Dept. of Electrical & Computer Engineering

Effat University

Jeddah, Saudi Arabia

Amro Yousef, Layan Turkistani, Tariq Tadmori **Title**: DIY Multi-functional Robot Car Using Arduino

**Course**: ECE202 and ECE254L

**Supervised by**: Dr.Nema Salem

**Table of content**

Abstract…………………………………………………………………………………………......3

Introduction………………………………………………………………………………………...3

Components and there features…………………………………………………………………...4,5

Chassis Assembly………………………………………………………………………….6

Sensors Integration………………………………………………………….………….…6

Building the Robotic Arm………………………………………………………………...6

Wiring and Connections……………………………………………………………….….7

Programming the Arduino………………………………………………………………………...7

Testing and Calibration…………………………………………………………………...7

Refinement and Iteration……………………………………………………………….…8

Code part…………………………………………………………………………………...8,9,10,11

Future Work and Innovation Pathways………………………………………………….12,13

Testing and Troubleshooting………………………………………………………….......13

Results……………………………………………………………………………………..14

Conclusion………………………………………………………………………………….15

Reference………………………………………………………………………………...…15

**Abstract**

The project report is based on the design of a DIY Multifunctional Robot Car using Arduino developed as a prototype of a teaching aid used hands-on for learning and teaching robotics and programming. This robot would make use of a set of components including an Arduino UNO, an L293D motor driver, servo motors, IR sensors, and an ultrasonic sensor to function in tasks such as line following, obstacle avoidance, and manipulation with a robotic arm. "This research paper allows tangible practical applications by combining simple electronic components to come up with a complex and interactive robotic system.".

**Introduction**

As such, the thriving interest in do-it-yourself robotics projects is pushing enthusiasts towards a plethora of applications, facilitated by versatile robotics platforms like Arduino. A great example of what can be done with an Arduino board in creating complex robotic systems is a robot car multifunctional designed by me. In this report, the construction of such a robot car will be put into consideration in depth, enlightening most of the insights gained and challenges faced through the process. Come with us on this mission into the great world of DIY robotics, where Arduino is the open door to limitless creativity and discovery for every robotics hobbyist.

**2. Components and Their features:**

* **Arduino UNO Board:**

The Arduino UNO is a microcontroller board based on the ATmega328P, equipped with 14 digital I/O pins, 6 of which support PWM (Pulse Width Modulation) outputs, and 6 analog inputs. It features a 16 MHz ceramic resonator, USB connection, power jack, ICSP (In-Circuit Serial Programming) header, and a reset button. The board can be powered through a USB cable or an external 6V supply and is programmable via the Arduino IDE (Integrated Development Environment) making it highly suitable for a wide range of electronic projects.

* **L293D Motor Driver Shield**

The L293D Motor Driver Shieldcan drive up to two stepper motors, whether unipolar or bipolar, with configurations including single coil, double coil, or interleaved stepping. It features four H-Bridges with the L293D chipset, supporting up to 0.6A per bridge (1.2A peak) and includes thermal shutdown protection and internal kickback protection diodes. The shield operates motors across a voltage range of 4.5VDC to 25VDC, making it versatile for diverse motor-driven projects.

* **PWM servo motor driver**

A servo motor drive capable of 180-degree rotation precisely controls the angular position of an object within a half-circle range of motion. It includes the servo motor and control electronics that regulate its operation. The system operates by receiving a pulse-width modulated (PWM) control signal indicating the desired position, while an internal feedback mechanism, typically a potentiometer, continuously monitors the motor's actual position. The drive compares this actual position with the desired position and corrects any deviation by adjusting the motor's operation, allowing the motor to turn anywhere within a 0 to 180-degree arc. This precision in controlling position, speed, and acceleration makes 180-degree servo motor drives essential in various applications like robotic arms, RC vehicle steering mechanisms, and automated doors.

* **Wooden arm**
* **IR infrared sensor features**

IR sensors are capable of non-contact detection and distance measurement, absorbing minimal voltage and offering features like material identification, heat sensing, and motion detection, making them versatile for various applications.

* **Ultrasonic sensor**

The Ultrasonic Distance Sensor features a supply voltage of 5V DC, a current of 15mA, a modulation frequency of 40Hz, an output range of 0-5V that goes high upon obstacle detection, a maximum beam angle of 15 degrees, a distance detection range of 2cm to 400cm, an accuracy of 0.3cm, and communicates via positive TTL pulses.

* **Gear motor**

A gear motor that operates up to 3 volts and can reach speeds of up to 1,000 RPM is designed to offer efficient power and speed within a compact size, suitable for small-scale applications where low voltage operation and high rotational speed are crucial.

* **Li-ion battery (7.6V)**

Lithium-ion batteries can last up to 15 years and offer 500-7000 discharge cycles depending on the chemistry, significantly outperforming lead-acid batteries in lifespan and cycle life, thereby reducing replacement costs.

* **Jumper wires**
* **4 wheels**

**3 Chassis Assembly**

**Attach Motors:** Fix the motors on a wooden board chassis using adhesive or screwing, so that wheels are spaced and aligned properly with each other.

**Install Wheels:** The wheels will be fastened on the motor shafts and aligned in such a way as to have free movement.

Securely attach the motor drivers, the Arduino UNO board, and all the other electronic components onto the chassis using the proper mounting hardware or adhesive, respectively ensuring that all the electronic parts are securely placed.

**3.1 Sensors Integration**

Under the front of the robot car, line tracking sensors will be installed, which will be able to sense the lines positioned on the surface. They will, by such means, guide the robot's direction.

**Distance detection sensor:** An ultrasonic sensor to be installed in the front of the robot car to calculate the measuring distance so that it faces the forward side.

**3.2 Building the Robotic Arm:**

**Design Arm Structure:** Arm Structure, What else. Determine the structure and degrees of freedom needed for a robotic arm in the JSON data, keeping in mind reach, pay-load carrying capacity, and maneuverability of the arm.

**Assemble Servo Motors:** Fix the servo motors in alignment to form the joints and segments of the robotic arm.

**Mount Arm:** Attach the robotic arm to the chassis of the robot car, ensuring it is balanced and can move freely without obstruction.

**3.3 Wiring and Connections**

**Connect motors:** Connect motors with the motor drivers and later to the Arduino UNO board, watching polarity and signal connections.

**Wire Sensors:** Connect the IR sensors and ultrasonic sensor to the Arduino UNO board properly, ensuring that they are powered correctly, and the signal pins are connected to the digital or analog input pins.

**Wire Servo Motors:** Wire the servo motors for the robotic arm to Arduino UNO board, interfacing and connecting the signal lines appropriately to enable control of motion.

**3.4 Programming the Arduino:**

**Write Code:** Use the Arduino IDE to write code for controlling the movement of the robot car, interpreting sensor data, and controlling the robotic arm.

**Motor Control:** Implement code to control the speed and direction of the motors based on sensor inputs and user commands.

**Sensor Interfacing:** Write code to read data from the IR sensors and ultrasonic sensor, implementing line tracking and obstacle avoidance algorithms.

**Robotic Arm Control:** Develop code to control the movement of the servo motors, enabling the robotic arm to perform tasks such as picking up objects and placing them in desired locations.

**3.5 Testing and Calibration:**

**Functional Testing:** Power up the robot car and perform comprehensive tests to ensure all components are functioning correctly, including motor movement, sensor readings, and robotic arm control.

**Calibration:** Fine-tune sensor parameters, motor control settings, and servo motor positions as needed to optimize the performance and reliability of the robot car.

**3.6 Refinement and Iteration:**

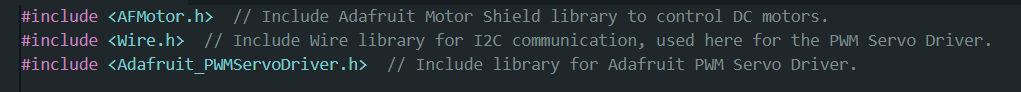
**Evaluate Performance**: Assess the performance of the robot car in various scenarios and identify areas for improvement or optimization.

**Adjust:** Modify code, adjust hardware configurations, or replace components as needed to address any issues or enhance functionality.

**Iterate:** Repeat the testing, evaluation, and refinement process iteratively until the robot car meets the desired specifications and performance criteria.

**4.** **Code part:**

* **library files are included**



* **objects are created for these libraries**

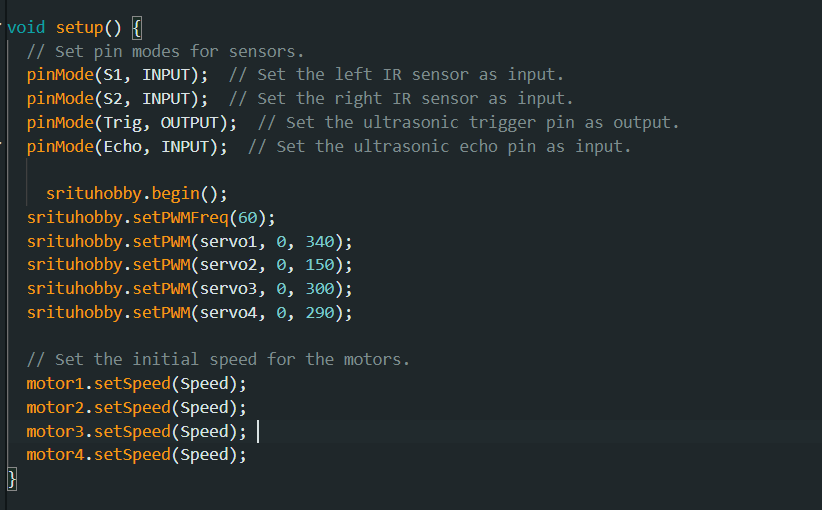
صورة تحتوي على نص, لقطة شاشة, الخط

تم إنشاء الوصف تلقائياً

* **the sensor pins and servo motor pins are defined**

صورة تحتوي على نص, لقطة شاشة, الخط

تم إنشاء الوصف تلقائياً



* **the setup function**
* **Follow line function**

صورة تحتوي على نص, لقطة شاشة, برمجيات, الخط

تم إنشاء الوصف تلقائياً

* **Void obstacle function**

صورة تحتوي على نص, لقطة شاشة, برمجيات, الخط

تم إنشاء الوصف تلقائياً

* **Ultra** **Sonic** **function**

صورة تحتوي على نص, لقطة شاشة, الخط

تم إنشاء الوصف تلقائياً

* **Removing obstacle function**

صورة تحتوي على نص, لقطة شاشة, الخط

تم إنشاء الوصف تلقائياً

**5. Future Work and Innovation Pathways:**

* Advanced Sensor Integration:

Explore the incorporation of advanced sensor technologies, such as lidar or camera-based systems, to enhance the robot car's perception capabilities. These sensors could enable more sophisticated navigation, object recognition, and environment mapping, paving the way for autonomous operation in complex scenarios.

* Machine Learning and AI Integration:

Investigate the integration of machine learning algorithms and artificial intelligence techniques to enable the robot car to learn from its environment, adapt to changing conditions, and make intelligent decisions autonomously. This could involve training neural networks for tasks such as object detection, path planning, and decision-making, expanding the robot car's capabilities beyond simple reactive behaviours.

* Autonomous Navigation:

Develop algorithms and strategies for autonomous navigation, allowing the robot car to explore its environment, navigate dynamic obstacles, and reach predetermined destinations without human intervention. This could involve implementing SLAM (Simultaneous Localization and Mapping) techniques, trajectory planning algorithms, and obstacle avoidance strategies to enable robust and reliable autonomous operation.

* Task-Specific Applications:

Explore the development of specialized applications and functionalities tailored to specific tasks or domains. For example, the robot car could be adapted for use in agriculture, search and rescue operations, or warehouse logistics, with custom-designed features and capabilities optimized for each application.

* Human-Robot Interaction:

Investigate methods for enhancing human-robot interaction, allowing users to interact with the robot car intuitively and efficiently. This could involve integrating speech recognition, gesture recognition, or natural language processing capabilities to enable seamless communication and collaboration between humans and robots.

* Collaborative Robotics:

Explore the potential for collaborative robotics, where the robot car works alongside humans to accomplish tasks in shared environments. This could involve developing algorithms and safety mechanisms to ensure safe and efficient interaction between humans and robots, enabling cooperative teamwork and synergy between human operators and autonomous robotic systems.

* Open-Source Development and Community Collaboration:

Foster an open-source development ecosystem and encourage collaboration within the DIY robotics community. By sharing designs, code, and resources openly, enthusiasts can collectively innovate and accelerate the development of new technologies and applications, driving the field of robotics forward and empowering individuals to create impactful solutions to real-world challenges.

**6. Testing and Troubleshooting**

During the project, detailed testing and troubleshooting exercises were done; all are geared at ensuring functionality and reliability of the DIY multifunctional robot car.

The main activities include the following:

* **Functional testing**: It was to ensure the functional status in each component, including motors, sensors, and wireless communication modules of the product, was established.
* **Sensor calibration:** Optimizing the sensor performance through changing the parameters for the right obstacle’s detection.
* **Navigation Testing:** Check the robot's capability for obstacle negotiation and path-following testing with due accuracy.
* **Behaviour testing with Autonomous:** Test decision-making ability of the robot with sensor real-time data.

**Troubleshooting:**

**Connection Issues:** An occasional connection problem was resolved by rechecking and securing all of the connections.

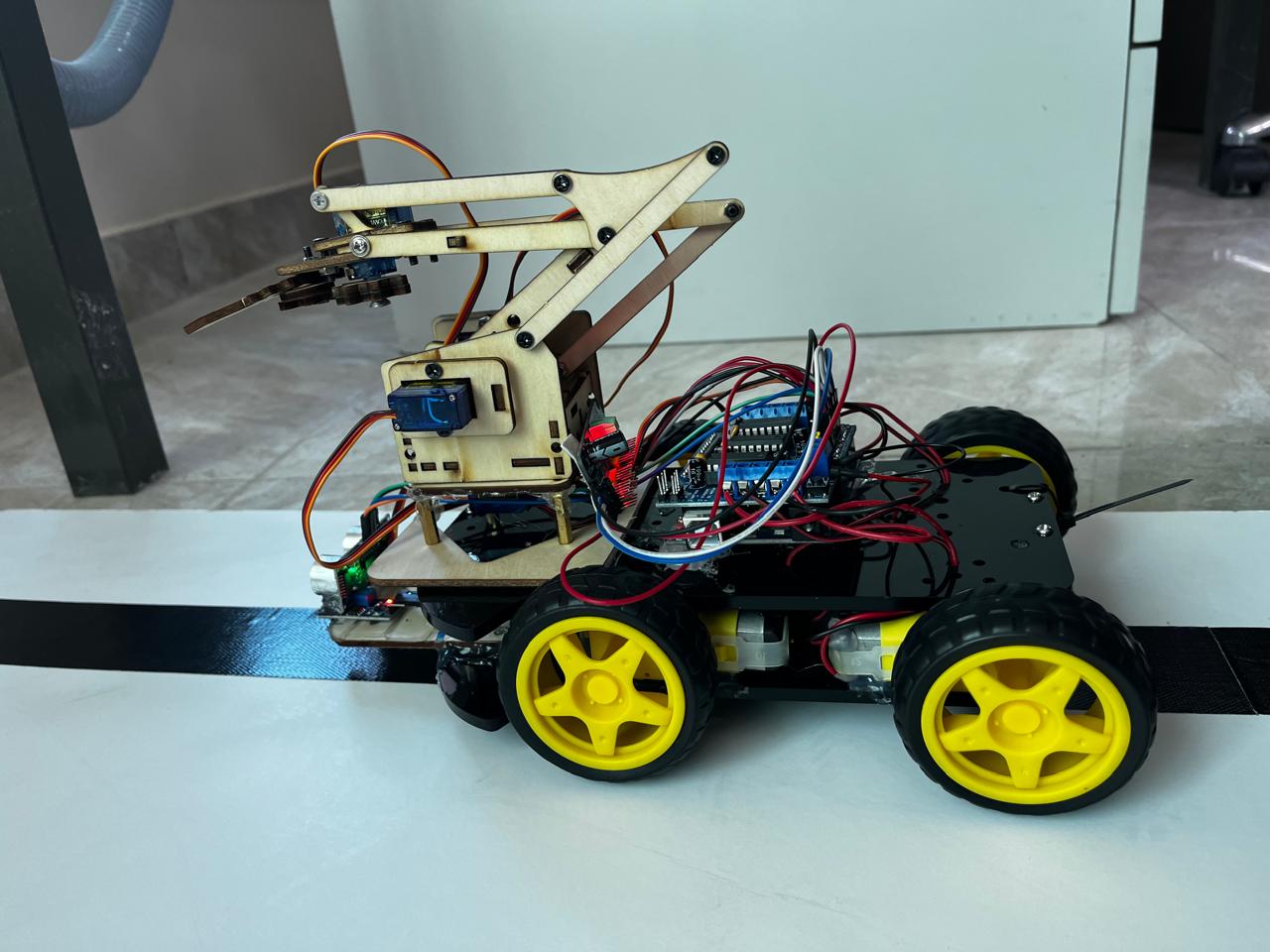
**Code Debugging:** Debugged Arduino code to clear up syntax errors and logic bugs.

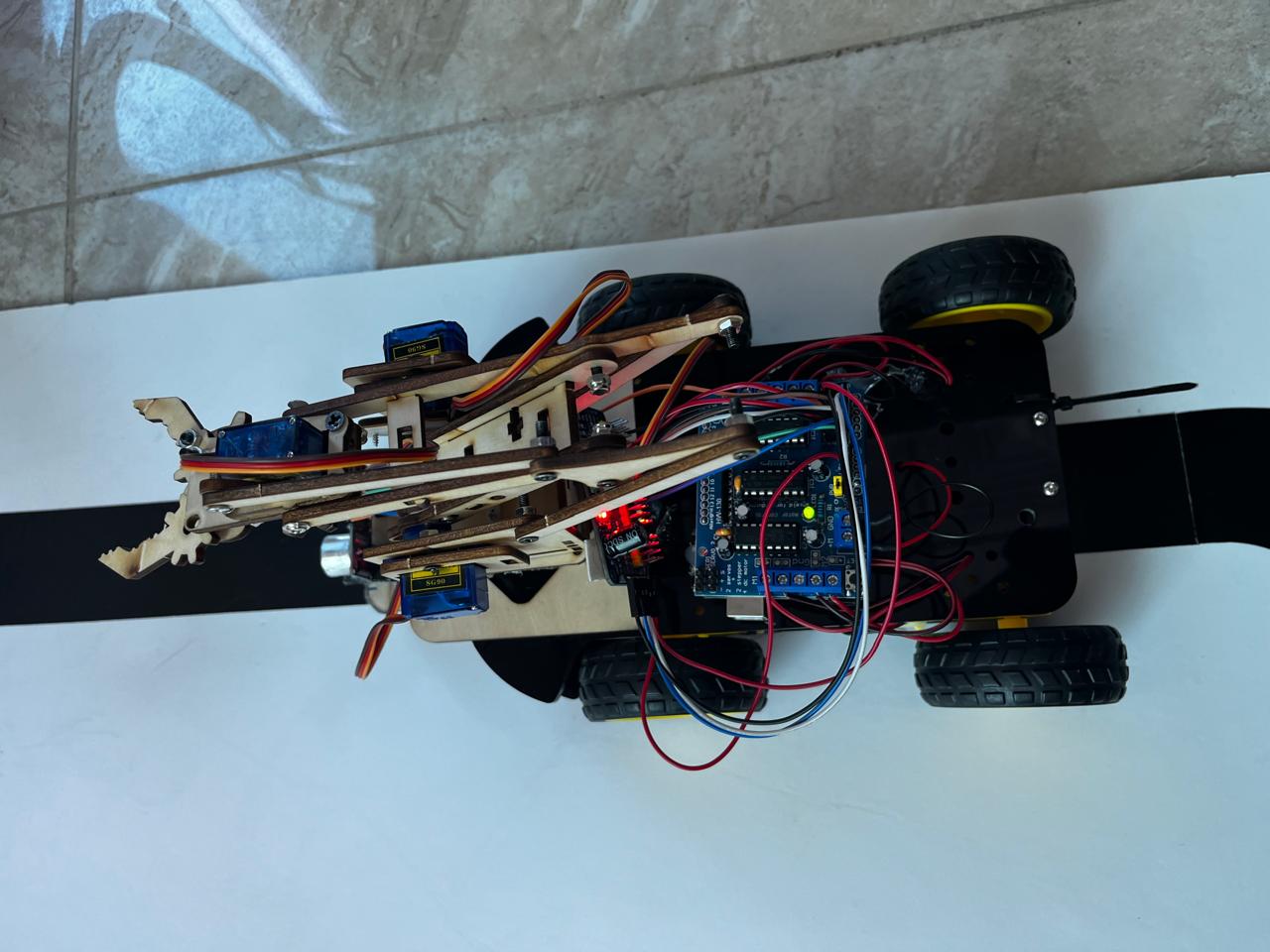
**Sensor Alignment:** Alignment of sensors was done to be more accurate and reliable.

**Interface:** Minimized electrical interference for clear and stable wireless communication.

**Optimization of Power Supply:** Ensured suitable power supply through optimized interconnection of batteries and battery management**.**

**7. Results**





A hand holding a toy car

Description automatically generated

**8. Conclusion**

In simple final words, it can be said that the whole process to design, build, and test a DIY Multifunctional Robot Car using Arduino is basically a journey full of learning, challenges, and ultimately, accomplishing.

This project has served enthusiasts in the community of DIY robotics, benefiting them in the form of practical experiences and valuable skills, insights directly obtained through hands-on practice and practical application of electronics, programming, and robotics principles. Each and every phase of the project, right from the planning stages to the final integration of sensors, motors, and actuators, has humbled us to understand Robotics and Automation.

This participative effort in selecting the components, fabricating the chassis, integrating sensors, and programming the Arduino is an example for the evincing of creative, teamwork-oriented, and problem-solving skills. While the project has been successful in the main part of developing a functional robot car that moves around in its environment and performs very basic tasks, there is always scope for further development and improvement.

Future improvements may include increasing the accuracy of sensor accuracy, optimization of motor control algorithms, and integration of sophisticated artificial processes for autonomous operation. These are the challenges and hard setbacks that come in the way, but then. the ultimate perseverance and dedication of the DIY robotics community do pay off one day. The enthusiasts will get a strong impulse to push further the limits of DIY robotics and inspire others in the same way on their way to research and discovery with a spirit of constant improvement and innovation. In essence, the DIY multi-functional robot car project serves as a testament to the ingenuity, creativity, and collaborative spirit of the maker community.

**9. Reference:**

https://srituhobby.com/how-to-make-a-diy-multi-functional-robot-car-using-arduino/#google\_vignette

<https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf>

<https://5.imimg.com/data5/PX/UK/MY-1833510/l293d-based-arduino-motor-shield.pdf>

<https://cdn-learn.adafruit.com/downloads/pdf/16-channel-pwm-servo-driver.pdf>

<https://techzeero.com/sensors-modules/ir-sensor/#google_vignette>

<https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>

<https://www.gotronic.fr/pj2-com-motor01-datasheet-2079.pdf>