

# Tomogachi

## Introducere

Proiectul implementează un joc portabil ce simulează un animal de companie ce va avea diverse nevoi precum:

- hrană
- apă
- somn
- joacă

Ele au asociate câte un procent și scad cu trecerea timpului. Dacă oricare dintre ele ajunge la 0, animalul va muri. Scopul este de a ține animalul în viață cât mai mult timp.

## Descriere generală

Pentru reumplerea procentelor, jucătorul trebuie să cicleze printr-un meniu stânga-dreapta și să aleagă acțiunea dorită. Pentru joaca cu animalul, va juca un minigame. Pentru procentul de somn, jocul în sine trebuie să se afle în întuneric.



## Hardware Design

Pentru a realiza acest proiect vom avea nevoie de:

Componente	Numar	Descriere generală	Link piesa
Placa de dezvoltare Arduino Uno	1	Placa de dezvoltare Arduino Uno R3 ATmega328P.	<a href="#">Arduino Uno R3 ATmega328P</a>
Senzor lumina ambientala TEMT6000	1	Tensiune de functionare: 3.3-5V Sensibilitate maxima: 570nm Unghi sensibilitate: 60grade Dimensiuni 14x8mm.	<a href="#">TEMT6000</a>

KMR-1.8 SPI	1	KMR-1.8 SPI este un modul LCD TFT de 1,8 ", cu rezoluția de 128 x 160 pixeli și 262k de culori. Interfața de afișare este în serie, deci este nevoie doar de 5 fire pentru control. Controlorul acestui modul LCD este ST7735. Dacă folosiți o placă Arduino, atunci ar trebui ca fiecare port IO să fie conectat cu o rezistență de 500 ~ 2KΩ. Ecranul suportă atât nivel de operare de 5V, cât și de 3.3V. Pentru alimentarea cu energie electrică, puteți utiliza alimentarea de 3.3V prin PIN-ul "3.3V" sau puterea 5-9V prin PIN-ul "VIN".	ST7735
Buzzer pasiv	1	Modulul constă într-un buzzer piezoelectric pasiv, care poate genera tonuri între 1,5 și 2,5 kHz prin comutarea și oprirea la frecvențe diferite, fie folosind întârzieri sau PWM.	Buzzer

Alte piese relevante:

- 8 rezistente pentru divizori de tensiuni
- 1 rezistența pentru buzzer
- 2 rezistente pentru ecran
- 3 butoane
- Baterie de 9V cu conector
- Buzzer pasiv
- Breadboard
- Pistol de lipit (honorable mention)

**Schema electrica:**

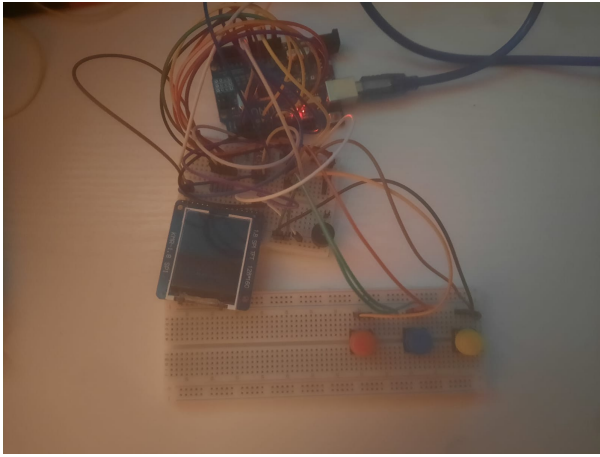


## Software Design

Am folosit bibliotecile ButtonDebounce pentru preluarea ușoară a semnalelor de la butoane și ucglib pentru controlarea ecranului.

[Repository Git Hub cu cod sursă](#)

## Montaj și demo



### [Video demo](#)

Calitatea videoclipului este proastă deoarece am camera stricată la telefon.

## Dificultăți

Biblioteca UCGLIB creează mult flickering pe ecran, poate fi rezolvat prin schimbarea la biblioteca ADAFRUIT.

Buzzer-ul este prea încet, poate fi rezolvat prin controlarea lui printr-un tranzistor sau amplificator.

## Download

Link pagina: <https://ocw.cs.pub.ro/courses/pm/prj2021/apredescu/tomodachi>

Link documentatie: [https://ocw.cs.pub.ro/courses/pm/prj2021/apredescu/tomodachi?do=export\\_pdf](https://ocw.cs.pub.ro/courses/pm/prj2021/apredescu/tomodachi?do=export_pdf)

## Jurnal

- Saptamana 1: alegerea proiectului
- Saptamana 2: cumparare piese necesare
- Saptamana 3: verificarea pieselor si montarea acestora pe breadboard
- Saptamana 4: cumpararea de piese suplimentare (fire tata-tata, un breadboard suplimentar si rezistente de 2.2k si 3.3 pentru diviziunile de tensiune, baterie de 9V si conector de baterie)
- Saptamana 5: scrierea si depanarea codului

## Bibliografie/Resurse

[Export to PDF](#)

[Tutorial ecran ST7735](#)

[Biblioteca debounce](#)

[Biblioteca UCGLIB](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/apredescu/tomodachi>



Last update: **2021/06/05 15:20**