

C4T3-LU (remote controlled land drone)

Designed by Ciocea Bogdan-Andrei, 333CA, May 2025

Contact: bogdan.ciocea@stud.acs.upb.ro

Github: [BogdanCiocea](#)

Introduction

Product description

- C4T3-LU is a land-based drone remotely controlled by hand gestures or button input.

Purpose

- Initially, the device was designed for ~~bombing~~ **clearing dangerous areas** where human access is not safe (e.g., military use).
- Later, after numerous high-risk field scenarios, it was reimaged as a **companion robot** — similar to a dog, hence the name.

Functionality

The device moves in response to user commands. It supports two input modes:

- Button control
- Hand gesture control

It can move in all primary directions: **FORWARD, BACKWARD, LEFT, RIGHT**, and diagonally.

Update: I added **the bomb**. Now the hand controller can activate it.

The idea behind it

I've always wanted a dog, but I'm allergic to animal fur — so I built one that doesn't shed and eats less.

Simple as that.

Why is it useful

Thanks to its friendly and interactive nature, the robot can be used for entertainment and companionship.

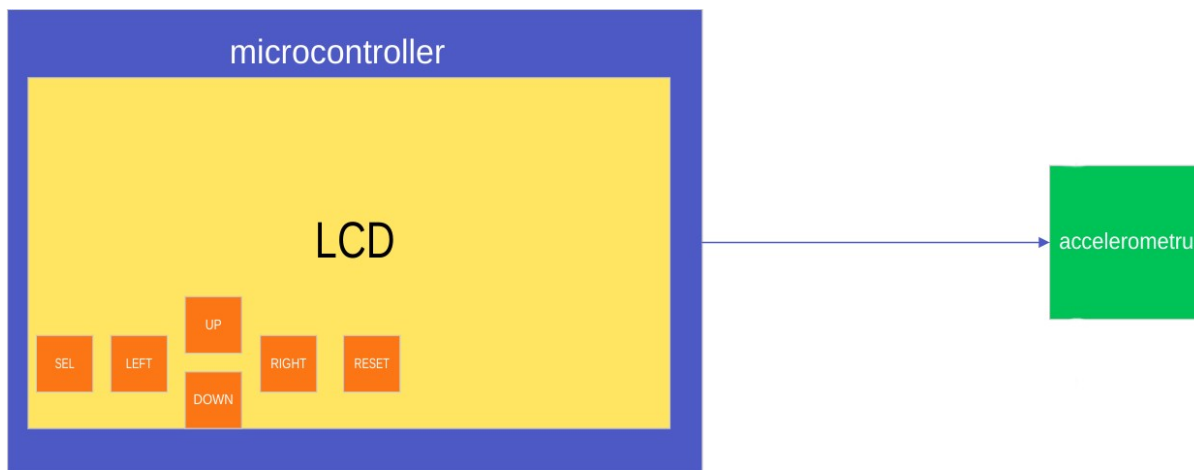
It can also serve as a **carrier**, helping users carry small objects.

Additionally, with a camera module added, it can be repurposed for **filming videos** or **remote surveillance**.

General description

The **hand controller** includes the following components:

- [LCD 1602 shield](#) - displays the current action sent to the robot and provides GPIO input via buttons.
- [ATmega328P and ESP8266 Board](#) - reads GPIO input and transmits commands wirelessly to the robot.
- A custom case made with a 3D printer.
- [Grove 3-Axis Digital Accelerometer](#) - communicates via I2C to detect angle rotation
- [I/O Grove Base Shield](#) - manages connections to all hardware components.



The **robot** includes the following components:

- [ATmega328P and ESP8266 Board](#) - receives the commands from the controller and rotate the motors accordingly
- [Gear Motor Dual Shaft](#) - controlled through a motor driver using **PWM** (for speed) and **digital I/O** (for direction).
- [Micro Servomotor SG90](#) - controlled via a single **PWM** pin to adjust rotation angle.
- [Support for ESP32 camera module](#) - can be used independently for **WiFi video streaming** or interfaced via UART/SPI if needed.
- [HC-SR04 Ultrasonic Sensor](#) - obstacle detection using **digital I/O** pins (trigger and echo).



The **bomb** includes the following components:

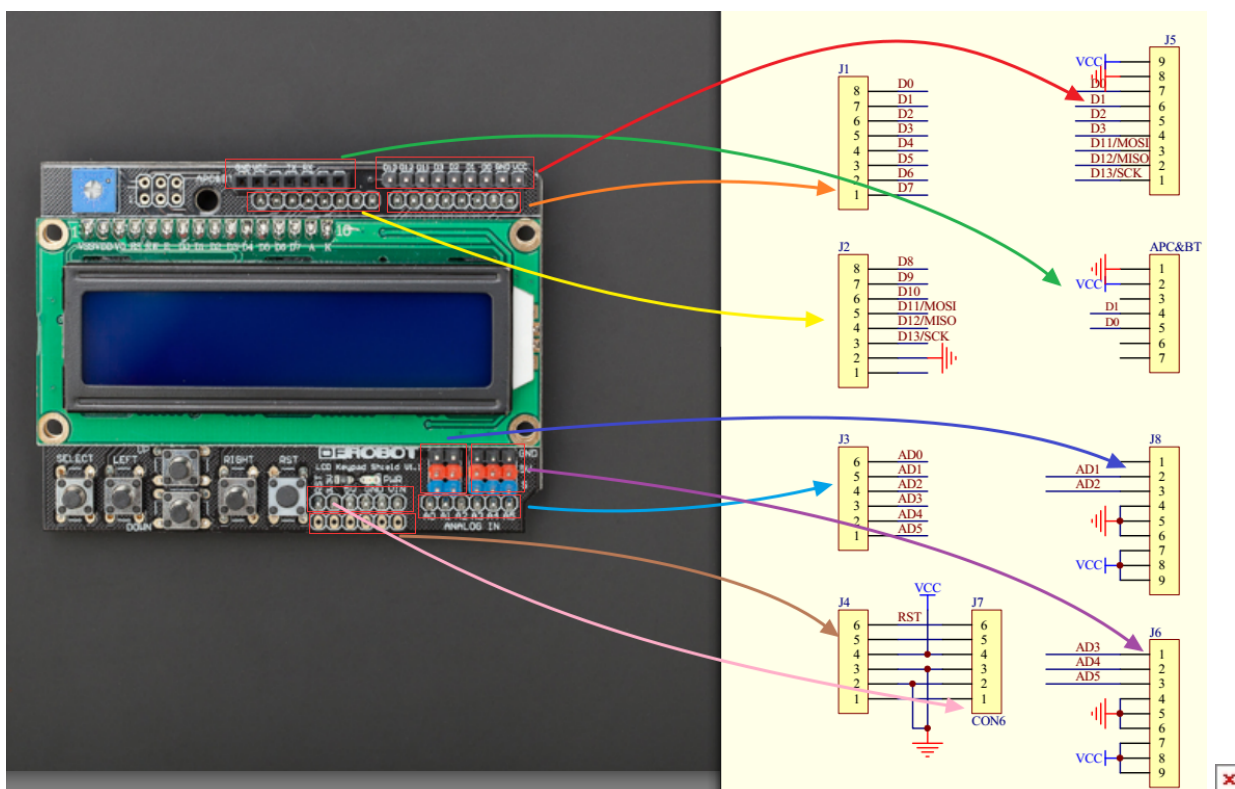
- **Buzzer** - a buzzer which connects to a digital port
- **Led** - a led which will light up to show it's active



Hardware Design

List of components:

- **LCD 1602 shield**



- **Grove 3-Axis Digital Accelerometer**



- **I/O Grove Base Shield**



- **ATmega328P and ESP8266 Board**



- **Gear Motor Dual Shaft**



- **Micro Servomotor SG90**



- [Support for ESP32 camera module](#)



- [HC-SR04 Ultrasonic Sensor](#)



- [Buzzer](#)



- [Led](#)

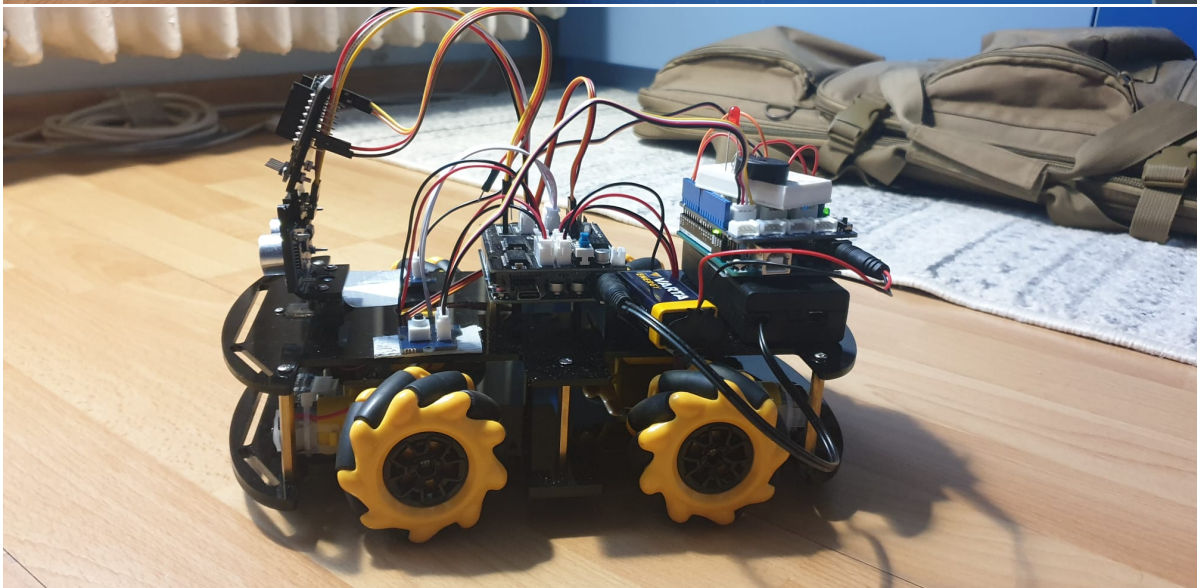


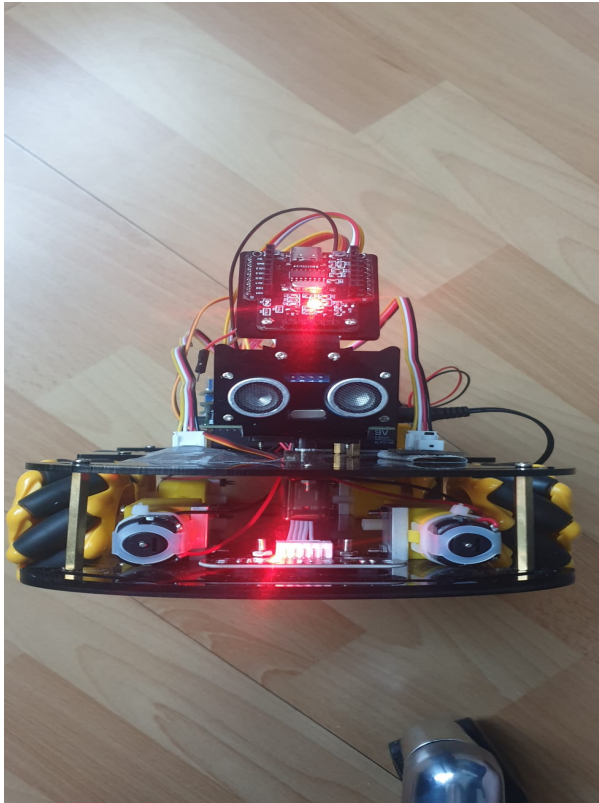
Bill of Materials

Component	Link	Datasheet	Price
LCD 1602 shield	Vendor	Datasheet	24 lei
Grove 3-Axis Digital Accelerometer	Vendor	Datasheet	46.38 lei
I/O Grove base shield	Vendor	Datasheet	38.08 lei
ATmega328P and ESP8266 Board	Vendor	Datasheet	180 lei
Gear Motor Dual Shaft	Vendor	Datasheet	128 lei
Micro Servomotor SG90	Vendor	Datasheet	12 lei
Support for ESP32 camera module	Vendor	Datasheet	61.32 lei
HC-SR04 Ultrasonic Sensor	Vendor	Datasheet	6.49 lei
Buzzer	Vendor	Datasheet	0.99 lei
Led	Vendor	Datasheet	0.30 lei

Total 497.56 lei

Images





Software Design

The code was written in ArduinoIDE. I used these libraries:

- [Arduino_SensorKit](#) - for communicating with the controller accelerometer
- [LiquidCrystal](#) - for showing on a 16×02 display the commands
- [ESP8266WiFi](#) - to know the MAC address of the boards
- [espnow](#) - for communicating with the boards
- [neotimer](#) - a non blocking timer for not using standard delays

The software consists of 6 source files — two for each board (ESP8266 and ATmega328P on controller, robot and bomb).

□ Hand Controller

The **hand controller** acts as a **master**. It reads user input from buttons or an accelerometer and sends commands to the robot over ESPNOW.

The board used combines two microcontrollers:

- **ATmega328P** – responsible for reading button presses, sensor input or just displaying commands.
- **ESP8266** – communication by sending commands using ESPNOW to the robot.

The `LCD shield` is used as a display to show the user which command is transmitted.

The ATmega328P sends user input as a **command string** to the ESP8266, which then formats and transmits the command to the slaves (robot and bomb).

The hand controller also can activate the bomb when the arm is controlling and the **SELECT** button is pressed.

□ Robot

The **robot** acts as a **slave**, waiting for incoming requests from the hand controller.

On receiving a command:

- The **ESP8266** extracts the command string.
- It forwards this string via **USART** to the **ATmega328P**.
- The **ATmega328P** interprets the command and activates the motors accordingly (via PWM and digital I/O signals).

□ Bomb

The **bomb** acts also as a **slave**.

On receiving a command:

- The **ESP8266** extracts the command string.
- It forwards this string via **USART** to the **ATmega328P**.
- The **ATmega328P** interprets the command and activates the bomb.
- The bomb cannot be deactivated for obvious reasons.

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**.

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/prj2025/iivasciuc/bogdan.ciocea> 

Last update: **2025/05/21 17:51**