




Politechnika Wrocławska

Wydział Elektroniki,
Fotoniki i Mikrosystemów 

Laboratorium Optoelektroniki i Fotoniki

Politechnika Wrocławska

Katedra Metrologii Elektronicznej i Fotonicznej

Tracker Optyczny

Optoelektronika 2 – projekt

Grupa 17

Wiktoria Golinowska 264379

Wiktor Równanek 263607

Jakub Wrona 263589

Wrocław 2024

Spis treści

1.	Wstęp	4
2.	Wprowadzenie	5
3.	Założenia.....	8
3.1.	Założenie funkcjonalne	8
3.2.	Założenia konstrukcyjne	8
4.	Opis części sprzętowej.....	9
4.1.	Schemat Blokowy	9
4.2.	Schemat Ideowy.....	9
4.3.	Kluczowe elementy elektroniczne	11
4.4.	Kluczowe Elementy Mechaniczne.....	13
5.	Opis części programowej.....	19
5.1.	Opis środowiska.....	19
5.2.	Schemat algorytmu głównego.....	19
5.3.	Opis działania podstawowych funkcji programu.....	20
5.4.	Opis działania pętli i funkcji wykorzystanych w projekcie	20
5.5.	Działanie programu.....	21
6.	Uruchomienie, kalibracja.....	23
7.	Pomiary testowe.....	24
7.1.	Rezystancja rezystorów i fotorezystorów	24
7.2.	Pomiar Napięcia na dzielniku	24
7.3.	Specyfikacja Urządzenia.....	25
8.	Instrukcja obsługi dla użytkownika	26
1.	Warunki bezpiecznego użytkowania urządzenia	26
2.	Uruchomienie urządzenia	26
3.	Praca urządzenia	27
4.	Wyłączenie Urządzenia.....	27
9.	Podsumowanie	28
	Bibliografia	29
	Dodatki.....	30

1. Wstęp

Projekt Trackera Optycznego zrealizowany został w ramach kursu Optoelektronika 2. Konstrukcja urządzenia opiera się na dwóch silnikach krokowych 28bjy-48, fotorezystorach, czujnikach krańcowych oraz płytce ewaluacyjnej Arduino Nano.

Urządzenie umożliwia śledzenie źródła światła w dwóch osiach, a w połączeniu z panelem fotowoltaicznym może być wykorzystane do zwiększenia ilości generowanej energii. Dzięki zastosowaniu silników krokowych możliwa była minimalizacja zużycia energii dzięki możliwości ich wyłączenia po ustawieniu optymalnej pozycji.

W niniejszym raporcie opisane zostały podstawowe zagadnienia teoretyczne dotyczące śledzenia światła słonecznego oraz przedstawione zostały założenia konstrukcyjne i funkcjonalne.

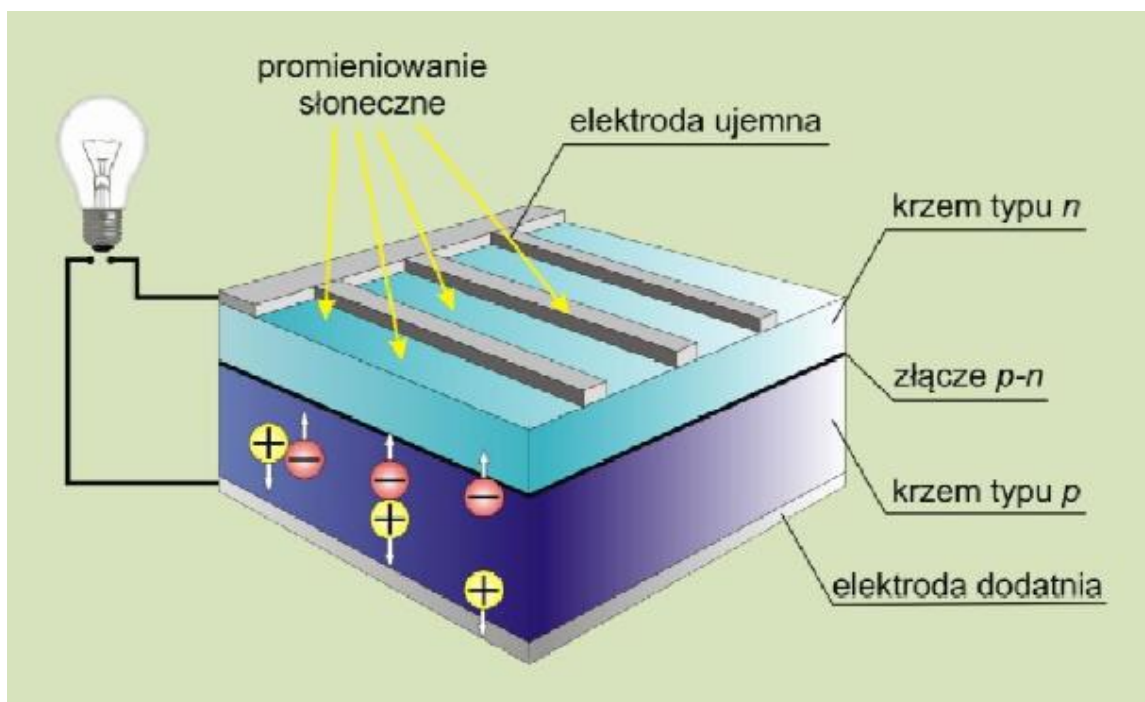
Budowa i zasada działania urządzenia będącego przedmiotem tego raportu została zaprezentowana na schematach blokowym oraz ideowym. W opisie części sprzętowej przedstawione zostały również rysunki techniczne obrazujące elementy mechaniczne.

Niniejszy raport obejmuje także opis części programowej - schemat blokowy algorytmu z opisem działania kluczowych funkcji oraz specyfikację techniczną urządzenia wraz z pomiarami testowymi. W rozdziale 8 zawarta została instrukcja użytkowania Trackera Optycznego.

2.Wprowadzenie

Śledzenie światła, znane również jako "light tracking" lub "solar tracking" w kontekście technologii fotowoltaicznych, odnosi się do systemów, które automatycznie dostosowują kąt nachylenia paneli fotowoltaicznych, aby zoptymalizować ich wydajność w zależności od położenia słońca na niebie. Te systemy, zwane również trackerami solarnymi, skierowują panele prostopadłe do promieni słonecznych, co zwiększa ich sprawność energetyczną.

Panele fotowoltaiczne działają na zasadzie efektu fotowoltaicznego, konwertując bezpośrednio energię światła na energię elektryczną. Ogniwa fotowoltaiczne które można zobaczyć na Rysunku 1, zazwyczaj wykonane są z półprzewodników, takich jak krzem. Generują przepływ prądu, gdy foton światła słonecznego uderza w ich strukturę. Panele składają się z wielu ogniw fotowoltaicznych połączonych szeregowo. Wytworzony prąd stały jest następnie przekształcany w prąd przemienny przez falownik, dostosowując jego parametry do sieci energetycznej. Licznik dwukierunkowy monitoruje bilans energii, uwzględniając zarówno energię wytworzoną przez panele, jak i tę pobraną z sieci zewnętrznej. [1]



Rysunek 1. Budowa ogniwa fotowoltaicznego [2]

Trackery solarne mogą być jednoosiowe lub dwuosiowe. Jednoosiowe zmieniają kąt nachylenia paneli w płaszczyźnie poziomej, podczas gdy dwuosiowe mogą dostosowywać kąt w obu płaszczyznach. Stosowanie tych systemów ma sens szczególnie w warunkach o niskim nasłonecznieniu, takich jak okres jesieni i zimy, gdy kąt padania promieni słonecznych jest niski i zmienia się w ciągu dnia. [3]

W rezultacie systemy śledzące słońce mogą zwiększyć wydajność instalacji fotowoltaicznej nawet o 45% w porównaniu do instalacji stałych. Niemniej jednak, ze względu na wyższe koszty inwestycyjne i eksploatacyjne, są one rzadziej stosowane w instalacjach domowych.

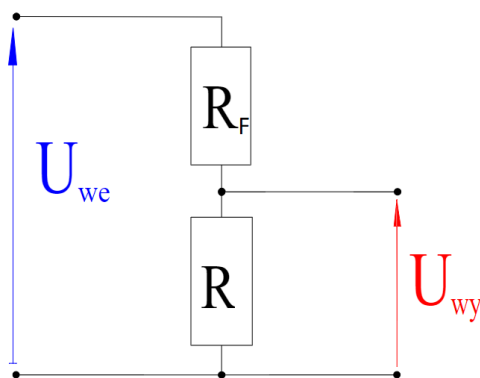
Przykładem miejsca gdzie systemy śledzące słońce są uważane za opłacalne są farmy fotowoltaiczne, ponieważ bardzo istotne jest tam maksymalizowanie produkcji energii na jednostkę powierzchni. [3]

Parametry opisujące urządzenia jakimi są trackery optyczne to m.in. zużycie energii (pobór mocy), liczba osi obrotu i maksymalne kąty obrotów, masa urządzenia, liczba oraz rozmiar możliwych do zainstalowania paneli fotowoltaicznych. [4]

Czujniki światła inaczej nazywane czujnikami fotoelektrycznymi reagują na zmianę intensywności docierającego do czujnika strumienia światła. W przypadku takich czujników może być wykorzystane światło widzialne jak również podczerwone czy laserowe. [5]

Fotorezystor, znany również jako rezystor światłoczuły, opiera swoje działanie na zjawisku fotoprzewodnictwa. Jest to półprzewodnikowy element elektroniczny, którego oporność elektryczna ulega zmianie w odpowiedzi na natężenie padającego światła. Kiedy fotony światła uderzają w strukturę półprzewodnikową fotorezystora, absorbują energię, co powoduje, że elektrony przechodzą do wyższych poziomów energetycznych. Ten proces prowadzi do zwiększenia zdolności przewodzenia materiału, co z kolei skutkuje zmniejszeniem oporu elektrycznego fotorezystora. [6]

Identyfikacja kierunku najjaśniejszego źródła światła odbywa się z wykorzystaniem 4 fotorezystorów pracujących w układach dzielnika napięciowego. Zasada działania takiego dzielnika przedstawiona została poglądowo na Rysunku 2 gdzie U_{we} jest napięciem stałym wynoszącym 5V. Jako element R_F zastosowano fotorezystor, którego rezystancja zmienia się wraz ze zmianą natężenia padającego światła. W efekcie zmianie ulega napięcie na wyjściu dzielnika U_{wy} . [7].



Rysunek 2. Dzielnik napięcia [7]

$$U_{WY} = \frac{U_{WE}}{R+R_F} \cdot R \quad (1)$$

Gdzie:

- U_{WY} to napięcie na wyjściu dzielnika
- U_{WE} to napięcie stałe wynoszące 5V
- R to rezystancja rezystora która wynosi $10k\Omega$
- R_F to rezystancja fotorezystora która jest zależna od naświetlenia.

Zasada działania tego dzielnika opiera się na zmianie rezystancji fotorezystora, co wpływa na napięcie wyjściowe U_{WY} . Dzięki temu, wykorzystując 4 takie układy można zidentyfikować kierunek najjaśniejszego źródła światła.

3. Założenia

W ramach kursu „Optoelektronika 2”, w trakcie jednego semestru akademickiego głównym celem było stworzenie urządzenia śledzącego światło, które mogłoby identyfikować kierunek źródła światła o największej intensywności. Następnie urządzenie automatycznie obracałoby się w kierunku wykrytego źródła światła ustawiając do niego prostopadle swoją płaszczyznę.

3.1. Założenie funkcjonalne

- Najważniejszym elementem funkcjonowania urządzenia jest możliwość wykrycia źródła światła, a następnie zdolność płynnego obracania się w jego stronę, zarówno w płaszczyźnie horyzontalnej jak i wertykalnej. Urządzenie powinno posiadać zdolność detekcji kierunku oświetlenia.
- Urządzenie powinno wyłączać silniki w stanie bezczynności w celu redukcji zużycia energii.
- Ustawianie się urządzenia prostopadle do światła powinno być możliwe za pomocą silników krokowych 28byj-48.
- Urządzenie powinno być zabezpieczone przed zapłataniem się lub zerwaniem przewodów przy pomocy przycisków krańcowych.
- Po spełnieniu powyższych warunków, urządzenie powinno mieć zdolność utrzymania oraz obracania panelu fotowoltaicznego.

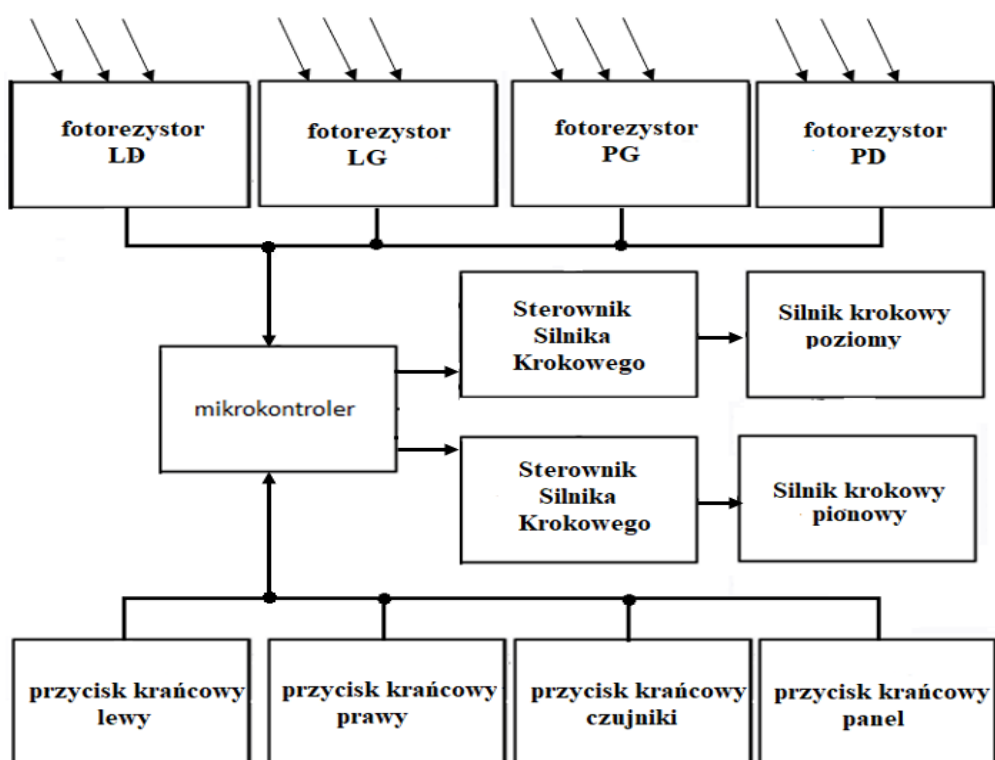
3.2. Założenia konstrukcyjne

- Urządzenie pod względem konstrukcji powinno składać się z trzech kluczowych elementów zrealizowanych w technologii druku 3D.
- Pierwszym elementem jest głowica. Jej budowa przedstawiona została na Rysunku 12. Zawiera 4 fotorezystory, rozmieszczone możliwie precyzyjnie w celu skutecznego rejestrowania zmian w oświetleniu.
- Kolejnym elementem konstrukcji jest ramię składające się z dwóch części przedstawionych na Rysunku 9 oraz na Rysunku 10, które wyposażono w silnik krokowy odpowiedzialny za poruszanie urządzenia w osi wertykalnej.
- Trzecim głównym elementem jest podstawa konstrukcji zaprezentowana na Rysunku 11. Zawiera silnik krokowy, który steruje osią horyzontalną, zapewniając regulację poziomą trackera.
- Dodatkowo na konstrukcji umieszczono 4 przyciski krańcowe, w celu ochrony przed zapłataniem się lub naderwaniem przewodów i utrzymania kontrolowanego ruchu elementów konstrukcyjnych.
- Centralny element logiczny stanowi mikrokontroler Arduino Nano, odczytując dane z fotorezystorów i sterując pracą obu silników. Dzięki temu układ jest w stanie reagować na zmieniające się warunki oświetleniowe, precyzyjnie śledząc źródło światła.

4.Opis części sprzętowej

4.1.Schemat Blokowy

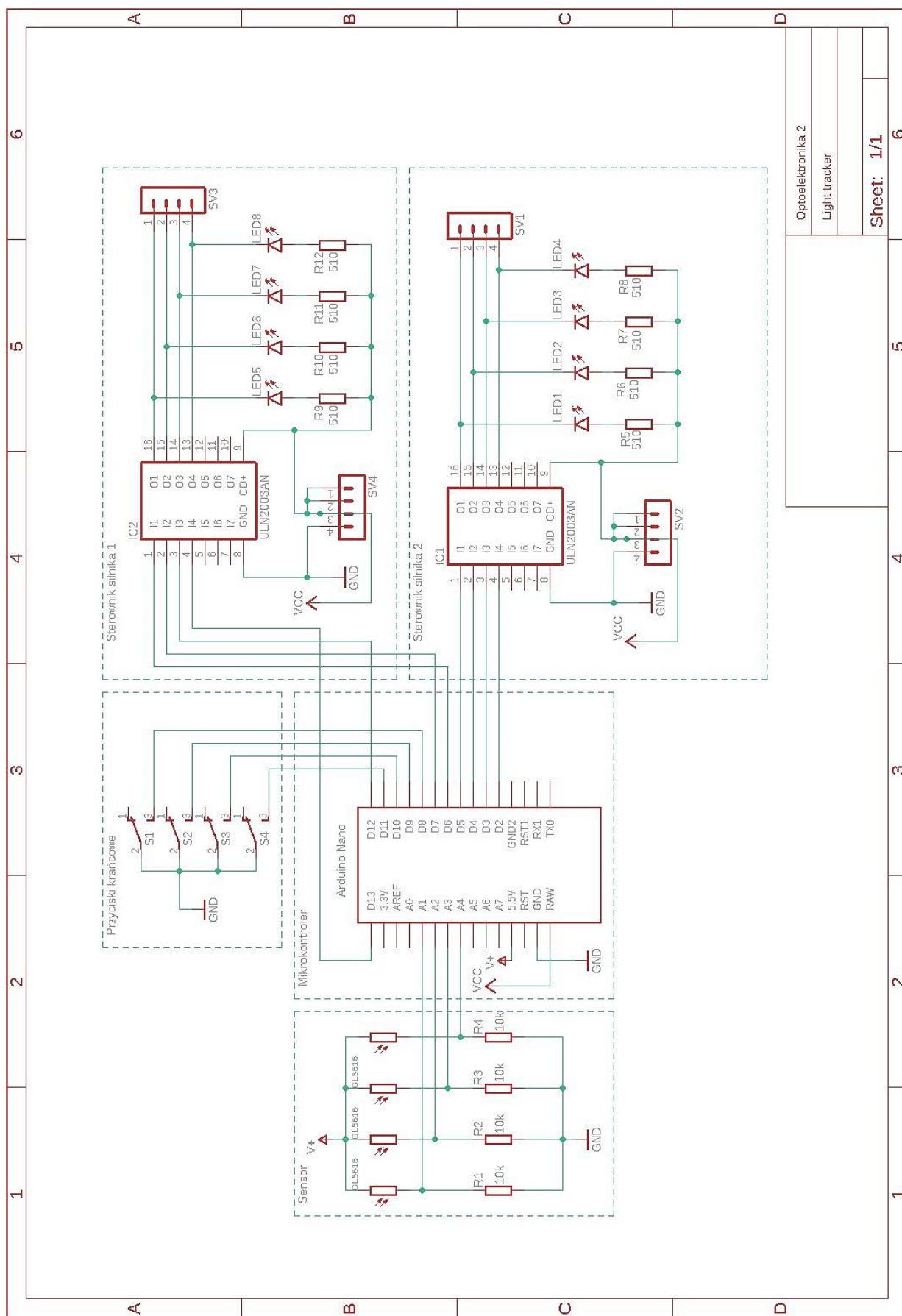
Schemat blokowy trackera optycznego przedstawiony został na Rysunku 3. Urządzenie składa się z czterech fotorezystorów odpowiadających za identyfikację kierunku oświetlenia głowicy urządzenia. Sygnał podawany jest na analogowe wejścia mikrokontrolera, który analizuje otrzymane dane w celu wyznaczenia najjaśniejszego punktu. Na ich podstawie wysyłane są sygnały na wejścia sterowników silników krokowych w celu obrócenia urządzenia w pożądanym kierunku do ustawienia się w optymalnej pozycji. Zastosowane przyciski krańcowe zapobiegają uszkodzeniom mechanicznym urządzenia poprzez np. zerwanie przewodów. W przypadku osiągnięcia położenia krańcowego w danej osi, mikrokontroler otrzymuje informację w postaci pojawienia się stanu niskiego na jego wejściu cyfrowym i uniemożliwia dalszy ruch w tym kierunku. Poglądowe zdjęcie zmontowanego urządzenia znajduje się na Rysunku 8.



Rysunek 3. Schemat blokowy urządzenia

4.2.Schemat Ideowy

W schemacie ideowym umieszczonym na Rysunku 4 można wyróżnić poszczególne bloki elementów takie jak blok sensora, mikrokontrolera, przycisków krańcowych oraz sterowniki obu silników. W bloku sensora znajdują się takie elementy jak cztery rezystory R1-R4 o wartościach 10k Ω oraz fotorezystory GL5616 o rezystancji jasnej wynoszącej od 5 do 10 k Ω . Przyciski krańcowe są umieszczone w kolejnym bloku odpowiadającym za zabezpieczenie urządzenia przed uszkodzeniem mechanicznym. Do tego w sterownikach silnika są wbudowane rezystory od R5 do R12 o wartościach 510 Ω , 8 diod czerwonych oznaczonych na schemacie od LED1 do LED8 oraz układy scalone ULN2003AN. Układ zasilany jest zewnętrznym zasilaczem, a napięcie wejściowe Vcc powinno mieścić się w zakresie 5 – 9V.

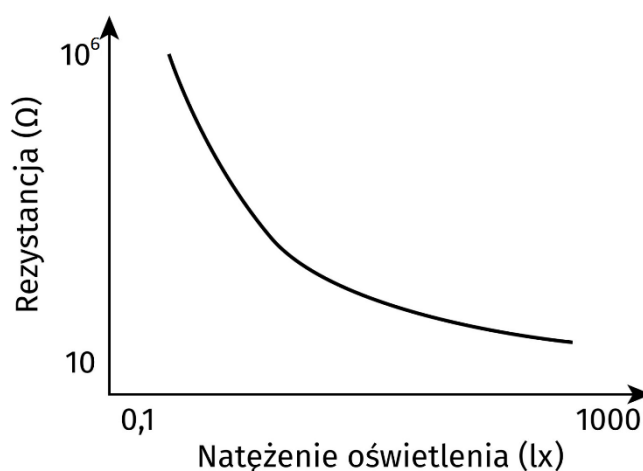


Rysunek 4. Schemat ideowy urządzenia

4.3. Kluczowe elementy elektroniczne

Fotorezystory

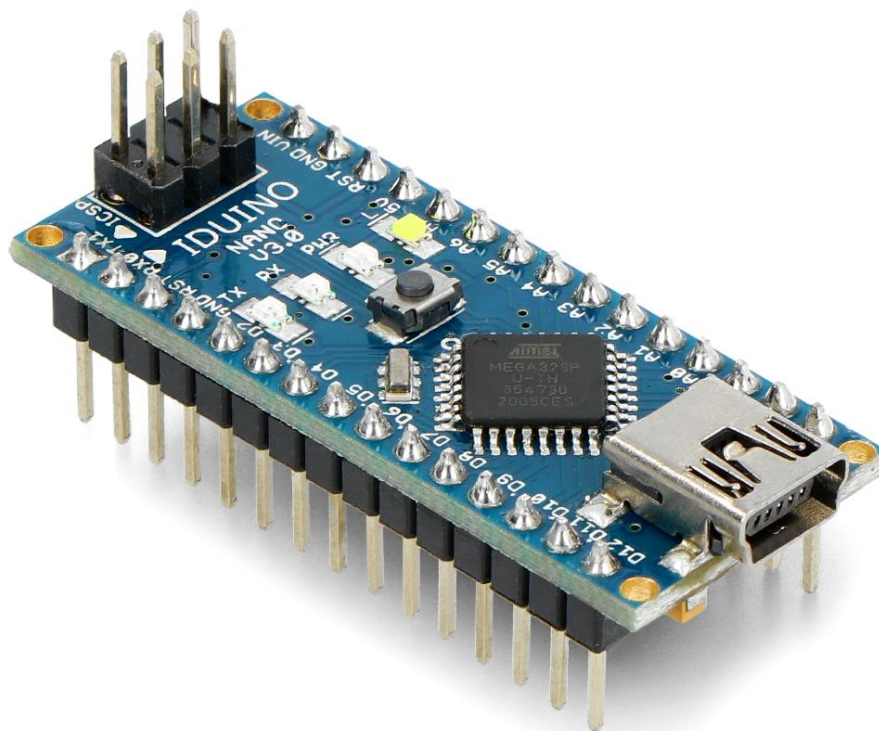
Przy konstrukcji modułu czujnika, wykorzystano 4 fotorezystory GL5616 o deklarowanej rezystancji jasnej 5 – 10k Ω . Rezystancja tych elementów zmienia się w zależności od natężenia padającego światła, co zostało przedstawione na poglądowej charakterystyce, na Rysunku 5. Fotorezystory umieszczone zostały w sposób umożliwiający pomiar natężenia oświetlenia w każdym z kierunków. Dzięki włączeniu ich do układu dzielnika napięcia, analogicznie do umieszczonego na Rysunku 4 schemacie ideowym, możliwe było zaobserwowanie zmian napięcia na poszczególnych wyjściach, w zależności od położenia źródła światła, a tym samym określenie kierunku obrotu całego urządzenia.



Rysunek 5. Poglądowa charakterystyka zależności rezystancji od natężenia oświetlenia fotorezystora [6]

Przyciski krańcowe

Przyciski krańcowe są kluczowym elementem projektu mającym na celu zabezpieczenie przed uszkodzeniem przewodów oraz ochronę przed ewentualnymi kolizjami mechanicznymi. Przykład takiego przycisku został przedstawiony na Rysunku 6. Przyciski krańcowe zostały odpowiednio umieszczone w obydwu osiach, aby zapewnić bezpieczny zakres pracy elementów mechanicznych urządzenia. W przypadku pojawienia się stanu niskiego na wyprowadzeniu przycisku, program nie pozwoli na dalszy ruch w kierunku określonym lokalizacją przycisku.



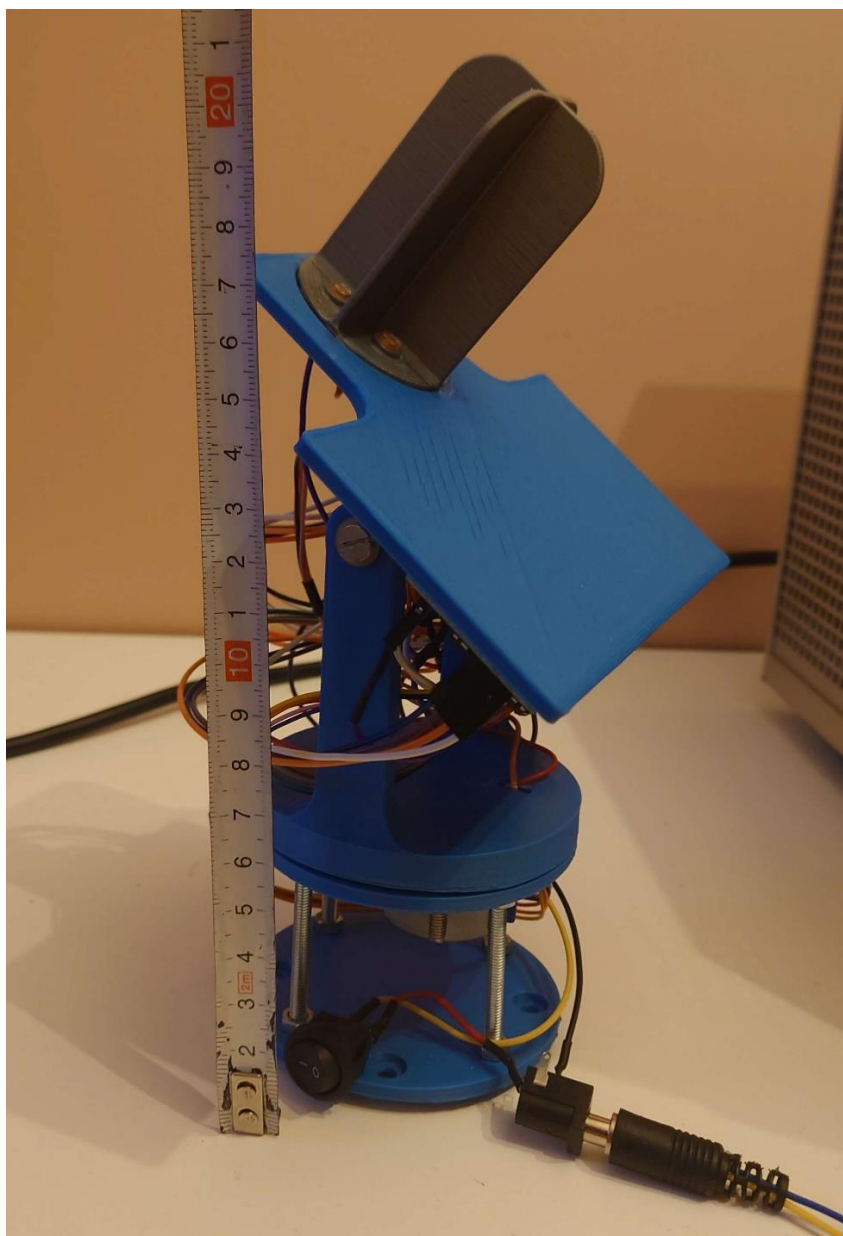
Rysunek 7. Arduino nano [8]

Silnik krokowy – 28byj-48 ze sterownikiem

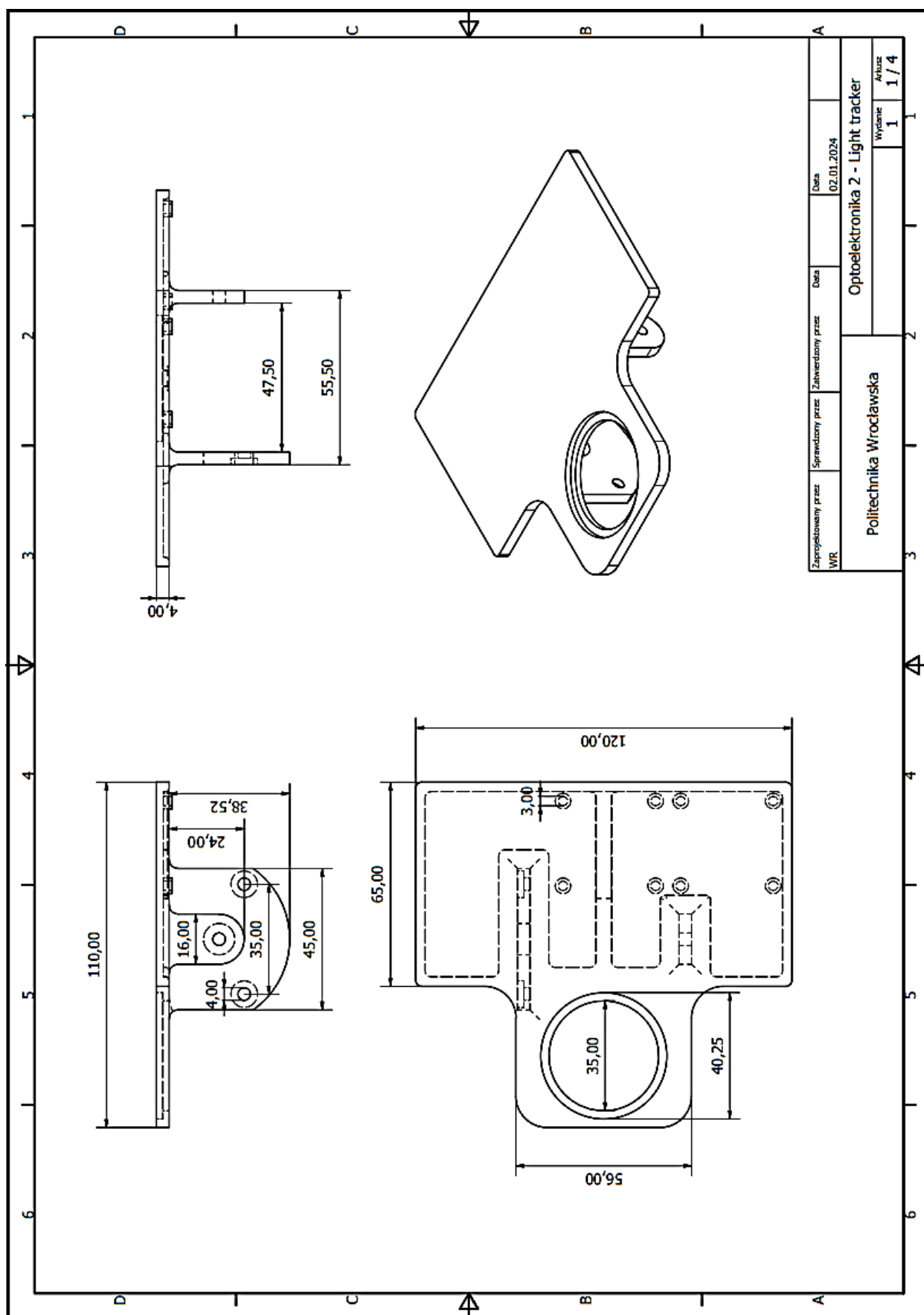
Silnik krokowy 28byj-48 może być precyzyjnie sterowany poprzez podawanie sygnałów elektrycznych na jego uzwojenia. Silnik ten ma wbudowaną przekładnię redukcijną o stosunku 64:1. Ponadto jest zasilany i sterowany przez moduł oparty na układzie ULN2003, który wykorzystuje tranzystory w układzie Darlingtona do wzmocnienia sygnału, tym samym chroniąc mikrokontroler podający impulsy przed zbyt dużym obciążeniem wyjść logicznych. W projekcie, silniki krokowe 28byj-48 zostały wykorzystane do obracania trackera w zależności od kąta padania promieni słonecznych. Moduł sterujący pozwala na regulację prędkości i kierunku obrotu silnika poprzez zmianę sekwencji i czasu trwania impulsów. [9]

4.4. Kluczowe Elementy Mechaniczne

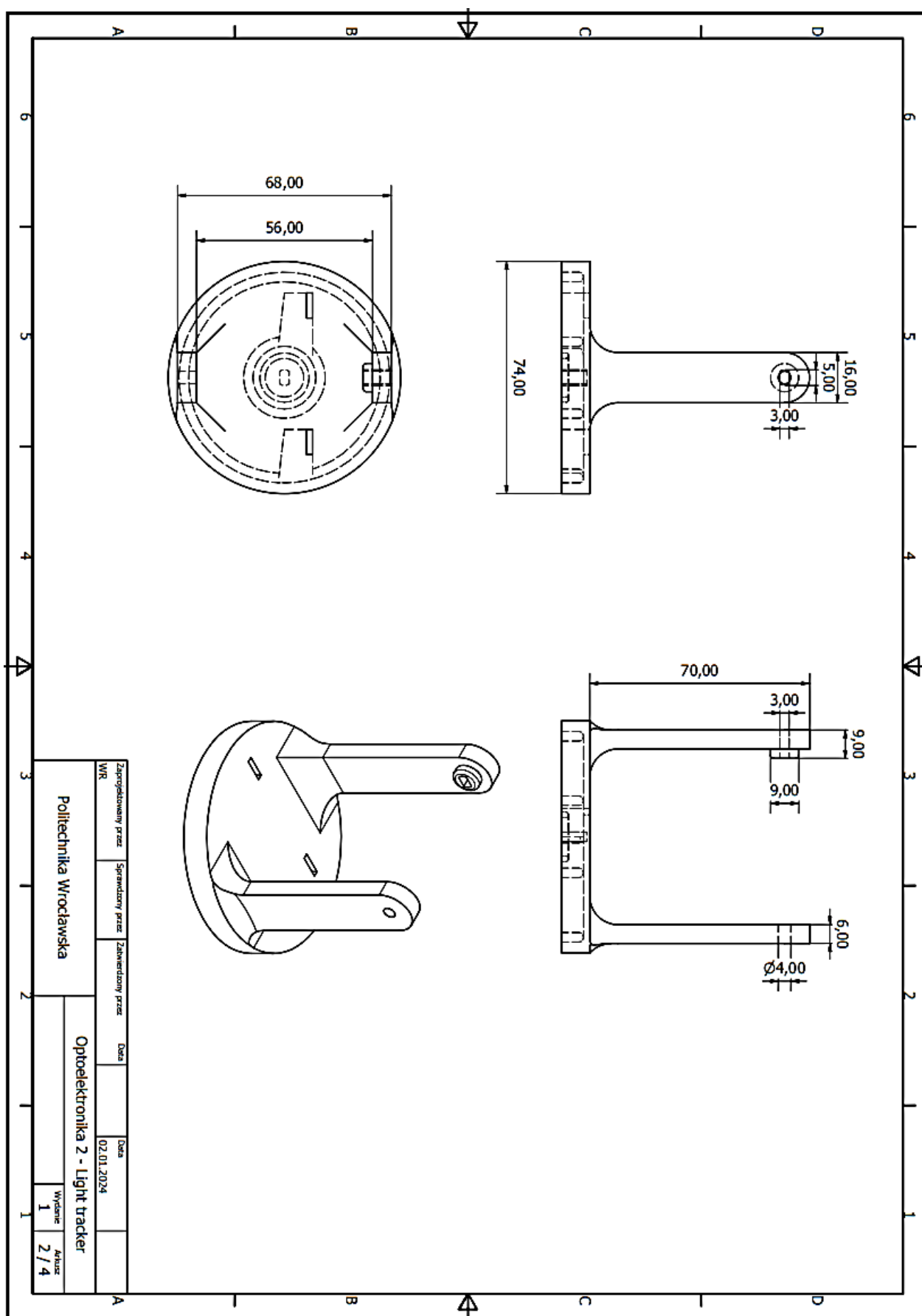
Konstrukcja mechaniczna urządzenia została zaprojektowana w programie Autodesk Inventor oraz zrealizowana w technologii druku 3D. Zespół elementów mechanicznych składa się z dwóch ruchomych części odpowiedzialnych za manipulację urządzeniem w dwóch osiach, podstawy z silnikiem oraz elementu pozwalającego na zamontowanie czujników światła w celu identyfikacji kierunku jego źródła. Elementy konstrukcyjne wraz z ich wymiarami zostały przedstawione na Rysunkach 9-12. Poniżej, na Rysunku 8 przedstawiony został sposób fizycznego montażu opisanego urządzenia wraz z jego wymiarem pogładowym.



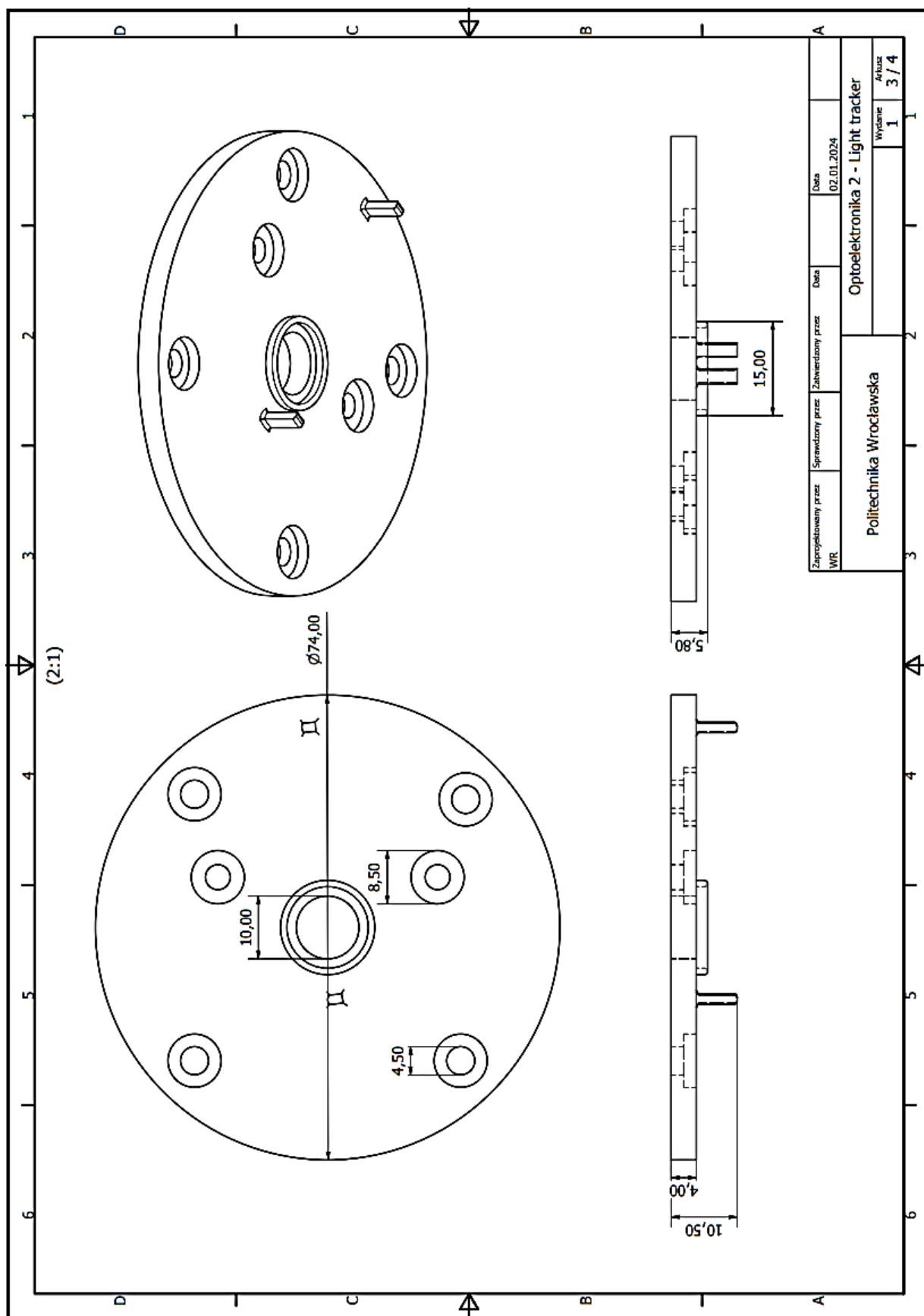
Rysunek 8. Zdjęcie zmontowanego urządzenia



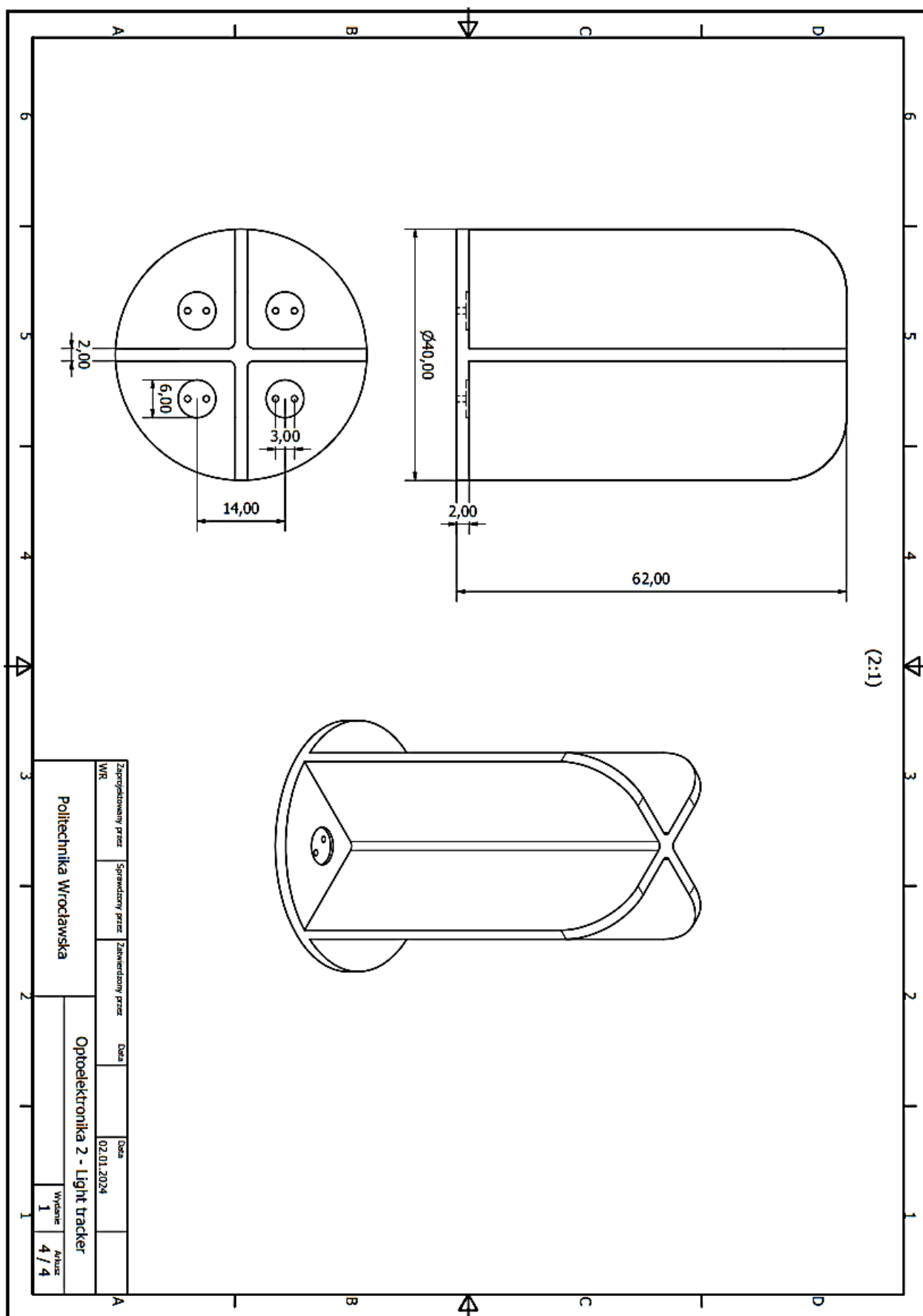
Rysunek 9. Element konstrukcyjny 1 - uchwyt panelu fotowoltaicznego oraz sensora światła poruszający się w płaszczyźnie wertykalnej



Rysunek 10. Element konstrukcyjny 2 - uchwyt poruszający się w płaszczyźnie horyzontalnej



Rysunek 11. Element konstrukcyjny 3 – nieruchoma podstawa trackera optycznego



Rysunek 12. Element konstrukcyjny 4 - sensor

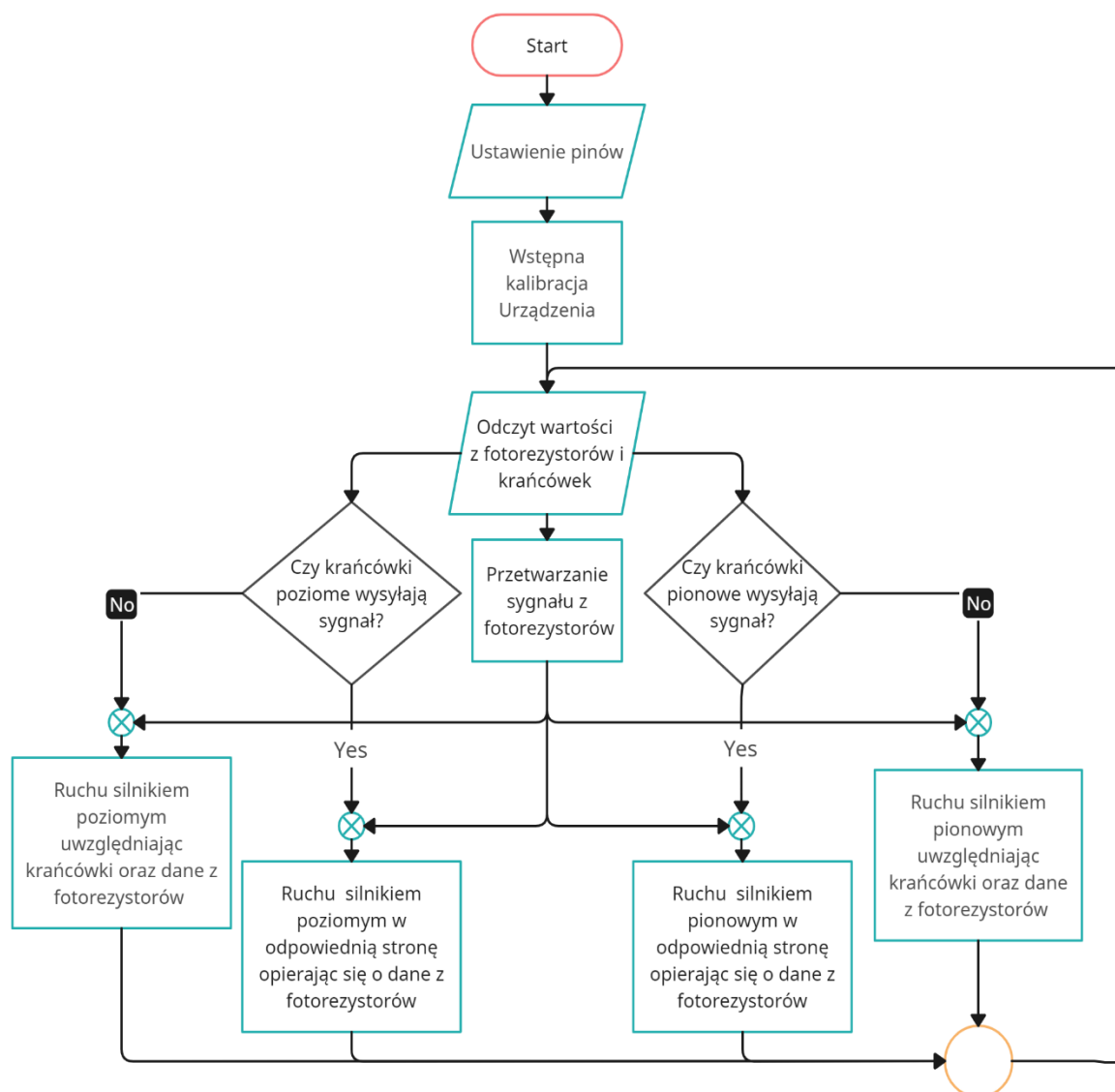
5. Opis części programowej

5.1. Opis środowiska

W trakcie realizacji projektu skorzystano z środowiska programistycznego Arduino IDE. Jest to środowisko stworzone przez firmę zajmującą się produkcją mikrokontrolerów, w tym również tego wykorzystanego do poruszania silnikami trackera. Narzędzie to było niezbędne do procesu programowania, kompilacji oraz wgrywania oprogramowania do mikrokontrolera. Jego zaletą jest dostęp do gotowych projektów, bibliotek, a także otwartego forum, na którym można szukać pomocy.

5.2. Schemat algorytmu głównego

Poniżej na Rysunku 13 przedstawiono algorytm główny programu. Dokładniejszy opis jego działania znajduje się w punktach 5.4-5.5.



Rysunek 13. Schemat algorytmu głównego

5.3. Opis działania podstawowych funkcji programu

Podczas uruchomienia środowiska Arduino IDE ukazuje się okno gotowe do wpisania kodu. Automatycznie pojawiają się w nim dwie funkcje `void setup()` oraz `void loop()`. Można je zauważyć na Rysunku 14. Pierwsza z nich służy do zdefiniowania wejść cyfrowych lub analogowych oraz umożliwienia monitorowania portu szeregowego, a także pierwszej kalibracji urządzenia. Jej główną cechą jest to, że wykonuje się tylko raz na początku uruchomienia programu. Druga funkcja jest niekończącą się pętlą, w której linii kodu są wykonywane w określonej przez programistę kolejności. Należy pamiętać, że wykonuje się ona cały czas, dlatego po zakończeniu kodu następuje powrót do pierwszej linii i ponowne rozpoczęcie działania programu.

```
void setup() {  
    // umieść tutaj swój kod instalacyjny, aby uruchomić go raz;  
}  
  
void loop() {  
    // umieść tutaj swój główny kod, aby uruchamiał się ciągle;  
}
```

Rysunek 14. Podstawowe funkcje dostępne w Arduino IDE po uruchomieniu

5.4. Opis działania pętli i funkcji wykorzystanych w projekcie

Początkowym elementem kodu jest zaimportowanie biblioteki `Stepper`. Jest to jedna z najczęściej używanych bibliotek, służąca do obsługi silników krokowych w projektach mikrokontrolerów. Umożliwia precyzyjną kontrolę ich ruchu poprzez korzystanie z gotowych funkcji i klas, które są przez nią dostarczane. Należy także pamiętać, że biblioteka ta musi być zainstalowana w danym środowisku, wykorzystywanym do pisania kodu, w celu poprawnego używania jej w projekcie. Program Trackera Optycznego korzysta zarówno z klasy `Stepper`, używanej do obsługi dwóch silników krokowych, jak i metody `step`, która używana jest do kontrolowania ilości kroków, jakie każdy z silników wykonuje. Umożliwia to precyzyjne sterowanie ruchem w płaszczyźnie poziomej i pionowej na podstawie danych z czujników oświetlenia. Przykładowe linie kodu z użyciem tej biblioteki zostały umieszczone na Rysunku 15. Natomiast szczegółowy opis ich działania znajduje się w punkcie 5.5 - Działanie programu.

```
#include <Stepper.h>  
  
Stepper Pion (kroki_na_obrot, 4, 5, 6, 7);  
  
Pion.step(1);
```

Rysunek 15. Przykłady użycia i zaimportowania biblioteki `Stepper`

Projekt Trackera Optycznego wykorzystuje szereg funkcji umożliwiających interakcję z mikrokontrolerem oraz sterowanie silnikami i czujnikami. Poniżej znajdują się opisy funkcji i struktur użytych w programie, które zostały stworzone na podstawie podanego źródła. [10]

- `pinMode()` - funkcja dwuargumentowa służąca do konfiguracji pinów mikrokontrolera. Określa czy dany pin będzie używany jako wejścia (INPUT) lub wyjścia (OUTPUT). Dodatkowo dzięki niej można także określić wejścia z włączonym wewnętrznym rezystorem (INPUT_PULLUP), co powoduje, że pin ma napięcie wysokie (HIGH), wyjątkiem jest gdy jest aktywowany przez zewnętrzne zdarzenie, np. wciśnięcie przycisku, co spowoduje obniżenie stanu na pinie do niskiego (LOW).
- `digitalRead()`-funkcja do odczytu stanu logicznego z danego pinu cyfrowego. Zwraca HIGH lub LOW, co odpowiada stanowi wysokiemu lub niskiemu na danym pinie.
- `analogRead()` –funkcja do odczytu wartości analogowej z określonego pinu. Przekształca sygnał analogowy na liczbę z zakresu 0-1023 poprzez odczytanie wartości na pinie analogowym w postaci 10-bitowej informacji.
- `if()` oraz `else` – if to struktura warunkowa służąca do realizacji kodu, który zostanie wykonany tylko wtedy, gdy określone wyrażenie logiczne jest prawdziwe, w przeciwnym razie zostanie wykonany kod wewnątrz bloku `else`.

5.5.Działanie programu

Poniżej opisano kolejne bloki programu:

- **Definicja pinów i konfiguracja silników**

Na samym początku programu w celu dobrego działania urządzenia nastąpiło odpowiednie zdefiniowanie pinów analogowych A1 do A4 jako wejść dla czujników światła. W tym przypadku dla fotorezystorów, pamiętając, że LG oznacza lewy górny, LD oznacza lewy dolny i w przypadku prawych oznaczenie to po prostu PG oraz PD. Następnym krokiem była definicja przycisków krańcowych, dwóch umieszczonych w podstawie trackera (`PrawyMax` oraz `LewyMax`) odpowiedzialnych za zabezpieczenie ruchu w osi poziomej oraz dwóch na jego ramieniu (`PionCzujniki` oraz `PionPanel`) zabezpieczających ruch w osi pionowej. Za definicję przycisków krańcowych odpowiedzialne są piny cyfrowe 8-11. Linie 59-66 przedstawiają konfigurację wszystkich pinów jako wejścia.

Do sterowania silnikami krokowymi również zostały użyte piny cyfrowe, tym razem były to piny 2-7 oraz 12 i 13. Ich konkretna konfiguracja (przykład na Rysunku 15) jest zdefiniowana w obiektach `Poziom` i `Pion` klasy `Stepper`, w całym kodzie programu umieszczonym w Dodatkach na Rysunku 17, w wierszach 24-25. Zmienna `kroki_na_obrot` określa liczbę kroków potrzebną do wykonania pełnego obrotu silnika krokowego, dlatego jest wartością stałą. Konfiguracja silników odbywa się również w liniach 69-70, gdzie ustawiana jest prędkość obrotowa silników.

- **Kalibracja początkowa**

Funkcja `kalibracjaPion` obraca silnik pionowy do momentu, aż nie zostanie aktywowany przycisk krańcowy `PionCzujniki`. Natomiast funkcja `kalibracjaPoziom` obraca silnik poziomy, dopóki nie zostanie aktywowany stan niski dla przycisku `PrawyMax`, kolejno wykonuje dodatkowe 400 kroków w przeciwnym kierunku. Wywołanie tych dwóch funkcji następuje w funkcji `setup()`, w liniach 73-74, dlatego wykonuje się tylko raz podczas pierwszego uruchomienia.

- **Funkcje wyłączające silniki**

Funkcje `wylaczPion` i `wylaczPoziom` ustawiają piny sterujące silnikami na niski poziom, co powoduje zatrzymanie silników. Nie jest to jednoznaczne z wstrzymaniem działania trackera. Powoduje jedynie minimalizację zużycia energii, ponieważ po wykryciu zmiany kierunku źródła światła silniki automatycznie się załączają. Funkcje te zostały przedstawione w Dodatkach na Rysunku 17 w wierszach 41-54.

- **Uśrednienie wartości**

Uśrednienie polega na odczytaniu wartości z dzielnika fotorezystorów, a następnie podzielenie ich na cztery strony: góra, dół, lewo i prawo. Piny odpowiadające za górę odnoszą się do PG oraz LG. Dla każdej ze stron obliczana jest średnia. Funkcja ta umieszczona jest w Dodatkach na Rysunku 18, w wierszach 79-82, w funkcji `loop()`. Parametr ten może być zmieniony zależności od preferowanego progu reakcji trackera

- **Wyznaczenie kierunków obrotów silników**

Z wcześniej uśrednionych wyników wyznaczamy stronę obrotu silnika na podstawie różnicy jednej z płaszczyzn dla zmiennych V (oś wertykalna) oraz H (oś horyzontalna). Jeżeli góra jest większa od dołu daje to nam dla V wynik dodatni, natomiast jeśli wartość dołu jest większa dostajemy wynik ujemny. Analogicznie jest na osi poziomej H. Funkcja ta umieszczona jest w Dodatkach na Rysunku 18, w wierszach 84-85 w funkcji `loop()`.

- **Ruch w osi wertykalnej**

Ruch silnika pionowego jest kontrolowany w zależności od stanu przycisku krańcowego (`PionPanel`) i wartości składowej ruchu pionowego (V). Jeśli przycisk jest naciśnięty oraz góra jest większa od dołu silnik kroczy w górę, jeśli nie to wyłącza się. W przypadku wciśnięcia drugiego przycisku (`PionCzujniki`) i wartości dołu większej od góry o 100 silnik wykona 300 kroków w prawo w celu zmiany położenia panelu. W przeciwnym razie, silnik kroczy w dół lub wyłącza się w zależności od wartości V.

- **Ruch w osi horyzontalnej**

Ruch silnika poziomego jest kontrolowany w zależności od stanu czujników krańcowych (`PrawyMax` i `LewyMax`) oraz wartości składowej ruchu poziomego (H). Jeśli prawy czujnik krańcowy jest aktywowany (stan niski) i wartość H przekracza 100, to silnik wykonuje ruch w prawo, a liczba kroków wynosi 1500. W przypadku aktywacji lewego czujnika krańcowego i wartości H mniejszej niż -100, silnik przemieszcza się w lewo, wykonując 1500 kroków. W przeciwnym razie, silnik wyłącza się, zatrzymując się w bieżącym położeniu.

6. Uruchomienie, kalibracja

Aby właściwie zamontować urządzenie, należy je umieścić na płaskiej powierzchni, wykorzystując do tego specjalnie przygotowane otwory, znajdujące się w podstawie, o średnicy 4 mm. Umożliwia to łatwe przymocowanie do podłoża za pomocą śrub lub wkrętów, co zapewnia stabilność w trakcie użytkowania.

Do poprawnego uruchomienia urządzenia niezbędne jest dostarczenie zasilania o napięciu stałym w zakresie od 5 do 9 V oraz wydajności prądowej co najmniej 0,8A. Optymalne warunki pracy osiąga się przy napięciu 7,5V i poborze prądu 0,8A. Należy pamiętać, że przekroczenie zakresu 9V może grozić uszkodzeniem sterowników oraz samych silników, podczas gdy zbyt niskie napięcie może skutkować nieprawidłowym funkcjonowaniem lub nawet brakiem reakcji urządzenia.

Po podłączeniu zasilania do podstawy urządzenia i włączeniu go za pomocą włącznika, należy odczekać 2 sekundy, ponieważ w tym czasie nastąpi automatyczna kalibracja Trackera Optycznego. Proces ten polega na wyszukaniu skrajnego położenia obu osi urządzenia przy pomocy przycisków krańcowych, a następnie ustawieniu się w pozycji neutralnej (oś pozioma w połowie swojego zakresu ruchu, oś pionowa równoległa do powierzchni ziemi) umożliwiającą znalezienie kierunku najjaśniejszego źródła światła. Po wykonaniu tej procedury Trackera stanie się gotowy do śledzenia światła obracając się w kierunku wykrytego źródła promieniowania. Urządzenie nie wymaga kalibracji przez użytkownika.

7. Pomiary testowe

W trakcie etapu konstrukcyjnego urządzenia przeprowadzono szereg testów i pomiarów, mających na celu weryfikację integralności oraz skuteczności jego działania.

7.1. Rezystancja rezystorów i fotorezystorów

W celu zapewnienia właściwej funkcjonalności urządzenia, istotne jest, aby wartości rezystorów i fotorezystorów były zbliżone do siebie, umożliwiając precyzyjne śledzenie światła przez trackera. W przypadku rezystorów, wyniki przedstawione w Tabeli 1 są spójne co potwierdza prawidłowe dobranie tych elementów.

W Tabeli 2 przedstawiono rzeczywiste parametry zastosowanych fotorezystorów. Ich parametry odbiegają od siebie w większym stopniu niż w przypadku rezystorów, ze względu m.in. na sposób pomiaru rezystancji jasnej. Oświetlenie tych elementów dokładnie w ten sam sposób było niemożliwe ze względu na warunki pomiarowe, a ewentualne niewielkie odchyłki natężenia padającego światła skutkowało odchyłkami w otrzymanych wynikach pomiarowych.

Tabela 1. Rezystancja Rezystorów

Element	Wartość zmierzona	Jednostka
Rezystor R1	10010	Ω
Rezystor R2	9994	Ω
Rezystor R3	10000	Ω
Rezystor R4	10010	Ω

Tabela 2. Rezystancja Fotorezystorów

Element	Rezystancja ciemna	Rezystancja Jasna	Jednostka
Fotorezystor LG	1.1M	1263	Ω
Fotorezystor LD	1.1M	1253	Ω
Fotorezystor PG	1M	1061	Ω
Fotorezystor PD	1.1M	1103	Ω

7.2. Pomiar Napięcia na dzielniku

Pomiar napięcia na wyjściach dzielników napięcia, złożonych z fotorezystorów oraz rezystorów 10k Ω przedstawiono w Tabeli 3. W przypadku rezystancji jasnej parametr ten był spójny, dla każdego z układów przyjmując podobne wartości. Dla rezystancji ciemnej (przy całkowitym zakryciu powierzchni fotorezystora) wartości zmierzonego napięcia wynosiły 0V. Niewielkie odchyłki parametrów badanych elementów wskazują na ich prawidłowe dobranie umożliwiające uzyskanie dostatecznej dokładności urządzenia.

Tabela 3. Napięcie na wyjściu dzielnika napięcia

Napięcie Uwyj2 dla elementu:	Dla Rezystancji ciemnej	Dla Rezystancji Jasnej	Jednostka
Fotorezystor LG	0,005	4,20	V
Fotorezystor LD	0,004	4,08	V
Fotorezystor PG	0,005	4,17	V
Fotorezystor PD	0,004	4,13	V

7.3. Specyfikacja Urządzenia

Urządzenie Tracker Optyczny zasilane jest ze źródła napięcia stałego, które jest dostarczane przez gniazdo wejściowe na podstawie urządzenia. Napięcie zasilania powinno mieścić się w zakresie od 5V do 9V, a wydajność prądowa źródła powinna wynosić co najmniej 0,6A, aby zapewnić stabilne i skuteczne działanie urządzenia.

W trakcie standardowego funkcjonowania, urządzenie zużywa określoną ilość prądu. Całkowite natężenie prądu pobieranego przez urządzenie wynosi 550 mA. Warto zauważyć, że mikrokontroler Arduino, stanowiący centralny element logiczny, sam w sobie pobiera około 45 mA prądu. Głównym elementem, który dominuje w zużyciu prądu, są silniki, które pobierają aż 500 mA. W Tabeli 4 przedstawiono prąd pobierany przez urządzenie podczas pracy – z włączonymi silnikami (550mA) oraz w stanie spoczynku (45mA).

Urządzenie posiada zdolność obrotu w płaszczyźnie poziomej z prędkością 6,45 obr/min, oraz zakresie wynoszącym 345°, zbliżając się niemal do pełnego okręgu. Warto zauważyć, że występują „luzy” konstrukcyjne, które mogą wpłynąć na zmianę zakresu obrotu silnika.

W płaszczyźnie pionowej urządzenie obraca się z prędkością 5 obr/min w zakresie wynoszącym 60°. Przekładnie zastosowanych silników są fabrycznie słabo spasowane, obszar w jakim oś posusza się swobodnie bez obracania silnikiem wynosi 5°. Może to wpływać na precyzję urządzenia. Pomiar szybkości obrotu został wykonany za pomocą stopera, czas został zmierzony od wciśnięcia jednej krawędzi do wciśnięcia drugiej i obliczony na podstawie obrotu na cały okrąg.

Tabela 4. Specyfikacja urządzenia

Parametr	Symbol	Min.	Typ.	Max.	Jednostka
Napięcie zasilania	Ucc	5	7,5	9	V
Pobierany prąd	Icc	45	550	550	mA
Kąt obrotu osi Poziomej	H	-	345	-	°
Kąt obrotu osi Pionowej	V	-	60	-	°
Szybkość obrotu osi Poziomej	Hv	-	5	-	obr/min
Szybkość obrotu osi Pionowej	Vv	-	6,45	-	obr/min
Waga urządzenia	Masa	-	265	-	g
Szerokość	y	-	-	120	mm
Wysokość	z	-	-	220	mm

8. Instrukcja obsługi dla użytkownika

1. Warunki bezpiecznego użytkowania urządzenia

W celu uniknięcia uszczerbku na zdrowiu oraz uszkodzenia urządzenia należy przestrzegać poniższych zaleceń:

- Przed rozpoczęciem użytkowania należy zapoznać się z instrukcją obsługi i zachować ją na przyszłość.
- Urządzenie nie powinno być ustawiane na pochyłych lub niestabilnych powierzchniach.
- Należy zabezpieczyć pole zasięgu działania trackera i usunąć z niego wszystkie przedmioty, które mogą zakłócać jego poprawną pracę.
- Podczas pracy należy sprawować permanentny nadzór nad urządzeniem, jeśli działa w pobliżu dzieci, należy zachować nad urządzeniem szczególną kontrolę.
- Podczas podłączania zasilania należy zachować wszystkie niezbędne środki ostrożności zgodne z zasadami BHP.
- Aby zabezpieczyć się przed zagrożeniem elektrycznym, nie należy zanurzać urządzenia ani przewodu w wodzie ani w żadnej innej cieczy.
- Nie należy dotykać przewodów umieszczonych na urządzeniu.
- Urządzenie nie powinno być narażone na warunki pogodowe typu deszcz, śnieg czy grad, ponieważ mogłoby to uszkodzić czujniki znajdujące się na jego głowicy.
- W przypadku stwierdzenia niepoprawnej pracy należy bezzwłocznie odłączyć zasilanie.
- Każde zmiany wprowadzone przez użytkownika mogą wpłynąć negatywnie na poprawne działanie urządzenia, dlatego nie zaleca się ich wykonywania.

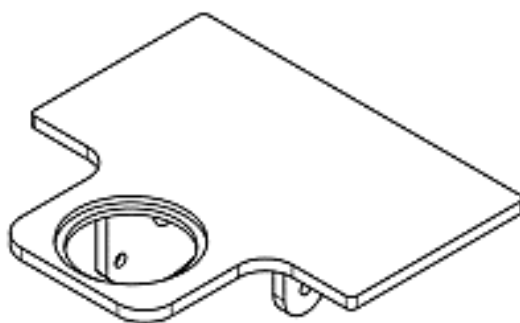
2. Uruchomienie urządzenia

Do prawidłowego rozpoczęcia działania urządzenia, należy umieścić je na równej i stabilnej powierzchni oraz podłączyć do niego zasilacz. Urządzenie posiada specjalnie przygotowane otwory o średnicy 4mm w podstawie, które umożliwiają jego przymocowanie do podłoża dla lepszej stabilności. Zasilanie układu znajduje się na podstawie urządzenia. Zasilacz powinien dostarczać napięcie stałe 5-9 V i prąd co najmniej 0,6A. Po podłączeniu zasilania, należy włączyć urządzenie przyciskiem zasilania zamontowanym na podstawie urządzenia. Następnie urządzenie po około 2 sekundach przeprowadzi kalibrację, która jest niezbędna do prawidłowego funkcjonowania trackera. Po tym procesie tracker jest gotowy do dalszej pracy.

3. Praca urządzenia

Poprawna praca urządzenia polega na namierzaniu najjaśniejszego źródła światła. W przypadku osiągnięcia maksymalnego zakresu pracy usłyszysz cichy, ale charakterystyczny klik. Wówczas urządzenie nie powinno poruszać się dalej w tą stronę. Pusta powierzchnia znajdująca się na górze urządzenia (patrz na Rysunek 1.) służy do przymocowania panelu fotowoltaicznego przez użytkownika.

W przypadku zablokowania się elementów mechanicznych należy poruszyć urządzeniem w celu umożliwienia dalszej pracy układu.



Rysunek 1 - Miejsce na zamontowanie panelu fotowoltaicznego

4. Wyłączenie Urządzenia

W celu poprawnego wyłączenia urządzenia należy użyć przełącznika znajdującego się na podstawie urządzenia i ustawić go w pozycji 0. Następnie wystarczy odłączyć przewód zasilający.

9. Podsumowanie

Podczas realizacji projektu wykorzystaliśmy wiedzę z dziedziny optoelektroniki do stworzenia urządzenia, którego działanie opiera się na detekcji światła i interpretacji wyników, co stanowiło jeden z celów naszego zadania semestralnego. Napotkaliśmy na wiele trudności, z których największą była realizacja mechaniczna urządzenia oraz problem z mocą zastosowanych silników. To spowodowało, że nie udało nam się zrealizować wszystkich założeń projektowych.

Spelnione założenia

- Urządzenie jest zabezpieczone przed uszkodzeniem przewodów
- Urządzenie wychwycą najjaśniejszy punkt a następnie ustawia się prostopadle do jego źródła światła
- Urządzenie wyłącza silniki w chwili spoczynku

Niespełnione założenia

Brak panelu fotowoltaicznego

Możliwe modyfikacje

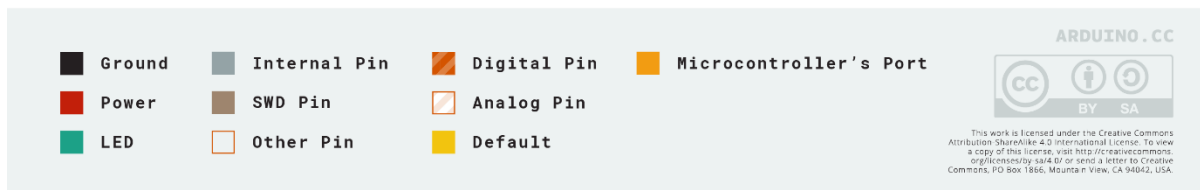
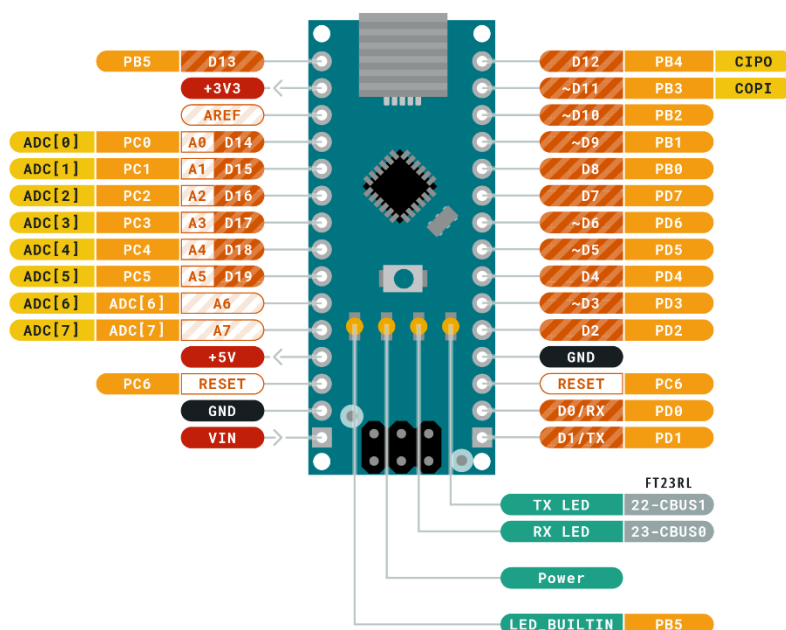
- Dodanie elementów z punktu 0.
- Zwiększenie zakresu obrotu urządzenia
- Hermetyczne zamknięcie elementów elektronicznych urządzenia w celu zabezpieczenia przed cieczą.
- Dodanie czujników wiatru
- Zasilanie urządzenia z panelu fotowoltaicznego
- Przeskalowanie projektu, wykorzystanie mocniejszych silników krokowych oraz zastosowanie przekładni ślimakowej w celu utrzymania większej masy na powierzchni urządzenia w postaci paneli fotowoltaicznych.

Bibliografia

- [1] „Jak działają panele fotowoltaiczne>,” [Online]. Available: <https://www.ekoradcy.pl/blog/jak-dziala-panel-fotowoltaiczny>. [Data uzyskania dostępu: 18 01 2024].
- [2] „budowa panela fotowoltaicznego,” [Online]. Available: <https://solar-volt.pl/images/aktualnosci/ogniwo-fotowoltaiczne-schemat.jpg>. [Data uzyskania dostępu: 18 01 2024].
- [3] „Systemy śledzące słońce,” synezaoze, [Online]. Available: <https://syntezaoze.pl/blog/systemy-sledzace-slonce-solar-tracking-system/>. [Data uzyskania dostępu: 2024 01 14].
- [4] „solar tracker - parametry,” [Online]. Available: <https://solar-tracker.pl/oferta/solar-tracker-dso-2000/>. [Data uzyskania dostępu: 2024 01 14].
- [5] „Czujnik fotoelektryczny,” [Online]. Available: https://pl.wikipedia.org/wiki/Czujnik_fotoelektryczny. [Data uzyskania dostępu: 2024 01 14].
- [6] „Fotorezystor,” [Online]. Available: <https://pl.wikipedia.org/wiki/Fotorezystor>. [Data uzyskania dostępu: 14 01 2024].
- [7] „Dzielnik napięcia,” [Online]. Available: https://pl.wikipedia.org/wiki/Dzielnik_napi%C4%99cia. [Data uzyskania dostępu: 14 01 2024].
- [8] „Arduino Nano,” Botland, [Online]. Available: https://botland.com.pl/img/art/inne/14313_8.jpg. [Data uzyskania dostępu: 14 01 2024].
- [9] „Silnik krokowy 28BYJ-48,” [Online]. Available: <https://abc-rc.pl/pl/products/silnik-krokowy-28byj-48-z-przekladnia-5v-0-3a-0-03nm-arduino-6290.html>. [Data uzyskania dostępu: 2024 01 14].
- [10] „arduino funkcje,” [Online]. Available: <https://plociennik.info/index.php/funkcje?showall=1>. [Data uzyskania dostępu: 21 01 2024].
- [11] „Arduino Nano,” [Online]. Available: <https://docs.arduino.cc/hardware/nano>. [Data uzyskania dostępu: 14 01 2024].
- [12] „Przycisk krańcowe,” TME, [Online]. Available: https://www.tme.eu/pl/details/ess0511310/mikroprzelaczniki-snap-action/ece/?brutto=1¤cy=PLN&gad_source=1&gclid=CjwKCAiAqY6tBhAtEiwAHeRopXEIJ28vUOZb85Sxt0nPdCrb3gKmqa-QNuHO-x0njubin3vzLnHpGhoCWg4QAvD_BwE. [Data uzyskania dostępu: 14 01 2024].

Dodatki

Dodatek A - Opis wyprowadzeń Arduino Nano



Rysunek 16. Arduino nano rozmieszczenie wyprowadzeń na płytce [11]

Dodatek B – Cały kod programu

```
1. //Light tracker, urządzenie służące do śledzenia światła
2. //wykorzystywane porty analogowe od 1 do 4 oraz cyfrowe od 2 do 13
3. //kod powstał 13.01.2024r
4. //Autorzy: Wiktoria Golinowska, Jakub Wrona, Wiktor Równanek

5. //Zaimportowanie biblioteki do sterowania silnikami
6. #include <Stepper.h>
7.
8. // Definicje pinów dla fotorezystorów
9. int LG = A1;
10. int PG = A2;
11. int PD = A3;
12. int LD = A4;
13. // Definicje pinów dla przycisków krańcowych
14. int PionPanel = 9;
15. int PionCzujniki = 8;
16. int PrawyMax = 10;
17. int LewyMax = 11;
18. // Zmienne przechowujące informacje o kierunku ruchu silników
19. int H,V;
20. // Stała definiująca ilość kroków na jeden pełny obrót silnika
21. const int kroki_na_obrot = 2048;
22.
23. // Obiekty dla silników krokowych
24. Stepper Poziom(kroki_na_obrot, 13, 2, 12, 3);
25. Stepper Pion(kroki_na_obrot, 4, 6, 5, 7);
26.
27. // Funkcja kalibracji początkowej dla silnika pionowego
28. void kalibracjaPion()
29.     while (digitalRead(PionCzujniki) == HIGH) {
30.         Pion.step(-1);
31.     }
32. }
33. // Funkcja kalibracji początkowej dla silnika poziomego
34. void kalibracjaPoziom() {
35.     while (digitalRead(PrawyMax) == HIGH) {
36.         Poziom.step(-1);
37.     }
38.     Poziom.step(400);
39. }
40. // Funkcja wyłączająca silnik pionowy
41. int wylaczPion(){
42.     digitalWrite(4, LOW);
43.     digitalWrite(5, LOW);
44.     digitalWrite(6, LOW);
45.     digitalWrite(7, LOW);
46.     return 0;
47. }
```

Rysunek 17. Pierwsza część kodu

```

48. // Funkcja wyłączająca silnik poziomy
49. int wylaczPoziom() {
50.   digitalWrite(13, LOW);
51.   digitalWrite(2, LOW);
52.   digitalWrite(12, LOW);
53.   digitalWrite(3, LOW);
54.   return 0;
55. }
56.
57. void setup() {
58.   // Konfiguracja pinów jako wejścia
59.   pinMode(LG, INPUT);
60.   pinMode(PG, INPUT);
61.   pinMode(PD, INPUT);
62.   pinMode(LD, INPUT);
63.   pinMode(PionPanel, INPUT_PULLUP);
64.   pinMode(PionCzujniki, INPUT_PULLUP);
65.   pinMode(PrawyMax, INPUT_PULLUP);
66.   pinMode(LewyMax, INPUT_PULLUP);
67.
68.   // Ustawienie prędkości obrotowej silników
69.   Poziom.setSpeed(8);
70.   Pion.setSpeed(8);
71.
72.   // Wywołanie funkcji kalibracji dla silnika pionowego i poziomego
73.   kalibracjaPion();
74.   kalibracjaPoziom();
75. }
76.
77. void loop() {
78.   //Odczyt wartości z fotorezystorów i obliczenie ich średniej wartości
79.   int gora = (analogRead(LG) + analogRead(PG)) / 2;
80.   int dol = (analogRead(LD) + analogRead(PD)) / 2;
81.   int prawo = (analogRead(PD) + analogRead(PG)) / 2;
82.   int lewo = (analogRead(LG) + analogRead(LD)) / 2;
83.
84.   // Obliczenie wartości składowych ruchu (pionowy i poziomy)
85.   V = gora - dol;
86.   H = prawo - lewo;
87.
88.   // Kontrola ruchu silnika pionowego uwzględniająca przyciski krańcowe
89.   if (digitalRead(PionPanel) == LOW) {
90.     Pion.step(V > 10 ? -2 : wylaczPion());
91.   } else if ((digitalRead(PionCzujniki) == LOW) and (V < -100)) {
92.     Poziom.step(300);
93.   } else if (digitalRead(PionCzujniki) == LOW) {
94.     Pion.step(V < -10 ? 2 : wylaczPion());
95.   } else {Pion.step(V > 10 ? -1 : (V < -10 ? 1 : wylaczPion()));
96.   }

```

Rysunek 18. Druga część kodu


```
97. // Kontrola ruchu silnika poziomego uwzględniająca przyciski krańcowe
98. if ((digitalRead(PrawyMax) == LOW) and (H > 100)) {
99.     Poziom.step(1500);
100. } else if (digitalRead(PrawyMax) == LOW) {
101.     Poziom.step(H > 10 ? 2 : wylaczPoziom());
102. } else if ((digitalRead(LewyMax) == LOW) and (H < -100)) {
103.     Poziom.step(-1500);
104. } else if (digitalRead(LewyMax) == LOW) {
105.     Poziom.step(H < -10 ? -2 : wylaczPoziom());
106. } else {
107.     Poziom.step(H > 10 ? -2 : (H < -10 ? 2 : wylaczPoziom()));
108. }}
```

Rysunek 19- Trzecia część kodu