

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO DE ESPECIALIZAÇÃO EM INTERNET DAS COISAS

LUIZ ALBERTO BASSAN JUNIOR

**IDENTIFICAÇÃO DE CONDUTOR VEICULAR POR BLUETOOTH
LOW ENERGY**

MONOGRAFIA DE ESPECIALIZAÇÃO

CURITIBA
2019

LUIZ ALBERTO BASSAN JUNIOR

IDENTIFICAÇÃO DE CONDUTOR VEICULAR POR BLUETOOTH LOW ENERGY

Monografia de Especialização, apresentada ao Curso de Especialização em Internet das Coisas, do Departamento Acadêmico de Eletrônica – DAELN, da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Especialista.

Orientador: Prof. Dr. Guilherme Luiz Moritz

CURITIBA
2019



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Curitiba

Diretoria de Pesquisa e Pós-Graduação
Departamento Acadêmico de Eletrônica
Curso de Especialização em Internet das Coisas



TERMO DE APROVAÇÃO

IDENTIFICAÇÃO DE CONDUTOR VEICULAR POR BLUETOOTH LOW ENERGY

por

LUIZ ALBERTO BASSAN JUNIOR

Esta monografia foi apresentada em 25 de Novembro de 2019 como requisito parcial para a obtenção do título de Especialista em Internet das Coisas. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. Dr. Guilherme Luiz Moritz
Orientador

Prof. M. Sc. Danillo Leal Belmonte
Membro titular

Prof. M. Sc. Omero Francisco Bertol
Membro titular

- O Termo de Aprovação assinado encontra-se na Coordenação do Curso -

AGRADECIMENTOS

Agradeço à minha família, em especial a minha esposa Telma, meus pais e minha irmã por todo o apoio incondicional prestado.

Agradeço à empresa Mobi7 pela oportunidade de trabalhar com tecnologias de ponta e em especial ao engenheiro responsável Gustavo Bergamo, que sempre me apoiou nos assuntos técnicos e dificuldades encontradas.

Por último, agradeço ao prof. Orientador Guilherme Moritz pela atenção e orientação para concluir essa monografia.

RESUMO

BASSAN JUNIOR, Luiz Alberto. **Identificação de condutor veicular por Bluetooth Low Energy**. 2019. 34 p. Monografia de Especialização em Internet das Coisas, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

Na indústria de rastreamento veicular, tem-se como um problema típico identificar qual condutor conduz qual veículo. Esse problema hoje em dia é resolvido por tecnologias da década de 90, principalmente o *iButton* com *One-Wire* que apresenta problemas de segurança e desgaste. Essa monografia propõe resolver esse problema utilizando tecnologia sem fio e que não necessita intervenção do condutor. Nesse cenário, faz-se uso de uma topologia de comunicação usando protocolo *Bluetooth Low Energy* baseando-se em beacons e receptores, discutindo-se os elementos dessa topologia, suas restrições e especificidades.

Palavras-chave: Internet das Coisas. Bluetooth Low Energy. Rastreamento veicular. *iButton*. Beacon.

ABSTRACT

BASSAN JUNIOR, Luiz Alberto. **Driver identification through Bluetooth Low Energy**. 2019. 34 p. Monografia de Especialização em Internet das Coisas, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

There is a typical problem in the vehicle tracking business which is to relate which driver has driven which vehicle. This problem is mostly solved using 90s technologies, mainly One-Wire protocol based iButton solution that has issues with security and wear. This work proposes to solve this problem using wireless technology that does not require driver interaction. The approach used is based on Bluetooth Low Energy beacons and receivers, whose restrictions and specificities will be explored in detail in this work.

Keywords: Internet of Things. Bluetooth Low Energy. Car Tracking. iButton. Beacon.

LISTA DE FIGURAS

Figura 1 – Motorista efetuando identificação por iButton	11
Figura 2 – Espectro de frequência BLE com ênfase para os canais de advertising..	13
Figura 3 – Camadas do protocolo Bluetooth Low Energy	14
Figura 4 – Detecção do condutor usando RSSI	16
Figura 5 – Diagrama em blocos da solução completa.....	17
Figura 6 – Pacote base padrão Eddystone	18
Figura 7 – Tipo de frame Eddystone UID	18
Figura 8 – Screenshot do aplicativo Beacon Simulator	19
Figura 9 – Placa customizada conectada ao debugger.....	20
Figura 10 – Arquitetura de software do BLE BOX Observer	21
Figura 11 – Processamento de pacotes.....	22
Figura 12 – Aplicativo de recebimento de pacotes UDP em aplicação C.....	28
Figura 13 – Pacotes na fila AWS SQS	28
Figura 14 – Exemplo de pacote UDP com evento de BEACON_DISAPPEAR	29

LISTA DE GRÁFICOS

Gráfico 1 – Principais custos de uma operação de transporte	9
Gráfico 2 – Taxa de recebimento de pacotes em função dos cenários de teste	25
Gráfico 3 – Parâmetro RSSI no cenário 1 (eixo y – valor em dBm RSSI, eixo x amostras capturadas no receptor).....	26
Gráfico 4 – Parâmetro RSSI no cenário 5 (eixo y – valor em dBm RSSI, eixo x amostras capturadas no receptor).....	26

LISTA DE QUADROS

Quadro 1 – Eventos de identificação de condutor.....	23
Quadro 2 – Cenários para levantamento de dados de RSSI e perda de pacotes	24

LISTA DE SIGLAS

BLE	<i>Bluetooth Low Energy</i>
GAP	<i>General Access Profile</i>
GATT	<i>Generic Attributes Generic Attribute Profile</i>
GPRS	<i>General Packet Radio Services</i>
GPS	<i>Global Positioning System</i>
IP	<i>Internet Protocol</i>
PKES	<i>Passive Keyless Entry System</i>
RSSI	<i>Received Signal Strength Indicator</i>
UDP	<i>User Datagram Protocol</i>

SUMÁRIO

1 INTRODUÇÃO	9
1.1 PROBLEMA	10
1.2 OBJETIVOS	11
1.2.1 Objetivo Geral	11
1.2.2 Objetivos Específicos	11
1.3 ESTRUTURA DO TRABALHO	12
2 FUNDAMENTAÇÃO TEÓRICA	13
2.1 BLUETOOTH LOW ENERGY (BLE)	13
2.2 RASTREAMENTO VEICULAR.....	15
2.2.1 Global Positioning System (GPS).....	15
2.2.2 GPRS / 2G	15
2.3 PASSIVE KEYLESS ENTRY SYSTEM (PKES)	15
3 DESENVOLVIMENTO	17
3.1 BEACON BLE	17
3.1.1 Protocolo de Transmissão.....	18
3.1.2 Implementação do Beacon.....	19
3.2 RECEPTOR BLE.....	19
3.2.1 BLE Box Observer.....	20
3.2.2 Rastreador.....	21
3.3 PROCESSAMENTO DE PACOTES	22
3.3.1 Infraestrutura	22
3.3.2 Protocolo de Identificação do Condutor	23
3.3.3 Aplicação Parser	23
4 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS	24
4.1 PERDA DE PACOTES	24
4.2 ANÁLISE DO COMPORTAMENTO PARÂMETRO RSSI	25
4.3 ANÁLISE DOS PARÂMETROS DO BEACON	27
4.4 RECEBIMENTO DE PACOTES UDP.....	27
5 CONCLUSÃO	30
REFERÊNCIAS.....	32

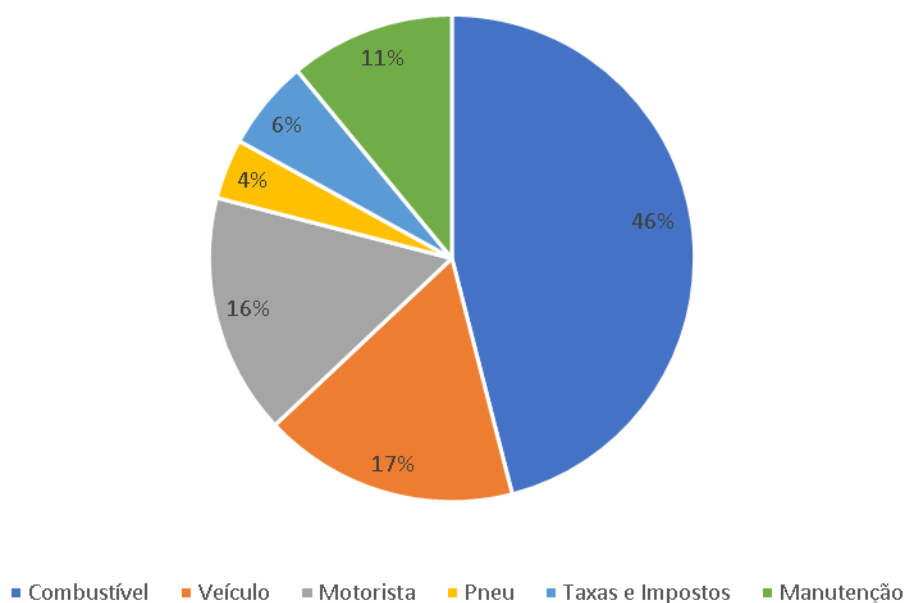
1 INTRODUÇÃO

Um dos principais pontos para o desenvolvimento econômico de qualquer nação é o desenvolvimento de uma rede de infraestrutura que permita, com o mínimo custo, o deslocamento de bens e pessoas entre quaisquer pontos. Em um país com dimensões continentais como o Brasil, esse quesito se mostra ainda mais relevante.

Segundo apurou Maia (2016), informações do Plano de Transporte e Logística da Confederação Nacional do Transporte (CNT) apontam que, no Brasil, os custos logísticos representaram 11,6% do Produto Interno Bruto (PIB). Em comparação, nos Estados Unidos, a porcentagem é de 8,7%. Além disso, segundo dados CNT, o modal rodoviário é usado em 61% da carga transportada nacionalmente (CNT, 2017).

Além da relevância macroeconômica acima explicitada, uma pesquisa realizada pela montadora Scania publicada no portal Transporte Econômico, apresentada no Gráfico 1, levantou os principais custos de uma operação de transporte (TRANSPORTE_ECONÔMICO, 2018).

Gráfico 1 – Principais custos de uma operação de transporte



Fonte: Transporte_Econômico (2018).

Ainda o Gráfico 1 mostra que os custos de manutenção e combustível somam cerca de 57% dos custos de uma operação de transporte. Tais custos se relacionam intimamente aos hábitos de condução do motorista. Caso um condutor dirija o veículo em velocidades altas incompatíveis com as vias e com altas rotações do motor, fatalmente consumirá mais combustível e tende a acelerar o desgaste de peças dos veículos, aumentando os custos de operação e reduzindo a margem dos players do segmento. Dessa forma, a efetiva gestão sobre os condutores e seus hábitos de condução nos veículos é fator muito relevante para a saúde de operação de uma empresa do segmento logístico.

1.1 PROBLEMA

Tendo-se em vista o contexto de mobilidade, diversas companhias ofertam soluções para gestão de frotas corporativas a partir de dados coletados de veículos leves ou pesados por meio de rastreadores instalados. Dentro desse contexto, além da necessidade de localização do ativo, faz-se relevante identificar o condutor desses veículos para propósitos de gestão, que oscilam desde identificação de hábitos potencialmente perigosos de condução passando por controle de jornada e escalas de motoristas.

No que se refere às tecnologias de identificação, merece destaque o iButton. O iButton é um dispositivo metálico cujo desenvolvimento remete há cerca de 30 anos, reportando seu identificador único no protocolo *One-Wire* de fácil implementação (YOUNG, 2018). Destaca-se como pontos positivos sua robustez mecânica e o fato de dispensar a utilização de bateria.

As soluções atuais para identificação do motorista funcionam principalmente baseadas em um chaveiro com iButton, como mostra a Figura 1, que é fornecido ao motorista. Para se identificar, o motorista deve encostar o seu iButton em um leitor de protocolo *One-Wire* acoplado ao rastreador e iniciar sua direção.

Figura 1 – Motorista efetuando identificação por iButton



Fonte: Fassa (2015).

Além de depender da ação do motorista para se identificar (não há identificação automática), o leitor de iButton deve ficar no painel do veículo. Em uma situação de roubo isso sinaliza que o carro está provido de rastreador, o que pode acarretar complicações à segurança física do condutor ou ainda aumentar a chance de remoção do rastreador e com isso, diminui as chances de recuperação do ativo em caso de sinistro. Outro fator que se mostra um limitante atual à adoção do iButton é sua dificuldade de integração a ambientes mais digitalizados como smartphones, uma vez que é uma solução da década de 80/90.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Desenvolver método de identificação de motorista utilizando *Bluetooth Low Energy* integrado a um ambiente de rastreamento veicular.

1.2.2 Objetivos Específicos

Etapas a serem realizadas para atingir-se o objetivo geral:

- Desenvolver beacon *Bluetooth Low Energy* (BLE) que será usado pelo motorista;

- Desenvolver método de detecção de proximidade de beacons BLE;
- Integrar hardware de detecção de beacons com rastreador veicular;
- Transmitir dados de beacons BLE por meio de pacotes UDP.

1.3 ESTRUTURA DO TRABALHO

Esta monografia de especialização está estruturada em 5 sessões, as quais são detalhadas a seguir:

- Capítulo 1: Apresenta a introdução ao trabalho, delimitando o escopo, objetivos e justificativa.
- Capítulo 2: Apresenta a fundamentação teórica que sustenta as tecnologias empregadas no trabalho, como por exemplo o Bluetooth Low Energy.
- Capítulo 3: Aspectos relacionados ao desenvolvimento da solução e aos blocos confeccionados para resolução do problema proposto.
- Capítulo 4: Apresentação de resultados e discussão dos dados.
- Capítulo 5: Conclusão do trabalho e discussão de trabalhos futuros.

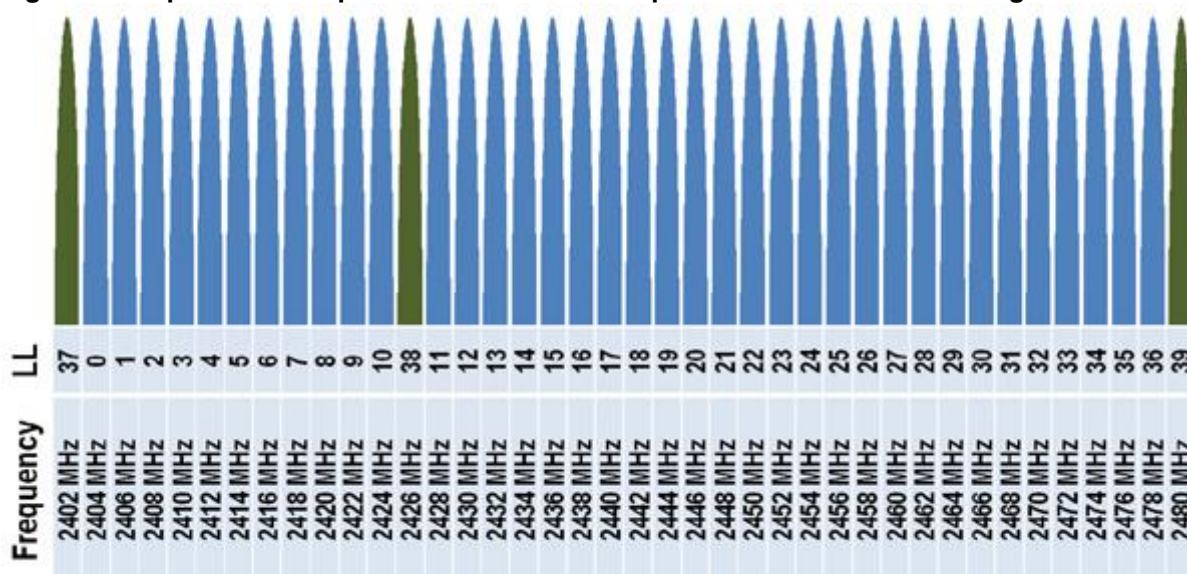
2 FUNDAMENTAÇÃO TEÓRICA

2.1 BLUETOOTH LOW ENERGY (BLE)

Define-se por *Bluetooth Low Energy* (BLE) a evolução do protocolo *bluetooth* tendo por foco aplicações de baixo consumo e orientadas pequenas distâncias (menores que 50m), introduzida na versão 4.0 do protocolo (PALATTELLA *et al.*, 2016).

O protocolo opera na banda ISM (aplicações industriais, científicas e médicas) de 2.4GHz, dispondo de um total de 40 canais de 2MHz, porém para otimização da energia, usam-se apenas 3 canais para o *scanning* e *advertising* de dispositivos (Figura 2).

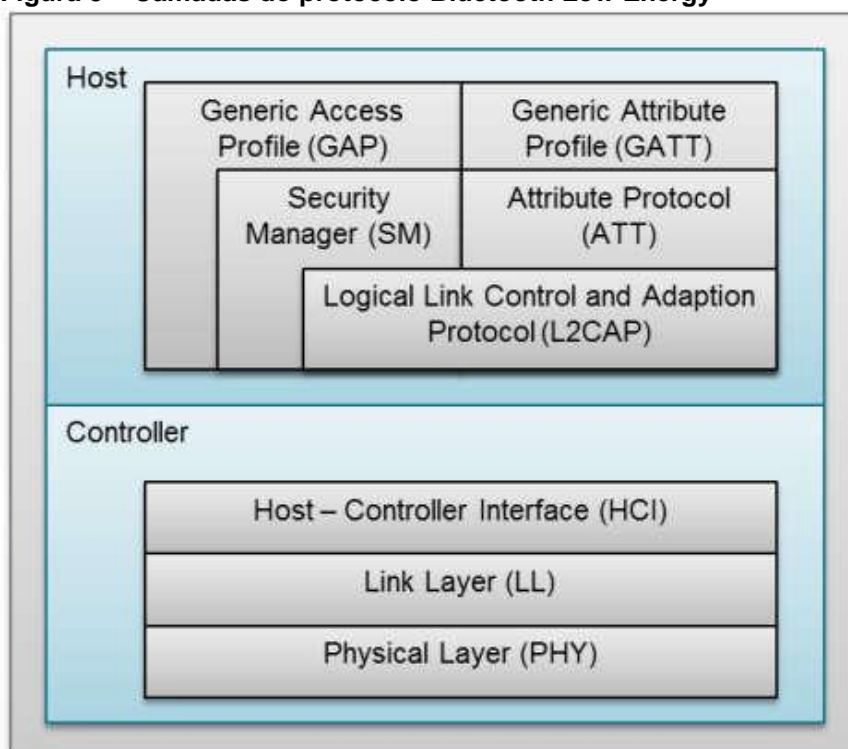
Figura 2 – Espectro de frequência BLE com ênfase para os canais de advertising



Fonte: Pedro (2011).

Conforme nota-se em Texas Instruments, em termos de organização das camadas de comunicação em um sistema BLE, apresentam-se dois elementos de destaque, como mostrado na Figura 3: a) GAP (*General Access Profile*) e b) GATT (*Generic Attribute Profile*) (TEXAS_INSTRUMENTS, 2016).

Figura 3 – Camadas do protocolo Bluetooth Low Energy



Fonte: Texas_Instruments (2016).

O GAP é responsável por aspectos relacionados à topologia da rede e ao estado de conexão do dispositivo, que pode ser: *Standby*, *Advertising*, *Scanning*, *Initiating* ou *Connected*.

Em termos de topologias típicas em ambientes de comunicação baseados em *Bluetooth Low Energy*, o GAP define 4 papéis típicos a serem assumidos por dispositivos:

- Transmissor: dispositivo propaga pacotes de *Advertising* mas não pode se conectar;
- Observador: dispositivo é capaz de procurar pacotes de *Advertising*, mas não é capaz de iniciar conexões;
- Periférico: além de propagar pacotes de *Advertising*, o dispositivo é passível de conexão mediante uma solicitação de um dispositivo central;
- Central: o dispositivo detecta pacotes de *Advertising* e é capaz de atuar como mestre ao estabelecer conexão com um Periférico.

A camada GATT atua no sentido de efetivamente possibilitar a troca de dados entre dois dispositivos caso estejam conectados.

Além das características técnicas acima, a alta disponibilidade de Smartphones compatíveis com BLE tendem a consolidar esse protocolo no uso de aplicações de curta distância em interações diretas com usuários (PALATTELLA *et al.*, 2016).

2.2 RASTREAMENTO VEICULAR

Nos setores de logística e transportes, a busca por eficiência acaba por demandar várias abordagens, dentre elas destaca-se o uso da localização em tempo real do veículo para reduzir custos. Tipicamente, os rastreadores são munidos de GPS e GPRS para comunicar sua localização em tempo real.

2.2.1 Global Positioning System (GPS)

Denomina-se GPS é um sistema composto por 24 satélites que periodicamente emitem sinais para receptores tendo por propósito calcular informações de latitude, longitude, altitude e velocidade (TIMBÓ, 2000).

2.2.2 GPRS / 2G

O *General Packet Radio Service* (GPRS) é um serviço de transmissão de dados que opera sobre a rede GSM com capacidade de transmitir até 115.2 kbps. Dada a sua alta cobertura e os requisitos de tráfego de dados de uma aplicação de rastreamento veicular, o uso do GPRS como meio de comunicação para a transmissão de informações de localização em tempo real é amplamente utilizado (CHADIL; RUSSAMEESAWANG; KEERATIWINTAKORN, 2008).

2.3 PASSIVE KEYLESS ENTRY SYSTEM (PKES)

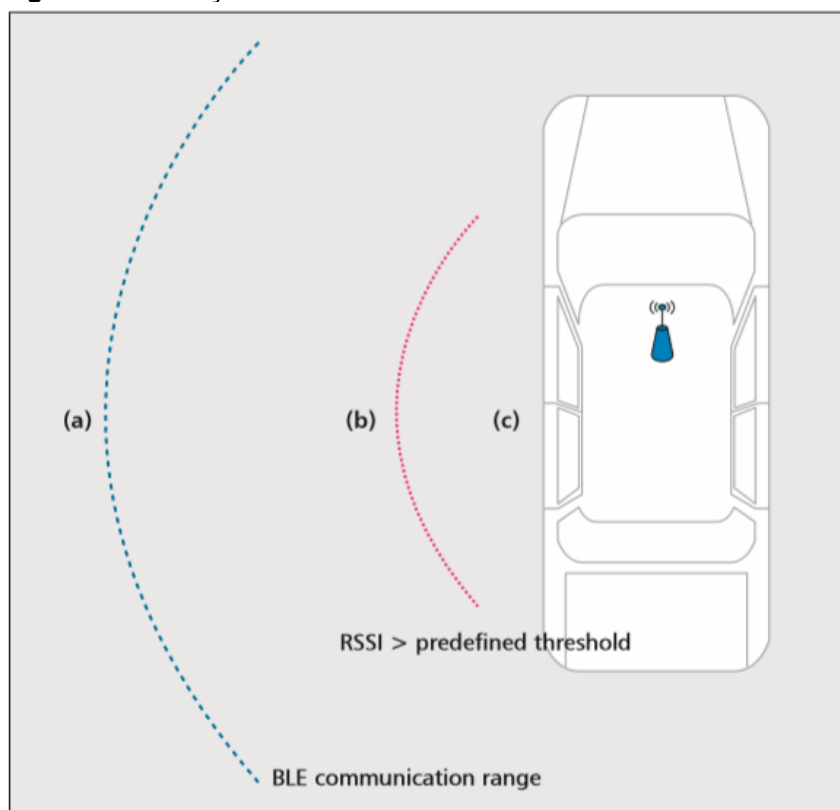
No contexto de aplicações veiculares, o *Passive Keyless Entry System* (PKES) se refere ao sistema capaz de detectar a chave na proximidade do veículo e destravar o veículo para acesso ao seu interior. Tipicamente, os fabricantes implementam essa funcionalidade usando RF de baixa frequência, mas existem provas de conceitos utilizando BLE para esse propósito.

Para se obter informação de proximidade, usa-se um parâmetro do lado do receptor denominado *Received Signal Strength Indicator* (RSSI). Esse indicador cuja unidade é dB, é um número relativo entre emissor e receptor que se relaciona à intensidade do sinal percebido pelo receptor. Dessa forma, na medida em que se perceber aumento do valor do RSSI aumentar, aumenta-se a potência do sinal recebido, logo, deduz-se que emissor e receptor estão em tendência de aproximação (GAO, 2015).

Em termos de topologia, o veículo possui um dispositivo central capaz de detectar os pacotes que o emissor irradia no ambiente. Uma vez detectado que determinado dispositivo está enviando pacotes, o dispositivo central deve avaliar se esse emissor está habilitado a atuar no veículo e validar o RSSI. Essa condição é representada pela região (a) na Figura 4.

Na medida em que o condutor se aproxima do veículo, o valor do RSSI vai aumentando. Ao aumentar e ultrapassar um certo valor, linha tracejada (b) na Figura 4, passa-se a considerar que o condutor está identificado e no interior do veículo.

Figura 4 – Detecção do condutor usando RSSI

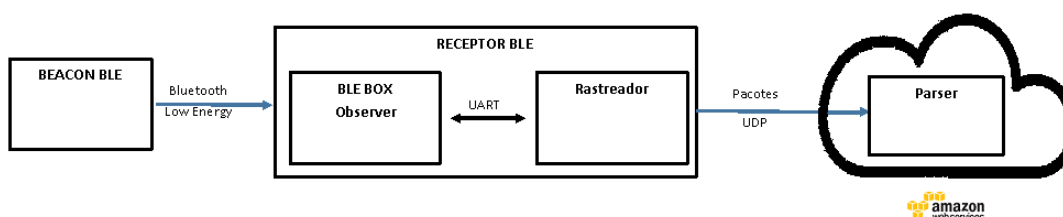


Fonte: Lin, Talty e Tonguz (2015).

3 DESENVOLVIMENTO

Para o desenvolvimento, imagina-se que o motorista estará de porte de um Beacon BLE, o qual irá transmitir sua identificação periodicamente (bloco BEACON BLE na Figura 5). Um dispositivo central (bloco RECEPTOR BLE na Figura 5) é responsável pela identificação e transmissão dos dados. Um elemento acoplado ao rastreador (bloco BLE BOX Observer na Figura 5), irá efetuar a varredura dos pacotes recebidos e uma vez identificando que seja uma identificação de motorista válida, irá comandar ao rastreador (atuando como *gateway*) a transmissão de um pacote informando o motorista identificado. Com isso, o rastreador transmite esse pacote e tal dado é recebido por uma aplicação em nuvem (Parser). Tais elementos utilizados no desenvolvimento podem ser visualizados na Figura 5.

Figura 5 – Diagrama em blocos da solução completa



Fonte: Autoria própria.

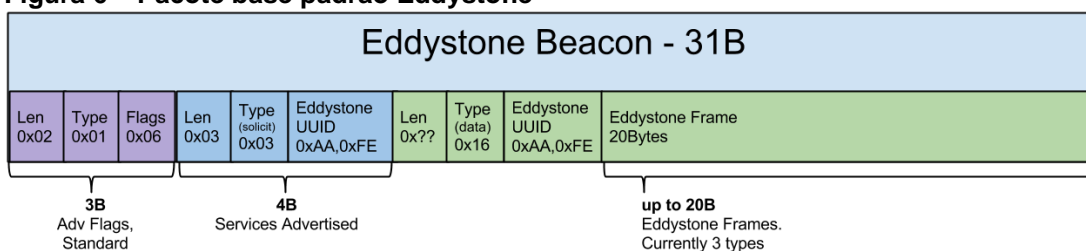
3.1 BEACON BLE

Denomina-se como Beacon BLE o elemento transmissor que propaga sua identificação usando o protocolo *Bluetooth Low Energy*. Esse transmissor irradia suas informações com um certo intervalo de transmissão e potência, parâmetros que serão definidos na análise de resultados e impactam diretamente na duração da vida útil do dispositivo.

3.1.1 Protocolo de Transmissão

Nesse projeto, utiliza-se o padrão *Google Eddystone* como protocolo base para transmissão de dados por parte dos beacons (Figura 6). O padrão *Eddystone* é um padrão aberto lançado em 2015 para dispositivos de detecção de proximidade baseado em BLE.

Figura 6 – Pacote base padrão Eddystone



Fonte: Adaptado de Arm_Mbed (2015).

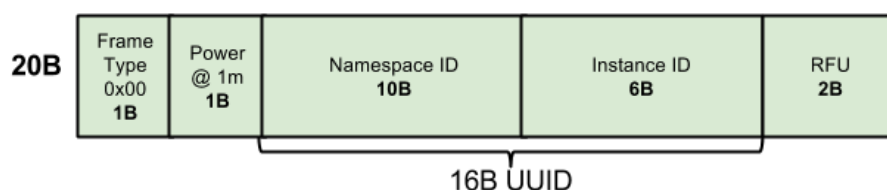
O padrão, apresentado ainda na Figura 6, especifica alguns tipos de pacotes, merecendo principal destaque:

- UID: identificador único do dispositivo composto por um *namespace* de 10 bytes e 6 bytes Instance ID;
- TLM: pacotes que podem trafegar dados relevantes para a operação do dispositivo, como bateria, temperatura, entre outros.

Nesse projeto, o frame típico será baseado no UID, apresentado na Figura 7, e possuirá o seguinte padrão:

- Frame Type: 0x00
- Power @1m: arbitrada como 0x10 – Esse valor é a potência transmitida a 1m de distância e necessitaria ser medido para ter um valor adequado;
- Namespace: 'mobi7ble' [hex = 0x6d6f626937626c650000]
- Instance ID: 123456789012

Figura 7 – Tipo de frame Eddystone UID

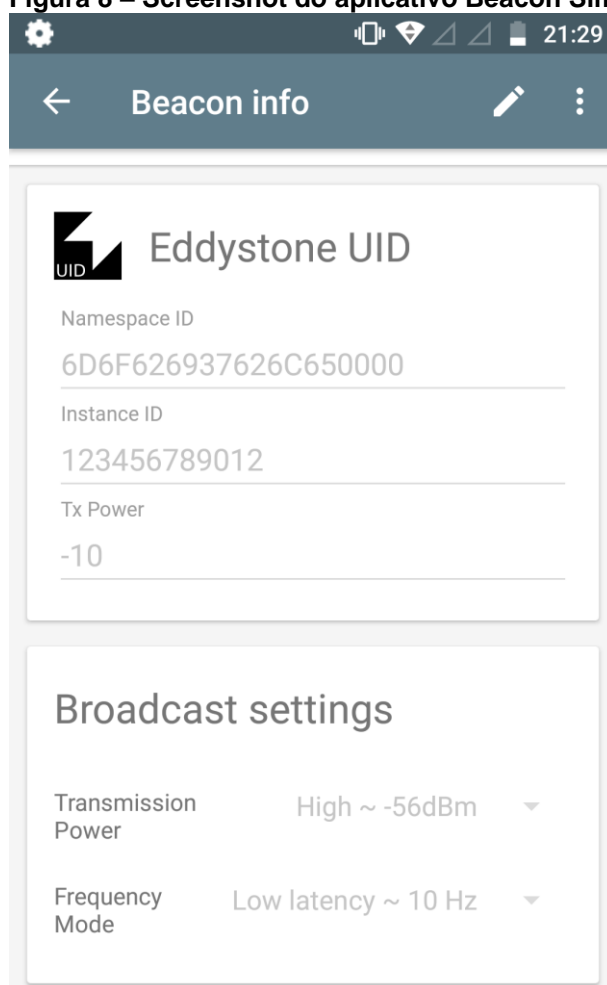


Fonte: Adaptado de Arm_Mbed (2015).

3.1.2 Implementação do Beacon

No desenvolvimento do projeto, usou-se o aplicativo Beacon Simulator, apresentado na Figura 8, disponível para Android (HIRIBARREN, 2019). Com esse aplicativo, pode-se emular o funcionamento de um beacon *Eddystone*, especificar o *payload*, além de parametrizar aspectos de potência e intervalo de transmissão.

Figura 8 – Screenshot do aplicativo Beacon Simulator



Fonte: Autoria própria.

3.2 RECEPTOR BLE

Denomina-se como receptor BLE o módulo responsável por efetuar a varredura periódica no ambiente do veículos em busca de beacons BLE elegíveis a serem condutores do veículo. Uma vez que esse condutor elegível é detectado a um certo nível de proximidade do veículo, um pacote UDP é enviado utilizando-se um rastreador veicular como *Gateway*.

3.2.1 BLE Box Observer

O BLE Box Observer é um hardware de elaboração própria baseado no chipset CC2640R2F. Esse microcontrolador é destinado a aplicações de baixo consumo em Bluetooth 4.2 e 5, além de possuir dois núcleos de processamento: um núcleo Cortex-M0 para a parte de Rádio Frequência e outro núcleo Cortex-M3 para a aplicação efetivamente.

No design customizado de hardware, apresentado na Figura 9, usaram-se 3 pinos como GPIO e uma interface UART com TX e RX para comunicação com o rastreador.

Figura 9 – Placa customizada conectada ao debugger

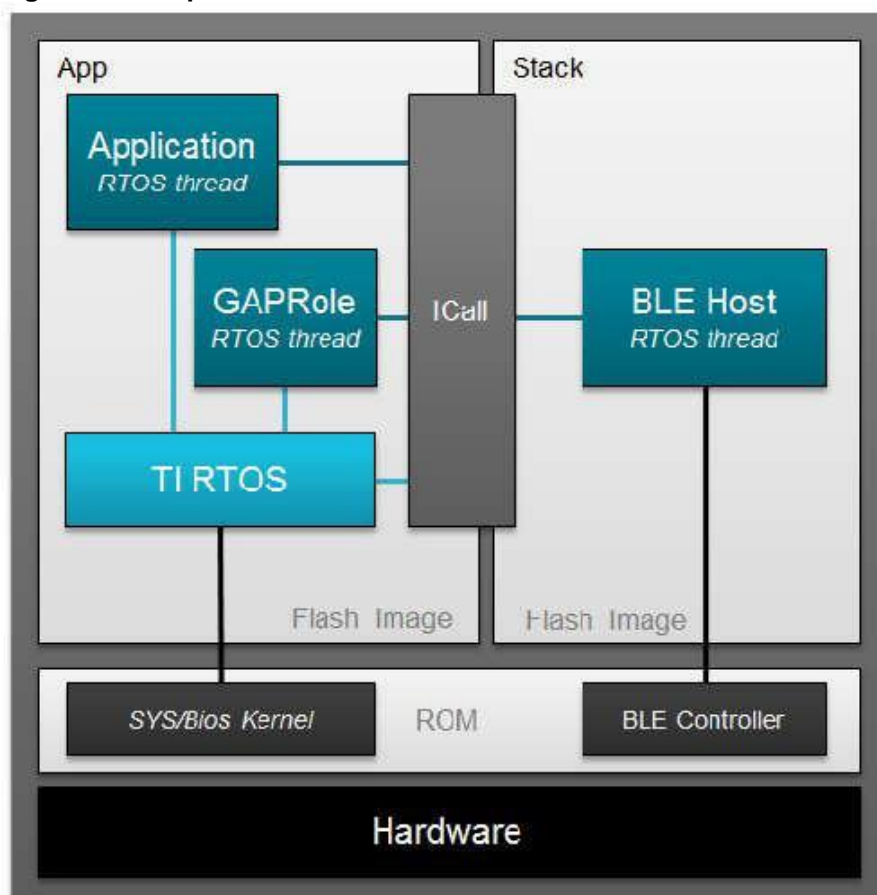


Fonte: Autoria própria.

No que se refere ao software embarcado, utilizou-se como sistema operacional o TI-RTOS da fabricante Texas Instruments (Figura 10). O TI-RTOS é um sistema operacional em tempo real, com suporte a várias *threads* e drivers de periféricos.

Além do sistema operacional, há embarcada uma imagem de biblioteca de funções Bluetooth (*stack*), a qual implementa as camadas mais baixas do protocolo Bluetooth até a interface GATT e GAP.

Figura 10 – Arquitetura de software do BLE BOX Observer



Fonte: Texas_Instruments (2018).

3.2.2 Rastreador

Na implementação desse projeto utilizou-se um rastreador comercial a fim de detectar eventos do veículo e ser usado como *gateway* para comunicação com um servidor remoto.

O rastreador utilizado dispõe de um total de três saídas, as quais podem ser usadas para acionar periféricos no veículo (por exemplo: Buzzer) e também oito entradas de sinais digitais, tipicamente usadas para detectar sinal de ignição e outros dados relevantes.

Pensando em uma aplicação veicular que é energizada por uma bateria de 12V, e pelo fato de GPS e o modem 2G consumirem muita energia estando ativos, implementou-se uma configuração no rastreador de tal forma que:

- Uma vez em movimento, o rastreador mantém GPS/2G ativos enviando pacotes em intervalo pequeno de tempo;

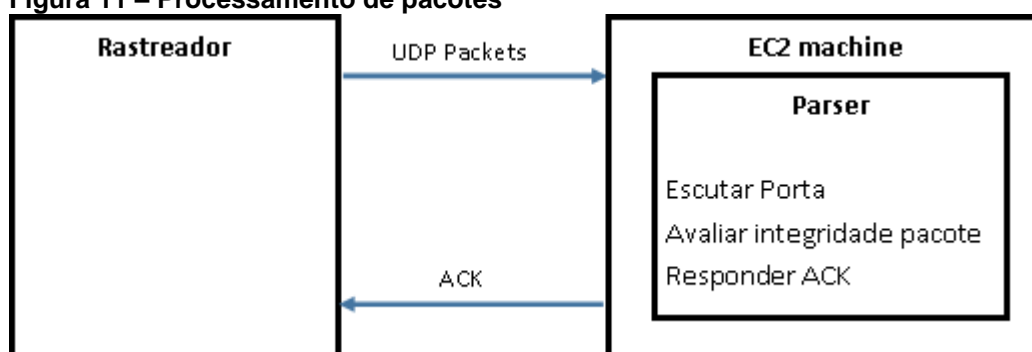
- Uma vez parado o veículo, o rastreador entra em modo de baixo consumo enviando informação a cada 30 minutos. Esse modo de baixo consumo pode ser interrompido pelo movimento do veículo ou caso o BLE BOX Observer deseje enviar um pacote de dados.

No que se refere ao uso do rastreador como gateway de mensagens do BLE BOX Observer, utilizou-se uma interface UART com protocolo específico para esse propósito. Especificamente, o BLE BOX deve enviar uma mensagem com um padrão específico e aguardar uma resposta de sucesso ou falha no envio.

3.3 PROCESSAMENTO DE PACOTES

Para o desenvolvimento do projeto, foi necessário desenvolver uma aplicação rodando remotamente a fim de realizar a interface com o dispositivo e realizar seu processamento. Esse processamento é resumido na Figura 11.

Figura 11 – Processamento de pacotes



Fonte: Autoria própria.

3.3.1 Infraestrutura

Decidiu-se por implementar a aplicação numa estrutura de computação em nuvem usando como ecossistema o ambiente Amazon Web Services (AMAZON_WEB_SERVICES, 2019). Para isso, usou-se o Serviço de EC2 (*Elastic Compute Cloud*) para subir uma máquina remotamente. Além disso, nas regras de segurança e firewall da máquina habilitou-se a entrada de pacotes do protocolo UDP na porta em que o rastreador se comunica.

3.3.2 Protocolo de Identificação do Condutor

Dentro do protocolo de comunicação do rastreador, decidiu-se criar eventos para identificar os eventos relacionados ao processo de identificação do condutor. Desconsiderando-se outras camadas do protocolo do rastreador e atendo-se exclusivamente ao *payload* de interesse, o pacote possui o formato conforme apresentado no Quadro 1.

Quadro 1 – Eventos de identificação de condutor

Campo	Tamanho (bytes)	Descrição	Exemplo
Evento de Identificação do Condutor	1	Identifica o evento ocorrido na identificação do <i>beacon</i>	0x81 – BEACON_FOUND (Beacon identificado pela primeira vez) 0x82 – BEACON_PING (Beacon identificado mandando pacotes periódicos) 0x83 – BEACON_DISAPPEAR (Beacon identificado parou de enviar pacotes)
Instance ID	6	Identifica o id do beacon transmitindo	0x123456789012
MAC BLE	6	Identifica o endereço físico do beacon transmitindo	0Xc6b0feba90c1

Fonte: Autoria própria.

3.3.3 Aplicação Parser

Foi desenvolvida uma aplicação em C/C++ para processar esses pacotes UDP. O software executa o seguinte processamento:

1. Escuta por pacotes UDP na porta de comunicação do rastreador;
2. Ao chegar o pacote, avalia se corresponde ao protocolo do dispositivo esperado e realiza avaliação de integridade;
3. Caso seja íntegro, a aplicação monta um pacote de ACK e o envia novamente ao dispositivo a fim de sinalizar que recebeu o pacote em questão;
4. Como última etapa, o pacote é mostrado na tela e encaminhada para uma fila de processamento na Amazon (serviço SQS – *Simple Queue Service*). Em uma aplicação comercial, supõe-se que algum consumidor irá consumir esses pacotes da fila e realizar o armazenamento dessas informações como for mais pertinente.

4 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

4.1 PERDA DE PACOTES

A fim de tornar a solução de detecção de condutor mais robusta e com melhor experiência ao usuário final, faz sentido avaliar a taxa de perda de pacotes para que se considere isso no código que fará a lógica de detecção do beacon.

Para esse propósito, foi customizado o firmware do BLE BOX Observer a fim de disponibilizar via interface UART os dados de *timestamp* de recebimento do pacote e o RSSI. No que se refere ao emissor, foram utilizados dois modelos comerciais de celulares rodando o app Beacon Simulator com uma taxa de transmissão de 3Hz.

Tendo-se estas informações como base, os cenários de testes verificados experimentalmente se fazem representados no Quadro 2. Ainda no Quadro 2, o parâmetro anteparo diz respeito à presença de um obstáculo entre transmissor e receptor que impeça a visada direta.

Quadro 2 – Cenários para levantamento de dados de RSSI e perda de pacotes

	Distância	Anteparo	Potência Tx (dBm)	Celular
Cenário 1	50cm	FALSE	-75	1
Cenário 2	50cm	TRUE	-75	1
Cenário 3	50cm	FALSE	-56	1
Cenário 4	1,5m	FALSE	-56	1
Cenário 5	1,5m	FALSE	-56	2

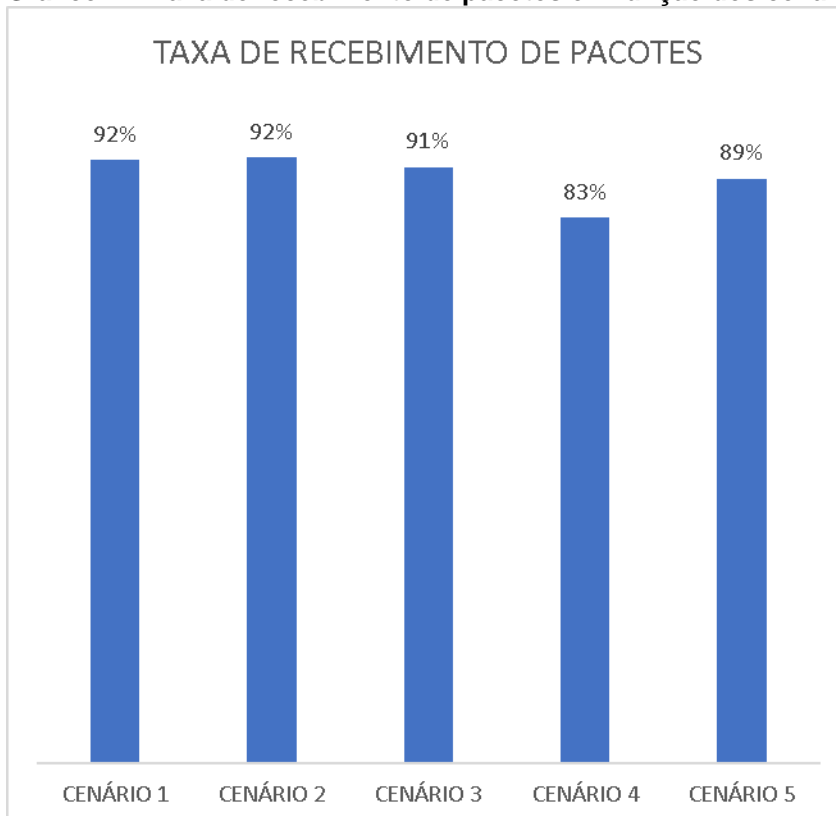
Fonte: Autoria própria.

Tendo-se em vista os cenários acima, faz-se relevante definir o parâmetro taxa de recebimento de pacotes. Tal parâmetro é um número entre 0 e 1 que é calculado pela razão entre a quantidade de pacotes recebidos pelo Receptor e a quantidade total de pacotes enviados pelo transmissor.

O Gráfico 2 mostra que nos testes efetuados a taxa de recebimento de pacotes oscilou entre 83% e 92%, confirmando a conclusão de Lin, Talty e Tonguz (2015), segundo o qual o protocolo *Bluetooth Low Energy* possui desempenho adequado para aplicações de comunicação no contexto veicular. No caso do artigo de Lin, Talty e Tonguz (2015), os números foram ligeiramente superiores, o que pode ser explicado pelo fato de no experimento conduzido nessa monografia

utilizaram-se pacotes de *Advertisement* ao passo que no artigo o autor tomou medidas estabelecendo a conexão BLE.

Gráfico 2 – Taxa de recebimento de pacotes em função dos cenários de teste



Fonte: Autoria própria.

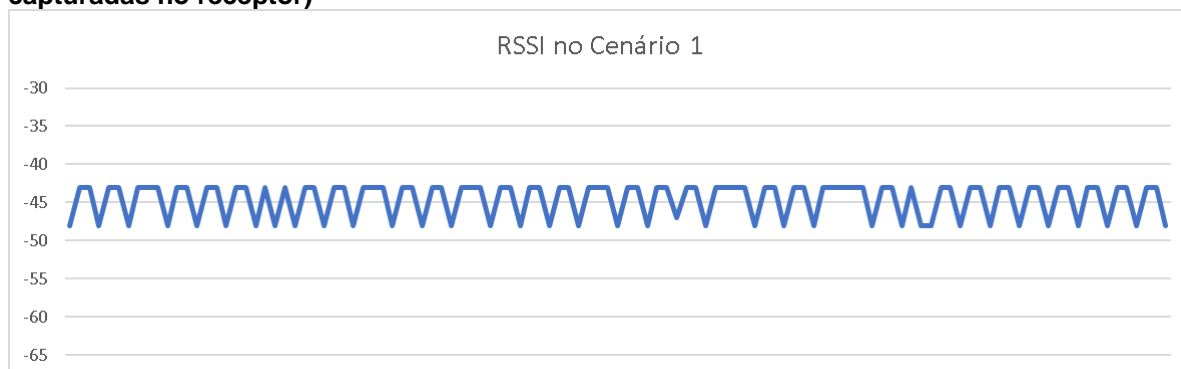
Outra conclusão interessante a partir dos Gráficos 1 e 2 é que a presença de anteparos não influenciou muito nas taxas, dado sua característica dielétrica. O que se conclui é que a distância é um fator relevante: ao se aumentar a distância de 50cm para 1,50m, percebeu-se que a taxa caiu de 91% a 83%. Para o cenário da aplicação dessa monografia, estima-se que a distância típica entre transmissor e receptor oscilará entre 0,50m e 1m, dessa forma, não se vislumbram impactos relevantes de perda de pacotes.

4.2 ANÁLISE DO COMPORTAMENTO PARÂMETRO RSSI

Faz-se relevante a avaliação do comportamento do parâmetro RSSI como base para detecção do condutor dentro do veículo. Diante disso, avalia-se a estabilidade desse parâmetro nos cenários descritos no item 4.1.

O Gráfico 3 mostra uma oscilação no valor do Rssi entre -48 e -43 dBm dada uma situação de visada entre receptor e transmissor.

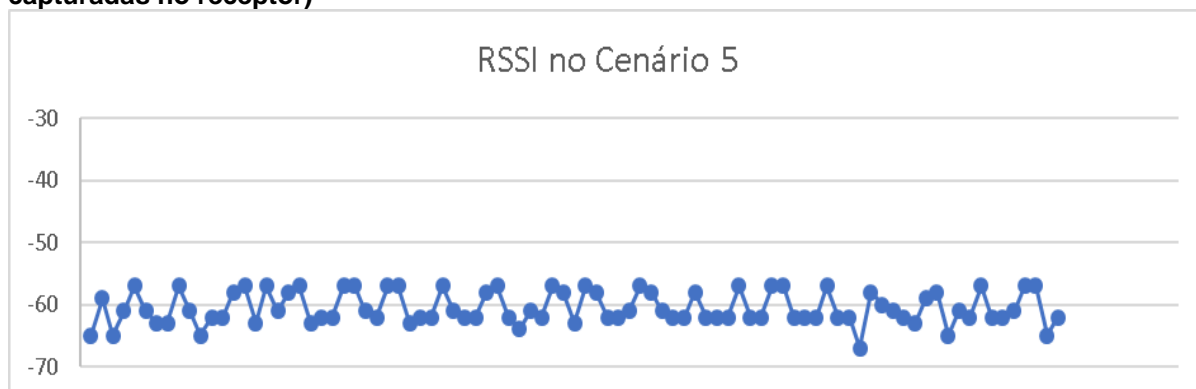
Gráfico 3 – Parâmetro RSSI no cenário 1 (eixo y – valor em dBm RSSI, eixo x amostras capturadas no receptor)



Fonte: Autoria própria.

No caso do Gráfico 4 é mostrada uma oscilação no valor do RSSI entre -57 e -65 dBm dada uma situação de visada entre receptor e transmissor. A diminuição do valor em relação ao Gráfico 2 é esperada, tendo-se em vista que com o aumento da distância entre transmissor e receptor (de 0,50m a 1,50m), é natural que a potência do sinal recebido pelo receptor sofra uma diminuição.

Gráfico 4 – Parâmetro RSSI no cenário 5 (eixo y – valor em dBm RSSI, eixo x amostras capturadas no receptor)



Fonte: Autoria própria.

Uma outra conclusão a partir dos gráficos acima é a oscilação natural do RSSI mesmo em condições estáveis. Assim como concluído por Karlsson e Lagerbielke (2017), verifica-se que o parâmetro é bastante suscetível a oscilações mesmo em condições de estabilidade de ambiente entre transmissor e receptor. Tal característica do parâmetro RSSI se mostra um desafio para aplicações que requeiram uma precisão alta de localização. No caso desta monografia, a qual

praticamente é uma detecção binária de condutor dentro ou condutor fora do veículo, considera-se que o parâmetro RSSI é suficiente para a resolução do problema proposto.

4.3 ANÁLISE DOS PARÂMETROS DO BEACON

Para se detectar o condutor o mais rápido possível, é necessário que se aumente a probabilidade de que o receptor receba um pacote válido do transmissor. Pensando nisso, deseja-se que o intervalo de transmissão seja o mais intenso possível na maior potência de transmissão possível.

Por outro lado, pensando-se na questão da bateria que tipicamente energiza um dispositivo transmissor, deseja-se que ele permaneça em modo de baixo consumo a maior quantidade de tempo possível.

Tendo-se em vista esses requisitos conflitantes, um ponto importante do projeto da solução é encontrar um compromisso entre esses dois requisitos. No caso específico dessa monografia, por se tratar de um beacon em telefone celular, isso não é tão crítico, mas em um cenário com hardware dedicado energizado a bateria, isso é crítico.

Tendo-se em vista as restrições acima, decidiu-se por utilizar o Aplicativo *Beacon Simulator* com os seguintes parâmetros:

- *Transmission Power: Low*. O aplicativo apresenta as opções de *Ultra Low*, *Low*, *Medium* e *High*.
- *Frequency Mode: Balanced 3Hz*. O aplicativo apresenta as opções de *Low Latency 10Hz*, *Balanced 3Hz* e *Low Power 1Hz*.

4.4 RECEBIMENTO DE PACOTES UDP

A última etapa de análise de resultados foi a implantação da solução final completa. Para isso, o firmware do BLE Observer foi parametrizado para detectar beacons a partir de um certo limiar de RSSI. Além disso, o BLE Observer passou a enviar esses pacotes usando interface UART para o rastreador.

O rastreador, uma vez tendo recebido estes dados pela interface UART, monta um pacote adicionando os dados de localização, data e horário, como mostrado na Figura 12. Esse pacote é transmitido usando protocolo UDP para um

identifica o identificador do Beacon e o campo amarelo trafega os dados do endereço MAC BLE.

Figura 14 – Exemplo de pacote UDP com evento de BEACON_DISAPPEAR

```
830546343793730101010400065DC084CC5CFAB91CF0D6F1FFE2A0E6540000000000000000000002E00
040004FFB92F0002080002000D83123456123456FAC615B093D0
0x83 - Evento de BEACON_DISAPPEAR
0x123456123456 - Instance ID do Beacon
0xFAC615B093D0 - MAC BLE do Beacon
```

Fonte: Autoria própria.

5 CONCLUSÃO

Na presente monografia, desenvolveu-se essa identificação usando-se *Bluetooth Low Energy*, incluindo o desenvolvimento três principais itens: *Beacon* para ser portado pelo motorista, detector de Beacons e recebimento desses pacotes em estrutura Cloud.

No caso específico desse trabalho, usou-se como Beacon um aplicativo em celular Android, porém uma aplicação real poderia ter um Hardware dedicado com uma bateria de menor capacidade. Nesse sentido, percebe-se que a especificação dos parâmetros de potência e latência de transmissão são fundamentais para entregar um equilíbrio adequado entre economia de bateria e experiência de detecção adequada pelo usuário.

No módulo detector de Beacons, percebeu-se como o parâmetro RSSI (indicador de força de sinal recebido) é bastante susceptível a variações em decorrência dos obstáculos, da distância, do ambiente e das características do transmissor. Com isso, uma aplicação de detecção ON-OFF baseada nesse parâmetro como a proposta desse trabalho se mostra pertinente, porém aplicações de detecção mais refinadas certamente demandam uma complexidade grande de implementação, como demonstrado no trabalho de Karlsson e Lagerbielke (2017).

O último dos blocos funcionais, cuja principal função é a retenção de pacotes UDP, mostrou-se apto para uma aplicação real. Como pontos de melhoria, poderia ser introduzida a retenção de dados históricos em banco de dados, o que certamente seria usado por uma aplicação consumidora que mostraria esses dados aos usuários de interesse.

No que se refere à segurança da solução, detectam-se vários pontos de melhoria na solução empregada nessa monografia. A solução adotada, mantendo os dados transmitidos estáticos ao longo do tempo abre brechas para que o dispositivo Beacon seja copiado. Diante disso, soluções de criptografia como o *Eddystone Ephemeral Identifier*, o qual irradia um identificador que varia a cada poucos minutos utilizando criptografia, o que dificulta a probabilidade da exploração de brechas de segurança na solução.

A execução dessa monografia mostra o grande potencial que o tema de IoT (*Internet of Things*, ou Internet das Coisas) apresenta para solucionar problemas de aplicações veiculares, seja no caso de uso explicitado no problema dessa

monografia ou em problemas mais amplos. Ainda, com o uso disseminado de smartphones munidos de chipsets BLE, tem-se um contexto adequado para interações diretas com o usuário nesse ambiente. Diante disso, cenários de uso que envolvem detecção de elementos ou comunicação com acessórios no ambiente veicular seriam passíveis de viabilização utilizando protocolo *Bluetooth Low Energy*.

REFERÊNCIAS

ARM_MBED. **BLE_EddystoneBeacon_Service**. Repositório de códigos arm MBED. Copyright© 2015 Arm Limited. Disponível em: <https://os.mbed.com/teams/Bluetooth-Low-Energy/code/BLE_EddystoneBeacon_Service/shortlog/>.. Acesso em: 03 nov. 2019.

AMAZON_WEB_SERVICES. **Amazon Simple Queue Service**: API reference. Copyright© 2019 Amazon Web Services, Inc. an. Disponível em: <<https://docs.aws.amazon.com/AWSSimpleQueueService/latest/APIReference/sqs-api.pdf#Welcome>>. Acesso em: 03 nov. 2019.

CHADIL, Noppadol; RUSSAMEESAWANG, Apirak; KEERATIWINTAKORN, Phongsak. **Real-time tracking management system using GPS, GPRS and Google Earth**. In: International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Tailândia, 2008. Disponível em: <<https://ieeexplore.ieee.org/document/4600454/authors#authors>>. Acesso em: 25 out. 2019.

CNT. **Transporte rodoviário**: Desempenho do setor, infraestrutura e investimentos. Brasília: Confederação Nacional do Transporte (CNT), 2017. Disponível em: <http://cms.cnt.org.br/Imagens%20CNT/PDFs%20CNT/Estudos%20CNT/estudo_transporte_rodoviario_infraestrutura.pdf>. Acesso em: 03 nov. 2019.

FASSA, Júlio César. **Motorista identificado! Conheça esse recurso do Contele Rastreador**. BLOG Contele rastreador, 15 jan. 2015. Disponível em: <<https://blog.contele rastreador.com.br/motorista-identificacao/>>. Acesso em: 03 nov. 2019.

GAO, Vincent. **Proximity and RSSI**. Bluetooth Blog, 21 set. 2015. Disponível em: <<https://www.bluetooth.com/blog/proximity-and-rssi/>>. Acesso em: 03 nov. 2019.

HIRIBARREN, Vincent. **Beacon Simulator**: Versão 1.5.1. Copyright© 2019 Google, modificado em: 23 mar. 2019. Disponível em: <https://play.google.com/store/apps/details?id=net.alea.beaconsimulator&hl=en_US>. Acesso em: 19 nov. 2019.

KARLSSON, Eric; LAGERBIELKE, Alexander. **Demonstrator development for phone as a key based on Bluetooth Low Energy**. Trabalho de Conclusão de Curso de Graduação em Engenharia Elétrica. Linköping University, Norrköping, Suécia, 2017.

LIN, Jiun-Ren; TALTY, Timothy; TONGUZ, Ozan K. **On the potential of Bluetooth Low Energy technology for vehicular applications**. IEEE Communications Magazine, Ad Hoc and Sensor Networks Series, v. 53, n. 1, p. 267– 275, jan. 2015. Disponível em: <<https://ieeexplore.ieee.org/document/7010544>>. Acesso em: 25 out. 2019.

MAIA, Flávia. **Brasil gasta o equivalente a 11,6% do PIB com logística em rodovias**: As causas para o alto custo logístico das mercadorias brasileiras esbarram na infraestrutura que contribui diretamente para o Custo Brasil. Correio Braziliense, Brasília, 11 ago. 2016. Cidades. Disponível em: <https://www.correio braziliense.com.br/app/noticia/cidades/2016/08/11/interna_cidad esdf,544003/brasil-gasta-o-equivalente-a-11-6-do-pib-com-logistica.shtml>. Acesso em: 03 nov. 2019.

PALATTELLA, M. R. *et al.* **Internet of Things in the 5G era: Enablers, architecture, and business models**. IEEE Journal on Selected Areas in Communications, v. 34, n. 3, p. 510-527, mar. 2016. Disponível em: <<https://ieeexplore.ieee.org/abstract/document/7397856/>>. Acesso em: 28 out. 2019.

PEDRO, Ricard M. **Bluetooth 4.0 Low Energy**: the future low-power consumption solution. Universitat Politecnica Catalunya, jul. 2011. Disponível em: <<http://upcommons.upc.edu/pfc/bitstream/2099.1/13249/1/memoria.pdf>>. Acesso em: 17 fev. 2019.

TEXAS_INSTRUMENTS. **SimpleLink™ CC26x2 SDK BLE5-Stack user's guide**. Copyright© 2010-2018 Texas Instruments. Disponível: <http://dev.ti.com/tirex/content/simplelink_cc26x2_sdk_2_20_00_36/docs/ble5stack/ble_user_guide/html/ble-stack-5.x-guide/index-cc26x2.html>. Acesso em: 03 nov. 2019.

TEXAS_INSTRUMENTS. **The CC2640R2F SDK platform**: Architecture. Copyright© 2016 Texas Instruments. Disponível: <http://dev.ti.com/tirex/content/simplelink_cc2640r2_sdk_1_40_00_45/docs/blestack/ble_user_guide/html/cc2640/architecture.html>. Acesso em: 03 nov. 2019.

TIMBÓ, M. A. **Levantamentos através do sistema GPS**. Minas Gerais: UFMG, 2000.

TRANSPORTE_ECONÔMICO. **5 dicas para economizar combustível**. Portal Transporte Econômico, 26 fev. 2018. Disponível em: <<https://transporteeconomico.com.br/economizar-combustivel/>>. Acesso em: 03 nov. 2019.

YOUNG, Christine. **The story behind the iButton**. mgineer Blog, 1 feb. 2018. Disponível em: <<https://www.maximintegrated.com/en/design/blog/iButton.html>>. Acesso em: 03 nov. 2019.