TABLE OF CONTENTS

LIST OF FIGURE

ABSTRACT

The induction motor stands as the most extensively utilized motor in industry, yet its faults can cause breakdowns and significant expenses. This project proposes a cost-effective system to address these issues. Through comprehensive discussions, a solution is presented that offers affordability without compromising effectiveness. By leveraging innovative approaches, the system ensures optimal performance while minimizing costs, making it a viable choice for industries seeking reliable motor fault detection and mitigation.

Emphasizing economic feasibility, the project explores methods to enhance fault detection accuracy while keeping expenses low, thus providing a practical solution to the challenges associated with induction motor maintenance.

Overall, this project contributes to the advancement of industrial practices by offering a cost-effective system tailored to address the critical issues related to induction motor faults, ultimately benefiting industries through improved operational efficiency and reduced expenses.

CHAPTER 1

INTRODUCTION

The induction motor serves as an indispensable component across various industrial sectors due to its reliability, efficiency, and versatility. Its simple construction, coupled with robust performance, makes it the motor of choice for a wide range of applications, from powering conveyor belts in manufacturing plants to driving pumps in water treatment facilities. However, despite its widespread use, induction motors are not immune to faults and failures. Problems such as bearing wear, winding insulation degradation, and rotor imbalance can lead to unexpected downtime, production losses, and increased maintenance costs for industries. Addressing these issues promptly and effectively is crucial to ensure uninterrupted operation and maximize productivity. Therefore, there is a pressing need to develop cost-effective solutions that can detect and mitigate faults in induction motors, minimizing downtime and optimizing operational efficiency for industries reliant on these vital machines.

The Induction Motor Protection System project aims to enhance the reliability and longevity of induction motors by implementing a comprehensive monitoring and control system. By integrating sensors to detect parameters such as temperature, vibration, and RPM, the project ensures real-time monitoring of the motor's operating conditions. In the event of any anomalies or factors indicating potential faults beyond acceptable limits, the system initiates automatic shutdown using a relay mechanism, preventing further damage and mitigating risks of breakdowns. This proactive approach to motor protection not only minimizes downtime and maintenance costs but also enhances overall operational efficiency by ensuring uninterrupted performance of critical industrial processes.

As we read through this report, we'll learn more about the "Induction motor protection" project and how it's useful in many industries as efficiently as possible.

1.1 Existing system

The existing system for induction motor protection commonly incorporates overload relays as a primary safeguard against excessive current and short circuits. These relays are installed in series with the motor circuit and are designed to detect abnormal current levels. When the current surpasses predetermined thresholds, indicative of potential faults such as overloading or short circuits, the overload relays trigger, interrupting power to the motor and preventing further damage. While effective in basic protection against overcurrent conditions, this existing system lacks the sophistication of modern technologies and does not offer real-time monitoring or advanced fault detection capabilities.

1.2 Proposed methodology

The proposed Induction Motor Protection system using Arduino, aims to monitor critical motor parameters like temperature, vibration, RPM, current, and input voltage. It comprises IR, vibration, temperature, current, and voltage sensors, alongside a motor driver, a 12V DC motor, and an Arduino Uno. The temperature sensor tracks motor temperature, the vibration sensor measures motor vibration, and the IR sensor detects motor RPM. Meanwhile, the current and voltage sensors gauge amps and input voltage, respectively. These sensors relay data to the Arduino Uno for analysis. Based on predefined thresholds, the Arduino Uno communicates with the motor driver to either activate or deactivate the motor. If the sensor data indicates safe motor operation, the motor runs as usual; otherwise, the motor is shut down. This system offers efficient induction motor monitoring, ensuring prolonged motor lifespan and aiding various industries.

1.3 Objective

The project aims to achieve the following:

1. Monitor Critical Motor Parameters: The system aims to continuously monitor vital parameters including temperature, vibration, RPM, current, and input voltage of the induction motor to ensure optimal performance and prevent potential faults.
2. Utilize Multiple Sensors: By integrating IR, vibration, temperature, current, and voltage sensors, the system seeks to accurately measure and track various aspects of motor operation, providing comprehensive data for analysis.
3. Enable Real-Time Data Analysis: Data collected from the sensors is processed by the Arduino Uno in real-time, allowing for immediate analysis of motor conditions and timely response to any deviations from normal operating parameters.
4. Implement Threshold-Based Control: The system defines predefined thresholds for each parameter, enabling the Arduino Uno to make informed decisions regarding motor activation or deactivation based on the sensor readings.
5. Ensure Motor Safety and Longevity: By promptly shutting down the motor in case of abnormal conditions, the system aims to prevent damage to the motor, extend its lifespan, and minimize the risk of costly downtime.
6. Support Various Industries: With its ability to provide efficient motor monitoring and protection, the system is designed to cater to the needs of diverse industries relying on induction motors, thereby enhancing operational reliability and productivity.

CHAPTER 2

LITERATURE REVIEW

Title: "Hardware Design of Network Model of Protection and Controller module for Three-Phase Induction Motors in Industrial Plants with Remote Monitoring System on a Centralized System"

Author: Rahul Santhosh, Vikram Siddharthan M, Sailakshmi and Sudha Yadav

|**Description:** Manufacturing Industries uses various equipments to perform its tasks, these instruments are controlled simultaneously at a centralized monitoring station. The instrument sends the task progress and measured values from various sensors to the central monitoring system via a local area network (LAN). Operational instructions are sent from the system to the instruments. This paper explains the design, development and troubleshooting of hardware design of network model of fault prevention and remote control of three-phase induction motors in industrial plant and improvements in module design. The system checks faults in 3 phase induction motor such as incorrect phase sequence, overload current, no-load current, surge voltages, imbalance phase loading & single phasing. Measured electrical parameters, fault conditions and motor performance status are sent to a central monitoring system via TCP/IP protocol in a local area network. Multiple modules can be monitored and operated remotely with a user interface at central monitoring system. Centralised monitoring system enables the user to control, check the performance parameters and record the operation from a remote station.

Title: "Fault Protection Technique for ZSI-fed Single-Phase Induction Motor Drive System"

Author: Vivek Sharma, M. J. Hossain and S. M. Nawazish Ali

Description: The Z-source inverter (ZSI) is getting popular for induction motor drives as it provides single-stage power conversion with voltage buck-boost ability. Recent surveys have reported that inverters are prone to failure due to internal faults on power semiconductor devices which significantly degrades the overall systems performance. The analysis and diagnosis of faults are significantly important for ensuring reliable operation of a drive systems. This paper rigorously analyzes effects of internal faults on a ZSI-fed single-phase induction motor drive system using MATLAB/Simulink. This study proposes a fault protection technique using bidirectional converter to facilitate post fault operation of inverter drives. The simulation circuits with the related harmonic spectrum are presented after introducing an open circuit fault and a short circuit fault in a drive system with and without a protection circuit. From systematic analyses it is found that fault occurrence leads to harmonic distortion to a significant level in the proposed drive system and therefore, affects the systems efficiency. The proposed fault protection technique proves to be an effective method to reduce harmonics in the drive system during a fault occurrence. THe FFT spectrum results are tabulated and compared.

Title: "No-Load and Over Load Protection for Single Phase Induction Motors"

Author: Rahul Santhosh, Sailakshmi, Vikram Siddharthan M and Sudha Yadav

Description: Electric motors are the commonly used energy converter in various machines. The electrical energy is converted to mechanical energy with the help of electric motors. To protect the electric motors & connected machines from overload or short circuit conditions and save electrical energy when a no-load condition persists, there are different types of motor protectors used. There are different load conditions which may spoil the motor or even leads to a fire hazard. These abnormal load conditions of the motor must be monitored and initiate the shutdown action after a suitable confirmation time delay. This article describes the working of a no-load and overload/ short circuit protection for single phase induction motors using a microcontroller Atmega16, firmware development, and PCB design. The no-load protection feature is a new concept that is normally not available in commercially marketed motor protection devices. The microcontroller is used for fast computing and to take appropriate control action quickly. By using a microcontroller, the Variable Site Parameters (VSP) like upper and lower current limits can be configured by field engineers and an interfaced LCD display is used to monitor the values of VSPs and the operational events including real time load current.

Title: “Investigation of Digital Protection Relay For Three-Phase Induction Motor”

Author: Tatyana Dimova

Description: The article considers a simplified circuit solution for protection of induction motors through a device which provides reliable protection of a wide range of faults leading to unwanted interruptions of a variety of technological processes. The design of the electrical device is classical in combination with an electronic unit for monitoring the electrical parameters of the circuit. The operation of the protective device has been tested, and the results compared with similar devices.

Title: “Set Point of the Thermal Overload Protection of Large Induction Motors by Simulated Testing”

Author: Noel de Jesús Orozco Morales; Enrique C. Quispe; Julio R. Gómez Sarduy

Description:

This paper proposes and evaluates an electrical and thermal dynamic model to simulate the effect of voltage unbalance in the parameterization of digital protection of the large induction motors. Based on space vectors, the electrical model has variable parameters with temperature, skin effect and saturation. The thermal capacitances and conductance of the thermal model are identified from the MATLAB Simulink parameter estimation tool. The 1.8 MW motor of the feedwater pump of a power plant is evaluated. The errors concerning the stable state temperatures of the manufacturer’s heating test are 1.1 °C for the stator and 0.56 °C for the frame. The temperature increase of the winding during operating cycles with variable load and different unbalanced voltage were simulated and compared with the thermal image of the relay, demonstrating the importance of these tests in the protection adjustment.

CHAPTER 3
PROPOSED METHOD

3.1 Block Diagram

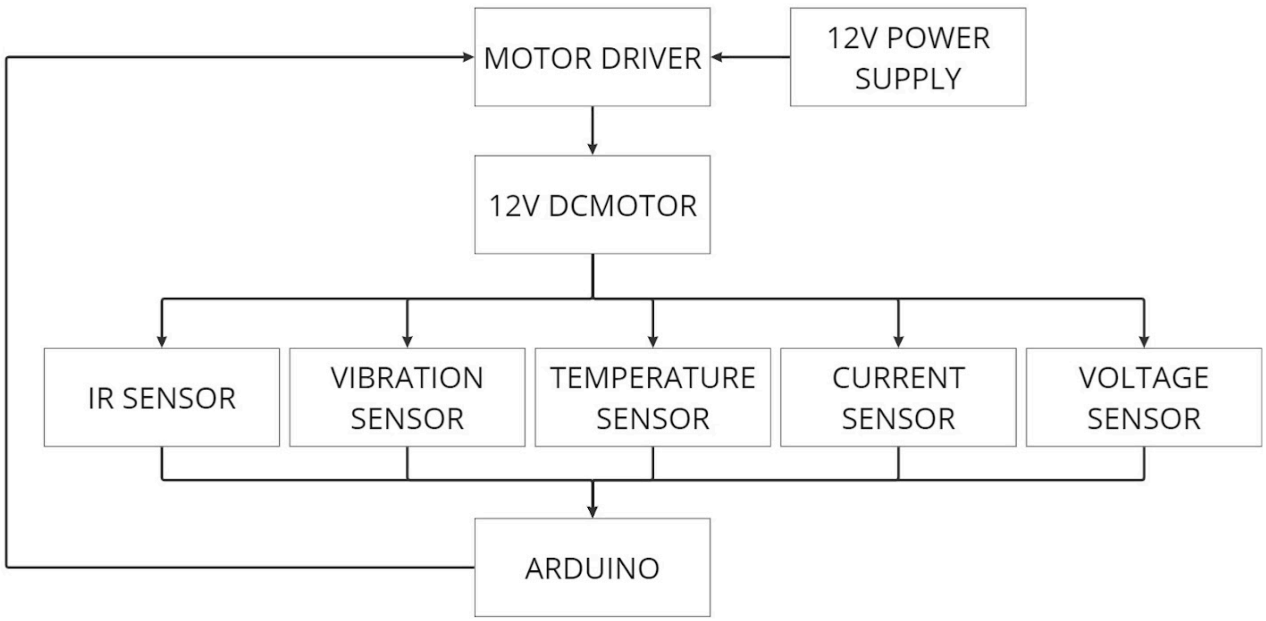

Figure 3.1.1: Block diagram of induction motor protection using arduino

3.2 Block Diagram Explanation

3.2.1 LM35D TEMPERATURE SENSOR

The LM35 series are precision integrated-circuit temperature devices with an output voltage linearly-proportional to the Centigrade temperature. The LM35 device has an advantage over linear temperature sensors calibrated in Kelvin, as the user is not required to subtract a large constant voltage from the output to obtain convenient Centigrade scaling. The LM35 device does not require any external calibration or trimming to provide typical accuracies of ±¼°C at room temperature and ±¾°C over a full −55°C to 150°C temperature range. Lower cost is assured by trimming and calibration at the wafer level. The low-output impedance, linear output, and precise inherent calibration of the LM35 device makes interfacing to readout or control circuitry especially easy. The device is used with single power supplies, or with plus and minus supplies. As the LM35 device draws only 60 µA from the supply, it has very low self-heating of less than 0.1°C in still air. The LM35 device is rated to operate over a −55°C to 150°C temperature range, while the LM35C device is rated for a −40°C to 110°C range (−10° with improved accuracy). The LM35-series devices are available packaged in hermetic TO transistor packages, while the LM35C, LM35CA, and LM35D devices are available in the plastic TO-92 transistor package. The LM35D device is available in an 8-lead surface-mount small-outline package and a plastic TO-220 package.

Specification :

- Operating Voltage: 4 V to 30 V
- Output Voltage: 10mV/°C
- Sensitivity: 10mV/°C
- Linearity Error: ±1°C (for 0°C to +100°C)
- Operating Temperature: -55°C to +150°C
- Output Impedance: 100 Ω
- Power Consumption: 60 µA (typical)
- Package Type: TO-92, TO-220, SOIC
- Output Type: Analog
- Accuracy: ±1°C (typical)

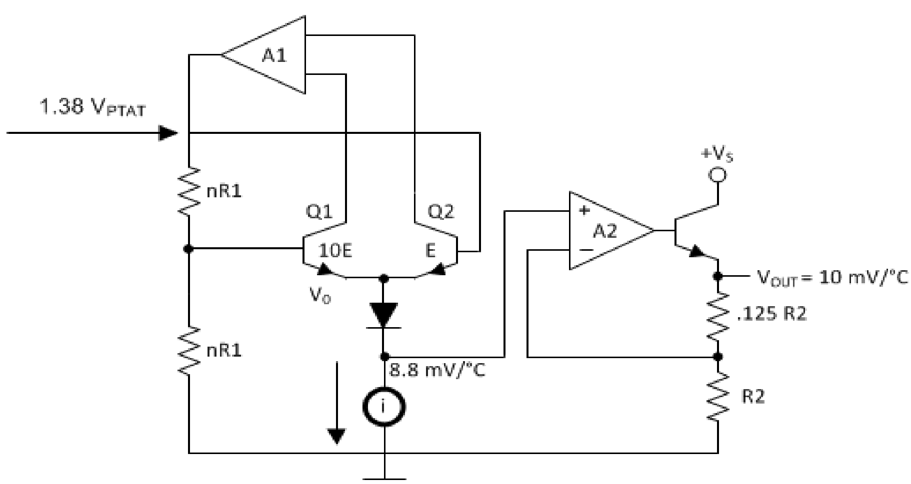

Figure 3.2.1: Block diagram of temperature sensor

3.2.2 IR SENSOR

Infrared Obstacle Avoidance IR Sensor Module (Active Low) has a pair of infrared transmitting and receiving tubes. When the transmitted light waves are reflected back, the reflected IR waves will be received by the receiver tube. The onboard comparator circuits does the processing and the green indicator LED comes to life.The module features a 3 wire interface with Vcc, GND, and an OUTPUT pin on its tail. It works fine with 3.3 to 5V levels. Upon hindrance/reflectance, the output pin gives out a digital signal (a low-level signal). The onboard preset helps to fine-tune the range of operation, the effective distance range is 2 cm to 80cm.

Specifications:

- Board Size 3.2 x 1.4cm
- Working voltage 3.3 to 5V DC
- Operating voltage 3.3V: ~23 mA, to 5V: ~43 mA
- Detection range 2cm – 30cm (Adjustable using potentiometer)
- Active output level The output is “0” (Low) when an obstacle is detected

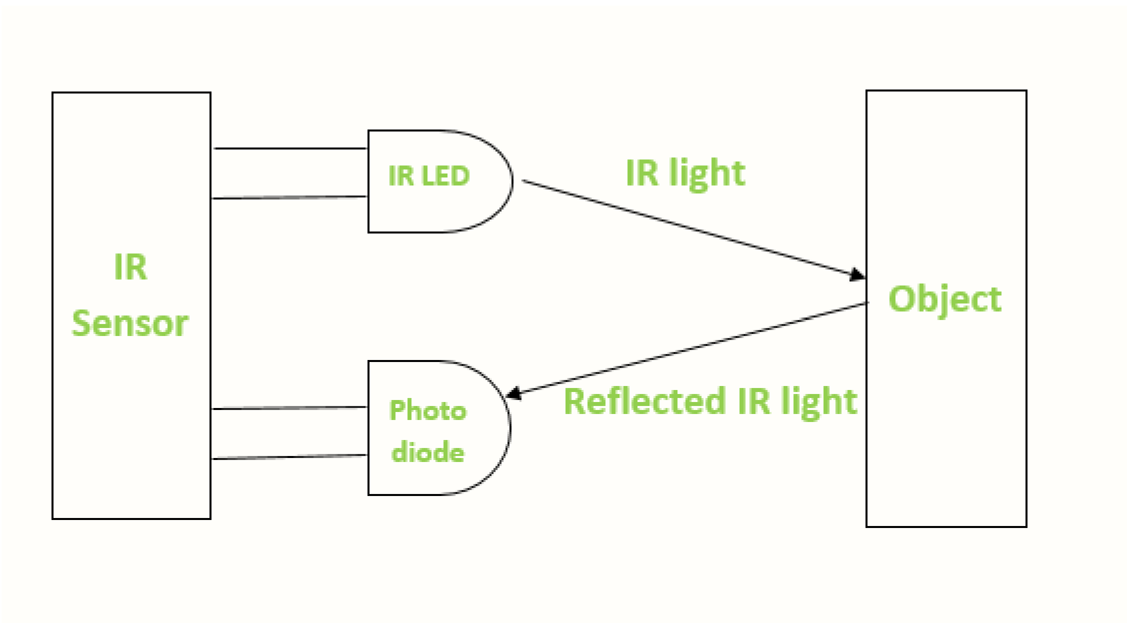

Figure 3.2.2: Block diagram of IR Sensor

3.2.3 L293

VIBRATION SENSOR

Single-roller type full induction trigger switch. When no vibration or tilt, the product is ON conduction state, and in the steady state, when a vibration or tilt, the switch will be rendered instantly disconnecting the conductive resistance increases, generating a current pulse signal, thereby triggering circuit. These products are completely sealed, waterproof, and dustproof. This sensor module produces logic states depending on vibration and external force applied on it. When there is no vibration this module gives logic LOW output. When it feels vibration then output of this module goes to logic HIGH. The working bias of this circuit is between 3.3V to 5V DC. Connect Vcc pin of sensor board to 5V pin of Arduino board, connect Gnd pin to Gnd pin of Arduino, Connect DO output signal pin of sensor board to Arduino digital pin D3. Do some calibration and adjust the sensitivity threshold, then upload the following sketch to the Arduino board.

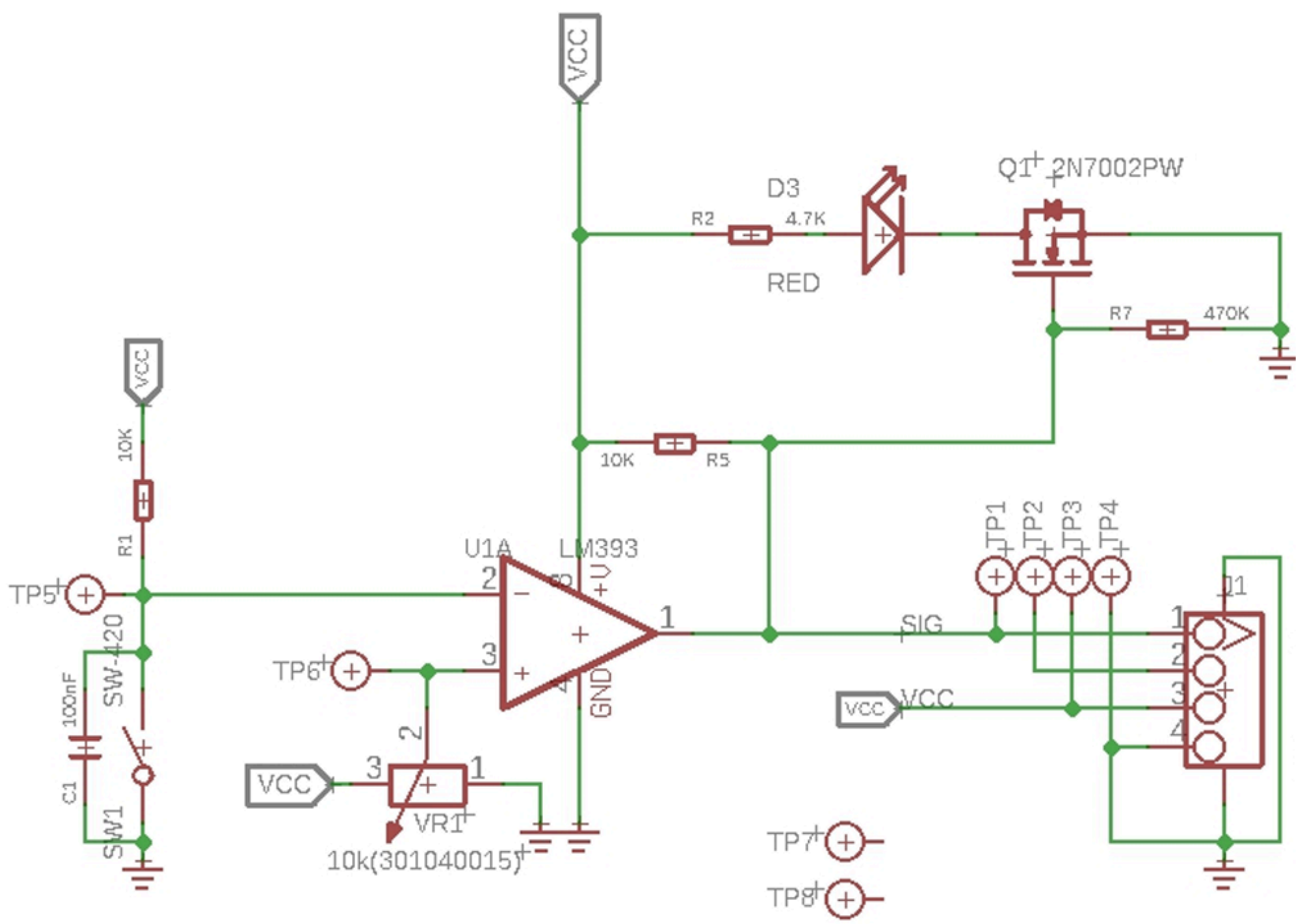

Figure 3.2.3: Block diagram of vibration sensor

3.2.4 ACS712 CURRENT SENSOR

The Allegro ACS712 provides economical and precise solutions for AC or DC current sensing in industrial, commercial, and communications systems. The device package allows for easy implementation by the customer. Typical applications include motor control, load detection and management, switched-mode power supplies, and overcurrent fault protection. The device consists of a precise, low-offset, linear Hall sensor circuit with a copper conduction path located near the surface of the die. Applied current flowing through this copper conduction path generates a magnetic field which is sensed by the integrated Hall IC and converted into a proportional voltage. Device accuracy is optimized through the close proximity of the magnetic signal to the Hall transducer. A precise, proportional voltage is provided by the low-offset, chopper-stabilized BiCMOS Hall IC, which is programmed for accuracy after packaging. The output of the device has a positive slope (>VIOUT(Q)) when an increasing current flows through the primary copper conduction path (from pins 1 and 2, to pins 3 and 4), which is the path used for current sensing. The internal resistance of this conductive path is 1.2 mΩ typical, providing low power loss. The thickness of the copper conductor allows survival of the device at up to 5× overcurrent conditions. The terminals of the conductive path are electrically isolated from the sensor leads (pins 5 through 8). This allows the ACS712 current sensor to be used in applications requiring electrical isolation without the use of opto-isolators or other costly isolation techniques. The ACS712 is provided in a small, surface mount SOIC8 package. The leadframe is plated with 100% matte tin, which is compatible with standard lead (Pb) free printed circuit board assembly processes. Internally, the device is Pb-free, except for flip-chip high-temperature Pb-based solder balls, currently exempt from RoHS. The device is fully calibrated prior to shipment from the factory.

Specifications:

- Supply Voltage: 4.5V~5.5V DC
- Measure Current Range: 30A
- Sensitivity: 100mV/A

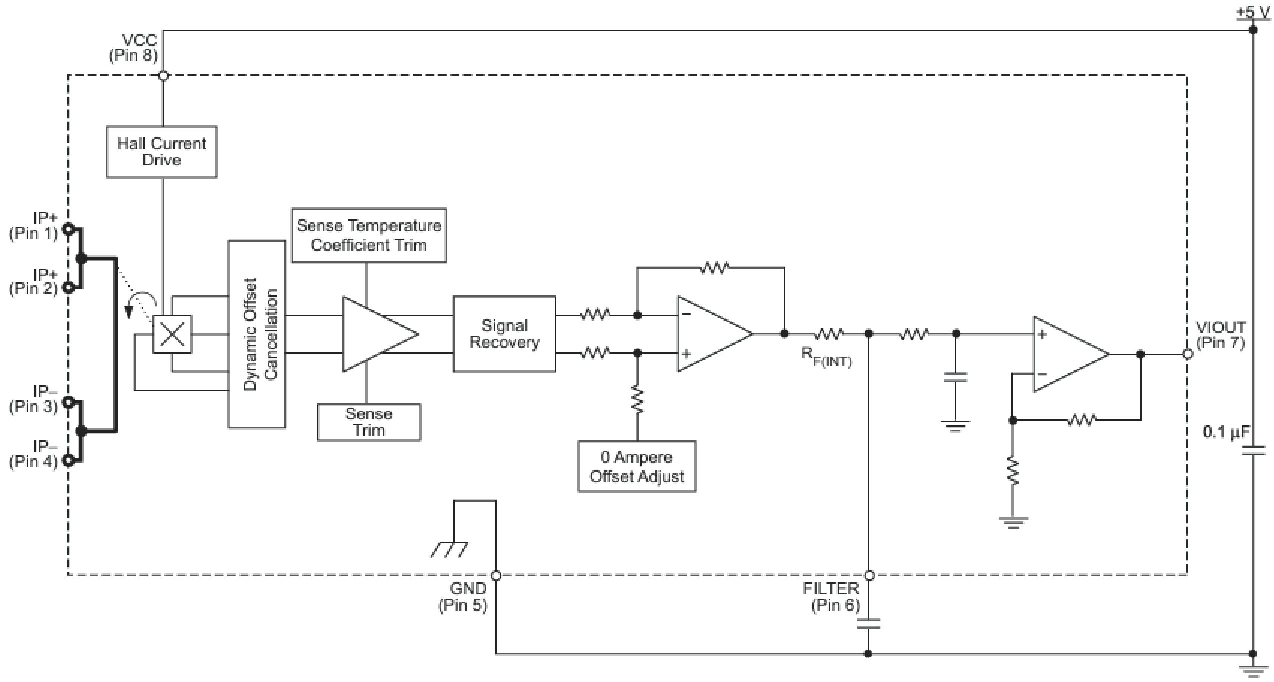

Figure 3.2.4: Block diagram of Current sensor

3.2.5 B25 VOLTAGE SENSOR

Voltage Detection Sensor Module 25V module is based on the principle of resistive voltage divider design, it can make the red terminal connector input voltage 5 times smaller. Arduino analog input voltages up to 5 v. The voltage detection module input voltage is not greater than 5Vx5=25V (if using 3.3V systems, the input voltage is not greater than 3.3Vx5=16.5V). It has a limit of Arduino analog input of 5V DC only. So if you wish to measure higher voltages, you will need to resort to another means. One way is to use a voltage divider. The one discussed here is found all over Amazon and eBay. It is fundamentally a 5:1 voltage divider using a 30K and a 7.5K Ohm resistor

Specifications:

- Input Voltage: 0 to 25V
- Voltage Detection Range: 0.02445 to 25
- Analog Voltage Resolution: 0.00489V
- Dimensions: 4 × 3 × 2 cm

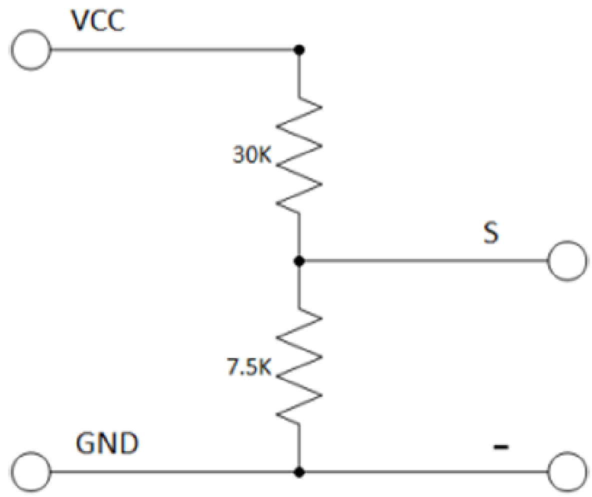

Figure 3.2.5: Block diagram of Voltage sensor

3.2.6 12V DC Geared Motor

150 RPM – 12V Centre Shaft DC Geared Motors are generally a simple DC motor with a gearbox attached to it. This can be used in all-terrain robots and variety of robotic applications. These motors have a 3 mm threaded drill hole in the middle of the shaft thus making it simple to connect it to the wheels or any other mechanical assembly. L298N H-bridge module with onboard voltage regulator motor driver can be used with this motor that has a voltage of between 5 and 35V DC. L298N H-bridge module with onboard voltage regulator motor driver can be used with this motor that has a voltage of between 5 and 35V DC.

Specifications:

- RPM: 150.
- Operating Voltage: 12V DC
- Gearbox: Attached Plastic (spur)Gearbox
- Shaft diameter: 6mm with internal hole
- Torque: 2 kg-cm
- No-load current = 60 mA(Max)
- Load current = 300 mA(Max).

3.2.7 L293 MOTOR DRIVER

The L293 and L293D devices are quadruple high-current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, DC and bipolar stepping motors, as well as other high-current/high-voltage loads in positive- supply applications. Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo- Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN. The L293 and L293D are characterized for operation from 0C to 70C. The L293D IC receives signals from the microprocessor and transmits the relative signal to the motors. It has two voltage pins, one of which is used to draw current for the working of the L293D and the other is used to apply voltage to the motors. The L293D switches it output signal according to the input received from the microprocessor.

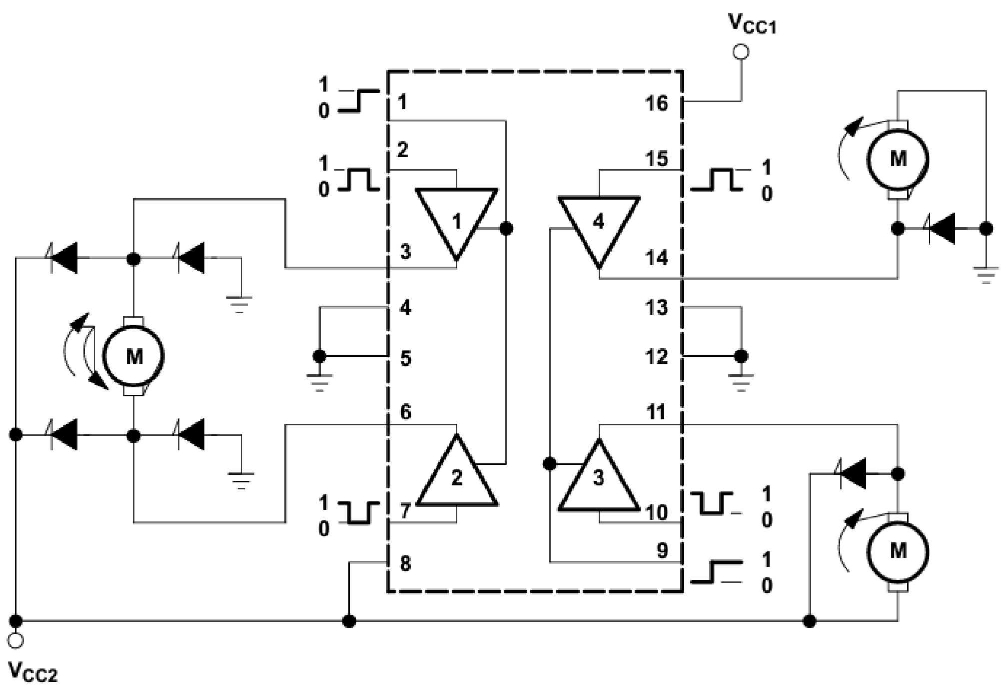

Figure 3.2.7: Block diagram of Motor Driver

Specifications:

- Supply voltage: 4.5 V to 12 V
- Current consumption: 600mA per motor
- Maximum Peak motor current: 1.2A
- Transition time: 300ns (at 5Vand 24V)
- Speed and Direction control is possible of both the motors
- Can be used to run Two DC or stepper motors with the same IC
- VCC: 4.5V – 12V Power Supply.
- GND: Ground.
- A1, A2 and B1, B2: input pins used for providing a control signal from the controller to run the motor in different directions.
- EN_A and EN_B: Enable pins.
- Connect 5v DC to EN_A and EN_B pin to operate the motor at its normal speed.
- MA: Motor -1 User can connect the motor terminal here.
- MB: Motor -2 User can connect another motor terminal here.

3.2.8 Arduino UNO

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller (MCU) and developed by Arduino.cc and initially released in 2010.The microcontroller 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.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. It can be powered by a USB cable or a barrel connector that accepts voltages between 7 and 20 volts, such as a rectangular 9-volt battery.

The hardware structure of Arduino Uno

- Microcontroller
- 14 Digital Pin
- 6 Analog Pins
- Power Supply

- Power Jack
- USB Port
- Reset Button

Microcontroller: Microcontroller is the central processing unit of Arduino Uno.

Digital Pins: There are 14 digital pins on Arduino Uno which can be connected to components like LED, LCD, etc.

Analog Pins: There are 6 analog pins on the Uno. These pins are generally used to connect sensors because all the sensors generally have analog values. Most of the input components are connected here.

Power Supply: The power supply pins are IOREF, GND, 3.3V, 5V, Vin are used to connecting sensors because all the sensors generally have analog values. Most of the input components are connected here.

USB Port: This port function is to program the board or to upload the program. The program can be uploaded to the board with the help of Arduino IDE and USB cable.

Reset Button: This is used to restart the uploaded program.

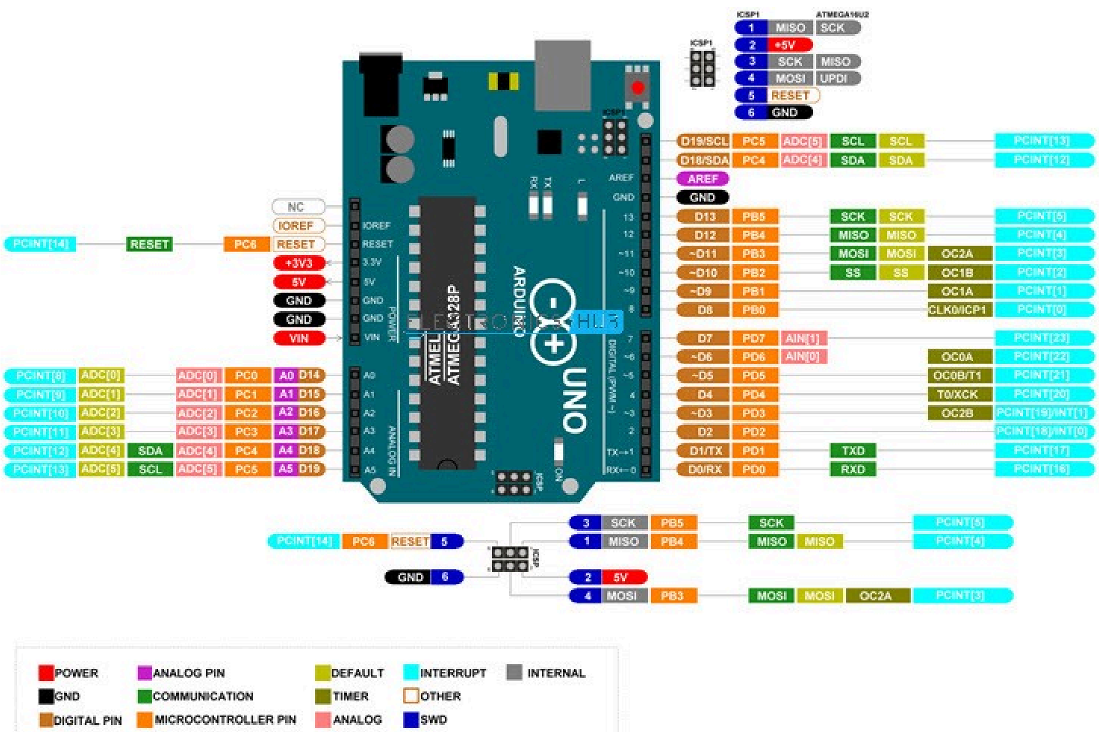

Figure 3.2.8 : Arduino UNO

Pin Number	**Pin Name**	**Description**	**Alternative Functions**
1	RX / D0	**Digital IO Pin 0 Serial RX Pin**	Generally used as RX
2	TX / D1	**Digital IO Pin 1 Serial TX Pin**	Generally used as TX
3	D2	Digital IO Pin 2	
4	D3	Digital IO Pin 3	Timer (OC2B)
5	D4	Digital IO Pin 4	Timer (T0/XCK)
6	D5	Digital IO Pin 5	Timer (OC0B/T1)
7	D6	Digital IO Pin 6	
8	D7	Digital IO Pin 7	
9	D8	Digital IO Pin 8	Timer (CLK0/ICP1)
10	D9	Digital IO Pin 9	Timer (OC1A)
11	D10	Digital IO Pin 10	Timer (OC1B)
12	D11	Digital IO Pin 11	SPI (MOSI) Timer (OC2A)
13	D12	Digital IO Pin 12	SPI (MISO)
14	D13	Digital IO Pin 13	SPI (SCK)
15	GND	Ground	
16	AREF	Analog Reference	
17	SDA / D18	Digital IO Pin 18	I2C Data Pin
18	SCL / D19	Digital IO Pin 19	I2C Clock Pin
19	NC	Not Connected	
20	IOREF	Voltage Reference	
21	RESET	Reset (Active LOW)	
22	3V3	Power	
23	5V	+5V Output from regulator or +5V regulated Input	
24	GND	Ground	
25	GND	Ground	

26	VIN	Unregulated Supply	
27	A0	Analog Input 0	Digital IO Pin 14
28	A1	Analog Input 1	Digital IO Pin 15
29	A2	Analog Input 2	Digital IO Pin 16
30	A3	Analog Input 3	Digital IO Pin 17
31	A4	Analog Input 4	Digital IO Pin 18 I2C (SDA)
32	A5	Analog Input 5	Digital IO Pin 19 I2C (SCL)

CHAPTER 4

IMPLEMENTATION AND WORKING

4.1 Circuit Diagram

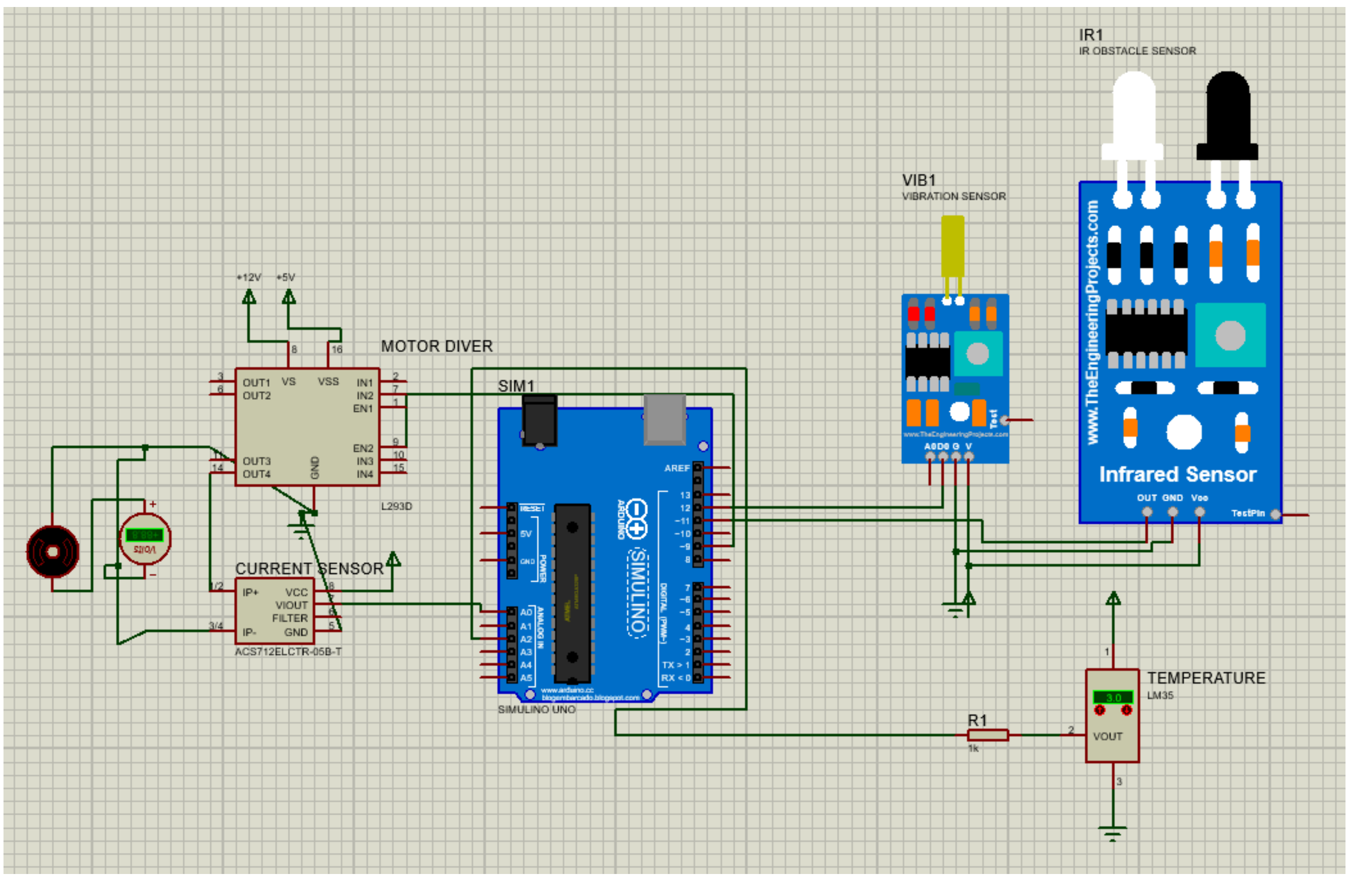

Figure 4.1: Circuit Diagram of Induction Motor Protection Using ARDUINO

4.2 Working

- The proposed Induction Motor Protection system utilizing Arduino aims to enhance motor reliability by monitoring critical parameters such as temperature, vibration, RPM, current, and input voltage. This system employs a combination of sensors and a microcontroller to ensure efficient monitoring and protection of the induction motor.
    1. **Sensor Integration:** The temperature sensor tracks motor temperature to prevent overheating.The vibration sensor measures motor vibration, indicating potential issues such as imbalance or misalignment.The IR sensor detects motor RPM, providing insights into motor speed and performance.Current and voltage sensors gauge amps and input voltage, respectively, to ensure optimal operating conditions.
    2. **Arduino Code:** A program is developed using the Arduino IDE to read the analog/digital signal from the sensors. This signal varies based on the condition of the motor. The analogRead()/digitalRead() function is used to obtain the sensor's output.
    3. **Sensors Interpretation:** The Arduino reads the analog/digital value from the sensors, which corresponds to the motor. If the value falls below a predetermined threshold level, it indicates that there needs to be a change in output to the motor driver.
    4. **Automated Turning Off:** If automated turning of the motor is implemented, the Arduino can deactivate the motor using the motor driver. The motor can be turned off using the motor driver.
    5. **Monitoring and Adjustment:** Monitoring the Serial Monitor output allows users to observe sensor readings and system responses. Sensor thresholds and turning off the motor can be based on the specific condition of the motor.

4.3 Applications

This system offers a reliable solution to monitor and safeguard critical motor parameters such as temperature, vibration, RPM, current, and input voltage. By utilizing sensors and an Arduino microcontroller, it provides real-time monitoring and analysis, enabling timely detection of potential faults. Industries ranging from manufacturing plants to water treatment facilities can benefit from this system by ensuring prolonged motor lifespan, minimizing downtime, and enhancing operational efficiency. Whether it's preventing overheating, detecting abnormal vibrations, or monitoring motor speed, this application of Arduino-based motor protection ensures reliable and uninterrupted motor performance, ultimately contributing to safer and more productive industrial operations.

4.4 Software implementation

```
const int ENA = 3;                    // Enable motor A
const int IN1 = 4;                   // Motor A input 1
const int IN2 = 5;                   // Motor A input 2
const int voltageSensorPin = A0;      // Analog pin for voltage sensor
const int currentSensorPin = A1;      // Analog pin for current sensor
const int irSensorPin = 2;            // Digital pin for IR sensor
const int vibrationSensorPin = 6;     // Digital pin for vibration sensor
const int resetButtonPin = 7;         // Digital pin for reset button
const int temperatureSensorPin = A2;  // Analog pin for LM35D temperature sensor

volatile unsigned long pulseCount;
unsigned long lastPulseCount;
unsigned long lastMillis;

void setup() {
  // Set the motor control pins as outputs
  pinMode(ENA, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);

  // Set up IR sensor
  pinMode(irSensorPin, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(irSensorPin), countPulses, FALLING);
```

```
  // Set up vibration sensor
  pinMode(vibrationSensorPin, INPUT);

  // Set up reset button
  pinMode(resetButtonPin, INPUT_PULLUP);

  Serial.begin(9600); // Initialize serial communication for debugging

  pulseCount = 0;
  lastPulseCount = 0;
  lastMillis = millis();
}

void loop() {
  // Read vibration sensor
  int vibrationState = digitalRead(vibrationSensorPin);

  if (vibrationState == LOW) {
    // Vibration detected, stop the motor and wait until reset button is pressed
    digitalWrite(ENA, LOW);  // Disable motor A
    Serial.println("Vibration detected, stopping motor.");

    while (digitalRead(resetButtonPin) == HIGH) {
      // Wait until reset button is pressed
    }
    // Reset pulse count and resume motor operation
    pulseCount = 0;
    digitalWrite(ENA, HIGH);
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, LOW);
  } else {
    // Read temperature sensor
    int temperature = analogRead(temperatureSensorPin);
    float voltage = temperature * (5.0 / 1023.0);
    float celsius = voltage * 100;  // LM35D outputs 10mV per degree Celsius
    Serial.print("Temperature: ");
    Serial.print(celsius);
    Serial.println(" degrees Celsius");

    // Check temperature condition
    if (celsius >= 50) { // Adjust threshold temperature as needed
      // Overheating detected, stop the motor and wait until reset button is pressed
```

```
      digitalWrite(ENA, LOW);  // Disable motor A
      Serial.println("Overheating detected, stopping motor.");

      while (digitalRead(resetButtonPin) == HIGH) {
        // Wait until reset button is pressed
      }
      // Reset pulse count and resume motor operation
      pulseCount = 0;
      digitalWrite(ENA, HIGH);
      digitalWrite(IN1, HIGH);
      digitalWrite(IN2, LOW);
    } else {
      // Read current sensor
      float current = (analogRead(currentSensorPin) - 512) * (5.0 / 1023.0) / 0.185; //
Convert analog reading to current (in Amperes)
      Serial.print("Current: ");
      Serial.print(current);
      Serial.println(" A");

      // Check current condition
      if (current >= 0.3) { // Adjust threshold current as needed
        // Excessive current detected, stop the motor and wait until reset button is pressed
        digitalWrite(ENA, LOW);  // Disable motor A
        Serial.println("Excessive current detected, stopping motor.");

        while (digitalRead(resetButtonPin) == HIGH) {
          // Wait until reset button is pressed
        }
        // Reset pulse count and resume motor operation
        pulseCount = 0;
        digitalWrite(ENA, HIGH);
        digitalWrite(IN1, HIGH);
        digitalWrite(IN2, LOW);
      } else {
        // No overheating or excessive current detected, rotate motor in one direction at full
speed
        digitalWrite(ENA, HIGH);  // Enable motor A
        digitalWrite(IN1, HIGH);
        digitalWrite(IN2, LOW);

        // Read and print voltage
        float voltage = analogRead(voltageSensorPin) * (5.0 / 1023.0) * (25.0 / 5.0); //
Convert analog reading to voltage
```

```
        Serial.print("Voltage: ");
        Serial.print(voltage);
        Serial.println(" V");

        // Calculate RPM
        unsigned long currentTime = millis();
        unsigned long elapsedTime = currentTime - lastMillis;
        if (elapsedTime >= 1000) {  // Calculate RPM every second
          unsigned long pulseDifference = pulseCount - lastPulseCount;
          float rpm = (pulseDifference / 20.0) * (60000.0 / elapsedTime); // 20 slots on the
encoder wheel
          Serial.print("RPM: ");
          Serial.println(rpm);
          lastPulseCount = pulseCount;
          lastMillis = currentTime;
        }
      }
    }
  }

  delay(1000); // Wait for 1 second
}

void countPulses() {
  pulseCount++;
}
```

4.5 Components

S.NO	COMPONENTS	MODEL TYPE (or) RANGE	Quantity
1	Temperature sensor	LM35D	1
2	Vibration sensor	SW-420	1
3	Current sensor	ACS712	1
4	Voltage sensor	(0-25)V DC	1
5	IR sensor	LM393	1
6	12V DC motor	12V DC	1
7	DC motor driver	L293D	1
8	DC adapter	12V	1
9	Arduino	UNO	1

CHAPTER 5

RESULTS AND OUTPUT

The successful operation of the module was guaranteed in order to thoroughly test the different Sensor Modules we provided.

Output of Sensor integrated

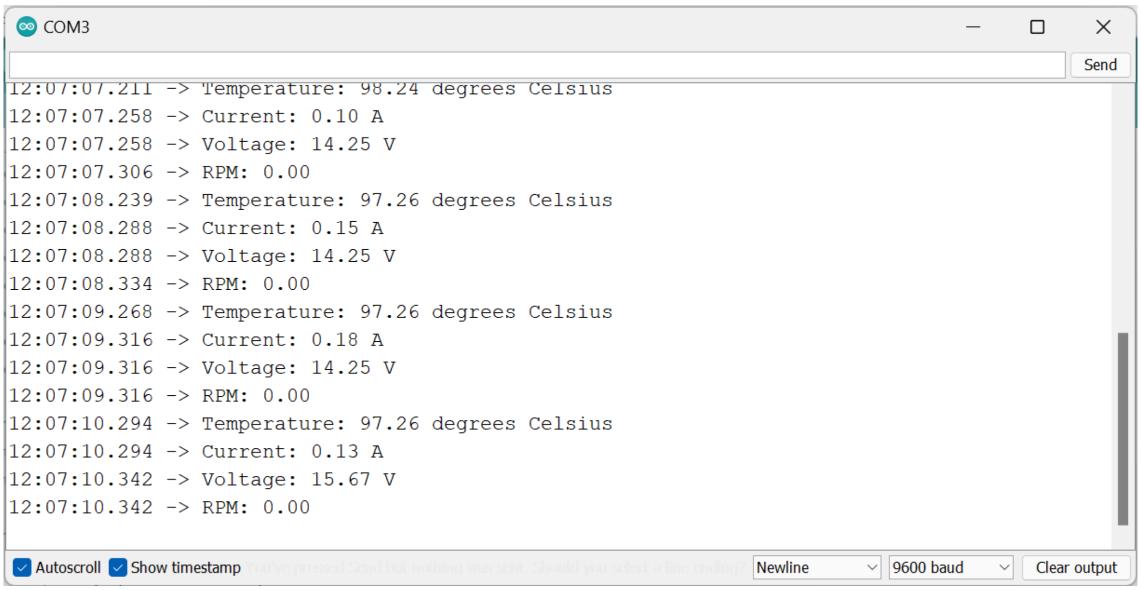

Figure 5.1: Output of Sensor Integrated

Hardware Output

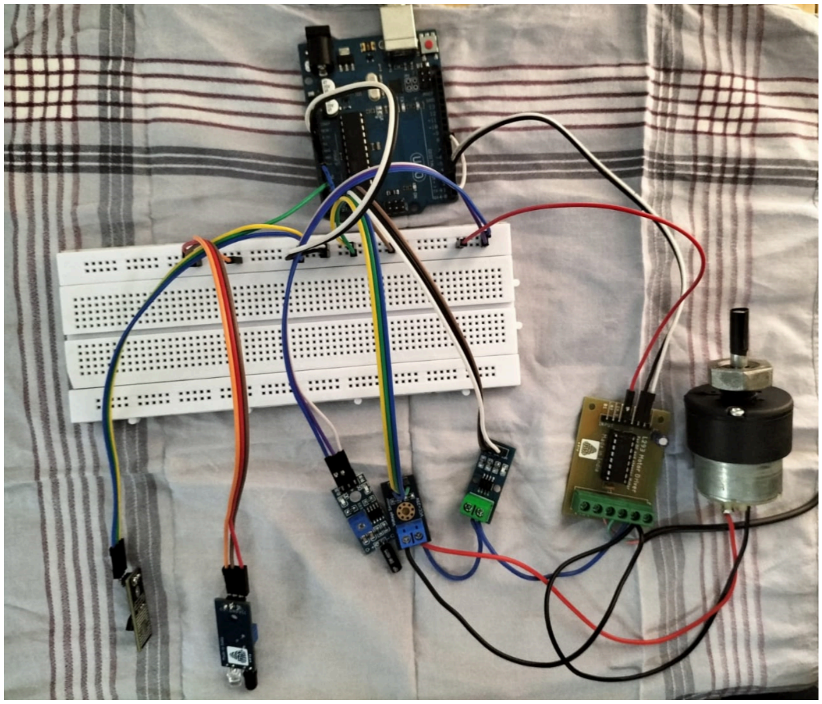

Figure 5.2: Hardware Output

CHAPTER 6
CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

In conclusion, the Induction Motor Protection system utilizing Arduino presents a comprehensive and effective solution for monitoring and safeguarding critical motor parameters in industrial settings.This project offers significant benefits, including prolonged motor lifespan, enhanced operational efficiency, and improved safety across various industries reliant on induction motors. Moving forward, further enhancements and refinements to this system could lead to even greater reliability and performance, solidifying its position as a valuable tool in ensuring the smooth operation of industrial processes.

6.2 Future Scope

The future scope of our project on induction motor protection holds immense potential, particularly in its adaptability to monitor multiple motors simultaneously, thereby ensuring the safety and operational integrity of an entire system. By extending the application to encompass numerous motors, our project addresses the need for a comprehensive and centralized approach to motor protection, especially in environments where the presence of labor may be limited. This expansion facilitates a proactive and automated system that not only safeguards individual motors but also contributes to the overall efficiency and longevity of the machinery. Taking the project a step further, the integration of Internet of Things (IoT) technology enhances the monitoring system's accuracy and responsiveness. By connecting the induction motor protection system to the IoT, real-time data on motor performance, temperature, and other critical parameters can be gathered and analyzed remotely. This not only streamlines maintenance efforts but also enables predictive maintenance strategies, reducing downtime and optimizing resource utilization. The amalgamation of multiple motor monitoring with IoT capabilities positions our project at the forefront of industrial automation, offering a sophisticated solution for ensuring motor safety and performance in diverse operational settings.

REFERENCES

[1] R. Santhosh, V. S. M, Sailakshmi and S. Yadav, "Hardware Design of Network Model of Protection and Controller module for Three-Phase Induction Motors in Industrial Plants with Remote Monitoring System on a Centralized System," 2022 International Conference on Innovations in Science and Technology for Sustainable Development (ICISTSD), Kollam, India, 2022, pp. 93-98, doi: 10.1109/ICISTSD55159.2022.10010403.

[2] V. Sharma, M. J. Hossain and S. M. N. Ali, "Fault Protection Technique for ZSI-fed Single-Phase Induction Motor Drive System," 2018 IEEE Region Ten Symposium (Tensymp), Sydney, NSW, Australia, 2018, pp. 30-35, doi: 10.1109/TENCONSpring.2018.8691986.

[3] R. Santhosh, Sailakshmi, V. S. M, S. Yadav, N. M and S. P, "No-Load and Over Load Protection for Single Phase Induction Motors," 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2021, pp. 462-466, doi: 10.1109/ICOSEC51865.2021.9591938.

[4] T. Dimova, "Investigation of Digital Protection Relay For Three-Phase Induction Motor," 2021 17th Conference on Electrical Machines, Drives and Power Systems (ELMA), Sofia, Bulgaria, 2021, pp. 1-4, doi: 10.1109/ELMA52514.2021.9503040.

[5] N. de Jesús Orozco Morales, E. C. Quispe, J. R. Gómez Sarduy and P. R. Viego Felipe, "Set Point of the Thermal Overload Protection of Large Induction Motors by Simulated Testing," 2022 IEEE ANDESCON, Barranquilla, Colombia, 2022, pp. 1-5, doi: 10.1109/ANDESCON56260.2022.9989526.