BIOMETRIC BASED SECURITY SYSTEM

ACKNOWLEDGEMENT

This is the only subject in which we can express our deep sense of gratitude to all those who have given their valuable guidance, support and help during project work. We offer our profound gratitude to the management for giving us this opportunity of amalgamating our theoretical knowledge with practical experience, in a creative way.

The knowledge that we got from the project work filled us with huge confidence of always thinking in creative manner. We got to learn in depth about construction, working and specification of various components. We even got the knowledge about software programming and even became well equipped with handling many electronic components. This experience is unparallel to any other form of learning.

We would like to thank our project guide Mrs. Archana Wasule for her invaluable guidance, contribution and motivation during our project. We also want to thank our HOD Mrs. Raji M.P., all the teachers for their support and helping us in solving many of the difficulties. A special thanks to Mrs. Sonali Sherigar and Mr. Umesh Mhapankar for their all-time encouragement, support, and guidance.

We are also indebted to all the lab assistants: Mr. Dhiraj S., Mr. Vishwas Patil, Mr. Manoj Parthe, Mr. Avadhut Ghadge, and Mr. Gokhale. for all their help and being more than just a teacher to us. We thank all of them for devoting their time and energy. We also thank our college management for providing us with modern devices which helped us during testing and troubleshooting our system, they made our task very simple.

Last but not the least we thank our classmates whom we have not mentioned above, we haven’t forgotten you. Words aren’t enough to express the deep gratitude we have towards you all. Everyday with people like you has been an exciting journey.

PREFACE

"SCIENTIST MAKE INVENTION'S BUT ENGINEERS MANUFACTURE THEM". An Engineer is the link between science and society. So he must be an conversant with the skills of craftsman as well as in's and out's of any device, instruments or a process.

We have taken sincere attempts to put before reads some useful information regarding our project.

We have carried this project work as a part of our studies. Even while doing so we tried our best to collect, to explain and carrying the matter in such a way as to form an independent unit itself. We are such that information contained in this volume would be useful to have little insight in the scope and dimension of this project in its true perspective.

PROJECT PLAN

PROJECT TITLE:

BIOMETRIC BASED SECURITY SYSTEM

OBJECTIVES:

1. To implement the Fingerprint based security system using microcontroller 89S52.

2. To establish a link between science and society.

3. To conversant with the skills of craftsman as well as in's and out's of any device or process.

4. To collect the best of matter in such a way as to form an independent unit itself.

5. To provide reads some useful information regarding our project.

GROUP FORMATION: JULY 2014

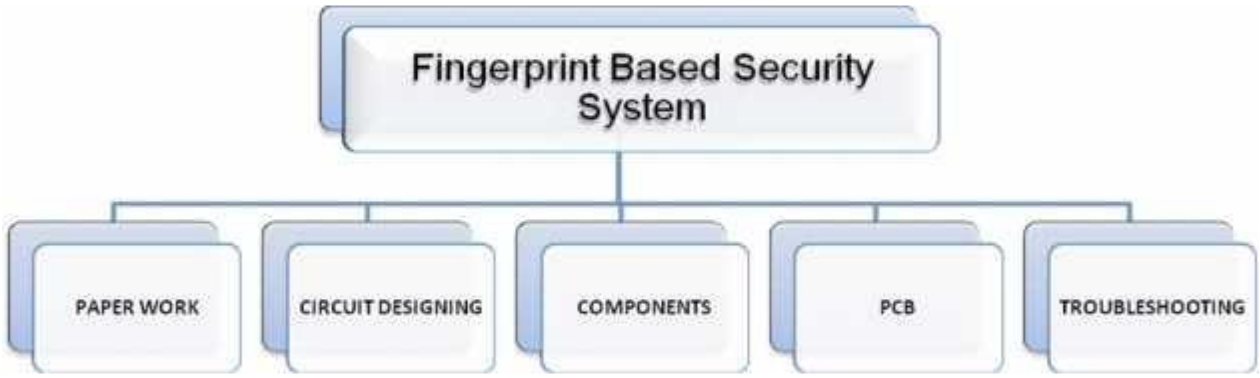
SELECTION OF PROJECT: JULY 2014

START DATE: 9th DECEMBER 2014

COMPLETEION DATE: 31ST MARCH 2015

COST: 11,500/-

WORK BREAKDOWN STRUCTURE (WBS):-



- Designing of tentative circuit.
- Selecting of components
- Designing of program to display User ID on LCD.
- Writing a sample program to check the functioning of circuit.
- Designing a Driver Circuit.
- Writing a program for displaying text on LCD.

- Designing of schematic diagram.
- Programming of Driver circuit according to the sensor and motor requirements.

- Search for alternatives if the desired one is not available.
- Biometric sensor Miaxis SM630.
- LCD and driver Circuit for motor.
- Power supply section.
- Wires, connectors, etc.

- General purpose PCB (for testing purpose)
- Copper platted PCB
- Etching
- Drilling
- Assembling

- Checking for shorts between the terminals
- Writing appropriate program for displaying text on the LCD.
- Writing a program for displaying user ID, text, and motor control, etc.

TABLE OF CONTENT

TOPIC **PAGE NO.**

CHAPTER 1: INTRODUCTION

1.1 INTRODUCTION

- The general trend of human nature is always longing for security Physically, Mentally and Socially. Fingerprint based security access control and time attendance systems are used for all kinds of office and service sector operations in the day-to-day environment. Fingerprint System based access control is more secured as compared to the conventional swipe card or ID cards because of the exclusive fingerprint for every entity.

- Fingerprint Security Systems have fascinated people. They have been used as a method of personal identification. The two key aspects of most of the Fingerprint System Biometric solutions are Finger Print identification and authentication. The process of identification tells us who an individual is, or in the negative sense tells us who they are not. Fingerprint Security is examined using two different sets of criteria. One way of looking at Fingerprint Security System is using their "Class Characteristics".

- Finger Print Security Systems can be used to get rid of so many issues such as Physical Access Control, Health care Biometrics, Fingerprint and Biometrics Locks, Biometric Sensors and Detectors, RFID Tags,

- RFID Readers, Road Barriers, RFID Smart Card, CCTV, Metal Detectors, LED Search Lights, Fire Alarm, Finger Print Movement Control, Physical Access Control, Optical Fingerprint Scanners, Optical Sensors, Card Locks Card Access Control Systems, Fingerprint Technology, Digital Fingerprint, USB Fingerprint Reader etc.

- Fingerprint System Authentication is a simpler process. It involves affirming or rejecting a claimed identity by matching a live template with an existing one. Most of these identification/ authentications are done by using Smart Card. The large growth of ID - Card based Fingerprint system security is wildly used in Public service applications. These cards are used for multiple purpose applications such as Digital parking meters, telephone, vending machines, ATM Cards, Digital Fingerprint Security System Identity, Personal Identification Verification Number etc.

- In this project the fingerprint module from Miaxis Biometrics is used. It can store up to 750 finger prints on its own memory. It can be controlled through its serial port.
- The microcontroller AT89S52 interacts with this module. We can add a fingerprint, delete a fingerprint and identify the fingerprint.
- To add a fingerprint, just put the finger on the module and press the ADD key. Now the microcontroller will send the ADD command to the module and the module will add it into the memory. To DELETE the finger follow the same as above.
- To identify the finger, press the Identify button and if the finger matches then the Relay is complemented. Also the fingerprint ID is displayed over the LCD display.
- This eliminates the need for keeping track of keys or remembering a combination password, or PIN. It can only be opened when an authorized user is present, since there are no keys or combinations to be copied or stolen, or locks that can be picked. Using Fingerprint saves time to gain access as compared to other methods like RFID card, Password or Key.

1.2 DESCRIPTION

- Personal Safes are revolutionary locking storage cases that open with just the touch of our finger. These products are designed as secure storage for medications, jewelry, weapons, documents, and other valuable or potentially harmful items. This utilizes fingerprint recognition technology to allow access to only those whose fingerprints we choose. It contains all the necessary electronics to allow us to store, delete, and verify fingerprints with just the touch of a button. Stored fingerprints are retained even in the event of complete power failure or battery drain. This eliminates the need for keeping track of keys or remembering a combination password, or PIN. It can only be opened when an authorized user is present, since there are no keys or combinations to be copied or stolen, or locks that can be picked.

- In this project the fingerprint module from Miaxis Biometrics is used. It can store up to 750 finger prints on its own memory. It can be controlled through its serial port.

- The microcontroller AT89S52 interacts with this module. We can add a fingerprint, delete a fingerprint and identify the fingerprint.

- To add a fingerprint, just put the finger on the module and press the ADD key. Now the microcontroller will send the ADD command to the module and the module will add it into the memory. To DELETE the finger follow the same as above.

- To identify the finger, press the Identify button and if the finger matches then the Relay is complemented. Also the fingerprint ID is displayed over the LCD display.

1.2 ADVANTAGES

- Fingerprint based security system is most secured system as compared to other systems. Reason is that RFID card or Keys of lock can be stolen, password may be leaked. However thumbnail of every human being is unique, so lock will not open unless the same person is present to give the impression of fingerprint.
- No need to carry the keys to open the lock or remember the password or any Pin number.
- One of the main advantages is that this system remembers the stored password even if the power supply is turned off.
- Scientific research and studies have proved that fingerprints do not change as you grow up.
- Using Fingerprint saves time to gain access as compared to other methods like RFID card, Password or Key.

1.4 COMMON APPLICATIONS

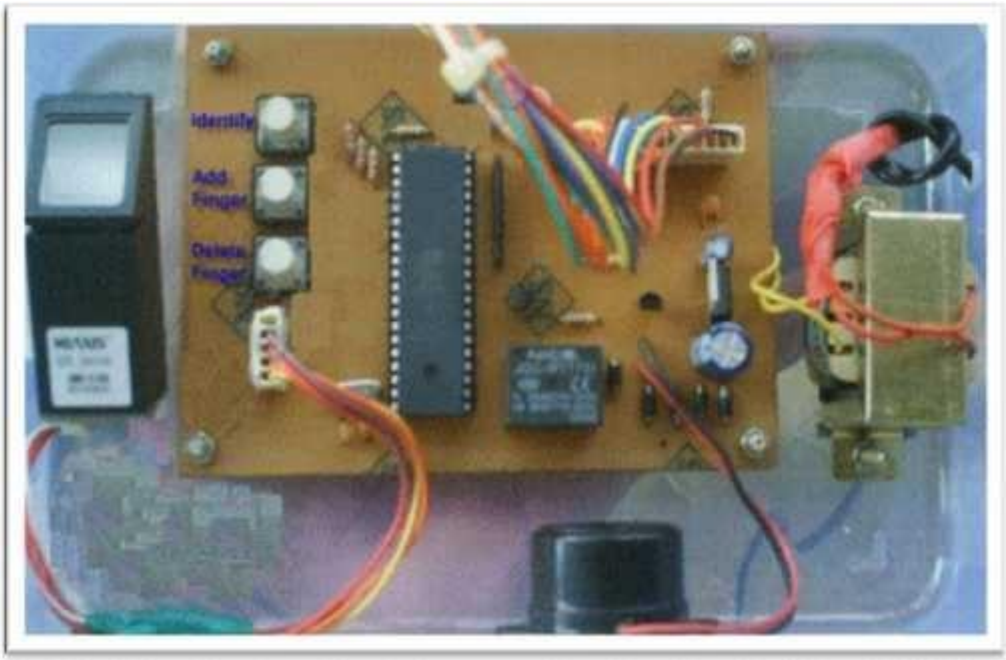
- Banking system
- Unique identification card
- Defense security
- Door lock system
- Network Logon
- Web access

- Airport checking
- ATMs
- Credit Cards
- Industrial application
- Home application

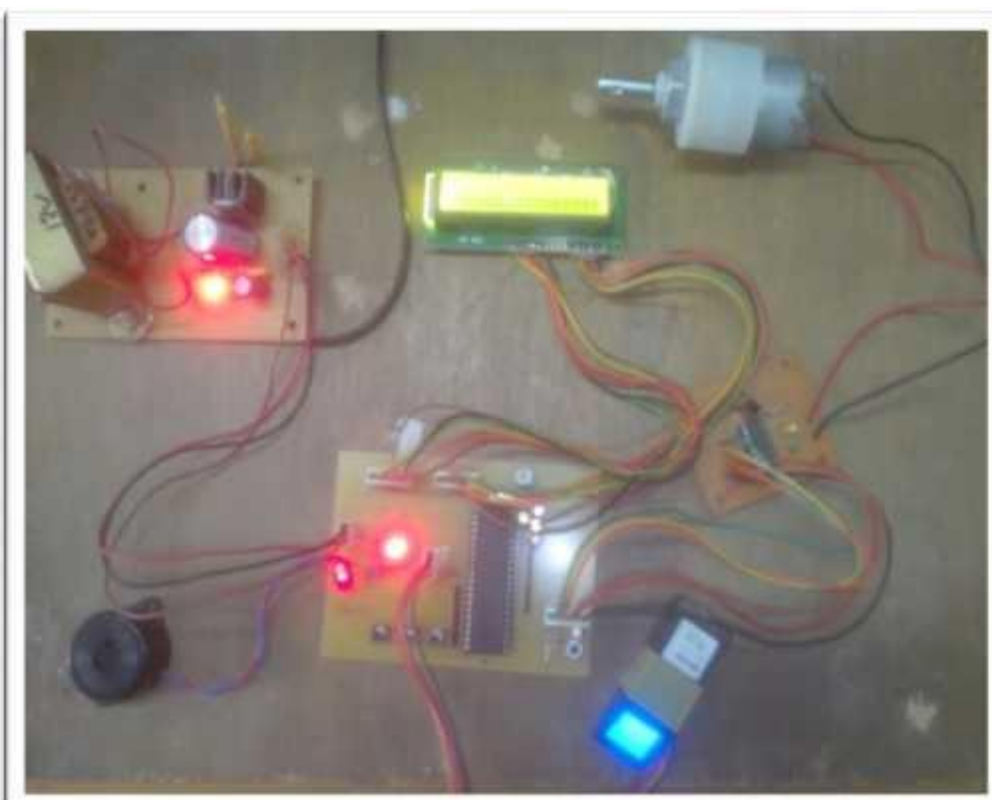
1.5 FUTURE DEVELOPMENTS

- The future work of this will be at the airport wherein the data will be stored in the PC so that the fraud which takes place will become less.
- We can send this data to a remote location using mobile or internet.
- We can use non-contact fingerprint sensor which is also called as touch-less 3D fingerprint scanner.
- We can implement other related modules like fire sensor, GSM modem.
- Improve the security performance in the voting machine.

1.6 EXPECTED PROJECT

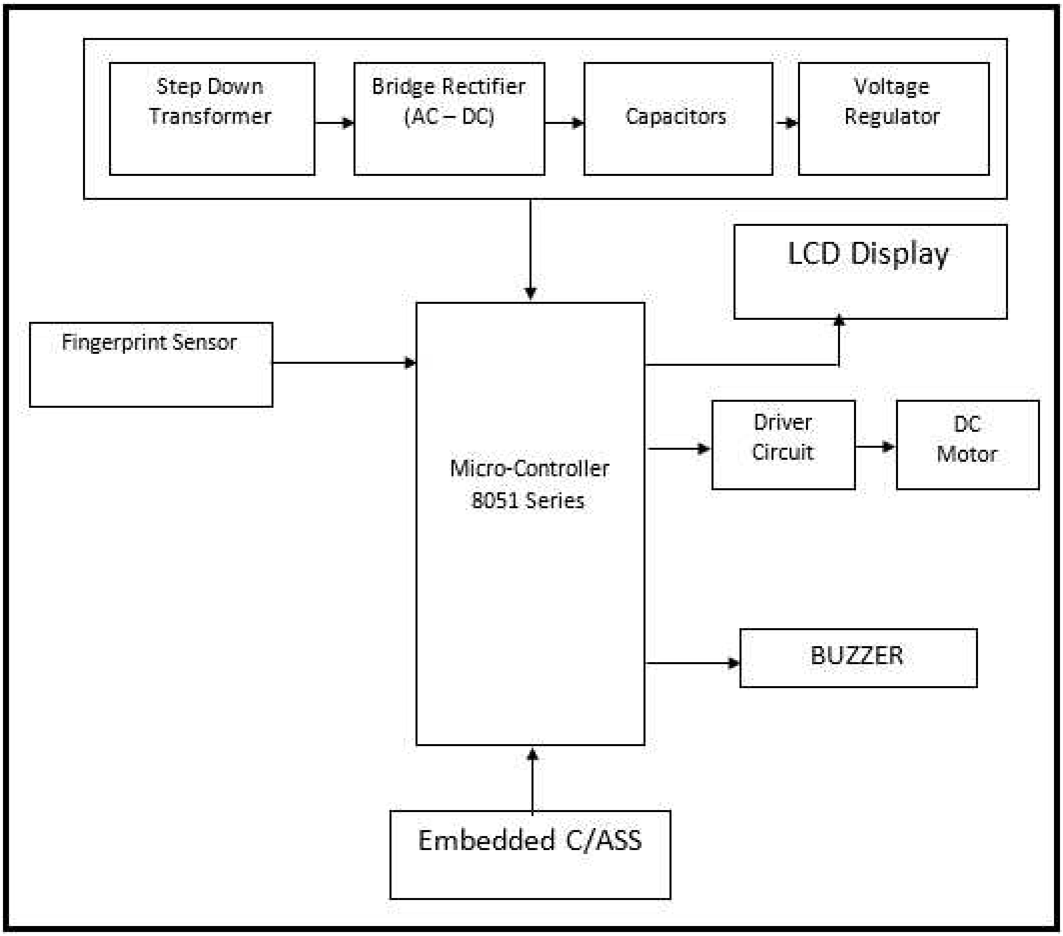


1.7 FINAL PROJECT



CHAPTER 2: LITERATURE REVIEW

2.1 Block Diagram



Block diagram consists of the following blocks:-

1. Power Supply
2. Microcontroller
3. Fingerprint Sensor Module
4. Driver Circuit
5. Buzzer
6. LCD
7. DC Motor

2.2 Circuit Diagram

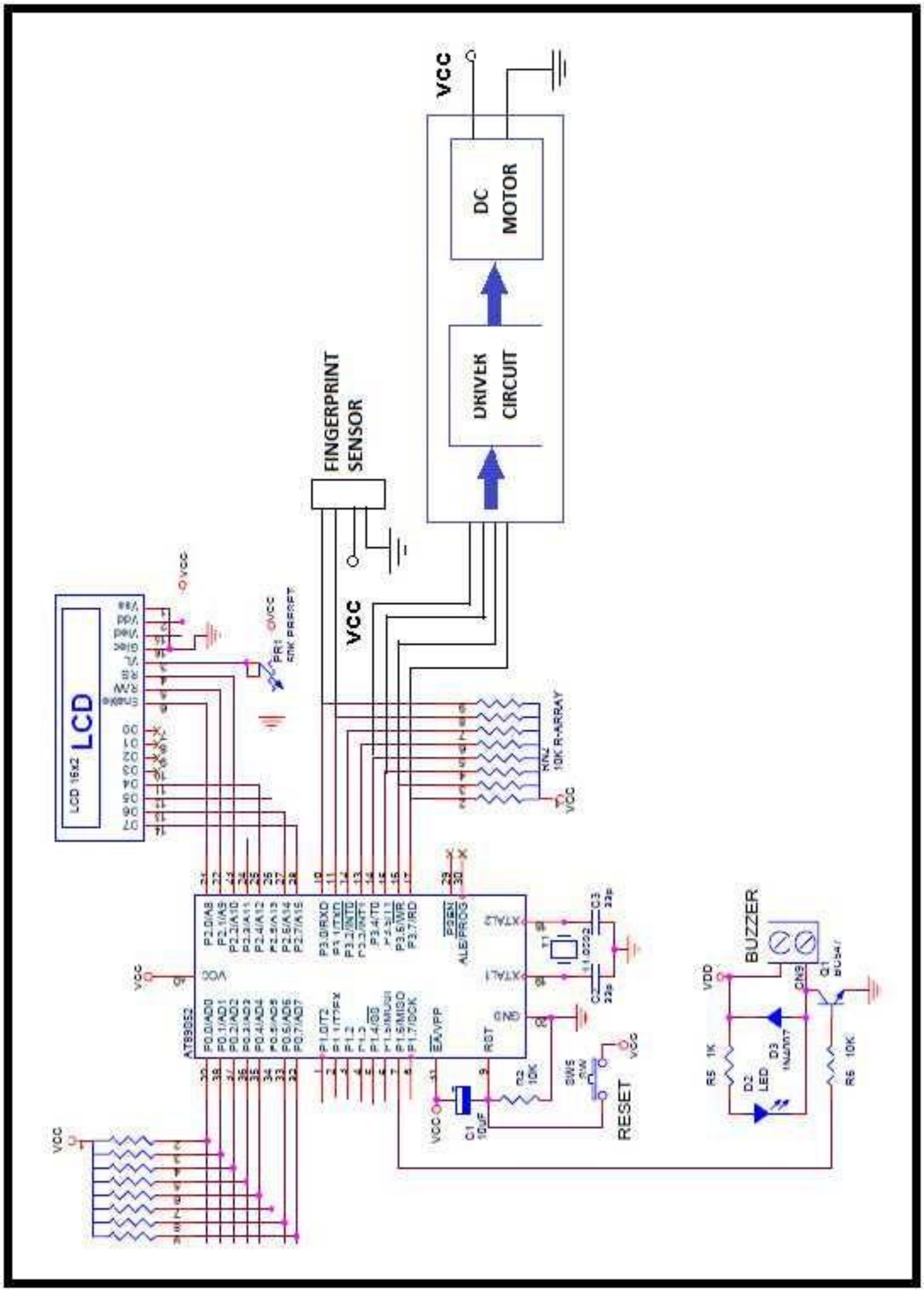
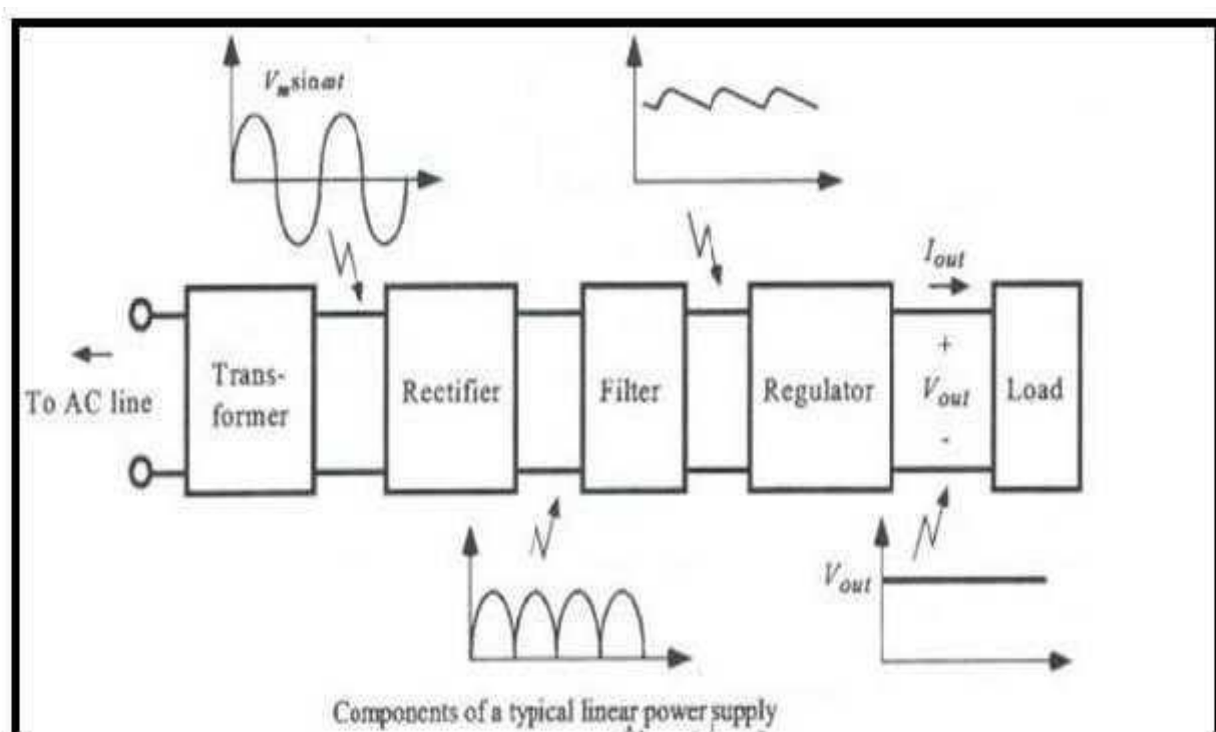


Figure shows the circuit diagram of a Fingerprint lock employing a Microcontroller IC 89S52.

It is divided into three parts:

1) Power supply section
2) Main circuit
3) Driver section

2.3 Power Supply Section



Components of a typical linear power supply

Figure shows block diagram of a regulated power supply.

Transformer:

- The transformer receives the input from ac mains that is 115/220/230V AC.
- Here a simple step-down transformer having turns per ratio 55:3 is used.
- The transformer step downs 220V AC into 12V AC which is fed to bridge rectifier circuit.
- The current rating of the used transformer is 1A.

Rectifier:

- Four diodes are used to form a bridge rectifier.
- Here four diodes / PN junction diode IN4007 having forward voltage / knee voltage of 0.7V and made up of silicon material is used to convert input 12VAC into pulsating DC output
- Rectifier efficiency is 81.2%
- TUF is 0.812
- Ripple frequency is 2*Fin
- Ripple factor is 0.482
- Peak voltage is Vm
- DC output voltage is 0.636Vm

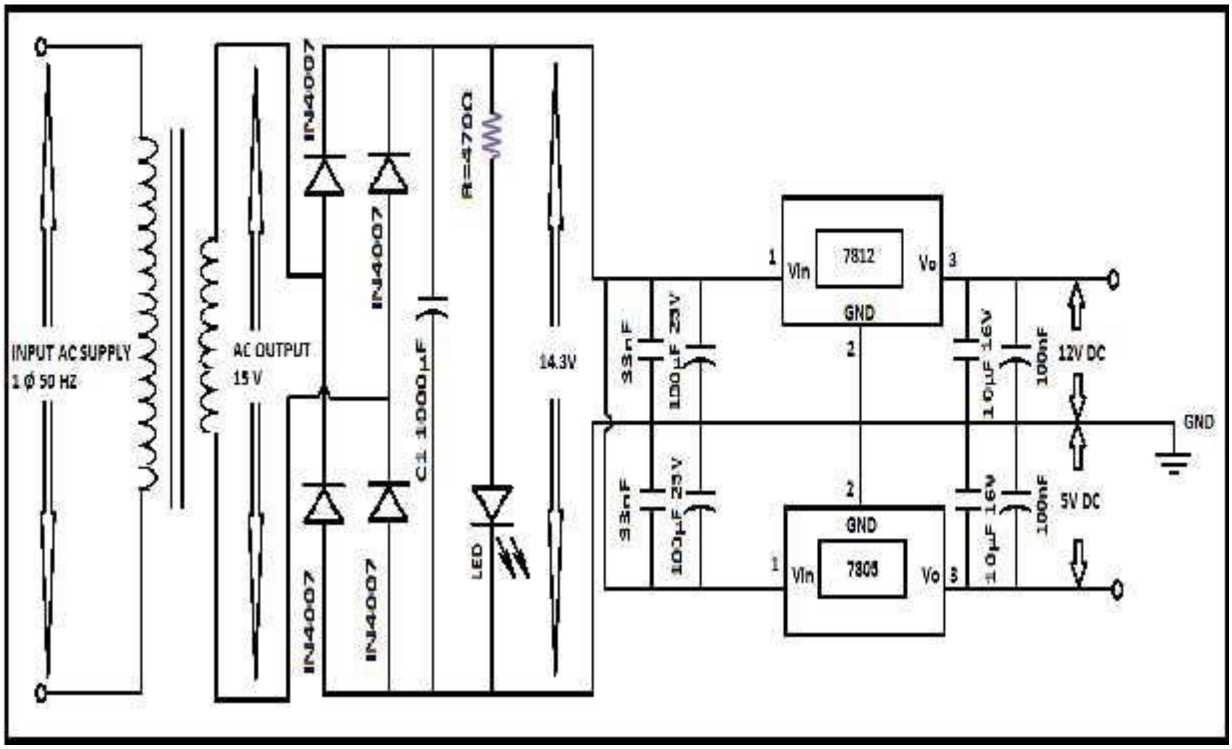
Filter:

- The output of bridge rectifier is not pure DC voltage it contains some unwanted AC components called ripples hence in order to remove it and to obtain a pure DC as output voltage a filter is ought to be used.
- Here a 1000µF 35V capacitor is used as filter.
- It is used for smoothing purpose that is to bypass AC components and to give pure DC as output.

Regulator:

- The output of filter is given to voltage regulator.
- The regulator is used to maintain output voltage constant in-spite of variations in input supply / AC mains or any change in load.
- Its handles the load and line regulation problem and stabilizes / maintains output voltage constant.
- Here 78xx series is used to meet the requirement of main circuit and driver circuit as well as dc motor.

POWER SUPPLY CIRCUIT DIAGRAM

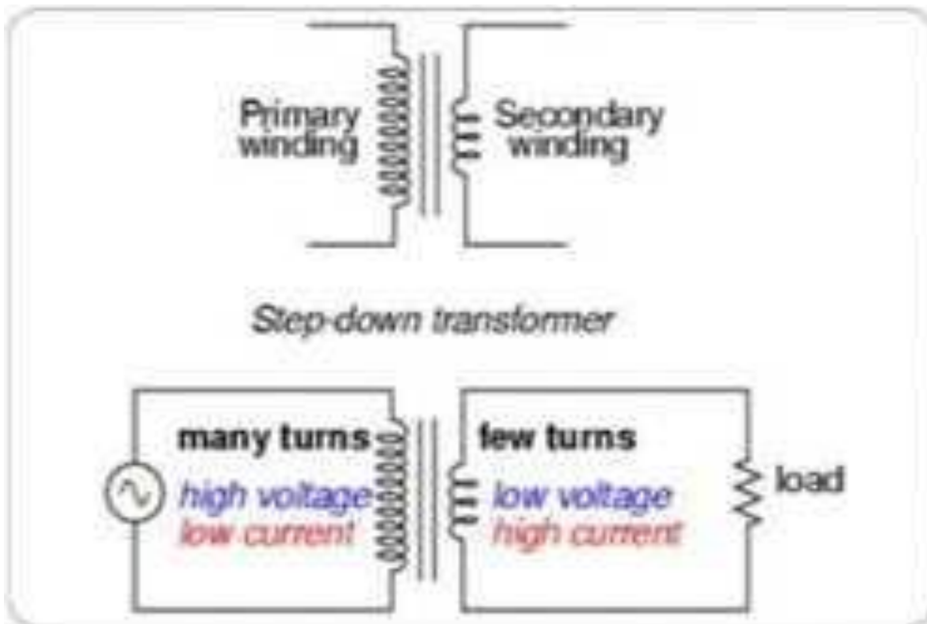


RATINGS OF IN4007

MAX. REVERSE VOLTAGE : 50V

MAX. FORWARD CURRENT : 1A

TRANSFORMER



TURNS RATIO = (Vp / Vs) = (Np / Ns)

where,

Vp = primary (input) voltage.

Vs = secondary (output) voltage

Np = number of turns on primary coil

Ns = number of turns on secondary coil

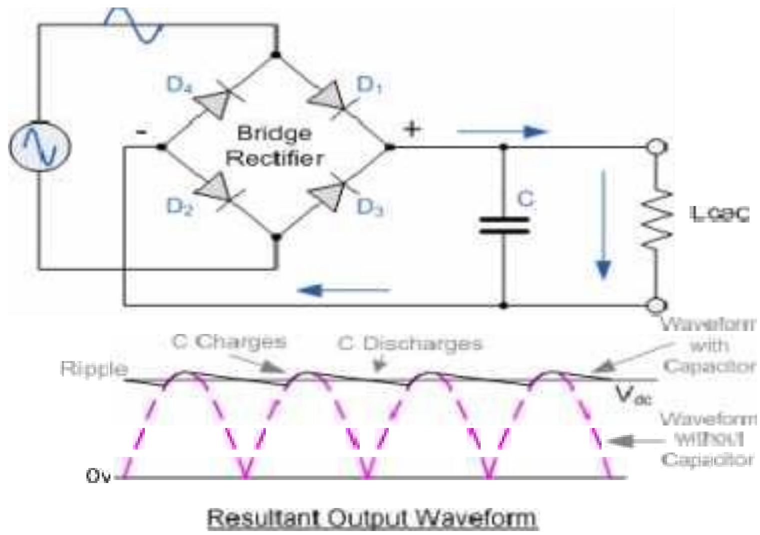
Ip = primary (input) current

Is = secondary (output) current.

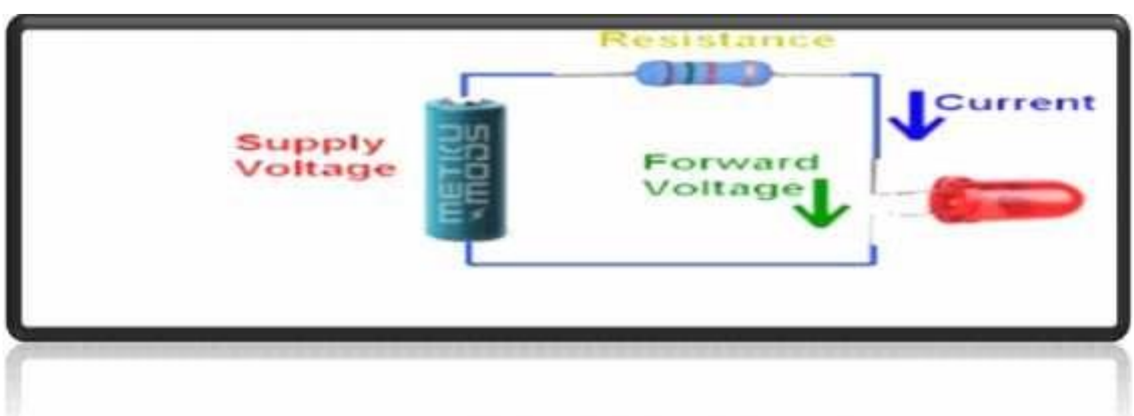
$$\frac{V_s}{V_p} = \frac{N_s}{N_p} = \frac{I_p}{I_s}.$$

FWR WITH FILTER

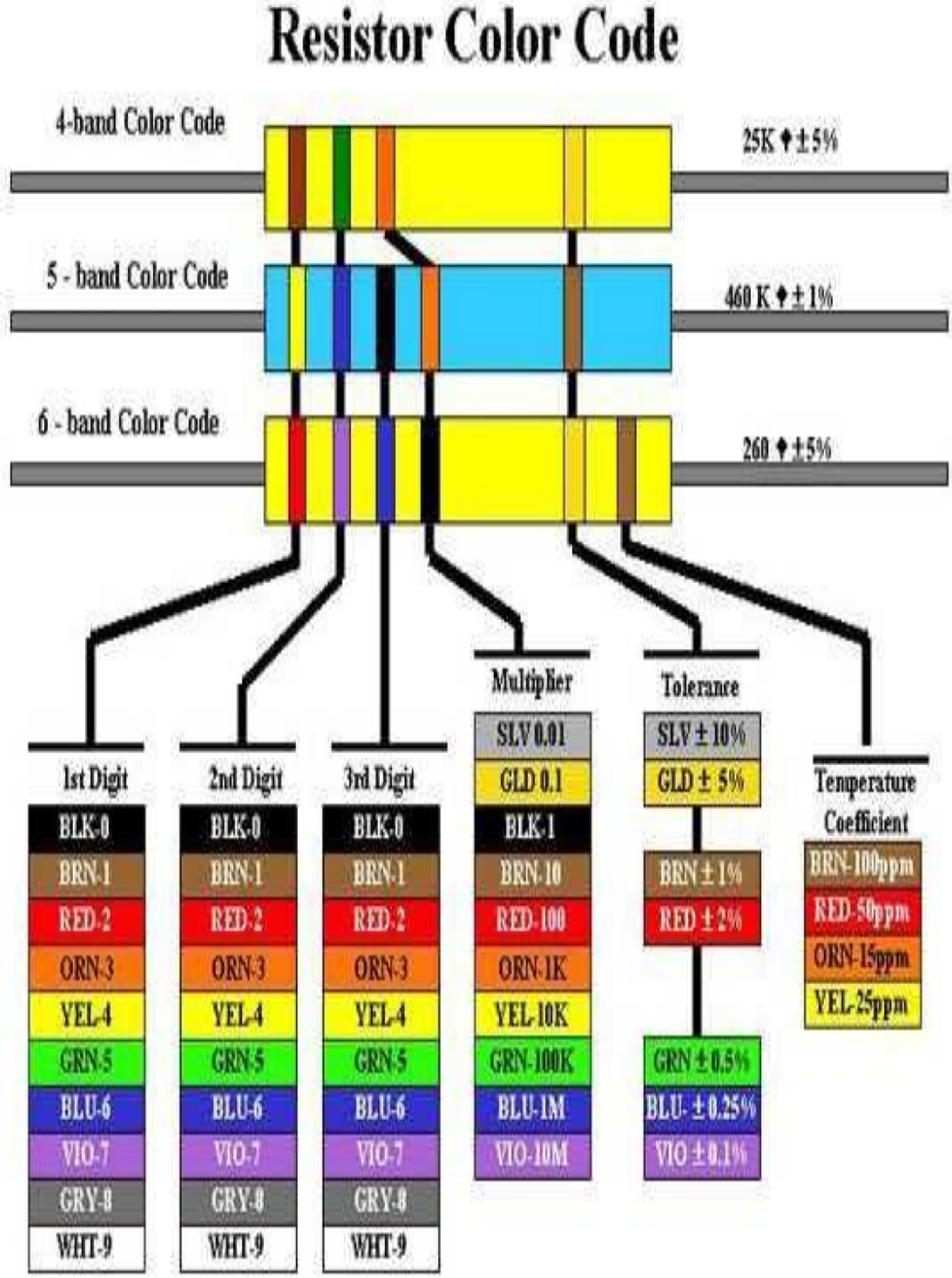
The capacitor used in circuit is **ALUMINIUM ELECTROLYTIC** capacitor of 1000µ F and 35V rating is used as smoothing capacitor. Figure shows the waveform across smoothing capacitor.



For indication of proper functioning of transformer, rectifier, and filter a resistor and led is connected in parallel.



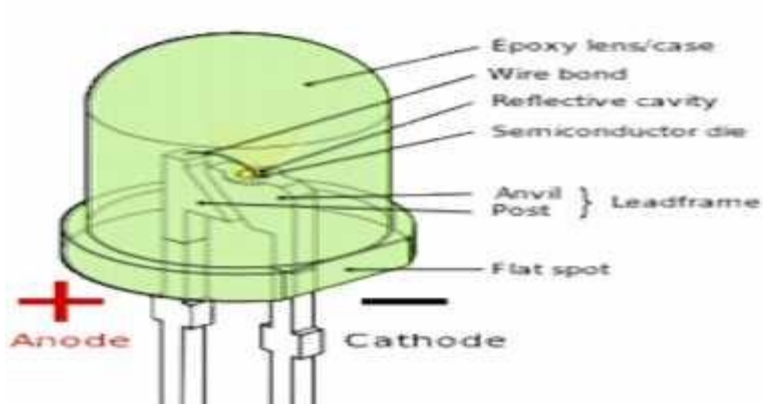
Here fixed resistor of 1KΩ is used to limit the current so that the indication led is not damaged.

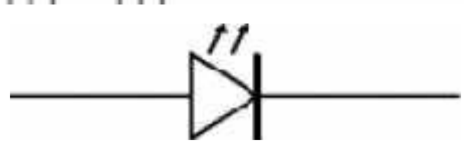

Resistor Color Code
4-band Color Code
25K ♦ ± 5%
5 - band Color Code
460 K ♦ ± 1%
6 - band Color Code
260 ♦ ± 5%
1st Digit
BLK-0
BRN-1
RED-2
ORN-3
YEL-4
GRN-5
BLU-6
VIO-7
GRY-8
WHT-9
2nd Digit
BLK-0
BRN-1
RED-2
ORN-3
YEL-4
GRN-5
BLU-6
VIO-7
GRY-8
WHT-9
3rd Digit
BLK-0
BRN-1
RED-2
ORN-3
YEL-4
GRN-5
BLU-6
VIO-7
GRY-8
WHT-9
Multiplier
SLV 0.01
GLD 0.1
BLK-1
BRN-10
RED-100
ORN-1K
YEL-10K
GRN-100K
BLU-1M
VIO-10M
Tolerance
SLV ± 10%
GLD ± 5%
BRN ± 1%
RED ± 2%
GRN ± 0.5%
BLU- ± 0.25%
VIO ± 0.1%
Temperature Coefficient
BRN-100ppm
RED-50ppm
ORN-15ppm
YEL-25ppm

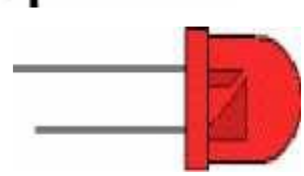
LED

PRINCIPLE OF OPERATION

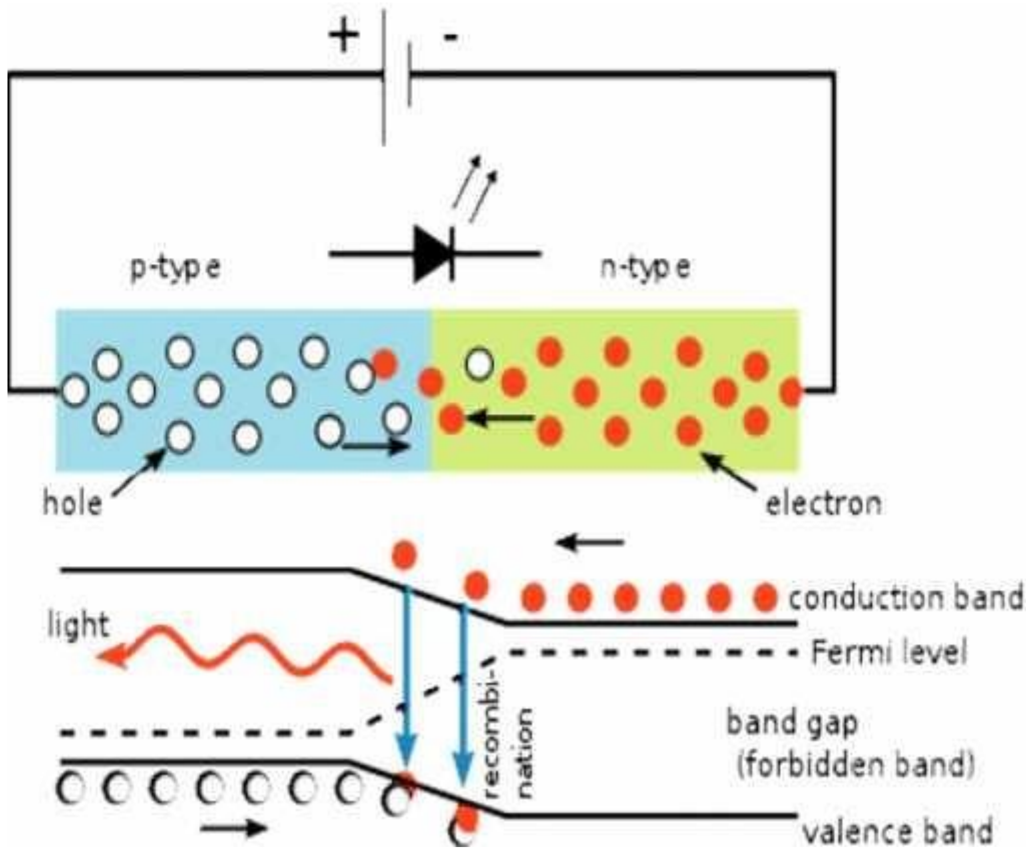
A P-N junction diode which emits lights when forward biased is known as light emitting diode. The amount of light output is directly proportional to the forward current. Thus higher the forward current, higher is the light output. When the LED is forward biased, the electrons and the holes move towards the junction and the recombination takes place. After recombination, the electrons lying in the conduction band of N- region falls into the holes lying in the valence band of a P-region. The difference of energy between the conduction band and valence band is radiated in the form of light energy in ordinary diodes. This energy is radiated in the form of heat. The materials used for the manufacturing of LED’s are gallium phosphide, gallium arsenide, gallium arsenide phosphide.

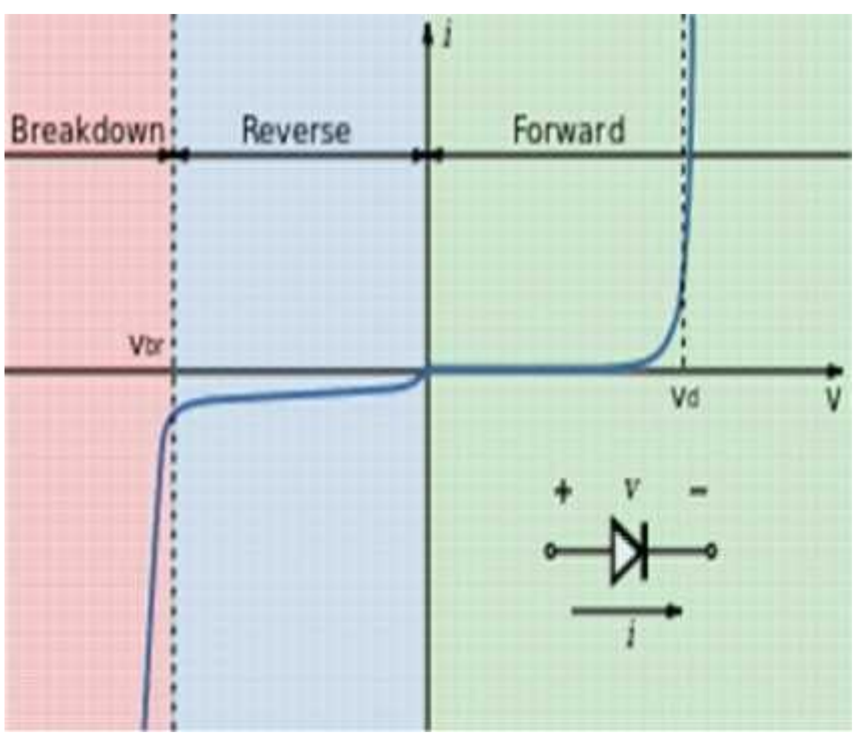


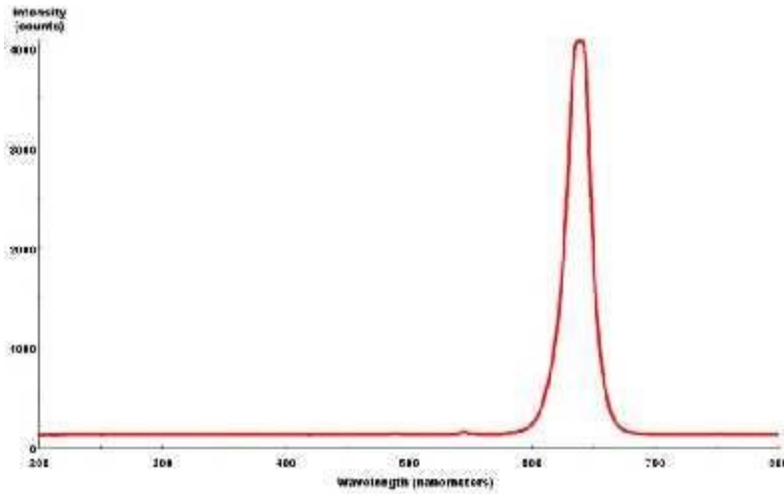




WORKING:


+
-
p-type
n-type
hole
electron
light
conduction band
Fermi level
band gap
(forbidden band)
recombi-nation
valence band


i
Breakdown
Reverse
Forward
Vbr
Vd
V
+
v
-
i



LED is used for indication purpose. Figure shows wavelength spectrum of **RED LED**

REGULATOR : **78XX SERIES**

Regulator

The Voltage Regulator is a circuit that will give a constant output voltage in-spite of changes in its input voltage or load current (IL).

Need of Regulator

i) To maintain constant output voltage in-spite of input voltage.

ii) To maintain constant output voltage in-spite of IL changes.

NAME	VOLTAGE
LM7805	+ 5 volts
LM7812	+ 12 volts

FEATURES OF 78XX (POSITIVE SERIES VOLTAGE REGULATOR)

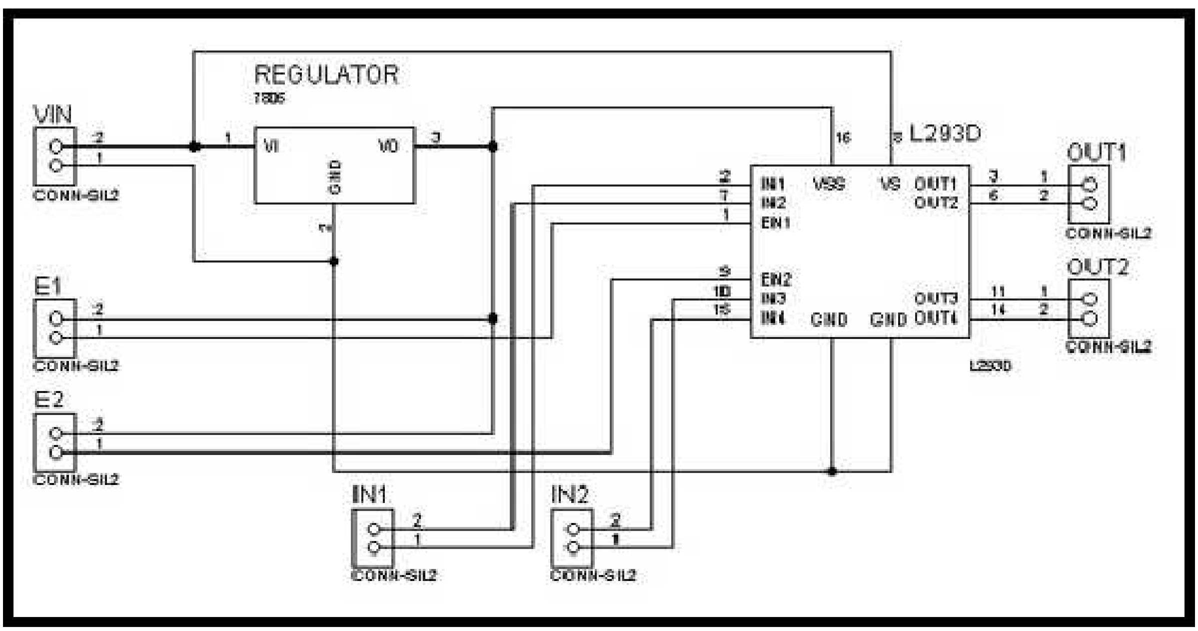
1) Output current up to 1A

2) Thermal overload protection

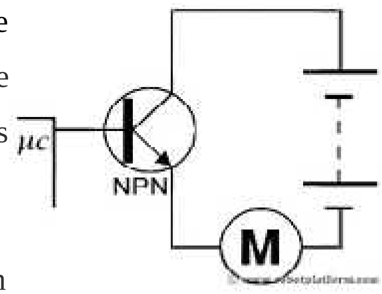
3) Short circuit protection

4) Output transistor safe operating area protection

2.4 Driver L293D



In our circuit only one motor is driven. Hence In2, En2 are disabled. MOTORS require more current then the microcontroller pin can typically generate hence a switch is required.



Switch such as Transistors, MOSFET, and Relay etc. which accepts a small current, amplify it and generate a larger current, which further drives a motor. This entire process is done a **motor driver**.

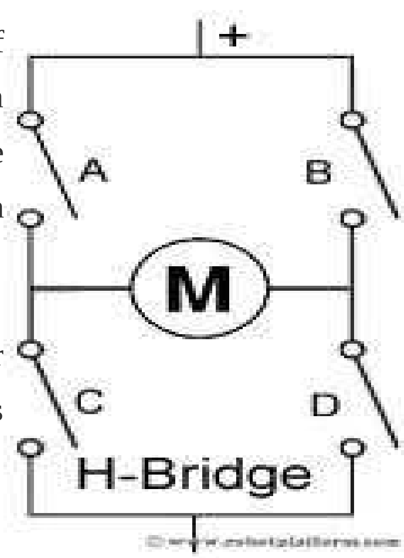
Motor driver is basically a current amplifier which takes a low-current signal from the microcontroller and gives out a proportionally higher current signal which can control and drive a motor. In most cases, a transistor can act as a switch and perform this task which drives the motor in a single direction.

Turning a motor ON and OFF requires only one switch to control a single motor in a single direction.

By reversing motor polarity direction of rotation can be changed.

This can be achieved by using four switches that are arranged in an intelligent manner such that the circuit not only drives the motor, but also controls its direction. Out of many, one of the most common and clever design is an H-bridge circuit where transistors are arranged in a shape that resembles the English alphabet "H".

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



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

H-bridges can be built from scratch using relays, MOSFETS, Field Effect Transistors (FET), bi-polar junction transistors (BJT), etc.

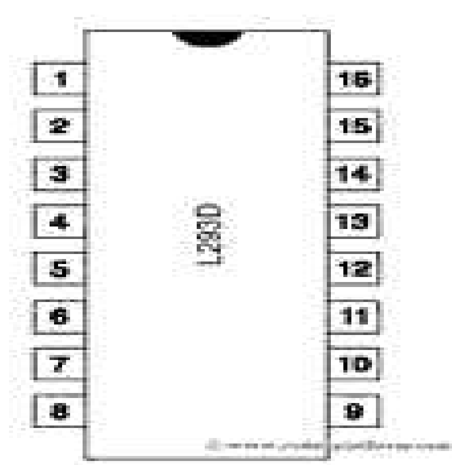
For small current requirement L293D IC is used.

L293D IC can interface two DC motors.

L293, L293B and few other versions also does the same job, but pick the L293D version as this one has an inbuilt flyback diode which protects the driving transistors from voltage spikes that occur when the motor coil is turned off.

Introduction to 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 microcontroller pins (if we do not use enablepins).



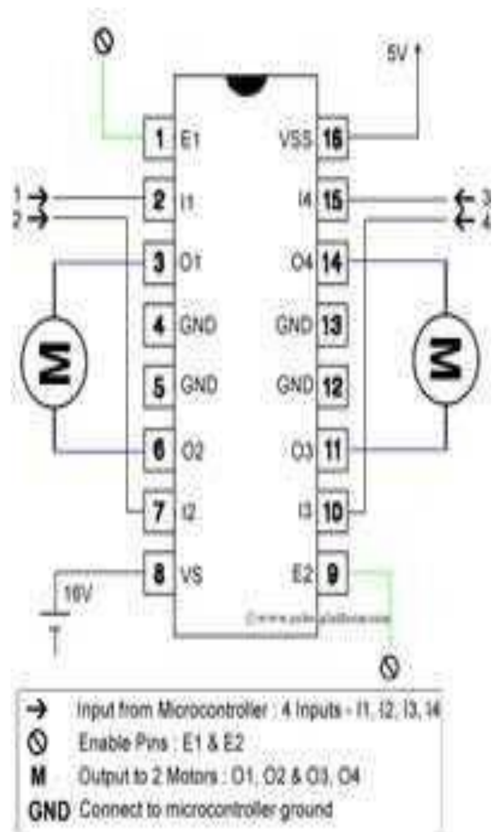
Some of the features of this IC are:

1. Output current capability is limited to 600mA per channel with peak output current limited to 1.2A (non-repetitive). This means we cannot drive bigger motors with this IC. However, most small motors used in hobby robotics should work. If we are unsure whether the IC can handle a particular motor, connect the IC to its circuit and run the motor with our finger on the IC. If it gets really hot, then beware... Also note the words "non-repetitive" if the current output repeatedly reaches 1.2A, it might destroy the drive transistors.
2. Supply voltage can be as large as 36V. This means we do not have to worry much about voltage regulation.
3. L293D has an enable facility that helps us to enable the IC output pins. If an enable pin is set to logic high, then state of the inputs match the state of the outputs. If we pull this low, then the outputs will be turned off regardless of the input states.
4. The datasheet also mentions an "over temperature protection" built into the IC. This means an internal sensor senses its internal temperature and stops driving the motors if the temperature crosses a set point.
5. Another major feature of **L293D** is its internal clamp diodes. This flyback diode helps us to protect the driver IC from voltage spikes that occur when the motor coil is turned on and off (mostly when turned off).
6. The logical low in the IC is set to 1.5V. This means the pin is set high only if the voltage across the pin crosses 1.5V which makes it suitable for use in high frequency applications like switching applications (up to 5KHz).
7. Lastly, this integrated circuit not only drives DC motors, but can also be used to drive relay solenoids, stepper motors etc.

L293D Connections

The circuit shown below is the most basic implementation of L293D IC (16 pin).

1. Pin 1 and Pin 9 are "Enable" pins. They should be connected to +5V for the drivers to function. If they pulled low (GND), then the outputs will be turned off regardless of the input states, stopping the motors. If we have two spare pins in our microcontroller, connect these pins to the microcontroller, or just connect them to regulated positive 5V.
2. Pin 4, Pin 5, Pin 12 and Pin 13 are ground pins which should ideally be connected to microcontroller's ground.
3. Pin 2, Pin 7, Pin 10 and Pin 15 are logic input pins. These are control pins which should be connected to microcontroller pins. Pin 2 and Pin 7 control the first motor (left); Pin 10 and Pin 15 control the second motor (right).
4. Pin 3, Pin 6, Pin 11, and Pin 14 are output pins. Tie Pin3 and Pin6 to the first motor, Pin11 and Pin14 to second motor.
5. Pin16 powers the IC and it should be connected to regulated +5V.
6. Pin 8 powers the two motors and should be connected to positive lead of a secondary battery. As per the datasheet, supply voltage can be as high as 36 V.



TRUTH TABLE

High +5V

Low 0V

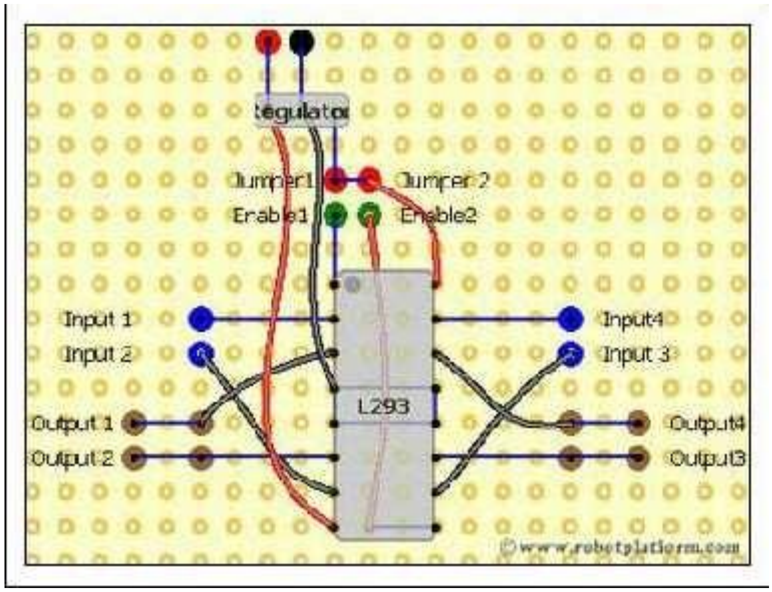
X Either high or low (don't care)

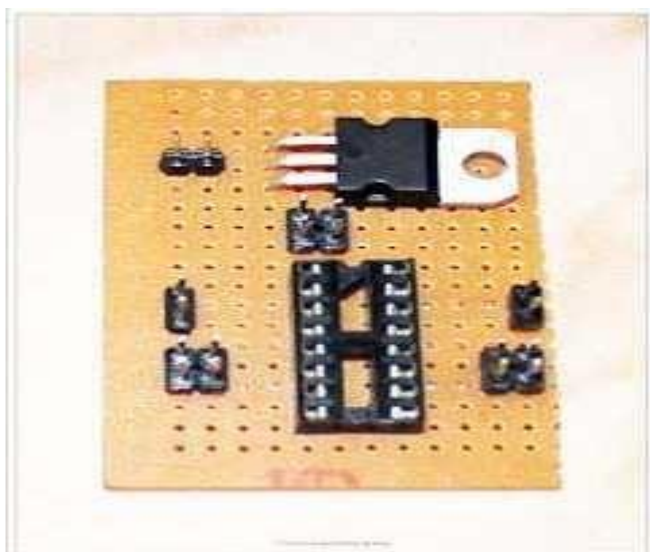
In the below truth table we can observe that if Pin 1 (E1) is low then the motor stops, irrespective of the states on Pin 2 and Pin 7. Hence it is essential to hold E1 high for the driver to function or simply connect enable pins to positive 5V.

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

With Pin 1 high, if Pin 2 is set high and Pin 7 is pulled low, then current flows from Pin 2 to Pin 7 driving the motor in anti-clockwise direction. If the states of Pin 2 and Pin 7 are flipped, then current flows from Pin 7 to Pin 2 driving the motor in clockwise direction.

The above concept holds true for other side of the IC too. Connect the motor to Pin 11 and Pin 14; Pin 10 and Pin 15 are input pins and Pin 9 (E2) enables the driver.





Whenever we implement it, always remember that enable pins are not connected. Either connect enable pins to your microcontroller pins and programmatically set it high or use a jumper and connect each enable pin to the header just above it (which is connected to +5V).

2.5 Microcontroller AT89S52

The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 8KB of In-System Programmable Flash memory. The device is manufactured using ATMEL's high-density non-volatile memory technology and is compatible with the industry standard 8051 instruction set and pin out. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional non-volatile 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

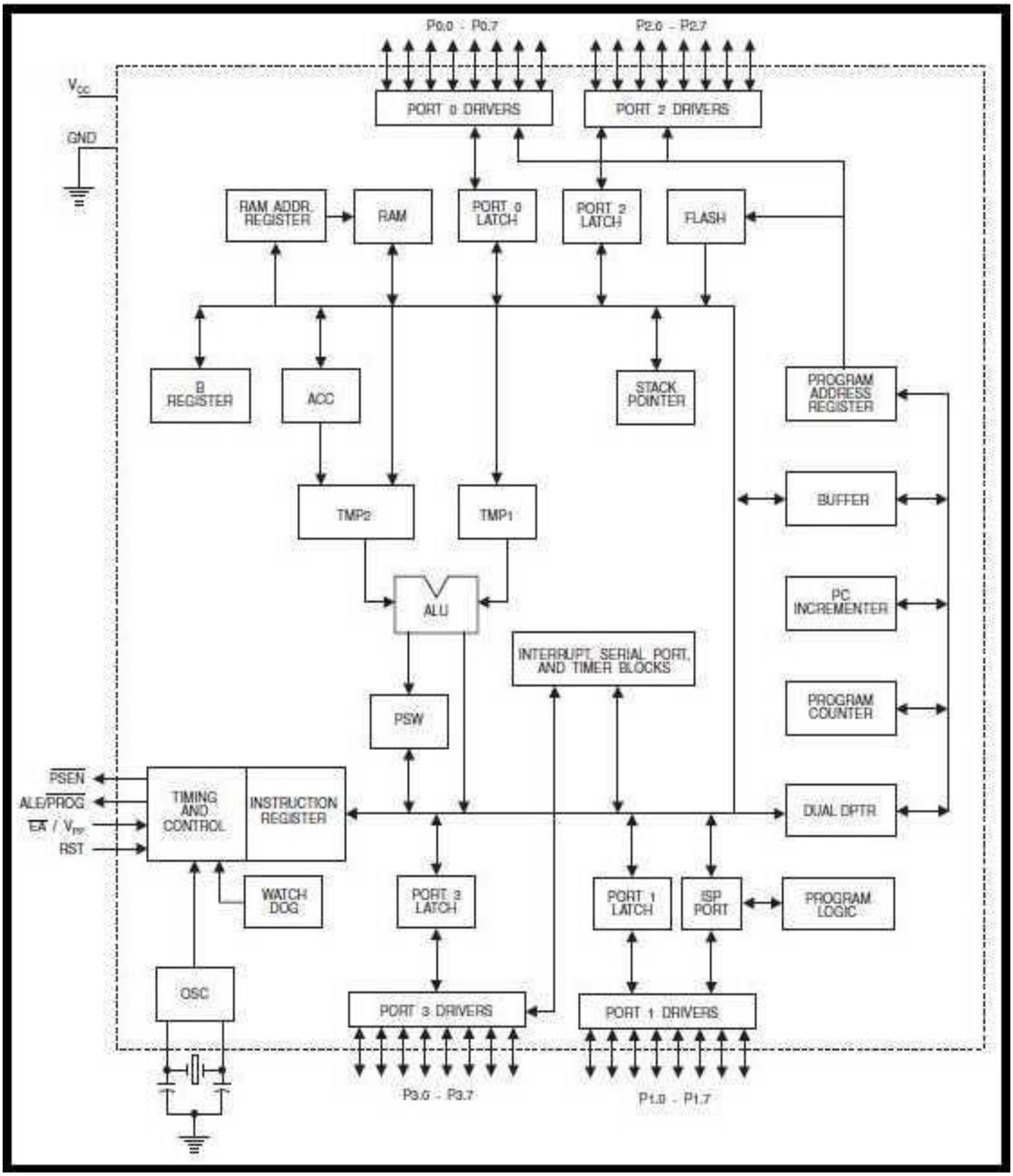
- 8K bytes of Flash
- 256 bytes of RAM
- 32 I/O lines
- Watchdog timer
- Two data pointers
- Three 16-bit timer/counters
- A six-vector two-level interrupt architecture
- A full duplex serial port
- On-chip oscillator
- Clock circuitry
- In addition, the AT89S52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes.

The Idle Mode stops the CPU by 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.

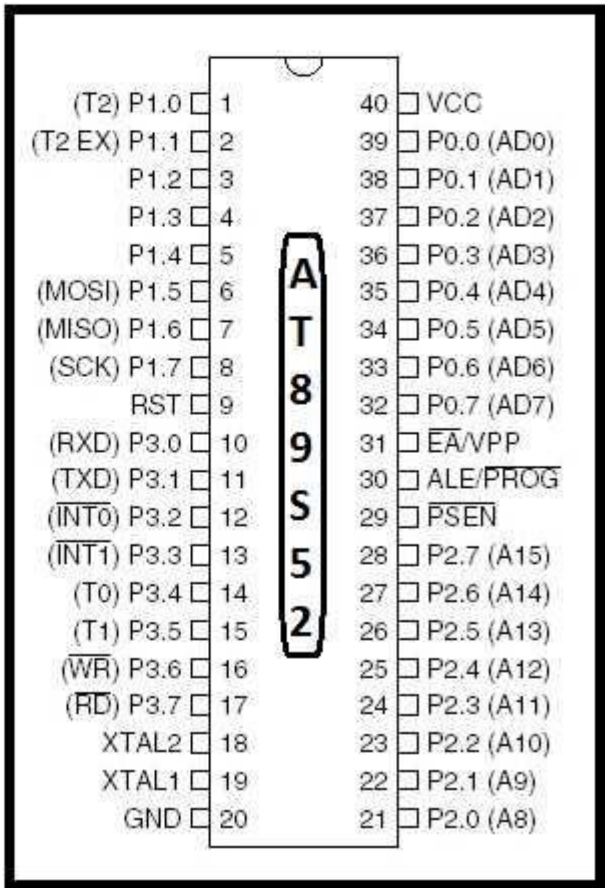
FEATURES

- Compatible with MCS®-51 Products
- 8K Bytes of In-System Programmable (ISP) Flash Memory
- 4.0V to 5.5V Operating Range
- Fully Static Operation: 0 Hz to 33 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Eight Interrupt Sources
- Full Duplex UART Serial Channel
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down Mode
- Watchdog Timer
- Dual Data Pointer
- Power-off Flag
- Fast Programming Time
- Flexible ISP Programming (Byte and Page Mode)
- Green (Pb/Halide-free) Packaging Option.

BLOCK DIAGRAM OF AT89S52



PIN CONFIGURATION OF AT89S52



PIN DIAGRAM OF AT89S52

PIN DESCRIPTION :

VCC :

Supply voltage.

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 access to external program and data memory. In this mode, P0 has internal pull-ups. 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. In addition, P1.0 and P1.1 can be configured to be the timer / counter 2 external count input (P1.0/T2) and the timer / counter 2 trigger input (P1.1/T2EX).

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 (IIL) because of the internal pull-ups. Port 2 emits the high-order address byte during fetches from external program memory and during access to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

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 (IIL) because of the pull-ups.

RST (RESET INPUT) :

A high on this pin for two machine cycles while the oscillator is running resets the device. This pin drives high for 98 oscillatory periods after the Watchdog times out. The DISRTO bit in SFR AUXR (address 8EH) can be used to disable this feature. In the default state of bit DISRTO, the RESET HIGH out feature is enabled.

ALE/PROG :

Address Latch Enable (ALE) is an output pulse for latching the low byte of the address during access 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 of the oscillator frequency and may be used for external timing or clocking purposes. One ALE pulse is skipped during each access to external data memory.

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, except that two PSEN activations are skipped during each access to external data memory.

EA/VPP :

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. If lock bit 1 is programmed, EA will be internally latched on reset. EA should be strapped to VCC for internal program executions. This pin also receives the 12V programming enable voltage (VPP) during Flash programming.

XTAL1 :

It is the input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2 :

It is the output from the inverting oscillator amplifier.

OSCILATOR CHARACTERISTICS :

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in

Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 6.2. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clock circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

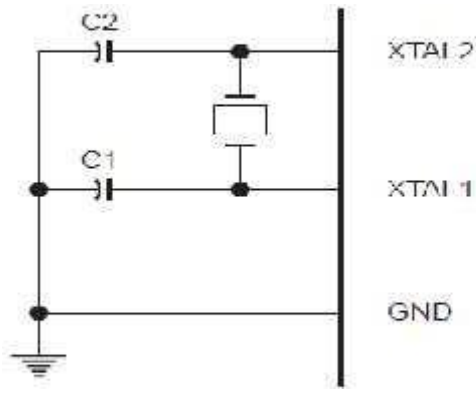


FIG 1: Oscillator Connections

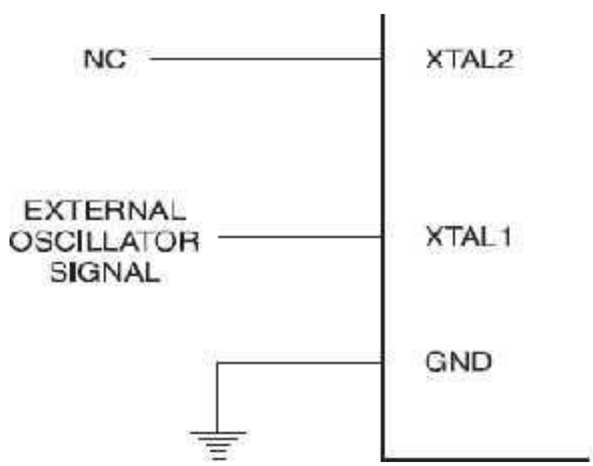


FIG 2: External Clock Drive Configuration

IDLE MODE :

In idle mode, the CPU puts itself to sleep while all the on chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the Special Functions Registers (SFR) remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

POWER DOWN MODE :

In the power down mode the oscillator is stopped, and the instruction that invokes power down is the last instruction executed. The on-chip RAM and Special Function Registers (SFR) retain their values until the power down mode is terminated. The only exit from power down is hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before VCC is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

2.6 BIOMETRIC SENSOR SM630 :

1) Easy to use protocol with LEDs, Switch and built-in controller.

2) The fingerprint sensor can read different fingerprints and store in its own flash memory.

3) The sensor can perform three functions namely Add (Enroll), Empty Database or Search Database and authenticates the ID of stored fingerprint.

4) Any of three functions can be called simply by making the pin low of the sensor or pressing onboard three switches.

5) The response is either ERROR or OK which is indicated by onboard LED.

6) The response is also returned as single serial data-byte.

7) The authenticated byte is a valid ID or error code.

8) The response byte is a single byte at 9600 bps thus making whole sensor very easy to use.

BASIC FUNCTION OF FINGERPRINT SCANNER :

It takes an image of our fingertip and compares it against the data of a previously scanned fingerprint. If the two match, then security is granted.

It compares the patterns of “ridges” and “valleys,” as the bumps on our fingertip are called, and determines if all elements match.

Most fingerprint scanners are made up of optical scanners, consisting of a Charge Coupled Device (CCD), a light sensory system commonly used among digital cameras and camcorders.

When we place our finger over the glass, the CCD takes a picture of the thumb print. It illuminates the ridges of our fingertip and creates an inverted image of the picture to place attention on the valleys.

If the image taken is of an acceptable level of quality and accuracy a decision made by the CCD based on the image definition and pixel darkness it begins comparing the captured fingerprint with the images stored on file. Another common form of fingerprint capturing is the capacitance scanner, which instead of sensing the print using light, it utilizes an electrical current. The ridges of our skin, the dermis, are electrically non-conductive, whereas the valleys of our fingertip, the sub-dermal layer, are conductive.

When touching a fingerprint sensor, it measures the minutest differences in conductivity caused by the presence of ridges.

Using these measurements, capacitance scanners create a picture of the fingerprint to compare against the original fingerprint.

Once our fingerprint is scanned, the sensor compares this new picture against the pre-stored print to determine if they match.

Comparing a full fingerprint against the newly scanned image is impractical because smudging could make an identical print look different from the scanned image. That method would also take a lot of processing power.

Instead, fingerprint scanners compare specific areas of the fingerprint, referred to as “minutiae.” Scanners and professional investigators concentrate on points where ridge lines end or split into multiple ridges known as bifurcations.

A single finger could have numerous minutiae, and rather than try to identify all of them, scanners simply need to find a preprogrammed number of them to determine the two prints are identical. When setting your initial fingerprint, the scanner will identify a series of minutiae and store that data rather than a picture of the entire fingerprint, thus ensuring higher levels of security and replicating a fingerprint based on these details alone is virtually impossible.

FEATURES :

● **Self-proprietary Intellectual Property**

Optical fingerprint collection device, module hardware and fingerprint algorithm are all self-developed by Miaxis.

● **High Adaptation to Fingerprint**

When reading fingerprint images, it has self-adaptive parameter adjustment mechanism, which improves imaging quality for both dry and wet fingers. It can be applied to wider public.

● **Low Cost**

Module adopts Miaxis' optical fingerprint collection device, which dramatically lowers the overall cost.

● **Algorithm with Excellent Performance**

SM630 module algorithm is specially designed according to the image generation theory of the optical fingerprint collection device. It has excellent correction & tolerance to deformed and poor-quality fingerprint.

● **Easy to Use and Expand**

User does not have to have professional know-how in fingerprint verification. User can easily develop powerful fingerprint verification application systems based on the rich collection of controlling command provided by SM630 module. All the commands are simple, practical and easy for development.

● **Low Power Consumption**

Operational Current <80mA , especially good for battery power occasions.

- **Integrated Design**

Fingerprint processing components and fingerprint collection components are integrated in the same module. The size is small. And there are only 4 cables connecting with HOST, much easier for installation and use.

- **Perfect Technical Support**

MIAXIS is the leading company in the fingerprint verification industry. It has an excellent customer service team ready to offer powerful technical support in user development of single byte. The code to send data to PC using MCU TX pin or add an LCD to show various functions and IDs returned.

TECHNICAL SPECIFICATIONS

Operating Voltage :

4.3V～6V

Rating Voltage :

6.5V（exceeding this value will cause permanent damage to the module）

Operating Current：

<80mA（Input voltage 5V）

Fingerprint Template :

768 templates

Search Time：

<1.5s（200 fingerprint, average value in test）

Power-on Time：

<200ms（Time lapse between module power-on to module ready to receive instructions）

Tolerated Angle Offset：

±45°

User Flash Memory：

64KByte

Interface Protocol :

Standard Serial Interface （TTL level）

Communication Baud Rate :

57600bps

Operating Environment：

Temperature: -10℃～+40℃

Relative humidity: 40％RH～85％RH（no dew）

2.7 LCD 16x2

INTRODUCTION

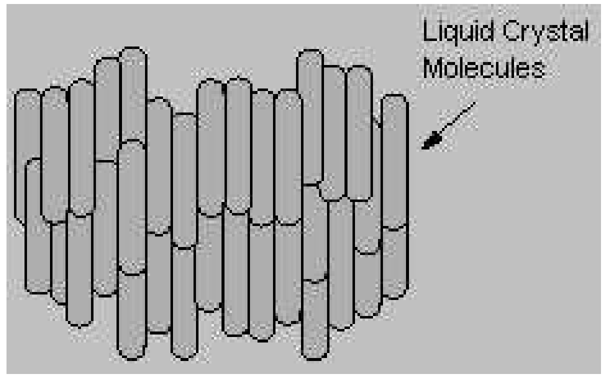
LCD creates images on a flat surface by shining light through a combination of liquid crystals and polarized glass. This technology differs from CRT because a CRT uses a beam of electrons projected through a large glass tube to create images.

ADVANTAGES

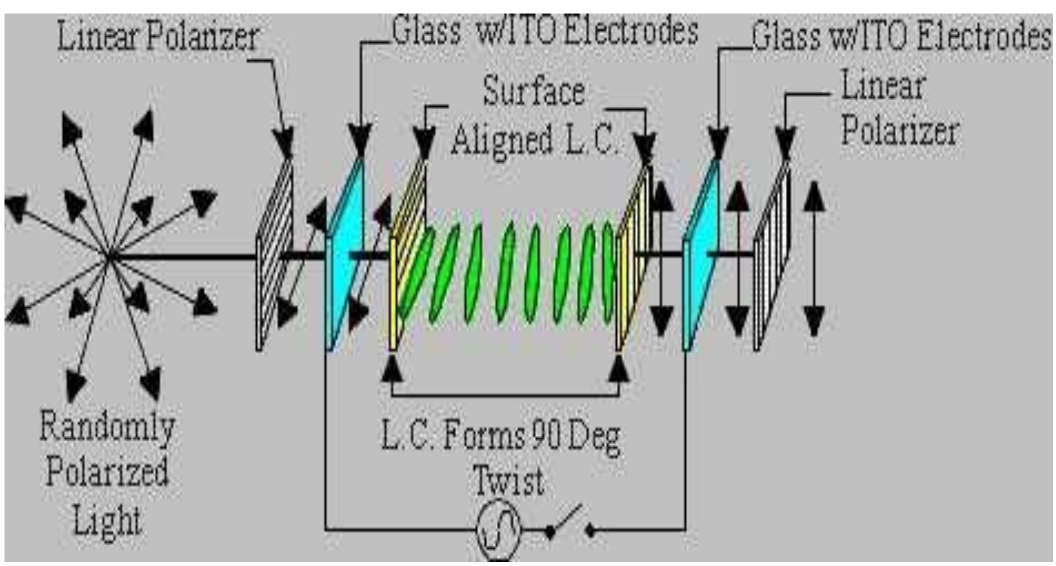
- Smaller size
- Less eye-strain
- Lower power consumption
- Less heat generation
- Lighter weight
- Better image contrast
- Energy Savings

Liquid crystal displays (LCDs) are a passive display technology. This means they do not emit light. Instead, they use the ambient light in the environment. By manipulating this light, they display images using very little power.

This has made LCDs the preferred technology whenever low power consumption and compact size are critical. Liquid crystal (LC) is an organic substance that has both a liquid form and a crystal molecular structure. In this liquid, the rod-shaped molecules are normally in a parallel array, and an electric field can be used to control the molecules. Most LCDs today use a type of liquid crystal called Twisted Nematic (TN). A visual of the molecule alignment is shown.



A Liquid Crystal Display (LCD) consists of two substrates that form a "flat bottle" that contains the liquid crystal mixture. The inside surfaces of the bottle or cell are coated with a polymer that is buffed to align the molecules of liquid crystal. The liquid crystal molecules align on the surfaces in the direction of the buffing. For Twisted Nematic devices, the two surfaces are buffed orthogonal to one another, forming a 90 degree twist from one surface to the other as shown in the figure given below.

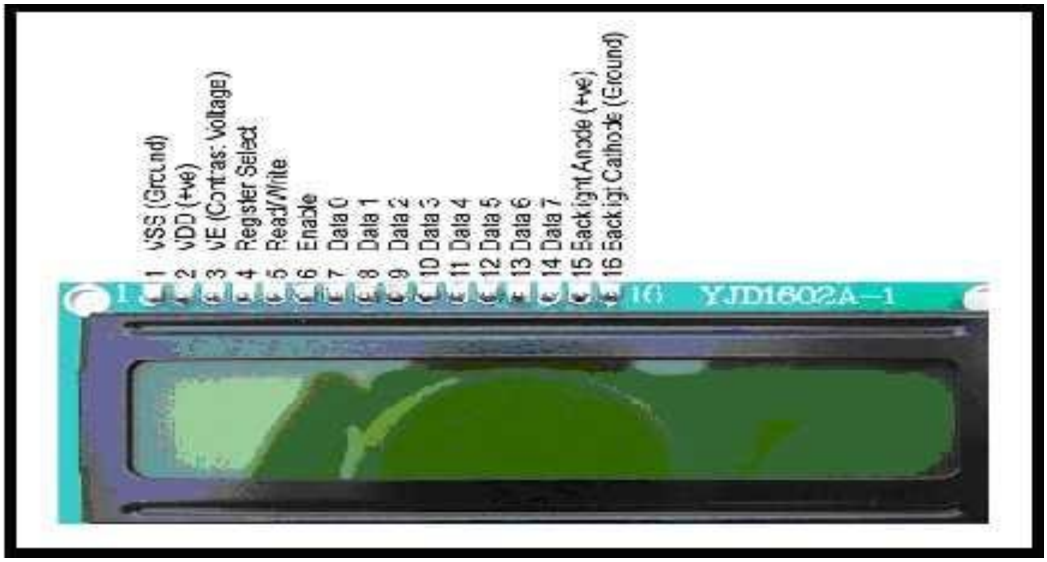


This helical structure has the ability to control light. A polarizer is applied to the front and an analyzer / reflector is applied to the back of the cell. When randomly polarized light passes through the front polarizer it becomes linearly polarized. It then passes through the front glass and is rotated by the liquid crystal molecules and passes through the rear glass. If the analyzer is rotated 90 degrees to the polarizer, the light will pass through the analyzer and be reflected back through the cell.

The observer will see the background of the display, which in this case is the silver gray of the reflector. The LCD glass has transparent electrical conductors plated onto each side of the glass in contact with the liquid crystal fluid and they are used as electrodes. These electrodes are made of Indium-Tin Oxide (ITO). When an appropriate drive signal is applied to the cell electrodes, an electric field is set up across the cell. The liquid crystal molecules will rotate in the direction of the electric field. The incoming linearly polarized light passes through the cell unaffected and is absorbed by the rear analyzer. The observer sees a black character on a sliver gray background. When the electric field is turned off, the molecules relax back to their 90 degree twist structure. This is referred to as a positive image, reflective viewing mode. Carrying this basic technology further, an LCD having multiple selectable electrodes and selectively applying voltage to the electrodes, a variety of patterns can be achieved.

Many advances in TN LCDs have been produced. Super Twisted Nematic (STN) Liquid Crystal material offers a higher twist angle (>=200° vs. 90°) that provides higher contrast and a better viewing angle. However, one negative feature is the birefringence effect, which shifts the background color to yellow-green and the character color to blue.

This background color can be changed to a gray by using a special filter.



LCD Pin	Symbol	Function	External Connection
1	Vss	Signal Ground (GND)	External ground (Power Section)
2	Vdd	VCC for LCD	Power supply for logic (+5v)
3	Vo	Contrast Adjust	Externally connect potentiometer
4	RS	Register Select signal	To micro-cotroller control pins
5	R/W	Read / Write select signal	To micro-cotroller control pins
6	E	Enable signal	To micro-cotroller control pins
7	DB0	Four low order bi-dirctional three-state data bus lines. These four are not used if 4-bit interface used	To micro-cotroller DATA pins
8	DB1		To micro-cotroller DATA pins
9	DB2		To micro-cotroller DATA pins
10	DB3		To micro-cotroller DATA pins
11	DB4	Four high order bi-dirctional three-state data bus lines. These pins used when 4-bit interface is used	To micro-cotroller DATA pins
12	DB5		To micro-cotroller DATA pins
13	DB6		To micro-cotroller DATA pins
14	DB7		To micro-cotroller DATA pins
15	1	LED (K)	Back light LED cathode terminai
16	15	LED (A)	Back light LED anode terminal

R/W (READ/WRITE SELECT SIGNALS) :

1) R/W= 0, the information is being written on LCD.

2) R/W=1, the information is being read from LCD.

Only one command that is “Get LCD status” is a read command all others are write command.
It is a control line. So R/W will be low for majority of the time.

REGISTERS :

There are two very important registers in the LCD. The RS pin is used for their selection.

1) RS= 0 : The Instruction command code register, allows the user to send command such as clear display, cursor at home, etc.

2) RS=1 : The data register allows the user to send data to be displayed at LCD.

ENABLE (EN) PIN :

It is used to tell the LCD that we are sending it data. A high to low pulse (of minimum length 450ns) before sending any command/data to LCD should be given.

How many characters can we send to LCD?

Display Data RAM (DDRAM) stores display data represented in 8-bit character codes. Its extended capacity is 80 x 8 bits, or 80 characters. The area in Display Data RAM (DDRAM) that is not used for display can be used as general data RAM. So whatever we send on the DDRAM is actually displayed on the LCD.

{For LCDs like 1x16, only 16 characters are visible, so whatever you write after 16 chars is written in DDRAM but is not visible to the user.}

00	01	02	03	C4	05	06	07		32	33	34	35	36	37	38	39	← Character position (dec.)
00	01	02	03	C4	05	06	07	••••••	20	21	22	23	24	25	26	27	← Row0 DDRAM address (hex)
40	41	42	43	44	45	46	47	••••••	60	61	62	63	64	65	66	67	← Row1 DDRAM address (hex)

(DDRAM address for 2 line LCD)

How does the ASCII value change to characters?

The answer is CGROM i.e. Character Generator ROM. It is used to convert ASCII values send by µC into 5 x 8 dot or 5 x 10 dot character patterns from 8-bit character.

(We can also add your own characters in the list)

How much time should be there in two consecutive Commands / Instructions ?

- To create a delay between two consecutive commands or instructions.

(Check the datasheet for exact time of execution of an instruction).

This may not be very appropriate as the delay is not very accurate.

- **BUSY FLAG (BF)**

The MSB of the LCD data bus (D7) acts as busy flag.

When BF = 1 means LCD is busy and will not accept next command or data.

When BF = 0 means LCD is ready for the next command or data to process.

This flag is internally set by LCD & can be monitored by µC for exact amount of delay.

To read Busy Flag, the conditions RS = 0 and R/W = 1 must be met.

STEPS TO PROGRAM:

1) Initialize the LCD.

2) Select the command or instruction register.

3) Set RW low (to write).

4) Send a high to low pulse on Enable pin.

5) Check if the LCD is busy.

6) Move to instruction or command function.

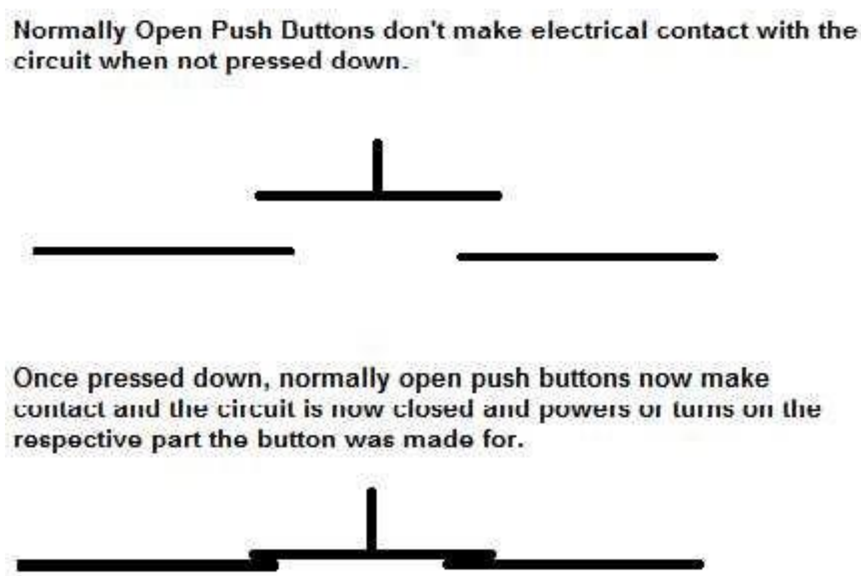
7) Repeat steps 4-7.

COMMON COMMANDS FOR LCD :

No.	Instruction	Hex	Decimal
1	Function Set: 8-bit, 1 Line, 5x7 Dots	0x30	48
2	Function Set: 8-bit, 2 Line, 5x7 Dots	0x38	56
3	Function Set: 4-bit, 1 Line, 5x7 Dots	0x20	32
4	Function Set: 4-bit, 2 Line, 5x7 Dots	0x28	40
5	Entry Mode	0x06	6
6	Display off Cursor off (clearing display without clearing DDRAM content)	0x08	8
7	Display on Cursor on	0x0E	14
8	Display on Cursor off	0x0C	12
9	Display on Cursor blinking	0x0F	15
10	Shift entire display left	0x18	24
12	Shift entire display right	0x1C	30
13	Move cursor left by one character	0x10	16
14	Move cursor right by one character	0x14	20
15	Clear Display (also clear DDRAM content)	0x01	1
16	Set DDRAM address or coursor position on display	0x80+add	128+add

2.8 Normally Closed (NC) Push Button

A Normally Open (NO) Push Button is a push button that, in its default state, makes no electrical contact with the circuit. Only when the button is pressed down does it make electrical contact with the circuit.



When the button is pressed down, the switch makes electrical contact and the circuit is now closed. Therefore, electricity can now flow to the other part of the circuit connecting to the push button and make the device turn or power on the respective part. Normally Open Push buttons are the most common type of push buttons used in devices and circuits.

SWITCH BASICS

A switch is a component which controls the openness or closeness of an electric circuit. They allow to control the over current flow in a circuit (without having to actually get in there and manually cut or splice the wires). Switches are critical components in any circuit which requires user interaction or control.

A switch can only exist in one of two states: open or closed. In the OFF state, a switch looks like an open gap in the circuit. This, in effect, looks like an OPEN CIRCUIT, preventing current from flowing.

In the ON state, a switch acts just like a piece of perfectly-conducting wire. A short CLOSES THE CIRCUIT, turning the system “ON” and allowing current to flow unimpeded through the rest of the system.

In order to change from one state to another, a switch must be actuated i.e. some sort of physical action must be performed to “flip” the switch’s state.
Momentary switches only remain active as long as they are actuated. If they are not being actuated, they remain in their “OFF” state.

2.9 IC Mounting Socket

An IC socket or Integrated Circuit socket, is used in devices that contain an integrated circuit. An IC socket is used as a placeholder for IC chips and is used in order to allow safe removal and insertion of IC chips because IC chips may become damaged from heat due to soldering.

TYPES OF IC SOCKETS

There are several different kinds of IC sockets at Future Electronics. We stock many of the most common types categorized by several parameters including type, style, number of pins, termination style / mounting, centerline / pitch and packaging type. Our parametric filters will allow us to refine your search results according to the required specifications.

IC Sockets from Future Electronics

Future Electronics has a wide range of IC sockets from several manufacturers that can be used for your integrated circuits. By going through the selection we will find the right IC socket of any pin size (8 pin, 14 pin, 16 pin, 18 pin, 20 pin, 28 pin, 40 pin,…), IC test sockets, IC socket connector, IC breadboard socket, IC socket adapter or any other type of IC socket once we decide. If we need Adaptors, BGA, DIMM, PGA, PLCC or SIP Sockets, Dual in Line Package Sockets or Test & Burn IC sockets, we will be able to choose from their technical attributes and our search results will be narrowed to match your specific IC socket application needs.

We deal with several manufacturers such as Aries, Mill-Max, TE Connectivity or Yamaichi, among others. We can easily refine our IC socket product search results by clicking our preferred IC socket brand from the list of manufacturers.

APPLICATIONS OF IC SOCKETS

IC sockets are used in applications where integrated circuit devices have short lead pins. They help in providing safe removal and insertion of IC chips. They are often found in desktop and server computers. They are also used for prototyping new circuits because they allow easy component swapping.

CHOOSING THE RIGHT IC SOCKET

With the FutureElectronics.com parametric search, when looking for the right IC sockets, we can filter the results by category. We carry the following categories of IC sockets:

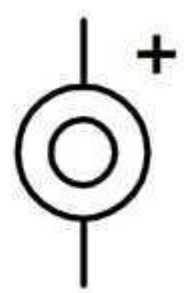
- Adaptors
- BGA
- DIMM
- Dual in Line Package Sockets
- PGA

- PLCC
- SIP Sockets
- Test & Burn

Once we choose the IC socket category, we can narrow them down by various attributes: by number of pins, type, style and termination style to name a few. We will be able to find the right IC test sockets, IC breadboard socket, IC socket adapter, IC socket connector or any other IC socket of any pin size (8 pin, 14 pin, 16 pin, 18 pin, 20 pin, 28 pin, 40 pin,…) by using these filters.

2.10 BUZZER

A **buzzer** or a **beeper** is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke.



Electronic symbol for a buzzer

2.11 50K PRESET

It is a small potentiometer which is used for adjustment, tuning and calibration in circuits. When it is used as a variable resistance (wired as a rheostat) they are called **preset resistors.** They are also known as trimpots.

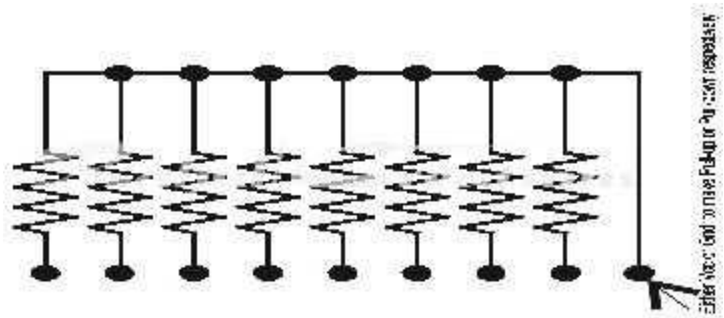
Trimpots or presets are normally mounted on Printed Circuit Boards (PCB) and adjusted by using a screwdriver.

The material used as a resistive track is varying, but the most common is either carbon composition or cermets. PRESETS are designed for occasional adjustment and can often achieve a high resolution when using multi-turn setting screws. When trimmer potentiometers/preset are used as a replacement for normal potentiometers, care should be taken as their designed lifespan is often only 200 cycles. It is connected to pin 3 of LCD display VL which is used for contrast control that is brightness control of LCD display.

2.12 10K ARRAY

Array can be used to pull-up or pull-down the microcontroller.

Equivalent circuit



This is a 10K resistor array in 9Pin package. With each pin equivalent to a resistor having resistance 10KΩ.

As resistor does not have any polarity the common terminal , it can be connected to either VCC or GND.

In our hardware it is used for pull up.

Pull up strategy:

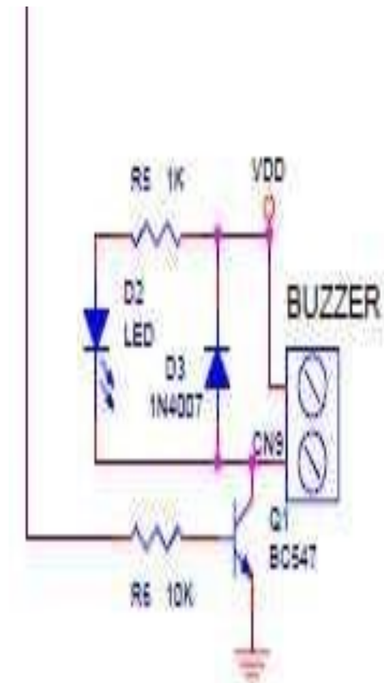
This resistor array package is useful for pull up one 8 bit port.

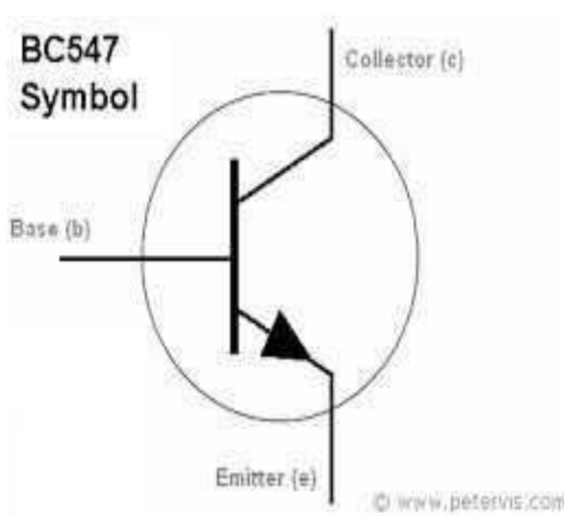
For this to achieve the first pin that is common terminal should be connected to VCC.

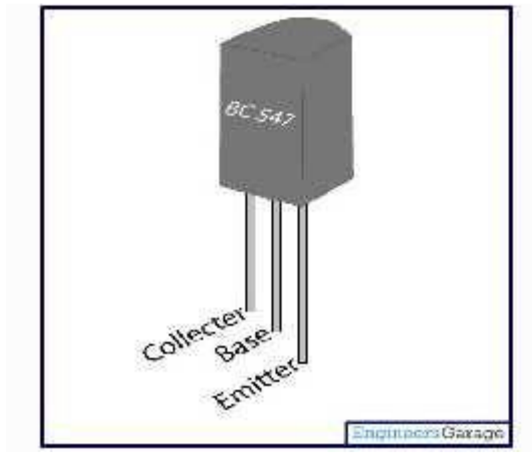
Pull down strategy:

This can be achieved as pull down operation if we connect pin 1 to GND and other pins to the controller port pins.

2.13 Transistor BC547







Since microcontroller 89S52 is not capable of sourcing current which falls below the minimum driving current hence transistor is used to drive buzzer.

Buzzer is a voltage control device and requires minimum 3mA current to drive it.

The gain of BC547 is 800 and the base resistor value chose is 10KΩ so as to obtain 3mA and above current.

The transistor used is NPN transistor and is used as a switch and is connected between ground terminal of buzzer and the circuit ground so as to sink current when acting as closed switch which enables buzzer beep sound. Also the indication led glows.

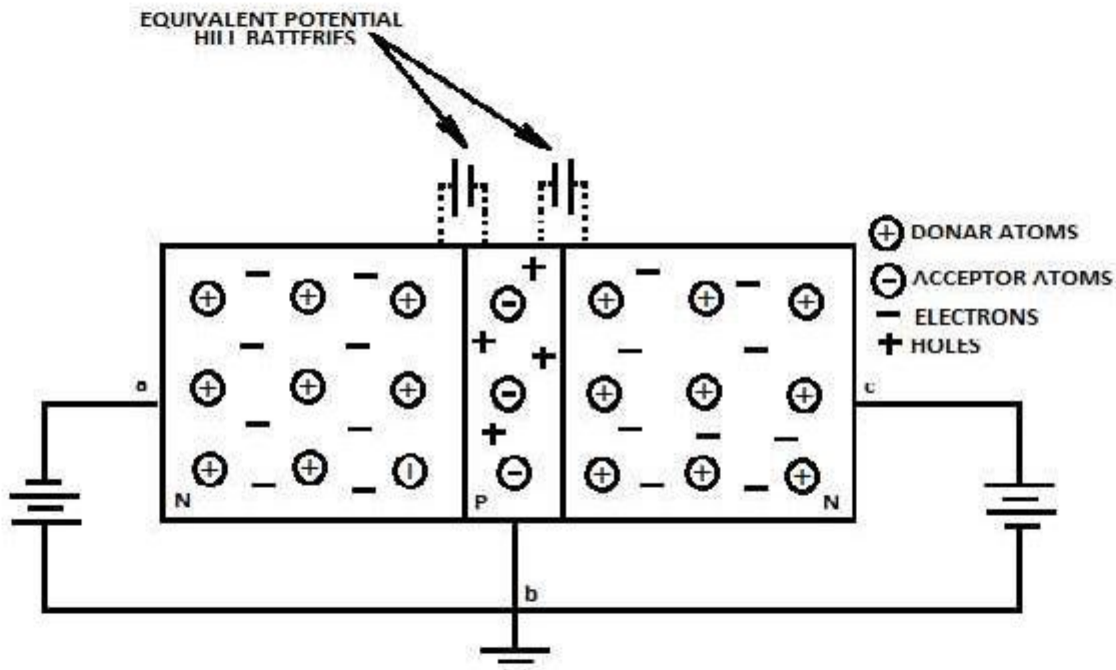
The buzzer and led are ON till the received signal is present.

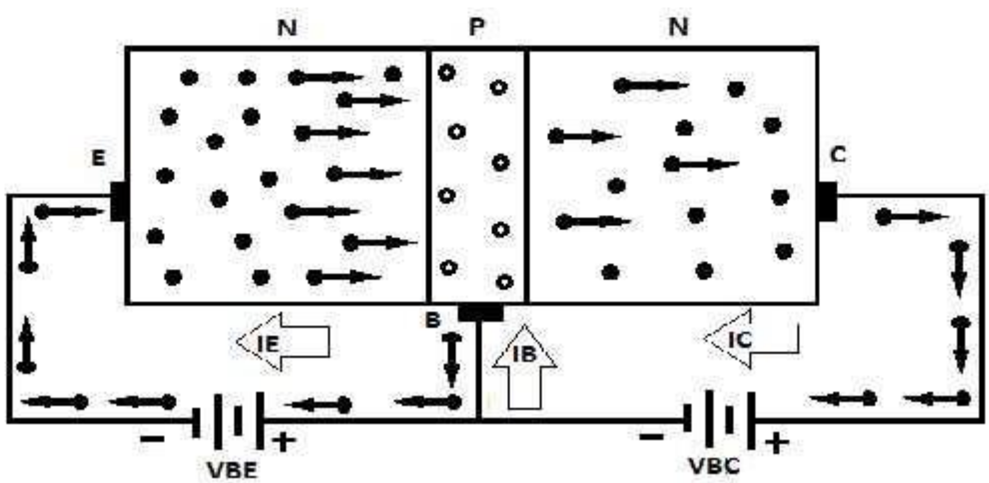
For that to be happen the transistor should be in conducting state that is acting as closed switch or else buzzer is turned OFF immediately. Transistor is turned OFF but led has delay of say 1s, hence fly wheel diode drives it till current reduces to 0 and led is turned OFF so that spikes are not imposed on buzzer which may cause damage.

Hence transistor provides reverse polarity protection.

1KΩ resistor is connected before led so as to limit the current since it can withstand maximum 20mA and hence provides protection to led from any damage.

Basics of NPN transistor





N-p-n transistor is made by sandwiching thin layer of p-type semiconductor between two layers of n-type semiconductor.

It has three terminals - Emitter, Base and collector.

The NPN transistor has two supplies, one is connected through the emitter base and one through the collector base.

The supply is connected such that emitter-base are forward biased and collector base are reverse biased.

It means, base has to be more positive than the emitter and in turn, the collector must be more positive than the base.

The current flow in this type of transistor is carried through movement of electrons.

Emitter emits electrons which are pulled by the base as it is more positive.

This ends up in the collector as it is more positive.

The forward bias causes the electrons in the N-type emitter to flow towards the base. This constitutes the emitter current IE.

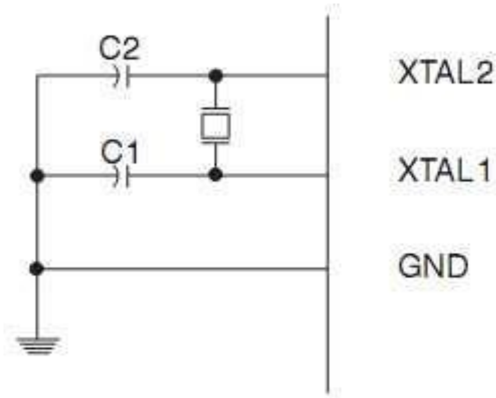
As these electrons flow through the P-type they tend to combine with the holes. As the base is lightly doped and very thin therefore only a few electrons (2%) combine with holes to constitute base current IB.

The remaining electrons (98%) cross over it to the collector region to constitute collector current IC.

In this way almost the entire emitter current flows in the collector circuit. It is clear that emitter current is sum of collector and base current.

IE=IB+IC

2.14 Crystal Oscillator



FEATURES

• Wide range of operating supply voltage: 1.50V to 5.5V

• Regulated voltage drive oscillator circuit for reduced power consumption and crystal drive current

• Optimized low crystal drive current oscillation for miniature crystal units

• PT7C5027Ax series: for Flip Chip Bonding

• PT7C5027Bx series: for Wire Bonding (type I)

• PT7C5027Cx series: for Wire Bonding (type II)

• Recommended oscillation frequency range

• Low frequency version: 10MHz to 60MHz

• Multi-stage frequency divider for low-frequency output support: 0.9MHz

• Built-in frequency divider and 50±5% output duty

• Selectable by version: f0, f0/2, f0/4, f0/8, f0/16, f0/32, f0/64

• - 40 to 85°C operating temperature range

• Standby function

„ High impedance in standby mode, oscillator stops

„ CMOS output duty level (1/2VDD)

• 15pF output drive capability

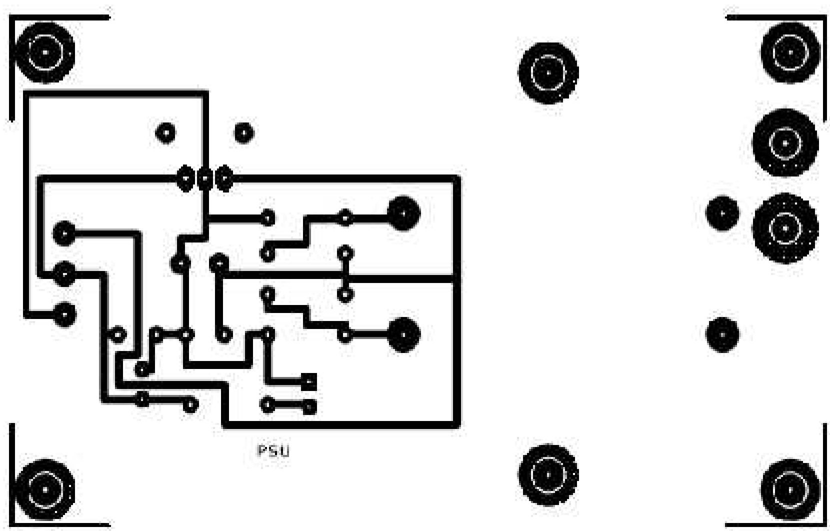
• Die form, Wafer form or 8 pin SOIC package

CHAPTER 3: IMPLEMETATION

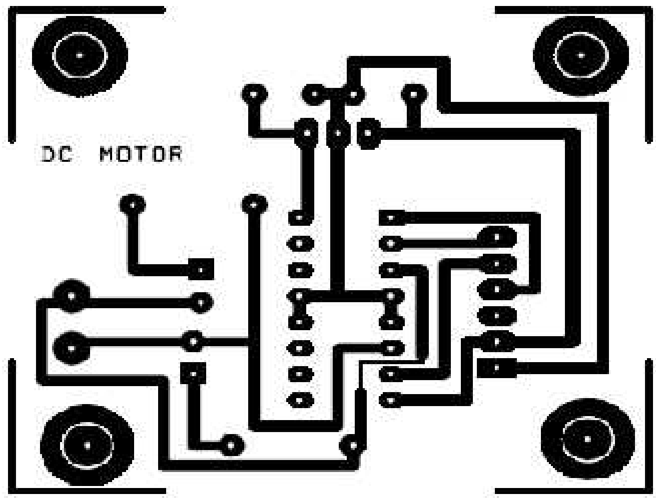
3.1 PCB Layout

Below are the PCB layouts, we had given for etching:

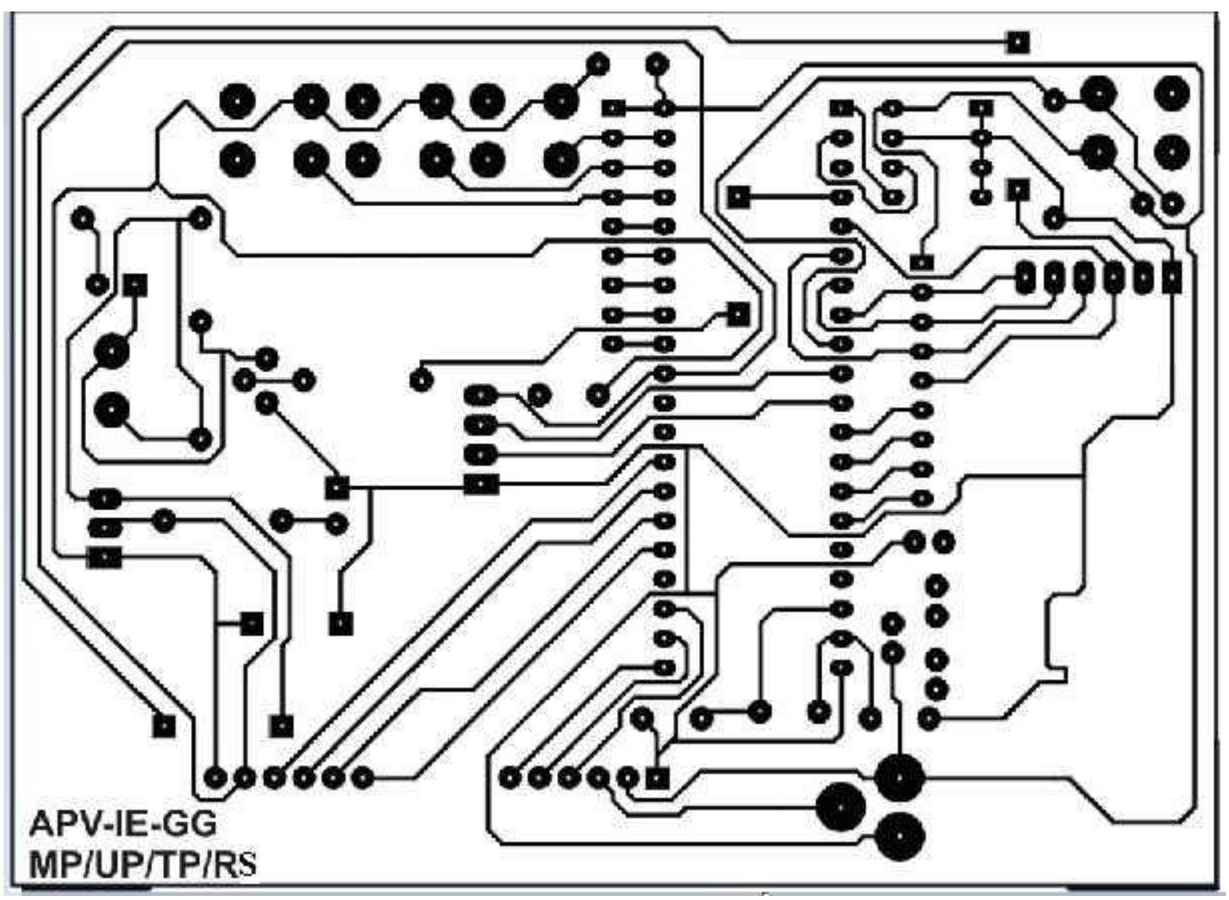
- **Power Supply Layout**



- **Driver Circuit**



- **Main Circuit**



3.2 Software Programming

- **Fingerprint Program**

```
#include <REGX51.H>          // standard 8051 defines
#include <stdio.h> // printf
#include "lcd.h"

sbit BUZZER = P1^3;

sbit ENA = P1^7;
sbit ENB = P1^6;
sbit PWM = P1^5;

// -=-=- Welcome LCD Message -=-=-=
lcdClear();
lcdGotoXY(0,0);         // 1st Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("Finger Print");
lcdGotoXY(0,1);         // 2nd Line of LCD
//        "xxxxxxxxxxxxxxxx"
lcdPrint("bank locker sys.");
delaysec(5);
lcdClear();
lcdGotoXY(0,0);         // 1st Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("AGNEL POLY.");
lcdGotoXY(0,1);         // 2nd Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("YEAR 2014-15");
delaysec(5);
```

```
lcdClear();
lcdGotoXY(0,0);         // 1st Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("Put finger:");
lcdGotoXY(0,1);         // 2nd Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("Search/Add/Empty");
//
EnableInterrupt();

//
}

void main(void)
{
while(1)
{
lcdClear();
lcdGotoXY(0,0);         // 1st Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("Put finger:");
lcdGotoXY(0,1);         // 2nd Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("Search/Add/Empty");
}
switch (NowKey)
{
case ADD_KEY:
lcdGotoXY(0,0);         // 1st Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("ADD...");
putc(100);
```

```
if(fid >= 0)
{
putc(fid);
lcdGotoXY(0,0);          // 2nd Line of LCD
sprintf(buf, "Added ID:" fid);
lcdPrint(buf);
}
NowKey = NO_KEY;
break;

case EMPTY_KEY:
iDelay = SCREEN_TIMEOUT;
lcdClear();
lcdGotoXY(0,0);          // 1st Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("EMPTY...");
putc(110);
break;

case SEARCH_KEY:
lcdClear();
lcdGotoXY(0,0);          // 1st Line of LCD
//      "xxxxxxxxxxxxxxxx"
lcdPrint("SEARCH...");
putc(120);
if(fid >= 0)
{
putc(fid);
lcdGotoXY(0,0);          // 2nd Line of LCD
sprintf(buf, "Found ID:"", fid);
lcdPrint(buf);

dcmotorOnCCW(); // LEFT
```

```
delaysec(2); //delay
dcmotorOff(); // STOP
delaysec(5); //delay
dcmotorOnCW(); // RIGHT
delaysec(2); //delay
dcmotorOff(); // STOP

} else {
// user not found
BUZZER = 1;
delaysec(3);
BUZZER = 0;
}
break;
}
}
}
```

➢ **Biometric Sensing Codes**

```
/***********************************************
command code & Respond code for
Addfingerprint
** ** **** ******* ******* ** ***************/
byte addFingerPrintO[] = {Ox4D,Ox58, OxlO, Ox03, Ox40, OxOO, OxOO, OxF8};
II8bytes byte addFingerPrintl[] = {Ox4D,Ox58, OxlO, Ox03, Ox40, OxOO, OxOl, OxF9};
II8bytes byte addFingerPrint2[] = {Ox4D,Ox58, OxlO, Ox03, Ox40, OxOO, Ox02, OxFA};
II 8bytes byte addFingerPrint3[] = {Ox4D,Ox58, OxlO, Ox03, Ox40, OxOO, Ox03, OxFB};
```

```
II 8 bytes byte addFingerPrint4[] = {Ox4D, Ox58, OxlO, Ox03, Ox40, OxOO, Ox04,
OxFC};
II8bytes byte operationSuccessAdd[] = {Ox4D,Ox58,Ox30,Ox02,Ox40,Ox31,Ox48};
II7bytes byte timeOutAdd[] = {Ox4D,Ox58, Ox30, Ox02, Ox40, Ox33, Ox4A};
II7bytes byteprocessFailureAdd[] = {Ox4D,Ox58,Ox30, Ox02, Ox40, Ox34, Ox4B};
11 7bytes byte parameterErrorAdd[] = {Ox4D,Ox58, Ox30, Ox02, Ox40, Ox35, Ox4C};


/** ********************************************
Command Code & Respond Code For Delete Fingerprint
** * ******* ******* *******

****** ******* ***/
II7bytes bytedeleteFingerPrintO[] = {Ox4D,Ox58,OxlO,Ox03,Ox42,OxOO,OxOO,OxFA};
II8bytes bytedeleteFingerPrintl[] = {Ox4D,Ox58,OxlO,Ox03,Ox42,OxOO,OxOl,OxFB};
II8bytes bytedeleteFingerPrint2[] = {Ox4D,Ox58,OxlO,Ox03,Ox42,OxOO,Ox02,OxFC};
II8bytes bytedeleteFingerPrint3[] = {Ox4D,Ox58,OxlO,Ox03,Ox42,OxOO,Ox03,OxFD};
II8bytes bytedeleteFingr<Print4[] = {Ox4D,Ox58,OxlO,Ox03,Ox42,OxOO,Ox04,OxFE};
II8bytes byte operationSuccessDelete[] = {Ox4D, Ox58, Ox30, Ox02,
Ox42, Ox31, Ox4A};
// 7 bytes byteparameterErrorDelete []= {Ox4D,Ox58,Ox30,Ox02,Ox42,Ox35,Ox4E};


/*****************************************************************
Add Fingerprint Function
** ** * ** ** * ** ** * ** ** * ** ** * ** ** * ** ** * ** *I
void addFingerO(){
   II writethe8bytesarrayofthecommandcode
   Serial.write (addFinqerPrintO, size of( addFinqer PrintO ));
   delay(lOOO);
 }
```

```
void addFingerl(){
   II writethe8bytesarrayofthecommandcode
   Serial.wtite (addFinqerPrintl, sizeof(addFinqerPrintl));
   delay(lOOO);
 }


void addFinger2(){
   II writethe8bytesarrayofthecommandcode
   Serial.wtite (addFinqerPrint2, sizeof(addFinqerPrint2));
   delay(lOOO);
 }


void addFinger3(){
   II writethe8bytesarrayofthecommandcode
   Serial.wtite (addFinqerPrint3, sizeof(addFinqerPrint3));
   delay(lOOO);
 }


Operation Function
 ** ** * ** ** * ** ** * ** ** * ** ** * ** ** * ** ** * ** *I
voidoperationSuccesful() {
   II  Now,replycontainsall7bytes
   fo<(Byte x=O; x<7; x++)
   {
      while(Serial.available()==0){};II Waitforabyte
      operation[x] =Serial.read();
```

```
// Compare 7bytes of received respond code with the add fingerprint operation of the fingerprint reader
b= memcmp(operationSuccessAdd, operation,
sizeof(operationSuccessAdd));
c = memcmp(processFailureAdd, operation,
sizeof(processFailureAdd));
d = memcmp(parameterErrorAdd, operation,
sizeof(parameterErrorAdd));
e = memcmp(timeOutAdd, operation,
sizeof(timeOutAdd));

// Compare 7bytes of received respond code with the delete fingerprint operation of the fingerprint reader
f = memcmp(operationSuccessDelete, operation,
sizeof(operationSuccessDelete));
g = memcmp(parameterErrorDelete, operation,
sizeof(parameterErrorDelete));

// Compare 7bytes of received respond code with the search fingerprint operation of the fingerprint reader
h= memcmp(operationSuccessSearch, operation,
sizeof(operationSuccessSearch));
i= memcmp(parameterErrorSearch, operation,
sizeof(parameterErrorSearch));
g = memcmp(processFailureSearch, operation,
sizeof(processFailureSearch));
k = memcmp(timeOutSearch, operation,
sizeof(timeOutSearch));
l = memcmp(noMatchFinger, operation,
sizeof(noMatchFinger));

// Compare 7 bytes of received respond code with the empty database operation of the fingerprint
reader
m = memcmp(operationSuccessEmpty, operation, sizeof(operationSuccessEmpty));
```

```
// Compare 7bytes of received respond code with the search database operation of the fingerprint reader
n = memcmp(operationSuccessDatabase, operation, sizeof(operationSuccessDatabase));
o = memcmp(operationFailureDatabase, operation, sizeof(operationFailureDatabase));
p = memcmp(notinRange, operation, sizeof(notinRange));
```

Biometric Sensor used in our Circuit to sense the fingerprints

➢ Header Files

89C51.H

```
/*****************************************************************************************************
**********************************************************************/
//------------....------------....--------------------------....--------------------------....-----------
#ifndef __89C51_H__
#define __89C51_H__
//------------....------------....--------------------------....--------------------------....-----------
//Byte Registers
sfr P0      = 0x80;
sfr SP      =   0x81;
sfr DPL     = 0x82;
sfr DPH     = 0x83;
sfr PCON    = 0x87;
sfr TCON    = 0x88;
sfr TMOD    = 0x89;
sfr TL0     = 0x8A;
sfr TL1     = 0x8B;
sfr TH0     = 0x8C;
sfr TH1     = 0x8D;
sfr P1      = 0x90;
sfr  SCON  =  0x98;
sfr  SBUF  =  0x99;
sfr P2 = 0xA0;
sfr IE      = 0xA8;
sfr P3      = 0xB0;
sfr IP      = 0xB8;
sfr PSW     = 0xD0;
sfr ACC     = 0xE0;
sfr B       = 0xF0;

//P0 Bit Registers
sbit P0_0 = 0x80;
sbit P0_1 = 0x81;
sbit P0_2 = 0x82;
sbit P0_3 = 0x83;
sbit P0_4 = 0x84;
sbit P0_5 = 0x85;
sbit P0_6 = 0x86;
sbit P0_7 = 0x87;

//P1 Bit Registers
sbit P1_0 = 0x90;
```

```
sbit P1_1 = 0x91;
sbit P1_2 = 0x92;
sbit P1_3 = 0x93;
sbit P1_4 = 0x94;
sbit P1_5 = 0x95;
sbit P1_6 = 0x96;
sbit P1_7 = 0x97;

//P2 Bit Registers
sbit P2_0 = 0xA0;
sbit P2_1 = 0xA1;
sbit P2_2 = 0xA2;
sbit P2_3 = 0xA3;
sbit P2_4 = 0xA4;
sbit P2_5 = 0xA5;
sbit P2_6 = 0xA6;
sbit P2_7 = 0xA7;

//P3 Bit Registers
sbit P3_0 = 0xB0;
sbit P3_1 = 0xB1;
sbit P3_2 = 0xB2;
sbit P3_3 = 0xB3;
sbit P3_4 = 0xB4;
sbit P3_5 = 0xB5;
sbit P3_6 = 0xB6;
sbit P3_7 = 0xB7;

//ACC Bit Registers
sbit ACC_0      = 0xE0;
sbit ACC_1 = 0xE1;
sbit ACC_2 = 0xE2;
sbit ACC_3 = 0xE3;
sbit ACC_4 = 0xE4;
sbit ACC_5 = 0xE5;
sbit ACC_6 = 0xE6;
sbit ACC_7  = 0xE7;

//B Bit Registers
sbit B_0 = 0xF0;
sbit B_1 = 0xF1;
sbit B_2 = 0xF2;
sbit B_3 = 0xF3;
sbit B_4 = 0xF4;
sbit B_5 = 0xF5;
sbit B_6 = 0xF6;
sbit B_7 = 0xF7;

//TCON Bit Registers
```

```
sbit  IT0  =  0x88;
sbit  IE0  =  0x89;
sbit  IT1  =  0x8A;
sbit  IE1  =  0x8B;
sbit  TR0  =  0x8C;
sbit  TF0  =  0x8D;
sbit  TR1  =  0x8E;
sbit TF1 = 0x8F;

//SCON Bit Registers
sbit RI = 0x98;
sbit TI   = 0x99;
sbit RB8 = 0x9A;
sbit TB8 = 0x9B;
sbit REN = 0x9C;
sbit SM2 = 0x9D;
sbit SM1 = 0x9E;
sbit SM0  = 0x9F;

//IE Bit Registers
sbit EX0 = 0xA8;
sbit ET0 = 0xA9;
sbit EX1 = 0xAA;
sbit ET1 = 0xAB;
sbit ES = 0xAC;
sbit ET2 = 0xAD;
sbit EA = 0xAF;

#endif
```

Delay.H

```
/*************************************************************
***********************************
//*************************************************************
**********************************/
#ifndef __DELAY_H__
#define  __DELAY_H__

#pragma SAVE
#pragma REGPARMS
extern void DelayL(unsigned long Delay);//Delay for long time
extern void DelayI(unsigned int Delay);//Delay for small time
extern void DelayU(unsigned int Delay);//Delay for specific microseconds
extern void DelayM(unsigned int Delay);//Delay for specific milliseconds
#pragma RESTORE

#endif
```

LCD.H

```
/*****************************************************************
***********************************


*****************************************************************
**********************************/
//------------....------------....--------------------------....--------------------------....-----
------
#ifndef __LCD4_H__
#define __LCD4_H__

sbit    LCD_CTRL_E        = LCD_DATA^0;
sbit    LCD_CTRL_RW       = LCD_DATA^1;
sbit    LCD_CTRL_RS       = LCD_DATA^2;
sbit    LCD_BUSY      = LCD_DATA^7;
sfr LCD_DATA = LCD_PORT; //P0, P1, P2, P3


#pragma SAVE
#pragma REGPARMS
void lcdGotoXY(unsigned char row, unsigned char col);
void lcdInit(void);

// moves the cursor/position to Home (upper left corner)
void lcdHome(void);

// clears the LCD display
void lcdClear(void);

// prints a series of bytes/characters to the display

extern void SetLCD(void);           //initiates LCD
extern void LCD(unsigned char L);//0-clear display, 1-Line 1, 2-Line 2
extern char putchar(char);          //prints single character on LCD
void lcdClear(void);

// moves the cursor/position to the row,col requested
// ** this may not be accurate for all displays
void lcdGotoXY(unsigned char row, unsigned char col);

// prints a series of bytes/characters to the display
void lcdPrint(char* sdata);

// print with length
void lcdPrintData(char* sdata, unsigned char nBytes);
#pragma RESTORE

#endif
```

Serial.H

```
/************************************************************
**********************************
************************************************************
**********************************/
#ifndef __SERIAL_H__
#define __SERIAL_H__

#pragma SAVE
#pragma REGPARMS
extern void SetSerial(unsigned long BaudRate,unsigned long XTAL,unsigned
char ClkDiv);//intitialize serial port,baudrate,xtal frequency,clock divider
extern void SetSerial115200(void);//set fixed baudrate for phlips
extern void Send(unsigned char c);//transmits one character
extern void SendString(unsigned char *String,unsigned char Len);//transmit
complete string
extern unsigned char Read(void);//reads one character & waits till it comes
extern unsigned char ReadD(unsigned int Delay);//reads one character &
waits for specified time only
#pragma RESTORE

#endif
```

3.3 Burning the Microcontroller

Keil Debugger:

We used Embedded C for programming the P89S51 microcontroller.

- µ Vision IDE from Keil combines project management; make facilities, source code editing, program debugging, and complete simulation in one powerful environment.
- Development platform is easy-to-use and it helps you quickly create embedded programs that work.
- µ Vision editor and debugger are integrated in a single application that provides a seamless embedded project development environment.
- The Keil µ Vision Debugger accurately simulates on-chip peripherals (UART, SPI, Interrupts, I/O Ports, A/D Converter, D/A Converter, and PWM Modules) of 8051 devices.
- Simulation helps us understand hardware configurations and avoids time wasted on setup problems.
- Additionally, with simulation, we can write and test applications before target hardware is available.

Hardware for development and debugging

After a long search on internet we came across a very easy and handy programmer to burn 89S51 microcontroller in ISP mode. The diagram given below gives the hardware look and the software screenshot for the specified program.

A multipurpose general board meant for programming and debugging was used for microcontroller debugging.

CHAPTER 4: TESTING AND TROUBLESHOOTING

4.1 Continuity Test

In electronics, a continuity test is the checking of an electric circuit to see if current flows (that it is in fact a complete circuit).

A continuity test is performed by placing a small voltage (wired in series with an LED or noise-producing component such as a piezoelectric speaker) across the chosen path.

If electron flow is inhibited by broken conductors, damaged components, or excessive resistance, the circuit is "open".

Devices that can be used to perform continuity tests include multi-meters which measure current and specialized continuity testers which are cheaper, more basic devices, generally with a simple light bulb that lights up when current flows.

An important application is the continuity test of a bundle of wires so as to find the two ends belonging to a particular one of these wires; there will be a negligible resistance between the "right" ends, and only between the "right" ends.

This test is performed just after the hardware soldering and configuration has been completed.

This test aims at finding any electrical open paths in the circuit after the soldering.

Many a times, the electrical continuity in the circuit is lost due to improper soldering, wrong and rough handling of the PCB, improper usage of the soldering iron, component failures and presence of bugs in the circuit diagram.

We use a multi meter to perform this test.

We keep the multi meter in buzzer mode and connect the ground terminal of the multi meter to the ground.

We connect both the terminals across the path that needs to be checked. If there is continuation then you will hear the beep sound.

4.2 Power ON Test

This test is performed to check whether the voltage at different terminals is according to the requirement or not.

We take a multi meter and put it in voltage mode.

Remember that this test is performed without microcontroller.

Firstly, we check the output of the transformer, whether we get the required 12 V AC voltage. While testing once we detected our transformer had failed as output was coming as 0V and hum sound (transformer inductance working sound) was not produced nor did the transformer was dissipating heat. Hence we had replaced the transformer.

Then we apply this voltage to the power supply circuit. Note that we do this test without microcontroller because if there is any excessive voltage, this may lead to damaging the controller.

We check for the input to the voltage regulator i.e., are we getting an input of 12V and an output of 5V. This 5V output is given to the microcontrollers' 40^{th}Pin. Hence we check for the voltage level at 40^{th} pin.

Similarly, we check for the other terminals for the required voltage. In this way we can assure that the voltage at all the terminals is as per the requirement.

4.3 Steps involved in Troubleshooting

Switch off the power supply.

Check for any defects, this can include open circuits, lack of continuity (gap in etched copper track) etc.

Next test the continuity of various tracks in the circuit. Make sure there are no open circuits.

Make sure that the IC pins which are supposed to be connected are well shorted.

Check for non-functioning IC.

Check all the resistors, capacitors. etc.

4.4 IC failure

Due to overheating in the circuit the IC may blow up. In order to test whether there is any fault in the IC remove the IC from the circuit and check for its utility using an IC tester.

Another way for checking the IC is its substitution with a similar IC.

In order to avoid the troubleshooting or re-soldering an entire IC it is advisable to solder an IC bed or IC socket to the circuit rather than the IC.

In such case if IC is found faulty it can be easily unplugged and new IC can be connected.

4.5 IC Testing Procedure

Enter the IC number using keyboard.

Press enter key.

IC number is displayed on display.

Press test.

Then it will show pass/fail of IC.

CHAPTER 5: CONCLUSION

This project made us familiar with the vast applications of **Fingerprint Based Security System.** We have applied our theoretical knowledge and have simultaneously gained practical experience, thus satisfying the basic objective of the "project".

With this project we have realized the importance of planning and organization of the work involved. Assigning different members of our project team and yet ensuring co-ordination amongst all has enabled us to complete the project in time.

We have been very fortunate to get the appropriate guidance, as and when required from our project guide and we once again thank them for the same.

APPENDICES

DATASHEETS

AT89S52

AT89S52

Table 24-1. Serial Programming Instruction Set

Instruction	Instruction Format Byte 1	Byte 2	Byte 3	Byte 4	Operation
Programming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx 0110 1001 (Output on MISO)	Enable Serial Programming while RST is high
Chip Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip Erase Flash memory array
Read Program Memory (Byte Mode)	0010 0000	xxx A12 A11 A10 A9 A8	A7 A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Read data from Program memory in the byte mode
Write Program Memory (Byte Mode)	0100 0000	xxx A12 A11 A10 A9 A8	A7 A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Write data to Program memory in the byte mode
Write Lock Bits[1]	1010 1100	1110 00 B1 B2	xxxx xxxx	xxxx xxxx	Write Lock bits. See Note (1).
Read Lock Bits	0010 0100	xxxx xxxx	xxxx xxxx	xxx LB3 LB2 LB1 xx	Read back current status of the lock bits (a programmed lock bit reads back as a "1")
Read Signature Bytes	0010 1000	xxx A12 A11 A10 A9 A8	A7 xxx xxx0	Signature Byte	Read Signature Byte
Read Program Memory (Page Mode)	0011 0000	xxx A12 A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Read data from Program memory in the Page Mode (256 bytes)
Write Program Memory (Page Mode)	0101 0000	xxx A12 A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Write data to Program memory in the Page Mode (256 bytes)

Note: 1. B1 = 0, B2 = 0 ---> Mode 1, no lock protection
B1 = 0, B2 = 1 ---> Mode 2, lock bit 1 activated
B1 = 1, B2 = 0 ---> Mode 3, lock bit 2 activated
B1 = 1, B2 = 1 ---> Mode 4, lock bit 3 activated

$\Big\}$ Each of the lock bit modes needs to be activated sequentially before Mode 4 can be executed.

After Reset signal is high, SCK should be low for at least 64 system clocks before it goes high to clock in the enable data bytes. No pulsing of Reset signal is necessary. SCK should be no faster than 1/16 of the system clock at XTAL1.

For Page Read/Write, the data always starts from byte 0 to 255. After the command byte and upper address byte are latched, each byte thereafter is treated as data until all 256 bytes are shifted in/out. Then the next instruction will be ready to be decoded.

27

1919D-MICRO-6/08

26. Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.6V
DC Output Current	15.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

27. DC Characteristics

The values shown in this table are valid for T_A = -40°C to 85°C and V_{CC} = 4.0V to 5.5V, unless otherwise noted.

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low Voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC}-0.1	V
V_{IL1}	Input Low Voltage ($\overline{EA}$)		-0.5	0.2 V_{CC}-0.3	V
V_{IH}	Input High Voltage	(Except XTAL1, RST)	0.2 V_{CC}+0.9	V_{CC}+0.5	V
V_{IH1}	Input High Voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC}+0.5	V
V_{OL}	Output Low Voltage[1] (Ports 1,2,3)	I_{OL} = 1.6 mA		0.45	V
V_{OL1}	Output Low Voltage[1] (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.45	V
V_{OH}	Output High Voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
V_{OH1}	Output High Voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ± 10%		-300	µA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pulldown Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode[1]	V_{CC} = 5.5V		50	µA

Notes: 1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
Maximum I_{OL} per port pin: 10 mA
Maximum I_{OL} per 8-bit port:
Port 0: 26 mA Ports 1, 2, 3: 15 mA
Maximum total I_{OL} for all output pins: 71 mA
If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
2. Minimum V_{CC} for Power-down is 2V.

LM7805

Block Diagram

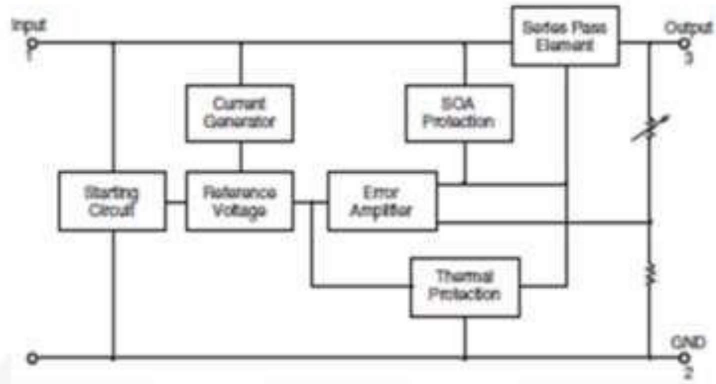


Figure 1. Block Diagram

Absolute Maximum Ratings

Stresses exceeding the absolute maximum ratings may damage the device. The device may not function or be operable above the recommended operating conditions and stressing the parts to these levels is not recommended. In addition, extended exposure to stresses above the recommended operating conditions may affect device reliability. The absolute maximum ratings are stress ratings only. Values are at $T_A = 25°C$ unless otherwise noted.

Symbol	Parameter		Value	Unit
V_I	Input Voltage	$V_O = 5$ V to 18 V	35	V
		$V_O = 24$ V	40	
$R_{θJC}$	Thermal Resistance, Junction-Case (TO-220)		5	°C/W
$R_{θJA}$	Thermal Resistance, Junction-Air (TO-220)		65	°C/W
T_{OPR}	Operating Temperature Range	LM78xx	-40 to +125	°C
		LM78xxA	0 to +125	
T_{STG}	Storage Temperature Range		- 65 to +150	°C

Electrical Characteristics (LM7805)

Refer to the test circuit, -40°C < T_J < 125°C, I_O = 500 mA, V_I = 10 V, C_I = 0.1 µF, unless otherwise specified.

Symbol	Parameter	Conditions		Min.	Typ.	Max.	Unit
V_O	Output Voltage	T_J = +25°C		4.80	5.00	5.20	V
		I_O = 5 mA to 1 A, $P_O \le$ 15 W, V_I = 7 V to 20 V		4.75	5.00	5.25	
Regline	Line Regulation[2]	T_J = +25°C	V_I = 7 V to 25 V		4.0	100.0	mV
			V_I = 8 V to 12 V		1.6	50.0	
Regload	Load Regulation[2]	T_J = +25°C	I_O = 5 mA to 1.5 A		9.0	100.0	mV
			I_O = 250 mA to 750 mA		4.0	50.0	
I_Q	Quiescent Current	T_J = +25°C			5	8	mA
ΔI_Q	Quiescent Current Change	I_O = 5 mA to 1 A			0.03	0.50	mA
		V_I = 7 V to 25 V			0.30	1.30	
$\Delta V_O/\Delta T$	Output Voltage Drift[3]	I_O = 5 mA			-0.8		mV/°C
V_N	Output Noise Voltage	f = 10 Hz to 100 kHz, T_A = +25°C			42		µV
RR	Ripple Rejection[3]	f = 120 Hz, V_I = 8 V to 18 V		62	73		dB
V_{DROP}	Dropout Voltage	T_J = +25°C, I_O = 1 A			2		V
R_O	Output Resistance[3]	f = 1 kHz			15		mΩ
I_{SC}	Short-Circuit Current	T_J = +25°C, V_I = 35 V			230		mA
I_{PK}	Peak Current[3]	T_J = +25°C			2.2		A

Notes:

2. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.
3. These parameters, although guaranteed, are not 100% tested in production.

BC547

FAIRCHILD
SEMICONDUCTOR®

BC546/547/548/549/550

Switching and Applications

- High Voltage: BC546, V_{CEO}=65V
- Low Noise: BC549, BC550
- Complement to BC556 ... BC560

1. Collector 2. Base 3. Emitter

546/547/548/549/550

NPN Epitaxial Silicon Transistor

Absolute Maximum Ratings T_a=25°C unless otherwise noted

Symbol	Parameter	Value	Units
V_{CBO}	Collector-Base Voltage : BC546 : BC547/550 : BC548/549	80 50 30	V V V
V_{CEO}	Collector-Emitter Voltage : BC546 : BC547/550 : BC548/549	65 45 30	V V V
V_{EBO}	Emitter-Base Voltage : BC546/547 : BC548/549/550	6 5	V V
I_C	Collector Current (DC)	100	mA
P_C	Collector Power Dissipation	500	mW
T_J	Junction Temperature	150	°C
T_{STG}	Storage Temperature	-65 ~ 150	°C

Electrical Characteristics T_a=25°C unless otherwise noted

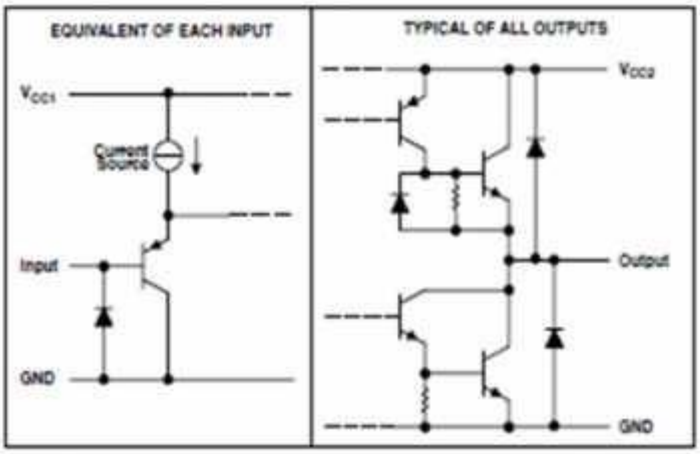
Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Units
I_{CBO}	Collector Cut-off Current	V_{CB}=30V, I_E=0			15	nA
h_{FE}	DC Current Gain	V_{CE}=5V, I_C=2mA	110		800	
V_{CE} (sat)	Collector-Emitter Saturation Voltage	I_C=10mA, I_B=0.5mA I_C=100mA, I_B=5mA		90 200	250 600	mV mV
V_{BE} (sat)	Base-Emitter Saturation Voltage	I_C=10mA, I_B=0.5mA I_C=100mA, I_B=5mA		700 900		mV mV
V_{BE} (on)	Base-Emitter On Voltage	V_{CE}=5V, I_C=2mA V_{CE}=5V, I_C=10mA	580	660	700 720	mV mV
f_T	Current Gain Bandwidth Product	V_{CE}=5V, I_C=10mA, f=100MHz		300		MHz
C_{ob}	Output Capacitance	V_{CB}=10V, I_E=0, f=1MHz		3.5	6	pF
C_{ib}	Input Capacitance	V_{EB}=0.5V, I_C=0, f=1MHz		9		pF
NF	Noise Figure : BC546/547/548 : BC549/550 : BC549 : BC550	V_{CE}=5V, I_C=200μA f=1KHz, R_G=2KΩ V_{CE}=5V, I_C=200μA R_G=2KΩ, f=30~15000MHz		2 1.2 1.4 1.4	10 4 4 3	dB dB dB dB

h_{FE} Classification

Classification	A	B	C
h_{FE}	110 ~ 220	200 ~ 450	420 ~ 800

L293D

schematics of inputs and outputs (L293D)



absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage, V_{CC1} (see Note 1)	36 V
Output supply voltage, V_{CC2}	36 V
Input voltage, V_I	7 V
Output voltage range, V_O	−3 V to V_{CC2} + 3 V
Peak output current, I_O (nonrepetitive, $t \le 5$ ms): L293	±2 A
Peak output current, I_O (nonrepetitive, $t \le 100$ µs): L293D	±1.2 A
Continuous output current, I_O: L293	±1 A
Continuous output current, I_O: L293D	±600 mA
Package thermal impedance, θ_{JA} (see Notes 2 and 3): DWP package	TBD°C/W
N package	67°C/W
NE package	TBD°C/W
Maximum junction temperature, T_J	150°C
Storage temperature range, T_{stg}	−65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. All voltage values are with respect to the network ground terminal.
2. Maximum power dissipation is a function of T_J(max), θ_{JA}, and T_A. The maximum allowable power dissipation at any allowable ambient temperature is $P_D = (T_J(max) - T_A)/\theta_{JA}$. Operating at the absolute maximum T_J of 150°C can affect reliability.
3. The package thermal impedance is calculated in accordance with JESD 51-7.

recommended operating conditions

			MIN	MAX	UNIT
	Supply voltage	V_{CC1}	4.5	7	V
		V_{CC2}	V_{CC1}	36	
V_{IH}	High-level input voltage	$V_{CC1} \le 7$ V	2.3	V_{CC1}	V
		$V_{CC1} \ge 7$ V	2.3	7	V
V_{IL}	Low-level output voltage		–0.3†	1.5	V
T_A	Operating free-air temperature		0	70	°C

† The algebraic convention, in which the least positive (most negative) designated minimum, is used in this data sheet for logic voltage levels.

electrical characteristics, V_{CC1} = 5 V, V_{CC2} = 24 V, T_A = 25°C

PARAMETER			TEST CONDITIONS		MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage		L293: I_{OH} = –1 A L293D: I_{OH} = –0.6 A		V_{CC2} – 1.8	V_{CC2} – 1.4		V
V_{OL}	Low-level output voltage		L293: I_{OL} = 1 A L293D: I_{OL} = 0.6 A			1.2	1.8	V
V_{OKH}	High-level output clamp voltage		L293D: I_{OK} = –0.6 A			V_{CC2} + 1.3		V
V_{OKL}	Low-level output clamp voltage		L293D: I_{OK} = 0.6 A			1.3		V
I_{IH}	High-level input current	A	V_I = 7 V			0.2	100	µA
		EN				0.2	10	
I_{IL}	Low-level input current	A	V_I = 0			–3	–10	µA
		EN				–2	–100	
I_{CC1}	Logic supply current		I_O = 0	All outputs at high level		13	22	mA
				All outputs at low level		35	60	
				All outputs at high impedance		8	24	
I_{CC2}	Output supply current		I_O = 0	All outputs at high level		14	24	mA
				All outputs at low level		2	6	
				All outputs at high impedance		2	4	

switching characteristics, V_{CC1} = 5 V, V_{CC2} = 24 V, T_A = 25°C

PARAMETER		TEST CONDITIONS	L293NE, L293DNE			UNIT
			MIN	TYP	MAX	
t_{PLH}	Propagation delay time, low-to-high-level output from A input	C_L = 30 pF, See Figure 1		800		ns
t_{PHL}	Propagation delay time, high-to-low-level output from A input			400		ns
t_{TLH}	Transition time, low-to-high-level output			300		ns
t_{THL}	Transition time, high-to-low-level output			300		ns

switching characteristics, V_{CC1} = 5 V, V_{CC2} = 24 V, T_A = 25°C

PARAMETER		TEST CONDITIONS	L293DWP, L293N L293DN			UNIT
			MIN	TYP	MAX	
t_{PLH}	Propagation delay time, low-to-high-level output from A input	C_L = 30 pF, See Figure 1		750		ns
t_{PHL}	Propagation delay time, high-to-low-level output from A input			200		ns
t_{TLH}	Transition time, low-to-high-level output			100		ns
t_{THL}	Transition time, high-to-low-level output			350		ns

SM630

3.2 Electrical Interface

Module is connected to HOST via 4PIN cable. The PIN definition is as follows:

No.	PIN Definition	Remarks
1	Power supply +	Power supply +
2	Module Tx	Open-drain output, need to use pull-up resistance in application (Typical value: 10KΩ)
3	Module Rx	Wide voltage input, 7V affordable
4	Power supply	Power supply —

Notes:

The PIN close to the edge of circuit board is PIN4: Power supply —.

Chapter 4 Communication Protocol

4.1 Command

No.	Name of Command	Command Code
1	Add fingerprint	0x40
2	Delete fingerprint	0x42
3	Search fingerprint	0x44
4	Empty fingerprint database	0x46
5	Search information in fingerprint database	0x4B
6	Download fingerprint template	0x50
7	Upload fingerprint template	0x52
8	Read ID number	0x60
9	Read user Flash	0x62
10	Write user Flash	0x64
11	Read product logo	0x80

4.2 Response Code

No.	Name of Command	Response Code
1	Receive correct	0x01
2	Receive error	0x02
3	Operation successful	0x31
4	Finger detected	0x32
5	Time out	0x33
6	Fingerprint process failure	0x34
7	Parameter error	0x35
8	Fingerprint matching with this ID found	0x37
9	No matching fingerprint with this ID	0x38
10	Fingerprint found	0x39
11	Fingerprint unfound	0x3A

4.3 Coding Method

The communication between HOST and Module must be coded as Communication Packet.

One communication packet includes the following:

Packet Head (2 bytes)

Packet flag (1 byte)

Packet length (1 byte)

Packet Content（N bytes）

Check sum（1 byte）

Packet head：0x4D 0x58

Packet flag：

0x10：command packet

0x20：data packet

0x21：last packet

0x30：response packet

Packet length：

Length of the Content in packet

Packet content：

Content of packet

Check sum：

Low 8 bytes of the SUM from packet head to check sum.

4.4 Brief Work Flowchart

Module waits for command from HOST after it is powered on. Module will respond by a Rx correct packet after receiving the correct command. Module will perform operations according to the command and will return corresponding information after the operation is successful. When the Module is performing operation, it will not respond to other command given by HOST. If the check sum for the received command is wrong, the module will send back receive error response.

Module receive correct packet:

0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

Module receive error packet:

0x4D + 0x58 + 0x30 + 0x01 + 0x02 + 0xD8

Chapter 5 Command Description

1 Add fingerprint

Description: Add fingerprint at the designated position

Length: 3 bytes

Format: Command code 0x40 + high byte of the to-be-added fingerprint ID + low byte of the to-be-added fingerprint ID

Flowchart:

After module receives the command to add fingerprint, it goes to the status of adding fingerprint. The flowchart is as follows:

For example:

1 HOST send command to add fingerprint at position 0:

0x4D + 0x58 + 0x10 + 0x03 + 0x40 + 0x00 + 0x00 + 0xF8

2 Module responds by receive correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 First time to press finger. Module will respond as operation successful after processing the first fingerprint:

0x4D + 0x58 + 0x30 + 0x02 + 0x40 + 0x31 + 0x48

4 Press finger again, and module will respond as operation successful after processing:

0x4D + 0x58 + 0x30 + 0x02 + 0x40 + 0x31 + 0x48

Remarks:

1 Fingerprint ID starts from 0

2 Fingerprint storage capacity: 768

3 If the ID is wrong in the command, module will responds as parameter error:

0x4D + 0x58 + 0x30 + 0x02 + 0x40 + 0x35 + 0x4C

4 If user press different finger at the first time and second time, or the fingerprint quality is poor, module will responds as fingerprint processing failure:

0x4D + 0x58 + 0x30 + 0x02 + 0x40 + 0x34 + 0x4B

5 If there is no finger pressing within 10 seconds, module will respond as time-out.

0x4D + 0x58 + 0x30 + 0x02 + 0x40 + 0x33 + 0x4A

2 Delete Fingerprint

Description: Delete the fingerprint of designated ID

Length: 3 bytes

Format: Command code 0x42 + high bytes of the to-be-deleted fingerprint ID + low bytes of the to-be-deleted fingerprint ID

For example:

1 HOST send command to delete fingerprint ID No. 0:

0x4D + 0x58 + 0x10 + 0x03 + 0x42 + 0x00 + 0x00 + 0xFA

2 Module responds as RX correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 Module responds as operation successful after execute the fingerprint deletion command:

0x4D + 0x58 + 0x30 + 0x02 + 0x42 + 0x31 + 0x4A

Remarks:

1 If fingerprint ID in the command is out of range, module will respond as parameter error:

0x4D + 0x58 + 0x30 + 0x02 + 0x42 + 0x35 + 0x4E

3 Search Fingerprint

Description: Searching for designated fingerprint within range

Length: 5 bytes

Format: Command code 0x44 + search high bytes of starting ID + search low bytes of low bytes + high bytes of the number of fingerprints searched + low bytes of the number of fingerprints searched

Flowchart:

After receiving the command to search fingerprint, module will get ready to status of searching fingerprint. See below flowchart:

For example:

1 HOST send command to search 16 fingerprints starting from 0:

0x4D + 0x58 + 0x10 + 0x05 + 0x44 + 0x00 + 0x00 + 0x00 + 0x10 + 0x0E

2 When the fingerprint is placed on the sensor window, module will respond as operation successful:

0x4D + 0x58 + 0x30 + 0x02 + 0x44 + 0x31 + 0x4C

3 If the fingerprint is found, module will return the following:

0x4D + 0x58 + 0x30 + 0x04 + 0x44+ 0x39 + high bytes of ID for the found fingerprint + low bytes of ID for the found fingerprint + check sum

4 If no matching fingerprint is found, module will return the following:

0x4D + 0x58 + 0x30 + 0x02 + 0x44 + 0x3A + 0x55

Remarks:

1 The number of the fingerprints that are searched starts from the ID of the first fingerprint, for example, the search starts from fingerprint ID 0, the number of fingerprints searched is 0x10, then the fingerprint ID actually being searched is 0～0x0F, altogether 0x10 fingerprints.

2 If the ID in the command is wrong, module will responds as parameter error:

0x4D + 0x58 + 0x30 + 0x02 + 0x44 + 0x35 + 0x50

3If the fingerprint quality is poor, module will respond as fingerprint processing failure:

0x4D + 0x58 + 0x30 + 0x02 + 0x44 + 0x34 + 0x4F

4 If there is no finger placing on the sensor with 10 seconds, module will respond as time out:

0x4D + 0x58 + 0x30 + 0x02 + 0x44 + 0x33 + 0x4E

4 Empty Fingerprint Database

Description: Empty all fingerprints in fingerprint database

Length: 1 byte

Format: Command code 0x46

For example:

1 HOST send command to empty fingerprint database:

0x4D + 0x58 + 0x10 + 0x01 + 0x46 + 0xFC

2 Module will respond as Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 Module will respond as operation successful after executing command to empty fingerprint database:

0x4D + 0x58 + 0x30 + 0x02 + 0x46 + 0x31 + 0x4E

5 Search Fingerprint Database Information

Description: Search and see if there is fingerprint matching the designated ID

Length: 3 bytes

Format: Command code 0x4B + high byte of the to-be-searched fingerprint ID + low byte of the to-be-searched fingerprint ID

For example:

1 HOST send command to search fingerprint with ID 0:

0x4D + 0x58 + 0x10 + 0x03 + 0x4B + 0x00 + 0x00 + 0x03

2 Module will respond as Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 If there is fingerprint with ID 0, module will return the following:

0x4D + 0x58 + 0x30 + 0x02 + 0x4B + 0x37 + 0x59

4 If there is no fingerprint with ID 0, module will then return the following:

0x4D + 0x58 + 0x30 + 0x02 + 0x4B + 0x38 + 0x5A

Remarks:

1 If the fingerprint ID in the command is out of range, module will respond as parameter error:

0x4D + 0x58 + 0x30 + 0x02 + 0x4B + 0x35 + 0x57

6 Download Fingerprint Template

Description: Download a fingerprint into the module

Length: 3 bytes

Format: Command code 0x50 + high byte of the to-be-downloaded fingerprint ID + low byte of the to-be-downloaded fingerprint ID

For example:

1 HOST send a command to download a fingerprint to the position for ID 0:

0x4D + 0x58 + 0x10 + 0x03 + 0x50 + 0x00 + 0x00 + 0x08

2 Module will respond as Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 HOST send the first data packet (packet content 128):

0x4D + 0x58 + 0x20 + 0x80 + ……128 bytes of data…… + check sum

4 Module will respond as Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

5 HOST send the second data packet (packet content 128):

0x4D + 0x58 + 0x21 + 0x80 + ……128 bytes of data…… + check sum

6 Module will respond as Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

7 Module will respond as operation successful: 0x4D + 0x58 + 0x30 + 0x02 + 0x50 + 0x31 + 0x58

Remarks:

1 If the fingerprint ID in the command is wrong, module will respond as parameter error:

0x4D + 0x58 + 0x30 + 0x02 + 0x50 + 0x35 + 0x5C

7 Upload Fingerprint Template

Description: Upload fingerprint template with the designated ID

Length: 3 bytes

Format: Command code 0x52 + high byte of the to-be-uploaded fingerprint ID + low byte of the to-be-uploaded fingerprint ID.

For example:

1 HOST send a command to upload a fingerprint to the position for ID 0:

0x4D + 0x58 + 0x10 + 0x03 + 0x52 + 0x00 + 0x00 + 0x0A

2 Module will respond as Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 Module send the first data packet

0x4D + 0x58 + 0x20 + 0x80 + ...128 bytes of data...... + check sum

4 HOST will respond as Rx correct 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

5 Module send the second data packet

0x4D + 0x58 + 0x21 + 0x80 + ...128 bytes of data...... + check sum

Remarks:

1 If the fingerprint ID in the command is wrong, module will respond as parameter error:0x4D + 0x58 + 0x30 + 0x02 + 0x52 + 0x35 + 0x5E

8 Read ID Number

Description: Read module ID number

Length: 1 byte

Format: Command code 0x60

Example:

1 HOST send a command to read Module ID number:

0x4D + 0x58 + 0x10 + 0x01 + 0x60 + 0x16

2 Module will respond as Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 Module will respond by 24-byte ID number after executing the command:

0x4D + 0x58 + 0x30 + 0x19 + 0x60 + ...24-byte ID number...... + check sum

Remarks:

1 ID number is set by manufacturer. User can read ID number only. Each module has its own ID number. User can tell different module by reading ID number.

9 Read User Flash

Description: Read the content of the designated address in user flash in the module

Length: 4 bytes

Format: Command code 0x62 + read high bytes of the address + read low bytes of the address + read the number

For example:

1 HOST send command to read 10 data starting from ADD 0 in user flash:

0x4D + 0x58 + 0x10 + 0x04 + 0x62 + 0x00 + 0x00 + 0x0A + 0x25

2 Module will respond as Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 Module will respond by the data it read:

0x4D + 0x58 + 0x30 + number of data read + 0x62 + ...data read...... + check sum

Remarks:

1 The memory is 64K bytes in user flash (ADD 1~0xFFFF). Maximum 128Byte data can be read at one time.

10 Write User Flash

Description: Write data in the designated address in user Flash

Length: N+4 bytes

Format: Command code 0x64 + high bytes of the address where data to be written + low bytes of the address where data to be written + number of data to be written + ...N bytes of data to be written......

For example:

1 HOST send a command to write 2 Byte data in to the ADD 0 in user Flash

0x4D + 0x58 + 0x10 + 0x06 + 0x64 + 0x00 + 0x00 + 0x02 + 0x00 + 0x00 + 0x21

2 Module will respond as Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 Module will respond after executing the command:

0x4D + 0x58 + 0x30 + 0x02 + 0x64 + 0x31 + 0x6C

Remarks:

1 The memory is 64K bytes in user flash. Maximum 128Byte data can be written at one time.

2 Please do not let address go beyond limit. It will cause unpredictable consequences if there is data outflow.

3 Please do not power off when writing in Flash. Powering off will cause unpredictable consequences.

11 Read Product Flag

Description: Read product flag

Length: 1 byte

Format: Command code 0x80

For example:

1 HOST send command to read product flag

0x4D + 0x58 + 0x10 + 0x01 + 0x80 + 0x36

2 Module will respond by Rx correct: 0x4D + 0x58 + 0x30 + 0x01 + 0x01 + 0xD7

3 Module will respond after executing command:

0x4D + 0x58 + 0x30 + 0x19 + 0x80 + ...24-byte product flag...... + check sum

Remarks:

1 Product flag is defined as 24Byte ASCII code, including 8-Byte manufacturer flag, 8-Byte product flag and 8-byte version flag.

PART LIST

		MICROCONTROLLER CIRCUIT	
11	2	CAPACITOR 104nF	3/-
12	5	CAPACITOR 10µF	10/-
13	2	CAPACITOR 33pF	3/-
14	2	LED	2/-
15	1	DIODE 1N4007	5/-
16	1	PRESET 50K	5/-
17	1	IC BC547	5/-
18	2	10K R-ARRAY	7/-
19	1	470 R RESISTOR	1/-
20	2	RESISTOR 10K	1/-
21	1	RESISTOR 1K	1/-
22	4	SWITCHES PRESS BUTTONS	24/-
23	1	LCD (16x2)	200/-
24	1	AT89S52 IC	60/-
25	1	CRYSTAL 11.0592 MHz	15/-
26	1	SENSOR	3600/-
27	3	6 PIN RELIMATE MALE CONNECTOR	36/-
28	1	3 PIN RELIMATE MALE CONNERTOR	8/-
29	1	4 PIN RELIMATE MALE CONNECTOR	12/-
30	1	IC MOUNTING STRIP	7/-
31	1	BUZZER	38/-
32		CONNECTING WIRES	15/-

		DRIVER CIRCUIT	
33	1	DC MOTOR (600mA & 6V-12V)	150/-
34	1	MOTOR DRIVE L293D	50/-
35	1	IC MOUNTING STRIP	4.50/-
36	2	CAPACITOR 104nF	3/-
37	2	LED	2/-
38	1	IC 7805	10/-

		PCB MAKING & ETCHING COST	
39	1	POWER SUPPLY CIRCUIT	200/-
40	1	MICROCONTROLLER CIRCUIT	800/-
41	1	DRIVER CIRCUIT	150/-

REFERENCES

REFERENCE BOOKS:

[1] Signals, Systems and Computers, 2004 Conference Record of the Thirty-Eighth Asilomar Conference on Publication 7-Nov-2004 Volume: 1, on page(s): 577-581 Vol.1.

[2] International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 10, October 2012.

[3] International Journals of Biometric and Bio-informatics , Volume (3) : Issue (1).

[4] R. A. Fisher Biometrics, Vol. 20, No. 2, In Memoriam: Ronald Aylmer Fisher, 1890-1962 (Jun., 1964), pp. 261-264.

[5] John Wharton: An Introduction to the Intel MCS-51TM Single-Chip Microcomputer Family, Application Note AP-69, May 1980.

Standard Books **Microcontroller** By V. Udayashankara

REFERENCE SITES:

[1] **http://tutorial.cytron.com.my/2013/01/29/interfacing-fingerprint-reader-integrated-sm630-with-arduino-uno/1-48/**

* Sensor Programming, Biometric and Error Checking.

[2] robokitsworld.com/**datasheets**/**SM-630**.**pdf**

* Sensor Datasheet pdf.

[3] www.projectsof**8051**.com/**fingerprint**-based-security-system/

* Understanding the overall circuit

[4] www.youtube.com/watch?v=WzAWPzLxQyc

* Connections and error checking

[5] www.engineersgarage.com/**microcontroller**/**8051**projects

*Troubleshooting and testing.

[6] www.engineersgarage.com/tutorials/how-to-program-a-**microcontroller**

* Programming Microcontroller

[7] www.powerguru.org/**driver-circuits**/

* Driver Circuit.

[8] www.expresspcb.com/

* PCB Designing.

[9] http://www.circuitstoday.com/5v-power-supply-using-7805

*for power supply section

[10] http://www.locxtronic.com/basiccircuitforpic16f877a.html

*Basic connections for pic like oscillator, power supply and reset