

Diagnosis of Fault and Monitoring of Small Wind Turbine Using Arduino

Srinivasa Reddy Padala

P Nithya Shree

Bala Hussain Pittala

S Rakshitha

Manjushree S

Department of Computer Science
GITAM School of Technology
GITAM University, Bengaluru
India

Abstract

India is the fourth country with the capacity to produce wind power after China, US and Germany. Wind energy has huge potential in the southern and western parts of mainland India along with few offshore wind sites. Wind turbines are one of the most important components in wind farms. Maintenance of wind turbine is time consumptive and expensive. In this paper, we implement fault detection and monitoring system for a small-scale wind turbine simulation using Arduino Uno.

Keywords: Wind turbine, IOT, Arduino, ThingSpeak API, Telegram

1. Introduction

With the rapid development of society and economy, the traditional energy sources are falling short of meeting the demand. Therefore, to keep up with the increasing pace of human development, it is necessary to develop a new source of energy. Wind energy is considered as one of the most preferred methods of renewable energy due to its inexhaustible advantages. The reliability of wind energy conversion system has garnered wide attraction. [1] Wind energy is considered as the cheapest renewable energy source. But the components used in this system are not, namely the wind turbine. The wind turbine is an important component of the system. Wind turbines harness the power of the wind and use it to generate electricity. A Fan and blades of a wind turbine might look similar, but they function oppositely. A fan uses electricity to produce wind whereas a turbine uses wind to produce electricity.

Wind turbine blades turn when hit by the wind, the blades of most turbines will start rotating at a wind speed of 3-5 m/s, which is approximately speed of a gentle breeze. It is this rotating motion of the blades that turns a shaft in the nacelle. The nacelle is a cuboidal structure which contains all the generating components of a wind turbine. It has a built-in generator to convert the kinetic energy of the shaft into electrical energy. This energy is then passed through a step-up transformer, which helps the voltage to be stepped up, which makes it easy to be transported on to the power grid or used by a local site. [A]

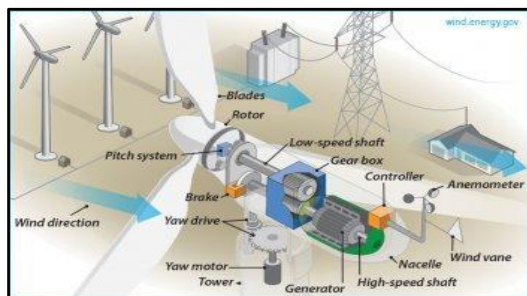


Fig 1. Cross-sectional image of a typical wind turbine.

Most wind turbines are built in remote locations such as mountains and offshore locations. While these locations might seem like the perfect location for their wind potential, but they suffer through inaccessibility. If any fault arises in the turbine, it will consume a lot of time and money to find out the fault and to rectify.

Introduction of fault detection and condition monitoring system is necessary to decrease the rate of failure of wind turbine components and downtime. Wind turbine fault detecting, and condition monitoring system is a tool used to evaluate the condition of the wind turbine subsystem. Speed, temperature, vibration is some of the parameters which are measured periodically by using sensors [10]. Along with them, we also have an IR sensor to detect foreign object intervention. Different sensors are installed in different parts of the turbine such as a gearbox, blades, generator etc. [2-8]. The measured values are monitored for consistency. In case of any anomaly, a notification is sent to authorised users. The main objective of this project is to increase the consistency of the wind turbine [2-7].

The interfacing to a personal computer is done by using an Arduino Uno board. Arduino Uno is a microcontroller-based board which is an open-source hardware and development software. We use the ThingSpeak API for real-time monitoring. The proposed system is implemented by connecting all sensors to Arduino Uno board, their data is analysed by Thing Speak API and Telegram is used as a visual interface which displays the status of wind turbine's health.

1.1 Arduino UNO

The Arduino Uno is a microcontroller board based on ATmega168 or ATmega328. In this project, we will be using ATmega328. It is an inexpensive and easily programmable microcontroller board. We are using Arduino language to program this system in Arduino User interface [b]. Through this user interface, we get data from ThingSpeak and send the notification through telegram.

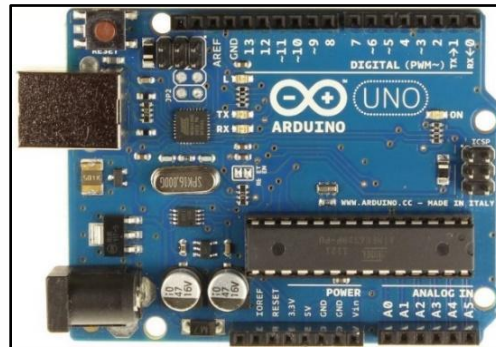


Fig 2. Arduino Uno board

1.2 Things Speak API

ThingSpeak is Application Programming Interface. ThingSpeak also acts as a web service for IoT. The interface covers basic communications infrastructure for entities within the IoT ecosystem. In addition, ThingSpeak enables customers to create programs from around information collected from the sensors. It enables almost real-time data collection, data processing and easy rendering for its users. [11-12]

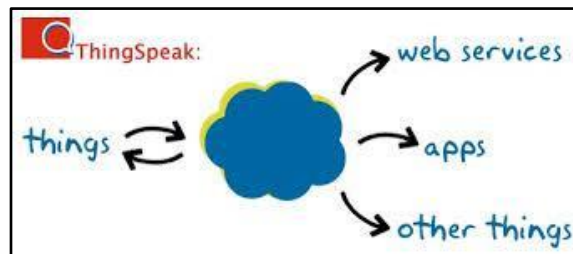


Fig 3. Thingspeak API as a cloud

1.3 Telegram

Telegram is an internet chat app. It's a cloud-based text messenger as well as an IP call service. They launched a platform for third-party developers to create bots [14]. Bots are program-operated telegram accounts. The customer end of the telegram is open-source software due to this; the app is popular among small-scale projects developer.

[13]



Fig 6. The User Interface on telegram

2. Existing System

2.1 IEC 61400-25 Standard

The International Standard IEC 61400-25 provides a standardized information exchange for monitoring and control of wind turbine. It consists of two main parts: local condition monitoring and central condition monitoring. The local monitoring part is located near the wind turbine while the central condition monitoring is done, as the name suggests is located at the control centre. In case of local condition monitoring, local functionality can vary from system to system. [8][9]Both models a lot of human intervention which is not the most suitable option for remoted located wind turbine.

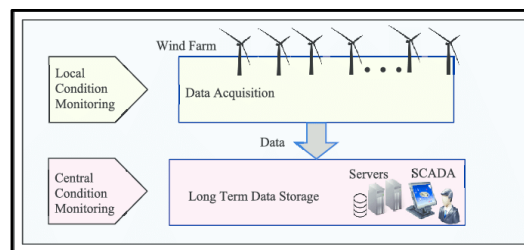


Fig 4. Schematic diagram of condition-based maintenance

3. Proposed System

The fault detection and condition monitoring system consist of a wind turbine, temperature sensor, vibration sensor, infrared sensor, Arduino UNO, NodeMCU and a display unit as mentioned in **fig 5**. We derive data from the sensors and send them to the ThingSpeak API by using a bot from telegram app which is our visual interface. In this section, we will see in detail how the system works.

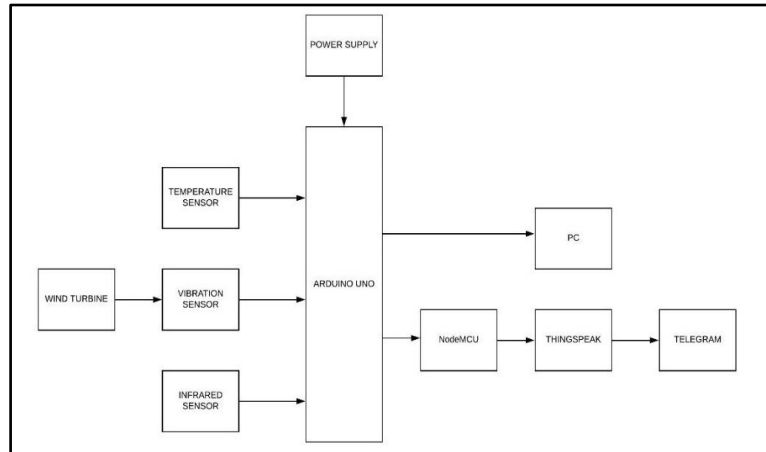


Fig 5. Block diagram of the proposed system

3.1 Hardware Components

3.1.1 Wind Turbine:

For a wind turbine, we used a small DC motor with 3V value. This motor is connected to a plastic fan-like structure which acts like blades. This system is connected to all the sensors and Arduino Uno. We chose a DC motor to provide a simulation model for an actual wind turbine.

3.1.2 Temperature Sensor:

The LM35 temperature sensor is used to measure the wind turbine temperature. It has an output voltage that is proportional to the Celsius temperature. It does not require any calibration and it is suitable for remote application. The LM35 is graded to operate a spectrum of -55° to $+150^{\circ}$ temperature scope.

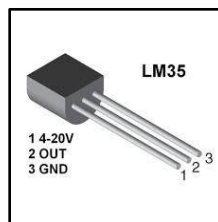


Fig 6. LM35 Temperature Sensor

3.1.3 Vibration Sensor

Piezoelectric Sensor is used to measure vibrations of the wind turbine components. Due to the usage of the piezoelectric effect, this device is called a piezoelectric sensor. The circuit is driven while the sensor stays stable, it displays a performance is maximal with respect to the module. As motion or vibration occurs, the circuit becomes disconnected and the output produced becomes null.



Fig 7: Vibration Sensor

3.1.4 Infrared Sensor

An infrared (IR) sensor is an electronic device that measures and detects infrared radiation in its surrounding environment. There are two types of IR sensor. [c]

Active IR Sensor: Active infrared sensors both emit and detect infrared radiation. In this project, we have used this one for the detection of faults in or near the wind turbine.

Passive IR sensor: Passive IR Sensor only detects infrared and it does not emit LED. PIR sensors are most commonly used in motion-based detection, such as in-home security systems.

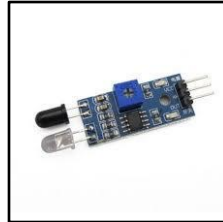


Fig 8: IR Sensor

3.1.5 Nodemcu

NodeMCU is a development kit that helps in prototyping and building various IoT products. It is an open-source firmware. It comprises of firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems.

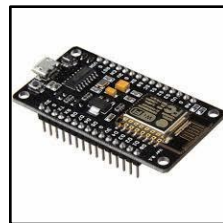


Fig 15: Nodemcu

3.1.6 LCD Display

An LCD (Liquid Crystal Display) screen acts as a display unit in this project. A 16x2 LCD means it can display 16 symbols for each segment and it consists of two such segments. We have used this as a local display unit which gives us updates on the health of our wind turbine consistently.

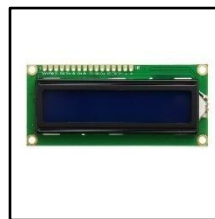


Fig 16: Liquid Crystal Display

3.2 Software Requirements

3.2.1 Arduino Uno

The Arduino Uno is a microcontroller board based on ATmega168 or ATmega328. In this project, we will be using ATmega328. It is an inexpensive and easily programmable microcontroller board. We are using Arduino language to program this system in Arduino User interface [b]. Through this user interface, we get data from ThingSpeak and send the notification through telegram.



Fig 17. Arduino Uno Board

3.2.2 Things Speak API

ThingSpeak is Application Programming Interface. ThingSpeak also acts as a web service for IoT. The interface covers basic communications infrastructure for entities within the IoT ecosystem. In addition, ThingSpeak enables customers to create programs from around information collected from the sensors. It enables almost real-time data collection, data processing and easy rendering for its users. [11-12]

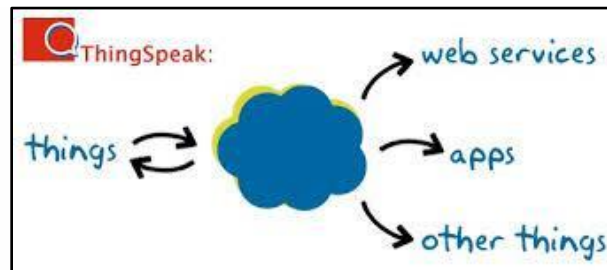


Fig 18. Thingspeak API as A Cloud

3.2.3 Telegram

Telegram is an internet chat app. It's a cloud-based text messenger as well as an IP call service. They launched a platform for third-party developers to create bots [14]. Bots are program-operated telegram accounts. The customer end of the telegram is open-source software due to this; the app is popular among small-scale projects developer. [13]



Fig 19. User Interface on Telegram

4. Implementation of the Model

4.1 Algorithms Used For This Model

The algorithms used for this model can be divided into two parts:

- i. Arduino and the sensors
- ii. Telegram (via ThingSpeak API)

4.1.1 Algorithm for Arduino and the Sensors

1. Begin
2. Input: Functioning of the wind turbine.
3. Output: Transference of data to cloud plus display on the LCD monitor.
4. Note all reading from sensors.
5. Send the readings to ThingSpeak API through the usage of NODEMCU.
6. Display the reading on the LCD screen.
7. End

4.1.2 Algorithm for Telegram

1. Begin - [Requests the user to “START” telegram bot]
2. Input: Data from the ThingSpeak API.
3. Output: Update about the health of the turbine.
4. Tests for connection
5. Sends message for information about sensors to ThingSpeak API
6. Receives information and displays it on its User Interface to update the user.
7. End

4.2 Implementation

In the implementation of the model, it can be noticed that the turbine (i.e. D.C motor) ran at smooth normal speed and all the details were shown in the user interface – for example-"motor is running at normal speed".

When a foreign object enters the perimeter of a turbine, AIR sensor emits lights and displays “obstacle detected “which can be potential fault state. Another potential fault state was erratic vibrations produced by the turbine when the motor runs at an abnormally high speed or low, the vibration sensor detects it and displays-"motor has created more torque" or "motor has created no torque". During such situations, there is a chance of an increase in temperature-"getting high temperature" which is not advisable for the health of the turbine. They detect almost all basic faulty scenarios that can be detected easily using these sensors.



Fig 20: Normal State

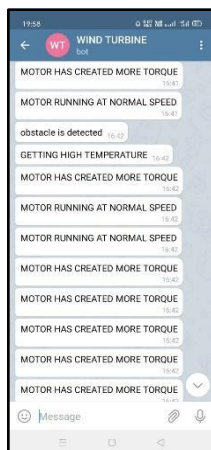


Fig 21: Fault State

5. Conclusion

In today's world of changing technology and the need for energy, there is also a great demand for energy monitoring. Thus, this demand for monitoring the development of systems such as IoT has helped us achieve this goal. The integrated system for wind energy system model gets its efficiency increased once it's linked to the internet of things.

This monitoring helps to work over distances. The system's power consumption is also relatively low. With the availability of solutions such as web interface, it is easier to access the output of the system on devices such as smartphones and personal computer. How the utilization of IoT a most valuable future technical paradigm is done is shown in this paper, by applying an IoT enabled solution for the condition monitoring of wind turbine.

When we follow traditional ways for this purpose the status of the turbine remains unknown between the periods of checking. But when we apply the proposed model of real-time condition monitoring of wind turbine is possible, so the faults can easily and early have identified, it helps to improve the life of the turbine and reduces the operation and maintenance cost. In this work, an integrated framework of different sensors, various communication networks, embedded devices and a wind turbine is used. Through this work, we can establish a simple fault detection and monitoring system for small wind turbines.

5.1 Future Scope

In future makes and extend that the proposed system is applicable for monitor the entire turbines in a wind a farm so that it is applicable for smart grid applications. Also, we can add many things add in future. Like different type of sensors for alternative required parameter monitoring.

References

Journals

- [1] Chen, F., Fu, Z. & Yang, Z. Wind power generation fault diagnosis based on deep learning model in the internet of things (IoT) with clusters. *Cluster Comput* **22**, 14013–14025 (2019).
<https://doi.org/10.1007/s10586-018-2171-6>
- [2] P. Devi*, V. Chandrasekaran, "Implementation of Arduino Based Condition Monitoring of Wind Turbine" *South Asian Journal of Engineering and Technology* Vol.2, NO.21 (2016) 134-139 – ISSN No: 2454-9614
- [3] W. Yang, P. Tavner, C. Crabtree, and M. Wilkinson, "Cost-effective condition monitoring for wind turbines," *IEEE Trans. Ind. Electron.*, vol. 57, no. 1, pp. 263–271, Jan. 2010.
- [4] Z. Hameed, Y. Hong, Y. Cho, S. Ahn, and C. Song, "Condition monitoring and fault detection of wind turbines and related algorithms: A review," *Renewable Sustainable Energy Rev.*, vol. 13, pp. 1–39, 2009.
- [5] P. Caselitz and J. Giebardt, "Rotor condition monitoring for improved operational safety of offshore wind energy converters," *J. Sol. Energy Eng.*, vol. 127, no. 2, pp. 253–261, 2005.

- [6] L. Y. Pao and K. E. Johnson, "Control of wind turbines," *IEEE Contr.Syst. Mag.*, vol. 31, no. 2, pp. 44–62, Apr. 2011.
- [7] S. S. Smater and A. D. Dominguez-Garcia, "A framework for reliability and performance assessment of wind energy conversion systems," *IEEE Trans. Power Syst.*, vol. 26, no. 4, pp. 2235–2245, Nov. 2011.
- [8] K. Kang, M. A. Ahmed and Y. Kim, "Implementation of condition monitoring and control system for small-scale wind turbines," *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, Dallas, TX, 2014, pp. 2122-2127.
- [9] IEC 61400-25-6, International Standard, 1st edition December 2010
- [10] R.A. Swartz, J.P. Lynch, B. Sweetman, R. Rolfes, and S. Zerbst, "Structure monitoring of wind turbines using wireless sensor networks, "Workshop on Sensor Networks for Civil Infrastructure Systems, Cambridge, England, pp. 1-8, 2008.
- [11] Introduction to the "Internet of Things" and ThingSpeak. ThingSpeak Community.
- [12] Maureira, Marcello A. Gómez. "ThingSpeak – an API and Web Service for the Internet of Things." (2014).
- [13] Wikipedia contributors. "Telegram (software)." *Wikipedia, The Free Encyclopaedia*. Wikipedia, The Free Encyclopaedia, 26 Apr. 2020. Web.
- [14] Telegram Bot Platform, Telegram, 24 June 2015, retrieved 1 September 2015

Web pages

- [a]<https://www.energy.gov/eere/wind/how-do-wind-turbines-work>
- [b]www.arduino.cc/en/main/Arduinoboarduno
- [c]www.fierceelectronics.com/sensors/what-ir-sensor