KARNATAK UNIVERSITY, DHARWAD

Accredited by NAAC with "A" Grade
University with Potential for Excellence

P.G. DEPARTMENT OF STUDIES IN ELECTRONICS

Mini Project Report on

"RFID BASED AUTOMATIC GATE"

UNDER THE GUIDANCE OF

S.R. Inamdar
Prof. And Chairman

And Co-Guide

Milind R. Pawar
Teaching Assistant
Department of Electronics
KARNATAK UNIVERSITY, DHARWAD

2020-21

SUBMITTED TO THE
KARNATAK UNIVERSITY, DHARWAD
IN PARTIAL FULFILLMENT OF DEGREE OF

Master Of Science
In
Electronics

By

Mr. Akshay N. Bhat
Ms. Varsha S. Hegde

ACKNOWLEDGMENT

We are wholeheartedly thankful to ***Prof. S.R. Inamdar,*** Chairman Department of Electronics K.U. Dharwad, who has always supported us at all the time.

With profound respect, we express our deep sense of gratitude to our guide ***Shri. Milind R. Pawar*** Teaching Assistant Department of Electronics K. U. Dharwad. We are thankful for his inspiring guidance and encouragement at every stage of the project. He is always a tremendous source of inspiration and invaluable help in channelizing our effort in the right direction and whose thoughts and guidance are reflected on every step's of our project.

We express our sincere thanks to ***Mr. Anup Jadav*** for providing all the required facilities for completion of the project and helping throughout the project.

We would be failing in our duty if we don't express our sincere thanks to our parents who have morally supported us by their encouragement without which the realization of this task would have been distant dream for us.

PROJECT TEAM

Table of Contents

ABSTRACT

With rapid increase in the use of vehicles there is need to have secure and automated vehicle entry system to the University campus or any parking areas of the public utility. With increasing labor costs and the lower efficiency, the manual system of management of entry gates is totally obsolete in modern day.

The most economic, fastest and totally contactless solution would be the implementation of **RFID** technology in the entry gates. The vehicle owners would be given an **RFID** card with Unique Number (Hard to Duplicate) which gives secure access at the entry point. They have to show the card at the scanner which is aptly located which matches the Unique ID number with the database and only the vehicle containing correct ID would be permitted to enter by letting the gate open automatically.

INTRODUCTION

This project is based on the technology of **RFID** (Radio Frequency Identification). Radio Frequency Identification (RFID) technology utilizes the electromagnetic fields for data transfer in order to perform automatic detection and tracking of tags or tags of objects. It can provide ways to design and implement relatively inexpensive systems.

In this century, people live a life which is solely dependent on technology. We wake up to the ringing of alarm from our smartphone, and we end our day with sending good night messages from the same smartphone. That is how much we are dependent on technology and **Internet of Things** is the future for mankind. In the present scenario, the world is straightly moving towards automation.

The present manual system of entry gate of which is predominant throughout India is costly, slow and very inefficient. The limitations of workers such as mistakes in handling the entry, wrongly recording the data may lead to unauthorized persons entering the campus. The mandatory breaks needed by the attendant will cause a long line of vehicles waiting to let in. With use of **RFID** technology not only the secure access to authorized personnel is guaranteed but also a 24×7 service without any break or rest is assured.

Also, I would like to shed some light on the benefits of automation with some statistical data.

Even with the most efficient manual system, each vehicle would have to wait for **90** seconds near the gate so that the security guard verifies the data in his system records the same in his log book and lets the vehicle pass.

Suppose there are 300 vehicles which needs entry inside the campus or parking area, then the time taken by them would be **300×90 = 27000** seconds. If we calculate this for a month (for both entry and exit) it would amount to a time of **27000** minutes or **450** Hours. Calculating the same for a year would amount to a staggering **5400** Hours, all of which leads to causing of pollution and wastage of fuel and money. That's why the proposed RFID technology would be not less than a revolution.

This system can be extended or modified to use it as Automatic Toll collection at toll plaza, Booking parking slots in places like Malls, Theaters.

SCOPE OF THE PROJECT

Whenever the matter of Integration of systems comes to mind, we think of a system having the following important features viz.

Accuracy: All the functionally bonded logical dependencies must be integrated.

Efficiency: The whole system should work under all circumstances and on a long run it should work efficiently irrespective of their proprietary format.

Cost Effectiveness: As our software do not require any special software for implementation hence, is less costly as compared to other existing system.

Any Prerequisite for the use: As the existing systems are not altered, and integration is done at the background, hence there is no need for any training.

FLEXIBILITY OF IMPLEMENTATION

The main power of this project, is the technology which is used, that is the RADIO FREQUENCY IDENTIFICATION. The basic power of this technology is that it's very flexible. Even with the slightest of change in project, the product can be shaped into a completely different implementation and all that can be because RFID is independent of every other hardware that can be used to boost up the system's performance.

RADIO FREQUENCY has vast implementation areas in medical, defense, transport and many latest products that are being developed is based on RFID solution. The main areas is animal tracking, human implants, vehicle tracking, speed tracking, physical implementation.

Following are the features and advancement of our project over presently existing system:

1. RFID tag cannot be cloned, so cannot be cheated.
2. Very efficient saving of time.
3. Wastage of money reduced.
4. Consumption of oil is reduced.
5. Pollution is reduced to a large extent.
6. Speedy transport.
7. Comparatively, less maintenance cost

BLOCK DIAGRAM:

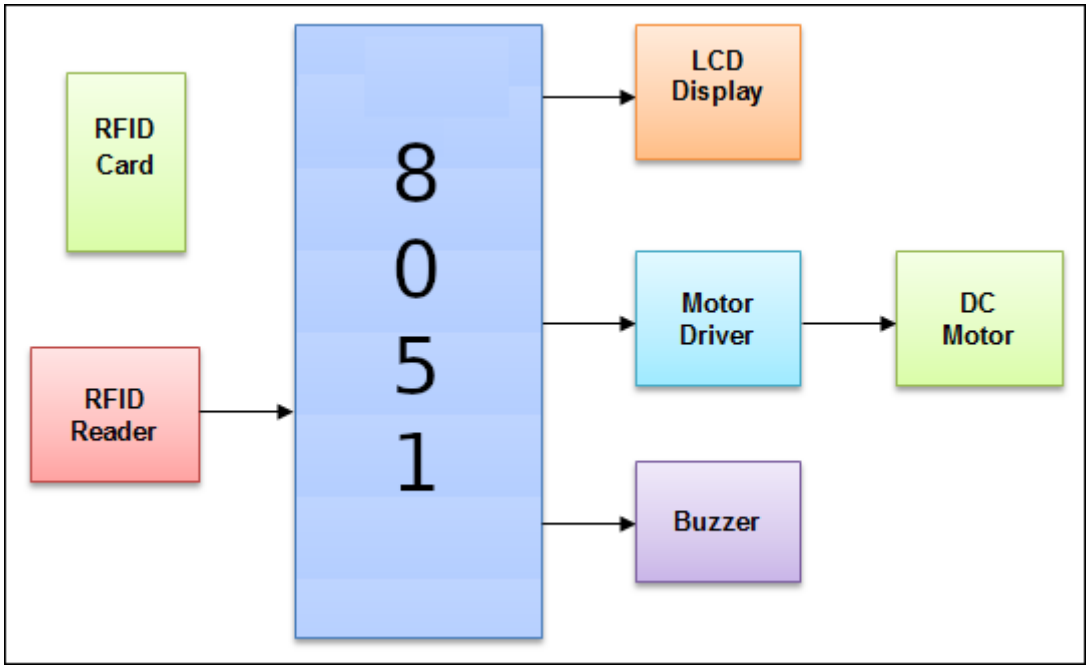


Figure 1: Block Diagram

Description :-

This picture shows the basic block diagram of our project. The 8051 micro-controller is used for the control system. In this system, the micro-controller is the main part of the system because it does the comparison and checks the data and the output results showed on the LCD. And then, the microcontroller sent the signal to the motor driver for opening the gate.

When the car owner shows the RFID card to RFID reader, it reads the RFID card and given to microcontroller. Microcontroller checks whether the ID card is valid or not. If it is not valid card then microcontroller displays information on LCD and buzzer gets on. If it is valid card then microcontroller gives information to motor and displays in LCD. Then motor opens the gate.

COMPONENT USED:

- Microcontroller 8051
- 8051 Development Board
- EM-18 RFID Reader Module
- RFID Tags
- LCD 16×2
- Motor Driver ULN2003A
- Stepper Motor

SOFTWARE USED:

- Keil µVision
- Proteus Design Suite
- SSTFlashFlex51

HARDWARE DESCRIPTION:

1. MICROCONTROLLER 8051

1.1 What is a Microcontroller?

A microcontroller 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 neither Ferroelectric RAM, 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 computer or other general purpose applications consisting of various discrete chips.

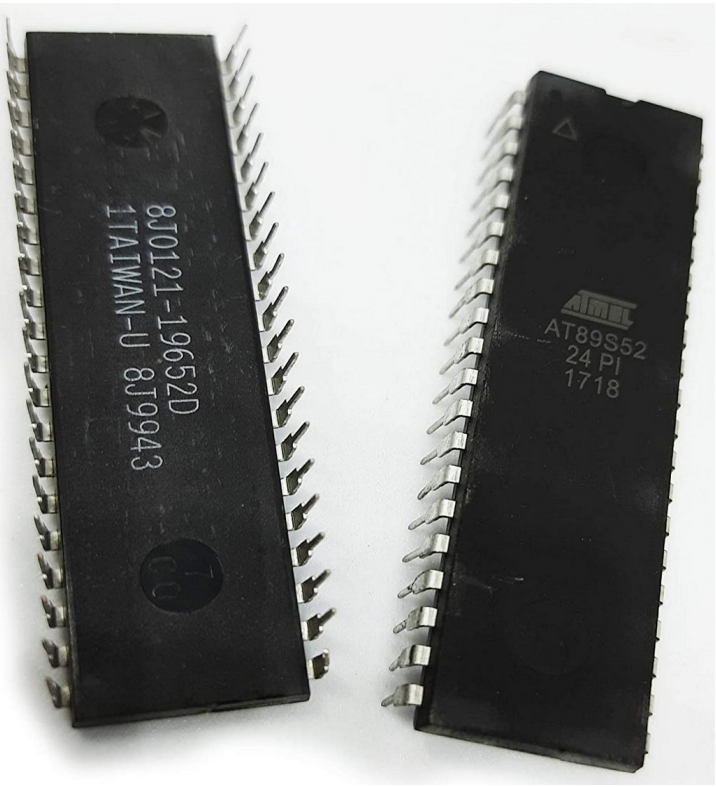


Figure 2: Microcontroller AT89S52

The microcontroller is one kind of integrated circuit that includes 40-pins with dual inline package or DIP, RAM-128 bytes, ROM-4kb & 16-bit timers–2. Based on the requirement, it includes addressable & programmable 4 - parallel 8-bit ports. In the 8051 microcontroller architecture, the system bus plays a key role to connect all the devices to the central processing unit. This bus includes a data bus - an 8-bit, an address bus-16-bit & bus control signals. Other devices can also be interfaced throughout the system bus like ports, memory, interrupt control, serial interface, the CPU, timers.

There are two buses in 8051 Microcontroller one for the program and another for data. As a result, it has two storage rooms for both programs and data of 64K by 8 sizes. The Microcontroller comprises 8-bit accumulator & an 8-bit processing unit. It also consists of 8 bit B register as majorly functioning blocks and 8051 microcontroller programming is done with embedded C language using Keil software. It also has several other 8 bit and 16-bit registers.

For internal functioning & processing Microcontroller, 8051 comes with integrated built-in RAM. This is prime memory and is employed for storing temporary data. It is an unpredictable memory that is its data can get be lost when the power supply to the Microcontroller switched OFF. This microcontroller is very simple to use, affordable less computing power, simple architecture & instruction set.

1.2 The History :-

The first microprocessor was the 4-bit Intel 4004 released in 1971, with the Intel 8008 and other more capable microprocessors becoming available over the next several years. However, both processors required external chips to implement a working system, raising total system cost, and making it impossible to economically computerize appliances.

The Smithsonian Institution credits **TI** engineers Gary Boone and Michael Cochran with the successful creation of the first microcontroller in 1971. The result of their work was the **TMS** 1000, which became commercially available in 1974. It combined read-only memory, read/write memory, processor and clock on one chip and was targeted at embedded systems.

Partly in response to the existence of the single-chip **TMS** 1000, Intel developed a computer system on a chip optimized for control applications, the Intel 8048, with commercial parts first shipping in 1977. It combined **RAM** and **ROM** on the same chip. This chip would find its way into over one billion **PC** keyboards, and other numerous applications. At that time, Intel's President, Luke J. Valenter, stated that the microcontroller was one of the most successful in the company's history, and expanded the division's budget over 25%.

Most microcontrollers at this time had concurrent variants. One had an erasable **EPROM** program memory, with a transparent quartz window in the lid of the package to allow it to be erased by exposure to ultraviolet light, often used for prototyping. The other was either a mask programmed **ROM** from the manufacturer for large series, or a **PROM** variant which was only programmable once; sometimes this was signified with the designation **OTP**, standing for "one-time programmable".

The **PROM** was of identical type of memory as the **EPROM**, but because there was no way to expose it to ultraviolet light, it could not be erased. The erasable versions required ceramic packages with quartz windows, making them significantly more expensive than the **OTP** versions, which could be made in lower-cost opaque plastic packages. For the erasable variants, quartz was required, instead of less expensive glass, for its transparency to ultraviolet-glass is largely opaque to **UV**, but the main cost differentiator was the ceramic package itself.

In 1993, the introduction of **EEPROM** memory allowed microcontrollers beginning with the Microchip to be electrically erased quickly without an expensive package as required for **EPROM**, allowing both rapid prototyping, and In System Programming. (**EEPROM** technology had been available prior to this time, but the earlier **EEPROM** was more expensive and less durable, making it unsuitable for low-cost mass-produced microcontrollers.)

The same year, Atmel introduced the first microcontroller using Flash memory, a special type of **EEPROM**. Other companies rapidly followed suit, with both memory types. Cost has plummeted over time, with the cheapest 8-bit microcontrollers being available for under 0.25 USD in quantity (thousands) in 2009, and some 32-bit microcontrollers around US$1 for similar quantities.

Nowadays, microcontrollers are cheap and readily available for hobbyists, with large online communities around certain processors. In the future, **MRAM** could potentially be used in microcontrollers as it has infinite endurance and its incremental semiconductor wafer process cost is relatively low.

1.3 Embedded design :-

A microcontroller can be considered a self-contained system with a processor, memory, and peripherals and can be used as an embedded system. The majority of microcontrollers in use today are embedded in other machinery, such as automobiles, telephones, appliances, and peripherals for computer systems. While some embedded systems are very sophisticated, many have minimal requirements for memory and program length, with no operating system, and low software complexity. Typical input and output devices include switches, relays, solenoids LEDs, small or custom liquid-crystal displays, radio frequency devices, and sensors for data such as temperature, humidity, light level etc. Embedded systems usually have no keyboard, screen, disks, printers, or other recognizable I/O devices of personal computer, and may lack human interaction devices of any kind.

1.4 Features Of 8051 Microcontroller :-

- 4.0**V** to 5.5**V** Operating Range
- Fully Static Operation: 0 Hz to 33 MHz
- Three-level Program Memory Lock
- Oscillator & **CLK** Circuit
- 8-bit **CPU** through two Registers A & B
- 8-bit Processor Status Word (**PSW**)
- 8-bit Stack Pointer
- 16-bit Program Counter
- Two 16-bit timer/counters : **T0** and **T1**
- 16-bit Program Counter & **DPRT** (Data Pointer)
- Two external and three internal vectored interrupts
- Control Registers like **PCON**, **SCON**, **SBUF**, **TMOD**, **TCON**, **IE**, & **IP**.
- 32 I/O pins arranged as four 8-bit ports such as **P0**, **P1**, **P2** & **P3**.
- 8K Bytes of In-System Programmable (**ISP**) Flash Memory Endurance: 10,000 Write/Erase Cycles which supports while programming the system.
- 256 x 8-bit Internal **RAM** where the first **RAM** with 128 Bytes from 00H to 7FH is once more separated into four banks through 8 registers in every bank, addressable registers -16 bit & general-purpose registers - 80.
- Special Function Registers (**SFRs**) of 128 bytes from 80H to FFH. These registers control various peripherals such as Serial Port, Timers, all I/O Ports, etc.
- Serial Data **Tx** & **Rx** for Full-Duplex Operation.

1.5 Pin Diagram of 8051 Microcontroller :-

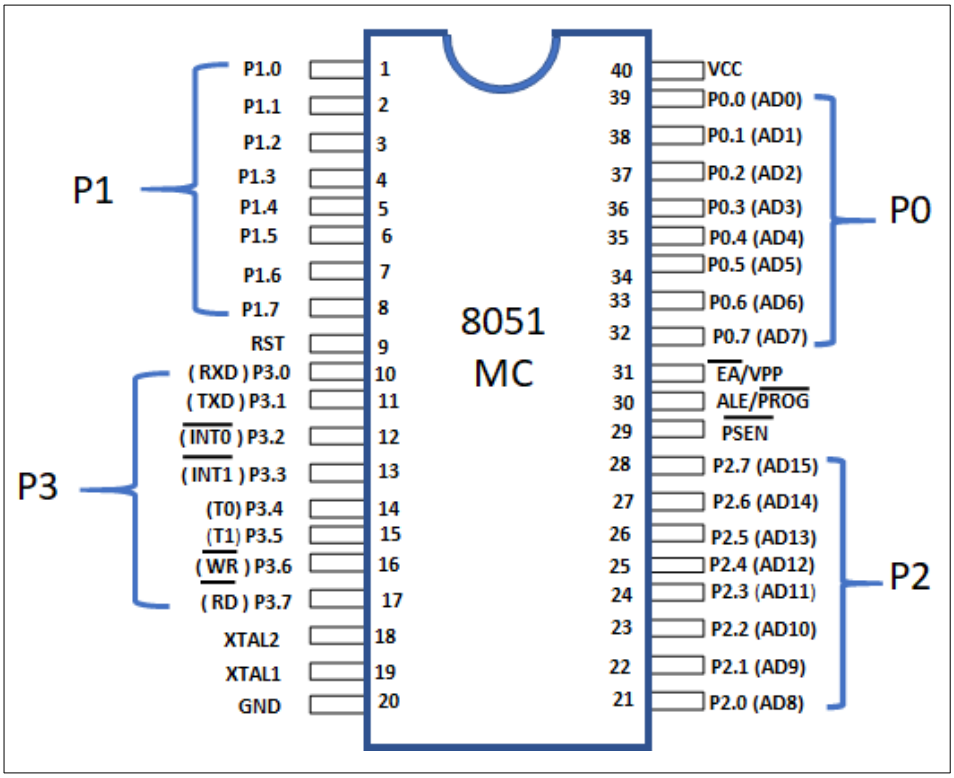


Figure 3: Pin Diagram Of 8051

8051 Microcontroller is available in 40 pin Dual Inline package (**DIP**). All these 40 pins perform different functions like read, write, interrupts, I/O operations, address etc. 8051 microcontroller have four I/O ports (Port-1, Port-2, Port-3, Port-4) where in each port has 8 pins. Each pin configured as an input pin or output pin depends on logic state of the pin. Therefore, out of 40 pins, 32 pins are allotted for I/O port. The remaining pins are assigned to **VCC**, **GND**, **XTAL1**, **XTAL2**, **RST**, **ALE**, **EA** and **PSEN**.

1.5 a) Explanation of the Pins :-

- **Pin 1 to Pin 8 (Port - 1) -** Pin 1 to Pin 8 is assigned to Port 1 for simple I/O operations. These ports are work as a bidirectional port. It means all the pins of port 1 work as an input pin or output pins. If Logic 1 (one) is applied to the I/O port it will act as an input pin and if logic 0 (zero) is applied to the I/O port it will act as an output pin.
- **Pin 9 (RST)** – It is a reset input Pin, which is used to reset the 8051 microcontrollers to its initial values when logic 1 is applied to this pin. It is active high pin.

- **Pin 10 to Pin 17 (Port-3)** – Pin 10 to Pin 17 are assigned to Port 3. This port is also a bidirectional **I/O** port like port 1. This port performs some special functions like interrupts, control signals, timer input, serial communication etc. The detail function of each pin are given below:
    - **Pin 10 (P3.0) (RXD) -**Pin 10 is used as a **RXD** (serial data receive pin) which is for serial input pin. By using this input signal microcontroller receives data for serial communication.
    - **Pin 11 (P3.1) (TXD)** – Pin 11 is used as a **TXD** (serial data transmit pin) which is serial output pin. By using this output signal microcontroller transmits data for serial communication.
    - **Pin 12 & Pin 13 (P3.2, P3.3) ($\overline{\text{INT0}}$, $\overline{\text{INT1}}$)** – Pins12 and 13 are used for External Hardware Interrupt 0 and Interrupt 1 respectively. When this interrupt is activated (i.e. when it is low), 8051 gets interrupted means it stopped whatever it is doing and jumps to the vector table where **ISR's** (Interrupt Service Routine) are stored and starts performing Interrupt Service Routine (**ISR**) from that vector location.
    - **Pin 14 & Pin 15 (P3.4, P3.5) (T0 and T1)** –Pin 14 and 15 are used for Timer 0 and Timer 1 external input. They can be connected with 16-bit timer/counter.
    - **Pin 16 (P3.6) ($\overline{\text{WR}}$)** –Pin 16 is used for external memory write that is writing data to the external memory.
    - **Pin 17 (P3.7) ($\overline{\text{RD}}$)** –Pin 17 is used for external memory read that reading data from external memory.
- **Pin 18 and Pin 19 (XTAL2 And XTAL1)** –Pins 18 and 19 that is **XTAL 2** and **XTAL 1** are the pins for interfacing external oscillator. Mostly, a Quartz Crystal Oscillator is connected here to get the system clock.
- **Pin 20 (GND)** – Pin 20 is the Ground Pin. It is connected to the **0V** (negative terminal) of the Power Supply.

- **Pin 21 to Pin 28 (Port 2)** – Pin 21 to pin 28 are port 2 pins. Port 2 is also a bidirectional Input /Output port that is, all pins of port 2 work as an input pin or as an output pins. But, this is only possible when we are not using any external memory. If we use external memory, then these pins will work as high order address bus (**A_8 to A_{15}**).
- **Pin 29 (PSEN)** – The Pin 29 is the Program Store Enable Pin (**PSEN**). It is used to enable external program memory and read a signal from the external program memory.
- **Pin 30 (ALE)** – Pin 30 is the Address Latch Enable Pin. This pin is used to enable or disable the external memory interfacing.
- **Pin 31 (EA)** – Pin 31 is the External Access Enable (**EA**) Pin. This pin allows external Program Memory. It is an input pin and connected from **VCC** or **GND**. If we want to access the program from external program memory, it must be connected with **GND**. If we want to use on-chip memory, it must be high (connected with **VCC**).
- **Pin 32 to Pin 39 (Port 0) (AD_0 to AD_7)** – Pin 32 to Pin 39 are Port 0 pins. When we don't use any external memory, these pins are used as a bidirectional pin like port 2 and port 3. But, when **ALE** or Pin 30 is at 1, then this port is used as data bus(**D_0 to D_7**). And when the **ALE** pin is at 0, then this port is used as a lower order address bus (**A_0 to A_7**).
- **Pin 40 (VCC)** – This pin is used to provide (**+5v**) power supply to the 8051 microcontroller circuit.

1.6 Interrupt

The most powerful and important features are interrupts in 8051 microcontrollers. In most of the real-time processes, to handle certain conditions properly, the actual task must be halt for some time – it takes required action – and then must return to the main task. For executing such type of programs, interrupts are necessary. It entirely differs from the polling method wherein the processor must check sequentially each device and ask whether the service is required or not while consuming more processor time.

Interrupts in 8051 microcontrollers are more desirable to reduce the regular status checking of the interfaced devices or inbuilt devices. Interrupt is an event that temporarily suspends the main program, passes the control to a special code section, executes the event-related function and resumes the main program flow where it had left off.

Interrupts are of different types like software and hardware, maskable and non-maskable, fixed and vector interrupts, and so on. Interrupt Service Routine (**ISR**) comes into the picture when interrupt occurs, and then tells the processor to take appropriate action for the interrupt, and after **ISR** execution, the controller jumps into the main program.

1.6 a) Types of Interrupts :-

1. Timer 0 overflow interrupt- TF0
2. Timer 1 overflow interrupt- TF1
3. External hardware interrupt- INT0
4. External hardware interrupt- INT1
5. Serial communication interrupt- RI/TI

The Timer and Serial interrupts are internally generated by the microcontroller, whereas the external interrupts are generated by additional interfacing devices or switches that are externally connected to the microcontroller. These external interrupts can be edge triggered or level triggered. When an interrupt occurs, the microcontroller executes the interrupt service routine so that memory location corresponds to the interrupt that enables it. The Interrupt corresponding to the memory location is given in the interrupt vector table below.

Interrupt Number	Interrupt Description	Address
0	EXTERNAL INT 0	0003h
1	TIMER/COUNTER 0	000Bh
2	EXTERNAL INT 1	0013h
3	TIMER/COUNTER 1	001Bh
4	SERIAL PORT	0023h

Figure 4: Interrupt Vector Table

Upon **'RESET'** all the interrupts get disabled, and therefore, all these interrupts must be enabled by a software. In all these five interrupts, if anyone or all are activated, this sets the corresponding interrupt flags as shown in the figure. All these interrupts can be set or cleared by bit in some special function register that is Interrupt Enabled (**IE**), and this in turn depends on the priority, which is executed by **IP** interrupt priority register.

1.6 b) Interrupt Enable Register **(IE)** :-

This register is responsible for enabling and disabling the interrupt. It is a bit addressable register in which **EA** must be set to one for enabling interrupts. The corresponding bit in this register enables particular interrupt like timer, external and serial inputs. In the below **IE** register, bit corresponding to 1 activates the interrupt and 0 disables the interrupt.

EA	–	–	ES	ET1	EX1	ET0	EX0

EA	IE.7	Disables all interrupts. If EA = 0, no interrupt will be acknowledged. If EA = 1, interrupt source is individually enable or disabled by setting or clearing its enable bit.
-	IE.6	Not implemented, reserved for future use*.
-	IE.5	Not implemented, reserved for future use*.
ES	IE.4	Enable or disable the Serial port interrupt.
ET1	IE.3	Enable or disable the Timer 1 overflow interrupt.
EX1	IE.2	Enable or disable External interrupt 1.
ET0	IE.1	Enable or disable the Timer 0 overflow interrupt.
EX0	IE.0	Enable or disable External Interrupt 0.

Figure 5: Interrupt Enable (IE) Register

1.6 c) Interrupt Priority Register (**IP**):-

It is also possible to change the priority levels of the interrupts by setting or clearing the corresponding bit in the Interrupt priority (**IP**) register as shown in the figure. This allows the low priority interrupt to interrupt the high-priority interrupt, but prohibits the interruption by another low-priority interrupt. Similarly, the high-priority interrupt cannot be interrupted. If these interrupt priorities are not programmed, the microcontroller executes in predefined manner and its order is **INT0**, **TF0**, **INT1**, **TF1**, and **SI**.

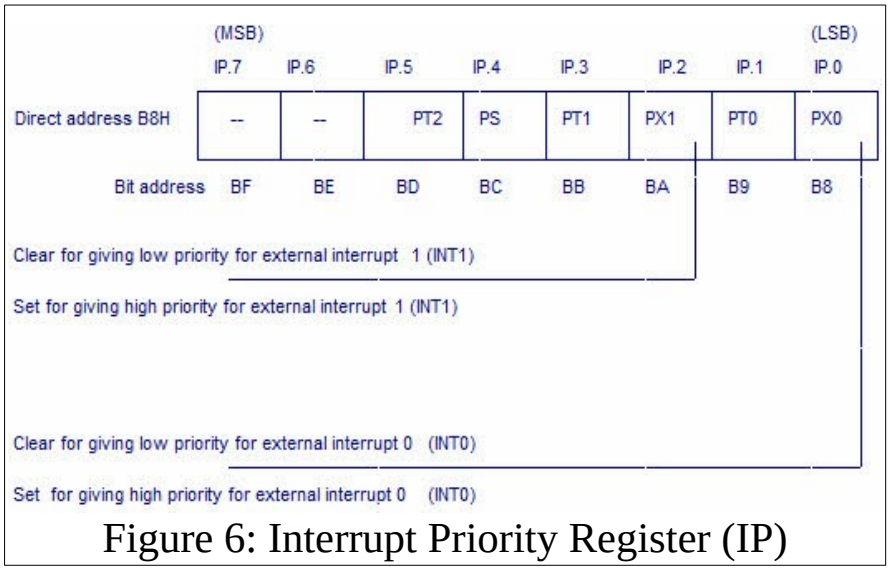


Figure 6: Interrupt Priority Register (IP)

1.6 d) **TCON** Register :-

In addition to the above two registers, the **TCON** register specifies the type of external interrupt to the 8051 microcontroller, as shown in the figure. The two external interrupts, whether edge or level triggered, specify by this register by a set, or cleared by appropriate bits in it. And, it is also a bit addressable register.

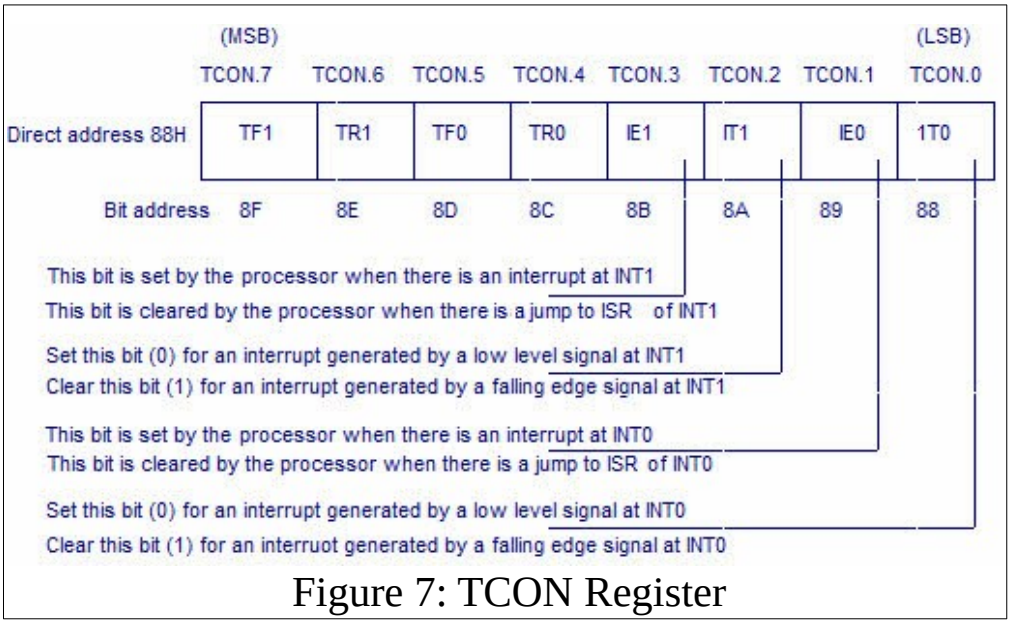


Figure 7: TCON Register

1.6 e) Serial Interrupts :-

Serial Interrupts are slightly different from the rest of the interrupts. This is due to the fact that there are two interrupt flags: **RI** and **TI**. If either flag is set, a serial interrupt is triggered. As you will recall from the section on the serial port, the **RI** bit is set when a byte is received by the serial port and the **TI** bit is set when a byte has been sent.

This means that when your serial interrupt is executed, it may have been triggered because the **RI** flag was set or because the **TI** flag was set--or because both flags were set. Thus, your routine must check the status of these flags to determine what action is appropriate. Also, since the 8051 does not automatically clear the **RI** and **TI** flags you must clear these bits in your interrupt handler.

1.7 Serial Port Operation / Communication :-

One of the 8051s many powerful features is its integrated **UART**, otherwise known as a serial port. The fact that the 8051 has an integrated serial port means that you may very easily read and write values to the serial port. If it were not for the integrated serial port, writing a byte to a serial line would be a rather tedious process requiring turning on and off one of the **I/O** lines in rapid succession to properly "clock out" each individual bit, including start bits, stop bits, and parity bits.

However, we do not have to do this. Instead, we simply need to configure the serial port's operation mode and baud rate. Once configured, all we have to do is write to an **SFR** to write a value to the serial port or read the same **SFR** to read a value from the serial port. The 8051 will automatically let us know when it has finished sending the character we wrote and will also let us know whenever it has received a byte so that we can process it. We do not have to worry about transmission at the bit level--which saves us quite a bit of coding and processing time.

1.7 a) Setting the Serial Port Mode :-

The first thing we must do when using the 8051s integrated serial port is, obviously, configure it. This lets us tell the 8051 how many data bits we want, the baud rate we will be using, and how the baud rate will be determined.

First, lets present the "Serial Control" (**SCON**) **SFR** and define what each bit of the **SFR** represents:

Bit	Name	Bit Address	Explanation of Function
7	SM0	9Fh	Serial port mode bit 0
6	SM1	9Eh	Serial port mode bit 1
5	SM2	9Dh	Multiprocessor Communications Enable
4	REN	9Ch	Receiver Enable. This bit must be set in order to receive characters.
3	TB8	9Bh	Transmit bit 8. The 9th bit to transmit in mode 2 and 3.
2	RB8	9Ah	Receive bit 8. The 9th bit received in mode 2 and 3.
1	TI	99h	Transmit Flag. Set when a byte has been completely transmitted.
0	SI	98h	Receive Flag. Set when a byte has been completely received.

Additionally, it is necessary to define the function of SM0 and SM1 by an additional table:

SM0	SM1	Serial Mode	Explanation	Baud Rate
0	0	0	8-bit Shift Register	Oscillator / 12
0	1	1	8-bit UART	Set by Timer 1 (*)
1	0	2	9-bit UART	Oscillator / 64 (*)
1	1	3	9-bit UART	Set by Timer 1 (*)

(*) **Note:** The baud rate indicated in this table is doubled if **PCON.7 (SMOD)** is set.

The **SCON**, **SFR** allows us to configure the Serial Port. Thus, we'll go through each bit and review its function.
The first four bits (bits 4 through 7) are configuration bits.

1.7 b) Writing to the Serial Port :-

Once the Serial Port has been properly configured as explained above, the serial port is ready to be used to send data and receive data. If you thought that configuring the serial port was simple, using the serial port will be a breeze.
To write a byte to the serial port one must simply write the value to the **SBUF** (99h) **SFR**.

1.7 c) Setting the Serial Port Baud Rate :-

Once the Serial Port Mode has been configured, the program must configure the serial port's baud rate. This only applies to Serial Port modes 1 and 3. The Baud Rate is determined based on the oscillator's frequency when in mode 0 and 2. In mode 0, the baud rate is always the oscillator frequency divided by 12. This means if you're crystal is 11.059Mhz, mode 0 baud rate will always be 921,583 baud. In mode 2 the baud rate is always the oscillator frequency divided by 64, so an 11.059Mhz crystal speed will yield a baud rate of 172,797.

1.7 d) Reading the Serial Port :-

Reading data received by the serial port is equally easy. To read a byte from the serial port one just needs to read the value stored in the **SBUF** (99h) **SFR** after the 8051 has automatically set the **RI** flag in **SCON**.

2. 8051 DEVELOPMENT BOARD

To get started with an embedded development, we need two major things that are development board and an **IDE** (Integrated Development Environment). A microcontroller development board is a printed circuit board (**PCB**) with circuitry and hardware designed to facilitate experimentation with a certain microcontroller boards features. The Development boards are combined with a processor, memory, chipset and on-board peripherals like LCD, Keypad, USB, serial port, ADC, RTC, Motor Driver ICs, etc. with debugging features. This will save us from messing with the connections with jumper wires and the board.

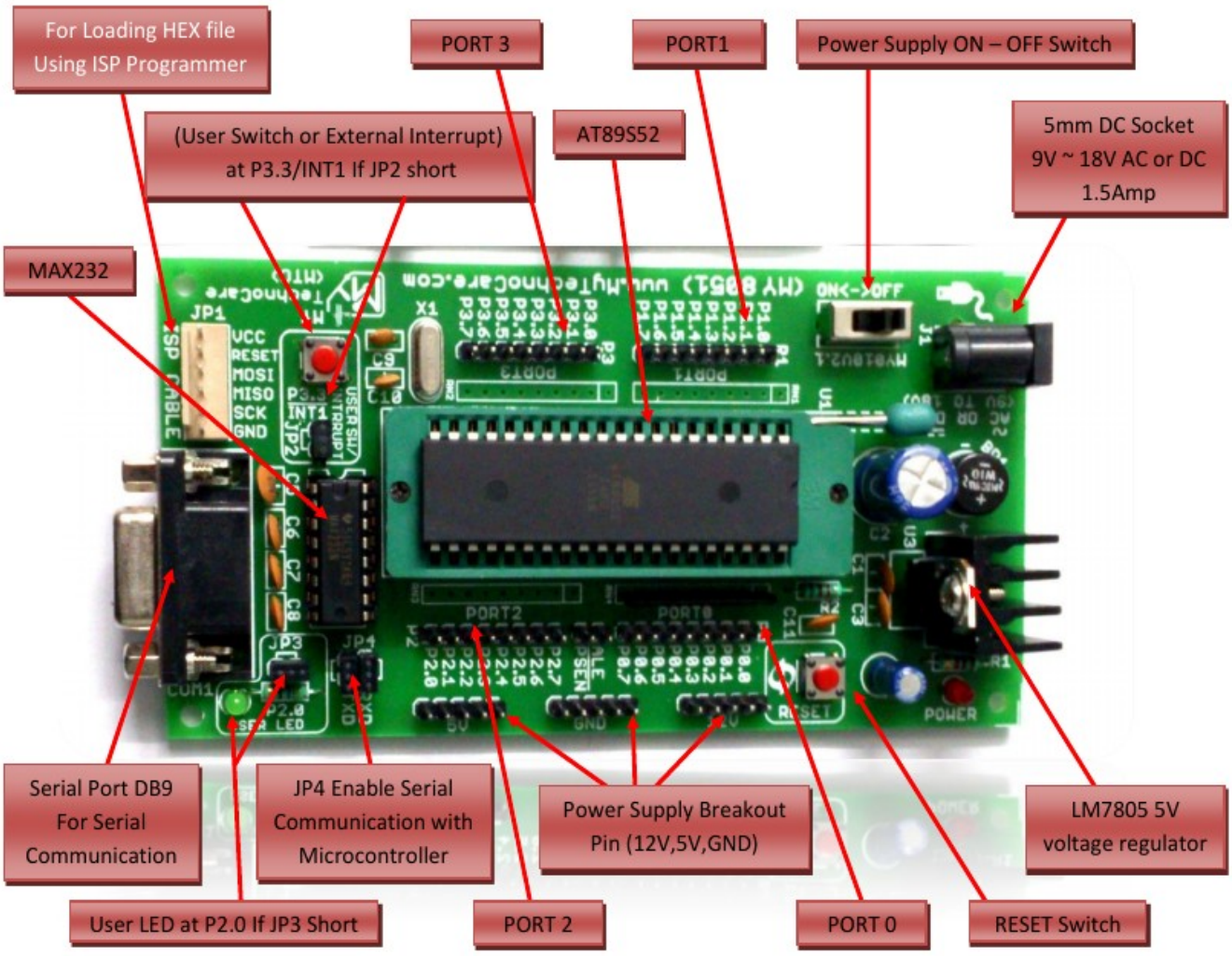


Figure 8: 8051 Microcontroller Development Board

The Specifications of Microcontroller Board are bus type, processor type, memory, number of ports, port type, and operating system. These are used to evaluate programs for embedded devices such as different controllers, home appliances, robots, point-of-sale (**PoS**) terminals, kiosks, and information appliances. Here, we will discuss on the characteristic between different development boards around the world. Each one of these has their own features as well as some drawbacks, and some development platforms are prominent for certain projects than others.

The microcontroller development board also known as Single board microcontroller. Nowadays, it is very simple and cheap to develop a single board microcontroller development kit. So many open source software's (IDEs) available to develop a microcontroller boards in order to develop a real time application. The above figure shows 8051 Microcontroller Development Board. The RS232 driver on board allows easy connection with PC or other embedded hardware. The board have User buttons and user LED. The bridge rectifier allows this board to be powered with both AC and DC power supply adapters. Operating Voltage is 9V to 18V AC or DC.

3. EM-18 RFID READER.

3.1 What is RFID?

RFID stands for **"Radio-Frequency Identification"**. The acronym refers to small electronic devices that consist of a small chip and an antenna. The chip typically is capable of carrying 2,000 bytes of data or less.

3.2 What is RFID Reader ?

The EM-18 RFID Reader module operating at 125kHz is an inexpensive solution for your RFID based application. The Reader module comes with an on-chip antenna and can be powered up with a 5V power supply. Power-up the module and connect to transmit pin of the module to receive pin of your microcontroller. Show your card within the reading distance and the card number is thrown at the output. Optionally the module can be configured for also a weigand output.

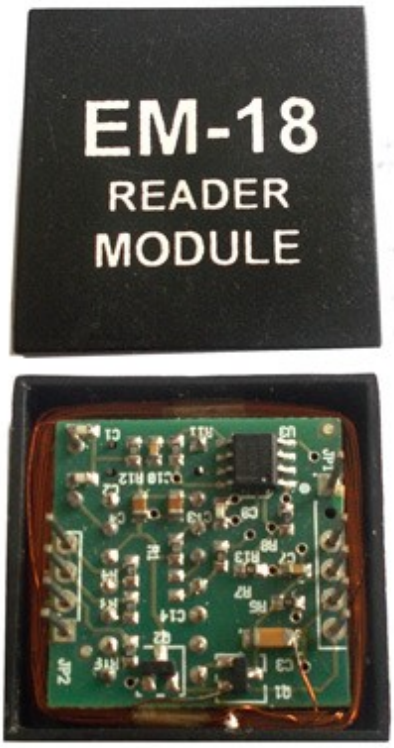


Figure 9: EM-18 RFID Reader Module

RFID EM18 working mechanism is quite simple. RFID Reader module has a transceiver (which can transmit and receives data) which generates a radio signal and transmits it through an antenna. This signal itself is the form of energy which is used to activate and power up the RFID tag.

3.3 EM-18 Pin Configuration :-

EM-18 is a nine pin device. Among nine pins, 2 pins are not connected, so we basically have to consider seven terminals.

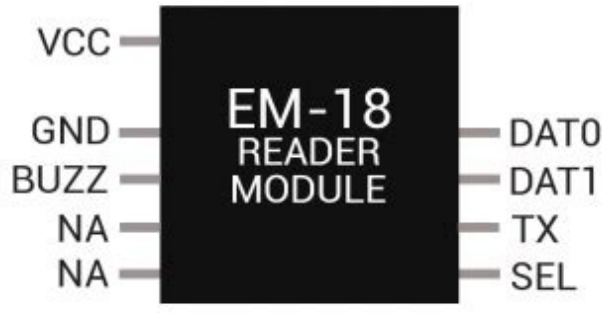


Figure 10: EM-18 Pin Configuration

Pin Number	Pin Description
VCC	Should be connected to positive of power source.
GND	Should be connected to ground.
BUZZ	Should be connected to BUZZER
NC	No Connection
NC	No Connection
SEL	SEL=1 then o/p =RS232 SEL=0then o/p=WEIGAND
TX	DATA is given out through TX of RS232
DATA 1	WEIGAND interface DATA HIGH pin
DATA 0	WEIGAND interface DATA LOW pin

3.4 EM-18 SPECIFICATION :-

- Operating voltage of EM-18: +4.5 V to +5.5 V
- Current consumption: Less than 50 mA
- Can operate on LOW power
- Operating temperature: 0 ºC to +80 ºC
- Operating frequency:125 KHz
- Communication parameter: 9600 bps
- Reading distance: 10 cm, depending on TAG
- Integrated Antenna

3.5 EM-18 RFID WORKING MECHANISM :-

- When an RFID tag comes in the range of signal transmitted by the reader (10 cm for EM18), the transponder of the tag is hit by this signal.

- RFID tag induced power from the electromagnetic field which is generated by the reader.

- Then, the transponder converts that radio signal into the usable power and after getting power, the transponder sends all the data/information it has stored in the chip, such as a unique ID to the RFID reader in the form of radio waves or **RF** signal.

- Then, RFID reader reads the data/unique ID (in the form of byte) & transmits it through serial **Tx** (transmit) pin from reader to **Rx** of microcontroller serially using **UART** communication.

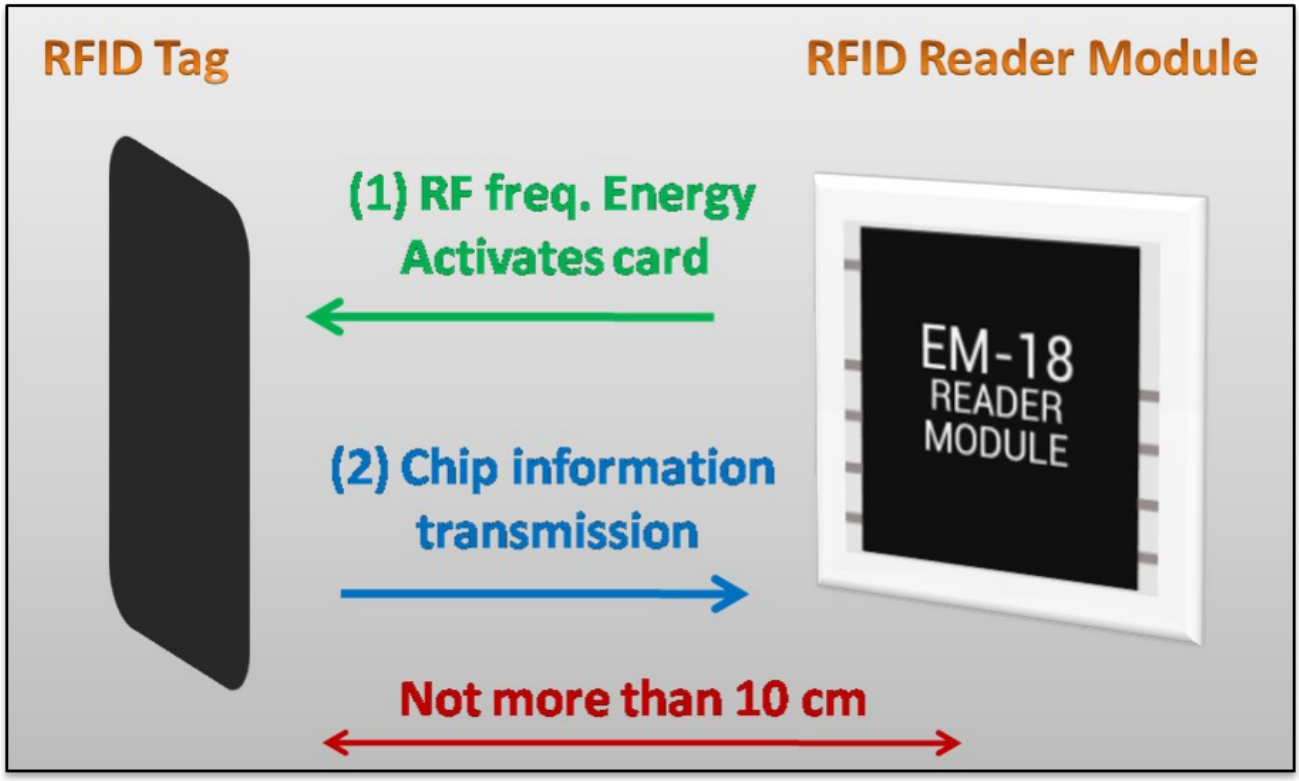


Figure 11: EM-18 RFID Working Mechanism

4. RFID TAGS :-

An RFID tag is a microchip combined with an antenna in a compact package; the packaging is structured to allow the RFID tag to be attached to an object to be tracked. "RFID" stands for Radio Frequency Identification.

The tag's antenna picks up signals from an RFID reader or scanner and then returns the signal, usually with some additional data (like a unique serial number or other customized information). RFID tags can be very small — the size of a large rice grain. Others may be the size of a small paperback book.

Figure 12: RFID Cards

Figure 13: RFID Tag Inside

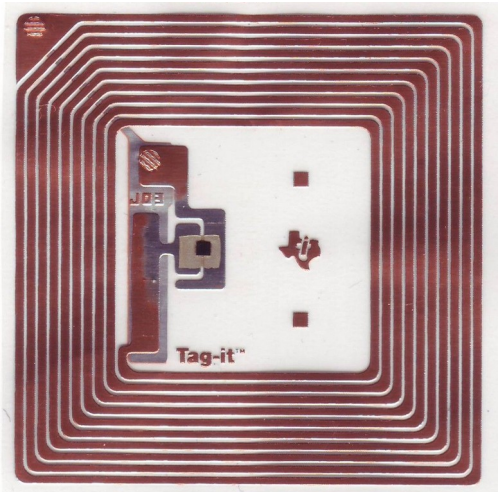


Figure 14: RFID Inside Coil

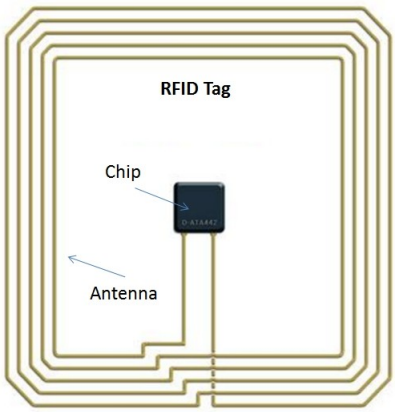


Figure 15: RFID Inside Coil 2

Figure 16: RFID Card Attached in a Car

RFID tag includes microchip with radio antenna mounted on substrate as shown in above figure 14 & 15, which carries 12 Byte unique Identification number. There are Two types of RFID tags:

- Active RFID Tag (Active Tag)
- Passive RFID Tag (Passive Tag)

4.1 Active RFID Tag (Active Tag) :-

An RFID tag is an active tag when it is equipped with a battery that can be used as a partial or complete source of power for the tag's circuitry and antenna. Some active tags contain replaceable batteries for years of use; others are sealed units. (Note that it is also possible to connect the tag to an external power source.)

4.1 a) The advantages of an active RFID tag are :-

- It can be read at distances of one hundred feet or more, greatly improving the utility of the device.
- It may have other sensors that can use electricity for power.

4.1 b) The disadvantages of an active RFID tag are :-

- The tag cannot function without battery power, which limits the lifetime of the tag.
- The tag is typically more expensive, often costing $20 or more each
- The tag is physically larger, which may limit applications.

4.2 c) Features of the Active RFID tags :-

- The Longest communication range of any tag
- The capability to perform independent monitoring and control
- The capability of initiating communications
- The capability of performing diagnostics
- The highest data bandwidth
- Active RFID tags may even be equipped with autonomous networking; the tags autonomously determine the best communication path

4.2 Passive RFID Tag (Passive Tag) :-

A passive tag is an RFID tag that does not contain a battery; the power is supplied by the reader. When radio waves from the reader are encountered by a passive RFID tag, the coiled antenna within the tag forms a magnetic field. The tag draws power from it, energizing the circuits in the tag. The tag then sends the information encoded in the tag's memory.

4.2 a) The advantages of a passive RFID tag are :-

- The tag functions without a battery; these tags have a useful life of twenty years or more.
- The tag is typically much less expensive to manufacture.
- The tag is much smaller (some tags are the size of a grain of rice). These tags have almost unlimited applications in consumer goods and other areas.

4.2 b) The disadvantages of a passive RFID tag are :-

- The tag can be read only at very short distances, typically a few feet at most. This greatly limits the device for certain applications.
- It may not be possible to include sensors that can use electricity for power.
- The tag remains readable for a very long time, even after the product to which the tag is attached has been sold and is no longer being tracked.

5.LIQUID CRYSTAL DISPLAY (LCD)

Nowadays, we always use the devices which are made up of LCDs such as CD players, DVD players, digital watches, computers, etc. These are commonly used in the screen industries to replace the utilization of CRTs. Cathode Ray Tubes use huge power when compared with LCDs, and CRTs heavier as well as bigger. These devices are thinner as well power consumption is extremely less. The LCD 16×2 working principle is, it blocks the light rather than dissipate.

5.1 What is the LCD 16×2?

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

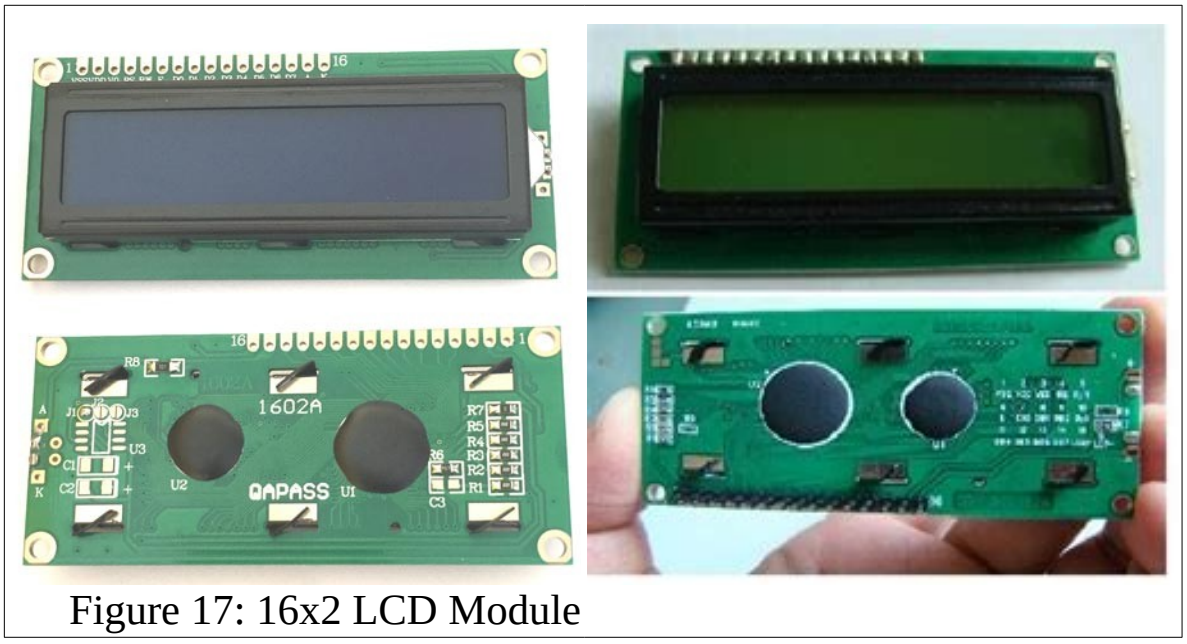


Figure 17: 16x2 LCD Module

Figure 18: 16x2 Green LCD

Figure 19: 16x2 Blue LCD

5.2 Features of 16×2 LCD module :-

- The operating voltage of this LCD is 4.7V-5.3V
- The utilization of current is 1mA with no backlight.
- The alphanumeric LCD's, meaning can display alphabets and numbers.
- It consists of two rows and each row can print 16 characters.
- Every character can be built with a 5×8 pixel box.
- This display can work on two modes like 4-bit & 8-bit
- It can also display any custom generated characters
- These are obtainable in Blue & Green Backlight

5.3 Registers of LCD :-

A 16×2 LCD has two registers like data register and command register. The RS (register select) is mainly used to change from one register to another. When the register set is '0', then it is known as command register. Similarly, when the register set is '1', then it is known as data register.

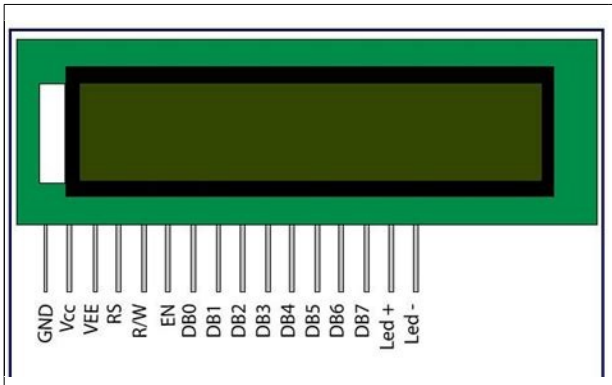
5.3 a) Command Register :-

The main function of the command register is to store the instructions of command which are given to the display. So that predefined tasks can be performed such as clearing the display, initializing, set the cursor place, and display control. Here commands processing can occur within the register.

5.3 b) Data Register :-

The main function of the data register is to store the information which is to be exhibited on the LCD screen. Here, the ASCII value of the character is the information which is to be exhibited on the screen of LCD. Whenever we send the information to LCD, it transmits to the data register, and then the process will be starting there. When register set =1, then the data register will be selected.

5.4 Pin Configuration of 16×2 LCD :-



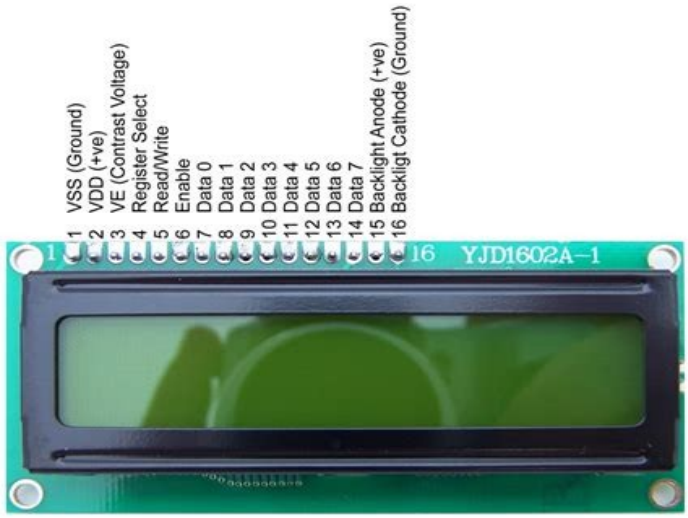


Figure 20: Pin Diagram

PIN No	Name	Function
1	VSS	Ground voltage
2	VCC	+5V
3	VEE	Contrast voltage
4	RS	Register Select 0 = Instruction Register 1 = Data Register
5	R/W	Read/ Write, to choose write or read mode 0 = write mode 1 = read mode
6	E	Enable 0 = start to latch data to LCD character 1= disable
7	DB0	Data bit 0 (LSB)
8	DB1	Data bit 1
9	DB2	Data bit 2
10	DB3	Data bit 3
11	DB4	Data bit 4
12	DB5	Data bit 5
13	DB6	Data bit 6
14	DB7	Data bit 7 (MSB)
15	BPL	Back Plane Light +5V or lower (Optional)
16	GND	Ground voltage (Optional)

Figure 21: Pin Configuration

5.4 a) Explanation :-

- Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
- Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.
- Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.
- Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1(1 = data mode, and 0 = command mode).
- Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).
- Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit & constantly held high.
- Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.
- Pin15 (+ve pin of the LED): This pin is connected to +5V
- Pin 16 (-ve pin of the LED): This pin is connected to GND.

The LCD requires 3 control lines (RS, R/W & EN) & 8 (or 4) data lines. The number on data lines depends on the mode of operation. If operated in 8-bit mode then 8 data lines + 3 control lines i.e., total 11 lines are required. And if operated in 4-bit mode then 4 data lines + 3 control lines i.e., 7 lines are required. How do we decide which mode to use? It's simple if you have sufficient data lines you can go for 8 bit mode & if there is a time constrain i.e., display should be faster, then we have to use 8-bit mode because basically 4-bit mode takes twice as more time as compared to 8-bit mode.

When RS is low (0), the data is to be treated as a command. When RS is high (1), the data being sent is considered as text data which should be displayed on the screen.

When R/W is low (0), the information on the data bus is being written to the LCD. When RW is high (1), the program is effectively reading from the LCD. Most of the time there is no need to read from the LCD, so this line can directly be connected to Gnd thus saving one controller line.

The ENABLE pin is used to latch the data present on the data pins. A HIGH – LOW signal is required to latch the data. The LCD interprets and executes our command at the instant the EN line is brought low. If you never bring EN low, your instruction will never be executed.

5.5 Command codes for 16×2 LCD :-

HEX Code	Function Of Commands
0F	Turn on the Display and Cursor
01	It will clear the screen
02	The cursor will return to the (0,0) position
04	It will shift the cursor to the left
05	It will shift the cursor to the right
06	The display will shift to the right
07	The display will shift to the left
0E	The display will on and the cursor will start blinking
80	The following command will force the cursor to start at the beginning.
C0	It will make the cursor to start from the second line
38	For 5×7 Matrix with 2 lines
83	Cursor at first line but third position
3C	It will activate the second line
08	The following command will turn off the display and the cursor.
C1	It will make the jump to the second line but at position one
0C	The display will on, but now there won't be any cursor on the screen.
C2	The following command will make the jump to the second line and position 2.

6. ULN2003A IC

Microcontroller has an internal program or as we can say a set of functions, which control the voltage at the output and input pins of a single Control Unit. These control functions also help us to generate the timer, PWM, interrupts, and switching method internally without affecting the whole controller circuit. The issue of generating multiple functions was solved by a simple controller.

Now the problem was how to control and minimize the circuit of high voltage DC devices. Microcontrollers operate at 5V while motors operate at different voltages (5V, 12V, 24V etc.). There was a wide usage of High voltage DC motors due to its energy efficiency. A logical circuit with Darlington transistor (NPN) was used to control the High DC load approximately equal to 50V and 500mA. This circuit could only be used for a single load. To solve this issue an IC name ULN2003 was introduced.

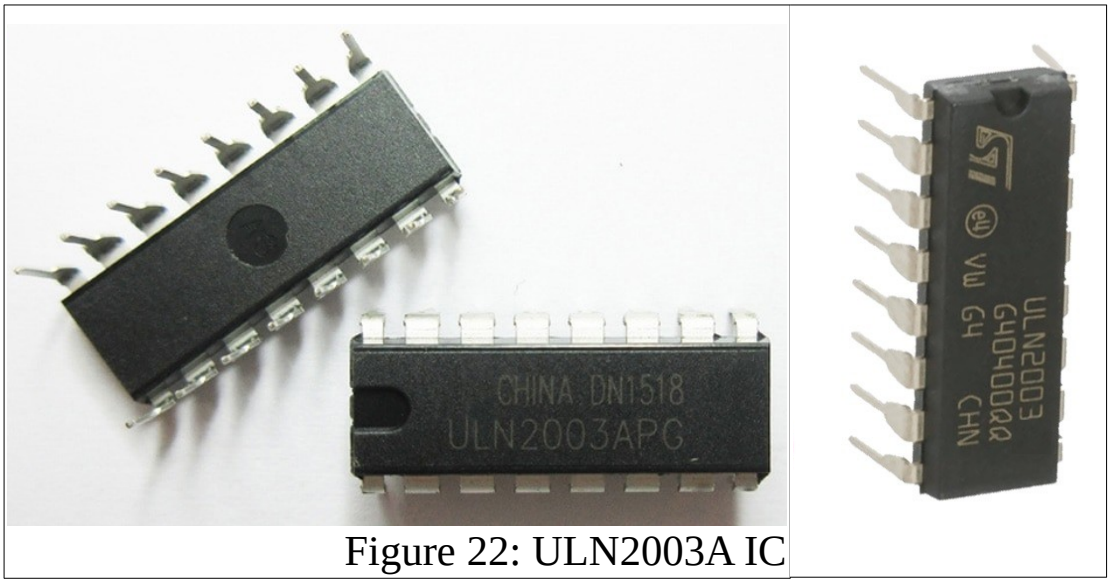


Figure 22: ULN2003A IC

ULN2003A comes with multiple functions. It has seven Darlington transistors installed which could help to control 7-Loads at the same time. It comes with 16 pins and multiple packing's like SOP, PDIP, TSSOP or SOIC. This could help the user to install the IC with any circuit without taking too much space as a transistors circuit. The output power source could apply separately to all the outputs, but the input will be the same as all microcontrollers.

The voltage range for any load is 50V, but the current range is 500mA which could be increased by combining the multiple output pins. ULN2003 comes with internal safety protection from back emf. It has an internal fly-back protection system which gives protection to the device.

ULN2003: Definition :-

- ULN2003 is a 16 Pin IC, consisting of 7 Darlington pairs (each pair protected with suppression diode) and thus has the capability to handle maximum 7 loads (could be inductive).

- In simple words, we have 7 drivers in a single ULN2003 chip and thus can control maximum 7 loads.

- Each Darlington pair can handle maximum 500mA load, while peak value is 600mA.

- Similarly, maximum output voltage of each Darlington pair is 50V.

- ULN2003 has 16 Pins, where inputs and their respective outputs are placed in-front of each other (for ease of circuit designing).

- Other than I/O Pins, we have Ground Pin where we need to provide 0V & Vcc (Common) Pin.

6.1 ULN2003A Pinout :-

- ULN2003A has 16 pins in total :-
  - 7 Input pins (Pin # 1 to Pin # 7)
  - 7 Output pins (Pin # 10 to Pin # 16)
  - 1 Ground pin (Pin # 8)
  - 1 COM pin (Pin # 9)

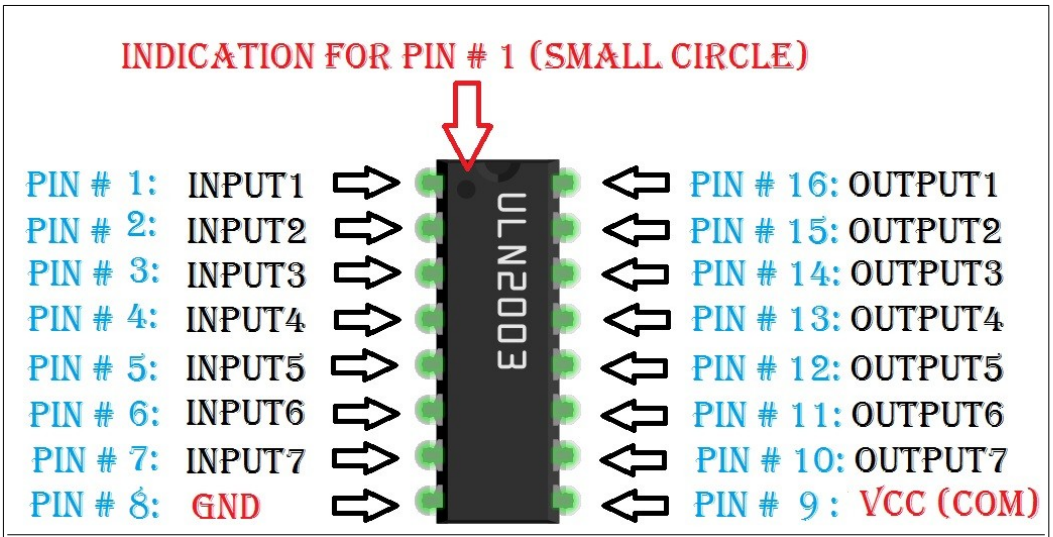


Figure 23: ULN2003A Pinout

Pin No	Name	Function	Description
1	In1	Input Pair 1	Input for 1st Channel
2	In2	Input Pair 2	Input for 2nd Channel
3	In3	Input Pair 3	Input for 3rd Channel
4	In4	Input Pair 4	Input for 4th Channel
5	In5	Input Pair 5	Input for 5th Channel
6	In6	Input Pair 6	Input for 6th Channel
7	In7	Input Pair 7	Input for 7th Channel
8	Ground	Common Emitter (0V)	Ground (0V)
9	Common	Common Clamp Diodes	Common Free Wheeling Diodes
10	Out7	Output Pair 7	Output for 7th Channel
11	Out6	Output Pair 6	Output for 6th Channel
12	Outt5	Output Pair 5	Output for 5th Channel
13	Out4	Output Pair 4	Output for 4th Channel
14	Out3	Output Pair 3	Output for 3rd Channel
15	Out2	Output Pair 2	Output for 2nd Channel
16	Out1	Output Pair 1	Output for 1st Channel

Figure 24: ULN2003A Pin Description

6.3 ULN2003A Darlington Pair & Free Wheeling Diodes :-

- ULN2003A consists of 7 identical Darlington pairs.
- A single Darlington pair consists of two bipolar transistors its maximum operating values are 50V & 500mA (peak 600mA).
- These two transistors of Darlington pair have common emitter while their collectors are open.
- ULN2003A has freewheeling diodes, which protects from back emf.
- So, if we are using any inductive load (i.e. relays), then we don't need to add extra diodes, if we are controlling it with ULN2003A.

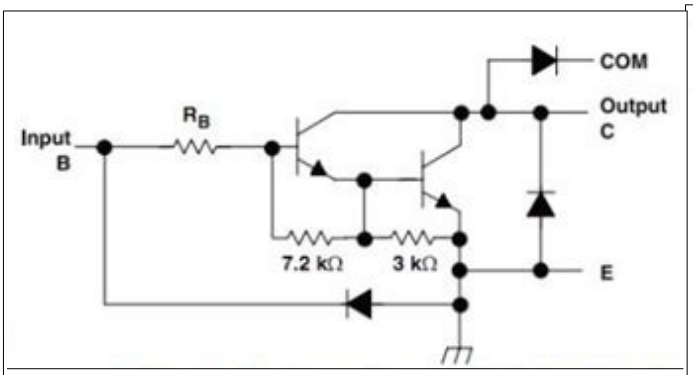


Figure 26: Darlington Circuit

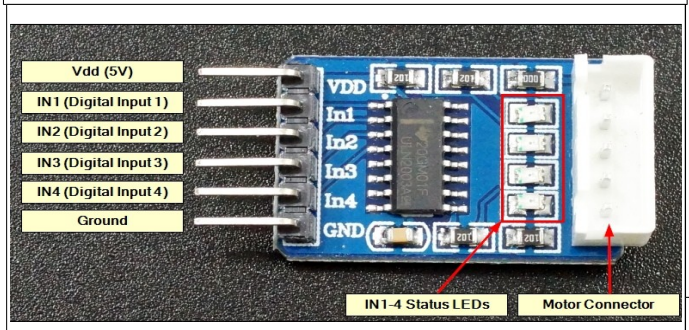


Figure 27: ULN2003A Stepper Motor Driver

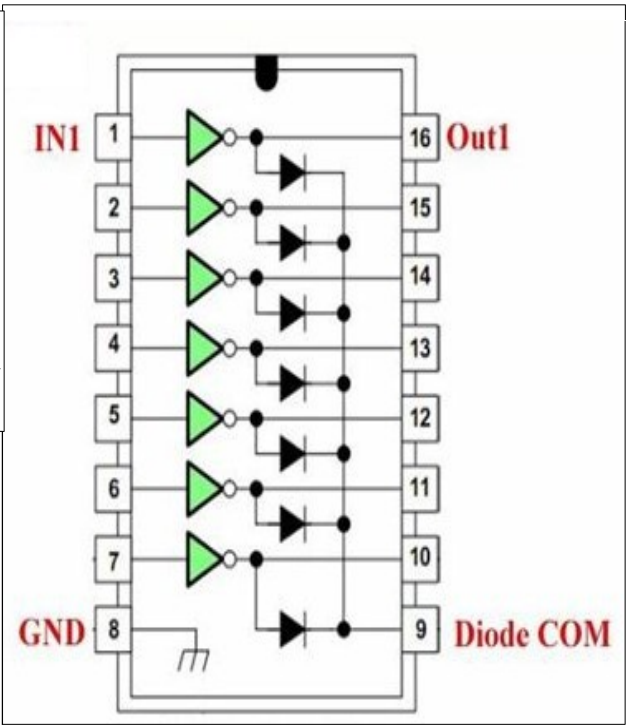


Figure 25: ULN2003A Logic Diagram

The ULN2003A stepper motor driver PCB provides a direct drive interface between microcontroller and stepper motor. The PCB provides 4 inputs for connection to microcontroller, power supply connection for the stepper motor voltage, and ON/OFF jumper, a direct connect stepper motor header and 4 LEDs to indicate stepping state. The driver board accepts a four bit command from any microcontroller and in turn applies the necessary power pulse to step the motor. At the heart of the driver is a ULN2003AN integrated circuit. The board can supply between 5V to 12V to the motor from an independent power supply. It also has a bank of LED's that correspond to the input signals received from the controller. They provide a nice visual when stepping.

6.4 Features Of ULN2003A IC :-

- It is able to handle the high DC voltage range of about 50V maxima
- It comes in another version with a voltage range of 100V.
- The current handling system is also 500mA for each input.
- The range of current could be increased by using the two pins for the same load.
- Comes with an internal clap diode to protect the device from Back EMF Protection
- ULN2003A has an internal fly-back system protection and a pin too which can be used for an inductive load.
- We can control by any low voltage device like Arduino, Microprocessor or any other controller or IC.
- It is available in all kind of packages like SOP, PDIP, TSSOP or
- ULN2003 output is compatible with all the TTL and 5-V CMOS logic
- It operates without using any power source attached directly to it.

Features	Values	Units
Drivers per package	7	--
I_{out} Typical	50	µA
I_{out} Max	500	mA
Peak output current	500	mA
Delay time	250	Ns
Output voltage	50	V
Switching voltage	50	V
Input compatibility	CMOS TTL	--
Rating	Catalog	--
Pin per package	16SOIC, 16SO, 16TSSOP, 16PDIP	--
Voltage @ lowest current	900	mV

Figure 28: ULN2003A Features

7. STEPPER MOTOR.

The invention of dedicated stepper motor driver cards and other digital control technologies for interfacing a stepper motor to PC-based systems are the reason for the widespread acceptance of stepper motors in recent times. Stepper motors become the ideal choice for automation systems that require precise speed control or precise positioning or both.

As we know that many industrial electric motors are used with closed-loop feedback control for achieving precise positioning or precise speed control, on the other hand, a stepper motor able to operate on an open-loop controller. This in turn reduces the total system cost and simplifies the machine design compared with servo system control.

7.1 What is a Stepper Motor?

Stepper Motor is a brushless electromechanical device which converts the train of electric pulses applied at their excitation windings into precisely defined step-by-step mechanical shaft rotation. The shaft of the motor rotates through a fixed angle for each discrete pulse. This rotation can be linear or angular. It gets one-step movement for a single pulse input.

When a train of pulses is applied, it gets turned through a certain angle. The angle through which the stepper motor shaft turns for each pulse is referred as the step angle, which is generally expressed in degrees.

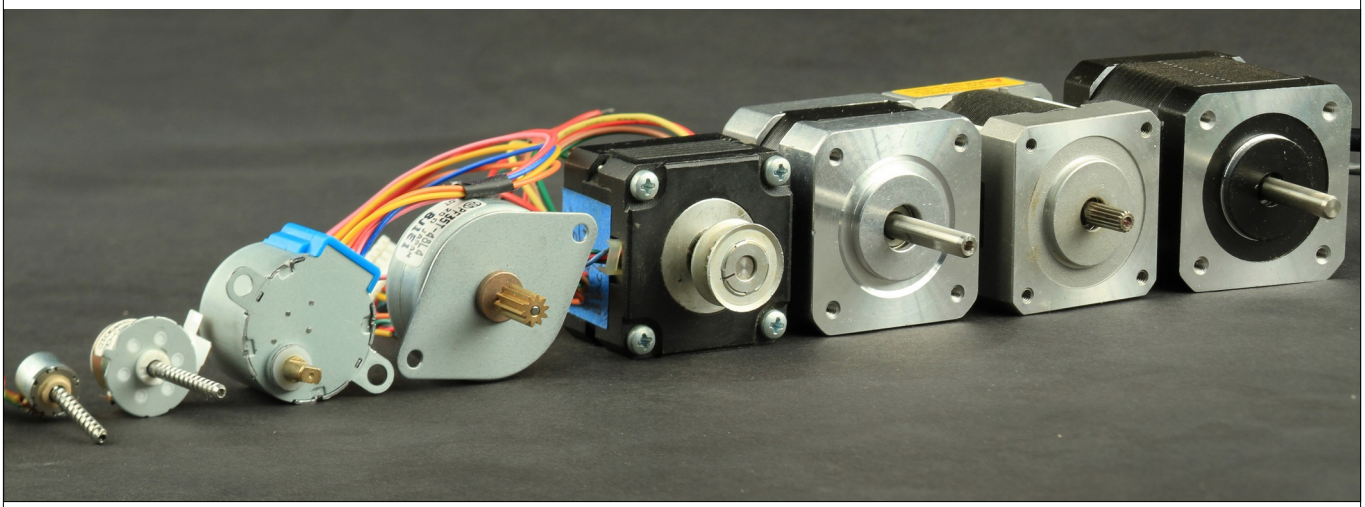


Figure 29: Stepper Motors

The number of input pulses given to the motor decides the step angle and hence the position of motor shaft is controlled by controlling the number of pulses. This unique feature makes the stepper motor to be well suited for open-loop control system wherein the precise position of the shaft is maintained with exact number of pulses without using a feedback sensor.

If the step angle is smaller, the greater will be the number of steps per revolutions and higher will be the accuracy of the position obtained. The step angles can be as large as 90 degrees and as small as 0.72 degrees, however, the commonly used step angles are 1.8 degrees, 2.5 degrees, 7.5 degrees and 15 degrees.

7.2 Construction & Working Principle :-

The **construction of a stepper motor** is fairly related to a DC motor. It includes a permanent magnet like Rotor which is in the middle & it will turn once force acts on it. This rotor is enclosed through a no. of the stator which is wound through a magnetic coil all over it. The stator is arranged near to rotor so that magnetic fields within the stators can control the movement of the rotor.

The stepper motor can be controlled by energizing every stator one by one. So the stator will magnetize & works like an electromagnetic pole which uses repulsive energy on the rotor to move forward. The stators alternative magnetizing as well as demagnetizing will shift the rotor gradually &allows it to turn through great control.

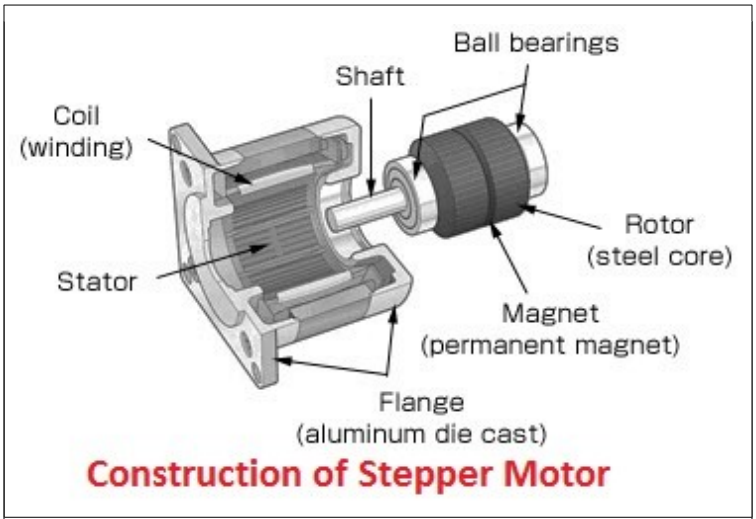


Figure 30: Construction of Stepper Motor

The **stepper motor working principle** is Electromagnetism. It includes a rotor which is made with a permanent magnet whereas a stator is with electromagnets. Once the supply is provided to the winding of the stator then the magnetic field will be developed within the stator. Now rotor in the motor will start to move with the rotating magnetic field of the stator. So this is the fundamental working principle of this motor.

In this motor, there is a soft iron that is enclosed through the electromagnetic stators. The poles of the stator as well as the rotor don't depend on the kind of stepper. Once the stators of this motor are energized then the rotor will rotate to line up itself with the stator otherwise turns to have the least gap through the stator. In this way, the stators are activated in a series to revolve the stepper motor.

7.3 Types of Stepper Motors :-

- Permanent Magnet Stepper Motor
- Variable Reluctance Stepper Motor
- Hybrid Synchronous Stepper Motor

7.4 Unipolar and Bipolar Stepper Motors :-

The above-mentioned motors can be unipolar or bipolar based on the coil winding arrangements. A unipolar motor is employed with two windings per phase and hence the direction of current flow through these windings changes the rotation of the motor. In this configuration, the current flow is through one direction in one coil and opposite direction in another coil.

The figure below shows 2-phase unipolar stepper motor wherein A and C coils are for one phase and B and D are for other phase. In each phase each coil carries current in opposite direction to that of other coil. Only one coil will be carrying current at a time in each phase for achieving particular direction of rotation. So just by switching the terminals to each coil, the direction of rotation is controlled.

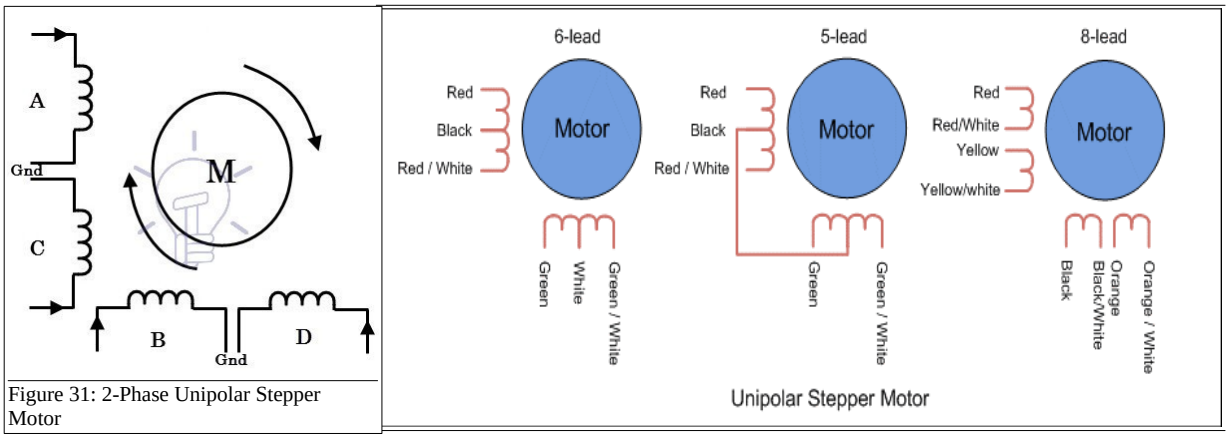


Figure 31: 2-Phase Unipolar Stepper Motor

In case of a bipolar stepper motor, each phase consists of a single winding rather than two in case of unipolar one. In this, the direction of rotation is controlled by reversing the current through the windings. Hence, it requires a complex drive circuit for current reversal.

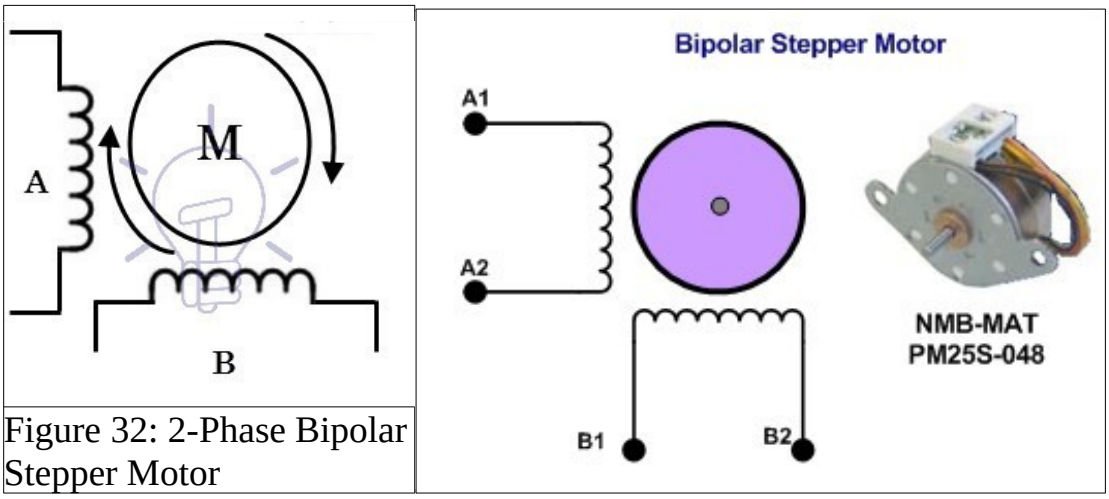


Figure 32: 2-Phase Bipolar Stepper Motor

7.5 Stepping Modes of a Stepper Motor

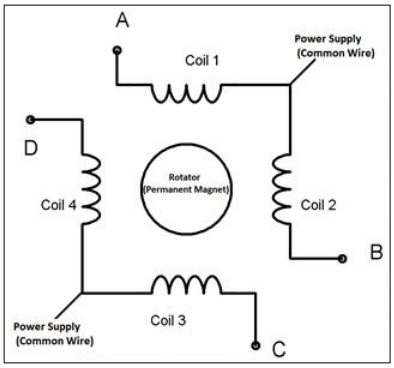
A typical stepping action causes the motor to step through a sequence of equilibrium positions in response to current pulses given to it. It is possible to vary the stepping action in different ways simply by changing the sequence through which stator windings are energized. The following are the most common operating or driving modes of stepper motors.

a) Wave step

b) Full step

c) Half step

d) Micro-stepping



7.5 a) Wave Step Mode :-

Wave step mode is the simplest of all modes in which only one winding is energized at any given time. Each coil of the phase is connected to the supply alternatively. The table below shows the order through which coils are energized in a 4-phase stepper motor.

In this mode motor gives maximum step angle compared to all other modes. It is the simplest and most commonly used mode for stepping; however the torque produced is less as it uses some part of the total winding at a given time.

Step	Coil A	Coil B	Coil C	Coil D
1	ON	OFF	OFF	OFF
2	OFF	ON	OFF	OFF
3	OFF	OFF	ON	OFF
4	OFF	OFF	OFF	ON

Figure 33:Signal Sequence of Wave Step Mode

7.5 b) Full Step Mode :-

In this drive or mode, two stator phases are energized simultaneously at any given time. When two phases are energized together, the rotor will experience the torque from both phases and comes to the equilibrium position, which will be interleaved between two adjacent wave step positions or 1-phase excitations. So this step provides better holding torque than wave step. The table below shows the full step drive for 4-phase stepper motor.

Step	Coil A	Coil B	Coil C	Coil D
1	ON	ON	OFF	OFF
2	OFF	ON	ON	OFF
3	OFF	OFF	ON	ON
4	ON	OFF	OFF	ON

Figure 34: Signal Sequence of Full Step Drive

7.6 c) Half Step Mode :-

It is the combination of both wave and full step modes. In this, single phase and dual phase excitations are carried out alternatively, i.e., one-phase ON, two-phases ON, and so on. The step angle in this mode becomes half of the full step angle. This drive mode has the highest torque and stability compared to all other modes. The table containing phase pulsing sequence for a 4-phase motor in half stepping is given below.

Step	Coil A	Coil B	Coil C	Coil D
1	ON	OFF	OFF	OFF
2	ON	ON	OFF	OFF
3	OFF	ON	OFF	OFF
4	OFF	ON	ON	OFF
5	OFF	OFF	ON	OFF
6	OFF	OFF	ON	ON
7	OFF	OFF	OFF	ON
8	ON	OFF	OFF	ON

Figure 35: Signal Sequence of Half Step Drive

7.6 d) Micro-stepping Mode :-

In this mode, each motor step is subdivided into several small steps, even hundreds of fixed positions, therefore a greater positioning resolution is obtained. In this, currents through the windings are continually varied in order to get very small steps. In this, two phases are excited simultaneously, but with the unequal currents in each phase.

For example, the current through phase -1 is held constant while the current through phase-2 is incremented in steps till the maximum value of current, whether it is negative or positive. The current in the phase-1 is then decreased or increased in steps till zero. Thus, the motor will produce a small step size.

7.7 Parameters of Stepper Motor

The stepper motor parameters mainly include step angle, steps for each revolution, steps for each second, and RPM.

Step Angle :-

The step angle of the stepper motor can be defined as the angle at which the motor's rotor turns once a single pulse is given to the stators input. The resolution of the motor can be defined as the number of steps of the motor and the number of revolutions of the rotor.

Resolution = Number of Steps/Number of Revolution of the Rotor

The motor's arrangement can be decided through the step-angle & it is expressed within degrees. The resolution of a motor (the step number) is the no. of steps which make within a single revolution of the rotor. When the step-angle of the motor is small then the resolution is high for the arrangement of this motor.

The exactness of the arrangements of the objects through this motor mainly depends on the resolution. Once the resolution is high then the accuracy will be low.

Some accuracy motors can create 1000 steps within a single revolution including 0.36 degrees of step-angle. A typical motor includes 1.8 degrees of step angle with 200 steps for each revolution. The different step angles such as 15 degrees, 45 degrees, and 90 degrees are very common in normal motors. The number of angles can change from two to six and a small step angle can be attained through slotted pole parts.

Steps for Each Revolution

The steps for each resolution can be defined as the number of step angles necessary for a total revolution. The formula for this is 360°/Step Angle.

Steps for Each Second

This kind of parameter is mainly used for measuring the number of steps covered within each second.

Revolution per Minute

The RPM is the revolution per minute. It is used to measure the frequency of revolution. So by using this parameter, we can calculate the number of revolutions in a single minute. The main relation between the parameters of the stepper motor is like the following.

Steps for Each Second = Revolution per Minute x Steps per Revolution / 60

SOFTWARE DESCRIPTION:

1. KEIL MVISION

Keil µVision is used for Programming. Keil is a German software subsidiary of Arm Holdings. It was founded in 1982 by Günter and Reinhard Keil, initially as a German GbR. In April 1985 the company was converted to Keil Elektronik GmbH to market add-on products for the development tools provided by many of the silicon vendors. Keil implemented the first C compiler designed from the ground-up specifically for the 8051 microcontroller.

- Keil Software to provide us with IDE (Integrated development environment)
- Tools for the 8051 and 251 family of microprocessors.
- With this tool, you can generate embedded applications for the multitude of 8051 and 251 derivatives.
- 8051 and 251 development took are listed below :
    - C51/C251 Optimizing C Cross-Compiler.
    - A51/A251 Macro Assembler.
    - 8051/251 Utilities (linker, object file converter, library manager).
    - dScope for Windows™ Source-Level Debugger/Simulator.
    - Vision for Windows™ Integrated Development Environment.

2. PROTEUS DESIGN SUITE.

The Proteus Design Suite is a proprietary software tool suite used primarily for electronic design automation. The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards.

It was developed in Yorkshire, England by Labcenter Electronics Ltd and is available in English, French, Spanish and Chinese languages.

2.1 General Description :-

- Proteus 8.0 is a Virtual System Modeling (VSM) that combines circuit simulation, animated components and microprocessor models to co-simulate the complete microcontroller based designs. This is the perfect tool for engineers to test their microcontroller designs before constructing a physical prototype in real time. This program allows users to interact with the design using on-screen indicators and/or LED and LCD's and, if attached to the PC, switches and buttons.
- One of the main components of Proteus 8.0 is the Circuit Simulation -- a product that uses a SPICE365 analogue simulator kernel combined with an event-driven digital simulator that allow users to utilize any SPICE model by any manufacturer.
- The micro-controller simulation in Proteus works by applying either a hex file or a debug file to the microcontroller part on the schematic. It is then co-simulated along with any analog and digital electronics connected to it. This enables its use in a broad spectrum of project prototyping in areas such as motor control, temperature control and user interface design. It also finds use in the general hobbyist community and, since no hardware is required, is convenient to use as a training or teaching tool.
- Proteus VSM comes with extensive debugging features, including breakpoints, single stepping and variable display for a neat design prior to hardware prototyping.
- In summary, Proteus 8.0 is the program to use when you want to simulate the interaction between software running on a microcontroller and any analog or digital electronic device connected to it.

3. SSTFLASHFLEX51

EasyIAP™ is a boot-strap loader (BSL) software that enables SST FlashFlex customers to download/upload their application software into/from the FlashFlex flash memory via In-Application Programming (IAP) while their system is running. This Application is used to burn the Hex file into microcontroller. Important features are: data download/upload, self-detection of the serial connection, and automatic reset after download.

CIRCUIT DIAGRAM :

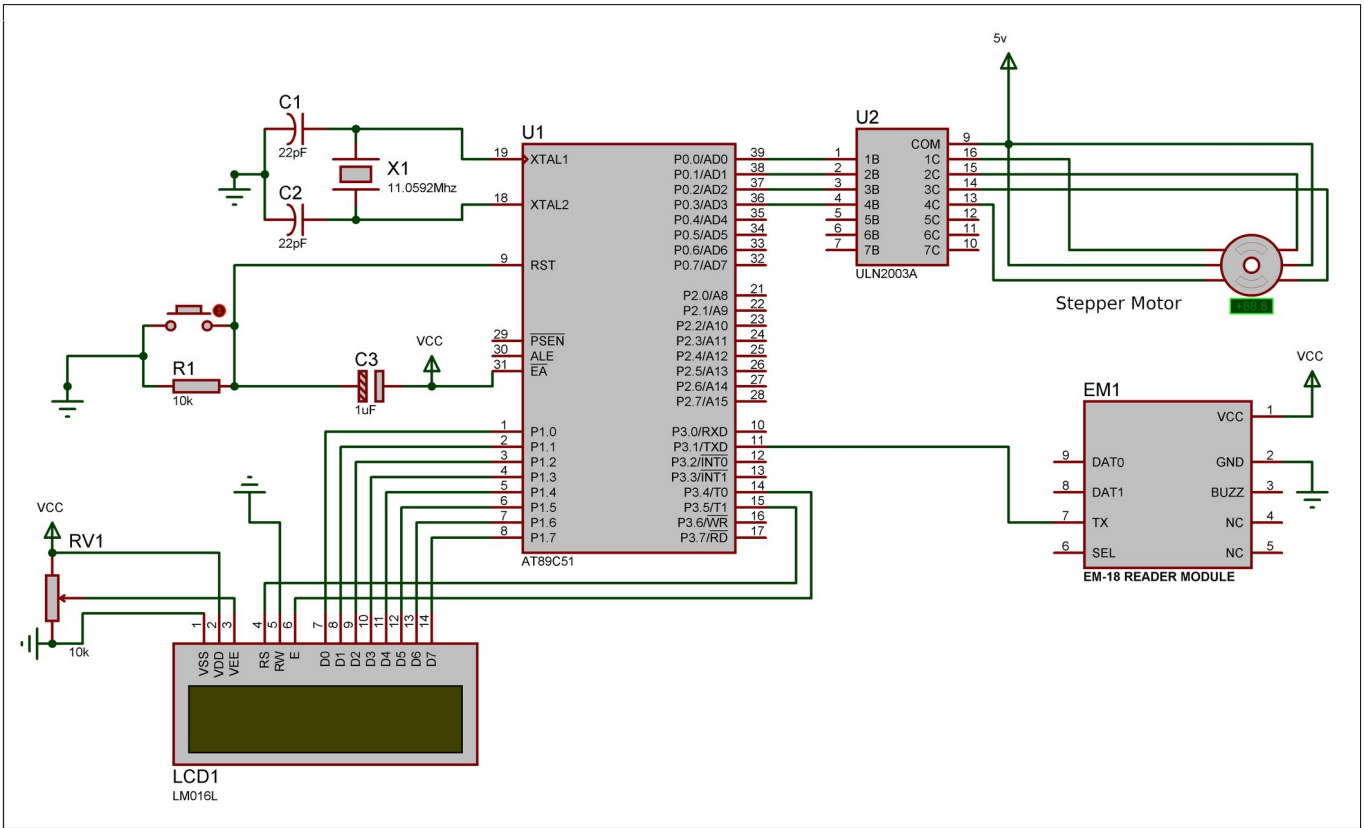


Figure 36: RFID Based Automatic Gate Circuit Diagram

Description :-

The circuit diagram shows the connection of RFID and Stepper Motor with microcontroller. Port P0 is used for stepper motor connection, Port P1 is used as data port for LCD and P3.4, P3.5; The pin P3.1 is used for RFID reader which transmits data seriously.

When car owner comes LCD shows the message "Automatic Gate", "Scan your ID". When the RFID card scans, the RFID reader readers the card and decodes the information and sends the decoded information (12 Unique Number) to the microcontroller. The microcontroller checks the received information and compares that information with already saved information. It compares with all the saved information, when the comparison is true (saved = received), microcontroller sends the signal to Stepper Motor and card information is showed on LCD. As the Stepper Motor gets the commands from microcontroller it starts to rotate. It rotates 90 degree waits for 10 seconds and reverse back to original position.

If the comparison is false (saved ≠ received), microcontroller sends the message "Wrong ID" to LCD & it displays. Stepper Motor doesn't rotate.

TROUBLESHOOTING

- We tested output of LCD with microcontroller to check it's properly working.

- We tested the Stepper Motor by rotating it, using the microcontroller to check it's properly working.

- The RFID reader module was checked by connecting to PC with the help of USB to UART (CP2102) converter and then by using HyperTerminal Application.

- Initially RFID reader module breakout board was not properly worked.

- Instead of use of RFID reader module breakout board, we used just RFID reader module.

Figure 37: USB to UART Converter

HOW TO RUN PROJECT

1. Initially burn the program to the 8051 microcontroller.
2. Now give the connections as per the circuit diagram.
3. Switch on the board supply.
4. The LCD shows the message “Please Scan Your ID”.
5. Scan the correct RFID card, LCD shows the ID number and shows the message “Opening the Gate”.
6. Gate will open waits for 10 second and returns to its original position.
7. If the scanned RFID card is incorrect, LCD shows the message “Wrong ID” and gate won’t open.

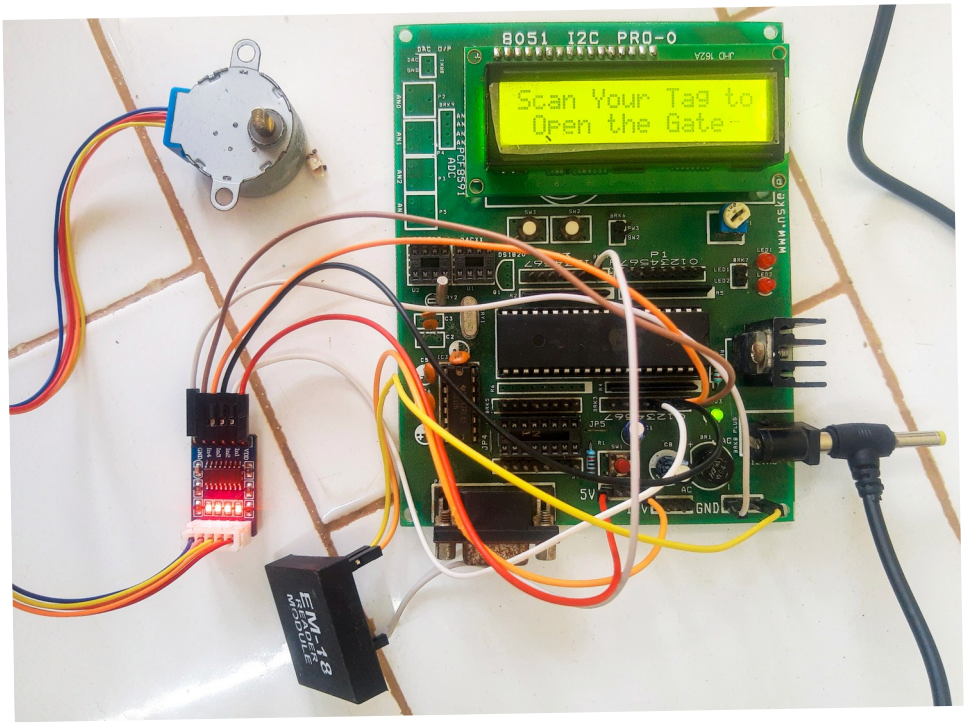


Figure 38: Basic Prototype of Project

FUTURE SCOPE OF THE PROJECT :

1. **Automatic Vehicle Identification:** The automatic vehicle identification (AVI) component of this system refers to the technologies that determine the identification or ownership of the vehicle so that the toll will be charged to the corresponding customer.

2. **Automatic Vehicle Classification:** Vehicle type and class may have differentiated toll amount. The vehicle type may include light vehicles like the passenger car or heavy vehicles like recreational vehicles. A vehicle's class can be determined by the physical attributes of the vehicle, the number of occupants in the vehicle, the number of axles in the vehicles and the purpose for which the vehicle is being used at the time of classification.

3. **Video Enforcement System:** When used for electronic toll collection, the video enforcement system (VES) captures images of the license plates of vehicles that pass through an electronic tollbooth without a valid electronic tag. Although the deployment of these technologies make the initial cost of installation very high, but there exists huge benefits accompanied by such high investment. These benefits are discussed in the upcoming section.

CONCLUSION

The Automatic gate system is based on RFID, a design scheme was put forward. It is low cost, high security, far communication and efficiency, etc. Automatic gate system using RFID is an effective measure to reduce management costs and fees, at the same time, greatly reduce noise and pollutant emission. By effectively utilizing RFID technique in the project we are able to represent the automation in gate, which will reduce the complete processing time by few seconds which is very important as well as helps to reduce money leakage in a very cost-effective manner.

BIBLIOGRAPHY :

- www.wikipedia.com
- www.scribd.com
- www.technovelgy.com
- www.engineersgarage.com
- www.circuitstoday.com
- www.8051projects.net
- www.electronicshub.org
- www.electricaltechnology.org
- www.elprocus.com

DATA SHEET :

- www.components101.com
- www.alldatasheet.com
- www.nskelectronics.com/in
- www.corelectronics.com