

Nerf Sentry Turret

Nume: Prioteasa Cristi Andrei

Grupa: 332CC

Introducere

Ce face

Proiectul este o tureta sentry care trage cu proiectile din burete folosite pentru armele NERF. Tureta are 2 grade de libertate (control pe axele OX, OY) si o magazie de 5-6 proiectile. Poate fi controlata prin joystick-ul atasat sau poate fi lasa in modul sentry in care va scana zona cu ajutorul unui senzor infrarosu si va incerca sa detecteze miscari si sa traga in directia miscarii.

Scopul proiectului

Scopul e de a avea o tureta sentry functionala care poate sa traga si la inclinatie (si pe axa OY) si un mod sentry in care va detecta miscarea si va trage in directia de variatie maxima a distantei masurate de senzorul infrarosu.

Ideea de la care am pornit

Initial pornisem de la o tureta dintr-un joc (tf2) si am ajuns la varianta asta pentru ca am reusit sa gasesc si cateva referinte online si posibil (ramane de vazut) sa nu am nevoie de a printa piese cu un 3d prinder pentru precizie ft mare. <https://www.youtube.com/watch?v=3Ma5ZCZVQRs>

De ce este util?

In afara de scopul didactic si de a combina in ceva mai multe domenii cum ar fi mecanica , electronica, programare si semnale , tureta reprezinta un model clasic care e folosit pentru pitchere de baseball sau hockey sau pentru paint ball sau airsoft.

Descriere generală

Turreta are cateva mecanisme cheie:

1. Sistemul de control al miscarilor

<<<Un ax trecut printr-un rulment si lipit la un capat de un servomotor care poate merge pana la 180 de grade va controla miscarea pe o axa.La capatul axului, pe o platforma va fi un al doilea ax perpendicular pe primul controlat de un alt servomotor care va controla miscarea pe cealalta axa. Pe acest al doilea ax se va afla mecanismul de tragere, senzorul infrarosu si rezerva de proiectile.>>>

Paragraful de mai sus reprezinta o abordare eleganta a proiectului, dar cum rulmentii nu s-au potrivit iar axul de metal s-a rupt, robotul va fi facut integral din carton lipit cu mult scoci si nu va mai avea 2 grade de libertate, ci doar unul.

2. Sistemul de tragere

Doua motoare DC de viteza mare si conectate la o baterie de 9V printr-un tranzistor si care primesc comanda ON/OFF de la arduino vor roti doua roti de cauciuc. Proiectilele vor cadea impinse de un elastic (pentru a facilita tragerea si cand turreta se afla sub un unghi) si vor fi impinse de un servomotor legat la o axa (pentru a translata miscarea de tip arc in miscare liniara). Cand proiectilul va ajunge intre rotile motoarele, datorita frecarii vor fi lansate cu o viteza destul de mare. Problema in aceste mecanism este positionarea foarte exacta a pieselor pe care sper sa reusesc sa o duc la capat, tinand cont ca folosesc carton.

3. Sistemul de depozitare de proiectile

Pentru ca nu am putut sa ating precizia de 1-2 mm folosind carton, nu mai exista si va putea trage doar cu un singur proiectil.

4. Modul sentry-mode

Cand tureta se afla in acest mod, se va misca pe axa orizontala si va masura distanta pana la obiecte creand initial o harta a imprejurimilor imediate. Apoi orice miscare reprezinta o variate (nu foarte rapida pt ca senzorul nu are rezolutie foarte mare) a distantei catre background si va fi detectata. Tureta va lansa apoi proiectile in acea directie, eventual cu un delay datorat rezolutiei senzorului ultrasonic.

Alimentez motoarele de la o baterie de 9V folosind un divizor de tensiune facut din doua rezistoare. Folosesc un tranzistor ca switch pentru a porni si a stinge motoarele la comanda. Motoarele pot fi stinse dupa prima tragere deoarece robotul nu are un incarcator. Motoarele trebuie sa stea pornite pe durata de viata a programului pentru ca accelareti de pornire e prea mica si nu ajung suficient de repede la o viteza de tragere cand se detecteaza o miscare.

Hardware Design



Lista de piese

- 2 x Motor 130-13200
- 1 x Arduino ATMEGA328p
- 1 x Baterie 9V
- 1 x Senzor ultrasonic HC-SR04
- 1 x Breadboard
- 1 x npn transistor
- 2 x Servomotor
- 2 x 1k resistor for voltage divider
- Carton pentru cadru si schelet
- LEDs
- 1 x Buzzer
- 40 x Fire
- Altele

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

Medii de dezvoltare si biblioteci

- Arduino IDE

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 plănuiți să le implementați
- (etapa 3) surse și funcții implementate


Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

Concluzii

Proiectul a avut nevoie de multe componente mecanice sau suporturi 3d-printate pe care eu am decis să le construiesc din carton, pierzând aici săptămâni la rând și, în final, rezultatul nu e tocmai unul dorit, dar funcționează așa cum doream. Am realizat după ce am comandat mare parte din piese ca ar fi trebuit să folosesc un stepper și un driver pentru rotație în loc de un simplu servo, dar proiectul funcționează și așa. Detectarea mișcării este controlată direct de precizia senzorului ultrasonic și funcționează bine la peste 1m. Partea electronică nu a fost extrem de complicată, probleme aparând la motoarele DC și servo care trageau mult curent din cauza frecării cu alte bucăți din robot. Din punct de vedere software se mai pot face îmbunătățiri, de exemplu folosind întreruperi pentru a verifica dacă s-a detectat o mișcare și o altă întrerupere pentru tragere, dar s-a dovedit a fi destul de complicat.

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**.

<https://github.com/PrioteasaAndrei/pm-2022-sentry-turret/tree/main>

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**.

<https://gist.github.com/WoLpH/0094b278a54f8f472c76b7eb0d0a8f89>

<https://arduinogetstarted.com/tutorials/arduino-motion-sensor-servo-motor>

<https://www.youtube.com/watch?v=47phOJFCXL0>

<https://somanyletech.com/9v-to-5v-converter-circuit/>

<https://create.arduino.cc/projecthub/onedeadmatch/control-dc-motor-with-npn-transistor-arduino-pwm-cdaf2e>

[Export to PDF](#)

From:

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

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2022/ndrogeanu/sentry_turret



Last update: **2022/06/02 10:38**