

Universidade Federal de Uberlândia

Faculdade de Engenharia Elétrica

VITOR HUGO PRADO CARDOSO

PROPOSTA DE UM MEDIDOR DE CONSUMO DE ENERGIA UTILIZANDO TECNOLOGIA DE INTERNET DAS COISAS

Uberlândia

2015

VITOR HUGO PRADO CARDOSO

PROPOSTA DE UM MEDIDOR DE CONSUMO DE ENERGIA UTILIZANDO TECNOLOGIA DE INTERNET DAS COISAS

Trabalho apresentado como requisito parcial de avaliação na disciplina Trabalho de Conclusão de Curso 2 do Curso de Engenharia Elétrica da Universidade Federal de Uberlândia.

Orientador: Renato Ferreira Fernandes Junior

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Assinatura do Orientador

Uberlândia

2015

Dedico este trabalho aos meus pais, pelo estímulo, carinho e compreensão.

AGRADECIMENTOS

Agradeço ao Professor Renato Ferreira Fernandes Júnior pelo incentivo, motivação e orientação deste trabalho.

RESUMO

Este trabalho apresenta uma aplicação do conceito de Internet das Coisas (Internet of Things, IoT) no projeto de um dispositivo voltado para a medição de consumo de energia de aparelhos elétricos. A metodologia proposta baseia-se na aplicação de protocolos de comunicação de redes de computadores, eletrônica de potência, transmissão e processamento de dados. A IoT consiste em prover a comunicação e troca de informações entre diferentes dispositivos eletrônicos, seguindo padrões propostos pela OpenWSN, a fim de criar uma rede universal para a troca de informações entre os objetos que estão presentes no cotidiano das pessoas.

ABSTRACT

This paper presents an application of the concept of Internet of Things (Internet of Things, IoT) in the design of a facing device for measuring energy consumption of electrical appliances. The methodology proposed is based on the application of a protocol communication of a computer network, power electronics, transmission and data processing. The IoT consists on providing communication and exchange of information between different electronic devices, following standards proposed by OpenWSN in order to create a universal network for information exchanging between the objects that are present in people's daily lives.

LISTA DE ILUSTRAÇÕES

[Figura 1 - Hype Cycle for Emerging Technologies, 2015 14](#_Toc437033935)

[Figura 2 - Arquitetura OpenWSN 17](#_Toc437033936)

[Figura 3 - Rede OpenWSN 18](#_Toc437033937)

[Figura 4 - OpenVisualizer 19](#_Toc437033938)

[Figura 5 - OpenVisualizer 19](#_Toc437033939)

[Figura 6 - Roteamento OpenVisualizer 20](#_Toc437033940)

[Figura 7 - Comunicação CoAP via browser 21](#_Toc437033941)

[Figura 8 - Captura de pacotes com Wireshark 22](#_Toc437033942)

[Figura 9 - Arquitetura da rede de sensores 23](#_Toc437033943)

[Figura 10 - Cliente CoAP para rede de sensores OpenWSN 25](#_Toc437033944)

[Figura 11 - Representação do mote sensor 26](#_Toc437033945)

[Figura 12 - Microcontrolador CC2538EM 27](#_Toc437033946)

[Figura 13 - Placa de desenvolvimento CC2538EM 28](#_Toc437033947)

[Figura 14 - Rede de sensores OpenWSN 29](#_Toc437033948)

[Figura 15 - Localização das portas SPI na Beaglebone Black 30](#_Toc437033949)

[Figura 16 - Sonoma MAXREFDES14# 31](#_Toc437033950)

[Figura 17 - Diagrama de fluxo de dados da Sonoma MAXREFDES14# 32](#_Toc437033951)

[Figura 18 - Componentes da Sonoma MAXREFDES14# 32](#_Toc437033952)

[Figura 19 - Comunicação entre mote 802.15.4 e placa de aquisição 35](#_Toc437033953)

[Figura 20 - Comunicação entre mote 802.11 e placa de aquisição 36](#_Toc437033954)

[Figura 21 – Montagem para medição de valores 38](#_Toc437033955)

[Figura 22 - Ambiente de testes fim-a-fim 802.15.4 40](#_Toc437033956)

[Figura 23 - Ambiente de testes fim-a-fim 802.11 40](#_Toc437033957)

[Figura 24 - Obtenção do resultado dos testes utilizando PostgreSQL 41](#_Toc437033958)

[Figura 25 - Ambiente montado para medição de consumo 42](#_Toc437033959)

[Figura 26 - Curva Gaussiana do tempo de resposta do teste 1 via Ethernet 44](#_Toc437033960)

[Figura 27 - Curva Gaussiana do tempo de resposta do teste 2 via Ethernet 44](#_Toc437033961)

[Figura 28 - Curva Gaussiana do tempo de resposta do teste 3 via Ethernet 45](#_Toc437033962)

[Figura 29 - Curva Gaussiana do tempo de resposta do teste 4 via Ethernet 45](#_Toc437033963)

[Figura 30 - Curva Gaussiana do tempo de resposta do teste 5 via Ethernet 46](#_Toc437033964)

[Figura 31 - Comparação das curvas de tempo de resposta via Ethernet 46](#_Toc437033965)

[Figura 32 - Histograma dos testes de tempo de resposta via Ethernet 47](#_Toc437033966)

[Figura 33 - Boxplot dos testes de tempo de resposta via Ethernet 47](#_Toc437033967)

[Figura 34- Curva Gaussiana do tempo de resposta do teste 1 via CoAP 48](#_Toc437033968)

[Figura 35- Curva Gaussiana do tempo de resposta do teste 2 via CoAP 48](#_Toc437033969)

[Figura 36- Curva Gaussiana do tempo de resposta do teste 3 via CoAP 49](#_Toc437033970)

[Figura 37- Curva Gaussiana do tempo de resposta do teste 4 via CoAP 49](#_Toc437033971)

[Figura 38- Curva Gaussiana do tempo de resposta do teste 5 via CoAP 50](#_Toc437033972)

[Figura 39 - Comparação das curvas de tempo de resposta via CoAP 50](#_Toc437033973)

[Figura 40 - Histograma dos testes de tempo de resposta via CoAP 51](#_Toc437033974)

[Figura 41 - Boxplot dos testes de tempo de resposta via CoAP 51](#_Toc437033975)

[Figura 42 - Monitoramento de tensão 52](#_Toc437033976)

[Figura 43 - Monitoramento de corrente 53](#_Toc437033977)

[Figura 44 - Monitoramento de potência 53](#_Toc437033978)

[Figura 45 – Ambiente de testes OpenWSN com relé 58](#_Toc437033979)

[Figura 46 - Verificação dos valores medidos pela Sonoma 66](#_Toc437033980)

[Figura 47 - Verificação dos valores medidos pela Sonoma 66](#_Toc437033981)

LISTA DE TABELAS

[Tabela 1 - Protocolos propostos pela OpenWSN para uma rede IoT 16](#_Toc437033982)

[Tabela 2 - Localização dos pinos da porta SPI no CC2538EM e MAXREFDES14# 29](#_Toc437033983)

[Tabela 3 - Formato do comando reconhecido pela MAXREFDES14# 32](#_Toc437033984)

[Tabela 4 - Endereço dos registradores da Sonoma MAXREFDES14# 33](#_Toc437033985)

[Tabela 5 - Método de conversão para variável de leitura 33](#_Toc437033986)

[Tabela 6 - Ordenação dos bytes para enviar comando 34](#_Toc437033987)

[Tabela 7 - Resultado da verificação de medições 39](#_Toc437033988)

[Tabela 8 - Resultados dos testes de tempo de resposta via Ethernet 43](#_Toc437033989)

[Tabela 9 - Resultados dos testes de tempo de resposta via CoAP 43](#_Toc437033990)

[Tabela 10 - Parâmetros de medição da Sonoma MAXREFDES14# 59](#_Toc437033991)

LISTA DE ABREVIATURAS E SIGLAS

CoAP – Constrained Application Protocol

FEELT – Faculdade de Engenharia Elétrica

IEEE – Instituto de Engenheiros Eletricistas e Eletrônicos

IEC – Comissão Eletrotécnica Internacional

IETF – Internet Engineering Task Force

IoT – Internet of Things

ISA – Sociedade Internacional de Automação

OpenWSN – Open Wireless Sensor Networks

SPI – Serial Peripheral Interface

UFU – Universidade Federal de Uberlândia

WSN – Wireless Sensor Network

SUMÁRIO

1 – INTRODUÇÃO 12

2 – MEDIÇÃO DE CONSUMO INTELIGENTE 13

2.1 – Conceito 13

2.2 – Aspectos da IoT 14

2.3 – Medidor de consumo 15

2.4 – OpenWSN 16

2.5 – Cenário de IoT utilizando OpenWSN 17

3 – PROPOSTA DE UM MECANISMO DE MEDIÇÃO DE CONSUMO 23

3.1 – Arquitetura de Medição usando IoT 23

3.1.1– Aplicação remota 24

3.1.2 – Roteador de borda (Gateway) 25

3.1.3 – Rede de sensores sem fio 26

3.1.3.1 – Microcontrolador CC2538EM 26

3.1.3.2 – Beaglebone 30

3.1.3.3 – Placa de aquisição de dados 31

3.1.3.4 – Comunicação entre mote e a placa de aquisição 34

3.1.3.5 – Leitura de medições 36

4 – METODOLOGIA 38

4.1 – Aferição e validação das medições 38

4.2 –Taxa de Entrega fim-a-fim 39

4.3 – Leitura Remota dos Parâmetros de Medição 41

5 – RESULTADOS 43

5.1 – Análise 43

5.2 – DISCUSSÃO 54

6 – TRABALHOS PUBLICADOS 54

7 – CONCLUSÕES 55

8 – REFERÊNCIAS 56

8 – APÊNDICE 58

### 1 – INTRODUÇÃO

A Internet das Coisas ou IoT (*Internet of Things*) é composta por inúmeros equipamentos sensores e atuadores que monitoram e interagem com o ambiente. A proposta da IoT viabiliza aplicações em diferentes cenários como aplicações urbanas na área de saúde, transporte, monitoramento de trafego e urbano, agronegócios, automação residencial, predial, industrial, entre outras [1].

Uma rede IoT é formada basicamente por nós sensores que são limitados em termos de energia, memória e processamento computacional. Em aplicações em redes urbanas, as tecnologias de redes de IoT dependem do cenário e dos requerimentos da aplicação como confiabilidade, taxa de entrega, segurança, consumo. Como o meio físico geralmente é o ambiente sem fio, o acesso e o uso do recurso precisa ser controlado de forma a minimizar ao máximo as perdas [2].

A análise do consumo de energia é feita, na maioria das vezes, com a medição da corrente instantânea de dreno em um dispositivo físico [11][12]. Medidores de consumo são importantes tanto em ambientes industriais quanto residenciais. Nas indústrias, onde predomina o uso de motores elétricos, a medição de consumo ajuda a prever e identificar falhas, planejar manutenções preventivas, identificar queda de rendimento ou desgaste excessivo em equipamentos elétricos. Para ambientes residenciais, medidores de consumo estão tornando-se populares de acordo com a evolução de tecnologias de automação residencial, e o desejo das pessoas em controlar dispositivos eletrônicos visando mais conforto, segurança e economia de energia. Medidores de energia são formados basicamente por um elemento sensor de corrente, um elemento sensor de tensão, e um dispositivo transdutor responsável por converter dados elétricos em informação que pode ser processada computacionalmente ou interpretada pelos usuários.

Neste trabalho serão descritas formas para se realizar a medição de consumo de energia elétrica e diferentes maneiras para que esta informação possa se tornar acessível aos usuários através da Internet das Coisas. Este trabalho visa propor uma arquitetura de IoT para medição inteligente de consumo de energia elétrica, descrevendo os principais componentes de uma rede de IoT e as tecnologias envolvidas nas redes de sensores sem fio.

### 2 – MEDIÇÃO DE CONSUMO INTELIGENTE

### 2.1 – Conceito

A Internet das Coisas (Internet of Things ou também IoT) é um conceito computacional que descreve o lugar onde, no futuro, os elementos físicos do cotidiano estarão conectados à internet, para então identificar outros dispositivos e estabelecer uma comunicação. A IoT é importante porque um objeto que pode apresentar a si mesmo torna-se digitalmente algo maior que o objeto por si só.

A maneira tradicional para realizar medição de consumo de energia elétrica requer o deslocamento de técnicos até cada ponto de consumo para realizar cada medição. Este método implica em um custo elevado, difícil acesso à informação, e possibilidade de erro humano ao realizar cada medição. Com base neste contexto se destacam tecnologias remotas e automatizadas de medição de consumo que visam simplificar e tornar acessível o monitoramento constante destes dispositivos. A Internet das Coisas se mostra como uma potente ferramenta que pode contribuir para que a medição de consumo remota possa ser possível e eficiente.

Quando muitos objetos interagem entre si dentro de um mesmo contexto, eles formam um “ambiente inteligente” [1]. Em outras palavras, sensores inteligentes implantados dentro de uma residência vão permitir que o proprietário controle tudo o que acontece lá. Sensores implantados em uma cidade vão permitir que as comunidades sejam melhor gerenciadas.

Este trabalho tem como objetivo propor, dentro do conceito de IoT, um dispositivo sensor de consumo de energia, interconectado a um ambiente inteligente, analisando o comportamento elétrico de um aparelho qualquer e compartilhando as informações com a rede de Internet das Coisas.

### 2.2 – Aspectos da IoT

Temas relacionados ao consumo e eficiência energética são atualmente muito debatidos devido à dependência cada vez maior da sociedade de fontes energéticas, e é um assunto primordial para os governos do mundo toda a gestão da energia de forma eficiente. O monitoramento da energia, seja local, através da medição residencial ou predial, de forma remota, através de grandes centros, no monitoramento de bairros e cidades é atualmente um grande tema de estudo.

A Figura 1 foi retirada de um estudo feito pela Gartner, uma empresa de consultoria dos Estados Unidos, que acompanha as tendências e expectativas de novas tecnologias, e indica o a Internet das Coisas, em julho de 2015, se encontra no auge da expectativa do seu ciclo de vida, ou seja, é um assunto fortemente debatido no meio acadêmico e industrial, e em breve novos produtos devem ser desenvolvidos e comercializados se beneficiando desta tecnologia.

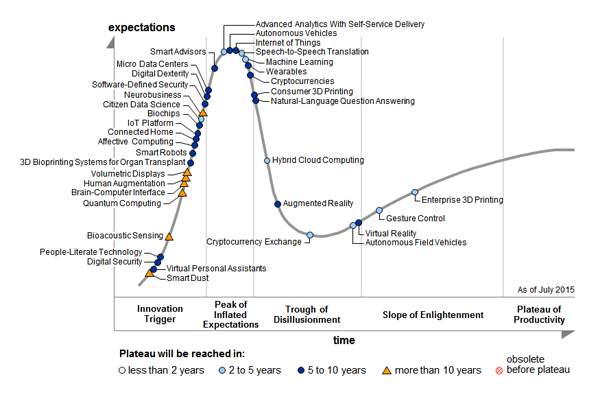


Figura 1 - Hype Cycle for Emerging Technologies, 2015

A Internet das Coisas é composta por inúmeros equipamentos sensores e atuadores que monitoram e interagem com o ambiente. A proposta da IoT viabiliza aplicações em diferentes cenários como aplicações urbanas na área de saúde, transporte, monitoramento de trafego e urbano, agronegócios, automação residencial, predial, industrial, entre outras [1].

Uma rede IoT é formada basicamente por nós sensores que são limitados em termos de energia, memória e processamento computacional. Em aplicações em redes urbanas, as tecnologias de redes de IoT dependem do cenário e dos requerimentos da aplicação como confiabilidade, taxa de entrega, segurança, consumo. Como o meio físico geralmente é o ambiente sem fio, o acesso e o uso do recurso precisa ser controlado de forma a minimizar ao máximo as perdas [2].

As tecnologias de redes de sensores sem fio estão baseadas no padrão IEEE802.15.4 e vários trabalhos têm sido propostos para atender os requisitos do padrão IoT. Para as camadas superiores, instituições como ISA (Sociedade Internacional de Automação) e o IEC (Comissão Eletrotécnica Internacional) e o IETF (Internet Engineering Task Force) trabalham em protocolos para atender aos diferentes cenários. Para as aplicações da IoT, o IETF tem conduzido os principais esforços na padronização da rede como o 6LoWPAN, RPL e CoAP. O grupo OpenWSN da universidade de Berkeley desenvolveu uma plataforma para prover um ambiente aberto de implementação seguindo padrões atuais para uma solução completa de IoT com o objetivo principal de mostrar a viabilidade de soluções de internet das coisas em diferentes cenários [3][4].

### 2.3 – Medidor de consumo

Atualmente é possível encontrar diversos dispositivos comerciais para medição de consumo de energia, mas embora muitos aparelhos tenham conexão wireless (no padrão IEEE 802.11 ou IEEE 802.15.4) e possam ser acessados remotamente, nenhum deles apresenta a possibilidade de interconexão com um ambiente de Internet das Coisas. Para atender aos requisitos de Internet das Coisas proposto pela ISA, IEE, IETF e IEC foi utilizado a proposta da OpenWSN aplicado ao medidor de consumo, visando assim obter um medidor de consumo de energia que implemente os requisitos necessários para atender aos padrões de IoT.

Uma segunda proposta de medição de consumo é através da utilização de uma rede Ethernet com comunicação via Socket, implementando uma topologia cliente servidor entre o mote e o roteador de borda. A comunicação Ethernet não será descrita neste trabalho por ser um padrão extensamente utilizado para comunicação entre dispositivos e de conhecimento geral.

A implementação de dois tipos diferentes de comunicação entre mote e roteador de borda tem como objetivo permitir a comparação entre sistemas diferentes.

### 2.4 – OpenWSN

OpenWSN (Open Wireless Sensor Network) é um projeto open source iniciado na Universidade de Berkeley (University of California, Berkeley) com objetivo de propor a implementação open-source de uma pilha de protocolos para desenvolvimento de tecnologias para Internet das Coisas utilizando software e hardware [17].

A pilha de protocolos proposta pelo projeto OpenWSN para a implementação de um ambiente de Internet das Coisas é especificada na Tabela 1.

Tabela 1 - Protocolos propostos pela OpenWSN para uma rede IoT

|  |  |
| --- | --- |
| **Camada** | **Protocolo** |
| Aplicação | HTTP/CoAP |
| Transporte | UDP/TCP |
| Rede | IPv6 / RPL |
| Adaptação | 6LowPAN (RFC4944) |
| Enlace (MAC) | IEEE802.15.4 |
| Física (PHY) | IEEE802.15.4 |

A Figura 2 descreve uma implementação típica utilizando os protocolos e ferramentas propostas pelo OpenWSN. Neste caso, a aplicação remota se comunica via internet com a rede de sensores sem fio através da conversão feita pelo roteador de borda. Os componentes apresentados na Figura 2 serão discutidos de forma mais detalhada nas sessões seguintes.



Figura 2 - Arquitetura OpenWSN

### 2.5 – Cenário de IoT utilizando OpenWSN

Nesta sessão será apresentado um cenário de Internet das Coisas utilizando os protocolos e ferramentas do projeto OpenWSN. Este capítulo tem como objetivo apresentar ao leitor um ambiente típico de IoT proposto pelo projeto OpenWSN.

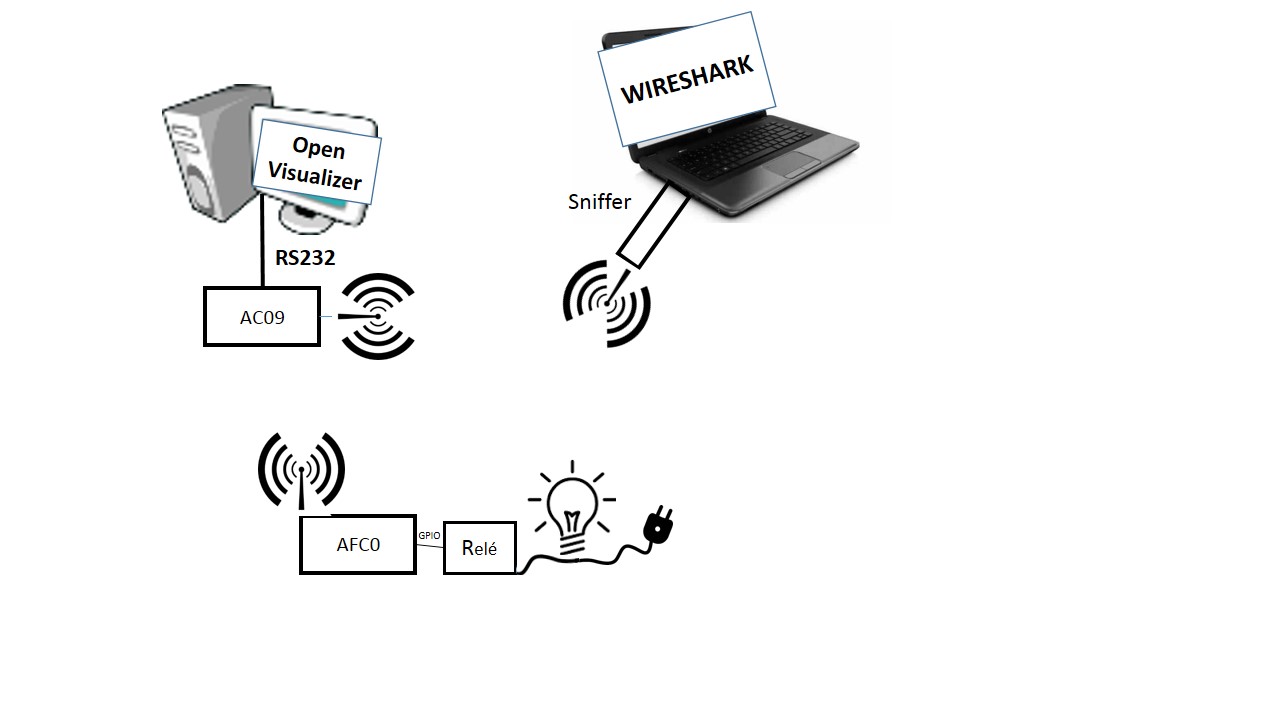


Figura 3 - Rede OpenWSN

O OpenVisualizer é a ferramenta que permite ao mote (nó com endereço MAC final AFC0) estabeleça conexão com o roteador de borda (nó com endereço MAC final AC09). O roteador de borda é um nó da rede que recebe o papel de DAG Root pois ele será o coordenador da rede ou o “sink”. O endereço completo MAC do DAG Root é 0012:4b00:02f4:ac09. Ele está conectado ao nó sensor AFC0 (endereço MAC 0012:4b00:02f4:afc0). O mote está ligado a um relé que aciona uma lâmpada. Para a análise da rede foi utilizado Wireshark como software de captura conectado a um sniffer da Freescale.

Neste sistema é possível acionar o relé e acionar a lâmpada do mote usando um browser no computador que está executando o OpenVisualizer. Isto pode ser facilmente estendido para qualquer outro equipamento web (smartphone, tablet, outro computador) desde que conectado na rede do OpenVisualizer.

As figuras 4, 5 e 6 mostram as telas do OpenVisualizer. O OpenVisualizer além de fazer o papel de roteador de borda (conversão das informações da web para a rede IEE 802.15.4) também disponibiliza um serie de diagnósticos de rede como detalhes sobre os nós vizinhos, quantidade de pacotes enviados e recebidos, topologia da rede, dentre outras informações.

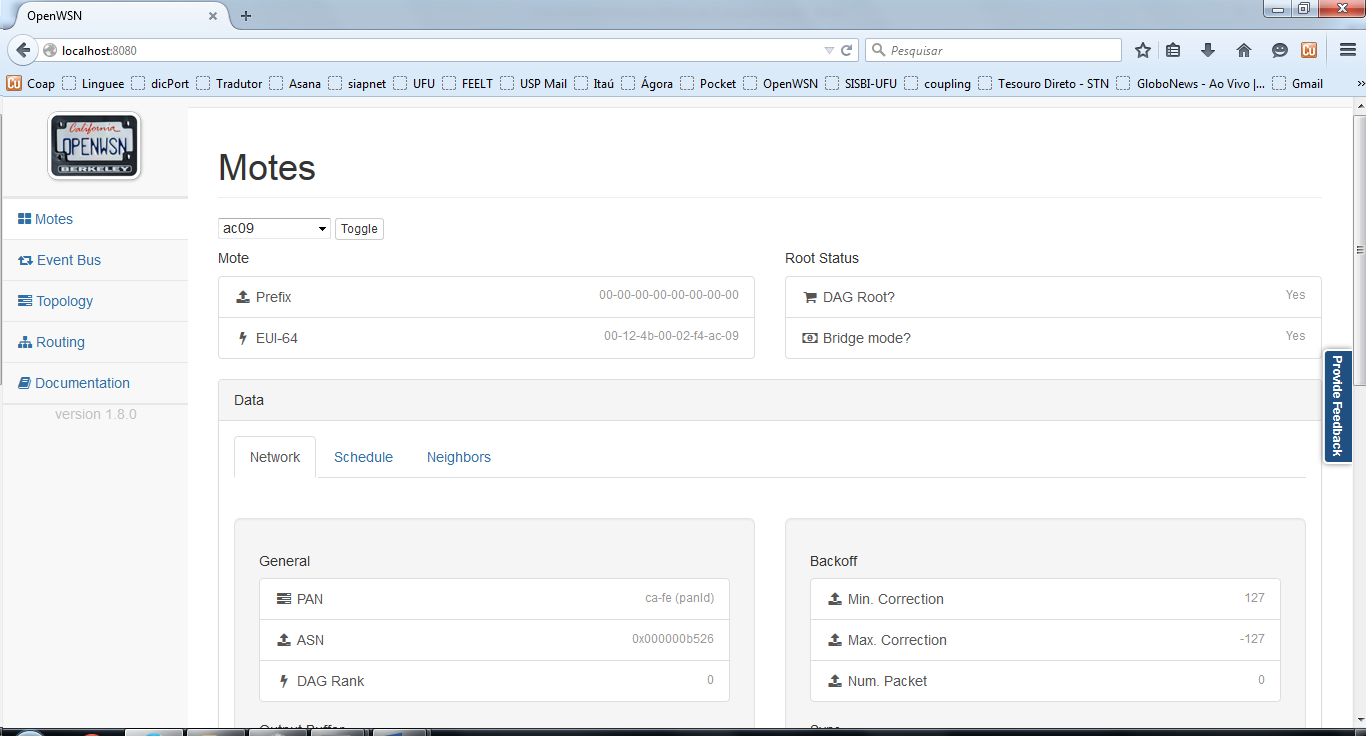


Figura 4 - OpenVisualizer

A Figura 4 mostra a tela principal do OpenVisualizer em um browser web. Nesta tela é possível que o usuário escolha o nó, identificado pelos 2 últimos bytes do endereço MAC, para atuar como DAGROOT. Por exemplo na Figura 4 é selecionado como DAGROOT o AC09 e na Figura 5 é mostrada a aba com a tabela de vizinhos do nó AC09. Neste caso o único nó vizinho é o AFC0.

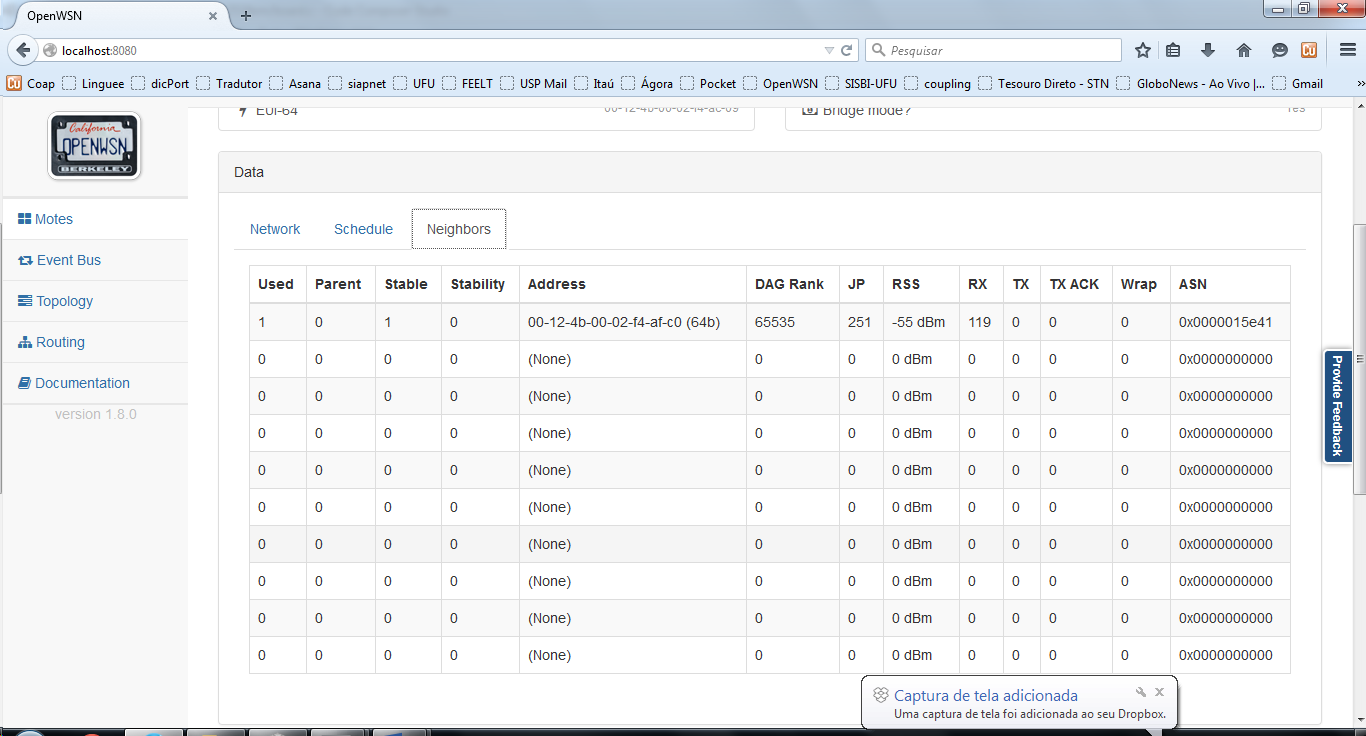


Figura 5 - OpenVisualizer

Na Figura 6 é mostrado a tabela de roteamento montada para a rede. Neste caso são mostrados somente os dois nós AC09 (DAGROOT) e o nó AFC0. Em uma rede mais complexa a seta vai indicar a quem cada nó está ligado na rede de topologia mesh. A seta indica o melhor caminho escolhido pelo nó na rede.

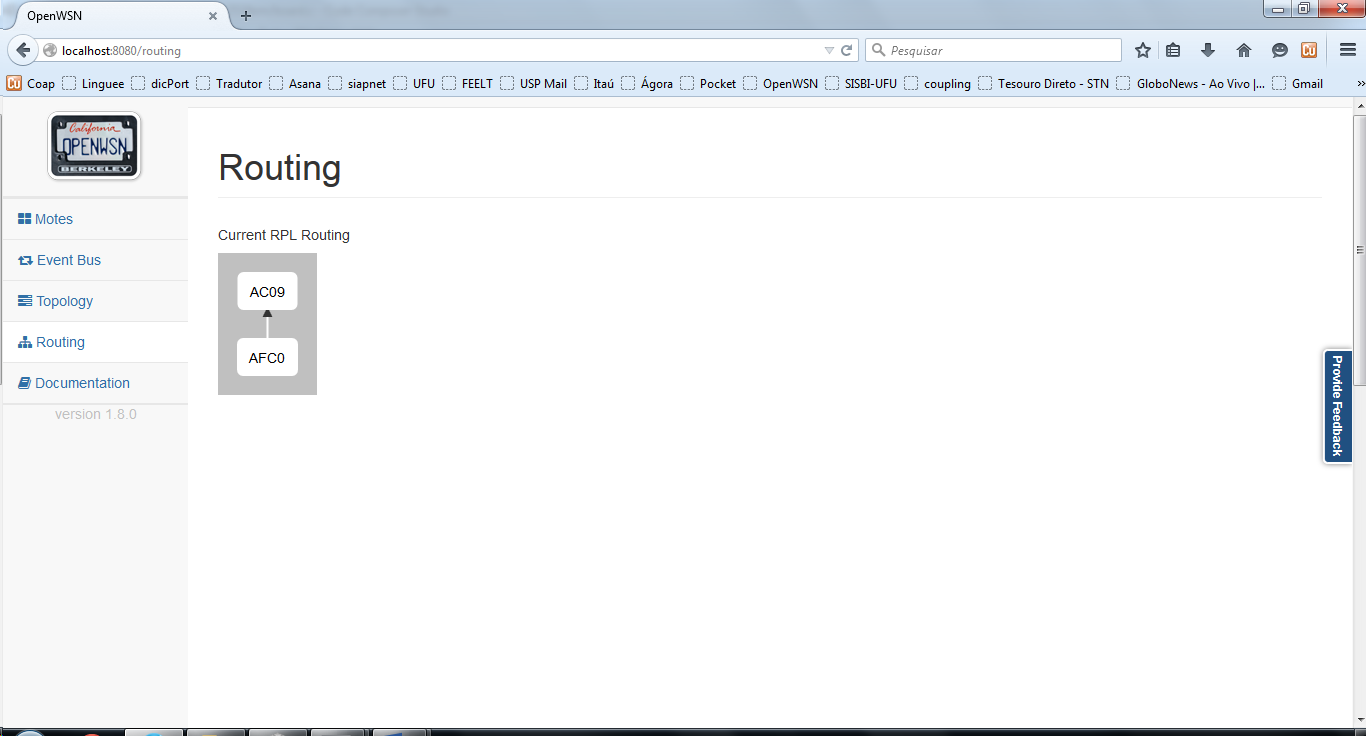


Figura 6 - Roteamento OpenVisualizer

Estas informações fornecidas pelo OpenVisualizer são informações internas do protocolo que cobre das camadas inferiores até camada de rede.

As camadas superiores UDP e CoAP são voltadas para a aplicação. O UDP pode ser testado por exemplo através de um comando ping no prompt de comando do Windows, que retornaria o status da conectividade entre os dispositivos. O CoAP retorna dados da aplicação, no nosso exemplo é o responsável por buscar as informações dos sensores e atuadores.

O browser Mozilla Firefox possui um plugin (Copper) que permite enviar os comandos GET, POST, PUT e Delete via CoAP. A Figura 7 mostra o plugin em funcionamento.

No browser foi enviado o comando GET:

* coap://[bbbb::12:4b00:3a6:4cbe]:5683/d

Entre colchetes está especificado o IP do mote, utilizando a porta 5683, e o parâmetro “/d” que representa uma solicitação das características do nó. A seguinte resposta é recebida:

* {"v":1,"m":"CC2538EMTI","id":1,"npts":1}

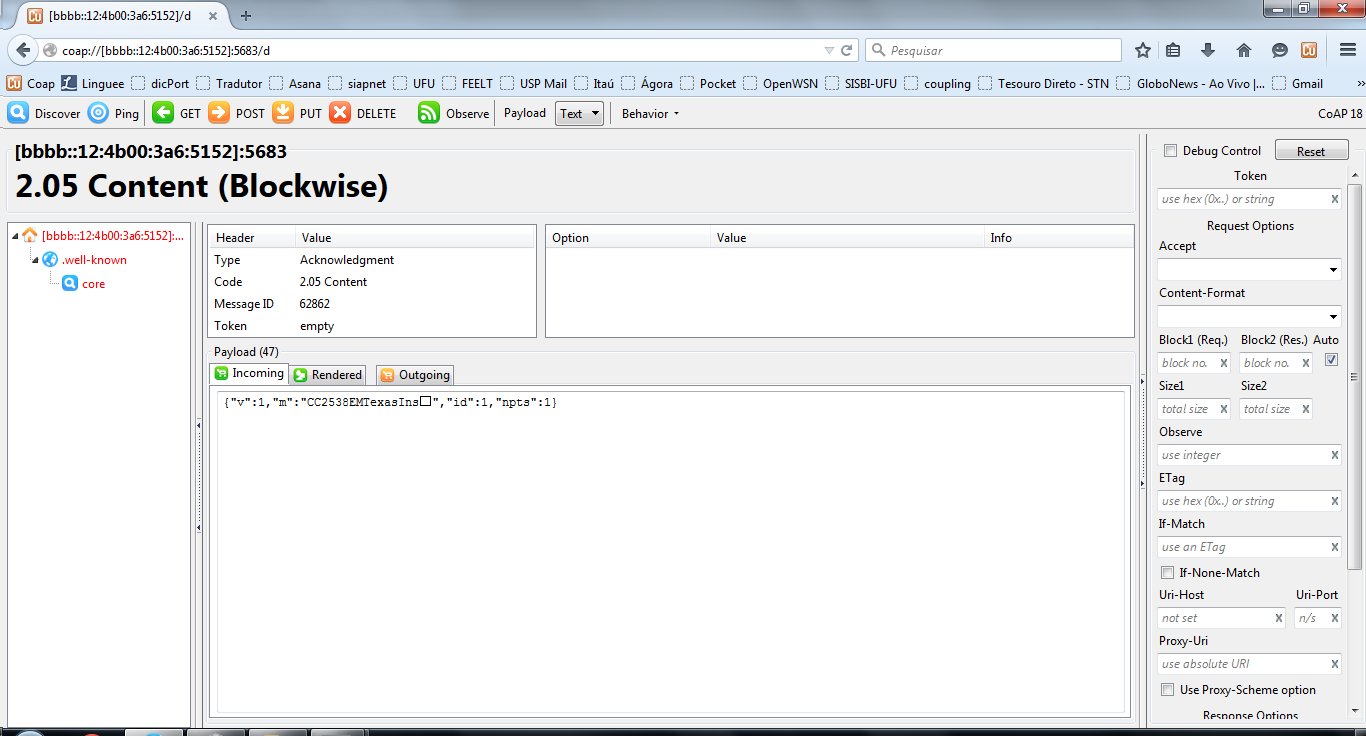


Figura 7 - Comunicação CoAP via browser

Na resposta o mote indica a descrição do nó “CC2538EMTI”, o identificador “1” e que ele possui apenas 1 parâmetro ("npts":1). Ainda na aplicação, um comando para ligar a lâmpada foi enviado com o seguinte formato:

* coap://[bbbb::12:4b00:3a6:4cbe]:5683/s/1/1

Neste caso foi enviado um PUT (comando de escrita). Entre colchetes está o IP do mote, o 5683 é a porta de conexão, “s/1” indica a comunicação com o parâmetro número 1, e o próximo “/1” indica que deseja ligar a lâmpada. Para desligar a lâmpada deve ser enviado o parâmetro “/s/1/0”.

Para saber o estado atual da lâmpada basta enviar um comando GET para o mesmo endereço do parâmetro ou seja:

* coap://[bbbb::12:4b00:3a6:4cbe]:5683/s/1

E a resposta poderia ser:

* {"v":1} - indicando que a lâmpada está ligada, ou
* {"v":0} - indicando que a lâmpada está desligada

A Figura 8 mostra a captura das comunicações feita com o Wireshark. É possível ver os comandos de requisição e de resposta do CoAP tanto para um PUT (frames número 201 e 204) quanto para um GET (frames número 211, 212, 218 e 219).

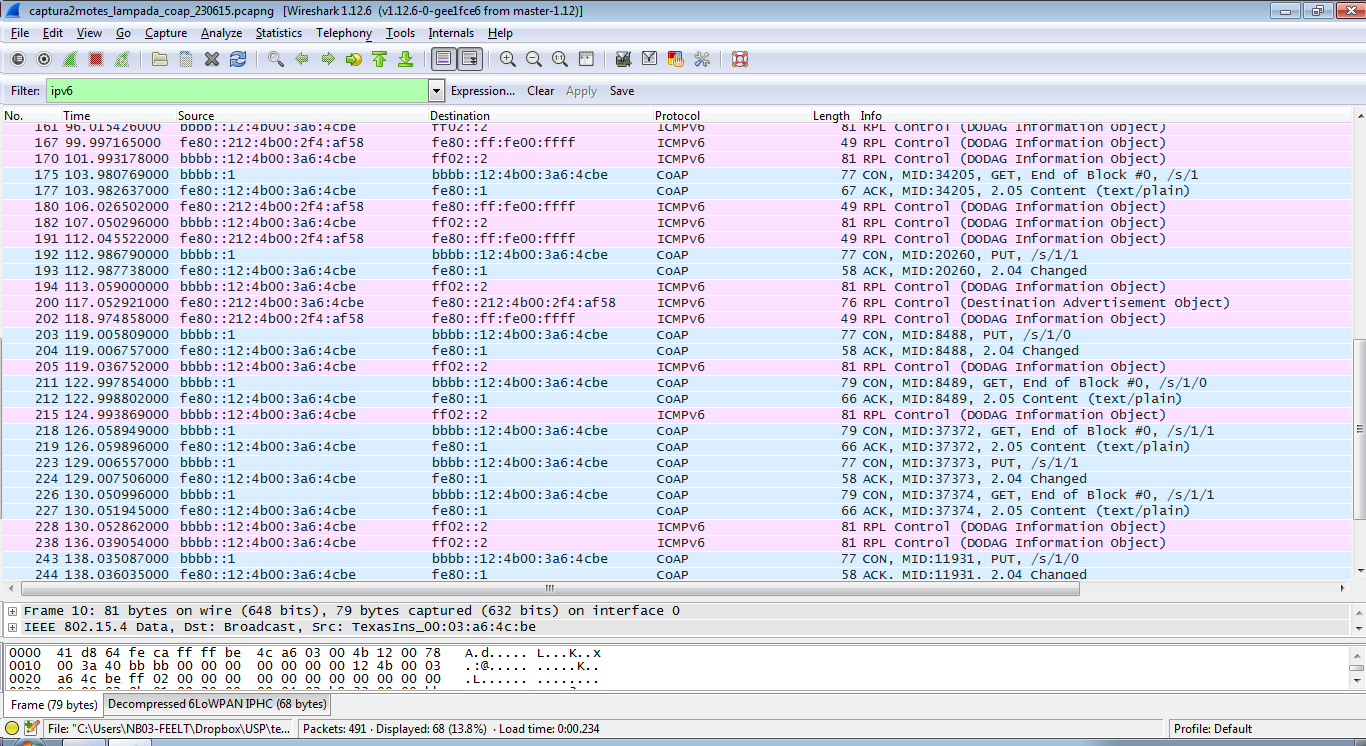


Figura 8 - Captura de pacotes com Wireshark

A Figura 45, no apêndice, mostra o experimento que foi montado para testar o OpenVisualizer, para então executar os testes demonstrados neste capítulo

### 3 – PROPOSTA DE UM MECANISMO DE MEDIÇÃO DE CONSUMO

### 3.1 – Arquitetura de Medição usando IoT

O analisador de consumo proposto tem como objetivo medir a quantidade de energia elétrica consumida por qualquer aparelho elétrico, coletando informações como tensão, corrente e potência consumida, e posteriormente, seguindo as normas e padronizações da IoT citadas no capítulo 2, prover o compartilhamento destas informações com os demais dispositivos da rede.

A proposta de medição deste trabalho propõe a análise de duas tecnologias baseadas em redes wifi (802.11) e redes de sensores sem fio (802.15.4). A arquitetura do medidor proposto pode ser representada pela Figura 9. Ela pode ser dividida em três partes principais: aplicação remota, roteador de borda (gateway) e rede de sensores sem fio.

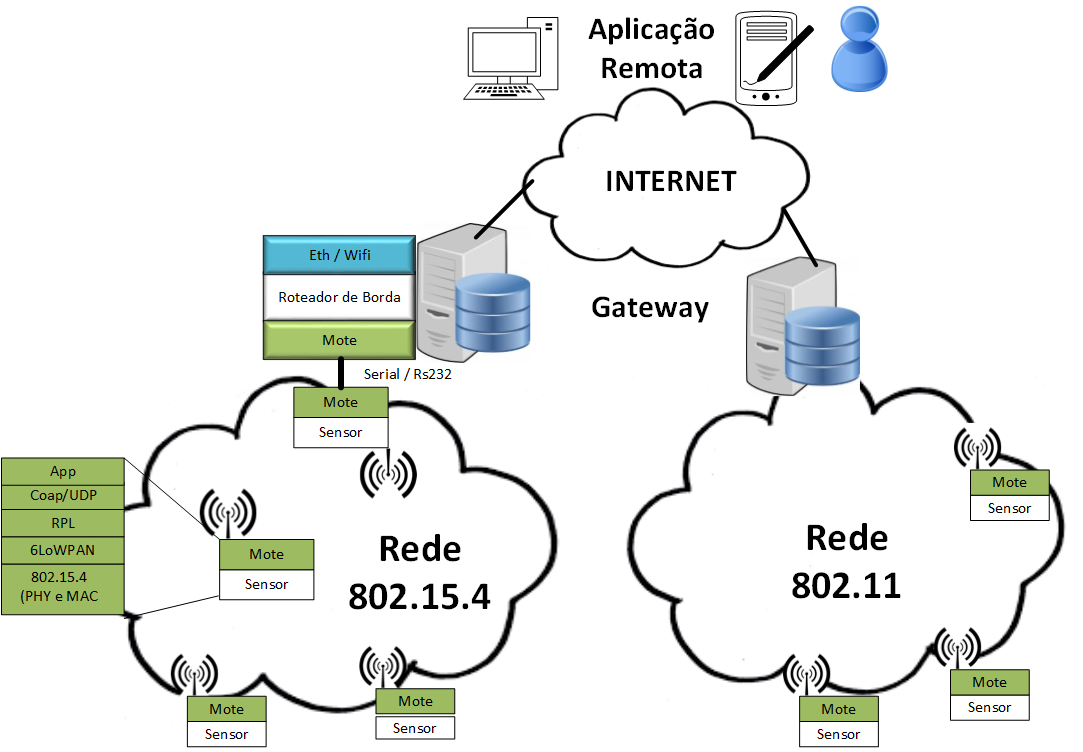


Figura 9 - Arquitetura da rede de sensores

### 3.1.1– Aplicação remota

O acesso remoto à informação pode ser feito, a princípio, por qualquer dispositivo conectado à internet, seja ele um computador, um tablet, um smartphone, ou qualquer outro dispositivo que tenha conexão com a internet. As aplicações podem ter diversas finalidades, podem ser desenvolvidas para controlar os dispositivos da rede de sensores e atuadores individualmente, controlar um sistema composto por diversos nós, ou apenas ler e armazenar informações.

Para a rede 802.15.4 a aplicação rodava o programa “Sensor Data Scanner”, um cliente web do ambiente OpenWSN demonstrado na Figura 10. O software fazia uma requisição CoAP para que chegaria ao nó sensor, que processaria o pedido e responderia a requisição pedida. Uma requisição de leitura de tensão teria o seguinte formato:

* Requisição GET: coap://[bbbb::12:4b00:040e:fc87]/s/0

Entre colchetes está especificado o IP do mote, e o parâmetro “/s/0” representa uma solicitação do parâmetro “tensão”, “/s/1” o parâmetro “corrente” e “/s/2” a potência ativa. A resposta, conforme a Figura 10 tem o formato:

* Resposta: {u’v’: 0.3}

Indicando que o valor medido é 0.3 (valor medido pelo sensor, sem conversão).

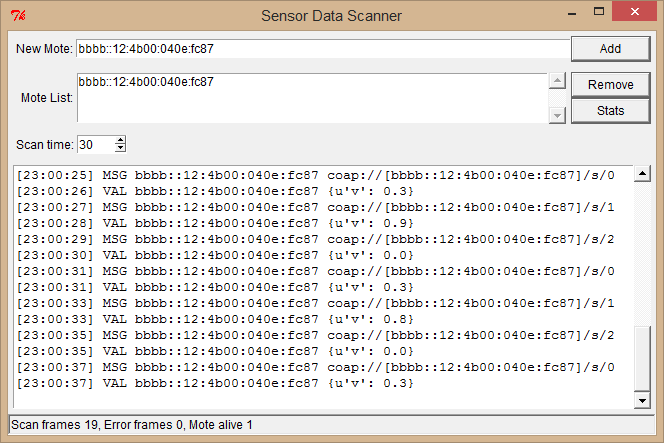


Figura 10 - Cliente CoAP para rede de sensores OpenWSN

Para a rede 802.11 foi utilizado um programa em Python desenvolvido especificamente para esta aplicação. O programa é constituído por dois scripts, cliente e servidor, sendo o primeiro executado no computador que executa as requisições, e o segundo sendo executado no microcontrolador. O software servidor abre um socket para conexão ethernet e aguarda até que o cliente se conecte a este socket e envie uma mensagem de requisição. Quando o cliente envia uma requisição o servidor envia como resposta a última leitura de todos os parâmetros lidos pelo sensor.

### 3.1.2 – Roteador de borda (Gateway)

O roteador de borda é o responsável por converter os protocolos de comunicação da internet para a rede de sensores sem fio e vice-versa.

Na rede 802.15.4 o roteador de borda é composto por um computador conectado a um mote (CC2538EM ou Beaglebone Black) com função de DAGROOT. O DAGROOT é o responsável por estabelecer a comunicação entre a rede de sensores sem fio via IEEE802.15.4.

Na rede 802.11 o roteador de borda não é necessário, pois a comunicação entre o gateway e o mote é feita via ethernet, ou seja, ambos utilizam o mesmo protocolo de comunicação.

### 3.1.3 – Rede de sensores sem fio

Na rede de sensores sem fio 802.15.4, a arquitetura do nó de medição (mote) é composta pelo módulo de aquisição de dados (Sonoma MAXREFDES14#) conectado ao microcontrolador (CC2538EM da Texas Instruments), que é o responsável por estabelecer a comunicação do mote com o roteador de borda via rede IEEE802.15.4.

Na rede de sensores usando o padrão 802.11, o mote implementado com a Beaglebone se comunica diretamente com o gateway da rede via IEE802.11. A comunicação entre o mote e o sensor (Sonoma MAXREFDES14#), também é feita via SPI. Os motes sensores são representados na Figura 11.

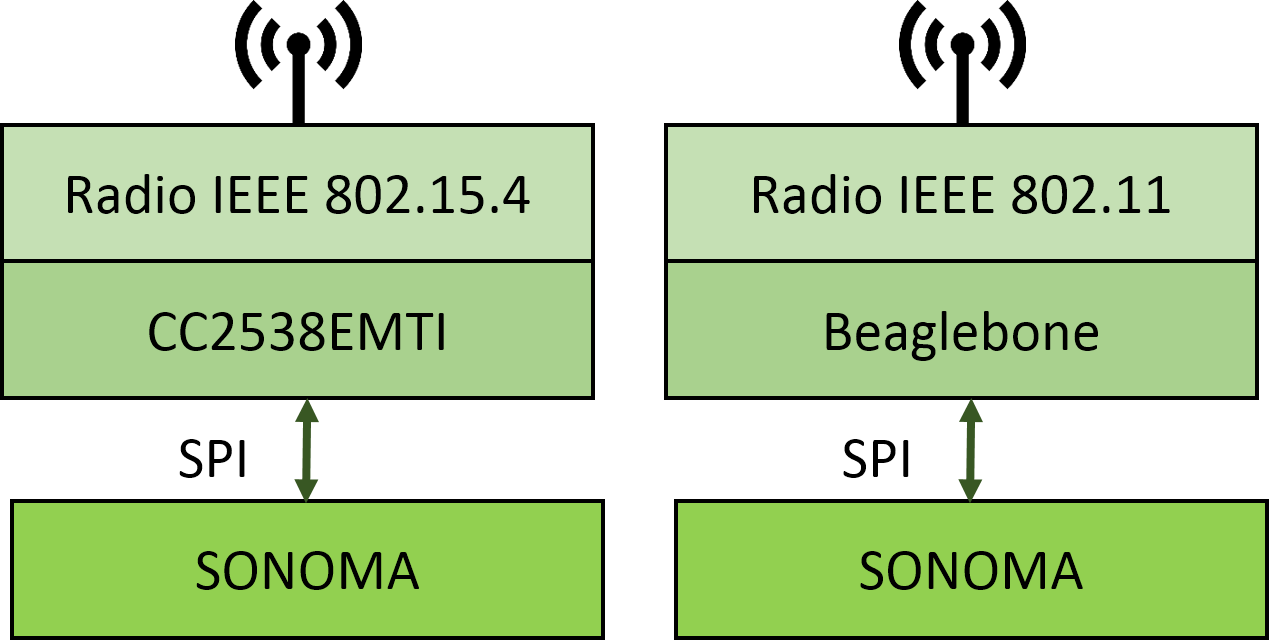


Figura 11 - Representação do mote sensor

### 3.1.3.1 – Microcontrolador CC2538EM

Para a rede 802.15.4 foi utilizado o microcontrolador CC2538 da Texas Instruments pois ele já era de domínio do grupo e também ele já era compatível com o ambiente de desenvolvimento do OpenWSN.

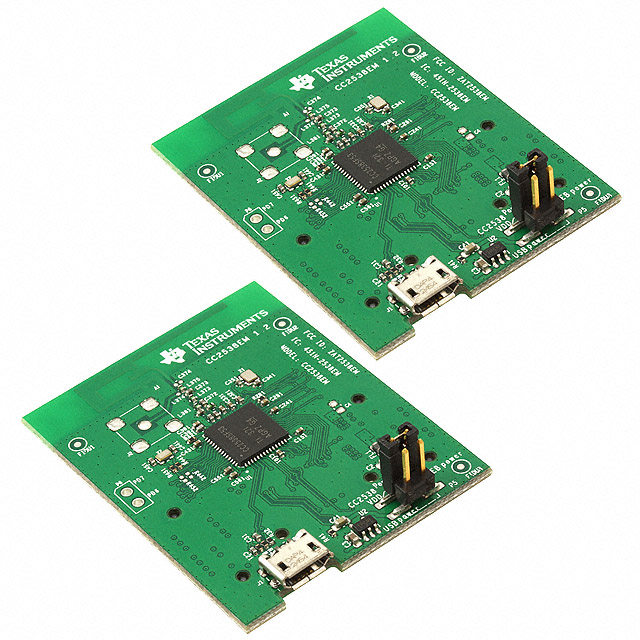


Figura 12 - Microcontrolador CC2538EM

O microcontrolador CC2538EM é baseado em um microprocessador ARM Cortex-M3 de 32 bits com clock de 32Mhz com 512 Kbytes de Flash, 32 Kbytes de RAM e com transciever de rádio 2.4Ghz IEEE802.15.4 interno com taxa de comunicação de 250 kbps do protocolo. O rádio tem uma sensibilidade de recepção de -97 dBm e transmissão programável até 7 dBM de potência. Isto possibilita aplicações de até 30 metros em ambiente interno (indoor) e de mais de 100 metros em ambiente externo (outdoor). A Figura 13 mostra a placa CC2538EM (acima) junto com a placa de desenvolvimento e teste “SmartRF06 Evaluation board”.

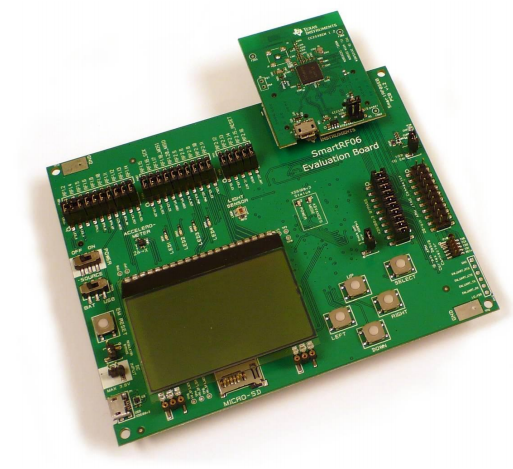


Figura 13 - Placa de desenvolvimento CC2538EM

O microcontrolador CC2538 (figura 12) possui como periféricos de entrada e saída ADC de 12 bits, pinos de GPIOs digitais, 4 timers além das interfaces de comunicação: 2 SPI, 2 UART, 1 I2C que possibilita ligar a placa de rádio a qualquer outra placa sensor.

Os testes foram realizados utilizando o cenário mostrado na figura 13. O ambiente é constituído por um computador executando o OpenVisualizer (principal ferramenta fornecida pelo OpenWSN, usada para gerenciar conexões) que é responsável por obter informações da rede e também servir como roteador de borda da rede. Um nó sensor foi ligado ao computador, fazendo o papel de DAGROOT [13], e os outros nós sensores estavam espalhados na rede. No total foram utilizados 8 nós sensores CC2538EM. Por fim um segundo computador foi utilizado para fazer a captura dos pacotes de dados que transitavam entre os dispositivos, utilizando o software Wireshark através de um sniffer dongle da Freescale.

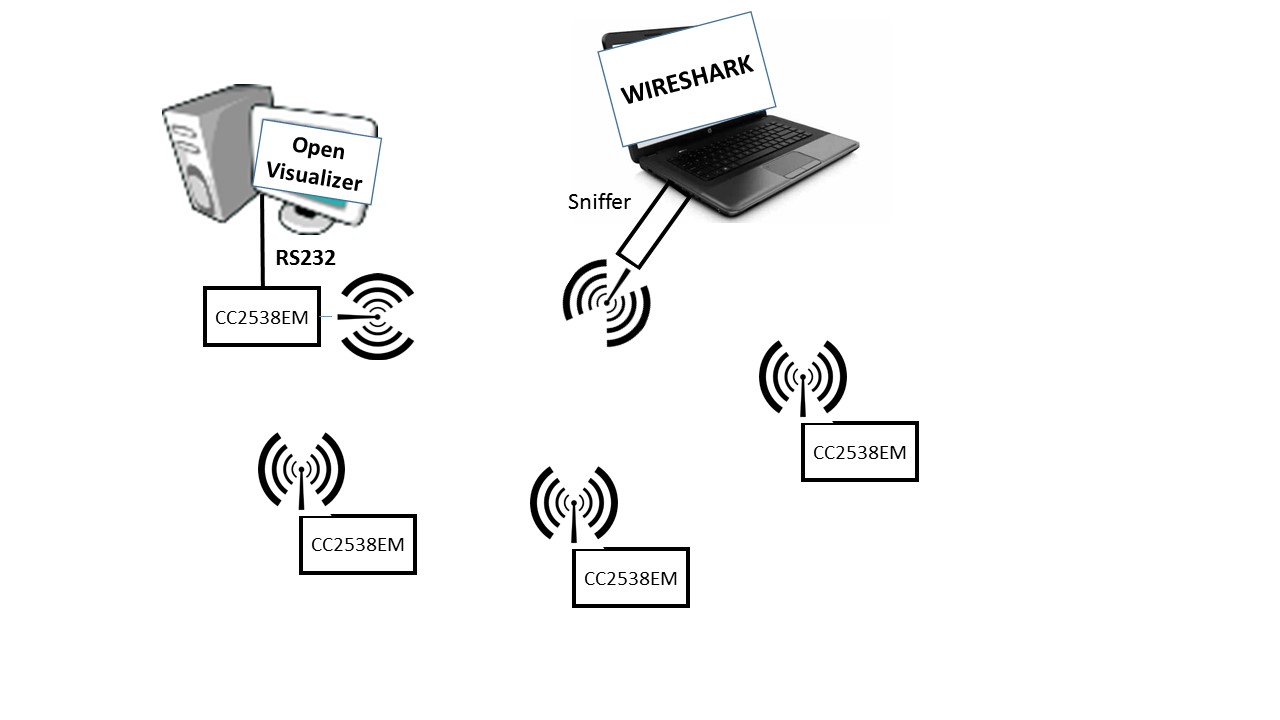


Figura 14 - Rede de sensores OpenWSN

A forma de comunicação habilitada por padrão na Sonoma MAXREFDES14# é via SPI. Os pinos utilizados para estabelecer a comunicação entre a placa de medição e o microcontrolador são descritos na tabela a seguir.

Tabela 2 - Localização dos pinos da porta SPI no CC2538EM e MAXREFDES14#

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CC2538EM** | | | **MAXREFDES14#** | |
| Pin | P1 | RF1 | Pin | Function |
| PA2 | P1.16 | RF1.16 | 4 | SPI\_SCK |
| PA3 | P1.14 | RF1.14 | 1 | SPI\_FSS |
| PA0 | P1.7 | RF1.7 | 2 | SPI\_MOSI |
| PA1 | P1.9 | RF1.9 | 3 | SPI\_MISO |
|  | P1.1 | RF1.1 | 5 | GND |
|  |  |  | 6 | 3V3 |

Na Tabela 2 os pinos da coluna P1 mostra a identificação do conector P1 da placa CC2538EM. A coluna RF1 indica os pinos da placa CC2538EM na placa de teste SMARTRF06, que em ambiente de desenvolvimento são mais acessíveis do que os conectores P1.

### 3.1.3.2 – Beaglebone

Para a rede 802.11 foi utilizado a placa Beaglebone Black. A Beaglebone é uma placa de desenvolvimento que possui um processador AM3358BZCZ100 da Texas Instruments. Tem 512 Mb de memória, 4Gb de armazenamento e conta ainda com um acelerador gráfico 3d. Sua proposta é ser um computador de baixo custo, que roda um sistema operacional Linux (o modelo utilizado neste projeto roda uma versão do Debian). A Beaglebone possui duas portas SPI, conforme demonstrado na Figura 15. Pela SPI 1 foi possível estabelecer a comunicação entre o microcontrolador e o módulo sensor e obter as leituras dos parâmetros de medição de energia.

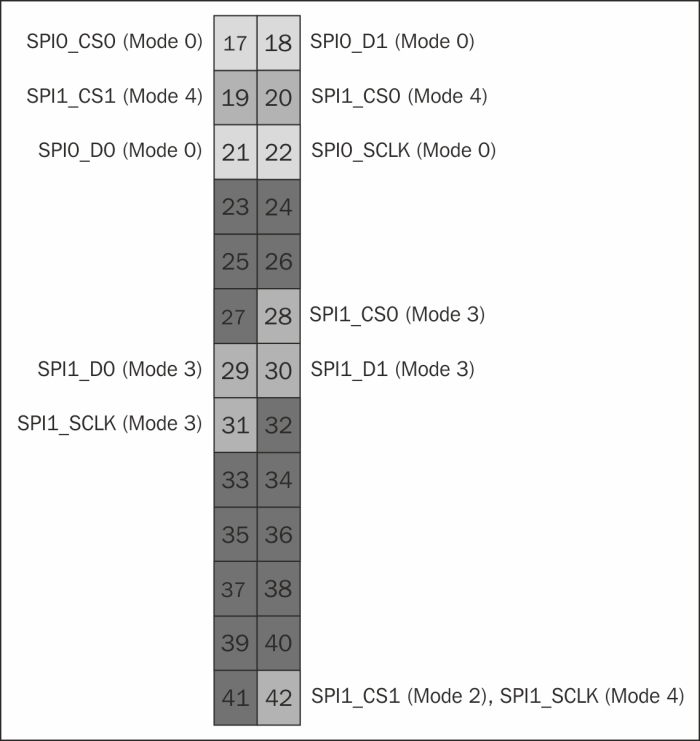


Figura 15 - Localização das portas SPI na Beaglebone Black

### 3.1.3.3 – Placa de aquisição de dados

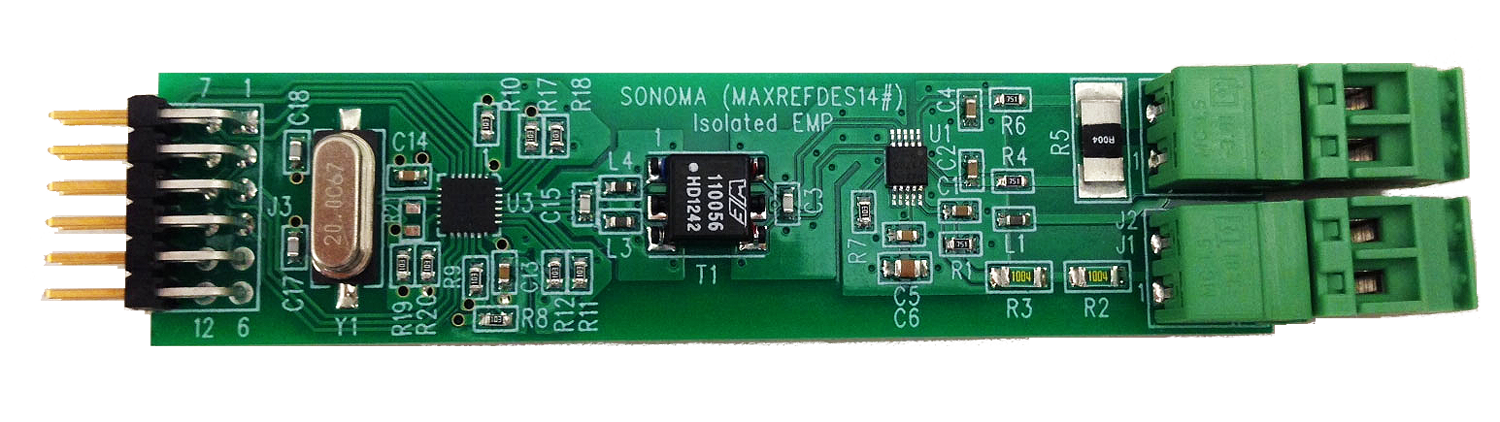


Figura 16 - Sonoma MAXREFDES14#

O hardware escolhido para ser utilizado na medição de consumo foi a placa de desenvolvimento Sonoma MAXREFDES14# (Figura 16). Este hardware foi o que se mostrou mais completo e acessível dentre as opções disponíveis no mercado, possuindo, dentre outras características, isolação entre alta potência e sistema de controle, comunicação SPI e UART, alta precisão e diversos outros recursos que serão apresentados a seguir [15].

Características:

* Medição de potência de alta precisão;
* Isolação galvânica de alta tensão;
* Ganho de Preset / parâmetros de offset;
* Resistor on-board de 4mΩ para detecção de corrente;
* Resistor on-board de divisão de tensão com razão de 2667: 1;
* Faixa de tensão de entrada AC Universal de 90 a 264VAC;
* Terminais plugáveis para AC (8A max);
* Placa de circuito impresso (PCB) de tamanho reduzido;
* Drivers de dispositivo;
* Exemplos de código-fonte em C;

Vantagens Competitivas

* Armazenamento não volátil de parâmetros de calibração e configuração;
* Isolação galvânica completa com um único transformador;

A Figura 17 e a Figura 18 descrevem o projeto de hardware da placa MAXREFDES14#.

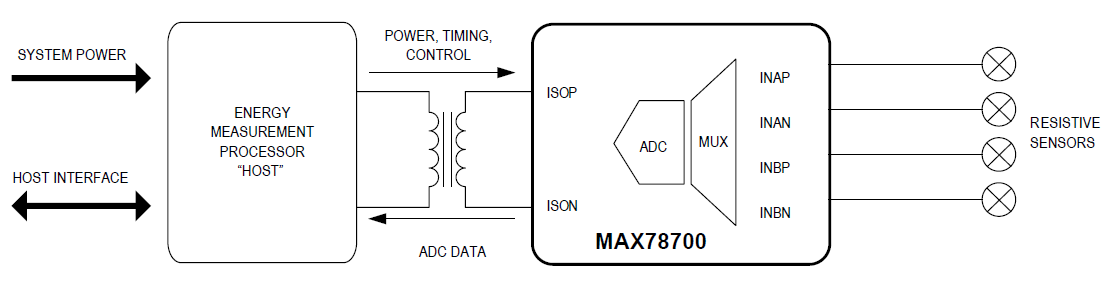


Figura 17 - Diagrama de fluxo de dados da Sonoma MAXREFDES14#

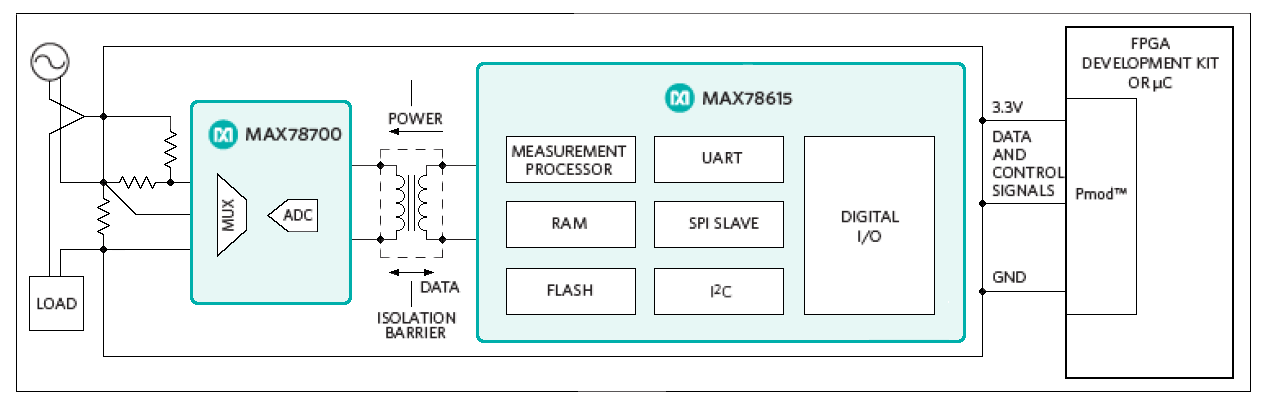
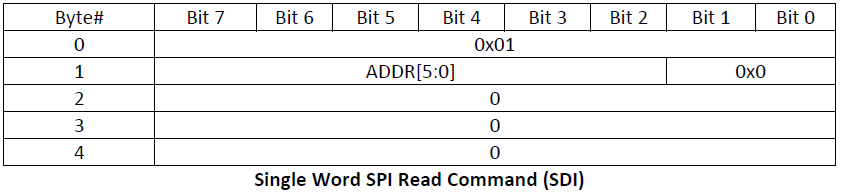


Figura 18 - Componentes da Sonoma MAXREFDES14#

A Sonoma MAXREFDES14# é configurada por padrão para se comunicar via SPI, e utiliza um protocolo próprio para se comunicar com outros dispositivos. Para haver comunicação com a placa é necessário seguir as instruções relativas à placa especificadas no manual fornecido pelo fabricante [15].

A mensagem de comando enviada pelo microcontrolador para a Sonoma deve seguir o formato especificado na Tabela 3.

Tabela 3 - Formato do comando reconhecido pela MAXREFDES14#



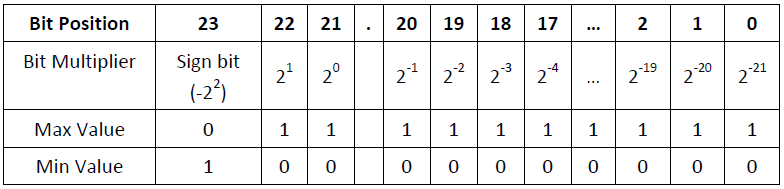
Para preencher o campo ADDR, que se refere ao endereço de memória que contém o valor no qual estamos interessados em obter, deve-se consultar o local do registrador no manual da placa. Na Tabela 4 são demonstrados alguns dos endereços de memória que podem ser utilizados para consulta. A Tabela 10 com todos os parâmetros que podem ser obtidos com a placa de aquisição de dados foi adicionada ao apêndice.

Tabela 4 - Endereço dos registradores da Sonoma MAXREFDES14#

|  |  |  |  |
| --- | --- | --- | --- |
| **End. do Registrador** | **Registrador** | **Tipo** | **Descrição** |
| 2B | VA\_RMS | S.23 | Tensão RMS |
| 3E | IA\_RMS | S.23 | Corrente RMS |
| 4B | WATT\_A | S.23 | Potência Ativa |

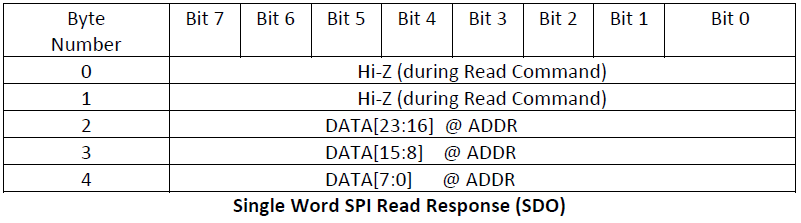
A coluna “Tipo” especifica a forma como o valor medido foi codificado na mensagem. O tipo do dado pode ser “INT”, ou “S.nn”, onde “INT“ indica que um valor inteiro entre 0 e 16777215 contido no 3 bytes de dados, e “S.nn” é uma representação onde o “S” indica o sinal do valor medido, e “.nn” indicam a quantidade de bits de casas decimais fracionárias. Esta conversão é demonstrada na Tabela 5.

Tabela 5 - Método de conversão para variável de leitura



Para que o microcontrolador consiga se comunicar com o protocolo da Sonoma, é necessário ainda ordenar os bits da mensagem de acordo com a tabela a seguir.

Tabela 6 - Ordenação dos bytes para enviar comando



Para a interpretação dos dados retornados pela placa de medição, a informação útil se encontra nas posições 2, 3 e 4 do vetor de bytes da resposta.

Após fazer a leitura do endereço de memória do sensor, sabendo-se o tipo no qual a informação foi armazenada, é possível fazer a conversão dos bytes recebidos para o valor da grandeza física (Volt, Ampere ou Watt, por exemplo). Um exemplo de programa em C que pode ser utilizado para realizar a conversão de valores está presente no apêndice.

A comunicação entre o microcontrolador e placa de aquisição pode ser estabelecida também através de I2C ou UART. Para alterar a forma de comunicação dos dispositivos é necessário consultar o manual de instruções da placa [15].

### 3.1.3.4 – Comunicação entre mote e a placa de aquisição

A comunicação entre o módulo de aquisição e o microcontrolador segue o mecanismo de Mestre/Escravo onde o Mote (CC2538 ou Beaglebone) é mestre, e a Sonoma é escravo. A Sonoma faz medições periodicamente e as mantém registradas no endereço de memória (Tabela 10) acessível pelo microcontrolador.

Na rede 802.15.4 uma tarefa realiza, a cada 250ms, uma leitura de todas as variáveis configuradas para medição. Apesar da placa Sonoma permitir escrita para configuração de alguns parâmetros (como ajustes de calibração) neste projeto são feitas apenas leituras de parâmetros da placa. Todos os parâmetros são armazenados em uma variável global (“ParametrosMedicao”) em memória volátil dentro do microcontrolador conforme mostrado na Figura 19. Estes parâmetros globais são acessíveis pela camada CoAP, que retorna como resposta ao gateway quando solicitado, o valor que está armazenado na variável global.

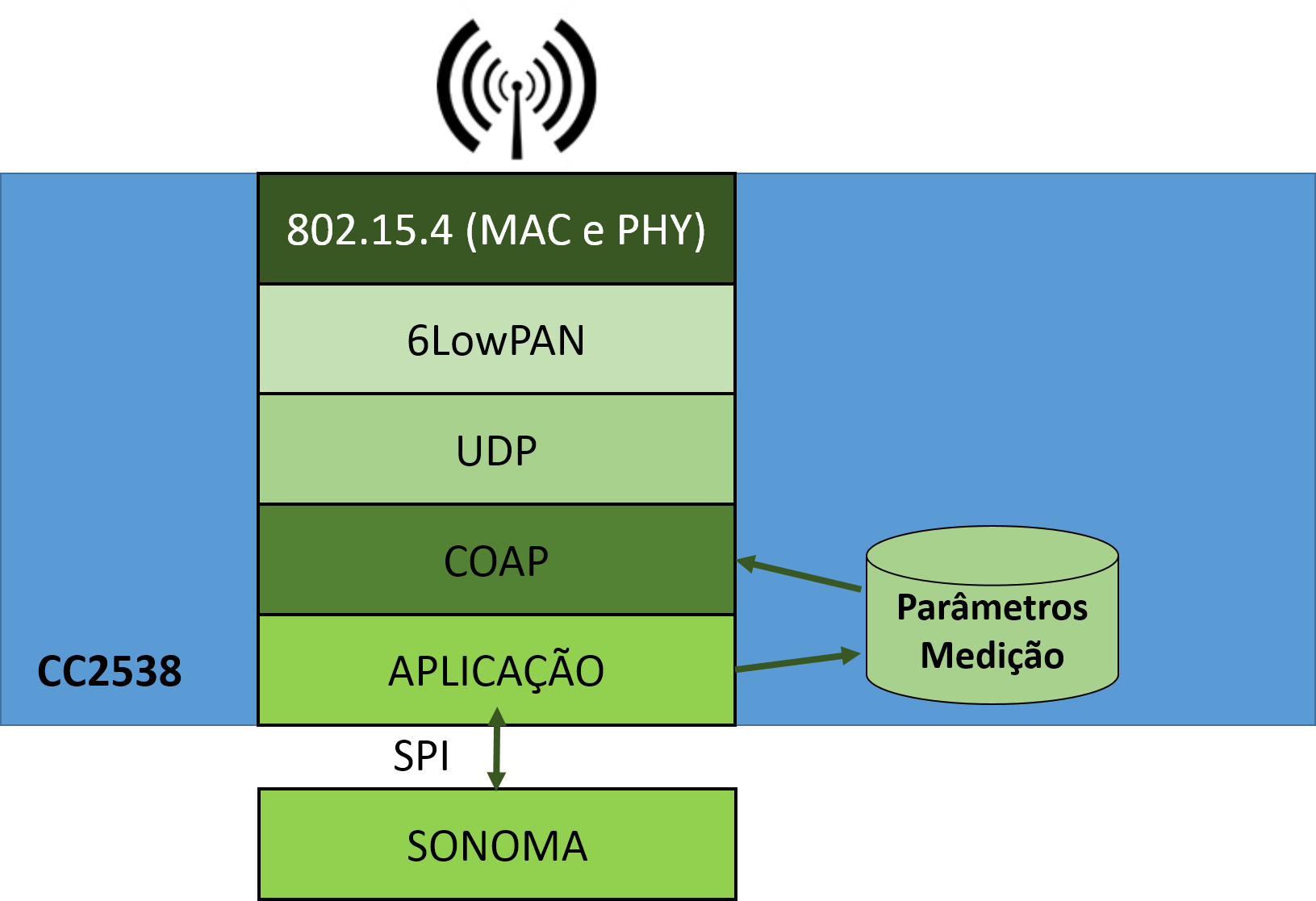


Figura 19 - Comunicação entre mote 802.15.4 e placa de aquisição

Na rede 802.11, o ambiente foi desenvolvido sobre a Beaglebone, que roda um sistema operacional, onde é possível manter um banco de dados (PostgreSQL) armazenando os valores lidos periodicamente (a cada 100ms). Quando o mote recebe uma requisição do gateway via socket Ethernet, a Beaglebone realiza uma consulta do último registro de medição registrado no banco de dados e envia como resposta ao gateway (Figura 20).

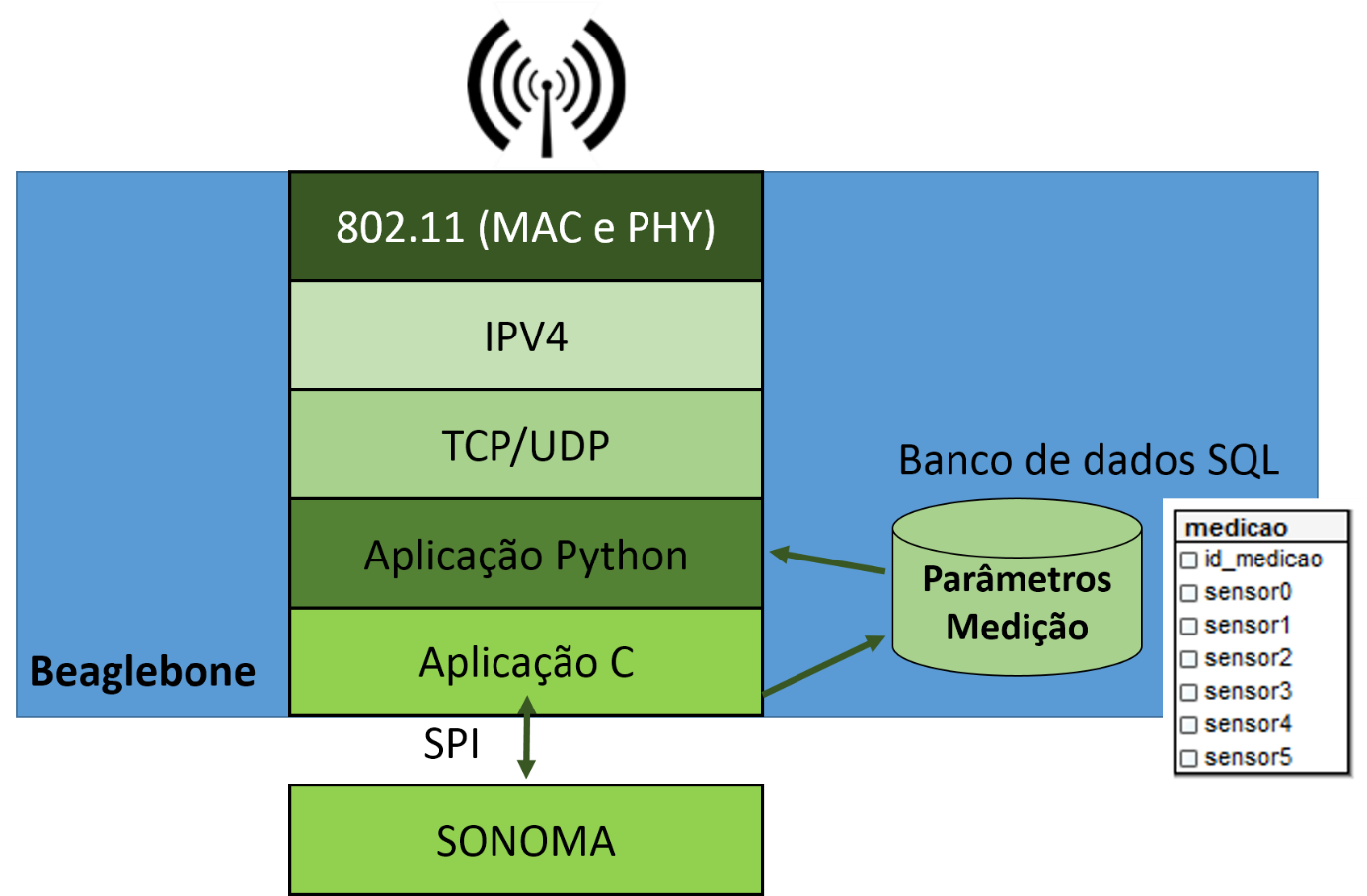


Figura 20 - Comunicação entre mote 802.11 e placa de aquisição

### 3.1.3.5 – Leitura de medições

Os valores retornados pela Sonoma na leitura de medições precisam ser convertidos utilizando um fator de escala. Ao multiplicar o valor salvo no endereço de memória, pelo fator de escala, obtém-se o valor real da medição. As equações 1, 2 e 3 foram retiradas do manual da placa de desenvolvimento MAXREFDES14#, e mostram o cálculo que deve ser feito para obter os principais parâmetros de medição, sendo eles tensão RMS, corrente RMS e potência ativa.

|  |  |  |
| --- | --- | --- |
|  |  | () |

|  |  |  |
| --- | --- | --- |
|  |  | () |

|  |  |  |
| --- | --- | --- |
|  |  | () |

Ao consultar o manual da placa MAXREFDES14# é possível obter os fatores de escala:

|  |  |  |
| --- | --- | --- |
|  | Full Scale Voltage (VFSCALE) = 667Vpk | () |

|  |  |  |
| --- | --- | --- |
|  | Full Scale Current (IFSCALE) = 50Apk | () |

Os fatores de escala apresentados (4 e 5) foram extraídos do site do fabricante, de acordo com o mesmo pode apresentar um erro de ±3%. Uma calibração do sensor pode ser feita para obter maior precisão [16].

### 4 – METODOLOGIA

### 4.1 – Aferição e validação das medições

Para verificar se as medições feitas com o módulo de aquisição de dados estavam sendo medidos corretamente, foi montado um circuito analógico resistivo, a tensão de alimentação foi ajustada em 5 valores diferentes, e foram registradas as medições do modulo de aquisição de dados e os valores medidos pelos equipamentos de laboratório (voltímetro, wattímetro e amperímetro). O esquema de ligação é similar ao representado na Figura 21. A figura 45 apresentada no apêndice mostra uma foto do ambiente de medição montado no laboratório.

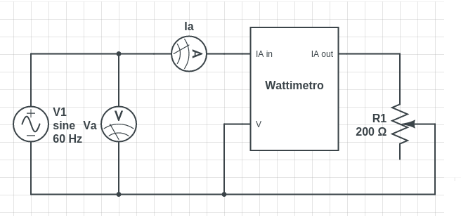


Figura 21 – Montagem para medição de valores

Como resultado dos testes de medição temos a Tabela 7. Nesta tabela estão descritos os valores medidos pelos equipamentos do laboratório, os valores medidos pela Sonoma, e o cálculo percentual da variação entre os valores medidos.

Tabela 7 - Resultado da verificação de medições

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MEDIÇÃO** | | | **SONOMA** | | | **Variação** | | |
| **VA (V)** | **IA (A)** | **Potência (W)** | **VA (V)** | **IA (A)** | **Potência (W)** | **VA (V)** | **IA (A)** | **Potência (W)** |
| 110 | 0.75 | 80 | 111.389 | 0.741 | 82.7747 | -1.263% | 1.200% | -3.468% |
| 150 | 0.97 | 140 | 148.527 | 0.984 | 145.8395 | 0.982% | -1.495% | -4.171% |
| 180 | 1.17 | 200 | 178.367 | 1.183 | 211.4056 | 0.907% | -1.154% | -5.703% |
| 200 | 1.3 | 240 | 199.159 | 1.330 | 315.8245 | 0.420% | -2.308% | -31.594% |
| 220 | 1.94 | 300 | 218.976 | 1.452 | 318.3257 | 0.465% | 25.129% | -6.109% |

A diferença nas medições foi pequena na maioria dos casos, principalmente se considerado que não há evidências de que os equipamentos utilizados para medição estavam aferidos. Nos dois casos pontuais onde a variação foi mais alta, sendo eles a medição de potência para 200V e medição de corrente para 220V a discrepância pode ser atribuída a variações abruptas na rede elétrica que não podiam ser captadas pelos equipamentos utilizados no laboratório. Para a proposta do trabalho pode-se concluir que a medição da Sonoma tem uma boa precisão. Para a utilização do sistema de medição na prática podem ser necessários mais testes para confirmar as variações nas medições.

## 4.2 –Taxa de Entrega fim-a-fim

Para verificar o tempo de resposta do sistema foram desenvolvidas aplicações, que eram executadas no gateway, e requisitavam periodicamente informações do mote. Como o objetivo neste caso era apenas obter os tempos de resposta de cada requisição, os valores de medição tais como corrente, tensão e potência foram ignorados. Apenas os tempos de resposta foram armazenados.

Em ambos os ambientes foram feitas 5 baterias de testes, intercalados e em dias e horários diferentes, ambos em um ambiente fechado, acima da mesma superfície (uma mesa de madeira), e com mote e DAGROOT afastados em aproximadamente 2 metros. O ambiente de testes 808.15.4 é descrito pela Figura 22, e o ambiente 802.11 pela Figura 23.

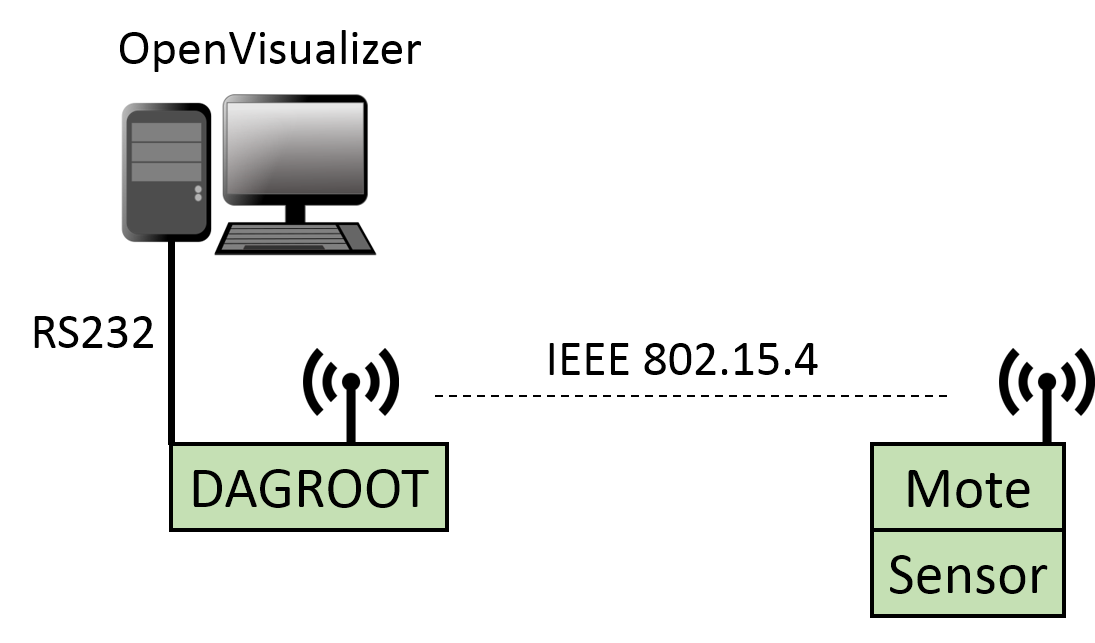


Figura 22 - Ambiente de testes fim-a-fim 802.15.4

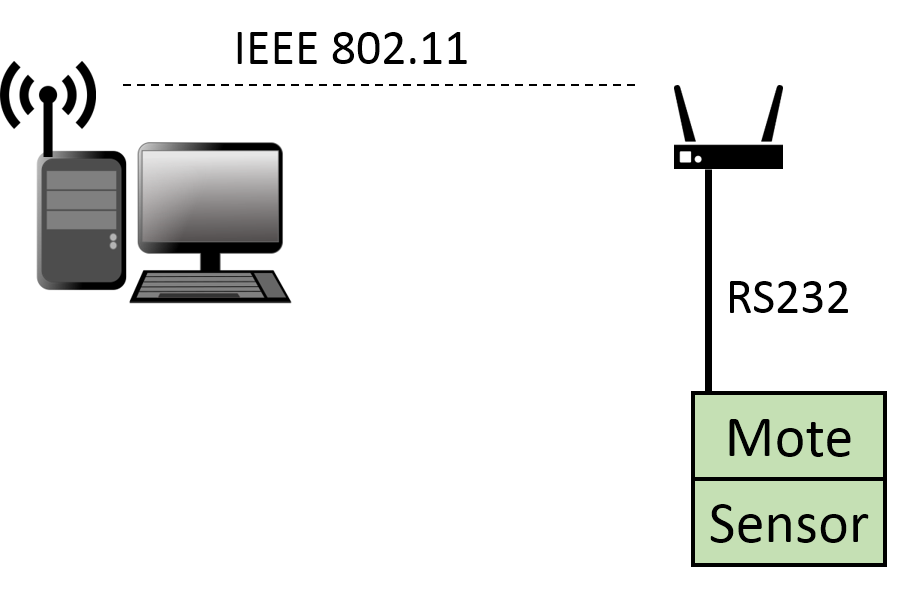


Figura 23 - Ambiente de testes fim-a-fim 802.11

Os tempos de resposta de cada medição foram salvos pela aplicação em um banco de dados relacional (PostgreSQL, Figura 24). Com os dados armazenados em uma tabela SQL foi possível calcular a média, desvio padrão e variância de cada sessão de testes, disponíveis na Tabela 8 e na Tabela 9 do capítulo de resultados.

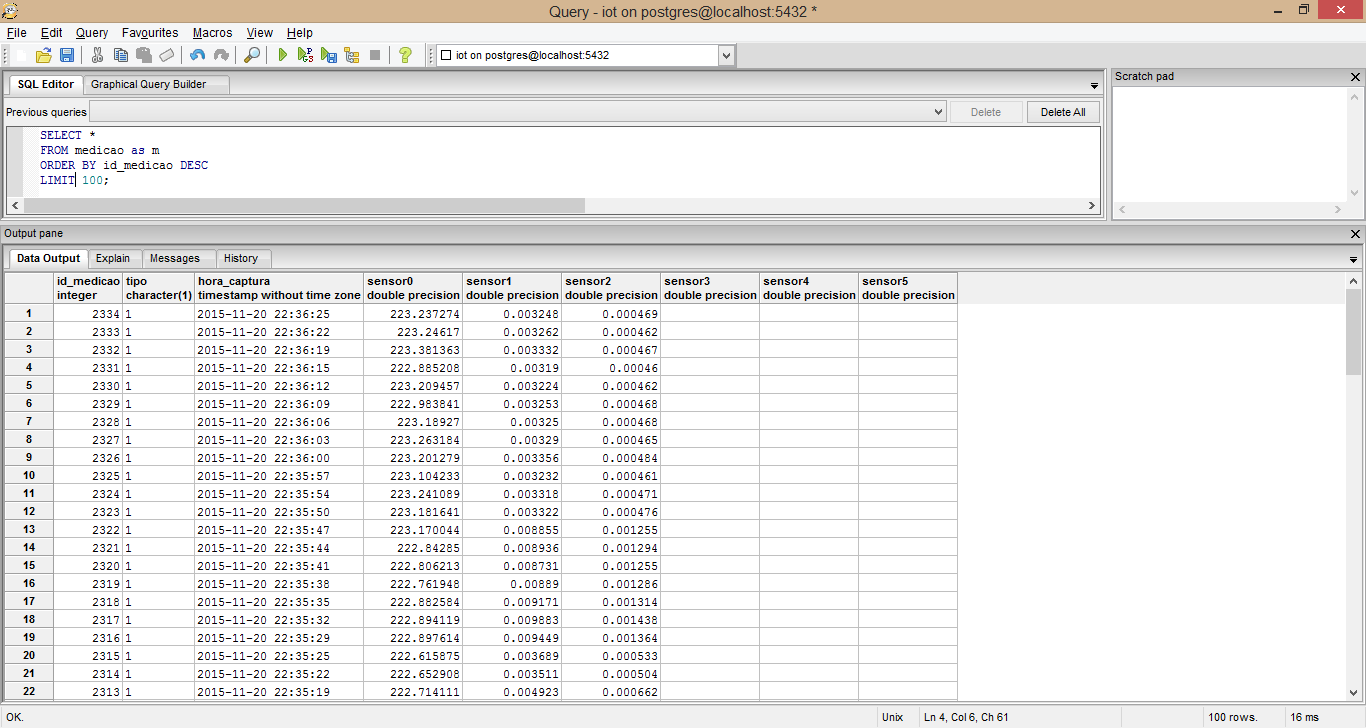


Figura 24 - Obtenção do resultado dos testes utilizando PostgreSQL

## 4.3 – Leitura Remota dos Parâmetros de Medição

Visando executar testes próximos de uma condição real de monitoramento de consumo de energia, foi montado um ambiente (Figura 25 - Ambiente montado para medição de consumo) onde um mote fazia a leitura dos principais parâmetros de consumo de energia de uma lâmpada incandescente de 25W e 220V.

O teste foi feito utilizando o ambiente 802.15.4, com requisições sendo feitas em intervalos de 3 segundos, durante o período de uma hora. Foram registrados a tensão (VA\_RMS), corrente (IA\_RMS) e a potência ativa (WATT\_A) consumidas pelo aparelho.

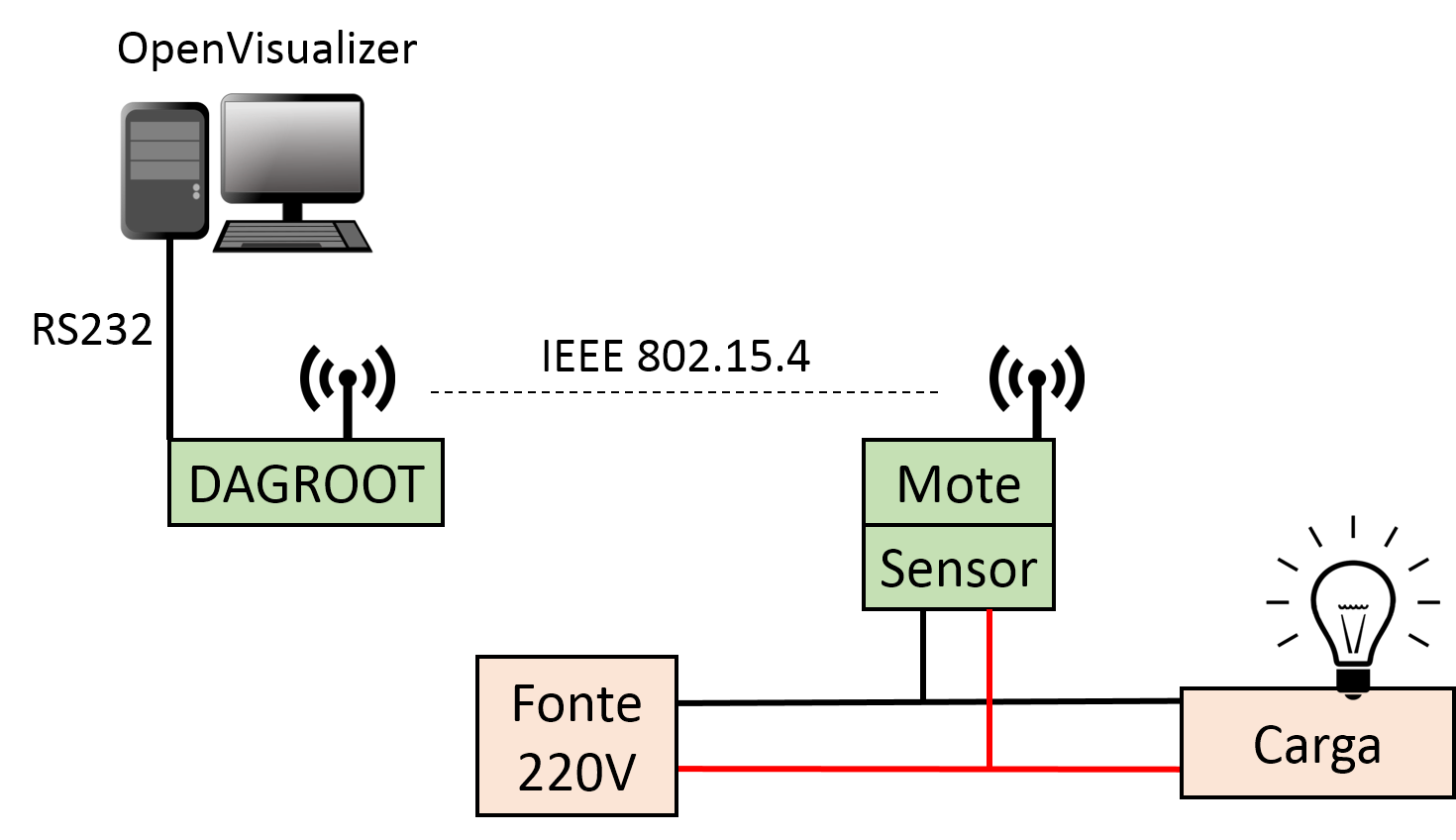


Figura 25 - Ambiente montado para medição de consumo

### 5 – RESULTADOS

### 5.1 – Análise

Testes de desempenho foram realizados com ambos os ambientes de desenvolvimento, Beaglebone Black e Texas Instruments CC2538EM. Para realizar estes testes as placas foram conectadas à placa de aquisição de dados (Sonoma MAXREFDES14#), e foram medidos os tempos de resposta para que o computador cliente conseguisse obter os parâmetros de leitura (tensão RMS, corrente RMS e potência ativa).

Tabela 8 - Resultados dos testes de tempo de resposta via Ethernet

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Quantidade de Medições** | **Pacotes Perdidos** | **Mínimo (ms)** | **Máximo (ms)** | **Média (ms)** | **desvio padrão (ms)** | **Variância** |
| 5803 | 29 | 62 | 969 | 187 | 89 | 7,927 |
| 62732 | 522 | 46 | 991 | 185 | 95 | 9,051 |
| 21496 | 0 | 62 | 957 | 164 | 44 | 1,894 |
| 9178 | 0 | 62 | 784 | 163 | 43 | 1,813 |
| 11154 | 2 | 59 | 891 | 164 | 42 | 1,737 |

Tabela 9 - Resultados dos testes de tempo de resposta via CoAP

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Quantidade de Medições** | **Pacotes Perdidos** | **Mínimo (ms)** | **Máximo (ms)** | **Média (ms)** | **desvio padrão (ms)** | **Variância** |
| 2651 | 1 | 237 | 641 | 383 | 61 | 3,725 |
| 1970 | 5 | 244 | 813 | 386 | 70 | 4,915 |
| 1260 | 2 | 254 | 693 | 378 | 59 | 3,466 |
| 1488 | 11 | 254 | 819 | 381 | 62 | 3,887 |
| 1834 | 9 | 246 | 988 | 384 | 67 | 4,451 |

As medições com tempo de resposta superior a 1000ms (1s) foram consideradas perdidas e foram descartadas. Um programa em Python foi desenvolvido para extrair os resultados do banco de dados, e utilizando a biblioteca “matplotlib” foi possível traçar a curva gaussiana de cada bateria de testes. Estas curvas são mostradas a seguir.

A seguir são apresentadas as curvas Gaussianas de cada uma das 5 baterias de testes feitos com a Beaglebone Black medindo o tempo de resposta da comunicação via Socket Ethernet:

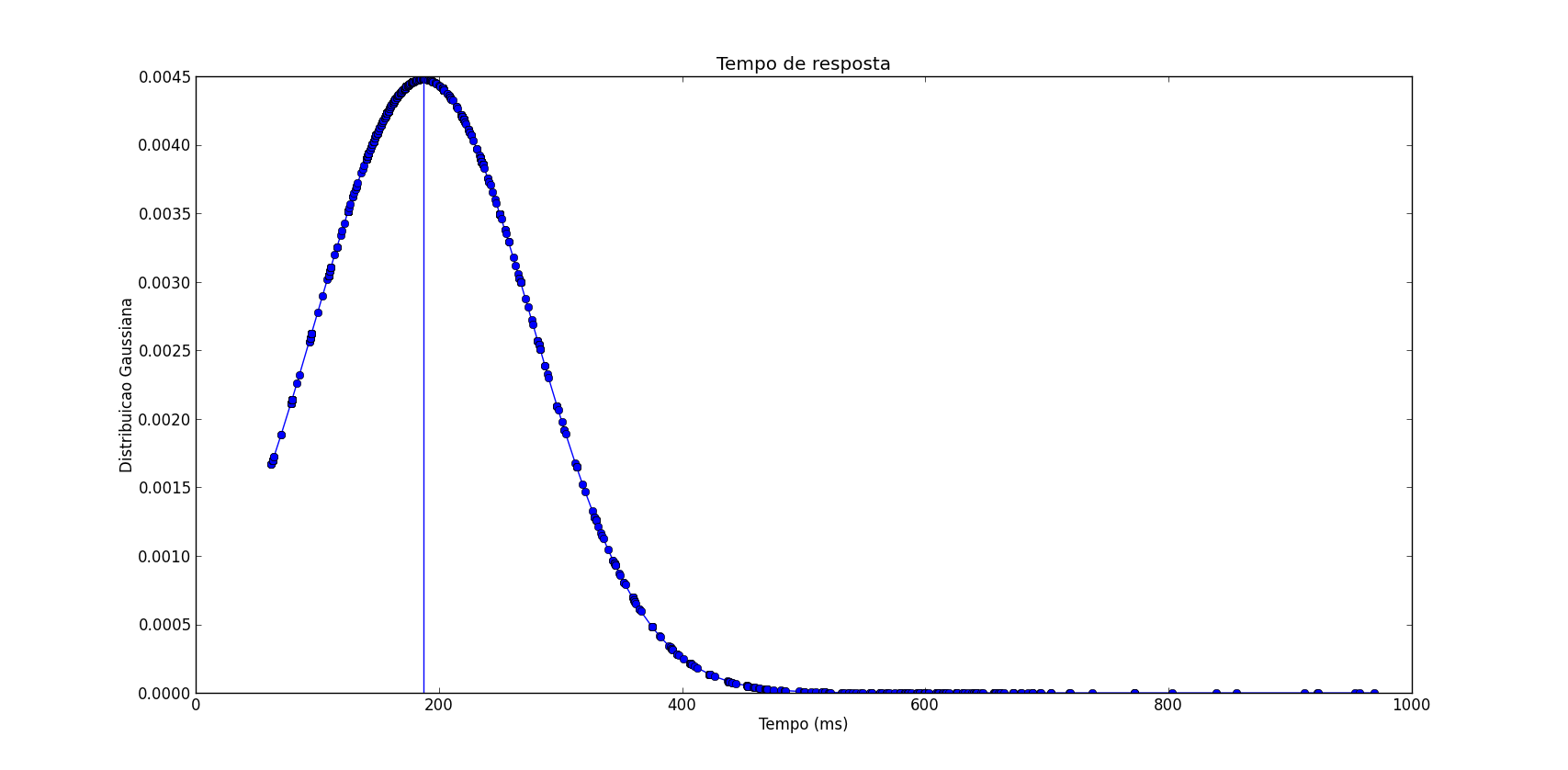


Figura 26 - Curva Gaussiana do tempo de resposta do teste 1 via Ethernet

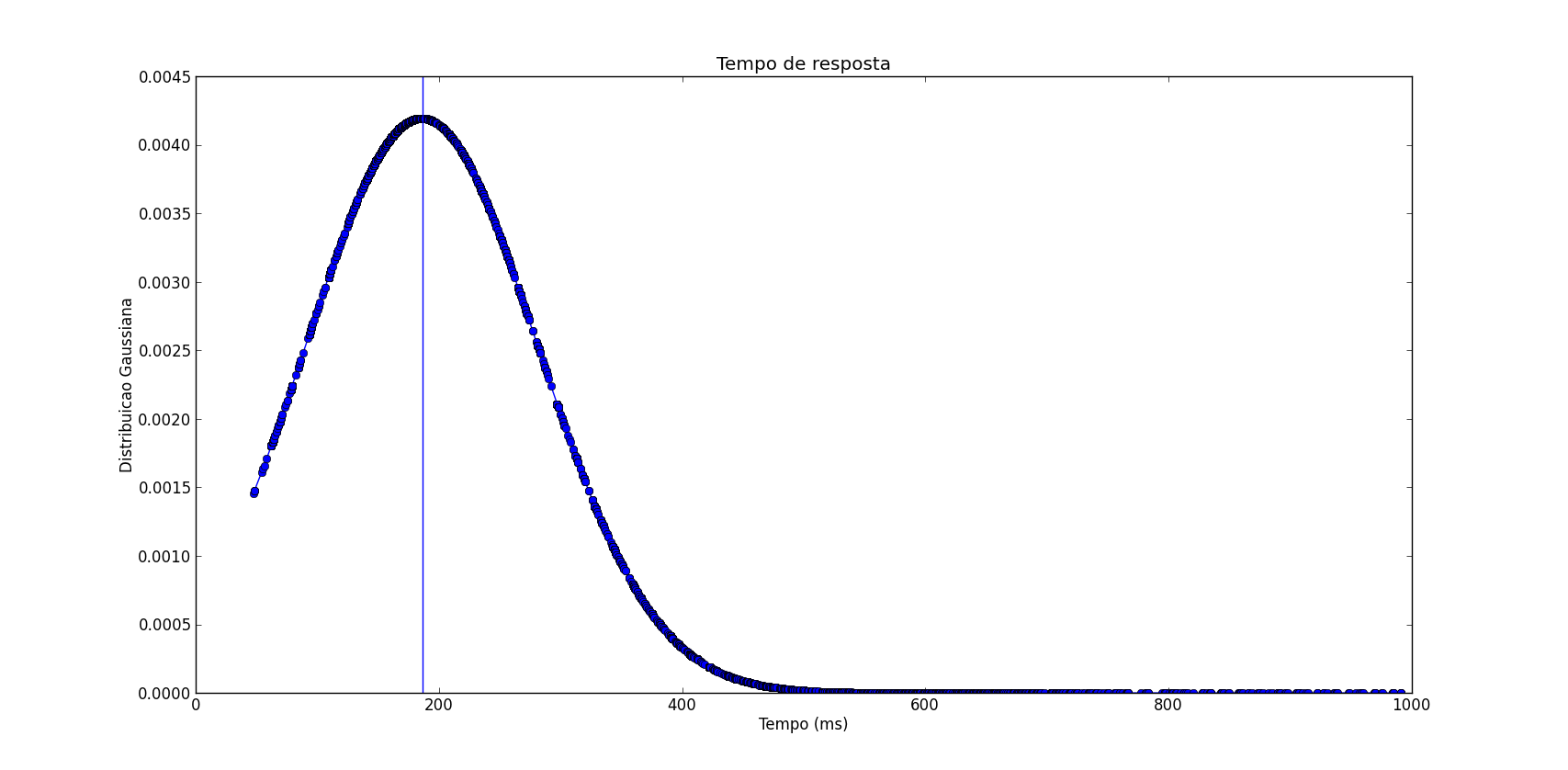


Figura 27 - Curva Gaussiana do tempo de resposta do teste 2 via Ethernet

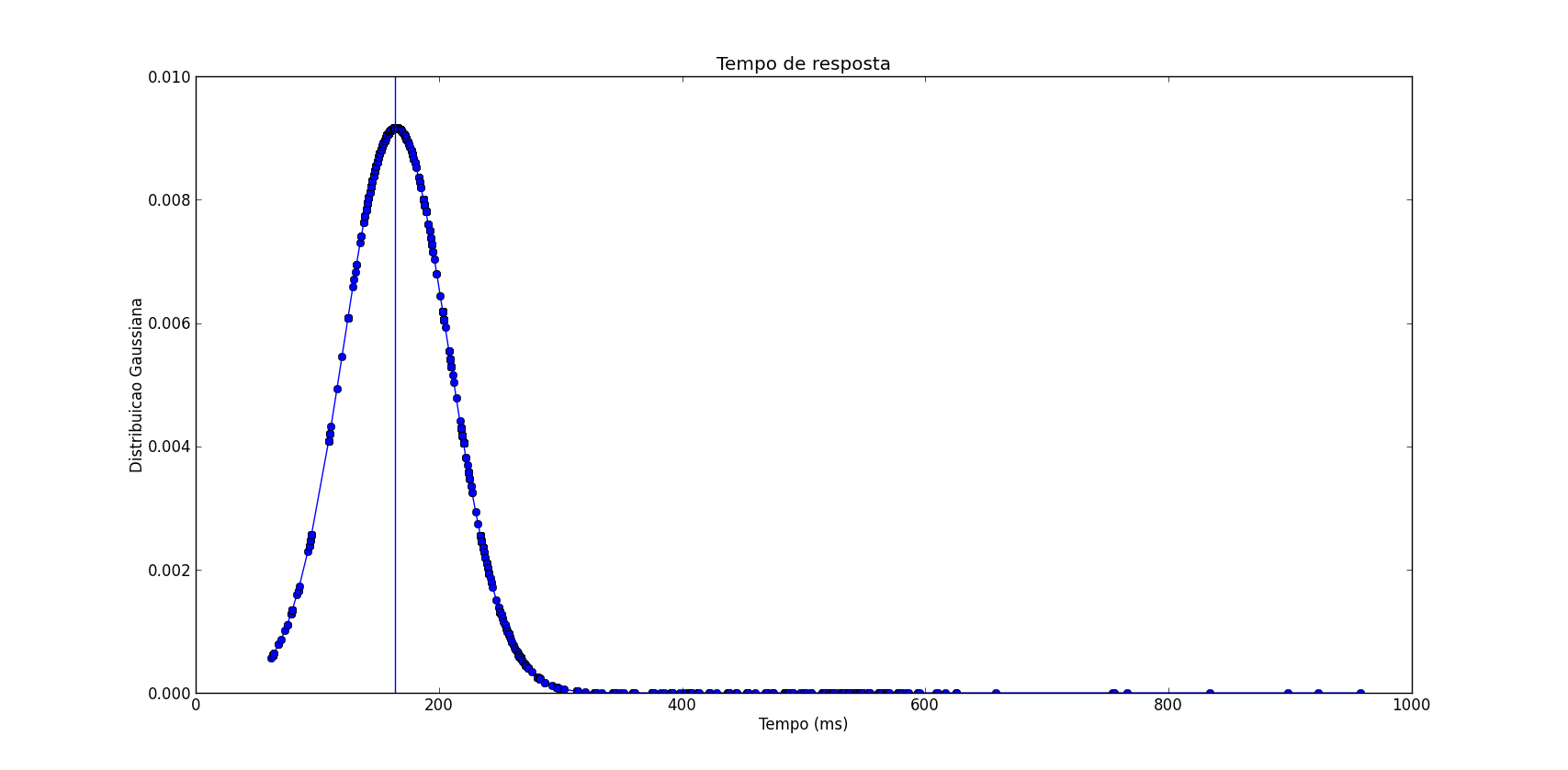


Figura 28 - Curva Gaussiana do tempo de resposta do teste 3 via Ethernet

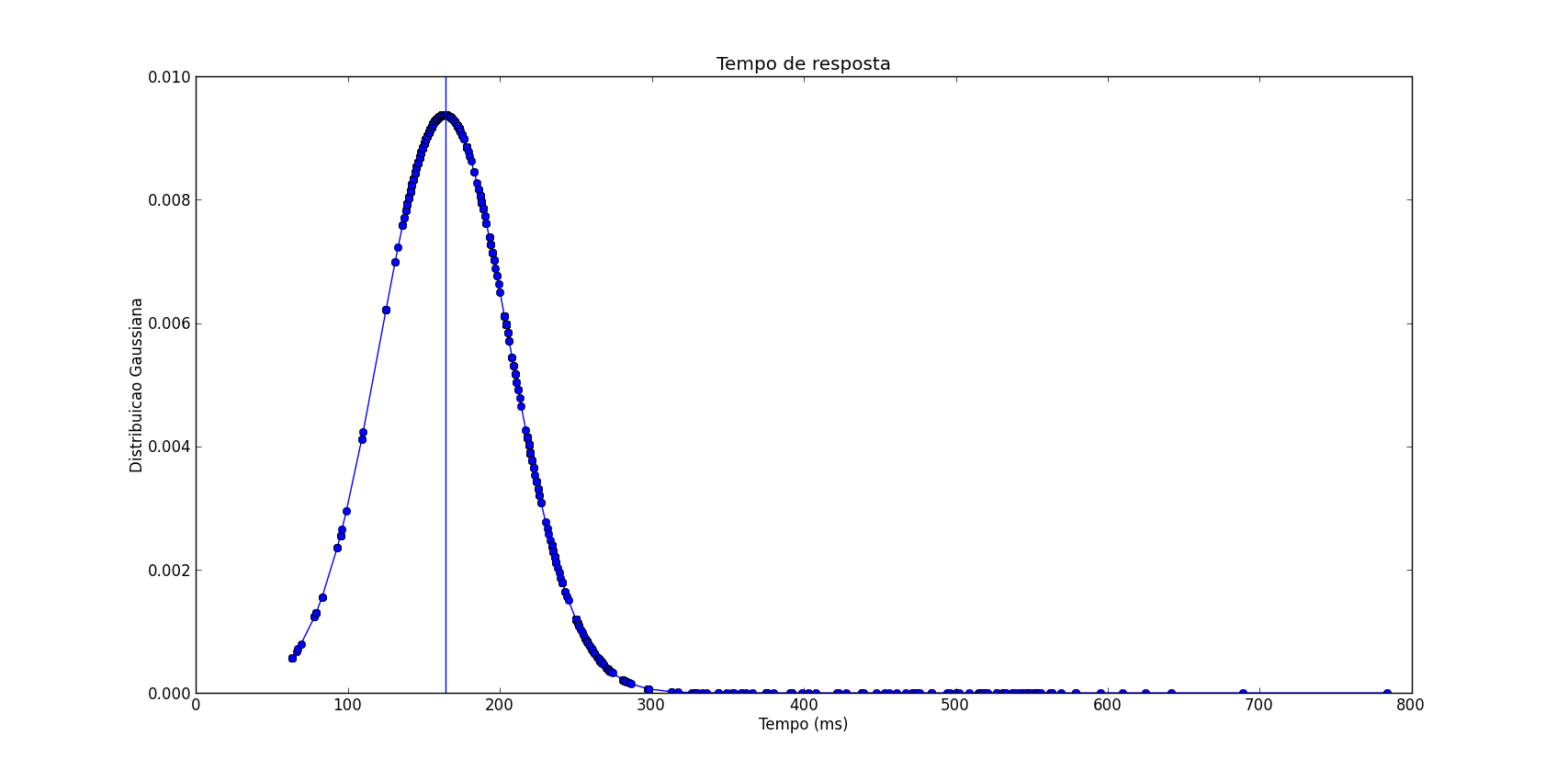


Figura 29 - Curva Gaussiana do tempo de resposta do teste 4 via Ethernet

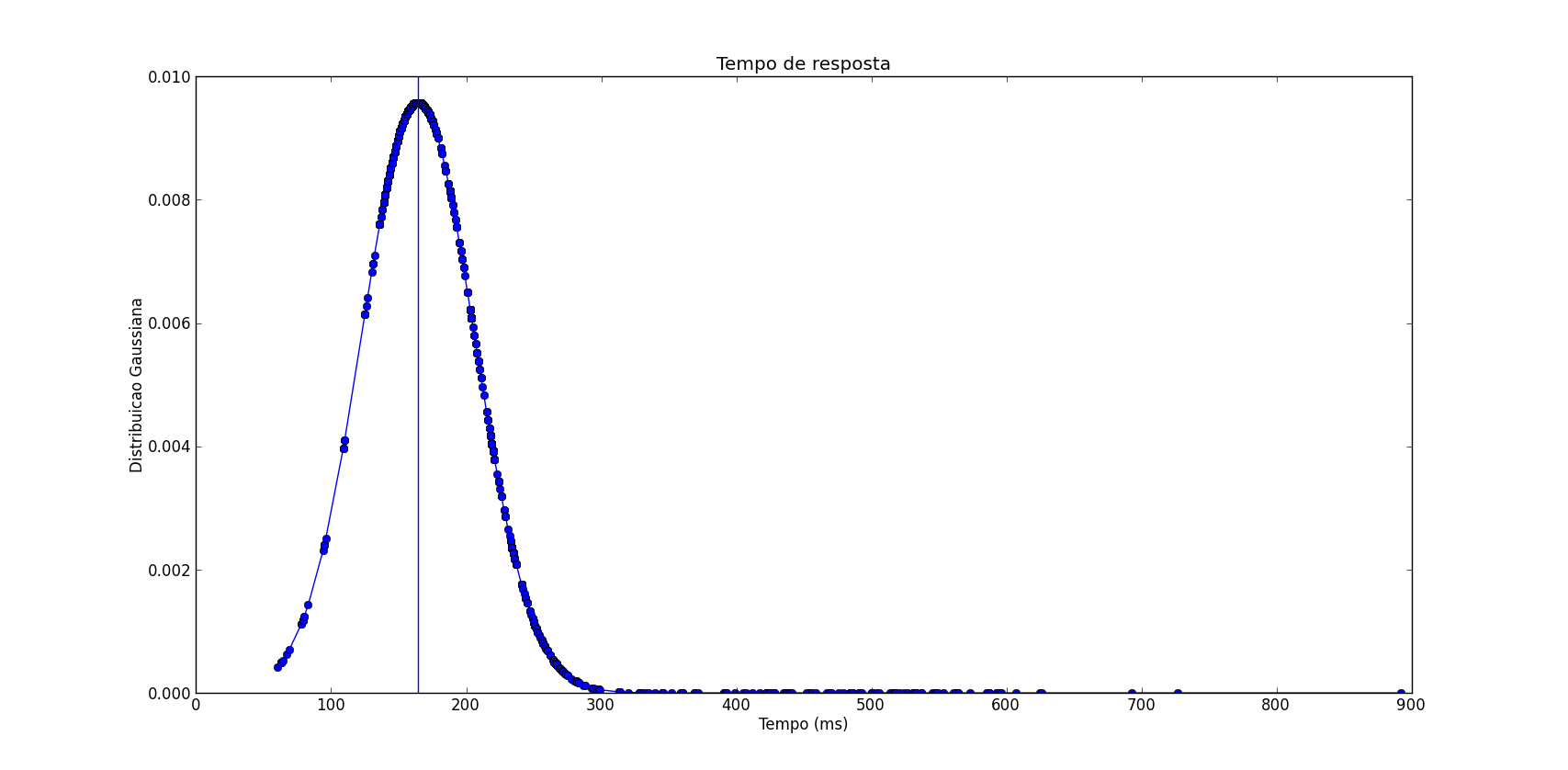


Figura 30 - Curva Gaussiana do tempo de resposta do teste 5 via Ethernet

Comparação entre as 5 curvas Gaussianas:

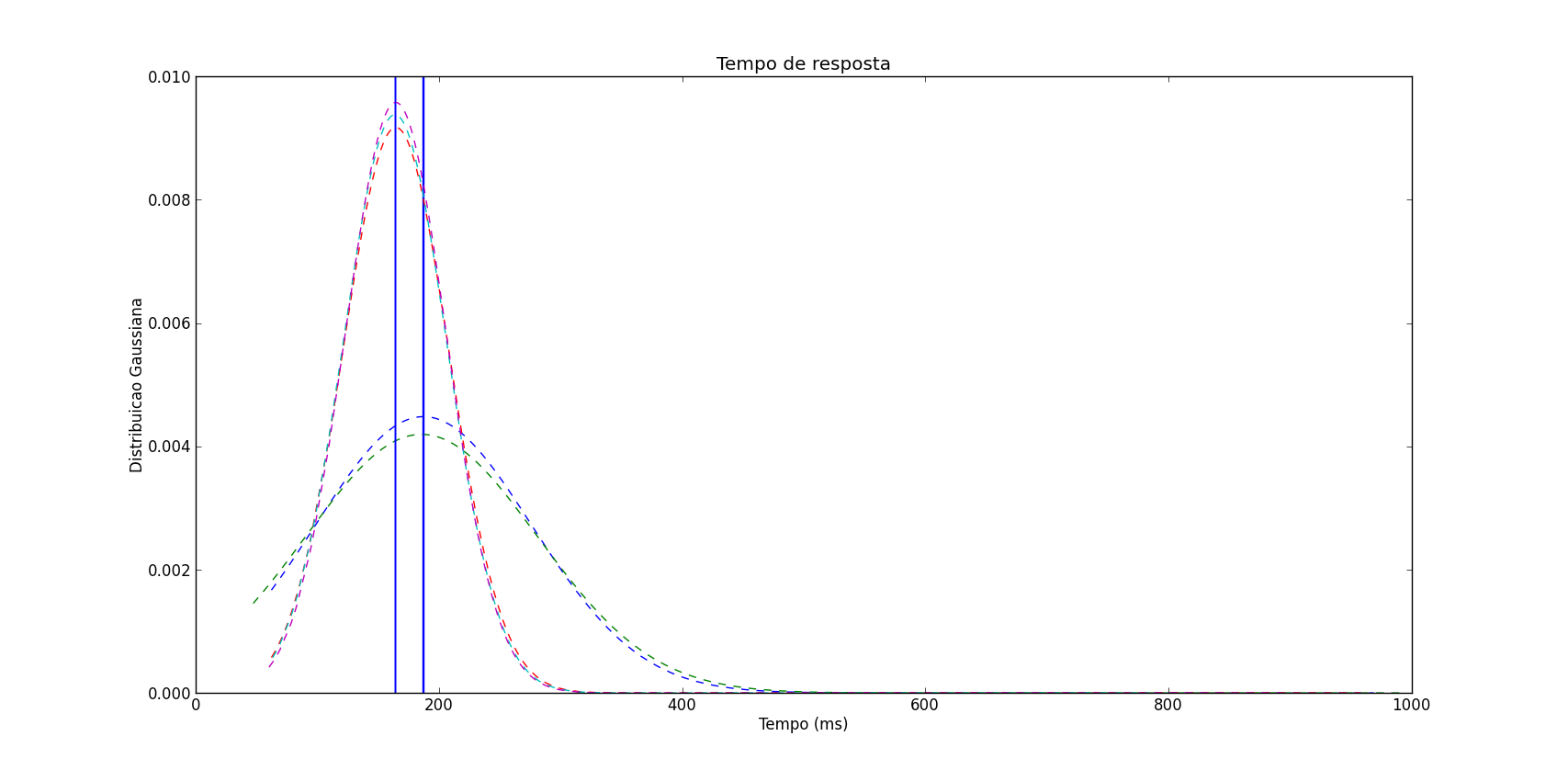


Figura 31 - Comparação das curvas de tempo de resposta via Ethernet

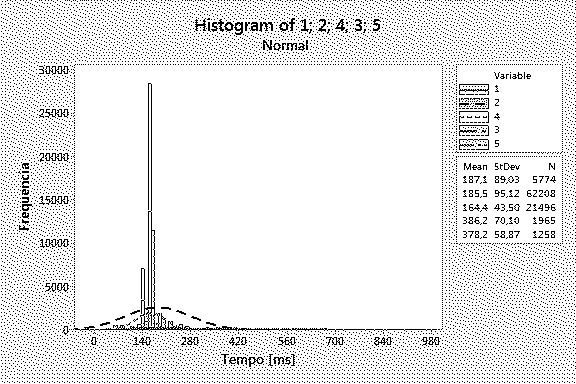


Figura 32 - Histograma dos testes de tempo de resposta via Ethernet

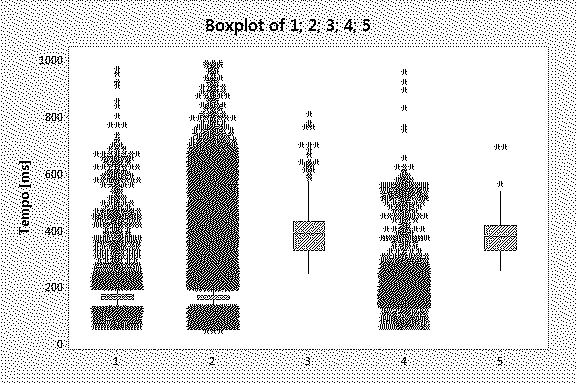


Figura 33 - Boxplot dos testes de tempo de resposta via Ethernet

A seguir são apresentadas as curvas Gaussianas de cada uma das 5 baterias de testes feitos com o mote CC2538 medindo o tempo de resposta da comunicação utilizando a plataforma OpenWSN:

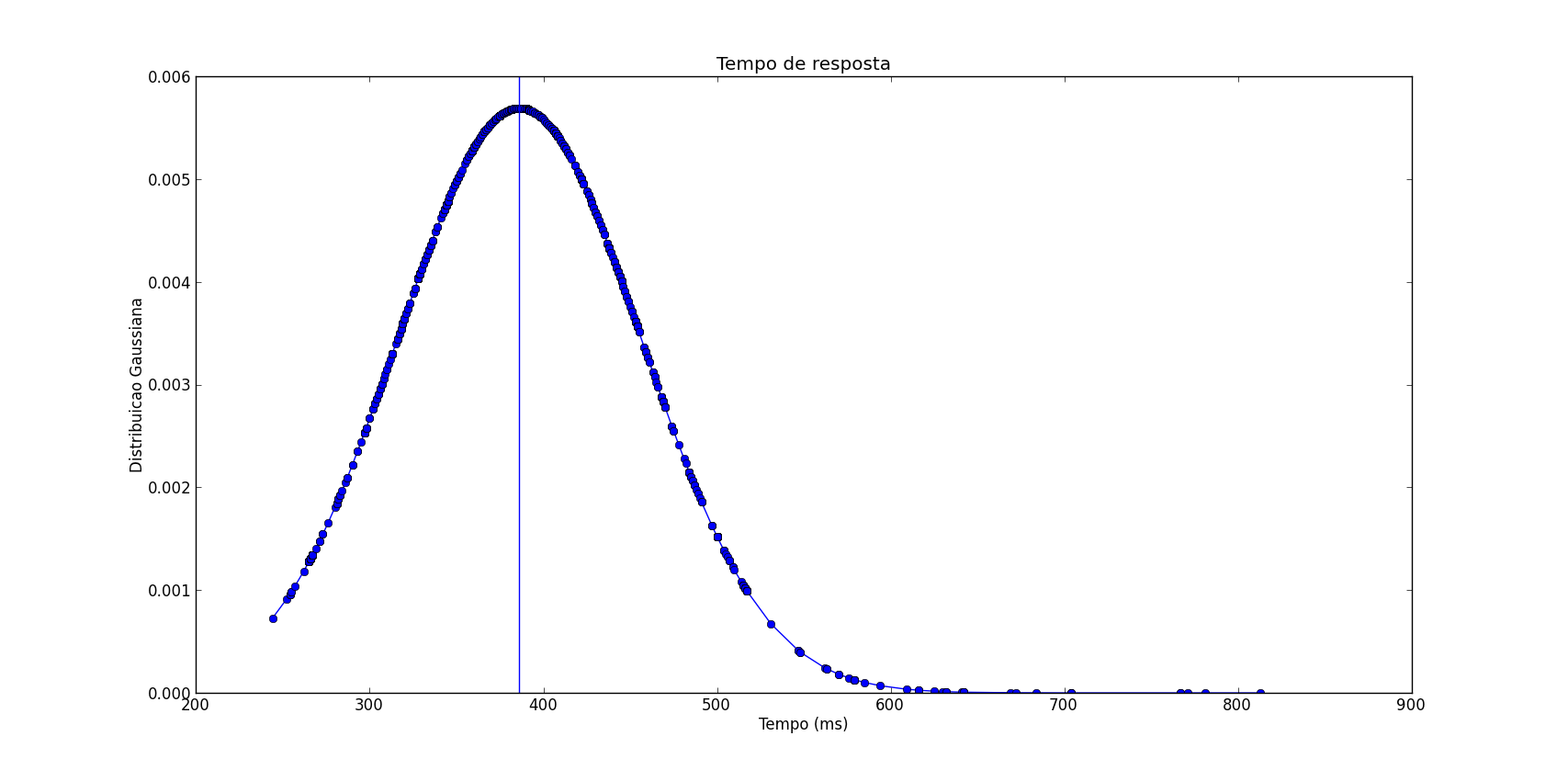


Figura 34- Curva Gaussiana do tempo de resposta do teste 1 via CoAP

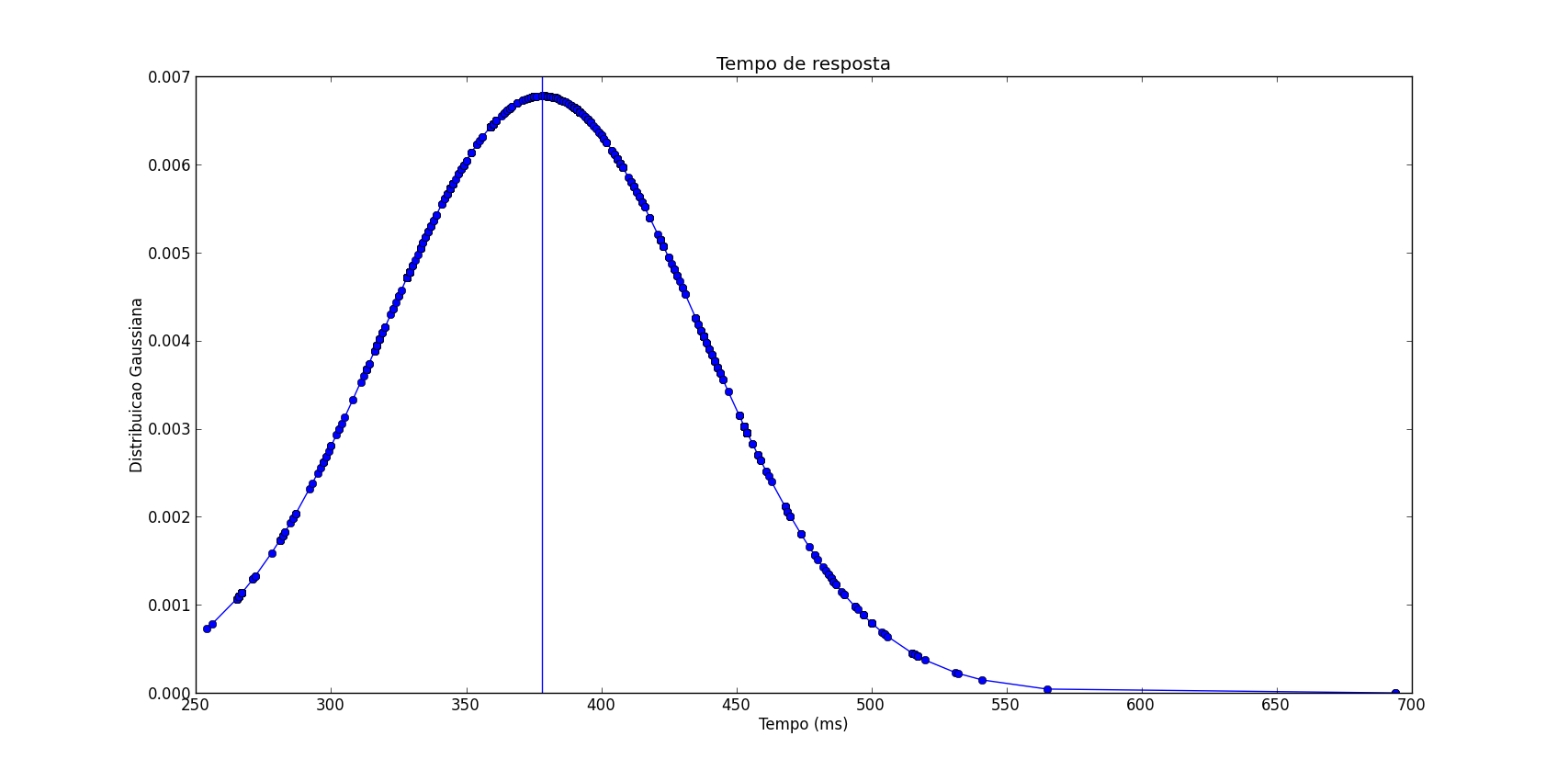


Figura 35- Curva Gaussiana do tempo de resposta do teste 2 via CoAP

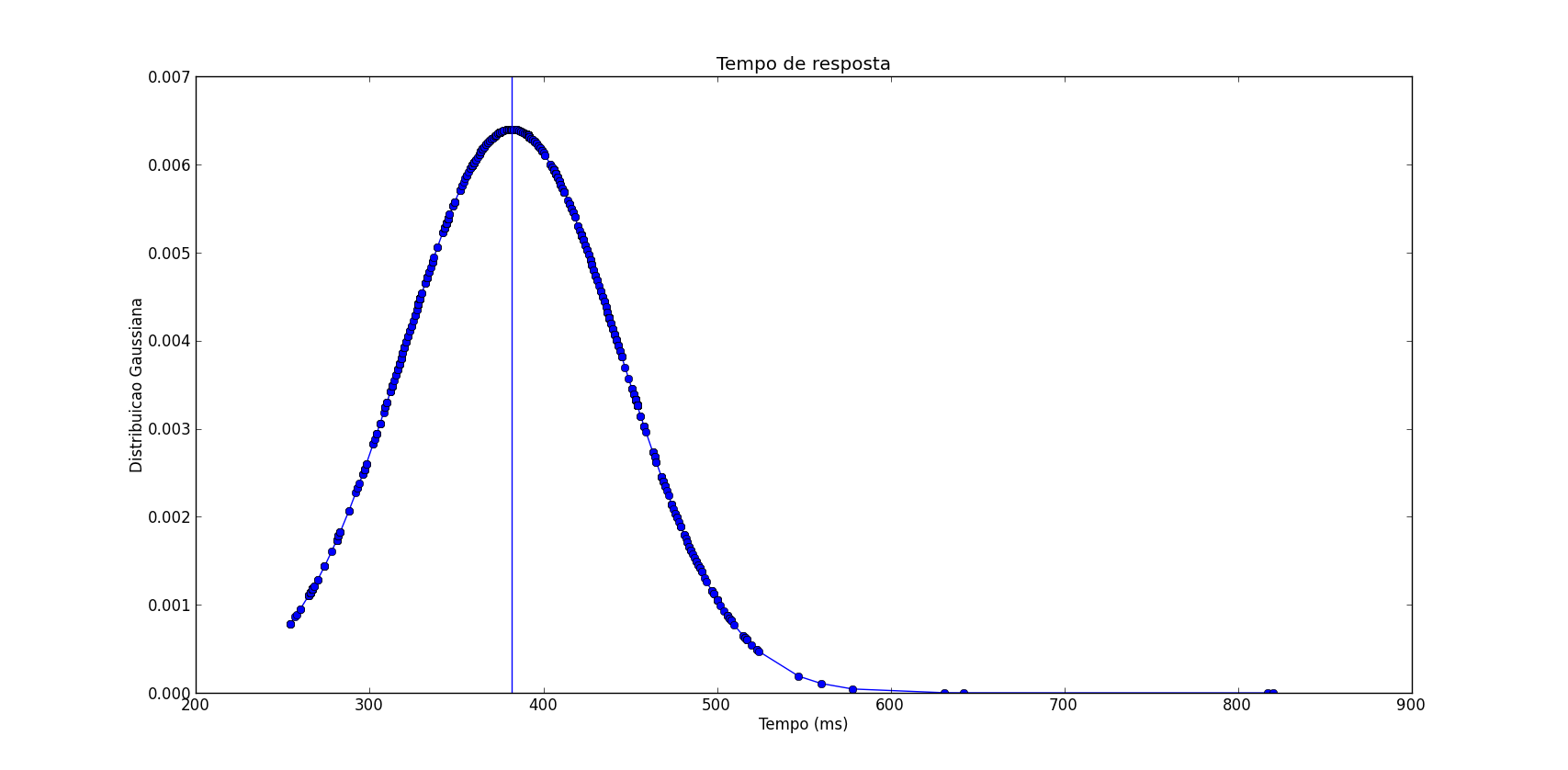


Figura 36- Curva Gaussiana do tempo de resposta do teste 3 via CoAP

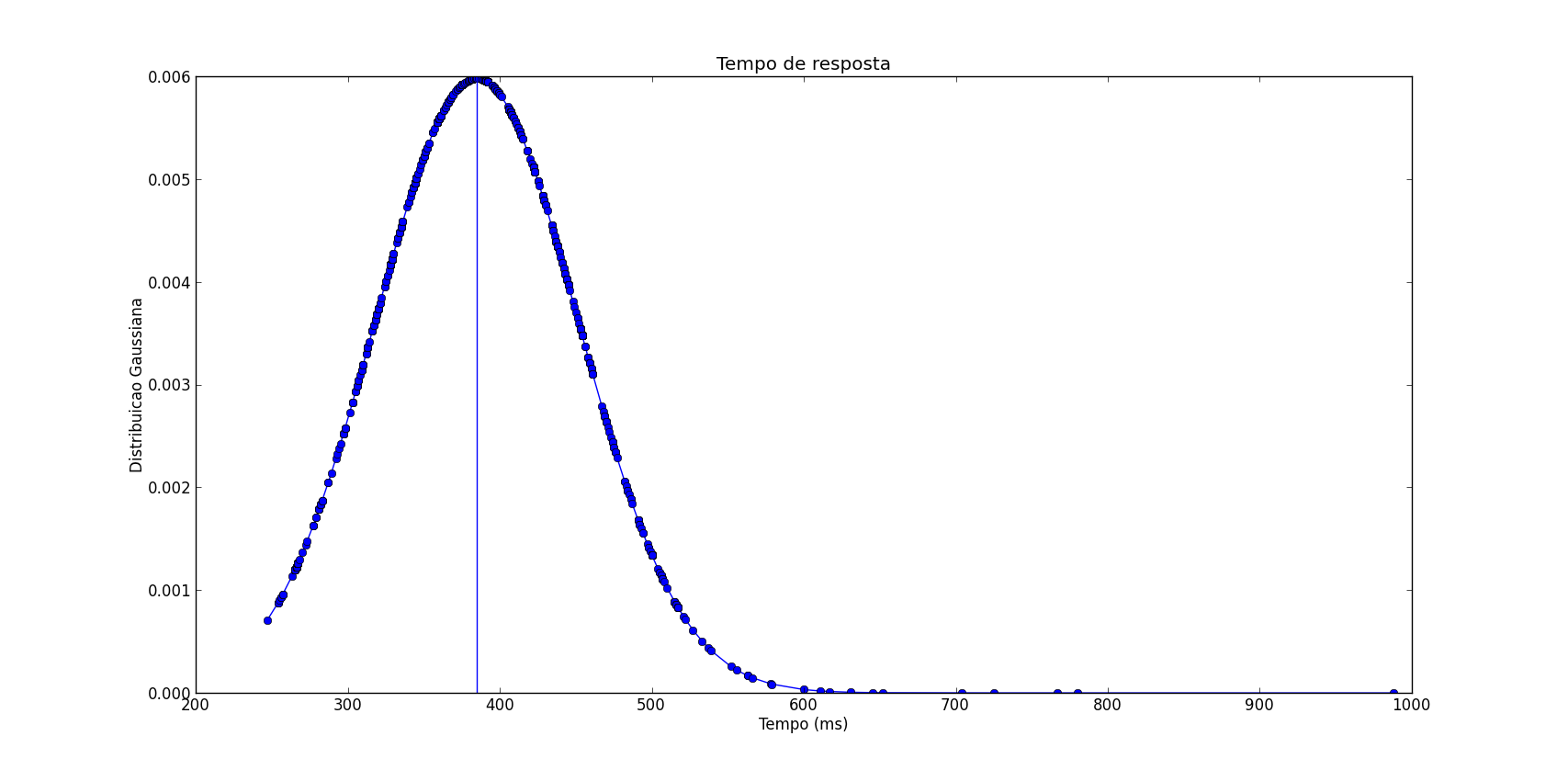


Figura 37- Curva Gaussiana do tempo de resposta do teste 4 via CoAP

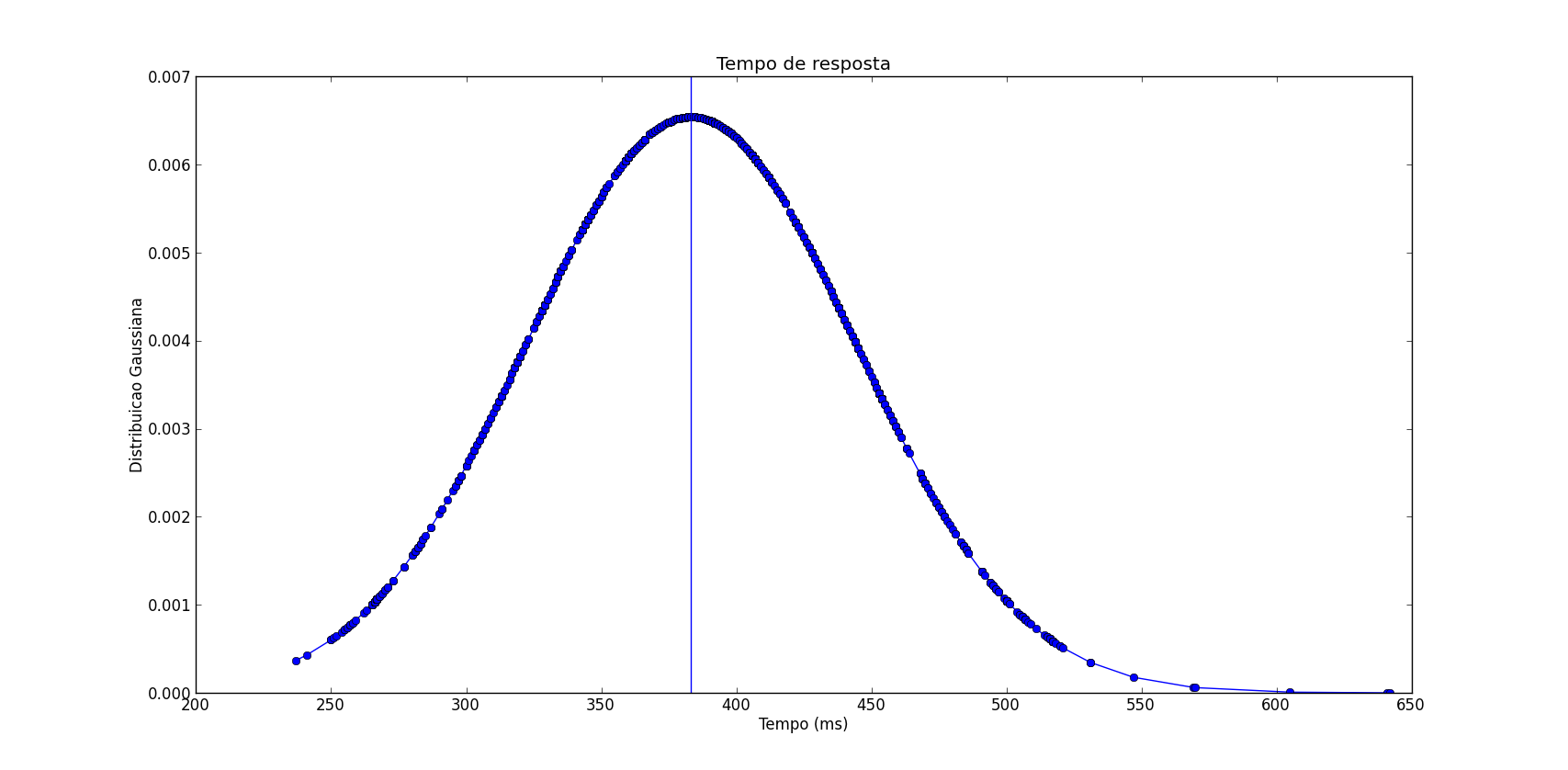


Figura 38- Curva Gaussiana do tempo de resposta do teste 5 via CoAP

Comparação entre as 5 curvas Gaussianas:



Figura 39 - Comparação das curvas de tempo de resposta via CoAP

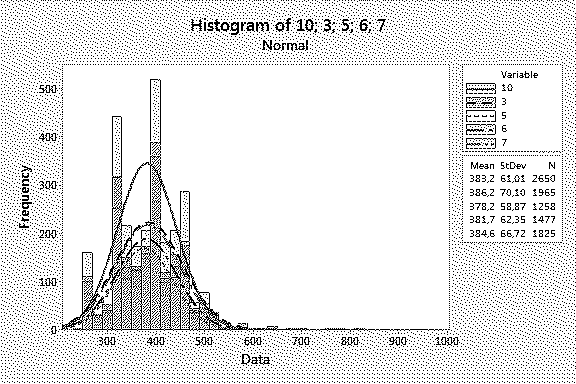


Figura 40 - Histograma dos testes de tempo de resposta via CoAP

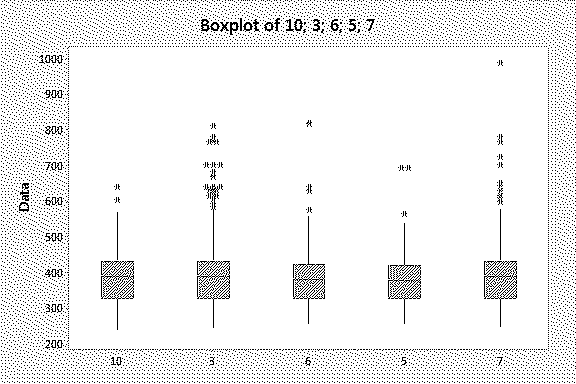


Figura 41 - Boxplot dos testes de tempo de resposta via CoAP

Analisando os resultados do tempo de resposta dos dois sistemas, considerando os dados da Tabela 8 e Tabela 9, observa-se que o intervalo de 1 segundo entre requisições é suficiente para ambos os sistemas.

Após realizar medições dos principais parâmetros de consumo de energia durante uma hora, os dados obtidos foram plotados nos gráficos a seguir. A Figura 42 mostra um gráfico plotado a partir dos registros da leitura remota dos parâmetros de medição. A Figura 43 e a Figura 44 mostram o consumo de corrente e de potência ativa da lâmpada.



Figura 42 - Monitoramento de tensão

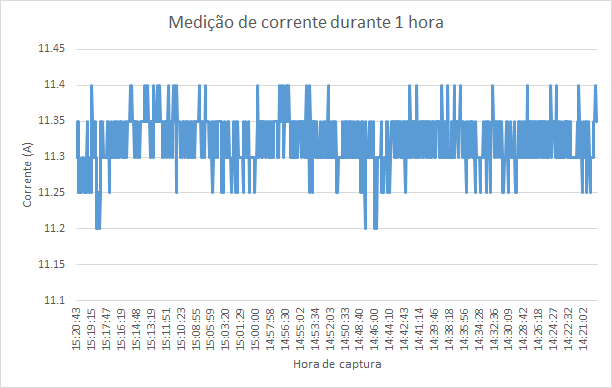


Figura 43 - Monitoramento de corrente

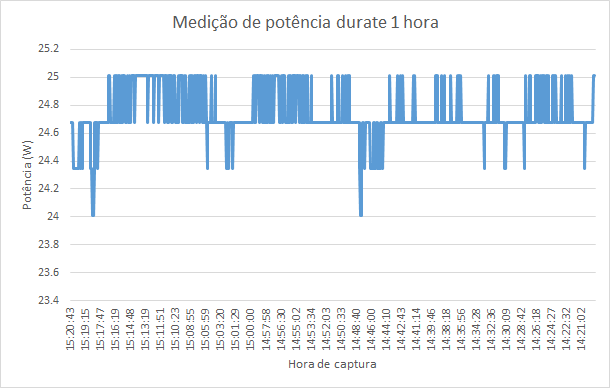


Figura 44 - Monitoramento de potência

### 5.2 – DISCUSSÃO

De acordo com as recomendações da IETF, IEEE, ISA, IEC e OpenWSN, o protocolo mais adequado para o desenvolvimento de uma rede de sensores utilizando tecnologia de Internet das Coisas IoT é o IEEE 802.15.4, presente no microcontrolador CC2538EMTI, que possibilita uma comunicação com baixo consumo de energia e hardware de baixo custo. Comparado com o ambiente IEEE 802.11, que teoricamente é um protocolo de comunicação mais rápido, o tempo de comunicação fim-a-fim da proposta utilizando OpenWSN se mostrou satisfatório.

A placa de aquisição de dados Sonoma MAXREFDES14#, possui uma vasta e detalhada documentação que permite acesso a uma enorme quantidade de parâmetros de medição, e atendeu aos requisitos do projeto, possuindo toda a documentação necessária para o manuseio do hardware e criação do protocolo de comunicação.

### 6 – TRABALHOS PUBLICADOS

Durante o desenvolvimento deste projeto foram apresentados dois artigos, em seminários nacionais, sendo eles:

* III CEEL – XIII Conferência de Estudos em Engenharia Elétrica, UFU, Uberlândia, MG. Brasil. “Proposta de um medidor de consumo utilizando tecnologia de Internet das Coisas”.
* III Congresso de Automação e Inovação Tecnológica Sucroenergética, Sertãozinho, SP. Brasil. “Aplicação da tecnologia de Internet das Coisas na medição de consumo”.

### 7 – CONCLUSÕES

Ambos os ambientes de desenvolvimento, tanto o IEEE 802.11 com a Beaglebone Black, quanto o IEEE 802.15.4 com o CC2538EM, da Texas Instruments, possuem especificações que satisfazem os requisitos do projeto em termos de processamento, memória e comunicação SPI. A principal diferença entre os ambientes é a forma como a comunicação é feita. Na placa da Texas Instruments a comunicação no nível da rede de sensores é feita via CoAP, um protocolo recomendado pelo OpenWSN, e que implementa uma rede mesh. O desenvolvimento de uma rede mesh ou utilização do protocolo CoAP na Beaglebone é possível, mas não há atualmente um padrão já desenvolvido que implemente o CoAP no microcontrolador. Além disso a Beaglebone Black necessita de módulos adicionais para se conectar a uma rede wireless, enquanto a CC2538EM já possui um módulo IEEE 802.15.4 integrado, que permite o desenvolvimento de dispositivos menores, com baixo consumo de energia e com um custo mais baixo.

Existem diversas formas de se introduzir o uso da tecnologia de Internet das Coisas na análise e medição de consumo. As soluções de Internet das Coisas apresentadas neste trabalho se mostraram eficientes e eficazes para a proposta de um medidor de consumo de energia, podendo servir como base para o desenvolvimento de soluções comerciais que tenham o mesmo objetivo.

### 8 – REFERÊNCIAS

[1] Zanella, A.; Bui, N.; Castellani, A.; Vangelista, L.; Zorzi, M. (2014) - Internet of Things for Smart Cities, Internet of Things Journal, IEEE, 2014.

[2] Beaudaux, J.; Gallais, A.; Montavont, J.; Noel, T.; Roth, D.; Valentin, E. (2014) - Thorough Empirical Analysis of X-MAC Over a Large Scale Internet of Things Testbed, Sensors Journal, IEEE, Vol. 14, Issue 2, pp 383-392, 2014.

[3] OpenWSN 2015. OpenWSN Protocol Stack. <https://openwsn.atlassian.net/wiki/display/OW/Protocol+Stack, 2015-01-23.> Acesso em: 03 dez. 2015.

[4] Palattella, M.R., Accettura, N., Vilajosana, X. Watteyne, T., Grieco, L.A., Boggia, G.; Dohler, M. (2013). Standardized protocol stack for the internet of (important) things. Communications Surveys Tutorials, IEEE, 15(3):1389–1406, Third 2013.

[5] Li, S., XU, L., Zhao, S. (2015) - The internet of things: a survey, Information Systems Frontiers, volume 17, number, 2, pp 243-259, Springer US, doi 10.1007/s10796-014-9492-7,2015.

[6] Chang, K.H.; Mason, B. (2012) – The IEEE 802.15.4g standard for smart metering utility networks, In Proceeding: Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on, pp 476-480, 2012.

[7] IEEE802.15.4e (2012) - The Institute of Electrical and Electronics Engineers (IEEE). Amendment to ieee std 802.15.4-2011) - part 15.4: Low-rate wireless personal area networks (lr-wpans) - amendment 1: Mac sublayer), http://standards.ieee.org/findstds/standard/802.15.4e-2012.html, 2014-01-23. [9] J. Beaudaux. Wireless sensor networks. Tese da Universidade da Franca, 2013.

[8] Thubert, P., Watteyne, T., Palattella, M.R., Vilajosana, X. Wang, Q. (2013) - IETF 6TSCH: Combining IPv6 Connectivity with Industrial Performance. In Proceeding: 2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing.

[9] Dutta, P.; Haggerty, S.D.; Chen, Y.; Liang, C.J.M., Terzis, A. - A-MAC: A versatile and efficient receiver-initiated link layer for low-power wireless., ACM Trans. Sen. Netw., Vol. 8, Issue 4, Pages 1-29, 2012.

[10] Winter, J. M. – Software De Análise de Roteamento de Dispositivos WirelessHart**.** 2010. 117 f. TCC (Graduação) - Curso de Engenharia Eletrica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2010.

[11] Antonopoulos, C., Prayati, A., Stoyanova, T., Koulamas, C. and Papadopoulos, G., (2009). “Experimental Evaluation of a WSN Platform Power Consumption”, IEEE International Symposium on Par-allel and Distributed Processing (IPDPS).

[12] Casilari, E., Cano-García, J. M., and Campos-Garrido, G., (2010). “Modeling of Current Consumption in 802.15.4/ZigBee Sensor Motes”, Sensors.

[13] INTERNET ENGINEERING TASK FORCE (IETF) (Org.). RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. Disponível em: <http://tools.ietf.org/html/rfc6550>. Acesso em: 11 jul. 2015.

[14] JANSSEN, Cory. Internet of Things (IoT). Disponível em:

<http://www.techopedia.com/definition/28247/internet-of-things-iot>. Acesso em: 11 jul. 2015.

[15] Maxim Integrated. Datasheet. MAX78615+LMU Isolated Energy Measurement Processor. Eletronic Publication, 2012.

[16] SONOMA (MAXREFDES14#): Isolated Energy Measurement. Disponível em:

<https://www.maximintegrated.com/en/app-notes/index.mvp/id/5723>. Acesso em: 04 dez. 2015.

[17] Projeto OpenWSN. Disponível em:

<https://openwsn.atlassian.net/wiki/display/OW>. Acesso em: 04 dez. 2015.

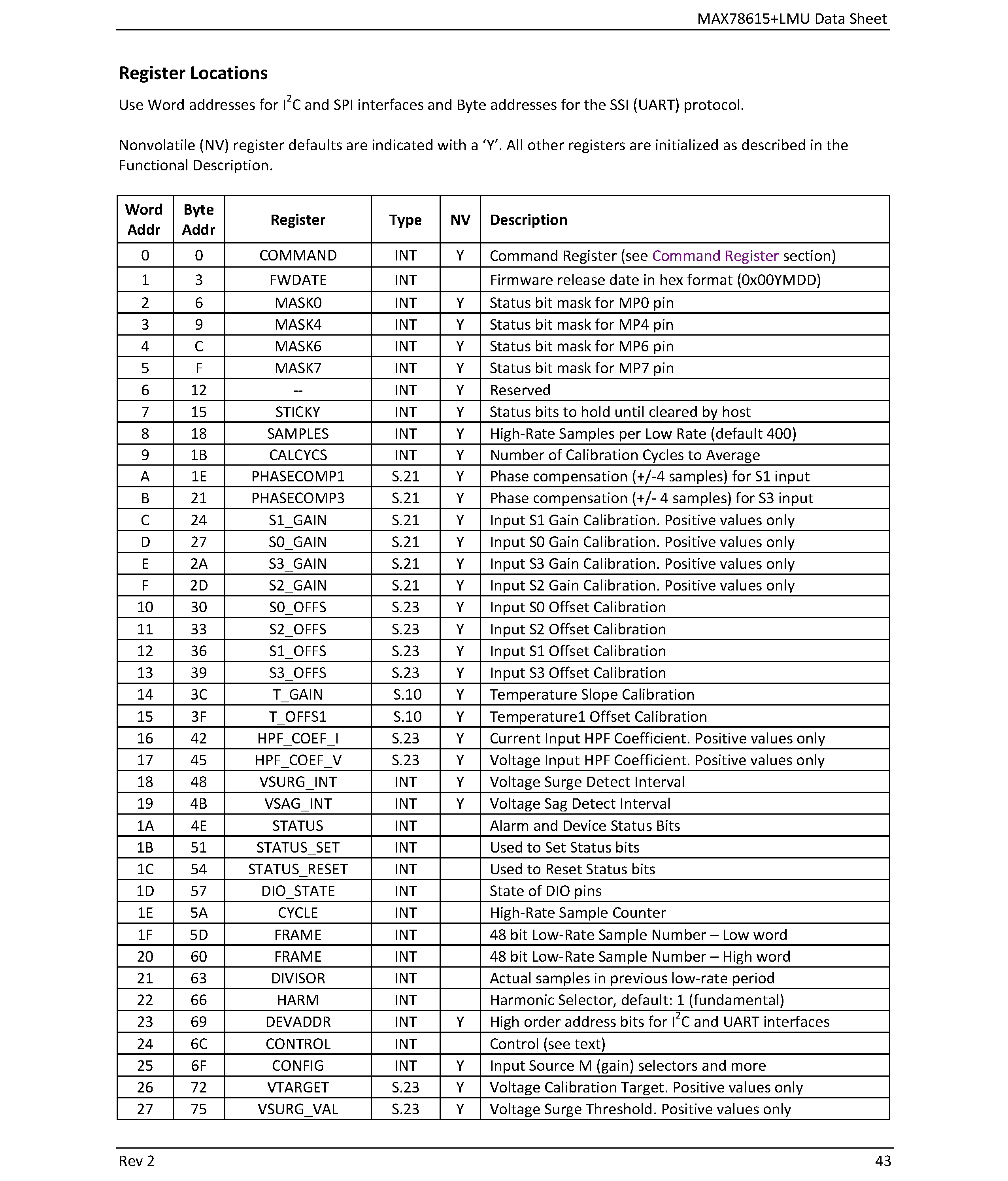
### 8 – APÊNDICE

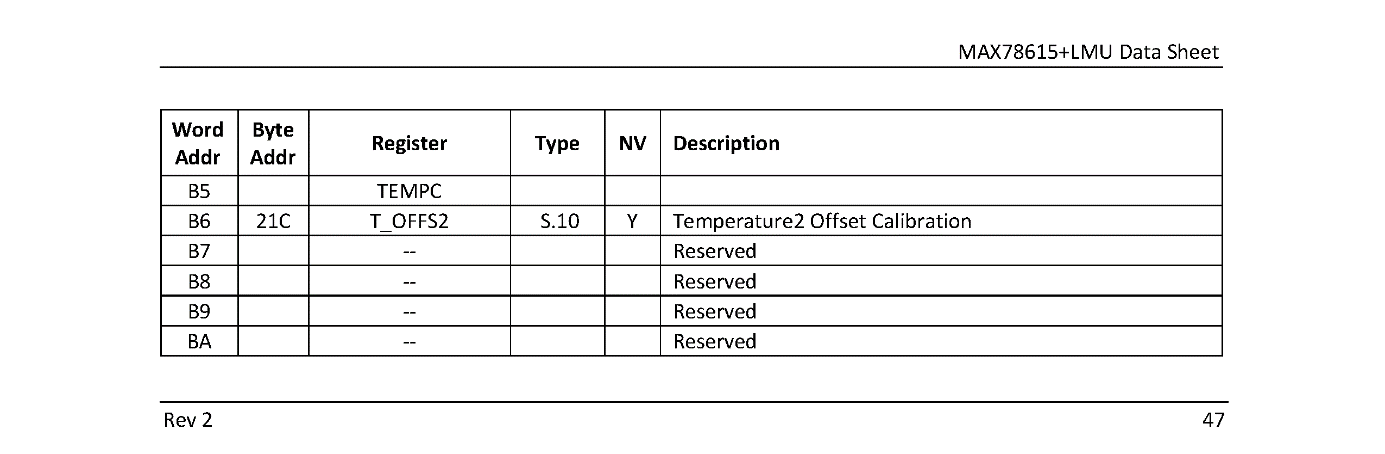


Figura 45 – Ambiente de testes OpenWSN com relé

Tabela a seguir mostra todos os parâmetros que podem ser lidos com a placa Maxim Sonoma MAXREFDES14#, assim como os endereços de registradores utilizados na sessão 3.4.3 – Placa de Aquisição de Dados.

Tabela 10 - Parâmetros de medição da Sonoma MAXREFDES14#



file-page44file-page45file-page46

Script de conversão em C para valores do tipo “INT”:

|  |
| --- |
| uint32\_t CalculaInt(uint8\_t \*rx){  uint32\_t retorno;  retorno = rx[2] + (rx[1] << 8) + (rx[0] << 16);  return retorno;  } |

Script de conversão em C para valores do tipo “S.nn”:

|  |
| --- |
| // Método analítico genérico.  // Passar o parâmetro 'int stype' com o valor 23 para S.23, 21 para S.21...  // O primeiro parâmetro recebido deve ser o endereço do primeiro dos 3 bytes  // de valores da resposta da placa de aquisição array  float CalculaSType(uint8\_t \*rx, int stype){  float retorno = 0;  int k;  int j;  for (int i = 1; i < 23; ++i)  {  if(i > 15){  k = 2;  j = 23 - i;  } else  if (i > 7){  k = 1;  j = 15 - i;  } else {  k = 0;  j = 7 - i;  }  if(i< 24 - stype){  retorno += rx[k] & (int)pow(2,j) ? 1/pow(2, (22 - stype)) : 0;  } else {  retorno += rx[k] & (int)pow(2,j) ? 1/pow(2,i) : 0;  }  }  retorno = rx[0] & 128 ? (-1) \* retorno : retorno; // Verifica o sinal  return retorno;  } |

Script de conversão em C para valores do tipo “S.23” de forma não recursiva:

|  |
| --- |
| float CalculaS23(uint8\_t \*rx){  float retorno = 0;  retorno += rx[0] & 64 ? 0.5 : 0; // 1/2  retorno += rx[0] & 32 ? 0.25 : 0; // 1/4  retorno += rx[0] & 16 ? 0.125 : 0; // 1/8  retorno += rx[0] & 8 ? 0.0625 : 0; // 1/16  retorno += rx[0] & 4 ? 0.03125 : 0; // 1/32  retorno += rx[0] & 2 ? 0.015625 : 0; // 1/64  retorno += rx[0] & 1 ? 0.0078125 : 0; // 1/128  retorno += rx[1] & 128 ? 0.00390625 : 0; // 1/256  retorno += rx[1] & 64 ? 0.001953125 : 0; // 1/512  retorno += rx[1] & 32 ? 0.0009765625 : 0; // 1/1024  retorno += rx[1] & 16 ? 0.00048828125 : 0; // 1/2048  retorno += rx[1] & 8 ? 0.000244140625 : 0; // 1/4096  retorno += rx[1] & 4 ? 0.0001220703125 : 0; // 1/8192  retorno += rx[1] & 2 ? 0.00006103515625 : 0; // 1/16384  retorno += rx[1] & 1 ? 0.00003051757813 : 0; // 1/32768  retorno += rx[2] & 128 ? 0.00001525878906 : 0; // 1/65536  retorno += rx[2] & 64 ? 0.000007629394531 : 0; // 1/131072  retorno += rx[2] & 32 ? 0.000003814697266 : 0; // 1/262144  retorno += rx[2] & 16 ? 0.000001907348633 : 0; // 1/524288  retorno += rx[2] & 8 ? 0.0000009536743164 : 0; // 1/1048576  retorno += rx[2] & 4 ? 0.0000004768371582 : 0; // 1/2097152  retorno += rx[2] & 2 ? 0.0000002384185791 : 0; // 1/4194304  retorno += rx[2] & 1 ? 0.0000001192092896 : 0; // 1/8388608  retorno = rx[0] & 128 ? (-1) \* retorno : retorno; // Verifica o sinal  return retorno;  } |

Imagens do processo de aferição da placa de medição de consumo (SONOMA):



Figura 46 - Verificação dos valores medidos pela Sonoma



Figura 47 - Verificação dos valores medidos pela Sonoma