**ABSTRACT**

Most of the artificial intelligence will eventually lead to robotics. Most neural networking, natural language processing , image recognition , speech recognition/synthesis research aims at eventually incorporating their technology into the epitome of robotics-the creation of a fully humanoid robot.

The field of robotics has been around nearly as long as Artificial intelligence –but the field has made little progress. This is only natural , since the field not only attempts to conquer intelligence, but also the body that embodies it-a formidable task. Robotics ,through, is not just about humanoid robots; but also about their commercial applications in manufacturing , safety and hundreds of other fields.

It is only relatively recently that robots have started to employ a degree of artificial intelligence in their work –many robots required human operators, or precise guidance throughout their missions. Slowly, robots are becoming more and more autonomous. Robotics is an absolutely fascinating field that interests most people. Robot is a system that contains sensors, control systems, manipulators, power supplies and software all working together to perform a task. Robot should have sensing, movement, energy and intelligence characteristics.

The project deals with one of the applications of vehicles. In this project one moving object is developed such that it is moved as per commands given by the voice and that is received by pic microcontroller using Bluetooth communication. This project is equipped with DC motor, Bluetooth module, finger print module, pic microcontroller along with the power supply units(batteries). Vehicle finds it applications in the real time.

**CHAPTER 1**

**OVERVIEW AND REQUIREMENT SPECIFICATIONS**

**1.1 INTRODUCTION**

It has always been a dream of human being to create machines that behave like humans. Recognizing the speech and responding accordingly is an important part of this dream. With the improvements of the technology and researches on artificial intelligent, this dream comes true relatively.

In this project, it is aimed to make a contribution to this dream. Controlling the machines and environment with speech makes human life easier and more comfortable. This project is a simple implementation of this approach. A robot is controlled by voice commands. Voice command is taken through a microphone, processed in computer and sent to the robot and finally the robot acts accordingly.

Speech is the most used way of communication for people. We born with the skills of speaking learn it easily during our early childhood and mostly communicate with each other with speech throughout our lives. By the developments of communication technologies in the last era, speech starts to be an important interface for many systems. Instead of using complex different interfaces, speech is easier to communicate with computers.

In this project, it is aimed to control a robot with speech commands. The robot is able to recognize spoken commands to move correctly. To give a direction to robot, first the voice command is send to the ANDROID phone. The android recognizes the command by speech recognition system. And then android converts the voice command to direction command that predefined and recognizable by robot. When the robot gets the direction command, it moves according to spoken command.

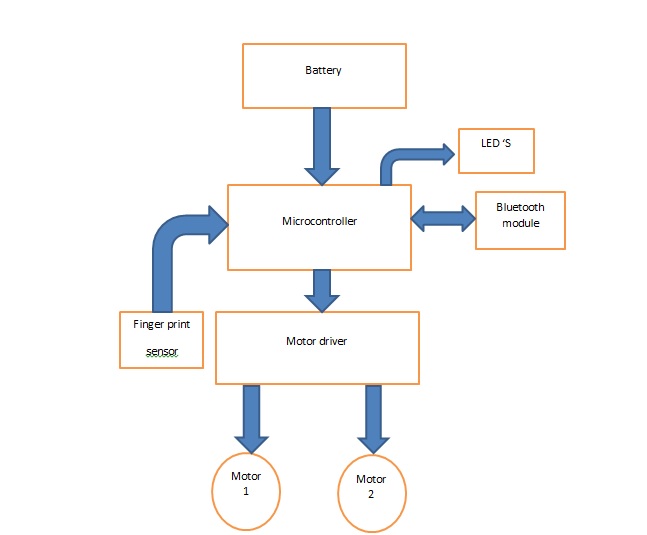
An embedded system is a system which is going to do predefined task is the embedded system and is even defined as combination of both software and hardware

A general purpose definition of embedded systems is that they are devices used to control, monitor or assist the operation of equipment, machinery or plant. “Embedded” reflects the fact that they are an integral part of the system. In many cases their embeddedness may be such that their presence is far from obvious to the casual observer and even the more technically skilled might need to examine the operation of a piece of equipment for some time before using able to conclude that an embedded control system was involved in its functioning. At the other extreme a general purpose computer may be used to control the operation of a large complex processing plant, and its presence will be obvious.

All embedded systems are including computers or microprocessors. Some of these computers are however very simple systems as compared with a personal computer.

The very simplest embedded systems are capable of performing only a single function or set of functions to meet a single pre determined purpose. In more complex systems an application program that enables the embedded system to be used for a particular purpose in a specific application determines the functioning of the embedded system. The ability to have programs means that the same embedded system can be used for a variety of different purposes. In some cases a microprocessor may be designed in such a way that application software for a particular purpose can be added to the basic software in a second process, after which it is not possible to make further changes. The applications software on such processer is sometimes referred to as firmware.

The simplest devices consist of a single microprocessor (often called a “chip”), which may itself be packaged with other chips in a hybrid system or Application specific integrated circuit (ASIC). Its inputs comes from a detector or sensor and its output goes to a switch or activator which (for example) may start or stop the operation of a machine or, by operating a valve, may control the flow of fuel to an engine.

As the embedded system is the combination of both software and hardware Fig 1.1.1 Block diagram of embedded system

Voice controlled robot is a mobile robot which is control by some specified voice commands. The mobile application is capable of identifying five commands which are “Stop”,”Forward”,”Back”,”Left”,”Right”and “Stop”. In this embedded systems project, we make a 2-WD robotic car which we can control using voice through a mobile application. Application listens and sends the instruction to the pic microcontroller using Bluetooth and then pic microcontroller performs the specified operation. Voice recognition application is not 100% accurate. The application is sensitive to the surrounding noises. It sometimes misinterprets the voice commands given to the robot. But you can design your own application which can ignore the surroundings and can receive your own voice only. Finger print sensor authenticate the access to the vehicle on starting and it also block the unauthorized users.

**Memory:** it is used to store data or address.

**Peripherals:** these are the external devices connected.

**Processors:** it is an IC which is used to perform some task.

Processors are classified in to 4 types like:

1. Microprocessor (µp)
2. Microcontroller (µc)
3. Digital signal processor (DSP
4. Application specific integrated circuits (ASIC).

**Microprocessor (µp):**

It is an electronic chip which performs arithmetic and logical operations with assistance of internal memory.

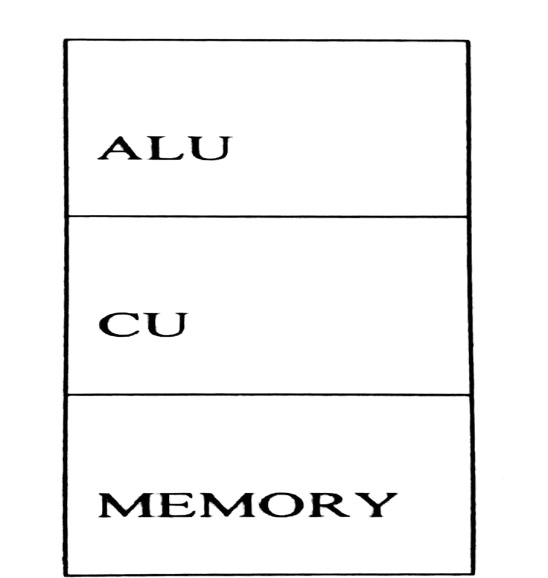


Fig1. 1.2 block diagram of Microprocessor

**Microcontroller (µc):**

It is a highly integrated micro processor designed for specific use in embedded systems.

**ALU**

**CU**

**MEMORY**

Fig 1.1.3 diagram of Microcontroller

**List of applications of embedded system**

1. Manufacturing and process control
2. Construction industry
3. Transport
4. Building and premises
5. Domestic service
6. Communications
7. Office systems and mobile equipment
8. Banking, finance and commercial
9. Medical diagnostics, monitoring and life support
10. Testing, monitoring and diagnostic systems

1. **SYSTEM OVERVIEW**

This project voice controlled robotic vehicle helps to control robot through voice commands received via android application .the integration of control unit with Bluetooth device is done to capture and read the voice commands. The robotic vehicle then operates as per command received via android application. For this PIC16F876A microcontroller is integrated in the system which makes it possible to operate the vehicle via android application. The controlling device may be any android based smart phone/tab etc having an android OS. The android controlling system provides a good interactive GUI that makes it easy for the user to control the vehicle. the transmitter uses an android application required for transmitting the data. The receiver end reads these commands and interprets them into controlling the robotic vehicle.

The android device sends commands to move the vehicle in forward, backward, right and left directions .after receiving the commands, the microcontroller the operates the motors to move the vehicle in four directions. The communication between android device and receiver is sent as serial communication data. The microcontroller program is designed to move the motor through a motor driver IC as per the commands sent by android device.

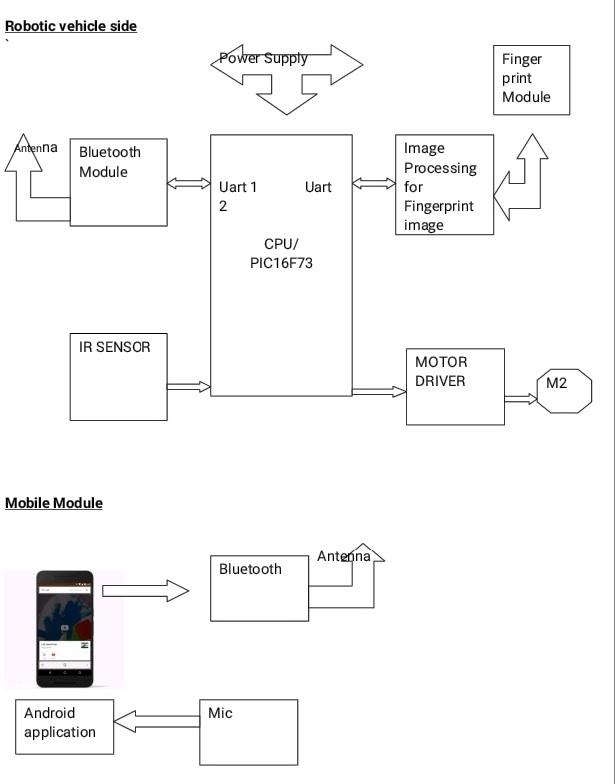
Vehicle security is an important issue these days due to rising number of vehicle thefts. Also one more issue vehicles is handling its keys. Keys need to be carried and misplacing keys or losing them will cause a serious issue. Here we propose a solution to this problem by using a fingerprint authenticated vehicle starter system. The system provides a secure and hassle free way to start /stop vehicle engine. User just need to scan finger to start the car, no need to carry any key. The system only allows authorised users to start the vehicle. Users can first register in to the system by scanning fingerprints. The system allows multiple users to register as authorised users. When into monitoring mode, the system checks for users to scan. On scanning, the system checks if user is authorised users only.

**Hardware required**

1. PIC16F876A
2. Fingerprint module
3. Motor driver
4. Motors
5. Bluetooth module
6. Batteries

Fig1. 2.1 robotic vehicle

1. **Block Diagram**

****

PIC16F876A

Block diagram is a diagram of a system in which the principal parts or functions are represented by blocks connected by lines that show the relationships of the blocks. They are heavily used in the engineering world in hardware design, electronic design, software design, and process flow diagrams.

Block diagrams rely on the principle of the black box where the contents are hidden from view either to avoid being distracted by the details or because the details are not known. We know what goes in, we know what goes out, but we can't see how the box does its work.

The above block diagram shows the important block involves in this project. There are mainly 6 blocks in this project.

1. Power supply
2. UART
3. Bluetooth module
4. Fingerprint module
5. PIC micro controller
6. Motor driver

**POWER SUPPLY**

Most of the power supplies convert high voltage AC mains electricity to suitable low voltage DC voltage supply. In this project, use battery as a power supply. In this project two section of batteries are used one for the motor driver (9v) and another for PIC microcontroller (5v).

 Fig 1.3.1 9v battery for PIC microcontroller

A battery is self contained, chemical power pack that can produce a limited amount of electrical energy wherever it’s needed. Unlike normal electricity, which flows to your home through wires that start of in a power plant, a battery slowly converts chemicals packed inside in to electrical energy, typically released over a period of days, weeks, months, or even years.

**MICROCONTROLLER**

A microcontroller (sometimes abbreviated **µC**, **uC** or **MCU**) is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. Program memory in the form of NOR flash or OTP ROM is also often included on chip, as well as a typically small amount of RAM. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications.

Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools, toys and other embedded systems. By reducing the size and cost compared to a design that uses a separate microprocessor, memory, and input/output devices, microcontrollers make it economical to digitally control even more devices and processes. Mixed signal microcontrollers are common, integrating analog components needed to control non-digital electronic systems.

Some microcontrollers may use 4-bit words and operate at clock rate frequencies as low as 4 kHz, for low power consumption (single-digit milli watts or microwatts). They will generally have the ability to retain functionality while waiting for an event such as a button press or other interrupt; power consumption while sleeping (CPU clock and most peripherals off) may be just nanowatts, making many of them well suited for long lasting battery applications. Other microcontrollers may serve performance-critical roles, where they may need to act more like a digital signal processor (DSP), with higher clock speeds and power consumption.

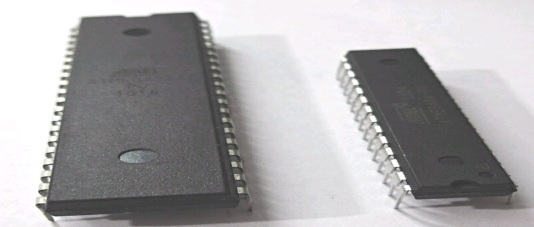


Fig 1.3.2 microcontroller

In this project we used PIC16F876A microcontroller, Low power, high performance 16 bit with 8k bytes of in system programmable flash memory. It consist of 256 bytes of EEPROM, 14kB program memory and 368bytes RAM. The operating voltage is 2-5.5V.

**UART (Universal Asynchronous Receiver/Transmitter)**

UART stands for universal asynchronous receiver transmitter. It’s not a communication protocol like SPI and 12C, but a physical circuit in a microcontroller, or a stand -alone IC. A UART main purpose is to transmit and receive serial data.

One of the best thing about UART is that it only uses to wires to transmit data between device. In UART communication, two UARTs communicate directly with each other. The transmitting UART converts parallel data from a controlling device like CPU in to serial form, transmits it in serial to the receiving UART, which then converts the serial data back in to parallel data for the receiving device. Only two wires are needed to transmit data between two UARTs. Data flows from the Tx pin of the transmitting UART to the Rx pin of the receiving UART.

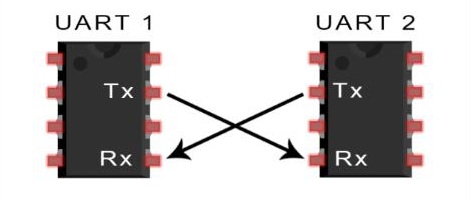


Fig 1.3.3 Data flow in UART

**BLUETOOTH MODULE**

A Bluetooth module is usually a hardware component that provides a wireless product to work with the computer; or in some cases, the Bluetooth may be an accessory or peripheral, or a wireless headphone or other product (such as cell phones can use.). If the computer (is this computer related?) has hardware support to use Bluetooth products and connections, then whatever it is you are trying to download and use, may work.

There are component Bluetooth wireless modules with a USB plug on them to add this BT to computers that did not have it built-in. Some of these modules may require drivers, but usually Mac OS X has drivers built into the system to support several products. In the chapter 4 we discussed in detailed about Bluetooth module.

These small size Bluetooth TTL transceiver modules are designed for serial communication (SPP - serial port profile). It allows your target device to both send or receive TTL data via Bluetooth technology without connecting a serial cable to your computer. The modules with the **HC-03** and **HC-05** firmware are the Master and Slave integrated Bluetooth serial modules with firmware which allows you to modify master and slave mode at any time. HC-03 is industrial grade products, HC-05 are commercial grade products.

The modules with the **HC-04** and **HC-06** firmware are the modules which are factory set to be Master or Slave modules. Master and slave mode cannot be switched from the factory setting. HC-04 is an industrial grade product; HC-06 is a commercial grade product. The modules with the **HC-09** firmware are replacements for the HC-06 and HC-07 modules.



Fig 1.3.4 HC-05 Bluetooth module

**MOTOR DRIVER**

In [electronics](http://en.wikipedia.org/wiki/Electronics), a driver is an [electrical circuit](http://en.wikipedia.org/wiki/Electrical_circuit) or other [electronic component](http://en.wikipedia.org/wiki/Electronic_component) used to control another circuit or other component, such as a high-power transistor.

They are usually used to regulate current flowing through a circuit or is used to control the other factors such as other components, some devices in the circuit. The term is often used, for example, for a specialized [integrated circuit](http://en.wikipedia.org/wiki/Integrated_circuit) that controls high-power [switches](http://en.wikipedia.org/wiki/Switches) in switched-mode [power converters](http://en.wikipedia.org/wiki/Power_converter). An amplifier can also be considered a driver for [loudspeakers](http://en.wikipedia.org/wiki/Loudspeaker), or a [constant voltage](http://en.wikipedia.org/wiki/Voltage_source) circuit that keeps an attached component amplifier, typically the driver circuit requires current gain, often the ability to discharge the following transistor bases rapidly, and low output impedance to avoid or minimize distortion.



Fig 1.3.5 L293D motor driver IC

The L293D is a popular motor driver IC that is usable from 6 to12V, at up to 1A total output current. By itself, the IC is somewhat difficult to wire and use, but the Compact L293D Motor Driver makes it much more convenient to use.

Board Special Features

* Four motor direction indicator LEDS
* Schottky EMF-protection diodes
* Socket pin connectors for easy logic interfacing

Enable pins are user accessible.

**FINGER PRINT MODULE**

****

Fig 1.3.6 finger print module

This is a finger print sensor module with TTL UART interface for direct connections to microcontroller UART or to PC through USB serial adaptor. The user can store the finger print data in the module and can configure it in 1:1 or 1:N mode for identifying the person. The finger print module can directly interface with 3V or 5V microcontroller.

Optical biometric finger print reader with great features can be embedded in to a variety of end products, such as: access control, attendance safety deposit box, car door locks.

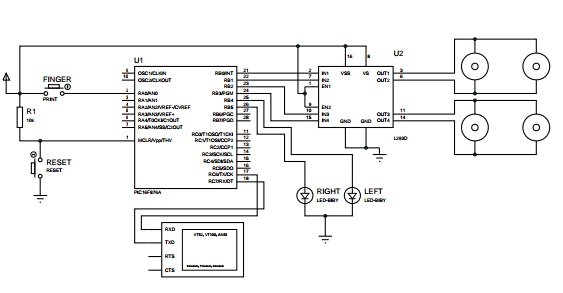
**Features:**

* 1. Integrated image collecting and algorithm chip together, all-in-one
  2. Finger print reader can conduct secondary development, can be embedded in to a variety of end products
  3. Low power consumption, low cost, small size, excellent performance
  4. Professional optical technology, precise module manufacturing techniques
  5. Good image processing capabilities, can successfully capture image up to resolution 500 dpi.

**CHAPTER 2**

**HARDWARE REQUIREMENT SPECIFICATIONS**

* 1. **CIRCUIT DIAGRAM**

**** Fig2. 1.1 circuit diagram of voice controlled robotic vehicle

In this project we use PIC16F876A microcontroller, acts as the brain of the device and controls all the other devices connected to it. A 4MHz crystal oscillator is used to generate the oscillations required to machine cycles inside the microcontroller. In this project port A, B and port C is used. Port B is set as the output port. The pin B0, B1, B2, B3 are the output of the PIC, they are connected to the input pin of the motor driver IC, the pins are 2,7, 10,15 respectively. The finger print module is connected to the port A. Bluetooth module is connected to the port c. The output of the motor driver IC is connected to the two parallel connected motor. In the finger print module can have the three switches Add, Clear and Search**.** In this circuit, crystal oscillator (4MHz), resistor (10kΩ) and switches are used.

**Crystal oscillator:**

It is an electronic circuit that uses the mechanical resonance of a vibrating crystal of piezoelectric material to create an electrical signal with a very precise frequency. This frequency is commonly used to keep track of time (as in quartz wristwatches), to provide a stable clock signal for digital integrated circuits, and to stabilize frequencies for radio transmitters. Using an amplifier and feedback, it is an especially accurate form of an electronic oscillator. In this project crystal oscillator frequency is 4 MHz. The crystal oscillator is shown below.



Fig 2.1.2 crystal oscillator

**Resistors:**

A resistor is a passive two-terminal electrical component that implements electrical resistance as a circuit element. Resistors act to reduce current flow, and, at the same time, act to lower voltage levels within circuits. Resistors may have fixed resistances or variable resistances, such as those found in thermistors, varistors, trimmers, photo resistors and potentiometers. Resistors are common elements of electrical networks and electronic circuits and are ubiquitous in electronic equipment.

**Switches:**

In electrical engineering, a switch is an electrical component that can break an electrical circuit, interrupting the current or diverting it from one conductor to another. The most familiar form of switch is a manually operated electromechanical device with one or more sets of electrical contacts, which are connected to external circuits. Each set of contacts can be in one of two states: either "closed" meaning the contacts are touching and electricity can flow between them, or "open", meaning the contacts are separated and the switch is no conducting.

**2. 2 CIRCUIT DIAGRAM EXPLANATION**

* + 1. **PIC16F876A MICROCONTROLLER**

A microcontroller can be compared to a small stand alone computer, it is a very powerful device, which is capable of executing a series of pre-programmed tasks and interfacing with other hardware devices. Being packed in a tiny integrated circuit (IC) whose size and weight is usually negligible, it is becoming the perfect controller for robots or any machines requiring some kind of intelligent automation. A small single microcontroller can be sufficient to control a small mobile robot, an automatic washer machine or a security system. Any microcontroller contains a memory to store the program to be executed, and a number of input/output lines that can be used.

**Features:**

1.368 bytes RAM

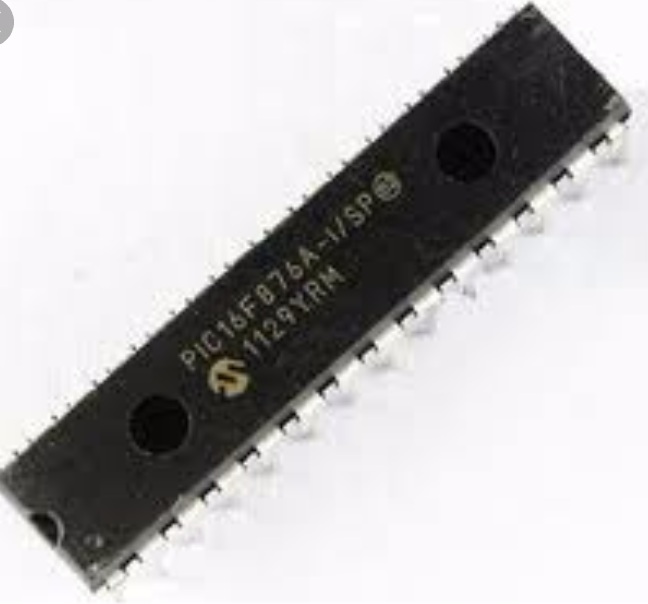
2. 14KB program memory

3. 8 channels of 10 bit A/D converter

4. 256 bytes of EEPROM

5. 5 PORTS and 35 I/O pins

6. 8k x 14 words of flash program memory

****Fig 2.2.1.1 PIC16f876A micro controller

**Status Register**

The Status register contains the arithmetic status of the ALU, the Reset status and the bank select bits for data memory. The Status register can be the destination for any instruction, as with any other register. If the Status register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the TO and PD bits are not writable, therefore, the result of an instruction with the Status register as destination may be different than intended. For example, CLRF STATUS, will clear the upper three bits and set the Z bit. This leaves the Status register as000u u1uu (where u = unchanged).It is recommended, therefore, that only BCF, BSF,SWAPF and MOVWF instructions are used to alter the Status register because these instructions do not affect the Z, C or DC bits from the Status register.

R/W-0 R/W-0 R/W-0 R-1 R-1 R/W-X R/W-X R/W-X

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| IRP | RP1 | RP0 | TO | PD | Z | DC | C |

Bit-7 bit-0

Table 2.1 status register

bit 7 **IRP:** Register Bank Select bit (used for indirect addressing)

1 = Bank 2, 3 (100h-1FFh)

0 = Bank 0, 1 (00h-FFh)

bit 6-5 **RP1:RP0**: Register Bank Select bits (used for direct addressing)

11 = Bank 3 (180h-1FFh)

10 = Bank 2 (100h-17Fh)

01 = Bank 1 (80h-FFh)

00 = Bank 0 (00h-7Fh)

Each bank is 128 bytes.

bit 4 **TO**: Time-out bit

1 = After power-up, CLRWDT instruction or SLEEP instruction

0 = A WDT time-out occurred

bit 3 **PD**: Power-down bit

1 = After power-up or by the CLRWDT instruction

0 = By execution of the SLEEP instruction

bit 2 **Z**: Zero bit

1 = The result of an arithmetic or logic operation is zero

0 = The result of an arithmetic or logic operation is not zero

bit 1 **DC**: Digit carry/borrow bit (ADDWF, ADDLW,SUBLW,SUBWF instructions)

(for borrow, the polarity is reversed)

1 = A carry-out from the 4th low order bit of the result occurred

0 = No carry-out from the 4th low order bit of the result

bit 0 **C**: Carry/borrow bit (ADDWF, ADDLW,SUBLW,SUBWF instructions)

1 = A carry-out from the Most Significant bit of the result occurred

* + - 1. = No carry-out from the Most Significant bit of the result occurred.

In this project, use the ports A, B, C, port B is set as the output port and port A and port C is set as the input port. Bluetooth module is connected to the port C, Rx and Tx pin respectively. Fingerprint module is connected to the port A.

**I/O PORTS:**

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

**PORT A:**

PORTA is a 6-bit wide, bidirectional port. The corresponding data direction register is TRISA. Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISA bit (= 0) will

make the corresponding PORTA pin an output (i.e., put the contents of the output latch on the selected pin).

Reading the PORTA register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, the value is modified and then written to the port data latch. Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin. The RA4/T0CKI pin is a Schmitt Trigger input and an open-drain output. All other PORTA pins have TTL input levels and full CMOS output drivers. Other PORTA pins are multiplexed with analog inputs and the analog VREF input for both the A/D converters and the comparators. The operation of each pin is selected by clearing/setting the appropriate control bits in the ADCON1 and/or CMCON registers.

The TRISA register controls the direction of the port pins even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

**PORT B:**

PORTB is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISB. Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISB bit (= 0)

will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin). Three pins of PORTB are multiplexed with the In-Circuit Debugger and Low-Voltage Programming function: RB3/PGM, RB6/PGC and RB7/PGD.

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit RBPU (OPTION\_REG<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset. Four of the PORTB pins, RB7:RB4, have an interrupton- change feature. Only pins configured as inputs can cause this interrupt to occur (i.e., any RB7:RB4 pin configured as an output is excluded from the interrupton- change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The “mismatch” outputs of RB7:RB4 are OR’ed together to generate the RB port change interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from Sleep. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

a) Any read or write of PORTB. This will end the mismatch condition.

b) Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared. The interrupt-on-change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt-on-change feature. Polling of PORTB is not recommended while using the interrupt-on-change feature. This interrupt-on-mismatch feature, together with software configurable pull-ups on these four pins, allow easy interface to a keypad and make it possible for wake-up on key depression.

**PORT C:**

PORTC is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISC. Setting a TRISC bit (= 1) will make the corresponding PORTC pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISC bit (= 0) will make the corresponding PORTC pin an output (i.e., put the contents of the output latch on the selected pin). PORTC is multiplexed with several peripheral functions. PORTC pins have Schmitt Trigger input buffers.

When the I2C module is enabled, the PORTC<4:3> pins can be configured with normal I2C levels, or with SM Bus levels, by using the CKE bit (SSPSTAT<6>). When enabling peripheral functions, care should be taken in defining TRIS bits for each PORTC pin. Some peripherals override the TRIS bit to make a pin an output, while other peripherals override the TRIS bit to make a pin an input. Since the TRIS bit override is in effect while the peripheral is enabled, read-modify write instructions (BSF, BCF, XORWF) with TRISC as the destination, should be avoided.

* + 1. **POWER SUPPLY**

Most of the power supplies convert high voltage AC mains electricity to suitable low voltage DC voltage supply. In this project, use battery as a power supply. In this project two section of batteries are used one for the motor driver (9v) and another for PIC microcontroller (5v).



Fig 2.2.2.1 9v battery for PIC microcontroller

A battery is self contained, chemical power pack that can produce a limited amount of electrical energy wherever it’s needed. Unlike normal electricity, which flows to your home through wires that start of in a power plant, a battery slowly converts chemicals packed inside in to electrical energy, typically released over a period of days, weeks, months, or even years.

The nine volt battery format is commonly available in primary carbon-zinc and alkaline chemistry, in primary lithium iron disulfide, and in rechargeable form in nickel-cadmium, nickel-metal hydride and lithium-ion. Mercury oxide batteries of this format, once common, have not been manufactured in many years due to their mercury content. Designations for this format include NEDA 1604 and IEC 6F22 (for zinc-carbon) or MN 1604 6LR61 (for alkaline). The size, regardless of chemistry, is commonly designated PP3-a designation originally reserved solely for carbon-zinc.

The most nine-volt alkaline batteries are constructed of six individual 1.5V cells enclosed in a wrapper. These cells are slightly smaller than LR8D425 AAAA cells and can be used in their place for some devices, even though they are 3.5mm shorter. Carbon zinc types are made with six flat cells in a stack, enclosed in a moisture-resistant wrapper to prevent drying. Primary lithium types are made with 3 cells in series

.

Fig2. 2.2.2 9-volt Battery snap

The battery has both terminals in a snap connector on one end. The smaller circular (male) terminal is positive, and the larger hexagonal or octagonal (female) terminal is the negative contact. The connectors on the battery are the same as on the connector itself; the smaller one connects to the larger one and vice versa. The same snap style connector is used on other battery types in the power pack (PP) series. Battery polarization is normally obvious since mechanical connection is usually only possible in one configuration. A problem with this style of connector is that it is very easy to connect two batteries together in a short circuit, which quickly discharges both batteries, generating heat and possibly a fire. Because of this hazard, 9-volt batteries are used.

**Chemical Reactions**

The positive and negative electrodes are separated by the chemical electrolyte. It can be liquid, but in an ordinary battery it is more likely to be dry powder.

When you connect the battery to a lamp and switch on, chemical reactions start happening. One of these reactions generates positive ions and electrons at the negative electrode. The positive ions flow into the electrolyte while the electrons flow around to the outside circuit to the positive electrode and make the lamp light up on the way. There’s a separate chemical reaction happening at the positive electrode, where incoming electrons recombine with ions taken out of the electrolyte, so completing the circuit.

The electrons and ion flows because of the chemical reactions happening inside the battery-usually two of them going on simultaneously. The exact reaction depend on the materials from which the electrodes and electrolyte are made. Whatever chemical reactions take place, the general principle of electrons going around the outer circuit, and ions reacting with the electrolyte, applies to all batteries. As a battery generates power, the chemicals inside it are gradually converted into different chemicals. Their ability to generate power dwindles, the batteries voltage slowly falls, and the battery eventually runs flat. In other words, if the battery cannot produce positive ions because the chemicals inside it have become depleted, it can’t produce electrons for the outer circuit either.

The another section of battery for the PIC microcontroller, here we used four batteries of 1.5V. The battery contained a conducting solution (electrolyte) of ammonium chloride, a cathode (positive terminal) of carbon, a depolarizer of manganese dioxide (oxidizer), and an anode ( negative terminal) of zinc (reluctant). Chemistry of this cell manufactured by dry cell.

**Construction**

The original form of the cell used a porous pot. This gave it a relatively high internal resistance and various modifications were made to reduce it. These included the “ Agglomerate block cell” and the “Sack cell”. Leclanche first , and Carl Gassner latter, strived to both to transform the original wet cell in to a more portable and more efficient dry cell.

**Porous pot:** In leclanche’s original cell the depolarizer ( in fact, the oxidizing agent in the cell), which consist of crushed manganese dioxide, was packed in to pot, and a carbon rod was inserted to act as the cathode(reduction reaction). The anode (oxidation reaction), which was a zinc rod, was then immersed along with the pot in a solution of ammonium chloride, This liquid solution act as the electrolyte.

**Agglomerate block cell:** In 1871 leclanche dispersed with the porous pot and replaced it with a pair of agglomerate blocks, attached to the carbon plates by rubber bands. These blocks were made by mixing the manganese dioxide with binding agents and pressing the mixture into moulds.

**Sack cell:** In this cell porous pot was replaced by a wrapping of canvas or sacking. In addition, the zinc rod was replaced by a zinc cylinder to give a larger surface area. It had a lower internal resistance than either of the above (porous and agglomerate).

**Starch addition:** In 1876. George Leclanche added starch to the ammonium chloride electrolyte in an effort to better jellifies it.

**Improved dry cell:** In 1888, a German physician, Carl Gassner improved the jellification process and produced a more portable dry cell by mixing plaster and hydrophilic chemicals with the ammonium chloride electrolyte.

****

Fig2.2. 2.3 1.5v battery

**2.2.3. UART (Universal Asynchronous Receiver/Transmitter)**

UART stands for universal asynchronous receiver transmitter. It’s not a communication protocol like SPI and 12C, but a physical circuit in a microcontroller, or a stand -alone IC. A UART main purpose is to transmit and receive serial data.

One of the best thing about UART is that it only uses to wires to transmit data between devices.

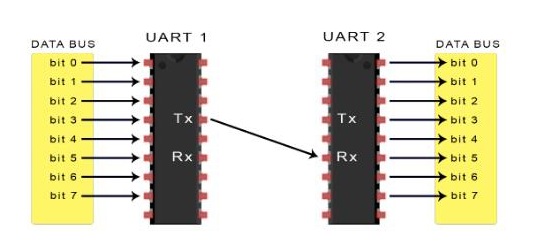


Fig 2.2.3.1 UART

**Introduction to UART communication**

In UART communication, two UARTs communicate directly with each other. The transmitting UART converts parallel data from a controlling device like CPU in to serial form, transmits it in serial to the receiving UART, which then converts the serial data back in to parallel data for the receiving device. Only two wires are needed to transmit data between two UARTs. Data flows from the Tx pin of the transmitting UART to the Rx pin of the receiving UART.

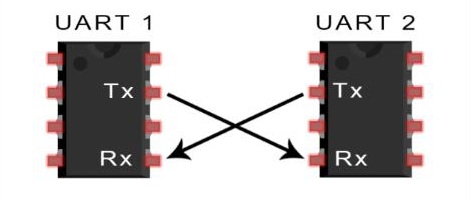


Fig 2.2.3.2 Data flow in UART

UARTs transfer data asynchronously, which means there is no clock signal to synchronize the output of bits from the transmitting UART to the sampling of bits by the receiving UART. Instead of clock signal, the transmitting UART adds start and stop bits to the data packet being transferred. These bits define the beginning and end of the data packet so the receiving UART knows to start reading the bits.

When the receiving UART detects a start bit, it starts to read the incoming bits at a specific frequency known as baud rate. Baud rate is a measure of the speed of the data transfer, expressed in bits per second (bps). Both UART must operate at about the same baud rate. The baud rate between the transmitting and receiving UART can only differ by about 10% before the timing of bits gets too far off. Both UARTs must also must be configured to transmit and receive the same data packet structure.

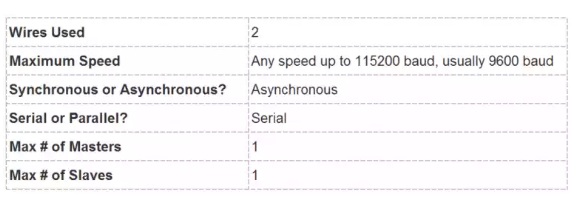
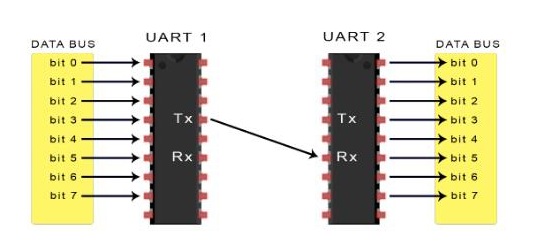


Table 2.2.3.1 UART feature

**WORKING OF UART**

The UARTs that is going to transmit data receives the data from the data bus. The data bus is used to sent data to the UART by another device like CPU, memory, or microcontroller. Data is transferred from the data bus to the transmitting UART in parallel form. After the transmitting UART gets the parallel data from the data bus, it adds a start bit, a parity bit, and a stop bit, creating the data packet. Next, the data packet is output serially, bit by bit at the Tx pin. The receiving UART reads the data packet bit by bits at Rx pin. The receiving UART then converts the data back into parallel form and removes the start bit, parity bit, and stop bits. Finally the receiving UART transfers the data packet in parallel tothe data bus on the receiving end.

 Fig2.2.3.3 flow of parallel data from data bus and parallel data to data bus

UART transmitted data is organized into packets. Each packet contains 1 start bit, 5 to 9 data bits (depending upon the UART), an optional parity bit, 1 or 2 stop bits.

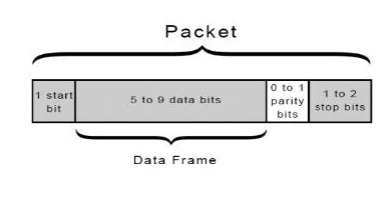


Fig 2.2.3.4 data packet

**START BIT**

The UART data transmission line is normally held at a high voltage level when it’s not transmitting data. To start the transfer of data, the transmitting UART pulls the transmission line from high to low for one clock cycle. When the UART detects the high to low voltage transition, it begins reading the bits in the data frame at the frequency of the baud rate.

**DATA FRAME**

The data frame contains the actual data begins transferred. It can be 5bits upto8 bits long if a parity bit is used. if no parity bit is used, the data frame can be 9 bits long. In most cases, the data is sent with the least significant bit first.

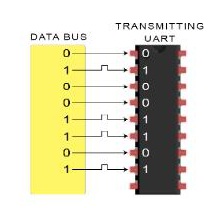
**PARITY**

Parity describes the evenness or oddness of the number. The parity bit is a way for the receiving UART to tell if any data has changed during transmission. Bits can be changed by electromagnetic radiation, mismatched baud rates, or long distance data transfers. After the receiving UART reads the data frame, it counts the number of bits with a value of 1 checks if the total is an even or odd number. If the parity bit is 0 (even parity), the 1 bits in the data frame should total to an even number. If the parity bit is a 1 (odd parity), the 1 bits in the data frame should total to an odd number. When the parity bit matches the data, the UART knows that the transmission was free of errors. But the parity bit is a 0,and the total is odd; or the parity bit is a 1, and the total is even, the UART knows that bits in the data frame have changed.

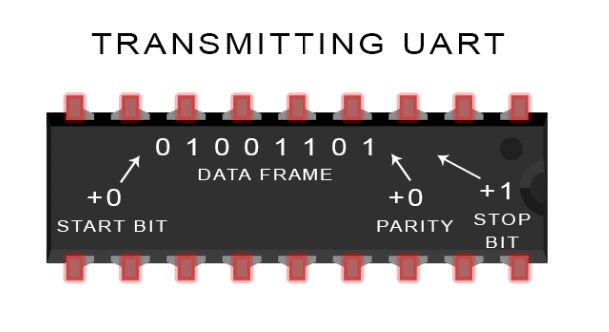
**STOP BITS**

To signal the end of the data packet, the sending UART drives the data transmission line from a low voltage to a high voltage for at least two bit durations.

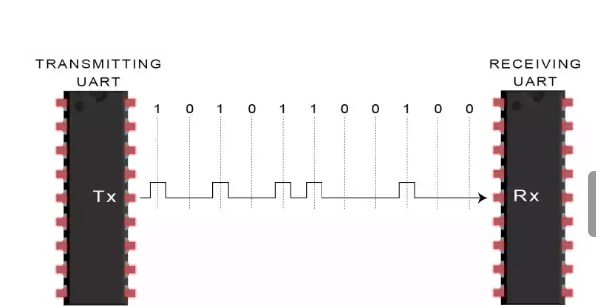
**STEPS OF UART TRANSMISSION**

1. The transmitting UART receives data in parallel from the data bus: ****

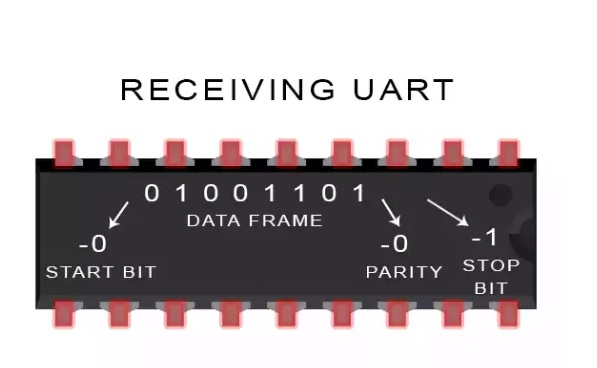
2.The transmitting UART adds the start bit, parity bit, and the stop bits to the data frame:



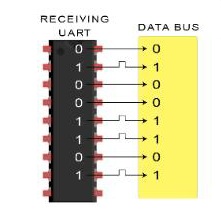
3.The entire packet is sent serially from the transmitting UART. The receiving UART samples the data line at the pre-configured baud rate:



4.The receiving UART discards the start bit, parity bit, and stop bit from the data frame.:



5.The receiving UART converts the serial data back in to parallel and transfer it to the data bus on the receiving end:



**ADVANTAGES AND DISADVANTAGES OF UARTS**

**Advantages:**

1. Only uses two wires
2. No clock signal is necessary
3. Has a parity bit to allow for error checking.
4. The structure of the data packet can be changed as long as both sides are set up for it.
5. Well documented and widely used method

**Disadvantages:**

1. The size of the data frame is limited to a maximum of nine bits
2. Does not support multiple slave or multiple master system
3. The baud rate of each UART must be within 10% of each other

**2.2.4. BLUETOOTH MODULE**

Bluetooth is a wireless protocol utilizing short-range communications technology facilitating data transmission over short distances from fixed and/or mobile devices, creating wireless personal area networks (PANs). The intent behind the development of Bluetooth was the creation of a single digital wireless protocol, capable of connecting multiple devices and overcoming issues arising from synchronization of these devices.

Bluetooth uses a very robust radio technology called frequency hopping spread spectrum. It chops up the data being sent and transmits chunks of it on up to 75 different frequencies. In its basic mode, the modulation is Gaussian frequency shift keying (GFSK). It can achieve a gross data rate of 1 Mb/s. Bluetooth provides a way to connect and exchange information between devices such as mobile phones, telephones, laptops, personal computers, printers, GPS receivers, digital cameras, and video game consoles over a secure, globally unlicensed Industrial, Scientific, and Medical (ISM) 2.4 GHz short-range radio frequency bandwidth.

The Bluetooth specifications are developed and licensed by the Bluetooth Special Interest Group (SIG). The Bluetooth SIG consists of companies in the areas of telecommunication, computing, networking, and consumer electronics.Bluetooth is a standard and communications protocol primarily designed for low power consumption, with a short range (power-class-dependent: 1 meter, 10 meters, 100 meters) based on low-cost transceiver microchips in each device.

Bluetooth enables these devices to communicate with each other when they are in range. The devices use a radio communications system, so they do not have to be in line of sight of each other, and can even be in other rooms, as long as the received transmission is powerful enough. Bluetooth device class indicates the type of device and the supported services of which the information is transmitted during the discovery process.

|  |  |  |
| --- | --- | --- |
| **Class** | **Maximum Permitted Power mW(dBm)** | **Range (approximate)** |
| **Class 1** | 100 mW (20 dBm) | ~100 meters |
| **Class 2** | 2.5 mW (4 dBm) | ~10 meters |
| **Class 3** | 1 mW (0 dBm) | ~1 meter |

Table2.2.4.1 Bluetooth max permitted power range

In most cases the effective range of class 2 devices is extended if they connect to a class 1 transceiver, compared to pure class 2 network. This is accomplished by the higher sensitivity and transmission power of Class 1 devices.

|  |  |
| --- | --- |
| **Version** | **Data Rate** |
| **Version 1.2** | 1 Mbit/s |
| **Version 2.0 + EDR** | 3 Mbit/s |
| **WiMedia Alliance (proposed)** | * + 1. - 480 Mbit/s |

Table2.2.4.2 Bluetooth versions

**BLUETOOTH PROFILES**

In order to use Bluetooth, a device must be compatible with certain Bluetooth profiles. These define the possible applications and uses of the technology.

**BLUETOOTH VS WI-FI IN NETWORKING**

Bluetooth and Wi-Fi have different applications in today's offices, homes, and on the move: setting up networks, printing, or transferring presentations and files from PDAs to computers. Both are versions of unlicensed wireless technology. Wi-fi differs from Bluetooth in that it provides higher throughput and covers greater distances, but requires more expensive hardware and may present higher power consumption. They use the same frequency range, but employ different modulation techniques. While Bluetooth is a replacement for cabling in a variety of small-scale applications, Wi-Fi is a replacement for cabling for general local area network access. Bluetooth can be taken as replacement for USB or any other serial cable link, whereas Wi-Fi is wireless Ethernet communications according to the protocol architectures of IEEE 802.3 with TCP/IP. Both standards are operating at a specified bandwidth not identical with that of other networking standards; the mechanical plug compatibility problem known with cables is replaced by the compatibility requirement for an air interface and a protocol stack.

#### BLUETOOTH DEVICES:

Bluetooth exists in many products, such as telephones, printers, modems and headsets. The technology is useful when transferring information between two or more devices that are near each other in low-bandwidth situations. Bluetooth is commonly used to transfer sound data with telephones (i.e. with a Bluetooth headset) or byte data with hand-held computers (transferring files).Bluetooth protocols simplify the discovery and setup of services between devices. Bluetooth devices can advertise all of the services they provide. This makes using services easier because more of the security, network address and permission configuration can be automated than with many other network types.

### OPERATING SYSTEM SUPPORT:

Apple has supported Bluetooth since Mac OS X v10.2 which was released in 2002. For Microsoft platforms, Windows XP Service Pack 2 and later releases have native support for Bluetooth. Linux has two popular Bluetooth stacks, BlueZ and Affix. The BlueZ stack is included with most Linux kernels and it was originally developed by Qualcomm. The Affix stack was developed by Nokia. FreeBSD features Bluetooth support since its 5.0 release. NetBSD features Bluetooth support since its 4.0 release. Its Bluetooth stack has been ported to OpenBSD as well.

## MOBILE PHONE REQUIREMENTS:

A mobile phone that is Bluetooth enabled is able to pair with many devices. To ensure the broadest support of feature functionality together with legacy device support. The OMTP forum has recently published a recommendations paper, entitled "Bluetooth Local Connectivity", see external links below to download this paper.This publication recommends two classes, basic and advanced, with requirements that cover imaging, printing, stereo audio and in car usage.

### COMMUNICATION AND CONNECTION:

A master Bluetooth device can communicate with up to seven devices. This network group of up to eight devices is called a piconet. A piconet is an ad-hoc computer network, using Bluetooth technology protocols to allow one master device to interconnect with up to seven active devices. Up to 255 further devices can be inactive, or parked, which the master device can bring into active status at any time.

At any given time, data can be transferred between the master and one other device, however, the devices can switch roles and the slave can become the master at any time. The master switches rapidly from one device to another in a round-robin fashion. (Simultaneous transmission from the master to multiple other devices is possible, but not used much.)Bluetooth specification allows connecting two or more piconets together to form a scatter net, with some devices acting as a bridge by simultaneously playing the master role and the slave role in one piconet.Many USB Bluetooth adapters are available, some of which also include an IrDA adapter. Older (pre-2003) Bluetooth adapters, however, have limited services, offering only the Bluetooth Enumerator and a less-powerful Bluetooth Radio incarnation. Such devices can link computers with Bluetooth, but they do not offer much in the way of services that modern adapters do.

### SETTING UP CONNECTIONS:

Any Bluetooth device will transmit the following information on demand:

* Device name.
* Device class.
* List of services.
* Technical information, for example, device features, manufacturer, Bluetooth specification used, clock offset.

Any device may perform an inquiry to find other devices to connect to, and any device can be configured to respond to such inquiries. However, if the device trying to connect knows the address of the device, it always responds to direct connection requests and transmits the information shown in the list above if requested. Use of device services may require pairing or acceptance by its owner, but the connection itself can be initiated by any device and held until it goes out of range. Some devices can be connected to only one device at a time, and connecting to them prevents them from connecting to other devices and appearing in inquiries until they disconnect from the other device.Every device has a unique 48-bit address. However these addresses are generally not shown in inquiries. Instead, friendly Bluetooth names are used, which can be set by the user. This name appears when another user scans for devices and in lists of paired devices.Most phones have the Bluetooth name set to the manufacturer and model of the phone by default. Most phones and laptops show only the Bluetooth names and special programs that are required to get additional information about remote devices. This can be confusing as, for example, there could be several phones in range named T610 (see Bluejacking).

### PAIRING:

Pairs of devices may establish a trusted relationship by learning (by user input) a shared secret known as a passkey. A device that wants to communicate only with a trusted device can cryptographically authenticate the identity of the other device. Trusted devices may also encrypt the data that they exchange over the airwaves so that no one can listen in. The encryption can, however, be turned off, and passkeys are stored on the device file system, not on the Bluetooth chip itself. Since the Bluetooth address is permanent, a pairing is preserved, even if the Bluetooth name is changed. Pairs can be deleted at any time by either device. Devices generally require pairing or prompt the owner before they allow a remote device to use any or most of their services. Some devices, such as mobile phones, usually accept OBEX business cards and notes without any pairing or prompts.

### AIR INTERFACE:

The protocol operates in the license-free ISM band at 2.4-2.4835 GHz. To avoid interfering with other protocols that use the 2.45 GHz band, the Bluetooth protocol divides the band into 79 channels (each 1 MHz wide) and changes channels up to 1600 times per second. Implementations with versions 1.1 and 1.2 reach speeds of 723.1 kbit/s. Version 2.0 implementations feature Bluetooth Enhanced Data Rate (EDR) and reach 2.1 Mbit/s. Technically, version 2.0 devices have a higher power consumption, but the three times faster rate reduces the transmission times, effectively reducing power consumption to half that of 1.x devices (assuming equal traffic load).

**HC SERIAL BLUETOOTH PRODUCTS**

HC serial Bluetooth products consist of Bluetooth serial interface module and Bluetooth adapter, such as:

(1) Bluetooth serial interface module:

Industrial level: HC-03, HC-04(HC-04-M, HC-04-S)

Civil level: HC-05, HC-06(HC-06-M, HC-06-S)

HC-05-D, HC-06-D (with baseboard, for test and evaluation)

(2) Bluetooth adapter:

HC-M4

HC-M6

This document mainly introduces Bluetooth serial module. Bluetooth serial module is used for converting serial port to Bluetooth. These modules have two modes: master and slaver device. The device named after even number is defined to be master or slaver when out of factory and can’t be changed to the other mode. But for the device named after odd number, users can set the work mode (master or slaver) of the device by AT commands.

The main function of Bluetooth serial module is replacing the serial port line, such as:

* 1. There are two MCUs want to communicate with each other. One connects to Bluetooth master device while the other one connects to slave device. Their connection can be built once the pair is made. This Bluetooth connection is equivalently liked to a serial port line connection including RXD, TXD signals. And they can use the Bluetooth serial module to communicate with each other.
  2. When MCU has Bluetooth salve module, it can communicate with Bluetooth adapter of computers and smart phones. Then there is a virtual communicable serial port line between MCU and computer or smart phone.
  3. The Bluetooth devices in the market mostly are salve devices, such as Bluetooth printer, Bluetooth GPS. So, we can use master module to make pair and communicate with them. Bluetooth Serial module’s operation doesn’t need drive, and can communicate with the other Bluetooth device that has the serial. But communication between two Bluetooth modules requires at least two conditions:

(1) The communication must be between master and slave.

(2) The password must be correct.

However, the two conditions are not sufficient conditions. There are also some other conditions basing on different device model.

**SELECTION OF THE MODULE**

The Bluetooth serial module named even number is compatible with each other; The salve module is also compatible with each other. In other word, the function of HC-04 and HC-06, HC-03 and HC-05 are mutually compatible with each other. HC-04 and HC-06 are former version that user can’t reset the work mode (master or slave). And only a few AT commands and functions can be used, like reset the name of Bluetooth (only the slaver), reset the password, reset the baud rate and check the version number. The command set of HC-03 and HC-05 are more flexible than HC-04 and HC-06’s. Generally, the Bluetooth of HC-03/HC-05 is recommended for the user.



Fig 2.2.4.2 HC-05

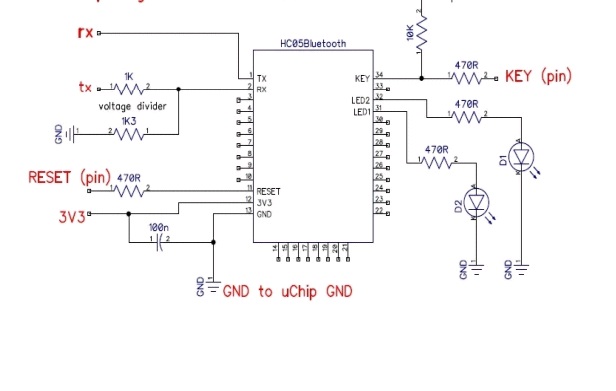
HC-05 has many functions and covers all functions of HC-06. The above commands are the most common ones. Besides this, HC-05 leaves lots of space for user. So HC-05 is better than HC-06 and recommended. HC-03 is similar with HC-05.

Fig 2.2.4.3 Internal Architecture of HC-05

|  |  |
| --- | --- |
| PIN 1 | UART\_TXD, Bluetooth serial signal sending PIN, can connect with MCU’s RXD PIN |
| PIN2 | UART\_RXD, Bluetooth serial signal receiving PIN, can connect with the MCU’s TXD PIN; there is no pull-up resistor in this PIN. But It needs to be added an eternal pull-up resistor. |
| PIN11 | RESET, the reset PIN of module, inputting low level can reset the module, when the module is in using, this PIN can connect to air. |
| PIN12 | VCC, voltage supply for logic, the standard voltage is 3.3V, and can work at 3.0-4.2V |
| PIN13 | GND |
| PIN31 | LED1, indicator of work mode. Has 3 modes:  When the module is supplied power and PIN34 is input high level, PIN31 output 1Hz square wave to make the LED flicker slowly. It indicates that the module is at the AT mode, and the baud rate is 38400;  When the module is supplied power and PIN34 is input low level, PIN31 output 2Hz square wave to make the LED flicker quickly. It indicates the module is at the pair able mode. IfPIN34 is input high level, then the module will enter to AT mode, but the output of PIN31 is still 2Hz square wave.  After the pairing, PIN31 output 2Hz square ware.  Note: if PIN34 keep high level, all the commands in the AT command set can be in application. Otherwise, if just excite PIN34 with high level but not keep; only some command can be used. More information has provided at chapter 2. |
| PIN32 | Output terminal. Before paired, it output low level. Once the pair is finished, it output high level. |
| PIN34 | Mode switch input. If it is input low level, the module is at paired or communication mode. If it’s input high level, the module will enter to AT mode. Even though the module is at communication, the module can enter to the AT mode if PIN34 is input high level. Then it will go back to the communication mode if PIN34 is input low level again. |

Table 2.2.4.3 pin description of HC-05

**FEATURES:**

* Wireless serial Bluetooth port.
* With free power adapter bottom board come with well power regulator.
* User can connect 3.3 to 5VDC and connect TX and RX to your control IO (general 3.3 to 5V digital input output of MCU or  IO is ok, or general TLL IO)
* Easy to connect this module with PC, just search and key "1234" pass code.
* With white SMD LED on the adapter board, can see the Bluetooth connection status.

**STEP TO CONNECT:**

* Connect the wiring, power up, while the device is not connected, the Bluetooth module board has a white LED flashing
* At PC side, search Bluetooth device.
* Found name called "HC-05" device
* Connect it, and pass code is "1234"
* While connection is ok, you can see the LED become always on

**USAGE:**

* Coupled Mode: Two modules will establish communication automatically when powered.
* PC hosted mode: Pair the module with Bluetooth dongle directly as virtual serial.
* Bluetooth protocol :  Bluetooth Specification v2.0+EDR
* Frequency :  2.4GHz ISM band
* Modulation :  GFSK(Gaussian Frequency Shift Keying)
* Speed : Asynchronous:  2.1Mbps(Max) / 160 kbps, Synchronous: 1Mbps/1Mbps
* Security :  Authentication and encryption
* Profiles :  Bluetooth serial port
* CSR chip : Bluetooth v2.0
* Wave band : 2.4GHz-2.8GHz, ISM Band
* Protocol : Bluetooth V2.0
* Voltage : 5V (3.6V-6V, NO more than 7V)
* User defined Baud rate:  4800, 9600, 19200, 38400, 57600, 115200, 230400,460800,921600, and 1382400.

**2.2.5 FINGER PRINT MODULE**

Finger print sensor is combination of R305 FP+PIC+MCU board that can read different finger prints and store its on flash memory. The sensor can perform three functions namely Add (enrol), empty database or search database and return the ID of stored finger print.

Any of the three functions can be called simply by making the pin low of the sensor or pressing onboard three switches. The response is either error or ok which is indicated by onboard LED. The response is also returned as single serial data byte. The return byte is valid ID or error code. The response byte is a single byte at 9600 bps thus making whole sensor very easy to use. We have provided indicating LEDs and function switch already so it’s ready to use when you receive it. Just give a power and start using the sensor using onboard switches. Then you can move on making external application using these functions.

Fingerprint is a narrow sense is an impression left by the friction ridges of a human finger. The recovery of finger prints from a crime sense is an important method of forensic science. Finger prints are easily deposited on suitable surface ( such as glass or metal or polished stone) by the natural secretions of sweat from the eccrine glands that are present in epidermal ridges. These are sometimes referred to as “chanced impressions”.

In an wider use of the term, fingerprints are the trace of an impression from the friction ridges of any part of a human or primate hand. A print from the sole of the foot can also leave an impression of friction ridges

Deliberate impressions of fingerprints may be formed by ink or other substances transferred from the peaks of friction ridges on the skin to a relatively smooth surface such as finger print card. Fingerprints records normally contain impressions from the pad on the last joint for fingers and thumbs, although fingerprint cards also typically record portions of lower joint areas of fingers.

Human fingerprints are detailed, nearly unique, difficult to alter, and durable over the life of an individual, making them suitable as long term markers of human identity. They may be employed by police or other authorities to identify individuals who wish to conceal their identity**.**

**BIOLOGICAL ASPECTS**

A friction ridge is a raised portion of the epidermis on the digits (fingers and toes), the palm of the hand or the sole of the foot, consisting of one or more connected ridge units of friction ridge skin. These are sometimes known as “epidermal ridges” which are caused by the underlying interface between the dermal papillae of the dermis and the inter papillary pegs of the epidermis. These epidermal ridges serve to amplify vibration triggered. These ridges may also assist in gripping rough surfaces and may improve surface contact in wet conditions.



Fig2.2.5.1 friction ridges on a finger

**FINGERPRINT PROCESSING**

Fingerprint processing has three primary functions: enrolment, searching and verification. Among these functions, enrolment which captures fingerprint image from the sensor plays an important role. A reason is that the way people put their fingerprints on a mirror to scan can effect to the result in the searching and verifying process. Regarding to verification function, there are several techniques to match fingerprints such as correlation-based matching, minutiae-based matching, ridge feature-based matching and minutiae-based algorithm. However, the most popular algorithm was minutiae based matching algorithm due to its efficiency and accuracy.

**Input: Two ways to trigger the function of finger print sensor**

1. Onboard switch: Add, Empty or Search.
2. Make pin low from external microcontroller for 5ms as per function required to be executed.

**Outputs (Response):** Two ways to monitor output response after a function is executed

1. Onboard LEDs: ERROR or OK
2. Reads byte after executing function

**TYPES OF FUNCTION**

There are namely three functions you can call for the fingerprint sensor.

**Add (Enrol) Function:** Adds a fingerprint to database and return a byte of newly added ID. Return values are from 0\*00 to 0\*FE. In case of error like no finger placed, return code is 0\*FF. Here 0\*FF means error executing function.

**Search Function:** When a finger is put and search functions is called, it returns a matching ID if found in its existing memory. Return values are from 0\*00 to 0\*FE. In case of error like no finger placed, return code is 0\*FF. Here 0\*FF means error executing function.

**Empty Function:** When you wish to empty all fingerprint data stored on sensor you can use this function. After executing this function, you will get 0\*CC as OK or 0\*FF in case of error.

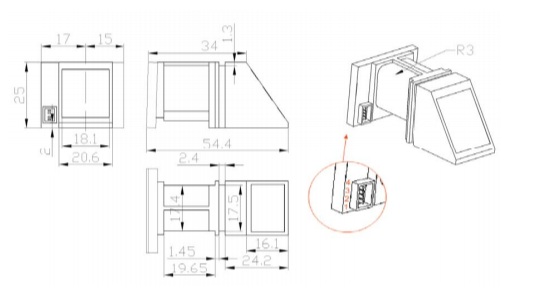


Fig 2.2.5.2 Fingerprint exterior interface

**FEATURES**

1. Easy to use

2. Status LEDs

3. Function Switches

4. Single byte response

5. Works at 5V

6. UART 9600bps response

* + 1. **MOTOR DRIVER**

L293D is a dual H-bridge motor driver integrated circuit (IC). Motor drivers act as current amplifiers since they take a low-current control signal and provide a higher-current signal. This higher current signal is used to drive the motors.L293D contains two inbuilt H-bridge driver circuits. In its common mode of operation, two DC motors can be driven simultaneously, both in forward and reverse direction. The motor operations of two motors can be controlled by input logic at pins 2 & 7 and 10 & 15. Input logic 00 or 11 will stop the corresponding motor. Logic 01 and 10 will rotate it in clockwise and anticlockwise directions, respectively.

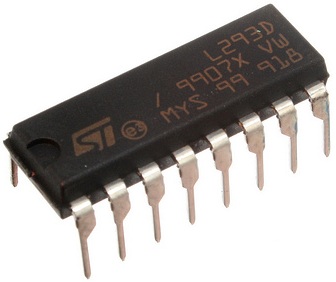


Fig 2.2.6.1 L293D IC

Enable pins 1 and 9 (corresponding to the two motors) must be high for motors to start operating. When an enable input is high, the associated driver gets enabled. As a result, the outputs become active and work in phase with their inputs. Similarly, when the enable input is low, that driver is disabled, and their outputs are off and in the high-impedance state.L293D is a dual H-Bridge motor driver, So with one IC we can interface two DC motors which can be controlled in both clockwise and counter clockwise direction and if you have motor with fix direction of motion the you can make use of all the four I/Os to connect up to four DC motors. L293D has output current of 600mA and peak output current of 1.2A per channel. Moreover for protection of circuit from back EMF output diodes are included within theIC. The output supply (VCC2) has a wide range from 4.5V to 36V, which has made L293D a best choice for DC motor driver. A simple schematic for interfacing a DC motor using L293D is shown below.

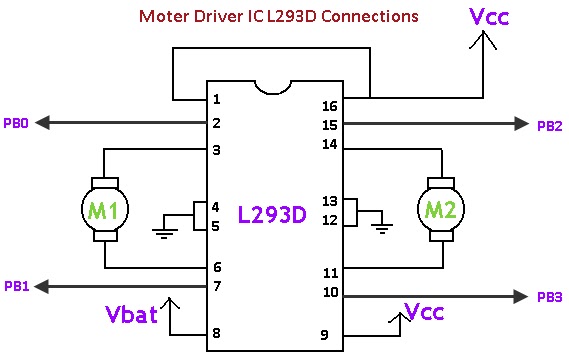


Fig 2.2.6.2 schematic interfacing of L293D

#### ****MOTOR DRIVER AND H-BRIDGE BASICS****

Generally, even the simplest robot requires a motor to rotate a wheel or performs particular action. Since motors require more current then the microcontroller pin can typically generate, you need some type of a switch (Transistors, MOSFET, Relay etc.,) which can accept a small current, amplify it and generate a larger current, which further drives a motor. This entire process is done by what is known as a **motor driver.** Motor driver is basically a current amplifier which takes a low-current signal from the microcontroller and gives out a proportionally higher current signal which can control and drive a motor. In most cases, a transistor can act as a switch and perform this task which drives the motor in a single direction.

Turning a motor ON and OFF requires only one switch to control a single motor in a single direction. What if you want your motor to reverse its direction? The simple answer is to reverse its polarity. This can be achieved by using four switches that are arranged in an intelligent manner such that the circuit not only drives the motor, but also controls its direction. Out of many, one of the most common and clever design is a H-bridge circuit where transistors are arranged in a shape that resembles the English alphabet "H".

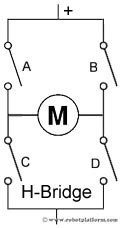


Fig 2.2.6.3 H-bridge motor connection

As you can see in the image, the circuit has four switches A, B, C and D. Turning these switches ON and OFF can drive a motor in different ways.

* Turning on Switches **A** and **D** makes the motor rotate clockwise
* Turning on Switches **B** and **C** makes the motor rotate anti-clockwise
* Turning on Switches **A** and **B** will stop the motor (Brakes)
* Turning off all the switches gives the motor a free wheel drive
* Lastly turning on **A** &**C** at the same time or**B** & **D** at the same time shorts your entire circuit. So, do not attempt this.

H-bridges can be built from scratch using relays, MOSFETs, field effect transistors (FET), bi-polar junction transistors (BJT), etc. But if your current requirement is not too high and all you need is a single package which does the job of driving a small DC motor in two directions, then all you need is a L293D IC. This single inexpensive package can interface not one, but two DC motors. L293, L293B and few other versions also does the same job, but pick the L293D version as this one has an inbuilt fly back diode which protects the driving transistors from voltage spikes that occur when the motor coil is turned off.

##### **L293D CONNECTIONS**

The circuit shown to the right is the most basic implementation of L293D IC. There are 16 pins sticking out of this IC and we have to understand the functionality of each pin before implementing this in a circuit.

* Pin1 and Pin9 are "Enable" pins. They should be connected to +5V for the drivers to function. If they pulled low (GND), then the outputs will be turned off regardless of the input states, stopping the motors. If you have two spare pins in your microcontroller, connect these pins to the microcontroller, or just connect them to regulated positive 5 Volts.
* Pin4, Pin5, Pin12 and Pin13 are ground pins which should ideally be connected to microcontroller's ground.
* Pin2, Pin7, Pin10 and Pin15 are logic input pins. These are control pins which should be connected to microcontroller pins. Pin2 and Pin7 control the first motor (left); Pin10 and Pin15 control the second motor (right).
* Pin3, Pin6, Pin11, and Pin14 are output pins. Tie Pin3 and Pin6 to the first motor, Pin11 and Pin14 to second motor, Pin16 powers the IC and it should be connected to regulated +5Volts
* Pin8 powers the two motors and should be connected to positive lead of a secondary battery. As per the datasheet, supply voltage can be as high as 36 Volts.

##### **TRUTH TABLE**

##### I have shown you where to connect the motors, battery and the [microcontroller](http://www.robotplatform.com/electronics/microcontroller/microcontroller.html). But how do we control the direction of these motors? Let us take an example: Suppose you need to control the left motor which is connected to Pin3 (O1) and Pin6 (O2). As mentioned above, we require three pins to control this motor - Pin1 (E1), Pin2 (I1) and Pin7 (I2). Here is the truth table representing the functionality of this motor driver.

|  |  |  |  |
| --- | --- | --- | --- |
| Pin 1 | Pin 2 | Pin 7 | Function |
| **High** | **High** | **Low** | **Turn Anti-clockwise (Reverse)** |
| **High** | **Low** | **High** | **Turn clockwise (Forward)** |
| **High** | **High** | **High** | **Stop** |
| **High** | **Low** | **Low** | **Stop** |
| **Low** | **X** | **X** | **Stop** |

Table2. 2.6.1: Truth Table.

High ~+5V, Low ~0V, X=either high or low (don't care)

In the above truth table you can observe that if Pin1 (E1) is low then the motor stops, irrespective of the states on Pin2 and Pin7. Hence it is essential to hold E1 high for the driver to function, or simply connect enable pins to positive 5 volts. With Pin1 high, if Pin2 is set high and Pin7 is a pulled low, then current flow from Pin2 to Pin7 driving the motor in anti-clockwise direction. If the states of Pin2 and Pin7 are flipped, then current flows from Pin7 to Pin2 driving the motor in clockwise direction.

## VOLTAGE SPECIFICATION

VCC is the voltage that it needs for its own internal operation 5v; l293D will not use this voltage for driving the motor. For driving the motor it has a separate provision to provide motor supply VSS (V supply).  It means if you want to operate a motor at 9V then you need to provide a Supply of 9V across VSS Motor supply.The maximum voltage for VSS motor supply is 36V. It can supply a max current of 600mA per channel. Since it can drive motors Up to 36v hence you can drive pretty big motors with this l293d.VCC pin 16 is the voltage for its own internal Operation. The maximum voltage ranges from 5v and up to 36v.

**2.2.7 DC MOTOR**

A DC motor is designed to run on DC electric power. Two examples of pure DC designs are Michael Faraday's homopolar motor (which is uncommon), and the ball bear in motor, which is (so far) a novelty. By far the most common DC motor types are the brushed and brushless types, which use internal and external commutation respectively to create an oscillating AC current from the DC source -- so they are not purely DC machines in a strict sense.

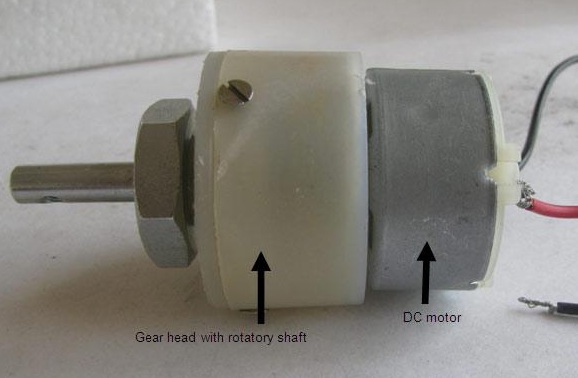


Fig 2.2.7.1: DC Motor

**TYPES OF DCMOTORS:**

* Brushed DC Motors
* Brushless DC motors
* Coreless DC motors

**BRUSHED DC MOTORS:**

The classic DC motor design generates an oscillating current in a wound rotor with a split ring commutator, and either a wound or permanent magnet stator. A rotor consists of a coil wound around a rotor which is then powered by any type of battery. Many of the limitations of the classic commutator. DC motor are due to the need for brushes to press against the commutator. This creates friction. At higher speeds, brushes have increasing difficulty in maintaining contact. Brushes may bounce off the irregularities in the commutator surface, creating sparks. This limits the maximum speed of the machine. The current density per unit area of the brushes limits the output of the motor.

### BRUSHLESS DC MOTORS:

Some of the problems of the brushed DC motor are eliminated in the brushless design. In this motor, the mechanical "rotating switch" or commutator/brushgear assembly is replaced by an external electronic switch synchronized to the rotor's position. Brushless motors are typically 85-90% efficient, whereas DC motors with brush gear are typically 75-80% efficient. Midway between ordinary DC motors and stepper motors lies the realm of the brushless DC motor. Built in a fashion very similar to stepper motors, these often use a permanent magnet external rotor, three phases of driving coils, one or more Hall Effect sensors to sense the position of the rotor, and the associated drive electronics.

**CORELESS DC MOTORS:**

Nothing in the design of any of the motors described above requires that the iron (steel) portions of the rotor actually rotate; torque is exerted only on the windings of the electromagnets. Taking advantage of this fact is the coreless DC motor, a specialized form of a brush or brushless DC motor. Optimized for rapid acceleration, these motors have a rotor that is constructed without any iron core. The windings are typically stabilized by being impregnated with Electrical epoxy potting systems.

# WORKING OR OPERATING PRINCIPLE OF DC MOTOR

A DC motor in simple words is a device that converts direct current (electrical energy) into mechanical energy. It’s of vital importance for the industry today, and is equally important for engineers to look into the **working principle of DC motor**in details that has been discussed in this article. In order to understand the **operating principle of dc motor** we need to first look into its constructional feature. The very basic construction contains a current carrying armature which is connected to the supply end through commutator segments and brushes and placed within the north south poles of a permanent or an electro-magnet as shown in the diagram below.

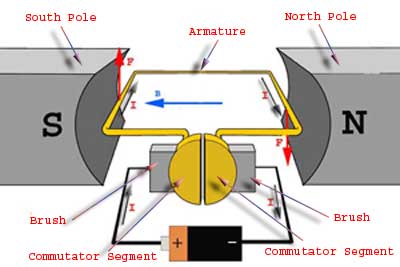


Fig 2.2.7.2 Operation of DC Motor.

For clear understanding the **principle of DC motor** we have to determine the magnitude of the force, by considering the diagram below.

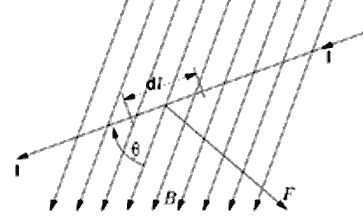


Fig 2.2.7.3: Magnitude of the force for DC Motor.

We know that when an infinitely small charge dq is made to flow at a velocity ‘v’ under the influence of an electric field E, and a magnetic field B, then the Lorentz Force dF experienced by the charge is given by:-

dF = dq(E + v X B)  
For the **operation of dc motor**, considering E = 0  
∴dF = dq v X B  
i.e. it’s the cross product of dq v and magnetic field B.  
or dF = dq (dL/dt) X B        [v = dL/dt]  
Where dL is the length of the conductor carrying charge q.  
or dF = (dq/dt) dL X B  
or dF = I dL X B        [Since, current I = dq/dt]  
or F = IL X B = ILB Sinθ  
or F = BIL Sinθ

From the 1st diagram we can see that the construction of a DC motor is such that the direction of electric current through the armature conductor at all instance is perpendicular to the field. Hence the force acts on the armature conductor in the direction perpendicular to the both uniform field and current is constant.

i.e. θ = 90°

So if we take the current in the left hand side of the armature conductor to be I, and current at right hand side of the armature conductor to be − I, because they are flowing in the opposite direction with respect to each other.

Then the force on the left hand side armature conductor, Fl = BIL Sin90° = BIL  
Similarly force on the right hand side conductor Fr = B( − I)L.Sin90° = − BIL

∴We can see that at that position the force on either side is equal in magnitude but opposite in direction. And since the two conductors are separated by some distance w = width of the armature turn, the two opposite forces produces a rotational force or a torque that results in the rotation of the armature conductor.Now let's examine the expression of torque when the armature turns crate an angle of α with its initial position.

The torque produced is given by  
Torque = force, tangential to the direction of armature rotation X distance.  
or τ = Fcosα.w  
or τ = BIL w cosα

Where α is the angle between the plane of the armature turn and the plane of reference or the initial position of the armature which is here along the direction of magnetic field. The presence of the term cosα in the torque equation very well signifies that unlike force the torque at all position is not the same. It in fact varies with the variation of the angle α. To explain the variation of torque and the principle behind rotation of the motor let us do a stepwise analysis.

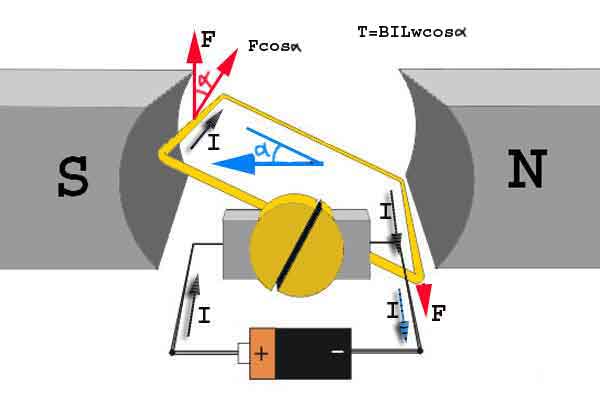


Fig 2.2.7.4: Motor rotating case 1

**Step 1:**  
Initially considering the armature is in its starting point or reference position where the angle α = 0.

∴ τ = BIL w cos0 = BILw

Since α = 0, the term cos α = 1, or the maximum value, hence torque at this position is maximum given by τ = BILw. This high starting torque helps in overcoming the initial inertia of rest of the armature and sets it into rotation.

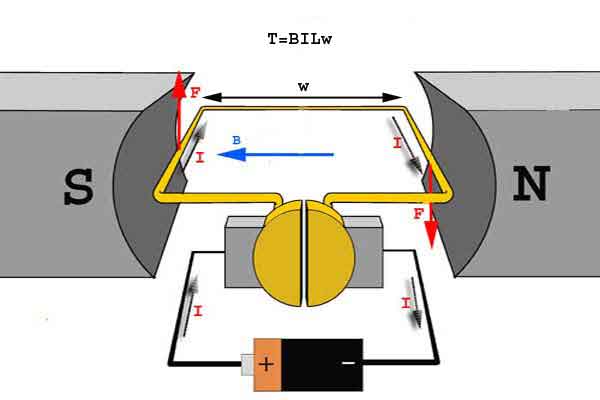


Fig 2.2.7.5: Motor rotating case 2

**Step 2:**

Once the armature is set in motion, the angle α between the actual position of the armature and its reference initial position goes on increasing in the path of its rotation until it becomes 90° from its initial position. Consequently the term cosα decreases and also the value of torque. The torque in this case is given by τ = BILwcosα which is less than BIL w when α is greater than 0°.

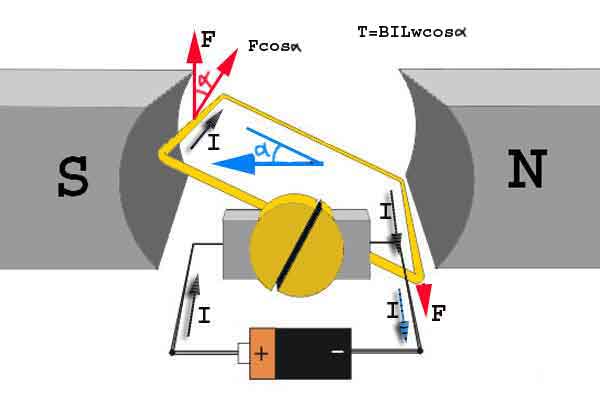


Fig 2.2.7.6: Motor rotating case 3

**Step 3:**

In the path of the rotation of the armature a point is reached where the actual position of the rotor is exactly perpendicular to its initial position, i.e. α = 90°, and as a result the term cosα=0,

The torque acting on the conductor at this position is given by.  
τ = BILwcos90° = 0

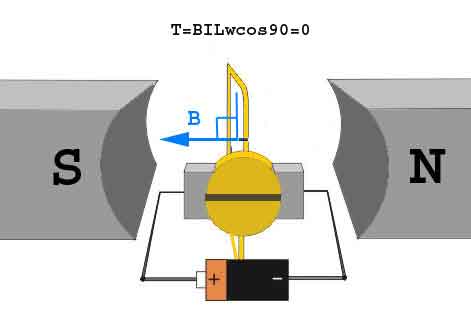


Fig 2.2.7.7: Motor rotating case 4

i.e. virtually no rotating torque acts on the armature at this instance. But still the armature does not come to a standstill; this is because of the fact that the operation of [dc motor](http://www.electrical4u.com/dc-motor-or-direct-current-motor/) has been engineered in such a way that the inertia of motion at this point is just enough to overcome this point of null torque. Once the rotor crosses over this position the angle between the actual position of the armature and the initial plane again decreases and torque starts acting on it again.

**ROBOTIC VEHICLE WORKING MECHANISM**

The working mechanism of the robot is based on the information passed from the android mobile phone via Bluetooth connection to the robot using a Bluetooth modem and vice versa.

When will give a commands by android phone that will transmitt and receive the information signals.by giving a commands it will move in the given command direction.here is the power supply is given to robot by storage batteries. By using batteries we are capable to run the robo.



Fig 2.2.7.8 architecture

In this voice controlled vehicle working is explained that , when finger is enrol the fingerprint scanner, it scan the ridges of the finger and the finger is matched the indication LED is On. If the finger is not matched there is another LED is bling. In the finger print module have the facility of add, search, clear the finger prints. If the finger is add in one time, then the finger is search for the next time entering.

The voice command is speak to the android application mobile, then the language code is converted to the ASCII code by the Google language converter that code is sent to the vehicle receiver UART. The data is transmitted by the Bluetooth device. The mobile’s Bluetooth and vehicles Bluetooth both are paired each other. The data is transmitted in the form of serial communication. When the code is received in the vehicle, then it goes to the microcontroller. The microcontroller’s program is identify the character to be sent and it call the function of operation. By using the logic of operation is goes to the motor driver then the driver is identify the logic and the motors rotate (clockwise or anti clockwise) as per the command logic.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MOTORS | FF | | BWD | | LEF | | RIG | | STOP | |
|  | P1 | P2 | P1 | P2 | P1 | P2 | P1 | P2 | P1 | P2 |
| M1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| M2 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |

Table 2.2.7.2 Working logic

In the above table FF, BWD, LEF, RIG represents the forward, backward, left, right directions respectively. M1 and M2 represent the two motors. The P1 and P2 represent the pins of the motors M1 and M2 respectively.

If the logic (1 0) represent the anti clock wise direction, logic (0 1) represent clock wise direction and the logic (0 0) represent the no rotation. If the vehicle moves to the forward direction the M1 rotate anti clock wise and M2 rotates clock wise direction. If the vehicle moves to backward M1 rotate clock wise and M2 rotate anti clock wise direction. If the vehicle moves to left both M1 and M2 are rotates in clock wise direction. If the vehicle moves to right, both M1 and M2 are rotates in anti clock wise direction. When the vehicle is stop, then the both motors are in rest.

In this project we include facility of type direction characters are in the android mobile that is works at the same function of the voice command.



Fig 2.2.7.9 commanding window

**CHAPTER 3**

**SOFTWAREREQUIREMENT SPECIFICATION**

In this project 3 types of soft wares are used. For the simulation of the project circuit is done by the proteous software. It is a software for the circuit designing and drawn the circuit. It is very helpful to users to create a circuit. In this project, using programming language is C language. Because for developing program by the PIC-C software. PIC-C is use most of the syntax and semantics of standard C, e.g., main () function, variable definition, data type declaration, conditional statements (if, switch. case), loops (while, or), functions, arrays and strings, structures and union, bit operations, macros, unions, etc. The executing software of this project is PIC-KIT. This software is helps to see the executing path of the program and to identify the locations of the each codes.

**3.1 PROTEOUS**

It is a software suite containing, schematic, simulation, as well as PCB designing. **ISIS** is the software used to draw schematics and simulate the circuits in real time. The simulation allows human access during run time, thus providing real time simulations. By using this software, designer can also develop 2D, 3D drawings for the product.

**Features:**

1. ISIS has wide range of components in its library.
2. ARES offers PCB designing up to 14 inner layers, with surface mount and through all packages.
3. It offers auto routing and manual routing options to the PCB designer.
4. The schematic drawn in the ISIS can be directly transferred ARES.

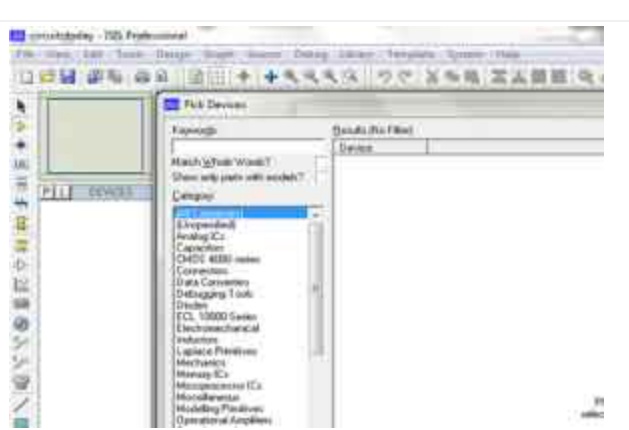


Fig 3.1.1 proteous software panel

**3.2 PIC- C:**

The use of C language to program PIC microcontroller is becoming too common. And most of the time its not easy to build an application in assembly which instead you can make easily in C. The PIC-C IDE compiler software is used for our programming. The PIC-C compiler for PIC16F876A microcontroller is a full featured ANSI compliant C compiler for the PIC16F87XX family.

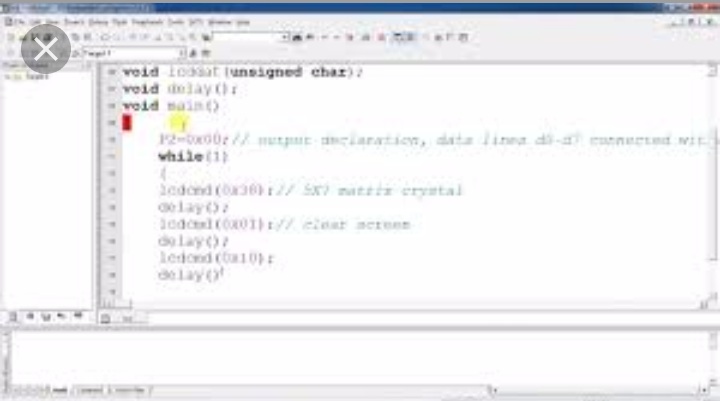


Fig 3.2.1 PIC-C programming window

**3.3.PIC-KIT:**

****

Fig 3.3.1 PIC-KIT

Microchip’s PICkit for the circuit debugger/programmer uses in circuit debugging logic incorporated in to provide a low cost hardware debugger and programmer.

**Features:**

1. Low cost
2. Minimum of additional hardware needed for debug
3. Expensive sockets or adapters are not required

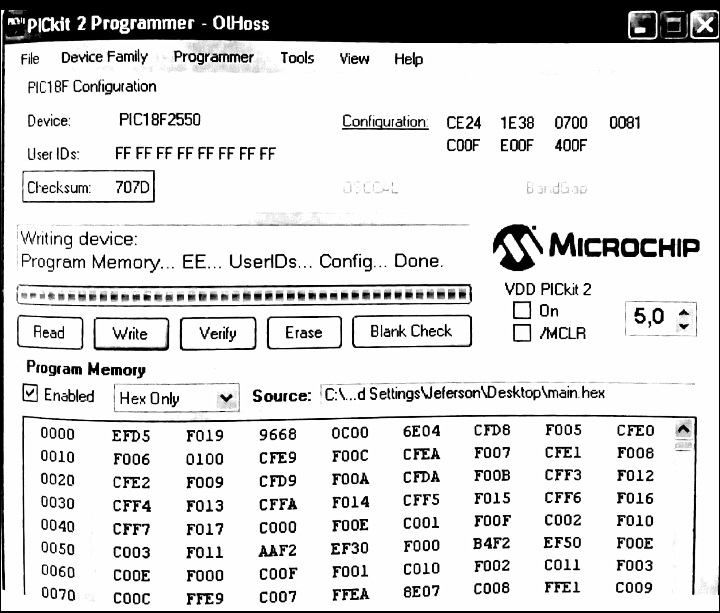


Fig 3.3.2 PIC kit memory window

**CHAPTER 4**

**IMPLIMENTATIONS**

4.1**.ROBOT WORKING ALGORITHM**

Step 1: Start

Step 2: The LED is blinking

Step 3: Enroll the finger on the finger print scanner. If it is matched go to step 4. If it is not matched go to step 2.

Step 4: Stop the LED blinking

Step 5: Get the character from the Bluetooth device, in the form of voice. If the command is forward, backward or left, right. Then call the function respect to the forward or backward and left or right. If the command is other than forward, backward, left, right, then go to the step 6.

Step 6: Stop

**2.progrm algorithm:**

**Main algorithm:**

Step 1: Start

Step 2: Set as port B as output port and port A (A0) as input port.

Step 3: Set the baud rate (9600 bps).

Step 4: Initialize character equals zero.

Step 5: If the A0 is zero, blink the LED otherwise go to step 6

Step 6: A0is one , then the character is not equal to zero. Otherwise LED blinking is off.

Step 7: If the character is “f”, then call the forward function. Otherwise character is equals B

Step 8: If the character is “B”, then call the backward function. Otherwise character is “L”.

Step 9: If the character is “L”, then call the left function. Otherwise character is R.

Step 10:If the character is “R”, then call the right function. Otherwise character is X.

Step 11: If the character is “X”, then call the stop function. Otherwise the engine is off.

**Forward and backward algorithm:**

Step 1: Start

Step 2: If the LED1 and LED2 are in high and m1\_pin1 is high,m1\_pin2 is low and m2\_pin1 is low,m2\_pin2 is high, then m1 rotates anticlockwise and m2 rotates clockwise direction. Then moves to forward , otherwise it moves to backward.

**Left and backward algorithm:**

Step1: Start

Step2: If the LED1 is high and LED2 is low and m1\_pin1 is low, m1\_pin2 is high and m2\_pin1 is low, m2\_pin2 is high, then the m1 and m2 are in clockwise direction. Then it moves to the left direction. Otherwise it moves to clockwise direction.

**Stop algorithm:**

Step 1: Start

Step 2: If the LED1 and LED2 are in low and m1\_pin1 is low, m1\_pin2 is low and m2\_pin1 is low, m2\_pin2 is low, then the m1 and m2 in the rest position. Otherwise it’s engine is off.

**4.2.ROBOT WORKING FLOW CHART**

START

LED BLINK

Finger print

Miss match

match

STOP Led blink

Character get from blue tooth

Forward/Backward

Left/Rigiht

STOP

**2 PROGRAM FLOW CHART**

START

Port B=Output port

Port A0=Input port

Baud rate=9600bps

Char ch=0

I s A0==0

**yes**

LED is blink

**NO**

Is A0==1

**NO**

Stop the LED blink

**YES**

Is ch!=0

**yes**

x

x

Is ch==F

**YES**

Call forward function

**NO**

Is ch==B

**YES**

Call backward

function

**NO**

Is ch==L

**YES**

Call left function

**NO**

Is ch==R

**YES**

Call Right function

**NO**

Is ch==X

**YES**

Call stop function

**NO**

Engine is off

STOP

**Forward and back ward function function:**

Start

LED1==1&& LED2==1

And ((m1\_pin1==1)(m1\_pin2==0))&&((m2\_pin1==0)(m2\_pin2==1))

**NO**

M2 rotates anti clock wise and m1 rotates clock wise direction

**YES**

Moves to backward

M1 rotates anti clock wise and m2 rotates clock wise direction

Moves to forward

**Left and right function:**

Start

LED1==1&& LED2==0

And ((m1\_pin1==0)(m1\_pin2==1))&&((m2\_pin1==0)(m2\_pin2==1))

**NO**

M1 and m2 rotates anti clock wise direction

**NO**

M1 and m2 rotates clock wise direction

Moves to right

Moves to left

**Stop function:**

Start

LED1==0&& LED2==0

And ((m1\_pin1==0)(m1\_pin2==0))&&((m2\_pin1==0)(m2\_pin2==0))

**NO**

Engine become off

**YES**

M1 and m2 are in rest.

Stop is perform

**CHAPTOR 5**

**CONCLUSION**

**A** highly reliable and easy system to accomplish a purpose design specific task such as distribution of medicine and food to the bed ridden patients specially in infected & inaccessible areas of the hospitals and medical Centre have been reported. The on-board intelligence helps providing situational awareness a basic requirement of the system to be operated by voice / tele confined for ascertaining a majority of other tasks in open loop environment. The operation by voice command could best be used for handicapped.

The outcome of the thesis is a simple robot which is controlled by a smart android phone& also receives the voice commands. This thesis aims to provide simple guidelines for people interested in building robots. As mentioned earlier, the project has been carried out several times and the aim of this thesis is to familiarize the students with fundamentals of PIC and Android to build anything possible. Although the thesis projects very little about the robot’s use in real world, but with the help of guidelines and the abundance of resources the outcome could be very beneficial for many people in the world. People with physical limitations such as handicapped people could use the feature to their wheel chair from this thesis to compensate their abilities.

The purpose of such robotic system is to help people with motor disabilities in controlling different widgets in daily life using mobile phone. The proposed idea can be expanded to control almost any device with Bluetooth receiver. In future we use a secured wireless channel using encryption and decryption. Consider larger bandwidth system should be onboard because video streaming service desired. Some of interfacing applications which can be made are controlling home appliances, robotics movements, Speech Assisted technologies, Speech to text translation, and many more. In future industries, home auto machine, agriculture is also developed by robotics. To reduce the labour efficiency, work efficiency, to reduce the working time to increase the productivity.

**Advantages:**

* By using robots we can control live video feed by giving voice commands.
* We can use voice control robot for a multiple ways. We can move any clockwise, anticlockwise direction, forward and backward by givingvoice commands.
* In industries we can control the machines by using robos.
* It is used in hazardous places.
* They are no need of food, nutrients need only electrical energy by means of battery storage, solar energy etc.
* You can program them to make them do exactly what you want them to do.
* They are more accurate than humans Eg no shaking when in a very important surgery, puts every screw in fabricating a car etc.
* You can send them to very dangerous places.
* You can make them do your job for you.
* They can perform tasks faster than humans and much more consistently and accurately. They can capture moments just too fast for the human eye to get, for example the Atlas detector in the LHC project can capture ~ 600000 frames per second while we can see at about 60.
* Most of them are automatic so they can go around by themselves without any human interference
* They can also useful for physically handicapped persons through wheel chairs etc.

They can reduce the labor work in industry by it portable takes the load from one place to another place. They can able to work always 24x7.

**Disadvantages:**

* They are very expensive to make they can reproduce but it could cost money for the materials.
* You need the right materials to make them.
* They can be very hard to program.
* You need highly trained people to make them.
* People can lose jobs in factories.
* It needs maintenance to keep it running.
* Low Bluetooth range approximately 100 meters.
* Not flexible with all cell phone.

**Applications:**

We believe such a system would find wide variety of applications. Menu driven systems such as e-mail readers, household appliances like washing machines, microwave ovens, and pagers and mobiles etc. will become voice controlled in future.

**Home automation**

The popularity of home automation has been increasing greatly in recent years due to much higher affordability and simplicity through Smartphone and tablet connectivity. The concept of the "Internet of Things" has tied in closely with the popularization of home automation.



Controlling Hardware Devices Like



Fig 5.1: Home Automation

**Wheelchairs:**

Based on our project the robot is controlled by giving a voice commands though android mobile. We can move wheel easily giving direction commands to android without hand movement.

Fig 5.2: Wheelchair Applications

**Surveillance device:**

Surveillance  is the monitoring of the behaviour, activities, or other changing information, usually of people for the purpose of influencing, managing, directing, or protecting them. This can include observation from a distance by means of electronic equipment (such as CCTV cameras).

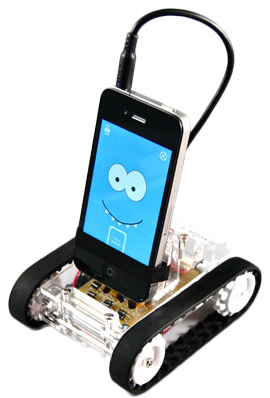


Fig 5.3: Surveillance Device

**Military applications:**

Mobile robots important role in military matters, from panel to dealing with potential explosives. “With suitable sensors and cameras to perform different missions, mobile robots are operated remotely for reconnaissance patrol and relay back video images to an operator.

Fig 5.4: Military Applications

**Industrial purposes**

Material handling is the most popular application with 38% of operational stock of industrial robots worldwide. This includes machine tending, palatalizing and various operations for metal machining and plastic mounding.In cement industries machines are operated by robos. To reduce the useof labours robots are used in loading purpose.

**Agricultural purposes**

Robots are also useful in agriculture purpose in the next century. For planting seeding, cropping etc.

* It helps physically disabled persons by carrying some objects from one place to another place using the arm structure in the robot.
* It guides the blind persons to reach a particular Destination by using the voice feature.
* The robot is useful in places where humans find difficult to reach but human.
* Voice reaches. E.g. in a small pipeline, in a fire-situations, in highly toxic areas.
* The robot can be used as a toy.
* It can be used to bring and place small objects.
* It is the one of the important stage of Humanoid robots.
* Command and control of appliances and equipment.
* Telephone assistance systems.
* It is also used in Data entry.
* Speech and voice recognition security systems.
* The photo electric sensor in the robot will sense the obstacles and it will make decisions according to the obstacles it encounters.

**APPENDIX**

**Hardware process:**

Fig 7.1 hardware board

Fig7.2 improved hardware board

Fig 7.3 completed hardware