

# Useless Box

**Autor:** [VASILESCU Alexandru-Madalin](#)

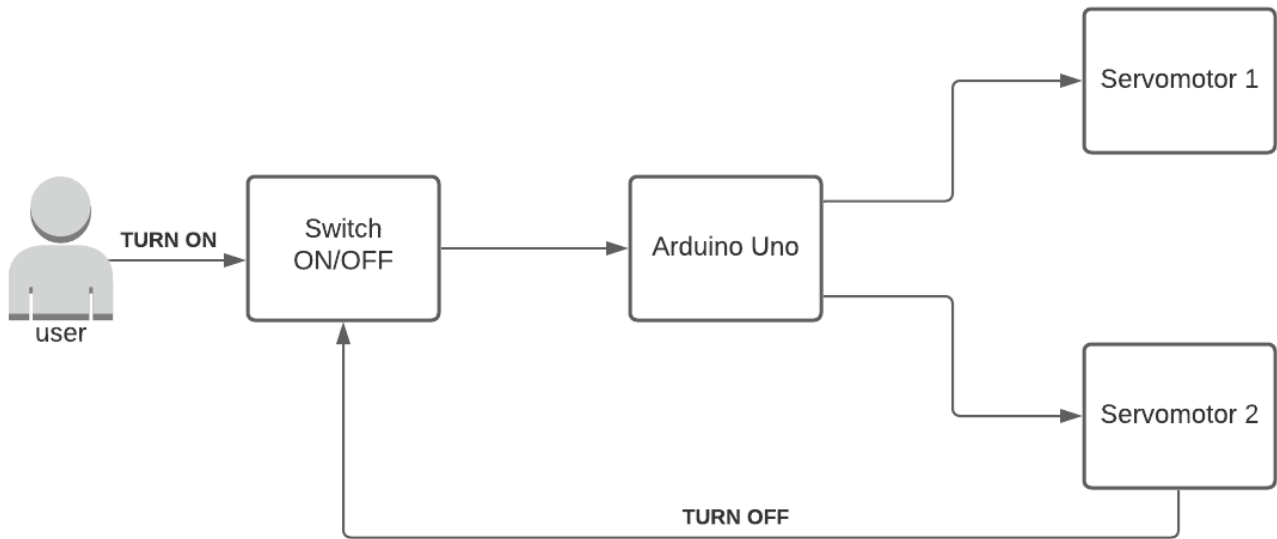
## Introducere

Proiectul reprezinta o implementare simpla a unui Useless Box, o jucarie simpla unde la comutarea unui switch un mecanism este declansat si un brat iese din cutie pentru a comuta din nou switchul. Acesta sunt comercializate si au devenit populare in ultima vreme. Multe variante ale acestora sunt disponibile, unde dispozitivele fac alte lucruri pe langa apasarea butonului sau sunt infrumusetate pentru a semana mai mult cu o jucarie. Eu am ales sa implementez o varianta de baza.

## Descriere generala

Dispozitivul este format dintr-un switch cu 2 randuri si 3 pozitii si 2 servomotoare. La apasarea butonului este declasat dispozitivul care actioneaza servomotoarele pe rand. Prima data este actionat un servomotor care are rolul de a ridica capacul cutiei. Apoi este actionat un alt servomotor care are rolul de a comuta switchul in pozitia off. Cele 2 servomotoare au atasate un brat din aluminiu pentru avea suficienta putere sa le miste. Servomotoarele sunt mici(folosite mai mult la jucarii) si nu au putere sa miste lucruri prea grele. Dupa ce servomotarele isi termina rotatia stabilita prin program se intorc la starea initiala pana cand butonul este din nou apasat.

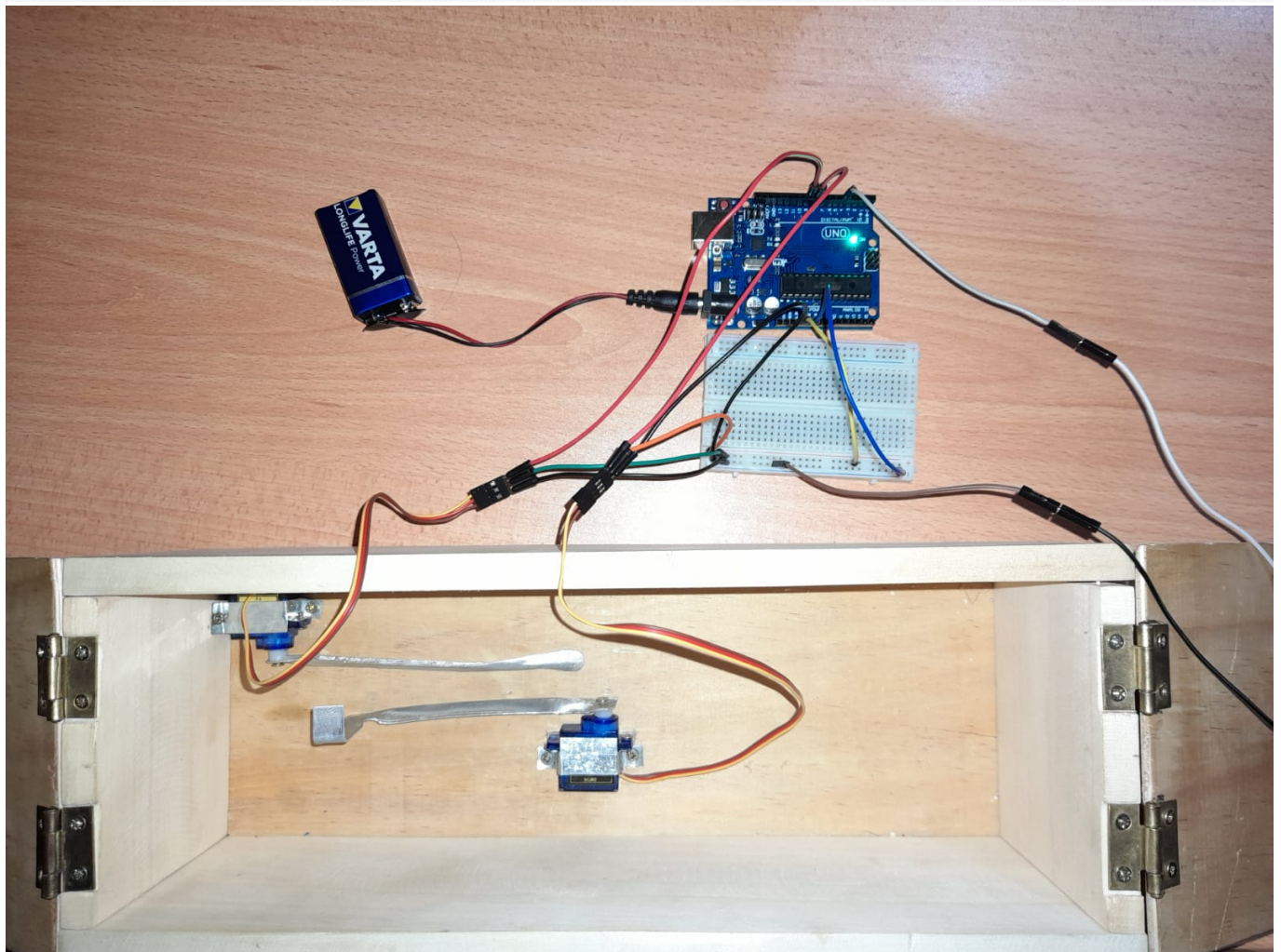
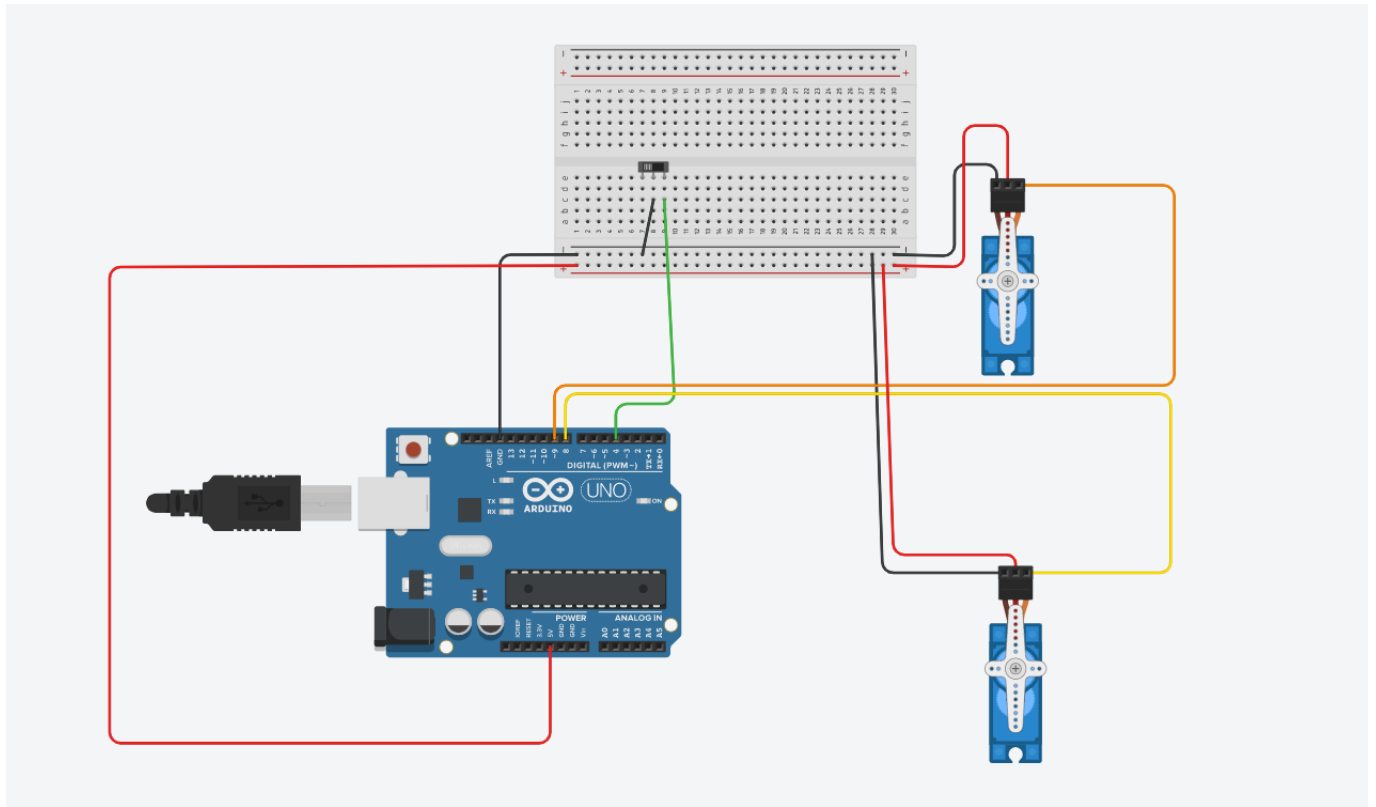
## Schema bloc



## Componente

- Arduino UNO R3
- Switch - 2 row 3 pos
- Servomotor x2
- Fire
- Baterie 9V
- Adaptor baterie
- Breadboard

## Hardware Design



## Software Design

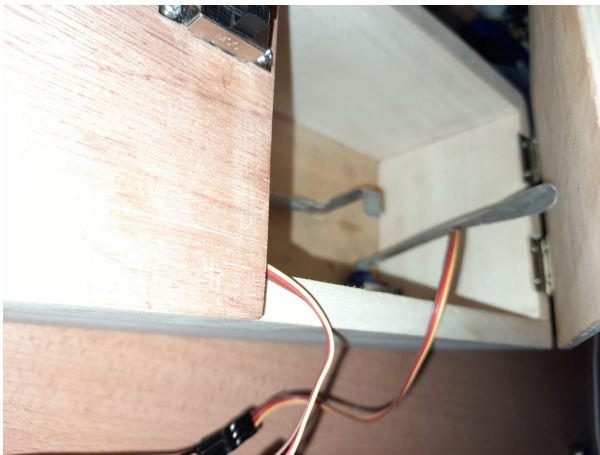
- Mediu de dezvoltare: Arduino IDE
- Biblioteci folosite: Servo, Stdlib(pentru generarea numarului random)

In implementare am folosit o intrerupere externa pentru detectarea comutarii butonului si functii de librarie pentru actionarea servomotoarelor.

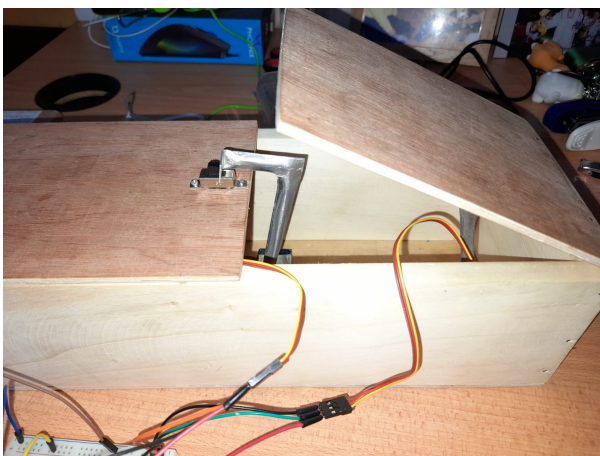
Am inceput prin a initializa servomotoarele. Le-am atasat pinii 8 si respectiv 9 si le-am setat la 0 la inceputul programului pentru a se reseta in orice stare ar fi ramas pe parcurs. Am setat pinul 4 ca input pentru a detecta schimbarile butonului. Am activat intreruperile externe si le-am activat doar pentru pinul 4.

In functia de tratare a intreruperilor doar modific o variabila globala, functiile mecanismului sunt apelate in loop(). Am ales doar modificare variabilei globale in tratarea intreruperii deoarece mecanismul are o durata de functionare foarte mare si ar fi intrerupt foarte mult decurgerea programului daca pastram totul in intrerupere.

Am creat mai multe functii, toate asemanatoare intre ele, pentru miscarile mecanismului. Prima data se executa rotatia servomotorului 1 care are rolul ridicarii capacului.



Apoi se executa rotatia servomotorului 2 pentru actionarea butonului.



Apoi in ordine invers se revine la starea initiala. Am folosit delayuri diferite intre functii pentru a modifica viteza servomotoarelor sau durata de asteptare intre executii(intre cele 2 servomotoare).

In loop folosesc un switch pentru a detecta care functie sa fie apelata si un index i care este ales random intr-un interval pentru a face jucaria putin mai nedeterminista si amuzanta.

Ca functii utilizate:

- servo.attach pentru a atasa un pin la servomotor
- servo.write pentru a seta unghiul servomotorului
- delay
- rand
- sei si cli pentru setarea intreruperilor

## Rezultate obtinute

- Demo: <https://www.youtube.com/watch?v=axESkWFMhhl>

Am obtinut o copie functionala dupa o jucarie populara in varianta ei de baza. In demo se poate observa cum functioneaza si cum este comutat butonul intai de mine si apoi comutat din nou de catre jucarie. Se pot observa 4 viteze si actionari diferite.

## Concluzii

Am reusit sa inteleg notiunile prezentate la laborator si sa le aplic. A fost foarte interesant, desi as fi preferat sa aleg ceva putin mai complex proiectul meu fiind destul de simplu de realizat. Am intampinat si unele dificultati datorate puterii servomotoarelor si rezistentei butonului initial, dar am reusit sa inlocuiesc componente si sa trec peste probleme.

## Download

[Arhiva proiect](#)

## Jurnal

- Realizare pagina
- Achizitionare componente
- Documentare
- Dezvoltare schema in Tinkercat
- Creare periferice(cutie, lamele de aluminiu)
- Asamblare componente
- Programare placuta
- Scriere documentatie

- Finalizare pagina wiki

## Bibliografie/Resurse

- <https://ocw.cs.pub.ro/courses/pm/lab/lab3-2021>
- [https://en.wikipedia.org/wiki/Servo\\_\(radio\\_control\)](https://en.wikipedia.org/wiki/Servo_(radio_control))
- <https://forum.arduino.cc/t/help-with-code-and-wiring-involving-3-position-switch-dc-motor/370709>
- <https://forum.arduino.cc/t/how-to-wire-these-6-terminal-toggle-switches-dpdt/527880>
- <https://www.arduino.cc/reference/en/libraries/servo/>

From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2021/avaduva/useless\\_box](http://ocw.cs.pub.ro/courses/pm/prj2021/avaduva/useless_box)

Last update: **2021/06/03 20:33**

