

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA  
CURSO DE ENGENHARIA INDUSTRIAL ELÉTRICA – ÊNFASE EM  
ELETRÔNICA/TELECOMUNICAÇÕES**

**VALERIA CATHERINE ZAVADZKI SANTOS  
VITOR KAZUO HIROZAWA**

**MONITOR DE DESEMPENHO DE GOLPES**

**TRABALHO DE CONCLUSÃO DE CURSO**

**CURITIBA**

**2014**

**VALERIA CATHERINE ZAVADZKI SANTOS**  
**VITOR KAZUO HIROZAWA**

## **MONITOR DE DESEMPENHO DE GOLPES**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Engenheiro Eletricista (Eletrônica/Telecomunicações), do Departamento Acadêmico de Eletrônica da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Sérgio Francisco Pichorim

Co-orientador: Prof. Dr. Rubens Alexandre de Faria

**CURITIBA**

**2014**

Folha destinada à inclusão da **Ficha Catalográfica** (elemento obrigatório somente para as dissertações) a ser solicitada ao Departamento de Biblioteca do Campus UTFPR (prazo: 3 dias) e posteriormente impressa no verso da Folha de Rosto (folha anterior).



Ministério da Educação  
**Universidade Tecnológica Federal do Paraná**  
Campus Curitiba  
Departamento Acadêmico de Eletrônica  
Curso de Engenharia Elétrica — Ênfase em  
Eletrônica/Telecomunicações



---

## **TERMO DE APROVAÇÃO**

**MONITOR DE DESEMPENHO DE GOLPES**

por

**VALERIA CATHERINE ZAVADZKI SANTOS**  
**VITOR KAZUO HIROZAWA**

Este Trabalho de Conclusão de Curso foi apresentado em 20 de março de 2014 como requisito parcial para a obtenção do título de Engenheiro Eletricista — Ênfase em Eletrônica/Telecomunicações. O candidato foi argüido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

---

Prof. Dr. Sérgio Francisco Pichorim  
Prof. Orientador

---

Prof. Dr. Rubens Alexandre de Faria  
Prof. Co-orientador

---

Prof. Dr. Kleber Kendy Horikawa Nabas  
Membro titular

---

Prof. Dr. Tasso Graeff Arnold  
Membro titular

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

Dedicamos este trabalho aos nossos pais  
e a Deus.

## AGRADECIMENTOS

Os autores deste trabalho agradecem os incentivos de amigos e familiares que ao longo do projeto foram compreensivos e deram suporte com palavras de ânimo e motivação, em especial às irmãs, Vanessa Zavadzki e Patricia Hirozawa, por nos auxiliarem na criação e arte do ícone do aplicativo desenvolvido no projeto. Ao professor de Educação Física Julio Cesar Bassan da UTFPR que proporcionou uma visão mais prática do projeto, cedendo seu tempo e o espaço de sua Academia para exemplificar as necessidades de um treinador de atletas de alto nível. Ao professor Ronie Müller por gentilmente emprestar seu kit de desenvolvimento, o qual contribuiu muito para o progresso do projeto. Ao amigo e engenheiro Luciano Gurgel, por responder a questões referentes aos lutadores de artes marciais e por disponibilizar sua Academia Arena para testes. Ao amigo e engenheiro Jardel Lucca por emprestar o saco de pancadas e dar grande suporte no experimento do pêndulo, auxiliando tanto na sua montagem, bem como no projeto, e também aos seus pais por cederem sua residência para efetuar os ensaios. Ao amigo Eduardo Bilk por auxiliar com conselhos e sugestões no desenvolvimento de software. Ao professor Dr. Walter Alberto de Siqueira Pedra da USP pelos conselhos e sugestões sobre como modelar a força do golpe de um pêndulo. Aos professores do Departamento Acadêmico de Física da UTFPR por ajudarem na modelagem da força do pêndulo. À academia Strikers House e ao professor Rafael por gentilmente nos ceder o espaço e suporte para a realização dos testes. Ao Almojarifado do Departamento Acadêmico de Mecânica da UTFPR por ceder a massa de aço para o pêndulo. Ao amigo Carlos Augusto Toledo Martins, ao engenheiro e professor Daniel Rossato, ao engenheiro Alexandre Hagihara e todos os amigos e colegas que deram dicas nos momentos de dificuldades. Claro, aos orientadores pelos esclarecimentos e conselhos durante o desenvolvimento do trabalho. E a Deus por dar-nos forças nos momentos mais difíceis.

## RESUMO

SANTOS, Valéria Catherine Zavadzki; HIROZAWA, Vitor Kazuo. **Monitor de desempenho de golpes**. 2014. 105 páginas. Trabalho de Conclusão de Curso de Bacharelado em Engenharia Industrial Elétrica – Ênfase em Eletrônica/Telecomunicações – Universidade Tecnológica Federal do Paraná. Curitiba, 2014.

Esportes de combate são esportes de contato competitivos, em que dois lutadores lutam um contra o outro respeitando certas regras de combate. Boxe e MMA são os mais populares e de crescimento mais rápido entre esses esportes. Tornaram-se uma importante fonte de entretenimento para os fãs e o público em geral. O treinamento dos lutadores de artes marciais utilizando sacos de pancada é uma das formas mais simples de aplicar e aperfeiçoar golpes, como socos e chutes. Nota-se, entretanto, que o atleta não tem nenhuma resposta quantitativa a respeito de sua performance. Com o intuito de fornecer ao lutador informações sobre o seu desempenho, o objetivo desse trabalho é desenvolver um sistema capaz de monitorar golpes de lutadores, de forma a quantificar a intensidade dos golpes em um treino, apresentando ao usuário do aplicativo de monitoramento um relatório de seu desempenho. Esse sistema possui um dispositivo sensor (62x39x17mm), no qual um acelerômetro tem a função de capturar as acelerações de golpes provenientes do lutador. Ele é fixado no saco de pancadas e contém um microcontrolador que faz a aquisição dos sinais de aceleração do sensor, processa-os de forma a obter informações úteis a respeito do golpe, como força, e envia na forma de pacote de dados, através de uma comunicação sem fio, *Bluetooth Low Energy*, para um *smartphone*, que conterá um aplicativo. Através desse aplicativo o lutador será capaz de visualizar o desempenho de seu treinamento. Como resultados, percebeu-se que há variações de resposta do sensor de acordo com a massa do saco de boxe. Assim, mostram-se relações estatísticas que dão o suporte para o correto cálculo de força. Com isso, o trabalho traz um sistema inédito com grandes perspectivas de aperfeiçoamento para futuras pesquisas.

**Palavras-chave:** Aplicativo em *smartphone*. *Bluetooth Low Energy*. *Bluetooth Smart*. Acelerômetro. Força de golpes.

## ABSTRACT

SANTOS, Valéria Catherine Zavadzki; HIROZAWA, Vitor Kazuo. **Punches performance monitor**. 2014. 105 pages. Conclusion work in Industrial Electrical Engineering – Electronics/Telecommunication – Federal Technology University – Paraná. Curitiba, 2014.

Combat sports are competitive sports where two fighters fight against each other abiding by certain rules of engagement. Boxing and mixed martial are the most popular and fastest growing combat sports and have become a major source of entertainment for fans and the general public. The training of martial arts fighters using punching bags is one of the simplest ways to apply and refine strokes, as punches and kicks. We notice, however, that the athlete has no quantitative response regarding your performance. In order to provide to fighters, information about their performance, this work aims to develop a system capable of monitoring fighter's punches, so quantify the intensity of the blows in a workout, presenting to users a monitoring application that reports their performance. This system will have a sensor board in which an accelerometer will serve to capture the accelerations of punches from the fighter. The sensor board is adapted to the punching bag and contains a microcontroller that must acquire acceleration signals from the sensor, filter the signal, process it in order to obtain useful information about the coup, as a force, and send it in the form of package data through a bluetooth low energy wireless communication to a smartphone, which will contain an application. Through this application the fighters will be able to view the performance of their training. As results, we noticed that there are variations of sensor response according to the mass of the punching bag. Then, we develop statistical relationships that give us support for the correct calculation of force. So, this work brings a totally new system with a great background for further research on the theme.

**Keywords:** Smartphone App. Bluetooth Low Energy. Bluetooth Smart. Accelerometer. Punches Force.

## LISTA DE ILUSTRAÇÕES

Figura 1: Diagrama do sistema de monitoramento de golpes. ....	15
Figura 2: Sistema massa-mola representando o princípio do acelerômetro. ....	23
Figura 3: Esquema de um acelerômetro capacitivo diferencial. ....	25
Figura 4: Logos Bluetooth®.....	28
Figura 5: Abstração do relacionamento App-BLE. ....	33
Figura 6: Relacionamento Central – Periférico.....	33
Figura 7: Periférico envia <i>advertising</i> ao Central.....	34
Figura 8: Métodos da framework Core Bluetooth.....	34
Figura 9: SensorTag da Texas Instruments. ....	35
Figura 10: Diagrama de Blocos do SensorTag.....	36
Figura 11: Estrutura da placa do SensorTag.....	37
Figura 12: Diagrama de Blocos do CC2541.....	39
Figura 13: Esquemático do CC2541. ....	40
Figure 14: Esquemático do Acelerômetro KXTJ9.....	41
Figura 15: Diagrama funcional do KXTJ9.....	42
Figura 16: Pilha de Protocolos BLE.....	43
Figura 17: Esquema de Eventos no BLE. ....	47
Figura 18: Papéis do iPhone e do SensorTag numa conexão BLE.....	49
Figura 19: Tarefas do software embarcado e suas respectivas prioridades. ....	50
Figura 20: Fluxograma do loop do OSAL .....	51
Figura 21: Duas imagens do aplicativo da Texas Instruments. ....	52
Figura 22: Diagrama do Over-The-Air Download (OAD). ....	53
Figura 23: Sinal de <i>clock</i> da I2C do acelerômetro.....	56
Figura 24: Tempo para ler dados do acelerômetro e liberar sua interrupção.....	57
Figura 25: Tempos na amostragem do acelerômetro.....	58
Figura 26: Sinal de interrupção do acelerômetro.....	58
Figura 27: Captura de tela do Software SmartRF Packet Sniffer BLE. ....	63
Figura 29: Sinal de corrente dividido em estados.....	64
Figura 29: Tempo de transmissão de pacotes de dados num evento de conexão....	66
Figura 30: Ícone do aplicativo "Fighter Pro". ....	69
Figura 31: Tela de entrada do "Fighter Pro".....	69
Figura 32: Tela com sensor encontrado e pronto para conexão. ....	70
Figura 33: Tela com conexão estabelecida e recebendo dados. ....	71
Figura 34: Tela de exportação dos dados recebidos em um arquivo .csv.....	72
Figura 35: Tela inicial da Macro Fighter Pro.....	73
Figura 36: Tela procura caminho do arquivo Log. ....	73
Figura 37: Tela da macro com o caminho inserido.....	74
Figura 38: Tela da macro com os dados preenchidos.....	74

Figura 39: Tela com a análise do treino. ....	75
Figura 40: Tela de confirmação de geração do relatório em PDF. ....	76
Figura 41: Tela de reinicialização da macro. ....	77
Figura 42: Orientação do acelerômetro. ....	78
Figura 43: Resposta da força a um impacto através do tempo. ....	80
Figura 44: Massa de aço 1020 com parafusos. ....	81
Figura 45: Massa fixada na haste do pêndulo. ....	81
Figura 46: Posição de impacto do sistema saco-pêndulo. ....	82
Figura 47: Modo de fixação do sensor no saco. ....	83
Figura 48: Teste para o saco de 80 kg. ....	83

Quadro 1: Valores de corrente para cada estado, considerando 6 pares de pacotes. ....	64
Quadro 2: Amostra de dados do teste com o pêndulo - Para saco de 80 kg. ....	90
Quadro 3: Desvio padrão das amostras colhidas no experimento do pêndulo para um saco de 80 kg. ....	90
Quadro 4 - Cronograma seguido no desenvolvimento do projeto. ....	93
Quadro 5 - Intervalo de tempo demandado para realização do projeto. ....	94
Quadro 6 – Estimativa de custos de desenvolvimento do modelo protótipo. ....	94
Quadro 7 – Principais eventos de riscos esperados no decorrer do trabalho e sua caracterização. ....	95

Gráfico 1: Sinal do impacto da massa do pêndulo solta a 70 cm. ....	84
Gráfico 2: Sinal do Impacto da massa do pêndulo solta a 90 cm com saturação no eixo z. ....	85
Gráfico 3: Força teórica calculada x Aceleração medida em unidades de G - Para um saco de pancadas de 28,4 kg. ....	86
Gráfico 4: Fator de consideração da massa - Para um saco de 28,4 kg. ....	87
Gráfico 5: Sinal do impacto da massa do pêndulo solto de 162 cm em um saco de 80 kg. ....	88
Gráfico 6: Sinais obtidos por 5 impactos da massa do pêndulo em um saco de 80 kg. ....	89
Gráfico 7: Resumo da amostra de dados do teste com pêndulo - Para um saco de 80 kg. ....	90
Gráfico 8: Força teórica calculada x Aceleração medida em unidade de G – Para um saco de 80 kg. ....	91
Gráfico 9: Força teórica calculada x Aceleração medida em unidades de G - Para um saco de 80 kg - Regressão Quadrática. ....	91
Gráfico 10: Fator de consideração da massa - Para um saco de 80 kg. ....	92

## LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

BLE	<i>Bluetooth® Low Energy</i>
iOS	Sistema operacional para dispositivos de mobilidade Apple
MMA	<i>Mixed Martial Arts</i>
VBA	<i>Visual Basic for Applications</i>
SDK	<i>Software Development Kit</i>
XCode	Ambiente de desenvolvimento
Wifi	<i>Wireless network</i>
Piconet	Uma rede de dispositivos conectados em ad hoc via bluetooth
RF	<i>Radio Frequency</i>
RAM	<i>Random-Access Memory</i>
LGA	<i>Land Grid Array</i>
TEMA	<i>TrackEye Motion Analysis</i>
FAB	<i>Functional Assessment of Biomechanics</i>
JPL	<i>Jet Propulsion Laboratory</i>
ISM	<i>Industrial, Scientific and Medical</i>
AFH	<i>Adaptive Frequency-hopping</i>
WLAN	<i>Wireless Local Area Network</i>
USB	<i>Universal Serial Bus</i>
Zigbee	Conjunto de especificações para a comunicação sem fio
Ad Hoc	Rede ponto a ponto
I2C	<i>Inter-Integrated Circuit</i>
BR	<i>Basic Rate</i>
EDR	<i>Enhanced Data Rate</i>
PHY	<i>Physical</i>
Mac OSX	Sistema operacional para computadores Apple
SIG	<i>Bluetooth Special Interest Group</i>
LE	<i>Low Energy</i>
MAC	<i>Alternate Media Access Control</i>
AES	<i>Advanced Encryption System</i>
DMA	<i>Direct Memory Access</i>
GFSK	<i>Gaussian Frequency-Shift Keying</i>
FHSS	<i>Frequency-Hopping Spread Spectrum</i>

API

*Application Programming Interface*

## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>14</b>
1.1 OBJETIVOS.....	14
1.1.1 Objetivos Específicos.....	14
1.1.2 Diagrama .....	15
1.1.3 Escopo do produto.....	15
1.2 JUSTIFICATIVA.....	16
1.3 RESULTADOS ESPERADOS .....	16
1.3.1 Tecnológicos.....	16
1.3.2 Científicos .....	17
1.3.3 Econômicos .....	17
1.3.4 Sociais .....	17
1.3.5 Ambientais .....	17
1.4 METODOLOGIA .....	17
1.4.1 Hardware .....	17
1.4.2 Aplicativo .....	18
<b>2 EQUIPAMENTOS DE MEDIÇÃO DE GOLPES.....</b>	<b>20</b>
<b>3 COMPORTAMENTO MECÂNICO DO GOLPE NO SACO DE PANCADAS.....</b>	<b>22</b>
<b>4 ACELERÔMETRO.....</b>	<b>23</b>
<b>5 BLUETOOTH®.....</b>	<b>26</b>
5.1 VERSÕES DO BLUETOOTH®.....	27
5.2 LOGOS BLUETOOTH®.....	27
5.3 BLUETOOTH® LOW ENERGY (BLE) .....	29
5.3.1 Por que BLE?.....	29
<b>6 APLICATIVO.....</b>	<b>31</b>
6.1 O SISTEMA IOS .....	31
6.1.1 Frameworks .....	32
<b>7 PROCEDIMENTOS METODOLÓGICOS.....</b>	<b>35</b>
7.1 SENSORTAG .....	35
7.1.1 Principais Componentes .....	36
7.1.2 Alimentação Elétrica .....	37
7.1.3 Conversor DC/DC .....	38
7.1.4 CC2541 .....	38
7.1.5 Acelerômetro KXTJ9.....	41
7.2 BLE STACK.....	42
7.2.1 Pilha de Protocolos BLE .....	43
7.2.2 Terminologias do GATT .....	44
7.2.3 Identificadores.....	45
7.2.4 Operação do GATT.....	45
7.2.5 Operações do BLE.....	46

7.3 OPERAÇÃO DO SENSORTAG .....	47
7.3.1 Camada de Sistema Operacional Abstrato (OSAL) .....	49
7.4 APLICATIVO DA TEXAS INSTRUMENTS .....	52
7.5 FUNÇÃO OAD (OVER-THE-AIR DOWNLOAD) .....	52
7.6 AMOSTRAGEM DO ACELERÔMETRO .....	53
7.6.1 Limitação do SensorTag .....	54
7.6.2 Alterações de firmware .....	54
7.7 CONSIDERAÇÕES SOBRE A NORMA SAE J211-1 (1995) .....	61
7.8 TEMPO DE DURACAO DA BATERIA (CR2032) DO SENSORTAG .....	62
7.9 DESENVOLVIMENTO DO APLICATIVO .....	67
7.9.1 Requisitos de Desenvolvimento .....	67
7.9.2 Funcionamento .....	67
7.10 TRATAMENTO DOS DADOS .....	72
7.11 MEDIÇÕES DE GOLPES E CALIBRAÇÃO .....	77
7.11.1 Medição de Impacto por Acelerômetro .....	78
<b>8 RESULTADOS E ANÁLISES.....</b>	<b>84</b>
<b>9 TEMPO E RECURSOS DESPENDIDOS NO PROJETO.....</b>	<b>93</b>
9.1 CRONOGRAMA .....	93
9.2 TEMPO DEMANDADO .....	94
9.3 CUSTOS.....	94
9.4 ANÁLISE DE RISCOS .....	95
<b>10 CONSIDERAÇÕES FINAIS.....</b>	<b>97</b>
<b>REFERÊNCIAS.....</b>	<b>98</b>

## 1 INTRODUÇÃO

Lutar é um instinto do ser humano. Os povos primitivos tiveram que aprender a lutar por necessidade de sobrevivência, seja por defesa, comida, abrigo, terra, interesses políticos ou sociais (YASSAKA, 2004).

Atualmente, a luta é vista como um esporte de combate que pode ser representada por diversas modalidades. Sem existir a necessidade de lutar fisicamente pela sobrevivência, sua finalidade se baseia em melhorar a saúde, ganhando força, preparo físico e habilidades de coordenação motora. Por outro lado, pode-se dizer que ela pode ser útil em algum caso de agressão, tornando-se um instrumento de defesa pessoal.

Profissionais da área de educação física e lutadores de artes marciais têm uma grande necessidade de um sistema de medição de força que traga mais mobilidade para o golpe do atleta, e ainda a necessidade de uma ferramenta que seja de fácil acesso para a gestão de dados de treinamento.

Através dessas necessidades, buscou-se desenvolver um sistema de monitoramento de golpes remoto e sem fio que auxiliasse atletas de luta.

### 1.1 OBJETIVOS

O projeto tem como objetivo desenvolver um dispositivo que capte os dados referentes à força aplicada a um saco de pancadas, por meio de um acelerômetro, e envie-os, via comunicação sem fio, para um aplicativo em *smartphone*.

#### 1.1.1 Objetivos Específicos

- 1) Desenvolver um módulo acelerômetro;
- 2) Implementar um algoritmo que processe sinais do acelerômetro para extrair informações úteis dos golpes;
- 3) Desenvolver um módulo de comunicação sem fio;
- 4) Desenvolver um *software* (aplicativo) para *smartphone*.

### 1.1.2 Diagrama



**Figura 1: Diagrama do sistema de monitoramento de golpes.**  
**Fonte: Autoria própria.**

### 1.1.3 Escopo do produto

Apresenta-se abaixo uma sugestão das interfaces que serão utilizadas no projeto. Elas poderão ser alteradas, se necessário, durante o desenvolvimento do sistema.

Serão utilizados os seguintes recursos:

- Saco de pancadas;
- Luvas de boxe;
- Módulo sensor e de comunicação sem fio CC2541 SensorTag;
  - o Características resumidas:

Simplifica aplicações de sensores de baixo consumo utilizando *Bluetooth® 4.0*; *Bluetooth* de baixo consumo de 2,4 GHz e *Proprietary RF System-on-Chip*, Núcleo microcontrolador 8051 de baixo consumo e de alta performance, 256 kB de *In-System-Programmable Flash*, 8 kB de RAM (*Random-Access Memory*), Interface I2C, Acelerômetro KXT J9 da Kionix

(*Tri-Axis, User Selectable  $\pm 2g$ , 4g, 8g, Digital (I2C) Output, 3x3x0.9mm LGA*), TPS62730 (*Step Down Converter With Bypass Mode for Ultra Low Power Wireless Application*) e Alimentação por bateria de 3V.

- Aplicativo para *Smartphone*;
- *Smartphone*;

## 1.2 JUSTIFICATIVA

Observa-se cada vez mais a utilização de tecnologias para guiar e melhorar o desempenho de esportistas de alto nível. Coletar informações a respeito da atividade de um atleta é de grande importância para os profissionais de educação física que trabalham para garantir a saúde de atletas e introduzir novas formas de treinamento, visando a otimização do tempo de treino com relação aos resultados a serem alcançados.

Com o propósito de disponibilizar essas informações de praticantes de artes marciais, os alunos decidiram implementar um sistema de monitoramento de golpes que informa um relatório do treino em um aplicativo de *smartphone*.

Além disso, almeja-se que o projeto seja destinado também para lutadores amadores, de forma que signifique um método de auxílio e motivação a atividades físicas.

## 1.3 RESULTADOS ESPERADOS

### 1.3.1 Tecnológicos

- Composição de um aplicativo acompanhando às tendências do mercado atual, adaptando-o para *smartphone*;
- Composição de um *software* embarcado para o *hardware* de um sistema de aquisição de força proveniente de golpes;

### 1.3.2 Científicos

- Composição de artigo científico;
- Composição de uma ferramenta de monitoramento de performance de lutadores.

### 1.3.3 Econômicos

- Composição de um sistema otimizando os custos de projeto, de forma a obter um produto de fácil acesso financeiro.

### 1.3.4 Sociais

- Composição de um sistema de fácil aquisição, permitindo que tanto atletas profissionais quanto amadores possam se beneficiar do seu uso.

### 1.3.5 Ambientais

- Composição um sistema que otimize o baixo consumo de energia.

## 1.4 METODOLOGIA

### 1.4.1 Hardware

O *hardware* a ser desenvolvido se localizará em um saco de pancadas, e contará com um sensor acelerômetro, um microcontrolador e um módulo de comunicação sem fio.

#### 1.4.1.1 Módulo Acelerômetro

O sensor acelerômetro irá permitir quantificar a força aplicada pelos lutadores em seus golpes, esse sinal será então adequado por um conversor analógico digital para poder ser processado pelo microcontrolador e enviado via comunicação sem fio ao smartphone. O acelerômetro utilizado é digital, significando que o conversor analógico digital está embutido no chip do acelerômetro.

Primeiramente, planeja-se a programação e verificação deste módulo, pois este é de vital importância para a construção do projeto.

#### 1.4.1.2 Módulo comunicação sem fio

A comunicação sem fio escolhida foi o *Bluetooth®*, pois consome menos energia que o Wifi (VOGLER, 2014; BALANI, 2007; DIFFEN, s.d.) e possui amplas referências bibliográficas quanto ao uso dessa tecnologia em aplicativos para *smartphones*, facilitando assim o trabalho dos alunos. Ainda, usaremos a versão 4.0 (*Bluetooth 4.0, Bluetooth Low Energy, Bluetooth Smart*) por ser compatível com a plataforma e *smartphone* escolhidos para o desenvolvimento do projeto, e também por ser possível a implementação futura em outras plataformas.

#### 1.4.2 Aplicativo

O aplicativo para *smartphone* receberá os dados obtidos pelo *hardware* desenvolvido onde a partir deles gerará uma análise estatística capaz de contribuir para o desenvolvimento do atleta.

A tela inicial do aplicativo determina a conexão com o dispositivo de aquisição de dados, quando encontrado o dispositivo e estabelecida a conexão mostrar-se-á outra tela onde o botão “começar treino” disparará o cronômetro na tela e fará a aquisição dos dados. O botão “terminar treino” congelará o cronômetro, finalizará a captura de dados e mostrará o gráfico do treino com as estatísticas estabelecidas: o valor máximo, o valor médio e o desvio padrão da amostra

adquirida com o treino realizado. Como dados estatísticos mostrar-se-á o valor máximo da amostra, sinalizando o golpe mais forte daquele treino, o valor médio, demonstrando a força média dos golpes, e o desvio padrão, onde mensura-se o quanto variou a intensidade dos golpes em torno da média, ou seja, podendo avaliar a resistência física durante o treino. Esses valores estatísticos do treino são de grande valia para um treinador e serão detalhados mais na sequência.

A plataforma de desenvolvimento escolhida para o projeto foi o iOS, onde a linguagem para a construção de aplicativos é o “objective-C”, próximo da linguagem C, que é conhecida pelos alunos. Enquanto que a plataforma Android contaria com mais tempo de estudo pois utiliza linguagem Java, não trabalhada ainda pelos membros do projeto e possui pouca documentação sobre o *Bluetooth* 4.0. Outro motivo é que um dos membros do projeto já dispunha de um *smartphone* preparado para a comunicação por BLE (*Bluetooth Low Energy*), que trabalha com iOS. Essa soma de fatores contribuiu para essa escolha, pois acabou tornando-se menos oneroso o desenvolvimento.

## 2 EQUIPAMENTOS DE MEDIÇÃO DE GOLPES

Existem muitos artifícios e materiais empregados para auxiliar lutadores de artes marciais a aprender a dar um soco eficiente, compreender a mecânica do chute, atacar e defender, além de exercícios para melhorar a resistência. Entretanto, raramente encontram-se aparelhos de tecnologia que auxiliem os combatentes no processo de aprendizagem (NAVAS et al., 2012).

Em reportagem exibida em um grande meio de comunicação nacional, foi explícita essa demanda. Na matéria, o preparador físico exalta que um bom lutador deve atender três requisitos: força, velocidade de golpes e reflexo (GLOBOESPORTE.COM, 2013). A força é importantíssima, pois a partir desse dado pode-se quantificar a possibilidade de um nocaute. O nocaute é o término de uma luta, ocorre após a aplicação de um golpe, como no boxe e MMA, e tem-se um vencedor, no caso, o lutador que aplicou o golpe (WIKIPEDIA NOCAUTE, 2014). Durante essa matéria do GloboEsporte.com (2013), para quantificar a força, foi utilizada uma plataforma de medição de força (balança) na posição vertical acoplada a um aparador de golpes, assim os socos e chutes aplicados no aparador tinham uma resposta equivalente de força em kg. No caso, o teste aplicado com atletas de ponta demonstrou que a força exercida equivale a pesos variando de 85 kg a 173 kg.

Em outra pesquisa, os autores tinham como objetivo mensurar a força de chutes de certa modalidade de arte marcial, o Muay Thai. Para isso contaram com um sistema com “dois sensores interdependentes ligados a uma placa de alumínio, inserido em espuma para a absorção do impacto, revestido com lona e fixado em uma estrutura metálica com regulagem de altura” de forma que, experimentalmente, definiram uma variável ‘f’ para quantificar a força, que seria equivalente ao impacto de uma massa de 1,222kg que partiu do repouso a uma altura de 1m e foi solta em queda livre (YASSAKA, 2004).

Mack et al. (2010) utilizam um recurso de instrumentação do lutador avaliando a biomecânica de um soco e sua correlação com a força. Para isso, lutadores amadores foram instruídos a bater em um manequim masculino (“*Hybrid III (HIII) 50th percentile male dummy*”), usualmente chamado de BOB, aplicando dois tipos de golpes: o gancho e o direto. A força do soco foi então calculada usando os dados

vindos do BOB, os quais foram adquiridos por três acelerômetros alocados no centro de gravidade da cabeça do BOB. Já a velocidade de pré-impacto da mão foi medida usando *TrackEye Motion Analysis* (TEMA), que usa uma câmera estática coletando 500 quadros por segundo. Ainda, os autores utilizaram o *Functional Assessment of Biomechanics* (FAB), que é uma nova ferramenta de análise de movimento e consiste em 13 sensores sem fio, que contêm acelerômetros, giroscópios e magnetômetros, para calcular os valores de força gerados pelo corpo do atleta.

Navas et al. (2012) utilizam o acelerômetro para auxiliar o lutador no treino do seu nível de guarda, para isso o projeto desenvolve uma luva instrumentada baseada em três métricas: número de socos, força do soco e monitoramento do nível de guarda. Nesse projeto, quando o lutador dá um golpe e acerta o alvo, nesse caso, o sensor de força, o dado coletado é entrada para um microprocessador Arduino. Esse dado é então manipulado pelas rotinas de *software* para determinar se o golpe conta como um soco válido ou não. Eles conseguem isso impondo um valor limite para assim contar apenas os golpes acima de tal valor de força, eliminando outros valores que não correspondem ao critério. Segundo os autores, fazendo dessa forma eles podem assegurar que o lutador está seguramente acertando o alvo, o saco, ao contrário de estar brincando com ele.

### 3 COMPORTAMENTO MECÂNICO DO GOLPE NO SACO DE PANCADAS

Para entender como trabalhar com a força em um saco de boxe, é primordial avaliarmos as condições mecânicas do sistema. Para isso, os alunos partiram dos conhecimentos básicos adquiridos em física, no início do curso de engenharia, e ainda de consultas a físicos para maiores esclarecimentos.

Basicamente o sistema saco-golpe está ligado à segunda lei de Newton e às equações básicas da dinâmica:

1. Segunda lei de Newton

$$\vec{F} = m \cdot \vec{a} \quad (1)$$

$$\vec{F} = \frac{d\vec{I}}{dt} \quad (2)$$

2. Equação de conservação da Energia

$$E_{mecânica} = \left( mgh + m \frac{v^2}{2} \right)_{final} - \left( mgh + m \frac{v^2}{2} \right)_{inicial} \quad (3)$$

3. Equação da conservação da quantidade de movimento

$$\Delta Q = m_{final}v_{final} - m_{inicial}v_{inicial} \quad (4)$$

$$\Delta \vec{Q} = \vec{I} \quad (5)$$

Para emular um golpe em um saco de boxe e verificar a repetitividade do processo, e assim validar a utilização do dispositivo proposto nesse projeto, construiu-se um pêndulo.

Com este, foi possível avaliar a força teórica versus força mensurada, e a partir dessa análise, pudemos estimar um fator de correção de modo que o resultado mostrado ao usuário seja compatível com a realidade.

## 4 ACELERÔMETRO

Os sensores acelerômetros convertem aceleração, tanto linear quanto angular em um sinal elétrico de saída. Ainda, eles utilizam a segunda lei de Newton,  $F = m.a$ , para a medição de força em um objeto que se tem uma massa conhecida. Pode-se medir a força de muitas maneiras, mas a mais utilizada é a aplicação de um sistema massa-mola em um corpo de prova, como na Figura 2.

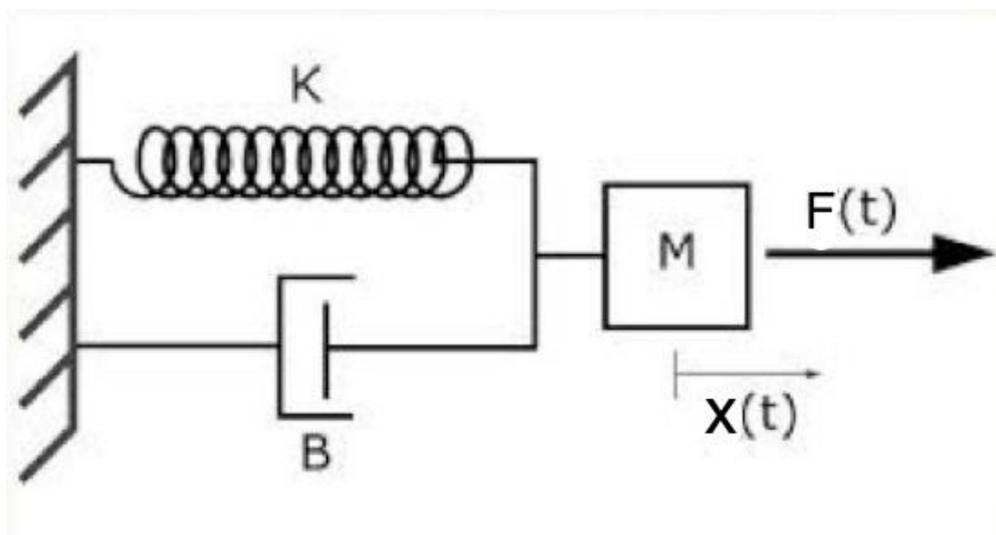


Figura 2: Sistema massa-mola representando o princípio do acelerômetro.  
Fonte: Adaptado de PUC-RJ (2014).

$$\vec{F} = m\vec{a} = m \frac{d^2\vec{x}}{dt^2} + B(\vec{x}) \frac{d\vec{x}}{dt} + K(\vec{x})\vec{x} \quad (6)$$

Para pequenos deslocamentos, a constante da mola  $K(x)$  é considerada constante. Então, em equilíbrio, a força exercida pela mola é igual à força vinda da aceleração do corpo de prova, pois o deslocamento da mola,  $x$ , é o parâmetro que pode ser convertido em um sinal elétrico.

Os dois métodos mais comuns são: medir a mudança na resistência de um material piezo resistivo e medir a mudança na capacitância entre materiais elétricos móveis e fixos. Há também uma maneira alternativa de medir diretamente a aceleração da força exercida em uma massa (corpo de prova), medir a mudança na carga de um material piezoelétrico.

No entanto, o tipo de sensor que é usado com maior frequência em acelerômetros é o sensor capacitivo. Acelerômetros capacitivos “*Single-Sided*”

podem ser construídos com o eixo sensível perpendicular ao plano da matriz de silicone e às placas do capacitor no plano da matriz, permitindo uma área muito grande do capacitor e que resultam numa elevada sensibilidade (sensores sísmicos, como o sensor de JPL costumam usar esse esquema) (GRAHAM, 2000).

No caso do acelerômetro utilizado no projeto, a sensibilidade da aceleração é baseada no princípio da capacitância diferencial decorrente do movimento induzido por aceleração do elemento sensor, que utiliza o modo de cancelamento comum para diminuir os erros da variação do processo, a temperatura e pressão ambiente (KIONIX, 2012). São sensores que usam um par de capacitores diferencial e são construídos com o eixo sensível no plano do dispositivo. Outra vantagem desses acelerômetros baseados em sensores capacitivos diferencial é um sistema eletrostático de *force-feedback* que pode ser usado para evitar não-linearidades nas molas e capacitores, mantendo a posição de massa quase fixa.

Essencialmente, o circuito de *force-feedback* mede um pequeno deslocamento da massa de prova, altera a sua estimativa da força de aceleração externa com base na equação para a mola e, em seguida, aplica-se a  $V_{\text{massa}}$  (tensão na massa de prova) necessária para restaurar a massa para a sua posição de repouso. A tensão de saída é linearmente proporcional à força eletrostática aplicada, como este é igual à força de aceleração em equilíbrio. O método de *force-feedback* mantém deslocamentos da massa extremamente pequenos para minimizar não-linearidades mecânicas e também permite uma equação ser linearizada.

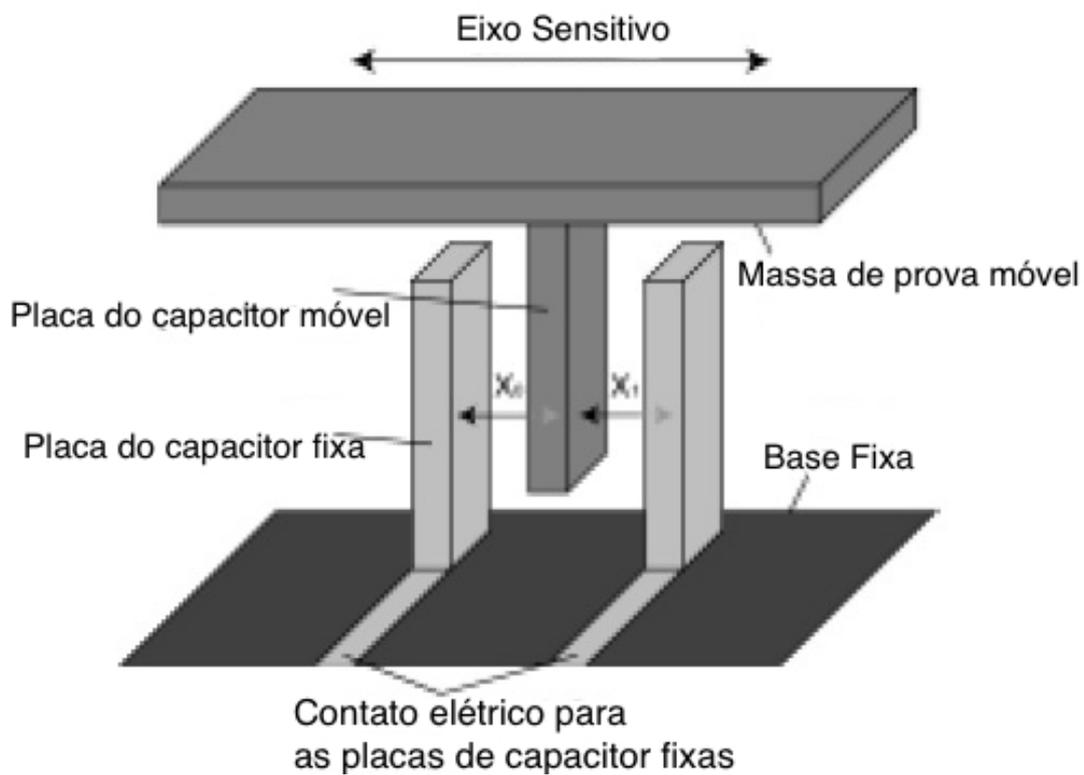


Figura 3: Esquema de um acelerômetro capacitivo diferencial.  
Fonte: Adaptado de Graham (2000).

## 5 BLUETOOTH®

Segundo Kansal, citado por Zhang et al. (2011), Bluetooth® é um padrão de tecnologia de comunicação sem fio de pequeno alcance que usa banda de rádio livre de licença e universal – Industrial, Scientific and Medical band (ISM) – na frequência de 2,4 GHz. Bluetooth tem uma grande capacidade contra interferências, pois atua em um espectro espalhado por meio de um procedimento chamado “Salto em Frequência Adaptativo” (Adaptive Frequency-hopping - AFH) como solução para evitar interferências de outras tecnologias operando na banda ISM de 2,4 GHz, como telefones sem fio, fornos de micro-ondas, WLAN, USB wireless e Zigbee.

O Bluetooth pode ser integrado em quase qualquer equipamento digital. Ele apresenta as vantagens do baixo consumo, característica importante para pequenos aparelhos, robustez e baixo custo. Ainda, o Bluetooth habilita conexões ad hoc temporárias, que não necessitam de roteadores ou estações base sem fio. Quando um dispositivo Bluetooth entra nas proximidades de rádio de outro dispositivo Bluetooth uma Piconet é estabelecida dinamicamente e automaticamente.

De acordo com Bluetooth® (2010, Vol. 1, pag. 17), existem duas formas de sistemas com a tecnologia Bluetooth: *Basic Rate* (BR) e *Low Energy* (LE). Os dois sistemas incluem descobrimento de dispositivo, estabelecimento de conexão e mecanismos de conexão. O sistema *Basic Rate* (Taxa Básica) inclui extensões de camadas opcionais como: *Enhanced Data Rate* (EDR), *Alternate Media Access Control* (MAC) e *Physical* (PHY).

Sistemas BR oferecem conexões síncronas e assíncronas com taxa de dados de 721,2 kbps para *Basic Rate*, 2,1 Mbps para *Enhanced Data Rate* e operação em alta velocidade em até 24 Mbps com 802.11 AMP (Bluetooth 3.0 + HS – *High Speed*).

Sistemas LE apresentam características destinadas a produtos que requerem baixo consumo de corrente, baixa complexidade e baixo custo. São projetados para serem utilizados em aplicações de baixa taxa de dados e têm baixo *duty cycles*.

## 5.1 VERSÕES DO BLUETOOTH®

De acordo com Alecrim (2013) o Bluetooth é uma tecnologia que está evoluindo constantemente, com isso surgem novas versões com melhores especificações com o decorrer do tempo.

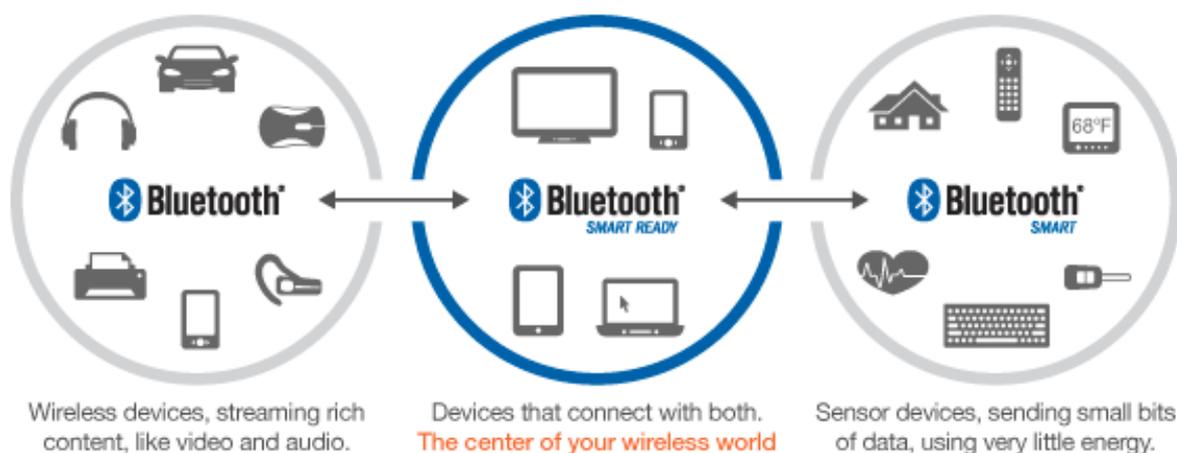
As primeiras versões foram a 1.0, 1.1 e 1.2, lançadas em 1999, 2001 e 2003 respectivamente, com a utilização do *Basic Rate*. A partir da versão 2.0 lançada em novembro de 2004 é que se pôde opcionalmente utilizar a versão com BR/EDR alcançando velocidades de transmissão maiores. Em 2007, lançou-se a versão 2.1 com algumas melhorias. Em 2009, veio a versão 3.0 + HS, como principal melhoria a alta taxa de transferência de dados, chegando a 24 Mbps, e um controle melhor do gasto de energia exigido para as conexões. Em 2010, lançou-se a versão 4.0, que usa o sistema *Low Energy* (LE), com principal enfoque na economia de energia. E em dezembro de 2013 foi lançada a versão 4.1.

Isso prova que a tecnologia Bluetooth está evoluindo muito e a tendência é que ela seja utilizada por muitos aparelhos portáteis de baixo consumo.

É importante destacar que o fato de ter várias versões não significa que um equipamento com uma especificação mais recente não funcione com outro com uma versão anterior, embora possa haver exceções.

## 5.2 LOGOS BLUETOOTH®

Segundo Bluetooth SIG (2013a), para entender a compatibilidade entre diversos dispositivos que apresentam as diversas versões de Bluetooth, utilizam-se três logos, conforme a Figura 4 abaixo:



**Figura 4: Logos Bluetooth®.**  
**Fonte: (BLUETOOTH SIG, 2013a).**

Equipamentos com o logo “Bluetooth®” possuem a versão tradicional já utilizada por muitos aparelhos sem fio, que fazem streaming de áudio e vídeo e altas transferências dados. Ela utiliza o sistema BR/EDR e corresponde a versões inferiores à versão 4.0.

Equipamentos com o logo “Bluetooth® Smart” usam a versão 4.0 ou posterior com o sistema LE. Aplicado para a nova geração de sensores (*gadgets*) que duram por meses a anos sem precisar de recarga.

Equipamentos com o logo “Bluetooth®” não são compatíveis com dispositivos “Bluetooth® Smart”.

Então, surgem os aparelhos com o logo “Bluetooth® Smart Ready” que usam a versão 4.0 ou posterior e que se comunicam com todas as diversas versões de Bluetooth. Estão presentes em *smartphones*, *tablets*, PCs, TVs. Pensando na compatibilidade futura, é possível atualizar o *software* desses dispositivos para que eles operem com os mais recentes *gadgets*.

Os mais recentes *smartphones* e *tablets* estão sendo fabricados com Bluetooth 4.0 que usa tecnologia “Bluetooth® Smart Ready”. Isso indica que no futuro, praticamente todos os dispositivos serão compatíveis com as diversas versões de Bluetooth (BR/EDR, LE e posteriores).

### 5.3 BLUETOOTH® LOW ENERGY (BLE)

Para o desenvolvimento do projeto em questão optou-se pela tecnologia Bluetooth® Low Energy.

O Bluetooth® Low Energy, também conhecido como Bluetooth® 4.0 ou Bluetooth® Smart, apresenta como sua principal característica a economia de energia. Ele tem como propósito transmitir pacotes de dados bem pequenos, consumindo menos energia que o BR/EDR. Essa economia é notada quando o dispositivo está ocioso, muito útil para telefones celulares que consomem muita energia quando o Bluetooth está ativo, mas não está sendo utilizado. Essa tecnologia tem como principal aplicação dispositivos portáteis, denominados *gadgets*, alimentados por pequenas baterias e que podem durar por anos sem serem recarregados. Esses *gadgets* normalmente são sensores que coletam dados específicos, como nível de glicose do sangue, frequência do batimento cardíaco, temperatura, pressão, entre outros, e transmitem para um dispositivo que contém uma aplicação, como um aplicativo em *smartphone*.

De acordo com Bluetooth® (2010), o rádio LE, como o rádio BR/EDR, opera na banda de frequência livre de 2,4 GHz ISM e apresenta um *transceiver* com *frequency hopping* para combater interferências e atenuações. A operação do rádio LE usa uma modulação em frequência binária mais simples para minimizar a complexidade do *transceiver* e conseqüentemente o consumo. A taxa de símbolo é de 1 Mega símbolo por segundo suportando uma taxa de 1 Mbps.

Segundo Bluetooth SIG (2013b), Sistemas Operacionais de dispositivos móveis incluindo iOS, Android, Windows Phone e BlackBerry, assim como OSX e Windows 8, suportarão nativamente o Bluetooth® Low Energy. A previsão é que em 2018 mais de 90% dos dispositivos *smartphones* que tiverem Bluetooth suportarão o padrão Low Energy.

#### 5.3.1 Por que BLE?

De acordo com a Texas Instruments (2013a), dentre as ferramentas *wireless* que possui, as tecnologias que tem como aplicação *sport* e *fitness* são BLE, ANT+ e

Sub 1-GHz (CC430). Dentre esses três o BLE e o ANT+ são os mais recomendados para aplicações em *smartphone*. Segundo esse documento, o BLE apresenta uma maior eficiência em termos de consumo de corrente do que o ANT+.

ANT e ANT+ tem muitas aplicações em *sport* e *fitness* mas como se encontrou um hardware pronto e com uma pilha de protocolos pronta, escolheu-se o BLE para dar mais agilidade no processo de desenvolvimento.

Segundo Kate (2013), o Bluetooth tende a ser mais abrangente do que o ANT+. Mais fabricantes estão optando pelo BLE do q pelo ANT+. Em alguns dispositivos que não tem suporte para ANT+ é necessário um *dongle* adicional para se conectar com dispositivos ANT+.

No endereço do site da ANT+ (2014) é possível obter uma lista dos dispositivos móveis que operam com ANT+. Sistemas IOS necessitam de um *dongle* ANT+.

Smith (2011) fez uma comparação entre ANT e BLE, a respeito do tempo de duração de uma bateria do tipo moeda de lítio de 225 mAh para uma taxa de 120 Bytes/s. A média de corrente para o ANT foi de 175,5  $\mu$ A, resultando num tempo de bateria de 52,64 dias. Para BLE, a média de corrente foi de 49  $\mu$ A, resultando numa duração de bateria de 191 dias.

Ainda, de acordo com Smith (2011), o BLE apresenta maior coexistência do que ANT+. Ou seja, o BLE tem maior habilidade de operar juntamente com outros aparelhos de radio frequência sem sofrer interferência. Muito disso se deve à utilização do esquema de saltos em frequência, enquanto que o ANT+ opera normalmente em um canal de frequência, sendo mais susceptível a interferências.

## 6 APLICATIVO

### 6.1 O SISTEMA IOS

O iOS (antes chamado de iPhone OS) é um sistema operacional móvel da Apple Inc. desenvolvido originalmente para o iPhone, também é usado em iPod touch, iPad e Apple TV.

O sistema operacional foi apresentado com o iPhone na Macworld Conference & Expo em 9 de janeiro de 2007, e lançado no mês de junho. Inicialmente, as aplicações de terceiros não eram permitidas. Steve Jobs argumentou que os desenvolvedores poderiam criar aplicativos na web que "se comportam como aplicações nativas no iPhone". Em 17 de outubro de 2007, a Apple anunciou que a SDK nativa estava em desenvolvimento e que eles esperassem para colocá-la nas "mãos dos desenvolvedores". (WIKIPEDIA IOS, 2014)

Hoje essa ferramenta está disponível e inclusive toda sua documentação. Ainda assim, muitas funcionalidades são restritas para o desenvolvedor amador, como testar o aplicativo "*on device*", ou seja, testar no próprio aparelho. Para testar funcionalidades como interação em redes sociais, e-mail e *features*, tal como o uso do acelerômetro existente no smartphone, é necessário desembolsar 99 dólares para uma assinatura anual de desenvolvedor, ou partir para o uso não recomendado do Jailbreak.

Os aplicativos para esse sistema operacional devem ser escritos e compilados especificamente para iOS e a arquitetura ARM de 64 bits. Ser escrito para iOS quer dizer que se deve desenvolver em linguagem Objective-C e compilado no XCode, que é o ambiente de desenvolvimento, onde se pode simular o aplicativo desenvolvido.

Como iOS usa uma variante do mesmo kernel XNU que é encontrada no OSX, o conjunto de ferramentas de *software* utilizado para o desenvolvimento em iOS também é baseado no Xcode.

O conteúdo do SDK é dividido nos seguintes conjuntos:

- Cocoa Touch
  - Multi-touch events and controls
  - Accelerometer support

- View hierarchy
- Localization (i18n)
- Camera support
- Media
  - OpenAL
  - audio mixing and recording
  - Video playback
  - Image file formats
  - Quartz
  - Core Animation
  - OpenGL ES
- Core Services
  - Networking
  - Embedded SQLite database
  - Core Location
  - Threads
  - CoreMotion
- Mac OS X Kernel
  - TCP/IP
  - Sockets
  - Power management
  - File system
  - Security

### 6.1.1 Frameworks

As frameworks são classes que provêm a interface para o desenvolvimento/escrita do software na plataforma iOS.

Entre as diversas frameworks disponíveis no SDK para iOS utilizamos para o aplicativo do projeto: Core Bluetooth, MessageUI, Core Foundation, Core Graphics, UIKit e XCTest. Dentre essas, Core Bluetooth foi o que nos demandou maior atenção e estudo.

Core Bluetooth fornece as classes necessárias para que os aplicativos iOS e Mac se comuniquem com dispositivos que são equipados com a tecnologia BLE.

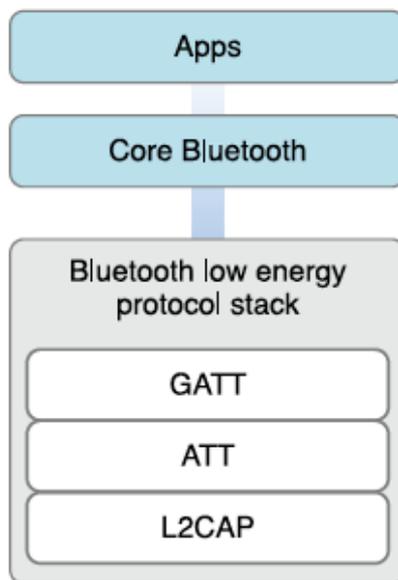


Figura 5: Abstração do relacionamento App-BLE.  
Fonte: (APPLE, 2013).

Em comunicação BLE, existem dois principais *players*: o central e os periféricos. Cada player tem um papel diferente a desempenhar na BLE. Um periférico tipicamente fornece dados a outros dispositivos. O central, geralmente utiliza as informações de um periférico para realizar alguma tarefa.

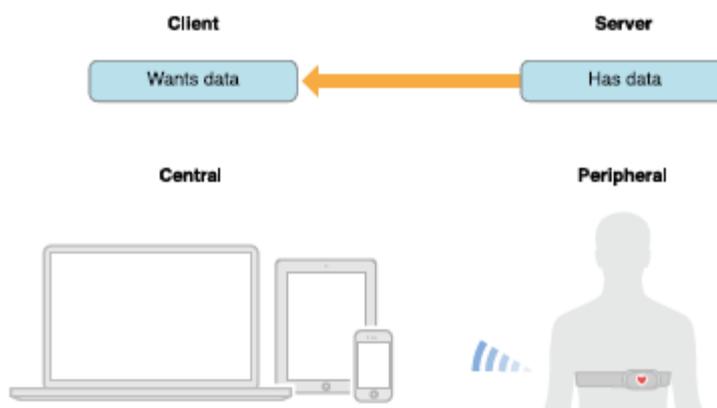


Figura 6: Relacionamento Central – Periférico.  
Fonte: (APPLE, 2013)

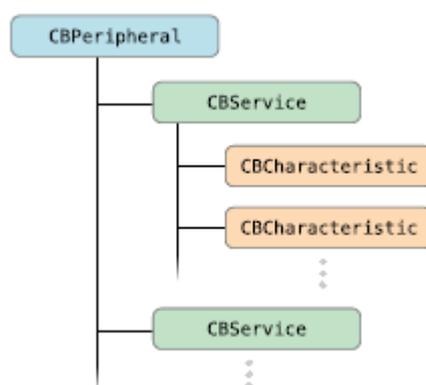
Dessa forma o Core Bluetooth trabalha com a pilha de protocolo BLE implementando métodos que tem como parâmetros as UUIDs, que falaremos posteriormente.



**Figura 7: Periférico envia *advertising* ao Central.**  
Fonte: (APPLE, 2013).

O *player* central procura estabelecer conexão com o periférico, depois de estabelecer com êxito uma conexão, ele pode descobrir a gama completa de serviços e características que o periférico tem a oferecer (dados de *advertising* podem conter apenas uma fração dos serviços disponíveis).

O central também pode interagir com o serviço de um periférico, lendo ou escrevendo o valor da característica desse serviço.



**Figura 8: Métodos da framework Core Bluetooth.**  
Fonte: (APPLE, 2013).

Os serviços e características de um periférico são identificados por UUIDs de 128 bits específicos do Bluetooth, que são representados por objetos CBUUID na framework Core Bluetooth. Embora nem todos os UUIDs que identificam um serviço ou característica são predefinidos pelo Grupo de Interesse Especial Bluetooth (SIG – sigla em inglês para Bluetooth Special Interest Group), Bluetooth SIG definiu e publicou uma série de UUIDs comumente usados que foram encurtados para 16-bits por conveniência.

## 7 PROCEDIMENTOS METODOLÓGICOS

Com o embasamento teórico apresentado nos capítulos anteriores, desenvolveu-se o projeto com o objetivo de medir golpes aplicados em um saco de pancadas, utilizando o acelerômetro presente no SensorTag que transmite os dados via BLE para um aplicativo no iPhone. Os dados obtidos pelo iPhone são processados numa Macro em Excel que informa o relatório dos golpes.

### 7.1 SENSORTAG

O hardware escolhido para se desenvolver o projeto foi o Kit SensorTag (CC2541DK-SENSOR) da Texas Instruments. Ele contém o CI CC2541 e 6 sensores MEMS como giroscópio, acelerômetro, sensor de umidade e temperatura, barômetro, magnetômetro, sensor de temperatura por infravermelho, e uma bateria para desenvolvimento de aplicações BLE em Smartphones. Segundo Araujo (2013), ele é considerado o primeiro kit de desenvolvimento BLE voltado para aplicações em *smartphones e tablets*.

A Figura 9 mostra uma foto do SensorTag.



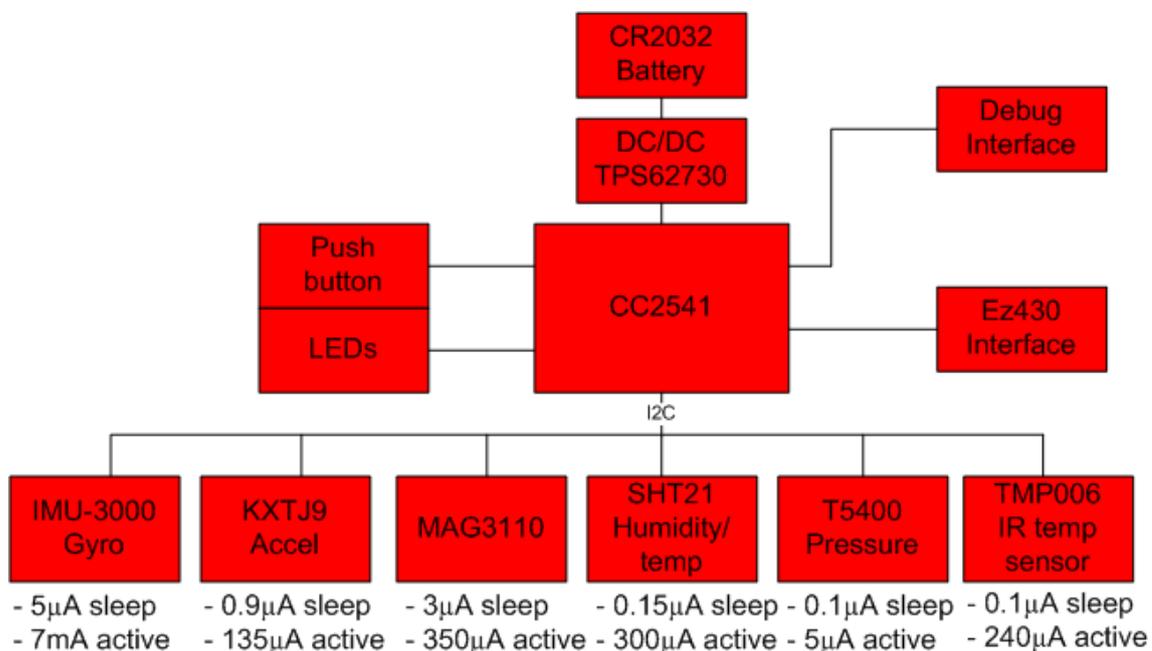
**Figura 9: SensorTag da Texas Instruments.**  
**Fonte: (ARAUJO, 2013)**

### 7.1.1 Principais Componentes

A seguir são listados os sensores presentes no SensorTag:

- Sensor de Temperatura por Infra-vermelho (Texas Instruments TMP006)
- Sensor de Humidade (Sensirion SHT21)
- Giroscópio (Invensense IMU-3000)
- Acelerômetro (Kionix KXTJ9)
- Magnetômetro (Freescale MAG3110)
- Barômetro (Epcos T5400)
- Sensor de temperatura On-chip (Integrado no CC2541)
- Sensor de tensão da Bateria (Integrado no CC2541)

Todos os 6 sensores principais usam a interface de comunicação serial I2C para trocar dados com o CC2541. A Figura 10 mostra o diagrama de blocos do SensorTag.



**Figura 10: Diagrama de Blocos do SensorTag.**  
**Fonte: (TEXAS INSTRUMENTS WIKI, 2013).**

Na Figura 11, a seguir, mostra-se como é estruturada a placa do SensorTag com seus principais componentes.

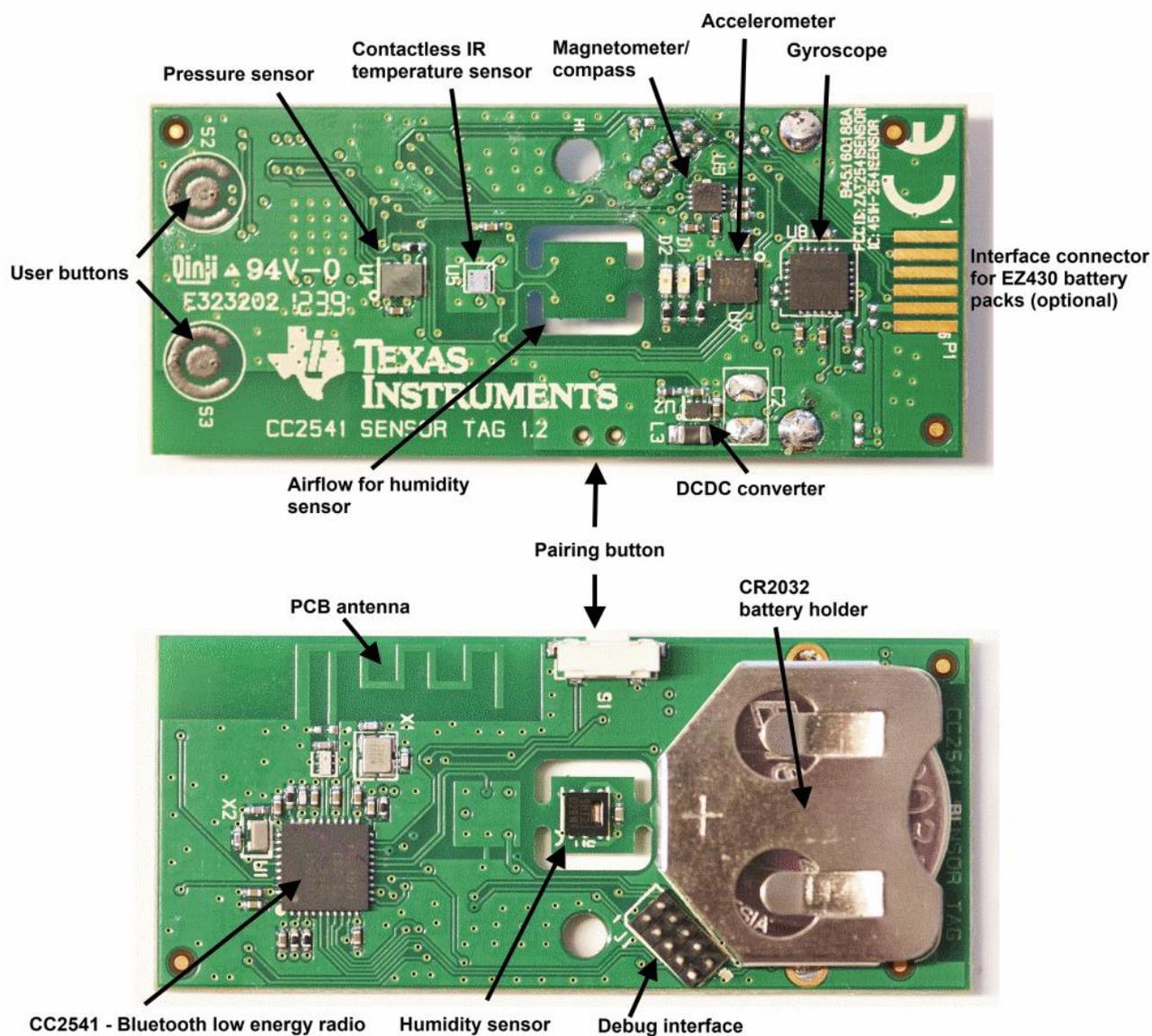


Figura 11: Estrutura da placa do SensorTag.  
Fonte: (TEXAS INSTRUMENTS WIKI, 2013).

### 7.1.2 Alimentação Elétrica

A alimentação elétrica do SensorTag é feita por uma bateria de lítio do tipo moeda (CR2032). Como ele consome pouca corrente, pode durar por meses com apenas uma bateria.

### 7.1.3 Conversor DC/DC

É o chip TPS62730, usado especificamente para aplicações *wireless ultra low power*. Com ele é possível aumentar o tempo de duração da bateria em 20%, pois reduz o consumo de corrente durante os modos de transmissão e recepção do rádio BLE, utilizando uma conversão de tensão altamente eficiente.

### 7.1.4 CC2541

O CC2541 é um sistema em um chip otimizado para aplicações Bluetooth de baixo consumo. Composto de:

- Núcleo 8051 de baixo consumo e fácil programação através do software livre de *royalties* fornecido pela Texas Instruments.
- Sistema de RF 2,4 GHz, totalmente compatível com BLE.
- Memória *Flash* de 256 kB programável *in-system*.
- 8 kB de memória RAM.
- Diversos periféricos como ADC 12 Bits, *timers*, coprocessador AES, DMA, interface serial, entre outros.

A Figura 12 mostra um diagrama de blocos com todos os periféricos.

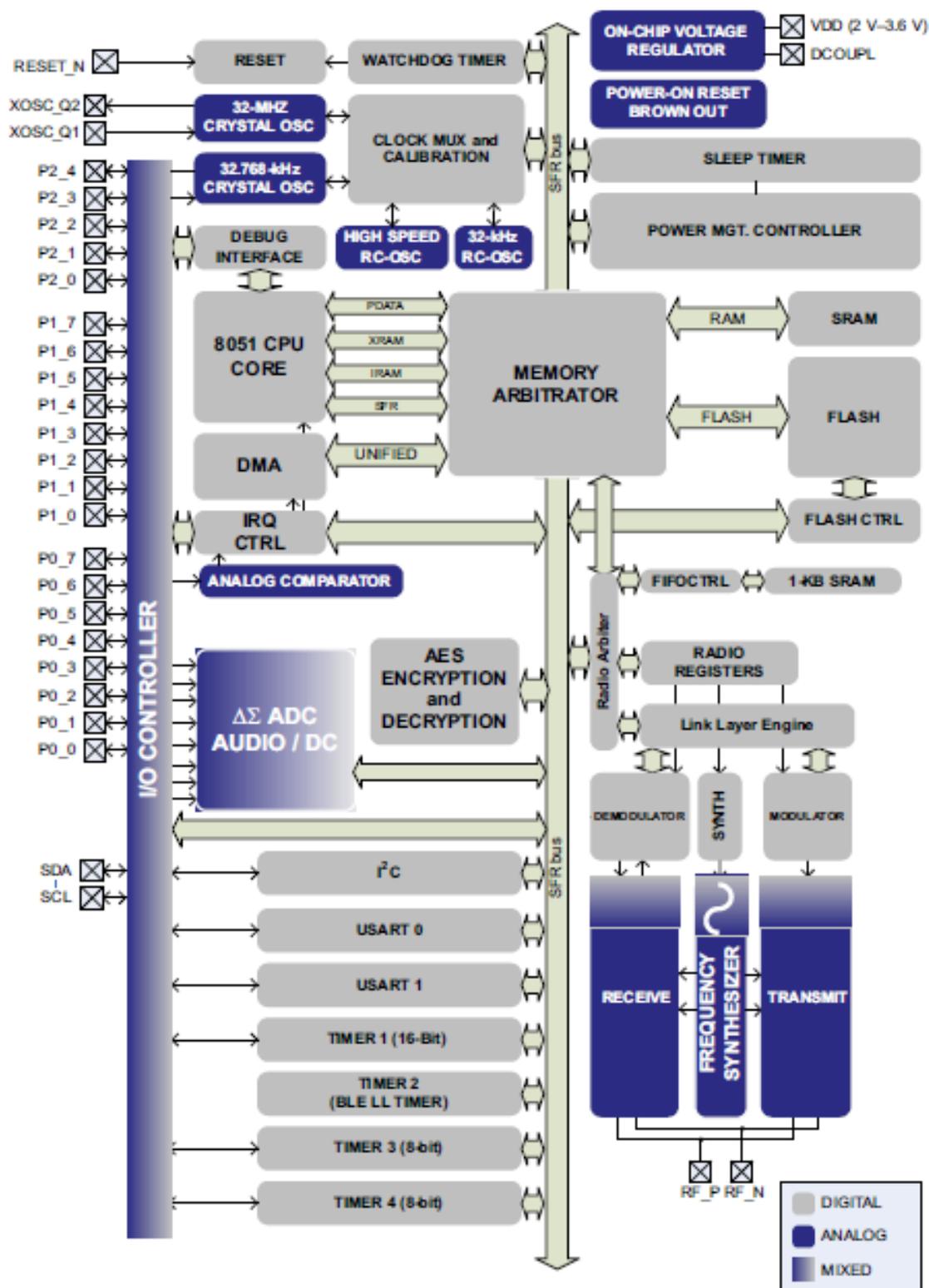


Figura 12: Diagrama de Blocos do CC2541.  
 Fonte: CC2541 datasheet (TEXAS INSTRUMENTS, 2012).

Devido ao alto grau de integração presente no CC2541 é possível implementar todo o sistema com poucos componentes externos, conforme ilustra a Figura 13 com o esquemático.

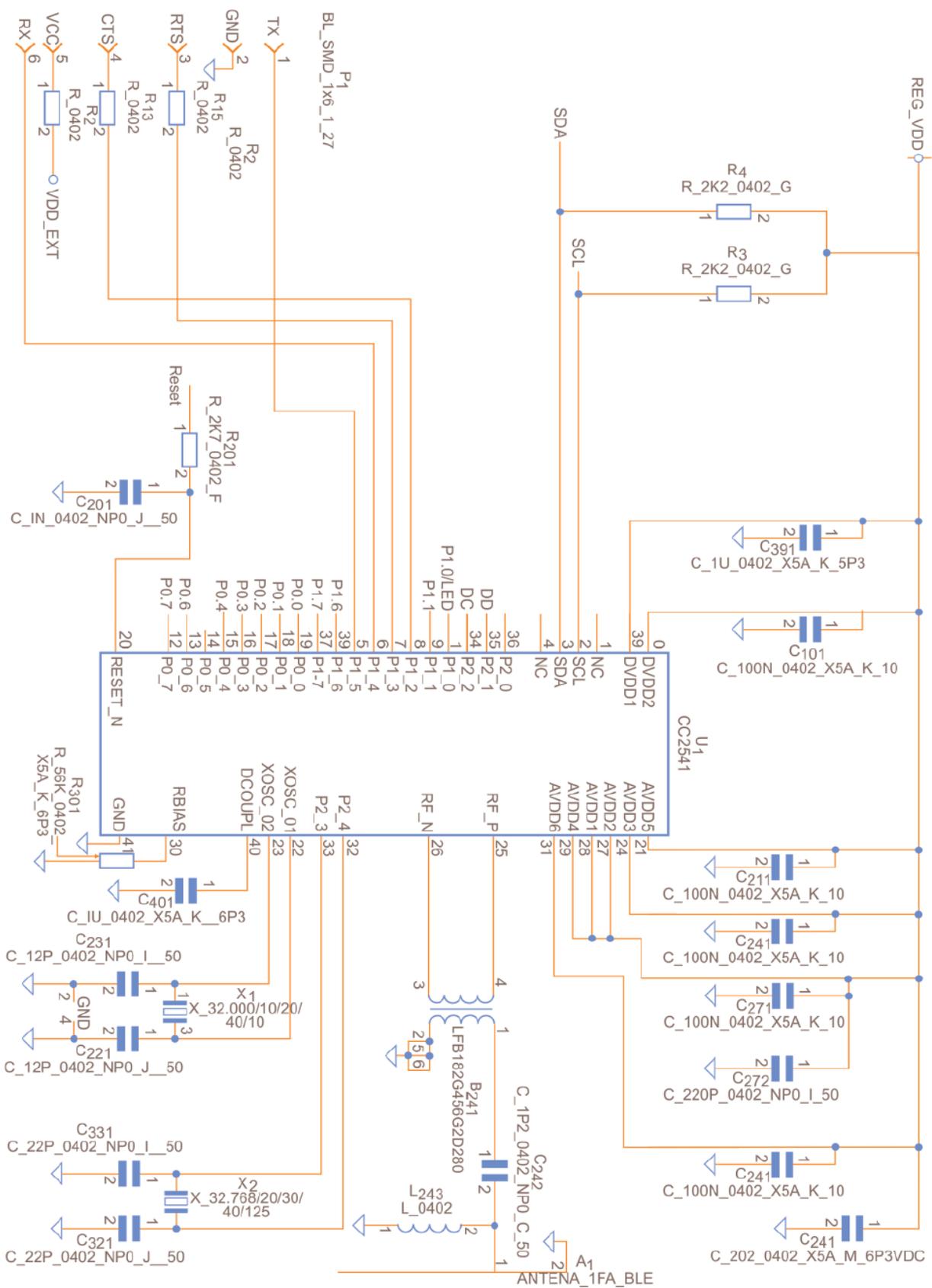


Figura 13: Esquemático do CC2541.  
 Fonte: (ARAUJO, 2013).

### 7.1.5 Acelerômetro KXTJ9

Para a aplicação do projeto, apenas se utilizou o acelerômetro como sensor. A Figure 14 mostra o esquemático dele com as ligações na placa.

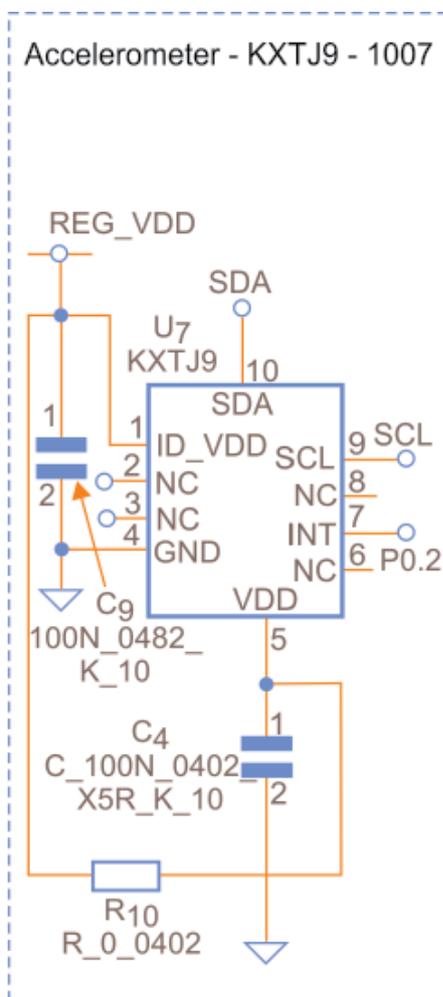


Figure 14: Esquemático do Acelerômetro KXTJ9.  
Fonte: (ARAUJO, 2013).

O KXTJ9, usado no projeto, é um acelerômetro de silício micro usinado tri axial com níveis de sensibilidade  $\pm 2g$ ,  $\pm 4g$  ou  $\pm 8g$ . Para a aplicação proposta configurou-o para operar em  $\pm 8g$ . O elemento sensor é fabricado usando a tecnologia de processo de micro-usinagem com plasma da Kionix. Reguladores de tensão são usados para manter constantes tensões operacionais internas na faixa de tensões de alimentação de entrada. Isso resulta em características operacionais estáveis na faixa de tensão de alimentação de entrada e erro de offset virtualmente indetectável. O protocolo digital I2C é usado para se comunicar com o microcontrolador e configurar as saídas (KIONIX, 2012).

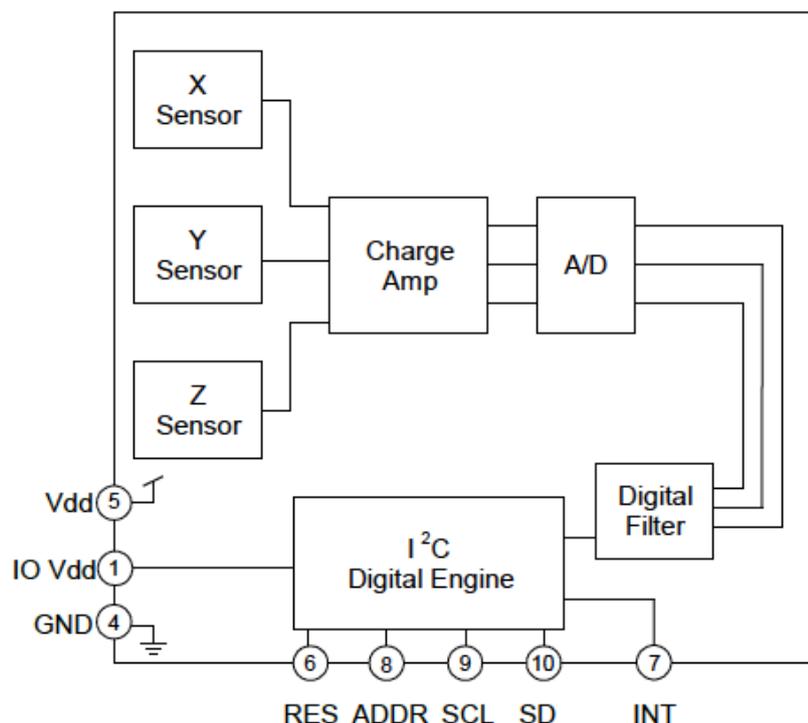


Figura 15: Diagrama funcional do KXTJ9.  
Fonte: (KIONIX, 2012).

## 7.2 BLE STACK

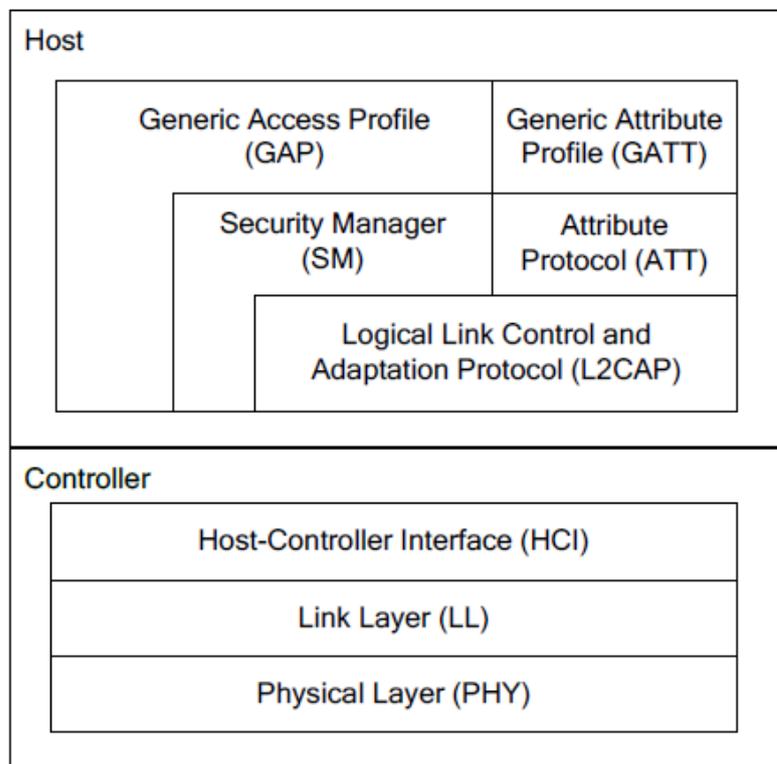
O SensorTag já vem com um firmware pré-programado com a aplicação SensorTag. A Texas Instruments fornece um pilha de protocolos (BLE Stack) que contém todo o software necessário para se iniciar o desenvolvimento com aplicações BLE *single-mode*. A utilizada no projeto foi a BLE-Stack 1.4.0 disponível em Texas Instruments (2013b). Essa pilha é atualizada com o passar do tempo com novas melhorias e é livre de *royalties*.

BLE-stack 1.4.0 inclui códigos objetos com a mais recente pilha de protocolos Bluetooth Low Energy, suportando múltiplas conexões, apresentando exemplos de projetos e aplicações, abordando uma variedade de perfis com o código fonte. Também apresenta muitas documentações para auxiliar no desenvolvimento.

Essa pilha é acessada através da IDE IAR Embedded Workbench para 8051 versão 8.20.

### 7.2.1 Pilha de Protocolos BLE

A pilha de protocolos BLE é ilustrada a seguir na Figura 16:



**Figura 16: Pilha de Protocolos BLE.**  
**Fonte: (TEXAS INSTRUMENTS, 2013c).**

A pilha de protocolos consiste de duas seções: o *Controller* e o *Host*. Essa divisão é característica do padrão Bluetooth tradicional, em que as duas seções são implementadas separadamente. Qualquer perfil ou aplicação é feito no topo das camadas GAP e GATT da pilha.

A camada PHY (Camada Física) consiste na operação de RF a 1 Mbps utilizando FHSS GFSK (*Gaussian Frequency-Shift Keying*) na banda de 2,4 GHz ISM.

A camada Link Layer (LL) controla os estados de RF do dispositivo, que podem ser: *standby*, *advertising*, *scanning*, *initiating* ou *connected*.

*Advertisers* transmitem dados sem estarem em uma conexão, enquanto os *scanners* escutam por *advertisers*. Um *initiator* é um dispositivo que responde a um *advertiser* com uma requisição de conexão (*connection request*). Se o *advertiser* aceitar, os dois, *advertiser* e *initiator*, entram no estado conectado (*connected*). Quando um dispositivo está conectado, ele pode ser considerado em uma das duas

funções: *master* ou *slave*. O dispositivo que iniciou a conexão (*initiator*) se torna o *master*, e o que aceitou a requisição de conexão se torna *slave*.

HCI (*Host-Controller Interface*) é a camada que provê o meio de comunicação entre o *Host* e o *Controller* via interface padrão, implementada por uma API de software ou por uma interface de hardware, como UART, SPI ou USB.

A camada L2CAP (*Logical Link Control and Adaptation Protocol*) faz os serviços de encapsulamento de dados para as camadas superiores, permitindo uma comunicação de dados lógicos fim-a-fim.

A camada SM (*Security Manager*) define métodos de pareamento e distribuição de chave, além de funções para outras camadas de modo que se conecte e troque dados com outro dispositivo seguramente.

A GAP (*Generic Access Profile*) faz a interface direta com a aplicação e perfis, controla o descobrimento de dispositivos e serviços de conexão. Além disso, controla a inicialização de características de segurança.

O protocolo ATT (*Attribute Protocol*) permite expor certos pedaços de dados (*attributes*) para outro dispositivo. O dispositivo que expõe *attributes* é o servidor (*server*) e o dispositivo ponto (*peer device*) é o cliente (*client*). É importante frisar que o estado da camada LL (*master* e *slave*) de um dispositivo é independente das funções ATT (*server* e *client*).

A camada GATT (*Generic Attribute Profile*) é o serviço de framework que define sub-processos para usar a camada ATT. Ela especifica a estrutura dos perfis. No BLE, todos os pedaços de dados que são usados por um perfil ou serviço são chamados de "*characteristics*". Todos os dados de comunicação entre dois dispositivos passa por sub-processos GATT. Portanto, a aplicação e perfis usam diretamente o GATT.

### 7.2.2 Terminologias do GATT

Todos os dispositivos BLE usam GATT. A API oferecida pelo dispositivo operando com BLE será tipicamente baseada nos conceitos do GATT. A seguir apresentam-se suas terminologias:

**Client (Cliente):** dispositivo que inicia comandos e requisições GATT, e aceita respostas, por exemplo, um computador ou *smartphone*.

**Server (Servidor):** dispositivo que recebe comandos e requisições GATT, e retorna respostas, por exemplo, um sensor de temperatura.

**Characteristic (Característica):** valor de dado transferido entre cliente e servidor, por exemplo, a tensão atual da bateria.

**Service (Serviço):** conjunto de características relacionadas, que operam juntas para executar uma função específica. Por exemplo, um serviço do acelerômetro que inclui características para o valor de uma medição de aceleração e período de tempo entre cada medição.

**Descriptor (Descritor):** fornece informação adicional sobre uma característica. Por exemplo, um valor de característica de temperatura pode ter uma indicação de sua unidade (Celsius), e os valores de máximo e mínimo que o sensor pode medir. Descritores são opcionais, cada característica pode ter quantos descritores achar necessário.

Ressalta-se que alguns serviços e valores de características podem ser usados com propósitos administrativos.

### 7.2.3 Identificadores

*Services, characteristics e descriptors* são coletivamente referenciados como *attributes* (atributos), e identificados por UUIDs (*Universally Unique Identifier*). O Bluetooth SIG reservou uma faixa de UUIDs. Para o SensorTag, o UUID base da Texas Instruments é F000xxxx-0451-4000-B000-000000000000, cada valor é representado em hexadecimal e os “x” são usados para identificar cada *attribute*.

### 7.2.4 Operação do GATT

O protocolo GATT fornece vários comandos para o cliente descobrir informações sobre o servidor. Podemos citar:

- Descobrimto do UUID para todos os serviços primários;

- Encontrar o serviço com um UUID fornecido;
- Encontrar serviços secundários para um serviço primário fornecido;
- Descobrir todos os *characteristics* para um serviço específico;
- Encontrar *characteristics* para um UUID específico.

Comandos de leitura (dados transferidos do servidor para o cliente) e de escrita (do cliente para o servidor) de valores de *characteristics* podem ser usados.

Leituras podem ser feitas usando o UUID de *characteristics* ou usando o *handle* (outro identificador).

Ainda, o GATT oferece *notifications* (notificações) e *indications* (indicações). O servidor pode enviar valores de *characteristics* para o cliente quando desejar através de notificações e indicações. As notificações não requerem uma resposta de recebimento por parte do cliente, enquanto que indicações requerem uma resposta de confirmação de recebimento. Para se usar notificações é preciso habilitá-las em determinados serviços por meio de um *Characteristic Value Write* (escrita num valor de característica). Isso evita que o cliente fique fazendo *polling* (leituras periódicas) no servidor à espera de um dado.

### 7.2.5 Operações do BLE

No BLE empregam-se dois esquemas de acesso múltiplo: *Frequency division multiple access* (FDMA) – acesso múltiplo de divisão de frequência, e *time division multiple access* (TDMA) – acesso múltiplo de divisão de tempo. Quarenta (40) canais físicos, separados por 2 MHz, são usados no FDMA. Três (3) são usados como canais de *advertising* e 37 como canais de dados. Um esquema de *polling* baseado no TDMA é usado da seguinte maneira: um dispositivo transmite um pacote em um tempo predeterminado, após um intervalo de tempo também predeterminado, um dispositivo correspondente responde com um pacote. O canal físico é subdividido em unidades de tempo conhecidas como “Eventos”. Dados são transmitidos entre dispositivos por meio de pacotes que são posicionados nesses eventos.

Existem dois tipos de eventos: Eventos de *Advertising* e Eventos de Conexão (*Connection Events*).

Eventos de *Advertising* são usados para estabelecer a conexão entre dois dispositivos. Depois de estabelecida a conexão é que ocorrem os eventos de conexão.

*Connection Events* são usados para transmitir pacotes de dados entre o *master* e o *slave*. Saltos em frequência ocorrem no início de cada evento de conexão. Num evento de conexão, *master* e *slave* alternam envios de pacotes de dados usando o mesmo canal físico de dados. O *master* inicia o começo de cada evento de conexão e pode encerrar a qualquer momento. A Figura 17 mostra como ocorre esse processo.

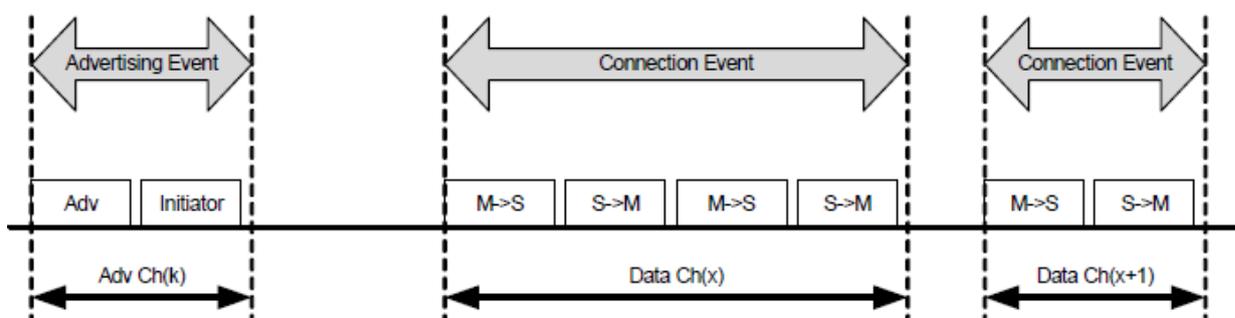


Figura 17: Esquema de Eventos no BLE.  
Fonte: (BLUETOOTH®, 2010).

Dispositivos em uma piconet usam um padrão de saltos em frequência, que é determinado por um algoritmo indicado num campo de um *connection request* enviado pelo *initiator*. O *initiator* fornece a referência de sincronização, conhecida como hop interval (intervalo de salto). Esse padrão de saltos usado no BLE é uma ordenação pseudorrandômica das 37 frequências da banda ISM. Ele pode ser adaptado para excluir porções de frequência que são usadas por dispositivos que causam interferência. A técnica de saltos adaptativos permite que o BLE coexista com outros sistemas que operam na banda ISM de forma estática (*non-hopping*).

### 7.3 OPERAÇÃO DO SENSORTAG

O SensorTag é usado como um dispositivo periférico (*slave*) em uma conexão BLE. Ele apresenta vários sensores e possui uma solução de software para os drivers desses sensores que são interfaceados por meio de um servidor GATT

que roda na TI BLE Stack. O servidor GATT contém o serviço primário de cada sensor para configuração e coleta de dados.

Na inicialização, ao se pressionar o botão de liga/desliga na lateral do SensorTag, este faz *advertising* com intervalos de 100ms. A conexão é estabelecida por um dispositivo central (Iphone 4s), então, os sensores podem ser configurados para medir dados. O dispositivo central deve ser capaz de:

- Escanear e descobrir o SensorTag. A resposta do *scan* contém o nome “SensorTag”;
- Estabelecer conexão baseada nos parâmetros de conexão definidos pelo usuário;
- Executar o descobrimento de serviços – descobrir *characteristics* pelo UUID;
- Operar como um cliente GATT – ler e escrever por meio de valores de *characteristics*.

O dispositivo central deve iniciar a conexão e portando se torna o *master*.

Para obter o dado de um sensor, este deve ser ativado, o que é feito por meio de um *Characteristic Value Write* (escrita num valor de característica) de um serviço apropriado.

Cada sensor do SensorTag apresenta um Serviço Primário GATT correspondente que contém vários *characteristics*. Um deles é o período, que define o intervalo de tempo em que são feitas as medições do sensor. Podem-se habilitar notificações para cada um dos sensores. Ao habilitá-las, o SensorTag (servidor) envia para o Iphone (cliente) os dados de cada nova medição do sensor por meio de notificações, a cada intervalo de tempo determinado pelo período do Serviço Primário GATT desse sensor.

A forma mais eficiente de coletar medições de sensores, considerando economia de energia, é:

- Habilitar a notificações;
- Habilitar o sensor;
- Quando uma notificação com dados é obtida pelo *master* (Iphone), desabilita-se o sensor.

As notificações permanecem ativas, de modo que toda vez que concluir o período de notificação de um dado sensor, este é habilitado, seu valor é lido, colocado para envio via BLE e então desabilitado.

A Figura 18 a seguir, mostra os papéis de cada dispositivo numa conexão BLE.



**Figura 18: Papéis do iPhone e do SensorTag numa conexão BLE.**  
Fonte: Autoria própria.

### 7.3.1 Camada de Sistema Operacional Abstrato (OSAL)

O funcionamento do software embarcado no SensorTag é baseado numa Camada de Sistema Operacional Abstrato (Operational System Abstraction Layer - OSAL) que faz o papel de um sistema operacional, porém não se é de fato. Ele opera num loop passando por diversas tarefas, ordenadas por prioridades, em que trata eventos presentes em cada tarefa. As tarefas de maior prioridade são as da pilha de protocolos do BLE, em que o índice menor representa a maior prioridade. A aplicação é a tarefa de menor prioridade.

A Figura 19 mostra as tarefas e seus índices de prioridade.

ID	Task
0	Link Layer
1	HAL
2	HCI
3	OSAL Callback Timer
4	L2CAP
5	GAP
6	GATT
7	SM
8	GAP Role
9	GAP Bond Manager
10	GATT Server
11	Application

**Figura 19: Tarefas do software embarcado e suas respectivas prioridades.**  
**Fonte: (STEWART, 2013).**

O fluxograma da Figura 20 mostra como ocorre esse loop do OSAL. Ao iniciar o sistema, as funções de inicializações da placa e das tarefas são executadas. Em seguida, entra-se na função `osal_run_system()` que executa o loop. Ele inicia pela tarefa de maior prioridade e verifica se há algum evento. Caso não tenha, passa para a tarefa seguinte de menor prioridade, e assim sucessivamente. Caso encontre algum evento, este é tratado por uma chamada de *call-back* e, então, o loop é reiniciado. No início desse loop, também se habilitam eventos de tarefas programados para ocorrerem periodicamente. Após passar por todas as tarefas, habilita-se um timer de *Wake-up* para “acordar” o sistema após certo período de tempo e, então, entra-se num modo *sleep* de baixo consumo em que não se faz nada. Para sair desse modo e reiniciar o loop, espera-se por um *Wake-up* ou por uma interrupção do sistema.

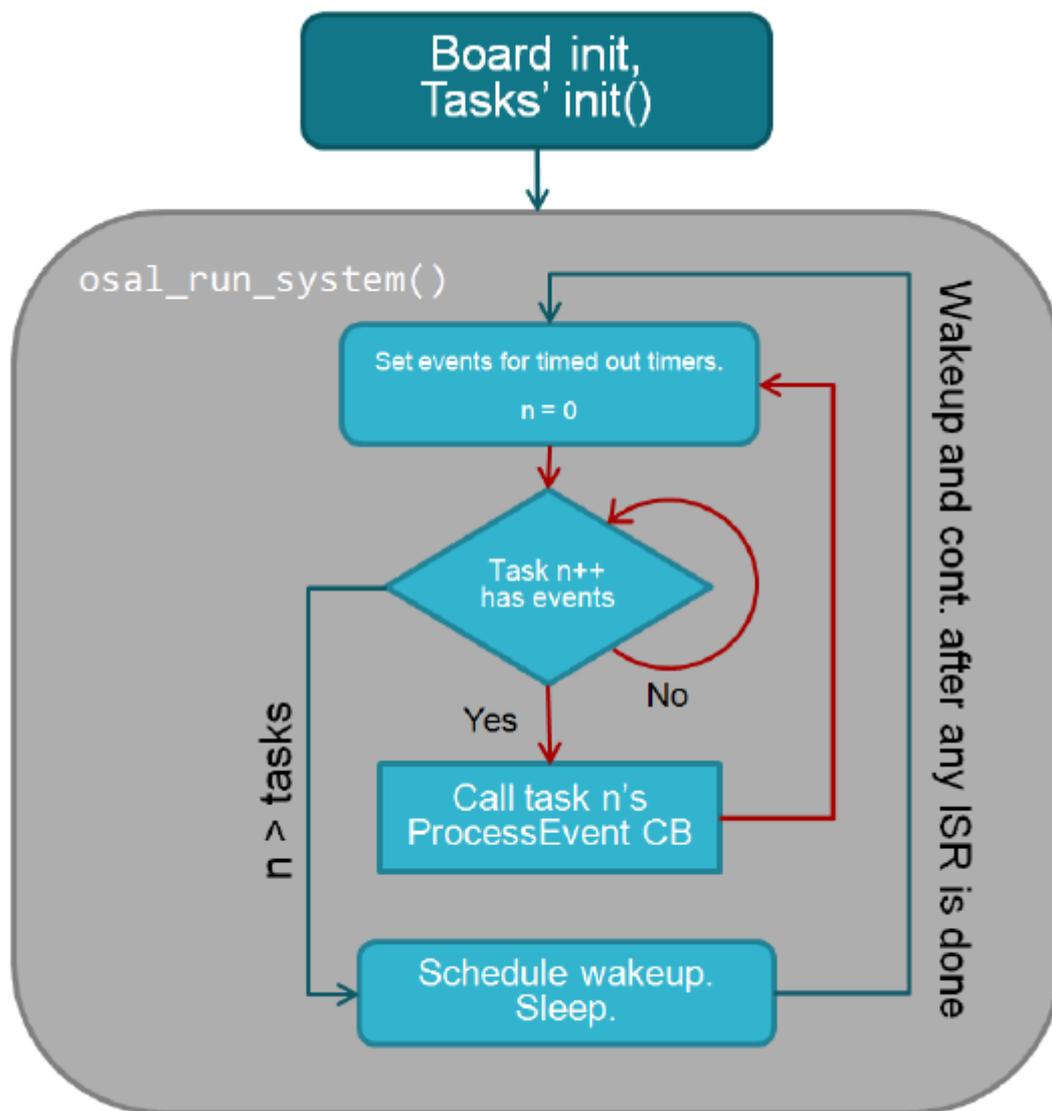


Figura 20: Fluxograma do loop do OSAL  
Fonte: (TEXAS INSTRUMENTS, 2013c)

Esses eventos de tarefas podem ocorrer por mudanças em atributos ou em estados do GATT, devido à troca de dados via BLE, ou por interrupções ou *timers* de *hardware*. Na tarefa da aplicação, cada sensor apresenta um evento correspondente que é habilitado periodicamente por *timers* configuráveis, assim possibilitando a leitura de dados de cada sensor. Existe também um evento de mensagem que pode ocorrer de forma assíncrona, por exemplo, quando um botão é pressionado.

## 7.4 APLICATIVO DA TEXAS INSTRUMENTS

Para facilitar o desenvolvimento de aplicações móveis com o SensorTag, a Texas Instruments disponibiliza gratuitamente um aplicativo para verificação das medidas dos sensores.

A Figura 21 mostra duas imagens de tela desse aplicativo.

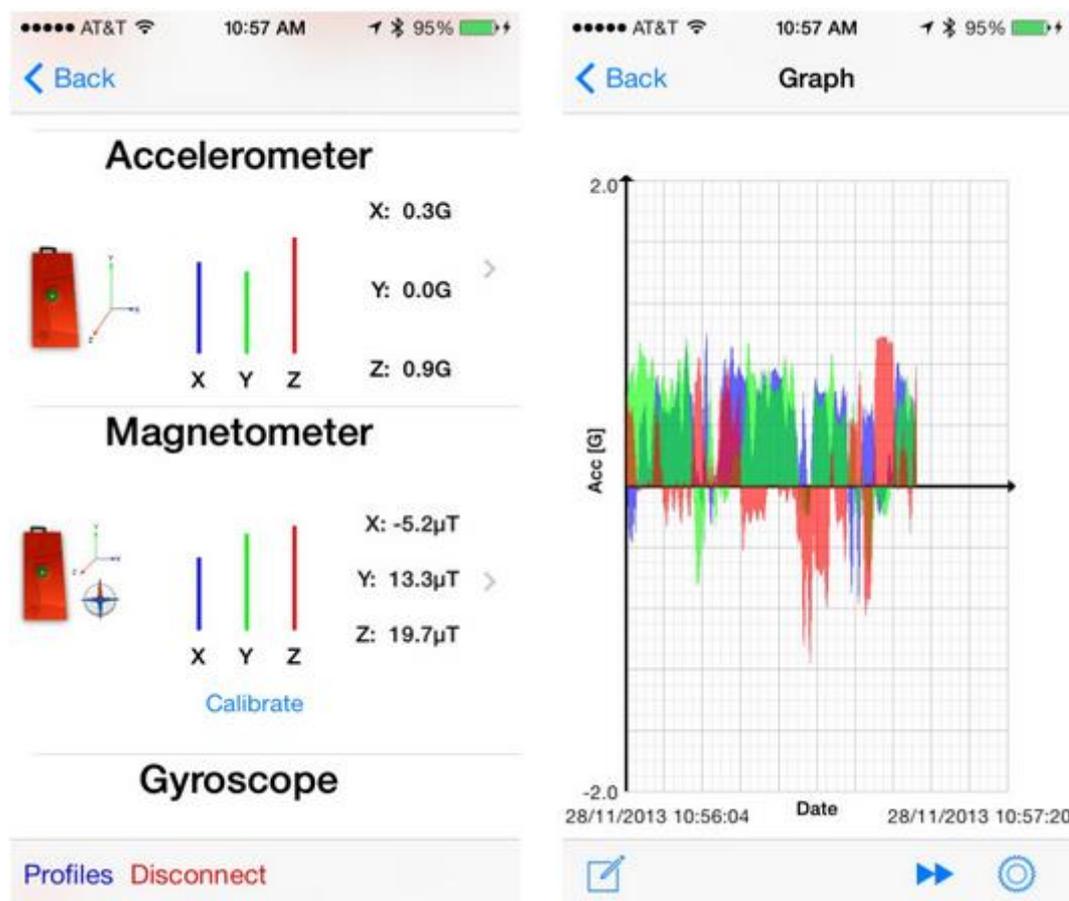


Figura 21: Duas imagens do aplicativo da Texas Instruments.  
Fonte: (DITTRICH, 2013).

## 7.5 FUNÇÃO OAD (OVER-THE-AIR DOWNLOAD)

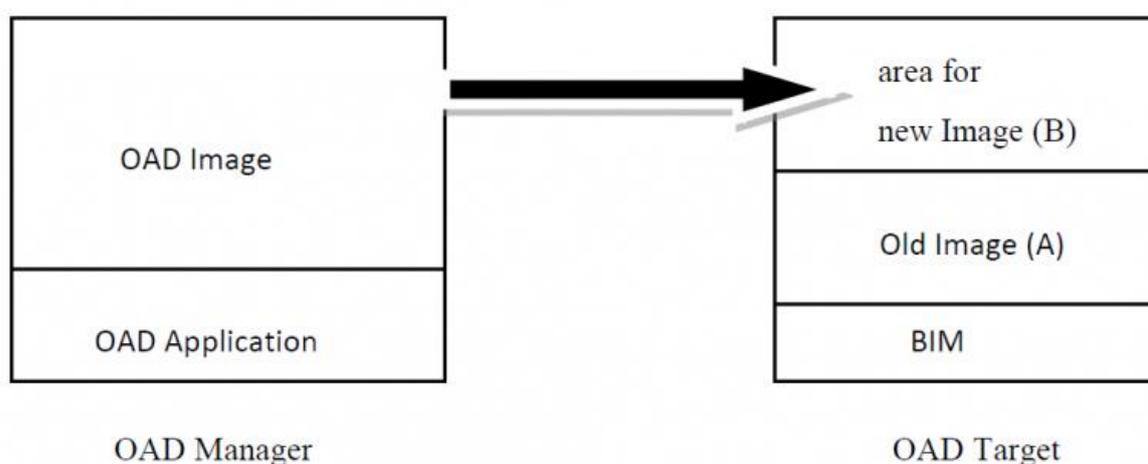
Uma função muito interessante do SensorTag é que é possível atualizar seu firmware através de um dispositivo central (*smartphone*), atuando como *OAD Manager* (Gerenciador do OAD), comunicando-se via rádio BLE.

Na memória *flash* de 256 kB do SensorTag reservam-se duas metades, onde em uma metade fica, por exemplo, a Imagem A que roda a aplicação atual, e

na outra metade fica a Imagem B reservada. O OAD usa o Gerenciador de Imagens de Boot (*Boot Image Manager* – BIM) que faz o processo de *download* de uma nova imagem gravando-a na metade oposta da imagem da aplicação.

Supondo que a Imagem A seja a aplicação atual e seja usada para executar o OAD, então a imagem nova gravada se torna a Imagem B. Após um *reset* no término do OAD, a Imagem B com o novo *firmware* se torna a imagem que executa a aplicação.

A Figura 22 mostra um diagrama desse processo.



**Figura 22: Diagrama do Over-The-Air Download (OAD).**  
 Fonte: (TEXAS INSTRUMENTS WIKI, 2013).

Usou-se no início do desenvolvimento esse processo de atualização de firmware por meio do aplicativo fornecido pela Texas Instruments, mas como houve a necessidade de uma alteração mais profunda dele, utilizou-se o Kit CC Debugger da Texas Instruments para dar mais agilidade no processo de desenvolvimento, bem como para fazer a depuração do código.

## 7.6 AMOSTRAGEM DO ACELERÔMETRO

Para se medir golpes de lutadores, usou-se o acelerômetro do SensorTag. Por meio de uma cinta elástica com Velcro®, prendeu-se o Sensortag firmemente no saco de pancadas de forma que os golpes fossem aplicados na superfície oposta a que o sensor se encontra.

Para que o acelerômetro meça um golpe de forma fidedigna, é necessário que ele tenha uma boa taxa de amostragem. Como o acelerômetro é digital, ele já possui um conversor analógico-digital (ADC) interno que faz a amostragem do sinal de aceleração.

#### 7.6.1 Limitação do SensorTag

De acordo com o servidor GATT do SensorTag, o menor período entre leituras do sensor acelerômetro, via BLE, é de 100ms. Ou seja, operar com o SensorTag em sua configuração normal utilizando o servidor GATT é inviável para a aplicação em questão, pois de acordo com Coelho (2008) e outras referências, um golpe ou um impacto tem duração entre 10 ms à 50 ms. Com isso, chegou-se a conclusão que o SensorTag, pelo jeito que ele foi concebido, é destinado para aplicações estáticas em que se mede estados dos sensores em períodos relativamente grandes de tempo.

Mesmo assim, procurou-se um jeito de conseguir utilizá-lo para a aplicação do projeto.

#### 7.6.2 Alterações de firmware

Para contornar a limitação do SensorTag, estudou-se a fundo o código do firmware para ler dados do acelerômetro de forma mais rápida. Uma solução encontrada foi utilizar o mesmo princípio dos botões. Ao se apertar um botão, envia-se um evento de mensagem que é tratado pela aplicação de forma assíncrona enviando uma notificação para o dispositivo cliente através do BLE. Para isso, algumas configurações do GATT foram desabilitadas, como o período de leituras do acelerômetro.

O acelerômetro, pelo seu manual (KIONIX, 2012), pode amostrar no máximo a uma frequência de 1600 Hz, correspondendo a um período entre amostras de 0,625 ms. Esse valor de amostragem é viável para a aplicação. Então, configurou-se

na inicialização do sistema para que o acelerômetro ficasse sempre ligado amostrando nessa frequência de 1600 Hz.

Ainda, na configuração inicial, habilitou-se o uso de interrupções do pino 7 do acelerômetro. Configurou-se da seguinte forma: toda vez que houver um dado novo de amostragem, a cada 0,625 ms, o pino de interrupção do acelerômetro vai para nível baixo. Assim, o CC2541 por meio desse pino, conectado em seu pino P0.2, consegue tratar cada novo dado de amostragem. No entanto, a cada nova interrupção, precisa-se tratá-la, lendo dados do acelerômetro, e liberá-la, através de um comando de leitura pela I2C no registrador INT\_REL, para gerar uma nova interrupção. Depois de feito esse comando de leitura, o pino de interrupção do acelerômetro vai para nível alto, possibilitando-o a gerar uma nova interrupção.

Nota-se, todavia, que o tempo entre acontecer uma interrupção, abaixando o nível do pino 7, e liberá-la, subindo o nível de seu pino, deve ser menor que o período de 0,625 ms de amostragem do acelerômetro. Fato esse que não foi possível. Mesmo assim, buscou-se obter um tempo de amostragem que fosse viável para a aplicação.

Quando o acelerômetro gera uma interrupção, o CC2541 trata-a por meio de uma rotina de tratamento de interrupção (ISR – *Interrupt Service Routine*). Inicialmente, tentou-se ler os dados de aceleração dos eixos x, y e z do acelerômetro e liberar o acelerômetro para gerar uma nova interrupção dentro da ISR, de forma que a interrupção do acelerômetro fosse rapidamente liberada. Mas como essas leituras são feitas pela I2C, que utiliza interrupções, gera-se um loop infinito dentro da ISR, travando o sistema.

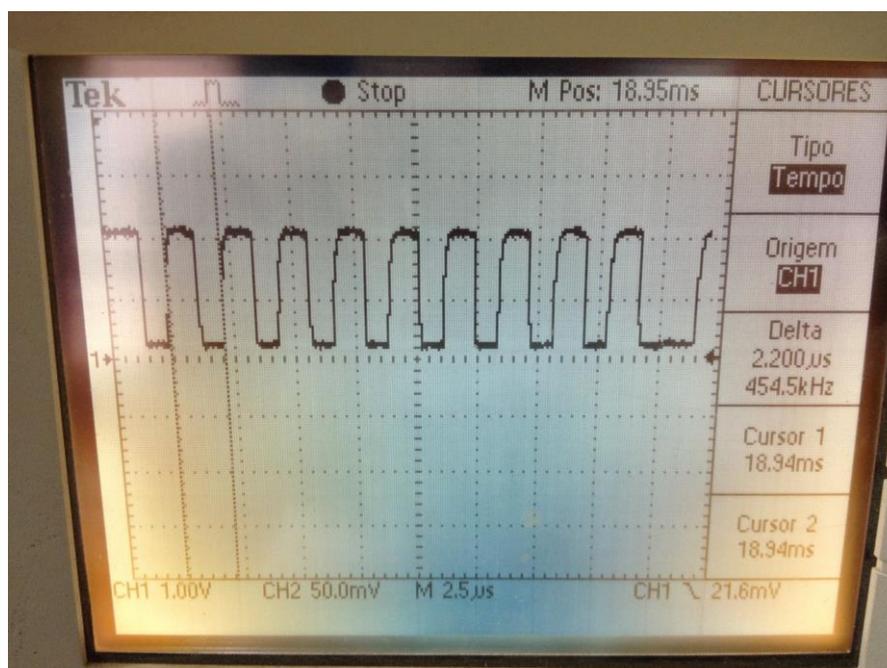
Então, utilizou-se outra solução. Na ISR, limpa-se o *flag* de interrupção do CC2541 e envia-se um evento de mensagem para aplicação. Na ISR, colocou-se também um indicador que é enviado junto com a mensagem. Esse indicador é um contador de 8 bits que é incrementado ciclicamente toda vez que ocorrer uma interrupção. Com isso, tem-se o controle do número de amostras do acelerômetro. Quando o loop do OSAL passar pela tarefa de aplicação encontrará o evento de mensagem que foi enviado pela ISR. No tratamento desse evento de mensagem, fazem-se as leituras de aceleração dos eixos x, y e z do acelerômetro e a leitura do INT\_REL para liberar o acelerômetro para gerar uma nova interrupção, e logo em seguida, antes de ocorrer uma nova interrupção do acelerômetro, coloca-se um pacote de dados no buffer de envio de dados para ser transmitido via notificação

BLE no *handle* do acelerômetro (0x30). Esse pacote de dados contém quatro bytes, referentes à aceleração do eixo x, y, z e mais o contador de interrupção da amostra lida.

#### 7.6.2.1 Características do clock da I2C do CC2541

Pelo datasheet do acelerômetro é possível operá-lo com uma frequência de clock de I2C de até 3,4 MHz. Como o CC2541 só consegue fornecer uma frequência de I2C de no máximo 533 kHz, o tempo de uma leitura de dado de um registrador do acelerômetro pela I2C fica maior.

Apesar de configurá-lo para operar com uma frequência de I2C de 533 kHz, na prática observa-se que ele opera em 454,5 kHz. Isso pode ser visto pela Figura 23, onde se mediu o sinal do pino SCL do acelerômetro.



**Figura 23: Sinal de *clock* da I2C do acelerômetro.**  
**Fonte: Autoria própria.**

Analisando esse sinal de *clock* consegue-se saber o tempo que leva para ler os dados x, y e z do acelerômetro e liberar a interrupção.

A Figura 24 mostra que esse tempo dura em torno de 600  $\mu$ s.

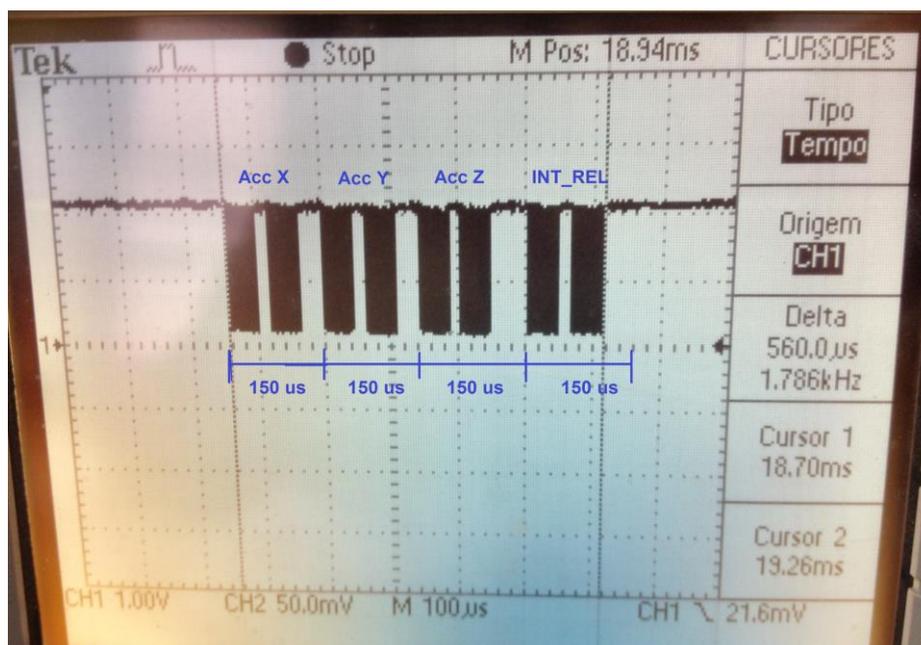


Figura 24: Tempo para ler dados do acelerômetro e liberar sua interrupção.  
Fonte: Autoria própria.

Para ler um dado de um registrador pela I2C, leva-se em torno de 150 μs, considerando os tempos de espera da operação.

#### 7.6.2.2 Tempo para ocorrer nova interrupção (t) e tempo de atendimento do evento de mensagem

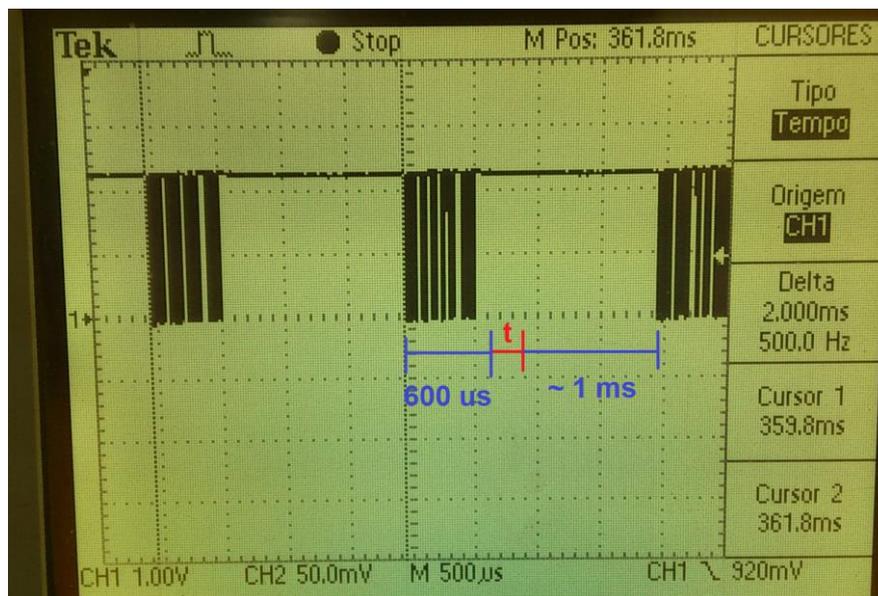
Além desses 600 μs para ler os dados do acelerômetro e liberar sua interrupção, há o tempo para que, após liberar a interrupção, ocorra uma nova interrupção que está programada para ocorrer a cada 0,625 ms, desde que liberada a interrupção. Esse tempo varia muito, pois depende de quando que é liberada a interrupção. Na Figura Z esse tempo é indicado como “t”.

Após ocorrer a nova interrupção, existe o tempo que a aplicação leva para atender o evento de mensagem gerado na ISR. É o tempo que dura o para o OSAL rodar por todas as tarefas e encontrar na tarefa da aplicação o evento de mensagem para ser tratado. Pelas observações que foram feitas com o osciloscópio, esse tempo é em torno de 1 ms.

Analisando todo esse ciclo para se ler dados do acelerômetro, conclui-se que o **tempo de amostragem é de 2 ms**, em média. Possivelmente ocorram

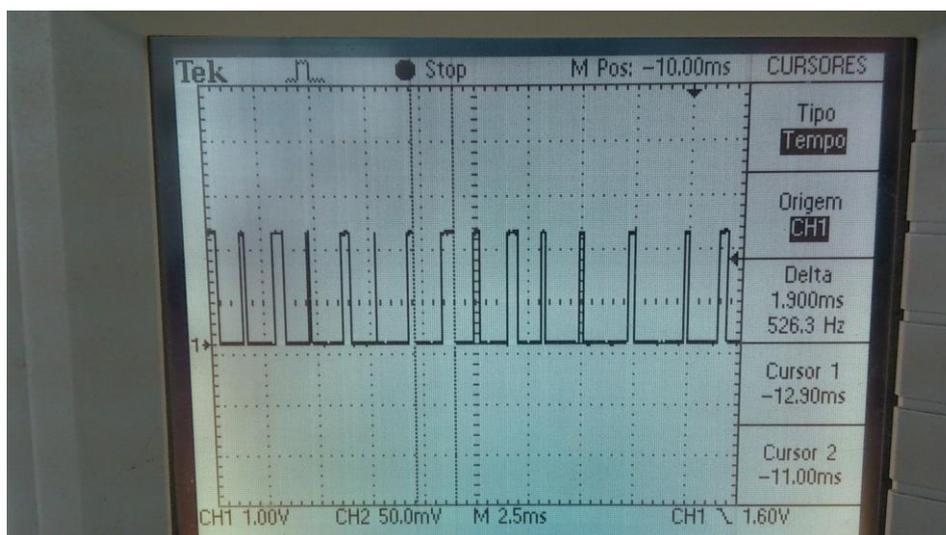
amostragens em intervalos um pouco menores, da mesma forma que pode ser que ocorram em intervalos um pouco maiores, mas em média ocorrem em 2 ms. Essa taxa de amostragem, apesar de não ser muito boa, foi utilizada para se desenvolver o projeto.

Na Figura 25 pode-se observar todo esse comportamento.



**Figura 25: Tempos na amostragem do acelerômetro.**  
Fonte: Autoria própria.

Na Figura 26 observa-se o sinal de interrupção do pino INT.



**Figura 26: Sinal de interrupção do acelerômetro.**  
Fonte: Autoria própria.

### 7.6.2.3 Características do BLE

Após estabelecida uma taxa de amostragem de 2 ms, esperava-se que houvessem dados do acelerômetro sendo enviados por meio de notificações BLE para o iPhone a cada 2 ms. No entanto, temos que considerar algumas limitações do BLE.

Numa conexão BLE, ocorrem trocas de dados a cada evento de conexão. Eventos de conexão ocorrem a cada período de tempo predefinido. Para uma comunicação entre o iPhone 4s e o SensorTag, o menor intervalo de conexão é de 30 ms. Ou seja, a cada 30 ms ocorre um evento de conexão em que se transmitem pacotes de dados. Mesmo que não haja pacotes de dados para se transmitir, ocorre pelo menos uma troca de dado entre o master e o slave, através de um par de pacotes vazios.

Ainda, existe um limite máximo de 6 pacotes de dados que se consegue transmitir a cada evento de conexão. Esse máximo valor de 6 pacotes é obtido, por meio da habilitação de uma função no software embarcado denominado de OVERLAPPED PROCESSING. De acordo com a TI BLE STACK, o maior tamanho do pacote de dados é de 20 bytes, considerando dados úteis de aplicação. Com isso, conclui-se que a cada 30 ms podem ser enviados no máximo 120 bytes de dados.

Como a amostragem adquire 4 bytes de dados a cada 2ms, em 30 ms haveriam um total de 60 bytes que seriam o suficiente para serem transmitidos. Porém, encontrou-se um comportamento estranho na amostragem do acelerômetro. Ao se esperar 5 amostras para serem enviadas em um pacote de 20 bytes, a amostragem do acelerômetro não ocorria conforme mostrado anteriormente. Amostrava-se algumas vezes a cada 2 ms e depois ficava um período grande de tempo até ocorrer o evento de conexão para voltar a amostrar. Notava-se que o tempo para se atender o evento de mensagem era muito grande. Com esse comportamento, essa solução foi descartada.

Para que se mantenha uma amostragem boa de 2 ms, deve-se colocar um pacote de dados, no buffer do rádio BLE para envio de uma notificação, a cada vez que ocorre uma amostragem. Esse comportamento estranho caracteriza uma dependência entre o fato de transmitir dados e o processo de amostragem do acelerômetro.

Mesmo que se enviassem pacotes de dados contendo apenas uma amostra a cada 2ms, apenas os 6 pacotes seriam enviados a cada 30ms. Além disso, existe o tempo em que o rádio leva para transmitir esses pacotes, a cada evento de conexão, que é em torno de 4 ms, perdendo tempo da amostragem. Nesse esquema, observou-se, como o auxílio do contador de interrupções, uma perda de 67% das amostras.

Dessa forma, optou-se por uma nova solução que foi utilizada no projeto.

#### 7.6.2.4 Uso de um limiar de detecção de golpes

Ao invés de transmitir todos os dados de amostragem do acelerômetro, o que implicaria num volume grande de dados, utilizou-se um limiar de detecção de golpes. Enviando todos os dados amostrados, a maioria deles seriam informações sem relevância, pois o que interessa são dados de golpes que ocorrem em períodos curtos bem específicos e espaçados no tempo.

O limiar de detecção de golpes funciona como se fosse um “*trigger*”. Utiliza-se o eixo z do acelerômetro em que se tem a direção do golpe para obter uma detecção de golpe. Quando a aceleração do eixo z for maior que 0,94 g ou menor que -0,94 g, ou seja, quando a aceleração do eixo z for maior que 0,94 g para ambos os sentidos, armazena-se todas as amostras seguintes em um vetor de 159 amostras de 4 bytes, contendo x, y, z e contador, na memória RAM de 8 kB. Esse tamanho do vetor de amostras foi o máximo de memória RAM que se tinha disponível, 636 bytes.

Após completar o preenchimento do vetor, o SensorTag envia ao aplicativo no Iphone 4s todas as amostras armazenadas na RAM. Isso é feito da seguinte forma, a cada evento de conexão (30 ms), o rádio BLE envia um pacote de notificação, contendo uma amostra armazenada na memória RAM, e assim sucessivamente até enviar todas as amostras. Envia-se apenas um pacote por evento de conexão para garantir que nenhuma amostra seja perdida. Após terminar de transmitir todos os dados, o sensor volta ao estado de amostragem a cada 2 ms, com seu detector de golpes habilitado. Quando se transmite esses dados armazenados na RAM, a amostragem do acelerômetro fica desabilitada, dessa

forma, deve-se esperar em torno de 5 segundos para que o sensor fique ativado para medir o próximo golpe. Então, o lutador deve ser orientado a aplicar golpes com intervalos de 5 segundos.

No entanto, para que a amostragem seja feita a cada 2 ms, é necessário que a cada amostra que surja, pela interrupção, um pacote de dados seja enviado via notificação BLE. O que se faz é um envio de notificação de um pacote de dado contendo um byte que é o valor do contador da amostra. Observa-se que há perda de pacotes, mas esses dados são somente para que a amostragem permaneça estável, tanto que se usa como *handle* de notificação um valor qualquer que não é visto pelo aplicativo em Iphone.

## 7.7 CONSIDERAÇÕES SOBRE A NORMA SAE J211-1 (1995)

A norma *SAE J211-1 (1995): Instrumentation for Impact Test* é uma regulamentação norte-americana para que instrumentações com impactos sejam feitas de forma fidedigna.

A taxa de amostragem do sensor não está em conformidade com a SAE J211-1 (1995) devido a limitações de hardware. De acordo com a norma, recomenda-se uma taxa de amostragem de no mínimo 10 kHz. A máxima taxa de amostragem obtida com o hardware utilizado foi de aproximadamente 500 Hz.

Em termos de resolução a SAE J211-1 (1995) recomenda utilizar dados digitais de pelo menos 10 bits (incluindo o sinal). O acelerômetro utilizado apresenta duas formas de configuração de acordo com seu *datasheet*:

- dados com sinal de 8 bits utilizando um byte;
- dados com sinal de 12 bits usando dois bytes.

Optou-se por usar apenas 8 bits, pois se utilizasse a configuração de 12 bits levar-se-ia quase o dobro do tempo para se ler os dados pela I2C. Em vez de um byte, seriam 2 bytes por leitura. Isso resultaria em um aumento de praticamente duas vezes o período de amostragem, reduzindo a quase a metade a frequência de amostragem de 500 Hz.

Utilizando dessa forma a resolução não está em conformidade com a norma, em que o bit menos significativo deve corresponder a aproximadamente 0,2% da máxima amplitude de fundo de escala. Como o acelerômetro utilizado mede de -8 g

a +8 g, tem-se uma resolução de 0,0625 g, correspondendo a 0,78% da máxima amplitude de fundo de escala.

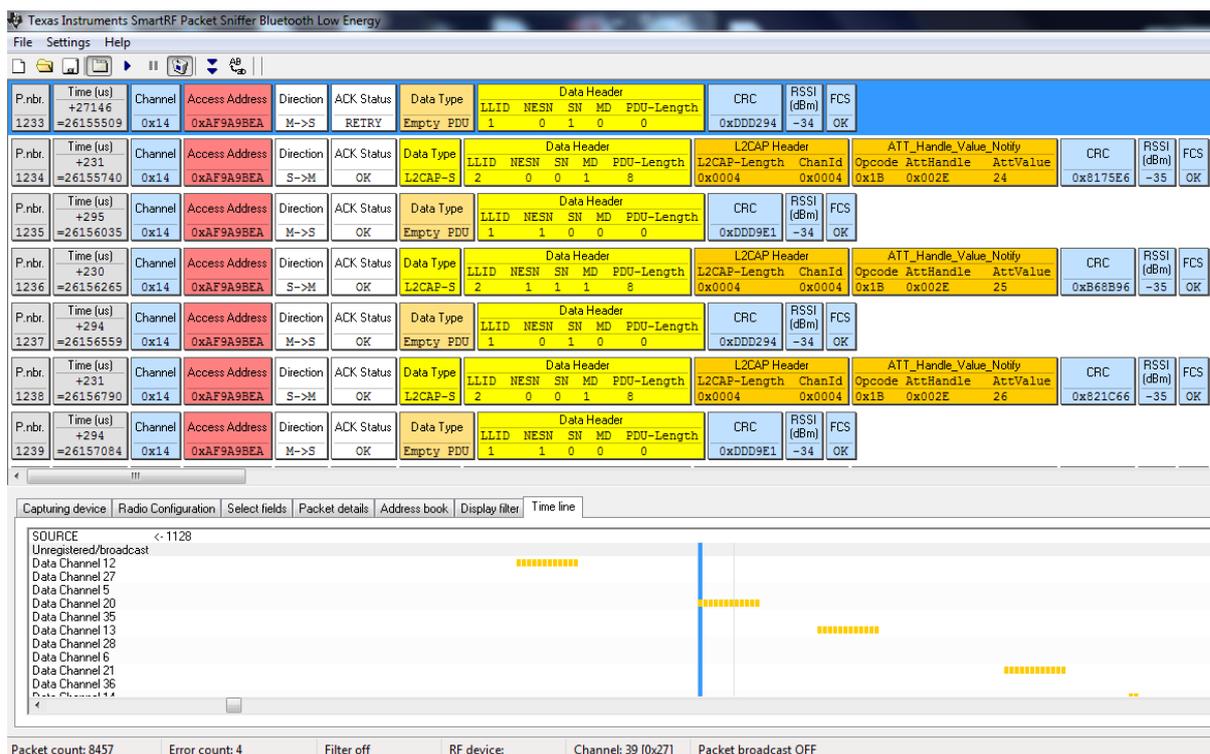
Os dados foram processados respeitando a norma SAE J211-1 (1995), adequando a escala e o nível zero matematicamente para se obter dados apropriados para a análise. Como o sensor fica fixado sempre numa posição predefinida, onde praticamente apenas o eixo y fica sob a influência da aceleração da gravidade, subtraiu-se 1 g do valor do eixo y para o cálculo do módulo de aceleração. Utilizou-se essa abordagem pois o módulo da aceleração de interesse corresponde ao primeiro pico de aceleração em que o eixo do saco pouco se altera.

Com relação ao tempo de impacto, seguiu-se a norma. De acordo com ela, o tempo de contato inicial (real ou simulado) deve ser conhecido em mais ou menos 1 ms e pode ser obtido observando o instante em que a aceleração excede um valor predeterminado, por exemplo 0,5g. Neste projeto, usaram-se dois limiares de +0,94 g e -0,94 g no eixo z (eixo do golpe) para definir um impacto e calculou-se o tempo de impacto através da curva do primeiro pulso de aceleração do eixo z.

## 7.8 TEMPO DE DURACAO DA BATERIA (CR2032) DO SENSORTAG

Neste tópico, estima-se por quanto tempo o SensorTag opera com uma bateria CR2032. O acelerômetro utilizado fica ligado direto em sua máxima potência e taxa de amostragem, consumindo em torno de 120  $\mu$ A. Para estimar o tempo de duração do dispositivo alimentado pela bateria CR2032 que possui uma capacidade típica de 230 mAh (ROPLA ELEKTRONIK, 2012), faz-se a análise a seguir.

O sensor fica periodicamente a cada 30ms, aproximadamente, trocando informações com o Iphone. Ou seja, a cada 30 ms ocorre um evento de conexão, em que pelo menos um par de pacotes de transmissão e recepção é trocado entre o Iphone e o sensor. Na aplicação desenvolvida, a cada evento de conexão, ocorrem 12 trocas de pacotes (6 pares). Obtém-se esse período entre eventos de conexão observando a Figura 27 a seguir do programa SmartRF Packet Sniffer Bluetooth Low Energy da Texas Instruments. Esse programa funciona como um *Sniffer*, “farejador”, em que se capturam os pacotes de dados de uma conexão BLE por meio de um *dongle* USB (CC2540 USB Dongle) da Texas Instruments.



**Figura 27: Captura de tela do Software SmartRF Packet Sniffer BLE.  
Fonte: Autoria própria.**

Pela Figura 27, observam-se os tempos entre cada pacote enviado pelo quadro de tempo (segundo quadro em cinza) dos pacotes de dados. Nota-se que são em pares, o Master (Iphone) envia um pacote vazio e o Slave (SensorTag) envia um pacote com dados. Não é possível ver todos os pacotes de um evento de conexão na Figura 27, mas como os tempos se repetem para cada transmissão, pode-se calcular o tempo entre eventos de conexão. Exceto o primeiro pacote de transmissão, observa-se que um par de trocas de pacotes corresponde a aproximadamente  $525 \mu\text{s}$  ( $295 \mu\text{s} + 230 \mu\text{s}$ ). Somando-se os 12 pacotes ( $27146 + 231 + 5 \cdot 525$ ), chega-se a aproximadamente 30 ms.

A Figura 28 a seguir, mostra o sinal do consumo de corrente durante um evento de conexão para um par de pacotes de transmissão. Nela, observa-se o consumo de corrente segmentado em estados.

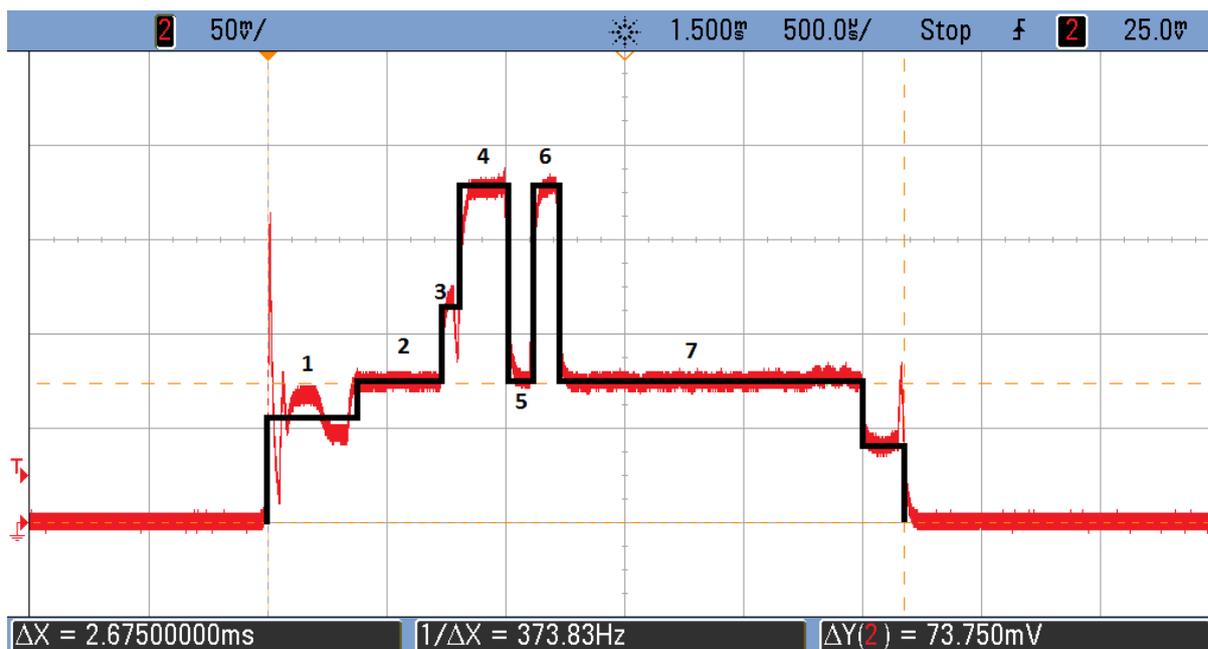


Figura 28: Sinal de corrente dividido em estados.  
Fonte: (KAMATH; LINDH 2012).

De acordo com o Kamath e Lindh (2012), pode-se calcular a média de corrente durante o evento de conexão. Com a seguinte fórmula:

$$\text{Média de corrente durante um evento de conexão} = \frac{[(\text{tempo do Estado 1}) \cdot (\text{corrente do Estado 1}) + (\text{tempo do Estado 2}) \cdot (\text{corrente do Estado 2}) + \dots]}{(\text{Tempo total acordado})}$$

Preenche-se o Quadro 1, a seguir, com os valores de consumo de corrente para cada estado, considerando 6 pares de pacotes de transmissão.

	Tempo ( $\mu\text{s}$ )	Corrente (mA)
Estado 1 ( <i>wake-up</i> )	400	6,0
Estado 2 ( <i>pre-processing</i> )	315	7,4
Estado 3 ( <i>pre-Rx</i> )	$80 * 6$	11,0
Estado 4 ( <i>Rx</i> )	$275 * 6$	17,5
Estado 5 ( <i>Rx-to-Tx</i> )	$105 * 6$	7,4
Estado 6 ( <i>Tx</i> )	$115 * 6$	17,5
Estado 7 ( <i>post-processing</i> )	1325	7,4
Estado 8 ( <i>pre-Sleep</i> )	160	4,1

Quadro 1: Valores de corrente para cada estado, considerando 6 pares de pacotes.  
Fonte: Adaptado de Kamath e Lindh (2012).

Utilizando a fórmula chega-se em 11,7 mA de consumo médio em um evento de conexão.

$$[(400 \mu\text{s}) \cdot (6 \text{ mA}) + (315 \mu\text{s}) \cdot (7.4 \text{ mA}) + (80 \cdot 6 \mu\text{s}) \cdot (11 \text{ mA}) + (275 \cdot 6 \mu\text{s}) \cdot (17.5 \text{ mA}) + (105 \cdot 6 \mu\text{s}) \cdot (7.4 \text{ mA}) + (115 \cdot 6 \mu\text{s}) \cdot (17.5 \text{ mA}) + (1325 \mu\text{s}) \cdot (7.4 \text{ mA}) + (160 \mu\text{s}) \cdot (4.1 \text{ mA})] / (5650 \mu\text{s}) = 11,6963 \text{ mA}$$

Considerando que o acelerômetro está sempre ligado consumindo 120  $\mu\text{A}$ , soma-se esse valor à média de corrente, resultando em um consumo de 11,8163 mA.

Sabe-se o quanto se consome de corrente durante o envio dos pacotes de dados. O próximo passo é calcular a média de consumo no intervalo de conexão inteiro de aproximadamente 30 ms. Para isso, usa-se a seguinte fórmula:

Média de corrente conectado =

$$[(\text{Intervalo de Conexão} - \text{Tempo total transmitindo dados}) \cdot (\text{Média de corrente sem estar transmitindo}) + (\text{Tempo total transmitindo dados}) \cdot (\text{Média de corrente durante um evento de conexão})] / (\text{Intervalo de Conexão})$$

Pela Figura 29, a seguir, o tempo em que se fica transmitindo pacotes é de 4,2 ms.

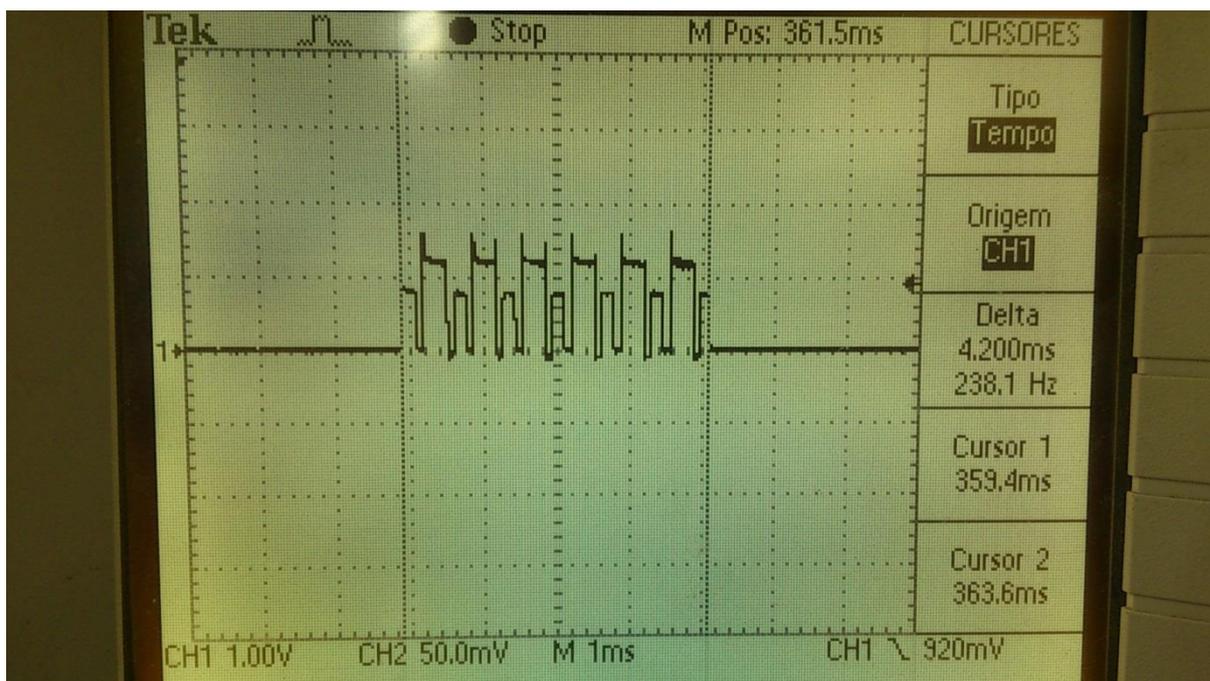


Figura 29: Tempo de transmissão de pacotes de dados num evento de conexão.  
Fonte: Autoria própria.

Utilizando a fórmula:

$$[(30 \text{ ms} - 4,2 \text{ ms}) \cdot (0,120 \text{ mA}) + (4,2 \text{ ms}) \cdot (11,8163 \text{ mA})] / (30 \text{ ms}) = 1,7575 \text{ mA}$$

Com esse valor de corrente pode-se calcular o tempo de duração da bateria.

Como ela apresenta 230 mAh, faz-se o seguinte cálculo:

$$230 \text{ mAh} / 1,7575 \text{ mA} = \mathbf{130,87 \text{ horas.}}$$

Essa é uma estimativa para se ter uma ideia de quanto a bateria dura. Não foram considerados os pequenos períodos em que o sensor fica em modo de baixo consumo consumindo  $1 \mu\text{A}$ , que seria o pequeno intervalo entre liberar a interrupção e gerar uma nova interrupção do acelerômetro. Também não se considerou os períodos em que se transmitem os dados que foram armazenados na RAM. Nesse período, menos de 6 pares de pacotes de transmissão são enviados, em torno de 1 par. Por esses fatores, o tempo de duração da bateria aumentaria. No entanto, outros fatores não observados podem acarretar num maior consumo de corrente. Para se obter um valor mais preciso desse tempo de duração da bateria, outros ensaios e análises devem ser feitos.

## 7.9 DESENVOLVIMENTO DO APLICATIVO

### 7.9.1 Requisitos de Desenvolvimento

Para desenvolver o aplicativo, nomeado de Fighter Pro, para iOS foi necessário:

- Computador com sistema operacional Mac OSX (OS X Mavericks)
- XCode
- Apple iPhone SDK 6.1 e 7.0
- CSR 4.0, para simulação do aplicativo com interface BLE
- iPhone 4S

### 7.9.2 Funcionamento

Os dados recebidos do SensorTag, através da comunicação BLE, são gravados no aplicativo em Iphone a cada transmissão de notificação no *handle* (UUID) do acelerômetro.

```

+(float) calcXValue:(NSData *)data {
    char scratchVal[data.length];
    [data getBytes:&scratchVal length:3];
    return ((scratchVal[0] * 1.0) / (256 / KXTJ9_RANGE));
}
+(float) calcYValue:(NSData *)data {
    char scratchVal[data.length];
    [data getBytes:&scratchVal length:3];
    return ((scratchVal[1] * 1.0) / (256 / KXTJ9_RANGE));
}
+(float) calcZValue:(NSData *)data {
    char scratchVal[data.length];
    [data getBytes:&scratchVal length:3];
    return ((scratchVal[2] * 1.0) / (256 / KXTJ9_RANGE));
}

```

O trecho de código acima é adaptado para Objective-C da página Wiki da Texas Instruments para o Sensor Tag, onde é disponibilizado em Java e em C. Nesta, eles recomendam usá-lo para fazer a leitura dos dados recebidos.

Neste projeto, cada dado recebido tem 4 bytes, conforme descrito anteriormente, em que o quarto byte foi inserido como controle para os testes realizados. Ele funciona como um contador, dessa forma podemos visualizar se entre os dados recebidos tivemos perdas. Contudo, apesar dele ter enorme relevância para o projetista, para o processamento e amostragem dos resultados para o usuário, ele não faz a menor diferença.

Ainda nesse trecho de código, temos uma variável que diz respeito ao *range*, ou os valores limites a serem considerados. No caso do acelerômetro que utilizamos temos três *ranges*: +/- 2G, +/-4G e +/-8G, usou-se 8 G devido à grande aceleração sofrida pelo saco no impacto de um golpe.

Após etapa de conversão dos bytes em *floats*, o aplicativo faz o cálculo do módulo da aceleração. Para este cálculo, levamos em consideração a posição inicial de repouso do sistema, que tem como aceleração  $x = 0.0$  g,  $y = 1.0$  g e  $z = 0.0$  g, assim o módulo é calculado:  $a = \sqrt{(x - 0,0)^2 + (y - 1,0)^2 + (z - 0,0)^2}$ .

Todos os dados recebidos vão sendo salvos, cada um em NSMutableArray dentro de uma *property* (tipo que pode ser chamado entre classes) com subclasse NSObject, assim como o valor cronometrado no momento recebido.

Esses dados são impressos em um arquivo .csv com o método “addAttachmentData” da framework MessageUI.

O aplicativo Fighter Pro tem um funcionamento simples e intuitivo. E para iniciar, basta um toque no ícone do aplicativo, destacado na Figura 30.

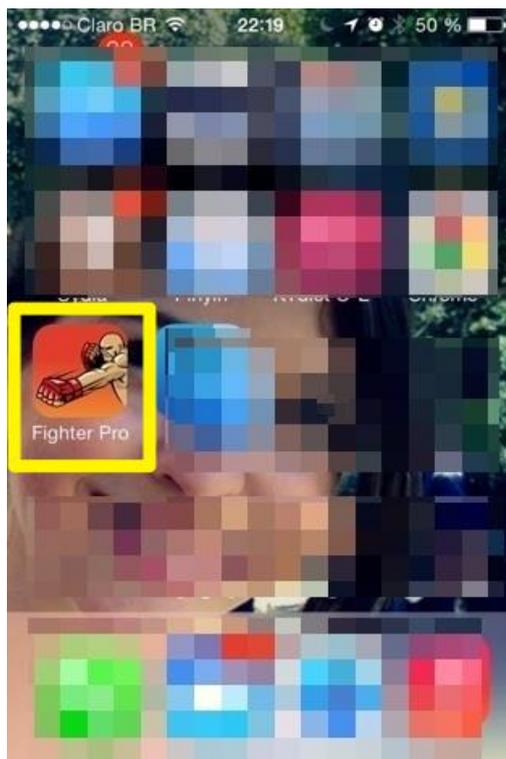


Figura 30: Ícone do aplicativo "Fighter Pro".  
Fonte: Autoria própria.

A Figura 31 mostra a tela inicial do aplicativo em que ela passa a fazer a varredura de dispositivos periféricos BLE, à procura de *advertisings*.

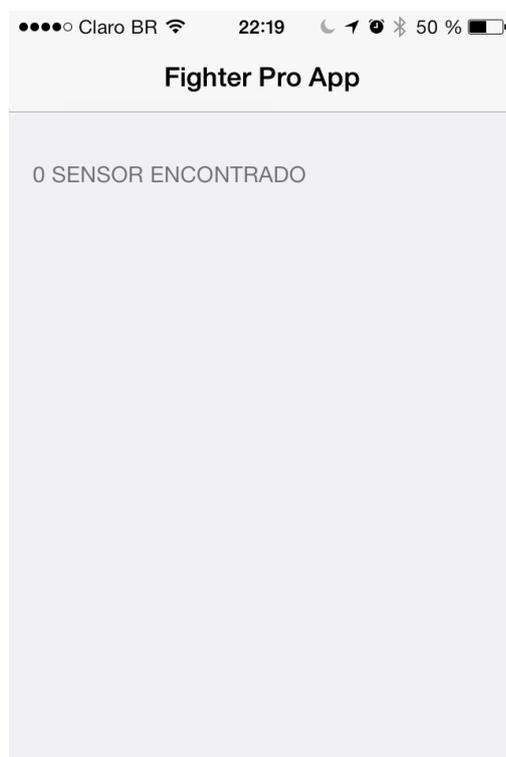


Figura 31: Tela de entrada do "Fighter Pro".  
Fonte: Autoria própria.

Na Figura 32 mostra a tela do aplicativo quando ele descobre um dispositivo BLE, e o mesmo fica disponível para conexão. É necessário que o usuário toque na descrição do dispositivo encontrado para conectar com o dispositivo.



**Figura 32: Tela com sensor encontrado e pronto para conexão.**  
**Fonte: Autoria própria.**

Na Figura 33 abaixo, realizando a conexão com o sensor, entra-se na tela de treino. Nessa, temos um cronômetro, quatro indicadores de intensidade com os valores capturados e um botão para o envio dos dados.

Para os indicadores de intensidade deve-se lembrar que nos eixos x, y e z há valores que variam de -8G a +8G, e para os valores absolutos da aceleração de 0 a 13,56G.



**Figura 33: Tela com conexão estabelecida e recebendo dados.  
Fonte: Autoria própria.**

Na Figura 34 abaixo, temos a tela após tocar no botão para exportação dos dados. Aqui, podemos exportar, para o endereço de preferência, um arquivo .csv com os dados obtidos durante o treino realizado.

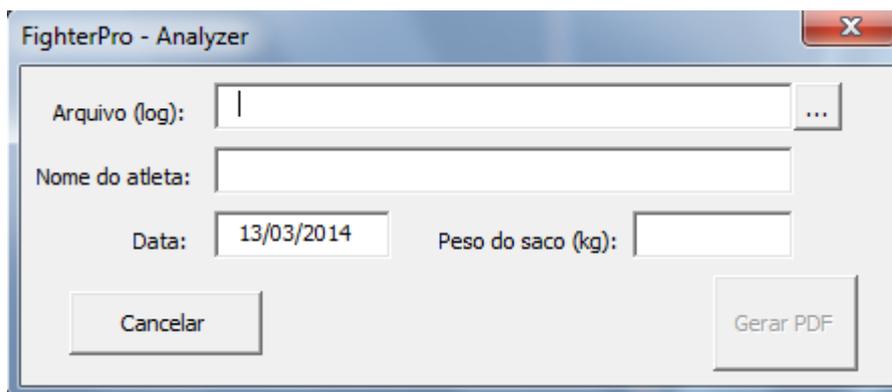


**Figura 34: Tela de exportação dos dados recebidos em um arquivo .csv**  
**Fonte: Autoria própria.**

## 7.10 TRATAMENTO DOS DADOS

Para o tratamento dos dados e visualização pelo usuário, os alunos elaboraram um programa em VBA e tem sua execução em Microsoft Office Excel. Optou-se por esta estratégia por ser uma ferramenta de melhor manipulação do usuário e por ser de fácil acesso.

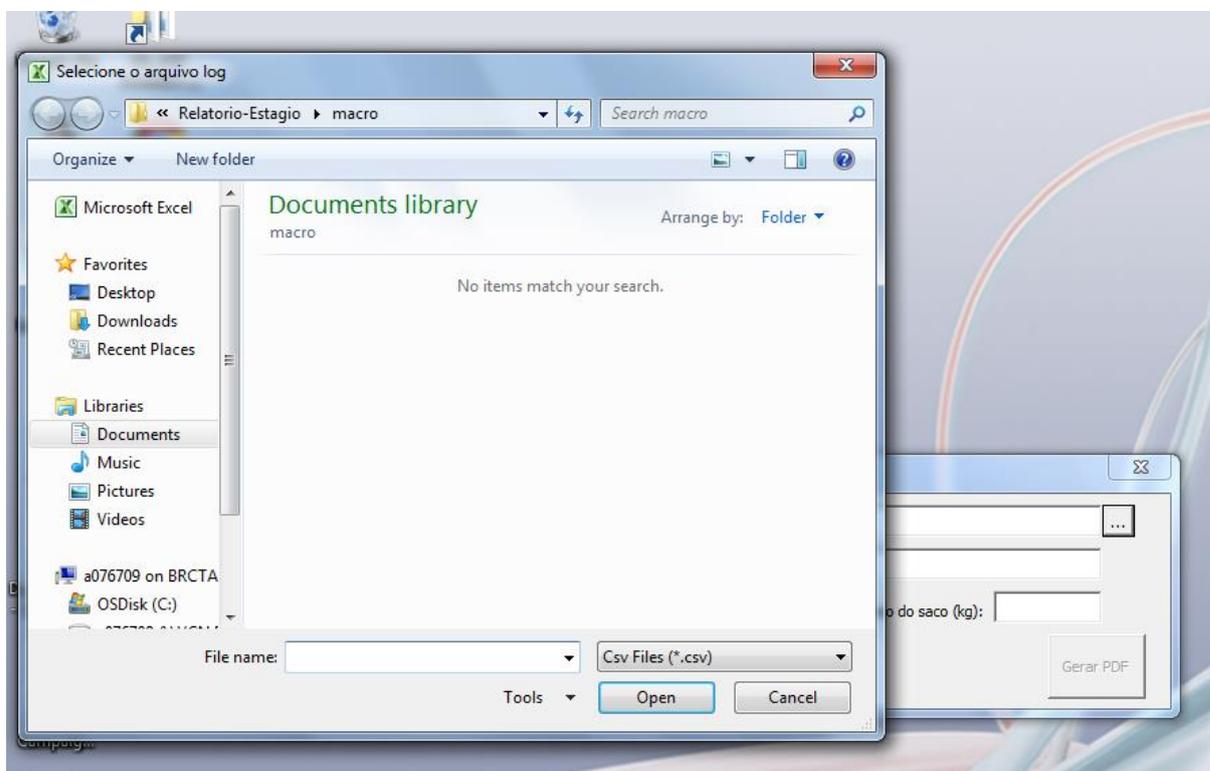
Para iniciar o funcionamento da Macro basta abrir o arquivo Macro FighterPro.xlsm, e assim abrirá a tela como mostrado na Figura 35:



**Figura 35: Tela inicial da Macro Fighter Pro.  
Fonte: Autoria própria.**

Nesta janela temos como campo inicial de preenchimento “Arquivo (log)”, ou seja, o arquivo .csv que foi enviado a partir do aplicativo. Ao usuário é recomendado não alterar o arquivo para assegurar o correto funcionamento do programa.

Para abrir um arquivo log basta clicar no botão “...” para procurá-lo no diretório em que foi salvo pelo usuário.



**Figura 36: Tela procura caminho do arquivo Log.  
Fonte: Autoria própria.**

Quando o arquivo é selecionado pelo usuário vem a tela da figura seguinte.

**Figura 37: Tela da macro com o caminho inserido.**  
**Fonte: Autoria própria.**

Agora o usuário pode preencher os campos “Nome do Atleta” e “Peso do saco (kg)”, com caracteres e números, respectivamente. Colocou-se a opção “Peso do saco” porque o valor da força é dependente da massa do saco.

**Figura 38: Tela da macro com os dados preenchidos**  
**Fonte: Autoria própria.**

Assim que identificados os dados, após pressionar a tecla “Enter”, esses são processados de forma a organizar os dados. Todo o processamento é feito ocultamente ao usuário, a este somente há uma interface de caixa de diálogo.

Primeiramente é calculada a Força, a partir da equação:  $\vec{F} = m \cdot \vec{a} \cdot FCM$ , onde FCM é o fator de consideração da massa, que será explorado nos resultados dos testes. Feito isso, é processado uma primeira média com todos os valores de força calculada, e inseridos esses dados em uma nova coluna. Em outra coluna, da mesma forma, são inseridos os valores que vem de um filtro para identificação de

picos. Nesse, é identificado, dentro de 2 segundos, qual é o maior valor de força, eliminando os outros. Dessa maneira, diminuimos consideravelmente o número de amostras deixando ao usuário uma visualização mais clara de seu treino, pois muitos dos dados são valores ao longo da curva de impacto, o que não é interessante para ele.

Ainda, desses valores filtrados é feita uma nova média, que terá, obviamente, seu valor acrescido. Fizemos isso para identificar os golpes efetivos. Se o valor de força for maior que essa nova média, esse é um golpe que ficou acima da média dos maiores valores, e isso informamos ao usuário no relatório que é produzido em PDF.

Dos valores filtrados plotamos o gráfico, e dos valores calculados, como média, desvio padrão, valor máximo e número de socos inserimos como informação ao relatório.

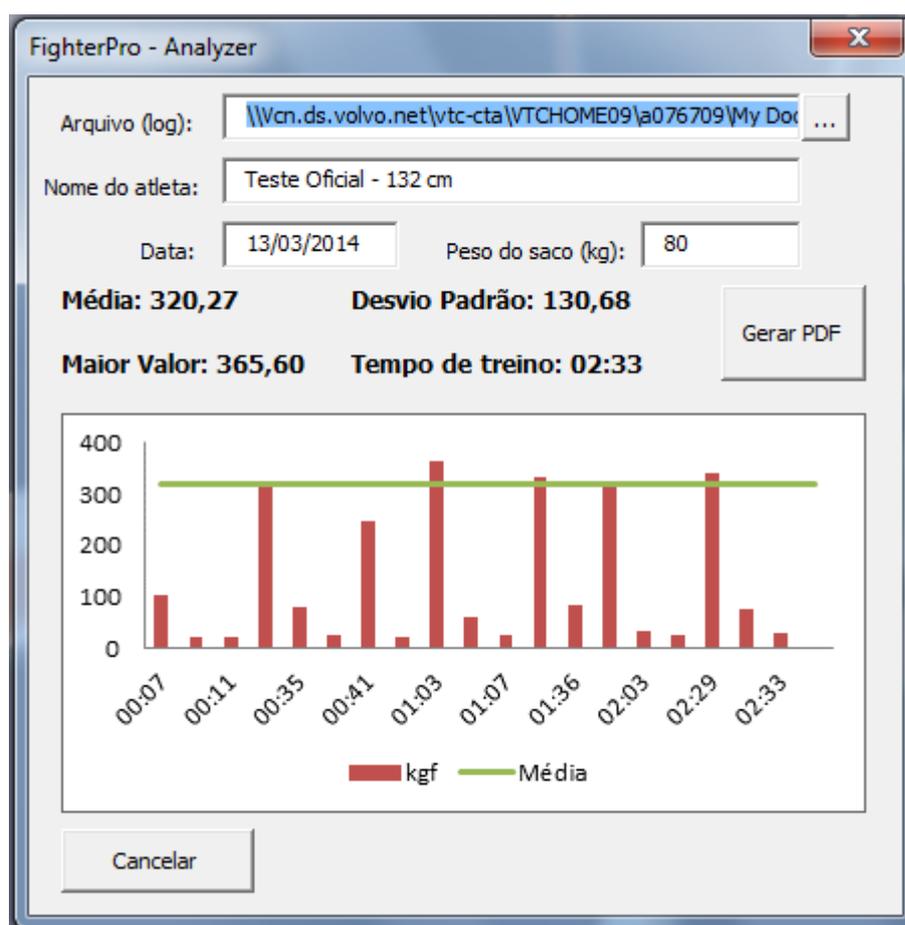
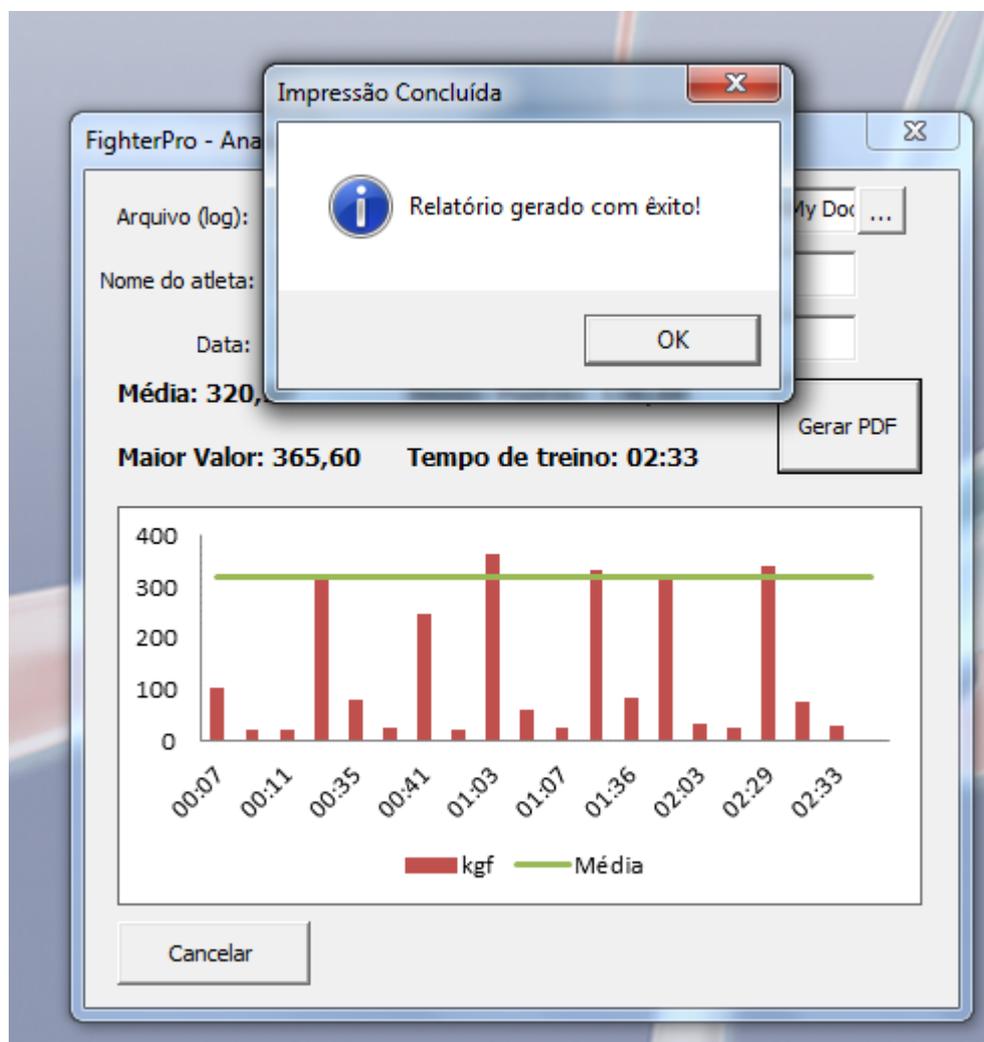


Figura 39: Tela com a análise do treino.  
Fonte: Autoria própria.

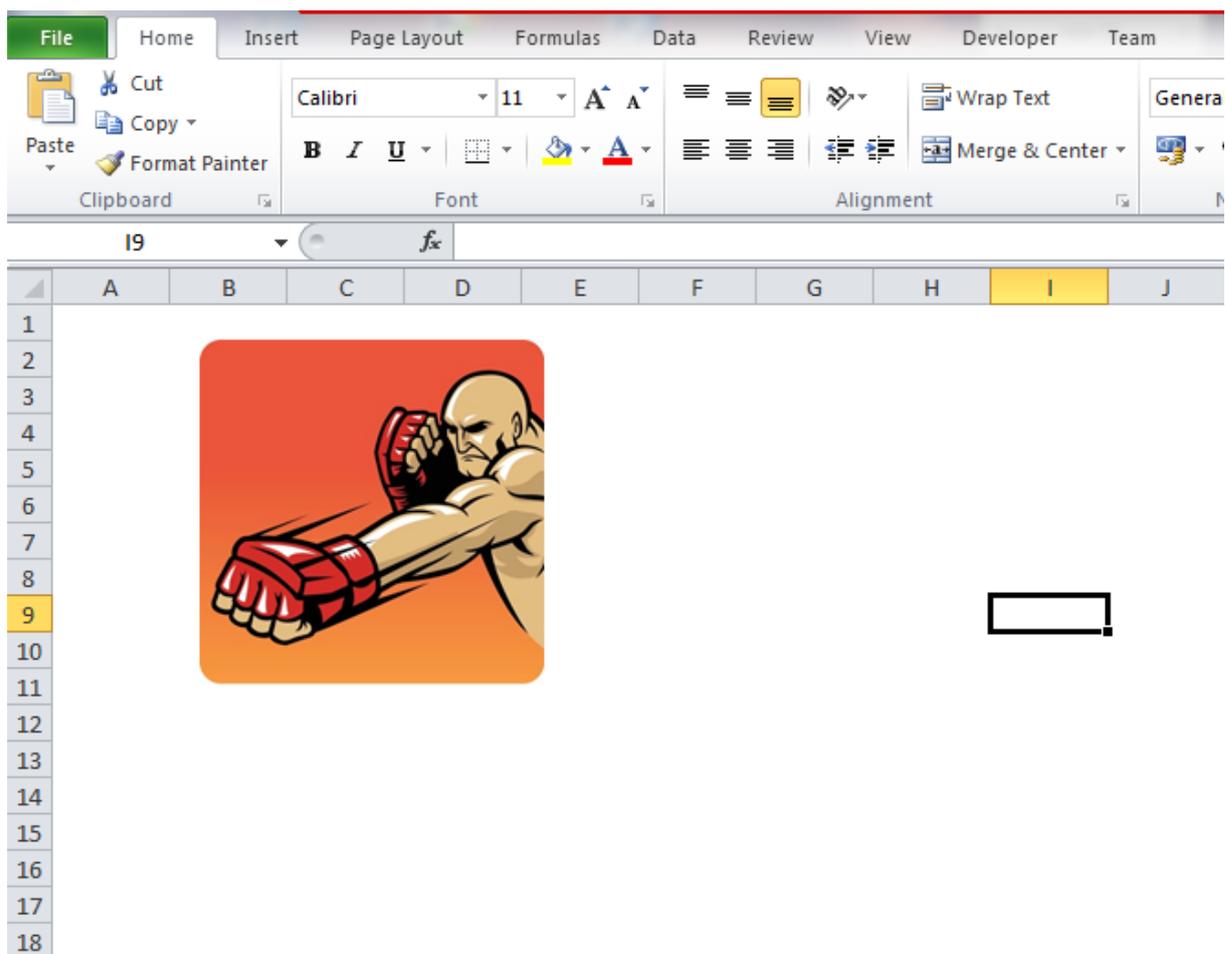
O usuário nesse momento pode apenas visualizar seu resultado de treino, assim como guardar um relatório em arquivo .pdf.



**Figura 40: Tela de confirmação de geração do relatório em PDF.**  
**Fonte: Autoria própria.**

Ao clicar em “Gerar PDF” um relatório, tal qual o presente no Anexo I, é salvo na mesma pasta em que se encontra a Macro Fighter Pro.

Para reiniciar o aplicativo é preciso dar um duplo clique na *label* “Nome do Atleta” e abrirá uma tela de Microsoft Office Excel com o ícone do aplicativo Fighter Pro, basta clicá-lo para reiniciar o programa e realizar uma nova análise.



**Figura 41: Tela de reinicialização da macro.**  
**Fonte: Autoria própria.**

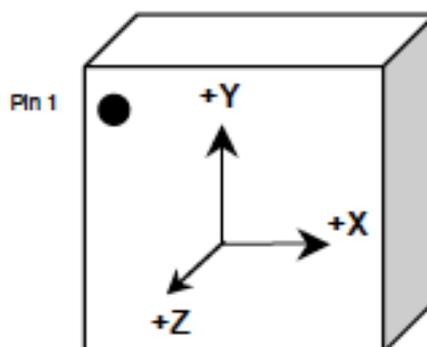
## 7.11 MEDIÇÕES DE GOLPES E CALIBRAÇÃO

De forma a demonstrar a validade do produto desenvolvido nesse projeto, os alunos conduziram experimentos constituídos a partir de embasamento teórico que será detalhado a seguir.

A construção de um pêndulo se mostrou uma opção de teste de impacto, inspirando-se no teste de impacto Charpy, que nos traz uma forma controlada de se obter repetitividade de resultados (SCHWAB, 2013).

### 7.11.1 Medição de Impacto por Acelerômetro

No momento de medição da força ao impacto da massa usamos o acelerômetro na seguinte orientação:



**Figura 42: Orientação do acelerômetro**  
**Fonte: (KIONIX, 2012).**

Onde o plano XY encontra-se na superfície, oposta ao golpe, do saco, com o eixo Z apontando para fora do saco e o eixo Y para cima.

Com o impacto, é produzido uma onda de choque, como será mostrada na Figura 43. Nesse caso, apenas consideramos o primeiro pico, pois apresenta a amplitude máxima do choque, o que é objeto de interesse nesse projeto. Ainda, é no eixo z que medimos o valor de  $\Delta t$ , pela disposição da área de impacto da massa normal a esse eixo.

O princípio do pêndulo consiste que a massa de aço seja solta de uma altura  $h$ , e atinja o saco de boxe, já instrumentado com o acelerômetro, com uma força teórica  $F$  gerando dados que serão enviados via BLE para o aplicativo "Fighter Pro" que faz a gravação dos dados e os exporta através de um arquivo .csv.

Após a observação da dinâmica do sistema, pudemos tomar as seguintes considerações para realizar o modelo teórico:

1. Colisão perfeitamente inelástica
2. Atrito com o ar, desprezível.
3. Desconsiderar deformações no saco de boxe durante o impacto
4. Desconsiderar a massa da haste do pêndulo

5. Desprezar o atrito dos apoios do sistema, rolamento e cordas.
6. O saco de pancadas está inicialmente em repouso

Deste modo, partimos da equação (4) para uma colisão inelástica, pois no impacto da massa com o saco temos esse tipo de comportamento:

$$m_p v_{pi} + m_s v_{si} = (m_p + m_s) v_f \quad (7)$$

Em que,  $m_p$  é a massa considerada do pêndulo,  $v_{pi}$  a velocidade do pêndulo antes do impacto,  $m_s$  a massa do saco de pancadas,  $v_{si}$  a velocidade do saco de pancadas antes do impacto e  $v_f$  a velocidade final do sistema.

Para chegar a  $v_{pi}$  aplicamos a equação (3) para o pêndulo:

$$E_{mecânica} = \left( mgh + m \frac{v^2}{2} \right)_{final} - \left( mgh + m \frac{v^2}{2} \right)_{inicial}$$

$$0 = \left( mgh + m \frac{v^2}{2} \right)_{final} - \left( mgh + m \frac{v^2}{2} \right)_{inicial}$$

$$\left( mgh + m \frac{v^2}{2} \right)_{inicial} = \left( mgh + m \frac{v^2}{2} \right)_{final}$$

Como o pêndulo parte do repouso, a sua velocidade inicial é nula, assim como a altura final já que esta atinge a referência do sistema na colisão. Assim vem:

$$m_p gh = m_p \frac{v_{pi}^2}{2}$$

$$v_{pi} = \sqrt{2gh} \quad (8)$$

Agora substituindo (8) em (7) e aplicando a consideração 6, tem-se:

$$v_f = \frac{m_p \sqrt{2gh}}{(m_p + m_s)} \quad (9)$$

Para o próximo passo foi necessário uma análise do padrão de sinal de um impacto em um gráfico da força pelo tempo. A Figura 43 mostra uma típica curva de resposta de uma força ao impacto.

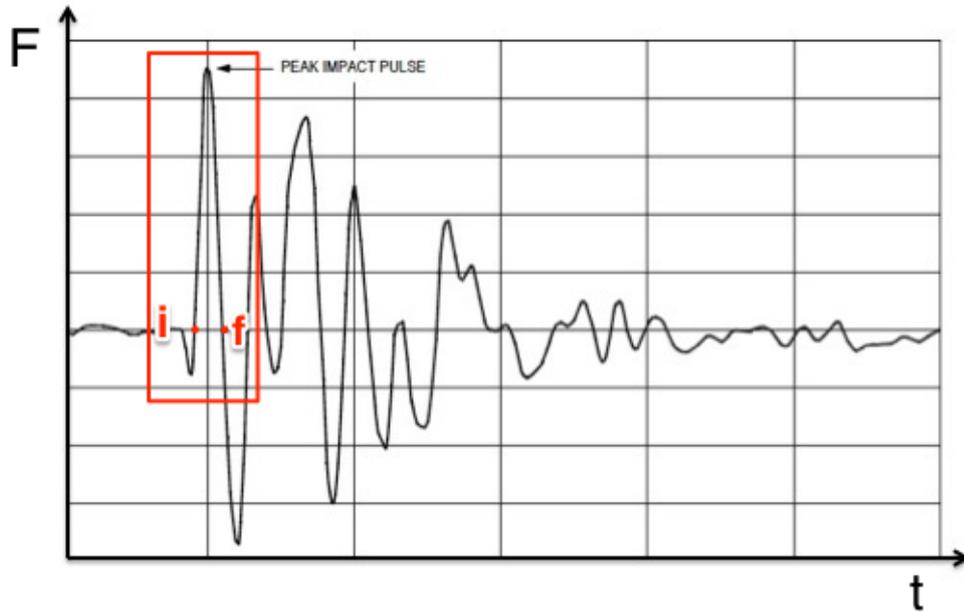


Figura 43: Resposta da força a um impacto através do tempo.  
Fonte: Adaptado de Freescale Semiconductor (2004).

Com o auxílio visual da Figura 43, desenvolve-se a equação (1), considerando o comportamento do saco que contém o acelerômetro preso:

$$\vec{F} = m \cdot \vec{a} \rightarrow \vec{F} = m \cdot \frac{d\vec{v}}{dt}$$

$$\int_i^f \vec{F} \cdot dt = \int_i^f m \cdot d\vec{v}$$

Na Figura 43, vemos que a integral  $\int_i^f \vec{F} \cdot dt$  muito se aproxima da área de um triângulo, então:

$$\frac{F\Delta t}{2} = m_s(v_f - v_i) \quad (10)$$

Agora, substituindo (9) em (10) e considerando 6, vem a equação teórica da força máxima aplicada por um pêndulo de massa  $m_p$  ao atingir um saco de massa  $m_s$ .

$$\mathbf{F} = \frac{2(m_p m_s) \sqrt{2gh}}{(m_p + m_s) \cdot \Delta t} \quad (11)$$

No arquivo .csv que exportamos do aplicativo temos na primeira coluna o tempo de recebimento do dado, na segunda, terceira e quarta colunas

respectivamente os valores de aceleração aferidos nos eixos x, y e z, e finalmente na quinta coluna o valor do módulo da aceleração sofrido pelo impacto da massa. Plotando esses dados, vemos uma curva parecida com a da Figura 43, então, a partir daí temos  $\Delta t$  pela curva de aceleração do eixo z. A altura h é a altura da qual soltamos a massa do pêndulo.

Com isso, fechamos o cálculo teórico e podemos fazer a comparação com a força mensurada pelo sensor, e assim, fazer a sua calibração. Ainda, esses dados serão usados para verificar a repetitividade que o sensor apresenta e poder estimar a sua confiabilidade de dados.

Para emular um golpe em um saco de boxe e verificar a repetitividade do processo, e assim validar a utilização do dispositivo proposto nesse projeto, os alunos construíram um pêndulo. O pêndulo consiste em uma haste rígida de madeira com uma massa de 7 kg de aço 1020 fixada ao fim da haste com parafusos como em Figura 44 e Figura 45 a seguir.



**Figura 44: Massa de aço 1020 com parafusos.**  
Fonte: Autoria própria.



**Figura 45: Massa fixada na haste do pêndulo.**  
Fonte: Autoria própria.

A Figura 46 mostra como o pêndulo bate no saco de 28,4 kg.



**Figura 46: Posição de impacto do sistema saco-pêndulo.  
Fonte: Autoria própria.**

A Figura 47 mostra como é fixado o sensor no saco.



**Figura 47: Modo de fixação do sensor no saco.  
Fonte: Autoria própria.**

A Figura 48 indica o uso do sensor no saco de aproximadamente 80 kg.



**Figura 48: Teste para o saco de 80 kg.  
Fonte: Autoria própria.**

## 8 RESULTADOS E ANÁLISES

Como comentado anteriormente, construiu-se um pêndulo para a modelagem e medição da força teórica variando a altura de soltura da massa do equipamento. Com os dados obtidos, pôde-se visualizar as amplitudes de aceleração do impacto, bem como, sua evolução no tempo.

Assim, a partir do experimento do pêndulo, temos como variáveis do sistema: altura, tempo de impacto e módulo de aceleração.

Inicialmente, realizou-se o teste para um saco de pancadas de 28,4 kg. No teste, a massa do pêndulo é solta a partir de alturas que variam de 70 cm a 110 cm. Ainda, a massa atinge o saco a uma altura de 60 cm do chão. Sendo assim, as alturas que são inseridas no cálculo na equação (11) são: 10, 20, 30, 40 e 50 cm.

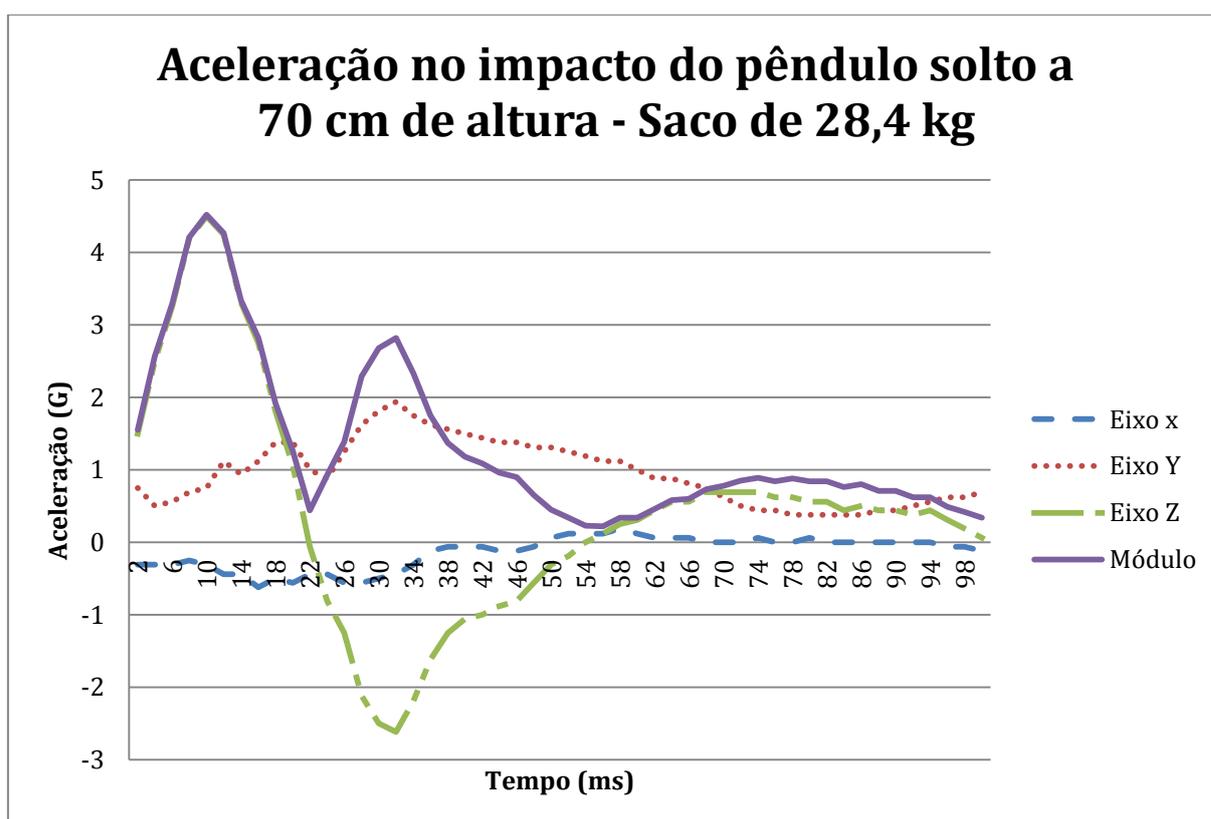
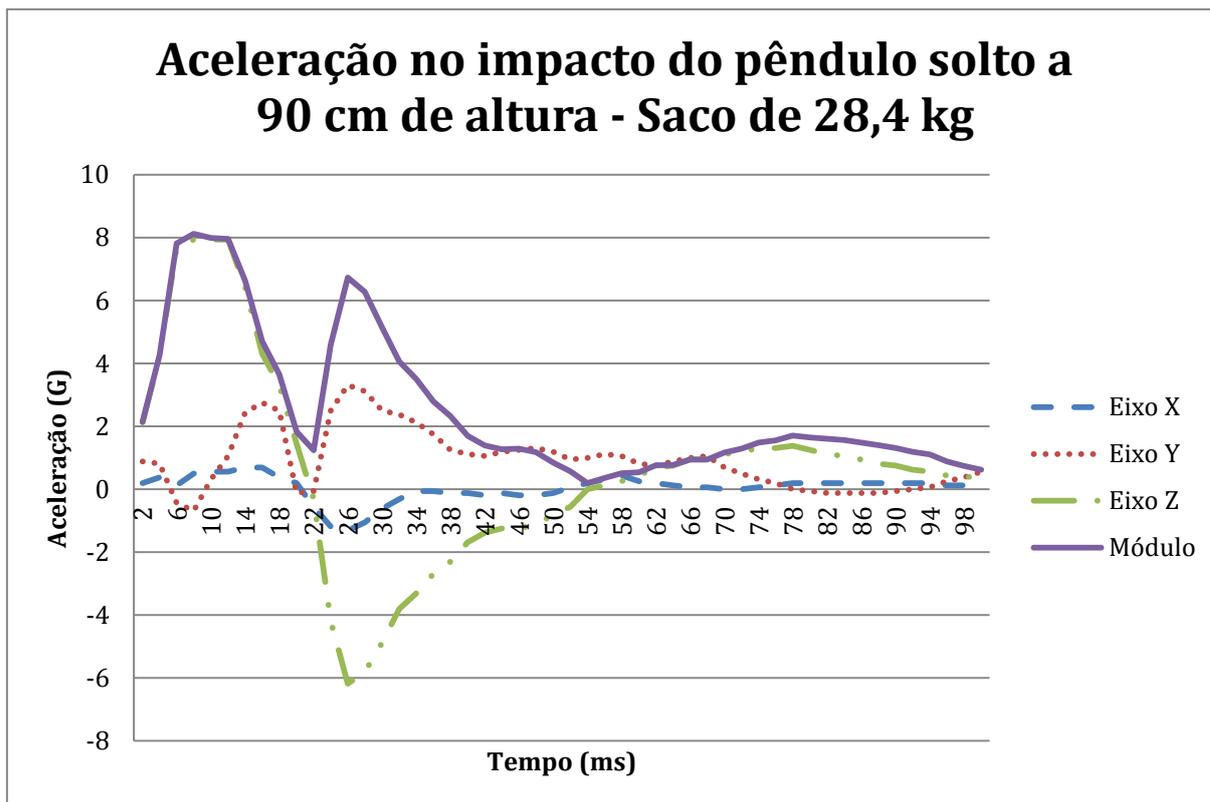


Gráfico 1: Sinal do impacto da massa do pêndulo solta a 70 cm.  
Fonte: Autoria própria.

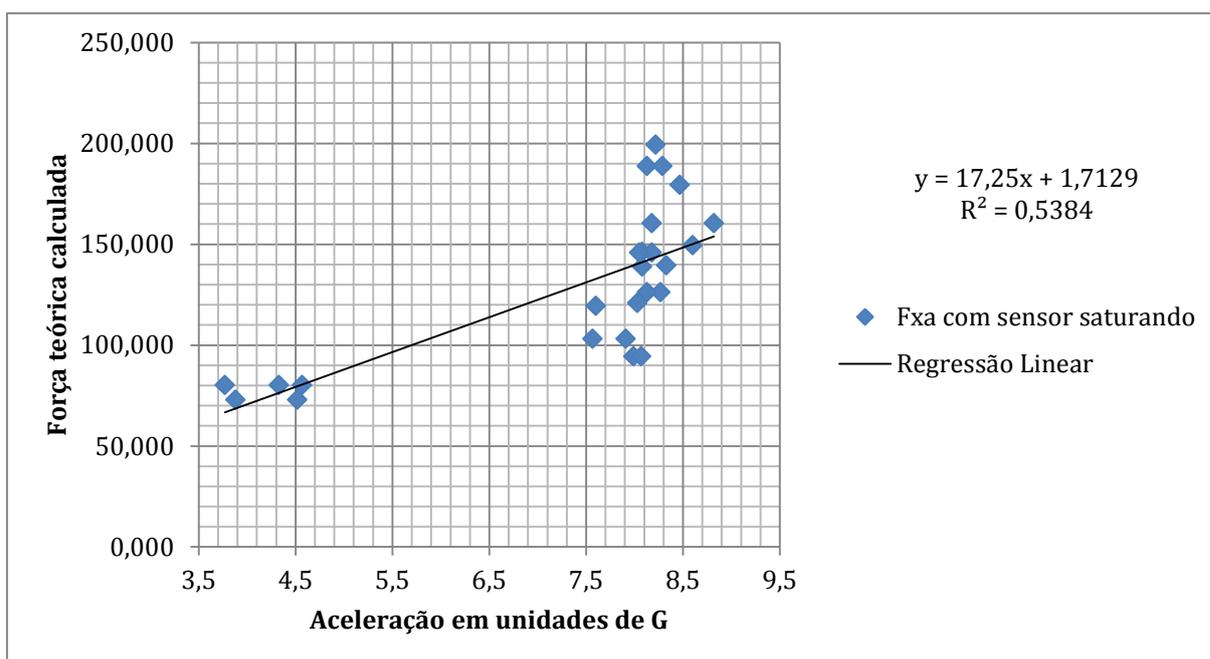


**Gráfico 2: Sinal do Impacto da massa do pêndulo solto a 90 cm com saturação no eixo z.**  
**Fonte: Autoria própria.**

Nesse experimento notou-se que, devido à geometria e distribuição de massa do saco, o sistema proporciona ao sensor uma sensibilidade de um percentual da massa do saco. Ou seja, no cálculo da força, a massa não é o valor total da massa do saco, e sim uma fração dele.

Ainda, ao se analisar os resultados do experimento, notou-se que para algumas alturas o valor obtido pelo acelerômetro havia saturado no eixo z, o eixo de incidência do impacto, como no Gráfico 2. Constatou-se que isso ocorreu justamente pela geometria e distribuição de massa no saco que restringe a operação correta do acelerômetro. Como se considera uma fração da massa do saco, o acelerômetro sofrerá uma aceleração relativa a uma massa inferior, saturando seu *range*.

Podemos visualizar esse comportamento no Gráfico 3, abaixo, em que se constata a saturação nos pontos agrupados à direita no gráfico.



**Gráfico 3: Força teórica calculada x Aceleração medida em unidades de G - Para um saco de pancadas de 28,4 kg**  
 Fonte: Autoria própria.

Constatado isso, partiu-se para um estudo em que o sensor passasse a oferecer confiabilidade nos dados, ou seja, que o sensor não aferisse valores com algum dos eixos saturados.

Para isso, consultando algumas fontes, entre as quais GloboEsporte.com (2013), National Geographic Channel (2013) e Ciências Olímpicas (2011), inferiu-se como força o valor dado pelas referências, como um golpe de atleta, e aceleração o nosso *range* máximo de aceleração em um dos eixos, 8G, e pôde-se estimar uma massa de saco a utilizar, mas ainda aplicou-se um **fator de consideração da massa (FCM)** de massa, que seria o valor da parcela da massa, para estimar a massa real mínima de saco que poder-se-ia utilizar. Então, levando em consideração um valor de soco forte de 3000N pôde-se fazer o cálculo:

$$\vec{F} = m \cdot \vec{a}$$

$$3000N = m \cdot 8G$$

$$m = 38,22 \text{ kg}$$

Entretanto, precisou-se aplicar o fator de consideração médio de massa, que foi calculado como sendo o percentual de massa que foi levado em conta no teste

realizado. Faz-se isso calculando, primeiramente, a força teórica medida com as variáveis de cada amostra,  $h$  e  $\Delta t$ , e verificando a massa considerada de acordo com a aceleração captada pelo sensor. Assim podemos dizer:

$$m\%_{considerada} = \frac{Força_{teoria}}{a} \times \frac{100}{m_s}$$

Para isso avaliou-se o FCM para esse experimento plotando um gráfico com os fatores em ordem crescente de Força. Portanto, verificou-se que o valor de consideração da massa médio tende a 61%, nesse caso.

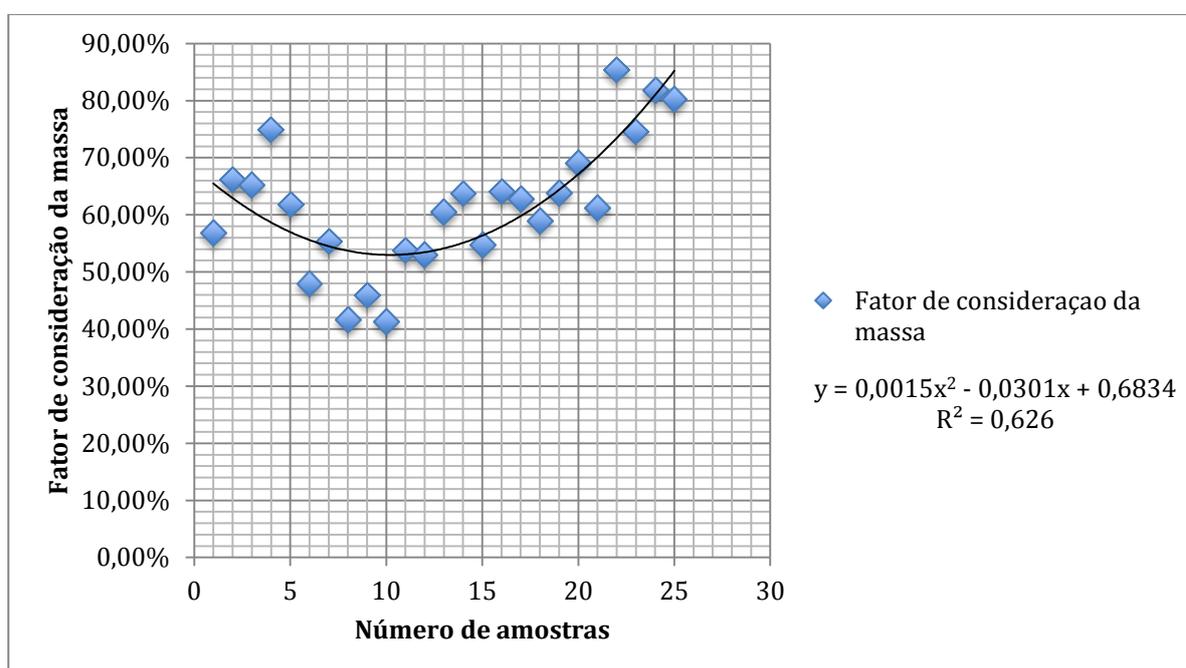


Gráfico 4: Fator de consideração da massa - Para um saco de 28,4 kg.

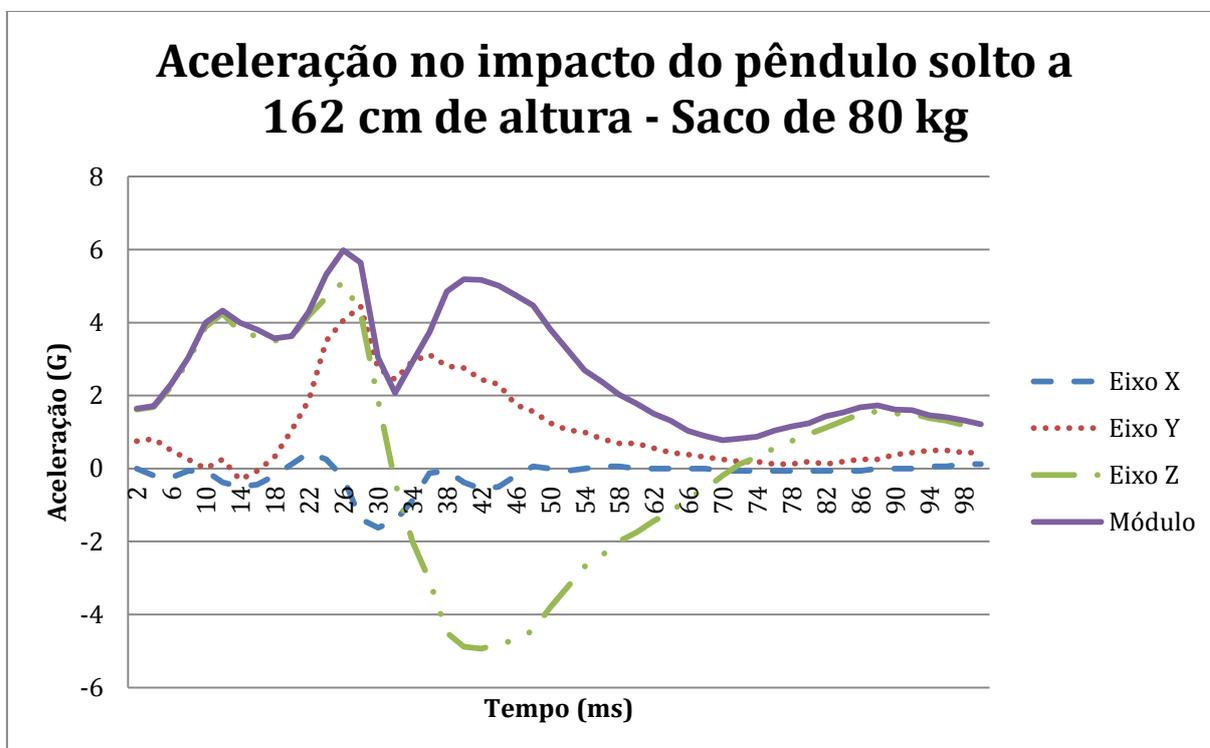
Fonte: Autoria própria.

Então, a massa real do saco que se deveria utilizar para um experimento com confiabilidade dos dados seria maior que 62,66 kg.

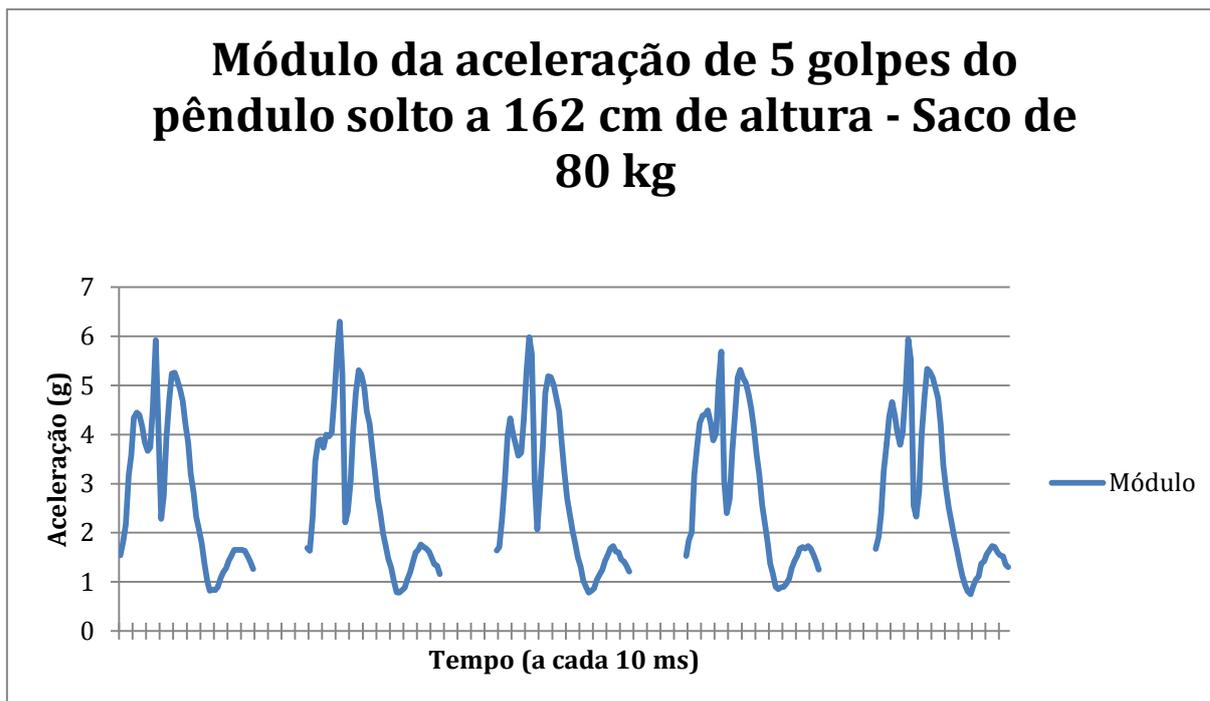
\* Nos gráficos que apresentaremos nesse projeto, procuramos apresentar a melhor regressão obtida através de  $R^2$ , coeficiente de determinação. Segundo Wikipedia (2014),  $R^2$  é uma medida de ajustamento de um modelo estatístico linear generalizado, como a Regressão linear, em relação aos valores observados. O  $R^2$  varia entre 0 e 1, indicando, em percentagem, o quanto o modelo consegue explicar os valores observados. Quanto maior o  $R^2$ , mais explicativo é modelo, melhor ele se ajusta à amostra.

A partir daí, um saco de maior massa foi procurado, e gentilmente a academia Strikers House, em Curitiba, proporcionou-nos esse material com um saco de pancadas de aproximadamente 80 kg, o suficiente para o modelo que planejamos.

Com este material conseguiu-se avaliar os dados com sucesso. Pôde-se observar que o sinal da aceleração não saturou mais em nenhum eixo e em nenhuma amostra. Nesse teste, tal qual o anterior, também se variou a altura do pêndulo para alterar as amplitudes de força no impacto, porém dessa vez a massa atinge o saco a uma altura de 92 cm do chão.



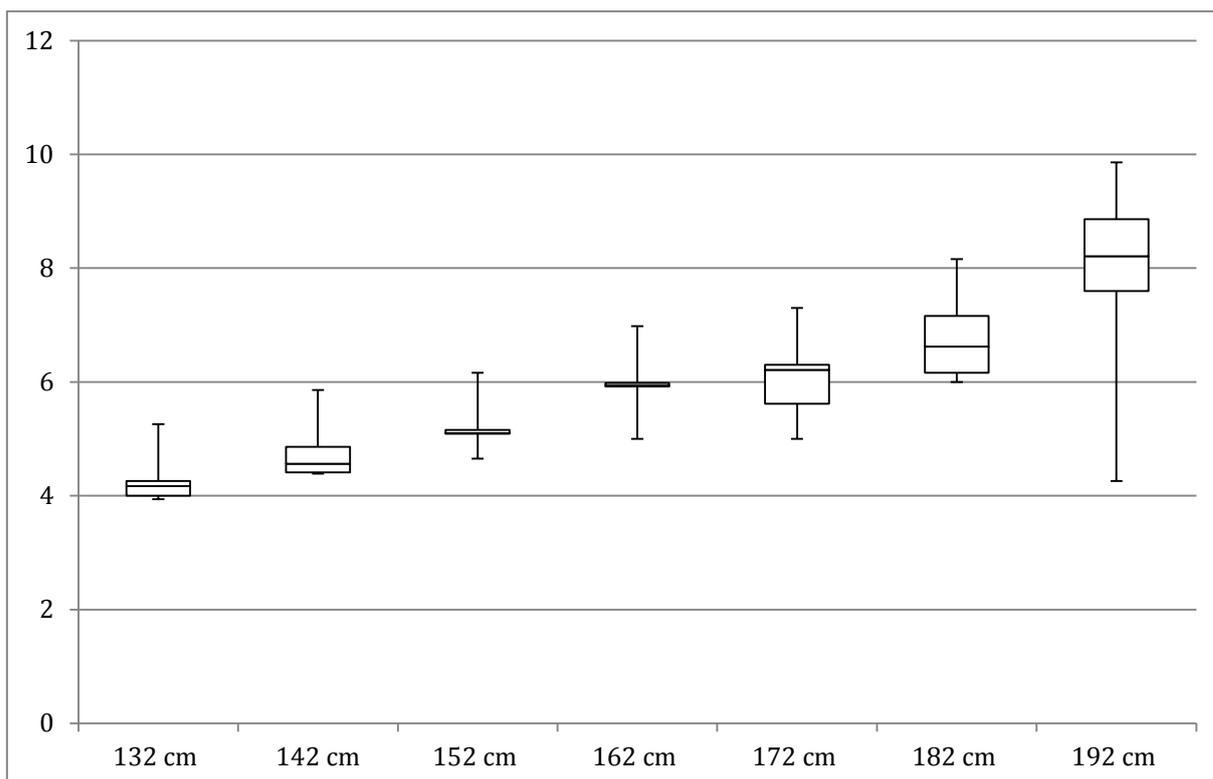
**Gráfico 5: Sinal do impacto da massa do pêndulo solto de 162 cm em um saco de 80 kg**  
 Fonte: Autoria própria.



**Gráfico 6: Sinais obtidos por 5 impactos da massa do pêndulo em um saco de 80 kg**  
**Fonte: Autoria própria.**

Observa-se que para esse novo saco, o padrão de curva do sinal se alterou. Entende-se que isso foi devido a características do saco como: viscosidade, maior massa, maior deformação, maior amortecimento, entre outros fatores decorrentes do impacto e da estrutura material do saco.

O conjunto dos dados do experimento com o saco de 80 kg pode ser visto num diagrama de caixa. Para efeito de interpretação desse diagrama considera-se que as caixas contêm os valores médios da amostra, e as linhas acima e abaixo representando o tamanho dos valores máximo e mínimo, respectivamente, demonstrando a dispersão da amostra.



**Gráfico 7: Resumo da amostra de dados do teste com pêndulo - Para um saco de 80 kg**  
**Fonte: Autoria própria.**

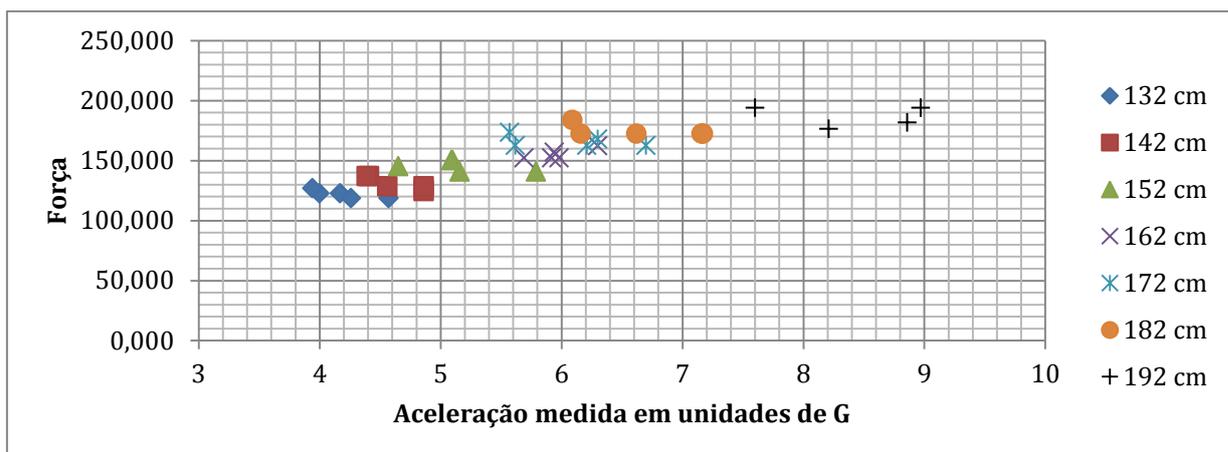
	132 cm	142 cm	152 cm	162 cm	172 cm	182 cm	192 cm
1	4	4,86	5,1	5,92	6,21	6,16	7,6
2	4,57	4,56	5,79	6,3	5,57	6,62	8,21
3	4,17	4,41	5,16	5,98	6,3	7,16	8,97
4	3,94	4,39	4,65	5,69	6,7	6,09	8,86
5	4,26	4,86	5,09	5,94	5,62	7,17	

**Quadro 2: Amostra de dados do teste com o pêndulo - Para saco de 80 kg**  
**Fonte: Autoria própria.**

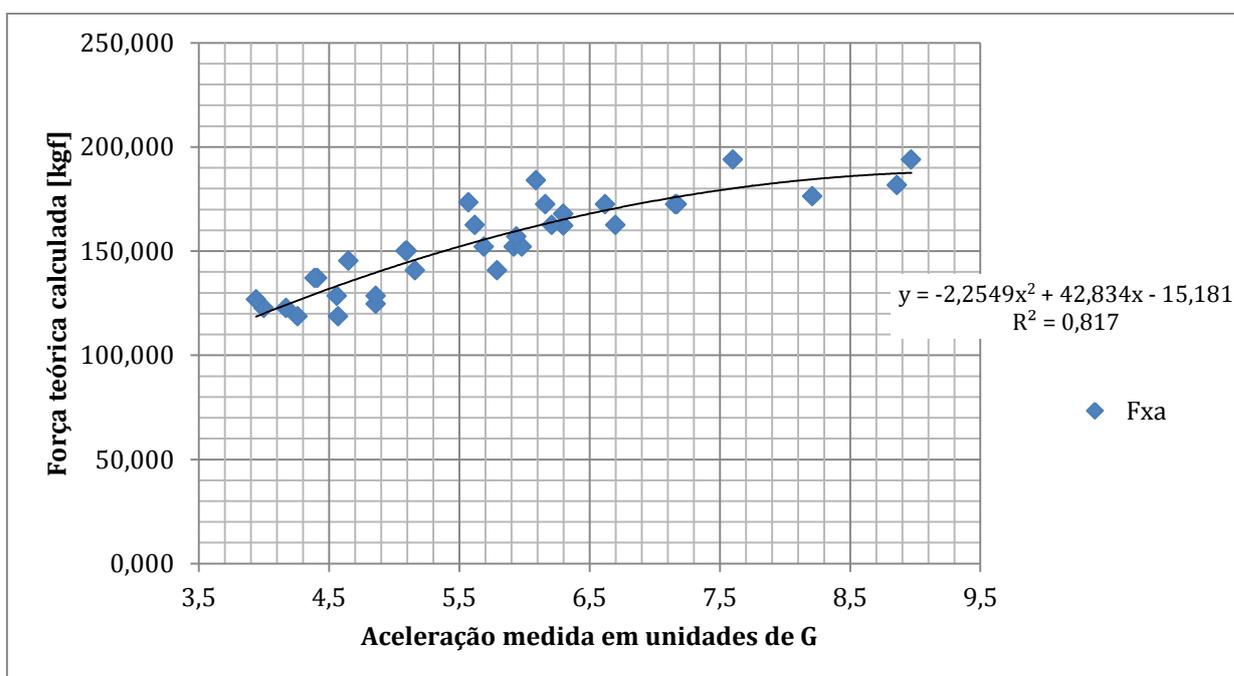
	132 cm	142 cm	152 cm	162 cm	172 cm	182 cm	192 cm
Desvio Padrão	0,25	0,23	0,41	0,22	0,48	0,52	0,86

**Quadro 3: Desvio padrão das amostras colhidas no experimento do pêndulo para um saco de 80 kg**  
**Fonte: Autoria própria.**

Nota-se que os valores de aceleração mensurados pelo acelerômetro se mostraram muito próximos uns dos outros, o que também se confirma analisando o desvio padrão das amostras. Essa correlação também se mostrou quando calculamos a Força Teórica para cada amostra.



**Gráfico 8: Força teórica calculada x Aceleração medida em unidade de G – Para um saco de 80 kg**  
**Fonte: Autoria própria.**



**Gráfico 9: Força teórica calculada x Aceleração medida em unidades de G - Para um saco de 80 kg - Regressão Quadrática**  
**Fonte: Autoria própria.**

Costuma-se entender Força x Aceleração como um gráfico de tendência linear, porém devido às restrições do sistema, tais como perdas de energia, absorção do impacto, geometria e distribuição de massa, temos uma leve variação da massa. Notou-se também que, para o saco de 80 kg, quanto mais intenso era o golpe aplicado (alturas de 182 cm e 192 cm), as acelerações medidas aumentavam em menor proporção em relação ao aumento de intensidade. Concluiu-se que

quanto mais intenso o golpe, mais o saco amortece e se deforma e maiores são as perdas de energia por viscosidade, calor e som.

Assim a melhor correlação ( $R^2$ ) conseguida para a amostra de dados do teste é a de uma equação quadrática.

Tal como no gráfico da Força pela aceleração, tivemos, obviamente, o mesmo comportamento para o fator de consideração da massa, o qual para esse saco de boxe, em sua maioria, tende a 35%.

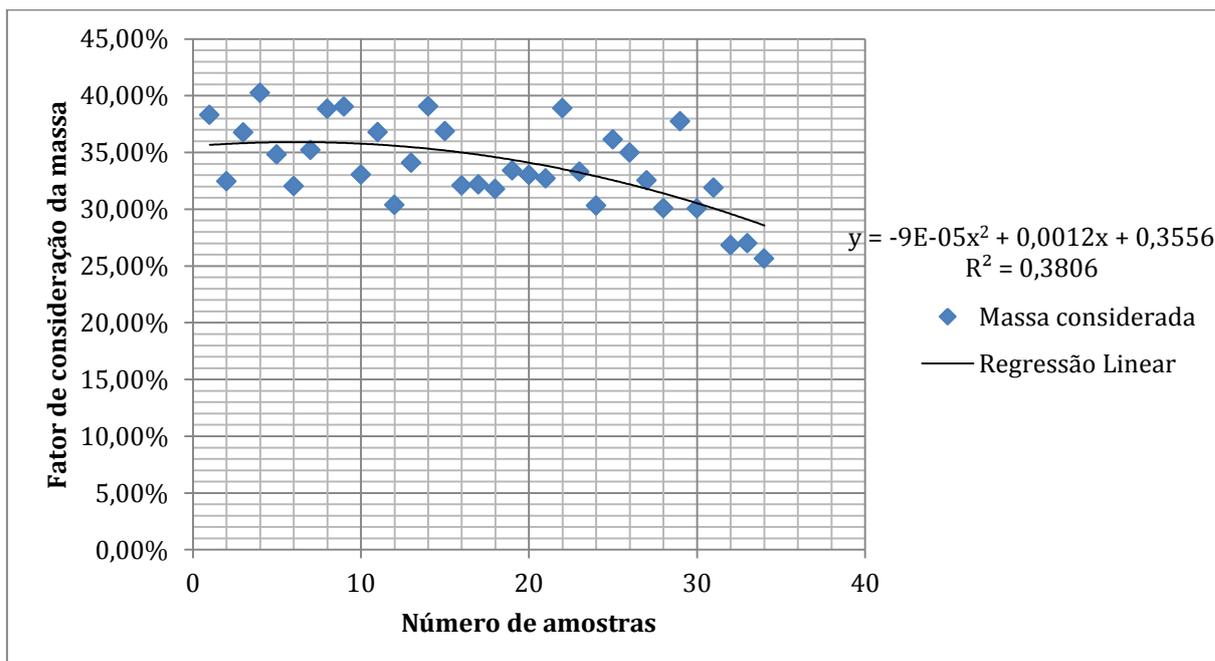


Gráfico 10: Fator de consideração da massa - Para um saco de 80 kg  
Fonte: Autoria própria.

## 9 TEMPO E RECURSOS DESPENDIDOS NO PROJETO

### 9.1 CRONOGRAMA

O cronograma efetivo do projeto é mostrado no Quadro 4 a seguir.

	Maio	Junho	Julho	Agosto	Setembro	Outubro	Novembro	Dezembro	Janeiro	Fevereiro	Março
Proposta do projeto	■										
Montagem módulo acelerômetro e sem fio		■	■								
Teste módulo acelerômetro			■								
Desenvolvimento do aplicativo			■	■	■	■	■	■	■		
Desenvolvimento do Firmware			■	■	■	■	■	■	■	■	
Desenvolvimento gerenciador de resultados									■	■	
Teste de recepção de dados do aplicativo					■	■	■	■	■	■	
Refinamento do uso de dados no app									■	■	
Refinamento da interface gráfica									■		
Testes com o produto final										■	■
Redação do TCC									■	■	
Ajustes do Relatório											■
Apresentação do TCC											■

**Quadro 4 - Cronograma seguido no desenvolvimento do projeto.**  
**Fonte: Autoria própria.**

O cronograma final efetivamente seguido teve algumas discrepâncias com o cronograma idealizado no início do projeto.

## 9.2 TEMPO DEMANDADO

O Quadro 5 a seguir apresenta o intervalo de tempo demandado, em horas, para cada atividade e o tempo total estimado para realização do projeto.

<b>Atividade</b>	<b>Tempo estimado (horas)</b>
Pesquisa de mercado e viabilidade	30
Pesquisa de tecnologia e soluções	120
Desenvolvimento do módulo de <i>software</i> e aplicativo	360
Desenvolvimento do <i>hardware</i> do dispositivo	30
Desenvolvimento do <i>firmware</i> do dispositivo	400
Testes para calibração	60
Testes de produto	10
Elaboração de documentação	250
<b>Total</b>	<b>1260</b>

**Quadro 5 - Intervalo de tempo demandado para realização do projeto.**

Fonte: Autoria própria.

## 9.3 CUSTOS

O Quadro 6 a seguir mostra os gastos estimados do projeto para desenvolvimento completo do projeto. Neste quadro, estão incluídos todos os custos adicionais.

<b>Descrição</b>	<b>Gasto estimado (R\$)</b>
3 PCB CC2541 - Bluetooth 4.0	336,5
2 sacos de boxe profissional	0,00
1 saco de boxe infantil	55,00
CC2540 USB Dongle (Sniffer)	230,00
CC Debugger	0,00
Pêndulo para testes	80,00
Baterias	15,00
Suporte para sensor	65,00
<b>Total</b>	<b>781,50</b>

**Quadro 6 – Estimativa de custos de desenvolvimento do modelo protótipo.**

Fonte: Autoria própria.

## 9.4 ANÁLISE DE RISCOS

Uma análise estimada dos riscos do projeto é apresentada no Quadro 7 a seguir. Nele, são considerados efeitos caso o evento indesejável ocorra, seu impacto no projeto e a ação que se pressupõe compensar o problema. Nela, um valor quantitativo de risco é estimado com base no produto do valor probabilidade de ocorrência e impacto. Os valores possíveis vão de zero a 100, sendo considerado risco alto o valor que exceda o valor 50, baixo se menor que 20 e médio para os outros caso

Grau	Risco	Efeito	Probabilidade	Impacto	Ação
32%	Falha na comunicação com o smartphone	Troca de tecnologia de comunicação	40%	80%	Reduzir: Verificar a compatibilidade de tecnologias
28%	Componentes com desempenho inferior ao esperado	Pesquisa de novos componentes	47%	60%	Eliminar: Pesquisar componentes que se adequem ao projeto
21%	Problemas com a composição do software (embarcado)	Novo projeto de software	30%	70%	Eliminar: Efetuar novo projeto
18%	Problemas com a composição do software (aplicativo)	Novo projeto de software	35%	50%	Eliminar: Efetuar novo projeto

Definições	Baixo	Alto
Probabilidade (P)	0%	100%
Impacto (I)	0%	100%
Grau	P x I	

Legenda		
Grau alto	Grau médio	Grau Baixo

**Quadro 7 – Principais eventos de riscos esperados no decorrer do trabalho e sua caracterização.**  
**Fonte: Autoria própria.**

De fato, a presunção de alta probabilidade de ocorrência para o caso de atrasos no desenvolvimento e problemas com o software causaram grande impacto. Atrasos decorreram principalmente da inexperiência dos autores com a linguagem

para o desenvolvimento do aplicativo, a falta de recursos de verificação de erros, e posterior dificuldade de importação dos mesmos.

## 10 CONSIDERAÇÕES FINAIS

Ao fim deste documento, pode-se concluir que chegamos ao objetivo proposto pelo trabalho, e ainda podemos dizer que superamos nossas primeiras expectativas por conseguir simplificar o layout do aplicativo e a forma de apresentação dos dados, proporcionando uma maior acessibilidade do sistema, fazendo com que a visualização e avaliação dos dados seja feita por uma ferramenta de uso comum (Microsoft Office Excel).

Nota-se que a medição de forças de golpes em um saco de pancadas é dependente dos materiais que o compõe. Para cada tipo de saco ocorre um comportamento característico no impacto, ocorrendo perdas de energia, por atrito, som, aquecimento, deformação, relacionadas a coeficientes de elasticidade e viscosidade do mesmo.

Com a pesquisa realizada em torno do tema, vemos um grande potencial de desenvolvimento do projeto para se conseguir resultados mais confiáveis. Como, por exemplo, o uso de mais sensores acelerômetros, de precisão e com maiores *ranges*, no saco de boxe, chegando a ter uma abrangência para todo o saco de boxe. Outra opção seria um estudo mais aprofundado e empírico do comportamento do saco numa colisão, considerando seus materiais como parâmetros. E ainda, o refinamento das ferramentas estatísticas para assegurar a confiabilidade dos dados.

## REFERÊNCIAS

- 1 NAVAS, Victor Xavier, et al. **Smart Glove**. Electr. & Comput. Eng. Dept., Stony Brook Univ., Stony Brook, NY, USA. Junho, 2012. Disponível em: <[http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=6223202&tag=1](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6223202&tag=1)>. Acesso em: 17 mar. 2014.
- 2 GRAHAM, Brian Barkley. **Using an Accelerometer Sensor to Measure Human Hand Motion**. 11 maio, 2000. Dissertação de Mestrado em Engenharia Elétrica e Ciência da Computação, Massachusetts Institute of Technology.
- 3 SAE. **SAE J211-1 (1995): Instrumentation for Impact Test, Part 1, Electronic Instrumentation**. Warrendale, PA, 1995, 25p.
- 4 MACK, Jacob, et al. **Amateur Boxer Biomechanics and Punch Force**. Biomedical Engineering Department, Wayne State University, Detroit, Michigan, USA. 2010. Disponível em: <<https://ojs.ub.uni-konstanz.de/cpa/article/view/4491>>. Acesso em: 17 mar. 2014.
- 5 FREESCALE SEMICONDUCTOR. **Impact Measurement Using Accelerometers**. Sensor Application Engineering, Singapore, A/P. Rev 3, Outubro, 2004.
- 6 VELLONI, José Eduardo P.. **Pêndulos**. Universidade Estadual de Campinas, 2007. Disponível em: <[http://www.ifi.unicamp.br/~lunazzi/F530\\_F590\\_F690\\_F809\\_F895/F809/F809\\_sem1\\_2007/JoseE\\_Lunazzi\\_2o\\_grau\\_PendulosRF.pdf](http://www.ifi.unicamp.br/~lunazzi/F530_F590_F690_F809_F895/F809/F809_sem1_2007/JoseE_Lunazzi_2o_grau_PendulosRF.pdf)>. Acesso em: 17 mar. 2014.
- 7 SANTOS, Antonio Domingues dos. **Momento Angular**. Instituto de Física, USP, 2010. Disponível em: <[http://moodle.stoa.usp.br/file.php/808/FEP111\\_MomentoAngular.pdf](http://moodle.stoa.usp.br/file.php/808/FEP111_MomentoAngular.pdf)>. Acesso em: 17 mar. 2014.
- 8 YASSAKA, Elcio Junji. **Proposta para analisar a variável força em praticantes de combate de alto nível**. 2004. 51 f. Trabalho de conclusão de curso (Graduação em Educação Física) - Faculdade Dom Bosco, 2004.
- 9 GLOBOESPORTE.COM. **Caratê é a modalidade com golpes mais efetivos, aponta enquete**. Esporte Espetacular, Rede Globo. Rio de Janeiro, 22 de

- dezembro de 2013. Disponível em:  
<<http://globoesporte.globo.com/programas/esporte-espetacular/noticia/2013/12/carate-e-modalidade-com-golpes-mais-efetivos-aponta-enquete.html>>. Acesso em: 17 mar. 2014.
- 10 SANTOS, Saray G. dos, et al. **Aceleração e tempo de duração de impacto em segmentos corporais do judoca durante a realização de *ukemi* em diferentes tipos de tatames.** *Rev. Port. Cien. Desp.* [online]. 2007, vol.7, n.2, pp. 156-166. ISSN 1645-0523.
- 11 WIKIPEDIA. **Nocaute.** Disponível em: <<http://pt.wikipedia.org/wiki/Nocaute>>. Acesso em: 17 mar. 2014.
- 12 TRAMA COMUNICAÇÃO. **Relatório mostra que número de academias no país cresceu 30%.** São Paulo, agosto de 2012. Disponível em: <<http://www.tramaweb.com.br/press-release/9660/relatorio-mostra-que-numero-de-academias-no-pais-cresceu-30.aspx>>. Acesso em: 26 set. 2013.
- 13 ACIL. **Mercado fitness dobra faturamento em dois anos.** Folha de Londrina, 2012. Disponível em: <<http://www.acil.com.br/noticias-detalle/30/07/2012/mercado-fitness-dobra-faturamento>>. Acesso em: 26 set. 2013.
- 14 ACADE. **7 tendências para alavancar a gestão de sua academia em 2013.** Gestão de academias, 21 de janeiro de 2013. Disponível em: <<http://www.acadesystem.com.br/blog/gestao-de-academias/7-tendencias-para-alavancar-a-gestao-de-sua-academia-em-2013/#.UiFXVza1FbE>>. Acesso em: 26 set. 2013.
- 15 FONTES, Prisca. **Academias comemoram crescimento de 30,81%.** O Fluminense, 2 de setembro de 2012. Disponível em: <<http://www.ofluminense.com.br/editorias/empregos-e-negocios/academias-comemoram-crescimento-de-3081>>. Acesso em: 26 set. 2013.
- 16 GONÇALVES, Glauber. **Brasil só perde para EUA em número de academias.** O Estado de S.Paulo, 25 de julho de 2010. Disponível em: <<http://www.estadao.com.br/noticias/impreso,brasil-so-perde-para-eua-em-numero-de-academias,585706,0.htm>>. Acesso em: 26 set. 2013.
- 17 EHLERT, Ana. **Setor de fitness projeta crescimento de 15% ao ano.** Bem Paraná, 3 de junho de 2013. Disponível em:

- <<http://www.bemparana.com.br/noticia/259746/setor-de-fitness-projeta-crescimento-de-15-ao-ano>>. Acesso em: 26 set. 2013.
- 18 AGÊNCIA DE NOTÍCIAS DA PREFEITURA DE CURITIBA. **Academias ganham roteiro de inspeção sanitária**. 2 de setembro de 2011. Disponível em: <<http://www.curitiba.pr.gov.br/noticias/noticia.aspx?codigo=24121>>. Acesso em: 26 set. 2013.
- 19 TAGIAROLI, Guilherme. **Desenvolvedores contam como é possível ganhar dinheiro com aplicativos gratuitos**. UOL Tecnologia, 11 de fevereiro de 2011. Disponível em: <<http://tecnologia.uol.com.br/celulares-telefonia/ultimas-noticias/2011/02/11/entenda-como-desenvolvedores-de-aplicativos-gratuitos-ganham-dinheiro.jhtm>>. Acesso em: 26 set. 2013.
- 20 ARAGÃO, Marianna. **Mais barato, celular com internet deslança**. Folha de São Paulo, 14 de março de 2013. Disponível em: <<http://www1.folha.uol.com.br/mercado/1245963-mais-barato-celular-com-internet-deslanca.shtml>>. Acesso em: 26 set. 2013.
- 21 JUNQUEIRA, Daniel. **Smartphones cada vez mais baratos contribuem para crescimento das vendas no Brasil**. Gizmodo, 14 de Março de 2013. Disponível em: <<http://www.gizmodo.com.br/SmartPhones-cada-vez-mais-baratos-contribuem-para-crescimento-das-vendas-no-brasil>>. Acesso em: 26 set. 2013.
- 22 SEBRAE. **Modelo Canvas estrutura de negócio de forma rápida e simples**. Agência Sebrae. Jornal Conexão Sebrae, MS. Campo Grande, 24 de julho de 2012. Disponível em: <<http://www.agenciasebrae.com.br/noticia/15179173/ultimas-noticias/modelo-canvas-estrutura-negocio-de-forma-rapida-e-simples/>>. Acesso em: 26 set. 2013.
- 23 VOGLER, Elise. **Bluetooth vs Wifi Power Consumption**. Science, Demand Media. 2014. Disponível em: <<http://science.opposingviews.com/bluetooth-vs-wifi-power-consumption-17630.html>>. Acesso em: 17 mar. 2014.
- 24 BALANI, Rahul. **Energy Consumption Analysis for Bluetooth, WiFi and Cellular Networks**. University of California. Los Angeles, 2007. Disponível em: <<http://nesl.ee.ucla.edu/fw/documents/reports/2007/poweranalysis.pdf>>. Acesso em: 17 mar. 2014.
- 25 DIFFEN. **Bluetooth vs Wifi**. Disponível em: <[http://www.diffen.com/difference/Bluetooth\\_vs\\_Wifi](http://www.diffen.com/difference/Bluetooth_vs_Wifi)>. Acesso em: 17 mar. 2014.

- 26 PUC – RJ. **Sistema Mola-Massa-Amortecedor**. Departamento de Engenharia Elétrica, 2014. Disponível em: <[http://www.maxwell.lambda.ele.puc-rio.br/20591/sistema\\_mola\\_amortecedor1.html](http://www.maxwell.lambda.ele.puc-rio.br/20591/sistema_mola_amortecedor1.html)>. Acesso em: 17 mar. 2014.
- 27 WIKIPEDIA. **IOS**. Disponível em: <<http://pt.wikipedia.org/wiki/IOS>>. Acesso em: 17 mar. 2014.
- 28 APPLE. **Core Bluetooth Programming Guide**. 2013. Disponível em: <[https://developer.apple.com/library/ios/documentation/NetworkingInternetWeb/Conceptual/CoreBluetooth\\_concepts/AboutCoreBluetooth/Introduction.html](https://developer.apple.com/library/ios/documentation/NetworkingInternetWeb/Conceptual/CoreBluetooth_concepts/AboutCoreBluetooth/Introduction.html)>. Acesso em: 17 mar. 2014.
- 29 ZHANG, Ye; MARTIKAINEN, Olli; PULLI, Petri; NAUMOV, Valeriy. **Real-time Process Data Acquisition with Bluetooth**. <sup>1</sup>University of Oulu, Department of Information Processing Science, Oulu, Finland. <sup>2</sup>The Research Institute of the Finnish Economy (ETLA), Helsinki, Finland, 2011.
- 30 BLUETOOTH®. **Specification of the Bluetooth System**. Version 4.0, 2010.
- 31 ALECRIM, Emerson. **Tecnologia Bluetooth: o que é e como funciona?**. Infowester, 9 de março de 2013. Disponível em: <<http://www.infowester.com/bluetooth.php>>. Acesso em: 17 mar. 2014.
- 32 BLUETOOTH SIG. **Bluetooth Brand**. 2013. Disponível em: <<http://www.bluetooth.com/Pages/Bluetooth-Brand.aspx>>. Acesso em: 17 mar. 2014.
- 33 BLUETOOTH SIG. **Mobile Telephony Market**. 2013. Disponível em: <<http://www.bluetooth.com/Pages/Mobile-Telephony-Market.aspx>>. Acesso em: 17 mar. 2014.
- 34 TEXAS INSTRUMENTS WIKI. **SensorTag User Guide**. 22 de outubro de 2013. Disponível em: <[http://processors.wiki.ti.com/index.php/SensorTag\\_User\\_Guide](http://processors.wiki.ti.com/index.php/SensorTag_User_Guide)>. Acesso em: 17 mar. 2014.
- 35 TEXAS INSTRUMENTS. **CC2541: 2.4-GHz Bluetooth™ low energy and Proprietary System-on-Chip**. SWRS110C. Janeiro, 2012. Revisado em novembro de 2012.

- 36 TEXAS INSTRUMENTS. **Wireless Connectivity Guide**. 2013.
- 37 TEXAS INSTRUMENTS. **Bluetooth low energy software stack and tools**. BLE-STACK. 12 de novembro de 2013. Disponível em: <<http://www.ti.com/tool/ble-stack?DCMP=wbu-blestack&HQS=ble-stack>>. Acesso em: 17 mar. 2014.
- 38 TEXAS INSTRUMENTS. **Texas Instruments CC2540/41 Bluetooth® Low Energy Software Developer's Guide v1.4.0**. SWRU271F Version 1.4. 2013.
- 39 KATE. **ANT+ vs. Bluetooth Smart – How to Choose Your Heart Rate Monitor**. NuMetrex, 13 de novembro de 2013. Disponível em: <[http://www.numetrex.com/our-pulse/2013/11/bluetooth\\_vs\\_ant.html](http://www.numetrex.com/our-pulse/2013/11/bluetooth_vs_ant.html)>. Acesso em: 17 mar. 2014.
- 40 ANT+. **ANT+ Mobile Products**. 2014. Disponível em: <<http://www.thisisant.com/business/opportunities/mobile/>>. Acesso em: 17 mar. 2014.
- 41 SMITH, Phill. **Comparisons between Low Power Wireless Technologies: Bluetooth low energy, ANT, ANT+, RF4CE, ZigBee, Wi-Fi, Nike+, IrDA and NFC**. CSR, 2011.
- 42 ARAUJO, Edriano Carlos de. **Kit SensorTag: CC2541DK-SENSOR** da Texas Instruments. Revista Saber Eletrônica Ano 48 n.467, Jan Fev, 2013.
- 43 WIKIPEDIA. **Bluetooth Low Energy**. Disponível em: <[http://en.wikipedia.org/wiki/Bluetooth\\_low\\_energy](http://en.wikipedia.org/wiki/Bluetooth_low_energy)>. Acesso em: 17 mar. 2014.
- 44 WIKIPEDIA. **R<sup>2</sup>**. Disponível em: <<http://pt.wikipedia.org/wiki/R%C2%B2>>. Acesso em: 17 mar. 2014.
- 45 KIONIX. **± 2g / 4g / 8g Tri-axis Digital Accelerometer Specifications**. Part Number: KXTJ9-1007 Rev. 4 Dec-2012. 2012. Disponível em: <<http://www.kionix.com/sites/default/files/KXTJ9-1007%20Specifications%20Rev%204.pdf>>. Acesso em: 17 mar. 2014.
- 46 NATIONAL GEOGRAPHIC CHANNEL. **A Força do Golpe de Capoeira**. Raridades da Capoeira (natgeotv.com). 2013. Disponível em:

- <[https://www.youtube.com/watch?v=YFcxmejgzlg&noredirect=1#aid=P-\\_EMBOrUSk](https://www.youtube.com/watch?v=YFcxmejgzlg&noredirect=1#aid=P-_EMBOrUSk)>. Acesso em: 17 mar. 2014.
- 47 CIÊNCIAS OLÍMPICAS. **Qual a Força de um Soco?**. 2011. Disponível em: <<http://cienciasolimpicas.blogspot.com.br/2011/04/qual-forca-de-um-soco.html>>. Acesso em: 17 mar. 2014.
- 48 COELHO, Guilherme Felício Mülbersted. **Caracterização das Forças de Impacto Mecânico no *Jump Kick* dos Professores de *Body Combat***. 2008. Dissertação de Mestrado. Programa de Pós Graduação em Educação Física, UFSC, Florianópolis.
- 49 KAMATH, Sandeep; LINDH, Joakim. **Measuring Bluetooth® Low Energy Power Consumption**. Application Note AN092 (SWRA347a). Texas Instruments. 2012.
- 50 STEWART, Greg. **TI Bluetooth® low energy**. Texas Instruments. Training, Outubro 2013.
- 51 DITTRICH, Max. **Bluetooth Low Energy SensorTag von Texas Instruments, das könnte man mit Bluetooth machen**. 2013. Disponível em: <<http://maxdittrich.com/2013/11/28/bluetooth-low-energy-sensortag/>>. Acesso em: 17 mar. 2014.
- 52 ROPLA ELEKTRONIK. **VARTA CR2032**. 2012. Disponível em: <<http://www.elektronik.ropla.eu/en/magazyn/magazyn/?ic=VMB0068>>. Acesso em: 17 mar. 2014.
- 53 SCHWAB, Rainer. **Charpy Impact Test**. Hochschule Karlsruhe (Karlsruhe University of Applied Sciences), Germany, 2013. Disponível em: <<http://www.youtube.com/watch?v=tpGhqQvftAo>>. Acesso em: 17 mar. 2014.

## ANEXO I



## FighterPro - Relatório

Atleta: **142 CM AJUSTADO**

Data: 15/03/2014

Média de força: **115,22 kgf**

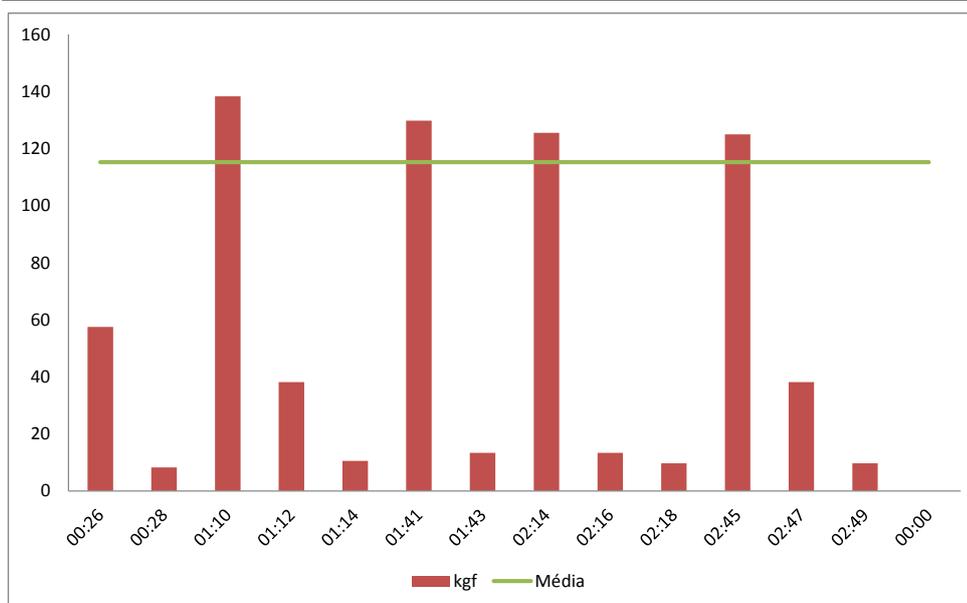
Desvio padrão de  
força dos golpes: **51,78**

Peso do saco: 80 kg

Nº de socos  
acima da média: 4

Golpe mais forte: **138,33 kgf**

Tempo total de  
treino (mm:ss): 02:49



*FighterPro é um sistema desenvolvido para auxiliar o atleta a melhorar seu treino com o foco desejado. O controle de força dos golpes pode mostrar ao atleta em qual parte do treino ele foi mais eficaz, mensurando o seu golpe em unidades de força (kgf).*

*Sistema desenvolvido por Valeria Catherine Zavadzki Santos e Vitor Kazuo Hirozawa.*