AN ECONOMICAL ASSISTANCE SYSTEM FOR PARALYSED PERSON

by

AMIT KUMAR YADAV (1406831028)

KM. SWATI AGRAWAL (1406831073)

Department of Electronics & Communication Engineering

Meerut Institute of Engineering & Technology

Meerut, U.P. (India)-250005

May, 2018

AN ECONOMICAL ASSISTANCE SYSTEM FOR PARALYSED PERSON

by

AMIT KUMAR YADAV (1406831028)

KM. SWATI AGRAWAL (1406831073)

Department of Electronics & Communication Engineering

Meerut Institute of Engineering & Technology

Meerut, U.P. (India)-250005

May, 2018

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas words have been included, we have adequately cited and referenced the original sources.

We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

NAME	Roll No.	Date	Signature
Amit kumar Yadav	1406831026		
Km. Swati Agarwal	1406831073		

CERTIFICATE

It is certified that the work contained in the project report titled "**AN ECONOMICAL ASSISTANCE SYSTEM FOR PARALYSED PERSON**", by "**Amit Kumar Yadav (1406831026) and Km. Swati Agarwal (1406831073)**", which is submitted in partial fulfilment of the requirement for the award of degree in Bachelor of Technology in Department of Electronics and Communication of Meerut Institute of Engineering & Technology, Meerut affiliated to Dr. A.P.J. Abdul Kalam Technical University, has been carried out under our supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other degree.

(Dr. Amit Kumar Ahuja) **DATE :**
Professor & Head
Department of ECE

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken at B. Tech. Final Year. We owe a special debt of gratitude to **Dr Amit Kumar Ahuja,** Department of Electronics and Communication Engineering, **our junior Md. Dilshad Khan & Amit Kumar Prajapati** from Meerut Institute of Engineering and Technology, Meerut for his constant support and guidance throughout the course of our work. Their sincerity, thoroughness, and perseverance have been a constant source of inspiration for us. It is only their cognizant efforts that our endeavours have seen the light of the day.

We also take the opportunity to acknowledge the contribution of Centre for Practical Learning and Designing (CPLD) for providing us with the resources and place to work on this project. We are thankful to a member of CPLD for being so supportive of the successful completion of this project.

Md.Dilshad Khan 1506831091

Amit Kumar Prajapati 1506810053

This project has been selected for **TI INNOVATION CHALLENGE** in 2017 which is a national level design competition for an innovative idea, organized by My gov., Department of science and technology and IIM Bangalore.

SIGNATURE **DATE**

Amit Kumar Yadav (1406831028)

Km. Swati Agarwal (1406831073)

ABSTRACT

We have developed a wireless, noncontact, unobtrusive, tongue-operated assistive Technology called the assistance system for paralysed person.This assistance system is a non-invasive development that explores how technology can be assistive for individuals with severe disabilities to communicate with their environment. This tongue control system leads to a more self-supportive independent life. Paralysis is a loss of muscle function for one or more muscles. Paralysis can be accompanied by a loss of feeling (sensory loss) in the affected area if there is sensory as well as motor damage.

The project is aimed at developing a device which will help the paralyzed person in communicating with their surroundings such as controlling a wheelchair, light-and-fan, Television, and, different other electronic gadgets etc. The number of the receiver can be increased according to the user need and multiple functions can be performed. This is proposed to be done with the help of tongue-movement, thus, the name of the device 'Tongue Module(TM)'.

The limitation of this device is that it will not work in some cases like damage of Hypoglossal Nerve, Vagal Nerve, and Glossopharyngeal Nerve. However, the occurrences of these problems are approximately 10%.
Thus, the proposed device shall be very useful for about 90% of the patients who are suffering from paralysis. This project aim is to develop a device that will be a boon for a paralysed person.

TABLE OF CONTENTS

List of Tables

List of Figures

CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION

“BEGIN to see yourself as a soul with a body rather than a body with a soul.” This small but meaningful quote said by WAYNE DYER highlights the point that although the body is the entire structure of a human organism, yet it is controlled and prohibited by a soul. When a person loses the ability to move this human formation, he is not losing everything because our soul, which is still inside, is the main thing that makes us continue our lives. A human body is a machine for living. It is the main structure that helps us move, play, work and does everything. If a person suffers the loss of the ability to move this body or becomes paralyzed, he thinks that life has stopped and lives in a very depressed and gloomy environment.

Paralysis is a loss of muscle function for one or more muscles. Paralysis can be accompanied by a loss of feeling (sensory loss) in the affected area if there is sensory damage as well as motor. About 1 in 50 people in the U.S. have been diagnosed with some form of paralysis, transient or permanent. Paralysis is most often caused by damage to the nervous system, especially the spinal cord. Other major causes are a stroke, trauma with nerve injury, poliomyelitis, cerebral palsy, peripheral neuropathy, ALS, Botulism, multiple Sclerosis .

When it comes to offering assistive help to disabled people, scientists are making new inventions. Even though it would take another decade or so for the technology to be complete, but scientists are making progress in using brain implants. Brain implants are being used for regaining the freedom of movement that any injury to spinal cord takes away. But what if we are able to make some device without any human brain involving. This project is basically aimed at developing low cost,efficient, robust and user-friendly device.

Our project is tongue-operated assistive technology which can potentially provide people with several disabilities with effective access and surrounding environmental control. It

translates users intention into control commands by detecting and classifying their voluntary tongue touched based motion.

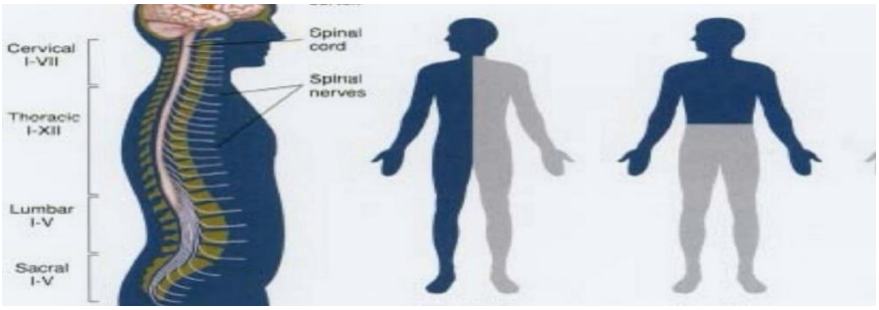


Fig: 1 Types of paralysis

We use tongue to operate the system because unlike the feet and the hands, which are connected by brain through spinal cord ,the tongue and the brain has direct connection through cranial nerve that generally escapes damage in severe spinal cord injuries or neuromuscular disease." tongue movements are fast, accurate and do not require much thinking, concentration or effort.With the motion of tongue ,the user has to press a very smooth switch which will result in digital signal according to his intention. The main aim of this project is to make paralyzed person self-sufficient to change intention into the digital output and do his/her own work. This whole device will be compact and power efficient and user-friendly and can be easily kept in the mouth. In the current scenario, research work is going with the help of hall effect sensor, pressure sensor but a final product is not available in the market. This project may be a boon in the lifestyle of the paralysed person .Our main aim is to develop the prototype for this project.

Fig : 2 Mouth

1.1 Why tongue controlled ?

- Tongue and mouth occupies the major part
- Muscles fibres in the tongueare similar to heart muscles fibres
- The low rate of perceived exertion
- Directly connected to the brain
- Hidden inside the mouth will give a certain degree of privacy

1.2 Scope and objective

Our objective and aim is to serve the humanity that every person have the right to live their life happily and independently. So we are making a tongue drive control or system for the Paralysed person or disabled person to control their environment easily like wheelchair. The disabled person and the person which consist of the spinal cord disease have the difficulty to perform daily life activity . The main scope of this project is to provide a assistive support to the paralyzed person. This may become a most useful device for making them a self-sufficient, This device will be different from many other products as it is a user-friendly and low-costdevice.

1.3 Objective of project

Here we are going to use a very basic electronics concept of simple electrical contact that triggers a circuit. For receiving an input from the user we are providing a push button of size around 2 mm inside the mouth which will be controlled only by Tongue pressure. Hence, the user will be able to give the desired command by moving his or her Tongue on a particular button.

The final product will have a voice module present that will guide the user in accessing the various predefined available command and present status of operations. Once the user confirms the input, the signal will pass to a microcontroller via an Xbee Pair which can communicate amongst each other and which will Process the signal according to a specific algorithm and hence drive the required electromechanical system.

This system will/may contain:

1) Wheelchair

2) Room Lighting System

3) Fan/Temperature Controlling System

4) User Specific Demand

5) Emergency Alert Sound System

At present, there is no such reliable and efficient system in the market. Few systems are being tested which are based on Hall Effect Sensors, Air Pressure and Movement of Face Muscle but their reliability is very low and is using quite sophisticated circuitry.

1.3 Market Analysis

1.3.1 Customer Need Identification –

Here is some statistical information on paralysis from around the world.

- 42.1 percent of paralysis cases happen due to injuries sustained in a motor vehicle accident.
- 26.7 percent of the paralysis cases happen as a result of a fall
- Almost 7.5 percent of cases of paralysis happen due to injuries sustained while playing some type of sport
- About 15.1 percent of paralysis cases happen as a result of some of the other type of injury sustained in a violent interaction.
- We went through the data on Christopher Reeve Foundations and also consulted local doctors and patients and they supported our idea of the device and gave various relevant details about the intricacies that will occur.

1.3.2 Market (SAM) Identification & Justification –

The following are statistics from various sources about hospitalizations and Paralysis:

- 0.14% (18,478) of hospital episodes were for cerebral palsy and other paralytic syndromes in England 2002-03 (Hospital Episode Statistics, Department of Health ,England, 2002-03)
- 77% of hospital consultations for cerebral palsy and other paralytic syndromes required hospital admission in England 2002-03 (Hospital Episode Statistics Department of Health, England, 2002-03)
- 52% of hospital episodes for cerebral palsy and other paralytic syndromes were for men in England 2002-03 (Hospital Episode Statistics, Department of Health, England, 2002-03)
- 48% of hospital episodes for cerebral palsy and other paralytic syndromes were for women in England 2002-03 (Hospital Episode Statistics, Department of Health, England, 2002-03).
- 40% of hospital admissions for cerebral palsy and other paralytic syndromes required emergency hospital admission in England 2002-03 (Hospital Episode Statistics, Department of Health, England, 2002-03) So, going by the figures of the market research, market for the product will be almost everywhere in the world as 1 of every 1000 people suffers from some symptoms of paralysis somehow or the other.

General Market will be Patients and also other distributors and Doctors through which the product can be made available to the user.

1.3 Product Differentiation w.r.t. Competition

At present, there is no such reliable and efficient system in the market. Few systems are being tested which are based on Hall Effect Sensors, Air Pressure and Movement of Face Muscle but their reliability is very low and is using quite sophisticated circuitry.

Thus this is a completely different and new product from what is already available in the market. Already existing products are only able to relieve the user from pain but do not provide them with a sense of freedom and independence. So in comparision with the existing product, our product is –

a.) Cost efficient

b.) Power efficient

c.) Reliable

d.) Light and compact

1.4 The Prototype

This prototype is made from scrap and we wanted to make it as cheap as possible. The total cost of this prototype is only Rs 3400/- only whereas we can see a lot of money is spent by many research organisation. This is the simplest prototype that can be made from scrap .we have collected all the necessary data so that its final product can be manufactured and released into the market .we have consulted many doctors and we have talked to manypeople suffering from paralysis about whether they are comfortable about the virtual product.

The first idea about the project come in my mind when I saw a person who is continuously helping a paralysed person. So I decided to make a device that will make paralysed person self-sufficient, and the journey of this project begins. It takes me around 6 months to find the substitute of technology used previously. I have tried many resistive wire,carbon strip potentiometer and then finally electronic switch. The final device sample will cost around 60k and it will be used by any person by mounting it into the mouth. The design has unique things which will be discussed further in the technical section. An India based industry can reduce the cost to less than half of cost given by the big computer.

CHAPTER-2

TECHNICAL BACKGROUND

2.1. Project model

This assistant system is a simple and efficient design which will bring a new way of experiencing life for the paralysed person. The whole prototype consist of the following 3 major sections namely:

1. Hardware (tongue module and trigger point)

2. Controller

3. Host (Software)

The controllers which in our case is AVR micro-controller, AT-mega 328P loaded with Arduino Uno bootloader. The controller is connected to the receiverxbee and it is attached to relay with signal wire. The controller acts according to the receiver program (code) burnt in it and gives the command to the relay attached to the different appliances. Hardware section contains tongue module, relay and 12V dc supply.

Software used in this project is Arduino IDE compatible with the Arduino Uno. The Arduino project provides the Arduino integrated development environment (IDE), which is a cross-platform application written in the programming language Java. It originated from the IDE for the languages *Processing* and *Wiring*. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple *one-click* mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus.

A program written with the IDE for Arduino is called a *sketch*. Sketches are saved on the development computer as text files with the file extension *.ino*. Arduino Software (IDE) pre-1.0 saved sketches with the extension *.pde*. The Arduino IDE supports the languages

C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. The user-written code only requires two basic functions, for starting the sketch and the main program loop, that is compiled and linked with a program stub *main()* into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program *avrdude*to convert the executable code into a text file in the hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

2.2 Components

The following components are used in the prototype.

2.2.1. ATmega 328P

The ATmega328 is a single chip micro-controller created by Atmel and belongs to the Mega AVR series.

Fig: 3 Atmega 328P

The Atmel 8-bit AVR RISC-based microcontroller combines 32 KB ISP flash memory with read-while-write capabilities, 1 KB EEPROM, 2 KB SRAM, 23 general purposes I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, 6-channel 10-bit A/D converter (8-channels in TQFP packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes and for more see **Appendix A**. The device operates at 1.8-5.5 volts. The device achieves throughputs approaching 1 MIPS per Mhz.

Today the ATmega328 is commonly used in 12 many projects and autonomous systems where a simple, low-powered, low-cost micro-controller is needed. Perhaps the most common implementation of this chip is on the popular Arduino development platform, namely the Arduino Uno and Arduino Nano models.

Table 2.1 Key Parameters of ATmega 328

PARAMETERS	VALUE
Flash	32 Kbytes
SRAM	2 Kbytes
EEPROM	1 Kbyte
Pin Count	32
Max. Operating Frequency	20 MHz
CPU	8-bit AVR
# of Touch Channels	16
Hardware QTouch Acquisition	No
Max I/O Pins	26
Ext Interrupts	24
USB Interface	No
USB Speed	No

2.2.2 Arduino Nano

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3.x) or ATmega168 (Arduino Nano 2.x). It has more or less the same functionality of the Arduino Duemilanove but in a different package.

Fig: 4 Arduino Nano V3.0

It lacks only a DC power jack and works with a Mini-B USB cable instead of a standard one. The Arduino Nano can be powered via the Mini-B USB connection, 6-20V unregulated external power supply (pin 30), or 5V regulated external power supply (pin 27). The power source is automatically selected to the highest voltage source.

Microcontroller	ATmega328P
Architecture	AVR
Operating Voltage	5 V
Flash Memory	32 KB of which 2 KB used by bootloader
SRAM	2 KB
Clock Speed	16 MHz
Analog I/O Pins	8
EEPROM	1 KB
DC Current per I/O Pins	40 mA (I/O Pins)
Input Voltage	7-12 V
Digital I/O Pins	22
PWM Output	6
Power Consumption	19 Ma
PCB Size	18 x 45 mm
Weight	7 g
Product Code	A000005

Table 2.2 specification of atmega 328P

Each of the 14 digital pins on the Nano can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kohms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip.
- External Interrupts: 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. See the attach Interrupt() function for details.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Nano has 8 analog inputs, each of which provides 10 bits of resolution (i.e. 1024 different values). By default, they measure from ground to 5 volts, though is it possible to change the upper end of their range using the analogReference() function. Analog pins 6 and 7 cannot be used as digital pins. Additionally, some pins have specialized functionality:

- I2C: 4 (SDA) and 5 (SCL). Support I2C (TWI) communication using the Wire library (documentation on the Wiring website).

There are a couple of other pins on the board:

- AREF. The reference voltage for the analog inputs. Used with analogReference().
- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

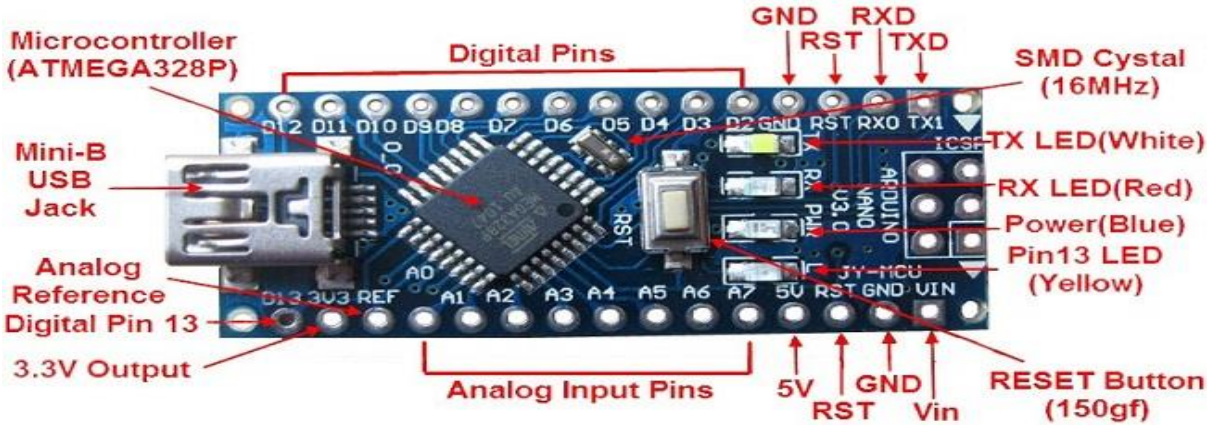


Fig 5: Aurdino nano

2.2.3. Relay Board

A **relay** is an electrically operated switch. Many relays use an electromagnet to mechanically operate a switch, but other operating principles are also used, such as solid-state relays. Relays are used where it is necessary to control a circuit by a separate low-power signal, or where several circuits must be controlled by one signal. The first relays were used in long distance telegraph circuits as amplifiers: they repeated the signal coming in from one circuit and re-transmitted it on another circuit. Relays were used extensively in telephone exchanges and early computers to perform logical .

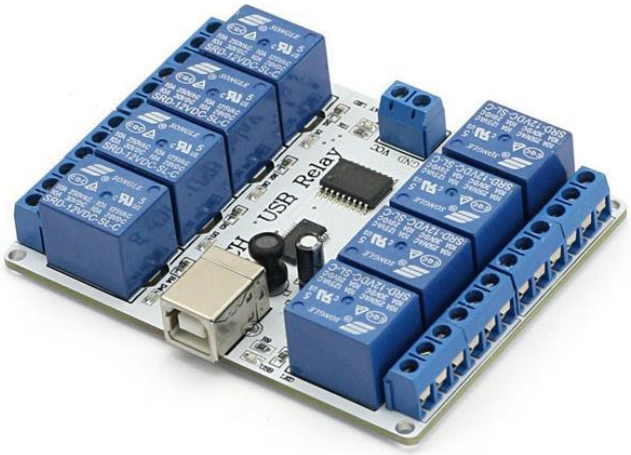


Fig: 6 Relay

A type of relay that can handle the high power required to directly control an electric motor or other loads is called a contactor. Solid-state relays control power circuits with no moving parts, instead of using a semiconductor device to perform switching. Relays with

calibrated operating characteristics and sometimes multiple operating coils are used to protect electrical circuits from overload or faults; in modern electric power systems these functions are performed by digital instruments still called "protective relays". Magnetic latching relays require one pulse of coil power to move their contacts in one direction, and another, redirected pulse to move them back. Repeated pulses from the same input have no effect. Magnetic latching relays are useful in applications where interrupted power should not be able to transition the contacts. Magnetic latching relays can have either single or dual coil. On a single coil device, the relay will operate in one direction when power is applied with one polarity and will reset when the polarity is reversed. On a dual coil device, when a polarized voltage is applied to the reset coil the contacts will transition. AC controlled magnetic latch relays have single coils that employ steering diodes to differentiate between operate and reset commands.

A simple electromagnetic relay consists of a coil of wire wrapped around a soft iron core, an iron yoke which provides a low reluctance path for magnetic flux, a movable iron armature, and one or more sets of contacts (there are two contacts in the relay pictured). The armature is hinged to the yoke and mechanically linked to one or more sets of moving contacts. The armature is held in place by a spring so that when the relay is de-energized there is an air gap in the magnetic circuit. In this condition, one of the two sets of contacts in the relay pictured is closed, and the other set is open. Other relays may have more or fewer sets of contacts depending on their function. The relay in the picture also has a wire connecting the armature to the yoke. This ensures continuity of the circuit between the moving contacts on the armature, and the circuit track on the printed circuit board (PCB) via the yoke, which is soldered to the PCB. When an electric current is passed through the coil it generates a magnetic field that activates the armature, and the consequent movement of the movable contact(s) either makes or breaks (depending upon construction) a connection with a fixed contact. If the set of contacts was closed when the relay was de-energized, then the movement opens the contacts and breaks the connection, and vice versa if the contacts were open. When the current to the coil is switched off, the armature is returned by a force, approximately half as strong as the magnetic force, to its relaxed position. Usually, this force is provided by a spring, but gravity is also used commonly in industrial motor starters. Most relays are manufactured to operate quickly.

In a low-voltage application this reduces noise, in a high voltage or current application, it reduces arcing.

When the coil is energized with direct current, a diode is often placed across the coil to dissipate the energy from the collapsing magnetic field at deactivation, which would otherwise, generate a voltage spike dangerous to semiconductor circuit components. Such diodes were not widely used before the application of transistors as relay drivers, but soon became ubiquitous as early germanium transistors were easily destroyed by this surge. Some automotive relays include a diode inside the relay case. If the relay is driving a large, or especially a reactive load, there may be a similar problem of surge currents around the relay output contacts. In this case, a circuit (a capacitor and resistor in series) across the contacts may absorb the surge. Suitably rated capacitors and the associated resistor are sold as a single packaged component for this commonplace use. If the coil is designed to be energized with alternating current (AC), some method is used to split the flux into two out-of-phase components which add together, increasing the minimum pull on the armature during the AC cycle. Typically this is done with a small copper "shading ring" crimped around a portion of the core that creates the delayed, out-of-phase component, which holds the contacts during the zero crossings of the control voltage.

2.2.4 Switch

An electrical switch is any device used to interrupt the flow of electrons in a circuit. Switches are essentially binary devices: they are either completely on (“closed”) or completely off (“open”). The simplest type of switch is one where two electrical conductors are brought in contact with each other by the motion of an actuating mechanism. Other switches are more complex, containing electronic circuits able to turn on or off depending on some physical stimulus (such as light or magnetic field) sensed. In any case, the final output of any switch will be (at least) a pair of wire-connection terminals that will either be connected together by the switch’s internal contact mechanism (“closed”) or not connected together (“open”).

Any switch designed to be operated by a person is generally called a *hand switch*, and they are manufactured in several varieties:

Toggle switch

Toggle switches are actuated by a lever angled in one of two or more positions. The common light switch used in household wiring is an example of a toggle switch. Most toggle switches will come to rest in any of their lever positions, while others have an internal spring mechanism returning the lever to a certain *normal* position, allowing for what is called “momentary” operation.

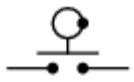
Pushbutton switch

Pushbutton switches are two-position devices actuated with a button that is pressed and released. Most pushbutton switches have an internal spring mechanism returning the button to its “out,” or “unpressed,” position, for momentary operation. Some pushbutton switches will latch alternately on or off with every push of the button. Other pushbutton switches will stay in their “in,” or “pressed,” position until the button is pulled back out. This last type of pushbutton switches usually has a mushroom-shaped button for easy push-pull action.

Selector switch

Selector switches are actuated with a rotary knob or lever of some sort to select one of two or more positions. Like the toggle switch, selector switches can either rest in any of their positions or contain spring-return mechanisms for momentary operation.

Joystick switch



A joystick switch is actuated by a lever free to move in more than one axis of motion. One or more of several switch contact mechanisms are actuated depending on which way the lever is pushed, and sometimes by how *far* it is pushed. The circle-and-dot notation on the switch symbol represents the direction of joystick lever motion required to actuate the contact. Joystick hand switches are commonly used for crane and robot control.

Pressure switch

Gas or liquid pressure can be used to actuate a switch mechanism if that pressure is applied to a piston, diaphragm, or bellows, which converts pressure to mechanical force.

2.2.5 BATTERY POWER SUPPLY

This can be used for power supply in the tongue module and one of the most importantparts due to reliability and long lasting **.**for this purpose we are going to use Batteries which are used in Implantable cardiac pacemakers present unique challenges to their developers and manufacturers in terms of high levels of safety and reliability as in our case. In addition, the batteries must have the longevity to avoid frequent replacements. Micro-electronics advances sharply reduce internal current drain concurrently decreasing the size and increasing functionality, reliability, and longevity. This battery lasts for about 10 years and even today is the power source for many long-lifetime.

Fig: 7 Battery power supply

Chemical Reaction

Conventional current flows through a device from anode to cathode. For a battery, the current flows from the negative anode, through the battery, to the positive cathode. Oxidation of metal occurs at the anode,

$$Li \rightarrow Li^{+} + e^{-}$$

and reduction of halide occurs at the cathode,

$$I_2 + 2e^{-} \rightarrow 2I^{-}$$

. The combined reaction is,

$$2Li + I_2 \rightarrow 2LiI$$

Conventional current flows from anode to cathode. The lithium reacts with iodine to form lithium-iodide, which grows in volume and increases the resistance.

Nuclear batteries were tried successfully for some period. Practical nuclear batteries use plutonium . It has a half-life of 87 years so the output degrades only by 11% in 10 years. How ever, it is highly toxic and 1μg in the bloodstream could be fatal. Early scientist used metallic plutonium whereas later ones used ceramic plutonium oxide. The plutonium emits alpha particles, which impact upon the container and generate heat. Thermopiles of dissimilar p- or n-doped bismuth telluride generate the electricity for the electronic circuits. Though these nuclear power sources had very long life, they were large and created problems when travelling between states and countries due to the presence of their radioactive fuel. They also must be removed at the time of death and

returned for proper disposal. Nuclear powered battery is no longer sold but still a small number of implanted nuclear devices that remain in use. Nuclear power sources became obsolete with the development of lithium batteries.

Fig: 8 Nuclear batteries

2.2.6 XBEE

Wi-Fi, as we know it, is a form of wireless communication based on the 802.11 standards as outlined by the IEEE (the Institute of Electrical and Electronics Engineers). If you're paying attention to it, you'll notice an appendage to this standard that is a single letter suffix (a, b, g, n). These letters correlate to internationally reserved frequency ranges of the radio spectrum for wireless communication, usually in the 2.4GHz and 5GHz frequencies. Phones, tablets, Personal Computers and the rest all rely on this standard of wireless communication to access local area networks and establish a connection to the internet; these devices connect to a network via a wireless hotspot that provides internet access (known as a star network topology).

And it works well; you'd be hard-pressed to find a modern-day home/business without a wireless hotspot today. It's the standard method that comes to mind when you think of Wi-Fi. There are a couple of other options that aren't the generic 802.11 alphabet soup which can be used for wireless transmission of data, Xbee / Zigbee being two of the popular ones. **Digi XBee** is the brand name of a family of form factor compatible radio modules from Digi International. The first XBee radios were introduced under

the **MaxStream** brand in 2005 and were based on the IEEE802.15.4 -2003 standard designed for point-to-point and star communications at over-the-air baud rates of 250 kbit/s. Two models were initially introduced — a lower cost 1 mW **XBee** and the higher power 100 mW **XBee-PRO**. Since the initial introduction, a number of new XBee radios have been introduced and an ecosystem of wireless modules, gateways, adapters and software has evolved.

The XBee radios can all be used with the minimum number of connections — power (3.3 V), ground, data in and data out (UART), with other recommended lines being Reset and Sleep. Additionally, most XBee families have some other flow control, input/output (I/O), an analogue-to-digital converter (A/D) and indicator lines built in. A version called the programmable XBee has an additional onboard processor for user's code. The programmable XBee and a surface-mount version of the XBee radios were both introduced in 2010.

FIG 9: XBEE

XBEE AND ZIGBEE

There are some differences between the similar looking names to be aware of. First off, Xbee refers to a family of devices from Digi that share a form factor, host interface and a group of protocols you can select from (Zigbee being one of these). Zigbee, on the other hand, is a mesh networking protocol built upon the 802.15.4 IEEE standard. So Zigbee protocol dictates how devices can communicate wirelessly and are one of the supported protocols of the Xbee products. Zigbee also has their line of goods too; however, they only support Zigbee communication protocols.

With the naming difference out of the way, let's look at why we would set up a network of XBee modules over a standard Wi-Fi network. Wi-Fi is the standard for wireless Internet connection because of it is high data transfer rate (54MBits/s); the real

downside being it has a high power consumption. If you only wanted to have a collection of micro-controllers attached to a network to share sensor data or something similar, standard Wi-Fi might be too thirsty for your project. Xbee is a lot more reasonable; it's got a much lower power consumption (25% of standard Wi-Fi) and similarly a lower data transfer (250kbit/s). Additionally, the ability to create a mesh network of XBee devices means each device can transmit and receive data through itself, acting as a node for the network. We call this network topology a mesh, and it's an entirely different take on a network system. If you were looking to create your Smart House system, mesh networking could offerless power consumption and greater range than the standard wi-fi setup we typically see.

2.2.7 12 V DC SUPPLY

It is one of the most important parts of this whole project. All the component at receiver side work on 12v dc current and the equipment which is to be controlled worked on 220/60 Hz ac current so first thing is to make a 12 v dc supply and the requiredcomponents and circuit diagram will be as follows.

- PCB Board
- Four 1N4001 diodes
- LM7812 regulator
- A transformer that has an output of 14v - 35v AC with an output current between 100mA to 1A, depending on how much power you will need. (I found a 16v 200mA transformer in a broken alarm clock.)
- 1000uF - 4700uF capacitor
- 1uF capacitor
- Two 100nF capacitors
- Heat sink (optional)

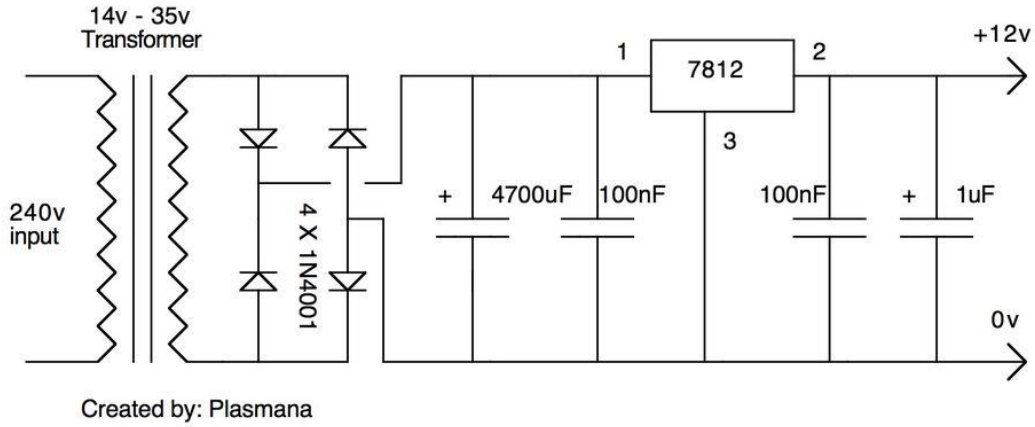


Fig: 10 12 V DC Supply

2.2.8 TRANSMITTER AND RECEIVER

As we are going to use this device in the medical field so most important things were the selection of transceiver which has a low power consumption and compact in size so that it can be used in the TM (Tounge module. The most challenging task was the to decide which wireless protocol and device can be used so that it will give us a good area of range as well as low power consumption and hence has a good efficiency.

We come to know about the ZigBee protocol and its specification as given in the appendix. As the figure is showing the only computer is replaced with a microcontroller, and all the other details are given in the technical section about configuration and other specifications.

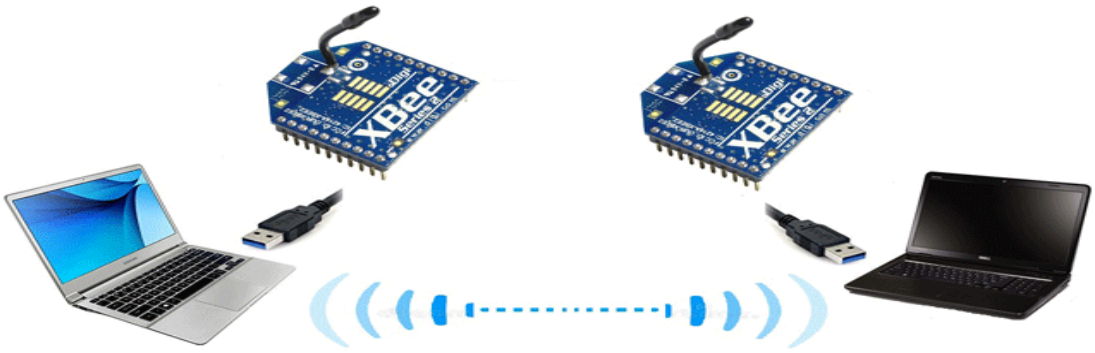


Fig: 11 Transmitter & Reciever

CHAPTER-3

INTERFACING AND CONFIGURATION

3.1 XBEE CONFIGURATION

For configuration of two xbee as transmitter and receiver or coordinator and end device, we need to install XCTU software.

XCTU supports configuration and communication for most Digi RF modules. XCTU uses a serial link to interact with these radio modules, providing an easy-to-use and intuitive graphical interface. The following is a complete list of XCTU-compatible RF modules:

- XBee SX
- XBee-PRO SX
- XBee 802.15.4
- XBee-PRO 802.15.4
- XBee ZB
- XBee-PRO ZB
- Programmable XBee-PRO ZB
- XBee ZB SMT
- XBee-PRO ZB SMT
- Programmable XBee-PRO ZB SMT
- XBee-PRO 900HP
- Programmable XBee-PRO 900HP
- XBee-PRO XSC

- XBee-PRO 900
- XBee-PRO DigiMesh 900
- XBee DigiMesh 2.4
- XBee-PRO DigiMesh 2.4
- XBee-PRO 868
- XBee Wi-Fi
- XBee 865LP
- Programmable XBee 865LP
- XBee Cellular
- XBee 868LP SX
- XBee Thread
- XBee3
- XTend® RF Module family
- XLR PRO radio solution
- XLR Module

3.1.1 Install XCTU - Windows

Follow the steps below to download and install XCTU on your computer.

1. Visit www.digi.com/xctu.
2. Click **Download XCTU**.
3. Under **Utilities**, click the Windows installer link.
4. When the file has finished downloading, run the executable file and follow the steps in the XCTU Setup Wizard.

3.1.2 Main toolbar

The main toolbar is located at the top of the application and is divided into three sections.

- The first section contains two icons used to add radio modules to the radio modules list. See Add radio modules to XCTU.

- The second section contains the static XCTU functionality that does not require a radio module. This section includes the XCTU tools, the XCTU configuration, the feedback form, and the help and updates functions. See XCTU tools and Configure XCTU.

- The third section contains tabs corresponding to the three XCTU working modes. To use this functionality, you must have added one or more radio modules to the list. See XCTU working modes.

3.1.3 Devices list

The radio modules list, or devices list, is located on the left side of the tool and displays the radio modules that are connected to your computer. If you know the serial port configuration of a radio module, you can add it to the list directly. You can also use the discovery feature of XCTU to find radio modules connected to your PC and add them to the list. See Add radio modules to XCTU.

Depending on the protocol of the local radio modules added, you can also add remote radio modules to the list using the module's search feature.

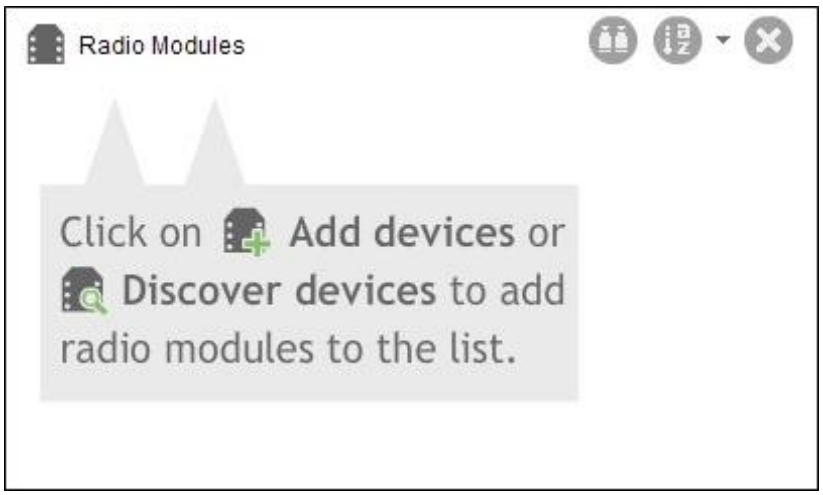


Fig 12 : Searching For Xbee Module

3.1.4 Discover local radio modules

XCTU can discover radio modules that are connected directly to your computer. You can use the discovery tool if you don't know the serial configuration of your radio module, don't know the port it is connected to, or want to add multiple modules at once.

1.) Click the **Discover radio modules** button on the XCTU toolbar. The **Discover modules** dialog box opens.

Fig: 13 pop-up window for xbee configuration

2. Select the serial ports you would like to scan for radio modules. Click **Next**.

3. Select any port parameters you would like to include in the search process.

   Note XCTU displays estimated discovery time in the **Set port parameters** dialog. Adding more port parameters to the search increases discovery time.

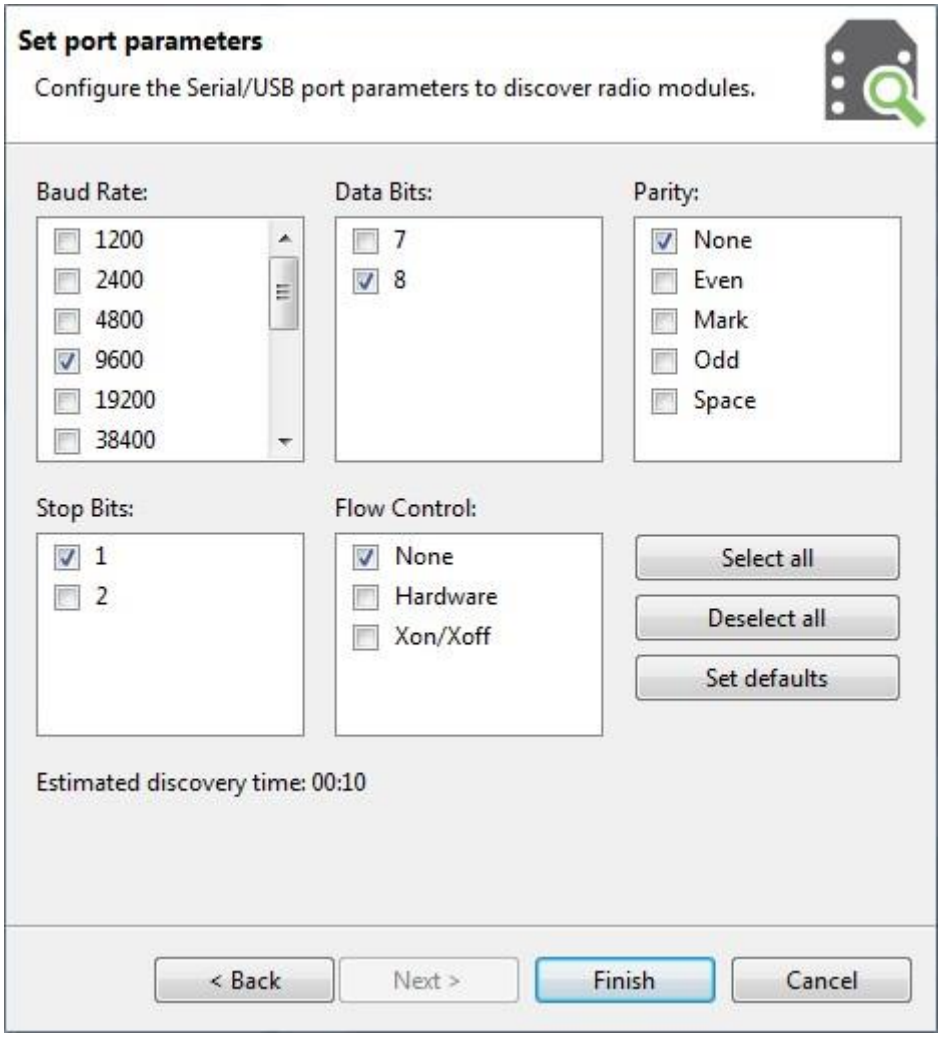


Fig 14: Selecting All The Parameter

4. Click **Finish** to initiate the discovery scan.

   A new dialog opens, displaying devices found and estimated time remaining. You can click **Stop** to halt the discovery process at any time. For example, you can stop the process if the modules you were looking for are already found.

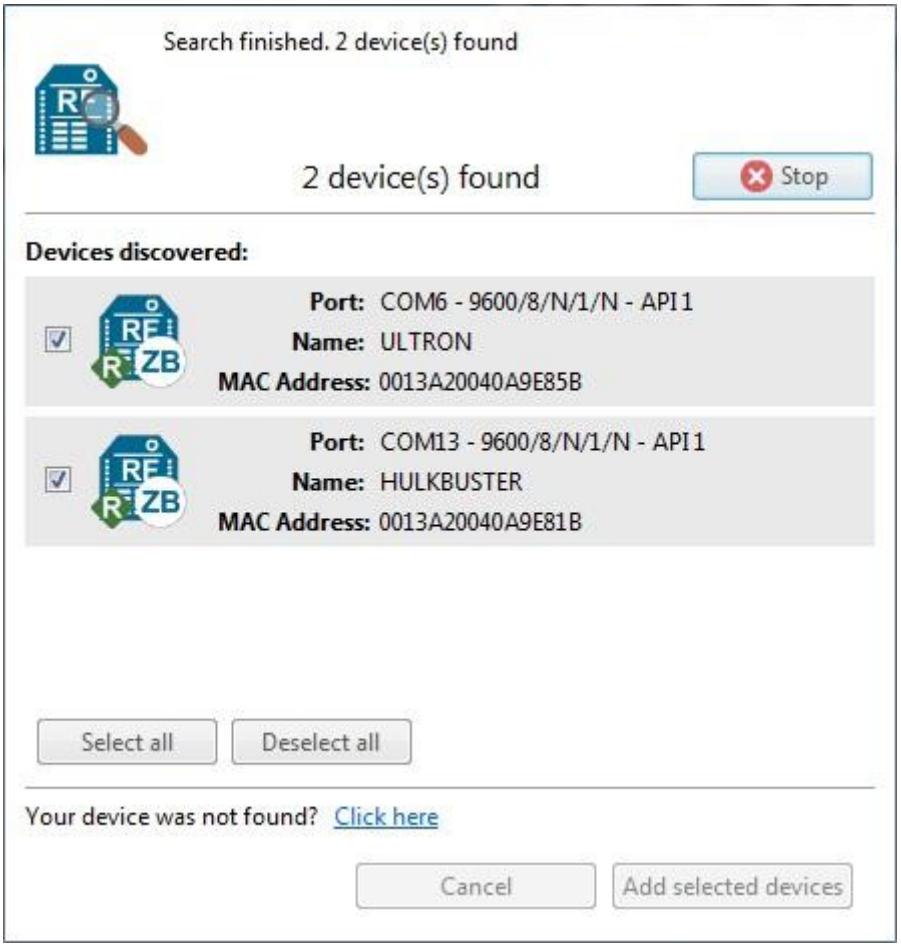


Fig 15:Connected xbee Module

5. Select the box next to the module(s) you want to add to your device list and click **Add selected devices**. The modules appear in the device list.

Fig 16: Searched Xbee Module

Now start configuring xbee as coordinator and router(end device).The router can be multiple but the coordinator is a single device.

3.2 Configure the XBEE as Coordinator

1. Make sure XBee module is placed properly on CP2102 Board

2. Connect USB cable from CP2102 UART Board to PC/Laptop

3. Open X - CTU application

4. **Note:** After Configuring the Zigbee Module as coordinator, Disconnect all the connections which are connected above and close the Application.

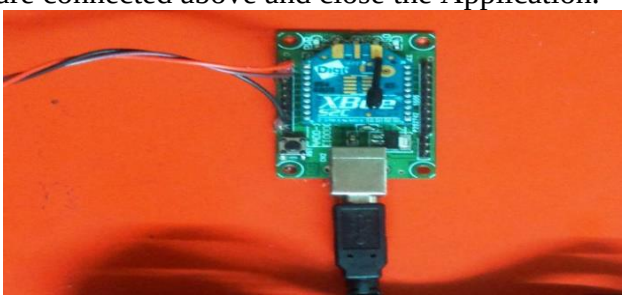


Fig 17: Connected coordinator module

And click on as shown below.

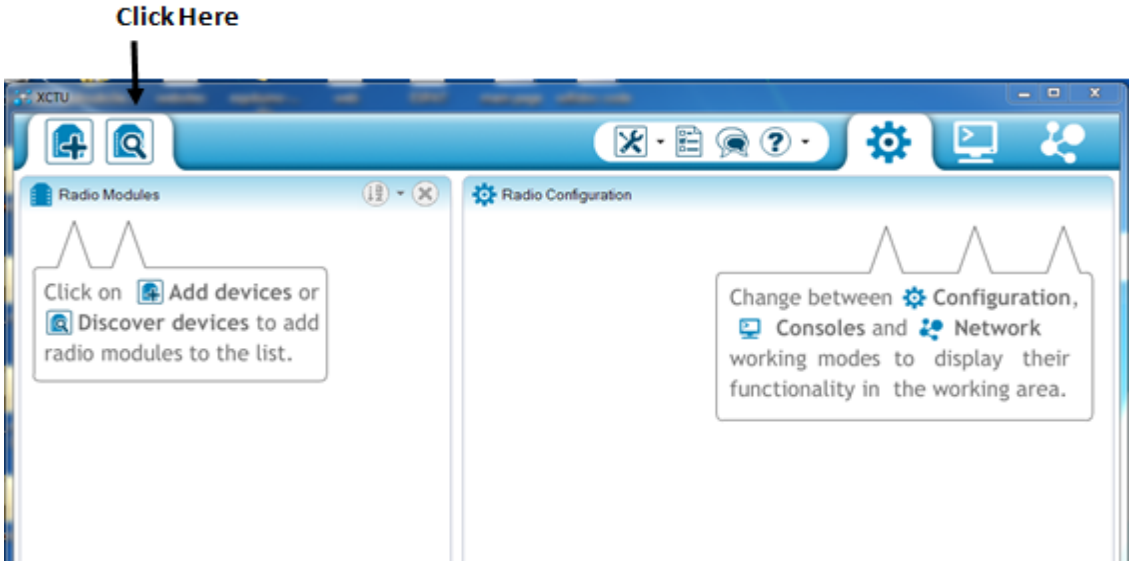


Fig 18: Searched for Device list

Select the **COM Port** (Silicon Labs CP210x USB to UART Bridge) and click on **Next** as show

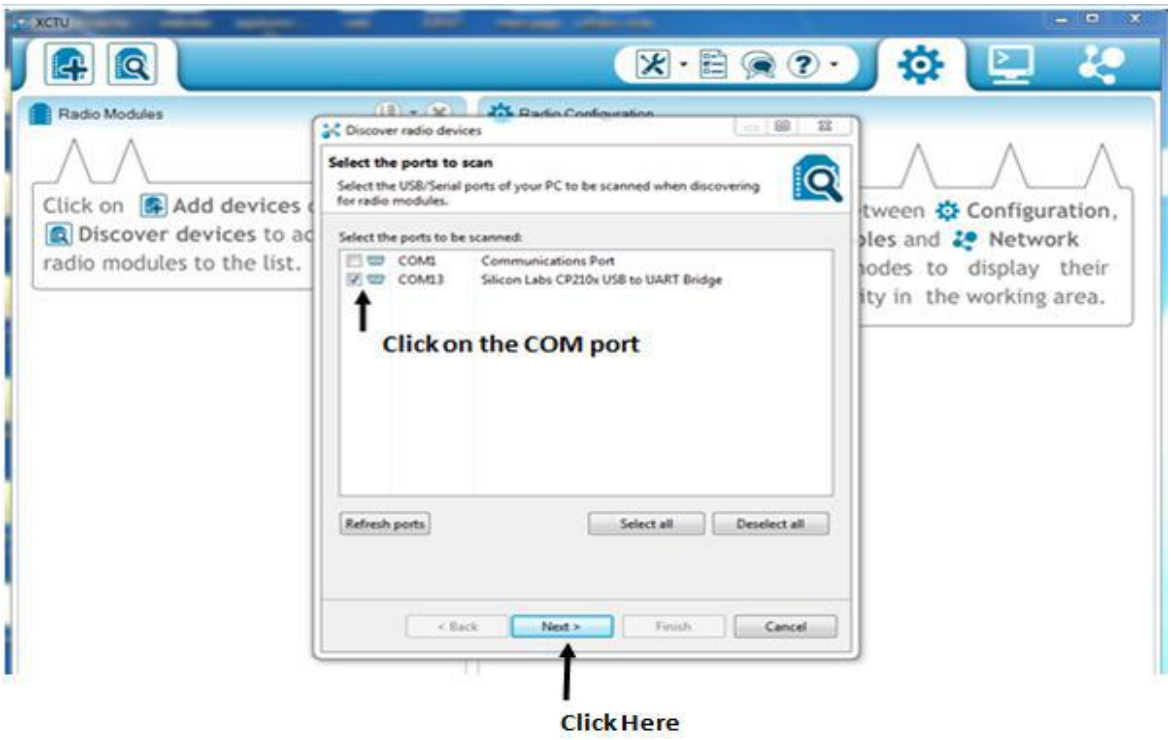


Fig 19: Coordinator COM port Selection

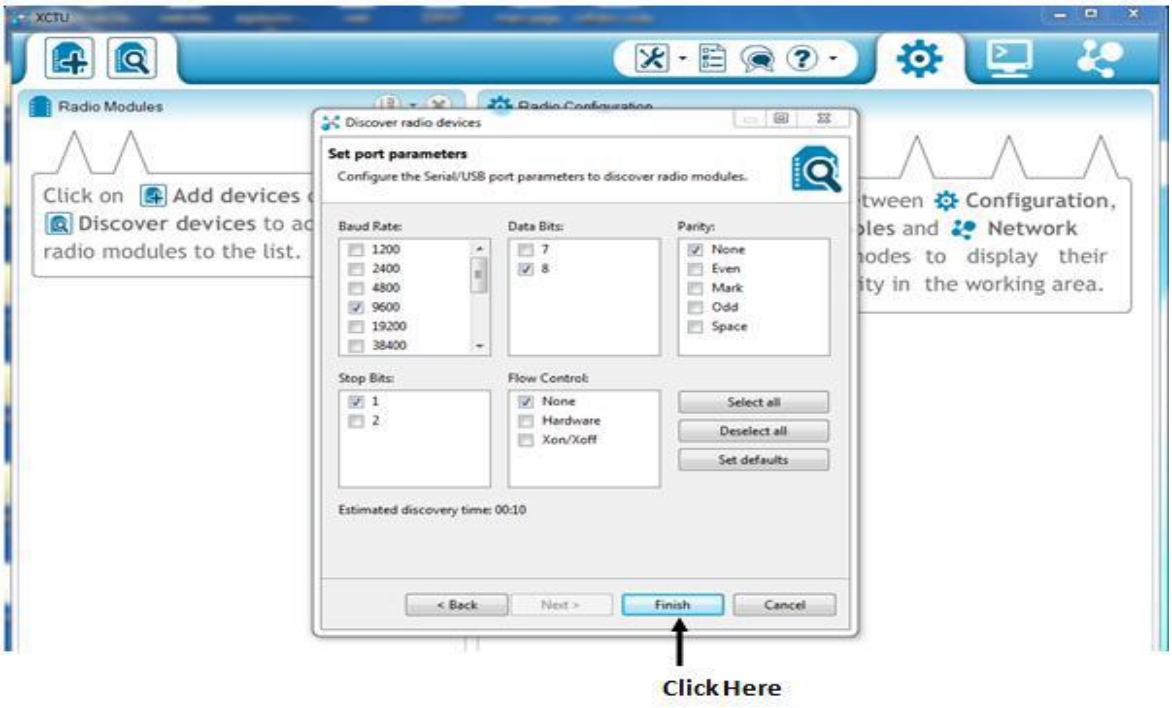


Fig 20: Coordinator Parameter Configuration

Now, click on **Add selected devices** and click **Finish**.

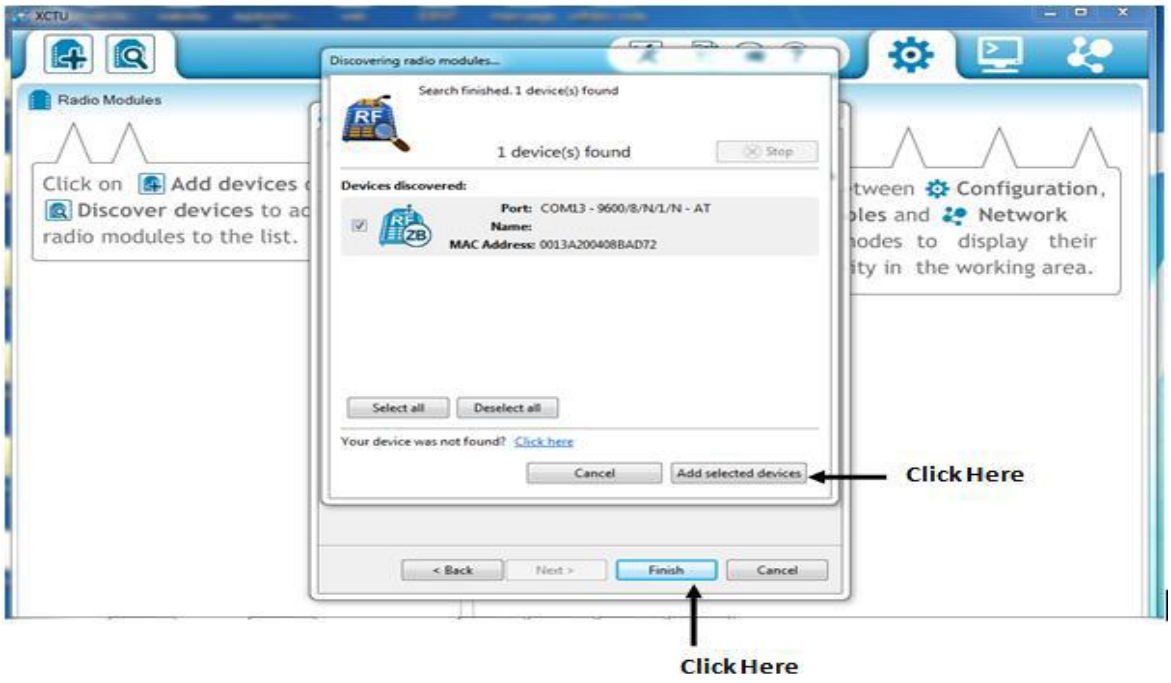


Fig 21: Finishing parameter selection

Fig 22: Selecting coordinator for configurations

Enter the **PAN ID** as your wish, here I have entered **1234**.

Change **CE Coordinator Enable** to – **Enabled [1]**

Click on **Icon** to save values entered**.**

Now, Scroll

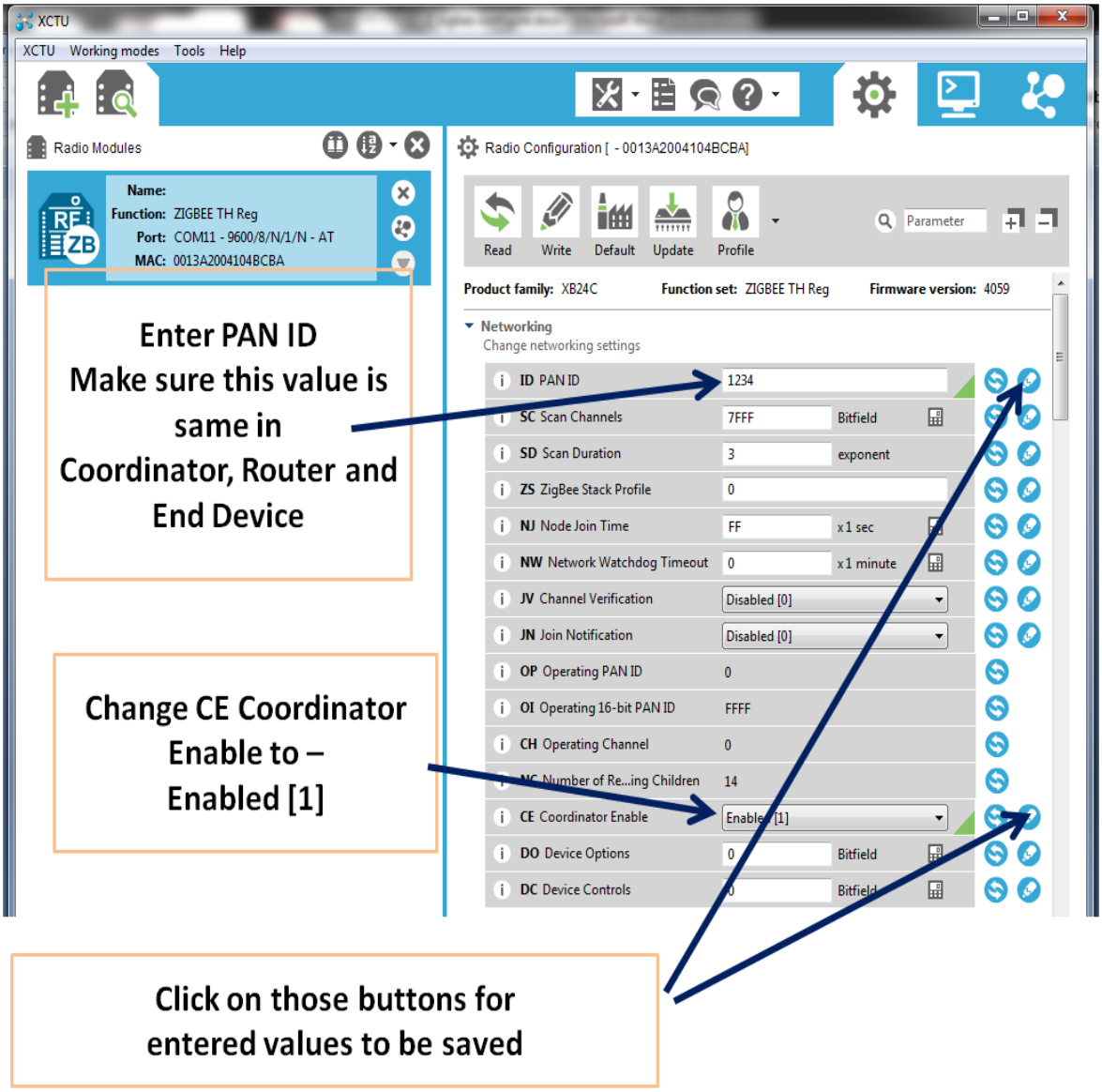


Fig 23: Configuring parameters of coordinator device

Now, Scroll down page.

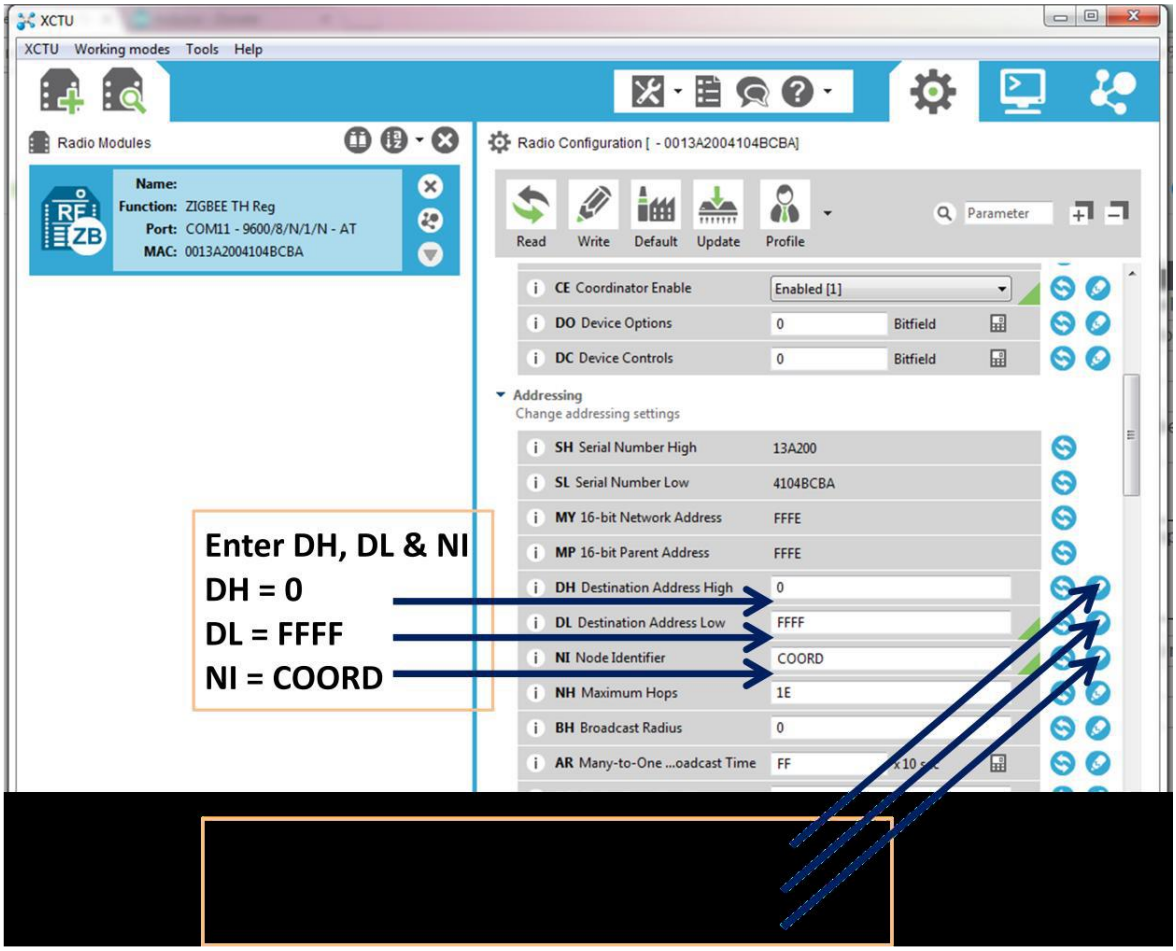


Fig 24: Destination address selection and write

Now, close the application.

3.3. Configure the XBEE as End Device

1. Make sure XBee module is placed properly on CP2102 Board

2. Connect USB cable from CP2102 UART Board to PC/Laptop

3. Open X - CTU application

4. **Note:** After Configuring the Zigbee Module as End Device, Disconnect all the connections which are connected above and close the Application.

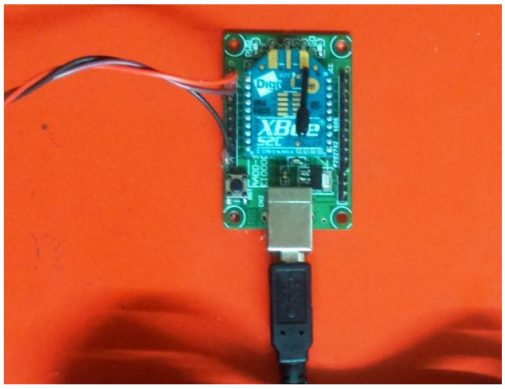


Fig 25: Connecting the End Device with Laptop

And click on as shown below.

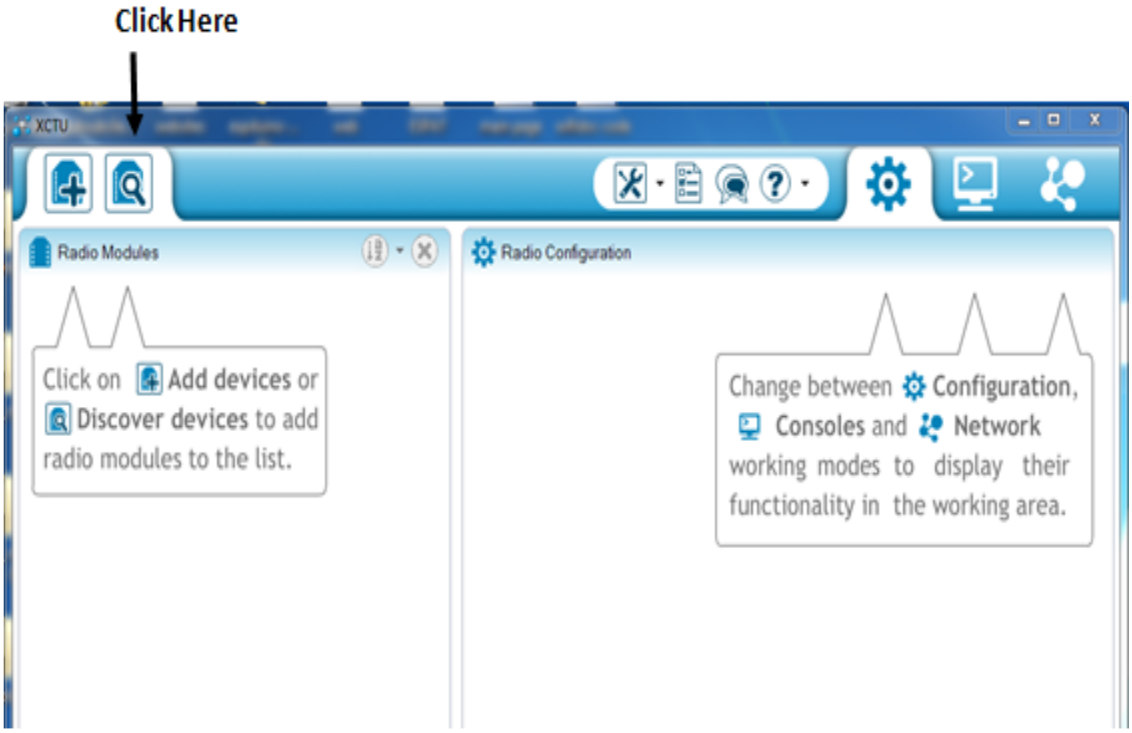


Fig 26: Searched for End device

Select the **COM Port** (Silicon Labs CP210x USB to UART Bridge) and click on **Next** as shown

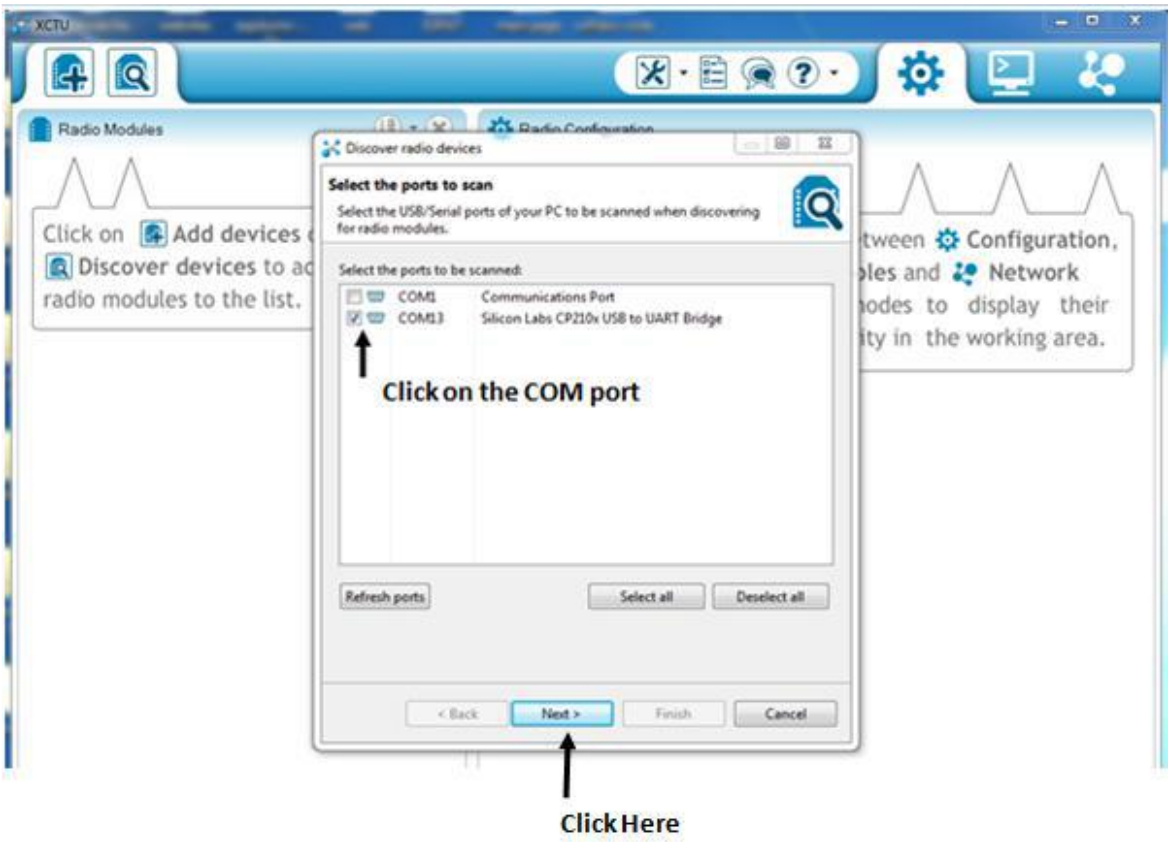


Fig 27: End device COM port selection

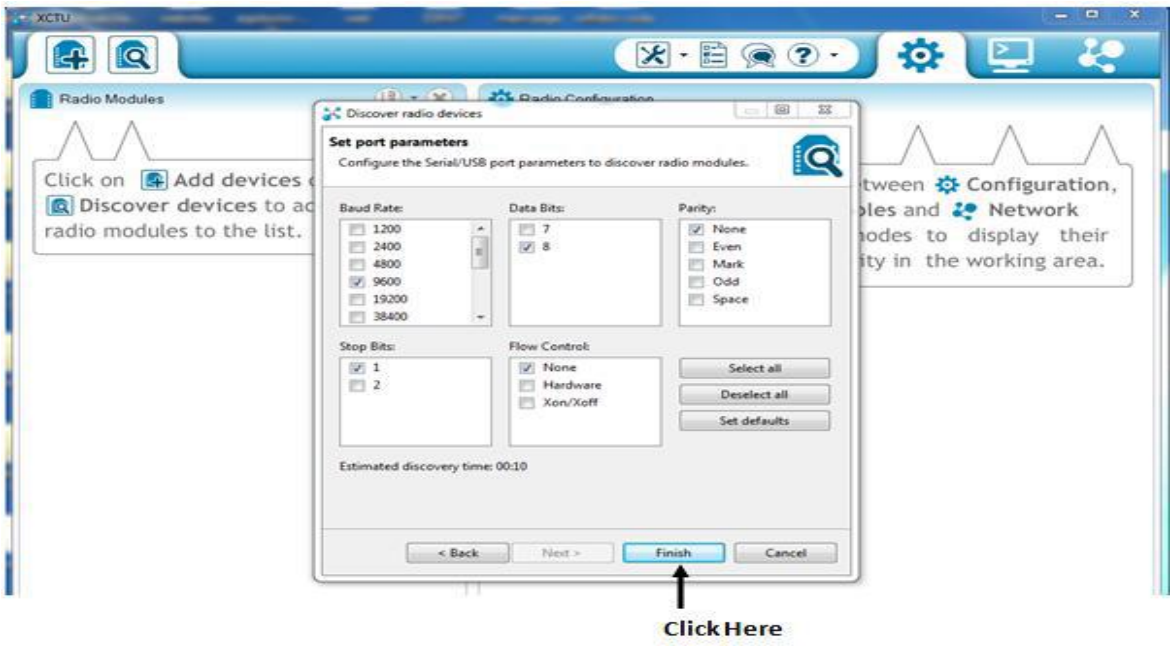


Fig 28: End Device Parameters selection

Now, click on **Add selected devices** and click **Finish**.

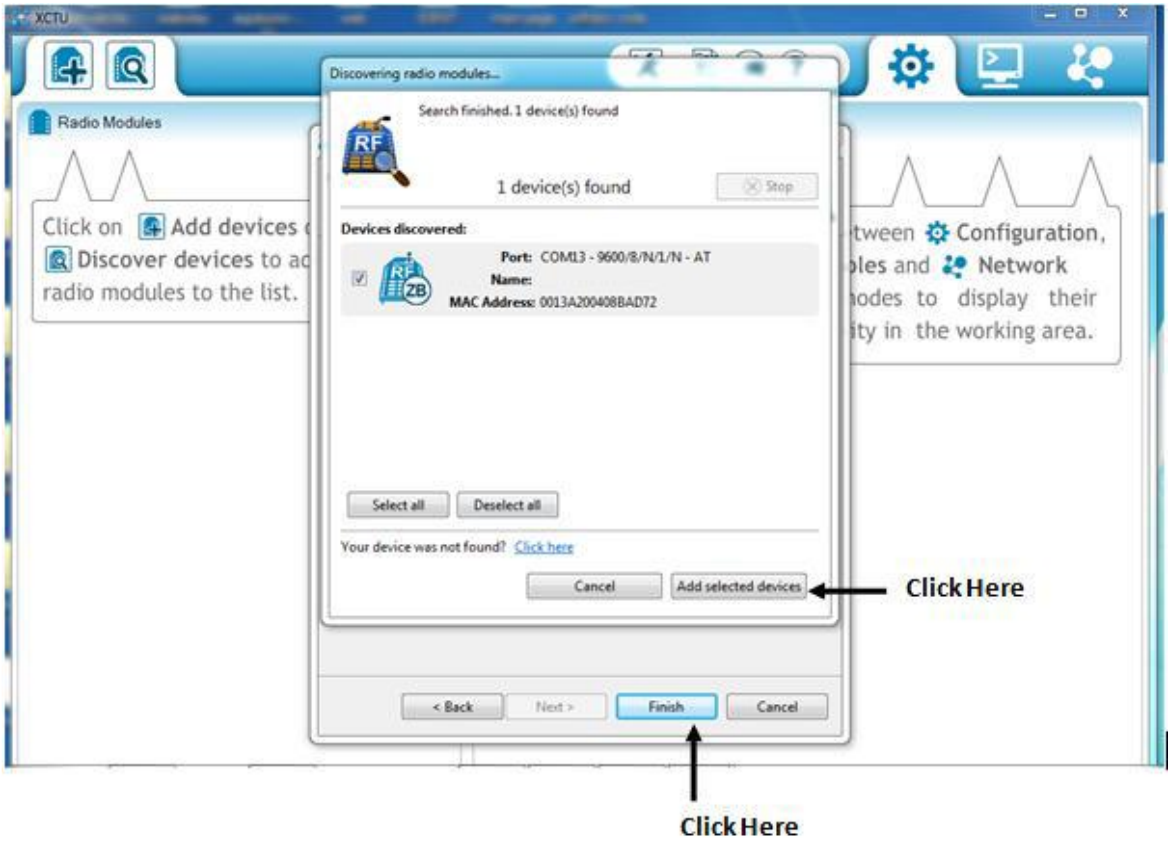


Fig 29: Finishing parameter configuration

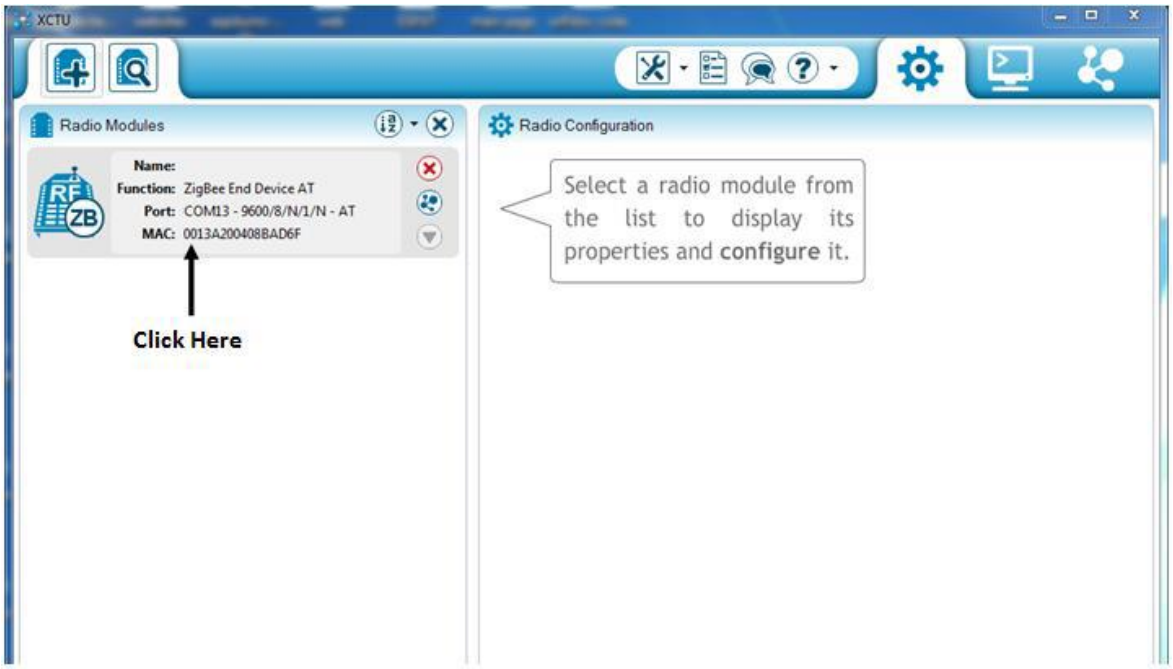


Fig 30: Reading the END device module

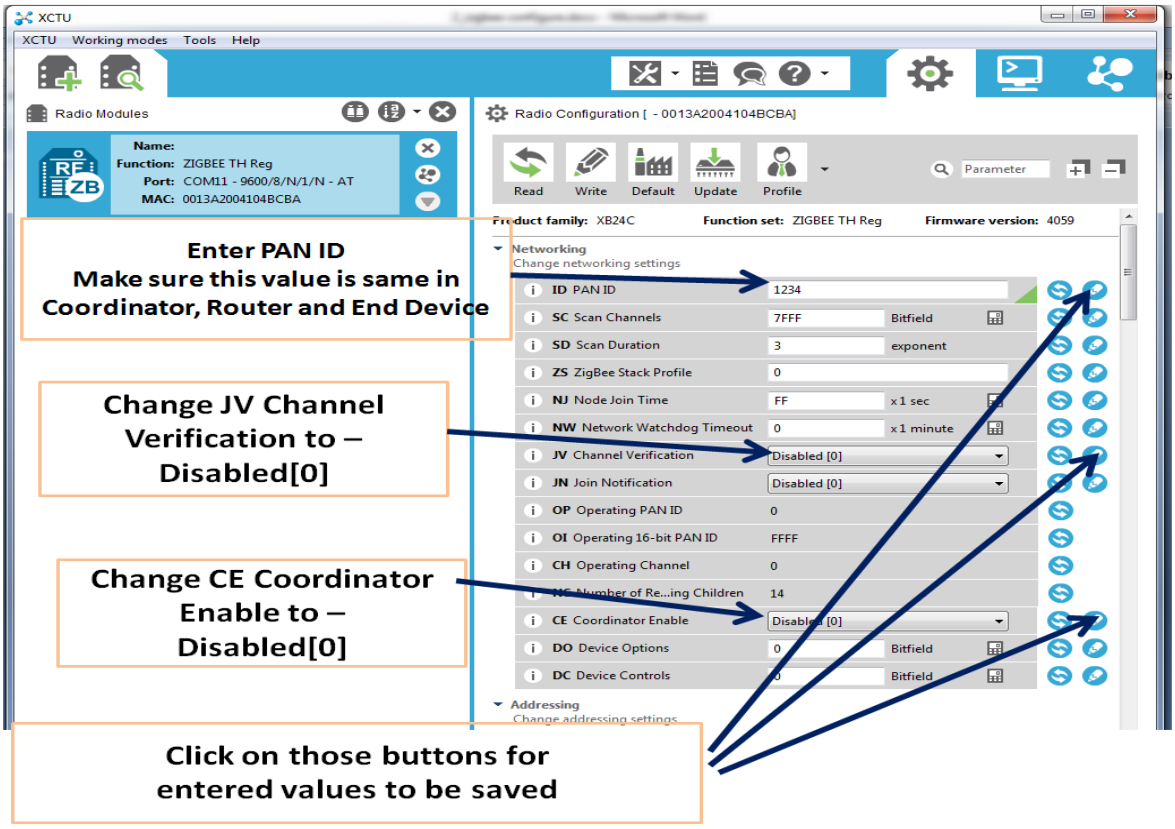


Fig 31: writing all the parameters in END device

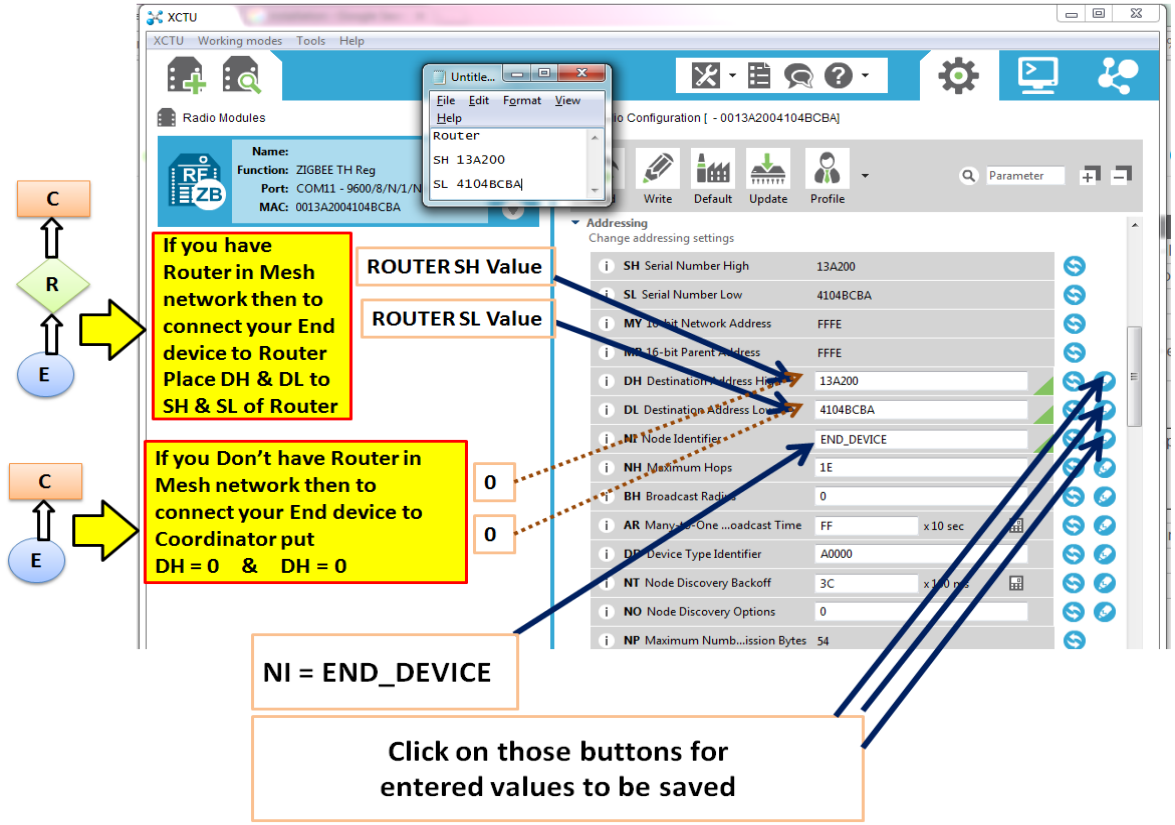


Fig 32: Selecting source address and writing

Now, close the application.

3.4 RELAY CONNECTION

The relay is an electrical switch using an electrical signal to turn on/off the current. It is also used to control a circuit by a low-power signal. On its body, you can see these characters and numbers:

- 10A 250VAC.
- 10A 125VAC.
- 10A 30VDC.
- 10A 28VDC.
- SRD-05VDC-SL-C.

This is what they mean:

- 10A 250VAC: The max current intensity of connection, where the voltage is equal to or less than 250V (AC), is 10A.
- SRD-05VDC-SL-C: Voltage of signal to control circuit is 5V.

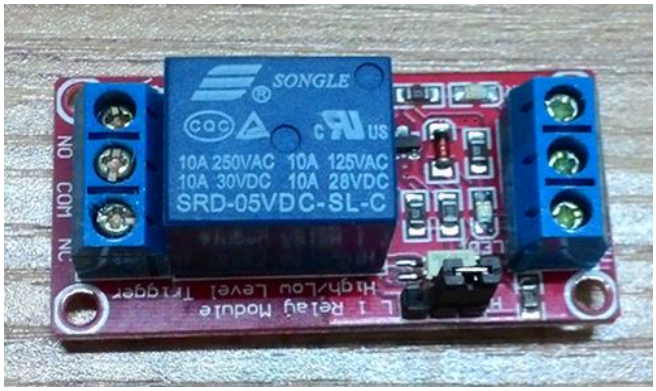


Fig: 33 Single channel Relay

State of relay

It is a switch so it has two main states: close and open.

- **Open state**: While opening, it interrupts circuit and current doesn't work in circuit.
- **Closed state**: While closing, it keeps current in circuit.

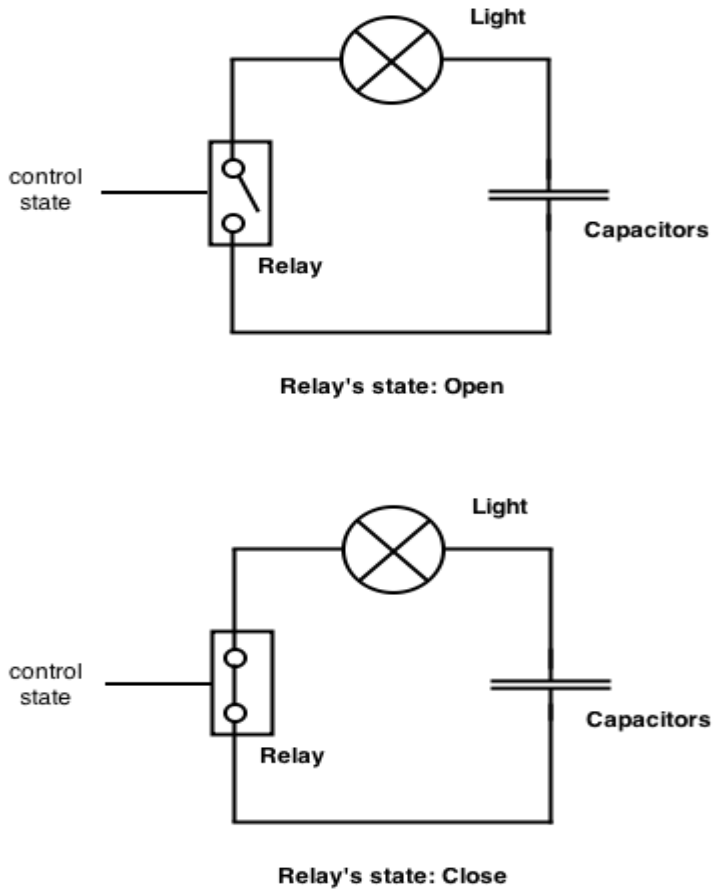


Fig: 34 Relay AC and DC connection

Relay's connections

There are six connections.

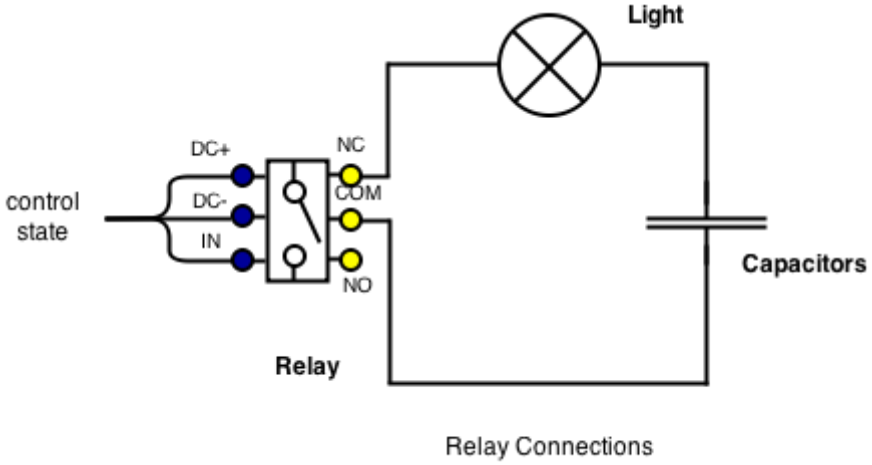


Fig: 35 Relay board connection

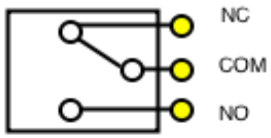
Three connections control the state of relay:

- **DC+**: Electrode positive. In this article, we will connect the wire with 5V to this connection.
- **DC-**: Electrode positive. Connect to GND.
- **IN**: a Signal connection that is used to control relay.

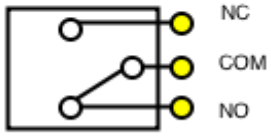
Three connections connect to the circuit:

- **COM (Common Connection)**: Connect to the power supply. If it's DC, we frequently connect it to positive of the power supply. If it's AC, we frequently connect it to hot wire.
- **NC (Normally Close)**: Connect to COM connection when there is no trigger in the relay. If we connect this connection to wire and don't trigger the relay with IN connection, COM and NC will be connected. When we trigger the relay with IN connection, COM and NC will be corrupted.
- **NO (Normally Open)**: Only connect to COM connection when having a trigger in the relay. If we connect this connection to wire and don't trigger the relay with IN connection, COM and NO will be corrupted. When we trigger the relay with IN connection, COM and NC will be connected.

Usually, you will use either NC or NO with COM to control circuit. To understand how NC, NO and COM work, you can see the following pictures:



Relay: Is normal



Relay: Is triggered

3.4.1 Trigger relay

To trigger it (COM connect to NO and disconnect to NC), we use IN connection. There are two kinds of relay using IN connection in different ways. They are "Low Voltage Trigger" and "High Voltage Trigger".

Low voltage trigger

When we connect into GND, the relay is triggered (COM connects to NO and disconnects to NC). It is not triggered when we connect into +5V (COM connects to NC and disconnect to NO).

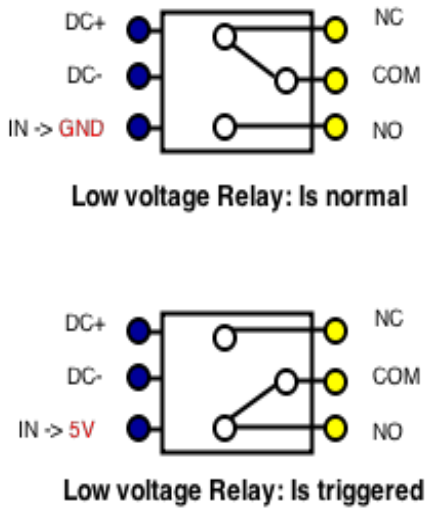


Fig: 36 low voltage trigger

High voltage trigger

The relay is not triggered when we connect into GND (COM connect to NC and disconnect to NO). When we connect into +5V, it is triggered (COM connect to NO and disconnect to NC).

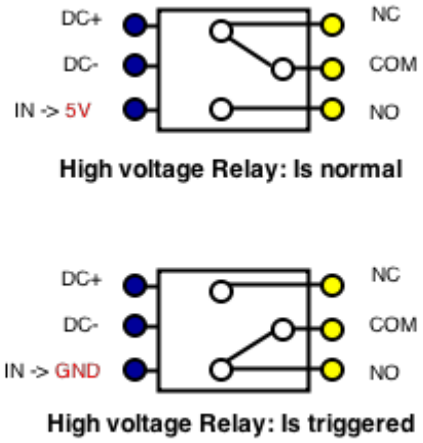


Fig: 37 High voltage trigger

We are using logic low totrigger the relay from receiver microcontroller in accordance with the transmitted signal from tongue module.

3.5 CONNECTION OF SWITCHES

A microcontroller in any embedded system utilizes I/O signals to communicate with the external devices. The simplest form of I/O is usually stated to as GPIO(General Purpose Input/Output). When the GPIO voltage level is low, then it is in high or high impedance state, then the pull up and pull-down resistors are used to ensure GPIO which is always in a valid state. Usually, the GPIO is arranged on a microcontroller as I/O. As an input, the microcontroller pin can take one of these states: high, low and floating or high impedance. When an i/p is driven above the i/p is a high threshold, it is a logic one.When the I/P is driven below the I/P, which is a low threshold, the input is logic 0. When in a floating or high impedance state, the I/P level is not constantly high nor low. To ensure the values of an I/P is always in a known state, pull up and pull-down resistors are used.The main function of pull-up and pull-down resistors is that the pull-up resistor pulls the signal to the high state unless it is driven low; and, a pull-down resistor pulls the signal to the low state unless it is driven high.

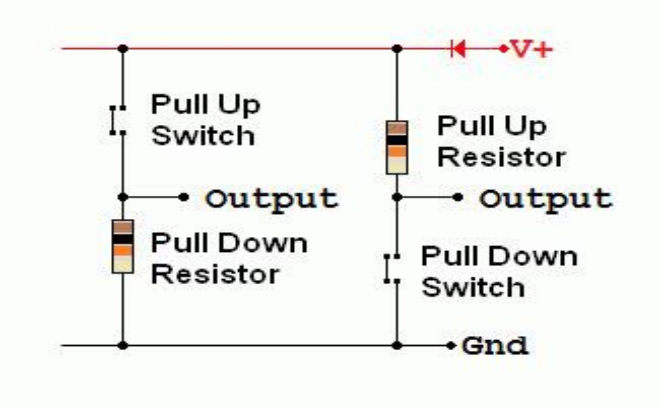


Fig: 38 connections of switches

3.5.1 Pull-up Resistors

Pull-up resistors are simply fixed value resistors,that are connected between the voltage supply and the particular pin. These resistors are used in digital logic circuits to ensure a logic level at a pin, which results in a state wherein the input/output voltage is nonexistence driving signal. Digital logic circuits consist of three states like high, low and floating or high impedance. When the pin is not pulled to a lower or a high logic level, then the high impedance state occurs. These resistors are used to solve the problem for the microcontroller by pulling the value to a high state, as seen in the figure. When the switch is open, the microcontroller's input would be floating and brought down only when the switch is closed. A typical pull-up resistor value is 4.7kilo Ohms but can change depending on the application.

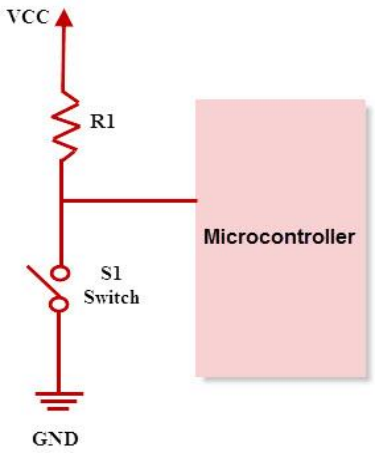


Fig: 39 Pull-up Resistor

3.5.2 Pull-Down Resistors

As pull up resistors, Pull-down resistors also work in the same way. But, they pull the pin to a low value. Pull-down resistors are connected to a particular pin on a microcontroller and the ground terminal. An example of a pull-down resistor is a digital circuit shown in the figure below. A switch is connected between the VCC and the microcontroller pin. When the switch is closed in the circuit, the input of the microcontroller is logic 1,but when the switch is open in a circuit, the pull-downresistor pulls down the input voltage to the ground (logic 0 or logic low value).

The pull-down resistor should have a higher resistance than the impedance of the logic circuit.

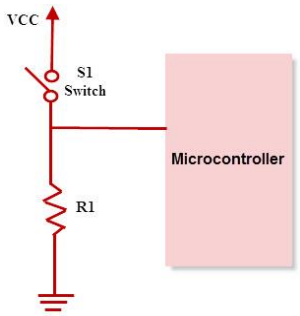
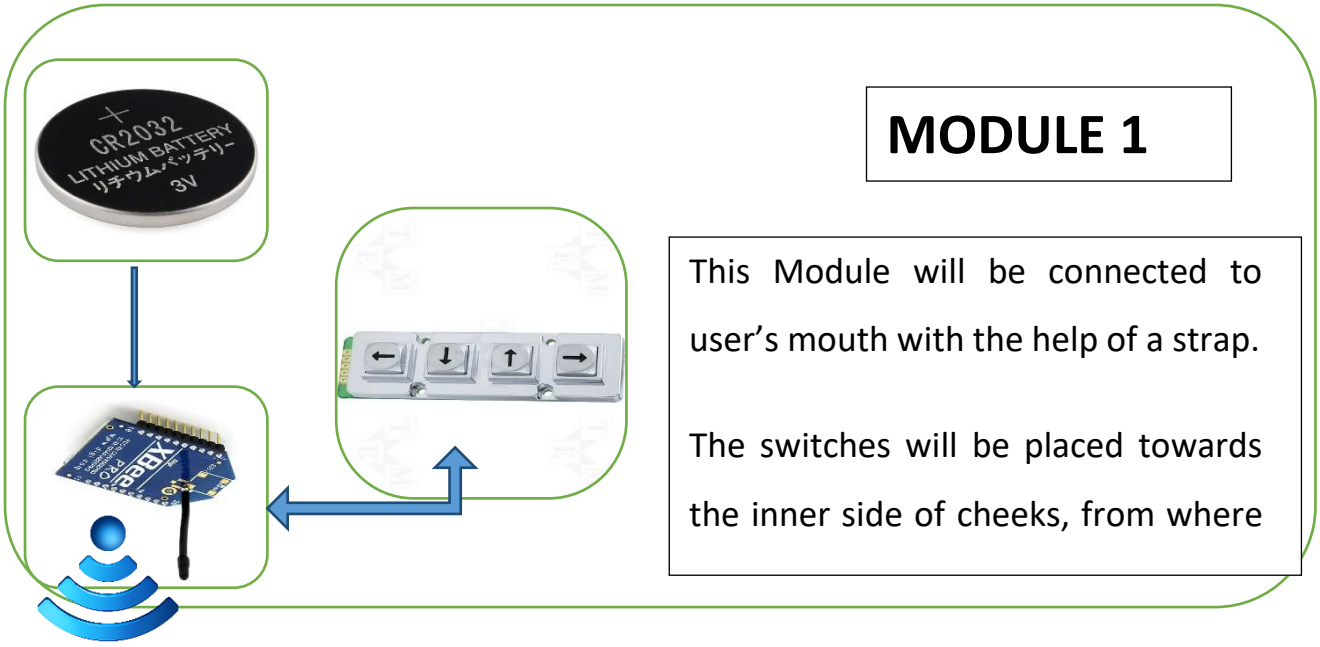


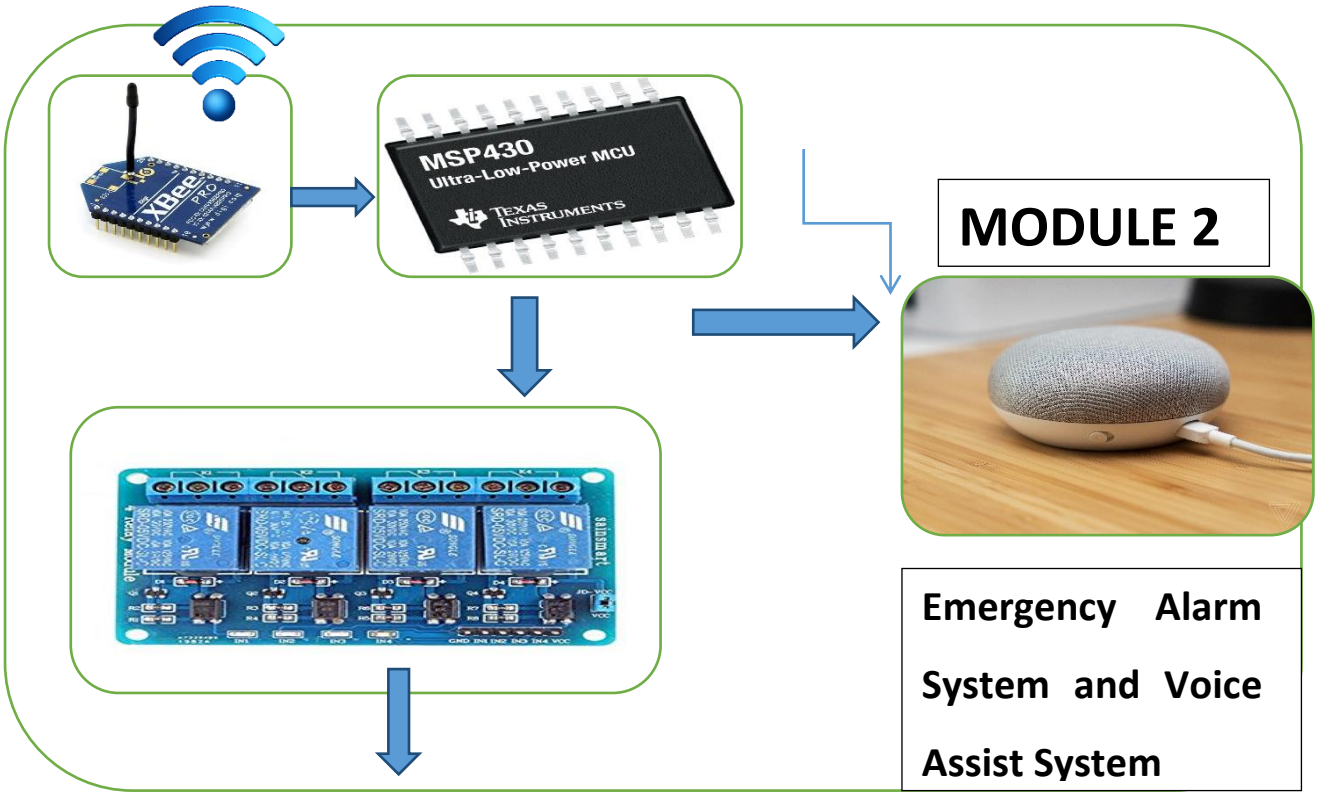
Fig: 40 Pull-down Resistor

Applications of Pull-Up and Pull-Down Resistors

- Pull-up and pull-down resistors are frequently used in interfacing devices like interfacing a switch to the microcontroller.
- Most of the microcontrollers have an inbuilt programmable pull-up/pull-down resistors.So Interfacing a switch with a microcontroller directly is possible.
- In general, pull up resistors are often used than pull-down resistors, although some microcontroller families have both pull-up and pull-down resistors.
- These resistors are often used in A/D converters to provide a controlled flow of current into a resistive sensor.
- Pull-up and pull-down resistors are used in I2C protocol bus, wherein the pull-up resistors are used to allow a single pin to act as an I/P or O/P.
- When it is not connected to an I2C protocol bus, the pin floats in a high impedance state. Pull down resistors are also used for outputs to afford a known O/P.

- **3.6 BASIC BLOCK DIAGRAM OF PROJECT**



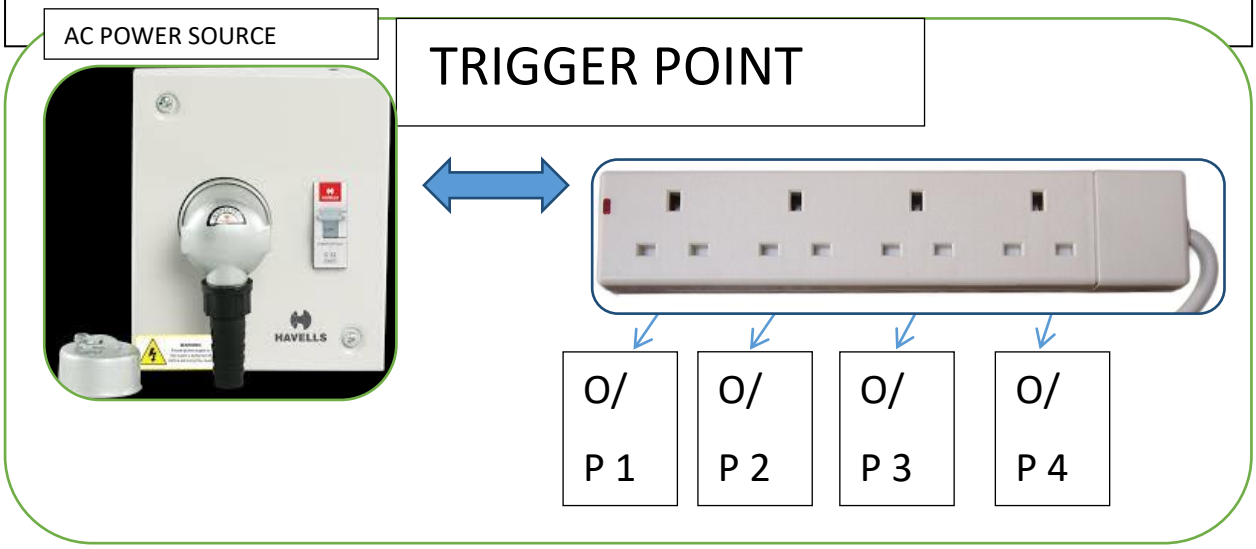


MODULE 2

Here XBee receives the input commands and sends it to the MSP430 for processing.

The Output is controlled via a relay which is connected to a fixed TRIGGER POINT which has all other devices such as fan, light etc.

Also, a speaker is connected to MSP for assisting the user and also to be used as an emergency alarm system if the user hits a desired key.



In this TRIGGER POINT MODULE, various outputs can be connected directly to the supply board as those board points will be connected directly to the relay and hence no mechanical switching will be required here.

CHAPTER-4

PROJECT DEVELOPMENT

4.1 Project Idea and Layout

The project development begins with the project idea has been conceived and required optimizations are already done. It is a step by step method of reaching the end/final project from the project idea. The idea of the low-cost assistance system for paralysed person came into account when I went to doctor for my slip-disc medical condition to a hospital and we saw that there is a requirement of the automatic assistance robots that will help the paralysed person when I search on web I found research is going on this but with other technology like facial muscle, air pressure and hall effect sensor but I want to make a simple device so I choose switches after lots of experiment.

- First, the idea is thought and after confirming its feasibility, research is done to check the history of the idea.
- Once proper information is gathered regarding the idea, we start converting the idea into a block diagram aka bird eye view of the project. Then we move into each block and make practical connections. This shows the connections and interfaces in the circuit.
- The circuit diagram is then usually realised on the breadboard to check it's working. We generally generate, what is called a prototype to simulate our circuit and make changes if required.
- This step is important as removal of errors and optimization is done majorly in this part.
- Once the circuit functioning is confirmed, we started making the design of the prototype starting from base to the final structure.
- We decided to have divide whole project into 3 parts, transmitter, receiver and their communication.
- Once the structure was completed, we interfaced each switch and xbee with the microcontroller board as well as with the relay.
- The relay is used to control the number of devices as needed by the patient.

- In this prototype, we have used many readymade components but in the final product, all the components should be connected properly.
- After the connections, we programmed the Arduino as per our required task and after few calibrations, the device started working properly.

4.2 Designing of Transmitter module

After going through all the collected information and breadboard testing final layout of the project is made on eagle software. The whole project is divided into many small modules for easy understanding.

First and most important is the tongue module as shown in the figure.

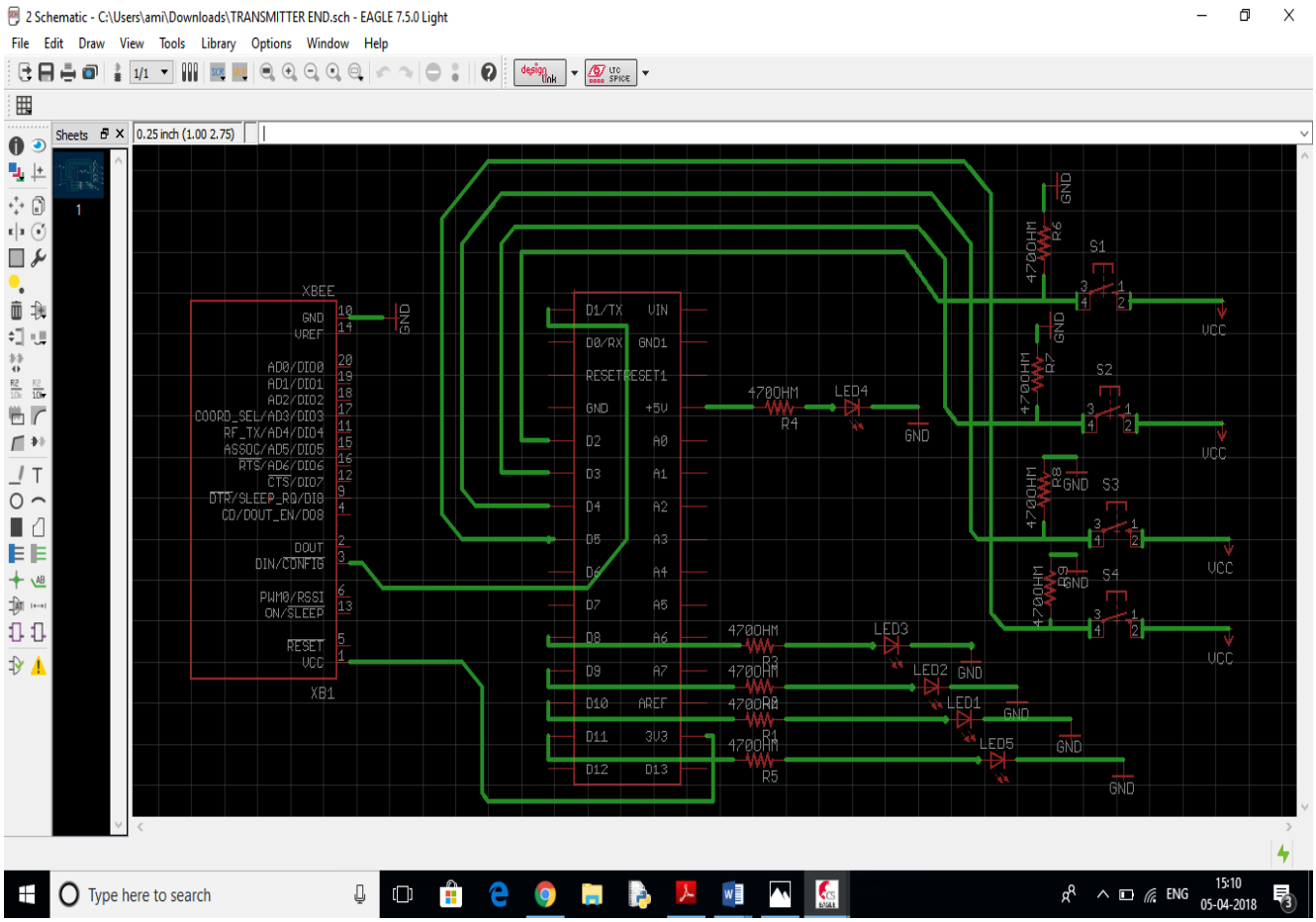


Fig : 41 Simulation and designing of Transmitter on eagle

This module consists of

- Long lasting battery power supply
- XBee module
- Aurdino nano
- Switches and resistor

4.3 Power supply module

Since at the receiver side all the connection is on AC supply we need dc power for microcontroller and xbee receiverand relay itself need DC supply for biasing.So it is always a good idea to develop a dc power supply that will directly transform ac supply for microcontroller use.

The layout of dc power supply is as follow:

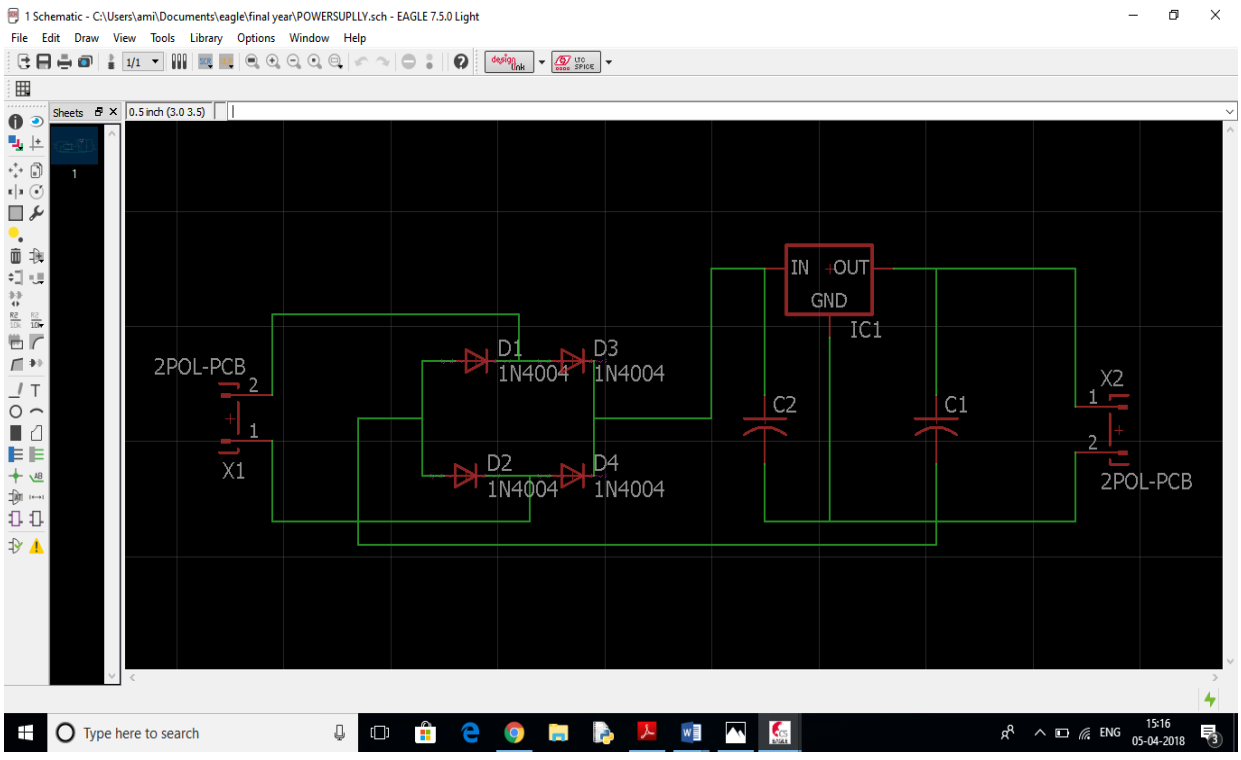


Fig: 42 Designing and simulation of power supply on eagle

It consist of

1) 4 diode
2) Lm1117 voltage regulator
3) 2 capacitor

Now the final module is the end devices as needed by the patient and we have made an electric board module which will help in controlling all the appliances.

4.4 Electric board module (END DEVICE)

It consists of

- Relay channel
- Xbee receiver
- Aurdino uno
- A different connecting node for appliances

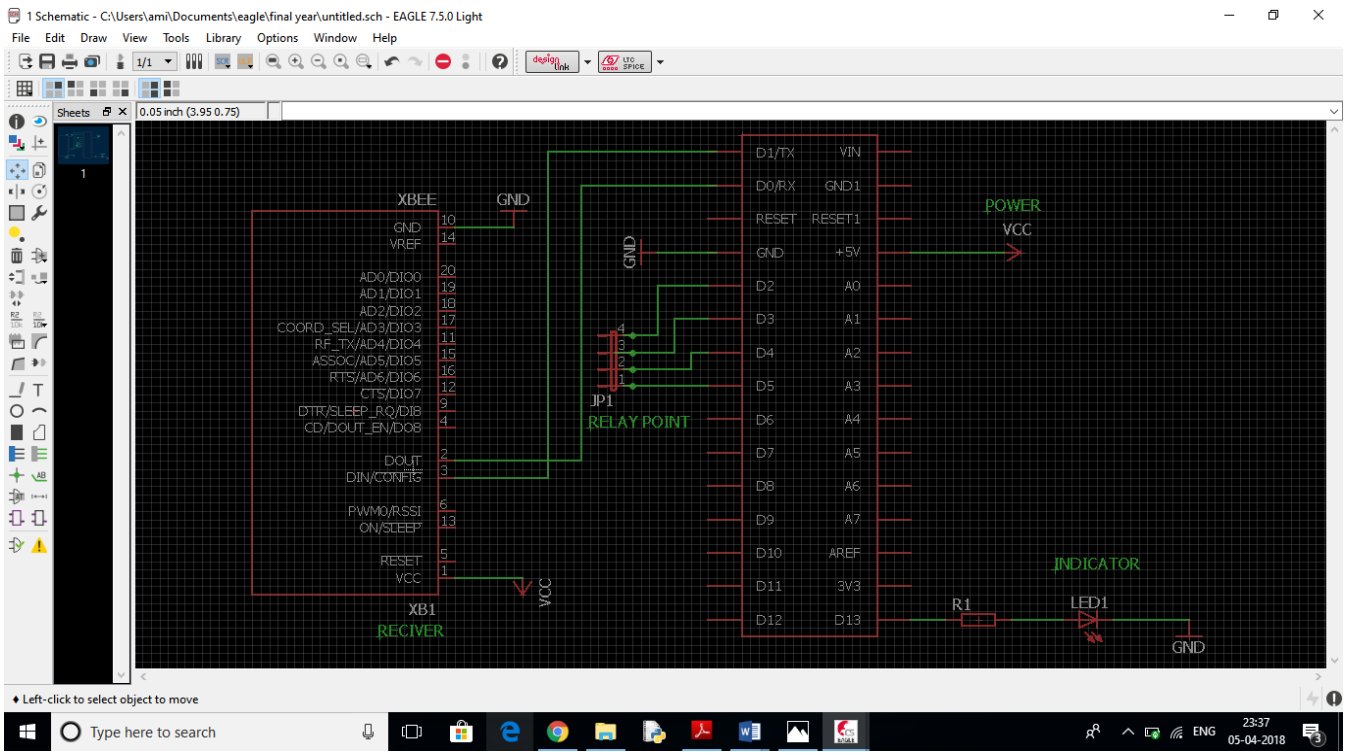


Fig: 43 Designing and Simulation of Reciever on eagle

4.5 FINAL VIEW

Final prototype look like as seen in fig.

Fig:44 Prototype of Transmitter

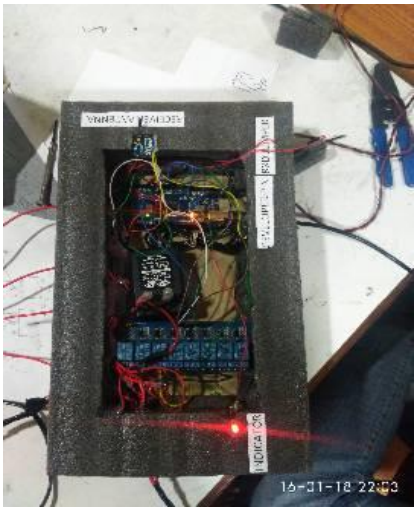


Fig: 45 Prototype pf Reciever

Fig: 46 12V DC power supply

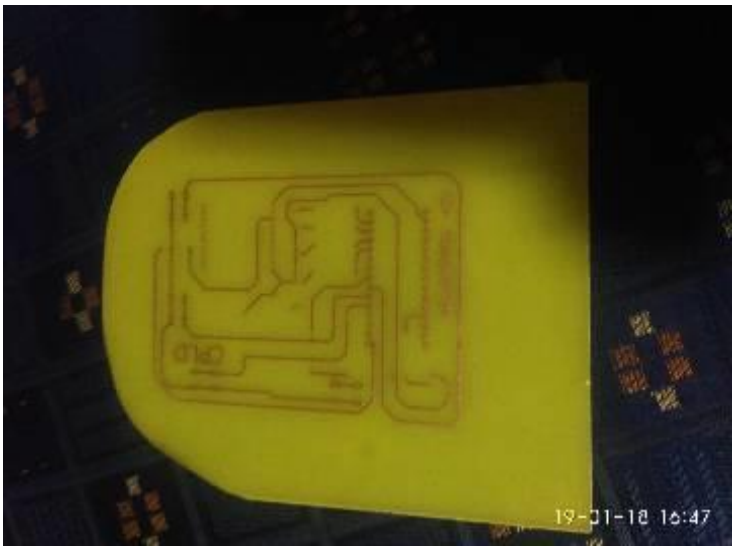


Fig: 47 Intermediate stage of pcb making

4.6 Requirements

The following are the list of requirements for designing and manufacturing of the working prototype.

4.6.1. Software Requirements

1. Arduino IDE 1.5.6-r2
2. Proteus and eagle
3. XCTU

4.6.2. Hardware Requirements

1. Aurdino nano
2. Aurdino uno
3. Set of 2 xbee
4. Relay board
5. Printer (for making base)
6. Pcb board
7. Switches
8. Li battery

4.7 Stages of development

This project has been divided into many stages from the starting, the first stage was gathering all the information and check the feasibility of this project. When we find that this project can be done then the second task was to divide into the different small module so that it can be done easily.

The whole project is divided into different stages like

1 Checking the alternative for resistive touch in the mouth.
2 Testing the different module individually
3 The configuration of xbee module through XCTU
4 Connecting all the smaller module together to form the whole project

Through all these stages one of the most difficultchallenges was the configuration and alternative of resistive touch in the mouth.

4.8 Cost of the prototype

S.NO.	MATERIAL	COST
1	AURDINO UNO	350
2	AURDINO NANO	150
3	XBEE	800 EACH
4	SWITCHES	50
5	RELAY	400
6	EXTRA	200

Table 4.1 Prototype cost

CHAPTER-5

PROGRAMME CODING

5.1 Program Code

The whole project is divided into 2 Parts:

1. Transmitter code
2. Receiver code

This code is typed in the Arduino IDE.

5.1.1 Transmitter code

```
Const int    Left = 2, Right=4, Middle=3;       //Pot at Arduino for input from switch

LED=13;                                  // declaration for led

bool value1, value2, value3 ;

        int check2 = 9;

        int check3 = 10;

        int check4 = 11;

 void setup ()

 {                                                    //Start the serial communication

 PinMode (Left, INPUT);

 PinMode (Middle, INPUT);

 PinMode (Right, INPUT);
```

```
 pinMode (LED, OUTPUT);
 pinMode (check2, OUTPUT);
 pinMode (check3, OUTPUT);
 pinMode (check4, OUTPUT);
 digitalWrite (Left, 1);
 digitalWrite(Middle, 1);
 digitalWrite(Left, 1);
 Serial.begin(9600);   //Baud rate must be the same as is on xBee module
}
void loop ()
 {                                          //Read the digital value from switch
 value1 =digitalRead (Left);
 value2 =digitalRead (Middle);
 value3 =digitalRead (Right);
 digitalWrite (LED, HIGH);                  //led for indicator that power is on
//Send the message:
 Serial.print('<');                         //Starting symbol
 Serial.print ('1');
 Serial.print (value1);                     //Value from 0 or 1
```

```
Serial.println ('>');                    //Ending symbol
Serial.print ('<');                      //Starting symbol
Serial.print ('2');
Serial.print (value2);                   //Value from 0 or 1
Serial.println ('>');                    //Ending symbol
Serial.print ('<');                      //Starting symbol
Serial.print ('3');
Serial.print (value3);                   //Value from 0 or 1
Serial.println ('>');                    //Ending symbol
      if (value1 == 1)                  //led on off for switching
      digitalWrite (check2, 1);
      else
      digitalWrite (check2,0);
      if (value2 == 1)
      digitalWrite (check3,1);
      else
      digitalWrite (check3, 0);
      if (value3 == 1)
      digitalWrite (check4, 1);
```

```
        else
         digitalWrite (check4, 0);
         delay (1000);                          //delay of 1 seconds in transmission
 }
```

5.1.2 Receiver code

```
//Variables
bool started= false,L1=true,L2=false,M1=true,M2=false,R1=true,R2=false;
  //declaration of on or off of left ,middle or right switch
bool ended = false;                          //True: Message is finished
char incomingByte ;                          //Variable to store the incoming byte
char msg[3];                                 //Message - array from 0 to 2 (digital 0 or 1)
byte index;                 //Index of array
const int lpin=2,mpin=3,rpin=4;  //pin attached to relay
void setup()
 {
          //Start the serial communication
  pinMode (lpin, OUTPUT);
  pinMode(mpin, OUTPUT);
  pinMode(rpin, OUTPUT);
```

```
  pinMode (13, OUTPUT);
//initially all the pin are high (relay trigger on low)
        digitalWrite(13,1);      //led indicator of device
        DigitalWrite(lpin, 1);
         digitalWrite (mpin, 1);
         digitalWrite(rpin, 1);

Serial.begin(9600);              //Baud rate must be the same as is on xBee module
}
void loop ()
   {
while (Serial.available()>0)
       {
           //Read the incoming byte
       Serial.flush();        // clear the previous buffer
       incomingByte = Serial. read();
//Serial.println(incomingByte);
 //Start the message when the '<' symbol is received
       if(incomingByte == '<')
```

```
{
   started = true;
   index = 0;
   msg[index] = '\0';                    // Throw away any incomplete packet
}                                        //End the message when the '>' symbol is received
   else if (incomingByte == '>')
{
   ended = true;
   break; // Done reading - exit from while loop!
}
//Read the message!
   else
{
   If (index < 4) // Make sure there is room
  {
    Msg [index] = incomingByte; // Add char to array
    index++;
    msg[index] = '\0'; // Add NULL to end
  }
```

```
        }
    }


If (started && ended)
    {
        int value = atoi(msg);          //character to integer conversion.
        If (value==11&&L1)
      {                                  //condition check for output
            digitalWrite (lpin, LOW);
            Serial.println("left on");
            L1=false;
            L2=true;
            Delay (1000);
       }
          else if (value==11&&L2)
          {
            DigitalWrite (lpin, HIGH);
            Serial.println ("left off");
            L2=false;
```

```
    L1=true;
    Delay (1000);
  }
  else if (value==21&&M1)
  {
     digitalWrite (mpin, LOW);
     Serial.println ("middle on");
     M1=false;
     M2=true;
     Delay (1000);

   else if (value==21&&M2)
   {
       digitalWrite (mpin, HIGH);
       Serial.println ("middle off");
       M2=false;
       M1=true;
Delay (1000);
     }
```

```
If (value==31&&R1)
{
   digitalWrite (rpin , LOW);
   Serial.println ("right on");
   R1=false;
   R2=true;
   Delay 1000);
 }
 else if (value==31&&R2)
 {
   digitalWrite (rpin, HIGH);
   Serial.println ("right off");
   R2=false;
   R1=true;
   Delay (1000);
   }
Serial.println (value);                    //Only for debugging
                                           // delay (1000);
```

```
            index = 0;
            msg[index] = '\0';
            started = false;
            ended = false;
        }
    }
```

CHAPTER-6

CONCLUSION, LIMITATION & FUTURE IMPROVEMENT

6.1 Conclusion

The successful completion of the project marked the achievement of goals that we started with. Like, we were able to communicate with the end devices as we targeted in our project with the Arduino as well as with the relay board. We programmed the Arduino Uno in such a way that it is able to control the appliances in the surroundings with the help of tongue module.

We have successfully completed the prototype of the low-cost assistance system for the paralysed person and it really worked according to our program code burnt in the ATmega 328P and control different appliances of the surrounding place automatically. This project may be an important step in making paralysed person self-sufficient.

6.2 Limitations

Nothing is perfect in this world, so every thing has some limitation so the project. Limitation of this project is that it can't work in some of the cases. The limitation of this device is that it will not work in some cases like damage of Hypoglossal Nerve, Vagal Nerve, and Glossopharyngeal Nerve. However, the occurrences of these problems are approximately 10%. In the above cases the mouth tongue is unable to move, and the operation of this device is based on movement of tongue. So, we have to take care of mounting this device.

Other limitation of the project is it should be fully checked if any radiation is coming from the device and it should not be harm full to the human body.

6.3 Future Improvement

In this whole project our main aim was to develop a real device which can be made and mount in the patient mouth so that he may become self-sufficient to do his/her work. For this we have consulted doctor to many hospital.

We have worked on the prototype and the device is working fine. Now we have to collect all the medical grade IC and all the equipment which can be used in manufacturing of real device. The main and real challenge in making the device is that we have to radiate a minimum power so that device will not heat up and is comfortable to the patient, as whole device is inside the mouth of patient. We know that the device should be as small as possible so we have to select SMD IC and it should be arranged in such a way that it should take minimum area. The thickness of brace is about 10mm and it should be mounted in between the braces. The pcb containing the component should be on flexible pcb and that should be easily mounted inside the braces .

The modified component needed to made the device are

1. Atmega tiny
2. Xbee SMD with power radiation of 1mw
3. Touchpad switch with water proof
4. Flexible PCB

For the development of this device we have consulted a company in Gujarat which will help us to make a single pcb board.

APPENDIX-A

ATmega 328p

- Pin diagram with Arduino mapping.

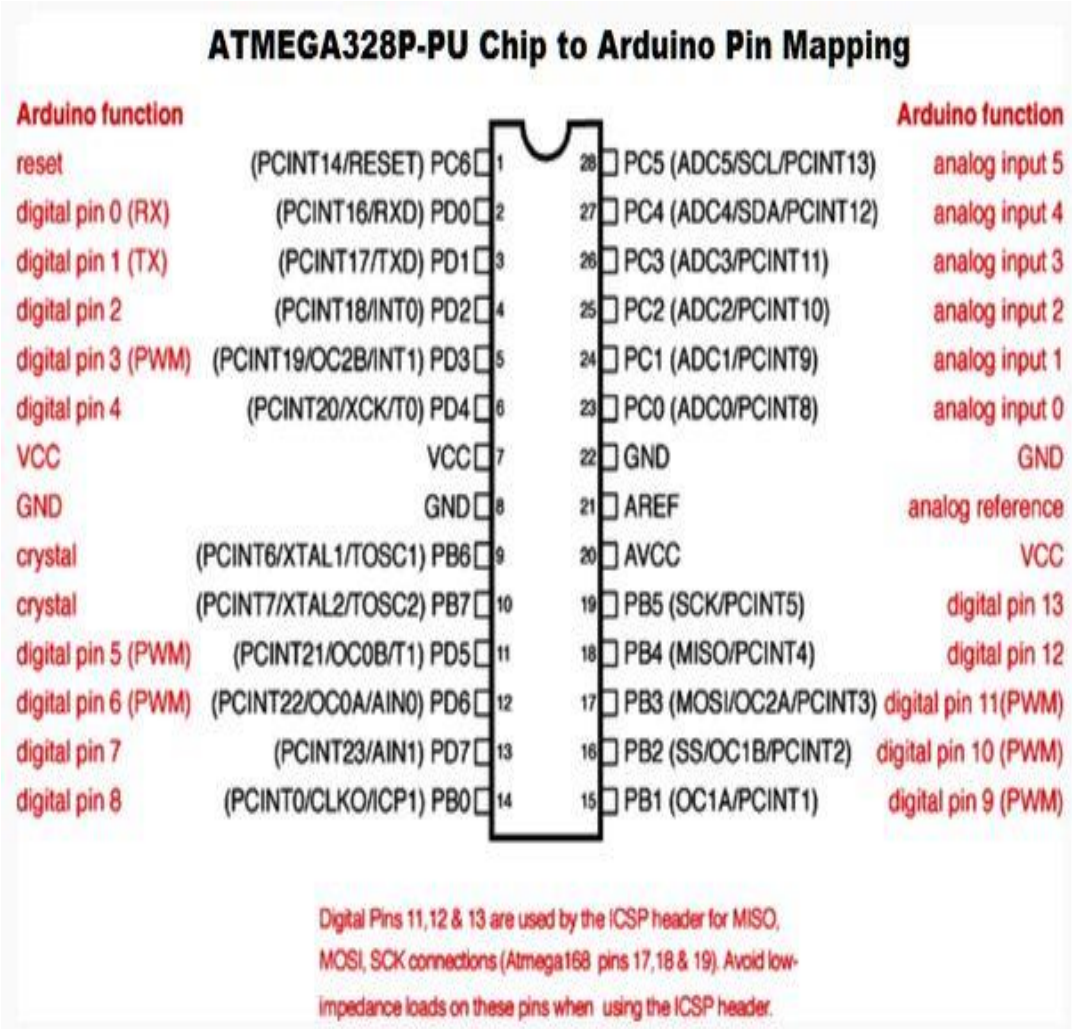


Table A.1 : pin diagram

- **Features**

Table A.1: Features of Arduino Uno with ATmega 328

Manufacturer	Atmel
Pins	28
Package Type	28 Pin DIP
Flash Program Memory	32 kbytes
EEPROM Data Memory	1 kbytes
SRAM Data Memory	2 kbytes
I/O Pins	23
Timers	Two 8-bit / One 16-bit
A/D Converter	10-bit Six Channel
PWM	Six Channels
RTC	Yes with Separate Oscillator
USART	Yes
External Oscillator	up to 20MHz

Table A.2: Pin Mapping of Arduino Uno with ATmega 328

APPENDIX-B

XBEE SPECIFICATION

- Pinout of xbee module

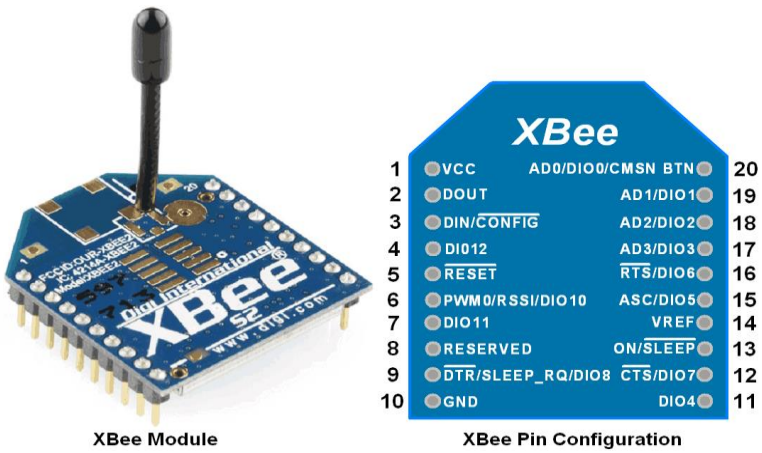


Fig. B.1: Pin diagram of xbee

- Specification of xbee

Specification	XBee
Supply Voltage	2.8 VDC – 3.4 VDC
RF Power	0 dBm, 1 mW
Outdoor Distance (LOS)	300 ft (90 m)
Indoor Distance	100 ft (30 m)
Current Draw, Receive	45 mA
Current Draw, Transmit	50 mA
Current Draw, Sleep	< 10 µA
RF Data Throughput	250 kbps
Operating Frequency, Channels	2.4 GHz, 16 Channels
Receiver Sensitivity	-92 dBm

Table. B.1: Xbee Specifications

- Pin description

Table 1-02. Pin Assignments for the XBee and XBee-PRO Modules
(Low-asserted signals are distinguished with a horizontal line above signal name.)

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	DO8*	Output	Digital Output 8
5	RESET	Input	Module Reset (reset pulse must be at least 200 ns)
6	PWM0 / RSSI	Output	PWM Output 0 / RX Signal Strength Indicator
7	PWM1	Output	PWM Output 1
8	[reserved]	-	Do not connect
9	DTR / SLEEP_RQ / DI8	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	AD4 / DIO4	Either	Analog Input 4 or Digital I/O 4
12	CTS / DIO7	Either	Clear-to-Send Flow Control or Digital I/O 7
13	ON / SLEEP	Output	Module Status Indicator
14	VREF	Input	Voltage Reference for A/D Inputs
15	Associate / AD5 / DIO5	Either	Associated Indicator, Analog Input 5 or Digital I/O 5
16	RTS / AD6 / DIO6	Either	Request-to-Send Flow Control, Analog Input 6 or Digital I/O 6
17	AD3 / DIO3	Either	Analog Input 3 or Digital I/O 3
18	AD2 / DIO2	Either	Analog Input 2 or Digital I/O 2
19	AD1 / DIO1	Either	Analog Input 1 or Digital I/O 1
20	AD0 / DIO0	Either	Analog Input 0 or Digital I/O 0

Table B.2: Pin description of aurdino.

- Top view of xbee.

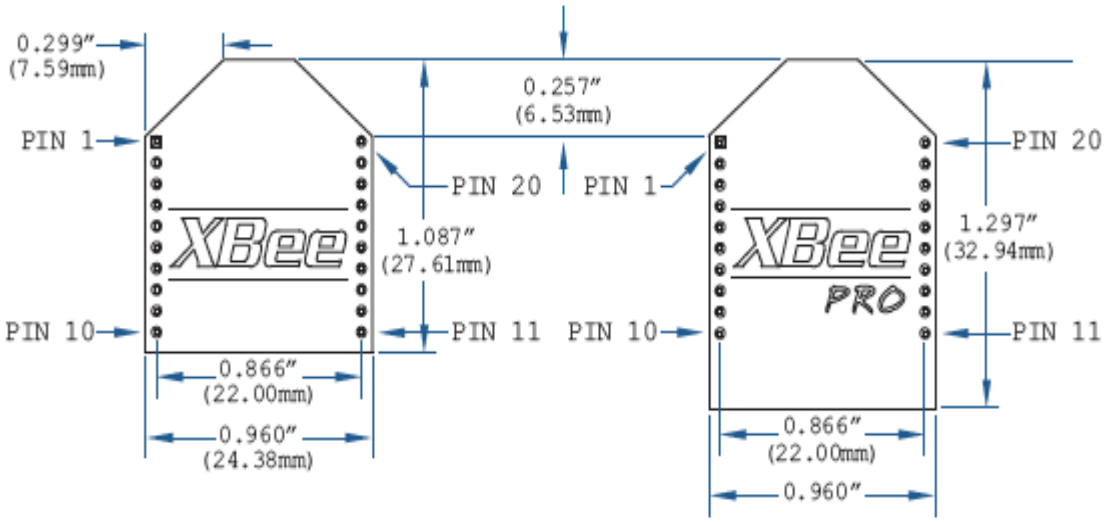


Fig B.2: Dimension of xbee

APPENDIX C

Relay Switch

- Circuit diagram of mini Relay.

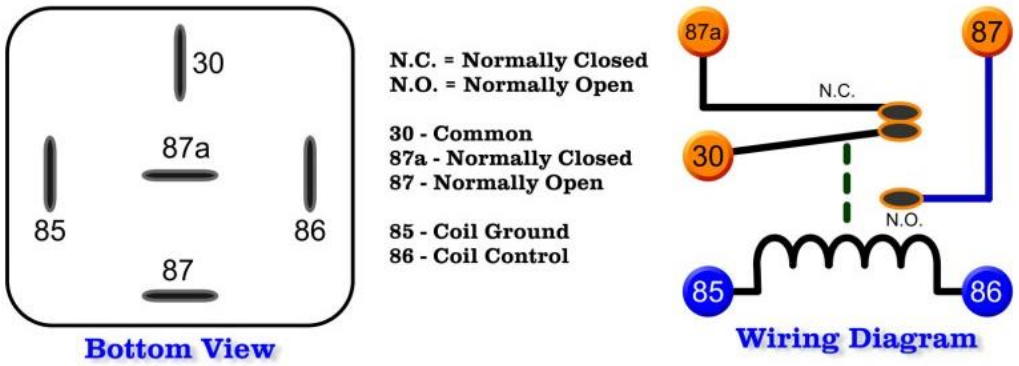


Fig. C.1: circuit diagram of mini Relay Circuit

- Specifications of Relay.

	12 Volt	24 Volt
Voltage	12-15 volts	24-30 volts
Pull-In Voltage	8 volts	16 volts
Drop-Out Voltage	1-5 volts	2.4 – 10 volts
Life Cycles	75 -200k depends on load	250,000
Max Amps NO	40	30
Max Amps NC	30	10
Coil Resistance	85 ± 5Ω	255 ± 15Ω

Table C.1: Power rating of Relay

REFERENCES

[1] 'Basic Concepts of Arduino', *http://www.arduino.cc/*.

[2] 'Various diagrams and descriptions', *Wikipedia, "http://www.wikipedia.com*.

[3]switches, https://www.electronics-tutorials.ws/logic/pull-up-resistor.html

[4] Relay,https://blog.siliconstraits.vn/relay-what-is-it-and-how-to-use-it/

[5] Xbee,"https://www.digi.com/support/supporttype?type=documentation

[6] Aurdino nano "https://store.arduino.cc/usa/arduino-nano".

[7] flexible pcb" https://www.adafruit.com/product/1894"

[8] paralysis "*Wikipedia, "http://www.wikipedia.com*

[9] Battery "https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1502062/ "

Nirankari Niwas,
Vill- panchviriya
Post- zilkabad
Chapra, Bihar, 841211
+91 8439330838
Yadav.amit.ec.2014@miet.ac.in

AMIT KUMAR YADAV

OBJECTIVE

Seeking a demanding position in a reputed industry where I can learn and enhance my skills, thereby applying my knowledge and ideas to serve the company for its growth.

EDUCATION

- **MEERUT INSTITUTE OF ENGINEERING & TECHNOLOGY – MEERUT – B.TECH**

  Affiliated to Uttar Pradesh Technical University and accredited by All India Council for Technical Education. Pursuing graduation with a major in **Electronics & Communication**, with an aggregate of **69.6%** till the 6th semesters.

- **D.B.S.D COLLEGE – GARKHA (SARAN) – INTERMEDIATE**

  Affiliated with Bihar school examination board, Completed my Intermediate in the year **2013** with majors in **Physics, Chemistry** and **Mathematics**. Scored 1st division with a percentage of **68.4 %.**

- **PVSS DAV PUBLIC SCHOOL – KODERMA – HIGH SCHOOL**

  Affiliated with Central Board of Secondary Education. Completed my High School in the year **2011** with a Cumulative Grade Point Average of **8.6** pts.

PROJECTS

ACADEMIC:

- **BOTOVEN** project of **E-yantra** in which two robots co-ordinate with each other to play different music node at a different location.
- **A pattern formation** for door opens using ultrasonic and aurdino.
- **LPG safety mechanism** and auto power cut project using MQ6 sensor and atmega microcontroller.
- Made a **Quadcopter (UAV**) to capture images send it on laptop screen.
- Currently working on **chaser drone** project of **E-yantra** to track wildlife roaming on farms and "catch" them.
- Currently working on **TDM(Tounge drive module)** assistant system for a paralyzed person for **TI INNOVATION CHALLENGE.**

SKILLS

- **Hardware**: PCB Fabrication, Soldering, MSP 430 Launchpad, Raspberry pie board, Atmel AVR board.
- **Programming Languages**: C, Python, MATLAB,
- **Software Applications**: Proteus, AVR Studio, MS Office, Keil, Code composer studio, eagle, pycharm, Cisco Packet Tracer, working on OpenCV (image processing).

COCURRICULARS

SUMMER INTERNSHIP:

- TEXAS INSTRUMENTS - INTERNSHIP PROGRAM IN MICROCONTROLLER BASED EMBEDDED SYSTEM DESIGN, Texas Instruments – Centre for Embedded Product Design, Netaji Subhas Institute of Technology, Delhi – (June 06, 2016 – July 01, 2016).
- **EAST CENTRAL RAILWAYS**-INTERNSHIP IN TELECOMMUNICATION AND SIGNALING. Venue of the internship was GM OFFICE HAJIPUR. (JUNE 2017).

- **CERTIFIED TRAININGS:**
- Embedded & Robotics basics (January 2, 2016 – January 18, 2016) conducted at **HP WINTER TRAINING NODAL CENTER**, NOIDA.
- Data network and Basics telecom (July 10, 2017 – August 9, 2017) conducted at **ALTTC BSNL GHAZIABAD**.
- 2G, 3G and 4G Network basics conducted at MIET Campus by **ERICSSON** career connect program from 23 Nov to 25 Nov 2017.

- **MINI PROJECTS:**
  - **Cross the bar** game using AVR microcontroller and LCD screen.
  - A **wireless remote control** mechanism for Robot motion using **DTMF technology** or by communicating over a **radio frequency**.
  - A small scale **fire extinguisher** using the **low frequency of sound** and speaker bass.
  - An **autonomous robot** using which can follow a line, which can avoid collisions and which can also avoid falling from edges using an ATMEGA16 microcontroller.
  - **MOOD LAMP** for generating different colour combination with the help of potentiometer.

- **WORKSHOPS:**
  - Organized a two-day workshop, coordinated by **Centre for Practical Learning and Designing (CPLD)**, MIET Meerut on printed circuit board fabrication.
  - Attended a two-day Workshop on Embedded System (AVR microcontroller) and Robotics organized by IIT Delhi.

EXTRA CURRICULARS

- Organized technical events in the college under the name of **teckspark**, a technical committee of the college.
- Was a member of organizing a committee in CULTURAL event in the college fest, K O L A H A L , 2016.
- Was a member of organizing committee in a literary event in the college fest, GLITZ-2015,2016.

INTERESTS AND HOBBIES

- Embedded Systems and Internet of Things
- Listening to music, Travelling, Reading novels, cooking food
- Meeting new people and learning new things
- Sports: cricket, badminton and athletic

ACHIEVEMENTS

- Member of **Centre for Practical Learning and Designing** (CPLD) in the college with a vision to promote a maker's culture in the college as it lacked such labs.
- Organized the **ROBOCHAMPS 2017**, And was CORE COORDINATOR of organizing committee.
- Won the **first prize** in **Road Rash** at zonal level technical Fest organized by **Dr A.P.J Abdul Kalam technical university in 2015-16**.
- Won the **second prize** in **maze runner** at zonal level technical Fest organized by **Dr A.P.J Abdul Kalam technical university in 2015-16.**
- Won the **first prize** in **Robo-Race** at zonal level technical Fest organized by Dr **A.P.J Abdul Kalam technical university in 2016-17**.
- Won the **first prize** in **ROBO WAR** at zonal level technical Fest organized by Dr **A.P.J Abdul Kalam technical university in 2016**.
- Participated in **state level Technical fest** in different robotics event in **2015-16, 2016-17.**
- Participated in **SMART INDIA HACKATHON 2017**, organized by Govt. of India.
- Participated in **Eyantra robotics competition 2016** and goes up to the final round.
- PROJECT IDEA selected in NIYANTRA 2017, among 180 ideas in the whole India.
- Organized the **Robochamps 2016**, national level Robolympaid and was a member of organizing committee.
- Currently working on a project of **TI INNOVATION CHALLENGE** organized by DST and IIM Bangalore.

DECLARATION

I hereby declare that all the information stated above is true up to the best of my knowledge.

Date:

Place: **Amit Kumar Yadav**