CE324 (Microprocessor Interfacing) Project: Light Following and Bluetooth Remote-Controlled Robot

Gul Zainab
Faculty of Computer Science & Engg.
GIK Institute of Engg. Sciences & Tech.
Topi, Khyber Pakhtunkhwa, Pakistan.
2021173

Romaisa
Faculty of Computer Science & Engg.
GIK Institute of Engg. Sciences & Tech.
Topi, Khyber Pakhtunkhwa, Pakistan.
2021538

Warda Farhan
Faculty of Computer Science & Engg.
GIK Institute of Engg. Sciences & Tech.
Topi, Khyber Pakhtunkhwa, Pakistan.
2021750

***Abstract*—The team utilized electronic hardware components, including a micro-controller, motors, motor driver and sensors, as well as pertinent program, to construct a multi-functional robot, with an app allowing users to switch between the light-following and remote-controlled functionalities of the robot.**

***Index Terms*—Arduino, sensor module, light-dependent, Bluetooth, DC motor, H-bridge, Arduino IDE, C, MIT App Inventor**

I. Introduction

A micro-controller is a compact integrated circuit designed to govern a specific operation in an embedded system, and includes a processor, memory and input/output (I/O) peripherals on a single chip. They are found in vehicles, robots, office machines, medical devices, mobile radio transceivers, vending machines and home appliances, among other devices. Therefore, using the micro-controller Arduino and the knowledge acquired from our Microprocessor Interfacing Lab, we are utilizing the the equipment issued to devise a working multi-functional robot.

II. Methodology

A. Hardware Components

1) Arduino UNO R3 with DIP cable
   Arduino UNO is a micro-controller able to receive input from various sensors or modules; it consists of 28 pins, which are listed below:
   - 14 digital pins (including TX and RX)
   - 6 analogue pins
   - 3 ground pins
   - 2 power sources (3.3 and 5 V)
   - 1 Vin pin
   - 1 AREF
   - 1 reset pin
2) Digital LDR Module
   Light detecting module, as the name suggests, detects light and its intensity; it 3 pins (VCC, GROUND, DIGITAL PIN), with advanced LDR modules also contain an analogue pin. Our team has made use of 1 3-pin LDR and 2 4-PIN LDR sensor modules.
3) L298 Motor Driver
   The L298 Motor Driver represents an integrated circuit (IC) used for the purpose of controlling motors–in this case, DC motors–in robots or any embedded systems. Consisting of 4 output pins, 4 input pins, 2 enable pins, and 3 power pins (12V, 5V, GROUND), it is also commonly referred to as an H-bridge.
4) HC-05 Module
   This module serves as the Bluetooth connection module, utilized for wireless communication with the robot. The 5 pins encompass:
   - 1 TX PIN (transmit information)
   - 1 RX PIN (retrieve information)
   - 1 VCC
   - 1 Ground
   - 1 State
   - 1 Key
5) 2WD Smart Robot Car Chassis Kit
   This kit consisted of a 2-wheel chassis, 2 DC motors, 2 main wheels, 1 small supporting wheel, and corresponding screws.
6) 12 V Power Source
   To supply this, lithium ion batteries were connected.

B. Pin Configurations

The team made use of three 3.7 V lithium ion batteries, providing a total of 11.4 V supply to motor driver (H-bridge). Right motor is connected to OUT1 and OUT2 of H-bridge, while left motor is connected to OUT2 and 4. The inputs pins of the H-bridge are connected to pin numbers 5, 4, 2, and 3 of the Arduino respectively.

In addition, three LDR sensor modules and 1 HC-05 module are utilised. Right, left and central LDRs are connected to the pins 8, 9, and 10 of the respectively, while the transmit pin of the HC-05 module is connected to the RX pin of the Arduino. Similarly,the receive pin of the Bluetooth module is connected to the TX pin of Arduino.

The 5 volt from H-bridge is connected to the breadboard, supplying power to the Arduino, sensors and the Bluetooth module; likewise is the case with ground.

C. Technologies

D. Working of the Program

First and foremost, pin configurations are set for motor control to initiate the serial communication. Motor control pins are defined as OUTPUT.

For implementing remote-control feature, in the loop section we added a “do() while()”; if Bluetooth input is available, the direction is read until ”Disable” is received. The user presses the buttons on the app to control the direction of the robot; therefore, the robot moves according to the directions communicated via the remote-control buttons.

If the direction is ”Forward,” ”Back,” ”Right,” ”Left,” or ”Stop,” it prints the corresponding letter and calls the respective user-defined motor control functions.

The functions for the direction/movement have been defined separately to ensure clarity and neatness of the code; the motors operate on opposite polarity in the case of our robot–that is, when positive terminal is low and negative is high, the motors spin clock-wise and hence wheels rotate in the forward direction. For forward and backward manoeuvre, both wheels rotate in one direction; however, for left and right turns, the left and right wheel respectively rotate forward whereas the counterpart wheel rotates in the opposite direction.

If no Bluetooth input is available, light-following functionality comes into action. It read values from LDR sensors, and thus, the conditions based on the LDR readings determine the robot’s behaviour. Digital Input 0 refers to light detected on the sensor and 1 refers to no light detected. If light falls on all three sensor, the robot will move forward; if it falls on the left and center sensor, or solely the left sensor, it will turn left. Similarly, if light falls on the right and center sensor, or solely the right sensor, it will turn right. Palpably, if no sensor detects any light, then the robot will cease motion.

E. Representing the Code

The C code that was uploaded to the Arduino can be aptly represented by the following pseudo-code:

Algorithm 1 Initialisation/Setup

```
Initialize pins for motor control (RMP, RMN, LMP, and LMN) and LDR sensors (LDR_R, LDR_L, and LDR_C)
Initialize a variable 'direction' to store Bluetooth input
Setup:
Set motor control pins (RMP, RMN, LMP, LMN) as OUTPUT
Begin serial communication at 9600 baud rate =0
```

Algorithm 2 represents the Bluetooth code within the main loop function.

The following table represents the output movement of the robot based on the light-detecting sensors:

III. Result

The output of our endeavour is a two-wheeled robot car that performs corresponding motion on detection of light from either/and left, right and central directions; furthermore, an app on MIT App Inventor was developed, which allows users to switch between the two functionalities of the robot, and provides direction buttons for remote control.

Algorithm 2 Bluetooth/Remote-Control Functionality

```
while there is Bluetooth input available (i.e: Selecting Bluetooth Mode from the MIT App) do
   Read the direction from Bluetooth
   if direction is "Forward": then
      Print "F" and call forward()
   else if direction is "Back": then
      Print "B" and call back()
   else if direction is "Right": then
      Print "R" and call right()
   else if direction is "Left": then
      Print "L" and call left()
   else if direction is "Stop": then
      Print "S" and call stop()
   end if
   Continue reading directions until "Disable" is received from the Bluetooth input
end while=0
```

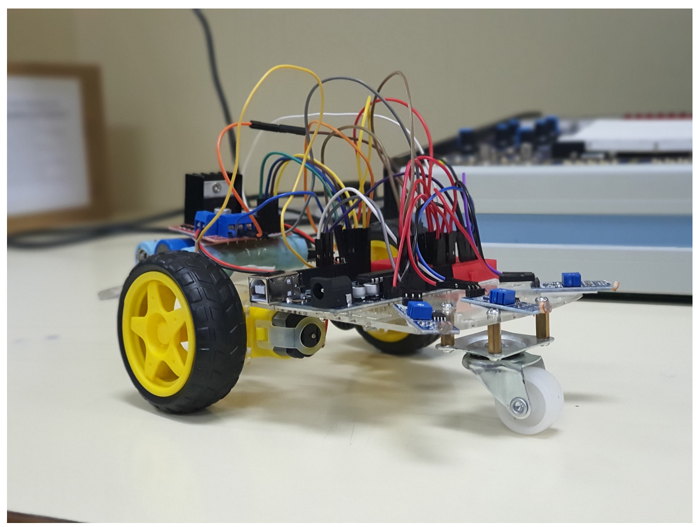

Fig. 1. Multi-Functional Robot Car

IV. Conclusion

Using hardware components, such as micro-controllers, sensors, motors, motor drivers and wires/cables, our team assembled an operational robot car, in which all the components were configured and connected accurately. In addition, the software aspect involved pertinent C code using Arduino IDE as well as logic-building using the MIT App Inventor tool.

References

[1] allaboutcircuits.com/projects/control-a-motor-with-an-arduino

[2] arduinogetstarted.com/tutorials/arduino-bluetooth
[3] maker.pro/arduino/tutorial/how-to-use-an-ldr-sensor-with-arduino