BANK LOCKER SECURITY SYSYTEM

A

MAJOR PROJECT REPORT

Submitted in partial fulfillment of the requirements
for the degree of

BACHELOR OF ENGINEERING
in
ELECTRONICS AND COMMUNICATION ENGINEERING
BY

ANIMESH JAIN	0105EC101017
ANKIT CHITRANSH	0105EC101018
ARPIT JAIN	0105EC101025
CHITRARATH BHARGAVA	0105EC101035
SHUBHAM MAWAR	0105EC101107

MAY 2014

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
ORIENTAL INSTITUTE OF SCIENCE AND TECHNOLOGY
BHOPAL (M.P)

An ISO 9001:2008 Certified Institution
Approved by AICTE, New Delhi
Affiliated to Rajiv Gandhi Technical University Bhopal (M.P.)

CHAPTER 1. INTRODUCTION

INTRODUCTION:

We keep our money in the bank for safety, so it is equally important for the bank to save itself from theft. In this era the number of larceny is increasing so it is very important to prevent banks from burglar. This circuit can prevent bank from thieves. MSG is automatically sent when any thief open your bank locker.

Also, with the help of GPS module tracking of the locker is also possible. This theft intimation system using GSM/GPS module can also be used in personal lockers and other commercial lockers.

This Project focuses onto implement GSM (Global System for Mobile Communication) based Banking Security System. This system is implemented using an embedded microcontroller. The embedded microcontroller used here is ATMEGA8L.

Actually, the aim of the project is to implement an Automatic Banking Security System. Security is the protection of something valuable to ensure that it is not stolen, lost, or altered. GSM Based bank safety locker security system provides more reliability and restricts the unauthorized person who is trying to enter into bank.

Primarily, the system functions with the help of different technologies like the Global Positioning System (GPS), traditional cellular network such as Global System for Mobile Communications (GSM) and other radio frequency medium. Today GSM fitted Banks, cars; ambulances, fleets and police vehicles are common sights on the roads of developed countries. GSM based bank safety locker security system is simple and costs less. When a GSM based bank safety locker security system is installed in a Bank & to enter the unauthorized person means the message will be transferred to a predefined number.

The functional units of our projects are:

GSM Module
GPS Module
DC Motor
LCD Display
Micro Controller ATMEGA8L

1.1 GSM MODULE

The GSM module consist of Wireless CPU, SIM card holder and power LED. It helps to transmit and receive the SMS with UART.

1.2 GPS MODULE:

The GPS Module gives the information of the latitude and longitude of the location of the locker.

1.3 DC MOTOR:

It works on the principle of electromagnetic induction. By switching the current ON or OFF in a coil its magnetic field can be varied and its armature can be rotated.

1.4 LIQUID CRYSTAL DISPLAY:

A liquid crystal display (LCD) is a flat panel display, electronic visual display, video display that uses the light modulating properties of liquid crystals (LCs). LCs do not emit light directly.

1.5 MICRO CONTROLLER ATmega8L:

The ATmega8 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8 achieves throughputs approaching 1 MIPS per MHz, allowing the system designer, to optimize power consumption versus processing speed.

CHAPTER 2. WORKING

WORKING:

The main aim of this project is to provide efficient way of security for banks. Which is an advanced door lock system used in the fields where security and secrecy is the primary constraint. The main objective is to design digital code lock which is used to reduce manual interference to the maximum extent.

GSM technology can provide a sophisticated theft alert system for bank locker system. The embedded I/O unit automates the inner door and entry door. The inner door always kept open. There are two modes in this project one is normal mode and another one is security mode. In normal mode an authorized person can open the locker key and he can close the entry door. At that time GSM never send the message to the required person. If any person tries to open the locker key in security mode, the inner door will be closed automatically and SMS is transferred to the required person's hand phone. After identifying the theft an authorized person can automate the inner door through the SMS.

The GSM module is connected with the microcontroller through serial port. Using 'AT' commands the SMS is transferred to the GSM module. The GSM module converts the digital information into airborne signals. Through GSM network the SMS is transferred to the required person's cell phone. This system offers better solution for the Bank security system and also it will help you to track the intruder.

The working could be explained as follows:

1. CASE-1: The person enters the key and opens the locker- The moment the locker is opened, an SMS saying "LOCKER OPENED" is sent to his cell phone with the help of GSM module together with GPS module depicting the latitude and longitude of the location of the locker.

2. CASE 2: If an unauthorized person tries to open the locker and enters a wrong password then an alert is sent to the administrator's cell phone.

CHAPTER 3. LIST OF COMPONENTS

LIST OF COMPONENTS

Component name	Description	Quantity
1. IC	ATMEGA8L	1
2. DIODE	1N4007	2
3. CAPACITOR	4700ufd	3
4. RESISTOR	1K	2
5. GSM MODULE	900MHz	1
6. GPS MODULE	1500MHz	1
7. MAX232	16PIN-IC,5C	2
8. MOTOR	12V	1
9. LCD	16PIN	1
10. BUZZER	5V	1
11. BATTERY	9V	1
12. LED	5V	1
13. SWITCH	Two Way	1

Table 2.1

CHAPTER 4. COMPONENT DESCRIPTION

COMPONENT DESCRIPTION:

4.1 TRANSFORMER

4.1.1 DEFINITION:

The transformer is a static electro-magnetic device that transforms one alternating voltage (current) into another voltage (current). However, power remains the some during the transformation. Transformers play a major role in the transmission and distribution of ac power.

4.1.2 PRINCIPLE:

Transformer works on the principle of mutual induction. A transformer consists of laminated magnetic core forming the magnetic frame. Primary and secondary coils are wound upon the two cores of the magnetic frame, linked by the common magnetic flux. When an alternating voltage is applied across the primary coil, a current flows in the primary coil producing magnetic flux in the transformer core. This flux induces voltage in secondary coil.

Transformers are classified as: -

(a) Based on position of the windings with respect to core i.e.

  (1) Core type transformer

  (2) Shell type transformer

(b) Transformation ratio:

  (1) Step up transformer

  (2) Step down transformer

(a) Core & shell types: Transformer is simplest electrical machine, which consists of windings on the laminated magnetic core. There are two possibilities of putting up the windings on the core.

1. Winding encircle the core in the case of core type transformer

2. Cores encircle the windings on shell type transformer

(b) Step up and Step down: In these, voltage transformation takes place according to whether the primary is high voltage coil or a low voltage coil.

  (1) Lower to higher-> Step up
  (2) Higher to lower-> Step down

4.2 MOTOR DRIVER IC (L293D):

L293D is a dual H-bridge motor driver integrated circuit (IC). Motor drivers act as current amplifiers since they take a low-current control signal and provide a higher-current signal. This higher current signal is used to drive the motors.

L293D contains two inbuilt H-bridge driver circuits. In its common mode of operation, two DC motors can be driven simultaneously, both in forward and reverse direction. The motor operations of two motors can be controlled by input logic at pins 2 & 7 and 10 & 15. Input logic 00 or 11 will stop the corresponding motor. Logic 01 and 10 will rotate it in clockwise and anticlockwise directions, respectively.

Enable pins 1 and 9 (corresponding to the two motors) must be high for motors to start operating. When an enable input is high, the associated driver gets enabled. As a result, the outputs become active and work in phase with their inputs. Similarly, when the enable input is low, that driver is disabled, and their outputs are off and in the high-impedance state.

The L293 and L293D are quadruple high-current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, dc and bipolar stepping motors, as well as other high-current/high-voltage loads in positive-supply applications.

4.2.1PIN DIAGRAM:

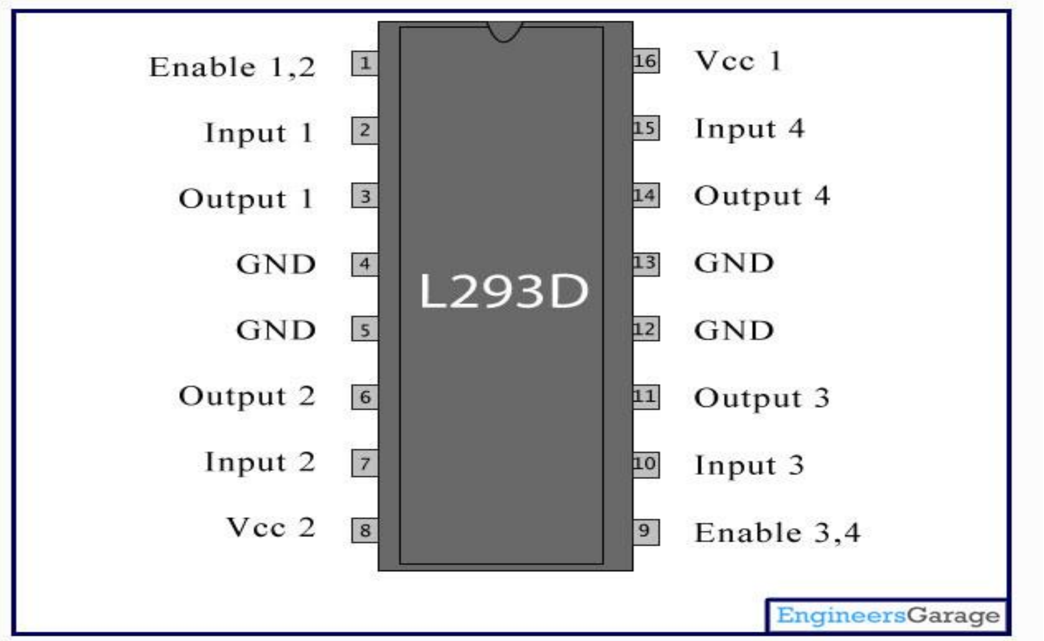


Figure 4.1

4.2.2 PIN DESCRIPTION:

Pin No.	**Function**	**Name**
1	Enable pin for Motor 1; active high	Enable 1,2
2	Input 1 for Motor 1	Input 1
3	Output 1 for Motor 1	Output 1
4	Ground (0V)	Ground
5	Ground (0V)	Ground
6	Output 2 for Motor 1	Output 2
7	Input 2 for Motor 1	Input 2
8	Supply voltage for Motors; 9-12V (up to 36V)	Vcc_2
9	Enable pin for Motor 2; active high	Enable 3,4
10	Input 1 for Motor 1	Input 3
11	Output 1 for Motor 1	Output 3
12	Ground (0V)	Ground
13	Ground (0V)	Ground
14	Output 2 for Motor 1	Output 4
15	Input2 for Motor 1	Input 4
16	Supply voltage; 5V (up to 36V)	Vcc_1

Table 2.1

4.3 DB9 CONNECTOR:

The term "DB9" refers to a common connector type, one of the D-Subminiature or D-Sub types of connectors. DB9 has the smallest "footprint" of the D-Subminiature connectors, and houses 9 pins (for the male connector) or 9 holes (for the female connector).

DB9 connectors were once very common on PCs and servers. DB9 connectors are designed to work with the EIA/TIA 232 serial interface standard, which determined the function of all nine pins as a standard, so that multiple companies could design them into their products. DB9 connectors were commonly used for serial peripheral devices like keyboards, mice, joysticks, etc. Also they are used on DB9 cable assemblies for data connectivity.

Today, the DB9 has mostly been replaced by more modern interfaces such as USB, PS/2, Firewire, and others. However, there are still many legacy devices that use the DB9 interface for serial communication.

Figure 4.2

4.3.1 APPLICATIONS:

1) COMMUNICATION PORTS:

The widest application of D-subs is for RS-232 serial communications, though the standard did not make this connector mandatory. RS-232 devices originally used the DB25, but for many applications the less common signals were omitted, allowing a DE9 to be used. The standard specifies a male connector for terminal equipment and a female connector for modems, but many variations exist. IBM PC-compatible computers tend to have male connectors at the device and female connectors at the modems. Early Apple Macintosh models used DE9 connectors for RS-422 serial interfaces (which can operate as RS-232). Later Macintosh models use 8-pin miniature DIN connectors instead.

On PCs, 25-pin and (beginning with the IBM-PC/AT) 9-pin plugs are used for the RS-232 serial ports; 25-pin sockets are used for the parallel printer ports (instead of the Centronics socket found on the printer itself which was inconveniently large for direct placement on the expansion cards).

Many uninterruptible power supply units have a DE9F connector on them in order to signal to the attached computer via an RS-232 interface. Often these do not send data serially to the computer but instead use the handshaking control lines to indicate low battery, power failure, or other conditions. Such usage is not standardized between manufacturers and may require special cables.

2) NETWORK PORTS

DE9 connectors were used for some token ring networks as well as other computer networks.

The Attachment Unit Interfaces that were used with 10BASE5 "thick net" in the 1980s and 1990s used DA15 connectors for connectivity between the Medium Attachment Units and (Ethernet) network interface cards, albeit with a sliding latch to lock the connectors together instead of the usual hex studs with threaded holes. The sliding latch was intended to be quicker to engage and disengage and to work in places where jack screws could not be used for reasons of component shape.

DE-9 connectors are commonly used in CAN: female connectors are on the bus while male connectors are on devices.

Computer video output

Figure 4.3

A female 9-pin connector on an IBM compatible personal computer may be a video display output such as MDA, Hercules, CGA, or EGA (rarely VGA or others). Even though these all use the same DE9 connector, the displays cannot all be interchanged and monitors or video interfaces may even be damaged if connected to an incompatible device using the same connector.

Later analog video (VGA and later) adapters generally replaced these connectors with DE15 high-density sockets (though some early VGA devices still used DE9 connectors). DE15 connectors are similar to DE9 connectors (see above).

Many Apple Macintosh models (beginning with the Macintosh II) used DA15 sockets for analogue RGB video out. Just prior to this, the Apple IIgs used the same connector for the same purpose, but in a non-compatible way. A digital (and thus also incompatible) RGB adapter for the Apple IIe also used a DA15F. And the Apple IIc used a DA15F for an auxiliary video port which was not RGB, but provided the necessary signals to derive RGB.

3) GAME CONTROLLER PORTS:

Starting in the late 1970s the Atari 2600 game console used modified DE9 connectors (male on the system, female on the controller) for its game controller connectors. The Atari connectors had bodies entirely of molded plastic without the metal shield, and they omitted the pair of fastening screws. In the years following, various video game consoles and home computers adopted the same connector for their own game ports, though they were not all interoperable. The most common wiring supported five digital connections (for up, down, left, and right movement, and one fire button), plus one pair of analog 100 kΩ potentiometers or paddles. Some computers supported additional buttons, and on some computers additional devices, such as a computer mouse, a light pen, and/or a graphics tablet were also supported via the game port. Unlike the basic one-button digital joysticks and the basic paddles, such devices were not typically interchangeable between different systems.

Systems utilizing the DE9 connector for their game port included the Atari 8-bit and ST lines; the Commodore VIC-20, 64, 128, and Amiga; the Amstrad CPC (which employed daisy-chaining when connecting two Amstrad-specific joysticks); the MSX, Sharp X68000, and FM-Towns, predominantly used in Japan; the Sega Master System and Sega Mega Drive/Genesis; and the Panasonic 3DO. The Sinclair ZX Spectrum lacked a built-in joystick connector of any kind but aftermarket interfaces provided the ability to connect DE9 joysticks.

Many Apple II computers also used DE9 connectors for joysticks, but they had a female port on the computer and a male on the controller, used analog rather than digital sticks, and the pin-out was completely unlike that used on the aforementioned systems. DE9 connectors were not used for game ports on the Apple Macintosh, Apple III, IBM PC systems, or most newer game consoles.

DA15S connectors are used for PC joystick connectors, where each DA15 connector supports two joysticks each with two analog axes and two buttons. In other words, one DA15S "game adapter" connector has 4 analog potentiometer inputs and 4 digital switch inputs. This interface is strictly input-only, though it does provide +5 V DC power. Some joysticks with more than two axes and/or more than two buttons use the signals designated for both joysticks. Conversely, Y-adapter cables are available that allow two separate joysticks to be connected to a single DA15 game adapter port; if a joystick connected to one of these Y-adapters has more than two axes or buttons, only the first two of each will work.

The IBM DA15 PC game connector has been modified to add a (usually MPU-401 compatible) MIDI interface, and this is often implemented in the game connectors on third-party sound cards, for example the Sound Blaster line from Creative Labs. The "standard" straight game adapter connector (introduced by IBM) has three ground pins and four +5 V power pins, and the MIDI adaptation replaces one of the grounds and one of the +5 V pins, both on the bottom row of pins, with MIDI In and MIDI Out signal pins. (There is no MIDI Thru provided.) Creative Labs introduced this adaptation.

The Neo Geo (system) AES game console also used the DA15 connector, however the pins are wired directly to the controller switches and is therefore not compatible with the regular DA15 PC game controllers.

4.4 BUZZER:

A buzzer or beeper is an audio signalling 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.

Figure 4.4

A piezoelectric element may be driven by an oscillating electronic circuit or other audio signal source, driven with a piezoelectric audio amplifier. Sounds commonly used to indicate that a button has been pressed are a click, a ring or a beep.

Uses:

4) Annunciator panels
5) Electronic metronomes
6) Game shows
7) Microwave ovens and other household appliances
8) Sporting events such as basketball games
9) Electrical alarms

4.5 POWER SUPPLY:

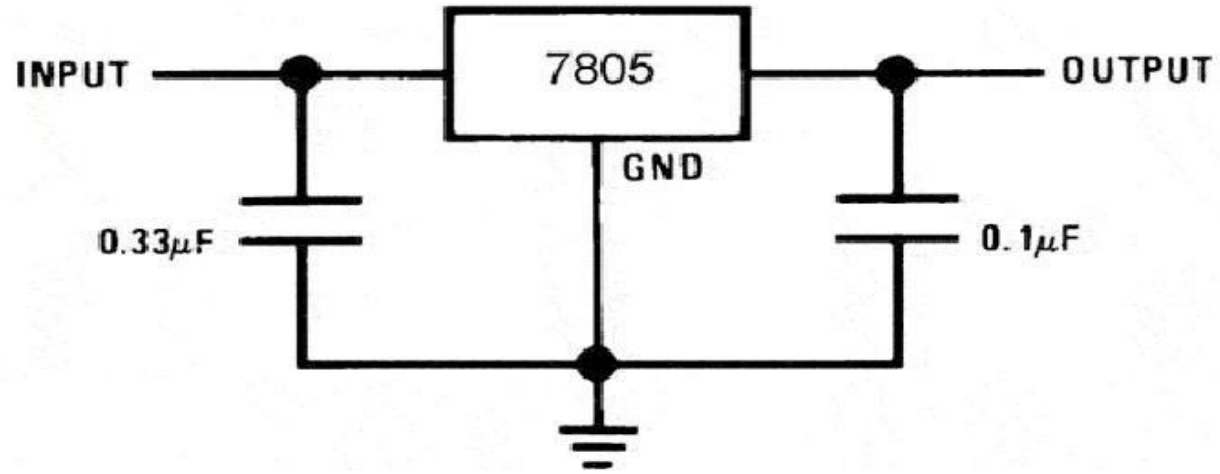


Figure 4.5 Circuit Diagram

This circuit is a small +5V power supply. The circuit will provide a regulated voltage to the external circuit which may also I am required in any part of the external circuit or the whole external circuit. The best part is that you can also use it to convert AC voltage to DC and then regulate it ,simply You need a transformer to make the AC main drop down to a safe value i.e. 12-15 volts and then us a rectifier to convert AC into DC.

This circuit can give +5V output at about 150 mA current, but it can be increased to 1 A when good cooling is added to 7805 regulator chip. The circuit has over overload and terminal protection. The capacitors must have enough high voltage rating to safely handle the input voltage feed to circuit. The circuit is very easy to build for example into a piece of veroboard.

If you need other voltages than +5V, you can modify the circuit by replacing the 7805 chips with another regulator with different output voltage from regulator 78xx chip family. The last numbers in the chip code tells the output voltage. Remember that the input voltage must be at least 3V greater than regulator output voltage ot otherwise the regulator does not work well.Dont forget to check the pin diagram before connecting the IC.

4.6 MICROCONTROLLER

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

4.6.1 USES OF MICROCONTROLLERS

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

4.6.2 ABOUT AVR ATMEGA 8L

The ATmega8 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8 achieves throughputs approaching 1 MIPS per MHz, allowing the system designer to optimize power consumption versus processing speed.

I/O and Packages:

23 Programmable I/O Lines, 28-lead PDIP

Operating Voltages:

2.7V - 5.5V (ATmega8L)

4.5V - 5.5V (ATmega8)

Power Consumption:

At 4 MHz, 3V, 25°C:

Active: 3.6mA

Idle Mode: 1.0Ma Power-down Mode: 0.5µA

Features:

1. High-performance, Low-power AVR® 8-bit Microcontroller
2. Advanced RISC Architecture
   - 130 Powerful Instructions – Most Single-clock Cycle Execution
   - 32 x 8 General Purpose Working Registers
   - Fully Static Operation
   - Up to 16 MIPS Throughput at 16 MHz
   - On-chip 2-cycle Multiplier
3. High Endurance Non-volatile Memory segments
   - 8K Bytes of In-System Self-programmable Flash program memory
   - 512 Bytes EEPROM
   - 1K Byte Internal SRAM
   - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
   - Data retention: 20 years at 85°C/100 years at 25°C [(1)]
   - Optional Boot Code Section with Independent Lock Bits In-System Programming by On-chip Boot Program
4. True Read-While-Write Operation
   - Programming Lock for Software Security
5. Peripheral Features
   - Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
   - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
   - Real Time Counter with Separate Oscillator
   - Three PWM Channels
   - 8-channel ADC in TQFP and QFN/MLF package
   - Eight Channels 10-bit Accuracy
   - 6-channel ADC in PDIP package Six Channels 10-bit Accuracy
   - Byte-oriented Two-wire Serial Interface
   - Programmable Serial USART
   - Master/Slave SPI Serial Interface
   - Programmable Watchdog Timer with Separate On-chip Oscillator
   - On-chip Analog Comparator
6. Special Microcontroller Features
   - Power-on Reset and Programmable Brown-out Detection
   - Internal Calibrated RC Oscillator
   - External and Internal Interrupt Sources
   - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
7. I/O and Packages
   - 23 Programmable I/O Lines
   - 28-lead PDIP, 32-lead TQFP, and 32-pad QFN/MLF
8. Operating Voltages
   - 2.7 - 5.5V (ATmega8L)
   - 4.5 - 5.5V (ATmega8)
9. Speed Grades
   - 0 - 8 MHz (ATmega8L)
   - 0 - 16 MHz (ATmega8)
10. Power Consumption at 4 MHz, 3V, 25°C

4.6.3 PIN DIAGRAM

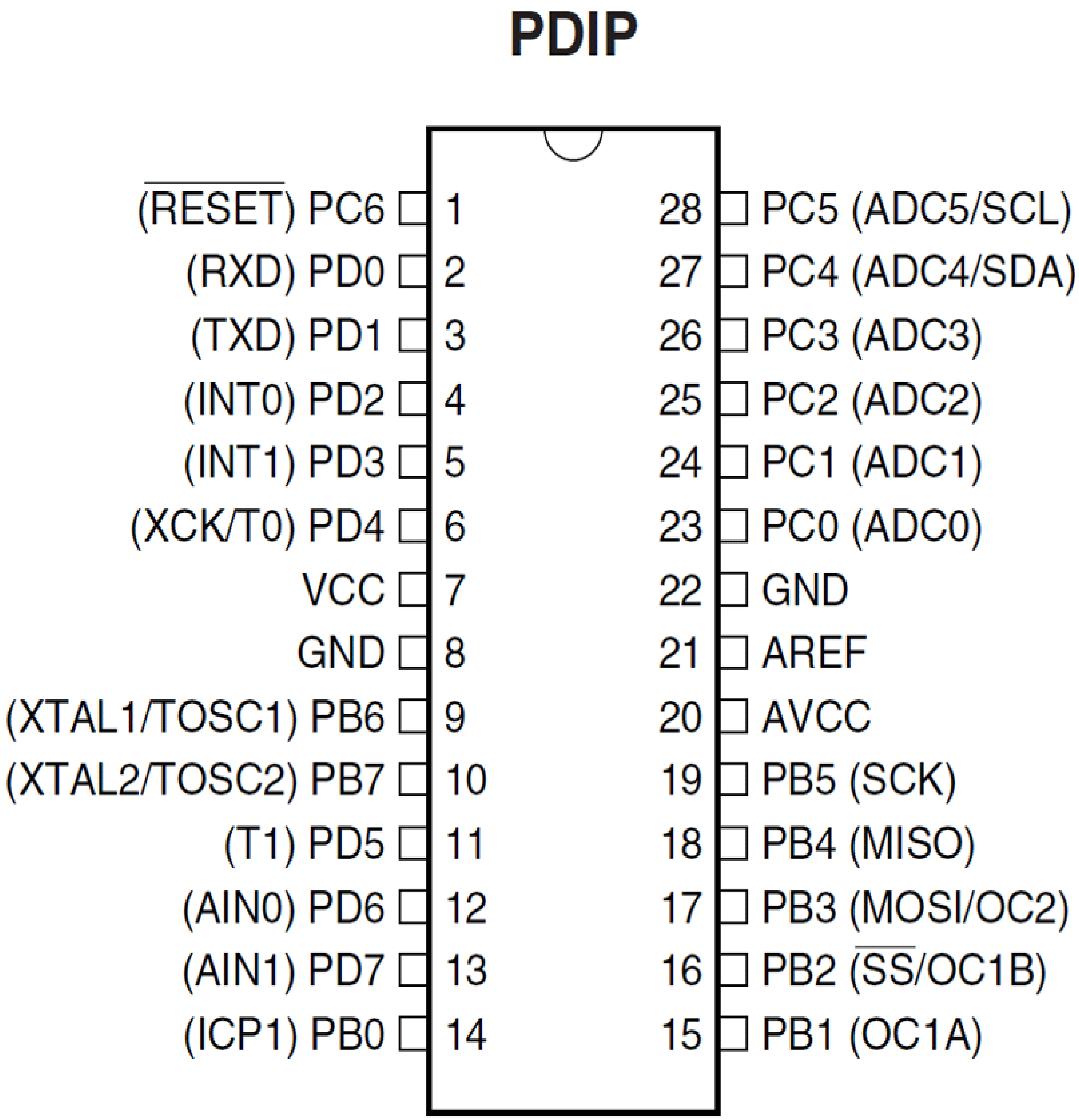


Figure 4.6

4.7 GLOBAL POSITIONING SYSTEM:

The Global Positioning System (GPS) is a satellite-based navigation system made up of a network of 24 satellites placed into orbit by the U.S. Department of Defence. [1] Military actions was the original intent for GPS, however in the 1980s, the U.S. government decided to allow the GPS program to be used by civilians. Weather conditions do not affect the ability for GPS to work. The systems works 24/7 anywhere in the world. There are no subscription fees or setup charges to use GPS.

Figure 4.7 GPS Module

4.7.1 GPS DEVICES MAY HAVE CAPABILITIES SUCH AS:

1. maps, including streets maps, displayed in human readable format via text or in a graphical format,
2. turn-by-turn navigation directions to a human in charge of a vehicle or vessel via text or speech,
3. directions fed directly to an autonomous vehicle such as a robotic probe,
4. traffic congestion maps (depicting either historical or real time data) and suggested alternative directions,
5. information on nearby amenities such as restaurants, fueling stations, and tourist attractions.

4.7.2 GPS MAY BE ABLE TO ANSWER:

1. the roads or paths available,
2. traffic congestion and alternative routes,
3. roads or paths that might be taken to get to the destination,
4. if some roads are busy (now or historically) the best route to take,
5. the location of food, banks, hotels, fuel, airports or other places of interests,
6. the shortest route between the two locations,
7. the different options to drive on highway or back roads.

4.8 GLOBAL SYSTEM OF MOBILE COMMUNICATION:

phone with its own unique phone number. Advantage of using this modem will be that you can use its RS232 port to communicate and develop embedded applications. Applications like SMS Control, data This GSM Modem can accept any GSM network operator SIM card and act just like a mobile transfer, remote control and logging can be developed easily.

The modem can either be connected to PC serial port directly or to any microcontroller through MAX232. It can be used to send and receive SMS or make/receive voice calls. It can also be used in GPRS mode to connect to internet and do many applications for data logging and control. In GPRS mode you can also connect to any remote FTP server and upload files for data logging.

This GSM modem is a highly flexible plug and play quad band SIM900A GSM modem for direct and easy integration to RS232 applications. Supports features like Voice, SMS, Data/Fax, GPRS and integrated TCP/IP stack.

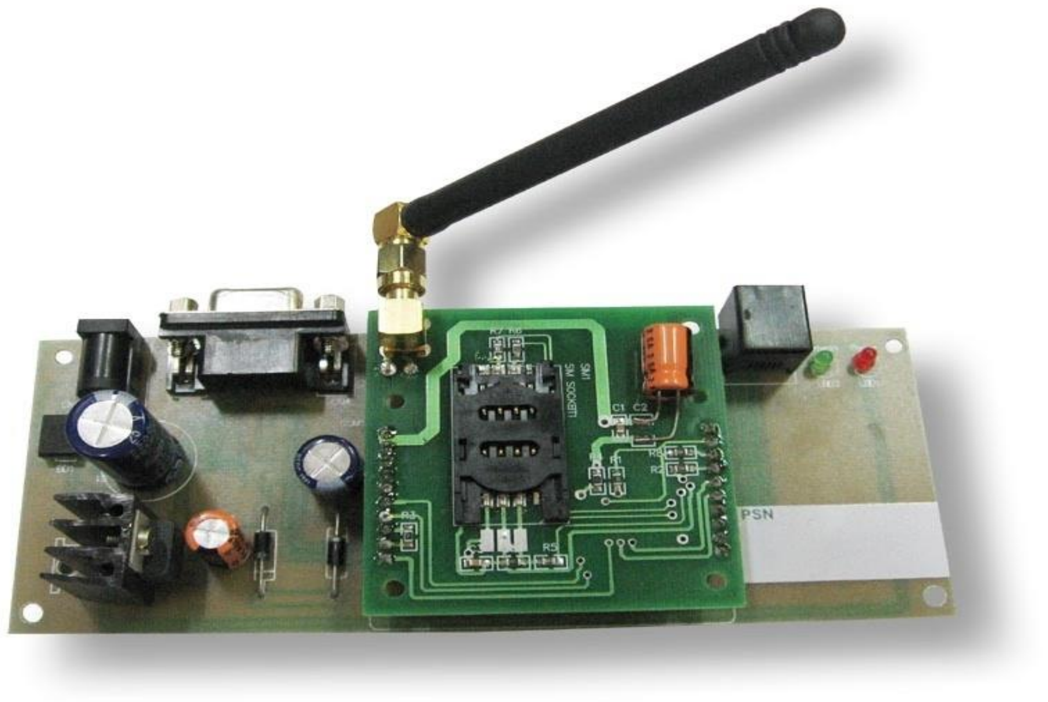


Figure 4.8 GSM Module

4.8.1 APPLICATIONS:

- SMS based Remote Control & Alerts
- Security Applications
- Sensor Monitoring
- GPRS Mode Remote Data Logging

4.9 MAX232

The MAX232 is an IC, first created in 1987 by Maxim Integrated Products, that converts signals from an RS-232 serial port to signals suitable for use in TTL compatible digital logic circuits. The MAX232 is a dual driver/receiver and typically converts the RX, TX, CTS and RTS signals.

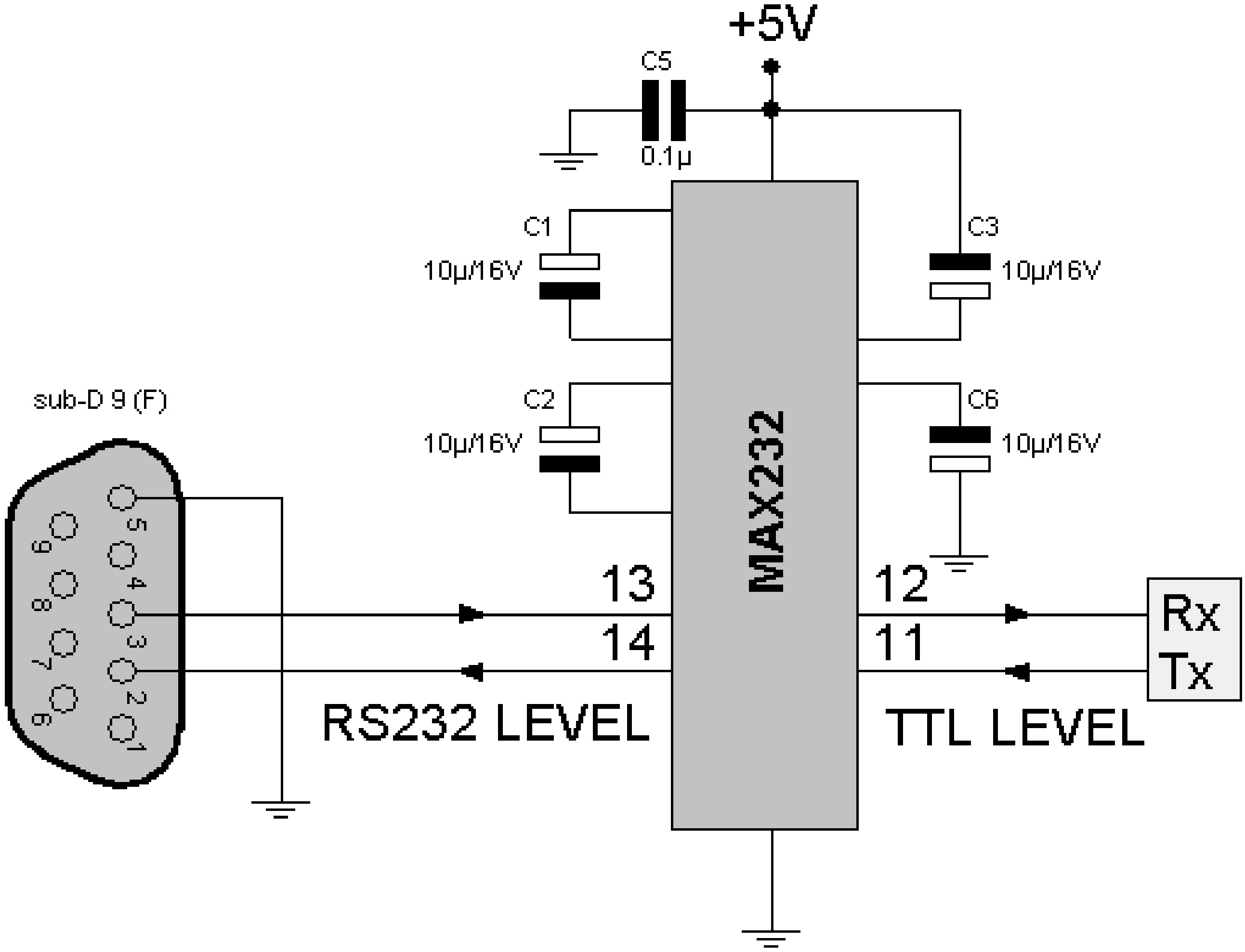


Figure 4.9 Circuit Diagram of MAX232

The drivers provide RS-232 voltage level outputs (approx. ± 7.5 V) from a single + 5 V supply via on-chip charge pumps and external capacitors. This makes it useful for implementing RS-232 in devices that otherwise do not need any voltages outside the 0 V to + 5 V range, as power supply design does not need to be made more complicated just for driving the RS-232 in this case.

The receivers reduce RS-232 inputs (which may be as high as ± 25 V), to standard 5 V TTL levels. These receivers have a typical threshold of 1.3 V, and a typical hysteresis of 0.5 V.

When communicating with various micro processors one needs to convert the RS232 levels down to lower levels, typically 3.3 or 5.0 Volts.

Serial RS-232 (V.24) communication works with voltages -15V to +15V for high and low. On the other hand, TTL logic operates between 0V and +5V . Modern low power consumption logic operates in the range of 0V and +3.3V or even lower.

Thus the RS-232 signal levels are far too high TTL electronics, and the negative RS-232 voltage for high can't be handled at all by computer logic. To receive serial data from an RS-232 interface the voltage has to be reduced. Also the low and high voltage level has to be inverted.

This level converter uses a Max232 and five capacitors. The max232 is quite cheap(less than 5 dollars) or if youre lucky you can get a free sample from Maxim.

The MAX232 from Maxim was the first IC which in one package contains the necessary drivers and receivers to adapt the RS-232 signal voltage levels to TTL logic. It became popular, because it just needs one voltage (+5V or +3.3V) and generates the necessary RS-232 voltage levels.

4.9.1 FEATURES:

Input supply voltage range, V_{CC} (see Note 1) . . . . −0.3 V to 6 V

Positive output supply voltage range, V_{S+} . . . . V_{CC} − 0.3 V to 15 V

Negative output supply voltage range, V_{S-} . . . . −0.3 V to −15 V

Input voltage range, V_I: Driver . . . . −0.3 V to V_{CC}+ 0.3 V

Receiver . . . . ±30 V

R1OUT, R2OUT . . . . −0.3 V to V_{CC} + 0.3 V

Short-circuit duration:

T1OUT, T2OUT . . . . Unlimited

Package thermal impedance, θ_{JA} (see Notes 2 and 3): D package . . . . 73°C/W

DW package . . . . 57°C/W

N package . . . . 67°C/W

NS package . . . . 64°C/W

Operating virtual junction temperature, T_J . . . . 150°C

The intermediate link is provided through MAX232. It is a dual driver/receiver that includes a capacitive voltage generator to supply RS232 voltage levels from a single 5V supply. Each receiver converts RS232 inputs to 5V TTL/CMOS levels. These receivers (R_1 & R_2) can accept ±30V inputs. The drivers (T_1 & T_2), also called transmitters, convert the TTL/CMOS input level into RS232 level.

The transmitters take input from controller’s serial transmission pin and send the output to RS232’s receiver. The receivers, on the other hand, take input from transmission pin of RS232 serial port and give serial output to microcontroller’s receiver pin. MAX232 needs four external capacitors whose value ranges from 1µF to 22µF.

4.9.2 THE REQUIRED PARTS:

- 1 female serial port connector
- 1 max 232
- 4 1uF capacitor
- 1 x 10uF capacitor
- Soldering iron, wires, breadboard etc.

4.10 LIQUID CRYSTAL DISPLAY:

A liquid crystal display (LCD) is a flat panel display, electronic visual display, video display that uses the light modulating properties of liquid crystals (LCs). LCs do not emit light directly.

They are used in a wide range of applications, including computer monitors, television, instrument panels, aircraft cockpit displays, signage, etc. They are common in consumer devices such as video players, gaming devices, clocks, watches, calculators, and telephones. LCDs have replaced cathode ray tube (CRT) displays in most applications. They are available in a wider range of screen sizes than CRT and plasma displays, and since they do not use phosphors, they cannot suffer image burn-in.

4.10.1 FEATURES:

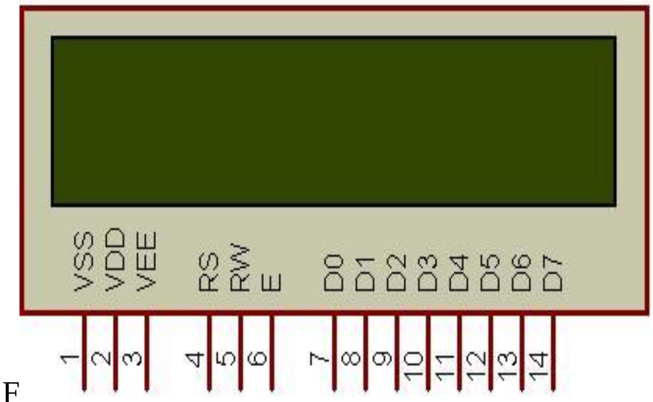
- 5 x 8 dots with cursor
- Built-in controller (KS 0066 or Equivalent)
- + 5V power supply (Also available for + 3V)
- 1/16 duty cycle
- B/L to be driven by pin 1, pin 2 or pin 15, pin 16 or A.K (LED)
- N.V. optional for + 3V power supply

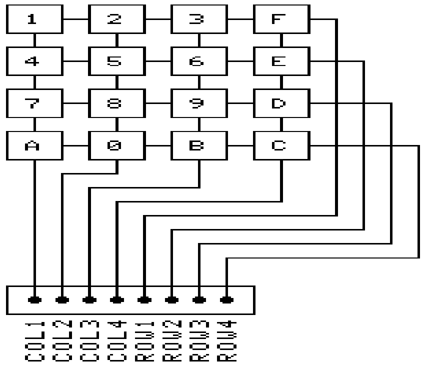
DISPLAY CHARACTER ADDRESS CODE:

Display Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DD RAM Address	00	01														0F
DD RAM Address	40	41														4F

Figure 4.10

A matrix keypad is a combination of conducting rows and column. The combination of these rows and columns is generated by the number present at their intersection and thus indicates the microcontroller to read data accordingly.





F
Figure 4.11 4X3 Matrix keypad

4.10.2 PIN DESCRIPTION:

PIN NUMBER	SYMBOL	FUNCTION
1	Vss	GND
2	Vdd	+ 3V or + 5V
3	Vo	Contrast Adjustment
4	RS	H/L Register Select Signal
5	R/$\overline{W}$	H/L Read/Write Signal
6	E	H →L Enable Signal
7	DB0	H/L Data Bus Line
8	DB1	H/L Data Bus Line
9	DB2	H/L Data Bus Line
10	DB3	H/L Data Bus Line
11	DB4	H/L Data Bus Line
12	DB5	H/L Data Bus Line
13	DB6	H/L Data Bus Line
14	DB7	H/L Data Bus Line
15	A/Vee	+ 4.2V for LED/Negative Voltage Output
16	K	Power Supply for B/L (OV)

Table 4.3

CHAPTER 5. BLOCK DIAGRAM OF CIRCUIT

BLOCK DIAGRAM OF CIRCUIT

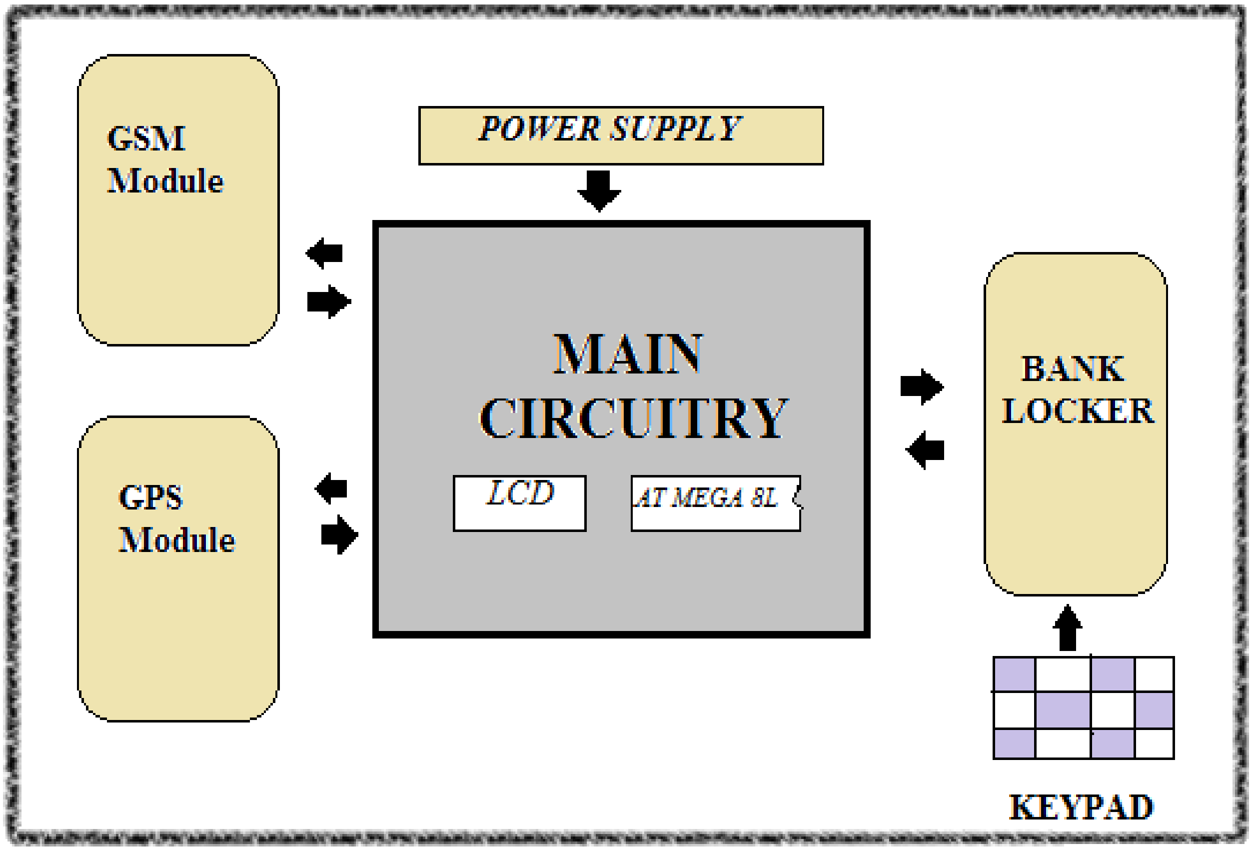


Figure 5.1

CHAPTER 6. PCB MANUFACTURING PROCESS

P.C.B MANUFACTURING PROCESS:-

It is an important process in the fabrication of electronic equipment. The design of PCBs (Printed Circuit Boards) depends on circuit requirements like noise immunity, working frequency and voltage levels etc. High power PCBs require a special design strategy.

The fabrication process to the printed circuit board will determine to a large extent the price and reliability of the equipment. A common target aimed is the fabrication of small series of highly reliable professional quality PCBs with low investment. The target becomes especially important for customer tailored equipments in the area of industrial electronics.

The layout of a PCB has to incorporate all the information of the board before one can go on the artwork preparation. This means that a concept which clearly defines all the details of the circuit and partly defines the final equipment, is prerequisite before the actual lay out can start. The detailed circuit diagram is very important for the layout designer but he must also be familiar with the design concept and with the philosophy behind the equipment.

6.1 BOARD TYPE:

The two most popular PCB types are:

6.1.1 SINGLE SIDED BOARDS

The single sided PCBs are mostly used in entertainment electronics where manufacturing costs have to be kept at a minimum. However in industrial electronics cost factors cannot be neglected and single sided boards should be used wherever a particular circuit can be accommodated on such boards.

6.1.2 DOUBLE SIDED BOARDS

Double-sided PCBs can be made with or without plated through holes. The production of boards with plated through holes is fairly expensive. Therefore plated through hole boards are only chosen where the circuit complexities and density of components does not leave any other choice.

6.2 DESIGN SPECIFICATIONS:

STEPS TAKEN WHILE PREPARING CIRCUIT

6.2.1 PCB DESIGNING:

The main purpose of printed circuit is in the routing of electric currents and signal through a thin copper layer that is bounded firmly to an insulating base material sometimes called the substrate. This base is manufactured with an integrally bounded layers of thin copper foil which has to be partly etched or removed to arrive at a pre-designed pattern to suit the circuit connections or other applications as required.

The term printed circuit board is derived from the original method where a printed pattern is used as the mask over wanted areas of copper. The PCB provides an ideal baseboard upon which to assemble and hold firmly most of the small components.

From the constructor's point of view, the main attraction of using PCB is its role as the mechanical support for small components. There is less need for complicated and time consuming metal work of chassis contraception except perhaps in providing the final enclosure. Most straight forward circuit designs can be easily converted in to printed wiring layer the thought required to carry out the inversion cab footed high light an possible error that would otherwise be missed in conventional point to point wiring .The finished project is usually neater and truly a work of art.

Actual size PCB layout for the circuit shown is drawn on the copper board. The board is then immersed in $FeCl_3$ solution for 12 hours. In this process only the exposed copper portion is etched out by the solution.

Now the petrol washes out the paint and the copper layout on PCB is rubbed with a smooth sand paper slowly and lightly such that only the oxide layers over the Cu are removed. Now the holes are drilled at the respective places according to component layout as shown in figure.

6.3 LAYOUT DESIGN:

When designing the layout one should observe the minimum size (component body length and weight). Before starting to design the layout we need all the required components in hand so that an accurate assessment of space can be made. Other space considerations might also be included from case to case of mounted components over the printed circuit board or to access path of present components.

It might be necessary to turn some components around to a different angular position so that terminals are closer to the connections of the components. The scale can be checked by positioning the components on the squared paper. If any connection crosses, then one can reroute to avoid such condition.

All common or earth lines should ideally be connected to a common line routed around the perimeter of the layout. This will act as the ground plane. If possible try to route the outer supply line to the ground plane. If possible try to route the other supply lines around the opposite edge of the layout through the center. The first set is tearing the circuit to eliminate the crossover without altering the circuit detail in any way.

Plan the layout looking at the topside to this board. First this should be translated inversely, later for the etching pattern large areas are recommended to maintain good copper adhesion. It is important to bear in mind always that copper track width must be according to the recommended minimum dimensions and allowance must be made for increased width where termination holes are needed. From this aspect, it can become little tricky to negotiate the route to connect small transistors.

There are basically two ways of copper interconnection patterns under side the board. The first is the removal of only the amount of copper necessary to isolate the junctions of the components to one another. The second is to make the interconnection pattern looking more like conventional point wiring by routing uniform width of copper from component to component.

6.4 ETCHING PROCESS:

Etching process requires the use of chemicals. acid resistant dishes and running water supply. Ferric chloride is mostly used solution but other etching materials such as ammonium per sulphate can be used. Nitric acid can be used but in general it is not used due to poisonous fumes.

The pattern prepared is glued to the copper surface of the board using a latex type of adhesive that can be cubed after use. The pattern is laid firmly on the copper using a very sharp knife to cut round the pattern carefully to remove the paper corresponding to the required copper pattern areas. Then apply the resistant solution, which can be a kind of ink solution for the purpose of maintaining smooth clean outlines as far as possible. While the board is drying, test all the components.

Before going to next stage, check the whole pattern and cross check with the circuit diagram. Check for any free metal on the copper. The etching bath should be in a glass or enamel disc. If using crystal of ferric- chloride these should be thoroughly dissolved in water to the proportion suggested. There should be 0.5 lt. of water for 125 gm of crystal.

The board should not be left in the bath a moment longer than is needed to remove just the right amount of copper. Inspite of there being a resistive coating there is no protection against etching away through exposed copper edges. This leads to over etching. Have running water ready so that etched board can be removed properly and rinsed. This will halt etching immediately.Drilling is one of those operations that calls for great care. For most purposes a 0.5mm drill is used. Drill all holes with this size first those that need to be larger can be easily drilled again with the appropriate larger size.

6.5 PCB LAYOUT:

The PCB Layout of the circuit after the etching process came out to be as under.

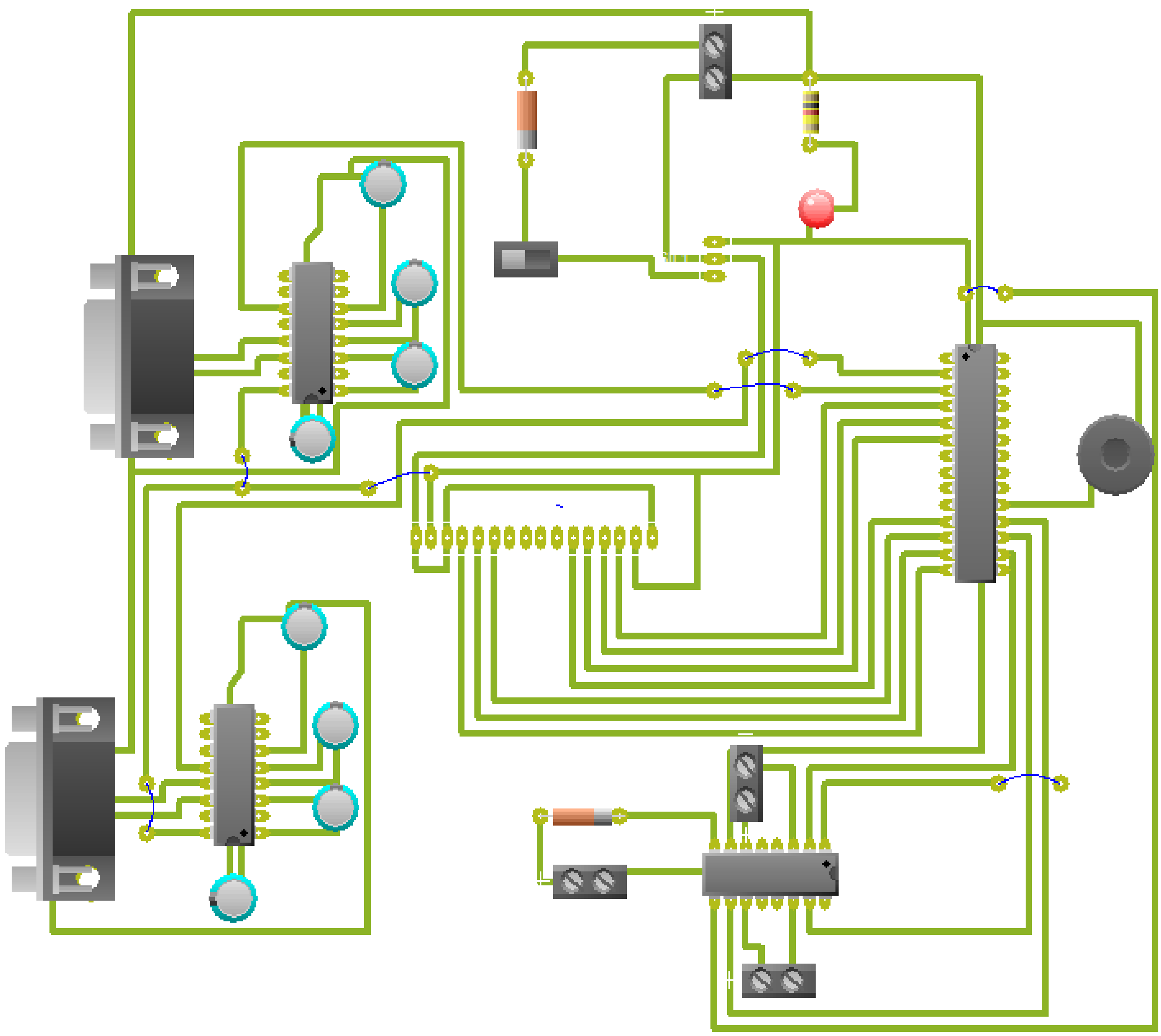


Figure 6.1

6.6 COMPONENT ASSEMBLY

From the greatest variety of electronic components available, which runs into thousands of different types it is often a perplexing task to know which is right for a given job.

There could be damage such as hairline crack on PCB. If there are, then they can be repaired by soldering a short link of bare copper wire over the affected part.

The most popular method of holding all the items is to bring the wires far apart after they have been inserted in the appropriate holes. This will hold the component in position ready for soldering.

Some components will be considerably larger .So it is best to start mounting the smallest first and progressing through to the largest. Before starting, be certain that no further drilling is likely to be necessary because access may be impossible later.

Next will probably be the resistor, small signal diodes or other similar size components. Some capacitors are also very small but it would be best to fit these afterwards. When fitting each group of components mark off each one on the circuit as it is fitted so that if we have to leave the job we know where to recommence.

Although transistors and integrated circuits are small items there are good reasons for leaving the soldering of these until the last step. The main point is that these components are very sensitive to heat and if subjected to prolonged application of the soldering iron, they could be internally damaged.

All the components before mounting are rubbed with sand paper so that oxide layer is removed .from the tips. Now they are mounted according to the component layout.

6.7 COMPONENT MOUNTING:

All components are mounted at their respective position as per the components layout. Proper precautions should be taken during mounting process.

6.8 SOLDERING:

Soldering is a process in which two or more metal items are joined together by melting and flowing a filler metal (solder) into the joint, the filler metal having a lower melting point than the work piece.

A soldered connection ensures metal continuity. The soldering process involves: Melting of the flux which in turn removes the oxide films on the metal to be soldered. Melting the solder which removes the impurities. The solder partially dissolve of the metal in the connection. The solder cools and fuses wit the metal.

The soldering techniques involves knowledge of:

1. Soldering iron
2. Soldering wire
3. Soldering procedure
4. Replacing components
5. Knowledge of good and bad soldering joints.
6. Disordering techniques
7. Soldering iron is an essential tool for soldering. A. Soldering iron should give sufficient heat a melt solder by heat transfer when the iron tip is applied to a connection to be soldered. The selection of the soldering iron can be made as regard to its tips size shape and wattage.
8. The soldering material is used to join together two or more metals at temperatures below their melting point. The solder alloy consists of Lead (37%) and Tin (63%). The continuous connection between two metal joint is made by solder materials.
9. Flux is a material used to aid soldering process. Flux is needed to scratch away the small film of oxide on the surface of metals to be soldered.

6.8.1 SOLDERING PROCEDURE

The soldering procedure involves selection of soldering iron cleaning of components to be soldered and cleaning of the PCB to be soldered. The soldering iron should be selected according to the job and should be powerful enough to provide heat. The tip of the soldering iron should be selected as per the space available for soldering. The component that has to be soldered should be properly bent and its leads should properly inserted in the PCB. Before. If one has already identified the fault component, then one should not try to remove or desolder the component. The components should simply be cut and taken out.

6.9 DESOLDERING TECHNIQUES

6.9.1 BY USING A DISORDERING WICK : Disordering wick is made of fine copper wire mesh. When this is applied to the heated components, the molten solder gets attached to the wire mesh by capillary action.

6.9.2 BY USING A DISORDERING PUMP: Disordering pump has a suction pump. The nozzle of the disordering pump is kept to the heated component. The molten solder is sucked by a spring action. Insertion in the PCB, the lead should be properly cleaned. After component has been inserted it can be soldered. The oxide on the PCB can be removed by using flux, sandpaper.

6.10 REPLACEMENT OF COMPONENT

In case of single sided PCB, the component to be removed can be disordered with the help of iron and flux. The only precaution that has to be taken is that track should not break while removing. In case of Through Hole PCB, care has the to be taken so that component while removing does not damaged the Through Hole. In this case the component is soldered on one side and the lead flows through the hole to the other sides, so disordering and removing becomes very difficult and required practice.

CHAPTER 7. CHRONOLOGY

CHRONOLOGY

The following steps have been followed in carrying out the project.

1. Study the books on the relevant topic.

2. Understand the working of the circuit.

3. Prepare the circuit diagram.

4. Prepare the list of components along with their specification. Estimate the costand procure them after carrying out market survey.

5. Plan and prepare PCB for mounting all the components.

6. Fix the components on the PCB and solder them.

7. Test the circuit for the desired performance.

8. Trace and rectify faults if any.

9. Give good finish to the unit.

10. Prepare the project report.

CHAPTER 8. CONCLUSION

CONCLUSION

At the far end of the project we feel that the whole working period was a great learning experience due to a number of friendly & unfriendly encounters with various situations. We also came across a lot of new concepts making our learning a value addition.

The project is only a demonstration of a system that can be developed into a full scale commercial utility for modern day scientific as well as corporate requirements. One of the improvement that is possible, by incorporating advanced IC systems the size of all the hardware units involved can be reduced to great limits. Also the number of users can be increased by simple duplication of circuitry.

An advanced future aspect can be actually controlling the movement of the object by means of this navigation system, sitting in a remote place. Once this feature is incorporated, the system can have innumerable applications in space science.

Beside this there may be many more practical applications of this system and there is a large scope of improvement in its current features.

THANKYOU

REFRENCES

REFERENCES

References for Technical Information are taken from the Following Books on Electronic Engineering:

- Instrument Theory – A.K. Sawhney
- Microcontroller By K. J. Ayala
- Integrated Electronics By Millmann &Halkias
- GSM Made Simple By George Lamb
- Global Positioning System By Pratap Misra And Per Enge

References for Articles and Datasheets are taken from the Following Sites:

- http://www.google.com (Search Engine)
- http://www.national.comJds/LM/LM35.pdf
- http://www.expresspcb.com
- http://en.wikipedia.org
- http://www.amazon.com

APPENDIX

APPENDIX 1

SOURCE CODE

```
#include <mega8.h>
#include <stdlib.h>

// Alphanumeric LCD functions
#include <alcd.h>
#include <delay.h>

// Standard Input/Output functions
#include <stdio.h>

char location[24];
char key()
{
char a='$';
while(1)
    {
        PORTC.3=0;PORTC.4=1;PORTC.5=1;
delay_ms(10);
if(PINB.4==0)
        {
         a='1';
break;
        }

if(PINB.5==0)
        {
         a='4';
break;
        }

if(PINC.1==0)
        {
         a='7';
break;
        }

if(PINC.2==0)
        {
         a='*';
break;
        }

        PORTC.3=1;PORTC.4=0;PORTC.5=1;
delay_ms(10);
if(PINB.4==0)
        {
         a='2';
break;
        }
```

```
if(PINB.5==0)
        {
          a='5';
break;
        }

if(PINC.1==0)
        {
          a='8';
break;
        }

if(PINC.2==0)
        {
          a='0';
break;
        }


        PORTC.3=1;PORTC.4=1;PORTC.5=0;
delay_ms(10);
if(PINB.4==0)
        {
          a='3';
break;
        }

if(PINB.5==0)
        {
          a='6';
break;
        }

if(PINC.1==0)
        {
          a='9';
break;
        }

if(PINC.2==0)
        {
          a='#';
break;
        }
    }

return a;
}

voidgps_read()
{
inti,count;
chara,flush;
delay_ms(500);
for(i=0;i<1000;i++)
    {
```

```
flush=UDR;
    }

while(1)
        {
        a=getchar();
if (a=='$')
            {
            a=getchar();
if(a=='G')
                {
                a=getchar();
if(a=='P')
                    {
                    a=getchar();
if(a=='G')
                        {
                        a=getchar();
if(a=='G')
                            {
                            a=getchar();
if(a=='A')
                                {
count=0;
while(count<2)
                                    {
                                    a=getchar();
if(a==',')
                                        {
count++;
                                        }
                                    }

for(i=0;i<24;i++)
                                    {
location[i]=getchar();
                                    }
break;
                                }
                            }
                        }
                    }
                }
            }
        }
return;
}

voidgsm_send()
{

    char at0[]={'a','t','+','c','m','g','s','=','"','8','8','7','1','9','5','3','2','3','1','"'};
    char at1[]={'a','t','+','c','m','g','s','=','"','9','0','3','9','4','9','2','9','5','1','"'};
    char at2[]={'a','t','+','c','m','g','s','=','"','9','0','9','8','5','0','4','4','3','3','"'};
char msg1[]="Alert! Bank Locker opened @ location : ";
inti,k;
char flush;
```

```
for (k=0;k<3;k++)
    {

delay_ms(500);
for(i=0;i<1000;i++)
        {
flush=UDR;
        }
delay_ms(500);

putchar('a');
putchar('t');
putchar('e');
putchar('0');
putchar(13);

delay_ms(100);
putchar('a');
putchar('t');
putchar('+');
putchar('c');
putchar('m');
putchar('g');
putchar('f');
putchar('=');
putchar('1');
putchar(13);
delay_ms(100);

for(i=0;i<20;i++)
        {
if(k==0)
putchar(at0[i]);


if(k==1)
putchar(at1[i]);


if(k==2)
putchar(at2[i]);
        }
putchar(13);
delay_ms(100);

for(i=0;i<=39;i++)
putchar(msg1[i]);

putchar(location[0]);
putchar(location[1]);
putchar(' ');
putchar('D');
putchar('e');
putchar('g');
putchar('r');
putchar('e');
```

```
putchar('e');
putchar(' ');
putchar(location[2]);
putchar(location[3]);
putchar(location[4]);
putchar(location[5]);
putchar(location[6]);
putchar(location[7]);
putchar(location[8]);
putchar(' ');
putchar('M');
putchar('i');
putchar('n');
putchar('u');
putchar('t');
putchar('e');
putchar('s');
putchar(' ');

putchar('N');
putchar('o');
putchar('r');
putchar('t');
putchar('h');
putchar(' ');
        // putchar(13);

putchar(location[12]);
putchar(location[13]);
putchar(location[14]);
putchar(' ');
putchar('D');
putchar('e');
putchar('g');
putchar('r');
putchar('e');
putchar('e');
putchar(' ');
putchar(location[15]);
putchar(location[16]);
putchar(location[17]);
putchar(location[18]);
putchar(location[19]);
putchar(location[20]);
putchar(' ');
putchar('M');
putchar('i');
putchar('n');
putchar('u');
putchar('t');
putchar('e');
putchar('s');
putchar(' ');

putchar('E');
putchar('a');
putchar('s');
```

```
putchar('t');
putchar(13);

delay_ms(200);
putchar(26);
    }

}



void main(void)
{
inti,pass_length;
charx,pass[16];
int temp=0,d1,d2,z;
char c[2], d[2];
// Declare your local variables here

// Input/Output Ports initialization
// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=Out Func1=Out Func0=In
// State7=T State6=T State5=P State4=P State3=T State2=0 State1=0 State0=T
PORTB=0x30;
DDRB=0b00001100;

// Port C initialization
// Func6=In Func5=Out Func4=Out Func3=Out Func2=In Func1=In Func0=Out
// State6=T State5=0 State4=0 State3=0 State2=P State1=P State0=0
PORTC=0x06;
DDRC=0x39;

// Port D initialization
// Func7=In Func6=In Func5=Out Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=0 State4=T State3=T State2=T State1=T State0=T
PORTD=0x00;
DDRD=0x20;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x33;

// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|CCompiler|Libraries|Alphanumeric LCD menu:
// RS - PORTB Bit 1
// RD - PORTB Bit 0
// EN - PORTD Bit 7
// D4 - PORTD Bit 6
```

```
// D5 - PORTD Bit 4
// D6 - PORTD Bit 3
// D7 - PORTD Bit 2
// Characters/line: 16
lcd_init(16);
delay_ms(1000);
lcd_clear();
gps_read();
lcd_puts(location);
delay_ms(3000);
while (1)
      {
      // Place your code here
lcd_clear();
lcd_putsf("Enter password");
lcd_gotoxy(0,1);

for(i=0;i<16;i++)
        {
            x=key();
if (x=='#')
            {
pass_length=i;
break;
            }
else
            {
pass[i]=x;
lcd_putchar(x);
delay_ms(400);
            }
        }
delay_ms(1000);

if (pass_length!=4)
        {
            PORTC.0=1;
lcd_clear();
lcd_putsf("Wrong Password");
delay_ms(1000);
            PORTC.0=0;
temp++;


if(temp==3)
                    {
temp=0;
gps_read();
gsm_send();

for(z=30;z>=0;z--)
                      {
lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("gate blocked");
                      d1=z/10;
                      d2=z%10;
```

```
itoa(d1,c);
itoa(d2,d);
lcd_gotoxy(14,0);
lcd_puts(c);
lcd_gotoxy(15,0);
lcd_puts(d);
delay_ms(1000);
                }
              }
        }

        else if(pass[0]=='1' & pass[1]=='2' & pass[2]=='3' & pass[3]=='4')
        {
lcd_clear();
lcd_putsf("Correct Password");
gps_read();
gsm_send();
delay_ms(1000);
lcd_gotoxy(0,1);
lcd_putsf("Opening Locker!");
          PORTB.2=1; PORTB.3=0;
delay_ms(2000);
          PORTB.2=0; PORTB.3=0;
lcd_clear();
lcd_putsf("Locker opened!");
delay_ms(2000);
lcd_clear();
lcd_putsf("Press any key to exit") ;
key();
          PORTB.2=0; PORTB.3=1;
delay_ms(2000);
          PORTB.2=0; PORTB.3=0;
        }
else
        {
temp++;
          PORTC.0=1;
lcd_clear();
lcd_putsf("Wrong Password");
delay_ms(2000);
          PORTC.0=0;

if(temp==3)
              {
temp=0;
gps_read();
gsm_send();
for(z=30;z>=0;z--)
                {
lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("gate blocked");
                d1=z/10;
                d2=z%10;
itoa(d1,c);
itoa(d2,d);
lcd_gotoxy(14,0);
```

```
lcd_puts(c);
lcd_gotoxy(15,0);
lcd_puts(d);
delay_ms(1000);
                }
              }
        }}}
```

APPENDIX 2: DATA SHEET 1

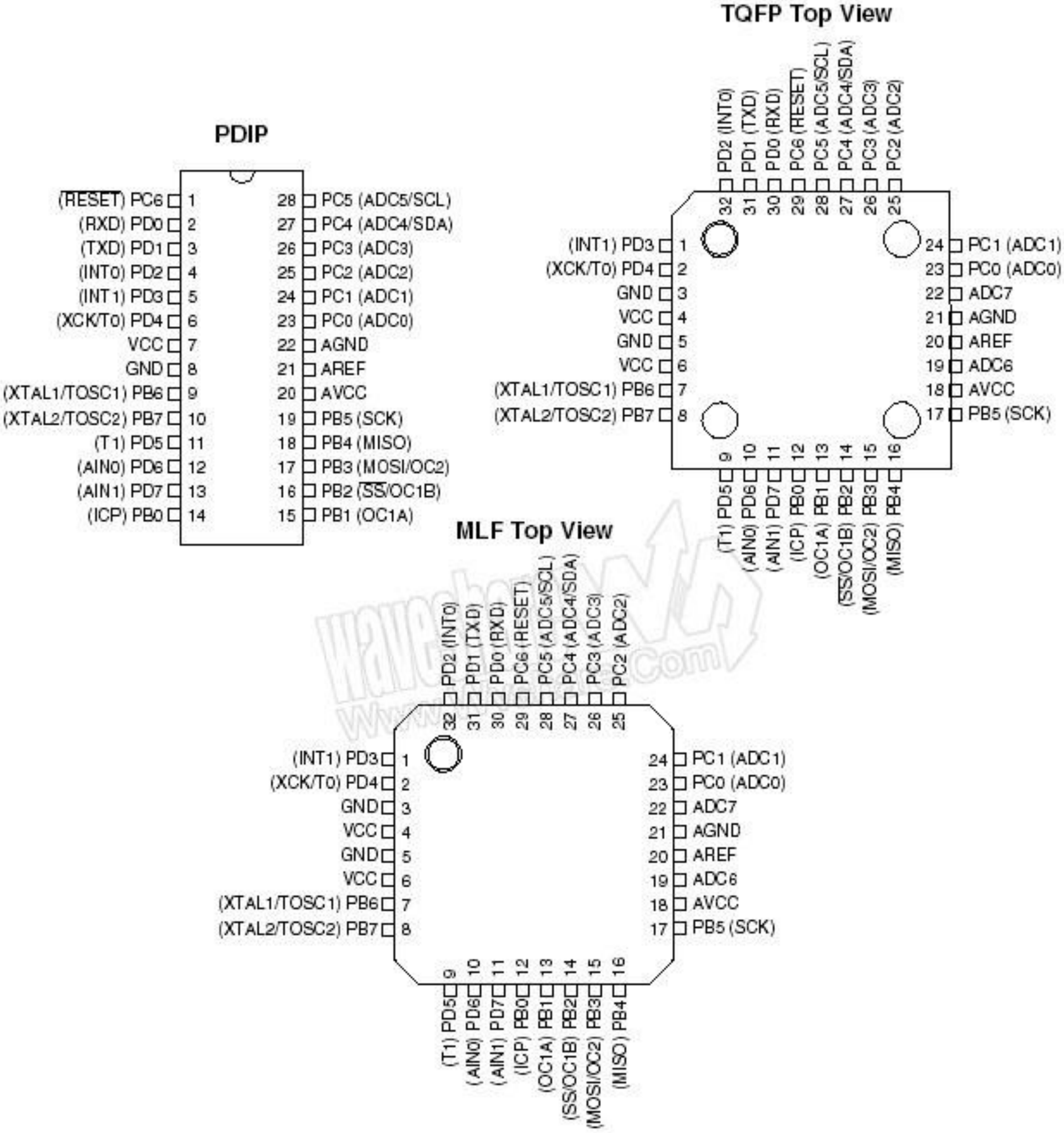
ATMEGA8L

Features

- High-performance, Low-power AVR®8-bit Microcontroller
- Advanced RISC Architecture
  - – 130 Powerful Instructions – Most Single-clock Cycle Execution
  - – 32 x 8 General Purpose Working Registers
  - – Fully Static Operation
  - – Up to 16 MIPS Throughput at 16 MHz
  - – On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
  - – 8K Bytes of In-System Self-programmable Flash program memory
  - – 512 Bytes EEPROM
  - – 1K Byte Internal SRAM
  - – Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - – Data retention: 20 years at 85°C/100 years at 25°C[1]
  - – Optional Boot Code Section with Independent Lock Bits
    In-System Programming by On-chip Boot Program
    True Read-While-Write Operation
  - – Programming Lock for Software Security
- Peripheral Features
  - – Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
  - – One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - – Real Time Counter with Separate Oscillator
  - – Three PWM Channels
  - – 8-channel ADC in TQFP and QFN/MLF package
    Eight Channels 10-bit Accuracy
  - – 6-channel ADC in PDIP package
    Six Channels 10-bit Accuracy
  - – Byte-oriented Two-wire Serial Interface
  - – Programmable Serial USART
  - – Master/Slave SPI Serial Interface
  - – Programmable Watchdog Timer with Separate On-chip Oscillator
  - – On-chip Analog Comparator
- Special Microcontroller Features
  - – Power-on Reset and Programmable Brown-out Detection
  - – Internal Calibrated RC Oscillator
  - – External and Internal Interrupt Sources

  - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
  - 23 Programmable I/O Lines
  - 28-lead PDIP, 32-lead TQFP, and 32-pad QFN/MLF
- Operating Voltages
  - 2.7 - 5.5V (ATmega8L)
  - 4.5 - 5.5V (ATmega8)
- Speed Grades
  - 0 - 8 MHz (ATmega8L)
  - 0 - 16 MHz (ATmega8)
- Power Consumption at 4 Mhz, 3V, 25°C
  - Active: 3.6 mA
  - Idle Mode: 1.0 mA
  - Power-down Mode: 0.5 µA

PIN CONFIGURATIONS

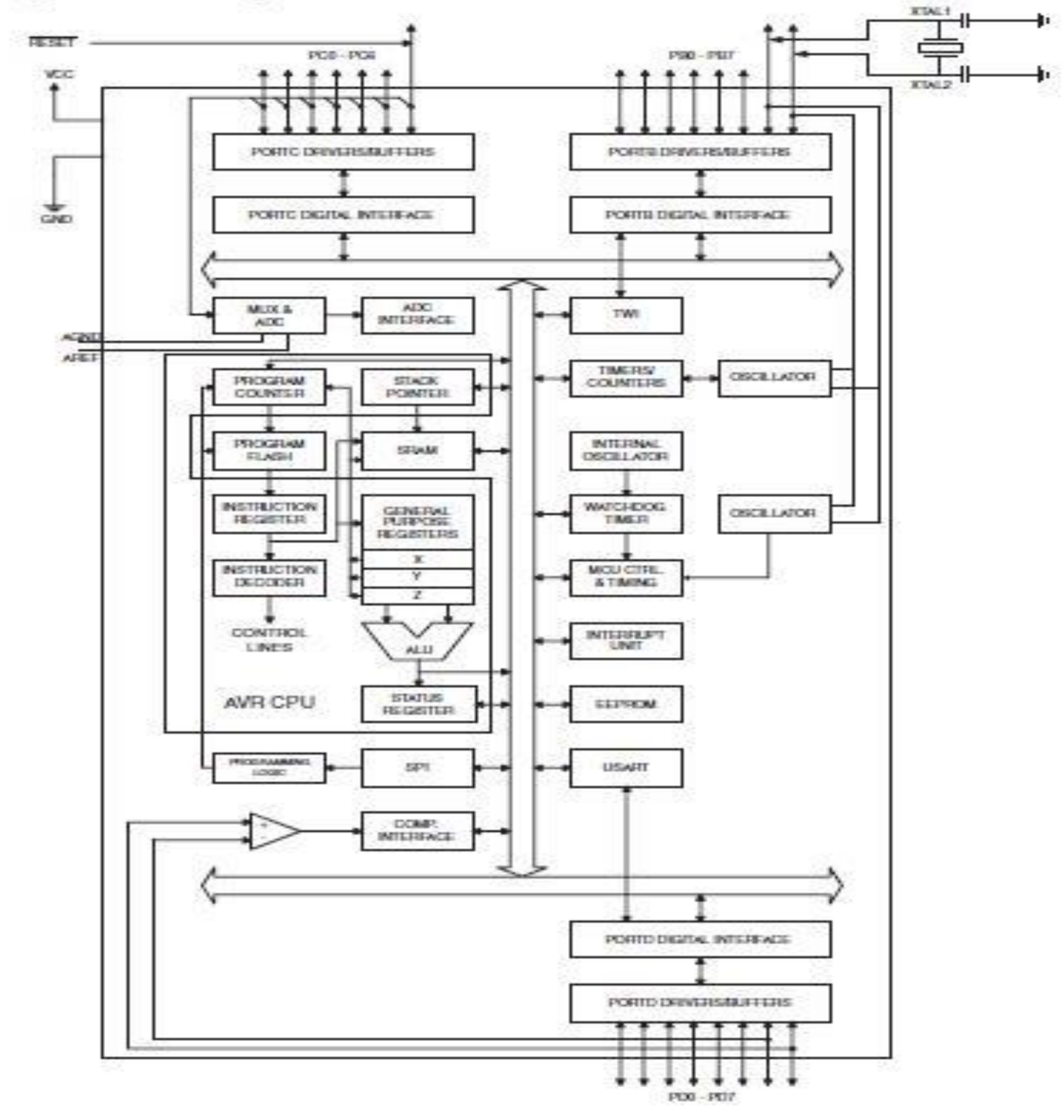


OVERVIEW

The ATmega8 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8 achieves throughputs approaching 1 MIPS per MHz, allowing the system designer to optimize power consumption versus processing speed.

BLOCK DIAGRAM:

Figure 1. Block Diagram



The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega8 provides the following features: 8K bytes of In-System Programmable Flash with Read-While-Write capabilities, 512 bytes of EEPROM, 1K byte of SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, a 6-channel ADC (eight channels in TQFP and QFN/MLF packages) with 10-bit accuracy, a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next Interrupt or Hardware Reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption.

The device is manufactured using Atmel’s high density non-volatile memory technology. The Flash Program memory can be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash Section will continue to run while the Application Flash Section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega8 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.
The ATmega8 AVR is supported with a full suite of program and system development tools, including C compilers, macro assemblers, program debugger/simulators, In-Circuit Emulators, and evaluation kits.

Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

PIN DESCRIPTION

VCC
Digital supply voltage. Magnitude of the voltage range between 4.5 to 5.5 V for the ATmega8 and 2.7 to 5.5 V for ATmega8L.

GND
Ground. Zero reference digital voltage supply.

PORTB (PB7.. PB0)
PORTB is a port I / O two-way (bidirectional) 8-bit with internal pull-up resistor can be selected. This port output buffers have symmetrical characteristics when used as a source or sink. When used as an input, the pull-pin low externally will emit a current if the pull-up resistor is activated it. PORTB pins will be in the condition of the tri-state when RESET is active, although the clock is not running.

PORTC (PC5.. PC0)
PORTC is a port I / O two-way (bidirectional) 7-bit with internal pull-up resistor can be selected. This port output buffers have symmetrical characteristics when used as a source or sink. When used as an input, the pull-pin low externally will emit a current if the pull-up resistor is activated it. PORTC pins will be in the condition of the tri-state when RESET is active, although the clock is not running.

PC6/RESET
If RSTDISBL Fuse programmed, PC6 then serves as a pin I / O but with different characteristics .. PC0 to PC5. If Fuse RSTDISBL not programmed, then serves as input Reset PC6. LOW signal on this pin with a minimum width of 1.5 microseconds will bring the microcontroller into reset condition, although the clock is not running.

PORTD (PD7.. PD0)
PORTD is a port I / O two-way (bidirectional) 8-bit with internal pull-up resistor can be selected. This port output buffers have symmetrical characteristics when used as a source or sink. When used as an input, the pull-pin low externally will emit a current if the pull-up resistor is activated it. PORTD pins will be in the condition of the tri-state when RESET is active, although the clock is not running.

RESET

Reset input pin. LOW signal on this pin with a minimum width of 1.5 microseconds will bring the microcontroller into reset condition, although the clock is not running. Signal with a width of less than 1.5 microseconds does not guarantee a Reset condition.

AVCC

AVCC is the supply voltage pin for the ADC, PC3 .. PC0, and ADC7 .. ADC6. This pin should be connected to VCC, even if the ADC is not used. If the ADC is used, AVCC should be connected to VCC through a low-pass filter to reduce noise.

Aref

Analog Reference pin for the ADC.

ADC7 .. ADC6

ADC analog input. There is only on ATmega8 with TQFP and QFP packages / MLF.

APPENDIX 3: DATA SHEET 2

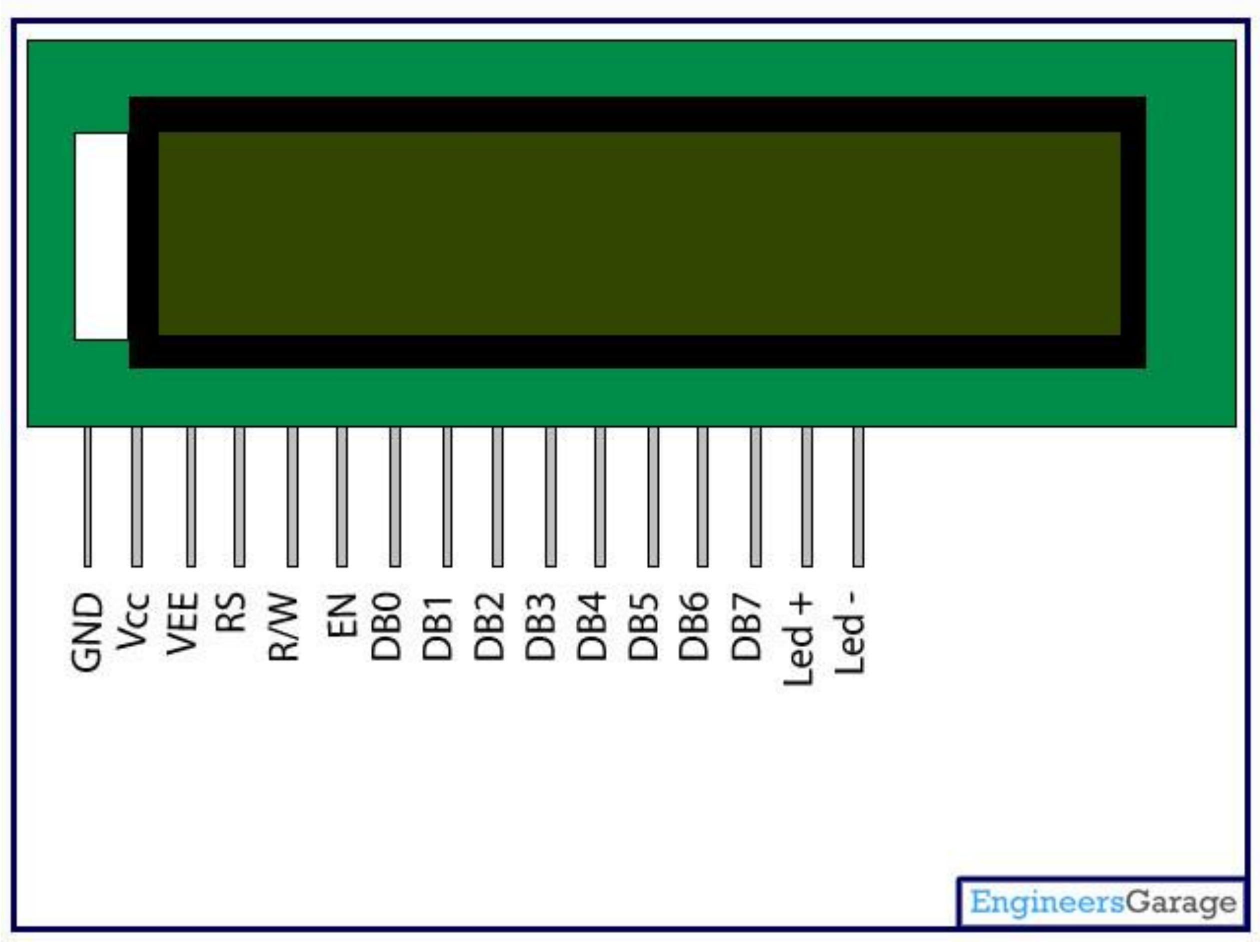
LIQUID CRYSTAL DISPLAY 16*2

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

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

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

PIN DIAGRAM:


GND
Vcc
VEE
RS
R/W
EN
DB0
DB1
DB2
DB3
DB4
DB5
DB6
DB7
Led +
Led -
EngineersGarage

PIN DESCRIPTION:

Pin No	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	Vcc
3	Contrast adjustment; through a variable resistor	V_{EE}
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable
7	8-bit data pins	DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15	Backlight V_{CC} (5V)	Led+
16	Backlight Ground (0V)	Led-

GENERAL INFORMATIOM

DESCRIPTION

This is an lcd display designed for e-blocks. it is a 16 character, 2-line alphanumeric lcd display connected to a single 9-way d-type connector. this allows the device to be connected to most e-block i/o ports.

The lcd display requires data in a serial format, which is detailed in the user guide below. the display also requires a 5v power supply. please take care not to exceed 5v, as this will cause damage to the device.

The 5v is best generated from the e-blocks multipogrammer or a 5v fixed regulated power supply.

The potentiometer rv1 is a contrast control that should be used to adjust the contrast of the display for the environment it is being used in.

features

· e-blocks compatible

· low cost

· compatible with most i/o ports in the e-block range (requires 5 i/o lines via 9 way d-type

connector)

· ease to develop programming code using flowcode icons.

GETTING STARTED

As can be seen the circuit diagram consists of a simple LCD circuit. To test this board you will need to apply 5V to the LCD Board via the screw terminal and then set the appropriate bits using the 9-

Way D-type connector. The details off the LCD configuration are stated in the Users' guide.

Testing the LED Board – LCD.hex

The following instructions explain the steps to test and use your LCD Board. The instructions assume that PPP is installed and functional. It also assumes that you are confident in sending a program to the PIC via the multiprogrammer.

The LCD.hex program will place a counter on to the LCD Board

1) Ensure power is supplied to all the necessary boards.

2) Insert the LCD board into Port B of the Multiprogrammer

3) Ensure that the Multiprogrammer is in correct configuration

- Fast mode (SW1 towards the centre of the board)

- Ensure that a 19.6608MHz crystal is inserted in the Multiprogrammer board

SW2 is not used when in Xtal mode so it doesn't matter it's position

4) Program the a PIC16F88 with the test program LCD.hex

5) Press the reset button on the Multiprogrammer.

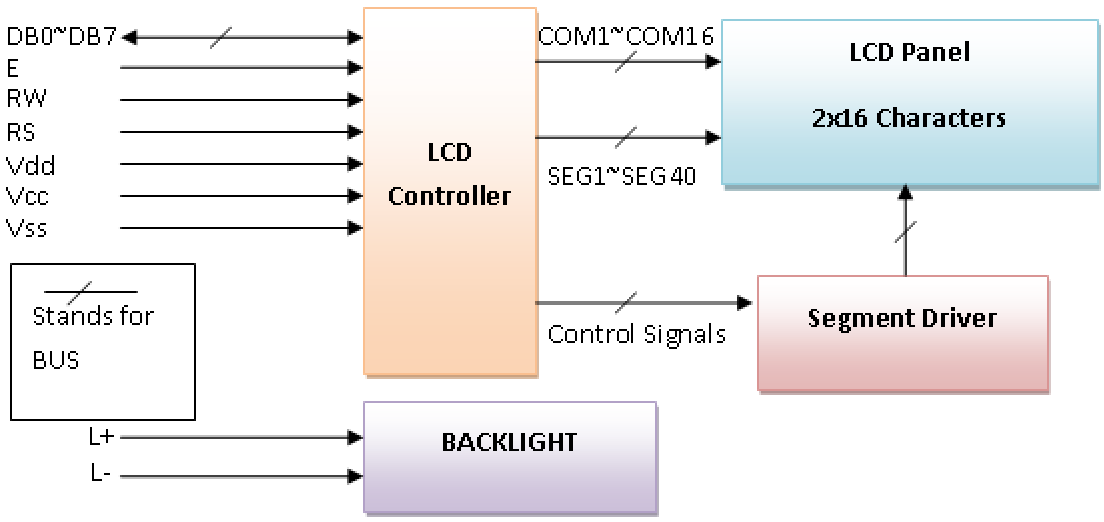
6) Adjust contrast accordingly.

This should satisfy that the LCD Board is fully functional!

PRECAUTIONS IN USE OF PLED MODULES

(1) Avoid applying excessive shocks to the module or making any alterations or modifications to it.
(2) Don\u2019t make extra holes on the printed circuit board, modify its shape or change the components of PLED module.
(3) Don\u2019t disassemble the PLEDM.
(4) Don\u2019t operate it above the absolute maximum rating.
(5) Don\u2019t drop, bend or twist PLEDM.
(6) Soldering: only to the I/O terminals.
(7) Storage: please storage in anti-static electricity container and clean environment

BLOCK DIAGRAM



APPENDIX 4: DATA SHEET 3

MAX232

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply TIA/EIA-232-F voltage levels from a single 5-V supply. Each receiver converts TIA/EIA-232-F inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V, and can accept □30-V inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232-F levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC library.

FUNCTIONS ON TABLE

INPUT TIN	OUTPUT TOUT
L	H
H	L

H = high level, L = low level

EACH RECEIVER

INPUT RIN	OUTPUT ROUT
L	H
H	L

H = high level, L = low level

The **MAX232** is an IC, first created in 1987 by Maxim Integrated Products, that converts signals from an RS-232 serial port to signals suitable for use in TTL compatible digital logic circuits. The MAX232 is a dual driver/receiver and typically converts the RX, TX, CTS and RTS signals.

The drivers provide RS-232 voltage level outputs (approx. ± 7.5 V) from a single + 5 V supply via on-chip charge pumpsand external capacitors. This makes it useful for implementing RS-232 in devices that otherwise do not need any voltages outside the 0 V to + 5 V range, as power supply design does not need to be made more complicated just for driving the RS-232 in this case.

The receivers reduce RS-232 inputs (which may be as high as ± 25 V), to standard 5 V TTL levels. These receivers have a typical threshold of 1.3 V, and a typical hysteresis of 0.5 V.

VERSIONS

The later MAX232A is backwards compatible with the original MAX232 but may operate at higher baud rates and can use smaller external capacitors – 0.1 μF in place of the 1.0 μF capacitors used with the original device.[1] The newer MAX3232 is also backwards compatible, but operates at a broader voltage range, from 3 to 5.5 V.

Pin-to-pin compatible versions from other manufacturers are ICL232, ST232, ADM232 and HIN232.

VOLTAGE LEVELS

It is helpful to understand what occurs to the voltage levels. When a MAX232 IC receives a TTL level to convert, it changes a TTL logic 0 to between +3 and +15 V, and changes TTL logic 1 to between -3 to -15 V, and vice versa for converting from RS232 to TTL. This can be confusing when you realize that the RS232 data transmission voltages at a certain logic state are opposite from the RS232 control line voltages at the same logic state. To clarify the matter, see the table below. For more information, see RS-232 voltage levels.

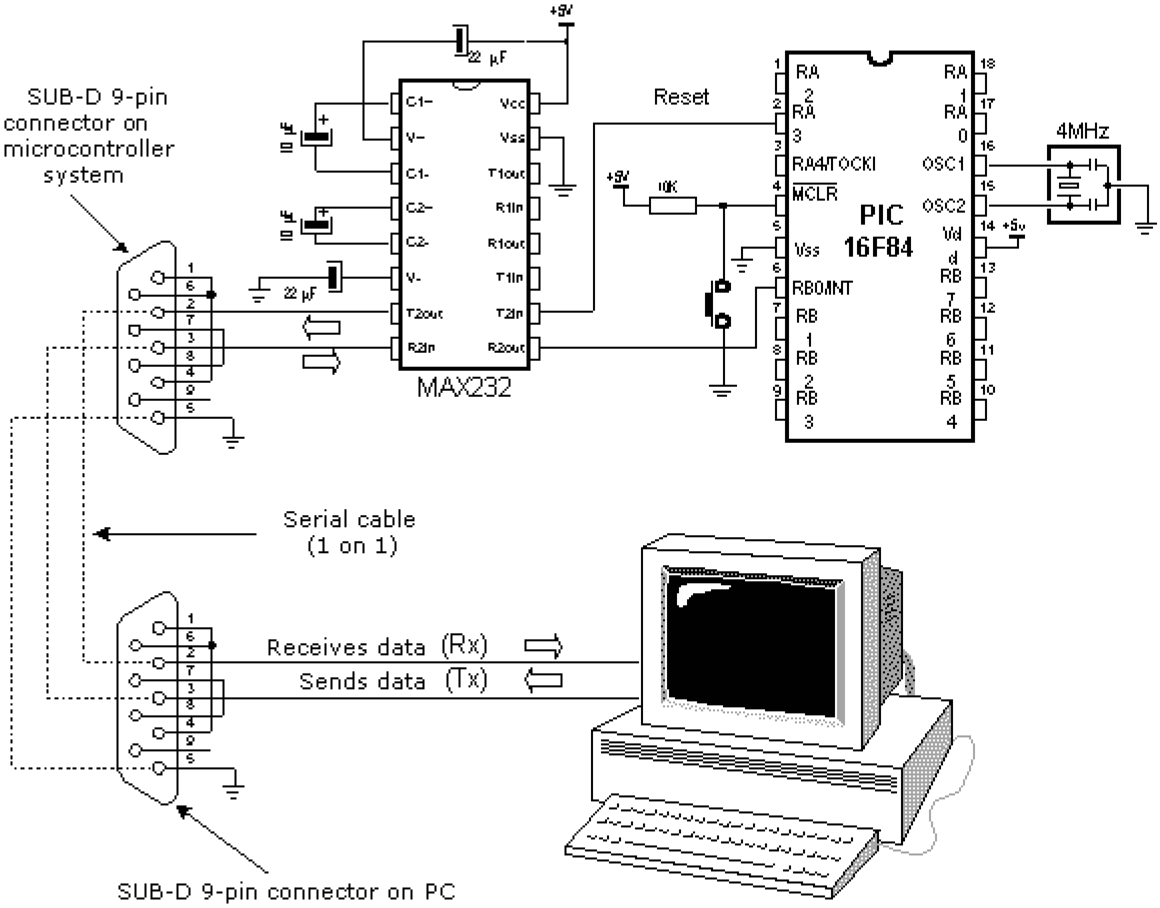
RS232 line type and logic level	RS232 voltage	TTL voltage to/from MAX232
Data transmission (Rx/Tx) logic 0	+3 V to +15 V	0 V
Data transmission (Rx/Tx) logic 1	-3 V to -15 V	5 V
Control signals (RTS/CTS/DTR/DSR) logic 0	-3 V to -15 V	5 V
Control signals (RTS/CTS/DTR/DSR) logic 1	+3 V to +15 V	0 V

APPLICATIONS

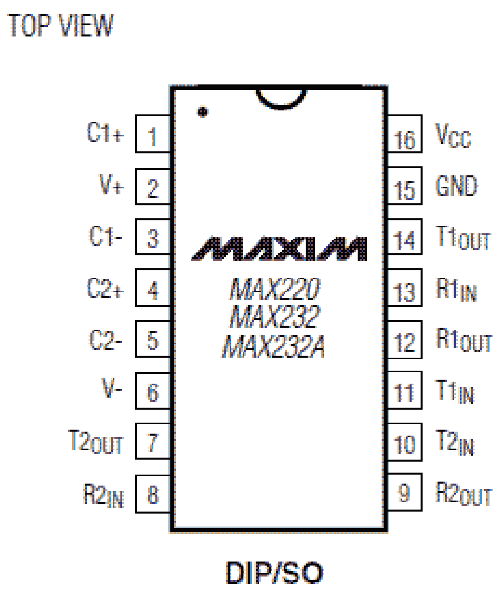
The MAX232(A) has two receivers (converts from RS-232 to TTL voltage levels), and two drivers (converts from TTL logic to RS-232 voltage levels). This means only two of the RS-232 signals can be converted in each direction. Typically, a pair of a driver/receiver of the MAX232 is used for TX and RX signals, and the second one for CTS and RTS signals.

There are not enough drivers/receivers in the MAX232 to also connect the DTR, DSR, and DCD signals. Usually these signals can be omitted when e.g. communicating with a PC's serial interface. If the DTE really requires these signals, either a second MAX232 is needed, or some other IC from the MAX232 family can be used. Also, it is possible to directly wire DTR (DB9 pin #4) to DSR (DB9 pin #6) without going through any circuitry. This gives automatic (brain dead) DSR acknowledgment of an incoming DTR signal.

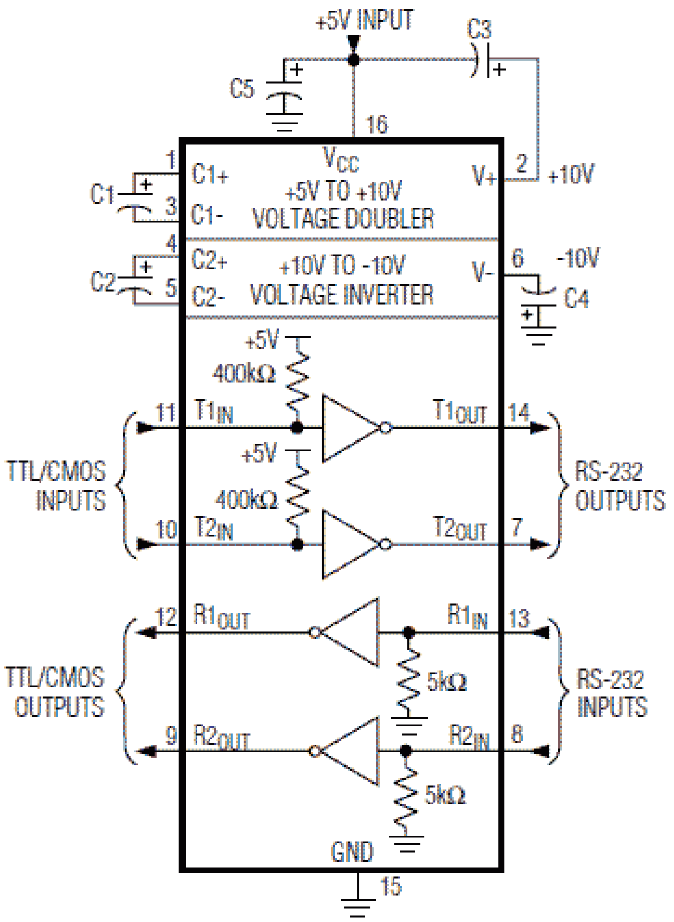
PIN DESCRIPTION



BLOCK DIAGRAM



CAPACITANCE (μF)					
DEVICE	C1	C2	C3	C4	C5
MAX220	0.047	0.33	0.33	0.33	0.33
MAX232	1.0	1.0	1.0	1.0	1.0
MAX232A	0.1	0.1	0.1	0.1	0.1



Diagrams continued in the full data sheet.