

SIMULAÇÃO DE APLICAÇÕES UTILIZANDO O PROTOCOLO DE COMUNICAÇÃO MQTT COM APLICAÇÕES EM AMBIENTES INDUSTRIAIS

Remington Phelipe da Silva Correa, Márcio José da Cunha, Marcelo Barros de Almeida, Josué Silva de Moraes

Laboratório de Automação Sistemas Eletrônicos e Controle - LASEC, Faculdade de Engenharia Elétrica - FEELT, Universidade Federal de Uberlândia – UFU, Uberlândia - MG

remington@ufu.br, mjcunha@ufu.br, marcelo.barros@ufu.br, josue@eletrica.ufu.br

Resumo - O objetivo deste trabalho é apresentar a utilização do protocolo de comunicação MQTT na emulação de aplicações com possíveis aplicações em ambientes industriais. Essa comunicação será realizada através de aplicações emuladas com auxílio da linguagem de programação Python, que irão realizar troca de informações entre si sem um mediador humano, assim como ocorre em comunicações IoT. Espera-se que com este trabalho haja uma maior pesquisa sobre os protocolos de comunicação IoT utilizados em ambientes industriais, bem como fomentar a utilização desses protocolos para a implementação de redes mais modernas, podendo assim trazer novidades para as indústrias brasileiras.

Palavras-Chave – MQTT, Ambientes Industriais, Protocolo de Comunicação, Redes Industriais, Internet das Coisas.

THE USE OF MQTT COMMUNICATION PROTOCOL APPLICATIONS ON INDUSTRIAL ENVIRONMENTS

Abstract - The objective of this work is to present the use of MQTT communication protocol emulating applications possible applied on industrial environments. This communication will be done through emulated application with the Python programming language, made to exchange information between them without a human intermediary, the same way that happen on IoT communication. Based on this work is expected to stimulate the research about IoT communication protocols used on industrial environments as well as encouraging the use of these protocols to improve industrial networks and being able to bring news to the Brazilian industries.

Keywords – MQTT, Industrial Environments, Communication Protocol, Industrial Networks, Internet of Things.

I – INTRODUÇÃO

A nova tendência de se conectar todos os dispositivos de um ambiente industrial com o objetivo de unificar todas as informações em um só local, para permitir que as melhores decisões sejam tomadas em tempo real com ou sem intervenção humana, provocará uma grande mudança na realidade dos ambientes industriais nos próximos anos [13]. A Base para essa nova revolução está no conceito de Internet das Coisas.

A Internet das Coisas (*IoT*) baseia-se em conectar todos os dispositivos, sejam eles softwares ou hardwares, com a Internet formando uma rede totalmente inteligente e automatizada [13]. Isso permite com que todas as informações sejam compartilhadas entre todos os dispositivos dessa rede, o que possibilita que os dispositivos aprendam sobre todos os eventos que ocorrem na rede, possam realizar previsões estatísticas e assim, tomar decisões para que eles fiquem imunes a alguma possível falha futura [7], [13], [14]. Como todos os dispositivos estão conectados à internet, haverá uma facilidade cada vez maior de um operador realizar um controle do sistema remotamente [14].

Uma das mais diversas áreas presentes na IoT está na sua aplicação em ambientes industriais, a *Industrial Internet of Things* (IIoT), que trata da integração de toda uma rede industrial com dispositivos trocando informações entre si sem a necessidade de operadores humanos intermediando tal comunicação.

Como o cenário IoT é relativamente novo na indústria brasileira, pesquisas relacionadas ao desenvolvimento e implementação de uma rede de comunicação industrial com a tecnologia IoT, visando o menor custo possível de implementação, desenvolvimento e que possibilite novas formas de comunicação entre os mais diversos protocolos de redes industriais, são um dos melhores caminhos para o avanço dessa área.

O conteúdo deste artigo tem como objetivo servir como pesquisa a respeito desse novo cenário de comunicação industrial, uma vez que pesquisas relacionadas a essa área são de extrema importância no país pois elas possibilitarão o



desenvolvimento da futura indústria brasileira através de uma modernização dos processos de fabricação e monitoramento.

Na Seção II será apresentado o detalhamento do conceito de Industrial Internet of Things, bem como a apresentação do protocolo de rede MQTT (Message Queue Telemetry Transport), utilizado para as aplicações desenvolvidas na Seção III.

Na terceira seção, será descrito um problema proposto para utilização do protocolo MQTT, que emula uma possível aplicação em ambientes industriais.

Na seção de número IV serão apresentados os resultados obtidos após a implementação das aplicações, bem como a explicação lógica de cada uma delas.

Finalmente na seção V é apresentada uma conclusão sobre o trabalho desenvolvido. Também é apresentado a satisfação quanto aos resultados obtidos com o projeto.

II – FUNDAMENTAÇÃO TEÓRICA

Nesta seção será apresentado o detalhamento de dois conceitos importantes para a compreensão do conteúdo presente neste artigo, o protocolo de comunicação *Message Queue Telemetry Transport* (MQTT) juntamente com o conceito de *Industrial Internet of Things* (IIoT).

A. O Protocolo *Message Queue Telemetry Transport*

O protocolo MQTT (*Message Queue Telemetry Transport*) é um protocolo de rede do tipo *publish/subscribe*, onde algum dispositivo publica suas mensagens em algum lugar e outro dispositivo subscreve, recebendo as mensagens relacionadas. Nesse tipo de protocolo não há comunicação direta entre o dispositivo que envia e o que recebe as mensagens [4], [6]. Todas as informações passam por um servidor que recebe as mensagens publicadas e aguarda até que um dispositivo requisite essa mensagem [2], [4].

Este protocolo é leve, simples e aberto, fazendo com que seja ideal para se usar para a comunicação M2M (*Machine to Machine*) e IoT (*Internet of Things*), onde as redes geralmente possuem uma alta latência, pequenas mensagens para serem entregues e onde a largura de banda disponível geralmente é mínima [4]. É um protocolo baseado em TCP/IP, possui 3 níveis de QoS (*Quality of Service*) e um mecanismo para alertar sobre desconexões anormais [4].

O MQTT foi criado com o objetivo de coletar dados de diversos dispositivos e transporta-los para uma base de dados. Por ser um protocolo extremamente leve, ele se torna ideal para monitoramento remoto, especialmente em conexões M2M. A Figura I mostra simplificada a estrutura de comunicação em MQTT [1].

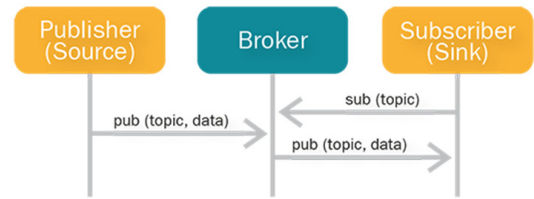


Fig. 1. Estrutura Simplificada de comunicação MQTT [3]

Como descrito anteriormente e mostrado na Figura 1, a estrutura de uma comunicação em MQTT baseia-se em duas instâncias, cliente e *broker*. O *broker* é um servidor virtual que armazena todas as informações advindas dos clientes, as separa em tópicos e as disponibiliza para que outros clientes busquem essas informações quando necessário. Já os clientes são aplicações, ou embarcadas em dispositivos ou em sistemas operacionais, que utilizam o *broker* como intermediário para receber ou enviar informações uns aos outros. Alguns podem utilizar tais informações para realizar processamentos e determinadas ações.

A implementação de uma rede industrial em MQTT traz uma série de benefícios:

- Maior eficiência na distribuição de informações;
- Reduz drasticamente o consumo de banda na rede;
- É um protocolo extremamente leve e de fácil implementação [5];
- Como é um protocolo do tipo *publish/subscribe*, ele coleta uma quantidade maior de informações utilizando menos banda de rede, quando comparado com outros protocolos do tipo *poll/response* [1], [2].

B. *Industrial Internet of Things*

O conceito de IIoT descreve a prática de aparelhos conectados uns aos outros através da internet em redes industriais e, apesar de ser um conceito não muito conhecido, já é realidade em diversos países.

A IIoT tem como objetivo conectar todos os hardwares e softwares presentes em uma rede de automação industrial através da internet, permitindo com que seja realizado uma automação inteligente da rede, onde todos os dispositivos podem compartilhar informações, notificar uns aos outros sobre todos os eventos que ocorrem ao seu redor e realizar uma otimização, minimizando desconexões e outros problemas recorrentes [7]. As aplicações utilizando IoT (*Internet of Things*) na indústria incluem controle de movimento, análise de diversos sistemas conectados, *Smart Grid*, manutenção preventiva e outras [8].

Este tipo de aplicação da IoT tem como objetivo otimizar essa rede, fazendo com que todos os dispositivos possam ser conectados entre si através de uma rede virtual, não havendo mais a necessidade de uma centralização das operações em um

único local [7]. Juntamente a isso, essa virtualização do ambiente permite com que todos os dispositivos possam ser auxiliados por aplicações externas que consigam prever um possível evento futuro ou realizar algum cálculo estatístico.

Graças a isso, indústrias que antes tinham todos seus processos, informações e trabalhadores centralizados em um único local, tem a possibilidade de utilizar dos conceitos de IoT para gerar uma série de facilidades e comodidades [7], [10]. Essa implementação na indústria permite uma descentralização de todos os processos, a comunicação em grandes distâncias entre máquinas, sensores e equipamentos de forma totalmente automática, e permite o monitoramento de um complexo ambiente industrial pela Internet [10].

Com esses temas devidamente explicados, a próxima seção irá descrever uma aplicação desenvolvida em MQTT pela equipe visando a aplicação e implementação dos conceitos aqui abordados.

III – DESENVOLVIMENTO DA REDE EM MQTT

Com o intuito de empregar o MQTT dentro do universo IIoT, foram criadas aplicações que emulem processos típicos em ambientes industriais. Utilizando dos conceitos abordados na seção anterior, foram implementadas algumas plantas industriais que emulem a situações, como mostra a Figura 2.

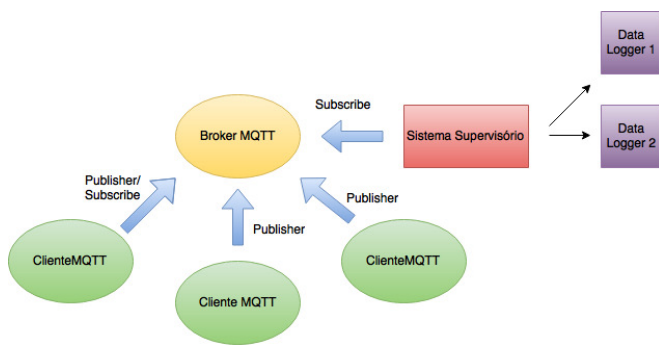


Fig. 2. Diagrama da planta implementada em MQTT

O diagrama simplificado do cenário criado em MQTT pode ser visto na Figura 2. Estão representados os principais elementos constituintes de uma rede MQTT, com a adição de algumas funcionalidades.

Nesta planta temos:

- Em verde aplicações MQTT do tipo *Publish* e *Publish/Subscribe* que emulam sensores e um atuador presentes em alguma das plantas e que tem função de gerar e/ou modificar os dados que serão trabalhados.
- Em amarelo temos o *broker* MQTT que utilizando dos conceitos vistos, conecta indiretamente todos os clientes

presentes na rede e também fornece um local para o fluxo de dados gerados.

- Em vermelho temos a emulação de um sistema supervisorio de dados que tem acesso a todos os dados presentes no *broker*. Ele pode tanto atuar diretamente em algum dos sensores presentes, como também realizar uma coleta de dados.
- Por último, em magenta, temos dois *Data Loggers* redundantes que copiam todas as informações presentes no servidor para arquivos texto de Log do sistema.

Após a definição desse cenário, deu-se início à implementação das aplicações que o emulam. Todas as aplicações foram implementadas através da linguagem de programação *Python*, juntamente com o módulo *paho-mqtt* desenvolvido para implementar MQTT utilizando essa linguagem.

A linguagem de programação *Python* foi escolhida para o desenvolvimento dessa aplicação por ser de rápida no desenvolvimento de aplicações complexas, sua facilidade de integração com a linguagem C e por possuir uma sintaxe relativamente limpa, o que permite a manutenção dos códigos-fonte com uma certa facilidade. Além disso, toda a equipe já tem experiência na programação utilizando essa linguagem, o que torna a implementação mais rápida e fácil.

Após isso, foi feito o levantamento de alguns *brokers* MQTT utilizados e disponíveis atualmente. A Tabela I nos mostra um breve comparativo entre eles, contemplando suas principais características.

Tabela I – Comparativo entre alguns brokers disponíveis

Server/broker	Principais características
IBM WebSphere MQ Telemetry	<ul style="list-style-type: none"> • É pago. • Possui ferramentas de gerenciamento simples. • Possui um processamento flexível de mensagens. • Utilizado com maior frequência para comunicações entre dispositivos móveis.
Mosquitto	<ul style="list-style-type: none"> • Open Source. • Permite utilização de um broker em nuvem ou a sua própria instalação. • Fácil Implementação.
Cloud MQTT	<ul style="list-style-type: none"> • É pago • O usuário tem acesso a um painel para controlar o <i>broker</i>. • Utiliza do broker <i>Mosquitto</i> para seu funcionamento

Após isso, foi decidido pela equipe a utilização do *broker Mosquitto* devido ao fato de ser simples, de fácil implementação, *Open Source*, além de disponibilizar uma biblioteca (módulo) totalmente funcional para Python e C++. Para utilizá-lo com aplicações que não necessitem de senhas, após a instalação basta executá-lo, uma vez que suas configurações padrão são comunicação via TCP, IP 127.0.0.1

e porta 1883. Todas essas configurações podem ser alteradas de acordo com a necessidade.

Em seguida foi criada uma pequena aplicação de teste que monitora todas as atividades presentes no *broker*, como tempo online, número de clientes conectados, quantidade de bytes recebidos e enviados, entre outros, como mostrado na Figura 3.

```
$$SYS/broker/publish/bytes/sent b'73'  
$$SYS/broker/bytes/received b'52'  
$$SYS/broker/bytes/sent b'756'  
$$SYS/broker/load/messages/recieved/1min b'1.83'  
$$SYS/broker/load/messages/sent/1min b'21.01'  
$$SYS/broker/load/messages/sent/5min b'4.52'  
$$SYS/broker/load/publish/sent/1min b'19.19'  
$$SYS/broker/load/publish/sent/5min b'4.12'  
$$SYS/broker/load/connections/1min b'0.91'
```

Fig. 3. Aplicação de teste do broker *Mosquitto*

Foi necessário o desenvolvimento desta aplicação visto que o *broker* escolhido serve simplesmente como um meio de conexão entre os dispositivos MQTT presentes na rede, não sendo possível o monitoramento direto através dele.

A Figura 3 representa alguns dos diversos tópicos presentes em um broker MQTT. É possível notar que cada informação presente no broker é designada por um caminho muito bem determinado, os tópicos. Estes tópicos são caminhos específicos até uma determinada mensagem que caracteriza várias informações sobre o local e dispositivo ao qual essa mensagem está alocada. São os clientes do tipo *Publisher* que são responsáveis pelo encaminhamento de mensagens para os tópicos, e sua classificação correta das mensagens entre os tópicos. Já os clientes tipo *Subscriber* utilizam desse caminho para monitorar as informações que cada um deseja obter.

A definição de maneira coerente dos tópicos em MQTT é de suma importância, pois enquanto em sistemas de comunicação industrial ponto-a-ponto as mensagens são enviadas para um endereço específico, em sistemas como MQTT as mensagens são enviadas diretamente para o *broker* e então distribuídas baseadas na escolha dos tópicos de cada cliente do tipo *Subscriber* [11].

Nos sistemas emulados foram utilizadas estruturas de tópicos que visam manter lógica e clara a forma de organização das informações presentes no *broker*. Essa estrutura se baseou na norma ISA 5.1 (*International Society of Automation*), a qual normatiza uma identificação funcional (TAG) para cada instrumento industrial em sistemas de medição e controle, sendo dada por um código de identificação que inclui um conjunto de letras e números. Abaixo é apresentada a estrutura utilizada nesse projeto que segue essa norma:

- */area/TAG (ISA 5.1)/point/operation/value*

Os caminhos acima descrevem um tópico geral existente no *broker* que possibilita a publicação ou subscrição das informações nele presentes. De maneira simplificada temos:

- */area/*: Indica de maneira escrita ou com um número de identificação a área em que o dispositivo de encontra.
- */TAG/*: Identificação funcional do dispositivo de controle e/ou medição de acordo com a norma ISA 5.1.
- */point/*: Sub identificação do controlador, uma vez que um mesmo aparelho que possui uma determinada TAG pode ter mais de um sensor ou aplicação conectado a ele.
- */operation/*: tipo de operação que será realizada, sendo elas *read* ou *write*.
/value/: Valor referente a todo o caminho descrito pelos clientes que publicam ou subscrevem o tópico. Sempre possui além de um valor a ser analisado, uma identificação que permite analisar o status atual do dispositivo fornecedor de informações.

O próximo passo foi construir as aplicações representadas em verde (Figura 2), do tipo *Publisher*, que emulam sensores presentes nas mais diversas partes de uma indústria. Eles possuem a função de mandar novas informações a respeito das variáveis de controle para o *broker* MQTT. Essas informações são coletadas em intervalos regulares e são alocadas em tópicos estruturados como mencionado acima.

Após isso, foi implementado a terceira aplicação em verde, do tipo *Publisher/Subscribe*, que emula um atuador presente em algum equipamento industrial, fazendo com que ele permaneça em determinado estado. Para isso, ele utiliza das informações provenientes dos sensores posicionados, e assim toma sua decisão através de rotinas previamente programadas.

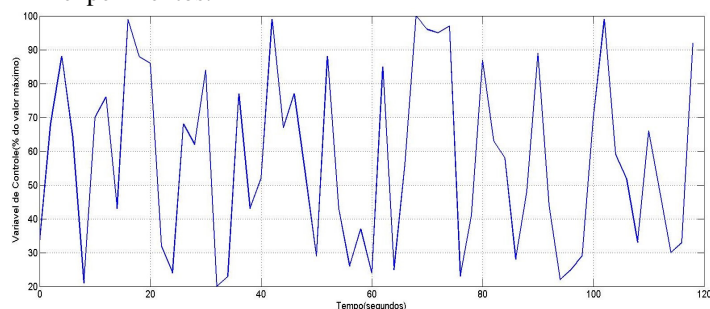
Com a finalização das aplicações em verde, deu-se início ao desenvolvimento de um sistema supervisorio (representado em vermelho), cujo objetivo é sobrescrever todos os tópicos disponíveis, realizar uma coleta em tempo real de todos os dados, e disponibiliza-os para *Dataloggers* (representados em magenta) que então enviam essas informações para arquivos de log do sistema.

Os *Dataloggers* são threads presentes no sistema supervisorio que analisam as informações presentes no próprio sistema e com isso geram arquivos separados sobre cada um dos tópicos mencionados anteriormente, juntamente com o horário de coleta dessas informações. Essas informações são então armazenadas em um arquivo texto a fim de servir como informação para algum programa ou operador humano que queira consultar alguns desses dados.

Essas informações foram tratadas através do software *MatLab* (ou similar) e, após isso, foi gerado um gráfico temporal com todas as informações colhidas pelos *Dataloggers* ocorridas nesse período.

IV – RESULTADOS OBTIDOS

Com o intuito de aplicar os conceitos relativos ao MQTT em ambientes industriais, foram realizados alguns testes nas aplicações descritas na seção anterior. Os resultados têm o objetivo unicamente de demonstrar o funcionamento da rede implementada, não sendo eles resultados obtidos com experimentos reais. Todas as situações que serão mostradas foram emuladas para uma maior facilidade de realização de experimentos.



O primeiro cenário emulado trata-se do recebimento de sinais provenientes de sensores que realizam a medição em intervalos constantes de uma grandeza de interesse. As medidas são provenientes de uma sub-rotina dentro da aplicação responsável por gerar esses dados. Nessa primeira situação são gerados valores aleatórios a uma taxa de 1 amostra por segundo. A Figura 4 nos mostra o comportamento dos valores gerados.

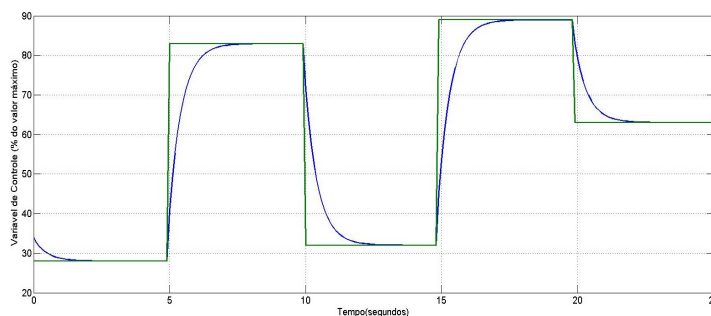
Fig. 4. Valores gerados aleatoriamente pelo sistema.

O gráfico representado pela Figura 4 nos mostra o registro dos *Dataloggers* MQTT a respeito dos dados gerados pelo primeiro teste realizado. Os dados aleatórios foram gerados sempre entre o intervalo entre 20% e 100% do valor máximo através de uma rotina que subscreve o tópico `/ground/PIC-104/point/value`. A partir disso, outra aplicação que subscreve o mesmo tópico (Sistema Supervisório) recebe essas novas informações e as encaminha para um *DataLogger* que guarda tais informações em um arquivo texto.

Com este teste foi possível executar e verificar a comunicação MQTT entre clientes e *broker*, assim como a avaliação a respeito do funcionamento correto da aplicação, bem como a verificação do sincronismo entre os threads presentes em cada uma das emulações.

A partir do teste descrito acima, foi proposto o desenvolvimento de um controle PID (Proporcional – Integral – Derivativo) como segundo teste a ser realizado. Este controlador foi implementado de maneira discreta, com uma taxa de amostragem de 10 amostras por segundo para que possa ser avaliado a resposta a um degrau de um sistema. A figura 5 nos mostra o comportamento da resposta do sistema através do tempo.

Fig. 5. Resposta do controlador PID ao degrau.



Como pode ser visto na Figura 5, os dados obtidos pelos *Dataloggers* a respeito do desafio proposto são condizentes com a resposta de um controlador PID com ganhos proporcional, integral e derivativo iguais, respectivamente a 0,2, 0,15, 0,03.

A cada 5 segundos, uma rotina presente no sistema supervisorio atua como um cliente MQTT e posta um novo *setpoint* para o sistema. Com isso, outra rotina presente em outra aplicação monitora esse valor através de uma subscrição ao tópico `/ground/PIC-104/point/setpoint` e aguarda até o momento em que uma nova mensagem chega no *broker*. No instante em que uma nova informação chega a esse tópico, uma terceira rotina presente no programa executa um controle PID da variável *setpoint* com relação a seu valor atual. A partir disso uma nova informação emulada dos sensores é gerada, permitindo o reinício do ciclo

V – CONCLUSÕES E RESULTADOS FUTUROS

Através desse projeto foi possível a implementação de uma comunicação básica em uma rede industrial utilizando o protocolo de comunicação MQTT. Nessa comunicação foram utilizadas aplicações escritas em linguagem Python que executam funções de clientes do tipo *Publish*, *Subscribe* e/ou *Publish/Subscribe*.

O desenvolvimento dessas aplicações faz parte de um projeto cujo objetivo é a integração do protocolo MQTT com uma outra rede que já esteja vinculada a um protocolo de comunicação diferente.

As aplicações desenvolvidas até o presente momento obtiveram resultados satisfatórios, uma vez que os resultados esperados por todas as aplicações criadas foram atingidos, no que se diz respeito à armazenagem permanente de informações, funcionamento do *broker*, funcionamento da comunicação via MQTT e fluxo de informações entre clientes MQTT através de um servidor.

O desenvolvimento desse projeto beneficia diretamente cursos da Área de Engenharia Elétrica que utilizam da área de Redes Industriais para a realização de trabalhos e pesquisas. Cursos como e de Engenharia de Controle e Automação, Engenharia de Computação e Engenharia de

Telecomunicações e outros podem utilizar desse projeto para realizar comunicação não somente em ambientes industriais, mas também em aplicações que envolvam comunicação de dispositivos controlados como CLPs (Controladores Lógicos Programáveis) em diversos ambientes.

É necessário ressaltar também um dos grandes desafios presentes nesse tipo de abordagem em ambientes industriais, a segurança das redes.

Quando comparadas com redes industriais tradicionais, esse novo tipo de rede industrial *IoT* ainda apresenta um déficit muito agudo quando se trata de segurança. Como redes *IoT* possuem um enorme fluxo de dados navegando através de dispositivos conectados na internet, há uma enorme exposição dessas informações para ameaças externas.

VI – AGRADECIMENTOS

Os autores gostariam de agradecer ao Laboratório de Automação Sistemas Eletrônicos e Controle (LASEC) da Faculdade de Engenharia Elétrica (FEELT) da Universidade Federal de Uberlândia (UFU) que, através do incentivo de pesquisa na área de Redes Industriais, permitiu a realização deste trabalho.

VII – REFERÊNCIAS

- [1] Inductive Automation (2016), *MQTT: The Leading Messaging Protocol for IIoT*. Acesso em 02 de abril de 2016, em: <https://inductiveautomation.com/what-is-mqtt>
- [2] M. Barros, “ *MQTT – Protocolos para IoT* ”, Acesso em: 02 de abril de 2016, em: <http://www.embarcados.com.br/mqtt-protocolos-para-iot/>
- [3] NOVOTEK (2016), *IoT Gateway with REST and MQTT interface*. Acesso em 03 de abril de 2016, em: <https://www.novotek.com/en/solutions/kepware-communication-platform/iot-gateway-with-rest-and-mqtt-interface>
- [4] OASIS (1998). *MQTT Version 3.1.1 Plus Errata 01*. Acesso em: 03 de abril de 2016, em: <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html>
- [5] R. Gupta, “ *5 Things to Know About MQTT – The Protocol for Internet of Things* ”, Acesso em: 18 de abril de 2016, em: https://www.ibm.com/developerworks/community/blogs/5things/entry/5_things_to_know_about_mqtt_the_protocol_for_internet_of_things?lang=en
- [6] M. Assis, “ *MQTT – O protocolo da IoT (Internet das Coisas)* ”, Acesso em: 01 de junho de 2016, em: <http://automatobr.blogspot.com.br/2014/08/mqtt-o-protocolo-da-iot-internet-das.html>
- [7] T. Paine, “ *Industrial Internet of Things and Communications at the Edge* ”, Kepware Technologies, Acesso em: 8 de abril de 2016, em: <https://www.accenture.com/us-en/labs-insightindustrial-internet-of-things.aspx>
- [8] M. Rouse, “ *Industrial Internet of Things (IIoT)* ”, Acesso em: 02 de junho de 2016, em: <http://internetofthingsagenda.techtarget.com/definition/Industrial-Internet-of-Things-IIoT>
- [9] World Economic Forum, Accenture. “ *Industrial Internet of Things: Unleashing the Potential of Products and Services* ”, Acesso em: 31 de maio de 2016, em: http://www3.weforum.org/docs/WEFUSA_IndustrialInternet_Report2015.pdf
- [10] C. B. Silveira, “ *O que é a Indústria 4.0 e como ela vai impactar o mundo* ”. Acesso em: 14 de abril de 2016, em: <http://www.citisystems.com.br/industria-4-0/>
- [11] V. Lampkin, W. T. Leong, L. Oliveira, S. Rawat, N. Subrahmanyam, *Building Smarter Planet Solutions with MQTT and WrbSphere MQ Telemetry*, IBM WebSphere, 1ª edição, IBM Corp, 2012.
- [12] M. Mielot, “ *Realize IIoT benefits with Industrial Wireless Technology* ”. Acesso em: 14 de abril de 2016, em: <http://www.belden.com/blog/industrialethernet/Realize-IIoT-Benefits-with-Industrial-Wireless-Technology.cfm>
- [13] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, “ *Internet of Things (IoT): A vision, architectural elements, and future directions* ”. Acesso em: 09 de junho de 2016, em: <http://www.sciencedirect.com/science/article/pii/S0167739X13000241>
- [14] J. Morgan, “ *A simple explanation of ‘Internet of Things’* ”. Acesso em: 09 de junho de 2016, em: <http://www.forbes.com/sites/jacobmorgan/2014/05/13/simple-explanation-internet-things-that-anyone-can-understand/>