

UNIVERSIDADE DE TAUBATÉ

Henrique Tonzar Rubiatti

**ESTAÇÃO DE MONITORAMENTO DE PARÂMETROS
AMBIENTAIS**

Taubaté-SP

2018

Henrique Tonzar Rubiatti

**ESTAÇÃO DE MONITORAMENTO DE PARÂMETROS
AMBIENTAIS**

Trabalho de Graduação, apresentado ao
Departamento de Engenharia Elétrica da Universidade de
Taubaté para obtenção do Título de Engenheiro Eletricista.

Orientador: Prof. Dr. Ruy Morgado de Castro

Taubaté-SP

2018

SIBi – Sistema Integrado de Bibliotecas / UNITAU

R896e Rubiatti, Henrique Tonzar
Estação de monitoramento de parâmetros ambientais / Henrique Tonzar
Rubiatti. -- 2018.
52 f. : il.

Monografia (graduação) – Universidade de Taubaté, Departamento de
Engenharia Mecânica e Elétrica, 2018.
Orientação: Prof. Dr. Ruy Morgado de Castro, Instituto Básico de Exatas.

1. Arduino. 2. I²C. 3. Parâmetros ambientais. I. Título. II. Graduação
em Engenharia Elétrica e Eletrônica.

CDD – 629.8

Henrique Tonzar Rubiatti

**ESTAÇÃO DE MONITORAMENTO DE PARÂMETROS
AMBIENTAIS**

Trabalho de Graduação, apresentado ao
Departamento de Engenharia Elétrica da Universidade de
Taubaté para obtenção do Título de Engenheiro Eletricista.

Data: 29 de novembro de 2018

Resultado: _____

BANCA EXAMINADORA

Prof. Dr. Ruy Morgado de Castro

Assinatura _____

Prof. Rubens Castilho Jr

Assinatura _____

Prof. Dr. Mauro Pedro Peres

Assinatura _____

À Minha Família.

Agradecimentos

Ao meu orientador, *Prof. Dr. Ruy Morgado Castro* que sempre esteve disposto a compartilhar seus conhecimentos. Sem a sua orientação, dedicação e auxílio, o estudo aqui apresentado seria praticamente impossível.

Ao Instituto de Estudos Avançados – IEAV, no qual este trabalho foi realizado. Agradeço, também, ao CNPq pela bolsa de iniciação tecnológica que me permitiu desenvolver esse projeto.

*“Tudo o que é incompreensível, nem
por isso deixa de existir.”*

Blaise Pascal

RESUMO

Este trabalho teve como objetivo o desenvolvimento de uma Estação de Monitoramento de parâmetros ambientais que está sendo utilizada no Laboratório de Radiometria e Caracterização de Sensores Eletro-ópticos – LaRaC, do Instituto de Estudos Avançados – IEAv, a fim de monitorar parâmetros de interesse como: temperatura, pressão, umidade, iluminação, e concentração de CO₂ a fim de obter um maior controle da reprodutibilidade dos experimentos lá realizados. A Estação foi desenvolvida utilizando-se a plataforma ARDUINO para que os custos do projeto não fossem elevados. Para a comunicação com os sensores foi utilizado o protocolo de comunicação I²C, da Phillips, que utiliza apenas duas portas de comunicação: SCL e SDA, uma vez que o número de portas do ARDUINO não seria suficiente para comportar todos os sensores. A Estação grava as informações obtidas em um cartão de memória, e exibe as informações mais relevantes em uma tela de cristal líquido e os disponibiliza na rede do instituto para que se crie um banco de dados para que se possa verificar possíveis alterações ambientais e determinar se são naturais ou resultado de algum mau funcionamento dos equipamentos de controle do laboratório como: refrigeração, desumidificadores de ar, iluminação durante as medições. As medições já realizadas indicam um bom funcionamento dos equipamentos. Este controle, eventualmente, possibilitará o estudo da correlação entre os parâmetros ambientais do laboratório e as medições de reflectância, diminuindo as incertezas nas medições.

Palavras-chave: ARDUINO. I²C. Parâmetros Ambientais.

ABSTRACT

This project describes the development of an Environmental Parameters Monitoring Station that is being used in the Laboratório de Radiometria e Caracterização de Sensores Eletro-ópticos – LaRaC, from Instituto de Estudos Avançados – IEAv to monitor parameters of interest such as: temperature, pressure, humidity, illumination and CO₂ concentration so we can have a higher control of reproducibility of the experiments that are realized in there. The station was developed utilizing an ARDUINO platform to keep a low-cost Project. To connect with all the sensors involved it was used the I²C communication protocol, from Phillips, that only uses 2 communication gates: SCL and SDA, once the number of I/O pins wouldn't be enough. The station will record all the data gathered in a memory card, will show the most important information in a LCD display and will provide this data on the institute network to be able to create a data bank to check environmental alterations and verify if they are natural or if they are result of some bad-functioning control devices from the laboratory such as: air conditioning system, dehumidifiers or illumination. These measures will allow the study of the correlation between the environmental parameters of the laboratory and the reflectance, diminishing the uncertainty of the measures. The measurements already made showed a good performance of the equipment's. This control will allow the study and correlation between environmental parameters and the reflectance measurements lowering the uncertainty of the measurements.

Key words: ARDUINO. I²C. Environmental Parameters.

LISTA DE FIGURAS

Figura 1 - Geometria de aquisição da medição da reflectância, onde um alvo é iluminado com irradiância (dE) a partir de ângulos zenital e azimutal específicos, e o sensor registra a radiância (dL) que sai do alvo em ângulos azimutal e zenital específico.....	13
Figura 2 - Goniômetro do LaRaC e suas principais estruturas constituintes.....	14
Figura 3 - Foto da medição de amostras de: a) grama sintética; e b) asfalto.....	16
Figura 4 - Estimativa do Fator de Reflectância de uma placa de Spectralon.....	16
Figura 5 - Transmitância atmosférica para uma altitude em relação ao solo de 1 km, e moléculas absorvedoras.....	18
Figura 6 - Modelos de ARDUINO: a) UNO; b) MEGA; c) NANO; e d) MKR WiFi 1010.....	25
Figura 7 - Diagrama do barramento I ² C com os resistores <i>pull-ups</i>	26
Figura 8 - Diagrama de comunicação entre <i>master</i> e <i>slaves</i>	27
Figura 9 - Diagrama dos estados digitais.....	27
Figura 10 - Diagrama de transferência de dados entre master e slave.....	28
Figura 11 - Gráfico da dependência da umidade com a temperatura.....	30
Figura 12 - Fotografia dos sensores DHT22, BH1750, CSS811 e INA219.....	30
Figura 13 - Desenvolvimento do <i>shield</i> no <i>software</i> Fritzing.....	34
Figura 14 - Gráfico da temperatura e umidade relativa de um sensor DTH22 e no gerador de umidade, modelo Thunder 2500, para medições realizadas a cada 30 segundos.....	35
Figura 15 - Placa desenvolvida e conectada ao ethernet <i>shield</i> , que por sua vez estava conectado a um ARDUINO MEGA.....	36
Figura 16 - Montagem da Estação Monitoramento com as vistas interna e externa.....	36
Figura 17 - Monitor serial da IDE do ARDUINO exibindo os dados obtidos pelos sensores..	37
Figura 18 - Gráfico de parâmetros ambientais obtidos com os sensores da Estação de Monitoramento.....	38

SUMÁRIO

1	INTRODUÇÃO.....	10
2	FUNDAMENTAÇÃO.....	12
2.1	REFLECTÂNCIA.....	12
2.2	GONIÔMETRO.....	13
2.3	DETERMINAÇÃO DO FATOR DE REFLECTÂNCIA.....	15
2.4	PARÂMETROS AMBIENTAIS.....	17
2.5	REQUISITOS PARA A ESTAÇÃO DE MONITORAMENTO.....	20
2.6	MICROCONTROLADORES.....	21
2.6.1	ARDUINO.....	23
2.6.2	Modelos de ARDUINO.....	24
2.7	COMUNICAÇÃO SERIAL.....	25
3	METODOLOGIA.....	29
3.1	LaRaC.....	29
3.2	DESENVOLVIMENTO INICIAL.....	30
3.3	ARDUINO MEGA E <i>SHIELD</i>	33
4	RESULTADOS.....	35
5	CONCLUSÃO.....	39
	Referências.....	40
	APÊNDICE A.....	42

1 INTRODUÇÃO

O Laboratório de Radiometria e Caracterização de Sensores Eletro-ópticos – LaRaC, do Instituto de Estudos Avançados – IEAv, conduz experimentos para a determinação de assinaturas espectrais, mais especificamente a reflectância, de superfícies de interesse para sensoriamento remoto. Com a construção e funcionamento de um equipamento, um Goniômetro (CASTRO et al, 2017), as medições podem se estender por até 40 horas. Assim, para ter maior controle de reprodutibilidade durante as medições que são realizadas, tornou-se necessário desenvolver e montar uma Estação de Monitoramento das condições ambientais do laboratório, reduzindo o número de variáveis a serem levadas em consideração na análise dos resultados, aumentando a confiabilidade dos dados obtidos.

O acompanhamento de parâmetros ambientais (temperatura, pressão, umidade, iluminação, etc.) é importante para assegurar a reprodutibilidade dos dados obtidos pelo Goniômetro, de maneira a criar um banco de dados que pode ser consultado para a verificação de eventuais imprecisões e discrepâncias durante as medições de reflectância. A alteração desses parâmetros pode indicar alterações atmosféricas naturais, assim como falhas no controle de temperatura do LaRaC, alteração no funcionamento dos desumidificadores de ar, se a iluminação do laboratório sofreu alguma mudança ou se a porta foi aberta. Caso alguma destas condições tenha se alterado, os dados podem ser descartados ou analisados de acordo com as alterações, evitando conclusões errôneas sobre os experimentos.

Por isso, também é necessário que se tenha informações horárias precisas possibilitando a comparação entre a marcação horária dos dados obtidos pelo sensor de reflectância e da Estação e, então, realizar comparações para determinar possíveis alterações que possam ter sido causadas por parâmetros ambientais.

O monitoramento da temperatura e da pressão atmosférica podem nos proporcionar informações de grande importância para que possamos obter conhecimento de como o sensor de reflectância se comportaria ao ser embarcado em um VANT, avião ou satélite. Tendo conhecimento das condições atmosféricas da região onde serão feitas as medições podemos prever o comportamento do sensor e obter resultados com mais segurança da superfície que foi mapeada.

Assim, para atender a estas necessidades (RUBIATTI, 2016), o objetivo geral do trabalho foi desenvolver uma Estação de Monitoramento de parâmetros ambientais, de baixo custo, para o LaRaC. Mais especificamente, os pré-requisitos da Estação de Monitoramento

eram: a) Realizar medições, a cada minuto, da temperatura, umidade, iluminação, em diversos pontos estratégicos do laboratório; b) Determinar o horário exato das medições; c) Indicar as informações mais relevantes em uma tela de cristal líquido; e d) Gravar os dados obtidos em um cartão SD.

Para atender a essas necessidades, sem que houvesse aumento significativo dos custos para a criação e desenvolvimento da Estação, optou-se pela utilização da plataforma ARDUINO, uma plataforma de prototipagem *open-source*, capaz de ler entradas analógicas e/ou digitais e transformá-las em saídas. Isto é feito através de sua linguagem de programação que é realizada em um *Software* específico, conhecido como IDE, baseado em processamento. Sua programação é flexível, tornando uma ferramenta que pode ser usada por iniciantes e por usuários avançados.

Devido ao número de sensores a serem utilizados, não haveria portas suficientes no ARDUINO para ligar cada um separadamente. Por isso, foi necessário utilizar um protocolo de comunicação que permita a integração/comunicação de vários sensores. Então, o barramento e protocolo de comunicação I²C, da Phillips, foi escolhido por que, convenientemente, é presente nas plataformas ARDUINO e permite a integração de até 112 sensores com o emprego de apenas duas portas: SDA (*Serial Data*) e SCL (*Serial Clock*). Entretanto, é conveniente lembrar, que existem limitações em relação ao endereçamento dos sensores.

Para apresentar as etapas de desenvolvimento do Trabalho, este documento, está dividido em 4 Capítulos, além desta Introdução. No Capítulo 2, de “Fundamentação” são apresentados alguns conceitos para o entendimento da necessidade do desenvolvimento da Estação de Monitoramento. No Capítulo 3, de “Metodologia” é apresentada a maneira pela qual a Estação foi planejada e construída. No Capítulo 4, de “Resultados” é apresentada a análise estatística com os resultados quantitativos. Em seguida são apresentadas conclusões do trabalho.

2 FUNDAMENTAÇÃO

Neste Capítulo são apresentados os fundamentos básicos sobre a determinação da reflectância em laboratório e as necessidades para o acompanhamento dos parâmetros ambientais que podem influenciar as medições. Também são abordados os princípios básicos da solução encontrada para as medições em laboratório.

Alguns dos temas abordados neste trabalho são atualizados muito constantemente por pertencerem às áreas de informática e tecnologia. Por este motivo, muitas das fontes de pesquisa utilizadas foram online, uma vez que os livros não acompanham essas atualizações eficientemente.

2.1 REFLECTÂNCIA

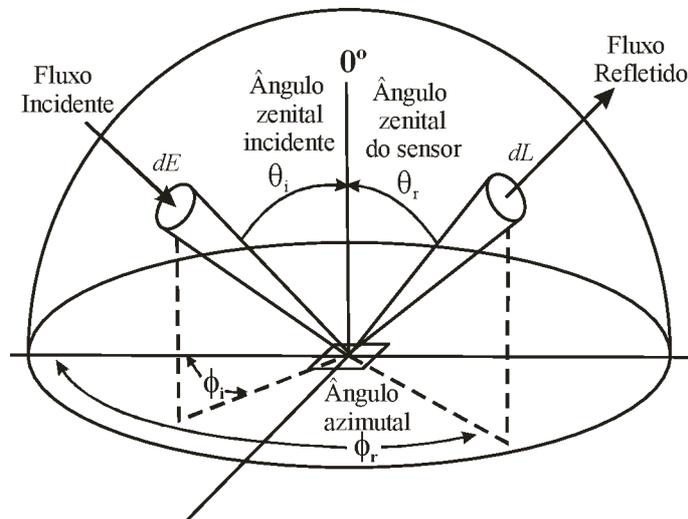
A reflectância, ρ , é a propriedade de uma determinada superfície em refletir a radiação eletromagnética, REM. A reflectância é uma grandeza adimensional, variando entre 0 e 1 para todos os materiais. Além disso, ela também é fornecida na literatura, de forma percentual. A maioria das superfícies apresenta um comportamento anisotrópico (não reflete igualmente a REM em todas as direções) para os mais diversos ângulos de iluminação. Este fenômeno pode ser descrito pela função de distribuição da reflectância bidirecional, BRDF.

Na Figura 1, podem ser observados detalhes da geometria de medição da reflectância. As posições da fonte de iluminação (o Sol, quando em campo) e do sensor são definidas por dois ângulos: ângulo zenital, θ , e ângulo azimutal, φ .

Nesse sentido, muitos estudos têm se desenvolvido com foco na caracterização de superfícies em termos do comportamento anisotrópico da reflectância, o qual pode ser descrito pela função de distribuição da reflectância bidirecional (BRDF, do termo em inglês *Bidirectional Reflectance Distribution Function*). A BRDF (JENSEN, 2009) é definida como a razão da radiância refletida em uma direção (θ_r , φ_r , ou seja, ângulos zenital e azimutal, respectivamente) e a irradiância incidente de uma direção específica (θ_i , φ_i) em uma superfície. Devido às dificuldades técnicas obtém-se, na prática, uma grandeza

equivalente denominada Fator de Reflectância, FR , ou ainda, $FR(\theta_i, \varphi_i; \theta_r, \varphi_r; \lambda)$ (JENSEN, 2009).

Figura 1 - Geometria de aquisição da medição da reflectância, onde um alvo é iluminado com irradiância () a partir de ângulos zenital e azimutal específicos, e o sensor registra a radiância (dL) que sai do alvo em ângulos azimutal e zenital específico.



Fonte: Jensen (2009).

2.2 GONIÔMETRO

Goniômetro é, genericamente, um instrumento para determinar ângulos. Entretanto, no nosso caso, o goniômetro, é um equipamento que permite o posicionamento de uma fonte de iluminação (por exemplo, uma lâmpada) e de um sensor (no caso um espectrorradiômetro), em ângulos zenitais e azimutais bem conhecidos, para determinar o fator de reflectância em laboratório. Assim, para que a reflectância seja determinada com acurácia e precisão, o goniômetro deve possuir características que permitam o ajuste e a determinação dos ângulos zenitais com precisão.

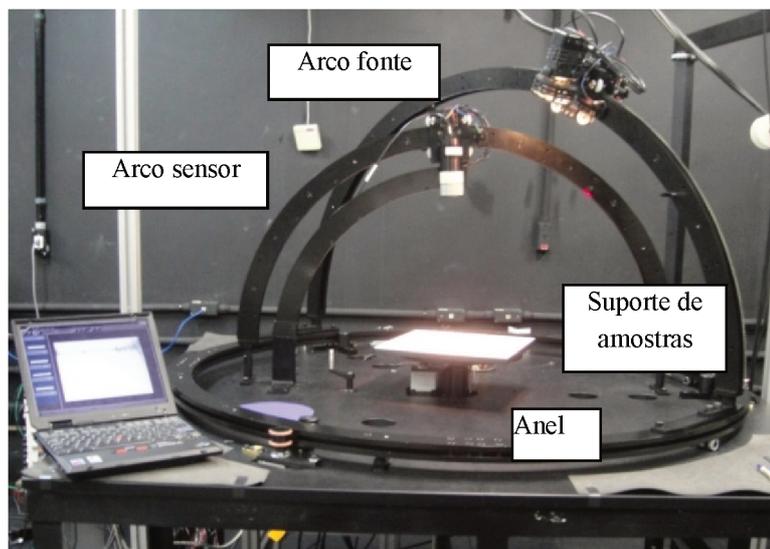
Deste modo, no início de 2015, foi então elaborado o Projeto Básico (CASTRO et al, 2015) para construção do Goniômetro do LaRaC. Algumas destas características eram: a) Um aro metálico, denominado “Anel”, montado na horizontal, com diâmetro interno de

aproximadamente 1,2 m; b) Um arco semicircular, com abertura de 180° , denominado “Arco Fonte”, montado sobre o Anel, com a finalidade de suportar e guiar o movimento de uma fonte luminosa. Os diâmetros, interno e externo, são aproximadamente de 1,2 e 1,3 m, respectivamente; c) Um arco, denominado “Arco sensor”, posicionado na vertical, fixado sobre a mesa metálica, a fim de suportar o movimento zenital de um sensor óptico. O arco, semicircular, com abertura de 180° e diâmetros, interno e externo, com dimensões de aproximadamente 1,0 e 1,1 m, respectivamente; d) Um suporte circular plano, denominado “Suporte das amostras”, posicionado no centro da base, onde estariam dispostas as amostras experimentais. O suporte deveria possuir diâmetro de 300 mm e rotacionar perfazendo um curso de 360° . Além disso, deveria possuir ajuste de altura de aproximadamente 0,1 m, para o correto posicionamento das amostras no plano formado pelo ângulo zero dos arcos.

Além disso, para assegurar o correto posicionamento do goniômetro, o Anel, o Arco Fonte e o Arco Sensor deveriam possuir demarcações separadas a 15° , que pudessem ser utilizadas como referência durante o movimento automático, contando com o auxílio de lasers e sensores de luz usados para detectar tais demarcações, ou até mesmo para a movimentação manual do equipamento. Desta maneira, o número de medições necessárias para compor um ciclo completo envolvendo todas as geometrias na escala de 15° é de 20736, compreendendo: 24 posições do ângulo azimutal da Fonte; 24 posições do ângulo azimutal do Sensor; 6 posições do ângulo zenital da Fonte; 6 posições do ângulo zenital do sensor. Com isto, são necessárias aproximadamente 60 h de medição para realizar a determinação completa da estimativa da BDRF de uma amostra. Mesmo eliminando milhares de medições (por efeito de sombras, etc.), obteríamos um tempo de aquisição de 50 h, que continua sendo um tempo elevado para realização das medições, visto que os parâmetros ambientais (temperatura, umidade, concentração de gases, etc.), a tensão da rede elétrica, etc., podem variar significativamente, mesmo em ambientes controlados, ao longo do tempo.

A partir das especificações básicas, o Goniômetro foi construído, testado e instalado no LaRaC, com o apoio da Agência Espacial Brasileira – AEB, por meio do financiamento do Projeto “Assinaturas espectrais”, Processo: 560228/2010-4 do Edital MCT/CNPq/AEB nº 33/2010. Na Figura 2 é apresentado o goniômetro do LaRaC (CASTRO et al, 2017).

Figura 2 - Goniômetro do LaRaC e suas principais estruturas constituintes.



Fonte: Castro et al (2017).

A partir de outubro de 2015 o Goniômetro estava apto a realizar medições da reflectância de superfícies de interesse. Com isto, a primeira tarefa foi verificar as eventuais geometrias, ou seja, as configurações de ângulos azimutais e zenitais, que acarretavam sombras da estrutura do Goniômetro sobre a amostra. A partir desta informação, os dados obtidos nestas geometrias não devem ser levados em consideração.

2.3 DETERMINAÇÃO DO FATOR DE REFLECTÂNCIA

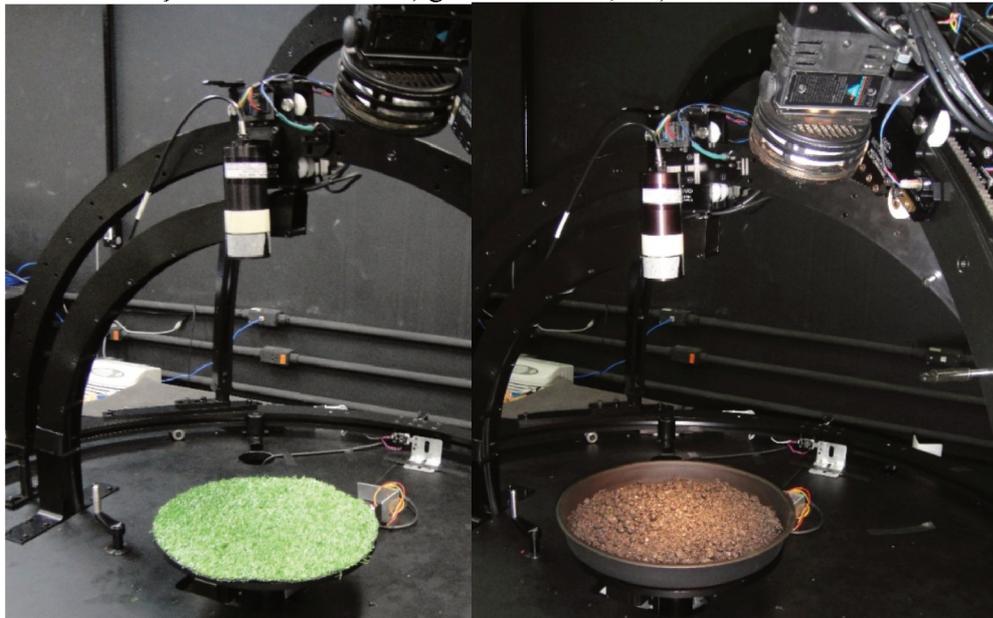
Com a construção do Goniômetro, foi possível realizar a determinação do Fator de reflectância, FR , em laboratório. O arranjo experimental, como podemos observar na Figura 2, envolve ainda: a) uma fonte de luz, no caso uma lâmpada incandescente de 50 W, acoplada a um iluminador, e ligada a uma fonte de tensão que deve manter a tensão e corrente constantes na lâmpada; e b) um espectrorradiômetro, que é alimentado por outra fonte de tensão (que deve manter a tensão e corrente constantes).

O espectrorradiômetro utilizado, um FildSpec da ASD (recentemente adquirida pela Malvern Panalytical). A captação da radiação eletromagnética é realizada por sistema composto de uma lente (para a diminuição do ângulo de visada) e um feixe de fibras óticas de 1,5 m, que “leva” a radiação até uma unidade eletro-óptica que realiza as medições. As medições são controladas por meio de um microcomputador. Com isto, no arranjo do goniômetro, a unidade eletro-óptica do espectrorradiômetro é posicionada abaixo da mesa, evitando que ela seja iluminada e que sua temperatura aumente com o passar do tempo.

Deste modo, neste arranjo (ver Figura 2), quando a lâmpada é ligada, o filamento é aquecido e produz radiação eletromagnética (Radiação de Corpo Negro), que atravessa a “atmosfera” (aproximadamente 0,5 m de ar) e reflete na superfície de a ser estudada. Parte da radiação refletida atravessa novamente a “atmosfera” (aproximadamente 0,5 m de ar) e chega a cabeça de medição do espectrorradiômetro. Realizando o mesmo procedimento com uma superfície de referência, determina-se o fator de reflectância de uma determinada superfície realizando a comparação entre as medições.

Na Figura 3 podem ser observadas as medições de amostras de grama sintética e de asfalto.

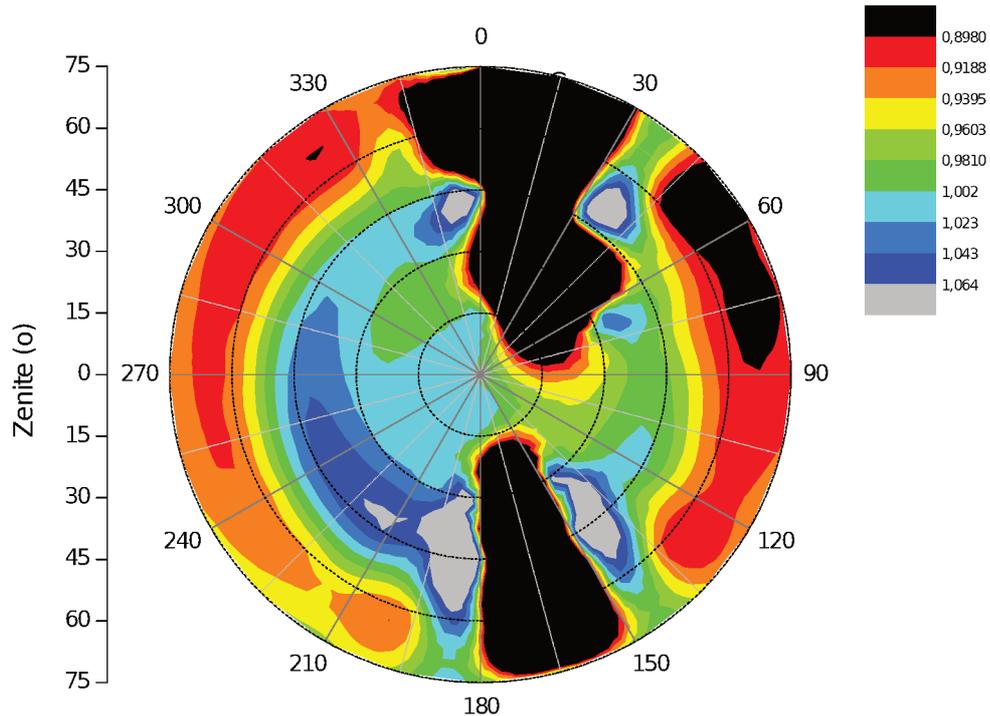
Figura 3 - Foto da medição de amostras de: a) grama sintética; e b) asfalto.



Fonte: Castro et al (2017)

Um exemplo do resultado destas medições, da reflectância de uma placa de Spectralon fabricada pela Labsphere, para variação dos ângulos, zenital e azimutal, da fonte, mantendo-se fixos os ângulos, zenital (em 30°) e azimutal (em 0°), do sensor para um comprimento de onda de 900 nm pode observado na Figura 4.

Figura 4 - Estimativa do Fator de Reflectância de uma placa de Spectralon.



Fonte: Castro et al (2017).

Na Figura 4, as regiões em preto compreendem as regiões de sombra geradas pela estrutura e componentes do goniômetro, embora essas delimitações não sejam exatas, tendo em vista possíveis reflexões, ainda que mínimas, por parte do entorno e da própria estrutura e considerando que a fonte luminosa não emite um feixe luminoso cilíndrico, condições estas que permitem melhor definição das regiões de sombra. Outra importante observação diz respeito à variação do fator de reflectância da placa de Spectralon. Em ângulos zenitais de iluminação maiores, o FR da placa tende a diminuir.

Vale lembrar, como mencionado anteriormente, que uma medição (para todas as geometrias disponíveis no goniômetro) pode consumir um tempo superior a 50 h. Como não é conveniente realizar esta medição de uma única vez, a medição é dividida em vários dias, perfazendo até duas semanas.

Como podemos perceber, a partir do arranjo e dos procedimentos experimentais, que existem diversos fatores que podem interferir no resultado da medição. Deste modo, para que tenhamos a precisão e a acurácia desejadas é importante verificar se estes parâmetros estão variando ao longo das medições.

2.4 PARÂMETROS AMBIENTAIS

De maneira simplificada, parâmetros ambientais podem ser considerados como uma série de fatores que podem interferir nas medições em um determinado ambiente.

Entre os diversos fatores que podem influenciar as medições espectrorradiométricas, conforme descritas na Seção 2.3, podemos destacar: temperatura; umidade relativa do ar; concentração de CO₂; intensidade luminosa; tensão e corrente; etc. Assim, levando em consideração o ambiente do LaRaC, podemos estabelecer critérios para o acompanhamento dos parâmetros ambientais.

Temperatura

A temperatura pode influenciar as medições de várias formas, sendo, no nosso caso, as mais importantes:

- a) Alterando as propriedades físicas, como por exemplo: a reflectância de uma determinada superfície ou a umidade do ar, de materiais e superfícies; e
- b) Alterando as características de funcionamento dos diversos equipamentos que compõe o arranjo experimental. Em especial podemos citar o espectrorradiômetro, cujos ganhos dos amplificadores dependem significativamente da temperatura.

Para mitigar os efeitos da temperatura nos equipamentos e nas propriedades físicas dos materiais, procura-se ligar os equipamentos duas horas antes do início medições, para que todo o arranjo esteja em equilíbrio térmico quando forem realizadas as medições.

Deste modo, se a variação da temperatura no arranjo experimental for pequena (alguns décimos de °C, ou até mesmo 1°C, como as observadas no LaRaC) não se espera uma alteração significativa nas medições. Isto indica que as medições de temperatura devem ser realizadas com uma resolução compatível ou melhor que décimos de °C.

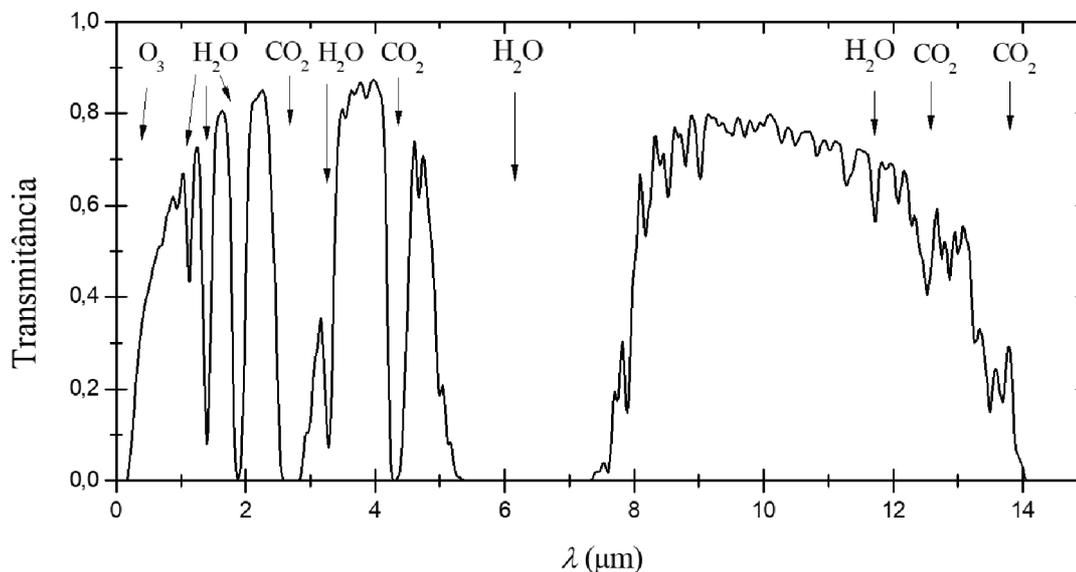
Idealmente as medições de temperatura devem ocorrer em vários pontos do arranjo experimental, para se ter uma ideia se a variação de temperatura ocorre em algum ponto específico.

Umidade relativa do ar

Em determinados comprimentos de onda a radiação eletromagnética é absorvida, em maior ou menor grau, pelas moléculas de água no ar, ver Figura 5. Assim, mesmo que a radiação eletromagnética percorra um caminho de apenas 1 m (da fonte a superfície e depois

até ao espectrorradiômetro), alguns comprimentos de onda podem ser significativamente afetados.

Figura 5 - Transmitância atmosférica para uma altitude em relação ao solo de 1 km, e moléculas absorvedoras.



Fonte: Andrade (2012)

Para mitigar os efeitos da umidade relativa do ar no laboratório são utilizados desumidificadores que mantêm a umidade a níveis baixos (em torno de 50 %) e constantes. Com isto, se a variação da umidade relativa do ar for pequena (alguns %, como as observadas no LaRaC) não se espera uma alteração significativa nas medições. Isto indica que as medições da umidade relativa do ar devem ser realizadas com uma resolução compatível ou melhor que 1 %. As medições da umidade relativa do ar devem ocorrer próximas das medições espectrorradiométricas, ou seja, na parte superior da mesa, próximo ao caminho óptico das radiações eletromagnéticas.

Concentração de CO₂

De maneira semelhante ao que ocorre com as moléculas de água, as moléculas absorvem a radiação eletromagnética em determinados comprimentos de onda, ver Figura 5.

Em geral, em ambientes abertos, a concentração de CO₂ tem aumentado significativamente ao longo dos anos devido a combustão de combustíveis fósseis (petróleo). Atualmente o valor para a concentração de CO₂ no ambiente é de aproximadamente 400 ppm (NOAA, 2018). Em laboratório, ou seja, em um ambiente fechado (que não existe grandes trocas de ar), a concentração de CO₂ se altera, principalmente, devido ao número de pessoas presentes no laboratório, pois elas exalam CO₂ no ambiente. Para mitigar este efeito,

as medições são realizadas de maneira automática sem a necessidade de supervisão, ou seja, sem a presença de pessoas no laboratório.

Como, não existe a possibilidade de controlar a concentração de CO₂ no LaRaC, torna-se importante medir a concentração de CO₂ durante as medições para saber se houve alguma variação durante as medições. Da mesma maneira que as medições da umidade relativa do ar as medições da concentração de CO₂ devem ocorrer próximas das medições espectrorradiométricas, ou seja, na parte superior da mesa, próximo ao caminho óptico das radiações eletromagnéticas.

Intensidade Luminosa

Como as medições de reflectância envolvem a radiação eletromagnética do visível ao infravermelho próximo (de 350 a 2500 nm), qualquer interferência de luzes pode influenciar os resultados obtidos. Para mitigar estes efeitos, o LaRaC bem como a maioria dos equipamentos é pintado com tintas preto fosco, ver Figuras 02 e 03, que diminui significativamente as reflexões internas de luzes do laboratório. Além disso, as medições são realizadas com todas as lâmpadas de iluminação apagadas.

Assim, as eventuais interferências nas medições podem ocorrer devido, por exemplo, a abertura de uma porta ou uma lâmpada esquecida acesa no laboratório. Deste modo, para verificar a existência destas ocorrências é importante medir a intensidade luminosa em diversos pontos do laboratório.

Tensão e corrente

As variações de tensão e corrente podem afetar, principalmente, de duas maneiras as medições:

- a) A alteração da tensão rede elétrica pode afetar o funcionamento dos equipamentos, tais como o espectrorradiômetro e as fontes de tensão; e
- b) A variação de tensão e corrente na lâmpada utilizada no iluminador, pode alterar a temperatura do filamento da lâmpada e conseqüentemente a intensidade da luminosidade produzida por ela.

Deste modo, o acompanhamento das eventuais variações de tensão deve ser realizado caso a caso, ou seja, na rede elétrica e de equipamento para equipamento. Com isto, os sensores de tensão e corrente, devem ser escolhidos com base nos limites de funcionamento, bem como na sua precisão.

2.5 REQUISITOS PARA A ESTAÇÃO DE MONITORAMENTO

Como mencionado na Seção 2.4, vários parâmetros ambientais podem afetar as medições espectrorradiométricas. Deste modo, em longas medições, torna-se necessário acompanhar estes parâmetros para verificar alguma, eventual, influência nas medições espectrorradiométricas.

Existem no mercado algumas soluções para o acompanhamento de até um conjunto destes parâmetros. Entretanto, não existe em uma única solução (equipamento) para o acompanhamento de todos os parâmetros simultaneamente. Por isso, optou-se por desenvolver uma Estação de Monitoramento de parâmetros ambientais.

Deste modo, um dos requisitos desta Estação de Monitoramento era realizar medições, a cada minuto, da temperatura, pressão, umidade, iluminação, tensão e corrente, em diversos pontos estratégicos do laboratório, com precisão e acurácia. Outro requisito era gravar os dados obtidos, preferencialmente, em um cartão SD. Para facilitar visualização dos resultados de alguns destes parâmetros ambientais, a Estação também deveria indicar as informações mais relevantes em uma tela de cristal líquido.

Como as medições destes parâmetros devem ser simultâneas as medições espectrorradiométricas, a Estação de Monitoramento deve fornecer/determinar o horário exato das medições, para posterior comparação com as medições do fator de reflectância. Ademais, para facilitar, a Estação deveria disponibilizar, *on line*, na rede do IEAv os resultados das medições.

Além destes requisitos a Estação deveria ser de baixo custo de fácil utilização. Deste modo, para fosse criada a Estação de Monitoramento, com estes requisitos, optou-se pela utilização de microcontroladores e módulos/sensores que vem, recentemente, influenciando o mercado de microeletrônica.

2.6 MICROCONTROLADORES

Os microcontroladores integram um microprocessador em conjunto com diversos outros componentes como conversores analógico-digitais, memórias, temporizadores, interface de comunicação serial, etc. Desta forma é possível reduzir custos e tamanho físico, além de conferir maior confiabilidade por serem fabricados já integrados, não havendo necessidade de cálculos e testes minuciosos para verificar o funcionamento dos componentes em conjunto.

Ao reduzirem o custo e o tamanho físico em relação a um projeto que usa um dispositivo microprocessado, tornam-se ferramentas versáteis para controle de dispositivos e processos de forma digital. Há, também, microcontroladores de sinal misto, que são capazes de integrar dispositivos analógicos e digitais no mesmo sistema.

Com consumo de energia relativamente baixo, normalmente na casa dos miliwatts, nos dá a possibilidade de entrar em modo de espera (*Sleep*) aguardando que um evento aconteça, como, por exemplo, o acionamento de uma tecla ou o recebimento de um sinal em sua interface. No modo de espera o consumo de energia pode chegar à valores como nanowatts, os tornando ideais em aplicações nas quais o baixo consumo de energia é decisivo para o sucesso.

O primeiro microcontrolador foi desenvolvido pela Texas Instruments em 1971 (MICROCONTROLLER, 2018) e sua comercialização iniciou-se em 1974. Ele combinou uma ROM, uma memória de leitura/gravação, um processador e um relógio em um único chip que tinha como alvos sistemas embarcados.

Os microcontroladores foram responsáveis pela evolução de equipamentos que estavam estagnados há anos como o motor a combustão, que agora pode alternar diferentes tipos de combustível.

Segundo Aureliano (2017) o microcontrolador possui algumas partes principais:

CPU: responsável por executar operações lógicas e matemáticas programadas.

Frequência de *clock*: base de tempo para as operações desempenhadas pela CPU e demais periféricos presentes no microcontrolador.

Memória *flash*: memória não volátil, ou seja, não tem seu conteúdo apagado com a falta de alimentação. É responsável por armazenar o programa a ser executado.

Memória RAM: memória volátil responsável por armazenar temporariamente variáveis do programa, como variáveis de entrada, ou variáveis de cálculo.

Memória EEPROM: memória não volátil, assim como a memória flash, tendo em comparação a esta uma menor velocidade de escrita.

Entradas: pinos configurados como entradas digitais detectam a tensão presente externamente neles, dados alguns valores limiares. Quando abaixo de determinado valor de tensão, dado presente na folha de dados do microcontrolador de interesse, o nível lógico lido é 0, já se dentro de uma faixa próxima do valor de alimentação,

também dependente do microcontrolador em questão, o nível lógico lido é 1. Conectando chaves ou interruptores a pinos de entrada é possível alterar estados de variáveis dentro do software em execução no microcontrolador, fazendo com que o programa execute alguma tarefa de interesse.

Saídas: pinos configurados como saídas digitais podem ser comandados por meio do software e ter seu valor de tensão assim alterado. Nível lógico 0 representa valor de tensão nulo, já nível lógico 1 representa valor de tensão de alimentação do circuito. A saída digital dos microcontroladores é a principal interface deste componente com o mundo real, bastando conectar um LED e fazê-lo piscar, ou um pequeno alto-falante para emitir um sinal sonoro.

Atualmente encontramos os mais diversos microcontroladores, que dependem da sua finalidade, custos, etc. Entre os principais microcontroladores utilizados no mercado podemos citar o ARDUINO.

2.6.1 ARDUINO

Segundo o texto “What is Arduino?” (ARDUINO, 2018) em uma tradução livre, o ARDUINO é uma plataforma eletrônica *open-source* baseada em um *hardware* e *software* de fácil utilização. São capazes de ler entradas – luzes em sensores, botões sendo pressionados e até mesmo mensagens no Twitter – e transformá-las em saídas – Ativando um motor, ligando um LED, publicando algo online. Você pode programá-lo enviando instruções ao seu microcontrolador. Para tal, é necessário que se use a linguagem de programação ARDUINO, baseada em *wiring*, e seu *software* (IDE), baseado em *Processing*.

A placa ARDUINO consiste em um microcontrolador Atmel AVR de 8-bits, com componentes complementares para facilitar a programação e incorporação para outros circuitos. Um importante aspecto é a maneira padrão que os conectores são expostos, permitindo o CPU ser interligado a outros módulos expansivos, conhecidos como *shields*.

A principal finalidade do ARDUINO em um sistema é facilitar a prototipagem, implementação ou emulação do controle de projetos interativos, a nível doméstico, comercial ou móvel, da mesma forma que o Controlador Lógico Programável – CLP controla sistemas de funcionamento industriais. Com ele é possível enviar ou receber informações de basicamente qualquer sensor ou sistema eletrônico.

Ao longo dos anos o ARDUINO tem sido utilizado em milhares de projetos, desde objetos do dia a dia até instrumentos científicos complexos. Uma comunidade mundial de estudantes, *hobbyistas*, artistas, programadores e profissionais desenvolveram e ainda desenvolvem conhecimento acessível a respeito dessa plataforma que pode ser utilizado por iniciantes e profissionais.

O ARDUINO nasceu na *Ivrea Interaction Design Institute* (ARDUINO, 2018) como uma ferramenta fácil para prototipagem tendo como alvos estudantes sem base em eletrônica e programação. Quando atingiu um maior número de pessoas a plataforma começou a mudar e se adaptar a novas necessidades e desafios. Diferenciando-se de uma simples placa 8-bit para produtos para aplicações IoT (*Internet of Things*), impressão 3D e ambientes incorporados. Todas as placas ARDUINO são completamente *open-source*, permitindo que usuários as utilizem de maneira independente e, eventualmente, as adaptem para suas necessidades particulares. O *software* também é *open-source* e cresce cada dia mais através da contribuição de usuários de todo o mundo.

A plataforma cresceu devido a alguns fatores como: seu baixo custo, que é inferior a US\$15,00, funciona em diferentes sistemas operacionais (Windows, Linux e Mac) enquanto a maioria dos microcontroladores está limitada ao Windows; possui uma programação simples e limpa tornando-a fácil de usar para iniciantes; e possui *hardware* e *software open-source* que permite que seja adaptada a situações específicas.

2.6.2 Modelos de ARDUINO

Atualmente existem vários modelos disponíveis com diferentes configurações para se adaptarem da melhor forma possível a sua necessidade e, também, podem ser encontrados *shields* que ampliam as possibilidades do ARDUINO, adicionando novas funções como conexão a internet, drivers para diversos tipos de motores, drivers para telas de cristal líquido, entre outras funções.

Em uma tradução livre de ARDUINO (2018) podemos verificar as principais diferenças entre os modelos de ARDUINO.

UNO: é um microcontrolador baseado no ATmega328P. Tem 14 pinos entrada/saída digitais, dos quais 6 podem ser usados como saídas PWM, 6 entradas analógicas, um cristal de quartzo de 16MHz, um conector USB, uma entrada de energia, um ICSP e um botão de reset. A sua memória *flash* é de 32 kB;

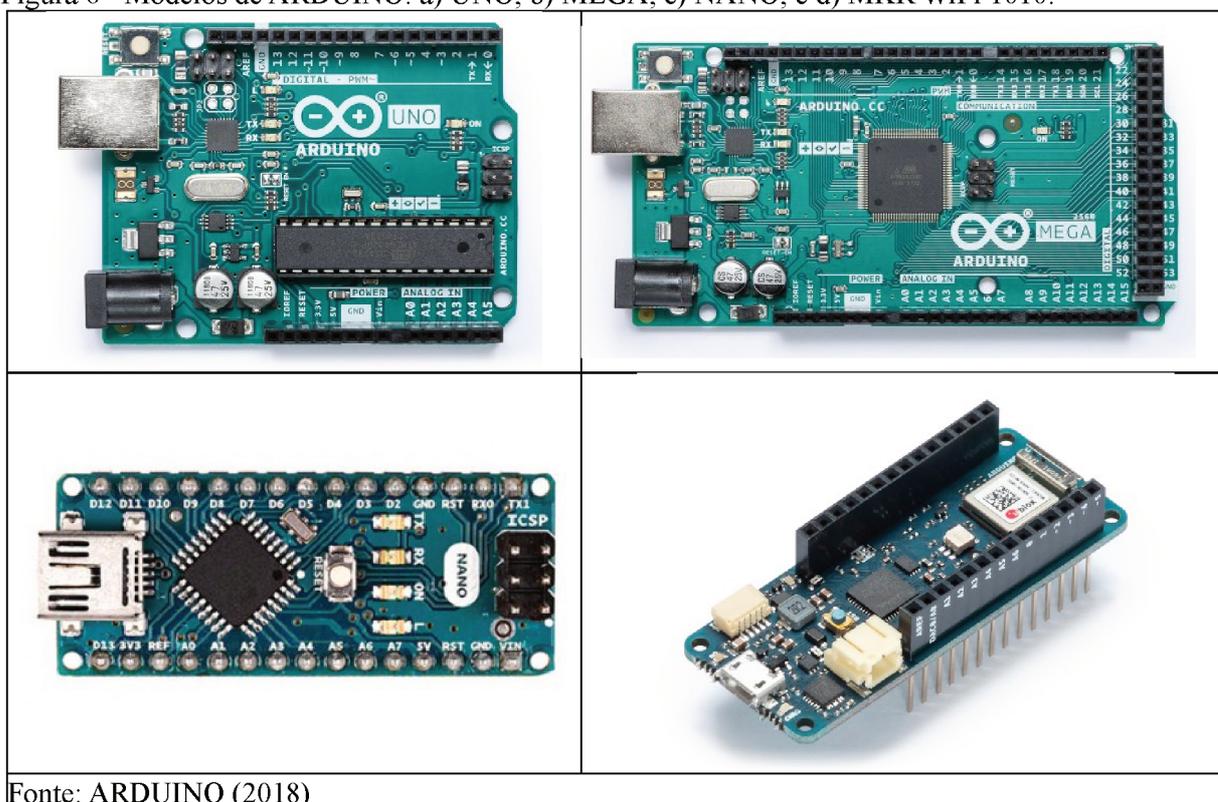
MEGA: É um microcontrolador baseado no ATmega2560. Tem 54 pinos de entrada/saída digitais, dos quais 15 podem ser usados como saídas PWM, 16 entradas analógicas, 4 UART's, um cristal oscilador de 16MHz, uma conexão USB, uma entrada de energia, um ICSP e um botão de reset. É compatível com a maioria dos *shields* para ARDUINO. A sua memória *flash* é de 256 kB;

NANO: É uma placa pequena, completa e baseada no ATmega328P. Ela tem mais ou menos a mesma funcionalidade de um ARDUINO Duemilanove, mas em um formato diferente. Nela só falta uma entrada DC e funciona com um conector Mini-B USB. A sua memória *flash* é de 32 kB; e

MKR WiFi 1010: É uma placa ARDUINO baseada no SAMD21 Cortex-M0+ 32bit Low Power ARM MCU, possuindo 14 pinos entrada/saída, e inclui a comunicação wifi. A sua memória *flash* é de 256 kB.

Na Figura 6 é possível observar estes modelos de ARDUINO.

Figura 6 - Modelos de ARDUINO: a) UNO; b) MEGA; c) NANO; e d) MKR WiFi 1010.



Fonte: ARDUINO (2018)

2.7 COMUNICAÇÃO SERIAL

Em telecomunicações e informática a comunicação serial é o processo de enviar dados, um bit de cada vez, de forma sequencial, num canal de comunicação. É diferente da

comunicação paralela, em que todos os bits de um byte são enviados juntos (SERIAL COMMUNICATION, 2018). Na parte física, utilizando um meio de comunicação metálico, a comunicação serial somente necessita de um par de fios para que a comunicação aconteça, uma vez que o dado é transmitido bit a bit. Nos primeiros sistemas de comunicação digital, a comunicação serial era vantajosa para transmitir dados a longas distâncias, pois, com apenas um par de fio a comunicação poderia ser realizada.

Atualmente, com a melhoria da qualidade e da velocidade dos meios de transmissão, a comunicação serial é mais vantajosa e amplamente utilizada para a troca de informações entre dispositivos, como um computador e uma impressora.

Entretanto, para que haja a comunicação entre os componentes de um determinado sistema é necessário que exista um barramento. O barramento pode ser entendido como um condutor ou circuito de baixa impedância ao qual podem ser conectados outros condutores ou circuitos.

Atualmente existem diversos barramentos seriais, tais como: Serial (RS232), USB, I²C, SPI, UART, etc. O Serial (ou RS232) é um padrão de protocolo para troca série de dados binários entre um DTE (terminal de dados, de *Data Terminal equipment*) e um DCE (comunicador de dados, de *Data Communication equipment*). O USB (abreviatura de Universal Serial Bus) é um padrão de indústria que estabelece especificações para cabos, conectores, e protocolos de comunicação para conexão, comunicação e provimento de energia entre computadores pessoais e seus dispositivos periféricos. O SPI (*Serial Peripheral Interface*) é um protocolo que permite a comunicação do microcontrolador com diversos outros componentes, formando uma rede. É uma especificação de interface de comunicação série síncrona usada para comunicação de curta distância, principalmente em sistemas embarcados. A interface foi desenvolvida pela Motorola e tornou-se um padrão de fato. Os UARTs (*Universal Asynchronous Receiver/Transmitter*) são utilizados para comunicar a maiores distâncias e a sua sincronização é feita por software.

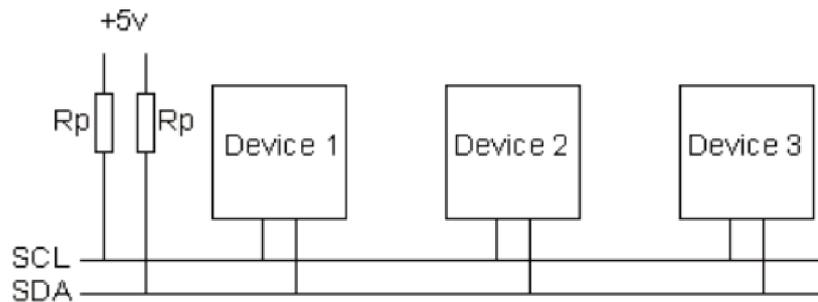
Para este trabalho optou-se pelo padrão I²C, que possui: duas linhas de dados (SCL e SDA); uma simples relação de *master* e *slave* entre os diversos componentes; diversas taxas de comunicação; e os componentes são endereçáveis por meio de um único endereço. O seu ponto fraco, está relacionado a uma baixa velocidade de transmissão de dados quando comparado com outros padrões. No entanto, como, neste projeto, as medições são realizadas em uma taxa (10 Hz, no pior caso) muito menor que os limites do barramento.

O protocolo I²C é um barramento de comunicação serial que utiliza apenas dois condutores, desenvolvido pela Philips no início da década de 90, este protocolo é muito utilizado para conectar periféricos de baixa velocidade a placas-mãe, microcontroladores e afins.

Tanto a unidade de controle quanto os periféricos devem ser compatíveis com o protocolo I²C, seja via hardware no próprio SoC (System-on-a-chip) ou utilizando circuitos integrados externos como o SC16IS750, ou até mesmo via software, através de um método chamado *bit-bang*, onde o funcionamento do protocolo é emulado bit a bit (CÂMARA, 2018).

O barramento I²C é composto de dois fios, SDA e SCL, e alimentação (VDD), tipicamente de 3.3 V ou 5 V. Os fios de comunicação possuem resistores *pull-ups*, como pode ser visto na Figura 7.

Figura 7 - Diagrama do barramento I²C com os resistores *pull-ups*.

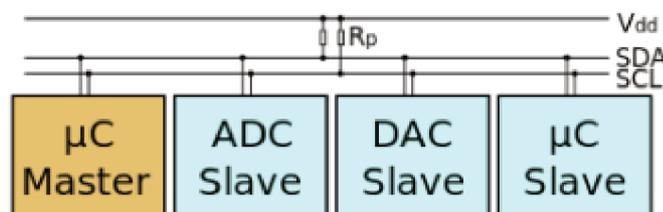


Fonte: (CÂMARA, 2018).

O número de dispositivos em um único barramento é limitado tanto pelo tamanho do endereço, que pode ser de 7 bits, 10 bits e até 16 bits; como por restrição de espaço, já que não se pode ultrapassar poucos metros de fios, pois a capacitância total máxima, algo em torno de 400 pf, impede o funcionamento correto do barramento.

O protocolo I²C tem dois tipos de dispositivos: *Master* e *Slave*. No qual o *Master* (mestre em inglês), é a unidade de controle responsável por coordenar todos os periféricos *Slaves* (escravos em inglês). A linha SCL é responsável pelo *clock* do barramento, e a linha SDA pela transmissão de dados. Um diagrama pode ser observado abaixo na Figura 8.

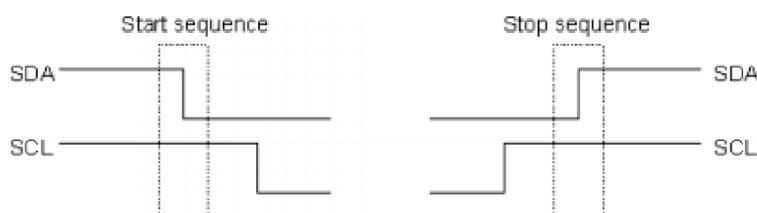
Figura 8 - Diagrama de comunicação entre *master* e *slaves*



Fonte: (CÂMARA, 2018).

Como se pode perceber na Figura 9, no estado neutro do barramento I²C é mantido o valor digital alto em ambas as linhas de comunicação. Para se iniciar a comunicação, SDA é trazido para o valor digital baixo pelo mestre. Para escrever dados no barramento, SCL pulsa, e a cada pulso, o valor em SDA é lido como um bit, começando do MSB.

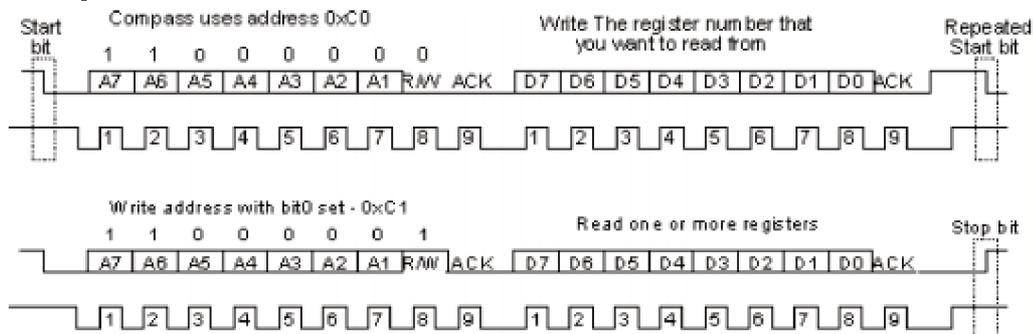
Figura 9 - Diagrama dos estados digitais.



Fonte: (CÂMARA, 2018).

Logo após SDA ser trazida para o valor digital mais baixo, o mestre escreve o endereço do dispositivo como qual ele deseja se comunicar, por exemplo, 0xC0, caso o dispositivo exista, ele responderá como um ACK (o caractere de controle utilizado no controle de transmissão de protocolo para identificar o recebimento de um pacote), e envia um pulso na linha SCL. Então se dá início a transferência de dados, o mestre escreve o endereço do registrador no escravo que ele deseja ler ou escrever (R/W) e opera então, em sequência, podendo ler/escrever um ou mais registrador. Na Figura 10 é exemplificado o que foi descrito.

Figura 10 - Diagrama de transferência de dados entre master e slave.



Fonte: (CÂMARA, 2018).

O barramento I²C juntamente do seu protocolo mais atual, versão 4.0, atualizado em 2012, pode chegar até 5 MHz, mas outras velocidades mais convenientes podem ser

escolhidas para SCL. Em Computadores, as placas-mãe geralmente utilizam velocidades mais baixas como 100 kHz e 10 kHz, já em microcontroladores e sistemas embarcados, periféricos como memórias EEPROM e visores de LCD pedem pelos extremos possíveis do desempenho desse sistema.

Esse protocolo de comunicação serial é um dos mais práticos, simples e de baixo custo disponíveis no mercado, empresas competidoras da Philips vêm desenvolvendo outros barramentos TWI (*two-wired* interface, interface de dois fios, em inglês), para competir com o I2C, que continua sendo o mais utilizado do ramo. Para projetos microcontrolados de pequeno porte e baixo custo, I2C se torna muito conveniente, visto que possibilita uma modularidade de dispositivos, o que não só reflete em menor custo, pois sistemas uma vez complexos podem ser construídos através de sistemas mais simples, mas também ajuda na manutenibilidade, uma vez que cada dispositivo pode ser protegido separadamente (ARDUINOBR, 2017).

3 METODOLOGIA

Como, no nosso caso, estamos trabalhando com medições de reflectância em um laboratório de radiometria, o LaRaC, é importante destacar, que várias das necessidades e, conseqüentemente, as soluções estudadas e encontradas estão relacionadas as características do laboratório.

Por isso se faz necessária uma breve descrição do laboratório e suas necessidades, para depois fazer uma discussão das soluções estudadas para o monitoramento dos parâmetros ambientais.

3.1 LaRaC

O LaRaC possui aproximadamente 50 m² e foi construído, especialmente, para a realização de medições espectrorradiométricas. Atualmente o laboratório conta com 5 “ambientes” ou arranjos experimentais distintos, que permitem medições de: a) Assinaturas espectrais (ou seja, do fator de reflectância, FR) de superfícies e materiais de interesse; b) Caracterização espectral; c) Caracterização radiométrica; d) Caracterização espacial; e e) Caracterização de dispositivos semicondutores.

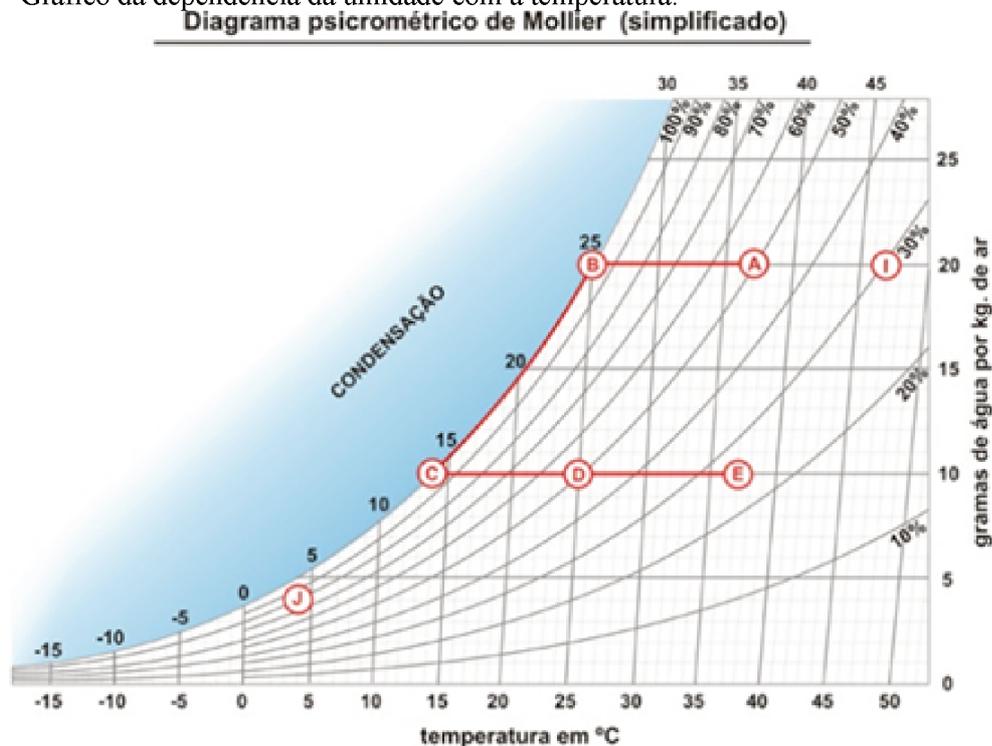
Ele conta com sistema de ar-condicionado e desumidificadores, que mantem a temperatura e a umidade do ar constantes. O uso do ar-condicionado e dos desumidificadores é essencial para se manter a reprodutibilidade dos experimentos, ensaios e testes realizados no laboratório. Na Figura 11, pode-se observar a importância do ar-condicionado em função da umidade relativa do ar.

O LaRaC também conta com um sistema de *nobreak*, que mantém a tensão constante e ininterrupta. Ele não possui janelas e o único acesso é por uma porta.

Em geral as medições são realizadas de modo automático, sem a necessidade de acompanhamento de uma pessoa. Em especial, nas medições com o Goniômetro, o laboratório fica vazio e com as luzes apagadas (a única luz acesa é a do iluminador) para não interferir nas medições.

Portanto, para verificar as eventuais variações nos parâmetros ambientais e tentar correlacionar com as medições de reflectância, foi necessário desenvolver a Estação de Monitoramento.

Figura 11 - Gráfico da dependência da umidade com a temperatura.



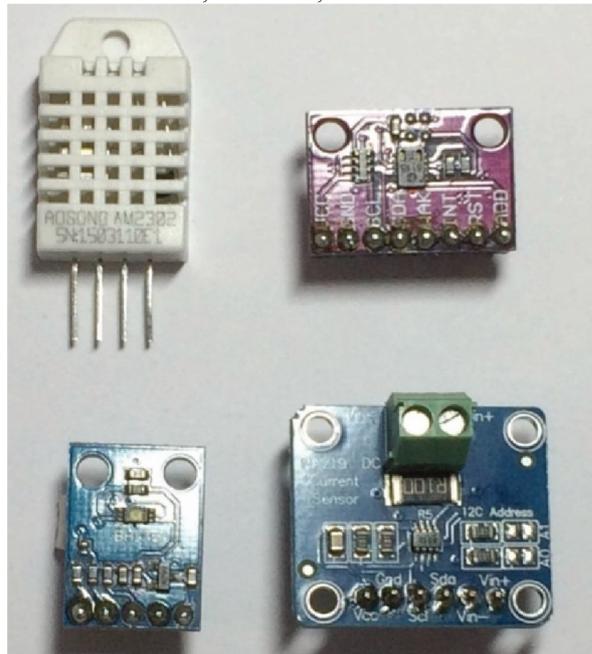
Fonte: (CYCLOAR, 2018)

3.2 DESENVOLVIMENTO INICIAL

Como mencionado anteriormente, a Estação seria desenvolvida com a microcontrolador ARDUINO e sensores com barramento/protocolo I²C, que fossem de baixo custo, mas que tivessem precisão adequada.

Deste modo, para o desenvolvimento inicial da Estação de Monitoramento foi utilizado um ARDUINO UNO. Os sensores utilizados foram selecionados a partir da disponibilidade no laboratório e por meio do estudo de seus *datasheets*, nos quais foram verificados seu funcionamento e precisão. Em seguida, os que não faziam parte do inventário do laboratório foram adquiridos baseados no seu custo e facilidade de aquisição. Com isto, os sensores selecionados foram: o BMP180 (pressão e temperatura) (BMP180, 2015); DHT22 (umidade e temperatura) (AOSONG, 2018); BH1750 (intensidade luminosa) (ROHM, 2010); CSS811 (Concentração de CO₂) (AMS, 2018); INA219 (Tensão e Corrente) (TEXAS INSTRUMENTS, 2015). Uma fotografia dos sensores: DHT22, BH1750, CSS811 e INA219, em seus respectivos módulos, pode ser observada na Figura 12.

Figura 12 - Fotografia dos sensores DHT22, BH1750, CCS811 e INA219.



Fonte: O Autor.

As principais características de cada um destes sensores são:

BMP180: Este é um sensor que realiza medições de pressão e fornece dados de 300 a 1100 hPa com uma resolução de até 0,01 Pa. Entretanto, para que as medições sejam realizadas com precisão é necessário determinar a temperatura ambiente. Por isso, ele também conta com um sensor de temperatura, que fornece dados de 0 até 65 °C com uma resolução de 0,1 °C;

DHT22: Muito embora o sensor DHT22 não possua um barramento/protocolo I²C, ele é um dos mais utilizados para a medição de temperatura e umidade, devido a sua facilidade de uso e custo. As medições de temperatura fornecem dados de -40 até 80 °C com uma resolução de 0,1 °C. As medições de umidade fornecem dados de 0 até 100 % com uma resolução de 0,1 %;

BH1750: este é um sensor que realiza medições de intensidade luminosa. Ele pode detectar de 0 lx até 100000 lx e possui uma resolução de 1 lux no modo de alta resolução;

CCS811: é uma plataforma que integra um sensor óxido de metal (MOX) para detectar uma grande quantidade de componentes orgânicos voláteis para monitorar a qualidade do ar em ambientes internos. A partir destes dados, ou seja,

dos componentes voláteis (VOCs), pode-se inferir a concentração de CO₂. Ele possui capacidade de determinar concentrações de dióxido de carbono entre 400 e 8192 ppm; e

INA219: é um “amperímetro” e “voltímetro” para corrente contínua. Ele pode verificar tensões entre 0 e 26 V e a corrente de -3,2 a + 3,2 A, com uma resolução de 0,8 mA.

Como o LaRaC possui um sistema com nobreak, bastante estável (CASTRO, 2018), não se achou necessário (pelo menos inicialmente) acompanhar estes parâmetros.

A implementação do sensor para a determinação da concentração de CO₂ na Estação de Monitoramento ficou comprometida. O problema foi a pouca disponibilidade de sensores de baixo custo para a determinação da concentração de dióxido de carbono disponíveis no mercado, principalmente no Brasil. Por isso, a sua aquisição envolvia uma importação, que demorou significativamente para ocorrer, impedindo a sua implementação na Estação de Monitoramento. Além disso, como o LaRaC já dispunha de um equipamento para a medição da concentração de CO₂, a sua não utilização na Estação de Monitoramento não comprometeu os resultados a serem obtidos.

É conveniente lembrar que a utilização de vários equipamentos distintos (microcomputador para controle do goniômetro, espectrorradiômetro, Estação de Monitoramento, equipamento para a medição da concentração de CO₂, etc.) dificulta a sincronização das medições e a eventual identificação de um problema, principalmente os que ocorrem em um intervalo de tempo muito pequeno.

Deste modo, dadas às condições de contorno apresentadas, a Estação de Monitoramento foi, inicialmente, elaborada com os sensores de parâmetros ambientais: DHT22; BMP180; e BH1750. A partir disto, foram obtidas as bibliotecas e códigos (programas) de exemplo de cada sensor. Com isto, cada sensor foi integrado na IDE do ARDUINO e testado independentemente. Depois, acrescentando um por vez, todos os sensores foram testados conjuntamente.

Além dos sensores para a determinação dos parâmetros ambientais foi necessário utilizar outros sensores e módulos eletrônicos para atender aos demais requisitos, ver Seção 2.5, da Estação de Monitoramento. Entre estes sensores e módulos podemos citar: um sensor magnético (magnetômetro) HMC5883 (HONEYWELL, 2013); um RTC (do inglês, *real time*

clock) PCF856 (NXP, 2015); um *display* LCD de 2 linhas e 16 caracteres por linha; e um ethernet *shield* (WIZNET, 2008).

As principais características de cada um destes sensores e módulos são:

HMC5883: é um magnetômetro, ou seja, um sensor magnético de 3 eixos, geralmente utilizado para medir o campo magnético da Terra, e fornecer a orientação para celulares e *tablets*. No nosso caso, a sua resolução (e os limites máximos e mínimos) das medições dos campos magnéticos não são significativos, visto que ele é utilizado (de uma maneira menos nobre, mas importante) para verificar se a porta do Laboratório foi aberta durante as medições;

PCF856: o RTC é um relógio que, quando sincronizado com o relógio de um microcomputador (como o que controla o goniômetro, por exemplo) fornece a mesma base de tempo, permitindo/facilitando a sincronização entre as medições da Estação de Monitoramento e as medições/informações dos outros equipamentos; e

Ethernet *shield*: O ethernet *shield* W5100 permite que o ARDUINO seja conectado na rede ethernet e, assim, disponibilize *on line* os dados obtidos na rede do IEAv, permitindo que se acompanhe remotamente as condições do laboratório. Além disso, o ethernet *shield* possui um *slot* para cartão SD, que permite a gravação dos dados.

De maneira semelhante à realizada com os sensores de parâmetros ambientais, foram obtidas as bibliotecas e códigos (programas) de exemplo de cada módulo eletrônico. Cada módulo foi integrado na IDE do ARDUINO e testado independentemente. Depois, acrescentando um por vez, todos os módulos foram testados conjuntamente.

Durante os testes verificou-se que a memória do ARDUINO UNO, de 32 kB, não era suficiente para incluir toda a programação, ou seja: código e bibliotecas, de todos os sensores e todos os módulos. Por isso, para o restante do desenvolvimento da Estação de Monitoramento, utilizou-se um ARDUINO MEGA, com memória de 256 kB.

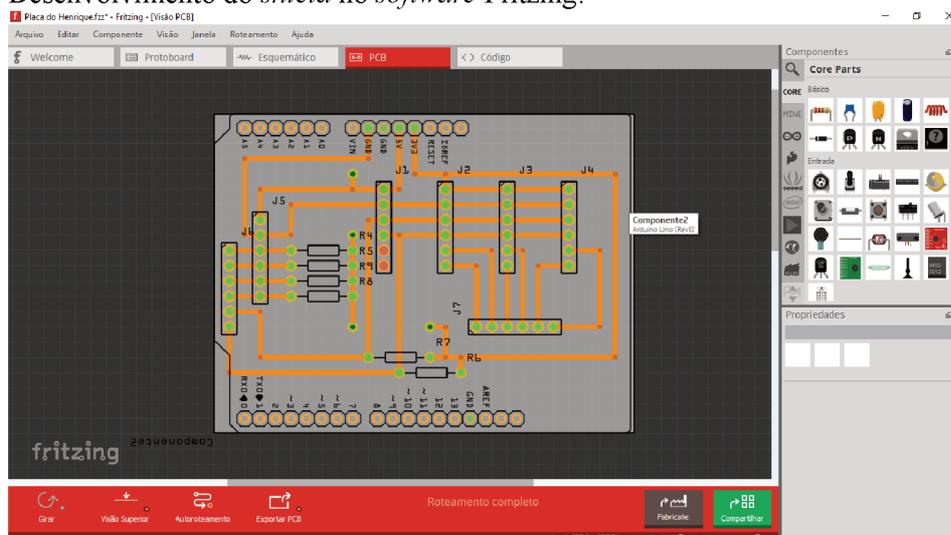
No início do código são adicionadas as bibliotecas necessárias para o funcionamento dos sensores e tela de cristal líquido. Em seguida cada componente de hardware é ativado, sensores e LCD (do inglês, *Liquid Crystal Display*) e algumas variáveis são criadas. Então, dentro do *Loop*, que é a parte do programa que irá se repetir indefinidamente, são realizadas as medições de cada sensor e outras variáveis adicionais são criadas. Os dados mais relevantes

são exibidos na tela de cristal líquido e as informações são salvas no cartão SD. Por fim, o programa calcula quanto tempo levou para realizar as todas as medições e gera um *delay* para que as próximas leituras se iniciem exatamente no tempo desejado.

3.3 ARDUINO MEGA E SHIELD

Depois dos testes iniciais a Estação foi desenvolvida tendo como plataforma o ARDUINO MEGA 2560. Durante os testes, também foi verificada a necessidade de distribuir os sensores em diversos pontos do LaRaC. Portanto, para facilitar a conexão dos sensores nos diversos pontos do laboratório, foi elaborado um *shield*, que foi desenvolvida no *software* FRITZING e produzido no IEAv, ver Figura 13. Também foram desenvolvidas e confeccionadas placas para a distribuição dos sensores nos vários pontos do LaRaC.

Figura 13 - Desenvolvimento do *shield* no *software* Fritzing.



Fonte: O Autor.

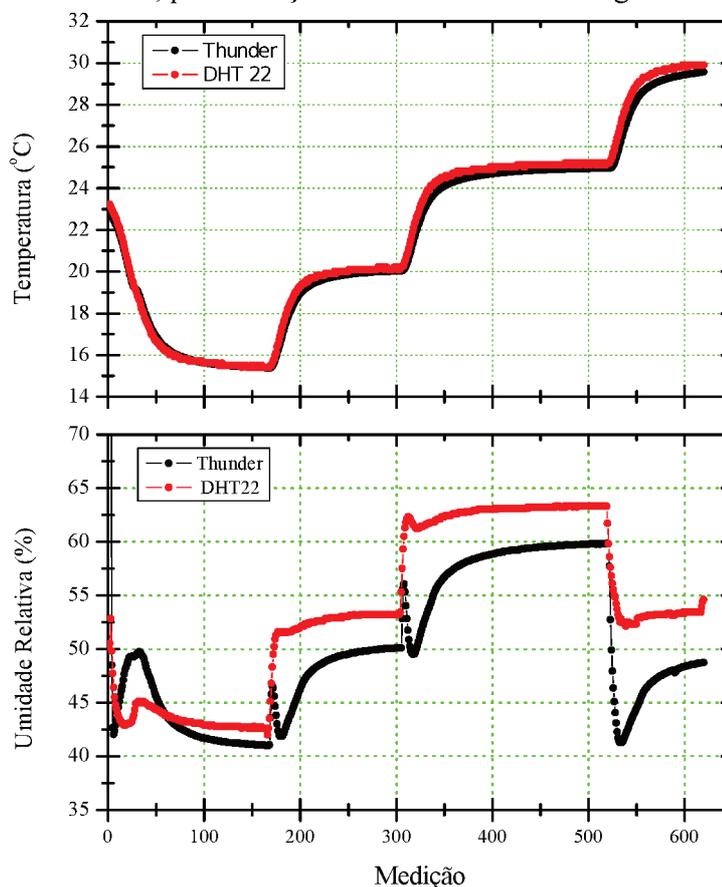
Após os testes isolados dos sensores, módulos, ethernet *shield* e o *shield* produzido, passou-se para a etapa de integração de todos estes componentes e o desenvolvimento do *software* específico da Estação de Monitoramento. Este desenvolvimento demandou a maior parte do tempo do projeto devido à utilização: do protocolo de comunicação I²C que requer um tipo particular de desenvolvimento do código para cada sensor; de muitos sensores; e do ethernet *shield*.

Finalizada a etapa do desenvolvimento do software, iniciou-se a sua fixação no laboratório e sua utilização.

4 RESULTADOS

Como comentado na Secção 3.2 cada sensor foi testado individualmente. Além disso, sempre que possível, os sensores utilizados neste projeto foram testados e comparados com equipamentos calibrados e de maior precisão para verificar as informações fornecidas pelos *datasheets*. Por exemplo, na Figura 14, pode-se verificar a caracterização de temperatura e da umidade relativa do sensor, DTH22, que foi realizada (MELO, 2016) utilizando um gerador de umidade, modelo Thunder 2500 do fabricante Thunder Scientific Corporation, que é o padrão utilizado para realizar a calibração de sensores de umidade na Divisão de Confiabilidade Metrológica Aeroespacial do Instituto de Fomento e Coordenação Industrial (IFI). No caso do sensor DTH22, a diferença absoluta de temperatura foi aproximadamente 0,3 °C e de 3 % para a umidade relativa, quando o sensor está em “equilíbrio” com ambiente, nas temperaturas e umidades aceitáveis para um laboratório. De maneira geral todos os sensores demonstraram-se confiáveis (ou seja, mesmo que valores absolutos não sejam precisos, o comportamento apresentado era o mesmo de um sensor mais confiável) para os fins de monitoramento de um ambiente de laboratório.

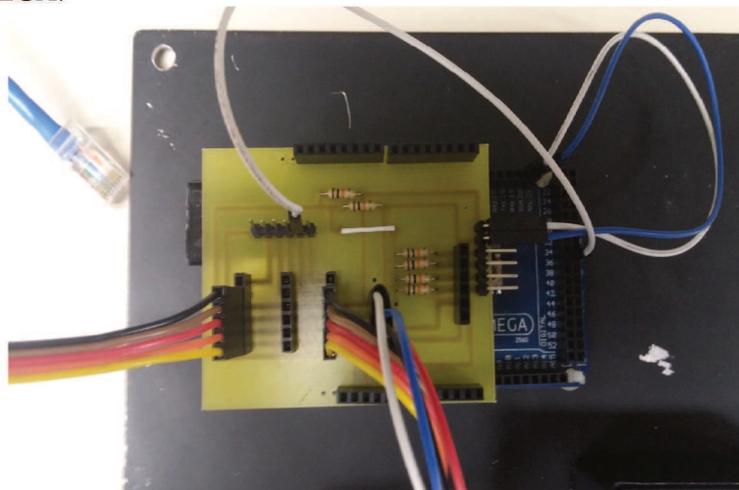
Figura 14 - Gráfico da temperatura e umidade relativa de um sensor DTH22 e no gerador de umidade, modelo Thunder 2500, para medições realizadas a cada 30 segundos.



Fonte: Melo (2016).

Depois de encerrados os testes com cada sensor foi efetuada a integração entre os diversos componentes da Estação de Monitoramento. Na Figura 15 pode ser observada a integração do ARDUINO MEGA, ethernet *shield* e o *shield* desenvolvido neste trabalho.

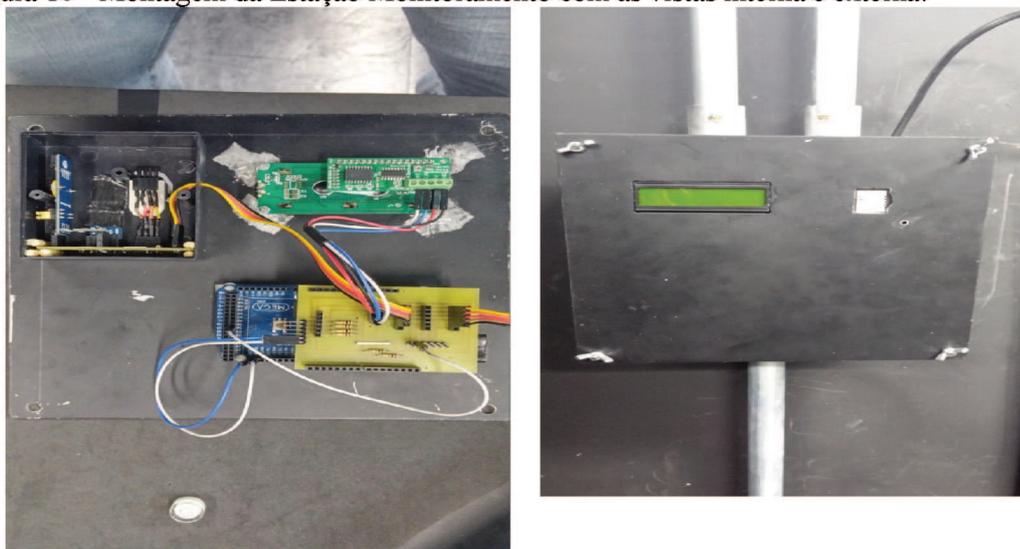
Figura 15 - Placa desenvolvida e conectada ao ethernet *shield*, que por sua vez estava conectado a um ARDUINO MEGA.



Fonte: O Autor.

Em seguida a Estação de Monitoramento foi inserida em uma caixa fixada na parede do laboratório. Nesta caixa foi colocada uma linha de 127 V, proveniente do *nobreak*, que permitia o funcionamento ininterrupto da Estação de Monitoramento, por meio conversor de tensão de 127 V AC para 12 V DC. Uma imagem da montagem, ainda provisória, da Estação pode ser observada na Figura 16.

Figura 16 - Montagem da Estação Monitoramento com as vistas interna e externa.



Fonte: O Autor.

A partir da integração dos sensores à Estação, foram efetuadas diversas séries de medições. Um exemplo destas medições pode ser observado na Figura 17 (RUBIATTI, 2017). O código desenvolvido pode ser observado no Apêndice A.

Figura 17 - Monitor serial da IDE do ARDUINO exibindo os dados obtidos pelos sensores.

```

final
// tudo

#include <Wire.h>
#include <SPI.h>
#include <Ethernet.h>
#include <EthernetUdp.h>
#include "Dns.h"

// Cartão SD
#include <SD.h>
// Seleciona o chipSelect do cartão SD - Selecionei o 10 e funciona
const int chipSelect = 4;
File dataFile;

// RTC, based on millis() + timer
#include "RTClib.h"
RTC_DS1307 rtc;

// Inclui o I2C
#include <LiquidTWI.h>
LiquidTWI ltwi(0); // Connect via I2C, default address #0 (A0-A2 not jumpered)

// Inclui o sensor de luz
#include <BH1750.h>
BH1750 lightMeter;
#define BH1750_I2CADRESS 0x23

// Inclui o barômetro - BMP180
#include <BMP180.h>
BMP180 barometer;
// Indica a pressão no nível do mar em Pascais.
float seaLevelPressure = 101325;

// Inclui o sensor de temperatura e umidade A e B - DHT22
#include <DHT.h>
DHT DHTs;
// ...

```

```

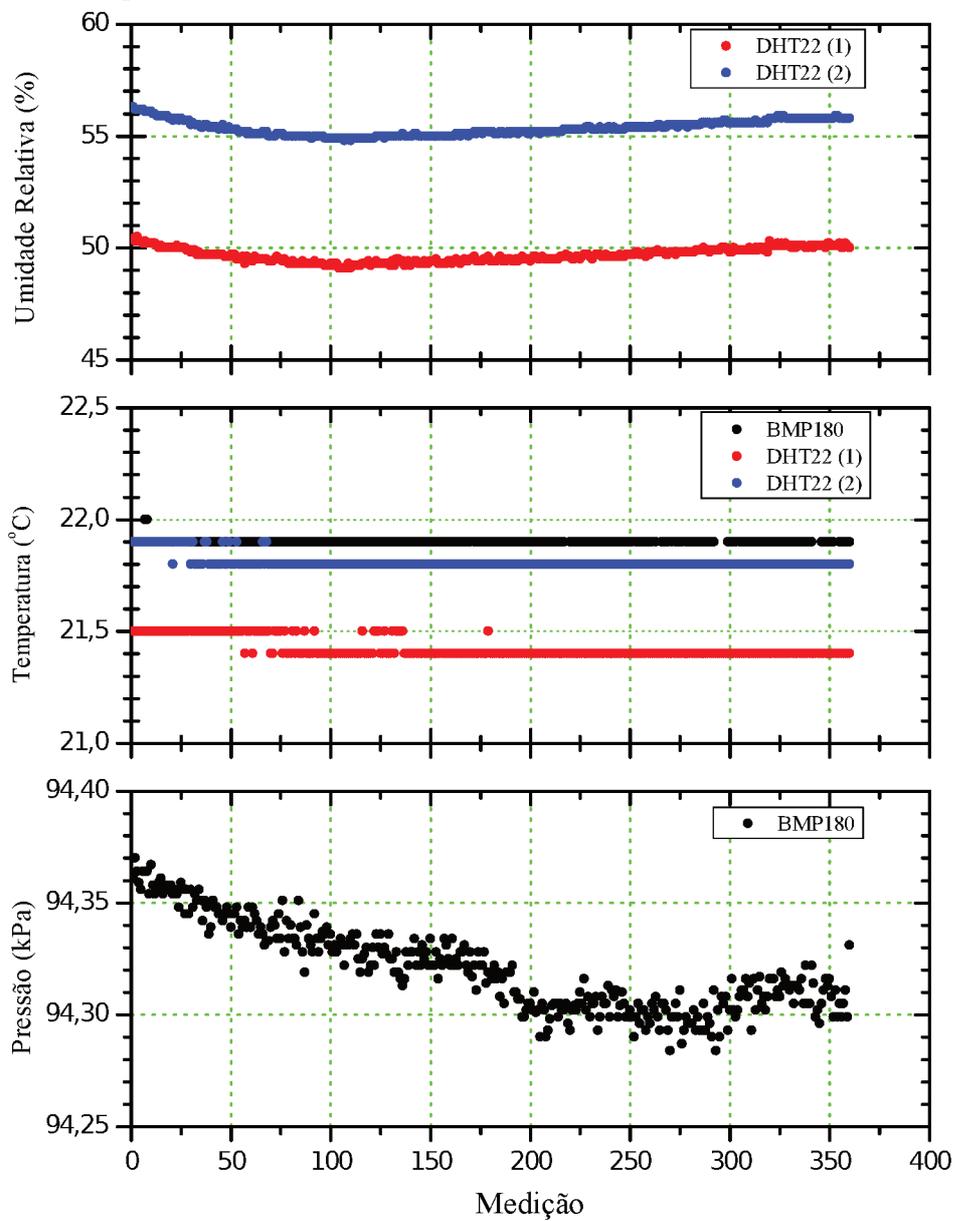
COM6 (Arduino Mega or Mega 2560)
Início
Início do Monitor
Início do LCD
Início do SD
Início do sensor de luz
Início do barômetro
Início do Magnetometro
Início do sensor de temperatura e umidade
DNS resolve.
ns3ch.forthnet.gr - 199.92.150.3
Seconds since Jan 1 1900 = 3655645979
Unix time = 1466651179
2015/11/04 14:13:01 2820 0 93510 23.10 22.90 58.10 NAN NAN -999.00 -1 2902
2015/11/04 14:13:02 3820 0 93512 23.10 23.00 58.20 NAN NAN -999.00 -1 2901
2015/11/04 14:13:03 4820 0 93519 23.10 23.00 58.20 NAN NAN -999.00 -1 4079
2015/11/04 14:13:04 5820 0 93512 23.10 22.90 58.00 NAN NAN -999.00 -1 5901
2015/11/04 14:13:05 6820 0 93509 23.10 22.90 58.00 NAN NAN -999.00 -1 6878
2015/11/04 14:13:06 7820 0 93515 23.10 22.90 58.10 NAN NAN -999.00 -1 7901
2015/11/04 14:13:07 8820 0 93512 23.10 22.90 58.10 NAN NAN -999.00 -1 8879
2015/11/04 14:13:08 9820 0 93515 23.10 22.90 58.00 NAN NAN -999.00 -1 9902
2015/11/04 14:13:09 10821 0 93509 23.10 22.90 58.00 NAN NAN -999.00 -1 10880
2015/11/04 14:13:10 11821 0 93514 23.10 22.90 58.10 NAN NAN -999.00 -1 11901
2015/11/04 14:13:11 12822 0 93519 23.10 22.90 58.10 NAN NAN -999.00 -1 12879
2015/11/04 14:13:12 13822 0 93508 23.10 22.90 57.90 NAN NAN -999.00 -1 13903
2015/11/04 14:13:13 14822 0 93505 23.10 22.90 57.90 NAN NAN -999.00 -1 14880
2015/11/04 14:13:14 15821 0 93511 23.10 23.00 57.90 NAN NAN -999.00 -1 15902
2015/11/04 14:13:15 16821 0 93509 23.10 23.00 57.90 NAN NAN -999.00 -1 16879
2015/11/04 14:13:16 17821 0 93506 23.10 22.90 57.90 NAN NAN -999.00 -1 17902
2015/11/04 14:13:17 18822 0 93516 23.10 22.90 57.90 NAN NAN -999.00 -1 18880
2015/11/04 14:13:18 19822 0 93512 23.10 22.90 57.90 NAN NAN -999.00 -1 19904
2015/11/04 14:13:19 20823 0 93514 23.10 22.90 57.90 NAN NAN -999.00 -1 20900
2015/11/04 14:13:20 21824 0 93503 23.10 22.90 58.00 NAN NAN -999.00 -1 21905
2015/11/04 14:13:21 22824 0 93509 23.10 22.90 58.00 NAN NAN -999.00 -1 22983
2015/11/04 14:13:22 23824 0 93516 23.10 22.90 57.90 NAN NAN -999.00 -1 23905
2015/11/04 14:13:23 24824 0 93515 23.10 22.90 57.90 NAN NAN -999.00 -1 24983
2015/11/04 14:13:24 25824 0 93511 23.00 22.90 58.00 NAN NAN -999.00 -1 25906
2015/11/04 14:13:25 26824 0 93505 23.00 22.90 58.00 NAN NAN -999.00 -1 26982
2015/11/04 14:13:26 27825 0 93506 23.00 22.90 58.10 NAN NAN -999.00 -1 27906
2015/11/04 14:13:27 28825 0 93515 23.00 22.90 58.10 NAN NAN -999.00 -1 28883
2015/11/04 14:13:28 29826 0 93506 23.00 22.90 58.30 NAN NAN -999.00 -1 29906
2015/11/04 14:13:29 30826 0 93509 23.00 22.90 58.30 NAN NAN -999.00 -1 30884
2015/11/04 14:13:30 31826 0 93512 23.00 22.90 58.10 NAN NAN -999.00 -1 31907

```

Fonte: O Autor.

Com isto a reprodutibilidade das medições da Estação de Monitoramento foi testada em diferentes locais do laboratório. Na Figura 18 pode-se observar os resultados de temperatura, pressão e umidade, para uma série de medição de aproximadamente 3 h, com medições realizadas a cada 30 s. Como podemos observar, o comportamento dos dados é bastante estável, com uma variação de aproximadamente: 2% na umidade relativa; 1°C na temperatura; e 100 Pa na pressão atmosférica. Vale lembrar que estas medições foram realizadas simultaneamente as medições do fator de reflectância com o goniômetro. Com isto, estes valores indicam, conforme esperado, que as condições ambientais do laboratório são bastante estáveis.

Figura 18 - Gráfico de parâmetros ambientais obtidos com os sensores da Estação de Monitoramento.



Fonte: O Autor

5 CONCLUSÃO

Levando em consideração os objetivos gerais do trabalho, ver Capítulo 1, foi construída uma Estação de Monitoramento, que realiza medições, a cada minuto, da temperatura, umidade, iluminação, em diversos pontos estratégicos do laboratório, registra o horário das medições, indica as informações mais relevantes em uma tela de cristal líquido e grava os dados obtidos em um cartão SD. Além disso, conseguiu-se entender o funcionamento do microcontrolador ARDUINO, bem como do barramento I²C.

O ARDUINO mostrou-se muito eficaz quando se leva em consideração seu preço, facilidade de se trabalhar e funcionalidade. Há uma grande comunidade de usuários que constantemente compartilham experiências e estão sempre dispostos a ajudar.

O protocolo de comunicação I²C possibilitou a implementação de um número maior de sensores, embora tenha aumentado a complexidade do código e não tenha suporte para a instalação de mais de 1 ou 2 sensores iguais. Este problema abre caminho para que novos estudos sejam realizados a fim de verificar a possível implementação de um multiplexador para tornar possível a utilização de um número maior de sensores que utilizam o mesmo endereço.

Obtivemos mais experiência com os *softwares* utilizados possibilitando que melhorias venham a ser feitas, tanto no código, quanto na placa que foi desenvolvida no FRITZING.

Em síntese, a Estação de Monitoramento tem apresentado resultados que vem sendo úteis aos usuários do LaRaC. Entretanto ela ainda pode ser melhorada abrindo caminho para a automatização do monitoramento das condições ambientes de vários locais e laboratórios. Com a criação de uma interface mais amigável ela pode se tornar mais simples e atrativa de se utilizar, possibilitando sua implementação onde houver necessidade.

Referências

- AMS, CSS811: *Ultra-Low Power Digital Gas Sensor for Monitoring Indoor Air Quality*, AMS, 2018, Disponível em: <https://ams.com/documents/20143/36005/CCS811_DS000459_6-00.pdf/c7091525-c7e5-37ac-eedb-b6c6828b0dcf>, Acessado em 01/11/2018.
- Andrade, L.L.; Galvão, L. S.; Castro, R. M. Conversão dos números digitais de imagens TIR do sensor HSS para radiação e estimativa da temperatura e emissividade. *Revista Brasileira de Cartografia*, n.64 v.4, p.487-502, 2012. ISSN: 1808-0936.
- AOSONG, *Temperature and humidity module - AM2302 Product Manual*, Disponível em <http://img.filipeflop.com/files/download/Datasheet_DHT22_AM2302.pdf>, Acessado em 01/11/2018.
- Aureliano, A., *Microcontroladores*, FIOZERA, 2017. Disponível em: <<https://fiozera.com.br/microcontroladores-914a59cbf7de>>, Acessado em: 01/11/2018.
- ARDUINO, Disponível em: <<https://www.arduino.cc/en/Guide/Introduction>>, Acessado em 01/11/2018.
- ARDUINOBR, I2C – Protocolo de Comunicação I²C – Protocolo de Comunicação, Disponível em: <http://www.arduino.br/arduino/i2c-protocolo-de-comunicacao>, Acesso em: 15/09/2017.
- BMP180, *BMP180: Digital Pressure Sensor Datasheet*, BOSCH, 2015, Disponível em: <https://ae-bst.resource.bosch.com/media/_tech/media/datasheets/BST-BMP180-DS000-12.pdf>, Acessado em 01/11/2018.
- Câmara, R. C. P., Protocolo I2C, Disponível em: <<http://univasf.edu.br/~romulo.camara/novo/wp-content/uploads/2013/11/Barramento-e-Protocolo-I2C.pdf>>. Acesso em: 01/11/2018.
- CASTRO, R. M.; Teixeira Jr, J. F.; Rubiatti, H. T.; Silva, M. M.; Silva, B. A.; Andrade, L. L.; Melo, M. C.; Barros, E. A. *Goniômetro: Especificações Gerais*, Comunicação privada, IEAv, 2015.
- CASTRO, R. M.; Teixeira Jr, J. F.; Andrade, L. L.; Rubiatti, H. T.; Melo, M. C.; Marinho, R. S.; Lauria, P. B.; Buarque, B. N. M. C.; Silva, M. M.; Marques, K. C. M.; Garagorry, A. B.; Barros, E. A.; Silva, B. A., *Goniômetro para medições de BRDF e emissividade de materiais de interesse em SR*. In: XVIII Simpósio Brasileiro de Sensoriamento Remoto, 2017, Santos. Anais do XVIII Simpósio Brasileiro de Sensoriamento Remoto. São José dos Campos: INPE, 2017.
- CASTRO, R. M. *LaRaC: Estabilidade do sistema de Nobreak*, Comunicação privada, IEAv, 2018.
- CYCLOAR, A Umidade Relativa e seu comportamento em função da temperatura, CYCLOAR, 2018, Disponível em: <<http://www.cycloar.ind.br/agricola/umidade.htm>>, Acessado em 01/11/2018.

- HONEYWELL, HMC 5883L: 3-Axis Digital Compass IC, Honeywell International Inc, 2013, Disponível em: <https://cdn-shop.adafruit.com/datasheets/HMC5883L_3-Axis_Digital_Compass_IC.pdf>, Acessado em 01/11/2018.
- JENSEN, J. R., *Sensoriamento remoto do ambiente: uma perspectiva em recursos terrestres*. São José dos Campos: Parêntese, 2009.
- LabVIEW, *Ambiente gráfico de desenvolvimento de sistemas LabVIEW*, National Instruments, 2017. Disponível em: <http://www.ni.com/labview/pt/>, acessado em 20/11/2017.
- McROBERTS, M., *Arduino Básico*, Editora Novatec, 2015.
- MELO, M. C., *Calibração do Sensor DTH22*, Comunicação Privada, IFI, 2016.
- MICROCONTROLLER. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2018. Disponível em: <<https://en.wikipedia.org/wiki/Microcontroller>>. Acesso em: 01/11/2018.
- NI845X, *NI 845X OEM: SPECIFICATIONS*. Disponível em: <<http://www.ni.com/pdf/manuals/290598a.pdf>>. Acesso em: 08/11/2017.
- NOAA, *A Global Network for Measurements of Greenhouse Gases in the Atmosphere: Trends in Atmospheric Carbon Dioxide*, NOAA, 2018. Disponível em: <<https://www.esrl.noaa.gov/gmd/ccgg/trends/global.html>>, Acessado em: 10/11/2018.
- NXP, *PCF8563: Real-time clock/calendar*, NXP Semiconductors, 2015, Disponível em: <www.nxp.com/documents/data_sheet/PCF8563.pdf>, Acessado em 01/11/2018.
- ROHM, *BH1750 Digital Light Sensor*, ROHM Co., 2010, Disponível em: <<https://img.filipeflop.com/files/download/Datasheet-bh1750fvi.pdf>>, Acessado em 01/11/2018.
- RUBIATTI, H. T., *Projeto de IC: Desenvolvimento de Sistema para Controle de Reprodutibilidade do Goniômetro e de Parâmetros Ambientais*, CGPIBICTI-IEAv, 2016.
- RUBIATTI, H. T., *Relatório Final de Atividades: Desenvolvimento de Sistema para Controle de Reprodutibilidade do Goniômetro e de Parâmetros Ambientais*, CGPIBICTI-IEAv, 2017.
- SERIAL COMMUNICATION. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2018. Disponível em: <https://en.wikipedia.org/wiki/Serial_communication>. Acesso em: 01/11/2018.
- TEXAS INSTRUMENTS, INA219 Zero-Drift, Bidirectional Current/Power Monitor With I2C Interface, Texas Instruments, 2015, Disponível em: <<http://www.ti.com/lit/ds/symlink/ina219.pdf>>, Acesso em: 01/11/2018.
- WIZNET, *W5100 Datasheet*, WIZnet Co., 2008 Disponível em: <https://img.filipeflop.com/files/download/Datasheet_W5100_v1_1_6.pdf>, Acesso em 01/11/2018.

APÊNDICE A

Software desenvolvido no IDE do ARDUINO para o controle da Estação de Monitoramento.

```
// Estacao de Monitoramento
// LaRaC -

#include <Wire.h>
#include <SPI.h>
#include <Ethernet.h>
#include <EthernetUdp.h>
#include "Dns.h"

// Cartao SD
#include <SD.h>
// Seleciona o chipSelect do cartão SD - Selecionei o 10 e funciona
const int chipSelect = 4;
File dataFile;

// RTC, based on millis() & timer
#include "RTCLib.h"
RTC_Millis rtc;

// Inclui o LCD
#include <LiquidTWI.h>
LiquidTWI lcd(0); // Connect via i2c, default address #0 (A0-A2 not jumpered)

// inclui o sensor de luz
#include <BH1750.h>
BH1750 lightMeter;
#define BH1750_I2CADDRa 0x23

// inclui o barometro - BMP180
#include <BMP180.h>
BMP180 barometer;
// Indica a pressao no nível do mar em Pascals.
float seaLevelPressure = 101325;

// inclui o sensor de temperatura e umidade A e B - DHT22
#include <DHT.h>
DHT DHTa;
DHT DHTb;

// Inclui o magnetometro - HMC5983
#include <HMC5983.h>
HMC5983 compass;
#define HMC5983_ADDRESS 0x1E

// configuracoes do Ethernet shield para acessar o UDP
// Enter a MAC address for your controller below.
byte mac[] = { 0x00, 0xAA, 0xBB, 0xCC, 0xDE, 0x02 };
unsigned int localPort = 80; // local port to listen for UDP packets
IPAddress timeServer(193,92,150,3); // time.nist.gov NTP server (fallback)
const int NTP_PACKET_SIZE= 48; // NTP time stamp is in the first 48 bytes
of the message
byte packetBuffer[ NTP_PACKET_SIZE]; // buffer to hold incoming and outgoing
packets
const char* host = "nsath.forthnet.gr"; // Use random servers through DNS
// A UDP instance to let us send and receive packets over UDP
EthernetUDP Udp;
DNSClient Dns;
IPAddress rem_add;

void setup()
```

```
{
// inicio do monitor
Serial.begin(9600);
while (!Serial) { ; }
Serial.println("Inicio");
Serial.println("Inicio do Monitor");

// Inicia o LCD
lcd.begin(16, 2);          // número de colunas e linhas do LCD
lcd.setBacklight(HIGH);
delay(500);
lcd.print("Funcionando");
lcd.setBacklight(LOW);
delay(500);
Serial.println("Inicio do LCD");

// inicia do cartao sd
Serial.println("Inicio do SD");
if (!SD.begin(chipSelect)) {
  Serial.println("Card failed, or not present");
  // don't do anything more:
  while(1);
}

// Inicia o sensor de luz - BH1750
Serial.println("Inicio do sensor de luz");
lightMeter.begin();

// Inicia o barometro - BMP180.
Serial.println("Inicio do barometro");
barometer = BMP180();
// When we have connected, we reset the device to ensure a clean start.
barometer.SoftReset();
// Now we initialize the sensor and pull the calibration data.
barometer.Initialize();

// Inicia o Magnetometro - HMC5983
Serial.println("Inicio do Magnetometro");
compass.begin();

// Inicia os sensor A e B de temperatura e umidade
Serial.println("Inicio do sensor de temperatura e umidade");
DHTa.setup(36); // data pin 2
DHTb.setup(7); // data pin 7

// variaveis para a data e tempo
int anox;
int mesx;
int diax;
int horx;
int minx;
int segx;

// Inicia a Ethernet e o UDP
if (Ethernet.begin(mac) == 0) {
  Serial.println("Failed to configure Ethernet using DHCP");
  // no point in carrying on, so do nothing forevermore:
  while(true);
}
Udp.begin(localPort);
Dns.begin(Ethernet.dnsServerIP() );

if(Dns.getHostByName(host, rem_add) == 1 ){
  Serial.println("DNS resolve...");
  Serial.print(host);
  Serial.print(" = ");
  Serial.println(rem_add);
}
```

```

    sendNTPpacket(rem_add);
}else{
    Serial.print("DNS fail...");
    Serial.print("time.nist.gov = ");
    Serial.println(timeServer); // fallback
    sendNTPpacket(timeServer); // send an NTP packet to a time server
}
delay(1000); // wait to see if a reply is available
if ( Udp.parsePacket() ) {
    // We've received a packet, read the data from it
    Udp.read(packetBuffer,NTP_PACKET_SIZE); // read the packet into the buffer

    //the timestamp starts at byte 40 of the received packet and is four bytes,
    // or two words, long. First, extract the two words:
    unsigned long highWord = word(packetBuffer[40], packetBuffer[41]);
    unsigned long lowWord = word(packetBuffer[42], packetBuffer[43]);
    // combine the four bytes (two words) into a long integer
    // this is NTP time (seconds since Jan 1 1900):
    unsigned long secsSince1900 = highWord << 16 | lowWord;
    Serial.print("Seconds since Jan 1 1900 = " );
    Serial.println(secsSince1900);

    // now convert NTP time into everyday time:
    Serial.print("Unix time = ");
    // Unix time starts on Jan 1 1970. In seconds, that's 2208988800:
    const unsigned long seventyYears = 2208988800UL;
    // subtract seventy years:
    unsigned long epoch = secsSince1900 - seventyYears;
    // print Unix time:
    Serial.println(epoch);
    DateTime future (epoch + 2 - (3600*3));
    anoX = (future.year());
    mesX = (future.month());
    diaX = (future.day());
    horX = (future.hour());
    minX = (future.minute());
    segX = (future.second());
}

// inicia o rtc
rtc.adjust(DateTime(anoX, mesX, diaX, horX, minX, segX));
}

void loop()
{
    // cria variaveis para armazenar os dados a serem gravados:
    String dataString = ""; // para o serial e o SD
    String Lin0 = ""; // linha 0 do LCD
    String Lin1 = ""; // linha 1 do LCD
    long timei;
    long timef;
    int tempo;

    // le o tempo inicial de um void loop
    // sever para calcular o tempo de cada rotina
    timei = millis();

    // Medições com o RTC - lendo a data e horario e colcando em uma string
    DateTime now = rtc.now();
    int ano;
    int mes;
    int dia;
    int hhh;
    int mm;
    int sss;

```

```
//ano
ano = now.year();
dataString += ano;
dataString += "/";

// mes
mes = now.month();
if (mes < 10){ dataString += "0";};
dataString += mes;
dataString += "/";
if (mes < 10){ Lin0 += "0";};
Lin0 += mes;
Lin0 += "/";

// dia
dia = now.day();
if (dia < 10){ dataString += "0";};
dataString += dia;
dataString += " ";
if (dia < 10){ Lin0 += "0";};
Lin0 += dia;
Lin0 += " ";

// hora
hhh = now.hour();
if (hhh < 10){ dataString += "0";};
dataString += hhh;
dataString += ":";
if (hhh < 10){ Lin0 += "0";};
Lin0 += hhh;
Lin0 += ":";

// minutos
mmm = now.minute();
if (mmm < 10){ dataString += "0";};
dataString += mmm;
dataString += ":";
if (mmm < 10){ Lin0 += "0";};
Lin0 += mmm;
Lin0 += ":";

// segundos
sss = now.second();
if (sss < 10){ dataString += "0";};
dataString += sss;
dataString += " ";
if (sss < 10){ Lin0 += "0";};
Lin0 += sss;
Lin0 += " ";

// imprime o tempo inicial do void loop
dataString += timei;
dataString += " ";

// Imprime a data e hora na linha 0 do LCD
lcd.setCursor(0, 0);
lcd.print(Lin0);

// Medicao com o sensor de lux
uint16_t lux = lightMeter.readLightLevel();
if (lux < 10 ){ dataString += " ";};
if (lux < 100){ dataString += " ";};
dataString += lux;
dataString += " ";
if (lux < 10){ Lin1 += " ";};
```

```

    if (lux < 100){ Lin1 += " ";};
    Lin1 += lux;
    Lin1 += " ";

// medicao com o barometro -
// obtem o dado de pressao em Pa pascal
long BMP180Pressure = barometer.GetPressure();
dataString += BMP180Pressure;
dataString += " ";
// obtem o dado de temperatura em oC
float BMP180Temperature = barometer.GetTemperature();
dataString += BMP180Temperature;
dataString += " ";
Lin1 += BMP180Temperature;
Lin1 += " ";

// Medicao de temperatura e umidade A
// obtem a temperatura em oC
float dht22atemperature = DHTa.getTemperature();
dataString += dht22atemperature;
dataString += " ";
Lin1 += dht22atemperature;
Lin1 += " ";
// obtem o dado em umidade relativa
float dht22ahumidity = DHTa.getHumidity();
dataString += dht22ahumidity;
dataString += " ";
Lin1 += dht22ahumidity;
Lin1 += " ";

// Medicao de temperatura e umidade B
// obtem a temperatura em cC
float dht22btemperature = DHTb.getTemperature();
dataString += dht22btemperature;
dataString += " ";
// obtem o dado em umidade relativa
float dht22bhumidity = DHTb.getHumidity();
dataString += dht22bhumidity;
dataString += " ";

// Medicao com o Magnetometro
// angulo em relacao ao norte magnetico
float ccc = -999;
ccc = compass.read();
// intrucoes para a leitura do campo mag nos tres eixos do sensor
Wire.beginTransaction(HMC5983_ADDRESS);
Wire.write(0x02);
Wire.write(0x01);
Wire.endTransmission();
delay(6);
// leitura do campo mag nos tres eixos do sensor (6 bytes)
// get the six data bytes, MSB and LSB for X, Z, Y (in that order)
Wire.requestFrom(HMC5983_ADDRESS, 6);
//"The heading output data will be the value in tenths of degrees
//from zero to 3599 and provided in binary format over the two bytes."
byte Z_MSB = Wire.read();
byte Z_LSB = Wire.read();
long HZ = (Z_MSB << 8) + Z_LSB;
dataString += ccc;
dataString += " ";
dataString += HZ;
dataString += " ";

// le e imprime o tempo final do void loop
timef = millis();
dataString += timef;

```

```

// Imprime os dados na linha 1 do LCD
  lcd.setCursor(0, 1);
  lcd.print(Lin1);

// Escreve o resultado no Serial
  Serial.println(dataString);

// Escreve o resultado no arquivo
  dataFile.println(dataString);

// salva o dado no SD
//  dataFile.flush();

  // open the file. note that only one file can be open at a time,
  // so you have to close this one before opening another.
  File dataFile = SD.open("datalog.txt", FILE_WRITE);

  // if the file is available, write to it:
  if (dataFile) {
    dataFile.println(dataString);
    dataFile.close();
  }
  // if the file isn't open, pop up an error:
  else {
    Serial.println("error opening datalog.txt");
  }

// le o tempo depois de "todas" as funções"
  timef = millis();

// calcula o tempo para o delay
  tempo = 1000 - timef + timei + 0;
  if (tempo<50){tempo=1000;}
  delay(tempo);
}

// send an NTP request to the time server at the given address
unsigned long sendNTPpacket(IPAddress& address)
{
  // set all bytes in the buffer to 0
  memset(packetBuffer, 0, NTP_PACKET_SIZE);
  // Initialize values needed to form NTP request
  // (see URL above for details on the packets)
  packetBuffer[0] = 0b11100011; // LI, Version, Mode
  packetBuffer[1] = 0; // Stratum, or type of clock
  packetBuffer[2] = 6; // Polling Interval
  packetBuffer[3] = 0xEC; // Peer Clock Precision
  // 8 bytes of zero for Root Delay & Root Dispersion
  packetBuffer[12] = 49;
  packetBuffer[13] = 0x4E;
  packetBuffer[14] = 49;
  packetBuffer[15] = 52;

  // all NTP fields have been given values, now
  // you can send a packet requesting a timestamp:

  Udp.beginPacket(address, 123); //NTP requests are to port 123
  Udp.write(packetBuffer,NTP_PACKET_SIZE);
  Udp.endPacket();
}

```