

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

RAQUEL FAUSTINO PIANCA

**SISTEMA DE TELEMETRIA PARA VEÍCULO ELÉTRICO COM
GERAÇÃO FOTOVOLTAICA AUXILIAR**

VITÓRIA
2018

RAQUEL FAUSTINO PIANCA

**SISTEMA DE TELEMETRIA PARA VEÍCULO ELÉTRICO COM
GERAÇÃO FOTOVOLTAICA AUXILIAR**

Trabalho de conclusão do curso de Engenharia Elétrica apresentado à Universidade Federal do Espírito Santo como requisito parcial para obtenção do título de Engenheiro eletricista.

Orientador: Prof. Dr. Oureste Elias Batista

Vitória
2018

RAQUEL FAUSTINO PIANCA

**SISTEMA DE TELEMETRIA PARA VEÍCULO ELÉTRICO COM GERAÇÃO
FOTOVOLTAICA AUXILIAR**

Trabalho de conclusão do curso de Engenharia Elétrica apresentado à Universidade Federal do Espírito Santo como requisito parcial para a obtenção do título de Engenheiro Eletricista.

19 de setembro de 2018.

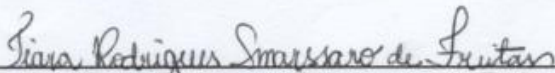
BANCA EXAMINADORA



Prof. Dr. Oureste Elias Batista



Prof. Dra. Eliete Maria de Oliveira Caldeira



Prof. Dra. Tiara Rodrigues Smarssaro de Freitas

A Elias Teixeira, que com sua presença fez
tudo ser luz sempre.

AGRADECIMENTOS

Agradeço aos meus pais por proporcionar as condições para eu chegar até aqui. Às minhas irmãs e irmão, pelo contínuo incentivo. Em especial, à minha irmã Sandra, sempre disposta a me ajudar e que cuidou de mim todos os anos passados na Universidade.

Agradeço aos Professores, que sempre estiveram dispostos a ajudar e contribuir para meu crescimento, sendo muitas vezes verdadeiros amigos. Em especial ao meu orientador, pela paciência, disponibilidade e amizade.

Agradeço também a Elias Teixeira, por ter me ajudado em todas as ocasiões, de todas as formas possíveis e por sempre acreditar no meu melhor.

Agradeço imensamente aos meus amigos pelos momentos compartilhados, que tornaram tudo mais fácil e feliz.

“O homem científico não almeja resultados imediatos. Ele não espera que suas ideias mais avançadas sejam rapidamente retomadas. Seu trabalho é como o de um agricultor para o futuro. Seu dever é estabelecer bases para aqueles que estão por vir e apontar o caminho a ser seguido.”

Nikola Tesla

RESUMO

Este projeto consiste no desenvolvimento de um sistema de telemetria que possui o microcomputador Raspberry PI como dispositivo central do sistema para leitura e processamento de variáveis relevantes na análise de desempenho energético de um veículo elétrico solar. O Projeto Veículo elétrico com geração solar fotovoltaica auxiliar propõe o monitoramento do sistema elétrico do mesmo para maximizar a eficiência energética do sistema e a vida útil de suas fontes de energia, sendo necessário, portanto um sistema de aquisição de dados para monitorar e avaliar as variáveis que influenciam nessa eficiência desejada. O sistema de telemetria desenvolvido foi projetado já se pensando em seu aprimoramento futuro, isso quer dizer que foi criada uma base geral para suporte a leitura e processamento de dados tanto analógicos quanto digitais, escolhendo sempre que possível, o caminho mais facilmente duplicável. Assim, este projeto não só estabeleceu as bases de um eficiente sistema de telemetria, como também apontou, sempre que possível, a direção a ser seguida em relação a assuntos relacionados, mas que não entram no escopo deste trabalho.

Palavras-chaves: Telemetria; Raspberry PI; Carro elétrico; Energia solar.

ABSTRACT

This project encapsulates the development of a telemetry system that has the Raspberry Pi micro-computer as the system's central device, which reads and processes relevant variables in the energy performance analysis of a solar electric vehicle. The Electric Vehicle Project with auxiliary photovoltaic solar generation aims to maximize the energy efficiency and the life cycle of its energy sources and therefore, a necessary data acquisition system is required to monitor and evaluate the variables which influence this desired efficiency. The developed telemetry system was designed considering its likely future improvement, which means that a general platform has been created to support the reading and processing of data, both analog and digital, selecting the most easily duplicable path. Thereby, this project not only sets up the basis for an efficient telemetry system, but also indicates, whenever possible, the direction to be followed in related topics that are not covered within this research.

Keywords: Telemetry; Raspberry PI; Electric car; Solar energy.

LISTA DE FIGURAS

Figura 1 – Raspberry PI 3 modelo B.....	17
Figura 2 – Conversor A/D ADS1115.....	17
Figura 3 – Sensor de corrente ACS758	20
Figura 4 – Princípio de funcionamento baseado no efeito Hall	21
Figura 5 – Sensor de corrente ACS712	21
Figura 6 – Sensor de tensão	22
Figura 7- Sensor de temperatura e umidade DTH11	23
Figura 8 – Módulo GY-521	24
Figura 9 – Localização dos pinos GPIO na Raspberry PI.....	25
Figura 10 – Conexão dos dispositivos no barramento no protocolo I ² C	26
Figura 11 – Triciclo à pedal	28
Figura 12 – Esquema Geral	Erro! Indicador não definido.
Figura 13 – Esquema Geral do sistema.....	29
Figura 14 – Componentes utilizados nos testes	31
Figura 15 – Conversor de Tensão de saída USB de 5 V	31
Figura 16 – Esquema de ligação do Conversor A/D, sensor de corrente e sensor de tensão.....	32
Figura 17 – Esquema de ligação do módulo GY-521 e do sensor DTH11	32
Figura 18 – Obtenção do endereço de IP da placa Raspberry PI	33
Figura 19 – Habilitação do modo SSH da Raspberry PI	34
Figura 20 – Terminal de simulação do PuTTY.....	34
Figura 21 – Acesso remoto ao terminal da Raspberry PI	35
Figura 22 – Acesso remoto ao ambiente gráfico da Raspberry PI através do cliente TightVNC	36
Figura 23 – Área gráfica da Raspberry PI	36
Figura 24 – Ligação do módulo GY-521 ao Raspberry PI	38
Figura 25 - Identificação do endereço I ² C do dispositivo no barramento	39
Figura 26 – Acesso remoto à plataforma de programação da linguagem Python	39
Figura 27 – Visualização dos dados em tempo real fornecidos pelo módulo GY-521	40
Figura 28 – Ligação do módulo GY-521, conversor A/D e sensor DHT11 ao Raspberry.....	41

Figura 29 – Visualização dos dados lidos pelo sensor DHT11	42
Figura 30 – Ligação do ADS1115 ao Raspberry	43
Figura 31 – Endereços I ² C do módulo GY-521 e do ADS1115 no barramento	43
Figura 32 – Visualização dos canais do conversor A/D	44
Figura 33 – Componentes utilizados nos testes do conversor A/D	45
Figura 34 – Leitura dos sensores analógicos através do Conversor A/D	46
Figura 35 – Leitura do conversor A/D	46
Figura 36 – Relações de tensão e corrente com a variação de irradiância e temperatura	50
Figura 37 – Características de temperatura da placa fotovoltaica CS6P -265P	51
Figura 38 – Sensor de corrente ACS758 a ser utilizado na leitura do motor	52

SUMÁRIO

1	INTRODUÇÃO.....	13
2	REFERENCIAL TEÓRICO.....	15
2.1	Telemetria	15
2.2	Raspberry PI	15
2.2.1	Raspberry PI 3 modelo B	16
2.3	Conversor analógico digital	17
2.4	Sensores.....	18
2.4.1	Sensor de corrente AC/DC 100 A de efeito Hall ACS758.....	20
2.4.2	Módulo sensor de corrente AC/DC 30 A de efeito Hall ACS712.....	21
2.4.3	Sensor de tensão	22
2.4.4	Sensor de temperatura	22
2.4.5	Módulo sensor GY-521	23
2.5	GPIO (<i>General Purpose Input/Output</i>)	24
2.6	Protocolo de comunicação <i>I²C</i>	26
2.7	Linguagem de programação <i>Python</i>	27
2.8	PuTTY.....	27
2.9	TightVNC.....	27
3	SISTEMA DE TELEMETRIA PROPOSTO.....	28
3.1	A placa Raspberry PI como dispositivo central do sistema	30
3.2	Componentes utilizados.....	30
4	TESTES E RESULTADOS	32
4.1	Esquemático das ligações dos sensores.....	32
4.2	Configurações da Raspberry PI	33
4.2.1	Configuração para acesso remoto via SSH e TightVNC	33
4.2.2	Configurações básicas de acesso	36
4.2.3	Bibliotecas instaladas.....	37
4.3	Circuito para a leitura do módulo GY-521	37
4.3.1	Hardware.....	37
4.3.2	Programação e resultados	38
4.4	Implementações para aquisição de dados dos sensores digitais	40

4.4.1 Hardware para o sensor DHT11	40
4.4.2 Programação e resultados	41
4.5 Implementação do Conversor A/D ADS1115.....	42
4.5.1 Hardware.....	42
4.5.2 Programação e resultados	43
5 ARMAZENAMENTO DE DADOS	47
6 RELEVÂNCIA DOS SENSORES UTILIZADOS NESTE TRABALHO PARA O ALCANCE DOS OBJETIVOS NO PROJETO DO VEÍCULO ELÉTRICO SOLAR ..	49
6.1 Temperatura e umidade na geração de energia solar fotovoltaica	49
6.2 Acelerômetro e giroscópio na análise da dinâmica do Veículo.....	51
6.3 Monitoramento de tensão e corrente.....	52
7 CONSIDERAÇÕES FINAIS	52
REFERÊNCIAS	55

1 INTRODUÇÃO

Os carros elétricos já são uma realidade em vários países do mundo, e a tendência é que continuem se expandindo no mercado, pois, em uma realidade onde a poluição do ar se tornou um grave problema, a possibilidade de veículos elétricos que utilizem fontes limpas e renováveis de energia elétrica, como a solar, é desejável e necessário.

Movidos principalmente por questões ambientais, a Europa e a China se encontram entre os principais investidores de tecnologias que tornem os veículos elétricos híbridos comercializáveis. Na Alemanha, por exemplo, está previsto para o segundo semestre de 2019 a produção em série do carro solar Sono Sion da Sono Motors, que possuirá uma autonomia de 250km, sendo 30km garantidos pela energia gerada pelas células fotovoltaicas [1]. Na Noruega, país mais adiantado na tecnologia de carros elétricos e híbridos, 37% dos carros vendidos hoje são 100% elétricos [2]. A China, que atualmente possui cerca de 1,2 milhões de automóveis em circulação, começou a testar uma rodovia inteligente na cidade de Jinan, que através de células fotovoltaicas, irá gerar energia para recarga dos veículos [3].

Já o Brasil caminha lentamente, segundo a Associação Nacional dos Fabricantes de Veículos Automotores (Anfavea), apenas 0,05% dos veículos comercializados são elétricos ou híbridos. Isso se deve tanto à falta de infraestrutura quanto à falta de incentivos fiscais do Governo Federal [4]. Nesse contexto, visto o enorme potencial do Brasil para geração de energia elétrica fotovoltaica devido ao seu clima tropical, é interessante que se invista em pesquisas e projetos que viabilizem o futuro dos carros elétricos solares no Brasil.

Apesar de possuir vantagens como utilização de fonte limpa e renovável de energia, melhor eficiência energética dos motores elétricos (cerca de 90%) em relação aos motores de combustão (que não passam de 40%), menor custo de operação e manutenção, o carro elétrico solar também apresenta pontos negativos, como baixa autonomia, pouca durabilidade da bateria e tempo de recarga elevado [5].

Assim, este trabalho propõe um sistema de telemetria para ser implementado no projeto Veículo elétrico com geração solar fotovoltaica auxiliar, para o monitoramento de variáveis que sejam relevantes no gerenciamento de energia do

sistema elétrico do veículo, para através dos dados adquiridos traçar estratégias para maximizar a vida útil das fontes de energia e a eficiência energética do sistema [6].

O objetivo geral deste trabalho é estabelecer as bases de um sistema de telemetria que seja capaz de coletar os dados pertinentes, processá-los e armazená-los para futuras análises. Seus objetivos específicos consistem em definir e configurar o dispositivo central de processamento do sistema, definir qual a melhor linguagem de programação a ser utilizada, estabelecer a comunicação com sensores analógicos e digitais, definir uma forma de armazenamento dos dados e analisar como os sensores utilizados neste projeto podem ser relevantes na avaliação de eficiência desejada.

A metodologia a ser utilizada no projeto consistirá primeiramente de uma pesquisa bibliográfica para a familiarização com o tema e assuntos relacionados, a fim de encontrar opções viáveis de soluções. Em seguida, será realizada a montagem de um protótipo que simula o funcionamento real do sistema de telemetria proposto e através de testes e simulações serão avaliados seu desempenho e aplicabilidade.

Este trabalho está estruturado em 7 Capítulos, sendo o Capítulo 1 esta introdução. O Capítulo 2 se trata do embasamento teórico necessário para o desenvolvimento do projeto e o Capítulo 3 faz uma breve descrição sobre o projeto do veículo elétrico com geração solar fotovoltaica auxiliar para o qual este trabalho será desenvolvido e descreve o sistema de telemetria proposto e os componentes utilizados para o seu desenvolvimento.

No Capítulo 4 são descritos os testes e resultados obtidos nos processos de configuração, montagem e leitura dos sinais dos sensores utilizados. O Capítulo 5 descreve a forma como foram armazenados os dados lidos pelos sensores.

O Capítulo 6 faz uma análise de como o uso dos sensores utilizados neste Trabalho podem ser relevantes para os objetivos do projeto descrito no Capítulo 3. No Capítulo 7 são feitas as considerações finais e dadas sugestões de continuidade deste Trabalho.

2 REFERENCIAL TEÓRICO

2.1 Telemetria

A palavra telemetria tem sua origem etimológica do grego *tele*, que significa remoto, e *metron*, que quer dizer medida. Assim, telemetria significa realizar medições a distância ou remotamente [7]. É utilizada em pontos geograficamente remotos ou que estão em constante movimento (como os automóveis, por exemplo), ou ainda, onde é inviável estabelecer cabeamento [7]. Muitas vezes a telemetria funciona em conjunto com o armazenamento de dados, que é a gravação dos canais de telemetria para futuras análises.

A telemetria pode se referir à transmissão de dados de diversas formas, como por fibra ótica e através de redes de computadores, por exemplo, não se limitando à transmissão sem fio como geralmente é atribuída a ela [9]. Sua aplicação ocorre em diversas áreas como agricultura, meteorologia, nos sistemas de tratamento de água e esgoto, no monitoramento de energia, entre várias outras [9].

Assim, a telemetria é fundamental para o desenvolvimento de um projeto, pois através dela que pode-se monitorar variáveis importantes para avaliação do seu desempenho.

2.2 Raspberry PI

A *Raspberry PI* é um computador de pequenas dimensões, desenvolvido pela *Raspberry PI Foundation* no Reino Unido. Possui todos os recursos de um computador, com a vantagem de seu reduzido tamanho e baixo custo [10]. Ao contrário de dispositivos similares, como Arduino, a *Raspberry PI* oferece suporte para a instalação de um sistema operacional completo, com todos os componentes essenciais de um PC (*Personal Computer*) de tamanho regular [11].

Pode operar com diversas distribuições do Linux, sendo a mais conhecida e recomendada, a distribuição *Raspbian*, que é uma derivação da distribuição *Debian* específica para a plataforma [12]. A *Raspberry PI* é uma excelente alternativa para o desenvolvimento de projetos que requerem capacidade de processamento e performances avançadas.

2.2.1 Raspberry PI 3 modelo B

Para a escolha do dispositivo a ser usado na leitura dos sensores, era necessário que este possuísse memória disponível para armazenamento interno de dados ou suporte para cartão de memória, estabelecesse facilmente comunicação com a *internet* e que suportasse uma linguagem de programação de fácil manipulação dos periféricos a ele conectado. Considerando esses requisitos para a escolha da placa controladora, escolheu-se a placa *Raspberry PI*, pois se já se possuía o modelo da mesma e esta atendia aos requisitos.

A versão do dispositivo escolhida foi a *Raspberry PI 3* modelo B. Ele será o dispositivo central do sistema de telemetria, irá ler as informações dos sensores, processá-las e armazená-las.

Seu processador é um *quad-core* de 1,2 GHz, que possui alta eficiência por *clock* e suporta instruções de 64 bits. Possui 1GB de memória RAM, possui suporte a redes sem fios sem precisar de adaptadores, 40 pinos GPIO (*General Purpose Input/Output*), 4 portas USB 2.0, *slot* para cartão micro SD (*Secure Digital*), interface de câmera, de display e porta *ethernet* [12].

A Figura 1 mostra a placa *Raspberry PI 3* modelo B. Suas principais características são:

- Processador *Broadcom BCM2837 64bit ARMv8 Cortex-A53 Quad-Core*
- *Clock* 1.2 GHz
- Memória RAM: 1GB
- Adaptador *Wifi* 802.11n integrado
- *Bluetooth* 4.1 BLE integrado
- Conector de vídeo HDMI
- 4 portas USB 2.0
- Conector *Ethernet*
- Interface para câmera (CSI)
- Interface para display (DSI)
- *Slot* para cartão micro *SD*
- Conector de áudio e vídeo
- *GPIO* de 40 pinos
- Dimensões: 85 x 56 x 17mm

Figura 1 – *Raspberry PI 3* modelo B

Fonte : [13]

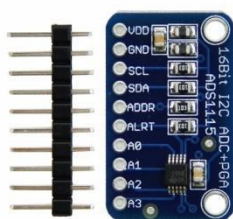
2.3 Conversor analógico digital

Um conversor A/D (Analógico/Digital) é um circuito eletrônico capaz de amostrar uma quantidade analógica e sair com uma representação digital da mesma [9]. Os conversores A/Ds podem ter diversos bits de precisão, dessa forma, podem ser de 8 bits, com 256 níveis de quantização, 10 bits com 1024 níveis, 12 bits com 4096 níveis e assim sucessivamente.

O conversor A/D normalmente é especificado em termos de quão depressa pode fazer a conversão. Isto pode ser especificado em termos da máxima taxa de gatilhamentos ou o pior caso de demora do gatilhamento até que o dado digital esteja disponível na saída [9].

O conversor A/D utilizado nesse projeto é o ADS1115 de 16 bits, mostrado na Figura 2, que possui 4 canais de alta precisão, fazendo uma amostragem de 860 amostras por segundo através do protocolo de comunicação I^2C . Funciona com tensões entre 2 e 5,5V, e a tensão máxima nos pinos analógicos é igual a tensão de alimentação. Os pinos analógicos podem ser programados como 4 pinos independentes, ou dois canais diferenciais [13].

Figura 2 – Conversor A/D ADS1115



Fonte: [14]

As especificações do ADS1115 são:

- Precisão de 16 bits
- Tensão de Operação – 2 à 5,5VCC
- Consumo de corrente - 150 μ A
- Taxa de dados programável – 8SPS à 860SPS
- Oscilador interno
- Interface de comunicação I²C
- 4 endereços I²C disponíveis
- Comparador programável
- Amplificador de ganho programável (PGA)
- Dimensões: 28 x 17 x 2,5mm

2.4 Sensores

Sensores são dispositivos capazes de transformar uma grandeza física em um sinal elétrico, respondendo a um estímulo físico/químico mensurável analogicamente [15]. Assim, ele capta e converte um fenômeno físico, como temperatura, luminosidade, em um sinal elétrico, fazendo uma interface entre o mundo físico e o mundo dos dispositivos eletrônicos. Sua saída geralmente fornece um sinal de tensão ou corrente que é enviada a um circuito eletrônico para processamento [16].

Apesar de suas características individuais únicas, os sensores possuem características gerais que devem ser observadas para se verificar sua adequação ao projeto a que se destina. Segundo [16] essas características são:

- Função de transferência: mostra a relação funcional entre um sinal de entrada físico e o sinal de saída elétrico.
- Sensibilidade: é a razão entre uma pequena variação do sinal elétrico para uma pequena variação do sinal físico. Assim, é derivada a partir da função de transferência em relação ao sinal de entrada. Por exemplo, pode-se falar em volts/Celsius, ou milivolts/decibel,

significando a variação em volts na saída do sensor em relação à variação da temperatura em Celsius, ou ainda a variação em milivolts em relação à variação da entrada em decibéis, respectivamente. Diz-se que um sensor possui alta sensibilidade quando uma pequena variação na entrada resulta em uma variação grande de tensão elétrica na saída do sensor.

- **Faixa dinâmica:** é a faixa de valores do sinal de entrada que podem ser convertidos em sinais elétricos pelo sensor. Sinais de entrada fora desta faixa ainda podem ser convertidos em sinais elétricos, porém podem resultar em valores imprecisos.
- **Precisão:** Define o maior valor de erro esperado entre um sinal de saída ideal e o sinal real, obtido. Por exemplo, pode-se ter um sensor de temperatura cuja incerteza é de 5%. Neste caso, o sensor é mais preciso do que outro cujo valor de incerteza é de 10%.
- **Resolução:** é a menor variação do sinal de entrada que é detectável. Diz respeito à capacidade do sensor de detectar uma mudança no sinal de entrada em um intervalo de tempo específico. Por exemplo, um sensor X pode detectar variações no sinal de entrada a cada 20ms, ou seja, 50 medições por segundo, enquanto um sensor Y, a cada 10ms, equivalente a 100 medições por segundo. Neste caso, o sensor Y possui maior resolução, pois é capaz de detectar mais amostras de sinal em um mesmo intervalo de tempo.
- **Temperatura de operação:** faixa de temperaturas na qual se pode garantir que o sensor irá operar como esperado, de acordo com suas outras características. Usar o sensor em um ambiente cuja temperatura está fora desta faixa pode ocasionar problemas que vão desde leituras incorretas até a inutilização do dispositivo.

- Níveis de saída: se referem aos valores de tensão gerados na saída do sensor. Podem ser níveis contínuos, variando diretamente com a medida física em sua entrada, ou ainda nível discreto, por exemplo, apresentando um nível de tensão alto para uma condição de entrada e outro nível baixo para uma condição diferente de entrada (sensor de disparo).

2.4.1 Sensor de corrente AC/DC 100 A de efeito Hall ACS758

O sensor de corrente ACS758, mostrado na Figura 3, é um sensor capaz de medir corrente alternada e corrente contínua de até 100 A. Seu princípio de funcionamento baseia-se no efeito *Hall*. Assim, ao circular uma corrente pelo sensor um campo magnético será gerado, e através do efeito *Hall*, uma tensão proporcional de saída será produzida [17], como mostrado na Figura 4. Os pinos de ligação para a passagem da corrente a ser medida são completamente isolados da saída.

As especificações do ACS758 são:

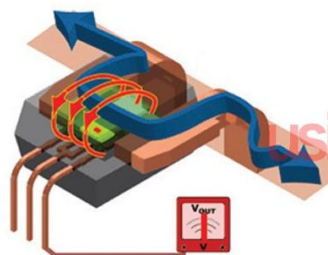
- Tensão de alimentação – 3 a 5,5 V
- Medição de corrente – até 100 A
- Saída analógica – 40mV/A
- Largura de banda – 120kHz
- Pinos – OUT/GND/VCC
- Dimensões (CxLxA) – 22x14x7mm
- Peso – 5g

Figura 3 – Sensor de corrente ACS758



Fonte: [19]

Figura 4 – Princípio de funcionamento baseado no efeito *Hall*



Fonte: [18]

2.4.2 Módulo sensor de corrente AC/DC 30 A de efeito Hall ACS712

O ACS712, mostrado na Figura 5, é capaz de fazer medições de -30A a +30A. Também utiliza o efeito *Hall*, gerando na saída do módulo uma tensão proporcional à corrente detectada de 66mV/A [20]. Este sensor possui sinal de saída analógica, seus três terminais de ligação são VCC, DATA e GND.

Figura 5 – Sensor de corrente ACS712



Fonte: [21]

As especificações do ACS712 são:

- Tensão de alimentação – 5 V
- Corrente máxima – 30 A
- Tempo de resposta – 5 us
- Largura de banda – 80kHz
- Razão de saída – 66mV/A
- Dimensões (C x L x A) – 32x13x14mm
- Peso – 3,2 g

2.4.3 Sensor de tensão

O Sensor de Tensão DC (*Direct Current*) 0-25 V mostrado na Figura 6 realiza medições de tensões contínuas na faixa de 0 a 25 V. Seu princípio de funcionamento é baseado em divisores de tensão. Na entrada do módulo, pode ser conectado um valor de tensão DC até cinco vezes maior que a tensão da porta analógica [22]. Para a *Raspberry PI*, por exemplo, pode ser conectado até 25V para 5 V de entrada ($5V * 5$) e para entrada de 3,3V será suportado até 16,5v ($3,3V * 5$).

Figura 6 – Sensor de tensão



Fonte: [22]

As especificações deste sensor são:

- Tensão de entrada – 0 a 25 V
- Faixa de detecção – 0,02445 - 25VDC
- Resolução analógica – 0,00489VDC
- Valor dos resistores – 30K Ω e 7,5K Ω
- Proporção de divisão – 5:1
- Tolerância do resistor – 1%
- Dimensões – 13mm(L) X 12mm(A) X 26mm(C)
- Peso – 2g

2.4.4 Sensor de temperatura

Sensores de temperatura fornecem uma saída proporcional à temperatura a que está exposto. A maioria dos sensores de temperatura possui um coeficiente de temperatura positivo, a tensão sobe à medida que a temperatura sobe, porém, existem sensores com coeficiente negativo, onde a tensão diminui à medida que a temperatura sobe [23].

Dentre as diversas opções de sensores disponíveis no mercado, para este projeto escolheu-se o módulo sensor DTH11 [24], mostrado na Figura 7, que monitora temperatura e também a umidade.

Figura 7- Sensor de temperatura e umidade DTH11



Fonte: [25]

Este sensor possui sinal de saída digital, seus três terminais de ligação são GND, DATA e VCC.

As especificações do sensor são:

- Tensão de alimentação – 3 ~ 5,5 V
- Faixa de corrente – 0,5 ~ 2,5 mA
- Faixa de operação de umidade – 20 ~90%
- Faixa de operação de temperatura – 0 a 50 ° *Celsius*
- Dimensões (C x L x A) – 28x13x8mm
- Peso – 3 g

2.4.5 Módulo sensor GY-521

Um módulo como o GY-521, mostrado na Figura 8, que contém um acelerômetro e giroscópio é essencial para avaliar o desempenho de um veículo em movimento, pois pode fornecer informações sobre sua aceleração, velocidade e posição. No módulo GY-521 tem-se em uma mesma placa um acelerômetro e um giroscópio de alta precisão, tudo isso controlado por um único circuito integrado, o MPU6050 [26].

O MPU6050 é um sensor de seis eixos, que fornece seis valores de saída, três valores do acelerômetro (correspondentes aos eixos X,Y e Z) e três do giroscópio. Tanto o acelerômetro como o giroscópio utiliza o protocolo *I²C* para se comunicar com controladores, no caso, a *Raspberry PI 3* modelo B. Esse CI (Circuito

Integrado) já possui nele um circuito eletrônico que cuida da parte dos complexos cálculos do sensor [27].

Possui alta precisão devido ao conversor analógico digital de 16 bits que possui para cada canal. E também possui um sensor de temperatura embutido, que permite medições entre -40° e $+85^{\circ}\text{C}$ [28].

A comunicação com a *Raspberry PI* usa a interface *I²C*, por meio dos pinos *SCL* (*Serial Clock*) e *SDA* (*Serial Data*) do sensor. A alimentação do módulo pode variar entre 3 e 5 V [29].

Figura 8 – Módulo GY-521



Fonte: [30]

As especificações do módulo sensor são:

- *Chip* – MPU-6050
- Tensão de Operação – 3 a 5V
- Conversor A/D – 16 bits
- Comunicação – Protocolo padrão *I²C*
- Faixa do giroscópio – ± 250 , 500, 1000, 2000 $^{\circ}$ /s
- Faixa do acelerômetro – ± 2 , ± 4 , ± 8 , ± 16 g
- Dimensões – 20 x 16 x 1mm

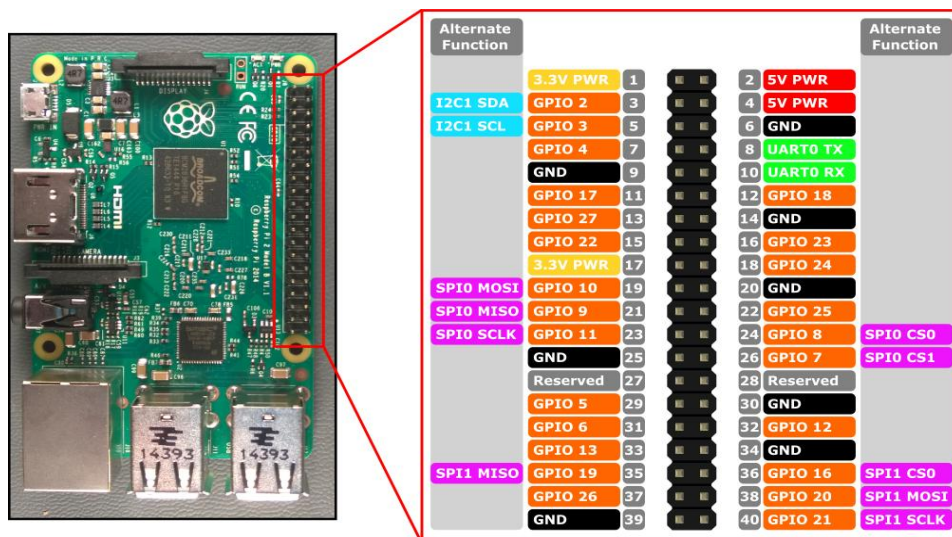
2.5 GPIO (*General Purpose Input/Output*)

O GPIO é o conjunto de pinos responsável por fazer a comunicação de entrada e saída de sinais digitais na *Raspberry PI*. O modelo usado neste projeto (*Raspberry PI* 3 modelo B) possui 40 pinos. São esses pinos que permitem a interação do microcontrolador da *Raspberry PI* com sistemas externos, fazendo a conexão do mundo físico com o digital.

A *Raspberry PI* não usa o sistema de lógica digital padrão chamado de TTL (*Transistor-transistor Logic*), onde o nível alto lógico é 5V, ele utiliza um sistema próprio onde o nível alto lógico é de 3,3V [31]. Assim, não se deve aplicar tensão maior que essa em suas portas, sob o risco da mesma ser danificada e ter-se também toda a placa queimada.

Na Figura 9 pode-se ver a posição e a identificação dos pinos GPIO na placa. É importante conhecer a localização dos pinos antes de usá-los, pois na placa do dispositivo não é mostrada a identificação dos mesmos.

Figura 9 – Localização dos pinos GPIO na Raspberry PI



Fonte: [32]

Os pinos representados pelas cores amarelo e vermelho são pinos de alimentação, de 3,3 e 5V respectivamente. E os da cor preta são pinos de terra.

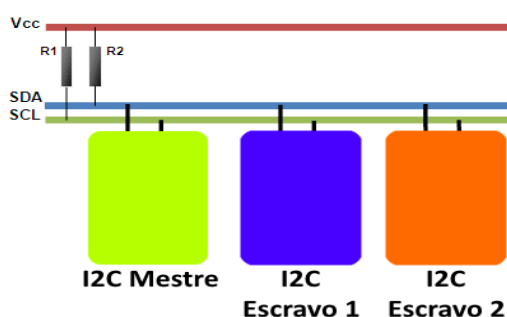
Neste projeto será utilizado os pinos GPIO 2 e 3 (SDA e SCL), responsáveis pela comunicação *I²C*, e alguns pinos dos representados apenas na cor laranja, responsáveis pelo envio e recebimento de dados digitais. Há também pinos digitais que possuem outros protocolos de comunicação como *SPI* (*Serial Peripheral Interface*), *RS-232*, comunicação serial via *UART* (*Universal Asynchronous Receiver-Transmitter*), o que torna o dispositivo acessível a uma faixa enorme de possíveis sensores a serem incorporados ao sistema.

2.6 Protocolo de comunicação I²C

Protocolos são utilizados em muitas áreas tecnológicas, pois possibilitam o transporte de informação entre dispositivos, estabelecendo regras e convenções que regem o funcionamento de diferentes comunicações [33]. Por serem utilizados em muitas áreas, diversos protocolos foram desenvolvidos ao longo dos anos para atender da melhor forma possível diferentes aplicações. Alguns exemplos são os protocolos I²C, UART e SPI [33].

Existem dois formatos que os protocolos podem seguir, serial e paralelo. O protocolo I²C é um protocolo de comunicação serial. I²C é um protocolo de barramento, ou seja, com os mesmo fios são conectado todos os dispositivos ao *setup* [34]. A maior vantagem desse protocolo de comunicação é que ele possibilita a comunicação simultânea com vários dispositivos utilizando poucas conexões [35]. As conexões são feitas por meio dos pinos SDA e SCL. A Figura 10 representa como os dispositivos estão conectados quando se utiliza o protocolo I²C.

Figura 10 – Conexão dos dispositivos no barramento no protocolo I²C



Fonte: [36]

Cada dispositivo I²C tem um endereço fixo para se comunicar com o microcontrolador e pode ser configurado para transmitir ou receber dados [36]. Assim, tanto a linha SDA como a SCL são bidirecionais, com os dispositivos alternando seu estado entre mestre e escravo [36].

O I²C trabalha no modelo mestre-escravo, com pelo menos um dispositivo atuando como mestre, e os demais dispositivos atuando como escravo. A função do mestre é coordenar a comunicação, sendo que é ele quem envia informações a determinado escravo ou consulta informações [37].

Devido à facilidade de comunicação e a simplicidade de seu funcionamento, o protocolo *I²C* é bastante utilizado como forma de comunicação entre sensores e microcontroladores.

2.7 Linguagem de programação *Python*

Python é uma linguagem de programação de alto nível, interpretada dinamicamente, com foco em legibilidade de código [11]. Ser uma linguagem interpretada significa dizer que ao se escrever um programa, este não será compilado (traduzido para uma linguagem de máquina), mas sim “lido” por outro programa (chamado de interpretador) que traduzirá para a máquina o que seu programa quer dizer [38].

Apesar de a *Raspberry PI* suportar programação em várias linguagens, optou-se pela utilização do *Python* neste projeto, pois além de ser uma linguagem de alto nível, possui uma sintaxe simples, e também é a mais indicada pelos próprios criadores do dispositivo, pois é a linguagem que mais facilita o controle dos pinos de comunicação externa (GPIO) da *Raspberry PI* utilizados no projeto.

2.8 PuTTY

PuTTY é um terminal de simulação *open source* desenvolvido para atuar como um cliente de conexões seguras para acesso remoto a servidores via *Shell Seguro* (*SSH*) e à construção de canais encriptografados entre servidores [39].

2.9 TightVNC

O *TightVNC* é um *software* que permite a visualização da tela de um computador remoto [40]. O *TightVNC* oferece dois modos de operação, cliente ou servidor. Numa comunicação de duas vias padrão, uma máquina atua como o cliente e o outro é o servidor. O servidor é uma máquina a ser controlada, e o cliente faz o controle remoto [41]. Dessa forma, o *software TightVNC* servidor deve ser instalado no computador que se deseja controlar, e o *TightVNC* cliente, no computador que irá realizar o controle.

3 SISTEMA DE TELEMETRIA PROPOSTO

O projeto do veículo elétrico com geração fotovoltaica auxiliar consiste em adaptar um triciclo a pedal motorizado, mostrado na Figura 11, onde será instalado um painel solar fotovoltaico que irá gerar energia para auxiliar a recarga do banco de baterias do veículo e outras funções, caracterizando assim, uma microrrede fotovoltaica autônoma móvel [6]. O objetivo geral da pesquisa, é gerenciar remotamente os fluxos de energia do sistema de potência do veículo, e com base na sua dinâmica, definir estratégias tecnológicas para maximizar a eficiência energética e a vida útil das fontes de energia [6].

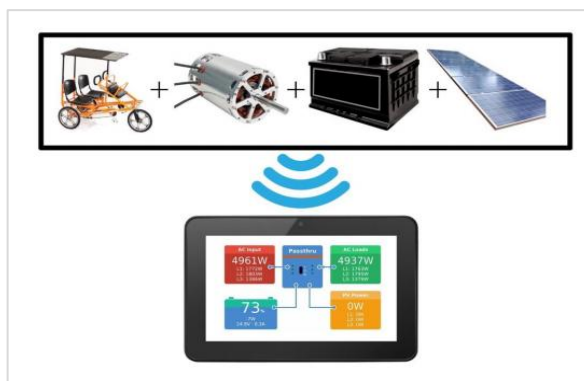
Figura 11 – Triciclo a pedal



Fonte: [6]

Para que esse monitoramento ocorra, o sistema contará com equipamentos gerenciadores e de controle das grandezas, como controlador de carga e gerenciador de energia. Também necessitará de um sistema de telemetria, para monitoramento em tempo real e armazenando de dados para análise posterior.

Figura 12 – Esquema Geral

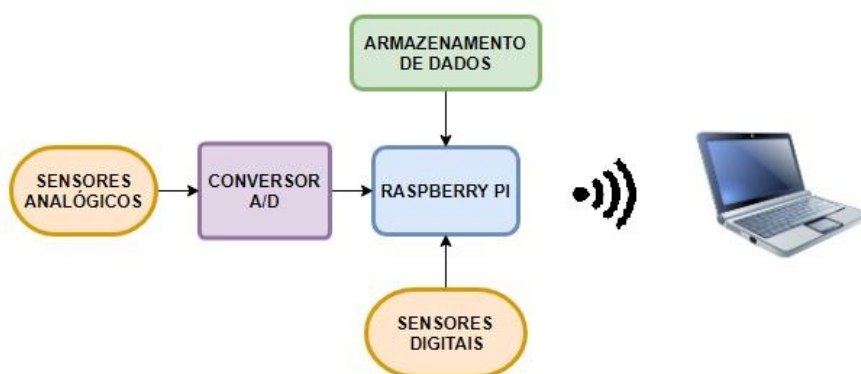


Fonte: [6]

Para que esse monitoramento ocorra, o sistema contará com equipamentos gerenciadores e de controle das grandezas, como controlador de carga e gerenciador de energia. Também necessitará de um sistema de telemetria, para monitoramento em tempo real e armazenando de dados para análise posterior.

O sistema de telemetria desenvolvido, mostrado no esquemático da Figura 13, não especificará em quais locais no sistema do projeto do veículo elétrico será feita as futuras medições, uma vez que este ainda não foi montado. Contudo, sabe-se, por exemplo, que certamente serão monitoradas corrente e tensão em diversos pontos do sistema para as análises energéticas necessárias. Assim, será projetado de forma que futuramente seja possível inserir mais sensores, sejam eles analógicos ou digitais.

Figura 13 – Esquema Geral do sistema



Fonte: Autor (2018)

3.1 A placa Raspberry PI como dispositivo central do sistema

No capítulo 2 pode-se ler as especificações desse dispositivo. Sua escolha se justifica por sua alta capacidade de processamento, de armazenamento de dados, pela sua relativa facilidade de programação devido ao fato de suportar várias linguagens e também por sua facilidade em se conectar com outros dispositivos, seja fisicamente ou virtualmente.

Os pinos de GPIO da *Raspberry PI* não fazem leitura de sinais analógicos, como explicado no capítulo 2, assim, se faz necessário o uso de um conversor A/D para fazer essa interface entre os dispositivos. Mas uma vez configurados a *Raspberry PI* e o conversor A/D e feito a programação de leitura dos canais, uma grande quantidade de sensores analógicos podem ser lidos, no caso específico deste projeto, 16 sensores.

3.2 Componentes utilizados

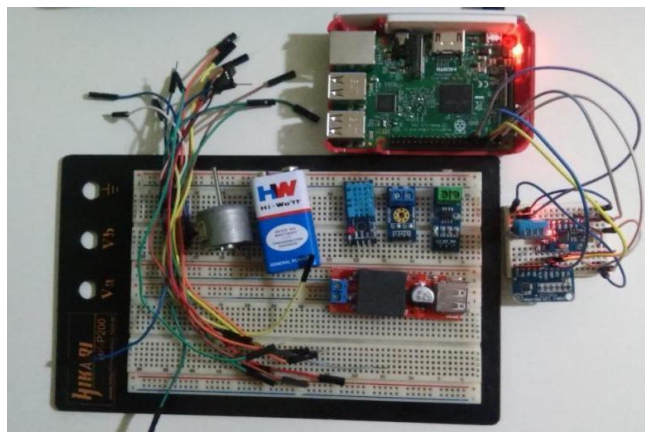
Para a escolha dos sensores utilizados considerou-se o protocolo de comunicação, se sua saída é analógica ou digital, e também a relevância que os dados fornecidos tem na análise do projeto do veículo elétrico solar.

Os componentes utilizados no projeto são mostrados na Figura 14. Representando os sensores digitais, escolheu-se o sensor de temperatura e umidade DHT11, que pode ser lido diretamente pelos pinos de GPIO da *Raspberry PI*. Seus dados podem ser usados futuramente para se analisar a influência da temperatura e umidade no desempenho dos módulos fotovoltaicos [15], por exemplo.

Os sensores analógicos utilizados serão o sensor de corrente ACS712 e um sensor de tensão que necessitarão do conversor A/D ADS1115 para se comunicar com a *Raspberry PI*.

Será também utilizado o módulo GY-521, que contém um acelerômetro e um giroscópio, que medem aceleração e rotação nos três eixos. Com a manipulação dos dados fornecidos, pode-se obter, por exemplo, velocidade do veículo, e relacioná-la com outras variáveis para análise de desempenho.

Figura 14 – Componentes utilizados nos testes



Fonte: Autor (2018)

O módulo conversor de saída USB 5 V mostrado na Figura 15 foi adquirido para fazer a alimentação da *Raspberry Pi* no sistema real a ser montado, ele pode ser conectado em até 25 V e fornece uma saída de 5 V. Porém, nos testes ele não foi utilizado, pois nestes, a *Raspberry Pi* estava sendo alimentada ou pelas saída USB de um *PC* ou estava usando seu próprio alimentador.

Figura 15 – Conversor de Tensão de saída USB de 5 V



Fonte: [45]

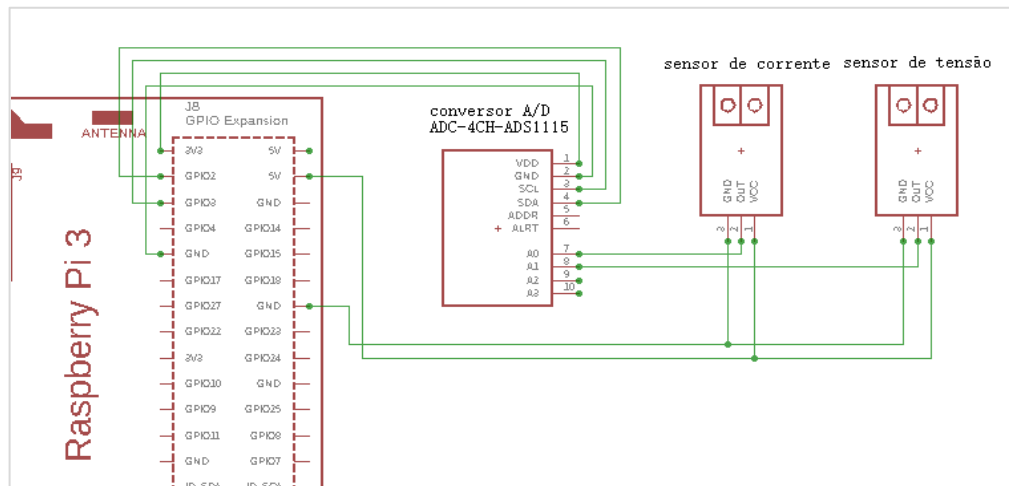
Deve-se ficar atento quanto a alimentação da placa, devendo ser de 5 V e uma corrente de no máximo 2,5 A. Caso a corrente seja maior a placa irá queimar, e caso seja menor, pode haver problemas no funcionamento da mesma, podendo a *Raspberry* não conseguir executar todas as funções desejadas. Ao se usar conversores de tensão ou carregadores conectados a rede elétrica, é recomendado que sejam testados antes, pois estes podem não estar corretamente isolados.

4 TESTES E RESULTADOS

4.1 Esquemático das ligações dos sensores

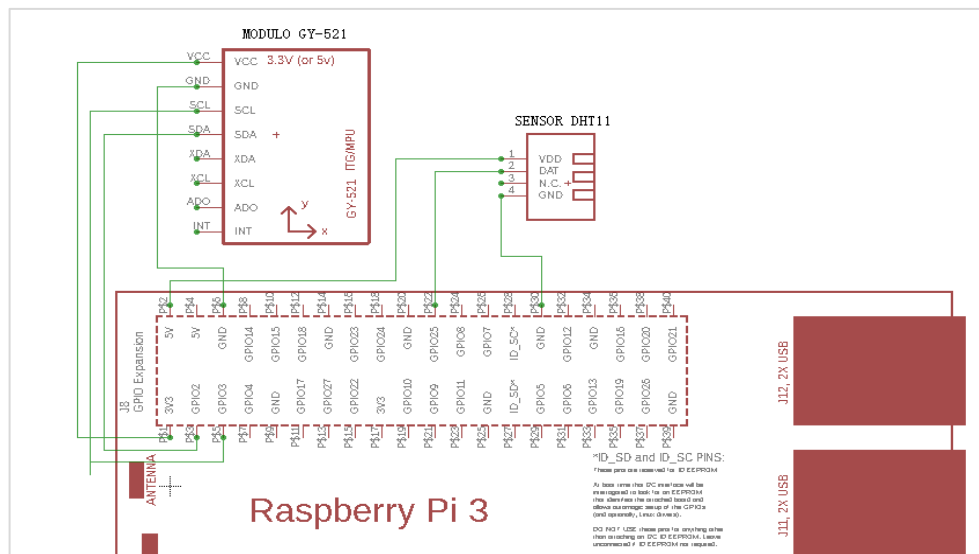
As Figuras 16 e 17 mostram o esquemático de ligação, feito no *software* E-CAD *EAGLE*, dos sensores utilizados ao *Raspberry Pi*.

Figura 16 – Esquema de ligação do Conversor A/D, sensor de corrente e sensor de tensão



Fonte: Autor (2018)

Figura 17 – Esquema de ligação do módulo GY-521 e do sensor DHT11



Fonte: Autor (2018)

4.2 Configurações da Raspberry PI

Para se utilizar a *Raspberry PI* como o dispositivo gerenciador desse sistema, algumas configurações foram necessárias para que este atendesse as necessidades do projeto. A primeira delas foi a configuração para o acesso remoto do dispositivo, para que este pudesse ser acessado de um computador qualquer, desde que este, esteja conectado na mesma rede que a *Raspberry PI*.

4.2.1 Configuração para acesso remoto via SSH e TightVNC

A possibilidade de acesso remoto é de alta importância, pois facilita a programação do dispositivo, uma vez que não é preciso conectar monitor, mouse e teclado ao *Raspberry PI* a fim de ter acesso à sua área de trabalho.

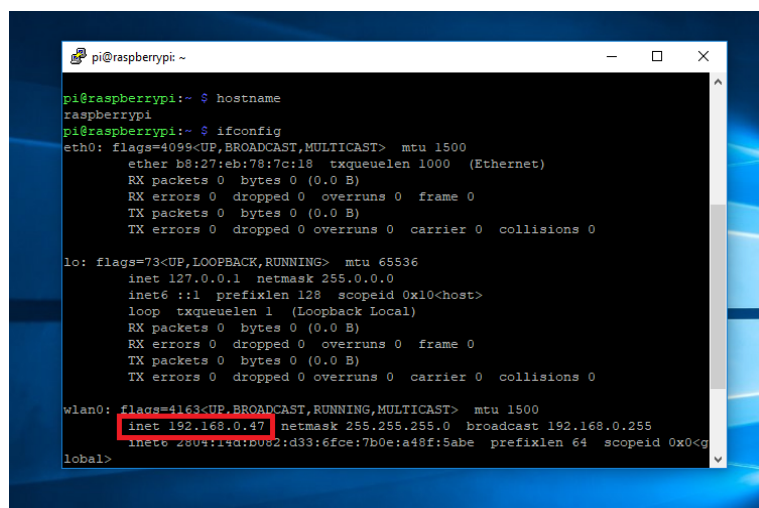
O acesso remoto pode ser feito para acessar apenas o terminal da *Raspberry PI* ou também para ter acesso gráfico de sua área de trabalho. Para este projeto foi preciso os dois tipos de acesso.

Primeiramente conectou-se um monitor, um mouse e um teclado à placa *Raspberry PI*, para o acesso à sua área de trabalho a fim fazer as configurações de rede necessárias. Feito isto, acessando o seu terminal, e digitando o comando:

Ifconfig

Descobriu-se o seu endereço de *IP (Internet Protocol)* na rede utilizada, necessário para o acesso remoto.

Figura 18 – Obtenção do endereço de IP da placa Raspberry PI



```
pi@raspberrypi:~$ hostname
raspberrypi
pi@raspberrypi:~$ ifconfig
eth0: flags=4096<UP,BROADCAST,MULTICAST> mtu 1500
    ether b8:27:eb:78:7c:18 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

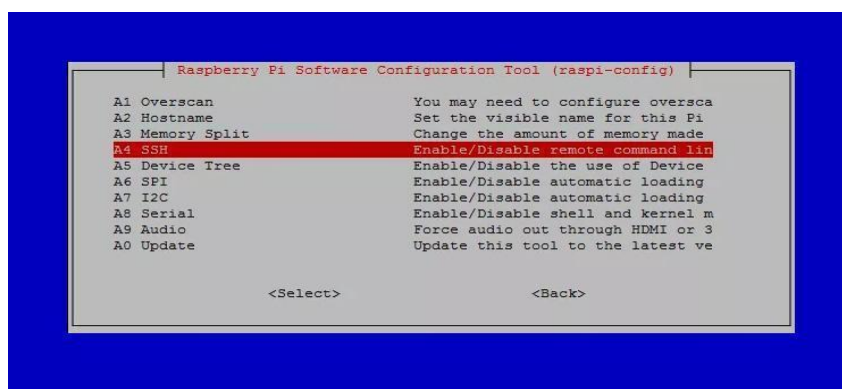
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1 (Loopback Local)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.47 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 2e04:14d:d002:d33:6fce:7b0e:a48f:Sabe prefixlen 64 scopeid 0x0<g
    local>
```

O acesso remoto utilizado foi feito via *SSH (Secure Socket Shell)* e *TightVNC*. O *SSH* também conhecido como *Secure Socket Shell* é um protocolo/aplicação que permite o acesso remoto ao terminal do *Raspberry PI* [42].

No menu de configurações básicas da *Raspberry PI* deve-se habilitar essa configuração, como mostrada na Figura 19.

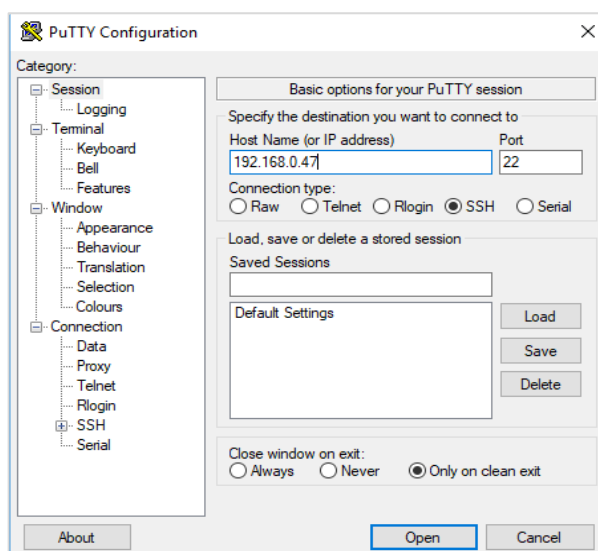
Figura 19 – Habilitação do modo *SSH* da *Raspberry PI*



Fonte: Autor (2018)

No *PC* que será utilizado para o acesso remoto foi instalado o terminal de simulação *PuTTY*. Ao executar o programa a janela mostrada na Figura 20 é exibida. Nela é inserido o endereço de *IP* da *Raspberry PI* para que a conexão remota com o terminal do mesmo seja estabelecida.

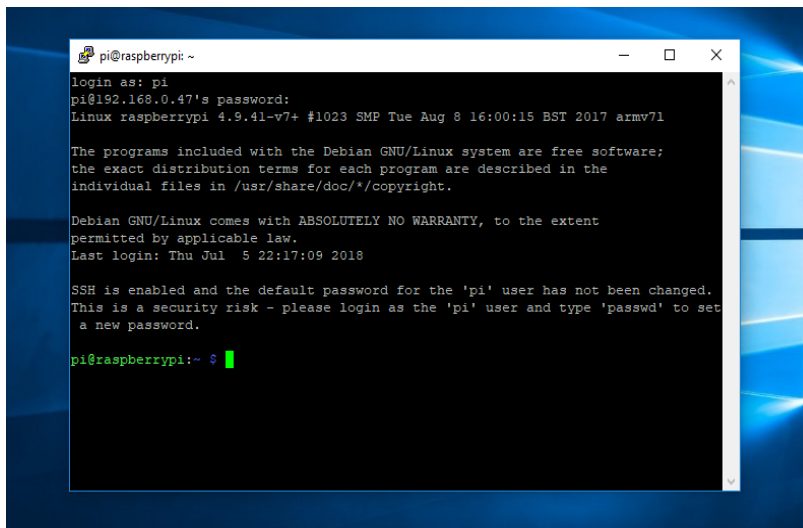
Figura 20 – Terminal de simulação do *PuTTY*



Fonte: Autor (2018)

Assim, o acesso ao terminal da *Raspberry PI* foi aberto e, inserido usuário e senha previamente configurados, passa-se ter controle sobre a *Raspberry PI* através do seu terminal como mostrado na Figura 21.

Figura 21 – Acesso remoto ao terminal da Raspberry PI



Fonte: Autor (2018)

Para o acesso ao ambiente de trabalho gráfico do *Raspberry PI*, instalou-se o servidor *TightVNC* no mesmo através do terminal, inserindo o comando:

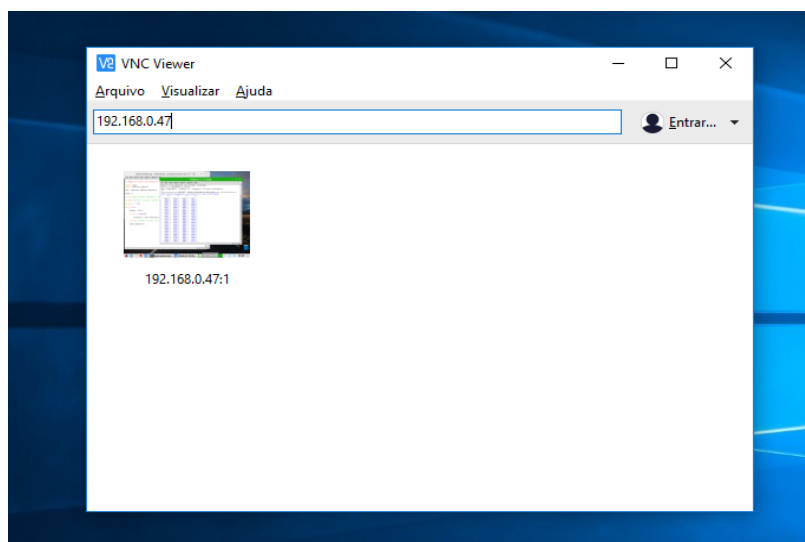
sudo apt-get install tightvncserver

Para executar o servidor basta digitar no terminal o comando:

Tightvncserver

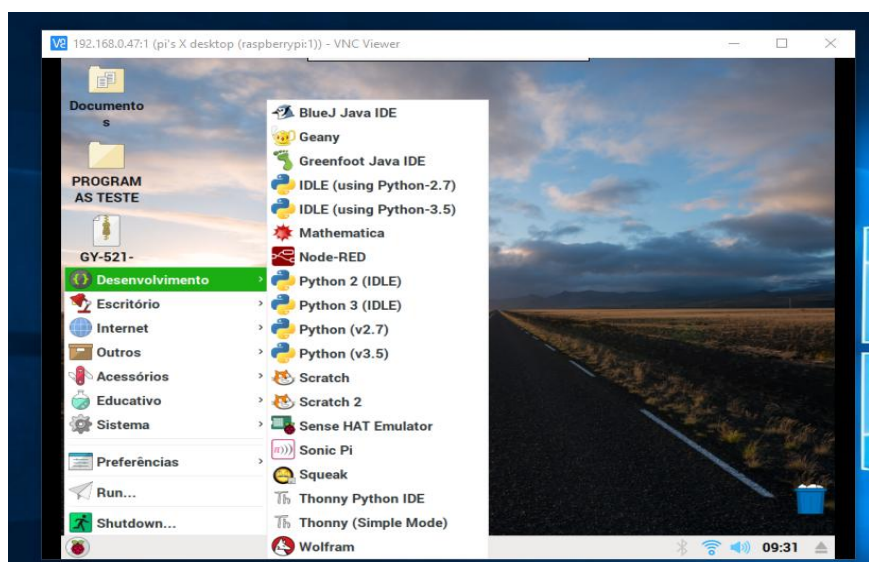
Em seguida, instalou-se o cliente para *PC* e definiu-se uma senha de acesso ao *TightVNC*. Agora, toda vez que se quiser acessar o ambiente gráfico de sua área de trabalho, basta fornecer o endereço de *IP* da *Raspberry PI* ao cliente *TightVNC* instalado no *PC* que a janela mostrada nas Figura 22 e 23 irá se abrir e fornecer o acesso remoto.

Figura 22 – Acesso remoto ao ambiente gráfico da *Raspberry PI* através do cliente *TightVNC*



Fonte: Autor (2018)

Figura 23 – Área gráfica da *Raspberry PI*



Fonte: Autor (2018)

4.2.2 Configurações básicas de acesso

Para haver a comunicação dos sensores com a *Raspberry PI* é necessário habilitar seus protocolos de comunicação. Nas suas configurações básicas, onde já se configurou o acesso remoto via *SSH*, deve-se habilitar os protocolos de comunicação *I²C*, *SPI* e também o acesso remoto as *GPIOs*.

4.2.3 Bibliotecas instaladas

Para uma maior otimização da programação de leitura dos sensores instalou-se bibliotecas de programação referentes a cada sensor, que já possuem funções prontas das configurações básicas desses dispositivos, como por exemplo, inicialização de comunicação.

Também foram instaladas bibliotecas em *Python* que possuem funções de acesso aos terminais de *GPIOs*.

As bibliotecas instaladas referentes aos sensores foram:

- *Adafruit Python ADS 1x15*: biblioteca responsável pelas configurações de inicialização do conversor analógico digital ADS1115.
- *Adafruit DHT*: responsável pelas inicializações dos sensores da família DHT.
- *GY-521*: responsável pelas configurações do módulo contendo acelerômetro e giroscópio.

As bibliotecas instaladas para controle dos *GPIOs* foram:

- *WiringPi*
- *Adafruit GPIO*
- *Gpiozero*
- *Pigpio*
- *RPi.GPIO*

As bibliotecas foram baixadas e instaladas com acesso via terminal.

4.3 Circuito para a leitura do módulo GY-521

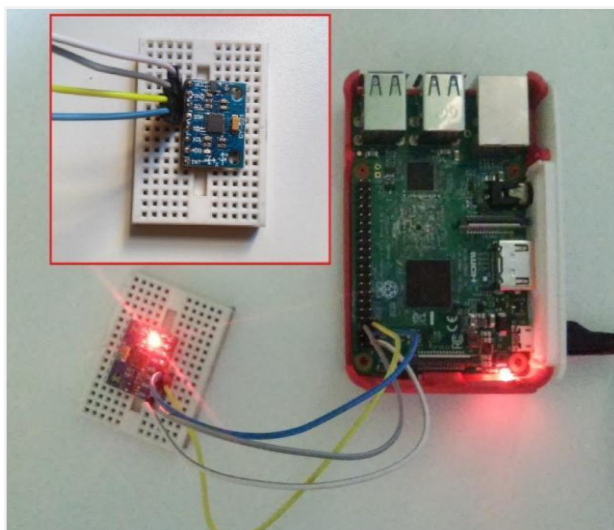
4.3.1 Hardware

A ligação do módulo GY-521 ao *Raspberry PI* é mostrada na Figura 24. Pode-se observar que além dos pinos de alimentação ainda restam outros quatro pinos, que neste projeto não serão utilizados.

Os pinos XDA e XCL trabalham neste módulo como um tipo de extensão para os pinos de comunicação SDA e SCL, tornando possível a conexão de outro dispositivo de mesmo padrão de comunicação I^2C , como por exemplo, um magnetômetro que pode aumentar as possibilidades do projeto [43].

Já o pino AD0, por sua vez, possui a capacidade de definir o endereço I^2C utilizado para a comunicação, caso o pino AD0 não esteja sendo utilizado, a comunicação estará definida no endereço 0x68 por padrão, porém, caso se queira conectar este pino em 3,3V, seu endereço será alterado para 0x69.

Figura 24 – Ligação do módulo GY-521 ao *Raspberry PI*



Fonte: Autor (2018)

4.3.2 Programação e resultados

Antes de dar início a programação, verificou-se se o dispositivo se encontrava corretamente conectado, acessando seu endereço I^2C para a comunicação. É recomendada essa verificação sempre que forem utilizados dispositivos de comunicação I^2C , para se ter certeza de que não haverá conflito na hora de ler o barramento. Digitando o comando ***i2cdetect -y 1*** no terminal, é devolvido como resposta o endereço do dispositivo como mostrado na Figura 25.

Figura 25 - Identificação do endereço I²C do dispositivo no barramento



```

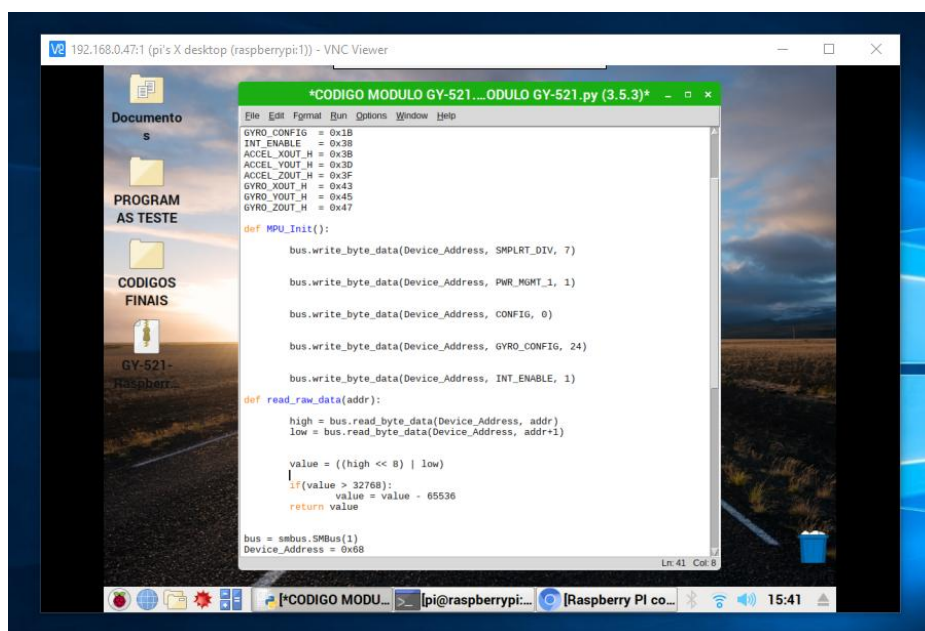
pi@raspberrypi: ~
Arquivo Editar Abas Ajuda
pi@raspberrypi:~$ i2cdetect -y 1
00:  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
10:  -- -- -- -- -- -- -- -- -- -- -- -- -- --
20:  -- -- -- -- -- -- -- -- -- -- -- -- -- --
30:  -- -- -- -- -- -- -- -- -- -- -- -- -- --
40:  -- -- -- -- -- -- -- -- -- -- -- -- -- --
50:  -- -- -- -- -- -- -- -- -- -- -- -- -- --
60:  -- -- -- -- -- 68 -- -- -- -- -- -- -- --
70:  -- -- -- -- -- -- -- -- -- -- -- -- -- --
pi@raspberrypi:~$

```

Fonte: Autor (2018)

A programação para leitura e uma primeira interpretação dos dados foi feita na linguagem *Python*, dentro do ambiente de desenvolvimento instalado no *Raspberry PI*, através do acesso remoto já mencionado. A Figura 26 mostra esse acesso.

Figura 26 – Acesso remoto à plataforma de programação da linguagem *Python*



```

*CODIGO MODULO GY-521...ODULO GY-521.py (3.5.3)*
File Edit Format Run Options Window Help
GYRO_CONFIG = 0x1B
INT_ENABLE = 0x38
ACCEL_XOUT_H = 0x3B
ACCEL_XOUT_L = 0x3D
ACCEL_ZOUT_H = 0x3F
GYRO_XOUT_H = 0x43
GYRO_XOUT_L = 0x45
GYRO_ZOUT_H = 0x47

def MPU_Init():
    bus.write_byte_data(Device_Address, SMPLRT_DIV, 7)

    bus.write_byte_data(Device_Address, PWR_MGMT_1, 1)

    bus.write_byte_data(Device_Address, CONFIG, 0)

    bus.write_byte_data(Device_Address, GYRO_CONFIG, 24)

    bus.write_byte_data(Device_Address, INT_ENABLE, 1)

def read_raw_data(addr):
    high = bus.read_byte_data(Device_Address, addr)
    low = bus.read_byte_data(Device_Address, addr+1)

    value = ((high << 8) | low)

    if(value > 32768):
        value = value - 65536
    return value

bus = smbus.SMBus(1)
Device_Address = 0x68

```

Fonte: Autor (2018)

Ao executar o programa, foram retornados os valores mostrados na janela da Figura 27. O giroscópio detecta a rotação nos eixos e fornece como saída graus por segundo (dps). O acelerômetro por sua vez, fornece a aceleração de um corpo em relação à gravidade.

Figura 27 – Visualização dos dados em tempo real fornecidos pelo módulo GY-521

```

Python 3.5.3 (default, Jan 19 2017, 14:11:04)
[GCC 6.3.0 20170124] on linux
Type "copyright", "credits" or "license()" for more information.
>>>
= RESTART: /home/pi/Área de Trabalho/CODIGOS FINAIS/CODIGO MODULO GY-521.py =

Lendo os dados do giroscopio e Acelerometro

Gx=0.44 °/s    Gy=0.48 °/s    Gz=-4.92 °/s    Ax=-0.70 g    Ay=0.53 g    Az=0.51 g
Gx=-1.29 °/s    Gy=0.22 °/s    Gz=-2.60 °/s    Ax=-0.62 g    Ay=0.48 g    Az=0.51 g
Gx=-0.14 °/s    Gy=0.40 °/s    Gz=0.38 °/s    Ax=-0.77 g    Ay=0.38 g    Az=0.45 g
Gx=-0.18 °/s    Gy=0.05 °/s    Gz=0.12 °/s    Ax=-0.75 g    Ay=0.42 g    Az=0.43 g
Gx=0.03 °/s    Gy=-0.11 °/s    Gz=0.56 °/s    Ax=-0.73 g    Ay=0.45 g    Az=0.43 g
Gx=-0.49 °/s    Gy=-0.05 °/s    Gz=-0.99 °/s    Ax=-0.64 g    Ay=0.58 g    Az=0.51 g
Gx=1.29 °/s    Gy=-2.49 °/s    Gz=0.07 °/s    Ax=-0.87 g    Ay=0.22 g    Az=0.40 g
Gx=6.07 °/s    Gy=-46.90 °/s    Gz=39.16 °/s    Ax=0.11 g    Ay=0.85 g    Az=1.28 g
Gx=-1.61 °/s    Gy=1.74 °/s    Gz=0.02 °/s    Ax=-0.73 g    Ay=-0.41 g    Az=0.60 g
Gx=14.09 °/s    Gy=3.24 °/s    Gz=-13.76 °/s    Ax=-1.16 g    Ay=0.52 g    Az=0.14 g
Gx=6.92 °/s    Gy=24.49 °/s    Gz=-44.40 °/s    Ax=-0.72 g    Ay=0.48 g    Az=0.31 g
Gx=-0.27 °/s    Gy=-7.80 °/s    Gz=9.63 °/s    Ax=-0.65 g    Ay=0.57 g    Az=0.44 g
Gx=-5.36 °/s    Gy=-18.48 °/s    Gz=10.34 °/s    Ax=-0.86 g    Ay=0.14 g    Az=0.28 g
Gx=0.85 °/s    Gy=-5.99 °/s    Gz=16.34 °/s    Ax=-0.02 g    Ay=-0.30 g    Az=0.92 g
Gx=2.60 °/s    Gy=-2.80 °/s    Gz=-3.95 °/s    Ax=-0.19 g    Ay=0.16 g    Az=1.09 g
Gx=1.85 °/s    Gy=2.19 °/s    Gz=1.82 °/s    Ax=-0.11 g    Ay=0.80 g    Az=0.41 g
Gx=-11.65 °/s    Gy=-56.83 °/s    Gz=115.89 °/s    Ax=-0.97 g    Ay=0.31 g    Az=0.44 g
Gx=-1.24 °/s    Gy=-30.20 °/s    Gz=0.40 °/s    Ax=-0.37 g    Ay=-0.09 g    Az=0.78 g
Gx=3.33 °/s    Gy=16.02 °/s    Gz=-4.69 °/s    Ax=-0.94 g    Ay=0.16 g    Az=0.36 g

```

Fonte: Autor (2018)

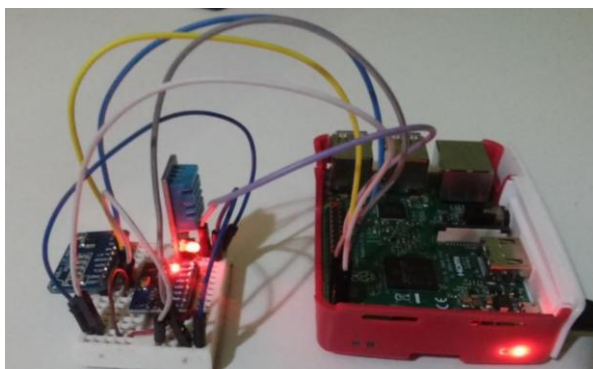
A janela dos dados mostrada na Figura 27 fornece os dados continuamente até que se interrompa a execução do programa, porém, caso seja necessário, pode-se determinar um tempo específico para a execução.

4.4 Implementações para aquisição de dados dos sensores digitais

4.4.1 Hardware para o sensor DHT11

Para a leitura do sensor de temperatura e umidade DHT11 foram feitas as ligações conforme o diagrama mostrado na Figura 18. A Figura 28 mostra a montagem utilizada do sensor DHT11 juntamente com o módulo GY-521 e o conversor A/D ADS1115.

Figura 28 – Ligação do módulo GY-521, conversor A/D e sensor DHT11 ao *Raspberry PI*



Fonte: Autor (2018)

A alimentação do DHT11, assim como a do acelerômetro, foi feita em 3,3 Volts para se aproveitar o mesmo terminal de alimentação. Mas também poderia ser feita em 5 Volts.

4.4.2 Programação e resultados

A programação de leitura do sensor DHT11 serve de base para qualquer sensor digital que se queira implementar futuramente no sistema, pois todas farão uso da biblioteca *RPI.GPIO*. O que eventualmente terá que mudar será o terminal da leitura digital e a biblioteca do respectivo sensor a ser acrescentado, caso exista uma.

Na rotina de programação deste sensor foi utilizada a biblioteca ***timer***. Por ser um sensor de temperatura, durante o processo de leitura, recomenda-se esperar um tempo considerável entre uma leitura e outra, visto que a mudança de temperatura precisa acontecer e se estabilizar para ser medida com maior precisão. Assim, a biblioteca ***timer*** facilita esse processo de espera, uma vez que este tempo é facilmente configurado por ela.

No programa utilizado, o tempo de espera entre as leitura foi de 5 segundos, que é o tempo recomendado pelo *datasheet* do sensor para uma leitura estável e confiável, como pode ser observado na Figura 29, que mostra a janela de exibição dos valores lidos pelo sensor.

Figura 29 – Visualização dos dados lidos pelo sensor DHT11



```
Python
File Edit Shell Debug Options Window Help
Python 2.7.13 (default, Jan 19 2017, 14:48:08)
[GCC 6.3.0 20170124] on linux2
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /home/pi/Área de Trabalho/CODIGOS FINAIS/ DHT11dados.py =====
***
Lendo os valores de temperatura e umidade
Temperatura = 27.0 Umidade = 64.0
Aguarda 5 segundos para efetuar nova leitura...
Temperatura = 27.0 Umidade = 65.0
Aguarda 5 segundos para efetuar nova leitura...
Temperatura = 27.0 Umidade = 65.0
Aguarda 5 segundos para efetuar nova leitura...
Temperatura = 27.0 Umidade = 66.0
Aguarda 5 segundos para efetuar nova leitura...
Temperatura = 28.0 Umidade = 67.0
Aguarda 5 segundos para efetuar nova leitura...
```

Fonte: Autor (2018)

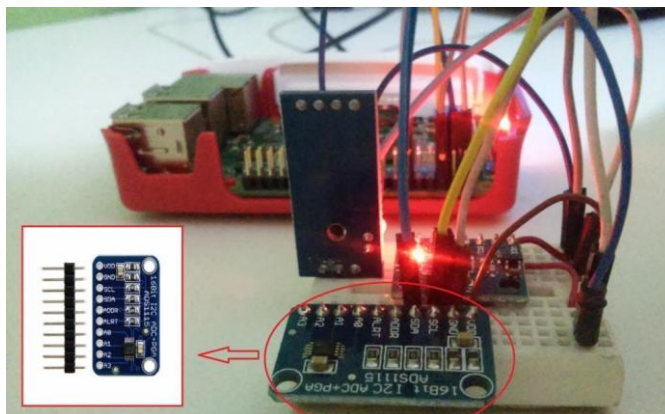
4.5 Implementação do Conversor A/D ADS1115

4.5.1 Hardware

A Figura 30 mostra a ligação do ADS1115 ao *Raspberry Pi*. O conversor A/D também foi alimentado com uma tensão de 3,3 Volts.

O conversor A/D ADS1115, assim como o módulo GY-521 se comunica através do protocolo *I²C*. Assim, tem-se dois dispositivos conectados aos terminais SCL (GPIO2) e SDA (GPIO 3) da *Raspberry Pi*. Mas, como visto anteriormente, isto não consiste em um problema, pois o protocolo *I²C* permite a comunicação de vários dispositivos simultâneos, desde que a cada dispositivo seja atribuído um endereço no barramento.

Figura 30 – Ligação do ADS1115 ao *Raspberry PI*



Fonte: Autor (2018)

4.5.2 Programação e resultados

Antes de iniciar a programação para o conversor A/D verificou-se, através do terminal do *Raspberry PI*, quais eram os endereços *I²C* do módulo GY-521 e do conversor A/D. Como mostrado na Figura 31, se apenas o conversor A/D estiver conectado ao barramento, o endereço *I²C* ocupado é o 0x48. Porém, como será usado junto com o sensor GY-521, o barramento apresenta dois endereços ocupados, provando assim, que o programa conseguirá diferenciar a leitura dos respectivos dispositivos.

Figura 31 – Endereços *I²C* do módulo GY-521 e do ADS1115 no barramento

```

pi@raspberrypi: ~
Arquivo Editar Abas Ajuda
pi@raspberrypi:~$ i2cdetect -y 1
 0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00: -- -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- 48 -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- -- -- -- -- -- -- --
pi@raspberrypi:~$ i2cdetect -y 1
 0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00: -- -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- 48 -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- 68 -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- -- -- -- -- -- -- --
pi@raspberrypi:~$
  
```

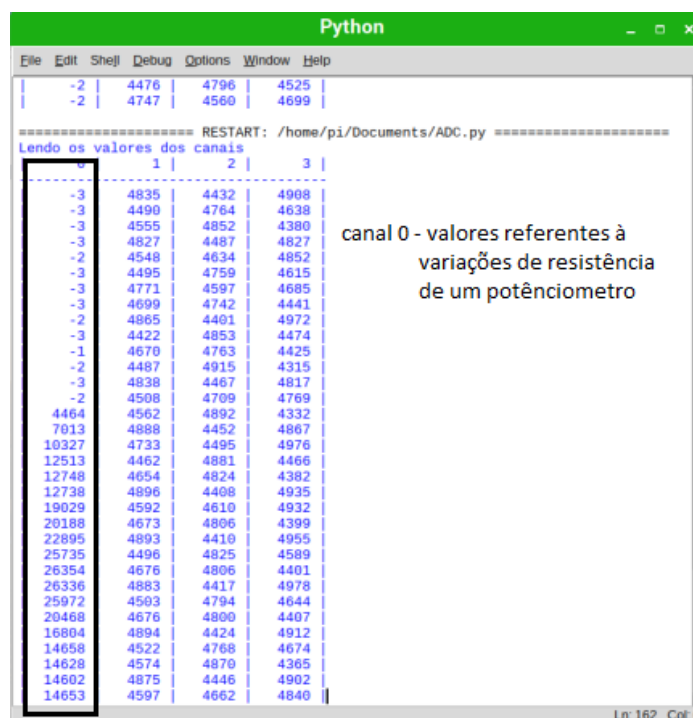
Fonte: Autor (2018)

Ao executar a programação do conversor A/D, uma janela se abre para exibir os valores que estão sendo lidos em cada canal. Esses dados são mostrados na forma de uma tabela de quatro colunas, uma para cada canal. O número de linhas pode ser definido na programação de acordo com o número de amostras que se deseja ler, ou, no caso de leitura em tempo real, não é definido, pois a leitura é realizada continuamente até que se interrompa a execução do programa.

Na Figura 32 pode-se observar a janela de visualização dos dados referentes a cada um dos quatro canais. Os canais foram ligados individualmente a um potenciômetro para visualização e verificação de seu correto funcionamento quanto à variação de saída digital em relação à entrada analógica.

O cabeçalho na primeira linha mostra o número do canal. O valor lido pelos canais é um número que varia de – 32768 a 32767. Um valor de zero, significa que o sinal está no nível de referência, 32767 significa que o sinal é igual ou maior que o valor máximo de tensão para o ganho atual (4,096V por padrão) e -32768 significa que é uma tensão negativa, abaixo da tensão de referência [44].

Figura 32 – Visualização dos canais do conversor A/D



canal 0 - valores referentes à variações de resistência de um potenciômetro	1	2	3
-3	4835	4432	4908
-3	4490	4764	4638
-3	4555	4852	4380
-3	4827	4487	4827
-2	4548	4634	4852
-3	4495	4759	4615
-3	4771	4597	4685
-3	4699	4742	4441
-2	4865	4401	4972
-3	4422	4853	4474
-1	4670	4763	4425
-2	4487	4915	4315
-3	4838	4467	4817
-2	4508	4709	4769
4464	4562	4892	4332
7013	4888	4452	4867
10327	4733	4495	4976
12513	4462	4881	4466
12748	4654	4824	4382
12738	4896	4408	4935
19029	4592	4610	4932
20188	4673	4806	4399
22895	4893	4410	4955
25735	4496	4825	4589
26354	4676	4806	4401
26336	4883	4417	4978
25972	4503	4794	4644
20468	4676	4800	4407
16804	4894	4424	4912
14658	4522	4768	4674
14628	4574	4870	4365
14602	4875	4446	4902
14653	4597	4662	4840

Fonte: Autor (2018)

O conversor A/D ADS1115 possui escalas de ganhos, que podem ser configurados via programação. Assim, caso se tenha sensores que fornecem sinais

fracos, pode-se configurar o conversor para que amplifique o sinal [44]. Esse recurso deve ser utilizado com cautela, pois um ganho muito alto pode elevar o sinal além da tensão máxima do conversor, provocando resultados imprecisos, e um ganho muito baixo pode reduzir o sinal a nível de ruído, impossibilitando a leitura [45].

Para visualizar a conversão de sinais reais de sensores, ligou-se ao sistema o sensor de corrente ACS712 e um sensor de tensão, nos canais 1 e 3 respectivamente.

A carga utilizada para o teste foi um micro motor DC ligado a uma bateria, suas características são:

Motor:

- Tensão – 9 V
- Corrente – 70 mA
- Velocidade de rotação – 8200 rpm

Bateria:

- Tensão – 9 V
- Modelo 6f22m

Os componentes utilizados são mostrados na Figura 33.

Para a realização do teste, a motor foi ligado à bateria e fechou-se o circuito com o sensor de corrente entre o motor e a bateria. Em paralelo com os terminais do motor foi ligado o sensor de tensão. As saídas dos sensores de corrente e tensão foram ligadas aos canais 1 e 3 do conversor.

A corrente que se esperava obter pela medição era a de 70mA, pois esta corrente é justamente a corrente do motor quando este está sem carga no seu eixo. E a tensão que se esperava obter depois da conversão era a própria tensão de 9 V.

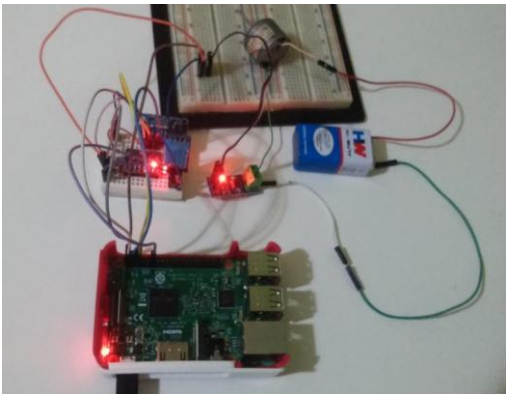
Figura 33 – Componentes utilizados nos testes do conversor A/D



Fonte: Autor (2018)

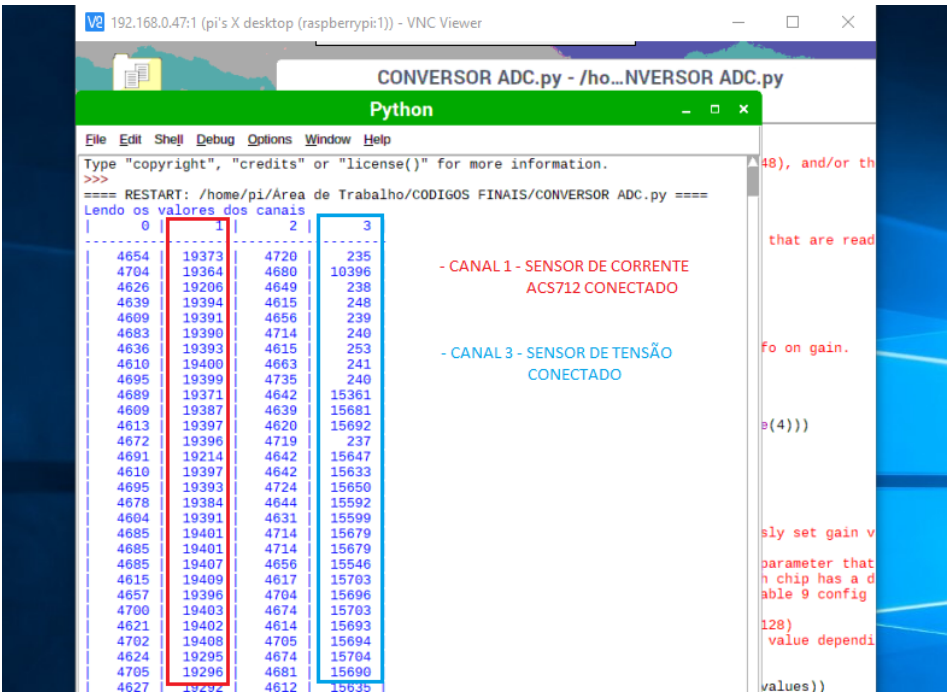
O circuito montado é mostrado na Figura 34 e a leitura dos canais do conversor na Figura 35.

Figura 34 – Leitura dos sensores analógicos através do Conversor A/D



Fonte: Autor (2018)

Figura 35 – Leitura do conversor A/D



Fonte: Autor (2018)

Os dados mostrados na Figura 34 são os valores digitais fornecidos por cada canal. Depois de esperar que os valores se estabilizem, os dados dos canais 1 e 3

foram manipulados via programação para fornecerem os valores reais de tensão e corrente lidos, e os valores obtidos foram: para a corrente obteve-se predominantemente os valores de 64 e 67 mA. E para a tensão o valor predominante lido foi de 8,6 V. Observa-se que os resultados ficaram muito próximos do esperado, porém, não foram exatos. A leitura dos canais se estabilizava em torno de um valor, mas ainda assim, as vezes variava um pouco. Isso pode ter ocorrido devido ao fato de não ter sido confeccionado placas de circuito impresso para o circuito do teste, o que tornaria as conexões mais estáveis e confiáveis. Também pode ser devido às conversões feitas, que inseriram erros devido aos arredondamentos feitos.

O conversor A/D ADS1115 possui quatro endereços I^2C disponíveis programáveis. Dessa forma, pode-se ter até quatro desses conversores ligados ao barramento I^2C do Raspberry, fornecendo assim 16 entradas para sensores analógicos, que seriam acessados por multiplexação da leitura do barramento.

5 ARMAZENAMENTO DE DADOS

Os dados de leitura mostrados anteriormente estavam sendo exibidos à medida que iam sendo lidos pelos sensores, mas apesar da importância de ver essa dinâmica em tempo real, em diversas aplicações necessita-se dos dados para uma análise posterior, para fazer as conversões necessárias, plotar gráficos e manipular os dados para conseguir outros. Dessa forma, é necessário o armazenamento desses dos valores lidos.

Neste trabalho foi feita a leitura de sensores digitais, diretamente ligados ao *Raspberry PI*, leitura de sensores analógicos através do conversor A/D, onde todos os sensores a ele conectados são lidos simultaneamente através do protocolo I^2C , e a leitura do módulo GY-521 também pelo protocolo I^2C . Assim, optou-se por fazer a gravação dos dados individualmente no programa, ao invés de uma gravação única para o programa geral, uma vez que a leitura de cada sensor ocorre em tempos diferentes devido a características da ação de cada um e que a *Raspberry PI* consegue realizar a execução de vários programas simultaneamente. Dessa forma, sensores que capturam muitas amostras por minuto, como o acelerômetro, por exemplo, não será prejudicado com a perda de informação, devido à ação de

sensores mais lentos como o DHT11, que precisa de pelo menos 5 segundos para garantir a estabilidade da temperatura que está lendo.

Para que essa solução fosse efetuada, foi criado um arquivo *csv* (*Comma-Separated Values*) para que, a cada leitura realizada pelos sensores, o valor lido fosse registrado no arquivo.

A escolha de criação de arquivos e mais especificamente de arquivo *csv* pra registrar os dados se deveu principalmente ao fato de a linguagem *Python* de programação ser muito eficaz e de simples manipulação quando se trata de criação e gravação de arquivos. Por ser uma linguagem de fácil entendimento, bastante intuitiva e pelo arquivo *csv* apresentar um formato que facilita a organização dos dados e também por ser um arquivo que pode mais facilmente ser importado entre servidores, a solução se mostrou simples e eficaz.

Enquanto em muitas linguagens de programação, a manipulação de arquivos é trabalhosa, necessitando da instalação de bibliotecas específicas e longos trechos de programação, na *Python* só são necessários aprender a criar um arquivo, abrir o arquivo, habilitar leitura ou escrita e fechar o arquivo, sendo que todas essas funções são feitas, na maioria das vezes, com um comando apenas de programação e possuem poucas funções que os manipulam.

Assim, a lógica seguida foi: criar um arquivo *csv*, habilitar a sua escrita (referente a cada sensor) e fechar o arquivo *csv* quando o programa parar com a leitura. As funções de manipulação dos arquivos em *Python* usadas nesse caso foram:

```
file = open ("Sensor1.csv", "a"), file.write(), file.read(),file.close()
```

A primeira função cria um arquivo onde serão gravados dados de determinado sensor, e através de *file.write()*, *file.read()* e *file.close()* manipula-se esse arquivo dentro da programação.

Os arquivos ficaram salvos na Raspberry, mas podem ser enviados para bancos de dados com maior memória.

6 RELEVÂNCIA DOS SENSORES UTILIZADOS NESTE TRABALHO PARA O ALCANCE DOS OBJETIVOS NO PROJETO DO VEÍCULO ELÉTRICO SOLAR

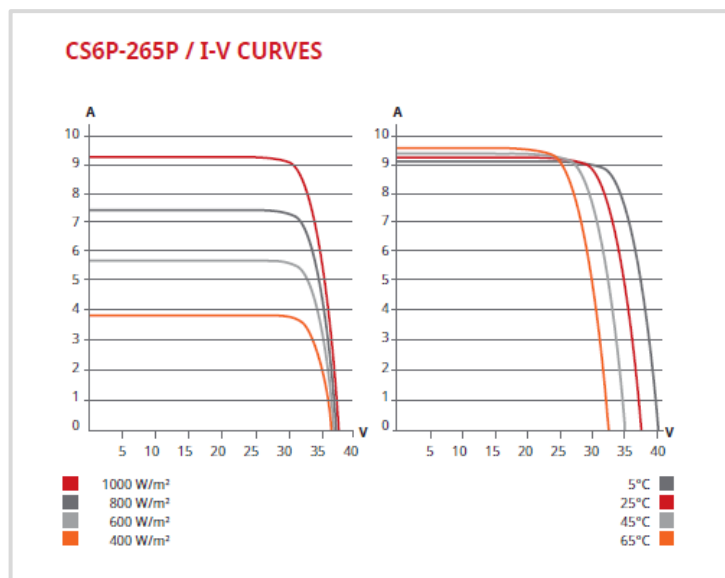
6.1 Temperatura e umidade na geração de energia solar fotovoltaica

A fim de melhor gerenciar a energia gerada pelo painel fotovoltaico e de também prolongar sua vida útil, informações como a temperatura do painel e a umidade do ar do ambiente é fundamental. Depois da irradiância, a temperatura é a variável mais estudada na influência do desempenho dos painéis solares.

A potência máxima em Watts fornecida pela ficha técnica dos painéis baseia-se na produção de energia na chamada “Condições Padrão de Teste”, específicas de laboratório e que não representam as condições reais do Brasil [46]. Nessas condições de teste, a placa é submetida a uma irradiância de 1000W/m^2 a temperatura de $25\text{ }^{\circ}\text{C}$ [46]. Assim, a potência nominal especificada está vinculada a essas condições e o painel, ao ser submetido a uma maior temperatura perderá eficiência.

Na prática, um painel instalado pode ficar até 20°C mais quente do que a temperatura do dia, ou seja, em um dia em que a temperatura ambiente for de 25°C , o painel poderá chegar a 45°C . Dessa forma, na prática, dificilmente se consegue obter a potência nominal. A Figura 36 mostra gráficos retirados da ficha técnica da placa CS6P-265P da *Canadian Solar*, que mostra a relação de tensão e corrente com a variação de irradiância e temperatura.

Figura 36 – Relações de tensão e corrente com a variação de irradiância e temperatura



Fonte: [47]

Observa-se que a corrente gerada é altamente afetada pela irradiância, já a tensão está mais relacionada com a temperatura.

Em um projeto onde se busca maximizar a utilização da energia produzida, a perda de eficiência devido à temperatura deve ser considerada, principalmente por estar localizado em uma região de clima tropical, onde as variações de temperatura são intensas. Com o monitoramento da temperatura e os dados técnicos do painel é possível calcular a potência real gerada, pois o coeficiente de temperatura de potência máxima ($P_{m\acute{a}x}$) fornecido diz o quanto de energia é perdida para cada °C (graus Celsius) acima de 25°C.

Usando como exemplo, o $P_{m\acute{a}x}$ da placa solar CS6P-265P da *Canadian Solar* retirado de sua ficha técnica e mostrado na Figura 37, e considerando que a temperatura real atingida pelo painel em um determinado dia foi de 45°C, tem-se que:

$$45^{\circ}\text{C} - 25^{\circ}\text{C} = 20^{\circ}\text{C}, \text{ e } 20 \times 0,41\% (P_{m\acute{a}x}) = 8,2\%$$

Assim, este painel perderá 8,2% de sua potência nominal.

Figura 37 – Características de temperatura da placa fotovoltaica CS6P -265P

TEMPERATURE CHARACTERISTICS	
Specification	Data
Temperature Coefficient (Pmax)	-0.41 % / °C
Temperature Coefficient (Voc)	-0.31 % / °C
Temperature Coefficient (Isc)	0.053 % / °C
Nominal Operating Cell Temperature	45±2 °C

Fonte: [50]

Nota-se que a perda de potência é significativa, devendo ser levada em consideração, principalmente por que a energia solar gerada pelas placas já é relativamente baixas.

A umidade relativa do ar também pode influenciar na redução da geração de potência, pois com o aumento da umidade mais partículas de água se encontram suspensas no ar, e a luz, ao atingir essas partículas podem sofrer refração, reflexão ou difração, e dessa forma, a incidência de radiação direta sobre os painéis solares será menor, diminuindo a corrente e afetando assim a potência.

6.2 Acelerômetro e giroscópio na análise da dinâmica do Veículo

Como explicado no capítulo 2, o módulo GY-521, composto por acelerômetro e giroscópio, fornece aceleração e velocidade angular nos três eixos. Os dados do módulo GY-521 podem ser muito úteis, pois análises da variação da aceleração podem dizer se o veículo está se movendo e qual é a sua velocidade, e juntamente com o giroscópio, consegue formar um eficiente sistema de navegação autônoma inercial, proporcionando precisão no posicionamento.

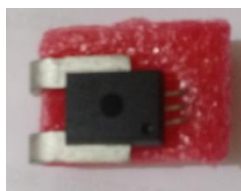
Dessa forma, o módulo GY-521 tem alta aplicabilidade na análise da dinâmica do veículo, pois através da manipulação de seus dados (aceleração e velocidade angular) pode-se obter velocidade e posição, que relacionadas a outras variáveis como energia gerada pelo sistema e energia consumida, irá fornecer um parâmetro para análise da eficiência na gestão da energia no sistema.

6.3 Monitoramento de tensão e corrente

Para a gestão da energia no sistema, será primordial o monitoramento do fluxo de potência no mesmo, dessa forma, sensores de tensão e corrente deverão ser estrategicamente utilizados, como por exemplo, para monitorar o painel solar e o motor.

O sensor de corrente ACS712 não poderá ser usado para a leitura de corrente drenada pelo motor, pois esta ultrapassará as especificações do ACS712. Para essa função foi adquirido o sensor mostrado na Figura 38, o ACS758, que também é um sensor de efeito *Hall*, porém mais robusto, suportando correntes de até 100 A.

Figura 38 – Sensor de corrente ACS758 a ser utilizado na leitura do motor



Fonte: Autor (2018)

7 CONSIDERAÇÕES FINAIS

Os veículos elétricos que utilizam energia solar, apesar de já ser uma realidade no mundo e de serem projetos sustentáveis em alguns aspectos, possuem ainda muitos desafios a serem vencidos, como por exemplo, o aumento da vida útil das baterias e uma maior autonomia. Assim, projetos que viabilizem estudos a respeito desse assunto devem ser incentivados, principalmente pelo fato de o Brasil ser um país com enorme potencial na área de energia solar devido a sua localização em clima tropical.

O desenvolvimento do presente trabalho consistiu em projetar um sistema de telemetria capaz de coletar dados para analisar as condições de operação do sistema elétrico do projeto descrito no capítulo 3, a fim de monitorar determinadas variáveis para a tomada de decisões que possibilitem um melhor uso das fontes de energia. Sistemas de telemetria são de fundamental importância para a avaliação e

análise de sistemas, pois gera um histórico de dados que permite traçar um perfil dos mesmos. O trabalho aqui desenvolvido, juntamente com os equipamentos gerenciadores de energia utilizados no projeto do veículo elétrico com geração solar fotovoltaica auxiliar, permitirá traçar um perfil do sistema de energia e fazer análises como as descritas no capítulo 7.

No decorrer da fase de pesquisas para o desenvolvimento do projeto, notou-se como algumas questões eram importantes para se ter um sistema de telemetria eficiente e robusto. Dessa forma, a preocupação central do trabalho passou a ser criar um sistema que servisse de base, que pudesse dar sustentação da forma mais abrangente possível a futuras implementações ao mesmo. Assim, a escolha da *Raspberry PI 3* modelo B, suas configurações de comunicação com os sensores, tanto analógicos quanto digitais, a escolha da linguagem de programação *Python* e as configurações de acesso remoto constituem um amplo alicerce para um eficiente sistema.

A *Raspberry PI 3* modelo B possibilita o uso dos mais variados tipos de sensores, uma vez que possui 40 pinos digitais e o problema da leitura de sensores analógicos foi resolvido com o uso do conversor A/D ADS1115, que devido ao uso do protocolo *I²C*, pode monitorar até 16 sensores analógicos, devendo-se atentar contudo para a limitação de corrente. Assim, mesmo sem imediatas modificações, o sistema proposto já está apto a agregar mais uma variedade de sensores, como por exemplo, sensores de irradiância, que juntamente com o DTH11 possibilitará uma análise mais completa da influência do clima local na geração de energia.

Assim, trabalhos futuros, com o objetivo de complementá-lo, devem ser focados principalmente em formas de visualização dos dados e de armazenamento, pois em longo prazo, é recomendado que se tenha um banco de dados com maior capacidade, fora da placa de gerenciamento do sistema. Também é interessante que se estude uma forma de acesso remoto mais amplo, que esse acesso possa ser feito entre dispositivos que estejam conectados a diferentes redes.

Outro assunto de futuros trabalhos que deve ser mencionado, é a criação de uma plataforma de visualização dos dados em forma de gráficos, que pode inclusive, ser a mesma do banco de dados. Mas enquanto não se desenvolvem essas soluções, que são mais complexas, robustas e demoradas, recomenda-se a possibilidade do uso de bibliotecas próprias para a geração de gráficos que sejam mais urgentes. Nesse contexto, deve ser mencionada a Matplotlib, que é uma

biblioteca de visualização de dados do *Python*, capaz de gerar gráficos de duas dimensões. No decorrer deste trabalho, esta biblioteca foi instalada na *Raspberry PI*, portanto, mais adiante deve ser explorada, pois esta é capaz de ler os arquivos csv e ir gerando gráficos em tempo real, atualizando à medida que o csv é atualizado.

REFERÊNCIAS

- [1] HENRIQUE, Gustavo. **Carro solar Sono Sion deve chegar em breve à produção em série.** Disponível em:< <https://www.kbb.com.br/detalhes-noticia/sono-sion-carro-solar/?ID=1382>>. Acesso em: 20 ago. 2018.

- [2] IZO, Alexandre. **Frota mundial de carros elétricos crescem 55% no primeiro semestre de 2018.** Disponível em:
<<https://revistaautoesporte.globo.com/Noticias/noticia/2018/08/frota-mundial-de-carros-eletricos-cresce-55-em-um-ano.html> > . Acesso em: 05 set. 2018

- [3] JARDIM, Guilherme. **China testa estrada inteligente que poderá recarregar carros elétricos.** Disponível em:
<<https://carroseletricos.blog.br/2018/04/13/china-testa-estrada-inteligente-que-podera-recarregar-carros-eletricos/>> . Acesso em: 14 mai. 2018

- [4] ROSA, Mayra. **Carros elétricos no Brasil: Realidade ou sonho.** Disponível em : <<http://ciclovivo.com.br/arq-urb/mobilidade/carros-eletricos-no-brasil-realidade-ou-sonho/> > . Acesso em: 07 set. 2018

- [5] J.C, Odair. **Vantagens e Desvantagens dos veículos elétricos.** Disponível em : <<http://inergiae.com.br/site/vantagens-e-desvantagens-do-veiculo-eletrico/> > . Acesso em: 02 jul. 2018

- [6] ELIAS, Oureste. **Veículo elétrico com geração solar fotovoltaica auxiliar: o gerenciamento de uma microrrede móvel.** Projeto de pesquisa proposto e aprovado ao Programa Primeiros Projetos (PPP) da FAPES.

- [7] SIGNIFICADOS. **Significado de Telemetria.** Disponível em:
<<https://www.significados.com.br/telemetria/>> . Acesso em: 09 mai. 2018

- [8] DICIONARIO. **Telemetria.** Disponível em:< <https://educalingo.com/pt/dic-pt/telemetria>>. Acesso em: 09 mai. 2018

- [9] NICOLI, Alessandro. **Telemetria e conceitos relacionados:** uma visão geral dos sistemas de telemetria com ênfase em aplicações aeroespaciais. São José dos Campos, SP, 2004.

- [10] TONDOLO, Sandy; SAUSEN, Paulo; DE CAMPO, Maurício. **Desenvolvimento de um sistema de aquisição de grandezas elétricas de baixo custo para acompanhamento em tempo real das condições de operação**. Relatório técnico-científico apresentado no XXIV. 2016

- [11] DALL AGNOL, Marcel. **Projeto de disciplina para o aprendizado em Linux e Raspberry PI**. 121f. Projeto de Graduação – Engenharia Mecatrônica, Universidade de São Paulo, Escola de Engenharia de São Carlos, 2014

- [12] MACHADO MAIA, Celso. **Sistema para aquisição de sinais de tensão e corrente utilizando a plataforma Beaglebone Black**. 92 f. Dissertação de Mestrado – Engenharia Elétrica, João Pessoa, 2015.

- [13] FILIPEFLOP. **Conversor analógico 4 canais ADS1115**. Disponível em: <<https://www.filipeflop.com/produto/conversor-analogico-digital-4-canais-ads1115/>> . Acesso em: 09 mai. 2018

- [14] CURTO CIRCUITO. **Conversor Analógico/Digital I²C – 16 bits – ADS1115**. Disponível em:< <https://www.curtocircuito.com.br/conversor-analogico-digital-i2c-16-bits-ads1115.html>> . Acesso em: 09 mai. 2018

- [15] NEVES, Saulo. **Tutorial para criar um sistema de aquisição de dados remoto, alimentado por células fotovoltaicas**. Universidade Federal de Ouro Preto – Departamento de Engenharia de Controle e Automação. Disponível em: <http://professor.ufop.br/sites/default/files/adrielle/files/tutorial_para_sistema_tel_emetrico_sustentavel.pdf> . Acesso em: 05 jun. 2018

- [16] DOS REIS, Fabio. **O que são Sensores**. Disponível em: <<http://www.bosontreinamentos.com.br/electronica/curso-de-eletronica/o-que-sao-sensores/>> . Acesso em: 05 jun. 2018

- [17] USINAINFO. **Sensor de corrente ACS758 de efeito Hall**. Disponível em : <<https://www.usinainfo.com.br/sensor-de-corrente-arduino/sensor-de-corrente-ac-dc-100a-efeito-hall-ac758--4495.html>> . Acesso em: 05 jun. 2018

- [18] FUTURE E ELECTRONIC. **Current Sensor ACS758 – 100 A**. Disponível em: <<https://store.fut-electronics.com/products/current-sensor-ac758-ac-or-dc-100a>> . Acesso em: 05 jun. 2018

- [19] GRUP ELEKTRONIK. **ACS758 Current Sensor**. Disponível em: <<https://www.grupelektronik.com/urun/acs758>> . Acesso em: 05 jun. 2018

- [20] USINAINFO. **Sensor de efeito Hall ACS712**. Disponível em:
<<https://www.usinainfo.com.br/sensor-de-corrente-arduino/sensor-de-corrente-ac-dc-30a-efeito-hall-ac712--2952.html> > . Acesso em: 06 jun. 2018

- [21] FILIPEFLOP. **Sensor de corrente ACS712 30 A**. Disponível em :
<<https://www.filipeflop.com/produto/sensor-de-corrente-ac712-30a-a-30a/> > . Acesso em: 06 jun. 2018

- [22] MASTER WALKER. **Sensor de tensão DC 0-25 V**. Disponível em:
<<https://www.masterwalkershop.com.br/sensor-de-tensao-dc-0-25v> > . Acesso em: 06 jun. 2018

- [23] FILGUEIRA, Tomaz. **Telemetria de um veículo Baja SAE através de rede CAN**. 49 f. Dissertação – Engenharia Mecatrônica, Universidade Federal do Rio Grande do Norte, Natal, 2016.

- [24] USINAINFO. **Módulo sensor de umidade e temperatura DHT11**. Disponível em < <https://www.usinainfo.com.br/sensor-de-umidade-arduino/modulo-sensor-de-umidade-e-temperatura-dht11-jumpers-2307.html> > . Acesso em : 04 jun. 2018

- [25] I2CELECTRONICS. **Sensor de temperatura y humedad DHT11**. Disponível em: < <http://i2celectronica.com/temperatura-y-humedad/32-sensor-de-temperatura-y-humedad-dht11.html> > . Acesso em: 20 jun. 2018

- [26] VIDADESILÍCIO. **Módulo GY-521 Acelerômetro e Giroscópio**. Disponível em : <<https://www.vidadesilicio.com.br/modulo-gy-521-acelerometro-giroscopio>> . Acesso em: 20 jun. 2018.

- [27] LIMA, Isabelle. **Aprenda a utilizar o acelerômetro**. Disponível em:
<<http://autocorerobotica.blog.br/aprenda-a-utilizar-o-acelerometro-mpu6050/>> . Acesso em: 20 jun. 2018

- [28] FILIPEFLOP. **Acelerômetro e Giroscópio 3 Eixos**. Disponível em :
<<https://www.filipeflop.com/produto/acelerometro-e-giroscopio-3-eixos-6-dof-mpu-6050/>> . Acesso em: 20 jun. 2018

- [29] BLOG. **Acelerômetro com Arduino**. Disponível em:
<<https://www.filipeflop.com/blog/tutorial-acelerometro-mpu6050-arduino/>> . Acesso em: 20 jun. 2018

- [30] ELEMENTS. **GY-521 3 axis analog gyro sensor** . Disponível em:
<<https://www.elementsonline.com/mpu6050-gy-521-3-axis-analog-gyro-sensors-accelerometer-module>>. Acesso em: 20 jun. 2018

- [31] ARAUJO, Thayron. **Raspberry PI B+: Introdução a porta GPIO**. Disponível em : < <http://blog.fazedores.com/raspberry-pi-b-introducao-porta-gpio/>> . Acesso em: 03 jun. 2018

- [32] BLOGGER-BROTHERS. **Raspberry PI NodeJS GPIO Control: rpi-gpio**. Disponível em:< <https://bloggerbrothers.com/2017/01/14/raspberry-pi3-nodejs-gpio-control-rpi-gpio/>> .Acesso em: 03 jun. 2018

- [33] ROBOCORE. **Comparação entre protocolos de comunicação serial**. Disponível em: < <https://www.robocore.net/tutoriais/comparacao-entre-protocolos-de-comunicacao-serial.html> > . Acesso em: 08 jun. 2018.

- [34] RODRIGUES, Valdinei. **I2C Protocolo de Comunicação**. Disponível em:
<<http://www.arduino.br.com/arduino/i2c-protocolo-de-comunicacao/>> . Acesso em: 08 jun.2018.

- [35] BUILDBOT. **Interfaces de comunicação OneWire, SPI e I²C**. Disponível em:
<<http://buildbot.com.br/blog/interfaces-de-comunicacao-onewire-spi-i2c/>> . Acesso em: 08 jun. 2018.

- [36] MECCOMELETRONICA. **I²C PROTOCOLO DE COMUNICAÇÃO**. Disponível em: < <http://www.meccomeletronica.com/site/data/uploads/i2c-protocolo-de-comunicacao.pdf> > . Acesso em: 08 jun. 2018.

- [37] ARDUINOBR. **I²C Protocolo de Comunicação**. Disponível em:
<<http://www.arduino.br.com/arduino/i2c-protocolo-de-comunicacao/>> . Acesso em: 08 jun. 2018.

- [38] GRUPO PET. **Tutorial de Introdução ao Python**. Programa de Educação Tutorial – Universidade Federal Fluminense – Engenharia de Telecomunicações, Niterói, 2009. Disponível em :
<http://www.telecom.uff.br/pet/petws/downloads/tutoriais/python/tut_python_2k100127.pdf> .Acesso em: 13 jun. 2018

- [39] TECHTUDO. **Faça o download do PuTTY e crie conexões seguras entre servidores.** Disponível em: < <https://www.techtudo.com.br/tudo-sobre/putty.html> > . Acesso em: 13 jun. 2018.
- [40] EST.UFPR. **Informações gerais sobre o uso do VNC.** Disponível em: <<http://www.est.ufpr.br/dicas/node18.html> > . Acesso em: 27 jun. 2018.
- [41] PTCOMPUTADOR. **Instruções TightVNC.** Disponível em: <<http://ptcomputador.com/Networking/other-computer-networking/77516.html>> . Acesso em: 27 jun. 2018
- [42] PPLWARE. **Raspberry PI – Como aceder-lhe remotamente.** Disponível em: <<https://pplware.sapo.pt/high-tech/tutorial-raspberry-pi-como-aceder-lhe-remotamente/>> . Acesso em: 20 jun. 2018.
- [43] USINAINFO. **Como obter valores para os eixos x, y e z do Acelerômetro e Giroscópio.** Disponível em: < <http://blog.usinainfo.com.br/como-obter-valores-para-os-eixos-x-y-e-z-do-acelerometro-e-giroscopio/> > . Acesso em: 20 jun. 2018
- [44] ADAFRUIT. **Raspberry PI Analog to Digital Converters.** Disponível em: <<https://learn.adafruit.com/raspberry-pi-analog-to-digital-converters/ads1015-slash-ads1115>> . Acesso em: 17 julh. 2018
- [45] BLOG – FILIPEFLOP. **Conversor analógico digital ADS1115 e sensor LM35.** Disponível em: < <https://www.filipeflop.com/blog/conversor-analogico-digital-ads1115-sensor-lm35/> > . Acesso em: 17 julh. 2018
- [46] PORTALSOLAR. **Folha de dados do Painel Solar – O que Você precisa saber.** Disponível em: < <https://www.portalsolar.com.br/folha-de-dados-do-painel-solar---o-que-voce-precisa-saber.html> > . Acesso em: 16 ago. 2018
- [47] CANADIAN SOLAR. **Ficha Técnica da Placa CS6P-265P.** Disponível em: <https://www.canadiansolar.com/downloads/datasheets/v5.4/Canadian_Solar-Datasheet-CS6PPSD_SmartDC-v5.4en.pdf> . Acesso em: 16 ago. 2018