IoT Based Bridge Health Monitoring System

Submitted in partial fulfillment of the

Requirement for the degree of

Bachelor of Engineering

in

Instrumentation Engineering

By

Mr. Ishaan K. Raipure(Reg.No.2016033700260106)
Mr. Vaibhav B. Pawar(Reg.No.2015033700274217)
Mr. Araf Z. Ansari(Reg.No.2016033700260153)
Mr. Rohit R. Jadhav(Reg.No.2016033700260211)

Under the guidance of

Prof. S. C. Rajgade

Department of Instrumentation Engineering,
Government College of Engineering,Chandrapur
(M.S.) India.

June-2020

Dedicated to Our Beloved Parents
and
Our Teachers

Who are the inspiration and power behind success of this work

Certificate

This is to certify that report entitled **"Iot Based Bridge Health Monitoring System"** is a bonafide record of project work carried out by **Mr.Ishaan Raipure, Mr.Vaibhav Pawar, Mr.Araf Ansari, Mr.Rohit Jadhav** under my supervision and guidance. This work is being submitted to the Government College of Engineering, Chandrapur (affiliated to Gondwana University, Gadchiroli) in partial fulfillment of the requirements for the Degree of **Bachelor of Engineering in Instrumentation** during the academic session **2019-2020**. The matter contained in this dissertation has not been submitted to any other university or institute for the award of any degree or diploma.

Prof. S. C. Rajgade
Guide,
Asst. Professor,
Department of Instrumentation Engg.,
GCOE, Chandrapur. (MS)

Forwarded by:

Dr. G. G. Bhutada
Head,
Department of Instrumentation Engg.,
GCOE, Chandrapur. (MS)

Dr. P. P. Bedekar
Principal
GCOE, Chandrapur. (MS)

Report Approval Sheet

Dissertation entitled as **"IoT Based Bridge Health Monitoring System"** by **Ishaan Raipure, Vaibhav Pawar, Araf Ansari, Rohit Jadhav** is approved for the degree of ***Bachelor Of Engineering in Instrumentation Engineering*** from Government College Of Engineering, Chandrapur (M.S.).

Prof. S. C. Rajgade
Guide,
Professor,
Dept. of Instrumentation Engineering,
GCOE, Chandrapur.(MS)

Examiner(s):

1. ----------

2. ----------

Acknowledgement

No words are sufficient to express my sincere gratitude to my guide, **Prof. S. C. Rajgade**, who has provided encouragement, assistance and invaluable guidance during the project work. It has been a great honour and privilege to work under their supervision and to be their student.

⋮

Submitted with regards.

Mr.Ishaan Raipure

Mr.Vaibhav Pawar

Mr.Afar Ansari

Mr.Rohit Jadhav

June, 2020.

Place : Chandrapur, Maharashtra,

India.

Abstract

The aim of this project is to support the construction of an efficient Health Monitoring System for ensuring the safety, using-life of bridges, preventing from the collapse affairs, protecting people lives, environment, and reducing unnecessary finance expenses.This report presents an effective method for application on vibration,temperature and water level signal analysis in Bridge Health Monitoring .The data that collected from sensors can be transported through GSM Module to the cloud storage which is Thingspeak cloud. The various signals from sensors are collected through wireless sensor network. These signals are very important in Bridge Health Monitoring System, because the variation of bridge signals indicates the changing of bridge structure state.The system working principle, hardware and software development of cloud storage has been carried in this report.

Contents

List of Figures

List of Tables

Abbreviations

IOT	Internet of Things
WSN	Wireless Sensor Network
TCP	Transmission Control Protocol
IP	Internet Protocol
GSM	Global System for Mobile Communications
GPRS	General Packet Radio Services
PWM	Pulse Width Modulation
USB	Universal Serial Bus
ICSP	In Circuit Serial Programming
TTL	Transistor Transistor Logic
SS	Slave Select
MOSI	Master Out Slave In
MISO	Master In Slave Out
SPI	Serial Peripheral Interface
LED	Light Emitting diode
IDE	Integrated Development Environment

Chapter 1

Introduction

The climate of today's world is full of uncertainty and we are building huge structures like a bridge in this uncertain climate where typhoons or earthquakes can be dangerous for the health of the structure like a bridge. The bridges are huge structure and include various factors like vibration, temperature, visibility which can get affected by natural and unnatural factors which can cause catastrophe. All these different threatening factors need to be monitored and requires specific knowledge about how to detect these factors in order to avoid any mishap. The definite information about the damaged site is paramount because of which the rescue team can reach the damaged site as soon as possible. This project is concerned with the integration of IoT, WSN, and instrumentation technology to develop a system, which is capable of monitoring various factors like vibration, temperature, and water level, which are responsible for the health of a bridge. The system is also capable of transmitting the data to a device like mobile and can store it to the cloud to analyze the condition of a bridge at any place. The following technologies are used in the making of this entire system to work sufficiently.

1.1 Internet of Things (IOT)

Internet of Things is a system which provides communication between different systems by connecting different devices with a network. This system does not neces-

sarily required any human to human or human to machine interaction. IoT is more about machine to machine interaction where two or more devices interact with each other over a network in order to complete a task or process. This device provides self decision making among the devices without any human interaction. In this study IoT helps in communication between implemented devices or sensors in bridge with rescue team's personal alarming devices.

1.2 Wireless Sensor Network(WSN)

Wireless Sensor Networks are sets of sensors specifically distributed at specific positions which are dedicated to a particular task like for monitoring and recording the environmental conditions like temperature, level, visibility etc, etc. These networks can be arranged in adhoc manner which requires wireless connectivity to so that the data being monitored can be transmitted from one place to another to make better decisions. In this system study this network helps in monitoring the health of bridge on the basis of factors like vibration, level and temperature.

1.3 Cloud Server

Cloud server is the network where various services are provided. This server is created by connecting various networks via internet. This server is used to provide services like data storage, web hosting, application or software use. In this project we used cloud server for data storing and its access for various analysis.

1.4 TCP/IP Protocol

TCP/IP stands for Transmission Control Protocol/Internet Protocol. These protocols are used to build the system under particular set. These sets of rules define

how data should be transmitted over the network.

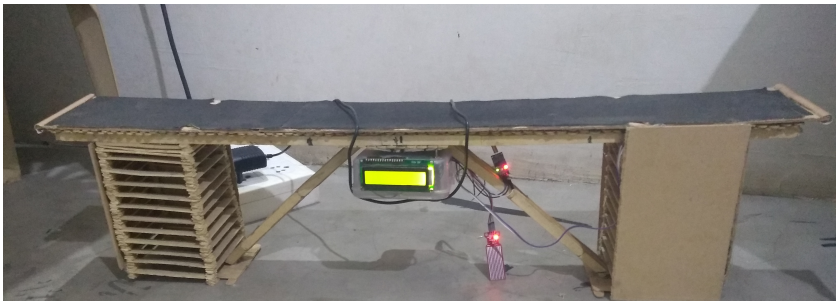


Figure 1.1: Prototype of System

1.5 Conclusion

In this chapter we have studied about need of monitoring the bridge including short information about IoT, WSN, Cloud Sever, TCP/IP Protocol.

Chapter 2

Literature Survey

Existing System of bridge safety management have the problems like Failure to collect data or monitor on-site conditions in real time. Data collection through visual assessments or use of large size electronic equipment have higher cost or higher power consumption, often resulting in inaccurate data.

In proposed system, we are going to use WSN technology. Here the detected data images are transmitted to the server and database for users to have real time monitoring of the bridge conditions via mobile telecommunication devices. Bridge Structural Health Monitoring (SHM) has been an intense research area for some time. Traditional, direct approaches are to collect acceleration signals by installing sensors on a bridge.

The drawback of such direct approaches is that they require a sophisticated and expensive electronic infrastructure with installation, maintenance and power support. Moreover, although it is easy to get a large number of data samples.

Year	Author	Paper Name	Publication details	Technology used
2017	Jin-Lian Lee,	Development of an IOT Based Bridge Safety Monitoring System	A multi sensor, dense wireless network for bridge monitoring	Real-Time
2015	Stevens	Research on railroad bridge monitoring plat- form based on IoT	log data from 3 channels 12 nodes over a sampling time of 4 minutes	Post sample delivery of logged data
2013	M.T. Rollins	Design of a WSN platform for longer environmental monitoring for IoT applications	2 local networks, each having 4 nodes, a base station connected to IEEE 802.11b wireless radio	Real-time
2011	Mike Waugh	Design of wireless sensor network based monitoring system for bridge	A wireless sensor network on a concrete box girder bridge alongside a wired system	Real-time

Table 2.1: Literature Survey of Bridge monitoring System

Chapter 3

Hardware Components Required

Following are the hardware components used to build the health monitoring of bridge.

3.1 Controller - Arduino UNO

3.1.1 Overview

Arduino Uno is a microcontroller board based on 8-bit ATmega328P microcontroller. Along with ATmega328P, it consists other components such as crystal oscillator, serial communication, voltage regulator, etc. to support the microcontroller. Arduino Uno has 14 digital input/output pins (out of which 6 can be used as PWM outputs), 6 analog input pins, a USB connection, A Power barrel jack, an ICSP header and a reset button.

3.1.2 Description

The 14 digital input/output pins can be used as input or output pins by using pinMode(), digitalRead() and digitalWrite() functions in arduino programming. Each pin operate at 5V and can provide or receive a maximum of 40mA current, and has an internal pull-up resistor of 20-50 KOhms which are disconnected by default. Out of these 14 pins, some pins have specific functions as listed below:

1. Serial Pins 0 (Rx) and 1 (Tx): Rx and Tx pins are used to receive and transmit TTL serial data. They are connected with the corresponding ATmega328P USB

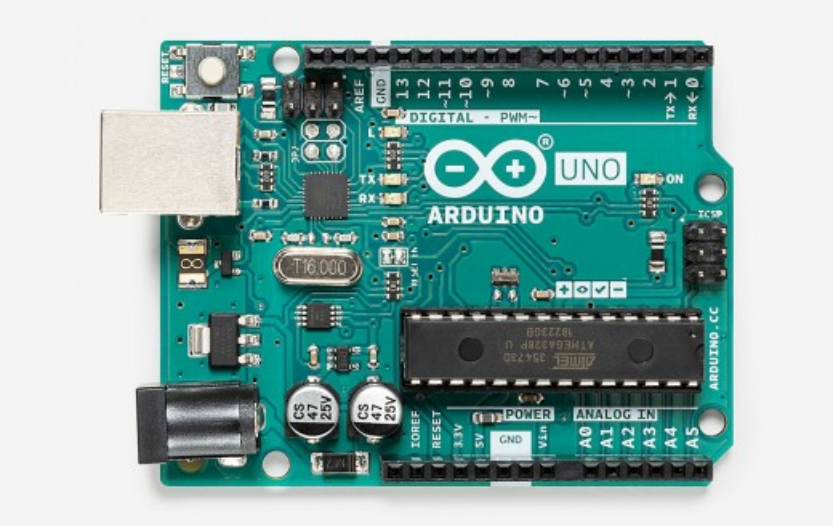


Figure 3.1: Adruino Uno Board

to TTL serial chip.

2. External Interrupt Pins 2 and 3: These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.

3. PWM Pins 3, 5, 6, 9 and 11: These pins provide an 8-bit PWM output by using analogWrite() function.

4. SPI Pins 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK): These pins are used for SPI communication.

5. In-built LED Pin 13: This pin is connected with an built-in LED, when pin 13 is HIGH – LED is on and when pin 13 is LOW, its off.

Along with 14 Digital pins, there are 6 analog input pins, each of which provide 10 bits of resolution, i.e. 1024 different values. They measure from 0 to 5 volts but this limit can be increased by using AREF pin with analog Reference() function. Analog pin 4 (SDA) and pin 5 (SCA) also used for TWI communication using Wire library. Arduino Uno has a couple of other pins as explained below:

1. AREF: Used to provide reference voltage for analog inputs with analogReference() function.

2. Reset Pin: Making this pin LOW, resets the microcontroller.

3.1.3 Communication

Arduino can be used to communicate with a computer, another Arduino board or other microcontrollers. The ATmega328P microcontroller provides UART TTL (5V) serial communication which can be done using digital pin 0 (Rx) and digital pin 1 (Tx). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The ATmega16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. There are two RX and TX LEDs on the arduino board which will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (not for serial communication on pins 0 and 1). A Software Serial library allows for serial communication on any of the Uno's digital pins. The ATmega328P also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus.

3.1.4 Arduino Uno to ATmega328 Pin Mapping

When ATmega328 chip is used in place of Arduino Uno, or vice versa, the image below shows the pin mapping between the two.

3.1.5 Programming Arduino

Once arduino IDE is installed on the computer, connect the board with computer using USB cable. Now open the arduino IDE and choose the correct board by selecting Tools¿Boards¿Arduino/Genuino Uno, and choose the correct Port by selecting Tools¿Port. Arduino Uno is programmed using Arduino programming language based on Wiring. To get it started with Arduino Uno board and blink the built-in LED, load the example code by selecting Files¿Examples¿Basics¿Blink. Once the example code (also shown below) is loaded into your IDE, click on the 'upload' button given

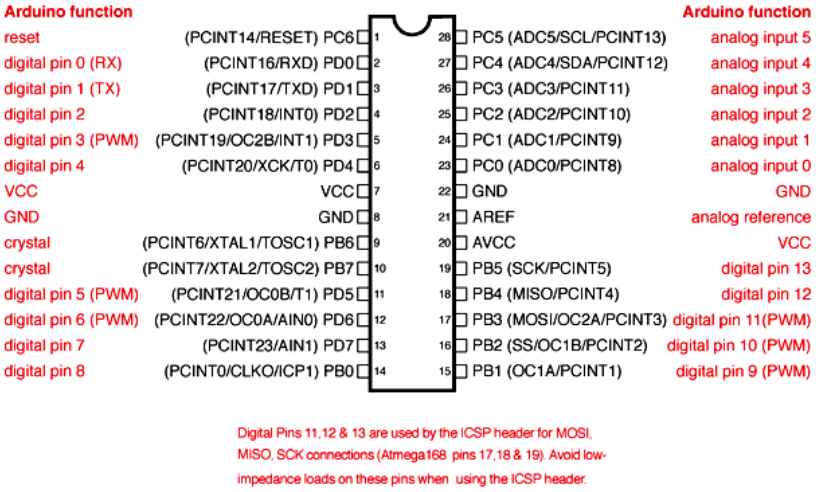


Figure 3.2: Adruino Uno to ATmega358 Pin Mapping

on the top bar. Once the upload is finished, you should see the Arduino's built-in LED blinking.

3.2 SIM800A Quad Band GSM/GPRS Module with RS232 Interface

3.2.1 Quick Overview

- Quad-band 850/900/1800/1900MHz.
- GPRS class 2/10.
- Input Voltage : 9V-12V DC.
- Control via AT commands (3GPP TS 27.007, 27.005 and SIMCOM enhanced
- AT command set).
- Configurable baud rate.
- Built-in SIM Card holder.

- Built in Network Status LED.

Figure 3.3: GSM/GPRS Module

3.2.2 Description

The SIM800A Quad-Band GSM/GPRS Module with RS232 Interface is a complete Quad-band GSM/GPRS solution in an LGA(Land grid array) type which can be embedded in the customer applications. SIM800A support Quadband 850/900/1800/1900 MHz, it can transmit Voice, SMS and data information with low power consumption.With a tiny size of 100 x 53 x 15 mm, it can fit into slim and compact demands of custom design. Featuring and Embedded AT, it allows total cost savings and fast time-to-market for customer applications. The SIM800A modem has a SIM800A GSM chip and RS232 interface while enables easy connection with the computer or laptop using the USB to the Serial connector or to the micro-controller using the RS232 to TTL converter. Once you connect the SIM800A modem using the USB to

RS232 connector, you need to find the correct COM port from the Device Manager of the USB to Serial Adapter. Then you can open Putty or any other terminal software and open a connection to that COM port at 9600 baud rate, which is the default baud rate of this modem.Once a serial connection is open through the computer or your micro-controlleryou can start sending the AT commands. When you send AT commands forexample "AT***r***" you should receive back a reply from the SIM800A modem saying "OK" or other response depending on the command sent.

3.2.3 Application

- Remote Data Monitor and Control.
- Water, gas and oil flow metering.
- AMR (automatic meter reading).
- Power station monitoring and control.
- Remote POS (point of sale) terminals.
- Traffic signals monitor and control.
- Fleet management.
- Power distribution network supervision.
- Central heating system supervision.
- Weather station data transmission.
- Hydro-logic data acquisition.
- Vending machine.
- Traffic info guidance.
- Parking meter and Taxi Monitor.
- Telecom equipment supervision (Mobile base station, microwave or optical

- relay station).

3.2.4 Features

- Quad-band 850/900/1800/1900MHz.
- GPRS class 2/10.
- Control via AT commands (3GPP TS 27.007, 27.005 and SIMCOM enhanced
- AT command set).
- High-Quality Product (Not hobby grade).
- 5V interface for direct communication with MCU kit.
- Configurable baud rate.
- Built-in SIM Card holder.
- Built-in Network Status LED.

3.3 16x2 LCD Display Module

16×2 LCD is named so because; it has 16 Columns and 2 Rows. There are a lot of combinations available like, 8×1, 8×2, 10×2, 16×1, etc. But the most used one is the 16*2 LCD, hence we are using it here. All the above mentioned LCD display will have 16 Pins and the programming approach is also the same and hence the choice is left to you.

3.3.1 Pin Description

- Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
- Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.

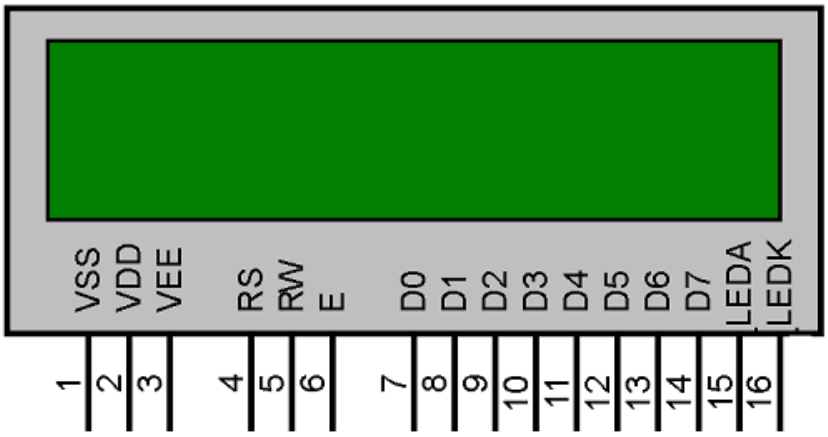


Figure 3.4: 16x2 LCD Display Pinout

- Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.

- Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1(0 = data mode, and 1 = command mode).

- Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).

- Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit constantly held high.

- Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.

- Pin15 (+ve pin of the LED): This pin is connected to +5V

- Pin 16 (-ve pin of the LED): This pin is connected to GND.

3.4 Some Sensor Modules

- SW-520D Tilt Sensor Module.
- Water Level Depth Detection Sensor Module.
- LM35 Temperature Sensor.

These Sensors modules are briefly discussed in next chapter.

Chapter 4

Project Sensor Modules

4.1 Tilt Sensor Switch Module

TILT SENSOR MODULE is a device used for knowing the planar movement. Although they are available in various types their basic function remains the same. Their function is to detect the plane shift from horizontal to vertical and sent of a signal when it happens. There are modules which could sense even "small plane shifts" but here we are going to discuss about simple contact type TILT SENSOR module.

4.1.1 Tilt Sensor Module Pin Configuration

TILT SENSOR MODULE is three pin devices as shown in figure. All three pins are compulsory for using the module.

Table 4.1: Configuration

Pin Name	**Description**
GND	Connected to ground
VCC	Connected to +5V
DO	Output of Tilt Sensor

Figure 4.1: Tilt Sensor

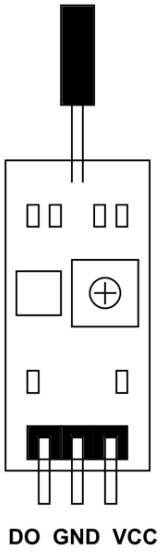


Figure 4.2: Pinout of Sensor

4.1.2 Tilt Sensor Module Features and Specifications

- Supply voltage: 3.3 V to 5V
- Output can directly connected to controller
- TTL level output
- Maximum output current : 15mA
- Can work on low voltages
- Maximum operating temperature: 0°C to + 80°C
- Easy interface
- Long life.

4.1.3 How to Use SW-520D Tilt Sensor Module

Before understanding how to use the module let us first understand the working of SW-520D tilt switch. For that consider the internal working of this tilt switch.

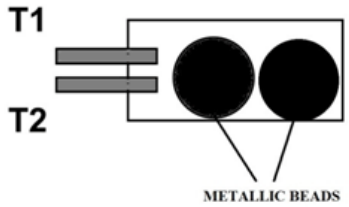


Figure 4.3: Tilt Sensor In Rest

As shown in figure inside SW-520D tilt sensor all we have is two metallic beads. And the output terminals are projected to the inside. When the tilt switch lies horizontally on the plane, the two metallic balls rest on the floor as shown in figure. Under this situation there will no contact between terminals T1 and T2. So T1 and T2 will be open.

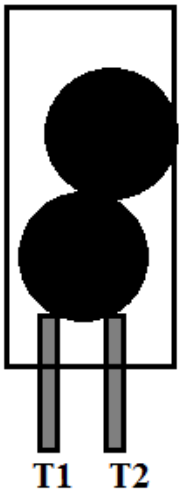


Figure 4.4: Tilt Sensor After some movement

When the tilt switch is moved from rest position, the metallic balls come to rest on the projected terminals. Because the ball is metallic, when they rest on terminals an electrical contact is formed between two terminals. So in this position T1 and T2 are short circuited.

When the tilt switch is moved to horizontal position again, ball leaves the contact to rest on floor as before. With that T1 and T2 contact breaks making them open circuit.

So SW-520D output terminals open circuit when the body lies horizontal and is short circuit when body is vertical. Hence using this tilt switch we can detect which plane the body is lying.

This tilt switch contact will have noise and cannot be connected to controllers directly. So we will connect this tilt switch to a simple op-amp comparator circuit to make a sensor module

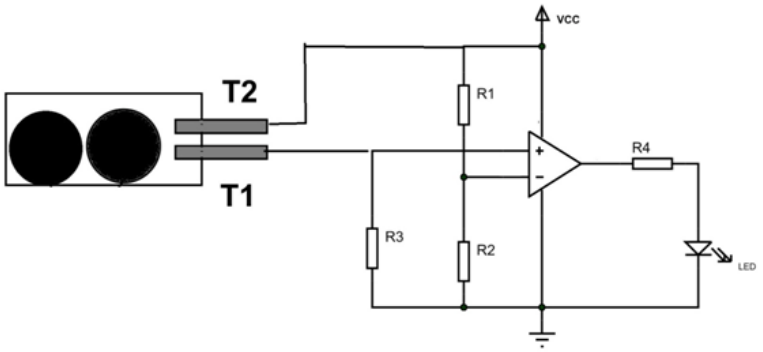


Figure 4.5: Tilt Sensor Circuit

Simplified tilt sensor module is shown in figure. Under normal situations T1 and T2 will be open and the entire VCC appears across it. So voltage at positive terminal of comparator will be zero. So the output of comparator will be low. At this time LED will be OFF.

When the tilt sensor is moved vertical, T1 and T2 will short circuit. At this time entire VCC appears across R3. At this time voltage at positive terminal of comparator will be VCC. With HIGH at positive terminal output of comparator will be HIGH. At this time LED will be ON.

This module output can be given to controller or other systems for using appropriately.

4.2 Water Level Depth Detection Sensor Module

Water sensor brick is designed for water detection, which can be widely used in sensing the rainfall, water level, even the liquate leakage.
This item can judge the water level through with a series of exposed parallel wires stitch to measure the water droplet/water size.This item can easily change the water size to analog signal, and output analog value can directly be used in the program function, then to achieve the function of water level alarm. This item have low power consumption, and high sensitivity, which are the biggest characteristics of this module. This item can be compatible with Arduino UNO,Arduino mega2560,Arduino ADK etc.

Figure 4.6: Water Level Depth Detection Sensor

4.2.1 Pin Definition

- ”S” stand for signal input
- ”+” stand for power supply
- ”-” stand for GND

4.2.2 Features of Sensor

- Working voltage: 5V
- Working Current: less than 20ma
- Interface: Analog
- Width of detection: 40mm x 6mm
- Working Temperature:10˚c to 0˚c
- Weight: 3g
- Size: 65mm x 20mm x 8mm
- Arduino compatible interface
- Low power consumption
- High sensitivity
- Output voltage signal: 0 to 4.2V

4.2.3 Application

- Rainfall detecting
- Liquid leakage
- Tank overflow detector

4.3 LM35 Precision Centigrade Temperature Sensors

The LM35 series are precision integrated-circuit temperature devices with an output voltage linearlyproportional 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 $\pm\frac{1}{4}$°C at room temperature and $\pm\frac{3}{4}$°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 A 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.

4.3.1 Features

- Calibrated Directly in Celsius (Centigrade)
- Linear + 10-mV/°C Scale Factor
- 0.5°C Ensured Accuracy (at 25°C)
- Rated for Full 55°C to 150°C Range
- Suitable for Remote Applications
- Low-Cost Due to Wafer-Level Trimming

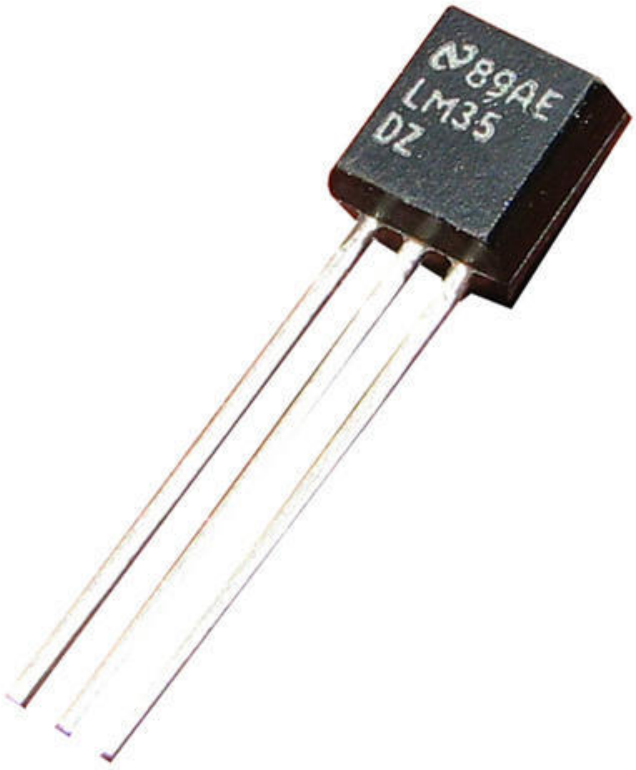


Figure 4.7: LM35 Temperature Sensor

- Operates From 4 V to 30 V
- Less Than 60-A Current Drain
- Low Self-Heating, 0.08°C in Still Air
- Non-Linearity Only $\pm\frac{1}{4}$°C Typical
- Low-Impedance Output, 0.1 for 1-mA Load

4.3.2 Pin-out And Pin specification

- Vcc: Input voltage is +5V for typical applications.
- Analog Out: There will be increase in 10mV for raise of every 1°C. Can range from -1V(-55°C) to 6V(150°C)
- Ground: Connected to ground of circuit

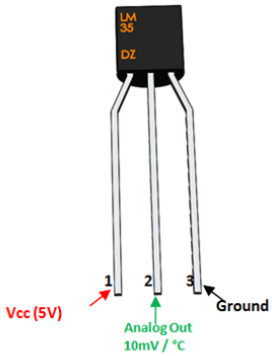
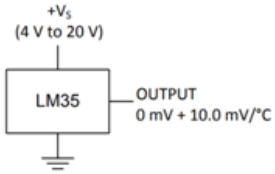


Figure 4.8: LM35 Temperature Sensor Pinout

4.3.3 How to use LM35 Temperature Sensor

LM35 is a precession Integrated circuit Temperature sensor, whose output voltage varies, based on the temperature around it. It is a small and cheap IC which can be used to measure temperature anywhere between -55°C to 150°C. It can easily be interfaced with any Microcontroller that has ADC function or any development platform like Arduino. Power the IC by applying a regulated voltage like +5V (VS) to the input pin and connected the ground pin to the ground of the circuit. Now, you can measure the temperate in form of voltage as shown below.



If the temperature is 0°C, then the output voltage will also be 0V. There will be rise of 0.01V (10mV) for every degree Celsius rise in temperature. The voltage can converted into temperature using the below formulae.

$$V_(out) = 10mv/^\circ C \times T$$

where,

- Vout is L35 output vtg.
- T is temprature in °C

4.3.4 LM35 Temperature Sensor Applications

- Measuring temperature of a particular environment
- Providing thermal shutdown for a circuit/component
- Monitoring Battery Temperature
- Measuring Temperatures for HVAC applications.

Chapter 5

Software Required for Coding

5.1 Arduino IDE

5.1.1 Introduction

The Arduino is a fantastic single-board microcontroller solution for many DIY projects, and, in this blog, we will look at the Integrated Development Environment, or IDE, that is used to program it.

First, you must download the IDE and install it. Start by visiting Arduino's software page. The IDE is available for most common operating systems, including Windows, Mac OS X, and Linux, so be sure to download the correct version for your OS.

The Arduino IDE is incredibly minimalist, yet it provides a near-complete environment for most Arduino-based projects. The top menu bar has the standard options, including "File" (new, load save, etc.), "Edit" (font, copy, paste, etc.), "Sketch" (for compiling and programming), "Tools" (useful options for testing projects), and "Help". The middle section of the IDE is a simple text editor that where you can enter the program code. The bottom section of the IDE is dedicated to an output window that is used to see the status of the compilation, how much memory has been used, any errors that were found in the program, and various other useful messages.

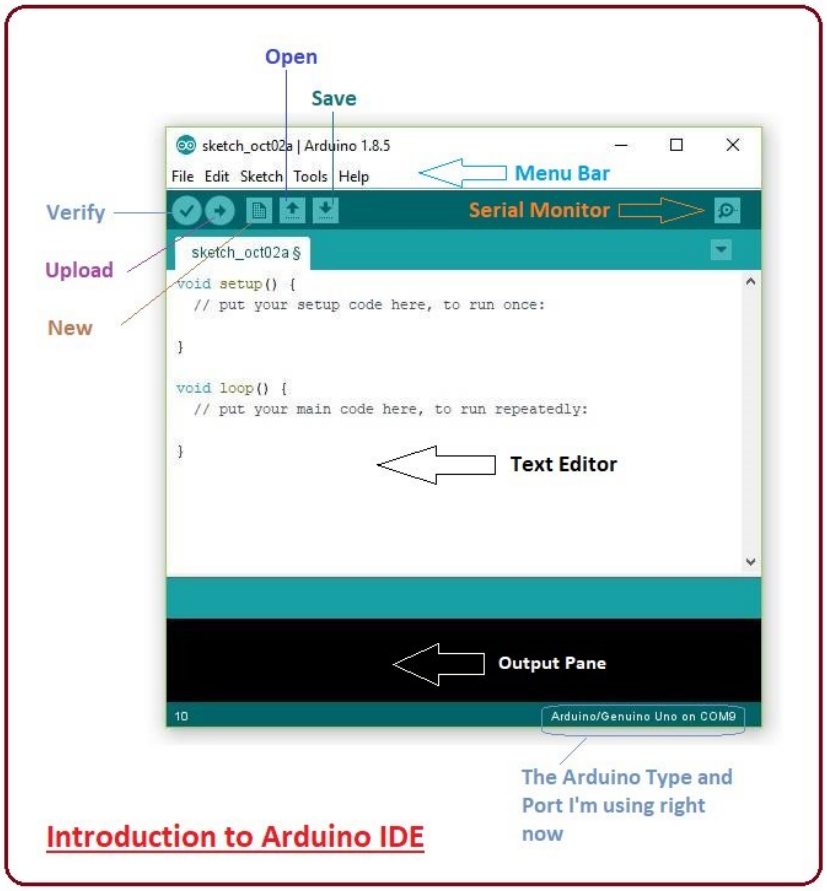


Figure 5.1: The Arduino IDE in its default state

Projects made using the Arduino are called sketches, and such sketches are usually written in a cut-down version of C++ (a number of C++ features are not included). Because programming a microcontroller is somewhat different from programming a computer, there are a number of device-specific libraries (e.g., changing pin modes, output data on pins, reading analog values, and timers). This sometimes confuses users who think Arduino is programmed in an "Arduino language." However, the Arduino is, in fact, programmed in C++. It just uses unique libraries for the device.

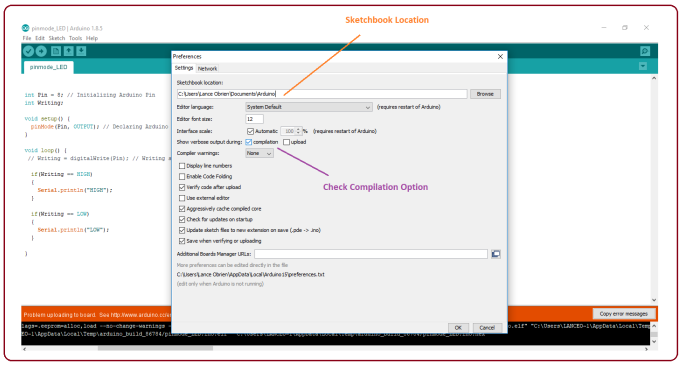
The bar appearing on the top is called Menu Bar that comes with five different options as follow

- File - You can open a new window for writing the code or open an existing

one. Following table shows the number of further subdivisions the file option is categorized into.

File	
New	This is used to open new text editor window to write your code
Open	Used for opening the existing written code
Open Recent	The option reserved for opening recently closed program
Sketchbook	It stores the list of codes you have written for your project
Examples	Default examples already stored in the IDE software
Close	Used for closing the main screen window of recent tab. If two tabs are open, it will ask you again as you aim to close the second tab
Save	It is used for saving the recent program
Save as	It will allow you to save the recent program in your desired folder
Page setup	Page setup is used for modifying the page with portrait and landscape options. Some default page options are already given from which you can select the page you intend to work on
Print	It is used for printing purpose and will send the command to the printer
Preferences	It is page with number of preferences you aim to setup for your text editor page
Quit	It will quit the whole software all at once

As you go to the preference section and check the compilation section, the Output Pane will show the code compilation as you click the upload button.



And at the end of compilation, it will show you the hex file it has generated for the recent sketch that will send to the Arduino Board for the specific task you aim to achieve.

- Edit - Used for copying and pasting the code with further modification for font
- Sketch - For compiling and programming

- Tools - Mainly used for testing projects. The Programmer section in this panel is used for burning a bootloader to the new microcontroller.

- Help - In case you are feeling skeptical about software, complete help is available from getting started to troubleshooting.

5.1.2 The Six Buttons

While more advanced projects will take advantage of the built-in tools in the IDE, most projects will rely on the six buttons found below the menu bar.

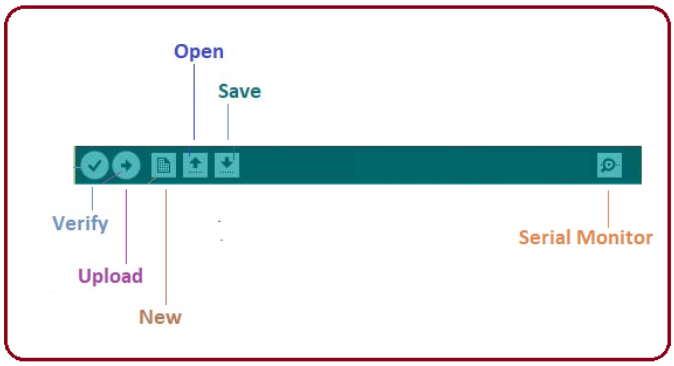


Figure 5.2: The button bar

1. The check mark is used to verify your code. Click this once you have written your code.

2. The arrow uploads your code to the Arduino to run.

3. The dotted paper will create a new file.

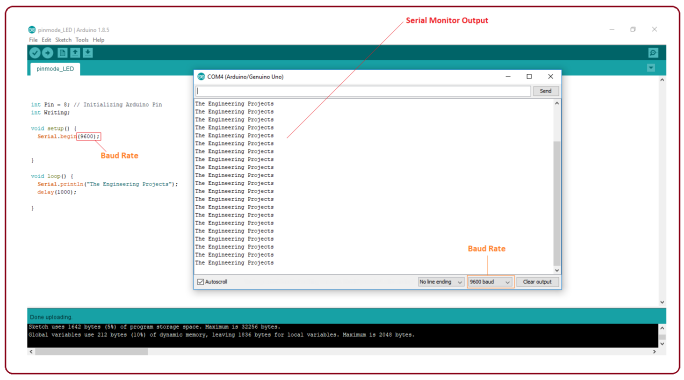
4. The upward arrow is used to open an existing Arduino project.

5. The downward arrow is used to save the current file.

6. The far right button is a serial monitor, which is useful for sending data from the Arduino to the PC for debugging purposes.

The button appearing on the top right corner is a Serial Monitor - A separate pop-up window that acts as an independent terminal and plays a vital role for sending and receiving the Serial Data. You can also go to the Tools panel and select Serial Monitor, or pressing Ctrl+Shift+M all at once will open it instantly. The Serial Monitor will actually help to debug the written Sketches where you can get a hold of how your program is operating. Your Arduino Module should be connected to your computer by USB cable in order to activate the Serial Monitor.

You need to select the baud rate of the Arduino Board you are using right now. For my Arduino Uno Baud Rate is 9600, as you write the following code and click the Serial Monitor, the output will show as the image below.



5.1.3 Libraries

Libraries are very useful for adding the extra functionality into the Arduino Module. There is a list of libraries you can add by clicking the Sketch button in the menu bar and going to Include Library.

As you click the Include Library and Add the respective library it will on the top of the sketch with a include sign. Suppose, I Include the EEPROM library, it will appear on the text editor as

$$include < EEPROM.h >$$

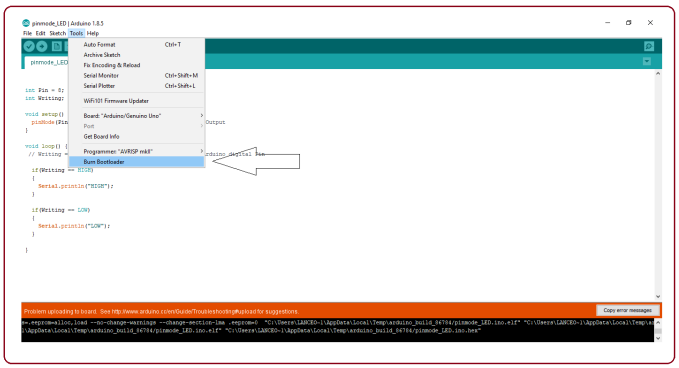
.

Most of the libraries are preinstalled and come with the Arduino software. However, you can also download them from the external sources.

5.1.4 Bootloader

As you go to the Tools section, you will find a bootloader at the end. It is very helpful to burn the code directly into the controller, setting you free from buying the external burner to burn the required code.

When you buy the new Arduino Module, the bootloader is already installed in-



side the controller. However, if you intend to buy a controller and put in the Arduino module, you need to burn the bootloader again inside the controller by going to the Tools section and selecting the burn bootloader.

There are plenty of other features available to consider on the IDE. But, having used many different types of microcontrollers and having been involved in multiple programming environments, it is shocking how simple the Arduino and its IDE is! In less than two minutes, you can get a simple C++ program uploaded onto the Arduino and have it running.

5.2 ThingSpeak Platform For Cloud Storage is Explained in next Chapter

Chapter 6

ThingSpeak Platform For Cloud Storage Facility

ThingSpeak is an IoT analytic platform service that allows you to aggregate, visualize, and analyze live data streams in the cloud. You can send data to ThingSpeak from your devices, create instant visualization of live data, and send alerts.

6.1 Description

ThingSpeak is a platform providing various services exclusively targeted for building IoT applications. It offers the capabilities of real-time data collection, visualizing the collected data in the form of charts, ability to create plugins and apps for collaborating with web services, social network and other APIs. We will consider each of these features in detail below. The core element of ThingSpeak is a 'ThingSpeak Channel'. A channel stores the data that we send to ThingSpeak and comprises of the below elements:

- 8 fields for storing data of any type - These can be used to store the data from a sensor or from an embedded device.
- 3 location fields - Can be used to store the latitude, longitude and the elevation.

These are very useful for tracking a moving device.

- 1 status field - A short message to describe the data stored in the channel.

To use ThingSpeak, we need to sign up and create a channel. Once we have a channel, we can send the data, allow ThingSpeak to process it and also retrieve the same. Let us start exploring ThingSpeak by signing up and setting up a channel.

6.2 ThingSpeak Key Features

ThingSpeak allows you to aggregate, visualize and analyze live data streams in the cloud. Some of the key capabilities of ThingSpeak include the ability to

- Easily configure devices to send data to ThingSpeak using popular IoT protocols.
- Visualize your sensor data in real-time.
- Aggregate data on-demand from third-party sources.
- Use the power of MATLAB to make sense of your IoT data.
- Run your IoT analytic automatically based on schedules or events.
- Prototype and build IoT systems without setting up servers or developing web software.
- Automatically act on your data and communicate using third-party services like Twilio or Twitter.

One of the key elements of an IoT system is an IoT service. ThingSpeak is one such application platform offering a wide variety of features. At the heart of ThingSpeak is a channel which can be used for storing and processing data collected from the 'things'. ThingSpeak also provides various apps for integration with web services, other APIs and social networks and provides the capability to create the

applications as plugins. It is a great platform with extensive possibilities to explore the integration of the Internet of Things.

Chapter 7

Working And Interfacing of System

7.1 Working

The bridge health monitoring system works on various factors like temperature, level, vibration and visibility. All the four factors are crucial for the health of bridge monitoring and three of them use same Arduino platform for its process. All the individual sensors sense the outer world condition the Arduino fetches this sensed data and process it and then using a GSM module the data is stored or displayed on a thing speak cloud server.

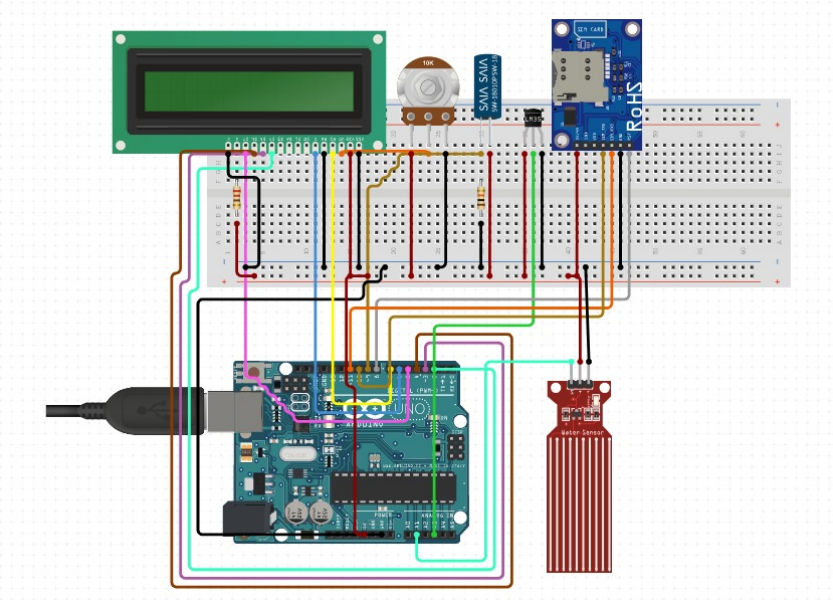


Figure 7.1: Interfacing Of Sensors and Display with Arduino

All the sensors require programming for the desired function. The programming of the system is done on the platform provided by Arduino which is called Arduino IDE. For receiving and sending data TCP/IP protocol is used. The display available in this system shows all the real time value of the different factors present in the system. Let's understand the working of each sensor working of the system when it is interfaced with Arduino using a common system circuit.

7.2 Interfacing Of Level Sensor with Arduino

An analog sensor is used in this system to measure water level around the bridge. Since the level of water under the bridge is an important factor in maintain its health thus it is important to measure water level. This sensor is placed at a particular position with a predefined initial value of level and a predefined maximum value over which the Arduino gives alert.

The water level sensor has 3 pins to connect:

- S(Signal): This pin is an analog output that will be connected to one of the analog inputs on your Arduino.
- + (VCC): This pin supplies power for the sensor. It is recommended to power the sensor with between 3.3V – 5V. Please note that the analog output will vary depending on what voltage is provided for the sensor.
- – (GND): This is a ground connection.

Following illustration shows the wiring

The level sensor when immersed in water produces a signal which is fetched by the Arduino and thus it process the data as per specified in the program.

The process begins with the declaration of the Arduino pins to which the sensor's + (VCC) and S (signal) pins are connected.

Next, we define a variablevalthat stores the current water level.

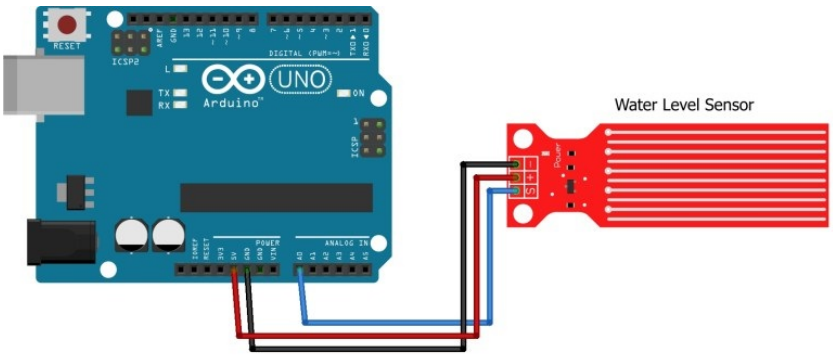


Figure 7.2: Interfacing Of Level Sensor with Arduino

Now in the Setup section, we first declare the power connection to the sensor as output, then we set it low so no power flows through the sensor initially. We also setup the serial monitor.

In the loop section, we call readSensor() function repeatedly at the interval of one second and print the returned value.

The readSensor() function is used to get the current water level. It turns the sensor ON, waits for 10 milliseconds, reads the analog value form sensor, turns the sensor OFF and then returns the analog value. And thus the water level is measured around the bridge.

7.3 Interfacing of Temperature Sensor with Arduino

LM35 can sense the temperature it is put around and transmit it to degrees Celsius. When the temperature rises above a predefined value then the Arduino connected with the sensor alerts the client via GSM module system.

The LM35 temperature sensor is a three pin device (VCC, OUT and GND) with an output voltage linearly related to Centigrade temperature. Since the LM35 output varies with dependent to the temperature, we need an ADC (Analog-to-Digital Converter) module to measure this voltage. The Arduino UNO board microcontroller (ATmega328P) has a 10-bit ADC module and a built-in fixed voltage reference of

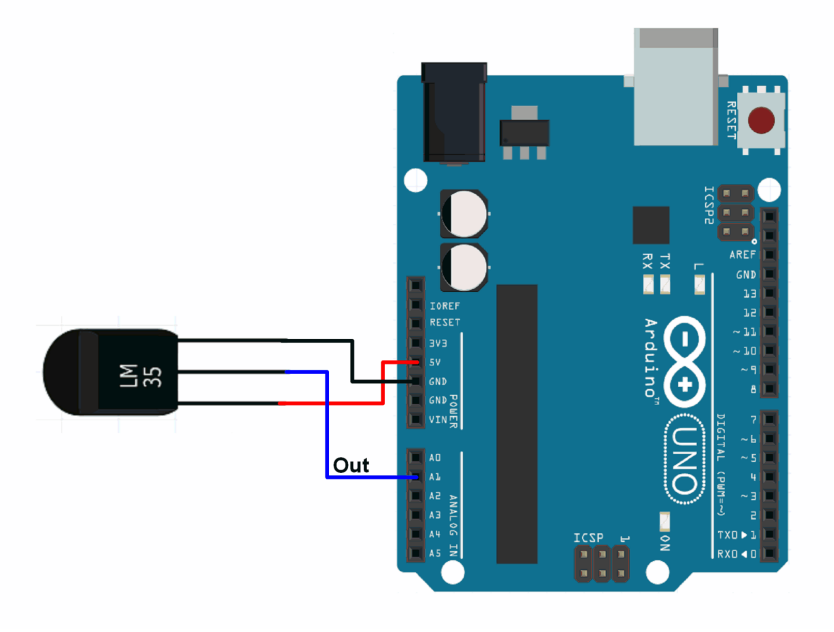


Figure 7.3: Interfacing Of Temperature Sensor with Arduino

1.1V. With the fixed voltage reference we get approximately an exact result.

7.4 Interfacing of Vibration Sensor(Tilt Sensor) with Arduino

As the connection is done as shown in diagram below and the upload do program is done the sensor is ready to function. The connection of LCD is same for all the sensors with Arduino as shown in LM35 sensor. When the sensor comes in contact of vibration it produces a signal which is fetched by Arduino.

If the sensed value is above the specified value the vibration sensor SW-420 Comes with breakout board that includes comparator LM 393 and Adjustable on board potentiometer for sensitivity threshold selection, and signal indication LED. This sensor module produce logic states depends 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

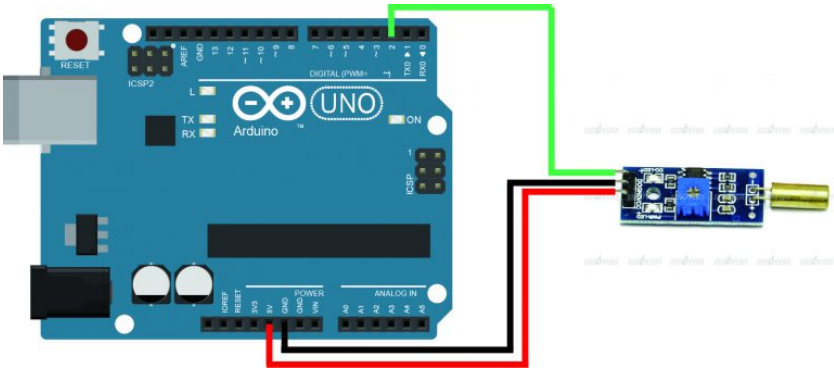


Figure 7.4: Interfacing Of Vibration Sensor with Arduino

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 Arduino board.

7.5 Interfacing of GSM Module with Arduino

GSM module is a hardware which is used to transmit and save data on cloud and also to send an alert message to the authority

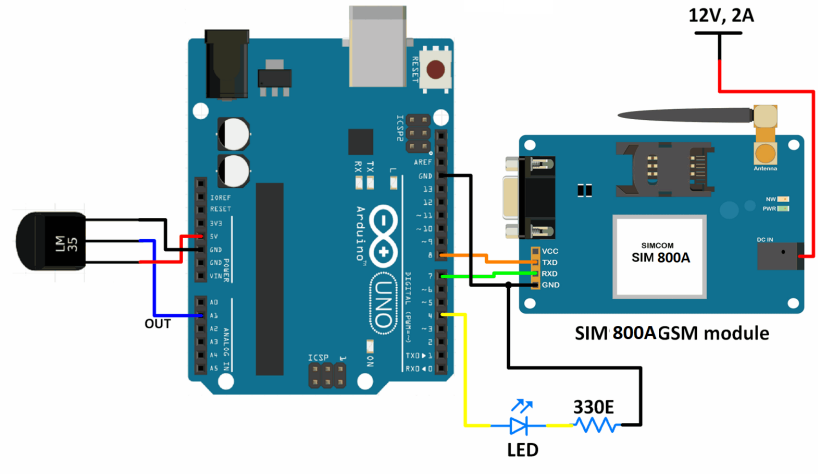


Figure 7.5: Interfacing Of GSM Module with Arduino

Turn ON the GSM module by providing 5V and GND.Insert the SIM card to GSM module and lock it.Initially blinking rate of network LED will be high. After

sometime observe the blinking rate of 'network LED' (GSM module will take some time to establish connection with mobile network).Once the connection is established successfully, the network LED will blink continuously for every 3 seconds.Even we can check the connection establishment of GSM module with mobile by making a call to the number of the SIM. If we hear a ring back, the GSM module has successfully established network connection.

Whenever the specified values of temperature,vibration and level goes above specified level the gsm modules sends an alert message to authority.

7.6 Interfacing of LCD with Arduino

The process of controlling the display involves putting the data that form the image of what you want to display into the data registers, then putting instructions in the instruction register. TheLiquid Crystal Librarysimplifies this for you so you don't need to know the low-level instructions.

The LCDs have a parallel interface, meaning that the microcontroller has to manipulate several interface pins at once to control the display. The interface consists of the following pins: A register select (RS) pinthat controls where in the LCD's memory you're writing data to. You can select either the data register, which holds what goes on the screen, or an instruction register, which is where the LCD's controller looks for instructions on what to do next.

A Read/Write (R/W) pinthat selects reading mode or writing mode

AnEnable pinthat enables writing to the registers

8 data pins (D0 -D7). The states of these pins (high or low) are the bits that you're writing to a register when you write, or the values you're reading when you read.

There's also adisplay constrast pin (Vo),power supply pins (+5V and Gnd)andLED Backlight (Bklt+ andBKlt-)pins that you can use to power the LCD, control the display contrast, and turn on and off the LED backlight, respectively.

The values of temperature, vibration and level is displayed on the screen is speci-

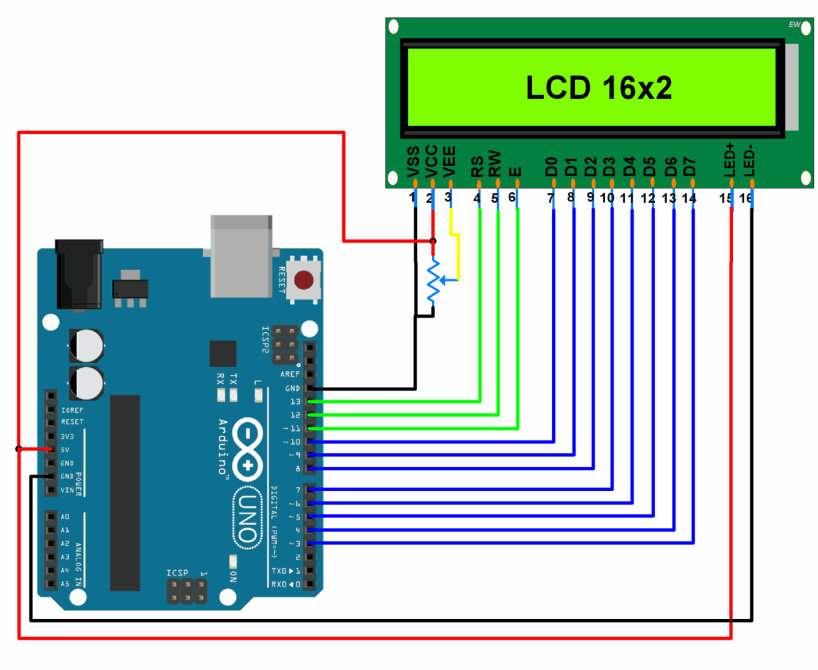


Figure 7.6: Interfacing Of 16x2 LCD Display with Arduino

fied in the program of the individual sensor.

Results from Pre-analysis

7.7 Vibration Data On Cloud

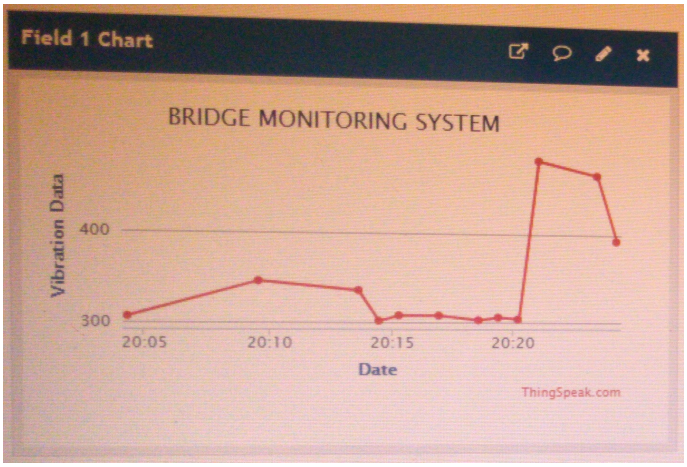


Figure 7.7: Vibration Data In Normal Condition

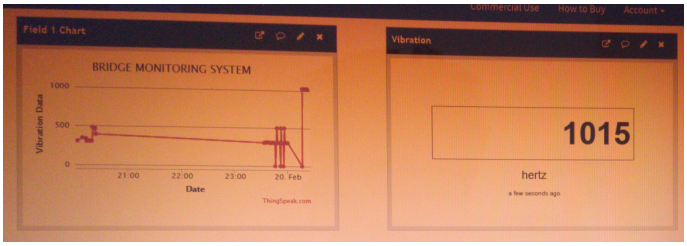


Figure 7.8: Vibration Data After Vibration

7.8 Water Level Data On Cloud

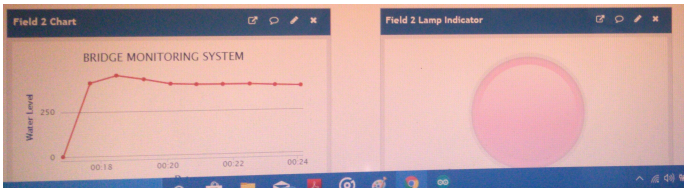


Figure 7.9: Level Of Water In Normal

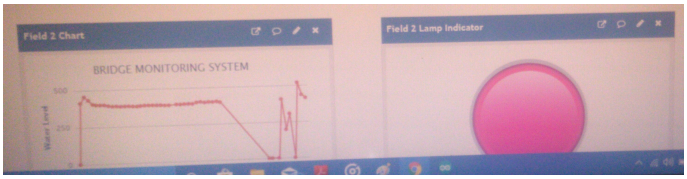


Figure 7.10: Level After Water Level Increases

7.9 Temperature Data On Cloud

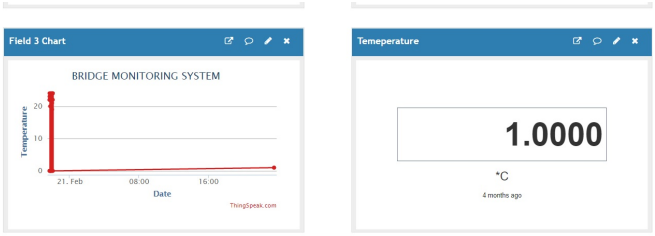


Figure 7.11: Temperature Data Before

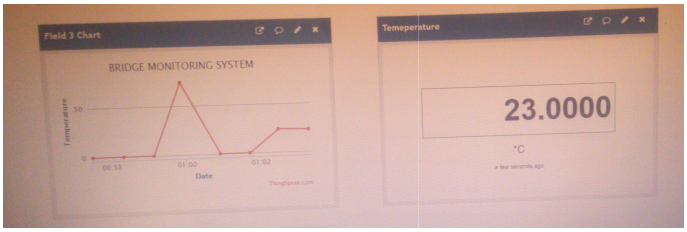


Figure 7.12: Temperature Data After

Conclusions And Scope For Future

Conclusions

- Even in developed nations like USA, it has been found that more than one out of every four bridges are structurally deficient.
- This wireless technology could avert the kind of bridge collapse that killed 13 and injured 145 along Minneapolis on Aug. 1, 2007 at one-hundredth the cost of current wired systems.
- This system can help in monitoring the bridge in an efficient, cost effective and reliable manner.
- The immediacy, low cost, low energy and compact size add up to a revolution in bridge safety monitoring, providing a heightened level of early warning capability.

Future Work

- Web camera can be fitted so that the density of the vehicles can be know by the people who enter the bridge. This can be done using MATLAB.(By replacing GSM and Arduino we put ARM 7 Processor and Zigbee Module)
- High Sensitive sensors like IR and UV sensor can be interface to detect collision.
- This project can be implemented in a two-way road bridges. But the the complexity of the project increases.

Bibliography

1. Gethin Roberts , Xiaolin Meng , Michele Meo , Alan Dodson , Emily Cosser , Emanuela Iuliano, Alan Morris (2003), A Remote Bridge Health Monitoring System Using Computational Simulation And Gps Sensor Data.

2. Ms.Arohi. D. Sonawane, Ms.Pooja. P. Vichare, Mr.Shubham. S. Patil, and Mr.Nitin. P. Chavande, "Bridge Monitoring System Using iot", Journal of Advances in Electrical Devices Volume 3, Issue 2, 2018.

3. Ashwini R, Sneha Shivan and Mesta, Varsha A U, Ravichandran G, Haritha K, Sivaraman Bridge Monitoring System Using Wireless Networks IJARJJE-ISSN Vol-2 Issue-5 2017.

4. Christian Lazo, Paulo Gallardo, and Sandra Cespedes Bridge Structural Health Monitoring System Supported By the Internet of Things ieee colcom 2015.

5. Jin-Lian Lee, Yaw-Yauan Tyan, Ming-Hui Wen, Yun-Wu Wu Development of an IOT- based Bridge Safety Monitoring System International Conference on Applied System Innovation IEEE-ICASI 2017.

6. Miss. Pooja Krishnath Patil, Prof. Dr S. R. Patil Structural Health Monitoring System using WSN for Bridges International Conference on Intelligent Computing and Control Systems ICICCS 2017.

7. Christian Lazo, Paulo Gallardo, and Sandra Cespedes, "A Bridge Structural Health Monitoring System Supported by the Internet of Things", IEEE COLCOM, 2015.