INDUSTRIAL INSTRUMENTATION LABPROJECT

TOPIC :- INTRODUCTION TO VIBRATION SENSOR

YEAR:- 3rd , SEMESTER:- 6th

Submitted by :- Bishal Rudra Paul(22UEE080)
Sandipan Som(22UEE057)
Anand Bharti(20UEE115)
Chaitanya Raj(21UEE071)
Diptanu Sarkar(22UEE088)
Tathagata Karmakar(22UEE078)

Submitted to :- Dr. DIPTANU DEY , Mr. RAJESH KUMAR

ACKNOWLEDGEMENT

We are very grateful to Mr. Rajesh Kumar Sir of Electrical Engineering Department of National Institute of Technology, Agartala, for providing us academic inputs, guidance and encouragement

Throughout the working period of our Project.

We would also like to express our heartfelt gratitude to Dr.Diptanu Dey Sir without whose permission and wise counsel we wouldn’t be able to pursue out work in this manner.

We would also like to thank all others who have directly or indirectly helped us in making this project a success.

Thanking You.

CONTENTS

- Introduction
- Materials Required
- Vibration Sensor Module SW-420
- Arduino Vibration Sensor Circuit Diagram
- Working of Vibration Sensor
- Arduino Uno Vibration Sensor Programming
- Testing the Arduino Vibration Sensor Circuit
- Complete Project Code
- Conclusion

Introduction:

Vibration sensors are widely used for detecting motion, shock, or impact in various applications such as security systems, industrial monitoring, and earthquake detection. These sensors help in identifying unusual mechanical activities, making them useful in predictive maintenance, safety alarms, and automated response systems. The SW-420 vibration sensor is a popular choice due to its simplicity and reliability. It operates on a normally closed principle, meaning that it remains in an inactive state until vibrations exceed a certain threshold. Once activated, the sensor outputs a digital signal that can be processed by a microcontroller such as an Arduino. There are several **types of vibration sensors** available in the market which can detect the vibration bys sensing acceleration or velocity and could provide excellent result.

In this project, we will interface the **SW-420 Vibration Sensor Module** with an **Arduino Uno** to detect vibrations and trigger actions accordingly. The system will turn on an LED when vibrations are detected, which can be extended to various applications like **security alerts, machinery monitoring, and earthquake detection systems**.

So in this project, a basic **vibration sensor module is interfaced with popular Arduino UNO** and whenever the vibration sensor detects any vibration or jerk an LED will start blinking**.**

Materials Required:

- Arduino UNO
- SW-420 Vibration Sensor Module
- 5mm LED
- Breadboard
- USB Cable for Uploading Programee
- Jumping wires

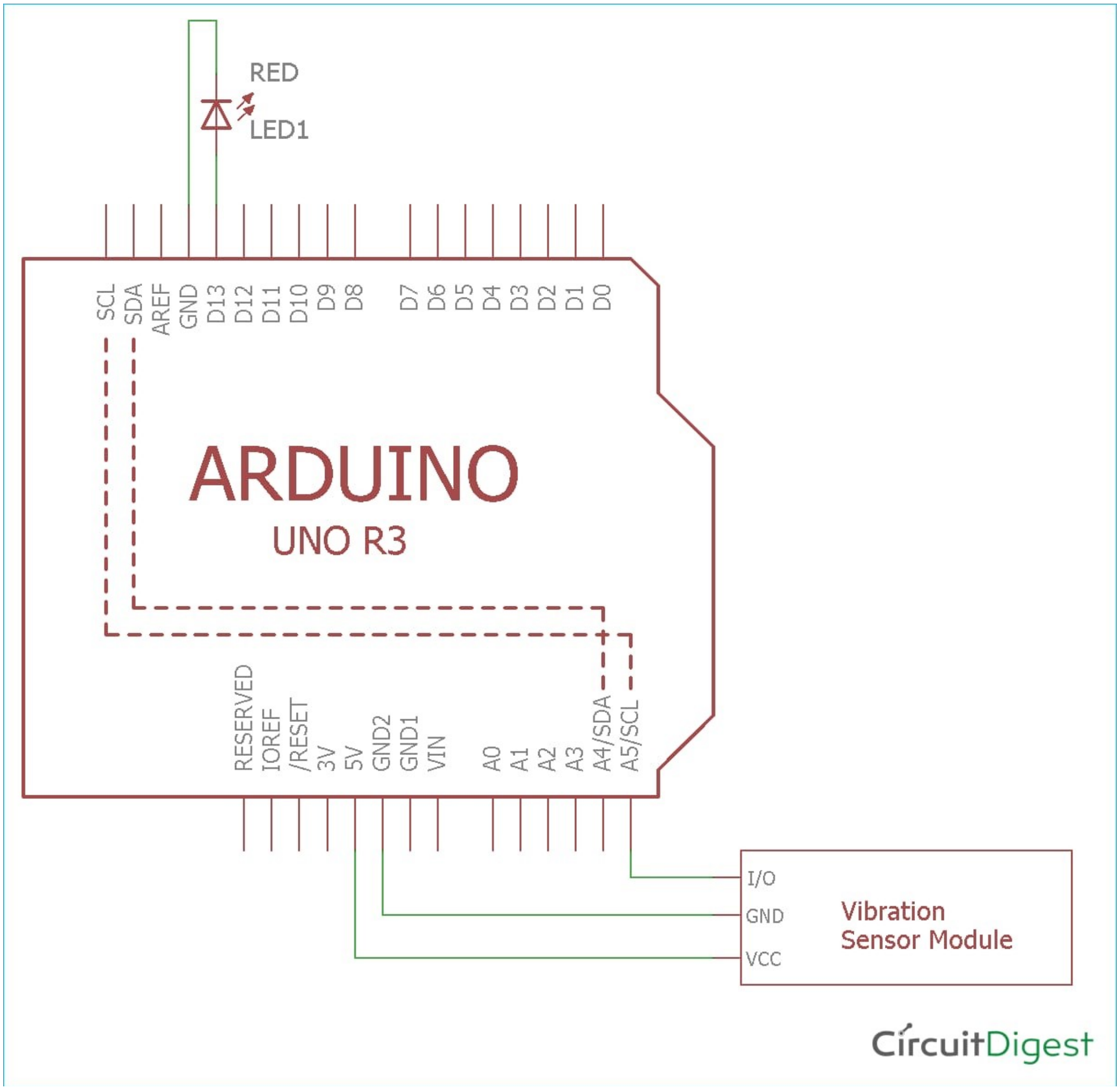
Vibration Sensor Module SW-420:

This is an **SW-420 vibration module**, which can work from 3.3V to the 5V. The sensor uses LM393 comparator to detect the vibration over a threshold point and provide digital data, Logic Low or Logic High, 0 or 1. **During normal operation, the sensor provides Logic Low and when the vibration is detected, the sensor provides Logic High.** There are three peripherals available in the module, two LEDs, one for the Power state and other for the sensor's output. Additionally, a potentiometer is available which can be further used to control the threshold point of the vibration. In this project, we will use 5V to power the module.

We used the same sensor in Anti-Theft Alert System using ATmega8 Microcontroller. Also a tilt sensor can be used to detect sudden accident.

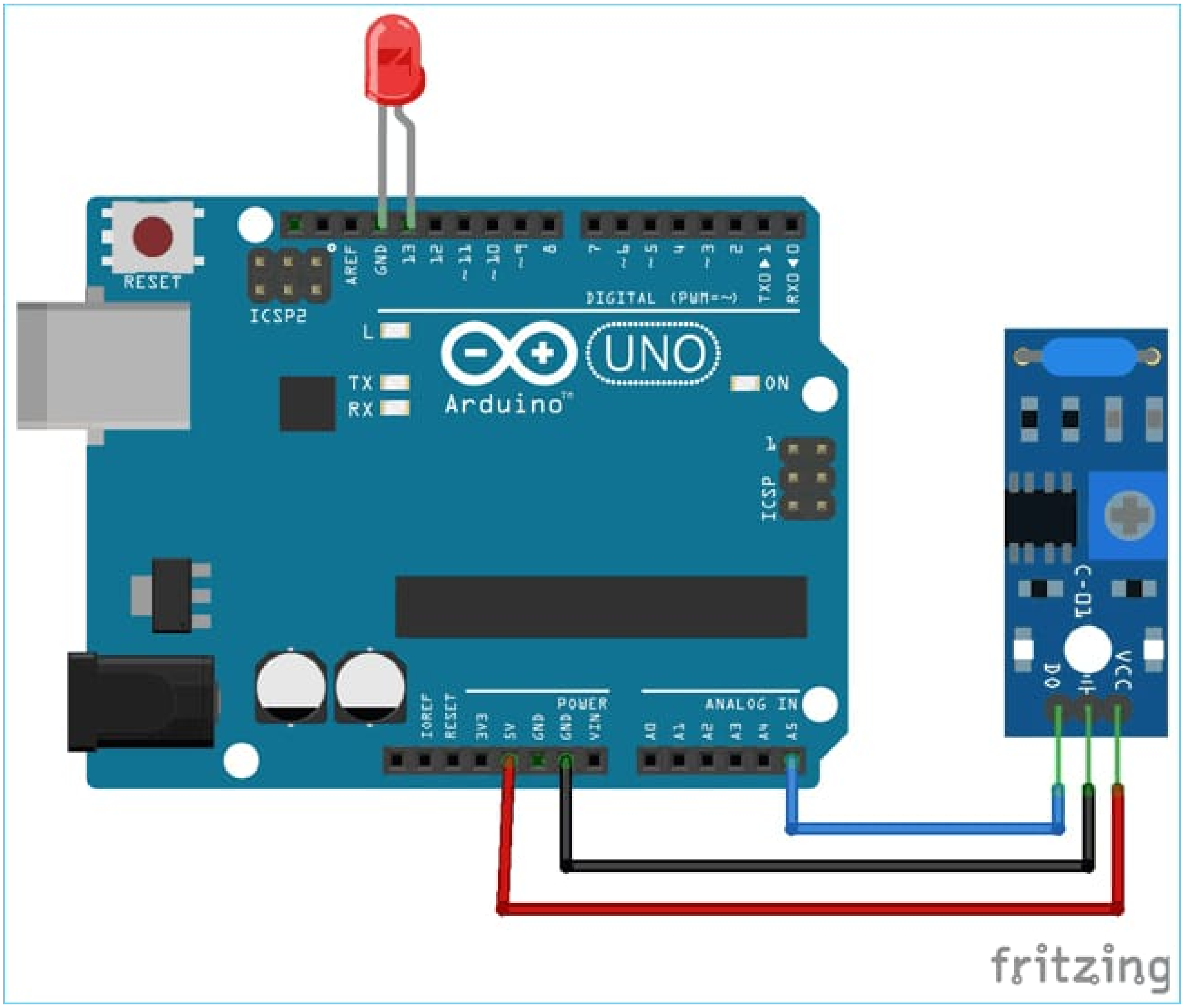
Arduino Vibration Sensor Circuit Diagram:

The schematic for **interfacing Vibration sensor with Arduino uno** is given below.



The LED is connected in the D13 pin. The module is powered using the available 5V pin in the Arduino. The Ground and the 5V pin are used to power up the Arduino whereas the A5 pin is used to get the data from the vibration sensor.

The circuit is constructed where the SW-420 module and LED are connected with Arduino Uno.



Working of Vibration Sensor:

The sensor works by detecting **mechanical vibrations**. When the intensity of vibrations crosses a certain threshold, the **comparator circuit (LM393)** outputs a digital HIGH signal. This signal can be read by an Arduino to trigger alarms or notifications.

Arduino Uno Vibration Sensor Programming:

Programming the Arduino UNO to interface vibration sensor doesn't require much effort as only the input pin should be monitored to make conclusion. The **complete code and working video are attached at the end**.

Initially the Header Files are included. The arduino header is included since this tutorial was written in Eclipse IDE with Arduino extension. This sketch will also work for Arduino IDE and while using this sketch in Arduino IDE, there is no need to include <Arduino.h> header.

```
#include <Arduino.h>
```

Here two macros are defined for ON and OFF.

```
#define ON 1

#define OFF 0
```

The below statement is used for integrating the LEDs and the Vibration Sensor. The vibration sensor is connected to pin A5. The inbuilt LED is also used which is directly connected in the board to pin 13. The 5mm LED is also connected to the pin 13.*

```
/*

* Pin Description

 */

int vibration_Sensor = A5;

int LED = 13;
```

Two integers are declared where the sensors present output and previous output will be stored, which will further used to detect whether the vibration is happening or not.

```
/*

 * Programme flow Description

 */

int present_condition = 0;

int previous_condition = 0;
```

The same pin which is declared as the peripheral connection, the direction of the pins are configured. The sensor pin as input and the LED pin as an output.

```
/*

 * Pin mode setup

 */
```

```
void setup() {
            pinMode(vibration_Sensor, INPUT);
            pinMode(LED, OUTPUT);
}
```

One function is written to blink the led twice. The delay can be configured by chaging the delay value.

```
void led_blink(void) {
            digitalWrite(LED, ON);
            delay(250);
            digitalWrite(LED, OFF);
            delay(250);
            digitalWrite(LED, ON);
            delay(250);
            digitalWrite(LED, OFF);
            delay(250);
}
```

In the loop function, **present and previous condition is compared**. If these two are not same, the **leds start to blink until the both are same**. In starting, the two variables hold 0 and the led remains off during the start of program. When there is some virbraion, ***present_condition*** variable becomes 1 and led start to blink. And again when vibrations stops both the vaiables becomes 0 and LED stops blinking.

```
void loop() {

            previous_condition = present_condition;

            present_condition = digitalRead(A5); // Reading digital data
from the A5 Pin of the Arduino.

            if (previous_condition != present_condition) {

                       led_blink();

                } else {

                       digitalWrite(LED, OFF);

            }

}
```

This finishes the programming the arduino UNO with Vibration sensor. The final step will be testing the whole setup.

Testing the Arduino Vibration Sensor Circuit:

The circuit doesn’t require additional breadboard. It can be simply tested using the Arduino UNO Board. The led is monitored when the vibration sensor is hit or if it changes its state. The led will blink connected to Pin 13 of Arduino UNO when there is some vibrations. If the vibration sensor doesn’t work then please check the connection and power. Avoid any loose connection between sensor and microcontroller.

So this is how a Vibration sensor can be interfaced with Arduino UNO.

Complete Project Code:

Copy Code

```
/*//==========================================================================
=========//
 * Vibration Sensor interfacing with Arduino
 * Date: - 15-04-2019
 * Author:- Sourav Gupta
 * For:- circuitdigest.com
 */
//===========================================================================
======//
#include <Arduino.h>
#include <stdio.h>
#define ON 1
#define OFF 0
/*
 * Pin Description
 */
int vibration_Sensor = A5;
int LED = 13;
/*
 * Programme flow Description
 */
int present_condition = 0;
int previous_condition = 0;
/*
 * Pin mode setup
 */
void setup() {
pinMode(vibration_Sensor, INPUT);
pinMode(LED, OUTPUT);
}
/*
 * Led blink
 */
void led_blink(void);
```

```
/*
```

```
 * main_loop
 */
void loop() {
previous_condition = present_condition;
present_condition = digitalRead(A5); // Reading digital data from the A5 Pin of the Arduino.
if (previous_condition != present_condition) {
led_blink();
} else {
digitalWrite(LED, OFF);
}
}
void led_blink(void) {
digitalWrite(LED, ON);
delay(250);
digitalWrite(LED, OFF);
delay(250);
digitalWrite(LED, ON);
delay(250);
digitalWrite(LED, OFF);
delay(250);
}
```

Conclusion:

This project demonstrates how to interface a **SW-420 Vibration Sensor Module** with an **Arduino** to detect vibrations. Such a setup can be used in security systems, earthquake alarms, and industrial monitoring applications.