|  |  |
| --- | --- |
| **العربية** | Communication and Electronics Department Microprocessors and Interconnection - EEC 243  Spring 2022 - 2023 |

Software and Hardware Report

|  |  |
| --- | --- |
| **Name** | **ID** |
| Ahmed Osama Mohamed Afifi  Mazen Mohamed Hassanen  Mostafa Mohamed Abdel-Azeem Hassanen  Mohamed Ashraf Elsayed Mahmoud  Ahmed Abdel-Hakem Abdel-Salam Ali | 20010038  20011161  20011950  20011488  20010124 |
|  |  |
| **Department:** | **Communication and Electronics** |

**Group Number:** 44.

**Hardware Project title:**

Zigzag and Circular path Autonomous Moving Robot. (5th Project in the suggestion list).

**Aim of the Hardware Project:**

The aim of creating this autonomous moving robot is to challenge ourselves to design a machine that can autonomously follow a designated path, whether it is zigzag or circular, with high precision by using five sensors and well-organized and uncomplicated code to achieve a reliable and efficient performance. this was complex system as we introduced PID (proportional–integral–derivative controller) to control the motion of the autonomous robot through the feedback to minimize the error and this was a big challenge for us and took much time and effort to achieve the desired behavior.

**List of the used components:**

1. ESP32 Microcontroller.
2. Five IR sensors.
3. Two mini-Dc with gearbox motors.
4. Two Wheels.
5. A Free Wheel.
6. Chassis of the robot.
7. H- Bridge.
8. Four Lithium batteries.

Diagram, schematic

Description automatically generated**Schematic of the circuit implemented:**

**Procedure to use this circuit:**

1. Switch on the robot by switching the Power button. This button is present to control the robot from being on all the time and to keep the batteries from running out.
2. Draw the desired path for the robot to follow by black color.
3. Press on the calibration button to make the robot start calibration mode by rotating around itself and take the values from the IR sensor and calculate the threshold, maximum and minimum value for each sensor.
4. After calibration, Put the robot on the designed path.
5. Press on the start button then wait one second and the robot will start to move on the black designed path.
6. The robot will follow the path and change its direction by the five IR sensors and by the help of the PID feedback to fix the errors in tracking the line and motor speed control.
7. Switch the Power button to stop the robot from moving.

**Budget of the project:**

* **Components budget ( You can click on the component to view it on the store ) :**

1- [ESP32](https://makerselectronics.com/product/esp32-development-board-wifi-and-bluetooth-with-ch340-usb-type-c) = 290 L.E

2- [IR sensors](https://makerselectronics.com/product/infrared-ir-obstacle-avoidance-sensor-3-pin) = 35 \* 5= 175 L.E

3- [Heat shrink](https://makerselectronics.com/product/heat-shrink-5mm-1m) /[Jumpers](https://makerselectronics.com/product/wires-10-cm-male-to-male-pins)/ [Resistors](https://makerselectronics.com/product/carbon-resistor-330%cf%89-0-25w-through-hole)/ [Capacitors](https://makerselectronics.com/product/capacitor-1uf-50v-5x11mm)/ Data lines / Pin headers / Rosetta’s /

Rosetta pluggable / [LEDs](https://makerselectronics.com/product/led-red-8mm) = 80 L.E

4- [Batteries](https://makerselectronics.com/product/18650-li-ion-battery-3-7v-rechargeable-recycled) = 4 \* 25 = 100 L.E

5-[MiniQ Dc with gearbox Motors](https://makerselectronics.com/product/miniq-motor-wheel-set-motor-wheel-6mm-bracket) = 2 \* 150 = 300 L.E

6- [Chassis of the robot](mailto:https://makerselectronics.com/product/2wd-simple-robot-chassis-kit) = 145 L.E

8- Double face = 30 L.E

9- Nuts and Bolts = 30 L.E

10- Map = L.E

11- Components to make PCB ( [Soldering iron](https://makerselectronics.com/product/handskit-soldering-iron-40w-no-112) / [Soldering Wire](https://makerselectronics.com/product/soldering-wire-0-8mm100gm-alloy63-37) / [Solution](https://makerselectronics.com/product/pcb-etching-solution-120ml) / [Board](https://makerselectronics.com/product/pcb-board-fiber-fr2) / [Driller](https://makerselectronics.com/product/driller-for-pcb-boards-wl-500) )

= 500 L.E

* **Workspace budget:**

-fees were around 400 to 500 L.E for the four days distributed on the team.

**Challenges that the team had and how to overcome them:**

1.we encountered some problems regarding the PCB fabrication like:

a) the tracks on the glossy paper didn’t stick very well on the copper PCB so we needed to use another one after cleaning the copper PCB with the dish wire.

b) some tracks were cut or scratched by acid, but we managed to put some tin on them and tested them with the Avo-meter and it functions just fine.

2.we had a little difficulty in figuring out the PID constants value so eventually we got to know them by trial-and-error technique.

3. we were a little bit confused with the behavior of the car as a motor was a bit faster than the other and was not functioning as expected so there were many thoughts as maybe the motor has some kind of technical flaws, so we replaced it by another, and it did not work as expected so we replaced it by another motor and the same happens. Back then we realized that the problem is not from the motor so after some brainstorming, we thought that maybe the jumpers are the problems, so we searched for data lines (known to have less flaws) but it was out of stock from every market however, we replaced the flawed jumpers by well-functioning ones and the same problem still happens.

We continued to brainstorm it even further until an idea brightened in our minds which was what if the problem was in one of the channels of the H bridge but after some testing it turned out to be one of the IR sensor was not functioning as expected and had some technical flaws, we were able to know this from observing the input value to the ESP 32 coming from the IR sensors where the center left sensor was giving wrong number comparing to the other four IR sensors. Eventually after replacing it with another brand new one the car was functioning well.

4. we noticed that the car’s priority was like (first noticed first response) as if somehow the extreme left or extreme right sensors noticed a black line before the center or the center right/ left it will act upon them, so we adjusted the priority of the system through coding to prioritize the center and center right/left then after them comes the extreme left/right.

5. We were thinking, what if after connecting the motors and the robot, and during operation, we discovered that it and the motor rotate in the opposite direction, so it will move backward instead of forward, and in this case we will have to disassemble it and then install it again, but we found a solution to this problem through the code, which is a correction factor that is multiplied by the motor speed. This coefficient is equal to 1 in the case of correct connections and the robot moves forward and is equal to negative 1 in the case of reverse connections and the robot does not move forward.

**References:**

**Code:**

1-mDriver.h:

Graphical user interface, text, application

Description automatically generatedText

Description automatically generated

Text

Description automatically generated

2-mDriver.cpp

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

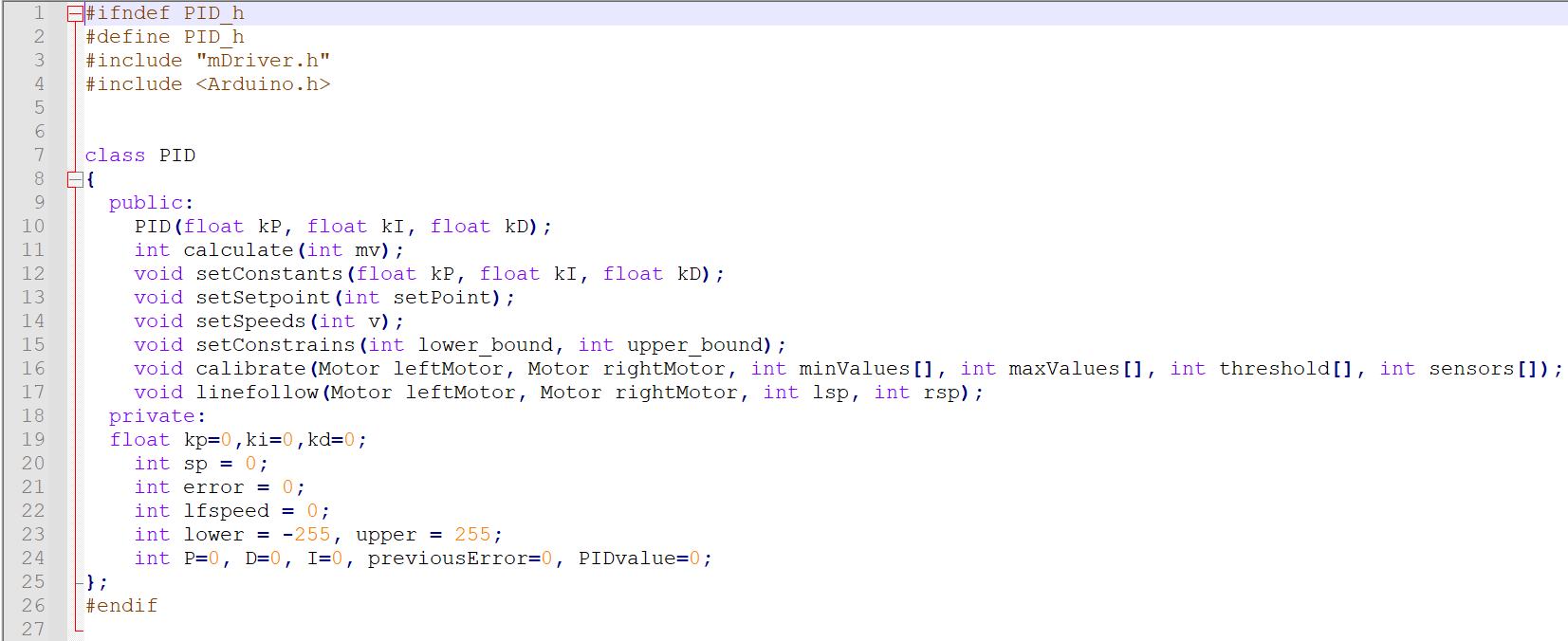
Text

Description automatically generated

Text, letter

Description automatically generated

3-PID.h:



3-PID.m:

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated