



ISEL – INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA
ADEETC – ÁREA DEPARTAMENTAL DE ENGENHARIA DE
ELECTRÓNICA E TELECOMUNICAÇÕES E DE COMPUTADORES

LEIM
LICENCIATURA EM ENGENHARIA INFORMÁTICA E MULTIMÉDIA
UNIDADE CURRICULAR DE PROJETO

Monitorização da Direção e Velocidade do Vento



Pedro Branco Santos (42370)

Orientador

Professor Doutor Porfírio Pena Filipe

setembro, 2020

Resumo

As alterações climáticas previstas a nível global irão ter um impacto significativo em Portugal Continental e nas Regiões Autónomas, exigindo, a médio e longo prazo, medidas de adaptação que permitam a redução do risco, quer a nível global quer a nível individual, na planificação das atividades e do investimento futuro. Neste âmbito considera-se importante para o cidadão, envolvido em atividades quotidianas, ter disponível informação meteorológica em tempo real sobre um local onde pode até não estar presente. Tipicamente as estações meteorológicas existentes no mercado não disponibilizam informação sobre o vento, nem disponibilizam essa informação através da Internet. Sendo assim, é oportuno conceber e implementar um sistema computacional com capacidade de processamento e comunicação que assegure a monitorização da direção e velocidade do vento, privilegiando a versatilidade e o baixo custo. O projeto Monitorização da Direção e Velocidade do Vento é composto por duas partes, nomeadamente: *hardware*, que consiste na ligação de sensores a um microcontrolador e *software*, que processa a aquisição de dados e que disponibiliza acesso via Web.

Abstract

Globally planned climate change will have a significant impact on mainland Portugal and the Autonomous Regions, requiring, in the medium and long term, adaptation measures that allow risk reduction, both globally and individually, planning activities and future investment. In this context it is considered important for the citizen, involved in daily activities, to have available real-time weather information about a place where he/she may not even be present. Typically the weather stations on the market do not provide information about the wind. Therefore, it is opportune to design and implement a computational system with processing and communication capacity that ensures the monitoring of the direction and speed of the wind, privileging versatility and low cost. The project Monitorização da Direção e Velocidade do Vento has two parts: hardware, which consists on the connection between sensors and a microcontroller, and software, which processes the acquisition of data, showing them through the Web.

Agradecimentos

O meu percurso académico teve os seus momentos memoráveis, mas esses momentos foram graças a certas pessoas que foram importantes quer para o desenvolver do curso quer para o projeto. Sendo assim, quero agradecer às pessoas que se seguem.

Quero agradecer à Ana Rita Alves por me ter acompanhado no meu curso do início ao fim.

Quero agradecer ao meu coordenador e engenheiro Porfírio Pena Filipe por me ter ajudado na realização do projeto, por ter discutido a ideia do projeto comigo e por ter disponibilizado o espaço para os testes reais.

Quero agradecer ao engenheiro Vitor Vaz da Silva por me ter emprestado a sua turbina para a realização do projeto.

Quero agradecer ao Sr. Eugénio Pena Filipe por ter colaborado na montagem do anemómetro caseiro.

Quero agradecer a todas os outros colegas que me ajudaram de forma indireta ou direta ao longo do curso.

Índice

Resumo	i
Abstract	iii
Agradecimentos	v
Índice	vii
Lista de Figuras	ix
1 Introdução	1
1.1 Enquadramento	1
1.1.1 Anemómetro de Ventoinha	1
1.1.2 Anemómetro de Fio Quente	2
1.1.3 Cata-Vento	3
1.2 Motivação	3
1.3 Objetivo	4
1.4 Contribuição	4
1.5 Organização do Documento	4
2 Tecnologias	7
2.1 Sensor de Direção	7
2.1.1 Acelerómetro MPU6050	7
2.1.2 Magnetómetro RM3100	9
2.2 Sensor de Velocidade	10
2.2.1 Turbina	10
2.2.2 Anemómetro R3-100	11
2.3 Microcontrolador	11

2.3.1	NodeMCU ESP32	12
2.3.2	NodeMCU ESP8266	13
2.3.3	NodeMCU ESP32-S2	14
2.4	Ambiente de Desenvolvimento	15
2.4.1	Arduino IDE	15
2.4.2	Eclipse	15
2.4.3	IDLE	16
2.5	Comunicação	16
2.5.1	Wi-Fi	16
2.5.2	Bluetooth	18
2.5.3	Wi-Fi LoRa	18
3	Projeto	21
3.1	Casos de Utilização	21
3.2	Arquitetura	22
3.2.1	Sensor de Direção	23
3.2.2	Sensor de Velocidade	24
3.2.3	Microcontrolador	24
3.3	Aplicação Web	24
3.3.1	Servidor	24
3.3.2	Página Principal	25
3.3.3	Página do Administrador	26
3.4	Montagem	27
4	Avaliação Experimental	29
4.1	Calibração	29
4.1.1	Calibração do Acelerómetro	29
4.1.2	Calibração da Turbina	30
4.2	Utilização	30
5	Conclusão	33
5.1	Trabalho Futuro	33
	Bibliografia	35

Lista de Figuras

1.1	Anemómetro de ventoinha (SE818)	2
1.2	Anemómetro de fio quente (8880 Impac)	2
1.3	Cata-vento	3
2.1	Acelerómetro MPU6050	8
2.2	Valores de registo da aceleração	8
2.3	Valores de registo do giroscópio	9
2.4	Tabela de registos do magnetómetro	9
2.5	Montagem para a comunicação SPI	10
2.6	Montagem para a comunicação I2C	11
2.7	Magnetómetro RM3100	12
2.8	Valores da referência da turbina	13
2.9	Turbina	14
2.10	Anemómetro R3-100	14
2.11	NodeMCU ESP32	15
2.12	Diagrama de pinos do NodeMCU ESP32	16
2.13	NodeMCU ESP8266	17
2.14	Diagrama de pinos do NodeMCU ESP8266	17
2.15	NodeMCU ESP32-S2	18
2.16	Diagrama de pinos do NodeMCU ESP32-S2	19
3.1	Casos de utilização	21
3.2	Arquitetura geral do sistema	22
3.3	Arquitetura otimizada do sistema	23
3.4	Página principal	25
3.5	Página de calibração	26
3.6	Montagem do sistema	27

3.7	Esquema do sistema	27
3.8	Anemómetro caseiro	28
3.9	Montagem	28
4.1	Posição inicial	30
4.2	Posição de teste	31

Capítulo 1

Introdução

Neste capítulo é apresentada uma introdução ao projeto intitulado Monitorização da Direção e da Velocidade do Vento descrevendo o enquadramento, a motivação e os respetivos objetivos.

1.1 Enquadramento

Tipicamente as estações meteorológicas existentes no mercado não disponibilizam informação sobre o vento e as que disponibilizam não permitem o acesso e/ou consulta via Internet.

No mercado existem equipamentos capazes de medir a direção ou a velocidade do vento. No entanto, poucos realizam simultaneamente ambas as medições.

Neste contexto, o projeto tem como motivação colmatar esta lacuna. De seguida são descritos exemplos de equipamentos existentes no mercado.

1.1.1 Anemómetro de Ventoinha

Tipicamente os anemómetros de ventoinha medem apenas a velocidade do vento. Esta medição é feita através de uma ventoinha incorporada no equipamento. A ventoinha gera energia consoante a velocidade do vento, ou seja, quanto mais forte for o vento mais energia é gerada. O preço deste tipo de anemómetros varia entre 10 e 100 euros.

O anemómetro ilustrado na figura [1.1](#), tem a opção de escolha de medição entre metros por segundo, com um limiar de 0 a 30, quilómetros por hora, com um limiar de 0 a 90, pés por minuto, com um limiar de 0 a 5860, nós,



Figura 1.1: Anemômetro de ventoinha (SE818)

com um limiar de 0 a 55, e milhas por hora, com um limiar de 0 a 65. O aparelho possui uma precisão de 5% e tem pouca autonomia [1].

1.1.2 Anemômetro de Fio Quente

Os anemômetros de fio quente medem apenas a velocidade do vento. O equipamento possui um sensor de fio quente que permite medir a velocidade do vento especialmente as velocidades baixas. O preço deste tipo de anemômetros varia entre 50 e 700 euros.



Figura 1.2: Anemômetro de fio quente (8880 Impac)

Na figura 1.2 é possível observar um exemplo do equipamento em causa. Para permitir ao sensor calcular a velocidade do vento, é necessário indicar a área do tubo, sensor do lado esquerdo, e a vazão [2] é calculada em metros

cúbicos por segundo ou pés cúbicos por minuto. Possui 5 tipos de leituras diferentes, ou seja, metros por segundo variando de 0,1 a 25, nós, variando entre 0,2 a 48,5, quilómetros por hora, variando entre 0,3 a 90, milhas por hora, variando entre 0,2 a 55,8, e pés por minuto, variando entre 20 a 4925. O aparelho possui uma precisão de 5% e uma entrada USB, *Universal Serial Bus*, que permite o carregamento da sua bateria [3].

1.1.3 Cata-Vento

Os equipamentos tipo cata-vento são caracterizados por possuírem uma cauda, que oscila entre 0° e 360° , que permite medir a direção do vento. Ao contrário dos anemómetros, o cata-vento não mede a velocidade. O preço deste tipo de equipamento varia entre 15 e 50 euros. O preço, em geral, está indexado ao tipo de material no qual são construídos os sensores.



Figura 1.3: Cata-vento

O Cata-Vento da figura 1.3 é constituído por alumínio e por uma interface RS485 [4], que é responsável pela comunicação dos valores medidos. O cata-vento é capaz de medir ângulos entre 0 e 360 graus a uma velocidade mínima do vento de 0,5 metros por segundo, uma precisão de 3 graus e com um tempo de resposta de 5 segundos [5].

1.2 Motivação

Este projeto foi criado por se considerar importante para o cidadão envolvido em atividades quotidianas ter disponível informação meteorológica, designadamente sobre o vento, em tempo real e relativa a um local onde pode até nem estar presente. Sendo assim, é oportuno conceber e implementar um

sistema computacional com capacidade de processamento e comunicação que assegure a monitorização da direção e velocidade do vento, privilegiando a versatilidade e o baixo custo.

1.3 Objetivo

O objetivo do projeto Monitorização da Direção e Velocidade do Vento, que tal como o nome indica, é facilitar a obtenção de informação relativa à direção e à velocidade do vento, em tempo real, tirando partido de tecnologias de baixo custos emergentes no âmbito da IoT, *Internet of Things*.

1.4 Contribuição

A contribuição principal deste projeto é um sistema computacional composto por duas componentes, designadamente: a componente de *hardware* que agrega sensores que permitem monitorizar a direção e velocidade do vento e a componente de *software* que consiste numa aplicação Web que mostra em tempo real a informação obtida.

1.5 Organização do Documento

Este documento está organizado em 5 capítulos.

No capítulo 1 caracteriza-se a introdução onde são descritos os objetivos, o enquadramento e a motivação do projeto proposto.

No capítulo 2 descreve-se as tecnologias onde são comparados os diversos tipos de sensores e microcontroladores que estão realizados com o projeto. O capítulo descreve os tipos de ambientes de desenvolvimento e as tecnologias de comunicação.

No capítulo 3 explica-se o projeto onde é apresentada a arquitetura e a relação entre o microcontrolador e os sensores. No final deste capítulo é ilustrado a montagem relacionada com o projeto.

No capítulo 4 apresenta-se a avaliação experimental, que consiste na calibração dos sensores e na utilização dos sensores num cenário real.

No capítulo 5 são realizadas algumas considerações finais e um possível trabalho futuro.

O repositório de projeto, incluindo o código, está disponível em [\[6\]](#).

Capítulo 2

Tecnologias

Neste capítulo são descritas as tecnologias estudadas no contexto da realização do projeto. São comparados os sensores utilizados neste projeto com outros sensores possíveis que poderiam ter feito parte deste trabalho. Irá ser detalhado o ambiente de desenvolvimento utilizado, os sensores e o tipo de comunicação utilizado para a realização do projeto.

2.1 Sensor de Direção

Para ser possível medir a direção do vento, foi necessário realizar uma pesquisa sobre possíveis tipos de sensores que fossem capazes para tal. De seguida são descritos dois sensores com características distinguidas e funcionalidades diversificadas.

2.1.1 Acelerómetro MPU6050

O acelerómetro com o modelo MPU6050 é capaz de medir a mudança de posição e a variação de velocidade, mas para o pretendido é apenas considerada a capacidade de detetar a mudança de posição.

Na figura [2.1](#) é possível observar os pinos deste acelerómetro. São apenas considerados os 4 primeiros pinos; o VCC, que é o pino de corrente que corresponde a 3.3V, o GND, que corresponde ao sinal de terra, o SCL, que corresponde ao *Serial Clock* e é responsável pelo tempo *Clock* do acelerómetro nos valores da mudança de posição, e o SDA, que corresponde ao *Serial Data* e é responsável pelos valores medidos do sensor. O *Serial Data* é composto por 6 *bytes*, dois para o eixo X, dois para o eixo Y e dois para o eixo Z.

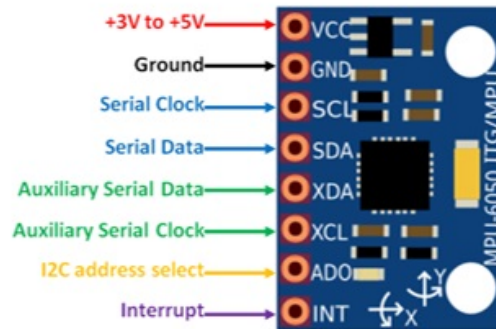


Figura 2.1: Acelerômetro MPU6050

Para serem realizados os cálculos da direção do vento, é necessário entender como é que o acelerômetro efetua a leitura de valores. Em primeiro lugar é iniciada a transmissão I2C, *Inter-Integrated Circuit*, de *bytes* do sensor, onde de seguida é selecionada a posição de registo para a leitura dos valores da componente do acelerômetro do sensor porque será necessário para cálculos mais tarde. Depois da leitura é realizado um pedido ao sensor para requisitar apenas 6 blocos de *bytes*, porque os valores medidos pelo sensor estão dispersos em 6 blocos, como mostra a figura 2.2.

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
3B	59	ACCEL_XOUT[15:8]							
3C	60	ACCEL_XOUT[7:0]							
3D	61	ACCEL_YOUT[15:8]							
3E	62	ACCEL_YOUT[7:0]							
3F	63	ACCEL_ZOUT[15:8]							
40	64	ACCEL_ZOUT[7:0]							

Figura 2.2: Valores de registo da aceleração

Com os valores da aceleração obtidos pelo sensor, é útil calcular os valores de *Roll* e *Pitch*, porque irão ser necessários para o cálculo do ângulo em X e em Y, visto que em Z não tem interesse calcular a grandeza relativa ao *Roll* e *Pitch* [7]. Com os valores de aceleração obtidos, é necessário realizar o mesmo procedimento para os valores de giroscópio, e para tal tem que se escolher a posição de registo de *bytes* do sensor, figura 2.3.

Os valores medidos do giroscópio, valores relacionados com a mudança de posição do sensor, são medidos em graus por segundo, portanto é importante

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
43	67	GYRO_XOUT[15:8]							
44	68	GYRO_XOUT[7:0]							
45	69	GYRO_YOUT[15:8]							
46	70	GYRO_YOUT[7:0]							
47	71	GYRO_ZOUT[15:8]							
48	72	GYRO_ZOUT[7:0]							

Figura 2.3: Valores de registo do giroscópio

apenas obter o valor do ângulo em graus.

Com os ângulos calculados e com os valores da aceleração previamente calculados, é feita uma conta final para calcular o ângulo em X e o ângulo em Y. Nesta operação, o valor do ângulo tem um peso de 0,96 e o valor da aceleração tem um peso de 0,04, que poderão ser ajustados para uma melhor precisão [8].

2.1.2 Magnetómetro RM3100

O magnetómetro RM3100 [9] é reconhecido por ter baixo consumo de bateria, frequência de amostragem grande e grande resolução, o que permite obter valores angulares precisos e bem calculados. Este sensor calcula ângulos usando o campo magnético da terra com um sensor geomagnético para precaver distorções na medição de valores, como por exemplo, o uso de telefones e computadores.

Register Description	Write Address (Hex)	Read Address (Hex)
X Axis Measurement (2)	24	A4
X Axis Measurement (1)	25	A5
X Axis Measurement (0)	26	A6
Y Axis Measurement (2)	27	A7
Y Axis Measurement (1)	28	A8
Y Axis Measurement (0)	29	A9
Z Axis Measurement (2)	2A	AA
Z Axis Measurement (1)	2B	AB
Z Axis Measurement (0)	2C	AC

Figura 2.4: Tabela de registos do magnetómetro

Este sensor consegue eliminar qualquer ruído nas medidas dos campos magnéticos, onde outras tecnologias semelhantes requerem processamento de sinais complexos e caros. Semelhante ao acelerómetro descrito no capítulo 2.1.1, para serem obtidos os valores medidos pelo magnetómetro é necessário requisitar um registo, que corresponde a uma posição no sensor, que irá retornar os valores medidos, utilizando comunicação I2C e SPI, *Serial Peripheral Interface*, dependendo da montagem que se faça. A figura 2.4 mostra a tabela com os registos do magnetómetro.

As figuras 2.5 e 2.6 mostram os diagramas de pinos associados ao magnetómetro RM3100 dependendo da comunicação que se escolha, SPI ou I2C respetivamente, e a figura 2.7 mostra o sensor descrito, RM3100.

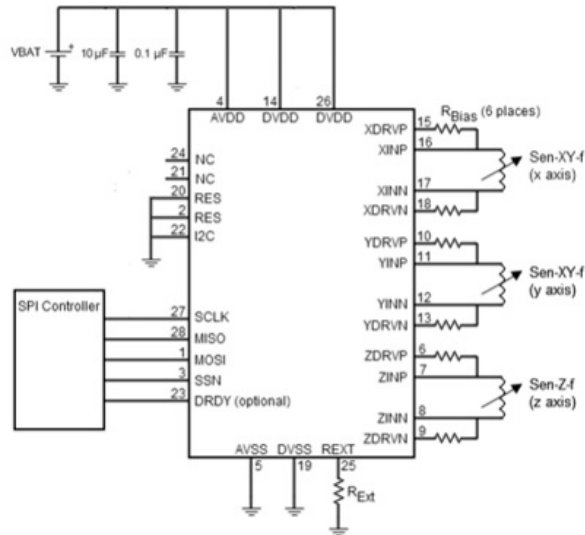


Figura 2.5: Montagem para a comunicação SPI

2.2 Sensor de Velocidade

A medição da velocidade do vento pode ser realizada de diversas formas, mas apenas irão ser descritas as duas mais utilizadas.

2.2.1 Turbina

A turbina [10] é composta por um gerador onde estão associadas 4 pás curvas para facilitar o movimento do vento, gerando um sinal de saída de tensão

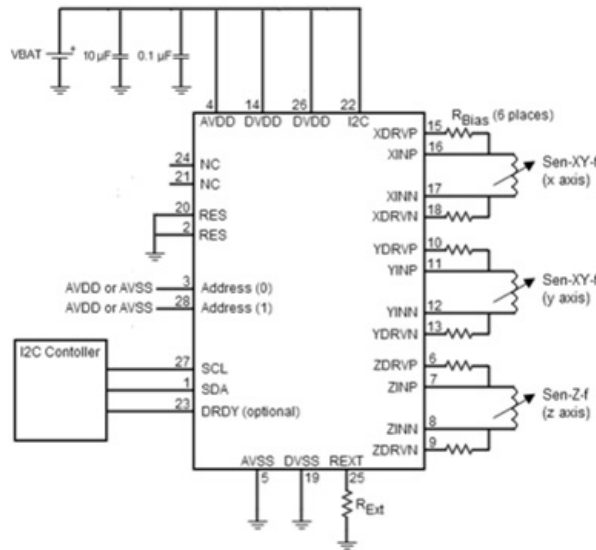


Figura 2.6: Montagem para a comunicação I2C

entre 12 e 24V. Para o projeto apenas são necessárias as pequenas voltagens, entre 0.02 e 2.5V, como a figura 2.8 o mostra.

Na figura 2.8 está representada a equivalência de valores entre volts e metros por segundo e na figura 2.9 é possível observar a turbina descrita.

2.2.2 Anemómetro R3-100

O anemómetro R3-100 [11], fabricado pela *Gill Instruments*, tem uma frequência de amostragem de 100Hz e é composto com 3 eixos ultrasónicos que servem para a realização de cálculos, nomeadamente o método de covariância *Eddy* [12]. O sensor tem a capacidade de medir ventos até 45 metros por segundo com uma precisão de 1%.

Na figura 2.10 está ilustrado o sensor descrito previamente.

2.3 Microcontrolador

Um microcontrolador é um pequeno computador que possui CPU, *Central Processing Unit*, memória RAM, *Random-access memory*, e ROM, *Read-only memory*, e é responsável por realizar uma tarefa específica onde existe um par de input/output na aplicação. De seguida irão ser descritos os microcon-



Figura 2.7: Magnetómetro RM3100

troladores que estão incorporados na gama NodeMCU, que é uma plataforma IoT, *open source* e de baixo custo [13]. Tem a capacidade de transformar a potência recebida, normalmente 5V, para uma potência de 3.3V para um melhor funcionamento do módulo e para não queimar a bateria.

2.3.1 NodeMCU ESP32

O microcontrolador ESP32 [14] possui um conceito simples e barato para o projeto proposto. Este modelo é composto pela linguagem de programação Lua e um dos projetos *open source* mais utilizados chamado SPIFFS, *Serial Peripheral Interface Flash File System*, que será descrito no capítulo 3.3.1. Este microcontrolador permite a ligação Wi-Fi, a ligação Bluetooth e a ligação Wi-Fi LoRa. A figura 2.11 mostra o microcontrolador em causa.

Descrevendo o módulo a que o microcontrolador está agregado, em pormenores técnicos, este possui 18 pinos ADC, *Analog to Digital Converter*, que convertem sinais analógicos em sinais digitais, 2 pinos DAC, *Digital to Analog Converter*, que convertem os sinais digitais em sinais analógicos, 16 pinos de output PVM, que permitem definir os seguintes 4 parâmetros, frequência de sinal, *Duty cycle*, canal PVM e pino GPIO onde se quer colocar o sinal de output, e 10 GPIO de toque, que permitem detetar a variação de algum corpo ou substância que tenha carga elétrica. Também possui 3 formas de comunicação série, SPI, I2C e UART, *Universal Asynchronous Receiver/-Transmitter* [15] ilustrado na figura 2.12.

O SPI é o mais rápido dos 3 protocolos referidos anteriormente e é caracterizado por ser utilizado em microcontroladores que necessitam de comunicar

Wind Rank	Wind Speed (m/s)	Output Voltage(V)
1	0.3-1.6	0.02-0.1
2	1.6-3.4	0.1-0.2
3	3.4-5.5	0.2-0.4
4	5.5-8.0	0.4-0.6
5	8.0-10.8	0.6-0.8
6	10.8-13.9	0.8-1
7	13.9-17.2	1-1.3
8	17.2-20.8	1.3-1.6
9	20.8-24.5	1.6-1.9
10	24.5-28.5	1.9-2.2
11	28.5-32.6	2.2-2.5

Figura 2.8: Valores da referência da turbina

com um ou mais dispositivos de forma rápida e pequenas distâncias [16].

O I2C é o intermédio dos 3 e apenas necessita de dois pinos para realizar esta comunicação, o pino GPIO 21 (SDA) e o pino GPIO 22 (SCL) [17].

O UART é o mais lento dos 3 e a grande vantagem que possui é a sua compatibilidade com microcontroladores simples [18].

2.3.2 NodeMCU ESP8266

O microcontrolador ESP8266 foi o primeiro a ser produzido da gama ESP. Este microcontrolador é fraco comparativamente aos outros porque apresenta uma frequência de *clock* de 80 MHz, não tem processador auxiliar, memória ROM, e memória RTC, *Real Time Clock*, isto é, não consegue guardar dados após o reinício do microcontrolador [19].

Na figura 2.13 representa o módulo onde está incorporado o microcontrolador ESP8266 e na figura 2.14 representa o diagrama de pinos do módulo.

Sendo o primeiro módulo a ser produzido, tem poucos pinos disponíveis. O módulo NodeMCU ESP8266 possui 17 pinos de *input* e *output*, embora



Figura 2.9: Turbina



Figura 2.10: Anemómetro R3-100

não possam ser utilizados todos devido aos números disponíveis, 3 pinos de 3,3 volts, 4 pinos de sinal terra e 4 pinos de data série [20].

2.3.3 NodeMCU ESP32-S2

O microcontrolador ESP32-S2 é o mais recente da gama ESP. O módulo associado a este microcontrolador tem 43 pinos GPIO programáveis, comunicação SPI, *Inter-IC Sound* [21], UART, ADC, DAC. Possui também 14 GPIO de toque programáveis, sendo isto uma solução para dispositivos com ecrãs de toque [22].

A figura 2.15 ilustra o módulo onde está incorporado o microcontrolador ESP32-S2 e a figura 2.16 representa o diagrama de pinos



Figura 2.11: NodeMCU ESP32

2.4 Ambiente de Desenvolvimento

Para ser possível realizar o código associado ao trabalho, teve de ser escolhido o *Integrated Development Environment*, IDE, mais indicado. Para tal, irão ser descritos alguns dos principais IDE disponíveis.

2.4.1 Arduino IDE

O Arduino IDE [23] possui uma linguagem simples, C++, a importação de bibliotecas é intuitiva e é a melhor plataforma para lidar com sensores. A linguagem C++ possui também um conjunto de tipos de variáveis que são consideradas importantes na programação de microcontroladores, como por exemplo o tipo *PROGMEM*, que permite que sejam guardados ficheiros no microcontrolador na memória *Flash*. A memória *Flash* permite ao utilizador fazer o reinício não apagando esses mesmos ficheiros. O Arduino também possui uma grande facilidade na leitura de valores sensoriais e no controlo de estado dos pinos do microcontrolador.

2.4.2 Eclipse

O Eclipse [24] é um IDE versátil, porque possui a capacidade de suportar diversas linguagens de programação, sendo a mais utilizada o JAVA. A importação de bibliotecas não é tão intuitiva como o IDE do Arduino e a instalação do IDE requer mais habilidade. A linguagem JAVA permite um controlo grande em termos de organização das classes e na utilização de Objetos, mas não permite uma utilização fácil no que torna a sensores.

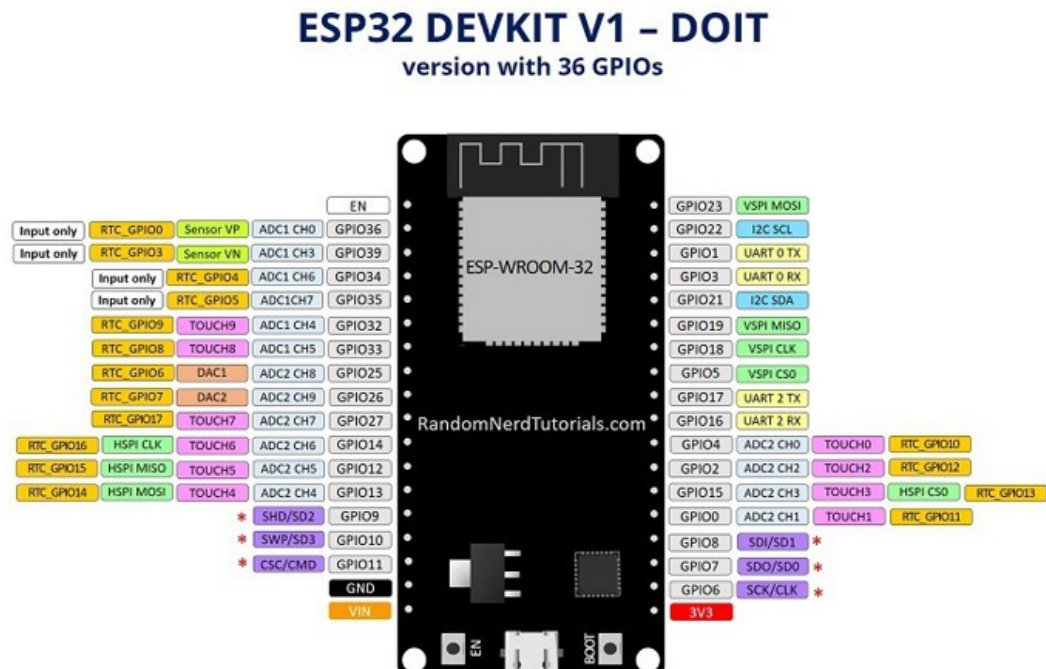


Figura 2.12: Diagrama de pinos do NodeMCU ESP32

2.4.3 IDLE

O IDLE [25] é um IDE que é conhecido por possuir a linguagem de programação Python, tendo como principal utilização em trabalhos que reque-rem processamento de imagem, alteração de ficheiros texto e na construção de jogos, logo não é o IDE mais apropriado para a utilização de valores sensoriais que é uma parte fulcral no projeto realizado.

2.5 Comunicação

Na atualidade existem 2 tipos de comunicação, Wi-Fi e Bluetooth, mas tem existido desenvolvimento num novo tipo de comunicação chamado Wi-Fi LoRa. Neste capítulo são descritas essas 3 comunicações.

2.5.1 Wi-Fi

O Wi-Fi [26] é um modo de comunicação sem fios utilizado por grande parte dos dispositivos para se poderem ligar à Internet, logo é um fator importante

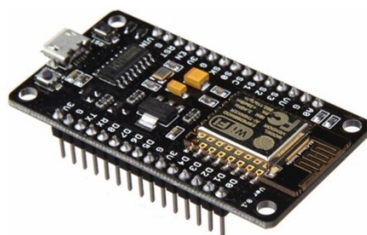


Figura 2.13: NodeMCU ESP8266

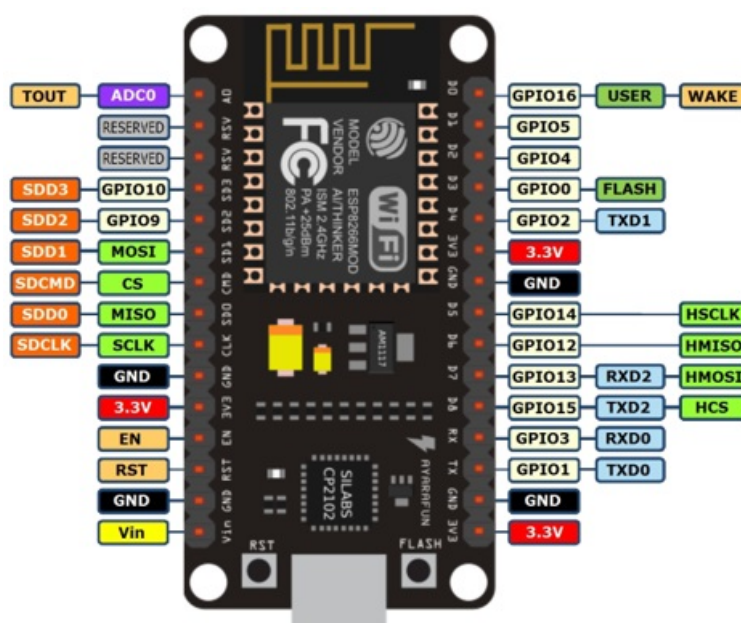


Figura 2.14: Diagrama de pinos do NodeMCU ESP8266

para a realização deste trabalho.

Este tipo de comunicação funciona de uma maneira muito básica, é necessário um *Access Point*, AP, que recebe informação da Internet. Depois converte esta informação para ondas rádio para que dispositivos com o correto adaptador sejam capazes de se ligar ao AP, sendo a área de ondas rádio que o AP produz pequena.

O Wi-Fi é baseado nos padrões da IEEE, *Institute of Electrical and Electronics Engineer*, 802.11 com uma frequência de 2,4 GHz e com débito binário de 2 Mbps.



Figura 2.15: NodeMCU ESP32-S2

2.5.2 Bluetooth

O Bluetooth [27] é um modo de comunicação sem fios que permite trocar ficheiros entre qualquer tipo de dispositivo que possua Bluetooth. Este modo de comunicação não será grande ajuda neste projeto porque apenas é utilizado um microcontrolador que controla quer a página Web, quer os valores medidos dos sensores.

Ao contrário do Wi-Fi, o Bluetooth abrange uma grande área e possui mais potência, mas em contrapartida, requer um alto consumo de energia e um *hardware* mais caro, o oposto dos objetivos pretendidos para este projeto. A frequência do Bluetooth corresponde à mesma frequência que o Wi-Fi.

2.5.3 Wi-Fi LoRa

O Wi-Fi LoRa, Long Range, [28] é um novo tipo de comunicação que é importante e o fator principal da IoT, e é caracterizada por ter grande alcance e por ser uma plataforma sem fios de baixo consumo energético, LPWAN [29]. Tem a capacidade de suportar milhares de mensagens por estação e oferece *Global Positioning System* [30], GPS, beneficiando aplicações com baixo custo energético. Com este tipo de comunicação é possível realizar-se o mesmo que é possível no Wi-Fi, mas com a limitação da largura de banda, até um máximo de 50 quilobits por segundo.

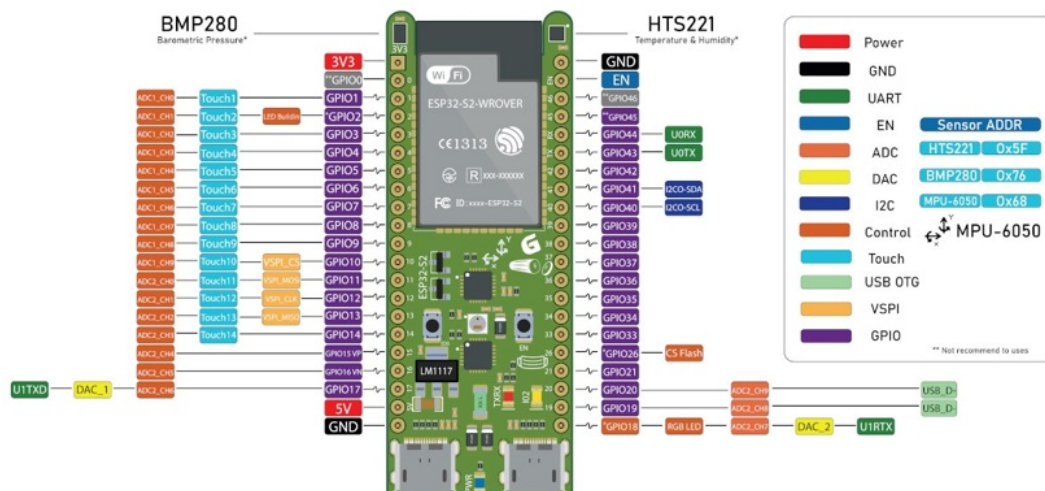


Figura 2.16: Diagrama de pinos do NodeMCU ESP32-S2

Capítulo 3

Projeto

Neste capítulo é realizada a descrição da implementação do projeto.

3.1 Casos de Utilização

Um caso de utilização é um tipo de classificador que está descrito por uma ou mais ações realizadas por atores. Segue-se a figura 3.1 que ilustra os casos de utilização deste projeto.

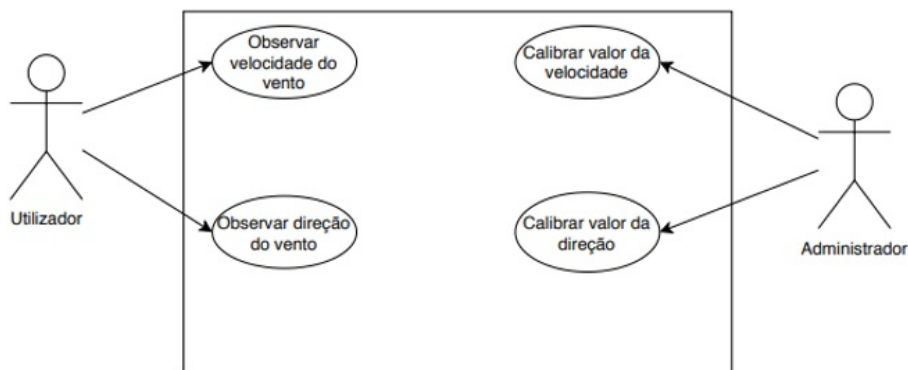


Figura 3.1: Casos de utilização

Como podemos observar existem 4 casos de utilização, que estão divididos pelos dois atores do sistema, o utilizador e o administrador. O utilizador consegue observar a velocidade do vento, valor medido pelo sensor de velocidade, e observar a direção do vento, valor medido pelo sensor de direção. O administrador consegue calibrar o valor da velocidade do vento e consegue

calibrar o valor da direção do vento, que serve para ajustes caso haja alguma falha no sistema.

3.2 Arquitetura

A figura 3.2 esquematiza a arquitetura geral idealizada para o projeto.

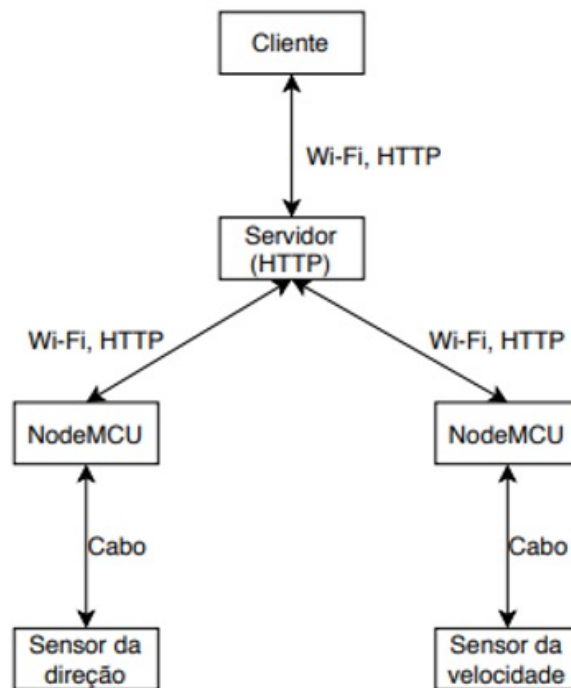


Figura 3.2: Arquitetura geral do sistema

A figura representa a arquitetura geral do sistema onde é constituída por dois módulos NodeMCU com o microprocessador ESP32 para controlar os valores da direção do vento e os valores da velocidade do vento, que depois esses valores iriam ser enviados para um servidor HTTP, *Hypertext Transfer Protocol* [31]. A arquitetura otimizada do sistema está representada na figura 20.

Como se observa na figura 3.3, a arquitetura otimizada do sistema consiste apenas num módulo NodeMCU com o microcontrolador ESP32 que desempenha a função de servidor e que comunica com os dois sensores, sensor da direção do vento e o sensor da velocidade do vento. No microcontrolador

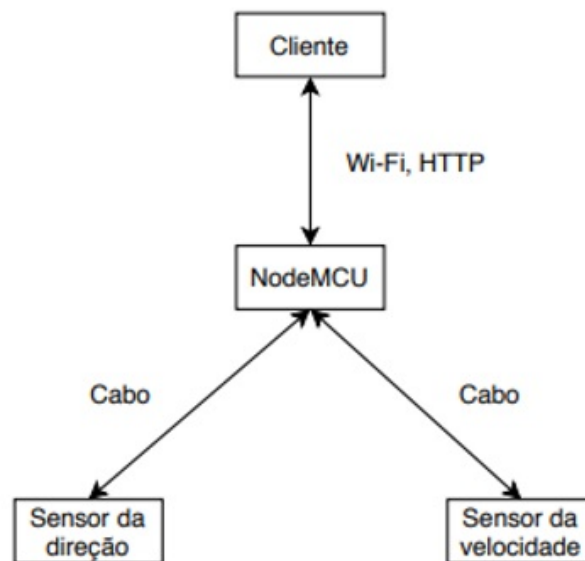


Figura 3.3: Arquitetura otimizada do sistema

são realizadas operações para obter os valores da direção e da velocidade do vento através dos valores brutos dos sensores. Um cliente acede à aplicação Web e consegue observar ambos os valores assim como uma rosa dos ventos que está relacionada com o ângulo calculado.

3.2.1 Sensor de Direção

A solução do sensor escolhida foi o acelerómetro MPU6050. Como descrito no capítulo 2.1, o magnetómetro exige mais conhecimento e é mais caro que o acelerómetro.

O acelerómetro foi o escolhido também, porque tem comunicação I2C de uma forma simples e eficaz, que requer simplesmente a ligação de dois cabos, enquanto que o magnetómetro tem que se alterar a configuração consoante o protocolo que se utilize. Na realização dos cálculos do acelerómetro foi utilizada a biblioteca MPU6050_tockn que utiliza a metodologia de cálculo descrita no capítulo 2.1.1.

Para a simplificação de cálculos, os valores são convertidos entre 0-360^o e entre cada 22,5^o é associada a imagem respetiva a cada intervalo de ângulos.

3.2.2 Sensor de Velocidade

A solução deste tópico foi a turbina, porque além de permitir uma ligação fácil e simples, tem uma tabela associada, figura 2.2, que facilitou os cálculos da velocidade. Para a velocidade é lido o valor de energia gerado pelo motor, entre 0 e 4095, que depois irá ser convertido para um valor de voltagem e que irá ser transformado em metros por segundo, consoante a tabela. Por fim, são convertidos os metros por segundo para quilómetros por hora.

3.2.3 Microcontrolador

A solução que prometia mais resultados e a solução mais barata entre os microcontroladores foi o NodeMCU ESP32. Este microcontrolador permitiu uma utilização simples e direta, baixo custo energético, pelo suporte de Wi-Fi e Bluetooth, apenas foi utilizado o Wi-Fi, e por possuir comunicação I2C.

O NodeMCU ESP32 recolhe os dados dos sensores limpos, sem qualquer processamento de dados associado à recolha, e depois são feitos os cálculos e operações necessárias para os dados poderem ser visualizados. O NodeMCU ESP32 funciona como um servidor, permitindo assim que o utilizador console os dados em tempo real da direção e da velocidade do vento.

3.3 Aplicação Web

Para o desenvolvimento deste projeto, foi criado uma aplicação Web que permite ao utilizador observar a direção e a velocidade atual do vento e ao administrador controlar os valores de ajuste da direção e da velocidade do vento, também controlar o reinício do módulo NodeMCU ESP32 caso seja necessário.

3.3.1 Servidor

Na construção da página, para a observação dos valores da direção e da velocidade do vento, foi necessário primeiro criar um servidor assíncrono através da biblioteca *ESPAsyncWebServer* [32] e da biblioteca *AsyncTCP* [33]. Com as bibliotecas incorporadas no IDE do Arduino, foi então construída a página HTML num *array* de *rawstrings*. Todos os pedidos da página são assíncronos, permitindo que não haja esperas e paragens na página.

Para a visualização dos valores da direção e velocidade do vento, são enviados os valores medidos através de pedidos *Asynchronous JavaScript and XML* [34], AJAX assíncronos e só é mudado o campo respetivo a estes valores, fazendo com que a página fique eficiente e leve em termos de carga computacional da página. Os valores da velocidade e da direção são atualizados de 1 em 1 segundo para manter os valores o mais atualizado possível.

O SPIFFS [35] é uma biblioteca *open source* que merece ser reconhecida, porque foi importante para guardar o ficheiro que contém as CSS, *Cascading Style Sheets*, das páginas, as imagens relativas às direções e aos valores de ajuste. Estes ficheiros são guardados na memória *flash* do NodeMCU ESP32 e permite que estes ficheiros sejam apagados quando se faz um reinício do microcontrolador.

3.3.2 Página Principal

Na página principal, o utilizador consegue observar os valores da direção e da velocidade do vento e também uma rosa dos ventos que varia consoante o ângulo medido atualmente, figura 3.4.



Figura 3.4: Página principal

3.3.3 Página do Administrador

Para ajustar os valores quer de velocidade quer de direção, foi criada uma página Web que permite efetuar esse ajuste. Na figura 3.5 está representada a página do administrador, onde apenas um administrador que tenha conhecimento da página é que pode modificar os valores. Também tem a funcionalidade de fazer um reinício automático caso seja necessário.

Calibração

Valor de ajuste de velocidade 1

Valor de ajuste da direção 0

Reset automático nao

Ajuste

Pedro Branco Santos

Figura 3.5: Página de calibração

Na página existem três campos com valores pré-definidos caso o administrador só queira mudar um dos valores. Os dois campos são do tipo *number* para evitar que o administrador escreva caracteres indesejados e o terceiro campo serve para fazer um reinício do acelerómetro caso seja necessário.

Este ajuste tem a função de manter os sensores bem calibrados caso exista uma falha. Na página de ajuste, o administrador seleciona os valores que quer colocar como ajuste, sendo o valor de ajuste da direção um valor aditivo e um valor de ajuste da velocidade é um valor multiplicativo.

3.4 Montagem

Após a página Web estar acabada, foi criado um anemómetro caseiro através dos sensores e do microcontrolador seleccionados no secção 3.2. O sistema foi montado de acordo com a figura 3.6. A figura 3.7 representada o esquema da montagem do sistema, mostrando a ligação entre os diversos componentes.

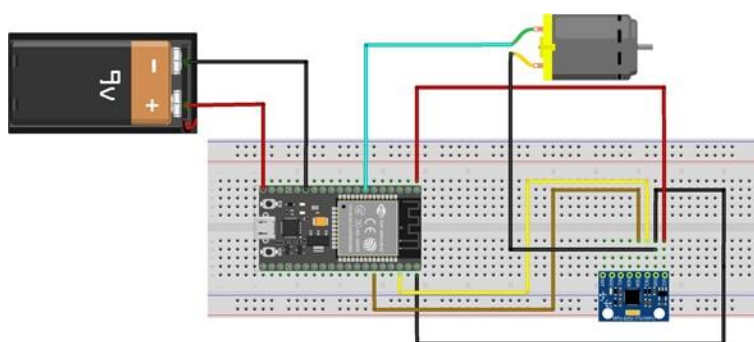


Figura 3.6: Montagem do sistema

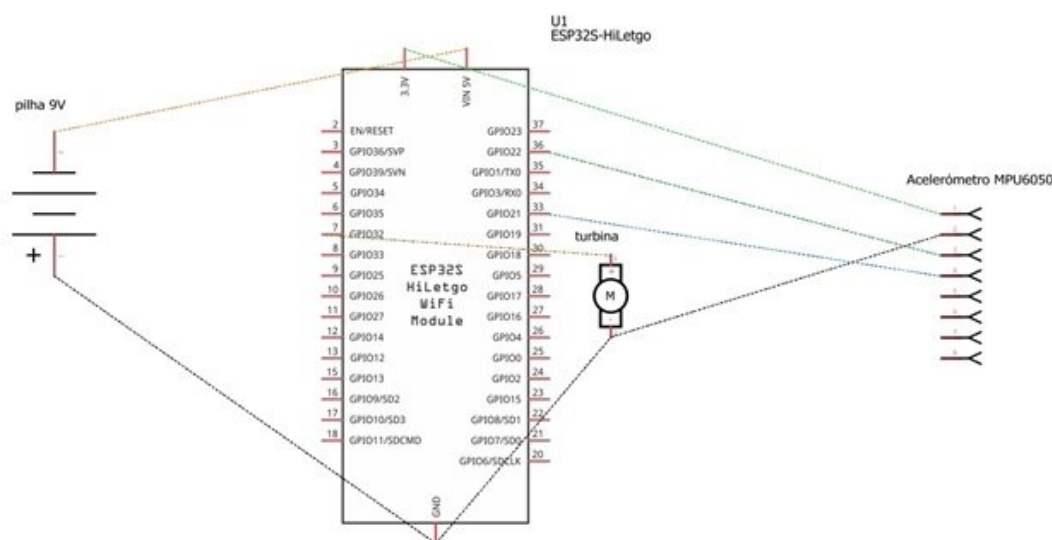


Figura 3.7: Esquema do sistema

De seguida, foi necessário arranjar uma caixa com dimensões pequenas para evitar ao máximo a afetação de valores medidos, isto é, dependendo do tipo de caixa utilizado obtém-se valores diferentes porque peso e o tamanho afetam a direção do vento na cauda do cata-vento montado na figura 3.8.

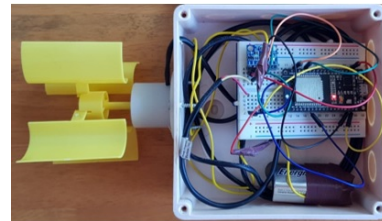


Figura 3.8: Anemómetro caseiro

A caixa utilizada foi uma caixa com 13 cm de largura, 13 cm de largura e 6 cm de altura e pesa 100 gramas. Dentro da caixa está uma *breadboard* com o microcontrolador, Nodemcu ESP32, ligado aos sensores e a uma pilha, permitindo assim ao microcontrolador não depender de corrente elétrica. A figura 3.9a, fora da caixa, e a figura 3.9b, dentro da caixa, ilustram a montagem presente da caixa.



(a) Fora da caixa



(b) Dentro da caixa

Figura 3.9: Montagem

Foi tido em consideração a possibilidade de humidade e a manutenção de posição do acelerómetro e para tal, foi colocada uma borracha impermeável nos arredores da caixa para prevenir a entrada de água ou humidade e colocado um plástico que encobre o volume da caixa interiormente, tentando manter ao máximo a posição do acelerómetro, respetivamente.

Capítulo 4

Avaliação Experimental

Neste capítulo são explicados as calibrações dos sensores e a sua utilização.

4.1 Calibração

A calibração do sistema consiste em parameterizar o acelerómetro para que possa mostrar ângulos em tempo real e ajustar a potência gerada pela turbina para determinar a velocidade do vento.

4.1.1 Calibração do Acelerómetro

A calibração do acelerómetro consiste no seu posicionamento em relação ao ponto cardeal norte e a altitude zero, isto é, o acelerómetro requer um calibre de 3 constantes em relação ao seu posicionamento relativo, para que se possa basear nessa posição relativa para o cálculo de ângulos.

Na página do administrador existe o campo de ajuste dos ângulos. Este campo tem a função de adicionar ou subtrair ao valor medido pelo acelerómetro uma constante, fazendo com que não seja necessário um reinício do NodeMCU ESP32, caso a diferença do valor medido não apresente uma grande diferença ao valor real.

O reinício do módulo NodeMCU ESP32 apenas é necessário quando o acelerómetro não apresenta valores que não têm qualquer relação com os valores reais.

4.1.2 Calibração da Turbina

A turbina gera valores de tensão consoante a velocidade do vento e é necessário verificar se a tensão gerada pela turbina é semelhante à velocidade do vento, que depois irá ser convertida para velocidade do vento considerando os valores já apresentados na figura 2.8. Neste processo terá que ser feito um ajuste na relação entre tensão e velocidade caso os valores medidos e reais não sejam semelhantes.

Na página do administrador tem um campo para o ajuste da velocidade após a conversão de tensão para velocidade. Este campo tem o objetivo de funcionar como um fator multiplicador caso a velocidade mesmo após as conversões não seja semelhante.

4.2 Utilização

Com os sensores calibrados, observou-se que o acelerómetro quando começa a medir os ângulos com direções que variam muito, os seus valores começam a afastar-se do real. As figuras 4.1 e 4.2 mostram o acelerómetro na mesma posição com espaços de tempo diferentes.

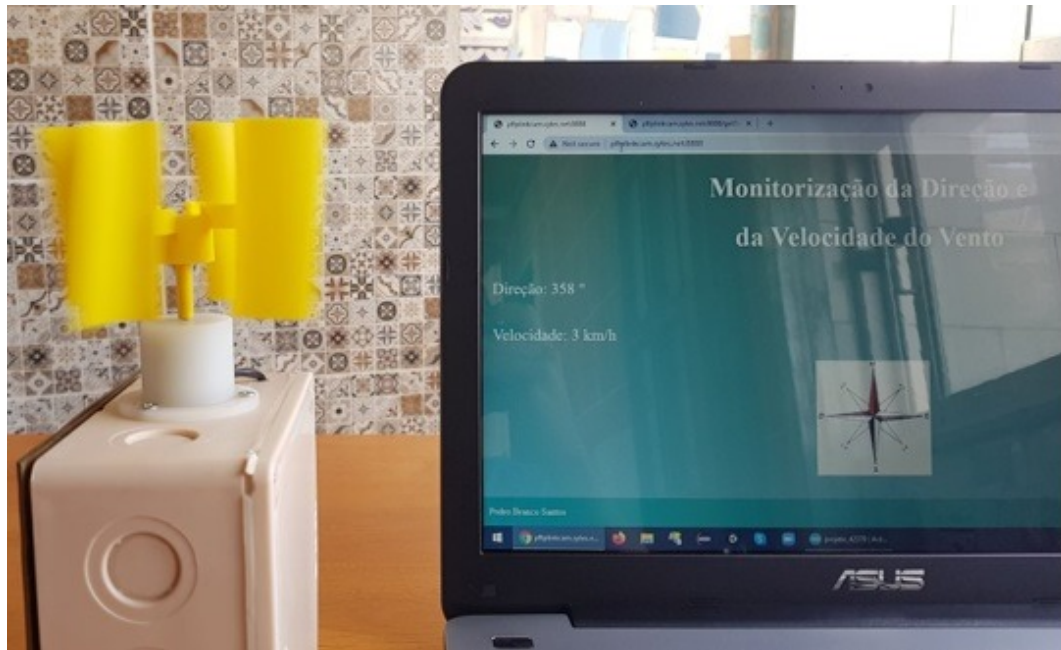


Figura 4.1: Posição inicial



Figura 4.2: Posição de teste

A solução para prevenir isto foi a página de ajuste, que requer um administrador que consiga ajustar os valores. Com esta página espera-se permitir ajustar em conformidade com os valores reais.

Outra solução foi fazer um reinício do módulo NodeMCU ESP32 para o acelerómetro recalibrar as variáveis de calibração.

Relativamente à velocidade medida pela turbina não foi possível determinar se os valores estão semelhantes ou não, visto que não se possui o equipamento necessário para determinar a precisão da turbina.

Capítulo 5

Conclusão

Os objetivos inicialmente propostos foram atingidos com sucesso, estando o sistema a monitorizar a direção e a velocidade do vento com sensores de baixo custo, sendo possível a sua consulta através do *link* **windmeter.sytes.net**.

Para o desenvolvimento do projeto, não foi necessária a aprendizagem de uma nova linguagem de programação, visto que as linguagens utilizadas foram C++, HTML, CSS e JavaScript. No entanto, para a realização deste projeto obrigou a entender o funcionamento do módulo ESP32.

5.1 Trabalho Futuro

Apesar dos objetivos do trabalho terem sido concretizados, existem alguns aspetos a melhorar:

- ⇒ Utilizar uma base de dados para a manter a informação sobre as medições realizadas.
- ⇒ Melhor separação de responsabilidades, para cada sensor ter o seu microcontrolador. No entanto, esta alteração requer equipamento adicional.
- ⇒ Implementar outras alternativas à existente, por exemplo, graficamente.
- ⇒ Melhores questões de segurança e privacidade, melhorando o protocolo de acesso aos sensores ou através de um *login*.

Bibliografia

- [1] “Anemómetro de Ventoinha,” July 2020. https://pt.made-in-china.com/co_bestone-meter/roduct_Digital-Anemometer-Wind-Speed-Meter-Anemograph-BE818-_hrryeirog.html.
- [2] “Vazão,” July 2020. <https://pt.wikipedia.org/wiki/Vaz%C3%A3o>.
- [3] “Anemómetro de Fio Quente,” July 2020. <https://www.impac.com.br/anemometro/anemometro-fio-quente-digital-hot-wire-ip-8880-impac-.html>.
- [4] “Interface RS485,” July 2020. <https://www.lammertbics.nl/comm/info/rs-485#:~:text=RS485%20is%20the%20most%20versatile,nodes%20communicate%20with%20each%20other>.
- [5] “Cata-vento,” July 2020. https://www.banggood.com/pt/Wind-Sensor-Garden-Signal-Output-Aluminum-Alloy-Wind-Direction-Sensor-Wind-Vane-Speed-Measuring-Instrument-p-1624988.html?gmcCountry=PT¤cy=EUR&createTmp=1&utm_source=googleshopping&utm_medium=cpc_bgs&utm_content=lijing&utm_campaign=ssc-pt-all-0316&ad_id=425769145324&gclid=EAIaIQobChMIya-IwP-46wIVVrLVCh1o4w2rEAQYASABEgLba_D_BwE&cur_warehouse=CN.
- [6] “Monitorização da Direção e Velocidade do Vento,” 2020. <https://github.com/pedro-santos423/pfc>.
- [7] “Roll and Pitch,” July 2020. https://www.nxp.com/files-static/sensors/doc/app_note/AN3461.pdf.
- [8] “Acelerómetro MPU6050,” 2020. <https://www.youtube.com/watch?v=UxABxSADZ6U&feature=youtu.be>.

-
- [9] “Magnetómetro RM3100,” 2020. <https://www.pnicorp.com/rm3100/>.
 - [10] “Turbina,” 2020. <https://www.ebay.com/itm/Wind-Speed-Sensor-12V-24V-Voltage-Signal-Output-Anemometer-Fit-for-Outdoor-Use/203015002412>.
 - [11] “Anemómetro R3-100,” 2020. <http://gillinstruments.com/products/anemometer/R3-100.html>.
 - [12] “Covariância Eddy,” 2020. https://en.wikipedia.org/wiki/Eddy_covariance.
 - [13] “NodeMCU,” 2020. <https://en.wikipedia.org/wiki/NodeMCU>.
 - [14] “ESP32,” 2020. <https://en.wikipedia.org/wiki/ESP32>.
 - [15] “Diagrama de pinos do ESP32,” 2020. <https://randomnerdtutorials.com/esp32-pinout-reference-gpios/>.
 - [16] “SPI,” 2020. <https://www.arduino.cc/en/reference/SPI>.
 - [17] “I2C,” 2020. <https://howtomechatronics.com/tutorials/arduino/how-i2c-communication-works-and-how-to-use-it-with-arduino/>.
 - [18] “UART,” 2020. <https://www.arduino.cc/reference/en/language/functions/communication/serial/>.
 - [19] “NodeMCU ESP8266,” July 2020. <https://maker.pro/esp8266/tutorial/a-comparison-of-the-new-esp32-s2-to-the-esp32>.
 - [20] “NodeMCU ESP8266 pinos,” July 2020. <https://iotbytes.wordpress.com/nodemcu-pinout/>.
 - [21] “Inter-IC Sound,” July 2020. <https://en.wikipedia.org/wiki/I%C2%B2S>.
 - [22] “NodeMCU ESP32-S2,” July 2020. <https://www.espressif.com/en/products/socs/esp32-s2>.
 - [23] “Arduino IDE,” July 2020. <https://www.arduino.cc/>.

-
- [24] “Eclipse IDE,” July 2020. <https://www.eclipse.org/>.
 - [25] “IDLE (Python),” July 2020. <https://www.python.org/>.
 - [26] “Wi-Fi,” July 2020. <https://3g.co.uk/guides/what-is-wi-fi-explained-in-simple-terms>.
 - [27] “Bluetooth,” July 2020. <https://www.explainthatstuff.com/howbluetoothworks.html>.
 - [28] “Wi-Fi LoRa,” July 2020. <https://www.semtech.com/lora/what-is-lora>.
 - [29] “LPWAN,” July 2020. <https://en.wikipedia.org/wiki/LPWAN>.
 - [30] “GPS,” July 2020. https://en.wikipedia.org/wiki/Global_Positioning_System.
 - [31] “HTTP,” July 2020. <https://developer.mozilla.org/en-US/docs/Web/HTTP>.
 - [32] “ESPAsyncWebServer,” July 2020. <https://github.com/me-no-dev/ESPAsyncWebServer>.
 - [33] “AsyncTCP,” July 2020. <https://github.com/me-no-dev/AsyncTCP>.
 - [34] “AJAX,” July 2020. https://www.w3schools.com/js/js_ajax_intro.asp.
 - [35] “SPIFFS,” July 2020. <https://tttapa.github.io/ESP8266/Chap11%20-%20SPIFFS.html>.