CHAPTER-1
INTRODUCTION

1.1 INTRODUCTION

The present strategy for menu requesting framework incorporates additional human endeavors for getting a request from the client, by giving the client a menu card on their table and furthermore charging is an exceptional consideration need to pay for each table and their orders. Thus, the menu requesting through an electronic framework interface will get an extreme reaction from the clients because of the efficient approach and more brilliant approach to convey. The remote correspondence interface will give a quicker and exact information transmission in a minimal expense. The fundamental intend to carry out this e-menu requesting framework is it's easy to understand interface just as to diminish human endeavors. The execution of electronic menu requesting frameworks might have a few contrasts in interface plan and procedure. The zig-honey bee correspondence is utilized as remote interface and the graphical LCD show with contact screen is utilized as client interface. In the new past there has been development in requesting and serving framework yet the outcomes are not particularly encouraging. With the headway in correspondence innovation the issues of being late engaged can be tackled. To viably run an eatery, efficient and cost enhancements are fundamental. Decrease on schedule by a couple of moments for each table can accelerate request handling, increment proficiency and lift benefits.

Eatery is a public spot which opens to all to offer food and refreshment to people groups. Eatery business is quite possibly the most beneficial business. Hence, the significance of food serving is of extraordinary importance. Throughout the long term, food and the overall occupation of serving have developed such a lot of that requirement for assistance and mechanization has been expanded. With the progression of mechanization cafés are expected to refresh with programmed framework for saving time in running an eatery. In late eatery frameworks, the servers keep the record of clients' orders and

afterward request goes to kitchen for planning. The proposed framework is the eatery menu requesting framework dependent on Zigbee. Programmed menu requesting framework through an electronic framework interface can be an ideal answer for work on the fields of eatery. In customary eateries the orders are taken from the clients by the servers. Paper menu needs additional human endeavors to get the orders from the clients. The menu cards are set on their tables.

1.2 AIM OF THE PROJECT

The system mainly aims in designing completely automated menu system in restaurants with the help of touch screen sensor and a color graphical LCD to provide a user friendly environment. There is no need of a person to take the order from the customer's table. The menu will be displayed automatically on the customer's table and we can directly order the menu with the help of either touch screen sensor. Now a days every restaurant is using pen and paper to place the order from the customer. This is wasting a lot of time in processing the order, the time is also required for sending the order that is placed to the kitchen. As this has to be changed and a new system has to be made, the proposed system aims in designing a fully automated ordering of menu system in the restaurants by using a touch screen, RFID, Zigbee and a LCD as the major components which will provide a friendly environment to the user.

1.3 METHODOLOGY

The background methodology involves the study about the wireless technologies in the market, alternatives for display methods and also about the bill processing and claim methods. There are various wireless technologies in the market in their category of communication ranges. While choosing a communication technology for out implementation, the first concern to make is, the requirement of communication range. The communication technology to be used should always be enough capable of providing the range of communication as per the application requirement and the frequency band should be enough to carry by the hardware implemented. The next concern about the communication

technologies is to choose the less expensive technology which will also satisfy the frequency range. Apart from this all, one more concern is about the modulation technique using in the communication technology. The modulation technique will effects the service quality in data exchange. The next step of research is about the interface/display technologies, the interface involves displaying the menu items on any output device. Since our proposed system consists of a portable handheld device for menu display at every table, it should be always less expensive and easy to operate by anyone. The portable interfaces can be used with microcontroller are having the choices like Alphanumeric LCD display, that it can display alphanumeric characters on it. The research about the billing methods followed by the most of the restaurants is all manual billing method by monitoring the items issued to a particular table, and finally they will issue a paper statement of bill to the customer.

1.4 SIGNIFICANCE OF THE WORK & APPLICATIONS

A smart restaurant menu ordering system using Zigbee technology can offer several benefits and applications to both restaurants and customers. Zigbee is a wireless communication protocol that can connect and control various devices in a restaurant setting. Here are some applications of such a system:

- **Contactless Ordering:** With the ongoing concern about hygiene and social distancing, a Zigbee-based menu ordering system allows customers to browse the menu and place orders from their smartphones, reducing the need for physical menus and face-to-face interactions with waitstaff.

- **Faster Service:** Customers can place orders directly from their tables, reducing the waiting time for service. This can lead to faster turnaround times and increased customer satisfaction.

- **Menu Customization:** Customers can easily customize their orders, making special requests or dietary preferences through the app, ensuring they receive their preferred dishes.

- **Language Support:** The system can support multiple languages, making it more accessible for international customers or those who may not be fluent in the local language.

- **Allergen Information:** The app can provide detailed information about allergens and ingredients, helping customers with food allergies or dietary restrictions make informed choices.

- **Loyalty Programs:** Restaurants can integrate loyalty programs into the app, allowing customers to earn rewards and discounts for repeat visits and orders.

- **Order Tracking:** Customers can track the status of their orders and receive notifications when their food is ready, reducing the need to check with waitstaff.

- **Kitchen Efficiency:** Orders placed through the Zigbee system can be directly transmitted to the kitchen, reducing errors in communication and streamlining food preparation.

- **Inventory Management:** The system can help the restaurant manage its inventory more efficiently by tracking popular dishes and ingredients, enabling better supply chain management.

- **Table Management:** The restaurant can track the occupancy of tables in real-time, helping with reservation management and ensuring that no table remains unoccupied.

CHAPTER-2
LITERATURE SURVEY

2.1 LITERATURE SURVEY

Harshada S Wabale introduced Automatic Menu Ordering System utilizing Zigbee and Arm Processor. All the data of suppers and record right off the bat recorded in this framework and director can get all the data from this framework. The client is track by 15693 RFID TAG as ID card. They get table number and dinners data from this tag. The counter uses this framework to take requests of client, and afterward at kitchen side this dinner data is gotten. At the point when the clients go into the eatery they get E-tag from the counter. They can pick any seat from this E-tag and afterward put that E-tag on detecting module on the table. This module will give all the data to framework by Zigbee. From this label framework recognize the situation of the client. Then, at that point server will serve the food as per the needs. B. Shabari, B. Ashok Nayak, August 2015, proposed Zigbee based E-menu requesting framework. The 802.15 Zigbee technology is utilized as remote correspondence standard.

The framework will comprise of two segments, one is a hand held gadget put on each table in the café and one more segment ought to be put at charging segment and supply area. The paper portrays about the calculation utilized in execution of cutting edge menu requesting framework by with a remote correspondence innovation zigbee and the means engaged with its convention stack. The framework additionally has a touch screen and graphical LCD interface for giving a more brilliant UI menu requesting. Prof. Dr. Usman Ali Shah, Faraz Ali, Sana Sohail, Haris Khan, May 2016, gave Intelligent Robotic Waiter Menu requesting System. The menu card is given utilizing android application. The request will be shipped off the counter comprising of PC through Wi-Fi connect and the subtleties will be saved in the data set of the counter. It expects Graphical User Interface (GUI) progressed contact screen module is used as menu requesting framework. Client can orchestrate through

this touch screen device put on each table in eatery. HTML is used for planning pages of cafe.

2.2 EXISTING SYSTEM

The present technique for menu requesting framework incorporates additional human endeavors for getting a request from the client, by giving the client a menu card on their table and furthermore charging is a unique consideration need to pay for each table and their orders. Thus, the menu requesting through an electronic framework interface will get an extreme reaction from the clients because of the efficient strategy and more intelligent approach to convey. The remote correspondence interface will give a quicker and precise information transmission in a minimal expense. The primary expect to carry out this emenu requesting framework is its easy to use interface just as to decrease human endeavors. The execution of electronic menu requesting frameworks might have a few contrasts in interface plan and strategy. The zig-honey bee correspondence is utilized as remote interface and the graphical LCD show with contact screen is utilized as client interface. In the new past there has been development in requesting and serving framework yet the outcomes are not especially encouraging. With the progression in correspondence innovation the issues of being late engaged can be tackled. To viably run a café, efficient and cost enhancements are fundamental. Decrease on schedule by a couple of moments for each table can accelerate request preparing, increment proficiency and lift benefits.

2.2.1 DISADVANTAGES OF EXISTING SYSTEM:

- Even though this process looks very simple, a waiter often makes errors while the note making or sending the order to the kitchen.
- Writing wrong room number, wrong meat temperature or some specific orders cause delays of the food and make customer having unsatisfied experience.

2.3 PROPOSED SYSTEM:

The project is proposed with the Zigbee innovation as the correspondence medium which carries out quicker requesting framework. The innovation ready to tackle need number of laborer, lessens the blunder on requesting food varieties by the clients. The e-menu food requesting framework depends on programming equipment foundation of Arduino (ATMega328p) and utilizing Zigbee short reach radio correspondence innovations. We have separated the framework in two areas one is handheld segment (client segment) and other is fundamental segment (proprietor segment), both segment comprises of Zigbee handsets. At handheld area GLCD with contact screen is given to submit the request and request sends further to primary segment through Zigbee handset. All the while ringer will demonstrate that request has shown up and LCD show which is at fundamental segment is utilized to show food menu request and cost. Café is a public spot which opens to all to offer food and refreshment to people groups. Cafe business is quite possibly the most beneficial business.

In this way, the significance of food serving is of incredible importance. Throughout the long term, food and the general occupation of serving have developed such a lot of that requirement for help and computerization has been expanded [1]. With the headway of robotization eateries are expected to refresh with programmed framework for saving time in running an eatery. In ongoing café frameworks, the servers keep the record of clients' orders and afterward request goes to kitchen for arrangement. The proposed framework is the café menu requesting framework dependent on Zigbee. Programmed menu requesting framework through an electronic framework interface can be an ideal answer for work on the fields of café. In conventional cafés the orders are taken from the clients by the servers. Paper menu needs additional human endeavors to get the orders from the clients. The menu cards are set on their tables. Along these lines, customary café framework needs human endeavors and it devours the time. The proposed framework can save time and diminish labor.

2.3.1 ADVANTAGES OF PROPOSED SYSTEM

A smart restaurant menu ordering system using Zigbee technology offers various advantages for both customers and restaurant owners. Some of the key benefits include:

- Efficient ordering
- Reduced Wait Timers
- Accurate Orders
- Improved Table Turnover
- Menu Flexibility
- Contactless Dining
- Data Collection
- Enhanced Customer Experience
- Personalization
- Reduced Paper Waste
- Cost Savings
- Scalability
- Ease of Use
- Remote Management.

CHAPTER 3
HARDWARE DESCRIPTION

3.1 ARDUINO UNO

The Arduino Uno is a microcontroller board based on the ATmega328 It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to serial converter. "Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

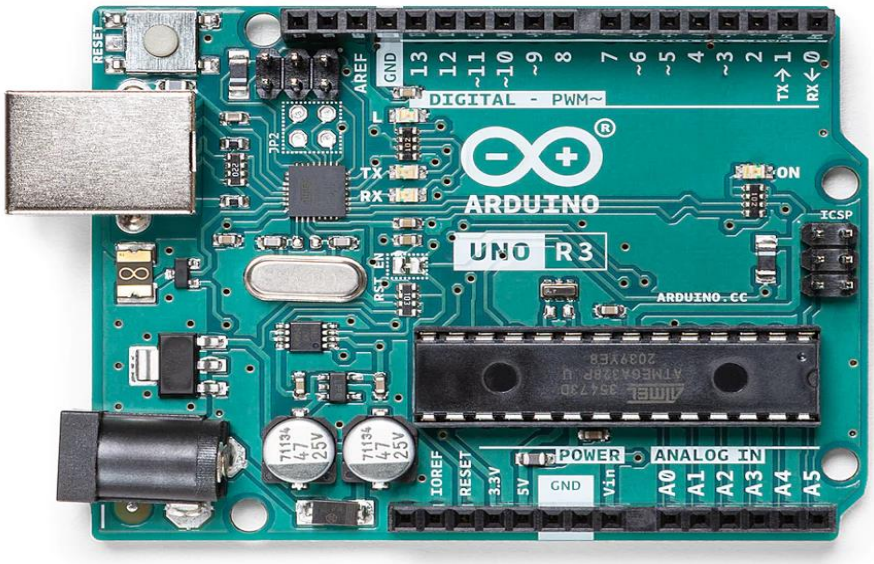


Fig 3.1: Arduino UNO

Arduino UNO is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button.

- **ATmega328p Microcontroller-** It is a single chip Microcontroller of the ATMEL family. The processor code inside it is of 8-bit. It combines Memory (SRAM, EEPROM, and Flash), Analog to Digital Converter, SPI serial ports, I/O lines, registers, timer, external and internal interrupts, and oscillator.

- **ICSP pin -** The In-Circuit Serial Programming pin allows the user to program using the firmware of the Arduino board.

- **Power LED Indicator-** The ON status of LED shows the power is activated. When the power is OFF, the LED will not light up.

- **Digital I/O pins-** The digital pins have the value HIGH or LOW. The pins numbered from D0 to D13 are digital pins.

- **TX and RX LED's-** The successful flow of data is represented by the lighting of these LED's.

- **AREF-** The Analog Reference (AREF) pin is used to feed a reference voltage to the Arduino UNO board from the external power supply.

- **Reset button-** It is used to add a Reset button to the connection.

- **USB-** It allows the board to connect to the computer. It is essential for the programming of the Arduino UNO board.
- **Crystal Oscillator-** The Crystal oscillator has a frequency of 16MHz, which makes the Arduino UNO a powerful board.
- **Voltage Regulator-** The voltage regulator converts the input voltage to 5V.
- **GND-** Ground pins. The ground pin acts as a pin with zero voltage.
- **Vin-** It is the input voltage.
- **Analog Pins-** The pins numbered from A0 to A5 are analog pins. The function of Analog pins is to read the analog sensor used in the connection. It can also act as GPIO (General Purpose Input Output) pins.

Arduino Uno is an open-source microcontroller board based on the processor ATmega328P. There are 14 digital I/O pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains all the necessary modules needed to support the microcontroller. Just plug it into a computer with a USB cable or power it with an AC-to-DC adapter to get started. In the event of a worst-case scenario, one could replace it with a new one as the Uno is very economical compared to other boards like raspberry pi, STM, etc. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega1612 (Atmega81/2 up to version R2) programmed as a USB- to-serial converters.

ATmega328P is a very advance and feature rich microcontroller. It is one of a famous microcontroller of Atmel because of its use in ARDUINO UNO board. It is a microcontroller from the Atmel's mega MVR microcontrollers family

(Later in 2016 the Atmel is obtained by Microchip Technology Inc, the microcontrollers manufactured in mega MVR family are designed for handling larger program memories and each microcontroller in this family contains different amount of ROM, RAM, I/O pins and other features and also they are manufactured in different output pins which are from 8 pins to hundreds of pins.

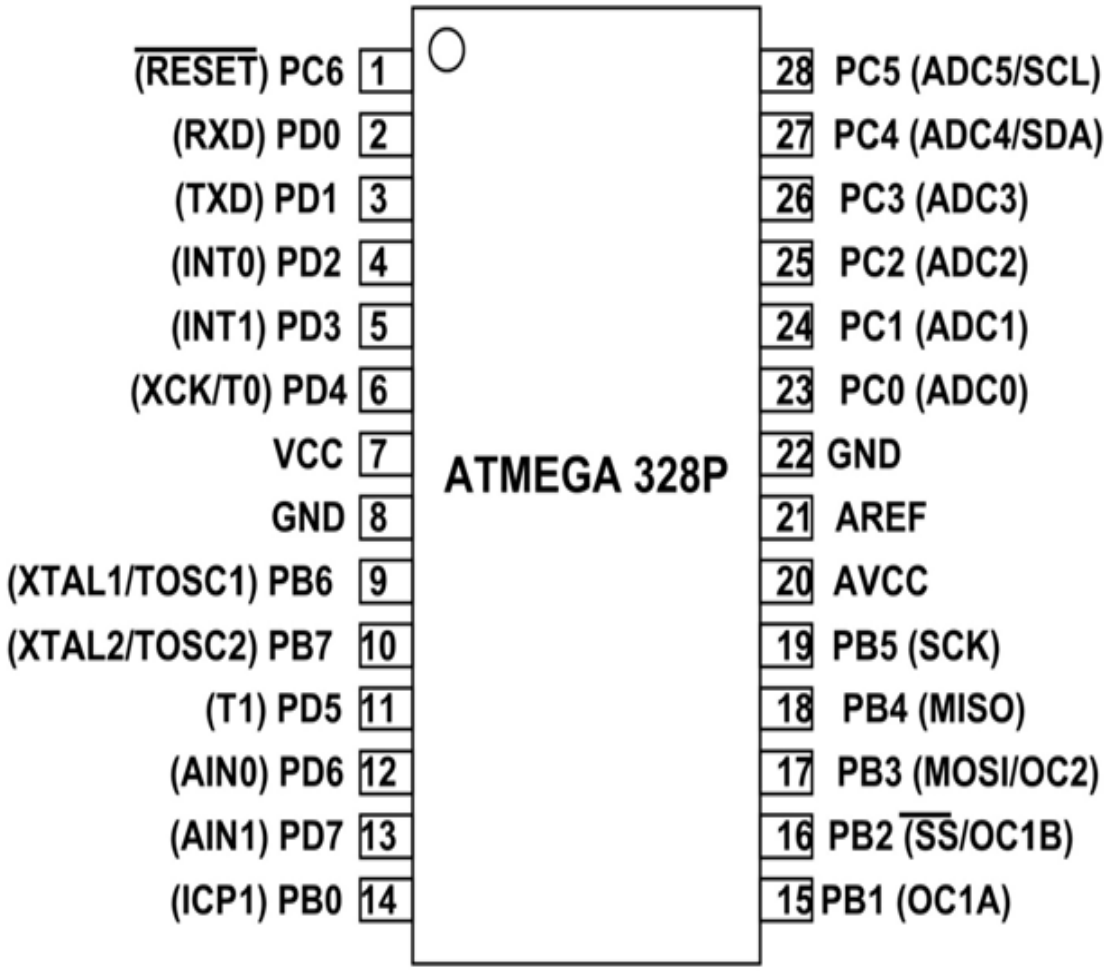


Fig 3.2: Pin Diagram

The internal circuitry of **ATmega328P** is designed with low current consumption features. The chip contains 32 kilobytes of internal flash memory, 1 kilo bytes of EEPROM and 2 kilo-bytes of SRAM. The EEPROM and the flash memory are the memories which saves information and that information still exits every-the power is disconnected or off but the SRAM is a memory which only saves the information until the power is supplied and when the power is disconnected all the information saved in SRAM will be erased.

3.1.1 SPECIFICATIONS

- Microcontroller: ATmega328P
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- In-out Voltage (limit): 6-20V
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- PWM Digital I/O Pins: 6
- Analog Input Pins: 6
- DC Current per I/O Pin: 20 Ma
- DC current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB (ATmega328P) of which 0.5 KB used by bootloader
- SRAM: 2 KB (ATmega328P)
- EEPROM: 1 KB (ATmega328P)
- Clock Speed: 16 MHz
- LED_BUILTIN: 13
- Length: 68.6 mm
- Width: 58.4 mm

3.1.2 APPLICATIONS

- Used in ARDUINO UNO, ARDUINO NANO and ARDUINO MICRO boards.
- Industrial control systems.
- SMPS and Power Regulation systems.
- Digital data processing.
- Analog signal measuring and manipulations.
- Embedded systems like coffee machine, vending machine.
- Motor control systems.
- Display units.
- Peripheral Interface system.

3.2 ZIG-BEE TRANSCEIVER

A Zigbee module is a hardware device that provides wireless communication between devices using the Zigbee protocol. Zigbee is a low-power wireless protocol designed for the Internet of Things (IoT) applications. It uses a mesh network topology, which means that devices can communicate with each other directly or through intermediate nodes, called routers or coordinators.

A Zigbee module typically includes a microcontroller, a Zigbee transceiver, and an antenna. The microcontroller is responsible for managing the communication between the module and the rest of the system. The Zigbee

transceiver allows the module to transmit and receive Zigbee signals, while the antenna is used to propagate these signals.

Zigbee modules can be used in a wide range of applications, including home automation, industrial automation, healthcare, and smart cities. They offer a low-power and low-cost solution for connecting IoT devices, enabling remote monitoring, control, and automation of various systems.

Fig 3.3: Zigbee Transceiver

3.2.1 SPECIFICATIONS OF ZIGBEE MODULE

Zigbee transceivers offer several features and characteristics that make them suitable for various applications, especially in the realm of low-power, short-range wireless communication. Here are some of the key features of Zigbee transceivers:

- **Low Power Consumption:** Zigbee transceivers are designed for low-power operation, making them ideal for battery-powered devices. They can operate for extended periods on a single set of batteries, which is crucial for many IoT and wireless sensor applications.

- **Mesh Networking:** Zigbee transceivers support mesh networking, allowing devices to communicate with each other in a self-organizing and self-healing network. This feature improves the network's reliability and coverage.

- **Low Data Rate:** Zigbee is intended for applications that don't require high data rates. It's well-suited for transmitting small amounts of data efficiently, such as sensor readings and control commands.

- **Reliability:** Zigbee uses a robust communication protocol, which includes mechanisms for collision avoidance and retransmission of lost packets. This helps ensure reliable data transmission in noisy environments.

- **Long Range:** While Zigbee is primarily designed for short-range communication, its range can be extended by using multiple hops in a mesh network. The range typically varies depending on factors like power output and environmental conditions.

- **Low Latency:** Zigbee offers low latency communication, making it suitable for applications where real-time control and responsiveness are required.

- **Security:** Zigbee includes built-in security features, such as encryption and authentication, to protect the data transmitted within the network. This is important for maintaining the privacy and integrity of the data.

- **Interoperability:** Zigbee Alliance, the organization behind the Zigbee standard, ensures interoperability between different Zigbee-certified devices, allowing various manufacturers' products to work together seamlessly.

- **Frequencies:** Zigbee operates in various frequency bands, including 2.4 GHz (worldwide) and 868 MHz and 915 MHz (region-specific), allowing for regulatory compliance and deployment flexibility.

- **Scalability:** Zigbee networks can scale from small setups with just a few devices to larger networks with hundreds of devices, making it versatile for different applications.
- **Ease of Installation:** Zigbee's self-configuring and self-healing mesh network topology simplifies the installation process, as devices can be added or removed without disrupting the network.
- **Wide Range of Applications:** Zigbee transceivers are used in a variety of applications, including smart home automation, industrial control and monitoring, healthcare, asset tracking, and more.

These features make Zigbee transceivers a popular choice for many IoT and wireless sensor network applications that require reliable, low-power, and short-range wireless communication.

3.2.2 APPLICATIONS OF ZIGBEE

- Home automation.

- Wireless sensor networks.

- Industrial control systems.

- Embedded sensing.

- Medical data collection.
- Smoke and intruder warning.
- Building automation.
- Remote wireless microphone configuration.

3.2.3 HC-12 WIRELESS COMMUNICATION MODULE

It is a wireless data transmitter and receiver module, that uses 433 megahertz frequency and can communicate to one thousand meter distance. It also has ability to communicate with more than one microcontrollers existing at longer distances. This module starts its operation at 3.2 volts to 5.5 volts, for its safe operation, it is also connected with a capacitor to reduces ripples in the output of battery. This Bluetooth module installed in industries to control different process and machines. It also used in the circuitry of different security systems. This module uses silicon's LABs Si4463 for (radio-frequency) RF data transmission features. In today's post, we will have a look its working, features, pinout and applications in different circuits.

Fig 3.4: HC-12 Module

3.2.4 CP2102 UART MODULE

CP2102 UART Module integrated with CP2102 IC from SiLabs, which is a single-chip USB to UART Bridge IC. This USB-to-UART bridge controller provides a simple solution to update the design of the RS-232 using minimal components and PCB space. Royalty-free Virtual COM Port (VCP) allows device drivers as a COM port in PC applications. This module allows asynchronous serial data bus (UART) with a USB 2.0 full-speed function controller, USB transceiver, and full modem control signal.

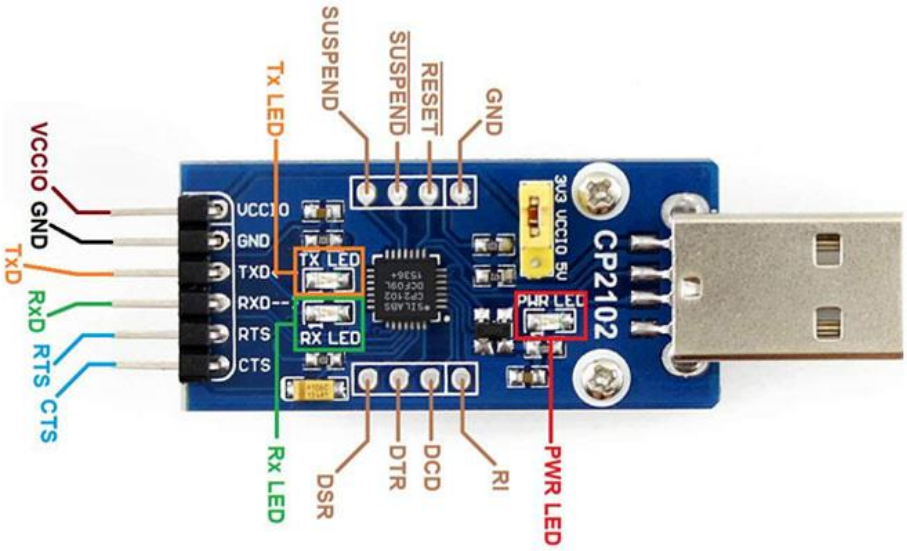


Fig 3.5: CP2102 UART Module

3.2.5 FEATURES OF CP2102 MODULE

- Stable and reliable chipset CP2102.
- USB specification 2.0 compliant with full-speed 12Mbps.
- Standard USB type A male and TTL 6pin connector.
- All handshaking and modem interface signals.
- Baud rates: 300 bps to 1 Mbps.

- Byte receives buffer; 640 bytes transmit buffer.

- Hardware or X-On/X-Off handshaking supported.

3.3 LCD DISPLAY (LIQUID CRYSTAL DISPLAY)

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment light- emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc.

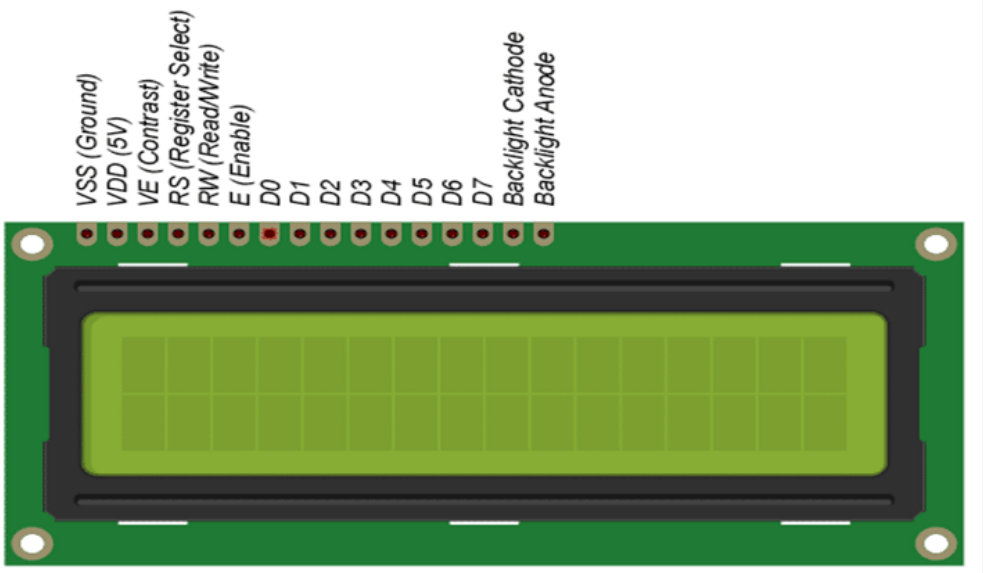


Fig 3.6: Liquid Crystal Display

A 16×2 LCD has two registers like data register and command register. The RS (register select) is mainly used to change from one register to another.

When the register set is ‘0’, then it is known as command register. Similarly, when the register set is ‘1’, then it is known as data register.

3.3.1 PIN CONFIGURATION OF LCD

Interface pin connections

PIN NO	Symbol	Function
1	VSS	GND
2	VDD	+5V
3	V0	Contrast adjustment
4	RS	H/L Register select signal
5	R/W	H/L Read/Write signal
6	E	H/L Enable signal
7	DB0	H/L Data bus line
8	DB1	H/L Data bus line
9	DB2	H/L Data bus line
10	DB3	H/L Data bus line
11	DB4	H/L Data bus line
12	DB5	H/L Data bus line
13	DB6	H/L Data bus line
14	DB7	H/L Data bus line
15	A	+4.2V for BKL — VCC Back Led
16	K	Power supply for BKL(0V) — GND Back Led

Table-3.1: Pin configuration of LCD

3.3.2 FEATURES

- The operating voltage of this LCD is 4.7V-5.3V
- It includes two rows where each row can produce 16-characters.
- The utilization of current is 1mA with no backlight.
- Every character can be built with a 5×8 pixel box.
- The alphanumeric LCDs alphabets & numbers.
- Is display can work on two modes like 4-bit & 8-bit.
- These are obtainable in Blue & Green Backlight.
- It displays a few custom generated characters.

These 16 x 2 LCD display modules are constant of 16 Columns and 2 Rows. The 1st row of this module has a total of 16 columns 0 to 15 and the position of the first row is 0. Also, the 2nd row has a total of 16 columns 0 to 15 and the position of the second row is position is 1. So, the total numbers of the column are 16 x 2 = 32. Its means 16 x 2 LCD module can display 32 characters at the same time.

The LCD consists of Data, Command and control registers. All the register helps to control the different kinds of functions on the LCD. The data and command registers take the input from digital pins D0-D7. Then controls pins help to differentiate between command/data registers.

3.3.3 SPECIFICATIONS OF LIQUID CRYSTAL DISPLAY

ITEMS	Unit	SPECIFICATIONS
Screen Diagonal	[mm]	546.86 (21.5")
Active Area	[mm]	476.064 (H) x 267.786 (V)
Pixels H x V	-	1920(x3) x 1080
Pixel Pitch	[um]	247.95 (per one triad) ×247.95
Pixel Arrangement	-	R.G.B. Vertical Stripe
Display Mode	-	AHVA, normally Black
White Luminance (Center)	[cd/m2]	1500 (Typ.)
Contrast Ratio	-	1000 (Typ.)
Response Time	[msec]	25 (Typ., G/G)
Power Consumption (LCD Module+Backlight unit)	[Watt]	38.1 (Typ.) LCD module : PDD (Typ.)= 2.3@White pattern,Fv=60Hz Backlight unit : P_{BLU} (Typ.) =35.8@Is= 47 mA
Weight	[Grams]	2.1 Kg
Outline Dimension	[mm]	501.1(H) × 292.2(V) ×16.12 (D) Typ
Electrical Interface	-	Dual channel LVDS , 8-bit RGB data input
Support Color	-	16.7M colors
Surface Treatment	-	Anti-Glare, 3H
Rotate Function		Unachievable
Display Orientation		Portrait/Landscape Enabled

Table-3.2: Specifications of LCD

3.3.4 APPLICATIONS OF LCD

- In most of the applications that's have only small values to show, uses the LCD.
- Most of the commercial meters use this module to represent the data output.
- In the toys and developing projects, it is still vastly in use.
- Black and white printers, it helps to show the printer settings and status.

3.4 KEYPAD

The 4x4 matrix keypad is a simple mechanism that resembles the numeric input on your computer keyboard, except that it has an additional '*,' '#' and 4 other auxiliary buttons that can be used for various functions in the application. The keypad is usually made of plastic materials and is relatively cheap compared to touchscreen displays.

A 4x4 matrix keypad can be implemented separately or within the physical product itself, such as a security access controller, where it is used for PIN identifications. Either way, the mechanism of the mechanical keypad remains the same when hardware and firmware designers are concerned.

If you've never designed with a 4x4 mechanical keypad, the best way to visualize the internal mechanism is a matrix of push-button switches. A 4x4 keypad has a total of 8 connections, where 4 of them are connected to the column and the remaining rows of the matrix of switches.

When an individual button is pressed, a connection is established between one of the rows and columns. The microcontroller then deciphers the physical button based on the index of the row and column that is activated.

Fig 3.7: 4x4 Keypad

3.4.1 SPECIFICATIONS OF 4x4 KEYPAD

A 4x4 keypad is a common input device with 16 keys arranged in a 4x4 matrix. Each key typically corresponds to a specific character or function. Here are the basic specifications of a typical 4x4 keypad:

- **Key Matrix:** The keypad consists of a 4x4 grid, which means it has four rows and four columns, resulting in a total of 16 keys.

- **Key Layout:** The keys are usually labeled with numbers (0-9) and alphabets (A, B, C, D, *, #, etc.) for alphanumeric input, along with special function keys.

- **Connector:** The keypad typically has a connector, such as a 7-pin connector, for interfacing with a microcontroller or other input devices.

- **Interface:** The keypad is designed to work with digital electronic devices and is usually connected to a microcontroller or other digital system. It may use protocols like GPIO (General Purpose Input/Output) for communication.

- **Switch Type:** The keys on a 4x4 keypad usually use dome switches or mechanical switches to register key presses. Dome switches are often more common in membrane keypads, while mechanical switches are found in more robust keypads.

- **Key Press Detection:** The keypad detects key presses by establishing a connection between a specific row and column when a key is pressed. The microcontroller can then determine which key was pressed based on this connection.

- **Electrical Characteristics:** The keypad typically operates at a low voltage and draws minimal current when not in use. The electrical characteristics will vary depending on the specific keypad model and manufacturer.

- **Mounting:** Keypads can be mounted on a panel or enclosure using screws or clips, depending on the design.

- **Materials:** Keypads are typically constructed from materials like plastic and have a protective overlay to resist wear and tear.

- **Size and Form Factor:** The size and form factor of a 4x4 keypad can vary, but it's usually compact and designed to be integrated into a device or control panel.

4x4 keypads are commonly used in various applications, including security systems, access control systems, industrial equipment, consumer electronics, and more, to provide a convenient input method for users. The specific features and specifications of a 4x4 keypad may vary from one model to another, so it's essential to refer to the manufacturer's documentation for detailed information on a particular keypad.

3.4.2 APPLICATIONS OF KEYPAD

- Access Control Systems

- Security Systems

- Data Entry Systems

- Industrial Control Systems

- Robotics

- Electronic projects.

3.5 POWER SUPPLY

3.5.1 INTRODUCTION

There are many types of power supply. Most are designed to convert high voltage AC mains electricity to a suitable low voltage supply for electronic circuits and other devices. A power supply can by broken down into a series of blocks, each of which performs a particular function. For example the **Arduino Adapter** is a12V regulated supply can be shown as below

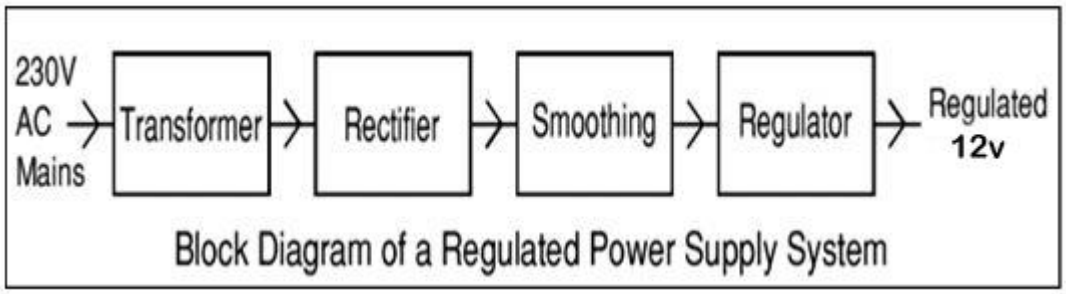


Fig 3.8: Block Diagram of a Regulated Power Supply System

Similarly, 15v regulated supply can also be produced by suitable selection of the individual elements. Each of the blocks is described in detail below and the power supplies made from these blocks are described below with a circuit diagram and a graph of their output:

3.5.2 TRANSFORMER

A transformer steps down high voltage AC mains to low voltage AC. Here we are using a center-tap transformer whose output will be sinusoidal with 12 volts peak to peak value.

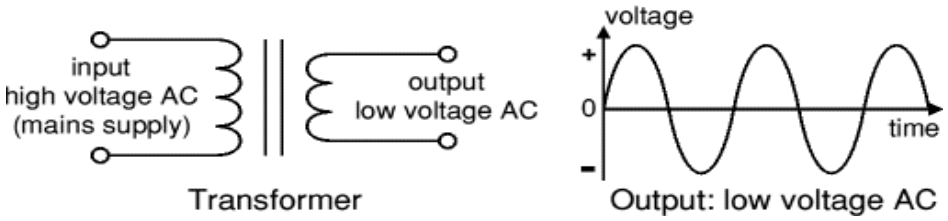


Fig 3.9: Output Waveform of transformer

The low voltage AC output is suitable for lamps, heaters and special AC motors. It is not suitable for electronic circuits unless they include a rectifier and a smoothing capacitor. The transformer output is given to the rectifier circuit.

3.5.3 RECTIFIER

A rectifier converts AC to DC, but the DC output is varying. There are several types of rectifiers; here we use a bridge rectifier.

The Bridge rectifier is a circuit, which converts an ac voltage to dc voltage using both half cycles of the input ac voltage. The Bridge rectifier circuit is shown in the figure. The circuit has four diodes connected to form a bridge. The ac input voltage is applied to the diagonally opposite ends of the bridge. The load resistance is connected between the other two ends of the bridge.

For the positive half cycle of the input ac voltage, diodes D1 and D3 conduct, whereas diodes D2 and D4 remain in the OFF state. The conducting diodes will be in series with the load resistance R_L and hence the load current flows through R_L.

For the negative half cycle of the input ac voltage, diodes D2 and D4 conduct whereas, D1 and D3 remain OFF. The conducting diodes D2 and D4 will be in series with the load resistance R_L and hence the current flows through R_L in the same direction as in the previous half cycle. Thus a bi-directional wave is converted into unidirectional.

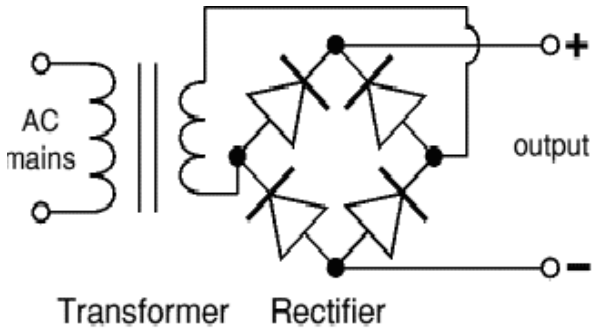


Fig 3.10: Rectifier circuit

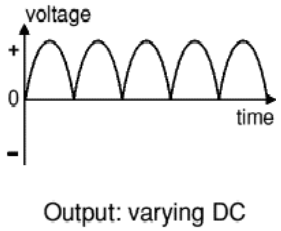


Fig 3.11: Output of the Rectifier

The varying DC output is suitable for lamps, heaters and standard motors. It is not suitable for lamps, heaters and standard motors. It is not suitable for electronic circuits unless they include a smoothing capacitor.

3.5.4 SMOOTHING

The smoothing block smoothes the DC from varying greatly to a small ripple and the ripple voltage is defined as the deviation of the load voltage from its DC value. Smoothing is also named as filtering.

Filtering is frequently effected by shunting the load with a capacitor. The action of this system depends on the fact that the capacitor stores energy during the conduction period and delivers this energy to the loads during the no conducting period. In this way, the time during which the current passes through the load is prolonging Ted, and the ripple is considerably decreased. The action of the capacitor is shown with the help of waveform.

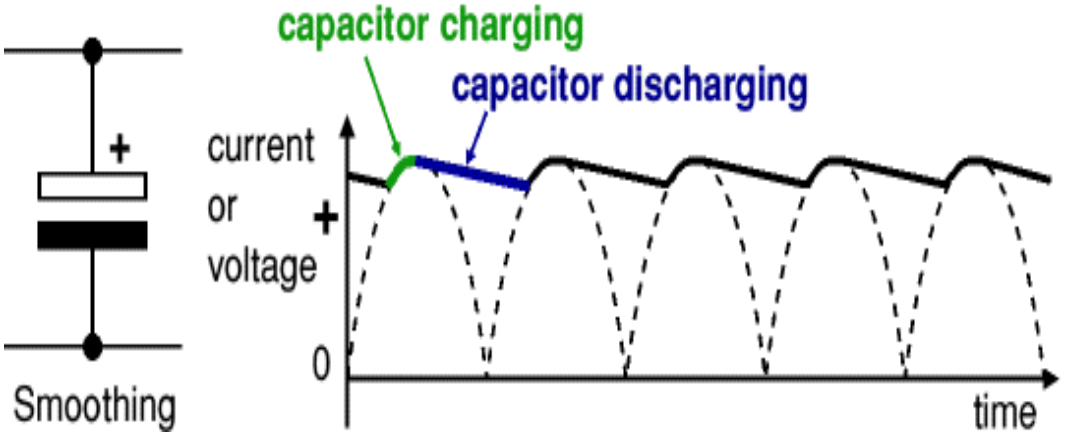


Fig 3.10: Smoothing action of capacitor

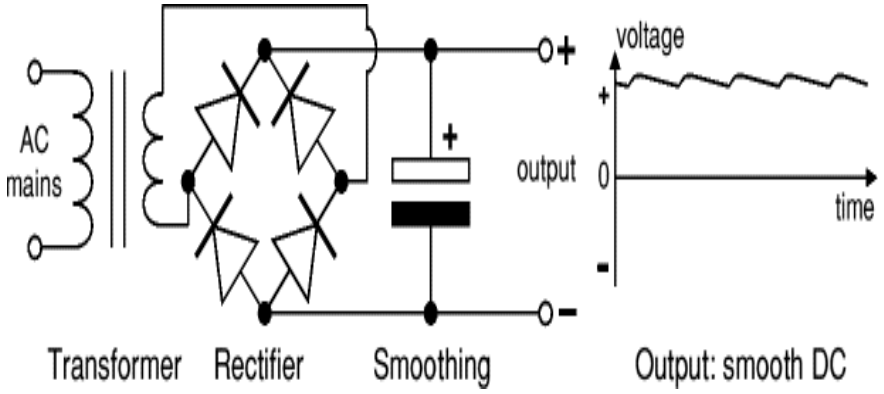


Fig 3.12: Waveform of the rectified output smoothing

3.5.5 REGULATOR

Regulator eliminates ripple by setting DC output to a fixed voltage. Voltage regulator ICs are available with fixed (typically 5V, 12V and 15V) or variable output voltages. Negative voltage regulators are also available. Many of the fixed voltage regulator ICs has 3 leads (input, output and high impedance). They include a hole for attaching a heat sink if necessary. Zener diode is an example of fixed regulator which is shown here.

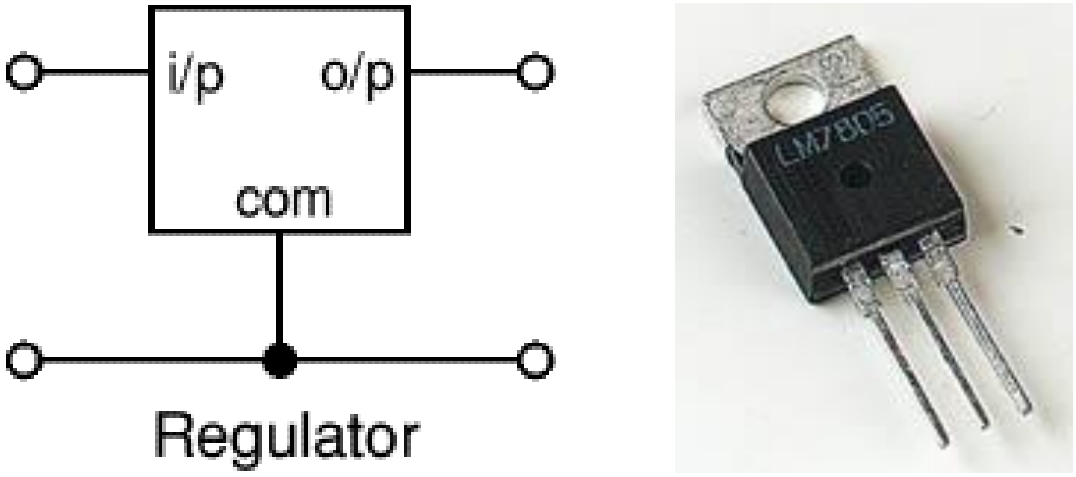


Fig 3.14: Regulator

Transformer + Rectifier + Smoothing + Regulator:

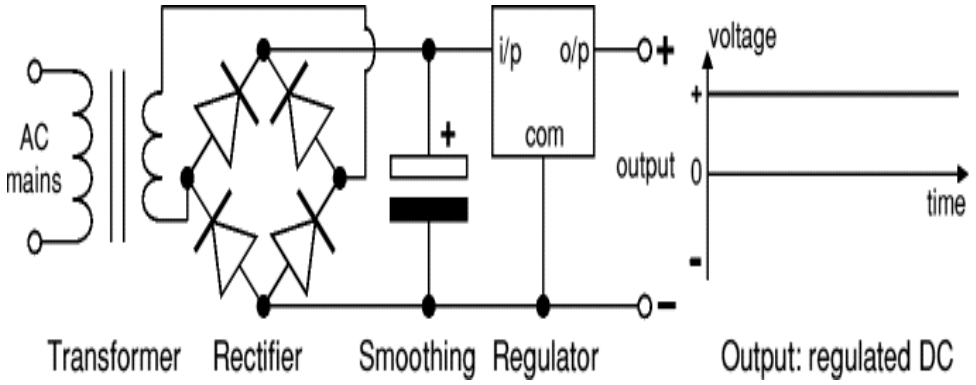


Fig 3.15: Full wave bridge rectifier

CHAPTER-4
SOFTWARE AND CODE

4.1 ARDUINO IDE:

Arduino is a prototype platform (open-source) based on an easy-to-use hardware and software. It consists of a circuit board, which can be programmed (referred to as a microcontroller) and a ready-made software called Arduino IDE (Integrated Development Environment), which is used to write and upload the computer code to the physical board.

4.1.1 About the Arduino IDE Tools

After learning about the main parts of the Arduino UNO board, we are ready to learn how to set up the Arduino IDE. Once we learn this, we will be ready to upload our program on the Arduino board. In this section, we will learn in easy steps, how to set up the Arduino IDE on our computer and prepare the board to receive the program via USB cable.

Step 1 – First you must have your Arduino board (you can choose your favorite board) and a USB cable. In case you use Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, you will need a standard USB cable (A plug to B plug), the kind you would connect to a USB printer as shown in the following image.

Fig 7.1: A to B standard USB cable

In case you use Arduino Nano, you will need an A to Mini-B cable instead as shown in the following image.

Step 2 – Download Arduino IDE Software.

You can get different versions of Arduino IDE from the Download page on the Arduino Official website. You must select your software, which is compatible with your operating system (Windows, IOS, or Linux). After your file download is complete, unzip the file.

Step 3 – Power up your board.

The Arduino Uno, Mega, Duemilanove and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks. Check that it is on the two pins closest to the USB port.

Connect the Arduino board to your computer using the USB cable. The green power LED (33abelled PWR) should glow.

Step 4 – Launch Arduino IDE.

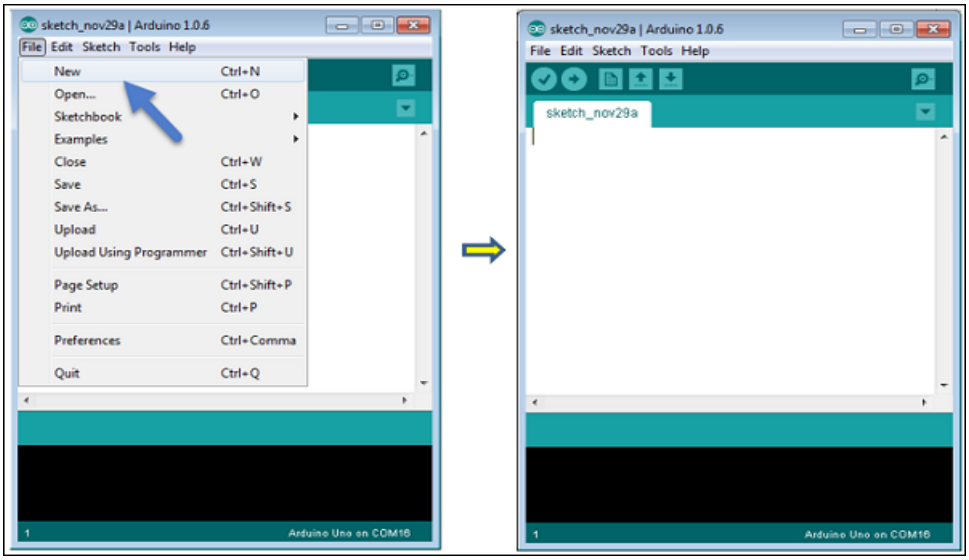
After your Arduino IDE software is downloaded, you need to unzip the folder. Inside the folder, you can find the application icon with an infinity label (application.exe). Double-click the icon to start the IDE.

Step 5 – Open your first project.

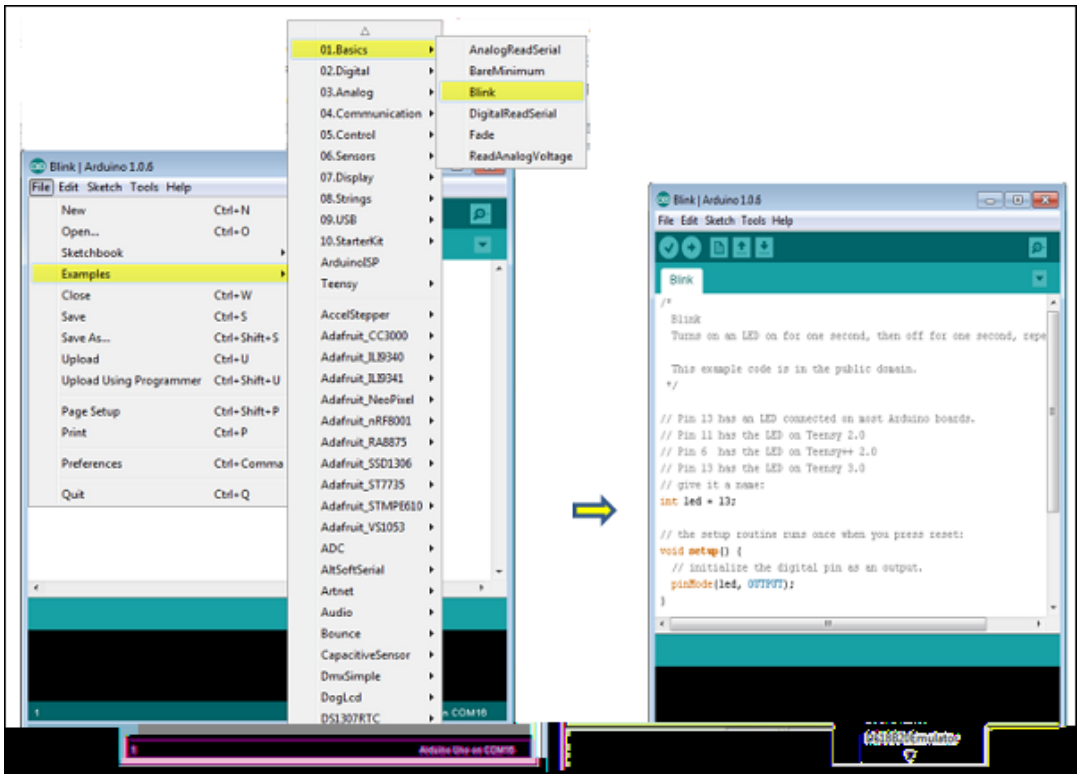
Once the software starts, you have two options –

- Create a new project.
- Open an existing project example.

To create a new project, select File → **New**.



To open an existing project example, select File → Example → Basics → Blink.



Here, we are selecting just one of the examples with the name **Blink**. It turns the LED on and off with some time delay. You can select any other example from the list.

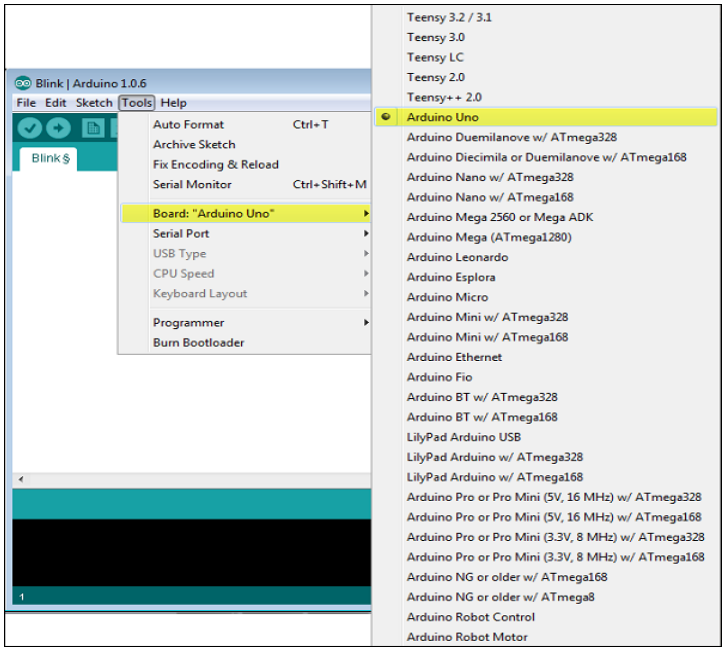
Step 6 – Select your Arduino board.

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer.

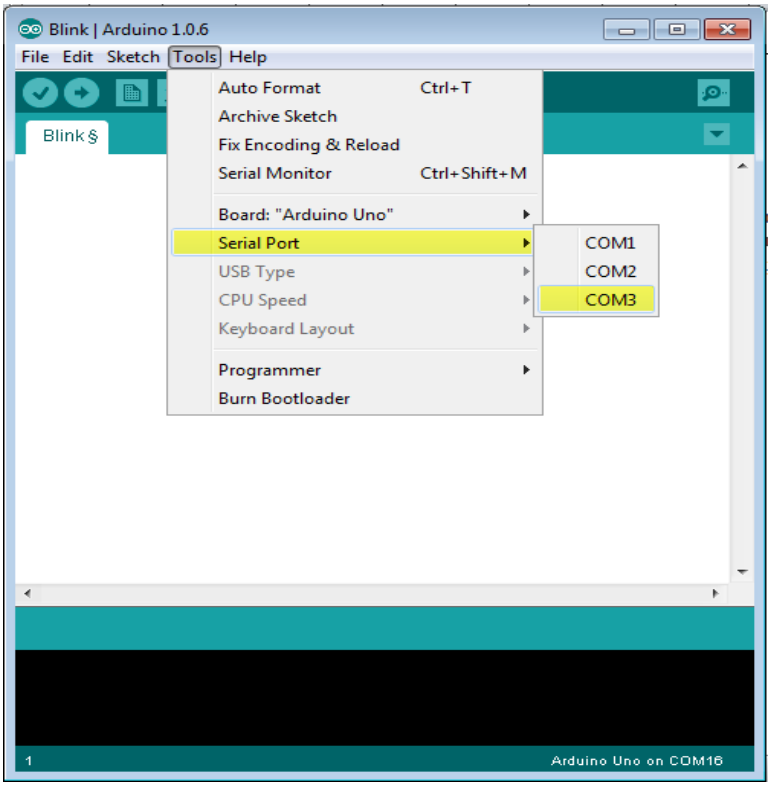
Go to Tools → Board and select your board.

Here, we have selected Arduino Uno board according to our tutorial, but you must select the name matching the board that you are using.

Step 7 – Select your serial port.



Select the serial device of the Arduino board. Go to **Tools → Serial Port** menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.



Step 8 – Upload the program to your board.

Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.

A – Used to check if there is any compilation error.

B – Used to upload a program to the Arduino board.

C – Shortcut used to create a new sketch.

D – Used to directly open one of the example sketch.

E – Used to save your sketch.

F – Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.

Note – If you have an Arduino Mini, NG, or other board, you need to press the reset button physically on the board, immediately before clicking the upload button on the Arduino Software.

4.2 CODE FOR THE SYSTEM

```
#include <LiquidCrystal.h>
const int rs = 13, en = 12, d4 = 11, d5 = 10, d6 = 9, d7 = 8;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

#include <Keypad.h>

const int ROW_NUM = 4; //four rows
const int COLUMN_NUM = 4; //three columns
int m1=A4;
int m2=A5;
int buzzer=A0;
int bill=0;
int amount=0;
char del='+';
char keys[ROW_NUM][COLUMN_NUM] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};
```

```
byte pin_rows[ROW_NUM] = {7,6,5,4}; //connect to the row pinouts of the keypad
byte pin_column[COLUMN_NUM] = {2, 3, A4,A5}; //connect to the column pinouts of the keypad

Keypad keypad = Keypad( makeKeymap(keys), pin_rows, pin_column, ROW_NUM, COLUMN_NUM );

const String password1 = "1"; // change your password here
String cc="*********";
int i=0;

String input_password;

void setup()
{

pinMode(m1,OUTPUT);pinMode(m2,OUTPUT);pinMode(buzzer,OUTPUT);
  digitalWrite(m1,LOW);digitalWrite(m2,LOW);digitalWrite(buzzer,HIGH);
  lcd.begin(16,2);
  Serial.begin(9600);
  input_password.reserve(32); // maximum input characters is 33, change if needed

  lcd.clear();lcd.print("MENU ORDERING");
  lcd.setCursor(0,1);lcd.print("SYSTEM USING ARDUINO");delay(3000);
  lcd.clear();lcd.print("ORDER 'C'");delay(1000);
}

void loop(){
  back:
```

```
  char key = keypad.getKey();


  if (key){
   lcd.clear();lcd.print(key);
   if(key==’A’)  {lcd.clear();lcd.print(cc);delay(1000);  input_password  =
“”;lcd.setCursor(0,1);lcd.print(“ TOTAL BILL:”);lcd.print(bill);delay(3000);}
if(key==’1’)  {password1=input_password;delay(1000);lcd.clear();lcd.print(“
DISH:1 Rs 10/-“);delay(1000);amount=10;del=’1’;}
if(key==’2’)  {password1=input_password;delay(1000);lcd.clear();lcd.print(“
DISH:2 Rs 20/-“);delay(1000);amount=20;del=’2’;}
if(key==’3’)  {password1=input_password;delay(1000);lcd.clear();lcd.print(“
DISH:3 Rs 30/-“);delay(1000);amount=30;del=’3’;}
if(key==’4’)  {password1=input_password;delay(1000);lcd.clear();lcd.print(“
DISH:4 Rs 40/-“);delay(1000);amount=40;del=’4’;}
if(key==’5’)  {password1=input_password;delay(1000);lcd.clear();lcd.print(“
DISH:5 Rs 50/-“);delay(1000);amount=50;del=’5’;}
if(key==’6’)  {password1=input_password;delay(1000);lcd.clear();lcd.print(“
DISH:6 Rs 60/-“);delay(1000);amount=60;del=’6’;}
if(key==’7’)  {password1=input_password;delay(1000);lcd.clear();lcd.print(“
DISH:7 Rs 70/-“);delay(1000);amount=70;del=’7’;}
if(key==’8’)  {password1=input_password;delay(1000);lcd.clear();lcd.print(“
DISH:8 Rs 80/-“);delay(1000);amount=80;del=’8’;}
if(key==’9’)  {password1=input_password;delay(1000);lcd.clear();lcd.print(“
DISH:9 Rs 90/-“);delay(1000);amount=90;del=’9’;}
if(key==’0’)
{lcd.clear();lcd.print(“COMPLETED...”);lcd.setCursor(0,1);lcd.print(“BILL
AMOUNT:”);lcd.print(bill);delay(1000);
Serial.print(“TABLE  NO:1  COMPLETED  TOTAL  BILL
AMOUNT:”);Serial.println(bill);delay(1000);
Serial.println(“** .....  **”);
}
   if(key == ‘*’)
```

```
  {
   input_password = “”; lcd.clear();lcd.print(“CLEAR
SCREEN”);delay(2000);lcd.clear();
  }
  else if(key == ‘C’)
  {

   if(input_password==”1”)
   {
    password1=input_password;delay(1000);
   lcd.clear();lcd.print(“ DISH:1 Rs 10/-“);bill=bill+10;delay(1000);

lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);lcd.print(“
ORDERED”);
    Serial.println(“ TABLE NO:1 DISH:1 “);delay(1000);cc[i]=’1’;i++;
   }
   else if(input_password==”2”)
   { password1=input_password;delay(1000);
   lcd.clear();lcd.print(“ DISH:2 Rs 20/-“);bill=bill+20;delay(1000);
   lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
   Serial.println(“ TABLE NO:1 DISH:2 “);delay(1000);lcd.print(“
ORDERED”);cc[i]=’2’;i++;
   }
   else if(input_password==”3”)
   { password1=input_password;delay(1000);
   lcd.clear();lcd.print(“ DISH:3 Rs 30/-“);bill=bill+30;delay(1000);
   lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
   Serial.println(“ TABLE NO:1 DISH:3 “);delay(1000); lcd.print(“
ORDERED”);cc[i]=’3’;i++;
   }
   else if(input_password==”4”)
   { password1=input_password;delay(1000);
```

```
      lcd.clear();lcd.print(" DISH:4 Rs 40/-");bill=bill+40;delay(1000);
      lcd.setCursor(0,1);lcd.print("BILL:");lcd.print(bill);delay(1000);
      Serial.println(" TABLE NO:1 DISH:4 ");delay(1000); lcd.print(" ORDERED");cc[i]='4';i++;
    }

   else if(input_password=="5")
    { password1=input_password;delay(1000);
     lcd.clear();lcd.print(" DISH:5 Rs 50/-");bill=bill+50;delay(1000);
     lcd.setCursor(0,1);lcd.print("BILL:");lcd.print(bill);delay(1000);
     Serial.println(" TABLE NO:1 DISH:5 ");delay(1000); lcd.print(" ORDERED");cc[i]='5';i++;
    }
    else if(input_password=="6")
    { password1=input_password;delay(1000);
     lcd.clear();lcd.print(" DISH:6 Rs 60/-");bill=bill+60;delay(1000);
     lcd.setCursor(0,1);lcd.print("BILL:");lcd.print(bill);delay(1000);
     Serial.println(" TABLE NO:1 DISH:6 ");delay(1000); lcd.print(" ORDERED");cc[i]='6';i++;
    }
    else if(input_password=="7")
    { password1=input_password;delay(1000);
     lcd.clear();lcd.print(" DISH:7 Rs 70/-");bill=bill+70;delay(1000);
     lcd.setCursor(0,1);lcd.print("BILL:");lcd.print(bill);delay(1000);
     Serial.println(" TABLE NO:1 DISH:7 ");delay(1000); lcd.print(" ORDERED");cc[i]='7';i++;
    }

else if(input_password=="8")
    { password1=input_password;delay(1000);
     lcd.clear();lcd.print(" DISH:8 Rs 80/-");bill=bill+80;delay(1000);
     lcd.setCursor(0,1);lcd.print("BILL:");lcd.print(bill);delay(1000);
```

```
      Serial.println(“ TABLE NO:1 DISH:8 “);delay(1000); lcd.print(“ ORDERED”);cc[i]=’8’;i++;
     }
     else if(input_password==”9”)
     { password1=input_password;delay(1000);
      lcd.clear();lcd.print(“ DISH:9 Rs 90/-“);bill=bill+90;delay(1000);
      lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
      Serial.println(“ TABLE NO:1 DISH:9 “);delay(1000); lcd.print(“ ORDERED”);cc[i]=’9’;i++;
     }

    else
    {
     lcd.clear();lcd.print(“INVALID OPTION...”);delay(1000);
      lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
    }
     input_password = “”; // clear input password
   }
else if(key == ‘D’)
   {

if(cc[0]!=del && cc[1]!=del && cc[2]!=del && cc[3]!=del && cc[4]!=del && cc[5]!=del && cc[6]!=del && cc[7]!=del && cc[8]!=del )
{
 lcd.clear();lcd.print(“NOT YER OREDER”);lcd.print(i);delay(1000);goto back;
}

for(int k=0;k<=8;k++)
{
  if(cc[k]==del)
  {
```

```
    lcd.clear();lcd.print("ITEM DELETED");delay(1000);
    cc[k]='*';i--;delay(1000);
  }

}

     if(bill<amount)
     {
      lcd.clear();lcd.print("TRY AGAIN");delay(1000); goto back;
     }
    if(bill>0)
    {
     if(input_password==”1”)
     {
       password1=input_password;delay(1000);
      lcd.clear();lcd.print(" DISH:1 Rs 10/-");bill=bill-10;delay(1000);
      lcd.setCursor(0,1);lcd.print("BILL:");lcd.print(bill);delay(1000);
      Serial.println(" TABLE NO:1 DISH:1 cancle  ");delay(1000);lcd.print("
CANCLE");
     }
     else if(input_password==”2”)
     { password1=input_password;delay(1000);
      lcd.clear();lcd.print(" DISH:2 Rs 20/-");bill=bill-20;delay(1000);
      lcd.setCursor(0,1);lcd.print("BILL:");lcd.print(bill);delay(1000);
      Serial.println(" TABLE NO:1 DISH:2 cancle ");delay(1000);lcd.print("
CANCLE");
     }
     else if(input_password==”3”)
     { password1=input_password;delay(1000);
      lcd.clear();lcd.print(" DISH:3 Rs 30/-");bill=bill-30;delay(1000);
      lcd.setCursor(0,1);lcd.print("BILL:");lcd.print(bill);delay(1000);
```

```
      Serial.println(“ TABLE NO:1 DISH:3 cancle”);delay(1000); lcd.print(“
CANCLE”);
     }
     else if(input_password==”4”)
     { password1=input_password;delay(1000);
      lcd.clear();lcd.print(“ DISH:4 Rs 40/-“);bill=bill-40;delay(1000);
      lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
      Serial.println(“ TABLE NO:1 DISH:4 cancle “);delay(1000); lcd.print(“
CANCLE”);
     }

    else if(input_password==”5”)
     { password1=input_password;delay(1000);
      lcd.clear();lcd.print(“ DISH:5 Rs 50/-“);bill=bill-50;delay(1000);
      lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
      Serial.println(“ TABLE NO:1 DISH:5 cancle”);delay(1000); lcd.print(“
CANCLE”);
     }
     else if(input_password==”6”)
     { password1=input_password;delay(1000);
      lcd.clear();lcd.print(“ DISH:6 Rs 60/-“);bill=bill-60;delay(1000);
      lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
      Serial.println(“ TABLE NO:1 DISH:6 cancle”);delay(1000); lcd.print(“
CANCLE”);
     }
     else if(input_password==”7”)
     { password1=input_password;delay(1000);
      lcd.clear();lcd.print(“ DISH:7 Rs 70/-“);bill=bill-70;delay(1000);
      lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
      Serial.println(“ TABLE NO:1 DISH:7  cancle”);delay(1000); lcd.print(“
CANCLE”);
     }
```

```
else if(input_password==”8”)
      { password1=input_password;delay(1000);
       lcd.clear();lcd.print(“ DISH:8 Rs 80/-“);bill=bill-80;delay(1000);
       lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
       Serial.println(“ TABLE NO:1 DISH:8 cancle”);delay(1000); lcd.print(“
CANCLE”);
      }
      else if(input_password==”9”)
      { password1=input_password;delay(1000);
       lcd.clear();lcd.print(“ DISH:9 Rs 90/-“);bill=bill-90;delay(1000);
       lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
       Serial.println(“ TABLE NO:1 DISH:9 cancle”);delay(1000); lcd.print(“
CANCLE”);
      }

     else
     {
      lcd.clear();lcd.print(“INVALID OPTION...”);delay(1000);
       lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
     }
      input_password = “”; // clear input password
    }
    }
else if(key == ‘B’)
    {
     if(bill>0)
     {

      if(input_password==”1”)
      {
        password1=input_password;delay(1000);
       lcd.clear();lcd.print(“ DISH:1 Rs 10/-“);delay(1000);
```

```
 lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
 Serial.println(“ TABLE NO:1 DISH:1 PLACED  “);delay(1000);
}
else if(input_password==”2”)
{ password1=input_password;delay(1000);
 lcd.clear();lcd.print(“ DISH:2 Rs 20/-“);delay(1000);
 lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
 Serial.println(“ TABLE NO:1 DISH:2 PLACED “);delay(1000);
}
else if(input_password==”3”)
{ password1=input_password;delay(1000);
 lcd.clear();lcd.print(“ DISH:3 Rs 30/-“);delay(1000);
 lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
 Serial.println(“ TABLE NO:1 DISH:3 PLACED”);delay(1000);
}
else if(input_password==”4”)
{ password1=input_password;delay(1000);
 lcd.clear();lcd.print(“ DISH:4 Rs 40/-“);delay(1000);
 lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
 Serial.println(“ TABLE NO:1 DISH:4 PLACED”);delay(1000);
}

else if(input_password==”5”)
 { password1=input_password;delay(1000);
 lcd.clear();lcd.print(“ DISH:5 Rs 50/-“);delay(1000);
 lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
 Serial.println(“ TABLE NO:1 DISH:5 PLACED”);delay(1000);
}
else if(input_password==”6”)
{ password1=input_password;delay(1000);
 lcd.clear();lcd.print(“ DISH:6 Rs 60/-“);delay(1000);
 lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
```

```
 Serial.println(“ TABLE NO:1 DISH:6 PLACED “);delay(1000);
}
else if(input_password==”7”)
{ password1=input_password;delay(1000);
 lcd.clear();lcd.print(“ DISH:7 Rs 70/-“);delay(1000);
 lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
 Serial.println(“ TABLE NO:1 DISH:7 PLACED”);delay(1000);
}

else if(input_password==”8”)
{ password1=input_password;delay(1000);
 lcd.clear();lcd.print(“ DISH:8 Rs 80/-“);delay(1000);
 lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
 Serial.println(“ TABLE NO:1 DISH:8 PLACED”);delay(1000);
}
else if(input_password==”9”)
{ password1=input_password;delay(1000);
 lcd.clear();lcd.print(“ DISH:9 Rs 90/-“);delay(1000);
 lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
 Serial.println(“ TABLE NO:1 DISH:9 PLACED”);delay(1000);
}

else
{
 lcd.clear();lcd.print(“INVALID OPTION...”);delay(1000);
  lcd.setCursor(0,1);lcd.print(“BILL:”);lcd.print(bill);delay(1000);
}
 input_password = “”; // clear input password
}
}
else
{
```

```
            input_password += key; // append new character to input password string
          }
        }
      }
```

CHAPTER-5

WORKING

5.1 BLOCK DIAGRAM

5.1.1 TRANSMITTER SECTION

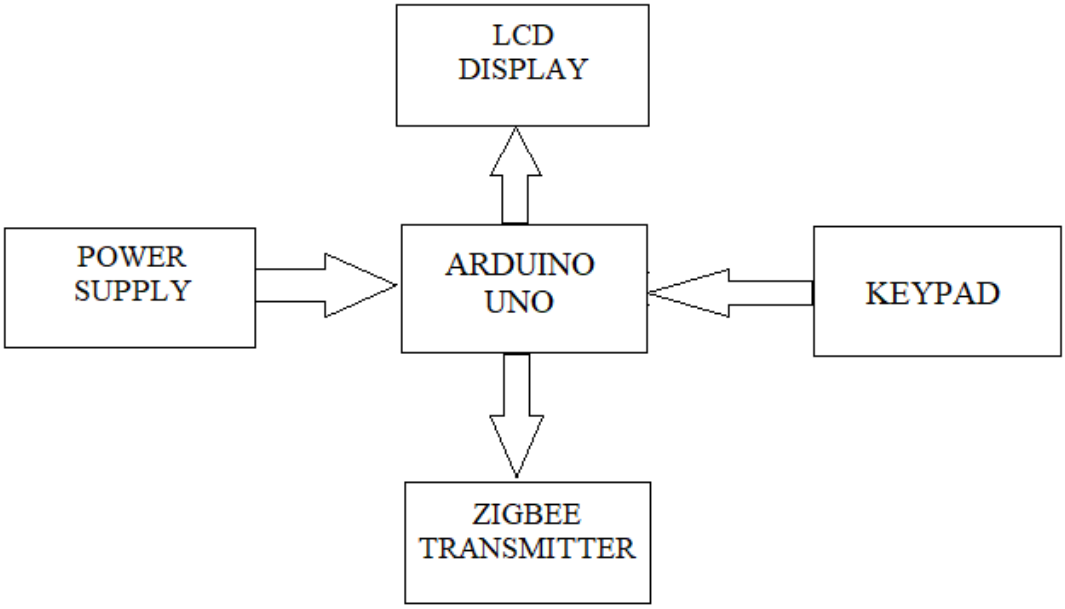


Fig 5.1: Transmitter Section

5.1.2 RECEIVER SECTION

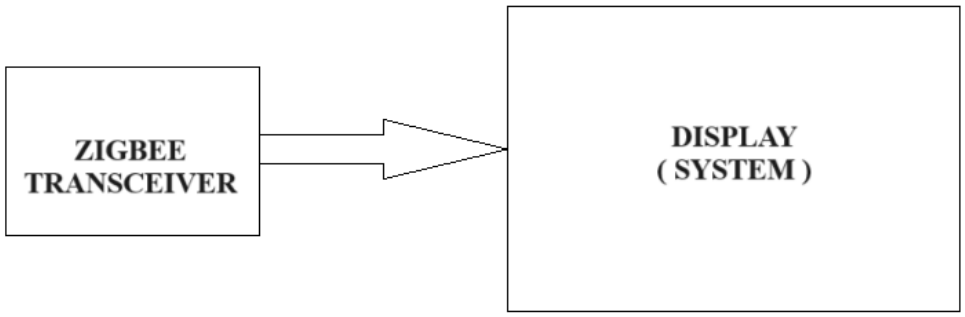


Fig 5.2: Receiver Section

Automation systems are increase in day to day life. It is the essential part in the field of electronics. It deals with transfer of data from one place to another place. Communication has major role in the successful data transfer and to get the acknowledgement from receiver. There are two mode of transmission; wired and wireless transmission. In wired transmission, data is transferred through a physical medium or a link whereas no physical link is used in wireless transmission. Both mediums have its own characteristics and advantages. Many times when we visit any restaurants due to overcrowded when order is being placed it takes more time to process and increases the man power to overcome such disadvantages a system is being implemented called as automatic hotel order processing system where users table consists of a keypad and LCD display on pressing the relevant code of the food item user can send that to the kitchen where waiter can take the order and send the acknowledgement to the customer. Then waiter serve the menu to the customer on time.

Hotel is one where technology and advancements in technology have not been utilized to the fullest potential. Traditional method that is commonly been used in hotels is by taking the customer's orders and writing it down on a piece of paper. Many solutions have been proposed for solving this issue. This project is again one attempt in the same direction. In this paper we discuss the automation for food ordering system. This system makes use of zigbee as a communication device and LCD display module compatible with Aurdino as hardware.

The basic principle of working of system is based on use of a handheld device placed on each table which is used to make an order at the hotel. The system uses a LCD display module which is placed on each customer's table for them to make order. Order is made by selecting the items displayed on LCD . The order will be sent from the customer section using zigbee communication and automatically will be displayed on a screen at the kitchen. The bill will be displayed at customer's table as well as at kitchen. The project will reduce the

time spent on making the orders and paying the bills, whereby the cost and man power also can be reduced.

5.2 ZIGBEE TECHNOLOGY

Zig-bee is a wireless device which communicates between transmitter and receiver. Zig-Bee is a standard that defines a set of communication protocols for low data-rate short-range wireless networking. ZigBee based wireless device operates in 868 MHz, 915 MHz, and 2.4 GHz frequency bands. The maximum data rate is 250 Kbps.

The ZigBee network is defined by the ZigBee Alliance and based on the IEEE 802.15.4 standard, which is target data RF embedded applications that require a low data rate, long battery life and secure networking. It is intended to operate in the 2.4GHz unlicensed ISM band [1 -2]. There is no large numbers of data which need to convey between the wireless ordering terminal build-in ZigBee module and the center node, and because of having no high requirement of data rate, so ZigBee is well suited for wireless ordering system. Each ZigBee modules includes an IEEE 802.15.4-compliant radio, an 8051 microcontroller, programmable I/O, flexible antenna and range solutions, Transmit range is up to 300m, which can meet the demand of wireless ordering system completely.

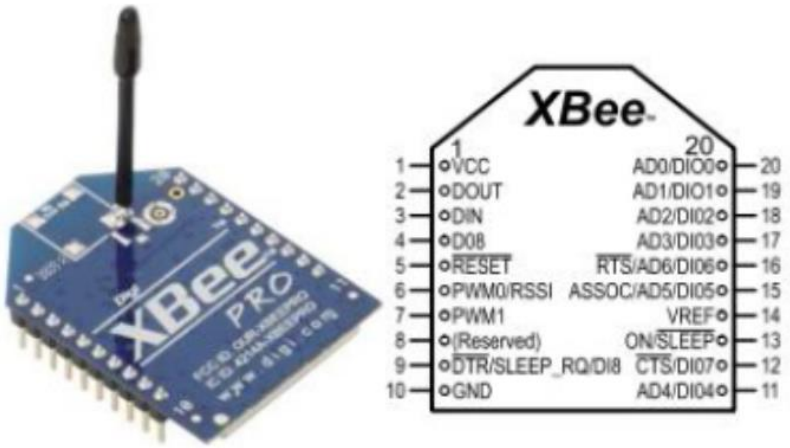


Fig 5.3: Zigbee Module

5.3 INTERFACING ZIGBEE WITH ARDUINO

Zigbee is a wireless communication protocol targeted for battery powered devices (it has both low power and low cost). It generally operates in the 2.4GHz range (although there are geographic variations), and supports data ranges from 20 to 250 kbits/s.

The transmission distance though, is small compared to the likes of LoRa. It is 10 to 100 m, whereas LoRa can transmit over a few kilo meters. Another thing to note is that Zigbee communication doesn't work very well if there is no line of sight between transmitter and receiver.

Even minor obstacles have been observed to significantly degrade the communication. Keep these limitations in mind when using Zigbee. You may want to look out for other options if your application can't meet these constraints. In order to make Zigbee work with Arduino, we will use the Xbee module.

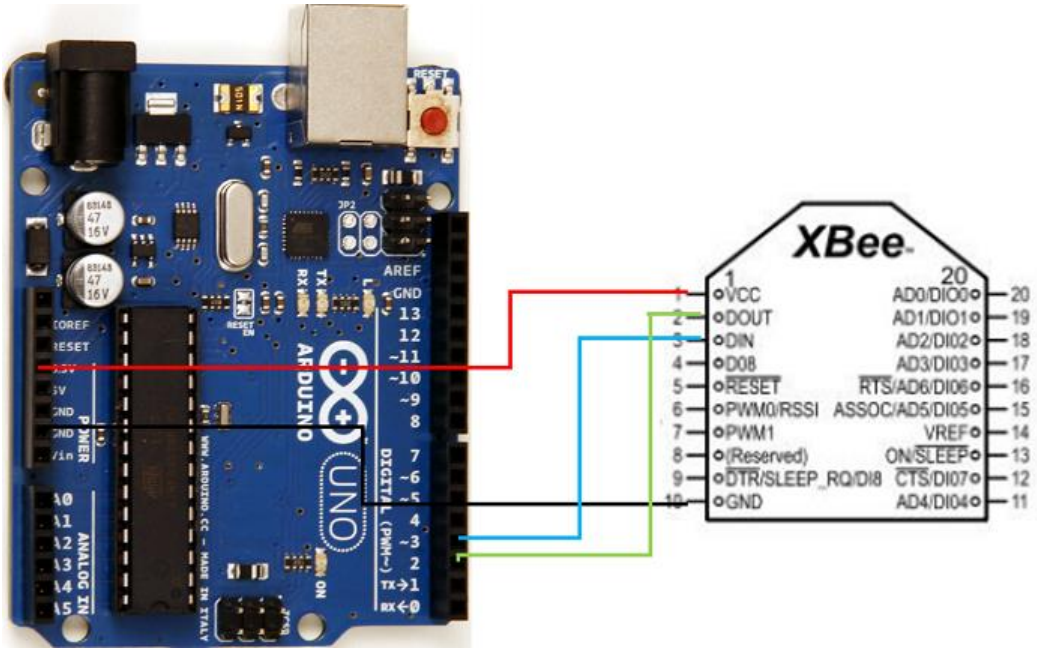


Fig 5.4: Interfacing Zigbee with Arduino

5.4 CIRCUIT DIAGRAM

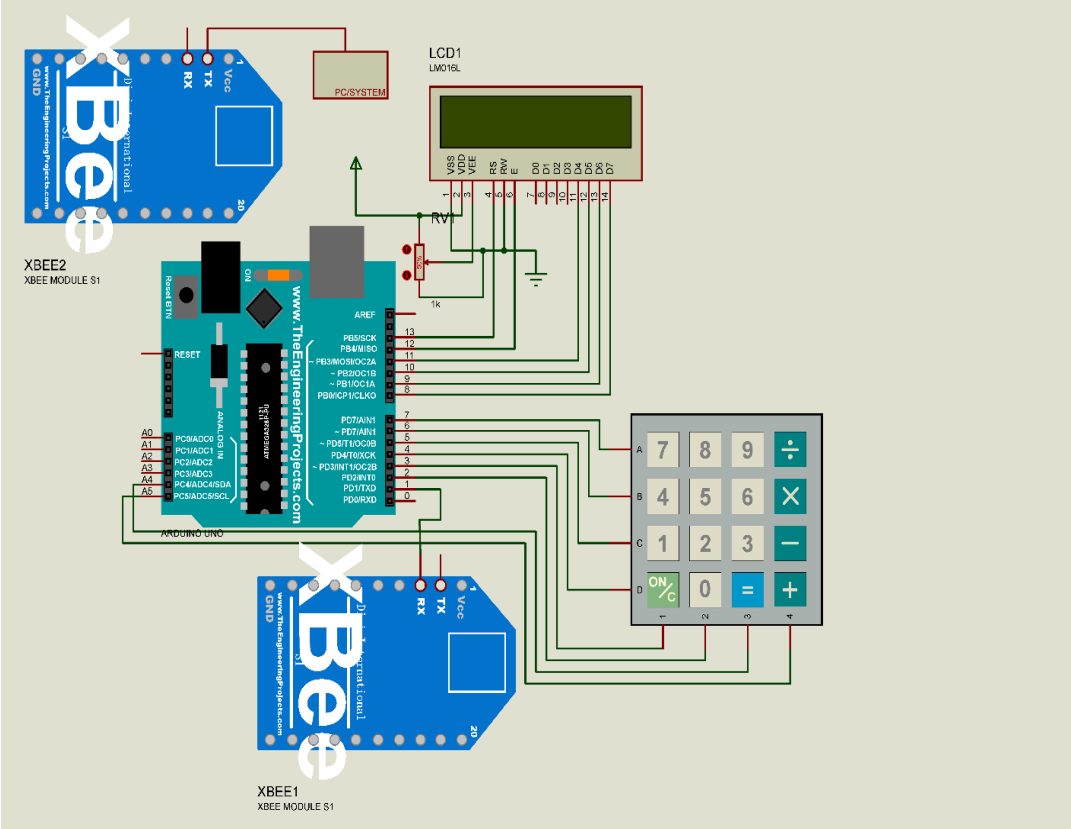


Fig 5.5: Circuit Diagram

5.5 WORKING PRINCIPLE

Customer will observe the menu list of hotel on LCD display. Customer will choose menu of his choice by selecting the respective menu. While doing this, buzzer will ON and LED start blinking which indicates that order has been successfully placed. This order will received by the waiter which will displayed on the LCD placed in kitchen. After receiving order waiter will send acknowledgment to the customer. After getting acknowledgement, customer knows about the confirmation order. If respective menu is not present, then waiter press the Reject button which gives the acknowledgement to the customer about the unavailability of menu or item and Re-order. Waiter serves the menu to the customer. Customer can add additional menu if he want. If customer don't want to take any menu he can press "Exit" button and then massage will come

"Are you sure to pay bill?" When customer press "YES" bill will generated on table. The basic principle of working of system is based on use of a handheld device placed on each table which is used to make an order at the hotel. The system uses a LCD display module which is placed on each customer's table for them to make order. Order is made by selecting the items displayed on LCD.

The order will be sent from the customer section using zig-bee communication and automatically will be displayed on a screen at the kitchen. The bill will be displayed at customer's table as well as at kitchen. The project will reduce the time spent on making the orders and paying the bills, whereby the cost and man power also can be reduced. The system is start from the customer's table. When the customer is sit on the table system is initialize and display the name of system. The various menus are display on LCD display; customer has to select the quantity of particular food item by pressing noted point on touchpad. If customer wants to increase the quantity then again press the touchpad. After selecting quantities of the entire food items bill is display on the screen. This order is now send to the kitchen side using zig-bee. At kitchen after receiving order reply is given to the customer using keypad. Customer is received reply of unavailable food item. Then customer again have to reply back to confirm the order. Then food is served according to the order. The order is also send to the manger also. At manager side after login web page is open which include all the information related to restaurant. Manager can add the food item, check the bill, change username password, and see the remaining food material in the kitchen in short manage all the activity.

CHAPTER 6
RESULTS

6.1 OFF CONDITION

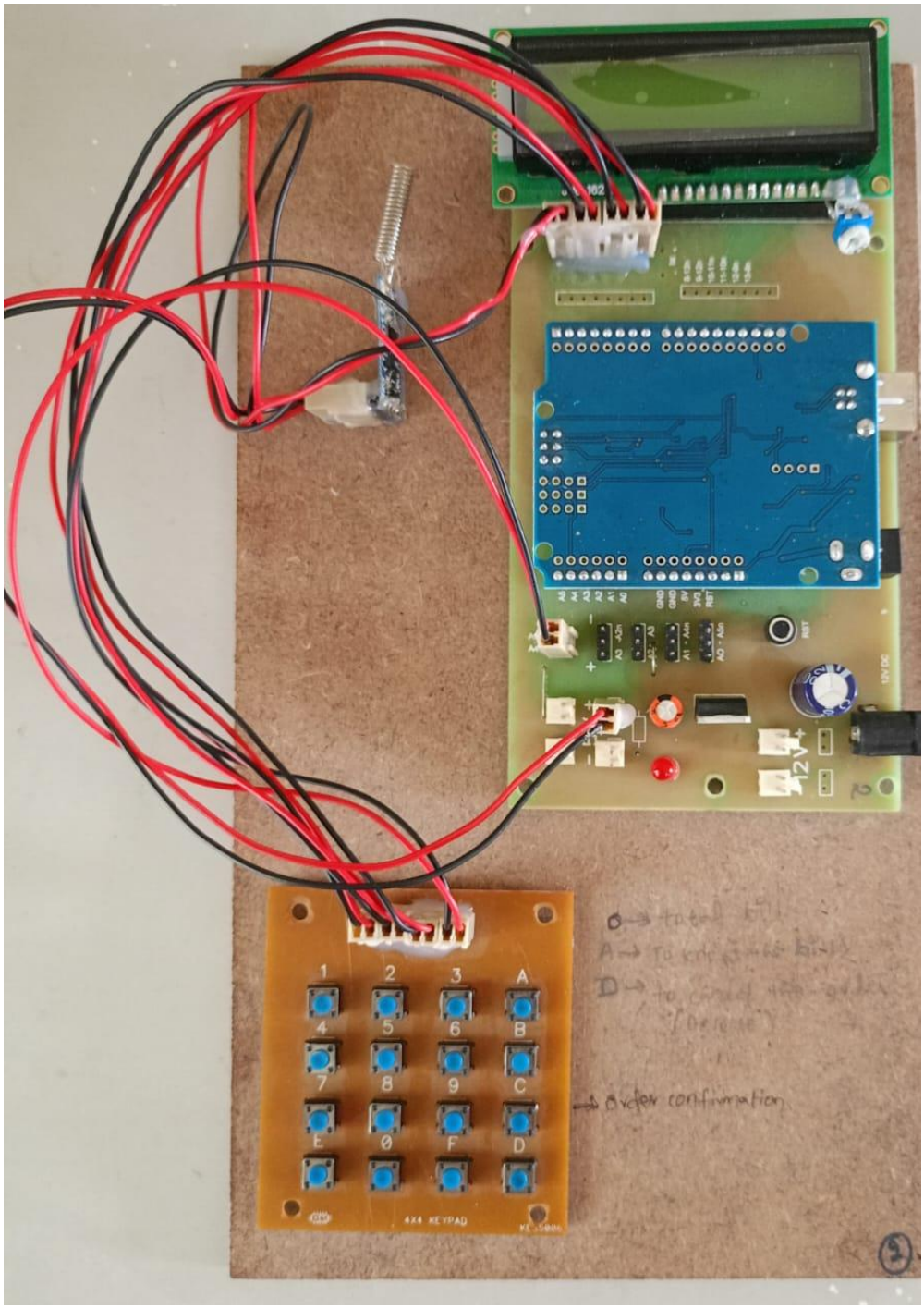
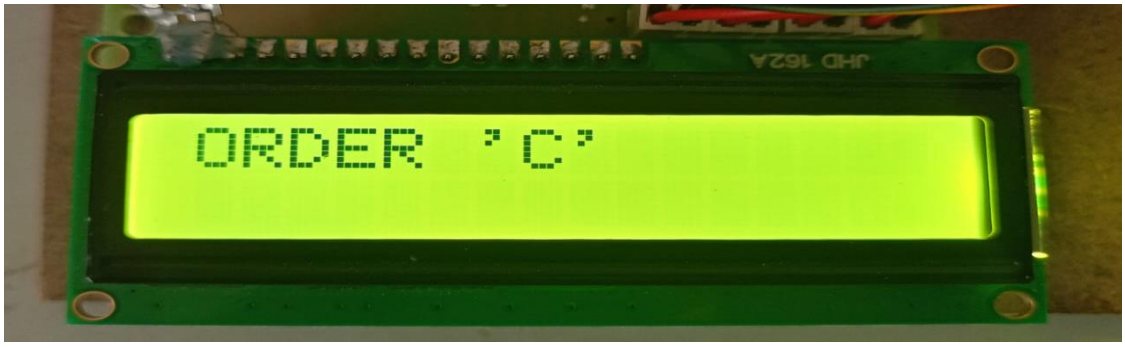


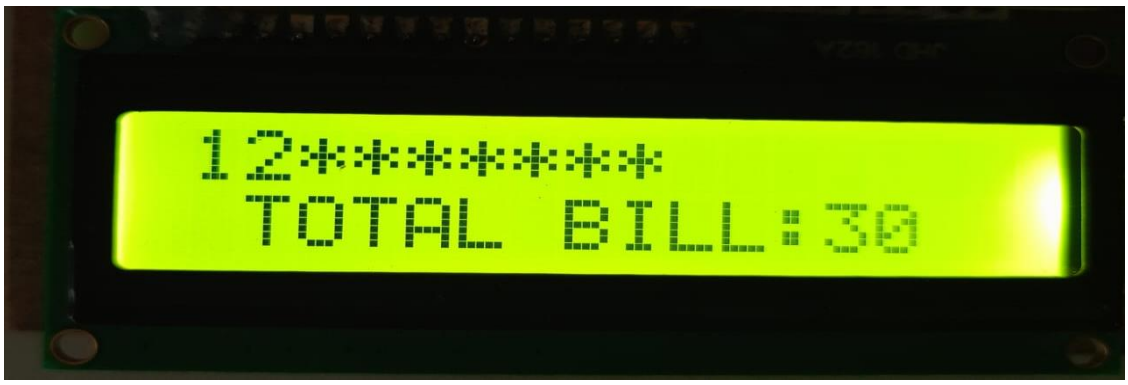
Fig 6.1:OFF Condition

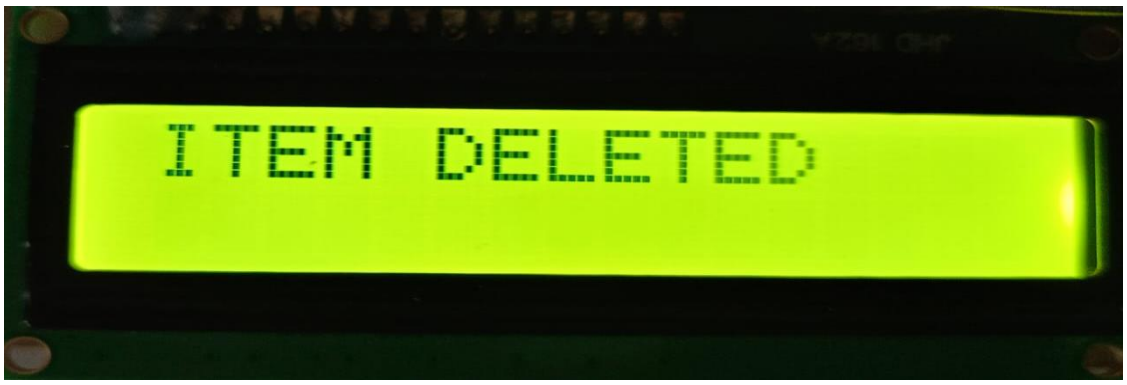
6.2 ON CONDITION




DISH:2 Rs 20/-


DISH:2 Rs 20/-
BILL:30 ORDERED


12*******
TOTAL BILL:30


ITEM DELETED

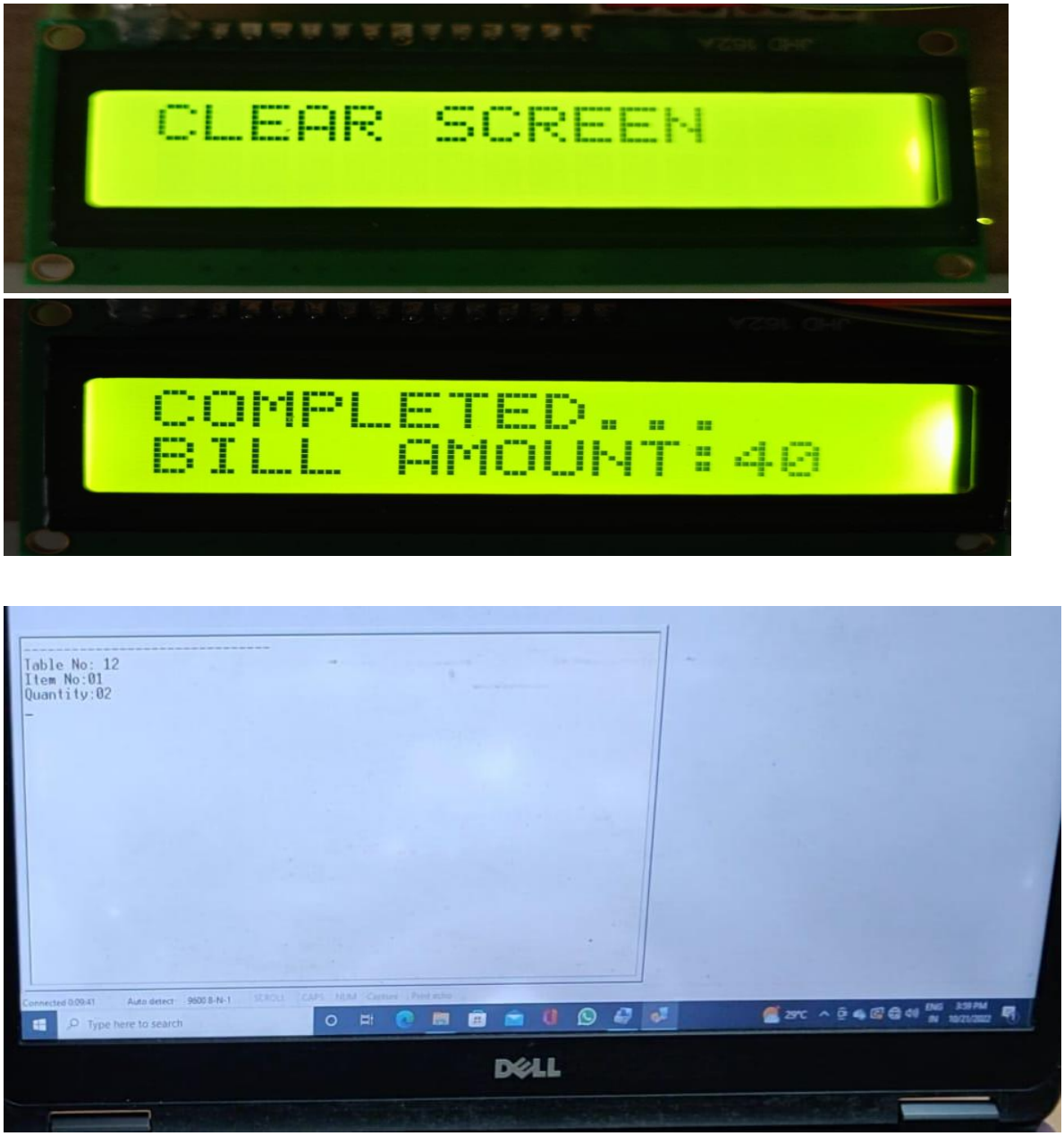


Fig 6.2:ON Condition

6.3 ADVANTAGES

- Simplifies ordering system
- Quick price updates
- It can give complete access to customer data
- 100% error free ordering processing rates
- Easy to operate

- Fast ordering system
- Labor work reduction
- Sophisticated security.

6.4 APPLICATIONS

- Restaurants
- Offices
- Industries
- IT hubs
- Institute staff rooms.

CHAPTER 7
CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSIONS

This system is convenient, effective and easy to improve the performance of restaurant's staff. In this system we present an automated food ordering system with real time customer feedback Increasing trends towards a smarter world, it will bring in a good profitable business. . It will also provide quality of service and customers satisfaction. Hotel is one where technology and advancements in technology have not been utilized to the fullest potential. Traditional method that is commonly been used in hotels is by taking the customer's orders and writing it down on a piece of paper. Many solutions have been proposed for solving this issue. This project is again one attempt in the same direction. In this paper we discuss the automation for food ordering system. This system makes use of zigbee as a communication device and LCD display module compatible with Aurdino as hardware.

7.2 FUTURE SCOPE

This Smart ordering system has great potential to make a big change in the hotel industry. This will entirely change the way the people order their food from restaurants or hotels. Currently this system is capable of having the Menu of a individual brand of restaurants. But it can be designed to encompass a Digital Menu of all the hotels and restaurants, all in one place which will give the customers more comfort. This way the customers can easily browse through the different menus of a variety of hotels and pick up the right dish according to there need and taste. This expansion to hold the menu details for all the hotels is the true potential of this system. This feature can also be added in future along with improvement in quicker and simpler, responsive user friendly interface. In this way this system can become a great product for the customers and entirely change the way food is ordered and the industry is operated.

REFERENCES/ BIBLOGRAPHY

- Ashutoshbhargave “Digital ordering system for restaurants using android” in international journal of scientific and research publication 2013.
- AjinkyakumarJadhav “Development of wireless ordering system for hotel” in international journal of emerging technology and advanced engineering 2015.
- Mayurdeshmukh, PiyushC.Mankar proposed “Implementation of smart restaurant with e-menu card” in international journal of computer Application 2015.
- International Journal of Computer Trends and Technology Smart Ordering System via Bluetooth The project is proposed with the Bluetooth technology as the communication medium, Peripheral Interface Controller (PIC) ,LCD and key pad
- Automated Food Ordering System with Real-Time Customer Feedback This project is proposed with wi-fi technology using Android mobile, laptop and broadband modem and wireless medium
- www.ijsred.com
- www.microcontrollerstudies/pic16f8xx.html
- www.zigbeeprotocols/crc.pdf
- www.touchtech/resistive4wire.html
- www.engineersgarage.com