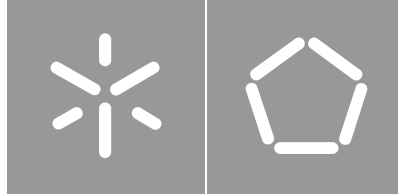


Universidade do Minho

Escola de Engenharia

José Alberto Fernandes Salgado

**Desenvolvimento de uma Estação
Meteorológica Compacta e de Baixo Custo
para Aplicações em Energias Renováveis**



Universidade do Minho

Escola de Engenharia

José Alberto Fernandes Salgado

**Desenvolvimento de uma Estação
Meteorológica Compacta e de Baixo Custo
para Aplicações em Energias Renováveis**

Dissertação de Mestrado
Mestrado Integrado em Engenharia
Eletrónica Industrial e Computadores

Trabalho efetuado sob a orientação do
Professor Doutor José Augusto Afonso
Professor Doutor João Luiz Afonso

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.



Atribuição
CC BY

<https://creativecommons.org/licenses/by/4.0/>

Agradecimentos

Queria agradecer aos meus orientadores, Professor José Augusto Afonso e Professor João Luiz Afonso por todo o apoio e interesse mostrado durante a realização desta dissertação e pela disponibilidade para tirar dúvidas.

Um obrigado ao pessoal do GEPE pelo apoio no material necessário para a realização da dissertação, e também um obrigado ao senhor Carlos Torres pelo apoio no desenvolvimento da placa de circuito impresso.

Um abraço a todos os colegas de curso que que estiveram sempre prontos para ajudar.

Por fim gostaria de agradecer à minha família pelo apoio neste percurso académico, que sem tal não seria possível.

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na elaboração do presente trabalho académico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração.

Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

Resumo

Nos últimos anos cada vez mais são utilizados painéis fotovoltaicos para produção de energia. Estes permitem utilizar a energia proveniente do sol em forma de radiação e transformar em energia elétrica. Muitos destes painéis têm como elemento fundamental na sua composição o silício, podendo este ter várias variações. Dependendo da variação de silício usado, o custo de produção de um painel fotovoltaico e a sua eficiência irá variar.

Nesta dissertação foi desenvolvida uma estação meteorológica capaz de medir diversas variáveis meteorológicas, integrando também três módulos fotovoltaicos de diferentes variações de silício na sua composição, com o objetivo de fazer uma correlação entre as condições meteorológicas e a potência produzida pelos módulos. O sistema desenvolvido permite que os dados medidos pela estação sejam guardados localmente, através de um cartão de memória. A transferência dos dados para o utilizador é feita através da rede sem fios Bluetooth Low Energy (BLE), sendo uma solução em que não são acrescentados custos associados. Os dados transferidos são visualizados através de uma aplicação Android, sendo estes guardados numa base de dados desenvolvida em SQLite. Nesta aplicação podem ser definidas as horas de início e fim de aquisição de dados, bem como o período de amostragem. A aquisição de dados é feita recorrendo a um microcontrolador integrado num módulo com um *transceiver* BLE. Foi concebida e simulada uma solução para a maximização da produção de energia pelos módulos fotovoltaicos, através do uso de um conversor com capacidades de MPPC (Maximum Power Point Control). O sistema é todo alimentado através de uma bateria LiPo (Lithium Polymer), podendo esta ser recarregada através de um módulo fotovoltaico.

O sistema desenvolvido foi testado através da aquisição de dados no exterior em dois dias diferentes. Foi possível a transferência e visualização de todos os dados através da aplicação Android desenvolvida. Os resultados obtidos permitem concluir que o módulo de silício monocristalino apresenta a melhor eficiência de entre as três tecnologias de silício para as condições meteorológicas da localização.

Palavras-chave: Android, Bluetooth Low Energy, Estação meteorológica, Painéis Fotovoltaicos, Energias Renováveis.

Abstract

In recent years more and more photovoltaic panels are used for energy production. These allow the use of energy from the sun in the form of radiation and transform it into electrical energy. Many of these panels have silicon as their fundamental element in their composition, which may have several variations. Depending on the form of silicon used, the cost of producing a photovoltaic panel and its efficiency will vary.

In this dissertation a weather station capable of measuring various weather variables was developed, integrating three photovoltaic modules of different silicon variations in its composition. The developed system allows the data measured by the station to be stored locally using a memory card. Data is transferred to the user device through a Bluetooth Low Energy (BLE) wireless network, which is a solution with no transfer costs associated. The transferred data is viewed through an Android application, and the data is stored in a database developed in SQLite. In this application it can be set the start and end times of data acquisition, as well as the sampling period. Data acquisition is done using a microcontroller integrated in a module with a BLE transceiver. A solution for maximizing the power output of photovoltaic modules was designed and simulated using a converter with maximum power point control (MPPC) capabilities. The entire system is powered by a LiPo (Lithium Polymer) battery, which can be recharged using a photovoltaic module.

The developed system was tested by acquiring outdoor data in two different days. It was possible to transfer and view all data through the developed Android application. It is concluded through the obtained results that the monocrystalline silicon module has the best efficiency among the three silicon technologies for the localization weather conditions.

Keywords: Android, Bluetooth Low Energy, Weather Station, Photovoltaic Panels, Renewable Energy.

Índice de conteúdos

Agradecimentos	i
Resumo	iii
Abstract	iv
Índice de conteúdos	v
Lista de acrónimos e siglas.....	ix
Lista de figuras	xi
Lista de tabelas	1
1. Introdução	1
1.1 Enquadramento e motivação	1
1.2 Objetivos	2
1.3 Estrutura da dissertação	3
2. Estado da arte.....	4
2.1 Estações meteorológicas.....	4
2.2 Aquisição dos dados meteorológicos	5
2.2.1 Velocidade do vento	5
2.2.2 Direção do vento	6
2.2.3 Irradiação solar	6
2.2.4 Temperatura	8
2.3 Painéis fotovoltaicos	9
2.3.1 Visão geral	9
2.3.2 Circuito equivalente.....	11
2.3.3 Impacto do ambiente na produção de energia	12

2.3.4	Maximum Power Point Tracking	14
2.3.5	Circuitos integrados para <i>energy harvesting</i>	15
2.4	Tecnologias de redes sem fios	16
2.4.1	Visão geral	16
2.4.2	Bluetooth Low Energy	18
2.4.2.1	Pilha protocolar	18
2.4.2.2	Formato dos dados do servidor GATT	21
2.4.2.3	Intervalo de conexão	23
2.5	Sistema operativo Android	24
2.5.1	Atividades	24
2.5.2	Fragmentos	27
2.5.3	Serviços.....	27
2.5.4	Android <i>manifest</i>	28
2.6	Trabalho relacionado	28
3.	Desenvolvimento do sistema	32
3.1	Arquitetura do sistema.....	32
3.2	Hardware da estação.....	32
3.2.1	PSoC 4 BLE.....	33
3.2.2	Temperatura e humidade.....	35
3.2.3	Velocidade e direção do vento	36
3.2.4	Irradiação solar.....	41
3.2.5	Medição e maximização da produção de energia	43
3.2.6	Alimentação da estação	48
3.2.7	Cartão de memória e módulo de comunicação.....	50
3.2.8	PCB para maximização da produção de energia	51

3.2.9	Desenho de suporte para impressão 3D.....	51
3.3	Aquisição, armazenamento e transferência de dados	52
3.3.1	Aquisição dos dados dos sensores e módulos fotovoltaicos.....	52
3.3.2	Armazenamento de dados.....	59
3.3.3	Programação da rede BLE	64
3.3.3.1	Configuração do componente BLE.....	64
3.3.3.2	Envio de dados para a base de dados GATT	68
3.3.3.3	Programa de depuração CySmart.....	68
3.4	Desenvolvimento da Aplicação em Android.....	69
3.4.1	Fluxograma da Aplicação	69
3.4.2	Procura e conexão de dispositivos BLE.....	70
3.4.3	Configuração da aquisição de dados	71
3.4.4	Transferência de dados.....	72
3.4.5	Base de dados SQLite.....	74
3.4.6	Visualização de dados.....	75
4.	Resultados e discussão.....	76
4.1	Protótipo desenvolvido.....	76
4.2	Simulação LTSpice	79
4.3	Aquisição de dados.....	86
4.4	Transferência de dados e análise do desempenho.....	88
4.5	Visualização de dados.....	97
4.5.1	Recarregamento da bateria	101
4.5.2	Conclusões sobre a potência produzida.....	102
5.	Conclusões.....	106
5.1	Conclusões.....	106

5.2	Sugestões de trabalho futuro	107
	Referências	109

Lista de acrónimos e siglas

ADC	Analog-to-Digital Converter
API	Application Programming Interface
BLE	Bluetooth Low Energy
CSS	Chirp Spread Spectrum
FTP	File Transfer Protocol
GAP	Generic Access Profile
GATT	Generic Attribute Profile
GPRS	General Packet Radio Service
IDE	Integrated Design Environment
IOT	Internet of Things
ISM	Industrial, Scientific and Medical
LAN	Local Area Network
LPWAN	Low Power Wide Area Network
MPP	Maximum Power Point
MPPC	Maximum Power Point Control
MPPT	Maximum Power Point Tracking
MTU	Maximum Transmission Unit
NTC	Negative Temperature Coefficient
PCB	Printed Circuit Board
PSOC	Programmable System-on-Chip
PV	Photovoltaics
RF	Radio Frequency

RSSI	Received Signal Strength Indication
RTC	Real Time Clock
SDK	Software Development Kit
SIM	Subscriber Identification Module
SRAM	Static Random-Access Memory
SPI	Serial Peripheral Interface
UART	Universal asynchronous receiver-transmitter
UUID	Universally Unique Identifier
WAN	Wide Area Network
WLAN	Wireless Local Area Network
WPAN	Wireless Personal Area Network

Lista de figuras

Figura 1. Estação meteorológica Ambient Weather WS-2902A [7].	4
Figura 2. Sistema de monitorização ENVIROMONITOR, composto por uma estação meteorológica atuando como gateway e múltiplas estações como nodos [8].	5
Figura 3. Descrição do funcionamento de um anemómetro baseado num fio aquecido [10].	6
Figura 4. Espectro da irradiação solar medido por alguns tipos de sensores usados em piranómetros [11].	7
Figura 5. a) Imagem do piranómetro CMP3 b) Diagrama de funcionamento do piranómetro CMP3 [15].	8
Figura 6. Descrição do funcionamento de um termopar.	9
Figura 7. Descrição do efeito fotovoltaico [18].	10
Figura 8. Exemplo de células fotovoltaicas em série e em paralelo e as respetivas tensões e correntes obtidas.	10
Figura 9. Circuito equivalente de uma célula ou painel fotovoltaico.	12
Figura 10. Exemplo de uma curva corrente-tensão (I/V) de um painel fotovoltaico.	12
Figura 11. Potência produzida pelo painel fotovoltaico quando exposto a diferentes condições atmosféricas [19].	13
Figura 12. Exemplo do gráfico I-V de um painel fotovoltaico para diferentes condições meteorológicas.	14
Figura 13. Arquitetura da stack de protocolos do BLE [33].	19
Figura 14. Formato de pacotes para várias camadas da stack do BLE.	20
Figura 15. Visão geral de uma possível arquitetura de uma aplicação BLE [35].	21
Figura 16. Formato e exemplo de um atributo.	22
Figura 17. Formato e exemplo de uma característica.	22
Figura 18. Exemplo do formato de dados de um servidor GATT para um serviço de medição da frequência cardíaca [35].	23
Figura 19. Exemplificação do intervalo de conexão entre dois dispositivos.	24
Figura 20. Mudança de uma atividade para outra através de um botão [37].	25
Figura 21. Ciclo de vida de uma atividade de uma aplicação Android [38].	26

Figura 22. Exemplo da utilização de um fragmento. Neste caso o fragmento hospeda um diálogo para escolher uma data.	27
Figura 23. a) Representação de um escravo da rede, que inclui um painel solar e algumas variáveis a serem medidas b) Representação da comunicação desde os escravos até ao computador, usando um coordenador como intermediário.	29
Figura 24. Arquitetura do sistema implementado em [40].	30
Figura 25. Arquitetura do sistema desenvolvido.....	32
Figura 26. Diagrama de blocos do hardware da estação meteorológica.	33
Figura 27. Kit CY8CKIT-BLE-A 4.2, desenvolvido pela Cypress Semiconductor [45]	34
Figura 28. Descrição do módulo PSoC 4 BLE [45].	35
Figura 29. Sensor de temperatura e humidade DHT22.	35
Figura 30. Protocolo de comunicação entre sensor DHT22 e o microcontrolador.	36
Figura 31. Circuito para medição da velocidade do vento, composto por um interruptor magnético (S1) e uma resistência pull-down (R1).	37
Figura 32. Cata-vento eletrónico e anemómetro utilizados.	Erro! Marcador não definido.
Figura 33. Circuito para deteção da direção do vento. S1 a S8 representa os interruptores magnéticos.	39
Figura 34. Bussola usada para calibrar o cata-vento.....	40
Figura 35. Circuito de acondicionamento usando o fotodiodo BPW34.....	41
Figura 36. Piranómetro PYR-P utilizado na calibração.....	42
Figura 37. Gráfico que relaciona a tensão de saída do circuito e a irradiação solar.	43
Figura 38. Módulos fotovoltaicos utilizados: (1) – Policristalino, (2) – Monocristalino, (3) – Amorfo (dois em série).	44
Figura 39. Circuito desenvolvido com o conversor LTC3129 para maximização da extração de energia dos módulos fotovoltaicos.	46
Figura 40. Esquemático da medição da produção de energia dos três módulos fotovoltaicos.	47
Figura 41. Divisor de tensão para medição da tensão entre a saída do conversor LTC3129 e a carga.....	47
Figura 42. Diagrama de blocos do circuito de alimentação da estação.....	48
Figura 43. Bateria LiPo de 3.7 V com capacidade de 1600 mAh.	48
Figura 44. Módulo para carregamento de baterias LiPo, baseado no circuito integrado TP4056.....	49
Figura 45. Regulador de tensão step-up/step-down S7V8A [54].	49
Figura 46. Sistema de recarregamento da bateria e medição da capacidade atual da bateria.....	50

Figura 47. Cartão de memória e leitor usados na estação.	51
Figura 48. Layout da PCB desenvolvida para medição da produção de energia.....	51
Figura 49. Desenho do suporte do anemómetro e cata-vento eletrónico.....	52
Figura 50. IDE PSoC Creator utilizado para desenvolver o firmware para o módulo PSoC 4 BLE.	53
Figura 51. Configuração do temporizador para um tempo de 100 ms.	54
Figura 52. Configuração do ADC do módulo PSoC 4 BLE.	54
Figura 53. Fluxograma da função que processa os dados obtidos no ADC.	55
Figura 54. Configuração do amplificador operacional no IDE PSoC Creator.....	56
Figura 55. Configuração do contador no IDE PSoC Creator.....	57
Figura 56. Configuração do componente RTC no IDE PSoC Creator.....	58
Figura 57. Fluxograma do processo de aquisição de dados na função main do programa.	59
Figura 58. Configuração de um bloco de comunicação para SPI mestre.	60
Figura 59. Pilha de funcionamento da biblioteca FatFs.	61
Figura 60. Formato da string de escrita no ficheiro de texto para um determinado dia.	61
Figura 61. Formato do ficheiro de texto info.txt.....	62
Figura 62. Formato do ficheiro de texto update.txt.....	62
Figura 63. Exemplo do formato dos ficheiros de texto para armazenar os dados.	62
Figura 64. Fluxograma do processo de armazenamento de dados em ficheiros de texto.	63
Figura 65. Fluxograma do processo de leitura de dados do cartão de memória.....	64
Figura 66. Configuração das características e do seu tipo e propriedade.	65
Figura 67. Configuração de definições gerais da camada GAP.....	67
Figura 68. Exemplo de envio de dados para a base de dados GATT.....	68
Figura 69. Procura de dispositivos BLE com o programa CySmart.....	68
Figura 70. Confirmação da transferência dos valores de teste através da característica dataArray.....	69
Figura 71. Fluxograma da aplicação Android desenvolvida.	70
Figura 72. Permissões no Android manifest para utilizar o módulo Bluetooth e a localização.	71
Figura 73. Verificação se o adaptador Bluetooth está ligado.	71
Figura 74. Algoritmo desenvolvido para transferência de dos dados medidos.....	73
Figura 75. Tabelas criadas para guardar os dados transferidos.	75
Figura 76. Protótipo da estação meteorológica desenvolvida.....	76
Figura 77. Descrição da estação meteorológica vista de cima com a caixa aberta ligada à placa de desenvolvimento.	77

Figura 78. Perfboard que integra toda a eletrônica para aquisição de dados, módulo PSoC 4 BLE, leitor do cartão de memória e alimentação da estação.	78
Figura 79. PCB desenvolvida para maximização da extração de energia dos módulos fotovoltaicos usando o conversor LTC3129.....	78
Figura 80. Circuito equivalente para uma célula ou módulo fotovoltaico.....	79
Figura 81. Circuitos equivalentes para os módulos fotovoltaicos amorfo (esquerda) e monocristalino/policristalino (direita) para obtenção dos gráficos I/V e P/V.	80
Figura 82. Configuração da simulação para fazer varrimento em DC.	80
Figura 83. Gráfico I/V para os módulos policristalino e monocristalino.	81
Figura 84. Gráfico P/V para os módulos policristalino e monocristalino ($V_{mpp} = 5 \text{ V}$; $MPP = 1,02 \text{ W}$).81	
Figura 85. Gráfico I/V para o módulo de silício amorfo.	81
Figura 86. Gráfico P/V para o módulo de silício amorfo ($V_{mpp} = 2,98 \text{ V}$; $MPP = 603,83 \text{ mW}$).	82
Figura 87. Circuito equivalente para módulo de monocristalino e policristalino com uma carga de 50Ω na saída.....	82
Figura 88. Tensão e potência de saída obtida no módulo sem MPPC (Potência saída = 624 mW ; $V_{out} = 5,58 \text{ V}$).	82
Figura 89. Simulação do conversor LTC3129 com o modelo do módulo fotovoltaico de silício monocristalino/policristalino.....	83
Figura 90. Valor da potência de saída do conversor e valores de V_{in} e V_{out} ao longo de 6 ms ($V_{mmpc} = 5 \text{ V}$, Carga = 50Ω).	84
Figura 91. Valor da potência de saída na saída do conversor e valores de V_{in} e V_{out} ao longo de 6 ms ($V_{mppc} = 5 \text{ V}$, Carga = 50Ω).	85
Figura 92. Ecrã da aplicação para iniciar aquisição de dados. Esquerda: Antes do início da aquisição. Direita: Depois do início da aquisição.....	86
Figura 93. Teste sobre a aquisição de dados.....	87
Figura 94. Estação meteorológica conectada à placa de desenvolvimento para depuração através de um terminal.	88
Figura 95. Página da aplicação em que se pode iniciar a transferência de dados.....	89
Figura 96. Diálogo do download de dados. Esquerda: para apenas uma data. Direita: caso o utilizador escolha todas as datas disponíveis.	90
Figura 97. Página da transferência de dados após ser feito o download de dados.....	91

Figura 98. Cenário de teste para medição do tempo para a transferência de dados da estação para o dispositivo Android.	92
Figura 99. Cenário de teste para medição do atraso da rede BLE.	93
Figura 100. Cenário de teste para medição do tempo de leitura de dados do cartão de memória.	96
Figura 101. Estação a adquirir dados no exterior.	97
Figura 102. Em cima: Temperatura. Em baixo: Humidade.	98
Figura 103. Em cima: Velocidade do vento. Em baixo: Direção do vento.	99
Figura 104. Irradiação solar medida ao longo de três horas.	99
Figura 105. Potência produzida pelos módulos fotovoltaicos ao longo de três horas.	100
Figura 106. Capacidade atual da bateria.	101
Figura 107. Correlação da capacidade atual da bateria com a irradiação atual.	102
Figura 108. Relação entre potência produzida pelo módulo de silício monocristalino e a irradiação solar.	103
Figura 109. Comparação da potência produzida pelos três módulos fotovoltaicos (Data de aquisição: 26/10/2019).	103
Figura 110. Comparação da potência produzida pelos três módulos fotovoltaicos (Data de aquisição: 18/11/2019).	104

Lista de tabelas

Tabela 1. Algumas tecnologias de painéis fotovoltaicos e as suas eficiências.	11
Tabela 2. Conversores DC-DC para energy harvesting.	16
Tabela 3. Correspondência entre a direção do vento e tensão de saída do circuito.	40
Tabela 4. Calibração do circuito de acondicionamento e a irradiação solar.	42
Tabela 5. Informação sobre os módulos fotovoltaicos adquiridos.	44
Tabela 6. Características criadas para suportar a transferência de dados.	66
Tabela 7. Resultados obtidos através da variação da tensão de MPPC.	84
Tabela 8. Tempos obtidos (em segundos) na transferência de dados entre o dispositivo PSoC e o dispositivo Android.	93
Tabela 9. Atraso (em segundos) na transferência de dados pela rede BLE para diferentes prioridades de conexão e débito obtido.	94
Tabela 10. Tempos de atraso (sem segundos) na transferência de dados pela rede BLE para diferentes MTUs.	95
Tabela 11. Tempo (em segundos) de leitura dos dados do cartão de memória.	96
Tabela 12. Tempo (em segundos) associado à transmissão de dados.	96
Tabela 13. Média da potência produzida pelos módulos fotovoltaicos.	104
Tabela 14. Média da potência produzida pelos módulos fotovoltaicos por 1000 mm ²	104

1. Introdução

1.1 Enquadramento e motivação

Nas últimas décadas a população mundial tem aumentado exponencialmente, aumentando também a demanda de energia. Combustíveis fósseis como o petróleo, carvão e gás natural abastecem 80 % da necessidade energética. Contudo, a maior parte destes tipos de combustíveis apresentam efeitos ambientais negativos devido às emissões de gases de efeito de estufa [1], bem como a sua limitação em quantidade. Tendo isto em conta, existe uma necessidade de usar alternativas de energia que reduzam estes efeitos, focando em fontes de energia que sejam também soluções a longo prazo.

A energia solar é uma categoria das energias renováveis que tem sido explorada para ser uma alternativa a combustíveis fósseis. A cada instante a superfície da Terra recebe 1.8×10^{11} MW de energia solar [2]. Embora não ser possível cobrir todas as necessidades energéticas apenas com esta fonte de energia, ela tem potencial de cobrir uma grande parte.

Para converter a energia solar em eletricidade um dos métodos possíveis é o uso de painéis fotovoltaicos. Esta tecnologia teve ao longo dos últimos anos bom progresso em termos de eficiência. São usados materiais semicondutores que exibem o efeito fotovoltaico para converter energia solar em energia elétrica. Estes semicondutores são feitos maioritariamente de silício, sendo este material bastante abundante. Este método não apresenta efeitos ambientais negativos. Outra vantagem é a possibilidade de criar aplicações de pequena ou grande escala, desde alguns mili-watts até centenas de megawatts. A maior desvantagem é que a energia solar apenas está disponível durante algumas horas por dia [3], e a sua eficiência vai depender da localização e condições ambientais e atmosféricas.

Existe uma variedade de tecnologias de painéis fotovoltaicos no mercado. A decisão de que tecnologia usar vai depender da localização geográfica, do tipo de aplicação e também da escala. Uma das mais utilizadas tecnologias é baseada em silício cristalino. O seu funcionamento é baseado no conceito da junção p-n. Existem vários subtipos de silício que podem ser usados para fabricar painéis fotovoltaicos, sendo os mais usados silício monocristalino e silício policristalino. O primeiro é mais eficiente, mas também mais caro. O segundo é menos eficiente, mas também mais barato, sendo este o tipo mais usado. Outro tipo de tecnologia usada é baseada em filmes-finos. Uma das tecnologias de

filmes-finos mais usadas utiliza o silício amorfo como material de construção. Esta produz menos energia quando comparando com tecnologias baseadas em silício policristalino, mas tem a vantagem de fabricar painéis flexíveis, podendo ser úteis em algumas aplicações [4].

Para determinar a tecnologia fotovoltaica mais adequada a utilizar num determinado local poderá ser útil e necessário obter informação sobre as condições ambientais e atmosféricas dessa localização [5]. Para fazer essa monitorização do ambiente é utilizada normalmente uma estação meteorológica, sendo que esta poderá medir bastantes tipos de dados. Estações meteorológicas têm bastantes áreas de aplicação possíveis, como na agricultura, em que variáveis como velocidade e direção do vento, precipitação e humidade são de grande importância. Outra aplicação possível é na aviação, em que a estação meteorológica implementada tem de ser de alta precisão, pois pequenos erros poderão levar a eventos catastróficos.

Nesta dissertação foi desenvolvida uma estação meteorológica integrando diferentes tecnologias de painéis fotovoltaicos, com o objetivo de correlacionar a potência produzida destes com as variáveis do ambiente medidas. Assim será possível ter uma melhor noção de qual tecnologia utilizar caso se pretenda instalar painéis fotovoltaicos numa determinada área.

1.2 Objetivos

Com esta dissertação pretende-se desenvolver uma estação meteorológica compacta e de baixo custo sendo capaz de medir a temperatura, a irradiação solar, a velocidade e direção do vento. A estação também irá integrar amostras de painéis fotovoltaicos de diferentes tecnologias de silício: monocristalino, policristalino e amorfo. Será feita a monitorização da potência produzida por cada uma destas amostras. Estes dados serão adquiridos através de um microcontrolador, com o qual serão processados e armazenados, sendo posteriormente enviados para um *smartphone* ou *tablet*, recorrendo à tecnologia de rede sem fios BLE (Bluetooth Low Energy). Para estes dados serem interpretados pelo utilizador, será desenvolvida uma aplicação em Android, a qual permite a visualização dos dados numa interface gráfica para melhor compreensão. Os objetivos principais são os seguintes:

- Medição de várias variáveis ambientais: temperatura, velocidade e direção do vento e irradiação solar.
- Medição da produção de energia com diferentes tecnologias de painéis fotovoltaicos.
- Armazenamento local dos dados medidos.

- Transmissão dos dados medidos através de uma rede sem fios para o utilizador.
- Aplicação com interface gráfica para visualização de todos os dados medidos.
- Criação de um protótipo que englobe todo o hardware necessário.
- Correlacionar a potência produzida pelos módulos fotovoltaicos com as condições meteorológicas medidas.

1.3 Estrutura da dissertação

Esta dissertação está organizada em 5 capítulos principais: Introdução, Estado da Arte, Desenvolvimento do Sistema, Resultados e Discussão, e Conclusões.

No capítulo do “Estado da Arte” são apresentados os fundamentos teóricos de várias áreas relacionadas com o tema desta dissertação, bem como trabalhos relacionados e comparação com os mesmos.

No capítulo do “Desenvolvimento do Sistema” é inicialmente descrita a arquitetura do sistema desenvolvido. É depois feita uma apresentação de todo o hardware utilizado na estação meteorológica, incluindo os sensores, painéis fotovoltaicos usados, alimentação da estação e o microcontrolador usado. A secção seguinte é dedicada ao firmware desenvolvido, descrevendo como foi feita a aquisição de todos os dados da estação, bem como o seu armazenamento local e transmissão através da rede sem fios BLE. Na última secção é abordado o desenvolvimento da aplicação em Android, necessária para visualizar os dados da estação.

No capítulo “Resultados e Discussão” é descrito todos os resultados obtidos em todas as partes do sistema desenvolvido, na aquisição de dados dos sensores e módulos fotovoltaicos, na transferência de dados e na visualização de dados através da aplicação Android desenvolvida.

No último capítulo, “Conclusões”, são apresentadas algumas conclusões tiradas do trabalho desenvolvido e também feitas algumas sugestões que possam complementar este trabalho.

2. Estado da arte

2.1 Estações meteorológicas

Uma estação meteorológica está normalmente equipada com instrumentos e equipamentos capazes de medir condições ambientais e atmosféricas. Estas podem variar no tipo de variáveis medidas dependendo do tipo de aplicação e localização. Algumas das variáveis medidas mais comuns são: temperatura, humidade, pressão atmosférica, velocidade e direcção do vento, irradiação solar e precipitação.

Existem vários tipos de estações meteorológicas dependendo da aplicação. Estas podem simplesmente ser usadas em habitações pessoais, na agricultura, em aeroportos ou até em navios. Na Figura 1 é apresentado o exemplo da estação de uso pessoal Ambient Weather WS-2902 [6], que pode ser conectada a uma rede Wi-Fi, sendo assim possível visualizar os dados medidos de forma remota. Esta estação mede a velocidade do vento, direcção do vento, índice pluviométrico, temperatura e humidade exterior, irradiação solar e radiação ultravioleta.



Figura 1. Estação meteorológica Ambient Weather WS-2902A [7].

Na Figura 2 é apresentado o sistema de monitorização ENVIROMONITOR [8], dedicado à agricultura, que usa uma estação meteorológica central como *gateway* e múltiplas outras estações distribuídas pela área como nodos da rede. A informação recolhida por esta estação central é depois enviada para a *cloud* através da Internet usando uma conexão celular, podendo-se aceder aos dados recolhidos através de um

computador ou dispositivos móveis. Cada nodo suporta até quatro sensores, incluindo sensores para a humidade ou salinidade do solo ou até sensores para medir o nível de água numa particular localização. As estações estão encapsuladas num invólucro resistente a condições ambientais duras e são usados painéis fotovoltaicos e baterias para fornecer energia às estações.

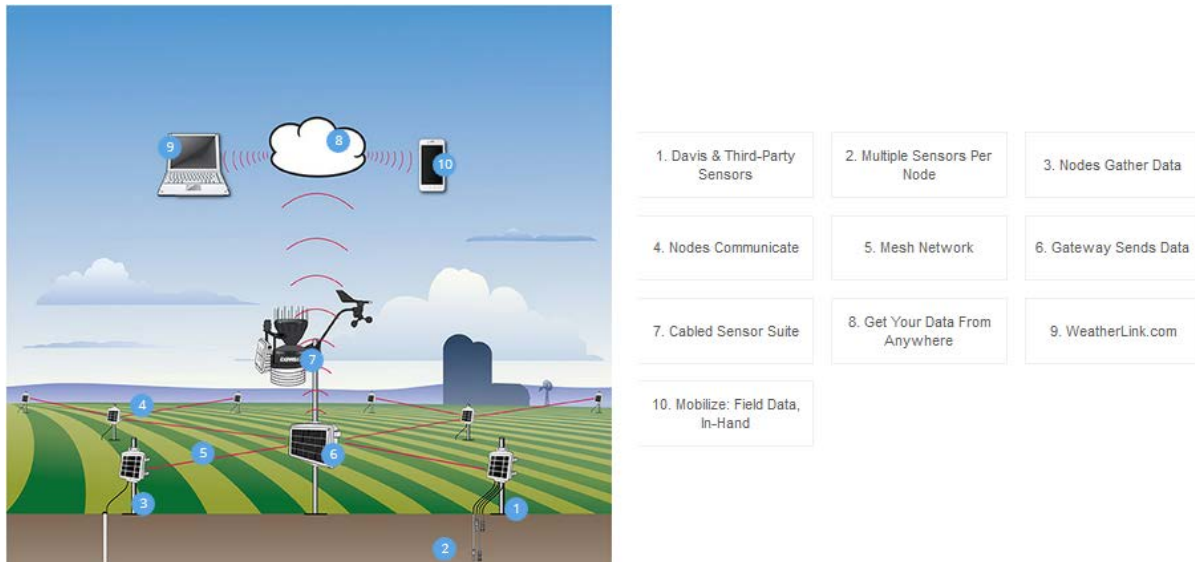


Figura 2. Sistema de monitorização ENVIROMONITOR, composto por uma estação meteorológica atuando como gateway e múltiplas estações como nodos [8].

2.2 Aquisição dos dados meteorológicos

2.2.1 Velocidade do vento

Para medição da velocidade do vento é normalmente utilizado um anemómetro. Este sensor poderá ter, mecanicamente, diferentes formatos. Um dos métodos é a utilização de um íman integrado na parte rotatória e de um sensor em que deteta o campo magnético criado por esse íman. Esse sensor poderá ser um sensor de efeito Hall [9] ou um contacto magnético. Os pulsos produzidos por estes sensores são depois adquiridos pelo microcontrolador, sendo feitos os cálculos necessários para a determinação da velocidade do vento.

Outro método é através de um transmissor e um sensor de infravermelhos, e usando secções brancas e pretas no eixo rotatório. Dado que a luz infravermelha não é refletida na secção branca, mas sim na secção preta, seria possível determinar o número de rotações do eixo.

Também é possível medir a velocidade do vento através da rapidez que um certo material arrefece a partir de uma determinada temperatura inicial. Este método, chamado de *Hot-wire Anemometer*, é

descrito na Figura 3, em que usa um fio aquecido eletricamente até uma certa temperatura. Com o arrefecimento do fio, a sua resistência muda, sendo essa variação medida através do uso de uma ponte de Weathstone, em que é feita a relação entre o valor da resistência e a velocidade do vento.

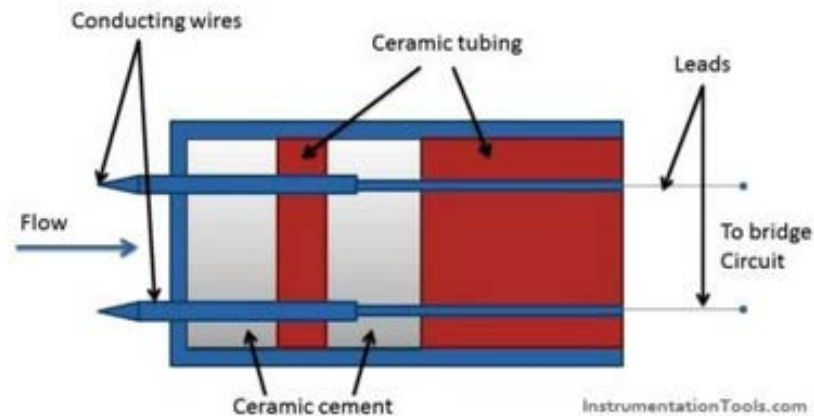


Figura 3. Descrição do funcionamento de um anemômetro baseado num fio aquecido [10].

2.2.2 Direção do vento

Para a detecção da direção do vento são usados cata-ventos eletrônicos. Em termos mecânicos, não diferem muito na sua estrutura. Tal como nos anemômetros, um íman pode ser integrado na parte rotatória, sendo depois usados múltiplos sensores de efeito de Hall ou contactos magnéticos para detetar diferentes direções possíveis do vento de norte a sul. Usando contactos magnéticos, é possível atribuir um contacto para cada direção do vento que se deseja medir, associando-se uma resistência de valor diferente a cada contacto. Dependendo de qual contacto estiver ativado, a resistência associada iria formar um divisor de tensão com uma outra resistência de valor conhecido, podendo-se assim associar um valor de tensão diferente para cada direção do vento.

Outro método possível é a integração mecânica de um potenciômetro ao eixo rotatório, sendo que a variação da resistência desse potenciômetro provoca uma variação na tensão de saída de um divisor de tensão, sendo possível, tal como explicado anteriormente, estabelecer uma relação entre essa tensão e a direção do vento.

2.2.3 Irradiação solar

Para medir a irradiação solar é normalmente usado um piranómetro. Este dispositivo pode variar no tipo de sensores que usam: *thermopiles*, fotodiodos, LDRs (Light Dependent Resistor), fototransistores ou células fotovoltaicas. O espectro de irradiação solar que atinge a superfície da Terra tem comprimentos

de onda entre 300 nm e 2800 nm, sendo que a parte do espectro medida pelo piranómetro vai depender do tipo de sensor usado (Figura 4).

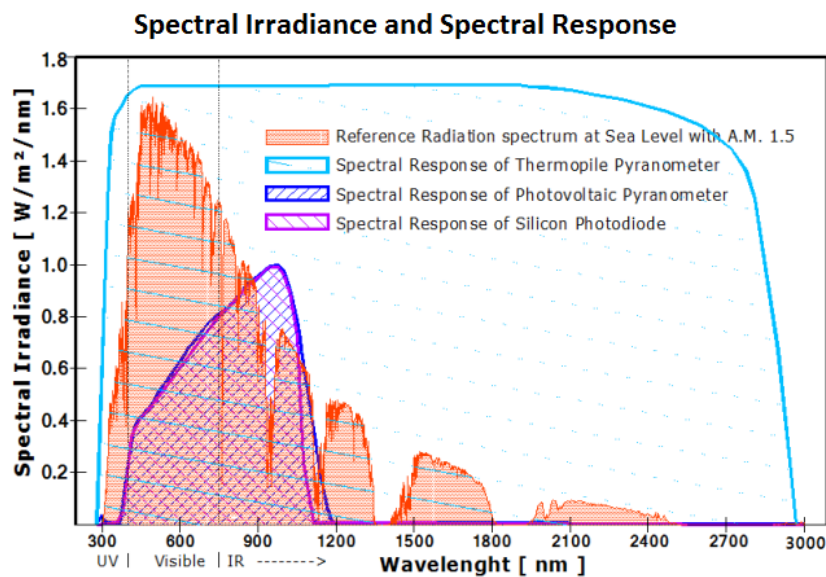


Figura 4. Espectro da irradiação solar medido por alguns tipos de sensores usados em piranómetros [11].

Thermopiles são sensores que convertem energia térmica em energia elétrica. São compostos por várias placas termoelétricas em série. A irradiação é obtida calculando a diferença entre a temperatura num setor preto, exposto ao sol, e a temperatura num setor branco, não exposto ao Sol. Estes sensores conseguem medir o espectro completo de radiação que atinge a superfície da Terra, mas em contrapartida são a opção mais cara [12].

Um fotodíodo converte radiação solar em corrente elétrica, usando o efeito fotoelétrico. Esta conversão pode ser influenciada pela temperatura, sendo que a corrente produzida aumenta com o aumento de temperatura. Este sensor consegue medir comprimentos de onda normalmente entre 440 nm e 1100 nm [13].

Um LDR é uma resistência em que o seu valor varia dependendo da intensidade de luz recebida. A resistência diminui com o aumento dessa intensidade. Estes apresentam uma sensibilidade menor que os fotodíodos e o seu valor irá depender muito da temperatura ambiente. Estes sensores conseguem apenas medir comprimentos de onda até 550 nm, mas têm a vantagem de serem a opção de menor custo.

Um fototransístor funciona de modo similar a um transístor normal, sendo que a diferença é a sua base ser controlada pela intensidade de luz recebida. É gerada uma corrente entre o coletor e o emissor proporcional à radiação incidente. Este tipo de sensores é mais sensível que um fotodíodo.

Na Figura 5 é apresentado um exemplo de piranómetro, neste caso o CMP3 [14], desenvolvido pela Omni Instruments. Este dispositivo usa *thermopiles* para o sensor de base, tendo uma saída analógica de 0 a 30 mV com uma sensibilidade de 5-20 $\mu\text{V}/\text{W}/\text{m}^2$. Usa também uma cúpula de vidro de 4 mm que permite receber a gama espectral total recebido na superfície da Terra.

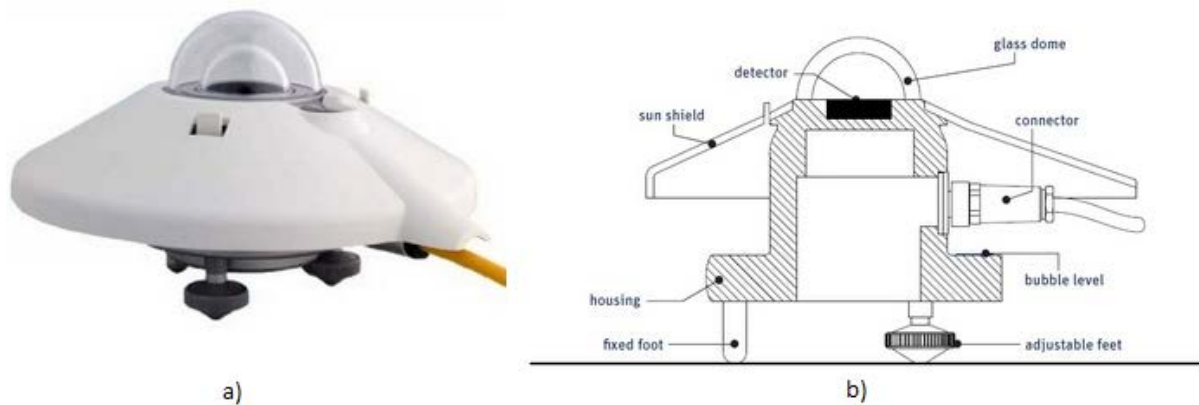


Figura 5. a) Imagem do piranómetro CMP3 b) Diagrama de funcionamento do piranómetro CMP3 [15].

2.2.4 Temperatura

Para a medição da temperatura ambiente existem vários tipos de métodos e sensores disponíveis, dependendo da aplicação. Estes podem ter saída analógica ou digital.

Uma possibilidade para medir temperatura é através do uso de metais puros, devido ao facto de estes terem um comportamento previsível na variação da sua resistência com a temperatura. Este tipo de sensor é chamado de RTD (Resistance Temperature Detector). O material mais usado para este sensor é a platina, devido a ser um metal facilmente disponível na sua forma pura, ter alto ponto de ebulição e ser resistente à corrosão. O valor da resistência vai aumentar com o aumento da temperatura.

Um termistor é um sensor de temperatura resistivo cujo princípio de funcionamento é semelhante ao do RTD, em que o valor da resistência muda com a mudança de temperatura. No entanto, o tipo de material usado é diferente, podendo ser feito de cerâmica ou de polímeros. Outra diferença é que a resistência diminui com o aumento de temperatura. Este tipo de sensores apenas tem uma boa precisão numa determinada gama de temperatura (normalmente entre -90 graus a 180 graus), enquanto que RTDs são mais usados para gamas de temperatura maiores.

Termopares usam o efeito de Seebeck como princípio de funcionamento. Para isto são usados dois tipos de metais, conectados em dois pontos. O nível de energia dos eletrões é diferente nos dois metais, criando um potencial termoelétrico entre eles (Figura 6). Este potencial será proporcional à temperatura

medida, sendo diferente para diferentes tipos de metais usados. Este tipo de sensores têm a vantagem de conseguirem medir temperaturas bastante altas, mas não são tão precisos como os RTDs ou termístores [16].

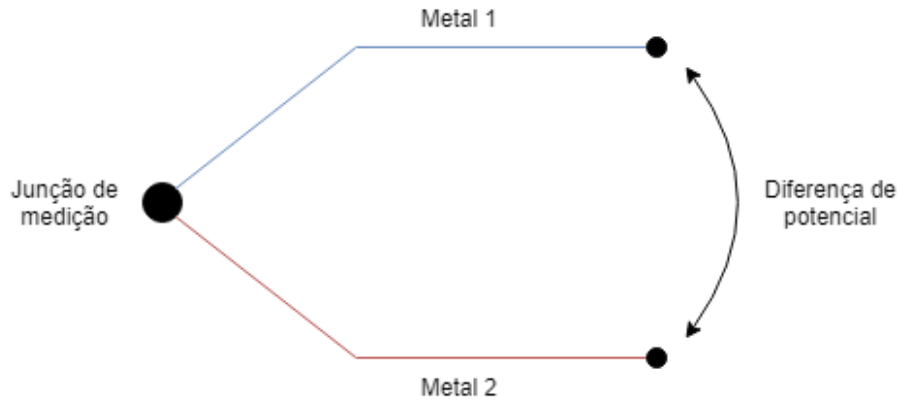


Figura 6. Descrição do funcionamento de um termopar.

2.3 Painéis fotovoltaicos

2.3.1 Visão geral

Painéis fotovoltaicos absorvem energia solar para produzir energia elétrica usando materiais semicondutores que exibem o efeito fotovoltaico, que consiste na criação de tensão e corrente elétrica em um material quando exposto à luz (Figura 7). Um fóton, partícula elementar da luz, quando atinge um material fotovoltaico, pode ter três destinos: passar diretamente pelo material, refletir no material ou ser absorvido pelo material, caso a energia do fóton seja maior que a energia necessária para promover um elétron de valência a um elétron de condução. Isto vai fazer com que elétrons sejam libertados, aumentando assim a corrente na célula fotovoltaica [17]. A quantidade de irradiação solar que é transformada em eletricidade por uma célula fotovoltaica define a sua eficiência.

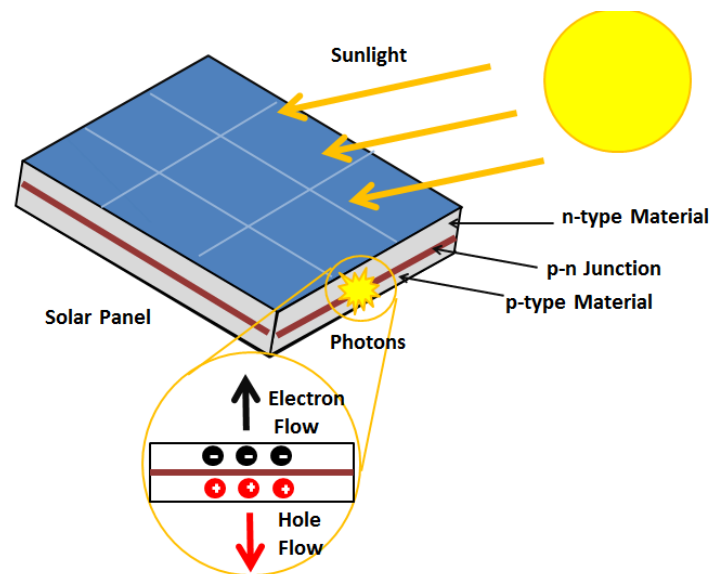


Figura 7. Descrição do efeito fotovoltaico [18].

Estes painéis são constituídos por módulos fotovoltaicos, que por sua vez são constituídos por células fotovoltaicas. Em grandes aplicações os painéis podem ainda constituir uma matriz fotovoltaica. As células fotovoltaicas podem ser associadas em série ou paralelo. Assumindo células com as mesmas especificações, colocando-se células em série a tensão total do módulo formado irá aumentar, sendo igual à soma da tensão de cada célula. Neste caso a corrente máxima do módulo irá ser a corrente máxima de uma célula. Colocando-se as células em paralelo a corrente máxima do módulo irá aumentar, sendo igual à soma da corrente máxima de cada célula. Em contrapartida, a tensão máxima do módulo será a tensão máxima de uma célula. Na maior parte dos casos os módulos fotovoltaicos usam uma associação de células em paralelo e em série, para obter determinados valores de tensão ou corrente, dependendo da aplicação. Na Figura 8 é apresentado um exemplo de duas células fotovoltaicas associadas em série e paralelo e as tensões e correntes obtidas.

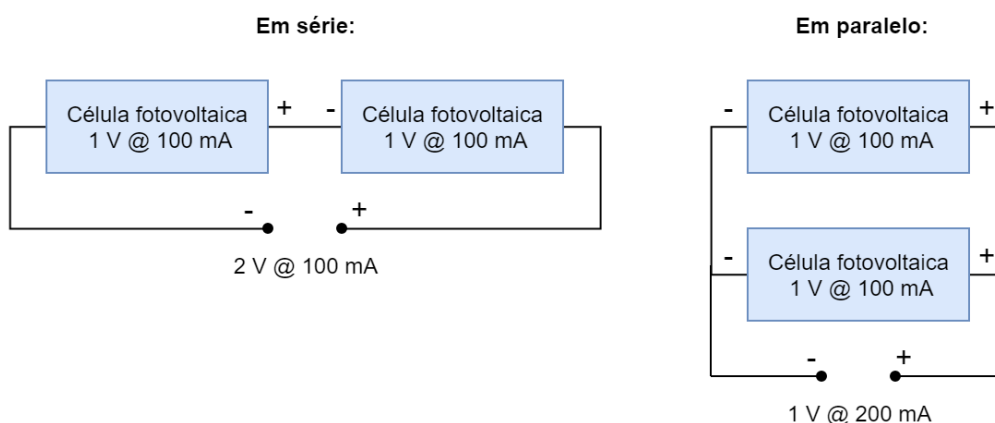


Figura 8. Exemplo de células fotovoltaicas em série e em paralelo e as respetivas tensões e correntes obtidas.

Existem vários materiais possíveis para a criação de módulos fotovoltaicos. O material mais comum é o silício, sendo usadas as suas variações mais comuns:

- **Silício monocristalino:** Apresenta a melhor eficiência (entre 12.5% e 15 %) [19] entre estas três tecnologias devido ao seu alto grau de pureza. Painéis fotovoltaicos baseados nesta tecnologia são os que necessitam menos espaço, sendo que para uma determinada potência irá necessitar um tamanho menor. Em contrapartida, é a tecnologia mais cara para fabricar painéis fotovoltaicos.
- **Silício policristalino:** Apesar de apresentar uma eficiência menor (entre 11 % e 14 %) [19] que a tecnologia de silício monocristalino, o seu custo também é menor devido à técnica usada na sua fabricação. No mercado estes painéis são os mais requisitados devido à sua relação preço/rendimento.
- **Silício amorfo:** Das tecnologias baseadas em silício é aquela que apresentam a menor eficiência (apenas entre 5 % e 7 %) [19]. Devido à sua baixa eficiência, painéis baseados nesta tecnologia necessitam de mais espaço para obter a mesma potência do que as tecnologias de silício cristalino. No entanto, o seu processo de fabrico é bastante barato. Outra vantagem é a possibilidade de fabricar painéis solares flexíveis, sendo útil em algumas aplicações.

Tabela 1. Algumas tecnologias de painéis fotovoltaicos e as suas eficiências.

Tecnologia	Eficiência
Silício Monocristalino (mono-Si)	12.5 - 15 %
Silício Policristalino (poly-Si)	11 - 14 %
Cobre Índio Seleniuro de Galio (CIGS)	10 – 13 %
Teluro de Cadmio (CdTe)	9 – 12 %
Silício Amorfo (a-Si)	5 – 7 %
Diseleneto de índio e cobre (CIS)	16.2 – 22.9 %

2.3.2 Circuito equivalente

Células fotovoltaicas podem ser representadas por um circuito elétrico equivalente. Em condições ideais funcionam como uma fonte de corrente em paralelo com um diodo. Para este diodo terá de se considerar as suas características, como a corrente inversa de saturação e o fator de idealidade. Para representar as perdas é adicionada uma resistência em série e outra em paralelo. Na Figura 9 é apresentado o circuito equivalente de uma célula ou painel fotovoltaico. A resistência R_S representa perdas nas ligações das células. A resistência R_{sh} representa a fuga de correntes nas ligações de alta condutividade da junção p-n [20].

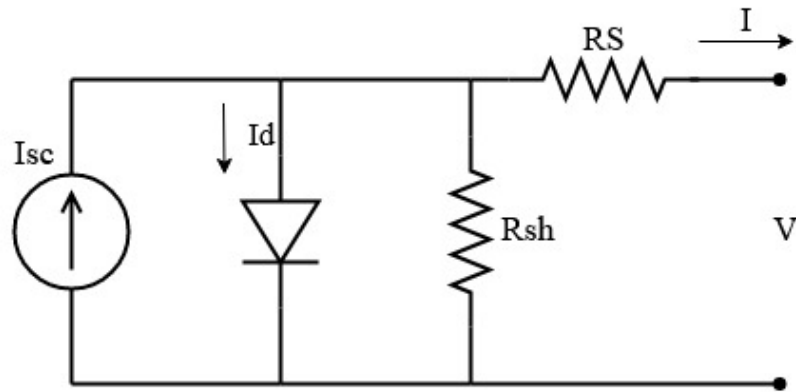


Figura 9. Circuito equivalente de uma célula ou painel fotovoltaico.

Os valores dos componentes deste circuito irão influenciar os valores da corrente de saída (I) e tensão de saída (V) e conseqüentemente o a curva do gráfico corrente-tensão (I/V). Na Figura 10 é apresentado um exemplo do gráfico I/V , com indicação de três características importantes: corrente de curto circuito (I_{sc}), tensão de circuito aberto (V_{oc}) e ponto de potência máximo (MPP – Maximum Power Point).

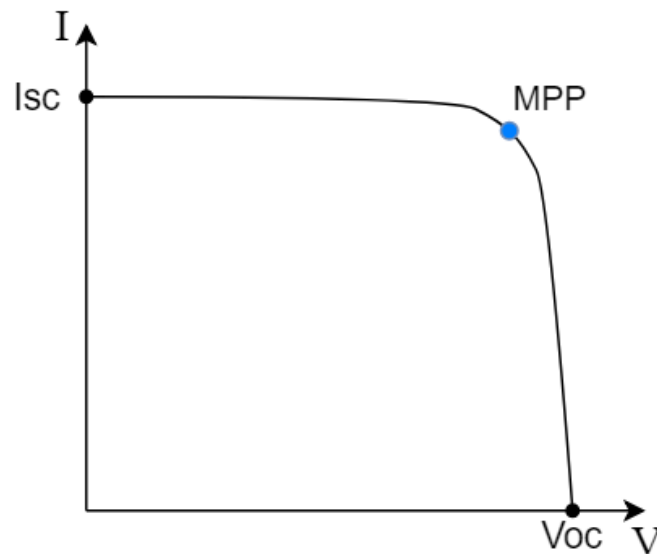


Figura 10. Exemplo de uma curva corrente-tensão (I/V) de um painel fotovoltaico.

2.3.3 Impacto do ambiente na produção de energia

A informação do ambiente onde serão instalados painéis solares fotovoltaicos é muito importante para prever a sua performance em produzir energia, bem como escolher a melhor tecnologia a usar. Irradiação solar e temperatura são, por exemplo, dois fatores que afetam esse processo. Este impacto irá ser diferente para diferentes tecnologias de painéis fotovoltaicos.

Em [5] foi concluído que o aumento de temperatura tem efeitos negativos na produção de energia em painéis fotovoltaicos usando tecnologia de silício cristalino (monocristalino e policristalino). O mesmo não acontece para painéis que usam a tecnologia CIS (Copper Indium Diselenide). A performance dos painéis baseado em silício cristalino foi melhor durante a manhã comparado com os painéis baseados em CIS, mas esta vantagem desaparece durante a tarde quando a temperatura e a irradiação são maiores. Em [21] também foi estudado o efeito da temperatura em módulos fotovoltaicos baseados em silício amorfo. Conclui-se que existe um efeito pequeno da temperatura na eficiência dos painéis, sendo que o aumento da temperatura diminui a eficiência dos módulos fotovoltaicos desta tecnologia.

Em [22] foram conduzidos testes para determinar o impacto de condições meteorológicas na redução da potência de um painel fotovoltaico comparado com um painel fotovoltaico em condições ideais. Os resultados mostraram que a acumulação de poeiras depois de duas semanas de exposição causou uma redução na potência de saída em cerca de 10,8 %. A humidade também mostrou ser um fator de impacto, em que a potência de saída desceu cerca de 40 % quando a humidade relativa era cerca de 76,32%, consequência de precipitação, comparando com o painel fotovoltaico de condições ideais (Figura 11). Em condições nubladas houve uma perda na potência de saída em cerca de 45 %, devido à diminuição da irradiação solar.

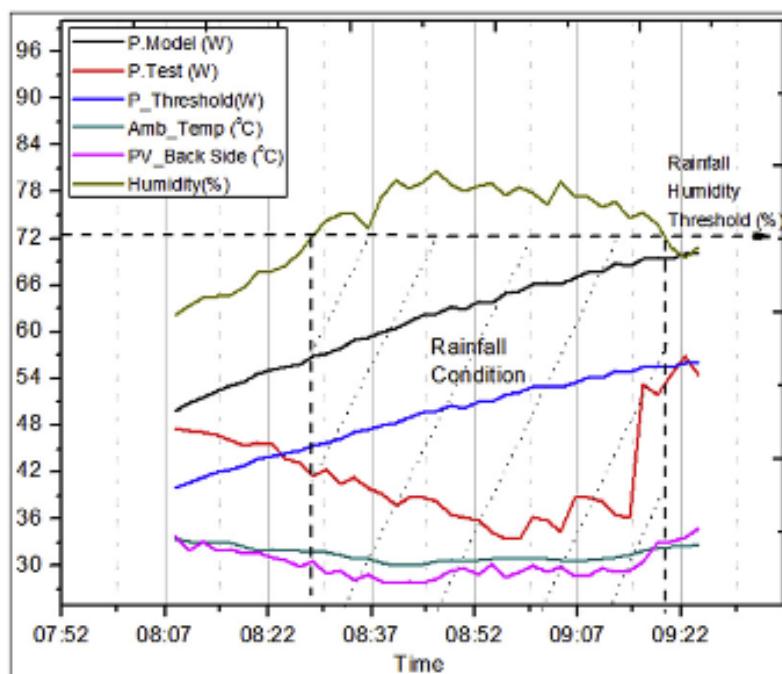


Figura 11. Potência produzida pelo painel fotovoltaico quando exposto a diferentes condições atmosféricas [19].

Em [23] foi estudado o impacto da temperatura e velocidade do vento na eficiência de instalações PV. Conclui-se que a temperatura é um fator importante no processo de conversão de energia solar em

energia elétrica. Com uma maior velocidade do vento, a eficiência na produção de energia aumentou, devido à diminuição da temperatura do painel. Foi mostrada a importância de arrefecer os painéis solares em aplicações de alta temperatura e sem ventilação.

A inclinação e orientação dos módulos PV é outro fator que pode afetar a eficiência. Dependendo desses fatores a irradiação solar recebida na superfície dos painéis solares irá ser diferente para diferentes regiões. Em [24] foi estudado o impacto destes fatores na performance de módulos PV em Singapura, comparando-se várias orientações e inclinações de 10 a 40 graus. Conclui-se que com uma orientação para este e com uma inclinação de 10 graus foi possível obter os melhores resultados naquela região.

2.3.4 Maximum Power Point Tracking

MPPT [25] é uma técnica usada para maximizar a extração de energia proveniente de painéis fotovoltaicos, e também em turbinas eólicas. Dependendo da localização e das condições meteorológicas e ambientais atuais, irá existir uma variação da potência produzida em painéis fotovoltaicos. Irradiação solar e temperatura são duas condições ambientais que afetam a produção de energia solar, como descrito na secção 2.3.3. O objetivo de um algoritmo MPPT permite procurar sempre o ponto de potência máximo (MPP), quando existem estas variações. Na Figura 12 é apresentado um exemplo de um gráfico das curvas de corrente/tensão de um painel fotovoltaico para diferentes condições meteorológicas.

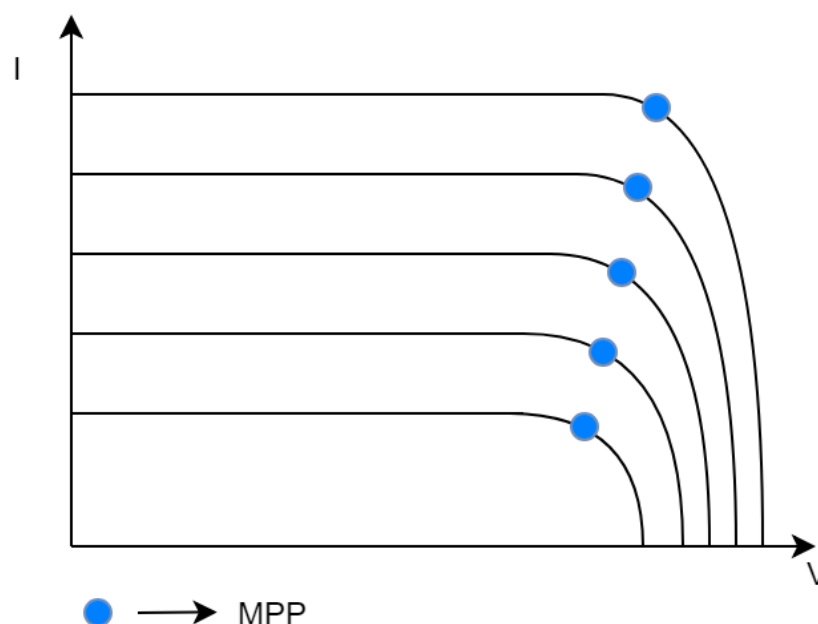


Figura 12. Exemplo do gráfico I-V de um painel fotovoltaico para diferentes condições meteorológicas.

Existem várias técnicas de MPPT possíveis de implementar, dependendo da aplicação, do custo e do nível de complexidade.

A técnica da tensão de curto-circuito fracionária (FOCV – Fractional Open-Circuit Voltage) mede a tensão de curto-circuito (V_{oc}), desligando temporariamente a carga do sistema fotovoltaico. Através desta tensão é possível descobrir qual o valor de V_{mmp} (Maximum Power Point Voltage) através da seguinte equação:

$$V_{mmp} \approx K_{oc} \times V_{oc} \quad (1)$$

O valor de K_{oc} pode variar entre 0,78 e 0,92 [25], dependendo do sistema fotovoltaico.

Perturbação e observação (P & O – Perturbation and Observation) é uma técnica em que é necessário medir a corrente e tensão, calculando a potência respetiva. É aplicada uma pequena perturbação na carga, variando o *duty-cycle* do conversor DC-DC numa direção (para a esquerda ou direita do MPP). Isto é repetido várias vezes e é verificado se a potência medida é maior ou menor que a potência medida anteriormente. Em caso positivo, a perturbação foi feita na direção correta. Em caso negativo, a perturbação foi feita na direção errada, sendo aplicada na outra direção. Isto é feito até que se chegue ao valor de MPP.

Na técnica de condutância incremental (Inc-Cond – Incremental Conductance) é calculada a derivada da potência produzida pelo painel fotovoltaico através da tensão. No MPP, este valor será zero, sendo negativo caso esteja à esquerda do MPP ou positivo caso esteja à direita do MPP. Calculando continuamente este valor é então possível saber qual o valor de MPP.

2.3.5 Circuitos integrados para *energy harvesting*

De modo a maximizar a extração de energia de painéis ou módulos fotovoltaicos, é normalmente utilizado um conversor DC-DC com capacidades de MPPT. Para painéis fotovoltaicos existe uma grande variedade no mercado destes conversores. Para módulos fotovoltaicos de pequenas dimensões, as opções são mais limitadas. Estes conversores estão normalmente disponíveis em circuitos integrados com capacidades de MPPT ou variações desta técnica. Uma das variações é chamada de MPPC (Maximum Power Point Control) [26] em que o conversor irá procurar ter a tensão de entrada igual a uma tensão de referência programada. Esta técnica aproveita o facto de a tensão de MPP não variar muito quando existem mudanças nas condições meteorológicas, ao contrário da corrente. Na Tabela 2 são apresentados alguns conversores DC-DC dedicados ao uso em *energy harvesting* [27]. Alguns destes

conversores como o LTC3105 ou BQ25570 podem ser usados para outras fontes de energia, como a energia termoelétrica proveniente de uma célula de Peltier.

Tabela 2. Conversores DC-DC para *energy harvesting*.

Nome	Topologia	Capacidade de MPPT	Tensão mínima entrada (V)	Tensão máxima entrada (V)	Potência máxima (mW)
LTC3105	Boost	MPPC	0,225	5	-
LTC3129	Buck/Boost	MPPC	2,42	15	-
LTC3331	Buck/Boost	-	3	19	-
BQ25570	Boost	MPPT	0,1	5,1	510
ADP5902	Boost	MPPT	0,08	3,3	600
SPV1040	Boost	MPPT	0,240	5,5	3000

2.4 Tecnologias de redes sem fios

2.4.1 Visão geral

Existem várias tecnologias de redes sem fios, com vantagens e desvantagens dependendo da aplicação desenvolvida.

Uma WLAN (Wireless Local Area Network) é uma tecnologia de redes sem fios que surgiu como evolução das LANs (Local Area Network) por cabos, nomeadamente as redes IEEE 802.3/Ethernet [28], que foram desenvolvidas inicialmente com o objetivo de interligar em rede computadores (e periféricos como impressoras) instalados num mesmo local. As redes LAN e WLAN são normalmente utilizadas em residências, escolas, universidades, escritórios etc. Hoje em dia uma das principais aplicações das WLANs consiste em possibilitar a conexão sem fios de dispositivos à Internet. O IEEE 802.11 [29] é o standard mais utilizado pelas WLANs modernas, sendo utilizado pela tecnologia Wi-Fi. A taxa de transmissão oferecida por esta tecnologia ao nível da camada física pode variar bastante, dependendo da versão, sendo 11 Gbit/s o valor máximo esperado na versão mais recente, IEEE 802.11ax [22]. O alcance também depende da versão e se a rede está localizada no interior ou exterior. No interior pode chegar a algumas dezenas de metros e no exterior a algumas centenas de metros.

Para comunicações de longo alcance e baixo consumo existem as redes LPWANs (Low Power Wide Area Networks) [30]. Em comparação com as redes WAN (Wide Area Network), as redes LPWAN oferecem baixo consumo de energia, mas em contrapartida têm baixa taxa de transmissão. Na área da Internet das Coisas (IoT – Internet of Things), estas tecnologias podem ser uma alternativa a outras redes mais tradicionais, como as redes celulares ou redes de curto alcance, dependendo da aplicação. Algumas das tecnologias de redes LPWAN podem ser destacadas:

- LoRa, desenvolvida pela Semtech Corporation, utiliza uma técnica espalhamento espectral (*spread spectrum*) derivada da conhecida técnica CSS (Chirp Spread Spectrum), sendo a primeira implementação de baixo consumo de energia usando esta técnica. A taxa de transmissão destas redes pode variar entre 0.3 e 37.5 kbps/s, dependendo do *spread factor* e da largura de banda do canal. O tamanho da trama pode ir até 250 bytes e o seu alcance pode chegar até 5 km numa zona urbana e 15 km numa zona rural [30].
- SigFox oferece uma solução baseada no princípio *end-to-end*, com a comunicação sendo feita entre os nodos da rede sem intermediários como *gateways* ou *routers*. Os nodos da rede são conectados a uma estação base usando a modulação BPSK (Binary Phase Shift Keying). A taxa de transmissão pode chegar a 100 bps no sentido ascendente (*uplink*) e 600 bps no sentido descendente (*downlink*). O alcance atingido é bastante grande comparado com outras LPWANs, podendo ir até 10 km numa zona urbana e 50 km numa zona rural [30].

Para comunicações de curto alcance existem as WPAN (Wireless Personal Area Network). Estas apresentam normalmente alcance menor do que as redes WLAN, mas também menor complexidade, custo e consumo de energia dos dispositivos, tendo como foco comunicações centradas em torno de um individuo, como a residência ou no local de trabalho. Em termos de taxa de transmissão, esta pode variar bastante dependendo da tecnologia. Bluetooth e ZigBee são algumas tecnologias de WPAN que se podem destacar:

- O Bluetooth é uma tecnologia de baixo alcance (até 100 metros) usada para transmitir dados entre dispositivos fixos ou móveis. Esta rede opera sem necessidade de licença na banda ISM (Industrial, Scientific and Medical) de 2.4 GHz, como a maior parte das redes WLAN e WPAN. A taxa de transmissão pode ser de 1 Mbit/s ou superior, dependendo da versão. Mais recentemente foi introduzida a versão de baixo consumo desta rede, chamada de Bluetooth Low Energy (BLE) [31], que não é diretamente compatível com a versão clássica do Bluetooth.

- O ZigBee é baseado no standard IEEE 802.15.4, sendo desenvolvido para ser uma solução mais simples e menor custo e consumo de energia do que outras WPANs. Tem um alcance direto de até 100 metros, que vai depender da localização e da potência de transmissão, mas possibilita o aumento da área de cobertura da rede através do uso de *routers* em topologias *multihop*. A taxa de transmissão desta rede é baixa comparado com o Bluetooth, podendo chegar até 250 kbit/s. Esta rede tem aplicações na automação residencial e industrial, dispositivos médicos, aquisição de dados de sensores, entre outros.

2.4.2 Bluetooth Low Energy

O BLE foi a rede sem fios escolhida no desenvolvimento desta estação. Esta rede é um protocolo de comunicação sem fios de curto alcance e baixo consumo. Comparado com o Bluetooth clássico apresenta um consumo bastante menor enquanto mantém o mesmo alcance. Versões mais recentes de sistemas operativos de dispositivos moveis apresentam suporte para BLE, como o IOS, Android e Windows Phone. Existe também suporte em sistemas operativos de desktop, incluindo o macOS, Linux, Windows 8 e Windows 10.

Ao longo dos últimos anos existiram várias iterações da rede Bluetooth. A primeira vez em que foi introduzida a tecnologia Bluetooth Low Energy foi na versão 4.0. Esta versão apresenta uma nova pilha protocolar e um consumo de energia muito menor comparado com as versões anteriores, mas mantendo alcance similar ao Bluetooth clássico. A versão 4.1 apresentou melhorias em termos de software, com novas técnicas de gestão de energia. A versão 4.2 apresentou um aumento no tamanho máximo do pacote de dados, aumentando o MTU (Maximum Transmission Unit) de 23 bytes para um máximo de 512 bytes. Isto permite débitos maiores que as versões anteriores. A versão 5.0 [32] foi introduzida com um foco maior em IoT. Permite velocidades superiores (até 2 Mbps) e maior alcance que as versões anteriores (até 300 metros). Apesar de não conter o sufixo Low Energy, usa na mesma o protocolo que permite baixo consumo de energia. A última versão apresentada é a 5.1, incluindo novas técnicas que permitem saber a direção da fonte de um sinal Bluetooth.

2.4.2.1 Pilha protocolar

Na Figura 13 é apresentada a arquitetura da pilha protocolar (*stack*) do BLE. Esta *stack* foi projetada para transmitir dados com baixo consumo de energia, sendo assim uma rede adequada para dispositivos alimentados a bateria.

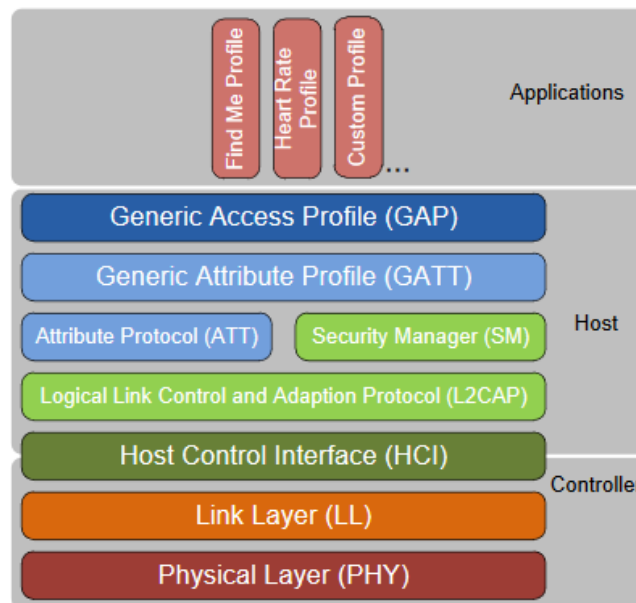


Figura 13. Arquitetura da stack de protocolos do BLE [33].

A *stack* é dividida em três secções: aplicações, hospedeiro (*host*) e controlador. A secção de aplicações, como o nome indica, contem as aplicações desenvolvidas para um determinado objetivo.

Na secção do hospedeiro é feito o tratamento das conexões entre dispositivos e dos dados a serem transmitidos. Aqui existem as camadas L2CAP (Logic Link Control and Adaptation Protocol), ATT (Attribute Protocol), SM (Security Manager), GAP (General Access Profile) e GATT (Generic Attribute Profile):

- A camada ATT define como os dados são organizados e acedidos.
- A camada L2CAP é responsável pela fragmentação e organização dos pacotes de dados que a camada ATT vai usar. Nesta camada também é definido o MTU (Maximum Transmit Unit) que a camada ATT pode utilizar.
- A camada SM é responsável pela segurança de uma conexão BLE.
- A camada GAP é responsável pelo funcionamento das conexões entre dispositivos BLE. Os dispositivos podem ser denominados periférico ou central. Um dispositivo periférico anuncia a sua presença e aceita ou rejeita a conexão do dispositivo central. Um dispositivo central escuta esses anúncios e inicia o processo de conexão entre os dois dispositivos. Nesta camada é controlada as configurações da conexão, como o intervalo de conexão, a latência do escravo e o tempo limite de supervisão.
- A camada GATT é responsável por definir como os dados definidos pela camada ATT são acedidos. Tal como a camada GAP, a camada GATT divide os dispositivos em duas categorias: servidor e

cliente. O cliente GATT é o dispositivo que configura e recebe dados do servidor GATT. O servidor GATT é o dispositivo que contém dados a serem transmitidos ao cliente GATT.

A secção do controlador corresponde ao hardware que transmite e recebe os sinais de radiofrequência. Aqui existem três camadas: a camada física (PHY), a camada de ligação (LL – Link Layer) e a camada da interface do controlo do hospedeiro (HCI – Host Controller Interface):

- A camada física corresponde ao módulo de radiofrequência (RF) na banda de 2.4 GHz, com uma taxa de transmissão de no mínimo 1 Mbits/s.
- A camada de ligação controla o estado do módulo RF do dispositivo na camada física.
- A camada HCI conecta a camada do controlador do hardware (PHY + LL) com o *firmware* da camada do hospedeiro.

Na Figura 14 é apresentado o formato do pacote de dados para várias camadas dentro da *stack* do BLE 4.2 e o tamanho dos seus respetivos campos (em bytes). Os dados que são transferidos de um dispositivo para o outro têm o formato definido pelo último pacote (camada física + ligação), sendo o valor máximo do campo Data de 255 bytes. Apesar do campo ATT Data do pacote de dados da camada L2CAP ter um máximo de 251 bytes, o campo Data do pacote da camada ATT poderá ser maior. Isto acontece porque o valor máximo de dados que é transferido entre as camadas ATT e acima poderá ser maior do que aquele definido na camada física + ligação [34]. Esse valor vai depender do MTU definido na conexão entre os dispositivos.

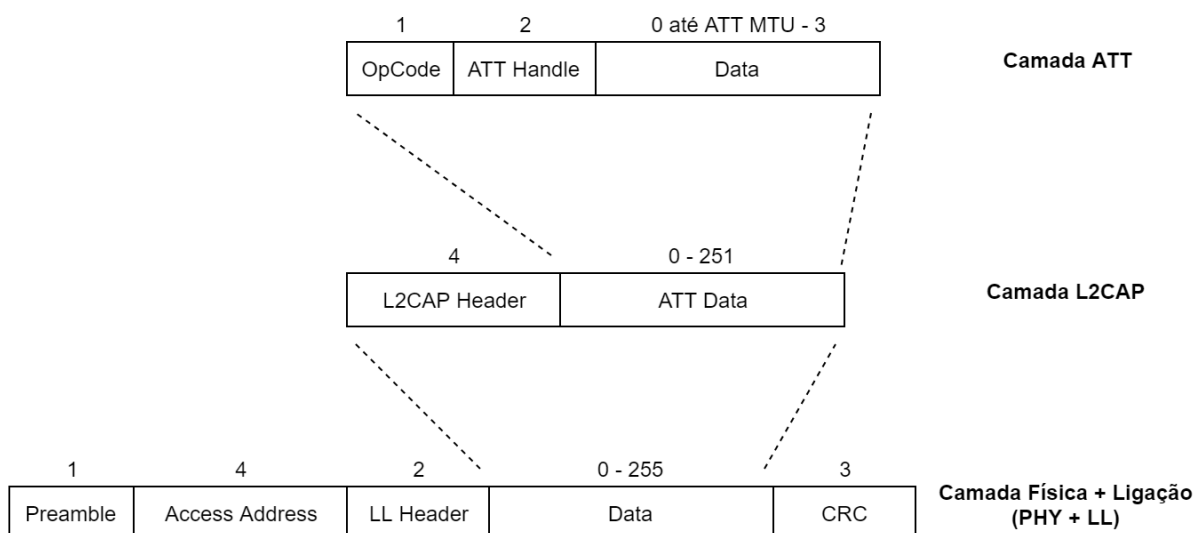


Figura 14. Formato de pacotes para várias camadas da stack do BLE.

Na Figura 15 é apresentado um exemplo de uma possível arquitetura de uma aplicação BLE. Neste exemplo o *smartphone* funciona como dispositivo GAP central e cliente GATT e o módulo BLE vai

funcionar como dispositivo GAP periférico e servidor GATT. O módulo BLE neste caso vai ter os dados a serem transmitidos para o *smartphone*, fazendo o anúncio da sua presença. O *smartphone* irá iniciar a conexão com o módulo BLE e transferir os dados.

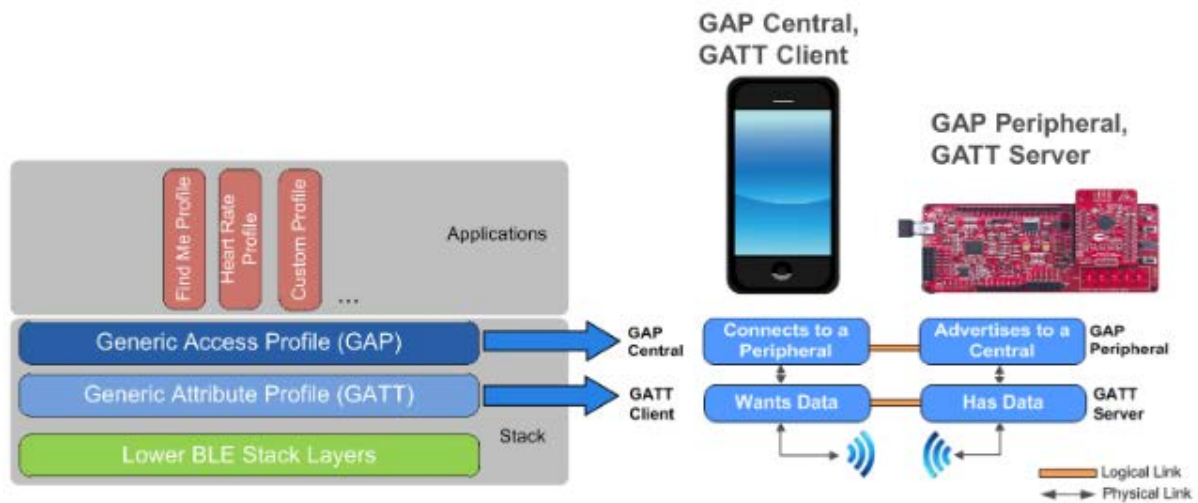


Figura 15. Visão geral de uma possível arquitetura de uma aplicação BLE [35].

2.4.2.2 Formato dos dados do servidor GATT

Um servidor GATT, como explicado anteriormente, guarda os dados a serem transmitidos ao cliente GATT. Para representar estes dados de uma forma organizada são usados atributos, características e serviços. Um serviço pode conter uma ou mais características, e uma característica pode ser composta por múltiplos atributos.

Um atributo é a parte menos abstraída na camada GATT, sendo o que realmente vai conter os dados. Este vai ser constituído por um endereço, um UUID (Universally Unique Identifier), que vai especificar o tipo de dados, o valor (os dados a serem transmitidos) de até 512 bytes e o tipo de permissão. A permissão permite identificar se o atributo é de escrita ou leitura, tipo de autenticação e de encriptação, entre outros. Quando é feita uma operação de escrita ou leitura, é apenas necessário saber o endereço do atributo no servidor GATT. Na Figura 16 é apresentado visualmente o formato de um exemplo de um atributo.

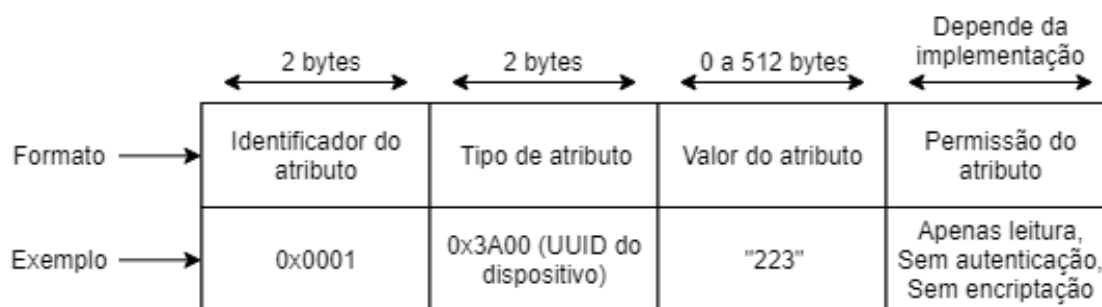


Figura 16. Formato e exemplo de um atributo.

Uma característica é composta por pelo menos dois atributos, que definem os dados a serem transmitidos de uma forma mais abstraída. Os dois atributos principais são o atributo de declaração da característica e o atributo do valor da característica. A declaração da característica é uma descrição das propriedades e tipo de dados do valor da característica. O atributo do valor da característica é o valor a ser transferido. Na Figura 17 é apresentado o formato e exemplo de uma característica.

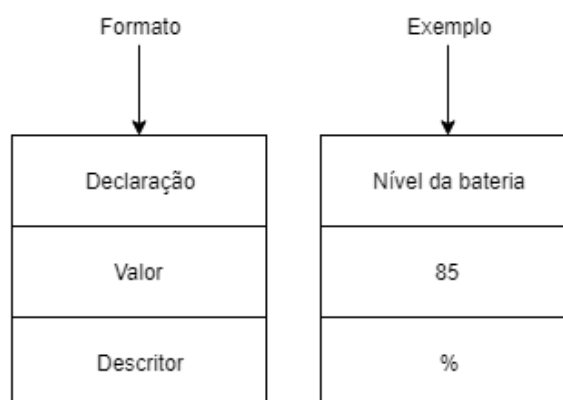


Figura 17. Formato e exemplo de uma característica.

Um serviço é composto por uma ou mais características que vão definir qual o objetivo ou função do dispositivo BLE. Na Figura 18 é apresentado o exemplo de um serviço para a medida da frequência cardíaca, incluindo características para a medida da frequência cardíaca, localização do sensor no corpo e ponto de controlo da frequência cardíaca. Também é apresentado o exemplo de como os dados de medição da frequência cardíaca são guardados no formato de dados pretendido, e quais as funções de cada dispositivo.

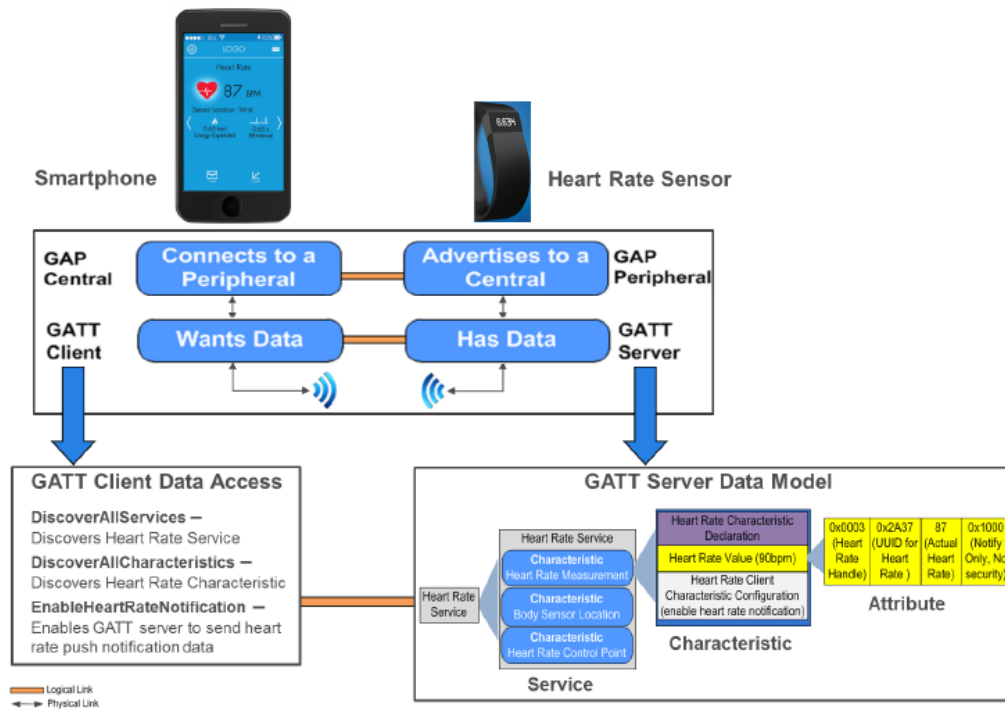


Figura 18. Exemplo do formato de dados de um servidor GATT para um serviço de medição da frequência cardíaca [35].

2.4.2.3 Intervalo de conexão

O intervalo de tempo entre eventos de transferência de dados é chamado de intervalo de conexão. O evento de transferência de dados também pode ser designado por evento de conexão. Neste evento de conexão, a comunicação entre os dispositivos é iniciada pelo dispositivo GAP central, o mestre (M), pelo envio de um pacote para o dispositivo GAP periférico, o escravo (E). Ao receber este pacote o dispositivo periférico envia um pacote para o dispositivo central. Isto poderá acontecer várias vezes durante um intervalo de conexão (Figura 19). O número de vezes que isto acontece irá afetar o débito da rede BLE, pois quantos mais pacotes forem enviados entre o dispositivo central e periférico num intervalo de conexão, maior será o débito. O número de pacotes que podem ser transmitidos num intervalo de conexão vai depender dos dispositivos usados e da sua implementação. O intervalo de conexão pode ter um valor entre 7.5 ms e 4 segundos. Este valor também irá afetar o débito de uma conexão BLE.

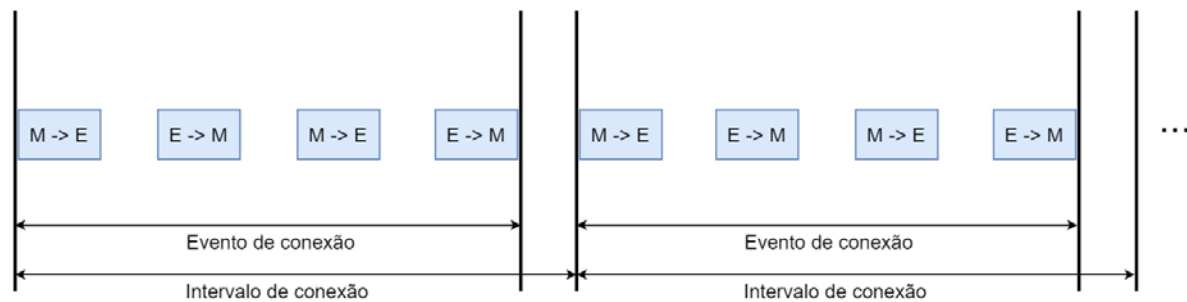


Figura 19. Exemplificação do intervalo de conexão entre dois dispositivos.

2.5 Sistema operativo Android

O Android é um sistema operativo desenvolvido pela Google baseado em Linux. Este sistema operativo foi projetado para dispositivos móveis, sendo usado em *smartphones*, *tablets*, relógios de pulso, e também mais recentemente em televisões. De acordo com [36], é o sistema operativo mais vendido no mundo para *smartphones* desde 2011 e para *tablets* desde 2013.

O desenvolvimento de aplicações em Android é feito usando o Android SDK (Software Development Kit), juntamente com uma linguagem de programação, podendo ser Java, Kotlin ou C++. Este kit inclui várias ferramentas de desenvolvimento, como bibliotecas, emulador, *debugger*, documentação e códigos exemplo. Apesar de inicialmente a Google suportar o ambiente de desenvolvimento (IDE – Integrated Design Environment) Eclipse, em dezembro de 2014 foi lançada a versão 1.0 do IDE Android Studio, ficando este a ser o principal IDE para desenvolvimento de aplicações Android.

2.5.1 Atividades

A atividade é o componente mais fundamental de uma aplicação Android, representando uma página da aplicação com a interface do utilizador. No arranque de uma aplicação é sempre iniciada uma atividade. Na Figura 20 é apresentado um exemplo básico de duas atividades, em que a segunda atividade é iniciada premindo um botão na primeira atividade.



Figura 20. Mudança de uma atividade para outra através de um botão [37].

As atividades têm como objetivo principal uma fácil interação entre o sistema e a interface do utilizador dentro da aplicação.

À medida que o utilizador navega dentro de uma aplicação, as atividades serão abertas e fechadas múltiplas vezes, passando por vários estados no seu ciclo de vida. O que acontece quando existe uma mudança nesse ciclo de vida pode ser programado de acordo com o objetivo da aplicação. Isto previne que aplicação aborte de modo inesperado ou que utilize recursos do sistema quando não é necessário. Dentro do ciclo de vida de uma atividade existem seis métodos: `onCreate`, `onStart`, `onResume`, `onPause`, `onStop` e `onDestroy`. Na Figura 21 é apresentado o que acontece durante o ciclo de vida de uma atividade, dependendo do que acontece na aplicação.

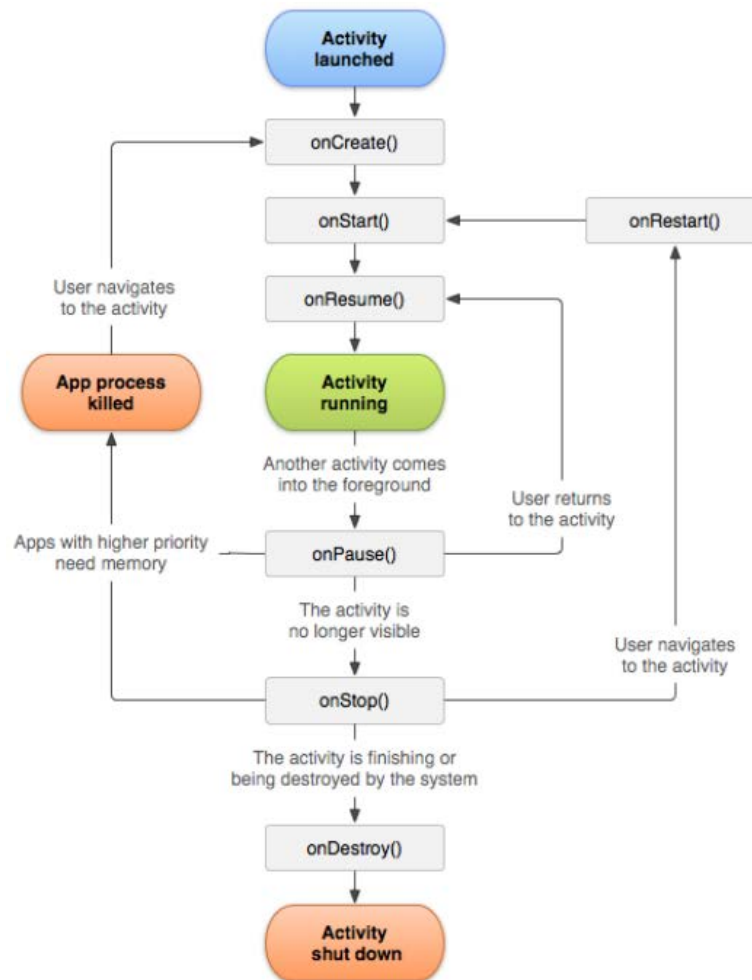


Figura 21. Ciclo de vida de uma atividade de uma aplicação Android [38].

Inicialmente é chamado o método `onCreate`, sendo este apenas chamado uma vez durante o ciclo de vida de uma atividade. Aqui, por exemplo, são instanciadas variáveis para uso de *text views* e botões. O método `onStart` faz a atividade ficar visível para o utilizador, sendo aqui onde o código de interface do utilizador é inicializado. Quando a aplicação está pronta para ser utilizada o método `onResume` é chamado. A aplicação fica neste estado até algo acontecer. Quando o utilizador deixa uma atividade, o método `onPause` é chamado. A atividade não é finalizada, apenas não está em primeiro plano. Aqui podem ser feitas modificações à atividade até esta ficar visível novamente. O método `onStop` é chamado quando a atividade já não é visível ao utilizador. Isto pode acontecer quando outra atividade é chamada por cima desta ou a atividade está para ser terminada. Para destruir uma atividade é chamado o método `onDestroy`, isto acontecendo sempre que uma atividade é finalizada. Este método também pode ser chamado para ser feita alguma configuração à atividade, como a rotação do dispositivo.

2.5.2 Fragmentos

Um fragmento representa um determinado comportamento da interface com utilizador de uma atividade. Uma atividade poderá ter múltiplos fragmentos, sendo que estes fragmentos poderão ser usados em diferentes atividades. Um exemplo de um fragmento é quando é necessário escolher uma data ou hora numa aplicação, sendo iniciado um diálogo para tal. Um fragmento terá o seu próprio ciclo de vida tal como uma atividade. A utilização de fragmentos permite ter uma maior flexibilidade na construção de interfaces com o utilizador. Na Figura 22 é apresentado a utilização de um fragmento que hospeda um diálogo que permite ao utilizador escolher uma data.



Figura 22. Exemplo da utilização de um fragmento. Neste caso o fragmento hospeda um diálogo para escolher uma data.

2.5.3 Serviços

Por vezes as aplicações em Android necessitam de efetuar operações que podem ser longas. Para este propósito normalmente é usado um serviço, sendo este um componente de uma aplicação Android que permite efetuar operações em *background*. Ao contrário das atividades, um serviço não tem uma interface com o utilizador. Alguns exemplos em que um serviço pode ser utilizado é, por exemplo, numa aplicação para reprodução de música ou quando é necessário a transferência de dados através da Internet ou Bluetooth.

2.5.4 Android *manifest*

No ficheiro Android *manifest* são descritas informações sobre a aplicação necessárias para o sistema operativo e também para a Google Play, caso a aplicação seja publicada. Aqui são declarados todos os componentes do sistema, atividade, serviços, recetores de transmissão e provedores de conteúdo, e todas as suas propriedades básicas (por exemplo o nome das classes associadas a esses componentes). Também é aqui que são declaradas todas as permissões para aceder a partes protegidas do sistema, como por exemplo, a localização do *smartphone*. Algumas aplicações necessitam determinado hardware ou software para funcionar, sendo essas especificações aqui descritas.

2.6 Trabalho relacionado

Em [39] foi desenvolvido um sistema de monitorização de painéis solares distribuídos, usando a rede de sensores sem fios ZigBee. Em cada localização de um painel solar foi medida a tensão e corrente produzida, bem como a temperatura e irradiação solar, sendo cada estação considerada um escravo da rede (Figura 23.a). Para a medição da tensão, foi simplesmente utilizado um divisor de tensão, sendo que para a medição da corrente, foi medida a queda de tensão numa resistência de baixo valor. Para medir a irradiação solar foi utilizado o fotodiodo PDB-C139, sendo usado um amplificador operacional para amplificar o sinal produzido, que depois é lido pelo ADC (Analog-to-Digital Converter) do microcontrolador. A temperatura foi medida usando um simples termistor NTC (Negative Temperature Coefficient). Todos os dados são adquiridos por um microcontrolador existente em cada painel solar, sendo depois enviados para um coordenador usando a rede ZigBee. Este coordenador está ligado a um computador via USB (Figura 23.b). Os dados foram guardados numa base de dados usando ficheiros de texto.

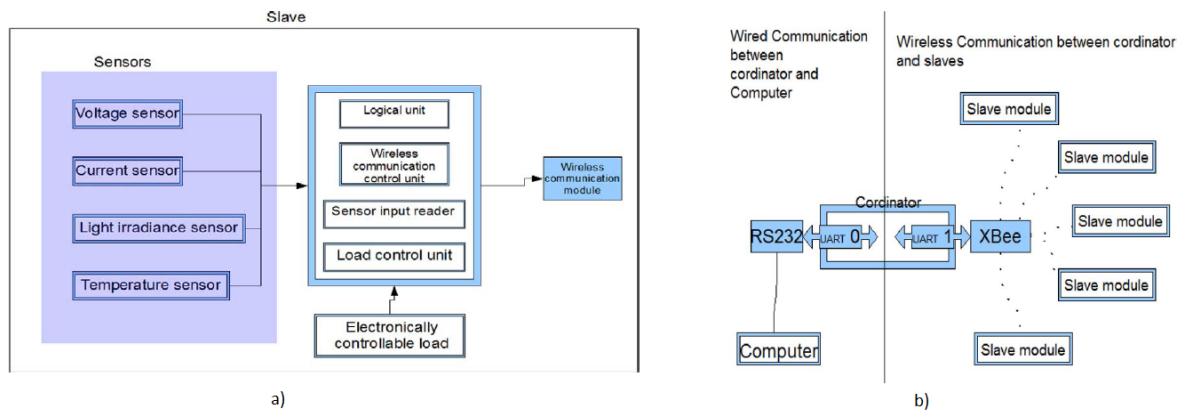


Figura 23. a) Representação de um escravo da rede, que inclui um painel solar e algumas variáveis a serem medidas b) Representação da comunicação desde os escravos até ao computador, usando um coordenador como intermediário.

Em [40] foi desenvolvido um *data logger* que tem como objetivo a medição das condições meteorológicas. Na Figura 24 é apresentada a arquitetura do sistema implementado. Foi usada uma conexão GPRS (General Packet Radio Service) usando um cartão SIM (Subscriber Identification Module) para transmitir os dados para um servidor FTP (File Transfer Protocol). Para adquirir os dados foi desenvolvida uma estação meteorológica que permite medir a temperatura, pressão barométrica, humidade, velocidade e direção do vento e quantidade de precipitação. O *data logger* é constituído por um microcontrolador ATmega64 e vários periféricos. Os dados foram guardados localmente através do uso de um cartão de memória. Foi também usado um RTC (Real Time Clock) externo para enviar comandos a determinada hora e data. A estação meteorológica envia relatórios de modo periódico com os valores medidos. Para isso, o *data logger* cria um ficheiro de texto para acomodar os novos dados, enviando-os depois para o servidor FTP. É possível também configurar o *data logger* através de uma comunicação série e um computador. Aqui podem ser configurados até 70 parâmetros, incluindo o endereço do servidor remoto, a data e hora atual, a periodicidade da aquisição dos dados da estação meteorológica, configurações da conexão GPRS etc.

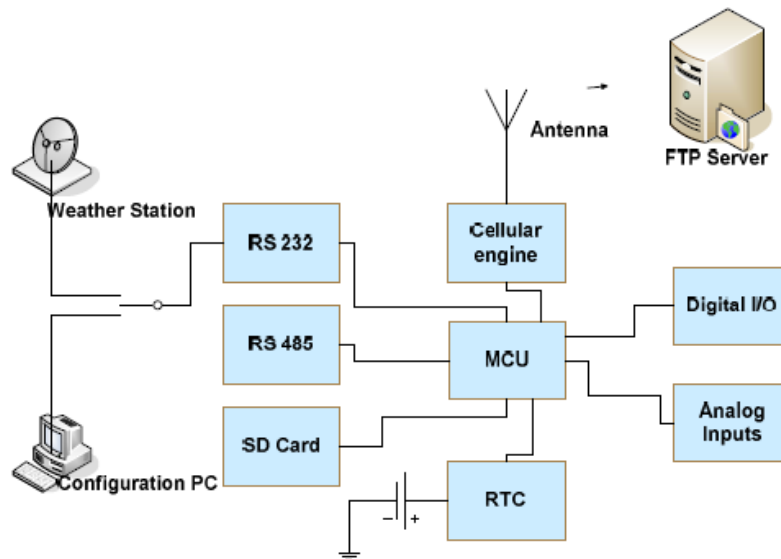


Figura 24. Arquitetura do sistema implementado em [40].

Em [41] foi desenvolvida uma estação meteorológica de baixo custo alimentada por painéis fotovoltaicos e por uma bateria do tipo LiPo com o objetivo de selecionar possíveis locais para o desenvolvimento de parque eólicos. A estação mede as seguintes variáveis ambientais: temperatura, velocidade e direção do vento, humidade relativa, pressão atmosférica, precipitação e qualidade do ar. Estes sensores estão embebidos numa placa de aquisição de dados meteorológicos chamada de OurWeather, contendo uma placa ESP8266 com antena Wi-Fi integrada. Uma Raspberry Pi comunica com a placa OurWeather através da rede Wi-Fi, recebendo todos os dados adquiridos. Estes dados são depois guardados numa base de dados desenvolvida em MySQL. Para alimentar a estação foram usados quatro painéis solares de 6 V em paralelo com uma corrente máxima de 1320 mA. Um controlador de carga para baterias foi usado para alimentar a bateria LiPo e também enviar energia, em caso de excesso, para um conversor DC-DC do tipo *boost* que alimenta a placa OurWeather e os seus periféricos.

Em [42] foi desenvolvido um sistema de monitorização de painéis fotovoltaicos e condições meteorológicas. Foram medidas várias variáveis meteorológicas, como a irradiação solar, a temperatura e a velocidade e direção do vento. A monitorização da potência produzida por um painel fotovoltaico foi feita através da medição da tensão e corrente a um determinado momento. Para fazer a aquisição e posterior processamento foi utilizado o microcontrolador Atmega8L. Os dados foram transmitidos para um servidor na Web recorrendo a um modem 3G, este utilizando um cartão SIM.

Em [43] foi desenvolvida uma estação meteorológica autossustentável, medindo várias condições meteorológicas: temperatura, humidade relativa, pressão atmosférica, intensidade luminosa e ocorrência

de precipitação. A alimentação da estação é feita recorrendo a um módulo fotovoltaico e uma bateria. Foi utilizado um algoritmo MPPT (Maximum Power Point Tracking) para extrair o máximo de potência possível do módulo fotovoltaico. Os dados recolhidos pela estação são transmitidos para um computador através de uma comunicação sem fios utilizando transdutores de rádio frequência. Para este computador foi desenvolvida uma aplicação que processa os dados recebidos e os apresenta ao utilizador através de uma interface gráfica.

Comparativamente com os trabalhos relacionados aqui descritos, a solução implementada permite guardar os dados localmente através de um cartão de memória, sendo estes apenas transferidos para um *smartphone* quando o utilizador se aproxima da estação, através de uma conexão BLE. Isto apresenta a vantagem de não ter custos associados à transferência de dados, trazendo a desvantagem de não se poder visualizar os dados de uma forma remota. Nos trabalhos mencionados os dados ou estão a ser transferidos continuamente para um coordenador/*gateway* ou tem custos associados à transferência. Outro objetivo desta dissertação consiste em explorar formas para maximizar o débito de uma conexão BLE para transmissão de grandes quantidades de dados, sendo útil para qualquer área de aplicação em que utilize esta rede sem fios. Na secção 3.3.3 e 3.4.4 é explicada abordagem utilizada para transferir dados e maximizar o débito do BLE.

Em termos de inovação, o desenvolvimento desta estação meteorológica visa integrar diferentes amostras de tecnologias de painéis solares, medindo a potência produzida em cada uma e efetuando a correlação com os resultados das variáveis do ambiente medidas. Isto permite ajudar na escolha da tecnologia a usar num determinado local. Apesar de haver outros trabalhos em que é feita a comparação de tecnologias de painéis fotovoltaicos (Secção 2.3.3), estes usam módulos ou painéis de um tamanho considerável, não podendo assim ser integrados numa estação meteorológica compacta.

3. Desenvolvimento do sistema

3.1 Arquitetura do sistema

A Figura 25 apresenta a arquitetura do sistema implementado. Pode-se dividir esta arquitetura em duas partes principais: estação meteorológica para aquisição de dados; e aplicação em Android para visualização de dados. Os dados adquiridos pela estação são processados e armazenados localmente num cartão de memória, utilizando um módulo da família PSoC (Programmable System-on-Chip) desenvolvido pela Cypress Semiconductor. Quando solicitado pelo utilizador, os dados são transferidos para um dispositivo Android através da rede BLE, onde são armazenados numa base de dados desenvolvida em SQLite, podendo assim serem visualizados quando o utilizador necessitar.

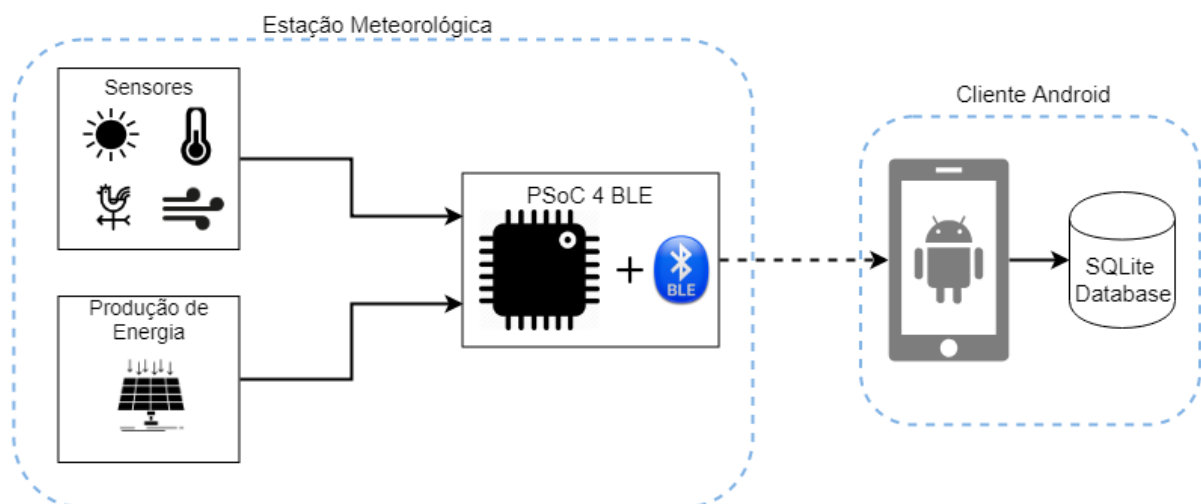


Figura 25. Arquitetura do sistema desenvolvido.

3.2 Hardware da estação

Na Figura 26 é apresentado um diagrama de blocos detalhado com todo o hardware utilizado na estação meteorológica. São usados quatro sensores para medir diversas variáveis ambientais: temperatura e humidade; velocidade do vento; direção do vento; e irradiação solar. São também utilizados três módulos fotovoltaicos de três tecnologias diferentes com o objetivo de medir a sua produção de energia. Um outro módulo fotovoltaico irá ser usado para recarregar uma bateria LiPo que irá ser usada para alimentar a estação meteorológica, quando as condições meteorológicas o permitirem. Todos os dados medidos pela estação são guardados num cartão de memória.

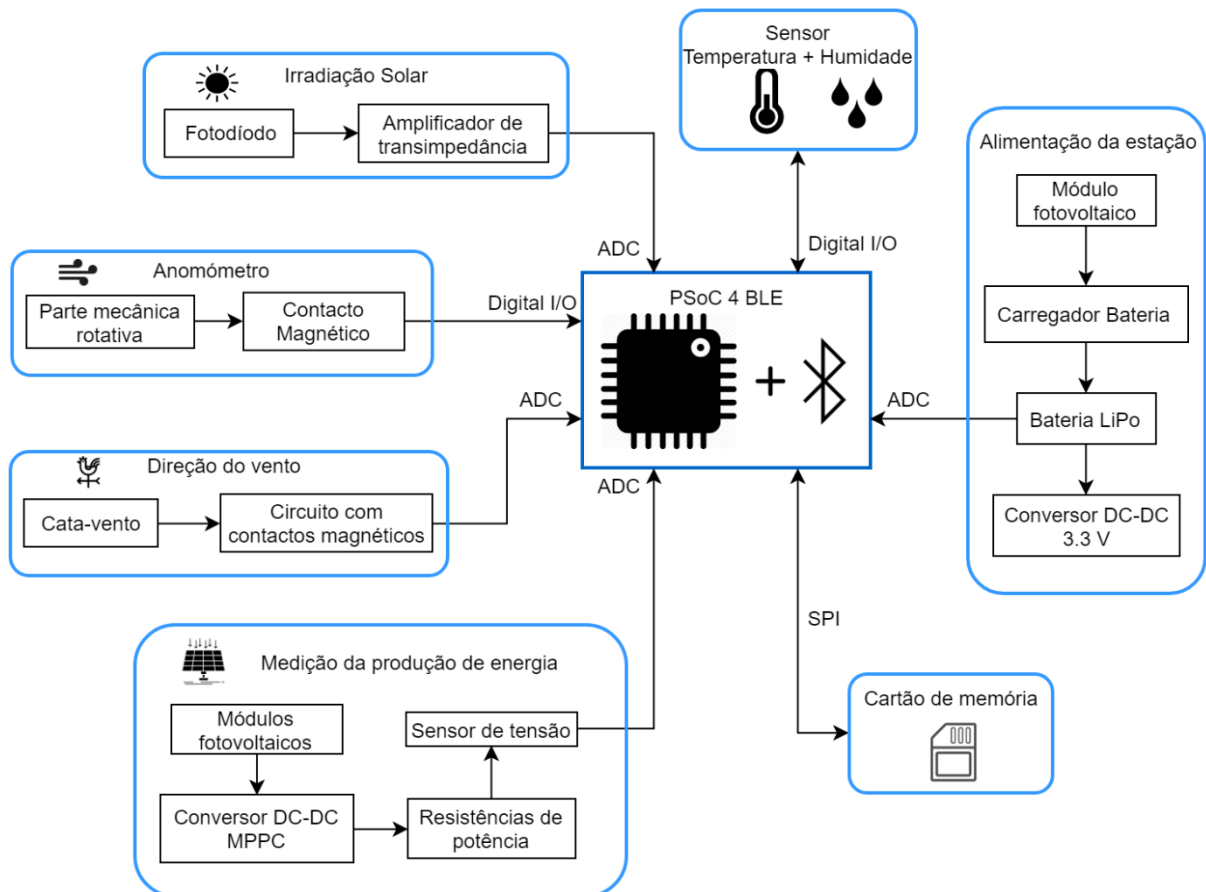


Figura 26. Diagrama de blocos do hardware da estação meteorológica.

3.2.1 PSoC 4 BLE

Para adquirir e processar os dados, bem como depois transmiti-los para o utilizador através da rede BLE foi usado um módulo PSoC 4 BLE, modelo CY8C4248LQI-BL583, sendo este obtido no kit CY8CKIT-BLE-A 4.2 [44] (Figura 27) desenvolvido pela Cypress Semiconductor. Neste kit também é disponibilizado uma placa de desenvolvimento que permite o encaixe do módulo PSoC 4 BLE, sendo utilizada durante o desenvolvimento para programação e depuração. No protótipo final foi apenas utilizado o módulo PSoC 4 BLE, através do desenvolvimento de uma placa de circuito impresso (PCB - Printed Circuit Board) com toda a eletrónica da estação meteorológica e que permite o encaixe do módulo.

Um microcontrolador PSoC é composto por um processador, blocos digitais e analógicos configuráveis e roteamento e interconexão programáveis. A disponibilidade de blocos configuráveis é o que distingue o PSoC de outros microcontroladores.



Figura 27. Kit CY8CKIT-042-BLE-A 4.2, desenvolvido pela Cypress Semiconductor [45] .

Este módulo usa o microcontrolador de 32 bits ARM Cortex-M0, tendo 256 KB de memória *flash* e 32 KB de SRAM (Static Random Access Memory). Contém quatro amplificadores operacionais programáveis e também um ADC com uma resolução máxima de 12 bits. Podem ser utilizados até 4 blocos de TCPWM (Timer Counter Pulse Width Modulation), para temporizadores, contadores ou PWM. Para comunicação série, podem ser usados até dois blocos, podendo ser estes configurados como I2C (Inter-Integrated Circuit) a mestre ou escravo, SPI (Serial Peripheral Interface) a mestre ou escravo, ou como UART (Universal Asynchronous Receiver-Transmitter). Para comunicação por radiofrequência, tem integrado um *transceiver* BLE compatível com o protocolo Bluetooth 4.2 e integra uma antena para 2.4 GHz. Na Figura 28 é apresentada uma descrição deste módulo. A alimentação do módulo pode ser feita através de qualquer um dos pinos VDDD, VDDA ou VDDR, estando estes em curto-circuito uns com os outros.



Figura 28. Descrição do módulo PSoC 4 BLE [45].

3.2.2 Temperatura e humidade

Para medir a temperatura e humidade foi usado o sensor DHT22 baseado no chip AM2302 [46]. Este sensor consegue medir temperaturas entre -40°C e 80°C e a humidade relativa de 0 a 100%. Internamente este sensor usa um termistor NTC para medir a temperatura e dois elétrodos com um substrato condutor para medir a humidade relativa.

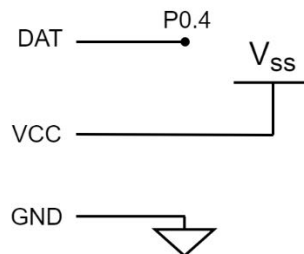
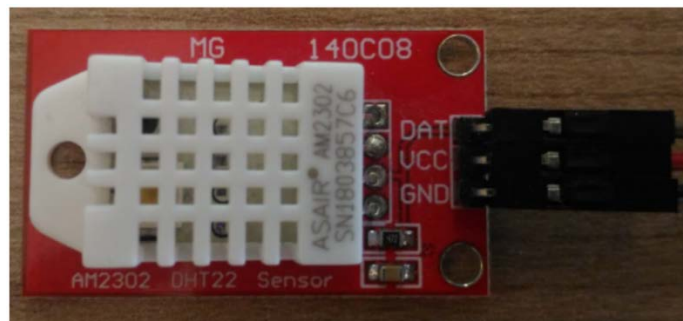


Figura 29. Sensor de temperatura e humidade DHT22.

A saída deste sensor é um sinal digital de quarenta bits: dezasseis para a temperatura, dezasseis para a humidade e oito para *checksum*. Para estabelecer a comunicação entre o sensor e o microcontrolador, este tem de inicialmente mandar um sinal de pedido de leitura para o sensor, sendo

que o sensor muda o seu estado de *standby* para *running*, enviando depois o sinal com toda a informação. Na Figura 30 é apresentado o protocolo completo para transmissão de dados. Para ser feita a transmissão, tanto o sensor como o microcontrolador têm de obedecer a um conjunto de tempos mínimos e máximos em que o sinal se encontra no estado Vcc ou Gnd. O sinal a tracejado corresponde à transferência dos valores da temperatura e humidade. Para representar um bit de valor "0" o sinal tem de estar no estado Vcc durante 26-28 μ s e para representar um bit de valor "1" o sinal tem de estar no estado Vcc durante 70 μ s. Antes de enviar cada bit o sensor espera 50 μ s no estado Gnd. Isto é repetido para todos os quarenta bits de dados.

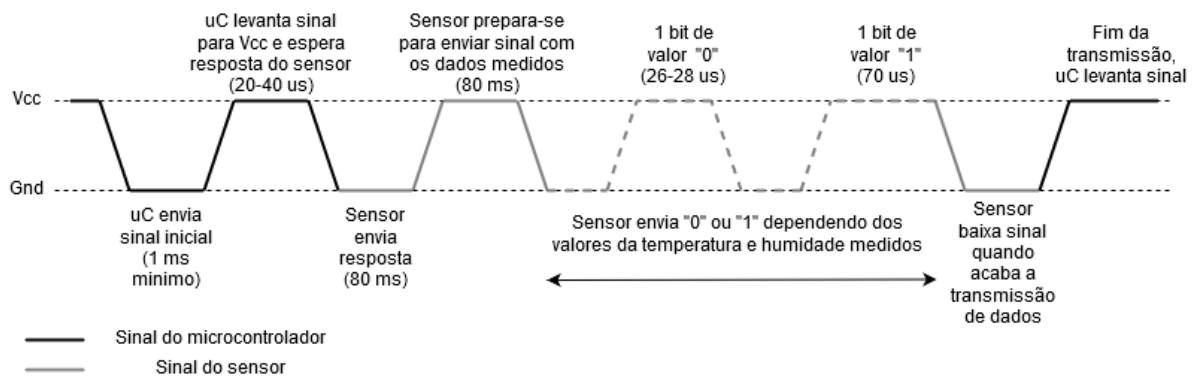


Figura 30. Protocolo de comunicação entre sensor DHT22 e o microcontrolador.

3.2.3 Velocidade e direção do vento

Para medir a velocidade de vento é usado um anemómetro. Este dispositivo é constituído por uma parte giratória ao qual está adaptado um íman. Ao rodar, a força magnética produzida por este íman é detetada por um interruptor magnético do tipo Reed switch (S1). A saída do sensor é conectada a um pino digital da placa de desenvolvimento sendo depois gerada uma interrupção externa. A velocidade do vento é calculada através do tempo necessário que o anemómetro necessita para completar uma volta, sendo depois calculado um valor em rotações por minuto (RPM). Na Figura 31 é apresentado o circuito do anemómetro. É usada uma resistência *pull-down* para garantir que a entrada do pino no microcontrolador não esteja a flutuar.

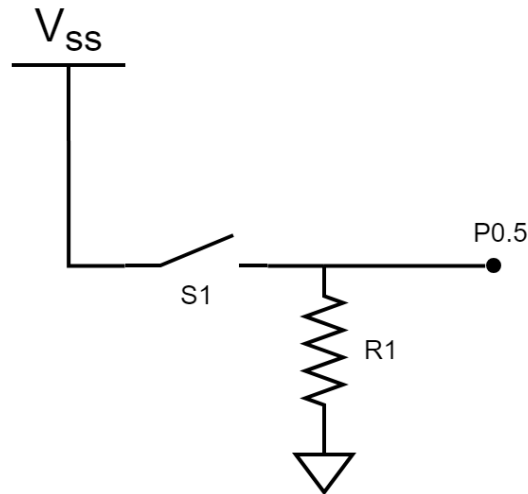


Figura 31. Circuito para medição da velocidade do vento, composto por um interruptor magnético (S1) e uma resistência pull-down (R1).

Devido ao posicionamento do íman na parte giratória do anemómetro, o contacto magnético é ativado duas vezes durante uma volta. Este tempo é depois medido (em microssegundos) através de um contador no microcontrolador, usando um pino digital em modo de interrupção externa. Através da equação (2) pode-se calcular as rotações por minuto (RPM).

$$RPM = \frac{60000000 \mu s}{\text{tempo para completar uma volta}} \quad (2)$$

O valor das rotações por minuto é depois convertido para quilómetros por hora (km/h). Para isso foi medido o raio (r) do anemómetro, sendo de 8 cm. Através da equação seguinte é transformado as RPM em km/h:

$$\frac{km}{h} = RPM \times \frac{1}{60} \times 2\pi \times r \times 3,6 \quad (3)$$

Por exemplo para 50 RPM, o equivalente em km/h será:

$$\frac{km}{h} = 50 \times \frac{1}{60} \times 2\pi \times 0,08 \times 3,6 = 1,51$$

Na Figura 32 é apresentado o anemómetro e cata-vento eletrónico usados na estação. O anemómetro está ligado ao cata-vento através de um conector RJ11. A saída do cata-vento leva os sinais dos dois sensores, utilizando também um conector RJ11.



Figura 32. Cata-vento eletrônico e anemómetro utilizados.

A direção do vento é determinada usando um cata-vento eletrônico. Este também usa um íman adaptado ao seu eixo rotatório, em conjunto com um circuito constituído por oito interruptores magnéticos do tipo *reed switch*, correspondendo cada um deles a uma direção possível. Os interruptores magnéticos são ativados pelo campo magnético produzido pelo íman. Dependendo da posição do conjunto eixo rotatório mais íman, é ativado um (ou dois) dos oito interruptores. Cada interruptor está em série com uma resistência de valor diferente, que depois irá fazer um divisor de tensão com outra resistência colocada na saída. Na Figura 33 é apresentado o diagrama esquemático do circuito para deteção da direção do vento. Fisicamente este circuito está representado numa PCB (Printed Circuit Board) com o formato de uma rosa dos ventos. Caso o conjunto do eixo rotatório mais íman esteja entre dois interruptores, estes estarão os dois ativados, fazendo o paralelo entre as duas resistências associadas a esses interruptores.

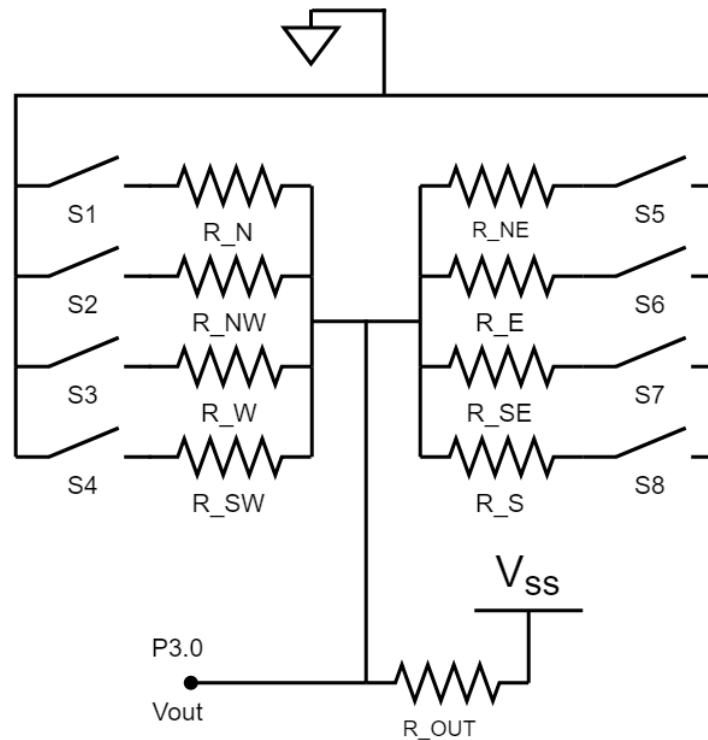


Figura 33. Circuito para detecção da direção do vento. S1 a S8 representa os interruptores magnéticos.

O valor de saída do divisor de tensão (V_{out}) é obtido com o valor da resistência da direção atual (R_{DIR}), o valor da resistência de saída (R_{Out}) e o valor de alimentação do circuito (V_{SS}). Esta tensão é depois adquirida pelo ADC do microcontrolador, sendo feita a sua correspondência com uma direção do vento. A equação seguinte apresenta como é feito o cálculo do valor de saída.

$$V_{out} = \frac{R_{DIR}}{R_{Dir} + R_{Out}} \times V_{SS} \quad (4)$$

Na Tabela 3 são apresentadas todas as direções possíveis, com os valores das resistências e valor da tensão de saída obtido, com $R_{out} = 10 \text{ k}\Omega$ e $V = 3,3 \text{ V}$. Por exemplo, quando a direção do vento se encontra para Norte, o valor da resistência ativada será de $33 \text{ k}\Omega$. Substituindo na equação (4), obtém-se:

$$V_{out} = \frac{33 \text{ k}\Omega}{33 \text{ k}\Omega + 10 \text{ k}\Omega} \times 3.3 = 2.53 \text{ V}$$

Tabela 3. Correspondência entre a direção do vento e tensão de saída do circuito.

Direção	Resistência (k Ω)	Tensão de saída (Volt)
N	$R_N = 33$	2,53
NNE	$R_N // R_{NE} = 6,57$	1,31
NE	$R_{NE} = 8,2$	1,49
ENE	$R_{NE} // R_E = 0,891$	0,27
E	$R_E = 1$	0,3
ESE	$R_E // R_{SE} = 0,688$	0,32
SE	$R_{SE} = 2,2$	0,60
SSE	$R_{SE} // R_S = 1,41$	0,41
S	$R_S = 3,9$	0,92
SSO	$R_S // R_{SW} = 3,14$	0,79
SO	$R_{SW} = 16$	2,03
OSO	$R_{SW} // R_W = 14,12$	1,93
O	$R_W = 120$	3,05
ONO	$R_W // R_{NW} = 42,12$	2,67
NO	$R_{NW} = 64,9$	3,15
NNO	$R_{NW} // R_N = 21,88$	2,26

Para calibrar a posição inicial da direção do vento do cata-vento, foi utilizado uma bussola, apresentada na Figura 34.



Figura 34. Bussola usada para calibrar o cata-vento.

3.2.4 Irradiação solar

Irradiação solar é um dos fatores ambientais mais importantes para a determinação da potência produzida por painéis fotovoltaicos. O espectro de radiação solar que atinge a superfície da Terra tem comprimentos de onda entre 300 nm e 2800 nm, mas os painéis fotovoltaicos conseguem absorver ondas apenas no espectro do visível (380 a 740 nm) [47]. Tendo em conta este fator, e também o custo e a disponibilidade, foi escolhido o fotodiodo BPW34 [48]. Este sensor tem uma boa gama no seu espectro de sensibilidade, entre 430 nm e 1100 nm. O sinal produzido pelo fotodiodo é acondicionado pelo circuito apresentado na Figura 35. Este circuito é conhecido como circuito de transimpedância, em que a corrente produzida pelo fotodiodo é transformada numa tensão proporcional na saída do amplificador. O valor de R1 (680 Ω) foi determinado experimentalmente de forma que o valor da saída nunca exceda o valor máximo da gama de entrada do ADC da placa (3,3 V), dado que no *datasheet* do fotodiodo BPW34 não é fornecida a informação do valor da corrente inversa em função da irradiação. É usado um condensador de *feedback* (C2) em paralelo com R1, para melhorar a estabilidade [49]. Na saída é usado um filtro passa-baixo com frequência de corte de 10 Hz para remover possível ruído proveniente do ambiente. Foi usado o amplificador operacional disponível no módulo PSoC 4 BLE.

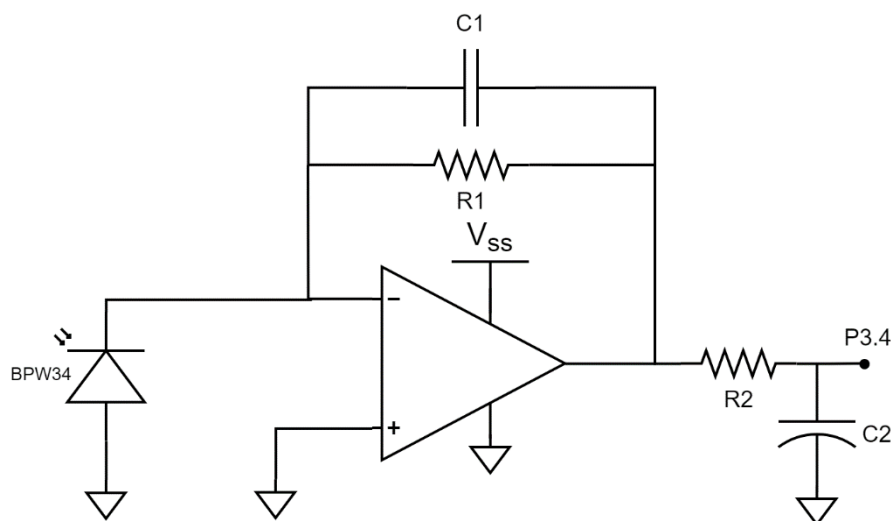


Figura 35. Circuito de acondicionamento usando o fotodiodo BPW34.

Para fazer a calibração entre o valor em tensão na saída do circuito e um valor da irradiação em W/m^2 foi usado o piranómetro PYR-P [50] desenvolvido pela Apogee, representado na Figura 36. Este sensor tem uma saída analógica linear de 1 mV por $5 W/m^2$, com um valor máximo de 220 mV ($1100 W/m^2$). Ao longo de um dia com céu limpo, foi medido o valor da saída do piranómetro e o valor da saída do circuito de acondicionamento do fotodiodo BPW34.



Figura 36. Piranómetro PYR-P utilizado na calibração.

Os resultados obtidos estão expressos na Tabela 4.

Tabela 4. Calibração do circuito de acondicionamento e a irradiação solar.

Saída do circuito de acondicionamento (V)	Saída piranómetro (mV)	Irradiação (W/m ²)
0,167	14	70
0,211	16	80
0,308	27	135
0,723	37	185
0,905	50	250
1,22	75	375
1,38	87	435
1,56	108	540
1,66	114	570
1,9	129	645
2,02	144	720
2,14	157	785
2,23	163	815

Na Figura 37 á apresentado graficamente o resultado da calibração, usando o método de regressão linear, através do qual foram obtidos os coeficientes angular e linear utilizados na equação (5).

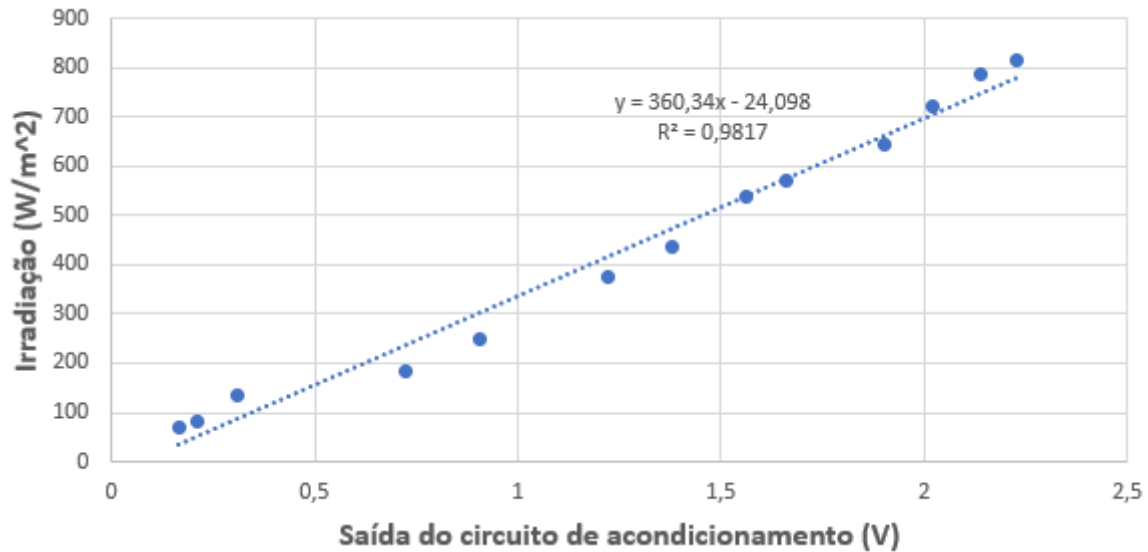


Figura 37. Gráfico que relaciona a tensão de saída do circuito e a irradiação solar.

Através da equação (5) obtida, é possível calcular o valor da irradiação solar em função do valor de tensão na saída (V_{out}) do circuito de condicionamento do sensor BPW34. Este cálculo é efetuado pelo microcontrolador após obter o valor de tensão no ADC.

$$Irradiação\ Solar = 360,34 \times V_{out} - 24,098 \quad (5)$$

3.2.5 Medição e maximização da produção de energia

Na estação foram instalados três módulos fotovoltaicos de três tecnologias diferentes: monocristalino, policristalino e amorfo. Para efeitos de comparação, os módulos adquiridos teriam de ter aproximadamente a mesma área. Infelizmente isto não foi possível para o módulo fotovoltaico da tecnologia de silício amorfo. Na Tabela 5 são apresentados alguns dados dos módulos fotovoltaicos adquiridos. Designa-se por área útil a área total das células fotovoltaicos dos módulos.

Tabela 5. Informação sobre os módulos fotovoltaicos adquiridos.

	Monocristalino	Policristalino	Amorfo flexível	Amorfo flexível (2 módulos em série)
Potência máxima	1 W	1 W	0,3 W	0,6 W
Corrente de curto circuito	250 mA	250 mA	300 mA	300 mA
Corrente de funcionamento	200 mA	200 mA	200 mA	200 mA
Tensão de circuito aberto	6 V	6 V	2 V	4,15 V
Tensão de funcionamento	5 V	5 V	1,5 V	3 V
Tamanho útil	85 x 60 mm	90 x 62 mm	93 x 36 mm	93 x 72 mm
Área útil	5100 mm ²	5580 mm ²	3348 mm ²	6696 mm ²

Dada a pequena área dos módulos fotovoltaicos de silício amorfo em comparação com os de silício cristalino, foram colocados dois destes módulos em série. Na Figura 38 são apresentados os módulos fotovoltaicos de diferentes tecnologias utilizados para medir a produção de energia.

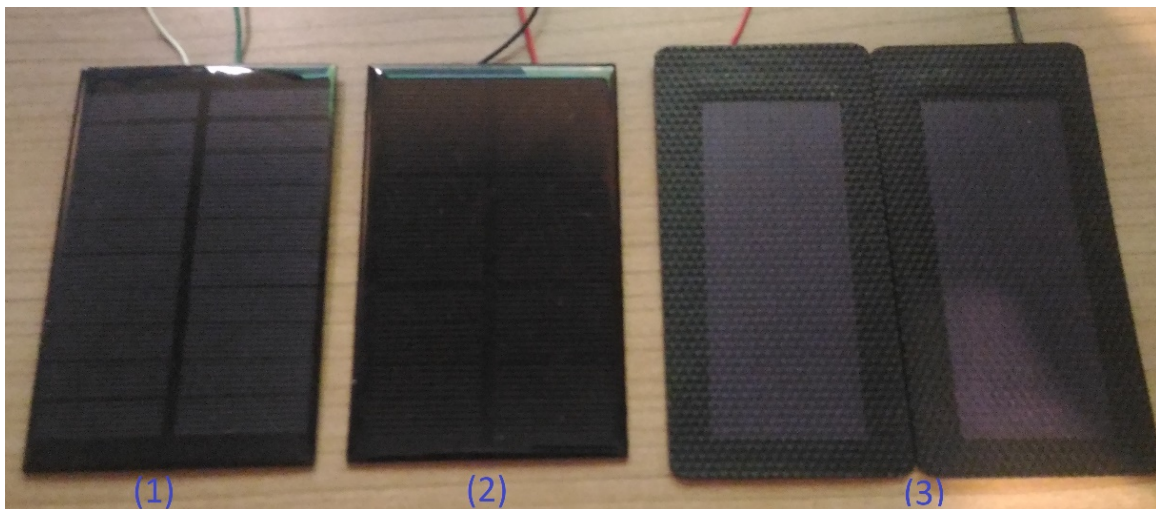


Figura 38. Módulos fotovoltaicos utilizados: (1) – Policristalino, (2) – Monocristalino, (3) – Amorfo (dois em série).

De modo a maximizar a extração de energia por parte dos módulos fotovoltaicos, foi usado o conversor DC-DC em topologia *buck-boost* LTC3129 [51] com MPPC (Maximum Power Point Control) integrado. Este conversor consegue atingir uma corrente de saída máxima de 200 mA e apresenta uma arquitetura de comutação de muito baixo ruído. Dependendo do modo de operação, consegue ter uma corrente quiescente de apenas 1,3 μ A, sendo indicado para sistemas alimentados a bateria. Pode ter

uma tensão de entrada entre 2,42 V e 15 V e a sua tensão de saída pode ser programada para um valor entre 1,4 V e 15,75 V.

Apesar deste conversor não ter MPPT integrado, tem a técnica chamada de MPPC. Com MPPC o conversor vai procurar ter a tensão de entrada igual a uma tensão de referência programada. Para fazer isto é ajustada a corrente no indutor externo que será ligado ao conversor, mantendo uma tensão mínima na entrada quando usando fontes de alta impedância, como no caso de módulos fotovoltaicos. Isto maximiza a potência transferida e previne a entrada (V_{in}) de baixar muito quando é adicionada uma carga. É recomendado que o valor do condensador na entrada do conversor seja maior que 22 μF , caso se use o MPPC do conversor. A programação da tensão de MPPC pode ser feita conectando um divisor de tensão externo ao pino MPPC. As resistências a usar podem ser calculadas a partir da seguinte equação:

$$V_{in}(MPPC) = 1,175 \times \left(1 + \frac{R5}{R6}\right) \quad (6)$$

A programação da tensão de saída também é feita recorrendo ao uso de um divisor de tensão entre o pino V_{out} e o pino FB. As resistências a utilizar são calculadas através da seguinte equação:

$$V_{out} = 1,175 \times \left(1 + \frac{R1}{R2}\right) \quad (7)$$

É necessário também a utilização de um indutor externo. De acordo com o *datasheet* do conversor LTC3129 [51], para tensões de entrada e saída menores que 8 V, o valor deste indutor pode ser entre 4,7 μH e 6,8 μH . Foi escolhido um indutor com o valor de 6,8 μH . Este indutor é colocado entre os pinos SW1 e SW2, em conjunto com dois condensadores de 2,2 nF entre os pinos SW e BST. Na Figura 39 é apresentado o diagrama esquemático do circuito desenvolvido com o conversor LTC3129.

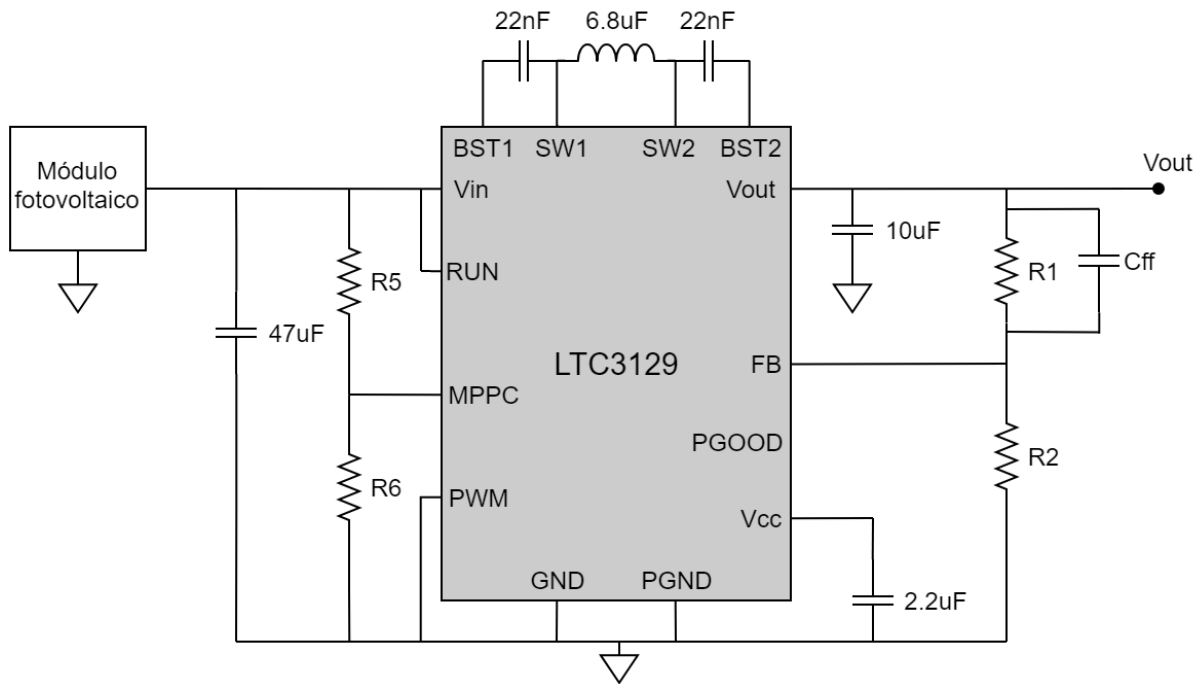


Figura 39. Circuito desenvolvido com o conversor LTC3129 para maximização da extração de energia dos módulos fotovoltaicos.

É recomendado um condensador de *feed-forward* (C_{ff}) em paralelo com a resistência R_1 , de modo a reduzir o *ripple* de saída e aumentar a resposta transiente. O valor para este condensador tem de obedecer à seguinte equação:

$$C_{ff} \text{ (pF)} = \frac{66}{R_1 \text{ (M}\Omega\text{)}} \quad (8)$$

Foi implementado um circuito com o conversor LTC3129 para cada tecnologia de módulos fotovoltaicos usados. Na saída de cada conversor foi ligada a uma resistência (RL) de potência de valor conhecido. Na Figura 40 está representado o esquemático do sistema para medição da produção de energia. Para calcular a potência (P) produzida por cada módulo, foi medida a tensão (U) em cada uma destas resistências. A potência produzida por cada painel pode ser então calculada através da seguinte equação:

$$P = \frac{U^2}{RL} \quad (9)$$

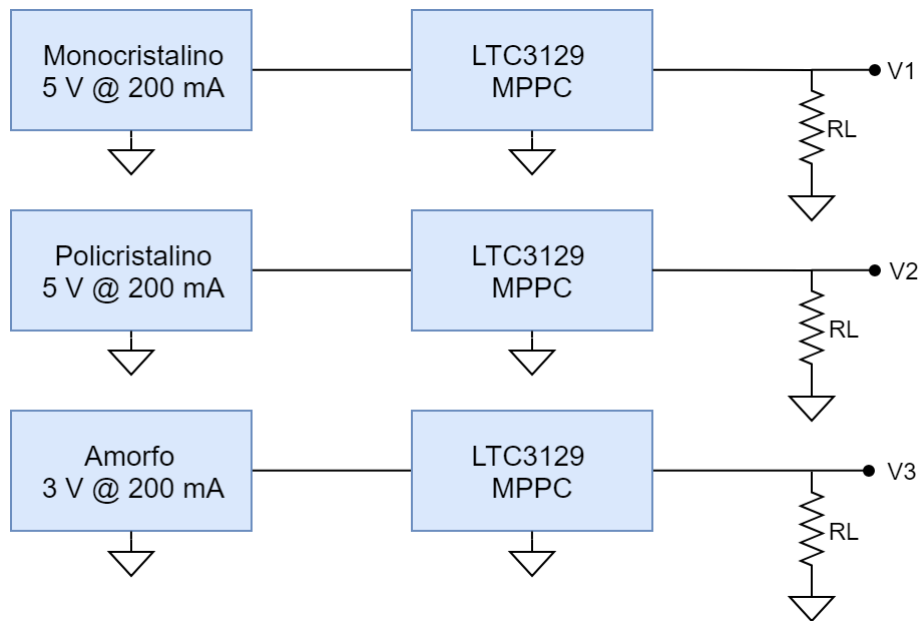


Figura 40. Esquemático da medição da produção de energia dos três módulos fotovoltaicos.

A tensão nestas resistências pode ser medida usando simplesmente um divisor de tensão, conforme representado na Figura 41. Este terá de ser dimensionado tendo em conta a tensão máxima permitida no ADC do microcontrolador (3,3 V). Terá de se ter em consideração que quanto menor for o valor destas resistências, maior será o impacto deste divisor de tensão na medição da potência. O valor das resistências para cada divisor de tensão terá de ser calculado de acordo com a tensão máxima esperada entre a saída do conversor e a resistência de carga (V_N), de acordo com a seguinte equação:

$$V_{ADC} = V_N \cdot \frac{R2}{R1 + R2} \quad (10)$$

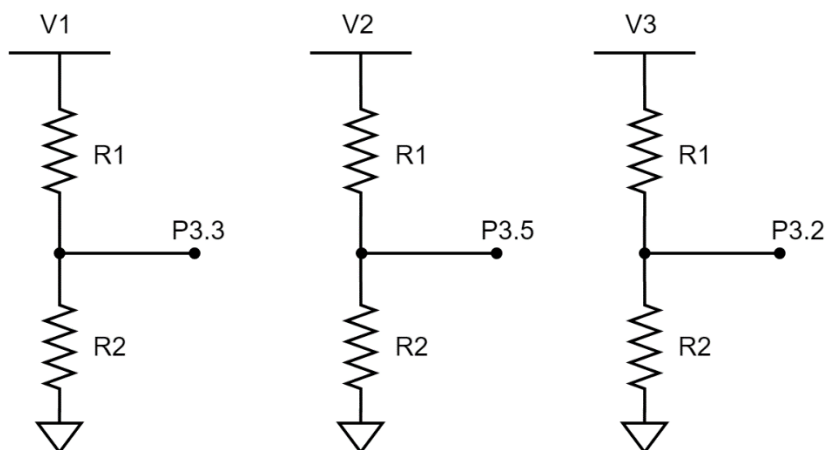


Figura 41. Divisor de tensão para medição da tensão entre a saída do conversor LTC3129 e a carga.

3.2.6 Alimentação da estação

Na Figura 42 é apresentado o diagrama de blocos do circuito de alimentação da estação.

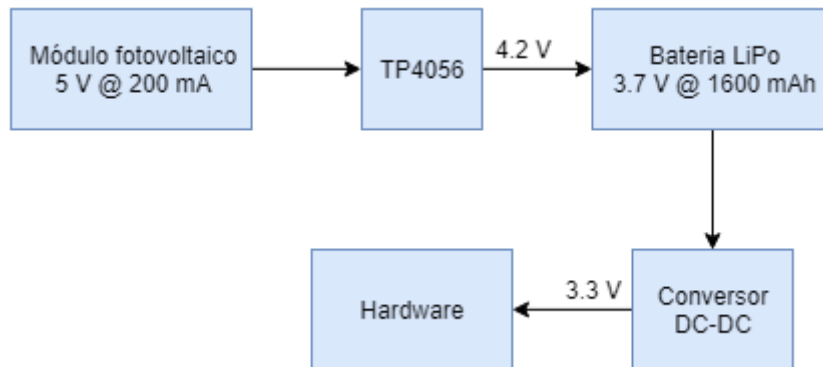


Figura 42. Diagrama de blocos do circuito de alimentação da estação.

Para alimentar todo o hardware da estação foi usada uma bateria de polímero de lítio (LiPo - Lithium Polymer) de 3,7 V com uma capacidade de 1600 mAh [52]. Esta bateria já vem com um circuito de proteção integrado, desligando o carregamento quando a bateria estiver na sua capacidade total, não deixando a bateria atingir um valor demasiado baixo de tensão (abaixo de 2,75 V) e também protegendo contra uma corrente com valor acima daquele desejado.

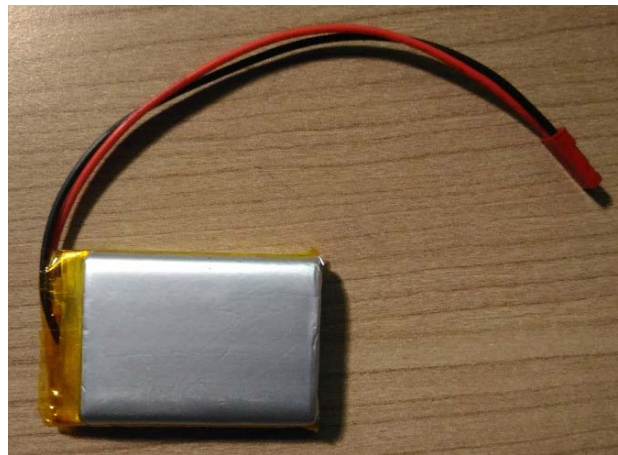


Figura 43. Bateria LiPo de 3.7 V com capacidade de 1600 mAh.

Para recarregar a bateria, caso as condições meteorológicas o permitirem, foi usado um módulo fotovoltaico extra além daqueles usados para a medição da produção de energia. O módulo fotovoltaico escolhido é baseado na tecnologia de silício monocristalino, com uma potência máxima de 1 W. Este módulo foi ligado diretamente ao gestor de carregamento da bateria LiPo.

Para gerir o carregamento da bateria foi utilizado um módulo baseado no circuito integrado TP4056 [53], representado na Figura 44. Este módulo produz uma tensão de alimentação fixa de 4,2 V e a máxima corrente de carga pode ser programada externamente usando apenas uma resistência. Por padrão esta resistência (R3) tem o valor de 1,2 k Ω , correspondendo a uma corrente máxima de carga de 1000 mA, mas pode ser modificado retirando-se a resistência e acrescentando uma de um valor maior para uma corrente mais pequena. O carregamento é automaticamente terminado quando a corrente de carga diminui para um décimo da máxima corrente programada, após atingir a tensão desejada de carga da bateria. Usando a resistência padrão, o carregamento seria terminado quando a corrente de entrada diminuísse para um valor menor que 100 mA. Sendo assim, através da informação do *datasheet*, esta resistência foi modificada para uma de 4,7 k Ω , correspondendo a uma corrente máxima de aproximadamente 260 mA, maximizando o aproveitamento da potência produzida pelo módulo fotovoltaico.

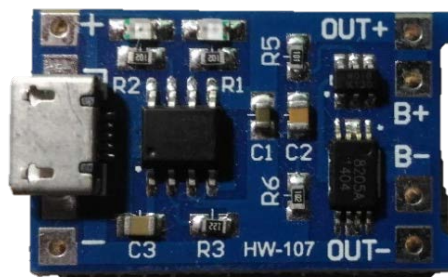


Figura 44. Módulo para carregamento de baterias LiPo, baseado no circuito integrado TP4056.

Foi escolhida uma tensão de alimentação de 3,3 V para os sensores e para o módulo PSoC 4 BLE. Dado que a tensão de uma bateria LiPo pode atingir valores próximos da tensão de alimentação que se requer (3,3 V), foi utilizado um regulador de tensão com topologia *buck-boost*. O regulador utilizado foi o S7V8A [54] da Pololu Electronics (Figura 45).



Figura 45. Regulador de tensão *step-up/step-down* S7V8A [54].

Para medir a capacidade da bateria seria necessário saber a quantidade de corrente que entra e sai da bateria. Para isso é normalmente utilizado um Coulomb *counter*, como por exemplo o circuito integrado BQ26220 [55]. De forma a simplificar este processo, o mesmo poderá ser feito apenas através do valor da tensão da bateria. Apesar de este método não ser muito preciso, dado que a tensão da bateria não é linear com a capacidade, é possível na mesma obter uma relação entre os dois valores caso a tensão da bateria não atinja valores abaixo de 3.6 V. Foi feita uma linearização da tensão da bateria com a sua capacidade, com 3.6 V correspondendo a 5 % e 4.2 V correspondendo a 100 %. Para que o valor medido da tensão da bateria não seja afetado pelo módulo fotovoltaico, foi utilizado o MOSFET de nível lógico IRLZ44N [56] para desligar o módulo TP4056 da bateria, quando é feita a medição da tensão da bateria pelo ADC, ligando o módulo novamente após concluída a medição. Este sistema está representado na Figura 46.

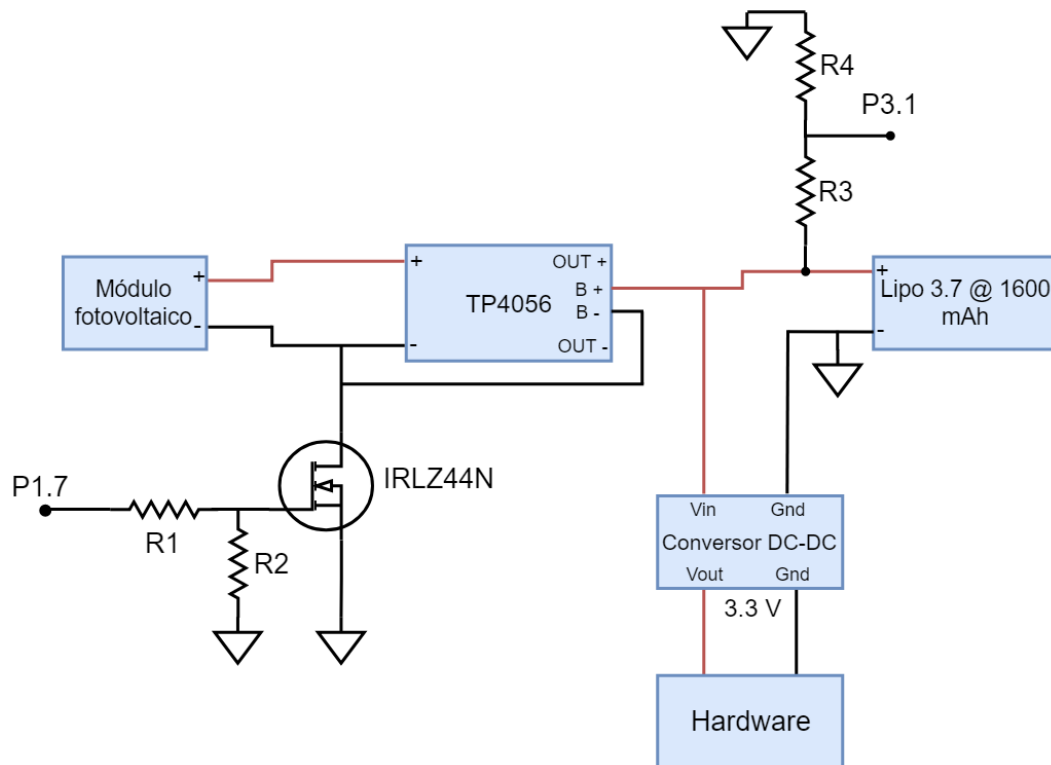


Figura 46. Sistema de recarregamento da bateria e medição da capacidade atual da bateria.

3.2.7 Cartão de memória e módulo de comunicação

Para armazenar os dados recolhidos pelos sensores e pelos painéis solares foi utilizado um cartão de memória, nomeadamente o SDHC 16 GB Adata Class 10 USH-I (Figura 47). Para ser feita a leitura e escrita no cartão de memória por parte do microcontrolador, foi utilizado um módulo dedicado para tal,

sendo a comunicação feita pelo protocolo SPI. Na Figura 47 é apresentado o cartão de memória e módulo utilizado na estação.

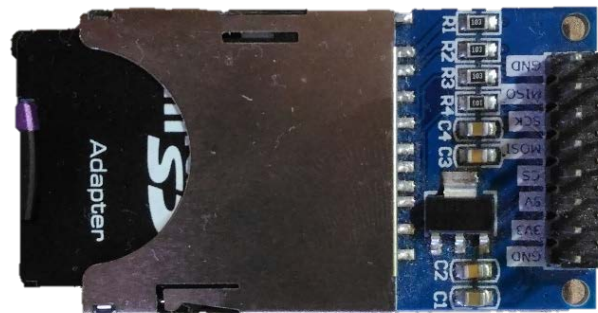


Figura 47. Cartão de memória e leitor usados na estação.

3.2.8 PCB para maximização da produção de energia

Para a eletrônica de maximização da produção de energia foi desenvolvida uma PCB (Printed Circuit Board) usando o programa PADS. Na Figura 48 é apresentado a *layout* da PCB desenvolvida, composta por três conversores LTC3129, cada um de acordo com esquemático da secção 3.2.5.

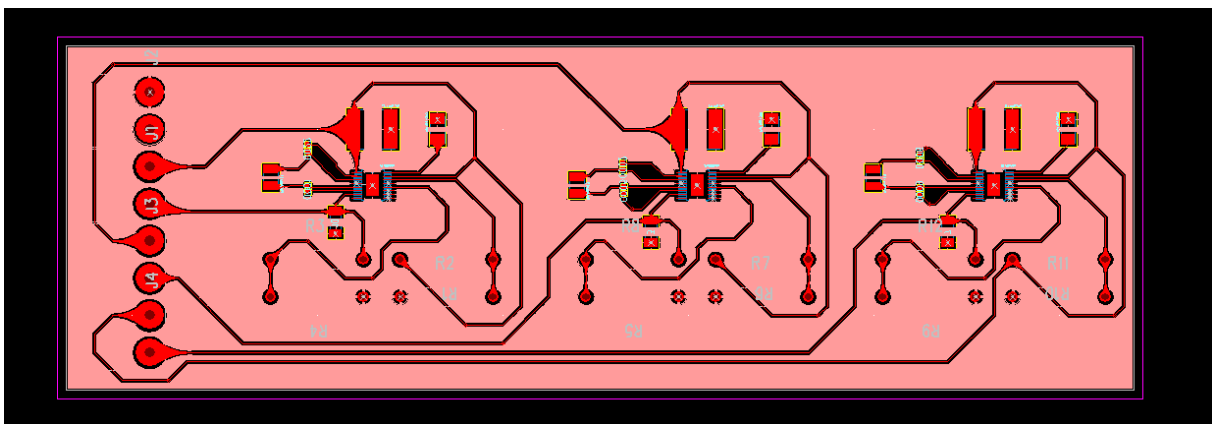


Figura 48. Layout da PCB desenvolvida para medição da produção de energia.

3.2.9 Desenho de suporte para impressão 3D

De modo a suportar o anemómetro e cata-vento eletrónico foi desenhado em 3D um suporte. Este desenho foi feito utilizando o website TinkerCad [57], que permite depois a conversão para ficheiro STL (de "*stereolithography*") para utilizar na impressão 3D. Na Figura 49 é apresentado o desenho do suporte para o anemómetro e cata-vento eletrónico.

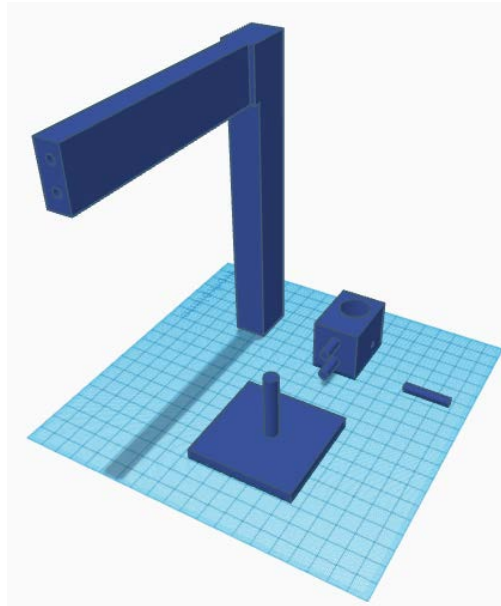


Figura 49. Desenho do suporte do anemômetro e cata-vento eletrônico.

3.3 Aquisição, armazenamento e transferência de dados

Para o desenvolvimento do *firmware* de aquisição, processamento, armazenamento e transmissão de dados via BLE para o módulo PSoC 4 BLE, foi usado o IDE (Integrated Development Environment) PSoC Creator, utilizando a linguagem de programação C. Este IDE dispõe de um ambiente de programação visual para configurar periféricos digitais e analógicos, através de uma janela de *design* do projeto desenvolvido. Os periféricos estão organizados em componentes que podem ser escolhidos através de um catálogo, colocando-os no *design* da forma necessária.

3.3.1 Aquisição dos dados dos sensores e módulos fotovoltaicos

Para adquirir os dados dos sensores e painéis solares foi necessário a programação de vários periféricos. Na Figura 50 é apresentado o IDE PSoC Creator onde se pode visualizar todos os componentes usados para aquisição, processamento, armazenamento e transmissão de dados.

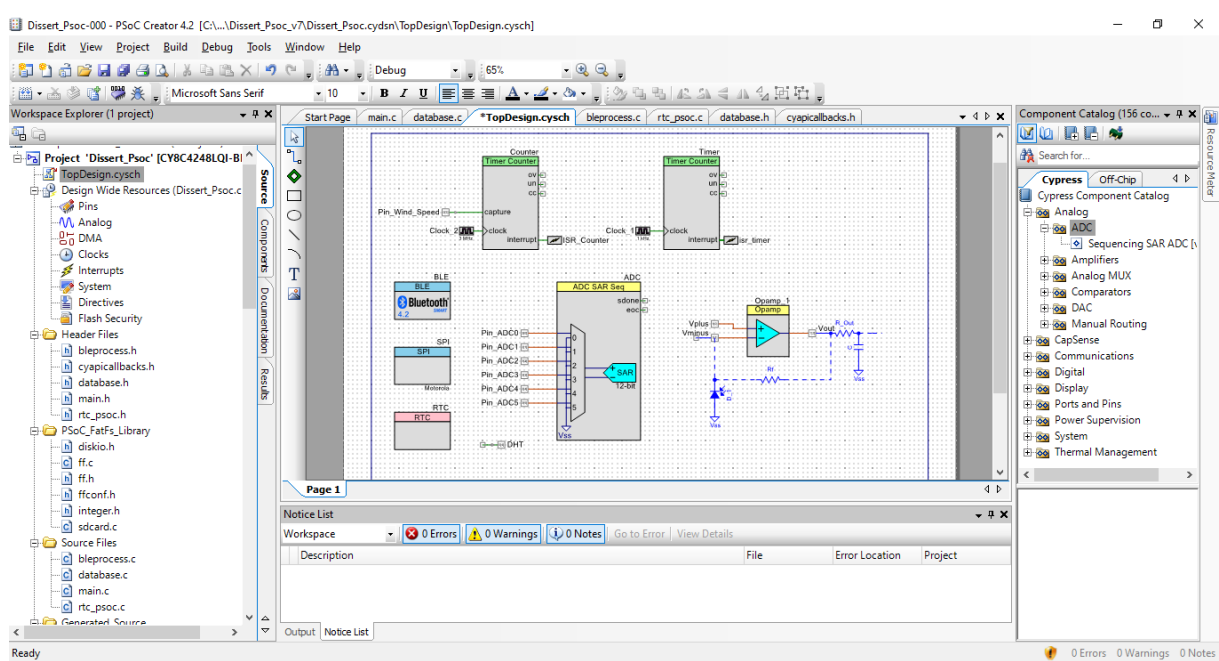


Figura 50. IDE PSoC Creator utilizado para desenvolver o *firmware* para o módulo PSoc 4 BLE.

Foram criados três arquivos .c para organizar o *firmware* desenvolvido, além do arquivo main.c que existe por padrão. O arquivo database.c gere tudo relacionado com a escrita e leitura de dados no cartão de memória. O arquivo bleprocess.c contém a *callback* para os eventos do BLE e outras funções relacionadas com o BLE. O arquivo rtc_psoc.c contém as funções relacionadas com o RTC do módulo. As funções relacionadas com a aquisição e processamento de dados encontram-se no arquivo main.c.

Foi necessário configurar de quanto em quanto tempo estes dados são adquiridos, ou seja, o período de amostragem. Neste sentido, foi programado um temporizador em modo interrupção (Figura 51), sendo esta disparada de 100 em 100 ms. Dependendo do período de amostragem escolhido pelo utilizador na aplicação Android, a interrupção vai disparar várias vezes até atingir esse valor. Por exemplo para um período de amostragem de 30 segundos, a interrupção terá de disparar 300 vezes. Através deste método, existe uma ampla gama de escolha do período de amostragem, desde 0,1 segundos até 6553 segundos. Ao atingir o valor pretendido, uma *flag* é ativada dentro da ISR (Interrupt Service Routine) da interrupção. Esta *flag* está constantemente a ser verificada no *loop* principal da função main, e, quando ativada, os dados são guardados no ficheiro de texto respetivo. Usando o mesmo temporizador foi utilizada uma outra *flag* que é ativada de um em um segundo, para adquirir os dados de modo a fazer uma média destes quando chegar o período de amostragem definido pelo utilizador.

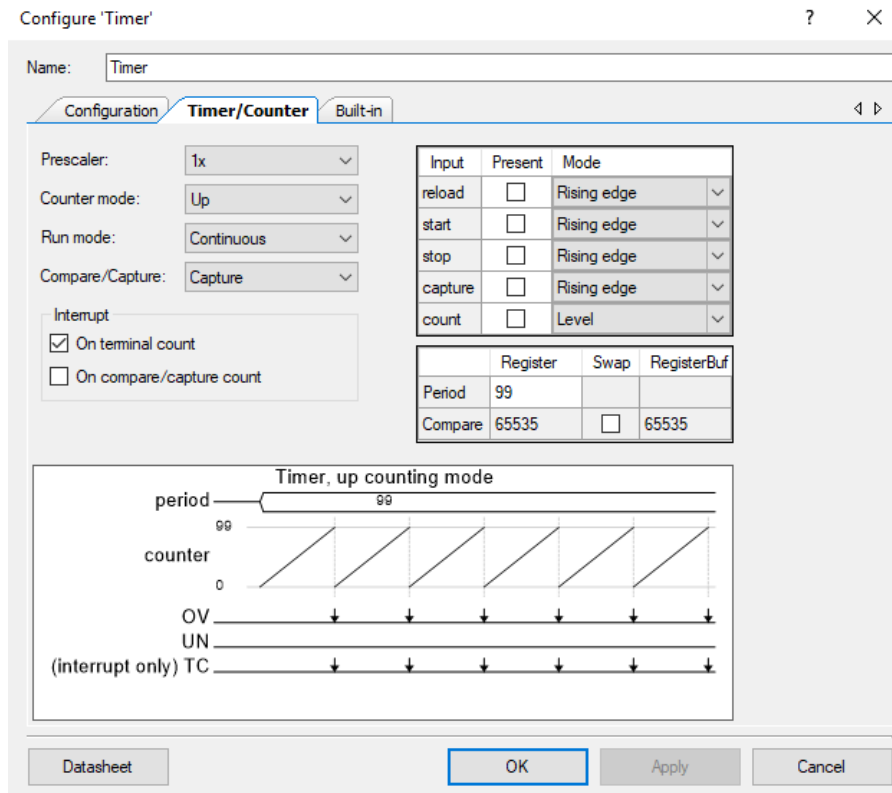


Figura 51. Configuração do temporizador para um tempo de 100 ms.

Alguns dos sensores utilizados têm uma saída analógica, sendo assim necessário o uso do ADC da placa. O ADC foi programado utilizando a sua resolução máxima (12 bits), tensão de referência de 3.3 V e um *clock* de 4 kHz (Figura 52). Foram programados seis canais do ADC para acomodar vários sensores. Antes de adquirir o valor final do ADC, este faz uma média de 128 valores, devido a pequenas flutuações que possam existir.

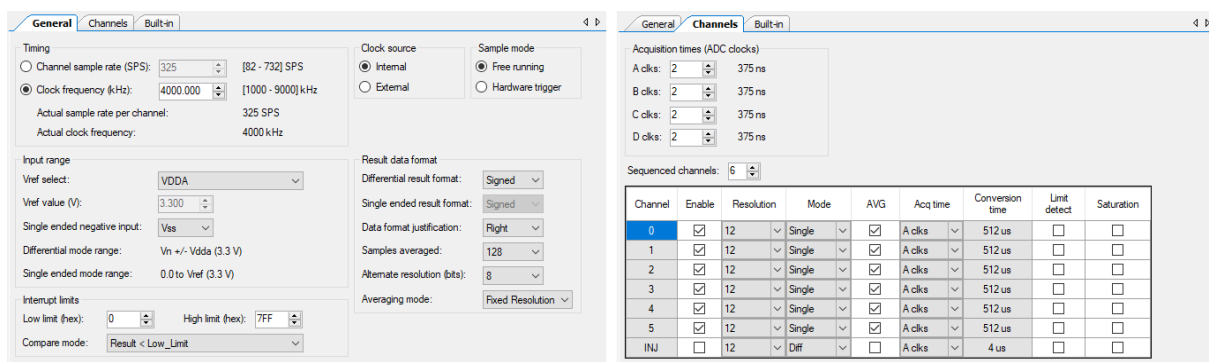


Figura 52. Configuração do ADC do módulo PSoC 4 BLE.

Foi criada uma função que usa como argumento de entrada um canal do ADC. O fluxograma que explica o funcionamento desta função está representado na Figura 53. Dependendo do canal escolhido,

é utilizado um *switch case* para processar os dados recolhidos pelo ADC. Depois de processado, o valor é guardado na variável correspondente.

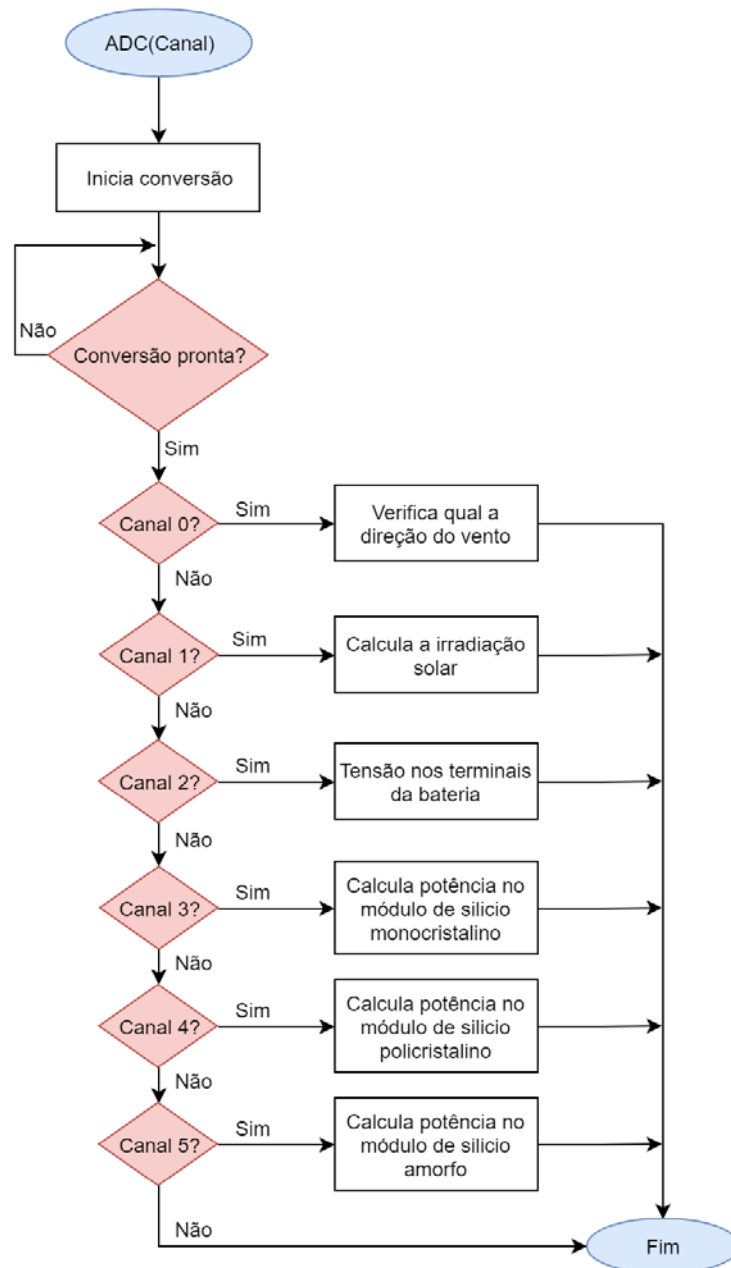


Figura 53. Fluxograma da função que processa os dados obtidos no ADC.

Dada a necessidade de um amplificador operacional para a medição da irradiação solar, foi utilizado um dos quatro amplificadores operacionais disponíveis no módulo PSoC 4 BLE (Figura 54), minimizando assim o uso de hardware extra. Este amplificador permite entrada e saída *rail-to-rail* e tem baixa impedância de saída. Foi também necessário associar três pinos a este componente, sendo estes programados para trabalhar com um sinal analógico. Na secção 3.2.4 é representado o esquemático do amplificador operacional e dos componentes extra para o seu funcionamento.

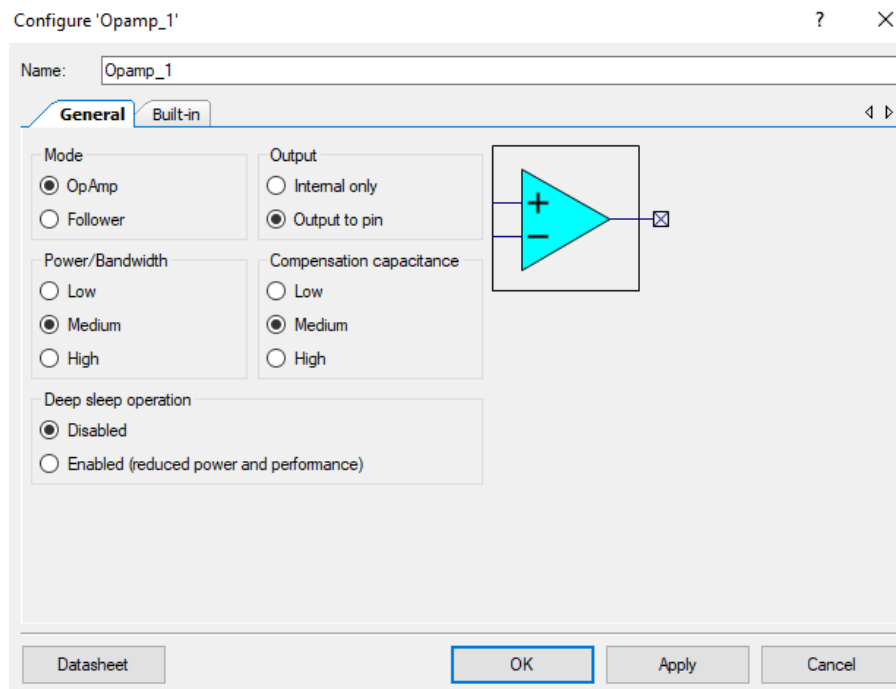


Figura 54. Configuração do amplificador operacional no IDE PSoC Creator.

Para a medição da velocidade do vento, foi programado um contador em modo interrupção (Figura 55), associando também um pino digital de entrada [58]. Este pino está ligado à saída do circuito de medição da velocidade do vento, constituído por um interruptor magnético e uma resistência de *pull-down*. A cada volta completa pelo anemómetro, o interruptor magnético é ativado duas vezes. O tempo que o anemómetro necessita para completar uma volta é medido, sendo calculado o valor das rotações por minuto (RPM) e em seguida o valor correspondente em km/h, de acordo com a equação (3) da secção 3.2.3.

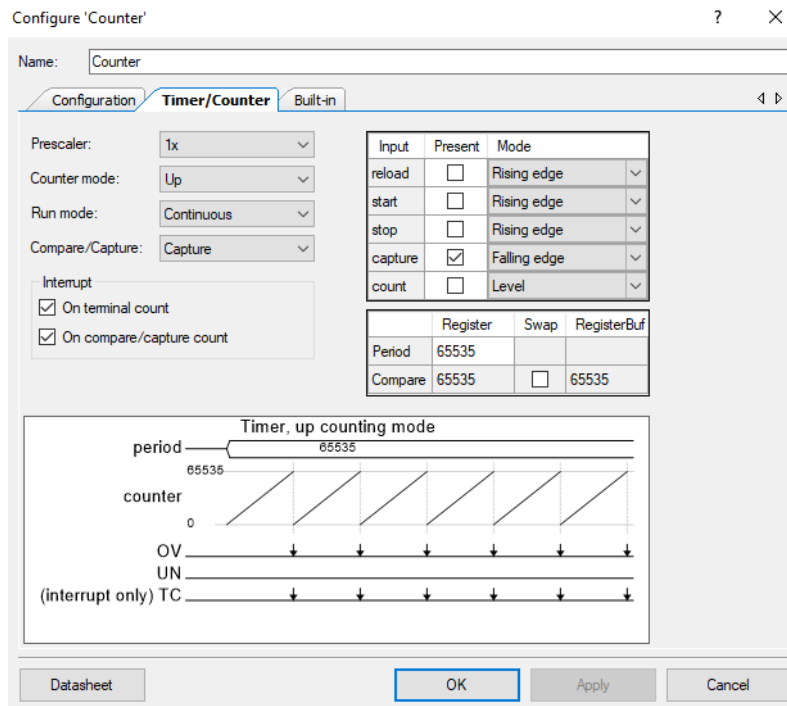


Figura 55. Configuração do contador no IDE PSoc Creator.

Foi utilizado o RTC (Real Time Clock) disponível no módulo PSoc 4 BLE para poder saber qual a data e hora atual. Isto é necessário para o sistema saber quando começa e acaba a aquisição de dados. Os horários de início e fim de aquisição são definidos pelo utilizador na aplicação Android. O componente RTC foi programado de acordo com a Figura 56. Este RTC usa o sistema Unix, que representa o número de segundos passados desde 1 de janeiro de 1970. A hora e data atual do RTC são atualizadas através do dispositivo Android, quando o sistema é inicializado pela primeira vez.

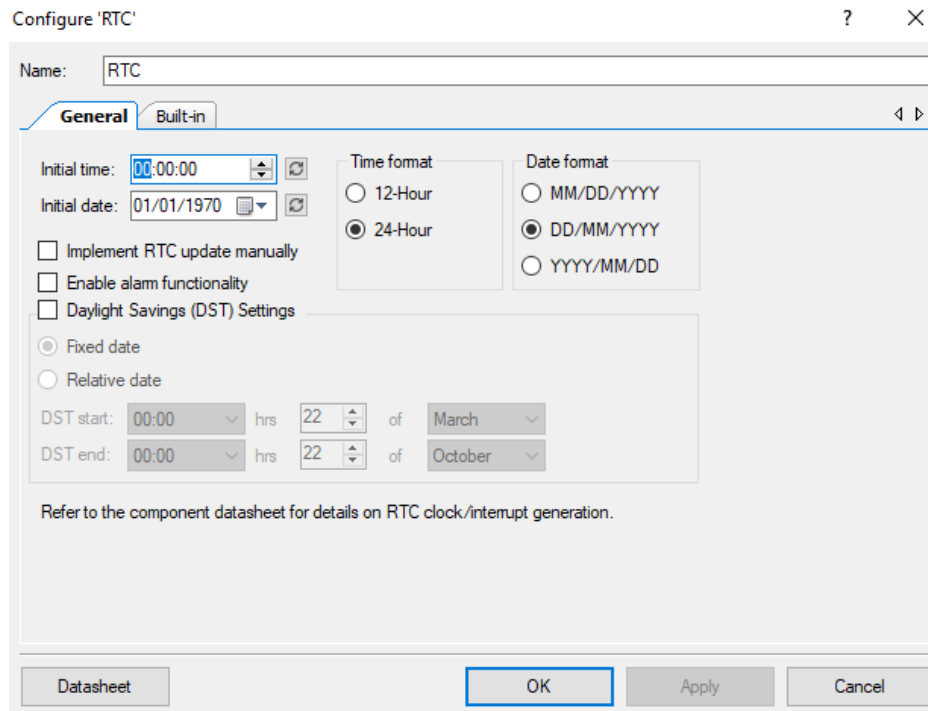


Figura 56. Configuração do componente RTC no IDE PSoC Creator.

Para a aquisição dos dados da temperatura e humidade foi programado um pino digital de forma bidirecional, podendo enviar e receber sinais. O algoritmo implementado foi descrito na secção 3.2.1. O pino é programado para obedecer a todos os tempos necessários para envio e receção do sinal [59].

Na Figura 57 é apresentado o fluxograma da aquisição de dados. Inicialmente são inicializados todos os periféricos e variáveis, antes do *loop* principal da função *main*. Os dados dos sensores e dos módulos fotovoltaicos são adquiridos e guardados de um em um segundo, sendo depois feita uma média dos valores quando chegado o período de amostragem. Dentro do *loop* é verificado se a *flag* correspondente ao período de amostragem foi ativada. Em caso positivo é calculada a média dos dados dos sensores e da potência dos módulos fotovoltaicos. Em caso negativo, fica à espera da ativação da *flag*. Depois de calculada as médias dos dados, estes são organizados numa *string*, que é usada como argumento da função de escrita de dados no cartão de memória. Aqui também se verifica se a hora atual do RTC é igual à hora de fim de aquisição definida pelo utilizador. Em caso positivo, é parada a aquisição de dados diária. Em caso negativo, a aquisição de dados é continuada.

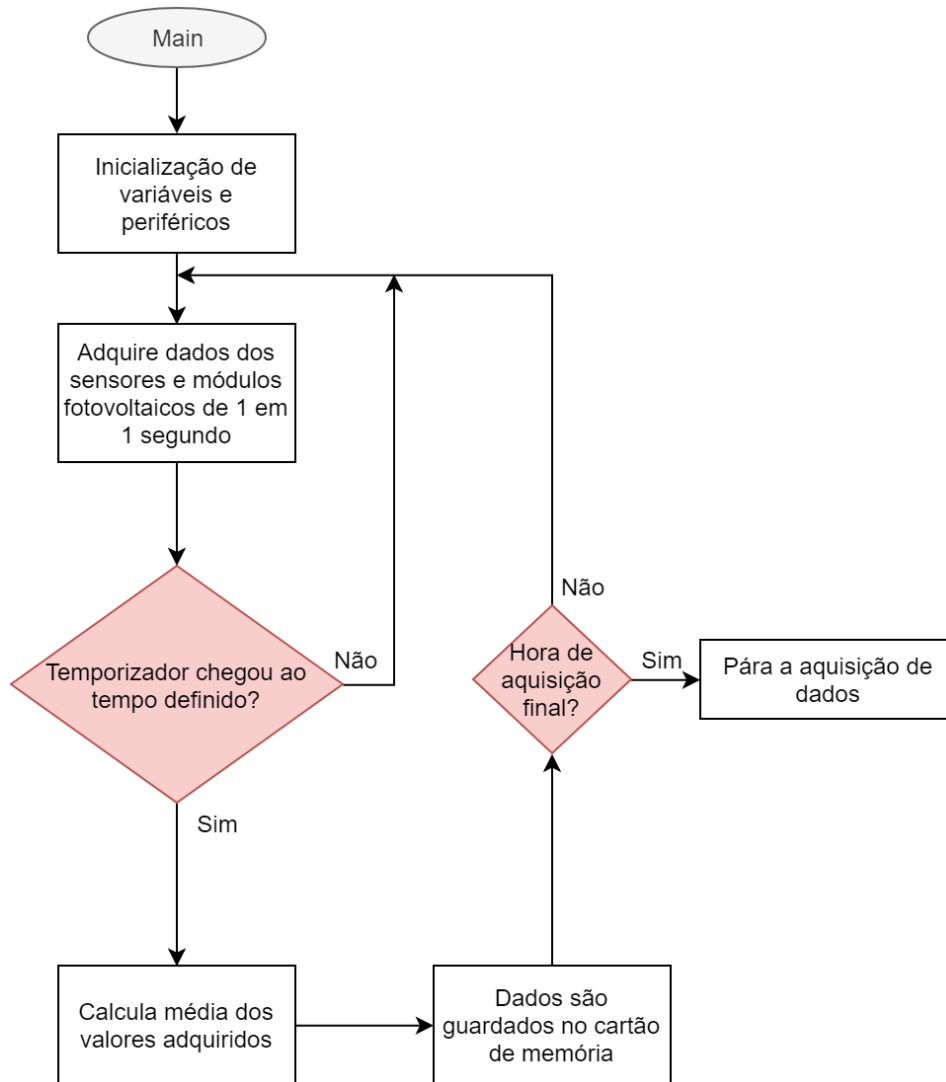


Figura 57. Fluxograma do processo de aquisição de dados na função main do programa.

3.3.2 Armazenamento de dados

Foi configurado um bloco de comunicação série como SPI (Figura 58) para comunicar com o leitor de cartões usado. O módulo PSoC 4 BLE irá funcionar como mestre nesta comunicação, sendo o módulo de leitor de cartões a funcionar como escravo. O *data rate* usado foi de 3 Mbps.

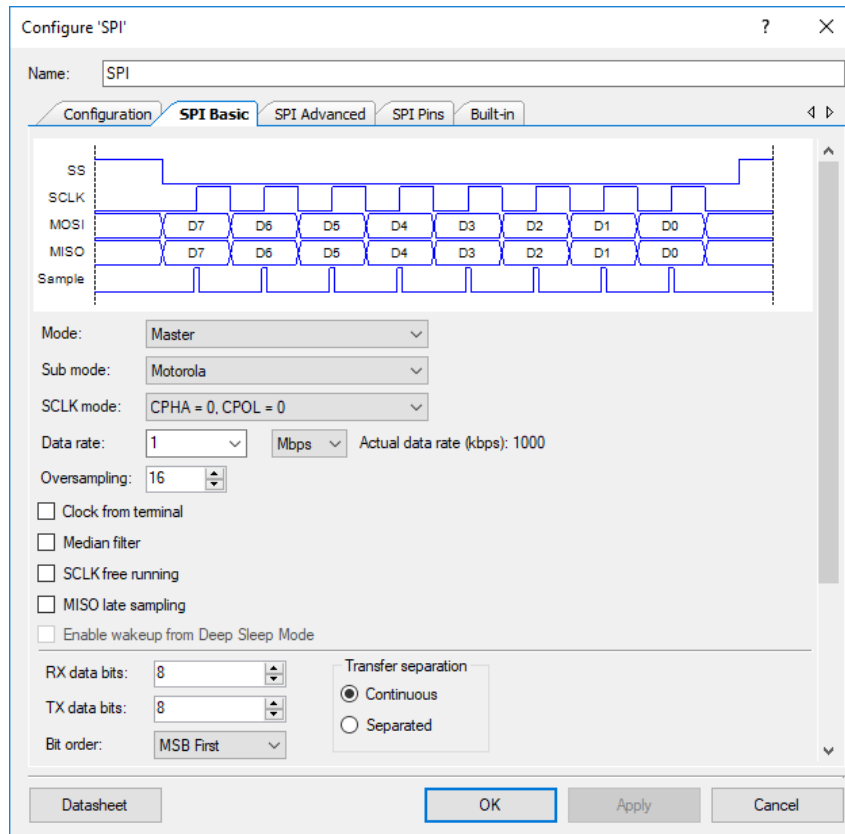


Figura 58. Configuração de um bloco de comunicação para SPI mestre.

Para ler e escrever em ficheiros de texto no cartão de memória foi utilizado o sistema de ficheiros de código aberto FatFs [60]. Este sistema foi desenvolvido para poder ser utilizado em diversas arquiteturas de microcontroladores. Foi usada uma biblioteca de código aberto em que é feito o *porting* desta biblioteca para dispositivos PSoC 4 [61]. Este sistema é baseado no sistema FAT (File Allocation Table), permitindo boa performance mesmo em dispositivos de baixo poder de processamento. Na Figura 59 é apresentada a pilha de funcionamento do sistema de ficheiros FatFs. Neste caso o controlo de baixo nível é feito através da comunicação SPI.

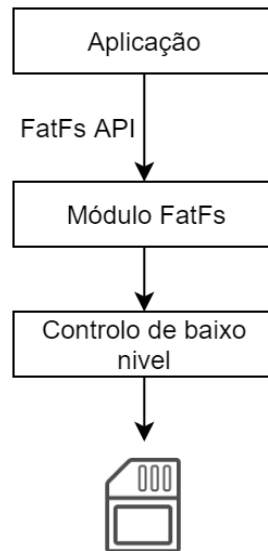


Figura 59. Pilha de funcionamento da biblioteca FatFs.

A função responsável pela escrita de dados no cartão de memória recebe como argumento uma *string* com todos os dados adquiridos, separados por espaço, como representado na Figura 60. Estes dados são depois convertidos para inteiro de 2 bytes aquando da sua transmissão. Dado que se pretende ter alguma resolução nos dados medidos, para transmitir os valores com uma casa decimal é usado o seu equivalente em inteiro (e.g., 22.3 fica guardado como 223).

Temperatura (°C)	Humidade (%)	Irradiação Solar (W/m ²)	Direção do vento (1 - N, 2 - NE etc.)	Velocidade do vento (km/h)	Potência módulo monocristalino (mW)	Potência módulo policristalino(mW)	Potência módulo amorfo (mW)	Bateria (%)
------------------	--------------	--------------------------------------	---------------------------------------	----------------------------	-------------------------------------	------------------------------------	-----------------------------	-------------

Figura 60. Formato da *string* de escrita no ficheiro de texto para um determinado dia.

Dado que os valores dos sensores e módulos são adquiridos de um em um segundo, foi criado um ficheiro de texto chamado de *temp.txt*, que guarda os valores adquiridos de um em um segundo até que chegue o período de amostragem definido pelo utilizador. Ao chegar o período de amostragem, é feita a média de todos os valores guardados neste ficheiro. Os dados do ficheiro de texto *temp.txt* são apagados e é iniciada novamente a aquisição de um em um segundo.

Para guardar os dados adquiridos no ficheiro de texto correspondente ao dia atual, é inicialmente lido o ficheiro de texto *info.txt*. O objetivo deste ficheiro é associar uma data a um nome de outro ficheiro de texto, contendo também a hora de início e fim de aquisição e o período de amostragem desse dia, todos estes dados separados por espaço (Figura 61). Neste caso o nome é um número ao qual será acrescentado a extensão *.txt*, começando em zero para o primeiro dia de aquisição de dados. Com um ficheiro de texto para cada dia, torna-se mais rápida a leitura de dados, em vez de guardar os dados todos num único ficheiro.

Número do dia de aquisição	Data	Hora do início da aquisição	Hora do fim da aquisição	Período de amostragem
----------------------------	------	-----------------------------	--------------------------	-----------------------

Figura 61. Formato do ficheiro de texto *info.txt*.

Existe também um outro ficheiro de texto, chamado *update.txt*, que é usado para guardar, caso solicitado pelo utilizador, a nova hora inicial e final de aquisição, bem como o novo período de amostragem. Após a conclusão da aquisição de dados do dia atual, este ficheiro de texto é lido e são atualizadas as novas definições. O formato deste ficheiro está apresentado na Figura 62.

Hora do início da aquisição	Hora do fim da aquisição	Período de amostragem
-----------------------------	--------------------------	-----------------------

Figura 62. Formato do ficheiro de texto *update.txt*.

Na Figura 63 é apresentado um exemplo do formato dos ficheiros de texto mencionados usados para armazenar os dados. Neste caso para os dois dias de exemplo, os dados foram adquiridos desde as oito horas da manhã até às 20 horas da noite, com um período de amostragem de 30 segundos. No dia seguinte a hora inicial e final de aquisição, bem como o período de amostragem, irão mudar para os valores descritos no ficheiro *update.txt*.

info.txt	0.txt
<pre>0 26/07/2019 08:00 20:00 30 1 27/07/2019 08:00 20:00 30</pre>	<pre>223 678 540 2 6 800 760 350 532 224 673 537 2 5 800 760 350 531</pre>
update.txt	1.txt
<pre>09:00 20:30 45</pre>	<pre>267 678 540 2 6 800 760 350 532 270 673 537 2 3 800 760 350 531</pre>

Figura 63. Exemplo do formato dos ficheiros de texto para armazenar os dados.

Cada vez que são escritos dados num ficheiro de texto, verifica-se se a data de aquisição já existe. Em caso positivo é aberto o ficheiro *info.txt* para ver o número associado a essa data e aberto o ficheiro de texto com esse número, procedendo à escrita dos dados. Em caso negativo é incrementado o número de dias e associado um novo número à nova data, atualizando o ficheiro *info.txt* e criando um ficheiro de texto para o novo dia. Na Figura 64 é apresentado num fluxograma todo o processo para armazenar os dados recolhidos pela estação.

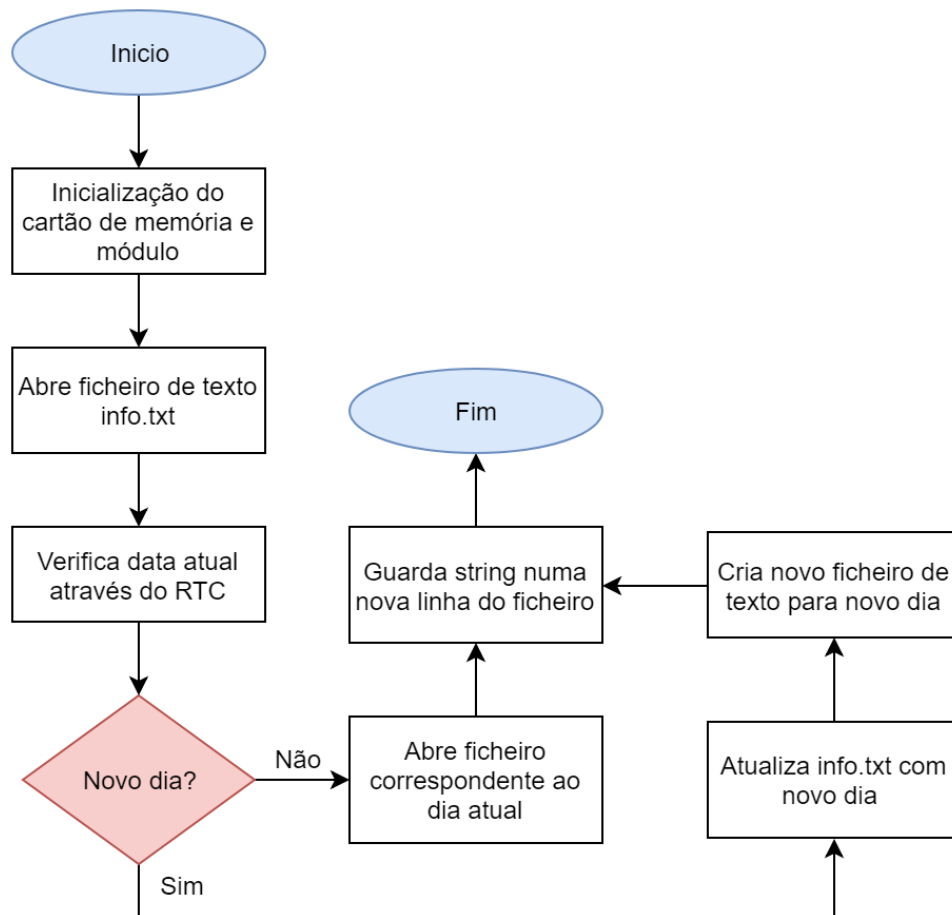


Figura 64. Fluxograma do processo de armazenamento de dados em ficheiros de texto.

Os dados armazenados são depois lidos para serem transferidos para o dispositivo Android. Este processo está apresentado na Figura 65. Foi desenvolvida uma função que tem como argumento uma data de um determinado dia. Através do ficheiro de texto *info.txt* é verificado se esta data existe. Em caso negativo é alertado o dispositivo Android que a data não existe. Em caso positivo é iniciada a leitura dos dados do ficheiro de texto associado a essa data. Ao ler os dados do ficheiro de texto, estes são guardados num buffer na memória SRAM. Após ser completa a leitura de dados, estes são transferidos para a base de dados GATT, e conseqüentemente, para o dispositivo Android. Caso os dados lidos do ficheiro de texto tenham um tamanho maior que o buffer definido (o tamanho deste é limitado), a leitura de dados do ficheiro de texto é interrompida, guardando a posição atual no ficheiro. São transferidos os dados atuais no buffer, e a após conclusão, é chamada a função novamente para ler os dados a partir da posição guardada.

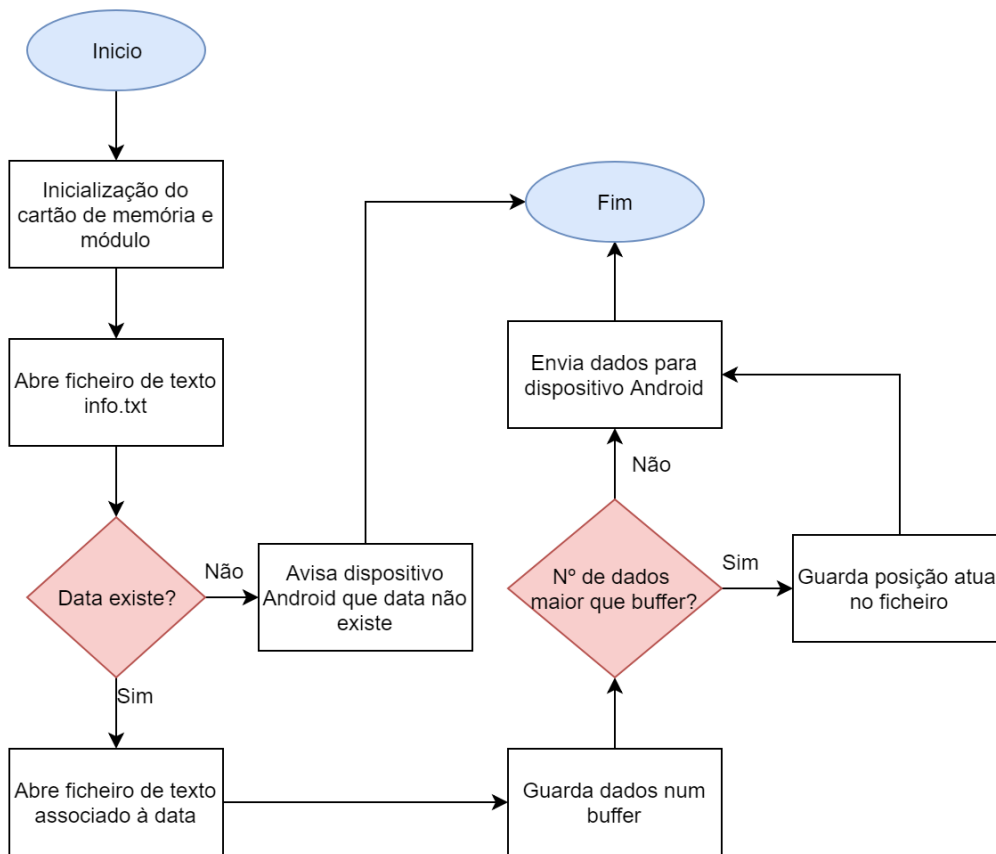


Figura 65. Fluxograma do processo de leitura de dados do cartão de memória.

3.3.3 Programação da rede BLE

Através do IDE PSoC Creator, é possível configurar e programar a rede BLE. Para isso foi escolhido o componente BLE e colocado no *design* do programa. Aqui é possível configurar a rede, definindo as características necessárias para a transferência de dados, bem como configurar definições GAP.

3.3.3.1 Configuração do componente BLE

O componente BLE foi configurado como GAP periférico, dado que será o dispositivo que irá anunciar a sua presença ao dispositivo GAP central (o dispositivo Android), quando este se aproxima da estação. Este dispositivo foi também configurado como servidor GATT, porque é o dispositivo que contém os dados a serem transferidos, sendo estes pedidos pelo cliente GATT (dispositivo Android).

Foram criadas várias características para suportar a transferência de dados do servidor GATT para o cliente GATT. Isto pode ser feito na aba *Profiles* do componente BLE (Figura 66). No total foram criadas nove características. O tipo refere-se ao tipo de dados, podendo ser, entre outros, *uint8*, *uint16 array*, *sint16*, *boolean*, etc. A propriedade da característica refere-se a como esta é transferida. Se for definida

como *Read*, o cliente GATT terá de fazer um pedido de leitura para ler a característica da base de dados GATT. Caso seja definida como *Notify*, a característica notifica o cliente GATT sempre que haja novos dados, sem este ter feito o pedido de leitura. Na Tabela 6 é apresentada todas as características criadas para suportar a transferência de dados entre dispositivos BLE, contendo a descrição de cada uma bem como a sua propriedade e tipo.

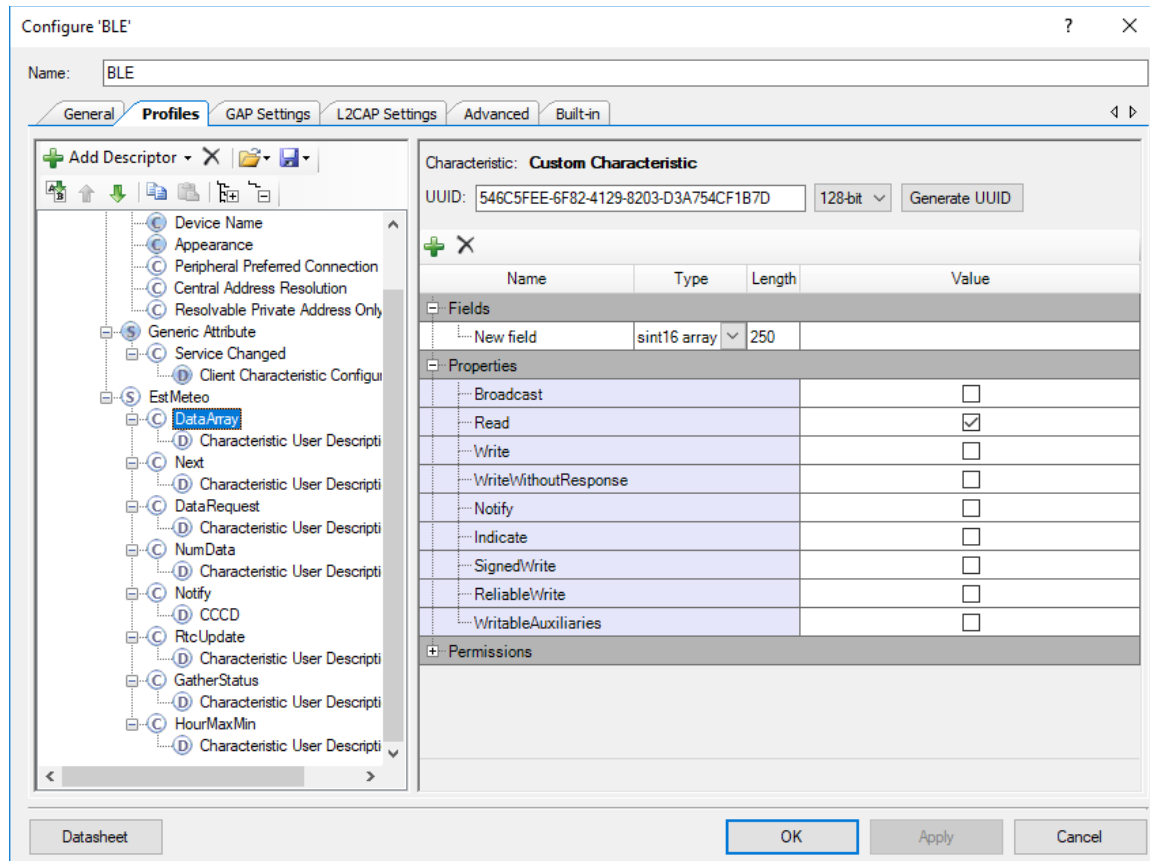


Figura 66. Configuração das características e do seu tipo e propriedade.

Tabela 6. Características criadas para suportar a transferência de dados.

Característica	Descrição	Tipo de dados	Propriedade
DataArray	Transfere os dados recolhidos pela estação 500 bytes de cada vez.	sint16 array [250]	Read
Next	Usada pelo dispositivo Android como confirmação dos dados transferidos.	uint8 [1]	Read/Write
DataRequest	Usada pelo dispositivo Android para pedir os dados de um determinado dia.	uint8 [20]	Write
NumBytes	Transfere para o dispositivo Android o número de bytes total a receber de um determinado dia.	uint16 [1]	Read
Notify	Usada para poder visualizar os últimos dados medidos pela estação em tempo real na aplicação	sint16 array [10]	Notify
RtcUpdate	Usada pelo dispositivo Android para atualizar o RTC do módulo com a data e hora atual caso necessário.	uint8[30]	Write
GatherStatus	Guarda o estado atual de aquisição.	uint8[3]	Read/Write
HourMaxMin	Usada para transferir as horas iniciais e finais de aquisição de um determinado dia.	uint8[20]	Read/Write
DatesSD	Transfere as datas de dias em que houve aquisição de dados.	uint8[500]	Read

Para minimizar o tempo necessário para a transferência dos dados entre a estação meteorológica e o dispositivo Android, é desejável que o débito da conexão BLE seja maximizada. Existem dois parâmetros que afetam o débito: o tamanho do MTU e o intervalo de transmissão entre pacotes (intervalo de conexão). O tamanho do MTU pode ser modificado para um valor entre 23 bytes e 512 bytes, e o intervalo de conexão entre 7.5 milissegundos e 4 segundos, em que para este último parâmetro, um valor menor corresponde a um maior consumo de energia. Dado que a transferência de dados não vai ser muito frequente foi escolhido o menor valor possível para o intervalo de conexão (7.5 ms), o tamanho do MTU foi definido em 500 bytes (Figura 67) e o tamanho do *payload* da camada de ligação para 251 bytes. Foi

também escolhido o valor máximo para a potência de transmissão da conexão (3 dBm), maximizando o alcance da rede (Figura 67).

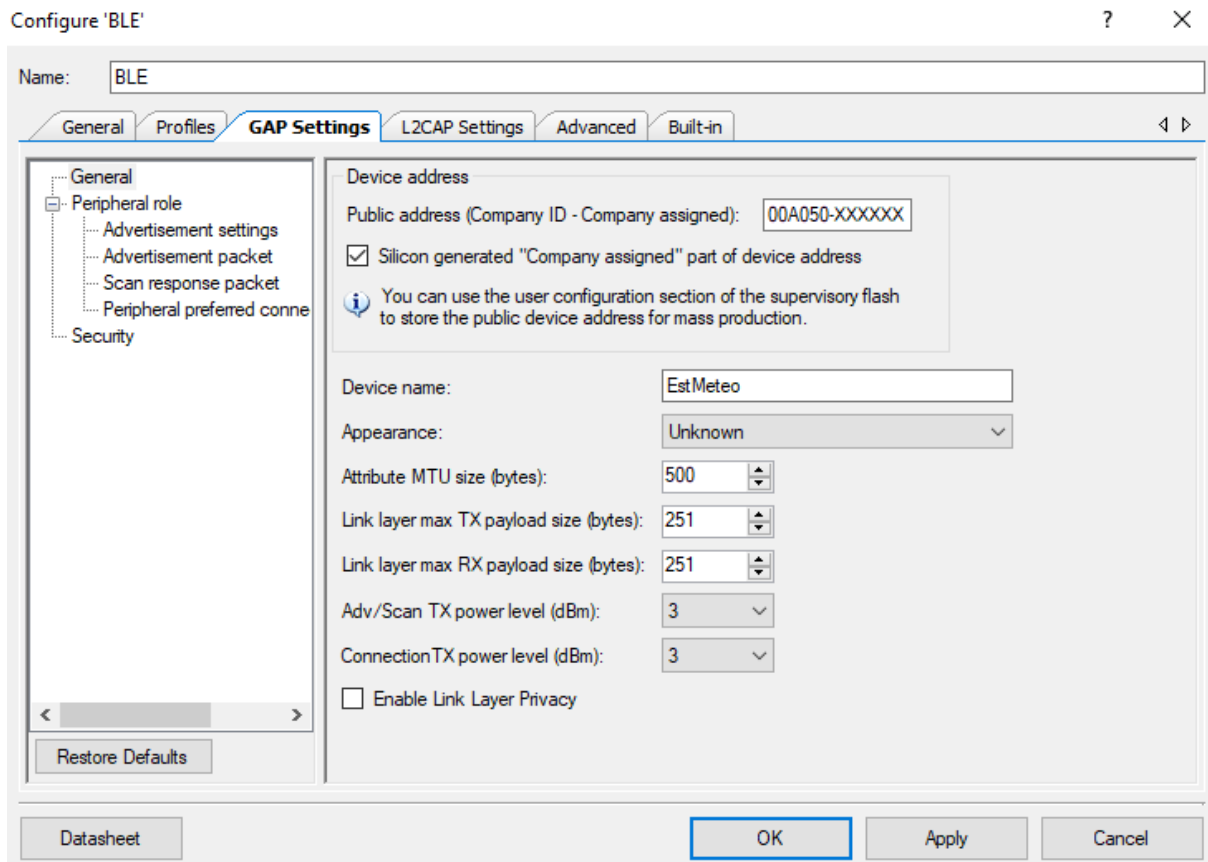


Figura 67. Configuração de definições gerais da camada GAP.

Os eventos que são gerados pela pilha protocolar BLE são processados por uma *callback*, chamada de BLE Stack Event Handler. Aqui é feita a gestão da conexão entre dispositivos BLE e configurações de como esta conexão deve reagir a diferentes tipos de eventos. Por exemplo, quando a pilha é inicializada ou uma conexão BLE é terminada, é dada a instrução para começar novamente o anúncio de presença do dispositivo BLE. Aqui também são processados os eventos de escrita em características feito pelo dispositivo Android. Por exemplo, para a transferência de dados entre os dispositivos BLE, é usada a característica Next, sendo escrito o valor de um nesta característica pelo dispositivo Android, sempre que forem recebidos 500 bytes e estes terem sido processados. Isto serve para confirmar o dispositivo PSoC que foram recebidos 500 bytes, e que pode ser iniciada a transferência dos próximos 500 bytes. O valor desta característica é verificado na *callback* através do evento CYBLE_EVT_GATTS_WRITE_REQ. A secção 3.4.4 explica em mais detalhe a transferência de dados entre os dispositivos.

3.3.3.2 Envio de dados para a base de dados GATT

Na Figura 68 é apresentado um exemplo de uma função em que envia dados para a base de dados GATT. É criada uma instância temporária de um valor no formato necessário para poder ser usado como argumento na função de escrita dos dados no atributo da característica. Esta função de escrita chama-se *CyBle_GattsWriteAttributeValue*. No exemplo da Figura 68 estão a ser enviados 500 bytes (tamanho da característica) de um *array* para a característica que transfere os dados recolhidos pela estação.

```
void update_data_array_GATT(int16 temp[250])
{
    CYBLE_GATTS_HANDLE_VALUE_NTF_T tempHandle;

    tempHandle.attrHandle = CYBLE_ESTMETEO_DATAARRAY_CHAR_HANDLE;

    tempHandle.value.val = (void *) temp;
    tempHandle.value.len = 500;

    CyBle_GattsWriteAttributeValue(&tempHandle, 0, &cyBle_connHandle, 0);
}
- }
```

Figura 68. Exemplo de envio de dados para a base de dados GATT.

3.3.3.3 Programa de depuração CySmart

Através do programa de depuração CySmart é possível verificar se os dados estão a ser corretamente transferidos para a base de dados GATT e conseqüentemente para o cliente GATT. Este programa está disponível para Windows e para Android. Para utilizar o programa em Windows, é necessário o uso do *dongle* disponível no kit CY8CKIT-BLE-A 4.2 mencionado na secção 3.2.1. Através deste programa é possível a procura de dispositivos BLE (Figura 69).

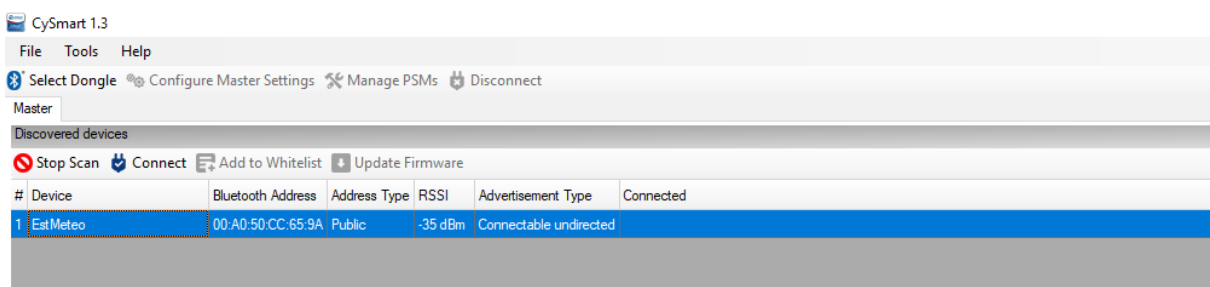


Figura 69. Procura de dispositivos BLE com o programa CySmart.

Na figura Figura 70 é apresentado um teste feito com a característica DataArray, enviando 18 valores de 2 bytes e verificado o resultado no programa CySmart. Os valores enviados foram os seguintes: {288,

586, 2801, 1, 87, 700, 700, 700, 35,288, 586, 2801, 1, 87, 700, 700, 700, 35}. No programa CySmart estes valores são apresentados no seu valor equivalente em hexadecimal.

UUID da característica DataArray		Valor da característica DataArray	
Primary Service Declaration			
0x0010	0x2800	Primary Service Declaration	FD:40:51:F5:D1:68:F4:A7:1B:41:05:24:B3:4B:66:45
Characteristic Declaration			
0x0011	0x2803	Characteristic Declaration	02:12:00:7D:1B:CF:54:A7:D3:03:82:29:41:82:6F:EE:5F:6C:54
0x0012	546C5FEE6F8241298203D3A754CF1B7D		20:01:4A:02:F1:0A:01:00:57:00:BC:02:BC:02:BC:02:23:00:20:01:4A:02:F1:0A:01:00:57:00:BC:02:BC:02:23:00:00:00:00
0x0013	0x2901	Characteristic User Description	288 586 2801 1 87 700 700 700 ...
Characteristic Declaration			
0x0014	0x2803	Characteristic Declaration	0A:15:00:D1:D5:44:33:54:77:3E:92:95:48:9E:85:A4:C2:59:25
0x0015	2559C2A4859E4895923E77543344D5D1		
0x0016	0x2901	Characteristic User Description	
Characteristic Declaration			
0x0017	0x2803	Characteristic Declaration	0A:18:00:6C:D2:96:56:A9:C4:09:94:5E:4F:14:69:B1:90:3A:56
0x0018	563A90B169144F5E9409C4A95696D26C		
0x0019	0x2901	Characteristic User Description	
Characteristic Declaration			
0x001A	0x2803	Characteristic Declaration	02:1B:00:18:F1:6E:70:F6:06:A7:89:26:44:77:66:A0:12:89:FB
0x001B	FBB912A06677442689A706F6706EF118		
0x001C	0x2901	Characteristic User Description	
Characteristic Declaration			
0x001D	0x2803	Characteristic Declaration	10:1E:00:98:1E:AD:1E:F4:E5:CF:98:55:4E:1B:DD:AE:EF:A6:87
0x001E	87A6EFAEDD1B4E5598CFE5F41EAD1E98		
0x001F	13608F3148914A7EBD61DA00DFD733DD		
Characteristic Declaration			
0x0020	0x2803	Characteristic Declaration	0A:21:00:5C:CD:DD:B3:0E:69:D1:92:62:4F:DB:D9:D9:3E:44:DB
0x0021	DB443ED9D9DB4F6292D1690EB3DDCDD5C		
0x0022	0x2901	Characteristic User Description	

Figura 70. Confirmação da transferência dos valores de teste através da característica DataArray.

3.4 Desenvolvimento da Aplicação em Android

A aplicação em Android foi desenvolvida usando o IDE Android Studio. A aplicação tem como objetivo a transferência dos dados recolhidos pela estação e a sua visualização. Na aplicação também é possível configurar algumas definições, como o horário de início e fim de aquisição, e o período de amostragem. Para fazer a procura, conexão e transferência de dados via BLE, foi seguida a API disponível no website Android Developers [62] e também os exemplos disponíveis pela Cypress Semiconductor [63].

3.4.1 Fluxograma da Aplicação

Na Figura 71 é apresentado o fluxo da aplicação Android desenvolvida. A primeira atividade da aplicação é dedicada à procura de dispositivos BLE. Aqui é apresentado numa lista o nome do dispositivo ou o seu endereço, bem como o RSSI (Received Signal Strength Indicator). Após a procura estar completa, é possível clicar num dos dispositivos da lista.

A próxima atividade estará dividida em dois fragmentos. O primeiro fragmento apresenta os últimos valores medidos pelos sensores da estação, possibilitando também a definição da hora de início e fim

de aquisição bem como o período de amostragem. O segundo fragmento é dedicado à transferência e consequente visualização de dados da estação. É possível escolher uma data para a qual os dados são transferidos, ou através de um interruptor, possibilitar a transferência dos dados de todas as datas em que dados foram adquiridos. Após iniciada a transferência de dados, um diálogo irá aparecer com uma barra de progresso. Ao concluir a transferência de dados, uma lista é atualizada com todas as datas disponíveis com dados a serem visualizados. Estes dados ficam guardados numa base de dados desenvolvida em SQLite. O utilizador poderá então clicar numa destas datas, o que abrirá uma nova atividade.

Esta nova atividade permite visualizar os dados transferidos através de gráficos. Foi usada a biblioteca de código aberto MPAndroidChart [64] para desenvolver esta atividade. No lado esquerdo desta atividade é possível escolher qual a variável a ser visualizada. Os dados transferidos previamente irão aparecer num gráfico.

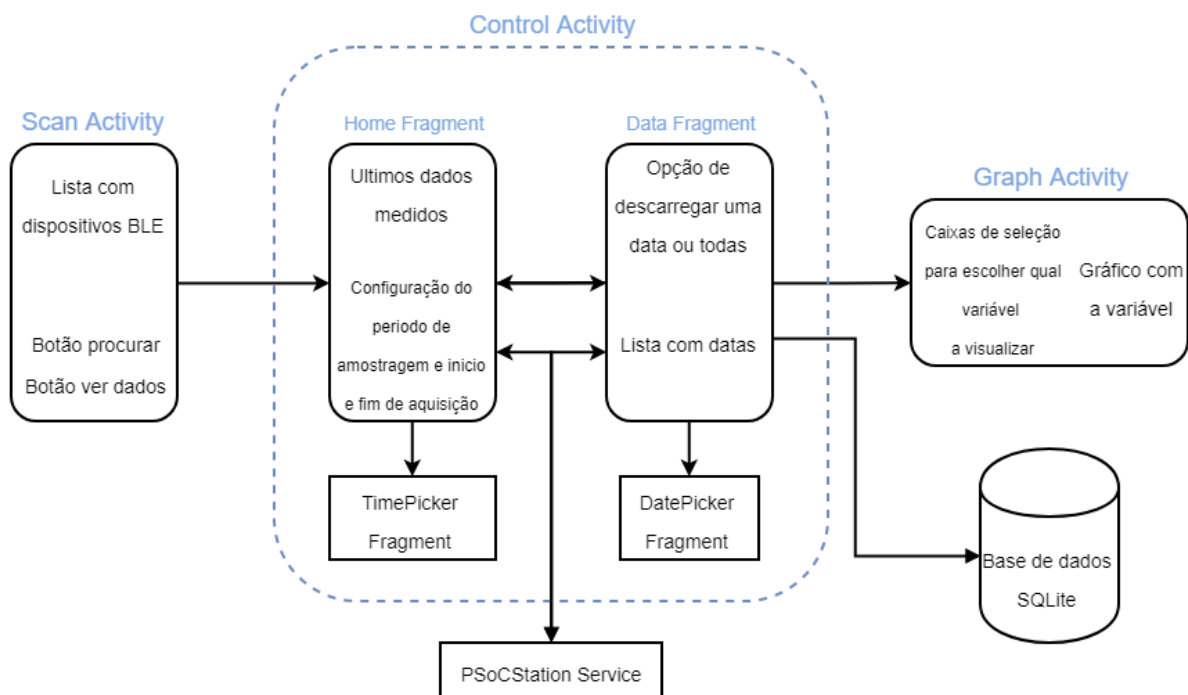


Figura 71. Fluxograma da aplicação Android desenvolvida.

3.4.2 Procura e conexão de dispositivos BLE

Para utilizar o módulo Bluetooth integrado no *smartphone* foi necessário definir algumas permissões no Android *manifest* (Figura 72). Foi também necessário definir a permissão de acesso à localização para o correto funcionamento da conexão BLE.

```
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
<!-- Location permission required for android 6.0 (Marshmallow) -->
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

Figura 72. Permissões no Android *manifest* para utilizar o módulo Bluetooth e a localização.

A primeira atividade da aplicação é dedicada à procura de dispositivos BLE. É verificado se o *smartphone* ou *tablet* suporta BLE e se o adaptador Bluetooth está ligado. Caso o adaptador não esteja ligado, é iniciado um diálogo que pede autorização ao utilizador para o ativar (Figura 73). O mesmo é feito para a localização.

```
if (!mBluetoothAdapter.isEnabled()) {
    Intent enableBtIntent = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
    startActivityForResult(enableBtIntent, REQUEST_ENABLE_BLE);
}
```

Figura 73. Verificação se o adaptador Bluetooth está ligado.

Para fazer a procura de dispositivos BLE é usado o método `startLeScan()`. Os resultados são depois retornados na *callback* `BluetoothAdapter.LeScanCallback`. Foi tido em consideração que a procura de dispositivos BLE consome bastante bateria. Para isso foi definido um tempo limite de procura de cinco segundos. Os resultados da procura são depois inseridos numa lista, mostrando o nome do dispositivo (ou endereço caso o nome não esteja disponível) e o RSSI.

Depois de escolhido o dispositivo que se pretende conectar, é iniciada uma nova atividade. Para esta atividade é transferido o endereço do dispositivo BLE pretendido, para estabelecer a conexão. Para lidar com a conexão e transferência de dados via BLE, foi criado um serviço chamado `PSoCStationService`. Dentro deste serviço é feita a conexão ao servidor GATT do dispositivo BLE através do método `device.connectGatt()`. O resultado da conexão é retornado na *callback* `BluetoothGattCallback`, no método `onConnectionStateChange()`. Em caso positivo é iniciada a descoberta de serviços e em caso negativo é mostrada uma mensagem ao utilizador que a conexão não foi possível ser estabelecida.

3.4.3 Configuração da aquisição de dados

No fragmento Home é possível configurar a hora de início e fim de aquisição dos dados, bem como o período de amostragem. Para escolher a hora de início e fim de aquisição foi desenvolvido um outro fragmento chamado `TimePickerFragment`, mostrando um diálogo com a possibilidade de escolher a hora

e minuto. Caso uma aquisição já esteja a ocorrer estas novas configurações apenas serão utilizadas no próximo dia de aquisição. Para transferir estes novos dados para a estação são utilizadas as características HourMaxMin e GatherStatus. Estas novas configurações são guardadas na estação num ficheiro de texto dedicado para tal, como descrito na secção 3.3.2.

Neste fragmento também é possível visualizar os últimos dados medidos pela estação em tempo real. Para isto foi utilizada a característica Notify, tendo como propriedade notificação. A cada um segundo estes valores são atualizados, após ser feita uma conexão ao servidor GATT.

3.4.4 Transferência de dados

Antes de a transferência ser iniciada, o dispositivo Android irá fazer um requisito de dados através da característica DataRequest. Esta característica tem como valor uma data com o formato dd/mm/aa. Este pedido é feito pelo utilizador na aplicação Android, no fragmento DataFragment. Caso se tenha ativado o interruptor para transferir os dados de todas os dias disponíveis, é utilizada a característica DatesSD. Esta característica irá procurar no ficheiro de texto info.txt todos os dias em que haja dados recolhidos e envia as datas correspondentes para o dispositivo Android. O dispositivo Android utiliza depois a característica DataRequest para cada data recebida.

Dado que o MTU foi definido para 500 bytes no dispositivo PSoC, o mesmo terá de ser feito no dispositivo Android, para que os dois dispositivos estejam em sintonia. Isso é feito recorrendo ao método requestMTU(), disponível através da classe BluetoothGatt.

Apesar de o intervalo de conexão ter sido definido para 7.5 ms no dispositivo PSoC, não existirá mudança deste parâmetro na conexão, dado ser o dispositivo Android (GAP central) a decidir este parâmetro. Para isso tem de ser usada a função requestConnectionPriority(), usando um de três possíveis valores para o argumento desta função:

- Zero para uma conexão equilibrada (CONNECTION_PRIORITY_BALANCED).
- Um para uma conexão com alta prioridade e baixa latência (CONNECTION_PRIORITY_HIGH).
- Dois para uma conexão de baixo débito, com pouco consumo de energia (CONNECTION_PRIORITY_LOW_POWER).

Apesar de na documentação não ser especificado o que acontece em cada uma destas prioridades [65], após alguns testes, o valor de um (CONNECTION_PRIORITY_HIGH) apresenta de facto melhor débito, como esperado.

Na Figura 74 é apresentado o algoritmo desenvolvido para a transferência de dados relativamente aos dados medidos pela estação. A transferência de dados por parte do dispositivo PSoC para o servidor GATT foi descrita na secção 3.3.3. Antes de receber os dados, o dispositivo Android necessita de saber quantos bytes irão ser transferidos. Esse valor é guardado na característica NumBytes, sendo esta lida pelo dispositivo Android antes de começar a transferência de dados. O dispositivo PSoC irá proceder à transferência dos primeiros 500 bytes para o servidor GATT. O dispositivo Android lê 500 bytes e escreve na característica Next o valor de um. Esta característica serve para alertar o dispositivo PSoC que a transferência de dados (500 bytes) foi completa. Ao verificar que o valor de Next mudou para um (está a zero por padrão), o dispositivo PSoC transfere os próximos 500 bytes. Isto é repetido até que o número de bytes transferidos seja igual ao valor da característica NumBytes. No final da transferência de dados de uma determinada data, são também enviadas as horas de início e fim de aquisição, através da característica HourMaxMin. A razão para ser usado a propriedade de leitura para transferir dados e uma característica de escrita (Next) para fazer o *acknowledge* da receção de dados, é que assim permite controlo total do fluxo de dados.

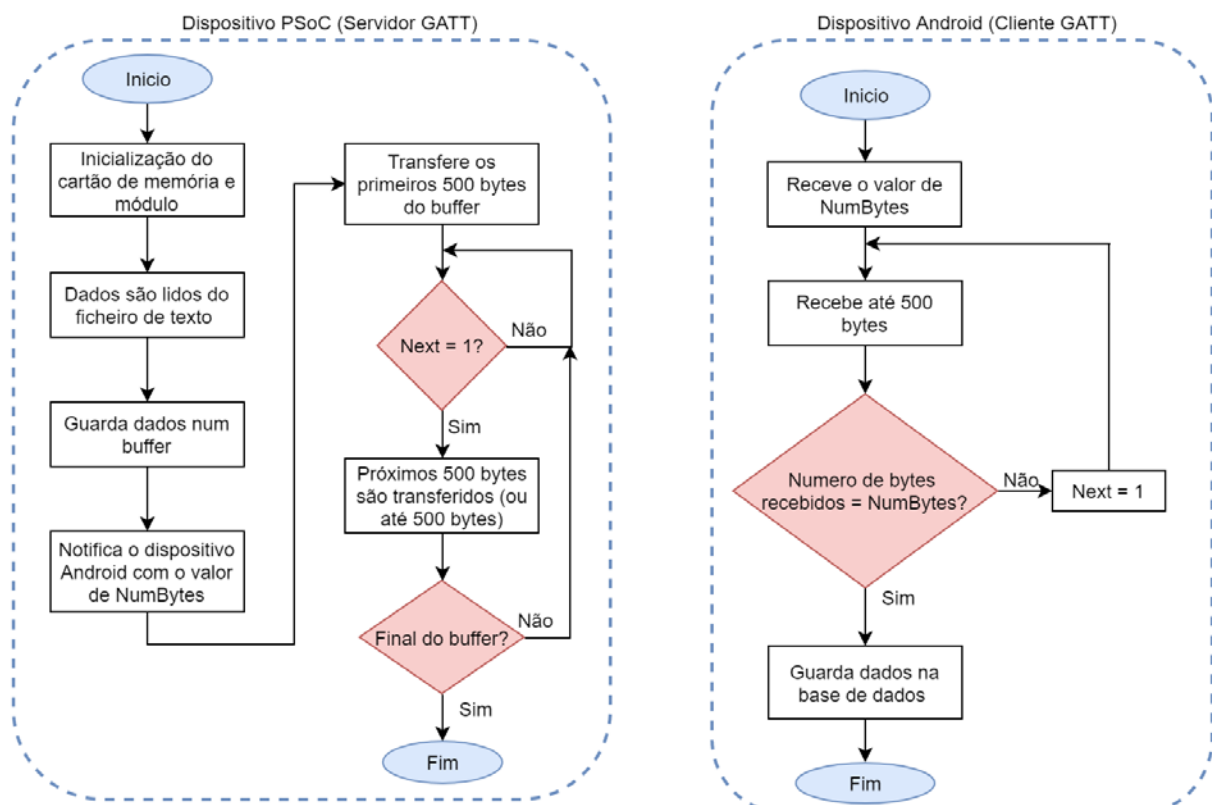


Figura 74. Algoritmo desenvolvido para transferência de dos dados medidos.

A característica usada para transferir os dados é a característica `dataArray`, tendo esta uma propriedade de leitura. Após ser feito o pedido de transferência de dados por parte do dispositivo Android

ao servidor GATT do dispositivo PSoC, os valores são retornados na *callback* `BluetoothGattCallback`, no método `onCharacteristicRead`. Dado que existem várias características com a propriedade de leitura, é verificado qual a característica que chamou este método, através do UUID associado a cada característica. É usado depois o método `getValue()` da classe `BluetoothGattCharacteristic` para retornar os valores transferidos, sendo estes guardados num buffer.

Para alertar a atividade `Control Activity` cada vez que é terminada a transferência de dados de uma característica, é enviado um *broadcast* [66] para essa atividade através da classe `Intent`. Na atividade foi implementado um *broadcast receiver* que determina o que fazer para cada diferente *broadcast* recebido. Estes dados são aqui processados e guardados na base de dados desenvolvida em `SQLite`.

3.4.5 Base de dados SQLite

Para ser possível guardar os dados transferidos no dispositivo Android, sendo assim possível a visualização destes a qualquer momento depois de feita a transferência, foi desenvolvida uma base de dados em `SQLite`. Para isso foram criadas duas tabelas, a tabela `DADOS` e a tabela `INFO`. A tabela `DADOS` guarda todos os valores transferidos. A tabela `INFO` guarda informação sobre esses dados transferidos, nomeadamente a hora de início e fim de aquisição de um determinado dia, bem como o período de amostragem e o número total de dados desse dia. Na Figura 75 é apresentada a estrutura das tabelas utilizadas. Quando se é escolhida uma data na lista do fragmento de transferência de dados, será iniciada a procura de todos os dados com essa data associada na tabela `DADOS`. Estes dados são depois retornados para a atividade de visualização de dados para serem representados graficamente. Será também iniciada uma procura na tabela `INFO`, para obter a hora de início e fim de aquisição, tendo em vista a sua apresentação no eixo de abcissas do gráfico de visualização de dados.

INFO	DADOS
*ID_Info	*ID_Dados
Data	Data
Hora inicial	Temperatura
Hora final	Humidade
Periodo de amostragem	Irradiação Solar
Número de dados	Direção do vento
	Velocidade do vento
	Módulo Mono
	Módulo Poly
	Módulo Amorfo
	Bateria

Figura 75. Tabelas criadas para guardar os dados transferidos.

3.4.6 Visualização de dados

Para visualizar os dados recolhidos pela estação foi desenvolvida a atividade GraphActivity. Esta atividade permite a visualização de dados através de gráficos, utilizando a biblioteca de código aberto MPAndroidChart [64]. O utilizador poderá escolher a variável a visualizar através de um conjunto de caixas de seleção, podendo estar apenas uma ativada de cada vez. A variável irá depois aparecer no gráfico, sendo este ajustado automaticamente dependendo do valor máximo e mínimo da variável. No eixo das abcissas estarão sempre representadas as horas de aquisição. É possível também ao utilizador fazer zoom no gráfico para, por exemplo, verificar qual o valor adquirido a uma certa hora.

4. Resultados e discussão

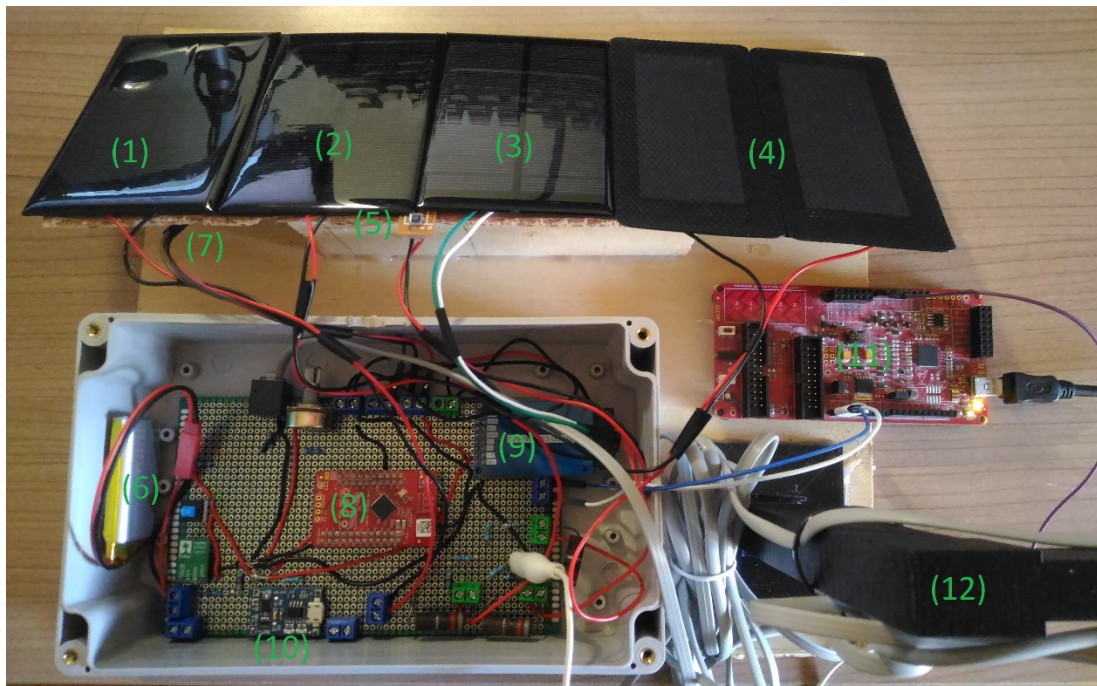
4.1 Protótipo desenvolvido

Na Figura 76 é apresentado o protótipo da estação meteorológica que foi desenvolvido. A estação inclui os sensores de temperatura e humidade, irradiação solar, velocidade e direção do vento, bem como os módulos fotovoltaicos. A eletrónica para adquirir todos os dados e a bateria estão dentro de um invólucro próprio. Foi usada uma inclinação de 40° em relação à vertical para os módulos fotovoltaicos, sendo este o valor recomendado para o mês de outubro no distrito de Braga, de acordo com o website em [67].



Figura 76. Protótipo da estação meteorológica desenvolvida.

Na Figura 77 é representado uma descrição detalhada da estação, incluindo o conteúdo do interior da caixa.



- (1) - Módulo de silício monocristalino (recarregamento da bateria)
- (2) - Módulo de silício monocristalino
- (3) - Módulo de silício policristalino
- (4) - Módulo de silício amorfo (dois em série)
- (5) - Fotodíodo BPW34
- (6) - Bateria LiPo 3.7 V @ 1600 mAh
- (7) - Sensor DHT22 (debaixo do suporte dos módulos)
- (8) - Módulo PSoC 4 BLE
- (9) - Leitor de cartão de memória
- (10) - Carregador de baterias LiPo TP4056
- (11) - Placa de desenvolvimento (apenas para depuração)
- (12) - Suporte para anemómetro e cata-vento

Figura 77. Descrição da estação meteorológica vista de cima com a caixa aberta ligada à placa de desenvolvimento.

Inicialmente o objetivo era fazer uma PCB para acomodar toda a eletrônica em conjunto com o módulo PSoC 4 BLE e o leitor de cartão de memória, mas naquela altura a máquina de fabricar PCBs disponível na universidade estava avariada. Sendo assim foi desenvolvida uma *perfboard*, apresentada na Figura 78, com toda a eletrônica para aquisição dos dados dos sensores, o módulo PSoC 4 BLE, o leitor de cartão de memória, bem como a eletrônica para alimentação da estação.

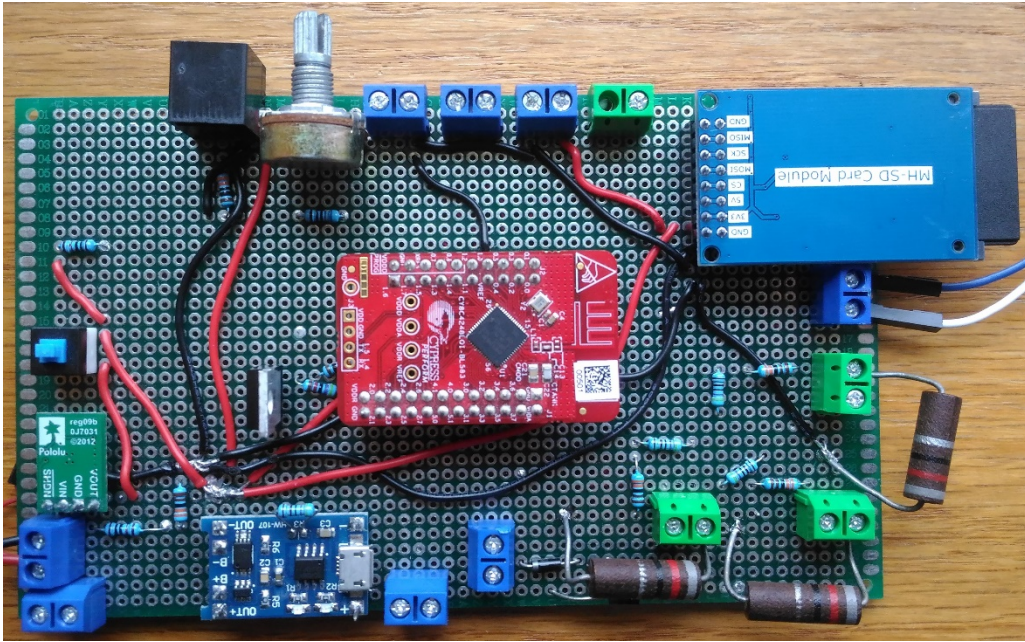


Figura 78. *Perfboard* que integra toda a eletrônica para aquisição de dados, módulo PSoC 4 BLE, leitor do cartão de memória e alimentação da estação.

Na Figura 79 é apresentada a PCB desenvolvida usando os conversores LTC3129 para maximização da extração de energia dos módulos fotovoltaicos. Esta parte infelizmente não foi totalmente implementada devido a problemas no desenvolvimento da PCB e do processo de soldagem do circuito integrado à PCB. A dificuldade encontrada foi devido aos conversores LTC3129 terem um *ground pad*, tendo este de estar soldado à PCB para dissipação de calor e conexão ao *ground*, em que as PCBs desenvolvidas na universidade não são as mais adequadas para isto. Foram feitas simulações deste conversor usando o programa LTSpice.

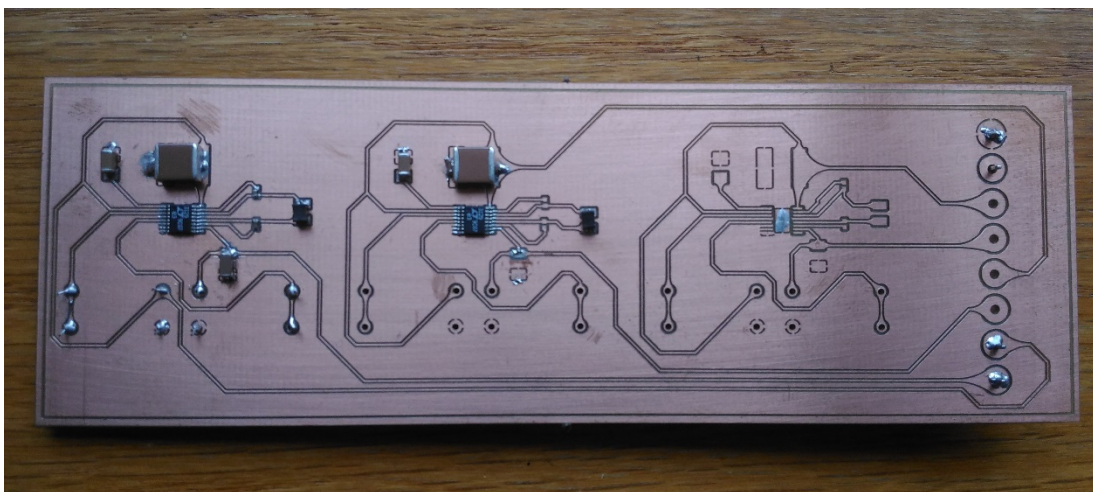


Figura 79. PCB desenvolvida para maximização da extração de energia dos módulos fotovoltaicos usando o conversor LTC3129.

4.2 Simulação LTSpice

Para simular o conversor LTC3129 e também os módulos fotovoltaicos, foi utilizado o programa LTSpice, desenvolvido pela Linear Technologies. Dado não existir um componente para módulos ou células fotovoltaicas neste programa, foi utilizado o circuito equivalente, apresentado na Figura 80.

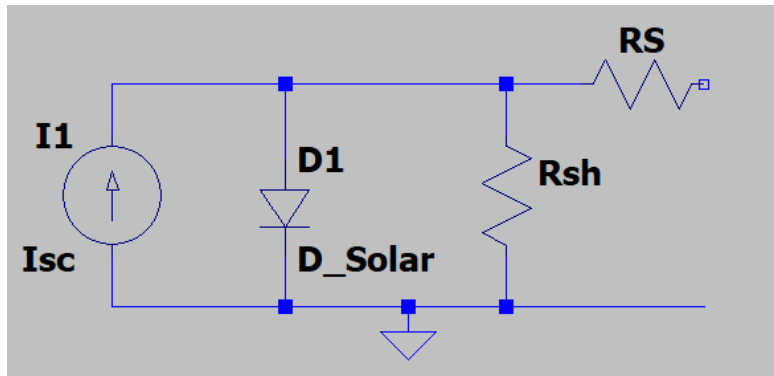


Figura 80. Circuito equivalente para uma célula ou módulo fotovoltaico.

Para cada circuito equivalente do módulo fotovoltaico foi necessário definir os valores de cada componente de modo a emular os valores dos módulos (Tabela 5 – Secção 3.2.5). Esses valores são o valor da fonte da corrente (I_{sc}), a corrente de saturação inversa do diodo (I_s), fator de idealidade do diodo (N), valor da resistência em série (R_S) e o valor da resistência *shunt* (R_{sh}). Através do valor de I_{sc} é possível ajustar a corrente máxima que o módulo pode fornecer. Mudando os valores de I_s e N do diodo pode-se ajustar a tensão de circuito aberto que o módulo produz e a tensão de potência máxima. O valor de R_S irá afetar também a tensão de potência máxima, sem afetar a tensão de circuito aberto. Ajustando R_{sh} consegue-se alterar o quão rápido o valor da corrente do módulo é reduzida com o aumento da tensão. Estas variáveis foram então ajustadas para obter valores aproximados aos da secção 3.2.5. Como os módulos de silício policristalino e monocristalino apresentam valores semelhantes, foi usado o mesmo modelo com os mesmos valores para as duas tecnologias. Na Figura 81 são apresentados os valores utilizados para obter as características dos módulos fotovoltaicos utilizados. Para obter os gráficos de I/V (corrente/tensão) e P/V (potência/tensão) foi também necessário adicionar uma fonte de tensão com valor de 0 V aos circuitos equivalentes.

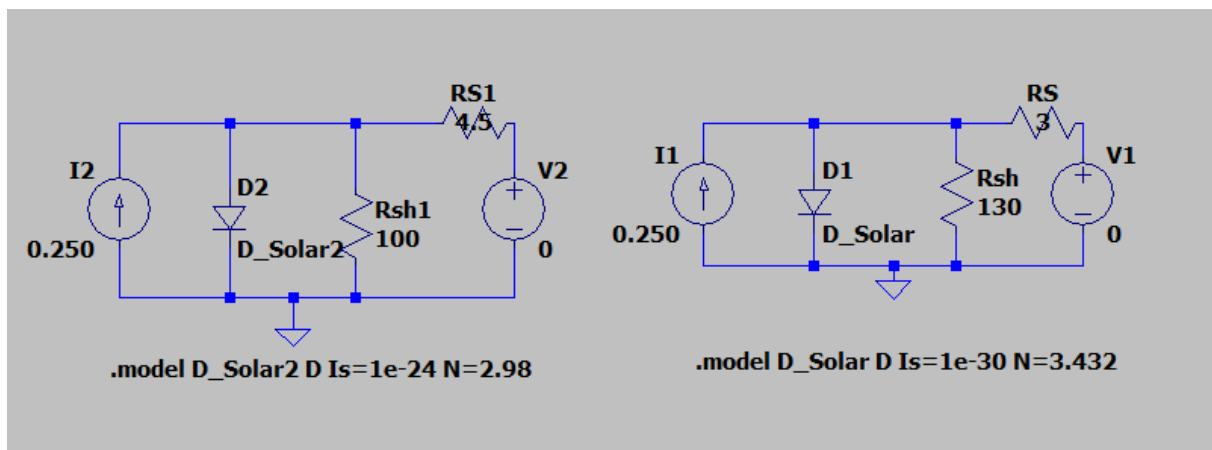


Figura 81. Circuitos equivalentes para os módulos fotovoltaicos amorfo (esquerda) e monocristalino/polícristalino (direita) para obtenção dos gráficos I/V e P/V.

Através da opção *Edit Simulation Command* foi feito um varrimento em DC de 0 a 6 volts com incrementos de 10 mV.

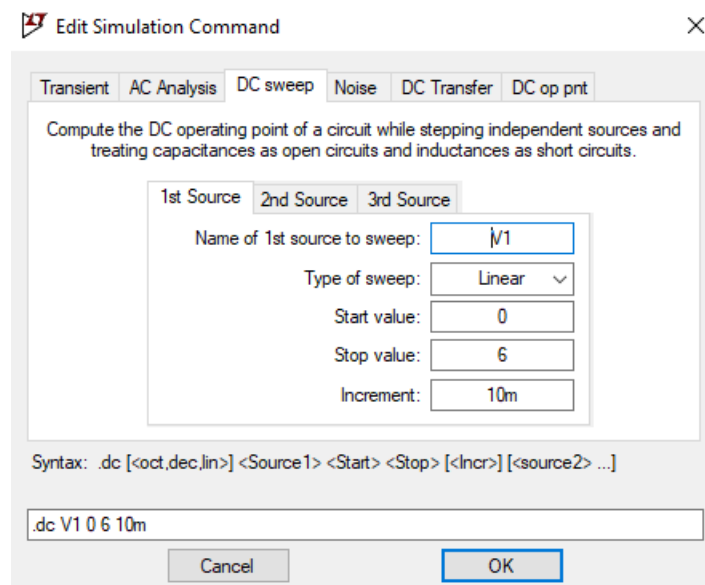


Figura 82. Configuração da simulação para fazer varrimento em DC.

Na Figura 83 é apresentado o gráfico I/V para o módulo de silício policristalino e monocristalino. Na Figura 84 é apresentado o gráfico P/V correspondente a estes módulos. Neste caso o valor de MPP é de 1.02 W.



Figura 83. Gráfico I/V para os módulos policristalino e monocristalino.



Figura 84. Gráfico P/V para os módulos policristalino e monocristalino ($V_{mpp} = 5\text{ V}$; $MPP = 1,02\text{ W}$).

Na Figura 85 é apresentado o gráfico I/V para o módulo fotovoltaico de silício amorfo (dois módulos em série). Na Figura 86 é apresentado o gráfico P/V correspondente, em que o valor de MPP é de 603 mW.



Figura 85. Gráfico I/V para o módulo de silício amorfo.

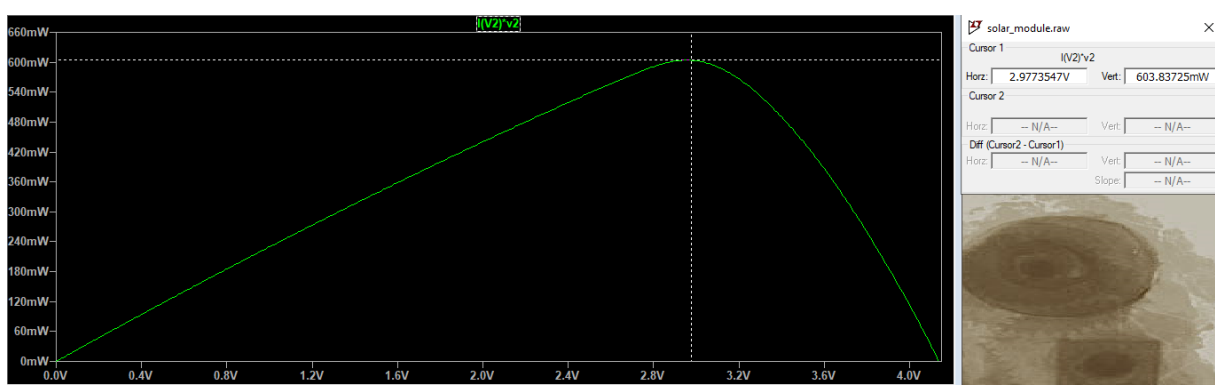


Figura 86. Gráfico P/V para o módulo de silício amorfo ($V_{mpp} = 2,98 \text{ V}$; $MPP = 603,83 \text{ mW}$).

Foi adicionada uma carga de 50Ω ao circuito equivalente que representa o módulo de silício monocristalino e policristalino, como apresentado na Figura 87.

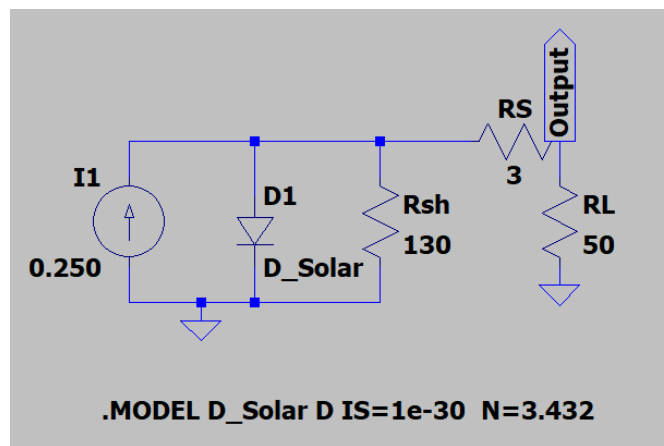


Figura 87. Circuito equivalente para módulo de monocristalino e policristalino com uma carga de 50Ω na saída.

O valor da tensão e potência de saída obtidos estão expressos na Figura 88.

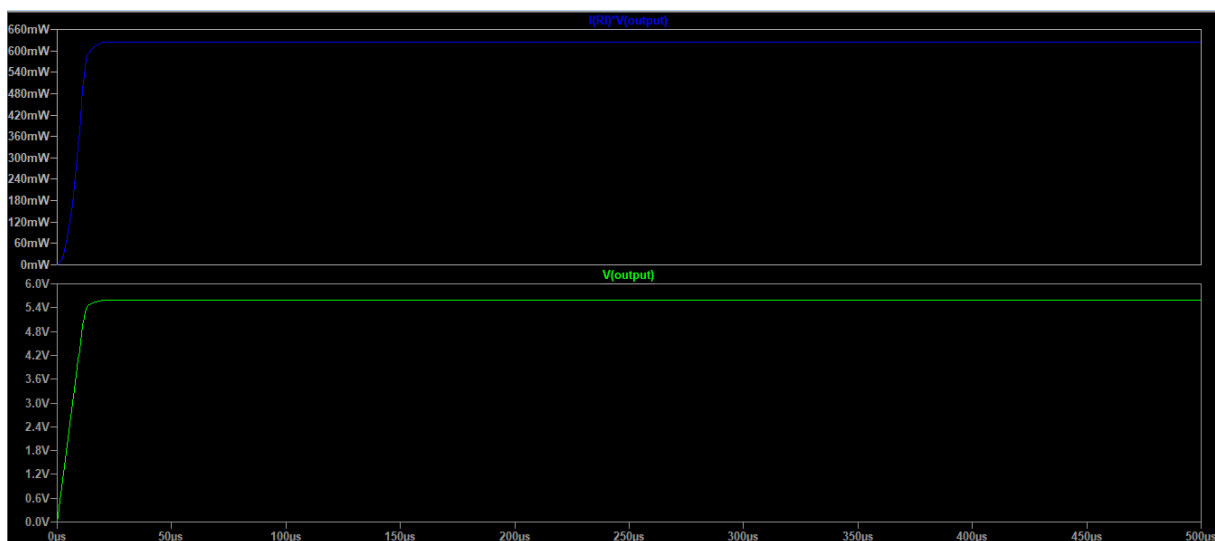


Figura 88. Tensão e potência de saída obtida no módulo sem MPPC (Potência saída = 624 mW ; $V_{out} = 5,58 \text{ V}$).

Comparando a tensão de saída (5,58 V) com o gráfico I/V correspondente para este módulo (Figura 84), a tensão está relativamente longe do valor da tensão no ponto de potência máxima (5 V). O valor da potência de saída foi de aproximadamente 624 mW.

Foi simulado o conversor LTC3129, usando a sua funcionalidade de MPPC. Este conversor necessita de vários componentes externos para o seu funcionamento, como representado na Figura 89. Através de um divisor de tensão entre o pino MPPC e o pino Vin, é possível configurar a tensão V_{mmpc} , em que o conversor irá procurar manter a tensão de entrada igual ao valor da tensão de referência programada. Através de outro divisor de tensão entre a saída do conversor e o pino FB, é possível configurar a tensão de saída. Os outros componentes foram adicionados de acordo com o *datasheet* [51] do conversor LTC3129. Na entrada do conversor estará a saída do modelo do módulo fotovoltaico simulado anteriormente, neste caso do módulo de silício monocristalino/policristalino.

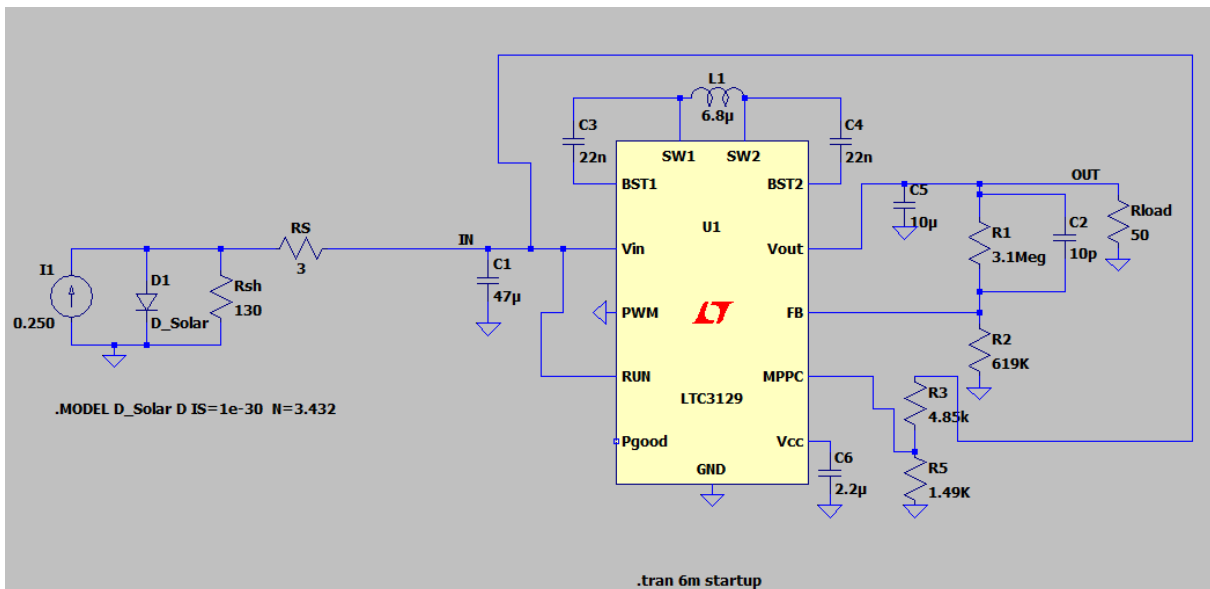


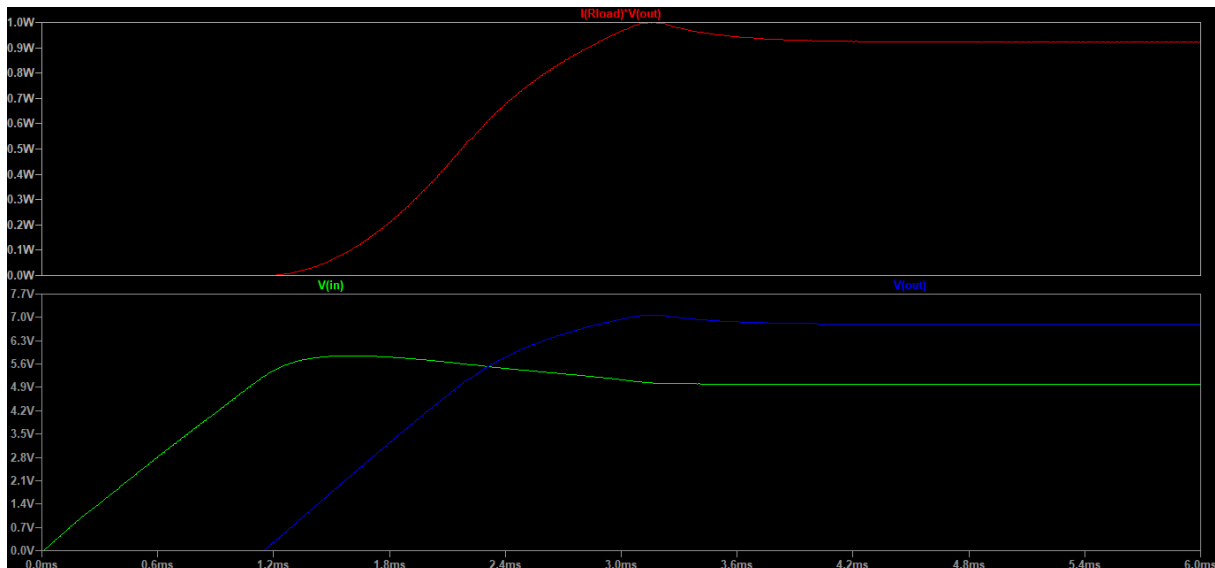
Figura 89. Simulação do conversor LTC3129 com o modelo do módulo fotovoltaico de silício monocristalino/policristalino.

Foi testado o impacto do valor que a tensão de MPPC (V_{mppc}) tem na potência produzida. Programou-se uma tensão de saída (V_{out}) para 7 V ($R1 = 3.1 \text{ M}\Omega$ e $R2 = 619 \text{ k}\Omega$) com uma carga de 50Ω , resultando numa potência de saída teórica de aproximadamente 1 W. Assim garante-se que será produzida a potência máxima na saída, dado que a potência máxima produzida pelo módulo fotovoltaico de silício policristalino e monocristalino é de 1 W. Neste caso assume-se que o módulo está em condições ideais. Através do valor da resistência R3 foi-se variando o valor da tensão de MPPC. Para isso foi feita uma simulação transiente de 0 ms a 6 ms, com V_{in} a começar de 0 V. Os resultados obtidos estão expressos na Tabela 7.

Tabela 7. Resultados obtidos através da variação da tensão de MPPC.

Vin (V)	Vmppc (V)	Vout (V)	Carga (Ω)	Potência de saída (mW)
3,94	4	6,09	50	741
4,45	4,5	6,47	50	837
4,77	4,8	6,68	50	892
5	5	6,79	50	921
5,42	5,5	6,15	50	759

Pode-se verificar que a potência máxima produzida na saída existe quando o valor de V_{mppc} é de 5 V, sendo esta a tensão de potência máxima do módulo de monocristalino e policristalino. Apesar de se ter programado a tensão de saída para 7 V, o valor máximo obtido é de 6.8 V. Isto deve-se ao facto de haver perdas no conversor, sendo que não irá conseguir uma potência de saída igual à potência produzida pelos módulos fotovoltaicos. Verifica-se também que o conversor irá sempre tentar manter V_{in} perto do valor de V_{mppc} programado, como esperado. A potência de saída é relativamente maior que aquela em que não foi utilizado o conversor LTC3129 entre o módulo e a carga (Figura 88), havendo um aumento de 47,6 %, para a situação em que foi usado um V_{mppc} de 5 V. Na Figura 90 é apresentado o gráfico da potência produzida na saída do conversor e dos valores de V_{out} e V_{in} para o melhor caso da Tabela 6, em que o V_{mppc} é de 5 V.

Figura 90. Valor da potência de saída do conversor e valores de V_{in} e V_{out} ao longo de 6 ms ($V_{mppc} = 5V$, Carga = 50 Ω).

Os resultados obtidos anteriormente foram para um caso em que as condições meteorológicas são ideais, estando o módulo fotovoltaico a produzir a potência máxima. Para simular o módulo numa

situação em que as condições meteorológicas não são ideais, em que os valores de tensão e corrente diminuem e, conseqüentemente, o valor de MPP, foi necessário mudar alguns valores do circuito equivalente. Foi diminuída a corrente da fonte de corrente para 200 mA e foi modificado o fator de idealidade do diodo (N) para 3.1, obtendo assim uma tensão de MPP de 4.5 V. O valor de V_{mppc} foi mantido a 5 V. Foi introduzido na entrada do circuito LTC3129 o módulo com estas novas especificações. O resultado obtido está expresso na Figura 91.

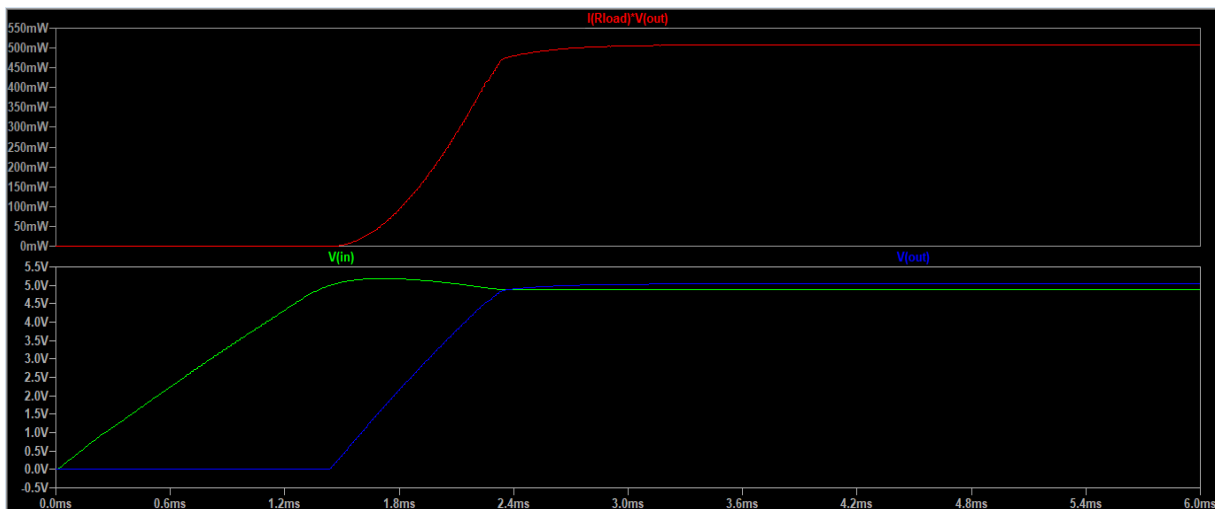


Figura 91. Valor da potência de saída na saída do conversor e valores de V_{in} e V_{out} ao longo de 6 ms ($V_{mppc} = 5V$, Carga = 50Ω).

Verifica-se que o conversor irá tentar manter a entrada perto do valor de V_{mppc} programado (5 V). Nesta situação verifica-se também a limitação da técnica de MPPC em relação à técnica de MPPT. Apesar de o valor da tensão não variar muito com a mudança de condições meteorológicas, ao contrário da corrente, vão na mesma existir situações em que o valor da tensão de MPP (4.5 V neste caso) se pode afastar bastante da tensão de V_{mppc} programada (5 V neste caso). Uma solução possível para este problema seria o uso de, por exemplo, três resistências de valores diferentes para R3 ou R5 (uma das resistências que controla o valor de V_{mppc}), estando sempre uma destas resistências ativadas dependendo do valor da irradiação solar medido. Estas teriam de ser calibradas anteriormente verificando qual seria o valor da tensão de MPP do módulo para uma determinada irradiação solar. A ativação de cada resistência poderia ser feita através de um microcontrolador e MOSFETs ou através do uso de comparadores.

4.3 Aquisição de dados

Através da aplicação Android desenvolvida é possível escolher a hora de início e fim de aquisição de dados, bem como o período de amostragem. Na Figura 92 é apresentado o ecrã da aplicação em que isto é possível. Aqui também são apresentados os últimos valores adquiridos pelos sensores, bem como o valor de RSSI e a capacidade atual da bateria. Foi feito um teste para aquisição de dados, iniciando e acabando a aquisição para apenas um minuto com um período de amostragem de 15 segundos.

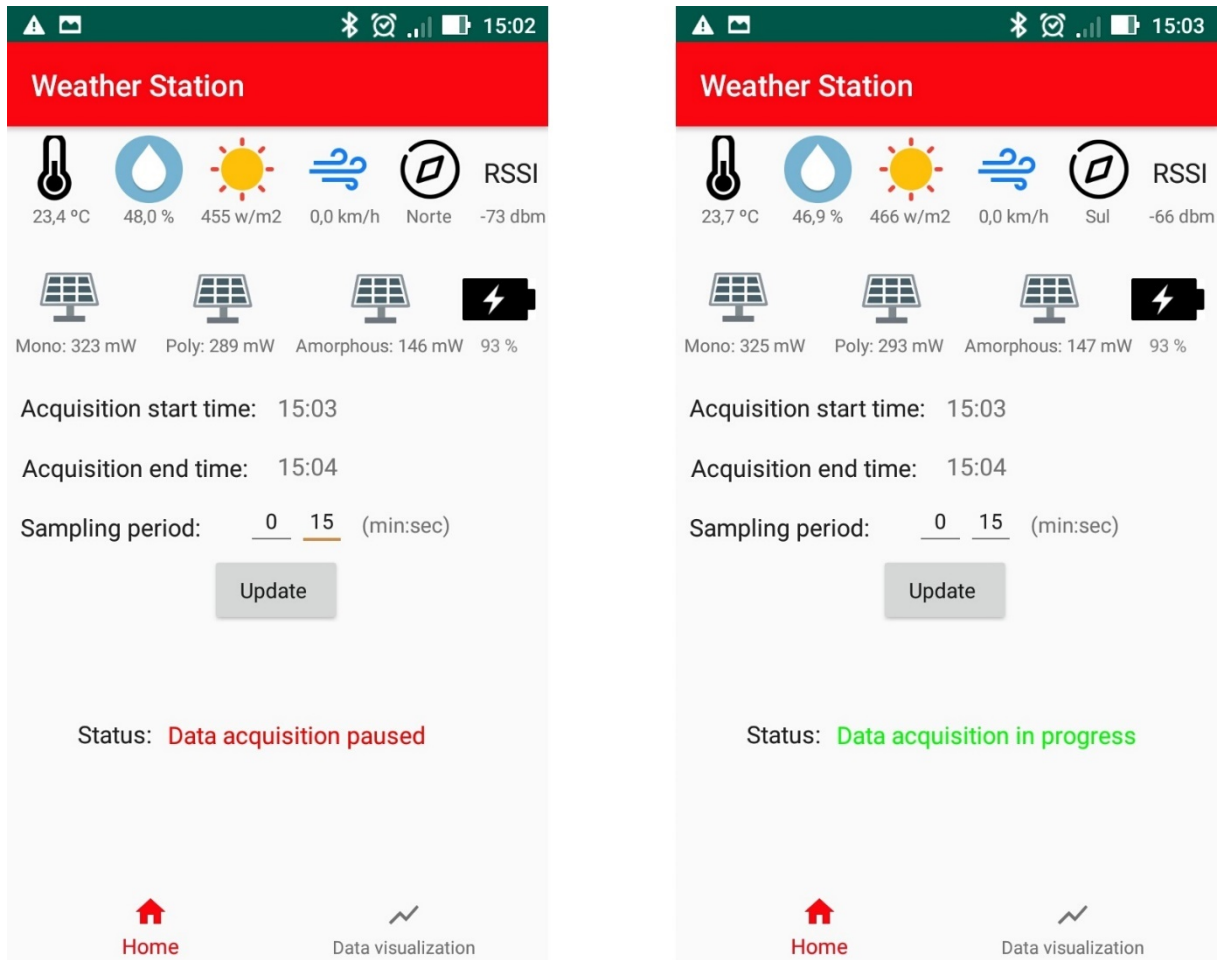


Figura 92. Ecrã da aplicação para iniciar aquisição de dados. Esquerda: Antes do início da aquisição. Direita: Depois do início da aquisição.

Na Figura 93 são apresentadas informações sobre a aquisição de dados obtidas num terminal para depuração através de uma conexão série. Inicialmente é verificado o número de dias existentes no ficheiro de texto info.txt. É criado um ficheiro de texto para o novo dia de aquisição. Os dados obtidos são depois armazenados neste ficheiro em formato de *string*.

```
Flag start = 1
Periodo de amostragem = 15
Inicio aquisicao de dados!

Acquisition
15:03:15 | 26/10/19
Media calculada
Nova data = 26/10/19
Ultima data = 22/10/19
Data counter = 3
Nova data
Day = 4
237 475 439 4 0 320 286 145 93
Hora fim de aquisicao: 15:04

Acquisition
15:03:30 | 26/10/19
Media calculada
237 468 463 4 0 324 292 146 93
Hora fim de aquisicao: 15:04

Acquisition
15:03:45 | 26/10/19
Media calculada
237 466 483 4 0 327 296 148 93
Hora fim de aquisicao: 15:04

Acquisition
15:04:00 | 26/10/19
Media calculada
237 465 491 4 0 329 298 148 93
Hora fim de aquisicao: 15:04
Flag_Start = 0
Fim da aquisicao de dados!
```

Figura 93. Teste sobre a aquisição de dados.

Na Figura 94 é apresentada a estação meteorológica conectada à placa de desenvolvimento através de uma comunicação UART, para ser feita a depuração de possíveis erros no sistema.

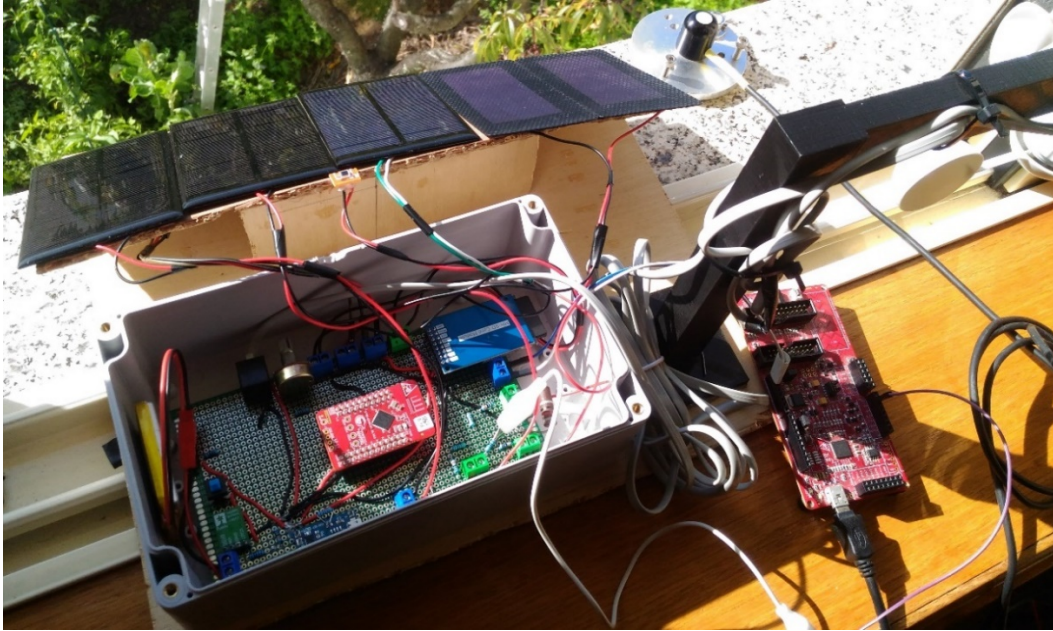


Figura 94. Estação meteorológica conectada à placa de desenvolvimento para depuração através de um terminal.

4.4 Transferência de dados e análise do desempenho

Na Figura 95 é apresentado o fragmento que gere a transferência de dados na aplicação Android. O utilizador pode escolher a data em que pretender fazer o *download* de dados, em que um diálogo para escolha de datas irá aparecer. Em alternativa pode ser ativado um interruptor em que serão transferidos todos os dados de todos os dias em que houve aquisição.

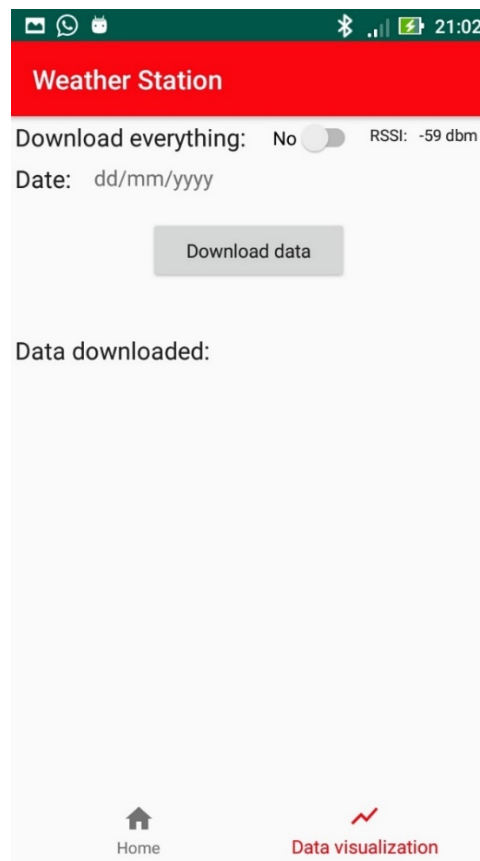


Figura 95. Página da aplicação em que se pode iniciar a transferência de dados.

Na Figura 96 é apresentado um exemplo dos diálogos que iram aparecer após o utilizador carregar no botão para iniciar o *download*. À medida que o dispositivo Android recebe os dados, uma barra de progresso será atualizada. Caso se tenha escolhido efetuar o *download* de todas as datas, irá aparecer o número de datas com dados que irão ser transferidas.

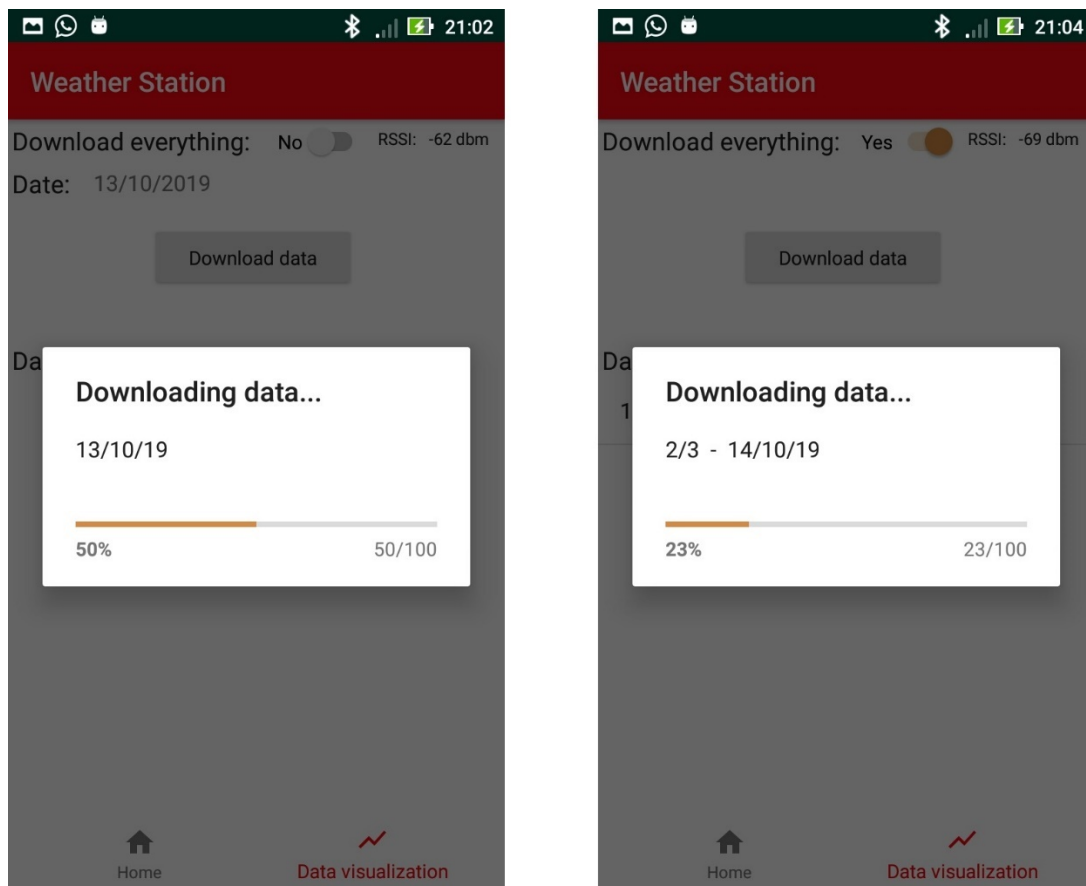


Figura 96. Diálogo do download de dados. Esquerda: para apenas uma data.
Direita: caso o utilizador escolha todas as datas disponíveis.

Após ser concluído o *download*, uma lista é atualizada com todas as datas transferidas e também com o início e fim de aquisição daquele dia (Figura 97). Pode-se então carregar numa data dessa lista para visualizar os dados, sendo iniciada a atividade de visualização de dados em gráficos.

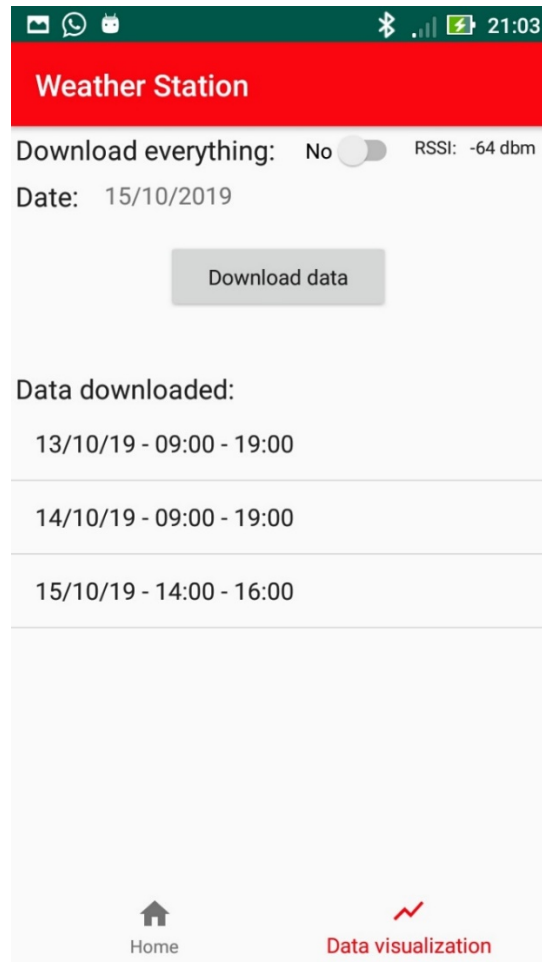


Figura 97. Página da transferência de dados após ser feito o download de dados.

Foi medido o tempo necessário para transferência de dados da estação para o dispositivo Android, que depende do número de bytes a serem transferidos. Este tempo pode ser decomposto em duas fases. Na primeira os dados são lidos do cartão de memória e guardados num buffer, prontos para serem transferidos. Na segunda fase os dados são lidos do buffer 500 bytes de cada vez e transferidos para o dispositivo Android através da rede sem fios BLE. Esta medição foi feita no dispositivo Android, desde que o utilizador carregue no botão para descarregar os dados até que os dados sejam todos transferidos, sendo este o tempo total que o utilizador espera para poder visualizar os dados. Para isso foi realizado um teste em que são transferidos os dados adquiridos durante um dia, desde as 9 horas da manhã até às 19 horas da tarde, com um período de amostragem de um minuto. Com nove dados de dois bytes (Secção 3.3.2 – Figura 54) a adquirir a cada minuto, o número de bytes a transferir será:

$$N^{\circ} \text{ bytes} = n^{\circ} \text{ horas} \times 60 \text{ min} \times 9 \text{ dados} \times 2 \text{ bytes} = 10 \times 60 \times 9 \times 2 = 10800 \quad (7)$$

Na Figura 98 é apresentado o cenário de teste descrito no parágrafo anterior.

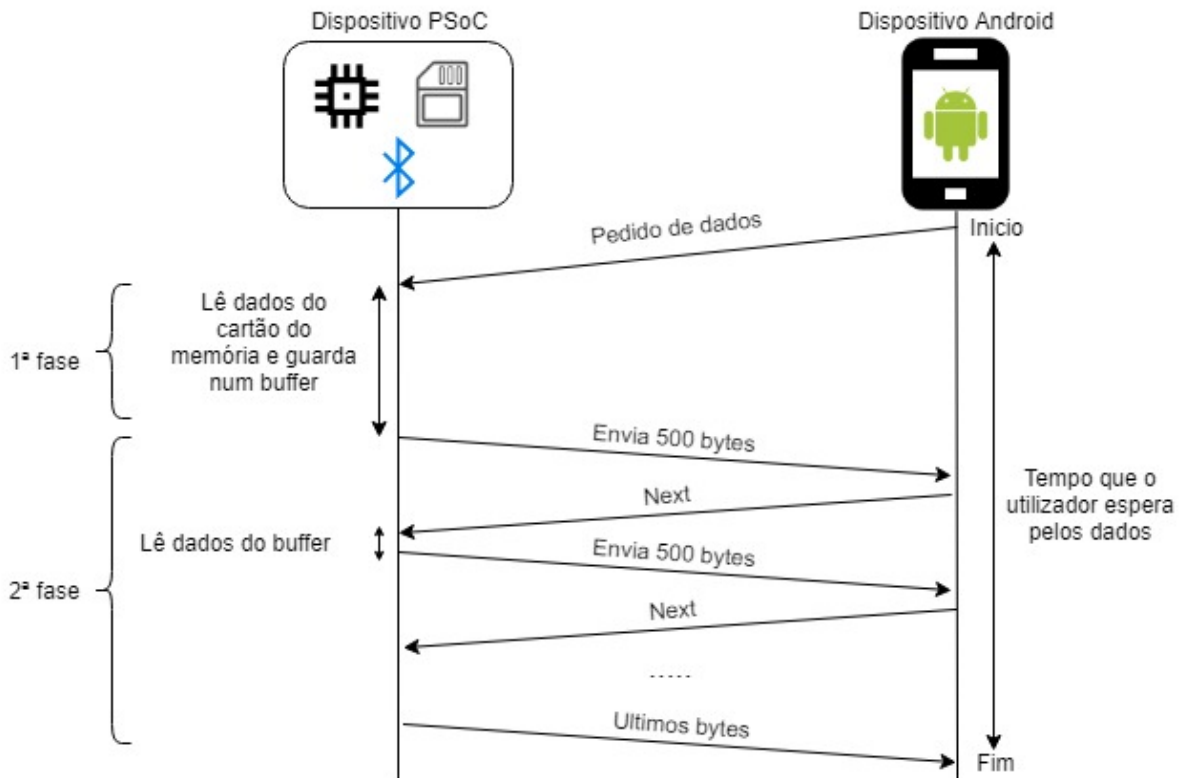


Figura 98. Cenário de teste para medição do tempo para a transferência de dados da estação para o dispositivo Android.

Para realizar este teste foi utilizada a função `System.elapsedRealtime()`, disponível no sistema Android, que armazena o tempo desde que a aplicação iniciou. Quando o utilizador carrega no botão para descarregar dados, o tempo atual é guardado numa variável chamada de `startTime`. Quando os dados são todos recebidos, o tempo atual é guardado numa outra variável chamada de `endTime`. A diferença entre o valor das duas variáveis fornece o tempo total que demorou para os dados serem transferidos para o dispositivo Android, do ponto de vista do utilizador.

Este teste foi repetido para outras situações, com diferentes quantidades de bytes a serem transferidos. Para cada situação o teste foi repetido dez vezes. O tamanho do MTU definido no dispositivo Android é de 500 bytes. A prioridade da conexão no dispositivo Android foi definida para `CONNECTION_PRIORITY_HIGH` com o uso da função `requestConnectionPriority()`. De mencionar que o dispositivo Android utilizado nestes testes é um Asus Zenfone 2 com a versão 6.0 do Android. Os resultados obtidos estão apresentados na Tabela 8 com o valor mínimo, médio e máximo em segundos, bem como o débito médio em kbps.

Tabela 8. Tempos obtidos (em segundos) na transferência de dados entre o dispositivo PSoC e o dispositivo Android.

Nº de bytes	Tempo mínimo	Tempo médio	Tempo máximo	Débito (kbps)
4320	4,958	5,174	5,381	6,68
10800	10,699	10,058	11,452	8,59
21600	24,298	24,531	24,904	7,13

No último caso, em que o número de bytes transferidos é o dobro do segundo caso, o tempo de transferência registado foi mais que o dobro. Isto deve-se ao facto de ter sido necessário ler o ficheiro de texto duas vezes, pois o buffer que contem os valores após serem lidos do cartão de memória é limitado devido à limitação do tamanho da SRAM do microcontrolador, tendo o buffer um tamanho máximo de 6480 para um tipo de inteiro de 16 bits, correspondendo a 12960 bytes. Como é necessário transferir 21600 bytes, o ficheiro de texto que contém os dados desse dia tem de ser lido duas vezes, acrescentando atraso na transferência de dados.

A seguir o tempo total foi decomposto em duas partes, que incluem o tempo de transferência através da rede BLE e o tempo de processamento inicial (ler dados do cartão de memória, ler dados do buffer entre outros). A medição do tempo de atraso devido ao BLE foi efetuada no dispositivo Android, desde que é executada a função que faz o pedido de leitura da característica que contem os dados no servidor GATT, até que a função `onCharacteristicRead()` seja chamada na *callback* para o correspondente UUID da característica. Esta característica tem um tamanho de 500 bytes. Neste caso não é considerada a transmissão do valor da característica Next. Na Figura 99 é descrito o cenário de teste.

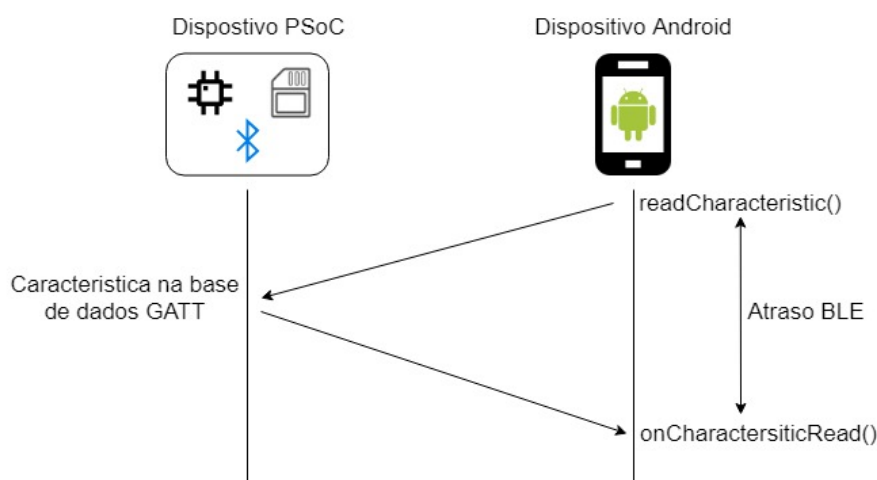


Figura 99. Cenário de teste para medição do atraso da rede BLE.

Ao longo da transferência de dados adquiridos de um dia (10800 bytes), foi medido este atraso para cada 500 bytes transferidos (22 vezes). A primeira situação foi feita sem o uso da função `requestConnectionPriority()`, e a segunda situação com o uso da função com a prioridade para `CONNECTION_PRIORITY_HIGH`. O MTU foi mantido a 500 bytes. Os resultados obtidos estão expressos na Tabela 9.

Tabela 9. Atraso (em segundos) na transferência de dados pela rede BLE para diferentes prioridades de conexão e débito obtido.

Prioridade da conexão	Nº de bytes	Tempo mínimo	Tempo médio	Tempo máximo	Débito (kbps)
Padrão (BALANCED)	500	0,186	0,329	0,542	12,16
HIGH	500	0,105	0,158	0,442	25,32

Fazendo o pedido de prioridade de conexão para `CONNECTION_PRIORITY_HIGH`, o débito aumenta para mais que o dobro da primeira situação com a prioridade por padrão. Isto deve-se provavelmente ao facto de que o intervalo de conexão é diminuído ou aumentado o numero de pacotes que envia por intervalo de conexão, apesar de isto não ser especificado na documentação [65]. Isto tem a desvantagem de usar mais energia que os outros tipos de prioridade. De acordo com [68], o mínimo intervalo de conexão que um dispositivo Android impõe é de 11,25 ms para a versão 6.0 do Android, podendo ser este o valor definido para uma conexão de alta prioridade.

Foi também testado o impacto do tamanho do *payload* da camada de ligação (corresponde ao campo Data da Figura 14 da secção 2.4.2.1), mudando este valor no dispositivo periférico através do PsoC Creator, mas não se verificou nenhuma alteração no débito. Uma possível explicação é o dispositivo central (*smartphone* Android) estar a ignorar esta configuração, estabelecendo um valor que é fixo para a conexão ou que muda apenas quando se altera a prioridade de conexão, em conjunto com o intervalo de conexão.

De acordo com [31], o débito obtido (*goodput*) para uma notificação e para um intervalo de conexão de 8 ms é de cerca 20 kbps e para duas notificações com um intervalo de conexão é de cerca de 43 kbps. No caso desta dissertação está a ser um algoritmo baseado nas propriedades de leitura e escrita de características e não de notificações. A segunda situação obtida na Tabela 9, com um débito de 25,32 Kbps, está entre os dois casos mencionados.

Em [68] foi medido o débito usando um *smartphone* com a mesma versão do Android usado nesta dissertação. O débito obtido na segunda situação da Tabela 9 vai de encontro aos resultados obtidos para um intervalo de conexão de 11,25 ms, quando são enviados dois a três pacotes por intervalo de conexão, apesar de nos testes realizados serem usadas notificações e não a propriedade de leitura como no caso desta dissertação.

O tamanho do MTU no dispositivo Android foi alterado para verificar o impacto deste parâmetro no débito da conexão BLE. Foi usado o mesmo cenário da Figura 99, transferindo 10800 bytes de dados e usando a prioridade de conexão de CONNECTION_PRIORITY_HIGH. Os resultados estão expressos na Tabela 10.

Tabela 10. Tempos de atraso (sem segundos) na transferência de dados pela rede BLE para diferentes MTUs.

MTU (bytes)	Nº de bytes	Tempo mínimo	Tempo médio	Tempo máximo	Débito (Kbps)
26	500	0,990	1,134	1,301	3,53
100	500	0,213	0,325	0,511	12,31
250	500	0,137	0,190	0,450	21,05
500	500	0,105	0,158	0,442	25,32

Foi também medido o tempo de leitura dos dados do cartão de memória. Este tempo é medido desde que é chamada a função que lê os dados do cartão de memória até que todos os dados sejam movidos para o buffer. Na Figura 100 é representado o cenário de teste. Inicialmente é lido o ficheiro info.txt para saber qual o número associado à data pedida e também saber qual a hora de início e fim de aquisição para esse dia. É lido o ficheiro de texto associado à data pedida para saber o número total de dados no ficheiro e enviar para o dispositivo Android. Os dados são depois lidos e guardados num buffer prontos para serem transferidos.

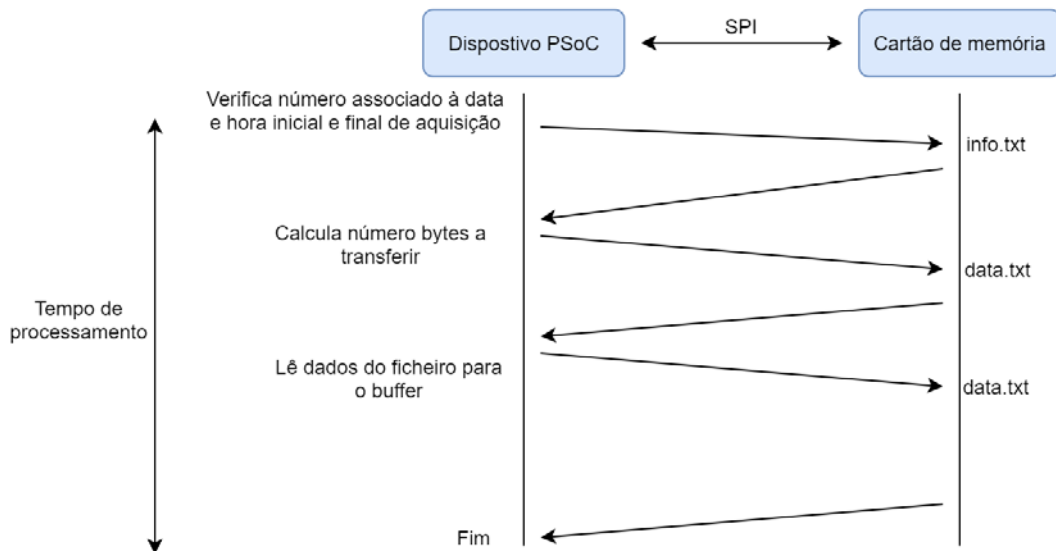


Figura 100. Cenário de teste para medição do tempo de leitura de dados do cartão de memória.

Na Tabela 11 são apresentados os resultados obtidos para diferentes quantidades de dados a ler.

Tabela 11. Tempo (em segundos) de leitura dos dados do cartão de memória.

Nº de bytes	Tempo mínimo	Tempo médio	Tempo máximo
4320	2,335	2,335	2,335
10800	5,035	5,036	5,036
21600	12,750	12,754	12,77

Como já explicado anteriormente, para o último caso existe um grande aumento no tempo de leitura devido à limitação do tamanho do buffer e, conseqüentemente, da necessidade de ler os dados do ficheiro de texto duas vezes.

Foram subtraídos os valores do tempo médio da Tabela 11 com os valores do tempo médio da Tabela 8, de modo a verificar qual o tempo total para transmissão de dados. Os resultados estão obtidos na Tabela 11.

Tabela 12. Tempo (em segundos) associado à transmissão de dados.

Nº de bytes	Tempo de transmissão médio	Débito (Kbps)
4320	2,839	12,17
10800	5,022	17,20
21600	11,777	14,67

Como se pode verificar, não é conseguido o mesmo débito para a segunda situação da Tabela 9. Isto deve-se a dois fatores que não foram medidos em nenhum dos casos em cima: tempo de processamento cada vez que é efetuada a leitura do buffer no dispositivo PSoC para enviar os próximos 500 bytes e o tempo de transmissão do valor da característica Next do dispositivo Android para o dispositivo PSoC.

4.5 Visualização de dados

Foram adquiridos dados durante o dia de 26 de outubro de 2019 entre as 16 horas e as 19 horas com um período de amostragem de 15 segundos, na localidade de Fafe do distrito de Braga. Na Figura 101 á apresentada a estação a adquirir dados no exterior.



Figura 101. Estação a adquirir dados no exterior.

Após se carregar numa das datas da lista na aplicação Android, é iniciada uma nova atividade dedicada à visualização de dados. Estes dados estão guardados na base de dados desenvolvida em SQLite. Através de um conjunto de caixas de seleção, pode-se escolher a variável a visualizar. O gráfico é ajustado automaticamente em função dos valores da variável a visualizar. No eixo das abcissas é representado o tempo desde o início até ao fim da aquisição. É possível fazer zoom no gráfico para

visualizar um determinado valor a uma determinada hora caso necessário. Na Figura 102 podemos ver os valores da temperatura e humidade medidos.

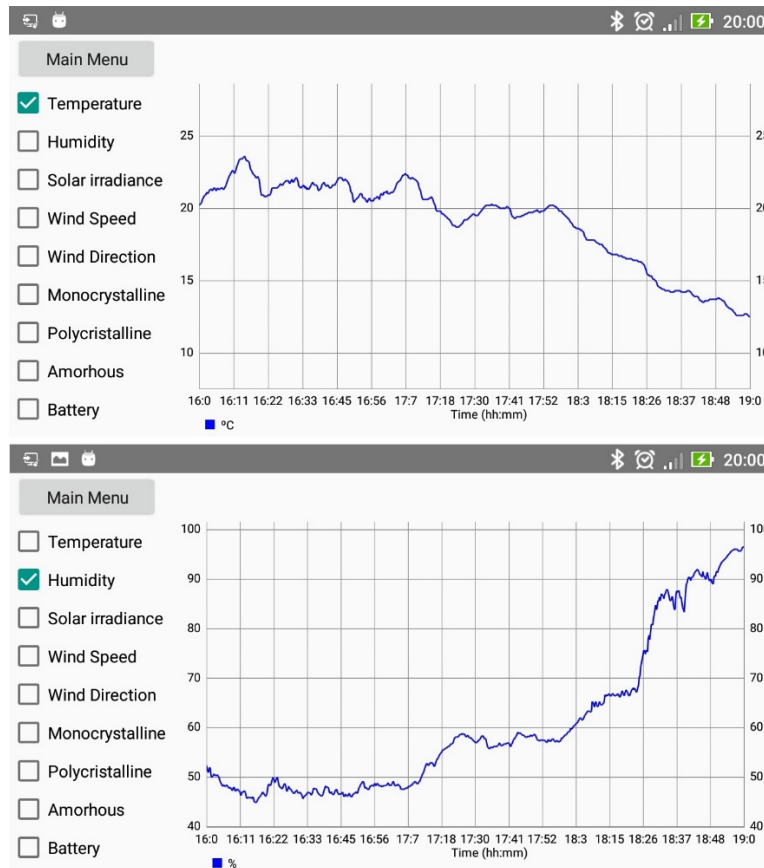


Figura 102. Em cima: Temperatura. Em baixo: Humidade.

Na Figura 103 são apresentados os valores da velocidade e direção do vento adquiridos. Estes valores podem ter sido afetados devido à posição da estação durante a aquisição de dados, por não ter estado num sítio com elevação suficiente, tendo os valores sido afetados pelo ambiente físico à volta.

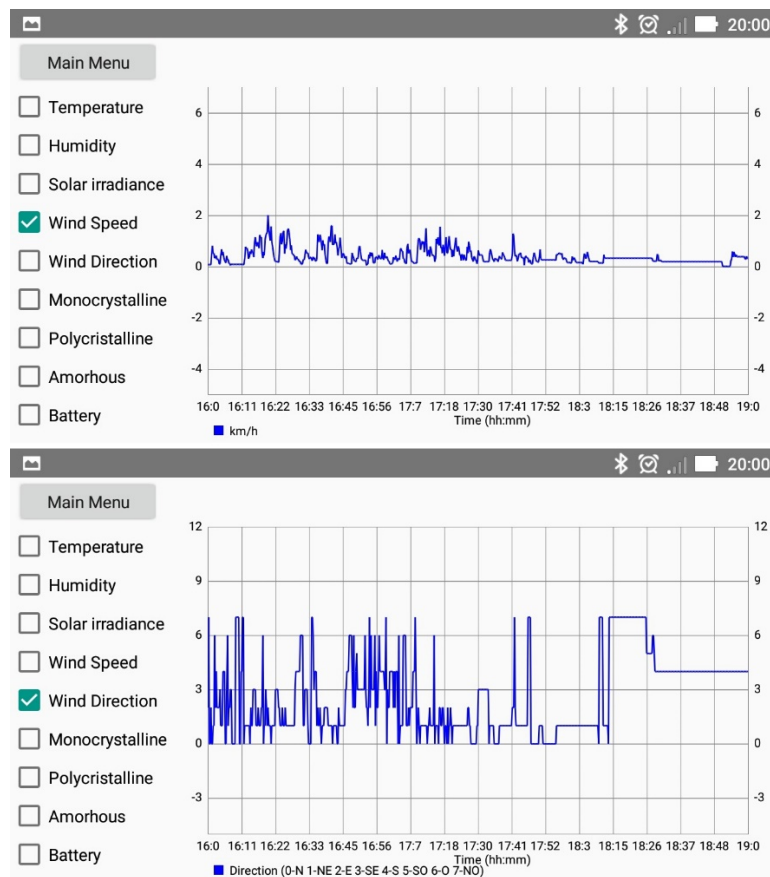


Figura 103. Em cima: Velocidade do vento. Em baixo: Direção do vento.

Na Figura 104 é apresentado o valor da irradiação solar medida.



Figura 104. Irradiação solar medida ao longo de três horas.

A medição da potência produzida pelos módulos fotovoltaicos tinha inicialmente como objetivo o uso do conversor LTC3129 com MPPC integrado, para maximizar a extração de energia. Infelizmente esta parte não foi possível implementar, mas foi provado o conceito através de simulações na secção 4.2. Os módulos fotovoltaicos usados foram conectados diretamente à resistência de carga, tendo esta o mesmo

valor (85Ω) para todos os módulos fotovoltaicos. Na Figura 105 é apresentado o valor da potência calculada para todos os módulos, de acordo com a fórmula da equação (9) da secção 3.2.5.

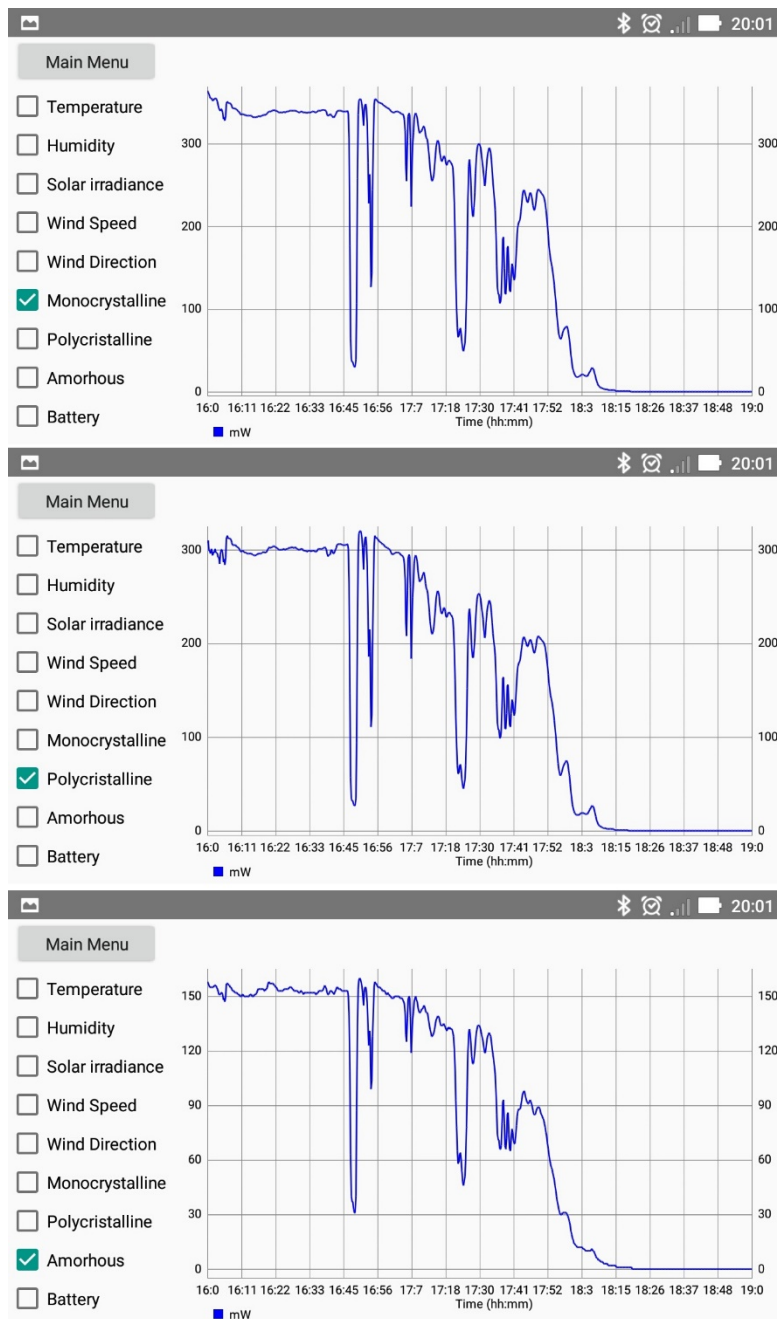


Figura 105. Potência produzida pelos módulos fotovoltaicos ao longo de três horas.

Na Figura 106 é apresentada a capacidade da bateria atual. Podemos notar que que estas medições não são fidedignas, devido a dois problemas. O primeiro é que a medição é feita simplesmente através da tensão atual da bateria, sendo que a tensão de uma bateria LiPo não é linear com a sua capacidade. Este problema pode ser resolvido medindo a capacidade da bateria através de um contador de Coulomb. O segundo, e o maior problema neste caso, é devido à tensão ser medida nos terminais da bateria

enquanto está a ser recarregada pelo módulo fotovoltaico. Este problema poderia ser resolvido utilizando um MOSFET para desligar o módulo do carregador da bateria quando é feita a medição da tensão da bateria, sendo ligado novamente para recarregar a bateria.

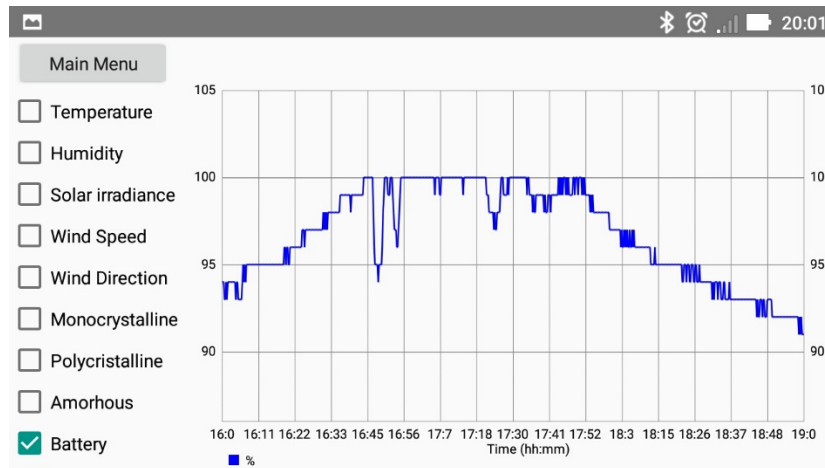


Figura 106. Capacidade atual da bateria.

4.5.1 Recarregamento da bateria

Os primeiros resultados obtidos apresentaram um problema em que a medida da tensão na bateria era afetada pelo módulo fotovoltaico. Este problema foi resolvido usando um MOSFET para desligar o módulo TP4056 da bateria, cada vez que é feita a medida da tensão da bateria pelo ADC. Foram adquiridos novos dados no dia 18 de novembro de 2019. Na Figura 107 é apresentado os resultados do recarregamento da bateria usando um módulo fotovoltaico de silício monocristalino e o módulo TP4056, e a correlação com a irradiação solar atual. Pode-se verificar o recarregamento da bateria caso as condições meteorológicas o permitam. Entre as 14:30 e as 15:00 houve uma diminuição da irradiação solar, que significa numa diminuição da potência produzida pelo módulo fotovoltaico. O módulo não irá produzir potência suficiente para acompanhar o consumo de energia do sistema. Aproximadamente às 16:30, com uma irradiação solar atual de aproximadamente 220 W/m^2 , o módulo fotovoltaico novamente deixa de produzir energia suficiente para acompanhar o consumo.

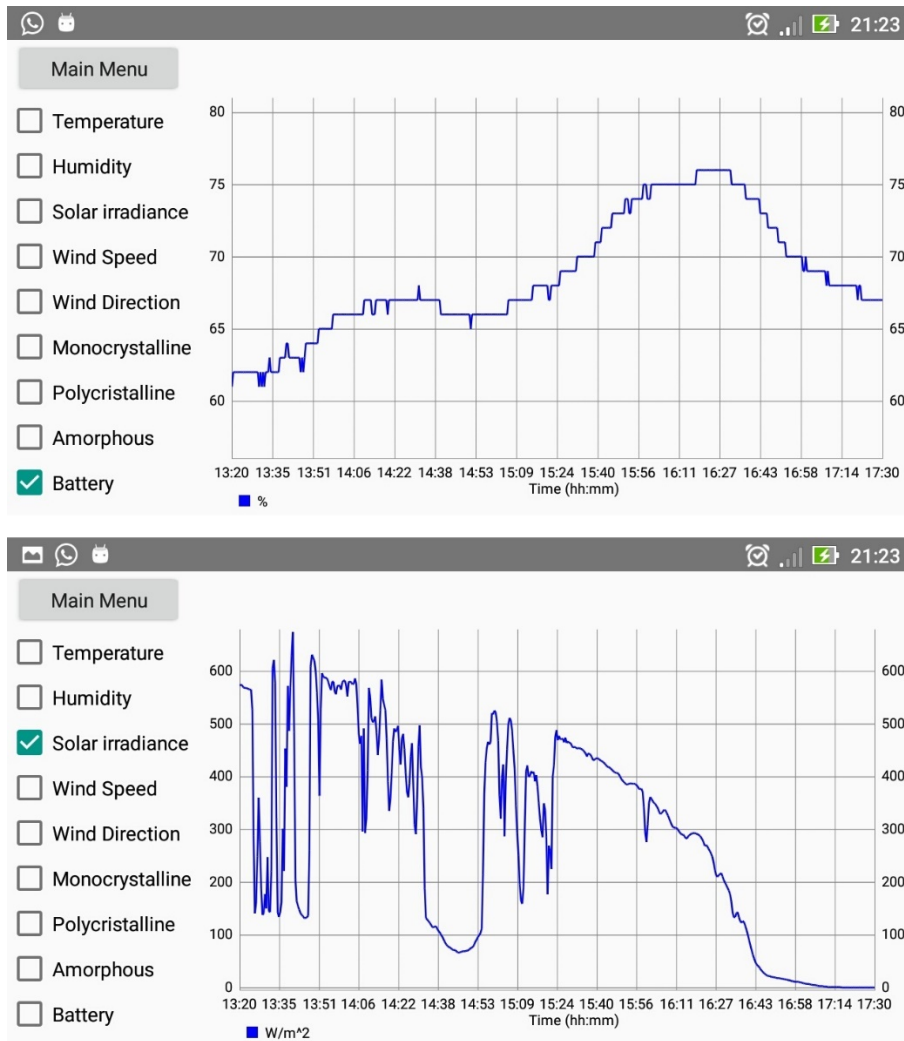


Figura 107. Correlação da capacidade atual da bateria com a irradiação atual.

4.5.2 Conclusões sobre a potência produzida

Na Figura 108 é apresentada a relação entre a irradiação solar medida e a potência produzida pelo módulo de silício monocristalino.

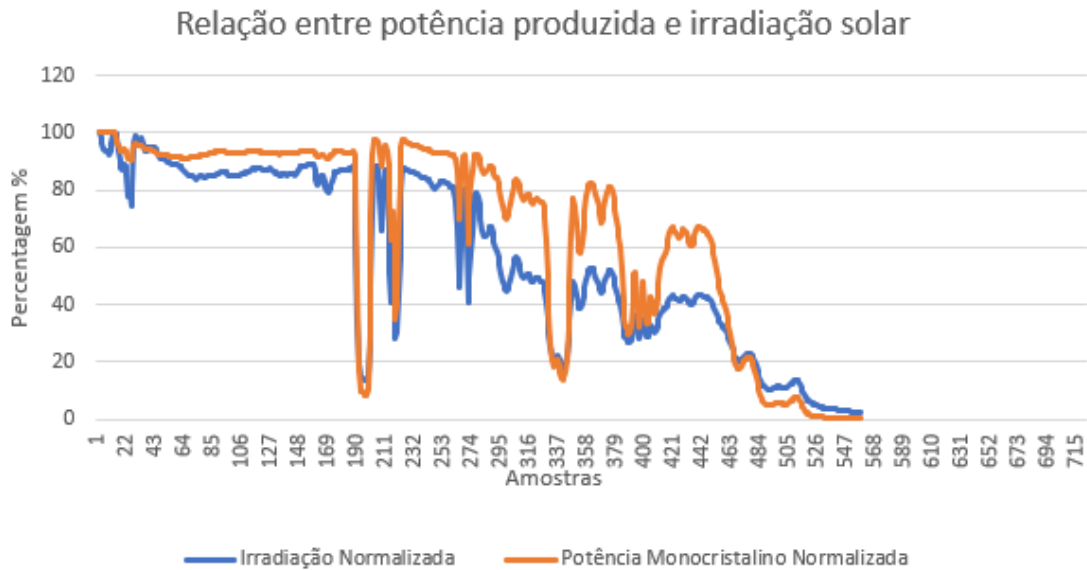


Figura 108. Relação entre potência produzida pelo módulo de silício monocristalino e a irradiação solar.

Como esperado, o valor da potência produzida pelo módulo depende bastante da irradiação solar atual. O mesmo acontece para os outros módulos. Este valor pode também depender da temperatura. A velocidade e direção do vento podem também influenciar a potência produzida, pelo facto de arrefecerem os módulos, diminuindo a temperatura deles. A resistência de carga pode também ser influenciada pela temperatura, e consequentemente, influenciando a medição da potência produzida.

Na Figura 109 e Figura 110 é apresentada a comparação da potência produzida entre as três tecnologias usadas, para os dias 26 de outubro de 2019 e 18 de novembro de 2019, respetivamente.

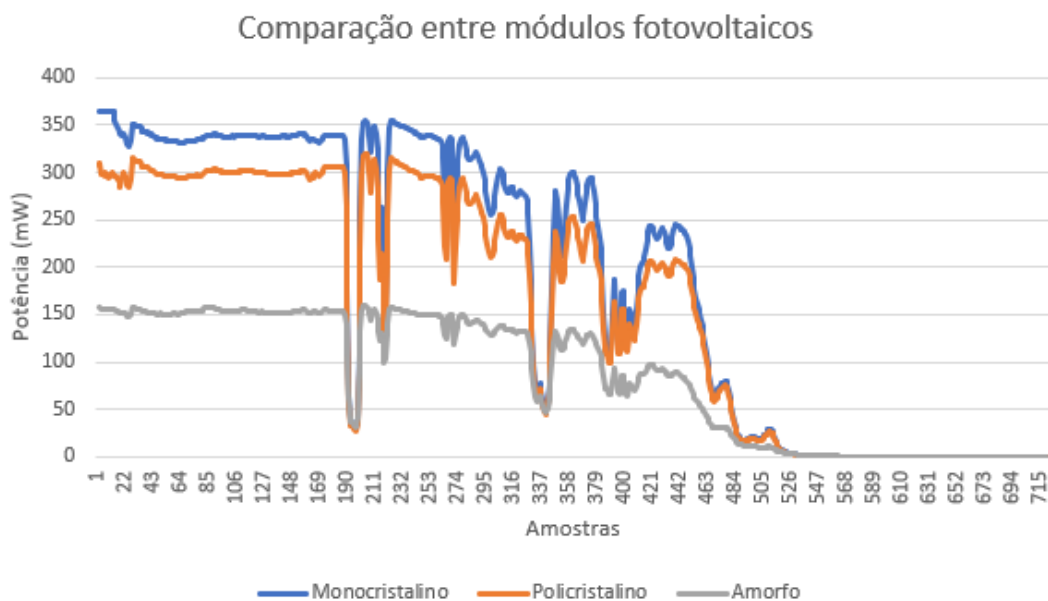


Figura 109. Comparação da potência produzida pelos três módulos fotovoltaicos (Data de aquisição: 26/10/2019).

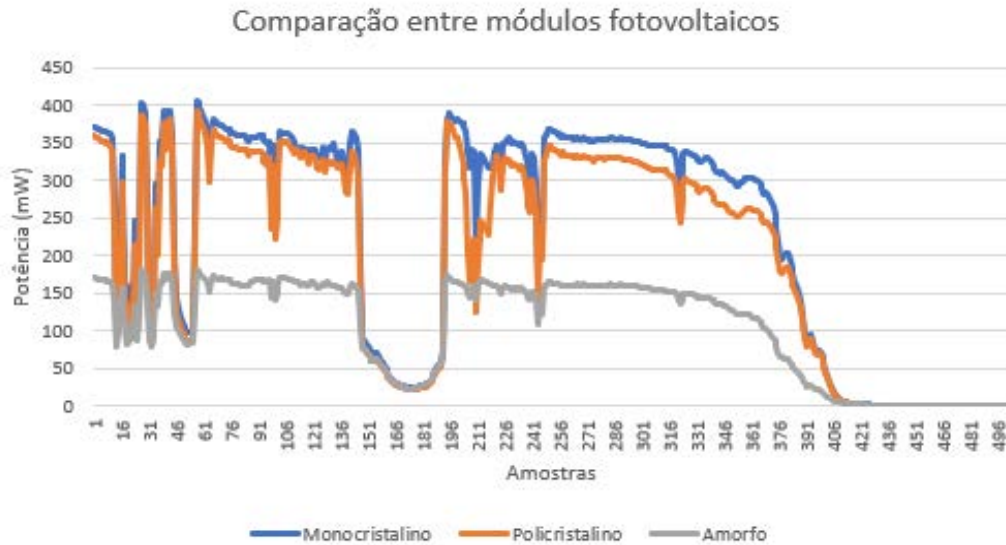


Figura 110. Comparação da potência produzida pelos três módulos fotovoltaicos (Data de aquisição: 18/11/2019).

Foi calculada a média da potência produzida pelos módulos para valores acima de 1 mW. Os valores obtidos estão expressos na Tabela 13.

Tabela 13. Média da potência produzida pelos módulos fotovoltaicos.

Data de aquisição	Potência Média (mW)			Irradiação Solar Média (W/m ²)	Temperatura Média (°C)
	Monocristalino	Policristalino	Amorfo		
26/10/19	238,06	207,81	108,68	301,33	19,0
18/11/19	256,53	234,87	120,37	323,38	10,3

Devido à diferença da área dos módulos fotovoltaicos (Tabela 5), foi calculado a potência produzida média por 1000 mm². Os resultados estão apresentados na Tabela 14.

Tabela 14. Média da potência produzida pelos módulos fotovoltaicos por 1000 mm².

Data de aquisição	Potência média por 1000 mm ² (mW)		
	Monocristalino	Policristalino	Amorfo
26/10/19	46,68	37,24	16,23
18/11/19	50,30	42,09	17,98

Pode-se concluir que a diferença na potência produzida entre as diferentes tecnologias está de acordo com a diferença entre as eficiências indicadas na Tabela 1, na secção 2.3.1. Pode-se verificar que a

diferença em percentagem da potência produzida pelo módulo de silício monocristalino e policristalino vai ser menor quando a temperatura é menor, e é maior quando a temperatura é maior. De acordo com [5], as tecnologias de silício cristalino têm uma descida na eficiência com o aumento da temperatura. Os resultados obtidos na Tabela 13 podem sugerir que com o aumento da temperatura, os módulos de silício policristalino podem ter uma descida maior na eficiência em comparação com os módulos de silício monocristalino. Isto vai de acordo com os resultados obtidos em [69], o qual se concluiu que as células de silício policristalino eram mais afetadas com o aumento da temperatura do que as células de silício monocristalino.

A tecnologia a escolher para um projeto de uma instalação de painéis solares terá de se ter em conta o espaço disponível para instalação de painéis, a potência necessária e as condições ambientais e meteorológicas do local da instalação. Tem de se considerar também o preço por Watt para cada tecnologia, sendo a tecnologia de silício monocristalino a mais cara e a de silício amorfo a mais barata. Para localizações em que a temperatura média anual seja relativamente alta, painéis de silício monocristalino talvez sejam a melhor opção. Esta tecnologia também poderá ser considerada para instalações com espaço limitado, mas que requer uma potência considerável. Caso não haja limitação de espaço, painéis de silício policristalino poderá ser a melhor opção. Em alguns casos em que é necessária a instalação de painéis numa superfície curva, ou necessidade de peso reduzido, painéis de silício amorfo é uma tecnologia a considerar. Esta última tecnologia também poderá ser uma solução para casos em que sejam necessárias potências na ordem das dezenas de mW, para por exemplo, casos de *energy harvesting* para alimentação de sensores. De realçar que nestes testes não foi usado nenhuma técnica de MPPT ou MPPC, em que a potência extra que pode ser extraída poderá ser diferente para diferentes tipos de tecnologia.

5. Conclusões

5.1 Conclusões

Nesta dissertação foi desenvolvida uma estação meteorológica de baixo custo dedicada às energias renováveis capaz de medir variáveis como a temperatura, humidade, irradiação solar e velocidade e direção do vento. Foram integrados três módulos fotovoltaicos de três tecnologias diferentes de silício, de modo a fazer uma comparação de como estes reagem a diferentes condições meteorológicas.

Todos os dados da estação foram adquiridos através de um microcontrolador embebido no módulo PSoC 4 BLE. Foram programados vários periféricos para acomodar esta aquisição de dados bem como seu processamento. Os dados recolhidos são armazenados localmente em ficheiros de texto através do uso de um cartão de memória.

Foi apresentada uma solução para a maximização da produção de energia, recorrendo ao conversor LTC3129 com MPPC integrado, em que seria usado um conversor para cada módulo fotovoltaico, de modo a maximizar a extração de energia. Na saída de cada conversor é utilizada uma resistência de carga de igual valor, sendo aqui medida a potência produzida por cada módulo. Apesar de esta solução não ter sido implementada na estação, o conceito foi provado recorrendo a simulações no programa LTSpice. A potência dos módulos foi medida sem a utilização do conversor. O módulo de silício monocristalino foi aquele que mostrou maior eficiência para as condições meteorológicas da localização.

Para alimentar toda a estação é utilizada uma bateria do tipo LiPo de 3.7 V. Foi usado também um conversor DC-DC de topologia *step-up-down* com uma saída de 3.3 V, de modo a poder alimentar o módulo PSoC 4 BLE e todos os sensores da estação. É usado um módulo fotovoltaico extra que pode recarregar a bateria caso as condições meteorológicas o permitam, através do uso de um carregador de baterias LiPo.

A transferência de dados para o utilizador foi feita recorrendo à tecnologia de rede sem fios BLE, sendo os dados transferidos para um dispositivo Android e guardados numa base de dados. Foram feitos testes sobre a transferência de dados, de modo a perceber o que causa maior atraso. O débito obtido

vai de acordo com resultados teóricos esperados. O tempo de processamento é o que causa maior atraso na transferência de dados, devido à leitura de dados dos ficheiros de texto.

A aplicação Android desenvolvida permite controlar a hora de início e fim de aquisição de dados bem como o período de amostragem. O utilizador pode aqui também escolher qual o dia em que pretende fazer a transferência de dados ou, em alternativa, fazer a transferência de dados de todos os dias em que houve aquisição de dados. O utilizador pode visualizar os dados recolhidos pela estação através de gráficos.

5.2 Sugestões de trabalho futuro

Para completar o trabalho desenvolvido nesta dissertação pode-se considerar as seguintes sugestões:

- Desenvolver uma PCB que integre toda a eletrónica da estação, inclusive o circuito de maximização da produção de energia utilizando o conversor DC-DC LTC3129. Devido a toda a eletrónica estar dentro de uma caixa, poderá ser usada uma antena exterior que conecte ao módulo PSoC 4 BLE.
- Poderão ser acrescentados mais sensores, como um sensor de chuva ou sensor de pressão atmosférica entre outros. Neste caso, teria que se ter em consideração também o limite de pinos para ADC ou pinos digitais, dado que todos os pinos do ADC já estão a ser usados no contexto desta dissertação, sendo assim necessário um outro microcontrolador ou um módulo ADC extra.
- Calcular a potência necessária para alimentar o sistema, e verificar quanto tempo o sistema poderia estar a funcionar assumindo que a bateria estava na capacidade máxima e não haveria carregamento da bateria. Com isto também seria possível verificar quanta potência seria necessária extrair dos módulos fotovoltaicos e dimensionar estes para a estação ser autossustentável. A capacidade da bateria a utilizar na estação poderia também ser dimensionada tendo em conta este fator.
- O sistema de recarregamento da bateria poderá ser melhorado. Em vez de utilizar um módulo fotovoltaico extra, pode ser desenvolvido um sistema que use os três módulos fotovoltaicos já existentes. Enquanto a medição de energia não esteja a ser efetuada, estes módulos poderiam estar a alimentar a bateria. Terminada a medição, um MOSFET poderia desligar a carga dos módulos e um outro MOSFET poderia fazer a ligação com o sistema de carregador de bateria + bateria LiPo.
- Poderão ser adicionados mais módulos fotovoltaicos de outras tecnologias.
- Desenvolver um suporte mecânico em que se possa alterar automaticamente o ângulo de inclinação dos módulos fotovoltaicos. Para isso, neste suporte poderia estar integrado um motor de passo ou

servomotor em que alterava o ângulo de inclinação de acordo com as definições do utilizador. Alternativamente a inclinação poderia ser alterada ligeiramente ao longo do dia de modo a possibilitar um estudo do impacto da inclinação na produção de energia.

- Pode-se acrescentar a possibilidade da estação de enviar os dados adquiridos para a Internet através do uso de um módulo 3G ou 4G. Apesar de isto de acrescentar custos na transferência de dados, o utilizador poderia decidir previamente quando a estação faz a transferência de dados através de uma conexão 3G/4G dependendo da disponibilidade do utilizador ir junto da estação.
- Explorar como minimizar o tempo necessário para transferir dados. No caso desta dissertação, o tempo de processamento (ler dados dos ficheiros, do buffer etc.) é o que causa mais atraso. Poderá também ser desenvolvido um algoritmo alternativo ao usado para transferência de dados através da rede BLE, em que sejam utilizadas notificações em vez da utilização da propriedade de leitura e escrita de características.

Referências

- [1] F. R. Pazheri, M. F. Othman, e N. H. Malik, «A review on global renewable electricity scenario», *Renew. Sustain. Energy Rev.*, vol. 31, pp. 835–845, 2014.
- [2] M. Q. Raza, M. Nadarajah, e C. Ekanayake, «On recent advances in PV output power forecast», *Sol. Energy*, vol. 136, pp. 125–144, 2016.
- [3] S. Lo Piano e K. Mayumi, «Toward an integrated assessment of the performance of photovoltaic power stations for electricity generation», *Appl. Energy*, vol. 186, pp. 167–174, 2017.
- [4] P. K. Nayak, S. Mahesh, H. J. Snaith, e D. Cahen, «Photovoltaic solar cell technologies: analysing the state of the art», *Nat. Rev. Mater.*, vol. 4, n. 4, pp. 269–285, 2019.
- [5] V. Perraki e P. Kounavis, «Effect of temperature and radiation on the parameters of photovoltaic modules», vol. 013102, n. 2016, 2017.
- [6] A. Weather, «Ambient Weather WS-2902A Smart Weather Station with WiFi Remote Monitoring and Alerts». [Em linha]. Disponível em: <https://www.ambientweather.com/amws2902.html>. [Acedido: 25-Ago-2019].
- [7] A. Weather, «Ambient Weather WS-2902A Smart Weather Station with WiFi Remote Monitoring and Alerts». .
- [8] D. Instruments, «EnviroMonitor System | Davis Instruments». [Em linha]. Disponível em: <https://www.davisinstruments.com/solution/enviromonitor-affordable-field-monitoring-system/#supportedensors>. [Acedido: 25-Ago-2019].
- [9] E. Ramsden, *Hall-Effect Sensors Theory and Application*. Newnes, 2006.
- [10] «Hot Wire Anemometer Principle - Instrumentation Tools». [Em linha]. Disponível em: <https://instrumentationtools.com/hot-wire-anemometer-principle/>. [Acedido: 26-Ago-2019].
- [11] W. C. Contributors, «File:Solar Spectrum and Spectral-Response.png», 2016. [Em linha]. Disponível em: https://commons.wikimedia.org/w/index.php?title=Special:CiteThisPage&page=File%3ASolar_Spectrum_and_Spectral-Response.png&id=221252821.

-
- [12] M. Al-Rasheedi, C. A. Gueymard, A. Ismail, e T. Hussain, «Comparison of two sensor technologies for solar irradiance measurement in a desert environment», *Sol. Energy*, vol. 161, n. December 2017, pp. 194–206, 2018.
- [13] M. Carrasco, A. Laudani, G. M. Lozito, F. Mancilla-David, F. R. Fulginei, e A. Salvini, «Low-Cost Solar Irradiance Sensing for PV Systems», *Energies*, vol. 10, n. 7, pp. 1–12, 2017.
- [14] O. Instruments, «The CMP3 Pyranometer is a Kipp & Zonen pyranometer». [Em linha]. Disponível em: <https://www.omniinstruments.co.uk/weather-stations-and-instruments/pyranometers-solar-irradiance/cmp3-pyranometer.html>. [Acedido: 25-Ago-2019].
- [15] O. Instruments, «The CMP3 Pyranometer is a Kipp & Zonen pyranometer». .
- [16] E. Tutorials, «Temperature Sensor Types for Temperature Measurement». [Em linha]. Disponível em: https://www.electronics-tutorials.ws/io/io_3.html. [Acedido: 28-Ago-2019].
- [17] K. W. Böer, «Solar cells - Chemistry Encyclopedia». [Em linha]. Disponível em: <http://www.chemistryexplained.com/Ru-Sp/Solar-Cells.html>. [Acedido: 12-Ago-2019].
- [18] J. D. Jordan Hanania, Kailyn Stenhouse, «Photovoltaic effect - Energy Education», 2015. [Em linha]. Disponível em: https://energyeducation.ca/encyclopedia/Photovoltaic_effect. [Acedido: 24-Ago-2019].
- [19] EMA, «Handbook for Solar Photovoltaic Systems», *Energy Mark. Authority, Singapore Publ.*, 2011.
- [20] J. Cubas, S. Pindado, e C. De Manuel, «Explicit expressions for solar panel equivalent circuit parameters based on analytical formulation and the Lambert W-function», *Energies*, vol. 7, n. 7, pp. 4098–4115, 2014.
- [21] S. Zahurul, N. Mariun, M. L. Othman, H. Hizam, I. Z. Abidin, e A. Toudeshki, «Ambient temperature effect on Amorphous Silicon (A-Si) Photovoltaic module using sensing technology», *Proc. Int. Conf. Sens. Technol. ICST*, vol. 2016-March, n. April 2014, pp. 235–241, 2016.
- [22] M. A. M. Ramli, E. Prasetyono, R. W. Wicaksana, N. A. Windarko, K. Sedraoui, e Y. A. Al-Turki, «On the investigation of photovoltaic output power reduction due to dust accumulation and weather conditions», *Renew. Energy*, vol. 99, pp. 836–844, 2016.
- [23] J. K. Kaldellis, M. Kapsali, e K. A. Kavadias, «Temperature and wind speed impact on the efficiency of PV installations. Experience obtained from outdoor measurements in Greece», *Renew. Energy*, vol. 66, n. May 2016, pp. 612–624, 2014.

-
- [24] Y. S. Khoo *et al.*, «Optimal orientation and tilt angle for maximizing in-plane solar irradiation for PV applications in Singapore», *IEEE J. Photovoltaics*, vol. 4, n. 2, pp. 647–653, 2014.
- [25] B. Subudhi e R. Pradhan, «A comparative study on maximum power point tracking techniques for photovoltaic power systems», *IEEE Trans. Sustain. Energy*, vol. 4, n. 1, pp. 89–98, 2013.
- [26] T. Barcelo, «Techniques to Maximize Solar Panel Power Output». [Em linha]. Disponível em: <https://www.analog.com/en/technical-articles/techniques-to-maximize-solar-panel-power-output.html>. [Acedido: 12-Nov-2019].
- [27] B. S. Kim *et al.*, «Wireless.Energy.Harvesting», vol. 102, n. 11, 2014.
- [28] D. Larrabeiti e J. A. Maestro, «Performance Evaluation of Energy Efficient Ethernet», vol. 13, n. 9, pp. 697–699, 2009.
- [29] D. J. Deng, S. Y. Lien, J. Lee, e K. C. Chen, «On Quality-of-Service Provisioning in IEEE 802.11ax WLANs», *IEEE Access*, vol. 4, n. c, pp. 6086–6104, 2016.
- [30] U. Raza, P. Kulkarni, e M. Sooriyabandara, «Low Power Wide Area Networks: An Overview», *IEEE Commun. Surv. Tutorials*, vol. 19, n. 2, pp. 855–873, 2017.
- [31] J. A. Afonso, A. J. F. Maio, e R. Simoes, «Performance Evaluation of Bluetooth Low Energy for High Data Rate Body Area Networks», *Wirel. Pers. Commun.*, vol. 90, n. 1, pp. 121–141, 2016.
- [32] M. Collotta, G. Pau, T. Talty, e O. K. Tonguz, «Bluetooth 5: A Concrete Step Forward toward the IoT», *IEEE Commun. Mag.*, vol. 56, n. 7, pp. 125–131, 2018.
- [33] Cypress Semiconductor, «Getting Started with PSoC®4 BLE». [Em linha]. Disponível em: <https://www.cypress.com/file/141171/download>. [Acedido: 28-Ago-2019].
- [34] PunchThrough, «Maximizing BLE Throughput», 2017. [Em linha]. Disponível em: <https://punchthrough.com/maximizing-ble-throughput-part-2-use-larger-att-mtu-2/>. [Acedido: 10-Nov-2019].
- [35] Cypress Semiconductor, «Getting Started with PSoC®4 BLE». .
- [36] «Google play store: number of apps 2019», 2019. [Em linha]. Disponível em: <https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/>. [Acedido: 24-Ago-2019].
- [37] A. Developers, «Start Another Activity |Android Developers». [Em linha]. Disponível em: <https://developer.android.com/training/basics/firstapp/starting-activity>. [Acedido: 03-Set-

- 2019].
- [38] Android, «Understanding the Activity Lifecycle». [Em linha]. Disponível em: <https://developer.android.com/guide/components/activities/activity-lifecycle>. [Acedido: 30-Ago-2019].
- [39] C. Ranhotigamage e S. C. Mukhopadhyay, «Field trials and performance monitoring of distributed solar panels using a low-cost wireless sensors network for domestic applications», *IEEE Sens. J.*, vol. 11, n. 10, pp. 2583–2590, 2011.
- [40] A. Kovacs, A. Nicolcioiu, J. Arhip, e G. Cașu, «Design and implementation of a GPRS remote data logger for weather forecasting», *IEEE Int. Conf. Commun.*, 2014.
- [41] D. A. Aponte-Roa, L. B. Montalvan, C. Velazquez, A. A. Espinoza, L. F. Velazquez, e R. Serrano, «Evaluation of a low-cost, solar-powered weather station for small-scale wind farm site selection», *I2MTC 2018 - 2018 IEEE Int. Instrum. Meas. Technol. Conf. Discov. New Horizons Instrum. Meas. Proc.*, pp. 1–5, 2018.
- [42] A. M. S. M. P. Coutinho, «Monitorização de painéis solares fotovoltaicos e condições meteorológicas», Universidade do Minho, 2016.
- [43] R. M. T. de Oliveira, «SWS - Estação Meteorológica Auto-Sustentável», Universidade do Minho.
- [44] Cypress Semiconductor, «CY8CKIT-042-BLE-A Bluetooth Low Energy 4.2 Compliant Pioneer Kit». [Em linha]. Disponível em: <https://www.cypress.com/documentation/development-kitsboards/cy8ckit-042-ble-bluetooth-low-energy-42-compliant-pioneer-kit>. [Acedido: 09-Jul-2019].
- [45] Cypress Semiconductor, «Bluetooth® Low Energy (BLE) Pioneer Kit Guide Cypress Semiconductor». [Em linha]. Disponível em: <https://www.cypress.com/file/234851/download>. [Acedido: 27-Ago-2019].
- [46] WaveShare, «DHT22 Temperature-Humidity Sensor». [Em linha]. Disponível em: https://www.waveshare.com/wiki/DHT22_Temperature-Humidity_Sensor.
- [47] M. K. Islam, T. Ahammad, E. H. Pathan, A. N. M. M. Haque, e M. R. H. Khandokar, «Analysis of Maximum Possible Utilization of Solar Radiation on a Solar Photovoltaic Cell with a Proposed Model», *Int. J. Model. Optim.*, vol. 1, n. 1, pp. 66–69, 2011.
- [48] Vishay Semiconductors, «bpw34.pdf». [Em linha]. Disponível em: <https://www.vishay.com/docs/81521/bpw34.pdf>. [Acedido: 09-Jul-2019].

-
- [49] J. Caldwell, «TI Designs – Precision : Verified Design 1 MHz , Single-Supply , Photodiode Amplifier Reference Design», n. November, pp. 1–21, 2014.
- [50] Apogee Instruments, «PYR_webmanual.pdf». [Em linha]. Disponível em: https://www.apogeeinstruments.com/content/PYR_webmanual.pdf. [Acedido: 26-Jul-2019].
- [51] Linear Technology, «LTC3129: 15V, 200mA Synchronous Buck-Boost DC/DC Converter with 1.3 μ A Quiescent Current», pp. 1–30, 2013.
- [52] C. Batteries, «PolymerLithium-IonBattery». [Em linha]. Disponível em: https://www.tme.eu/Document/ecd7153f9cded8de088b1cfe0a577062/lp903450_1600mah_pcm_2.54.pdf. [Acedido: 17-Nov-2019].
- [53] NanJing Top Power ASIC Corp., «TP4056». [Em linha]. Disponível em: <https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Prototyping/TP4056.pdf>. [Acedido: 27-Jul-2019].
- [54] «Pololu Adjustable Step-Up/Step-Down Voltage Regulator S7V8A». [Em linha]. Disponível em: <https://www.pololu.com/product/2118>. [Acedido: 20-Out-2019].
- [55] Texas Instruments, «High-Performance Battery Monitoring IC With Coulomb Counter». [Em linha]. Disponível em: <https://www.ti.com/lit/ds/slus521b/slus521b.pdf>. [Acedido: 19-Nov-2019].
- [56] I. Rectifier, «IRLZ44N». [Em linha]. Disponível em: <http://www.irf.com/product-info/datasheets/data/irlz44n.pdf>. [Acedido: 14-Nov-2019].
- [57] «Tinkercad | Create 3D digital designs with online CAD». [Em linha]. Disponível em: <https://www.tinkercad.com>. [Acedido: 17-Jul-2019].
- [58] «Time period measurement in PSoC 4». [Em linha]. Disponível em: <https://community.cypress.com/docs/DOC-11017>. [Acedido: 14-Abr-2019].
- [59] Narashiman, «GitHub - Narashiman/PSoC4_DHT11». [Em linha]. Disponível em: https://github.com/Narashiman/PSoC4_DHT11. [Acedido: 05-Fev-2019].
- [60] Elm-chan.org, «FatFs - Generic FAT Filesystem Module». [Em linha]. Disponível em: http://elm-chan.org/fsw/ff/00index_e.html. [Acedido: 26-Jul-2019].
- [61] Hackingchips, «GitHub - hackingchips/PSoC_FatFs». [Em linha]. Disponível em: https://github.com/hackingchips/PSoC_FatFs. [Acedido: 02-Mar-2019].
- [62] Android, «Bluetooth Low Energy Overview | Android Developers». [Em linha]. Disponível em:

-
- <https://developer.android.com/guide/topics/connectivity/bluetooth-le>. [Acedido: 18-Set-2019].
- [63] Cypresssemiconductor, «GitHub - cypresssemiconductor/bleapp». [Em linha]. Disponível em: <https://github.com/cypresssemiconductorco/bleapp>. [Acedido: 17-Nov-2019].
- [64] P. Jahoda, «MPAndroidChart». [Em linha]. Disponível em: <https://github.com/PhilJay/MPAndroidChart>. [Acedido: 16-Set-2019].
- [65] Android Developers, «BluetoothGatt | Android Developers». [Em linha]. Disponível em: [https://developer.android.com/reference/android/bluetooth/BluetoothGatt.html#requestConnectionPriority\(int\)](https://developer.android.com/reference/android/bluetooth/BluetoothGatt.html#requestConnectionPriority(int)). [Acedido: 23-Out-2019].
- [66] «Broadcasts overview | Android Developers». [Em linha]. Disponível em: <https://developer.android.com/guide/components/broadcasts>. [Acedido: 10-Out-2019].
- [67] «Solar Angle Calculator». [Em linha]. Disponível em: <http://www.solarelectricityhandbook.com/solar-angle-calculator.html>. [Acedido: 16-Out-2019].
- [68] J. Tosi, F. Taffoni, M. Santacatterina, R. Sannino, e D. Formica, «Throughput Analysis of BLE Sensor Network for Motion Tracking of Human Movements», *IEEE Sens. J.*, vol. 19, n. 1, pp. 370–377, 2019.
- [69] J. Adeeb, A. Farhan, e A. Al-Salaymeh, «Temperature Effect on Performance of Different Solar Cell Technologies», *J. Ecol. Eng.*, vol. 20, n. 5, pp. 249–254, 2019.