

Broloboc

Autor: [Adrian Mihalcea](#)

Introducere

Proiectul va folosi un modul cu giroscop conectat la un Arduino pentru a indeplini functia unui boloboc: verificarea inclinarii unei suprafete.

Descriere generală

Culegerea datelor despre inclinare va fi realizata folosind un modul cu giroscop MPU-6150. Acesta va fi legat prin I2C la Arduino, iar masuratorile acestuia vor fi afisate pe un 7-segment display (unghiul curent in grade, in modul).

Se va folosi unghiul de referinta (ce inseamna o suprafata plana) ca fiind cel de la pornirea Arduino-ului, corectand pentru erorile giroscopului prin functia `print_errors` descrisa la software design.

Voi folosi, de asemenea, un buzzer care va indica printr-un baz cand suprafata pe care este plasat bolobocul este orizontala. Voi considera o suprafata ca fiind orizontala daca este in intervalul (-2, 2) grade.



Hardware Design

Componente utilizate:

- Arduino UNO Plusivo
- 2 x breadboard: unul mic pentru display si buzzer si unul mare pentru giroscop
- Modul Accelerometru si Giroscop MPU6150
- Buzzer
- 7-segment 4 digit display

Schema electrica completa a proiectului a fost realizata in EAGLE. Singura non-conformitate cu realitatea este ca nu am gasit fix MPU6150 in niciun library asa ca am folosit un MPU6050 (o piesa identica in ceea ce priveste cablajul).

Am ales sa montez placuta cu giroscop pe propriul ei breadboard pentru a-i da cat mai multa libertate de miscare. Un urmator pas pentru acest proiect ar fi montarea tuturor componentelor in interiorul unei carcase, dar pana atunci am mers pe ideea unor cabluri cat mai lungi pentru giroscop si a unor miscari nu foarte bruste.

Software Design

Biblioteci folosite: Wire.h, SevSeg.h (linked mai jos)

Codul meu a fost dezvoltat folosind Arduino IDE. Integreaza Arduino-ul cu placuta cu giroscop, display-ul si buzzer-ul.

Setup: Comunicatia cu modulul MPU-6150 este realizata pe I2C (folosind biblioteca Wire). Se trezeste acest modulul printr-un 0 trimis la registrul sau 0x6B. Comunicatia cu 7-segment display se realizeaza folosind biblioteca SevSeg, in Setup setandu-se setarile acestuia (pinii pentru segmente si separatoare) si afisand 0000. Optional, se poate apela functia Print_error, descrisa mai jos.

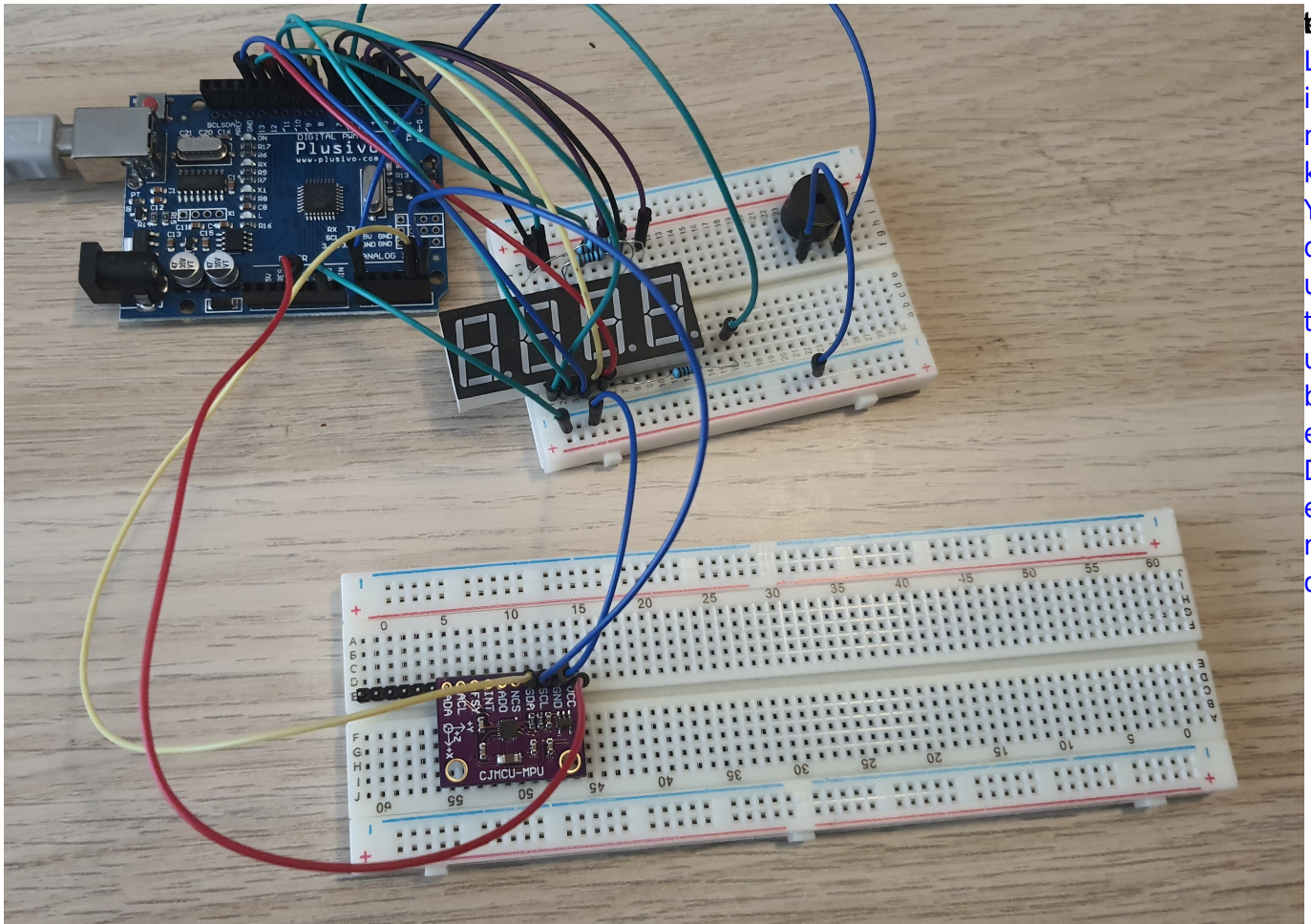
Print_error: Am descoperit ca giroscopul meu are niste erori vizibile din fabricatie, care il cauzau sa driftuiasca in timp cu ~ 0.5 grade/secunda. Pentru a lua in calcul aceste erori, am facut o functie de one-time setup care realizeaza 200 de masuratori, presupunand ca giroscopul este pe o suprafata plana, face media acestora si printeaza eroarea gasita astfel. Scazand aceasta eroare din fiecare masuratoare, obtin niste rezultate mult mai precise si stabile.

Loop: In loop, la fiecare iteratie improspatez display-ul si comunic prin I2C cu MPU-6150. Astfel ii cer 6 registre incepand cu adresa 0x43 (conform datasheetului_ si primesc cele 6 registre corespunzand masuratorilor pentru 3 axe, fiecare axa in cate 2 registre. Folosind masuratorile pentru axa oX calculez, conform datasheet-ului, diferenta de unghi fata de ultima masuratoare si updatez unghiul curent, tinand cont si de eroarea medie masurata cu print_error.

Acest rezultat este afisat pe display folosind functia setNumber din library-ul pentru control de 7-segment display. Daca intervalul de toleranta de (-2, +2) grade este respectat, scoate zgomot si buzzer-ul, controlat cu functiile Arduino Tone si noTone.

Rezultate Obținute

Proiectul isi indeplineste foarte bine sarcina si face niste masuratori destul de precise (un unghi de 90 de grade este afisat ca 88-92 de grade, in functie de cat de lina este miscarea). Singura mare dificultate pe care am intampinat-o a fost fix legata de faptul ca nu am lipit conexiunile la giroscop (ceea ce eu am crezut initial ca va fi mai usor), asa ca a trebuit sa am foarte multa grija cand miscam



Link YouTube Demo

Download

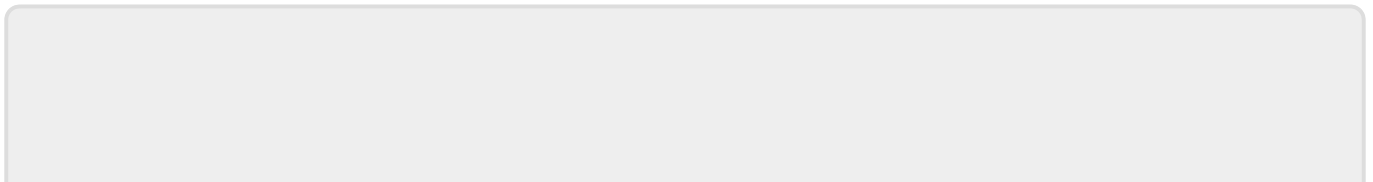
[broloboc.zip](#)

Bibliografie/Resurse

[Versiune PDF a acestei pagini](#)

[Datasheet MPU-6150](#)

[Library 7-segment display](#)



From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/cghenea/broloboc>



Last update: **2021/06/04 17:16**