Czech Technical University

Information and Automation Technology

Department of Instrumentation and Control Engineering

FACULTY OF MECHANICAL ENGINEERING CTU IN PRAGUE

Home Automation Using Arduino

Bachelor Thesis

Fahad Bin Ahmed

Supervisor: Ing. Libor Straka, Ph.D.

Declaration of Authorship

I, Fahad Bin Ahmed, declare that this thesis titled, 'Home Automation Using Arduino' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a bachelor's degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

____

Date:

____

"Nothing can suppress a human's curiosity."

Eren Yeager

BACHELOR'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Ahmed Fahad Bin** Personal ID number: **463777**

Faculty / Institute: **Faculty of Mechanical Engineering**

Department / Institute: **Department of Instrumentation and Control Engineering**

Study program: **Theoretical Fundamentals of Mechanical Engineering**

II. Bachelor's thesis details

Bachelor's thesis title in English:

Home Automation using Arduino

Bachelor's thesis title in Czech:

Domácí automatizace s Arduinem

Guidelines:

1. Do the research home automation and possibilities of Arduino.
2. Design the circuit and implement it on the Arduino board.
3. Design a simple example of home automation system.

Bibliography / sources:

[1] Schwartz Marco: Arduino Home Automation Projects. Packt Publishing, 2014. ISBN: 9781783986071
[2] Banzi, Massimo: Getting Started with Arduino. Maker Media, Inc (2011). ISBN 9781449309879

Name and workplace of bachelor's thesis supervisor:

Ing. Libor Straka, Ph.D., Division of electrotechnics, FME

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **30.04.2021** Deadline for bachelor thesis submission: **24.08.2021**

Assignment valid until: ____________

Ing. Libor Straka, Ph.D.
Supervisor's signature

Head of department's signature

prof. Ing. Michael Valášek, DrSc.
Dean's signature

III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

Date of assignment receipt

Student's signature

CVUT-CZ-ZBP-2015.1

Czech Technical University

Abstract

Faculty of Mechanical Engineering
Department of Instrumentation and Control Engineering

Bachelor of Science

by

Fahad Bin Ahmed

This thesis is based on the research of the technology of Arduino which is an open-sourced electronic prototyping platform based on the relationship between easy-to-use hardware and software. It is intended for the great imaginative minds who are interested in creating interactive objects or environments. Specifically, I have chosen home automation to showcase the influence of Arduino in the coming years, so the home automation circuit is built around an Arduino Uno board, a HC-05 Bluetooth module, 4-Channel relay module, a breadboard and some jumper wires. After studying and experimenting with the pins on the boards we form a relay circuit which acts as a switch for the devices. The power supply I will be using for the relay circuit will be a 12V DC component since connection with your actual appliances is going to require a 220V AC power supply which is quite dangerous without the right environmental conditions and safety measures. Further connection and circuit details will be mentioned later. In the end a conclusion is drawn after the work done in this project and some results are obtained which is a very small overview of what influence Arduino and Internet of Things could have in our societies and lives in the coming few years.

Acknowledgements

I would like to thank my supervisor Ing. Libor Straka, Ph.D. to give me the opportunity to work and research on something I was interested in along with the guidance and knowledge he has provided me with. At the same time, I would like to thank the Czech Technical University in Prague for providing me with the required tools and equipment for completing this project along with the opportunity to complete my bachelor's degree in "Information and Automation Technology". I would also like to thank and appreciate Ing. Vladimir Hlavac, Ph.D. for all the assistance they have provided me with during the tough times throughout the duration of the Project.

I appreciate the efforts the professors and the study department have taken in helping me reach where I am today. The knowledge gained from my time in this university has helped me make and work on this project which reflects all the knowledge I have grasped during my time here.

I would also like to thank my family and friends for providing me with the emotional support I required to get through university.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Home Automation or in other words domotics is basically building automation for your home called a smart home or smart house. Building automation refers to the automatic centralized control of a building's HVAC (i.e. Heating, Ventilation and Aircon), electrical, lighting & Shading, Access Control & Security Systems and other interrelated systems through a BAS which is a Building Automation System. The main objectives of building automation to be looked at are improved occupant comfort, efficient operation of all the systems in the building, reduction in energy consumption along with reduced operating and maintaining costs, bumping up the security measures, remote access & operation and finally the improved life cycle of electrical equipment and other utilities.[1]

Home automation has been around us for a while now and according to the charts the global market for home automation has reached an estimated 60.2$ billion by the end of 2020 and it still is on the rise. However, it is considered to be luxurious and not for the common man.

The content of this thesis will be focusing heavily on understanding home automation using a brilliant platform (Arduino) which will help us tackle the issues especially the economical issues of home automation.

1.1 IOT (Internet of Things)

This is the main concept behind my project. Internet of Things basically refers to the network of physical objects i.e. "things"—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the Internet. The main concept of a network of smart devices was discussed as early as 1982, with a modified Coca-Cola vending machine at Carnegie Mellon University. It was the first ARPANET-connected appliance which was able to report its inventory and whether newly loaded drinks were cold or not. Peter T. Lewis who is the co-founder of the first U.S. Cellular Tel Company "Cellular One" mentioned that IoT is the integration of people, processes and technology with connectable devices and sensors to enable remote monitoring, status, manipulation and evaluation of trends of such devices.[2]

1.1.1 Applications of IoT

The extensive varieties of applications for IoT devices can be divided into consumer commercial, industrial, and infrastructure spaces[3]. A few examples of the applications are:

- Medical and Healthcare – Also known as Internet of Medical Things (IoMT) is referred as smart healthcare which focuses on collection and analysis of data for creating a digitized healthcare system, connecting available medical resources and healthcare services.
- V2X communications – Also known as vehicular communication systems which consists of three main components vehicle to vehicle communication (V2V), vehicle to infrastructure communication (V2I) and vehicle to pedestrian communications (V2P). V2X is the first step to autonomous driving and connected road infrastructure.
- Manufacturing - The IoT can connect various manufacturing devices equipped with sensing, identification, processing, communication, actuation, and networking capabilities. Network control and management of these equipment, asset and situation management, or manufacturing process control allow IoT to be used for industrial applications and smart manufacturing in a way where we can save a lot of time and energy and avoid a percentile loss of manpower and money.
- Agriculture and Food – The numerous IoT applications in farming such as collecting data on temperature, rainfall, humidity, wind speed, pest infestation, and soil content helps automate farming. The overall objective here is that the data from sensors, in addition with the farmer's knowledge and intuition about his/her farm, can help increase farm productivity, and help reduce costs. Researchers at the Loughborough University based on IoT technology have designed an innovative digital food waste tracking system which supported the decision making in real-time to combat and reduce the food waste issues. So Basically, food safety, improving the logistics, enhancing the supply chain transparency and food wastage reduction is what being focused on with IoT technology.
- Military Applications – IoMT also known as The Internet of Military Things is the application of IoT technology focusing on surveillance,

reconnaissance and other combat-related objectives. A few examples are IoBT also known Internet of Battle Things executed by the U.S. Army Research Laboratory (ARL) that focuses on the basic science related to IoT technology that enhances the capabilities of Army soldiers and The Ocean of Things is a project which is a DARPA-led program designed to establish an Internet of Things across large ocean areas for the purposes of collecting, monitoring, and analyzing environmental and vessel activity data.

- Metropolitan Scale Deployment – Imagine a smart city without any human involvement whatsoever. Well, there are a few projects ongoing or planned in the future which have demonstrated the IoT technology on a large scale. For example, Songdo, South Korea, the first of its kind which is fully equipped and wired, is gradually being built, with approximately 70 percent of the business district completed as of June 2018. Much of the city is planned to be wired and automated, with little or no human intervention.[5]
- Home Automation – Finally the focus of my research is as simple as a system designed to take control of IoT technology in the palm of your hands using a few relatively cheaper equipment and a little bit of your brain. This system is going to help you manage your home with minimum efforts and in turn reduce power consumption with the principle of time management.

1.1.2 Arduino

One of the key components in this project is the microcontroller also known as an Arduino board which makes the IoT technology a reality. Arduino is an open-source hardware and software company, project and user community which design single board microcontrollers and microcontroller sets for building digital devices. It's hardware products are licensed under a CC-BY-SA license, while the software is licensed under the GNU Lesser General Public License. These boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various breadboards used for prototyping or shields which are expansion boards and other circuits. The boards also feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs. These microcontrollers can be programmed using C and C++ programming languages using the Arduino Language which is a standard API. The best part is that Arduino provides an integrated development environment (IDE) and a command line tool.

The Arduino project first began in 2005 as a tool for students at the Interaction Design Institute Ivrea in Ivrea, Italy. Their objective was to provide an efficient and an easy way for novices and professionals to create devices that interact with their environments using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats and motion detectors.

Hardware: As an open-source hardware product most of the boards consist of an Atmel 8-bit AVR microcontroller with a varying amount of flash memory, pins and features. The boards use single or double-row pins or female headers that facilitate the incorporation into other circuits and connections for programming. Also, most of these boards include a 5 V linear regulator which is a system used to maintain the steady voltage. It acts as a resistor since the resistance of the regulator varies with accordance to the input voltage and the load. The boards also include a 16 MHz crystal oscillator that's uses the mechanical resonance of a vibrating crystal of piezoelectric material which creates an electric signal with constant frequency (often used to keep track of time) or a ceramic resonator which is an electronic component consisting of a piece of a piezoelectric ceramic material with two or more metal electrodes attached to it. The resonant mechanical vibrations in the device generate an oscillating signal of a specific frequency.

Figure 1 An example of a crystal oscillator

[https://upload.wikimedia.org/wikipedia/commons/c/c7/16MHZ_Crystal.jpg]

Figure 2 An example of a ceramic resonator

[https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTtDFr3J68-sOCDmaPutBNxX9tH1Fq-vWzIyg&usqp=CAU]

To simplify uploading of programs to the on-chip flash memory the microcontrollers are pre-programmed with a boot loader. Most of the current Arduino boards are programmed via Universal Serial Bus (USB). The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. The 14 digital I/O pins are further divided into 6 of which can produce a pulse-width modulated signals. PWM signals also known as pulse-duration modulation (PDM waves) is a procedure used to reduce the average power delivered by an electrical signal, by effectively chopping it up into discrete parts. The average value of voltage (and current) fed to the load is controlled by turning the switch between supply and load on and off at a fast rate.

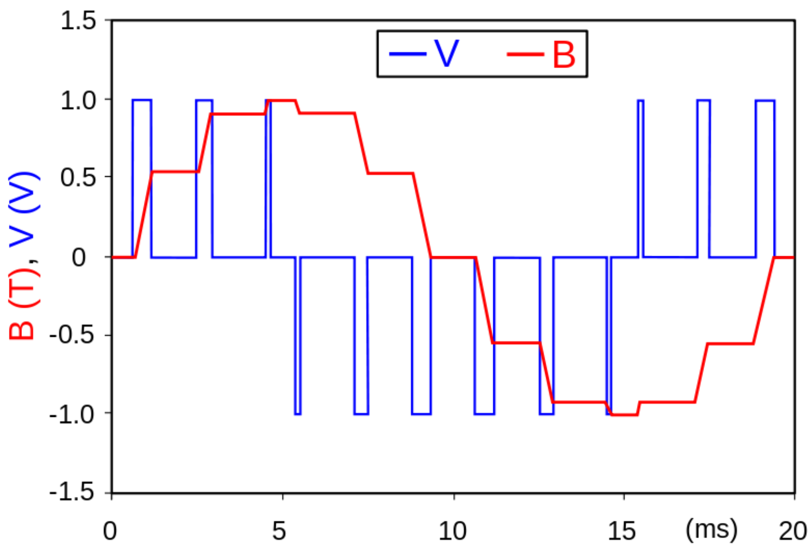

Figure 3 An example of PWM in an idealized inductor

[https://i.stack.imgur.com/9xU0r.png]

And 6 other pins which can act as analog inputs or digital I/O pins.

Software: Any language may be used to program Arduino hardware using compilers that produce machine code for the target processor. Atmel provides a development environment for 8-bit AVR and 32-bit ARM Cortex-M based microcontrollers: AVR Studio (older) and Atmel Studio (newer). The Arduino integrated development environment (IDE) is a Java-based cross-platform application (for Windows, macOS, and Linux) that originated from the IDE for the languages Processing and Wiring. Additionally, it contains a message area, a text console, a toolbar with buttons for common functions, and an operation menu hierarchy. Using special rules of code structuring, Arduino IDE supports languages like C and C++.AVRDUDE, a program included in the Arduino IDE, converts executable code into a text file in hexadecimal encoding for loading into the Arduino board by a loader program embedded in its firmware.

This open-source nature of the Arduino project has allowed the publication of various free software libraries that other developers are using to augment their projects.

SHIELDS: Printed Circuit expansion boards also known as shields which can be easily attached to the pins are compatible with Arduino boards. Several shields can be made by DIY. Some of the applications are providing motor controls for 3D printing and other applications, satellite navigation, ethernet, LCD and breadboarding which allows us to go crazy with prototyping.

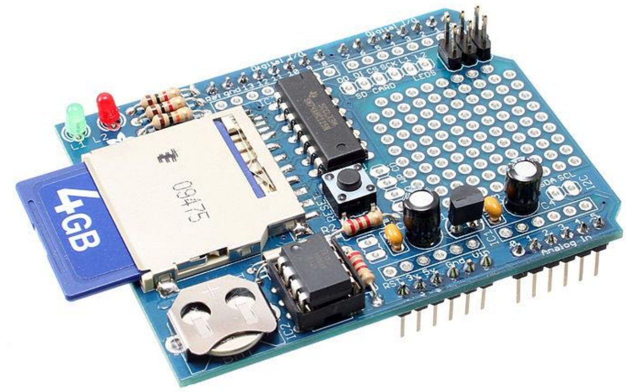

Figure 4 A datalogging shield with a Secure Digital (SD) card slot and a real-time clock (RTC) chip as well as a few places for adding components and modules.

[https://upload.wikimedia.org/wikipedia/commons/thumb/e/ee/ARSH-09-DL_03.jpg/1024px-ARSH-09-DL_03.jpg]

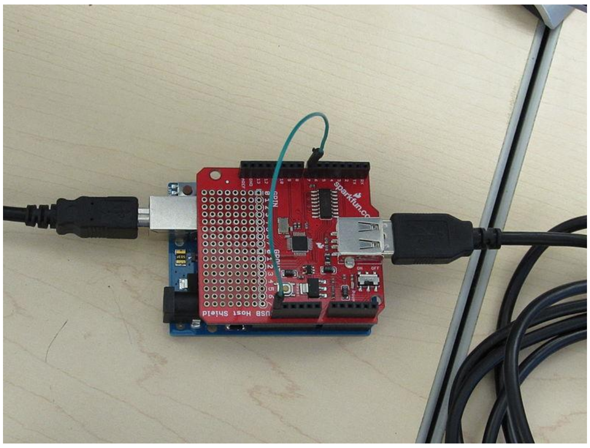

Figure 5 A USB host shield which allows an Arduino board to communicate with USB devices such as a keyboard/mouse .[https://upload.wikimedia.org/wikipedia/commons/thumb/1/17/Closeup_of_usbhost_shield_with_jumper.JPG/800px-Closeup_of_usbhost_shield_with_jumper.JPG]

Official boards

17 versions of the Arduino hardware have been commercially produced and Some examples of them are:

1) Arduino Uno - A microcontroller board developed by Arduino.cc, the Arduino Uno is based on the Microchip ATmega328P microcontroller. The word "uno" means "one" in Italian. It was chosen to mark the initial release of Arduino Software which makes it the first in a series of USB-based Arduino boards.

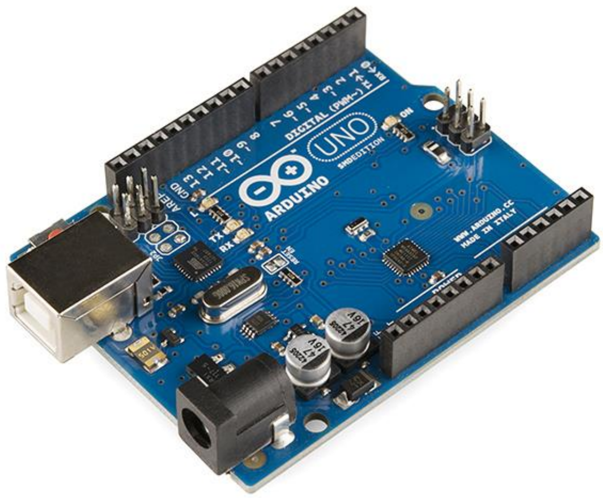

Figure 6 Arduino Uno SMD R3 [https://upload.wikimedia.org/wikipedia/commons/3/38/Arduino_Uno_-_R3.jpg]

2) Arduino Mega - The Arduino Mega is a microcontroller board based on the ATmega2560. It is unique due to the 54 digital I/O pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button.

Figure 7 Arduino 2560 Mega

[https://upload.wikimedia.org/wikipedia/commons/thumb/0/01/Arduino_Mega.jpg/800px-Arduino_Mega.jpg]

3) Arduino Micro - A built-in USB interface eliminates the need for an additional processor to communicate with a computer, allowing the Micro board to be viewed as a mouse and keyboard in addition to a virtual serial port (CDC).[6]

Figure 8 Arduino Micro

[https://upload.wikimedia.org/wikipedia/commons/thumb/c/c9/2x3_pin_header_on_Arduino_Micro.jpg/800px-2x3_pin_header_on_Arduino_Micro.jpg]

1.2. Summary

The main focus in the first chapter was the concept or the basic principle behind this project which is the technology of Internet of Things. As I have researched a bit and mentioned the applications of IoT technology, we understand the advantages coming from it and how to tackle the disadvantages such as efficiency and economic issues which will be brought upon after the project. Home automation is the specific example that I have chosen to showcase the extent of IoT technology by creating a DIY circuit focused on an Arduino microcontroller. The Arduino board was chosen due to its convenient open-source nature and the freedom it gives us to manifest such complex technologies.

The Arduino microcontroller's main function is to read the inputs and produce an output in the form of a response. For the output response, it will be required to program the board using its custom Arduino programming language.

This project is a small-scale representation of what IoT technology can achieve, it is basically to showcase the principle behind this technology and also understanding the issues that come with it while proposing a few solutions on how to tackle them.

Chapter 2

2.1 Equipment and Components

We will be focusing on the technical side of this project in this chapter where several components that are required for the circuit connection will be discussed thoroughly. Some of the components that will be essential to the functionality of this project are an Arduino microcontroller, a HC-05/HC-04/HC-06 Bluetooth module, a 4-Channel Relay module, some jumper wires & a breadboard and finally some test devices such as LED's etc.

2.1.1 Arduino Microcontroller (Arduino UNO R3 Clone)

Figure 9 Arduino Uno R3 (Clone)

[https://www.jsumo.com/arduino-uno-r3-clone-with-usb-cable-usb-chip-ch340-2-13-B.jpg]

For this Project I have wanted to be the most efficient in every way possible and since Arduino can be considered as luxury for some of the crowd out there, we will be using a clone version of the Arduino Uno R3 microcontroller board.
So, WHAT is a Microcontroller?
Microcontrollers are integrated circuits that can be viewed as small computers. They contain a processor, memory, and input/output peripherals. They are designed to perform certain operations in embedded systems. Because they are constructed with adequate onboard memory and generic I/O pins for operations, microcontrollers may directly interact with sensors and other programmable components. They can run just one program and not a full operating system without requiring extra processing components. First it gathers the input, processes the information, and then gives an output based on a certain action from the information gathered.

A brief description of the specifications of the microcontroller will be explained by the pin diagram and a table.

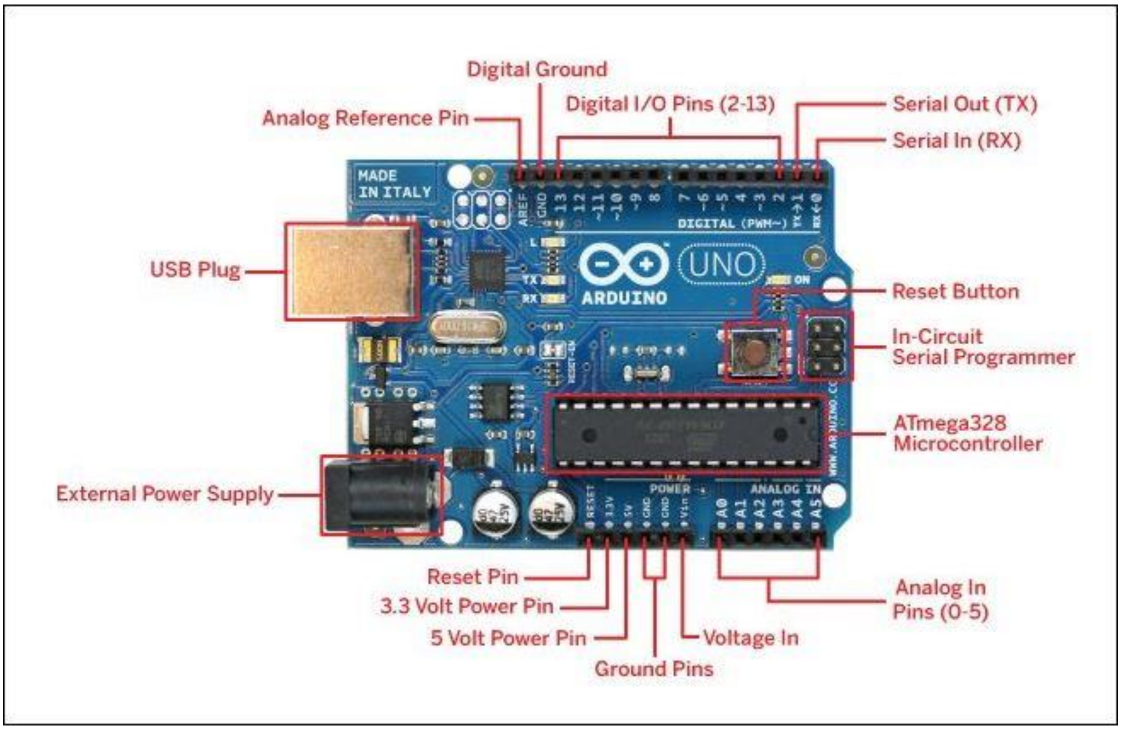

Figure 10 Arduino Uno R3 Pin Diagram[https://www.elprocus.com/wp-content/uploads/Arduino-Uno-Pin-Diagram.jpg]

No.	Specifications	Purpose
1	Vin Pin	The Vin pin or the input voltage pin can be used to supply voltage
2	5V Power Pin	Regulated power supply used to power the microcontroller and its components
3	3.3V	A 3.3 supply voltage can be generated with the onboard regulator. The highest drawn current will be 50 mA
4	GND	Ground pins for the RPS
5	Reset	Resets the microcontroller
6	A0 – A5 pins	These pins are used for analog inputs from 0-5V
7	Digital Pins 2- 13	Can be used as input or output pins
8	Digital pins 0(RX) & 1(TX)	These pins are used to receive and transmit TTL serial data. Rx – receiving pin Tx – transmitting pin
9	AREF	Also known as the analog reference pin which configures the reference voltage used for analog input

Table 1 The microcontroller specifications

For programming the board, it is preferable to use the official integrated development environment (IDE) for the Arduino which supports java and C language. The code, which is also called a sketch, is then compiled in the software, and transferred to the board by a USB cable. It also comes preprogrammed with a bootloader that allows us to upload a new code without the use of an external hardware programmer. The bootloader is a section of the program memory which runs before the main code runs. First, it checks if the computer is trying to program it and uploads the program into the memory (without overwriting the bootloaders memory) However, if the computer is not trying to upload a code then it tells the microcontroller chip to run the code that's already stored in the memory. [7]

2.1.2 HC-05 Bluetooth Module

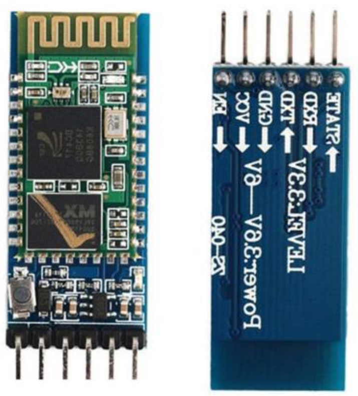

Figure 11 example of a HC-05 Bluetooth module[https://images-na.ssl-images-amazon.com/images/I/61ebRN7-TrL.jpg]

One of the key components used in this circuit is the HC-05 Bluetooth module is popular since you can add two-way wireless functionality to your projects.
The Pinout configuration is explained in the table.

No.	Specifications	Purpose
1	Enable Pin	This pin toggles between Data Mode (low) and AT command mode (high) It's in Data mode by default.
2	Vcc Pin	Powers the module (5v)
3	Tx-transmitter	Transmits serial data. Everything received via Bluetooth will be given out as serial data
4	Rx -Receiver	Receives serial data. All data given to this pin is transmitted through Bluetooth

6	State	The state pin is used as a feedback to check if Bluetooth is working properly.

Table 2 Pinout config of Bluetooth module

The Operating Voltage for the module is in the range of 4V to 6V (Typically +5V) and the Operating Current is around 30mA. The module works with Serial communication (USART) and is TTL compatible. It utilizes a transmission technology in which the data signal is modulated by a narrowband carrier which changes frequency ("hops") over a wide band of frequencies. The hopping may seem random, but it is prescribed by an algorithm known to the receiving system. This method is known as the frequency hopping spread spectrum. It can an operate in Master, Slave or Master/Slave mode which allows it to be easily interfaced with devices such as laptops or mobile phones. The module communicates with the help of USART at 9600 baud rates hence it is easy to interface with any microcontroller that supports USART. So, if you are looking for a Wireless module that could transfer data from your computer or mobile phone to microcontroller or vice versa then this module might be the right choice for you.

Some of the alternative modules that can be used are the HC-04/HC-06 module which are very similar to the HC-05 Bluetooth module.

Some popular applications of this module are wireless communication between two microcontrollers, Communication with Laptop, Desktops and mobile phones, Data Logging application, Wireless Robots and Home Automation.[8]

2.1.3 4-Channel Relay Module

One of the other key components in this circuit is the four-channel relay module which contains four 5V relays which associates switching and isolates the components which in turn makes interfacing with the microcontroller easy using minimum components and connections.

No.	Specifications	Purpose
1	GND	This is the ground reference for the module
2	IN 1-4	Input pins to activate the relay according to their number
3	Vcc	Power supply for the relay module
4	JD-Vcc	Alternate pin for power supply

Table 3 Pinout config of a 4 Channel relay module

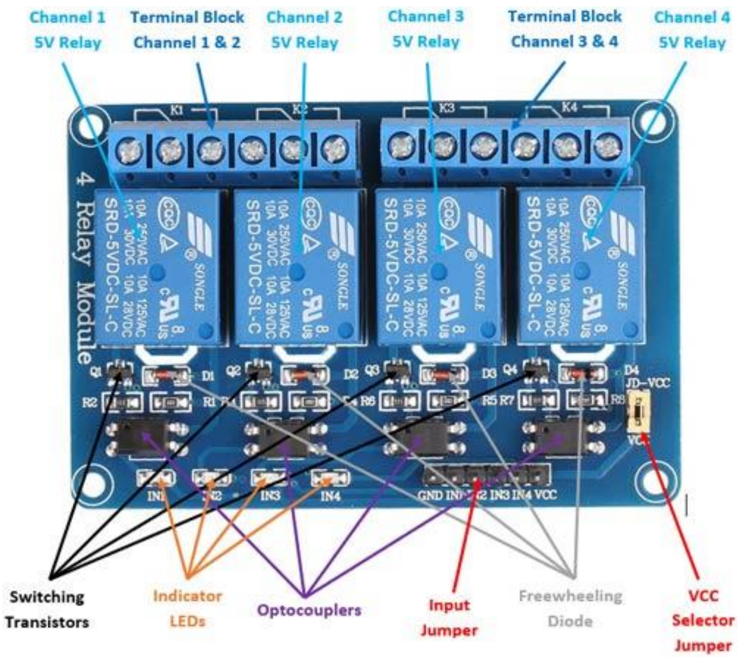

Figure 12 Understanding the 4-Channel Relay module [https://components101.com/sites/default/files/component_pin/Four-Channel-Relay-Module-Pinout.jpg]

The four-channel relay module has four 5V relays as well as the accompanying switching and isolating components such as transistors, diodes, LED's, etc. There are two terminal blocks, each with six terminals, and two relays share each block. The terminals are screw style, allowing for quick and easy hookups to mains wiring. The four relays on the module are rated for 5V, which implies that when there is about 5V across the coil, the relay is engaged. Each relay's contacts are rated for 250VAC, 30VDC, and 10A in each case, as indicated on the body of the relays.

The switching transistors serve as a buffer between the relay coils, which require high currents, and the low-current inputs. They magnify the input signal so that the coils in the relays may be driven. Because the coils represent an inductive load, the freewheeling diodes avoid voltage spikes across the transistors when the relay is turned off. The indicator LEDs glow when the coil of the respective relay is energized which indicates that the relay is active. The optocouplers form an additional layer of isolation between the load being switched and the inputs. The input jumper contains the main V-CC, GND, and input pins for easy connection using female jumper wires.

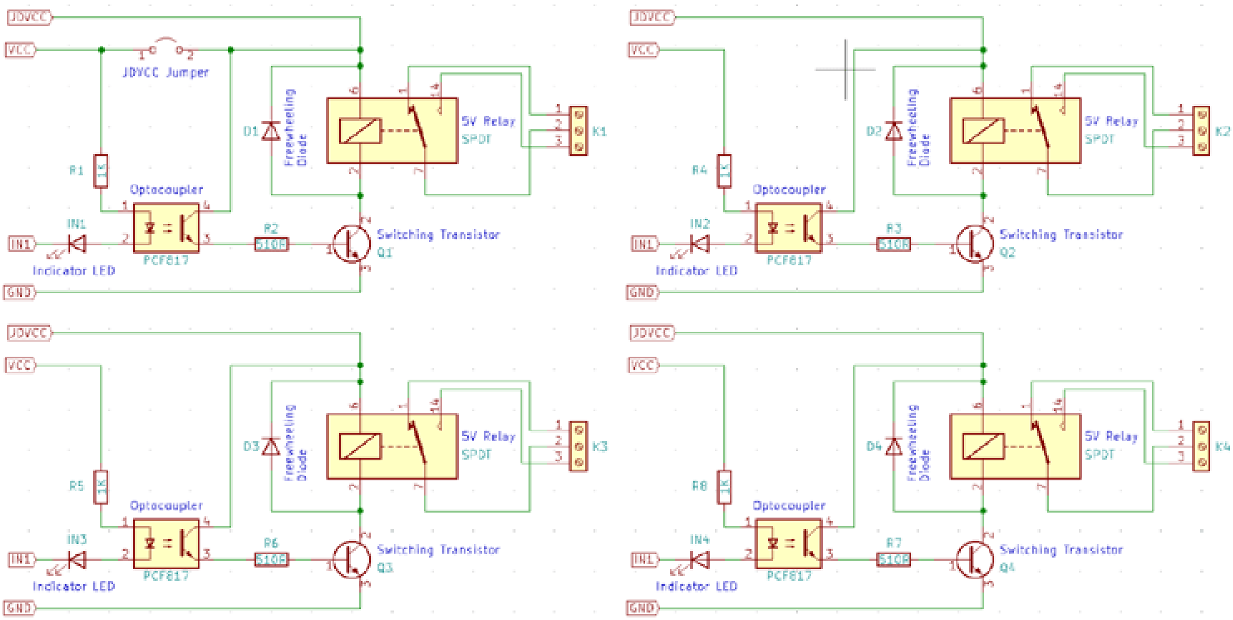

Figure 13 Internal circuit of the relay attached to the module [https://components101.com/sites/default/files/inline-images/Four-Channel-Relay-Module-Circuit-Diagram.png]

Each relay on the board has the same circuit, and the input ground is common to all four channels. Because there is an optional additional layer of isolation, the driver circuit for this relay module differs differently from standard relay driving circuits. When the JD-VCC jumper is shorted, the relay and the input share the same VCC; when the jumper is open, a separate power supply is required to power the relay coil and optocoupler output.

The main objective behind using this module is that it can be used to switch multiple loads at the same time since there are four relays on the same module. This is useful in creating a central hub from where multiple remote loads can be powered. Therefore, it is very useful for tasks like home automation where the module can be placed in the main switchboard and can be connected to loads in other parts of the house that can be controlled using a microcontroller such as the Arduino Uno.[9]

2.1.4 Breadboard & Jumper Wires

A breadboard is a plastic board with many small holes that is used to build and test circuits. It has holes on it that are joined internally in a specific manner, as illustrated in the illustration below. The green line connecting the holes indicates that they are connected internally. Power is represented by the red line, which is generally connected to the power rail. The blue line denotes Ground, which is normally connected to the ground of the circuit.

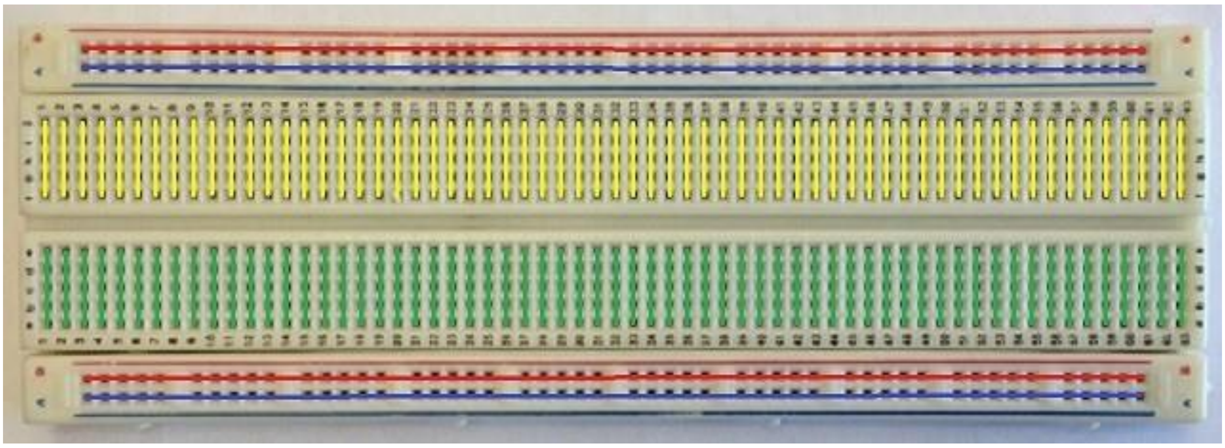

Figure 14 Breadboard internal patterning

[https://components101.com/sites/default/files/inline-images/Breadboard-Connection.jpg]

Follow the circuit schematic and connect one component in a line to utilize a breadboard for your circuit. After double-checking all the connections, connect the battery last. Keep an eye out for typical blunders like mixing ground and supply, connecting to the wrong rail, and not properly setting ICs, among others.[10]

Jumper Wires - A jump wire also known as a jumper, jumper wire, jumper cable, DuPont wire or cable is an electrical wire or a group of electrical wires in a cable, with a connector or pin at each end that is normally used to interconnect the components of a breadboard or other prototype in a test circuit, internally or with other equipment or components, without the need for soldering.

Some of the different types of jumper cable are:

- Solid tips – used to connect on/with a breadboard or female header connector.

- Crocodile tips – are used, among other applications, to temporarily bridge sensors, buttons, and other elements of prototypes with components or equipment that have arbitrary connectors.
- Banana connectors – are commonly used on test equipment for DC and low-frequency AC signals.
- Registered jack) – are commonly used in telephone (RJ11) and computer networking (RJ45).
- RCA connectors – are often used for audio, low-resolution composite video signals, or other low-frequency applications requiring a shielded cable.
- RF connectors – are used to carry radio frequency signals between circuits, test equipment, and antennas.
- RF jumper cables - Jumper cables are smaller and more bendable corrugated cables which is used to connect antennas and other components to network cabling.[11]

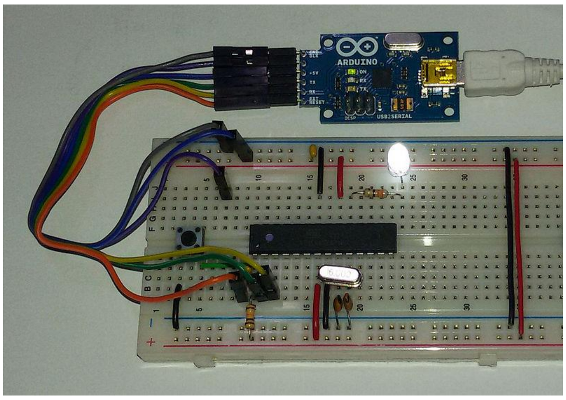

Figure 15 Example of how jumper wires are used [https://upload.wikimedia.org/wikipedia/commons/thumb/8/8f/Arduino_Breadboard_ATmega328P_USB2Serial.jpg/220px-Arduino_Breadboard_ATmega328P_USB2Serial.jpg]

2.1.5 Test/Sample Devices

Normally any electronic device can be controlled using this circuit depending on the scale of the circuit and the tolerances of the components used. In this particular protype given the small tolerances of the components any device which works under 12V DC is applicable. Some examples are LED bulbs, car cigar lighters, etc.
With Professional help if I switch to a 220AC power supply for the circuit, with certain possibility most of the home appliances can be controlled via this circuit.

3.1 Construction of the Prototype Circuit

This thesis will now dive into the electrical assembly of the components and the programming of the microcontroller board, after covering the major components needed in the previous section.

The prototype circuit is supposed to represent us the principle of IoT on a small scale which is your basic home automation.

3.1.1 Schematics

. The word "schematic" is basically the blueprint of your project. So, a schematic diagram is a graphical representation of a plan or a model that is presented in a simple, accessible way. Schematics use simple lines and symbols to communicate information such as what, how, and where.

To get the general idea of the connections between the components, I have come up with a circuit diagram also known as a schematic using block diagrams for the main electronic components in my project.

After further research I have noticed that there are quite a few variations of connections between the electrical components to achieve positive feedback. The schematic I have presented is a simple one with direct connection between the Bluetooth module and the Arduino microcontroller. The test device used is your average LED bulb.

To Understand the connections between the components better, the schematic presented below is without a breadboard and jumper wires which I have used in the real prototype for easier connections between the components.

After going through with various connections, I have come up with a schematic on Smartdraw that best describes the circuit.[12]

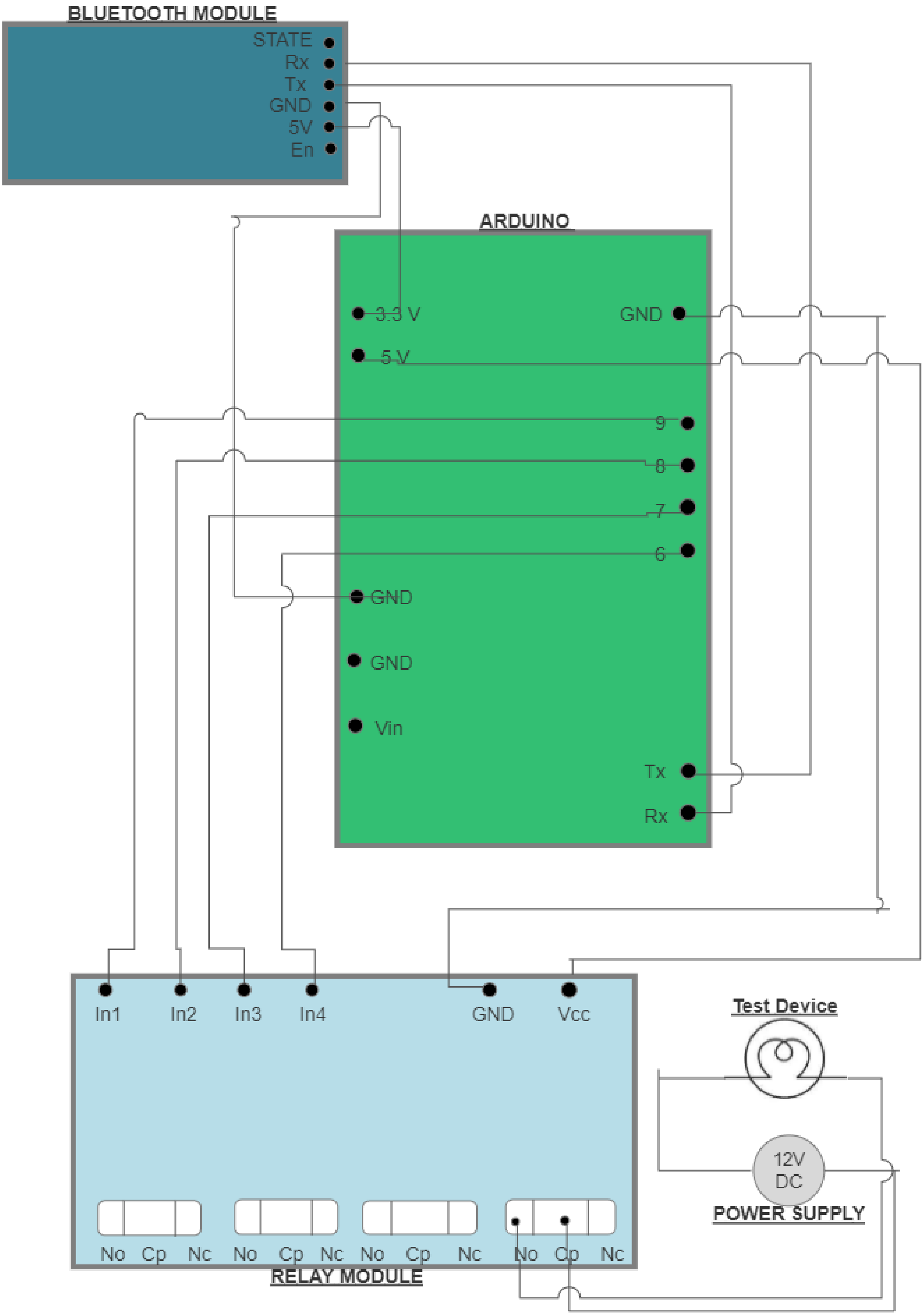

Figure 16 Example Schematic Of my prototype circuit
[https://lh3.googleusercontent.com/zj0fovlD8K9GlLTZNpSINmbjTzjXiOek8Txvwhnn6m1h1aeQ-KMSIUqWUhPip8QN6FT7KA=s85]

3.1.2 Connections of the Hardware components in the circuit

Using a breadboard and a few jumper wires the connection between the electrical components is made easier.
A relation of the connections between the Bluetooth module and the Arduino microcontroller is briefly described in this table below. [13]

Bluetooth Module	**Arduino Board**
Rx Pin	Tx Pin
Tx Pin	Rx Pin
GND Pin	GND Pin
5V Pin	3.3V Pin

Table 4 Connections of the Bluetooth module and the Arduino board

The TXD pin and RXD pin, or the transmitter pin and receiver pin of the Bluetooth module must be connected to serial ports to RXD pin and TXD pin respectively which are predetermined on the Arduino board as the serial communication ports. These serial ports are known as Universal Asynchronous Receiver/Transmitter (UART) and are physical circuits in a microcontroller. The UART's main purpose is to transmit and receive serial data. The transmitting UART converts parallel data into serial form before transmitting it to the receiving UART before its converted back to parallel data.
The GND Pins of both the components are connected to each other respectively.
This is important since the ground is where the voltage is 0 and it acts as a common/reference point for all the electric components.
The power supply pins of both the components are connected respectively which may vary from 3.3V to 5V.[15]

Similarly, a relation of the connections between the other major 2 components which is the relay module, and the Arduino board is explained briefly in the table below.

Relay Module	Arduino Board
In1 Pin	No. 9 Pin
In2 Pin	No. 8 Pin
In3 Pin	No. 7 Pin
In4 Pin	No. 6 Pin
Vcc Pin	5V Pin
GND Pin	GND Pin

Table 5 Connections of the Relay module and the Arduino board

The relay control ports i.e. In1, In2, In3 and In4 pins are supplied with low level signals from the Arduino board when connected to the No. 9, No. 8, No. 7, and the No. 6 pins respectively. This allows the circuit to be controlled by the microcontroller.

The Power supply pins, and the ground pins have the same role as explained before and are connected with each other.[16]

3.1.3 Software Programming and Algorithms

The microcontroller will need to be programmed when all of the components have been assembled and connected in a circuit. It is suggested, but not required, to utilize the Arduino software, often known as the IDE, with Arduino boards (integrated development environment). The Arduino IDE allows you to build programs and then upload them to the device for execution.[18]

These programs are created in the Arduino programming language, which is based on a collection of C++ functions that are called from pre-installed libraries and sent to a C++

compiler directly. A fairly simple structure is employed while programming the Arduino board. The Arduino program is divided into two sections: preparation and execution, each with a collection of statements.[19]

The preparation block is known as the setup function which is the first to be executed when the program starts running. Its main purpose is to define the functionality of each pin and the initial state of the Arduino upon boot. For example, when creating a basic setup function between the relay module and the Arduino microcontroller according to the connections of my prototype model, the function will look as follows:

void setup () {

pinMode(relay1, OUTPUT); // Define the relayIn1 pin as an output

pinMode(relay2, OUTPUT); // Define the relayIn2 pin as an output

pinMode(relay3, OUTPUT); // Define the relayIn3 pin as an output

pinMode(relay4, OUTPUT); // Define the relayIn4 pin as an output

}

The second block, the execution block, is a loop function that stores and executes statements and conditions for reading inputs and triggering outputs. In programming languages, a loop is a predetermined sequence of instructions that are continuously repeated until a given condition is fulfilled or the program is terminated. The primary instructions for the circuit will be included in this block.

First, we define the*SoftwareSerial* library and use the command *mySerial* to define the ports for transmitting and receiving serial data. Defining the relay ports and introducing a char *Val* is the next step.

Using the digitalWrite() command we set the trigger pins of the relay module on high in order to start sending pulses of ultrasonic waves. After the waves bounce off an obstacle, they're picked up by the receiver (echo pin).[20] For example:

digitalWrite(relay1,HIGH);

digitalWrite(relay2,HIGH);

digitalWrite(relay3,HIGH);

digitalWrite(relay4,HIGH);

The next line of codes is for the serial data from the Bluetooth android app which we can either build using the MIT app inventor site or use a 3rd party application on the app store. This is where the char *Val* plays a role in reading the serial data which we print using the mySerial.read command

```
void loop() {

//Check data serial from Bluetooth android App

if( mySerial.available() >0 ) {

val = mySerial.read();

Serial.println(val);

}
```

The final lines of codes are to define the relay in particular situations such as when one of the relay is on/off or when all of the relays are on/off. We use an else if loop to carry out these operations. By sending low signals, the relay is switched on when needed.

For example:

```
//Relay is on

if( val == '1' ) {

digitalWrite(relay1,LOW); }

else if( val =='2' ) {

digitalWrite(relay2,LOW); }

else if( val == '3' ) {

digitalWrite(relay3,LOW); }

else if( val == '4' ) {

digitalWrite(relay4,LOW); }
```

The final source code may look something like this. The logic sequence of the code is visually represented in a flowchart for better understanding:

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(10, 11);
#define relay1 2
#define relay2 3
```

```
#define relay3 4
#define relay4 5
char val;
void setup() {
pinMode(relay1,OUTPUT);
pinMode(// Define the relayIn1 pin as an output relay2,OUTPUT);
pinMode(relay3,OUTPUT);
pinMode(relay4,OUTPUT);
digitalWrite(relay1,HIGH);
digitalWrite(relay2,HIGH);
digitalWrite(relay3,HIGH);
digitalWrite(relay4,HIGH);
mySerial.begin(9600);
Serial.begin(9600);
}
void loop() {
//check data serial from Bluetooth android App
if( mySerial.available() >0 ) {
val = mySerial.read();
Serial.println(val);
}
//Relay is on
if( val == '1' ) {
digitalWrite(relay1,LOW); }
else if( val =='2' ) {
digitalWrite(relay2,LOW); }
else if( val == '3' ) {
digitalWrite(relay3,LOW); }
else if( val == '4' ) {
digitalWrite(relay4,LOW); }
//relay-all on
else if( val == '9' ) {
digitalWrite(relay1,LOW);
```

```
digitalWrite(relay2,LOW);
digitalWrite(relay3,LOW);
digitalWrite(relay4,LOW);
}
//relay is off
else if( val == 'A' ) {
digitalWrite(relay1,HIGH); }
else if( val == 'B' ) {
digitalWrite(relay2,HIGH); }
else if( val == 'C' ) {
digitalWrite(relay3,HIGH); }
else if( val == 'D' ) {
digitalWrite(relay4,HIGH); }
//relay-all off
else if( val == 'I' ) {
digitalWrite(relay1,HIGH);
digitalWrite(relay2,HIGH);
digitalWrite(relay3,HIGH);
digitalWrite(relay4,HIGH);}
}
```

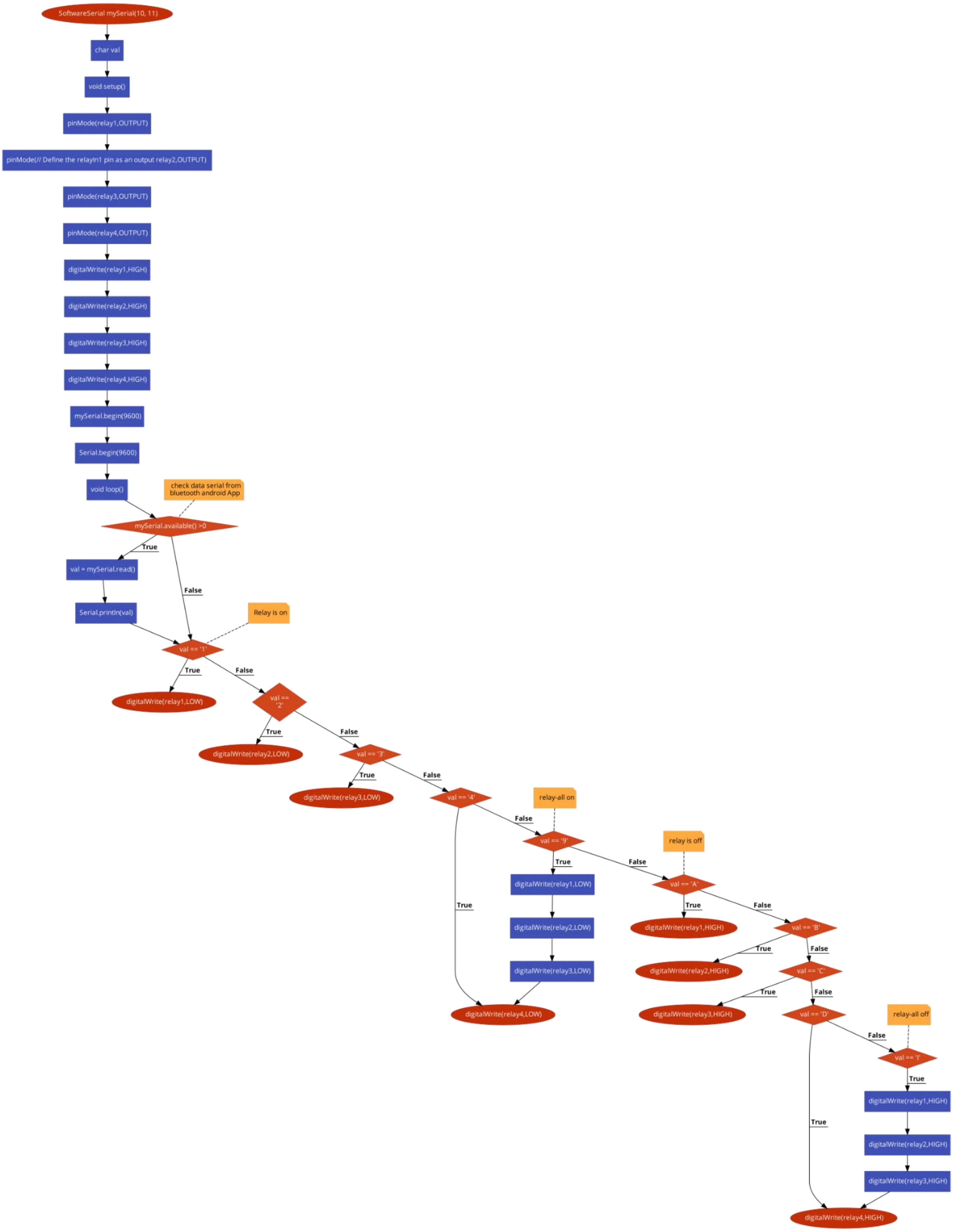

Figure 17 A flowchart representing the logic sequence of the code[21] [https://lh3.googleusercontent.com/FrEno4Mrn-MexLiW9-i-2jojG0Ho5p_55zpVIyHj8qsyuehpu8yx8aRcbbAkDB75wLIGCg=s8]

The application used to control the devices can be made via the MIT app inventor which is a very easy way of creating a prototype application or using a 3rd party application from the app store. After testing it out with a few apps, I have preferred to pick an application from the app store that supports my prototype.[22]

3.2 Testing and Experimenting

After assembling all the electrical components in the circuit with the Arduino as the center piece controlling them, we have achieved the goal of controlling the device using just our fingertips. We can switch the LED lights on/off using the application on our mobile phone which we have connected to the Bluetooth module by pairing them with each other.

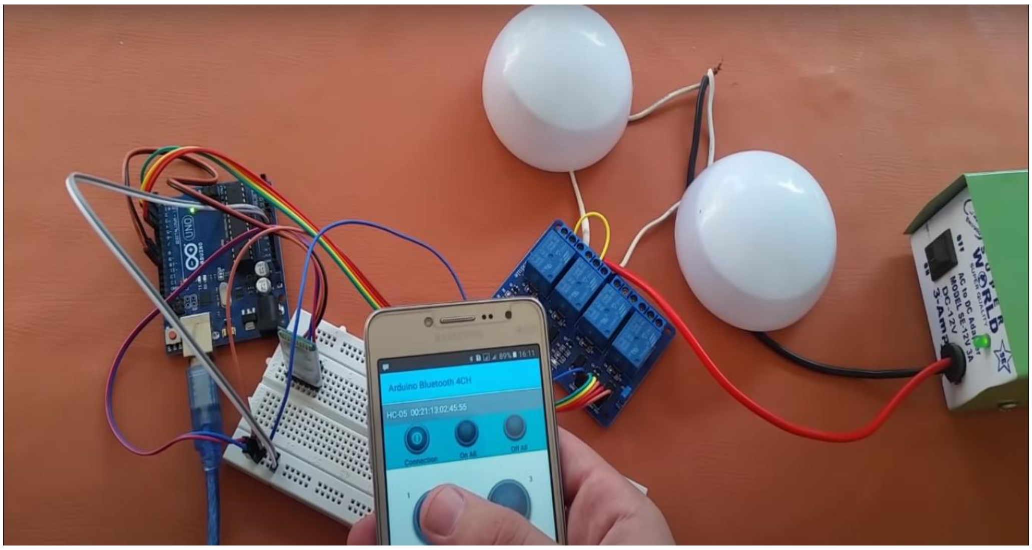

Figure 18 A representation of the complete circuit with both the LED's switched off [https://lh3.googleusercontent.com/TB2NkjCKQ1iN4hi-JAFHsKTnsMRWK0Finn0V-JMrLW6t_gTJYxxtu5zNcn6vDJFviyki=s170]

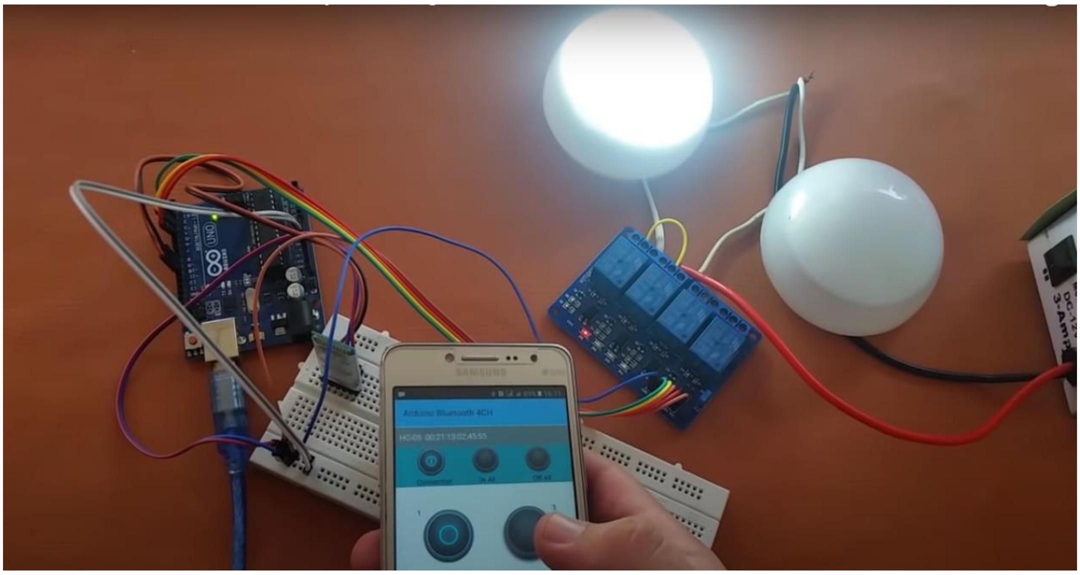

Figure 19 A representation of the complete circuit with one of the LED’s switched on [https://lh3.googleusercontent.com/Cqi59hJgBVrKNgGmw6keumKx806nin403ZyNCKvqtWRhAdvqk8RU7quOM-kdb-F0bE33ew=s165]

Figure 20 A representation of the complete circuit with both the LED’s switched on [https://lh3.googleusercontent.com/Paj7pIFTmruR9kmCkeM2XfEfRfk2aAsSEvPGrvTDujB9qvQNoWWO3NiEZLU76pZwnQI=s166]

Chapter 4

4.1 Conclusion

This thesis has examined the possibility of developing a system using the principle of Internet of Things and the challenges associated with such a project. The initial plan was to build a circuit powered by 220V AC current and represent the actual influence that this system can have in your daily household. Unfortunately, due to a lack of proper equipment, surroundings, and professional assistance, it was just too dangerous to conduct it.

The approach to develop this project was made with the help of an Arduino microcontroller board where all key components for controlling all the electrical components necessary for the result were directly connected to the board in order to create a combined network of inputs and outputs between them. The Bluetooth module was used to control the test device (LED) by connecting your mobile phone using an application. The relay module is one of the most important parts in this project, it is based on the electromagnetic attraction theory. When the relay circuit detects a fault current, it activates the electromagnetic field, which generates a brief magnetic field. Therefore, it allows us to control the circuit using the electrical components attached to the circuit. A code is developed and uploaded into the board which makes sense of these inputs and then runs them in a loop containing conditions for the behavior of the components. Depending on the values of the inputs(i.e. whether you want to switch on or switch off the device) the code checks which conditions are satisfied and then proceeds to send signals to the components. The behavior of the components then reflects if the device is on/off.

The concept of the system was shown by controlling a small device such as an LED since the circuit was powered by 12V DC current with the touch of your finger sitting miles away from the actual switch on the wall. The real possibilities are endless which I have briefly explained in the beginning of this project. Although there are many positive sides to this concept, I believe the negative sides could be dangerous if we human beings start misusing it by letting the machines control us completely in every aspects of our life. Therefore, we must have strong minds and always be the one in charge of what's happening around us even if we are adjusting to the non-conventional ways of life.[25]

4.1 Future Work and Propositions

Despite the overall success of this project, each part of the system could be improved. As it stands, the system represents the bare minimum functionality expected of a home automation system (in our case) or using the principle of Internet of Things in an application. One improvement can be done by using 220V AC current and controlling the major devices in your house like the TV, Air conditioner or the Refrigerator .However such a task requires Professional help with the right surroundings and conditions. This is fundamentally important if a functional home automation system is to be produced for the consumers. There are quite a few out there already but most of them are integrated with a device and does not allow you to have the freedom to tweak it much unless you can afford the expensive ones. I believe the principle of Internet of things will play a major role in the sustainable development industry whether it be in the form of reducing energy consumptions or reduce the amount of labor , time, and capital.

I would further like to study and research the ways of developing a sustainable urban system controlled and monitored by IOT in the future since the current state of this world is not heading towards the best direction. I believe it is our duty to keep the world as the way that was given by our ancestors since we are borrowing it from our future children/generation. It is our duty as fellow human beings to connect with each other and make changes for the better in whatever ways possible for us since even the smallest of efforts could lead to a major change.[27]

Thank you for your time and patience.

:)

References

[1] " A Brief History of Smart Home Automation [online]. [cit. 2021- 07-16]. Available from: https://zeusintegrated.com/blog/item/a-brief-history-of-smart-home-automation," [Online].

[2] "What is the IoT? Everything you need to know about the Internet [online]. 2018 [cit. 2021-07-16]. Available from: https://www.zdnet.com/article/what-is-the-internet-of-things-everything-you-need-to-know-about-the-iot-right-now/," [Online].

[3] "The 9 most important applications of the Internet of Things (IoT) [online]. [cit. 2021-07-16]. Available from: https://www.fracttal.com/en/blog/the-9-most-important-applications-of-the-internet-of-things [Online].

[4] "Advantages and Disadvantages of (IoT) - Javatpoint [online]. [cit. 2021-07-16]. Available from: https://www.javatpoint.com/iot-advantage-and-disadvantage," [Online].

[5] "IoT for Sustainable Development Project [online]. [cit. 2019-03-16]. Available from: https://widgets.weforum.org/iot4d/index.html" [Online].

[6] "Wikimedia Arduino [online]. [cit. 2021-07-16].Available from: https://en.wikipedia.org/wiki/Arduino," [Online].

[7] "Arduino Uno. Components101. (n.d.). [online]. [cit. 2021-07-20]. Available from: https://components101.com/microcontrollers/arduino-uno." [Online].

[8] "Bluetooth Module HC-05. Components101. (n.d.). [online]. [cit. 2021-07-20]. Available fromhttps://components101.com/wireless/hc-05-bluetooth-module." [Online].

[9] "5V 4-Channel Relay Module. Components101. (n.d.). [online]. [cit. 2021-07-20]. Available fromhttps://components101.com/switches/5v-four-channel-relay-module-pinout-features-applications-working-datasheet" [Online].

[10] "Breadboard. Components101. (n.d.). [online]. [cit. 2021-07-20]. Available from:https://components101.com/misc/breadboard-connections-uses-guide." [Online].

[11] "Jumper wires. Components101. (n.d.). [online]. [cit. 2021-07-20]. Available from: https://components101.com/search/node?keys=jumper%20wires." [Online].

[12] " How to Make a Schematic Diagram [online]. [cit. 2021-07-20]. Available from: https://www.coreldraw.com/en/tips/design/technical/make-a-schematic-diagram/ [Online].

[13] " AN INTRODUCTION TO ARDUINO UNO PINOUT [online]. [cit. 2021-07-20]. Available from: https://www.circuito.io/blog/arduino-uno-pinout/." [Online].

[14] " Arduino Uno Pins – A Complete Practical Guide [online]. [cit. 2021-07-20]. Available from: https://roboticsbackend.com/arduino-uno-pins-a-complete-practical-guide/." [Online].

[15] " How Bluetooth Module Interfacing with Microcontrollers [online]. [cit. 2021-07-20]. Available from: https://www.edgefxkits.com/blog/interfacing-hc-05-bluetooth-module-with-microcontroller/" [Online].

[16] " Power Relay Modules [online]. [cit. 2021-07-20]. Available from: https://www.geppowerproducts.com/standard-products/power-distribution-fuse-relay-holders-fuse-blocks/relay-modules/" [Online].

[17] " Arduino UNO Reference Design [online]. [cit. 2021-07-20]. Available from: https://www.arduino.cc/en/uploads/Main/arduino-uno-schematic.pdf" [Online].

[18] " Arduino Coding Environment and basic tools [online]. [cit. 2021-07-20]. Available from: https://www.circuito.io/blog/arduino-code/"[Online].

[19] " How to go from coding tutorials to building your own projects [online]. [cit. 2021-07-20]. Available from: https://www.freecodecamp.org/news/how-to-go-from-coding-tutorials-to-building-your-own-projects-b9ab51074980/" [Online].

[20] " Arduino programming for beginners [online]. [cit. 2021-08-04]. Available from: https://www.hackerearth.com/blog/developers/arduino-programming-for-beginners/" [Online].

[21] "Code2Flow , flowchart maker [online]. [cit. 2021-07-20]. Available from: https://app.code2flow.com/manage/flowcharts" [Online].

[22] "Home Automation System Using a Simple Android App [online]. [cit. 2021-07-20]. Available from: https://www.electronicsforu.com/electronics-projects/hardware-diy/arduino-home-automation-system-android." [Online].

[23] " HOME AUTOMATION AND SECURITY SYSTEM [cit. 2021-08-04]. Available from: https://aircconline.com/acii/V3N3/3316acii03.pdf"[Online].

[24] " The future of IoT: 10 predictions about the Internet of Things [online]. [cit. 2021-08-04]. Available from: https://us.norton.com/internetsecurity-iot-5-predictions-for-the-future-of-iot.html [Online].

[25] Future IoT [online]. [cit. 2021-08-04]. Available from: https://www.ericsson.com/en/future-technologies/future-iot [Online].

[26] " The way IoT helps to reach the Sustainable Development Goals [online]. [cit. 2021-08-04]. Available from: https://futurium.ec.europa.eu/en/connect-university/events/way-iot-helps-reach-sustainable-development-goals/" [Online].

[27] " Challenges and Opportunities of the Internet of Things for Global Development to Achieve the United Nations Sustainable Development Goals [cit. 2021-08-04]. Available from: https://www.researchgate.net/publication/339394462_Challenges_and_Opportunities_of_the_Internet_of_Things_for_Global_Development_to_Achieve_the_United_Nations_Sustainable_Development_Goals [Online].