

# Zelda Guardian

**Nume:** Mihnea Tudor

**Grupa:** 331CAb

## Introducere

Proiectul este inspirat dintr-un joc numit The Legend of Zelda: Breath of the Wild, si **incearca** sa imite un Guardian stationar. Proiectul in sine este practic un robot alcatuit din doua parti, corpul si capul.

Cea mai complexa componenta este capul, care se roteste orizontal, in jurul axei Oy, si urmareste orice obiect suficient de apropiat, cu ajutorul unor senzori de distanta. Dupa un anumit timp in care obiectul ramane in raza de actiune a robotului, acesta va arunca o bila in directia generala a tintei. Din motive de siguranta, s-a renuntat la ideea care includea un laser.

Resurse video:

- Un videoclip care demonstreaza cum functioneaza robotul in joc, si care a fost sursa principala de inspiratie, se poate gasi [aici](#).
- Videoclipul care mi-a dat speranta ca proiectul este fezabil este facut de cei de la Adafruit, si se poate gasi [aici](#).

## Descriere generală

La nivel logic, robotul functioneaza pe baza unui automat finit de stari. Starile robotului sunt: Idle, Wake Up, Follow, Attack, Survey, Rambo. Fiecare este explicata in sectiunea de Software Design, alaturi de o reprezentare grafica a automatului.

La nivel fizic, robotul este alcatuit din 2 parti: corpul si capul. Corpul contine un singur Motor servo 360 care misca orizontal intreg capul. Elicea motorului este prinsa cu suruburi de un suport solid, motorul fiind cel care se misca in jurul elicei. De acel motor este prins un alt suport, o placa mare, dreptunghiulara, ce reprezinta capul, pe care sunt puse toate celelalte piese.

Capul are mai multe functionalitati. Prima este cea de detectie si urmarire a obiectelor apropiate. Doi senzori ultrasonici HC-SR04 si unul Sharp vor fi amplasati in jurul "ochiului", astfel:

- cel analogic deasupra, pentru a detecta distanta obiectului aflat direct in fata
- unul digital la stanga si unul la dreapta, pentru a detecta eventualele miscari ale obiectului

Partea complicata este lansatorul de bile. Acesta este amplasat pe placa dreptunghiulara care reprezinta capul, si trage cu bile printr-o gaura dintr-un perete facut tot din acelasi material. Acesta este format din 2 motoare DC apropiate, care invart niste roti. Bilele care ating cele doua roti, aflate intr-o rotatie continua, vor fi lansate in fata. Bilele sunt tinute intr-un tub din plastic, prin care vor circula cand se lanseaza cateva din fata. Cand ajung la roti, ele vor fi aruncate. Pentru a controla fluxul bilelor, am folosit un servomotor care nu e continuu, care "deschide" si "inchide" tubul, pentru a permite urmatoarei bile sa fie lansata.

O schema bloc initiala se poate gasi in figura urmatoare:

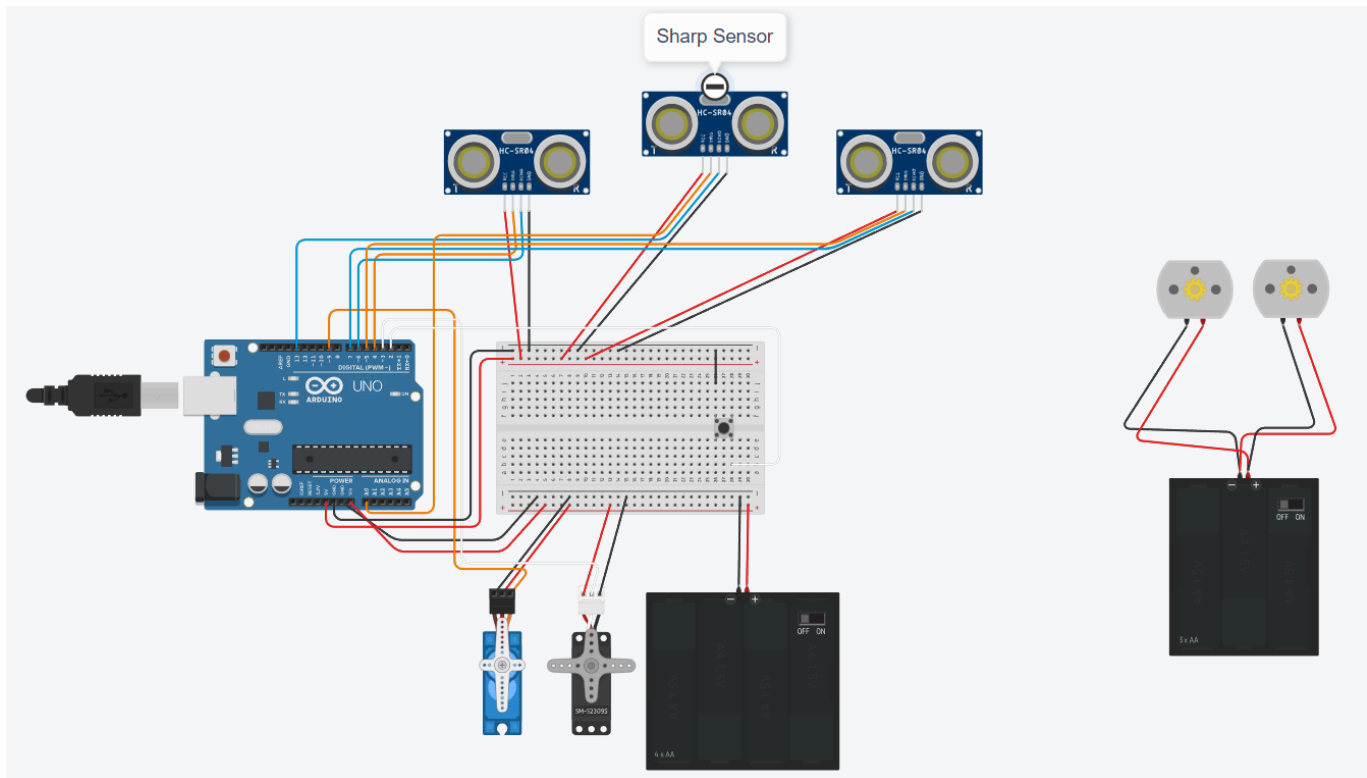


## Hardware Design

Lista componente:

- 1x Arduino Uno R3 ATmega328P
- 1x Breadboard 830 puncte MB102
- 1x Motor servo 360 12kg
- 1x Motor Servo MG90S 180g
- 2x Motor DC R140
- 2x Senzor ultrasonic HC-SR04
- 1x Senzor Sharp GP2Y0A60SZLF
- 1x Buton
- 7x Baterii AA
- Multe Fire Jumper

Schema hardware, realizata in Tinkercad:



## Software Design

Arduino.

Pe partea de software, robotul functioneaza pe baza unui automat finit de stari. Starile sunt reprezentate grafic in figura de mai jos, si explicate in continuare:

- **Idle** - robotul nu are niciun obiect in fata sau in jur, asa ca "doarme"
- **Waking Up** - robotul a detectat un obiect in fata, asa ca se "trezeste"; daca obiectul inca este in fata dupa 2 secunde, incepe sa atace
- **Follow** - robotul urmareste obiectul stanga / dreapta pentru o perioada determinata de timp, si poate ataca daca obiectul inca e in raza de actiune, sau poate intra in starea de Survey
- **Attack** - robotul arunca o bila in directia in care a ramas indreptat, ia o pauza, si revine in modul Follow
- **Survey** - robotul si-a pierdut tinta, asa ca face un ultim efort de a o gasi, cautand alte tinte in jur (se uita in stanga sau dreapta, in functie de sensul ultimei cautari), pentru un timp determinat, dupa care se "culca" la loc (intra in Idle); daca gaseste, reintra in Follow
- **Rambo** - indiferent de starea anterioara, la apasarea butonului, robotul intra in starea Rambo pentru 5 secunde, in care arunca toate bilele ramase; la final, se intoarce in starea Idle (fiind obosit, trebuie sa se odihneasca)



## Rezultate Obținute

Mai jos este un mic demo care arata diversele stari prin care trece robotul, dar si cum functioneaza

sistemul de tragere.

## Concluzii

As fi vrut sa adaug multe lucruri la proiect: LEDuri, buzzer, mai multi pereti, culori, o parte software mai stabila etc, dar mai greu cu timpul semestrul asta. Desi nu arata exact ca materialul de referinta, mi-a placut foarte mult sa lucrez la acest robot, e ceva complet diferit de oricare alta tema de pana acum.

Cu dedicatie tuturor fanilor Zelda ♥.

## Download

Codul proiectului se poate gasi aici: [zelda\\_guardian.zip](#).

## Jurnal

*1 iunie 2023* - adaugare buton + documentatie (intarziere datorata unor probleme tehnice neprevazute)

*29-30 mai 2023* - terminat restul :)

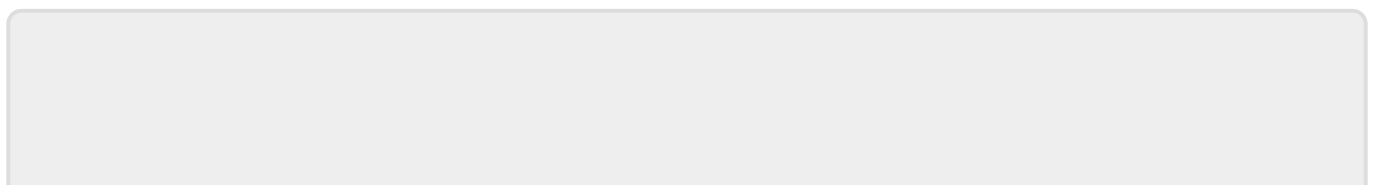
*05 mai 2023* - terminat documentatie v1.0

*04 mai 2023* - creare pagina wiki

## Bibliografie/Resurse

- [Adafruit Zelda Guardian](#)
- [Ping Pong Ball Launcher](#)
- [Tennis Ball Launcher](#)
- [Sharp Sensor Tutorial](#)
- [Ultrasonic Sensor Tutorial](#)

[Export to PDF](#)



From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2023/gpatru/zelda\\_guardian](http://ocw.cs.pub.ro/courses/pm/prj2023/gpatru/zelda_guardian)



Last update: **2023/06/01 16:03**