A

PROJECT REPORT

On

AUTOMATIC RAIN SENSING CAR WIPER USING ARUDINO.

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF DIPLOMA

IN

ELECTRONICS & COMMUNICATION ENGINEERING.

SUBMITTED BY

18253-EC-009	MANISHA KUMARI.
18253-EC-004	K. SHIVA CHARAN REDDY
18253-EC-032	E. MAMATHA.
18253-EC-042	M. RANJITHSING.
18253-EC-049	G. USHA VARSHINI.

UNDER THE GUIDANCE OF

S. BASWARANI (Professor)

Department of Electronics & Communication

Engineering VNRVJIET

2nd SHIFT POLYTECHNIC

DIPLOMA IN ELECTRONICS & COMMUNICATION ENGINEERING

VALLURUPALLI NAGESWARA RAO

VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

VIGNANA JYOTHI NAGAR, BACHUPALLY, NIZAMPET (S. O),

HYDERABAD - 500 090

2020– 2021.

2nd SHIFT POLYTECHNIC

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI

INSTITUTE OF ENGINEERING & TECHNOLOGY

CERTIFICATE

This is to certify that the project report entitled **"AUTOMATIC RAIN SENSING CAR WIPER USING ARDUINO"** has been carried out at VNR VJIET, Hyderabad and submitted by the following students.

18253-EC-009	MANISHA KUMARI.
18253-EC-004	K. SHIVA CHARAN REDDY.
18253-EC-032	E. MAMATHA.
18253-EC-042	M. RANJITH SINGH RATHOD.
18253-EC-049	G. USHA VARSHINI.

In partial fulfilment of the requirements for the award **of Diploma in Electronics &Communication Engineering** to **State Board of Technical Education and Training, Telangana** at **VNR Vignana Jyothi Institute of Engineering & Technology** during the periodof 2020-2021, he/she carried out a bonafide work under the guidance and supervision of the under signed. The results embodied in this project report have not been submitted to any another University or institute for the award of any Degree.

S. BASWARANI

Assistant professor,

Electronics & Communication Engineering

VNR VJIET

Dr. Y. Padma Sai

Head of the Department,

Electronics & Communication Engineering,

VNR VJIET.

APPROVAL CERTIFICATE

Viva–Voice examination conducted for the dissertation work entitled **"AUTOMATIC RAIN SENSING CAR WIPER USING ARUDINO**".

18253-EC-009	MANISHA KUMARI.
18253-EC-004	K. SHIVA CHARAN REDDY.
18253-EC-032	E. MAMATHA.
18253-EC-042	M. RANJITH SINGH RATHOD.
18253-EC-049	G. USHA VASHINI.

is conducted on the work is approved for the award of degree of Diploma in **Electronics& Communication Engineering.**

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

Our sincere thanks to **Ms S. BASWARANI**, Department of Electronics and Communication Engineering, 2nd SHIFT POLYTECHNIC, VNR VJIET, who motivated us during project work, guiding us throughout inspire of her schedules and commitments.

We also thankful to **Dr. G. VIJAY KUMAR**, Assistant Professor& Co-coordinator of DECE, VNR VJIET, Hyderabad, for extending support to provide project work.

Our Sincere thanks to **Dr. Y. PADMA SAI** Head of the Department, Electronics and Communication Engineering, VNR VJIET & all the faculty of the Department for the encouragement and guidance provided.

We extend our sincere thanks to **Dr. B. SATYANARAYANA,** Vice Principal, 2ndpolytechnic, VNR VJIET, Hyderabad, for shift his help & co-operation in providing support for project work.

A special note of thanks to **Dr. C. D. NAIDU**, Principal, VNRVJIET and management of VNRVJIET for providing facilities for project work.

We sincerely thank all of them who have helped us either directly or indirectly during the completion of the project tenure.

18253-EC-009	MANISHA KUMARI.
18253-EC-004	K. SHIVA CHARAN REDDY.
18253-EC-032	E. MAMATHA.
18253-EC-042	M. RANJITH SINGH RATHOD.
18253-EC-049	G. USHA VARSHINI.

Place: Bachupally

Date:

DECLARATION

We do declare that this work entitled" **"AUTOMATIC RAIN SENSING CAR WIPER USING ARUDINO"** submitted in the department of Electronics and communication engineering by us to **VNR VJIET**, Bachupally in partial fulfilment of the requirement for the degree of diploma in electronic communication and engineering is a bonafide record of our own work carried out under the supervision of **Ms S. BASWARANI**.

Also, we declare that the matter embodied in this thesis has not been submitted by us in full or any part thereof for the award of any other degree/diploma in this institute or any other institute or university previously.

18253-EC-009	MANISHA KUMARI.
18253-EC-004	K. SHIVA CHARAN REDDY.
18253-EC-032	E. MAMATHA.
18253-EC-042	M. RANJITH SINGH RATHOD.
18253-EC-049	G. USHA VARSHINI.

Place: Bachupally

Date:

ABSTRACT

This project is designed to build a car wiper that automatically detects the rainfall intensity and regulates the frequency of wiper operation. It is built, using Arduino UNO board. A rain sensing module is used for measuring the intensity of rainfall. And a servo motor is used for controlling the wiper movements. An LCD module is also attached to the controller for displaying the rainfall intensity. By measuring the amount of rainfall , controller will adjust the speed of servo motor . Servo is controlled by generating PWM signal at its signal line.

Objectives of the Project

- Measuring the amount of rainfall.
- Displaying the intensity of rainfall in an LCD module.
- Controlling the servo motor speed according to the output from the sensor module.

CONTENTS

LIST OF FIGURES PAGE NO

LIST OF TABLES PAGE NO

CHAPTER 1

INTRODUCTION

A car wiper is a device which is used to remove droplets of rainwater from a windscreen. Nowadays, each and every vehicle is provisioned with the wiper to avoid the accidents and to decrease the human intervention in controlling the wiper to ensure luxury. A wiper generally consists of a metal arm and a long rubber blade. In some vehicles, pneumatic power is used. Here, the metal arm gets powered by an electric motor. The blade moves in clock-wise and counter clock-wise direction on the glass, pushing the water from the surface of the glass. Modification of speed is automatically done based on the intensity of the rainfall. Two synchronized radial type arms are used in most of the automobiles, whereas pantograph arms are used in commercial automobiles. Wipers are automated in many ways. These days' automobiles consist of a series of mechanical parts which are automated by an electric motor. In this, we propose an unmanned wiper which senses rain and starts automatically and switches off automatically when the rain stops. By using this, there will be no need for physical intervention of human to control the speed of the wiper. For this purpose, we use a rain sensor to detect the rain and then the signal is managed by Arduino and takes the required action. Over the last ten years, the advancement in the automobile industry has been increased to find modern techniques to increase safety. There are many reasons behind the vehicles which are not equipped with automatic car wipers. Many automobile companies made an attempt to construct the automatic car wiper at low cost which is not only economical but also efficient. This paper is all about the attempt they tried to construct.

In the present day situation, only luxury vehicles are equipped with automatic rain sensing car wipers. Our work is created to show the need to use an automatic car wiper system which starts automatically when the rain starts. The speed of the wiper is also adjusted automatically based on the intensity of the rainfall. Such a system ensures the safety of a ride. A lot of reasons are responsible for accidents but the major reason for the occurrence of accidents during the rainy season is due to a lack of proper vision. The objective is to construct a self-starting car wiper system which starts automatically on sensing the rainfall. Automatic adjustments to the wiper speed are made based on the intensity of rainfall. The project is constructed using Arduino, Rain sensor, Servo motor and an LCD Module which displays the intensity of the rainfall. Adjustment to the speed of the wiper is made according to the intensity of rainfall which improves and ensures the safety. This project is a small step towards the comfortability and to save our time.

CHAPTER-2

DESCRIPTION OF COMPONENTS

2.1 ARDUINO UNO

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board. The Arduino platform has become quite popular with people just starting out with electronics, and for good reason. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware (called a programmer) in order to load 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 out the functions of the micro-controller into a more accessible package.

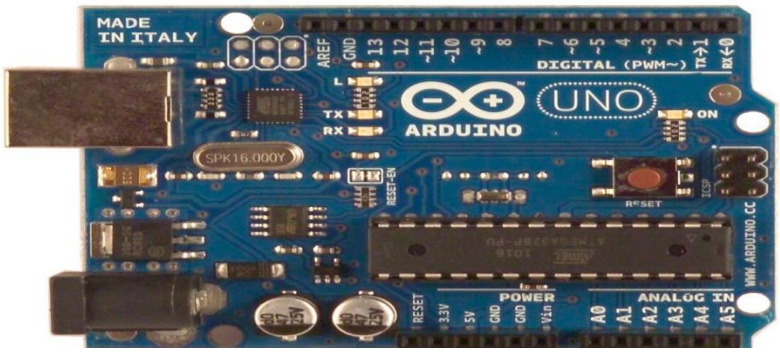


Fig: 2.1: Arduino uno

2.1.1 Pin description

It comprises 14-digit I/O pins. From these pins, 6-pins can be utilized like PWM outputs. This board includes 14 digital input/output pins, Analog inputs-6, a USB connection, quartz crystal-16 MHz, a power jack, a USB connection, resonator-16Mhz, a power jack, an ICSP header and an RST button.

The power supply of the Arduino can be done with the help of an exterior power supply otherwise USB connection. The exterior power supply (6 to 20 volts) mainly includes a battery or an AC to DC adapter.

The connection of an adapter can be done by plugging a center-positive plug (2.1mm) into the power jack on the board. The battery terminals can be placed in the pins of VIN as well as GND. The power pins of an Arduino board include the following.

VIN: The input voltage or VIN to the Arduino while it is using an exterior power supply opposite to volts from the connection of USB or else RPS (regulated power supply). By using this pin, one can supply the voltage.

5Volts: The RPS can be used to give the power supply to the microcontroller as well as components which are used on the Arduino board. This can approach from the input voltage through a regulator.

3V3: A 3.3 supply voltage can be generated with the on board regulator, and the highest draw current will be 50 mA.

GND: GND (ground) pins

Memory: The memory of an ATmega328 microcontroller includes 32 KB and 0.5 KB memory is utilized for the Boot loader), and also it includes SRAM-2 KB as well as EEPROM1KB.

Input and Output: We know that an arguing Uno R3 includes 14-digital pins which can be used as an input otherwise output by using the functions like pin Mode (), digital Read (), and digital Write (). These pins can operate with 5V, and every digital pin can give or receive 20mA, & includes a 20k to 50k ohm pull up resistor. The maximum current on any pin is 40mA which cannot surpass for avoiding the microcontroller from the damage. Additionally, some of the pins of an Arduino include specific functions.

Serial Pins: The serial pins of an Arduino board are TX (1) and RX (0) pins and these pins can be used to transfer the TTL serial data. The connection of these pins can be done with the equivalent pins of the ATmega8 U2 USB to TTL chip.

External Interrupt Pins: The external interrupt pins of the board are 2 & 3, and these pins can be arranged to activate an interrupt on a rising otherwise falling edge, a low-value otherwise a modify in value.

PWM Pins: The PWM pins of an Arduino are 3, 5, 6, 9, 10, & 11, and gives an output of an 8-bit PWM with the function analog Write ().

SPI (Serial Peripheral Interface) Pins: The SPI pins are 10, 11, 12, 13 namely SS, MOSI, SPI communication with the help of the SPI library.

LED Pin: An arguing board is inbuilt with a LED using digital pin-13. Whenever the digital pin is high, the LED will glow otherwise it will not glow.

TWI (2-Wire Interface) Pins: The TWI pins are SDA or A4, & SCL or A5, which can support the communication of TWI with the help of Wire library.

AREF (Analog Reference) Pin: An analogue reference pin is the reference voltage to the inputs of an analogue i/p using the function like analog reference.

Reset (RST) Pin: This pin brings a low line for resetting the microcontroller, and it is very useful for using an RST button toward shields which can block the one over the Arduino R3 board.

Communication: The communication protocols of an Arduino Uno include SPI, I2C, and UART serial communication.

UART: An Arduino Uno uses the two functions like the transmitter digital pin1 and the receiver digital pin0. These pins are mainly used in UART TTL serial communication.

I2C: An Arduino UNO board employs SDA pin otherwise A4 pin & A5 pin otherwise SCL pin is used for I2C communication with wire library. In this, both the SCL and SDA are CLK signal and data signal.

SPI Pins: The SPI communication includes MOSI, MISO, and SCK.

MOSI (Pin11): This is the master out slave in the pin, used to transmit the data to the devices

MISO (Pin12): This pin is a serial CLK, and the CLK pulse will synchronize the transmission of which is produced by the master.

SCK (Pin13): The CLK pulse synchronizes data transmission that is generated by the master. Equivalent pins with the SPI library are employed for the communication of SPI. ICSP (in-circuit serial programming) headers can be utilized for programming **AT mega microcontroller** directly with the boot loader.

2.1.2 Arduino Uno R3 Programming

- The programming of an Arduino Uno R3 can be done using IDE software. The microcontroller on the board will come with pre-burned by a boot loader that permits to upload fresh code without using an exterior hardware programmer.
- The communication of this can be done using a protocol like STK500.

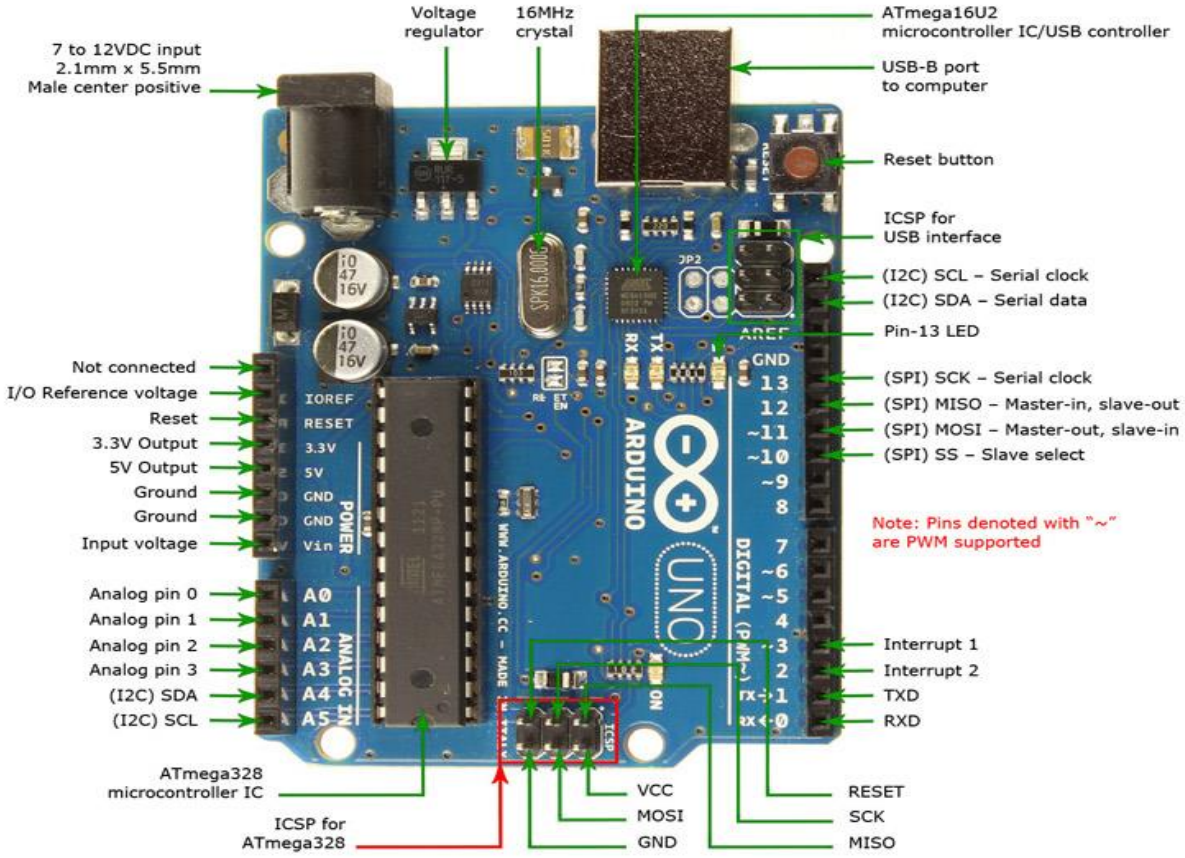


Fig 2.2: Arduino pin diagram

2.1.3 Different Types of Arduino Boards

The list of Arduino boards includes the following such as

1. Arduino Uno (R3)
2. Lily pad Arduino
3. Red Board
4. Arduino Mega (R3)
5. Arduino Leonardo

2.2 SG-90 SERVO MOTOR

A servo motor is a rotary actuator or linear actuator that allows for exact control of sharp or linear position,velocity and acceleration.By measuring the amount of rainfall , controller will adjust the speed of servo motor.Servo is controlled by generating PWM signal at its signal line.As mentioned earlier a servo motor is used here for wiper movements. Servo motor is a special kind of high torque dc motor whose shaft can be adjusted to desirable position by generating an appropriate PWM signal at its signal line.The position of the servo will be at 0 degree at the beginning.

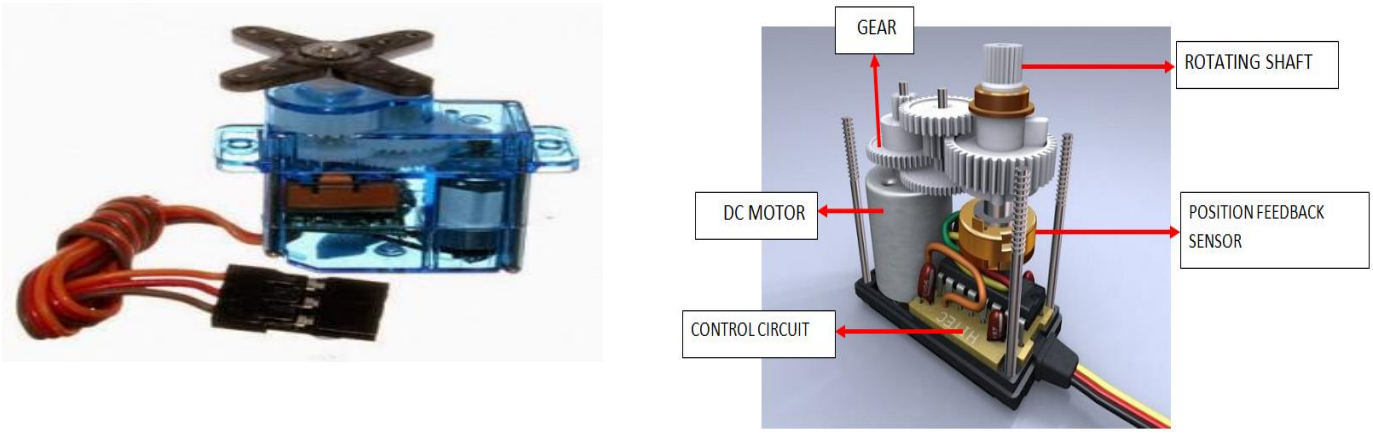
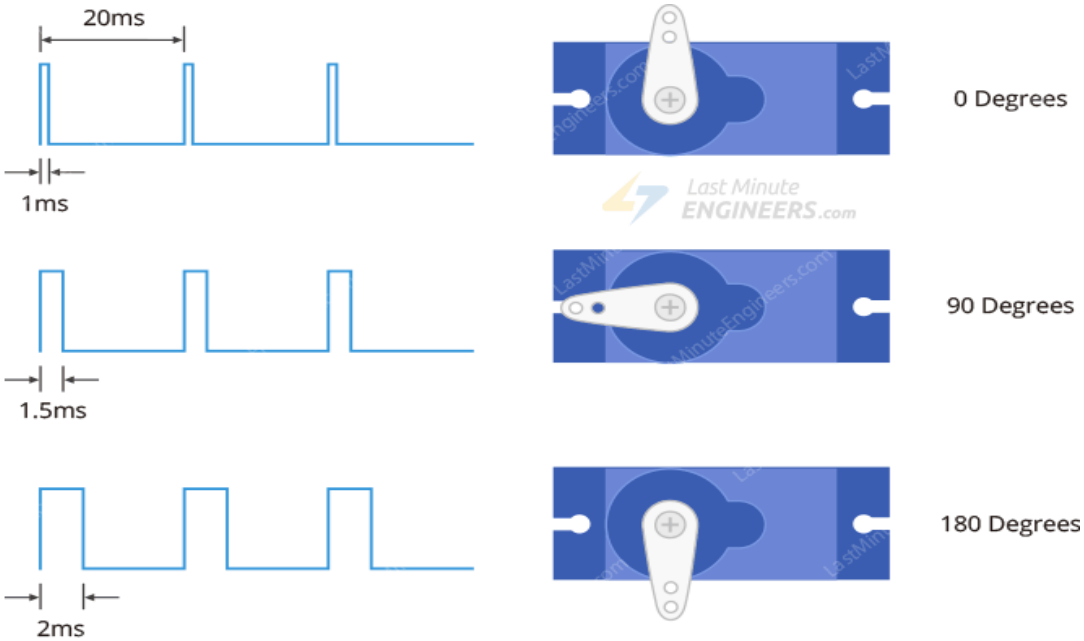


Fig 2.3: SG-90 SERVO MOTOR

You can control the servo motor by sending a series of pulses to the signal line. A conventional analog servo motor expects to receive a pulse roughly every 20 milliseconds (i.e. signal should be 50Hz).

The length of the pulse determines the position of the servo motor.



- If the pulse is high for 1ms, then the servo angle will be zero.
- If the pulse is high for 1.5ms, then the servo will be at its center position.
- If the pulse is high for 2ms, then the servo will at 180 degrees.
- Pulses ranging between 1ms and 2ms will move the servo shaft through the full 180 degrees of its travel.

2.2.1 Specifications:

- Operating Voltage is +5V typically
- Torque: 2.5kg/cm
- Operating speed is 0.1s/60°
- Gear Type: Plastic
- Rotation : 0°-180°
- Weight of motor : 9gm
- Package includes gear horns and screws

2.2.2 Applications:

- Used as actuators in many robots like Biped Robot, Hexapod, robotic arm etc..
- Commonly used for steering system in RC toys
- Robots where position control is required without feedback
- Less weight hence used in multi DOF robots like humanoid robots

2.3 LIQUID CRYSTAL DISPLAY

We come across LCD displays everywhere around us. Computers, calculators, television sets, mobile phones, digital watches use some kind of display to display the time.

An LCD is an electronic display module which uses liquid crystal to produce a visible image. The 16×2 LCD display is a very basic module commonly used in DIYs and circuits. The 16x2 translates a display 16 characters per line in 2 such lines.

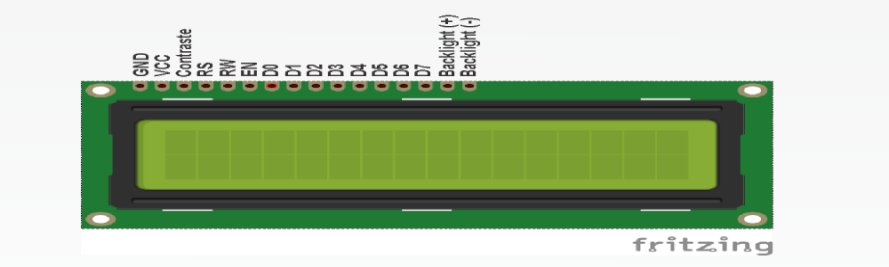


Fig 2.4: 16x2 LCD

2.3.1 Construction

Simple facts that should be considered while making an LCD:

- The basic structure of LCD should be controlled by changing the applied current.
- We must use a polarized light.
- Liquid crystal should able be to control both of the operation to transmit or can also able to change the polarized light.

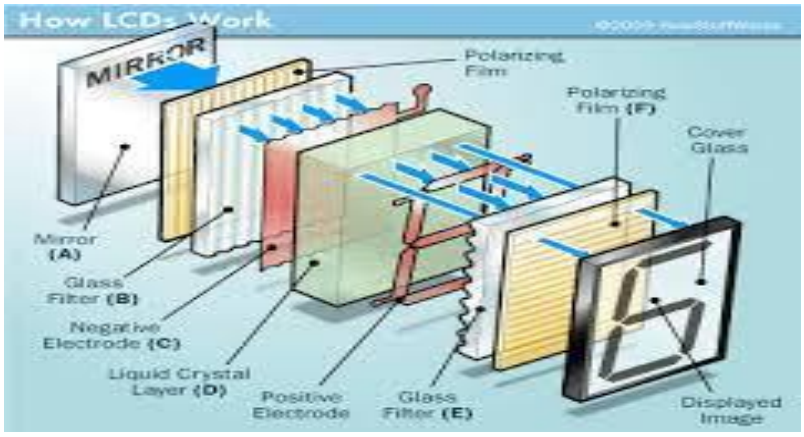


Fig 2.5: construction of LCD

As mentioned above that we need to take two polarized glass pieces filter in the making of the liquid crystal. The glass which does not have a polarized film on the surface of it must be rubbed with a special polymer which will create microscopic grooves on the surface of the polarized glass filter. The grooves must be in the same direction of the polarized film. Now we have to add a coating of pneumatic liquid phase crystal on one of the polarized filters of the polarized glass. The microscopic channel causes the first layer molecule to align with filter orientation. When the right angle appears at the first layer piece, we should add a second piece of glass with the polarized film. The first filter will be naturally polarized as the light strikes it at the starting stage.

Thus, the light travels through each layer and guided on the next with the help of molecule. The molecule tends to change its plane of vibration of the light in order to match their angle. When the light reaches to the far end of the liquid crystal substance, it vibrates at the same angle as that of the final layer of the molecule vibrates. The light is allowed to enter into the device only if the second layer of the polarized glass matches with the final layer of the molecule.

2.3.2 Working

The principle behind the LCD's is that when an electrical current is applied to the liquid crystal molecule, the molecule tends to untwist. This causes the angle of light which is passing through the molecule of the polarized glass and also cause a change in the angle of the top polarizing filter. As a result, a little light is allowed to pass the polarized glass through a particular area of the LCD. Thus, that particular area will become dark compared to other. The LCD works on the principle of blocking light. While constructing the LCD's, a reflected mirror is arranged at the back. An electrode plane is made of indium-tin oxide which is kept on top and a polarized glass with a polarizing film is also added on the bottom of the device. The complete region of the LCD has to be enclosed by a common electrode and above it should be the liquid crystal matter.

Next comes to the second piece of glass with an electrode in the form of the rectangle on the bottom and, on top, another polarizing film. It must be considered that both the pieces are kept at right angles. When there is no current, the light passes through the front of the LCD it will be reflected by the mirror and bounced back

2.3.3 Pin configuration

Pin Number	**Pin Name**	**Description**
1	Vss (Ground)	Ground pin connected to system ground
2	Vdd(+5v)	Powers the LCD with +5v (4.7v to 5.3v)
3	VE (contrast v)	Decides the contrast level of display. Grounded to get maximum contrast
4	Register selects	Connected to microcontroller to shit between command/data register
5	Read/write	Used to read or write data normally grounded to write data to lcd
6	Enable	Connected to microcontroller pin and toggled between 1 and 0 for data acknowledgement
7	Data pin 0	Data pins 0 to 7 forms a 8-bit data line. They can be connected to send 8-bit data These LCD'S can also operate on 4-bit mode in such
8	Data pin 1	
9	Data pin 2	
10	Data pin 3	
11	Data pin 4	

12	Data pin 5	case data pin 4,5,6 and 7 will be left free
13	Data pin 6	
14	Data pin 7	
15	Led positive	Backlight LED pin positive terminal
16	Led negative	Backlight LED pin negative terminal

Table 2.1: pin configuration

2.3.4 Features of 16×2 LCD module

1. Operating Voltage is 4.7V to 5.3V.
2. Current consumption is 1mA without backlight.
3. Alphanumeric LCD display module, meaning can display alphabets and numbers.
4. Consists of two rows and each row can print 16 characters.
5. Each character is built by a 5×8-pixel box.
6. Can work on both 8-bit and 4-bit mode.
7. It can also display any custom generated characters.
8. Available in Green and Blue Backlight.

2.3.5 Important command codes for LCD

S.NO	Hex Code	Command to LCD instruction Register
1	01	Clear display screen
2	02	Return home
3	04	Decrement cursor (shift cursor to left)
4	06	Increment cursor (shift cursor to right)
5	05	Shift display to right
6	07	Shift display to left
7	08	Display off, cursor off
8	0A	Display off, cursor on
9	0C	Display on, cursor off
10	0E	Display on, cursor blinking
11	0F	Display on, cursor blinking
12	10	Shift cursor position to left
13	14	Shift cursor position to right
14	18	Shift the entire display to left
15	1C	Shift the entire display to the right
16	80	Force cursor to beginning (1st line)

17	C0	Force cursor to beginning (2nd line)
18	38	2 lines and 5*7 matrix

Table 2.2: commands codes for LCD.

2.3.6 Displaying Custom Characters on 16X2 LCD

Generating custom characters on LCD is not very hard. It requires the knowledge about custom generated random-access memory (CG-RAM) of LCD and the LCD chip controller. Most LCDs contain Hitachi HD4478 controller.

CG-RAM is the main component in making custom characters. It stores the custom characters once declared in the code. CG-RAM size is 64 bytes providing the option of creating eight characters at a time. Each character is eight bytes in size.

CG-RAM address starts from 0x40 (Hexadecimal) or 64 in decimal. We can generate custom characters at these addresses. Once we generate our characters at these addresses, now we can print them on the LCD at any time by just sending simple commands to the LCD. Character addresses and printing commands are below.

CG-RAM Characters	CG-RAM Address (Hexadecimal)	Commands to display Generated Characters (Decimal)
1st Character	0x40	0
2nd Character	0x48	1
3rd Character	0x50	2
4th Character	0x58	3
5th Character	0x60	4
6th Character	0x68	5
7th Character	0x70	6
8th Character	0x78	7

Table 2.3: Displaying characters.

In the table above you can see starting addresses for each character with their printing commands. The first character is generated at address 0x40 to 0x47 and is printed on LCD by just

sending simple command 0 to the LCD. The second character is generated at address 0x48 to 0x55 and is printed by sending data.

2.3.7 Interfacing LCD with Arduino

RS pin of the LCD is connected to the pin 12 of the Arduino. The LCD of R/W pin is connected to the ground. The pin 11 of the Arduino is connected to the enable signal pin of LCD module. The LCD module & Arduino module are interfaced with the 4-bit mode in this project. Hence there are four input lines which are DB4 to DB7 of the LCD. This process very simple, it requires fewer connection cables and also, we can utilize the most potential of the LCD module. The digital input lines (DB4-DB7) are interfaced with the Arduino pins from 5- 2. To adjust the contrast of the display here we are using a 10K potentiometer. The current through the back LED light is from the 560-ohm resistor. The external power jack is provided by the board to the Arduino. Using the PC through the USB port the Arduino can power. Some parts of the circuit can require the +5V power supply it is taken from the 5V source on the Arduino board.

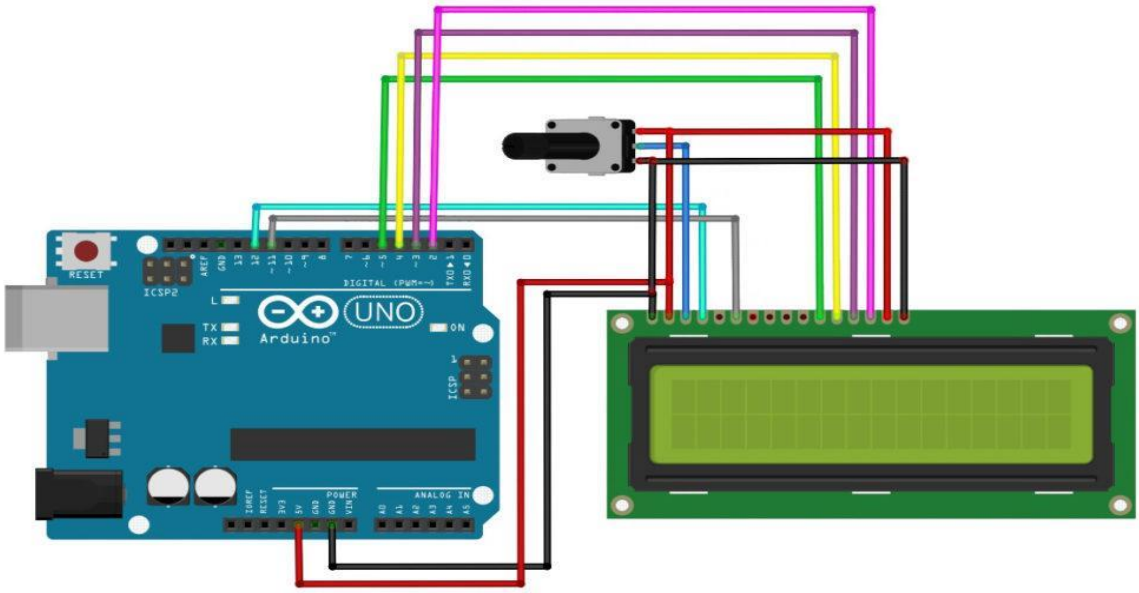


Fig 2.6: Interfacing LCD with Arduino uno.

2.4 JUMPER WIRES

A jump wire (also known as jumper wire, or jumper) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering.

Individual jump wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment.

2.4.1 Types

Fig 2.7: Jumper wires with crocodile clips.

Jumper wires at the end of a multi-coloured ribbon cable are used to connect the pin header at the left side of a blue USB2Serial board to a white breadboard below. Another jumper cable ending in a USB micro male connector mates to the right side of the USB2Serial board. Red and black tinned jump wires can be seen on the breadboard.

There are different types of jumper wires. Some have the same type of electrical connector at both ends, while others have different connectors. Some common connectors are:

1. Solid tips – are used to connect on/with a breadboard or female header connector. The arrangement of the elements and ease of insertion on a breadboard allows increasing the mounting density of both components and jump wires without fear of short-circuits. The jump wires vary in size and colour to distinguish the different working signals.
2. Crocodile clips – are used, among other applications, to temporarily bridge sensors, buttons and other elements of prototypes with components or equipment that have arbitrary connectors, wires, screw terminals, etc.
3. Banana connectors – are commonly used on test equipment for DC and low-frequency AC signals.
4. Registered jack (RJnn) – are commonly used in telephone (RJ11) and computer networking (RJ45).

5. Which is used to connect antennas and other components to network cabling. Jumpers are also used in base stations to connect antennas to radio units. Usually, the most bendable jumper cable diameter is 1/2"RCA connectors – are often used for audio, low resolution composite video signals, or other low-frequency applications requiring a shielded cable.
6. RF connectors – are used to carry radio frequency signals between circuits, test equipment, and antennas.
7. RF jumper cables Jumper cables is a smaller and more bendable corrugated cable.

Fig 2.8: male to female connectors.

A connector type with pins instead of holes. These connectors are inserted into a female connector. Good examples of **male connectors** are power plugs and coaxial cables. In the example picture, the power cord connector on the left-side with holes is a female connector, and on the right-side with pins that connects to the wall outlet is a male connector.

A female connector is a connector attached to a wire, cable, or piece of hardware, having one or more recessed holes with electrical terminals inside, and constructed in such a way that a plug with exposed conductors (male connector) can be inserted snugly into it to ensure a reliable physical and electrical connection. A female connector is also known as a jack, outlet, or receptacle. This type of connector can be recognized by the fact that, when it is disconnected therefore are not likely to make accidental contact with external objects or conductors.

The most common female connector is a two- or three-prong electrical outlet, also known as a wall outlet. Other often-encountered examples include telephone jacks, the jacks for headsets, the chassis connectors for coaxial cable, and some D-shell connectors for computer serial and parallel ports.

Fig 2.8 (a): male to male connector

2.5 BREADBOARD

A breadboard is a solderless device for temporary prototype with electronics and test circuit designs. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate. The breadboard has strips of metal underneath the board and connect the holes on the top of the board. The metal strips are laid out as shown below. Note that the top and bottom rows of holes are connected horizontally and split in the middle while the remaining holes are connected vertically.

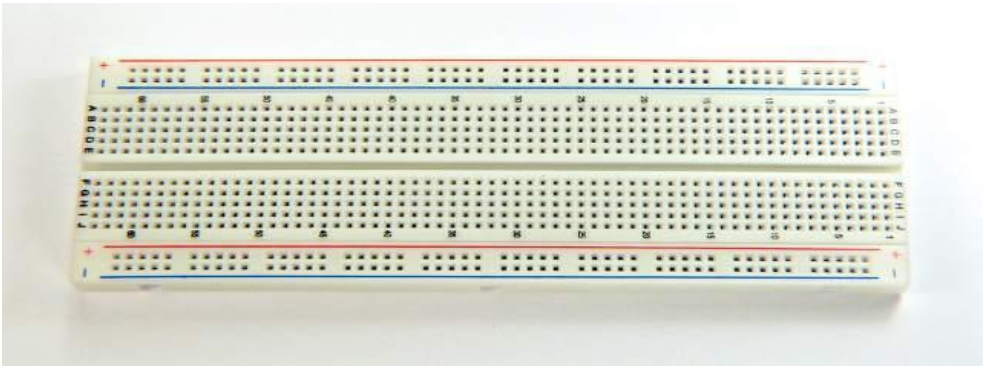


Fig 2.9: Layout of breadboard

Note how all holes in the selected row are connected together, so the holes in the selected column. The set of connected holes can be called a node:

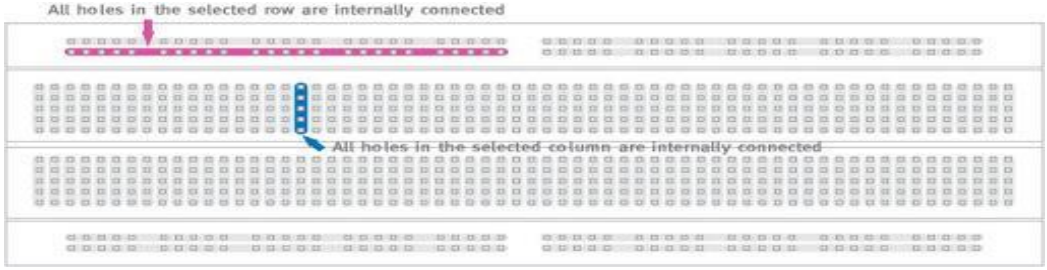


Fig 2.10: Internal structure of breadboard

To interconnect the selected row (node A) and column (node B) a cable going from any hole in the row to any hole in the column is needed:

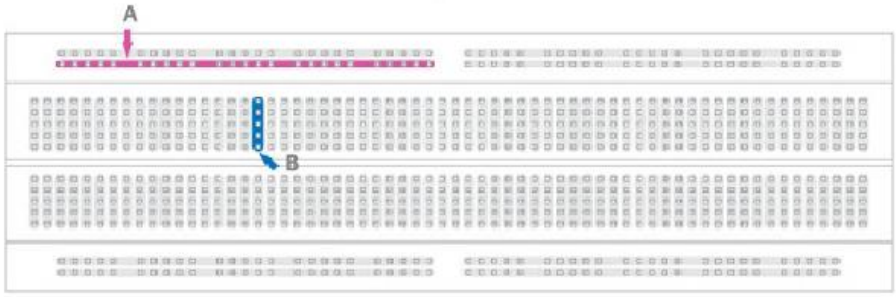


Fig 2.10 (a): serial connection

Now the selected column (node B) and row (node A) are interconnected.

Fig 2.10 (b): Parallel connections

2.6 RAIN SENSOR

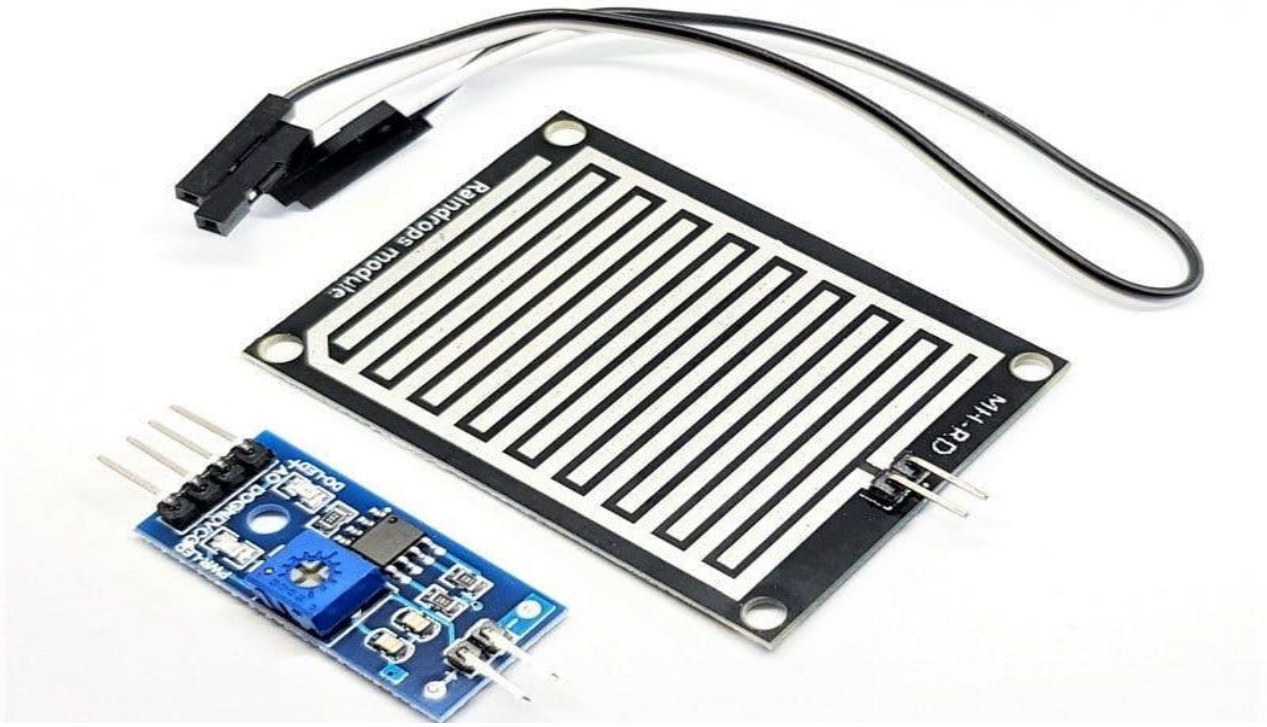


Fig 2.6 Rain Sensor

The sensing pad with series of exposed copper traces, together acts as a variable resistor (just like a potentiometer) whose resistance varies according to the amount of water on its surface.

This resistance is inversely proportional to the amount of water:

The more water on the surface means better conductivity and will result in a lower resistance.

The less water on the surface means poor conductivity and will result in a higher resistance.

The sensor produces an output voltage according to the resistance, which by measuring we can determine whether it's raining or not.

2.6.1 RAIN SENSING PAD

The sensor contains a sensing pad with series of exposed copper traces that is placed out in the open, possibly over the roof or where it can be affected by rainfall.

Usually these traces are not connected but are bridged by water.

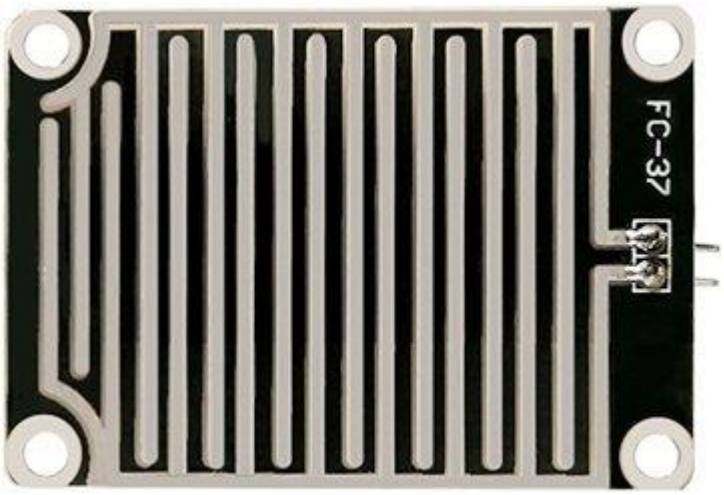


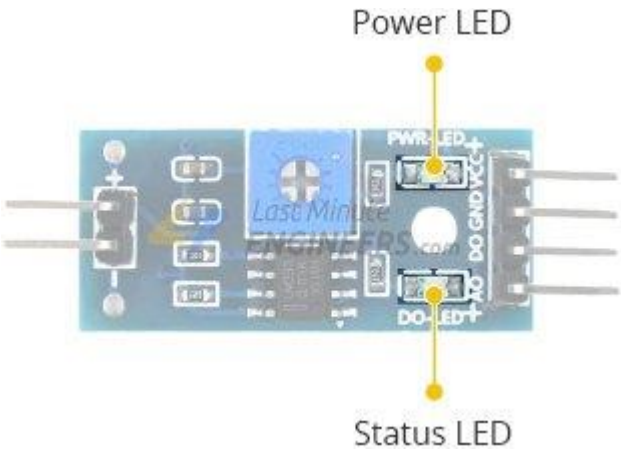
Fig 2.6.1 Rain Sensor Pad

2.6.2 The MODULE

The sensor also contains an electronic module that connects the sensing pad to the Arduino.The module produces an output voltage according to the resistance of the sensing pad and is made available at an Analog Output (AO) pin.The same signal is fed to a LM393 High Precision Comparator to digitize it and is made available at an Digital Output (DO) pin.

The module has a built-in potentiometer for sensitivity adjustment of the digital output (DO).

You can set a threshold by using a potentiometer; So that when the amount of water exceeds the threshold value, the module will output LOW otherwise HIGH.



Apart from this, the module has two LEDs. The Power LED will light up when the module is powered. The Status LED will light up when the digital output goes LOW.

2.6.3 RAIN SENSOR PINOUT

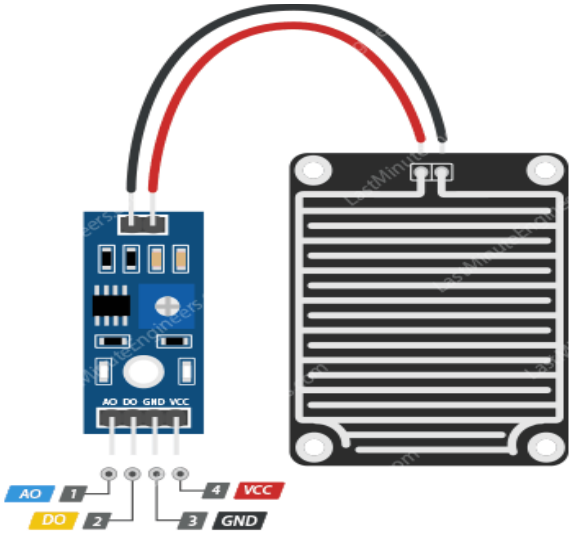


FIG 2.6.3

AO (Analog Output) pin gives us an analog signal between the supply value (5V) to 0V.

DO (Digital Output) pin gives Digital output of internal comparator circuit. You can connect it to any digital pin on an Arduino or directly to a 5V relay or similar device.

GND is a ground connection.

VCC pin supplies power for the sensor. It is recommended to power the sensor with between 3.3V – 5V. Please note that the analog output will vary depending on what voltage is provided for the sensor.

CHAPTER 3

SOFTWARE DESCRIPTION

Step 1 − First you must have your Arduino board (you can choose your favourite board) and a USB cable. In case you use Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, you will need a standard USB cable.

Fig 3.0: A to B connector

Step 2 − Download Arduino IDE Software.

You can get different versions of Arduino IDE from the Download page on the Arduino Official website. You must select your software, which is compatible with your operating system (Windows, IOS, or Linux). After your file download is complete, unzip the file.

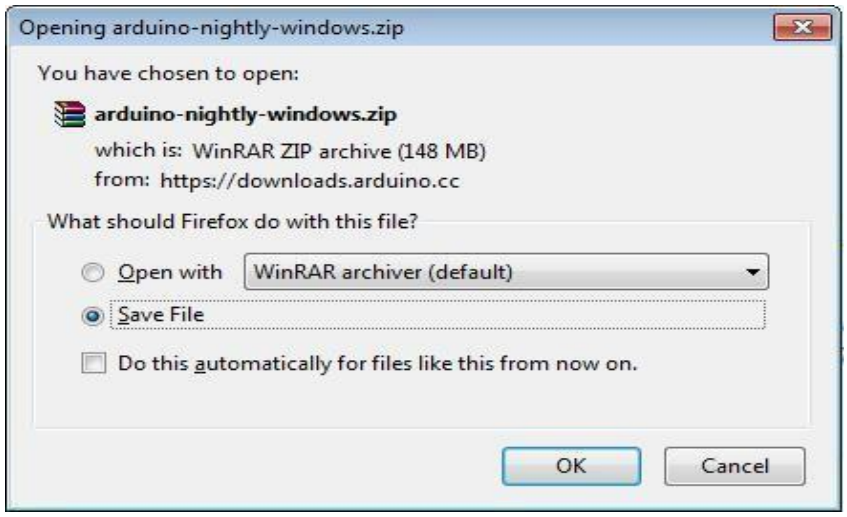


Fig 3.1: Downloading Arduino IDE

Step 3 − Power up your board.

The Arduino Uno, Mega, Duemilanove and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply.

If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks. Check that it is on the two pins closest to the USB port. Connect the Arduino board to your computer using the USB cable. The green power LED (labelled PWR) should glow.

Step 4 − Launch Arduino IDE.

After your Arduino IDE software is downloaded, you need to unzip the folder. Inside the folder, you can find the application icon with an infinity label (application.exe). Double click the icon to start the IDE.

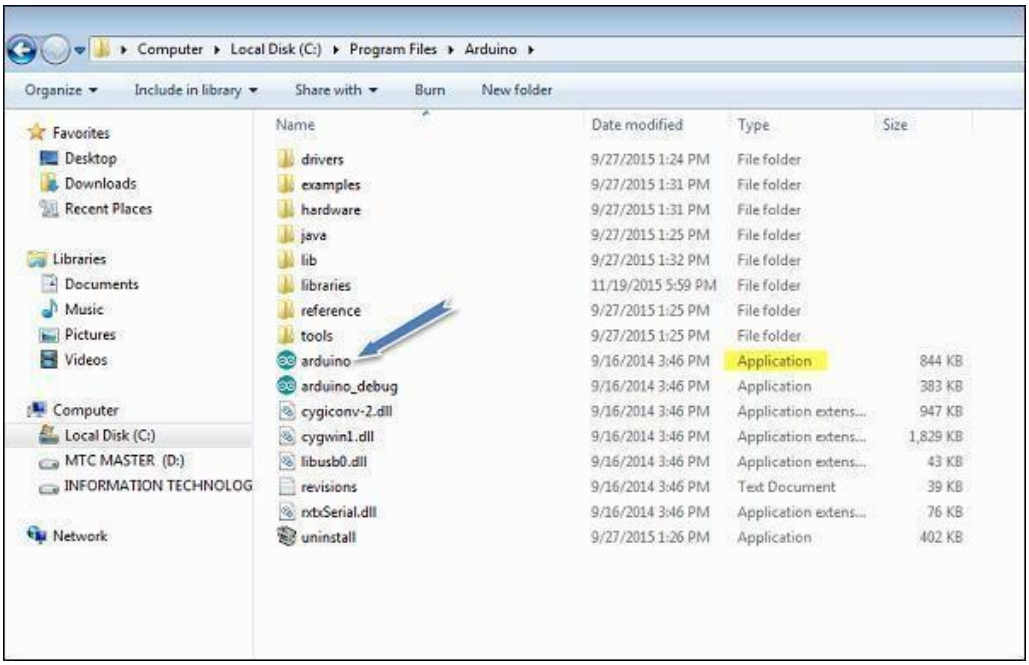


Fig 3.2: launch Arduino IDE

Step 5 - Open your first project. Once the software starts, you have two options.

- Create a new project.
- Open an existing project example.

To create a new project, select File → New.

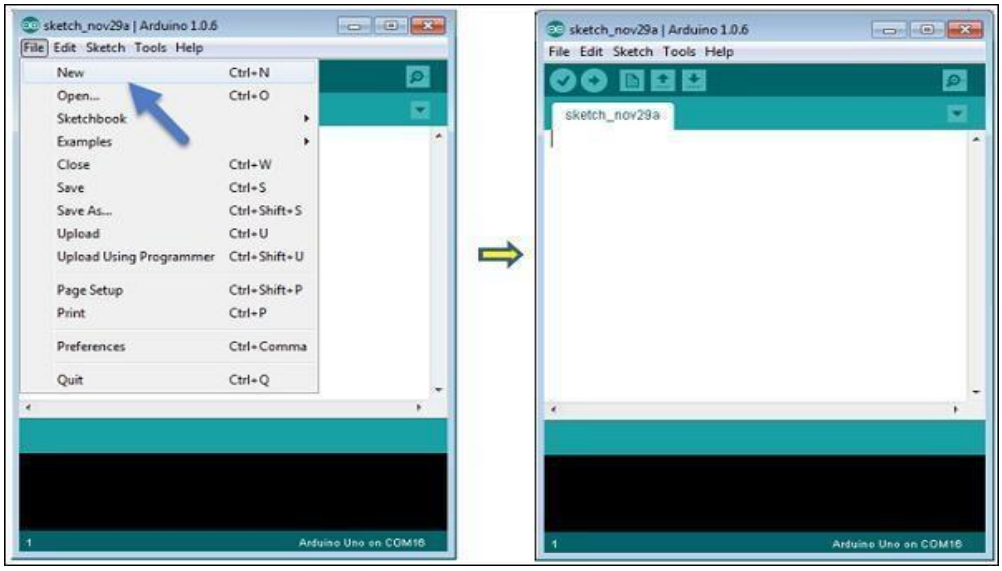


Fig 3.3: to create a new project

To open an existing project example, select File → Example → Basics → Blink.

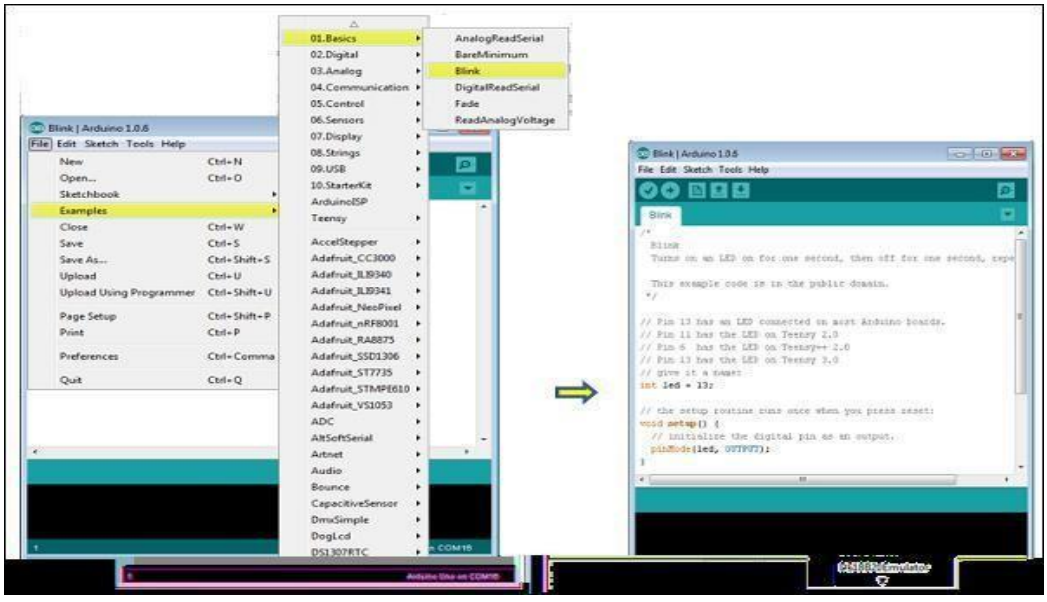


Fig 3.3(a): Opening project

Here, we are selecting just one of the examples with the name Blink. It turns the LED on and off with some time delay. You can select any other example from the list.

Step 6 − Select your Arduino board

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer. Go to Tools → Board and select your board.

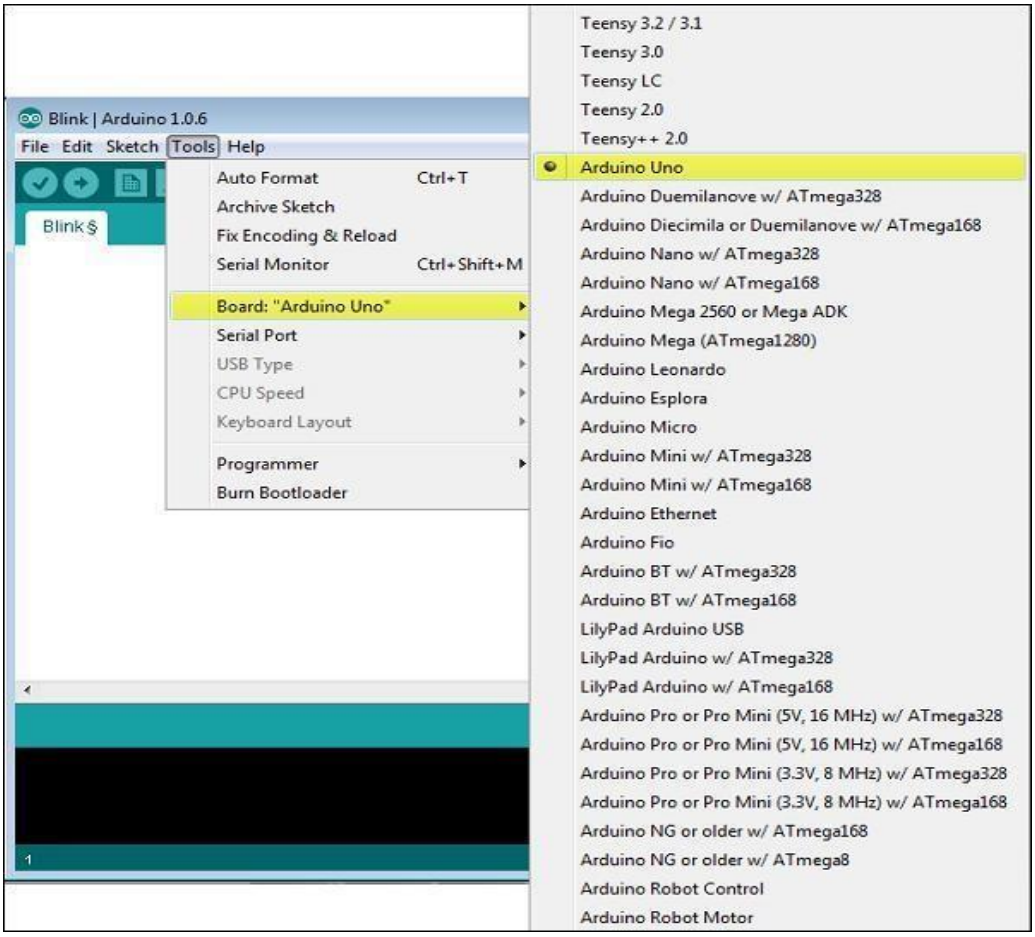


Fig 3.4: Selecting Arduino board

Here, we have selected Arduino Uno board according to our tutorial, but you must select the name matching the board that you are using.

Step 7 − Select your serial port.

Select the serial device of the Arduino board. Go to Tools → Serial Port menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports).

To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.

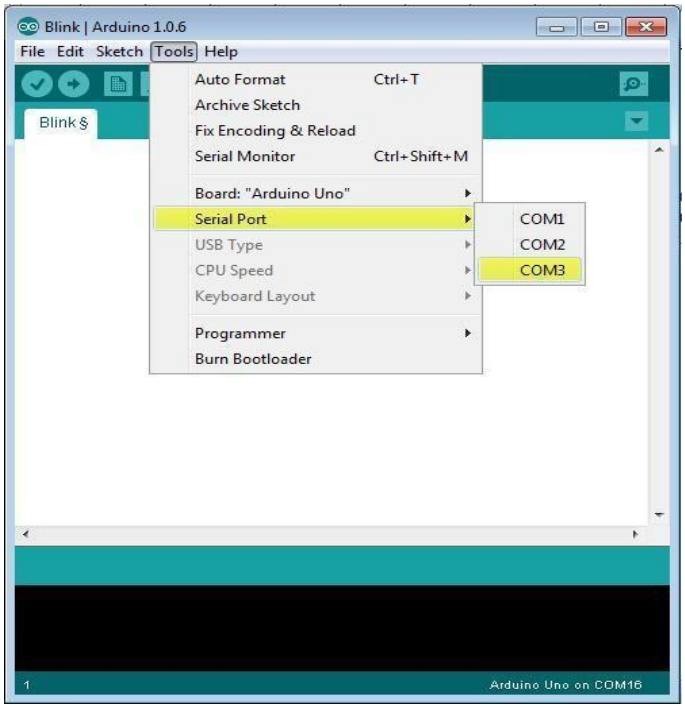


Fig 3.5: serial port selection.

Step 8 −Upload the program to your board.

Before explaining how we can upload our program to the board, we must demonstrate thefunction of each symbol appearing in the Arduino IDE toolbar.

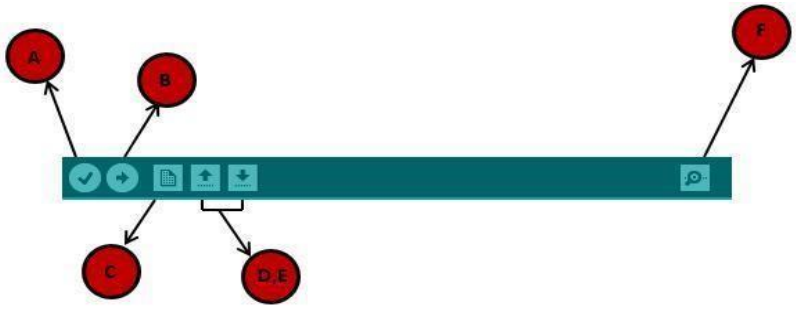


Fig 3.6: Uploading Program.

A − Used to check if there is any compilation error.

B - Used to upload a program to the Arduino board.

C− Shortcut used to create a new sketch.

D− Used to directly open one of the example sketches.

E− Used to save your sketch.

F− Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.

Note − If you have an Arduino Mini, NG, or other board, you need to press the reset button physically on the board, immediately before clicking the upload button on the Arduino Software.

CHAPTER 4

WORKING AND EXPLANATION

4.1 BLOCK DIAGRAM

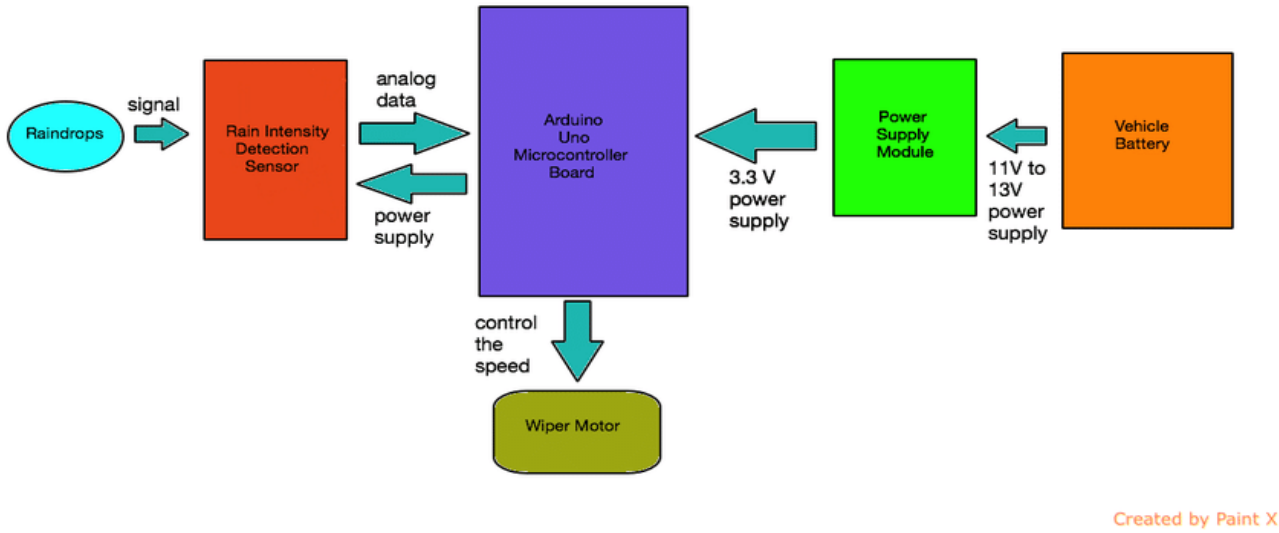


Fig 4.1 Block Diagram

EXPLAINATION

The main block diagram consists of rain sensor, servo motor, Arduino UNO, LCD and power supply.

The Arduino is the main microcontroller in this system and all other module/hardware is connected to it directly. The rain sensor senses the water value on the glass and sends it to Arduino.

Arduino will process the data and check the water value. The compared value is used to drive servo motor and LCD module. An algorithm designed for this generates variable pulse width for different instant value of linear voltage from the water sensor.

The flow chart as shown in the figure represents complete design of proposed automatic wiper system without human interrupt.

4.2 CIRCUIT DIAGRAM

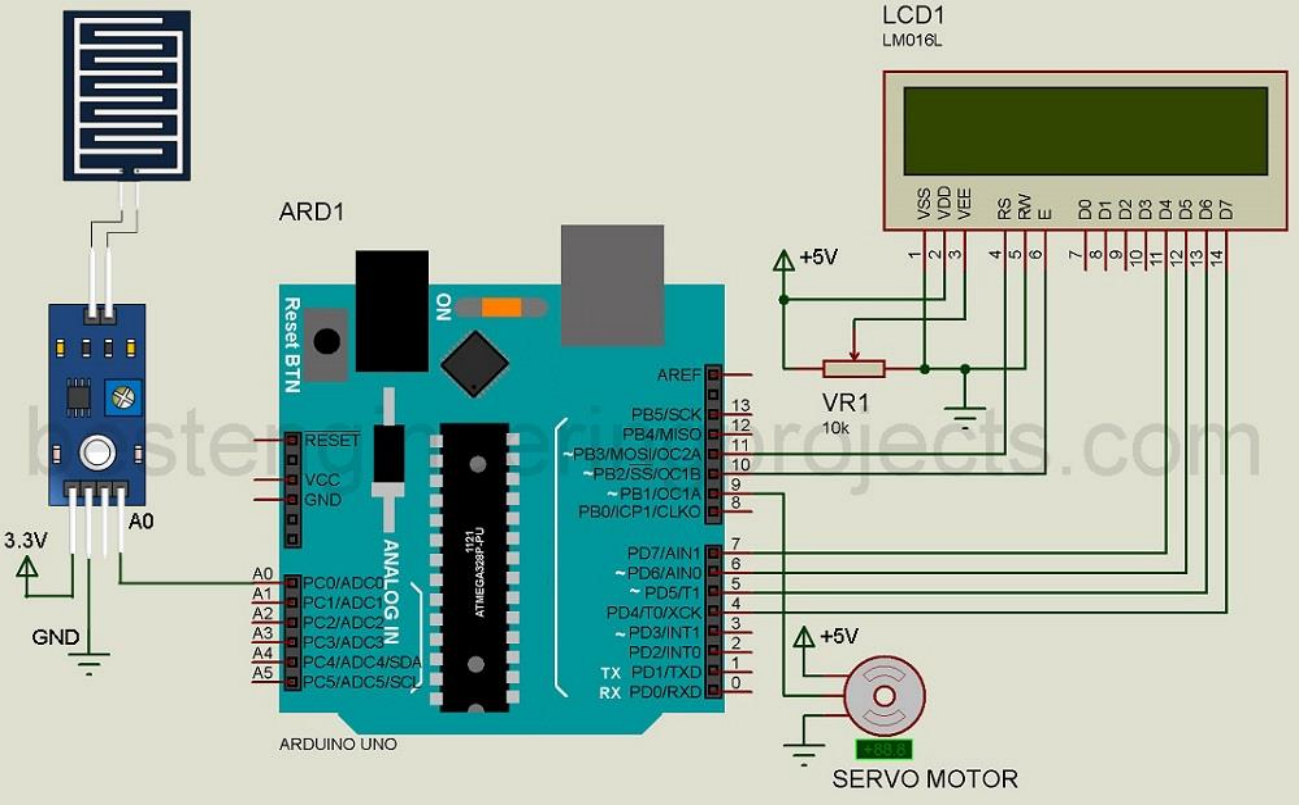


Fig 4.2 circuit diagram of automatic rain sensing car wiper using Arduino

CIRCUIT EXPLAINATION:

The rain intensity module which we are using for this particular project is MH-RD rain detector. This module comprises 4 pins: Vcc, A0, D0 and Gnd. And, two of the pins; Vcc and Gnd of this module are interfaced with the supply pins of the arduino module. Out of two remaining pins from the module, A0 supplies analog output where as D0 supplies digital output. But, here in our case, we need to monitor the intensity variation of the rainfall and thus we ignore the digital output pin D0 and connect the analog output from module; A0 to one of the analog input pins of the arduino uno board.

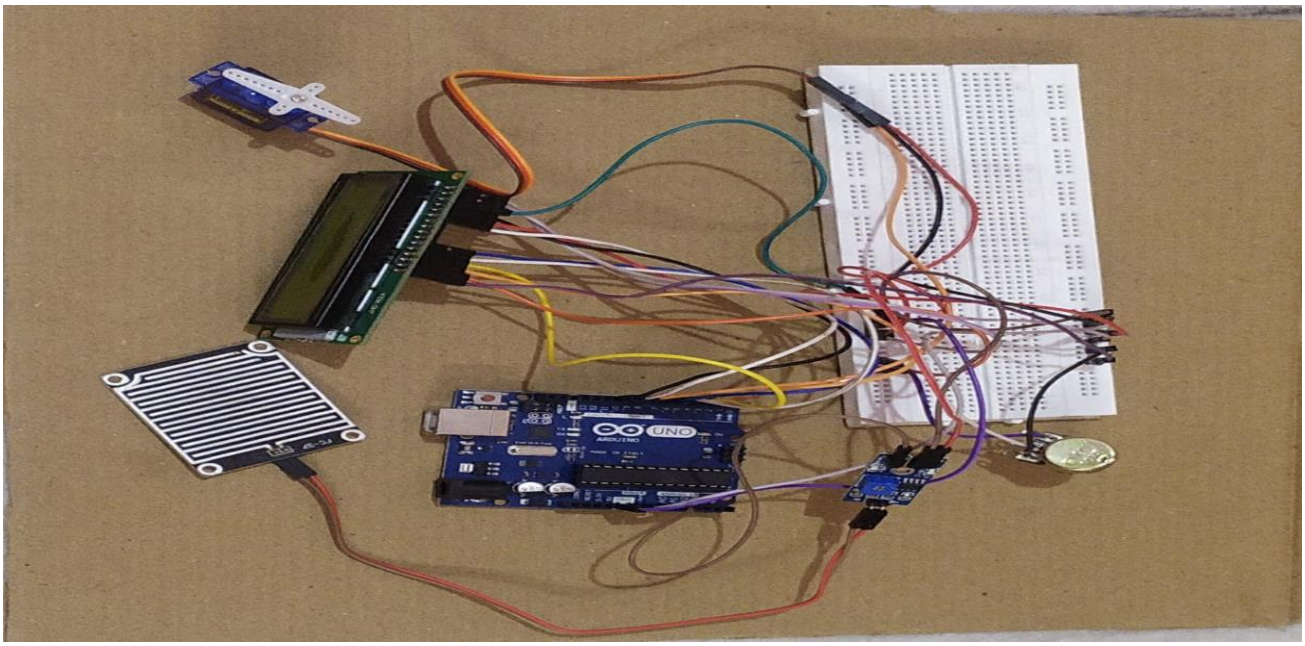
For wiper movements control, a servo motor is installed. The servo motor produces suitable PWM output signal which is responsible to adjust its shaft to a desirable position. The output signal from servo motor is supplied as input to digital pin 9 which is one of the PWM pins of Arduino UNO. A 16*2 JHD162A LCD does the task of displaying the intensity of rainfall on its screen. This LCD is based on the HD44780 driver from Hitachi. The Arduino pins 13, GND and 10 are matched with

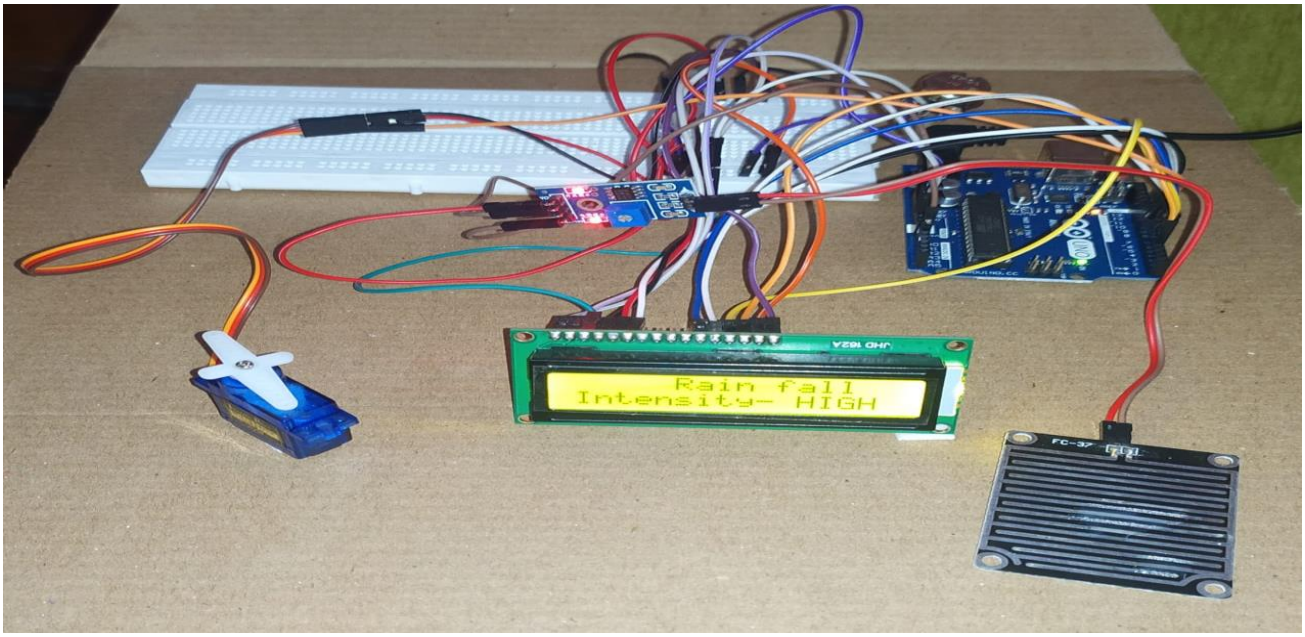
control pins of the LCD; RS, RW and En respectively. Similarly, Arduino pins 7,6,5 and 4 are linked with data pin D4-D7 of the LCD.

Initially, as the project is activated, the wiper is switched back to zero degree position by the servo motor. The controller keeps on monitoring the signal received from the rain detector module. When the rain detector module senses rainfall, the controller checks if it crosses a threshold value set in the device. As soon as that happens, the controller triggers the servo motor which than starts to work. The strength of the signal decides the speed of the servo motor. The rainfall intensity is continuously displayed on the LCD screen.

CHAPTER- 5

RESULT





CHAPTER 6

ADVANTAGES AND APPLICATIONS

Advantages

- It is easy to design and manufacture as all the components are easily available.
- It consumes less power.
- It reduces the accidents caused by “Toxler Effect”.
- Its operating principle is very easy.
- Component’s cost is low.
- Hand free calling/
- Automotive safety and convenience.
- Free from wear adjustment.

Applications

- Used in 4 wheel vehicles.

CONCLUSION

CONCLUSION:

The automatic car wiper system was developed to sense the rain and wipe the glass by moving the windshield wipers. By using automatic car wiper system, the purpose of driver's response to control the wiper is automated. It is demonstrated and proved that the rain sensors response to the rain for moving the windshield wipers is less than 400 milliseconds. Though the automatic car wiper is designed using rain sensor and Arduino, it can be advanced by replacing the rain sensors with IR sensors for accurately determining and detecting the rainfall. If you opt for using a wiper which is not only economical but also efficient, using a rain sensor is the best. To advance the movement and to change the system different sensors which are useful for this purpose can be selected.

APPENDIX

SOFTWARE REQUIREMENTS

```
#include &lt;LiquidCrystal.h&gt;

#include &lt;Servo.h&gt;


LiquidCrystal lcd(11,10,7,6,5,4);

Servo myservo;


int InPosition = 0;

int SensorValue = 0;


void setup() {
```

```
Serial.begin(9600);
lcd.begin(16,2);
myservo.attach(9);
lcd.clear();
lcd.setCursor(0,0);
lcd.print(&quot;Automatic Auto.&quot;);
lcd.setCursor(0,1);
lcd.print(&quot;Wiper System&quot;);
delay(3000);
lcd.clear();
lcd.setCursor(5,0);
lcd.print(&quot;Rain fall&quot;);
lcd.setCursor(0,1);
lcd.print(&quot;Intensity-&quot;);
}

void wipe(int Speed)
{

if(Speed&gt;400){Speed=5000;lcd.print(&quot; LOW &quot;);}
if(Speed&gt;350&amp;&amp;Speed&lt;=400){Speed=3000;lcd.print(&quot;MEDIUM&quot;);}
if(Speed&lt;=350){Speed=500;lcd.print(&quot; HIGH &quot;);}
for (InPosition = 180; InPosition &gt;= 0; InPosition--) {
myservo.write(InPosition);
```

```
delay(3);
}
for (InPosition = 0; InPosition &lt;= 180; InPosition++) {
myservo.write(InPosition);
delay(3);
}
delay(Speed);
}


void loop()
{
lcd.setCursor(10,1);
SensorValue = analogRead(A0);
Serial.println(SensorValue);
if(SensorValue&gt;600){myservo.write(180);lcd.print(&quot; NIL &quot;);delay(1000);}
if(SensorValue&lt;=600){lcd.setCursor(10,1);wipe(SensorValue);}
}
```

REFERENCES

1. Automatic Rain Operated Wiper System in Automobile: A Review by Kothari Mohit, Shah Amit, Patel Vipul and Kadakia Nishant in International Journal for Scientific Research & Development| Vol. 3, Issue 02, 2015| ISSN:2321-0613.
2. Automatic control of vehicles: An Analysis by Kushal Sarin, Jatin Sethi, Anshuman Gupta, Usha Tiwari and Ishan Mathur in International Journal of Advanced Research in Computer and Communication Engineering April 2013.
3. Review report on soc on various platforms for vehicles by Sheeja S. Suresh and Anuradha S. Joshi1 in International Research Journal of Engineering and Technology.
4. Rainy Weather Recognition from in-vehicle Camera Images for Driver Assistance by I. Ide, Y. Mekada, H. Murase, Y. Tamatsu, H. Kurihara, T. Takahashi, and T. Miyahara, in IEEE Intelligent Vehicles Symposium, 2005.
5. KadakiaNishant, A Kothari, Mohit A Shah, Amit V Patel Vipul R: Automatic Rain Operated Wiper System in Automobile, International Journal for Scientific Research & Development Vol. 3, Issue 02, 2015.
6. AHM FazleElah and Mohammad ShafiurRehman in Intelligent Windshield for Automotive Vehicles 17th International Conference on Computer and Info. Technology 22-23 December 2014. International University, Dhaka Bangladesh.
7. H. Kurihara, T. Takahashi, I. Ide, Y. Mekada, H. Murase, Y. Tamatsu, and T. Miyahara, “Rainy Weather Recognition from invehicle Camera Images for Driver Assistance,” In IEEE Intelligent Vehicles Symposium, 2005, pp. 205-210.
8. Anuradha S. Joshi1, Sheeja S. Suresh, "review report on soc on various platforms for vehicles", International Research Journal of Engineering and Technology (IRJET).
9. Tapan S. Kulkarni, Harsh S. Holalad, July 2012, “Semi-Automatic Rain Wiper System,” International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, Volume 2, Issue 7.
10. N. M. Z. Hashim, July 2013. “Smart Wiper Control System,” International Journal of Application or Innovation in Engineering & Management (IJAIEM), ISSN 2319 – 4847, Volume 2, Issue 7.
11. K. V. Viswanadh, January-2015, “Design & Fabrication of Rain Operated Wiper Mechanism using Conductive Sensor Circuit,” International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 4, Issue 01.