EST Practical Activity Report
Submitted for

ENGINEERING DESIGN-II (UTA024)

Submitted by:

Roll Number (Student)	Name of Student
102203274	Muskan Chalana
102203253	Asmi Gaurav
102203252	Kanav Wadhawan
102203965	Jaideep Singh
102203966	Mohit Singh

BE Second Year

BATCH – 2CO6

Submitted to:

Dr. Karamjeet Singh

Computer Science and Engineering Department

TIET, Patiala

Jan-May 2024

DECLARATION

We affirm that the contents of this project report stem from our independent efforts conducted during our Engineering-design II Computer Lab sessions, overseen by **Mr. Karamjeet Singh**. We assert that all statements and conclusions presented herein are the result of our own research endeavors.

Furthermore, we confirm the originality of the work encapsulated in this report, undertaken under the overarching guidance of our supervisor. Adhering to the University's guidelines, we have meticulously crafted this report.

Moreover, we declare that this project is the culmination of our individual contributions and has not been submitted to any other educational institution for degree consideration.

INDEX

S. No.	Name of Experiments	Sign
1.	Introduction to Arduino Micro-Controller.	
2.	Write a program in Arduino to blink a LED. *a.* Single LED *b.* Multiple LEDs	
3.	Write a Arduino program to design and both forward and reverse pattern (1, 2, 3, 4, 5) sequence for different color LEDs using following function: *c.* delay () d. for ()	
4.	Write a program to demonstrate sending data from the computer to the Arduino board and control brightness of LED.	
5.	*Serial Communication:* WAP to print following pattern using for loop. ********** * Roll No._____ ********** Name: _____ ********** Branch: _____ **********	
6.	Write a program to change the intensity of the single LED bulb using: i. digitalRead () ii. analogRead ()	
7.	Write a program to change the intensity of the given LEDs for the sequence 35214 in for both forward and reverse order.	

8.	Write a program to demonstrate control of DC Motor using forward, backward, left, right turn motion and clock-wise/anti clock-wise rotation.	
9.	Write a program to demonstrate the working of the buggy	
10.	Write a program to read values of IR Sensor using analog and digital read and convert buggy into normal line follower robocar.	
11.	To demonstrate the use of ultrasonic sensor by integrating line follower robocar with obstacle avoidance capability.	
12.	Write a program to read the pulse width of gantry transmitter and trigger stop buggy function by detecting individual gantry.	
13.	Write a program to demonstrate Xbee module communication between 2 PCs using X-CTU.	
14.	Bronze Challenge: Single buggy around track twice in clockwise direction, under full supervisory control. Buggy can detect an obstacle, Parks safely. Prints state of the track and buggy at each gantry stop.	

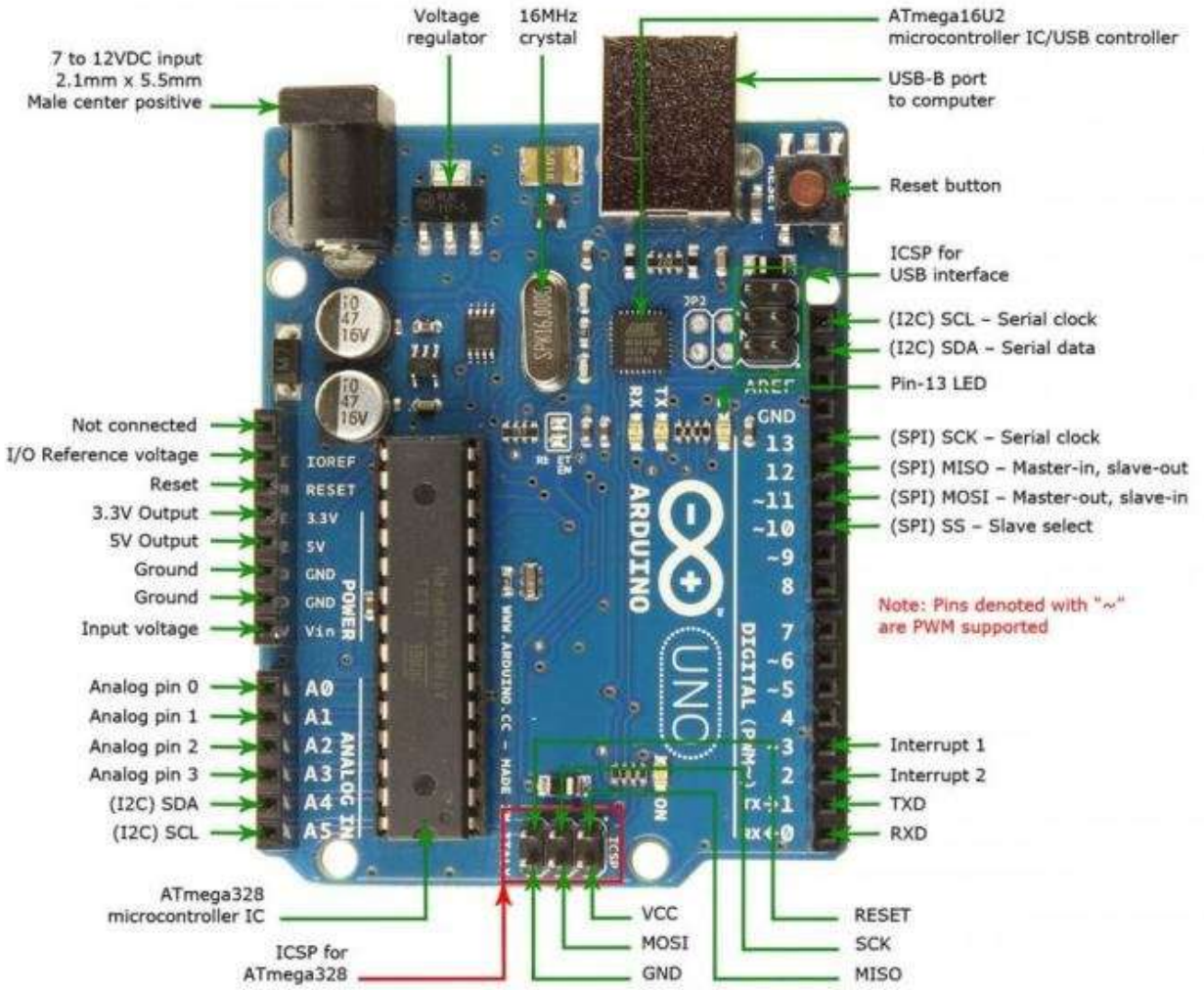
EXPERIMENT-1

OBJECTIVE: Introduction to Arduino Micro-Controller.

HARDWARE USED:

Sr. No	Name of Components	Value
1.	Arduino Uno Micro-Controller	--

FIGURE:



THEORY:

The Arduino Uno is an open-source the Microchip ATmega328P microcontroller and develop d by Arduino.cc and initially released in 2010. The Arduino board is a microcontroller-based development platform that has become immensely popular for hobbyists, students, and professionals alike. It was created to provide an accessible and versatile platform for building interactive projects and prototypes.

1. **USB Power:** Arduino boards can draw power from a PC or laptop via a USB cable, making them convenient for development and testing without the need for external power supplies.
2. **Power Jack:** Alternatively, Arduinos can be powered directly from an AC power supply using a power jack, offering flexibility in deployment and usage scenarios.
3. **Reset Button**: This button allows users to reset the Arduino board, effectively restarting the program execution from the beginning, which can be useful for troubleshooting or reinitializing the system.
4. **Pins:** Arduino boards feature multiple pins used to connect various components such as sensors, actuators, and displays. These pins provide both voltage and ground connections, facilitating communication and power delivery to external devices.
5. **Analog Pins**: Analog pins on Arduino boards are used to read analog signals from sensors, converting them into digital values that can be processed by the microcontroller. This enables the Arduino to interact with a wide range of analog sensors and devices.
6. **Power Pins**: These pins provide power (operating voltage) and ground connections to external components, ensuring proper functioning of connected devices and circuits.
7. **Digital Pins**: Arduino boards offer digital input/output pins that can be configured to either read logic states (0 or 1) or output digital signals. These pins support a variety of functions and can be used for tasks such as controlling LEDs, reading switches, or communicating with other digital devices.
8. **PWM Pins**: Pulse Width Modulation (PWM) pins on Arduino boards allow for the generation of analog-like signals by varying the duty cycle of a square wave signal. This modulation technique is commonly used for tasks such as controlling the brightness of LEDs or the speed of motors.
9. **ATmega328P Microcontroller**: The ATmega328P is a powerful yet energy-efficient 8-bit AVR microcontroller commonly found in Arduino boards like the Arduino Uno. Serving as the central processing unit, it executes program instructions and manages input/output operations, earning it the nickname "brain of Arduino."

RESULT ANALYSIS: In this experiment, we get to know about basics of Arduino Uno Microcontroller and its various functions and components.Overall, this experiment provides a hands-on learning experience in electronics, programming, and microcontroller interfacing, empowering students to apply theoretical concepts in practical projects and explore the capabilities of Arduino-based systems.

EXPERIMENT-2

OBJECTIVE: Write a program in Arduino to blink a LED.

a. Single LED

b. Multiple LEDs

SOFTWARE USED: Arduino IDE

HARDWARE USED:

Sr No.	Name of the Component	Value
1.	Arduino Uno Board	1
2.	Breadboard	1
3.	Jumper Wires	10
4.	LED	1
5.	Resistor	220 ohm

Theory:

1. **Resistor:-** Resistor is a fundamental component in electronics, serving as a two-terminal passive device employed to regulate or restrict the flow of electric current within electrical circuits. It enables precise control over the amount of resistance introduced into a circuit. Among the essential elements within an electronic setup, resistors play a paramount role. Their primary function entails curtailing current flow and diminishing voltage within targeted segments of the circuit. Constructed typically with copper wires wound around a ceramic rod and coated with insulating paint, resistors facilitate the manipulation of electrical flow with precision.

Fig. 2.1 Colour coded carbon resistor

2. **LED:** A light-emitting diode (LED) is a semiconductor-based light emitter that emits light upon the passage of current through it. Within the semiconductor material, electrons recombine with electron holes, resulting in the emission of photons as energy is released. LEDs are characterized by two leads: a positive (Anode) lead and a negative (Cathode) lead. The Anode (+) lead is typically identified by a triangle symbol, while the Cathode (-) lead is denoted by a line. Typically, the longer lead of an LED corresponds to the positive (Anode) terminal, while the shorter lead corresponds to the negative (Cathode) terminal.

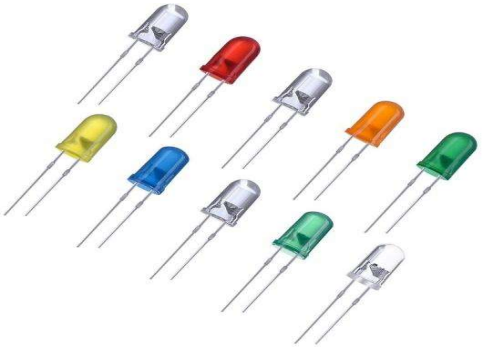


Fig. 2.2 LED

3. **Arduino UNO Board:** The Arduino Uno is a microcontroller board that utilizes the ATmega328P microcontroller (datasheet). It features 14 digital input/output pins, with 6 of them capable of functioning as PWM (Pulse Width Modulation) outputs. Additionally, it offers

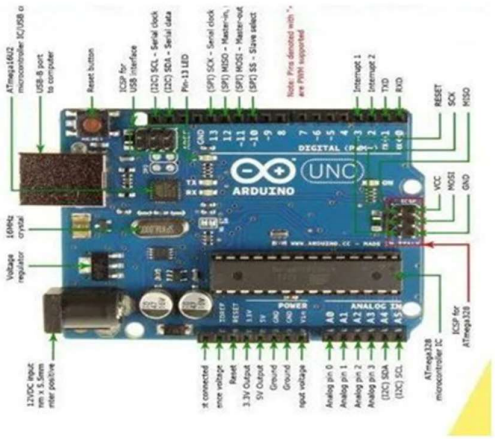


Fig. 2.3 Arduino UNO Board

4. **Breadboard:** A breadboard, also known as a solderless breadboard or protoboard, serves as a foundation for constructing semi-permanent prototypes of electronic circuits. It provides a platform for arranging and connecting components such as resistors, capacitors, LEDs, and other electronic elements in a wired configuration.

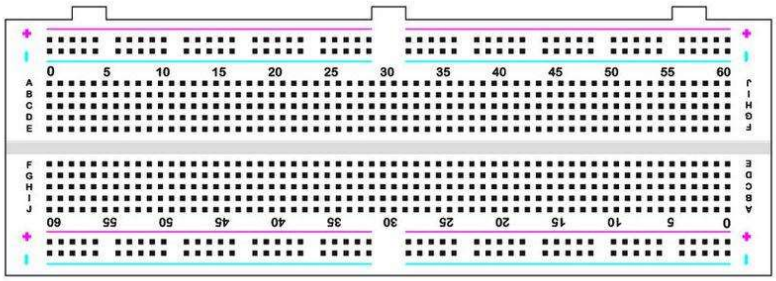


Fig 2.4 Breadboard

5. **Jumper Wires:** A jump wire, also referred to as a jumper or DuPont wire, is an electrical wire or a set of wires within a cable, featuring a connector or pin at each extremity. Typically, it is employed to establish connections between components on a

breadboard, prototype, or testing circuit, either internally or with external equipment or components, eliminating the need for .

Fig. 2.5 Jumper Wires

Circuit Diagram: Using the same code for both SINGLE and MULTIPLE LED blinking.

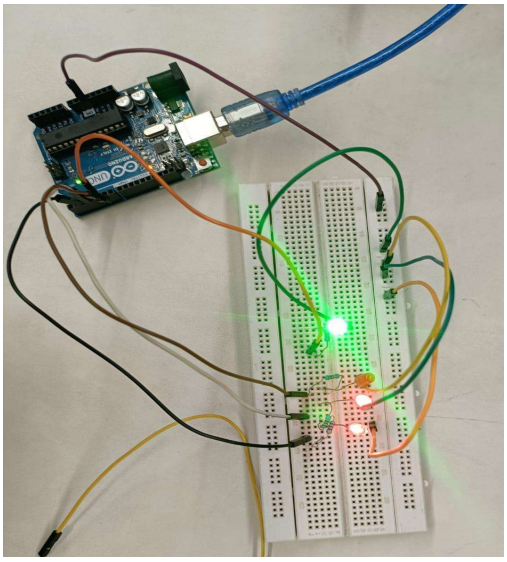


Fig 2.6 MULTIPLE LED in HIGH mode

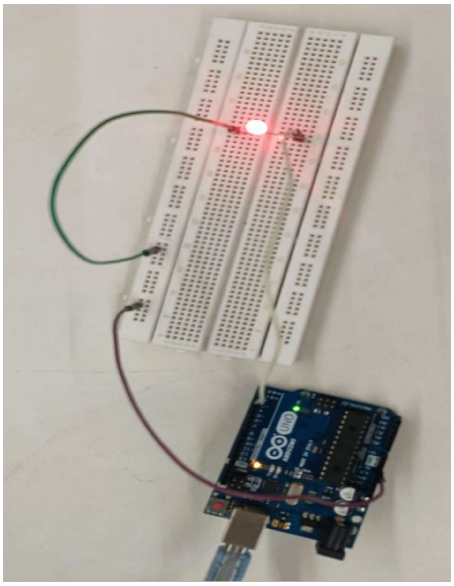


Fig 2.7 SINGLE LED in HIGH mode

CODE:

```
pins ={2,3,4};
void setup()
{
for (int i=0;i<3;i++)
{
   pinMode(pins[i],OUTPUT);
}
}
void loop()
{
  for (int i=0;i<3;i++)
{
   digitalWrite(pins[i],HIGH);
}
delay(1000);
for (int i=0;i<3;i++)
{
   digitalWrite(pins[i],LOW);
}
delay(1000);
}
```

RESULTS:

In this experiment, we learnt how to blink an LED using Arduino Uno. If the circuit is set up correctly, the LED connected to pin 13 of the Arduino should blink on and off . Learners would gain practical experience in connecting components on a breadboard, writing code to control hardware, and observing the effects of their programming on physical device . This experiment lays the foundation for more complex projects involving Arduino and

electronics, providing a hands-on introduction to basic concepts in microcontroller programming and circuit design.

EXPERIMENT-3

OBJECTIVE: Write a Arduino program to design and both forward and reverse pattern (1, 2, 3, 4, 5) sequence for different color LEDs using following function:

c. delay ()

d. for ()

SOFTWARE USED: Arduino IDE

HARDWARE USED:

Sr No.	Name of the Component	Value
1.	Arduino Uno Board	1
2.	Breadboard	1
3.	Jumper Wires	10
4.	LED	4
5.	Resistor	4x220 ohm

Theory:

1) **Resistor:** The term "resistor" refers to a device that acts as a two-terminal passive electrical component that is used to limit or regulate the flow of electric current in electrical circuits. And it also allows us to introduce a controlled amount of resistance into an electrical circuit. The most important and commonly used components in an electronic circuit are resistors. A resistor's main job is to reduce current flow and lower voltage in a specific section of the circuit. It's made up of copper wires that are wrapped around a ceramic rod and coated with insulating paint.

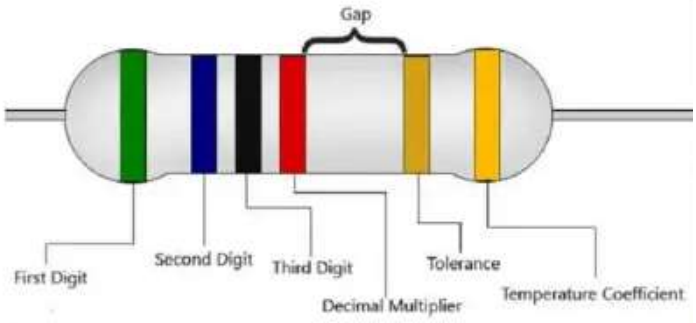


Fig. 3.1 Colour coded carbon resistor

2) **LED:** A light-emitting diode is a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. A LED has a positive (Anode) lead and a negative (Cathode) lead. The Anode (+) is marked with a triangle, and the Cathode (-) is marked with a line. The longer lead of an LED is generally the positive (Anode), while the shorter lead is the negative (cathode).

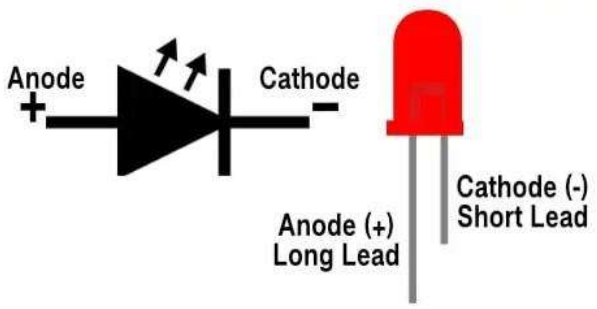


Fig. 3.2 LED

3) **Arduino UNO Board:** Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button.

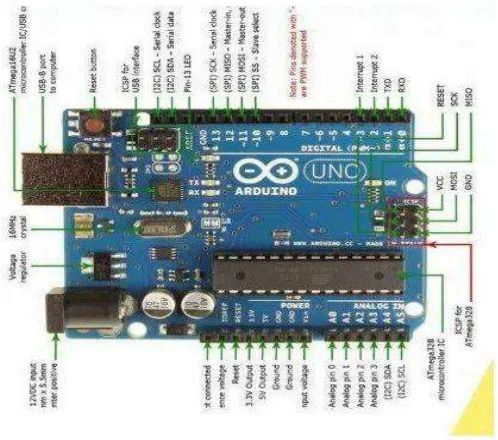


Fig. 3.3 Arduino UNO Board

4) **Breadboard:** A breadboard, solderless breadboard, or protoboard is a construction base used to build semi-permanent prototypes of electronic circuits. A breadboard is used to place components (resistor, capacitor, LED's etc.) that are wired together.

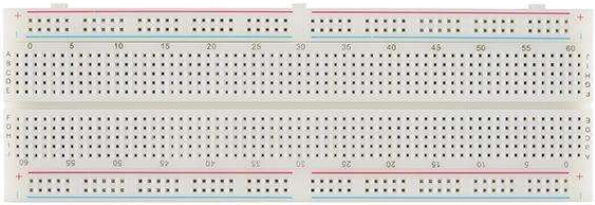


Fig. 3.4 Breadboard

5) **Jumper Wires:** A jump wire (also known as jumper, jumper wire, DuPont wire) is an electrical wire, or group of them in a cable, with a connector or pin at each end , 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.

Fig. 3.5 Jumper Wires

Circuit Diagram:

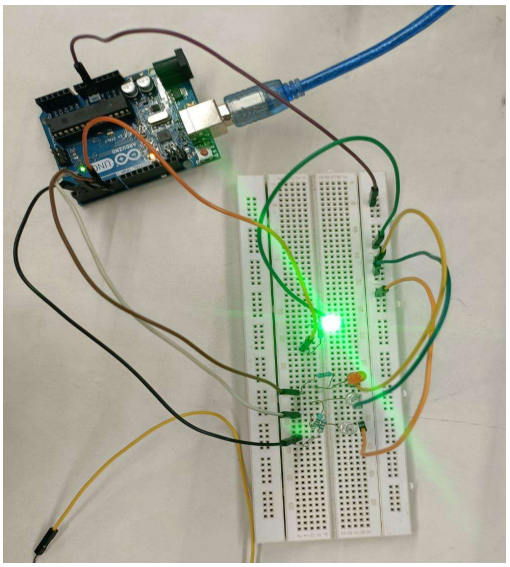


Fig. 3.6 Representing

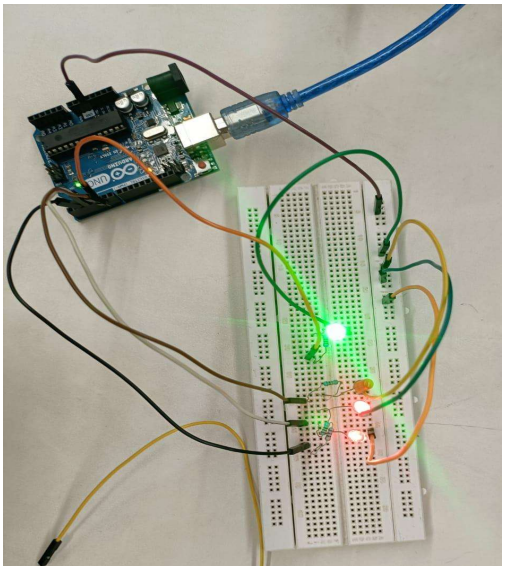


Fig. 3.7 Representing

CODE:

```
// Define LED pins
const int ledPins[] = {3, 6, 9, 10, 11};

void setup() {
  // Set LED pins as output
  for (int i = 0; i < 5; i++) {
    pinMode(ledPins[i], OUTPUT);
  }
}

void loop() {
  for (int i = 0; i < 5; i++) {
```

```
   digitalWrite(ledPins[i], HIGH);
   delay(500);
   digitalWrite(ledPins[i], LOW);
  }

  for (int i = 4; i >= 0; i--) {
   digitalWrite(ledPins[i], HIGH);
   delay(500);
   digitalWrite(ledPins[i], LOW);
  }
}
```

RESULTS:

In this experiment, we learnt how to generate a pattern of blinking of multiple LEDs using Arduino Uno.

Signature of Faculty member

EXPERIMENT-4

2. **OBJECTIVE:** Serial Communications:

- Serial.begin(9600);
- Serial.print();
- Serial.println()
- Serial.read()
- Serial.write()

SOFTWARE USED: Arduino IDE

HARDWARE USED:

Sr No.	Name of the Component	Value
1.	Arduino Uno Board	1
2.	Serial Monitor	1

Theory:

1) **Serial Monitor:** The serial monitor is the 'tether' between the computer and the Arduino - it lets us send and receive text messages, handy for debugging and also controlling the Arduino from a keyboard. Serial Data is sent over a single wire (usually travels over USB) and consists of a series of 1's and 0's sent over the wire. Data can be sent in both directions

2) **Arduino UNO Board:** Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button.

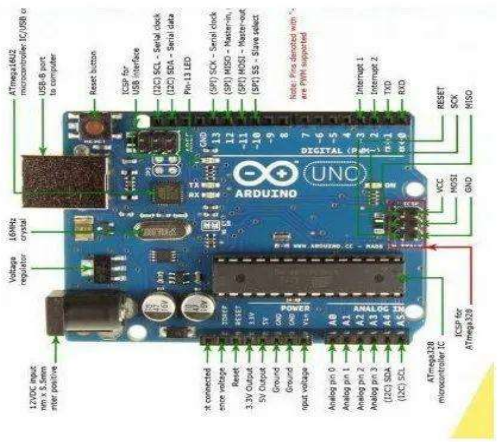


Fig. 4.3 Arduino UNO Board

3) **Serial.begin()** is a function in Arduino used to initialize serial communication. It configures the data rate (in bits per second, or baud rate) for serial data transmission. This function must be called before any communication occurs via the serial port.

4) **Serial.print()** is a function in Arduino used to send data over the serial port. It sends data as human-readable ASCII text.data can be of various types: integer, float, character, string, etc.

5) **Serial.println()** is a function in Arduino used to send data over the serial port, similar to Serial.print(). However, Serial.println() adds a newline character ("\n") to the end of the data being sent. This causes the cursor in the serial monitor to move to the beginning of the next line after printing the data.

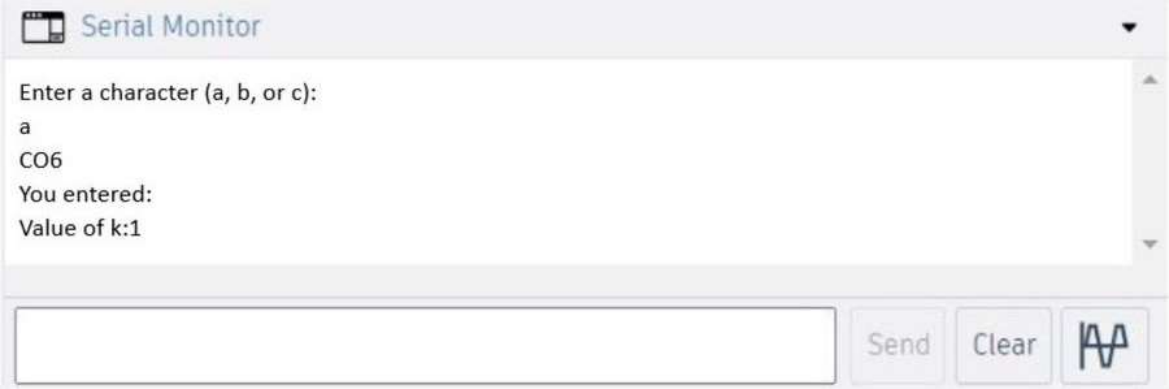
6) **Serial.read()** is a function in Arduino used to read data from the serial port. It reads incoming serial data byte by byte.incomingByte is an integer variable that holds the byte of data read from the serial port. The returned value is in the range of 0 to 255.

7) **Serial.available()** is a function in Arduino used to check whether there are any bytes of data available to be read from the serial input buffer. It returns the number of bytes available for reading.

8) **Serial.write()** is a function in Arduino used to send binary data over the serial port. It sends raw bytes of data without any formatting or interpretation.Serial.write() is useful when you need to send binary data or non-textual data, such as raw sensor readings or binary commands, over the serial port.

9) **Serial.parseInt()** is a function in Arduino used to read an integer value from the serial input buffer. It reads characters from the serial input buffer until it encounters a non-numeric character, and then converts the collected characters into an integer value.

OUTPUT:



CODE:

```
void setup() {
  Serial.begin(9600);
}

void loop() {
if(Serial.available()>0){
  Serial.println("Enter a character (a, b, or c):");
  char inputChar = Serial.read();
  Serial.println(inputChar);
  Serial.write("CO6\n");
  Serial.print("You entered: ");

  int k = Serial.parseInt();
  Serial.print("Value of k:\n");
  Serial.println(k);
```

```
  delay(1000);
}
}
```

RESULTS:

With the help of the above program we get to know about how the serial communication takes place and how the serial monitor is responsible for the same.

Signature of Faculty member

EXPERIMENT-5

OBJECTIVE: Serial Communication:

WAP to print following pattern using for loop:

****
Roll No._____
****
Name: _____
****
Branch: _____
**********.

SOFTWARE USED: Arduino IDE

HARDWARE USED:

Sr No.	Name of the Component	Value
1.	Arduino Uno Board	1

Theory:

10) **Arduino UNO Board:** Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button.

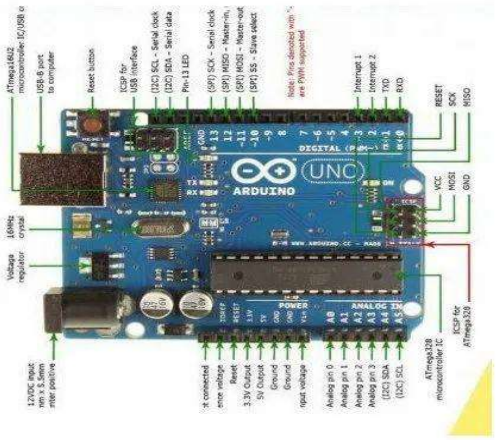


Fig. 5.1 Arduino UNO Board

11) **Serial Monitor:** The serial monitor is the 'tether' between the computer and the Arduino - it lets us send and receive text messages, handy for debugging and also controlling the Arduino from a keyboard. Serial Data is sent over a single wire

(usually travels over USB) and consists of a series of 1's and 0's sent over the wire. Data can be sent in both directions

CODE:

```
void setup()
{   Serial.begin(9600);
}
void loop()
{
  int i=0;
  for(int i=0;i<42;i++)
  {
    Serial.print('*');
  }
  Serial.println("\n Roll No. 102203966");
  for(int i=0;i<28;i++)
  {
  Serial.print('*');
  }
  Serial.println("\n Name: Mohit Singh");
  for(int i=0;i<38;i++)
  {  Serial.print('*');
}
  Serial.println("\n Branch: COE");
```

```
 for(int i=0;i<28;i++)
 {
 Serial.print('*');
 }
 for(int i=0;i<40;i++)
 {
 Serial.print('*');
 }
 for(int i=0;i<28;i++)
 {
 Serial.print('*');
 }
 exit(0);
}
```

OUTPUT:

RESULTS: In this experiment, we learnt how to input data from the user and print the data on the monitor screen using Serial Monitor.

Signature of Faculty member

EXPERIMENT-6

OBJECTIVE: Write a program to change the intensity of the single LED bulb using:

1. digitalRead ()
2. analogRead ()

SOFTWARE USED: Arduino IDE

HARDWARE USED:

Sr No.	Name of the Component	Value
1.	Arduino Uno Board	1
2.	Breadboard	1
3.	Jumper Wires	4
4.	LED	1
5.	Resistor	1x220 ohm

Theory:

1) **Resistor:** The term "resistor" refers to a device that acts as a two-terminal passive electrical component that is used to limit or regulate the flow of electric current in electrical circuits. And it also allows us to introduce a controlled amount of resistance into an electrical circuit. The most important and commonly used components in an electronic circuit are resistors. A resistor's main job is to reduce current flow and lower voltage in a specific section of the circuit. It's made up of copper wires that are wrapped around a ceramic rod and coated with insulating paint.

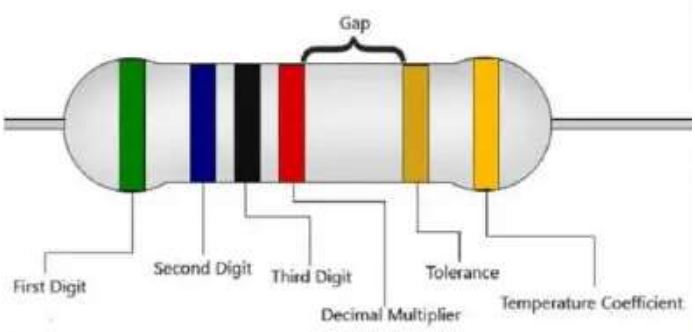


Fig. 6.1 Colour coded carbon resistor

2) **LED:** A light-emitting diode is a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. A LED has a positive (Anode) lead and a negative (Cathode) lead. The Anode (+) is marked with a triangle, and the Cathode (-) is marked with a line. The longer lead of an LED is generally the positive (Anode), while the shorter lead is the negative (cathode).

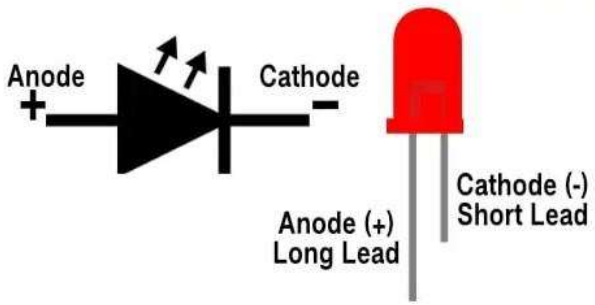


Fig. 6.2 LED

3) **Arduino UNO Board:** Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button.

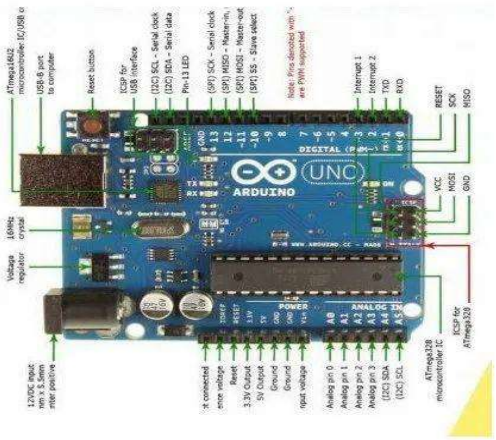


Fig. 6.3 Arduino UNO Board

4) **Breadboard:** A breadboard, solderless breadboard, or protoboard is a construction base used to build semi-permanent prototypes of electronic circuits. A breadboard is used to place components (resistor, capacitor, LED's etc.) that are wired together.

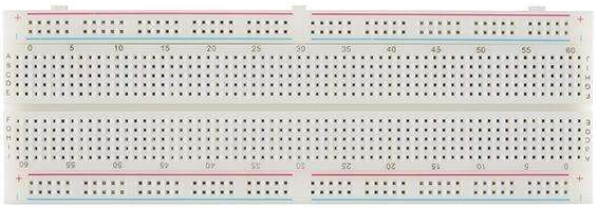


Fig. 6.4 Breadboard

5) **Jumper Wires:** A jump wire (also known as jumper, jumper wire, DuPont wire) is an electrical wire, or group of them in a cable, with a connector or pin at each end , 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.

Fig. 6.5 Jumper Wires

Circuit Diagram:

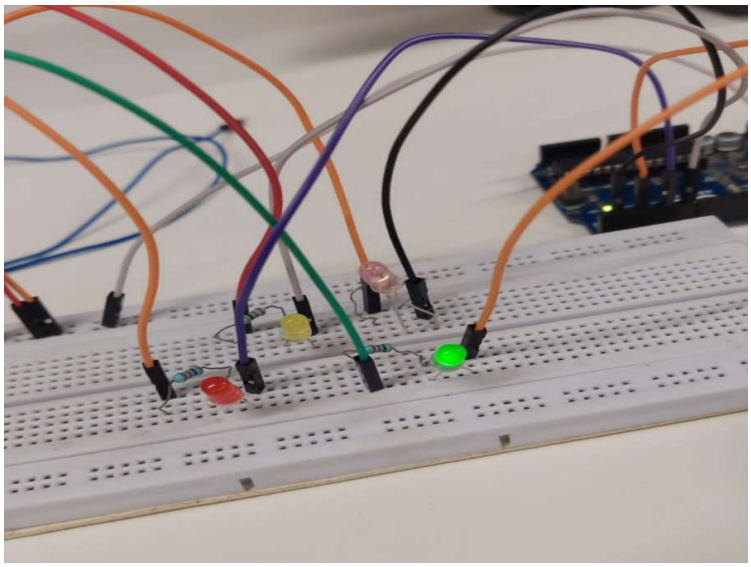


Fig. 6.6 Controlling the brightness of LED using digitalRead() and analogRead()

CODE:

```
int led[]= (1, 3, 5, 7);

void setup() {
```

```
for(int i=0; i<5; i++)
pinMode(led[i], OUTPUT);
   }
}
void loop() {
for(int i=0 i<4; i++) {
for (int br=0 ; br<=255 ; br=br+5) {
analogWrite(led[i], br);
delay(100);
}
delay(1000);
for(int br=255; br>=0; br=br-5 {
analogWrite(led[i], br);
delay(100);
}
delay(1000);
  }
}
```

RESULT:

In this experiment, we learnt how to control the brightness of LED using both analogRead() and digitalRead ().

Signature of Faculty member

EXPERIMENT-7

OBJECTIVE: Write a program to change the intensity of the given LED's for the sequence 35214 in for both forward and reverse order.

SOFTWARE USED: Arduino IDE

HARDWARE USED:

Sr No.	Name of the Component	Value
1.	Arduino Uno Board	1
2.	Breadboard	1
3.	Jumper Wires	10
4.	LED	5
5.	Resistor	5x220 ohm

Theory:

1) **Resistor:** The term "resistor" refers to a device that acts as a two-terminal passive electrical component that is used to limit or regulate the flow of electric current in electrical circuits. And it also allows us to introduce a controlled amount of resistance into an electrical circuit. The most important and commonly used components in an electronic circuit are resistors. A resistor's main job is to reduce current flow and lower voltage in a specific section of the circuit. It's made up of copper wires that are wrapped around a ceramic rod and coated with insulating paint.

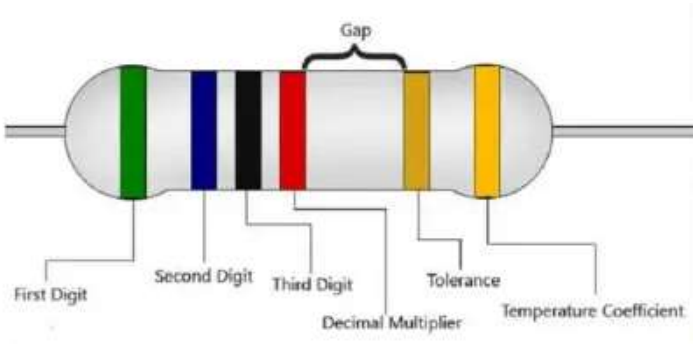


Fig. 7.1 Colour coded carbon resistor

2) **LED:** A light-emitting diode is a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. A LED has a positive (Anode) lead and a negative (Cathode) lead. The Anode (+) is marked with a triangle, and the Cathode (-) is marked with a line. The longer lead of an LED is generally the positive (Anode), while the shorter lead is the negative (cathode).

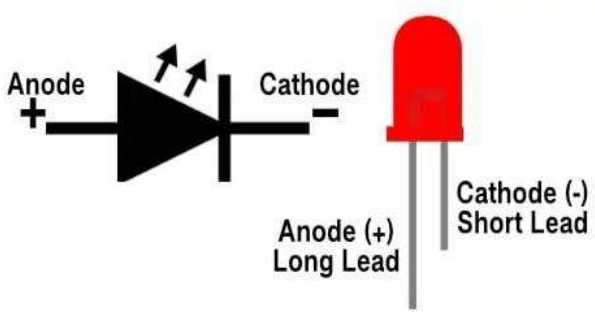


Fig. 7.2 LED

3) **<u>Arduino UNO Board:</u>** Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button.

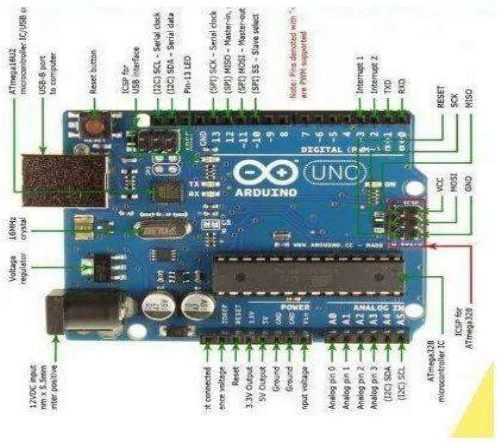


Fig. 7.3 Arduino UNO Board

4) **<u>Breadboard:</u>** A breadboard, solderless breadboard, or protoboard is a construction base used to build semi-permanent prototypes of electronic circuits. A breadboard is used to place components (resistor, capacitor, LED's etc.) that are wired together.

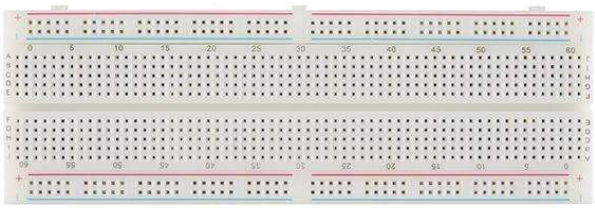


Fig. 7.4 Breadboard

5) **Jumper Wires:** A jump wire (also known as jumper, jumper wire, DuPont wire) is an electrical wire, or group of them in a cable, with a connector or pin at each end , 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.

Fig. 7.5 Jumper Wires

Circuit Diagram:

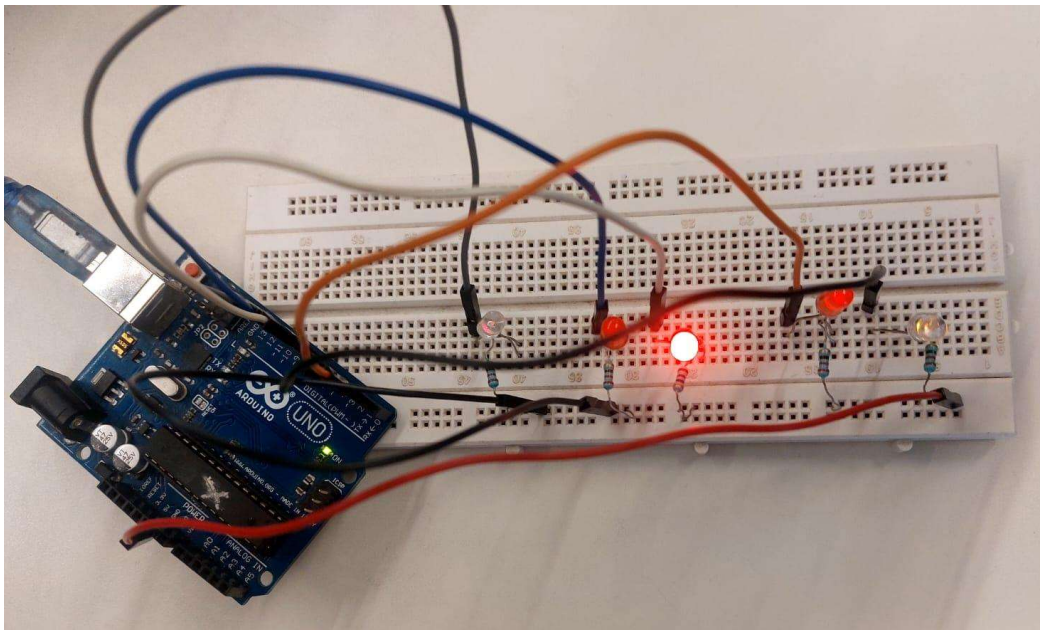


Fig. 7.6 3rd LED glowing for the sequence (35214)

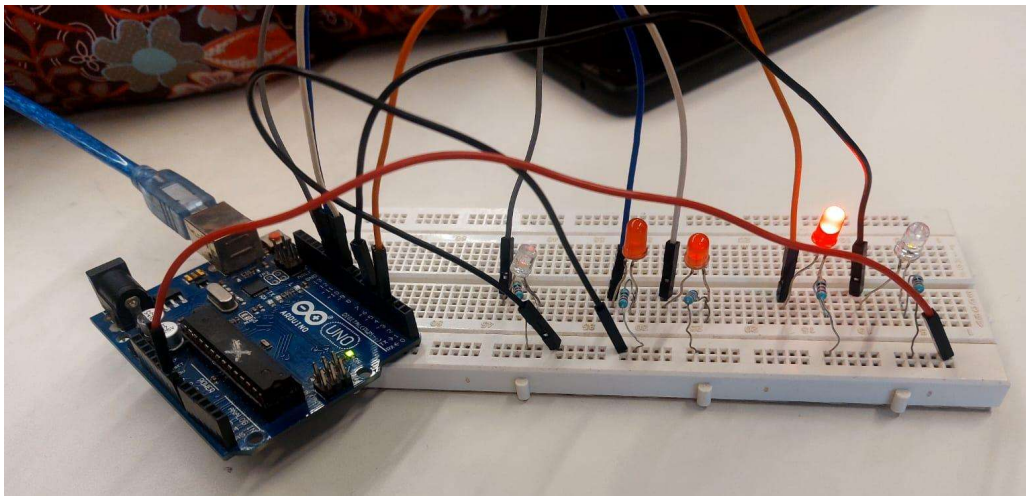


Fig. 7.6 3rd LED glowing for the sequence (35214)

CODE:

```
pins[]={3,5,9,10,11};
void setup() {
for(int i=0;i<5;i++){

 pinMode(pins[i],OUTPUT);
}
 void loop() {
 for(int i=0;i<=255;i+=10)
 {
  analogWrite(9,i);
  delay(100);
 }
 for(int i=255;i>0;i-=10)
 {
  analogWrite(9,i);
```

```
 delay(100);
}
 for(int i=0;i<=255;i+=10)
{
 analogWrite(11,i);
 delay(100);
}
for(int i=255;i>0;i-=10)
{
 analogWrite(11,i);
 delay(100);
} for(int i=0;i<=255;i+=10)
 {
   analogWrite(3,i);
   delay(100);
 }
 for(int i=255;i>0;i-=10)
 {
   analogWrite(3,i);
   delay(100);
 }  for(int i=0;i<=255;i+=10)
 {
   analogWrite(5,i);
   delay(100);
 }
 for(int i=255;i>0;i-=10)
```

```
  {
    analogWrite(5,i);
    delay(100);
  }  for(int i=0;i<=255;i+=10)
  {
    analogWrite(10,i);
    delay(100);
  }
  for(int i=255;i>0;i--)
  {
    analogWrite(10,i);
    delay(5);
  }
}
```

RESULTS:

In this experiment, we learnt how to control brightness of multiple LED's using analogRead() and analogWrite() and Serial Monitor.

Signature of Faculty membe

EXPERIMENT-9

OBJECTIVE:

Write a program to illustrate the working of the buggy by making the different pattern by your own.

SOFTWARE USED: Arduino IDE

HARDWARE USED:

Sr No.	Name of the Component	Value
1.	Arduino Uno Board	1
2.	Buggy	1

Theory:

1. **Arduino UNO Board:** Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button.

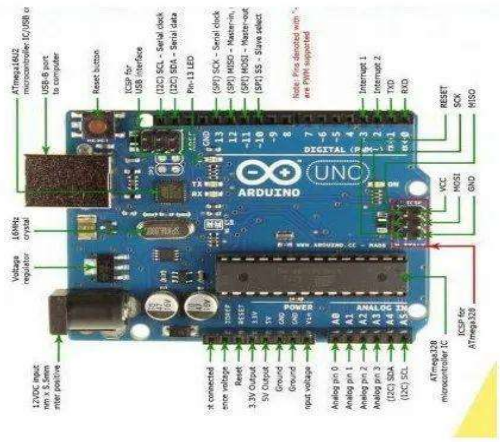


Fig. 8.1 Arduino UNO Board

2. **Buggy:** DC (Direct Current) motor converts electrical energy into mechanical energy.A motor driver is an electronic device that controls the speed and direction of a motor. It allows bidirectional control (forward and backward motion) and speed control using PWM Arduino is an open-source electronics platform based on easy-to-use hardware and software. It consistof a microcontroller board (such as the ATmega328P) and a development environment

for writing and uploading code. The motor driver (L298N) typically has input pins for controlling each motor's direction (IN1, IN2, IN3, IN4) and enable pins (ENA, ENB) for PWM-based speed control. Functions like forward(), backward(), left(), right(), rotate Clockwise(), and rotate AntiClockwise() are implemented to control the motors' direction and rotation.

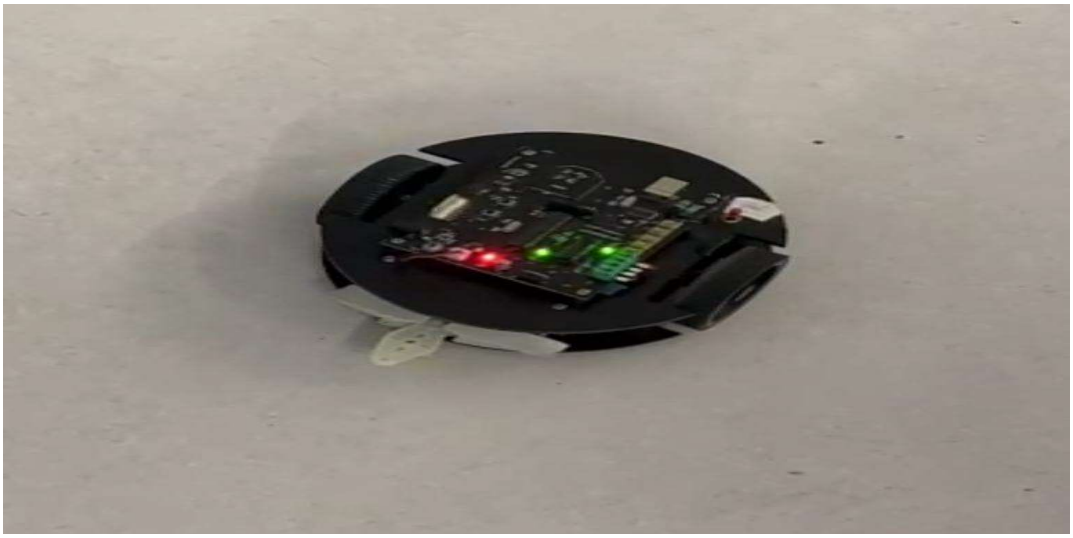


Fig. 8.2 Buggy

CODE:

```
const int pins[] = {5,6,7,8};

void setup() {

 for(int i=0;i<4;i++){

   pinMode(pins[i],OUTPUT);

 }

}

//forward movement of buggy

void forward(){

 digitalWrite(5,HIGH);

 digitalWrite(6,LOW);
```

```
 digitalWrite(7,LOW);
 digitalWrite(8,HIGH);
}
//backward movement of buggy
void backward(){
 digitalWrite(5,LOW);
 digitalWrite(6,HIGH);
 digitalWrite(7,HIGH);
 digitalWrite(8,LOW);
}
// left movement of buggy
void left(){
 digitalWrite(5,HIGH);
 digitalWrite(6,LOW);
 digitalWrite(7,HIGH);
 digitalWrite(8,LOW);
}
// right movement of buggy
 void right(){
 digitalWrite(5,LOW);
 digitalWrite(6,HIGH);
 digitalWrite(7,LOW);
 digitalWrite(8,HIGH);
}
```

```
//clockwise movement of buggy
void clockwise()
{
right();
delay(80);
forward();
delay(80);
}
for (int i=0; i<100;i++)
{
clockwise();
}
//anticlockwise movement of buggy
void anticlockwise()
{
left();
delay(80);
forward();
delay(80);
}
for (int i=0; i<100;i++)
{
anticlockwise();
}
```

```
void loop(){
    //1. plus pattern
    forward();
    backward();
    backward();
    forward();
    left();
    forward();
   backward();
    right();
    right();
    forward();
    backward();
    //2. heart pattern
    forward();
    delay(2000);
    for(int j=0;j<60;j++){
        right();
        delay(50);
       forward();
        delay(50);
        j++;
    }
```

```
    left();
    delay(900);
    for(int j=0;j<65;j++){
        right();
        delay(50);
        forward();
        delay(50);
        j++;
    }
    forward();
   delay(2000);
}
```

RESULT:

In this experiment, we learnt how to run the buggy and make the different pattern for the same with the help of the code we write.

Signature of Faculty member