

## REMOTE MONITORING SYSTEM USING MQTT PROTOCOL

L.M. Varalakshmi<sup>1</sup>, R. Kurinjimalar<sup>2</sup> and A.Vijayalakshmi<sup>3</sup>  
<sup>1,2,3</sup>Department of ECE, SMVEC, Pondicherry, India

**ABSTRACT:** In this paper we discuss and create an Infrastructure management system, which can be implemented by connecting sensors to monitor various device activities and control those data from anywhere to provide greater flexibility to the user. By sensing the environment an appropriate solution can be provided before it leads to any major harm. The sensed signals from the BeagleBone Black Board are pushed to the MQTT protocol via Wi-Fi, as the emerging protocol of IoT is MQTT (Message Queuing Telemetry Transport) protocol. The MySQL Database continuously stores the received data from the protocol. The server continuously monitors the data stored in the database, calculates and displays it in webpage and analytics are made to fix the threshold so that any unconditional harm can be avoided and the environment is managed in a effective manner. With this infrastructure management system the user can create and deploy it in industrial as well as home automation purpose, as it provide security to database management system.

**Keywords:** Beaglebone black, Sensors, MySQL database, Internet of Things, MQTT, Airtel 3G modem

### INTRODUCTION

Internet of things (IoT) is no longer future tech trend because of an everyday growing number of objects connected to the Internet worldwide. IoT can be defined as a network of objects which are equipped with sensors and network access modules. The IoT technology uses various databases to store, manage and monitor the data [12-16]. A database is a assortment of data that is ordered so that it can easily be accessed, controlled, and restructured.

In addition, databases are sometimes classified according to their secretarial approach. The most common approach is the relational database, in which data is defined so that it can be simplified and accessed in a number of different ways. MySQL is a fast, easy to use RDBMS being used for many small and big businesses. Client/ Server Architecture of MySQL follows database server (MySQL) and arbitrarily many clients (application programs), which communicate with the server, that is, they query data, save changes,

etc. Here in this paper, the Infrastructure management refers to monitoring this, a sensor senses the status of an environment and updates to web server. The MySQL database is directly connected to and controlling the devices. Remote monitoring is widely held now days because it affords ease, security and efficiency. In the Eclipse IDE and the MQTT protocol to receive the data from board. If the user is distant from home, they can access and made any changes they need i.e. switching the AC on/off, switching lights on/off etc. User can use local PC, Laptop and smartphones. This paper will describe the approach of controlling the environment by using web server.

### HARDWARE SPECIFICATION

#### Beagle Bone Black

Beagle boards are tiny computers with all the capability of today's desktop machines, without the bulk, expense, or noise. It is a low-cost, community-supported development platform for developers and hobbyists. The best features of beagle bone black and its advantage over beagle bone is shown in the Table.1

COMPARISON	BEAGLEBONE BLACK	BEAGLE BONE
PROCESSOR	AM3358BZCZ100, 1GHZ	AM3359ZCZ72, 720MHZ
VIDEO OUT	HDMI	None
DRAM	512MB DDR3L 800MHZ	256MB DDR2 400MHZ
FLASH	4GB eMMC, Usd	uSD
SERIAL	Header	Via USB
PWR EXP HEADER	No	Yes
POWER	210-460 mA@5V	300-500 mA@5V
ONBOARD JTAG	Optional	Yes, over USB

Table.1 Comparison

Boot Linux in under 10 seconds and get started on development in less than 5 minutes with just a single USB cable.

As it is clear from the above table that beagle bone black is more faster than other versions of it. The Beagle bone black board is shown in the Figure 1 with the connector and switch locations.

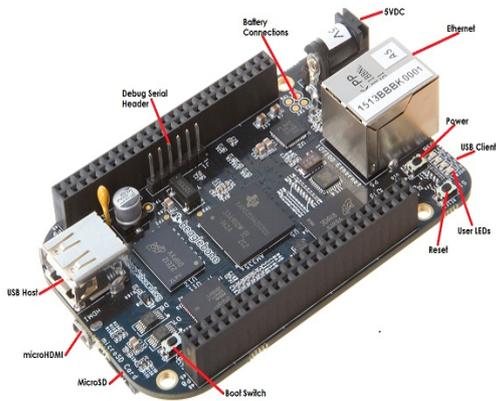


Fig. 1 BEAGLE BONE BLACK

**GPS Module and its Connection**

GPS receiver is found in most smartphones, many new automobiles, and they are used to track commerce all over the globe. The GPS receiver gets a signal from each GPS satellite. The satellites transmit the exact time the signals are sent. The GPS module has to be connected to the beagle bone black so that the device location can be identified remotely. The Fig 2 shows the connection between beagle bone black and GPS module.

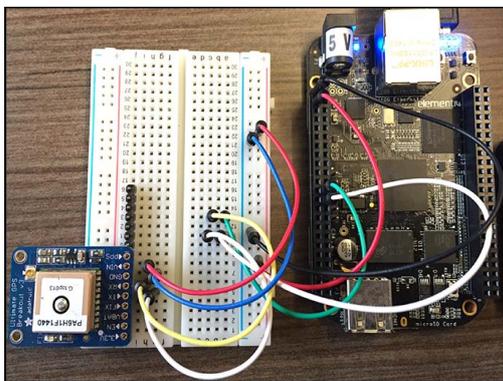


Fig. 2 GPS module with Beagle Bone Black

The current system should run in Debian or Ubuntu Linux on Beaglebone. The steps show the method on how to connect GPS module with the beagle bone black board.

- a. Attach the GPS breakout board using the breakout pins

- b. Connect TX (GPS Transmit Pin) to UART1\_RXD (pin 26 on header P9 BBB Receive Pin)
- c. Connect RX (GPS Receive Pin) to UART1\_TXD (pin 24 on header P9 BBB Transmit Pin)
- d. Connect GND to pin 1 on header P9
- e. Connect VIN to the 3.3V source at pin 3 on header P9.

**Temperature Sensor**

The DHT11 is a basic, ultra low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data. It uses 2.5mA max current during conversion. It is also good for 20-80% humidity readings with 5% accuracy and good for 0-50°C temperature readings  $\pm 2^\circ\text{C}$  accuracy.

**PCB Board and overall connection**

Printed circuit board is the most common name but may also be called 'printed wiring boards' or 'printed wiring cards'. It is a board that has lines and pads that connect various points together. The overall beagle bone black connection is shown in the Fig 3 which gives the detailed soldering of the sensors and GPS module.

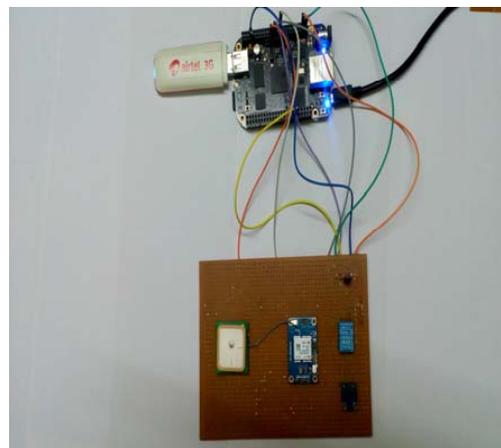


Fig. 3 Beagle Bone Black with sensors on PCB

The Beagle Bone sends the sensed data like Temperature, humidity and GPS values into the MQTT broker via Wi-Fi module. The data can be published using a specific topic.

```
defon_publish(mosq, obj, mid):
print("mid: " + str(mid))
```

```
mqttc.publish("topic", "mv=0 C=0 F=0")
```

By giving the connection details the specified topic will be published to the MQTT broker.

Thus the overall hardware components are accessed by the beagle bone black board and send the data continuously to the broker. The board can be accessed remotely from anywhere of the world.

**SOFTWARE SPECIFICATION**

**MQTT**

MQTT stands for message queuing telemetry transport. Unlike COAP, TCP, UDP, it is used because MQTT specializes in low-bandwidth, high-latency environments it is an ideal protocol for machine-to-machine (M2M) communication. Basically there are three components in MQTT publisher, subscriber and broker. The basic broker architecture that shows temperature sensor connected to it.

A Topic from the subscriber is send via broker to access the temperature of a particular environment.

The TCP/IP network connects the publisher and the subscriber as shown in Fig. 4

The Topic is subscribed from the server side and it is given from the client side. The subscription program is,

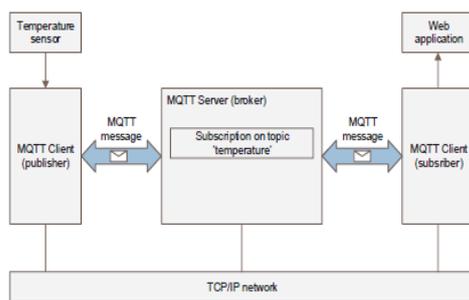


Fig. 4 MQTT Protocol Model

The subscription coding is,  
 client.subscribe("Topic");  
 MqttMessage message = new MqttMessage();

This Java coding is processed in the Eclipse IDE where Eclipse is the widely used Java IDE contains a base

[workspace](#) and an extensible [plug-in](#) system for customizing the environment. These data need a storage to save its value and here we are using apache MySQL database as a backend.

**MySQL Database**

MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages. Its standard form is well-known SQL data language. MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but can be increased to a theoretical limit of 8 million terabytes (TB).

This database stores the data that is subscribed from the MQTT Broker. To store the data in the database a connection need to be established between the Eclipse IDE and the database. Using JDBC driver an connection is made and subscribed information is stored in the MySQL.

**OpenGTS**

The stored data from the database should be displayed and viewed in the webpage. For this an web based source is required which is known as OpenGTS.

OpenGTS is a web based open source device [tracking](#) services framework OpenGTS is implemented in Java and requires an [database](#) server as a backend. It requires a Web-based authentication, i.e. each account can support multiple users, and each user has its own login password and controlled access to sections within their account. It supports the data collection and storage of device tracking and [telemetry](#) data from remote devices GPS tracking device independent: Devices from different manufacturers can be tracked simultaneously. The user "Role" administration support "System Administrator" account support providing the ability to create new accounts through the web-interface. "System Information" page which summarizes basic GTS system attributes. These are listed in the main menu page of the openGTS. The overall details about the device in located in the main menu itself.

**Block Diagram**

The overall method can be illustrated from the below architecture Fig. 5. This architecture explains the connection between the hardware and the software side.

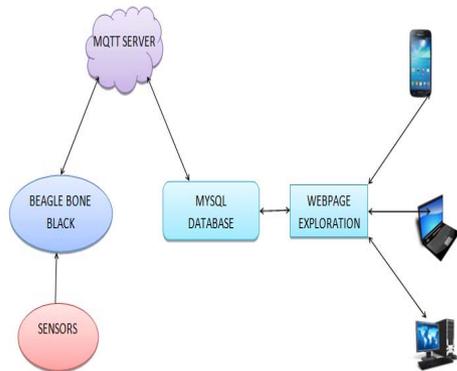


Fig. 5 Remote Monitoring Architecture

**RESULTS AND DISCUSSION**

**Connecting MQTT broker with MySQL Database**

The TCP port 1883 is used to connect the MQTT broker and the Eclipse. From this port with an equivalent IP address the data from the broker is stored in the database. Once the data is stored a message is popped up saying that new user is added to the database. The Fig. 6 Shows the connection between MySQL database and the eclipse.

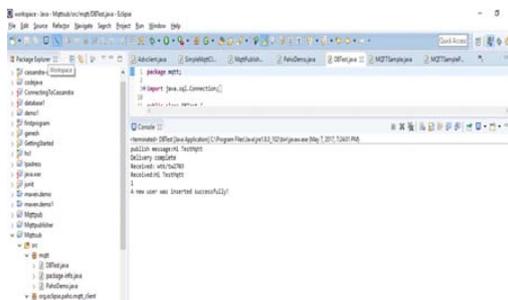


Fig. 6 MySQL with Eclipse

**Webpage Creation**

To provide basic user authentication an email, account name, password is required. So that the entire data received is kept with full security. The OpenGTS login page is shown in the Fig. 7.

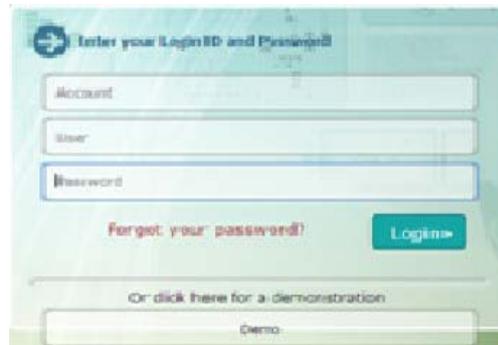


Fig. 7 User Authentication

**Reports**

GPS device tracking has its own independence that devices from different manufacturers can be tracked simultaneously. It has Customizable reports, using this an internal XML-based reporting engine, detail and summary reports can be customized to show historical data for a specific device. The Trip Detail Report displays the start time/location, and stop time/location, of various "trips" made within the selected date range. The Geozone/Geofence Report displays the arrival times and departure times of various travelled geozones/geofences, with the total amount of time spent within each geozone/geofence. The data with the latitude and longitude values are shown in Fig. 8

#	Date	Time	Status	Latitude	Longitude	Speed (K)	Address
1	2017-05-02	11:54:03	Location	11.889979, 79.3493		286.47	
2	2017-05-02	11:54:06	Location	11.889979, 79.3493		286.47	
3	2017-05-02	11:54:08	Location	11.889979, 79.3493		286.47	
4	2017-05-02	11:54:07	Location	11.889979, 79.3493		286.47	
5	2017-05-02	11:54:08	Location	11.889979, 79.3493		286.47	
6	2017-05-02	11:54:08	Location	11.889979, 79.3493		286.47	
7	2017-05-02	11:54:12	Location	11.889979, 79.3493		286.47	
8	2017-05-02	11:54:12	Location	11.889979, 79.3493		286.47	
9	2017-05-02	11:54:12	Location	11.889979, 79.3493		286.47	
10	2017-05-02	11:54:14	Location	11.889979, 79.3493		286.47	
11	2017-05-02	11:54:12	Location	11.889979, 79.3493		286.47	
12	2017-05-02	11:54:12	Location	11.889979, 79.3493		286.47	
13	2017-05-02	11:54:12	Location	11.889979, 79.3493		286.47	

Fig. 8 Reports

**Mapping Service**

Customizable mapping service provides OpenGTS comes with the support for [OpenLayers/OpenStreetMap](#) in addition to support for Google Maps, Microsoft Virtual Earth, and [Mapstraction](#). Web-service support for accessing the GTS database and querying report, mapping, and database information in XML format.

The location and the area of the device can be found from the latitude and longitude values that is sent from the GPS module. Using this values the appropriate location can be identified by the satellite and it is shown in the map service of OpenGTS. It is also available for Google map, Microsoft Virtual Earth. The Fig. 9 gives the clear look about the mapping of device.

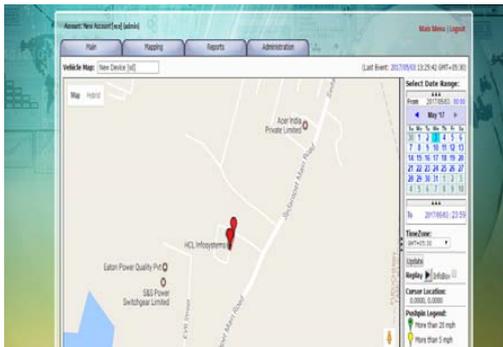


Fig. 8 Mapping Service

Thus the environment can be managed on time and all data that is received by the MQTT broker is stored in the database and equivalent reports are made in the table.

## CONCLUSION

The designed infrastructure management system provides a smart infrastructure-service architecture, which employs standard interface for devices at any infrastructure to separate the logic and user interface. Moreover, this system applies BeagleBone Black with community broker MQTT to provide smart services such as managing environment by deploying operations, reducing the manual labor, providing electronic information services, supporting diverse services, and extending the community's integration with the surrounding environment. Therefore, a complete and integrated infrastructure system can be achieved.

The whole setup and mechanism has a better and efficient advantage over the other protocols like COAP, TCP/UDP, and HTTP etc. With this infrastructure management system one can use and deploy it in industrial as well as home automation purpose, as it provides security to database management system.

## FUTURE SCOPE

In future many sensor [17-23] are added and by using sensed data notification can be send. The various application need variety of sensors to be deployed, by doing this one can able to achieve flawless managing system. And also the database can be changed to provide better access and storage facility.

## REFERENCES

- [1] YAN Wenbo, WANG Quanyu, GAO Zhenwei, "Smart home implementation based on internet and wi-fi technology" Control Conference (CCC), 2015 34th Chinese.
- [2] F. Xia, L.T. Yang, L. Wang, and A. Vinel, "Internet of things, " International Journal of Communication Systems, vol. 25, pp. 1101-1102, 2012.
- [3] Vladimir Vujovic and Mirjana Maksimovic, "Raspberry Pi as a Wireless Sensor Node: Performances and Constraints", Faculty of Electrical Engineering, East Sarajevo, Bosnia and Herzegovina, IEEE 2014.
- [4] D. Thangavel, X. Ma, A. Valera, H. Tan, and C.K. Tan, "Performance evaluation of MQTT and CoAP via a common middleware," Intelligent Sensors, Sensor Networks and Information Processing, April 2014.
- [5] Andy Stanford-Clark and Hong Linh Truong, Andy Stanford-Clark and Hong Linh Truong, "MQTT for Sensor Networks (MQTT-SN) Protocol Specification", November 14, 2013.
- [6] K. Tang et al., "Design and implementation of push notification system based on the MQTT protocol," in International Conference on Information Science and Computer Applications, September 2013.
- [7] Rossello-Busquet, A, Soler J, Dittmann L, "A Novel YAN Wenbo, WANG Quanyu, GAO Zhenwei, "Smart home implementation based on internet and wi-fi technology" Control Conference (CCC), 2015 34th Chinese.
- [8] MQTT, OASIS Standard, [Online]. Available: <http://mqtt.org/>
- [9] Ponte Eclipse Project, "Connecting Things to Developers," Available: <http://eclipse.org/ponte>, 2016 (accessed on Sep 21, 2016).
- [10] Eclipse.org., "Paho is an [eclipse.org](http://eclipse.org) project," Available:<http://eclipse.org/paho/> 2014 (accessed on Oct 21, 2016).
- [11] Eclipse Mosquitto. (2016, Oct. 12). [Online]. Available: <http://mosquitto.org/>- 63

- [12] Padmapriya, V., Gowri, V., LakshmiPriya, K., PremKumar, K., Thiyagarajan, B., "Perspectives, motivations and implications of big data analytics", (2015) ACM International Conference Proceeding Series, 06-07-March-2015, art. no. 2743099,
- [13] Rao, D.N., Sathian, D., Dhavachelvan, P., Raghav, R.S., Prem Kumar, K., "Big data scalability, methods and its implications: A survey of current practice", (2015) ACM International Conference Proceeding Series, 06-07-March-2015, art. no. 2743121,
- [14] Karthikeyan, P., Sathian, D., Raghav, R.S., Abraham, A., Dhavachelvan, P., "A comprehensive survey on variants and its extensions of BIG DATA in cloud environment", (2015) ACM International Conference Proceeding Series, 06-07-March-2015, art. no. 2743097,
- [15] Padmapriya, V., Gowri, V., LakshmiPriya, K., Vinothini, S., PremKumar, K., "Demystifying challenges, opportunities and issues of Big data frameworks", (2015) ACM International Conference Proceeding Series, 06-07-March-2015, art. no. 2743110,
- [16] Bandi, R., Gouse, S., "A comparative analysis for big data challenges and big data issues using information security encryption techniques1, 2", (2017) International Journal of Pure and Applied Mathematics, 115 (8 Special Issue), pp. 245-251.
- [17] Ganesan, M., PremKumar, A., KumaraKrishnan, S., Lalitha, E., Manjula, B., "A novel based algorithm for the prediction of abnormal heart rate using Bayesian algorithm in the wireless sensor network", (2015) ACM International Conference Proceeding Series, 06-07-March-2015, art. no. 2743118,
- [18] Pradeepa, D., Halima Begam, T., Rajaguru, D., Jaiganesh, S., Vengattaraman, T., "A contemporary research analysis on discrete wireless sensor networks routing algorithms", (2015) International Journal of Applied Engineering Research, 10 (3), pp. 2039-2042.
- [19] Satyanarayana, K.V.V., Kathavate, P., Reddy, L.S.S., "Energy aware routing protocol with QoS constraint in wireless multimedia sensor networks", (2017) Journal of Advanced Research in Dynamical and Control Systems, 9 (Special Issue 12), pp. 1449-1457.
- [20] Buvaneshvari, M., Uthayakumar, J., "Fuzzy based clustering to maximize network lifetime in wireless mobile sensor networks", (2017) Journal of Advanced Research in Dynamical and Control Systems, 9 (Special Issue 12), pp. 2133-2143.
- [21] Rajakumar, R., Dhavachelvan, P., Vengattaraman, T., "GWO-LPWSN: Grey Wolf Optimization Algorithm for Node Localization Problem in Wireless Sensor Networks", (2017) Journal of Computer Networks and Communications, 2017, art. no. 7348141, .
- [22] Uthayakumar, J., Vengattaraman, T., "A simple data compression algorithm for anomaly detection in Wireless Sensor Networks", (2017) International Journal of Pure and Applied Mathematics, 117 (19 Special Issue), pp. 403-409.
- [23] Uthayakumar, J., Vengattaraman, T., "Data compression algorithm to maximize network lifetime in wireless sensor networks", (2017) Journal of Advanced Research in Dynamical and Control Systems, 9 (Special Issue 12), pp. 2156-2167.



