Pioneer: Journal of Advanced Research and Scientific Progress (JARSP)
Volume: 02 Issue: 08 | 2023 ISSN: 2751-7551
http://innosci.org

Home Automation Using Arduino and Bluetooth

Zainab Zaki, Hiba Ahmed, Shahad Shakir, Dr. Ali Alnaji, Dr. Aws Basil
Middle Technical University, Electrical Engineering Technical College, Baghdad, Iraq

Abstract: The main objective of this project is to develop a home automation system using an Arduino board with Bluetooth being remotely controlled by any Android OS smart phone. As technology is advancing so houses are also getting smarter. Modern houses are gradually shifting from conventional switches to centralized control system, involving remote controlled switches. Presently, conventional wall switches located in different parts of the house makes it difficult for the user to go near them to operate. Even more it becomes more difficult for the elderly or physically handicapped people to do so, Remote controlled home automation system provides a most modern solution with smart phones. In order to achieve this, a Bluetooth module is interfaced to the Arduino board at the receiver end while on the transmitter end, a AUTOMATION application on the cell phone sends ON / OFF commands to the receiver where loads are connected. By touching the specified location on the GUI, the loads can be turned ON / OFF remotely through this technology.

INTRODUCTION

1.1. Overview:

In the past few years there are so many inventions in the field of consumer electronics such as cellular phone , air conditions , home security devices . and home theaters . All these appliances can be easily controlled by a single controller , using personal area network in a home environment . Busy environment and personal limitation the market is going towards the home automation and networking and Bluetooth is an ideal solution for this purpose. In buildings, temperature and other electronic devices can be easily controlled by home automation but high degree of computer work is involved. This report demonstrates a simple home automation system which consists of remote mobile, host controller, and several home appliances. At home, we generally operate (switch on / off) all the electrical and i electronics appliances such as fan, light, cooler, air conditioner, and so on through switches of the regular switch board. This manual switching of any home appliance is an inconvenient method for physically disabled or elders or even for normal young guys when frequent switching operation is required. Thus, this conventional manual switching method has to be overcome by an easier method of switching.

This can be done using an advanced switching method like a remote control for electronic home appliances. We can use the unconventional remote-control technology for controlling the home appliances easily without using the fixed wall switch boards. There are different types of remote-control technologies such as infrared or IR re mote technology, radio frequency or RF remote control technology, android based remote control, GSM based remote control. DTMF based remote control, Bluetooth remote control technology, and so on using different wireless communication systems.

1.2. Aim of the Project:

The aim of the project is to design and construct a home automation system that will remotely switch on or off any household appliances connected to it, using a microcontroller and Bluetooth-based android application.

1.3. Project Objectives:

The objective of this project is to implement a low-cost, reliable and scalable home automation system that can be used to remotely switch on or off any household appliance, using a microcontroller to achieve hardware simplicity, low-cost short messaging service for feedback, and voice dial from any phone to toggle the switch state.

CHAPTER 2 THEORETICAL BACKGROUND

2. System Description:

The proposed system has two main parts hardware and software. The hardware part consists of three main hardware components smartphone, an Arduino board, and a Bluetooth module. The software part consists of Arduino integrated development environment (IDE) and Bluetooth terminal smartphone application, which is used for wireless communication between smartphone and Arduino board.

2.1. Arduino Uno:

Arduino is a tool for making computers that can sense and control more of the physical world than your desktop computer. It is an open-source physical computing platform based on a simple microcontroller board and a development environment for writing software for the board. Arduino can be used to develop interactive objects, taking inputs from a variety of switches or sensors and controlling a variety of lights, motors, and other physical outputs. Arduino projects can be stand-alone, or they can communicate with software running on your computer (e.g., Flash, Processing.)[1]

The boards can be assembled by hand or purchased preassembled; the open-source IDE can be downloaded for free. 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 ceramic resonator, 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 an AC-to-DC adapter or battery to get started. As shown in figure 2.1

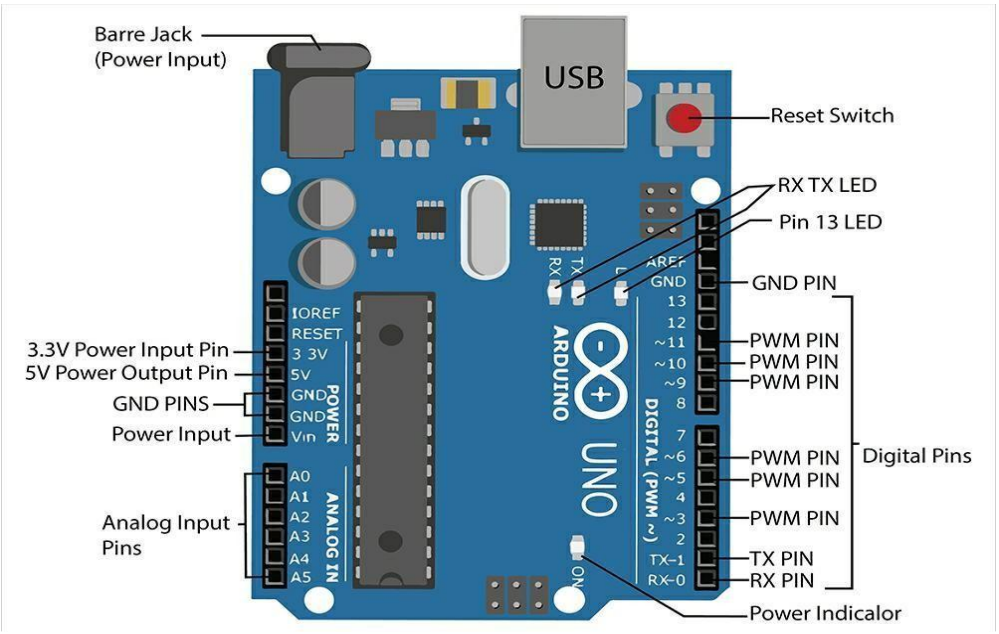


Figure 2.1 (Arduino Uno Board)

2.1.1 Arduino Uno Pinout Configuration:

Pin Category	Pin Name	Details
Power	Vin, 3.3V, 5V, GND	Vin: Input voltage to Arduino when using an external power source. 5V: Regulated power supply used to power microcontroller and other components on the board. 3.3V: 3.3V supply generated by the on-board voltage regulator. The maximum current draw is 50mA. GND: ground pins.
Reset	Reset	Resets the microcontroller.

Analog Pins	A0 – A5	Used to provide analog input in the range of 0-5V
Input/output Pins	Digital Pins 0 - 13	It can be used as input or output pins.
Serial	0(Rx), 1(Tx)	Used to receive and transmit TTL serial data.
External Interrupts	2, 3	To trigger an interrupt.
PWM	3, 5, 6, 9, 11	Provides 8-bit PWM output.
SPI	10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK)	Used for SPI communication.
Inbuilt LED	13	To turn on the inbuilt LED.
TWI	A4 (SDA), A5(SCA)	Used for TWI communication.
AREF	AREF	To provide reference voltage for input voltage.

2.1.2 Arduino Uno Technical Specifications

Microcontroller	ATmega328P – 8-bit AVR family microcontroller
Operating Voltage	5V
Recommended InputVoltage	7-12V
Input Voltage Limits	6-20V
Analog Input Pins	6 (A0 – A5)
Digital I/O Pins	14 (Out of which 6 provide PWM output)
DC Current on I/O Pins	40 mA
DC Current on 3.3V Pin	50 mA
Flash Memory	32 KB (0.5 KB is used for Bootloader)
SRAM	2 KB
EEPROM	1 KB
Frequency (Clock Speed)	16 MHz

Other Arduino Boards

Arduino Nano, Arduino Pro Mini, Arduino Mega, Arduino Due, ArduinoMKR1000 Wi-Fi Board, Arduino Leonardo

2.1.3. How to use Arduino Board

The 14-digital input/output pins can be used as input or output pins by using pinMode (), digitalRead () and digitalWrite () functions in Arduino programming. Each pin operates at 5V and

can provide or receive a maximum of 40mA current, and has an internal pull-up resistor of 20-50 KOhms which are disconnected by default. Out of these 14 pins, some pins have specific functions as listed below:

1. Serial Pins 0 (Rx) and 1 (Tx): Rx and Tx pins are used to receive and transmit TTL serial data. They are connected with the corresponding ATmega328P USB to TTL serial chip.
2. External Interrupt Pins 2 and 3: These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
3. PWM Pins 3, 5, 6, 9, and 11: These pins provide an 8-bit PWM output by using analogWrite () function.
4. SPI Pins 10 (SS), 11 (MOSI), 12 (MISO), and 13 (SCK): These pins are used for SPI communication.
5. In-built LED Pin 13: This pin is connected with a built-in LED, when pin 13 is HIGH – LED is on and when pin 13 is LOW, its off.

Along with 14 Digital pins, there are 6 analog input pins, each of which provides 10 bits of resolution, i.e. 1024 different values. They measure from 0 to 5 volts, but this limit can be increased by using AREF pin with an analog Reference() function.

- Analog pin 4 (SDA) and pin 5 (SCA) are also used for TWI communication using Wire library.

Arduino Uno has a couple of other pins as explained below:

- ✓ AREF: Used to provide reference voltage for analog inputs with analogReference () function.
- ✓ Reset Pin: Making this pin LOW, resets the microcontroller.

2.2. Bluetooth module HC-05

Bluetooth is a technology for wireless communication. It is designed to replace cable connections. It uses serial communication to communicate with devices. It communicates with the microcontroller using a serial port (USART). Usually, it connects small devices like mobile phones, PDAs, and TVs using a short-range wireless connection to exchange documents. It uses the 2.45GHz frequency band. The connection can be point-to-point or multi-point, where the maximum range is 10 meters. The transfer rate of the data is 1Mbps [5].

HC-05 Bluetooth module provides a switching mode between master and slave mode which means it is able to use neither receiving nor transmitting data.

Comparing it to the HC-06 module, which can only be set as a Slave, the HC-05 can be set as master as well, which enables making communication between two separate Arduino Boards. You can use a Bluetooth module simply for a serial port replacement to establish connection between MCU, PC to your embedded project and etc.

2.2.1. Pins of HC-05 Bluetooth Module

The HC-05 module supports UART, USB as well as SPI communication, and depending on the application, necessary pins can be used. In my case, the board uses the UART communication. Coming to the pins of the Bluetooth Module, generally, four pins are sufficient for successfully enabling a wireless communication link, but the modules produced nowadays come with six pins, namely: VCC, GND, TX, RX, EN, and STATE [5].

Figure 2.2. Bluetooth module HC-05 configuration.

2.2.2. HC-05 Pinout Configuration:

Pin Number	Pin Name	Description
1	Enable / Key	This pin is used to toggle between Data Mode (set low) and ATcommand mode (set high). By default, it is in Data mode
2	Vcc	Powers the module. Connect to +5V Supply voltage
3	Ground	Ground pin of module, connect to system ground.
4	TX– Transmitter	Transmits Serial Data. Everything received via Bluetooth will begiven out by this pin as serial data.
5	RX– Receiver	Receive Serial Data. Every serial data given to this pin will bebroadcasted via Bluetooth
6	State	The state pin is connected to on board LED, it can be used as afeedback to check if Bluetooth is working properly.
7	LED	Indicates the status of module ✓ Blink once in 2 sec: Module has entered Command Mode ✓ Repeated Blinking: Waiting for connection in Data Mode ✓ Blink twice in 1 sec: Connection successful in Data Mode
8	Button	Used to control the Key/Enable pin to toggle between Data andcommand Mode

2.3. Single-Channel Relay Module:

A relay is an electromechanical device that uses an electric current to open or closethe contacts of a switch. The single-channel relay module is much more than just a plain relay. It comprises components that make switching and connection easier and act as indicators to show if the module is powered and if the relay is active or not [6].

2.3.1. Single-Channel Relay Module Pin Description

Pin Number	Pin Name	Description
1	Relay Trigger	Input to activate the relay
2	Ground	0V reference
3	VCC	Supply input for powering the relay coil

4	Normally Open	Normally open terminal of the relay
5	Common	Common terminal of the relay
6	Normally Closed	Normally closed contact of the relay

2.3.2. Single-Channel Relay Module Specifications

- ✓ Supply voltage – 3.75V to 6V
- ✓ Quiescent current: 2mA
- ✓ Current when the relay is active: ~70mA
- ✓ Relay maximum contact voltage – 250VAC or 30VDC
- ✓ Relay maximum current – 10A
- ✓ Components Present on a 5V Single Channel Relay Module.

The following are the major components present on a relay module as shown in figure 2.3.3

5V Relay, Transistor, Diode, LEDs, Resistors, Male Header pins, 3-pin screw-type terminal connector, etc.

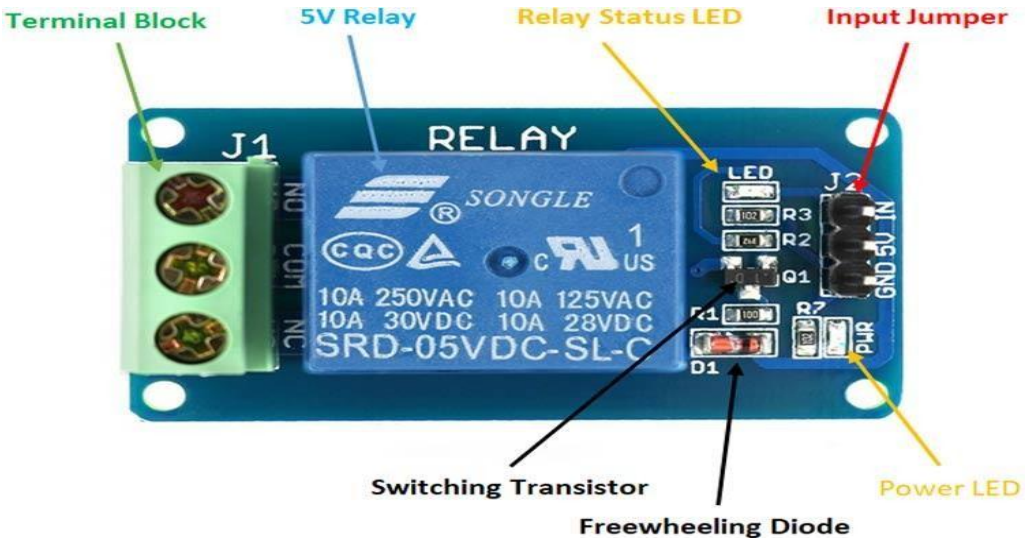


Figure 2.3.3. Major components present on a relay module.

2.3.3. Understanding 5V Single-Channel Relay Module:

The single-channel relay module is much more than just a plain relay. It contains components that make switching and connection easier and act as indicators to show if the module is powered and if the relay is active.

First is the screw terminal block. This is the part of the module that is in contact with mains, so a reliable connection is needed. Adding screw terminals makes it easier to connect thick mains cables, which might be difficult to solder directly. The three connections on the terminal block are connected to the normallyopen, normally closed, and common terminals of the relay.

The second is the relay itself, which, in this case, is a blue plastic case. Lots of information can be gleaned from the markings on the relay itself. The part number of the relay on the bottom says "05VDC", which means that the relay coil is activated at 5V minimum – any voltage lower than this will not be able to reliably close the contacts of the relay. There are also voltage and current markings, which represent the maximum voltage and current, the relay can switch. For example, the top left marking says "10A 250VAC", which means the relay can switch a maximum load of

10A when connected to a 250V mains circuit. The bottom left rating says "10A 30VDC", meaning the relay can switch a maximum current of 10A DC before the contacts get damaged.

The „relay status LED" turns on whenever the relay is active and provides an indication of the current flowing through the relay coil.

The input jumper is used to supply power to the relay coil and LEDs. The jumper also has the input pin, which when pulled high, activates the relay.

The switching transistor takes an input that cannot supply enough current to directly drive the relay coil and amplifies it using the supply voltage to drive the relay coil. This way, the input can be driven by a microcontroller or sensor output. The freewheeling diode prevents voltage spikes when the relay is switched off.

The power LED is connected to V_{CC} and turns on whenever the module is powered.

2.4. AC Bulb:

These are light-emitting devices. In this project, AC Bulbs are used instead of LEDS. Also, LEDs can be used to Automate and control. An incandescent bulb or lamp is light with wire filament which is heated to such a high temperature so that it glows by producing visible light i.e., incandescence. When an electriccurrent is passed through the filament, it gets heated. The glass bulb is filled with inert gas in order to protect the filament from oxidation. The figure-2.4 below depicts AC volt incandescent bulb.

Following are the advantages of Incandescent Bulb:

- It is less expensive due to lower initial cost.
- It is easier to dim with rheostats.
- It produces warmer color compare to fluorescent and tungsten-halogen lamps.
- It generates relatively high light output.
- It can be dimmed or controlled.
- It is easy to install.
- It is available in various shapes, sizes and applications.
- It can be switched ON immediately.

Figure 2.4. AC volt incandescent bulb.

2.5. Jumper Wires:

Jumper wires are used for making connections between items on your breadboard and your Arduino's header pins. Use them to wire up all your circuits. Jumper wires typically come in three versions: (1) male-to-male, (2) male-to-female, and

(3) female-to-female, as shown in Figure 2.10. Male ends have a pin protrudingand can plug into things, while female ends do not and are used to plug things into.Male-to-male jumper wires are the most common and what you likely will use most often [8][9].

Figure 2.5 Jumper wires types

2.6. Breadboard:

A breadboard is a rectangular board with many mounting holes. They are used for creating electrical connections between electronic components and single-board computers or microcontrollers such as Arduino and Raspberry Pi. The connections aren't permanent, and they can be removed and placed again. In fact, you can even replace components to customize your project or work on a completely different one using the same breadboard. The vertical columns of the breadboard are called terminals, while the long horizontal rows are called power rails because they are mostly used to connect the power supply to the breadboard. The positive rails are indicated by red lines, while the negative rails are indicated by black ones. Breadboards are used to help you connect components to complete your basic circuit. The reason it's called breadboard dates back to when electronics components were much bigger, and people would actually use wooden breadboards (boards used to cut bread) to connect electronic circuits [6]. The breadboard image is shown in Figure 2.6.

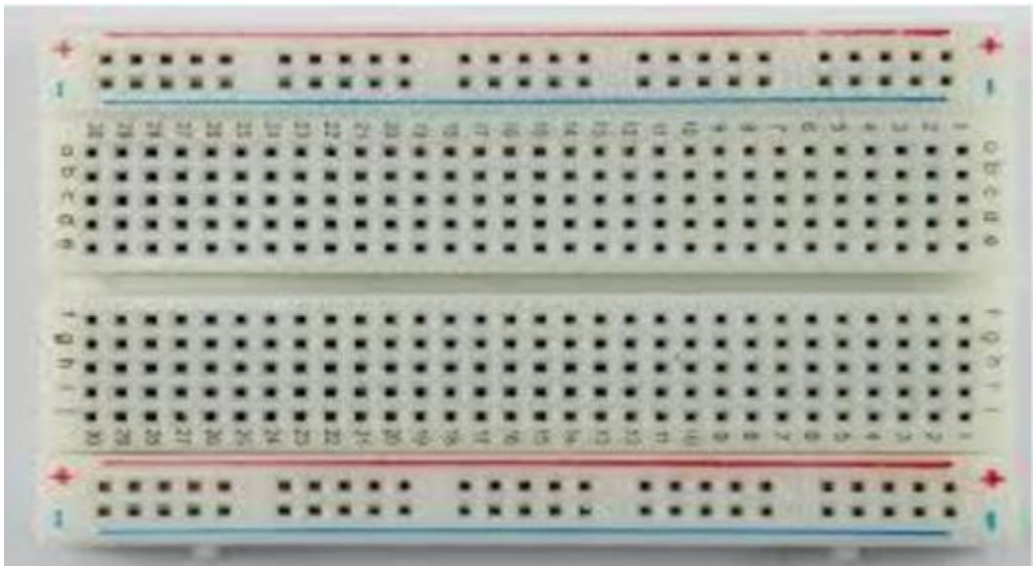


Figure 2.6: Arduino breadboard.

2.7. Communication Protocols:

In the Electronic world, communication protocols are the backbone links for an embedded system, thus, they are very important for us Seekers to understand what they are, why they are used, and the difference between communication protocols.

First thing first, wired communication protocols are simply a set of rules that allow two or more entities of a communication system to transmit information via a physical medium. Therefore, the syntax, semantics, and synchronization of communication and possible error recovery methods between communication systems are all defined by the term "protocol". Protocols can be implemented by both hardware and software or a combination of both. Further, each protocol has its own application area. In this blog, I will walk you through some of the most common communication protocols used in embedded designs, and of course, there are plenty more for you to explore!

2.8. Types of Communication Protocols in Embedded Systems:

Communication protocols are widely classified into two types, as stated in the diagram

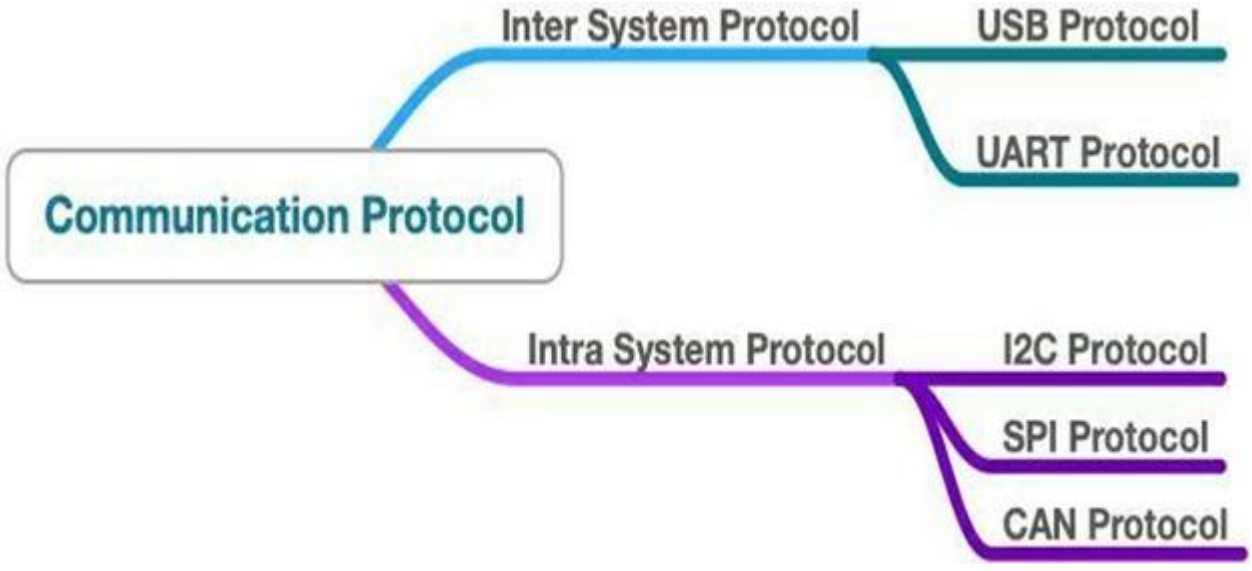


Figure 2.8

2.9. Inter System Protocol:

The inter-system protocol using to communicate the two different devices. Like communication between computer to microcontroller kit. The communication is done through an inter-bus system

A. USB Protocol:

Universal Serial Bus (USB) Protocols provide a fast master/slave interface using a tiered star topology supporting up to 127 devices with up to 6 tiers. A PC is generally the master or host, and each of the peripherals linked to it acts as a slave or device. USB 1.X and 2.X, use 4 lines, Vcc, Ground, and D+/D-, a twisted pair of data lines using NRZI (Non-Return to Zero Invert) encoding, as the USB pinout shown below. Data is transmitted in the form of packets, which is composed of 8 bits (1 byte) with the LSB (Least Significant Bit) transmitted first. REF: Components 101

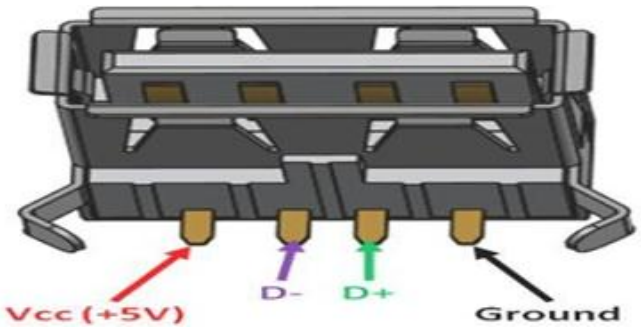


Figure A: USB protocols

Advantages of the USB protocol:

- ✓ Fast and simple.
- ✓ Plug and play functionality.
- ✓ Near universal adoption.

Disadvantages of the USB protocol

- ✓ Requires powerful master device.
- ✓ Specific drivers are required.

B. UART Protocol:

Universal Asynchronous Receiver/Transmitter (UART) is not a communication protocol but a physical circuitry that converts parallel data into serial data. In UART communication, two UARTS communicate directly with each other. The transmitter UART converts data from a controlling device like a CPU into serial form, and transits it in serial to the receiving UART, which then converts the serial data back into parallel data for the receiving device. REF: *Circuit Basics*

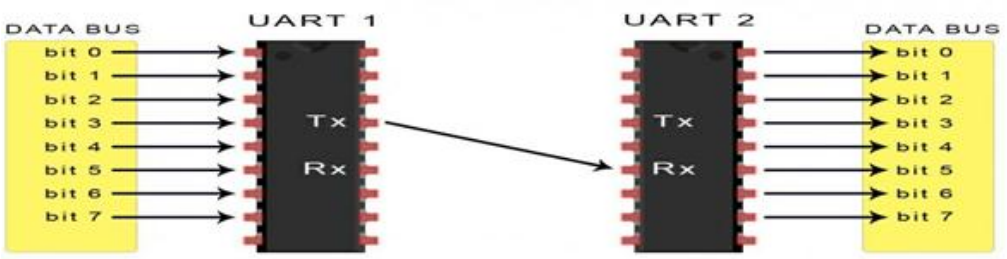


Figure B. UART

Only two wires are needed to transmit data between two UARTs; data flows from Tx pin of the transmitting UART to the Rx pin of the receiving UART. UART transmits data asynchronously, which means that no clock signal is needed in transmitting and receiving data. Thus, UART uses start and stop bits with actual data bits. When the receiving UART detects a start bit, it starts to read the incoming bits at a specific frequency known as the baud rate. Both UARTs must operate at about the same baud rate. The baud rate between the transmitting and receiving ends can only differ by about 10% before the timing of bits gets too far off.

Advantages of UART protocol:

- ✓ Only requires two wires.
- ✓ No clock signal is necessary.
- ✓ Cost-effective.

Disadvantages of UART protocol:

- ✓ The size of the data frame is limited to a maximum of 9 bits.
- ✓ Doesn't support multiple master/slave functionality.

2.10. Intra System Communication Protocols:

The Intra system protocol establishes communication between components within the circuit board. In embedded systems, intra-system protocol increases the numberof components connected to the controllers.

On-device communication protocols are known as intra-system communication protocols. These are used in scenarios such as sending a sensor's value to an MCU. Most of our Grove Modules communicate via these protocols.

a) Serial Peripheral Interface (SPI)

SPI (Serial Peripheral Interface) is a common communication protocol used by many different chipsets. Devices that communicate via SPI are in a master/slave relationship. The master is the controlling device (usually a microcontroller), while the slave (usually a sensor, display, or memory chip) takes instruction from the master. SPI uses 4 wires named MOSI (Master Out Slave In), MISO (Master In Slave Out, SS (Slave Select), and SCLK (Serial Clock). Further, the SS line is usedto select the appropriate slave by pulling the SS low where it is normally held high. If the SS line is low the chip will begin to take instruction from the master. However, a master must have multiple GPIO pins available if it is to speak with multiple slaves. To get around this, some devices will use a multiplexor to select the slaves. REF: *Circuit Basics*

SPI is full-duplex, meaning it is able to send and receive data simultaneously. While the master is providing new instructions or information via the MOSI line, the slave can return messages or results via the MISO line. This allows for faster overall throughput.

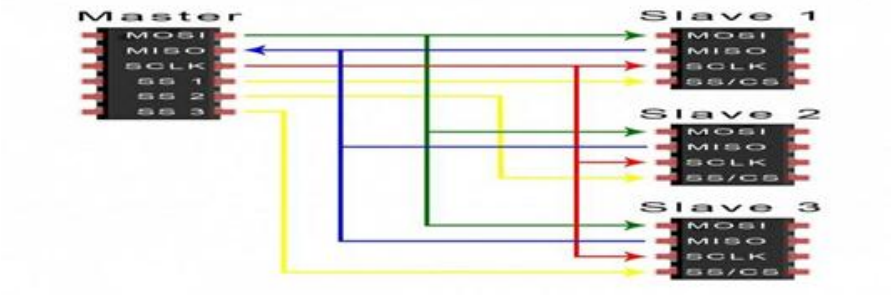


Figure a) Serial Peripheral Interface (SPI)

Advantages of SPI protocol

- ✓ One master supports multiple slaves
- ✓ No start and stop bits, so the data can be transmitted continuously without interruption or overhead
- ✓ Can transmit and receive simultaneously (full-duplex)
- ✓ Higher data transfer rate than I2C

Disadvantages of SPI protocol

- ➢ Require more wires than other protocols
- ➢ More slaves lead to circuit complexity
- ➢ Related to 1 and 2, this scales horribly

b) Inter-Integrated Circuit (I2C) Communication Protocol

(I2C) Inter-Integrated Circuit, pronounces "eye-two-see" or "eye-squared-see", combines the best features of SPI and UART. With I2C, you are able to connect multiple slaves to a single master (just like SPI) or have multiple masters controlling single or even multiple slaves. This is extremely useful when you are logging data into a single LCD from more than one micro-controller. For example, our Grove – LCD RGB Backlight and our Grove – Quad Alphanumeric Display also use I2C for communications. REF: *Circuit Basics*

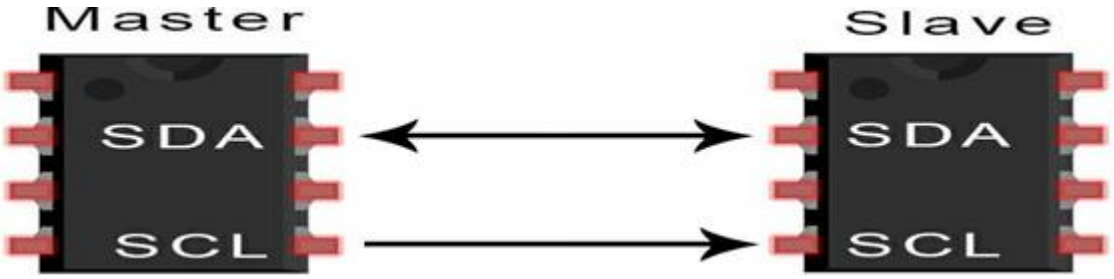


Figure b) Inter-Integrated Circuit (I2C) Communication Protocol.

Similar to UART, I2C only uses two wires to transmit data between devices: SDA(Serial Data) and SCL(Serial Clock). Like SPI, I2C is synchronous, so the output of the bits is synchronized to the sampling of bits by a clock signal shared between the master and the slave. The clock signal is always controlled by the master.

Advantages of I2C protocol

1. One master support multiple slaves connectivity
2. Support multiple masters and multiple slaves
3. Higher data transfer rate than I2C
4. Can support over 1000 devices using just two lines

Disadvantages of I2C protocol

- ✓ Limited speed
- ✓ A single bus line for transmitting and receiving
- ✓ The size of the data frame is limited to 8 bits

c) CAN (Controller Area Network) Protocols

CAN protocol can be defined as the set of rules for transmitting and receiving messages in a network of electronic devices. It was designed for robust and flexible performance in harsh environments, particularly for industrial and automotive applications. In particular, CAN was developed to reduce cable wiring,so the separate electronic control units (ECUs) inside a vehicle could communicatewith only pair of wires, as stated below. REF: *All about circuits*

Figure c) : CAN (Controller Area Network) ProtocolsAdvantages of CAN protocol

- ✓ Secured and fast protocol
- ✓ Low cost and reliable

Disadvantages of CAN protocol

- ✓ Automotive oriented.
- ✓ Complex protocol.

CHAPTER 3 EXPERIMENT

3.1. Introduction:

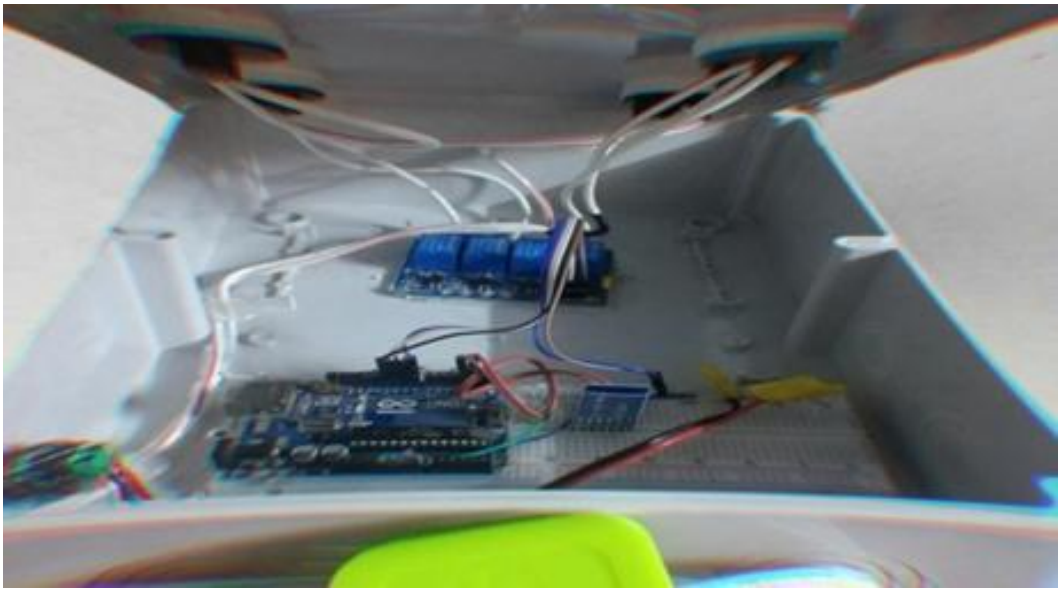
In this chapter, How to connect components and turn on Bluetooth and the results of experiments that were conducted to control lamps (simulating medical devices) via Bluetooth and using Arduino are presented.

3.2. Connection Table [10]:

Arduino UNO		HC-05 Bluetooth
(+5V) VCC		VCC
GND (Ground)		GND (Ground)
TX Pin		RX Pin
RX Pin		TX Pin
Arduino UNO		Relay Module
(+5V) VCC		VCC
GND (Ground)		GND
12 Pin		OUT Pin
Arduino UNO	**LED**	220-ohm Resistor
11 Pin	Anode Pin	
GND (Ground)		Terminal 1
	Cathode Pin	Terminal 2
220 Volt AC Supply	**AC Bulb**	Relay Module
		Normally Open

Phase		Common
	Terminal 1	Normally Closed
Neutral	Terminal 2	

- ✓ All the connections should be done carefully.
- ✓ switch off the power supply during connection.
- ✓ Cross-check all the connections after completion.



3.3. Arduino Code for home automation;

```
-#include <SoftwareSerial.h>
SoftwareSerial mySerial(3, 2);// RX, TX
String device;
#define R1 8
#define R2 9
#define R3 10
#define R4 11
void setup() {
  Serial.begin(9600);
  mySerial.begin(9600);
  pinMode(R1,OUTPUT);
  pinMode(R2,OUTPUT);
  pinMode(R3,OUTPUT);
  pinMode(R4,OUTPUT);
  digitalWrite(R1,HIGH);
  digitalWrite(R2,HIGH);
  digitalWrite(R3,HIGH);
  digitalWrite(R4,HIGH);
}
void loop() {
 while (mySerial.available()){ delay(10);
  char c = mySerial.read();
  device += c;}
if (device.length() > 0) {
    device = device.substring(0,1);
    Serial.println(device);
    if (device == "A"){digitalWrite(R1,LOW);}
    if (device == "a"){digitalWrite(R1,HIGH);}
    if (device == "B"){digitalWrite(R2,LOW);}
    if (device == "b"){digitalWrite(R2,HIGH);}
    if (device == "C"){digitalWrite(R3,LOW);}
    if (device == "c"){digitalWrite(R3,HIGH);}
    if (device == "D"){digitalWrite(R4,LOW);}
    if (device == "d"){digitalWrite(R4,HIGH);}

    device = "";}
}
```

CHAPTER 4 RESULTS AND DISCUSSION

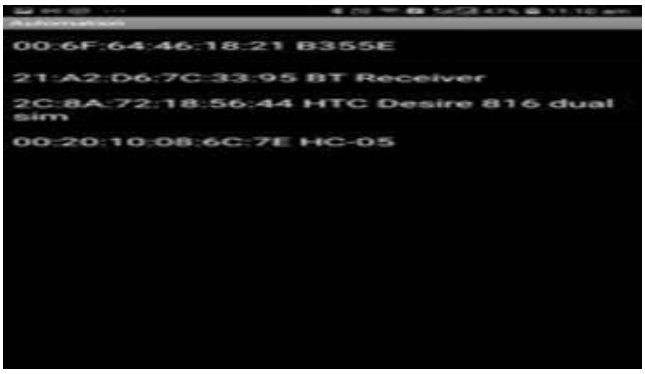
4.1. Download Android Bluetooth control app apt

Here the code is given you only need to upload the code to the Arduino using Arduino IDE software. If you don't know the process you can refer to our tutorial given in the Arduino tutorial under the blog section. If you have any query you canask us in the comment section and also you can reach us on Instagram.

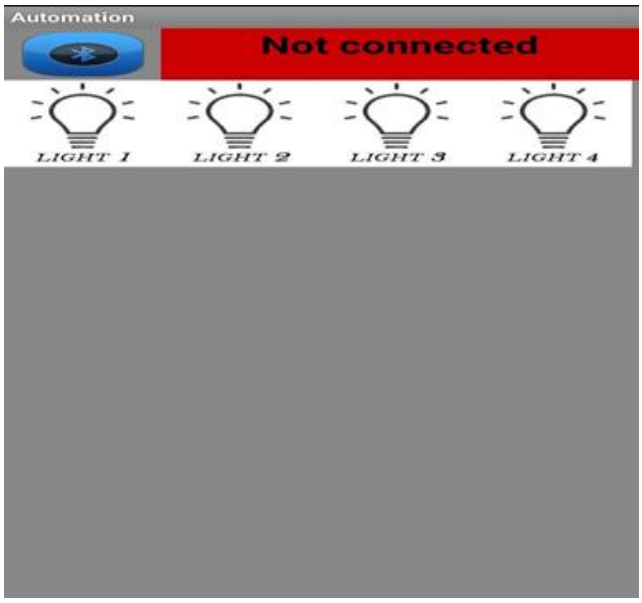
4.2. How to configure the app with the Bluetooth:

- First, open your phone Bluetooth setting

- Pair with a new device.
- connect with the HC-05 with 1234 default password.
- Click on the "Connect to Bluetooth" option in the app.



- Select HC-05 to connect to the same Bluetooth.
- After connected Bluetooth, your app screen will be like the above image now you can control the system, with the on and off button.



CHAPTER 5 CONCLUSIONS AND RECOMMENDATIONS

5.1. Conclusions:

This circuit is considered a means of remote control of medical and home devices as well (that is, it is considered a simulation of devices via Bluetooth. We can control the on and off of household and medical devices, and any other device can be connected to this circuit instead of lamps.

5.2. Future Works:

1. For example, when this project is linked to the field of medical devices, this circuit enables the patient in hospital rooms to control the devices in the room without the help of the nurse. He can turn on and off any device in the room, for example, the TV.
2. An alarm device can also be connected to this circuit so that when the patient feels that he needs to call the doctor, he can with one click throughhis phone via Bluetooth and by pressing the play button to give an alarm to the doctor and call him when needed and without needing to be monitored by a nurse or medical team.
3. Bluetooth can control the devices in the room only. That is, it has a limited control range, but we can also replace Bluetooth with Wi-Fi, where through Wi-Fi also, we can turn on and off devices remotely and on a larger scale. For example, we can connect the devices for intensive care on this circuit so that the doctor can control the devices connected to the patient from a distance, and more than one device can also be linked so that the doctor can control the devices connected to several patients at the same time.
4. A respirator or heart monitor measuring body temperature these devices can be connected to this circuit and through the nurse, she gives a signal to the specialist doctor when she notices a defect in the patient and the doctor, through Wi-Fi controls these devices connected to the patient.

REFERENCES

1. https://www.bartleby.com/essay/Arduino-Uno-FK2DAAS8J38W
2. https://components101.com/microcontrollers/arduino-uno
3. https://components101.com/wireless/hc-05-bluetooth-module
4. https://www.electronicshub.org/hc-05-bluetooth-module/
5. https://electrosome.com/hc-05-serial-bluetooth-module/amp/
6. https://components101.com/misc/breadboard-
7. https://www.seeedstudio.com/blog/2019/07/03/basic-electronics-wired- communication-
8. https://www.exploringarduino.com/parts/jumper-wires/
9. http://blog.sparkfuneducation.com/what-is-jumper-wire,
10. https://techatronic.com/home-automation-using-arduino-and-bluetooth/