International Journal of Scientific Research in Science, Engineering and Technology
Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com)
doi : https://doi.org/10.32628/IJSRSET229627

The Intelligent Robot for Serving Food

Zainab Khyioon Abdalrdha [*1], Noor Ali Hussein[*2], Raghda Sattar Jabar [*3]
[*1]Assistant Professor, [*2] Assistant Programmer, [*3]Assistant Teacher
[*1]Department of Computer Science, College of Basic Education, Mustansiriyah University, Bagdad-Iraq
, [*2] Mustansiriyah University/ College of Education/ Avi Cenna, Bagdad-Iraq
[*3]Department of Quality Assurance, Mustansiriyah University, Bagdad-Iraq

Article Info
Volume 9, Issue 6
Page Number : 273-283

Publication Issue :
November-December-2022

Article History
Accepted : 20 Nov 2022
Published: 05 Dec 2022

ABSTRACT

All over the world, robotics technology is rapidly replacing manuals of work. Customers in traditional cafes, restaurants, and hotels are facing a variety of problems due to crowding during peak hours, lack of waiters, manual order processing, and the Covid-19 pandemic. The purpose of this paper is to discuss where a waiter robot can be designed and developed that can be used for restaurant automation. Where we rely on a robot to address these defects using black tracking plans, where the bot can track the movement of the black line of the tables, and then the bot reaches the table to place orders, where the customer presses the order through the center. The program is through a bank machine, and a tablet is placed on each table. The program is installed in it to choose the order, and after selecting the order by the customer, this is on the part of the customer upon request. After the request, a lamp lights near the cashier with a table that wants a specific service, and this is according to the existing lamp counter where there is a board near the cashier in which foods and juices are placed and it consists of (1-8) A lamp and a list of foods or juices are indexed by each one separately, and the service will be transferred via the smart car according to the requests requested by the customer. Thus, this paper presents a method to protect people from this virus and not transmit it to others, and the paper suggested was programmed according to the Arduino environment and the Bluetooth module.

Keywords : Arduino UNO R3, hardware platforms, drive motor, IR (Transmitter and Receiver) Sensor, HC-05–Bluetooth.

I. INTRODUCTION

In today's world, the advancement of scientific methodologies and technological innovation has made it possible to apply autonomy from traditional robotics to application-specific robots. Hourly wages will increase dramatically. There is less of a need for young people to work as waiters in hotels and fast food establishments as the global information technology sector grows. Over the past few decades, there has been an increase in interest in research and development initiatives for social services and healthcare. One area of robotics called "social robotics" is used to benefit people [1]. In the modern world, social robots can communicate with people, participate in society in all its facets, and comprehend social concepts [2]. The development of robotic technologies that can read human thoughts and interpret actions has led to the introduction of many novel designs and procedures. Such robots are used in assistive robotics, for instance, to help the elderly, the ill, and injured people [3]. Due to their adaptability, these robots can operate in a variety of ways, depending on the circumstance [4]. Robots have learned from us thus far, but the day will come when the teacher will also be the student. Robotic automation in restaurants is becoming more and more common [5]. The following is how this paper is structured: The section that follows presents the work's purpose, the part third that follows presents the design of an intelligent robot that serves food, part (4) that follows presents the proposed system, part (5) follows presents the proposed system's architecture, the section (6) that follows presents the proposed system's implementation, and the section (8) that follows presents the proposed system's conclusions.

II. This work's purpose

A waiter robot that could be used in restaurants, where robotics technology is quickly replacing manual labor, is the aim of this study. Due to peak-hour crowding, a lack of servers, manual order processing, as well as other factors, such as the existence of the Covid 19 pandemic, which has caused the virus to spread, customers in restaurants and hotels deal with a variety of problems. This project relies on its design on the Arduino environment, black lines for tracking, and Bluetooth to transfer and deliver services from clients to the waiter to meet the expectations. The customer places his or her order by pressing a button, then proceeds to the next table to record the orders of the other customers, and so on.

III. Design of Intelligent robot for serving food

In this section, we'll go through the tools that were used to create this Intelligent robot for serving food, the hardware components that were used to set up it, and the software components that were utilized to program it.

Hardware Components:

The Intelligent robot for serving food with Arduino has various prerequisites, the most significant of which are as follows:

1. UNO R3 (Arduino)
2. Bluetooth
3. drive
4. USB cord
5. 9-volt power supply
6. CAR robot
7. Switch
8. Sensor for infrared light.

Software Components:

Integrated development environment (IDE) for open source software: The IDE makes it simple to write code and submit it to the board. IDEs have long been popular on the Mac, OS, and Mac OS, which support multi-language development. Additionally, to fully functional software for Bluetooth Control Lamp and Bluetooth Remote Controller. As mentioned earlier, a smart food service robot contains hardware components. Now we will discuss each of them in a simplified manner.

• **Arduino Uno R3 Overview:** It is currently in its third and latest revision or R3. Using ATmega328 as its basis, and it is a microcontroller board. The ATmega328 contains 32 KB. It includes two types of memory: 2 KB SRAM and 1 KB EEPROM. It contains a reset button, an ICSP connector, a power jack, a USB port, and a place for Circuit System Programming (ICSP). In addition, it has 20 digital I/O pins, 6 of which can be used as PWM outputs and 6 as analog inputs [6]. The AVR microcontroller chips are the basis of the Arduino board, which uses about 80mA of 5V current when unplugged or disconnected. Arduino has a clock speed of 16MHz, which enables it to complete a task more quickly than another processor or controller. The AVR chip runs continuously at 16MHz no matter what the code is doing; It never stops, so the code you give it to run has very little effect on how much current it uses. The only way to stop code execution and greatly reduce the current consumption of the AVR chip is to put it into one of the "sleep modes". All other components on the Uno will continue to draw their usual amounts of current. In addition, there are no "sleep mode" examples provided by Arduino, so one has to find other coding examples provided by the user. I2C and SPI communication are supported by the Arduino board. The Arduino software comes with an I2C wiring library and an SPI library for SPI connections. Arduino Uno R3 comes with the components shown in Figure 1 [7]

Figure 1: Arduino Uno R3.

- **HC-05–Bluetooth:** Bluetooth technology, which enables wireless connections to both mobile phones and home devices, is a standard feature of mobile phones. Controlling your home automation is easy with the help of Bluetooth technology. The technology is safe and affordable. The system uses the Bluetooth Arduino board. The user interface for the cell phone is provided by the Python program. For a distance of 10 meters and a speed of 1 Mbps, the frequency band is more than 2.4 GHz ISM [8]. Figure (2) shows in this part, HC-05 can be used to establish communication between MCU and PC for data transmission because it provides good wireless transmission and good reception serial data.

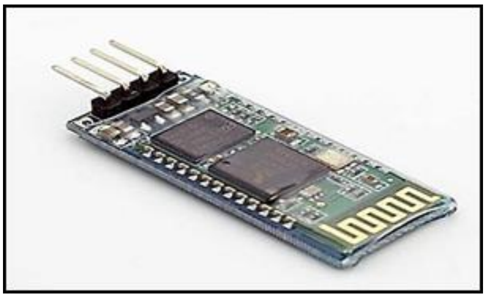

Figure 2 : Bluetooth module HC-05

- **Drive motor:** Easily control the speed and direction of two DC motors or a bipolar stepper motor through an Arduino-compatible L298N H-bridge dual motor controller with 2A. Motors whose voltage is between 5 and 35 VDC can be used with the L298N H bridge module [9]. In addition, a 5V regulator is built into the device, allowing you to take a 5V supply from the board in Figure 3 if your supply voltage is up to 12V.

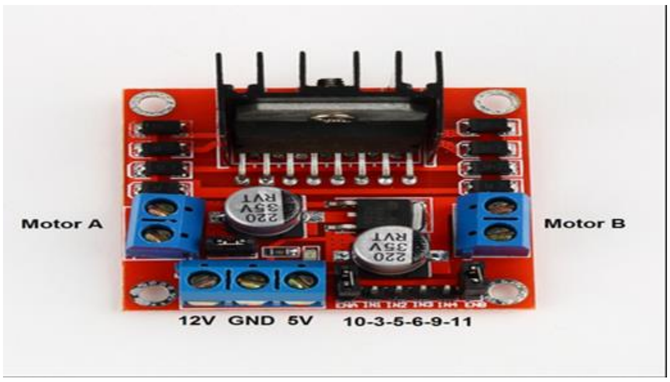

Figure. 3 drive motor

- **IR-sensor:** In addition to detecting motion, an IR sensor can gauge an object's heat. As opposed to emitting it, these kinds of sensors are referred to as passive IR sensors because they only measure infrared radiation. The human presence is identified using a passive infrared (PIR) sensor. By utilizing the infrared radiation that a person's body emits, the Grid-EYE sensor can identify them. A certain waveband of infrared energy is emitted by every human being. The temperature of the material changes as a result of the absorption of incident radiation [10].

- **Power Unit:** The entire system is powered by a 12V-12Ah battery power supply.
- **Battery Charger:** The battery can be fully charged in 5 hours using a 12V-20Ah rechargeable battery circuit [11].

IV. The proposed system:

The waiter robot was designed and developed in the Arduino environment, which served as the foundation for the proposed system in this paper. This waiter robot could be used to automate restaurants, where customers encounter numerous issues due to peak-hour traffic, waiter shortages, manual order processing, and the Covid-19 pandemic. Where we rely on a robot that tracks the movement of the black line of the tables, and then the robot arrives at the table to submit requests, Whereas the customer presses the request through a position program through their bank account, and a tablet is placed above each table in which the program is installed to choose the order, and after the customer selects the request, the signal is near, we rely on a robot that tracks the movement of the black line of the tables, and the robot arrives at the table to submit requests. A board near the cashier has a list of the foods and juices that are indexed from 1 to 8 and is where the table that is seeking specific service from the cashier is located, as indicated by the lighted color, depending on the requests made by the client, the service is transferred via the smart car for each one individually.

V. System Architecture of the Proposed System.

This section demonstrates the system's proposed general architecture as shown in figures (4), (5. a), and (5. b), and the block diagram for a waiter robot for restaurants that uses an Arduino Uno (6). Manual labor is being replaced by robots. manually operated cafe systems. Several issues are readily apparent. The robot waiter is a novel idea, and many fast food chains could automate their restaurants using it. A line that follows the robot for which the sensors are used is the waiter robot. There are three sections to the project. The application on the menu bar, the robot that performs the service delivery, and the board containing the requests selected by the customer are the three main components. These requests are presented in the form of lambs numbered 1 through 8, each of which stands for a different type of food, such as chicken, chicken shawarma, juice, meat shawarma, appetizers, etc.

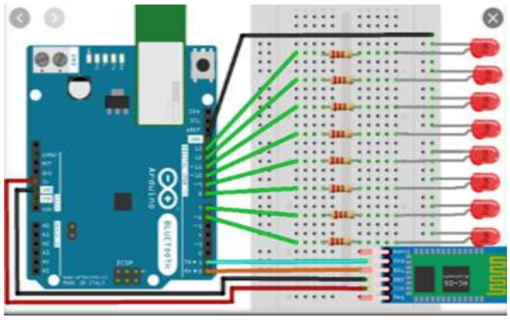

Figure .4 Connect 8 LEDs on the Arduino for the menu bar.

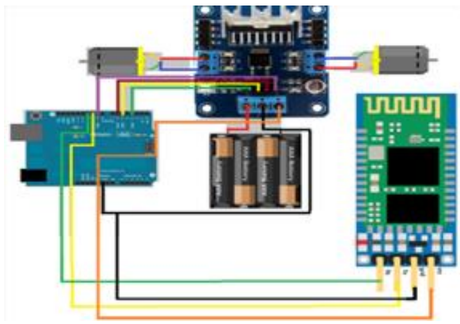

Figure.5. a robot car circuit diagram.

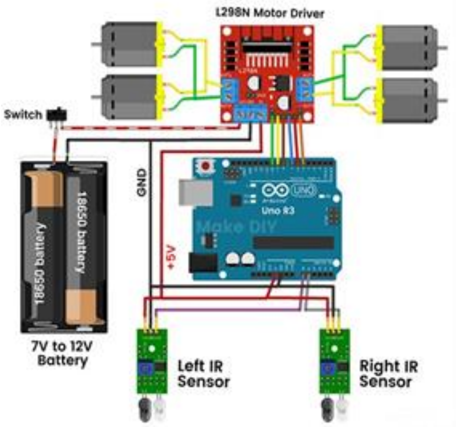

Figure .5. b a robot car circuit diagram for connecting an IR sensor

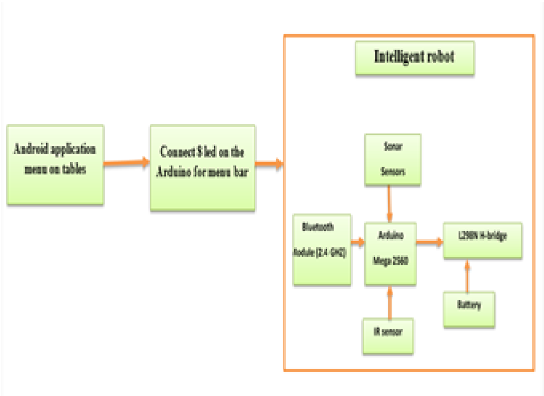

Figure 6. Block diagram for an Arduino-powered waiter robot for restaurants

VI. The Execution of the Proposed System

Now we are installing the application for this project on the smartphone then we open the program and connect to Bluetooth and we run the car via smartphone Software. In this paper, we used the Arduino program through which the search code is written This program is downloaded from the official website of Arduino the code is written in the IDE programming language.

Once the device is turned on, we will connect to and control the car robot's Bluetooth and then run the program that will control the car, as shown in Figure (7).

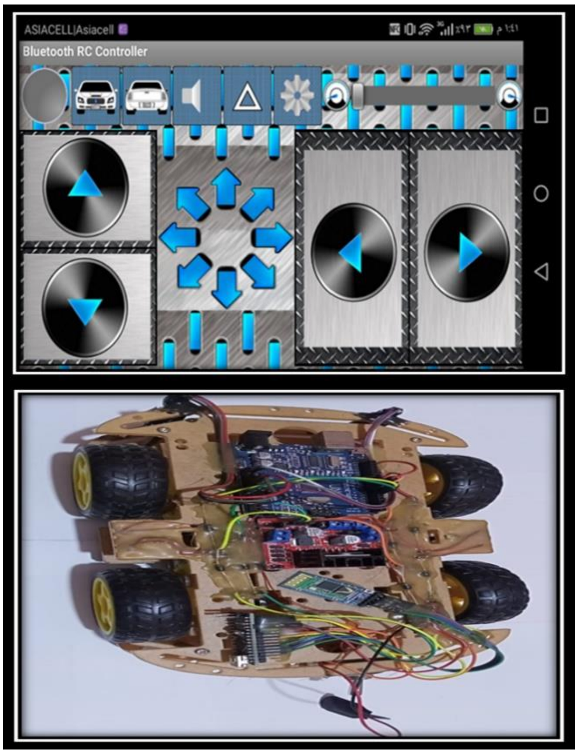

Figure .7 controls the vehicle's Bluetooth for the car& car robot

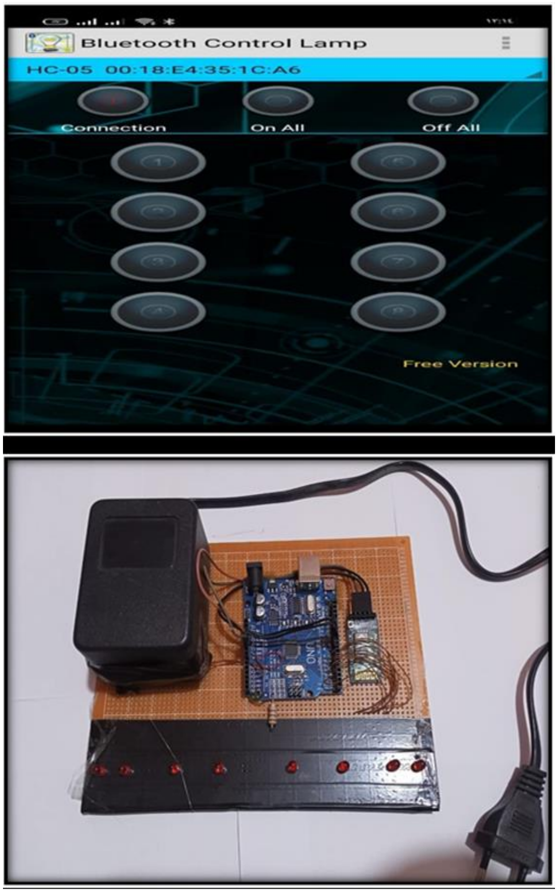

Figure .8 Bluetooth menu bar & menu bar control lamp

VII. Executing the Robot Waiter Robot for Restaurants code

```
#include "SoftwareSerial.h"// import the serial library\
const int IN1=3;
const int IN2=5;
const int IN3=6;
const int IN4=9;
 int Bluetooth Data;
 Software Serial HC05(10, 11); // RX, TX
 int SPEED_Control=200;
void setup () {
// fIRST, define the Motor's pin as an OUTPUT
pin Mode (IN1, OUTPUT) ;// Right Motor 1st wire
pin Mode (IN2, OUTPUT) ;// Right Motor 2nd wire
pin Mode (IN3, OUTPUT) ;// left Motor 1st wire
pin Mode (IN4, OUTPUT) ;// left Motor 2nd wire
 HC05.begin(9600);
}
void FORWARD (int Speed) {
  //When we want to let Motor to Rotate clock wise
  // just void this part on the loop section.

  analog Write(IN1,0);
  analog Write (IN2, Speed);
  analog Write (IN3, Speed);
  analog Write(IN4,0);
}
void BACKWARD (int Speed) {
  //When we want to let Motor to Rotate Counter clock wise
  // just void this part on the loop section.
  analog Write (IN1, Speed);
  analog Write(IN2,0);
  analog Write(IN3,0);
  analog Write (IN4, Speed);
}
void LEFT (int Speed) {
  //When we want to let Motor to Rotate Counter clock wise
  // just void this part on the loop section.
  analog Write(IN1,0);
  analog Write (IN2, Speed);
  analog Write(IN3,0);
  analog Write (IN4, Speed);
}
void RIGHT (int Speed) {
  //When we want to let Motor to Rotate Counter clock wise
  // just void this part on the loop section.
  analog Write (IN1, Speed);
  analog Write(IN2,0);
  analog Write (IN3, Speed);
  analog Write(IN4,0);
}
void Stop () {
  //When we want to let Motor to Rotate clock wise
  // just void this part on the loop section.
  analog Write(IN1,0);
  analog Write(IN2,0);
  analog Write(IN3,0);
  analog Write(IN4,0);
```

```
}
void loop () {
 //Rise Up
  if (HC05.available()) {
 Bluetooth Data=HC05.read();
 Serial.println(Bluetooth Data);
  if (Bluetooth Data=='F') {   // if number 1 pressed ....
   FORWARD(SPEED_Control);
   }
  if (Bluetooth Data=='B') {   // if number 1 pressed ....
  BACKWARD(SPEED_Control);
   }
  if (Bluetooth Data=='L') {   // if number 1 pressed ....
  RIGHT(SPEED_Control);
   }
  if (Bluetooth Data=='R') {   // if number 1 pressed ....
   LEFT(SPEED_Control);
   }
    if (Bluetooth Data=='S') {   // if number 1 pressed ....
  Stop ();
   }
   if (Bluetooth Data=='0') {SPEED_Control=0;} //Speed
   if (Bluetooth Data=='1') {SPEED_Control=50;}//Speed
   if (Bluetooth Data=='2') {SPEED_Control=100; }//Speed
   if (Bluetooth Data=='3') {SPEED_Control=120;}//Speed
   if (Bluetooth Data=='4') {SPEED_Control=140;}//Speed
   if (Bluetooth Data=='5') {SPEED_Control=160;}//Speed
   if (Bluetooth Data=='6') {SPEED_Control=180;}//Speed
   if (Bluetooth Data=='7') {SPEED_Control=200;}//Speed
   if (Bluetooth Data=='8') {SPEED_Control=220;}//Speed
   if (Bluetooth Data=='9') {SPEED_Control=240;}//Speed
   if (Bluetooth Data=='q') {SPEED_Control=255;}//Speed
  }
}
```

Code Bluetooth

```
#include <SoftwareSerial.h>
Software Serial my Serial (10, 11); //Pin10 TX, pin 11 RX connected to-->
Bluetooth TX, RX
#define relay1 2
#define relay2 3
#define relay3 4
#define relay4 5
#define relay5 6
#define relay6 7
#define relay7 8
#define relay8 9
char Val;
void setup () {
  pin Mode (relay1, OUTPUT);
  pin Mode (relay2, OUTPUT);
  pin Mode (relay3, OUTPUT);
  pin Mode (relay4, OUTPUT);
  pin Mode (relay5, OUTPUT);
  pin Mode (relay6, OUTPUT);
  pin Mode (relay7, OUTPUT);
  pin Mode (relay8, OUTPUT);
```

```
 digital Write (relay1, HIGH);
 digital Write (relay2, HIGH);
 digital Write (relay3, HIGH);
 digital Write (relay4, HIGH);
 digitalWrite (relay5, HIGH);
 digitalWrite (relay6, HIGH);
 digitalWrite (relay7, HIGH);
 digitalWrite (relay8, HIGH);
 mySerial.begin(9600);
 Serial. Begin (9600);
}
void loop () {
//cek data serial from Bluetooth android App
if (mySerial.available() >0) {
  Val = mySerial.read();
  Serial.println(Val);
}
//Relay is on
 if (Val == 'A') {
  digitalWrite (relay1, HIGH);}
 else if (Val == 'B') {
  digitalWrite (relay2, HIGH);}
 else if (Val == 'C') {
  digitalWrite (relay3, HIGH);}
 else if (Val == 'D') {
  digitalWrite (relay4, HIGH);}
else if (Val == 'E') {
  digitalWrite (relay5, HIGH);}
 else if (Val == 'F') {
  digitalWrite (relay6, HIGH);}
 else if (Val == 'G') {
  digitalWrite (relay7, HIGH);}
else if (Val == 'H') {
  digitalWrite (relay8, HIGH);}

//relay is off
 if (Val == '1') {
  digitalWrite (relay1, LOW);}
 else if (Val == '2') {
  digitalWrite (relay2, LOW);}
 else if (Val == '3') {
  digitalWrite (relay3, LOW);}
 else if (Val == '4') {
  digitalWrite (relay4, LOW);}
else if (Val == '5') {
  digitalWrite (relay5, LOW);}
 else if (Val == '6') {
  digitalWrite (relay6, LOW);}
 else if (Val == '7') {
  digitalWrite (relay7, LOW);}
else if (Val == '8') {
  digitalWrite (relay8, LOW);}
 if (Val =='9') {
  digitalWrite (relay1, LOW);
  digitalWrite (relay2, LOW);
  digitalWrite (relay3, LOW);
```

```
    digitalWrite (relay4, LOW);
    digitalWrite (relay5, LOW);
    digitalWrite (relay6, LOW);
    digitalWrite (relay7, LOW);
    digitalWrite (relay8, LOW);}
if (Val =='I') {
digitalWrite (relay1, HIGH);
    digitalWrite (relay2, HIGH);
    digitalWrite (relay3, HIGH);
    digitalWrite (relay4, HIGH);
    digitalWrite (relay5, HIGH);
    digitalWrite (relay6, HIGH);
    digitalWrite (relay7, HIGH);
    digitalWrite (relay8, HIGH);}}
```

code tracker line: -

```
#define in1 9
#define in2 8
#define in3 7
#define in4 6
#define enA 10
#define enB 5
 int M1_Speed = 80; // speed of motor 1
 int M2_Speed = 80; // speed of motor 2
 int LeftRotationSpeed = 250; // Left Rotation Speed
 int RightRotationSpeed = 250; // Right Rotation Speed
 void setup () {
 pin Mode (in1, OUTPUT);
 pin Mode (in2, OUTPUT);
 pin Mode (in3, OUTPUT);
 pin Mode (in4, OUTPUT);
  pin Mode (enA, OUTPUT);
  pin Mode (enB, OUTPUT);
   pin Mode (A0, INPUT); // initialize Left sensor as an input
   pin Mode (A1, INPUT); // initialize Right sensor as an input
}
void loop () {
 int LEFT_SENSOR = digital Read(A0);
 int RIGHT_SENSOR = digital Read(A1);
if (RIGHT_SENSOR==0 && LEFT_SENSOR==0) {
  forward (); //FORWARD
}
 else if (RIGHT_SENSOR==0 && LEFT_SENSOR==1) {
  right (); //Move Right
 }
 else if (RIGHT_SENSOR==1 && LEFT_SENSOR==0) {
  left (); //Move Left
}
 else if (RIGHT_SENSOR==1 && LEFT_SENSOR==1) {
  Stop (); //STOP
 }
}
void forward ()
{
     digitalWrite (in1, HIGH);
     digitalWrite (in2, LOW);
     digitalWrite (in3, HIGH);
```

```
    digitalWrite (in4, LOW);
     analog Write (enA, M1_Speed);
     analog Write (enB, M2_Speed);
}
void backward ()
{
    digitalWrite (in1, LOW);
    digitalWrite (in2, HIGH);
    digitalWrite (in3, LOW);
    digitalWrite (in4, HIGH);
     analog Write (enA, M1_Speed);
     analog Write (enB, M2_Speed);
}

void right ()
{
    digitalWrite (in1, LOW);
    digitalWrite (in2, HIGH);
    digitalWrite (in3, HIGH);
    digitalWrite (in4, LOW);
     analog Write (enA, LeftRotationSpeed);
     analog Write (enB, RightRotationSpeed);
}
void left ()
{
    digitalWrite (in1, HIGH);
    digitalWrite (in2, LOW);
    digitalWrite (in3, LOW);
    digitalWrite (in4, HIGH);
     analog Write (enA, LeftRotationSpeed);
     analog Write (enB, RightRotationSpeed);
}
void Stop ()
{
    digital Write (in1, LOW);
    digital Write (in2, LOW);
    digital Write (in3, LOW);
    digital Write (in4, LOW);
}
```

VIII. CONCLUSION

The following are the main findings from the development and application of the suggested system:

1. This suggested system is compatible with C (Arduino)
2. The robotic waiter system is a concept for a design that combines mobile, autonomous, omnidirectional robots with an ecosystem to provide the robots with information about their surroundings. This makes it possible to quickly translate research into practice.
3. The robots are made to be effective and efficient substitutes for human labor in situations where there is a shortage of workers for routine, repetitive tasks like bringing food to the table. Additionally, the robot's capacity to hold multiple orders significantly reduced the workload of a waiter during busy times.
4. Robots are made to be a productive and efficient alternative, especially in the wake of the COVID-19 pandemic, to safeguard customers and waitstaff from the spread of the virus.

IX. ACKNOWLEDGMENT

The authors would like to thank AL_Mustansiriyah University (www.uomusiriyah.edu.iq), Baghdad-Iraq for its support in the present work.

X. REFERENCES

[1]. David Kushner (2011-10-26). "The Making of Arduino". IEEE Spectrum.

[2]. Justin Lahart (27 November 2009). "Taking an Open-Source Approach to Hardware". The Wall Street Journal. Retrieved 2014-09-07.

[3]. Biggs, John. "CEO controversy mars Arduino's open future". TechCrunch. Retrieved 2017-12-22.

[4]. M. Asif, M. Sabeel, Mujeeb-ur-Rahman, and Z. H. Khan, " Waiter Robot – Solution to Restaurant Automation". MDSRC - 2015 Proceedings, 14-15 November 2015 Wah/Pakistan

[5]. Hernando Barragán (2016-01-01). "The Untold History of Arduino". arduinohistory.github.io. Retrieved 2016-03-06.

[6]. Makthal Goverdhan Rao1, Abbugari Hemant Raj2 ,and Boggarapu Sai Rohit3, etc,"ARDUINO BASED ANIMATRONIC HAND", International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 05 | May 2020, http link https://www.irjet.net/archives/V7/i5/IRJET-V7I51063.pdf

[7]. Sonali Sen , Shamik Chakrabarty , Raghav Toshniwal, Ankit Bhaumik," Design of an Intelligent Voice Controlled Home Automation System ", International Journal of Computer Applications, Vol. 121, No.15, Jul. 2015.

[8]. Prashant R. Chandre, Aarti R. Salunke," Home Automation using Android Application & Bluetooth", International Journal on Recent and Innovation Trends in Computing and Communication, Volume:3,No. 2, Feb. 2015.

[9]. Dickson Neoh Tze How, Mohd Zafri Baharuddin*, Syed Sulaiman Kaja Mohideen, "Modular Motor Driver with Torque Control for Gripping Mechanism", Procedia Engineering 41 (2012) 1476 – 1482, 1877-7058 © 2012 Published by Elsevier Ltd. doi: 10.1016/j.proeng.2012.07.338

[10]. Pooja Ajmera, "A Review Paper on Infrared sensor", International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Published by, www.ijert.org VIMPACT - 2017 Conference Proceedings

[11]. Vo Nhu Thanh," DEVELOPMENT OF RESTAURANT SERVING ROBOT USING LINE FOLLOWING APPROACH", ISSN 1859-1531 - THE UNIVERSITY OF DANANG, JOURNAL OF SCIENCE AND TECHNOLOGY, VOL. 17, NO. 12.1, 2019.

Cite this article as :

Zainab Khyioon Abdalrdha, Noor Ali Hussein, Raghda Sattar Jabar, "The Intelligent Robot for Serving Food ", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Online ISSN : 2394-4099, Print ISSN : 2395-1990, Volume 9 Issue 6, pp. 273-283, November-December 2022. Available at doi : https://doi.org/10.32628/IJSRSET229627
Journal URL : https://ijsrset.com/IJSRSET229627