Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, June : 2024

Wifi-Enabled Industrial Monitor and Controller with Integrated Linux

Dr.R. Lakshmi Narayana Academic Consultant, *Department of Instrumentation, Sri Krishna Devaraya University Anathapuram,* AP India
Dr.Nagabhushan Raju Konduru Professor *Department of Instrumentation, Sri Krishna Devaraya University Anathapuram,* AP India
Dr. Sailaja Dasaraju Lecture in Physics *Government College (Autonomous), Anathapuram,* AP India

Abstract :

The creation of intelligent and networked industrial systems is crucial in the age of Industry 4.0. In order to provide a flexible and effective industrial monitoring and control system, this project, "Wireless Industrial Monitor and Controller Using Embedded Linux (RTOS)," explores the state-of-the-art combination of wireless communication protocols and embedded Linux technology. With the help of platforms like the BeagleBone Black and the powerful embedded Linux operating system, this system offers real-time data collecting, analysis, and control of industrial operations. It allows for smooth remote monitoring and control through wireless connectivity choices like Wi-Fi, increasing overall industrial automation, decreasing downtime, and improving operational efficiency.

This project demonstrates how embedded Linux has the ability to shape the factories of the future and symbolises the spirit of innovation and digital change within the industrial sector. In today's Internet of Things (IoT) environment, embedded systems are essential because they make it easier to integrate hardware and software for data collection, processing, and transmission. In order to build a strong Internet of Things solution, this research investigates the synergistic potential of merging the BeagleBone Black and NodeMCU platforms. The core hub for data collecting and processing is the BeagleBone Black, which runs a Debian OS based on Linux and makes use of its vast computational capacity and networking choices. Through serial transmission, it communicates with a variety of sensors and devices to retrieve data. With the help of Blynk IoT, the NodeMCU, which has Wi-Fi capabilities, serves as a bridge between the BeagleBone Black and the IoT environment, sending the collected data to distant servers or cloud-based services. This cooperative strategy highlights the adaptability and scalability of embedded systems in Internet of Things applications by facilitating effective data gathering and real-time monitoring.

Keywords: BeagleBone Black, LM35, Python, RTC

Introduction:

Combining online and embedded technologies for a web-managed data logger solution for electronic equipment. With an Ethernet network and a web browser, you may remotely access and monitor the onsite devices without being restricted by time or location to get real-time status updates. Numerous benefits come with this kind of embedded web server, including low power consumption, compact size, affordable price, and customisable design.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, June : 2024

It is a useful method for guiding users to embedded Linux systems over the internet. Using an embedded web server, embedded networking can be implemented [1]. The ARM architecture produced a microprocessor with excellent performance at a reasonable price and power consumption.

It is the most widely used 32-bit microprocessor and has been utilised extensively in embedded system architecture. Simultaneously, with the swift advancement of internet technology, internet-based data loggers are become more prevalent, and electronic systems are shifting from being used as system status monitors to being remote internet data loggers. The browser can be used immediately without the need to install extra client software, and the client is free to monitor the status of the specific electronic equipment [2]. In order to monitor, maintain, and lower the cost of the system, this paper presents a solution for embedded Linux-based system access via the internet. Cross-platform Linux operating system transplantation is a feasible option that offers a quick, easy, and convenient way to construct embedded Linux systems [3].

SYSTEM ARCHITECTURE

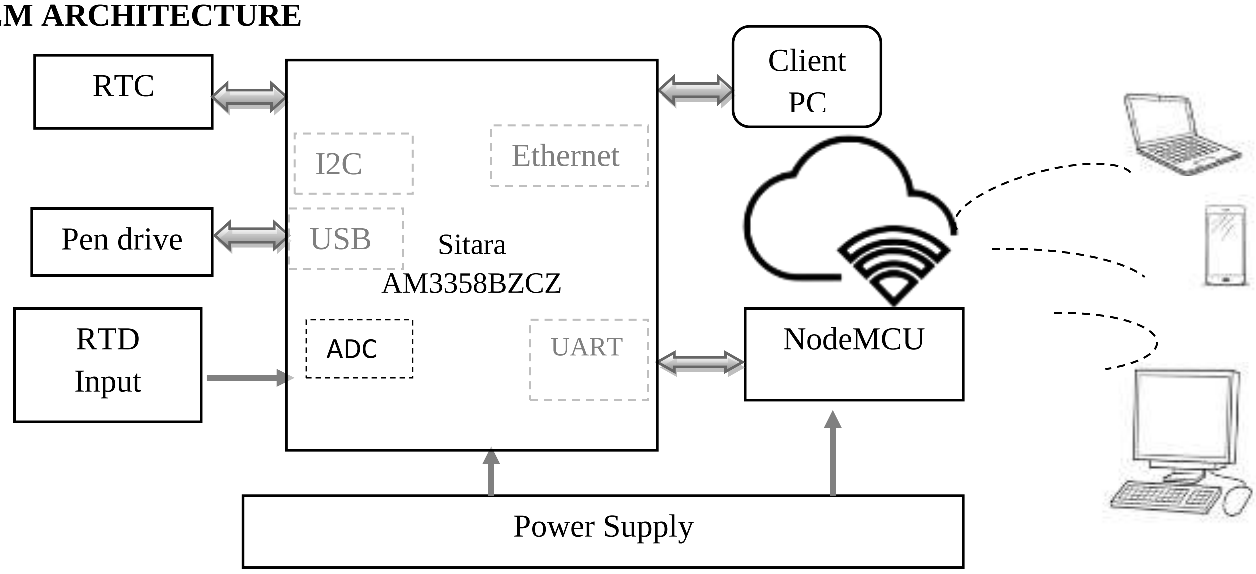

Figure.1. Block diagram of WIMCE system

Beaglebone Black :

Among single-board computers (SBCs), the BeagleBone Black is a development board that is both powerful and adaptable. The BeagleBoard.org Foundation created this little SBC, which is well-liked by professionals, educators, and hobbyists due to its performance, connection, and user-friendliness. We shall examine the BeagleBone Black development board's features, capabilities, uses, and importance in this post.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, June : 2024

The BeagleBone Black is incredibly feature-rich, which adds to its appeal and versatility:

• Processor: The board's 1 GHz ARM Cortex-A8 processor strikes a mix between power efficiency and performance.
• Memory: The BeagleBone Black has 512MB of DDR3 RAM, which is sufficient for a variety of jobs and applications.
• Connectivity: As seen in fig. 2, the board has several connectivity choices, including as Ethernet, USB ports, HDMI output, and a microSD card slot.
• GPIO: With 65 GPIO pins, the BeagleBone Black facilitates seamless integration with an extensive array of sensors, actuators, and more peripherals.
• Operating System: A variety of operating systems, including Linux versions like Debian and Ubuntu, are supported by the board .
• Cape Compatibility: The BeagleBone Black may be expanded using extension boards called "capes" to add features like WiFi communication and motor control, among others.

The BeagleBone Black is capable of much more than just what its hardware can do :
• Teaching Aid: The board is a great teaching aid for embedded systems, electronics, and programming. Hands-on learning is facilitated by its open-source nature and GPIO pins.
• Prototyping and Maker Projects: The BeagleBone Black is a great tool for custom electronic project creation and prototyping because of its GPIO pins and strong community support.
• Internet of Things (IoT): Building IoT devices that need data collecting, analysis, and communication is appropriate for this board due to its processing capability and networking possibilities.
• Automation and Robotics: The BeagleBone Black may operate as the brains behind robotics projects, directing actuators, sensors, and motors for a range of uses.
• Embedded Systems Development: Because of its tiny size, computing power, and expandability, it's a useful tool for creating and testing embedded systems.
The BeagleBone Black has become more important for a number of reasons:
• Open Source Philosophy: The board adopts an open-source stance, enabling users to contribute to, alter, and customise the platform's hardware and software components.
• Community Support: There is a thriving and active community for the BeagleBone Black that exchanges projects, guides, and troubleshooting advice. This assistance encourages group problem-solving and learning.
• Affordability: The BeagleBone Black is affordable for a wide spectrum of amateurs and professionals, even with its features.

The BeagleBone Black development board has made a name for itself as a strong and adaptable tool for learning, project creation, and prototyping. Its appeal in the fields of embedded systems, robotics, IoT, and maker culture can be attributed to its mix of computing power, connection, GPIO capabilities, and community support. The BeagleBone Black is still a vital tool for people who want to experiment with electronics, programming, and creativity as technology develops.

With an inbuilt micro HDMI connector, 512MB of DDR3L DRAM, 4GB of onboard flash memory operating at 1 GHz speed, and a TI Sitara AM3358 ARM Cortex-A8 processor, the Beaglebone Black is a low-cost development platform the size of a credit card. For smaller embedded applications and physical computing, it remains ideal. The hardware system's block diagram, shown in Figure 1, shows the Ethernet, power converter input, USB port for data storage, and real-time clock.

NodeMCU :

A well-liked open-source microcontroller board built on the ESP8266 Wi-Fi module is called NodeMCU. It is made to make Internet of Things (IoT) project development and prototyping simple. The NodeMCU board is a great option for applications that need wireless connection because of its small form factor and integrated Wi-Fi connectivity. It is driven by an inexpensive, low-power, highly integrated microprocessor called the ESP8266, which combines a microcontroller with Wi-Fi capabilities. The NodeMCU board has a USB interface for programming and power supply, along with GPIO pins for attaching other sensors and peripherals. It may be programmed by a variety of developers using the Arduino IDE or other well-known programming environments.

Sofware Development :

BeagleBone Black Python programming provides an adaptable and user-friendly environment for creating a variety of applications. Developers can use the BeagleBone Black's capabilities for tasks like sensor data processing, control systems, web services, and Internet of Things applications by utilising the rich libraries and Python's ease of use. Python easily blends into the development environment of a Linux-based operating system like Debian running on the BeagleBone Black, facilitating quick prototyping and reliable application development. Here, data is read from an analogue channel and sent to a serial port using Python. Using a relay, control the GPIO to turn on or off the output devices.

Website implementation using MySQL, PHP, and HTML. The foundational language of websites is called Hyper Text Markup Language [8]. Information is interpreted and displayed on your computer screen by a web browser. A text file with tiny markup elements that instruct the web browser how to display the page is called an HTML file. A computer language called Hypertext Preprocessor (PHP) [9] enables the creation of dynamic content that communicates with databases. PHP is an HTML-embedded server side programming language. Basically, web-based real-time event data loggers are developed using PHP. Information is sent to a web page and stored in a database table using the MySQL tool. It's true that the MySQL database management system has grown in popularity recently, particularly among the open source and Linux communities. It is popular for a number of reasons, including its speed and ease of setup, administration, and use. Applications built using MySQL may be created in a wide range of languages, and the database operates on numerous UNIX and Windows versions [10].

Experimental Setup of Proposed Method:

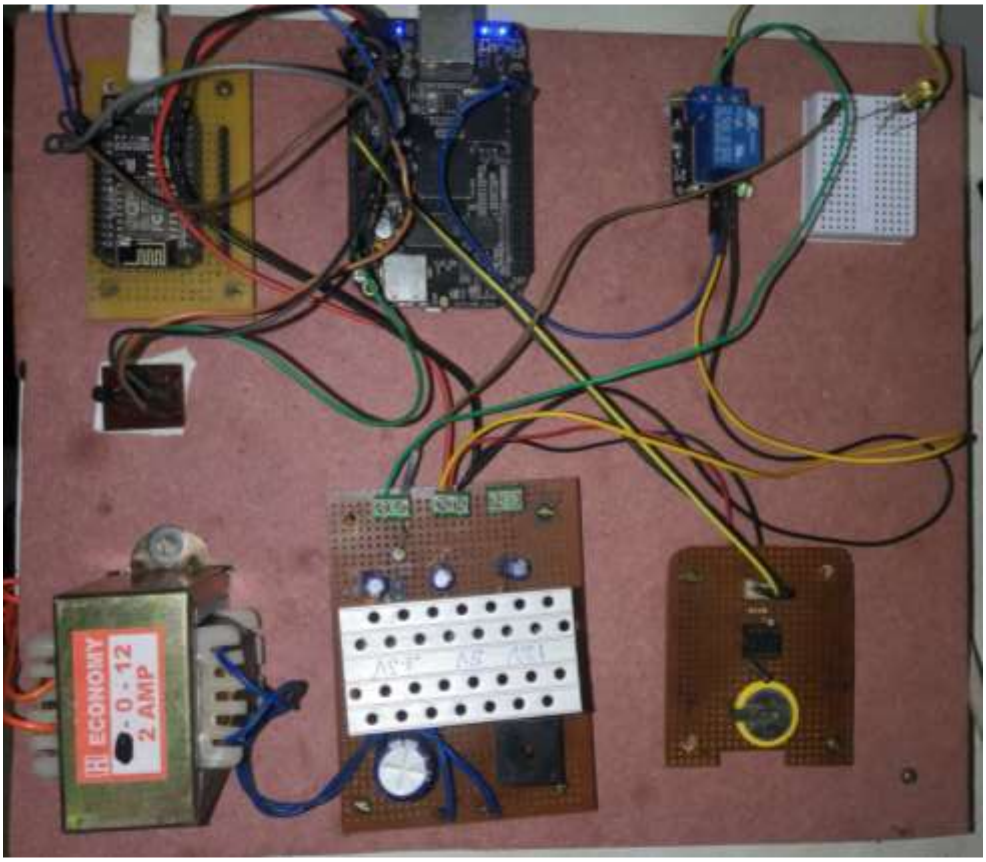

Figure : Photograph of proposed system setup

Beaglebone Black Source code:

import Adafruit_BBIO.ADC as ADC

import Adafruit_BBIO.GPIO as GPIO

import Adafruit_BBIO.UART as UART

import time

import serial

from time import sleep

```
relay = "P8_8"
Sensor = "P9_40"
ADC.setup(Sensor)
UART.setup("UART1")
GPIO.setup(relay,GPIO.OUT)
GPIO.output(relay,GPIO.LOW)
ser=serial.Serial(port="/dev/ttyO1",baudrate=115200)
ser.close()
ser.open()
while True:
    Temperature = (float(ADC.read_raw(Sensor))*4096)/1800
    Celsius = (Temperature)/22.75
    Fahrenheit = (Celsius * 9/5) + 32
    print('Temp in C: %d Temp in F: %d '%(Celsius , Fahrenheit))
    ser.write("Temperature: ")
    ser.write(str(Celsius).encode())
    ser.write("{:.2f}\n".format(Celsius).encode())
    ser.write(b'\n')
    time.sleep(2)

    if Celsius < 35:
        sleep(1)
```

```
            GPIO.output(relay,GPIO.LOW)

            ser.write("0")

           ser.write(b'\n')

        else:

            sleep(1)

            GPIO.output(relay,GPIO.HIGH)

            ser.write("1")

           ser.write(b'\n')

time.sleep(1)

ser,close()

GPIO.cleanup()
```

Output:

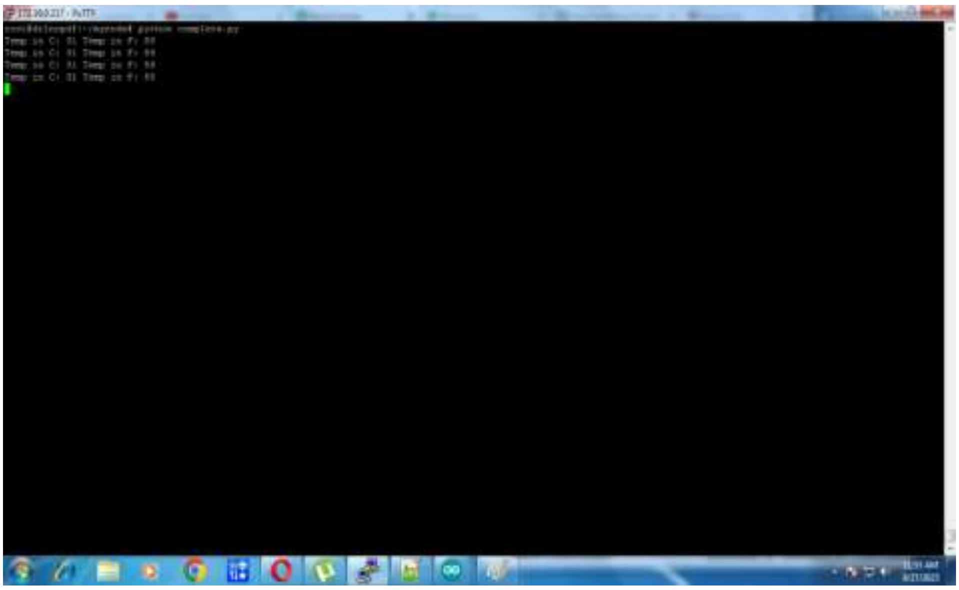

Figure : Screen shot of Beaglebone black terminal output

Nodemcu Source code:

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#ifdef ESP32
#pragma message(THIS EXAMPLE IS FOR ESP8266 ONLY!)
#error Select ESP8266 board.
#endif
float temperature = 0.00;
int ledstatus = 0;
int incomingByteTemp = 0;
int incomingByteAct = 0;
BlynkTimer timer;
char auth[] = "aqsc8WhyMK1SBiQNtRECM1fXQ1mQbwX7";
#define BLYNK_TEMPLATE_ID "TMPLjLUMQCPT"
#define BLYNK_TEMPLATE_NAME "Fire safety"
char ssid[] = "HonorPlay_5C31";    // Your WiFi credentials.
char pass[] = "123456789";  // Set password to "" for open networks.
void setup() {
// Open serial communications and wait for port to open:
Serial.begin(115200);
Blynk.begin(auth, ssid, pass);
}
void loop() {
if (Serial.available() > 0) {
   // Read data from the serial port
   String data = Serial.readStringUntil('\n');
   Serial.print("Received data: ");
```

```
  Serial.println(data);
  // Send the data to Blynk
  Blynk.virtualWrite(V1, data); // Assuming you have a Widget set to display data on V1
  Blynk.virtualWrite(V2, data); // Assuming you have a Widget set to display data on V1
 }
 delay(200);
 Blynk.run();
 timer.run();
}
```

Output :

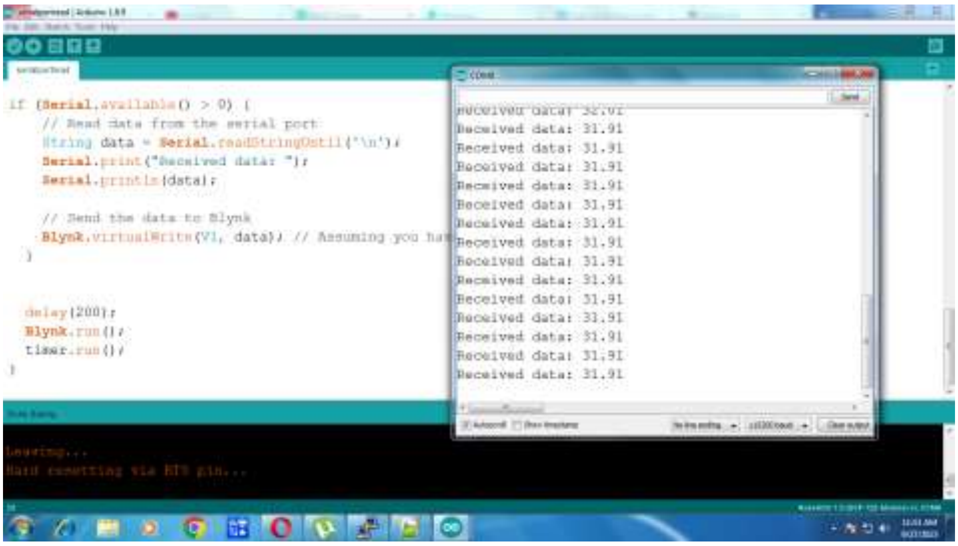

Figure : Screen shot of Nodemcu serial output

Results and Discussions :

Several important conclusions and discussions arise from the suggested system, in which the BeagleBone Black reads temperature data from an LM35 sensor and sends it via serial communication to a NodeMCU, which then forwards the data to the Blynk IoT cloud platform.

Accurate temperature readings are possible with the trustworthy analogue temperature sensor LM35. Temperature readings are precise and high accuracy thanks to the BeagleBone Black's accurate analog-to-digital conversion capabilities. Data transfer between the

BeagleBone Black and NodeMCU via serial communication turns out to be dependable. Efficient and error-free data transfer is made possible by the direct connection made possible by UART (Universal Asynchronous Receiver-Transmitter).

By getting temperature data from the BeagleBone Black and sending it to the Blynk IoT cloud platform, the NodeMCU acts as a useful middleman. Its ability to connect to cloud services is enabled by its built-in Wi-Fi, highlighting its versatility as an IoT device. Blynk is an intuitive Internet of Things platform that makes real-time monitoring, control, and data visualisation simple. A user-friendly interface for remotely viewing and managing temperature data is made possible by the integration of Blynk.

Figure displays the source code that was written in the Arduino IDE and the data that was continuously read from Beaglebone Black. The figure shows the Blynk IoT dash board with temperature data and actuator status.

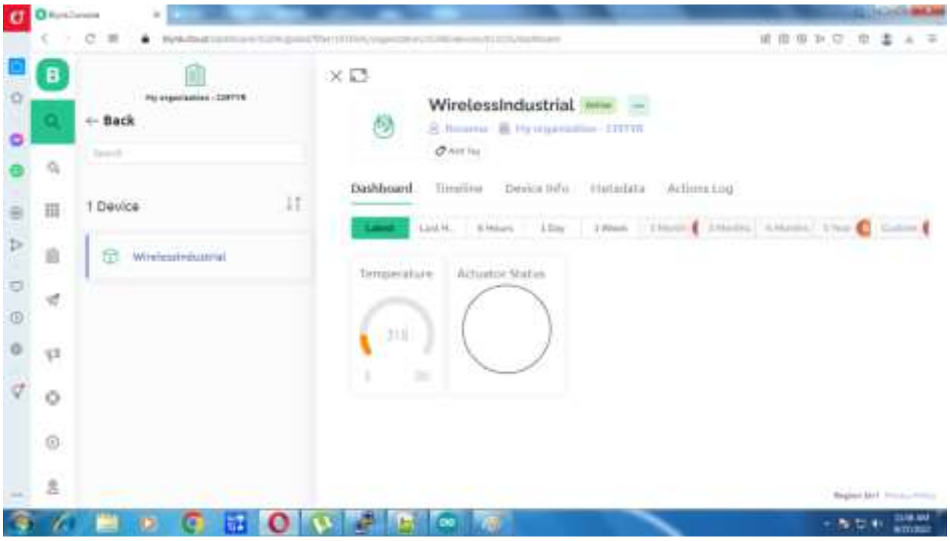

Figure : Screen shot of Blynk IoT dashboard

Conclusion :

It is possible to create an effective and scalable IoT solution, as shown by the BeagleBone Black's successful implementation of reading temperature data from an LM35 sensor and passing it to a NodeMCU for further forwarding to the Blynk IoT cloud platform. By using this technology, costs and space requirements will be reduced. One advantage of the Wireless Industrial Monitor and Controller Using Embedded Linux (RTOS) is that it may function autonomously in the case of a power loss.

An adaptable and scalable Internet of Things architecture can be created by connecting the NodeMCU and BeagleBone Black via a serial connection. Compared to the conventional approach of using PC-based Unix or Windows servers, the BeagleBone Black embedded web server is a practical solution for industry automation or smart homes to collect the data and replicate it with current and past data. The client asks that this be taken care of.

The NodeMCU board has also had a significant impact on the IoT development sector, helping countless individuals and companies to bring their ideas to life. This study demonstrates the possibilities for real-time monitoring and control in a variety of industries while highlighting the significance of accuracy, dependability, and security in IoT applications.

References:

[1]. Choi M-J, Ju H-T. An efficient embedded web server for web-based network element management. Network Operations and Management Symposium, 2000. NOMS 2000. 2000 IEEE/ IFIP; 2000 Apr; Honolulu. p. 187–200.
[2]. Manivannan M, Kumaresan N. Embedded web server and gprs based advanced industrial automation using linux rtos. Int J Eng Sci Tech. 2010; 2(11):6074–81.
[3]. Zhan mei-qiong and Ji chang-peng; "Research and Implementation of Embedded Web Server", International Conference on Multimedia and Information Technology, PP. 123-125, Dec 2008.
[4]. Mi-Joung Choi, Hong-Taek Ju, Hyun-Jun Cha, Sook-Hyang Kim, and J. Won-Ki Hong, "An Efficient Embedded Web Server for Web-based Network Element Management," International Journal of Network Management, Vol. 10, May 2000.
[5]. Zhang Wenya, "Design and Implementation of Network Video Monitoring System Based on Embedded Linux," Southwest Jiaotong University Master Degree Thesis, May 2009, pp. 147-148.
[6]. https://www.sparkfun.com/datasheets/Components/DS1307.pdf
[7]. https://learn.adafruit.com/adding-a-real-time-clock-to-beaglebone-black/overview
[8]. HTML The Complete Reference By Powell, Thomas A, Publisher MGH, ISBN: 0072129514.
[9]. PHP for the Web By LARRY ULLMAN, Publisher Peachpit Press, Copyright © 2011 by Larry Ullman.
[10]. MySQL Cook book, By Paul DuBois; Publisher O'Reilly on dated October 2002; ISBN:0-596-00145-2, pages 1022.