



جامعة الأمير محمد بن فهد PRINCE MOHAMMAD BIN FAHD UNIVERSITY

College of Engineering

Department of Electrical Engineering

Fall 2019-2020

Assessment III Project Report

Restaurant Service System (RSS)

In a fulfillment of the requirements for the
Degree of Bachelor of Science in Electrical Engineering

Team Members

	Student Name	Student ID
1	Abdullah Almakhaitah	201400610
2	Hamid Bin Hilal	201400673
3	Ahmed Al Turki	201402385
4	Abdilrahman Yusuf	201500647
5	Ali Al Ali	201402853

Project Advisors

Advisor Name: **Dr. Sadiq Alhuwaidi**

Abstract

This project is about designing a robot that provides typical services to a restaurant. The robot consists of subsystems to have it act as a waiter delivering the meals to a designated table. The structure of restaurant service system (RSS) enables the robot to carry the food in a plate from the kitchen after receiving the order through a created ordering system to the assigned table using the line following technique and through the use of a pixy camera. Throughout two phases which are Design Methodology and Assessment III courses at Prince Mohammad bin Fahd University, the team members are able to complete the task successfully starting from literature search, acquiring components, and testing, integrating and assembling the subsystems.

Table of Contents

1. INTRODUCTION	7
1.1 PROJECT DEFINITION	7
1.2 PROJECT OBJECTIVES	7
1.3 PROJECT SPECIFICATIONS	7
1.4 PRODUCT ARCHITECTURE AND COMPONENTS	8
1.5 APPLICATIONS	9
2. LITERATURE REVIEW	9
2.1 PROJECT BACKGROUND	9
2.2 PREVIOUS WORK	10
2.3 COMPARATIVE STUDY	13
3. SYSTEM DESIGN	14
3.1 DESIGN CONSTRAINTS	14
3.2 DESIGN METHODOLOGY	16
3.3 PRODUCT SUBSYSTEMS AND COMPONENTS	16
3.3.1 PRODUCT SUBSYSTEM 1: MOTOR CONTROL	17
3.3.2 PRODUCT SUBSYSTEM 2: WIRELESS	20
3.3.3 PRODUCT SUBSYSTEM 3: SENSORS	24
3.3.4 PRODUCT SUBSYSTEM 1: ANDROID APP	27
3.4 IMPLEMENTATION	27
4. SYSTEM TESTING AND ANALYSIS	29
4.1 SUBSYSTEM 1: LINE FOLLOWING & MOTOR CONTROL	29
4.2 OVERALL RESULTS, ANALYSIS, AND DISCUSSION	29
4.3 ASSESSMENT III	30
4.4 FINAL PROTOTYPE	37
5. PROJECT MANAGEMENT	40
5.1 PROJECT PLAN	40
5.2 CONTRIBUTION OF TEAM MEMBERS	42
5.3 PROJECT EXECUTION MONITORING	42
5.4 CHALLENGES AND DECISION MAKING	43
5.5 PROJECT BILL OF MATERIALS AND BUDGET	44
6. PROJECT ANALYSIS	44
6.1 LIFE-LONG LEARNING	44
6.2 IMPACT OF ENGINEERING SOLUTIONS	45

6.3 CONTEMPORARY ISSUES ADDRESSED	45
<u>7. CONCLUSIONS AND FUTURE RECOMMENDATIONS.</u>	<u>46</u>
7.1 CONCLUSIONS.....	46
7.2 FUTURE RECOMMENDATIONS	46
<u>8. REFERENCES.....</u>	<u>47</u>
<u>APPENDIX A: PROGRESS REPORTS</u>	<u>49</u>
<u>APPENDIX B: DATASHEETS</u>	<u>51</u>
<u>APPENDIX C: OPERATION MANUAL</u>	<u>53</u>
<u>APPENDIX D: PROGRAM CODES</u>	<u>54</u>

Table of Figures

Figure 1: Product architecture of the restaurant service system (RSS).	8
Figure 2: Old chassis of the restaurant service system (RSS).	9
Figure 3: Robot prototype for waiter service by the Department of Electrical and Electronic Engineering , BRAC.	11
Figure 4: Robotic waiter for restaurants implemented in Design Document course (ECE 445).	12
Figure 5: Categories of the restaurant service system (RSS).	16
Figure 6: subsystems and components of the restaurant service system (RSS).	17
Figure 7: Subsystem 1 connections of the motor and motor control.	17
Figure 8: Motor used in the restaurant service system (RSS) project.	18
Figure 9: Motor controller used in the restaurant service system (RSS) project.	19
Figure 10: The Bluetooth connections in subsystem 2.	21
Figure 11: IR Receiver connections in subsystem 2.	21
Figure 12: HC 05 Bluetooth module.	22
Figure 13: IR receiver used in subsystem 2 of the restaurant service system (RSS) project.	23
Figure 14: Ultrasonic connections in subsystem 3 of the restaurant service system (RSS) project.	24
Figure 15: IR line following sensor connections in subsystem 3 of the restaurant service system (RSS) project.	25
Figure 16: SRF 05 ultrasonic sensor used in subsystem 3 of the restaurant service system (RSS) project.	25
Figure 17: IR line following sensor used in subsystem 3 of the restaurant service system (RSS) project.	26
Figure 18: System connection of the ordering system used in the restaurant service system (RSS) project.	27
Figure 19: System implementations flowchart of the restaurant service system in Assessment III course.	28
Figure 20: Serial monitor of the Arduino of the restaurant service system (RSS) project.	30
Figure 21: Modified subsystems of the restaurant service system (RSS) project.	31
Figure 22: New chassis of the restaurant service system (RSS).	31
Figure 23: Motor controller of the restaurant service system (RSS).	32
Figure 24: Johnson motor of the restaurant service system (RSS).	32
Figure 25: Pixy cam used in the restaurant service system (RSS) project.	33
Figure 26: Table detection sensor used in the restaurant service system (RSS).	33
Figure 27: Ordering system used in the restaurant service system (RSS).	34
Figure 28: Welcoming and ordering page of the ordering system used in the restaurant service system (RSS).	35
Figure 29: Coupons and other features of the ordering system in the restaurant service system (RSS).	36
Figure 30: Receiving order in the kitchen of the restaurant service system (RSS).	36
Figure 31: Integrated subsystem in the chassis of the restaurant service system (RSS).	37
Figure 32: Final prototype of the restaurant service (RSS) robot.	38
Figure 33: Final prototype of the restaurant service system (RSS) with the team members.	38
Figure 34: Final prototype of the restaurant service system (RSS) while functioning to go to a designated table.	39
Figure 35: RSS robot serving some customers during the Electrical Engineering Demo day.	39
Figure 36: A suggested restaurant layout of the RSS robot.	40
Figure 37: Final progress report during Design Methodology course of the restaurant service system (RSS).	41
Figure 38: Final progress report during Assessment III course of the restaurant service system (RSS).	41
Figure 39: In between progress report 1.	49
Figure 40: In between progress report 2.	49
Figure 41: In between progress report 3.	50
Figure 42: Arduino pinout.	51
Figure 43: IR sensor datasheet.	52

Table of Tables

Table 1: Comparison between our robot and previous robot discussed in Section 2.2.....	13
Table 2: Motor specifications used in the restaurant service system (RSS) project.	18
Table 3: Reason for selecting each component in the RSS robot.	34
Table 4: Contribution of team members to implement the RSS robot.....	42
Table 5: Total cost of the RSS robot.....	44

1. Introduction

1.1 Project Definition

This project is a smart restaurant system that simplifies the process of ordering food. It combines a robot that is mobile and goes around tables in the restaurant to deliver the orders and an ordering system that is used by the customers to place their respective orders through android application. The project is named Restaurant Service System (RSS).

1.2 Project Objectives

The objectives of the project are the following:

1. Autonomous meal delivery robot that carries the meals from the restaurant kitchen to the designated tables.
2. The order from the robot via tablet then, relayed back to the restaurant kitchen.
3. Orders when prepared are dispatched from the restaurant kitchen to the designated tables.

1.3 Project Specifications

The functioning project specifications are:

1. Robot stops at the designated table using IR sensor.
2. Line detection by pixy cam (red line).
3. Table delivery based on voice command.
4. Obstacle avoidance using ultrasonic.
5. Robot can return to the kitchen by waving.
6. Ordering system (an Android App).

From the metrics standing point, the estimated speed of the robot would be a maximum of 3 km/h and minimum of 1 km/h. The expected load it can handle is around 10 kg. It uses a voltage of 12 V DC and lasts for 6 hours of operation without charging. The expected number of meals it can serve within a full hour is 30, but it depends on the restaurant space and table spots.

From the marketing standing point, the robot must be easy to use. It is a fully integrated system which can avoid obstacles easily, follows the path appropriately using the pixy cam and line following, knows the designated table with the IR subsystem, and is efficient cost-wise.

1.4 Product Architecture and Components

Figure 1 shows the product architecture of the restaurant service system (RSS).



Figure 1: Product architecture of the restaurant service system (RSS).

Figure 2 demonstrates the old chassis of the restaurant service system (RSS).



Figure 2: Old chassis of the restaurant service system (RSS).

1.5 Applications

RSS is a novel idea that finds its applications in a wide array of fields. Though the system is now designed exclusively for restaurants, it can also find application in other applications mentioned below:

1. Hospitals to deliver medicine to the room.
2. Warehouses.
3. Offices.
4. Schools.

2. Literature Review

2.1 Project background

Nowadays, the robotic technology has been involved in a variety of places including the working environments. It has gradually replaced the manual work implemented by creatures. For example, in manual café systems, the customer can face the delay problems of ordering or

receiving the food, especially when the restaurant is crowded and the waiters are a few. The robot waiter is an innovation which can be implemented for restaurants autonomously to resolve such problem. It functions in various ways, such as using a pixy camera with line following robot where it defines its route accordingly. The project has two important parts namely the Menu Bar and the Robot itself. The robot waiter will work on the phenomenon of line following. Three sensors have been used. Two sensors were installed on the sides and are used for table counting, i.e. if the robot counts one, it means that it has stopped on the first table, and if the robot counts two, the robot has stopped on the second table for few seconds and so on. The third sensor is located in the front side of robot detecting the objects coming in front of robot while moving.

Robots are used to serve humanity. The branch of robotics that plays such a vital role is called “social robotics”. Social robots in today’s scenario are now communicating with human, interacting and relating to society in all aspect and are capable of understanding social terms. Due to the modernization in robotic technologies, many new designs and mechanisms are being implemented which are able to read human thoughts and understand actions. Such robots find vast applications in robotics e.g. to help out injured, sick, and elder people. Those robots are adaptive, i.e. they can be used in multi-mode as per scenario. So far, the robots are those who learn from us, but that time will not be so far when the teacher will then be a learner. There is an ever rising trend in using robots in restaurants for automation. These robots can welcome guests, take orders, and serve food to customers. Designing such robots can be effective to learn advance concepts in human-robot interaction, develop new models and protocols for communication as well as use new architectures for real time path planning, guidance, and control.

2.2 Previous Work

In this section, we will study similar projects that have been done in the field of restaurant

automation. This makes us understand better how the implementation can be done and what challenges we may face in the future.

a. Robot for Waiter Service, Department of Electrical and Electronic Engineering , BRAC

Figure 3 shows the implemented robot prototype for waiter service by the Department of Electrical and Electronic Engineering , BRAC.

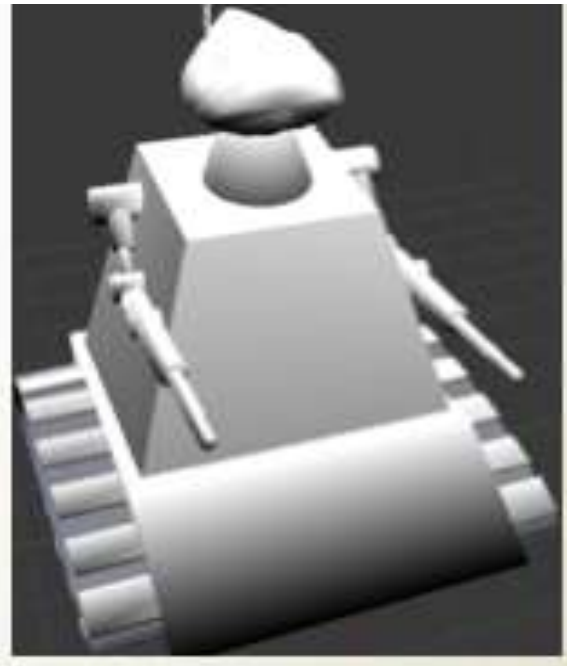


Figure 3: Robot prototype for waiter service by the Department of Electrical and Electronic Engineering , BRAC.

The next bullets summarize their implementation:

- The Waiter-Bot is an autonomous robot which has the ability to follow a designated path like a pathfinder with the help of IR sensor arrays and reach its intended destination.
- It is an Arduino based robotic design implemented to seek out and detect its required destination and perform its deliberate tasks with precision and accuracy.

- The Waiter-Bot consists of a simple mechanical design which has simple mechanism with which performing the necessary tasks becomes easier. Due to its design parameters, it also requires less power and draws less current which allows us to work with the robot safely.

b. Robotic Waiter for Restaurants, ECE 445 Design Document, TA: Xinrui Zhu

Figure 4 shows robotic waiter for restaurants implemented in Design Document course (ECE 445) in one of the Japanese universities.

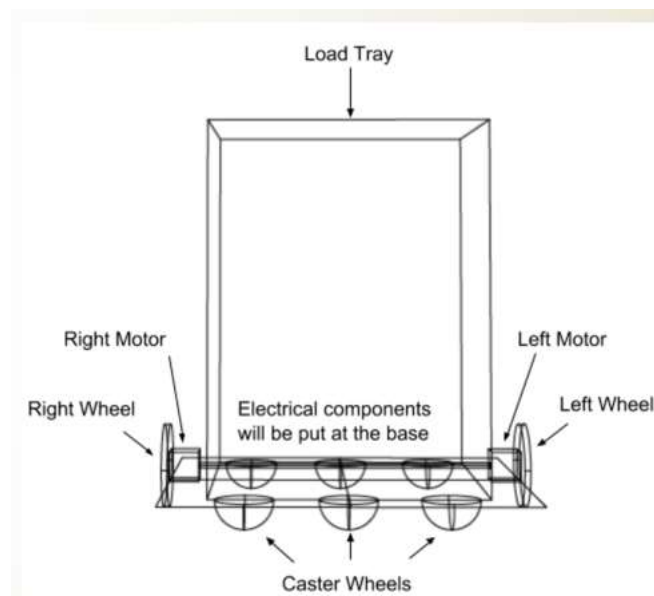


Figure 4: Robotic waiter for restaurants implemented in Design Document course (ECE 445).

The research to optimize restaurant operations has been extensive. Therefore, robots have been used and are currently used in the market to cut cost and increase efficiency. In the China's northeastern Zhejiang province, a restaurant has deployed robotic waiters. Each robot costs more than \$9,400. With the region's minimum monthly wage of \$300, it may require some time for the restaurant to come to a breakeven. However, in a long term, the restaurant may find itself saving money from the employment of less waiters.

In contrast, business owners in the USA may require less time than in China to come to a breakeven. This may be due to the much higher minimum wage. For example, with the D.C's minimum wage of \$11.50, this translates to approximately \$1,000 per month for every server a restaurant employs. Therefore, within a year, restaurant owners are guaranteed to save money with the assumption that no maintenance are required for the robot in that duration.

With the account of current robots costing at least \$9000, we propose to reduce the costs of robots. In real applications, since the cost of our robots are much lower, it will be relatively easy to increase the number of working robots per unit area.

2.3 Comparative Study

Table 1 states the comparison between the implemented robot the previous work discussed in Section 2.2

Table 1: Comparison between our robot and previous robot discussed in Section 2.2.

Projects	1	2	Our Project
Ordering system (Tablet)			✓
IR sensor (Table detection)	✓	✓	✓
Speaker			✓
Android Screen	✓	✓	✓
Obstacle avoidance	✓	✓	✓
Autonomous	✓	✓	✓
Bluetooth			✓
Greeting diners			✓

3. System Design

3.1 Design Constraints

WIFI standard:

IEEE 802.11 is part of the IEEE 802 set of LAN protocols, and specifies the set of media access control (MAC) and physical layer (PHY) protocols for implementing wireless local area network (WLAN) Wi-Fi computer communication in various frequencies, including but not limited to 2.4, 5, and 60 GHz frequency bands.

They are the world's most widely used wireless computer networking standards, used in most home and office networks to allow laptops, printers, and smartphones to talk to each other and access the Internet without connecting wires. They are created and maintained by the Institute of Electrical and Electronics Engineers (IEEE) LAN/MAN Standards Committee (IEEE 802). The base version of the standard was released in 1997, and has had subsequent amendments. The standard and amendments provide the basis for wireless network products using the Wi-Fi brand. While each amendment is officially revoked when it is incorporated in the latest version of the standard, the corporate world tends to market to the revisions because they concisely denote capabilities of their products. As a result, in the marketplace, each revision tends to become its own standard.

Outdoor protection:

IP 65 Protection is designed for the project. No ingress of dust; complete protection against contact (dust tight). A vacuum must be applied. Test duration of up to 8 hours based on air flow. Water projected by a nozzle (6.3 mm) against enclosure from any direction shall have no harmful effects

Microcontroller:

IEEE Standard 1855 is the first IEEE standard technology developed in the area of fuzzy logic. Its main characteristic is the interoperability, a design feature that enables system designers to develop fuzzy inference engines without taking into account the hardware/software constraints imposed by the specific architecture on which the system will be deployed. Thanks to this feature, a fuzzy system can be integrated into different types of architectures without any need to carry out porting strategies. This feature is particularly crucial in the area of embedded systems where, for each kind of device, a variety of applications, communication protocols, software libraries and programming tools, exists. In this context, Arduino TM technology represents one of the most popular architectures, thanks to its ease of development and prototyping. This paper shows how the native extendibility feature of IEEE Standard 1855 enables the design of a fuzzy rule-based systems in fully interoperable fashion on Arduino TM architectures and, as a consequence, allows designers to focus on fuzzy concepts, without any need to consider the hardware/software details related to the specific Arduino TM system.

IR Transceivers:

The NEC protocol uses pulse distance encoding of the bits. Each pulse is a $560\mu\text{s}$ long 38 kHz carrier burst ($26.3\mu\text{s}$). Logical bits are transmitted as follows:

Logical '0' – a $562.5\mu\text{s}$ pulse burst followed by a $562.5\mu\text{s}$ space, with a total transmit time of 1.125ms
Logical '1' – a $562.5\mu\text{s}$ pulse burst followed by a 1.6875ms space, with a total transmit time of 2.25ms . When a key is pressed on the remote controller, the message transmitted consists of the following, in order:

1. A 9ms leading pulse burst.
2. A 4.5ms space.
3. The 8-bit address for the receiving device.

4. The 8-bit logical inverse of the address.
5. The 8-bit command.
6. The 8-bit logical inverse of the command.
7. A final 562.5 μ s pulse burst to signify the end of message transmission.

3.2 Design Methodology

During the Design Methodology course, we breakdown the restaurant service system (RSS) into two categories as shown in Figure 5.



Figure 5: Categories of the restaurant service system (RSS).

3.3 Product Subsystems and Components

Figure 6 demonstrates the subsystems and components of the restaurant service system (RSS).

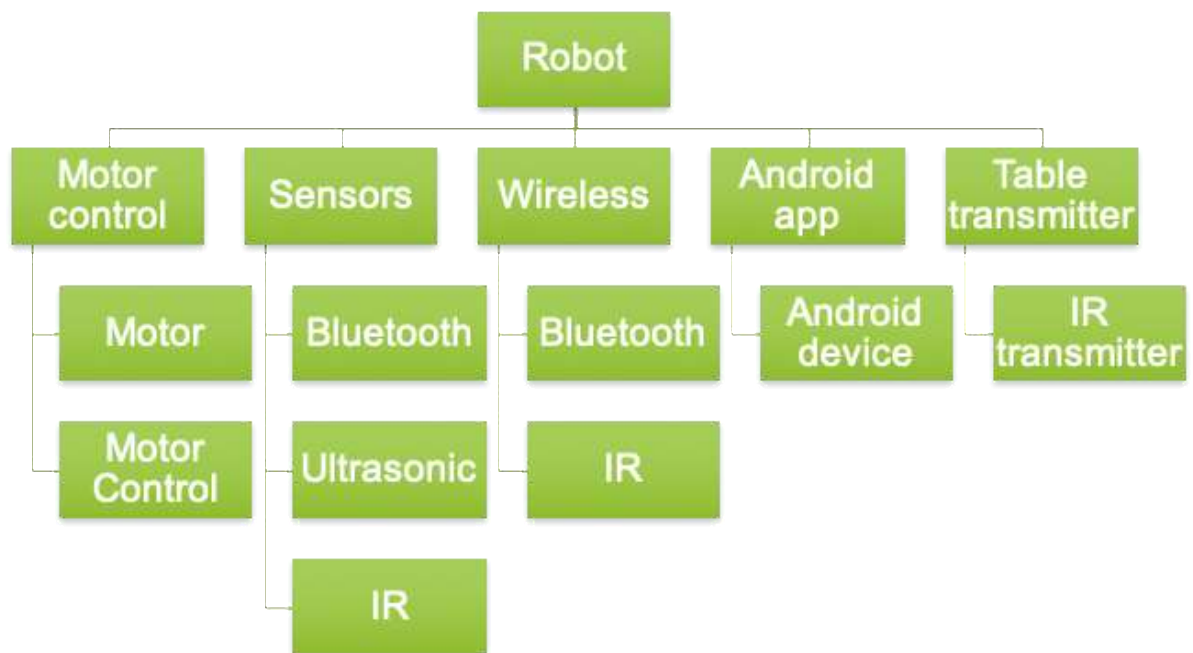


Figure 6: subsystems and components of the restaurant service system (RSS).

3.3.1 Product Subsystem 1: Motor control

This subsystem contains the motor and the motor controller. Figure 7 shows the system connections of subsystem 1.

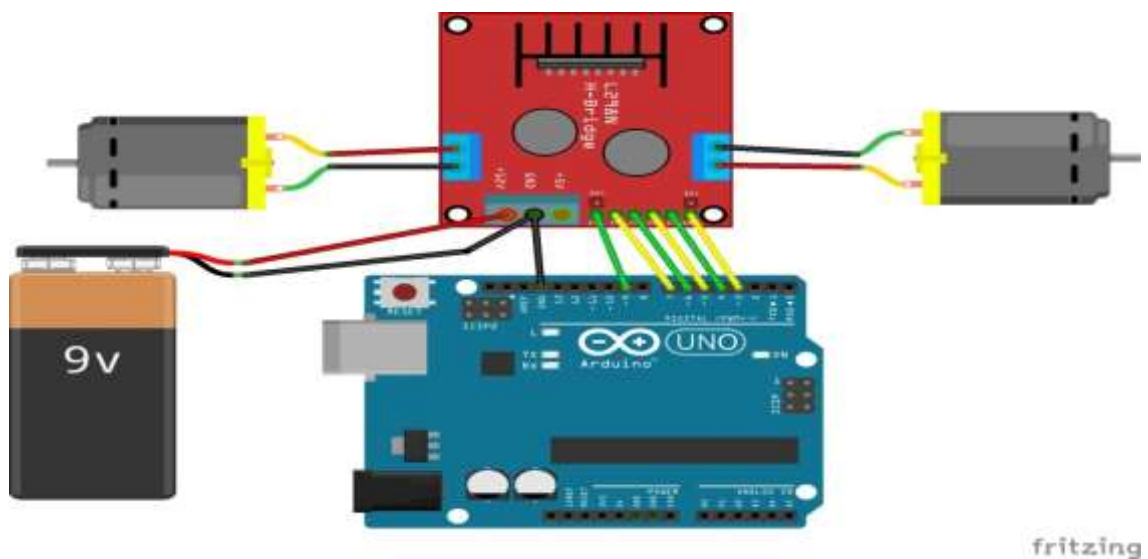


Figure 7: Subsystem 1 connections of the motor and motor control.

The system components are the moto and motor controller. Figure 8 shown the motor used in the restaurant service system (RSS) project.

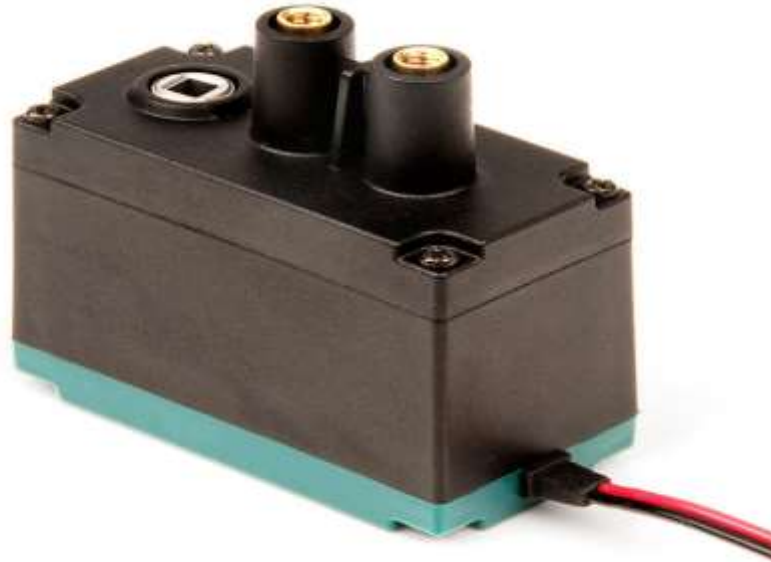


Figure 8: Motor used in the restaurant service system (RSS) project.

Table 2 states the specifications of the motor used in the restaurant service system (RSS) project.

Table 2: Motor specifications used in the restaurant service system (RSS) project.

Rotation	100 degrees
Stall Torque	6.5 in-lbs
PWM Pulse Width	1-2ms
Mix/Max Voltage	4.4 V – 15 V(Servo life will be reduced operating outside the VEX Controller range of 5.5 V - 9.0 V)
Weight	0.11 lbs.
Wiring	Black - ground; Orange - (+) power; White - PWM Control signal
Current Draw	20 mA to 1.5 A per servo
List of Contents	

1	VEX Servo Motor
1	VEX Servo Motor Gear Set (4 total gears, 1 replacement for each gear inside)
2	6-32 x 1/4in. Screws
2	6-32 x 1/2in. Screws
1	Clutch Post
1	Inventor's Guide Insert

Figure 9 shows the motor controller used in the restaurant service system (RSS) project.

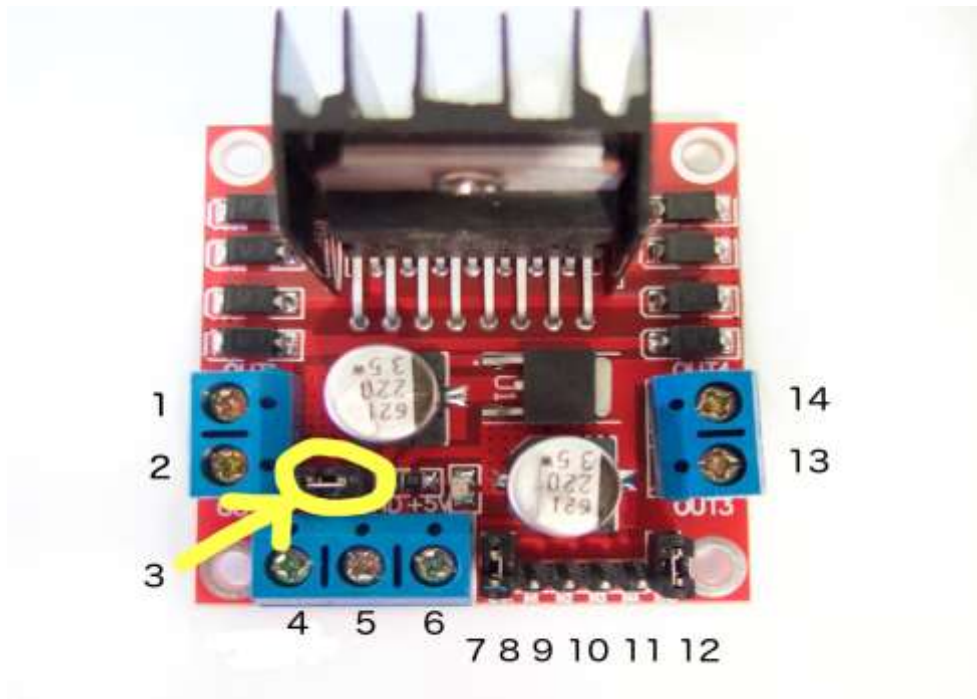


Figure 9: Motor controller used in the restaurant service system (RSS) project.

The configurations and connections of this motor controller are as follows:

Pin 1: DC motor 1 "+" or stepper motor A+.

Pin 2: DC motor 1 "-" or stepper motor A-.

Pin 3: 12 V jumper - remove this if using a supply voltage greater than 12 V DC. This enables power to the onboard 5 V regulator.

Pin 4: Connect your motor supply voltage here, maximum of 35V DC. Remove 12V jumper if >12 V DC.

Pin 5: GND.

Pin 6: 5 V output if 12 V jumper in place, ideal for powering your Arduino (etc).

Pin 7: DC motor 1 enable jumper. Leave this in place when using a stepper motor. Connect to PWM output for DC motor speed control.

Pin 8: IN1.

Pin 9: IN2.

Pin 10: IN3.

Pin 11: IN4.

Pin 12: DC motor 2 enable jumper. Leave this in place when using a stepper motor. Connect to PWM output for DC motor speed control.

Pin 13: DC motor 2 "+" or stepper motor B+.

Pin 14: DC motor 2 "-" or stepper motor B-.

3.3.2 Product Subsystem 2: Wireless

This subsystem contains the Bluetooth and IR Receiver connections. Figures 10 and 11 show the system connections of the Bluetooth and IR Receiver, respectively in subsystem 2.

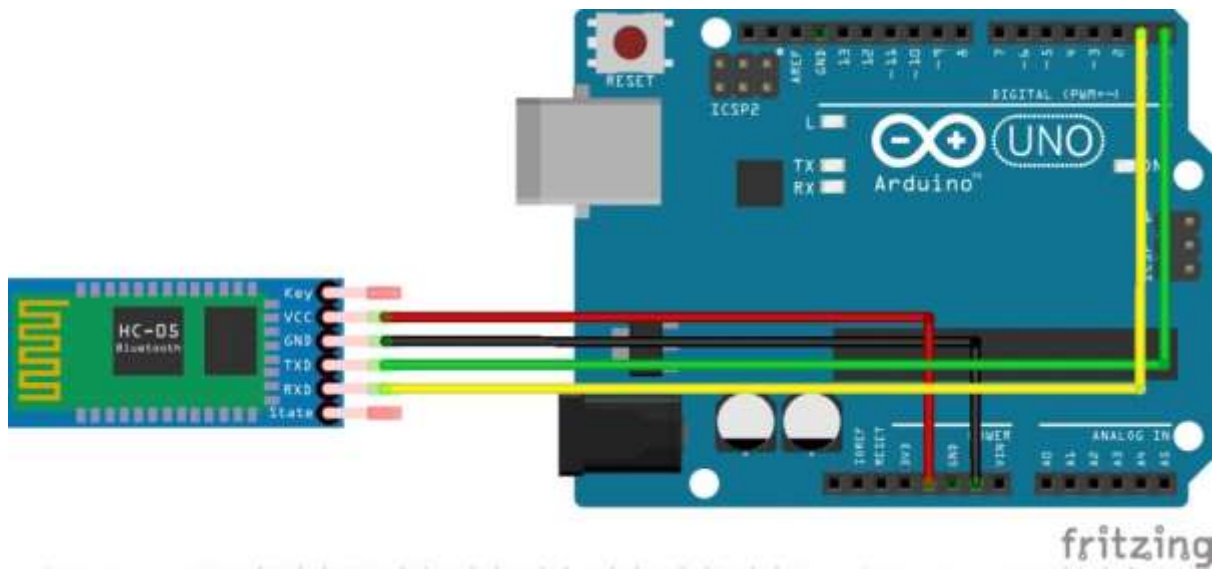


Figure 10: The Bluetooth connections in subsystem 2.

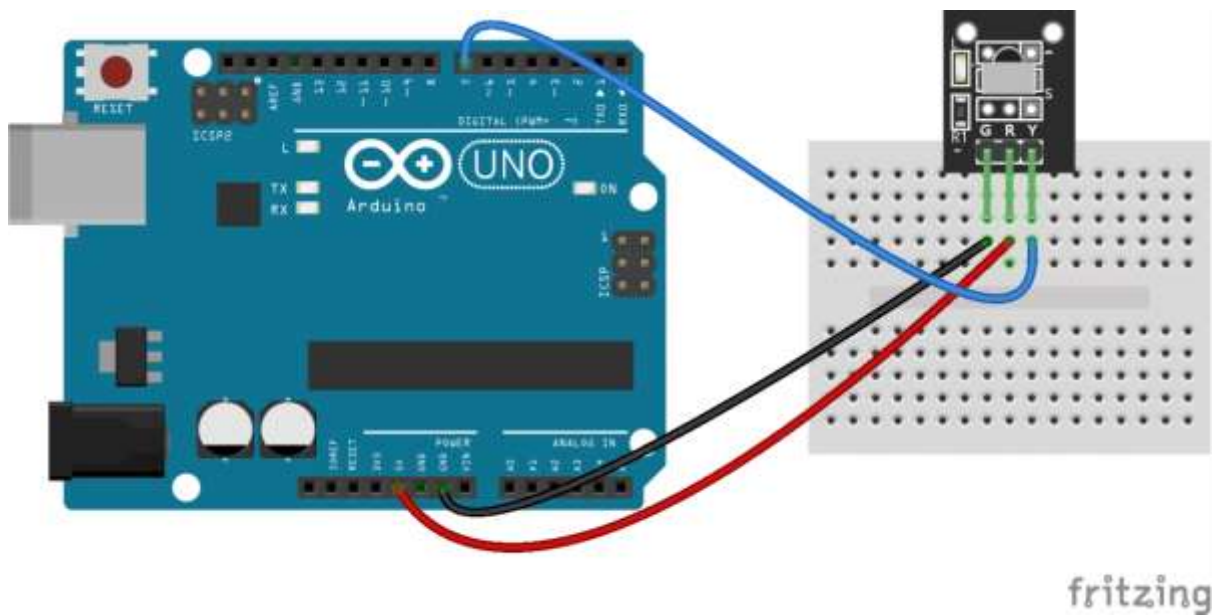


Figure 11: IR Receiver connections in subsystem 2.

Subsystem 2 consists of a HC 05, which is a Bluetooth module and IR receiver. Figure 12 shows the HC 05 Bluetooth module used in subsystem 2 of the restaurant service system (RSS) project.



Figure 12: HC 05 Bluetooth module.

The HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. The HC-05 Bluetooth module can be used in a Master or Slave configuration, making it a great solution for wireless communication. This serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4 GHz radio transceiver and base band. It uses CSR Blue core 04-External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature).

Figure 13 shows the IR receiver used in subsystem 2 of the restaurant service system (RSS) project.



Figure 13: IR receiver used in subsystem 2 of the restaurant service system (RSS) project.

This IR receiver is a lone infra-red sensor for use with an Arduino or Raspberry Pi. It can be used to decode infra-red signals from remote controls such as a TV remote or some RC vehicle remotes. It is especially designed to be reliable and easy to use with infra-red filters and a pre-amplifier IC built in. The infra-red sensor is just the sensor and you will need to connect it to the appropriate input and output pins on the Arduino or Raspberry Pi for it to function. This is simple as it only has 3 connections, power, ground, and signal out. This sensor is only a receiver and not a transmitter. Some features of the IR Receiver are:

- Photo detector and preamplifier in one package.
- Internal filter for PCM frequency.
- Inner shield, good anti-interference ability.
- High immunity against ambient light.
- Improved shielding against electric field disturbance.
- 3.0 V or 5.0 V supply voltage; low power consumption.
- TTL and CMOS compatibility.
- 8ms data pause time codes are acceptable.

Some specifications of the IR Receiver are:

- Typical -80 dBm sensitivity.

- Up to +4 dBm RF transmit power.
- 3.3 to 5 V I/O.
- PIO (Programmable Input/Output) control.
- UART interface with programmable baud rate.
- With integrated antenna.
- With edge connector.

Bluetooth connection

3.3.3 Product Subsystem 3: Sensors

This subsystem contains the Ultrasonic and IR line following sensors. Figures 14 and 15 show the system connections of the Ultrasonic and IR line following sensors, respectively used in subsystem 3 of the restaurant service system (RSS) project.

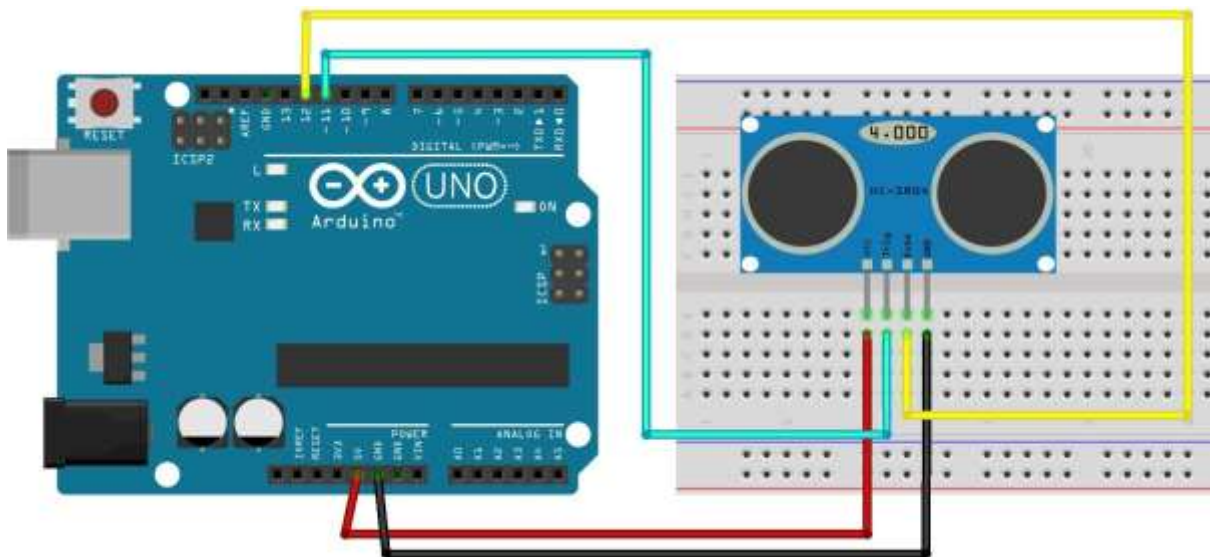


Figure 14: Ultrasonic connections in subsystem 3 of the restaurant service system (RSS) project.

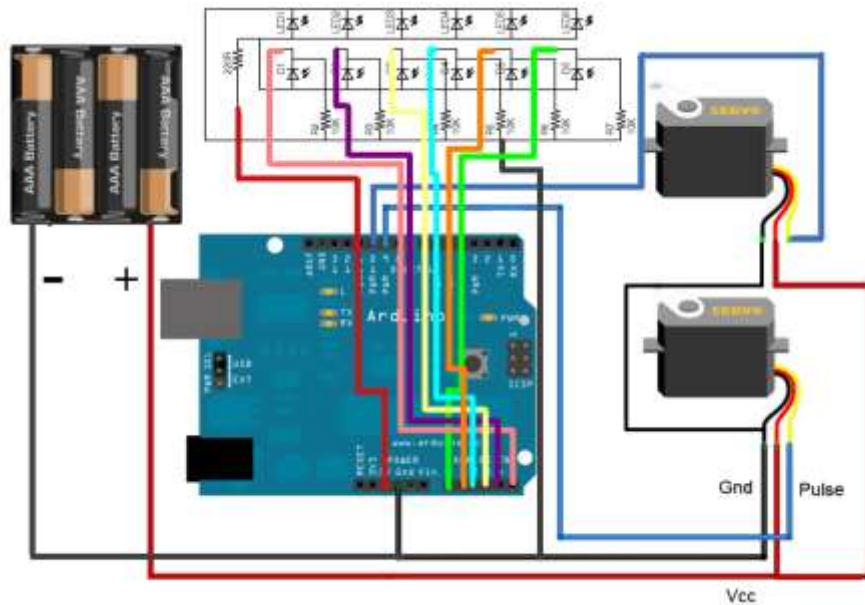


Figure 15: IR line following sensor connections in subsystem 3 of the restaurant service system (RSS) project.

Subsystem 3 consists of an Ultrasonic SRF 05 and IR line following sensor. Figure 16 shows the SRF 05 ultrasonic sensor used in subsystem 3 of the restaurant service system (RSS) project.



Figure 16: SRF 05 ultrasonic sensor used in subsystem 3 of the restaurant service system (RSS) project.

The ultrasonic sensor measures the distance of the nearest object, sending the result to the serial port. It can work from 2 cm to 3 m. It measures the time spent by the signal to reach the object and return to the sensor. The connections of the ultrasonic sensor are as follows:

- Vcc -> 5 V.
- Trig -> pin 13 (digital pin).
- Echo -> pin 12 (digital pin).
- Out ->.
- GND -> GND.

Figure 17 shows the IR line following sensor used in subsystem 3 of the restaurant service system (RSS) project.



Figure 17: IR line following sensor used in subsystem 3 of the restaurant service system (RSS) project.

The sensor works by detecting reflected light coming from its own infrared LED. By measuring the amount of reflected infrared light, it can detect transitions from light to dark (lines) or even objects directly in front of it. The mounting hole lets you easily connect one or more of these to the front or back of your robot chassis. Some features of the line following sensor are:

- Photosensitive diode.
- Mounting holes provided.
- Working power: 2.5~12 V.
- Working current: 18~20 mA (5 V)
- 3 Pin interface: VCC, OUT, GND.

3.3.4 Product Subsystem 1: Android app

The other category of the restaurant service system is the ordering system through an Android app. The system connection of this particular subsystem is shown in Figure 18.

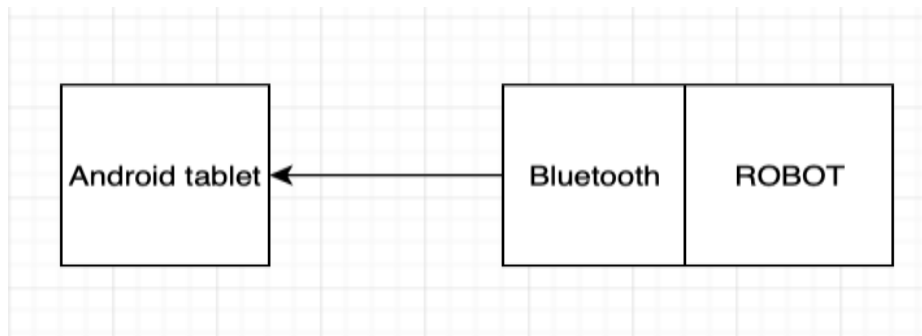


Figure 18: System connection of the ordering system used in the restaurant service system (RSS) project.

The components of this subsystem include the android tablet that will be placed on top of the robot and the Bluetooth chip that have been explained in the section 3.3.2.

3.4 Implementation

The implementations of the subsystems were performed during both Design Methodology and Assessment III courses. Figure 19 shows the flowchart of the subsystems implemented in Assessment III course.

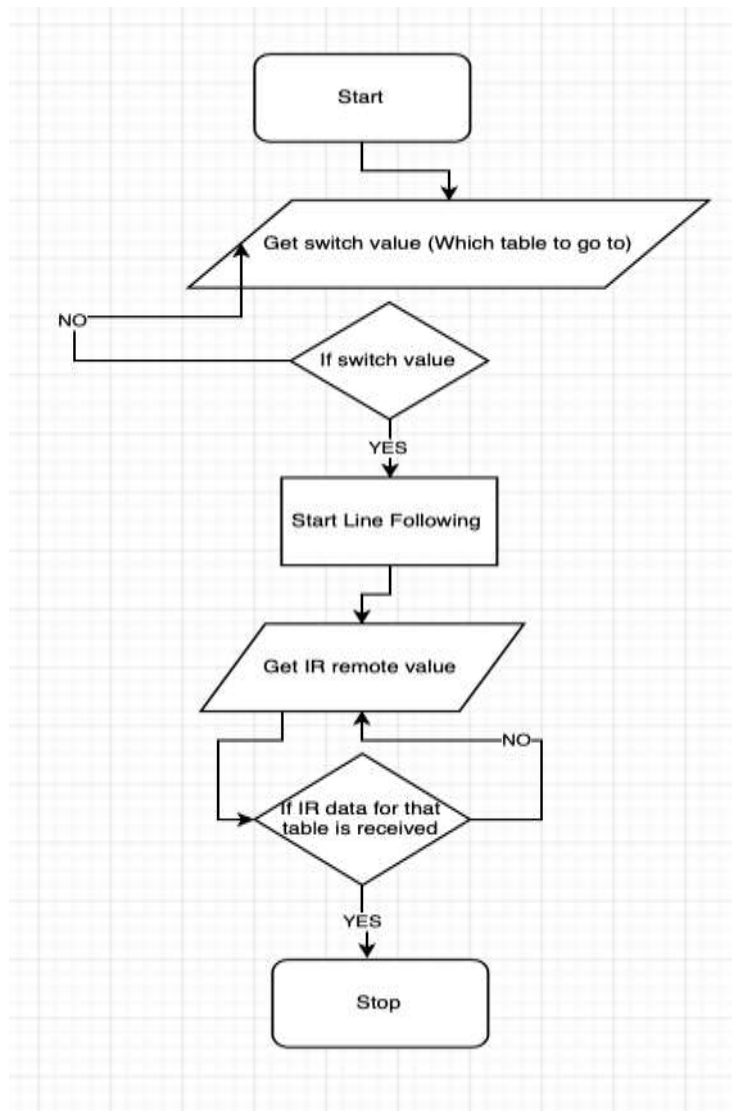


Figure 19: System implementations flowchart of the restaurant service system in Assessment III course.

Those systems are:

1. Line following
2. IR remote
3. Obstacle detection
4. Table buttons

4. System Testing and Analysis

4.1 Subsystem 1: Line Following & Motor Control

Objectives

1. Make sure there was sensory data to the controller and that all sensors were functional.
2. Make sure that the motor control connections were proper.

Setup

1. The line following sensors were connected to power and ground. The digital output from the sensors was fed to the Arduino digital pins.
2. The motor controller was connected to the motor and the control pins were connected to the Arduino.
3. The Arduino was coded to follow the black line.
4. The entire system was connected to power.

Results

1. The motor movement was observed and it was able to change the speed and direction based on the input from the sensors.
2. Motor control was fully functional and the motors were controlled for all directions.

4.2 Overall Results, Analysis, and Discussion

In the section above, the test routines of the systems were explained. In this section, the main findings are discussed.

1. There was a great friction in the wheels of the robot. Hence, it was not able to turn left or right.
2. When load was placed on the robot, it was not able to move properly. Hence, there was also deficiency in torque.

3. The wheels were not stable causing the height of the robot to change which deemed the line following method ineffective.

Figure 20 shows the performed test using the serial monitor of the Arduino of the restaurant service system (RSS) project.

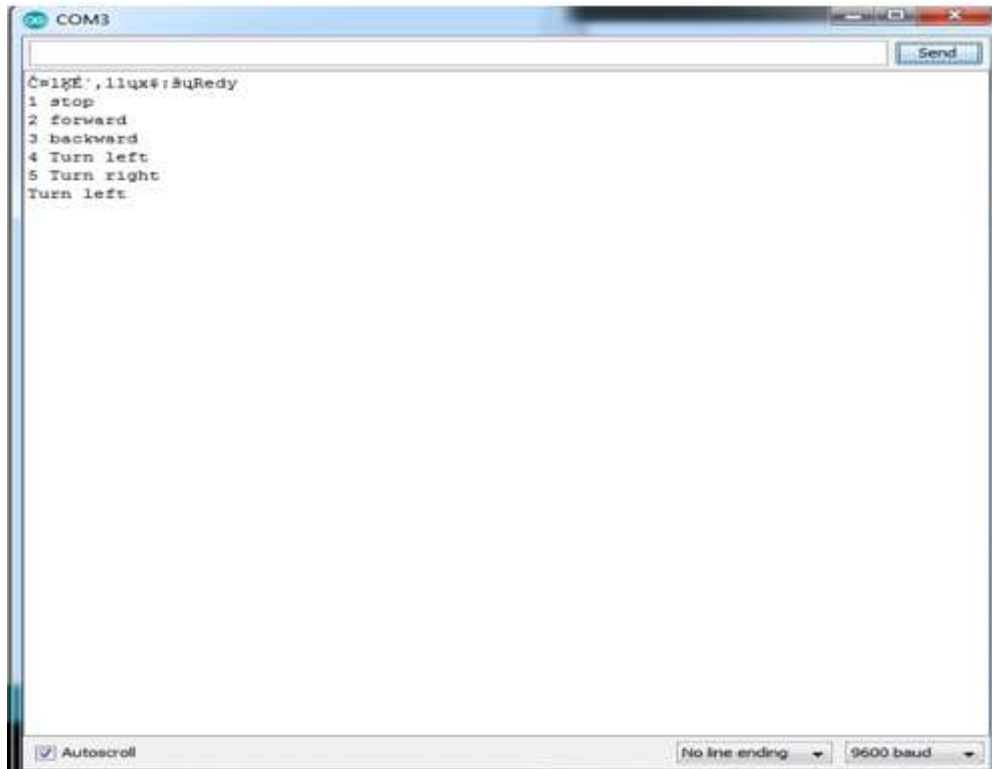


Figure 20: Serial monitor of the Arduino of the restaurant service system (RSS) project.

4.3 Assessment III

In this course, we completed our work in Design Methodology course. In the beginning of this course, we started working on finding solution for some challenges we faced in our project during the Design Methodology course. Our project specifications were implemented and satisfied within the subsystems. The subsystems were modified as shown in Figure 21.

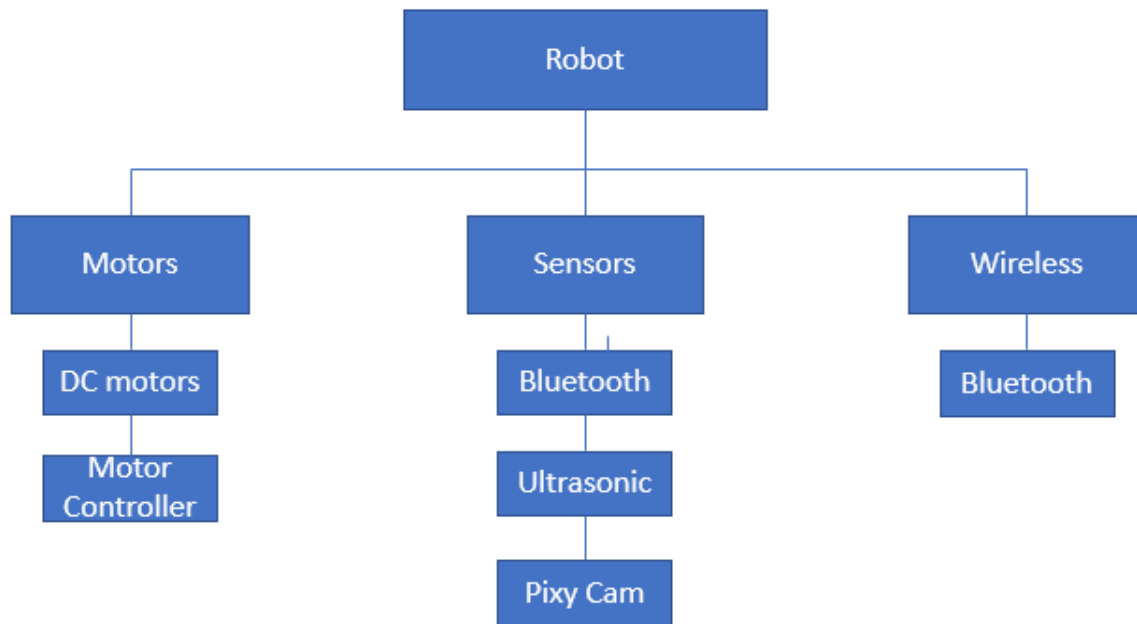


Figure 21: Modified subsystems of the restaurant service system (RSS) project.

In this section only the upgraded subsystems are mentioned next:

Major Upgrade: New Chassis (Figure 22)

Features:

- Good stability.
- Carrying more weight.
- Good movement with the wheels

Specifications:

- Size: 300 mm × 200 mm.
- Load carry more than 25 kg.
- Length: 87 mm.
- Diameter of axial: 6 mm.
- Axial length: 21 mm (axis course).



Figure 22: New chassis of the restaurant service system (RSS).

Product Subsystem 1: Motor Controller: RKI-1341 (Figure 23)

Features:

- Simple connectivity to IO pins of any MCU.
- Compatible with motors rated up to 24 V.
- Can easily deliver 20 A of current during normal operation
- Braking feature included without affecting the performance of an MCU

Specifications:

- Input voltage: 7 V minimum to 30 V maximum.
- Continuous current (< 1seconds) ~ 20 A.
- Absolute maximum peak current ~ 50 A.
- Continuous current (> 10seconds) ~ 5 A (without heat sink on MOSFETS).



Figure 23: Motor controller of the restaurant service system (RSS).

Product Subsystem 1: New Motor: Johnson Motor (Figure 24)

Features:

- Recommended to be used with DC motor driver 20 A. or dual DC motor driver 20 A.
- High torque.
- Proportional size.

Specifications:

- 60 RPM 12 V DC motors with metal gearbox and metal gears.
- 18000 RPM base motor.
- 6 mm diameter shaft with M3 thread hole.
- Gearbox diameter: 37 mm.



Figure 24: Johnson motor of the restaurant service system (RSS).

- Motor diameter: 28.5 mm.
- Length 63 mm without shaft.
- Shaft length 30 mm.
- 180 gm weight.
- 38 kg.cm holding torque.
- No-load current = 800 mA, Load current = up to 7.5 A (Max).

Product Subsystem 2: Sensors (Pixy Camera) (Figure 25)

Features:

- Very accurate.
- Cheap.
- Easy to connect.
- Proportional size.

Specifications:

- Processor: NXP LPC4330, 204 MHz, dual core
- Image sensor: Aptina MT9M114, 1296 x 976 resolution with integrated image flow processor.
- Field of view: 60 degrees horizontal, 40 degrees vertical.
- Dimensions: 1.5" x 1.65" x 0.6".
- Arduino cable, USB cable and mounting brackets included.

Product Subsystem 3: Table Detection (Figure 26)

Table Detection is performed by the IR sensor, whenever the photodiode transmitter signal is blocked from reflecting the signal, the receiver indicates that table is reached.



Figure 25: Pixy cam used in the restaurant service system (RSS) project.



Figure 26: Table detection sensor used in the restaurant service system (RSS).

Table 3 states the reason for selecting each component in the subsystems.

Table 3: Reason for selecting each component in the RSS robot.

Component	Reason	Alternative
Motors	High torque, geared	Stepper motor (less torque)
Pixy cam	More accurate and easy to use	IR sensor (less accurate)
Motor controller	High current capacity	L298 N (less current)
Bluetooth chip	Easy to use, cheap and locally available	Bluesmirf sparkfun chip
Battery 12 V	Cheap, easy to replace, reachable	Lithium ion
IR sensor	Giving signal 0 and 1, cheap , easy to use	No alternate chip

Product Subsystem 4: Ordering System (Figure 27)

Ordering systems is designed to maximize customers comfort. It is built in a way that allows customers to make their orders easily. It consists of an android application and a web.

Features:

- Accessible and easy to use.
- Has a lot of options to maximize customers' service.
- Contains services such as loyalty and offers.

Specifications:

- Android application.
- Web platform to view orders.
- Made orders can be printed automatically.
- Storage quantity can be automated (out of stock).

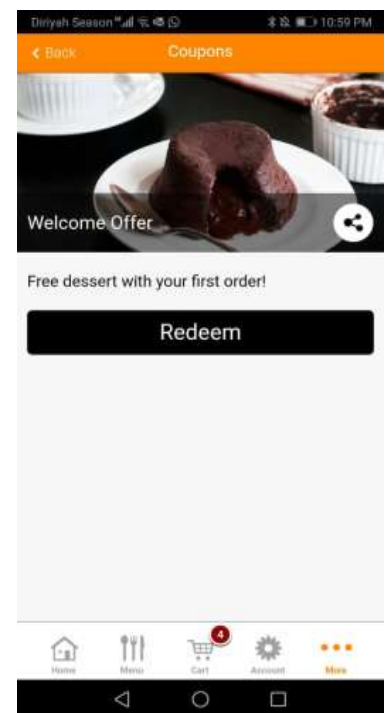


Figure 27: Ordering system used in the restaurant service system (RSS).

Figures 28-30 demonstrate some of the features that the ordering system has in the restaurant service system (RSS).

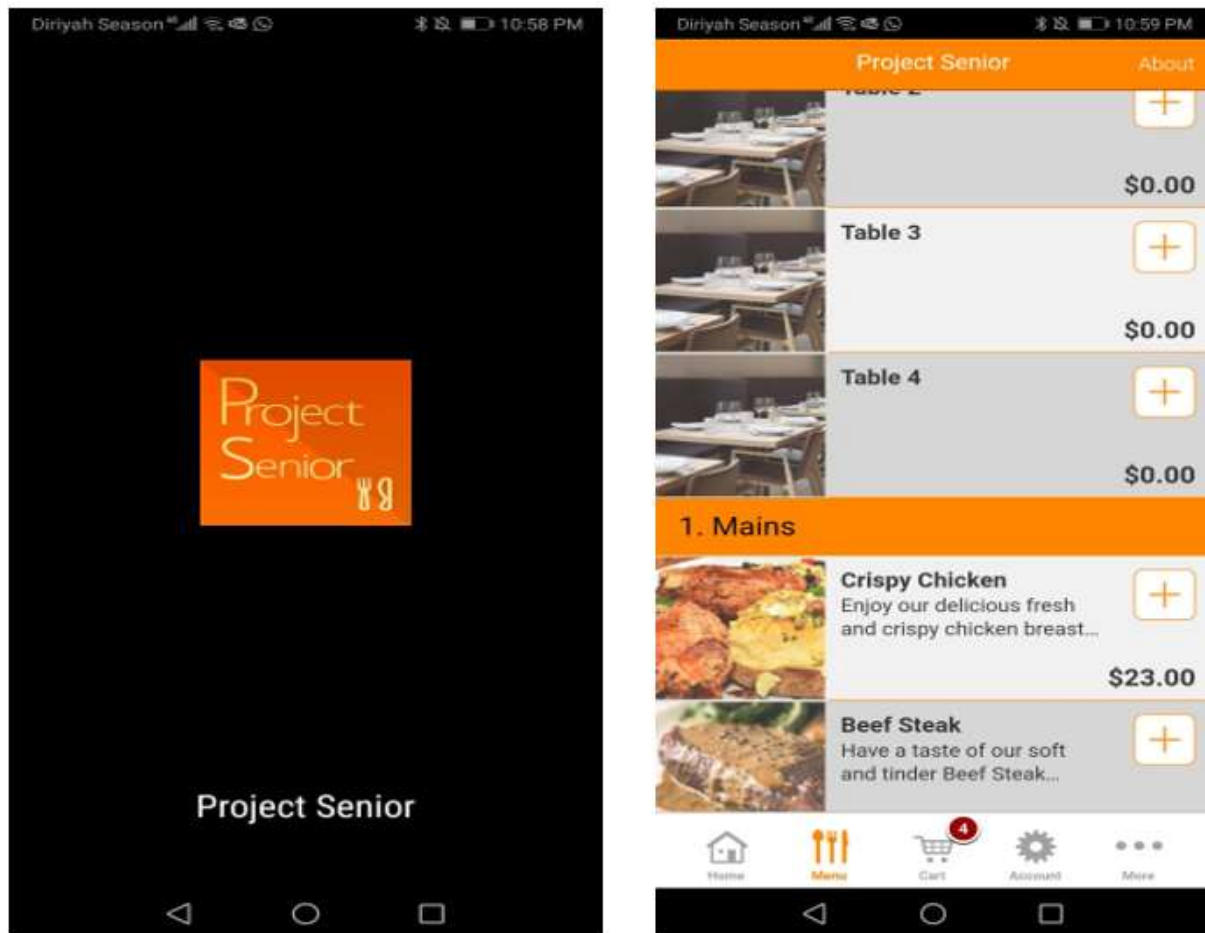


Figure 28: Welcoming and ordering page of the ordering system used in the restaurant service system (RSS).

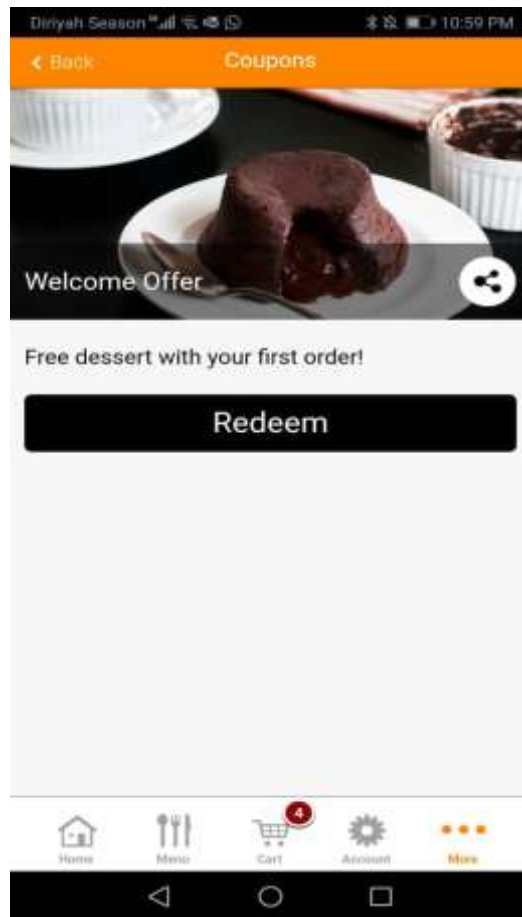


Figure 29: Coupons and other features of the ordering system in the restaurant service system (RSS).



Figure 30: Receiving order in the kitchen of the restaurant service system (RSS).

4.4 Final Prototype

Figure 31 shows the integration of the chassis with the subsystems of the restaurant service system (RSS).

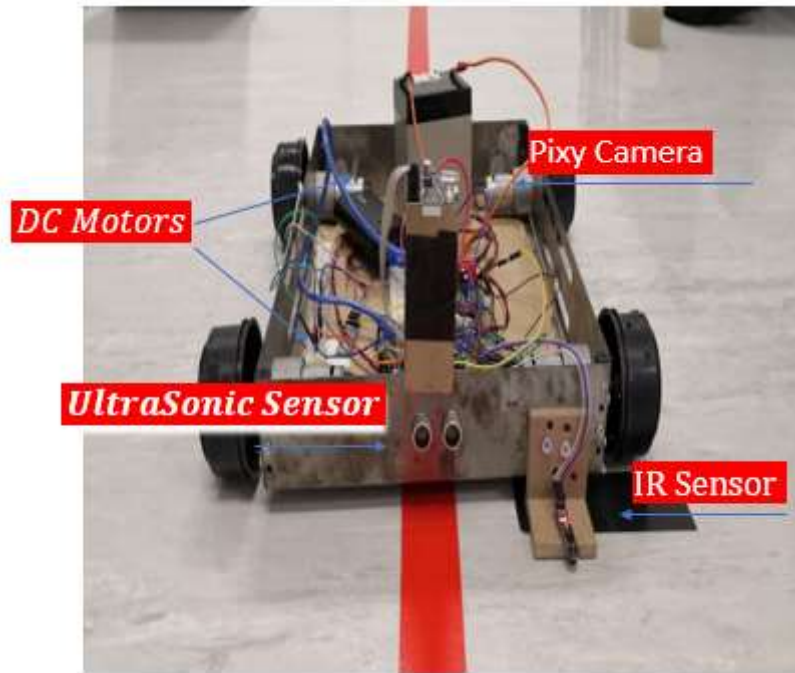


Figure 31: Integrated subsystem in the chassis of the restaurant service system (RSS).

Figure 32-35 shows the final prototype of the restaurant service system (RSS) robot.

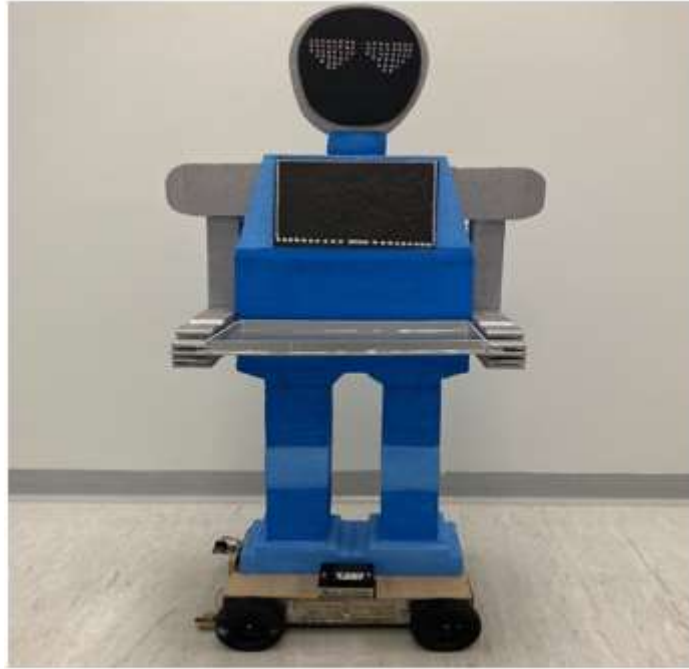


Figure 32: Final prototype of the restaurant service (RSS) robot.



Figure 33: Final prototype of the restaurant service system (RSS) with the team members.



Figure 34: Final prototype of the restaurant service system (RSS) while functioning to go to a designated table.



Figure 35: RSS robot serving some customers during the Electrical Engineering Demo day.

The route of the RSS robot can be chosen by the user. For example, Figure 36 shows a restaurant layout that the RSS robot can use as a suggested route.

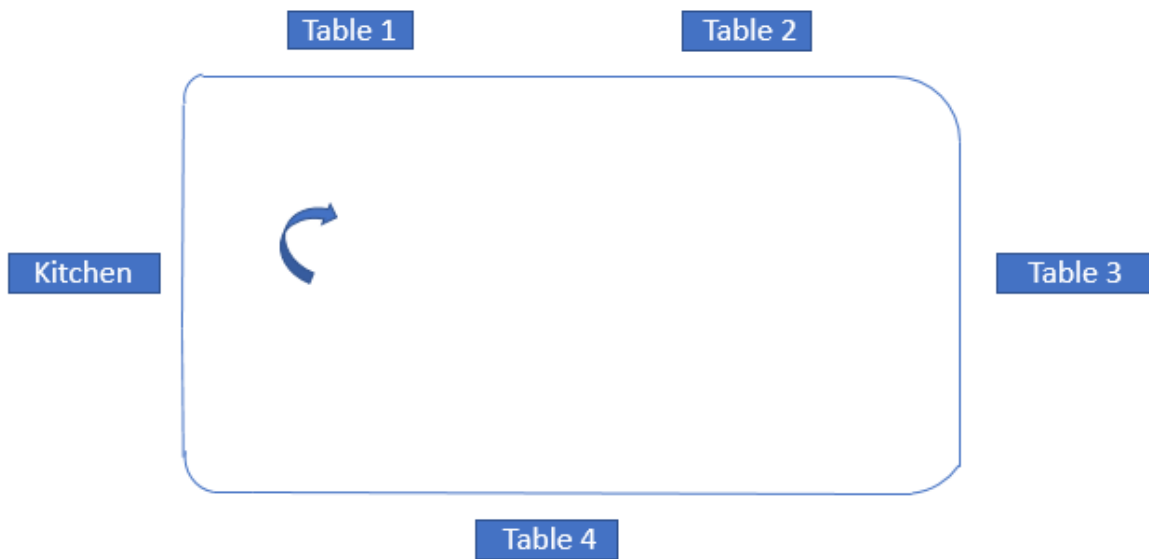


Figure 36: A suggested restaurant layout of the RSS robot.

5. Project Management

5.1 Project Plan

Figures 31 and 32 show the final progress reports during Design Methodology and Assessment III, respectively of the restaurant service system (RSS).

Restaurant Service System							Design Methodology & PM		Spring 2019					
a. Abdullah Almkhaitah 201400610							Project PLAN Instructor: Dr. Sadiq Al Hwaidi Period Highlight: 1 Actual (beyond plan) % Complete (beyond plan)							
b. Hamid Bin Hilal 201400673														
c. Ahmed Al Turki 201402385														
d. Ali Al Ali 201402853														
e. Abdulrahman Al Yousef 201500647														
ACTIVITY	PLAN START	PLAN URATIO	Assigned T _a	ACTUAL START	ACTUAL URATIO	PERCENT COMPLETE	Periods (Weeks 1-15)							
Form teams	1	1	ALL	1	1	100%	[Gantt chart bars]							
Select topic & advisor	2	1	ALL	2	2	100%	[Gantt chart bars]							
Write proposal and plan	2	1	ALL	2	2	100%	[Gantt chart bars]							
Background research & PE	3	2	ALL	3	2	100%	[Gantt chart bars]							
Search and acquire compo	3	3	a,b	5	3	100%	[Gantt chart bars]							
Design subsystem 1 (Robo	4	3	a,c,d	4	11	100%	[Gantt chart bars]							
Make the robot chasis	5	4	..	5	4	100%	[Gantt chart bars]							
Install motors	5	2		6	2	100%	[Gantt chart bars]							
Install motor controller, s	5	2		6	3	100%	[Gantt chart bars]							
Install battery, program m	6	5		7	2	100%	[Gantt chart bars]							
Activate basic line followi	6	1		7	4	80%	[Gantt chart bars]							
Prepare midterm Presenta	7	3	c,d	7	3	100%	[Gantt chart bars]							
Install IR sensor	8	2		8	2	100%	[Gantt chart bars]							
Install bluetooth	9	3		8	3	100%	[Gantt chart bars]							
Design subsystem 2 (port	9	4		8	5	100%	[Gantt chart bars]							
Prepare housing for the tr	10	2		10	2	100%	[Gantt chart bars]							
Make IR transmitter for t	10	1		11	2	100%	[Gantt chart bars]							
Make contract with restau	11	2		6	8	0%	[Gantt chart bars]							
Enable one way delivery	12	2		12	2	80%	[Gantt chart bars]							
Test all the systems	12	2		12	3	100%	[Gantt chart bars]							
Prepapre final report	12	3		10	5	60%	[Gantt chart bars]							
Prepapre final presentati	13	2		12	4	100%	[Gantt chart bars]							

Figure 37: Final progress report during Design Methodology course of the restaurant service system (RSS).

Title: Resturant Service System							Advisor: Dr. Sadiq Alhuwaidi		Design II (ASSE 3)		Fall 2019			
a. Abdullah Almkhaitah 201400610							Project PLAN & Progress ProgRpt No. 5 Plan updated (Date): Nov 21, 2019 Instructor: Dr. Sadiq Alhuwaidi Period Highlight: 6 Actual (beyond plan) % Complete (beyond plan)							
b. Hamid Bin Hilal 201400673														
c. Ahmed Al Turki 201402385														
d. Ali Al Ali 201402857														
e. Abdulrahman Yusuf 201500647														
ACTIVITY	PLAN START	PLAN URATIO	Assigned T _a	ACTUAL START	ACTUAL URATIO	PERCENT COMPLETE	Periods (Weeks 1-15)							
Prepare plaas	1	1	ALL	1	1	100%	[Gantt chart bars]							
Test all systems	2	1	ALL	2	2	100%	[Gantt chart bars]							
Test robot movement	3	1	ALL	3	3	100%	[Gantt chart bars]							
Integrate all systems	4	2	ALL	4	7	100%	[Gantt chart bars]							
Testing integrated system	5	3	b,d,e	5	18	100%	[Gantt chart bars]							
Design robot chassis	6	1	ALL	6	10	100%	[Gantt chart bars]							
Test robot chassis	7	1	ALL	8	11	100%	[Gantt chart bars]							
Initiate Andriod engagment	6	1	d,e	6	4	100%	[Gantt chart bars]							
Prepare midterm presentation	7	1	ALL	7	2	100%	[Gantt chart bars]							
Design ordering system	9	2	d,e	8	10	100%	[Gantt chart bars]							
Test ordering system	9	2	d,e	10	12	100%	[Gantt chart bars]							
Test basic functionality for route	8	1	d,e	12	13	100%	[Gantt chart bars]							
Test the entire RSS system	11	1		11	14	100%	[Gantt chart bars]							
Prepapre final report	12	2	a,e	13	14	100%	[Gantt chart bars]							
Prepapre final presentation	12	1	d,e	12	13	100%	[Gantt chart bars]							
Prepare project demo	12	2	ALL	12	15	100%	[Gantt chart bars]							
Submit Rpt/PPT/Brochure ...	14	1	b,e	12	15	100%	[Gantt chart bars]							
Progress Details:							Issues (delay ...):							

Figure 38: Final progress report during Assessment III course of the restaurant service system (RSS).

5.2 Contribution of Team Members

Table 4 states the contribution of the team members during both courses in the restaurant service system (RSS) project.

Table 4: Contribution of team members to implement the RSS robot.

Task	Ali	Ahmed	Hamid	Abdilrahman	Abdullah
Search and acquire components	30%	10%	20%	20%	20%
Design and Implementation	15%	35%	20%	20%	10%
Research and Problem Solving	25%	25%	10%	20%	20%
Physical Construction	20%	30%	10%	20%	20%
Sensors	10%	20%	30%	10%	30%
Coding and Testing	15%	15%	15%	40%	15%
Reports and Presentations	20%	20%	20%	20%	20%

5.3 Project Execution Monitoring

- Meetings with Advisors:
 - Weekly team meeting.
 - Bi weekly meetings with advisor.
- Team meetings:
 - Team meetings would be communicated through the WhatsApp group for joint decision making.
- Other activities:
 - During our meetings, one session would be theory and the other session would be implementation and testing.

5.4 Challenges and Decision Making

This project was planned after the idea was finalized. To plan it better, a literature search should have been done to study similar projects. Hence, before beginning any work on a project, the design and concept should have been finalized. The team members have tried their best to follow the plan that was made on the onset. However, during execution and implementation, there were many challenges that we have faced some of which are listed below:

1. Line following could not be achieved as the robot was not able to turn left or right.
2. The selected motor lacked to enough torque.
3. The wheels attached to the motor were not stable which causes the IR line following sensors to miss the line.
4. We have decided to change the motors of the project.
5. We are considering to remove the IR remote receiver and replace it with horizontal lines for stopping in the project.
6. We have decided to scrap the IR line following sensors and replace it with an image recognition camera for better detection of the line.
7. The customer needs to wave at the robot to have the robot leave back to the kitchen. As a result, a message has to be delivered to the customer to know this information and the procedure to have the customer notify the robot.
8. For the suggested restaurant layout in Figure 36, the RSS robot can be enhanced to have it go to table 4, for example, in anticlockwise direction instead of crossing all tables to save the time.

5.5 Project Bill of Materials and Budget

Table 5 stated the total cost of the restaurant service system (RSS) robot.

Table 5: Total cost of the RSS robot.

Item	Cost (SR)
Ultrasonic Sensors	200
Arduino	150
IR sensor	100
Robot Chassis	600
Motor Controller	250
Motors	400
Ordering System App	300
Total	2000

6. Project Analysis

6.1 Life-long Learning

A final engineering project is a mammoth task. There is a lot of challenges that we face and overcome. In this project, we have learned the following technical and non-technical skills:

1. Leadership.
2. Team work.
3. Communication and presentation.
4. Planning and coordination.
5. Motor control.
6. Arduino hardware.
7. Arduino IDE software.
8. IR remote receiver.
9. IR line following sensors.
10. Pixy Camera.

6.2 Impact of Engineering Solutions

From the very onset, our goal was to create a project that creates an impact on the society. With the Saudi vision 2030, there is a lot of emphasis on the use of technology to make ease in the market. The main impact of the project has been summarized in the following point:

- Handicapped people can be hired by the restaurant to remotely control the robot and manage the services.

6.3 Contemporary Issues Addressed

Our project has impacted the following temporary issues:

1. The cost of hiring is very high in Saudi Arabia. By the use of the RSS robot, restaurants can save on labor costs.
2. The Saudi vision 2030 pushes for the use of robotics. The RSS robot falls in the same line.

7. Conclusions and Future Recommendations.

7.1 Conclusions

This course is a partial completion of the final design project as a fulfilment of the Engineering program at Prince Mohammad bin Fahd University. The conceptual design of this project was set at the beginning of the course. The team has followed the project plan and accomplished all of the tasks including major challenges faced. These include the friction of the wheels that has limited the completion of the line following mode. There was also realization of the project implementation which has lead us to cancel the use of IR line following sensors and instead use an image recognition camera. At the end of this course, we are confident and excited to complete the second part of the project.

7.2 Future Recommendations

Since the new technologies and robotic life are ongoing research areas, the completion of this particular system cab go beyond expectations. Some recommendations include:

1. Change the line following sensor and replace it with the camera.
2. Remove the IR remote receiver as it is redundant.
3. Work on mapping the restaurant with the triangular approach.
4. Let the RSS robot go in both clockwise and anticlockwise directions.

5. Have a screen on the robot to for ordering purposes and entertaining the customer.
6. Save some voice command where the robot can use to interact with the customer in a smart way.

8. References

[1] http://nesl.ee.ucla.edu/courses/ee202a/shared/samples/projets/2008f/Enhao_Yap.pdf.

[2]

http://www.nexrobotics.com/index.php?page=shop.product_details&flypage=flypage.tpl&category_id=32&product_id=185&option=com_virtuemart&Itemid=45

[3]

https://github.com/eyantra/Autonomous_Waiter_Robot_using_Firebird_ATmega2560/blob/master/Documentation/Autonomous_Waiter_Robot.pdf

[4]

<https://github.com/akshar100/eyantrafirebirdresources/tree/master/Fire%20Bird%20V%20ATMEGA2560%20Robots%202010-12> <https://decibel.ni.com/content/docs/DOC-30926>

[5] <http://www.iosrjournals.org/iosr-jeee/Papers/Vol6- issue5/L0658084.pdf>

[6] <http://www.ifr.org/service-robots/statistics>

[7] http://www.cmosexod.com/micro_uart.htm

[8] <http://www.best-microcontroller-projects.com/hardware-interrupt.html>

[9] [http://www.avrtutor.com/tutorial/interrupt/interrupts .php](http://www.avrtutor.com/tutorial/interrupt/interrupts.php)

[10] Bekey GA. On Autonomous Robots. The Knowledge Engineering Review. 1998; 13(2): 143- 146.

[11] Bajestani SEM & Vosoughinia A. Technical Report of Building a Line Follower Robot. 2010 International Conference on Electronics and Information Engineering (ICEIE 2010); 2010. p. V1-1 – V1-5.

[12] Masuda T, Misaki D. Development of Japanese Green Tea Serving Robot “T-Bartender”. Proceeding of the IEEE International Conference on Mechatronics & Automation; July 2005; Niagara Falls, Canada. Fukuroi-shi, Toyosawa, Japan: Department of Mechanical Engineering, Shizuoka Institute of Science and Technology; 2005. p. 1069- 1074.

[13] Limbu DK, Yeow KT, Lawrence TC Por. FusionBot: A Barista Robot. 2012. Singapore: Institute for Info comm Research. IEEE. p. 341.

[14] Takahashi Y, Hosokawa M, Mochizuki T. Tea Serving Robot. SICE; 29-31 July 1997. Tokushima, Japan. p. 1111-1114. [14] Takahashi Y, Nakamura

[15] <https://courses.engr.illinois.edu/ece445/getfile.asp?id=12502>

Appendix A: Progress Reports

Restaurant Service System		Assessment III		FALL 19/20														
a. Abdullah Almkhaitah 201400610		Project PLAN Instructor: Dr. Sadiq Al Huwaidi Period Highlight: 1 (Actual) / 1 (Plan) / 1 (Actual) Actual (beyond plan) / Complete (beyond plan)																
b. Hamid Bin Hilal 201400673																		
c. Ahmed Al Turki 201402385																		
d. Ali Al Ali 201402853																		
e. Abdulrahman Yusuf 201500647																		
ACTIVITY	PLAN START	PLAN URATIO	Assigned To	ACTUAL START	ACTUAL URATIO	PERCENT COMPLETE	Periods (Weeks 1-15)											
Prepare plans	1	1	ALL	1	1	100%	Actual											
Test all systems	2	1	ALL	2	2	100%	Actual											
Test robot movement	3	1	ALL	3	3	100%	Actual											
Integrate all systems	4	2	ALL	4	5	80%	Actual											
Testing integrated system	5	3	b,d,e	5		35%	Actual											
Design robot chasis	6	1				50%	Actual											
Test robot chasis	7	1				0%	Actual											
Initiate Andriod engagmen	6	1				0%	Actual											
Desinge ordering system	9	2				0%	Actual											
Test ordering system	9	2				0%	Actual											
Test robot route	8	1				0%	Actual											
Test basic functionality fo	8	1				0%	Actual											
Test all robot systems usin	9	1				0%	Actual											
Test the entire RSS system	11	1				0%	Actual											
Prepare final report	9	3				0%	Actual											
Prepare final presentation	10	1				0%	Actual											
Progress Details:							Issues											
all components used for the robot sysem are availabe and functioning properly however, there might be upgrades for the robot chasis to provide higher stability							Lots of troubleshooting when integrated system is tested which may cause set backs and delays											

Figure 39: In between progress report 1.

Title: Resturant Service System		Advisor: Dr. Sadiq Alhuwaidi		Design II (ASSE 3)		Fall 2019												
a. Abdullah Almkhaitah 201400610		Project PLAN & Progress ProgRpt No. 2 Plan updated (Date): Oct 10, 2019 Instructor: Dr. Sadiq Alhuwaidi Period Highlight: 6 (Actual) / 6 (Plan) / 6 (Actual) Actual (beyond plan) / Complete (beyond plan)																
b. Hamid Bin Hilal 201400673																		
c. Ahmed Al Turki 201402385																		
d. Ali Al Ali 201402857																		
e. Abdulrahman Yusuf 201500647																		
ACTIVITY	PLAN START	PLAN URATIO	Assigned To	ACTUAL START	ACTUAL URATIO	PERCENT COMPLETE	Periods (Weeks 1-15)											
Prepare plans	1	1	ALL	1	1	100%	Actual											
Test all systems	2	1	ALL	2	2	100%	Actual											
Test robot movement	3	1	ALL	3	3	100%	Actual											
Integrate all systems	4	2	ALL	4	5	80%	Actual											
Testing integrated system	5	3	b,d,e	5	6	50%	Actual											
Design robot chassis	6	1	ALL	6		50%	Actual											
Test robot chassis	7	1				0%	Actual											
Initiate Andriod engagment	6	1	e,a,c	6		0%	Actual											
Prepare midterm presentation	7	1	ALL			50%	Actual											
Design ordering system	9	2				0%	Actual											
Test ordering system	9	2				0%	Actual											
Test basic functionality for route	8	1				0%	Actual											
Test the entire RSS system	11	1				0%	Actual											
Prepapre final report	12	2				0%	Actual											
Prepapre final presentation	12	1				0%	Actual											
Prepare project demo	12	2				0%	Actual											
Submit Rpt/PPT/Brochure ...	14	1				0%	Actual											
Progress Details:							Issues (delay ...):											
Chassis upgrade plans are finished. Integratd system test progression reached 50% .							Integrated system not completed, piiri cam is not installed yet because the upgraded chassis will take some time to arrive											

Figure 40: In between progress report 2.

2	Title: Resturant Service System		Advisor: Dr. Sadiq Alhuwaidi				Design II (ASSE 3)		Fall 2019															
3	a. Abdullah Almakhaitah 201400610						Project PLAN & Progress																	
4	b. Hamid Bin Hilal 201400673						ProgRpt No. 3																	
5	c. Ahmed Al Turki 201402385						Plan updated (Date): Oct 24, 2019																	
6	d. Ali Al Ali 201402857						Instructor: Dr. Sadiq Alhuwaidi																	
7	e. Abdulrahman Yusuf 201500647						Period Highlight: 6																	
8	ACTIVITY		PLAN	PLAN	Assigned	ACTUAL	ACTUAL	PERCENT	Periods (Weeks 1-15)															
9		TAR	URATIO	To	START	URATIO	COMPLETE		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
11	11	Prepare plans	1	1	ALL	1	1	100%																
12	12	Test all systems	2	1	ALL	2	2	100%																
13	13	Test robot movement	3	1	ALL	3	3	100%																
14	14	Integrate all systems	4	2	ALL	4		65%																
15	15	Testing integrated system	5	3	b,d,e	5		50%																
16	16	Design robot chassis	6	1	ALL	6		50%																
17	17	Test robot chassis	7	1	a,e	8		25%																
18	18	Initiate Andriod engagment	6	1	e,a,c	6		0%																
19	19	Prepare midterm presentation	7	1	ALL	7	2	100%																
20	20	Design ordering system	9	2	e,c	8		15%																
21	21	Test ordering system	9	2				0%																
22	22	Test basic functionality for route	8	1				0%																
23	23	Test the entire RSS system	11	1				0%																
24	24	Prepapre final report	12	2				0%																
25	25	Prepapre final presentation	12	1				0%																
26	26	Prepare project demo	12	2				0%																
27	27	Submit Rpt/PPT/Brochure ...	14	1				0%																
29	Progress Details:												Issues (delay -):											
30	is has been upgraded and tested, however, assembling and installing integrated system and further tes		required to ensure the desired fuctionality of the robot.										Assembling and installing all sensors along with tests will require some time which may cause delays.											
31																								
32																								
33																								
34																								
35																								

Figure 41: In between progress report 3.

Appendix B: Datasheets

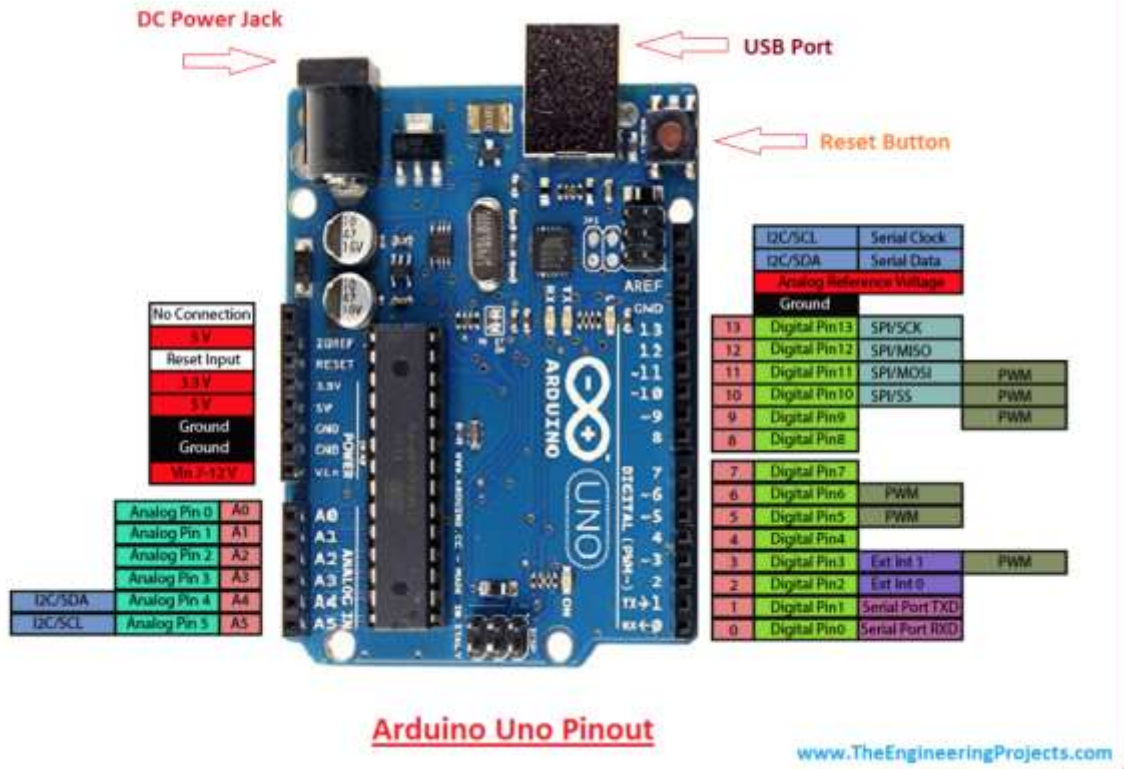


Figure 42: Arduino pinout.



www.vishay.com

TSOP322.., TSOP324.., TSOP344.., TSOP348..

Vishay Semiconductors

IR Receiver Modules for Remote Control Systems



DESIGN SUPPORT TOOLS

[click logo to get started](#)



MECHANICAL DATA

Pinning for TSOP348.., TSOP344..:

1 = OUT, 2 = GND, 3 = V_S

Pinning for TSOP322.., TSOP324..:

1 = OUT, 2 = V_S , 3 = GND

FEATURES

- Very low supply current
- Photo detector and preamplifier in one package
- Internal filter for PCM frequency
- Supply voltage: 2.5 V to 5.5 V
- Improved immunity against ambient light
- Insensitive to supply voltage ripple and noise
- Material categorization: for definitions of compliance please see www.vishay.com/doc799912



DESCRIPTION

The TSOP322.., TSOP348.., TSOP324.., and TSOP344.. series are miniaturized IR receiver modules for infrared remote control systems. A PIN diode and a preamplifier are assembled on lead frame, the epoxy package contains an IR filter.

The demodulated output signal can be directly connected to a microprocessor for decoding.

The TSOP324.., TSOP344.. series devices are optimized to suppress almost all spurious pulses from energy saving lamps like CFLs. The AGC4 used in the TSOP324.., TSOP344.. may suppress some data signals. The TSOP322.., TSOP348.. series are provided primarily for compatibility with old AGC2 designs. New designs should prefer the TSOP324.., TSOP344.. series containing the newer AGC4.

These components have not been qualified according to automotive specifications.

PARTS TABLE					
AGC	LEGACY, FOR LONG BURST REMOTE CONTROLS (AGC2)			RECOMMENDED FOR LONG BURST CODES (AGC4)	
Carrier frequency	30 kHz	TSOP34830	TSOP32230	TSOP34430	TSOP32430
	33 kHz	TSOP34833	TSOP32233	TSOP34433	TSOP32433
	36 kHz	TSOP34836	TSOP32236	TSOP34436 ⁽¹⁾⁽²⁾⁽³⁾	TSOP32436 ⁽¹⁾⁽²⁾⁽³⁾
	38 kHz	TSOP34838	TSOP32238	TSOP34438 ⁽⁴⁾⁽⁵⁾	TSOP32438 ⁽⁴⁾⁽⁵⁾
	40 kHz	TSOP34840	TSOP32240	TSOP34440	TSOP32440
	56 kHz	TSOP34856	TSOP32256	TSOP34456 ⁽⁶⁾⁽⁷⁾	TSOP32456 ⁽⁶⁾⁽⁷⁾
Package	Mold				
Pinning	1 = OUT, 2 = GND, 3 = V_S	1 = OUT, 2 = V_S , 3 = GND	1 = OUT, 2 = GND, 3 = V_S	1 = OUT, 2 = V_S , 3 = GND	
Dimensions (mm)	6.0 W x 6.95 H x 5.6 D				
Mounting	Leaded				
Application	Remote control				
Best choice for	⁽¹⁾ RC-5 ⁽²⁾ RC-6 ⁽³⁾ Panasonic ⁽⁴⁾ NEC ⁽⁵⁾ Sharp ⁽⁶⁾ r-step ⁽⁷⁾ Thomson RCA				

Figure 43: IR sensor datasheet.

Appendix C: Operation Manual

In the beginning, turn on the battery via the switch and the microcontroller will turn on.

1. The kitchen will receive the order from the customer, who makes the order via the designed ordering system.
2. The chief will start cooking the order. Once the food is ready, the chief will place the food in the plate fixed on the robot.
3. The robot will take the order from restaurant staff through the Bluetooth by opening the Arduino Voice Control Application (Available on IOS and Android) by restaurant staff.
4. The staff would ask the robot to go to designated table by the voice command (e.g. Table 1).
5. Once the robot arrives at the designated table. The customer takes the food.
6. The customer has to wave at the robot by its foot in front of the robot to make it move back to the kitchen.

Appendix D: Program Codes

```
#include <NewPing.h>

int realv;

#include <SPI.h>

#include <Pixy.h>

int x_c=0;

int h_c=0;

int input_ir=26;

#include <SoftwareSerial.h>

Pixy pixy;

int counter=0;

int dist1;

int dist2;

bool wval_flag=true;

bool flag=true;

bool flagger=true;

    bool f1=true;

        bool f2=true;

int checker;

int check;

    bool f3=true;

        bool f4=true;

            bool f5=true;

                bool f7=true;

                    bool f6=true;

int enA = 6;

int in1 = 2;
```

```

int in2 = 3;

int enB = 7;
int in3 = 4;
int in4 = 5;
NewPing sonar1( 40,42, 100);
NewPing sonar2( 38,36, 100);

SoftwareSerial b(8, 9);
String readdata;

void calc_dist2()
{

    delay(50);
    unsigned int uS2 = sonar2.ping();
    dist2=uS2 / US_ROUNDTRIP_CM;
    Serial.print(dist2);
Serial.println("cm");

if(dist2>25 || dist2==0 || dist2<5)
{
    dist2=99;
    dist1=99;

}
else
{Serial.println("Obstacle detected");
dist1=0;
    dist2=0;
}
}

```

```

    }

    }
void calc_dist1()
{

    delay(50);
    unsigned int uS1 = sonar1.ping();
    dist1=uS1 / US_ROUNDTRIP_CM;
    Serial.print(dist1);
Serial.println("cm");

if(dist1>25 || dist1==0 || dist1<5)
{
    dist1=99;

    }
    else
    {Serial.println("Obstacle detected");
    dist1=0;
    }

    }

void calc_dist()

```

```

{

    calc_dist2();
    calc_dist1();

}

void pixyGet()
{
    static int i = 0;
    int j;
    uint16_t blocks;
    char buf[32];

    // grab blocks!
    blocks = pixy.getBlocks();

    if (blocks)
    {
        i++;

        if (i%2==0)
        {
            sprintf(buf, "Detected %d:\n", blocks);

            for (j=0; j<blocks; j++)
            {
                sprintf(buf, "  block %d: ", j);
            }
        }
    }
}

```

```

    x_c= pixy.blocks[j].x;
    h_c= pixy.blocks[j].height;
    Serial.println(x_c);

if(dist1==99)
{
if(x_c>260)
{rights();
delayMicroseconds(180);
}
else

if(x_c<100)
{lefts();
delayMicroseconds(180);
}
else
{
backs();
delayMicroseconds(180);

}

}

else

if(dist1==0 || dist2==0)
{ Serial.println("Obs detected");
stops();}

```

```

        }
    }
}
else
{
    stops();
}

}

void setup() {
    Serial2.begin(9600);

    Serial.begin(9600);

    Serial.print("Starting...\n");
    pinMode(input_in, INPUT);
    pixy.init();

}
//-----
//

void stops()
{

    // now turn off motors

```

```
analogWrite(enA, 0);  
analogWrite(enB, 0);  
digitalWrite(in2, HIGH);  
digitalWrite(in4, HIGH);
```

```
}
```

```
// void left()
```

```
void rights()
```

```
{
```

```
analogWrite(enA, 10);
```

```
analogWrite(enB, 60);
```

```
}
```

```
// void right()
```

```
void lefts()
```

```
{
```

```
analogWrite(enA, 45);
```

```
analogWrite(enB, 15);
```

```
}
```

```

//void forward()
void forwards()
{

    analogWrite(enA, 30);
    analogWrite(enB, 30);

}

//void back()
void backs()
{
    analogWrite(enA, 30);
    analogWrite(enB, 30);
    digitalWrite(in3, HIGH);
    digitalWrite(in1, HIGH);

}

void counterloop()
{
Serial.println(counter);
if(digitalRead(input_ir)<1 && flag==true)
{counter=counter+1;
    flag=false;
}
if(digitalRead(input_ir)>0)
{
flag=true;
}
}

```

```

void getSerial2()
{

checker=check;

  if (Serial2.available()){
char c = Serial2.read(); //Conduct a serial read

  check=c;

Serial.println(check);
  if(c<47 || c>55)
  {
    if(wval_flag==true)
    {Serial2.write("Wrong value please say command again");
    delay(2000);
    wval_flag=false;
    setup();
    loop();
    }

if(checker==check)
{
wval_flag==false;
}
else
wval_flag==true;

```

```

getSerial2();
    }
    else
    {   realv=c-48;
        Serial.println("This is int value");

        Serial.println(realv);
        Serial.println("This is int value");
Serial2.write("RSS robot going to table");
Serial2.write(c);
Serial2.write("thanks for the command");

```

```

    if(realv==1)
    {}
    if(realv==2)
    {}
    if(realv==3)
    {}
    if(realv==4)
    {}
    gototable();
    }
    }
    // Serial2.write('a');
    }
void gototable()
{
    while(counter!=5)
    {
        counterloop();
        pixyGet();

```

```

calc_dist();
if(counter==1 && f1==true)
{Serial2.write("table 1 reached");
  stops();

  f1=false;
}
if(counter==2 && f2==true)
{Serial2.write("table 2 reached");
  stops();

  f2=false;
delay(300);
}
if(counter==3 && f3==true)
{Serial2.write("table 3 reached");
  stops();
  f3=false;
delay(300);
}
if(counter==4 && f4==true)
{Serial2.write("table 4 reached");
  stops();

  f4=false;
delay(300);
}
if(counter==5 && f5==true)
{Serial2.write("kitchen reached");
  stops();

  f5=false;

```

```

delay(300);

    }
    if(counter==realv)
    {

        while(dist1!=0 && flagger==true)
            {stops();
            calc_dist();

            if(fb==true)
            { Serial.println(" Stopped TABLE REACHED");
              Serial2.write("commanded table reached");

              Serial2.write("Please wave to continue");
              delay(300);

              fb=false;
            }

        if(dist1==0 || dist2==0)
        flagger=false;

            }
            if(f7==true)
            {Serial2.write("User waved moving ahead");
              delay(300);

              f7=false;
            }

    }

```

```

}

stops();
Serial.println("FINISHED LOOP");
    Serial2.write("done sir resetting values");
    delay(1000);

counter=0;
flagger=true;
f1=true;
    f2=true;

f3=true;

f4=true;

f5=true;
    f6=true;
    f7=true;
}

void loop() {
getSerial2();
//backs();
//counterloop();

//pixyGet();
//calc_dist();
}

```