A Major Project report on

"STM32 BASED WIRELESS BLACKBOX FOR CARS USING SENSORS"

Submitted

In partial fulfillment of the Requirement for the award of the Degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING

By

J. HARSHINI	19W91A0482
K. SRIHITHA	19W91A04A2
G. NAVEEN KUMAR	20W95A0410

Under the Esteemed Guidance of

Mrs. A. JHANSI RANI

Associate Professor

Department of Electronics & Communication Engineering

MALLA REDDY INSTITUTE OF ENGINEERING AND TECHNOLOGY

Approved by A.I.C.T.E. - New Delhi, Affiliated to J.N.T. University Hyderabad

Maisammaguda, Dhulapally, Secunderabad-500 100

December-2022

Malla Reddy Institute of Engineering and Technology

(Sponsored by Malla Reddy Educational Society)

ISO 9001-2008 Certified institution, Affiliated to JNTU, Hyderabad

Maisammaguda, Dhulapally (Post via Hakimpet), Sec'Bad - 500 100.

Department of Electronics and Communication Engineering

CERTIFICATE

This is to certify that the Major project entitled "STM32 BASED WIRELESS BLACKBOX FOR CARS USING SENSORS" that is being submitted by J. HARSHINI (19W91A0482),K. SRIHITHA (19W91A04A2),G. NAVEEN KUMAR (20W95A0410) under the guidance of MRS. A. JHANSI RANI for the award of B.Tech Degree in ELECTRONICS AND COMMUNICATION ENGINEERINGfrom the MALLAREDDY INSTITUTE OF ENGINEERING & TECHNOLOGY, Maisammaguda (Affiliated to JNTU Hyderabad) is a record of Bonafide work carried out by them under our guidance and supervision. The results embodied in this Major project have not been submitted to any other university or institute for the award of any degree.

Project Guide	HOD
MRS. A. JHANSI RANI	Dr. P. SAMPATH KUMAR
Department of ECE	Department of ECE

EXTERNAL EXAMINER	PRINCIPAL
	Dr. M Ashok

Malla Reddy Institute of Engineering and Technology

(Sponsored by Malla Reddy Educational Society)

ISO 9001-2008 Certified institution, Affiliated to JNTU, Hyderabad

Maisammaguda, Dhulapally (Post via Hakimpet), Sec'Bad - 500 100.

Department of Electronics and Communication Engineering

DECLARATION

We, J. HARSHINI (19W91A0482), K. SRIHITHA (19W91A04A2), G. NAVEEN KUMAR (20W95A0410) hereby declare that the Major project entitled "STM32 BASED WIRELESS BLACKBOX FOR CARS USING SENSORS" is bonafide work done and submitted under the guidance of MRS. A. JHANSI RANI in partial fulfillment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in ELECTRONICS AND COMMUNICATION ENGINEERING.

DEPARTMENT OF ECE

J. HARSHINI	19W91A0482
K. SRIHITHA	19W91A04A2
G. NAVEEN KUMAR	20W95A0410

ACKNOWLEDGEMENT

We are very much thankful to Director, **Shri. P. PRAVEEN REDDY** for giving us this opportunity to do this major project. We express our deep sense of gratitude to him for his constant guidance and inspiring words.

We express our profound thanks to our Principal, **Dr. M. ASHOK,** for extending all the college facilities for the completion of the major project.

We would like to thank **Dr. PATTEM SAMPATH KUMAR** Professor and Head of the Department of Electronics and Communication Engineering for having provided the freedom to use all the facilities available in the department, especially the laboratories and the library, at anytime.

We feel highly obliged to our project coordinator **Mrs. A. JHANSI RANI (Ph.D),** Asst Professor and project guide **Mrs. A. JHANSI RANI (Ph.D),** Assistant Professor Department of Electronics and Communication Engineering for their constant encouragement and moral support. They have been a source of valuable guidance, suggestions and kindness during the course of the project work. We find no words to express our gratitude and thanks to them.

We sincerely thank all the staff of the Department of Electronics and Communication Engineering, for their timely suggestions, healthy criticism and motivation during the course of our study. We would also like to thank our friends for always being there to provide required help or support. With great respect and affection, we thank our parents who were the backbone behind our deeds.

Finally, we express our immense gratitude with pleasure to one and all who have either directly or indirectly contributed to our need at right time for the development and execution of project work

J. HARSHINI 19W91A082

K. SRIHITHA 19W91A04A2

G. NAVEEN KUMAR 20W95A0410

CONTENTS

S.NO	TOPIC		PAGE NO.
	COTENTS		I
	LIST OF FIGURES		Iv
	LIST OF TABLES		v
	ABSTRACT		1
CHAPTER - 1	**INTRODUCTION**		2
CHAPTER - 2	**LITERATURESURVEY**		3
	2.1	Existing System	3
	2.2	Proposed System	3
CHAPTER - 3	**HARDWARE REQUIRED**		5
	3.1	Power Supply	5
	3.2	Ultrasonic Sensor	6
	3.3	MEMS	8
	3.4	LCD (Liquid Cristal Display)	8
	3.4.1	Description Of 16x2	9
	3.5	Pin Description	12
	3.6	Control Lines	12
	3.7	Initialization By Instructions	16
	3.8	Flowchart	17
CHAPTER - 4	**GPS (GLOBAL POSITIONING SYSTEM)**		18
	4.1	Basic Concept Of GPS	18

	4.2	GPS Commands List	20
CHAPTER - 5	**GSM (Global System for Mobile communication)**		35
	5.1	Definition Of GSM	35
	5.2	History	35
	5.3	GSM Frequencies	35
	5.3.1	Mobile Telephony Standards	36
	5.4	Second Generation of Mobile Networks (2G)	36
	5.5	Introduction To the GSM Standard	38
	5.5.1	GSM Standards	38
	5.5.2	TDMA In Brief	39
	5.5.3	The Concept of Cellular Network	40
	5.5.4	Architecture Of the GSM Network	40
	5.6	Introduction To Modem	42
	5.7	Basic Concepts of SMS Technology	47
CHAPTER - 6	**TEMPERATURE SENSOR (LM35)**		48
	6.1	Precision Centigrade Temperature Sensor	48
	6.2	The Characteristic of This LM35 Sensor Is	49
CHAPTER - 7	**SOFTWARE REQUIRED**		50
	7.1	Arduino	50
	7.2	Board Types	51
	7.3	Arduino Family	54
	7.3.1	Block Diagram of Power Supply	56
	7.4	Description Of Power Supply	56

CHAPTER - 8	**DHT11 SENSOR**		57
	8.1	What Is A DHT11 Sensor	57
	8.1.1	Working Principle of DHT11 Sensor	57
CHAPTER - 9	**RELAYA**		59
	9.1	Basics On Relay Handling	59
	9.2	Pole And Throw	61
CHAPTER – 10	**WORKING**		64
CHAPTER - 11	**CODE**		65
CHAPTER - 12	**OUTPUT**		72
	FUTURESCOPE		75
	CONCLUSION		76
	REFERENCES		77

LIST OF FIGERS

S.NO	TITLE		PAGE NO.
1.	FIG 3.1	Breadboard power supply module	05
2.	FIG 3.2	Ultrasonic sensor	06
3.	FIG 3.4.1	Electrical Block Diagram	11
4.	FIG 3.4.2	power supply for LCD	11
5.	FIG 3.5	pin diagram of 1x16 lines LCD	12
6.	FIG 3.6	character details in LCD	15
7.	FIG 3.7	flow chart of lcd	16
8.	FIG 4.1	Position calculation	19
9.	FIG 5.5.2	TDMA	39
10.	FIG 5.5.4	Architecture of the GSM Network	41
11.	FIG 5.6.1	Modem	42
12.	FIG 5.6.2	GSM/GPRS modem	43
13.	FIG 5.6.3	Connection between PC and GSM	45
14.	FIG 7.1.1	Arduino Uno	51
15.	FIG 7.2.2	Board description	53
16.	FIG 7.3.1	Arduino Family	55
17.	FIG 7.3.2	Block Diagram of Power Supply	56
18.	FIG 7.3.3	Schematic Diagram of Power Supply	56
19.	FIG 8.1.2	DTH11 Sensor	58
20.	FIG 9.1	Remote control electrical switches	59
21.	FIG 9.1.1	Relay	60

LIST OF TABLES

S.NO		TITLE	PAGE NO.
1.	Tab 3.4.1	16 x 2 Alphanumeric LCD Module Specifications	10
2.	Tab 3.4.2	Address locations for a 1x16 line LCD	11
3.	Tab 3.5.1	Pin specifications	12
4.	Tab 5.3.1	Mobile telephone standards	36
5.	Tab 7.1	Atmega328 specifications	50
6.	Tab 7.2.1	Arduino boards based on ATMEGA32u4 microcontroller	52
7.	Tab 7.2.3	pin Board description	54

STM32 BASED WIRELESS BLACKBOX FOR CARS USING SENSORS

ABSTRACT

The main purpose of this wireless black box project is to develop a vehicle black box system that can be installed into any vehicle all over the world. This paradigm is often designed with minimum range of circuits. Wireless black box is basically a device that will indicate all the parameters of a vehicle crash and will also store and display its parameters such as temperature, location, vibration, alcohol limit etc. At the time of accident, the message will be sent from the system built inside the car to the registered mobile numbers such as emergency numbers of police stations, hospitals, family members, owner etc. We have used various types of sensors like temperature sensor, which is used to measure temperature. Vibration sensor measures vibrations felt by the car during accident. Alcohol sensor is located on the steering wheel which will indicate whether the driver is drunk. Gyroscope sensor is used to indicate tilt during the accident. GSM module, GPS module are some of the devices used in this project which helps in accomplishing the output.

Keywords- Arduino, Gas sensor, Temperature sensor, Vibration sensor, Accelerometer, GSM, GPS, SMS.

CHAPTER-1

INTRODUCTION

In today's world as the population increases day by day the numbers of vehicle also increase on the road and highways. This results in accident that leads to the traffic jams and people do not get the help instantaneously. Road accidents constitute the major part of the accident deaths all over the world. This takes a toll on the property as well as causes human life loss because of unavailability of immediate safety facilities. Complete accident prevention is unavoidable but atleast repercussions can be reduced. In highly populated Countries like India, everyday people lose their lives because of accidents and poor emergency facilities. These lives could have been saved if medical facilities are provided at the right time. In many situations the family members or the ambulance and police authority is not informed in time. This result in delaying the help reached to the person suffered due to accident. In order to give treatment for injured people, first we need to know where the accident happened through location tracking and then send a message to your related one or to the emergency services.

CHAPTER-2

LITERATURESURVEY

The BlackBox concept is derived from the aviation industry, a flight recorder, colloquially known as a Blackbox; although it is now orange coloured for easy search, is an electronic recording device placed in an aircraft for the purpose of facilitating the investigation of aviation accidents and incidents. Existing system is a device installed to record vehicle crashes or accidents. Electronically sensed problems in the engine or a sudden change in wheel speed trigger's this device. It do not track the vehicle movement or the driver's action and do record several types of important data only few seconds before the collision, due to the existing system is designed to monitor activation of airbag.

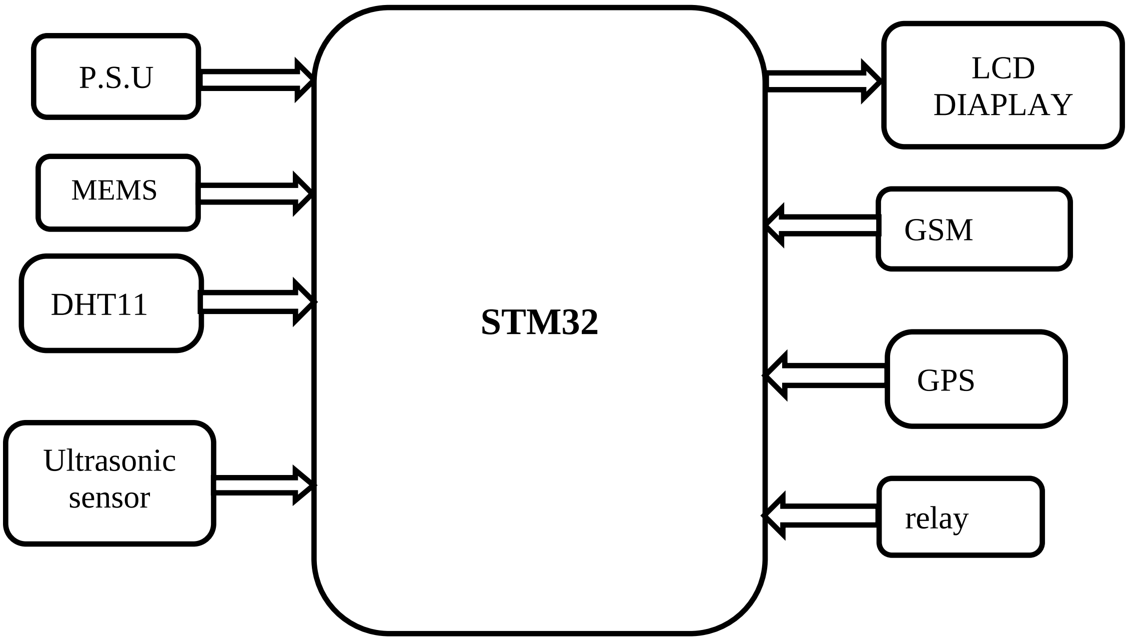
2.1 EXISTING SYSTEM

There are existing systems for wireless black boxes. For example, a Vehicle Black Box system based on 433MHz long-range wireless modulation technology has been developed for safe driving issues by utilizing gas and flames sensors[1]. Another example is a Smart Wireless Black Box with Intelligent Facial Recognition System for Prevention of Accidents and Theft of Vehicles Using Raspberry Pi Along with Sensors Based on IoT[3].

2.2 PROPOSED SYSTEM

There are several proposed systems for wireless black boxes. For example, one proposed system is a black box using a MEMS accelerometer sensor and GPS location tracking system for accidental monitoring. When an accident happens, a GSM sends the location of the vehicle to an authorized mobile phone[3]. Another proposed system is a Wireless black box that will indicate all the parameters of a vehicle crash and will also store and display its parameters every three seconds such as date, time, IR, MEMS, location, vibration, alcohol limit, etc[4].

Block Diagram:



Hardware Required:

- Power supply
- MEMS
- Ultrasonic sensor
- Lcd display
- GPS
- GSM
- LM32
- DHT11
- Relay

Software Required:

- ARDUINO IDE

CHAPTER-3

HARDWARE REQUIRED

3.1 POWER SUPPLY:

This is a 3.3V/5V MB102 Breadboard Power Supply Module which provides a dual 5 V and 3.3 V power rails and has a multi-purpose female USB socket.

The 3.3V/5V **MB102** Breadboard Power Supply Module securely fits in a standard 400 or 800 tie points breadboard it also features reverse polarity protection, the module can take 6.5V to 12V input and can produce 3.3V and +5V.

The module can also output 5V on USB connector or input through the USB connector. It's a must-have product for experimenters those have to test/prototype electronic circuits on the breadboard.

Features:

1. Breadboard power supply module, compatible with 5V, 3.3V.
2. Apply to MB102 breadboard
3. Fluctuation two road independent control can switch over to 0 V, 3.3 V, 5 V
4. On-board two groups of 3.3V, 5V DC output plug pin, convenient external lead use.

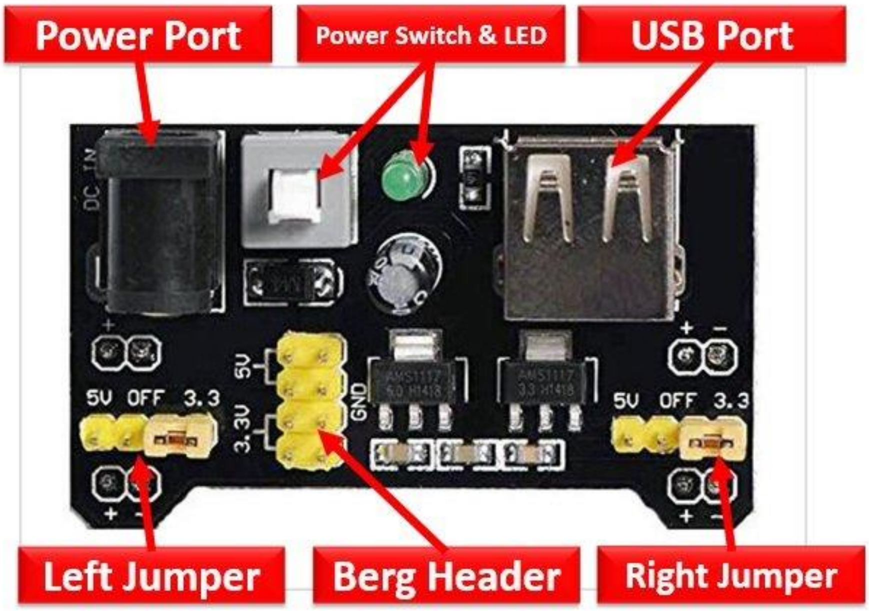


FIG 3.1. Breadboard power supply module

3.2 ULTRASONIC SENSOR

Ultrasonic sensors work on a principle similar to radar or sonar which evaluate attributes of a target by interpreting echoes from radio or sound waves. They generate high frequency sound waves and evaluate the echo which is received back by the sensor. This technology can be used to measure wind speed, fullness of a tank, speed through air or water, humidity, sonar, medical ultrasonography, burglar alarms, and non-destructive testing.

Systems typically use a transducer which generates sound waves in the ultrasonic range, above 18,000 hertz, by turning electrical energy into sound, then upon receiving the echo turn the sound waves into electrical energy which can be measured and displayed.

- VCC -> Arduino +5V pin
- GND -> Arduino GND pin
- Trig -> Arduino Digital Pin 2
- Echo -> Arduino Digital Pin 2

Fig 3.2. Ultrasonic sensor

Features:

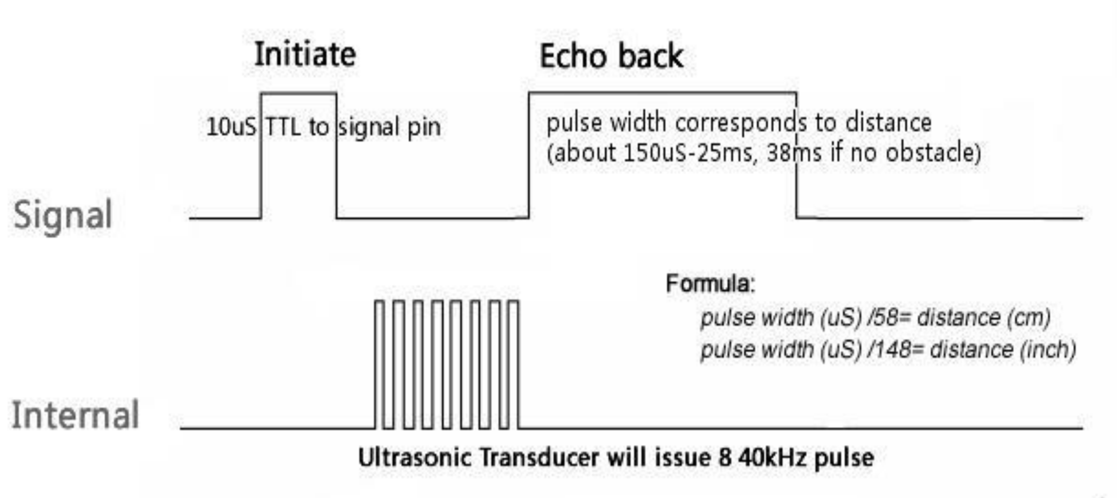
- Detecting range: 3cm-4m
- Best in 30-degree angle
- Electronic brick compatible interface
- 5VDC power supply
- Breadboard friendly
- Dual transducer
- Arduino library ready

▪ Specification:

Supply voltage	5V
Global Current Consumption	15 mA
Ultrasonic Frequency	40k Hz
Maximal Range	400 cm
Minimal Range	3 cm
Resolution	1 cm
Trigger Pulse Width	10 µs
Outline Dimension	43x20x15 mm

▪ Usage:

Hardware Installation



A short ultrasonic pulse is transmitted at the time 0, reflected by an object. The senor receives this signal and converts it to an electric signal. The next pulse can be transmitted when the echo is faded away. This time period is called cycle period. The recommend cycle period should be no less than 50ms. If a 10µs width trigger pulse is sent to the signal pin, the Ultrasonic module will output eight 40kHz ultrasonic signal and detect the echo back. The measured distance is proportional to the echo pulse width and can be calculated by the formula above. If no obstacle is detected, the output pin will give a 38ms high level signal.

3.3. MEMS:

MEMS stands for Microelectromechanical Systems. It is a technology that allows mechanical structures to be miniaturized and thoroughly integrated with electrical circuitry, resulting in a single physical device that is actually more like a system. The system indicates that mechanical components and electrical components are working together to implement the desired functionality. MEMS devices can range from several millimeters to less than one micrometer in size.

- **APPLICATIONS OF MEMS:**

MEMS technology has a wide range of applications. Some common commercial applications of MEMS include inkjet printers, which use piezoelectric or thermal bubble ejection to deposit ink on paper, accelerometers in modern cars for a large number of purposes including airbag deployment and electronic stability control, and inertial measurement units for navigation and motion sensing. MEMS can also be found in systems ranging across automotive, medical, electronic, communication, semiconductor, defines, aerospace, and more.

3.4. LCD (Liquid Cristal Display):

A liquid crystal display (LCD) is a thin, flat display device made up of color or monochrome pixels arrayed in front of a light source or reflector. It is inexpensive, easy to use, and can produce a readout using the 5X7 dots plus cursor of the display. For an 8-bit data bus, it requires a +5V supply plus 10 I/O lines, while for a 4-bit data bus it only requires the supply lines plus 6 extra lines.

Features:

1) Interface with either 4-bit or 8-bit microprocessor.
2) Display data RAM
3) 80x08 bits (80 characters).
4) Character generator ROM
5) 160 different 5007 dot-matrix character patterns.
6) Character generator RAM
7) different user programmed 5007 dot-matrix patterns.
8) Display data RAM and character generator RAM may be Accessed by the microprocessor.

9) Numerous instructions

10) Clear Display, Cursor Home, Display ON/OFF, Cursor ON/OFF, Blink Character, Cursor Shift, Display Shift.

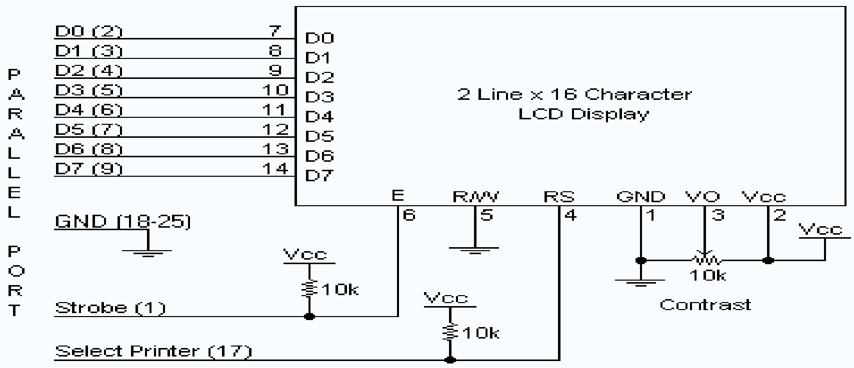
11) Built-in reset circuit is triggered at power ON.

12) Built-in oscillator.

3.4.1. Description Of 16x2:

This is the first interfacing example for the Parallel Port. We will start with something simple. This example doesn't use the Bi-directional feature found on newer ports, thus it should work with most, if no all Parallel Ports. It however doesn't show the use of the Status Port as an input. So what are we interfacing? A 16 Character x 2 Line LCD Module to the Parallel Port. These LCD Modules are very common these days, and are quite simple to work with, as all the logic required to run them is on board.

- **Schematic Diagram:**



- Above is the quite simple schematic. The LCD panel's Enable and Register Select is connected to the Control Port. The Control Port is an open collector / open drain output. While most Parallel Ports have internal pull-up resistors, there are a few which don't. Therefore by incorporating the two 10K external pull up resistors, the circuit is more portable for a wider range of computers, some of which may have no internal pull up resistors.
- We make no effort to place the Data bus into reverse direction. Therefore we hard wire the R/W line of the LCD panel, into write mode. This will cause no bus conflicts on the data lines. As a result we cannot read back the LCD's internal Busy Flag which tells us if the LCD has accepted and finished processing the last instruction. This problem is overcome by inserting known delays into our program.

- The 10k Potentiometer controls the contrast of the LCD panel. Nothing fancy here. As with all the examples, I've left the power supply out. You can use a bench power supply set to 5v or use a onboard +5 regulator. Remember a few de-coupling capacitors, especially if you have trouble with the circuit working properly.

16 x 2 Alphanumeric LCD Module Features:

- Intelligent, with built-in Hitachi HD44780 compatible LCD controller and RAM providing simple interfacing
- 61 x 15.8 mm viewing area
- 5 x 7 dot matrix format for 2.96 x 5.56 mm characters, plus cursor line
- Can display 224 different symbols
- Low power consumption (1 mA typical)
- Powerful command set and user-produced characters
- TTL and CMOS compatible
- Connector for standard 0.1-pitch pin headers

Pin	Symbol	Level	Function
1	V_{SS}	-	Power, GND
2	V_{DD}	-	Power, 5V
3	Vo	-	Power, for LCD Drive
4	RS	H/L	Register Select Signal H: Da Input L: Instruction Input
5	R/W	H/L	H: Data Read (LCD->MPU) L: Data Write (MPU->LCD)
6	E	H,H->L	Enable
7-14	DB0-DB7	H/L	Data Bus; Software selectable 4- or 8-bit mode
15	NC	-	NOT CONNECTED
16	NC	-	NOT CONNECTED

Table 3.4.1: 16 x 2 Alphanumeric LCD Module Specifications

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

Data can be placed at any location on the LCD. For 16×1 LCD, the address locations are:

POSITION		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
ADDRESS	LINE1	00	01	02	03	04	05	06	07	40	41	42	43	44	45	46	47

Table 3.4.2: Address locations for a 1x16 line LCD

Character-based modules are available in a variety of shapes and sizes, with line lengths of 8,16,20,24,32 and 40 characters. Different LC technologies exist, such as "supertwist" which offers improved contrast and viewing angle. Back lighting is available, either electro-luminescent or LED.

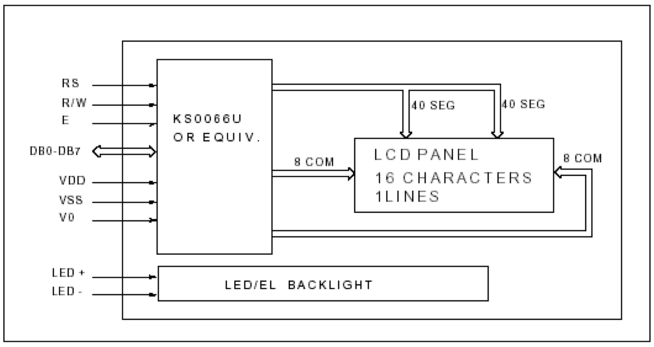


Fig 3.4.1.: Electrical Block Diagrm

Power supply for LCD driving:

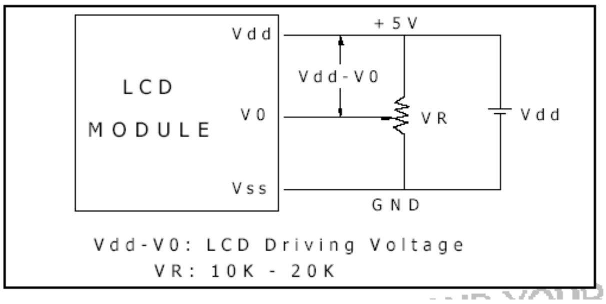


Fig 3.4.2: power supply for LCD

3.5 PIN DESCRIPTION:

Most LCDs with 1 controller have 14 Pins and LCDs with 2 controller has 16 Pins (two pins are extra in both for back-light LED connections).

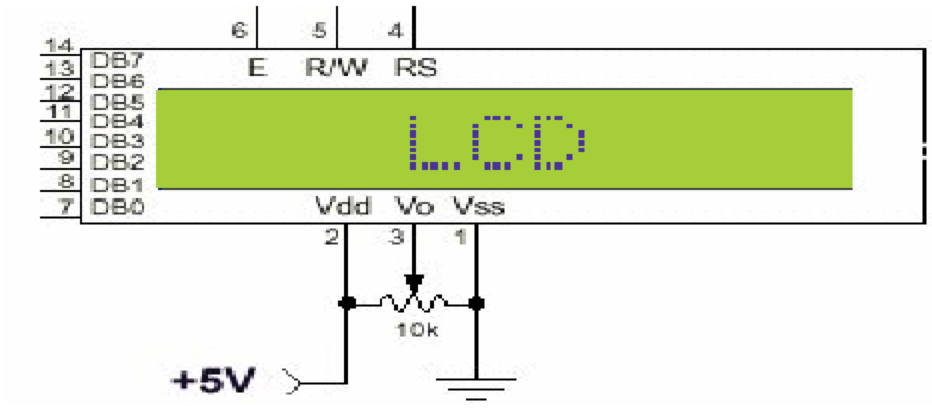


Fig 3.5: pin diagram of 1x16 lines LCD

PIN	SYMBOL	FUNCTION
1	Vss	Power Supply(GND)
2	Vdd	Power Supply(+5V)
3	Vo	Contrast Adjust
4	RS	Instruction/Data Register Select
5	R/W	Data Bus Line
6	E	Enable Signal
7-14	DB0-DB7	Data Bus Line
15	A	Power Supply for LED B/L(+)
16	K	Power Supply for LED B/L(-)

Table 3.5.1: Pin specifications

3.6 CONTROL LINES:

EN: Line is called "Enable." This control line is used to tell the LCD that you are sending it data. To send data to the LCD, your program should make sure this line is low (0) and then set the other two control lines and/or put data on the data bus. When the other lines are completely ready, bring EN high (1) and wait for the minimum amount of time required by the LCD datasheet (this varies from LCD to LCD), and end by bringing it low (0) again.

RS:

Line is the "Register Select" line. When RS is low (0), the data is to be treated as a command or special instruction (such as clear screen, position cursor, etc.). When RS is high (1), the data being sent is text data which sould be displayed on the screen. For example, to display the letter "T" on the screen you would set RS high.

RW:

Line is the "Read/Write" control line. When RW is low (0), the information on the data bus is being written to the LCD. When RW is high (1), the program is effectively querying (or reading) the LCD. Only one instruction ("Get LCD status") is a read command. All others are write commands, so RW will almost always be low.

Finally, the data bus consists of 4 or 8 lines (depending on the mode of operation selected by the user). In the case of an 8-bit data bus, the lines are referred to as DB0, DB1, DB2, DB3, DB4, DB5, DB6, and DB7.

Logic status on control lines:

• E - 0 Access to LCD disabled

- 1 Access to LCD enabled

• R/W - 0 Writing data to LCD

- 1 Reading data from LCD

• RS - 0 Instructions

Writing data to the LCD:

1) Set R/W bit to low
2) Set RS bit to logic 0 or 1 (instruction or character)
3) Set data to data lines (if it is writing)
4) Set E line to high
5) Set E line to low

Read data from data lines (if it is reading) on LCD:

1) Set R/W bit to high
2) Set RS bit to logic 0 or 1 (instruction or character)
3) Set data to data lines (if it is writing)
4) Set E line to high
5) Set E line to low

Entering Text:

First, a little tip: it is manually a lot easier to enter characters and commands in hexadecimal rather than binary (although, of course, you will need to translate commands from binary couple of sub-miniature hexadecimal rotary switches is a simple matter, although a little bit into hex so that you know which bits you are setting). Replacing the d.i.l. switch pack with a of re-wiring is necessary.

SWITCHES:

The switches must be the type where On = 0, so that when they are turned to the zero position, all four outputs are shorted to the common pin, and in position "F", all four outputs are open circuit.

All the available characters that are built into the module are shown in Table 3. Studying the table, you will see that codes associated with the characters are quoted in binary and hexadecimal, most significant bits ("left-hand" four bits) across the top, and least significant bits ("right-hand" four bits) down the left.

Using the switches, of whatever type, and referring to Table 3, enter a few characters onto the display, both letters and numbers. The RS switch (S10) must be "up" (logic 1) when sending the characters, and switch E (S9) must be pressed for each of them. Thus the operational order is: set RS high, enter character, trigger E, leave RS high, enter another character, trigger E, and so on.

The first 16 codes in Table 3, 00000000 to 00001111, ($00 to $0F) refer to the CGRAM. This is the Character Generator RAM (random access memory), which can be used to hold user-defined graphics characters. This is where these modules really start to show their potential,

offering such capabilities as bar graphs, flashing symbols, even animated characters. Before the user-defined characters are set up, these codes will just bring up strange looking symbols.

Codes 00010000 to 00011111 ($10 to $1F) are not used and just display blank characters. ASCII codes "proper" start at 00100000 ($20) and end with 01111111 ($7F). Codes 10000000 to 10011111 ($80 to $9F) are not used, and 10100000 to 11011111 ($A0 to $DF) are the Japanese characters.

Fig 3.6: character details in LCD

3.7 Initialization by Instructions:

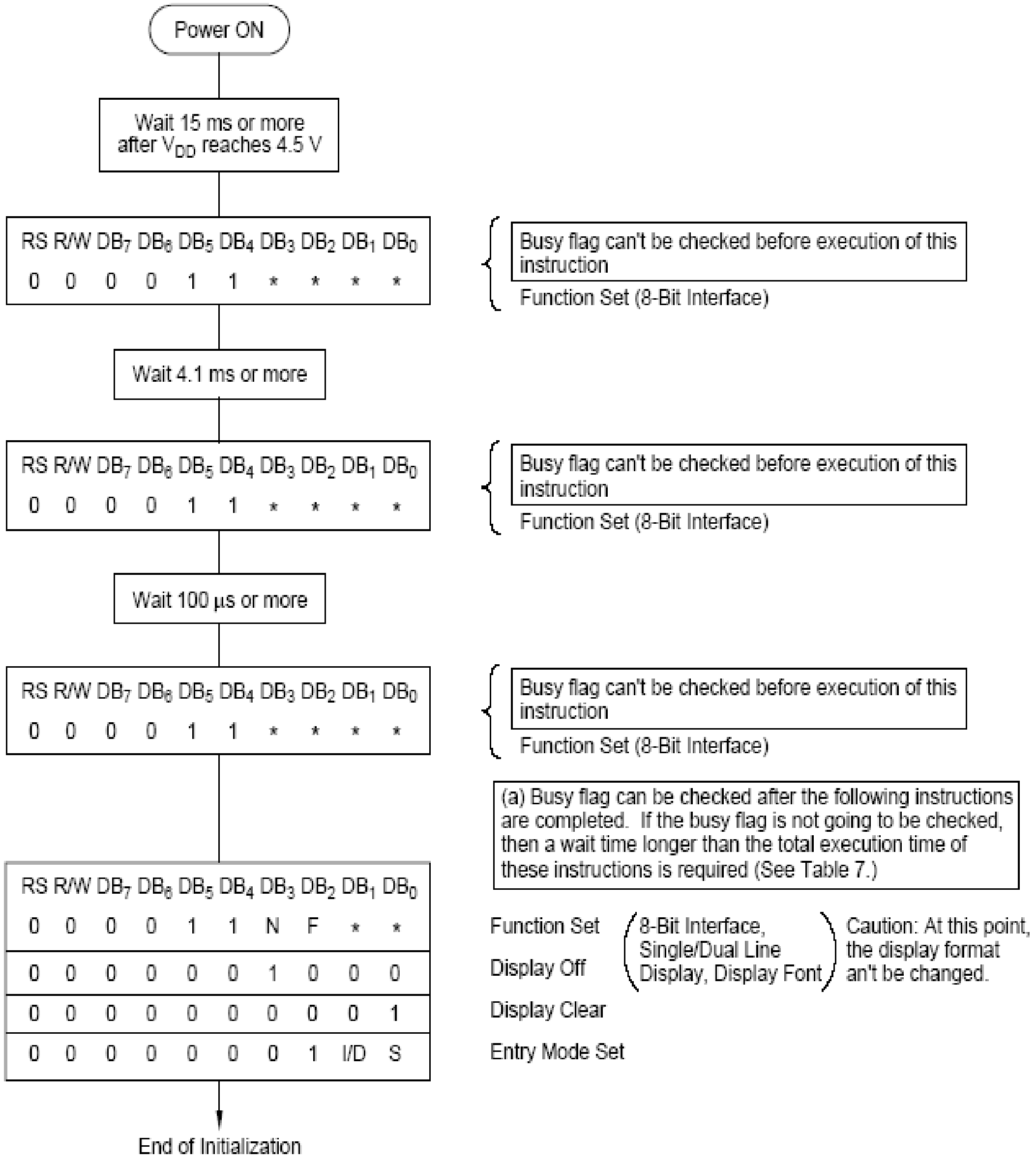
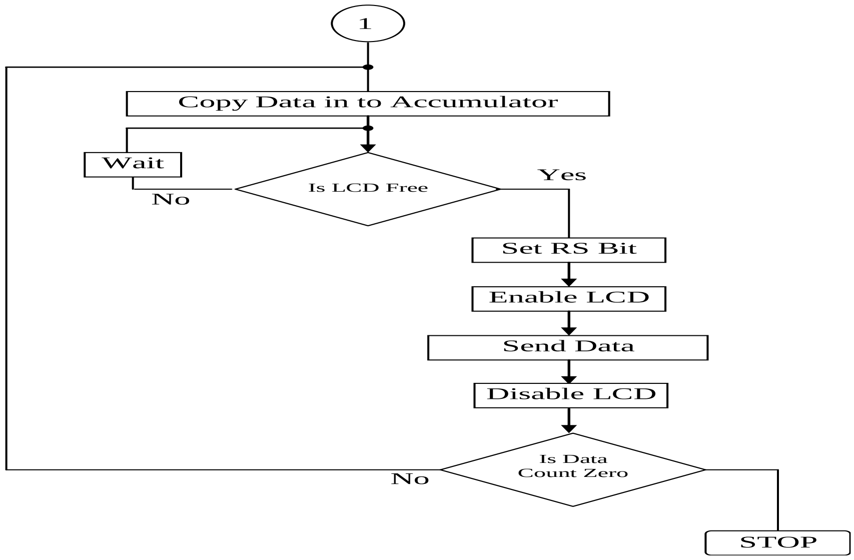
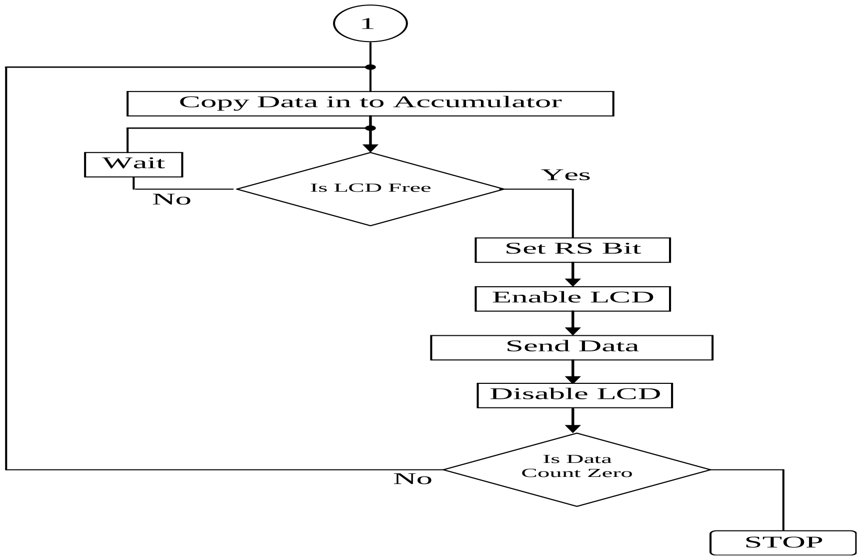


Fig 3.7: flow chart of lcd

If the power conditions for the normal operation of the internal reset circuit are not satisfied, then executing a series of instructions must initialize LCD unit. The procedure for this initialization process is as above show.

3.8 FLOWCHART:





CHAPTER-4

GPS (GLOBAL POSITIONING SYSTEM)

4.1 Basic concept of GPS

A GPS receiver calculates its position by precisely timing the signals sent by GPS satellites high above the Earth. Each satellite continually transmits messages that include

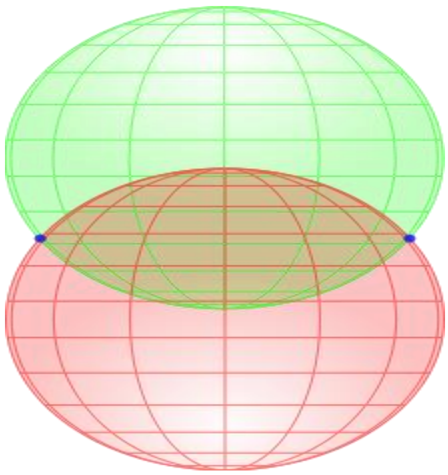
- the time the message was transmitted
- precise orbital information (the ephemeris)
- the general system health and rough orbits of all GPS satellites (the almanac).

The receiver uses the messages it receives to determine the transit time of each message and computes the distance to each satellite. These distances along with the satellites' locations are used with the possible aid of trilateration, depending on which algorithm is used, to compute the position of the receiver. Many GPS units show derived information such as direction and speed, calculated from position changes.

Three satellites might seem enough to solve for position since space has three dimensions and a position near the Earth's surface can be assumed. However, even a very small clock error multiplied by the very large speed of light[35] — the speed at which satellite signals propagate — results in a large positional error. Therefore, receivers use four or more satellites to solve for the receiver's location and time. The very accurately computed time is effectively hidden by most GPS applications, which use only the location. A few specialized GPS applications do however use the time; these include time transfer, traffic signal timing, and synchronization of cell phone base stations.

Although four satellites are required for normal operation, fewer apply in special cases.For example, a ship or aircraft may have known elevation. Some GPS receivers may use additional clues or assumptions (such as reusing the last known altitude, dead reckoning, inertial navigation, or including information from the vehicle computer) to give a less accurate (degraded) position when fewer than four satellites are visible.

- **Position calculation introduction**



Two sphere surfaces intersecting in a circle

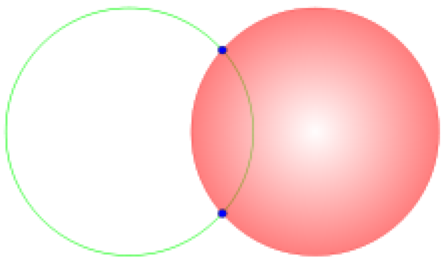


Fig 4.1: Position calculation

Surface of sphere intersecting a circle (not a solid disk) at two points

To provide an introductory description of how a GPS receiver works, error effects are deferred to a later section. Using messages received from a minimum of four visible satellites, a GPS receiver is able to determine the times sent and then the satellite positions corresponding to these times sent. The x, y, and z components of position, and the time sent, are designated as $[x_i, y_i, z_i, t_i]$where the subscript i is the satellite number and has the value 1, 2, 3, or 4. Knowing the indicated time the message was received $\tilde{t}_r$, the GPS receiver could compute the transit time of the message as $(\tilde{t}_r - t_i)$, if $\tilde{t}_r$would be equal to correct reception time, t_r. A pseudo range, $p_i \triangleq (\tilde{t}_r - t_i)c$, would be the traveling distance of the message, assuming it traveled at the speed of light, c.

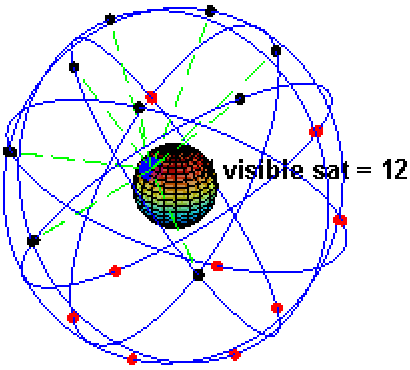
A satellite's position and pseudorange define a sphere, centered on the satellite, with radius equal to the pseudorange. The position of the receiver is somewhere on the surface of this sphere. Thus, with four satellites, the indicated position of the GPS receiver is at or near the intersection of the surfaces of four spheres. In the ideal case of no errors, the GPS receiver would be at a precise intersection of the four surfaces.

If the surfaces of two spheres intersect at more than one point, they intersect in a circle. The distance between these two points is the diameter of the circle of intersection. Another figure, Surface of Sphere Intersecting a Circle (not a solid disk) at Two Points, illustrates the intersection. The two intersections are marked with dots. Again, the article trilateration clearly shows this mathematically.

For automobiles and other near-earth vehicles, the correct position of the GPS receiver is the intersection closest to the Earth's surface.For space vehicles, the intersection farthest from Earth may be the correct one.

The correct position for the GPS receiver is also the intersection closest to the surface of the sphere corresponding to the fourth satellite.

4.2 GPS COMMANDS LIST



NMEA entry A NMEA input mode is also supported by some devices. This mode offers a standardized method to update or add waypoint and route data, despite the fact that not many programmers support it. The unit will receive the data and add to or replace the information already stored in memory because there is no handshaking or commanding involved in NMEA mode. When receiving NMEA input, the receiver saves data based on how the sentence is understood. While some receivers accept conventional NMEA input, this cannot be used to transmit a command to the unit; rather, it can only be used to update a waypoint or other similar duty. Commands could be sent using secret input phrases. The Magellan upload and download maintenance protocol supports a modified WPL message that includes comments, altitude, and icon data because it is based on NMEA phrases. Depending on the DPT phrase or the MTW to read the water temperature, some marine equipment may accept input for alerts such as deep or shallow water. For instance, the Garmin Map76 accepts inputs for DPT, MTW (temperature), and VHW (speed). Initialization from NMEA input could be used by other devices.

GGA - essential fix data which provide 3D location and accuracy data.

$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47

Where:
GGA-Global Positioning System Fix Data
123519-Fix taken at 12:35:19 UTC
4807.038, N-Latitude 48 deg 07.038' N
01131.000,E-Longitude 11 deg 31.000' E

1 Fix quality: 0 = invalid
1 = GPS fix (SPS)
2 = DGPS fix
3 = PPS fix
4 = Real Time Kinematic
5 = Float RTK
6 = estimated (dead reckoning) (2.3 feature)
7 = Manual input mode
8 = Simulation mode

08 Number of satellites being tracked

0.9 Horizontal dilution of position

545.4,M Altitude, Meters, above mean sea level

46.9,M Height of geo id (mean sea level) above WGS84

Time since the most recent DGPS update, in seconds (empty field) Station ID for DGPSThe altitude should be questioned if the geoid's height is lacking. Instead of reporting geoid altitude, some non-standard implementations report altitude with regard to the ellipsoid. Some devices don't even mention negative heights. The only sentence that mentions altitude is this one. GPS DOP and operational satellites are GSA. The nature of the repair is described in further detail in this clause. It provides the DOP and the numbers of the satellites currently utilised in the solution. DOP (dilution of precision) is a measure of how the fix's accuracy is impacted by satellite geometry. It is a number without units, where smaller is preferable. Using 4 satellites, a 3D fix

$GPGSA,A,3,04,05,,09,12,,,24,,,,,2.5,1.3,2.1*39

Where:

GSA Satellite status
A Auto selection of 2D or 3D fix (M = manual)
3 3D fix - values include: 1 = no fix
2 = 2D fix
3 = 3D fix
04,05... PRNs of satellites used for fix (space for 12)
2.5 PDOP (dilution of precision)
1.3 Horizontal dilution of precision (HDOP)
2.1 Vertical dilution of precision (VDOP)
*39 the checksum data, always begins with *

GSV - Satellites in View shows data about the satellites that the unit might be able to find based on its viewing mask and almanac data. It also shows current ability to track this data. Note that one GSV sentence only can provide data for up to 4 satellites and thus there may need to be 3 sentences for the full information. It is reasonable for the GSV sentence to contain more satellites than GGA might indicate since GSV may include satellites that are not used as part of the solution. It is not a requirment that the GSV sentences all appear in sequence. To avoid overloading the data bandwidth some receivers may place the various sentences in totally different samples since each sentence identifies which one it is..
SNR, or Signal to Noise Ratio, is a useful indirect value in the NMEA standard, ranging from 0 to 99 dB. It is an indirect but more useful value than raw signal strength. Manufacturers send different ranges of numbers with different starting numbers, so the values cannot be used to evaluate different units. The range of working values in a GPS typically shows a difference of 25 to 35 between the lowest and highest values, with 0 being a special case.

$GPGSV,2,1,08,01,40,083,46,02,17,308,41,12,07,344,39,14,22,228,45*75

Where:

GSV Satellites in view
2 Number of sentences for full data
1 sentence 1 of 2
08 Number of satellites in view

01 Satellite PRN number

40 Elevation, degrees
083 Azimuth, degrees
46 SNR - higher is better
for up to 4 satellites per sentence
*75 the checksum data, always begins with *

RMC - NMEA has its own version of essential gps pvt (position, velocity, time) data. It is called RMC, The Recommended Minimum, which will look similar to:

$GPRMC,123519,A,4807.038,N,01131.000,E,022.4,084.4,230394,003.1,W*6A

Where:
RMC Recommended Minimum sentence C
123519 Fix taken at 12:35:19 UTC
A Status A=active or V=Void.
4807.038,N Latitude 48 deg 07.038' N
01131.000,E Longitude 11 deg 31.000' E
022.4 Speed over the ground in knots
084.4 Track angle in degrees True
230394 Date - 23rd of March 1994
003.1,W Magnetic Variation
*6A The checksum data, always begins with *

Note that, as of the 2.3 release of NMEA, there is a new field in the RMC sentence at the end just prior to the checksum. For more information on this field see here.

GLL - Geographic Latitude and Longitude is a holdover from Loran data and some old units may not send the time and data active information if they are emulating Loran data. If a gps is emulating Loran data they may use the LC Loran prefix instead of GP.

$GPGLL,4916.45,N,12311.12,W,225444,A,*1D

Where:
GLL Geographic position, Latitude and Longitude
4916.46,N Latitude 49 deg. 16.45 min. North
12311.12,W Longitude 123 deg. 11.12 min. West
225444 Fix taken at 22:54:44 UTC
A Data Active or V (void)
*iD checksum data

Note that, as of the 2.3 release of NMEA, there is a new field in the GLL sentence at the end just prior to the checksum. For more information on this field see here.

VTG - Velocity made good. The gps receiver may use the LC prefix instead of GP if it is emulating Loran output.

$GPVTG,054.7,T,034.4,M,005.5,N,010.2,K*48

where:

VTG Track made good and ground speed
054.7,T True track made good (degrees)
034.4,M Magnetic track made good
005.5,N Ground speed, knots
010.2,K Ground speed, Kilometers per hour
*48 Checksum

Note that, as of the 2.3 release of NMEA, there is a new field in the VTG sentence at the end just prior to the checksum. For more information on this field see here.

Receivers that don't have a magnetic deviation (variation) table built in will null out the Magnetic track made good.
Decode of some Navigation Sentences

WPL - Waypoint Location data provides essential waypoint data. It is output when navigating to indicate data about the destination and is sometimes supported on input to redefine a waypoint location. Note that waypoint data as defined in the standard does not define altitude, comments, or icon data. When a route is active, this sentence is sent once for each waypoint in the route, in sequence. When all waypoints have been reported, the RTE sentence is sent in the next data set. In any group of sentences, only one WPL sentence, or an RTE sentence, will be sent.

$GPWPL,4807.038,N,01131.000,E,WPTNME*5C

With an interpretation of:

WPL Waypoint Location
4807.038,N Latitude
01131.000,E Longitude
WPTNME Waypoint Name
*5C The checksum data, always begins with *

AAM - Waypoint Arrival Alarm is generated by some units to indicate the Status of arrival (entering the arrival circle, or passing the perpendicular of the course line) at the destination waypoint.

$GPAAM,A,A,0.10,N,WPTNME*32

Where:
AAM Arrival Alarm
A Arrival circle entered
A Perpendicular passed
0.10 Circle radius
N Nautical miles
WPTNME Waypoint name
*32 Checksum data

APB - Autopilot format B is sent by some gps receivers to allow them to be used to control an autopilot unit. This sentence is commonly used by autopilots and contains navigation receiver warning flag status, cross-track-error, waypoint arrival status, initial bearing from origin waypoint to the destination, continuous bearing from present position to destination and recommended heading-to-steer to destination waypoint for the active navigation leg of the journey.

Note: some autopilots, Robertson in particular, misinterpret "bearing from origin to destination" as "bearing from present position to destination". This is likely due to the difference between the APB sentence and the APA sentence. for the APA sentence this would be the correct thing to do for the data in the same field. APA only differs from APB in this one field and APA leaves off the last two fields where this distinction is clearly spelled out. This will result in poor performance if the boat is sufficiently off-course that the two bearings are different.

$GPAPB,A,A,0.10,R,N,V,V,011,M,DEST,011,M,011,M*3C

where:

APB   Autopilot format B
A   Loran-C blink/SNR warning, general warning
A   Loran-C cycle warning
0.10   cross-track error distance
R   steer Right to correct (or L for Left)
N   cross-track error units - nautical miles (K for kilometers)
V   arrival alarm - circle
V   arrival alarm - perpendicular
011,M   magnetic bearing, origin to destination
DEST   destination waypoint ID
011,M   magnetic bearing, present position to destination
011,M   magnetic heading to steer (bearings could True as 033,T)

BOD - Bearing - Origin to Destination shows the bearing angle of the line, calculated at the origin waypoint, extending to the destination waypoint from the origin waypoint for the active navigation leg of the journey.

$GPBOD,045.,T,023.,M,DEST,START*01

where:

BOD   Bearing - origin to destination waypoint
045.,T   bearing 045 True from "START" to "DEST"
023.,M   bearing 023 Magnetic from "START" to "DEST"
DEST   destination waypoint ID
START   origin waypoint ID
*01   checksum

BWC - Bearing & Distance to Waypoint using a Great Circle route. Time (UTC) and distance & bearing to, and location of, a specified waypoint from present position along the great circle path.

$GPBWC,225444,4917.24,N,12309.57,W,051.9,T,031.6,M,001.3,N,004*29

where:

BWC   Bearing and distance to waypoint - great circle

225444 UTC time of fix 22:54:44
4917.24,N Latitude of waypoint
12309.57,W Longitude of waypoint
051.9,T Bearing to waypoint, degrees true
031.6,M Bearing to waypoint, degrees magnetic
001.3,N Distance to waypoint, Nautical miles
004 Waypoint ID
*29 checksum

RMB - The recommended minimum navigation sentence is sent whenever a route or a goto is active. On some systems it is sent all of the time with null data. The Arrival alarm flag is similar to the arrival alarm inside the unit and can be decoded to drive an external alarm. Note the use of leading zeros in this message to preserve the character spacing. This is done, I believe, because some autopilots may depend on exact character spacing.

$GPRMB,A,0.66,L,003,004,4917.24,N,12309.57,W,001.3,052.5,000.5,V*20

where:

RMB Recommended minimum navigation information
A Data status A = OK, V = Void (warning)
0.66,L Cross-track error (nautical miles, 9.99 max),
steer Left to correct (or R = right)
003 Origin waypoint ID
004 Destination waypoint ID
4917.24,N Destination waypoint latitude 49 deg. 17.24 min. N
12309.57,W Destination waypoint longitude 123 deg. 09.57 min. W
001.3 Range to destination, nautical miles (999.9 max)
052.5 True bearing to destination
000.5 Velocity towards destination, knots
V Arrival alarm A = arrived, V = not arrived
*20 checksum

RTE - RTE is sent to indicate the names of the waypoints used in an active route. There are two types of RTE sentences. This route sentence can list all of the waypoints in the entire route or it can list only those still ahead. Because an NMEA sentence is limited to 80 characters there may need to be multiple sentences to identify all of the waypoints. The data about the waypoints themselves will be sent in subsequent WPL sentences which will be sent in future cycles of the NMEA data.

$GPRTE,2,1,c,0,W3IWI,DRIVWY,32CEDR,32-29,32BKLD,32-I95,32-US1,BW-32,BW-198*69

$GPXTE,A,A,0.67,L,N*6F

Where:

XTE Cross track error, measured
A General warning flag V = warning
(Loran-C Blink or SNR warning)
A Not used for GPS (Loran-C cycle lock flag)
0.67 cross track error distance

L Steer left to correct error (or R for right)
N Distance units - Nautical miles
*6F checksum

Other sentences that may be useful

ALM - GPS Almanac Data contains GPS week number, satellite health and the complete almanac data for one satellite. Multiple messages may be transmitted, one for each satellite in the GPS constellation, up to maximum of 32 messages. Note that these sentences can take a long time to send so they are not generally sent automatically by the gps receiver. (Sorry I don't have an exact example of the sentence.) Note that this sentence breaks the 80-character rule. Also note that this sentence is often accepted as input so that you can preload a new almanac in a receiver.

```
$GPALM,A.B,C.D,E,F,hh,hhhh,...
```

Where:

ALM Almanac Data being sent
A Total number of messages
B Message number
C Satellite PRN number
D GPS week number (0-1023)
E Satellite health (bits 17-24 of message)
F eccentricity
hh t index OA, almanac reference time
hhhh sigma index 1, inclination angle
... OMEGADOT rate of right ascension
SQRA(A) root of semi-major axis
Omega, argument of perigee
Omega index 0, longitude of ascension node
M index 0, mean anomaly
a index f0, clock parameter
a index f1, clock parameter

HCHDG - Compass output is used on Garmin etrex summit, vista , and 76S receivers to output the value of the internal flux-gate compass. Only the magnetic heading and magnetic variation is shown in the message.

```
$HCHDG,101.1,,,7.1,W*3C
```

where:

HCHDG Magnetic heading, deviation, variation
101.1 heading
,, deviation (no data)
7.1,W variation

ZDA - Data and Time

```
$GPZDA,hhmmss.ss,dd,mm,yyyy,xx,yy*CC
$GPZDA,201530.00,04,07,2002,00,00*60
```

where:
hhmmss HrMinSec(UTC)
dd,mm,yyy Day,Month,Year
xx local zone hours -13..13
yy local zone minutes 0..59
*CC checksum

MSK - Control for a Beacon Receiver

$GPMSK,318.0,A,100,M,2*45

where:
318.0 Frequency to use
A Frequency mode, A=auto, M=manual
100 Beacon bit rate
M Bitrate, A=auto, M=manual
2 frequency for MSS message status (null for no status)
*45 checksum

MSS - Beacon Receiver Status

$GPMSS,55,27,318.0,100,*66

where:
55 signal strength in dB
27 signal to noise ratio in dB
318.0 Beacon Frequency in KHz
100 Beacon bitrate in bps
*66 checksum

Proprietary Sentences

Proprietary sentences can either be output from the gps or used as input to control information. They always start with P which is followed by a 3 character manufactures code and additional characters to define the sentence type.
Garmin

The following are Garmin proprietary sentences. "P" denotes proprietary, "GRM" is Garmin's manufacturer code, and "M" or "Z" indicates the specific sentence type. Note that the PGRME sentence is not set if the output is set to NMEA 1.5 mode.

$PGRME,15.0,M,45.0,M,25.0,M*1C

where:
15.0,M Estimated horizontal position error in meters (HPE)
45.0,M Estimated vertical error (VPE) in meters
25.0,M Overall spherical equivalent position error

$PGRMZ,93,f,3*21

where:

93,f Altitude in feet

3 Position fix dimensions 2 = user altitude

3 = GPS altitude

This sentence shows in feet, regardless of units shown on the display.
Note that for units with an altimeter this will be altitude computed
by the internal altimeter.

$PGRMM,NAD27 Canada*2F

Currently active horizontal datum

PSLIB

Proprietary sentences are used to control a Starlink differential beacon receiver. (Garmin's DBR is Starlink compatible as are many others.) When the GPS receiver is set to change the DBR frequency or b/s rate, the "J" sentence is replaced (just once) by (for example): $PSLIB,320.0,200*59 to set the DBR to 320 KHz, 200 b/s.

$PSLIB,,,J*22 Status request

$PSLIB,,,K*23 configuration request

These two sentences are normally sent together in each group of sentences from the GPS. The three fields are: Frequency, bit Rate, Request Type. The value in the third field may be: J = status request, K = configuration request, or null (blank) = tuning message. The correct values for frequency range from 283.5-325.0 KHz while the bit rate can be set to 0, 25, 50, 100 or 200 bps.

Magellan

Magellan uses proprietary sentences to do all of their waypoint and route maintenance. They use the MGN prefix for their sentences. This use is documented in their interface specification and will not be repeated here. Here is an example of a sentence sent by the GPS Companion product:

$PMGNST,02.12,3,T,534,05.0,+03327,00*40

where:

ST status information

02.12 Version number?

3 2D or 3D

T True if we have a fix False otherwise

534 numbers change - unknown

05.0 time left on the gps battery in hours

+03327 numbers change (freq. compensation?)

00 PRN number receiving current focus

*40 checksum

A tracklog on a Meridian is made up of propretary sentences that look like:

$PMGNTRK,4322.061,N,07948.473,W,00116,M,173949.42,A,,020602*67
$PMGNTRK,4322.058,N,07948.483,W,00090,M,174202.45,A,,020602*69.

where

TRK Tracklog
4322.071 Latitude
N North or South
07948.473 Longitude
W East or West
00116 Altitude
M Meters or Feet
173949.42 UTC time
A Active or Void
,, Track Name
020602 date
*67 checksum

Motorola
The PMOTG is used by Motorola Oncore receivers to send a command to the receiver. This command is used to set the output of the sentence to a particular frequency in seconds (or to 0) or to switch the output formula to motorola binary, gps, or loran.

$PMOTG,xxx,yyyy

where:
xxx the sentence to be controlled
yyyy the time interval (0-9999 seconds)

or $PMOTG,FOR,y

where:
y MPB=0, GPS=1, Loran=2

Rockwell International

The Rockwell chipset is used on a number of gps receivers. It outputs some proprietary sentences with the PRWI prefix and accepts input from some special sentences similar to the approach used by Magellan. It can also be switched to a separate binary mode using a proprietary sentence. The input sentence most used to initialize the unit is $PRWIINIT and one output sentence is $PRWIRID

$PRWIRID,12,01.83,12/15/97,0003, *42

Note: Commas may be used to signify using existing data. If units are supplied then the data must be present. Speed and direction must be supplied together. Lat/Lon must be supplied together. UTC time and date must be supplied together. If heading is magnetic then lat/lon needs to be supplied along with UTC time and date.

The sentences available for the Rockwell Jupiter chipset are: GGA, GSA, GSV, VTG, RMC and some proprietary sentences.

SiRF

The SiRF chips offer several input sentences for customization, including a binary protocol that allows different implementations to behave differently. These sentences require a fixed number of input fields, no null fields, and a checksum. Sentences 100 and 102 set serial ports, with 100 setting main port A and 102 setting DGPS input port B. An extra field can be used to switch the interface to binary mode, which requires 8 bits, 1 stop bit, and no parity. The NMEA command can be used to switch the interface back to NMEA, but it is not recommended to use it unless necessary, as it could render GPS inoperative.

$PSRF100,0,9600,8,1,0*0C
$PSRF102,9600,8,1,0*3C
where
$PSRF100
0 0=SiRF, 1=NMEA - This is where the protocol is changed.
9600 b/s rate 4800, 9600, 19200, 38400
8 7, 8 Data bits
1 0, 1 Stop bits
0 0=none, 1=odd, 2=even Parity
*0C checksum

The sentences 101 and 104 can be used to initialize values to be used by the gps. Supplying these values can shorten the initial lock time. If the clock offset is set to 0 then an internal default will be used. Sentence 101 supplies data in the internal ECEF (Earth centered, Earth Fixed) format in meters while sentence 104 supplies the data in the traditional Lat / Lon format.

$PSRF101,-2686700,-4304200,3851624,95000,497260,921,12,3*22

$PSRF104,37.3875111,-121.97232,0,95000,237759,922,12,3*3A

where

$PSRF104

37.3875111 Latitude in degrees

-121.97232 Longitude in degrees

0 Ellipsoid Altitude in meters

95000 Clock offset

237759 GPS Time of Week in seconds

922 GPS Week Number

12 Channel count (1 to 12)

3 Reset config where

1 = warm start, ephemeris valid

2 = clear ephemeris, warm start (First Fix)

3 = initialize with data, clear ephemeris

4 = cold start, clear all data

8 = cold start, set factory defaults

*3A checksum

The sentence 103 is used to control which NMEA sentences are to be sent and how often. Each sentence type is controlled individually. If the query bit is set then the gps responds by sending this message in the next second no matter what the rate is set to. Note that if trickle power is in use (can only be set in binary mode) then the actual update rate will be the selected update rate times the trickle rate which could mean that the data will be sent less frequently than was set here.

$PSRF103,05,00,01,01*20

where

$PSRF103

05 00=GGA

01=GLL

02=GSA

03=GSV

04=RMC

05=VTG

00 mode, 0=set rate, 1=query

01 rate in seconds, 0-255

01 checksum 0=no, 1=yes

*20 checksum

The 105 sentence controls a debug mode which causes the gps to report any errors it finds with the input data. $PSRF105,1*3E would turn debug on while $PSRF105,0*3F would turn it off.

Magnavox

The old Magnavox system used mostly proprietary sentences. The Magnavox system was acquired by Leica Geosystems in 1994. Information on this system can be found at this site.

The NMEA sentences themselves are described here. They all use the MVX prefix and include:

Control Port Input sentences

* $PMVXG,000 Initialization/Mode Control - Part A
* $PMVXG,001 Initialization/Mode Control - Part B
* $PMVXG,007 Control Port Configuration
* $PMVXG,023 Time Recovery Configuration
* $CDGPQ,YYY Query From a Remote Device / Request to Output a Sentence

Control Port Output Sentences

* $PMVXG,000 Receiver Status
* $PMVXG,021 Position, Height, Velocity
* $PMVXG,022 DOPs
* $PMVXG,030 Software Configuration
* $PMVXG,101 Control Sentence Accept/Reject
* $PMVXG,523 Time Recovery Configuration
* $PMVXG,830 Time Recovery Results

Sony

The Sony interface uses a proprietary sentence that looks like:

$PSNY,0,00,05,500,06,06,06,06*14

where

PSNY

0 Preamp (external antenna) status

0 = Normal

1 = Open

2 = shorted

00 Geodesic system (datum) 0-25, 0 = WGS84

05 Elevation mask in degrees

500 Speed Limit in Km

06 PDOP limit with DGPS on

06 HDOP limit with DGPS on

06	PDOP limit with DGPS off
06	HDOP limit with DGPS off
*14	Checksum

Sample Streams

These streams will be modified when a route is active with the inclusion of route specific data.

Garmin

Garmin g12 sentences for version 4.57

```
$GPRMC,183729,A,3907.356,N,12102.482,W,000.0,360.0,080301,015.5,E*6F
$GPRMB,A,,,,,,,,,,,,V*71
$GPGGA,183730,3907.356,N,12102.482,W,1,05,1.6,646.4,M,-24.1,M,,*75
$GPGSA,A,3,02,,,07,,09,24,26,,,,,1.6,1.6,1.0*3D
$GPGSV,2,1,08,02,43,088,38,04,42,145,00,05,11,291,00,07,60,043,35*71
$GPGSV,2,2,08,08,02,145,00,09,46,303,47,24,16,178,32,26,18,231,43*77
$PGRME,22.0,M,52.9,M,51.0,M*14
$GPGLL,3907.360,N,12102.481,W,183730,A*33
$PGRMZ,2062,f,3*2D
$PGRMM,WGS 84*06
$GPBOD,,T,,M,,*47
$GPRTE,1,1,c,0*07
$GPRMC,183731,A,3907.482,N,12102.436,W,000.0,360.0,080301,015.5,E*67
$GPRMB,A,,,,,,,,,,,,V*71
```

While originally a military project, GPS is considered a dual-use technology, meaning it has significant military and civilian applications. GPS has become a widely deployed and useful tool for commerce, scientific uses, tracking, and surveillance.

CHAPTER-5

GSM (Global System for Mobile communication)

5.1 Definition of GSM:

GSM is a digital mobile telephone system widely used in Europe and other regions. It uses a variation of TDMA and is the most widely used of the three digital wireless telephone technologies. GSM digitizes and compresses data, sending it down a channel with two other streams in its own time slot. It operates at either 900 MHz or 1,800 MHz frequency bands and supports voice calls, data transfer speeds up to 9.6 kbit/s, and SMS transmission.

5.2 History

The Group Special Mobile (GSM) was created in 1982 by the European Conference of Postal and Telecommunications Administrations (CEPT). In 1987, a memorandum of understanding was signed by 13 countries to develop a common cellular telephone system. In 1989, responsibility was transferred to the European Telecommunications Standards Institute (ETSI) and phase I of the GSM specifications were published in 1990. The first GSM network was launched in 1991 by Radiolinja in Finland with joint technical infrastructure maintenance from Ericsson. By 1997, GSM service was available in more than 100 countries and has become the de facto standard in Europe and Asia.

5.3 GSM Frequencies:

GSM networks operate in a number of different frequency ranges, with most 2G networks operating in the 900 MHz or 1800 MHz bands, while some countries in the Americas use the 850 MHz and 1900 MHz bands, while most 3G networks in Europe operate in the 2100 MHz band. GSM-900 uses 890-915 MHz for uplink and 935-960 MHz for downlink, providing 124 RF channels spaced at 200 kHz. E-GSM adds 50 channels to the original GSM-900 band. GSM transmission power is limited to 2 watts in GSM850/900 and 1 watt in GSM1800/1900. International roaming capability allows users to access the same services when travelling abroad as at home, while satellite roaming has extended service access to areas where terrestrial coverage is not available.

5.3.1 Mobile Telephony Standards:

Standard	Generation	Frequency band	Throughput	
GSM	2G	Allows transfer of voice or low-volume digital data.	9.6 kbps	9.6 kbps
GPRS	2.5G	Allows transfer of voice or moderate-volume digital data.	21.4-171.2 kbps	48 kbps
EDGE	2.75G	Allows simultaneous transfer of voice and digital data.	43.2-345.6 kbps	171 kbps
UMTS	3G	Allows simultaneous transfer of voice and high-speed digital data.	0.144-2 Mbps	384 kbps

Table 5.3.1: Mobile telephone standards

- **AMPS** (Advanced Mobile Phone System), which appeared in 1976 in the United States, was the first cellular network standard. It was used primarily in the Americas, Russia and Asia. This first-generation analogue network had weak security mechanisms which allowed hacking of telephones lines.
- **TACS** (Total Access Communication System) is the European version of the AMPS model. Using the 900 MHz frequency band, this system was largely used in England and then in Asia (Hong-Kong and Japan).
- **ETACS** (Extended Total Access Communication System) is an improved version of the TACS standard developed in the United Kingdom that uses a larger number of communication channels.

The first-generation cellular networks were made obsolete by the appearance of an entirely digital second generation.

5.4Second Generation of Mobile Networks (2G)

The second generation of mobile networks marked a break with the first generation of cellular telephones by switching from analogue to digital. The main 2G mobile telephony standards are

- GSM is the most commonly used standard in Europe and supported in the US, using the 900 MHz and 1800 MHz frequency bands in Europe and the 1900 MHz band in the US.
- **CDMA** (Code Division Multiple Access) uses a spread spectrum technique that allows a radio signal to be broadcast over a large frequency range.

- **TDMA** (Time Division Multiple Access) uses a technique of time division of communication channels to increase the volume of data transmitted simultaneously. TDMA technology is primarily used on the American continent, in New Zealand and in the Asia-Pacific region.

With the 2G networks, it is possible to transmit voice and low volume digital data, for example text messages (**SMS**, for Short Message Service) or multimedia messages (**MMS**, for Multimedia Message Service). The GSM standard allows a maximum data rate of 9.6 kbps.

Extensions have been made to the GSM standard to improve throughput. One of these is the **GPRS** (General Packet Radio System) service which allows theoretical data rates on the order of 114 Kbit/s but with throughput closer to 40 Kbit/s in practice. As this technology does not fit within the "3G" category, it is often referred to as **2.5G**

The **EDGE** (Enhanced Data Rates for Global Evolution) standard, billed as **2.75G**, quadruples the throughput improvements of GPRS with its theoretical data rate of 384 Kbps, thereby allowing the access for multimedia applications. In reality, the EDGE standard allows maximum theoretical data rates of 473 Kbit/s, but it has been limited in order to comply with the IMT-2000 (International Mobile Telecommunications-2000) specifications from the ITU (International Telecommunications Union).

3.3.3G

The IMT-2000 (International Mobile Telecommunications for the year 2000) specifications from the International Telecommunications Union (ITU) defined the characteristics of **3G** (third generation of mobile telephony). The most important of these characteristics are:

1. High transmission data rate.
2. 144 Kbps with total coverage for mobile use.
3. 384 Kbps with medium coverage for pedestrian use.
4. 2 Mbps with reduced coverage area for stationary use.
5. World compatibility.
6. Compatibility of 3rd generation mobile services with second generation networks.

3G offers data rates of more than 144 Kbit/s, allowing access to multimedia uses such as video transmission, video-conferencing or high-speed internet access. UMTS is the main 3G standard used in Europe and uses WCDMA encoding. HSDPA is a third generation mobile telephony protocol, able to reach data rates of 8-10 Mbps, using the 5 GHz frequency band and WCDMA encoding.

5.5 Introduction to the GSM Standard:

The **GSM** (Global System for Mobile communications) network is at the start of the 21st century, the most commonly used mobile telephony standard in Europe. It is called as Second Generation (2G) standard because communications occur in an entirely digital mode, unlike the first generation of portable telephones. When it was first standardized in 1982, it was called as **Group Special Mobile** and later, it became an international standard called **"Global System for Mobile communications"** in 1991.

The GSM standard uses the 900 MHz and 1800 MHz frequency bands in Europe and the 1900 MHz band in the United States, resulting in tri-band and bi-band portable telephones. It allows a maximum throughput of 9.6 kbps, allowing transmission of voice and low-volume digital data.

5.5.1 GSM Standards:

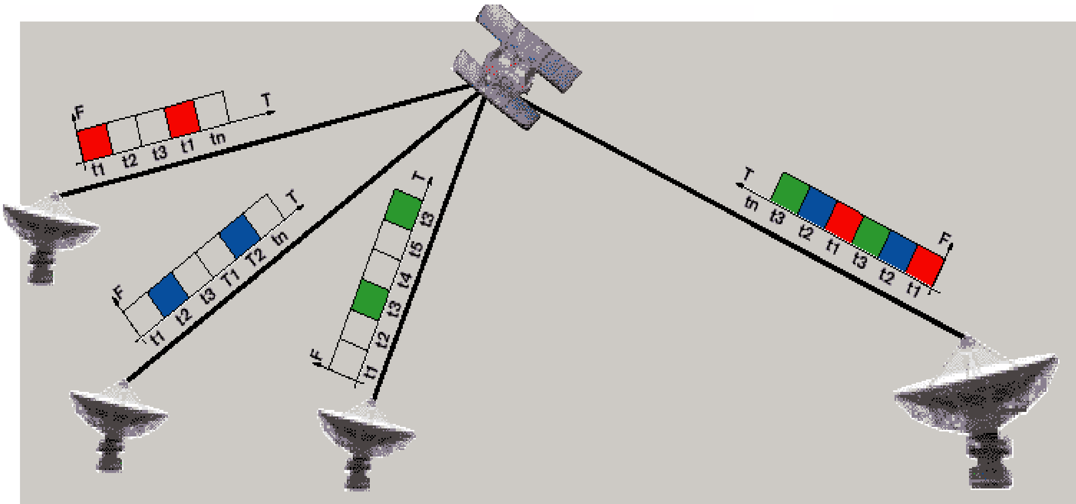
GSM uses narrowband TDMA, which allows eight simultaneous calls on the same radio frequency. There are three basic principles in multiple access, FDMA (Frequency Division Multiple Access), TDMA (Time Division Multiple Access), and CDMA (Code Division Multiple Access). All three principles allow multiple users to share the same physical channel. But the two competing technologies differ in the way user sharing the common resource.

TDMA allows the users to share the same frequency channel by dividing the signal into different time slots. Each user takes turn in a round robin fashion for transmitting and receiving over the channel. Here, users can only transmit in their respective time slot.

CDMA uses a spread spectrum technology that is it spreads the information contained in a particular signal of interest over a much greater bandwidth than the original signal. Unlike TDMA, in CDMA several users can transmit over the channel at the same time.

5.5.2 TDMA in brief:

Time Division Multiple Access (TDMA) is a type of multiplexing where two or more channels of information are transmitted over the same link by allocating a different time interval for each channel. GSM technology uses frequency hopping to reduce the effect of co-channel interference, fading and multipath.



Time Division Multiple Access

Fig 5.5.2: TDMA

TDMA systems still rely on switch to determine when to perform a handoff. Handoff occurs when a call is switched from one cell site to another while travelling. The TDMA handset constantly monitors the signals coming from other sites and reports it to the switch without caller's awareness. The switch then uses this information for making better choices for handoff at appropriate times. TDMA handset performs hard handoff, i.e., whenever the user moves from one site to another, it breaks the connection and then provides a new connection with the new site.

Advantages of TDMA:

There are lots of advantages of TDMA in cellular technologies.

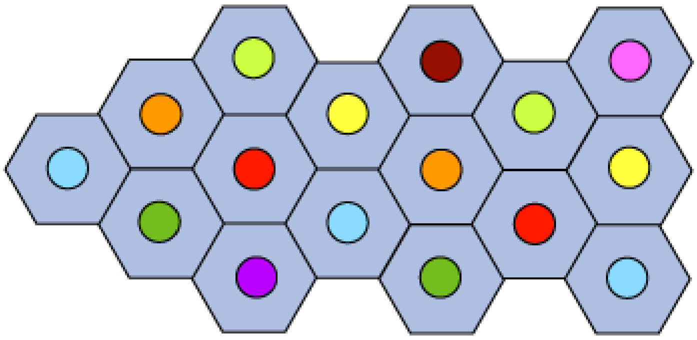
1. It can easily adapt to transmission of data as well as voice communication.
2. Operator can use 64 kbps to 120 Mbps of data rates for services such as fax, voice band data, SMS, multimedia and video conferencing.
3. Since TDMA technology separates users according to time, it ensures that there will be no interference from simultaneous transmissions

Disadvantages of TDMA:

One major disadvantage using TDMA technology is that the users has a predefined time slot. When moving from one cell site to other, if all the time slots in this cell are full the user might be disconnected. Likewise, if all the time slots in the cell in which the user is currently in are already occupied, the user will not receive a dial tone.

5.5.3 The concept of cellular network

Mobile telephone networks are based on the concept of **cells**, circular zones that overlap to cover a geographical area.



Cellular networks are based on the use of a central transmitter-receiver in each cell, called a "**base station**" (or Base Transceiver Station, written **BTS**). The smaller the radius of a cell, the higher is the available bandwidth. So, in highly populated urban areas, there are cells with a radius of a few hundred meters, while huge cells of up to 30 kilometers provide coverage in rural areas. Cells in a cellular network must be separated by a distance of two to three times the diameter of the cell in order to avoid interference.

5.5.4 Architecture of the GSM Network

In a GSM network, the user terminal is called a mobile station, which is made up of a SIM card and a mobile terminal. The terminals are identified by IMEI and IMSI codes, which can be protected with a PIN code. Communications occur through a radio link (air interface) between a mobile station and a base station. The SIM card allows each user to be identified independently of the terminal used during communication with a base station.

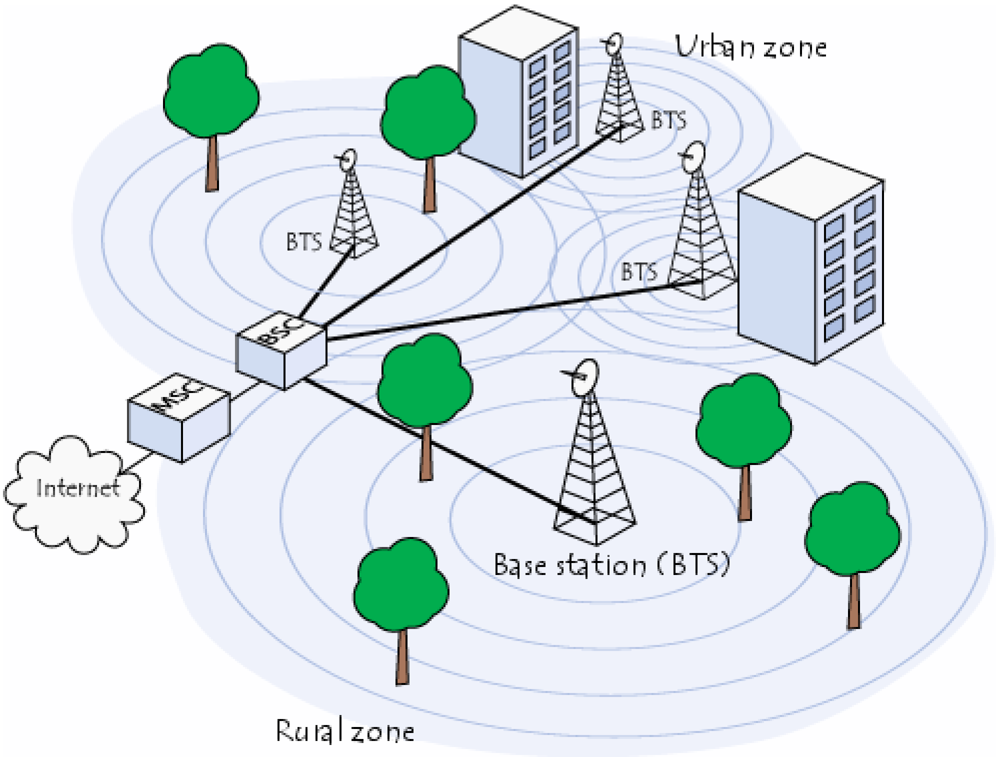


Fig 5.5.4: Architecture of the GSM Network

All the base stations of a cellular network are connected to a **base station controller** (**BSC**) which is responsible for managing distribution of the resources. The system consisting of the base station controller and its connected base stations is called the **Base Station Subsystem** (**BSS**).

Finally, the base station controllers are themselves physically connected to the **Mobile Switching Centre** (**MSC**), managed by the telephone network operator, which connects them to the public telephone network and the Internet. The MSC belongs to a **Network Station Subsystem** (**NSS**), which is responsible for managing user identities, their location and establishment of communications with other subscribers. The MSC is generally connected to databases that provide additional functions:

1. The **Home Location Register** (**HLR**) is a database containing information (geographic position, administrative information etc.) of the subscribers registered in the area of the switch (MSC).
2. The **Visitor Location Register** (**VLR**) is a database containing information of users other than the local subscribers. The VLR retrieves the data of a new user from the HLR of the user's subscriber zone. The data is maintained as long as the

user is in the zone and is deleted when the user leaves or after a long period of inactivity (terminal off).

3. The **Equipment Identify Register** (**EIR**) is a database listing the mobile terminals.
4. The **Authentication Centre** (**AUC**) is responsible for verifying user identities.
5. The cellular network formed in this way is designed to support mobility via management of handovers (movements from one cell to another).

   Finally, GSM networks support the concept of **roaming** i.e., movement from one operator network to another.

5.6 Introduction to Modem:

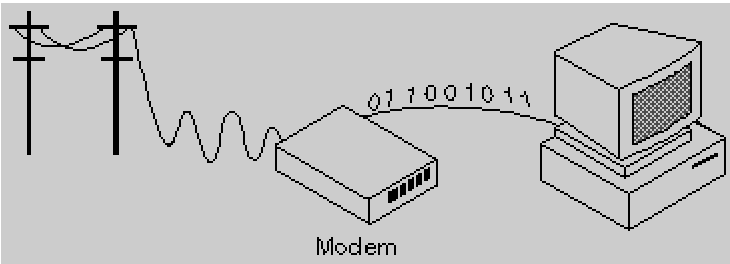


Fig 5.6.1: Modem

Modem stands for modulator-demodulator.

A modem is a device or program that enables a computer to transmit data over telephone or cable lines. Computer information is stored digitally, whereas information transmitted over telephone lines is transmitted in the form of analog waves. A modem converts between these two forms.

The RS-232 interface is the standard interface for connecting external modems to computers, and there are also modems that come as an expansion board. There are a number of different protocols for formatting data to be transmitted over telephone lines, some of which are official standards, while others have been developed by private companies. Most modems have built-in support for the more common protocols at slow data transmission speeds, but at high transmission speeds, the protocols are less standardized.

Apart from the transmission protocols that they support, the following characteristics distinguish one modem from another:

- **Bps:** Modems are measured in terms of baud rates, with the slowest rate being 300 baud (25 cps). At higher speeds, modems are measured in terms of bits per second (bps), with the fastest modems running at 57,600 bps. Data cannot be received at a faster rate than it is being sent.
- **Voice/data:** Many modems support a switch to change between voice and data modes. In data mode, the modem acts like a regular modem. In voice mode, the modem acts like a regular telephone. Modems that support a voice/data switch have a built-in loudspeaker and microphone for voice communication.
- **Auto-answer:** An auto-answer modem enables the computer to receive calls in the absence of the operator.
- **Data compression:** Some modems perform data compression, which enables them to send data at faster rates.
- **Flash memory:** Some modems come with flash memory rather than conventional ROM which means that the communications protocols can be easily updated if necessary.
- **Fax capability:** Most modern modems are fax modems, which mean that they can send and receive faxes.

GSM Modem:

A GSM modem is a wireless modem that works with a GSM wireless network The main difference between them is that a dial-up modem sends and receives data through a fixed telephone line while a wireless modem sends and receives data through radio waves.

Fig 5.6.2: GSM/GPRS modem

A GSM modem can be an external device or a PC Card / PCMCIA Card. An external GSM modem is connected to a computer through a serial cable or USB cable, while a PC Card / PCMCIA Card is designed for use with a laptop computer. It requires a SIM card from a wireless carrier to operate, just like a GSM mobile phone.

A SIM card contains the following information:

- Subscriber telephone number (MSISDN)
- International subscriber number (**IMSI, International Mobile Subscriber Identity)**
- State of the SIM card
- Service code (operator)
- Authentication key
- PIN (**Personal Identification Code**)
- PUK (**Personal Unlock Code**)

Computers use AT commands to control modems. Both GSM modems and dial-up modems support a common set of standard AT commands. In addition to the standard AT commands, GSM modems support an extended set of AT commands. These extended AT commands are defined in the GSM standards. With the extended AT commands, the following operations can be performed:

- Reading, writing and deleting SMS messages.
- Sending SMS messages.
- Monitoring the signal strength.
- Monitoring the charging status and charge level of the battery.
- Reading, writing and searching phone book entries.

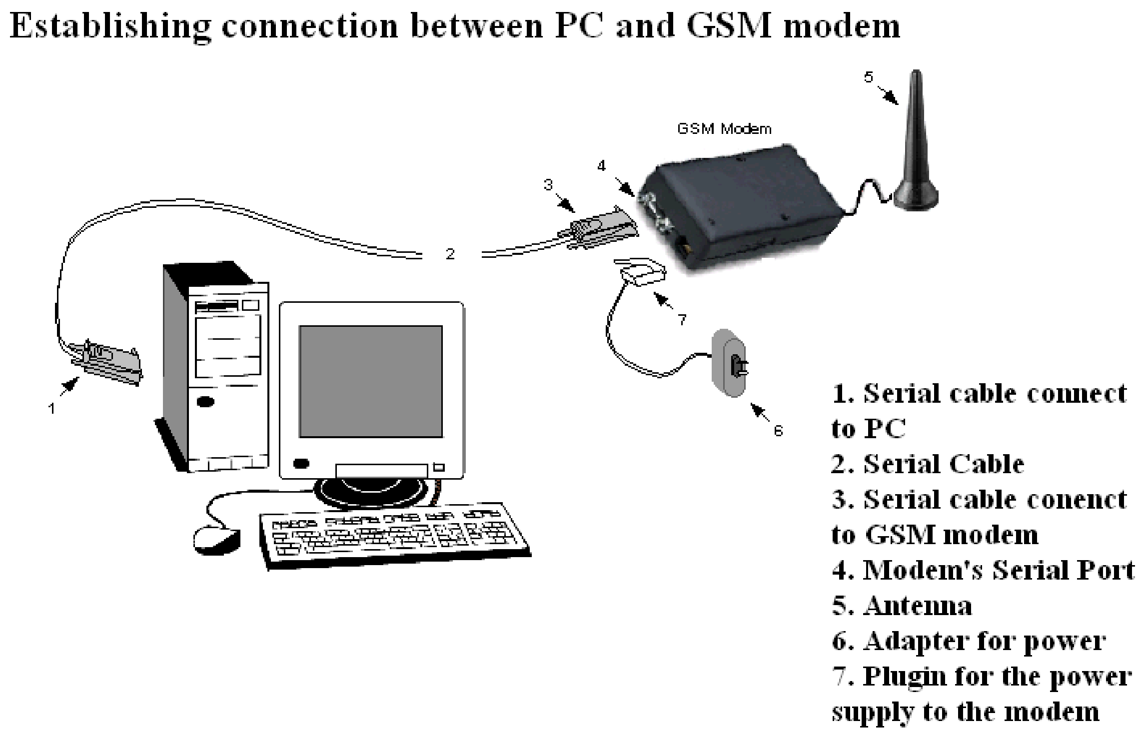


Fig 5.6.3: Connection between PC and GSM

The number of SMS messages that can be processed by a GSM modem per minute is very low i.e., about 6 to 10 SMS messages per minute.

Introduction to AT Commands

AT commands control modems, abbreviated as ATtention, and are used to control wired dial-up modems like ATD, ATA, ATH, and ATO. These commands are also supported by GSM modems and mobile phones.

Besides this common AT command set, GSM modems and mobile phones support an AT command set that is specific to the GSM technology, which includes SMS-related commands like AT+CMGS (Send SMS message), AT+CMSS (Send SMS message from storage), AT+CMGL (List SMS messages) and AT+CMGR (Read SMS messages).

It should be noted that the starting "AT" is the prefix that informs the modem about the start of a command line. It is not part of the AT command name. For example, D is the actual AT command name in ATD and +CMGS is the actual AT command name in AT+CMGS.

Some of the tasks that can be done using AT commands with a GSM modem or mobile phone are listed below:

- Get basic information about the mobile phone or GSM modem. For example, name of manufacturer (AT+CGMI), model number (AT+CGMM), IMEI number (International Mobile Equipment Identity) (AT+CGSN) and software version (AT+CGMR).
- Get basic information about the subscriber. For example, MSISDN (AT+CNUM) and IMSI number (International Mobile Subscriber Identity) (AT+CIMI).
- Get the current status of the mobile phone or GSM/GPRS modem. For example, mobile phone activity status (AT+CPAS), mobile network registration status (AT+CREG), radio signal strength (AT+CSQ), battery charge level and battery charging status (AT+CBC).
- Establish a data connection or voice connection to a remote modem (ATD, ATA, etc).
- Send and receive fax (ATD, ATA, AT+F*).
- Send (AT+CMGS, AT+CMSS), read (AT+CMGR, AT+CMGL), write (AT+CMGW) or delete (AT+CMGD) SMS messages and obtain notifications of newly received SMS messages (AT+CNMI).
- Read (AT+CPBR), write (AT+CPBW) or search (AT+CPBF) phonebook entries.
- Perform security-related tasks, such as opening or closing facility locks (AT+CLCK), checking whether a facility is locked (AT+CLCK) and changing passwords(AT+CPWD).
- Control the presentation of result codes / error messages of AT commands. For example, the user can control whether to enable certain error messages (AT+CMEE) and whether error messages should be displayed in numeric format or verbose format (AT+CMEE=1 or AT+CMEE=2).
- Get or change the configurations of the mobile phone or GSM/GPRS modem. For example, change the GSM network (AT+COPS), bearer service type (AT+CBST), radio link protocol parameters (AT+CRLP), SMS center address (AT+CSCA) and storage of SMS messages (AT+CPMS).
- Save and restore mobile phone or GSM/GPRS modem configurations, such as SMS messaging settings, using AT+CSAS and AT+CRES, as GSM modems offer better AT commands support.

5.7 Basic concepts of SMS technology

1. Validity Period of an SMS Message

SMS messages are temporarily stored in the SMS center when the recipient's mobile phone is offline. A validity period is set to prevent the message from being forwarded when the recipient becomes online. A mobile phone should have a menu option to set this period, which automatically includes it in outbound SMS messages.

2. Message Status Reports

To check if an SMS message has arrived successfully, set a flag in the message to notify the SMS center of a status report. The status report is sent to the user's mobile phone. A mobile phone should have a menu option to set the status report feature on or off, which automatically sets the corresponding flag in outbound SMS messages. The feature is turned off by default on most mobile phones and GSM modems.

3. Message Submission Reports:

The SMS center sends a message submission report to mobile phones to check for errors or failures. If no errors are found, the mobile phone may notify the user of the failure. If the report is lost, the mobile phone is notified. If successful, the SMS center ignores the new message and sends a report, preventing the same message from being sent multiple times.

4. Message Delivery Reports:

The recipient's mobile phone sends a message delivery report to the SMS center, indicating if there are errors or failures. If no errors are found, the recipient receives a positive report. If a status report is requested, the SMS center sends it to the sender. If the report is not received, the center concludes the message is lost and terminates the message.

CHAPTER-6

TEMPERATURE SENSOR (LM35)

6.1 Precision Centigrade Temperature Sensor

In this project, in order to obtain the fan speed based on temperature, initially this temperature value has to be read and fed to the microcontroller. This temperature value has to be sensed.Thus a sensor has to be used and the sensor used in this project is LM35. It converts temperature value into electrical signals.

LM35 series sensors are precision integrated-circuit temperature sensors whose output voltage is linearly proportional to the Celsius temperature. The LM35 requires no external calibration since it is internally calibrated. . The LM35 does not require any external calibration or trimming to provide typical accuracies of ±1⁄4°C at room temperature and ±3⁄4° Cover a full −55 to +150°C temperature range.

The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only 60 µA from its supply, it has very low self-heating, less than 0.1°C in still air.

Features

- Calibrated directly in ° Celsius (Centigrade)
- Linear + 10.0 mV/°C scale factor
- 0.5°C accuracy guaranteed (at +25°C)
- Rated for full −55° to +150°C range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than 60 µA current drain
- Low self-heating, 0.08°C in still air
- Nonlinearity only ±1⁄4°C typical
- Low impedance output, 0.1 W for 1 mA load

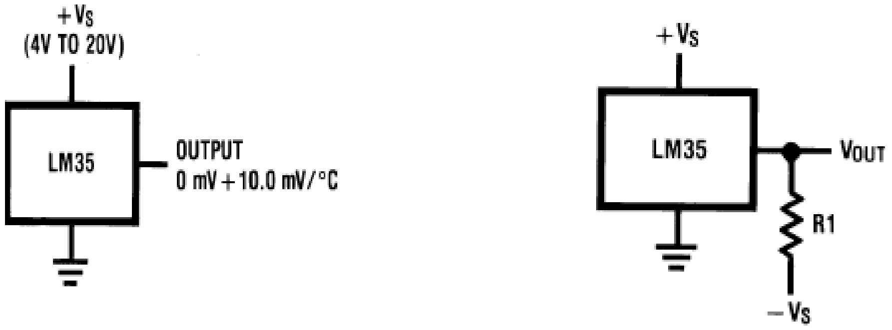


Fig 6.1.1

6.2 The characteristic of this LM35 sensor is:

For each degree of centigrade temperature, it outputs 10milli volts.

ADC accepts the output from LM35 and converts that data into digital form which is sent to microcontroller for further processing.

CHAPTER-7

SOFTWARE REQUIRED

7.1 ARDUINO:

Arduino Uno is a microcontroller board based on the 8-bit ATmega328P microcontroller. It has 14 digital input/output pins, 6 analog input pins, a USB connection, Power barrel jack, an ICSP header and a reset button. The ATmega328 microcontroller contains 32 general purpose working registers, which can carry one instruction in one clock cycle.

Specifications:

Microcontroller	ATmega328P – 8 bit AVR family microcontroller
Operating Voltage	5V
RecommendedInput Voltage	7-12V
Input Voltage Limits	6-20V
Analog Input Pins	6 (A0 – A5)
Digital I/O Pins	14 (Out of which 6 provide PWM output)
DC Current on I/O Pins	40 Ma
DC Current on 3.3V Pin	50 Ma
Flash Memory	32 KB (0.5 KB is used for Boot loader)
SRAM	2 KB
EEPROM	1 KB
Frequency (Clock Speed)	16 MHz

Table 7.1: Atmega328 specifications

ARDUINO:

Arduino is a prototype platform (open-source) based on an easy-to-use hardware and software. It consists of a circuit board, which can be programmed (referred to as a microcontroller) and a ready-made software called Arduino IDE (Integrated Development Environment), which is used to write and upload the computer code to the physical board.

The key features are −

- Arduino boards are able to read analog or digital input signals from different sensors and turn it into an output such as activating a motor, turning LED on/off, connect to the cloud and many other actions.
- You can control your board functions by sending a set of instructions to the microcontroller on the board via Arduino IDE (referred to as uploading software).
- Unlike most previous programmable circuit boards, Arduino does not need an extra piece of hardware (called a programmer) in order to load a new code onto the board. You can simply use a USB cable.
- Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program.
- Finally, Arduino provides a standard form factor that breaks the functions of the micro-controller into a more accessible package.

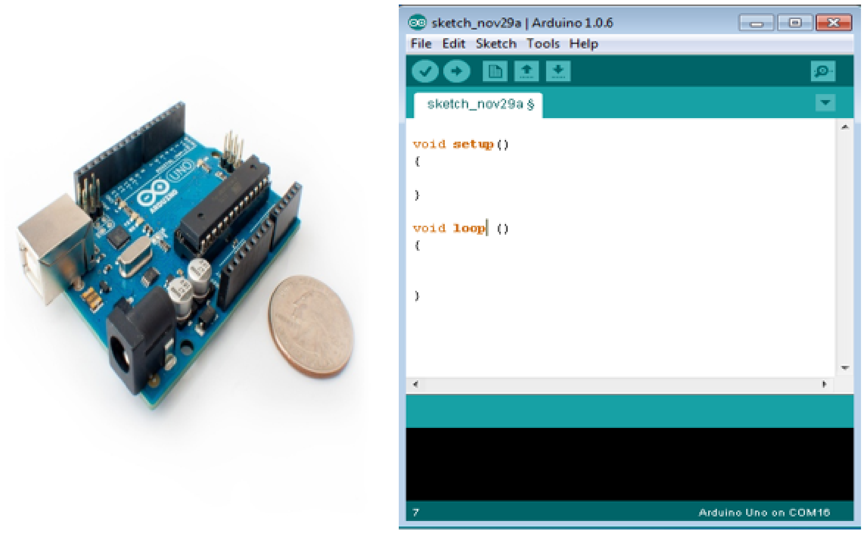


Fig7.1.1: Arduino Uno

7.2 BOARD TYPES

Arduino boards, varying in microcontrollers, are all programmed through the Arduino IDE, ensuring compatibility across various devices.

The differences are based on the number of inputs and outputs (the number of sensors, LEDs, and buttons you can use on a single board), speed, operating voltage, form factor etc. Some boards are designed to be embedded and have no programming interface (hardware), which you would need to buy separately. Some can run directly from a 3.7V battery, others need at least 5V.

Here is a list of different Arduino boards available.

Board Name	Operating Volt	Clock Speed	Digital i/o	Analog Inputs	PWM	UART	Programming Interface
Arduino Uno R3	5V	16MHz	14	6	6	1	USB via ATMega16U2
Arduino Uno R3 SMD	5V	16MHz	14	6	6	1	USB via ATMega 16U2
Red Board	5V	16MHz	14	6	6	1	USB via FTDI
Arduino Pro 3.3v/8 MHz	3.3V	8MHz	14	6	6	1	FTDI-Compatible Header
Arduino Pro 5V/16MHz	5V	16MHz	14	6	6	1	FTDI-Compatible Header
Arduino mini 05	5V	16MHz	14	8	6	1	FTDI-Compatible Header
Arduino Pro mini 3.3v/8mhz	3.3V	8MHz	14	8	6	1	FTDI-Compatible Header
Arduino Pro mini 5v/16mhz	5V	16MHz	14	8	6	1	FTDI-Compatible Header
Arduino Ethernet	5V	16MHz	14	6	6	1	FTDI-Compatible Header
Arduino Fio	3.3V	8MHz	14	8	6	1	FTDI-Compatible Header
LilyPad Arduino 328 main board	3.3V	8MHz	14	6	6	1	FTDI-Compati
LilyPad Arduino simple board	3.3V	8MHz	9	4	5	0	FTDI-Compatible Header

Table7.2.1: Arduino boards based on ATMEGA32u4 microcontroller

BOARD DESCRIPTION:

In this chapter, we will learn about the different components on the Arduino board. We will study the Arduino UNO board because it is the most popular board in the Arduino board family. In addition, it is the best board to get started with electronics and coding. Some boards look a bit different from the one given below, but most Arduinos have majority of these components in common.

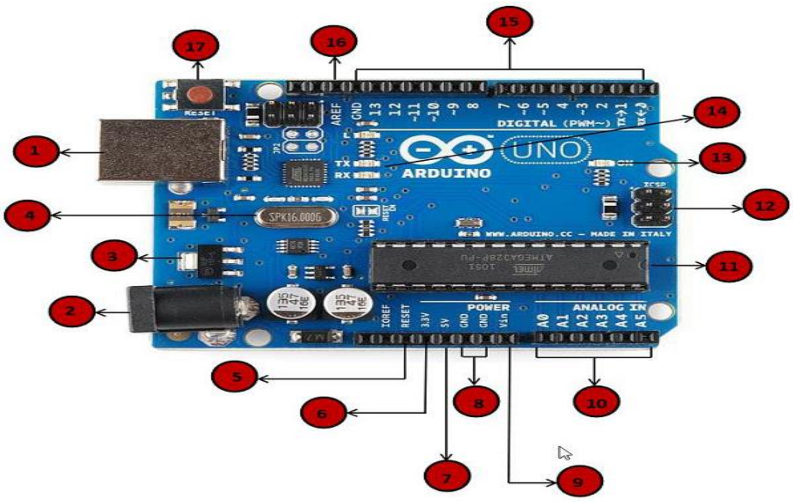


Fig 7.2.2: Board description

1	Power USB Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection (1).
2	Power (Barrel Jack) Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack (2).
3	Voltage Regulator The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.
4	Crystal Oscillator Arduino calculates time by using the crystal oscillator, which has a frequency of 16,000,000 Hertz or 16 MHz.
5, 17	Arduino Reset You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET (5).
6, 7, 8, 9	Pins (3.3, 5, GND, Vin) 3.3V (6) − Supply 3.3 output volt

	5V (7) − Supply 5 output volt Most of the components used with Arduino board works fine with 3.3 volt and 5 volt. GND (8)(Ground) − There are several GND pins on the Arduino, any of which can be used to ground your circuit. Vin (9) − This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.
10	Analog pins The Arduino UNO board has six analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be read by the microprocessor.
11	Main microcontroller The Arduino board has its own microcontroller (11) which is usually of the ATMEL Company. To load a new program from the Arduino IDE, you must know the IC's construction and functions. This information is available on the top of the IC and can be found in the data sheet.
12	ICSP pin ICSP (12) is an AVR, a programming header for the Arduino, and an SPI (Serial Peripheral Interface), which slaves the output device to the master of the SPI bus.
13	Power LED indicator This LED should light up when you plug your Arduino into a power source to indicate that your board is powered up correctly. If this light does not turn on, then there is something wrong with the connection.
14	TX and RX LEDs The Arduino UNO board has two labels: TX (transmit) and RX (receive). They appear at digital pins 0 and 1, and on the TX and RX leds. The speed of flashing depends on the baud rate used.
15	Digital I/O The Arduino UNO board has 14 digital I/O pins (15) of which 6 provide PWM output. These pins can be configured to read logic values or drive modules, and the pins labeled "~" can generate PWM.
16	AREF AREF stands for Analog Reference. It is sometimes, used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

Table 7.2.3: pin Board description

7.3 ARDUINO FAMILY

Arduino makes several different boards, each with different capabilities. In addition, part of being open-source hardware means that others can modify and produce derivatives of Arduino boards that provide even more form factors and functionality. If you're not sure which one is right for your project , check this guide for some helpful hints. Here are a few options that are well suited to someone new to the world of Arduino.

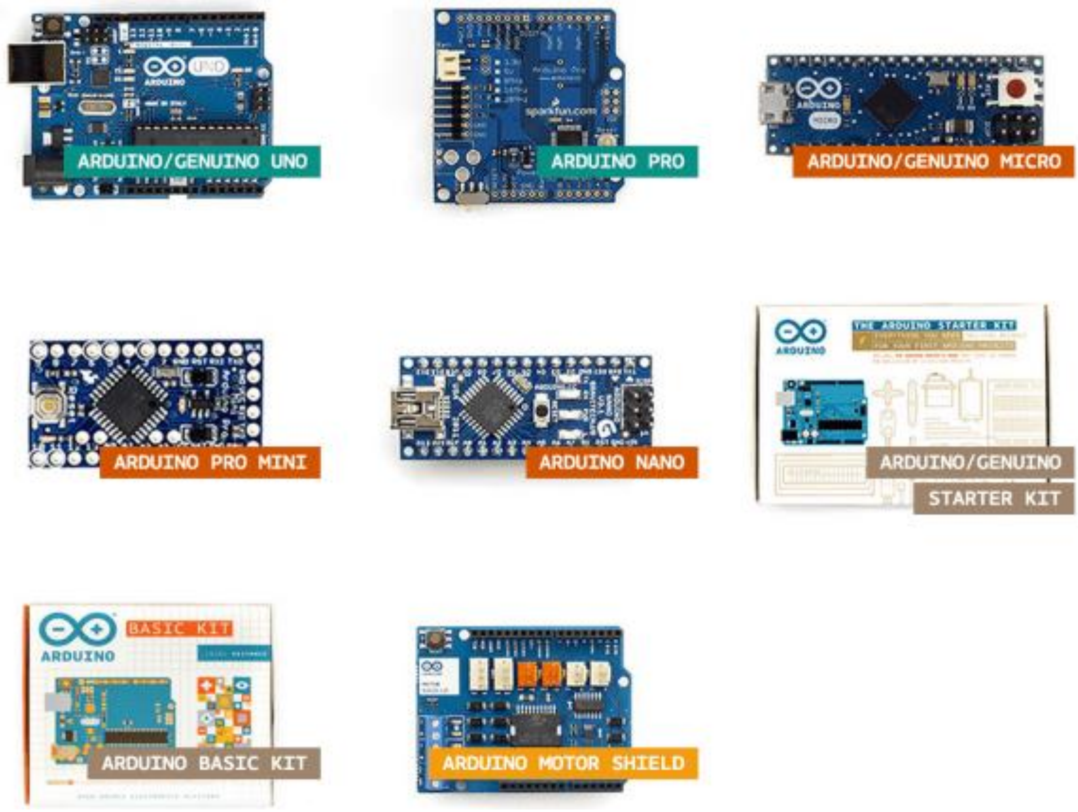


Fig 7.3.1: Arduino Family

PIN DESCRIPTION OF ATMEGA328

ADVANTAGES OF ARDUINO

- It is cheap
- It comes with an open supply hardware feature that permits users to develop their own kit
- The software of the Arduino is well-suited with all kinds of in operation systems like Linux, Windows, and Macintosh, etc.
- It also comes with open supply software system feature that permits tough software system developers to use the Arduino code to merge with the prevailing programming language libraries and may be extended and changed.
- For beginners, it is very simple to use.

APPLICATIONS:

ATMEGA328 is commonly used in many projects and autonomous systems where a simple, low- powered, low- cost microcontroller is needed.

Perhaps the most common implementation of this chip is on the popular Arduino development platform, namely the Arduino UNO and Arduino Nano model

POWER SUPPLY UNIT:

Power supplies convert high voltage AC mains electricity to low voltage for electronic circuits and other devices. Regulated DC Power Supply maintains output voltage constant despite AC mains fluctuations or load variations.

7.3.1 BLOCK DIAGRAM OF POWER SUPPLY:

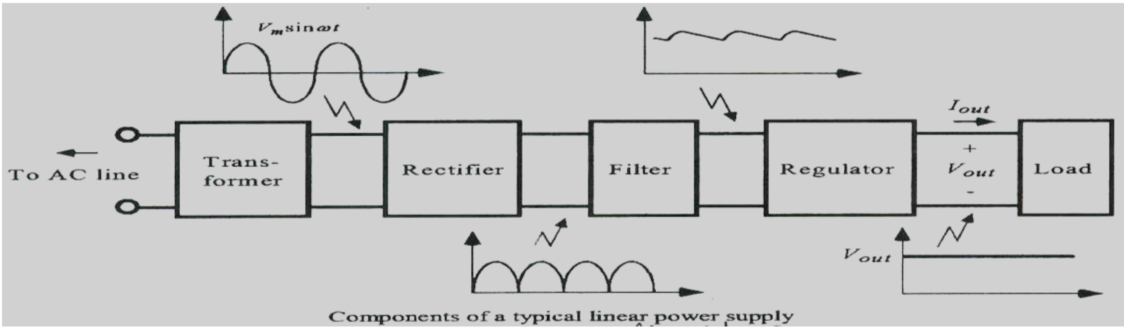


Fig 7.3.2: Block Diagram of Power Supply

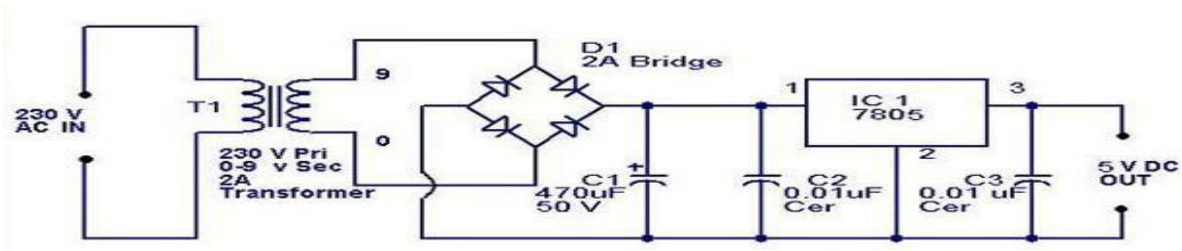


Fig 7.3.3: Schematic Diagram of Power Supply

7.4 DESCRIPTION OF POWER SUPPLY:

A power supply is a component that supplies power to at least one electric load. Typically, it converts one type of electrical power to another, but it may also convert a a different form of energy – such as solar, mechanical, or chemical - into electrical energy.

A power supply is a device that converts AC current to DC current, usually located at the rear of a component. It is also known as a power supply unit, power brick or power adapter.

CHAPTER-8

DHT11 SENSOR

Humidity is the measure of water vapour present in the air. The level of humidity in air affects various physical, chemical and biological processes. In industrial applications, humidity can affect the business cost of the products, health and safety of the employees. So, in semiconductor industries and control system industries measurement of humidity is very important. Humidity measurement determines the amount of moisture present in the gas that can be a mixture of water vapour, nitrogen, argon or pure gas etc… Humidity sensors are of two types based on their measurement units. They are a relative humidity sensor and Absolute humidity sensor. DHT11 is a digital temperature and humidity sensor.

8.1 What is a DHT11 Sensor?

DHT11 is a low-cost digital sensor for sensing temperature and humidity. This sensor can be easily interfaced with any micro-controller such as Arduino, Raspberry Pi etc… to measure humidity and temperature instantaneously.

DHT11 humidity and temperature sensor is available as a sensor and as a module. The difference between this sensor and module is the pull-up resistor and a power-on LED. DHT11 is a relative humidity sensor. To measure the surrounding air this sensor uses a thermistor and a capacitive humidity sensor.

8.1.1 Working Principle of DHT11 Sensor:

DHT11 sensor consists of a capacitive humidity sensing element and a thermistor for sensing temperature. The humidity sensing capacitor has two electrodes with a moisture holding substrate as a dielectric between them. Change in the capacitance value occurs with the change in humidity levels. The IC measure, process this changed resistance values and change them into digital form.

For measuring temperature this sensor uses a Negative Temperature

coefficient thermistor, which causes a decrease in its resistance value with increase in temperature. To get larger resistance value even for the smallest change in temperature, this sensor is usually made up of semiconductor ceramics or polymers.

The temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. Humidity range of this sensor is from 20 to 80% with 5% accuracy. The sampling rate of this sensor is 1Hz .i.e. it gives one reading for every second. DHT11 is small in size with operating voltage from 3 to 5 volts. The maximum current used while measuring is 2.5mA.

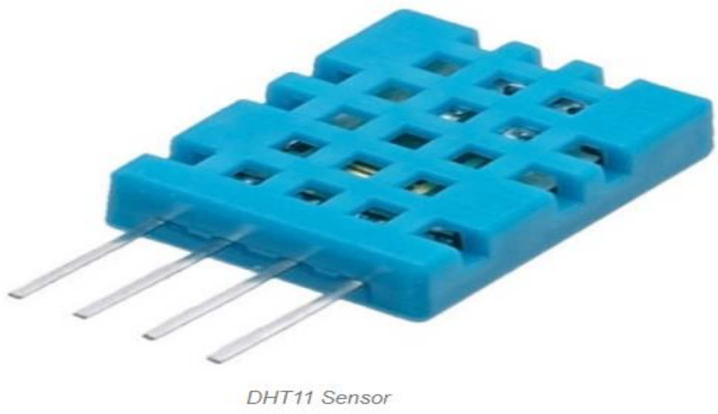


Fig 8.1.2: DTH11 Sensor

DHT11 sensor has four pins- VCC, GND, Data Pin and a not connected pin. A pull-up resistor of 5k to 10k ohms is provided for communication between sensor and micro-controller.

Applications

This sensor is used in various applications such as measuring humidity and temperature values in heating, ventilation and air conditioning systems. Weather stations also use these sensors to predict weather conditions. The humidity sensor is used as a preventive measure in homes where people are affected by humidity. Offices, cars, museums, greenhouses and industries use this sensor for measuring humidity values and as a safety measure.

CHAPTER-9

RELAYA

A relay is an electrically operated switch. These are remote control electrical switches that are controlled by another switch, such as a horn switch or a computer as in a power train control module, devices in industries, home based applications. Relays allow a small current pin, 4-pin, 5-pin, and 6-pin, single switch or dual switches. Relays are used throughout the automobile. Relays which come in assorted sizes, ratings, and applications, are used as remote-control switches. A typical vehicle can have 20 relays or more.

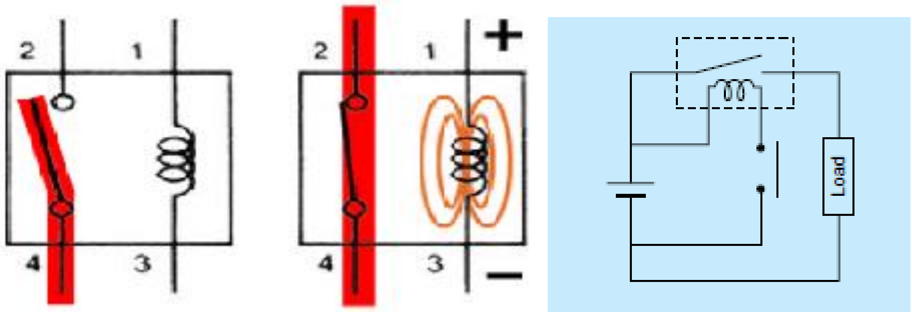


Fig 9.1: Remote control electrical switches

9.1 BASICS ON RELAY HANDLING

- To maintain initial performance, care should be taken to avoid dropping or hitting the relay.
- Under normal use, the relay is designed so that the case will not detach. To maintain initial performance, the case should not be removed. Relay characteristics cannot be guaranteed if the case is removed.
- Use of the relay in an atmosphere at standard temperature and humidity with minimal amounts of dust, SO 2, H 2 S, or organic gases is recommended.
- Please avoid the use of silicon-based resins near the relay, because doing so may result in contact failure. (This applies to plastic sealed type relays, too.)
- Care should be taken to observe correct coil polarity (+, –) for polarized relays.
- Proper usage requires that the rated voltage be impressed on the coil. Use rectangular waves for DC coils and sine waves for AC coils.
- Be sure the coil impressed voltage does not continuously exceed the maximum allowable voltage.

- Absolutely avoid using switching voltages and currents that exceed the designated values.
- The rated switching power and life vary depending on the load and operating conditions, so it is important to check carefully.
- type of load and operating conditions before use.
- Do not exceed the usable ambient temperature values listed in the catalog.
- Use the flux-resistant type or sealed type if automatic soldering is to be used.
- Use alcohol based cleaning solvents when cleaning is to be performed using a sealed type relay.
- Avoid ultrasonic cleaning of all types of relays.
- Avoid bending terminals, because it may cause malfunction.
- As a guide, use a Faston mounting pressure of 40 to 70N {4 to 7kgf}for relays with tab terminals.

A relay is used to isolate one electrical circuit from another, allowing a low current control circuit to make or break an electrically isolated high current circuit path. The basic relay consists of a coil and a set of contacts, which are pulled together

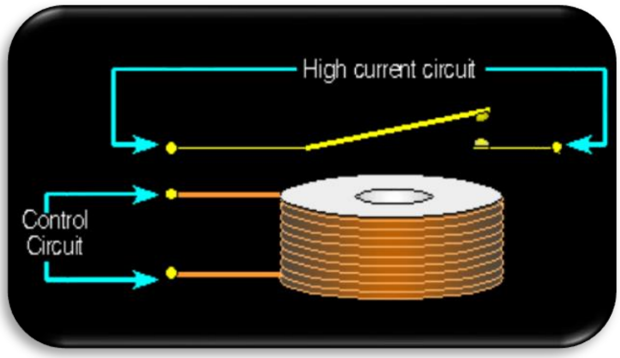


Fig 9.1.1: Relay

when voltage is applied. The diagram below shows the parts of a simple relay.

Operation:

A current flowing through a coil attracts an armature that is mechanically linked to a moving contact. When the current is switched off, the armature is returned by a spring. Latching relays require operation of a second coil to reset the contact position. Solid-state relays operate a thyristor or other solid-state switching device with a transformer or light-emitting diode to trigger it.

9.2 Pole and throw:

Since relays are switches the terminology applied to switches is also applied to relays. A relay will switch one or more poles, each of whose contacts can be thrown by energizing the coil in one of three ways:

- Normally-open (**NO**) contacts connect the circuit when the relay is activated; the circuit is disconnected when the relay is inactive. It is also called a **Form A** contact or "make" contact.
- Normally-closed (**NC**) contacts disconnect the circuit when the relay is activated; the circuit is connected when the relay is inactive. It is also called a **Form B** contact or "break" contact.
- Change-over (CO) or double-throw (DT) contacts control two circuits: one normally-open and one normally-closed with a common terminal. This type of contact is also known as a Form C contact or "transfer" contact, if it utilizes "make before break" functionality.

SPST

SPST relay stands for Single Pole Single Throw relay. Current will only flow through the contacts when the relay coil is energized.

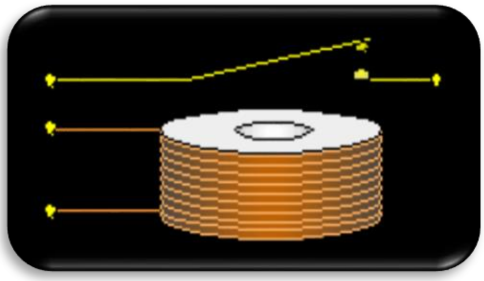


Fig: SPST Relay

SPDT Relay:

SPDT Relay is the most commonly used relay in car audio, with current flowing between movable and fixed contacts.

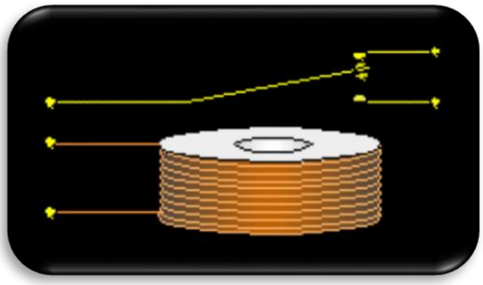


Fig: SPDT Relay

DPST Relay:

DPST relay stands for Double Pole Single Throw relay, where two separate sets of contacts are pulled down to make contact with their stationary counterparts when energized. De-energized, there is no complete circuit path.

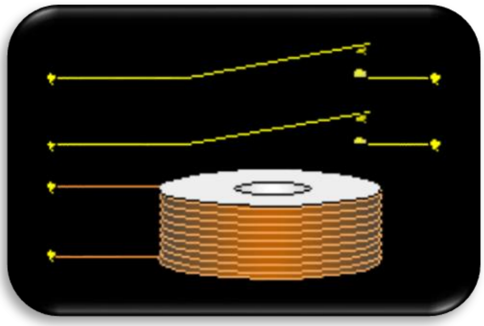


Fig: DPST Relay

DPDT Relay

DPDT relay stands for Double Pole Double Throw relay. It operates like the SPDT relay but has twice as many contacts. There are two completely isolated sets of contacts.

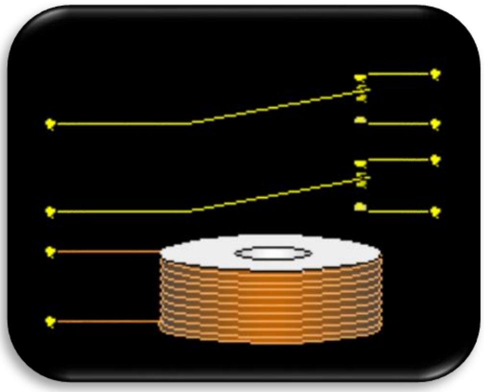


Fig: DPDT Relay

This is a 4 Pole Double Throw relay. It operates like the SPDT relay but it has 4 sets of isolated contacts.

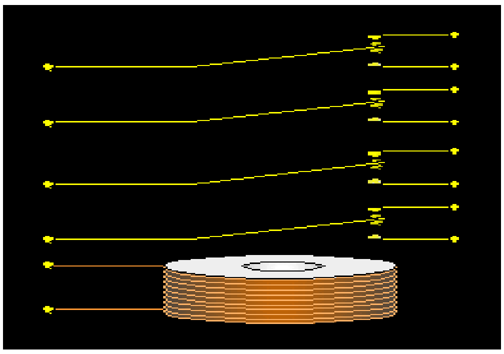


Fig: 4 Pole Double Throw relay

Applications:

Relays are used:

- To control a high-voltage circuit with a low-voltage signal, as in some types of modems,
- To control a high-current circuit with a low-current signal, as in the starter solenoid of an automobile,
- To detect and isolate faults on transmission and distribution lines by opening and closing circuit breakers (protection relays),
- To generate sound. A vibrator, described above, creates a buzzing sound because of the rapid oscillation of the armature. This is the basis of the electric bell, which consists of a vibrator with a hammer attached to the armature so it can repeatedly strike a bell.
- To perform time delay functions. Relays can be used to act as an mechanical time delay device by controlling the release time by using the effect of residual magnetism by means of a inserting copper disk between the armature and moving blade assembly.

CHAPTER-10

WORKING

A wireless black box is a device that displays vehicle crash parameters every three seconds, including date, time, temperature, location, vibration, and alcohol limit. It sends messages to registered mobile numbers during accidents. Sensors like temperature, humidity, vibration, alcohol, and gyroscopic sensors measure these parameters. These sensors send signals to a microcontroller like an Arduino mega2560, which uses GSM, SD card, and GPS modules to process the data. The device also helps in detecting potential injuries and providing valuable information to emergency responders.

APPLICATIONS:

Wireless black boxes have several applications. One of the main applications is in vehicles, where they can be used to record and store data about the vehicle's performance and status. Wireless black boxes can also be used in other transportation systems such as airplanes and trains to record and store data about their performance and status.

ADVANTAGES:

- There are several advantages to having a wireless black box in your car. One advantage is that it can improve your driving skills. When you have a black box fitted, you'll probably find yourself thinking far more consciously about your driving habits.
- Black box data can be used to deter theft, settle claims, and help insurance companies with their vehicle crash investigations. It can also improve road status to reduce the death rate.

CHAPTER - 11
CODE

```
#include <LiquidCrystal.h>
const int rs = PB12, en = PB13, d4 = PB14, d5 = PB15, d6 = PA11, d7 = PA12;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
char res[130];
int tmp = PA1;
int m1=PA2;
int buz=PA0;
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_ADXL345_U.h>
Adafruit_ADXL345_Unified accel = Adafruit_ADXL345_Unified (12345);
const int trigPin = PA4;
const int echoPin = PA5;
long duration;
int distance;

void serialFlush(){
  while(Serial1.available() > 0) {
    char t = Serial1.read();
  }
}

 char mob[15] = {"9014104775\0"};


char gps_location[40];
void gps()
{

  lcd.clear();
  lcd.setCursor(0, 1);lcd.print("READING GPS");delay(1000);
   while(1)
   {
    if(Serial1.find("$GPRMC"))
    {
      for(int i=0;i<14;i++)
      {
        while(!Serial1.available());
        char ch = Serial1.read();
      }
      for(int i=0;i<24;i++)
      {
        while(!Serial1.available());
        gps_location[i] = Serial1.read();
        Serial1.println(gps_location[i]);
      }
```

```
      break;
    }
   }
   lcd.setCursor(0, 1);lcd.print("READING GPS DONE");
     delay(2000);

}
 void sendmsg(char *num,char *msg,char *loc)
{
  Serial1.print("AT+CMGS=\"");
  Serial1.print(num);
  Serial1.println("\"");
  delay(2000);
  Serial1.println(msg);
  Serial1.println(loc);
  delay(1000);
  Serial1.write(0x1a);
  delay(2000);
}
void setup() {
 pinMode(trigPin, OUTPUT);
 pinMode(echoPin, INPUT);
  pinMode(m1,OUTPUT);
     pinMode(buz,OUTPUT);
 pinMode(tmp,INPUT);
  digitalWrite(m1,HIGH);
  digitalWrite(buz,LOW);
delay(1000);
  Serial.begin(9600);
  Serial1.begin(9600);
     if(!accel.begin())
  {
    Serial.println("Ooops, no ADXL345 detected ... Check your wiring!");
    while(1);
  }
  lcd.clear();lcd.setCursor(0, 0);lcd.print("ADXL ok");
  lcd.begin(16,2);
  lcd.clear();lcd.setCursor(0, 0);lcd.print("BLACK BOX");
  delay(2000);
  lcd.clear();lcd.setCursor(0, 0);lcd.print("GSM INIT");
    Serial1.println("AT");  delay(1000);
  Serial1.println("ATE0");  delay(1000);
  Serial1.println("AT+CMGF=1");  delay(1000);
  Serial1.println("AT+CNMI=1,2,0,0");  delay(1000);
  Serial1.println("AT+CSMP=17,167,0,16");  delay(1000);
     gps();
  sendmsg(mob," PERSON  Location at:",gps_location);delay(3000);

}
```

```
void loop() {
  sensors_event_t event;
  accel.getEvent(&event);
 int xx = event.acceleration.x;
 int yy = event.acceleration.y;
 int zz = event.acceleration.z;
  Serial.print("X: "); Serial.print(event.acceleration.x); Serial.print("  ");
  Serial.print("Y: "); Serial.print(event.acceleration.y); Serial.print("  ");
  Serial.print("Z: "); Serial.print(event.acceleration.z); Serial.print("  ");Serial.println("m/s^2
");
 digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = duration * 0.034 / 2; // Speed of sound wave divided by 2
   Serial.println("distance : ");
    Serial.print(distance);
 int td = analogRead(tmp)-500;
lcd.clear();
 lcd.clear();
    lcd.setCursor(0, 0); lcd.print("X:");lcd.print(event.acceleration.x);
    lcd.setCursor(9, 0); lcd.print("Y:");lcd.print(event.acceleration.y);
    lcd.setCursor(8, 1); lcd.print("T:");lcd.print(td);
    lcd.setCursor(0, 1); lcd.print("D:");lcd.print(distance);
              delay(200);
   if(distance < 5)
  {
    lcd.clear();lcd.setCursor(0,0);lcd.print("accident occured");
    lcd.setCursor(0,1);lcd.print("D:");lcd.print(distance);delay(500);
     digitalWrite(m1, LOW);
        gps(); delay(1000);
      sendmsg(mob,"accident detected by sensor at:",gps_location);delay(2000);
    }
  if(event.acceleration.x > 4)
        {

     lcd.clear();lcd.setCursor(0,0);lcd.print("VEHICLE AXIS");delay(1000);
          lcd.setCursor(0,0);lcd.print("CHANGED");delay(1000);
           digitalWrite(m1,LOW);
      digitalWrite(buz,HIGH);delay(500);digitalWrite(buz,LOW);
  delay(500);
           gps(); delay(1000);
      sendmsg(mob,"accident detetced by mems axis at:",gps_location);delay(2000);
        }


  if(td > 100 )
```

```
{
  lcd.clear();
  lcd.setCursor(0, 0);lcd.print("high temperature");
  digitalWrite(buz,HIGH);delay(500);digitalWrite(buz,LOW);
  delay(500);
  gps(); delay(1000);
sendmsg(mob,"vehicle high temperature Location at:",gps_location);delay(2000);
  delay(500);
  }
  else
  {
    digitalWrite(buz,LOW);
   digitalWrite(m1,HIGH);
   delay(1000);
  }


}
//
//    FILE: dht22.cpp
// VERSION: 0.1.00
// PURPOSE: DHT22 Temperature & Humidity Sensor library for Arduino
//
// DATASHEET:
//
// HISTORY:
// 0.1.0 by Rob Tillaart (01/04/2011)
// inspired by DHT11 library
//

#include "dht.h"

#define TIMEOUT 10000

/////////////////////////////////////////////////////
//
// PUBLIC
//


// return values:
//  0 : OK
// -1 : checksum error
// -2 : timeout
int dht::read11(uint8_t pin)
{
	// READ VALUES
	int rv = read(pin);
	if (rv != 0) return rv;
```

```
	// CONVERT AND STORE
	humidity    = bits[0];  // bit[1] == 0;
	temperature = bits[2];  // bits[3] == 0;

	// TEST CHECKSUM
	uint8_t sum = bits[0] + bits[2]; // bits[1] && bits[3] both 0
	if (bits[4] != sum) return -1;

	return 0;
}

// return values:
//  0 : OK
// -1 : checksum error
// -2 : timeout
int dht::read22(uint8_t pin)
{
	// READ VALUES
	int rv = read(pin);
	if (rv != 0) return rv;

	// CONVERT AND STORE
	humidity    = word(bits[0], bits[1]) * 0.1;

	int sign = 1;
	if (bits[2] & 0x80) // negative temperature
	{
		bits[2] = bits[2] & 0x7F;
		sign = -1;
	}
	temperature = sign * word(bits[2], bits[3]) * 0.1;


	// TEST CHECKSUM
	uint8_t sum = bits[0] + bits[1] + bits[2] + bits[3];
	if (bits[4] != sum) return -1;

	return 0;
}

/////////////////////////////////////////////////////
//
// PRIVATE
//

// return values:
//  0 : OK
// -2 : timeout
int dht::read(uint8_t pin)
{
```

```
	// INIT BUFFERVAR TO RECEIVE DATA
	uint8_t cnt = 7;
	uint8_t idx = 0;

	// EMPTY BUFFER
	for (int i=0; i< 5; i++) bits[i] = 0;

	// REQUEST SAMPLE
	pinMode(pin, OUTPUT);
	digitalWrite(pin, LOW);
	delay(20);
	digitalWrite(pin, HIGH);
	delayMicroseconds(40);
	pinMode(pin, INPUT);

	// GET ACKNOWLEDGE or TIMEOUT
	unsigned int loopCnt = TIMEOUT;
	while(digitalRead(pin) == LOW)
		if (loopCnt-- == 0) return -2;

	loopCnt = TIMEOUT;
	while(digitalRead(pin) == HIGH)
		if (loopCnt-- == 0) return -2;

	// READ THE OUTPUT - 40 BITS => 5 BYTES
	for (int i=0; i<40; i++)
	{
		loopCnt = TIMEOUT;
		while(digitalRead(pin) == LOW)
			if (loopCnt-- == 0) return -2;

		unsigned long t = micros();

		loopCnt = TIMEOUT;
		while(digitalRead(pin) == HIGH)
			if (loopCnt-- == 0) return -2;

		if ((micros() - t) > 40) bits[idx] |= (1 << cnt);
		if (cnt == 0)   // next byte?
		{
			cnt = 7;
			idx++;
		}
		else cnt--;
	}

	return 0;
}
//
// END OF FILE
```

```
//
//
//    FILE: dht.h
// VERSION: 0.1.00
// PURPOSE: DHT Temperature & Humidity Sensor library for Arduino
//
//     URL: http://arduino.cc/playground/Main/DHTLib
//
// HISTORY:
// see dht.cpp file
//

#ifndef dht_h
#define dht_h

#if ARDUINO >= 100
 #include "Arduino.h"
#else
 #include "WProgram.h"
#endif

#define DHT_LIB_VERSION "0.1.00"

class dht
{
public:
	int read11(uint8_t pin);
    int read22(uint8_t pin);
	double humidity;
	double temperature;

private:
	uint8_t bits[5];  // buffer to receive data
	int read(uint8_t pin);
};
#endif
//
// END OF FILE
//
```

CHAPTER-12
OUTPUT

Fig:

Fig:


JHD 162A
READING GPS DONE

Fig:

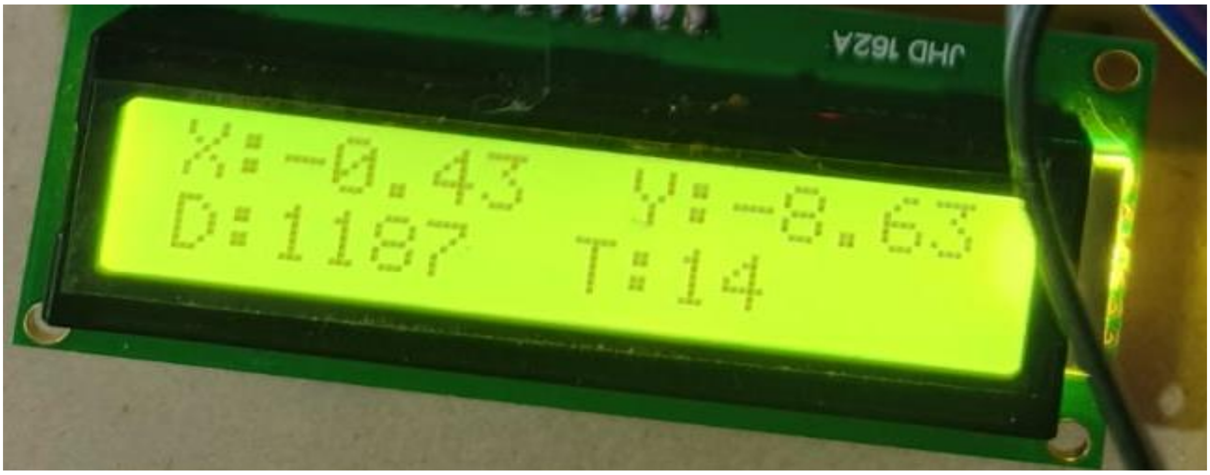

JHD 162A
X:-0.43 Y:-8.63
D:1187 T:14

Fig:


X:-0.16 Y:-10.4
D:21 T:19

Fig:

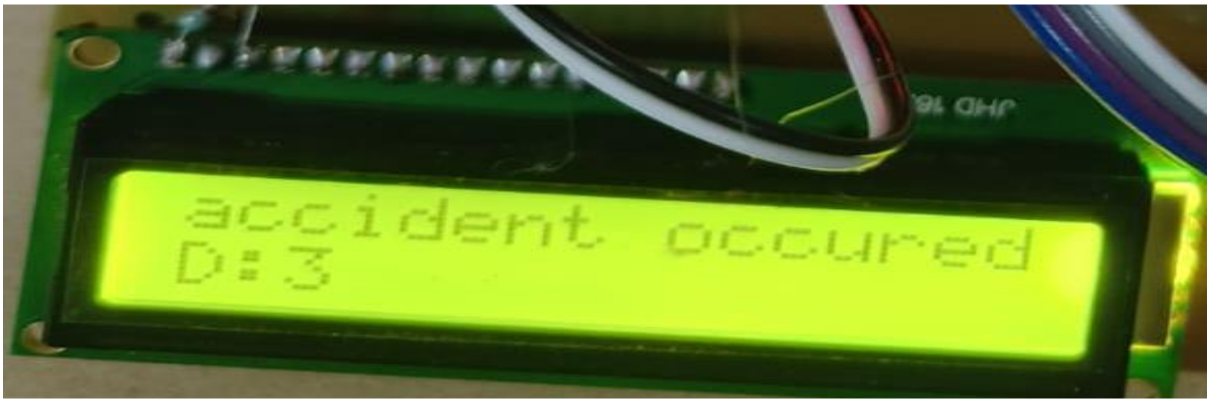

accident occured
D:3

Fig:

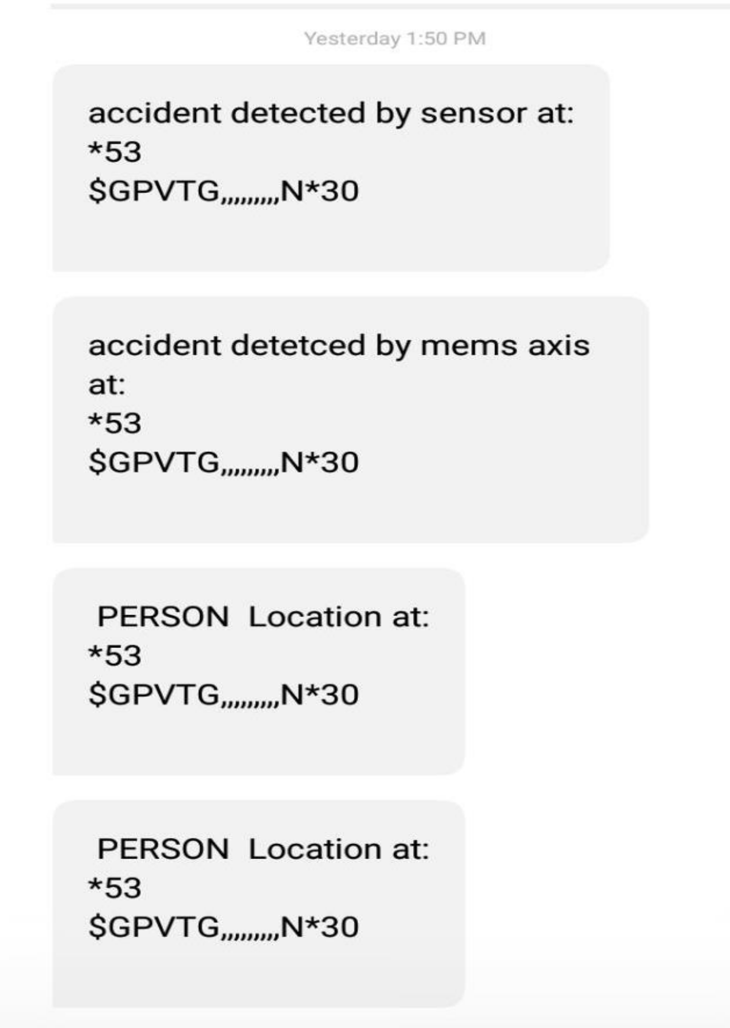

Yesterday 1:50 PM
accident detected by sensor at:
*53
$GPVTG,,,,,,,,,N*30
accident detetced by mems axis at:
*53
$GPVTG,,,,,,,,,N*30
PERSON Location at:
*53
$GPVTG,,,,,,,,,N*30
PERSON Location at:
*53
$GPVTG,,,,,,,,,N*30

Fig:

FUTURESCOPE

The future scope of wireless black boxes is vast. One potential area of development is the integration of wireless black boxes with IoT (Internet of Things) technology. This would allow for real-time monitoring and analysis of vehicle performance and status, which could help improve road safety and decrease the death rate[6]. Another potential area of development is the use of intelligent facial recognition systems in conjunction with wireless black boxes to prevent accidents and theft of vehicles.

GPS Values are sent to the Cloud for tracing the location .SMS Sent along with the location 5. Conclusion and Future Scope In this Modern era, IoT Technology is improving rapidly to help the issues mostly concern the world. This paper mainly focuses on alerting the driver from the Collision situations and using Cloud Computing Services, the location can be easily traced. Our contribution is that we proposed a low power micro-controller which can be used in the hardware implementation as its main controller in the automation of this device. with the meaningful support of the Em-bedded system IOT and Cloud computing, we strongly believe that Intelligent Vehicle Black Box using IoT will be reliable, power ef-ficient in the real time applications. In the Future scope the power can be almost decreased with the GPS and GSM modules which can easily integrated with the hardware along with a rechargeable battery and can serve the device in a longer period

CONCLUSION

More than 50% deaths in India occur due to road accidents. A considerable part of these incidents is due to lack of immediate medical assistance for the accident victim. The proposed system Wireless Black Box using MEMS accelerometer and GPS tracking for accidental monitoring of vehicles mainly aims at providing immediate assistance for accident victims even in remote areas where human help and medical services cannot be expected. In conclusion, an innovative wireless black box using MEMS accelerometer and GPS tracking system has been developed for motorcycle accidental monitoring. The system can detect the accident from an accelerometer signal using a threshold algorithm and locate the vehicle through a GPS module. After an accident is detected, short alarm massage data (alarm massage and position of accident) will be sent via GSM network. The system has been tested in real world applications and the test results are reliable without any false alarm.

REFERENCES

[1] N Purushotham, J Madan Kumar, “Wireless Black Box using MEMS Accelerometer andGPS Tracking for Accidental Monitoring of Vehicles”, March-2021.

[2] P Ajay Kumar Reddy, P Dileep Kumar, K Bhaskar Reddy, E Venkataramana, MChandrasekhar Reddy, “BLACK BOX FOR “VEHICLES”.

[3] Shaik khadar Basha1, P Sireesh Babu, “Wireless black box report for tracking of accidental monitoring in vehicles”.

[4] Ramchandra Patil, Shivaraj Hublikar,“Design and Implementation Of Car Black Box With Collision Avoidance System using ARM”.

[5] Pallakila Sai Avinash, K.ThenKumari, "Wireless Black Box Using MEMS Accelerometer and GPS Tracking for Accidental Monitoring of Vehicles”.

[6] N Watthanawisuth, T. Lomas and A Tuantranont, “Wireless Black Box Using MEMS Accelerometer and GPS Tracking for Accidental Monitoring of Vehicles”, 2-7 Jan 2020.

[7] Abid khan, Ravi Mishra “GPS –GSM Based Tracking System” International Journal of Engineering Trends and Technology- Volume 3 Issue 2- 0.