

## ESTUDO E DESENVOLVIMENTO DE GATEWAYS E BEACONS BLE PARA RASTREAMENTO DE PESSOAS EM AMBIENTES FECHADOS

### STUDY AND DEVELOPMENT OF GATEWAYS AND BEACONS BLE FOR TRACKING PEOPLE IN CLOSED ENVIRONMENTS

Brenda Morel Bueno Ferreira<sup>1</sup>

Wilker Patrik Carlos Antunes<sup>2</sup>

**RESUMO:** O presente artigo buscou discorrer sobre o rastreamento de pessoas em ambientes fechados, se utilizando de dispositivos (microcontroladores e microprocessadores) portadores da tecnologia BLE (do inglês *Bluetooth low Energy*). Somado a isso é apresentado o conceito de técnica de trilateração para realizar o rastreamento de dispositivos a partir da intensidade do sinal *Bluetooth* recebido.

**Palavras-chave:** *Bluetooth* de baixa energia. Rastreamento. Microcontrolador. Gateway. Beacon.

**ABSTRACT:** The article presents people tracking devices indoors using devices (microcontrollers and microprocessors) designed from BLE (Bluetooth low Energy) technology. Added to this, the concept of trilateration technique is presented to perform device tracking based on the received Bluetooth signal strength.

1329

**Keywords:** Bluetooth low Energy. Tracking. Microcontroller. Gateway. Beacon.

## 1 INTRODUÇÃO

Com o desenvolvimento tecnológico aumentando cada vez mais, foi sendo necessário meios de fazer operações inteligentes, ou seja, interações que não precisam de um controlador (no caso, um ser humano dando comandos síncronos), ou ainda, interação objeto-humano que sejam mínimas. Com essa necessidade em vista, este amplo caminho no mercado em lacuna, foi desenvolvido o *IoT*. Com o desenvolvimento desta tecnologia é possível uma comunicação perfeita entre pessoas, processos e coisas, de modo que por meio de computação de baixo custo, nuvem, big data, análise e tecnologias móveis, as coisas físicas podem compartilhar e coletar dados com intervenção humana mínima. O mundo físico encontra o mundo digital, e eles cooperam.

<sup>1</sup> Graduando em Engenharia Elétrica, Universidade Estadual do Oeste do Paraná (Unioeste). Paraná, Brasil. [bmorelbueno@gmail.com](mailto:bmorelbueno@gmail.com)

<sup>2</sup> Graduando em Engenharia Elétrica, Universidade Estadual do Oeste do Paraná (Unioeste). Paraná, Brasil. [Wilker.pc12@hotmail.com](mailto:Wilker.pc12@hotmail.com).

Assim, visando a integração entre o mundo digital e físico, o projeto apresentado visa implementar ferramentas do *IoT* dentro do dia a dia das pessoas. Foi realizado o estudo do uso da tecnologia Bluetooth 4.0, e sua aplicabilidade através do protocolo de comunicação MQTT, e através de tal, foi desenvolvido um projeto de localização inteligente dentro de ambientes fechados.

O trabalho em questão conta com o estudo bibliográfico e a descrição detalhada da criação do projeto em questão, que consiste no desenvolvimento de um gateway através de um microcontrolador e um dispositivo beacon através de um microprocessador.

## 2. REVISÃO BIBLIOGRÁFICA

Nesta seção será realizada uma breve revisão da bibliografia disponível sobre o assunto tratado no artigo.

### 2.1 *IoT*– *Internet of Things*

*Se tivéssemos computadores que soubessem tudo sobre as coisas em geral, usando dados que coletassem sem a nossa ajuda, seríamos capazes de rastrear e contar tudo, e reduzir bastante o desperdício, a perda e os custos. Nós saberíamos quando é necessário substituir, reparar ou fazer um recall de um produto, e se estão novos ou ultrapassados. Precisamos capacitar os computadores com seus próprios meios de coletar informações, para que possam ver, ouvir e cheirar o mundo sozinhos, com toda a sua glória aleatória. O RFID e a tecnologia de sensores capacitam os computadores a observar, identificar e entender o mundo sem as limitações dos dados inseridos pelos humanos. (ASHTON; KEVIN, 1999).*

1330

Somado a isso, Kevin Ashton, em uma entrevista concedida à Revista Inovação em Pauta, construiu uma das inúmeras definições a qual pode ser atribuída ao termo *IoT*,

*Um ponto de encontro entre as redes de comunicações humanas (Internet) e o mundo real das coisas, onde não mais apenas "usaremos um computador", mas onde o "computador se use" independentemente, de modo a tornar a vida mais eficiente. Os objetos – as "coisas" – estarão conectados entre si e em rede, de modo inteligente, e passarão a "sentir" o mundo ao redor e a interagir. (FINEP, 2015, Apud FERRASI, 2017).*

Segundo HOLZ, CÁSSIA; TOSINI; GUEDES (2010), com o aprimoramento desta tecnologia ao longo dos anos, hoje pode-se notar algumas de suas funcionalidades no mercado, como acesso à tecnologia de sensores de baixo custo e baixa potência, maior conectividade entre protocolos de rede para a internet na transferência de dados, aumento na disponibilidade de plataformas na nuvem, *machine learning* e análise avançada, inteligência artificial (IA) e entre outras coisas. Em escala industrial, nota-se seu uso em manufatura inteligente, manutenção preventiva e preditiva, redes elétricas inteligentes, cadeias de suprimentos digitais e inteligentes.

## 2.2 O *Bluetooth low Energy* – BLE

Juntamente à internet das coisas, uma tecnologia relevante a ser abordada é a chamada BLE (*Bluetooth low Energy*), utilizada como uma das formas de conexão entre os dispositivos “inteligentes”. Para uma melhor compreensão, primeiramente é necessário ter conhecimento sobre a definição da tecnologia Bluetooth tradicional a qual as pessoas estão mais acostumadas a utilizarem no dia a dia. Segundo ALECRIM (2008), “Bluetooth é um padrão global de comunicação sem fio e de baixo consumo de energia que permite a transmissão de dados entre dispositivos, desde que um esteja próximo do outro. Uma combinação de hardware e software é utilizada para permitir que esse procedimento ocorra entre os mais variados tipos de aparelhos. A transmissão de dados é feita por meio de radiofrequência, permitindo que um dispositivo detecte o outro independente de suas posições, sendo necessário apenas que ambos estejam dentro do limite de proximidade (via de regra, quanto mais perto um do outro, melhor).”

Diretamente ligado a tecnologia bluetooth tradicional, existe o chamado BLE (*Bluetooth low Energy* ou *Bluetooth* de baixa Potência), que se trata de uma das vertentes do *Bluetooth*. De acordo com ARAUJO e VASCONCELLOS (2012), essa variação da tecnologia *Bluetooth* foi desenvolvida para ser mais eficiente em termos de consumo de energia elétrica, fazendo com que fosse possível alimentar dispositivos dotados desta tecnologia com baterias menores. Além disso, um dispositivo BLE tende a funcionar realizando conexões que duram milissegundos e permanecendo no modo *sleep* na maior parte do tempo, o que possibilita essa economia energética. “alguns beacons têm capacidade de transmitir sinal durante 2 anos ou mais com a mesma bateria. Este é um aspecto de enorme importância uma vez que grande parte dos *Beacons* não permitem a substituição da bateria devido às suas condições de concepção e requisitos suportados (por exemplo, resistentes à água).” (PAIVA, 2016).

Juntamente a isso, dispositivos *Bluetooth* de Baixa Potência carregam uma série de vantagens em seu uso além da economia de energia elétrica. MELO (2016) cita algumas dessas vantagens como por exemplo, a capacidade de realizar múltiplas conexões e enviar pequenas mensagens (com velocidade máxima de 1 Mbps) para vários dispositivos ao mesmo tempo, envio de informações de sensores dos dispositivos e baixo custo de aquisição.

Em relação as características de funcionamento deste tipo de tecnologia, tem-se que, “O BLE habilita a transmissão de dados em dois modos, o primeiro ocorre com os dispositivos conectados e o segundo é o modo *advertising* (*broadcast*). O modo conectado utiliza a camada *Generic Attribute* (GATT). Um importante exemplo de aplicação deste modo ocorre nos

aplicativos de configuração dos *Beacons*, onde o usuário pode selecionar os parâmetros mais relevantes para a sua aplicação. Já o modo *advertising* utiliza a camada *Generic Access Profile* (GAP), transmitindo dados de um-para-muitos. Este é o modo padrão de operação dos *Beacons* para transmissão de dados” CIPRIANO (2018).

O modo de funcionamento mais importante a ser tratado é o *advertising* que envia pequenos pacotes de dados em intervalos de 20 ms a 10 segundos, sendo que quanto menor o intervalo, menor será a duração da bateria, porém um dispositivo poderá ser descoberto mais rapidamente. (PAIVA, 2016, p.15).

O tamanho do pacote de mensagem *BLE* pode variar de 8 a 47 *bytes*, sendo que o mesmo é dividido em atributos.

**Preamble:** Utilizado no gerenciamento dos protocolos internos; **Access Address:** Para as mensagens em *broadcast*, sempre seguirá o padrão *0x8E89BED6*; **PDU:** Este é dividido em *PDU header* e *PDU* carga útil, sendo que o primeiro trata-se das informações sobre o tipo e tamanho dos dados que serão enviados. Já o *PDU* carga útil carrega as informações desejadas; **CRC:** Calculado a partir do *PDU header* mais *PDU* carga útil. (MELO, 2016). Somado a isso, o *PDU* carga útil ainda traz o **MACAddress** (Endereço do Controle de Acesso de Mídia), que carrega o identificador único do dispositivo emissor. Assim, restam 31 *bytes* para os dados gerais a serem utilizados.

### 2.3 O protocolo MQTT

O MQTT é um protocolo de comunicação extremamente leve e rápido, desenvolvido originalmente pela empresa IBM e é utilizado principalmente na comunicação *M2M* (*Machine-to-Machine*) (ANDRADE et al. apud LAMPKIN et al., 2012). Ele apresenta diversos benefícios para dispositivos com limitações de recursos como a troca de mensagem assíncrona, sem a dependência de resposta ou mensagens de confirmação, tamanho reduzido das mensagens trocadas, baixo uso dos recursos de processamento e memória, comunicação otimizada para sensores e dispositivos remotos e etc.

O protocolo MQTT implementa a arquitetura *publish/subscriber* na sua comunicação, com um servidor central conhecido como *broker* (intermediário). De acordo com a arquitetura *publish/subscriber*, uma comunicação através do protocolo MQTT funciona da seguinte maneira, o *publisher* publica as informações para o *broker* (servidor) através de um tópico, e assim os *subscribers* inscritos nesse mesmo tópico recebem essas informações do *broker*. Tanto *publishers*

como *subscribers* são clientes na arquitetura implementada pelo MQTT, e conectam-se a apenas um *broker*. Porém, um mesmo cliente MQTT pode ser *subscriber* e *publisher* de diversos tópicos. (MELO et al apud LAMPKIN et al., 2012).

## 2.4 O dispositivo Beacon

O Beacon é um transmissor que sabe a localização de dispositivos através de uma conexão por um sinal periódico que contém uma certa quantidade de informações, limitada por sua capacidade. Essa conexão se dá por meio da tecnologia de rádio, que é usada para identificar a localização, ela é usada em tecnologias como *Wi-Fi*, *Bluetooth* e *RFID*. Neste caso, o Beacon utiliza-se da conexão *Bluetooth*.

O Beacon indica sua presença com um sinal periódico, de modo que um dispositivo capaz de o identificar pode localizá-lo, esses dispositivos podem ser celulares, relógios inteligentes, computadores de modo geral, aparelhos inteligentes que usam o BLE 4.0, etc. Para cada Beacon é dado um identificador único, quando um dispositivo móvel identifica o seu sinal, ele lê exatamente esse identificador e calcula a distância em que o mesmo se encontra, baseado em sua data (que é previamente configurada, podendo ser *Eddystone* ou *iBeacon*, de acordo com a fabricação do dispositivo). Se o aparelho estiver com o *Bluetooth* ativado e for compatível com a versão requerida de tal, ele entra em ação e pode ter informações do aplicativo do beacon, podendo conter desde informações básicas, como apenas uma URL, até um anúncio. A complexidade da informação que é passada para o dispositivo móvel depende da configuração feita previamente no Beacon utilizado.

1333

## 3 MATERIAIS E MÉTODOS

Esta seção tem como finalidade apresentar os materiais utilizados e suas características, somado a apresentação dos métodos que fez-se uso.

### 3.1 Raspberry pi

De modo sucinto, o *Raspberry Pi* trata-se de um poderoso minicomputador com hardware em uma única placa de tamanho reduzido. Esse dispositivo pode ser programado em algumas diferentes linguagens, como C e Python.

No mercado existem os modelos A e B, sendo que o foco neste artigo será no modelo B que foi o utilizado. Este possui quatro portas *USB*, uma porta *ethernet*, 1GB de memória *RAM*, *BCM43438 wireless LAN* e *Bluetooth Low Energy (BLE) on board*.

Como sistema operacional, o *Raspberry Pi* utiliza uma distribuição *Linux* oficial para este dispositivo, o chamado *Raspbian*.

### 3.2 Esp-32

Fabricado pela *Espressif Systems*, o *Esp-32* é um microcontrolador de alta performance, consumo de energia consideravelmente baixo, conexão *WiFi*, e módulo *Bluetooth* integrado. Especificamente, este dispositivo possui 4MB de memória *flash*, *CPU dual-core*, 448 Kbytes de memória *ROM* e 520 Kbytes de memória *RAM*.

Sua programação pode ser facilmente feita através da *IDE* do *Arduino* e o código pode carregado para a placa a partir de um cabo *micro-USB*.

### 3.3 Beacon H3

De maneira sucinta, o *Beacon H3* é um dispositivo produzido pela *MOKOSMART*. Este tipo de dispositivo possui alcance de até 150 metros, dependendo do ambiente e das interferências a que é submetido. Possui uma faixa de potência de transmissão de -40 dBm a +4dBm.

### 3.4 Método de Trilateração

A trilateração trata-se de uma técnica que se utiliza de valores de intensidade de sinal recebidos por 3 pontos distintos. A partir da recepção dessas informações necessárias, o método permite estimar a posição de um determinado dispositivo dentro ou fora de um ambiente (*indoor* ou *outdoor*).

Primeiramente, é necessário determinar o  $RSSI@1m$ , que é o valor do sinal recebido a 1 metro entre o emissor e o receptor. Após isso, estima-se através de testes práticos o coeficiente de perda no percurso  $n$ , através da equação 1:

$$n = - \frac{RSSI - RSSI@1m}{10 * \text{Log}(d)} \quad (\text{Equação 1})$$

Onde  $RSSI$  é a intensidade do sinal à uma certa distância  $d$ .

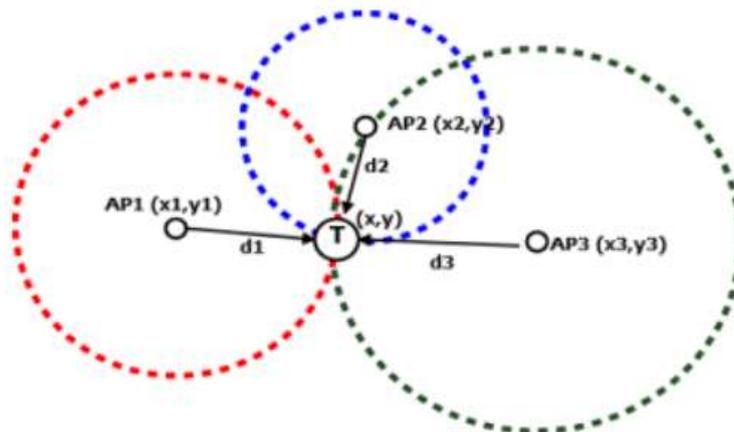
Sendo assim, é possível determinar a posição de um dispositivo BLE através das equações 2 e 3:

$$x = \frac{(d_1^2 - d_2^2 + p^2)}{2 * p} \quad \text{(Equação 2)}$$

$$y = \frac{(d_1^2 - d_3^2 + q^2 + r^2)}{2 * r} - x * \frac{q}{r} \quad \text{(Equação 3)}$$

Em que,  $p$ ,  $q$  e  $r$  representam pontos em que os gateways estão posicionados, sendo eles  $P_1$  (0, 0),  $P_2$  (p, 0), e  $P_3$  (q, r). O esquema do método de trilateração pode ser observado na figura 1 a seguir.

Figura 1 - Esquema do método de Trilateração



Fonte: Adaptado de (COUTINHO; LIMA; MENDES, 2021)

## 4 Resultados e discussões

Aqui serão descritos e comentados todos os resultados obtidos, baseando-se na teoria abordada anteriormente.

### 4.1 Utilização do Esp-32 para criação de um Beacon

O código foi desenvolvido através da IDE do Arduino e transferido ao dispositivo através de conexão USB.

O objetivo do programa foi utilizar a tecnologia BLE existente no microcontrolador, e fazê-lo se comunicar com algum outro dispositivo que tivesse compatibilidade para tal. Uma vez inicializado e funcionando, o ESP recebeu o comando de acender o LED azul presente no

mesmo, e ainda, mostrar uma mensagem no *serial* do *ARDUINO.IDE*, indicando que o código está funcionando.

#### 4.1.1 Testes de alcance com o Beacon desenvolvido

A priori, algumas considerações precisaram ser feitas, sendo elas:

- Nome atribuído ao *Beacon*: beacon-bre (definido no código desenvolvido);
- Aplicativo utilizado para o teste: *Nfr Connect* (disponível para dispositivos *Android* na *Playstore*);
- *Tx Power Level*: 3dBm;
- Ambiente: Sala fechada com interferências externas irrelevantes

A **tabela 1** a seguir mostra os dados provenientes da medição realizada.

**Tabela 1** - Dados obtidos a partir da medição realizada

Distância (metros)	mS	dBm	Slave Connection Interval Range
0	45	-41	40.00ms - 80.00ms
0.25	44	-50	40.00ms - 80.00ms
0.5	43	-56	40.00ms - 80.00ms
0.75	43	-57	40.00ms - 80.00ms
1.00	43	-61	40.00ms - 80.00ms
1.25	43	-65	40.00ms - 80.00ms
1.50	44	-69	40.00ms - 80.00ms
1.75	43	-75	40.00ms - 80.00ms
2.00	45	-77	40.00ms - 80.00ms
2.25	43	-73	40.00ms - 80.00ms
2.50	43	-79	40.00ms - 80.00ms
2.75	45	-80	40.00ms - 80.00ms
3.00	--	--	-----

**Fonte:** Elaborado pelo autor

Analisando os valores obtidos é possível observar que 3 metros foi a distância máxima em que o aplicativo utilizado no celular conseguiu reconhecer o *Beacon*. Ressaltando que a distância máxima em que o dispositivo *BLE 4.0* funciona em questão está relacionada ao dispositivo usado para se criar um *Beacon*, que no caso foi o *ESP-32*, assim como o celular e o aplicativo em questão. Para a criação do dispositivo foi usado apenas o microprocessador, sem nenhum outro dispositivo que ajuda a aumentar o alcance *BLE*, sendo a exatidão de 3 metros

máximos para a criação de um *Beacon* com o ESP-32 não sendo totalmente exata, podendo variar de acordo com as condições do ambiente e do dispositivo.

## 4.2 Utilização do *Raspberry pi* para criação do *Gateway*

Nesta seção serão descritos os processos realizados para a criação de um *Gateway* a partir de um *Raspberry pi*. Vale ressaltar que aqui os testes foram realizados utilizando o *Beacon H3*.

### 4.2.1 Captação e publicação de valores de *RSSI* no tópico *MQTT*

Primeiramente, foi definido que o algoritmo seria desenvolvido na linguagem Python, já que grande parte da bibliografia envolvendo o assunto gira em torno dessa linguagem de programação.

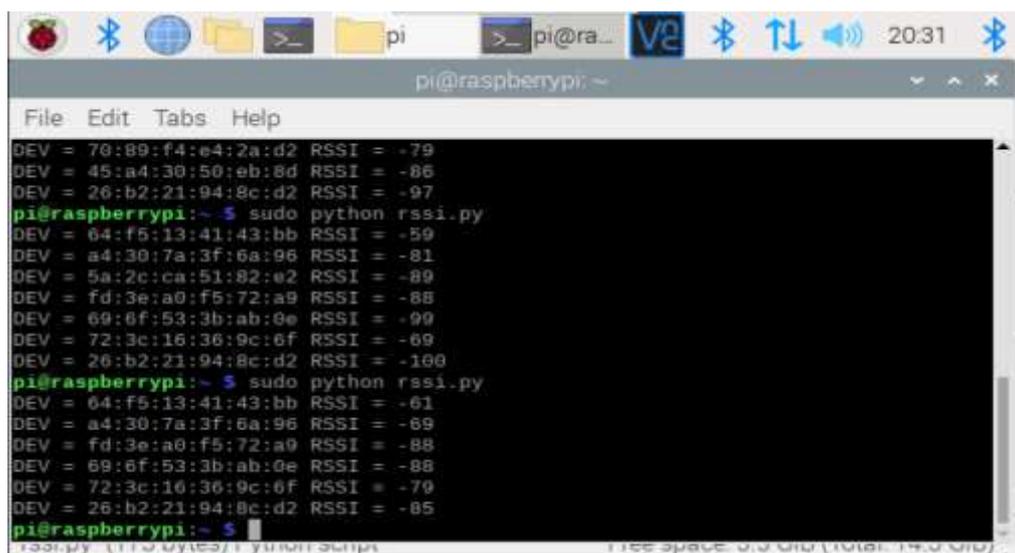
Para facilitar a maneira de se obter os dados de *RSSI*, optou-se por utilizar a biblioteca *bluepy*, desenvolvida com foco em projetos que envolvam dispositivos *BLE*. O *download* da biblioteca se deu através dos seguintes comandos no *prompt* de comandos do *Raspberry*:

```
sudo apt-get install python-pip libglib2.0-dev
sudo pip install bluepy
```

Após isso foi possível desenvolver uma rotina capaz de escanear, identificar e mostrar o endereço *MAC* e o valor de *RSSI* de todos os dispositivos *BLE*'s presentes nas proximidades. É possível conferir os dados obtidos a partir da **figura 2**.

1337

**Figura 2** - Endereço *Mac* e Valores *RSSI*



```
pi@raspberrypi: ~
File Edit Tabs Help
DEV = 70:89:f4:e4:2a:d2 RSSI = -79
DEV = 45:a4:30:50:eb:8d RSSI = -86
DEV = 26:b2:21:94:8c:d2 RSSI = -97
pi@raspberrypi: ~ $ sudo python rssi.py
DEV = 64:f5:13:41:43:bb RSSI = -59
DEV = a4:30:7a:3f:6a:96 RSSI = -81
DEV = 5a:2c:ca:51:82:e2 RSSI = -89
DEV = fd:3e:a0:f5:72:a9 RSSI = -88
DEV = 69:6f:53:3b:ab:0e RSSI = -99
DEV = 72:3c:16:36:9c:6f RSSI = -69
DEV = 26:b2:21:94:8c:d2 RSSI = -100
pi@raspberrypi: ~ $ sudo python rssi.py
DEV = 64:f5:13:41:43:bb RSSI = -61
DEV = a4:30:7a:3f:6a:96 RSSI = -69
DEV = fd:3e:a0:f5:72:a9 RSSI = -88
DEV = 69:6f:53:3b:ab:0e RSSI = -88
DEV = 72:3c:16:36:9c:6f RSSI = -79
DEV = 26:b2:21:94:8c:d2 RSSI = -85
pi@raspberrypi: ~ $
```

Fonte: Elaborado pelo autor

Somado a isso, havia a necessidade de que essas informações fossem publicadas em um determinado tópico *MQTT* para serem posteriormente utilizadas nos cálculos de posição do

*Beacon*. Sendo assim, foi necessária a realização do download do *mosquitto clients*, a fim de utilizar o *Raspberry* como um *Publisher MQTT*.

Para se trabalhar com esse protocolo, em linguagem *python* existe uma biblioteca que facilita no processo. Seu *download* foi realizado a partir do comando.

```
pip install paho-mqtt
```

A quantidade de dispositivos *BLE*'s presentes nas proximidades poderia ser um problema no momento de obtenção e visualização dos dados, sendo assim foi definido um *Beacon* específico no código desenvolvido para que só fossem publicados os valores provenientes dele.

É válido ressaltar que o código foi aplicado em 3 *Raspberrys*, já que precisa-se dessa quantidade de gateways posicionados em pontos distintos e coletando os dados necessários para a aplicação do método de trilateração. A mudança do código de um *gateway* para outro se deu apenas no tópico em que cada *publisher* publica suas informações coletadas, sendo eles: **Dispositivos/*rssi/gateway1* ; Dispositivos/*rssi/gateway2* ; Dispositivos/*rssi/gateway3*.**

#### 4.2.2 Determinação dos parâmetros de cálculo

Para o cálculo da posição de um beacon a partir do método de trilateração que utiliza os valores de *RSSI*, é necessário que se determine o valor médio de *RSSI* a uma distância de 1 metro e também o coeficiente de perda de percurso *n*. Esses coeficientes foram definidos através de testes práticos feitos num ambiente de 15x3 metros, composto por móveis e paredes que implicam em uma interferência do sinal.

Primeiramente, realizou-se medições para se obter o valor médio de *RSSI* a uma distância de 1 metro entre o *gateway* e o *beacon*. Os dados obtidos podem ser conferidos na tabela 2.

**Tabela 2** - Dados para obtenção do *RSSI@1m*

Ambiente livre	Interferencia de parede de alvenaria	Interferencia de porta de madeira
-52	-56	-56
-52	-62	-60
-52	-55	-59
-52	-61	-60
-49	-61	-56
-59	-53	-59
-59	-55	-56
-49	-53	-58
-52	-53	-58
-49	-53	-57
-59	-60	-56
-49	-61	-58
-52	-55	-56
-59	-53	-58
-59	-61	-56
-53,53333333	-56,8	-57,53333333
	<b>RSSI@1m</b>	
	<b>-55,95555556</b>	

**Fonte:** Elaborado pelo autor

Após isso foi definido o coeficiente de perda de percurso através da equação 1.

O cálculo foi realizado variando-se a distância de 1 a 20 metros com intervalos de 1 metro.

O valor obtido para o parâmetro  $n$  foi de **2,958290**.

#### 4.2.3 Elaboração de algoritmo para cálculo da posição de um Beacon

A priori, é importante frisar que o código foi desenvolvido utilizando um microcontrolador *ESP-32* juntamente com a IDE do Arduino.

Nessa parte do projeto utilizou-se o *ESP-32* atuando como um *Subscriber MQTT*, ou seja, seu papel foi se inscrever nos tópicos definidos anteriormente e receber os dados captados pelos tres gateways utilizados, e assim realizar o cálculo da posição em que o beacon se encontrava dentro de determinado ambiente.

Somado a isso, fez-se o uso dos valores em metros das posições em que cada gateway foi posicionado, juntamente com os valores dos coeficientes ja definidos na seção anterior. As posições de cada gateway foram:

*Gateway 1: (0, 0) metros*  
*Gateway 2: (4,5, 3) metros*  
*Gateway 3: (11,5, 0) metros*

Vale salientar que a altura em que cada dispositivo estava era de *1,20 metros*.

#### 4.2.4 Testes de posição

valores para cada ponto em que o *Beacon* estivesse. Assim fez-se a média e determinou-se o ponto estimado pelo algoritmo.

O primeiro teste foi realizado com o beacon na posição *P1(5 , 0) metros*. Valores se encontram na **tabela 3**.

**Tabela 3** - Valores de posição em metros –  $P1(5 , 0)$

x	y
5,98	-6,34
5,98	-5,24
6,07	-9,7
5,98	-4,61
5,57	-5,12
5,8	-5,67
5,91	-18,06
7,49	-1,71
6,55	-1,89
6,79	-2,89
6,96	-2,95
6,07	-4,6
6,55	-2,33
5,57	-5,12
5,57	-5,72
<b>5,392</b>	<b>-5,463333333</b>

**Fonte:** Elaborado pelo autor

Para a segunda obtenção de dados, o beacon foi colocado em  $P_2(4.5, 1)$  metros. A tabela 4 a seguir contém os dados coletados.

**Tabela 4:** Valores de posição em metros –  $P_2(4.5, 1)$

x	y
4,69	-6,92
-1,11	1,78
4,13	-8,63
4,05	-8,81
4,04	-8,83
4,05	-8,81
4,13	-8,63
4,37	-6,15
4,69	-11,06
5,86	-12,81
5,78	-12,69
-2,17	3,67
4,69	-11,06
4,51	-2,37
4,37	-10,29
<b>3,738667</b>	<b>-7,44067</b>

Fonte: Elaborado pelo autor

O terceiro teste foi realizado em  $P_3(11.5, 2.5)$  metros. É possível conferir os valores a partir da tabela 5.

**Tabela 5:** Valores de posição em metros –  $P_3(11.5, 2.5)$

x	y
12,55	-10,36
11,57	1,01
12,47	-1,71
14,94	-8,05
13,73	-11,34
15,07	8,9
8,78	-15,46
15,07	3,89
12,55	-1,83
12,47	-13,61
8,78	-1,41
13,61	-20,65
8,78	-22,74
13,73	-7,58
12,59	-14,01
<b>12,446</b>	<b>-7,66333</b>

Fonte: Elaborado pelo autor

## CONCLUSÃO

A partir da análise dos dados obtidos e dos estudos realizados é possível concluir que a tecnologia *Bluetooth low Energy* aliada ao uso de microcontroladores pode ser uma poderosa ferramenta quando se trata de localização de pessoas. Com a análise dos adquiridos, é possível

concluir que os pontos de localização, fornecidos pela técnica de trilateração aplicada no microcontrolador, não conseguiram atingir a exatidão. Isso pode se dar pelo fato de haver interferências consideráveis no ambiente, fazendo com que diminuísse a precisão da localização dos *beacons*.

## REFERÊNCIAS

- [1] ALECRIM, Emerson. Tecnologia Bluetooth: O que é e como funciona?. **InfoWester**. São Paulo, 30 jan. 2008. Disponível em: <<https://www.infowester.com/sobre.php>>. Acesso em: 24 nov. 2020.
- [2] ARAUJO, A. S; VASCONCELLOS, P. **Bluetooth Low Energy**. Rio de Janeiro, [s.d].
- [3] CHEGNG, Rung-Shiang. HONG, Wei-Jun. WANG, Jheng-Syun, LIN, W Kawuu. **Seamless Guidance System Combining GPS, BLE Beacon, and NFC Technologies**. Hindawi Publishing Corporation Mobile Information Systems. Volume 2016, Article ID 5032365, pg 12.
- [4] CIPRIANO, Wagner. **Localização Indoor Baseada em Tecnologia Bluetooth de Baixa Energia**. Dissertação (Mestrado em Modelagem Matemática e Computacional) – Centro de Educação Tecnológica de Minas Gerais. Belo Horizonte, 80 f. 2018.
- [5] DE MELO, Emmanuel CA et al. **Uma arquitetura de Beacons customizáveis para Internet das Coisas**. In: Anais do IX Simpósio Brasileiro de Computação Ubíqua e Pervasiva. SBC, 2017.
- [6] DE SOUSA OLIVEIRA, Ana Paula; CARVALHO, Sérgio T. **Localização Usando Beacons em Ambientes Internos para Monitoramento Remoto de Pacientes**.
- [7] DO PRADO, Milena Sant’Ana et al. **Beacons e a Ponte UX: controle da informação em instituições de ensino**. **Revista Multiplicidade**. v. 7, n. 7, 2016.
- [8] ESTE, Willian Reginato. **Internet das coisas: Tecnologias Habilitadoras e Workflow**. Dissertação (Mestrado em Ciências. Área de Concentração: Sistemas de Potência) - Escola Politécnica da Universidade de São Paulo. São Paulo, 102 f. 2019.
- [9] FERRASI, Faberson Augusto. **O Uso de Mídias Locativas no Universo da Internet das Coisas: construindo uma prova de conceito**. Dissertação (Mestrado em Mídia e Tecnologia) - Faculdade de Arquitetura, Artes e Comunicação, Universidade Estadual Paulista “Júlio de Mesquita Filho”. Bauru, 53 f. 2017.
- [10] HOLZ, Franciane Cássia; TOSINI, Juliana; GUEDES, Anibal Lopes. **Informática na Educação-O Emprego da Tecnologia Bluetooth e Rôbo Lego Mindstorms no Aprendizado de Crianças**. Anais SULCOMP, v. 5, 2010.
- [11] JEON, Eun, Kang. SH, James. SOONSAWAD, Perm. NG, Chet, Pai. **BLE Beacons for Internet of Things Applications: Survey, Challenges and Opportunities**. IEEE INTERNET OF THINGS JOURNAL, VOL. A, NO. B, 2018.

- [12] MEKKO, Kais. BAJIC, Eddy. MEYER, Fernand. **Indoor Positioning System for IoT Device based on BLE Technology and MQTT Protocol**. Research Centre for Automatic Control of Nancy, Campus Sciences, BP 70239, Vandoeuvre, 54506, France.
- [13] NEWMAN, Nic. **Opinion Piece Apple iBeacon technology briefing**. Journal of Direct, Data and Digital Marketing Practice. VOL. 15 NO. 3 PP 222-225.
- [14] REBONATTO, Marcelo Trindade; ECKSTEIN, Cássio; REBONATTO, Cintia Sonale. **Utilização de beacons em locais externos: Um estudo em um campus universitário no sul do Brasil**. In: **Anais do XII Simpósio Brasileiro de Computação Ubíqua e Pervasiva**. SBC, 2020. p. 131-140.
- [15] TORRES, Claudines Taveira. **O Uso de Tecnologia de Dispositivos Móveis para Desenvolvimento de Protótipo de App para Acessar Informações dos Objetos do Museu dos Ferroviários de Bauru**. Dissertação (Mestrado em Mídia e Tecnologia) - Faculdade de Arquitetura, Artes e Comunicação, Universidade Estadual Paulista. Bauru, 124 f. 2017.