



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** II **Month of publication:** February 2022

DOI: <https://doi.org/10.22214/ijraset.2022.40436>

www.ijraset.com

Call: ☎ 08813907089

E-mail ID: ijraset@gmail.com

The Design of Co-operative Communication model using esp8266 for triangulation of network model for IoT Environment

Payal Badwaik¹, Harshal Nipane², Anjali Dhopte³, Saurabh Kaypurkar⁴, Prof. Vivek. N. Mahawadiwar⁵

^{1, 2, 3, 4}Student, Department of Electronics Engineering, K.D.K.C.E. Nagpur, India

⁵Professor, Department of Electronics Engineering, K.D.K.C.E. Nagpur, India

Abstract: *The Internet of Things (IOT) allows objects to connect over the internet in order to gather and exchange data. The hardware Arduino, sensors, selectors, and internet comparability are commonly used in these devices. In this context, the MQTT is critical for exchanging data or information between devices in the Internet of Things without knowing one other's identities. This paper discusses various communication service models in the Internet of Things. Model A uses serial USB as a transmission medium, whereas Model B employs the Message Queuing Telemetry Transport protocol, which connects the system to the internet (ESP8266-12). The concept of author and subscriber is utilized in communication. A agent server is used to bring out or subscribe to messages. The agent is in-charge of disseminating messages to consumers based on the message topic. In MQTT, the agent is also known as the server. Mosquitto, Adafruit, and hiveMQ are some of the agents utilized in MQTT. We have demonstrated by the applications of triangulation network for NODE-MCU using co-operative communication using hardware with MQTT protocol.*

I. INTRODUCTION

Internet of Things provide the ability to connect a large number of things or devices or internet. These things or devices have solitary identities. In IOT lightweight protocol like MQTT is used for data transmission. It is a author and a subscriber based protocol which allows multiple devices to communicate with each other over a wireless network. In Message Queuing Telemetry Transport author and subscriber do not busted to know each other's identity. It is preferably suited for the IOT nodes which have constrained abilities and assets. Any of the MQTT relation consider two types of agents: the first is consumers of MQTT and another is MQTT agentserver. The Information disseminated by protocol is known as application message. A author can send the application messages and subscriber can request for that application messages to get the information associated with that message. The agent permits the various consumers to connect with each other.

The bring outers have talked about the Arduino service interface programming model, a modern programming model that gives an administration deliberation to effectively compute new capabilities to microcontroller and furthermore offer help for networked boards deploys a scope of systems, including Message Queuing Telemetry Transport, socket relations and so on.

II. WIRELESS NETWORK

To deploy a wireless network, first operators decide the type of wireless network to be deployed by considering expected number of users to serve, type of traffic, the amount of throughput to provide and the number of access points or cells needed and placement positions to achieve optimal coverage.

This is done by interviewing stakeholders or by using network monitoring tool. This enables the operator to gather information about the goals and expected uses of the wireless network. Secondly, the operator carries out a predictive site survey using a planning tool to estimate how many APs or cells are needed, where to place them and the required power levels to set so as to meet the coverage and capacity requirements.

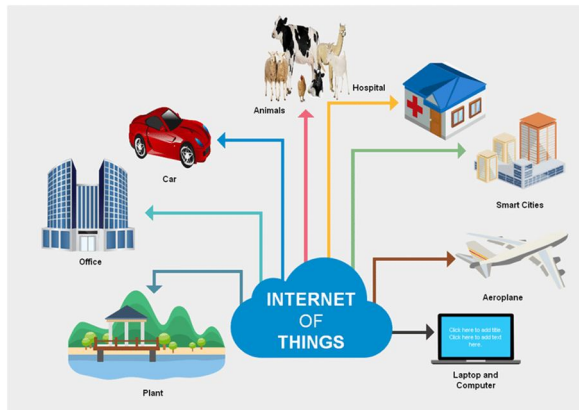
Lastly a manual real-time site survey is carried out to verify the accuracy of predictive survey to the real environment of application. Necessary adjustments are then made to realize the network design goals and deliver a wireless solution that meets the level of service required. This traditional network deployment strategy results in a well organized uniform cell/AP of known as a homogeneous network which is not a sufficient solution for capacity in present day wireless networks.

III. TECHNOLOGY

A. Internet of Things

Internet of Things (IoT) is a new approach to wireless communication where numerous devices are interconnected using different wireless communication technologies like WiFi, bluetooth, ZigBee and cellular network all together to make the world a smart and small place, closing the gap of space. LoRaWAN is a low power WAN technology that appears promising for Internet of Things. In a LoRaWAN, thousands to millions of smart devices are connected to the gateway through only one hop of connectivity. A good architecture for IoT technology is shown and explained in and an overview of bluetooth technology was explored.

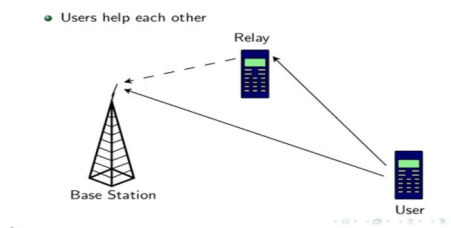
The IoT concept is very popular these days because of the availability of low power, low cost sensor nodes and their numerous applications like different system management like street light, building, parking, energy efficiency, fire fighting, home automation, personal assistant system, smart lighting solution, weather monitoring, agriculture, smart city, sports.



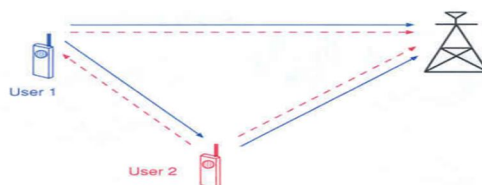
In all these applications, numerous sensors are distributed around the coverage space and then linked in chunks of mesh network with APs to communicate with the sensors and extend data to the coordinator/gateway or to another AP. The target is that all sensors in the network transmit their data through to the gateway which takes the data over to the Internet. If an AP is overloaded, data is lost, therefore, monitoring the network to identify locations where more APs are busted would reduce the risk of data loss.

IV. DESIGN

- 1) In CCCS each wireless user is assumed to transmit data as well as act as a cooperative agent for another user.
- 2) Trade-offs in code rates and transmit power busted.



- 3) More power is busted because each user is transmitting for both users, however, the baseline transmit power for both users may be reduced because of diversity gain.
- 4) A user transmit both the own bits as well as some information for the co-worker, but the spectral efficiency of each user enhance due to cooperation diversity the channel code rate can be increased.



- Question: is cooperation worth?
 - YES, as it will be later shown.

V. IMPLEMENTATION DETAILS

The implementation details for various service models.

Requirement for setup:

- 1) Node MCU esp8266
- 2) Arduino
- 3) LED (Light Emitting Diode)
- 4) Breadboard
- 5) Wires (Male to Female)
- 6) Laptops with internet relation
- 7) 420 OHM resistor

a) *Service model A:* Communication in IoT using serial correlation.

Connect client to Arduino(micro-controller) using a Serial relation(USB).

In Figure 1, three led are connected to Arduino in pin number 3,4,5 respectively. These LED turn off or turn on according to the instruction given by the system.

This is the Service model of communication between the objects with the help of USB or wire or without Wi-Fi.

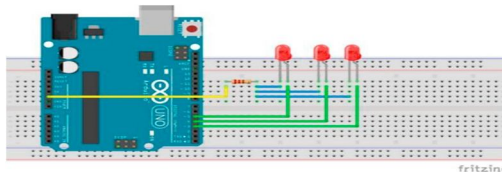


Fig. 1: relation of Arduino

b) *Service model B:* Communication in IoT using Wi-Fi or internet (MQTT protocol)

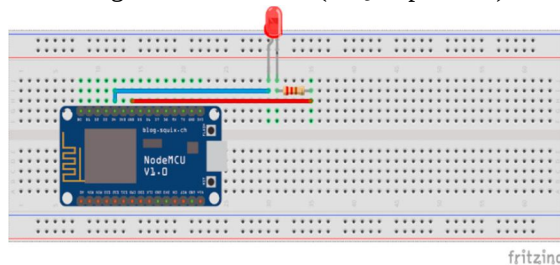


Fig. 2: LED Circuit for the subscriber side

Setup an MQTT relation using NodeMcu shown. Node MCU is connected to a LED on the breadboard in Figure 3. This Node MCU is connected to a system and this setup is connected to MQTT broker for further communication to perform the following tasks:

- Bring out and subscribe a message using MQTT broker on the same machine.
- Bring out a message on MQTT broker and subscribe to a topic from the server to obtain the information .
- LED circuit (as shown in figure 3) is linked on the side of subscriber. Purpose of this Service model is to build a MQTT bring outer-subscriber system so that the bring outer or client bring outes digital values (0 and 1) to the broker and then the subscriber subscribe to the bring outer's topic, information of that bring outed message is then delivered to the subscriber through the broker. The subscriber's side LED will be turned ON or OFF according to the values entered on the client side.
- Design one bring out and different subscribers, different bring outers and one subscriber & different bring outers and subscribers using MQTT along with it also observe that if there is a loss of data or lag while bring outing messages from two subscribers. In this service model, we have exploited the services of MQTT which can be used to communicate between respective set of devices. Client bring out a message and forward it to multiple subscribers. Also, we have presented a service model where client receive message from respective subscribers and observe if there is any lag or loss of data due to simultaneousness. In this manner, we have created a network of multiple bring outers and subscribers.

The Difference between all the above service models will be in their code sketch. Algorithms used:

Step No. 1: Connect to the Internet.

Step No. 2: If not connected to the internet, attempt correlation.

Step No. 3: Else print the IP address of the device, on the serial monitor.

Step No. 4: Attempt MQTT relation with the specified MQTT Broker.

Step No. 5: Once connected we can bring out the message on the broker.

Below algorithm is applied for all the service model with MQTT in this paper.

Wiring Instructions for setup:

- Take a breadboard.
- Next, insert LED into the breadboard.
- Just Connect the cathode of the LED to the output pin of the Node MCU.
- The anode of the LED is connected to the ground pin of Node MCU.
- The Node-MCU is powered by USB.

A. Single Bring Outer and Multiple Subscribers

In case of multiple subscribers, the object name of a subscriber acts as the group identifier. The MQTT broker will send information to only one member of the group at once and to all the members in a round robin fashion. Therefore, if we intend to have multiple subscribers, subscribing to the same bring outer we must give different object names to the subscribers. The identification of the subscribers is independent of the network they are connected.

B. Multiple Bring outers and Subscribers

Majority of the code will remain same. Difference will be that each bring outer will bring out multiple topics under “client. Bring out” and each subscriber will subscribe to multiple topics using client Subscribe.

In all the service models discussed, we observed that though there are some lags, but there is no loss of data while transfer of multiple information over MQTT.

C. Node MCU esp8266 (Version 1.0)

Node MCU (see figure no 4) is an open source firmware and development kit that helps in prototyping IoT. It has the following features:

- 1) Arduino like hardware I/O
- 2) USB TTL included, plug and play
- 3) Lowest cost Wi-Fi



Fig. 3: Node MCU esp8266

D. Software Origination

Arduino IDE is adopted as a software for programming. HiveMQ uses two services (Pub and Sub) to bring out and subscribe to a messages. MQTT, an android application is another MQTT client that associates with the Mosquitto MQTT broker and sends or distributes to a particular topic.

VI. DESIGN

In Service model A, Arduino is used for communication between computer and LED. There are three LEDs connected to the Arduino. The communication here is done via serial USB rather than an internet or Wireless Fidelity.

In Service model B, MQTT server is set-up using HiveMQ.

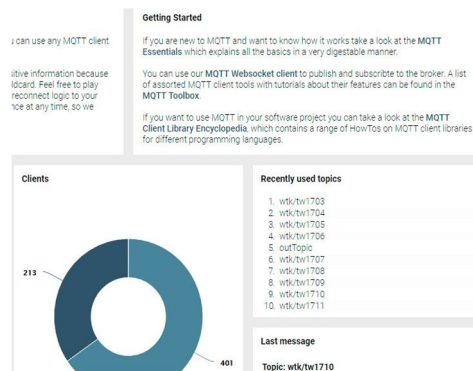
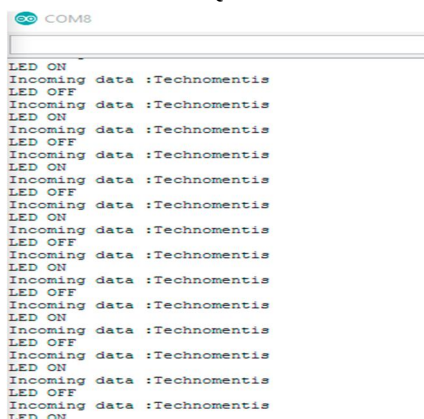


Fig. 4: Bring outed data on MQTT server (OutTopic)

In figure No. 3, serial monitor of Arduino IDE is showing both the bring outed and subscribed messages. Here the bring outed message is 'outTopic' and the subscribed message is 'Hello World' and this bring outed message appears in the Mqtt broker (mosquitto).

Figure 4, demonstrates subscription to a topic from the HIVE MQ broker.



The screenshot shows the serial monitor output of an Arduino. It displays a series of messages: 'LED ON', 'Incoming data : Technomentis', 'LED OFF', 'Incoming data : Technomentis', and so on, repeating several times. The messages are received over a COM5 connection.

Fig. 8: Serial monitor output for LED status

In Service Model B, LED will turn ON or OFF according to the command (e.g. ON or OFF) given by the bring outer.

The serial monitor is presented in figure 8: Figure 9 shows the output of Service Model B (point no. 4). Bring outer window shows the messages that are bring outed by the client. Figure 10 shows the subscriber windows and the information of the subscribed topic.



Fig 9: Bring outer's Serial Monitor



Fig 10:- Hardware setup 1

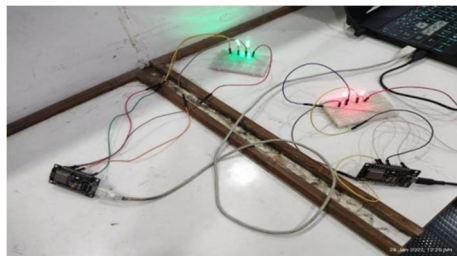


Fig 11:- Hardware setup 2

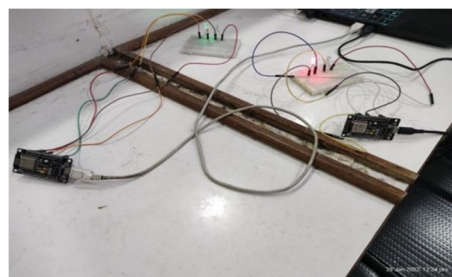


Fig 12:- Hardware setup 3

Fig 11 to fig 14 shows the co-operative communication prototype build on 2 nodemcu using MQTT protocols which has been optimized for the formation of the triangular network and doing high-end data bandwidth.

VII. CONCLUSION

This MQTT protocol permits the correspondence between devices. This paper focussed on the execution of Arduino, NodeMcu and MQTT protocol and also describe different possible service models for communication among the IoT devices. We test this protocol by bring out and subscribe the message on the same machine, different machine, and then we put up a bring out on one device and a subscriber on another device. LED is connected to the subscriber, this led ON or OFF according to the message bring out by the client on the broker and also examine the MQTT protocol with various cases like create one bring out and different subscribers, different bring outers and one subscriber & different bring outers and subscribers along with it also observe that if there is lost information or slack while distributing messages from two subscribers without a moment's delay. Therefore we found that MQTT protocol is suitable for transfer information is less than the serial relation because it depends on the speed of Wi-Fi.

REFERENCES

- [1] Number of 4G mobile subscribers worldwide 2010-2020 | Forecast.
- [2] Cisco. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper - Cisco, 2017
- [3] Seiamak Vahid, Rahim Tafazolli, and Marcin Filo. In Fundamentals of 5G Mobile Networks, pages 63–104. John Wiley & Sons, Ltd, Chichester, UK, may 2015.
- [4] Muhammad Zeeshan Shakir, Mohamed-Slim Alouini, Khalid A. QaErchin Serpedin, and Muhammad Ali Imran. The Heterogeneous small-cell networks toward reducing the CO2 emissions of mobile communications industry using uplink power adaptation. IEEE Communications Magazine, 51:52–61, jun 2013.
- [5] Symposium on Wireless Communications Systems (ISWCS), pages 465–469. IEEE, aug 2014.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)