Design of an Energy Efficient IoT enabled Smart System based on DALI network over MQTT protocol

[1]Aniruddha Sinha [2]Shubham Sharma [3]Puja Goswami [4]Vivek Kumar Verma [5]Munish Manas
[1,2,3,4]Department of Electronics & Communication Engineering, Tezpur University (A Central University)
Tezpur University (A Central University), Napaam
Tezpur, Assam, India.
[1]sinha.anu10@gmail.com, [2]shubham.boko31@gmail.com, [3]Pujagoswami999@gmail.com, [4]12vivek94@gmail.com
[5]Department of Electrical Engineering, Central University of Haryana,
Haryana, India
[5]msd.gkg85@gmail.com

***Abstract* - The term "Internet of Things" or "IoT" refers to a hot and virgin area that is gaining importance day by day with increasing advancement in technology. With the help of micro-controllers like Arduino and micro-computers like Raspberry Pi, cheap devices can be used to measure sensor data and send it over the internet. Also, DALI (Digital Addressable Lighting Interface) is a new module of communication between the electrical equipment and a server or computer that will result in an effective realization of a smart-grid. This paper covers the communication of a DALI network of a group of lights with a Raspberry Pi (broker) over MQTT (Message Queuing Telemetry Transport) protocol keeping the concept of IoT in the background. This method proved to optimize electricity usage by optimal controlling the intensity of LEDs at various floors of the CDAC smartbuilding.**

***Index Terms* – Internet of Things (IoT), DALI, MQTT, Paho C Client Library, Smart Grid, Raspberry Pi.**

I. INTRODUCTION

In line with the mainstream technological innovations the emerging IoT technology aims at improving the quality of life of general masses and the efficiency of the civic amenities in a sustainable fashion while optimising economic investments by private and public sector. The IoT applications can be found in various vital fields such as: energy, health, transportation, environment, etc. Millions of application in all these fields can work optimally if there is a strong interconnection among IoT enabled devices [1]. Interconnection among IoT enabled devices via communication channels and protocols is not only a simple technological subject matter but it interests also other characteristics such as privacy, standardization, legal issues, etc. [2]. This unsurprisingly induces newer innovative challenges in IoT environment, which inspires industrial and academics researchers to go further in IoT sustainability research [3]. Confidently, the technological interfacing and communications among the IoT enabled devices (things) is essential to empower the IoT environment [4]. This is the reason why few vital functions will be performed by few key technological components and techniques, which have evolved to guarantee interconnection among heterogeneous devices adapting to the usage of very less supplies in terms of both computational time, memory, functionalities and energy resources.

This paper offers an analysis report of a small project done by the authors at CDAC, Pune to configure the electrical arrangement of a building in a smart manner using the concepts of DALI and IoT.

Accordingly, the rest of the paper is organized as follows: Section II identifies the basic apparatus required for the project and a brief description about the various protocols and modules that have been exhausted in the project. Section III provides an overview of the method through which the project was carried out and the working principles of the project. Section IV presents a brief of the results achieved from the project and an analysis of how smart grids are efficient investments for people in the time to come. Section V consists of the conclusion and outlook. Following the conclusion is a short note of acknowledgement by the authors to the people who have helped during the course of this project.

II. SYSTEM MODELLING

A. Hardware and Software used

In order to carry out this project, we made use of the following set of hardware:

- A Raspberry Pi 3 Model B
- DALI LED drivers, preferably LCM 60 DA.
- Group of LED lights

The software comprised of:

- Linux/Debian terminal

978-1-5090-6218-8/17/$31.00 ©2017 IEEE

- Eclipse Paho client library for MQTT communication
- Any C programming IDE
- MQTT mosquitto broker

Before getting into the process of performing the project, it is imperative to acquaint ourselves with the basic concepts behind the modules and protocols in use:

1) MQTT: MQTT is a lightweight messaging protocol that provides resource-constrained network clients with a simple way to distribute telemetry information. The protocol, which uses a publish/subscribe communication pattern, is used for machine-to-machine (M2M) communication and plays an important role in the Internet of Things (IoT). MQTT allows devices to send (publish) information about a given topic to a server that functions as an MQTT message broker. The broker then pushes the information out to those clients that have previously subscribed to the client's topic. To a human, a topic looks like a hierarchical file path. MQTT is a good choice for wireless networks that experience varying levels of latency due to occasional bandwidth constraints or unreliable connections.

Should the connection from a subscribing client to the broker get broken, the broker will buffer messages and push them out to the subscriber when it is back online. Should the connection from the publishing client to the broker be disconnected without notice, the broker can close the connection and send subscribers a cached message with instructions from the publisher.

There are many platforms of MQTT protocol, the most popular being the HiveMQ and the Mosquitto platforms.

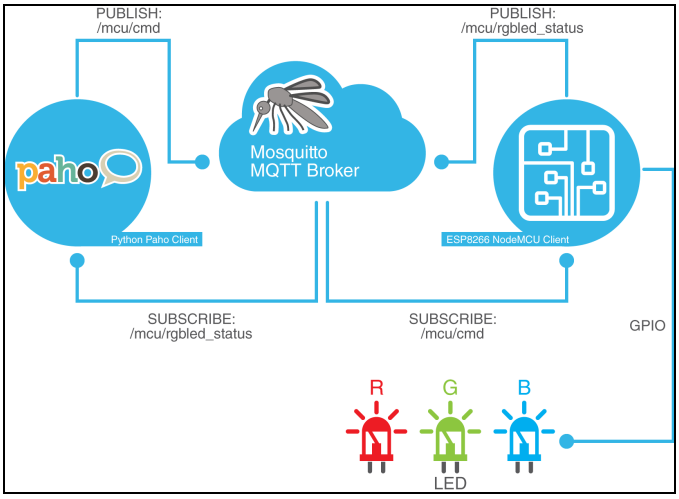


Fig. 1 Interaction of MQTT broker with the publisher and subscriber. [5]

In Fig.1, a Mosquitto MQTT broker acts as the server for communication between two clients (ESP8266 Node MCU and Paho Python), both of which can act as either a subscriber or a publisher. However, at a time, one will act as the publisher while the other will be the subscriber. This communication happens over a topic which is common to both ends. This maintains the singularity in communication over the same broker among various clients without any interference .

2) Eclipse Paho: The word ***pāho*** is a Maori word that means *to broadcast, make widely known, announce, disseminate, transmit*. The Paho project was created to provide scalable open-source implementations of open and standard messaging protocols aimed at new, exisiting, and emerging applications for Machine-to-Machine (M2M) and Internet of Things (IoT). Objectives include effective levels of decoupling between devices and applications, designed to keep markets open and encourage the rapid growth of scalable Web and Enterprise middleware and applications. Paho initially started with MQTT publish/subscribe client implementations for use on embedded platforms, and in the future will bring corresponding server support as determined by the community [6].

Paho client libraries can be written in many programming languages, viz. C, C++, Java, Python, etc. For this project, we performed Paho client programming in C.

3) *DALI*: DALI is a data protocol and transport mechanism that was jointly developed and specified by several manufacturers of lighting equipment. The common platform of DALI enables equipment from different manufacturers to be connected together. DALI network consists of a controller and one or more lighting devices (e.g., electrical ballasts, LED drivers and dimmers) that have DALI interfaces [10]. The controller can monitor and control each light by means of a bi-directional data exchange. The DALI protocol permits devices to be individually addressed and it also incorporates Group and Scene broadcast messages to simultaneously address multiple devices (e.g., "Group 1 goto 100%" or "Recall Scene 1"). DALI devices include LED drivers, fluorescent HF ballasts, low voltage transformers, PE cells, motion detectors, wall switches and gateways to other protocols. There can be up to 64 DALI devices on a single DALI network. Sites requiring more than 64 devices are implemented by having multiple separate DALI networks, each with up to 64 devices. These separate networks are then linked together with DALI gateways and a data bus running a high level protocol.

III. METHODOLOGY

This project works on the basic MQTT protocol, i.e. a publisher sends some data to an MQTT broker (created on the Raspberry Pi itself) which is further transferred to the subscriber. Both the publisher and subscriber are subscribed to the same topic. Each Raspberry Pi system is capable of handling at most 64 lights (clients) as per the DALI protocol.

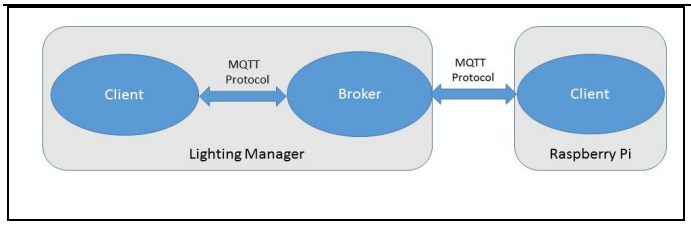
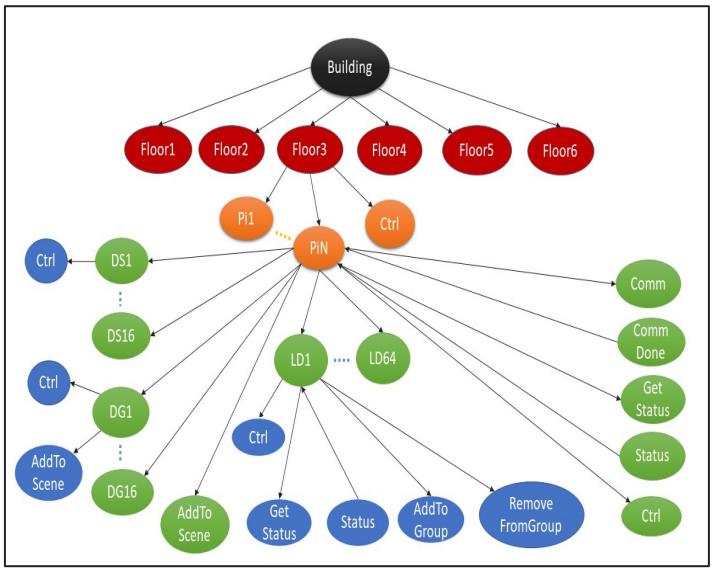


Fig. 2 General communication between the terminals in the project

Fig.2 gives a pictorial representation of the skeleton of the project where Lighting manager represents the console of DALI drivers, each of which connects to maximum LEDs



(nodes).

Fig. 3 Sequential range of topics undertaken by current research methodology for optimal energy saving with more than 64 devices DALI Network

The first stage in the project is to begin with the installation of Paho C client library on the Raspberry Pi. Thereafter, the mosquitto MQTT platform is installed. Once the installation process is over, the structure of topics is brought into existence. As is evident from Fig.3, the range of topics available is diverse. Each topic corresponds to a different set of topic. The hierarchy of topics is as follows:

a) *Building -> Floor n -> Pi n*: Each floor can have many Raspberry Pis to drive numerous DALI drivers. Hence, once the communication happens over this topic, the user has the option to control the state of either one or more than one LEDs lights on that floor through the corresponding Pi.

b) *Building -> Floor n -> Ctrl*: If the user wishes to control the entire floor and all Raspberry Pis and DALI drivers on that floor, this topic will be sent.

c) *Building -> Floor n -> Pi n -> Comm*: This topic is sent by the publisher to start the communication with the subscriber (DALI drivers).

d) *Building -> Floor n -> Pi n -> Comm Done*: This topic returns a callback signal to the broker whether successful communication has been formed with the client. If not, the broker will retry communication with the client until it's done.

e) *Building -> Floor n -> Pi n -> Get Status*: If the user wishes to know the current brightness status of the lights or of a particular DALI driver, this topic will be communicated with.

f) *Building -> Floor n -> Pi n -> Status*: The client returns its current status to the user as a return call to the above topic.

g) *Building -> Floor n -> Pi n -> Ctrl*: The user can get control over a single Pi on a particular floor and control all the DALI drivers with a single command through this topic.

h) *Building -> Floor n -> Pi n -> DS N -> Ctrl*: The term DS in this topic implies 'DALI Scene' and N represents the particular DALI driver that has 4 nodes. This topic enables the user to only control the brightness scene and make changes to it, perhaps like moderating the dim and bright lighting according to the situation. For instance, if it is a projection room, the lights are bright initially but may be automatically dimmed by the user once the projection of a presentation or movie begins in the room.

i) *Building -> Floor n -> Pi n -> DG N ->Ctrl*: The term DG here signifies 'DALI Group' and N is same as in the above topic. Here, there are total 16 DGs under each Pi and each DG consists of 4 nodes. We can control the settings and state of each group here, however cannot change the scene settings.

j) *Building -> Floor n -> Pi n -> DG N -> Add to Scene*: This topic is useful when the user wishes to access a particular DG and add some brightness settings to its current scene without altering it.

k) *Building -> Floor n -> Pi n -> Add to Scene*: Instead of changing the scenes of each DG or DS, one can also add settings to the scene of all 64 nodes under one Pi through this topic.

l) *Building -> Floor n -> Pi n -> LD N -> Ctrl*: The term LD in this topic refers to 'LED DALI', i.e. instead of having the nodes (LEDs) as groups, they can also be individually placed under a Raspberry Pi

thereby giving access to a Pi to 64 LDs. Hence, with this topic the state of the LDs can be controlled individually.

m) *Building -> Floor n -> Pi n -> LD N -> Get Status*: Same as topic e), only difference being that an individual LD's status is being sought for this time by the user.

n) *Building -> Floor n -> Pi n -> LD N -> Status*: Similar to topic f), this one returns the status of the LD which was sought by the user in the earlier topic.

o) *Building -> Floor n -> Pi n -> LD N -> Add to Group*: There is also an option to add the individual LDs to an existing DG or into a new one, keeping in mind that maximum number of nodes in each group should not exceed 4.

p) *Building -> Floor n -> Pi n -> LD N -> Remove from Group*: The LDs can be removed from the groups that they are a part of too.

How does it work?

The above range of topics describe the vast scope of available options to the user for controlling and monitoring a network of lights either in a home or even in a building. However, this is achieved through programming in Paho C client library.

We program both ends of the broker in such a way that firstly a client is created. A user-friendly environment is developed where the topics are listed on the screen, and the user can select a topic of his choice. Once the topic is also initialized, the subscriber client however tries connection with all the topics. Depending on the topic that the publisher has connected to, the subscriber reaches out to the set of topics corresponding to that topic and returns its status to the publisher via the broker.

Depending on the choice entered by the user, the corresponding topic gets generated on the publishing side. This choice of topic is sent to the MQTT broker (Raspberry Pi), which further channelizes the communication of data with the subscriber over this topic. The subscriber however generates all possible topics and tries to connect to each one of them at very short intervals after creating the client. The communication takes place when the topics of both publisher and subscriber match with each other. At this instant, the message from the publisher is published to the subscriber, and the resulting set of operations corresponding to that topic are performed. Eventually, we finally see the result physically with the varying brightness level of the lights or LEDs. The same process continues and various publishers can publish to the same subscriber over different topics. Also, the connection process is made in such a way that even if the connection is lost or broken from either sides, the client tries to reconnect with the broker and does not disconnect, unless disconnected manually by the user.

IV. RESULTS

Although we approached this project at its beginning stage, yet by implementing the proposed methodology many results were realized by us towards the end of it. The terminal windows consisting of the publishing and subscribing operations has been shown in section III of this paper.

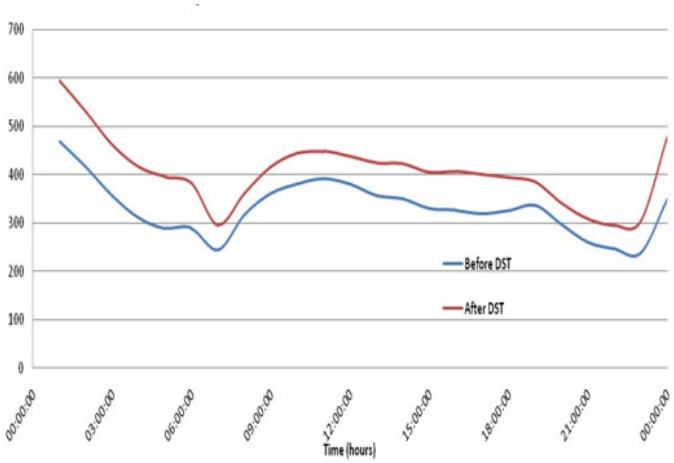


Fig. 4 Graphs of energy saving (in kWh) on Y-axis before and after DALI Scheme Transfer (DST) in a day

The analytics of this shows that on the peak periods of usage, the load can be increased and decreased vice versa via DALI controller. Also, the user desired topic can be sent to the MQTT broker (Raspberry Pi), which further publishes the data to the subscriber for optimally setting the brightness of LEDs as shown in Fig. 5a and 5b. This causes the energy saving in the electricity usage as shown in Fig. 4, and this will play a huge role in the energy saving in urban areas paving the way for a smarter demand response on consumer side.

```
pi@raspberrypi:~/cdac_dali_project/publisher $ ./publish

Enter the Floor to control :    6

Enter the Pi number for the control :   6


Enter your choice with the  number for its corresponding operation :

        [0]. Control the entire floor : 'Building/Floor No/Ctrl'
        [1]. Control the particular Pi with the given Clientid : 'Building/Floor No/Pi No/Ctrl
        [2]. Communicate the Pi : 'Building/Floor No/Pi No/Comm'
        [3]. Get status of the lights : 'Building/Floor No/Pi No/Get Status'
        [4]. Add the lights to scene : 'Building/Floor No/Pi No/AddToScene'
        [5]. Send Control to the lower level : 'Building/Floor No/Pi No/(LD,DG,DS)/Ctrl'
        [6]. Lower group, get its status : 'Building/Floor No/Pi No/(LD)/GetStatus'
        [7]. Lower group, add to scene : 'Building/Floor No/Pi No/(DG)/AddToScene'
        [8]. Lower group, add to group : 'Building/Floor No/Pi No/(LD)/AddToGroup'
        [9]. Lower group, remove from group : 'Building/Floor No/Pi No/(LD)/RemoveFromGroup'
3
Enter the value(0 | 1) for getting status
1
led_mqtt_service started

Press Ctrl+C within 4 seconds to exit from the program.

Connection successful

Publishing to topic Building/Floor6/Pi6/GetStatus


Publish successful

Enter the Floor to control :
```

Fig. 5a Terminal window for the publisher

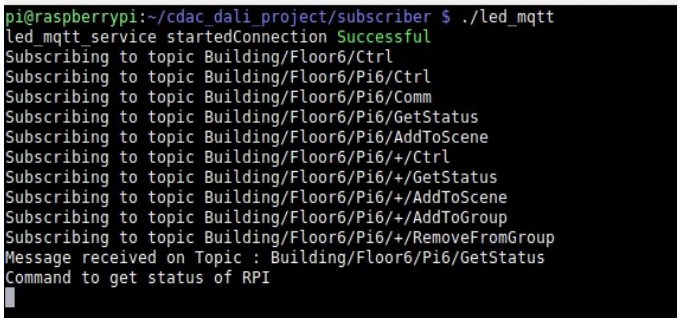


Fig. 5b. Terminal window for the subscriber

V. CONCLUSION

The coming decade is going to witness a revolutionary change in the manner of usage of energy and smart monitoring systems are going to come in vogue. We have been able to successfully implement a smart energy saving methodology via communication of a DALI network with a Raspberry Pi over MQTT (Message Queuing Telemetry Transport) protocol keeping the concept of IoT as the background for optimizing the brightness control of LEDs at different floors in the smart building of CDAC Pune. The constraint of using only 64 DALI devices on a single DALI network was overcome by having multiple separate DALI networks, each with up to 64 devices in the entire set up with reduced delay and maximum energy saving.

ACKNOWLEDGMENT

The internship opportunity at CDAC, Pune was a great chance for us to learn as well as to develop our concepts in the area of IoT and Smart Energy Management. Special thanks to Mr. Ashish Kulkarni (HoD, Embedded Systems, CDAC Pune), Mr. Tarun Deep Kohli (Joint Director, CDAC Pune) and Mr. Mayur Kulkarni (Project Engineer, CDAC Pune) for guiding us throughout the course of this project and helping us with new ideas.

REFERENCES

[1] Zaidi, Syed Ali Raza, et al. "Enabling IoT empowered smart lighting solutions: A communication theoretic perspective." Wireless Communications and Networking Conference Workshops (WCNCW), 2014 IEEE. IEEE, 2014.

[2] Zhang, Yuejun, Ping Zhou, and Mingguang Wu. "Research on DALI and Development of Master-Slave module." 2006 IEEE International Conference on Networking, Sensing and Control. IEEE, 2006.

[3] Alkar, Ali Ziya, and UmitBuhur. "An Internet based wireless home automation system for multifunctional devices." IEEE Transactions on Consumer Electronics 51.4 (2005): 1169-1174.

[4] Kovatsch, Matthias, Markus Weiss, and Dominique Guinard. "Embedding internet technology for home automation." Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on. IEEE, 2010.

[5] http://www.allaboutcircuits.com/uploads/articles/Introduction-to-the-MQTT-Protocol-on-NodeMCU-(2).png

[6] https://wiki.eclipse.org/Paho

[7] Munish Manas, " Development of preferential regulations, transmission tariffs, and critical technological components for the promotion of smart grid globally", *Economics and Policy of Energy and the Environment*, Franco Angeli Inc.(SCI Indexed), vol. 75, issue 2, pp. 107-130, 2015, DOI: 10.3280/EFE2015-002008.

[8] Zaheeruddin and Munish Manas, "A New Approach for the Design and Development of Renewable Energy Management System through Microgrid Central Controller", Energy Reports, Elsevier Inc., vol. 1, pp. 156-163, 2015.

[9] https://en.wikipedia.org/wiki/Digital_Addressable_Lighting_Interface

[10] Zaheeruddin; Manas, M., "Development of Self-Sustainable Technologies for Smart Grid in India," in Computational Intelligence & Communication Technology (CICT), 2015 IEEE International Conference on , vol., no., pp.563-568, 13-14 Feb. 2015.

[11] Zaheeruddin and Munish Manas "Analysis of Design of technologies, tariff Structures and regulatory policies for sustainable growth of the Smart grid" in Taylor and Francis's "Energy Technology and Policy" Journal, Volume 2, issue 1, pp. 28-38, 2015.

[12] B. Saharia and Munish Manas, "Viability analysis of Photovoltaic/Wind hybrid standalone Distributed Generation based system as applicable in isolated community of north-east region of India", Distributed Generation and Alternative Energy Journal , vol. 32, issue 1, PP. 49-80 • January 2017.

[13] W. Wu, M. Wu and Y. Liu, "A Design of Embedded DALI Controller," *2006 4th IEEE International Conference on Industrial Informatics*, Singapore, 2006, pp. 1237-1240. doi: 10.1109/INDIN.2006.275816