AMERICAN INTERNATIONAL UNIVERSITY - BANGLADESH

2020-21, Fall

COURSE NAME: **MICROPROCESSOR AND EMBEDDED SYSTEMS**
SECTION: **F**

PROJECT REPORT

Project Title: **Table Token Generator and Indicator in Restaurant using Micro-controller**

SUBMITTED TO:

NAME: **TAHSEEN ASMA MEEM**

Group Members:

Name	ID
Mohammad Fahim Shahriar	18-37258-1
Nusrat Alam Chaiti	18-37417-1
Nyme Ahmed	18-36105-1
Methila Farzana Woishe	18-37706-1
Nila Sultana	18-36512-1
Tamanna Zaman Bristy	18-37685-1

Table Token Generator and Indicator in Restaurant using Micro-controller

First Author: Mohammad Fahim Shahriar (18-37258-1), Second Author: Nusrat Alam Chaiti (18-37417-1), Third Author: Nyme Ahmed (18-36105-1), Fourth Author: Methila Farzana Woishe (18-37706-1), Fifth Author: Nila Sultana (18-36512-1), Sixth Author: Tamanna Zaman Bristy (18-37685-1)

***Abstract*- Over the years, technology has tremendously revolutionized the restaurant industry. Much of the innovation has been with point-of-sale (POS) operations. But still there is some scope to improve the customer service. We tried to come up with a solution that makes interacting with waiters much easier, faster and more convenient. Instead of shouting or using some bells, Customers can generate a token by pushing a button .With this the waiter can be informed which table needs attention and can quickly respond to the needs of the customer. This will allow the customers to experience a quiet and fine dining atmosphere which in future will insure better customer loyalty and profit for the restaurant. This paper deals with designing a system that will make interacting with waiters as well as serving customers easier and more convenient.**

***Keywords*- Arduino, Restaurant, 7 segment display, Queue, Micro-controller, Table indicator, Token Generator.**

I. Introduction

A. State of the Art:

Mostly the present technology emphasizes on Billing and processing order but there are very few systems that are focused on giving customers quicker response based on customer needs. As per our research currently there is no system which is being developed in Arduino as a table token system. The closest system that is sort of similar is a quiz buzzer system that is used by educators. But it has a very specific focus with comparatively smaller economy focus. Our project is based on giving a better experience to customers by improving the response time. This system will make interaction of waiters and customers more convenient. The project can gain popularity as many restaurants and fast food chains are expanding as this is a fast industry but are having a lot of difficulty in managing customers as the business grows larger. Instead of relying on table-hosts memory, the system will give a notification buzzer that will indicate that a certain numbered table requires attention as well as store the sequence of table numbers that has pushed the buttons. As a result, a waiter can quickly respond to the needs of the many customers. This also removes the communication problem with waiters when they are low in numbers.

B. Motivation:

Giving a fine dining experience and great customer service was the goal of this project. The idea of this project comes from personal

experiences in restaurants and fast food chains. Our collective experiences showed a need for a project that addresses the customer waiter interaction problems. Nowadays, many restaurants manage their business by manual, especially taking customer orders manually or with waiters in inefficient ways. Some are using (POS) systems mostly for billing. There is a specific lack of use of technology in normal fast food chains. Upon conceptualizing the project idea, we looked for similar projects and ideas. The closest thing was the quiz buzzer system. We worked on the idea and implemented a new and more practical and economically profitable solution that will serve customers and employees alike. This system will make communication with the waiter and customer easier and faster. This can be achieved by designing a system where the customer will generate a token by pushing a button while sitting in a table. This will be indicated in the segment display by showing the number of the table. A noise will be made by a notification buzzer that will indicate that a certain numbered table requires attention. As a result, a waiter can quickly respond to the needs of the customer. Simultaneously, the customer doesn't have to shout or use some annoying bells to indicate his needs. And since there is an automated queue, the waiter can track the sequence of tables that need attention. This makes servicing and getting service in the restaurant faster, easier and more convenient.

II. Literature Review

A study of 94,404 customers visiting a popular Indian restaurant during a 12-month period had found that generally customers have to deal with a longer waiting time for basic restaurant services. Upon reviewing, it seems there lies a system where a user by using either the website, a mobile application, by phone, or, sometimes, on site, can reserve a table. The customer is immediately sent a confirmation email and an SMS. The automated queue management software updates the status whenever the queue position changes [1]. By gathering the concept of queue for reserving a table, we implement a queue system to track the sequence of table numbers that require services asked by customers using Arduino NANO. When customers click the push button of any table, the allocated table number will automatically be stored in the queue and display the queue. A waiter can use the queue to track the sequence of tables.

While looking for projects similar to ours, we come across some quiz buzzer systems that use a sort of similar principle but are used in a very different use case. To operate the quiz buzzer with the same idea of sounding an alarm sound and showing the team number that hits the button first, three distinct approaches were used. Often these systems are designed by 8051 processors. These approaches are the Fastest Finger First Indicator Circuit that uses all microprocessors' basic functionality, the Quiz Game Controller using the PIC16F877A microcontroller that uses this microcontroller's features along with UNL2003 and the Quiz Buzzer using the AT89C51 8051 microcontroller [2]. Though 8051 microcontroller is a part of an old generation microprocessor, we implement a different alteration of the idea using Arduino NANOs that serve a more commercial need.

For choosing the proper Arduino module, we reviewed papers to understand the properties and differences each model holds. However,

for our particular project, we took a modular approach for designing. To satisfy our goal, Arduino Nano was the better choice since it was very similar in core performance, with relatively less number of pins but with higher economic value, especially considering the modular design [3].

While reviewing more papers, we came across a paper that deals with increasing the interactive participation of students in learning by using Berkeley and Cristian algorithms[4].Since we are working with a system where increased instructiveness is not a primary concern, we didn't have to rely on any algorithms rather we rely on the human need of receiving services. As for increasing the input size, this can be achieved by adding a bigger Arduino board or increasing the number of smaller Arduinos for the purpose of modularity.

One of the cheap way of displaying the table number is using a 7-segment display. The display works on the principles of light emitting diodes that are forward biased. Combination of diodes represent a particular number in the display. In [5], we get a clear idea of how a countdown circuit works using Arduino and seven segment display. We use the information provided to code our Arduino to represent a 7-segment display.

One of the papers dealt with is to test student awareness on the combinatorial resistor relation calculation that can be either in sequence, parallels, or both in combination. By supplying the desired resistor output, the kit will ask questions while the students need to base the connection on the available resistor values in order to get the desired output resistor. Once the link is successfully built, the educational kit will deliver the response. The package is built with other components such as LCDs, LEDs, keypads, buzzers and resistors using the Arduino Uno Microcontroller. The key theme is to define the student's total resistance value by performing a basic Voltage Divider Rule definition (VDR) [6]. By getting the concept of these, we add a resistor box for connecting the switch of each table. This prevents short circuit of the entire system and keeps the system stable. We also use a general resistor in the transistor for the speaker which indicates that any table needs attention by pressing the allocated switch in the table. We further use a resistor in the dequeuer switch for removing a table number after serving that table.

A common error that happens in any restaurant that eventually leads to customer frustration is serving the wrong order. A unique menu was offered to each customer in consideration of their taste. Using new IT concepts such as Data Mining, Predictive Analysis and Artificial Intelligence, this idea is applied as a smartphone app. The specific food recommendation and order for each customer is created by analyzing their information on social media and the system notifies the customer by measuring the waiting time. To prevent confusion between orders and allow the customer to select the menu according to their taste in a minimum time, it provides a fully automated restaurant management system [7]. To avoid confusing between various orders of customers', we implement a token for each table. By following that token, the waiter will take the order individually and serially maintaining the queue. Customers can easily give their desired choice for ordering food when a waiter is coming to their table. This paper also gives a direction on what can be done on a future iteration of the project.

Low cost touch screen based Restaurant Management is a Personal Digital Assistant (PDA) based food ordering system. Framework as a workaround using an android Smartphone or tablet. The device consists of a smartphone/tablet holding the android at the customer table. Customers get all the info on the menu by using this application. The kitchen display of the customer tablet communicates directly with each other through Wi-Fi. Customer orders would enter the kitchen module immediately. This wireless application is user-friendly, saves time [8].These concepts give a vivid idea on what can be done on a large scale to restaurants using IoT based technology. The switch based approach for service is the first step towards eventual implementation of such a system. This also reduces the standing in line issue which is a bigger concern now considering the pandemic situation. Customers can easily press their allocated table switch for any services including placing orders. Table hosts will arrive to provide service according to the queue sequence.

Many people face a lot of problems in cafes, restaurants, hotels and lodges because of plentiful peak hours, the inconvenience of work and the handling of manual order. A paper suggests an automated automation device to order food, drinks, cleaning tables, etc. The order from the menu bar is transmitted on the wireless network through the kitchen. The LCD, keypad, and Bluetooth module are included in the menu bar. Using the electronic menu bar when a person places an order. Using the contact network, the order is sent to the reception and kitchen. Food is then transferred to the person [9]. Though the waiters of our project are doing their job manually, eventually we are planning to make an automated waiter in future for making our project more flexible and efficient.

III. Methodology and Modeling

A. Introduction:

Our project works on the basis of interaction between a customer and a waiter/waiters. Here, the customer calls for service, and the waiter provides it. Our project deals with managing this interaction in a way that makes it convenient and efficient. To do so, we designed our project that takes input from customer tables, display that specific table, uses sound to indicate the need of service and stores the sequence to make table management easier for the waiter. This process is also managed by the waiter/cashier. This project is a simulated design that can be constructed into an actual physical product.

To design the project, we used Arduino chips as our base and expanded upon that using its basic principles and properties. As a microcontroller, Arduino is a very convenient hardware that can be programmed to be used for implementing various projects. We programmed the Arduino based on our projects need and then added components with to finally design our desired system.

B. Working principle:

Our system is designed upon following some key principles. Using these rules, we have constructed the structure and workings of the project.

1. Switching Principle: Here we used switching principles of a switch circuit to change the states of Arduino pins from high to low. Since the system is normally in active high state, we use the switches to change a state to active low, which

works as our input that can be interpreted to provide specific output signals.

2. Arduino Coding Principle: Through Arduino coding, we have pre-defined some states and also set some instructions to be completed for those state changes. Through the coding, we check whether any state goes from active high to low and based on the pin number, we change the states of the output pins accordingly to get our desired outcome.
3. Transistor switching: Here when we get the high voltage on the base of the transistor, the emitter collector path will be active and current will flow through the speaker to ground. Thus, we will get speaker sound, which is also fine-tuned in the code section of Arduino.
4. Display Principle: It follows the principle of 7-segment display. Input is checked in Arduino and for a specific input, a specific output sequence is given which in turn lights up a specific number on the display.
5. Queue Principle: Here, using the basic principles of a queue, the button pressing sequence of the tables is stored which is displayed in the queue and can also be dequeuer using a dequeening switch

C. Description of the components:

1. IN4007 Diode: 1N4007 is a PN intersection rectifier diode. These sorts of diodes permit as it were the stream of electrical current in one course as it were. A diode may be a gadget which allows current streamflow as it were one heading. That's the current ought to continuously stream from the Anode to cathode. For 1N4007 Diode, the most extreme current carrying capacity is 1A it can withstand crests up to 30A. Subsequently ready to utilize this in circuits that are planned for less than 1A. The turnaround current is 5uA which is irrelevant. The control dissemination of this diode is 3W.
2. 2N2222 NPN Transistor: The 2N2222 is a common NPN bipolar intersection transistor (BJT) utilized for common reason low-power increasing or exchanging applications. It is planned for low to medium current, low control, medium voltage, and can work at decently tall speeds. Moreover, it is a NPN transistor thus the collector and emitter will be cleared out open (Turn around one-sided) when the base stick is held at ground and will be closed (Forward one-sided) when a signal is given to the base stick.
3. 7Seg-mpx1-cc Common Cathode Display: The seven segments shows are the most seasoned however one of the effective sorts of show utilized in implanted applications. This show has nothing more than 8 Led interiors. These 8 LEDs are isolated into each portion which can be named as a, b, c, d, e, f, g, Display as appeared within the picture over. These whole 8 section LEDs have one conclusion of their pins pulled out of the module as appeared over and the other closes are associated together and pulled out as the Common stick. This way we will control more than one segment at a time to speak to the numeric number 0-9 conjointly few Letter sets as appeared on the realistic picture underneath.
4. Arduino Nano R3: The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3. x). It has more or less the same usefulness of the Arduino Duemilanove, but in a distinctive bundle. It needs to be a DC control jack, and

works with a Mini-B USB cable rather than a standard one. The Arduino Nano is the smallest microcontroller within the Arduino family and has in this manner the least number of pins and associations.

5. SPST Push Button: The SPST push button is a basic ON/OFF switch, that's utilized to put through or break the association between two terminals. The control supply for the owl circuit is given by this switch. A basic PST switch has appeared underneath. The application of SPST switch is a light switch given underneath and it is additionally called as a flip switch. Push-button switches work on basic electronic components to enact or deactivate apparatus. A push-button can push-to-make or push-to break (SPST Transient Switch), where it makes or breaks contact with an electronic framework.
6. Loudspeaker: Loudspeaker, moreover called speaker, in sound generation, gadget for changing over electrical vitality into acoustical flag vitality that's transmitted into a room or open discussion. The term flag vitality shows that the electrical vitality contains a particular form, corresponding, for case, to discourse, music, or any other flag within the run of being capable of being heard frequencies (generally 20 to 20,000 hertz). The loudspeaker ought to protect the basic character of this flag vitality in acoustical shape. This definition of a loudspeaker prohibits such gadgets as buzzers, gongs, and sirens, in which the acoustical flag vitality does not compare in frame to the electrical flag.
7. Lamp**:** An electric lamp is a routine light emanating component utilized totally different circuits, primarily for lighting and demonstrating purposes. The development of light is very straightforward, it has one fiber encompassing which, a straightforward glass made spherical cover is given. The fiber of the light is basically made of tungsten because it has a high softening point temperature. A light radiates light vitality as the lean little tungsten fiber of light shines without being liquefied, whereas current streams through it.
8. LM016L LCD display: This LCD display is a very basic module and is very commonly used in various devices and circuits. It’s also a faster way to show status messages in our project. LM016L LCDs have 14 pins configuration. They are VSS, VDD, VEE, RS, RW, E, D0-D7. VSS pin connected to system ground. VDD pin powers the LCD with +5V (4.7V-5.3V). The VEE pin decides the contrast level of the display. Register Select (RS) pin connected to Microcontroller to shift between command/data register. Read/Write (RW) pin used to read or write data. Enable (E) pin connected to Microcontroller Pin and toggled between 1 and 0 for data acknowledgement. Data pins 0 to 7 form an 8-bit data line. They can be connected to Microcontroller to send 8-bit data.

D. Implementation:

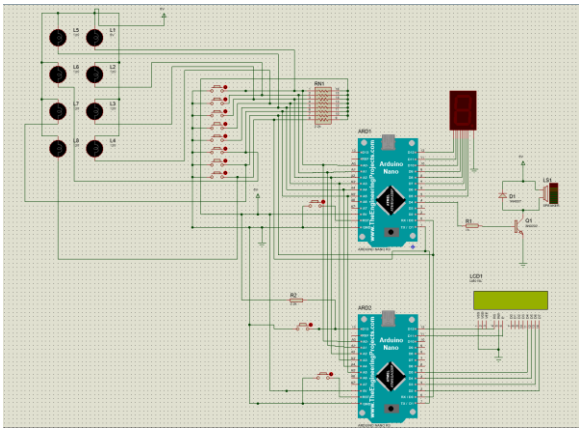

Figure 01: Circuit Design

Here, we have a switch network of 8 switches. One terminal of the network is connected with 8 input pins (A0-A5, D0-D1) of Arduino 1 and the other terminal is connected to ground. There is also a bulb network .One terminal of the bulb network is connected with 5V and another terminal is connected to the switch network. The bulb and table push buttons will be attached to the table. We then have a resistor box connecting with the Arduino input pins with 5V power, keeping them in active high state. The output pins (D12-D5) are connected to 7-segment display and D4 is connected with the speaker through a 2n2222 NPN transistor and 1N4007 diode. A reset push button is also attached with the reset pin. The output pins D2-D5, D11, D12 are connected with the 16x2 LED to display the queue.

The second Arduino uses the same input configuration as the first Arduino except one difference. Here, the D13 input pin is connected with a push button that works for dequeuer operation. The reset buttons and dequeuer button will be placed near the cashier table and the displays and the buzzer will be attached in a central position.

E. Measurement/ Test setup:

Our project tries to make easier and faster interaction between the customer and the waiter. The engineering problem that lies here is understanding basic principles of various electronics and using those principles to structure a project using physical components that will solve the initial problem of inefficient and unsatisfying customer waiter-interaction.

To tackle the problem, we have proposed our project throughout this paper. The key point of solving this issue lies in understanding how the system actually works.

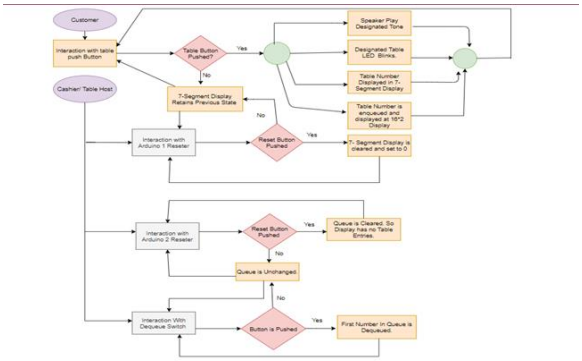

Figure 02: System diagram of the proposed project

When a customer pushes the switch, the 7-segment display will show the table no, the lamp attached to the table will blink, the speaker will play a tone and the table number will be stored in the queue. So the waiter will be alert and can provide service to that table while keep an eye on the queue to see the sequence of tables that need attention.

The cashier or the waiter/table-host can reset the 7-segment display when the service is provided and can dequeuer the serviced table no from the queue. They can also reset the queue.

In this way, the customer can quickly alert the waiter that he requires service without calling or shouting and the waiter can provide service to the customer quickly while also keeping track of the tables that he or others have to serve. So in this way, the project

solves our initial problem with satisfactory results. Further data on project effectiveness can be collected when it is implemented in real life using hardware and placed in actual restaurants.

F. Cost Analysis:

Serial	Product Name	Quantity	Price (BDT)
1	1N4007 Diode	1	1.36*1 = 1.36
2	2N2222 npn Transistor	1	2.25*1 = 2.25
3	7seg-mpx1-cc Common Cathode Display	1	10.39*1 = 10.39
4	Arduino Nano R3	2	200*2 = 400
5	SPST Push Button (2 pin)	11	10*11 = 110
6	Loudspeaker	1	100*1 = 100
7	Lamp	8	20*8 = 160
8	LM016L LCD Display	1	160*1 = 160
9	Resistor	2	1*2 = 2
10	8 Way Resistor Network	1	1*14 = 14
Total Cost			960.00

Figure 03: Total cost of the project

Here, we have chosen local vendors as suppliers to reduce cost since online outlets tend to price more. We have only included products that we actually need. However, quality and functionality has not been compromised to do so. A key property of this design is it is being modular .As a result, if parts of the device stop working, then it will be very easy to detect, troubleshoot and repair, thus making the device more consumer friendly. If Arduino UNO was used, the cost would have increased by 600tk with little extra benefit. While an Arduino mega can be used to implement logic of both Arduino Nanos, it will still be 700tk more expensive and due to using a single Arduino mega, we would have lost the modularity benefit of the project. This is why we used two Arduino Nanos that served both modularity and affordability and helped us create a functional project that is also under 1000tk and thus very much affordable for a single restaurant with up to 8 tables. For more tables, additional modules can be added as an upgrade package.

IV. Result and Discussion

A. Simulation:

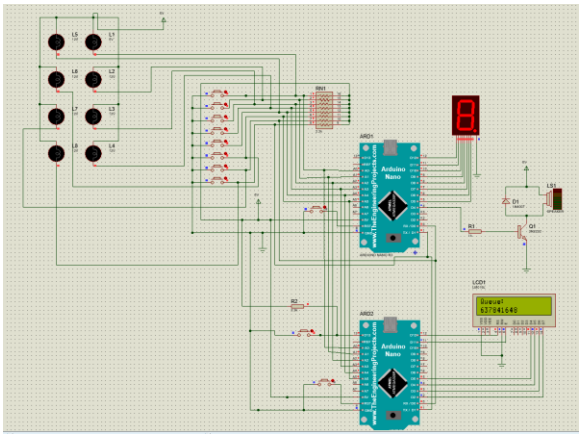

Figure 04: Circuit being simulated in Proteus

In previous sections, the circuit structure and workings have been explained. Here, we will explain how actually the system works.

The Arduino 1 is coded in a way so that the output pins go to high state in such a sequence that the 7-segment display shows 0.The Arduino 2 uses liquid crystal header file to initialize the output to work as the controller for the 16x2 LED display. Two strings are used in the coding of Arduino 2. One is used to show the word ‘Queue:’ and another string is used to work as a queue to store and modify table number sequence. The Arduino 1 also has an additional function that plays a user specified tone through the speaker.

Both Arduinos are coded in the same input configuration to check for state from high to low. Initially, the input pins are set to high by 5V supply. When a table button is pushed, that line is connected to the ground and so that input pin goes to low state. For this change in input state, the Arduino 1 sets the output pins accordingly to high state to show the table number in 7-segment display by powering specific diodes. The output pin

connected to the speaker will also go to high state and the speaker will play the tone. Meanwhile, Arduino 2 will concatenate the table number to the queue string. Also, since the lamp with the table is connected to the ground, the circuit will be complete and the lamp will blink.

The reset switches connects the reset pin to the ground and refreshes associated systems. Refreshing Arduino 1 sets the 7-segment display to 0 again and refreshing Arduino 2 will clear the queue string.

Arduino 2 has an additional button that works in the same principle as the input switches. Using it will remove the first number of the queue string thus dequeening the queue.

The codes for both Arduinos are given below:

```
//Arduino 1
int buzzer=4;
void setup()
{
pinMode(A0,INPUT); //analog pins as input
pinMode(A1,INPUT);
pinMode(A2,INPUT);
pinMode(A3,INPUT);
pinMode(A4,INPUT);
pinMode(A5,INPUT);
pinMode(0,INPUT);   //digital pins as input
pinMode(1,INPUT);
pinMode(12,OUTPUT);     //Digital     pins     as
//output for seven seg display
pinMode(11,OUTPUT);
pinMode(10,OUTPUT);
pinMode(9,OUTPUT);
pinMode(8,OUTPUT);
pinMode(7,OUTPUT);
pinMode(6,OUTPUT);
pinMode(5,OUTPUT);
pinMode(4,OUTPUT); // output for buzzer
// set initially display to 0
digitalWrite(12,HIGH); // seg a
digitalWrite(11,HIGH); // seg b
digitalWrite(10,HIGH); // seg c
digitalWrite(9,HIGH); //seg d
digitalWrite(8,HIGH); //seg e
digitalWrite(7,HIGH); // seg f
digitalWrite(6,LOW); //seg g // hiding the //middle
line causing display to show 0
digitalWrite(5,HIGH); //seg .
digitalWrite(4,LOW); // buzzer as off
}
void loop()
{
int a=digitalRead(A0); // reading pin values //for 8
buttons
int b=digitalRead(A1);
int c=digitalRead(A2);
int d=digitalRead(A3);
int e=digitalRead(A4);
int f=digitalRead(A5);
int g=digitalRead(0);
int h=digitalRead(1);
if(a==LOW) //if table 1 customer press //button
{
digitalWrite(11,HIGH); // print 1
digitalWrite(10,HIGH);
digitalWrite(9,LOW);
digitalWrite(8,LOW);
digitalWrite(7,LOW);
digitalWrite(6,LOW);
digitalWrite(12,LOW);
Alarm();
}
else if(b==LOW) //if table 2 customer press button
{
digitalWrite(11,HIGH); // print 2
digitalWrite(12,HIGH);
digitalWrite(9,HIGH);
digitalWrite(8,HIGH);
digitalWrite(7,LOW);
digitalWrite(6,HIGH);
digitalWrite(10,LOW);
Alarm();
}
else if(c==LOW)   //if table 3 customer press
button
{
digitalWrite(11,HIGH); // print 3
digitalWrite(12,HIGH);
digitalWrite(10,HIGH);
digitalWrite(9,HIGH);
digitalWrite(6,HIGH);
digitalWrite(7,LOW);
digitalWrite(8,LOW);
Alarm();
}
else if(d==LOW)   //if table 4 customer press
//button
```

```
{
digitalWrite(11,HIGH); // print 4
digitalWrite(10,HIGH);
digitalWrite(7,HIGH);
digitalWrite(6,HIGH);
digitalWrite(8,LOW);
digitalWrite(9,LOW);
digitalWrite(12,LOW);
Alarm();
}
else if(e==LOW)  //if table 5 customer press button
{
digitalWrite(10,HIGH); // print 5
digitalWrite(12,HIGH);
digitalWrite(9,HIGH);
digitalWrite(7,HIGH);
digitalWrite(6,HIGH);
digitalWrite(11,LOW);
digitalWrite(8,LOW);
Alarm();
}
else if(f==LOW)  //if table 6 customer press //button
{
digitalWrite(10,HIGH); // print 6
digitalWrite(12,HIGH);
digitalWrite(9,HIGH);
digitalWrite(8,HIGH);
digitalWrite(7,HIGH);
digitalWrite(6,HIGH);
digitalWrite(11,LOW);
Alarm();
}
else if(g==LOW)  //if table 7 customer press button
{
digitalWrite(10,HIGH); // print 7
digitalWrite(12,HIGH);
digitalWrite(9,LOW);
digitalWrite(8,LOW);
digitalWrite(7,LOW);
digitalWrite(6,LOW);
digitalWrite(11,HIGH);
Alarm();
}
else if(h==LOW)  //if table 8 customer press //button
{
digitalWrite(10,HIGH); // print 8
digitalWrite(12,HIGH);
digitalWrite(9,HIGH);
digitalWrite(8,HIGH);
digitalWrite(7,HIGH);
digitalWrite(6,HIGH);
digitalWrite(11,HIGH);
Alarm();
}
}
void Alarm()
{
digitalWrite(4,HIGH); //the buzzer sound
tone(buzzer,261,100); //Frequency of SA
delay(200);
tone(buzzer,293,100); //Frequency of RE
delay(200);
digitalWrite(4,LOW);  //without sound
//when delay time changed,the frequency //changed
}

//Arduino 2
// include the library code:
#include <LiquidCrystal.h>
//initialize the library by associating any //needed LCD interface pin
// with the arduino pin number it is connected //to
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
String  str1="";
void setup()
{
pinMode(A0,INPUT);
pinMode(A1,INPUT);
pinMode(A2,INPUT);
pinMode(A3,INPUT);
pinMode(A4,INPUT);
pinMode(A5,INPUT);
pinMode(0,INPUT);
pinMode(1,INPUT);
pinMode(13,INPUT);
//set up the LCD's number of columns and //rows:
lcd.begin(16, 2);
}
void loop() {
```

```
int a=digitalRead(A0); // reading pin values
//for 8 buttons for table and 1 for deque
int b=digitalRead(A1);
int c=digitalRead(A2);
int d=digitalRead(A3);
int e=digitalRead(A4);
int f=digitalRead(A5);
int g=digitalRead(0);
int h=digitalRead(1);
int i=digitalRead(13);
if(a==LOW)
{
str1=str1+1;
delay(300);
}
else if(b==LOW)
{
str1=str1+2;
delay(300);
}
else if(c==LOW)
{
str1=str1+3;
delay(300);
}
else if(d==LOW)
{
str1=str1+4;
delay(300);
}
else if(e==LOW)
{
str1=str1+5;
delay(300);
}
else if(f==LOW)
{
str1=str1+6;
delay(300);
}
else if(g==LOW)
{
str1=str1+7;
delay(300);
}
else if(h==LOW)
{
str1=str1+8;
delay(300);
}
else if(i==LOW)
{
str1.remove(0,1);
delay(200);
lcd.clear();
}
 // set the cursor to column 0, line 1
 // (note: line 1 is the second row, since
//counting begins with 0):
lcd.setCursor(0, 0);
lcd.print("Queue: ");
lcd.setCursor(0, 1);
lcd.print(str1);
}
```

B. Experimental Result:

Switch No.(Random Selection)	Associated Lamp State	Input low Arduino Pin (Active low)	Output Pins(0=Low,1=High)								7 Segment Display Output	Active LED DIODES	Queue Output
			12 (a)	11 (b)	10 (c)	9 (d)	8 (e)	7 (f)	6 (g)	5 (.)	4 (buzzer)			
No button pushed	NULL	None	1	1	1	1	1	1	0	1	0	0.	a,b,c,d,e,f,	
1	BLINKS	A0	0	1	1	0	0	0	0	1	1	1.	b,c,.	1
2	BLINKS	A1	1	1	0	1	1	0	1	1	1	2.	a,b,d,e,g,.	12
3	BLINKS	A2	1	1	1	1	0	0	1	1	1	3.	a,b,c,d,g,.	123
Arduino1 reset(Display)	NULL	RST(Arduino 1)	1	1	1	1	1	1	0	1	0	0.	a,b,c,d,e,f,.	123
5	BLINKS	A4	1	0	1	1	0	1	1	1	1	5.	a,c,d,f,g,.	1235
6	BLINKS	A5	1	0	1	1	1	1	1	1	1	6.	a,c,d,e,f,g,.	12356
Arduino2 reset(Queue)	NULL	RST(Arduino 2)	1	0	1	1	1	1	1	1	0	6.	a,c,d,e,f,g,.	
8	BLINKS	D1	1	1	1	1	1	1	1	1	1	8.	a,b,c,d,e,f,g,.	8
4	BLINKS	A3	0	1	1	0	0	1	1	1	1	4.	b,c,f,g,.	84
7	BLINKS	D0	1	1	1	0	0	0	0	1	1	7.	a,b,c,.	847
Deque	NULL	D13	1	1	1	0	0	0	0	1	0	7.	a,b,c,.	47

Figure 05: Data table of the system

- Each table switch has an associated lamp attached with it which will blink when switch is pushed. The reset buttons have no lamps attached with them.
- A0-A5, D0, D1, RST pins are input pins that functions when in active low state.
- D12-D5 are output pins that are attached to diodes of seven segment display, D4 is attached to buzzer circuit.
- Depending on the diode combination, seven segment display displays a number, indicating the table needing attention.

- Pressing the reset button of Arduino 1 will reset the segment display while pressing the reset button of Arduino 2 will reset the queue.

Since the system is designed based on HIGH and LOW logic, the theoretical and testing values will always match. If they don't match, then that indicates manufacturing defects on the hardware or buggy code compiled for Arduino chips.

C. Software Based Comparison:

In our original/initial version of the project, we implemented the system in tinker cad. Since it was an early iteration, there were some significant limitations. Comparing the final Proteus version and the tinker cad version not only provides a decent comparison between the two projects and two different mediums of project design, but also gives insight on how the project progressed over time.

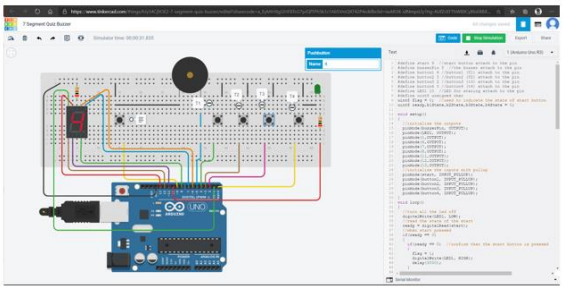

Figure 06: Original Tinker-cad version

1. The initial version had an extra start switch which had to be turned on every time manually to allow input from the table. This method was heavily inefficient as there can be multiple inputs at a time from the tables but these won't be registered thus creating a bad scenario. On the Proteus version, there is no such hassle, multiple inputs are counted without the need of manual intervention by the waiter. The system is plug and play based. So no start switch required.
2. If two customers press buttons at the same time, then there is no way of handling both of them in the tinker cad build. In the Proteus version, this has been handled by the inclusion of a queue.
3. Since the tinker cad version has no queue associated, the features of enqueuer and dequeuer cannot be implemented. The Proteus version has a robust automated enqueuer and a manual dequeuer system.
4. Comparing the two software tools side by side, Proteus provides more functionality, equipment libraries and features compared to tinker cad. Tinker cad lacks in component diversity and implementation methods. This is very apparent in this project as there is no built in Arduino Nano chip. As a result, we have to implement the project using Arduino Uno, which for our purpose, is expensive.
5. Another difference in implementation between Proteus and tinker cad is that tinker cad will require a breadboard for proper arrangement of the components. Due to the constructions of the breadboard, a good amount of pins remain unused to prevent congested circuits. This means we need two breadboards in tinker cad to develop this project which will not only increase the size of the system but also will introduce additional cost for breadboards.

From these comparisons, we can conclude that while Proteus and tinker cad are similar, Proteus overall suits better for our project due to the vast component library and more robust system. It is also apparent that the project we have done in Proteus has improved significantly from our initial tinker cad design.

V. Impact of professional engineering solutions on society and environment

The problem we solved through our project has a vast impact in social-economic progress in the long run. Our project can enhance the customer waiter relation by making the service giving-taking process easier and faster. This in turn, will ensure more satisfied customers resulting in increasing number of customers coming to the restaurant. So the restaurant can now profit more. Since restaurant industry is competitive, other competitors will also try to buy this project. This will greatly benefit us. Meanwhile this increased use of technology in fast food chains or restaurants will shift the industry in a more tech focused situation. Restaurants will more likely buy tech solutions and install them. This means more opportunity for us to grow as a manufacturer. Eventually this shift will improve the economic condition of the society.

Our capstone project also plays a very important role in our society in various ways in the current pandemic situation. People have to maintain social distance to prevent the spread of COVID'19 virus. By utilizing the queue process, we can prevent standing on a line and can save us and others from the virus.

Our project also brings benefits to a person with a physical handicap who can't easily move around without assistance. They can simply push the button for ordering their foods and also use it as a signal for assistance.

Overall our project not only solve our initial engineering problem, but also introduces new opportunities and benefits that helps and improves the social and economic wellbeing of our society.

VI. Conclusion

In this capstone project, we tried to design a system that will make customers get service from waiters easier and more convenient. The idea came to us from personal experiences that convinced us that there is need for such a system and after some in-depth searching, we realized this is not a common technology and so we saw potential in such a system that can really push the restaurant industry forward. To do so, we used our existing knowledge of microcontrollers and electronics and used basic electronics principles to construct a design to serve our purpose.

In each iteration, we have faced newer challenges and limitations and we worked on them to improve the quality of the project. While we have made significant progress from our initial design, we still have some limitations such as:

1. Reset switch and dequeuer switch are fixed in a position through wires.
2. Lack of LED included push buttons.
3. No system to calculate and store the delay between pressing the button and getting the service from table-host.

These problems can be fixed in future implementations. We can fix them by introducing:

1. Wireless reset/dequeuer button using Wi-Fi modules.
2. Smartphone App used by Cashier/ Table host for rapid dequeening/ resetting.
3. Database system to store delay of interaction and service provided.
4. LED push button without additional lamp.

Since we are dealing with a pandemic, currently we couldn't implement the project using hardware. But as pandemic situation starts to fade, we want to manufacture this

project for industrial production and sell it to restaurants. We have also realized the effect this project can have on social-economic conditions of the society .This is a project that can provide good service with low cost thus bringing a fair bit of economic value in society. Since it is a fairly low cost and efficient system, there is opportunity for good profit margins. And with proper funding and marketing, we expect to make it as a successful commercial product.

References

[1] J. D. Vries, D. Roy, and R. D. Koster, " Journal of Operations Management Volume 63, November 2018, Pages 59-78," Journal of Operations Management, vol. 63, pp. 59–78, Nov. 2018.

[2] Y. B. Sundaresan and K. Perumal, "A Comparative Analysis of Low Cost Solutions for Quiz Controller and Classroom Access Control Systems," Research Journal of Pharmacy and Technology, Sep. 2016.

[3] L. Louis, "WORKING PRINCIPLE OF ARDUINO AND USING IT AS A TOOL FOR STUDY AND RESEARCH," International Journal of Control, Automation, Communication and Systems, vol. 1, Apr. 2016.

[4] H. Hayat, P. Patel, and A. Salgado, "Distributed Quiz," International Journal of Control, Automation, Communication and Systems, vol. 1, Apr. 2016.

[5] U. C. N. and P. A. A., "Development Of An Arduino Based Countdown Time Reminder For Conduct Of Examination In Institutions Of Learning," Journal of Multidisciplinary Engineering Science Studies, vol. 4, no. 11, Nov. 2018.

[6] A. F. Z. Abidin and A. Adam, Resducational Kit The Development of An Electronic Quiz Board That Test Engineering Students Knowledge on Resistors Concept in Electrical Circuit, vol. 8, no. 1, May 2019.

[7] V. Liyanage, A. Ekanayake, H. Premasiri, P. Munasinghe and S. Thelijjagoda, "Foody - Smart Restaurant Management and Ordering System," 2018 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Malambe, Sri Lanka, 2018, pp. 1-6, doi: 10.1109/R10-HTC.2018.8629835.

[8] M. D. Jakhete and P. C. Mankar, "Implementation of Smart Restaurant with e-menu Card," International Journal of Computer Applications, vol. 119, pp. 23–27, 2015.

[9] S. Afsheen, L. Fathima, Z. Khan, and M. Elahi, " A self-sufficient waiter robo for serving in restaurants," International Journal of Advance Research, Ideas and Innovations in Technology, vol. 3, no. 5, May 2018.