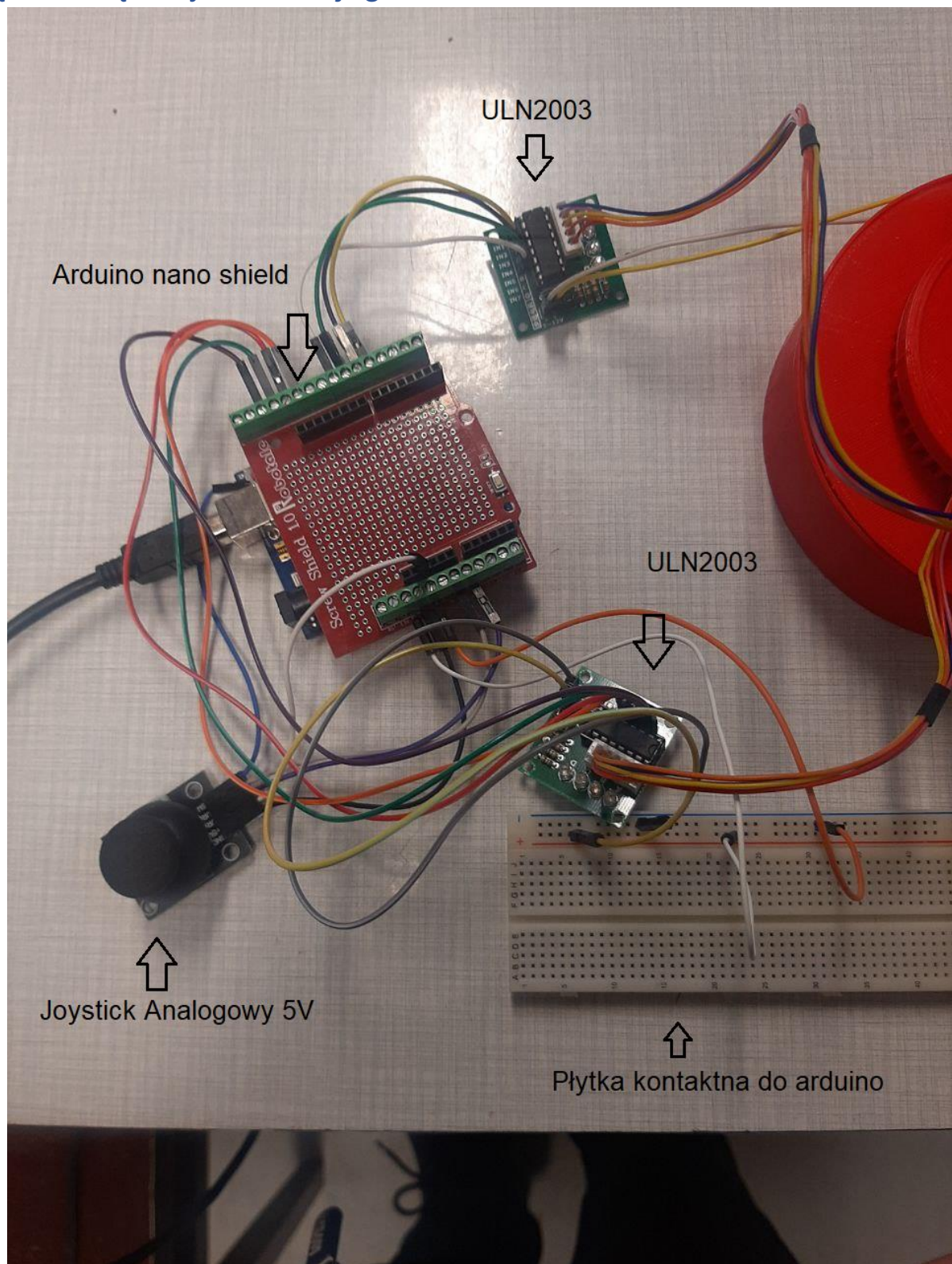


Rysunek 4 Schemat robota o dwóch stopniach swobody

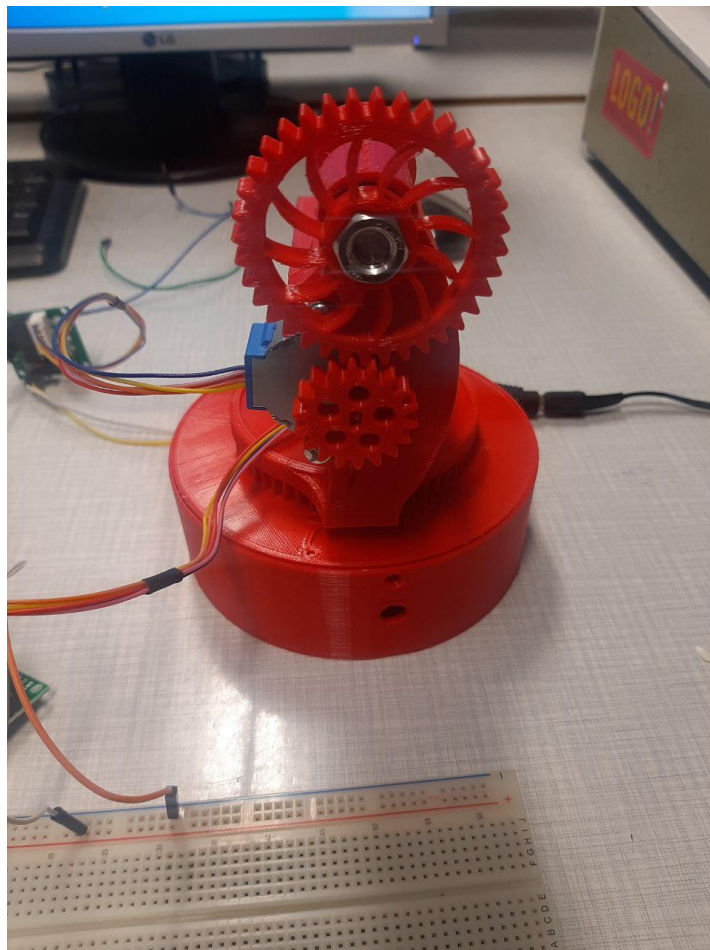


Rysunek 5 Główna przestrzeń robocza robota

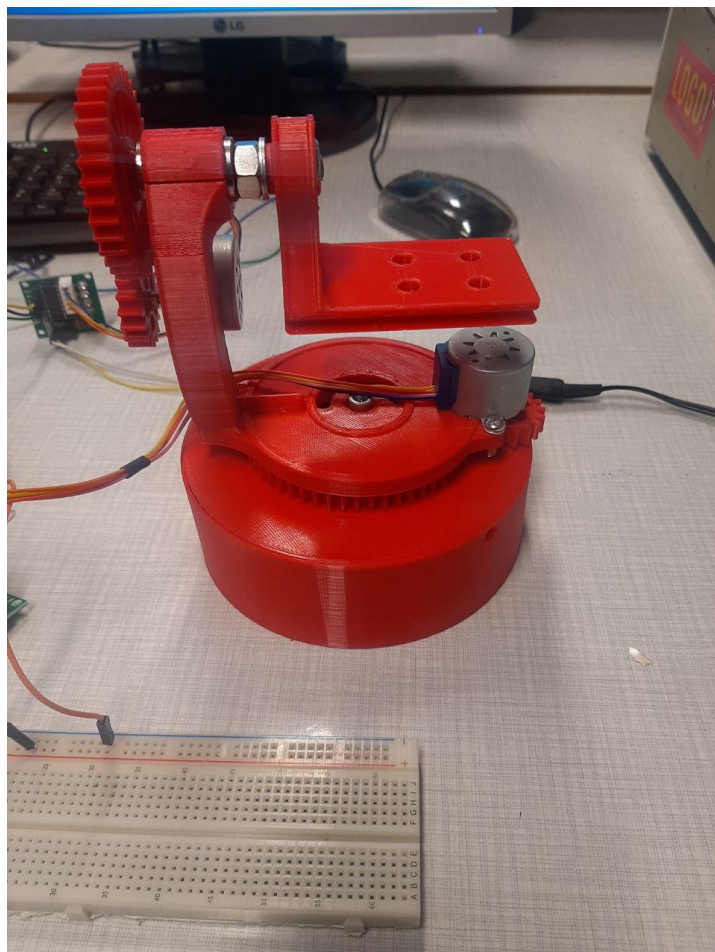
Wygląd zewnętrzny robota i jego działanie



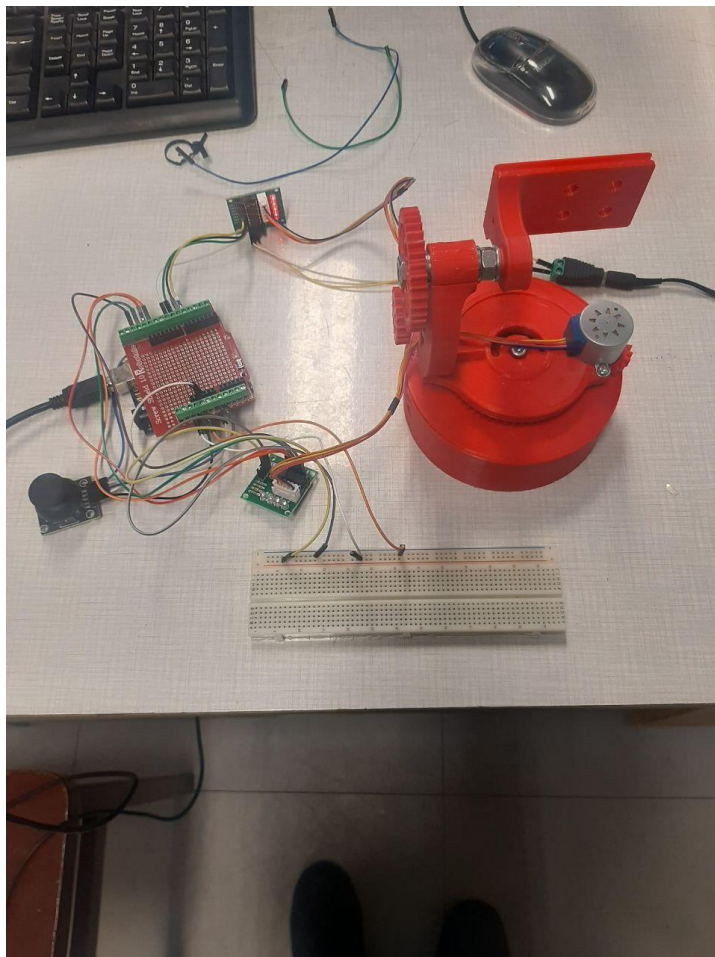
Rysunek 6 Rzeczywiste podłączenie do arduino



Rysunek 7 Wygląd robota po wydrukowaniu na 3d drukarce i złożeniu go do kupy



Rysunek 8 Wygląd robota z boku



Rysunek 9 Sprawdzenie pracy robota

Wnioski

Z wypróbowanymi okazało się że silników w 5 V nie wystarcza dla normalnego obracania elementów robota, dlatego że czasami w niektórych pozycjach nim było ciężko obracać się (nie wystarczało mocy). To chyba jedyny problem jaki wyniknął podczas jego pracy, wszystkie inne jego zespoły działały normalnie. Do systemu sterowania chyba nie wystarcza tylko jakiegoś modułu do zdalnego sterowania i akumulatora w 5V (12V), żeby można było sterować robotem zdalnie.

Schemat podłączenia jest trochę inny niż podłączenie w rzeczywistości, dlatego że nie szukamy łatwych szlaków, ale ogólnie można było by zrobić podłączenie jak na rysunku 3.