Vehicle To Vehicle Communication

Project by :

1) Gaurav Sharad Patil (B120373116)

(Department of Electronics & Telecommunication)

2)Nandkishor Vithoba Shirsat (B120373152)

(Department of Electronics & Telecommunication)

Bachelor's Project Submitted to the Department of Electronics & Telecommunication of JSPM's Rajarshi Shahu College of Engineering, Tathawade in the year 2016

CHAPTER 1:

INTRODUCTION

1. INTRODUCTON:-

Using vehicle-to-vehicle (V2V) communication, a vehicle can detect the position and movement of other vehicles up to a quarter of a kilometer away. In a real world where vehicles are equipped with a simple antenna, a computer chip and GPS (Global Positioning System) Technology, your car will know where the other vehicles are, additionally other vehicles will Know where you are too whether it is in blind spots, stopped ahead on the highway but hidden from view, around a blind corner or blocked by other vehicles.

The vehicles can anticipate and react to changing driving situations and then instantly warn the drivers with emergency warning messages. If the driver doesn't respond to the alerts message, the vehicle can bring itself to a safe stop, avoiding a collision.

1.1 **Problem Statement**:-

A driver should constantly keep an eye on vehicles closest to her/his location in order to avoid collisions. Unfortunately, the driver often does not see closest vehicles due to obstacles (other vehicles, trees, buildings, etc.). In this paper, we introduce a novel type of query called a continuous range k-nearest neighbor (CRNN) query in vehicular ad-hoc networks, and propose a scheme for query processing.

The main objective of this processing is to avoid the collision between the two vehicles when the driver should not able to control the vehicle. i.e. When driver should get disable to control the vehicle then this system should provide the proper response to the system for collision avoidance.

The aim of our project is to avoid the collision between two or more vehicles on road for the safety purpose using this system. Following objectives can be achieved through this proposed work.

- Study and understand the arduino board which emerging trends in now a days. This part includes the programming in these microcontrollers.

- To make the schematic of the required circuit and components selection

- PCB designing.

- Soldering the board and mounting it in enclosure.

- This task also consists of creating the program code in embedded C language to perform the task of LCD interfacing with the real time operation system .

- Final testing of prototype.

1.2 System operation:

V2V communications systems are composed of devices, installed in vehicles, that use dedicated short-range radio communication (DSRC) to exchange messages containing vehicle information (e.g., vehicle's speed, heading, braking status). V2V devices use this information from other vehicles and determine if a warning to the vehicle's driver is needed, which could prevent a vehicle crash.

V2V messages have a range of approximately 300 meters, which exceeds the capabilities of systems with ultrasonic sensors, cameras, and radar – in some cases, by nearly twice the distance, allowing more time to warn drivers. In addition, these radio messages can "see" around corners or "through" other vehicles addressing, for example, scenarios such as those where an oncoming vehicle emerges from behind a truck, or perhaps from a blind alley.

In those situations, V2V communications can detect the threat much earlier than radar or camera sensors.

Additionally, V2V technology can also be combined existing radar and cameras to provide even greater benefits than either approach alone. This combined approach could also augment system accuracy, becoming a foundation for realizing automated vehicles on the Nation's roadways.

For more detailed information on how NHTSA believes the various levels of vehicle automation will help reduce crashes and how on-board systems may someday work cooperatively with V2V technology, see NHTSA's Preliminary Statement of Policy on Vehicle Automation (May 2013).

CHAPTER 2:

METHODOLOGY

METHODOLOGY

2.1 APPROACHES TOWARDS THE PROJECT:-

The main objective is to exchange some crucial real time data between the vehicles using DSRC (Dedicated Short Range Communication). Here, we are using infrared communication to achieve collision avoidance on emergency breaking and a low power wireless transceiver to provide 360 degree information sharing.

2.2 BLOCK DIAGRAM:

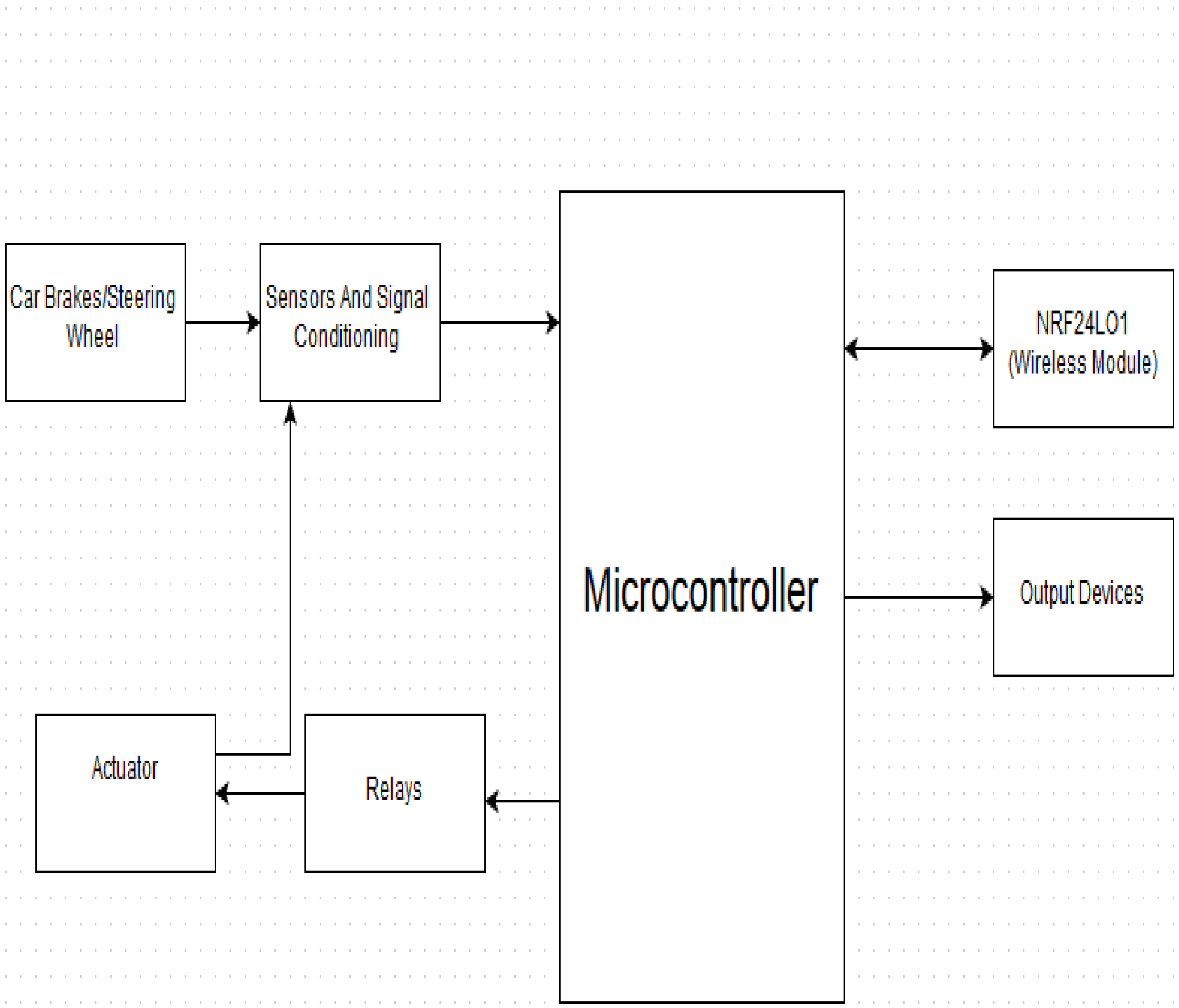

Car Brakes/Steering Wheel
Sensors And Signal Conditioning
Microcontroller
NRF24L01 (Wireless Module)
Output Devices
Actuator
Relays

Figure 1: Block diagram.

2.3 Block Diagram Description:

In this project we are designing the vehicle to vehicle communication for safety on road. The vehicles which are communicate with each other using different parameters like IR sensors and advance technology like wifi or Zigbee and much more things. Basically wireless communication which controlling and actuating parts should requires more accuracy for the better results.

The main feature of this project is arduino ATMEGA328 which perform the main function of the controlling part in this project. The consolation which gives a better solution for controlling the vehicle. The vision of the project which is totally depends on this controlling parts and their accuracy. When the IR receiver module gets the IR signal from the transmitter then trough receiver module the arduino functions the particular operation of relay operation and transmit the signal to actuate the actuator in a specific time.

2.4 Component Description:

2.4.1 Arduino ATMEGA328 Microcontroller:

The Arduino microcontroller is an easy to use yet powerful signal board computer that has gained considerable traction in the hobby and professional market. The arduino is open source, which means hardware is reasonably priced and development software is free. With the Arduino board, you can write programs and create interfacecircuits to read switches and other sensors, and to control motors and lights with very little effort.

The Duemilanove board features at ATmega 328 microcontroller operating at 5V with 2Kb of RAM,32 Kb of flash memory for storing programs and 1 Kb of EEPROM for storing parameters. The clock speed is 16 MHz, which translate to about executing about 300000 lines of C source code per second.

The board has 14 digital I/O pins and 6 analog inputs pins.There is a USB connector for talking to the host computer and a DC power jack for connecting an external 6-20 V power source, for example a 9 V battery, when running a program while not Connected to the host computer. Headers are provided for interfacing to the I/O pins using 22 g solid wire or header connectors.

2.4.2 Arduino board:

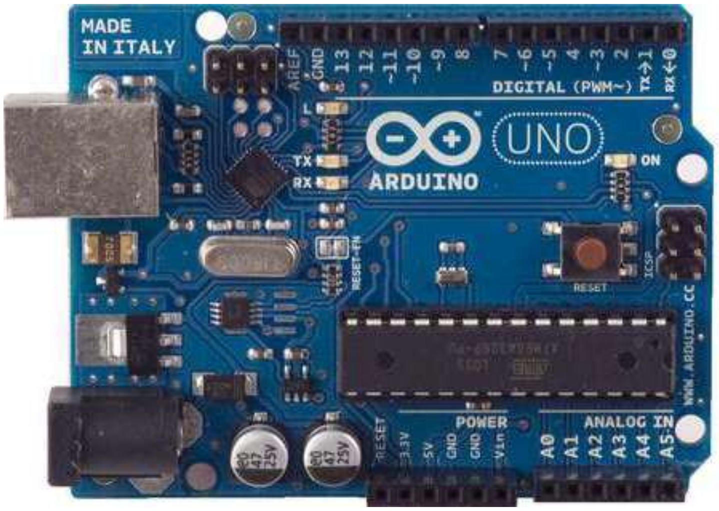


Figure No.2:Arduino Board.

The Arduino programming language is a simplified version of C/C++. If you know C, programming the Arduino will be familiar. If you do not know C, no need to worry as only a few commands are needed to perform useful functions.

An important feature of the Arduino is that you can create a control program on the host PC, download it to the Arduino and it will run automatically. Remove the USB cable connection to the PC, and the program will still run from the top each time you push the reset button.

2.4.3 Arduino Atmega328 Pin Diagram:-

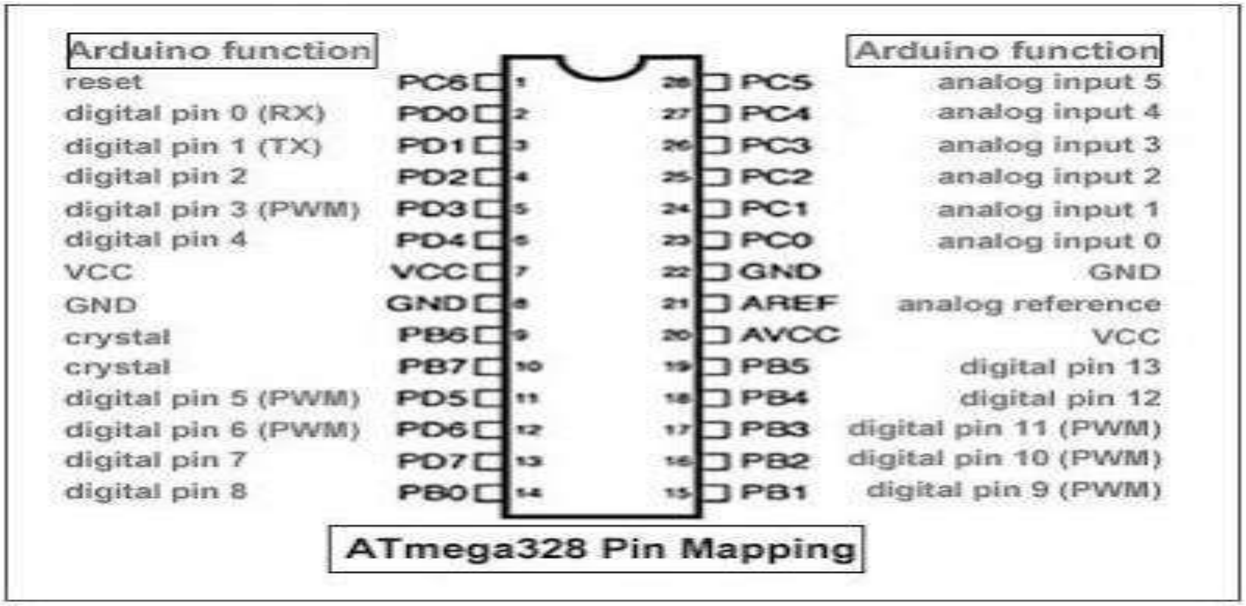

Arduino function
Arduino function
reset PC6 1 28 PC5 analog input 5
digital pin 0 (RX) PD0 2 27 PC4 analog input 4
digital pin 1 (TX) PD1 3 26 PC3 analog input 3
digital pin 2 PD2 4 25 PC2 analog input 2
digital pin 3 (PWM) PD3 5 24 PC1 analog input 1
digital pin 4 PD4 6 23 PC0 analog input 0
VCC VCC 7 22 GND GND
GND GND 8 21 AREF analog reference
crystal PB6 9 20 AVCC VCC
crystal PB7 10 19 PB5 digital pin 13
digital pin 5 (PWM) PD5 11 18 PB4 digital pin 12
digital pin 6 (PWM) PD6 12 17 PB3 digital pin 11 (PWM)
digital pin 7 PD7 13 16 PB2 digital pin 10 (PWM)
digital pin 8 PB0 14 15 PB1 digital pin 9 (PWM)
ATmega328 Pin Mapping

Figure No. 3: Arduino PIN Diagram.

Remove the battery and put the Arduino board in a closet for six months. When you reconnect the battery, the last program you stored will run. This means that you connect the board to the host PC to develop and debug your program, but once that is done, you no longer need the PC to run the program.

2.4.4 LCD Display:

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

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

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

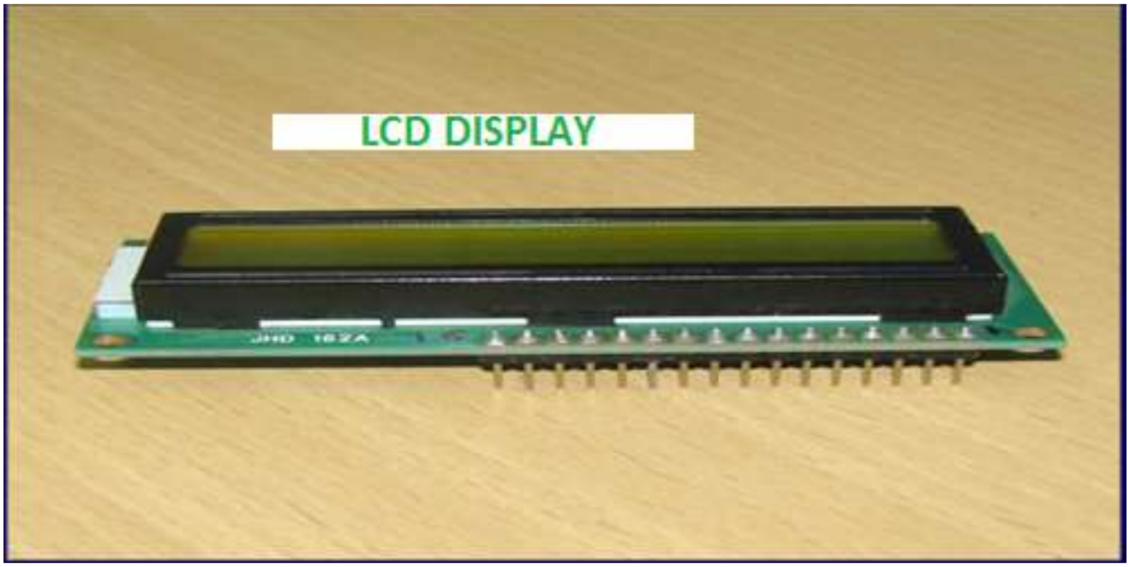


Figure No.4: LCD DISPLAY

2.4.5 LCD SPECIFICATION:-

Product Name	**16*2 LCD Module character**
Display Format	**16 character *2 lines**
Character Format	**5*8 dots**
Driving voltage signal	**+5V**
Operating temperature	**0 deg C-+50 deg C**
Module size	**80mm*36mm*13.2mm**

Table no.1: LCD Specification.

Pin Diagram:-

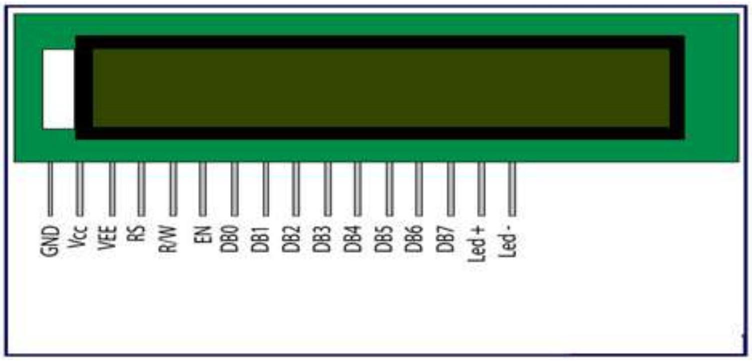


Figure No. 5: PIN Diagram Of LCD

Sr.No.	**Function**	**Name**
1	Ground (0V)	Ground
2	Supply voltage	Vcc
3	Contrast adjustment through a variable register	Vee
4	Selection command register when low and high data register	Register select
5	Low to write to the register, high to read from the register	Read/Write
6	Sends data pins when a high to read low pulse is given	Enable
7	Data pin	DB0
8	Data pin	DB1
9	Data pin	DB2
10	Data pin	DB3
11	Data pin	DB4
12	Data pin	DB5
13	Data pin	DB6
14	Data pin	DB7
15	Backlight Vcc (5V)	Led+
16	Backlight Ground	Led -

Table No.2: PIN Description.

2.4.6: L293D Detailed Description:-

Overview:-

The L293D are quadruple high-current half-H drivers. These devices are designed to drive a wide array of inductive loads such as relays, solenoids, DC and bipolar stepping motors, as well as other high-current and high-voltage loads. All inputs are TTL compatible and tolerant up to 7 V. Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo-Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN.

When an enable input is high, the associated drivers are enabled, and their outputs are active and in phase with their inputs. When the enable input is low, those drivers are disabled, and their outputs are off and in the high-impedance state. With the proper data inputs, each pair of drivers forms a full-H (or bridge) reversible drive suitable for solenoid or motor applications.

On the L293, external high-speed output clamp diodes should be used for inductive transient suppression. On the L293D, these diodes are integrated to reduce system complexity and overall system size. A VCC1 terminal, separate from VCC2, is provided for the logic inputs to minimize device power dissipation. The L293 and L293D are characterized for operation from 0°C to 70°C.

L293D PIN :-

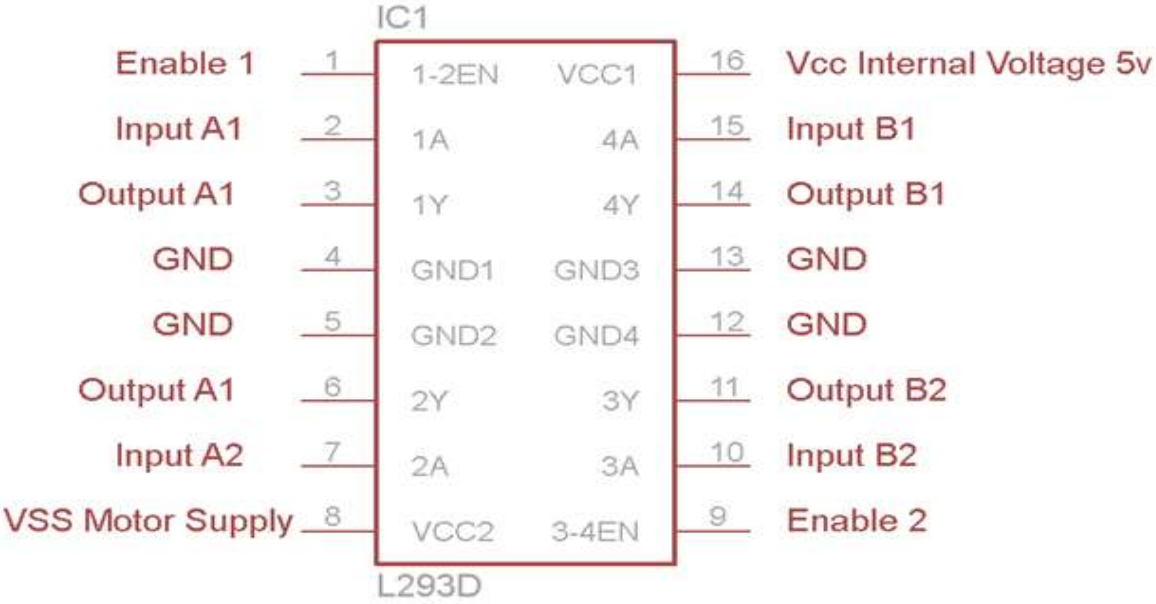


Figure No.6: L293D PIN.

Feature Description:

1) Wide Supply-Voltage Range: 4.5 V to 36 V
2) Separate Input-Logic Supply current half-H drivers.
3) The L293 is designed to provide bidirectional drive currents of up to 1 A at Internal ESD Protection voltages from 4.5 V to 36 V. The L293D is designed
4) High-Noise-Immunity Inputs to provide bidirectional drive currents of up 600-mA Output Current 1 A Per Channel (600 mA for at voltages from 4.5 V to 36 V. Both devices are L293D) designed to drive inductive loads such as relays, solenoids, DC and bipolar stepping motors, as well as Peak Output Current 2 A

Per Channel (1.2 A for other high-current/high-voltage loads in positive-L293D) The L293x has TTL-compatible inputs and high voltage outputs forinductive load driving. Current outputs can get up to 2 A using the L293.

5) There are 4 input pins for l293d, pin 2,7 on the left and pin 15 ,10 on the right as shown on the pin diagram. Left input pins will regulate the rotation of motor connected across left side and right input for motor on the right hand side. The motors are rotated on the basis of the inputs provided across the input pins as LOGIC 0 or LOGIC 1.
6) In simple you need to provide Logic 0 or 1 across the input pins for rotating the motor.

Working of L293D:-

There are 4 input pins for l293d, pin 2,7 on the left and pin 15 ,10 on the right as shown on the pin diagram. Left input pins will regulate the rotation of motor connected across left side and right input for motor on the right hand side. The motors are rotated on the basis of the inputs provided across the input pins as LOGIC 0 or LOGIC 1.

In simple you need to provide Logic 0 or 1 across the input pins for rotating the motor.

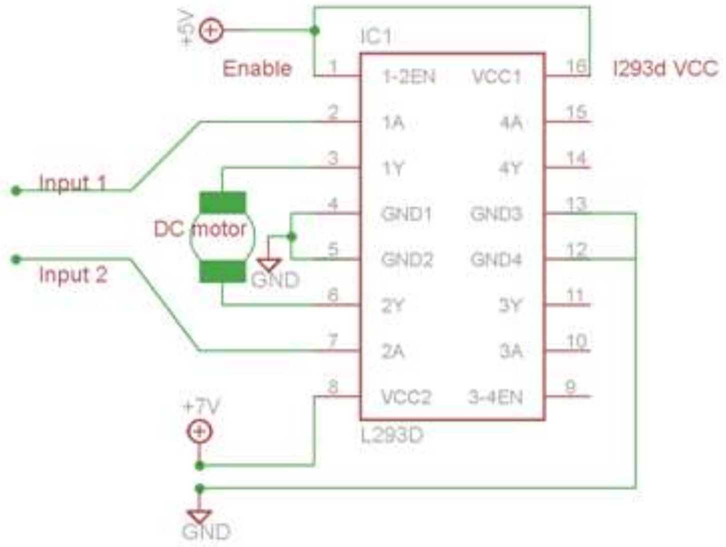


Figure No.7: L293D PIN Description.

Voltage Specification:

VCC is the voltage that it needs for its own internal operation 5v; L293D will not use this voltage for driving the motor. For driving the motors it has a separate provision to provide motor supply VSS (V supply). L293d will use this to drive the motor. It means if you want to operate a motor at 9V then you need to provide a Supply of 9V across VSS Motor supply.

The maximum voltage for VSS motor supply is 36V. It can supply a max current of 600mA per channel. Since it can drive motors Up to 36v hence you can drive pretty big motors with this l293d.

VCC pin 16 is the voltage for its own internal Operation. The maximum voltage ranges from 5v and up to 36v.

2.4.7: TSOP 1738:

The TSOP 1738 is a member of IR remote control receiver series. This IR sensor module consists of a PIN diode and a pre amplifier which are embedded into a single package. The output of TSOP is active low and it gives +5V in off state. When IR waves, from a source, with a center frequency of 38 kHz incident on it, its output goes low.

Lights coming from sunlight, fluorescent lamps etc. may cause disturbance to it and result in undesirable output even when the source is not transmitting IR signals. A band pass filter, an integrator stage and an automatic gain control are used to suppress such disturbances. TSOP module has an inbuilt control circuit for amplifying the coded pulses from the IR transmitter.

A signal is generated when PIN photodiode receives the signals. This input signal is received by an automatic gain control (AGC). For a range of inputs, the output is fed back to AGC in order to adjust the gain to a suitable level. The signal from AGC is passed to a band pass filter to filter undesired frequencies. After this, the signal goes to a demodulator and this demodulated output drives an npn transistor. The collector output of the transistor is obtained at pin 3 of TSOP module. Members of TSOP17xx series are sensitive to different center frequencies of the IR spectrum. For example TSOP1738 is sensitive to 38 kHz whereas TSOP1740 to 40 kHz center frequency.

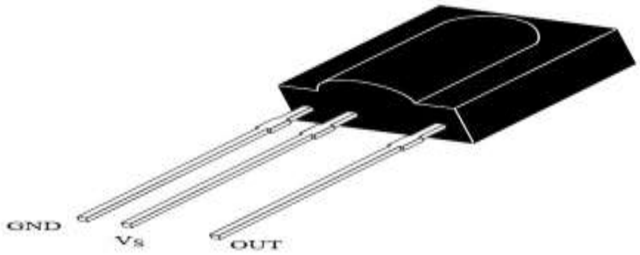


Figure No.8: TSOP

AVAILABLE TYPE FOR DIFFERENT CARRIER FREQUENCIES:-

TYPE	Fo
TSOP1730	30 KHZ
TSOP1733	33 KHZ
TSOP1736	36 KHZ
TSOP1737	36.7 KHZ
TSOP1738	38 KHZ

TSOP1740	**40 KHZ**
TSOP1756	**56 KHZ**

Table No .3: Different Carrier Frequencies.

FEATURES:

- Photo detector and preamplifier one package.
- Internal filter for PCM frequency.
- Improved shielding against electrical field disturbance.
- TTL and CMOS compatibility.
- Output active low.

2.4.8: DC MOTOR:

DC Motor are used basically drive the application as per the requirement provided in the software. The DC motor works on the 12V .To drive a dc motor, we need a dc motor driver called L293D. This DC motor driver is capable of driving 2dc motors at a time. In order to protect the motor from a back EMF generated by the dc motor while changing the direction of rotation, the dc motor driver have an internal protection suit. We can also provide the back emf protection suit by connecting 4 diode configurations across each dc motor.

Figure No.9: DC MOTOR.

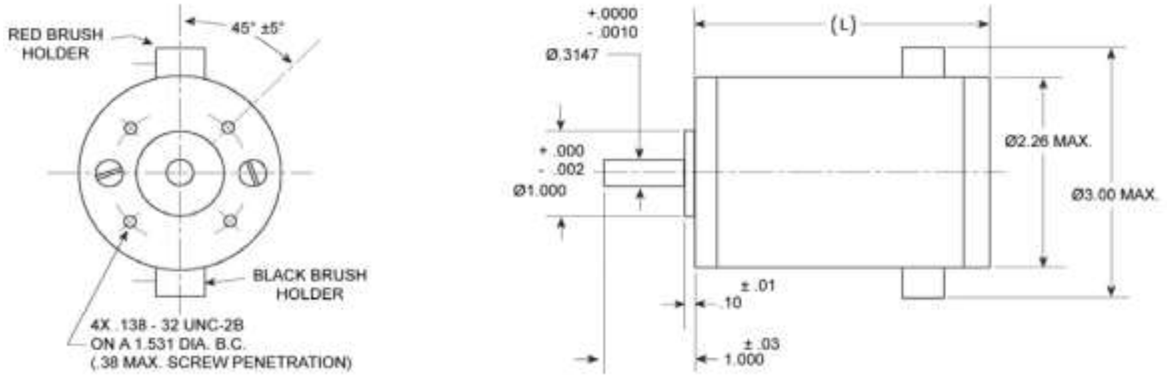


Figure No.10: Motor internal Mechanism.

2.4.9: DC MOTOR DRIVER:

The device is a monolithic integrated high voltage, high current four channel driver designed to accept standard DTL or TTL logic levels and driver inductive loads(such as relays solenoids, DC and stepper motor) and switching power transistor. To simplify use as two bridges each pair of channel is equipped with an enable input. A separate supply input is provided for the logic, allowing operation at a lower voltage and internal clamp diodes are included.

This device is suitable in switching application at 5 KHz. The L293D is assembled in a 16 lead plastic package which has 4 center pins connected together and used for heat sinking the L293D is assembled in a 20 lead surface mount which has 8 center pins connected together and used for heat sinking.

The L293D is a 16-pin Motor Driver IC which can control a set of two DC motors simultaneously in any direction. The L293D is designed to provide bidirectional drive currents of up to 600 mA (per channel) at voltages from 4.5 V to 36 V (at pin 8!). You can use it to control small dc motors - toy motors. Sometimes it can be extremely hot.

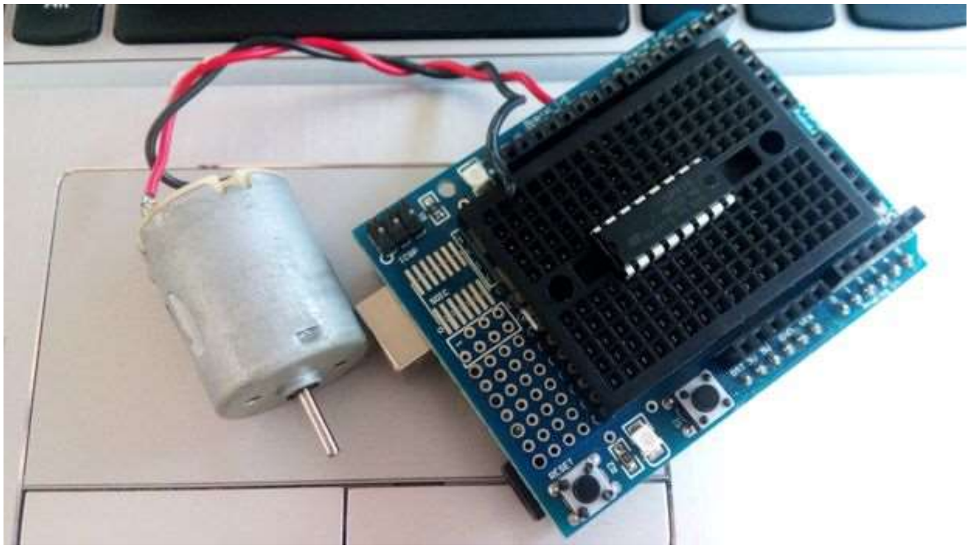


Figure No.11: Interfacing of motor with Arduino.

You can keep playing with that by clicking the "Edit" button and start making your own modifications to the code. For example try to combine parts of code to move both motors simultaneously.

Try to use analogWrite(pin, PWM value) instead digitalWrite(pin, HIGH/LOW) to control the speed of motors!

2.4.7RELAY:

A relay is usually an electromechanical device that is actuated by an electrical current. The current flowing in one circuit causes the opening or closing of another circuit. Relays are like remote control switches and are used in many applications because of their relative simplicity, long life, and proven high reliability. Relays are used in a wide variety of applications throughout industry, such as in telephone exchanges, digital computers and automation systems. Highly sophisticated relays are utilized to protect electric power systems against trouble and power blackouts as well as to regulate and control the generation and distribution of power.

In the home, relays are used in refrigerators, washing machines and dishwashers, and heating and air-conditioning controls. Although relays are generally associated with electrical circuitry, there are many other types, such as pneumatic and hydraulic. Input may be electrical and output directly mechanical, or vice versa.

It is on/off switch which uses 12V power supply. It is used to make the switch on or off. Here we used 12V single change over relay.

A relay is an electrically operated switch. A simple electromagnetic relay consists of a coil wire wrapped around a soft iron core. An iron core yoke which provide a reluctance path for a coil magnetic flux, a movable iron armature, and one or more sets of contacts (there are two in the relay pictured).

The armature is hinged to the yoke and mechanically linked to one or more sets of moving contacts. It is held in place by a spring so that when the relay is de-energized there is an air gap in the magnetic circuit. In this condition, one of the two sets contacts in the relay pictured is closed, and their function. The ensures continuity of the circuit between the moving contacts on the soldered to PCB.

Relay working:-

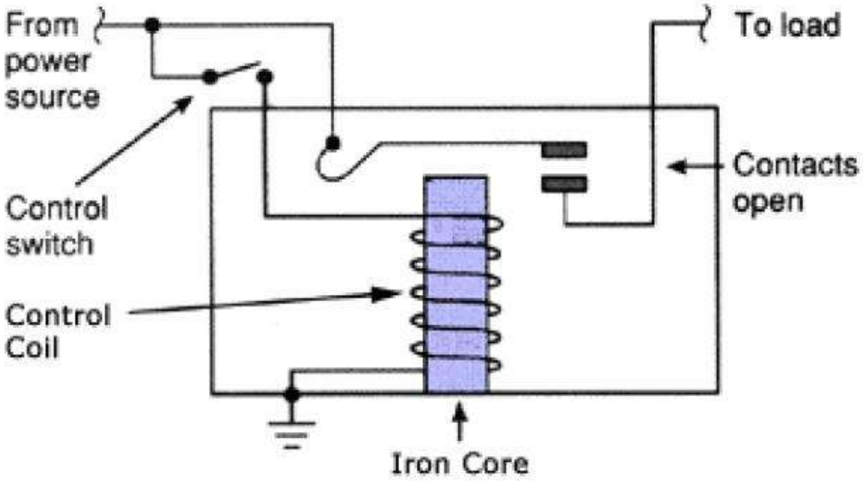


Figure No.12: Relay working.

When an electric current is passed through the coil is generates a magnetic field that activates the armature, and the consequence movement of the movable contact(s) either makes the or breaks (depending upon construction)a connection with the fixed contact. if the set of contacts was closed when the relay was de-energized, then the movement opens the contact and breaks the connection, and vice versa if the contacts were open.

When the current to the coil is switched off, the armature is returned by a forced, approximately half as strong as the current to the coil is switched off, the armature is returned back a forced, approximately half as strong as the magnetic forced, to its industrial motor starters. Most relays are manufactured to operate quickly. In a low-voltage application this reduces the noise; in a high voltage or current application it reduces arcing.

Relay points:-

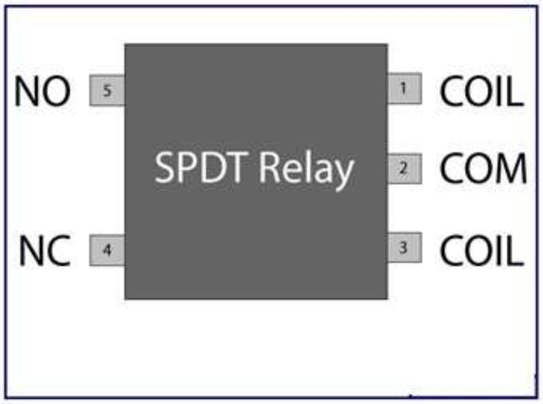


Figure No.13: Relay Points.

When the coil is energized with the direct current, a diode is often placed across the coil to dissipate the energy from the collapsing magnetic field at deactivation. Which would otherwise generate a voltage spikes dangerous to voltage semiconductor circuit components?

Some automotive relay includes a diode inside the relay case. Alternatively, a contact protection network consisting of a capacitor and resistor in series (snubber circuit) may absorb the surge. If the coil is designed t be energized with alternating current (AC), a small copper “shading ring” can be the minimum pull on the armature during the AC cycle.

CHAPTER 3:

SYSTEM DESIGN

POWER SUPPLY:-

5V supply design:-

The basic step in the designing of any system is to design the power supply required for that system. The steps involved in the designing of the power supply are as follows,

1) Determine the total current that the system sinks from the supply.
2) Determine the voltage rating required for the different components.

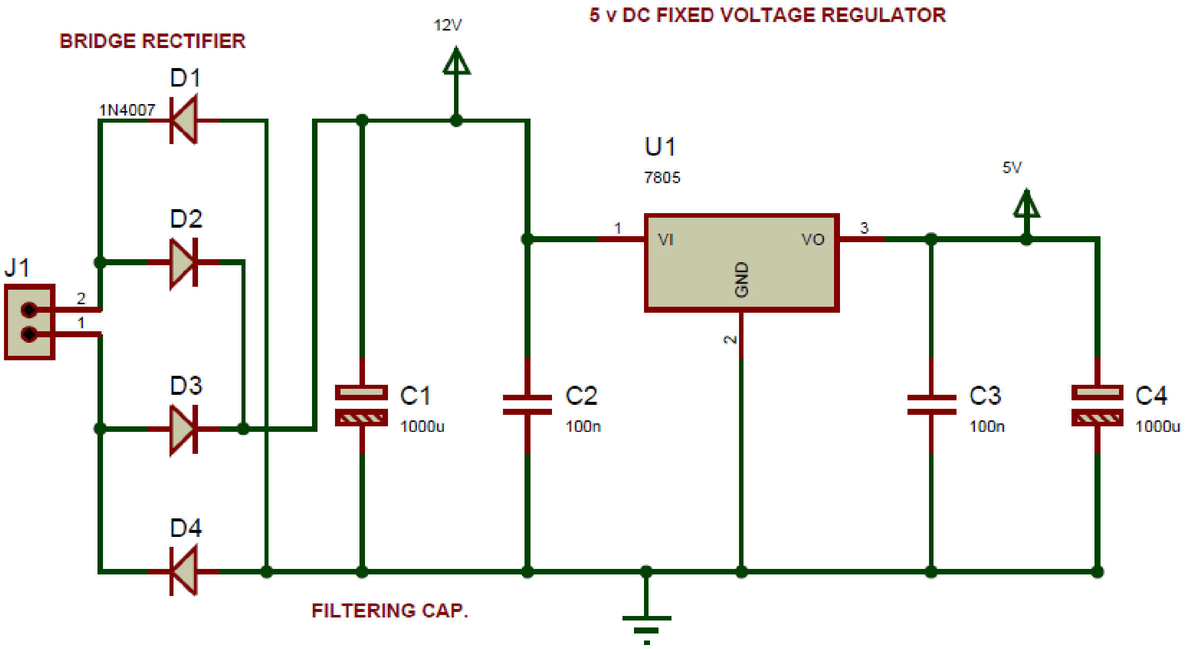


Figure No.14: 5V Power Supply.

The bridge rectifier and capacitor i/p filter produce an unregulated DC voltage which is applied at the I/P of 7805.As the minimum dropout voltage is 2v for IC 7805, the voltage applied at the input terminal should be at least 7 volts.

C1 (1000 μf / 65v)is the filter capacitor and C2 and C3 (100 nf) is to be connected across the regulator to improve the transient response of the regulator. Assuming the drop out voltage to be 2 volts, the minimum DC voltage across the capacitor C1 should be equal to 7volts (at least).

Power supply design of the Project:

The average voltage at the output of a bridge rectifier capacitor filter combination is given by,

Vin(DC) = Vm – Idc / 4 f C1

Where ,Vm = √2 Vs and Vs = rms secondary voltage

Assuming Idc to be equal to max.load current, say 100mA

C = 1000 Gf / 65v , f=50hHz

19 = Vm – 0.1 / 4*50*1000*10^{-6}

19= Vm – 0.1 / 0.2

Vm =19.5 volts

Hence the RMS secondary Voltage

Vrms = vm/ √2

= 19.5 / √2

=19,5 / 1.4421

=13.5 volts

So we can select a 15v secondary Voltage.

In our system most of the components used require 5 V as operating voltage such as micro controller, MAX 232, MCT2E etc. The total current, which our circuit sinks from the power supply, is not more than 100mA. We have used Regulator IC 7805 that gives output voltage of 5V.The minimum input voltage required for the 7805 is near about 7v.

Therefore, we have used the transformer with the voltage rating 230v-10v and current rating 500mA. The output of the transformer is 12 V AC. This Ac voltage is converted into 12 V DC by Bridge rectifier circuit.

The reasons for choosing the bridge rectifier are:-

a) The TUF is increased to 0.812 as compared the full wave rectifier.

b) The PIV across each diode is the peak voltage across the load =V_m, not $2V_m$ as in the two diode rectifier

Output of the bridge rectifier is not pure DC and contains some AC some AC ripples in it. To remove these ripples we have used capacitive filter, which smoothens the rippled output that we apply to 7805 regulators IC that gives 5V DC. We preferred to choose capacitor filters since it is cost effective, readily available.

CHAPTER 4:

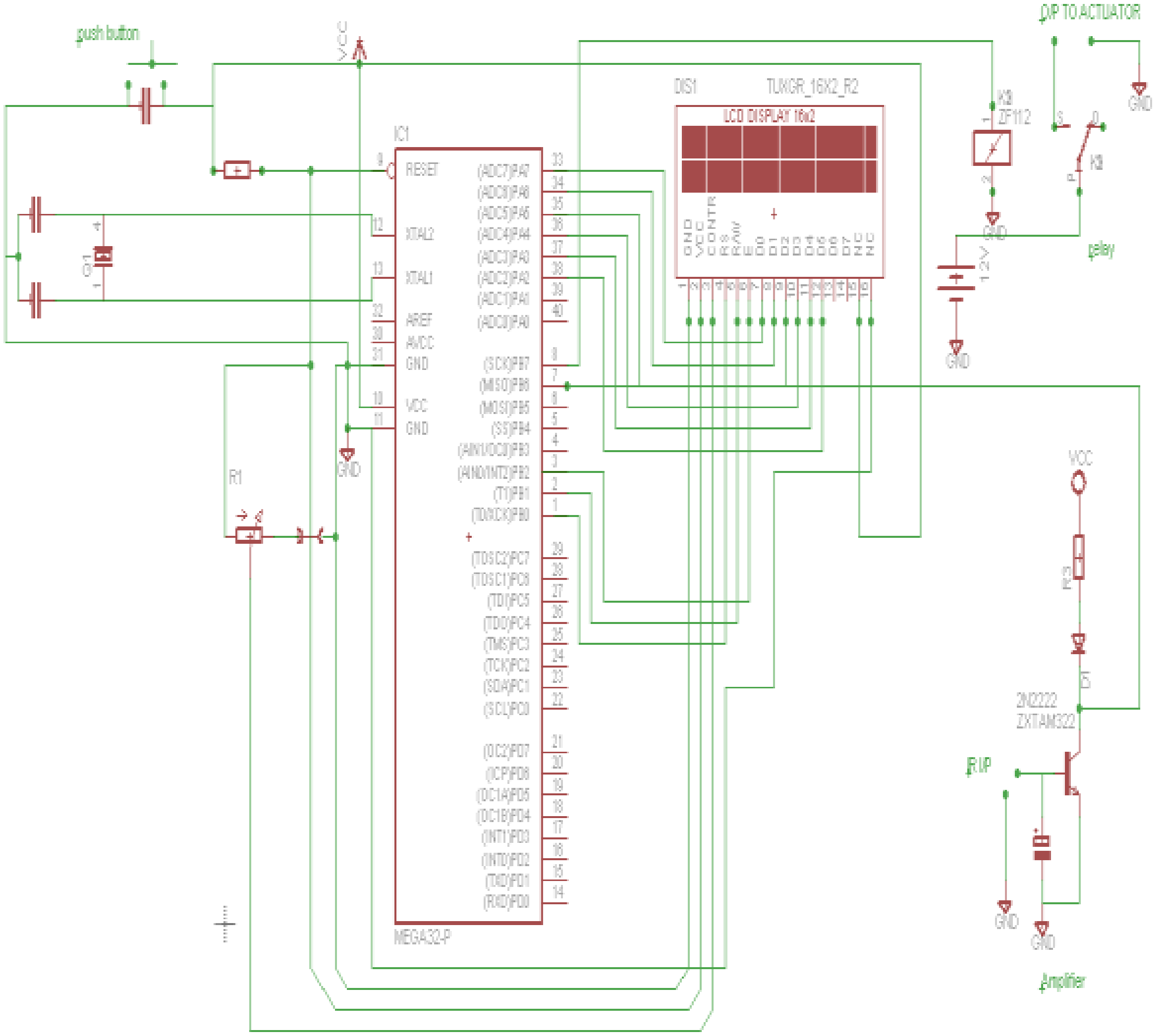
HARDWARE DESIGN

4.1 COMPONENT SPECIFICATION:

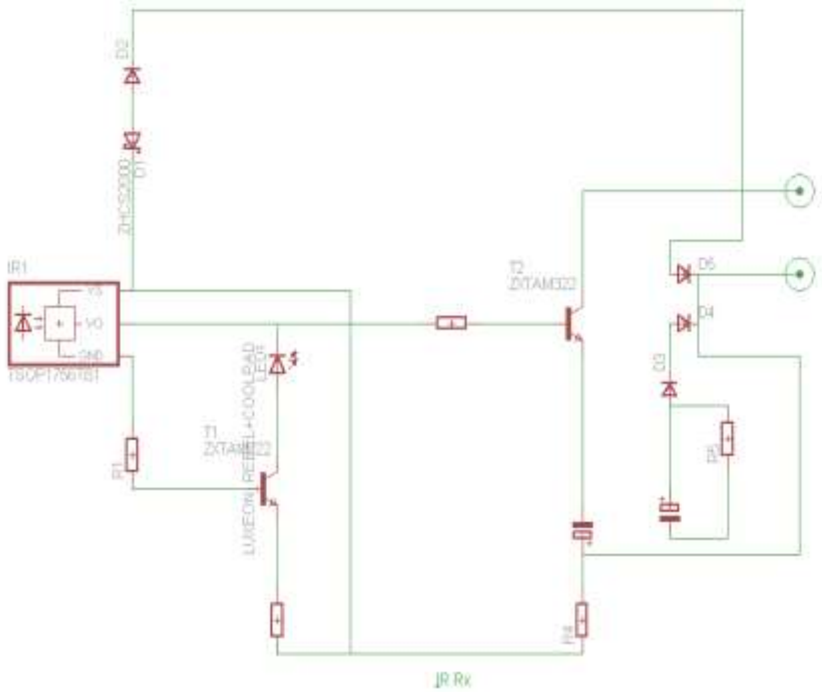
Sr.No.	Component	Rating
1.	Power Supply	5V,12V
2.	Microcontroller	ATMEGA 328
3.	IR Circuit Frequency	38 KHz
4.	LCD Display	16x2 Display
5.	DC Motor Drivers	IC L293D
6.	Regulator	IC7805,IC7812

Table No. 4: Component selection

4.2 Circuit Diagram:



IR CIRCUIT DIAGRAM:-



CHAPTER 5:

SOFTWARE DESING

5.1 Software Requirement.

Basics of Programming Languages

All sequential programming languages have four categories of instructions. First are *operation* commands that evaluate an expression, perform arithmetic, toggle states of I/O lines, and many other operations. Second are *jump* commands that cause the program to jump immediately to another part of the program that is tagged with a label. Jumps are one way to break out of the normal line-by-line processing mode.

For example, if you want a program to repeat over and over without stopping, have the last line of the program be a jump command that takes the program back to its first line. Third are *branch* commands that evaluate a condition and jump if the condition is true. For example, you might want to jump only if a number is greater than zero. Or, you might want to jump only if the state of an i/o line is low.

Fourth are *loop* commands that repeat a section of code a specified number of times. For example, with a loop you can have a light flash on and off exactly six times. Most programming languages contain a relatively small number of commands. The complexity of computers comes from combining and repeating the instructions several million times a second.

Here's a generic program.

1. Do this
2. Do that
3. Jump to instruction 6
4. Do the other thing
5. All done, sleep
6. If switch closed, do that thing you do
7. Jump to instruction 4

The function apples is everything between the set of braces that follows “apples()”. When the

function completes, the program jumps back to the line following the line that called the function.

5.2 Basic programming of Arduino:

The Arduino runs a simplified version of the C programming language, with some extensions for accessing the hardware. In this guide, we will cover the subset of the programming language that is most useful to the novice Arduino designer. All Arduino instructions are one line. The board can hold a program hundreds of lines long and has space for about 1,000 two-byte variables. The Arduino executes programs at about 300,000source code lines per sec.

Program Formatting and Syntax

Programs are entered line by line. Code is case sensitive which means "myvariable" is different than "MyVariable".

Statements are any command. Statements are terminated with a semi-colon. A classic mistake is to forget the semi-colon so if your program does not compile, examine the error text and see if you forgot to enter a colon.

Comments are any text that follows “//” on a line. For multi-line block comments, begin with“/*” and end with “*/”

Constants are fixed numbers and can be entered as ordinary decimal numbers (integer only) or in hexadecimal (base 16) or in binary (base 2) as shown in the table below Decimal Hex Binary17100 0x64 B01100100

Labels are used to reference locations in your program. They can be any combination of letters ,numbers and underscore (_), but the first character must be a letter. When used to mark allocation , follow the label with a colon. When referring to an address label in an instruction line, don't use the colon. Here's an example repeat:

```
DigitalWrite(2,HIGH);
delay(1000);
digitalWrite(2,LOW);
delay(1000);
goto repeat;
```

Use labels sparingly as they can actually make a program difficult to follow and challenging to debug. In fact, some C programmers will tell you to never use labels.

Variables are allocated by declaring them in the program. Every variable must be declared. If available is declared outside the braces of a function, it can be seen everywhere in the program. If it is declared inside the braces of a function, the variable can only be seen within that function. Variables come in several flavors including byte (8-bit, unsigned, 0 to 255), word (16-bit,unsigned, 0 to 65,536), int (16-bit, signed, -32,768 to 32,767), and long (32-bit,signed, -2,147,483,648 to 2,147,483,647). Use byte variables unless you need negative numbers or numbers larger than 255, then use int variables. Using larger sizes than needed fills upprecious memory space. Variable declarations generally appear at the top of the program.

```
byte i;
word k;
int length;
int width;
```

Variable names can be any combination of letters and numbers but must start with a letter.

Names reserved for programming instructions cannot be used for variable names and will give you an error message **Symbol s**are used to redefine how something is named and can be handy for making the code more readable. Symbols are defined with the

"#define" command and lines defining symbols should go at the beginning of your program. Here's an example without symbols for the case where an LED is connected to pin 2.

```
void setup()
{
pinMode(2,OUTPUT);
}
void loop()
{
digitalWrite(2,HIGH); // turn LED on
18
delay(1000);
digitalWrite(2,LOW); // turn LED off
delay(1000);
}
```

Here is the same using a symbol to define "LED"

```
#define LED 2 // define the LED pin
void setup()
{
pinMode(LED,OUTPUT);
}
void loop()
{
digitalWrite(LED,HIGH);
delay(500);
digitalWrite(LED,LOW);
delay(500);
}
```

Note how the use of symbols reduces the need for comments. Symbols are extremely useful to define for devices connected to pins because if you have to change the pin that the device connects to, you only have to change the single symbol definition rather than going through the whole program looking for references to that pin.

5.3 Program Structure

All Arduino programs have two functions, setup() and loop(). The instructions you place in thestartup() function are executed once when the program begins and are used to initialize. Use it to set directions of pins or to initialize variables. The instructions placed in loop are executed repeatedly and form the main tasks of the program. Therefore every program has this structure

```
void setup()
{
// commands to initialize go here
}
void loop()
{
// commands to run your machine go here
}
```

The absolute, bare-minimum, do-nothing program that you can compile and run is
void setup() {} void loop() {}The program performs no function, but is useful for clearing out any old program. Note that the compiler does not care about line returns, which is why this program works if typed all on one line.

CHAPTER 6:

RESULT

6. Results:-

Vehicle to vehicle communication system (V2V) can send and receive the vehicle information by wireless communication, and can use as a safety driving assist for driver. Currently, it is investigated to clarify an appropriate activation timing for collision information, caution and warning. This study focused on the activation timing of collision information (Provide objective information for safe driving to the driver) on V2V, and an effective activation timing of collision information, and the relationship between the activation timing and the accuracy of the vehicle position.

CHAPTER 7:

FUTURE SCOPE AND CONCLUSION

7.1 Applications:-

1. Hard Braking Ahead warning.
2. Control Loss warning.
3. Intersection collision warning.
4. Adaptive Lighting.
5. Co-operative Lane changing.
6. High speed Vehicle approach warning.

7.2 Future scope:-

Though this thesis has been successful in overcoming the limitations of existing routing approaches and improved the performance of routing in V2V environment, it does not take other areas of V2V into account. The subject not yet saturated and a lot of future research works can be pursued in the following areas.

Bandwidth: Due to the limited bandwidth of channel, there is a need for some techniques for controlling the amount of data sent to the network. This problem is addressed in congestion control. A key task for the future is to properly specify the communication requirements of V2V applications and to derive the corresponding optimal tuning of parameters of the communication system, taking into account the current channel and traffic situation.

Security:

Security is an important issue for routing in V2V, because many applications will effect life-or-death decisions and illicit tampering can have devastating consequences. The characteristics of V2V make the secure routing problem more challenging and novel than it is in other communication networks. Another challenge related to routing is efficient data dissemination and data sharing in V2V. Additional areas for improvement include the integration of privacy and security mechanisms into routing protocols and the establishment of priority routes for emergency and safety messages.

Operational Tests: Simply introducing V2V will not automatically and monotonically increase safety and efficiency. To gain better understanding of real-world V2V, field operational tests must be conducted all over the world.

7.3 Conclusion.

This project proposes a Vehicular Collision Warning Communication protocol to improve road safety. In particular, it defines congestion control policies for emergency warning messages so that a low emergency warning message delivery delay can be achieved and a large number of co-existing abnormal vehicles can be supported. It also introduces a method to eliminate redundant emergency warning messages, exploiting the natural chain effect of emergency events.

CHAPTER 8:

<u>REFERENCES</u>

8.1 "A Vehicle-to-Vehicle Communication Protocol for Cooperative Collision Warning"

1. Xue Yang University of Illinois at Urbana-Champaign xueyang@uiuc.edu

2. Jie Liu Microsoft Research liuj@microsoft.com

3. Feng Zhao Microsoft Research zhao@microsoft.com

4. Nitin H. Vaidya University of Illinois at Urbana-Champaign nhv@uiuc.edu

Conference Name:- Mobile And Ubiquitous Systems: Networking And Service,2004 MOBIQOUITOUS 2004 The First International Annual Conference.

8.2 "Hybrid vehicular communications based on V2V-V2I protocol switching"

Anna Maria Vegni/and Thomas D.C. Littley/Department of Applied Electronics, University of Roma Tre, Rome, Italyamvegni@uniroma3.Department of Electrical and Computer Engineering, Boston University, Boston MA, USAtdcl@bu.edu December 2, 2011.

Conference Name:-Vehicle Information Communication System 2011 Conference.

8.3 "Vehicle-to-Vehicle Communications Readiness of V2V Technology for Application"

Harding, J., Powell, G., R., Yoon, R., Fikentscher, J., Doyle, C., Sade, D., Lukuc, M., Simons, J., & Wang, J. (2014, August).*Vehicle-to-vehicle communications: Readiness of V2V technology for application.* (Report No. DOT HS 812 014). Washington, DC: National Highway Traffic Safety Administration.

Conference Name: **-** National Highway traffic system administration April 2014 IEEE conference.

8.4 "Data Naming in Vehicle-to-Vehicle Communications"

Lucas Wang1, Ryuji Wakikawa2, Romain Kuntz2, Rama Vuyyuru2, and Lixia Zhang1

1. Computer Science Department, University of California, Los Angeles
2. Toyota Info Technology Center, USA Mountain View, CA

Conference Name:-Computer Communication workshop (INFOCOM WKSHPS), 2012 IEEE Conference.

8.5 “Monitoring Neighboring Vehicles for Safety via V2 Communication”

1. BoXuOuriWolfson Department of Computer Science University of Illinois at Chicago {boxu, wolfson}@cs.uic.edu

2. HyungJu Cho Daum Communications Jeju, South Korea hjcho@daumcorp.com

Conference Name: - Vehicular Electronics And Safety(ICVES0),2011 IEEE International Conference.