10/20/2022

Human Following Trolley

Final Report

CODE ERROR TEAM
SHEHAN, RANEESHA, KAVINDU, RAQEEB AND NANDITHA.

Acknowledgement

Our thanks to Miss Irushi Our supervisor are beyond words. Thanks, you for your patience, feedbacks and advice, ma'am. Additionally, without the knowledge and experience that our friends so kindly shared, we would not have been able to conduct this literature review. Furthermore, this project would not have been possible without the kind assistance of my teammates.

Additionally, we appreciate the assistance with editing, late-night critique sessions, and moral support from our fellow classmates, especially our seniors. We should also express our gratitude to the university's librarians and research participants, who had an impact on and motivated us.

The last thing I should mention is our families, especially our parents. Throughout this process, their confidence in us has kept our spirits and motivation strong.

Contents

Acknowledgement 1

Introduction. 3

Chapter 1: Synopsis. 4

1.1) Background of the project 4

1.2) Problem Identification 5

1.3) Aim and objectives 5

Aim 5

Objective 5

Chapter 2: Literature Review. 6

2.1) Components 6

2.2) Comparisons with other similar projects. 10

Chapter 3: Methodology. 12

3.1) What is Methodology? 12

3.2) Conceptual Designs 12

3.2.1) Conceptual Design 01 12

3.2.2) Conceptual Design 02 14

3.2.3) Conceptual Design 03 15

3.3) Optimum Solution 16

Chapter 4: Design and Implementation. 17

4.1) Design and Implementation 17

4.1.1) Components Used 17

4.1.2) Specifications 23

4.2) Software Simulation 26

4.3) Hardware Simulation 38

Chapter 5: Testing and Analysis. 45

5.1) Testing 45

5.2) Analysis 45

Chapter 6: Conclusion 46

6.1) Limitations 46

6.2) Future Improvements 46

Introduction.

Robotics is an interdisciplinary branch of engineering and science that includes mechanical engineering, electronic engineering, information engineering, computer science, and others. Robotics involves design, construction, operation, and use of robots, as well as computer systems for their perception, control, sensory feedback, and information processing. The goal of robotics is to design intelligent machines that can help and assist humans in their day-to-day lives and keep everyone safe (Wrike, 2006-2022). Robotics develops machines that can substitute for humans and replicate human actions. Robots can be used in many situations and for lots of purposes, but today many are used in dangerous environments including inspection of radioactive materials, bomb detection and deactivation, manufacturing processes, or where humans cannot survive e.g., in space, under water, in high heat, and clean up and containment of hazardous materials and radiation. Robotic technology has increased appreciably in past couple of years. Such innovations were only a dream for some people a couple of years back. But in this rapid moving world, now there is a need of robot such as "A Human Following Robot" that can interact and co-exist with them. To perform this task accurately, robot needs a mechanism that enables it to visualize the person and act accordingly. The robot must be intelligent enough to follow a person in the crowded areas, vivid environment and in indoors and outdoors places (Ng, 2015). The image processing carried out to get the information about the surroundings visually are a very important thing. The following points should be carefully noted while doing the processing. The luminosity conditions should be very stable and should not fluctuate. The ranges should be set properly for the desired environment on which to perform the tracking. The target should not be very far from the visual sensor as the distance matters a lot. We should avoid the use of such colors around the robot that matches with that of the target. Otherwise, the robot would get confused. Typically, human following robots are equipped with several different diverse combination of sensors i.e., light detection and ranging sensor, radio frequency identification module (RFID), laser ranger finder (LFR), infrared (IR) sensing modules, thermal imaging sensors, camera, wireless transmitter/receiver etc. for recognition and locating the target. All the sensors and modules work in unison to detect and follow the target. (Scribd, Copyright © 2022)

Chapter 1: Synopsis.

1.1) Background of the project

Robotics technology has advanced tremendously in recent years. The majority of conventional robots are still often employed in industrial settings, such as companies that assemble cars (Božek, 2014). Meanwhile, applications for intelligent robots in daily life have grown in popularity. The care of the elderly is now provided by robots that are friendly to humans (Leroux, 2013) (Bouakaz, 2014) (Wu, 2013). A robot that follows humans is designed to strengthen the bond between people and the machine (Kazuyuki. M., 2004)

For instance, at hospitals, airports, and retail malls, the robot may transport big loads for people. In a variety of circumstances, the robot can serve as a helper for humans. The trolley or cart is often self-served by the customer in a market or shopping mall. Some of us find this scenario less convenient since we have to manage looking for food and watching for kids at the same time. Additionally, because the cart would be heavier to push with more goods added, this will be a difficulty for elderly and pregnant women. This study made a suggestion for an intelligent following sensor shopping cart to enable people shop efficiently at markets or shopping centres without having to push a cart around.

At the entrance of the supermarket or shopping complex, the consumer may request an automatic shopping cart, which will remain with them all the way to the checkout counter and will be additionally charged a small amount for the service provided by the trolley. The shopping cart is going to follow that one particular shopper and transport their items. With the aid of ultrasonic sensor and IR sensor, this will ensure that there are no obstructions, and will always remain within a range of 1 meter from the consumer at all times. for further precision the customer will get a RFID tag unique to the trolley so that the tracking accuracy will increase.

As for the ultrasonic sensor, it tracks the customers movement so that if the customer passes the 1-meter range the robot will accelerate to catch up and will slow down or go back if the range is less than 1 meter. RFID is an identifying system that uses wireless frequency and tiny chips to process and send information in an environment without physical touch. At the moment, mapping, navigation, and localization have seen significant advancements because to RFID technology. This is because each RFID tag's identifier is unique. Therefore, this technology is reliable to our robot as its accurate as well.

1.2) Problem Identification

Over 60% of persons over 65 have a physical impairment that prevents them from participating in activities of daily living. Shopping is a vital life activity that older folks may find problematic. Shopping is described as an activity associated to inspecting or purchasing goods.

One of the most prevalent self-reported daily activities issues among older adults is difficulty shopping, which rises with age, physical disability, and the number of coexisting disorders. A series of intricate and physically taxing duties, including social and physical interactions in the neighbourhood linked to leaving the house and traveling to the store, are involved in shopping for food and other requirements.

The capacity to shop alone is likely to be hampered by environmental obstacles outside the home and on the way to the store. Disability models place a strong emphasis on the influence of environmental variables in creating disability.

Elderly and Disable people find it difficult to carry more goods from various stores in a mall. So, it's more likely to them to ask for help. We're proposing this device to overcome this problem.

1.3) Aim and objectives

Aim

This project is to develop a prototype of an autonomous human following trolley. Our main aim is to help the people (mostly elderly and disabled) to carry the goods easily and to save their time.

Objective

This project come up with a robust aimed human following shopping trolley, here firstly the hardware system is implemented to show how the final product of the project looks like. Next, the algorithm should be designed to input the algorithm to the project. One of our main objectives is to develop the human following trolley which can carry goods to follow the user by maintaining the inputted distance by avoiding the obstacles. Finally, using the RFID Tag (Radio Frequency Identification), where the trolley can identify the authorized customer with the relevant tag. (Ng, 2015)

Chapter 2: Literature Review.

2.1) Components

To carry on this project, we use the following components;

Arduino Mega

RFID module (Radio Frequency Identification Device)

Ultrasonic sensor

IR sensor

Motor Driver

Gear motor

Robot wheel

Li-ion Battery

Servo Motor

Battery holder

Jumper wires

Switch

Dot board

Arduino Mega

The microcontroller board known as the Arduino Mega 2560 is based on the ATmega2560. It features a 16 MHz crystal oscillator, 54 digital input/output pins (of which 15 may be used as PWM outputs), 16 analog inputs, 4 hardware serial ports (UARTs), a USB connector, a power jack, an ICSP header, and a reset button.

This has an Operating Voltage of 5V and the Limited Input Voltage is 6V - 20V but the recommended Input Voltage is 7V – 12V. DC Current per I/O pin is 20mA and DC Current for 3.3V pin is 50mA. And Arduino mega 2560 contains a Flash memory of 256kB of which 8kB used by bootloader, SROM of 8kB and EEPROM of 4kB.

It comes with everything required to support the microcontroller; to get started, just use a USB cable to connect it to a computer, or an AC-to-DC converter or battery to power it. The majority of shields created for the Uno and previous boards, Duemilanove or Diecimila, are compatible with the Mega 2560 board (Arduino, 2021).

RFID module (Radio Frequency Identification Device).

RFID (Radio Frequency Identification) is a wireless technology that consists of two components: tags and readers. The reader is a device with one or more ports that broadcast radio waves and receive signals from RFID tags. There are various kinds of RFIDs which are being used in the modern world, out of them, RC-522, PN-532 are the most similar RFIDs for a human following trolley.

RC-522-This is a circuit with highly integrated modulation and demodulation (which the circuit can separately identify signals). To link the output driver stage to the antenna, a minimal number of external devices can be used. The power supply needed is 2.3v to 3.3v. The maximum reader distance is 50mm where it depends on the length of the antenna. It consists of 3 power saving modes such as hardware power down, software power down and transmitter power down. It has a working temperature of -30 to +85 with an inbuilt temperature sensor to stop the transmission of RF when the chip is too heated. (Nicho, 2020)

PN-532- This is most used and popular RFID for NFC in the market, and this circuit with highly integrated analog circuit and demodulation, where it is used in every high tec mobile phones. This model is capable of communicating with phones (NFC) and also read and write to tags and cards. It is capable of communicating directly between two mobile phones wirelessly in short range. As the RC-522, this is also capable of reading up to 50mm.Where the temperature operating on the RFID is also similar to RC-522. Where the power supply needed is 2.7v to 5.5v. (Nicho, 2020)

Out of the above 2 RFID models (RC-522 and PN-532), RC-522 is a high frequency radio chip compared to PN-532 which supports ISO14443A (an international wireless connection and security standard) protocol. Also, the RC-522 consists of inbuilt temperature to detect high heat in the chip which is also one of more reliable reasons that RC-522 is a good selection. When considering power consumption, the RC-522 uses low power than the PN -532. The RC-522 is more affordable, which means it is more cost effective than the PN-532 chip. And finally, the RC-522 is easy to use.

So, according to above comparison we have selected the RC-522 RFID to use in our project tag.

Ultra-Sonic Sensor

An ultrasonic sensor is typically employed as a proximity sensor, but it may also be utilized as a level sensor. Because of the qualities of an ultrasonic sensor and ultrasonic sound, it is feasible to detect objects and measure distances with the same sensor. Also, ultrasonic sensor can replace the function of a traditional proximity switch. Ultrasonic sensors are now employed in every technological industry in the globe. (Danny, 2019)

HC-SR04-The HC-SR04 ultrasonic sensor measures the distance to an object using sonar. This sensor has a reading range of 2cm to 400cm (0.8inch to 157inch) with an accuracy of 0.3cm (0.1inches). The main specifications for this sensor are 5V operating voltage, 15mA operating current ,40KHz operating frequency, 3mm precision and a dimension of 45x20x15mm. Two ultrasonic transducers make up the sensor. The transmitter sends out ultrasonic sound pulses, while the receiver listens for reflected waves. It is essentially a SONAR that is used in submarines to identify underwater objects. (Dejan, 2015)

US-100- This sensor also measures the distance of an object using sonar, which this sensor has all the similar features of the HC-SR04, expect a precision of 1mm also the trig signal can be from 3V to 5V. This sensor is not a huge user-friendly sensor, also the cost of this sensor is little bit high. (Akbari, 2019)

When comparing the HC-SR04 and US-100, the HC-SR04 gives the best accurate results, also the HC-SR04 is easy to use and the speed of the sensor is high than the US-100. This sensor is a cheap sensor, where this is the most popular ultra-sonic sensor in the market.

IR sensor

There are many uses for infrared technology in both industry and daily life. The key benefits of IR sensors are their low power requirements, straightforward construction, and practical functionality. IR signals cannot be seen by the human eye. IR radiation may be found in the visible and microwave portions of the electromagnetic spectrum. These waves generally have wavelengths between 0.7 and 1000 m. The three areas of the IR spectrum are near-infrared, mid-infrared, and far-infrared. The wavelength spans the near-infrared range (0.75 to 3 m), the mid-infrared range (3 to 6 m), and the far-infrared area (greater than 6 m).

A device known as an infrared sensor produces an electric signal after detecting infrared radiation from its surroundings. Both movement and object heat may be measured using an infrared sensor.

Although infrared radiation is invisible to the human eye, it may be detected by the infrared sensor. A photodiode that can detect infrared light is referred to as an infrared sensor. The resistances and output voltages alter in according to the intensity of the IR light as it strikes the photodiode.

Motor Driver

A motor driver IC is an integrated circuit chip that controls motors in autonomous robotics and embedded electronics. A motor driver is unquestionably anything that causes the motor to move in accordance with the provided instructions or inputs (high and low). It receives low voltage from the controller/processor and controls a real motor that requires high input voltage. In layman's terms, a motor driver IC directs the motor depending on orders or instructions received from the controller. (Anon., 2021)

L293D- It is a four-channel monolithic integrated high voltage, high current driver that accepts normal DTL or TTL logic levels and drives inductive loads (such as relays, solenoids, DC and stepping motors) and switching power transistors. The main features are, it is possible to utilize the same IC to power two DC motors, Controlling the speed and direction is possible, Vcc2 (Vs) motor voltage: 4.5V to 36V, Peak motor current: 1.2A, Continuous motor current: 600mA, there is an option for automatic thermal shutdown and also there are 16-pin DIP, TSSOP, and SOIC packages available. (Anon., 2021)

ULN2003- It is a high-voltage, high-current Darlington array featuring seven open collector Darlington pairs and seven common emitters. Each channel is rated at 500 mA and has a peak current rating of 600 mA. For inductive load driving, suppression diodes are added, and the inputs are pinned opposite the outputs to simplify board layout. The main features are, contains seven high-voltage and high-current Darlington pairs, each rated for 50V and

500mA, with input pins that may be activated by +5V. Can be controlled directly by logic devices such as Digital Gates, Arduino, PIC, and so on. Also, there are 16-pin DIP, TSSOP, and SOIC packages available. (Anon., 2021)

The L293D application are used to drive high current motors using digitals circuits, used to drive stepper motors, used to drive high current LEDs, latching relay is possible due these properties considering to ULN2003, and also it is the most used motor driver and the it is cheaper than the ULN2003. So, we have chosen the L293D motor driver due to the above reasons.

Gear motor

A gear motor combines a motor and gearbox into one unit. When a gear head is added to a motor, the speed is decreased but the production of torque is increased. In terms of gear motors, speed (rpm), torque (lb-in), and efficiency (%) are the most crucial variables. You must first calculate the load, speed, and torque requirements for your application in order to choose the best gear motor. To meet all application needs, ISL Products provides a range of Spur Gear Motors, Planetary Gear Motors, and Worm Gear Motors. The majority of our DC motors can be enhanced by one of our distinctive gearheads to give you a highly effective gear motor solution.

Robot wheel

The Dual Axis Yellow Gear Motor pairs well with this Robot Plastic Tire Wheel.

Li-ion battery

Two battery packs of 9V and 6V and 9V battery for the Arduino and all other components and a 6V battery for the motors are used to power the robot. The sensor that identifies lines is this one.

Servo Motor

For the above mention project, we use SG90 servo motor with high output power while being small and light. Servo rotates around 180 degrees (90 in each direction) and functions similarly to larger types of servos. These servos may be controlled by any servo code, hardware, or library. Utilized to move objects without the need to construct a motor controller with feedback and a gear box, especially because it can fit in tight spaces. Hardware and three horns (arms) are included. (EE, 2020)

2.2) Comparisons with other similar projects.

There are many various ways to create a robot that follows a person, but the three methods utilized most frequently by designers are as follows:

Human following robot using LRS (Laser range scanner)

Human following robot using Ultrasonic sensor

Human following robot using LIDAR sensor

Human following robot using Image processing technology

Thus, in the above mention design is based on using Ultrasonic sensor method so in the below paragraphs are used to make a comparison between them.

Human following robot using LRS.

This robot follows the target person by detecting the location of the target person's shins using data from an LRS (laser range scanner). The constructed wheel type robot, Laser Range Scanner, Web camera, and control PC make up this human-following robot. Only the robot scenario is captured using the web camera. The ROS (Robot Operation System) is used by this robot to implement the system. For the purpose of detecting a human, the LRS is attached to the robot (Hokuyo, 2020).

This method has some advantages and disadvantages when compared to the Ultrasonic method.

LRS TOF10120	**Ultrasonic HC-SR04**
Higher cost ($10 USD)	Low Cost ($1.25 USD)
Sealed to environment	Exposed to environment
Serial connection & 12C Bus	1-pin or 2-pin digital I/O
Very small size	Larger size
Quick response	Slow response
Narrow field of view	Wider field of view
10cm – 180cm range	2cm – 400cm range
3 – 5 VDC	5 VDC
35mA working current	15mA working current
Much complex coding	Much simple coding

(Workshop, 2020)

From above table we can conclude that Ultrasonic is more efficient and win the comparison.

Human Following Robot Using Image Processing.

This is related to human-robot interaction via image processing. The goal of this project is to support and track a certain individual. A human tracking algorithm will be created to guide the human following robot in tracking a specific individual. A camera captures images that are then processed by the Raspberry Pi, which controls the robot to follow the target individual. Ultrasonic sensors, on the other hand, are employed for safe distance tracking, obstacle avoidance, and precise tracking. The Raspberry Pi is utilized as a platform for analyzing data acquired from sensors and then executing commands to control the movement of the human-following robot. The robot was designed to track the target individual accurately and avoid obstacles throughout the following period. (K.GOH, 2016)

When comparing the image processing, it has used Raspberry Pi and the coding is more complicated as the feedback code of the image changes every time as the individual changes.

Human following Sensor using Lidar

LiDAR sensors are a type of laser distance sensor that measures the range, or depth from a surface. They work by emitting pulses in all directions and measuring how long it takes for them to bounce back off targets. The working principle of LiDAR sensors is similar to that of ultrasonic sensors. The only difference is the frequency in which they operate, while LiDAR's use a laser beam instead of sound waves for measuring distance and analysing objects with laser beams generated from an array or cluster.

The limitations of using LiDAR include a higher cost as compared to ultrasonic and IR sensors. It is also harmful for the naked eye- high end LiDAR devices may use stronger pulses that could affect human eyes, which means it must have a safety guard installed on top of each sensor in order not be damaged by sunlight exposure or bright light reflections off water surfaces. Narrow point detection can miss some objects like glass and items in close proximity to the floor. Ultrasonic sensors are a cost-effective solution for detecting, counting, and identifying objects without requiring physical contact. They can be used in environments with high levels of dust or humidity and perform better than infrared sensors when there is a lot of smoke.

So, it is better to use ultrasonic sensor than the LiDAR for the human following trolley.

Chapter 3: Methodology.

3.1) What is Methodology?

A methodology is essentially a grouping of several approaches, rules, procedures, practices, and methods. Project methods are detailed, exacting, and often include a list of actions and procedures at each stage of the project's life cycle. Methodology is either the act of doing something or the branch of logic that studies and evaluates reasoning. Methodology is a way for carrying out an experiment, to put it simply. We'll talk about three approaches in this part that helped us reach our goal. The most ideal solution will then be chosen following the comparison. (manegment, 2017)

3.2) Conceptual Designs

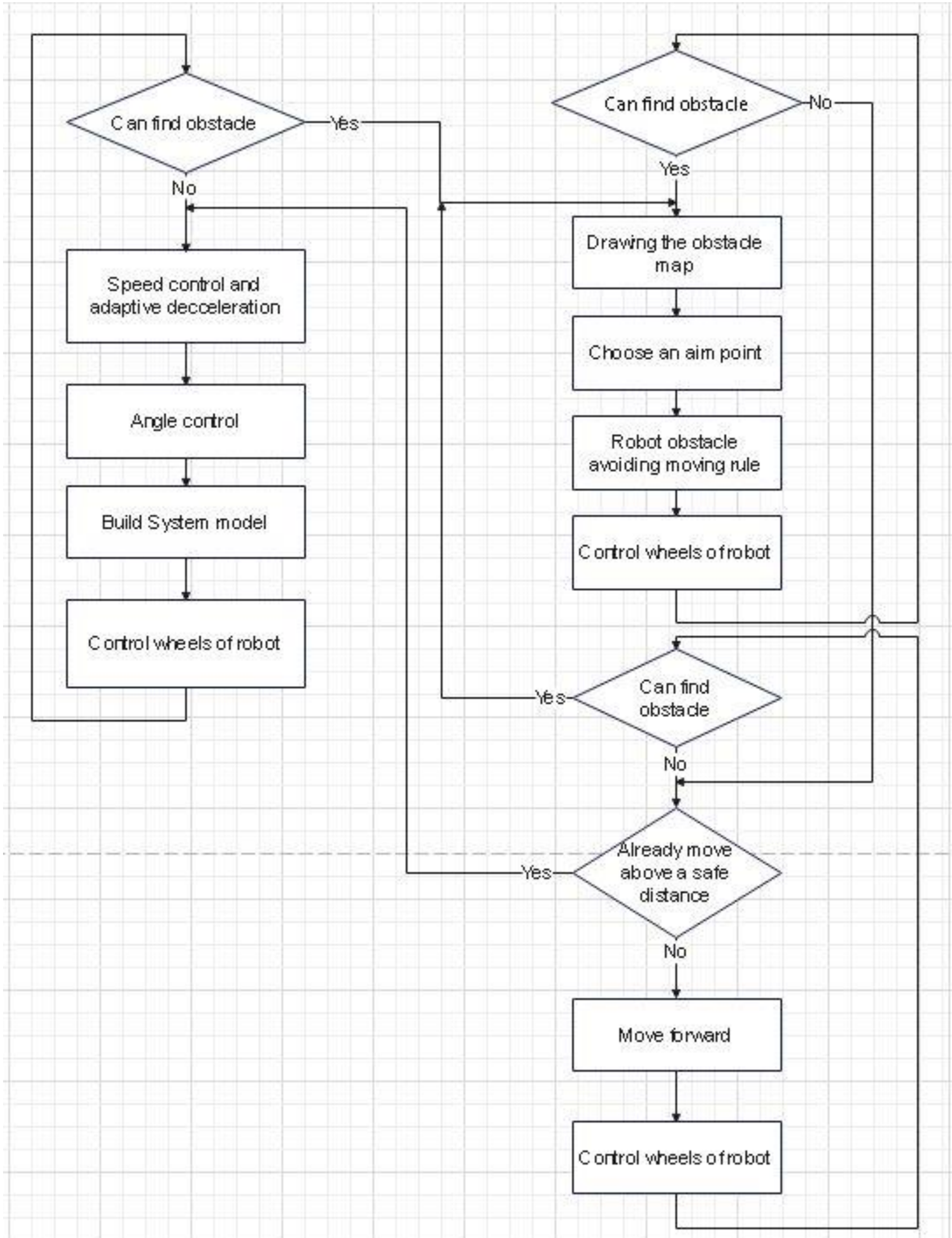
1) Conceptual Design 01: Human Detecting and Following Mobile Robot Using a Laser Range Sensor.
2) Conceptual Design 02: An autonomous human-following robot using RGB-D camera and 2D LIDAR.
3) Conceptual Design 03: Human Detecting and Following Robot Using RFID and Ultrasonic Sensor.

3.2.1) Conceptual Design 01

Many research teams have seen robots that can detect people. Some robots under study employ an LR sensor of this kind for detection. When the laser sensor is in operation, the target is hit by laser pulses from the laser emitting diode. The laser light is dispersed throughout the object after being reflected by it. After being picked up by the optical system and returning to the sensor receiver, some of the scattered light is imaged on the avalanche photodiode. The avalanche photodiode is an optical sensor that also has an inbuilt amplification function. As a result, it has the ability to detect very weak optical impulses and transform them into matching electrical signals. The laser ranging sensor is a frequently used laser sensor that measures the target distance by logging and analyzing the interval between the light pulse's transmission and reception. The laser sensor must measure the transmission time very precisely due to the very rapid speed of light. (Researchgate, 2014)

But networked laser range scanner should set up in a smart environment. However, the majority of these studies only focus on the challenges of detecting and/or following humans, without describing the constrained circumstances in which the robot must operate, such as the response time and restricted velocity. These investigations were created with the assumption that either the target human will move slowly enough or that the robot speed is infinite. (Researchgate, 2014)

On a mobile robot with two wheels, with a human-following control system. The robot has top-mounted laser range scanners that are positioned 108 cm above the ground to locate the target's waist. Based on this configuration, a control system was created, as seen in Fig. 1, that consists of two components: a human detection function and a human following algorithm that incorporates a human following function and an obstacle-avoiding function. (Researchgate, 2014)

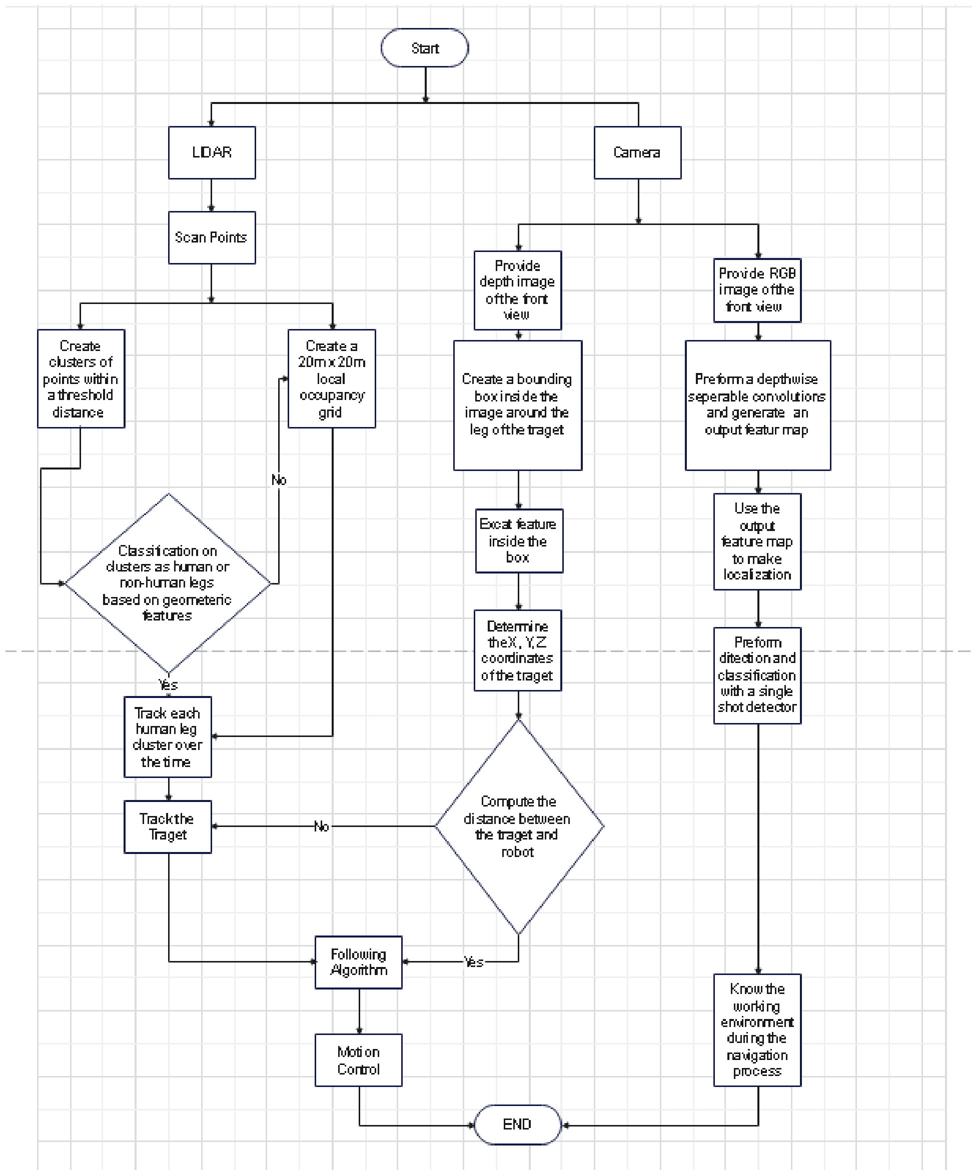


3.2.2) Conceptual Design 02

Using a person-following framework and a combination of an RGB-D camera and LIDAR, we want to create a completely autonomous robot that can operate both inside and outside. The suggested framework employs feature-point extraction to determine the target location using the camera and LIDAR cluster creation and classification for detection and tracking. With a little amount of deep learning, our system can execute human-following tasks in a variety of lighting settings (day and night). Additionally, it can function effectively in both indoor and outdoor settings. Third, the suggested method attempts to retrieve the target when the human leaves the robot's field of vision and returns while avoiding freezing. The following figure provides a description of the entire system.

The scan points received from the LiDAR are utilized to build local grid maps as well as clusters (of the environment). The map is used to improve tracking by seeing where monitored persons are located and to help with data association. Every iteration, the local map is updated using each cluster that isn't linked to a monitored person, or, in other words, isn't a human leg.

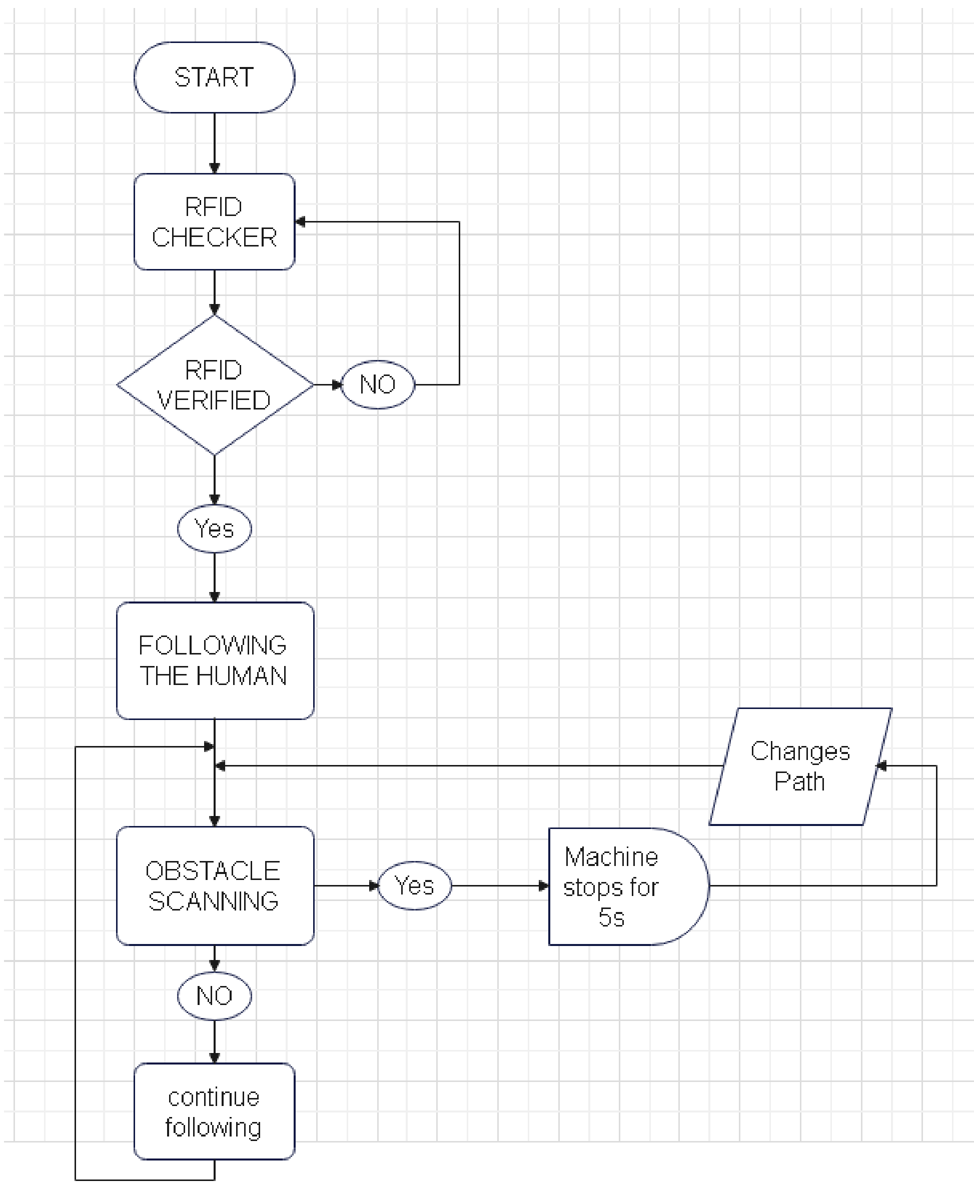
The camera provides a visual track of the surroundings, and the closest human standing in front of the robot is intended to be the objective to follow. The target individual is tracked using the established real-world coordinates, the robot's proximity to the target, and the leg cluster tracking data from the LiDAR. The object recognition and classification work done with the camera also aims to provide the robot a better understanding of its working environment during navigation.



3.2.3) Conceptual Design 03

In order to track people, this idea primarily makes use of ultrasonic and infrared sensors. Technology for radio frequency identification is used for authentication. The RFID procedure acts as the robot's control key. Those with a valid RFID tag are the only ones who can operate the robot.

The robot's key is the RFID tag. Technically, the robot's ability to follow the human is enabled by the unique identifying number of the RFID tag. To choose whether to run the code or not, the trolley should identify a genuine RFID tag, the code commands the robot to follow the individual using ultrasonic and infrared sensors.



After the tag is discovered, the programming to track the human starts to run. The ultrasonic sensor first determines whether the target being tracked is real. The infrared and ultrasonic sensors work in tandem throughout the operation. They use infrared and ultrasonic technology to detect people, then send the information to the Arduino. The motor driver receives information from the Arduino, which serves as a CPU.

This project's programming component is essential to completing all of the operations. The RFID reader in the programme looks for the right RFID tag and blocks access to any other tags.

If the correct tag is found access is authorized and it enters into the human following algorithm using ultrasonic sensors and at the same time the robot will also follow the obstacle avoiding algorithm.

3.3) Optimum Solution

When comparing above 3 conceptual designs the components used and the technology used are different from each other.

Comparison between Conceptual design 1 and Conceptual design 3

Conceptual design 1- uses Laser Ranging sensors to detect humans, here, the target is hit by the laser pulses and then reflected back where next the optical sensor identifies the receiver signals. This laser has an ability to detect even weak pulses. In this module the network laser radar scanner should set up in a smart environment.

Conceptual design 3- uses Ultra Sonic sensors, infrared sensors and a rfid tag. Here, the ultrasonic sensor uses to manage the distance between the trolley and the human by reflecting back the sound it releases. Then, the IR sensors keep the trolley in the desired path of the customer. This whole process starts when the Rfid tag identifies whether a permanent customer is selected and identified by the Rfid.

Although the conceptual design 1 uses laser ranging sensors, the sensors take a bit a delay to get the reflection of the laser back and also the costs of the laser are too high. Also, the scanner has to set up a smart environment. But the conceptual design 3, the ultrasonic sensor can detect distance between human and trolley than the LR sensors, this module uses IR sensors to keep the trolley in the path. And also, the design 3 uses a RFID tag, where the trolley can be unlocked only if the card it activated to a specific person, which at no point the trolley can be stolen.

Comparison between Conceptual design 2 and Conceptual design 3

Conceptual design 2- This module uses LIDAR and a combination of an RGB-D camera. Here the human is scanned by the lidar and the image of the human is shown by the camera. LIDAR sensor can scan 270 degrees, can react fast because this uses a light beam instead of sound ray. Lidar sensors detection range is from few meters to a maximum of 200m.

Conceptual design 3- uses Ultra Sonic sensors, infrared sensors and a rfid tag. Here, the ultrasonic sensor uses to manage the distance between the trolley and the human by reflecting back the sound it releases. Then, the IR sensors keep the trolley in the desired path of the customer. This whole process starts when the Rfid tag identifies whether a permanent customer is selected and identified by the Rfid.

Here, the conceptual design 2 uses LIDAR and a combination of an RGB-D which is far more expensive than the materials used in the conceptual design 3 and LIDAR has difficulty detecting objects at closer distance. But the ultrasonic sensor can detect closer distance objects. And in conceptual design 3 the ultrasonic sensor is used to detect the distance between human and trolley as LIDAR sensors are enable to detect closer objects, this module uses IR sensors to keep the trolley in the path. And also, the design 3 uses a RFID tag, where the trolley can be unlocked only if the card it activated to a specific person, which at no point the trolley can be stolen. So, therefore conceptual design 3 is better than conceptual design 2.

Chapter 4: Design and Implementation.

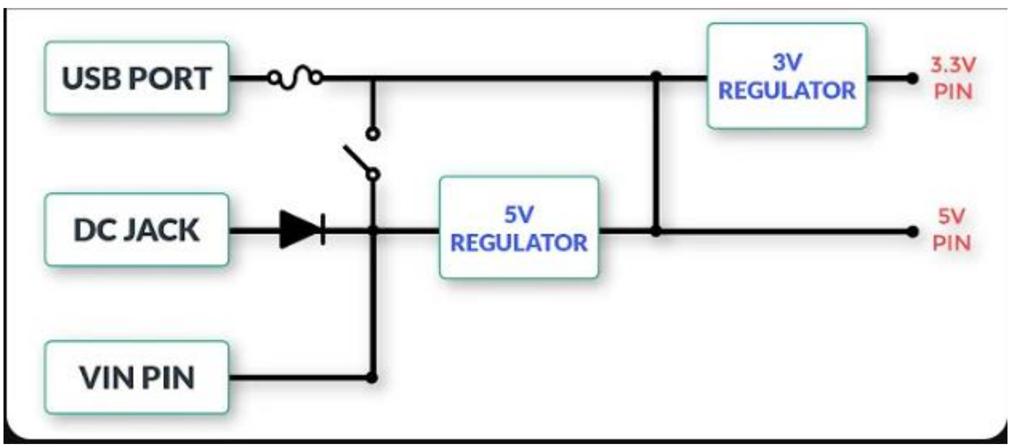
4.1) Design and Implementation

Software design is a creative activity in which you determine the relationships between software components based on the needs of a client. The process of making the design into a program is called Implementation.

4.1.1) Components Used

Arduino Mega

The microcontroller board known as the Arduino Mega 2560 is based on the ATmega2560. It features a 16 MHz crystal oscillator, 54 digital input/output pins (of which 15 may be used as PWM outputs), 16 analog inputs, 4 hardware serial ports (UARTs), a USB connector, a power jack, an ICSP header, and a reset button.



Powering Arduino Boards can be done in 3 ways; **First** way is by the ***USB Port*** by using this method you can only give a maximum voltage of 3.3V and it only turns the 3.3V regulator on and this method cannot be used for components that are using 5V. **Secondly *VIN Pin*** by this way you can use both the 5V regulator and 3.3V regulator. And by this way you can use the full potential of the Arduino Board**. Thirdly *DC Jack*** this is the best way to power an Arduino board because there is diode to stop reverse polarity and this way you can use both the 5V regulator and 3.3V regulator. And this is the best and safest way to power an Arduino.

RFID Module – RC522

The RC522 RFID module based on the MFRC522 IC from NXP is one of the cheapest RFID options you can get online for less than four dollars. It usually comes with an RFID card tag and a key fob tag with 1KB of memory. And the best part is that it can write a tag that means you can store any message in it.

The RC522 RFID reader module is designed to create a 13.56MHz electromagnetic field and communicate with RFID tags (ISO 14443A standard tags).

The reader can communicate with a microcontroller over a 4-pin SPI with a maximum data rate of 10 Mbps. It also supports communication over I2C and UART protocols.

The RC522 RFID module can be programmed to generate an interrupt, allowing the module to alert us when a tag approaches it, instead of constantly asking the module “Is there a card nearby?”.

The module’s operating voltage ranges from 2.5 to 3.3V, but the good news is that the logic pins are 5-volt tolerant, so we can easily connect it to an Arduino or any 5V logic microcontroller without using a logic level converter.

RC522 RFID Module Pinout

The RC522 module has a total of 8 pins that connect it to the outside world. The connections are as follows:

RC522 RFID Reader Writer Module Pinout

***VCC* -** supplies power to the module. This can be anywhere from 2.5 to 3.3 volts. You can connect it to the 3.3V output from your Arduino. But remember that connecting it to the 5V pin will probably destroy your module.

RST -is an input for reset and power-down. When this pin goes low the module enters power-down mode. In which the oscillator is turned off and the input pins are disconnected from the outside world. Whereas the module is reset on the rising edge of the signal.

***GND* -** is the ground pin and needs to be connected to the GND pin on the Arduino.

***IRQ* -** is an interrupt pin that alerts the microcontroller when an RFID tag is in the vicinity.

***MISO / SCL / Tx* -** pin acts as master-in-slave-out when SPI interface is enabled, as serial clock when I2C interface is enabled and as serial data output when the UART interface is enabled.

***MOSI (Master Out Slave In)* -** is the SPI input to the RC522 module.

SCK - (Serial Clock) accepts the clock pulses provided by the SPI bus master i.e., Arduino.

***SS / SDA / Rx* -** pin acts as a signal input when the SPI interface is enabled, as serial data when the I2C interface is enabled and as a serial data input when the UART interface is enabled. This pin is usually marked by encasing the pin in a square so that it can be used as a reference to identify other pins.

Ultrasonic Sensor – HC-SR04

An HC-SR04 ultrasonic distance sensor actually consists of two ultrasonic transducers. One acts as a transmitter that converts the electrical signal into 40 kHz ultrasonic sound pulses. The other acts as a receiver and listens for the transmitted pulses.

When the receiver receives these pulses, it produces an output pulse whose width is proportional to the distance of the object in front. This sensor provides excellent non-contact range detection between 2 cm to 400 cm (~13 feet) with an accuracy of 3 mm.

Since it operates on 5 volts, it can be connected directly to an Arduino or any other 5V logic microcontroller.

Let’s take a look at its pinout.

VCC - supplies power to the HC-SR04 ultrasonic sensor. You can connect it to the 5V output from your Arduino.

***Trig (Trigger)* -** pin is used to trigger ultrasonic sound pulses. By setting this pin to HIGH for 10µs, the sensor initiates an ultrasonic burst.

***Echo pin* -** goes high when the ultrasonic burst is transmitted and remains high until the sensor receives an echo, after which it goes low. By measuring the time, the Echo pin stays high, the distance can be calculated.

IR Sensor HW-201

An infrared proximity sensor or IR Sensor is an electronic device that emits infrared lights to sense some aspect of the surroundings and can be employed to detect the motion of an object. As this is a passive sensor, it can only measure infrared radiation.

IR Sensor Pinout

The IR sensor has a 3-pin connector that interfaces it to the outside world. The connections are as follows:

VCC - is the power supply pin for the IR sensor which we connect to the 5V pin on the Arduino.

OUT pin - is a 5V TTL logic output. LOW indicates no motion is detected; HIGH means motion is detected.

GND - Should be connected to the ground of the Arduino.

The working of the IR sensor module is very simple, it consists of two main components: the first is the IR transmitter section and the second is the IR receiver section. In the transmitter section, IR led is used and in the receiver section, a photodiode is used to receive infrared signal and after some signal processing and conditioning, you will get the output.

An IR proximity sensor works by applying a voltage to the onboard Infrared Light Emitting Diode which in turn emits infrared light. This light propagates through the air and hits an object, after that the light gets reflected in the photodiode sensor. If the object is close, the reflected light will be stronger, if the object is far away, the reflected light will be weaker. If you look closely toward the module. When the sensor becomes active it sends a corresponding Low signal through the output pin that can be sensed by an Arduino or any kind of microcontroller to execute a particular task. This module is that it has two onboard LEDs built-in, one of which lights on when power is available and another one turns on when the circuit gets triggered.

This sensor is used to detect proximity or to avoid obstacles. This Sensor is popular among beginners as these are low power, low cost, rugged, and feature a wide sensing range that can be trimmed down to adjust the sensitivity.

This sensor has three pins two of which are power pins levelled VCC and GND and the other one is the sense/data pin. It has an onboard power LED and a signal LED the power LED turns on when power is applied to the board the signal LED turns on when the circuit is triggered. This board also has a comparator Op-amp that is responsible for converting the incoming analog signal from the photodiode to a digital signal. We also have a sensitivity adjustment potentiometer; with that, we can adjust the sensitivity of the device. Last and finally, we have the photodiode and the IR emitting LED pair which all together make the total IR Proximity Sensor Module.

Connecting the IR sensor to any microcontroller is really simple. As we know this sensor outputs a digital signal and processing this signal is very easy. There exist two methods to do so first, you can always check the port in an infinite loop to see when the port changes its state from high to low, or the other way is to do it with an interrupt if you are making a method is recommended. Power the IR with 5V or 3.3V and connect ground to ground.

Next, we have our infinite loop. In the infinite loop, we first read the sensor pin with the digitalRead() function and store the value to sensor Status variable. Then we check to see if the output of the sensor is high or low, if the output of the sensor is high that means no motion is detected, else motion is detected, we also print this status in the serial monitor window.

Motor Driver L293D Shield

we can’t connect motors to microcontrollers or controller board such as Arduino directly in order to control them since they possibly need more current than a microcontroller can drive so we need drivers. The driver is an interface circuit between the motor and controlling unit to facilitate driving. Drives come in many different types.

Driving electromotors needs a high current. In addition, spinning direction and speed are two important parameters to be controlled. These requirements can be handled by using a microcontroller (or a development board like Arduino). But there is a problem; Microcontrollers cannot provide enough current to run the motor and if you connect the motor to the microcontroller directly, you may damage the microcontroller. For example, Arduino UNO pins are limited to 40mA of current which is far less than the 100-200mA current necessary to control a small hobby motor. To solve this, we should use a motor driver. Motor drivers can be connected to the microcontroller to receive commands and run the motor with a high current.

L293D shield is a driver board based on L293 IC, which can drive 4 DC motors and 2 stepper or Servo motors at the same time. Each channel of this module has the maximum current of 1.2A and doesn’t work if the voltage is more than 25v or less than 4.5v. So be careful with choosing the proper motor according to its nominal voltage and current. For more features of this shield let’s mention compatibility with Arduino UNO and MEGA, electromagnetic and thermal protection of motor and disconnecting circuit in case of unconventional voltage raise.

While using this shield 6 analog Pins (which can be used as digital pins too), pin 2 and pin 13 of Arduino are free.

In the case of using Servo motor, pins 9, 10, 2 are in use.

In the case of using DC motor, pin11 for #1, pin3 for #2, pin5 for #3, pin6 for #4 and pins 4, 7, 8 and 12 for all of them are in use.

In the case of using Stepper motor, pins 11 and 3 for #1, pins 5 and 6 for #2 and pins 4, 7, 8 and 12 for all of them are in use.

The first argument stands for the number of the motors in the shield and the second one stands for the motor speed control frequency. The second argument can be MOTOR12_2KHZ, MOTOR12_8KHZ, MOTOR12_8KHZ, and

MOTOR12_8KHZ for motors number 1 and 2, and it can be MOTOR12_8KHZ, MOTOR12_8KHZ, and MOTOR12_8KHZ for motors number 3 and 4. And if it left unchecked, it will be 1KHZ by default.

Arduino IDE library and examples are suitable for driving a Servo motor.

Determine motor speed in rpm. The first argument is the amount of step needed to move, the second one is to determine direction (FORWARD or BACKWARD), and the third argument determines the steps type: SINGLE (Activate a coil), DOUBLE (Activate two coils for more torque), INTERLEAVED (Continuous change in the number of coils from one to two and vice versa to double precision, however, in this case, the speed is halved), and MICROSTEP (Changing the steps is done slowly for more precision. In this case, the torque is lower). By default, when the motor stops moving, it maintains its status. You must use the function motor.release() to release the motor.

Gear Motor

Rotation Direction CW: [+] terminal connected to the positive power supply; [-] terminal connected to negative

power, clockwise is deemed by the direction of the output shaft

Servo Motor SG90

Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with 3 horns (arms) and hardware

Lion Battery 18650

One of the most popular and commonly found cells in the battery pack or power supply are 18650 battery cells, here we have listed all the possible technical 18650 battery specifications.

The 18650 battery is a Li-ion battery named after its 18mm × 65mm cylindrical size (diameter × height). When compared to AA size, it’s height and diameter both are larger. They are not replacements for AA or AAA size cells.

The 18650 battery has a nominal voltage of 3.7V and has capacity between 1200mAh and 3600mAh (read as milli-Amp-hours).

The battery type 18650 is popular in rechargeable and high current draining devices considering its high-level capabilities like 250+ charge cycle and higher energy density.

The 18650 Li-ion battery due to its adaptability all around can be found in various fields, say, electric cars/ scooters, power banks, utility gadgets such as emergency lamps, torchlight, etc.

Its safety property along with its high output current and energy capacity makes this battery popular.

4.1.2) Specifications

Technical Specifications of RFID RC522

Frequency Range: 13.56 MHz ISM Band

Host Interface: SPI / I2C / UART

Operating Supply Voltage: 2.5 V to 3.3 V

Max. Operating Current: 13-26mA

Min. Current (Power down): 10µA

Logic Inputs: 5V Tolerant

Read Range: 5 cm

Technical Specifications of Ultrasonic sensor HC- SR04

Operating Voltage: DC 5V

Operating Current: 15mA

Operating Frequency: 40KHz

Max Range: 4m

Min Range: 2cm

Ranging Accuracy: 3mm

Measuring Angle: 15 degrees

Trigger Input Signal: 10µS TTL pulse

Dimension:45 x 20 x 15mm

Technical Specifications of Gear Motor

Rated Voltage: 6V DC

No load speed: 12000±15%rpm

No load current: ≤280mA

Operating voltage: 1.5-6.5V DC

Starting Torque: ≥250g.cm(according to ourself developed blade)

starting current: ≤5A

shaft gap 0.05-0.35m

Technical Specifications of Servo Motor SG90

Speed (sec): 0.1

Torque (kg-cm): 2.5

Weight (g) :14.7

Voltage: 4.8 – 6 V

Technical Specifications of 18650 battery

Component type: Battery/ Cell

Battery type: Li-Ion Technology

Model No: 18650

Size/ dimension: 18 x 65 mm Standard (dd x hh)

Voltage: 3.7V

Capacity:1200-3600mah Per cell

Operating voltage: 2.5-4.2 V

Cut-off voltage: 2 - 2.5 volts

Weight: 30gms to 55gms

Charge density (Energy per cell):1.5 - 11.5Wh

Charge discharge cycle: 250 to 2000 In number ()

Optimum /Minimum charging time: 2.5 hrs to 3.5 hrs

Per cell life - 36+ months

Charging method - CC and CV

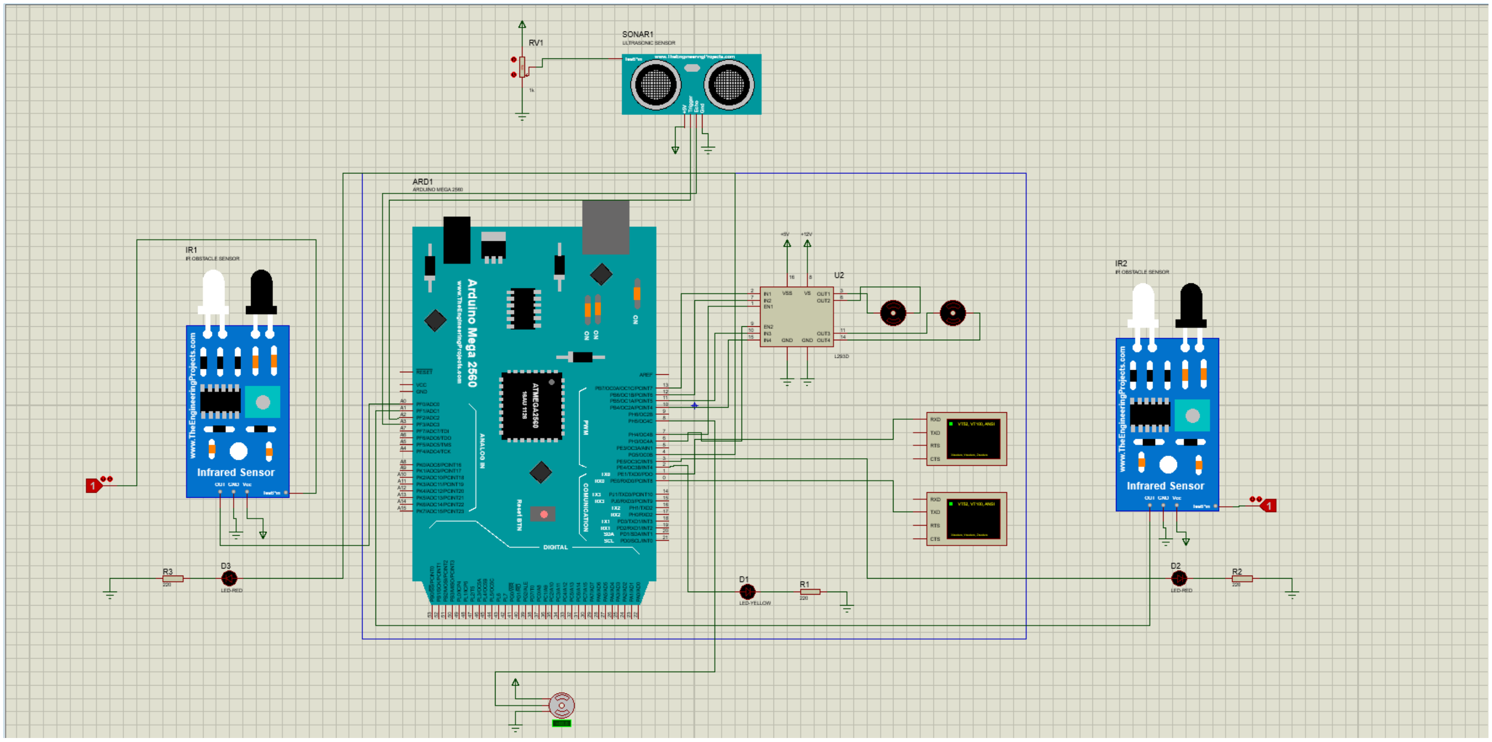
Charging voltage 4.2v - 5v

Internal resistance - High

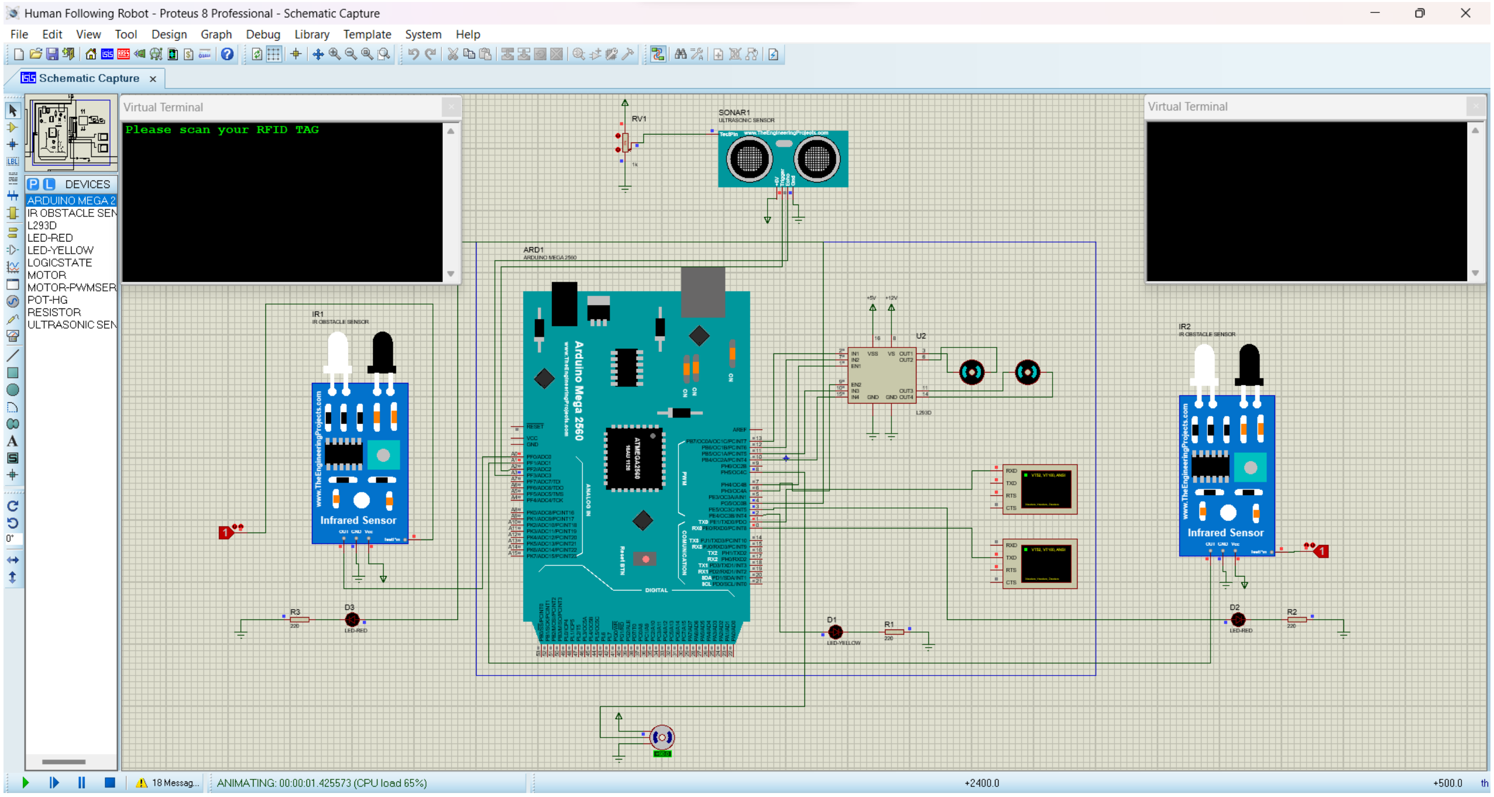
4.2) Software Simulation

Software simulation of this project was done by *Proteus 8 Professional* app and due to some lack of components we have different Codes for software and hardware simulations.

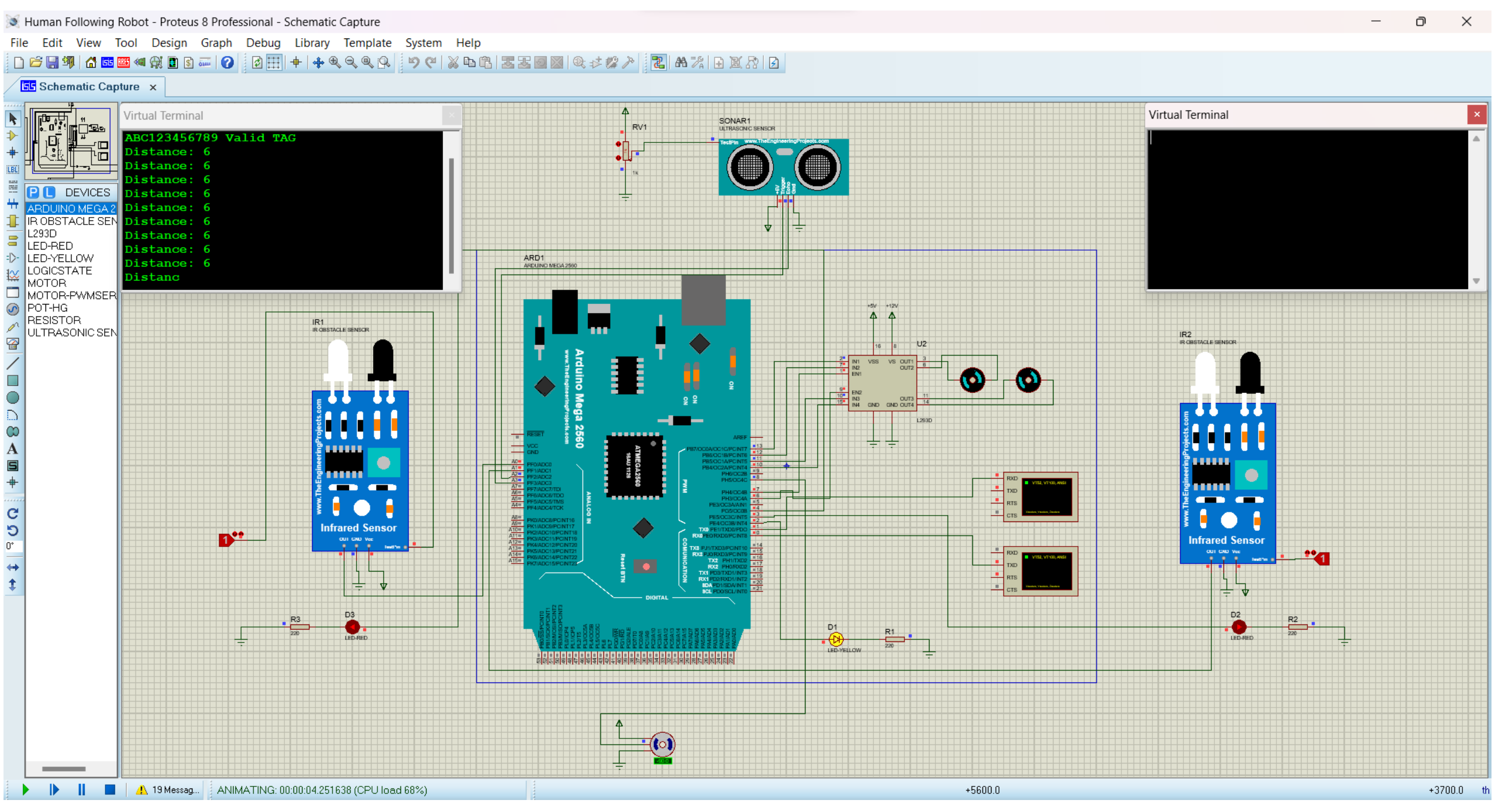
- Full software simulation without operating.



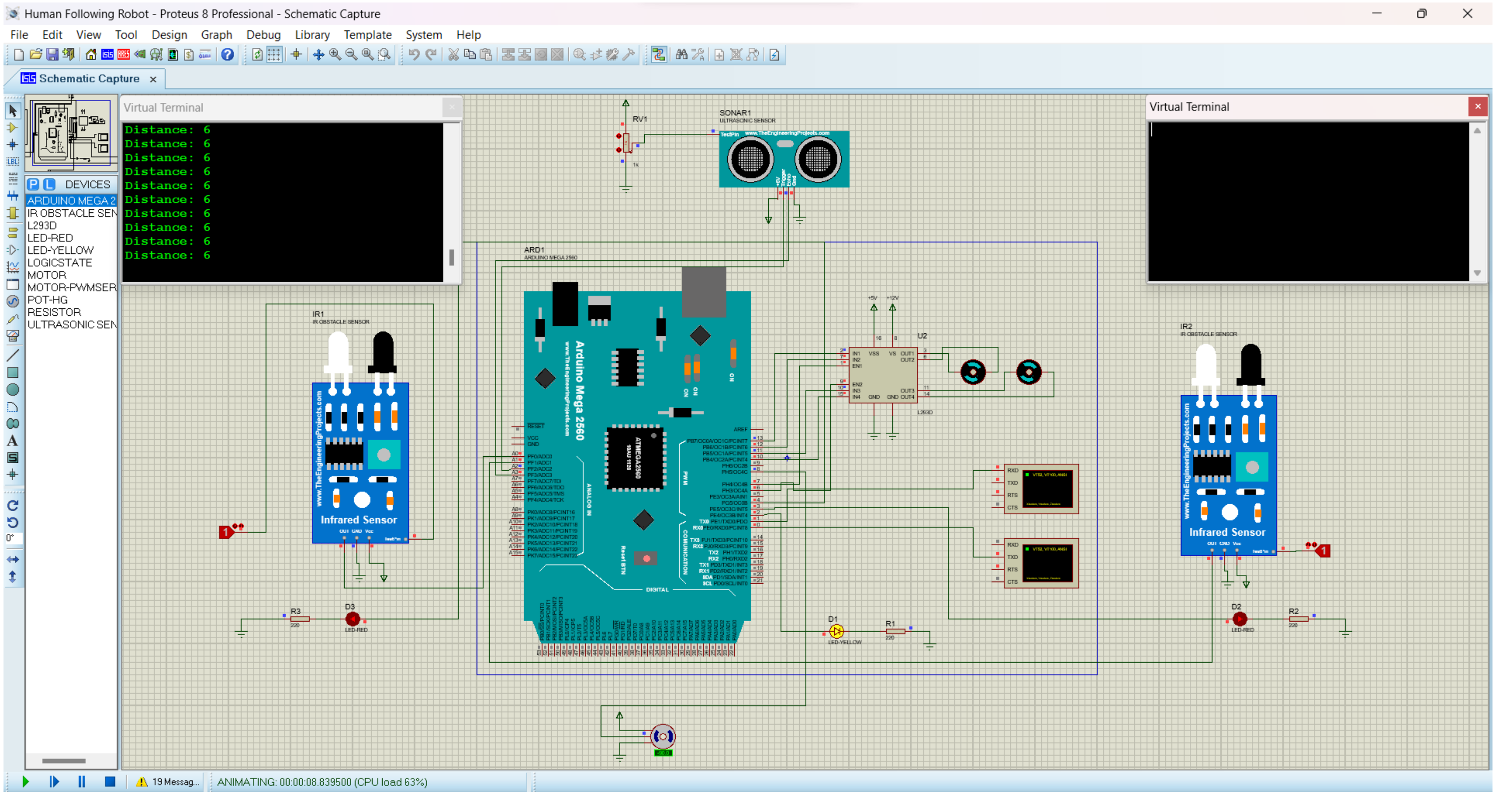
- Moment where we need to scan the RFID card.



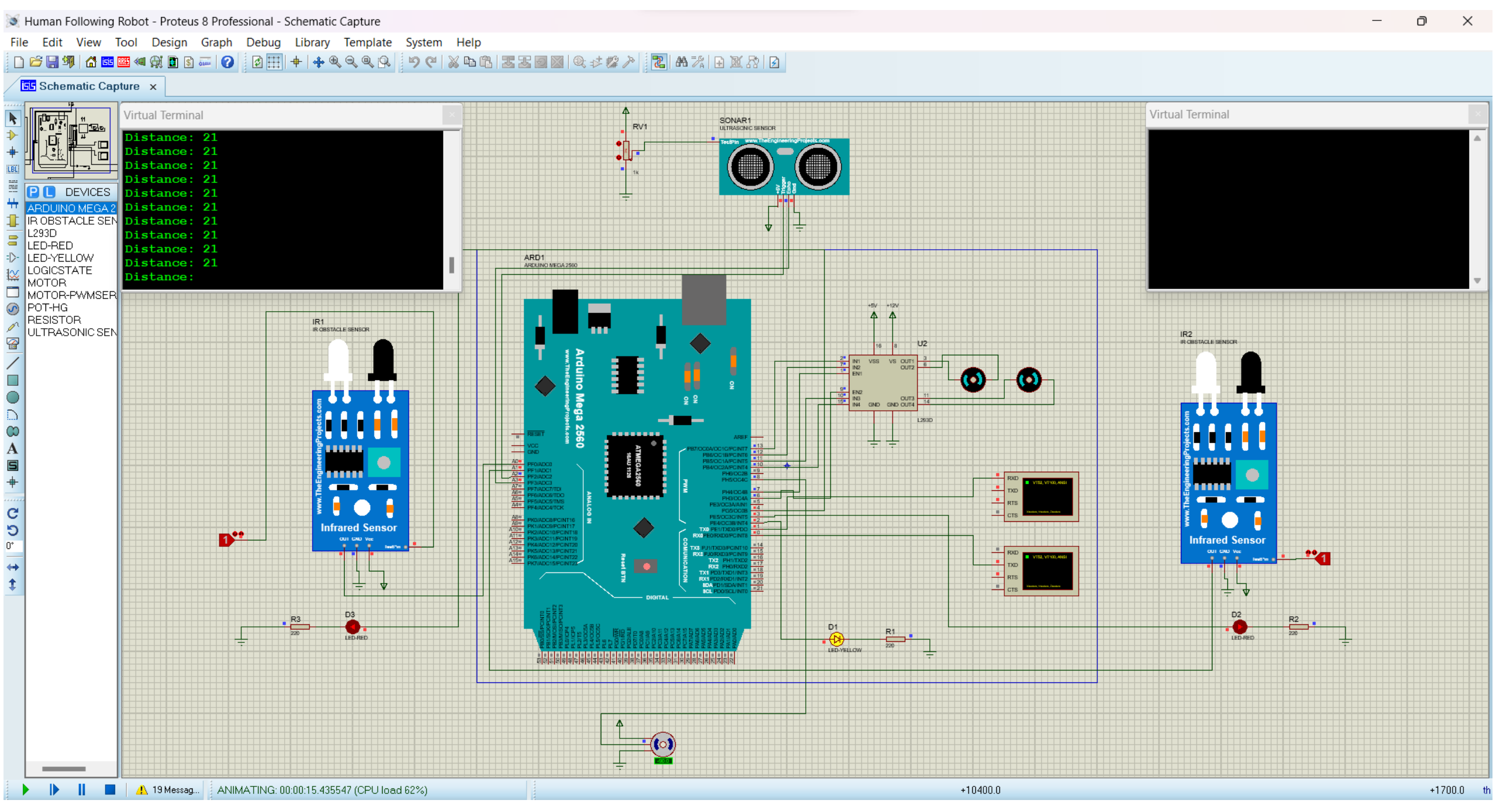
- If RFID is valid only then the yellow LED glows.

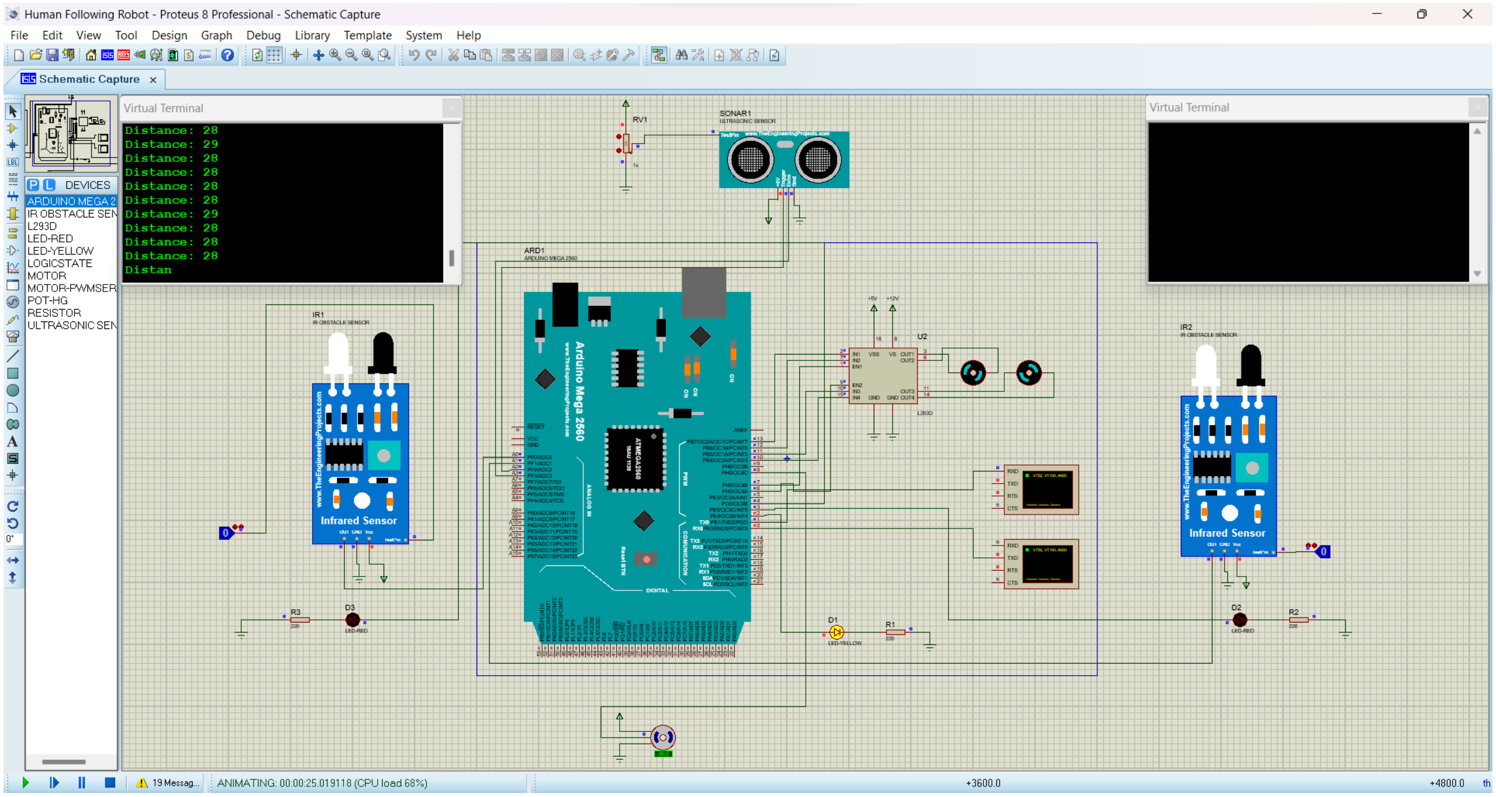


- Trolley moving Backward as the distance is less than 10.

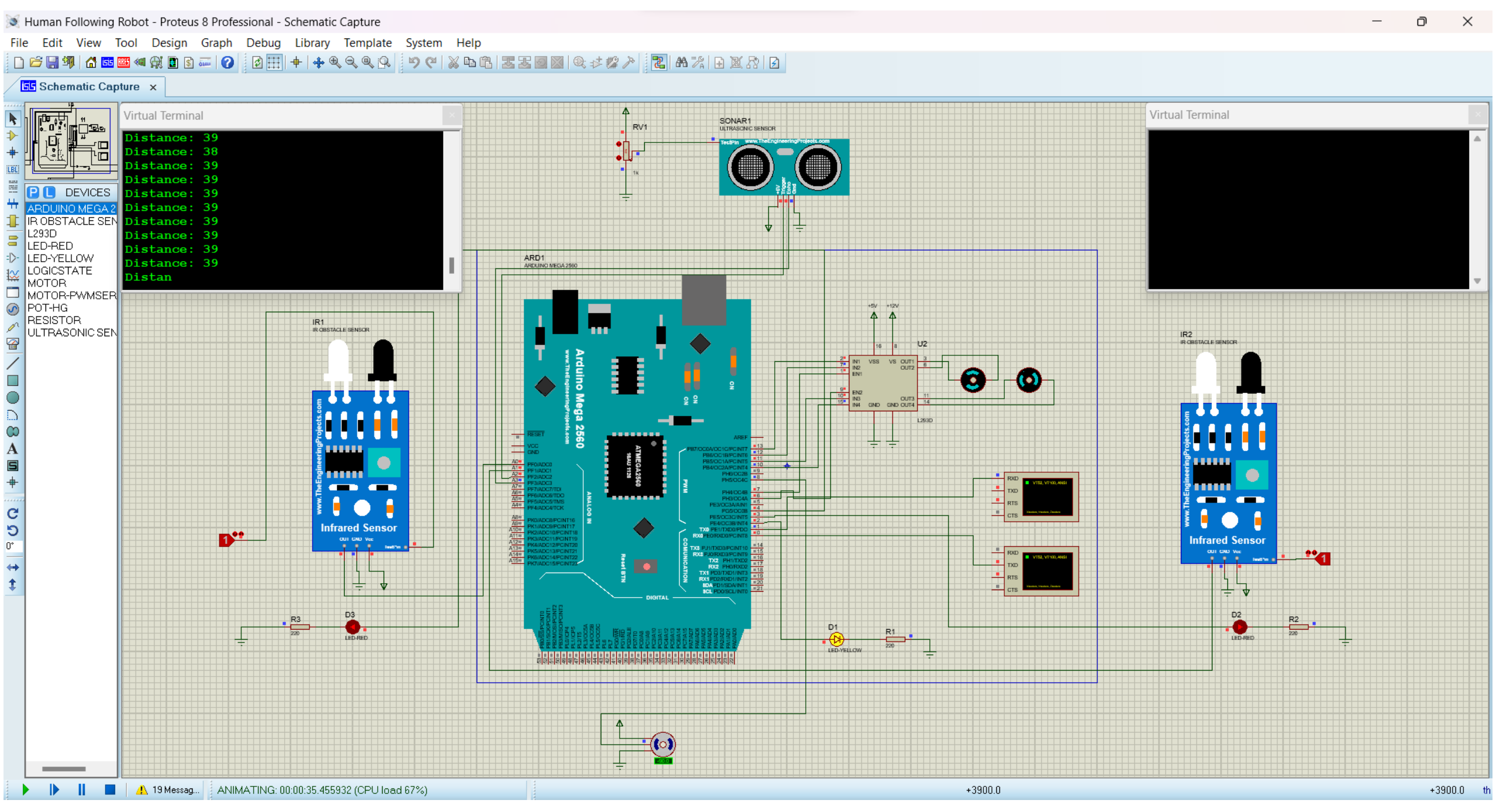


- Trolley stops as the distance between human and trolley is greater than 10 and less than 30.

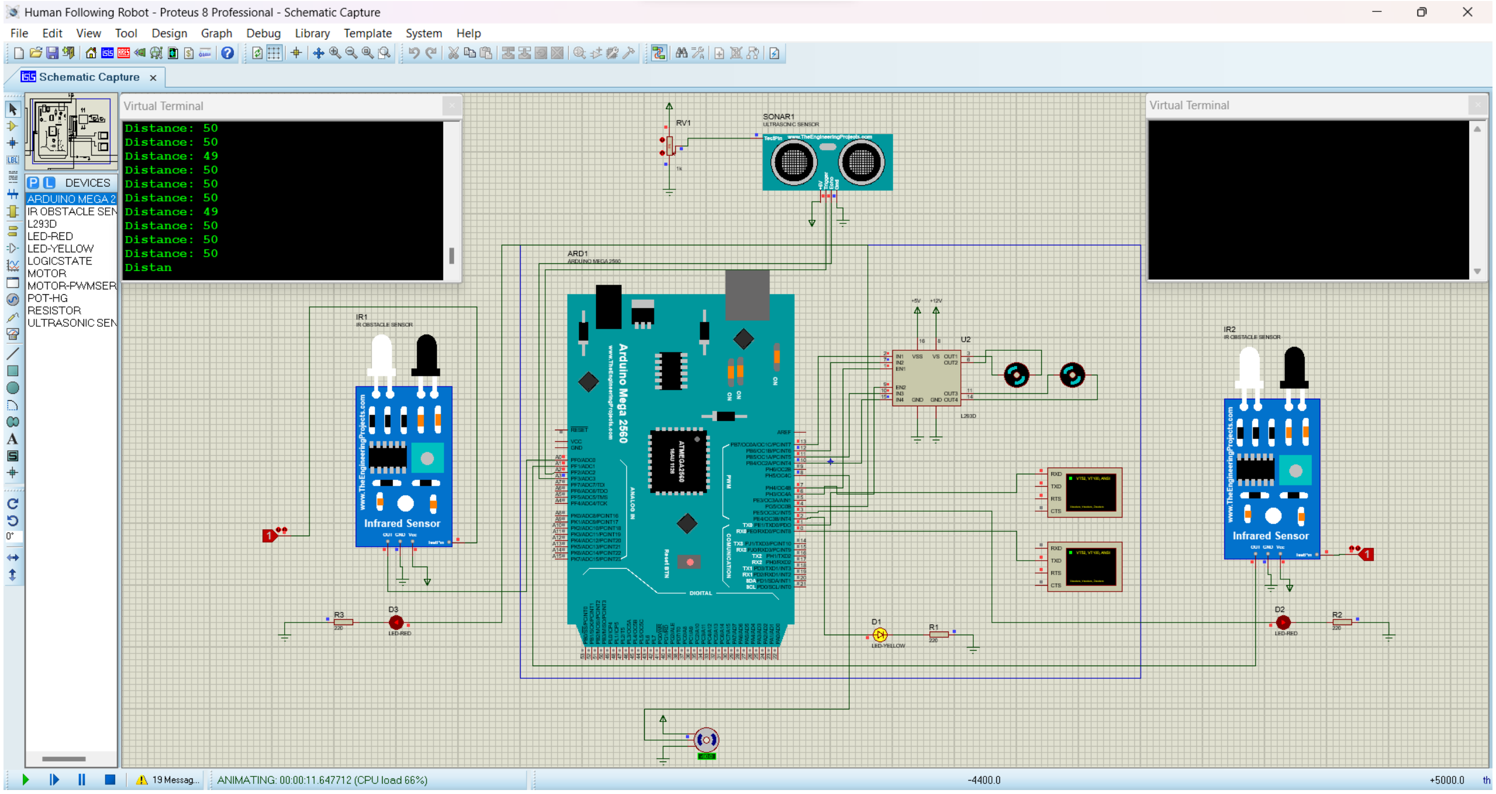




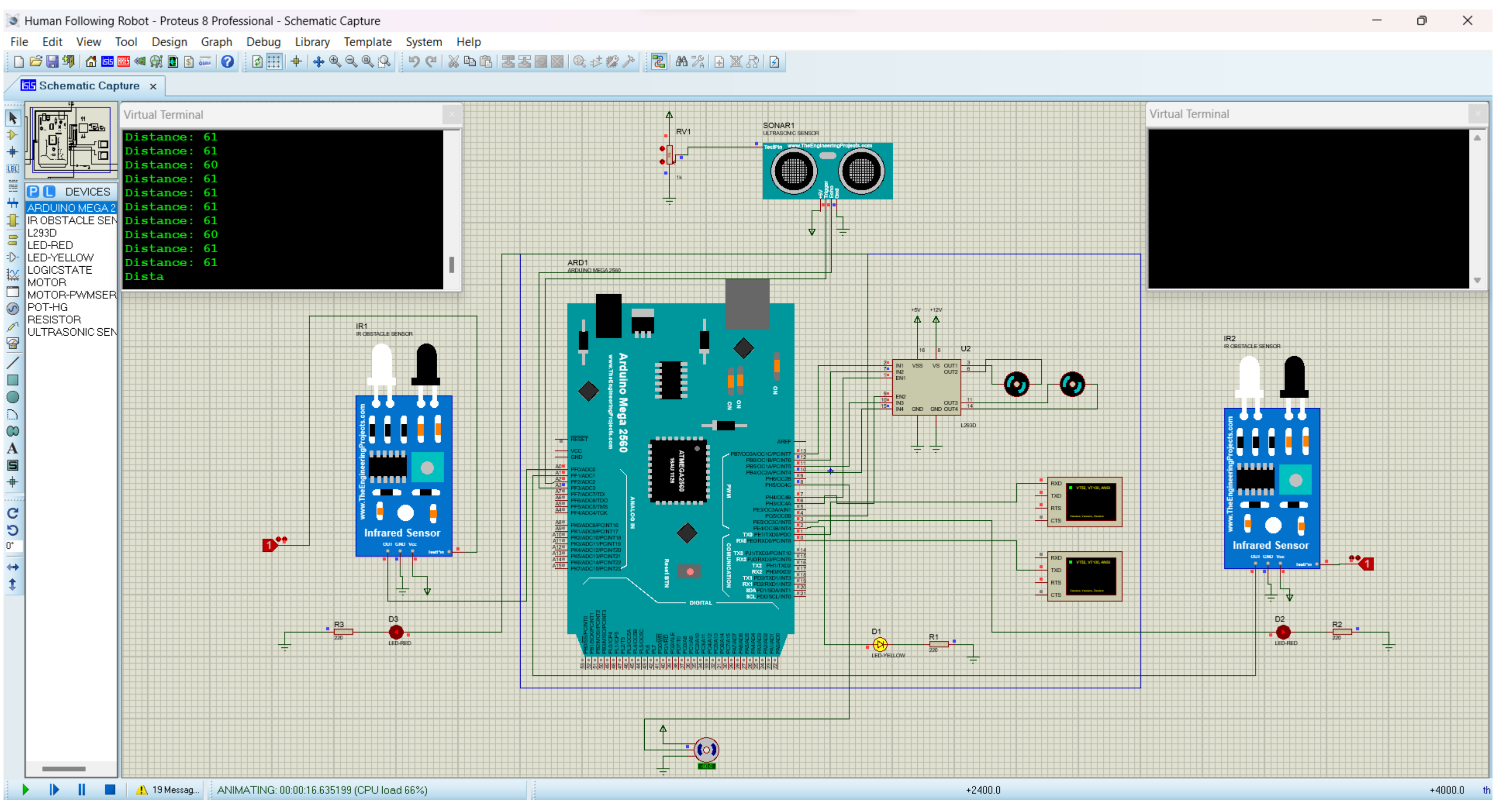
- Trolley reduces its speed as it reaches the distance 30 < distance < 50.



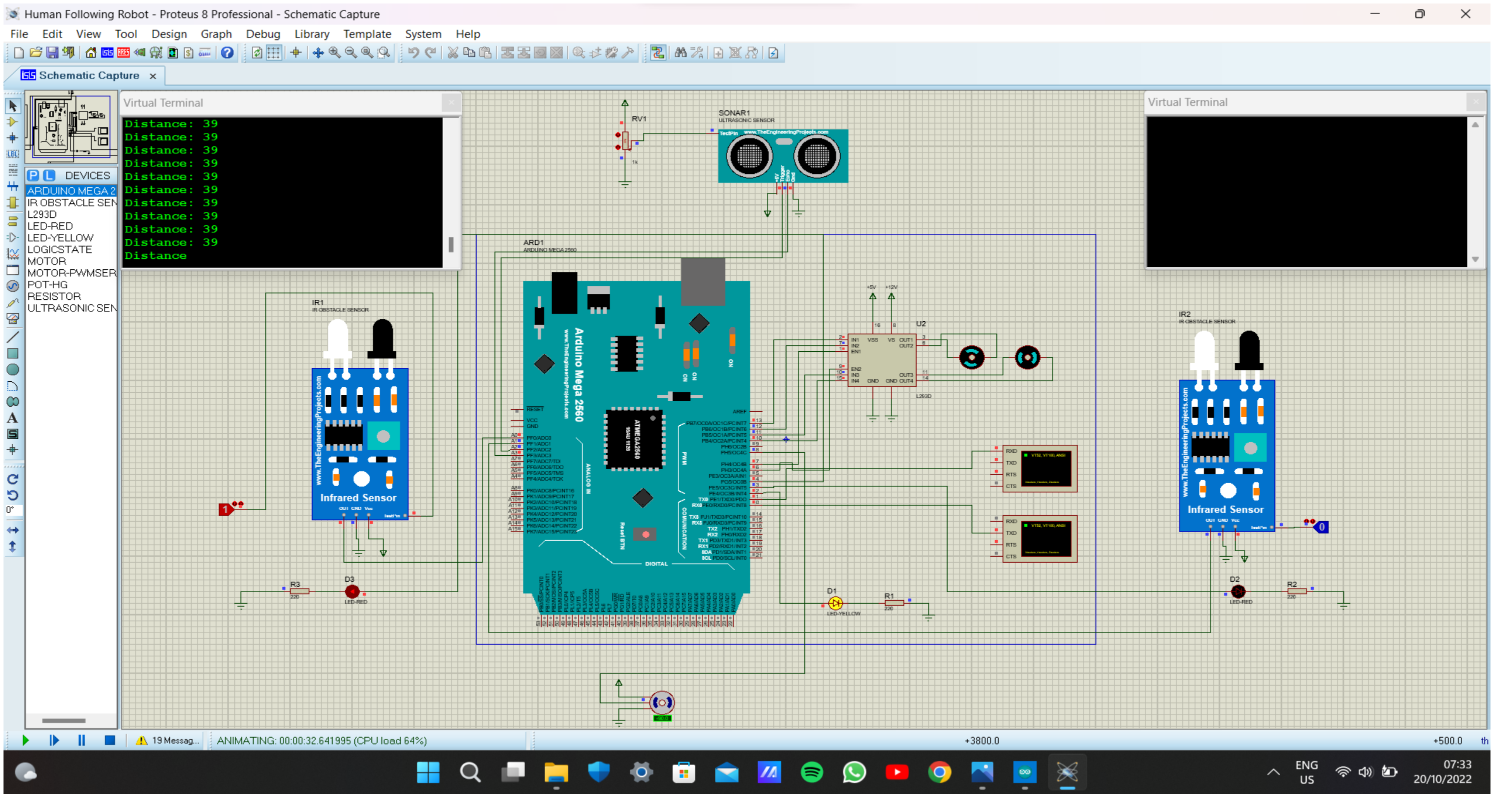
- Trolley reaches a constant speed when distance is equal to 50 and this distance is assumed as 1m.



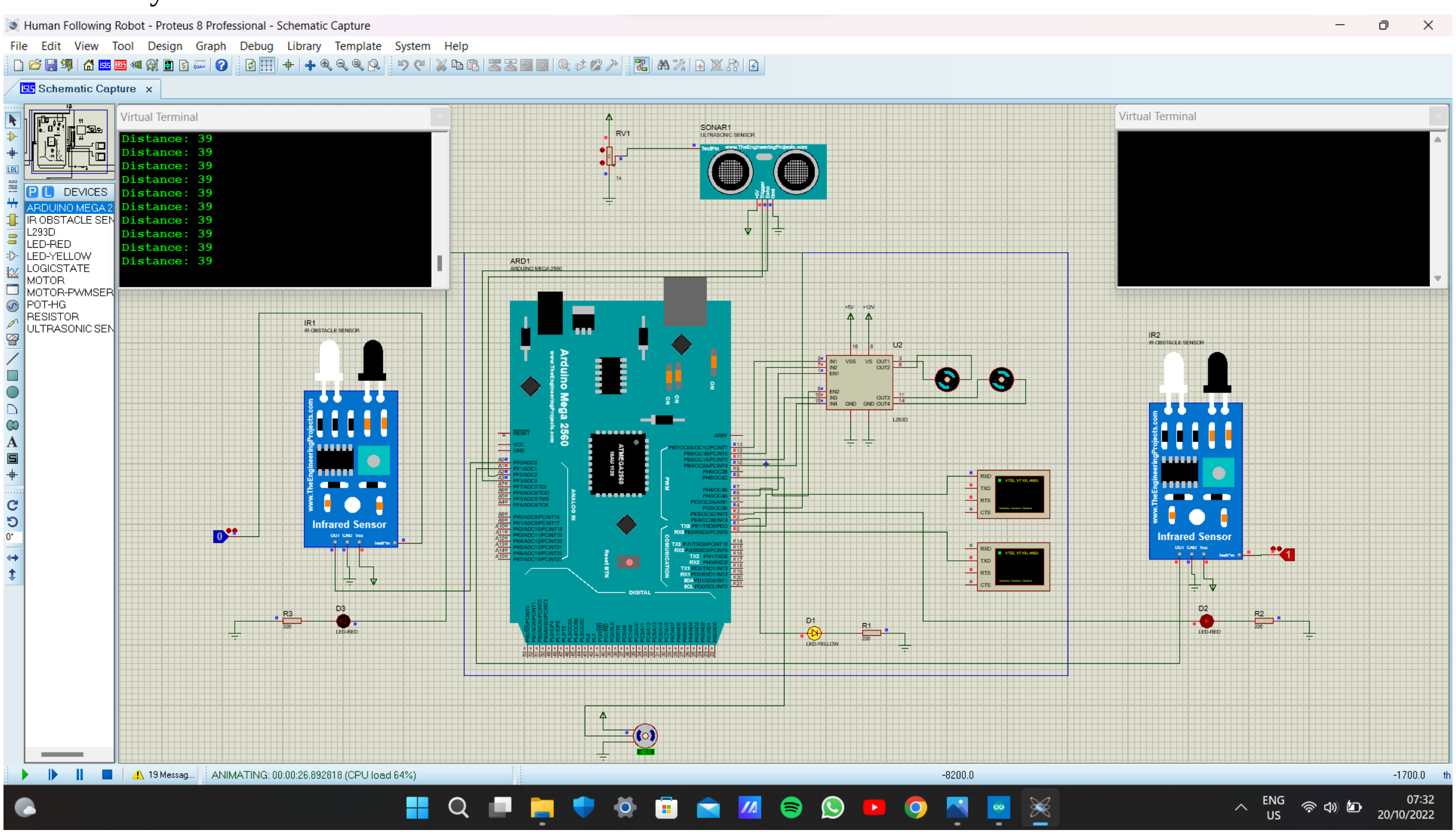
- Trolley increases its speed to keep the constant speed when the distance is greater than 50.



- Trolley turns Right.



- Trolley turns Left.



Full code of the Software simulation.

```
#include<Servo.h>
Servo myservo;
int pos = 0;
int count = 0;
char c;
String id;

int en1 = 7;
int en2 = 6;

const int in1 = 13;
const int in2 = 12;
const int in3 = 11;
const int in4 = 10;

const int trigPin = A2; // Trigger Pin of Ultrasonic Sensor
const int echoPin = A3; // Echo Pin of Ultrasonic Sensor

long duration;
int distance;

#define LEFT A0 // The number of the left infrared sensor pin
#define RIGHT A1 //The number of the Right infrared sensor pin

const int ledPin1 =  4;
const int ledPin2 =  3;

void setup() {

myservo.attach(8);
Serial.begin (9600) ; // Starting Serial Communication
pinMode(2, OUTPUT); //initialising pin 2 as output for the RFID
Serial.println("Please scan your RFID TAG");

for(pos = 90; pos <= 180; pos += 1){
myservo.write(pos);
delay(15);
}

for(pos = 180; pos >= 0;pos-= 1){
myservo.write(pos) ;
delay (15) ;
}

for(pos = 0; pos >= 90;pos-= 1){
myservo.write(pos) ;
delay (15) ;
}

 pinMode (en1, OUTPUT);
 pinMode (en2, OUTPUT);
```

```
 pinMode (in1, OUTPUT) ;
 pinMode (in2, OUTPUT) ;
 pinMode (in3, OUTPUT) ;
 pinMode (in4, OUTPUT) ;

 pinMode (trigPin, OUTPUT) ; // initialising pin A2 as output
 pinMode (echoPin, INPUT) ; // initialising pin A3 as input

 pinMode (RIGHT, INPUT) ; //initialize the infrared sensor sensor pin as an input:
 pinMode (LEFT, INPUT) ;

 pinMode (ledPin1, OUTPUT); //initialize the LED pin as an output:
 pinMode (ledPin2, OUTPUT);

 }

void loop () {

while(Serial.available()>0)
  {
    c = Serial.read();
   count++;
   id += c;
   if(count == 12)
    {
      Serial.print(id);
      //break;

      if(id=="ABC123456789")
      {
        Serial.println(" Valid TAG");
        digitalWrite(2, HIGH);
      }
      else
      {
      digitalWrite(2, LOW);
      Serial.println(" Invalid TAG");
      }
    }
  }

 digitalWrite (trigPin, LOW) ;
 delay (2) ;
 digitalWrite (trigPin, HIGH) ;
 delay (10) ;
 digitalWrite (trigPin, LOW) ;

 duration = pulseIn (echoPin,HIGH) ;
 distance = (duration*0.034/2) ;
 Serial.print ("Distance: ");
 Serial.println (distance) ;

 int Right_Value=digitalRead(RIGHT) ;
 int Left_Value=digitalRead(LEFT) ;
```

```
// Move Forward in a constant speed

if((id=="ABC123456789") && (distance == 50) && (Right_Value==1) && (Left_Value==1))
{
 digitalWrite (in1,HIGH) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,HIGH) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 150);
    analogWrite(en2, 150);
}

else if((id=="ABC123456789") && (distance == 50) && (Right_Value==0) && (Left_Value==0))
{
 digitalWrite (in1,HIGH) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,HIGH) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 150);
    analogWrite(en2, 150);
}

//Move Forward fast
else if((id=="ABC123456789") && (distance>50) && (Right_Value==1) && (Left_Value==1))
{
 digitalWrite (in1,HIGH) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,HIGH) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 200);
    analogWrite(en2, 200);
}

else if((id=="ABC123456789") && (distance>50) &&(Right_Value==0) && (Left_Value==0))
{
 digitalWrite (in1,HIGH) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,HIGH) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 200);
    analogWrite(en2, 200);
}
//Move Forward Slow
else if((id=="ABC123456789") && (distance>30 && distance<50) && (Right_Value==1) &&
(Left_Value==1))
{
 digitalWrite (in1,HIGH) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,HIGH) ;
```

```
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 100);
    analogWrite(en2, 100);
}

else if((id=="ABC123456789") && (distance>30 && distance<50) && (Right_Value==0) &&
(Left_Value==0))
{
 digitalWrite (in1,HIGH) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,HIGH) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 100);
    analogWrite(en2, 100);
}

// Right turn

else if((id=="ABC123456789") && (Right_Value==0) && (Left_Value==1))
{
 digitalWrite (in1,HIGH) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,LOW) ;
 digitalWrite (in4,HIGH) ;

    analogWrite(en1, 150);
    analogWrite(en2, 150);
}

//Left turn

else if((id=="ABC123456789") && (Right_Value==1) && (Left_Value==0))
{
 digitalWrite (in1,LOW) ;
 digitalWrite (in2,HIGH) ;
 digitalWrite (in3,HIGH) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 150);
    analogWrite(en2, 150);
}

//Move Backward

else if((id=="ABC123456789") && (Right_Value==1) && (Left_Value==1) && (distance<10))
{
 digitalWrite (in1,LOW) ;
 digitalWrite (in2,HIGH) ;
 digitalWrite (in3,LOW) ;
 digitalWrite (in4,HIGH) ;

    analogWrite(en1, 150);
```

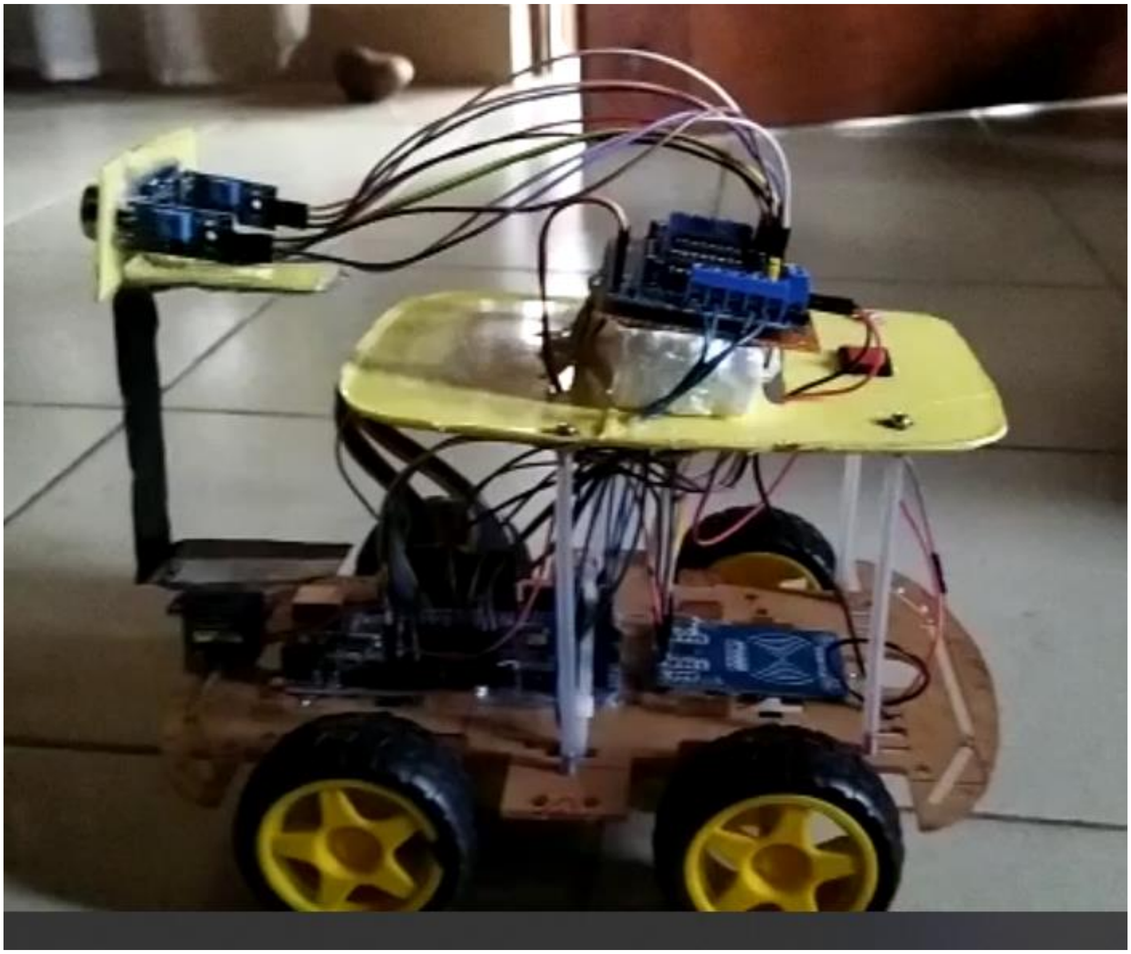
```
    analogWrite(en2, 150);
}

else if((id=="ABC123456789") && (Right_Value==0) && (Left_Value==0) && (distance<10))
{
 digitalWrite (in1,LOW) ;
 digitalWrite (in2,HIGH) ;
 digitalWrite (in3,LOW) ;
 digitalWrite (in4,HIGH) ;

    analogWrite(en1, 150);
    analogWrite(en2, 150);
}

// Stop

else if((id=="ABC123456789") && (Right_Value==1) && (Left_Value==1) && (distance>10 &&
distance<30))
{
 digitalWrite (in1,LOW) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,LOW) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 0);
    analogWrite(en2, 0);
}

else if((id=="ABC123456789") && (Right_Value==0) && (Left_Value==0) && (distance>10 &&
distance<30))
{
 digitalWrite (in1,LOW) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,LOW) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 0);
    analogWrite(en2, 0);
}

else if((id=="ABC123456789") && (Right_Value==1) && (Left_Value==0) && (distance>10 &&
distance<30))
{
 digitalWrite (in1,LOW) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,LOW) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 0);
    analogWrite(en2, 0);
}

else if((id=="ABC123456789") && (Right_Value==0) && (Left_Value==1) && (distance>10 &&
distance<30))
```

```
{
 digitalWrite (in1,LOW) ;
 digitalWrite (in2,LOW) ;
 digitalWrite (in3,LOW) ;
 digitalWrite (in4,LOW) ;

    analogWrite(en1, 0);
    analogWrite(en2, 0);
}

{
 int sensorState1 = digitalRead (LEFT); //Read the state of the Left infrared sensor

 if (sensorState1 == HIGH) //If the infrared sensor detects any object or obstacle
 {
  digitalWrite(ledPin1, HIGH); //Turn LED ON
  }
 else //If the infrared sensor does not detect any object or obstacle, then sensorState
state is LOW
 {
  digitalWrite(ledPin1, LOW); //Turn LED OFF
  }
 int sensorState2 = digitalRead (RIGHT); //Read the state of the Right infrared sensor

 if (sensorState2 == HIGH)
 {
  digitalWrite(ledPin2, HIGH);
  }

 else
 {
  digitalWrite(ledPin2, LOW);
  }
}
}
```

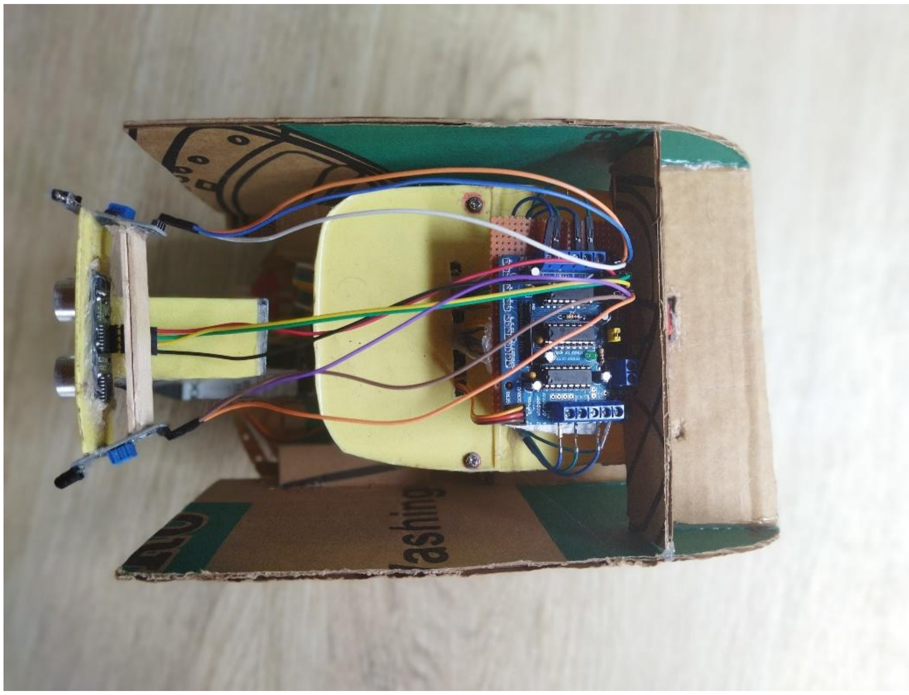
https://drive.google.com/drive/folders/1X-qQyFA3eJcOagkt1PhybQSTVaSiSIMy?usp=sharing

4.3) Hardware Simulation

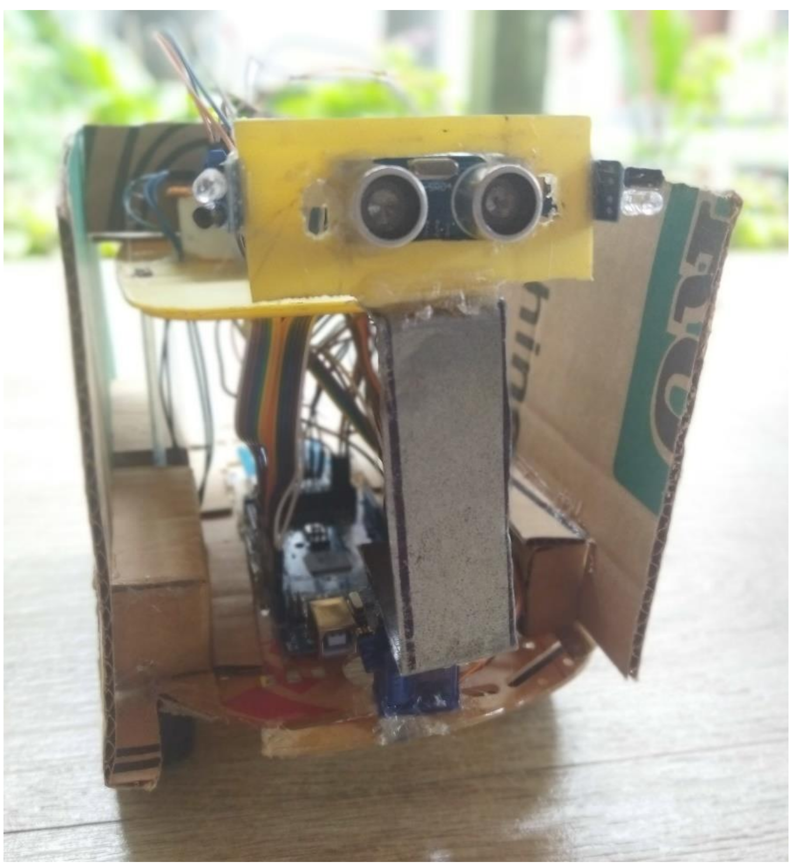
- Side View



- Top View



- Front View



Full code used for Hardware.

```
#include<NewPing.h>              // ultrasonic headerfolder
#include<SPI.h>               // allows to communicate with spi devices
#include<MFRC522.h>             // RFID headerfolder for MFRC522 (SPI) read/write a rfid
card or tag
#include<Servo.h>             // servo motor
#include<AFMotor.h>           // motor driver shield

#define SS_PIN 53                       // RFID pin
#define RST_PIN 49
MFRC522 mfrc522(SS_PIN, RST_PIN);
#define RIGHT A5                   // Arduino pin tied to Right IR
#define LEFT A0                    // Arduino pin tied to Left IR
#define TRIGGER_PIN A3             // Arduino pin tied to trigger pin on the ultrasonic
sensor
#define ECHO_PIN A1                 // Arduino pin tied to echo pin on the ultrasonic
sensor
#define MAX_DISTANCE 100          // Maximum distance we want to ping for (in
centimeters)
NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);   //NewPing setup of pins and
maximum distance
AF_DCMotor Motor1(1,MOTOR12_1KHZ);           // define motor with PWM frequencies
AF_DCMotor Motor2(2,MOTOR12_1KHZ);
AF_DCMotor Motor3(3,MOTOR34_1KHZ);
AF_DCMotor Motor4(4,MOTOR34_1KHZ);
Servo myservo;
int pos =0;    //initial position

void setup()
{
  Serial.begin(9600);  // data exchange rate per sec
  SPI.begin();           //initializes the SPI bus
  mfrc522.PCD_Init();
  Serial.println("Scan RFID Card");  // Scan the RFID card

 myservo.attach(10);  //servo motor
 {
 for(pos = 90; pos <= 180; pos += 1){   // servo motor rotates 90 degrees clockwise and
90 degrees anti-clockwise
  myservo.write(pos);
  delay(15);
 } for(pos = 180; pos >= 0; pos-= 1) {
  myservo.write(pos);
  delay(15);
 }for(pos = 0; pos<=90; pos += 1) {
  myservo.write(pos);
  delay(15);
 }
 }
 pinMode(RIGHT, INPUT);  // IR inputs
 pinMode(LEFT, INPUT);
}
```

```
void loop()
{
  if ( ! mfrc522.PICC_IsNewCardPresent())      // If a new PICC placed to RFID reader
continue
  {
    return;
  }
  if ( ! mfrc522.PICC_ReadCardSerial())         // Since a PICC placed get a serial and
continue
  {
    return;
  }

  //show UID on serial monitor
  Serial.print("UID tag :");
  String content="";
  byte letter;
  for (byte i = 0; i <mfrc522.uid.size; i++)     // for size of uid.size write
uid.uidByte to readCard
  {
    Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? "0" : " ");
    Serial.print(mfrc522.uid.uidByte[i], HEX);
    content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? "0" : " "));
    content.concat(String(mfrc522.uid.uidByte[i], HEX));

    if (content.substring(1) == "B0 C4 B2 30")    //  the UID of the card that want to
give access
    Serial.println("Authorized access");
    Serial.println();

  delay(50);
  unsigned int distance = sonar.ping_cm();
 Serial.print("distance");
 Serial.println(distance);
 int Right_Value = digitalRead(RIGHT);
 int Left_Value = digitalRead(LEFT);
 Serial.print("RIGHT");
 Serial.println(Right_Value);
 Serial.print("LEFT");
 Serial.println(Left_Value);

 if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==0) &&
(distance==10)&&(Left_Value==0))                // forward by keeping the distance
{
  Motor1.setSpeed(150);
  Motor1.run(FORWARD);
  Motor2.setSpeed(150);
  Motor2.run(FORWARD);
  Motor3.setSpeed(150);
  Motor3.run(FORWARD);
  Motor4.setSpeed(150);
  Motor4.run(FORWARD);
 }
```

```
 else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==1) &&
(distance==10)&&(Left_Value==1))       //Forward by keeping the distance
{
  Motor1.setSpeed(150);
  Motor1.run(FORWARD);
  Motor2.setSpeed(150);
  Motor2.run(FORWARD);
  Motor3.setSpeed(150);
  Motor3.run(FORWARD);
  Motor4.setSpeed(150);
  Motor4.run(FORWARD);
 }
 else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==0) &&
(distance>10)&&(Left_Value==0))       //Forward but speed up to keep the distance
{
  Motor1.setSpeed(200);
  Motor1.run(FORWARD);
  Motor2.setSpeed(200);
  Motor2.run(FORWARD);
  Motor3.setSpeed(200);
  Motor3.run(FORWARD);
  Motor4.setSpeed(200);
  Motor4.run(FORWARD);
 }
 else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==1) &&
(distance>10)&&(Left_Value==1))       //Forward but speed up to keep the distance
{
  Motor1.setSpeed(200);
  Motor1.run(FORWARD);
  Motor2.setSpeed(200);
  Motor2.run(FORWARD);
  Motor3.setSpeed(200);
  Motor3.run(FORWARD);
  Motor4.setSpeed(200);
  Motor4.run(FORWARD);
 }
else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==0) && (distance>5 &&
distance<10)&&(Left_Value==0))       //Forward but speed down to keep the distance
{
  Motor1.setSpeed(130);
  Motor1.run(FORWARD);
  Motor2.setSpeed(130);
  Motor2.run(FORWARD);
  Motor3.setSpeed(130);
  Motor3.run(FORWARD);
  Motor4.setSpeed(130);
  Motor4.run(FORWARD);
 }
 else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==1) && (distance>5 &&
distance<10)&&(Left_Value==1))       //Forward but speed down to keep the distance
{
  Motor1.setSpeed(130);
  Motor1.run(FORWARD);
  Motor2.setSpeed(130);
```

```
  Motor2.run(FORWARD);
  Motor3.setSpeed(130);
  Motor3.run(FORWARD);
  Motor4.setSpeed(130);
  Motor4.run(FORWARD);
 }

else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==0) && (Left_Value==1))
// Left
{
  Motor1.setSpeed(150);
  Motor1.run(FORWARD);
  Motor2.setSpeed(150);
  Motor2.run(FORWARD);
  Motor3.setSpeed(150);
  Motor3.run(BACKWARD);
  Motor4.setSpeed(150);
  Motor4.run(BACKWARD);
 }
else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==1)&&(Left_Value==0))
// Right
{
  Motor1.setSpeed(150);
  Motor1.run(BACKWARD);
  Motor2.setSpeed(150);
  Motor2.run(BACKWARD);
  Motor3.setSpeed(150);
  Motor3.run(FORWARD);
  Motor4.setSpeed(150);
  Motor4.run(FORWARD);
 }
else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==0) &&
(distance==5)&&(Left_Value==0))           // Stop
{
  Motor1.setSpeed(0);
  Motor1.run(RELEASE);
  Motor2.setSpeed(0);
  Motor2.run(RELEASE);
  Motor3.setSpeed(0);
  Motor3.run(RELEASE);
  Motor4.setSpeed(0);
  Motor4.run(RELEASE);
 }
  else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==1) &&
(distance==5)&&(Left_Value==1)) // Stop
{
  Motor1.setSpeed(0);
  Motor1.run(RELEASE);
  Motor2.setSpeed(0);
  Motor2.run(RELEASE);
  Motor3.setSpeed(0);
  Motor3.run(RELEASE);
  Motor4.setSpeed(0);
  Motor4.run(RELEASE);
```

```
 }
else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==1) &&
(distance==5)&&(Left_Value==0))          // Stop
{
  Motor1.setSpeed(0);
  Motor1.run(RELEASE);
  Motor2.setSpeed(0);
  Motor2.run(RELEASE);
  Motor3.setSpeed(0);
  Motor3.run(RELEASE);
  Motor4.setSpeed(0);
  Motor4.run(RELEASE);
 }
  else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==0) &&
(distance==5)&&(Left_Value==1)) // Stop
{
  Motor1.setSpeed(0);
  Motor1.run(RELEASE);
  Motor2.setSpeed(0);
  Motor2.run(RELEASE);
  Motor3.setSpeed(0);
  Motor3.run(RELEASE);
  Motor4.setSpeed(0);
  Motor4.run(RELEASE);
 }
 else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==0) &&
(distance<5)&&(Left_Value==0))   // Backward
{
  Motor1.setSpeed(150);
  Motor1.run(BACKWARD);
  Motor2.setSpeed(150);
  Motor2.run(BACKWARD);
  Motor3.setSpeed(150);
  Motor3.run(BACKWARD);
  Motor4.setSpeed(150);
  Motor4.run(BACKWARD);
 }
  else if((content.substring(1) == "B0 C4 B2 30") && (Right_Value==1) &&
(distance<5)&&(Left_Value==1)) // Backward
{
  Motor1.setSpeed(150);
  Motor1.run(BACKWARD);
  Motor2.setSpeed(150);
  Motor2.run(BACKWARD);
  Motor3.setSpeed(150);
  Motor3.run(BACKWARD);
  Motor4.setSpeed(150);
  Motor4.run(BACKWARD);
 }
else if((Right_Value==1)&&(Left_Value==1))                // Stop if the card is not
detected
{
  Motor1.setSpeed(0);
  Motor1.run(RELEASE);
```

```
  Motor2.setSpeed(0);
  Motor2.run(RELEASE);
  Motor3.setSpeed(0);
  Motor3.run(RELEASE);
  Motor4.setSpeed(0);
  Motor4.run(RELEASE);
 }
 }
 }
```

https://drive.google.com/file/d/1gRCj3dRQx6OK65BKgab0FJM4Ol-_j5Ki/view?usp=sharing

Chapter 5: Testing and Analysis.

5.1) Testing

Test Number	Test Name	Expected Result	Actual Result
01	Detection Range of RFID Reader and Tag	1m	1cm
02	Detection Range of IR sensor	1m	1cm
03	Detection Range of Ultrasonic Sensor	1m	10cm – 15cm
04	Battery life consumed by trolley	5hr	30mins

5.2) Analysis

Taking ***Test number 01*** the expected result of the rfid range was 1m but unfortunately the actual result was very low value. This difference is due to the RFID component used, as this is a prototype sample, we didn't use the RFID reader *UFD RFID CJ2503A* which are used by industrial companies because it's way more expensive than the RFID used by us. As a result, we get a huge difference between the results.

Taking ***Test number 2*** the expected and the actual results has a huge gap. It's because this is a prototype sample, because in industry they don't use IR sensors instead of IR they used LIDAR sensors or Laser Range sensors and when taking LIDAR sensors there are way more expensive than the IR sensor used by us. And the IR sensor used by us must be tune manually. This also effect the gap which is shown in the results.

Taking ***Test number 03*** there's a huge difference this is not due to the component used this is because when building the trolley, we can get an actual result as 1m but there will be a delay and when comparing the Ultrasonic sensor with the IR sensors, IR sensors has a range less than 1cm so if we code the Ultrasonic for 1m then there will be some problems like turning right and left. Therefore, we adjusted the Ultrasonic sensor for 10cm – 15cm range, that's why we get a huge difference between the results.

Taking ***Test number 04*** we expected a battery life for 5hrs but when we made the hardware and test, we got an actual result of battery life for 30mins. As this is a prototype, we cannot use more batteries because that will make the prototype heavier and it will be difficult to move for the trolley. So, as a result we used 2 li-ion batteries of1200mAh. Due to that we got the difference between Expected and Actual Results.

Chapter 6: Conclusion

6.1) Limitations

- The load carried by the trolley depends on the capacity of the battery and motor.
- The requirement of charging the battery of the trolley makes a demand in the trolley. It is also necessary to have a greater number of trolleys as the trolleys are not available every time specially during charging period.
- The human should be in close to the trolley so that the trolley will have the ability to detect the RFID tag and follow the human.
- There is need of maintaining and servicing the trolley.
- The RFID tags should be kept safe, it should not be misplaced.
- There is an additional cost compared to the manual trolley.

6.2) Future Improvements

In the future development of our smart trolley, we can design an which will have the categories of the items available in the mall. The map of the mall also will be fed to the memory in the trolley and as per the customer selection the trolley will be moved to that particular space.

Next, we can implement an in-built barcode in the trolley so that customer can check the product details or even possible to check out the product by patting online.

Then we can either upgrade the RFID tags or can add camera so that the trolley will accurately follow the specific customer. We also can develop wireless charging system for the trolley when it reaches the trolley parking.

The final development is the improvement in our app in which customer can add all the items that needs to purchased and when the customer is just strolling the mall and when he comes close to the object which he requires the trolley will give a notification to the customer. Based on customer health certain food will be recommended based on the health condition of the customer. These are the few developments that would be developed in the upcoming developments of the smart trolley for malls.

References

Akbari, M., 2019. *Electropeak.* [Online]
Available at: https://electropeak.com/learn/interfacing-us-100-ultrasonic-distance-sensor-with-arduino/
[Accessed 10 September 2022].

Anon., 2021. *Components101.* [Online]
Available at: https://components101.com/articles/what-is-motor-driver-h-bridge-topology-and-direction-control#:~:text=L293D%20and%20ULN2003%20are%20the,simple%20robots%20and%20RC%20cars.
[Accessed 13 september 2022].

Anon., 2021. *Utmel.* [Online]
Available at: https://www.utmel.com/components/a-comparison-article-about-l293d-uln2003?id=1979&sscid=91k6_d9pf4
[Accessed 13 september 2022].

Arduino, 2021. *store-usa.arduino.cc.* [Online]
Available at: https://store-usa.arduino.cc/products/arduino-mega-2560-rev3

Bello-Salau, H. O. A. M. D. I. R. A. C., 2021. A new automated smart cart system for modern shopping centres.. *Bulletin of Electrical Engineering and Informatics,* doi:10.11591/eei.v10i4.27(10(4)), p. 2028–2036. .

Bouakaz, S. V. M. B. C. M.-E. A. F. B. S. P. F. G. E. R. S. D. E. T. P. V. J.-P. a. C. T., 2014. In: *CIRDO: Smart Companion for Helping Elderly to Live at Home for Longer.* s.l.:IRBM, pp. 100-108.

Božek, P., 2014. Automated Detection Type Body and Shape Deformation for Robotic Welding Line. In: *Advances in Systems Science.* s.l.:s.n., pp. 229-240.

Danny, 2019. *Fierceelectronics.* [Online]
Available at: https://www.fierceelectronics.com/sensors/what-ultrasonic-sensor
[Accessed 10 september 2022].

Dejan, 2015. *Howtomechatronics.* [Online]
Available at: https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/
[Accessed 10 September 2022].

EE, 2020. *Servo Motor.* [Online]
Available at: http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf
[Accessed 10 09 2022].

Hokuyo, 2020. *Autonomoustuff.* [Online]
Available at: http://www.autonomoustuff.com/hokuyo-urg-041x-ug01.html
[Accessed 10 9 2022].

K.GOH, 2016. *semanticscholar.* [Online]
Available at: https://www.semanticscholar.org/paper/Human-Following-Robot-using-Image-Processing-for-

Goh/d370d90961f15a89e4bf059cb1bedf16d86be22c
[Accessed 10 9 2022].

Kazuyuki. M., L. J.-H. a. H. H., 2004. Human-Following Mobile Robot in a Distributed Intelligent Sensor Network. In: *IEEE Transactions on Industrial Electronics.* s.l.:s.n., pp. 229-237.

Leroux, C. L. O. B. G. M. W. M. Y. D. L. C. C. M. J.-C. ,. L. V. a. F. C., 2013. In: *ARMEN: Assistive Robotics to Maintain Elderly People In Natural Environment.* s.l.:IRBM, pp. 101-107.

manegment, P., 2017. *Paymoapp.com.* [Online]
Available at: https://www.paymoapp.com/blog/project-management-methodologies/#:~:text=Essentially%2C%20a%20methodology%20is%20a,of%20the%20project's%20life%20cycle.
[Accessed 28 9 2022].

Ng, Y. L., 2015. *Research Gate.* [Online]
Available at:
https://www.researchgate.net/publication/275222428_Automatic_Human_Guided_Shopping_Trolley_with_Smart_Shopping_System
[Accessed 29 August 2022].

Nicho, 2020. *Seeedstudio.* [Online]
Available at: https://www.seeedstudio.com/blog/2020/05/29/which-is-best-rfid-nfc-mifare-chip-selection-guide-rc522-vs-pn532/
[Accessed 9 September 2022].

R.Sonawane, A. C. H., 2021. Smart trolley with human follower. *International Research Journal of Engineering and Technology,* I(8), pp. 1799-1807.

Researchgate, 2014. *Researchgate.net.* [Online]
Available at:
https://www.researchgate.net/publication/269866695_Human_Detecting_and_Following_Mobile_Robot_Using_a_Laser_Range_Sensor
[Accessed 28 9 2022].

Sayali N Joshi, V. K. P. P. S. D. a. H. K. B., 2019. Design and Development of Human Following Trolley.. *International Journal of Innovative Science and Research Technology,* IV(4), pp. -.

Scribd, Copyright © 2022. *Scribd.* [Online]
Available at: https://www.scribd.com/document/514822265/REPORT-HUMAN-FOLLOWING-ROBOT
[Accessed 27 08 2022].

Teknologi, J., 2008-2022. *ResearchGate.* [Online]
Available at:
https://www.researchgate.net/publication/275222428_Automatic_Human_Guided_Shopping_Trolley_with_Smart_Shopping_System
[Accessed 27 08 2022].

Workshop, 2020. *Dronebotworkshop.* [Online]
Available at: https://dronebotworkshop.com/laser-vs-ultrasonic-distance-sensor-tests/
[Accessed 10 9 2022].

Wrike, 2006-2022. *Wrike.* [Online]
Available at: https://www.wrike.com/professional-services-guide/project-definition/
[Accessed 27 08 2022].

Wu, Y.-H. W. J. C.-L. V. K. L. C. M. D. D. L. P. B. J. C. D. V. G. M. a. R. A.-S., 2013. In: *Designing an Assistive Robot for Older Adults: The ROBADOM Project.* s.l.:IRBM, pp. 119-123.