**Elektronika i oprogramowanie urządzenia**

**Wstęp:**

Głównym zadaniem elektroniki i zaimplementowanego programu jest sterowanie silnikami krokowymi w taki sposób, by teleskop ustawił się na konkretną gwiazdę, a następnie ją śledził w miarę jej ruchu po nocnym niebie. W tym rozdziale będą omówione podstawy teoretyczne, które posłużą do napisania algorytmów sterowania silnikami, zaprojektowany układ elektroniczny, schemat blokowy sterowania silnikami oraz zademonstrowane zostanie działanie programu.

**Podstawy teoretyczne**

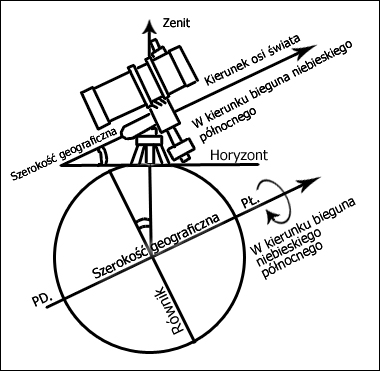
Uwagi do poniższego punktu: opis skupia się na prostym wytłumaczeniu zjawisk zachodzących podczas ruchu naszej planety bez zagłębiania się w szczegółowe zagadnienia i problemy jak różnica pomiędzy dobą słoneczną, a dobą gwiezdną albo nieliniowość czasu gwiezdnego. Takie podejście wynika z tego, że byłby potrzebny bardzo obszerny opis tych zjawisk, a i tak algorytm programu nie będzie ich brał pod uwagę ze względu na bardzo trudną implementację – przykładowo: doba gwiezdna trwa 23h i 56 minut albo wpływ położenia naszej planety w układzie słonecznym. Uznano, że błędy są pomijalnie małe oraz głównym celem jest obserwacja odległych gwiazd, a nie np. Marsa.

Na poniższym rysunku 1 przedstawiono oznaczenie poszczególnych osi w teleskopie o montażu paralaktycznym.



Rysunek 1 Osie w teleskopie o montażu paralaktycznym. Latitude - szerokość, Polar Axis - oś biegunowa; Declination Axis - oś deklinacji

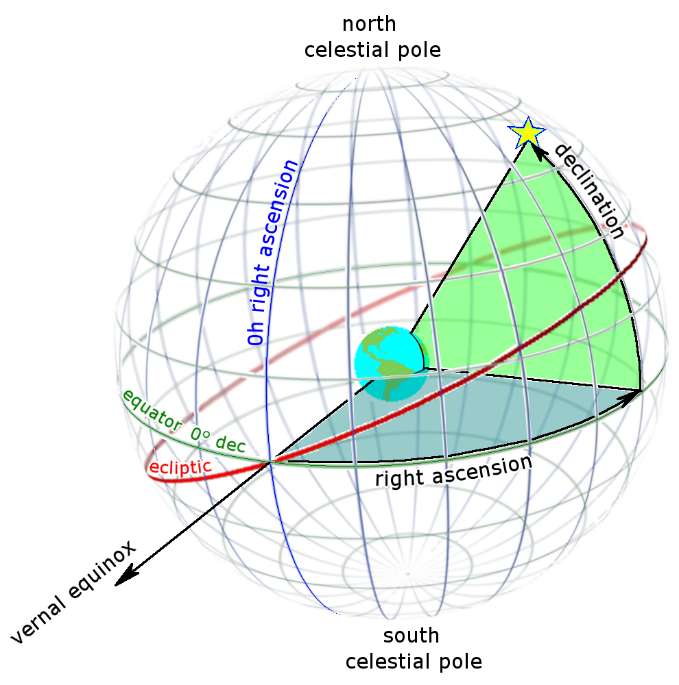
Kąt szerokości ustawiamy na zgodny z naszą szerokością geograficzną oraz kątem pochylenia teleskopu. Na rysunku 2 przedstawiono jakie ma znaczenie prawidłowe ustawienie tego kąta.



Rysunek 2 Zobrazowanie znaczenia ustawienie kąta szerokości

Oś deklinacji ustawiamy na wartość równą pozycji na nieboskłonie naszej obserwowanej gwiazdy. Na rysunku 3 przedstawiono zobrazowanie znaczenia ustawienia kąta deklinacji. Dzięki ustawieniu tych dwóch omówionych parametrów nasz teleskop jest nakierowany na wybrany przez nas punkt – obiekt astronomiczny. Niemniej ze względu na ciągły obrót kuli ziemskiej wokół własnej osi obserwujemy pozorny ruch gwiazd na niebie. Należy stosować w takim wypadku nadążano korekcję, by móc zastosować m.in. długą ekspozycję do zdjęć.

Do w pełni prawidłowego ustawienia teleskopu trzeba ustawić oś biegunową.



Rysunek 3 Zobrazowanie znaczenia deklinacji i rektascensji

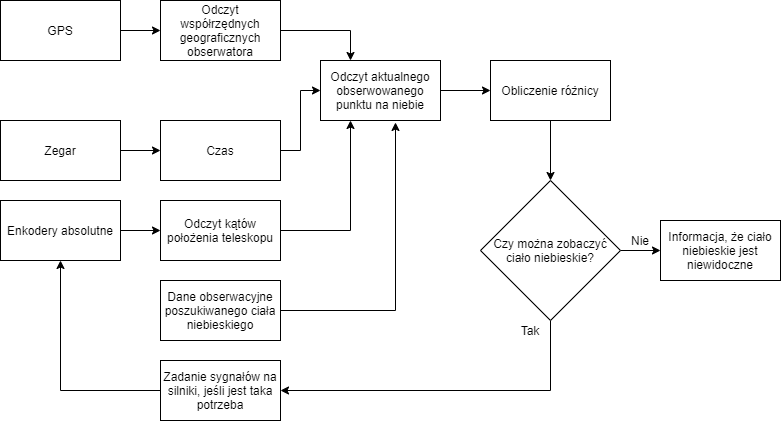
Oś biegunową teleskopu powinniśmy ustawić zgodnie z wynikiem sumy:

Lokalny czas gwiazdowy Greenwhich + długość geograficzna obserwatora - Rektascensja poszukiwanej gwiazdy = lokalny kąt godzinny.

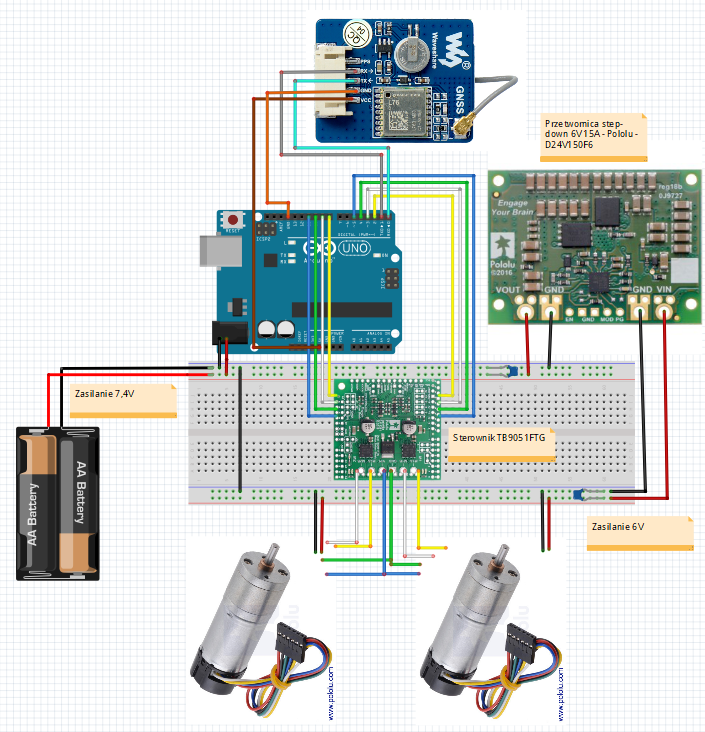
W praktyce błędne ustawienie osi rektascensji teleskopu wpływa jedynie na to, że obraz będzie obrócony o pewien kąt. Niemniej kompensując obrót ziemi względem własnej osi obraz w obiektywie będzie nieruchomy.

**Schemat blokowy algorytmu**

Uwaga, teleskop musi być ustawiony w osi Północ-Południe. Wynika to z tego, że użytkownik będzie wstanie znacznie precyzyjnej ustawić w tej osi niż program z absolutnymi enkoderami. Wynika to m.in. z błędów od deklinacji magnetycznych oraz że nie każdy teleskop ma ruchomą 4 oś.



**Zaprojektowany układ elektronicznych**



Użyte komponenty:

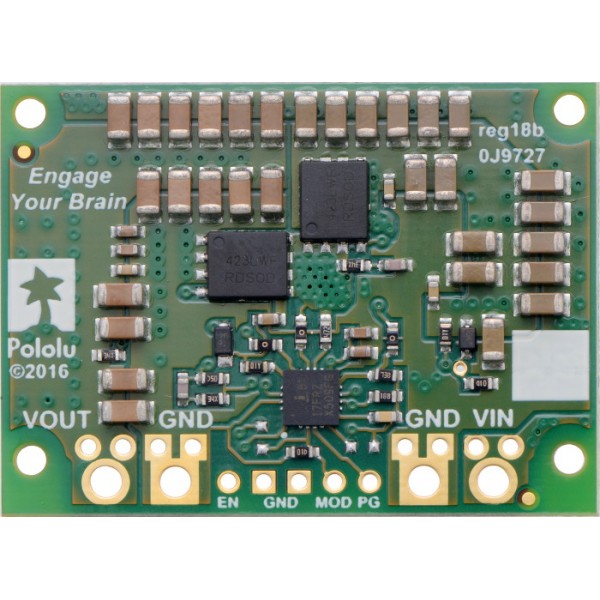
Arduino Uno Rev3

Został użyty ze względu na odpowiednią moc obliczeniową, wystarczającą ilość wyprowadzeń pinów cyfrowych, posiadanie interfejsów komunikacyjnych UART i USB. Może być zasilane prądem o napięciu 7-12V. Również ważne jest to, że moduł jest bardzo łatwo programować, oraz jest w posiadaniu członków zespołu.

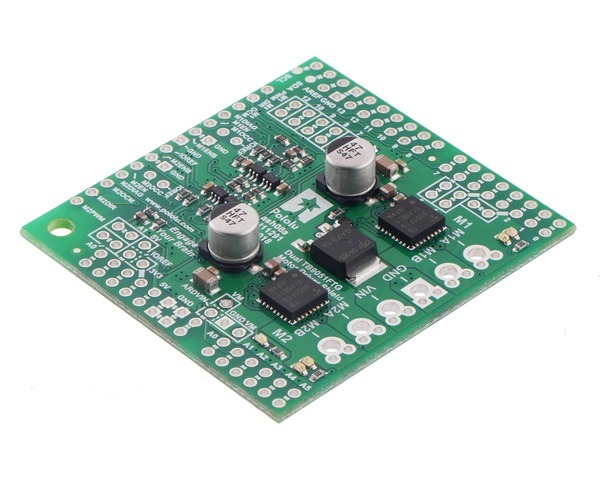
Zasilanie – akumulator Li-Ion LG INR18650 MG1 3,7V 2900mAh 10A

Ze względu na to, że teleskop będzie działał bez możliwości pobierania prądu z sieci, postanowiono użyć zasilania akumulatorowego. Parametrami branymi do wyboru zasilania całej elektroniki i mikrosilników były napięcie znamionowe, pojemność oraz prąd rozładowywania. Do prawidłowego funkcjonowania całej elektroniki napięcie musi być 6V dla silników oraz 7V-12V dla modułu Arduino. Akumulatory są podłączone szeregowo w celu zapewnienia docelowego napięcia 7,4V. Oprócz tego, prądy rozładowania zgodnie z obliczeniami będą wynosiły do 6A. Akumulatory to gwarantują z dużą nawiązką, co jest szczególnie ważne w momencie, gdy cały system jest uruchamiany i przez krótki czas są duże prądy rozładowywania. Również ważnym czynnikiem jest pojemność, która powinna być wystarczająca do zasilania całego systemu przez określony czas. Innym ważną przyczyną wyboru tych ogniw był rozmiar AA, co ułatwia dobór ładowarki przez użytkownika. Pewnym mankament jest to, że ten typ akumulatorów ma zmniejszoną pojemność w temperaturach poniżej 0 stopni Celsjusza.

Przetwornica step down – D24V150F6 6V 15A – Pololu 2882

Wybór tej przetwornicy wynika z posiadania odpowiednego napięcia wejściowego 7,2V do 40V zasilanego ogniwami, napięcia wyjściowego 6V umożliwiające uzyskanie odpowiednich napięć na silnikach, oraz wysoki prąd wyjściowy 15A, co bez problemowo zasili zastosowane silniki i pozwala na ewentualne zastosowanie bardziej prądożernych silników w przyszłych modyfikacjach.

Sterownik silnika krokowego – Dual TB9051FTG Motor Driver Shield

Silniki krokowe wymagają sterowników, by móc łatwiej nimi sterować za pomocą mikrokontrolera. Stosuje się zwykle dedykowany układ elektroniczny. Zastosowanie powyższego układu wynika z tego, że jest dedykowany do zastosowanych silników. Oprócz tego producent zapewnia odpowiednie wsparcie w postaci bibliotek na Arduino.

Moduł GPS – L76X Multi GNSS

 Powyższy moduł lokalizacyjny zastosowano ze względu na szybkie i dokładne pozycjonowanie (do 30s), małe zużycie energii, napięcie zasilania umożliwiające zasilanie bezpośrednio z pinu Arduino, posiadanie już gotowych bibliotek na Arduino oraz małe rozmiary (32x25 mm).

**Implementacja programu w wersji uproszczonej**

**Kosztorys**

|  |  |  |  |
| --- | --- | --- | --- |
| Nazwa | Sztuk | Cena za 1 szt. | Koszt |
| Arduino UNO | 1 | 92,00 zł | 92,00 zł |
| 499:1 Metal Gearmotor 25Dx73L mm LP 6V with 48 CPR Encodera firmy Pololu | 2 | 130,00 zł | 260,00 zł |
| Akmulator LG INR18650 MG1 | 2 | 7,00 zł | 14,00 zł |
| L76X GPS Module | 1 | 73,00 zł | 73,00 zł |
| Przetwornica step-down D24V150F6 | 1 | 170,00 zł | 170,00 zł |
| Sterownik TB9051FTG | 1 | 75,00 zł | 75,00 zł |
| Drobny osprzęt: kondensatory 220nF/kable itd. | - | 100,00 zł | 100,00 zł |
|  |  | Całość: | 784,00 zł |

Warto dodać, że komercyjne systemy kosztują tego typu kosztują zwykle od 5000$ dolarów w góre.

Dodać porównanie z komercyjnymi rozwiązaniami?

Moje notatki:

Algorytm nastawiania:

3 silniki w układzie german equatorial:

Do Deklinacji(raz dobrze ustawione):

Odczytujemy obecny kierunek w płaszczyźnie równika. Ustawiamy dla nowych warunków. np. 89o15’50,79” dla Polaris.

Do Rektascensji(nadążnie):

Względem południa Greenwich dodajemy różnice w czasie oraz poprawkę i mamy kąt godzinny dla naszych warunków lokalnych (a nie rektascensje dla Grennwich).

Nie ma to jakoś dużego wpływu, po za tym, że gwiazdy będą obracana w czasie rzeczywistym. Czyli wystarczy, by się obracała oś rektascensji 1obr/24h.

Do Szerokości:

Silnik, który ustawia szerokość zgodnie z naszym położeniem na półkuli. Warszawa to 52o13’56’’. + kąt pochylenia naszego teleskopu(np. wzgórze)

<https://teleskopy.pl/porady/images/ustawienie_teleskopu_na_biegun_2.jpg>

<https://teleskopy.pl/ustawienie_teleskopu_z_montazem_paralaktycznym.html>

<https://www.navipedia.pl/astro.html>

<http://gwiazdozbiory.eulersoft.com.pl/wspolrzedne.html>

<https://resources.basicmicro.com/pololu-encoder-wiring/>

<http://home.agh.edu.pl/~bartus/index.php?action=efekty&subaction=arduino&item=40>

Arduino: Sygnał GPS odczytuje, godzinę. KOMPAS albo prawidłowe ustawienie w pozycji NS.

Zasilanie: Silniki 2 precyzyjne momentowe (deklinacja, szerokość) i 1 serwo do rektascensji.

5V i 12/24V ? Do arduino: 10 diód i przesuwanie się o ułamek kąta

Wybór elektroniki i programowanie!

Jak dobiorę wszystkie elementy to podesłać do Filipa z kartami katalogowymi!

Lista elementów:

- Arduino UNO z modułem GPS (7-12V 0,9A)

- 3x(2?) 499:1 Metal Gearmotor 25Dx73L mm LP 6V with 48 CPR Encodera firmy Pololu (6V, 2,4A każdy) oraz sterowniki: <https://www.pololu.com/product/2520>

- przetwornica step-up <https://botland.com.pl/pl/przetwornice-step-up/596-pololu-boost-przetwornica-regulowana-4-25v-2a-pololu-799.html> z 6V do ok. 9V dla Arduino

- 3x(2?) sterownik ULN2003A

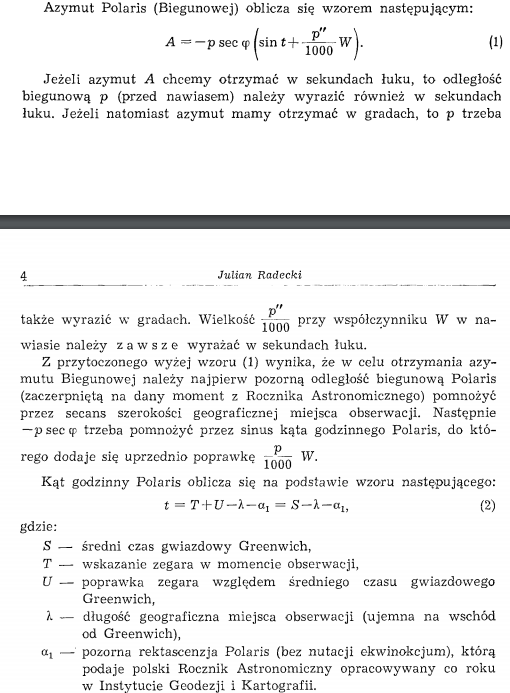
- baterie AA dające 6V (4x szeregowo) i 8A (4x równolegle; praca ciągła ), czyli 16 baterii + kondensatory. Wariant b: Można też użyć akumulatorów 18650, wtedy 7,4V (2xszeregowo) i 10A(1x równoległa; praca ciągła), czyli 2 ogniwa dla <https://www.batlit.pl/ogniwa_ICR18650MG1> , ale przy ich pojemności (2 850 mAh) to się rozładuje w ok. 30 minut pracy ciągłej. Przetwornica step-down <https://botland.com.pl/pl/przetwornice-step-down/7614-d24v150f6-przetwornica-step-down-6v-15a-pololu-2882.html> dla silników i Arduino bez.

- uchwyty do silnika(?) <https://www.pololu.com/product/4831>

- kompas <https://botland.com.pl/pl/akcelerometry/14543-cyfrowy-kompas-hmc1022-i2cuart-gy-26.html>

- GPS: <https://botland.com.pl/pl/moduly-gps/14643-modul-l76x-multi-gnss-gpsbdsqzss-waveshare-16332.html>

Dwa silniki!!!!



<http://212.180.216.235/dlibra/docmetadata?id=672&from=publication>