



**Universidade de São Paulo
Escola de Engenharia de São Carlos
Departamento de Engenharia Elétrica**

Trabalho de Conclusão de Curso

**O EMPREGO DO ACELERÔMETRO ADXL345 PARA A OBTENÇÃO DA
ACELERAÇÃO, VELOCIDADE E DESLOCAMENTO NO EIXO XY DE
UM SISTEMA PARA LEITURA DE AMBIENTES**

Autor:
Vitor Cabello Pereira da Silva

Orientador:
Prof. Dr. Carlos Dias Maciel

São Carlos, 2018

VITOR CABELLO PEREIRA DA SILVA

**O EMPREGO DO ACELERÔMETRO ADXL345 PARA A
OBTENÇÃO DA ACELERAÇÃO, VELOCIDADE E
DESLOCAMENTO NO EIXO XY DE UM SISTEMA PARA
LEITURA DE AMBIENTES**

Trabalho de Conclusão de Curso apresentado à Escola de
Engenharia de São Carlos, da Universidade de São Paulo
Curso de Engenharia Elétrica com ênfase em Eletrônica

ORIENTADOR: Prof. Dr. Carlos Dias Maciel

São Carlos

2018

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da
EESC/USP com os dados inseridos pelo(a) autor(a).

Cabello Pereira da Silva, Vitor
CS586o O emprego do acelerômetro ADXL 345 para a obtenção
o da aceleração, velocidade e deslocamento no eixo XY de
um sistema para leitura de ambientes / Vitor Cabello
Pereira da Silva; orientador Carlos Dias Maciel. São
Carlos, 2018.

Monografia (Graduação em Engenharia Elétrica com
Enfase em Eletrônica) -- Escola de Engenharia de São
Carlos da Universidade de São Paulo, 2018.

1. Acelerômetro. 2. Raspberry Pi. 3. Velocidade. 4.
Deslocamento. 5. Integração Trapézoidal. I. Título.

FOLHA DE APROVAÇÃO

Nome: Vitor Cabello Pereira da Silva

Título: "O emprego do Acelerômetro ADXL345 para a obtenção da aceleração, velocidade e deslocamento no Eixo XY de um sistema para leitura de ambientes"

Trabalho de Conclusão de Curso defendido e aprovado
em 26 / 11 / 2018,

com NOTA 8,5 (alto , muito), pela Comissão Julgadora:

Prof. Associado Carlos Dias Maciel - Orientador - SEL/EESC/USP

Mestre Victor Hugo Batista Tsukahara - Doutorando - SEL/EESC/USP

*Mestre Talysson Manoel de Oliveira Santos - Doutorando -
SEL/EESC/USP*

Coordenador da CoC-Engenharia Elétrica - EESC/USP:
Prof. Associado Rogério Andrade Flauzino

Agradecimentos

Agradeço ao professor Dr. Carlos Dias Maciel, por ter aceitado me orientar no presente projeto, me auxiliando nos pontos de dúvidas e em seu desenvolvimento. Agradeço também pelo incentivo em extrair o melhor dos seus alunos durante as aulas que são por ele ministradas.

Ao meu pai, Edison, minha mãe Elaine, meu irmão Rafael e minha namorada Carolina pelo apoio no desenvolvimento do projeto e principalmente pelo apoio e compreensão em todos os momentos de dificuldade e momentos de privações que a Universidade requer para a obtenção de um aprendizado consolidado.

Aos meus amigos de sala de aula, que foram fundamentais durante todo o meu período de graduação, ajudando nos estudos para as provas, nos trabalhos que foram realizados e nos períodos de divertimento para deixar o ambiente de estudo mais leve descontraído. Gostaria de agradecer em especial ao meu amigo Jordão, por sempre estar disposto a me ajudar e a me apoiar em todos os pontos de dúvidas e incertezas.

Por fim, a todos os professores e funcionários do Departamento de Engenharia Elétrica e Computação (SEL/EESC/USP) que fazem todo o trabalho para prover aos alunos um ensino com a melhor qualidade possível.

Resumo

Os acelerômetros possuem mais de 100 anos de existência e ao longo dos anos vêm sendo cada vez mais aplicados em diversos projetos na indústria e na ciência. Em contrapartida, o micro computador Raspberry Pi se tornou muito popular nos últimos tempos para projetos que envolvem a programação de componentes eletrônicos. O presente projeto visou utilizar um acelerômetro digital ADXL 345 e uma Raspberry Pi 2 para a obtenção da aceleração, velocidade e deslocamento no plano XY de um carrinho. Para a obtenção da velocidade e deslocamento foi utilizado somente os dados fornecidos pelo acelerômetro, sendo realizado o método da integral trapezoidal a fim de minimizar os erros relativos à integração da aceleração. Por fim, foram realizados testes para verificar a efetividade do método utilizado, a sua exatidão e desempenho.

Palavras-chave: acelerômetro; Raspberry Pi; velocidade; deslocamento; integração trapezoidal.

Abstract

The accelerometers have more than 100 years of existence and over the years have been increasingly applied in various projects in industry and science. In contrast, the micro computer Raspberry Pi has become very popular in recent times for projects that involve the programming of electronic components. The present project aimed to use an ADXL 345 digital accelerometer and a Raspberry Pi 2 to obtain acceleration, velocity and displacement in the XY plane of a cart. To obtain the velocity and displacement, only the data provided by the accelerometer were used, and the method of the trapezoidal integral was performed in order to minimize the errors related to the acceleration integration. Finally, tests were performed to verify the effectiveness of the method used, its accuracy and performance.

Key-words: accelerometer; Raspberry Pi; velocity; displacement; trapezoidal integration.

Lista de Figuras

Figura 1: Acelerômetro B&K Modelo 4303 – provável primeiro acelerômetro comercial piezoelétrico (~1945 - ~1948). [2]	19
Figura 2: Exemplo de um acelerômetro MEMS. [18].....	20
Figura 3: Mapeamento do conector GPIO – Raspberry Pi 2 Model B. [7].....	26
Figura 4: Raspberry Pi 2 Model B.	27
Figura 5: Configuração da estrutura do protocolo de comunicação I2C [13].....	29
Figura 6: Início de comunicação do protocolo I2C [13].....	30
Figura 7: Final da comunicação do protocolo I2C [13].....	30
Figura 8: Configuração dos pinos do ADXL345 [16].....	31
Figura 9: Diagrama de conexão para funcionamento da interface I2C [15].....	32
Figura 10: Diagrama de blocos funcionais do ADXL 345 [15].....	33
Figura 11: Diagrama com a conexão entre o ADXL 345 e Raspberry Pi 2	35
Figura 12: Entrada no protocolo de comunicação I2C da Raspberry Pi Model B.....	37
Figura 13: Fluxograma de leitura e atuação do acelerômetro ADXL345	40
Figura 14: Relação de ângulos com os vetores x, y e z.....	42
Figura 15: Demonstração da integração trapezoidal [17].....	43
Figura 16: Demonstração da integração trapezoidal com a divisão da altura em intervalos menores[17].....	44
Figura 17: Fluxograma do algoritmo implementado para a aquisição da velocidade e deslocamento do componente.....	46
Figura 18: Calibração do Acelerômetro ADXL 345 no eixo Z [19].....	47
Figura 19: Resultado da calibração do componente no eixo Z.....	47
Figura 20: Calibração do Acelerômetro ADXL 345 no eixo X [19].....	48
Figura 21 Resultado da calibração do componente no eixo X.....	48
Figura 22: Calibração do Acelerômetro ADXL 345 no eixo Y [19].....	49
Figura 23: Resultado da calibração do componente no eixo Y.....	49
Figura 24: Posição inicial do acelerômetro.....	50
Figura 25: Decomposição da força da gravidade na posição inicial.....	50
Figura 26: Valores de aceleração, velocidade e deslocamento instantâneos para a posição inicial.....	51
Figura 27: Posição angular dos eixos X e Y para o componente fixo em sua posição inicial.....	51
Figura 28: Deslocamento de 30 cm ao longo do eixo X.....	52

Figura 29: Deslocamento de -30 cm ao longo do eixo X.....	53
Figura 30: Deslocamento de -30 cm ao longo do eixo Y.....	53

Lista de Tabelas

Tabela 1: Relação dos pinos do ADXL345.....	31
Tabela 2: Relação de pinos utilizados na Raspberry Pi Model B V1.1 para a conexão com o ADXL 345.....	34
Tabela 3: Relação entre a resolução e a taxa de conversão para o componente ADXL345.....	38

Índice

1. Introdução	15
1.1. Motivação.....	15
1.2. Objetivos.....	16
1.3. Organização do trabalho.....	17
2. Embasamento teórico	18
2.1. Transdutor – Acelerômetro.....	19
2.2. A raspberry Pi.....	21
2.3. O conceito de aceleração, velocidade e deslocamento.....	21
3. Materiais e Métodos	25
3.1. Materiais.....	25
3.1.1. A Raspberry Pi 2.....	25
3.1.2. A programação Python.....	28
3.1.3. A comunicação I2C.....	29
3.1.4. O ADXL 345.....	30
3.2. Métodos.....	33
3.2.1. Montagem Física.....	35
3.2.2. Implementação do código do sistema em linguagem Python.....	36
3.2.2.1. Implementação da comunicação serial.....	36
3.2.2.2. Implementação do acelerômetro ADXL345.....	37
3.2.2.3. Implementação para obtenção da velocidade e deslocamento.....	41
4. Resultados	47
4.1 Implementação do algoritmo da leitura e funcionamento do acelerômetro ADXL345.....	47
4.2 Implementação do algoritmo para aquisição de velocidade e deslocamento do sistema.....	50

5. Conclusão	55
6. Referências Bibliográficas	57

Introdução

1.1 Motivação

Desde o seu surgimento, em 1880 [1], até os dias atuais, o acelerômetro vem sendo empregado cada vez mais pela indústria tecnológica em seus projetos. Dentre os diversos detectores de posição e orientação presentes no mercado o acelerômetro é o mais comum a ser utilizado devido a sua ampla faixa de aplicações [11].

Por conta desta vasta faixa de aplicações, ao longo dos anos houve uma grande evolução em seu funcionamento e em suas características, surgindo novas fontes de pesquisas e de usos industriais (principalmente nos telefones celulares).

O presente trabalho pretende relacionar e elucidar o conceito de aceleração, velocidade e deslocamento por meio de um acelerômetro digital ADXL 345. A fins de estudo a implementação do acelerômetro ocorre de um desmembramento de um projeto de construção de um robô para a leitura e projeção virtual de ambientes físicos, sendo o acelerômetro ADXL 345 componente do projeto para mensurar a velocidade e quantificar o deslocamento do robô no plano XY, possibilitando assim a compreensão do seu deslocamento a medida em que a leitura do ambiente é feita.

A implementação do funcionamento do componente será realizada em um microcomputador Raspberry Pi 2 e o desenvolvimento implementado em linguagem Python. A comunicação entre o acelerômetro ADXL 345 e a Raspberry será feita utilizando o protocolo de comunicação serial I2C.

Para eliminar a interferência da força gravitacional na aceleração de deslocamento do sistema serão utilizados dois métodos. O primero visa eliminar a força gravitacional inicial presente no sistema, desta forma, será zerado em cada um dos eixos a componente gravitacional inicial. O segundo método visa eliminar a componente gravitacional instântanea, desta forma, será calculado o ângulo de defasagem para verificar a componente gravitacional que influêncie no momento da aferição em relação a componente inicial, e então esta componente será descontada.

Por fim, para a aquisição da velocidade e deslocamento será utilizado o método de integração trapezoidal, onde a propagação de erros por conta da constante integrativa é minimizada. Os resultados foram analisados através de testes que serão expostos nas seções a seguir.

1.2 Objetivos

Este trabalho foi idealizado de forma a cumprir os seguintes objetivos:

- Compreender o correto funcionamento, o processo de comunicação e implementação do acelerômetro ADXL 345;
- Estudar e apresentar uma forma de comunicação entre a Raspberry Pi e o acelerômetro ADXL 345;
- Utilizar uma linguagem de programação para realizar a comunicação e o correto funcionamento entre a Raspberry Pi 2 e o acelerômetro ADXL 345;
- Estudar uma forma de viabilidade de obtenção da velocidade de deslocamento por meio do acelerômetro ADXL 345 sem que ocorra a interferência da constante gravitacional e da constante integrativa no resultado final do sistema;
- Realizar testes individuais do funcionamento de cada um dos pontos citados acima e posteriormente fazer a integração em um único sistema a ser implementado em um projeto de construção de um robô para a análise e projeção de ambientes.

1.3 Organização do trabalho

O projeto contempla o uso de diversos componentes e ferramentas que serão detalhados ao longo desta monografia. Desta maneira, para fazer o uso correto de todos esses componentes e das ferramentas necessárias para o seu funcionamento foi necessário um aprofundado estudo teórico de cada um dos componentes. Em seguida foi necessária à realização do desenvolvimento de implementação dos componentes e ideias separadamente, para ao fim ocorrer uma integração entre as partes e apresentar os resultados e as possíveis conclusões a serem tomadas.

Desta maneira, o trabalho é disposto em 5 capítulos. O Capítulo 1 é composto pela introdução, nesta visa-se expor o contexto dos acelerômetros e sua constante demanda de evolução, a motivação da realização do trabalho e como foram traçados os objetivos para que o seu desenvolvimento fosse realizado com objetividade e foco em resultados. O Capítulo 2 visa introduzir os conceitos e fundamentos teóricos que serviram de base para o desenvolvimento do projeto. Cada componente utilizado é estudado em sua teoria explanando as possibilidades de

execução e funcionamento. No Capítulo 3 são feitas duas principais divisões, Materiais e Métodos. Em materiais são expostos as definições das ferramentas utilizadas para o desenvolvimento do projeto. Em métodos são expostos os desenvolvimentos individuais do projeto, sendo demonstrados os métodos e algoritmos utilizados para sua implementação e a composição de suas partes para a composição do projeto final. No Capítulo 4 são apresentados os resultados individuais de cada etapa exposta no capítulo 3, ocorrendo uma breve discussão para cada resultado apresentado. Por fim, no Capítulo 5 é apresentado a conclusão do projeto, detalhando os conceitos apresentados e quais ensinamentos podem-se retirar de cada um deles, além de seus possíveis problemas e limitações.

2. Embasamento teórico

Neste capítulo serão apresentados os temas e linhas de pesquisa e desenvolvimento que foram utilizados como base para o desenvolvimento deste projeto.

2.1 Transdutor – Acelerômetro

O acelerômetro é um dispositivo largamente aplicado na indústria e na ciência que tem por objetivo medir a aceleração própria de um sistema. O seu surgimento se dá no ano de 1880, com os irmãos Pierre e Jacques Curie [1] com a descoberta do cristal piezoelétrico, onde o estresse mecânico é conectado a carga superficial de um cristal. Por apresentar um conceito com diversas possibilidades de aplicação prática, diversos estudos foram voltados para a produção de um componente com viés comercial. Foi quando no ano de 1938 [1], Hans J. Meier desenvolveu no *Massachusetts Institute of Technology* (MIT) o primeiro acelerômetro comercial, um acelerômetro medidor de tensão. Este acelerômetro consistia em filamentos de fios ligados em 4 tiras de suporte suportando um bloco de aço de 3.434 libras [3]. Esse conjunto conseguia medir uma aceleração de 2g com vibrações de até 4Hz, suportando estudos dos efeitos de terremotos em tubos de água. Em seguida, no ano de 1943 Louis Statham fundou a Statham Instrument Company, uma empresa de fabricação de transdutores medidores de tensão de fios não acoplados, sendo a primeira empresa neste ramo [2].

Com a evolução das pesquisas e testes foi comprovado que os acelerômetros piezoelétricos eram mais confiáveis do que os medidores de tensão, uma vez que possuem resposta plana em frequências de até 10.000 Hz enquanto os medidores de tensão até 200 Hz [1]. Essas diferenças eram muito importantes para a aplicação nas quais os acelerômetros eram empregados, como por exemplo na detecção de falhas mecânicas em aeronaves.

Desta forma, na década de 1940 iniciaram-se os estudos mais a fundo para a produção comercial em grande escala dos acelerômetros piezoelétricos. A primeira empresa a comercializar estes componentes foi a Brüel & Kjær, em 1942 [2]. O primeiro modelo comercial desenvolvido pela empresa foi o 4303, constituído de material salino (cristais de sal Rochelle), suas sensibilidades variavam de 30-35 mV/g e suas frequências ressonantes entre 2-3 kHz. Por volta dos anos 50 os elementos cerâmicos substituíram os cristais de sal de Rochelle, dobrando a sensibilidade dos acelerômetros e aumentando sua frequência de ressonância para aproximadamente 5kHz.



Figura 1: Acelerômetro B&K Modelo 4303 – provável primeiro acelerômetro comercial piezoelétrico (~1945 - ~1948). [2]

Os acelerômetros de compressão foram introduzidos pela B&K no final dos anos 50 e em seguida ocorreram modificações de design em 1964 para reduzir a suscetibilidade ao carregamento e a tensão da base. A empresa trabalha até os dias atuais com componentes eletrônicos como transdutores para sons e vibrações, focando suas soluções em sistemas completos, sendo uma das empresas com o sistema mais “turnkey” do mercado [2].

Em seguida diversas outras empresas especializadas em fabricar acelerômetros piezoelétricos foram sendo fundadas. Com o aumento da competitividade no mercado, novas tecnologias foram empregadas nos materiais de fabricação dos componentes, deixando-os mais robustos, com faixas de funcionamento em frequência e temperatura maiores e com novas concepções de montagem, estruturas e funcionamento. Um bom exemplo desse desenvolvimento pode ser explicitado pelo lançamento do modelo 818 da Kistler Instrument Company, em 1968, um componente que foi revolucionário na época por apresentar apenas um cabo para sua alimentação e obtenção de sinais, baixa impedância e em um tamanho convencional para a época [2]. Essas características eram consideradas importantes pois entregavam ao usuário praticidade de instalação, de conexão e medição.

Após a era dos acelerômetros piezoelétricos, nos anos 90, os acelerômetros entraram na era da eletrônica digital. Uma mudança significativa na tecnologia dos acelerômetros foi à utilização do silício para a sua fabricação. Muitos acelerômetros atuais fazem uso deste material em suas construções devido a sua composição e características físicas, é o caso dos acelerômetros piezoresistentes, acelerômetros de capacitância variável e os sistemas MEMS, ou, sistemas microeletromecânicos [4].

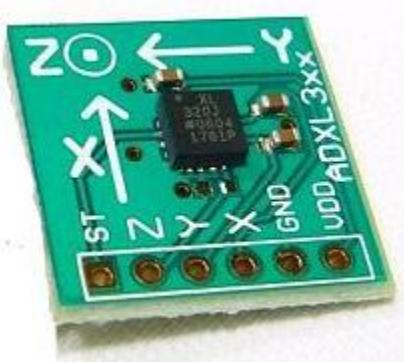


Figura 2: Exemplo de um acelerômetro MEMS. [18]

Atualmente, os acelerômetros são componentes muito difundidos no mercado, com larga aplicação na indústria e ciência. São utilizados em diversos setores como sistemas mecânicos, no setor de saúde, sistemas de navegação de aeronaves e mísseis, em automóveis, em aparelhos eletrônicos portáteis como drones, celulares e tablets, entre outros. Devido elevada taxa de aplicação esses componentes tiveram que ser alterados para atender aos diversos requisitos de aplicação, atualmente podem ser encontrados em diversos tamanhos, massas, sensibilidades, eixos de medição, faixas de amplitude e frequência [4].

Desta forma, pode concluir que os acelerômetros são sensores ou transdutores que medem acelerações. Com o passar dos anos diversas formas de detecção de aceleração foram empregadas por meio deste componente. A aceleração é uma medida da variação da velocidade no tempo e pode ser obtida segundo uma, duas ou três direções, utilizando acelerômetros uni, bi ou triaxiais, respectivamente. Tipicamente, os acelerômetros são constituídos por uma massa de reação suspensa por uma estrutura estacionária. Este aparelho pode ser visto como um transdutor massa-mola, que se encontra no interior de um sensor. Sempre que este acelera, a inércia faz com que a mola resista. A força exercida pela massa é equilibrada pela mola e, como o deslocamento permitido pela mola é proporcional à força aplicada, a aceleração do corpo é proporcional ao deslocamento da massa. A magnitude da aceleração aplicada é vista por

instrumentos ou circuitos como um impulso elétrico. O impulso elétrico é depois processado por circuitos externos, podendo ser usado em inúmeras aplicações. Pode-se utilizar acelerômetros para medir não apenas acelerações (dinâmicas), como também inclinação, rotação, vibração, colisão e gravidade (acelerações elásticas), constituindo assim um aparelho de elevada utilidade para projetos na área da eletrônica e robótica [5].

2.2 A Raspberry Pi

A Raspberry Pi é um computador com tamanho e custo extremamente reduzido que permite a utilização de editores de texto, planilhas, o acesso a navegadores de internet e principalmente o desenvolvimento de programas utilizando uma série de linguagens de programação, como a linguagem C++, Java, Python entre outras [10].

Desenvolvido pela Raspberry Pi Foundation com um intuito didático de estimular o estudo da ciência da computação, todos os componentes do microcomputador foram desenvolvidos a fim de otimizar o seu custo sem nenhum limitador importante de desempenho. O seu preço é de aproximadamente US\$ 25 a US\$ 35 [10], e, por este motivo, a Raspberry Pi vem sendo largamente utilizada para o desenvolvimento de projetos eletrônicos e soluções de problemas.

A Raspberry Pi possui uma vasta comunidade na internet com fóruns de discussões de implementação e diversas bibliotecas prontas que podem ser utilizadas como base para diversas aplicações, facilitando o desenvolvimento de novos projetos. Além dessa característica o computador possui diversas entradas que podem ser utilizadas para a conexão de hardwares periféricos, facilitando ainda mais o seu funcionamento.

2.3 O conceito de aceleração, velocidade e deslocamento

O movimento de um corpo ou de uma partícula pode ser descrito de duas maneiras, a primeira é através do seu vetor posição (vetor que une o seu ponto inicial até o seu ponto atual – no instante da aferição do deslocamento) e a segunda é através das equações cartesianas obtidas pela trajetória (lugar geométrico das posições do corpo ao realizar a sua movimentação) [12]. Desta forma, essas duas maneiras podem ser assim representadas:

- Vetor posição: $\vec{r}(t) = x(t).\vec{i} + y(t).\vec{j} + z(t).\vec{k}$, onde \vec{i}, \vec{j} e \vec{k} são os versores unitários do eixo cartesiano;
- Equações cartesianas: $y = y(t)$, cada uma delas representa a posição da partícula de $z = (t)$ acordo com o eixo de deslocamento.

É possível perceber que tanto o vetor posição quanto as equações cartesianas estão diretamente relacionados e podem ser obtidas de forma complementar.

Definindo agora o vetor velocidade, e assumindo as mesmas duas maneiras de descrição do movimento, tem-se que a velocidade pode ser descrita por uma expressão vetorial ou por uma expressão escalar.

Na expressão vetorial, considerando o movimento de uma partícula ou de um corpo definido pelo vetor de posição $\vec{r} = \vec{r}(t)$, tem-se que no instante t a partícula/corpo está em uma posição P_1 e no instante $t + \Delta t$ a partícula/corpo está na posição P_2 . Desta forma, a velocidade vetorial instantânea no instante t é definida por [12]:

$$\vec{V}_P = \vec{V}(t) = \lim_{\Delta t \rightarrow 0} \frac{\Delta \vec{r}}{\Delta t} = \frac{d\vec{r}(t)}{dt} \quad (\text{Equação 1})$$

No limite a direção do vetor velocidade instantânea é a do sentido do movimento. Expandindo agora para cada um dos eixos, temos a velocidade instantânea definida por [12]:

$$\bullet \quad V_x(t) = \frac{dx(t)}{dt} = x'(t) = \dot{x}(t) \quad (\text{Equação 2})$$

$$\bullet \quad V_y(t) = \frac{dy(t)}{dt} = y'(t) = \dot{y}(t) \quad (\text{Equação 3})$$

$$\bullet \quad V_z(t) = \frac{dz(t)}{dt} = z'(t) = \dot{z}(t) \quad (\text{Equação 4})$$

Tomando atenção agora para expressão escalar, tem-se o movimento de uma partícula/corpo definido pela sua trajetória. Supondo que no instante t a partícula esteja no ponto P_1 , tendo percorrido desde t_0 uma distância s . Quando estivermos no instante $t + \Delta t$ a partícula irá se encontrar no ponto P_2 , tendo percorrido uma distância de $s + \Delta s$. Desta forma, assumindo o mesmo princípio assumido para a expressão vetorial, temos que a velocidade pode ser definida como [12]:

- $v = \frac{ds}{dt}$ (Equação 5)

ou, que o deslocamento é definido como:

- $s = s(t) = \int_t^{t+\Delta t} v(t). dt$ (Equação 6)

Novamente pode-se relacionar as duas formas de descrição da velocidade como sendo formas semelhantes e que estão relacionadas de forma complementares.

A aceleração de um corpo é definida como o aumento da velocidade por unidade de tempo. Assim como na velocidade e no deslocamento, a aceleração também pode ser dividida como aceleração vetorial e como aceleração escalar.

Na expressão vetorial, considerando a velocidade de uma partícula ou de um corpo definido pelo vetor de velocidade $\vec{v}(t)$ em um instante t , e no instante $t + \Delta t$ a partícula/corpo está com uma velocidade $\vec{v}(t + \Delta t)$. Desta forma, a aceleração vetorial instantânea no instante t é definida por [12]:

$$\vec{a}(t) = \lim_{\Delta t \rightarrow 0} \frac{\Delta \vec{v}(t)}{\Delta t} = \frac{d\vec{v}(t)}{dt} = \frac{d^2\vec{r}(t)}{dt^2} \quad (\text{Equação 7})$$

Portanto, foi observado que a aceleração instantânea é a primeira derivada no tempo da velocidade vetorial e a segunda derivada no tempo do vetor de posição. O mesmo vale-se para a integração, a velocidade é a integral no tempo da aceleração e o deslocamento é a integral no tempo da velocidade ou a integral dupla no tempo da aceleração. Expandindo agora para cada um dos eixos, tem-se a aceleração instantânea definida como:

- $a_x = \frac{dv_x(t)}{dt} = \frac{d^2x}{dt^2} = v_x'(t) = \ddot{x}(t)$ (Equação 8)

- $a_y = \frac{dy(t)}{dt} = \frac{d^2y}{dt^2} = v_y'(t) = \ddot{y}(t)$ (Equação 9)

- $a_z = \frac{dz(t)}{dt} = \frac{d^2z}{dt^2} = v_z'(t) = \ddot{z}(t)$ (Equação 10)

Tomando agora como base a expressão escalar, ou seja, a grandeza do vetor aceleração instantânea, tem-se [12]:

- $a = \frac{dv}{dt}$ (Equação 11)

ou, que o a velocidade e deslocamento são definidos como:

- $v = v(t) = \int_t^{t+\Delta t} a(t). dt$ (Equação 12)

- $s = s(t) = \int \int_t^{t+\Delta t} a(t). dt$ (Equação 13)

Desta forma, pode-se observar que todos os conceitos de aceleração, velocidade e deslocamento estão diretamente relacionados, sendo a dependência entre eles direta. Ou seja, tendo a informação da aceleração de uma partícula ou de um corpo, é possível obter as informações de sua velocidade instantânea e do seu deslocamento.

3. Materiais e Métodos

Serão apresentados individualmente os materiais e os métodos utilizados para o desenvolvimento do projeto. Na divisão a seguir esses itens serão todos integrados para ser apresentado o resultado final da consolidação da solução para o problema apresentado.

3.1 Materiais

Nesta seção do capítulo 3 serão apresentados todos os materiais utilizados para a implementação do projeto.

3.1.1 A raspberry Pi 2

Para o desenvolvimento do projeto foi utilizado especificamente a Raspberry Pi 2 model B V1.1, este modelo possui um processador de 900MHz e uma RAM de 1GB de memória, permitindo a conexão de sensores, displays e outros componentes eletrônicos utilizando um conector GPIO (General Purpose Input/Output) de 40 pinos que pode ser mapeado através da figura 3.

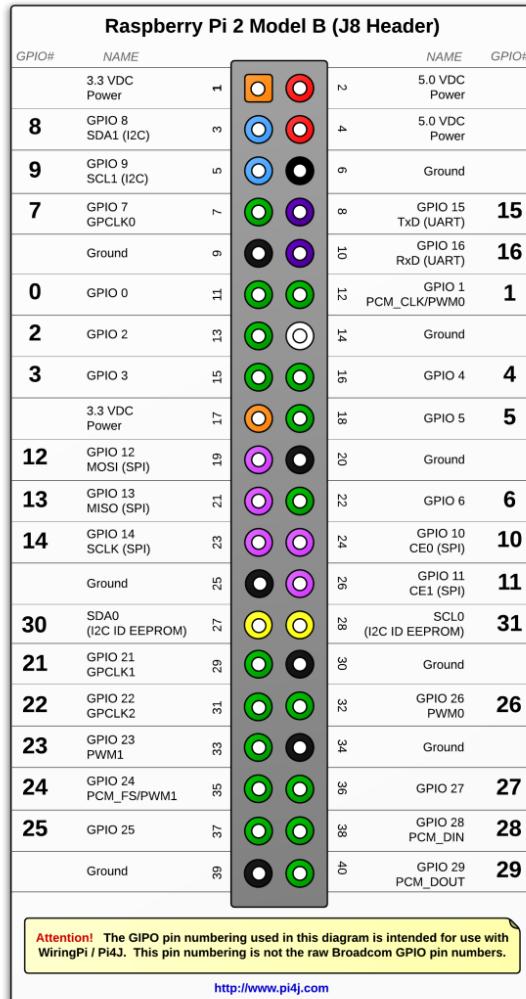


Figura 3: Mapeamento do conector GPIO – Raspberry Pi 2 Model B. [7]

A Raspberry Pi 2 model B é composta por um conector GPIO de 40 pinos, 4 portas USB que podem ser utilizadas para a instalação de teclado, mouse ou de qualquer outro dispositivo que utilize essa entrada, um conector Ethernet para a conexão do micro computador a internet, um conector câmera, uma entrada HDMI para a conexão com um monitor, uma saída áudio e vídeo (conector P2 – 4 polos), uma entrada slot para cartão miniSD, uma entrada micro-USB que é responsável pela alimentação da placa e por fim um conector DSI Display [8].



Figura 4: Raspberry Pi 2 Model B.

A entrada slot para cartão miniSD é parte fundamental para o funcionamento da Raspberry Pi 2 uma vez que o sistema operacional é instalado via cartão miniSD. O sistema operacional é o Raspbian, um sistema baseado em Linux e otimizado para ser utilizado no micro computador.

A Raspberry se tornou muito popular nos últimos tempos para projetos que envolvem programação de componentes eletrônicos, desta forma, foram criados diversas comunidades na internet com bibliotecas prontas para serem utilizadas em vastas aplicações, facilitando o seu uso e a criação de novos projetos com as mais diversas finalidades. Porém, muitas vezes a utilização destas bibliotecas prontas podem reduzir o desempenho do projeto como um todo, reduzindo a velocidade de processamento do programa a ser implementado ou aprimorado. Desta forma, devem existir diversos indicadores para constatar a eficiência do projeto desenvolvido e assim, otimizar a sua aplicação.

3.1.2 A programação Python

As linguagens de programação são responsáveis por traduzir a linguagem humana para a linguagem binária, ou seja, a linguagem utilizada por computadores. Desta forma, ao longo do tempo foram sendo desenvolvidas uma série de linguagens para fazer essa tradução, sendo elas classificadas em diversos níveis a medida em que se aproximam a linguagem humana [9].

Neste contexto, uma linguagem de baixo nível representa uma maior proximidade em relação à linguagem binária (utilizada pelos computadores), desta forma, é considerada uma linguagem mais difícil de ser programada. Uma linguagem de alto nível representa uma maior proximidade em relação à linguagem humana, elas possuem a finalidade de tornar a programação mais fácil de ser realizadas pelos seres humanos uma vez que a lógica utilizada está estruturada de uma forma mais similar à estrutura do pensamento humano [9].

A linguagem de programação utilizada para a implementação deste projeto foi a linguagem Python. Esta é uma linguagem de programação de alto nível, muito intuitiva e que procura ser fácil e bastante flexível para ser empregada a diversas situações. É uma linguagem interpretada, ou seja, ao se escrever um código, ele é interpretado por um outro programa para ser traduzido a linguagem binária. A linguagem não é traduzida diretamente para a linguagem binária a partir do código fonte [9].

O interpretador necessário para o uso da linguagem Python pode ser obtido facilmente. Nos sistemas operacionais Linux e OS X da Apple ele já vem instalado de uma forma padrão, não sendo necessária a sua instalação. No caso do popular sistema Windows, o interpretador pode ser instalado facilmente no próprio site da linguagem Python. A Raspberry Pi utiliza o sistema operacional Linux, desta forma, o interpretador já vem instalado em sua forma padrão.

A programação em linguagem Python consiste basicamente nas seguintes operações e comandos: Variáveis, Strings, Operações Matemáticas, Entrada de Dados, Estruturas de Controle, Funções, Listas e Expressões Booleanas. Esses comandos e operações são combinados de uma forma estruturalmente lógica a fim de passar comandos que são interpretados e repassados para as ações executadas pelo computador.

3.1.3 A comunicação I2C

O protocolo de comunicação utilizado para o desenvolvimento do projeto foi o protocolo I2C (*Inter-Integrated-Circuit*). O protocolo I2C descreve o funcionamento de um barramento de comunicação serial que tem como base utilizar apenas 2 fios, o SDA (Serial Data) e o SCL (Serial Clock) [13]. Este protocolo foi inventado pela Philips na década de 90 com o intuito de fazer a conexão de periféricos de baixa velocidade a placas mãe, microcomputadores e microcontroladores, e é o protocolo de comunicação serial mais utilizado no mercado, sendo ideal para projetos de microcontroladores de baixo custo, uma vez que possibilita a construção de sistemas mais complexos por meio de diversos sistemas mais simples [14].

Assim como visto, o protocolo I2C utiliza dois barramentos. O SDA é responsável pelo envio e recepção de dados, desta forma, por conta do SDA ser constituído apenas por um fio, não é possível realizar simultaneamente a recepção e o envio de dados, apenas uma das informações é enviada num determinado instante T [14]. Já o SCL é responsável por escrever os dados no barramento, a cada pulso do SCL o valor contido em SDA é lido como um bit, começando a sequência de leitura [13].

O protocolo I2C possui dois tipos de dispositivos, o *Master* e o *Slave*. O *Master* é responsável por coordenar todos os dispositivos periféricos ou, os chamados *Slaves*. A configuração desta estrutura pode ser vista na figura 5 abaixo.

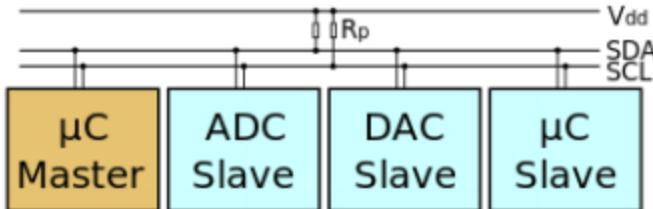


Figura 5: Configuração da estrutura do protocolo de comunicação I2C [13].

O modo de transmissão funciona através da seguinte maneira: quando em espera a comunicação fica em um estado neutro, tanto o SDA quanto o SCL estão em nível alto, ou seja valor digital 1. Quando há uma alteração de nível no SDA e este vai para um nível digital de sinal baixo, ou seja, valor digital 0, é iniciada a comunicação [13], como é possível de ser visto na

figura 6. Assim que iniciada a comunicação com a alteração de nível do SDA, o mestre escreve o endereço do registrador que ele deseja se comunicar, habilitando a comunicação, desta forma, o SCL começa a pulsar, sendo a cada pulso o valor SDA lido como um bit [14].

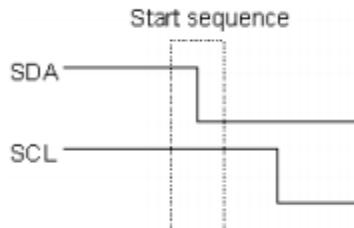


Figura 6: Início de comunicação do protocolo I2C [13].

Após estabelecida e realizada a comunicação, a sequência é interrompida no momento em que o SDA quanto o SCL estão em nível alto, ou seja valor digital 1.

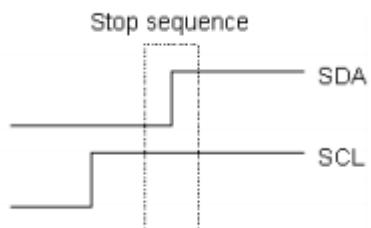


Figura 7: Final da comunicação do protocolo I2C [13].

Um ultimo ponto a ser destacado é o limite de desempenho do protocolo de comunicação I2C, de acordo com o sua versão mais atualizada, a versão 4.0, o barramento pode chegar a uma frequência de transmissão de dados de 5MHz [13], porém frequências arbitrárias menores do que esta podem ser escolhidas pelo desenvolvedor do projeto.

3.1.4 O ADXL 345

O acelerômetro utilizado para a implementação deste projeto foi o ADXL 345. Este componente é ideal para projetos que envolvam dispositivos móveis, onde é necessária a

aferição da gravidade resultante da movimentação do dispositivo. A sua aferição é realizada nos 3 eixos de movimentação (x,y e z) com uma alta resolução (13 bits) ou $\pm 16g$.

O ADXL 345 é um acelerômetro de 3 eixos de baixa potência, fino, pequeno e extremamente resistente, com saída digital e pode ser acessado por meio do protocolo de comunicação I2C [15]. O componente conta com uma série de funções, entre elas a detecção de inatividade de funcionamento (proporciona a economia de energia), a detecção do alcance de acelerações previamente estabelecidas, sensor de batidas que detecta toques simples ou duplos e a detecção de queda livre, que indica se o dispositivo sofreu alguma queda. O componente possui a configuração expressa na figura 8:



Figura 8: Configuração dos pinos do ADXL345 [16].

O componente é composto pelos seguintes pinos:

Pinos	Descrição
1	Alimentação 1 – 5.0V
2	Alimentação 2 – 3.3V
3	GND (<i>Ground</i>)
4	VS (<i>Supply Voltage</i>)
5	CS (<i>Chip Select</i>)

6	SCL (<i>Serial CLock</i>)
7	SDA (<i>Serial DAta</i>)
8	SDO (<i>Serial Data Output</i>)
9	IN1 (<i>Interrupt 1</i>)
10	IN2 (<i>Interrupt 2</i>)

Tabela 1: Relação dos pinos do ADXL345

Desta forma, os pinos 1 e 2 correspondem a alimentação do componente, o ADXL345 opera em duas tensões diferentes, 5.0V ou 3.3V, depende da tensão de alimentação que o usuário irá utilizar. Neste projeto foi utilizada a tensão de 3.3V para alimentar o componente. O pino 3 corresponde ao terra do componente, este pino é de extrema importância para o funcionamento pois representa uma referência de tensão. Como dito no parágrafo anterior, o componente pode ser acessado através da interface I2C ou também pode ser acessado pela interface SPI, em ambos os casos o componente atua como escravo. Assim, quando acessado pela interface SPI temos o pino 5 (CS), o pino 8 (SDO) e o pino 6 (SCL) para ser estabelecida a comunicação em 3 fios. Neste projeto será utilizada a interface I2C, desta forma, o pino 5 (CS) deve ser mantido sempre em nível alto e os pinos 6 (SCL) e 7 (SDA) realizam a comunicação com o microcontrolador (Raspberry Pi 2 Model B). O pino SDO permite a seleção de um endereço I2C alternativo e também é utilizado no projeto. Os pinos 9 e 10 são pinos de interrupção, eles não serão utilizados neste projeto. O diagrama de conexão para o componente funcionando através da interface I2C pode ser visto na figura 9.

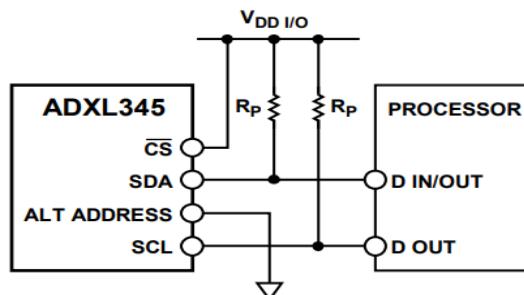


Figura 9: Diagrama de conexão para funcionamento da interface I2C [15]

Por fim, tomado as atenções para o funcionamento teórico do componente, podemos destacar que o ADXL 345 é uma estrutura de polissilicio que é construída sobre uma placa de silício. Sobre a placa de silício existem micro molas de polissilicio que suspendem a estrutura sobre a superfície e proporciona uma resistência contra as forças de aceleração resultantes sobre a placa. Desta maneira, a deflexão da estrutura é medida utilizando capacitores diferenciais que correspondem a placas fixas independentes que são fixadas à massa móvel. A aceleração resultante desvia o feixe e desequilibra o capacitor diferencial, resultando em uma saída do sensor cuja a amplitude é proporcional à aceleração. A demodulação sensível à fase é utilizada para determinar a magnitude e a polaridade da aceleração [15]. O diagrama completo do sistema pode ser visto detalhadamente na figura 10.

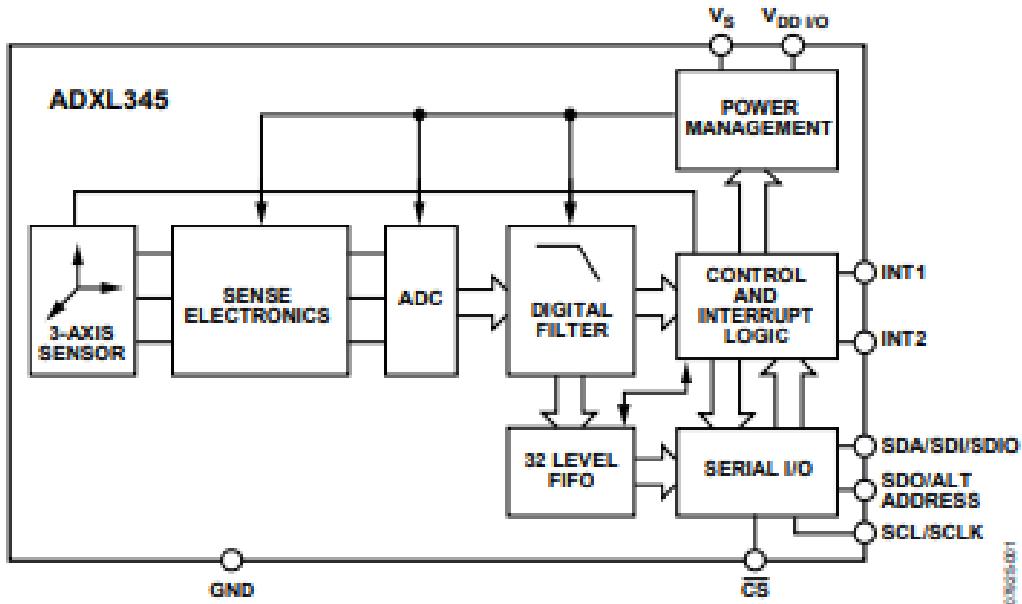


Figura 10: Diagrama de blocos funcionais do ADXL 345 [15]

3.2 Métodos

Nesta seção do capítulo 3 serão apresentados todos os métodos utilizados para a implementação do projeto, sendo a montagem física e o código implementado em linguagem Python para o correto funcionamento do acelerômetro ADXL345 para as suas funcionalidades propostas. Desta forma, no item montagem física serão expostos as conexões necessárias para o correto funcionamento do componente. Em seguida será exposto o código utilizado para

conectar o componente à raspberry Pi 2 e quais foram as lógicas utilizadas para a obtenção da velocidade e deslocamento do sensor.

3.2.1 Montagem física

A conexão para o funcionamento do acelerômetro ADXL 345 com a Raspberry Pi 2 é simples, assim como mencionado anteriormente, será utilizada a interface I2C, desta forma, a conexão do componente com a Raspberry deve seguir os princípios de funcionamento para esta interface.

Primeiramente, deve-se destacar os pinos utilizados pelo acelerômetro ADXL345 que fazem correspondência com a pinagem da Raspberry Pi 2 Model 2 V1.1. Assim como visto na seção 3.1.3, para a conexão do componente com a configuração da interface I2C deve-se utilizar os pinos 6 (SCL) e 7 (SDA) para fazer a transferência de dados, os pinos 2 (VCC – 3.3V) e 3 (GND) para sua correta alimentação e referência, o pino 5 (*Chip Select*) e o pino 8 (*Serial Data Output*). A relação da pinagem do componente pode ser visto na tabela 1 e relacionado fisicamente pela figura 8. Desta forma, foi realizada a conexão do acelerômetro com a Raspberry Pi, e como pode observado na figura 3, tem-se a relação da pinagem do microcontrolador, os pinos utilizados portanto são:

Pinos	Descrição
1	3.3 V (<i>Power</i>)
3	GPIO 8 (SDA1) – <i>Serial Data 1</i>
5	GPIO 9 (SCL1) – <i>Serial Clock 1</i>
6	<i>Ground 1</i>
9	<i>Ground 2</i>
17	3.3 V (<i>Power 2</i>)

Tabela 2: Relação de pinos utilizados na Raspberry Pi Model B V1.1 para a conexão com o ADXL 345

Desta forma, a conexão foi realizada da seguinte forma:

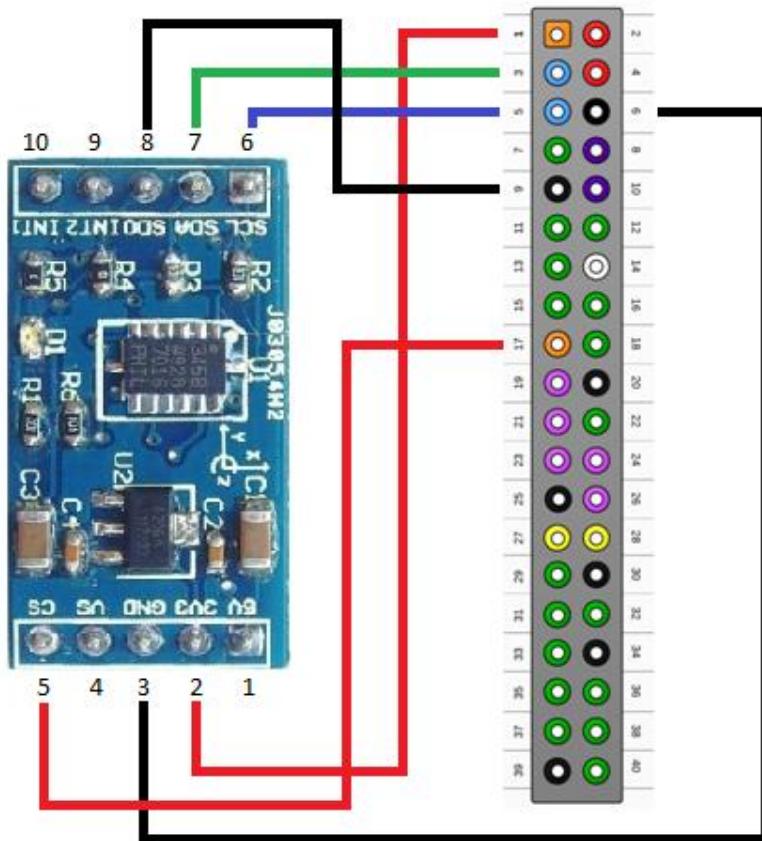


Figura 11: Diagrama com a conexão entre o ADXL 345 e Raspberry Pi 2

Na figura 10, os fios em vermelho simbolizam os pinos com VCC (3.3V), os fios e preto simbolizam os pinos em Terra (*Ground*), o fio em verde simboliza o SDA (*Serial DAta*) e o fio em azul simboliza o SCL (*Serial CLock*). Como pode ser observado, o pino 5 CS (*Chip Select*) se encontra conectado ao pino 17 (VCC - 3.3V), esta conexão é explicada pois para a conexão com a interface I2C o CS deve permanecer sempre em alta tensão [15]. Outro ponto a ser destacado é o pino 8 SDO (*Serial Data Output*) que se encontra conectado ao pino 9 (*Ground*), esta conexão também é explicada pois para a conexão com a interface I2C o SDO ou *Alt Address* deve estar conectado ao terra do circuito [15].

3.2.2 Implementação do código do sistema em linguagem Python

Com a implementação física do circuito realizada, o próximo passo do projeto é realizar a implementação da comunicação serial do sistema (comunicação I2C), o código em Python para a coleta da aceleração do componente nos eixos X, Y e Z e a implementação do código para a obtenção da velocidade e o deslocamento no plano XY, estas implementações serão vistas nos tópicos a seguir.

3.2.2.1 Implementação da comunicação serial

Foi implementado a configuração na Raspberry Pi 2 Model B para a habilitação da conexão I2C. Para tal, foram realizados os seguintes procedimentos:

1. Foram adicionados os módulos I2C na configuração da Raspberry Pi, de acordo com os seguintes comandos: “sudo nano /etc/modules”, “i2c-bmc2708” e “i2c-dev”; Desta forma, os módulos I2C já estavam habilitados;
2. Em seguida foi necessário fazer uma reinicialização do microcomputador para a realização das mudanças requisitadas. Foi realizado um reboot na barra de comandos;
3. Para a instalação dos softwares necessários para o correto funcionamento da interface de conexão I2C foi realizado o seguinte comando: “sudo apt-get install python-smbus i2c-tools git-core”;
4. Para finalizar a habilitação e o correto funcionamento do acelerômetro interfaceado pelo protocolo I2C, foi realizado o teste de verificação do endereço de entrada do componente no protocolo de comunicação I2C. O ADXL345 possui como padrão a sua comunicação através do endereço 53 do protocolo I2C, desta forma, ao realizar o seguinte comando: “sudo i2cdetect -y 1”, deve-se obter o endereço 53 na matriz de comunicação do protocolo. Com a realização deste comando foi obtido o seguinte resultado exposto na figura 12:

```

File Edit Tabs Help
pi@raspberrypi:~ $ sudo i2cdetect -y 1
      0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00: --
10: --
20: --
30: --
40: --
50: --  53 --
60: --
70: --
pi@raspberrypi:~ $

```

Figura 12: Entrada no protocolo de comunicação I2C da Raspberry Pi Model B

Portanto, pode-se verificar que a comunicação serial entre a Raspberry Pi 2 e o ADXL345 foi realizada com sucesso.

3.2.2.2 Implementação do acelerômetro ADXL345

Após a implementação da comunicação serial entre a Raspeberry Pi 2 e o acelerômetro ADXL345 foi desenvolvido um código em linguagem Python para a obtenção dos valores de aceleração nos eixos x, y e z do acelerômetro.

Neste código foi implementado o funcionamento do sistema com uma largura de banda de 50Hz e uma escala de resolução de medida máxima de até 2g. Foi definido este limite para obter uma elevada precisão da medida realizada, uma vez que com o limite de resolução máximo de 16g a precisão da aferição da aceleração realizada é comprometida. Outro fator determinante para a escolha da resolução máxima de 2g é a característica de uso que o acelerômetro será empregada, o componente será utilizado para realizar aferições da aceleração do deslocamento de um carrinho de monitoração de ambiente, desta forma, esta aceleração não ultrapassará 2g.

```

def __init__(self, address = 0x53):
    self.address = address
    self.setBandwidthRate(BW_RATE_50HZ)
    self.enableMeasurement()

```

Em seguida foi realizada a implementação da permissão de conexão entre a Raspberry e o ADXL com as condições pré-estabelecidas, sendo expressa pelo código abaixo:

```
def enableMeasurement(self):
    bus.write_byte_data(self.address, POWER_CTL, MEASURE)

def setBandwidthRate(self, rate_flag):
    bus.write_byte_data(self.address, BW_RATE, rate_flag)
```

Após a habilitação da conexão de acordo com os corretos endereços para o funcionamento do componente, foi estabelecida a relação de alocação dos bytes para cada um dos eixos. Cada um dos eixos do componente é composto de 16 bits, sendo um para a leitura do sinal e os outros 15 bits para a aferição de sua magnitude, correspondendo assim um total de 65536 valores possíveis (2^{15}).

```
def getAxes(self, gforce = True):
    bytes = bus.read_i2c_block_data(self.address, AXES_DATA, 6)

    x=bytes[0] | (bytes[1] << 8)
    if(x & ( 1 << 16-1)):
        x = x - (1<<16)

    y=bytes[2] | (bytes[3] << 8)
    if(y & ( 1 << 16-1)):
        y = y - (1<<16)

    z=bytes[4] | (bytes[5] << 8)
    if(z & ( 1 << 16-1)):
        z = z - (1<<16)
```

Vale destacar, que para as condições de resolução máxima estabelecidas, é necessário o uso de uma taxa de conversão de 3.9mG/LSB, como pode ser visto na tabela 3:

Range	mG/LSB
All g-range when FULL_RES bit = 1	3.9
±2g, 10 bit mode	3.9
±4g, 10 bit mode	7.8
±8g, 10 bit mode	15.6
±16g, 10 bit mode	31.2

Tabela 3: Relação entre a resolução e a taxa de conversão para o componente ADXL345 [15]

Desta forma, a taxa de conversão foi multiplicada em cada um dos eixos utilizados, sendo representado no trecho abaixo:

```
x = x*SCALE_MULTIPLIER  
y = y*SCALE_MULTIPLIER  
z = z*SCALE_MULTIPLIER
```

Por fim, a escala dos valores obtidos é expressa em relação à gravidade (G), como o intuito do projeto é a obtenção da velocidade e do deslocamento que envolve o componente ADXL345, o resultado obtido em cada eixo foi multiplicado pela constante de gravidade, *EARTH_GRAVITY_MS2*, definida como 9,80665m/s², sendo a aceleração final expressa em m/s².

```
x = x * EARTH_GRAVITY_MS2  
y = y * EARTH_GRAVITY_MS2  
z = z * EARTH_GRAVITY_MS2
```

Para finalizar, foi retornado na função criada o valor de cada um dos eixos de medição (eixos x, y e z). Diante do exposto, pode-se perceber que a implementação do código é simples, deve ser seguido os parâmetros e endereços pré estabelecidos no *Datasheet* do componente. Segue o fluxograma da implementação do código:

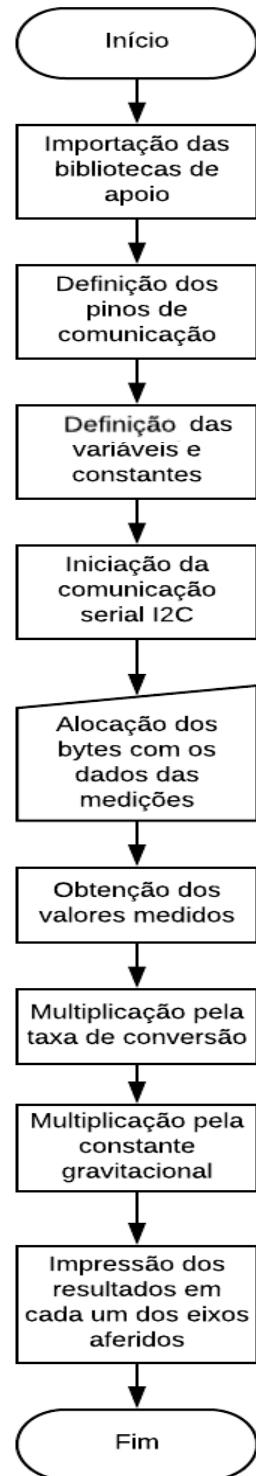


Figura 13: Fluxograma de leitura e atuação do acelerômetro ADXL345

3.2.2.3 Implementação para obtenção da velocidade e deslocamento

Após a implementação do acelerômetro ADXL345, deve-se utilizar a aceleração obtida para calcular a velocidade e o deslocamento do componente que será inserido no carrinho. Neste ponto, duas considerações devem ser feitas a fim da obtenção do correto funcionamento do código.

A primeira delas é referente à componente gravitacional que incide sobre o acelerômetro. O acelerômetro irá sempre apresentar uma componente gravitacional que pode estar decomposta entre os seus eixos, desta forma, esta componente deve ser eliminada para não influenciar na medida de sua velocidade e de seu deslocamento. Para realizar essa eliminação da componente gravitacional, inicialmente foi realizada uma medição dos 10 primeiros pontos iniciais da aceleração, esses pontos foram somados em cada um dos seus respectivos eixos e foi extraída uma média para cada um dos eixos. Essa lógica é expressa no código abaixo:

```

for i in range(10):
    adxl345 = ADXL345()
    axes = adxl345.getAxes(True)
    Ax = axes['x']
    Ay = axes['y']
    Az = axes['z']
    Axm = Axm + Ax
    Aym = Aym + Ay
    Azm = Azm + Az

Axm = Axm/10
Aym = Aym/10
Azm = Azm/10

```

Desta forma, os valores Axm , Aym e Azm são as componentes iniciais da aceleração do sistema, ainda estático, logo, correspondem a influência inicial da gravidade no sistema e devem ser subtraídas dos valores de aceleração que serão utilizados para a medição da velocidade e do deslocamento do ADXL345.

Um segundo ponto que deve ser considerado para a composição da interferência da gravidade no sistema é a componente gravitacional de acordo com a movimentação realizada em relação a posição inicial. Após a saída do ponto inicial, uma nova componente gravitacional deve ser considerada. Desta forma, foi adotado um referencial padrão, com o eixo z sempre perpendicular a componente gravitacional, a movimentação do carrinho ocorre sempre no plano XY, deve-se portanto calcular o ângulo de movimentação realizado e fazer a projeção da

componente gravitacional que está inserida nos eixos x e y para o eixo z. Os ângulos desta projeção podem ser vistos na figura 14:

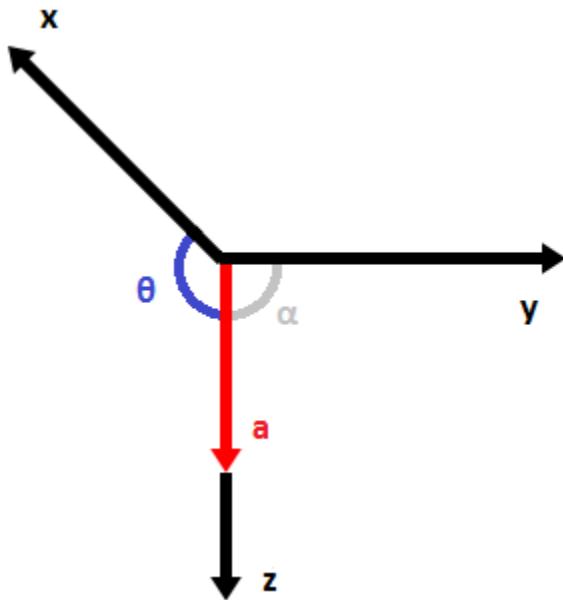


Figura 14: Relação de ângulos com os vetores x, y e z

Para $A_x = x$, $A_y = y$ e $A_z = z$, tem-se a seguinte relação para os ângulos expostos:

$$\theta = \arcsin(A_y) \quad (\text{Equação 14})$$

$$\alpha = \arctan\left(\frac{A_x}{\sqrt{A_y^2 + A_z^2}}\right) \quad (\text{Equação 15})$$

A medida do ângulo para a correção deve ser realizada entre a diferença do ângulo inicial e ângulo atual, ela é então expressa no código abaixo:

```
x = axes['x']
y = axes['y']
z = axes['z']

c=((x/EARTH_GRAVITY_MS2)/(math.sqrt((y/EARTH_GRAVITY_MS2)**2+(z/EARTH_GRAVITY_MS2)**2)))
cm=((Axm/EARTH_GRAVITY_MS2)/(math.sqrt((Aym/EARTH_GRAVITY_MS2)**2+(Azm/EARTH_GRAVITY_MS2)**2)))
teta=(math.asin(y/EARTH_GRAVITY_MS2)-math.asin(Aym/EARTH_GRAVITY_MS2))*57.2858
alpha=(math.atan(c)-math.atan(cm))*57.2858
```

Onde x é a medida da aceleração no eixo x atual, y é a medida da aceleração no eixo y atual, z é a aceleração no eixo z atual e Axm, Aym e Azm são a média da aceleração nos eixos x, y e z inicial para os 10 primeiros pontos medidos.

A componente de aceleração utilizada como base foi a componente z, assim, deve ser realizada a projeção das componentes x e y para o eixo z. De acordo com a figura 12, pode-se determinar que a componente dos eixos será expressa por:

$$\text{Componente de } A_x: A_{xz} = A_x \cdot \cos \theta$$

$$\text{Componente de } A_y: A_{yz} = A_y \cdot \cos \alpha$$

Desta forma, para zerar as componentes relativas a força da gravidade nos eixos x e y deve-se subtrair a componente inicial e a componente A_x e A_y atual, sendo expressa por:

$$Ax = x - Axm - x * \text{math.cos}(teta) \quad \text{e} \quad Ay = y - Aym - y * \text{math.cos}(alpha) \quad \text{onde}$$

A_x e A_y são as acelerações nos eixos x e z sem a influência da gravidade, x e y são as acelerações atuais nos eixos x e y, teta é o ângulo de diferença entre a componente x e z de acordo com o vetor atual e o vetor inicial e o alpha é o ângulo de diferença entre a componente y e z de acordo com o vetor atual e o vetor inicial.

O segundo ponto a ser considerado para a implementação é a utilização da integral na aceleração para a obtenção da velocidade e do deslocamento. Existem diversas formas de implementação desta integral, a forma utilizada neste projeto foi a integração trapezoidal, a fim de evitar o acúmulo de erros por meio da constante de integração. A integração trapezoidal é baseada na aproximação da integral da função utilizada a área do gráfico que ela descreve [17], como pode ser visto na figura 15:

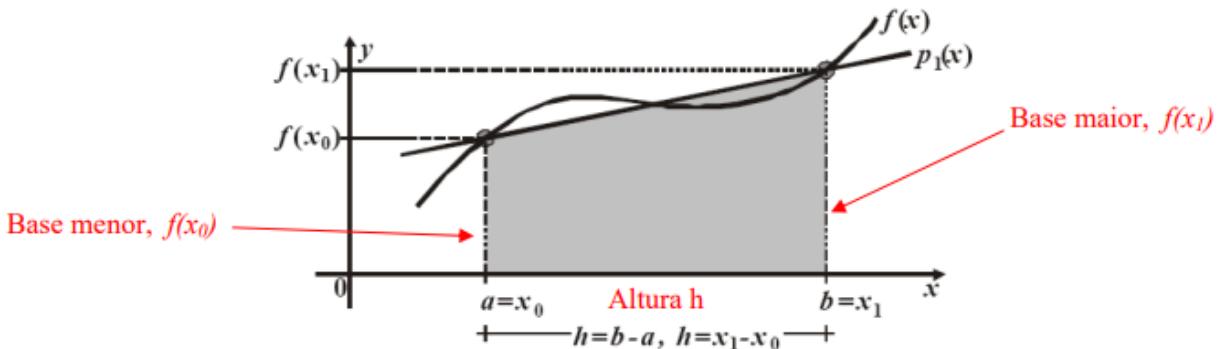


Figura 15: Demonstração da integração trapezoidal [17]

A medida que a altura h (intervalo ab) vai sendo dividida em tamanhos menores a taxa de erro vai sendo diminuída, como pode ser visto na figura 16:

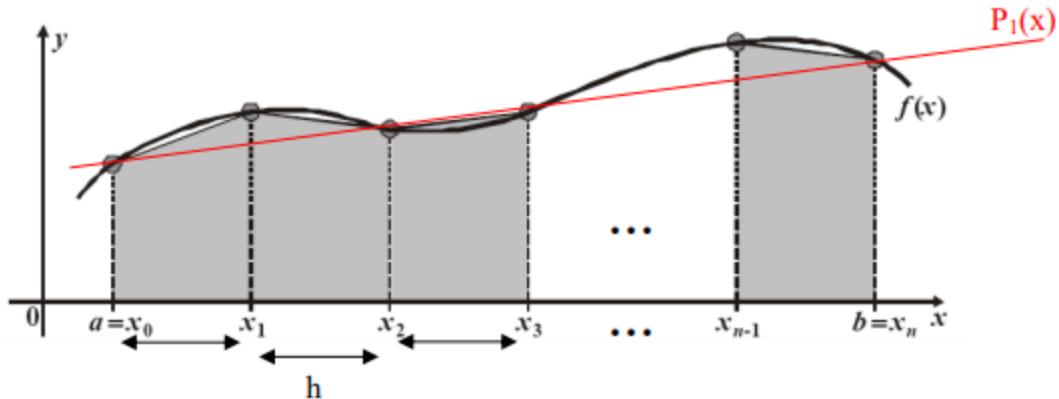


Figura 16: Demonstração da integração trapezoidal com a divisão da altura em intervalos menores [17]

Desta forma, quanto mais divisões são realizadas na altura h , menor é o erro na integração final. O ideal por tanto é realizar diversas divisões e fazer a soma das áreas calculadas a fim de obter o menor erro possível. Para o projeto implementado foi utilizado uma taxa de amostragem de 0,02 segundos, esta taxa pode ser considerada uma taxa de amostragem confiável para as pretensões do projeto, uma vez que o acelerômetro não permite taxas menores por conta de uma limitação na sua frequência de leitura das acelerações ($f=50$ Hz). Como visto, os acelerômetros possuem uma elevada precisão, porém são caracterizados por apresentarem um ruído bastante expressivo, desta forma, são expressos os resultados das aferições realizadas a cada 50 pontos captados, sendo feita uma média entre eles a fim de minimizar a influência deste ruído presente no sistema, os resultados são expressos a cada 1 segundo ($50 \cdot 0,02$). Portanto, o código implementado pode ser visto nos trechos abaixo:

```

if x > 0:
    Ax = x - Axm - x*math.cos(teta)
    Vx = Aix * T + abs((Ax - Aix) / 2) * T
    Dx = Vix * T + abs((Vx - Vix) / 2) * T
else:
    Ax = x - Axm - x*math.cos(teta)
    Vx = Aix * T - abs((Ax - Aix) / 2) * T
    Dx = Vix * T - abs((Vx - Vix) / 2) * T

if y > 0:
    Ay = y - Aym - y*math.cos(alpha)
    Vy = Ai y * T + abs((Ay - Ai y) / 2) * T
    Dy = Vi y * T + abs((Vy - Vi y) / 2) * T
else:
    Ay = y - Aym - y*math.cos(alpha)
    Vy = Ai y * T - abs((Ay - Ai y) / 2) * T
    Dy = Vi y * T - abs((Vy - Vi y) / 2) * T

if z > 0:
    Az = z - Azm
    Vz = Aiz * T + abs((Az - Aiz) / 2) * T
    Dz = Viz * T + abs((Vz - Viz) / 2) * T
else:
    Az = z - Azm
    Vz = Aiz * T - abs((Az - Aiz) / 2) * T
    Dz = Viz * T - abs((Vz - Viz) / 2) * T

Axm1 = Axm1 + Ax F
Aym1 = Aym1 + Ay F
Azm1 = Azm1 + Az F
Vxm1 = Vxm1 + Vx F
Vym1 = Vym1 + Vy F
Vzm1 = Vzm1 + Vz F
Dxm1 = Dxm1 + Dx F
Dym1 = Dym1 + Dy F
Dzm1 = Dzm1 + Dz F

```

Vale destacar a divisão realizada para a integração das acelerações positivas e das acelerações negativas. Essa divisão se deve, pois realizamos a soma do resultado da velocidade e do deslocamento instantâneo em valores absolutos, assim, quando o valor inicial é menor do que 0 a soma seria expressa como uma variação negativa desta velocidade, o que provocaria um erro sobre o resultado.

Por fim, segue um algoritmo simplificado do código implementado:

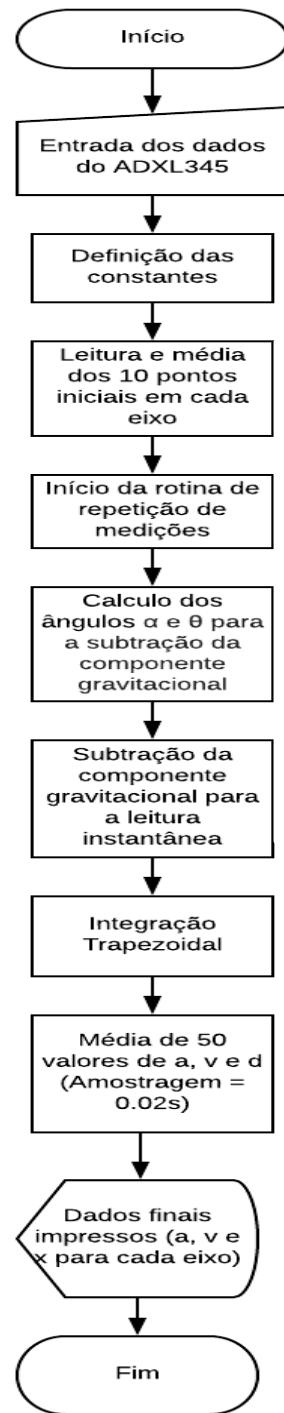


Figura 17: Fluxograma do algoritmo implementado para a aquisição da velocidade e deslocamento do componente

4. Resultados e Discussões

Neste capítulo serão apresentados e discutidos os resultados do desenvolvimento do projeto, sendo primeiramente expostos os resultados da implementação do algoritmo da leitura e funcionamento do acelerômetro ADXL345 e em seguida serão apresentados os resultados referentes à implementação do algoritmo para a obtenção da velocidade e do deslocamento por meio da aceleração expressa pelo acelerômetro ADXL345.

4.1 Implementação do algoritmo da leitura e funcionamento do acelerômetro ADXL345

Após a implementação do algoritmo para a leitura e o funcionamento do acelerômetro ADXL345 foi realizado o seu teste de funcionamento. Este teste visa o posicionamento do acelerômetro em posições pré-estabelecidas para verificar se as medidas apresentadas conferem a realidade. Para tal, foram realizados os testes de acordo com as figuras 18, 20 e 22:

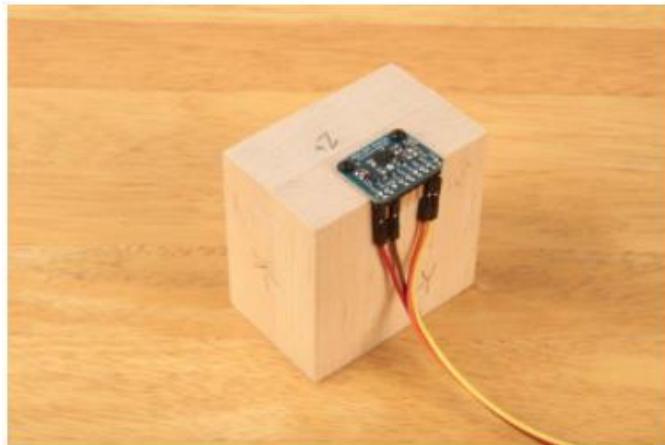


Figura 18: Calibração do Acelerômetro ADXL 345 no eixo Z [19]

```
ADXL345 on address 0x53:
  x = 0.000m/s^2
  y = 0.353m/s^2
  z = 9.689m/s^2
```

Figura 19: Resultado da calibração do componente no eixo Z

Como o esperado, com o acelerômetro posicionado perpendicularmente ao eixo Z toda a componente da gravidade estará sendo refletida no eixo Z do componente, assim como pode-se ser visto na figura 19.

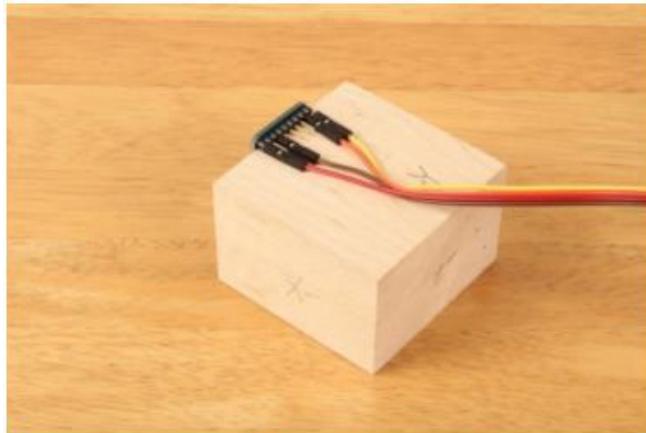


Figura 20: Calibração do Acelerômetro ADXL 345 no eixo X [19]

```
ADXL345 on address 0x53:  
x = 10.199m/s^2  
y = -0.039m/s^2  
z = 0.000m/s^2
```

Figura 21: Resultado da calibração do componente no eixo X

Assim como no eixo Z, com o acelerômetro posicionado perpendicularmente ao eixo X, toda a componente gravitacional foi refletida no eixo X do componente, como pode ser visto experimentalmente na figura 21.

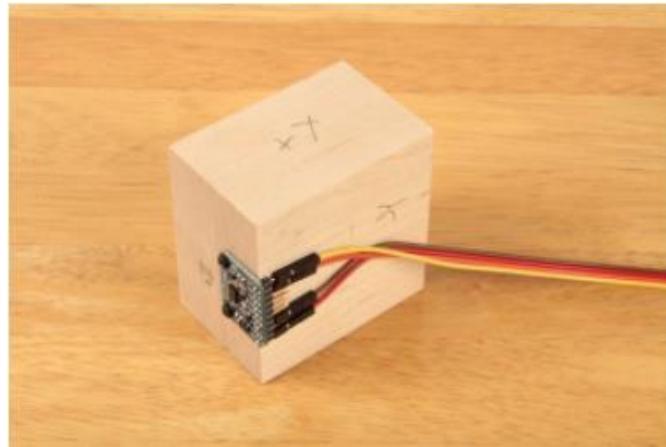


Figura 22: Calibração do Acelerômetro ADXL 345 no eixo Y [19]

```
ADXL345 on address 0x53:
x = 0.079m/s^2
y = 10.434m/s^2
z = 0.235m/s^2
```

Figura 23: Resultado da calibração do componente no eixo Y

Assim como nos eixos X e Z, com o acelerômetro posicionado perpendicularmente ao eixo Y, toda a componente gravitacional foi refletida no eixo Y do componente, como pode ser visto experimentalmente na figura 23.

Desta forma, pode-se analisar e concluir que a implementação do código para a conexão e funcionamento do ADXL345 com a Raspberry Pi 2 foi realizada corretamente. Todos os pontos de calibração foram testados e os resultados obtidos foram consistentes com as suas interpretações físicas reais, ou seja, a componente gravitacional se manteve estritamente no eixo analisado com resultados muito próximos a força gravitacional ($9,80665 \text{ m/s}^2$). Devido a componente gravitacional, não foi possível a realização de testes de deslocamentos precisos, estes testes serão realizados na próxima seção deste capítulo, onde a componente gravitacional foi retirada para evitar erros na medição da velocidade e deslocamento do sistema.

4.2 Implementação do algoritmo para aquisição da velocidade e deslocamento do sistema

Após a implementação do algoritmo para aquisição da velocidade e deslocamento do sistema foi realizado o seu teste de funcionamento, sendo realizados os testes para a eliminação da influência da gravidade (influência da gravidade na posição inicial e influência da gravidade na posição atual) e o teste integrativo, para a verificação da velocidade e deslocamento no plano XY.

Primeiramente, para o teste da eliminação da força da gravidade na posição inicial, o acelerômetro foi fixado perpendicularmente ao eixo Z, logo, no plano XY. Este é definido como o seu plano de aplicação, esta posição pode ser vista na figura 24.

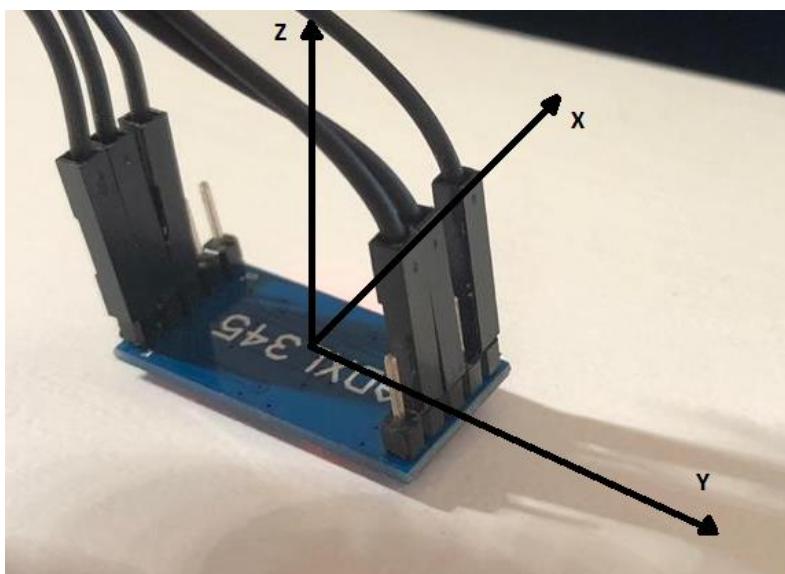


Figura 24: Posição inicial do acelerômetro.

Foi obtido os seguintes valores de aceleração A_{xm} , A_{ym} e A_{zm} iniciais:

```
Axm = 0.019 m/s^2
Aym = -0.020 m/s^2
Azm = -10.542 m/s^2
```

Figura 25: Decomposição da força da gravidade na posição inicial.

Esta decomposição de forças já era esperada, uma vez que o acelerômetro se mantém fixo com o eixo Z perpendicular a força da gravidade, toda a componente gravitacional estará no eixo Z, assim como encontrado.

Desta forma, para o acelerômetro fixo na posição inicial, tem-se os seguintes valores de aceleração, velocidade e deslocamento:

```
Aceleracao [m/s^2] :
Ax = 0.037 m/s^2
Ay = 0.039 m/s^2
Az = 0.024 m/s^2
Velocidade [m/s] :
Vx = 0.001 m/s
Vy = 0.001 m/s
Deslocamento [m] :
Dx = 0.000m
Dy = 0.000m
```

Figura 26: Valores de aceleração, velocidade e deslocamento instantâneos para a posição inicial.

Os valores de aceleração, velocidade e deslocamento para o sistema parado em sua posição inicial estão de acordo com o esperado, uma vez que o sistema encontra-se parado os valores encontrados nos eixos X e Y devem ser 0. Os pequenos valores apresentados devem-se ao ruído pertencente ao circuito, porém esses são muito pequenos, não contribuindo em uma medida expressiva para o comprometimento do resultado obtido.

Atentando-se agora aos ângulos θ e α referentes à diferença angular entre a posição inicial e a posição atual dos eixos X e Y respectivamente, para o sistema fixo em sua posição inicial temos os seguintes valores:

```
Teta = -0.229°
Apha = -0.107°
```

Figura 27: Posição angular dos eixos X e Y para o componente fixo em sua posição inicial

Assim como com a aceleração, velocidade e deslocamento, espera-se que os ângulos referentes à diferença angular entre a posição inicial e a posição atual em relação aos eixos X e Y também sejam 0. Isso acontece pois com o sistema fixo em sua posição inicial não houve movimentação do sistema, logo, os ângulos se mantiveram os mesmos em relação aos iniciais. O pequeno valor encontrado é referente ao ruído presente no sistema, sendo irrisório o seu comprometimento nos valores finais obtidos.

Estabelecendo agora a análise de resultados para a movimentação do sistema, serão realizados testes de deslocamento no eixo X e no Y para verificar a exatidão da medição realizada.

O primeiro teste realizado foi um deslocamento de 30 cm através do eixo X. Desta forma, foi fixado um “trilho” de movimentação de 30 cm sobre o eixo X, e feito o deslocamento do sistema ao longo deste eixo, tentando realizar um aumento constante da aceleração do sistema. Os resultados podem ser vistos na figura 28:

```
Aceleracao [m/s^2] :
Ax_F = 3.193 m/s^2
Ay_F = -0.394 m/s^2
Az_F = -0.085 m/s^2
Velocidade [m/s] :
Vx_F = 0.354 m/s
Vy_F = -0.008 m/s
Deslocamento [m] :
Dx_F = 0.180m
Dy_F = -0.000m
```

Figura 28: Deslocamento de 30 cm ao longo do eixo X

Um segundo teste realizado foi um deslocamento de -30 cm através do eixo X. Desta forma, o mesmo “trilho” foi fixado sobre o eixo X, e feito o deslocamento do sistema ao longo do sentido negativo deste eixo, tentando realizar um aumento constante da aceleração do sistema. Testando assim, a movimentação no sentido negativo do eixo. Os resultados podem ser vistos na figura 29:

```

Aceleracao [m/s^2]:
Ax_F = -3.543 m/s^2
Ay_F = 0.346 m/s^2
Az_F = -0.114 m/s^2
Velocidade [m/s]:
Vx_F = -0.372 m/s
Vy_F = 0.010 m/s
Deslocamento [m]:
Dx_F = -0.150m
Dy_F = 0.000m

```

Figura 29: Deslocamento de -30 cm ao longo do eixo X

Por fim, um terceiro teste foi realizado. Neste, o mesmo “trilho” foi posicionado no eixo Y, foi feito o deslocamento do sistema ao longo do sentido negativo deste eixo, tentando realizar um aumento constante da aceleração do sistema. Testando assim, a movimentação do sistema neste eixo. Os resultados podem ser vistos na figura 30:

```

Aceleracao [m/s^2]:
Ax_F = 0.277 m/s^2
Ay_F = -2.859 m/s^2
Az_F = 0.245 m/s^2
Velocidade [m/s]:
Vx_F = 0.010 m/s
Vy_F = -0.316 m/s
Deslocamento [m]:
Dx_F = 0.000 m
Dy_F = -0.120m

```

Figura 30: Deslocamento de -30 cm ao longo do eixo Y

Como pode ser visto, os deslocamentos aferidos nos eixo X e Y foram de 18cm, -15cm e -12cm. As diferenças encontradas devem-se a alguns fatores, entre eles estão o ruído presente no sistema, o erro de integração que passa a ser inevitável e a curta distância de deslocamento.

Este ultimo é um fator limitante de teste do sistema, não houve tempo hábil da instalação do sistema no carrinho pois este não estava finalizado.

Desta forma pode-se concluir que apesar do correto funcionamento do sistema, ele não apresenta elevada exatidão de medição de velocidade e deslocamento. Essa dificuldade de exatidão de medição da velocidade e deslocamento já era esperada, uma vez que os acelerômetros não são componentes indicados para a obtenção dessas medidas. Porém, deve-se destacar que o comportamento dinâmico do sistema para a exibição dos resultados de aceleração, velocidade e deslocamento a medida em que os eixos eram movimentados se apresentou muito confiável. A aceleração do sistema também apresentou resultados muito positivos, a eliminação da força da gravidade do sistema foi implementada com sucesso e a expressão da aceleração é resultante apenas da movimentação do sistema, de acordo com a ideia inicial a ser implementada.

5. Conclusão

Este trabalho se apresentou como um dos mais complexos desenvolvidos durante a graduação uma vez que foram utilizados componentes e uma linguagem de programação que ainda não havia realizado contato durante o curso. Através do problema proposto no Capítulo 1, Introdução, foi possível tirar uma carga de aprendizado muito alta, além de estar em contato com situações de soluções de problemas interdisciplinares.

Por meio dos componentes utilizados foi possível aprofundar e colocar em práticas diversos conhecimentos obtidos na disciplina de transdutores, foi possível também colocar em prática conceitos obtidos nas disciplinas de física 1, física 2, e álgebra linear, além dos conceitos de processamentos de sinais digitais obtidos na disciplina de PDS e conceitos obtidos na disciplina de Sinais e Sistemas. Como explicitado nos Capítulos 2 e 3, Embasamento Teórico e Materiais e Métodos, respectivamente, onde foi abordado todo o contexto teórico utilizado no projeto, os materiais utilizados para sua implementação e os respectivos métodos para a solução do problema proposto.

Conforme discutido no capítulo 4, a implementação da conexão do ADXL345 com a Raspberry Pi 2 foi realizada com sucesso, resultando no funcionamento do componente de acordo com o esperado. Este funcionamento pôde ser comprovado com a realização dos testes de calibração, onde o componente apresentou resultados exatamente como o esperado de acordo com o posicionamento para a obtenção da força gravitacional em cada um dos eixos X, Y e Z.

Com relação à implementação do algoritmo para a leitura da aceleração, velocidade e deslocamento do sistema, algumas considerações podem ser feitas. A primeira delas é a referente à eliminação da componente gravitacional no sistema de medição, a subtração da componente gravitacional inicial juntamente com a subtração da componente instantânea funcionou muito bem para o resultado final, não havendo influência da força gravitacional nas medições. Um segundo ponto a ser destacado foi à utilização da integral trapezoidal para obtenção da velocidade e deslocamento, este método se apresentou como o método mais preciso por não apresentar uma constante de integração indefinida, porém resultou em valores de deslocamentos imprecisos. Apesar do sistema não ser empregado em sua utilização final para a realização dos testes práticos, nos testes de medição realizados ele apresentou problemas quanto a sua exatidão. Um último ponto a ser destacado foi o resultado expresso

sempre como uma média entre 50 aferições, este método resultou em medidas de aceleração, mais confiáveis, reduzindo a influência do ruído presente no acelerômetro no sistema.

Por fim, alguns pontos que podem ser aperfeiçoados no projeto são a implementação do eixo Z na subtração da componente gravitacional instantânea, desta forma o carrinho pode se movimentar no eixo X, Y e Z e não somente no plano XY, permitindo a descida em rampas. Outro fator importante para o aperfeiçoamento é a instalação do sistema desenvolvido no carrinho, este ponto consolida o desenvolvimento do projeto e não conseguiu ser realizado pois o carrinho se encontrava ainda em desenvolvimento ao final do projeto.

Uma limitação encontrada no projeto é a possibilidade da expressão da velocidade e deslocamento somente de forma instantânea, essa característica se deve a dependência somente da aceleração para este cálculo, desta forma, para um deslocamento a velocidade constante, onde não há aceleração, este movimento não é detectado. Porém, uma variação na velocidade é detectada por conta da influência da aceleração para essa troca de velocidade.

Uma segunda limitação muito importante do sistema é a sua exatidão, assim, se o usuário pretende utilizar um sistema extremamente preciso não é aconselhável a utilização do acelerômetro para a aferição da velocidade e deslocamento do sistema.

6. Referências Bibliográficas

[1] Giménez, Carlos; Universidad Católica “Nuestra Señora de Asunción” - Teoría y Aplicaciones de la Informática 2

Disponível em <http://jeuazarro.com/wp-content/uploads/2014/10/Acelerometro.pdf>

Acesso em 17 de abril de 2018 às 21:03.

[2] Walter, Patrick L.; “The History of the Accelerometer” – Texas Christian University, Ft. Worth, Texas

Disponível em <http://qringtech.com/wpcontent/uploads/2014/01/HistoryOfTheAccelerometer.pdf>

Acesso em 18 de abril de 2018 às 19:48

[3] Stein, P. K., “The Early Strain Gage Accelerometers: The Inventors and Their Times,” The Shock and Vibration Bulletin – Part II, Shock and Vibration Information Analysis Center (SAVIAC), 67th Shock and Vibration Symposium, Monterrey, CA, Novembro de 1996

[4] Milligan, Donald J.; Homeijer, Brian D. (2012). «An ultra-low noise MEMS accelerometer for seismic imaging». SENSORS.

Acesso em 24/04/2018 às 21:36

[5] Lígia J. Figueiredo, Ana R. Gafainz, Gustavo S. Lopes e Rúben Pereira. Aplicações de Acelerômetros. IAS 2007 – Instrumentação e Aquisição de Sinais. Monografia. Lisboa, Portugal, 19 de Dezembro 2007

[6] Gomes, M., Bássora L. A. “Sensoriamento Inercial de um birrotor”. Centro Universitário do Norte Paulista. SIICUSP 2014 – 22º Simpósio Internacional de Iniciação Científica e Tecnológica da USP

[7] “The Pi4J Project – Java I/O library for the Raspberry Pi”

Disponível em 27 de abril de 2018 às 20:45

<http://pi4j.com/pins/model-2b-rev1.html>

Acesso em 28 de abril de 2018 às 19:45

[8] “Raspberry Pi / Learning Raspberry Pi”,

Disponível em:

<https://learn.adafruit.com/introducing-the-raspberry-pi-2-model-b?view=all>

Acesso em 28 de abril de 2018 às 20:20

- [9] Grupo PET-Tele – “Tutorial de Introdução ao Python”, Niteroi – RJ, Maio de 2009
- [10] Richardson M., Wallace S. “Primeiros Passos com o Raspberry Pi”. Primeira edição. Novatec Editora Ltda, março/2013
- [11] “Acelerômetros: Uso Em Celulares E Detecção De Velocidade. Parte 1” Disponível em: <http://www.decom.ufop.br/imobilis/acelerometros-uso-em-celulares-e-deteccao-de-velocidade-parte-1/>
Acesso em 4 de setembro de 2018 às 19:35
- [12] “Cinemática dos corpos e das partículas” Disponível em
https://civil.fe.up.pt/pub/apoio/ano2/mec2/aulas_teoricas/elementos_apoio/apontamentos/Mec2-texto-cap1.pdf
Acesso em 5 de setembro de 2018 às 21:30
- [13] Camara, Romulo “Barramento e Protocolo I2C”, UNIVASF, disponível em:
<http://www.univasf.edu.br/~romulo.camara/novo/wp-content/uploads/2013/11/Barramento-e-Protocolo-I2C.pdf>
Acesso em 6 de outubro de 2018 às 14:20
- [14] Mendonça, S. H. – “SPI e I2C” – Faculdade de Engenharia da Universidade de Porto, disponível em: <https://paginas.fe.up.pt/~hsm/docencia/comp/spi-e-i2c/>
Acesso em 6 de outubro de 2018 às 16:30
- [15] ANALOG DEVICES, Manual de dados do fabricante ADXL345, disponível em:
<https://www.sparkfun.com/datasheets/Sensors/Accelerometer/ADXL345.pdf>
Acesso em 21 de outubro de 2018 às 20:18
- [16] Quadros D. “Módulo Acelerômetro ADXL345 – Introdução” – DQSoft, disponível em:
<http://dqsoft.blogspot.com/2015/07/modulo-acelerometro-adxl345-introducao.html>
Acesso em 22 de outubro de 2018 às 22:11
- [17] Pilling S. “Calculo Numérico”, UNIVAP, disponível em
https://www1.univap.br/spilling/CN/CN_Capt6.pdf
Acesso em 31 de outubro de 2018 às 02:10

[18] “The Accelerometer: Introduction”, disponível em
http://www.pyroelectro.com/tutorials/accel_intro/

Acessado em 25/04/2018 às 20:55

[19] “Programming and Calibration”, Adafruit, disponível em <https://learn.adafruit.com/adxl345-digital-accelerometer/programming>

Acessado em 31/10/2018 às 16:07