**Elektronika i programowanie**

**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 drogi po nocnym niebie. W tym rozdziale będą omówione podstawy teoretyczne do napisania algorytmów sterowania silnikami, zaprojektowany układ elektroniczny, schemat blokowy sterowania silnikami oraz zademonstrowanie działania programu w wersji uproszczonej.

**Podstawy teoretyczne do sterowania w poszczególnych osiach teleskopu:**

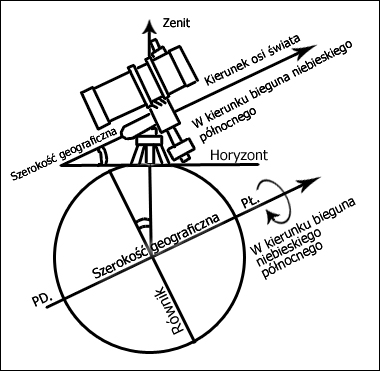
Uwagi do poniższego punktu: opis skupia się na prostym wytłumaczeniu zjawisk zachodzących podczas ruchu naszej planety bez zagłębiania się w szczegółowe zagadnienia i problemy jak różnica pomiędzy dobą słoneczną, a dobą gwiezdną albo nieliniowość czasu gwiezdnego. Takie podejście wynika z tego, że byłby 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.



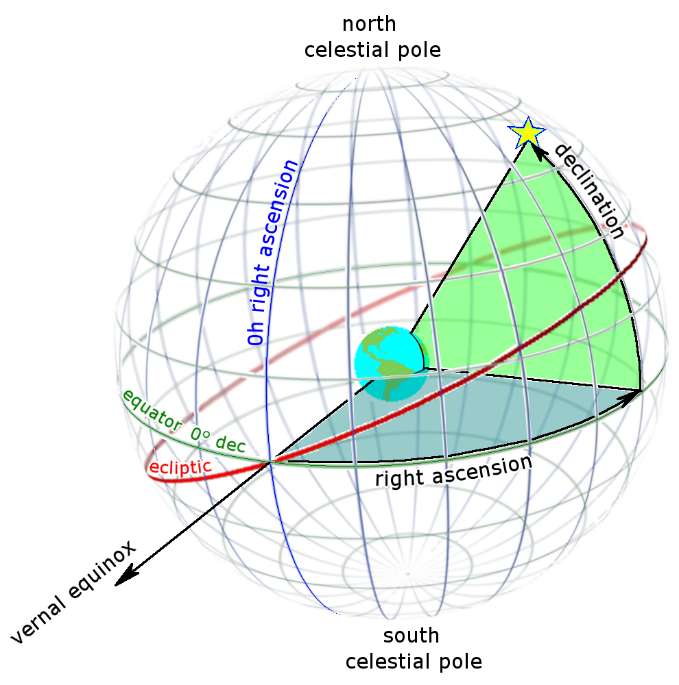
Rysunek 1 Osie w teleskopie o montażu paralaktycznym. Latitude - szerokość, Polar Axis - oś biegunowa; Declination Axis - oś deklinacji

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 wartość równą naszej obserwowanej gwieździe. Na rysunku 3 przedstawiono zobrazowanie znaczenia ustawienie kąta deklinacji. Dzięki ustawieniu tych dwóch omówionych parametrów nasz teleskop jest skierowany w wybrany przez nas punkt(gwiazdę). Nie mniej ze względu na ciągły obrót kuli ziemskiej wokół własnej osi obraz będzie się obracała wokół właśnie tego punktu (choć w tempie ok. 1obr/dzień). 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. Nie mniej kompensując obrót ziemi obraz w obiektywie będzie nieruchomy.

**Zaprojektowany układ elektronicznych**

**Schemat blokowy algorytmu**

**Implementacja programu w wersji uproszczonej**

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>

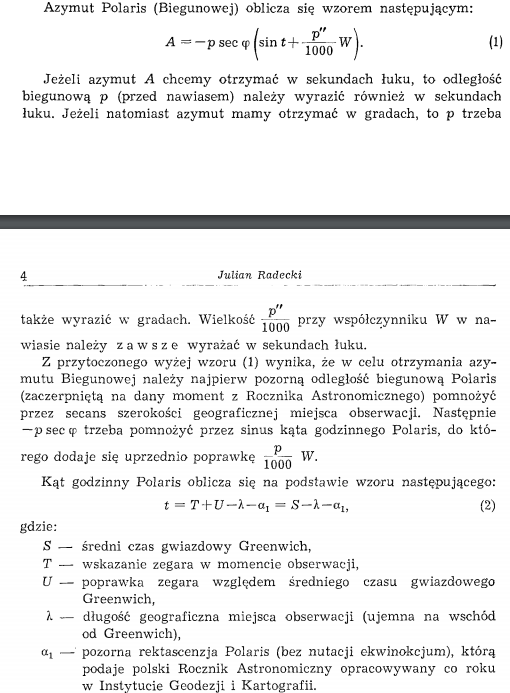
- przetwornica step-up <https://botland.com.pl/pl/przetwornice-step-up/596-pololu-boost-przetwornica-regulowana-4-25v-2a-pololu-799.html> z 6V do ok. 9V dla Arduino

- 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>



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