

RoboDance

Introducere

RoboDance este un robot in miniatura ce executa mișcări repetate ale mâinii pentru a simula 'dansul', in timp ce cântă. Scopul acestuia este de a înveseli oameni ce petrec foarte mult timp la birou. Ideea de la care am pornit a fost o metoda ciudatica de debug: "Rubber Duck Debugging Method" in care explici unei rățuște de cauciuc codul pe care l-ai scris in speranța ca vei reuși sa repari programul; si daca se poate cu o rățușcă, de ce nu se poate si cu un roboțel? Fiind o posibilă sursa de amuzament, acest roboțel poate fi util in ridicarea moralului oamenilor ce au job-uri la birou, ducând totodata la creșterea productivității. }

Descriere generală

Servomotoarele vor fi conectate la modulul cu driver de motoare L298N, dar si la Arduino cu pinul de semnal. Modulul va fi, in consecință conectat la jack-ul DC, care este conectat la sursa de alimentare. De asemenea, atât senzorul tactil, buzzerul, cât si LED-urile vor fi conectate la Arduino. La primirea unui semnal de la senzorul tactil, microcontroller-ul va începe sa trimită semnale către buzzer pentru producerea de sunete, către LED-uri pentru a fi aprinse si oprite(simularea clipirii) cât si către cele doua servomotoare astfel încât mișcările mâinilor sa fie simulate cu succes. Dupa un moment de timp robotelul va opri dansul.

P.S.: Datorita faptului ca nu este necesara alimentarea de la o sursa externa alaturi de un driver de motor, s-a renuntat la urmatoarele piese: L298N, Jack DC, Sursa Alimentare.

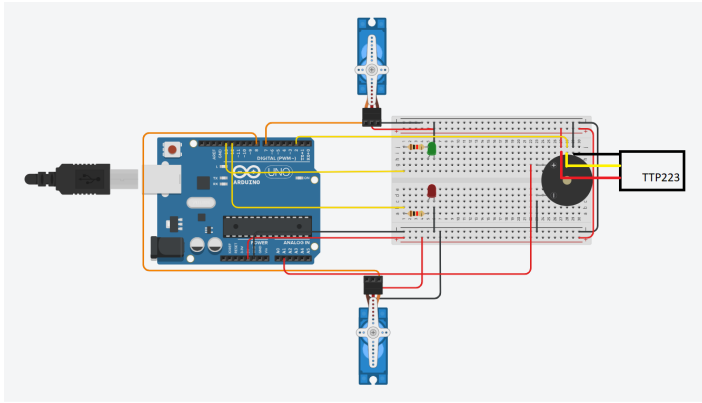


Hardware Design

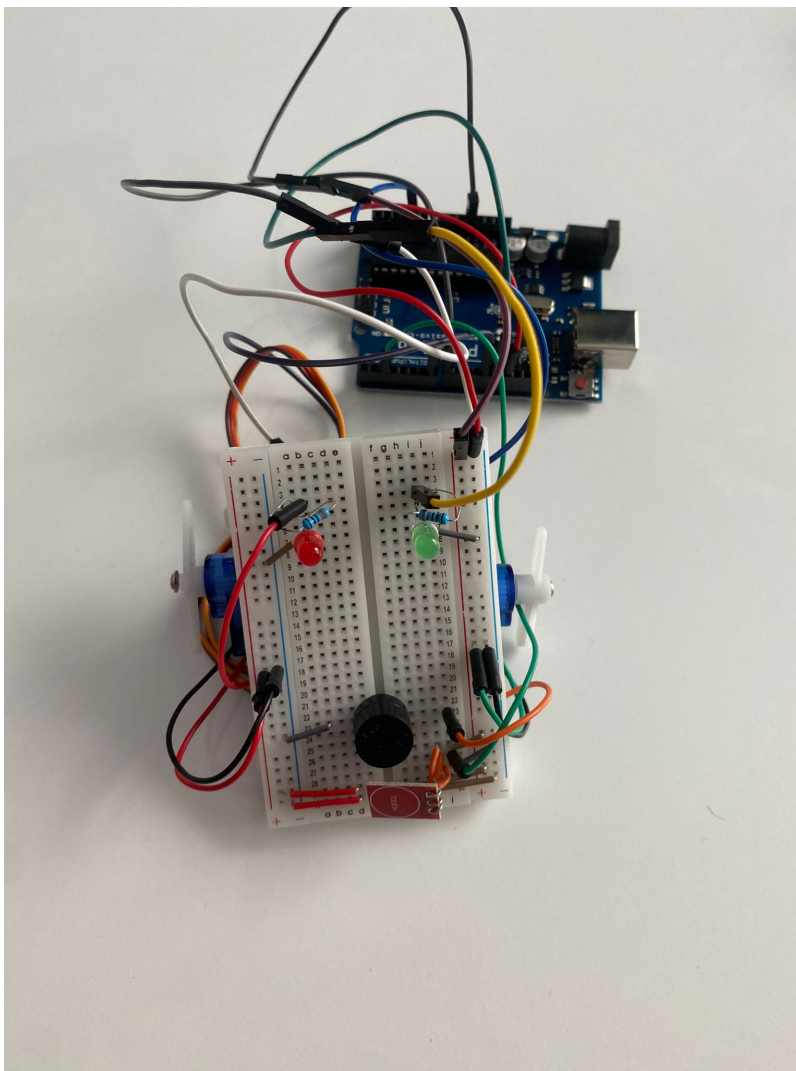
Lista piese:

- Arduino
- Modul driver L298N(x)
- 2 servomotoare SG190
- Breadboard
- LED(1 beculeț roșu, 1 beculeț verde)
- Jack DC(x)
- Sursa alimentare 5V(x)
- Senzor tactil TTP223
- Buzzer

Schema Electrica:



Implementarea schemei electrice:



Aici puneți tot ce ține de hardware design:

- listă de piese
- scheme electrice (se pot lua și de pe Internet și din datasheet-uri, e.g. <http://www.captain.at/electronic-atmega16-mmc-schematic.png>)
- diagrame de semnal
- rezultatele simulării

Software Design

Descrierea codului aplicației (firmware):

- mediu de dezvoltare (if any) (e.g. AVR Studio, CodeVisionAVR)
- librării și surse 3rd-party (e.g. Procyon AVRlib)
- algoritmi și structuri pe care plănuți să le implementați
- (etapa 3) surse și funcții implementate

Mediul de dezvoltare folosit pentru realizarea proiectului a fost Arduino IDE, în timp ce biblioteca externă folosită este Servo.h, utilizată pentru a putea controla servomotoarele SG90. Ca parte a codului propriu-zis, s-a realizat implementarea a 3 funcții: void setup(), void loop(), void do_dance(). Funcția do_dance() are ca scop setarea unei variabile pe 1 astfel încât funcția loop() să execute, într-un for, stingerea și aprinderea becurilor LED, pornirea buzzer-ului cu o anumită valoare și mișcarea servomotoarelor. Pe de altă parte, funcția setup() asigură atasarea pinilor 7 și 8 la obiectele de tip Servo ce au ca scop să poată porni servomotoarele, împreună cu setarea pinilor 12, 13 și A1 drept output(LED-urile + Buzzer-ul) și setarea pinului 2 drept input. Pe lângă aceste inițializări, funcția setup() asigură funcționalitatea unei întreruperi în momentul în care senzorul touch detectează o apăsare(do_dance()) este funcția de întrerupere).

Codul sursă este:

[stan_petrisor_334ca_source_code_final.txt](#)

Libraria externă folosită este:


[servo-master-petrisor.zip](#)

Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

Concluzii

Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună .

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume_student** (dacă este cazul). **Exemplu:** Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru_alin**.

Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2024/rpascale/stan.petrisor> 

Last update: **2024/05/20 19:47**