CHAPTER - 1
Introduction

CHAPTER – 1

Introduction

1.1 Introduction

The World Wide Web has made it possible to send a lot of data from one side of the world to the other side in almost no time. The use of the Internet for real-time interaction of the remote controlling and monitoring of the plants would give us many advantages. This technology cannot only be used in the industry, but also in the field of medicine, education, etc. Although all this looks promisingly two main problems should be faced before the web based control and monitoring can be implemented. The first one is the aspect of time delay, which can lead to irregular data transmission and data loss. In the worst-case this can make the whole system unstable. The other one is the problem of security. When malicious hackers can grant access to a system the consequences can be catastrophic. Other problems concern the distance or logistics. If something goes wrong with the system, a lot of time and preparation can be needed before somebody can intervene to requirement specifications and system implementation. This report is organized as follows. In the section 2 the definition of web based control and monitoring will be given as well as a short history about it. The section 3 describes design issues, which should be considered when developing these kinds of systems. The security problems and measures are investigated in the section 4. The section 5 gives a list of advantages when using the web based control and monitoring. This section provides also some examples. Finally, the section 6 gives some concluding remarks.

1.2 History of Web Based

As a basis for the possible next generation of control systems, the concept of the Internet based process control has been introduced in recent years. To date, most research work on the Internet based process control has resulted in small-scale demonstrations like Sun Microsystems and Cyberonix, Foxboro, and Valmet. Most of them were developed in Java. Additionally, the OPC (Open Process Control) Foundation (1998) is working on supporting XML within Visual Studio so that the Internet based process control using XML can become a reality. Intuitive Technology Corp. (1999) has provided web@aGlance for feeding real-time data to a Java graphics console. In addition, Invensys plc is presently collaborating with Oxford University's Department of Engineering 2 Science to develop web-based instruments. Some companies try to produce Internet control systems as a control device. Some researchers in this area, from higher education institutions, focus on developing web-based virtual control laboratories for distance learning purposes. A remotely located user has possibility to conduct experiments in the laboratory via the Internet. The students can continuously access their hypothetical experiments setup. The major advantage of this virtual laboratory is the minimal cost needed to set up a laboratory, as it only requires a robust communication network (Yeung K. and Huang J., 2001). The International Federation of Control (IFAC) has held the first workshop on Internet Based Control Education in Spain in 2002. The Scada on Web system funded by the European Council targets Internet based protocols enabling the monitoring and optimisation of the process via the web. It is hoped that the specific web based approach towards the development of an online framework will eventually result in the adoption of Scada on Web as an industry standard for transporting large volume of process data online (Yang et al., 2002).

1.3 Web Based Monitoring and Control

The web based monitoring and control (WBMC) can be described as the whole of operations performed to control or monitor a system situated in a closed network. WBMC is based on existing technologies, programming languages as Perl and Java and specifications as HTTP, TCP and UDP.

1.4 Structure of Our Report

The thesis is divided into 7 chapters in total. All the research conducted by us while developing this web based monitoring system is divided and sequentially presented in these chapters. The chapters are arranged in such a way that all the information about power, power measurement, gear motor, working principle of web based industry monitoring system, methodology etc. It can be found sequentially for better understanding of the project. The aim was to discuss the research work by providing as much information as possible and through the results obtained from analysis of each step. The first chapter is named as „Introduction" and it is for the introduction purpose of this thesis where motivation behind the research work is mentioned and benefit of web system monitoring and control. In order for constructing a device capable of measuring real power, it was essential to have clear conception of power and its working principle. The second chapter named as „Concept of Real Power and arduino" comprises information regarding real power, different types of sensor and their working principle. The third chapter contains information regarding methodology and practical implementation of the method. Working procedure and steps of construction is explained using figures and functional block diagram in this chapter. The next chapter comprises schematic diagram of the design and some images that has been developed during the research. Chapter 5 includes discussion about the microcontroller programming where the sequential process of the microcontroller is explained through a flow chart and chapter 6 is comprised of the results and performance analysis. In chapter 7, the total expenditure of the construction is estimated and also include advantages/disadvantages of our web based industry monitoring system is discussed along with the possibilities of future advancements.

CHAPTER – 2

Concept of Power and Web System

2.1 Introduction

2.2 Power

2.3 Working Principle

2.4 Microcontroller

CHAPTER – 2

Concept of Power and Speed of motor

2.1 Introduction

In this chapter, elaborate discussion on power consumption and web system is given. The equipment of power measurement as well as its general classification and construction/mechanism is also discussed in this chapter. Detailed overview of microcontroller based industry monitoring system and working principle is also given at the end.

2.2 Power

There is AC power supply used which is converted of dc power supplies used in. Electric motors are efficient because they require little maintenance and they are not very noisy. Pneumatics acts as a power supply by making the use of compressed air with different variety of sizes but pneumatic can't rely only on pneumatics as their power supply, they require having another source of power like gasoline or even electric to be able to provide the compressed air. The disadvantages of this type of power are that they are very noisy, heavier and larger than any other power source and needs a lot of maintenance. The third type of power supply is hydraulic which also needs another type of supply like pneumatics in order to move into its components. The disadvantage of this type is that it requires a lot of maintenance of the tubes used.

Figure 2.1: Lithium Battery with the Charger Kit

There is important sources of electric power which are ac power convert in dc power. Solar cells produce electricity by making use of power from sunlight, but in order to produce a decent amount of power; a su-cient amount of solar cells connected in a solar panel has to be used. In this project, it has used adaptor (9V) as a main power source.

2.3 Working Principle

In this project web based industrial monitoring system where we used five sensor. We used a conveyor belt, a motor, and a DC power supply. We also used a switch for operating conveyor belt separately and all other source we connected from arduino. Arduino connected to the dc source by 12v the system is on and work all devices. We connected the input sensor pin in 5v arduino and the ground pin connected GND1. In the normal condition the display show the result at temperature humidity, the normal gas level and sound level of normal condition. But when we produce smoke the gas level is changes, if we create noise the sound level is increases, when conveyor belt is on and the number of product passes through the conveyor belt it counted by IR sensor and it weight measured by load cell. All data we can see at the display. We use wifi module to send data at website for this at we create a mobile hotspot when our mobile hotspot is on the wifi module send data at website and we can easily monitoring all value from anywhere.

2.4 Arduino (Microcontroller)

A microcontroller is a self-contained system with peripherals, memory and a processor that can be used as an embedded system. Most programmable microcontrollers that are used today are embedded in other consumer products or machinery including, phones, peripherals, automobiles or other household appliances for computer system. Programmable microcontrollers contain general purpose input/output pins. The number of these pins varies depending on the microcontroller. Input output devices include solenoids, relays, LCD displays, sensors and switches for different types of data. Considering the RAM size, Flash size, number of input/output lines, speed and supply, we selected the ATMEGA328p AVR chip which is embedded in Arduino microcontroller board for our project work. Arduino is an open source embedded development platform which is easy-to-use. It comprises of Hardware boards and Software tools. It has 14 digital input/output pins of which 6 can be used as PWM outputs, 6 analog inputs, a 16MHz ceramic resonator, a Universal Serial Bus connection, a power jack, an ICSP header, and a reset button. Detail of Arduino is discussed in APENDIX-A.

CHAPTER - 3

Design and Methodology

CHAPTER - 3

Design & Methodology

3.1 Introduction

In this chapter we discussed about the design methodology and working principle of our developed web based industry monitoring system through the schematic diagram and functional block diagram.

3.2 Schematic Diagram

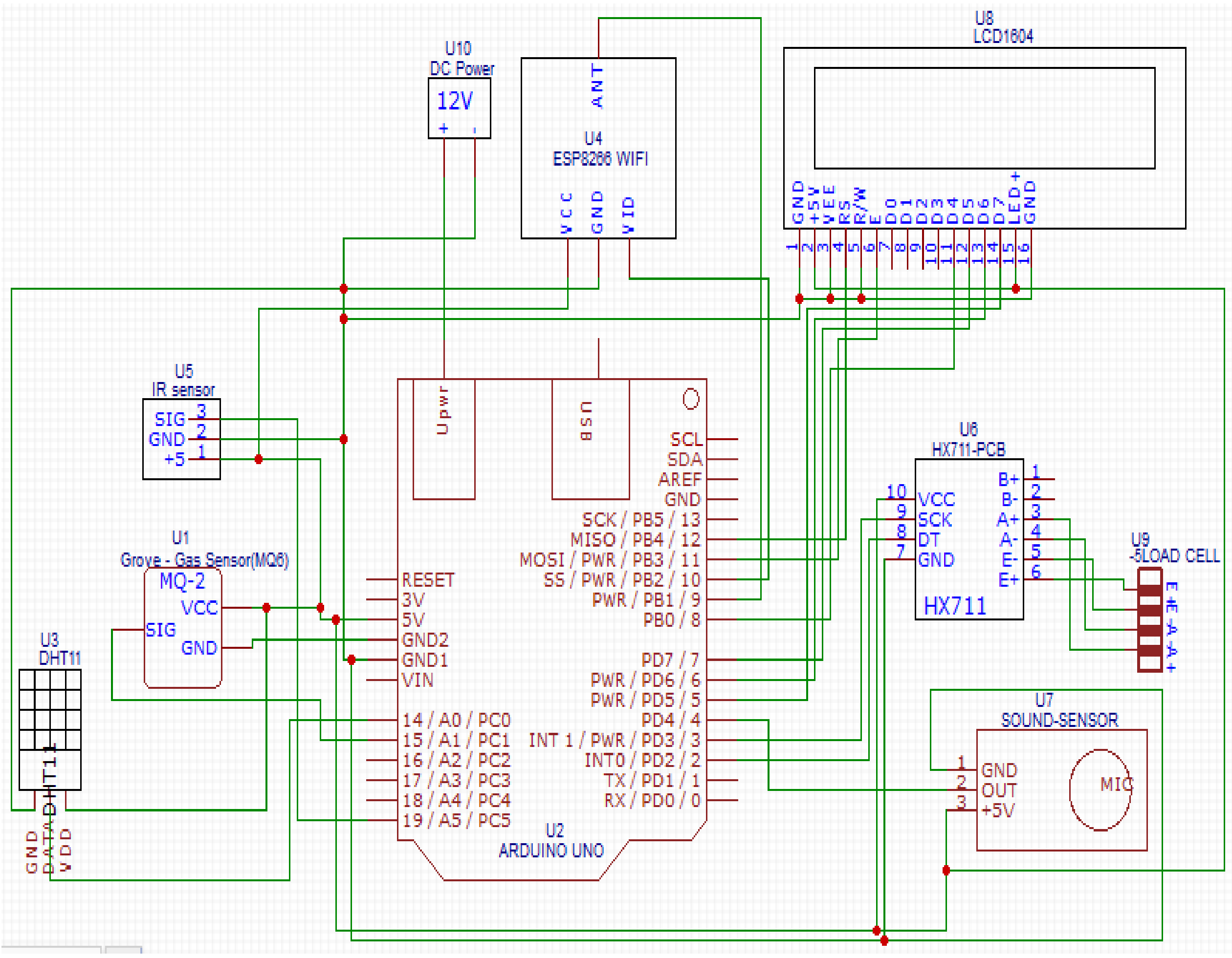

Figure 3.1: Schematic Diagram of Web Based Industry Monitoring System

3.3 Block Diagram

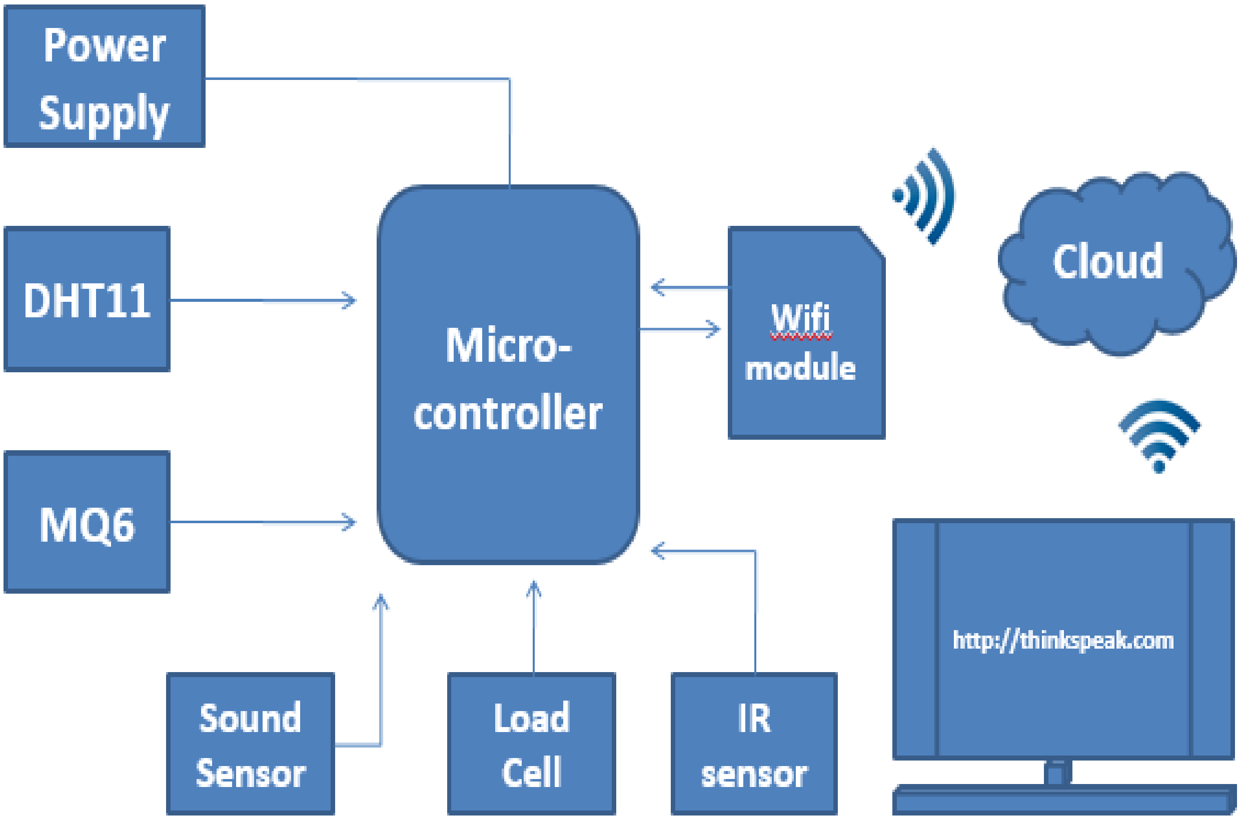

Figure 3.1: Block Diagram of Web Based Industry Monitoring System

CHAPTER - 4

Hardware Design

4.1 Introduction

4.2 Image of Developed Web Based Industry Monitoring

4.3 Image of Real Hardware

4.4 List of Used Components

CHAPTER - 4

Hardware Design

4.1 Introduction

For simulation and design purpose, two software named as Proteus Fritzing and other online tools were used. The schematic diagram is made using an online design tool named “Easyeda”. Images of the developed web based industry monitoring system is given to show the external and internal view.

4.2 Image of Developed Web Based Industry Monitoring System

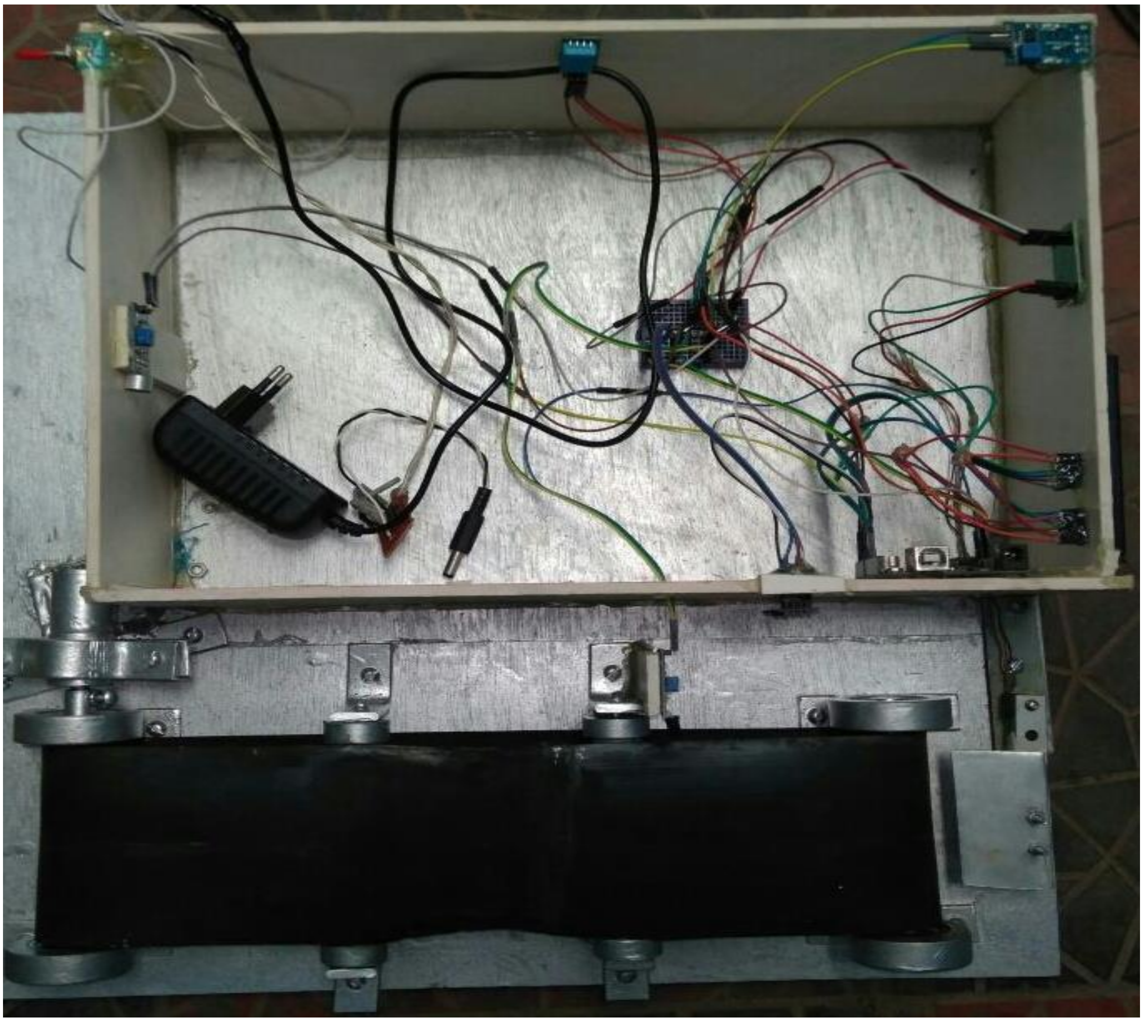

Figure 4.1: External View of Web Based Industry Monitoring System

4.3 Image of Real Hardware

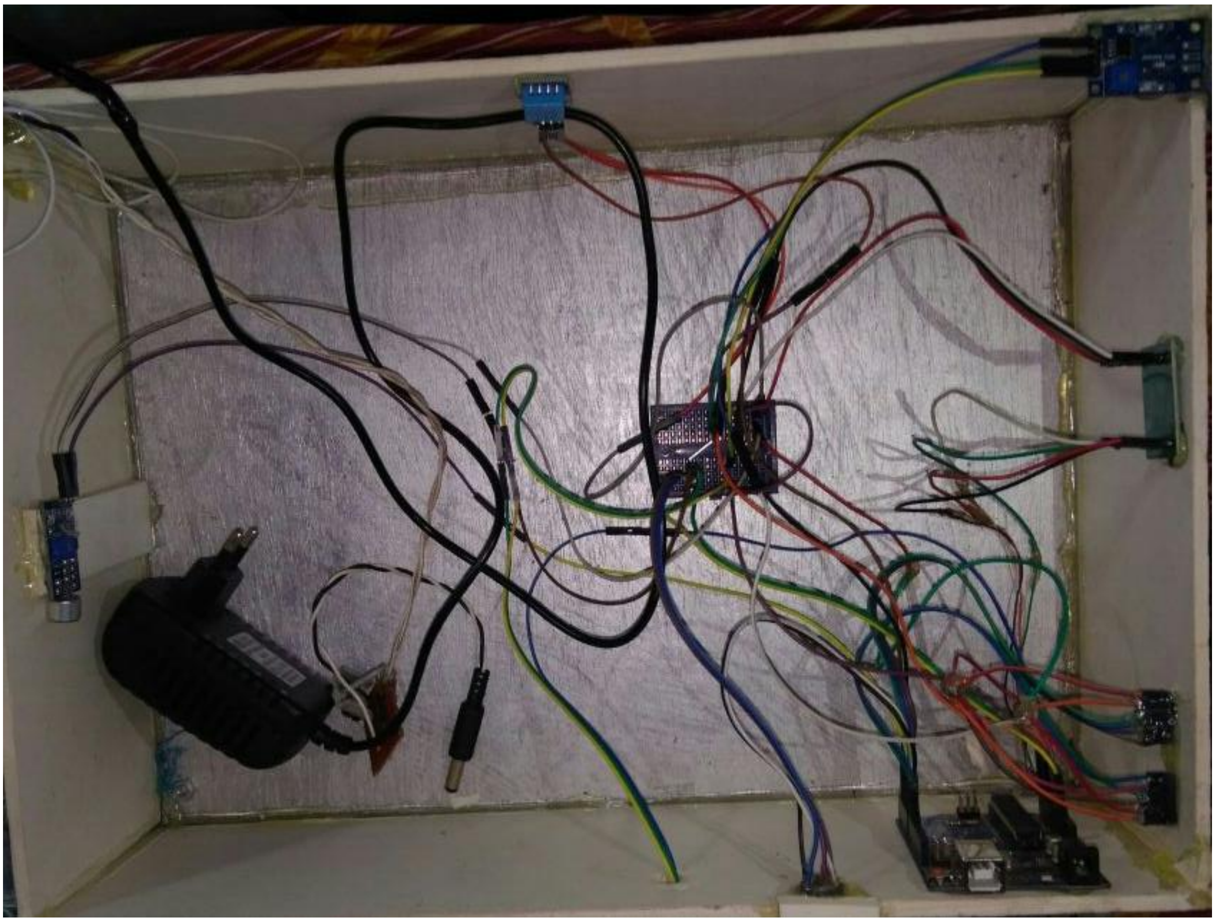

Figure 4.2: Internal view of the Web Based Industry Monitoring System

4.4 List of Used Components

1. Microcontroller/ Arduino-UNO
2. DH11
3. Sound Sensor
4. Load Cell
5. IR Sensor
6. Wi-Fi Module
7. Gear Motor
8. MQ6 Gas Sensor
9. HX711-PCB
10. LCD Display (16*4)
11. Conveyor Belt
12. DC Power Supply
13. Wire

1. Arduino

Figure 4.3: Microcontroller; Arduino UNO

Detail on Arduino is provided in APENDIX-A

2. DH11

Figure 4.4: DH11

Temperature and Humidity Sensors:

The DHT11 sensor for measuring temperature and humidity. The sensors is very popular for electronics hobbyists because it is very cheap but still providing great performance. Here are the main specifications is the DHT11 temperature range is from 0 to 50 degrees Celsius with +-2 degrees accuracy. Also the DHT22 sensor has better humidity measuring range, from 0 to 100% with 2-5% accuracy, while the DHT11 humidity range is from 20 to 80% with 5% accuracy.

There are two specification where the DHT11 is better than the DHT22. That’s the sampling rate which for the DHT11 is 1Hz or one reading every second, while the DHT22 sampling rate is 0,5Hz or one reading every two seconds and also the DHT11 has smaller body size. The operating voltage of both sensors is from 3 to 5 volts, while the max current used when measuring is 2.5mA.

DHT11 Working Principle

The sensor consist of a humidity sensing component, a NTC temperature sensor (or thermistor) and an IC on the back side of the sensor. For measuring humidity they use the humidity sensing component which has two electrodes with moisture holding substrate between them. So as the humidity changes, the conductivity of the substrate changes or the resistance between these electrodes changes. This change in resistance is measured and processed by the IC which makes it ready to be read by a microcontroller. On the other hand, for measuring temperature these sensors use a NTC temperature sensor or a thermistor. A thermistor is actually a variable resistor that changes its resistance with change of the temperature. These sensors are made by sintering of semi-conductive materials such as ceramics or polymers in order to provide larger changes in the resistance with just small changes in temperature. The term “NTC” means “Negative Temperature Coefficient”, which means that the resistance decreases with increase of the temperature.

3. Sound Sensor

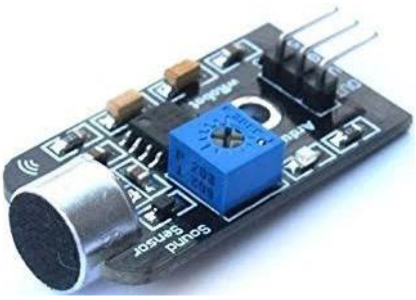

Figure 4.5: Sound Sensor

Sound Sensor can detect the sound intensity of the environment. The main component of the module is a simple microphone, which is based on the LM386 amplifier and an electret microphone. This module's output is analog and can be easily sampled and tested. Features: Easy to use-- 1. Provides analog output signal. 2. Easily integrates with Logic modules on the input side of Grove circuits 3. Specifications 4. Item Value 5. Operating Voltage Range 3.⅗ V 6. Operating Current (Vcc =5V) 4~5 mA. 7. Voltage Gain (V=6V, f=1 kHz) 26 dB 8. Microphone sensitivity (1 kHz) 52-48 dB 9. Microphone Impedance 2.2k Ohm 10. Microphone Frequency 16-20 kHz. Microphone S/N Radio 54 dB. This sound sensor is used to detect whether there's sound surround or not, please don't use the module to collect sound signal. For example, you can use it to make a sound control lamp, but not as a recording device.

4. Load Cell

Figure 4.6: Load Cell

A load cell is a sensor or a transducer that converts a load or force acting on it into an electronic signal. This electronic signal can be a voltage change, current change or frequency change depending on the type of load cell and circuitry used. There are many different kinds of load cells. We offer resistive load cells and capacitive load cells. Resistive load cells work on the principle of piezo-resistivity. When a load/force/stress is applied to the sensor, it changes its resistance. This change in resistance leads to a change in output voltage when a input voltage is applied. Capacitive load cells work on the principle of change of capacitance which is the ability of a system to hold a certain amount of charge when a voltage is applied to it. For common parallel plate capacitors, the capacitance is directly proportional to the amount of overlap of the plates and the dielectric between the plates and inversely proportional to the gap between the plates. A load cell is made by using an elastic member (with very highly repeatable deflection pattern) to which a number of strain gauges are attached. In this particular load cell shown to the right, there are a total of four strain gauges that are bonded to the upper and lower surfaces of the load cell. When the load is applied to the body of a resistive load cell as shown above, the elastic member, deflects as shown and creates a strain at those locations due to the stress applied. As a result, two of the strain gauges are in compression, whereas the other two are in tension.

5. IR Sensor

Figure 4.7: IR Sensor

This IR Proximity Sensor is a multipurpose infrared sensor which can be used for obstacle sensing, color detection, fire detection, line sensing, etc and also as an encoder sensor. The sensor provides a digital output. The sensor outputs a logic one (+5V) at the digital output when an object is placed in front of the sensor and a logic zero (0V), when there is no object in front of the sensor. An onboard LED is used to indicate the presence of an object. This digital output can be directly connected to an Arduino, Raspberry Pi, AVR, PIC, 8051 or any other microcontroller to read the sensor output. IR sensors are highly susceptible to ambient light and the IR sensor on this sensor is suitably covered to reduce effect of ambient light on the sensor. The sensor has a maximum range of around 40-50 cm indoors and around 15-20 cm outdoors. Features: 1. Can be used for obstacle sensing, color detection (between basic contrasting colors). 2. Comes with an easy to use digital output. 3. Can be used for wireless communication and sensing IR remote signals. 4. Sensor comes with ambient light protection.5. The sensor a hole of 3mm diameter for easy mounting

Specification: Operational Voltage: 5V, Ambient Light & RGB Color Sensing, Proximity Sensing, Gesture Detection, Operating Range: 4-8in (10-20cm), I2C Interface (I2C Address: 0x39).

6. Wi-Fi Module

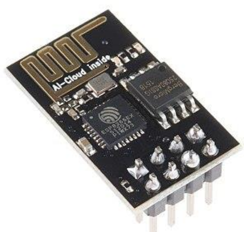

Figure 4.8: Wi-Fi Module

The ESP8266 is a low-cost Wi-Fi microchip with full TCP/IP stack and microcontroller capability produced by Shanghai-based Chinese manufacturer, Espressif Systems. The chip first came to the attention of western makers in August 2014 with the ESP-01 module, made by a third-party manufacturer, Ai-Thinker. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands. However, at the time there was almost no English-language documentation on the chip and the commands it accepted. The very low price and the fact that there were very few external components on the module which suggested that it could eventually be very inexpensive in volume, attracted many hackers to explore the module, chip, and the software on it, as well as to translate the Chinese documentation. The ESP8285 is an ESP8266 with 1 MB of built-in flash, allowing for single-chip devices capable of connecting to Wi-Fi. The successor to these microcontroller chips is the ESP32.

7. Gear Motor

Figure 4.9: Gear Motor

This is the popular digital gear motor, 360º continuous rotation in both direction, CW and CCW. Kind of special. It is still control it with standard RC (Radio Control) signal but it is specially designed for robotics because it can rotates 360 °. Perfect for mobile robot and very useful in robotics and automation project. And with 360º rotation, the servo shaft will not stay/hold at certain angle with the control signal. A normal/standard RC servo will move to certain angle/position accordingly to control signal where normally named PCM (Pulse Code Modulation) or PPM (Pulse Position Modulation). The signal is a continuous of TTL, High and Low voltage, with 20 ms of period or 50Hz frequency. For more info on standard RC servo control. But this is special servo which offer 360º, yet the control signal is still RC signal, but instead of move and stay at certain position, it rotate at certain direction with certain speed. For of course it can always control it via arduino too. Not only it uses the latest digital controller (The name digital servo) that provide better stability, and fast response; it also come with metal gear, offering robust and rugged usage. Now, it can control DC brush motor directly from controller such as Arduino without needing a motor driver in between.

8. MQ6 Gas Sensor

Figure 4.10: MQ6 Gas Sensor

The MQ-2 Gas sensor can detect or measure gasses like LPG, Alcohol, Propane, Hydrogen, CO and even methane. The module version of this sensor comes with a Digital Pin which makes this sensor to operate even without a microcontroller and that comes in handy when you are only trying to detect one particular gas. When it comes to measuring the gas in ppm the analog pin has to be used, the analog pin also TTL driven and works on 5V and hence can be used with most common microcontroller. Using an MQ sensor it detects a gas is very easy. You can either use the digital pin or the analog pin to accomplish this. Simply power the module with 5V and we

should notice the power LED on the module to glow and when no gas it detected the output LED will remain turned off meaning the digital output pin will be 0V. Remember that these sensors have to be kept on for pre-heating time (mentioned in features above) before you can actually work with it. Now, introduce the sensor to the gas you want to detect and you should see the output LED to go high along with the digital pin, if not use the potentiometer until the output gets high. Now every time your sensor gets introduced to this gas at this particular concentration the digital pin will go high (5V) else will remain low (0V). Can also use the analog pin to achieve the same thing. Read the analog values (0-5V) using a microcontroller, this value will be directly proportional to the concentration of the gas to which the sensor detects. You can experiment with this values and check how the sensor reacts to different concentration of gas and develop your program accordingly.
Features: Operating Voltage is +5V. Can be used to Measure or detect LPG, Alcohol, Propane, Hydrogen, CO and even methane. Analog output voltage: 0V to 5V. Digital Output Voltage: 0V or 5V (TTL Logic). Preheat duration 20 seconds. Can be used as a Digital or analog sensor
Applications: Detects or measure Gases like LPG, Alcohol, Propane, Hydrogen, CO and even methane. Air quality monitor. Gas leak alarm.

9. HX711-PCB

Figure 4.11: HX711-PCB

The SparkFun Load Cell Amplifier is a small breakout board for the HX711 IC that allows you to easily read load cells to measure weight. By connecting the amplifier to my microcontroller we will be able to read the changes in the resistance of the load cell and with some calibration we'll be able to get very accurate weight measurements. This can be handy for creating your own industrial scale, process control, or simple presence detection of an object. The HX711 uses a two wire interface (Clock and Data) for communication. Any microcontroller's GPIO pins should work and numerous libraries have been written making it easy to read data from the HX711. Load cells use a four wire wheatstone bridge to connect to the HX711. These are commonly colored RED, BLK, WHT, GRN, and YLW. Each color corresponds to the conventional color coding of load cells: 1.Red (Excitation+ or VCC) 2.Black (Excitation- or GND) 3.White (Amplifier+, Signal+, or Output+) 4.Green (A-, S-, or O-) 5.Yellow (Shield). The YLW pin acts as an optional input that is not hooked up to the strain gauge but is utilized to ground and shield against outside EMI (electromagnetic interference). Please keep in mind that some load cells might have slight variations in color coding.

10. LCD Display

Figure 4.12: LCD Display

A liquid-crystal display (LCD) is a flat-panel display or other electronically modulated optical device that uses the light-modulating properties of liquid crystals. Liquid crystals do not emit light directly, instead using a backlight or reflector to produce images in colour or monochrome.[1] LCDs are available to display arbitrary images (as in a general-purpose computer display) or fixed images with low information content, which can be displayed or hidden, such as preset words, digits, and 7-segment displays, as in a digital clock. They use the same basic technology, except that arbitrary images are made up of a large number of small pixels, while other displays have larger elements. LCDs are used in a wide range of applications including LCD televisions, computer monitors, instrument panels, aircraft cockpit displays, and indoor and outdoor signage. Small LCD screens are common in portable consumer devices such as digital cameras, watches, calculators, and mobile telephones, including smartphones. LCD screens are also used on consumer electronics products such as DVD players, video game devices and clocks. LCD screens have replaced heavy, bulky cathode ray tube (CRT) displays in nearly all applications. LCD screens are available in a wider range of screen sizes than CRT and plasma displays, with LCD screens available in sizes ranging from tiny digital watches to huge, big-screen television sets. Since LCD screens do not use phosphors, they do not suffer image burn-in when a static image is displayed on a screen for a long time (e.g., the table frame for an aircraft schedule on an indoor sign). LCDs are, however, susceptible to image persistence.[2] The LCD screen is more energy-efficient and can be disposed of more safely than a CRT can. Its low electrical power consumption enables it to be used in battery-powered electronic equipment more efficiently than CRTs can be. By 2008, annual sales of televisions with LCD screens exceeded sales of CRT units worldwide, and the CRT became obsolete for most purposes.

11. Conveyor Belt

Figure 4.13: Conveyor Belt

A conveyor belt is the carrying medium of a belt conveyor system (often shortened to belt conveyor). A belt conveyor system is one of many types of conveyor systems. A belt conveyor system consists of two or more pulleys (sometimes referred to as drums), with an endless loop of carrying medium—the conveyor belt—that rotates about them. One or both of the pulleys are powered, moving the belt and the material on the belt forward. The powered pulley is called the drive pulley while the unpowered pulley is called the idler pulley. There are two main industrial classes of belt conveyors; Those in general material handling such as those moving boxes along inside a factory and bulk material handling such as those used to transport large volumes of resources and agricultural materials, such as grain, salt, coal , ore , sand, overburden and more. History. Primitive conveyor belts were used since the 19th century. In 1892, Thomas Robins began a series of inventions which led to the development of a conveyor belt used for carrying coal, ores and other products. In 1901, Sandvik invented and started the production of steel conveyor belts. In 1905. Richard Sutcliffe invented the first conveyor belts for use in coal mines which revolutionized the mining industry. In 1913, Henry Ford introduced conveyor-belt assembly lines at Ford Motor Company's Highland Park, Michigan factory. In 1972, the French society REI created in New Caledonia the longest straight-belt conveyor in the world, at a length of 13.8 km. Hyacynthe Marcel Bocchetti was the concept designer. In 1957, the B. F. Goodrich

Company patented a conveyor belt that it went on to produce as the Turnover Conveyor Belt System. Incorporating a half-twist, it had the advantage over conventional belts of a longer life because it could expose all of its surface area to wear and tear. Möbius strip belts are no longer manufactured because untwisted modern belts can be made more durable by constructing them from several layers of different materials. In 1970, Intralox, a Louisiana-based company, registered the first patent for all plastic, modular belting. Structure. The belt consists of one or more layers of material. It is common for belts to have three layers: a top cover, a carcass and a bottom cover. The purpose of the carcass is to provide linear strength and shape. The carcass is often a woven or metal fabric having a warp & weft. The warp refers to longitudinal cords which characteristics of resistance and elasticity define the running properties of the belt. The weft represents the whole set of transversal cables allowing to the belt specific resistance against cuts, tears and impacts and at the same time high flexibility. The most common carcass materials are steel, polyester, nylon, cotton and aramid. The covers are usually various rubber or plastic compounds specified by use of the belt. Steel conveyor belts are used when high strength class is required. For example, the highest strength class conveyor belt installed is made of steel cords. Application. Today there are different types of conveyor belts that have been created for conveying different kinds of material available in PVC and rubber materials. Material flowing over the belt may be weighed in transit using a beltweigher. Belts with regularly spaced partitions, known as elevator belts, are used for transporting loose materials up steep inclines. Belt Conveyors are used in self-unloading bulk freighters and in live bottom trucks. Belt conveyor technology is also used in conveyor transport such as moving sidewalks or escalators, as well as on many manufacturing assembly lines. Stores often have conveyor belts at the check-out counter to move shopping items. Ski areas also use conveyor belts to transport skiers up the hill.

12. DC Power Supply (12V AC/DC Adaptor)

Figure 4.14: 12V AC/DC Adaptor

A power supply is an electrical device that supplies electric power to an electrical load. The primary function of a power supply is to convert electric current from a source to the correct voltage, current, and frequency to power the load. As a result, power supplies are sometimes referred to as electric power converters. Some power supplies are separate standalone pieces of equipment, while others are built into the load appliances that they power.

Examples of the latter include power supplies found in desktop computers and consumer electronics devices. Other functions that power supplies may perform include limiting the current drawn by the load to safe levels, shutting off the current in the event of an electrical fault, power conditioning to prevent electronic noise or voltage surges on the input from reaching the load, power-factor correction, and storing energy so it can continue to power the load in the event of a temporary interruption in the source power (uninterruptible power supply).

13. Wire

Figure 4.15: Wire

CHAPTER - 5

Software and Programming

CHAPTER - 5

Software and Programming

5.1 Introduction

In this chapter, sequential operation of the microcontroller is described through flow chart and also the microcontroller coding is discussed. The ATMEGA328p microcontroller embedded in Arduino development board is programmed using the Arduino Software. The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with computers. The programming language used in Arduino software is basically C/C Object.

5.2 Programming

The most important consideration of an ADC is its resolution. Resolution is the number of binary bits output by a converter. ADC circuits take an analog signal, which is continuously variable, and resolve it into one of many discrete steps. It is very important to know how many of these steps there are in total. The ADC of Arduino has 10 bit resolution which provides 1024 counts: 0 to 1023 decimal. It can represent up to 1024 unique conditions of signal measurement. Over the range of measurement from 0% and 100%, there will be exactly 1024 unique binary numbers output by the converter. The ADC measures the voltage from both the sensors, but does not provide the absolute voltage value. Instead it provides a numerical value that gives the sampled voltage with respect to a voltage reference. For Arduino having 5 volt reference and 10 bit A/D converter the returned value will be 0 for zero volt applied and 1023 for 5 volt applied. To take samples, following codes are used and then samples are scaled accordingly to convert them to real value –

```
#include <LiquidCrystal.h>
#include "HX711.h"
HX711 cell(2, 3);
LiquidCrystal lcd(12, 11, 5, 6, 7, 8);
#include<Timer.h>
Timer t;
#include <SoftwareSerial.h>
SoftwareSerial mySerial(9, 10);
#define heart 13
#include <DHT.h>
#define dht_dpin 14
#define DHTTYPE DHT11
DHT dht(dht_dpin, DHTTYPE);
int humi,tem;
long val = 0;
```

```
float count = 0;
int state=LOW;
int laststate=LOW;
int counter=0;
int sensorValue;
int GasSensorPin= 15;
const int MIC = 4; //the microphone amplifier output is connected to pin A0
int adc;
int dB, PdB;
char *api_key="VIJ43IHU2C7X68O9";     // Enter your Write API key from ThingSpeak
static char postUrl[150];
void httpGet(String ip, String path, int port=80);
void setup()
{
  Serial.begin(9600);
   mySerial.begin(115200);
  lcd.begin(20, 4);
   dht.begin();
  lcd.setCursor(0,0);
  lcd.print("Industry Monitoring");
  lcd.setCursor(0,1);
  lcd.print("Over Website");
  delay(3000);
  lcd.clear();
 lcd.print("WIFI Connecting");
 lcd.setCursor(0,1);
 lcd.print("Please wait....");
 lcd.setCursor(0,2);
 Serial.println("Connecting Wifi....");
 lcd.print("Connecting Wifi....");
 connect_wifi("AT",1000);
 connect_wifi("AT+CWMODE=1",1000);
 connect_wifi("AT+CWQAP",1000);
 connect_wifi("AT+RST",5000);
 connect_wifi("AT+CWJAP=\"WEBindustry\",\"12345678\"",10000);
 Serial.println("Wifi Connected");
 lcd.clear();
 lcd.print("WIFI Connected.");
 pinMode(heart, OUTPUT);
 delay(5000);
 t.oscillate(heart, 1000, LOW);
       t.every(20000, send2server);
  pinMode(A5, INPUT);
  pinMode(18, OUTPUT);
```

```
  state= digitalRead(A5);
}
void loop()
{
  /////////////////////////load cell/////////////
  count = count + 1;
  // Use only one of these
  val = ((count-1)/count) * val    +  (1/count) * cell.read(); // take long term average
 // val = 0.5 * val    +   0.5 * cell.read(); // take recent average
  Serial.println ((val-8268730)/-256.02);
lcd.setCursor(0,0);
lcd.print((val-8261259)/-362.53);
lcd.print("-Weight in (gm)");
//////////////////////////product  counter///////////////////
  {
if (state==1 && laststate==0)
{
counter=counter+1;
delay(500);
lcd.setCursor(0,1);
lcd.print("    -No of Prod.goes");
lcd.setCursor(0,1);
lcd.print(counter);
}
laststate=state;
state=digitalRead(A5);
}
 delay(10);
//////////////////GAS SENSOR///////////////////
{
  sensorValue=analogRead(GasSensorPin);
  lcd.setCursor(0,2);
  lcd.print("Gas:");
  lcd.print(sensorValue,DEC);
   lcd.print("ppm");
}
 delay(10);
/////////////////////////////////SOUND////////////////
{
  PdB = dB; //Store the previous of dB here
adc= analogRead(MIC); //Read the ADC value from amplifer
Serial.println (adc);//Print ADC for initial calculation
dB = (adc+60) / 19.5; //Convert ADC value to dB using Regression values
if (PdB!=dB)
lcd.setCursor(0,3);
```

```
lcd.print ("Sound:");
lcd.print (dB);
lcd.print ("dB");
}
 delay(10);
{
  lcd.setCursor(0,4);
  lcd.print("Humi: ");
  tem= dht.readTemperature();
  lcd.print(humi);   // printing Humidity on Serial
  lcd.print("%;");
  lcd.print("Tem:");
  humi= dht.readHumidity();
  lcd.print(tem);   // Printing temperature on Serial
  lcd.print(char(223));
  lcd.print("C");
}
 delay(2000);
  t.update();
if (sensorValue > 220 | dB> 120 )
{digitalWrite(18,HIGH);}
else
{digitalWrite(18,LOW);}
}
void send2server()
{
  char tempStr[8];
  char humidStr[8];
  char counterStr[8];
  char weightStr[8];
  char GasStr[8];
  char SoundStr[8];
  dtostrf(tem, 5, 3, tempStr);
  dtostrf(humi, 5, 3, humidStr);
  dtostrf((counter=counter+1), 5, 3, counterStr);
  dtostrf(((val-8261259)/-362.53), 5, 3, weightStr);
  dtostrf(sensorValue, 5, 3, GasStr);
  dtostrf(dB, 5, 3, SoundStr);
 sprintf(postUrl,
"update?api_key=%s&field1=%s&field2=%s&field3=%s&field4=%s&field5=%s&field6=%s",a
pi_key,tempStr,humidStr,counterStr,weightStr,GasStr,SoundStr);
  httpGet("api.thingspeak.com", postUrl, 80);
}
//GET https://api.thingspeak.com/update?api_key=VIJ43IHU2C7X68O9&field1=0
```

```
//https://api.thingspeak.com/update?api_key=VIJ43IHU2C7X68O9&field1=0
void httpGet(String ip, String path, int port)
{
  int resp;
  String atHttpGetCmd = "GET /"+path+" HTTP/1.0\r\n\r\n";
  //AT+CIPSTART="TCP","192.168.20.200",80
  String atTcpPortConnectCmd = "AT+CIPSTART=\"TCP\",\""+ip+"\","+port+"";
  connect_wifi(atTcpPortConnectCmd,1000);
  int len = atHttpGetCmd.length();
  String atSendCmd = "AT+CIPSEND=";
  atSendCmd+=len;
  connect_wifi(atSendCmd,1000);
  connect_wifi(atHttpGetCmd,1000);
}
void connect_wifi(String cmd, int t)
{
  int temp=0,i=0;
  while(1)
  {
    lcd.clear();
    lcd.print(cmd);
    Serial.println(cmd);
    mySerial.println(cmd);
    while(mySerial.available())
    {
    if(mySerial.find("OK"))
    i=8;
    }
    delay(t);
    if(i>5)
    break;
    i++;
  }
  if(i==8)
  {
   Serial.println("OK");
  lcd.setCursor(0,1);
 lcd.print("OK");
  }
  else
  {
   Serial.println("Error");
   lcd.setCursor(0,1);
  lcd.print("Error");
```

}
}

5.3 ThingSpeak Soft

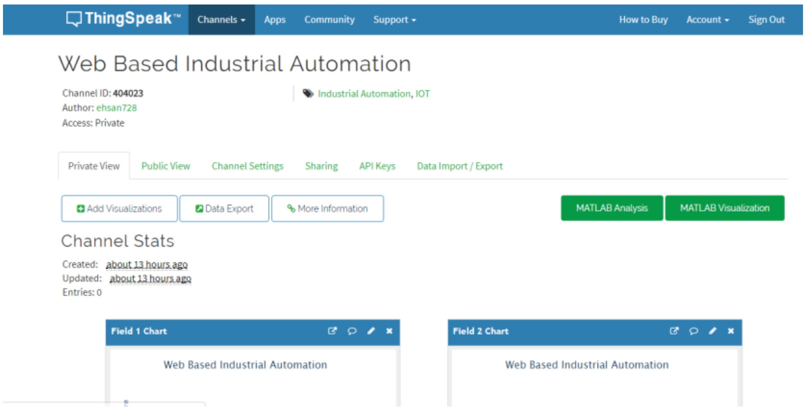

Figure 5.1: ThingSpeak Software for Web connection

ThingSpeak is an IOT analytics platform service from MathWorks, the makers of MATLAB and Simulink. ThingSpeak allows you to aggregate, visualize, and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices or equipment. Execute MATLAB code in ThingSpeak, and perform online analysis and processing of the data as it comes in. ThingSpeak accelerates the development of proof-of-concept IoT systems, especially those that require analytics. You can build IoT systems without setting up servers or developing web software. For small- to medium-sized IoT systems, ThingSpeak provides a hosted solution that can be used in production.

5.3.1 Access Our Data Both Online and Offline

ThingSpeak stores all the information you send it in one central location in the cloud, so you can easily access your data for online or offline analysis. Your private data is protected with an API key that you control. When you are logged in to your ThingSpeak account, you can use the web to securely download the data stored in the cloud. You can also programmatically read your data in CSV or JSON formats using a REST API call and the appropriate API key. Your devices can also read data from a ThingSpeak channel by subscribing to an MQTT topic. Import data from third-party web services including climate data from NOAA, public utility data from local utility providers, and stock and pricing data from financial providers. It also use that data together with the data you are collecting from your devices and equipment to investigate correlations and develop predictive algorithms. MATLAB users can import data stored in ThingSpeak into the MATLAB desktop environment using the ThingSpeak Support Toolbox.

Figure 5.2: Remotely Visualize Sensor Data in Real Time

5.3.2 Create Streaming Analytics, and Integrate with the Systems

Operationalize your analytics using the Time Control and React apps. With the Time Control app, you can schedule a computation to run once a day, once an hour, or as quickly as once every 5 minutes. The React App is used for condition monitoring. You can monitor the data coming in from your devices and set up an alert when the data indicates something may need attention. For example, you could configure ThingSpeak to send an email when the humidity on your plant floor exceeds a certain value. More broadly, your analyses can trigger events that push data from ThingSpeak to other web applications like Salesforce via REST APIs.

CHAPTER - 6

Performance Analysis

CHAPTER - 6

Performance Analysis

6.1 Introduction

In this chapter performance of the developed web based industry monitoring system is discussed. After final practical implementation of the model, the results were seen that our monitoring system is very quickly.

6.2 Results

6.2.1 Result Show in LCD Display Monitoring

Figure 6.1: In Normal Position All Sensor are Active

Figure 6.2: Number of product Show (IR Sensor)

Figure 6.3: Gas Value and Humidity Value is Increase (Gas Sensor)

Figure 6.4: Wight Value and Sound Value is Increase (Sound Sensor)

6.2.2 Result Show in Web Based Monitoring

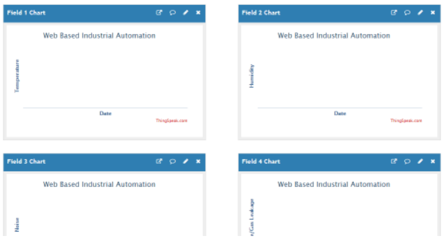

Figure 6.5: Normal Mood in Web Based Monitor

Figure 6.6: Sound Level Increase

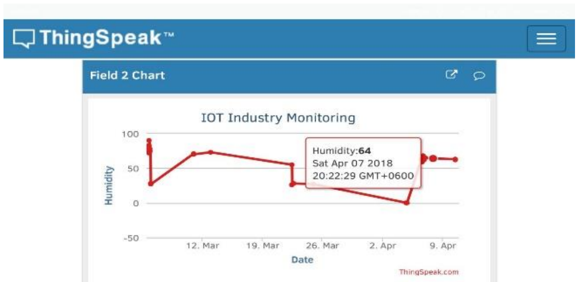

Figure 6.7: Humidity Increase

Figure 6.8: Number of Product Weight

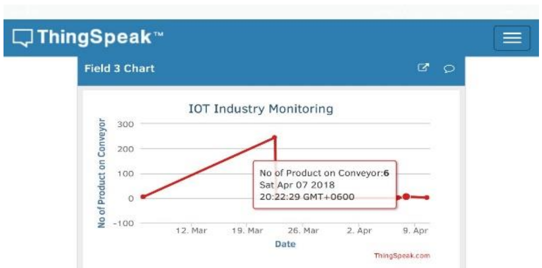

Figure 6.9: Number of Product Count

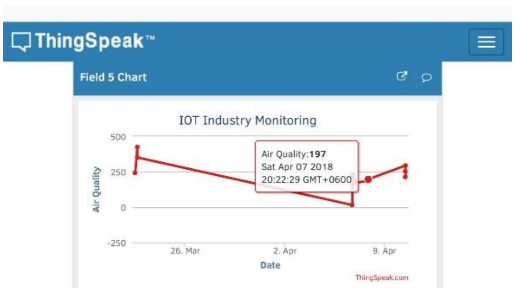

Figure 6.10: Increasing Air Quantity

- All sensor are work properly and delay time is very good.
- All value is show in display.
- By this System we can monitoring industry system and save data.
- It also show in WiFi base website for that reason industry system can monitoring at any place without delay.

6.3 Advantages of Our Web Based Industry Monitoring System

- This type of device is more efficetive any other product house.
- There is no need additional workers for monitoring.
- For this monitoring it creates more safety and lose is become more less.
- Mostly important it can monitoring at any place with web based soft.

CHAPTER-7

Conclusion

7.1 Limitations

7.2 Future studies

CHAPTER-7

Conclusion

7.1 Limitations

- Our all sensor work a little bit, Such as- Temperature sensor (0-s0^{o}c+_2), Gas sensor (200-1000ppm), Load cell sensor (0-5 kg).
- We can not control it just monitoring.

7.2 Future Studies

- We can switching this system by using a WiFi module NOT-NUS.
- We can include voice control for manual instruction.

REFERENCES

1. http://wiki.seeedstudio.com/Grove-Sound_Sensor/
2. https://www.makerlab-electronics.com/product/ir-proximity-sensor/
3. https://howtomechatronics.com/tutorials/arduino/dht11-dht22-sensors-temperature-and-humidity-tutorial-using-arduino/.
4. https://en.m.wikipedia.org/wiki/Conveyor_belt
5. https://en.wikipedia.org/wiki/DC_motor

APENDIX-A

Pin Diagram of Arduino is given bellow for better understanding of various I/O pins.

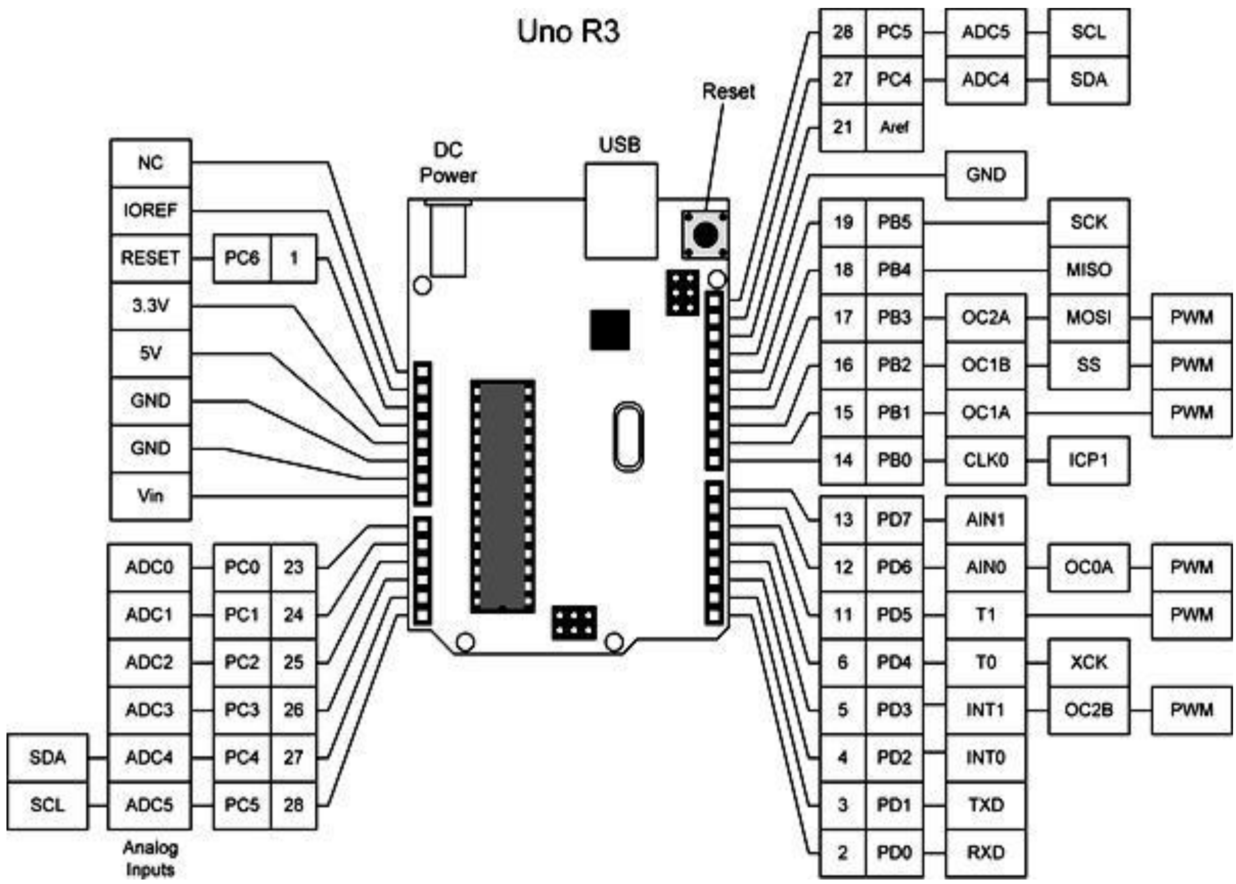

Figure : Pin Diagram of Arduino UNO

The elaborate functional description of ATMEGA328P has been given in its datasheet. Here, we only discussed about some important terms & features of this Microcontroller.

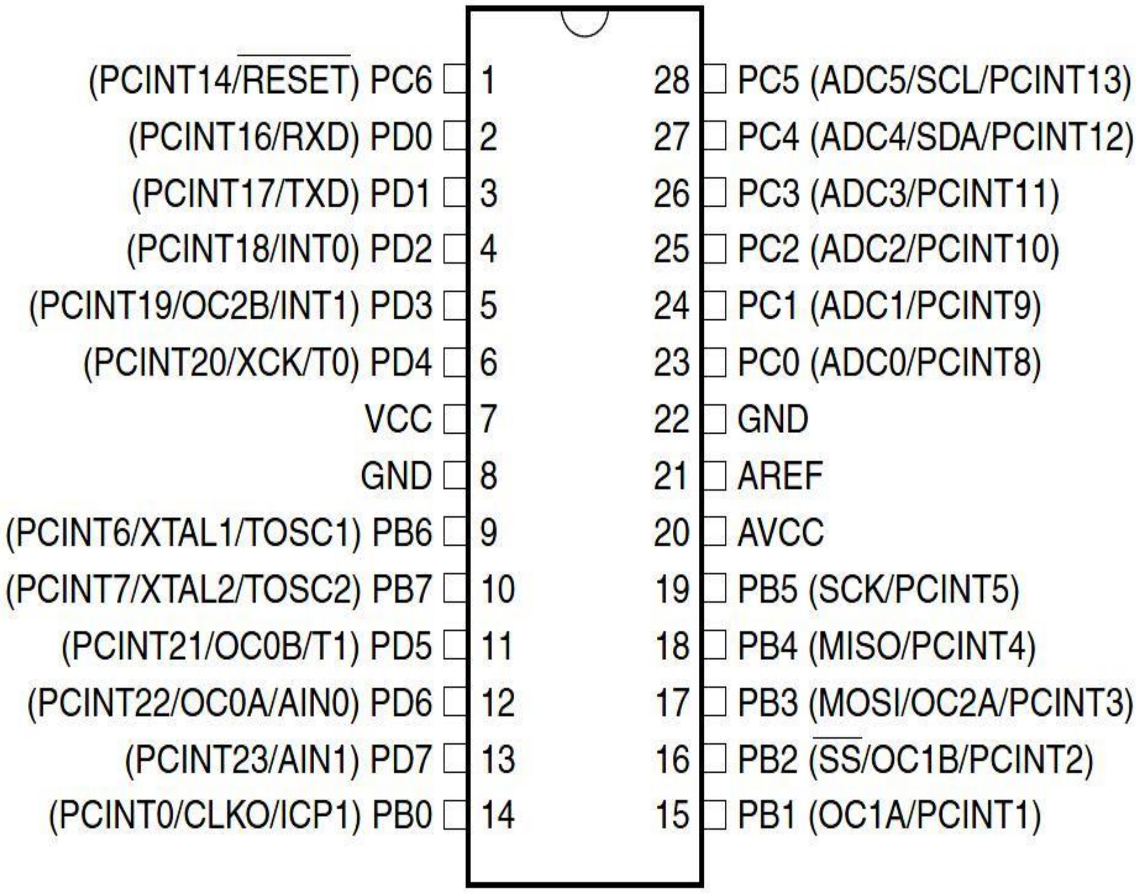

Figure: Pin Diagram of ATMEGA328P

Block Diagram of ATMEGA328P :

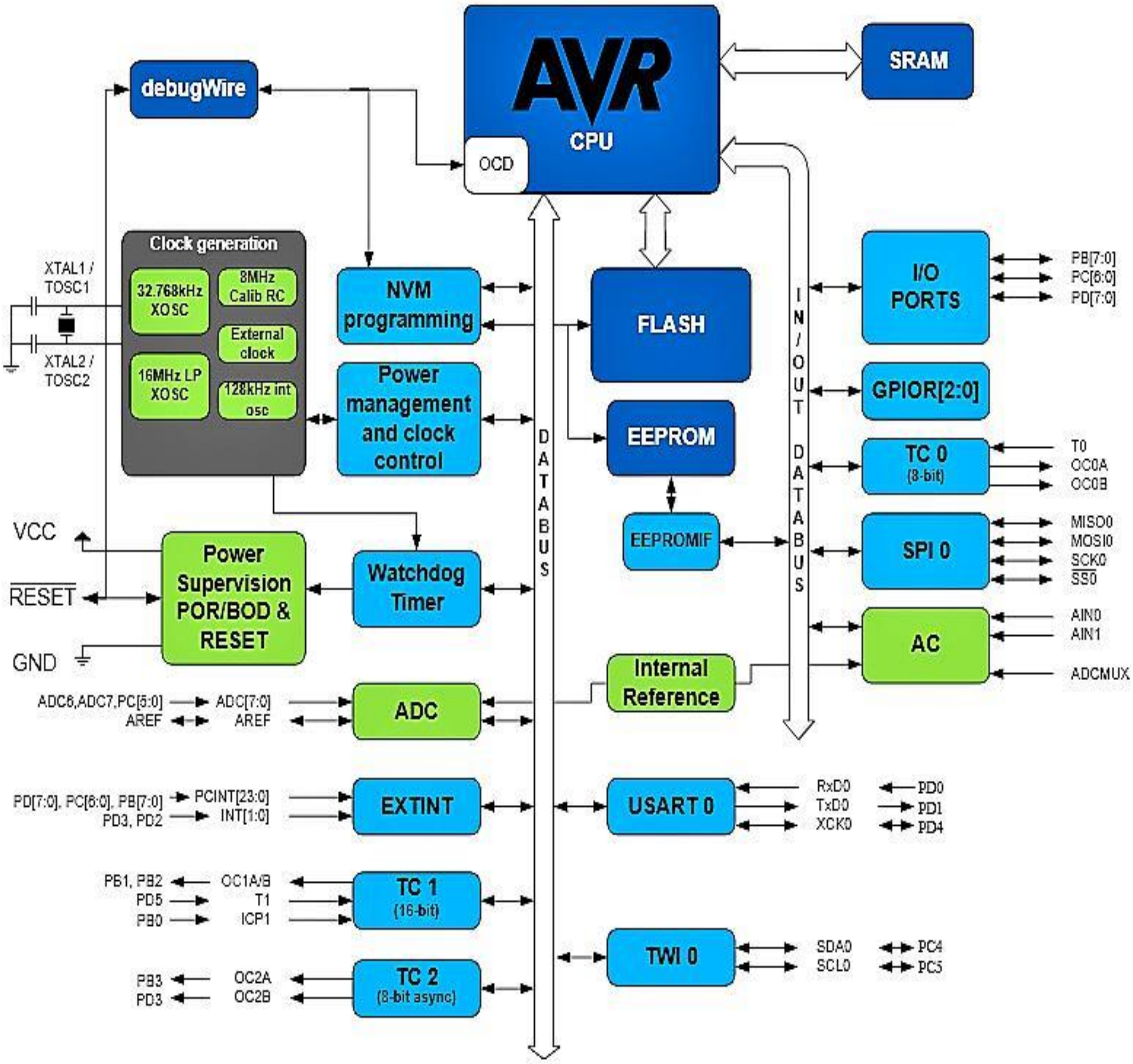

Figure : Block Diagram of ATMEGA328 Microcontroller

Pin Description of ATMEGA328P :

VCC:
Digital supply voltage.

GND:
Ground.

Port B (PB[7:0]) XTAL1/XTAL2/TOSC1/TOSC2 :
Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.
Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier. If the Internal Calibrated RC Oscillator is used as chip clock source, PB[7:6] is used as TOSC[2:1] input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

Port C (PC[5:0]) :
Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC[5:0] output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

PC6/RESET :
If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C. If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a Reset.

Port D (PD[7:0]) :
Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

AV_{CC} :

AV_{CC} is the supply voltage pin for the A/D Converter, PC[3:0], and PE[3:2]. It should be externally connected to V_{CC}, even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter. Note that PC[6:4] use digital supply voltage, V_{CC}.

AREF :

AREF is the analog reference pin for the A/D Converter.

ADC[7:6] (TQFP and VFQFN Package Only) :

In the TQFP and VFQFN package, ADC[7:6] serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

Block Diagram of ADC Module :

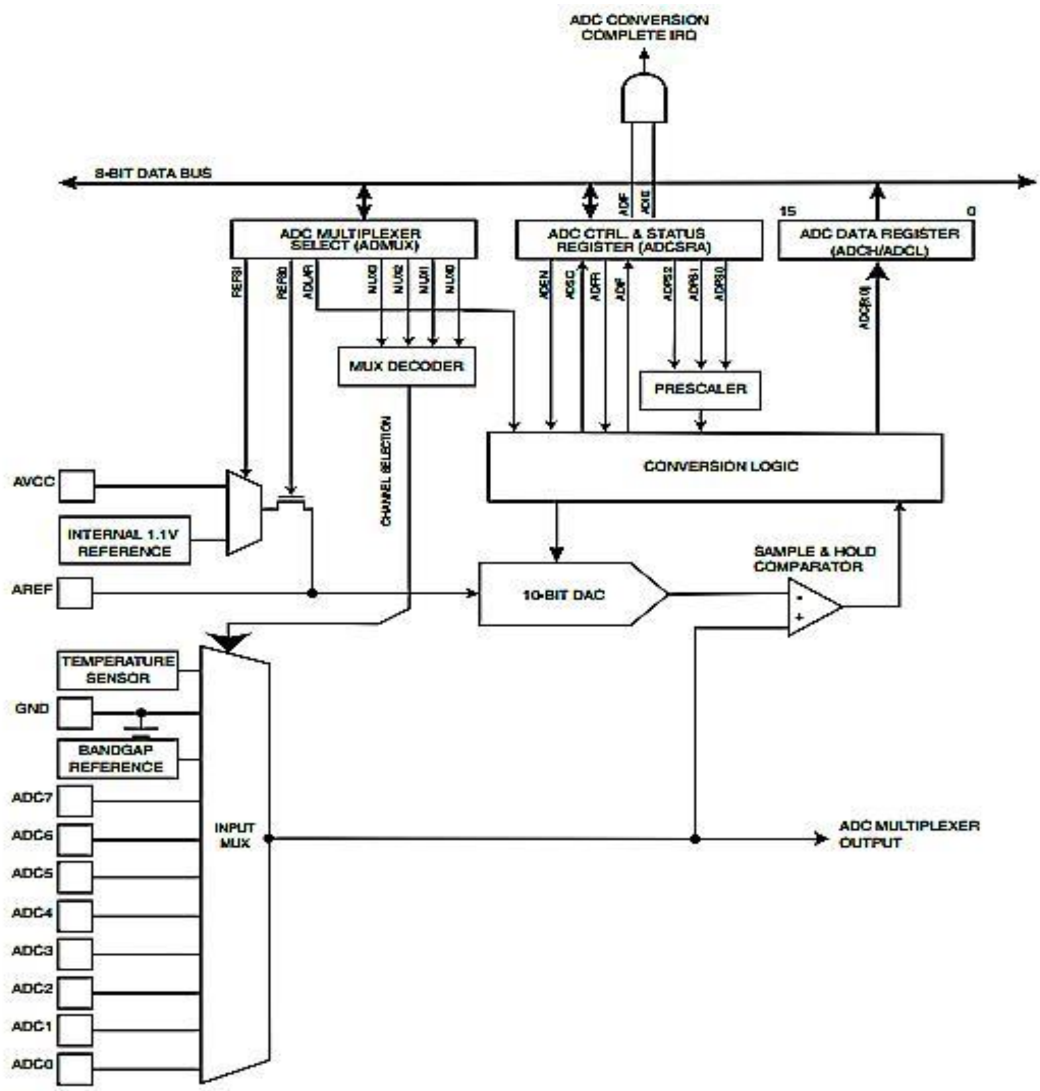

Figure : Block Diagram of AD converter of the microcontroller

Description of the ADC :

- The ATmega328/P features a 10-bit successive approximation ADC.
- The ADC is connected to an 8-channel Analog Multiplexer which allows eight single-ended voltage inputs constructed from the pins of Port A.
- The ADC contains a Sample and Hold circuit which ensures that the input voltage to the ADC is held at a constant level during conversion.
- The ADC has a separate analog supply voltage pin, AVCC. AVCC must not differ more than ±0.3V from VCC.
- The ADC converts an analog input voltage to a 10-bit digital value through successive approximation.
- The analog input channel is selected by writing to the MUX bits in ADMUX.
- Any of the ADC input pins, as well as GND and a fixed band gap voltage reference, can be selected as single ended inputs to the ADC. The ADC is enabled by setting the ADC Enable bit, ADEN in ADCSRA.
- Voltage reference and input channel selections will not go into effect until ADEN is set.
- The ADC generates a 10-bit result which is presented in the ADC Data Registers, ADCH and ADCL. By default, the result is presented right adjusted, but can optionally be presented left adjusted by setting the ADLAR bit in ADMUX.
- The ADC has its own interrupt which can be triggered when a conversion completes. When ADC access to the Data Registers is prohibited between reading of ADCH and ADCL, the interrupt will trigger even if the result is lost.