Project Report:

Autonomous Robot with Obstacle Avoidance and Auto Parking

Participants:

- Andrew Botros 20201322493
- Eriny Wagih 2022513146
- Sandra Wassim 20221440830
- Seveen Samir 2103117
- Marina George 20221443430
- Carine Emad 20221440878

Dr/ Habrouk

Eng/ Ahmed Eldawy

1. **Problem Description:**

Obstacle Avoidance:

The objective of this project is to design and implement an autonomous robot capable of navigating its environment, avoiding obstacles, and autonomously parking itself. The primary challenge lies in developing a system that allows the robot to detect obstacles in its path and navigate around them effectively.

Auto Parking

The robot should be able to identify a parking space and autonomously maneuver itself into the parking spot. This requires precise sensing and maneuvering capabilities to ensure safe and efficient parking.

Challenges Associated:

While the concept of autonomous robots with obstacle avoidance and auto parking capabilities holds significant promise, several challenges must be addressed for successful implementation:

- **1- Sensor Limitations:** The effectiveness of obstacle avoidance and auto parking algorithms heavily relies on the accuracy and reliability of sensors. Common sensors such as ultrasonic sensors may have limitations in detecting certain types of obstacles or accurately measuring distances in varying environmental conditions.
- **2- Environmental Factors:** Environmental factors such as lighting conditions and surface textures can impact sensor performance and navigation accuracy. Developing robust algorithms that can adapt to these environmental variables is essential for ensuring consistent performance in real-world scenarios.
- **3- Complexity of Parking Scenarios:** Parking environments can be highly diverse and complex. Autonomous robots must be capable of recognizing and maneuvering within various parking scenarios, including parallel parking, perpendicular parking, and angled parking, while considering factors such as space availability, clearance requirements, and vehicle dimensions.
- **4- Integration with Existing Infrastructure:** Implementing autonomous robots in real-world environments requires seamless integration with existing infrastructure and regulations. This includes compatibility with parking management systems, adherence to traffic laws and safety regulations, and consideration of pedestrian and vehicular traffic patterns.

2. System Design:

Hardware Components:

The robot comprises various hardware components, including:

1) **Body of the Car:**
    - The body of the car serves as the structural framework to house and protect the internal components of the robot.
    - It provides mounting points for attaching sensors, motors, and other hardware components.
2) **Mega Arduino:**
    - The Mega Arduino serves as the central processing unit of the robot, coordinating the operation of all components.
    - It runs the control algorithms for obstacle avoidance and auto parking, receiving input from sensors and sending commands to motors.
3) **DC Motor:**
    - DC motors are used to drive the wheels of the robot, enabling forward and backward movement as well as steering.
    - They receive commands from the microcontroller to adjust speed and direction based on the desired navigation or parking behavior.
    - DC motors should be selected based on torque requirements, power efficiency, and compatibility with the robot's chassis and wheel configuration.
4) **Servo Motor:**
    - Servo motors are typically used for precise control of steering mechanisms or other moving parts requiring angular positioning.
    - In the context of the autonomous robot, a servo motor may be employed for tasks such as controlling the orientation of sensors or adjusting the angle of a sensor mount.

Interaction:

The microcontroller interacts with the sensors to gather environmental data, processes this information using algorithms, and commands the motor drivers to execute appropriate actions.

3. Solution Implementation:

Obstacle Avoidance:

The robot employs ultrasonic sensors for obstacle detection. These sensors emit ultrasonic waves and measure the time taken for the waves to reflect back from obstacles, thereby calculating distances. The obstacle avoidance algorithm utilizes this distance data to navigate around obstacles. It follows a simple yet effective approach:

1) **Scan Environment:** The robot continuously scans its surroundings using ultrasonic sensors.
2) **Detect Obstacles:** If an obstacle is detected within a predefined range, the robot determines its position and size.
3) **Calculate Path:** Based on obstacle position and size, the robot calculates an alternative path to avoid collision.
4) **Adjust Movement:** The robot adjusts its direction or speed to navigate around the obstacle, ensuring safe traversal.

Auto Parking:

The robot employs a combination of distance sensors and line-following techniques for parking space detection and maneuvering. The algorithm for auto parking involves the following steps:

1) **Search for Parking Space:** The robot scans its surroundings using ultrasonic sensors to identify vacant parking spaces. It measures distances to nearby objects to determine potential parking spots.
2) **Evaluate Parking Space:** It assesses the size and accessibility of detected parking spaces based on the distances measured. The robot analyzes whether the space is large enough and has suitable clearance for parking.
3) **Approach Parking Space:** Once a suitable space is found, the robot maneuvers towards it while following a predefined path. It adjusts its direction and speed to navigate towards the parking spot while avoiding any obstacles in the path.
4) **Align with Parking Spot:** As the robot approaches the parking spot, it uses line-following techniques to align itself with the boundaries of the parking spot. This ensures accurate positioning for parking.
5) **Execute Parking Maneuver:** Upon reaching the parking spot, the robot precisely maneuvers into the parking spot using feedback from the distance sensors. It adjusts its position as necessary to achieve optimal alignment within the parking space.

4. Cost Analysis:
The overall cost of the robot solution includes expenses for components, sensors, and additional materials. The cost breakdown is as follows:

Elements we use	Number	Costs
Robot chassis 4WD	pair	135
Yellow wheels	4	4*25=100
Ultrasonic distance sensor	3	3*45=135
Ultrasonic holder	4	4*15=60
DC gear motor	2	2*40=80
total	**510**	

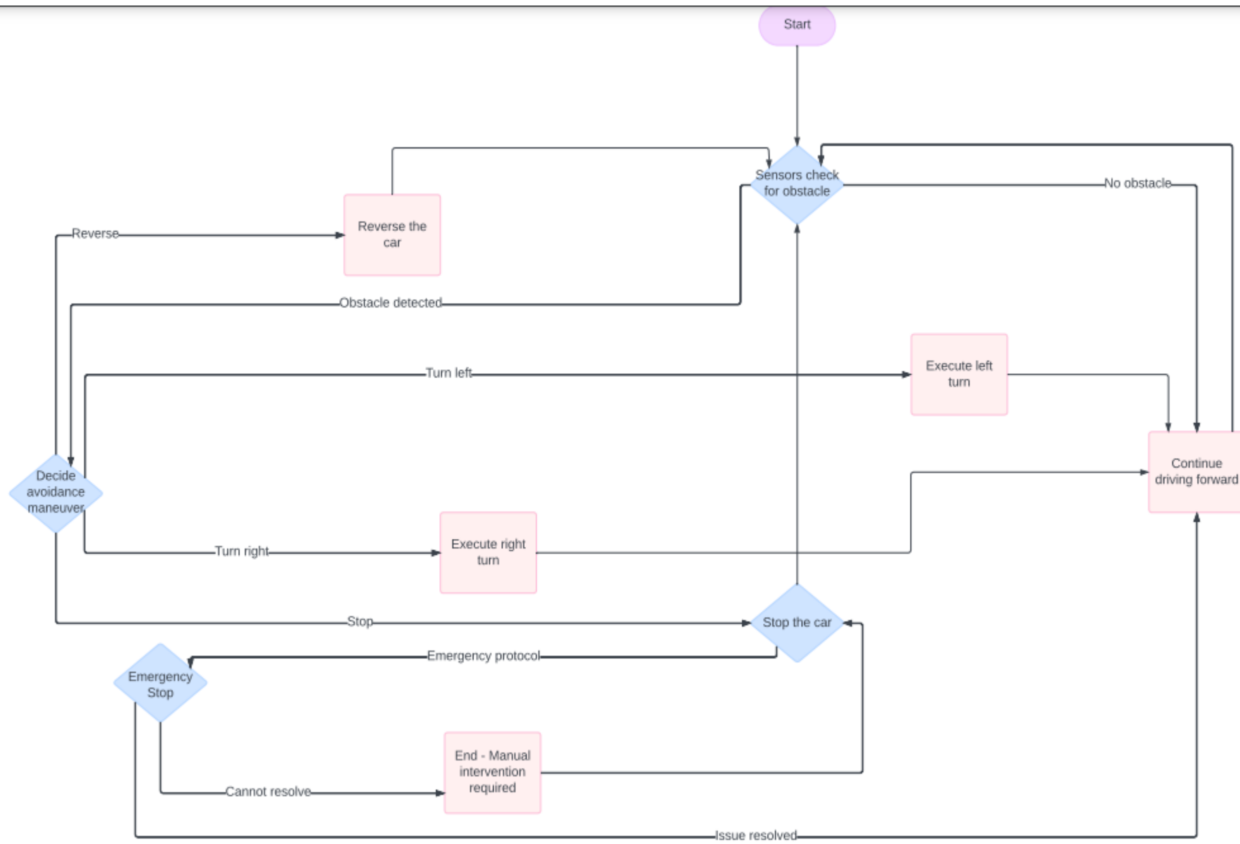
The total cost of the robot is **510**EGP.

Cost of the rest of components we used like:
Arduino: Arduino uno cost 465 but we had Arduino mega that actually cost 726, it seems to be more expensive but we were already had it so it cost us 0.
bread board: costs 40 but it costs us 0.
Wires: the set of 40 pin wires costs 40 but it also costs us 0.
servo motor: cost 110, but costs us 0.
Screws: the one of it costs 1 or 2 but all we use (4) costs us 0 as we had it before so we used it in order to not to buy again and cost us much.
Shipping & delivery: costs us 0 as part of us go themselves buy the components.
Software & internet: 0.
Overhead costs: like electricity, workspace and etc. costs 0.

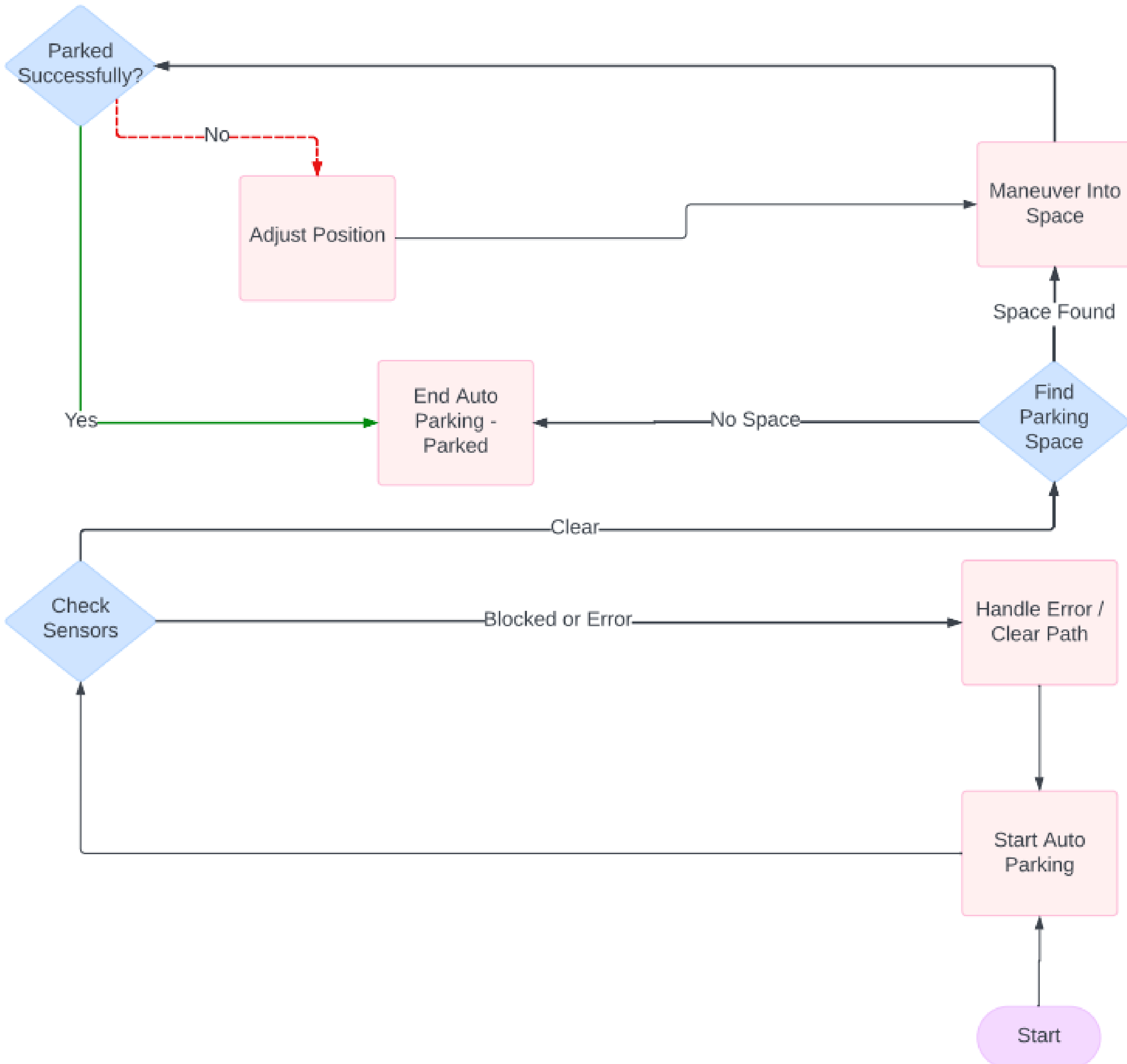
So we can say that **we saved** almost **659**EGP besides delivery, software and overhead costs.

5. Detailed Flowchart:

Obstacle avoidance:



Auto parking



Conclusion:

The autonomous robot with obstacle avoidance and auto parking capabilities offers a practical solution for navigating complex environments and parking scenarios. Through careful design and implementation of hardware components, sensors, and algorithms, the robot demonstrates effective performance while ensuring cost-effectiveness and practicality. Further refinement and optimization can enhance its functionality and applicability in real-world scenarios.