

**RAILWAY LEVEL CROSSING GATE CONTROL BY
ANDROID APPLICATION**

CHAPTER 1

INTRODUCTION

1.1 EXISTING SYSTEM

The opening and closing of railway gate is traditionally operated through manual lever pulling method. This method leads to a lot of accidents due to the rational technique and lever jamming. Railway is a lifeline of India and it is being the cheapest modes of transportation are prepared over all other means of transportation. When we go through the daily News Papers we come across many accidents in rail road railings. Rail road related accidents are more dangerous than other transportation accidents in terms of severity and death rate etc. Therefore more effort are necessary for improving safety.

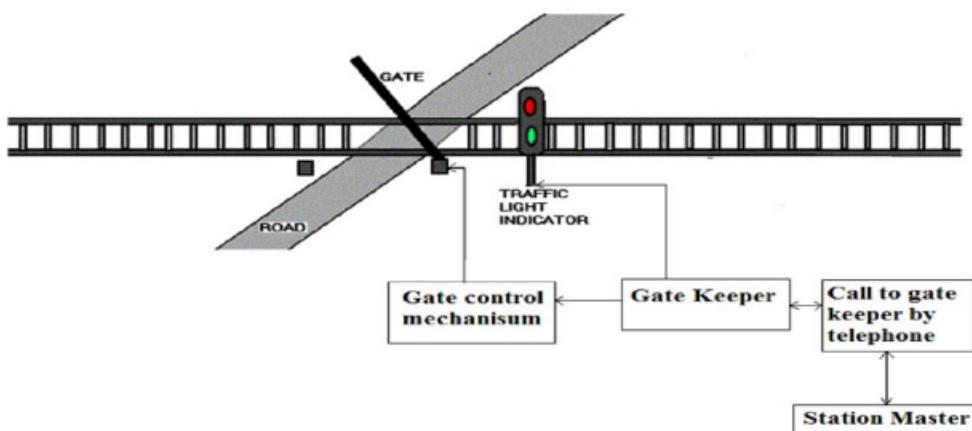


Fig 1.1

In the above fig 1.1 It says that currently the system used for the opening and closing of railway gate is traditionally operated through manual lever pulling method.

1.2 DISADVANTAGES OF EXISTING SYSTEM

- Gate can be manually pulled and release the gate but this consume man power.
- Possible to occur the human death.
- As the system requires human operator the human error affects the system.

1.3 PROPOSED SYSTEM

In this project a Smart-phone mobile is used which contain Android app connect to railway. This Bluetooth module sends data to Micro-controller. Due to Bluetooth app, Now this signal goes to motor driver IC LM293D which has two input pins and two output pins and four ground pins. It require 12V supply. It drives the signal in specific direction. This signal goes to DC motor . DC Motor is used to open or close the gate. It works according to micro-controller. Whenever pin is set gate remains close and when pin reset becomes active, gate remains open.SO, Firstly gate remains open at railway gate as there is no any train passes from railway gate. Controller access signal from module and follows condition of set according to incoming signal and passes to DC motor through motor driver. Then DC motor close the gate. In this way DC motor rotates in clockwise and anticlockwise direction to open or close the gate. Whenever gate open or close, the status of gate can display on LCD and Alert through Buzzer.

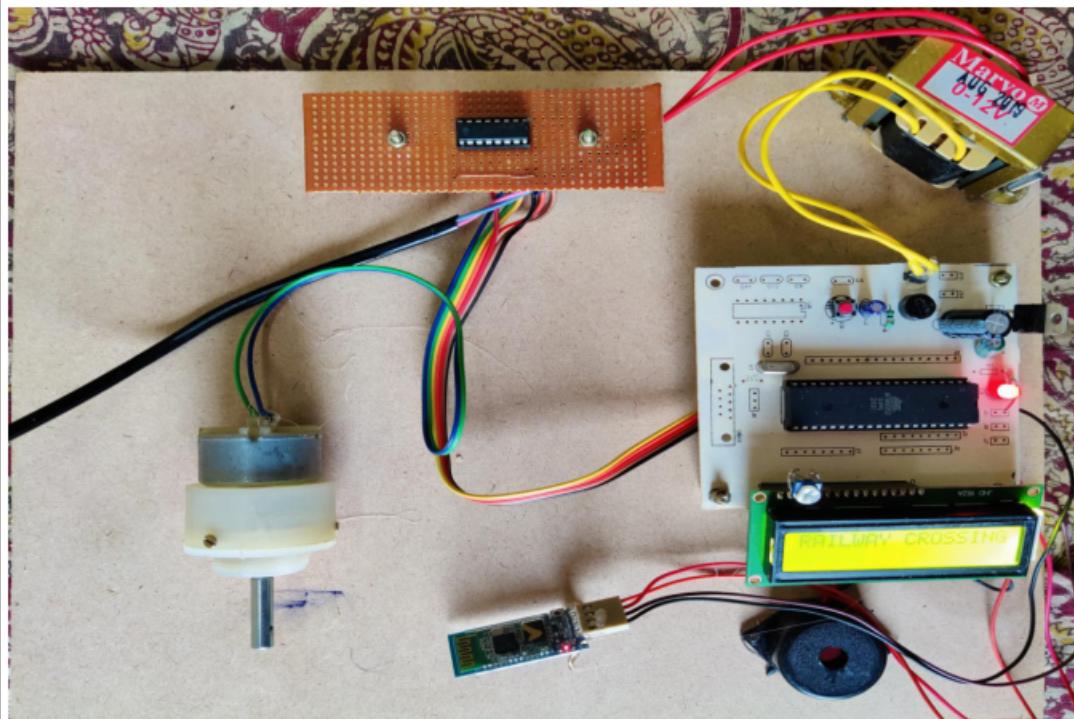


Fig 1.2

The fig 1.2 explains that system a smart-phone mobile is used which contain Android app connect to railway.

1.4 ADVANTAGES OF PROPOSED SYSTEM

- Low cost.
- Prevention of accidents inside the gate.
- Real time traffic status.
- Time saving as the gate keeper operates the system through mobile application.
- High flexibility.
- Reduces the human effect.

CHAPTER 2

PROJECT DESCRIPTION

2.1 BLOCK DIAGRAM

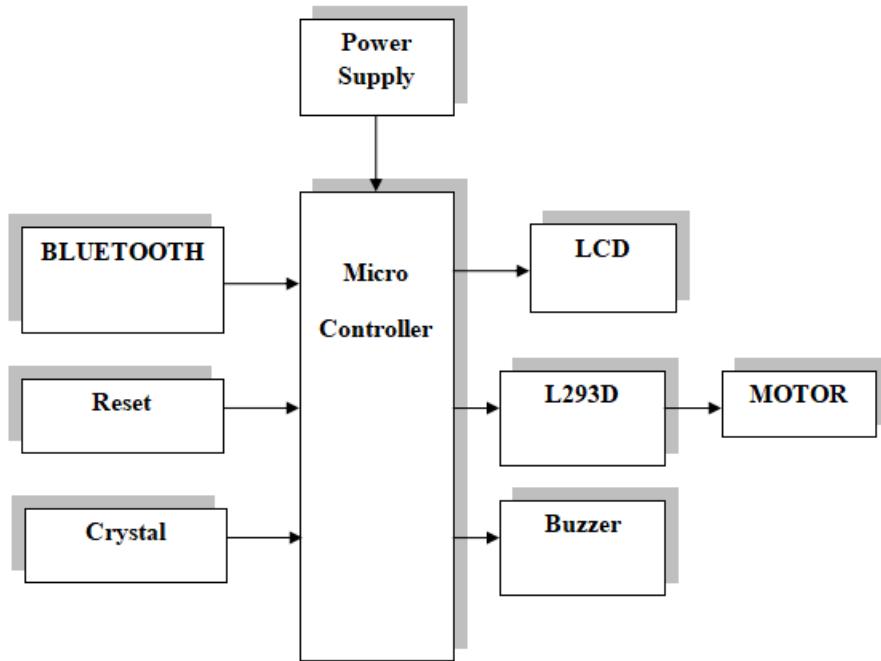


Fig. 2.1 Block Diagram

2.2 BLOCK DIAGRAM EXPLANATION

- Power supply is used to give the input power required to run the controller. It converts 230v AC supply to 5-12v DC Supply.
- Micro controller 8052 is used to take serial inputs and provide the required output.
- lcd is used to display the information to the reader in the form of message.
- A buzzer or beeper is a signaling device, usually electronic, typically used in automobiles, house hold appliances such as a microwave oven, or game shows.
- L293D IC is a typical Motor Driver IC which allows the DC motor to drive on any direction. This IC consists of 16-pins which are used to control a set of two DC motors instantaneously in any direction.
- The stepper motor converts a pulsing electrical current, controlled by a stepper motor driver, into precise one-step movements of this gear-like toothed component around a central shaft. Each of these stepper motor pulses moves the rotor through one precise and fixed increment of a full turn.
- HC-05 Bluetooth Module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. ... HC-05 Bluetooth module provides switching mode between master and slave mode which means it able to use neither receiving nor transmitting data

CHAPTER 3

HARDWARE COMPONENTS

3.1 POWER SUPPLY

The power supply section is the section which provide +5V for the components to work. IC LM7805 is used for providing a constant power of +5V.

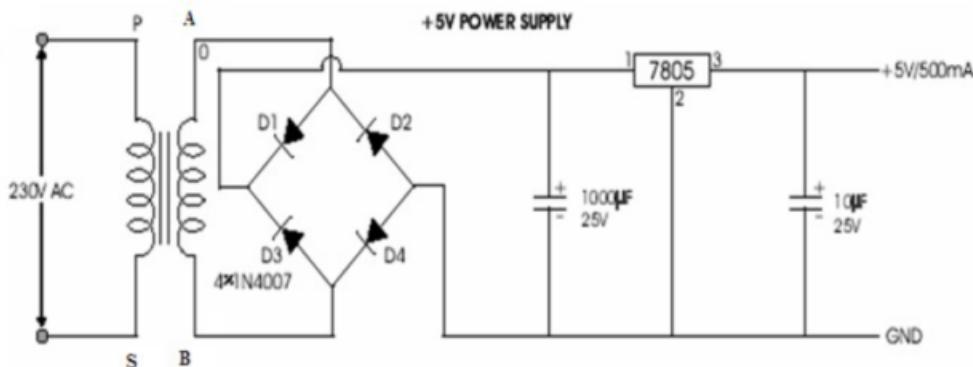


Fig.3.1: Internal circuit diagram of power supply

The ac voltage, typically 220V, is connected to a transformer, which steps down that ac voltage down to the level of the desired dc output. A diode rectifier then provides a full-wave rectified voltage that is initially filtered by a simple capacitor filter to produce a dc voltage. This resulting dc voltage usually has some ripple or ac voltage variation.

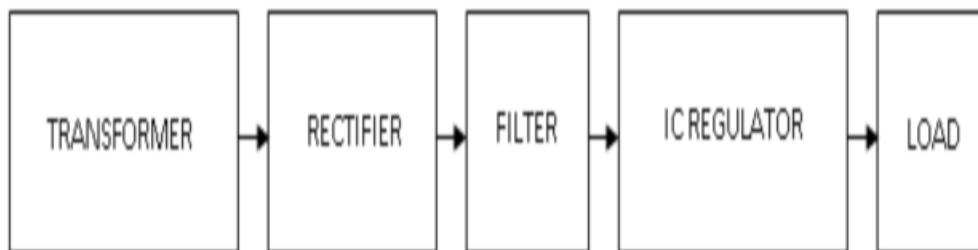


Fig 3.2 Block Diagram of Power Supply

3.1.1 Transformer

Transformers convert AC electricity from one voltage to another with little loss of power. Step-up transformers increase voltage, step-down transformers reduce voltage. Most power supplies use a step-down transformer to reduce the dangerously high mains voltage (230V in India) to a safer low voltage.

The input coil is called the primary and the output coil is called the secondary. There is no electrical connection between the two coils; instead they are linked by an alternating magnetic field created in the soft-iron core of the transformer. Transformers waste very little power so the power out is (almost) equal to the power in. Note that as voltage is stepped down current is stepped up. The transformer will step down the power supply voltage (0-230V) to (0- 6V) level.

3.1.2 Rectifier

There are several ways of connecting diodes to make a rectifier to convert AC to DC. The bridge rectifier is the most important and it produces full-wave varying DC. A full-wave rectifier can also be made from just two diodes if a center-tap transformer is used, but this method is rarely used now that diodes are cheaper. A single diode can be used as a rectifier but it only uses the positive (+) parts of the AC wave to produce half-wave varying DC.

Bridge Rectifier

When four diodes are connected as shown in figure, the circuit is called as bridge rectifier. The input to the circuit is applied to the diagonally opposite corners of the network, and the output is taken from the remaining two corners. Let us assume that the transformer is working properly and there is a positive potential, at point A and a negative potential at point B. the positive potential at point A will forward bias D3 and reverse bias D4.

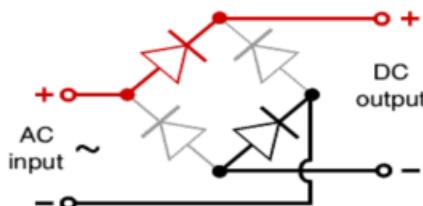


Fig. 3.3 Bridge Rectifier

The negative potential at point B will forward bias D1 and reverse D2. At this time D3 and D1 are forward biased and will allow current flow to pass through them; D4 and D2 are reverse biased and will block current flow.

One advantage of a bridge rectifier over a conventional full-wave rectifier is that with a given transformer the bridge rectifier produces a voltage output that is nearly twice that of the conventional full-wave circuit.

3.1.3 Voltage Regulators

Voltage regulators comprise a class of widely used ICs. Regulator IC units contain the circuitry for reference source, comparator amplifier, control device, and overload protection all in a single IC. IC units provide regulation of either a fixed positive voltage, a fixed negative voltage, or an adjustable set voltage. The regulators can be selected for operation with load currents from hundreds of milli amperes to tens of amperes, corresponding to power ratings from milli watts to Tens of watts.

A fixed three-terminal voltage regulator has an unregulated dc input voltage, V_i , applied to one input terminal, a regulated dc output voltage, V_o , from a second terminal, with the third terminal connected to ground.

The series 78 regulators provide fixed positive regulated voltages from 5 to 24 volts. Similarly, the series 79 regulators provide fixed negative regulated voltages from 5 to 24 volts. Voltage regulator ICs are available with fixed (typically 5, 12 and 15V) or variable output voltages. They are also rated by the maximum current they can pass. Negative voltage regulators are available, mainly for use in dual supplies. Most regulators include some automatic protection from excessive current ('overload protection') and overheating ('thermal protection').

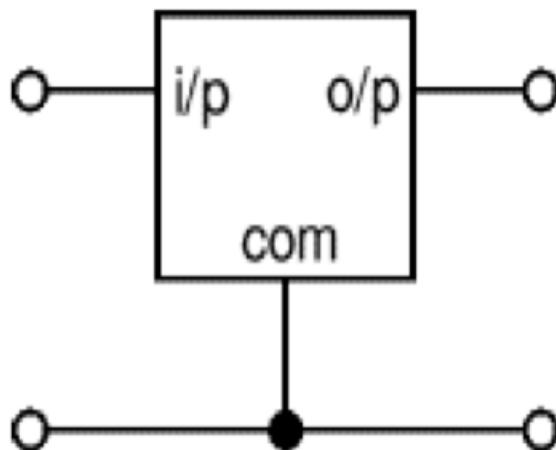


Fig.3.4 Regulator

3.2 MICRO-CONTROLLER

A Micro-controller (or MCU) is a computer-on-a-chip used to control electronic devices. It is a type of microprocessor emphasizing self-sufficiency and cost-effectiveness, in contrast to a general-purpose microprocessor (the kind used in a PC). A typical micro-controller contains all the memory and interfaces needed for a simple application, whereas a general purpose microprocessor requires additional chips to provide these functions.

A micro-controller is a single integrated circuit with the following key features:

- central processing unit - ranging from small and simple 8-bit processors to sophisticated 32- or 64-bit processors
- input/output interfaces such as serial ports
- RAM for data storage
- ROM, EEPROM or Flash memory for program storage
- clock generator - often an oscillator for a quartz timing crystal, resonator or RC circuit

3.2.1 Description

The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 8K bytes of in-system programmable Flash memory. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pin-out. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the Atmel AT89S52 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications.

The AT89S52 provides the following standard features: 8K bytes of Flash, 256 bytes of RAM, 32 I/O lines, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset.

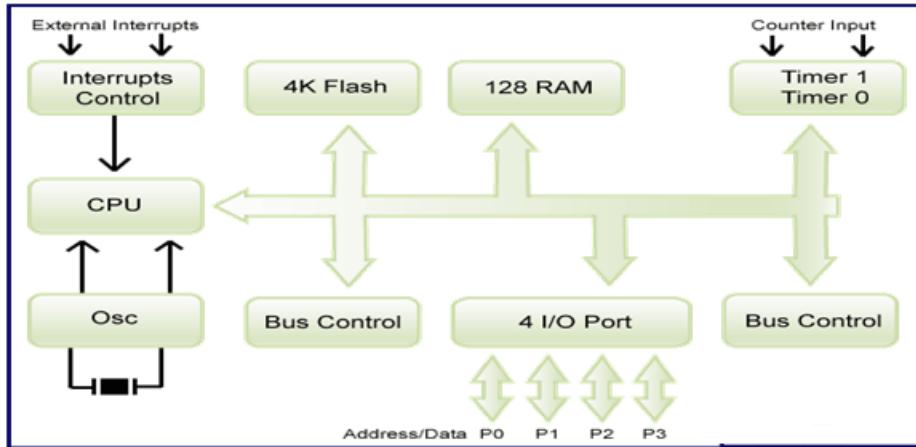


Fig 3.5 BLOCK DIAGRAM OF 8052

3.2.2 PIN DIAGRAM

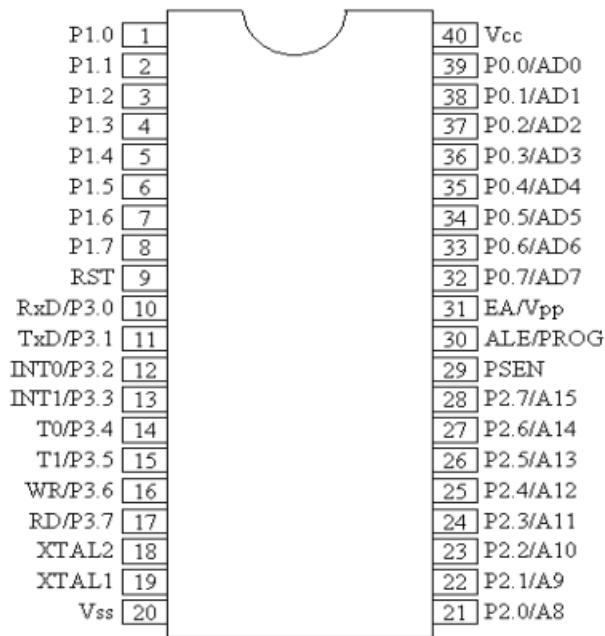


Fig.3.6 PIN DIAGRAM OF 8052

3.2.3 PIN DESCRIPTION

40 pin VCC Supply voltage.

20 pin GND Ground.

Port 0

Port 0 is an 8-bit open drain bidirectional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs. Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pull-ups are required during program verification.

Port 1

Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (IIL) because of the internal pull-ups.

Port 2

Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current because of the internal pull-ups. In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses, Port 2 emits the contents of the P2 Special Function Register. Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3

Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current because of the pull-ups. Port 3 also serves the functions of various special features of the AT89S52, as shown in the following table.

Table 1-Port 3 Pin Description of Micro-Controller

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	INT0 (external interrupt 0)
P3.3	INT1 (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	WR (external data memory write strobe)
P3.7	RD (external data memory read strobe)

RST

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

WR

It is active low write O/P control signal. During External RAM (Data memory). Write to external RAM.

RD

It is active low read O/P control signal. During External RAM (Data memory). Read from External RAM.

XTAL1 Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2 Output from the inverting oscillator amplifier.

ALE/PROG

Address Latch Enable (ALE) is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (PROG) during Flash programming. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction.

PSEN

Program Store Enable (PSEN) is the read strobe to external program memory. When the AT89S52 is executing code from external program memory, PSEN is activated twice each machine cycle.

EA

External Access Enable. EA must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, EA will be internally latched on reset. EA should be strapped to VCC for internal program executions.

3.2.4 REGISTERS

Registers and accumulators serve as temporary memory locations during CPU operations. The exchange of information among them takes place through one or more internal buses. The length of each register is equal to the width of the internal data bus.

Registers are categorized as general-purpose registers and special-purpose registers.

A general-purpose register may be used as an accumulator or as a data register for (arithmetic) and logic operations. An accumulator is a register for storing the results of an arithmetic operation. A general-purpose register may also be used as an address register.

3.2.5 Memory Organization

MCU-51 devices have a separate address space for Program and Data Memory. Up to 64K bytes each of external Program and Data Memory can be addressed.

The instructions to be executed by the microcontroller CPU and the data to be operated on by these instructions are stored in memory. When the CPU accesses the information stored in memory, it is performing a read operation. When the CPU sends information to be stored in the memory, it is performing a write operation. Memory is classified as internal and external memory. Internal memory is on-chip memory and is a semiconductor type with low capacity and high speeds. External memory is outside the chip and includes the semiconductor type and serial memory such as magnetic disks, magnetic tapes, and bubble memory. Semiconductor memory may be volatile or nonvolatile. Volatile memory loses its contents after the power is removed from the memory chip. Nonvolatile memory does not lose its contents when power is removed. The nonvolatile memory can store information permanently or at least semi-permanently (ten years or more).

Data Memory

The AT89S52 implements 256 bytes of on-chip RAM. The upper 128 bytes occupy a parallel address space to the Special Function Registers. This means that the upper 128 bytes have the same addresses as the SFR space but are physically separate from SFR space. When an instruction accesses an internal location above address 7FH, the address mode used in the instruction specifies whether the CPU accesses the upper 128 bytes of RAM or the SFR space. Instructions which use direct addressing access the SFR space.

UART (Universal Asynchronous Receiver and Transmitter)

One of the micro-controller features making it so powerful is an integrated UART, better known as a serial port. It is a full-duplex port, thus being able to transmit and receive data simultaneously and at different baud rates. Without it, serial data send and receive would be an enormously complicated part of the program in which the pin state is constantly changed and checked at regular intervals. When using UART, all the programmer has to do is to simply select serial port mode and baud rate. When it's done, serial data transmit is nothing but writing to the SBUF register, while data receive represents reading the same register. The micro-controller takes care of not making any error during data transmission.

TIMERS

Timer is used to generate time delay, a timer always counts up. It doesn't matter whether the timer is being used as a timer, a counter, or a baud rate generator: A timer is always incremented by the microcontroller. There are 3 timers i.e., Timer 0, Timer 1 and Timer 2.

Timer 0 and 1

Timer 0 and Timer 1 in the AT89S52 operate the same way as Timer 0 and Timer 1 in the 8051.

Timer T0 is a 16 bit timer. The 16-bit Timer 0 is accessed as low byte and high byte. the low byte is called TL0(Timer 0 low byte)and the high byte is referred to as TH0(Timer 0 high byte).

As like Timer 1 is also 16-bit timer is split into two bytes, referred to as TL1 (Timer 1 low byte) and TH1(Timer 1 high byte). These timer are accessible in the same way as the Timer 0.

Timer 2

Timer 2 is a 16-bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit C/T2 in the SFR T2CON (shown in Table 5-2). Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. Timer 2 consists of two 8-bit registers, TH2 and TL2. In the Timer function, the TL2 register is incremented every machine cycle. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

INTERRUPTS

The AT89S52 has a total of six interrupt vectors: two external interrupts (INT0 and INT1), three timer interrupts (Timers 0, 1, and 2), and the serial port interrupt.

The Timer 0 and Timer 1 flags, TF0 and TF1, uses the cycle in which the timers overflow. The values are then polled by the circuitry in the next cycle. However, the Timer 2 flag, TF2, is set at S2P2 and is polled in the same cycle in which the timer overflows.

3.2.6 Working of Micro Controller

Micro-controllers are embedded inside devices to control the actions and features of a product. Hence, they can also be referred to as embedded controllers. They run one specific program and are dedicated to a single task. They are low power devices with dedicated input devices and small LED or LCD display outputs.

3.3 LIQUID CRYSTAL DISPLAY

3.3.1 Introduction

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other segment LEDs. The reasons being: LCD's are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animations and so on.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD.

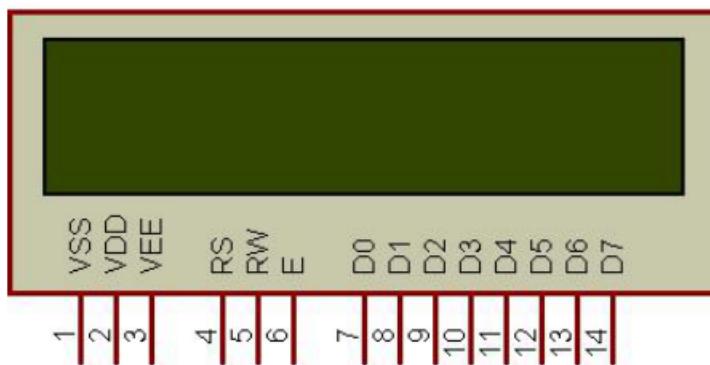


Fig.3.7 16x2 LCD

3.3.2 WORKING PRINCIPLE

A liquid-crystal display (LCD) is a flat-panel display or other electronically modulated optical device that uses the light-modulating properties of liquid crystals combined with polarizers. Liquid crystals do not emit light directly,[1] instead using a backlight or reflector to produce images in color or monochrome.[2] LCDs are available to display arbitrary images (as in a general-purpose computer display) or fixed images with low information content, which can be displayed or hidden, such as preset words, digits, and seven-segment displays, as in a digital clock.

LCD WORKING

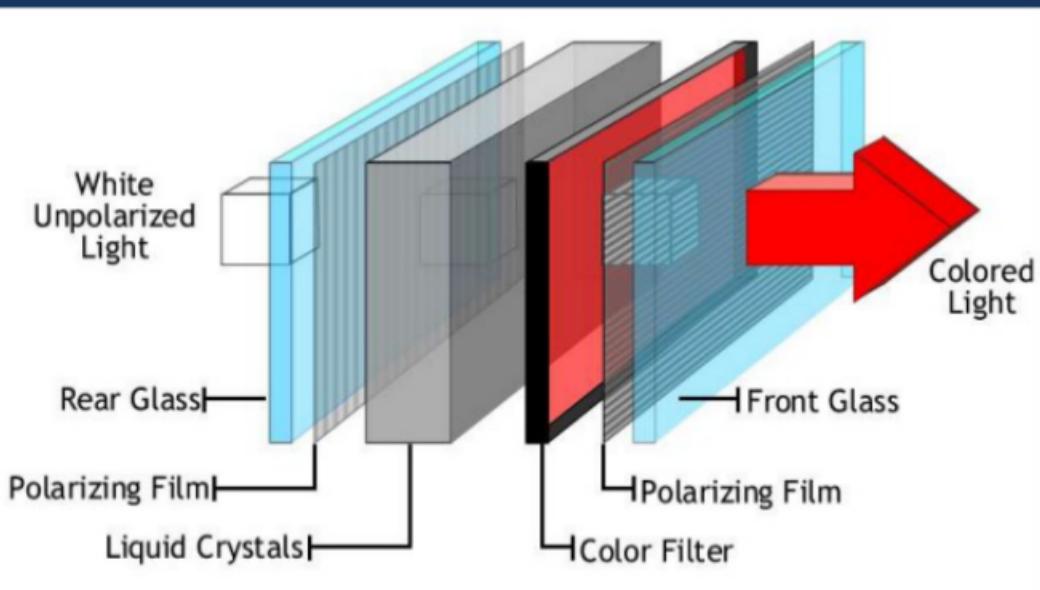


Fig.3.8 LCD Working

3.3.3 Pin Description

Most LCD's with 1 controller has 14 Pins and LCD's with 2 controller has 16 Pins (two pins are extra in both for back-light LED connections). Pin description is shown in the table below.

Pin Configuration table for a 16X2 LCD character display:

Table 2- Pin Configuration of LCD

Pin Number	Symbol	Function
1	Vss	Ground Terminal
2	Vcc	Positive Supply
3	Vdd	Contrast adjustment
4	RS	Register Select; 0→Instruction Register, 1→Data Register
5	R/W	Read/write Signal; 1→Read, 0→ Write
6	E	Enable; Falling edge
7	DB0	Bi-directional data bus, data transfer is performed once, thru DB0 to DB7, in the case of interface data length is 8-bits; and twice, through DB4 to DB7 in the case of interface data length is 4-bits. Upper four bits first then lower four bits.
8	DB1	
9	DB2	
10	DB3	
11	DB4	
12	DB5	
13	DB6	
14	DB7	
15	LED-(K)	Back light LED cathode terminal
16	LED+(A)	Back Light LED anode terminal

3.3.4 INTERFACING CIRCUIT

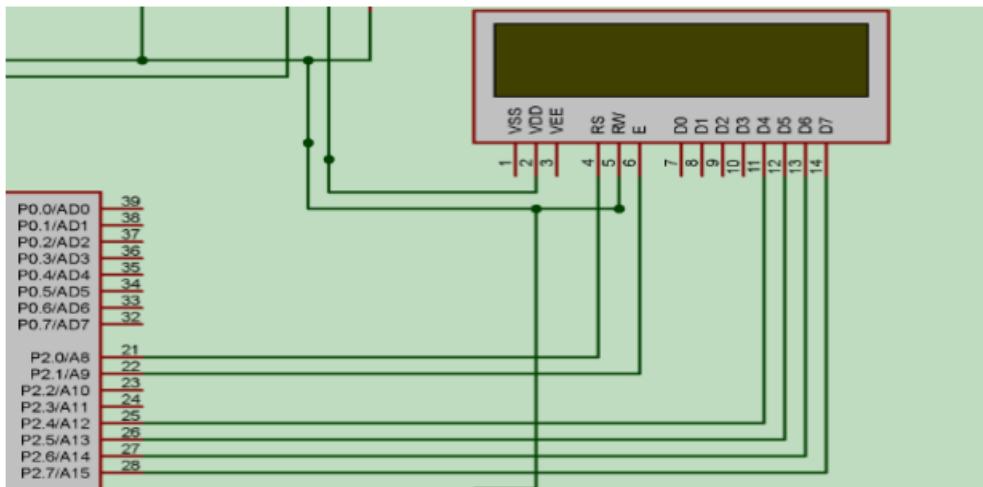


Fig.3.9 Interface Diagram of Lcd with 8052 Micro-Controller

Liquid crystal displays interfacing with Controller

The LCD standard requires 3 control lines and 8 I/O lines for the data bus.

- **8 data pins D7:D0**

Bi-directional data/command pins.

Alphanumeric characters are sent in ASCII format.

- **RS: Register Select**

RS = 0 → Command Register is selected

RS = 1 → Data Register is selected

- **R/W: Read or Write**

0 → Write, 1 → Read

- **E: Enable (Latch data)**

Used to latch the data present on the data pins.

A high-to-low edge is needed to latch the data.

3.4 BLUETOOTH

Bluetooth is a wireless technology standard for exchanging data over short distances (using short-wavelength UHF radio waves in the ISM band from 2.4 to 2.485 GHz) from fixed and mobile devices and building personal area networks (PANs).

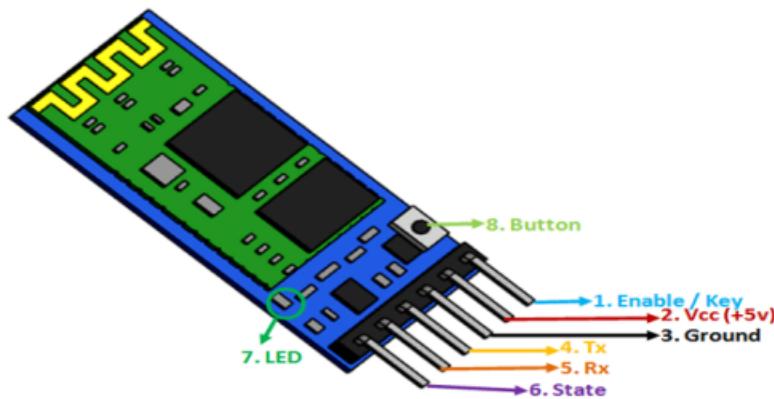


Fig.3.10 BLUETOOTH

Bluetooth networking transmits data via low-power radio waves. It communicates on a frequency of 2.45 gigahertz (actually between 2.402 GHz and 2.480 GHz, to be exact). This frequency band has been set aside by international agreement for the use of industrial, scientific and medical devices (ISM). By comparison, the most powerful cell phones can transmit a signal of 3 watts.

Some specification of Bluetooth

- Operates in the 2.4 GHZ band which is globally available
- It has 79 channels
- 1600 hops per second
- Can support up to 8 devices in a piconet
- Omni-directional, non-line of sight transmission through walls
- 10m to 100m range
- Low cost, ₹200
- 1mW power

3.4.1 Working of Bluetooth

Bluetooth works by the simple principle of sending and receiving data in the form of radio waves. Every Bluetooth enabled device has a card-like attachment known as the Bluetooth adapter. It is this Bluetooth adapter that sends and receives data. A Bluetooth adapter has a particular range of connection. One electronic adaptor can notice another Bluetooth device only if the second device is present within the range of the first device. When they are within the range, they can strike up a connection between themselves. Striking up of connection between two Bluetooth devices are known as paring of devices.

3.4.2 INTERFACING CIRCUIT

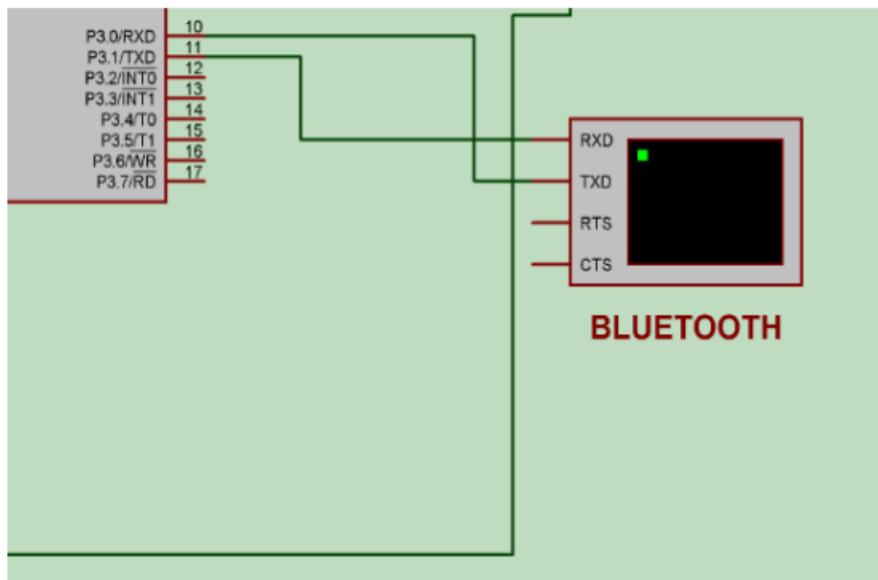


Fig.3.11 Interface Diagram of Bluetooth with 8052 Micro-Controller

3.5 DRIVER CIRCUIT (L293D)

L293D IC generally comes as a standard 16-pin DIP (dual-in line package). This motor driver IC can simultaneously control two small motors in either direction; forward and reverse with just 4 micro-controller pins.

3.5.1 Description

It works on the concept of H-bridge. H-bridge is a circuit which allows the voltage to be flown in either direction

There are two Enable pins on L293D. Pin 1 and pin 9, for being able to drive the motor, the pin 1 and 9 need to be high. For driving the motor with left H-bridge you need to enable pin 1 to high. And for right H-Bridge you need to make the pin 9 to high. If anyone of the either pin 1 or pin 9 goes low then the motor in the corresponding section will suspend working. It's like a switch.

3.5.2 Pin Diagram

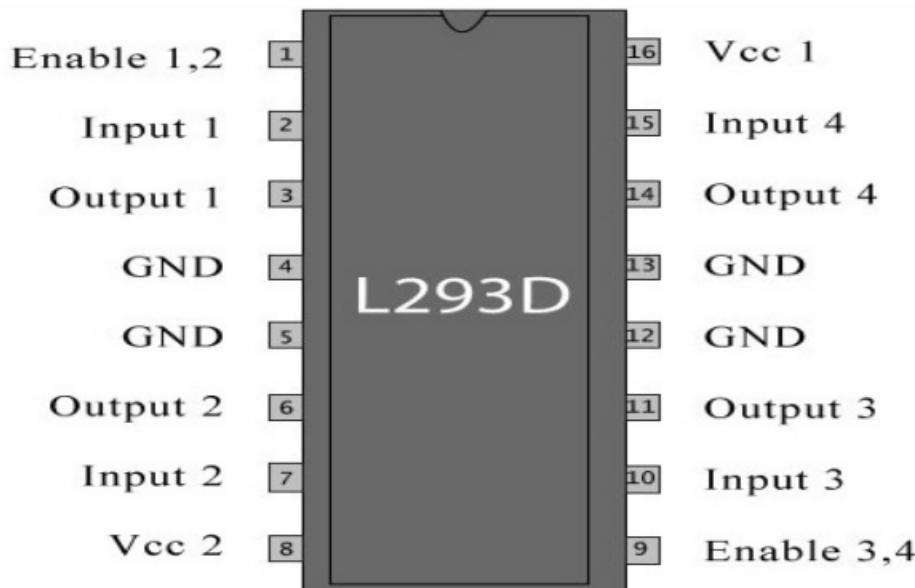


Fig.3.12 Pin Diagram of L293D

3.5.3 Pin Description

Table- 3: Pin Description of L293D

Pin No	Function	Name
1	Enable pin for Motor 1; active high	Enable 1,2
2	Input 1 for Motor 1	Input 1
3	Output 1 for Motor 1	Output 1
4	Ground (0V)	Ground
5	Ground (0V)	Ground
6	Output 2 for Motor 1	Output 2
7	Input 2 for Motor 1	Input 2
8	Supply voltage for Motors; 9-12V (up to 36V)	Vcc 2
9	Enable pin for Motor 2; active high	Enable 3,4
10	Input 1 for Motor 1	Input 3
11	Output 1 for Motor 1	Output 3
12	Ground (0V)	Ground
13	Ground (0V)	Ground
14	Output 2 for Motor 1	Output 4
15	Input2 for Motor 1	Input 4
16	Supply voltage; 5V (up to 36V)	Vcc 1

3.5.4 Working of L293D

The 4 input pins for this L293d, pin 2,7 on the left and pin 15 ,10 on the right as shown on the pin diagram. Left input pins will regulate the rotation of motor connected across left side and right input for motor on the right hand side. The motors are rotated on the basis of the inputs provided across the input pins as LOGIC 0 or LOGIC 1. In simple you need to provide Logic 0 or 1 across the input pins for rotating the motor.

3.5.5 INTERFACING CIRCUIT

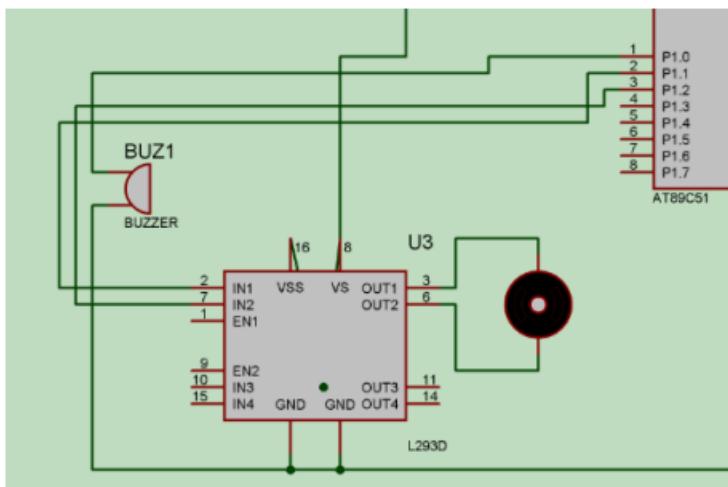


Fig.3.13 Interface Diagram of Driver Circuit with 8052 Micro-Controller

3.6 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. DC motor is designed to run on DC electric power. Two examples of pure DC designs are Michael Faraday's homo-polar motor (which is uncommon), and the ball bearing motor, which is (so far) a novelty.

We in our project are using brushed DC Motor, which will operate in the ratings of 12v DC 0.6Amp. The speed of a DC motor can be controlled by changing the voltage applied to the armature or by changing the field current. The introduction of variable resistance in the armature circuit or field circuit allowed speed control. Modern DC motors are often controlled by power electronics systems called DC drives.

3.6.1 Working of DC Motor

The stepper motor converts a pulsing electrical current, controlled by a stepper motor driver, into precise one-step movements of this gear-like toothed component around a central shaft. Each of these stepper motor pulses moves the rotor through one precise and fixed increment of a full turn.



Fig.3.14 DC Motor

3.7 BUZZER

A buzzer or beeper is a signaling device, usually electronic, typically used in automobiles, house hold appliances such as a microwave oven, or game shows.

It most commonly consists of a number of switches or sensors connected to a control unit that determines if and which button was pushed or a preset time has lapsed, and usually illuminates a light on the appropriate button or control panel, and sounds a warning in the form of a continuous or intermittent buzzing or beeping sound. Initially this device was based on an electromechanical system which was identical to an electric bell without the metal gong (which makes the ringing noise). Often these units were anchored to a wall or ceiling and used the ceiling or wall as a sounding board. Another implementation with some AC-connected devices was to implement a circuit to make the AC current into a noise loud enough to drive a loudspeaker and hook this circuit up to a cheap 8-ohm speaker. Nowadays, it is more popular to use a ceramic-based piezoelectric sounder like a

Sonalert which makes a high-pitched tone. Usually these were hooked up to "driver" circuits which varied the pitch of the sound or pulsed the sound on and off.

In game shows it is also known as a "lockout system," because when one person signals ("buzzes in"), all others are locked out from signalling. Several game shows have large buzzer buttons which are identified as "plungers".

3.7.1 Working of Buzzer

The buzzer consists of an outside case with two pins to attach it to power and ground. When current is applied to the buzzer it causes the ceramic disk to contract or expand. This then causes the surrounding disc to vibrate. That's the sound that you hear.



Fig.3.15 Buzzer

CHAPTER 4

SOFTWARE SPECIFICATION

4.1 KEIL SOFTWARE

Keil compiler is a software used where the machine language code is written and compiled. After compilation, the machine source code is converted into hex code which is to be dumped into the microcontroller for further processing. Keil compiler also supports C language code.

General Introduction

Keil Software is the leading vendor for 8/16-bit development tools (ranked at first position in the 2004 Embedded Market Study of the Embedded Systems and EE Times magazine). Keil Software is represented world-wide in more than 40 countries. Since the market introduction in 1988, the Keil C51 Compiler is the de facto industry standard and supports more than 500 current 8051 device variants. Now, Keil Software offers development tools for ARM.

Keil Software makes C compilers, macro assemblers, real-time kernels, debuggers, simulators, integrated environments, and evaluation boards for the 8051, 251, ARM, and XC16x/C16x/ST10 microcontroller families.

Keil Software is pleased to announce simulation support for the Atmel AT91 ARM family of microcontrollers. The Keil µVision Debugger simulates the complete ARM instruction-set as well as the on-chip peripherals for each device in the AT91 ARM/Thumb microcontroller family. The integrated simulator provides complete peripheral simulation. Other new features in the µVision Debugger include:

- An integrated Software Logic Analyzer that measures I/O signals as well as program variables and helps developers create complex signal processing algorithms.
- An Execution Profiler that measures time spent in each function, source line, and assembler instruction. Now developers can find exactly where programs spend the most time.

"Using nothing more than the provided simulation support and debug scripts, developers can create a high-fidelity simulation of their actual target hardware and environment. No extra hardware or test equipment is required. The Logic Analyzer and Execution Profiler will help developers when it comes time to develop and tune signaling algorithms." said Jon Ward, President of Keil Software USA, Inc.

μVision3 Overview

The μVision3 IDE is a Windows-based software development platform that combines a robust editor, project manager, and makes facility. μVision3 integrates all tools including the C compiler, macro assembler, linker/locator, and HEX file generator. μVision3 helps expedite the development process of your embedded applications by providing the following:

- Full-featured source code editor,
- Device database for configuring the development tool setting,
- Project manager for creating and maintaining your projects,
- Integrated make facility for assembling, compiling, and linking your embedded applications,
- Dialogs for all development tool settings,
- True integrated source-level Debugger with high-speed CPU and peripheral simulator,
- Advanced GDI interface for software debugging in the target hardware and for connection to Keil ULINK,
- Flash programming utility for downloading the application program into Flash ROM,
- Links to development tools manuals, device datasheets & user's guides.

The μVision3 IDE offers numerous features and advantages that help you quickly and successfully develop embedded applications. They are easy to use and are guaranteed to help you achieve your design goals.

The μVision3 IDE and Debugger is the central part of the Keil development tool chain. μVision3 offers a Build Mode and a Debug Mode.

In the μVision3 Build Mode you maintain the project files and generate the application.

In the μVision3 Debug Mode you verify your program either with a powerful CPU and peripheral simulator or with the Keil ULINK USB-JTAG Adapter (or other AGDI drivers) that connect the debugger to the target system. The ULINK allows you also to download your application into Flash ROM of your target system.

4.2 STEPS TO WRITE AN ASSEMBLY LANGUAGE PROGRAM IN KEIL AND HOW TO COMPILE IT:

1. Install the Keil Software in the PC in any of the drives.
2. After installation, an icon will be created with the name “Keil uVision3”. Just drag this icon onto the desktop so that it becomes easy whenever you try to write programs in keil.
3. Double click on this icon to start the keil compiler.
4. A page opens with different options in it showing the project workspace at the leftmost corner side, output window in the bottom and an ash coloured space for the program to be written.
5. Now to start using the keil, click on the option “project”.
6. A small window opens showing the options like new project, import project, open project etc. Click on “New project”.
7. A small window with the title bar “Create new project” opens. The window asks the user to give the project name with which it should be created and the destination location. The project can be created in any of the drives available. You can create a new folder and then a new file or can create directly a new file.
8. After the file is saved in the given destination location, a window opens where a list of vendors will be displayed and you have to select the device for the target you have created.
9. The most widely used vendor is Atmel. So click on Atmel and now the family of microcontrollers manufactured by Atmel opens. You can select any one of the microcontrollers according to the requirement.
10. When you click on any one of the microcontrollers, the features of that particular microcontroller will be displayed on the right side of the page. The most appropriate microcontroller with which most of the projects can be implemented is the AT89C51. Click on this microcontroller and have a look at its features. Now click on “OK” to select this microcontroller.
11. A small window opens asking whether to copy the startup code into the file you have created just now. Just click on “No” to proceed further.
12. Now you can see the TARGET and SOURCE GROUP created in the project workspace.
13. Now click on “File” and in that “New”. A new page opens and you can start writing program in it.
14. After the program is completed, save it with any name but with the .asm extension. Save the program in the file you have created earlier.
15. You can notice that after you save the program, the predefined keywords will be highlighted in bold letters.

16. Now add this file to the target by giving a right click on the source group. A list of options open and in that select “Add files to the source group”. Check for this file where you have saved and add it.
17. Right click on the target and select the first option “Options for target”. A window opens with different options like device, target, output etc. First click on “target”.
18. Since the set frequency of the microcontroller is 11.0592 MHz to interface with the PC, just enter this frequency value in the Xtal (MHz) text area and put a tick on the Use on-chip ROM. This is because the program what we write here in the keil will later be dumped into the microcontroller and will be stored in the inbuilt ROM in the microcontroller.
19. Now click the option “Output” and give any name to the hex file to be created in the “Name of executable” text area and put a tick to the “Create HEX file” option present in the same window. The hex file can be created in any of the drives. You can change the folder by clicking on “Select folder for Objects”.
20. Now to check whether the program you have written is errorless or not, click on the icon exactly below the “Open file” icon which is nothing but Build Target icon. You can even use the shortcut key F7 to compile the program written.
21. To check for the output, there are several windows like serial window, memory window, project window etc. Depending on the program you have written, select the appropriate window to see the output by entering into debug mode.
22. The icon with the letter “d” indicates the debug mode.
23. Click on this icon and now click on the option “View” and select the appropriate window to check for the output.
24. After this is done, click the icon “debug” again to come out of the debug mode.
25. The hex file created as shown earlier will be dumped into the microcontroller with the help of another software called Proload/Topwin.

CHAPTER 5

IMPLEMENTATION

5.1 SCHEMATIC DIAGRAM

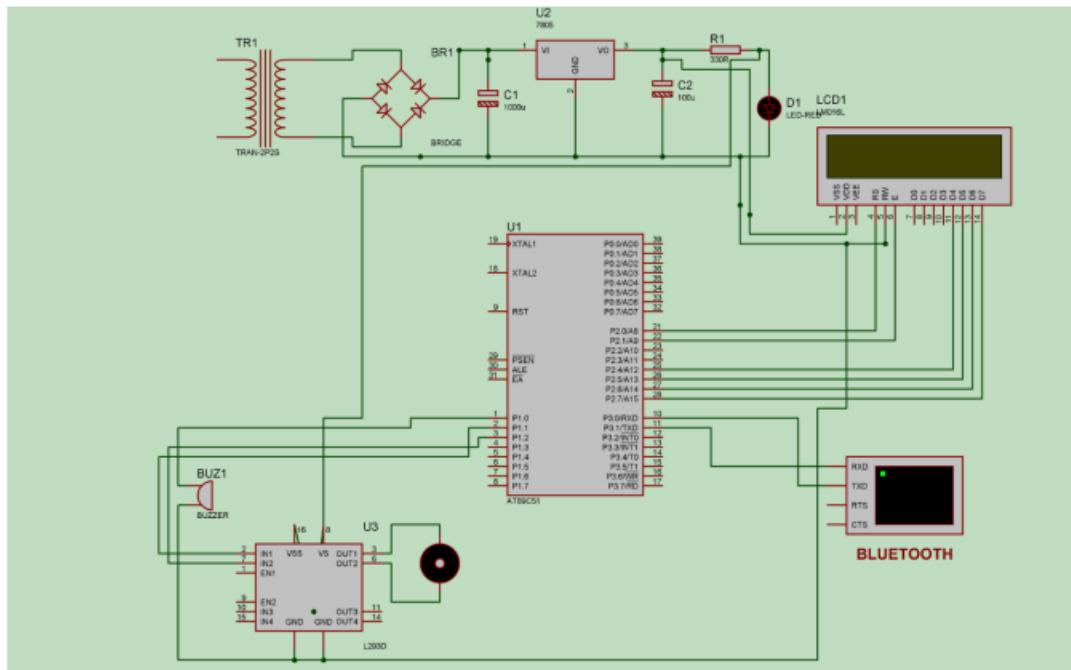


Fig.5.1 SCHEMATIC DIAGRAM

In this circuit the microcontroller is used to control the DC Motor according to the our input through android mobile. The voltage from the mains (220/240V AC) is stepped down by a transformer to 12V. Then the 12V DC passes through the voltage regulator to give a clean 5V DC. According to the readings from the ASCII commands, Motor is controlled. Crystal oscillator is connected in between pin 18 and pin 19 of Micro controller, those are pins if we want to provide external clock to the microcontroller. 0.1 μ F bypass capacitor used on the output pin +5 V of the voltage regulator to smooth out the supply voltage to microcontroller and LCD.

5.2 FLOW CHART

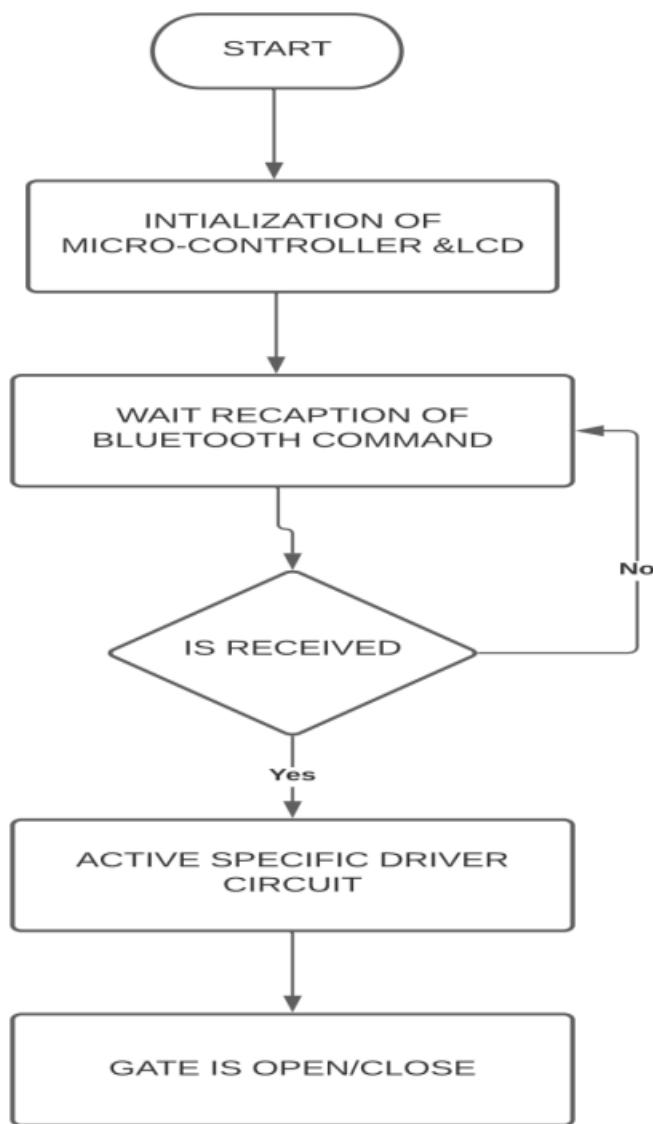


Fig.5.2 Flow Chart

5.3 Flow Chart Explanation

- The above flow chart shows the complete flow of the System.
- When application starts LCD displays “Railway Crossing” user can pair Bluetooth through Bluetooth terminal Hc-05 Android Application.
- Send ASCII command if it matches to predefined value then it Activate Driver Circuit.
- Active specific Driver Circuit causes Dc Motor to Rotate.
- Rotation of Motor Leads to Gate Open/Close.
- If it doesn't matches ASCII then no process Happen.

CHAPTER 6

SIMULATION AND DESIGN

6.1 OUTPUT SCREEN-SHOTS

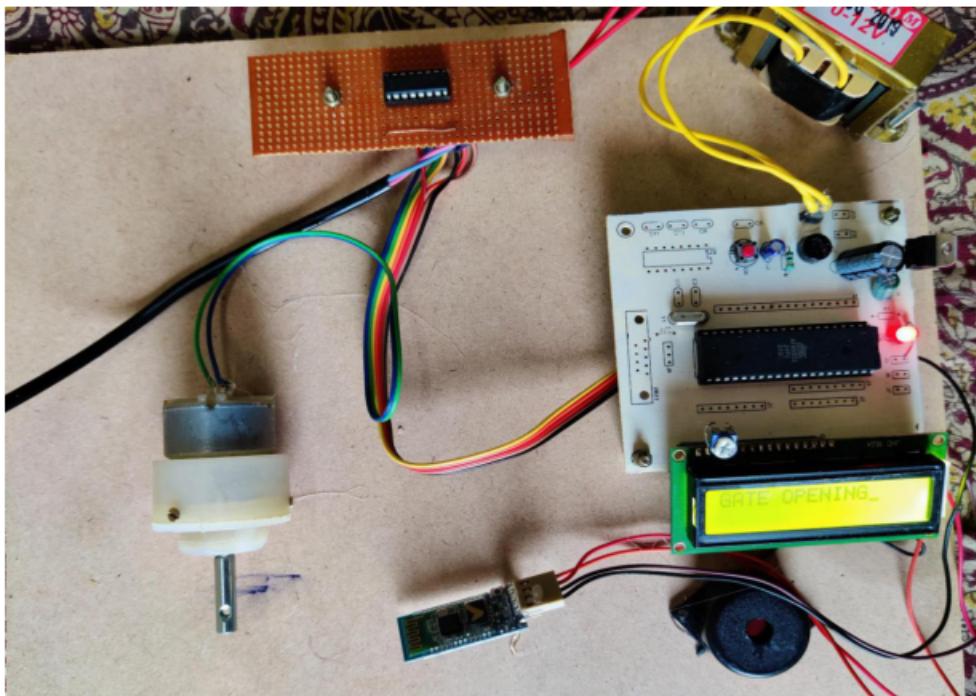


Fig.6.1 GATE OPENING

The above figure is One of the outputs of this kit is displaying message on LCD screen and the other one is in the form of sound which comes from buzzer. After the input from the operator is obtained all the internal processes are carried out and the output is sent to LCD and buzzer. A predefined value is assigned for Gate Closing. The Bluetooth detects the input ASCII command "O" and displays the message "GATE OPENING".

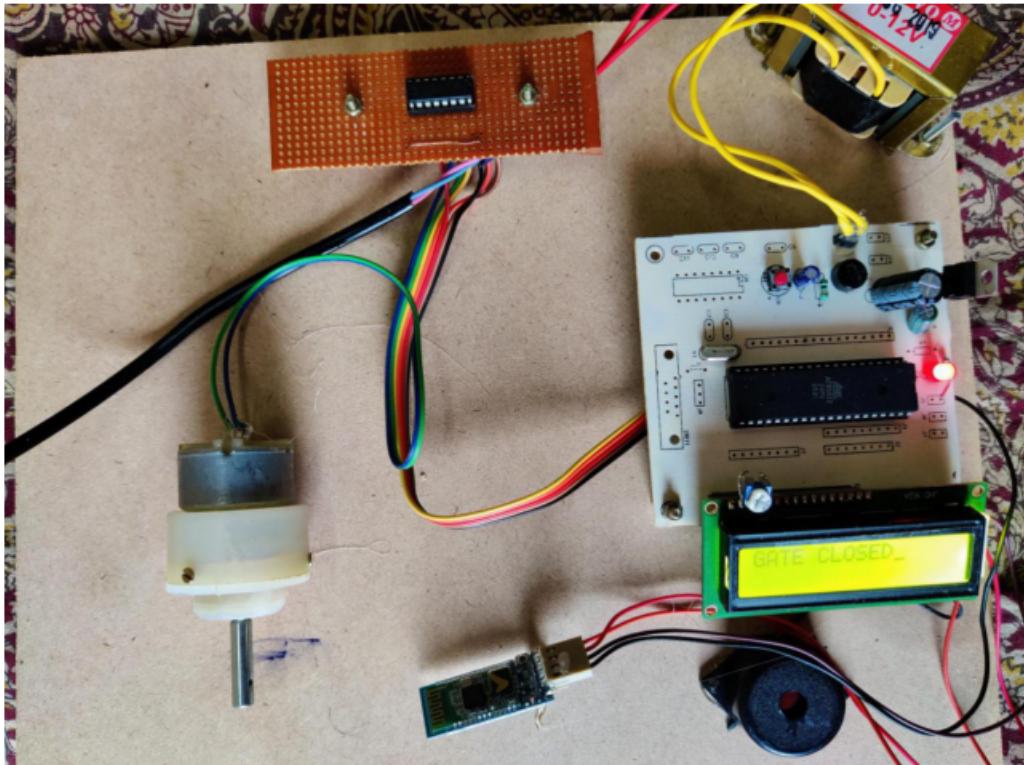


Fig.6.2 GATE CLOSED

One of the outputs of this kit is displaying message on LCD screen and the other one is in the form of sound which comes from buzzer. After the input from the operator is obtained all the internal processes are carried out and the output is sent to LCD and buzzer. A predefined value is assigned for Gate Closing. The Bluetooth detects the input ASCII command "C" and displays the message "GATE CLOSED".

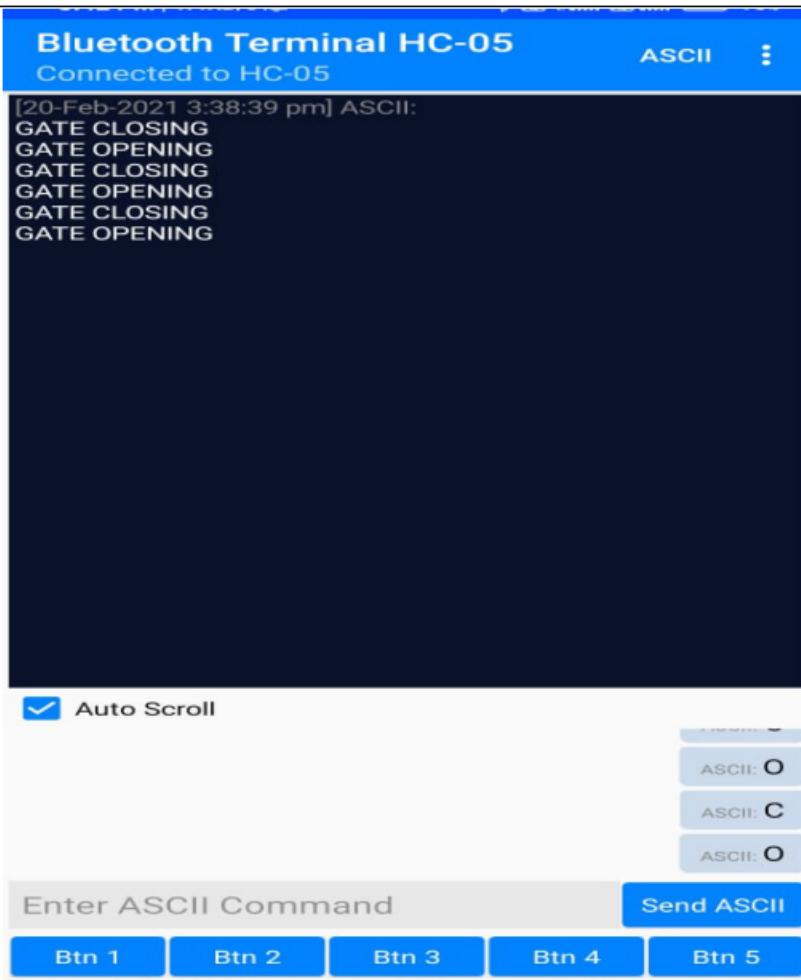


Fig.6.3 Bluetooth Terminal HC-05 App Screen Shot

This figure shows the message sent to user on mobile through Bluetooth Terminal HC-05 Android Application. when the ASCII command “C” is Send then gate is closing which displays message “GATE CLOSING”. If ASCII command “O” is Send then gate will opening which displays message “GATE OPENING”.

CHAPTER 7

CONCLUSION

From the above information of this system up to now, surely comes to know that it is highly reliable effective and economical at dense traffic area, sub urban area and the route where frequency of trains is more.

As it saves some auxiliary structure as well as the expenditure on attendant it is more economical than traditional railway crossing gate system.

We know that though it is very beneficial but it is also impossible to install such system at each and every place, but it gives certainly a considerable benefit to us, thereby to our nation.

7.1 FUTURE SCOPE

This proposed model suffers from the drawbacks that it cannot be used for very high Speed trains and at the hilly areas where the Bluetooth, a low-powered device designed with limits its data communications speed may not perform correctly, Henceforth the future work can concentrate on making it possible work on high speeds too.

Appendix-A

REFERENCES

- [1] en.wikipedia.org/wiki/Rail_inspection
- [2] en.wikipedia.org/wiki/Nondestructive_testing
- [3] en.wikipedia.org/wiki/Level_crossing
- [4] en.wikipedia.org/wiki/Laser_scanning
- [5] B. Brailson Mansingh, "Automation in unmanned railway level crossing", in Intelligent Systems and Control (ISCO), pg. 1 – 4, 2015
- [6] Onder Halis Bettemir, "Detection of railway track from image by heuristic method", in Signal Processing and Communications Applications Conference (SIU), pg. 1366 – 1369, 2015
- [7] Selvamraju Somalraju, Vigneshwar Murali, Gourav Saha, Dr.V.Vaidehi, "Robust Railway Crack Detection Scheme (RRCDS) Using LEDLDR Assembly," in Networking, Sensing and Control, vol. 6, iss. 3, pg. 453-460 , May2012
- [8] Qiao Jian-hua; Li Lin-sheng; Zhang Jing-gang; "Design of Rail Surface Crack- detecting System Based on Linear CCD Sensor," IEEE Int.Conf. on Networking, Sensing and Control, vol. 14, no. 4, pp. 961-970, April 2008
- [9] K. Vijayakumar, S.R. Wylie, J. D. Cullen, C.C. Wright, A.I. Shammaa, " Non invasive rail track detection system using Microwave sensor," Journal of App. Phy., vol. 9, iss. 11, pg. 1743-1749, June 2009
- [10] Richard J. Greene, John R. Yates and Eann A. Patterson, "Crack detection in rail using infrared methods," Opt. Eng. 46, 051013, May 2007
- [11]. R.J. Greene, J.R. Yates, E.A. Patterson, "Rail Crack Detection: An Infrared Approach to In-service Track Monitoring," SEM Annual Conference & Exposition on Experimental and Applied Mechanics, vol. 112, nos. 23, pp. 291301, May 2006
- [12] Giovanni M. Carlomagno, Carosena Meola: HComparison between thermographic techniques for frescoes NDTH NDT & E International, Volume 35, Issue 8, December 2002, pg. 559-565.
- [13] Hartless, P.C.M.; Whalley, D.C. Automated thermographic inspection of surface mount solder joints. Automotive Electronics, 1991., Eighth International Conference on Volume , Issue , 28-31 Oct 1991 Pg.178 – 181.

BIOGRAPHY



NAME: CH K CHAITANYA
HT.NO:17831A0414
CONTACT NO:9948344091
E-MAIL:Chaitanya7c@gmail.com



NAME: K RAJA
HT.NO:17831A0435
CONTACT NO:9666800736
E-MAIL:maminiraja.0904@gmail.com



NAME: K SAITEJA
HT.NO:17831A0436
CONTACT NO:8977767167
E-MAIL:kanikesaiteja@gmail.com