

Politechnika Świętokrzyska w Kielcach

Wydział Mechatroniki i Budowy Maszyn

Teoria sterowania – zadanie projektowe

Projekt układu optymalizacji produkcji energii elektrycznej poprzez sterowanie położeniem panelu fotowoltaicznego.

Autorzy:

inż. Emil Bugajski

inż. Grzegorz Soch

Prowadzący:

mgr inż. Marta Grzyb

Kielce, 2022

Zawartość

1. Wprowadzenie.....	3
1.1. Obiekt badań.....	3
2. Wstęp teoretyczny	3
2.1. Przegląd dostępnych rozwiązań.....	4
2.2. Założenia projektowe.....	5
3. Projektowany model – część mechaniczna.....	6
4. Projektowany model – Hardware	6
4.1. Raspberry Pi Pico	6
4.1.1 Problem multipleksowania ADC	7
4.2. Napędy	7
4.2.1 Napęd osi X.....	9
4.2.2 Napęd osi Y	9
4.3. Panel fotowoltaiczny	10
4.4. Woltomierz cyfrowy	10
4.5. Sensor.....	11
5. Projektowany model – Software	12
5.1. Algorytm różnicowy	12
5.2. Dobrany regulator.....	13
5.3. Wybrane fragmenty programu	14
6. Podsumowanie i wnioski	17
6.1. Możliwa rozbudowa i udoskonalenia układu.....	18
7. Literatura.....	19

1. Wprowadzenie

Efekt fotowoltaiczny po raz pierwszy zaobserwowany został w 1839 roku przez Aleksandra Edmunda Becquerel'a. Zjawisko polega na powstaniu siły elektromotorycznej w jednorodnym półprzewodniku pod wpływem oświetlenia i jest wynikiem zachodzących procesów fizycznych. Aby zobrazować wielkość energii pochodzącej ze Słońca należy podkreślić że wynosi ona około $178 \cdot 10^{15} \text{ W}$. Jest więc około 30000 razy większa niż całkowita moc wszystkich urządzeń zainstalowanych na całej kuli Ziemskiej[1]. Panele fotowoltaiczne stają się coraz bardziej powszechne i stanowią istotny punkt we współczesnej technologii. Niniejsza praca zawiera proces projektowania, wykonania i zaimplementowania algorytmu nadążnej regulacji panelu słonecznego o małej mocy.



Rysunek 1. Rysunek poglądowy panelu fotowoltaicznego.

1.1. Obiekt badań

Obiektem badań jest układ regulacji nadążnej podążający za słońcem. Zadanie układu sterowania polega na optymalizacji wielkości wyprodukowanej energii poprzez algorytm ustawiający panel prostopadłe do słońca. Według niektórych badań możliwy wzrost energii elektrycznej przy zastosowaniu tego typu układu może wynosić nawet 40% uwzględniając energię potrzebną do poruszania panelem słonecznym. [1] Układy nadążne wraz ze wzrostem popularności paneli stają się coraz bardziej powszechne.

2. Wstęp teoretyczny

Celem pracy jest zaprojektowanie platformy dla panelu fotowoltaicznego o dwóch stopniach swobody, samoczynnie podążającej za słońcem. Zadanie obejmuje zarówno projekt mechaniczny jak i algorytm sterowania. Elementem sensorycznym układu będzie czujnik składający się z czterech fotorezystorów oraz prostopadłych ścian przesłonowych opisanych w podrozdziale 4.5. Całość napędzana będzie dwoma serwonapędami. Układem

sterował będzie 32-bitowy mikrokontroler z rodziny ARM z serii Cortex M0+ RP2040. W celu minimalizacji kosztów serwonapędu przyjęto, że maksymalny zakres ruchu azymutalnego słońca to 180° z powody wysokiej ceny serw o większym zakresie ruchu.

2.1. Przegląd dostępnych rozwiązań

Na rynku dostępnych jest wiele komercyjnych jak i hobbystycznych systemów nadążnych do paneli fotowoltaicznych. Każdy z nich opiera się o jedną z dwóch metod sterowania takim układem. Pierwsza z nich polega na wyliczeniu aktualnej pozycji słońca na niebie, które możliwe jest gdy znane są następujące parametry:

- szerokość geograficzna,
- deklinacja słońca,
- wysokość,
- azymut,
- dzień roku,
- czas.

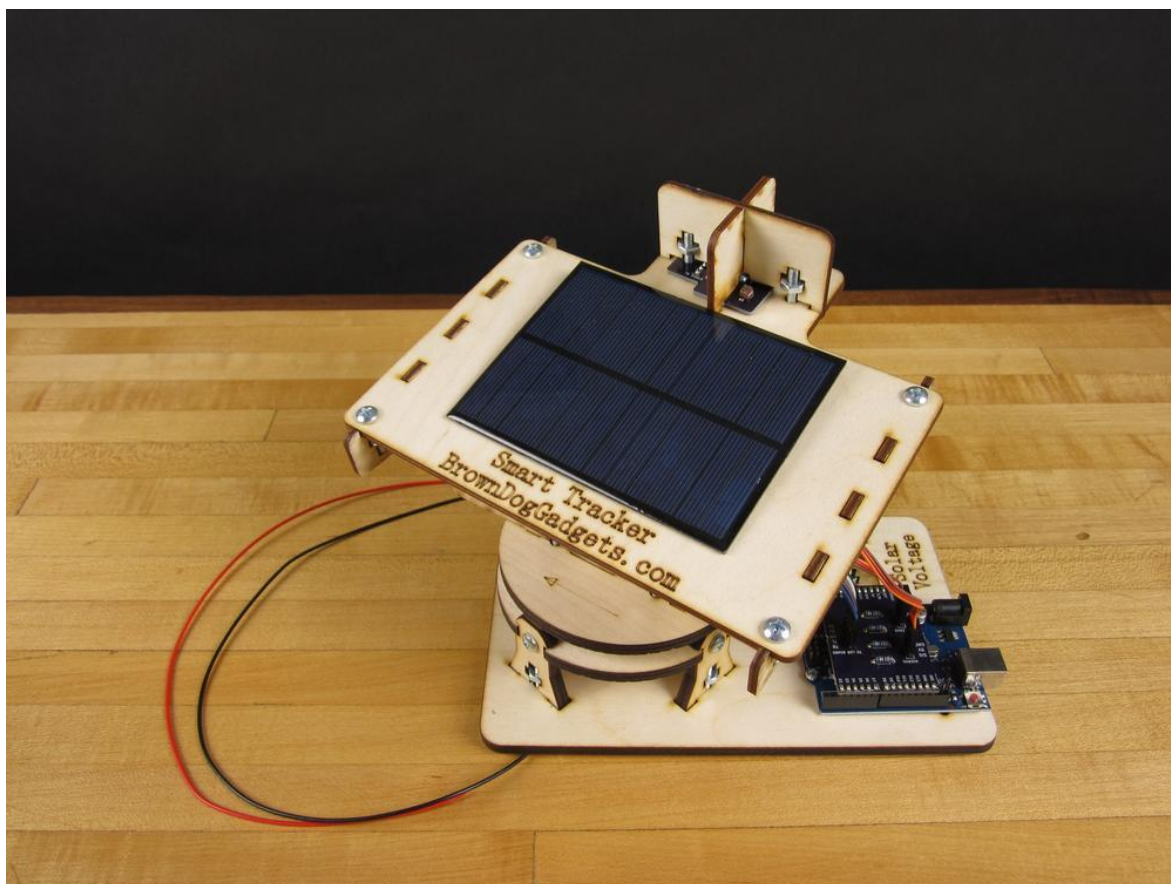
Drugim podejściem jest zastosowanie układu pomiarowego składającego się z czterech fotorezystorów. Są to elementy półprzewodnikowe, których rezystancja ulega zmianie pod wpływem padającego na jego powierzchnię promieniowania elektromagnetycznego na przykład promieniowania widzialnego lub podczerwieni. Rezystancja elementu zależy od natężenia oświetlenia fotorezystora, jego rezystancja w ciemności jest bardzo duża i może osiągnąć wartość rzędu megaomów, a przy silnym oświetleniu może zmaleć do kilku omów. Fotorezystory przy zastosowaniu odpowiednich przesłon umożliwiają wyliczenie kąta padającego światła.

Przykładem komercyjnego zastosowania układu sterowania położeniem panelu jest *Smart Flower* wyprodukowany przez firmę *Smartflower Solar*. Rozwiązanie to oparte jest na wyliczaniu położenia słońca z zależności matematycznej.



Rysunek 2. *Smart Flower* wyprodukowany przez firmę *Smartflower Solar*. [2]

Temat ten jest popularny wśród hobbystów i możemy znaleźć wiele rozwiązań opartych głównie na mikrokontrolerach i zastosowaniu układu pomiarowego na bazie fotorezystorów.



Rysunek 3. Przykładowy układ nadążny za słońcem. [3]

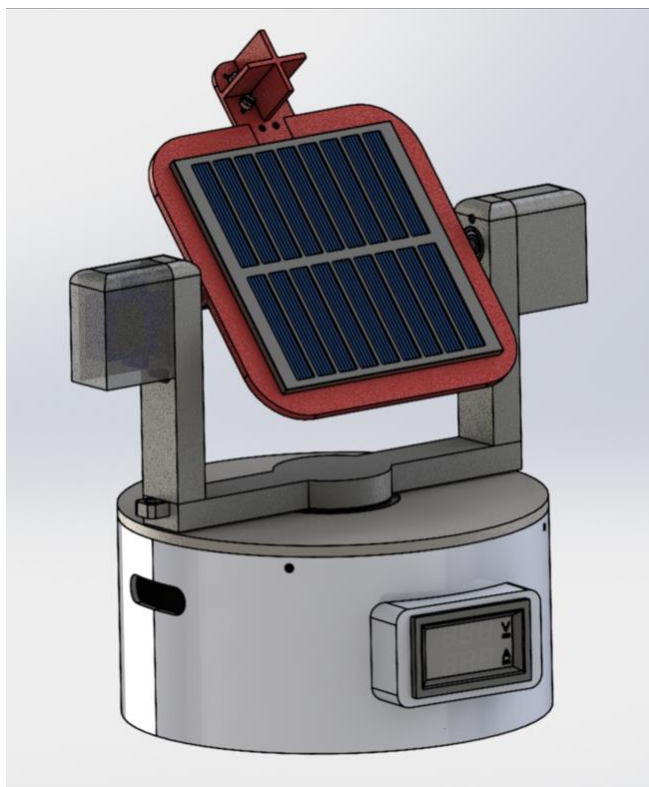
Pokazany na rysunku 3 projekt wykorzystuje 4 fotorezystory do wykrywania kąta padania światła. Platforma na której umiejscowiony jest panel fotowoltaiczny sterowana jest w dwóch osiach za pomocą dwóch serwonapędów. Cały układ sterowany jest za pomocą mikrokontrolera Arduino.

2.2. Założenia projektowe

Założono, że słońce porusza się o maksymalny kąt w poziomie 180° . W praktyce kąt ten jest dużo większy w lecie, jednak ograniczał nas zakres ruchu serwa. Nie wpływa to w żaden sposób na algorytm sterowania. Przyjęto również, że układ w momencie uruchamiania ustawiony będzie w kierunku południowym, tak aby zakres ruchu serwomechanizmu obejmował jak największy przedział czasu padania promieni słonecznych.

3. Projektowany model – część mechaniczna

Wszystkie elementy konstrukcyjne z wyłączeniem łożysk zostały zaprojektowane w programie SOLIDWORKS, a następnie wykonane w technologii druku 3D. Celem konstrukcji było zapewnienie dwóch stopni swobody platformy panelu, umożliwienie przekazania napędu oraz poprawienie walorów wizualnych projektu przez poprowadzenie przewodów oraz schowanie elektroniki w obudowie podstawy. Przestrzenny model złożeniowy projektowanego układu zaprezentowano na rysunku 4.



Rysunek 4. Wizualizacja układu stworzona w oprogramowaniu SOLIDWORKS.

4. Projektowany model – Hardware

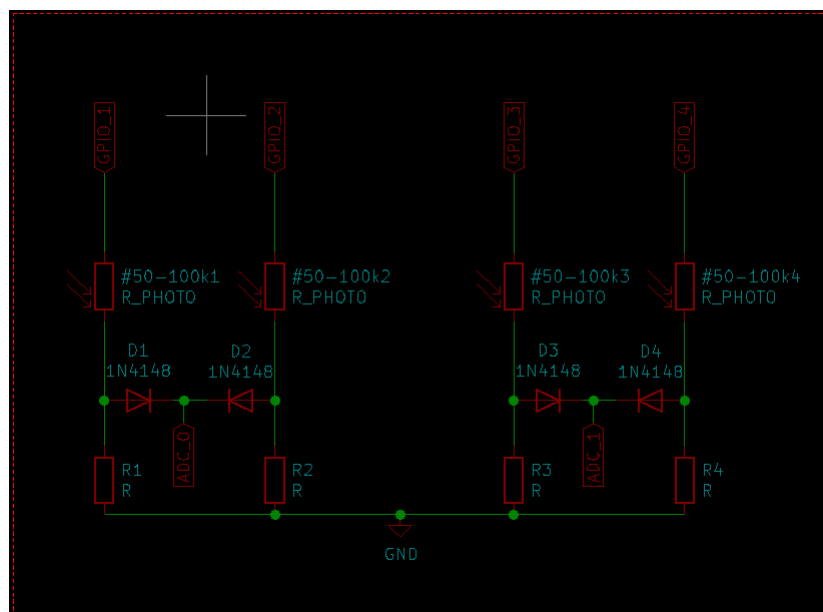
Układ elektryczny zmontowano na płytce prototypowej oraz umieszczono w podstawie modelu. Przewody poprowadzono po jednym z ramion obrotnicy tak aby znajdowały się jak najbliżej osi obrotów i były jak najmniej podatne na zginanie.

4.1. Raspberry Pi Pico

Mikrokontroler sterujący układem to moduł Raspberry Pi Pico wyposażony w dwurdzeniowy układ RP2040. Jest to oparty na architekturze ARM Cortex M0+ o taktowaniu maksymalnym do 133 MHz. Zdecydowano się na ten model, ze względu na możliwość łatwego programowania w języku MikroPython.

4.1.1 Problem multipleksowania ADC

Zaprojektowany czujnik kąta padania promieni świetlnych zbudowany jest z 4 fotorezystorów. Niezbędna jest więc możliwość odczytu każdego z osobna. Niestety dobrany mikrokontroler posiada jedynie 3 piny mogące obsłużyć przetwornik analogowo – cyfrowy. W związku z powyższym konieczne było zaprojektowanie multipleksera ADC. To rozwiązanie pozwala na obsłużenie kilku fotorezystorów jednym kanałem. Schemat układu wykorzystującego prosty dzielnik napięcia oraz układ multipleksowania przedstawiono na rysunku 5.



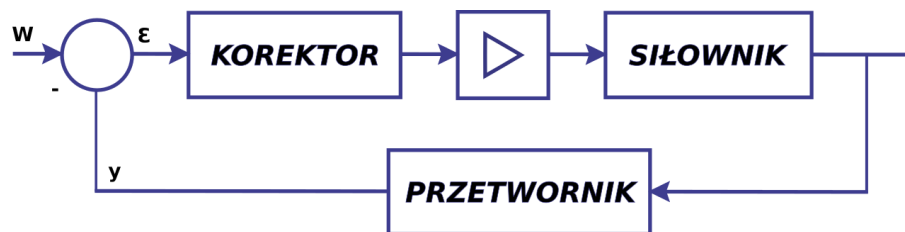
Rysunek 5. Układ multipleksowania pomiarów analogowych.

Każdy fotorezystor (xR_PHOTO) zasilany jest bezpośrednio z pinu mikrokontrolera ($GPIO_x$). Pozwala to na softwareowe wybranie czujnika z którego chcemy dokonać pomiaru. Układ wyposażono w cztery diody (D1-D4) w kierunku zaporowym tak aby w trakcie pomiaru prąd płynął wyłącznie przez zasilany sensor. Trzeba mieć jednak świadomość wady takiego rozwiązania. Na diodzie występuje spadek napięcia, dodatkowo każda dioda jak i rezystor posiada pewną tolerancję wykonania, co prowadzi do szumu pomiarowego.

4.2. Napędy

Mając na uwadze charakterystykę pracy napędów wykorzystanych do sterowania poszczególnymi osiami, którego zakresy przemieszczenia kątownego wahają się w granicach $\pm 90^\circ$ i nie mają potrzeby rozwijania dużych prędkości jak i momentów obrotowych. Działające na napęd siły są stosunkowo niewielkie, zachodzi jedynie potrzeba precyzyjnego pozycjonowania. Zdecydowano się więc na serwomechanizmy.

Serwomechanizm jest to zamknięty układ sterowania (układ regulacji) ze sprzężeniem zwrotnym, w którym sygnałem wejściowym jest (w naszym przypadku) położenie. Wartość wzorcowa porównywana jest z przetworzonym przez przetwornik bieżącym sygnałem wyjściowym i powstały w ten sposób uchyb podawany jest na człon korekcyjny, a dalej na wzmacniacz. Wzmocniony sygnał trafia do siłownika (może to być odpowiedni silnik elektryczny), którego przemieszczenie jest wartością wyjściową układu. Zadaniem serwomechanizmu jest likwidacja błędów regulacji (uchybu przemieszczenia), powstających na skutek zmian wielkości wzorcowej, a więc klasyfikujemy go jako układ nadążny. Serwomechanizm ma strukturę typowego układu regulacji, nie steruje jednak obiektem technologicznym, lecz siłownikiem w celu usprawnienia działania toru wykonawczego [4]. Schemat blokowy serwomechanizmu pokazany jest na rysunku 6.



Rysunek 6. Schemat blokowy działania serwomechanizmu [4]

Typowe serwomechanizmy modelarskie, które zostały zastosowane w projekcie sterowane są przez mikrokontroler przy pomocy regulacji sygnału napięciowego PWM (ang. *Pulse-Width Modulation*). Jest to sygnał o stałej amplitudzie i częstotliwości a zmiennym wypełnieniu. Serwomechanizm posiada wbudowany sterownik silnika (analogowy lub cyfrowy), który odczytuje wysyłany przez mikrokontroler sygnał sterujący i na podstawie szerokości dostarczanego impulsu ustawia swoją pozycję.

Przyjęto układ współrzędnych zgodnie z poniższym rysunkiem



Rysunek 6. Umowna lokalizacja położenia układu współrzędnych panelu fotowoltaicznego.

4.2.1 Napęd osi X

Jako napęd osi X dobrano serwomechanizm *PowerHD HD-1160A*. Wybrano je ze względu na zwiększony moment obrotowy wynoszący 2 [kg·cm]. Serwonapęd pokazany jest na rysunku 7, a jego podstawowe parametry zebrane są w tabeli 1.



Rysunek 7. Serwonapęd PowerHD HD-1160A. [5]

Tabela 1. Specyfikacja serwonapędu PowerHD HD-1160A.

Napięcie zasilania	4.8 – 6.0 V
Zakres ruchu	0 – 180 °
Moment	2.0 kg·m (0.19 Nm)
Prędkość	0.12 s/60°
Maksymalny pobór prądu	680 mA
Typ	Analogowe
Częstotliwość sygnału sterującego	50 Hz
Maksymalny zakres impulsów	800 – 2200 μs
Wymiary	29 x 12 x 30.2 mm
Masa	16 g

4.2.2 Napęd osi Y

Napęd osi Y stanowi serwonapęd *Tower Pro SG-90*. Wybrany został ze względu na mniejsze wymiary gabarytowe oraz małą wagę. Serwomechanizm pokazany jest na rysunku 8, a jego podstawowe dane techniczne zebrane są w tabeli 2.



Rysunek 8. Serwonapęd Tower Pro SG-90. [6]

Tabela 2. Specyfikacja serwonapędu Tower Pro SG-90.

Napięcie zasilania	4.8 – 6.0 V
Zakres ruchu	0 – 180 °
Moment	1.8 kg·m (0.18 Nm)
Prędkość	0.12 s/60°
Typ	analogowe
Wymiary	22 x 12 x 27 mm
Masa	9 g

4.3. Panel fotowoltaiczny

Dobrano ogniwo słoneczne małej mocy, którego aktualne napięcie wyjściowe jest mierzone i wyświetlane za pomocą woltomierza cyfrowego. Podstawowe parametry ogniwa przedstawiono w tabeli 3.



Rysunek 9. Panel fotowoltaiczny. [7]

Tabela 3. Dane techniczne panelu fotowoltaicznego.

Moc	0.7 W
Napięcie wyjściowe	5.5 V
Wymiary	95 x 95 x 3 mm

4.4. Woltomierz cyfrowy

Wyświetlanie aktualnego napięcia wyjściowego ogniwa słonecznego odbywa się przy pomocy miernika cyfrowego z wyświetlaczem. Pozwala on na pomiar napięcia oraz natężenia prądu z częstotliwością 2 Hz.

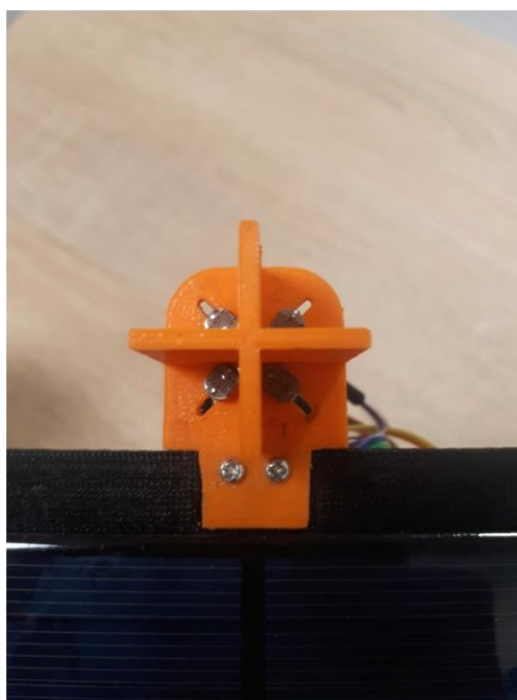


Rysunek 10. Miernik cyfrowy [8]

Miernik pozwala na pomiar napięcia stałego w zakresie 0 – 99.9 V oraz natężenia prądu od 0 do 50 A. Zasilanie części logicznej waha się od 4.5 – 30 V. Deklarowana dokładność pomiarowa wynosi 1%.

4.5. Sensor

W celu wyznaczenie kierunku padania promieni świetlnych zaprojektowano czujnik składający się z 4 jednakowych fotorezystorów przedzielonych przegrodami. Przegrody mają na celu rzucenie cienia na fotorezystory co z kolei wytworzy różnicę ich wskazań. Fotorezystor na którego rzucony został cień będzie miał większą rezystancję. Dzięki tej zależności możemy wyznaczyć w którą stronę należy obracać układ. Będzie to nasz sygnał sterujący do zaimplementowanego regulatora.



Rysunek 11. Układ pomiarowy złożony z czterech fotorezystorów oddzielonych przegrodami.

Bardzo ważnym aspektem jest aby czujnik zamontowany był równolegle do panelu. Każda odchyłka równoległości płaszczyzn spowoduje niedokładność regulacji.

Wadą takiego sensora jest fakt, że uszkodzenie jednego fotorezystora spowoduje zakłócenie pracy układu. Zdiagnozowanie uszkodzenia czujnika jest bardzo trudne do wykrycia w programie. Dodatkowo fotorezystory posiadają swoją dokładność. Korzystna byłaby ich kalibracja jednak wymaga to stworzenia idealnych warunków oświetleniowych, co też nie jest łatwe.

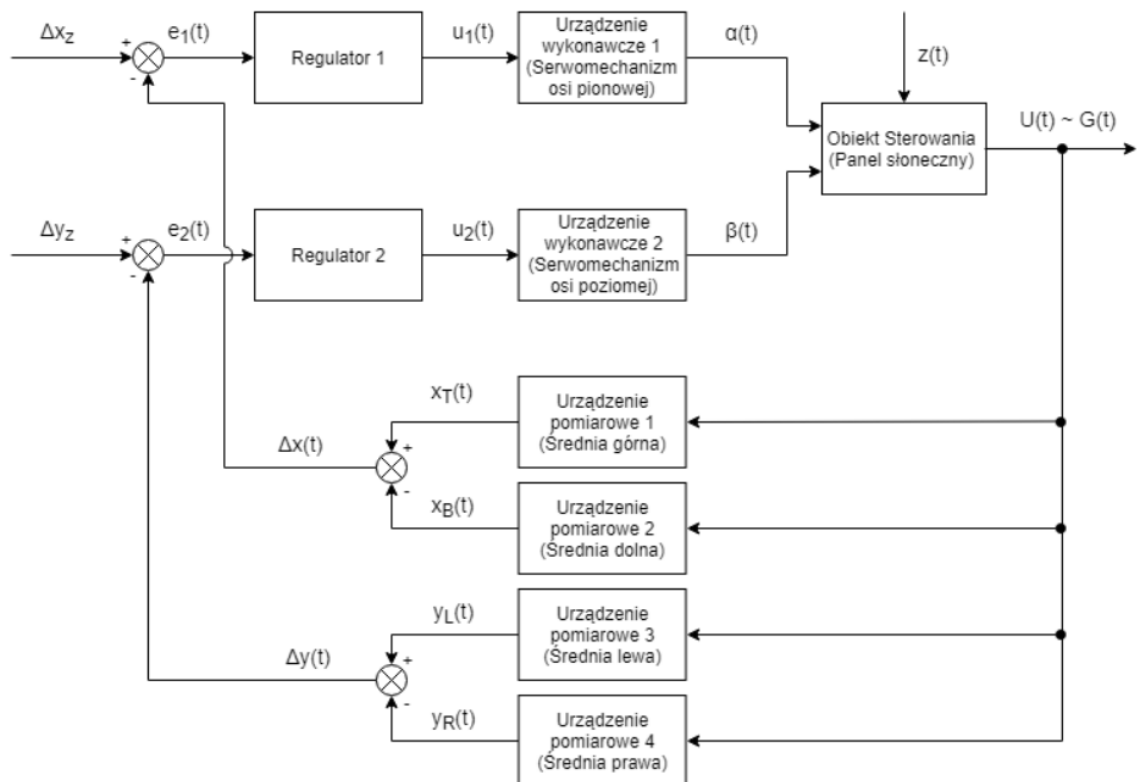
5. Projektowany model – Software

Najczęściej stosowanymi algorytmami sterowania położenia panelu fotowoltaicznego są algorytmy różnicowe i zegarowe. W realizowanym projekcie zdecydowano się na implementację algorytmu różnicowego. Skrypt napisano w języku *microPython*.

5.1. Algorytm różnicowy

Algorytm różnicowy jest podstawowym automatycznym algorytmem działającym w pętli sprzężenia zwrotnego. Ideą jego działania jest utrzymywanie pozycji panelu słonecznego w taki sposób, aby uzyskać i utrzymać jak największą ilość padającego na niego światła. Dzięki temu możliwa jest stała optymalizacja pracy panelu, co z kolei przekłada się na większą ilość uzyskiwanej energii. Zasadą działania algorytmu różnicowego jest praca w oparciu o dane zbierane przez urządzenia pomiarowe – fotorezystory. Przyjęto po jednej parze czujników na każdą z osi: pionową i poziomą. Układ ten możemy więc podzielić na dwa mniejsze układy regulacji dla każdej osi.

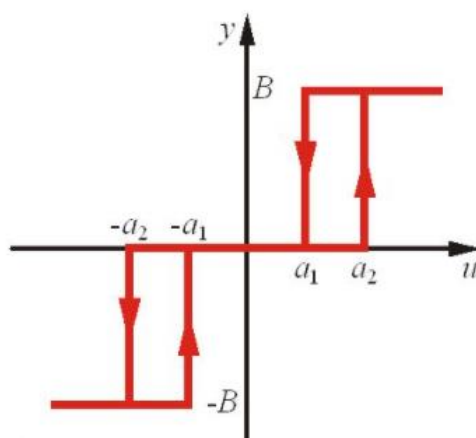
Największa produkcja energii elektrycznej jest wtedy gdy promienie świetlne padają prostopadle do płaszczyzny PV. W idealnym przypadku wskazania wszystkich fotorezystorów byłyby więc jednakowe. W praktyce zerowa różnica wskazań powodowałoby ciągłą pracę układów napędowych, co generowałoby niepotrzebne straty energii. Ma to związek z określoną czułością fotodetektorów. Aby temu zapobiec wprowadzono histerezę. Algorytm regulacji różnicowej przedstawiono na rysunku 12.



Rysunek 12. Schemat blokowy przedstawiający zasadę działania układu[9].

5.2. Dobrany regulator

Do sterowania położeniem serw zaimplementowano w mikrokontrolerze dwa regulatory trójstanowe, mogący przyjmować stany: zmiany położenia w lewo, braku ruchu, zmiany położenia w prawo po jednym dla każdej osi(x,y). Taki regulator powoduje pracę krokową układu, ale w założeniu prędkość obrotu słońca jest niewielka. Nastawę histerezy przyjęto doświadczalnie.



Rysunek 13. Zasada działania histerezy.

5.3. Wybrane fragmenty programu

Cały kod programu zawarty jest w załączniku do niniejszego projektu. Poniżej przedstawiono najważniejszą funkcję oraz omówiono ich działanie.

read_ADC(photo_resistor_number)

Funkcja obsługuje odczyt wartości napięcia z przetworników analogowo – cyfrowych z zastosowaniem softwarowego multipleksowania. Jako argument przyjmuje numer fotorezystora dla którego chcemy odczytać, załącza stan wysoki dla wybranego czujnika oraz niski dla pozostałych. Zwracana jest wartość uśredniona z 1000 pomiarów.

```

1  def read_ADC(photo_resistor_number):
2      mean=0
3
4      if photo_resistor_number ==1:
5          photo_res_1.on()
6          photo_res_2.off()
7          photo_res_3.off()
8          photo_res_4.off()
9          sleep(0.001)
10         for i in range(1,1000):
11
12             mean+=ADC_0.read_u16()/1000
13
14             return mean
15
16     if photo_resistor_number ==2:
17         photo_res_1.off()
18         photo_res_2.off()
19         photo_res_3.off()
20         photo_res_4.on()
21         sleep(0.001)
22         for i in range(1,1000):
23
24             mean+=ADC_0.read_u16()/1000
25             return mean
26
27     if photo_resistor_number ==3:
28         photo_res_1.off()
29         photo_res_2.off()
30         photo_res_3.on()
31         photo_res_4.off()
32         sleep(0.001)
33         for i in range(1,1000):
34
35             mean+=ADC_1.read_u16()/1000
36
37             return mean
38
39     if photo_resistor_number ==4:
40         photo_res_1.off()
41         photo_res_2.on()
42         photo_res_3.off()
43         photo_res_4.off()
44         sleep(0.001)
45         for i in range(1,1000):
46
47             mean+=ADC_1.read_u16()/1000
48
49             return mean

```

Program 1. Funkcja `read_ADC` mierząca wartość dla każdego fotorezystora

Funkcja **check_rotate** sprawdza czy wystąpiła różnica wskazań poszczególnych fotorezystorów, oraz czy różnica jest większa od histerezy. Jeżeli tak to zmieniana jest wartość **position_X** która jest sygnałem sterującym dla serwonapędu osi X

```
47 def check_rotate(a,b,c,d):
48     global histerese
49     global position_X
50     if (a+c-histerese) > (b+d):
51         if position_X < 90:position_X -=1
52     if (b+d-histerese) > (a+c):
53         if position_X > -90:position_X +=1
54     print('a+c',a+c,' ? b+d',b+d)
55     return 0
56
```

Program 2. Funkcja **check_rotate** zadająca sygnał sterujący dla serwonapędu osi X.

Funkcja **check_decline** działa analogicznie tylko realizuje sterowanie dla osi Y.

```
57 def check_decline(a,b,c,d):
58     global histerese
59     global position_Y
60     if (a+b-histerese)>(c+d):
61         if position_Y < 90:position_Y +=1
62     if (c+d-histerese)>(a+b):
63         if position_Y > 0:position_Y -=1
64     print('a+b',a+b,' ? c+d',c+d)
65     return 0
66
```

Program 3. Funkcja **check_decline** zadająca sygnał sterujący dla serwonapędu osi Y.

Poniżej przedstawiono pętlę główną programu. Algorytm w pierwszej kolejności wywołuje funkcje **read_ADC** czytające wartości przetworników dla każdego fotorezystora i przypisuje je do zmiennych, obliczana jest maksymalna, minimalna, średnia wartość oraz rozstęp. Nadpisywana jest wartość histerezy(przygotowanie do późniejszej możliwości zmiany histerezy w trakcie pracy układu). Wywoływane są funkcję **check_rotate** oraz **check_decline** wystawiające sygnał zadany dla osi X i Y, następnie wyświetlana jest paczka danych oraz wypracowywane są pozycje obydwu osi.


```

67 whileTrue:
68
69     ph1=read_ADC(1)
70     ph2=read_ADC(2)
71     ph3=read_ADC(3)
72     ph4=read_ADC(4)
73
74     maximum = max_value(ph1,ph2,ph3,ph4)
75     minimum = min_value(ph1,ph2,ph3,ph4)
76     srednia = mean_value(ph1,ph2,ph3,ph4)
77     rozstep = maximum - minimum
78
79     histerese =800
80
81     check_rotate(ph1,ph2,ph3,ph4)
82     check_decline(ph1,ph2,ph3,ph4)
83
84     print('RES_1',ph1)
85     print('RES_2',ph2)
86     print('RES_3',ph3)
87     print('RES_4',ph4)
88     print('_____')
89     print('srednia',srednia)
90     print('max      ',maximum)
91     print('min      ',minimum)
92     print('rozstep',rozstep)
93     print('_____')
94     work_rotate(position_X)
95     work_decline(position_Y)
96
97

```

Program 4. Pętla główna programu.

6. Podsumowanie i wnioski

Zrealizowano cel pracy jakim było zaprojektowanie i wykonanie układu optymalizującego produkcję energii elektrycznej. Przetestowano model i stwierdzono poprawność jego działania. Złożony model przedstawiono na Rys.14 Układ zachowuje się zgodnie z założeniami projektowymi.



Rysunek 14. Widok zbudowanego prototypu.

Istnieje kilka metod sterowania położeniem panelu fotowoltaicznego. W niniejszej pracy zastosowano algorytm różnicowy. Badania modelu wykazały poprawność działania zaimplementowanego algorytmu. Dużą wadą tego rozwiązania jest fakt, że uszkodzenie jednego fotorezystora wprowadza znaczące zakłócenia pracy układu a taka sytuacja jest trudna do wykrycia.

Zaproponowane przełożenie napędu osi poziomej wymaga przeprojektowania lub wykonania z bardziej odpornego na skręcanie materiału.

Gotowy model różni się nieco od projektu za sprawą fimezji osób obsługujących drukarkę 3D, które z niewiadomych przyczyn do każdej części użyły innego koloru filamentu.

Układ może być w łatwy sposób rozbudowywany na wielu płaszczyznach.

6.1. Możliwa rozbudowa i udoskonalenia układu

Układ zaprojektowano tak aby w przyszłości mógł być łatwo rozbudowywany. Pozostawiono jeden wolny kanał przetwornika analogowo – cyfrowego, który może być w wykorzystany do przyłączenia potencjometru, którym zmieniana byłaby by wartość histerezy. Byłoby to bardzo edukacyjne rozwiązanie móc zaobserwować zachowanie układu zmieniając w trakcie jego pracy histerezę. Można pokusić się o programowe wyliczanie optymalnej histerezy w zależności od parametrów: średniej, mediany, rozstępu lub innych parametrów statystycznych.

Warto w przyszłości rozbudować układ o algorytm sterowania zegarowego, wymaga to jednak dołączenia zegara czasu rzeczywistego.

W przypadku dalszych badań tego typu układów bardzo dobrym rozwiązaniem byłoby zaproponowanie wewnętrznego prowadzenia przewodów elektrycznych w osiach obrotu wykorzystując złącza rotacyjne. Umożliwiłoby to pozbycie się nieestetycznych i podatnych na uszkodzenia przewodów elektrycznych będących poza obrysem urządzenia.

7. Literatura

- [1] Trzasko W.: Analiza wydajności dwuosiowego solarnego układu nadążnego. Pomiary Automatyka Robotyka Nr 1/2018
- [2] <https://smartflower.com/>
- [3] <https://www.browndoggadgets.com/products/dual-axis-smart-solar-tracker>
- [4] <https://pl.wikipedia.org/wiki/Serwomechanizm>
- [5] <https://botland.com.pl/serwa-typu-medium/2303-serwo-powerhd-hd-1160a-medium-6953524540047.html>
- [6] <https://botland.com.pl/serwa-typu-micro/16561-serwo-okystar-sg-90-micro-180-5904422326159.html>
- [7] <https://botland.com.pl/panele-sloneczne-malej-mocy/3695-ogniwo-sloneczne-07w55v-95x95x3mm-5904422302856.html>
- [8] <https://a.allegroimg.com/s720/1120e2/2acd76034f8f982e852ad468781e/Miernik-napiecia-amperomierz-0-100V-50A-Woltomierz>
- [9] Hubert Sawicki, Robert Piotrowski „Zaprojektowanie, wykonanie i sterowanie panelem słonecznym” („Rynek Energii” – 3/2019)