Sistem de auto-orientare pentru telescop

Autor: Vîrtan Răzvan-Nicolae

Grupa: 334CC

Introducere

Descriere

Proiectul propune realizarea unui sistem care să orienteze un telescop de masă după un anumit corp ceresc. Ne vom baza pe o bază de date stocată pe un telefon / laptop, din care vom putea selecta un anumit corp ceresc. Coordonatele corpului ceresc și ale telescopului vor fi transmise prin bluetooth la Arduino, care va face calculele bazat pe inputul de la un giroscop și va reorienta telescopul prin intermediul a 2 motoare pas cu pas.

Scop

Scopul este de a avea un sistem cât mai precis care să poziționeze un telescop pentru a privinde în fov-ul (field of view-ul) său un anumit corp ceresc. Anumite ajustări pot fi făcute ulterior și de factorul uman, însă trebuie minimizate.

Descriere generală

Utilizatorul va selecta un obiect ceresc (ex. stea, planetă) dintr-o listă pusă la dispoziție în interfața grafică a aplicației. Coordonatele acestui obiect ceresc vor fi apoi trimise la Arduino prin bluetooth, folosind modulul HC06.

Pe lângă coordonatele corpului după care vrem să ne orientăm, avem nevoie și de latitudinea și longitudinea la care se află telescopul, pe care le vom transmite tot prin bluetooth, profitând de faptul că putem afla facil coordonatele geografice într-o aplicație, folosind API-uri web.

Telescopul va fi prins într-o montură de tip Donbas, care permite rotația acestuia în plan orizontal și transversal, pentru o orientare cât mai simplă. Cu ajutorul unui giroscop, vom măsura unghiurile făcute de telescop față de o poziție de referință.

Corelând datele primite prin Bluetooth cu cele primite de la giroscop, Arduino va face calculele necesare pentru a vedea cum trebuie să corecteze poziția telescopului, după care îi va modifica poziția rotind telescopul în cele 2 planuri, prin intermediul a 2 motoare pas cu pas cu pas (folosim steppere pentru o mai bună precizie de poziționare).

Schema bloc

Hardware Design

Componente

- Arduino UNO
- Giroscop MPU6050 + sistem de prindere pe telescop
- Modul Bluetooth HC 06
- 2 motoare pas cu pas
- Fire de conexiune
- Drivere A4988
- BreadBoard / PerfBoard / PCB
- Telescop de Masă
- Sistem de prindere telescop (tablă / lemn, șuruburi etc.)

Partea Mecanică

Partea mecanică a proiectului urmărește simularea unei monturi Donbas, folosită de astronomii amatori pentru ușurința de poziționare a telescoapelor (doar două grade de libertate). Șuruburile clasice care permit telescopului să se rotească în jurul axei transversale și orizontale au fost înlocuite de axele a două motoare pas cu pas, care vor realiza orientarea automată. Prinderea a fost realizată pe un cornier suficient de mare pentru a permite prinderea telescopului.

Schema electrică

Pe lângă modulul bluetooth și giroscop, al căror scop a fost descris în introducere, am ales să nu controlez direct prin Arduino motoarele pas cu pas, ci să folosesc 2 drivere adiționale (câte unul pentru fiecare motor). Motoarele pas cu pas au un pas standard de 1.8 grade, precizie insuficientă pentru scopul proiectului. Folosind driver-ele A4988, am putut reduce de 16 ori pasul motoarelor, obținând astfel o precizie de 1.8 / 16 = aprox. 0.11 grade, deci de ordinul zecimii de grad hexazecimal.

Schema a fost realizată în eagle. Deoarece acesta e un program utilizat în design-ul de PCB-uri, nu am găsit biblioteci care să ofere simboluri pentru motoare pas cu pas, întrucât acestea nu aparțin în general PCB-ului. Din acest motiv, am lăsat capetele firelor care pleacă de la driver-ele de motoare pas cu pas libere și le-am etichetat corespunzător.

Software Design

Mediul de dezvoltare: Arduino IDE + VSCode în Linux pentru aplicația client în Python

Biblioteci și resurse 3rd-party

Pentru comunicarea cu giroscopul am utilizat biblioteca i2cdev. De asemenea, am avut nevoie de un cod inițial pentru a afla eroarea giroscopului și a face corecțiile necesare. Pentru o comunicare mai

ușoară cu modulul GPS și pentru integrarea automată a unghiurilor totale de deviația, am folosit în final biblioteca MPU6050_light

Pentru comunicarea cu modulul bluetooth am avut nevoie de bliblioteca SoftwareSerial, care este însă builtin și nu a trebuit instalată.

Algoritmul de reglare automată a poziției

Algoritmul pleacă de la presupunerea că telescopul este inițial orientat către un punct de referință (în acest caz, Steaua Polară). După ce datele obiectului ceresc, data și ora curentă și locația geografică au fost primite prin bluetooth, calculăm diferența dintre poziția noastră inițială și poziția finală la care trebuie să ajungă telescopul pentru a prinde în câmpul său vizual target-ul de pe bolta cerească.

Deplasarea telescopului are loc în 2 pași:

1. calculăm deviația în plan orizontal δ față de poziția dorită. Vom avea de făcut δ / 0.11 pași cu primul motor pentru a ajunge la acea poziție. La fiecare 15 pași, ne oprim și vedem cu ajutorul giroscopului cât ne-am deplasat față de cât ar fi trebuit să ne deplasăm, pentru a corecta eventualele erori și a actualiza numărul de pași pe care îl mai avem de făcut.
2. repetăm algoritmul de mai sus pentru deviația în plan transversal.

Funcții principale

- **readBluetoothData()**: citește un string de la programul client prin bluetooth, pâna la apariția caracterului `\n`
- **readFromUser()**: citește datele stelei, poziția geografică și timpul curent UTC de la utilizator, prin bluetooth
- **azalt()**: convertește coordonatele din sistemul celest în sistemul orizont (azimut și altitudine)
- **setup()**: inițializează pinii, setează modulul comunicarea cu modulul bluetooth și coordonatele stelei polare pentru referință
- **loop()**: citește datele unui corp ceresc de la utilizator și deplasează telescopul pentru a-l cuprinde în fov-ul său

Aplicația client

Interfața sistemului de auto orientare e reprezentată de o aplicație în linie de comandă scrisă în Python. Aceasta ofera un meniu din care utilizatorul își poate selecta corpul ceresc după care vrea să orienteze telescopul. După realizarea selecției, aplicația află data și ora curentă și locația geografică (folosind modulul `geocoder`) și le trimite către modulul HC06 (văzut pe Linux ca o intrare în `/dev`).

Mai multe detalii referitoare la implementarea codului sunt oferite în README-ul din repository-ul

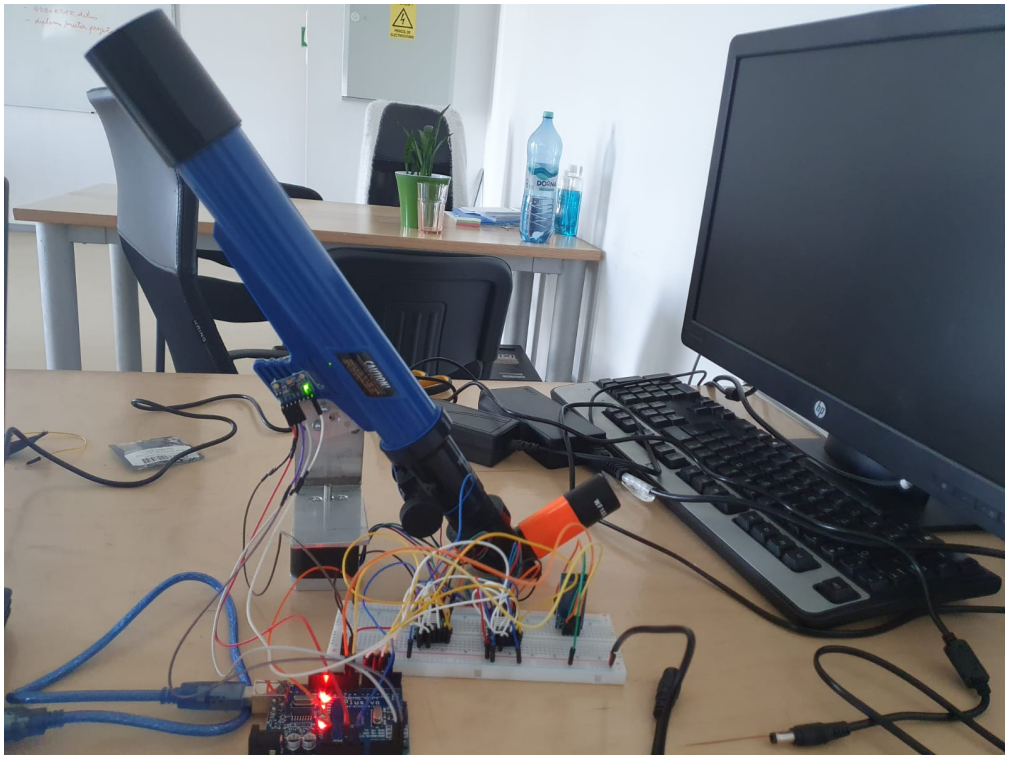
github al proiectului (vezi secțiunea Download).

Rezultate Obţinute

Utilizând componente nu foarte costisitoare, am reușit construirea unui sistem de orientare a telescopului la un preț rezonabil având în vedere precizia oferită. Am testat sistemul utilizând în principal aplicația Stellarium (pentru testare, oferă avantajul de a putea urmări corpuri cerești la orice oră, nu doar noaptea). Se poate observa că majoritatea corpurilor cerești din lista oferită în aplicația client sunt cuprinse după orientare în FOV-ul telescopului, sau se află la o deviație de câteva grade față de acesta.

De asemenea, sistemul se folosește pe cât posibil de laptop, pentru a evita utilizarea unor componente suplimentare, cum ar fi un modul GPS, și pentru a oferi o interfață cât mai utilă utilizatorului.

În imaginea de mai jos poate fi observat sistemul obținut (într-un stadiu în care partea de cable management nu era încă realizată).



Concluzii

Deși aparent simplu, Arduino oferă posibilitatea de conectare pentru o serie largă de module, ceea ce permite utilizarea sa nu doar în prototipuri demonstrative, ci și în proiecte în care precizia e un element esențial. De asemenea, acesta poate fi combinat cu computer-ul, pentru a oferi interfețe cât mai accesibile.

Download

Schema în eagle: schema_electrica_telescop.zip

Codul - publicat într-un repository github care conține și un fișer README cu mai multe detalii despre implementarea codului.

Jurnal

- **27.04.2022**: Alegere temă și validare cu asistentul
- **20.04.2022**: Documentare cu privire la partea matematică, sisteme de prindere
- **06.05.2022**: Achiziționare componente centrale
- **11.05.2022**: Creare pagină de wiki
- **20.05.2022**: Realizare parte mecanică
- **22.05.2022**: Realizare cod de bază pentru deplasarea telescopului și comunicarea cu giroscopul și modulul bluetooth
- **23.05.2022 - 26.05.2022**: Orientarea telescopului
- **27.05.2022**: Pagină de wiki

Bibliografie/Resurse

- “Practical Astronomy with your Calculator” 3rd edition, Petter Duffet-Smith
- https://www.instructables.com/Star-Track-Arduino-Powered-Star-Pointer-and-Tracke/
- https://www.instructables.com/Arduino-Star-Finder-for-Telescopes/
- https://lastminuteengineers.com/a4988-stepper-motor-driver-arduino-tutorial/
- https://lastminuteengineers.com/mpu6050-accel-gyro-arduino-tutorial/
- https://wired.chillibasket.com/2015/01/calibrating-mpu6050/
- https://create.arduino.cc/projecthub/RucksikaaR/interfacing-the-hc-06-bluetooth-module-with-arduino-f9c315
- http://www.convertalot.com/celestial_horizon_co-ordinates_calculator.html
- https://github.com/rfetick/MPU6050_light

Export to PDF