Higher institute for engineering and technology – kafrElshiekh

Shaft encoder

By:techno team

Dr.osama tharwat

1. Introduction

A shaft encoder, also known as a rotary encoder or simply encoder, is a device that converts the angular position or motion of a shaft into an electrical signal. It provides a way to measure the rotational movement of a shaft or wheel and is commonly used in various applications, including robotics, industrial machinery, and automation systems.

There are different types of shaft encoders, but they generally fall into two main categories: incremental and absolute.

Incremental Encoder:

Provides information about the relative motion or position change.
Typically generates a series of pulses as the shaft rotates.
The output is in the form of pulse counts, and the controller needs a reference point (often called a home position) to determine the absolute position.

Absolute Encoder:

Provides information about the absolute position of the shaft at any given moment.
Each position of the shaft corresponds to a unique code or set of electrical signals.
Does not require a reference point and can provide the exact position even after power-off.
The basic components of a shaft encoder include a rotating disk or code wheel attached to the shaft, and sensors or detectors that read the markings or patterns on the disk. The sensors convert the

mechanical motion into electrical signals that can be interpreted by a controller or a computer.

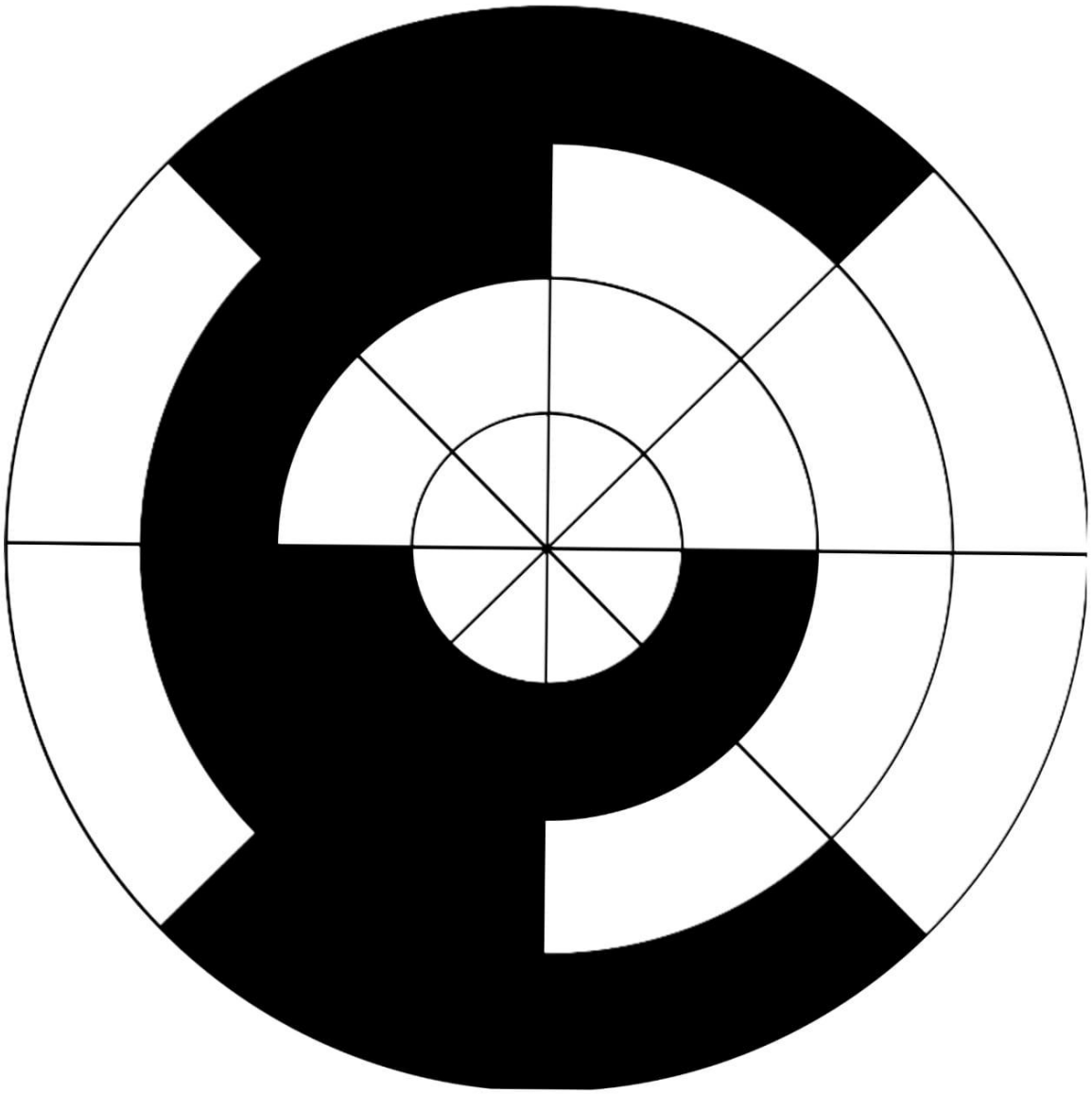
Shaft encoders find application in a wide range of systems where precise position or motion feedback is crucial, such as in CNC machines, robotic arms, motor control systems, and various industrial automation processes. They play a vital role in providing accurate and real-time feedback for controlling and monitoring rotational movements.

2. Project:

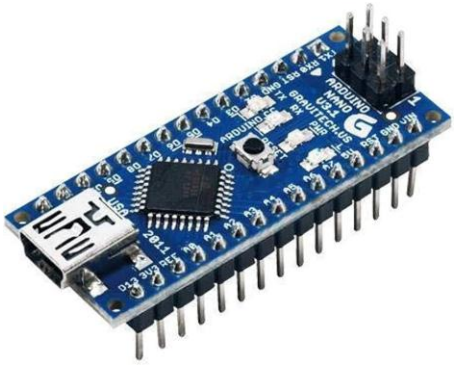
The main idea of the project is the existence of a periodic loop divided into eight components, each part is divided into three components so that the part is either hollow or solid.

After that, the gray code of this disk is generated so that the empty part is 1 and the solid part is 0.

After that, the number of letters and the speed of the cycle are calculated through gray code and processed by Arduino.



3. Component



Arduino nano

Motor DC

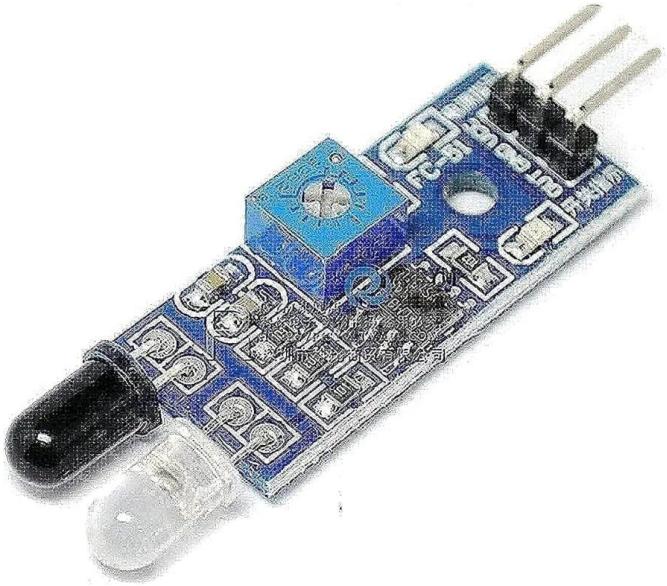
Battery 9V

Wires

Bread board

resistor

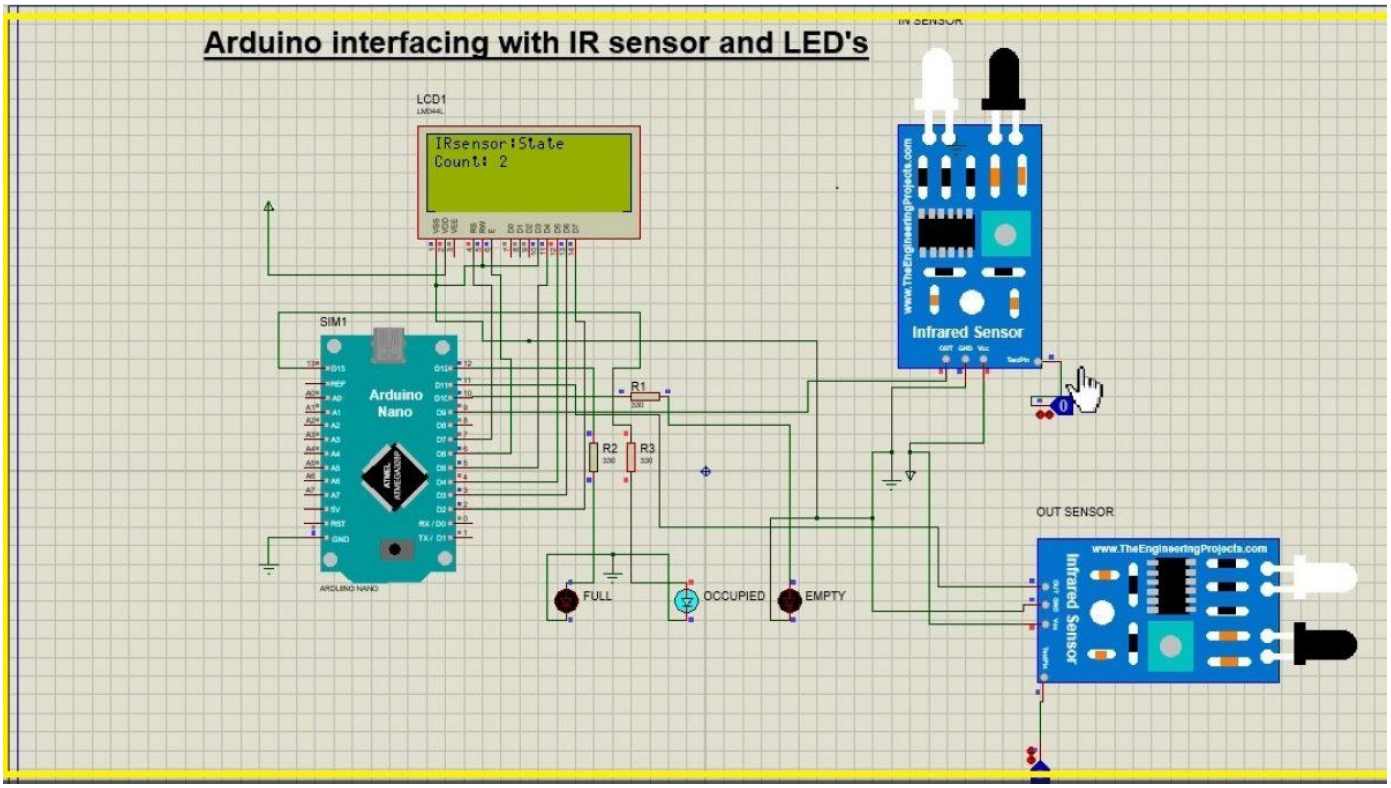
Fixed resistor



IR sensor

LCD

4. Connecting the circuit



5. Code

```
int IRpin1 = 6;
 int ledPin1 = 5;
 int IRpin2 = 4;
 int ledPin2 = 3;
 int IRpin3 = 2;
 int ledPin3 = 13;
 byte startReadingValue1;
 byte startReadingValue2;
 byte startReadingValue3;
bool readingsTaken = false;
byte combinedStartReadingValues;
double fractionOfNumber = .125;
int numberOfTurns = 0;
int swap1;
int swap2;
int swap3;
const float pi = 3.14;
const float diameter = 11.5;
```

```
unsigned long lastTime;
int cycle = 0;
float Speed = 0.0;

#include <LiquidCrystal.h>
LiquidCrystal lcd(7, 8, 9, 10, 11, 12);
void setup() {
  pinMode(IRpin1, INPUT);
  pinMode(ledPin1, OUTPUT);
  pinMode(IRpin2, INPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(IRpin3, INPUT);
  pinMode(ledPin3, OUTPUT);
  Serial.begin(9600);
  lcd.begin(16, 2);
    startReadingValue1 = digitalRead(IRpin1);
  if (!readingsTaken) {
    startReadingValue1 = digitalRead(IRpin1);
    startReadingValue2 = digitalRead(IRpin2);
    startReadingValue3 = digitalRead(IRpin3);
    readingsTaken = true;
  }



  }

  //combinedStartReadingValues = startReadingValue1 + startReadingValue2 +
startReadingValue3;

void loop(){
int   readValue1 = digitalRead(IRpin1);
   int readValue2 = digitalRead(IRpin2);
   int readValue3 = digitalRead(IRpin3);

    lcd.setCursor(0, 1);
        lcd.print("speed:");
    if (readValue1 == 0) {
      digitalWrite(ledPin1, LOW);
    }
    else if (readValue1 == 1)
    {
      digitalWrite(ledPin1, HIGH);
    }
     if (readValue2 == 0) {
      digitalWrite(ledPin2, LOW);
    }
    else if (readValue2 == 1) {
      digitalWrite(ledPin2, HIGH);
    }
```

```
      if (readValue3 == 0) {
       digitalWrite(ledPin3, LOW);
     }
     else  if (readValue3 == 1){
       digitalWrite(ledPin3, HIGH);
     }

    if (readValue1 != startReadingValue1 && readValue2 != startReadingValue2 &&
  readValue3 != startReadingValue3) {
       fractionOfNumber = fractionOfNumber + 0.125;
     startReadingValue1 = swap1;
     swap1 = readValue1;
     startReadingValue2 = swap2;
     swap2 = readValue2;
     startReadingValue3 = swap3;
     swap3 = readValue3;
     if (fmod(fractionOfNumber , 1) == 0) {
   numberOfTurns++;
   lcd.setCursor(0, 0);
     lcd.print("NUM: ");
     lcd.setCursor(4, 0);
   lcd.print(numberOfTurns*2);
   //lastTime = millis();
    } }
    if (numberOfTurns > 0) {
     unsigned long currentTime = millis();
     Speed = numberOfTurns * pi * diameter / (currentTime / 1000.0);
         lcd.setCursor(7, 1);
         lcd.print(Speed);
         lcd.print(" m/s");

    }
}
```

6. conclusion

At the end of this project, the output can be produced from the gray code and the number of turns and rotation speed can be calculated.

والله ولى التوفيق.