MINI PROJECT-1 (22ET5PWMP1)

A Project Report

ON

AGROBOT

Submitted in partial fulfilment of the requirements for the award of degree of

BACHELOR OF ENGINEERING IN

ELECTRONICS AND TELECOMMUNICATION ENGINEERING

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI

SUBMITTED BY:

NAIVEDYA SHUKLA	**1BM22ET037**
BHARAT SAI B	**1BM21ET010**

Under the guidance of

DR. M VASANTHA LAXMI

Department of Electronics and Telecommunication Engineering
(Autonomous Institution under VTU, Belagavi)

BMS COLLEGE OF ENGINEERING, BENGALURU-560019 (Autonomous Institution under VTU)

Department of Electronics and Telecommunication

BONAFIDE CERTIFICATE

This is to certify that the project titled “***AGROBOT***” is a bonafide work carried out by **NAIVEDYA SHUKLA (1BM22ET037)**, in partial fulfilment for the completion of **MINI PROJECT-1 (22ET5PWMP1)** during the academic year 2024-2025.

DR. M VASANTHA LAXMI

Dr. Balachandra K
(HOD)

External Examination

Name of the Examiners

Signature with date

ACKNOWLEDGEMENT

We take this opportunity to thank our friends for helping us through the process of completing this project.

We also thank the kind strangers on the internet who make such projects possible by lending their hands to address questions and are ready to share their knowledge with those who seek it without anything in return.

We finally thank God for helping us come this far despite all the hardships and the pressures we faced along our journey; and ourselves for our perseverance throughout the stress and pressure we have been subjected to these past few years.

And we thank anybody who has actually taken their time to read through the entirety of this report we have drafted, despite its shortcomings.

TITLE: AGROBOT

Naivedya Shukla (1BM22ET037) and Bharat Sai B (1BM21ET010), V Semester students from Department of Electronics and Telecommunication Engineering, BMSCE, Bangalore-19

DATE:

Objective:

- To design and develop a cost-effective, user-friendly seed sowing robot that can be used for small-scale farming, backyards, and terrace farming.
- To automate the seed sowing process, reducing the need for manual labor and increasing efficiency.
- To incorporate obstacle detection capabilities, allowing the robot to navigate around obstacles in its path.
- To integrate a soil moisture sensor to determine the moisture content of the soil before sowing.
- To control the robot wirelessly using a Bluetooth module and a user-friendly smartphone application.

Description:

The Agro Bot is a compact and affordable seed sowing robot designed to assist with small-scale farming. It addresses the challenges of seed sowing in limited spaces like backyards and terraces, where traditional methods or large machinery are impractical. The robot is equipped with a seed sowing mechanism, an obstacle detection sensor, a soil moisture sensor, and a Bluetooth module for wireless control.

The seed sowing mechanism is designed to sow seeds at a consistent depth and spacing, ensuring optimal planting conditions. The obstacle detection sensor, typically an ultrasonic sensor, allows the robot to identify and avoid obstacles in its path, preventing collisions and ensuring continuous operation. The soil moisture sensor measures the moisture level of the soil, helping to determine if conditions are suitable for planting. The Bluetooth module enables wireless control of the robot through a smartphone application, providing a user-friendly interface for operating the robot and monitoring its functions.

ABSTRACT

Agriculture plays an important role in economic status in India. Agro - Technology is the process of applying technological innovation occurring in daily life and applying that to the agriculture sector which improves the efficiency of the crop produced. It also

develop a better mechanical machine to help the agriculture field which reduces the amount and time of work spent on the crop. Automation in agricultural processes provides the benefits of increased efficiency, productivity, healthy crops, efficient use of resources, material and labor cost savings. Compared to traditional methods this process has various advantages in sowing the seeds.

The seed sowing robot is regulated and hand-operated that slashes exercise of farmers and increases the capability of seed sowing compared to normal planting which was performed by farm workers. Seed sowing robot is Bluetooth controlled. Sensors are used for obstacle detection and robots change their path accordingly. The soil moisture sensor helps in determining the moisture content of the soil thus helps us in deciding whether to grow the plant in that particular soil. The proposed system is a boon which combines robotics with agriculture and capable of moving around the field and sowing the seeds in the pre- determined row along the rows in an automated way. It increases the planting efficiency. Accuracy rate will be high compared to traditional sowing process. Seed sowing robots have been built earlier but the robot which we have built can be used for small agricultural lands, backyards and large terrace farming. It is designed using low-cost equipment so it can be easily available for small scale farmers thus making it user friendly. The robot performs different operations which provides a better solution for low productivity nations.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Automation in agriculture is still in the developing stage due to lack of technical knowledge, advanced technology, and machinery. A few years ago, agriculture was carried away by conventional methods such as bullock carts. Later machinery came into existence such as tractors, tillers etc. But the problem with the tractor is it causes a lot of pollution and consumes large power.

As a few years passed by, and technology started progressing robotics came into existence helping the farmers to reduce their physical labour and helped in providing planned farming. Hence using modern technology in farming activity provides greater support to the common people.

Automation plays a significant role in enhancing agriculture production needs. Once automation and agriculture are accepted, the adoption rates will become high and technology costs will drop. Autonomous agriculture allows farmers to reduce the environmental impact, increase precision in an effective manner. The idea of applying robotics technology in agriculture is very new. In agriculture, the opportunities for robot-enhanced productivity are immense and robots are appearing on farms in various guises and in increasing numbers. In Modern world, Automation robot is used in many of the fields such as defence, surveillance, medical field, industries and so on.

Manual technique incorporates broadcasting the seeds by hand. Sometimes method of digging i.e., making gaps and dropping seeds is done by labours. Likewise, a couple of bullocks are utilized to convey the substantial hardware of leveling and seed dropping. Seed plantation in day-to-day life is done by tractor in farms. The conventional method for seeding is the manual one. It requires more time and the problem with the workforce is seen. In the present scenario domestic farming is the most frequent problem people are facing. Compared to large agricultural fields, we can use huge tractors and big robots for farming, but for domestic purposes like farming in small grasslands, backyards and large terrace farming big machinery cannot be used. So, we require small robots that make the job easier. Driverless robots are designed to replace human labour. Developing advanced technology in farming is needed, especially seed sowing.

Bluetooth module can be used which helps in directing the robot. This will reduce human effort and time. Innovative technologies are needed to push out yield frontiers, utilize inputs more efficiently and diversify to more sustainable and higher value cropping patterns.

The energy needed for robotic machines is less as compared with other machines like tractors or any agriculture tools. In seed sowing machine system battery powered wheels are used. The plantations of seeds are automatically done by using servo motor. The robot works based on command given by the controller. Various sensors are used for sensing various parameters along the robotic path. The microcontroller, being the heart of the robotic system, manipulates the entire action of the robotic system. It also controls wheel motion by controlling the DC motors. Motor driving circuit drives the DC motor, which controls the wheel motion. Controlling of the robot mainly requires some means of communication. One of the means of communication is the wireless Bluetooth connectivity.

HC-05 Bluetooth module is used to control the robot using Smartphone. The Bluetooth application is user friendly and data exchanging between robots and smartphones is done systematically. The project consists of a moisture sensor which gives information by displaying the values about the moisture level of the soil. The ultrasonic sensor is used to detect the obstacle

s if present. The Agro-bot developed performs seed sowing process powered by external power supply and operated with the help of android application. Future growth needs to be more rapid, more widely distributed and better targeted.

1.1 PROBLEM DEFINITION

The problem which is occurred at present scenario is domestic farming which is important. For small areas such as backyards, small grasslands or even large terrace farming ,we can neither use conventional method of farming nor we can use skilled farmers.

The main problem of farming is seed sowing .Proper seed sowing is very necessary on which the whole process of plant growth is dependent. Our robot is a obstacle detecting, seed sowing robot which can detect if any obstacle present and if no obstacle is present it will move further thus by sowing the seed.

Thus the robot which we have built at present can be used for domestic purpose but it can be modified in such a way that it can be used in agricultural fields also. Thus we have a real time solution for a real time problem.

FIGURE: 1.1

CHAPTER 2

LITERATURE SURVEY

It is capable of digging the soil , seeds from the hopper is dropped into the field, water is pumped from the tank using Arduino Mega[1]. The advantages are It is operating all control systems like ploughing, seed sowing and water sprinkling through an automated seed sowing robot The disadvantages are Leakage in the water tank causes short circuit. Use of rechargeable battery consumes more energy.

The Bluetooth HC-05 module is fixed on to the robot which receives signals from the Bluetooth electronics app and sends these signals to the microcontroller for processing of operations[2]. The advantages are Temperature and humidity sensors measure temperature and humidity respectively.Soil moisture sensor is used to measure moisture content of the soil.The disadvantages are Since a LCD display is used ,the values has to be checked everytime by the robot itself.

Two pairs of heterogonous sensors were employed to detect obstacles along the path of the mobile Robot using Arduino Uno[3]. The advantages are The evaluation on the autonomous system shows that it is capable of avoiding obstacles, ability to avoid collision and change its position. The disadvantages are it is slightly difficult to control without bluetooth or wifi.

When Object is near the sensor, it measures the distance and gives output When object near the sensor the LEDs are ON and when object is away from the sensor then LEDs are OFF[4]. The advantages are It can be used as a movable Surveillance System. It can be controlled remotely. The disadvantages are it is use for short distance only. The range of the bluetooth module is less

In this project corn cultivation is done. Many different processes in the system likes seed sowing, leveling and water spraying is done using Arduino Mega[5]. The advantages Increases the speed of sowing process and accuracy of seed placing .It can measure the space between the seeds. The disadvantages It is being restricted to only one kind of seed electrical energy is being consumed due to the batteries which has been used .

CHAPTER 3

HARDWARE AND SOFTWARE COMPONENTS

3.1 HARDWARE REQUIREMENTS: -

Hardware components are physical equipment which is used to achieve a specific objective.

3.1.1 ARDUNIO UNO: -

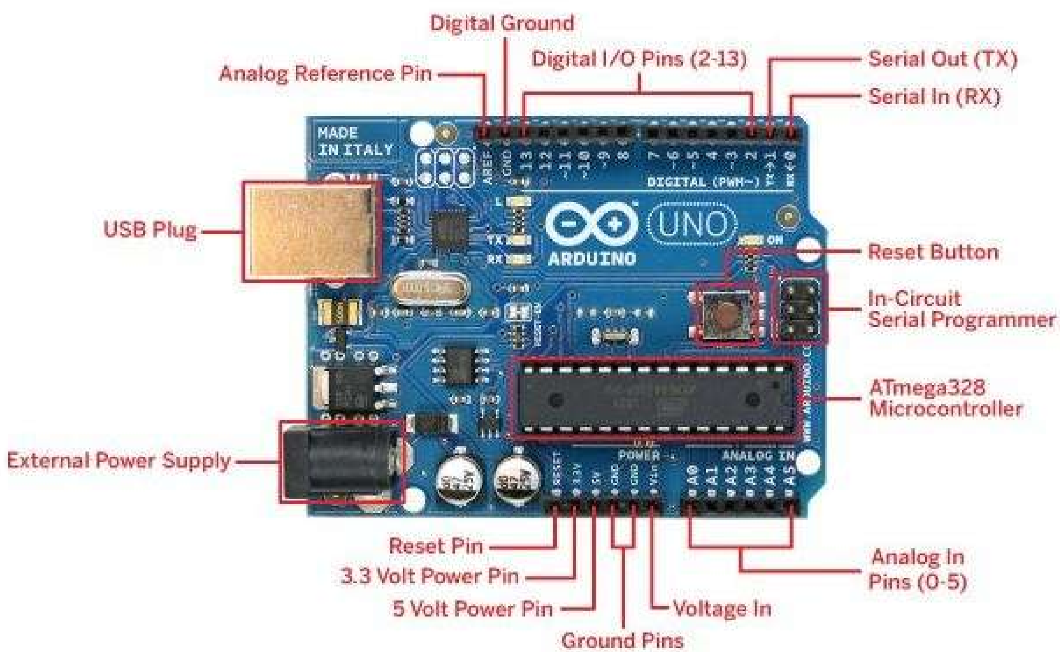


FIGURE:3.1.1

The Arduino Uno is open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and Analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits.[1] The board has 14 digital I/O pins (six capable of PWM output), 6 Analog I/O pins, and is programmable with the Arduino

IDE (Integrated Development Environment), via a type B USB cable.[4] It can be powered by the USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. It is like the Arduino Nano and Leonardo.[5][6] The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available.

The word "uno" means "one" in Italian and was chosen to mark the initial release of Arduino Software. The Uno board is the first in a series of USB-based Arduino boards;[3] it and version 1.0 of the Arduino IDE were the reference versions of Arduino, which have now evolved to newer releases.[4] The ATmega328 on the board comes pre-programmed with a bootloader that allows uploading new code to it without the use of an external hardware programmer.[3]

The Arduino project started at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy. At that time, the students used a BASIC Stamp microcontroller, at a cost that was a considerable expense for many students. In 2003, Hernando Barragan created the development platform Wiring as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas, who are known for work on the Processing language. The project goal was to create simple, low-cost tools for creating digital projects by non-engineers.

The Wiring platform consisted of a printed circuit board (PCB) with an ATmega168 microcontrollers, an IDE based on Processing, and library functions to easily program the microcontroller.[8] In 2003, Massimo Banzi, with David Mellis, another IDII student, and David, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they forked the project and renamed it *Arduino*. Early Arduino boards used the FTDI USB-to-serial driver chip and a ATmega168.[8] The Uno differed from all preceding boards by featuring the ATmega328P microcontroller and an ATmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

SPECIFICATIONS:

Microcontroller: Microchip ATmega328P

- Operating Voltage: 5 Volts
- Input Voltage: 7 to 20 Volts
- Digital I/O Pins: 14 (of which 6 can provide PWM output)
- UART: 1
- I2C: 1
- SPI: 1
- Analog Input Pins: 6
- DC Current per I/O Pin: 20 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB used by bootloader
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz
- Length: 68.6 mm
- Width: 53.4 mm
- Weight: 25 g

PIN FUNCTIONS: -

GENERAL:

- **LED**: There is a built-in LED driven by digital pin 13. When the pin is high value, the LED is on, when the pin is low, it is off.
- **VIN**: The input voltage to the Arduino board when it is using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

- **5V**: This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 20V), the USB connector (5V), or the VIN pin of the board (7-20V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator and can damage the board.
- **3V3**: A 3.3volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND**: Ground pins.
- **IOREF**: This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.
- **Reset**: Typically used to add a reset button to shields that block the one on the board.[7]

Each of the 14 digital pins and 6 Analog pins on the Uno can be used as an input or output, under software control. They operate at 5 volts. Each pin can provide or receive 20 mA as the recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50K ohm. A maximum of 40mA must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller. The Uno has 6 analog inputs, labelled A0 through A5; each provides 10 bits of resolution (i.e., 1024 different values).

SPECIAL PIN FUNCTIONS:

- **Serial** / UART: pins 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL serial chip.
- **External interrupts**: pins 2 and 3. These pins can be configured to trigger an interruption on a low value, a rising or falling edge, or a change in value.
- PWM (pulse-width modulation): pins 3, 5, 6, 9, 10, and 11. Can provide 8-bit PWM output with the Analog Write () function.

- SPI (Serial Peripheral Interface): pins 10 (SS), 11 (MOSI), 12 (MISO), and 13 (SCK). These pins support SPI communication using the SPI library.
- **TWI** (two-wire interface) / I²C: pin SDA (A4) and pin SCL (A5). Support TWI communication using the Wire library.
- **AREF** (Analog reference): Reference voltage for the Analog input.
- The Arduino uno has facilities for communicating with a computer, another Arduino board, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware uses standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software Serial library allows serial communication on any of the Uno's digital pins.
-

3.1.2 MOTOR DRIVER SHIELD:

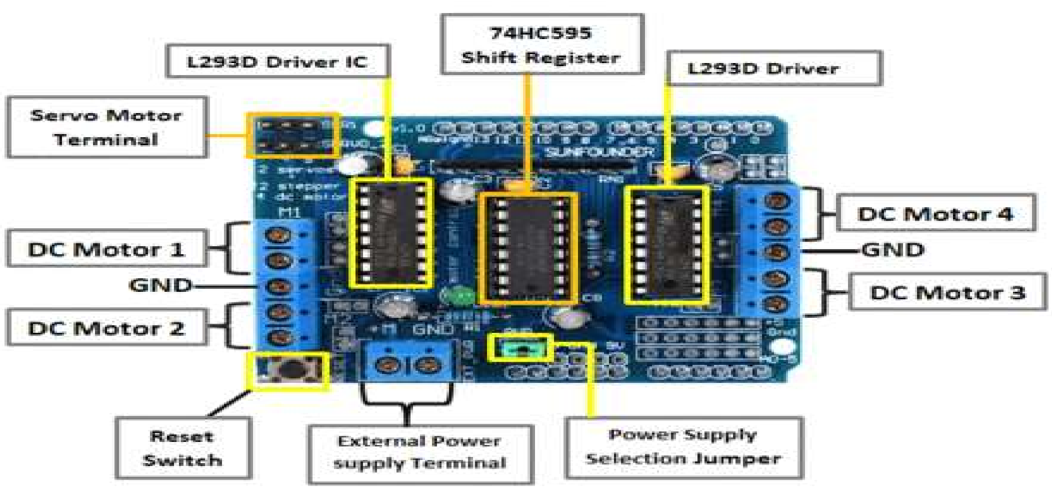


FIGURE:3.1.2.1

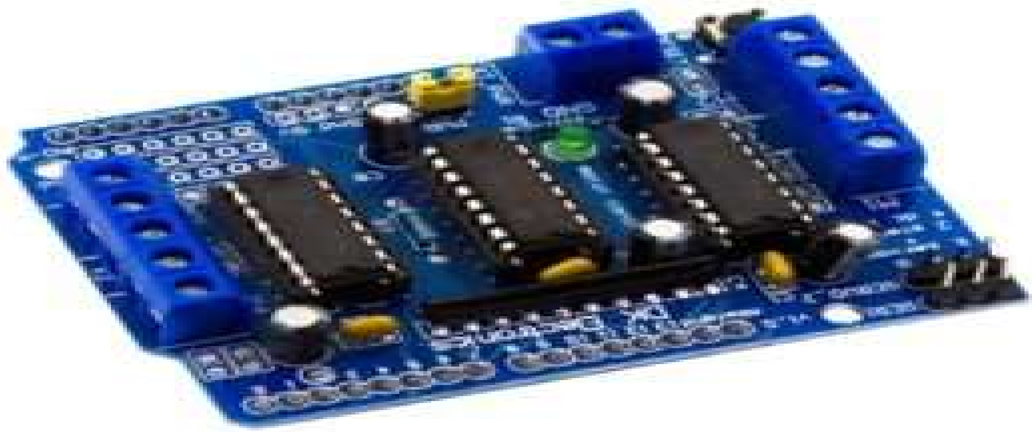


FIGURE:3.1.2.2

The Arduino Motor Shield is based on the L293D, which is a Half-bridge driver designed to drive inductive loads such as relays, solenoids, DC and stepping motors. It lets you drive two DC motors with your Arduino board, controlling the speed and direction of each one independently. You can also measure the motor current absorption of each motor, among other features. The shield is also compatible with DTMF module, which means you can quickly create projects by plugging DTMF modules to the board.

However, these motors typically cannot be driven directly by Arduino or another microcontroller. This is because of their higher current and power ratings, so motor shields or driver ICs are used instead. These shields or ICs isolate a motor's power supply and use control logic from the microcontroller circuitry.

One of the most popular motor driver shields used with Arduino is the L293D. The full-featured L293D motor driver shield can control up to four bi-directional DC motors with 8-bit speed selection, two stepper motors, and two servo motors.

The L293D motor driver shield includes two L293 motor driver ICs and a 74HC595 shift register IC. The shield has several important components.

The motor driver: The L293D is a dual-channel H-bridge motor driver that can control two DC motors or a stepper motor at one time. As there are two L293D ICs on the shield, it's technically capable of controlling four DC motors. This is ideal for two and four-wheel robot platforms. The IC consists of two H-bridges to control the motors, each delivering p to 0.6A to a motor.

The shift register: The 74HC595 is an 8-bit serial input and serial/parallel output shift register. It's used to extend four Arduino GPIO (or another microcontroller) to eight direction control pins for two of the L293D motor driver ICs.

The power supply: The power supply to the shield can be used for both the shield and Arduino or the two can use separate power supplies. If sharing a common power supply, a power jumper must be placed on the shield. The power supply can be provided through Arduino's USB port, its DC jack, or from the shield's 2-pin EXT-PWR block. If a separate power supply is used, the shield's power must be input at the 2-pin EXT-PWR block.

To control the DC motors, use:

- Pin 11 for the motor port M1
- Pin 3 for the motor port M2
- Pin 5 for the motor port M3
- Pin 6 for the motor port M4
- Pins 4, 7, 8, and 12 may also all be used.

To control the stepper motors, use:

- Pins 11 and 3 for the motor port M1-M2
- Pins 5 and 6 for the motor port M2-M3
- Pins 4, 7, 8, and 12 may also all be used.

3.1.3 ULTRASONIC SENSORS:

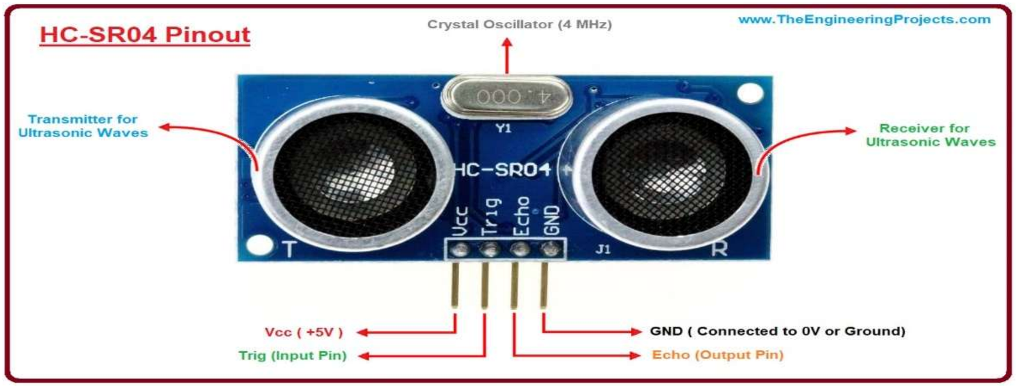


FIGURE 3.1.3

An ultrasonic sensor is an instrument that measures the distance to an object using ultrasonic sound waves. An ultrasonic sensor uses a transducer to send and receive ultrasonic pulses that relay back information about an object's proximity. High-frequency sound waves reflect from boundaries to produce distinct echo patterns.

Ultrasonic sensors work by sending out a sound wave at a frequency above the range of human hearing. The transducer of the sensor acts as a microphone to receive and send the ultrasonic sound. Our ultrasonic sensors, like many others, use a single transducer to send a pulse and to receive the echo. The sensor determines the distance to a target by measuring time lapses between the sending and receiving of the ultrasonic pulse.

The working principle of this module is simple. It sends an ultrasonic pulse out at 40kHz which travels through the air and if there is an obstacle or object, it will bounce back to the sensor. By calculating the travel time and the speed of sound, the distance can be calculated.

Ultrasonic sensors are a great solution for the detection of clear objects. For liquid level measurement, applications that use infrared sensors, for instance, struggle with this particular use case because of target translucence.

For presence detection, ultrasonic sensors detect objects regardless of the color, surface, or material (unless the material is very soft like wool, as it would absorb sound.) To detect transparency and other items where optical technologies may fail, ultrasonic sensors are a reliable choice.

Our ultrasonic distance, level, and proximity sensors are commonly used with microcontroller platforms like Raspberry Pi, ARM, PIC, Arduino, Beagle Board, and more. Ultrasonic sensors transmit sound waves toward a target and will determine its distance by measuring the time it took for the reflected waves to return to the receiver. This sensor is an electronic device that will measure the distance of a target by transmitting ultrasonic sound waves and then will convert the reflected sound into an electrical signal.

Our sensors are often used as proximity sensors. Ultrasonic sensors are also used in obstacle avoidance systems, as well as in manufacturing. Our Short Range sensors offer the opportunity for closer range detection where you may need a sensor that ranges objects as close to 2cm. These are also built with very low power requirements in mind, as well as environments where noise rejection is necessary.

Ultrasonics are Independent of:

- Light
- Smoke
- Dust
- Color
- Material (except for soft surfaces, i.e., wool, because the surface absorbs the ultrasonic sound wave and doesn’t reflect sound.). This is the HC-SR04 ultrasonic distance sensor. This economical sensor provides 2cm to 400cm of non-contact measurement functionality with a ranging accuracy that can reach up to 3mm. Each HC-SR04 module includes an ultrasonic transmitter, a receiver and a control circuit.

- There are only four pins that you need to worry about on the HC-SR04: VCC (Power), Trig (Trigger), Echo (Receive), and GND (Ground). You will find this sensor very easy to set up and use for your next range-finding project!
- This sensor has additional control circuitry that can prevent inconsistent "bouncy" data depending on the application.

3.1.4 BLUETOOTH MODULE: -

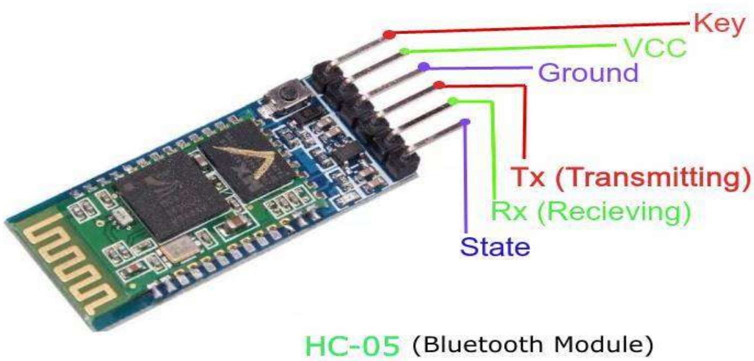


FIGURE:3.1.4

STATE	STORES STATE OF THE CONNECTION. PAIRED OR DISCONNECTED
RX	RECIEVE PIN OF THE MODULE. VOLTAGE DIVIDER IS USED TO CONNECT IT TO TX
TX	TRANSMIT PIN. CAN BE CONNECTED DI-RECTLY TO RX OF MCU
GND	GROUND PIN. FOR POWER INPUT
VCC	FIVE VDC PIN. FOR POWER INPUT
EN	ENABLE PIN. TO ENABLE OR DISABLE BT.

- The **HC-05** is a popular module which can add two-way (full-duplex) wireless functionality to your projects. You can use this module to communicate between two microcontrollers like Arduino or communicate with any device with Bluetooth functionality like a Phone or Laptop. There are many android applications that are already available which makes this process a lot easier. The **HC-05** has two operating modes, one is the Data mode in which it can send and receive data from other

Bluetooth devices and the other is the AT Command mode where the default device settings can be changed. We can operate the device in either of these two modes by using the key pin as explained in the pin description.

It is very easy to pair the HC-05 module with microcontrollers because it operates using the Serial Port Protocol (SPP). Simply power the module with +5V and connect the Rx pin of the module to the Tx of MCU and Tx pin of module to Rx of MCU During power up the key pin can be grounded to enter Command mode,

Technical Specifications: -

- Serial Bluetooth module for Arduino and other microcontrollers
- Operating Voltage: 4V to 6V (Typically +5V)
- Operating Current: 30mA
- Range: <100m
- Works with Serial communication (USART) and TTL compatible
- Follows IEEE 802.15.1 standardized protocol
- Uses Frequency-Hopping Spread spectrum (FHSS)
- Can operate in Master, Slave or Master/Slave mode
- Can be easily interfaced with Laptop or Mobile phones with Bluetooth.

The below table shows the HC-05 Configurations are as follows.

PIN NO	PIN NAME	DESCRIPTIONS
1	Enable / Key	This pin is used to toggle between Data Mode (set low) and AT command mode (set high). By default, it is in Data mode.
2	VCC	Powers the module. Connect to +5V Supply voltage.
3	Ground	Ground pin of module, connect to system ground.
4	TX Transmitter	Transmits Serial Data, everything received connected to this pin will be broadcasted via Bluetooth.
5	RX Receiver	Receive Serial Data. Every serial data given to this pin will be broadcasted via Bluetooth.
6	State	The state pin is connected to on board LED, it can be used as a feedback to check if Bluetooth is working properly.
7	LED	Indicates the status: - • Blink once in 2 sec: Module has entered Command Mode • Repeated Blinking: Waiting for connection in Data Mode • Blink twice in 1 sec: Connection successful in Data Mode
8	Button	Used to control the Key/Enable pin to toggle between Data and command Mode.

TABLE 1

Pair HC-05 and smartphone:

1. Search for new Bluetooth device from your phone. You will find Bluetooth device with “HC-05” name.
2. Click on connect/pair device option; default pin for HC-05 is 1234 or 0000.

After pairing two Bluetooth devices, open terminal software (e.g., Teraterm, Realterm etc.) in PC, and select the port where we have connected USB to serial module. Also select default baud rate of 9600 bps. In smart phone, open Bluetooth terminal application and connect to paired device HC-05.

It is simple to communicate, we just must type in the Bluetooth terminal application of smartphone. Characters will get sent wirelessly to Bluetooth module HC-05. HC-05 will automatically transmit it serially to the PC, which will appear on terminal. Same way we can send data from PC to smartphone.

3.1.5 DC MOTORS: -

FIGURE 3.1.5.1

A **DC motor** is any of a class of rotary electrical motors that converts direct current electrical energy into mechanical energy. The most common types rely on the forces produced by magnetic fields. All types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current in part of the motor.

DC motors were the first form of motor widely used, as they could be powered from existing direct-current lighting power distribution systems. A DC motor's speed can be controlled over a wide range, using either a variable supply voltage or by changing the strength of current in its field windings. Small DC motors are used in tools, toys, and appliances. The universal motor can operate on direct current but is a lightweight brushed motor used for portable power tools and appliances. Larger DC motors are currently used in propulsion of electric vehicles, elevator and hoists, and in drives for steel rolling mills. The advent of power electronics has made replacement of DC motor with AC motors possible in many applications.

DC motor is highly used by hobbyists who start exploring electronics. Hence this motor is very simple and easy to use. You can use any normal 9V battery or even a 5V supply since this motor has a operating ranges from 4.5V to 9V. To make it rotate just connect the positive (+) side of battery to one terminal and the Negative (-) sign of the battery to the other end and you should see the motor rotating. If you want to reverse the speed of the motor, simply interchange the terminals and direction will also be reversed. To control the speed of the motor you must vary the voltage supplied to the Motor the easiest way to do this is using a Potentiometer. There are also many other ways to achieve this. Also remember that the motor can consume up to 250mA during loaded conditions so make sure you supply could source it. If you are controlling, it through any Digital IC or any Microcontroller you should use a motor driver IC like, L293D or ULN2003 these IC’s will also let you to control the direction of the motor easily.

SPECIFICATIONS: -

- standard 130 Type DC motor.
- Operating Voltage: 4.5V to 9V.
- Recommended/Rated Voltage: 6V
- Current at No load: 70mA (max).
- No-load Speed: 9000 rpm.
- Loaded current: 250mA (approx.).
- Rated Load: 10g*cm.
- Motor Size: 27.5mm x 20mm x 15mm.
- Weight: 17 grams.

FIGURE 3.1.5.2

Applications

- Toy cars
- Windmill projects
- Basic Electronics projects
- As Robot wheels.

3.1.6 SOIL MOISTURE SENSORS: -

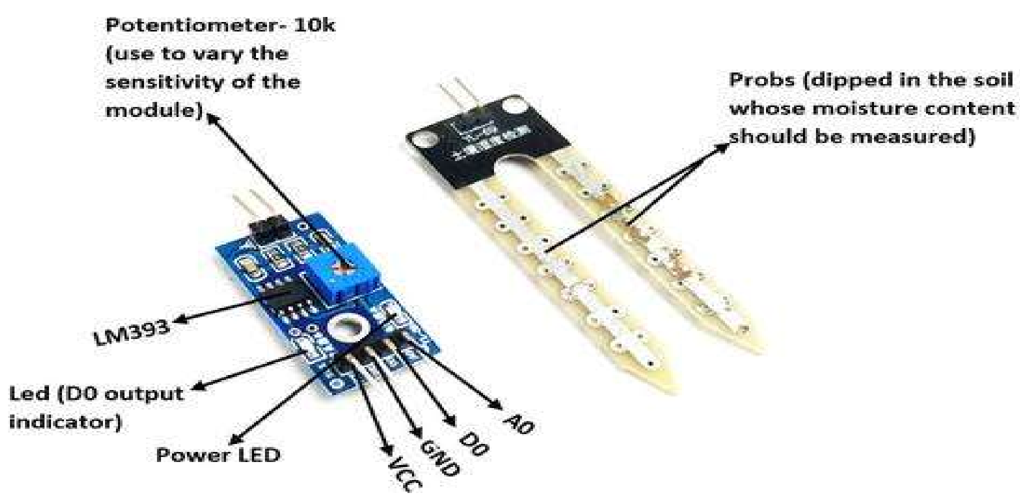


FIGURE 3.1.6.1

Soil moisture sensors measure the volumetric water content in soil.[1] Since the direct gravimetric measurement of free soil moisture requires removing, drying, and weighing of a sample, soil moisture sensors measure the volumetric water content indirectly by using some other property of the soil, such as electrical resistance, dielectric constant, or interaction with neutrons, as a proxy for the moisture content.

The relation between the measured property and soil moisture must be calibrated and may vary depending on environmental factors such as soil type, temperature, or electric conductivity. Reflected microwave radiation is affected by the soil moisture and is used for remote sensing in hydrology and agriculture. Portable probe instruments can be used by farmers or gardeners.

Soil moisture sensors typically refer to sensors that estimate volumetric water content. Another class of sensors measure another property of moisture in soils called water potential; these sensors are usually referred to as soil water potential sensors and include tensiometers and gypsum blocks. This **soil moisture sensor module** is used to detect the moisture of the soil. It measures the volumetric content of water inside the soil and gives us the moisture level as output. The module has both digital and Analog outputs and a potentiometer to adjust the threshold level.

Moisture Sensor Module Features & Specifications

- Operating Voltage: 3.3V to 5V DC
- Operating Current: 15mA
- Output Digital - 0V to 5V, Adjustable trigger level from preset
- Output Analog - 0V to 5V based on infrared radiation from fire flame falling on the sensor
- LEDs indicating output and power
- PCB Size: 3.2cm x 1.4cm
- LM393 based design
- Easy to use with Microcontrollers or even with normal Digital/Analog IC small, cheap and easily available.

About Soil moisture sensor module is as follows,

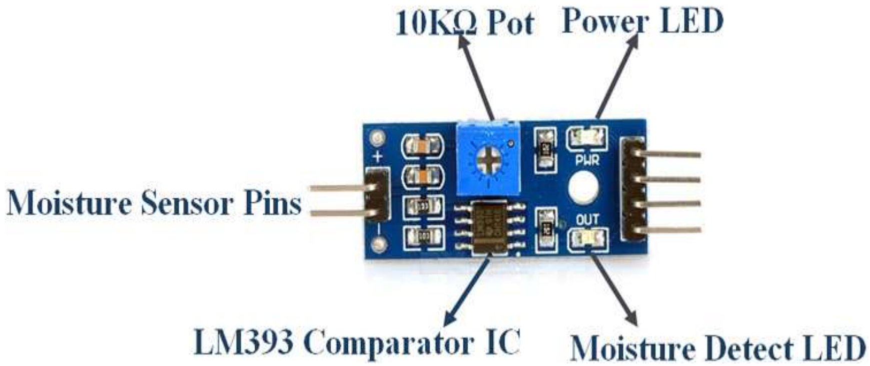


FIGURE 3.1.6.2

LM393 IC

LM393 Comparator IC is used as a voltage comparator in this Moisture sensor module. Pin 2 of LM393 is connected to Preset (10KΩ Pot) while pin 3 is connected to Moisture sensor pin. The comparator IC will compare the threshold voltage set using the preset (pin2) and the sensor pin (pin3).

Moisture Sensor

The moisture sensor consists of two probes that are used to detect the moisture of the soil**.** The moisture sensor probes are coated with immersion gold that protects Nickel from oxidation. These two probes are used to pass the current through the soil and then the sensor reads the resistance to get the moisture values.

Preset (Trimmer pot)

Using the onboard Preset you can adjust the threshold (sensitivity) of the digital output.

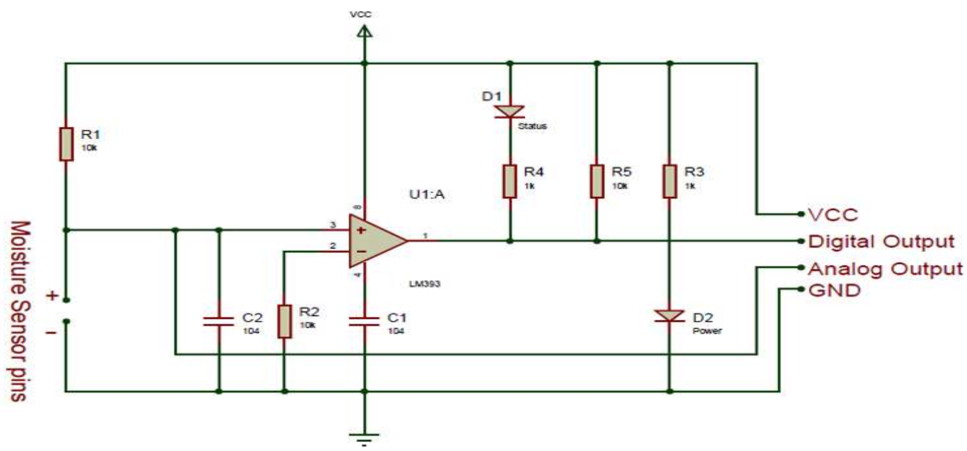


FIGURE:3.1.6.2

The Above Circuit Is for Soil sensor Module. Moisture sensor module consists of four pins i.e. VCC, GND, DO, AO. Digital out pin is connected to the output pin of LM393 comparator IC while the analog pin is connected to Moisture sensor. Using a Moisture sensor module with a microcontroller is very easy. Connect the Analog/Digital Output pin of the module to the Analog/Digital pin of Microcontroller.

Connect VCC and GND pins to 5V and GND pins of Microcontroller. After that insert the probe inside the soil. When there is more water present in the soil, it will conduct more electricity that means resistance will be low and the moisture level will be high.

3.1.7 SERVO MOTOR: -

FIGURE:3.1.7

A Servo Motor is a small device that has an output shaft. This shaft can be positioned to specific angular positions by sending the servo a coded signal. As long as the coded signal exists on the input line, the servo will maintain the angular position of the shaft. If the coded signal changes, the angular position of the shaft changes. In practice, servos are used in radio-controlled airplanes to position control surfaces like the elevators and rudders. They are also used in radio-controlled cars, puppets, and of course, robots.

Servos are extremely useful in robotics. The motors are small, have built-in control circuitry, and are extremely powerful for their size. It also draws power proportional to the mechanical load. A lightly loaded servo, therefore, does not consume much energy the guts of a servo motor is shown in the following picture. You can see the control circuitry, the motor, a set of gears, and the case. You can also see the 3 wires that connect to the outside world. One is for power (+5volts), ground, and the white wire is the control wire.

A servo consists of a Motor (DC or AC), a potentiometer, gear assembly, and a controlling circuit. First of all, we use gear assembly to reduce RPM and to increase torque of the motor. Say at initial position of servo motor shaft, the position of the potentiometer knob is such that there is no electrical signal generated at the output port of the potentiometer. Now an electrical signal is given to another input terminal of the error detector amplifier. Now the difference between these two signals, one comes from the potentiometer, and another comes from other sources, will be processed in a feedback mechanism and output will be provided in terms of error signal.

This error signal acts as the input for the motor and motor starts rotating. Now the motor shaft is connected with the potentiometer and as the motor rotates so the potentiometer and it will generate a signal. So, as the potentiometer's angular position changes, its output feedback signal changes. After some time, the position of potentiometer reaches a position where the output of potentiometer is same as external signal provided.

At this condition, there will be no output signal from the amplifier to the motor input as there is no difference between external applied signal and the signal generated at potentiometer, and in this situation motor stops rotating.

Working:

The servo motor has some control circuits and a potentiometer (a variable resistor, aka pot) connected to the output shaft. In the picture above, the pot can be seen on the right side of the circuit board. This pot allows the control circuitry to monitor the current angle of the servo motor.

If the shaft is at the correct angle, then the motor shuts off. If the circuit finds that the angle is not correct, it will turn the motor until it is at a desired angle. The output shaft of the servo is capable of traveling somewhere around 180 degrees. Usually, it is

somewhere in the 210-degree range, however, it varies depending on the manufacturer. A normal servo is used to control an angular motion of 0 to 180 degrees. It is mechanically not capable of turning any farther due to a mechanical stop built on to the main output gear.

The power applied to the motor is proportional to the distance it needs to travel. So, if the shaft needs to turn a large distance, the motor will run at full speed. If it needs to turn only a small amount, the motor will run at a slower speed.

Servo motor applications are also commonly seen in remote-controlled toy cars for controlling the direction of motion, and it is also very widely used as the motor which moves the tray of a CD or DVD player. Besides these, there are hundreds of servo motor applications we see in our daily life.

Advantages:

- If a heavy load is placed on the motor, the driver will increase the current to the motor coil as it attempts to rotate the motor. There is no out-of-step condition.
- High-speed operation is possible.

Disadvantages:

- Since the servomotor tries to rotate according to the command pulses but lags, it is not suitable for precision control of rotation.
- Higher cost.
- When stopped, the motor's rotor continues to move back and forth one pulse, so that it is not suitable if you need to prevent vibration.

APPLICATIONS: -

- In Industries they are used in machine tools, packaging, factory automation, material handling, printing converting, assembly lines, and many other demanding applications robotics, CNC machinery, or automated manufacturing.
- They are also used in radio-controlled airplanes to control the positioning and movement of elevators.

- They are used in robots because of their smooth switching on and off and accurate positioning.
- They are also used by the aerospace industry to maintain hydraulic fluid in their hydraulic systems.
- They are used in many radio-controlled toys.

- They are used in electronic devices such as DVDs or Blue-ray Disc players to extend or replay the disc trays.
- They are also being used in automobiles to maintain the speed of vehicles.

3.1.8 LITHIUM-ION BATTERIES: -

FIGURE:3.1.8

A lithium-ion battery or Li-ion battery is a type of rechargeable battery.A lithium-ion (Li-ion) battery is an advanced battery technology that uses lithium ions as a key component of its electrochemistry. During a discharge cycle, lithium atoms in the anode are ionized and separated from their electrons. ... Li-ion batteries can use a number of different materials as electrodes.The typical estimated life of a Lithium-Ion battery is about two to three years or 300 to 500 charge cycles, whichever occurs first. One charge cycle is a period of use from fully charged, to fully discharged, and fully recharged again.

Advantages:-

- High energy density
- Self-discharge
- Low maintenance
- Cell voltage
- Load characteristics
- No requirement for priming

Disadvantages :-

- It lasts only two to three years after manufacturer.
- It is sensitive to high temperatures.
- If the battery is completely discharged, it can no longer be recharged again.
- It is relatively expensive.

3.2 SOFTWARE COMPONENTS: -

ARDUNIO AND ITS SOFTWARE

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

The Arduino board is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc.[2][3] The board is equipped with sets of digital and Analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits.[1] The board has 14 digital I/O pins (six capable of PWM output), 6 Analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable.[4] It can be powered by the USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. It is similar to the Arduino Nano and Leonardo.[5][6] The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website.

The Arduino software is published as open-source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.

The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it.

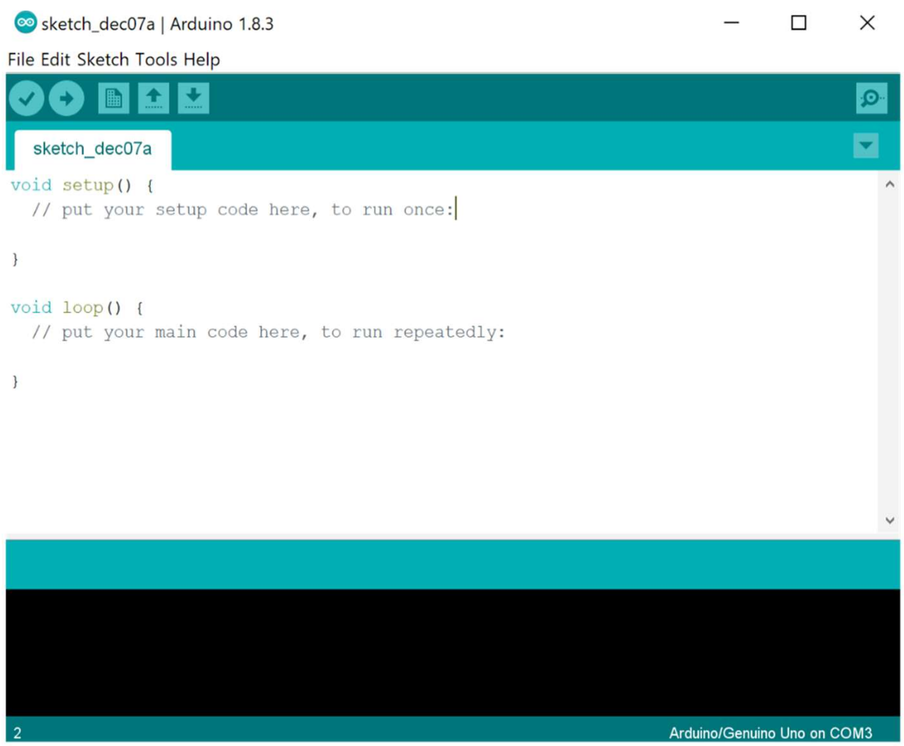


FIGURE 3.2

A program for Arduino hardware may be written in any programming language with compilers that produce binary machine code for the target processor. Atmel provides a development environment for their 8-bit AVR and 32-bit ARM Cortex-M based microcontrollers: AVR Studio (older) and Atmel Studio (newer).

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, and Linux) that is written in the Java programming language. It originated from the IDE for the languages Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2.

The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub *main()* into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

Arduino Pro IDE (alpha preview) was released. The system still uses Arduino CLI (Command Line Interface), but improvements include a more professional development environment, autocompletion support, and Git integration. The application frontend is

based on the Eclipse Theia Open-Source IDE. The main features available in the alpha release are:

- Modern, fully featured development environment
- Dual Mode, Classic Mode (identical to the Classic Arduino IDE) and Pro Mode (File System view)
- New Board Manager
- New Library Manager
- Board List
- Basic Auto-Completion (Arm targets only)
- Git Integration
- Serial Monitor
- Dark Mode.

A *sketch* is a program written with the Arduino IDE.[64] Sketches are saved on the development computer as text files with the file extension. ino. Arduino Software (IDE) pre-1.0 saved sketches with the extension. pde.

A minimal Arduino C/C++ program consists of only two functions:

- setup (): This function is called once when a sketch starts after power-up or reset. It is used to initialize variables, input and output pin modes, and other libraries needed in the sketch. It is analogous to the function main ().
- loop (): After setup () function exits (ends), the loop () function is executed repeatedly in the main program. It controls the board until the board is powered off or is reset.

Most Arduino boards contain a light-emitting diode (LED) and a current limiting resistor connected between pin 13 and ground, which is a convenient feature for many tests and program functions.[68] A typical program used by beginners, akin to Hello, World!, is "blink", which repeatedly blinks the on-board LED integrated into the Arduino board. This program uses the functions pin Mode(), digitalWrite(), and delay(), which are provided by the internal libraries included in the IDE environment.[69][70][71] This program is usually loaded into a new Arduino board by the manufacturer.

Libraries

The open-source nature of the Arduino project has facilitated the publication of many free software libraries that other developers use to augment their projects.

Operating Systems

There is a Xinu OS port for the atmega328p (Arduino Uno and others with the same chip), which includes most of the basic features.[72] The source code of this version is freely available.

There is also a threading tool, named Protothreads. Protothreads are described as extremely lightweight stackless threads designed for severely memory constrained systems, such as small embedded systems or wireless sensor network nodes. Protothreads provide linear code execution for event-driven systems implemented in C.

CHAPTER 4

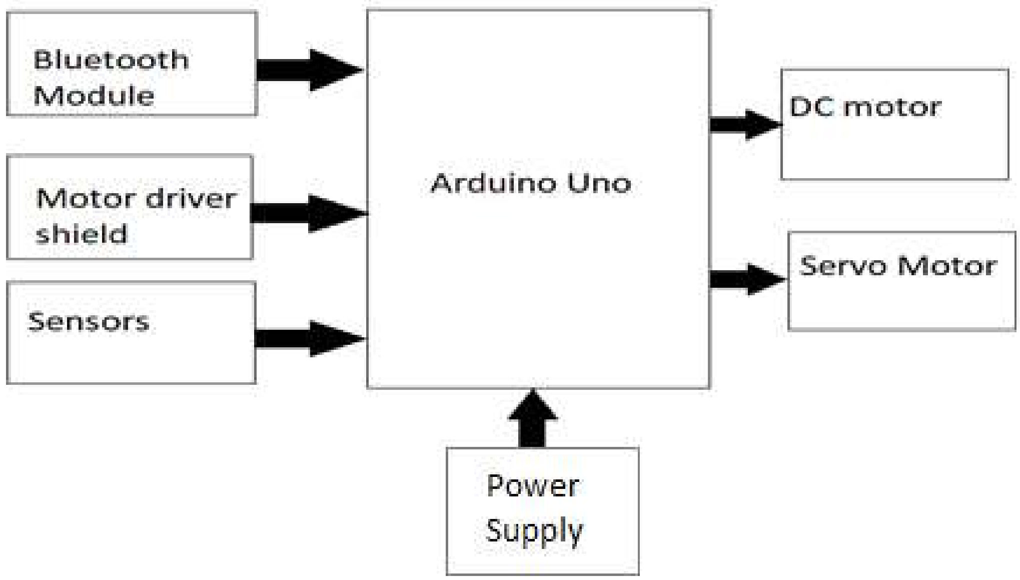
IMPLEMENTATION AND METHADOLOGY

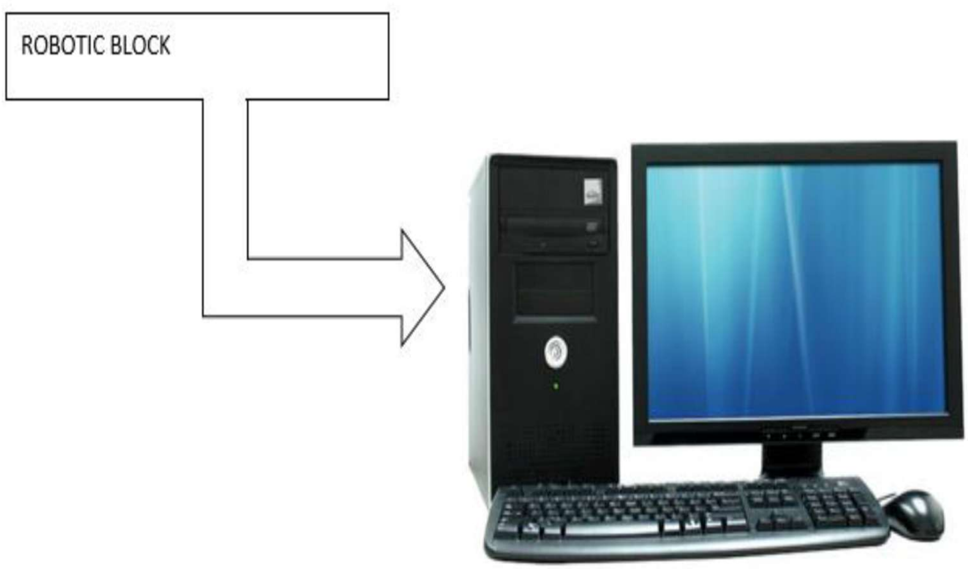
4.1 METHODOLOGY

The first step chosen was to do literature survey to know more about the inspection methods, different types of robots and design constraints related to each one of them, the mechanism chosen by each of them, etc. This phase dealt with referring to many standard papers. After the literature survey, the most plausible design was decided upon and all the calculations associated with it were done. Different problems that occurred with agriculture were jotted down and identification of methods to treat them was found out. The main objective of the project was to improve the seed sowing process by means of automated way. HC-05 Bluetooth module is used to operate the robot through mobile app. Soil moisture sensor is used to check the moisture conditions of soil. Seed sowing is done with the help of 360-degree continuous rotation servo motor. Ultrasonic sensors which can transmit and receive ultrasonic pulses can detect obstacles within the mentioned range. The codes required for seed sowing process and to detect obstacle were written and dumped on an Arduino Uno. The controlling system consisting of microcontroller, motor driver shield, dc motors and different sensors was mounted on the model and synchronized with the mechanical part.

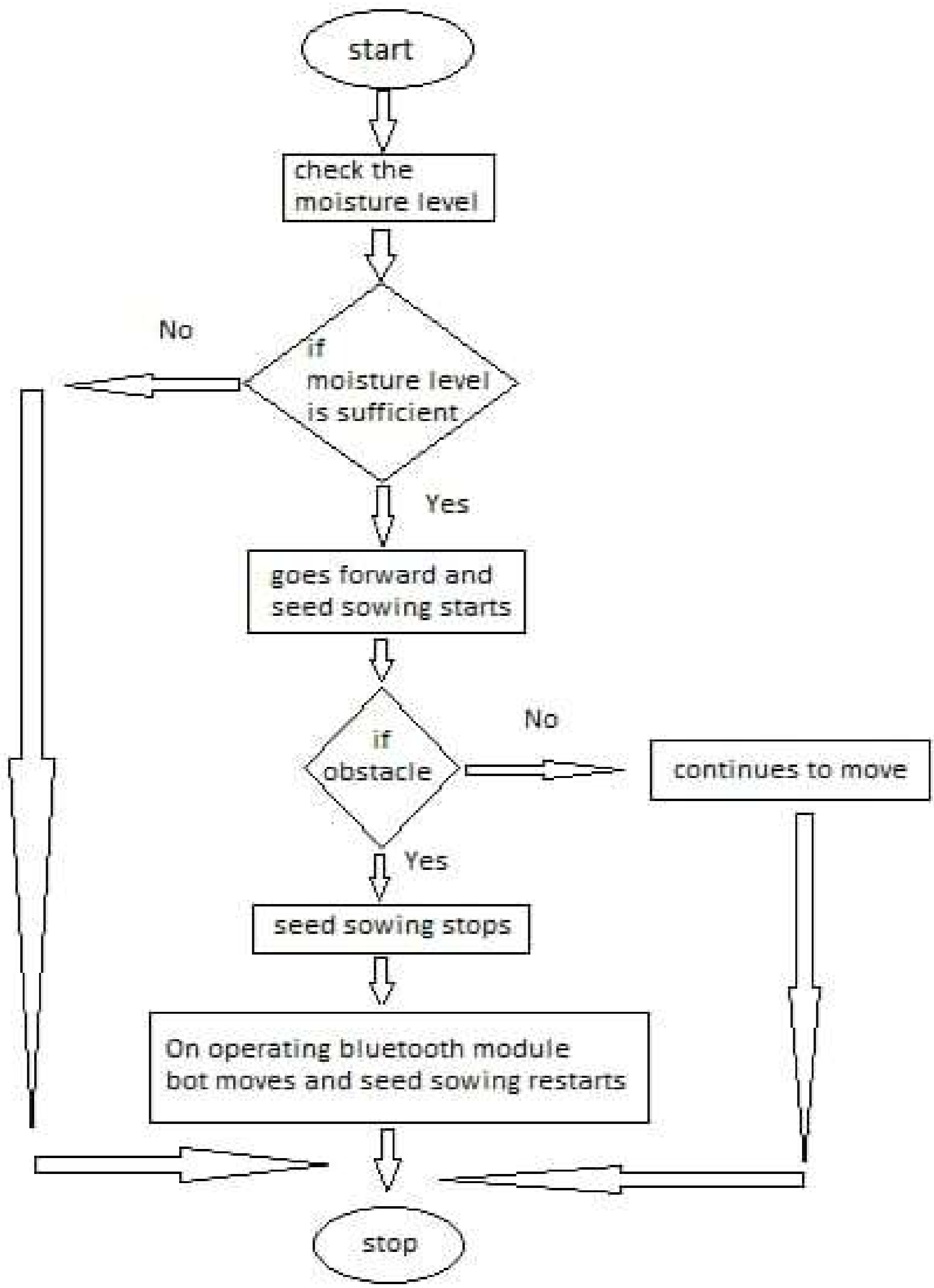
As shown in the block diagram, input to the Arduino Uno is Bluetooth module, motor driver shields which is used to run DC motors and different sensors. In the output part, DC motors are used for the movement of robots and servo motors are used for seed sowing process and with soil moisture sensor.

BLOCK DIAGRAM:





FLOWCHART


start
check the moisture level
No
if moisture level is sufficient
Yes
goes forward and seed sowing starts
No
if obstacle
continues to move
Yes
seed sowing stops
On operating bluetooth module bot moves and seed sowing restarts
stop

4.2 IMPLEMENTATION

Initially the soil moisture sensor is operated using the Bluetooth module. A separate button is created for the soil moisture sensor operation and on pressing the button the servo motor that helps in immersing the moisture sensor is activated. The soil moisture level is indicated on the android device. Once we know that the moisture level is sufficient, we can consider sowing the seeds. The servo motor automatically lifts the moisture sensor back after a few seconds. After the moisture value is known we start operating the bot's movement using the android device.

When the forward button on the device is pressed, the bot starts moving forward. The servo motor of the seed sowing wheel also starts rotating simultaneously with the bot in motion. If the obstacle appears in front of the bot, the sensor that is fixed in front of the bot immediately sense the obstacle and the bot stops at a certain distance away from the bot. The distance at which the bot must stop when the obstacle detected is mentioned in the code. Using the left and right controls present on the device, the bot is made to travel around the obstacle. During the left-right movements of the bot, the seed sowing is stopped i.e., the servo motor of the seed sowing wheel does not rotate during the left-right movements of the bot to avoid the falling of seeds at random places. Once the bot is free without any obstacles, on pressing the forward button the bot is made to move forward again with seed sowing happening simultaneously.

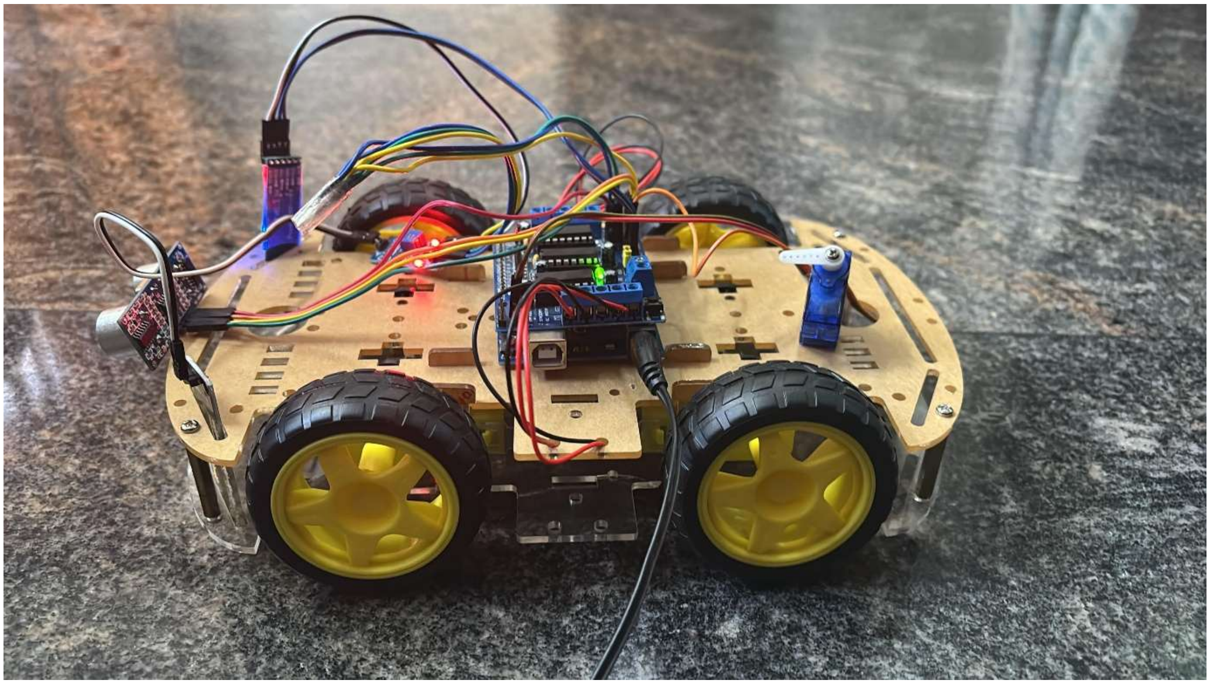


FIGURE 4.2

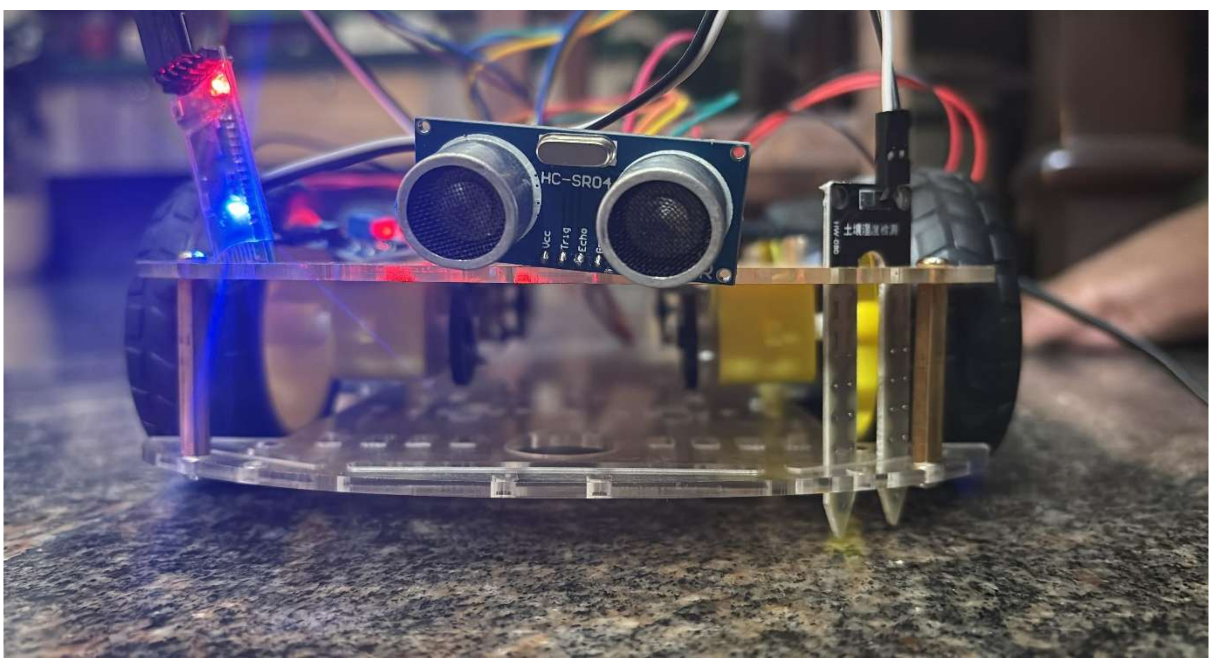


FIGURE 4.2.1

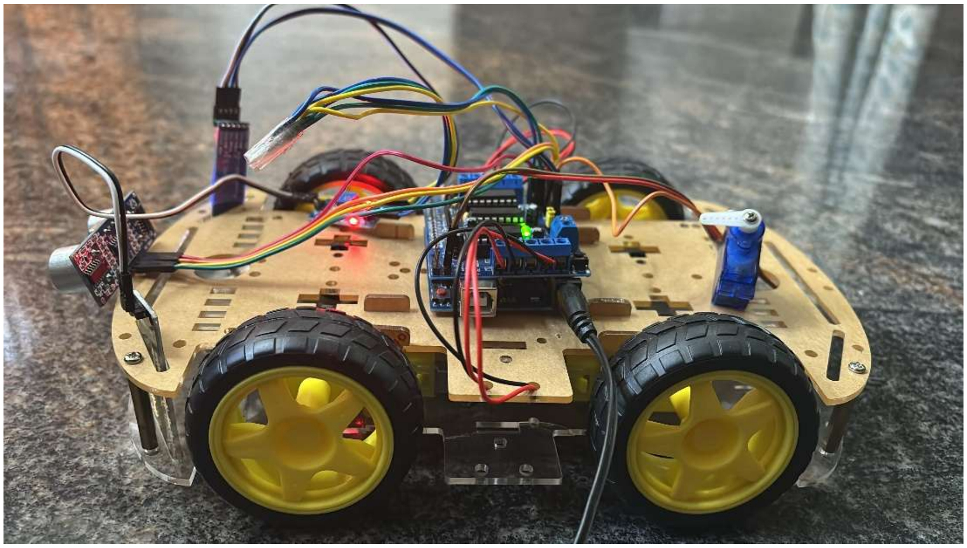


FIGURE 4.2.2

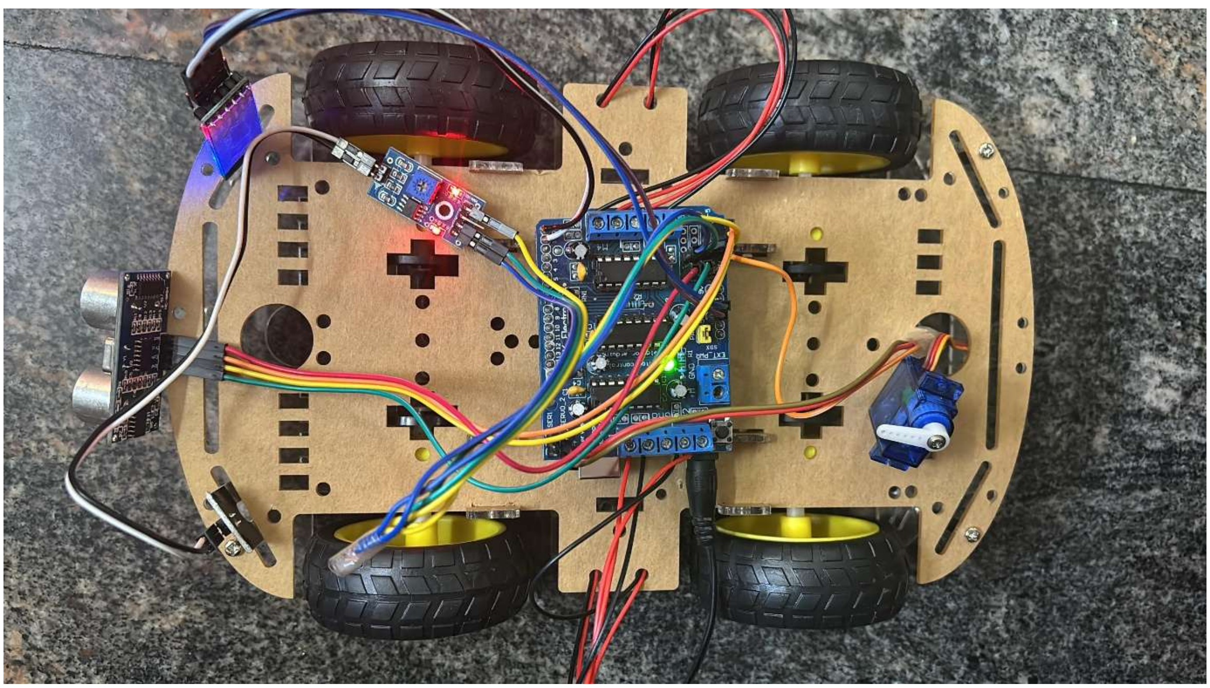


FIGURE 4.2.3

CHAPTER 5

INTERFACING

In computing, an interface is a shared boundary across which two or more separate components of a system exchange information. The exchange can be between software, hardware, peripheral devices, humans and combinations of these.

5.1 DC MOTOR WITH H-BRIDGE INTERFACING

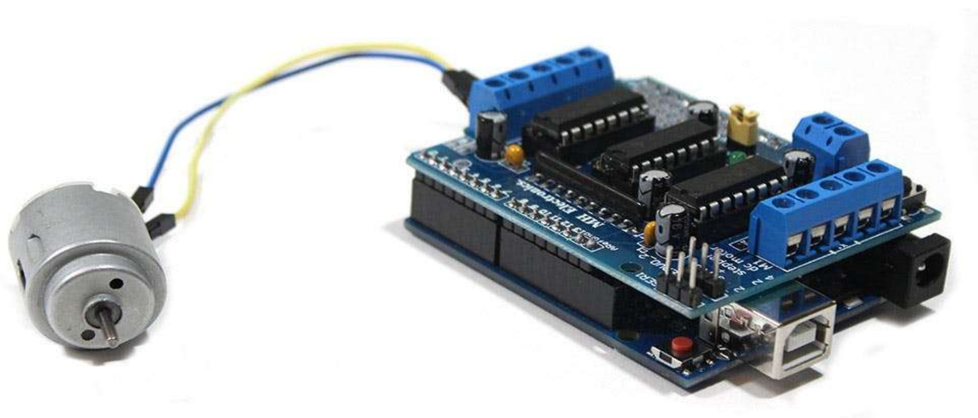


FIGURE 5.1.1

CONNECTIONS:

To connect the DC motors, two five-pin screw terminals are at the edges of the shield. These terminals are labelled as M1, M2, M3, and M4. Those DC motors with voltage ratings of between 4.5 to 25V can be connected to these terminals and will deliver power up to 600 mA.

To communicate with the shield, we need to install **AFMotor.h** library so that we can issue simple commands to control DC

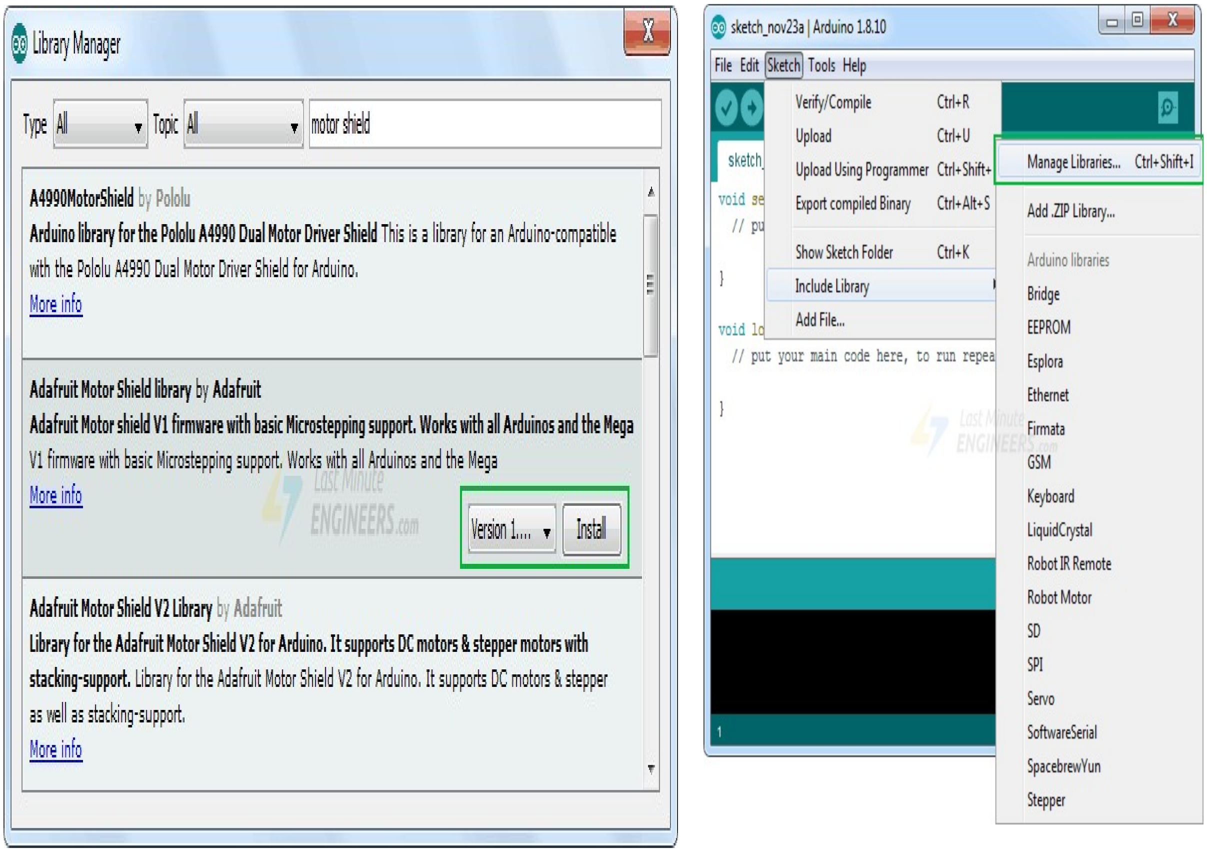

Library Manager
Type All
Topic All
motor shield
A4990MotorShield by Pololu
Arduino library for the Pololu A4990 Dual Motor Driver Shield This is a library for an Arduino-compatible with the Pololu A4990 Dual Motor Driver Shield for Arduino.
More info
Adafruit Motor Shield library by Adafruit
Adafruit Motor shield V1 firmware with basic Microstepping support. Works with all Arduinos and the Mega
V1 firmware with basic Microstepping support. Works with all Arduinos and the Mega
More info
Version 1....
Install
Adafruit Motor Shield V2 Library by Adafruit
Library for the Adafruit Motor Shield V2 for Arduino. It supports DC motors & stepper motors with stacking-support. Library for the Adafruit Motor Shield V2 for Arduino. It supports DC motors & stepper as well as stacking-support.
More info
sketch_nov23a | Arduino 1.8.10
File Edit Sketch Tools Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+
Export compiled Binary Ctrl+Alt+S
Show Sketch Folder Ctrl+K
Include Library
Add File...
// put your main code here, to run repea
Manage Libraries... Ctrl+Shift+I
Add .ZIP Library...
Arduino libraries
Bridge
EEPROM
Esplora
Ethernet
Firmata
GSM
Keyboard
LiquidCrystal
Robot IR Remote
Robot Motor
SD
SPI
Servo
SoftwareSerial
SpacebrewYun
Stepper

FIGURE 5.1.2

```
#include <AFMotor.h>
```

The Library you need to control the motor:

```
AF_DCMotor motor(1, MOTOR12_64KHZ)
```

Defining the DC motor you are using.

The first argument stands for the number of the motors in the shield and the second one stands for the motor speed control frequency. The second argument can be MOTOR12_2KHZ, MOTOR12_8KHZ, MOTOR12_8KHZ, and MOTOR12_8KHZ for motors number 1 and 2, and it can be MOTOR12_8KHZ, MOTOR12_8KHZ, and MOTOR12_8KHZ for motors number 3 and 4. And if it left unchecked, it will be 1KHZ by default.

```
motor.setSpeed(200);
```

Defining the motor speed. It can be set from 0 to 255.

```
void loop() {
motor.run(FORWARD);
delay(1000);
motor.run(BACKWARD);
delay(1000);
motor.run(RELEASE);
delay(1000);
}
```

Function motor.run() specifies the motor's motion status. The status can be FORWARD, BACKWARD, and RELEASE. RELEASE is the same as the brake but it may take some time until the motor's full stop.

5.2 Ultrasonic sensor interfacing

Connections: The configuration pin of HC-SR04 is VCC (1), TRIG (2), ECHO (3), and GND (4). The **supply voltage** of VCC is **+5V** and you can attach TRIG and ECHO pin to any Digital I/O in your Arduino Board.

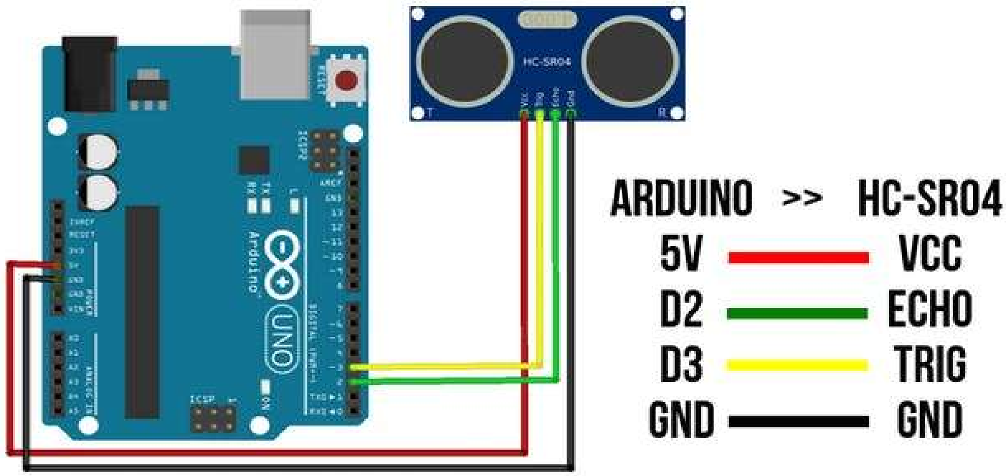
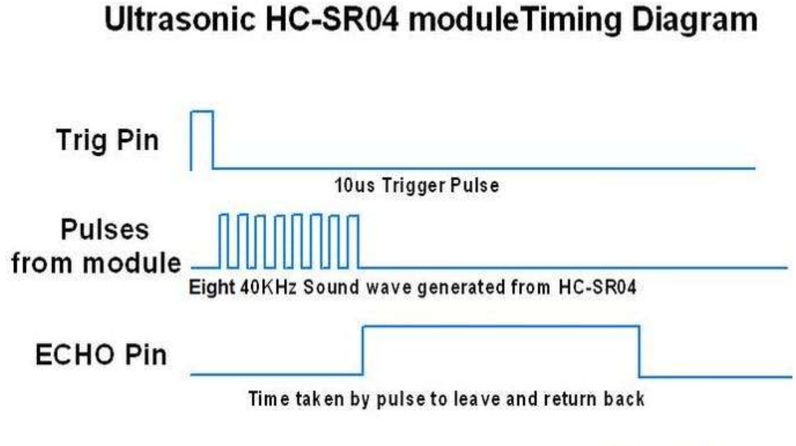


FIGURE:5.2

WORKING:

In order to generate the ultrasound, we need to set the Trigger Pin on a High State for 10 µs. That will send out an 8-cycle sonic burst which will travel at the speed sound and it will be received in the Echo Pin. The Echo Pin will output the time in microseconds the sound wave travelled.



Ultrasonic HC-SR04 timing diagram

FIGURE:5.2.1

For example, if the object is 20 cm away from the sensor, and the speed of the sound is 340 m/s or 0.034 cm/µs the sound wave will need to travel about 588 microseconds. But what you will get from the Echo pin will be double that number because the sound wave needs to travel forward and bounce backward. So, in order to get the distance in cm we need to multiply the received travel time value from the echo pin by 0.034 and divide it by 2.

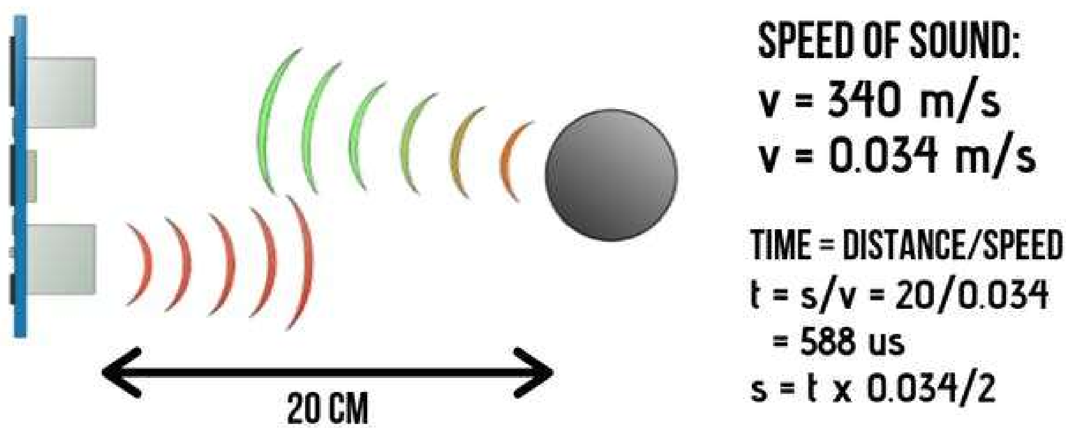


FIGURE 5.2.2

Distance calculating

CODE:

```
#define echoPin 2 // attach pin D2 Arduino to pin Echo of HC-SR04
#define trigPin 3 //attach pin D3 Arduino to pin Trig of HC-SR04

// defines variables
long duration; // variable for the duration of sound wave travel
int distance; // variable for the distance measurement

void setup() {
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT
  pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT
  Serial.begin(9600); // // Serial Communication is starting with
9600 of baudrate speed
  Serial.println("Ultrasonic Sensor HC-SR04 Test"); // print some
text in Serial Monitor
  Serial.println("with Arduino UNO R3");
}
void loop() {
  // Clears the trigPin condition
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin HIGH (ACTIVE) for 10 microseconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  // Reads the echoPin, returns the sound wave travel time in
microseconds
  duration = pulseIn(echoPin, HIGH);
  // Calculating the distance
  distance = duration * 0.034 / 2; // Speed of sound wave divided by
2 (go and back)
  // Displays the distance on the Serial Monitor
  Serial.print("Distance: ");
  Serial.print(distance);
```

```
  Serial.println(" cm");
}
```

5.3 HC 05 BLUETOOTH MODULE INTERFACING

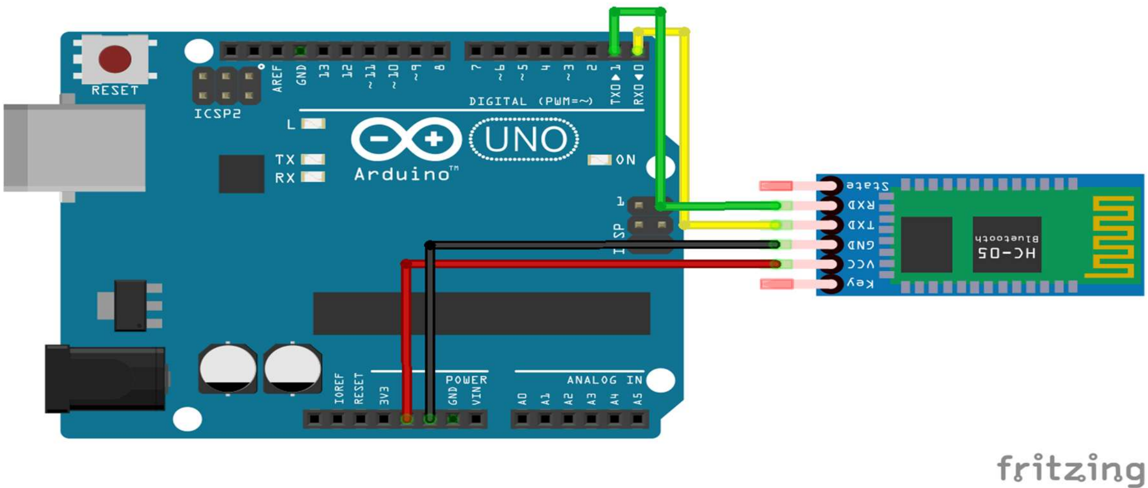


FIGURE 5.3

WORKING:

The module communicates with the help of USART (Universal Synchronous/Asynchronous Receiver/Transmitter) at the baud rate of 9600. and it also support other baud rate. So, we can interface this module with any microcontroller which supports USART. The HC-05 can operate in two modes. One is Data mode and other is AT command mode. When the enable pin is "LOW" the HC-05 is in Data Mode. If that pin set as "HIGH" the module is in AT command mode. Here we operate this module in Data Mode.

CONNECTIONS:

Rx - Tx

Tx - Rx

5v - +5v

GND - GND

```
char inputByte='z';
```

```
void setup(){
Serial.begin(9600);
pinMode(13,OUTPUT);
}
```

```
void loop() {
while(Serial.available()>0){
  inputByte= Serial.read();
  Serial.println(c);
  if (inputByte=='Z'){
  digitalWrite(13,HIGH);
  }
  else if (inputByte=='z'){
  digitalWrite(13,LOW);
  }
  }
}
```

5.4 **SERVO MOTOR INTERFACING**

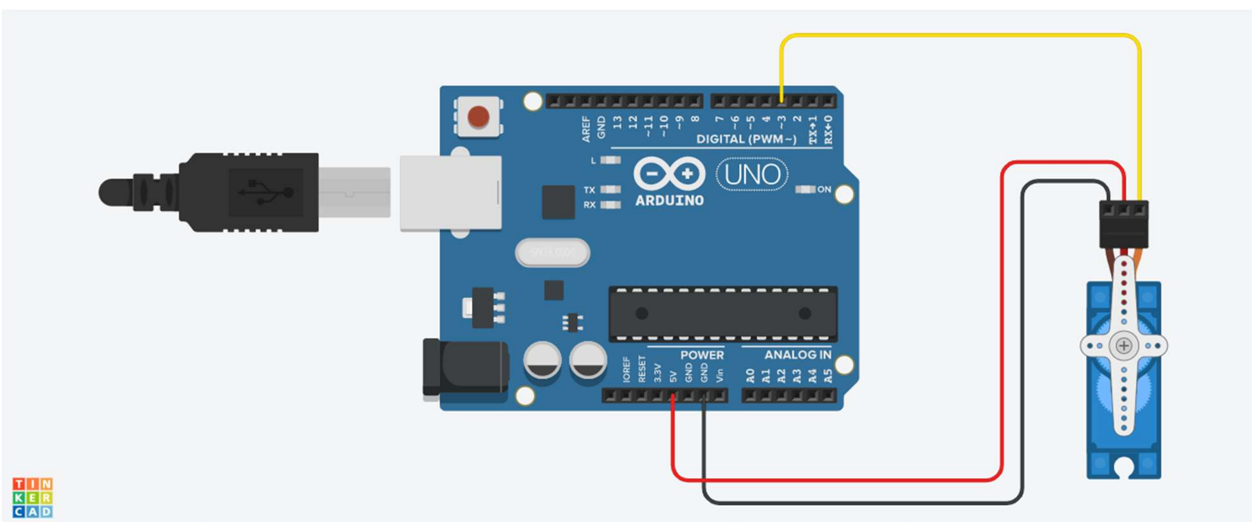


FIGURE 5.4

CONNECTIONS:

1. The servo motor has a female connector with three pins. The darkest or even black one is usually the ground. Connect this to the Arduino GND.
2. Connect the power cable that in all standards should be red to 5V on the Arduino.
3. Connect the remaining line on the servo connector to a digital pin on the Arduino

```
#include<Servo.h>
Servo Myservo;
int pos;
void setup ()
{
Myservo.attach(3);
}

void loop ()
{
  for (pos=0; pos<=180; pos++) {
Myservo.write(pos);
delay (15);
}
  delay (1000);
   for (pos=180; pos>=0; pos--) {
Myservo.write(pos);
delay (15);
}
 delay (1000);
  }
```

5.5 SOIL MOISTURE SENSOR INTERFACING

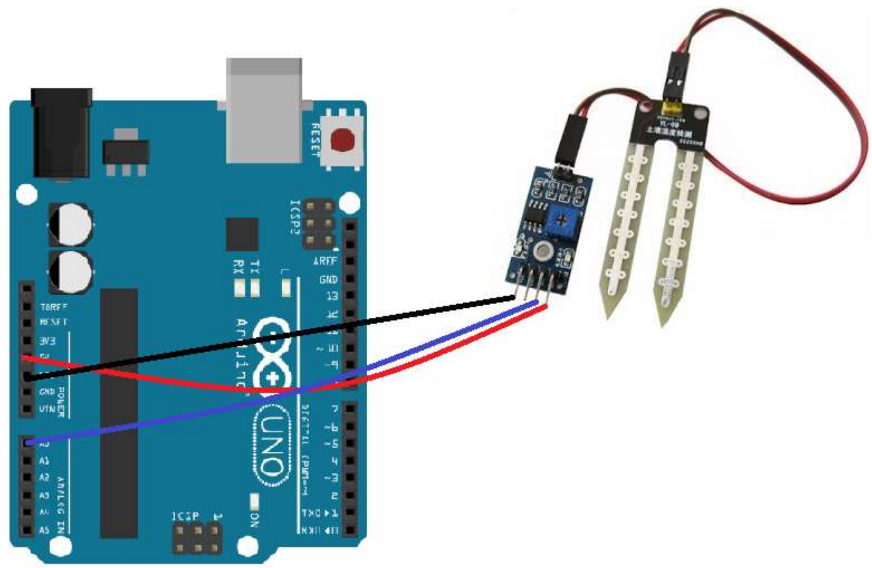


FIGURE 5.5

Working of Sensor

The soil moisture sensor consists of two probes which are used to measure the volumetric content of water. The two probes allow the current to pass through the soil and then it gets the resistance value to measure the moisture value.

When there is more water, the soil will conduct more electricity which means that there will be less resistance. Therefore, the moisture level will be higher. Dry soil conducts electricity poorly, so when there will be less water, then the soil will conduct less electricity which means that there will be more resistance. Therefore, the moisture level will be lower.

CONNECTIONS:

- VCC of FC-28 to 5V of Arduino
- GND of FC-28 to GND of Arduino
- A0 of FC-28 to A0 of Arduino

CODE:

```
int sensor_pin = A0;
int output_value;
void setup () {
Serial.begin(9600);
Serial.println("Reading from the Sensor ...");
delay (2000);
}
void loop () {
output_value= analogRead(sensor_pin);
output_value = map(output_value,550,0,0,100);
Serial.print("Mositure: ");
Serial.print(output_value);
Serial.println("%");
delay (1000);
}
```

5.6 COMPLETE CIRCUIT DIAGRAM

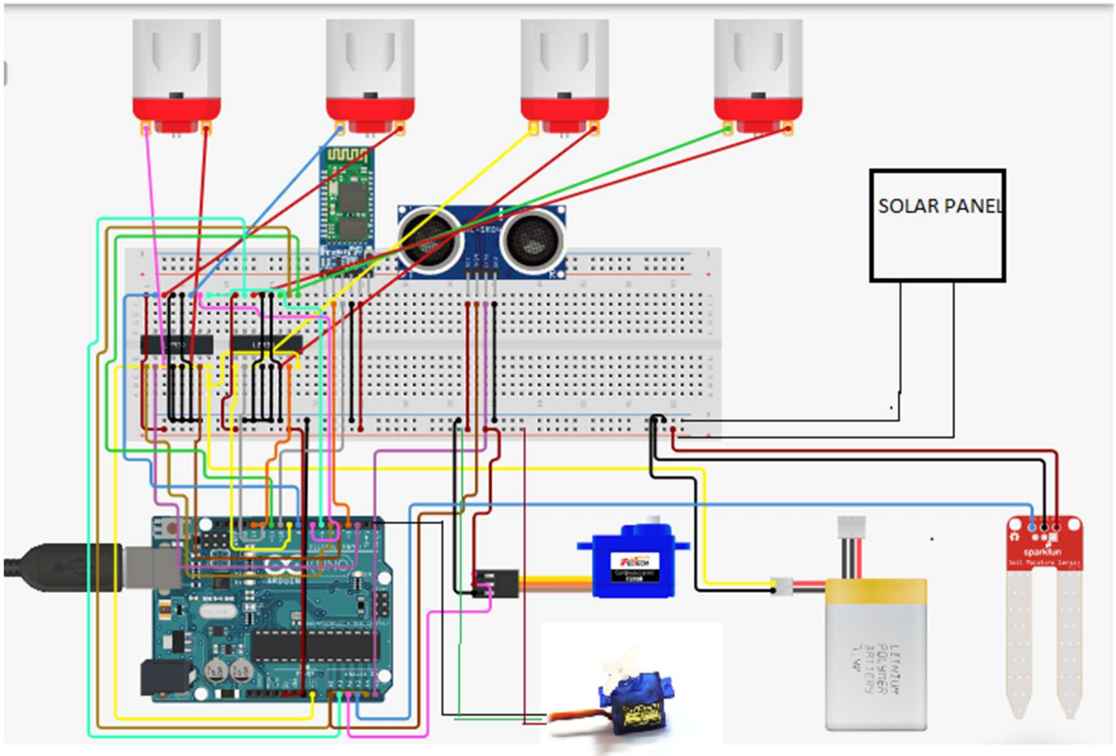


FIGURE 5.6

CHAPTER 6

RESULTS CONCLUSIONS AND FUTURE WORK

6.2 CONCLUSION

In this system we are operating an automated seed sowing robot .Seed Sowing Robot consists of Arduino Atmega 328P microcontroller, l293d driver, battery, servo motor,solar panels,ultrasonic sensor,soil moisture sensor.

Robot will sow the seed on the ground until it finds an obstacle.If the obstacle is found it stops and before sowing the seed on the field it usually turns right thus moving forward.

The servo motor is used to enables the soil moisture sensor into the ground to check the moisture content of the soil and thereby helps in sowing the seeds into the ground.

FIGURE:6.2

6.3 ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- It reduces human effort and work.
- It is of Low cost.
- It is very small in size and lightweight which makes it easy to carry.
- Moisture sensor checks the moisture level of soil which helps us to decide whether to sow the seed or not.
- No complexity in operating the robot thus making it user friendly.

DISADVANTAGES

- The range of the Bluetooth module is very small therefore it is confined to a small area.
- Only small seeds can be used for sowing.

6.5 FUTURE WORK

The solar panels can be used to recharge the batteries. Water pumps and pesticides can be mounted on the robot.

CHAPTER 7
REFERENCES

[1] V.Thirumaran , B.Vignesh , V.Vanjinathan “Automation in Seed Sowing by using Smart Agri Robot, International Journal for Research in Applied Science & Engineering Technology (IJRASET), Apr 2019.

[2]Nagashree R N Asst Prof, Dept of TE GSSSIETW Mysuru India “A Multipurpose Agricultural Robot for Automatic Ploughing , Seeding and Plant Health Monitoring. IETE – 2020.

[3].Madhavan, S.Vigneshkumar, M.Dineshkumar, Bannari Amman Institute of Technology “AUTOMATIC CORN SEED SOWING PROCESS BY USING MOBILE ROBOT”. International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), March 2017.

[4]Oladimeji Ayamolowo and Samuel Olushola Dada, Afe Babalola University, Nigeria. “DEVELOPMENT OF AN ARDUINO-BASED OBSTACLE AVOIDANCE ROBOTIC SYSTEM FOR AN UNMANNED VEHICLE”. ARPN Journal of Engineering and Applied Sciences, FEBRUARY 2018.

[5]R.VAIRAVAN[1], S.AJITH KUMAR[2], “Obstacle Avoidance Robotic Vehicle Using Ultrasonic Sensor, Android And Bluetooth For Obstacle Detection”. PSN College of Engineering and Technology Tirunelveli,Feb-2018.

[6] B. Annapurna, B. Anusha, S.V.S Prasad, B. Somanaidu ," Automated Seed Sowing Machine Using Atmega2560". International Journal of Innovative Technology and Exploring Engineering (IJITEE), March, 2019.

[7] Shalini S, Kalaiselvi S, Logeshwari M, Pavithra P, "Implementation of Farm Weeding and Soil Moisturization Techniques using Agribot". International Journal of Engineering and Advanced Technology (IJEAT), February 2020.

[8] 1 M.Aravind kumar,2 Akkarapalli sanjeev reddy, 3 k.sagadevan., "AUTOMATIC SEED SOWING & IRRIGATION AGRIBOT USING ARDUINO**".** International Journal of Pure and Applied Mathematics, 2018

[9] Mrs.R.Thulasimani , "AUTOMATIC SEED SOWING". International Journal of Advanced Science and Engineering Research, 2018

[10] Praveen Barapatre, Jayantilal N. Patel , "Determination of Soil Moisture using Various Sensors forIrrigation Water Management". International Journal of Innovative Technology and Exploring Engineering (IJITEE), May, 2019