PROTOTYPING AN EXPERIMENTAL MACHINE-TYPE MODEL

BUZDUGAN Andrei[1], CHIRICU Rareș Dumitru[1] and S.l.dr.ing. DUGĂEȘESCU Ileana[2]

[1]Faculty: Industrial Engineering and Robotics, Study field: Industrial Informatics, Year of study: 2,
e-mail: andreibuzdugan20@gmail.com
[2]Faculty of Industrial Engineering and Robotics, Manufacturing Engineering Department, University POLITEHNICA of Bucharest

ABSTRACT: Ziggi is a remote-controlled car, built using an Arduino Uno board, an L293D shield, and an HC-05 Bluetooth module, which allows wireless control of the car. Equipped with four DC motors for movement, Ziggi can be controlled through an Android app. The car features an intelligent 'return home' function, allowing it to follow the reverse route. The chassis and auxiliary mounting elements were designed, modeled, and created using a 3D printer. Additionally, the car is equipped with a GoPro camera, providing real-time visualization of the route it travels, thus facilitating control and navigation of the car in various environments.

KEYWORDS: Arduino, Ziggi, GPS, car

1. Introduction

The general aspects of the work relate to the design and construction of a remote-controlled car, called Ziggi. This project involves the use of components such as Arduino Uno, Bluetooth HC-05, DC motors and an L293D shield for motor control. Focusing on the integration and coordination of these components, the general aspects aim to create a functional and efficient vehicle that can easily be wirelessly controlled. The main objective of this work is to create an experimental machine-type model that will have functionalities such as remote control, movement in all directions, returning to the starting point through the 'return home' function, and real-time route visualization. Another important objective is the development of an Android application for the wireless control of the car. Thus, the project aims to integrate accessible technologies and efficient programming methods to build an intelligent and versatile vehicle, controlled through a dedicated application on mobile devices.

2. Current stage

In the documentation process for creating the Ziggi car, various sources and examples of similar robotic vehicles were sought. However, no exact or very close examples to the proposed concept for Ziggi were found. A project found in the specialized literature [1] was analyzed, where the author describes the creation of a GPS-controlled robotic vehicle. This robot uses a GPS module, a microcontroller, and a motor driver to move autonomously along a predetermined route. The system uses GPS coordinates to guide the vehicle to its destination and to avoid obstacles encountered along the way [1].

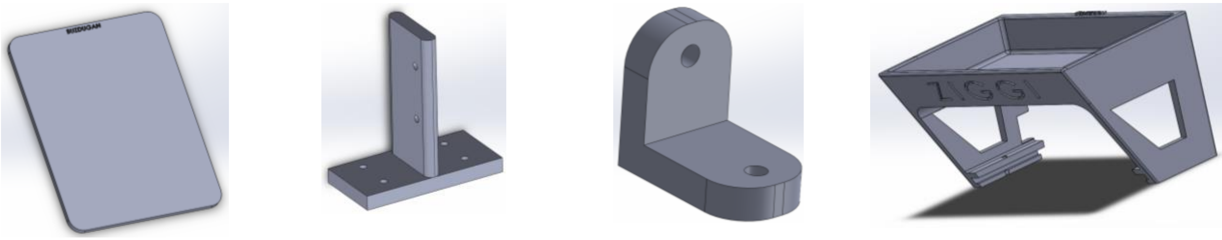
One of the main differences between Ziggi, the constructed car, and the robot presented in the article [1], is the navigation system. While Ziggi uses a route recording method based on motor control and direction of movement, the GPS robot relies on GPS coordinates to establish the route. This means that Ziggi can be used both indoors and outdoors, having the advantage of retaining the traveled route. In contrast, the GPS-controlled robot requires an open space to receive the GPS signal, thus limiting its use in enclosed areas.

3. Prototype Development

In this chapter, the main stages of creating the experimental prototype are presented, exploring in detail each aspect and how they were approached to ensure functionality, efficiency, and proper assembly of the Ziggi car.

3.1. Designing the component parts of the experimental model

All 3D-printed components were modeled and designed using specialized software.



a) b) c) d)

Fig. 1. Modeled component parts

The design stages of each component are as follows: the work plan is chosen, the sketch is drawn, and the correct dimensions are established. Then, the piece is extruded to obtain the 3D modeled components [2].

In Figure 1a the car chassis is shown. Its sketch is a rectangle with dimensions of 150x120 mm. The extrusion is 4 mm to ensure increased strength, considering that the components presented in point 3.3 will be mounted on it.

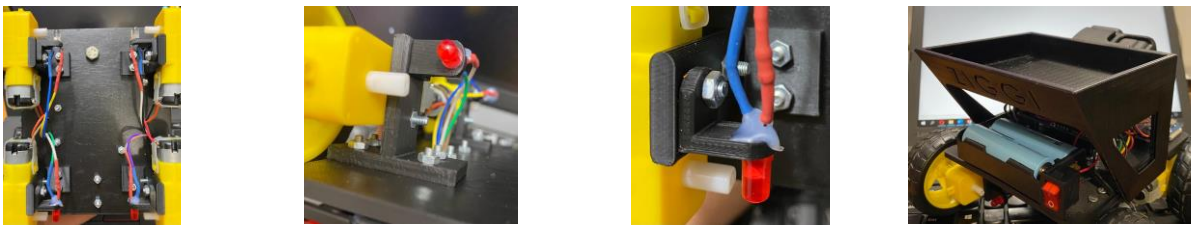
The support in Figure 1b is used for mounting the motors on the chassis with the help of M3 screws.

In Figure 1c a support for LEDs can be seen. It started from the sketch of the letter L, for which the dimensions were established, then circles with a diameter of Φ3 mm were drawn, and the corners were rounded. The thickness of the support is 3 mm.

A detachable support, shown in Figure 1d, is provided for attachment to the chassis, which can be used for transporting various objects.

3.2. 3D printing of experimental model components

After completing the modeling of the parts, the next step was to obtain the .STL files for the purpose of 3D printing them.



a) b) c) d)

Fig. 2. Printed component parts

For 3D printing, the Ultimaker Cura software [3] program was used. The material of choice is PLA (Polylactic Acid), printing was performed with a 40% infill, the print bed temperature was set at

60°C, and the nozzle temperature at 200°C. The parts shown in Figure 1b and 1c were 3D printed in four copies.

3.3. Assembling the component parts of the experimental model

After completing the printing process of the components, the assembly stage followed. In this stage a connection scheme was developed to allow for the efficient connection of the Arduino UNO board, the motor shield, motors, LEDs, the on/off button, 220Ω resistors, the HC-05 Bluetooth module, and the batteries, as shown in Figure 3.

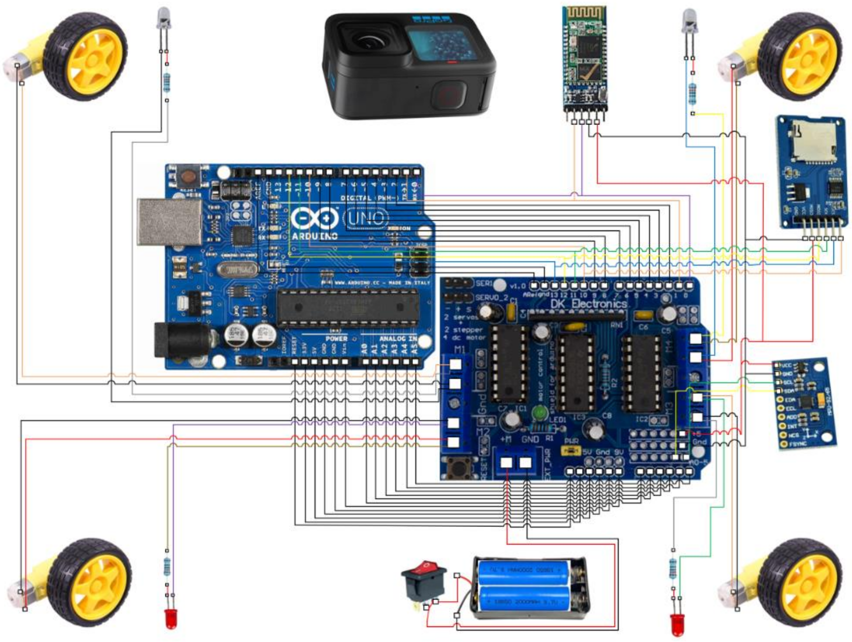


Fig. 3. Connection diagram between components

The assembly of the components was carried out in several stages, and Figure 4 shows some illustrative photos of the assembly and interconnection process of the parts.

In the first step the pins were soldered onto the L293D shield, as well as the wires needed for connecting the HC-05 Bluetooth module, specifically the RX and TX pins. The connections between the module and the Arduino UNO control board were made crosswise (RX-TX and TX-RX), as this configuration is commonly found when using this module. Additionally, the wires from the MicroSD module were soldered for the CS, SCK, MOSI, and MISO pins.

In Figure 4a the L293D shield can be seen, which allows connection to the analog pins (A0-A5) and to six +5V and GND pins, through soldered connector pins. Figure 4b shows the stage where the board was mounted on the vehicle chassis. Figure 4c highlights the soldered RX and TX wires corresponding to the TX and RX pins on the module, and Figure 4d illustrates the shield connected to the board. The next figure shows the stage of analyzing the available space on the chassis for assembling the following components.

Figure 4f highlights the mounting of the ON/OFF switch and the battery holder.

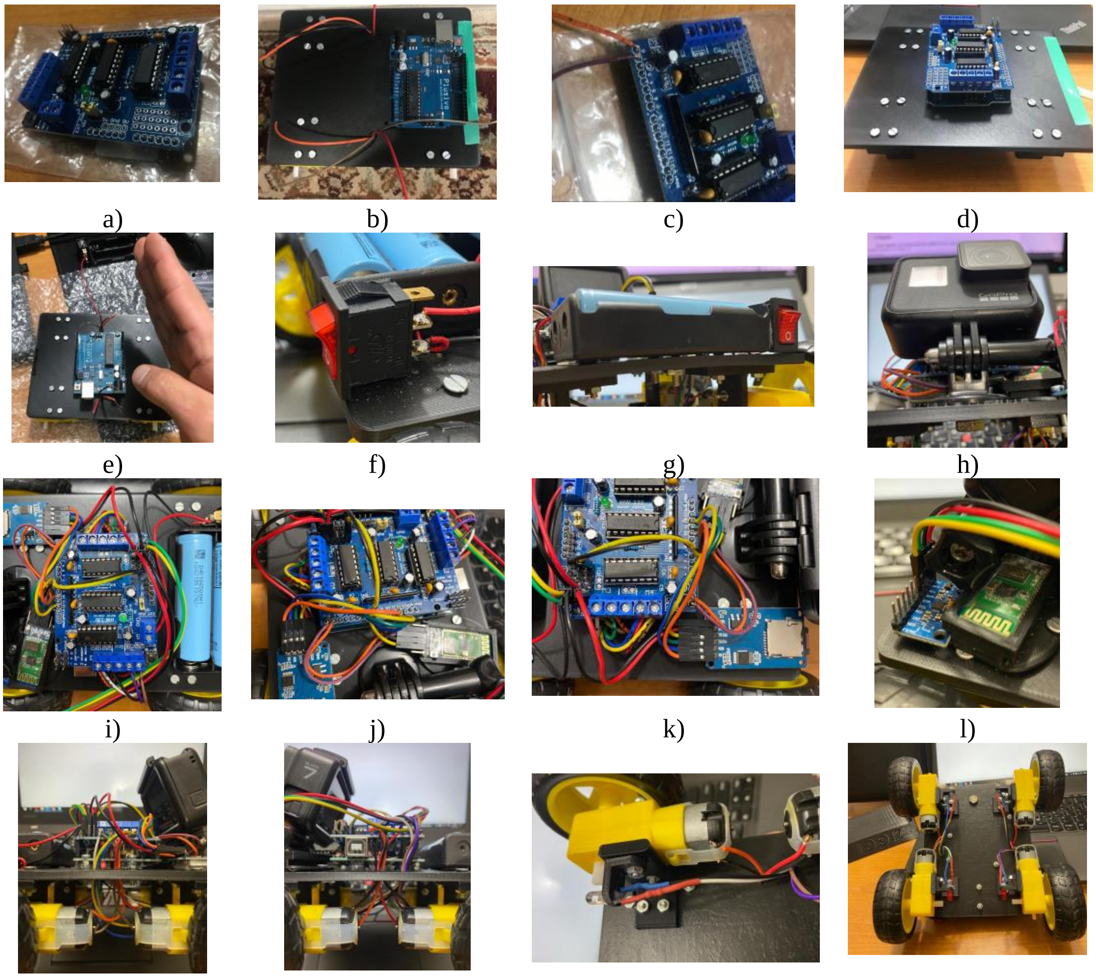
Figure 4g shows how the battery holder is attached to the vehicle chassis. The following image highlights how the GoPro camera is mounted on the vehicle chassis.

Figures 4i and j highlight all the connections on the upper side of the chassis, including the connection method of the HC-05 Bluetooth module.

The next two figures illustrate the mounting of the MPU-92/65 and MicroSD modules. These modules were mounted on the car chassis and can be used to obtain greater accuracy of the return path within the "return home" command. The MPU-92/65 module can be used to obtain a more accurate measurement of the XYZ position coordinates, with the data being stored on the attached MicroSD card. These components can be used for further development of the experimental model.

In Figures 4m and n the side views of the chassis can be seen from both the right and left sides.

Figure 4o shows the assembly method of the 3D printed parts for fixing the DC motors and connecting the LEDs, using 220-ohm resistors. The last figure highlights all the connections on the lower side of the chassis [4].



a) b) c) d) e) f) g) h) i) j) k) l) m) n) o) p)

Fig. 4. Assembly steps of the component parts

4. Description of functionality and testing of the experimental model

To implement the main control functions of the car, the code was developed and uploaded using the Arduino IDE development platform [5].

In developing this code, the AFMotor.h and LinkedList.h libraries [5] were used to simplify motor control and store the robot's coordinates in an efficient way, using the following functions listed below:

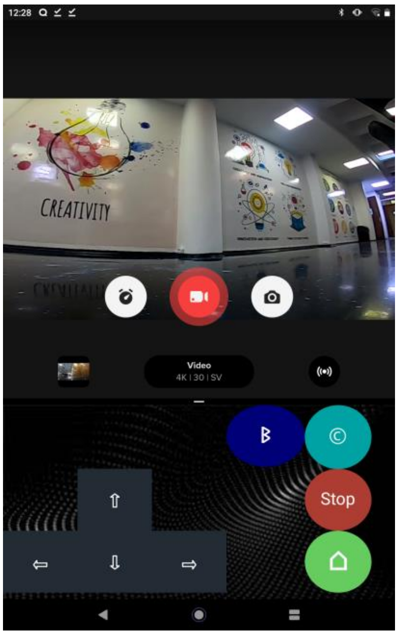
- The "controlMotors" function is important because it controls the robot's movement through the motors. It receives direction commands "F, B, L, R" and stop and clear commands "C" and "S". Through this function, the robot's direction and speed can be controlled.

- The "updatePosition" function is important for updating the robot's coordinates and storing them in LinkedList. It receives the robot's current direction and updates the coordinates accordingly. After updating, a "Coordinate" object with the new coordinates and the current direction is created and added to the end of the "LinkedList path".

- The "returnHome" function is essential for allowing the robot to return to the starting point, reversing the path taken by the robot. To achieve this, the motors are controlled by the "H" command to move in the opposite direction, according to the route, and wait for a specific duration to ensure the robot has stopped before moving to the next coordinate.

- Finally, the "loop" function is important for processing user commands through the serial port. It controls the robot's movement through the "controlMotors" and "updatePosition" functions and allows the user to send commands to control the robot.

In Figure 5a the user interaction interface is illustrated, which includes two components: the real-time viewing application from GoPro and the control one where customized buttons have been designed to facilitate an intuitive experience in sending instructions to the mechanical device [6], [7].



a)

b)

Fig. 5. a) Real-time visualization and control application of the experimental model, b) Final model

The search and connection functions of the application to the HC-05 Bluetooth module have been implemented, and the buttons corresponding to the uppercase letters in the source code (F, B, L, R, C, S, H) have been configured as follows:

F - forward;
B - backward;
L - left;
R - right;
C - clear;
S - stop;
H - return "home".

To evaluate the functionality of the code, factors such as the friction coefficient, which depends on the material of the contact surface between the wheels and the terrain, were analyzed. We included in the code the adjustment of the rotation factor for left and right turns, allowing the optimization of the vehicle's behavior on various surfaces.

Another factor influencing the direction's accuracy is the slight curvature of the chassis obtained during 3D printing, which affects the alignment and balance of the steering system. By adjusting some parameters in the program, these imperfections can be compensated for and the steering performance of the vehicle can be improved under various operating conditions.

5. Conclusions

The Ziggi project is a remotely controlled car, made by integrating Arduino Uno components, Bluetooth HC-05, DC motors, and an L293D shield. This versatile vehicle can be controlled through an Android application and allows movement in all directions, returning to the starting point, and real-time route visualization. The chassis and components were designed and 3D printed, and assembly and programming were detailed in subsection 3.3. As a further development, improving the return route's precision by adding the MPU-92/65 and MicroSD components can be considered.

6. Bibliography

[1]. CIRCUIT CELLAR, GPS Guides Robotic Car, available at: https://circuitcellar.com/research-design-hub/gps-guides-robotic-car-2/ , accessed on: 15.04.2023;
[2]. SOLIDWORKS 2015: A Power Guide for Beginner and Intermediate Users, ISBN: 1511798424, 2015;
[3]. ULTIMAKER CURA, 3D printing software, available at: https://ultimaker.com/software, accessed on: 18.04.2023;
[4]. Cazan-Gheorghiu, C., Electronics and Robotics. First Steps, Editorial Libris, Brașov, 2018, ISBN 978-606-8953-89-262;
[5]. ARDUINO IDE, Professional software, available at: https://www.arduino.cc/en/software, accessed on: 20.04.2023;
[6]. MIT APP INVENTOR, Create Apps!, available at: http://appinventor.mit.edu, accessed on: 21.04.2023;
[7]. GOPRO, Quik, available at: https://gopro.com/en/us/shop/quik-app-video-photo-editor, accessed on: 22.04.2023;