International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 10 (2018) pp. 7806-7810
© Research India Publications. http://www.ripublication.com

IOT Based Circuit Breaker Monitoring & Control

Abhijit Das[1], Dr. K. Kalimuthu[2] , S.S. Biswas[3]

[1]M. Tech. Final Year Student, [2]Associate Professor (ECE), [3]Scientific Officer
[1,2]SRM Institute of Science & Technology, Kattankulathur, India.
[3]BHAVINI, Kalpakkam, India

Abstract

This paper presents Internet of Thing (I.O.T) based monitoring & control of circuit breaker. Circuit breaker is an important component of Industrial Electrical System. It is used for protection & switching. Hence, reliable operation of circuit breaker is essential. Circuit breaker ages over time & number of operations. This raises a concern regarding reliability of circuit breaker operation. In order to ascertain reliability of circuit breaker, it is general practice to carry out preventive maintenance at fixed time intervals. The main disadvantage of this maintenance approach is unnecessary downtime & offline usage of separate diagnostics equipment although the circuit breaker is healthy. This increases the maintenance cost of circuit breakers. Moreover, in present practice, the control of circuit breaker is realized through hardwired control logic which increases the size of control & metering cabinet of the circuit breaker & prevents integration of Internet of Things. This put a limitation on decision making process as circuit breakers data are not accessible on the fly. Presently monitoring of circuit breaker is being carried out through proprietary solutions like Remote Terminal Units & SCADA. Proprietary solution raises a concern regarding reliability & security of the safety/safety related/strategic application as the backend implementation of proprietary solution is not accessible by the user.

In this paper, an attempt was made to develop monitoring & control scheme of a typical circuit breaker using Arduino Mega 2560 embedded microcontroller along with Ethernet Shield for integration of Internet of Things. Circuit breaker parameters like load current, trip coil current, close coil current, spring charging motor current, number of closing operations, number of tripping operations etc. are monitored. The monitored data are uploaded to Internet of Things platform “ThingSpeaks” in order to make circuit breaker data available on the fly for effective decision making. Monitored circuit breaker parameters are used to determine health of the circuit breaker in order to ascertain its reliable operation & to determine its maintenance/replacement needs. This paper is an effort to develop automated circuit breaker monitoring & control systems that diagnose the electrical and mechanical health of circuit breaker in real time. This is a shift in the maintenance paradigm from time-based maintenance to as-needed maintenance. This shift comes with the benefit of maintaining adequate circuit breaker performance while reducing overall maintenance costs & unnecessary downtime. Moreover, open source platform is used which eliminates the concern regarding reliability & security of the safety/safety related/strategic application as complete source code implementation is open & fully accessible by the user.

Keyword: Internet of Thing, Circuit Breaker, ThingSpeak, Arduino.

INTRODUCTION

Circuit breakers are used for protection & switching in industrial electrical system. Hence, reliable operation of circuit breaker is essential. Failure of circuit breaker can cause huge damage to industrial electrical system including revenue loss & fatality. circuit breakers are subjected to stress during their operation as they contain many mechanical/electrical components, has to carry rated or fault power & participates in arc quenching methods. Due to this circuit breaker ages over time & number of operations. This raises a concern regarding reliability of circuit breaker operation.

In order to ascertain reliability of circuit breaker, it is general practice to carry out preventive maintenance at fixed time intervals. In preventive maintenance program, the circuit breakers are taken out of service at regular intervals & its parameter like trip coil current, close coil current, spring charging motor current etc. are measured using special diagnostic equipment. Moreover, visual inspection & cleaning of its components like fixed contact, moving contact, tripping coil, closing coil, spring charging motor, operating mechanism etc. are carried out in order to ensure its healthiness. Being a time-based maintenance approach, downtime of the system increases although the circuit breaker is healthy. Moreover, use of special diagnostic equipment increases the maintenance cost of circuit breakers.

One more important issue with reference to circuit breaker is that the control of switchgear is realized through hardwired control logic which increases the size of control & metering cabinet of the circuit breaker & prevents integration of Internet of Things. This approach puts a limitation on timely execution of any modification in the circuit breaker control circuit. Also there is limitation on decision making process as circuit breakers data are not accessible on the fly.

This could be realized through proprietary solutions like Remote Terminal Units & SCADA. But, proprietary solution raises a concern regarding cost effectiveness, after sales support. Moreover, reliability & security of the safety/safety related/strategic application is a concern as the backend implementation of proprietary solution is not accessible by the user. Similar incident had already occurred in Iran by a malicious computer worm called “Stuxnet”. Stuxnet is a malicious computer worm, first uncovered in 2010 by Kaspersky Lab. Till now no country has openly admitted responsibility of Stuxnet. Stuxnet specifically targets programmable logic controllers (PLCs), which allow the automation of electromechanical processes such as those used to control machinery on factory assembly lines, amusement rides, or centrifuges. Stuxnet functions by targeting machines using the Microsoft Windows operating system and networks, then seeking out Siemens Step7 software. Stuxnet reportedly compromised Iranian PLCs, collecting information on industrial systems and causing the fast-spinning centrifuges to tear themselves apart. Stuxnet’s design and architecture are not domain-specific and it could be tailored as a platform for attacking modern supervisory control and data acquisition (SCADA) and PLC systems (e.g., in factory assembly lines or power plants). Stuxnet reportedly ruined almost one fifth of Iran's nuclear centrifuges. Targeting industrial control systems, the worm infected over 200,000 computers and caused 1,000 machines to physically degrade. This thing can be avoided if the development environment is open & completely in user’s control.

In this paper, an attempt was made to develop monitoring & control scheme of a typical circuit breaker using open source Arduino Mega 2560 embedded microcontroller along with Ethernet Shield for integration of Internet of Things. Circuit breaker parameters like load current, trip coil current, close coil current, spring charging motor current, number of closing operations, number of tripping operations etc. are monitored. The monitored data are uploaded to Internet of Things platform “ThingSpeaks” in order to make circuit breaker data available on the fly for effective decision making. Monitored circuit breaker parameters are used to determine health of the circuit breaker in order to ascertain its reliable operation & to determine its maintenance/replacement needs. This paper is an effort to develop automated circuit breaker monitoring & control systems that diagnose the electrical and mechanical health of circuit breaker in real time. This is a shift in the maintenance paradigm from time-based maintenance to as-needed maintenance. This shift comes with the benefit of maintaining adequate circuit breaker performance while reducing overall maintenance costs & unnecessary downtime. Moreover, open source platform is used which eliminates the concern regarding reliability & security of the safety/safety related/strategic application as complete source code implementation is open & fully accessible & controlled by the user.

DETAILED DESIGN BASIS OF THE PROJECT

A. Need for the proposed system

Circuit breakers are used for protection & switching & an important component of industrial electrical system. In order to ascertain reliability of circuit breaker & at the same time to reduce the downtime due to time based preventive maintenance, online monitoring of circuit breaker health parameter is needed. Deployment of software-based control instead of hardwired control reduces the size of control & metering cabinet. Integration of Internet of Things make the circuit breaker health data available on the fly for effective decision making regarding maintenance of circuit breaker. Deployment of open source platform eliminates the concern regarding reliability & security of the safety/safety related/strategic application as complete source code implementation is open & fully accessible & controlled by the user.

The proposed system facilitates automated circuit breaker monitoring & control that diagnose the electrical and mechanical health of circuit breaker in real time. This is a shift in the maintenance paradigm from time-based maintenance to as-needed maintenance. This shift comes with the benefit of maintaining adequate circuit breaker performance while reducing overall maintenance costs & unnecessary downtime.

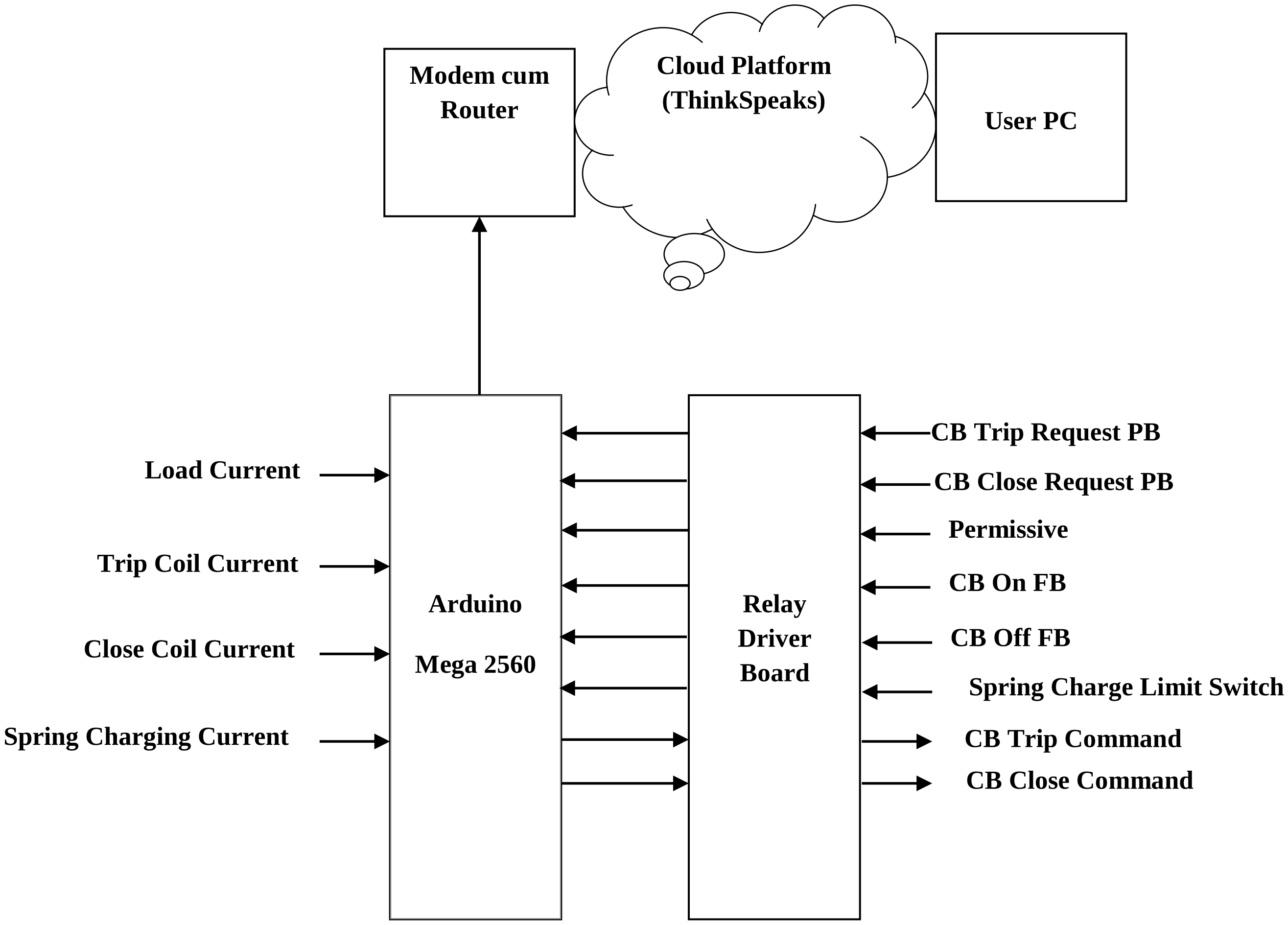
B. Objective

a) The main objective of this paper is to use open source platform to continuously monitor and control the circuit breaker.

b) There is a shift in the maintenance paradigm from time-based maintenance to as-needed maintenance. This shift comes with the benefit of maintaining adequate circuit breaker performance while reducing overall maintenance costs & unnecessary downtime.

C. Block Diagram of the proposed system

For rapid prototyping, Arduino/BeagleBone board can be used to prove the concept. But in this project we used an Arduino Mega 2560 board where 10 bit ADC are used to log the continuous data and digital pins are used to control the circuit breaker. The circuit breaker mechanism is simulated through contactors. An indicative schematic of the system is shown below.



D. Hardware required for the system

Arduino Mega 2560 Development Board

8-Channel Relay Driver Board

ACS712 Current Sensor

Contactors for Simulating Circuit Breaker Mechanism

Push Buttons & Switches

Lamp with Holder for Simulating Load

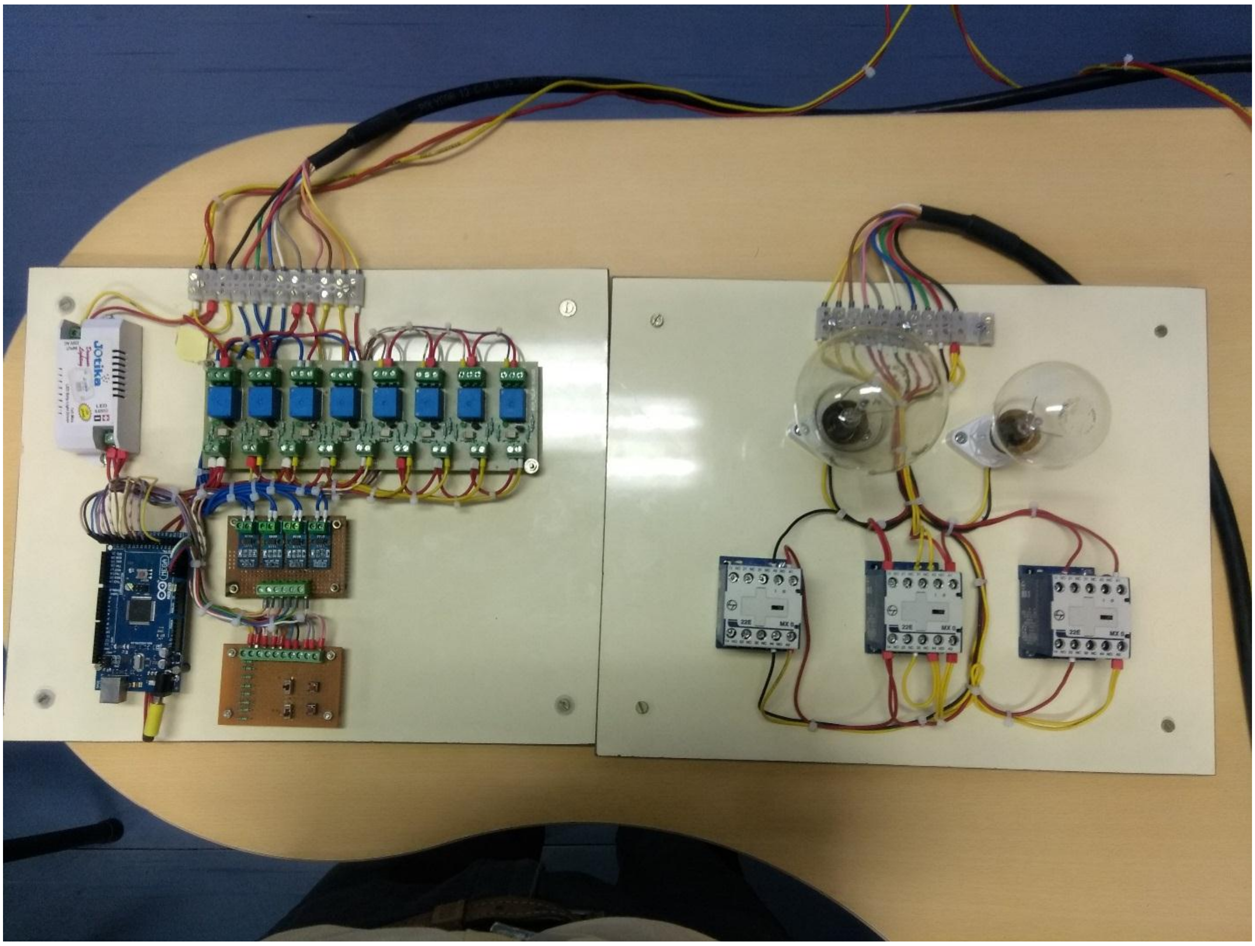
Modem cum Router

User PC to View Data

E. Familiarization of ThingSpeak

According to its developers, "ThingSpeak is an open source Internet of Things (IoT) application and API to store and retrieve data from things using the HTTP protocol over the Internet or via a Local Area Network. ThingSpeak enables the creation of sensor logging applications, location tracking applications, and a social network of things with status updates". ThingSpeak was originally launched by ioBridge in 2010 as a service in support of IoT applications. ThingSpeak has integrated support from the numerical computing software MATLAB from MathWorks, allowing ThingSpeak users to analyze and visualize uploaded data using Matlab without requiring the purchase of a Matlab license from Mathworks. ThingSpeak has a close relationship with Mathworks, Inc. In fact, all of the ThingSpeak documentation is incorporated into the Mathworks' Matlab documentation site and even enabling registered Mathworks user accounts as valid login credentials on the ThingSpeak website.[5] The terms of service[6] and privacy policy[7] of ThingSpeak.com are between the agreeing user and Mathworks, Inc. ThingSpeak has been the subject of articles in specialized "Maker" websites like Instructables, Codeproject and Channel 9.

F. *Experimental Setup*



G. *Arduino Code to Read Voltage*

```
#include "ThingSpeak.h"

#if defined(ARDUINO_ARCH_ESP8266) ||
defined(ARDUINO_ARCH_ESP32)
        #error "EPS8266 and ESP32 are not compatible with
this example."
#endif

#if !defined(USE_WIFI101_SHIELD) &&
!defined(USE_ETHERNET_SHIELD) &&
!defined(ARDUINO_SAMD_MKR1000) &&
!defined(ARDUINO_AVR_YUN)
  #error "Uncomment the #define for either
USE_WIFI101_SHIELD or USE_ETHERNET_SHIELD"
#endif

#if defined(ARDUINO_AVR_YUN)
    #include "YunClient.h"
    YunClient client;
#else
  #if defined(USE_WIFI101_SHIELD) ||
defined(ARDUINO_SAMD_MKR1000)
    // Use WiFi
    #include <SPI.h>
    #include <WiFi101.h>
    char ssid[] = "<YOURNETWORK>";    //  your
network SSID (name)
    char pass[] = "<YOURPASSWORD>";   // your
network password
    int status = WL_IDLE_STATUS;
    WiFiClient  client;
  #elif defined(USE_ETHERNET_SHIELD)
    // Use wired ethernet shield
    #include <SPI.h>
    #include <Ethernet.h>
    byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE,
0xED};
    EthernetClient client;
  #endif
#endif

#ifdef ARDUINO_ARCH_AVR
  // On Arduino:  0 - 1023 maps to 0 - 5 volts
  #define VOLTAGE_MAX 5.0
  #define VOLTAGE_MAXCOUNTS 1023.0
#elif ARDUINO_SAMD_MKR1000
  // On MKR1000:  0 - 1023 maps to 0 - 3.3 volts
  #define VOLTAGE_MAX 3.3
  #define VOLTAGE_MAXCOUNTS 1023.0
#elif ARDUINO_SAM_DUE
  // On Due:  0 - 1023 maps to 0 - 3.3 volts
  #define VOLTAGE_MAX 3.3
  #define VOLTAGE_MAXCOUNTS 1023.0
#endif

unsigned long myChannelNumber = 31461;
const char * myWriteAPIKey =
"LD79EOAAWRVYF04Y";

void setup()
{

  #ifdef ARDUINO_AVR_YUN
    Bridge.begin();
  #else
    #if defined(USE_WIFI101_SHIELD) ||
defined(ARDUINO_SAMD_MKR1000)
        WiFi.begin(ssid, pass);
    #else
      Ethernet.begin(mac);
    #endif
  #endif

  ThingSpeak.begin(client);

}

void loop()
{
  // Read the input on each pin, convert the reading, and set
each field to be sent to ThingSpeak.
  // On Uno,Mega,Yun:  0 - 1023 maps to 0 - 5 volts
  // On MKR1000,Due: 0 - 4095 maps to 0 - 3.3 volts
  float pinVoltage = analogRead(A0) * (VOLTAGE_MAX
/ VOLTAGE_MAXCOUNTS);
  ThingSpeak.setField(1,pinVoltage);

  // Write the fields that you've set all at once.
  ThingSpeak.writeFields(myChannelNumber,
myWriteAPIKey);

  delay(20000); // ThingSpeak will only accept updates
every 15 seconds.
}
```

H. *Advantage of the Proposed System*

Circuit breaker health data can be accessed from anywhere in the world.

Efficient and low cost design.

Real time monitoring.

Improve circuit breaker reliability and minimize downtime through as needed maintenance approach.

Minimize condition monitoring costs through unified monitoring of various parameters for the entire circuit breaker.

Integrate with your facility's computerized maintenance management software (CMMS).

CONCLUSION

This paper shows a conceptual implementation of IOT based Circuit Breaker Monitoring & Control which will reduce the size of the circuit breaker & facilitate the concept of as needed maintenance approach. Moreover, this will eliminate the concern regarding security vulnerability of third party system as the platform is opensource. Being a prototype conceptual implementation, future work can be done on EMI/EMC compliance, implementation of open source inferential engine in order to receive recommendation on maintenance, integration with computerized maintenance management software, development of open source hosting server etc.

REFERENCES

[1]. Jason M. Byerly, Carey Schneider, Robert Schloss, Isaac West “Real-Time Circuit Breaker Health Diagnostics,” 43rd Annual Western Protective Relay Conference, October 2016.

[2]. M. Kezunovic, G. Latisko, N. Ved, “Real-Time Monitoring and Assessment of Circuit Breaker Operations for Diagnostics and Control Applications”, Texas A&M University, College Station, TX 77843-3128.

[3]. Payman Dehghanian, Tomo Popovic, Mladen Kezunovic, “Circuit Breaker Operational Health Assessment via Condition Monitoring Data”, Department of Electrical and Computer Engineering, Texas A&M University College Station, Texas, 77843, USA.

[4]. Brent L. Carper, Principal Engineer, “The X, Y and Z of Circuit Breaker Control”, 34th Annual Hands-On Relay School, March 2017.